Tencent Cloud Mini Program Platform

Guide to Mini Program Development

Product Documentation



©2013-2022 Tencent Cloud. All rights reserved.

Copyright Notice

©2013-2024 Tencent Cloud. All rights reserved.

Copyright in this document is exclusively owned by Tencent Cloud. You must not reproduce, modify, copy or distribute in any way, in whole or in part, the contents of this document without Tencent Cloud's the prior written consent.

Trademark Notice

All trademarks associated with Tencent Cloud and its services are owned by Tencent Cloud Computing (Beijing) Company Limited and its affiliated companies. Trademarks of third parties referred to in this document are owned by their respective proprietors.

Service Statement

This document is intended to provide users with general information about Tencent Cloud's products and services only and does not form part of Tencent Cloud's terms and conditions. Tencent Cloud's products or services are subject to change. Specific products and services and the standards applicable to them are exclusively provided for in Tencent Cloud's applicable terms and conditions.

Contents

- Guide to Mini Program Development
  - Mini program introduction and development environment
  - Mini program code composition
    - JSON Configuration
    - WXML Template
    - WXSS Style
    - JavaScript
  - Guidance
    - Learn about mini programs
      - Mini Program Host Environment
      - Operation mechanism
      - Code structure
      - Directory Structure
    - Configuring mini program
      - Global and Page Configuration
    - Mini Program Framework
      - Introduce
      - Logic Layer
      - View Layer
    - Custom Components
    - Basic competencies
      - Network
      - Storage
      - File System
      - Subpackage Loading
      - Canvas
      - Custom tabBar
      - Multi-threaded Worker
      - DarkMode Adaptation Guide
  - Framework
    - Mini program configuration
      - Project Configuration
      - Global Configuration
      - Page Configuration
    - Framework Interface

©2013-2022 Tencent Cloud. All rights reserved.

App
Page
Custom Components
modularity
Basic function
WXML Syntax Reference
Introduce
Data Binding
List Rendering
Conditional Rendering
Templates
Referencing
WXS Syntax Reference
Introduce
WXS Modules
Variable
Annotations
Operators
Statements
Data Types
Base Library
Components
Component Overview
Native Components
View Container
Fundamentals
Map Component
Media Component
Open Capabilities
Canvas
Navigation Component
Form Component
API
API Overview
Fundamentals
Routing
Redirect
Forwarding

©2013-2022 Tencent Cloud. All rights reserved.

- Interface
  - Window
  - Font
  - Interaction
  - Navigation Bar
  - Background
  - TabBar
  - Pull-down refresh
  - Scroll
  - Animation
  - Custom component
  - Menu
- Network
  - UDP Communication
  - mDNS
  - Initiate request
  - Upload
  - Websocket
  - Download
  - TCP Communication
- Data Caching
- Data Analysis
- canvas
  - Canvas Overview
  - OffscreenCanvas
  - Path2D
  - Canvas
  - canvasContext
  - Color
- Worker
- Media
  - rich text
  - Video
  - Audio recording
  - Background audio
  - Map
  - Real-time audio/video
  - Transparent Video

©2013-2022 Tencent Cloud. All rights reserved.

- Camera
- Image
- Audio

WXML

File

Device

- Contact person
- Vibration
- Battery Level
- Screen
- Memory
- Device Orientation
- Accessibility
- Network
- Keyboard
- Phone number
- Accelerometer
- Compass
- Gyroscope
- SMS
- Code scanning
- Encryption
- NFC—Ndef
- NFC—NfcA
- NFC—NfcB
- NFC—NfcF
- NFC—NfcV
- WiFi
- Calendar
- Clipboard
- NFC—getNFCAdapter
- NFC—NFCAdapter
- NFC—IsoDep
- NFC—MifareClassic
- NFC—MifareUltralight
- Bluetooth - Low Energy Central Device
- Bluetooth - Low Energy Peripheral Device
- Bluetooth - Beacon

©2013-2022 Tencent Cloud. All rights reserved.

Bluetooth - General
Location
Open Interfaces
Custom APIs
Sensitive API
JS SDK
Web-View
API description
Basic Library
Basic Library Introduction
Base library maintenance
IDE Operation Instructions
Overview
Interfaces
Settings
Code editing
Mini program debugging
Additional information

©2013-2022 Tencent Cloud. All rights reserved.

Guide to Mini Program Development
Mini program introduction and development environment

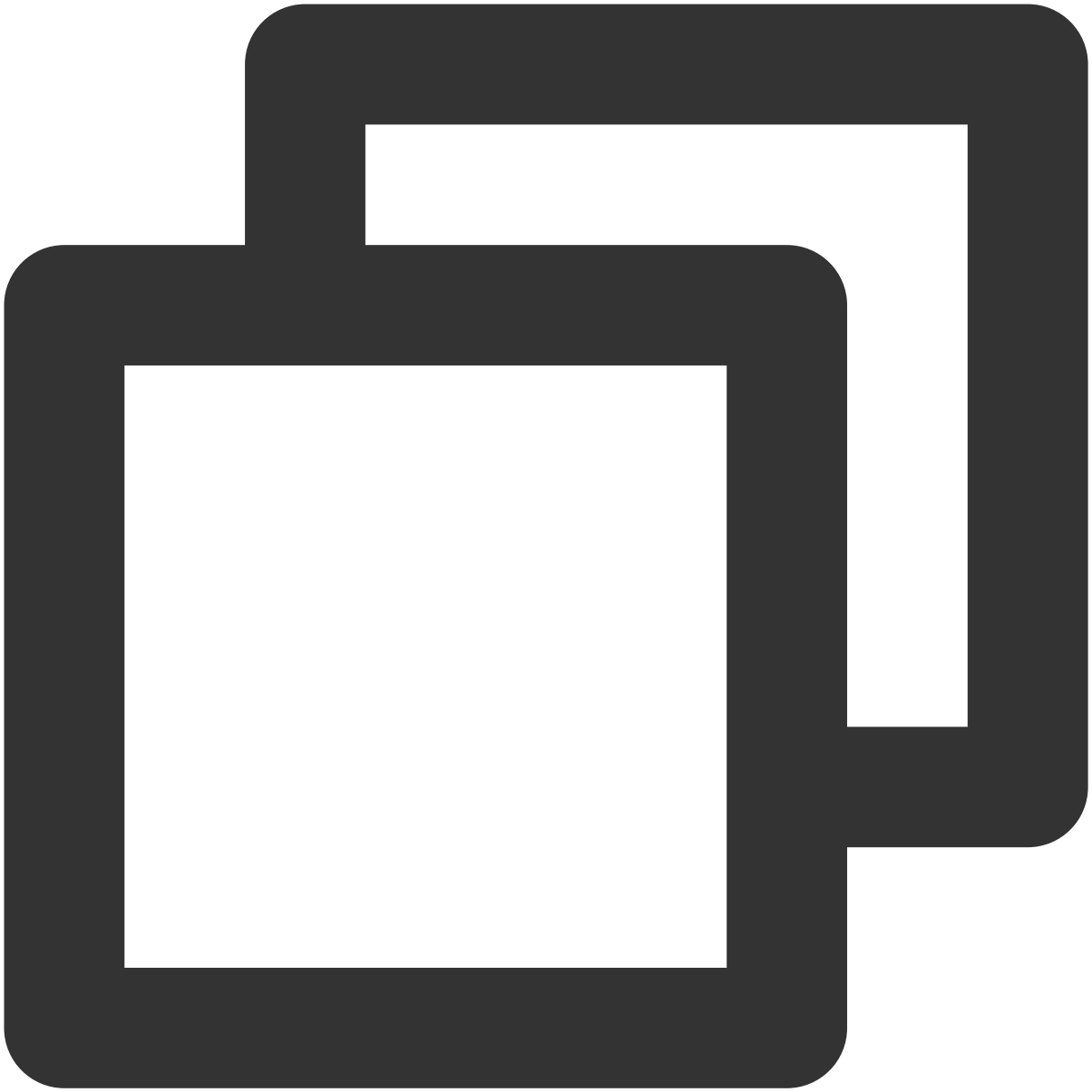
Last updated：2024-05-10 11:26:34

Mini programs are a new way to connect users and services. They can be easily accessed and distributed within an application, and provide an outstanding user experience. Developers can easily develop and publish a mini program after learning.

1. Hello World

Before introducing the history and technical details of mini programs, follow our steps to develop this Hello World.
Step 1: Go to the corresponding to your own operating system to download and install the installation package.
Step 2: Open the TCMPP developer tool and click New Mini program Project. Leave alone the AppID and select No AppID when you create a new project and uncheck the option of "Create general quick startup template".
Step 3: Add necessary code.
Create `app.json` in the root directory, which reads as follows:

©2013-2022 Tencent Cloud. All rights reserved.



```
  "pages": ["pages/index/index"]
}
```

Create a new pages directory under the root directory, then create a new index directory under the pages directory, and then create two files index.wxml and index.js under the index directory.

The content of index.wxml: <text>Hello World</text>

The content of index.js: Page({})

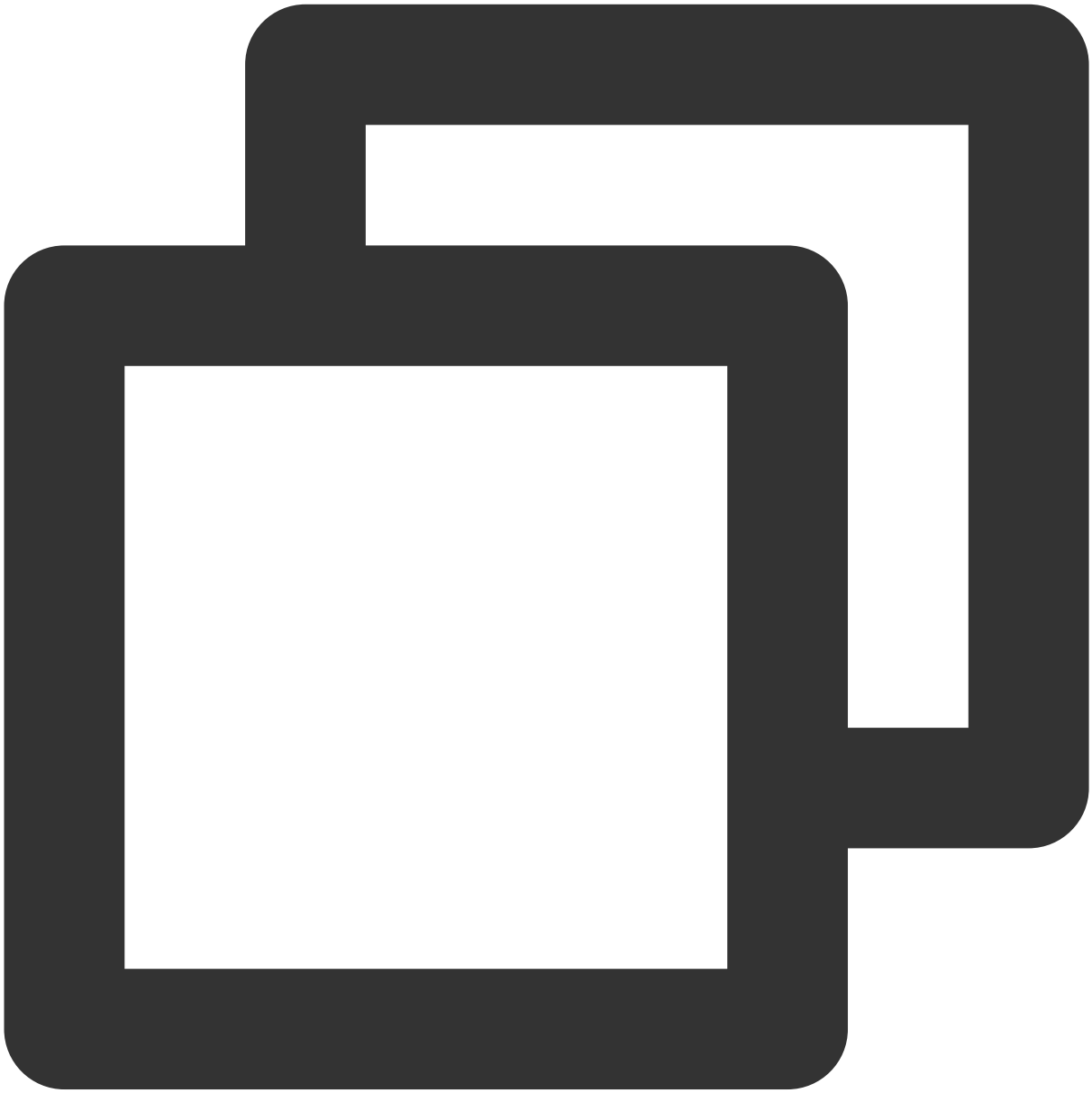
By writing these lines of code, Hello World appears in the simulator of the TCMPP developer tool.

©2013-2022 Tencent Cloud. All rights reserved.

2. Mini program introduction

2.1 History of mini program

From a technical perspective, mini programs are not a concept that comes out of nowhere. When WebView became an important entry point for the mobile Web, the JS API came into being.



```
WeixinJSBridge.invoke('imagePreview', {
    current: 'http://inews.gtimg.com/newsapp_bt/0/1693121381/641',
    urls: [ // URL list of all images, array format
        'https://img1.gtimg.com/10/1048/104857/10485731_980x1200_0.jpg',
```

©2013-2022 Tencent Cloud. All rights reserved.

```
            'https://img1.gtimg.com/10/1048/104857/10485726_980x1200_0.jpg',
            'https://img1.gtimg.com/10/1048/104857/10485729_980x1200_0.jpg'
        ]
    }, function(res) {
        console.log(res.err_msg)
    })
```

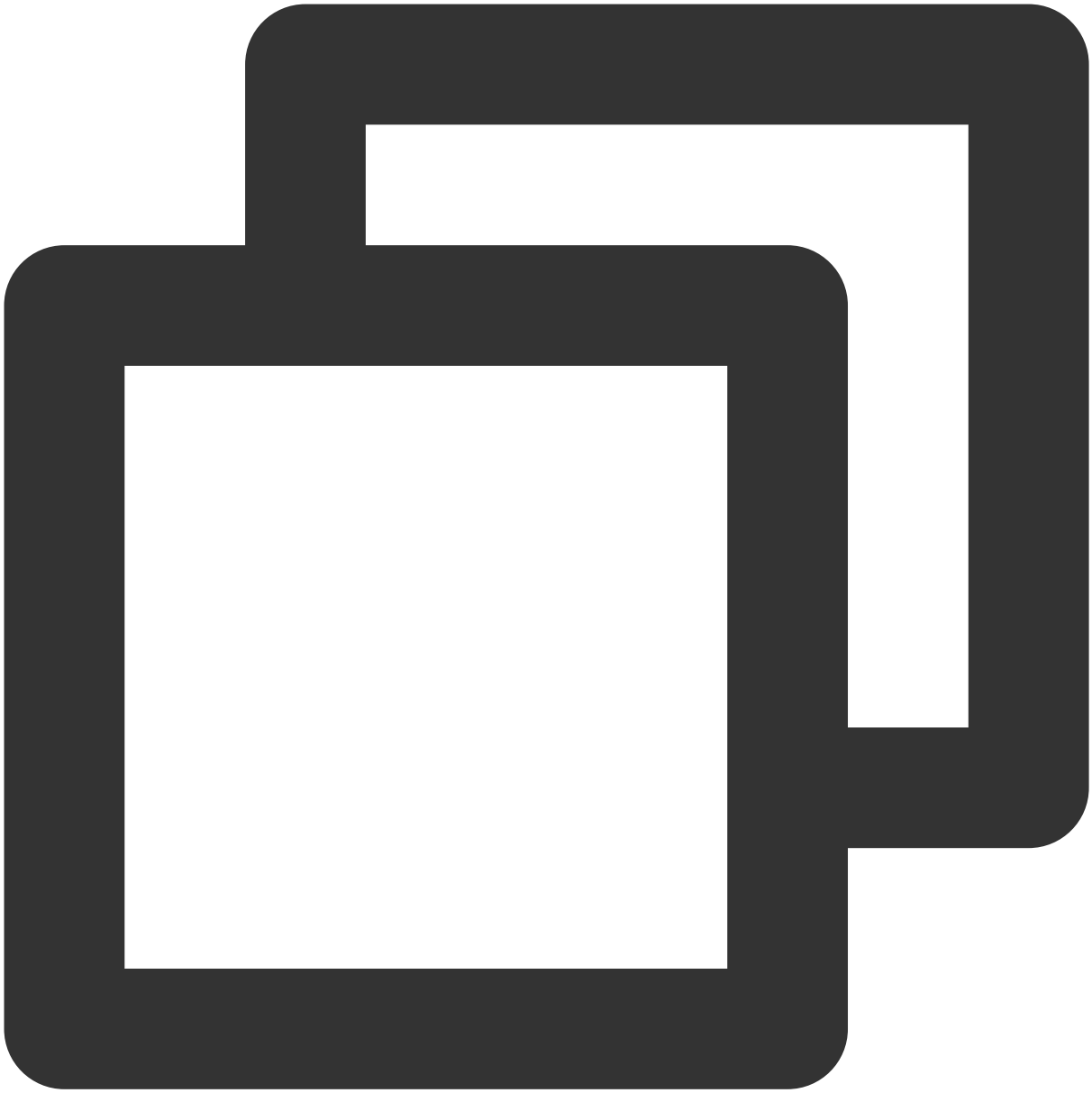
This JS API calls the native components to browse the image. Compared to the introduction of an additional JS image preview component library, this method is very simple and efficient.

In fact, WeChat officials have not exposed such calls to users outside the Tencent company. This kind of API was initially provided to their internal business usage. After many external developers discovered that, they tried to use it, gradually becoming the actual standard of WeChat's web pages.

The platform has a complete set of built-in web development toolkit, named JS-SDK, including dozens of APIs such as shooting, recording, voice recognition, QR code, map, payment, sharing, vouchers, etc. It opens up a whole new world for Web developers to use native capabilities within the host application to do things that were previously impossible or difficult to do.

Call native images in the following way:

©2013-2022 Tencent Cloud. All rights reserved.



```
wx.previewImage({
   current: 'https://img1.gtimg.com/10/1048/104857/10485726_980x1200_0.jpg',
   urls: [ // URL list of all images, array format
     'https://img1.gtimg.com/10/1048/104857/10485731_980x1200_0.jpg',
     'https://img1.gtimg.com/10/1048/104857/10485726_980x1200_0.jpg',
     'https://img1.gtimg.com/10/1048/104857/10485729_980x1200_0.jpg'
   ],
   success: function(res) {
     console.log(res)
   }
})
```

©2013-2022 Tencent Cloud. All rights reserved.

JS-SDK solves the problem that mobile web page's capabilities are insufficient by exposing the host application APIs, allowing Web developers to have more capabilities. However, it does not fix the poor experience issues for mobile web pages.

When a user accessed a web page, a blank white screen would occur before the browser started displaying. And this problem became worse on the mobile platforms because of device performance and network speed. We focused on solving this problem by providing an enhanced version of JS-SDK, which includes an important feature named offline storage for Web resources.

This feature is similar to Application Cache of HTML5, but avoids some shortcomings.

In the internal test, we found that offline storage can solve some problems, but for some complex pages, the white screen issue still existed. For example, for pages loaded with a large number of CSS or JavaScript files, the execution time of these files take up a large number of UI threads. Under this circumstances, even though the offline storage can speed up resource loading, the white screen still displays. Additionally, the file-based Cache is more complicated when the code file is updated, and the capability requirement for developers is higher.

Except for the white screen, the poor web experience problems also include delayed page response, mainly manifested in two aspects: unsmooth page switch and delayed click responses.

For some experienced Web developers, they will use SPA framework to simulate native page switching in the client. They usually simulate multiple pages in a WebView, through CSS processing, combined with the refined script code to achieve smooth click response and page switching. However, not all developers have the time and energy to make the page experience great.

The problem that the host application faces is how to design a better system, so that all developers can get a better experience in the host application. This problem cannot be solved by previous JS-SDK and a new system is required for web developers to achieve:

Fast loading

Powerful capabilities

Native experience

Easy-to-use and secure data openness

Efficient and simple development

This system is the small program that will be explained in detail in this book.

2.2. Difference between mini program development and ordinary web page development

The primary programing language for mini programs is JavaScript, so mini program development is often compared to the traditional web page development. The two share many similarities, making it relatively easy for front-end developers to transit from the web development to mini program development. However, there are still some differences:

In web development, the rendering and scripting threads are mutually exclusive, which is why long-running scripts can cause unresponsive pages. In mini programs, however, the two are separate and run in separate threads. Web developers can use various DOM APIs exposed by browsers for DOM selection and manipulation. As mentioned

©2013-2022 Tencent Cloud. All rights reserved.

above, the logical layer and rendering layer of the mini programs are separate, with the logical layer running in JSCore without a complete browser object, resulting in a lack of relevant DOM and BOM APIs. This difference means that some familiar libraries for front-end developers, such as jQuery and Zepto, cannot run in mini programs. At the same time, JSCore environment is different from the NodeJS environment, so some NPM package cannot run in mini programs either.

Web developers need to deal with various browser environment, such as IE, Chrome, QQ browser on the PC side, and Safari, Chrome and various WebViews in iOS and Android systems on the mobile side. For mini program developers, they mainly need to deal with the host application clients of the two major operating systems: iOS and Android, and mini program development tools. These three major running environments of mini programs are also different, as shown in the following table:

Operating environment	logic layer	render layer
iOS	JavaScriptCore	WKWebView
Android	X5 JSCore	X5 browser
IDE	NWJS	Chrome WebView

When web developers develop web pages, they only need to use a browser and use some auxiliary tools or editors. The development of mini programs is different. It needs to go through the process of applying for a mini program account, installing mini program developer tools, configuring projects, etc.

3. Mini program features

For ordinary users, mini programs make application accessible easily. Users can directly open them by scanning QR code, searching or a share from friends. The mini programs enhance service providers customer outreach with excellent user experience.

For developers, the fast loading and rendering capabilities of the mini program framework, along with the accompanying cloud capabilities, operational, and data aggregation capabilities, allow developers to focus on specific business logic development without dealing with trivial tasks.

4. Preparing for mini program development

Developers can automatically generate the corresponding App ID after creating a mini program on the platform, and provide different versions of IDE download addresses. Developers can choose the appropriate version to download. For details, please refer to Getting Started.

©2013-2022 Tencent Cloud. All rights reserved.

Mini program code composition
JSON Configuration

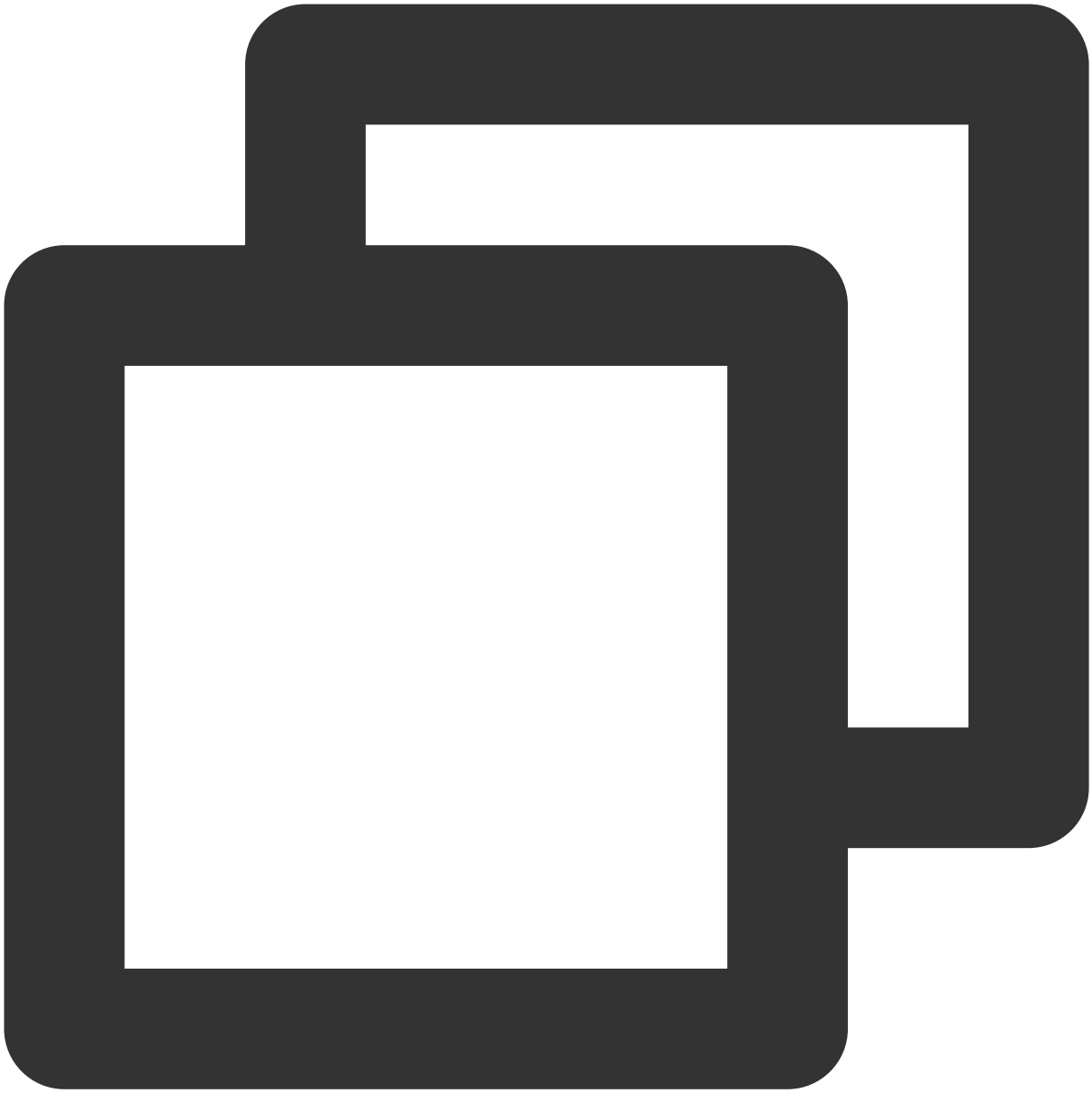
Last updated：2024-05-10 11:19:46

JSON is a data format, not a programming language. For mini programs, JSON plays the role of static configuration.

Example

Let's start by looking at an example. Please find the app.json file in the root directory and open it in the code editor. And the app.json file code for this document is as follows:

©2013-2022 Tencent Cloud. All rights reserved.



```
{
  "pages":[
    "pages/index/index",
    "pages/logs/logs"
  ],
  "window":{
    "backgroundTextStyle":"light",
    "navigationBarBackgroundColor": "#fff",
    "navigationBarTitleText": "WeChat",
    "navigationBarTextStyle":"black"
  }
```

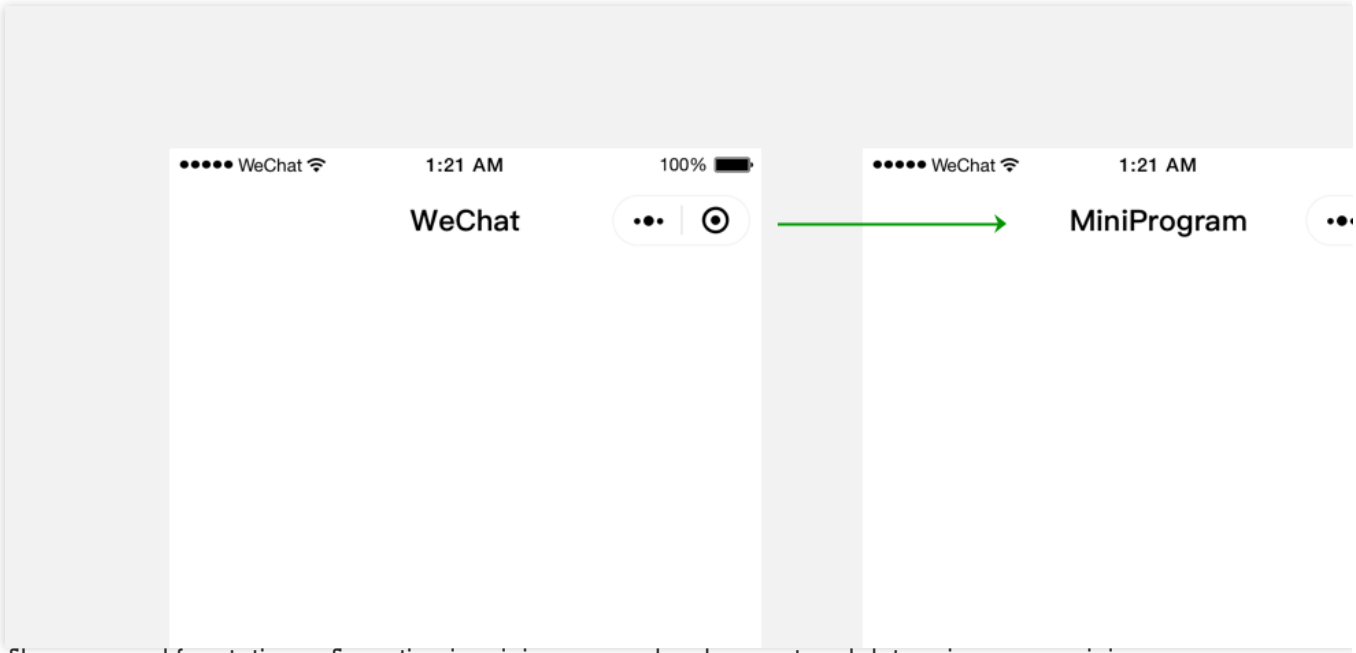
©2013-2022 Tencent Cloud. All rights reserved.

```
}
```

Change the value of "navigationBarTitleText" in line 9 from WeChat to MiniProgram.



Save the file and upon automatic IDE refresh, you can see that the title at the top of emulator has changed from WeChat to MiniProgram.



JSON files are used for static configuration in mini program development and determine some mini program performance before it runs. Please note that the mini program cannot dynamically update the JSON configuration file during the running process in order to make corresponding changes.

JSON syntax

©2013-2022 Tencent Cloud. All rights reserved.

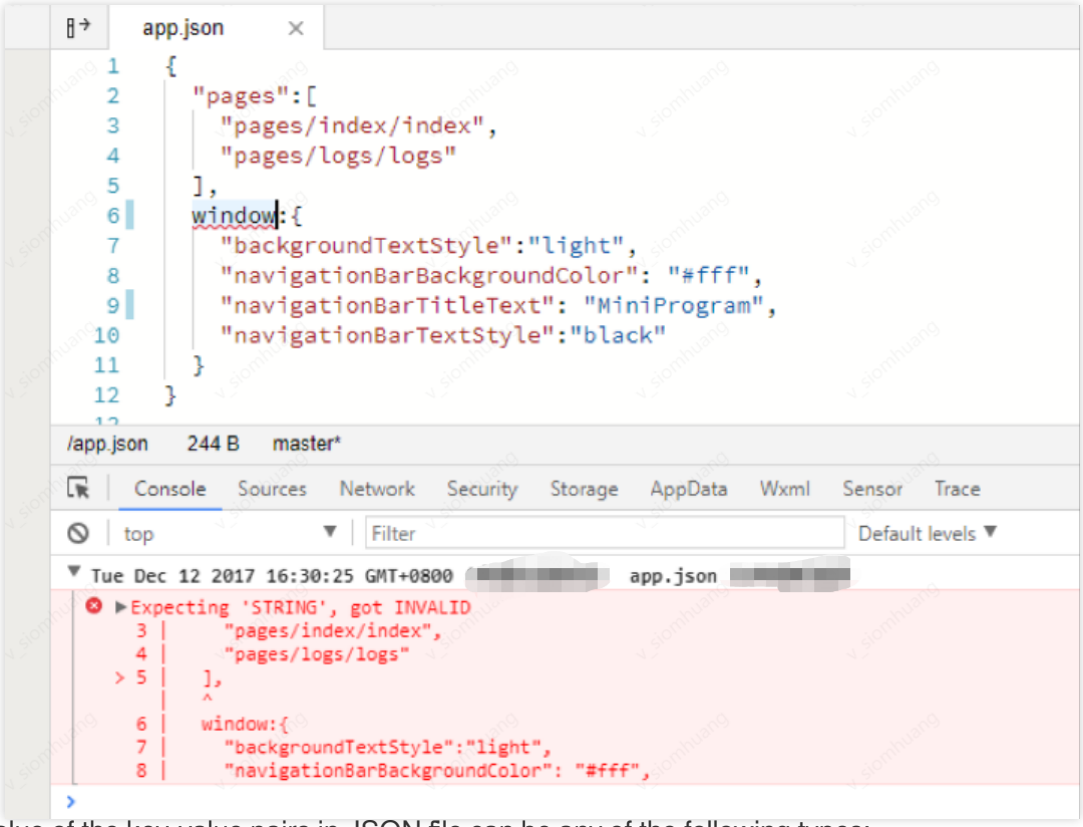
Compared with XML file, JSON file is a lightweight data exchange format, as it usually doesn't require specialized tools for reading or modifying.

JSON file code is wrapped in curly brackets {} and the data is in a key-value format.

```
"pages":[
    "pages/index/index"
    "pages/logs/logs"
],
"window" :{
    "backgroundTextstyle":"light"
    "navigationBarBackgroundColor":"#fff"
    "navigationBarTitleText":"Wechat"
    "navigationBarTextstyle":"black"
    }
}
```

It looks like JavaScript's object expression, but it's different.

The keys in JSON file must be double-quoted strings. Common mistakes include forgetting to enclose the keys in double quotes or mistakenly enclose them in single quotes.



The value of the key-value pairs in JSON file can be any of the following types:

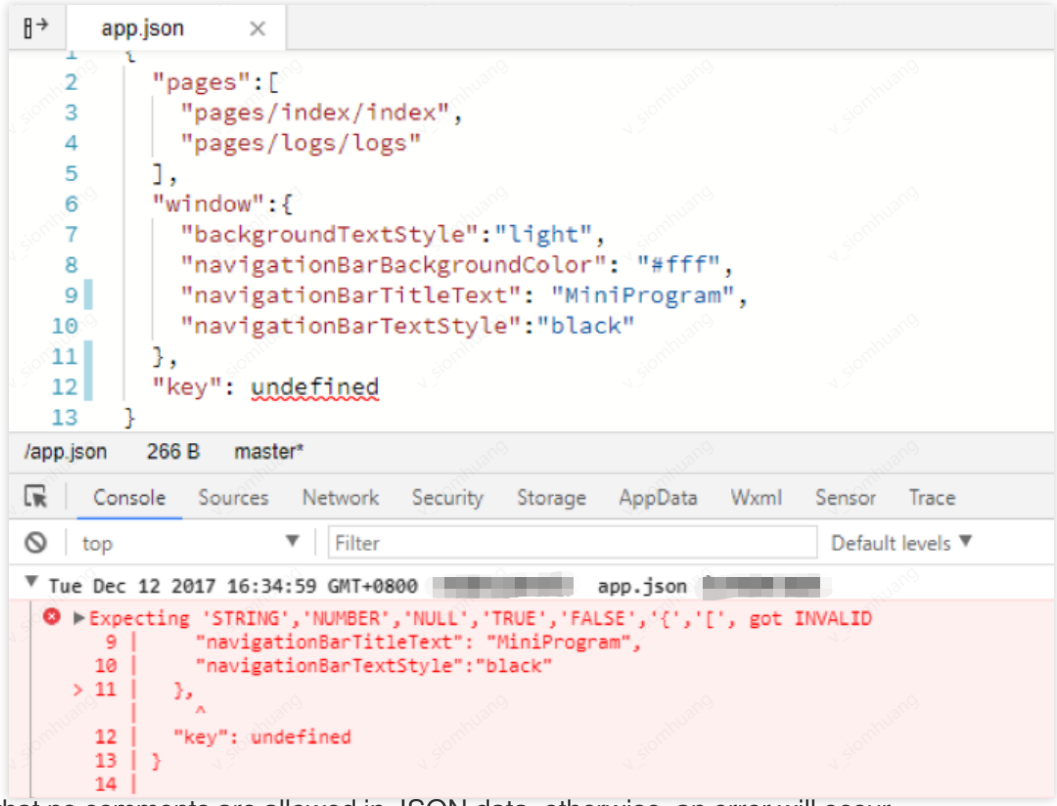
1. Number, including floating-point numbers and integers.
2. String that enclosed in double quotes.
3. Boolean, true or false.
4. Array that enclosed in square brackets [].

©2013-2022 Tencent Cloud. All rights reserved.

5. Object that enclosed in curly brackets {}.

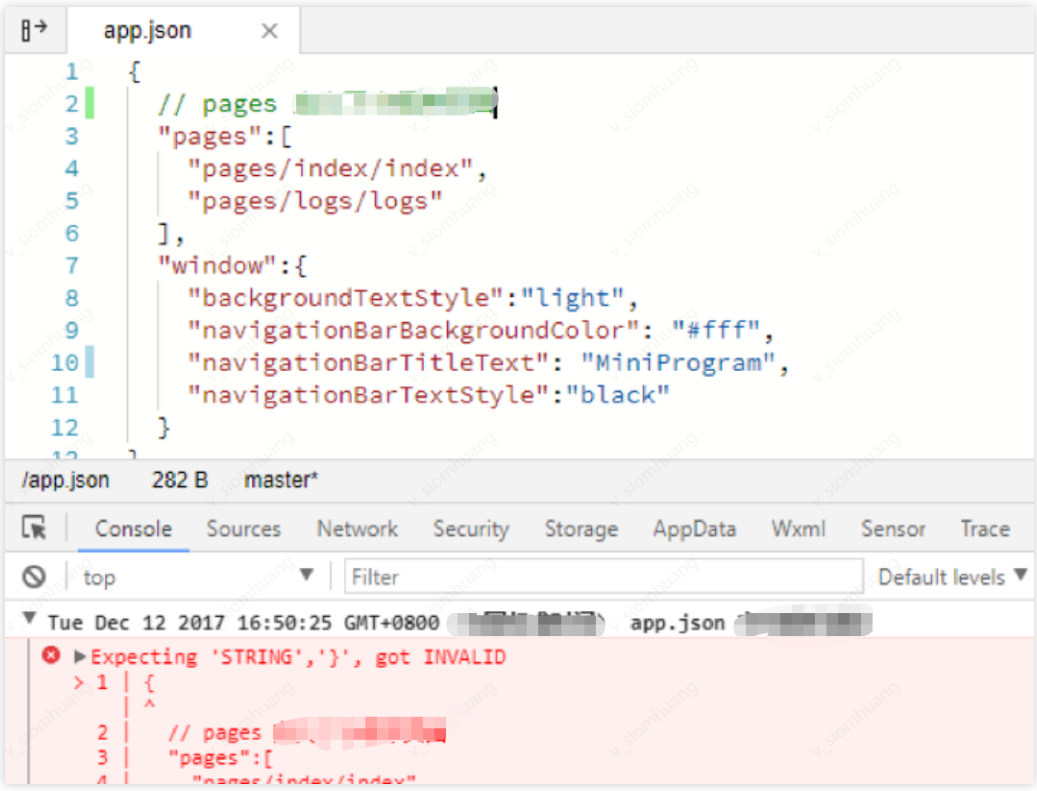
6. Null.

Other value types will trigger an error, for example, in JavaScript.



Note that no comments are allowed in JSON data, otherwise, an error will occur.

©2013-2022 Tencent Cloud. All rights reserved.


app.json
{
// pages
"pages":[
"pages/index/index",
"pages/logs/logs"
],
"window":{
"backgroundTextStyle":"light",
"navigationBarBackgroundColor": "#fff",
"navigationBarTitleText": "MiniProgram",
"navigationBarTextStyle":"black"
}
/app.json 282 B master*
Console Sources Network Security Storage AppData Wxml Sensor Trace
top
Filter
Default levels
Tue Dec 12 2017 16:50:25 GMT+0800
app.json
Expecting 'STRING','}', got INVALID
> 1 | {
| ^
2 | // pages
3 | "pages":[

WXML Template

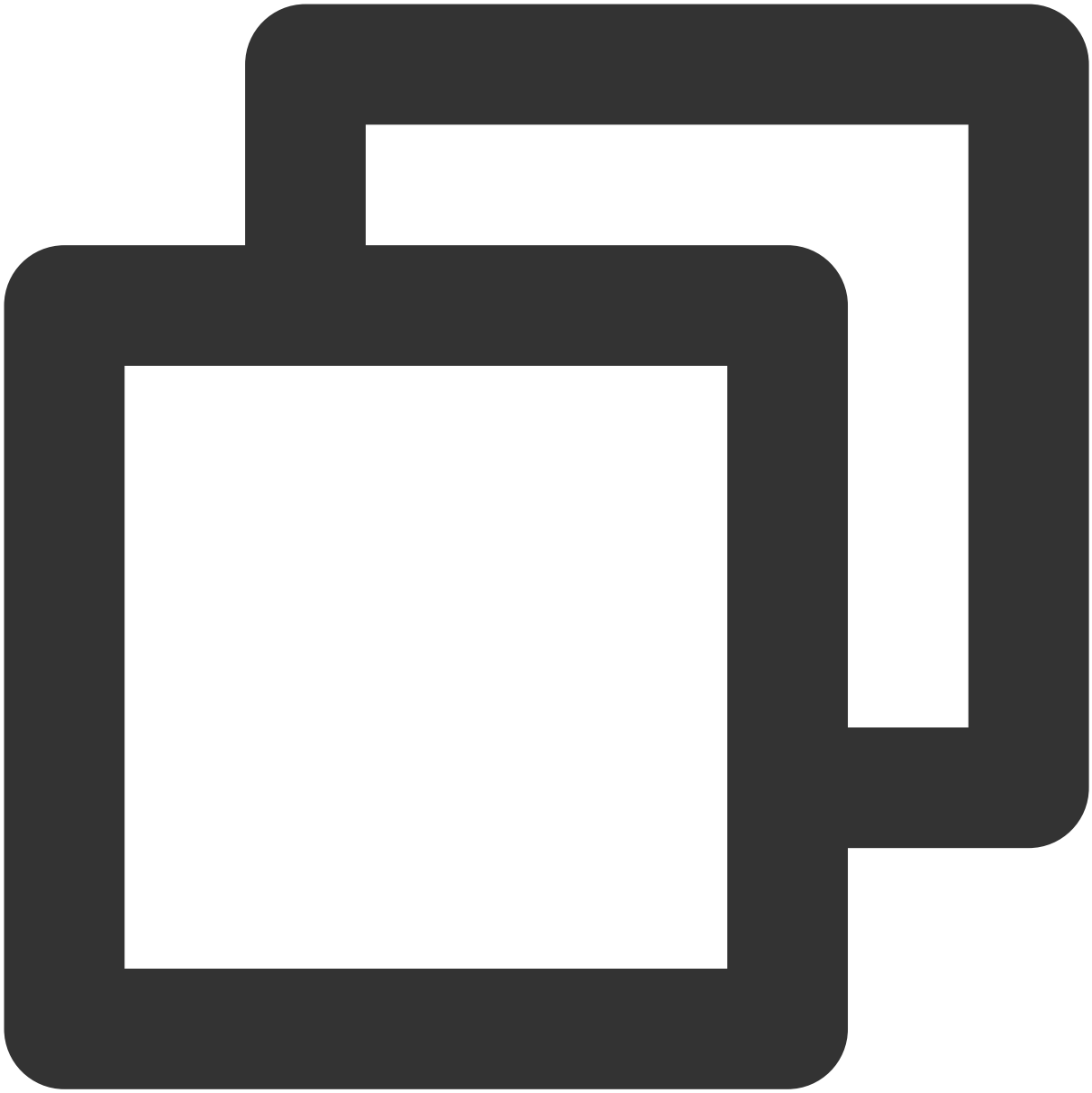
Last updated：2024-05-10 11:20:57

WeiXin Markup Language (WXML) is a set of markup language for framework design. It can be used with basic components and event systems to construct page structures.
Find the app.json file in the root directory and open it in the IDE. Add a new line `pages/wxml/index` above the line `pages/index/index` and save file. After the simulator is refreshed automatically, you can find the `pages/wxml/index.WXML` file in the editor. This document describes how to modify this file.

Overview

The suffix of the WXML file is `.wxml` . Open the pages/wxml/index.WXML file and you will find that the WXML statements are similar to the HTML syntax if you are familiar with the HTML coding.

©2013-2022 Tencent Cloud. All rights reserved.

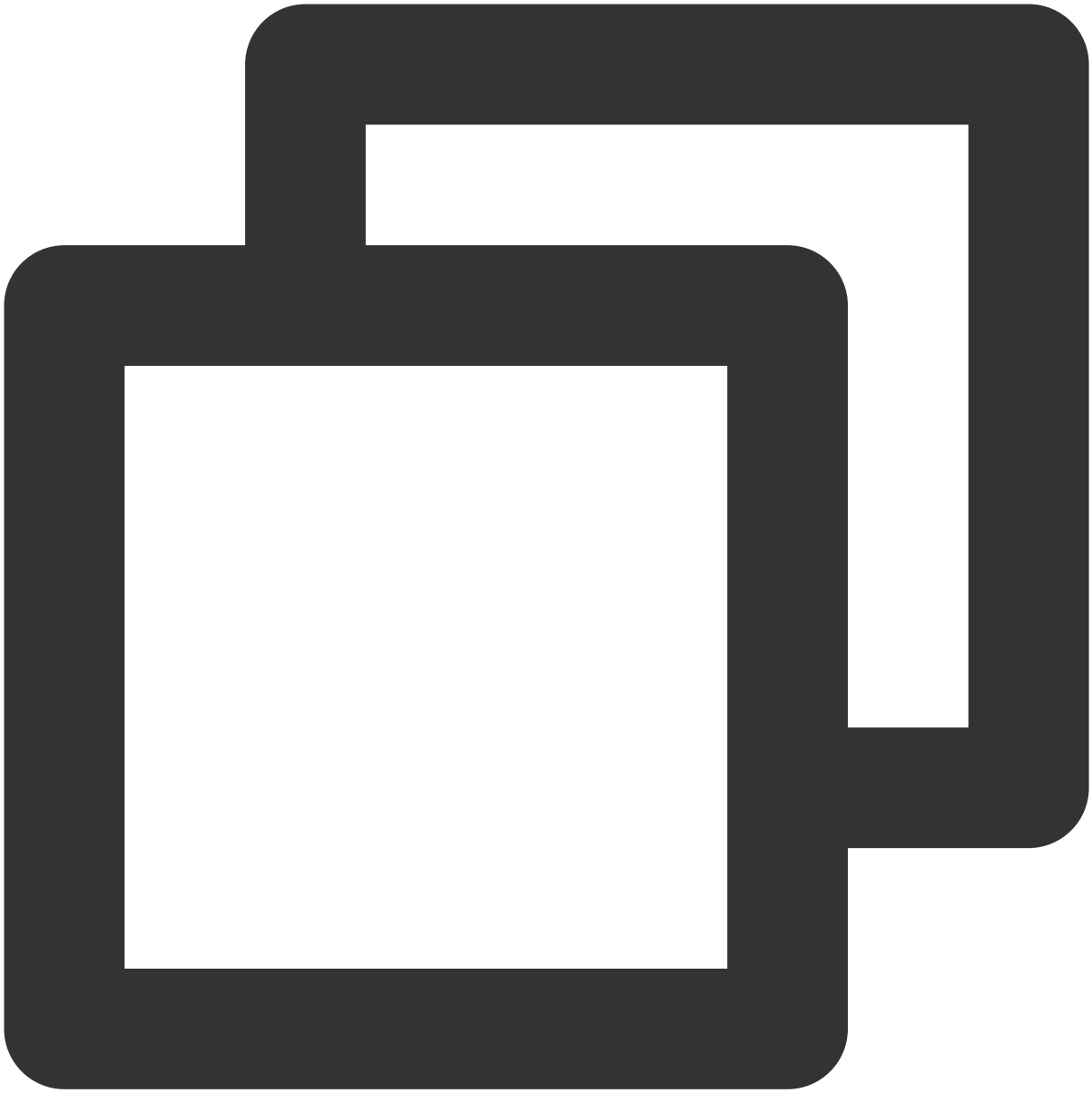


```
<!--pages/wxml/index.wxml-->

<text>pages/wxml/index.wxml</text>
```

The basic syntax of WXML without any logical features is as follows:

©2013-2022 Tencent Cloud. All rights reserved.

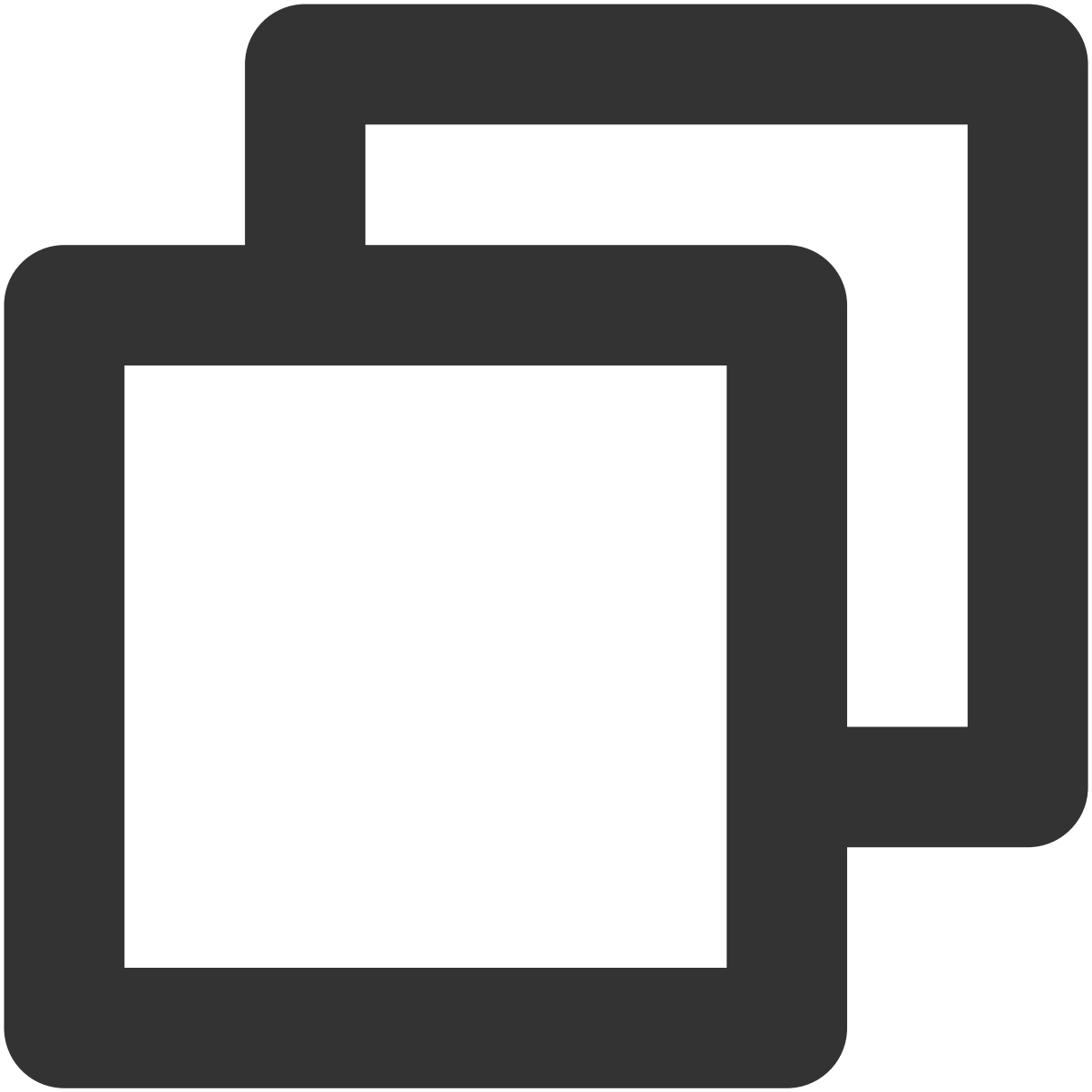


```
<!-- Add comment here-->

<Tag name Attribute name 1="Attribute value 1" Attribute name 2="Attribute value 2"
```

A complete WXML statement consists of an opening tag and a closing tag, which can wrap either content or other WXML statements. This is similar to HTML. The difference is that WXML statements must be strictly closed, otherwise, a compilation error might occur.

©2013-2022 Tencent Cloud. All rights reserved.

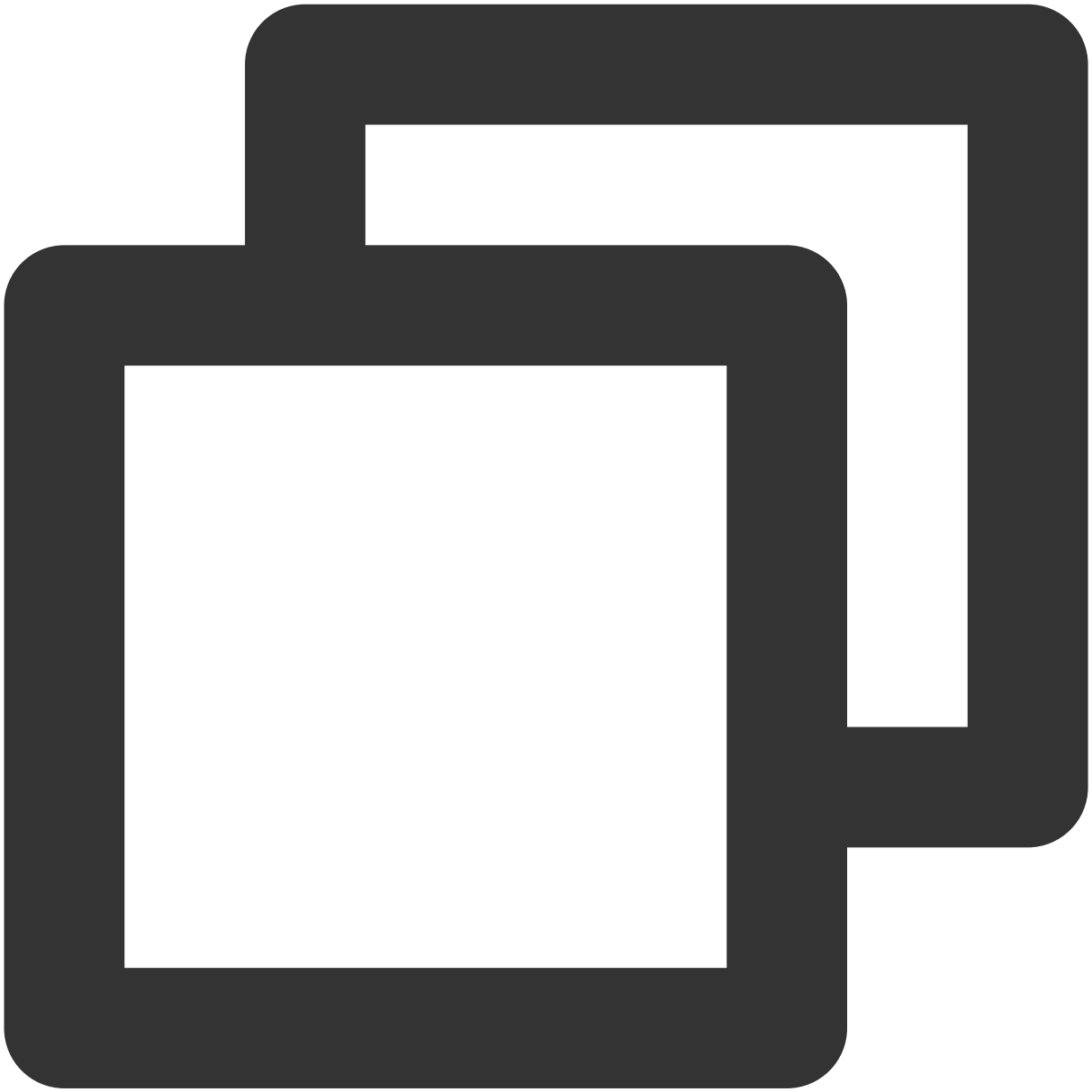


```
<text>hello world


<!--
text is not closed, resulting in a compile error:
VM148:2 ./pages/wxml/index.wxml
 end tag missing, near text
> 1 | <text>hello world
    | ^
-->
```

©2013-2022 Tencent Cloud. All rights reserved.

Tags can have attributes that provide more information about the WXML element. Attributes are always defined in the opening tag. Most of the attributes are in key="value" format except for some special ones. Note that attributes in WXML are case-sensitive, which means that class and Class are different attributes in WXML. The following is an example of text tags:



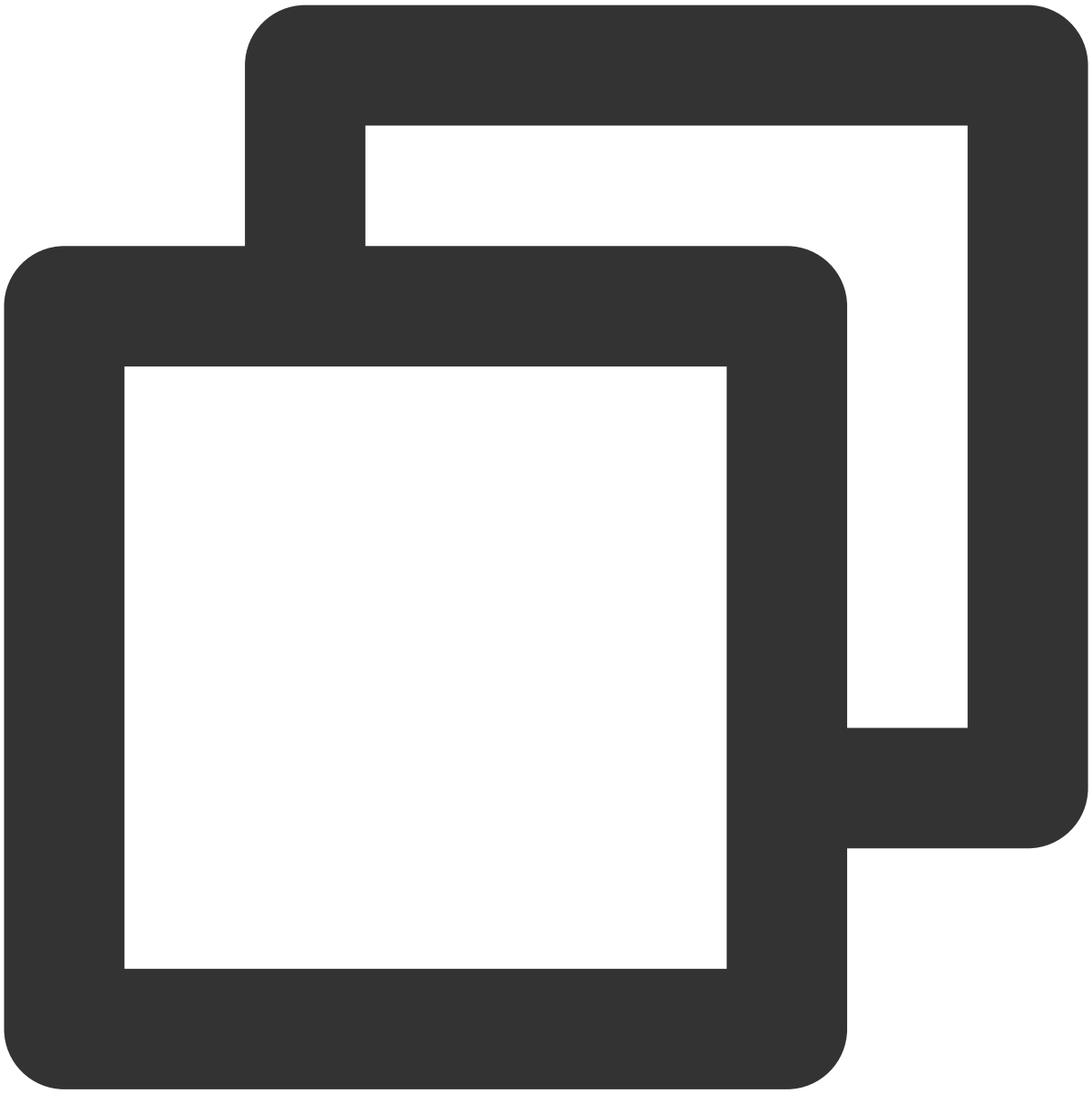
```
<!-- A simple text tag-->
<text>hello world</text>

<!-- view contains text tags -->
<view>
  <text>hello world</text>
```

©2013-2022 Tencent Cloud. All rights reserved.

```
</view>
```

An example of image tags with attributes is shown below:

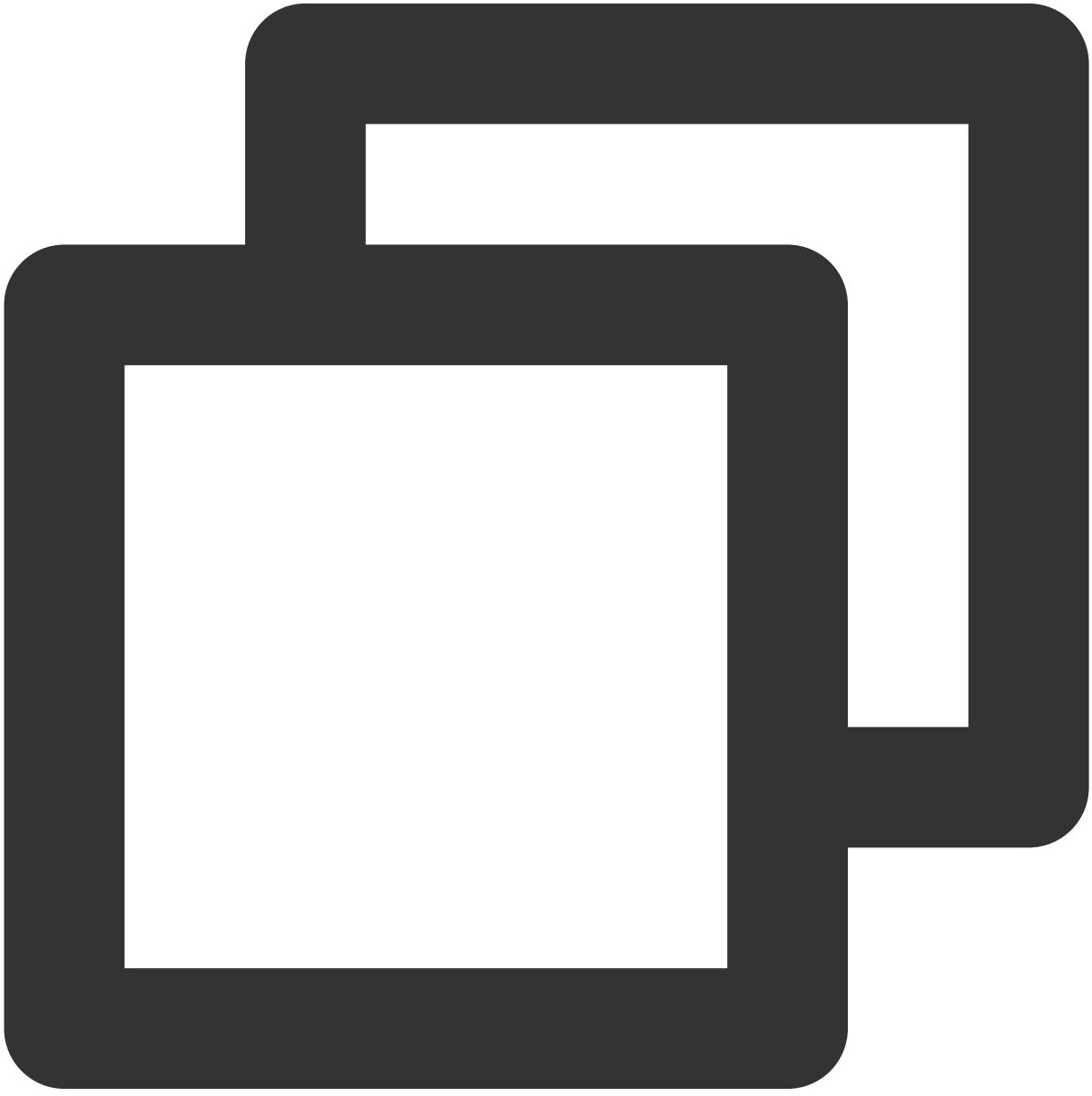


```
<image class="userinfo-avatar" src="./image/a.png" ></image>
```

Data binding

©2013-2022 Tencent Cloud. All rights reserved.

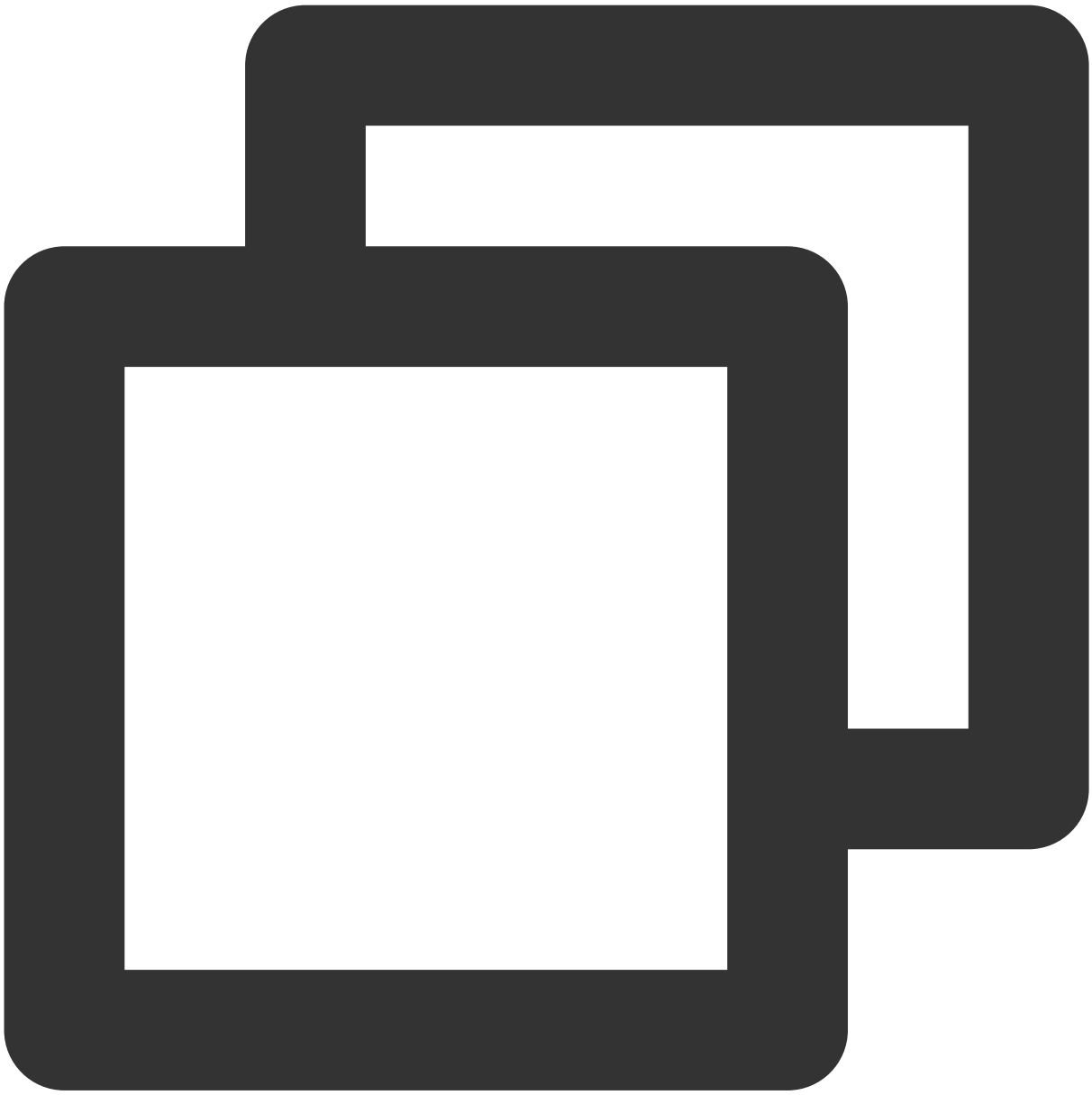
UI page displays must be dynamically updated in response to data variations at specific times or alterations caused by user actions. This necessitates incorporating dynamic page rendering capabilities during mini program execution. In Web development, developers use JavaScript to update pages in real time through DOM manipulation API. In mini program, use this data binding provided by the WXML to implement this.
Here is an example:
Modify the content of the pages/wxml/index.wxml file a little bit, as shown below:



```
<!--pages/wxml/index.wxml-->
<text>Current time: {{time}}</text>
```

©2013-2022 Tencent Cloud. All rights reserved.

After the file is saved and refreshed in the IDE, the simulator does not display the current time, because we have not set any initial value for time. Open the pages/wxml/index.js file and add `time:` in the data curly brackets `(new Date()).toString()`, as shown below:



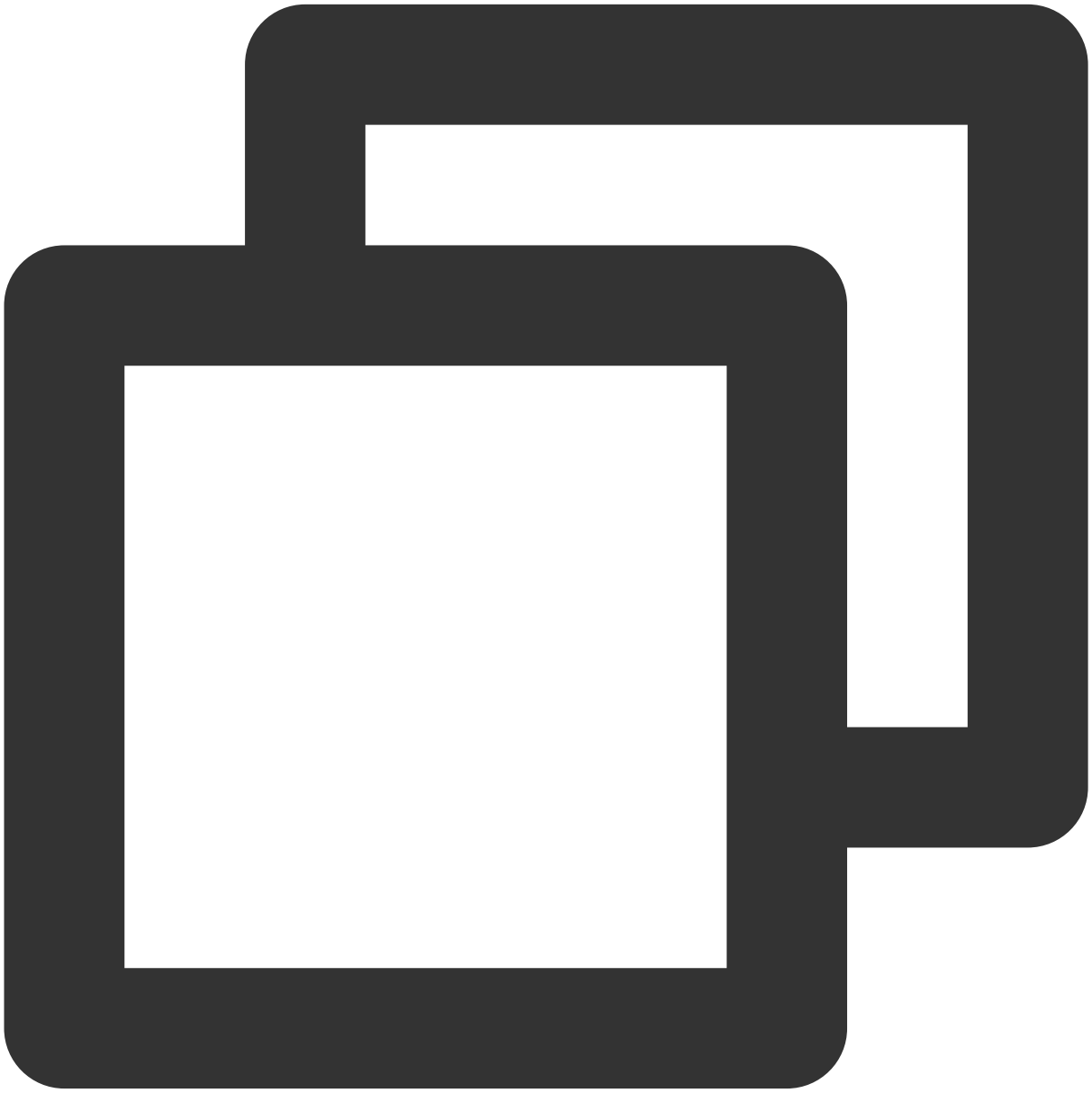
```
// pages/wxml/index.js
Page({
  /**
   * Initial page data
   */
  data: {
    time: (new Date()).toString()
```

©2013-2022 Tencent Cloud. All rights reserved.

```
  },
})
```

After saving and refreshing, simulator displays the correct current time, and the compile time will be updated.

WXML binds the data object attributes in the WXML file and the corresponding JavaScript file by {{variable name}}. In order to keep it simple, the following format is used to show this coding logic. You can use the first comment part to show the data structure in the script file corresponding to WXML, as shown below:

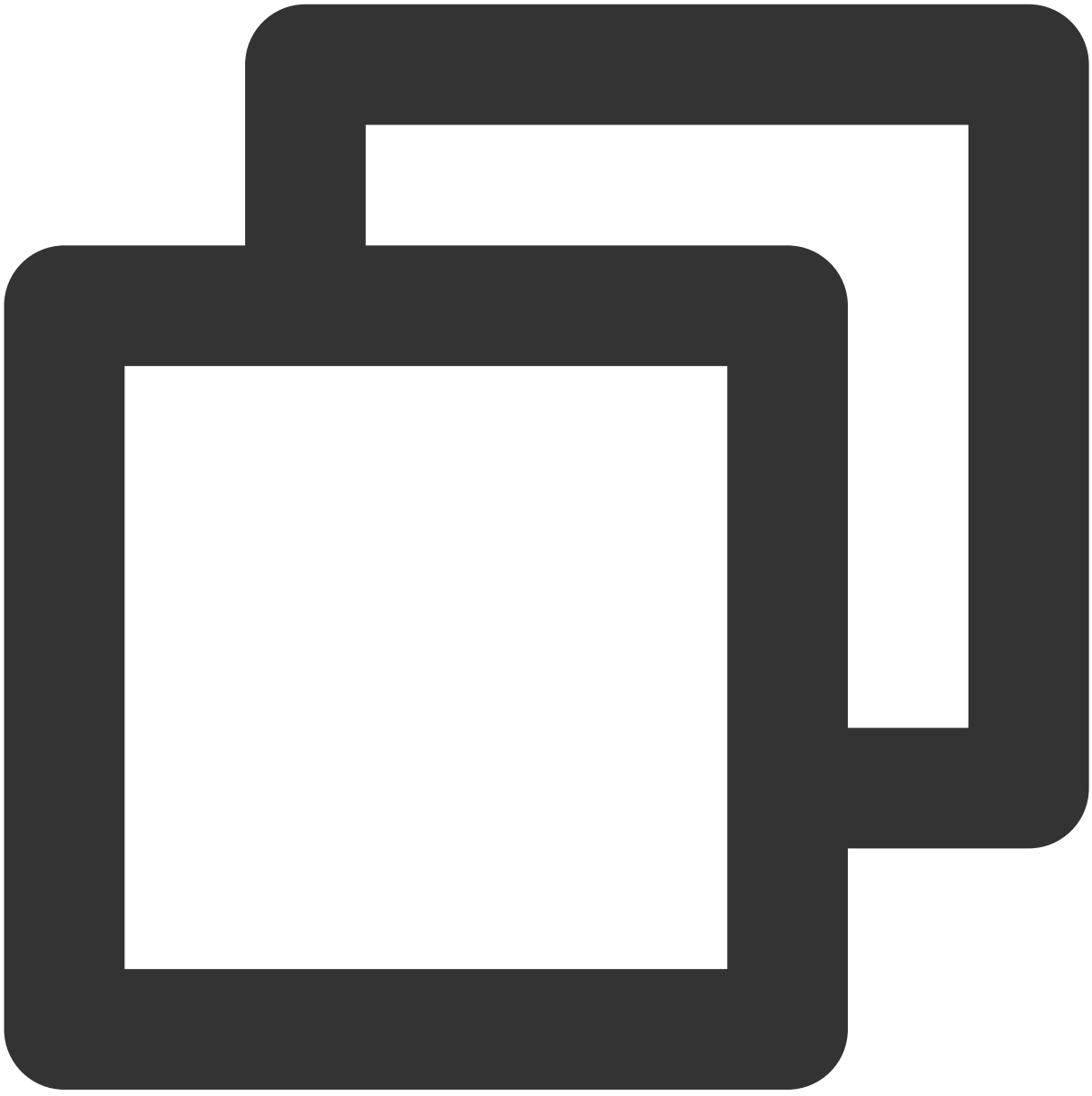


```
<!--
{
  time: (new Date()).toString()
```

©2013-2022 Tencent Cloud. All rights reserved.

```
}
-->
<text> Current time: {{time}}/text>
```

Attribute values can also be changed dynamically and must be wrapped in double quotes, as shown below:

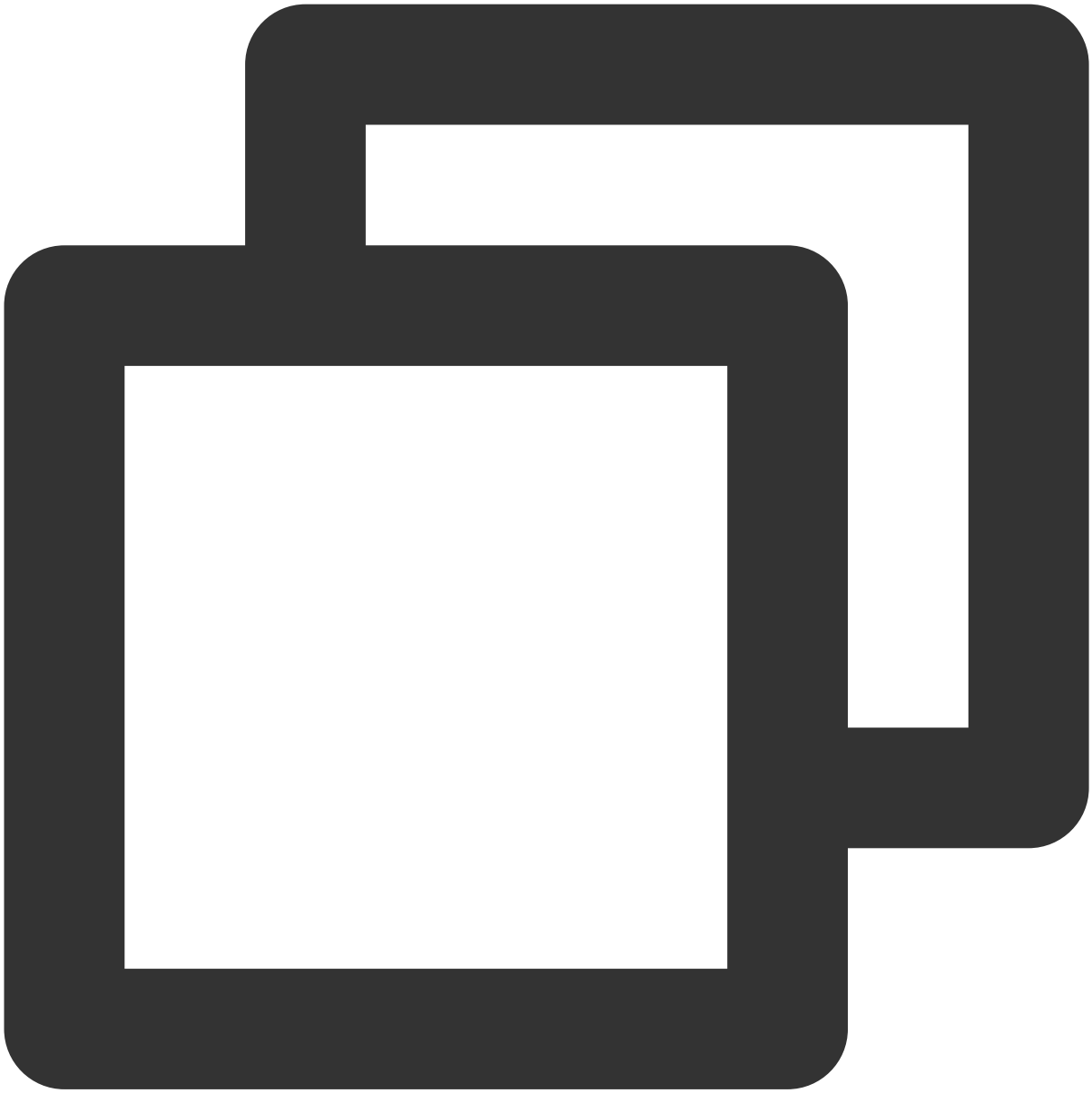


```
<!-- Correct coding -->
<text data-test="{{test}}"> hello world</text>


<!-- Wrong coding  -->
<text data-test={{test}}> hello world </text >
```

©2013-2022 Tencent Cloud. All rights reserved.

Please note that the variable is case-sensitive, that is, {{name}} and {{Name}} are two different variables, as shown below:



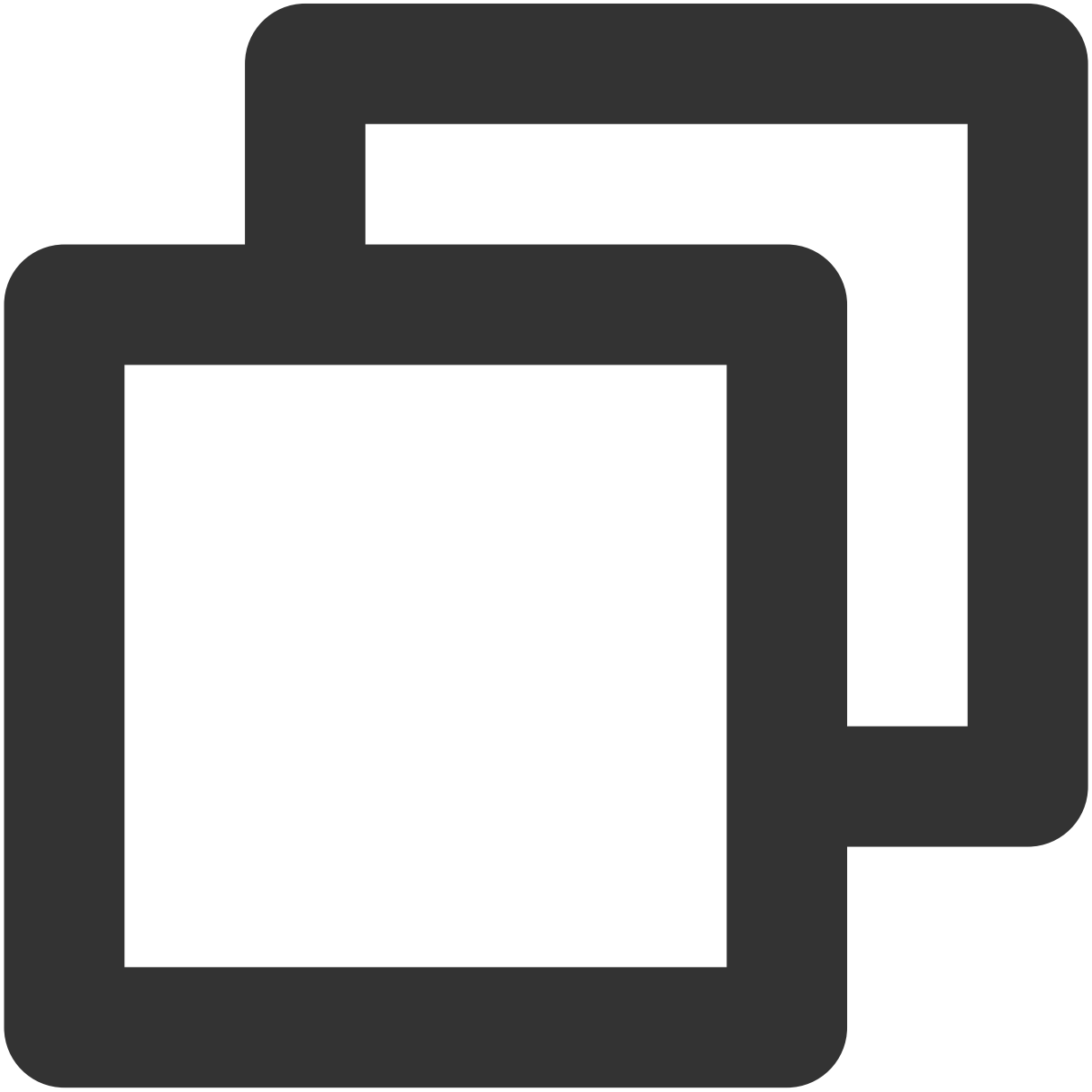
```
<!--
{
  w: 'w',
  W: 'W'
}
-->
```

©2013-2022 Tencent Cloud. All rights reserved.

```
<view>{{w}}</view>
<view>{{W}}</view>


<!-- Output
w
W
-->
```

Note that variables or variables is set to will not be synchronized to the wxml, as shown below:



©2013-2022 Tencent Cloud. All rights reserved.

```
<!--
{
  var2: undefined,
  var3: null,
  var4: "var4"
}
-->


<view>{{var1}}</view>
<view>{{var2}}</view>
<view>{{var3}}</view>
<view>{{var4}}</view>


<!--
Output：
null
var4
-->
```

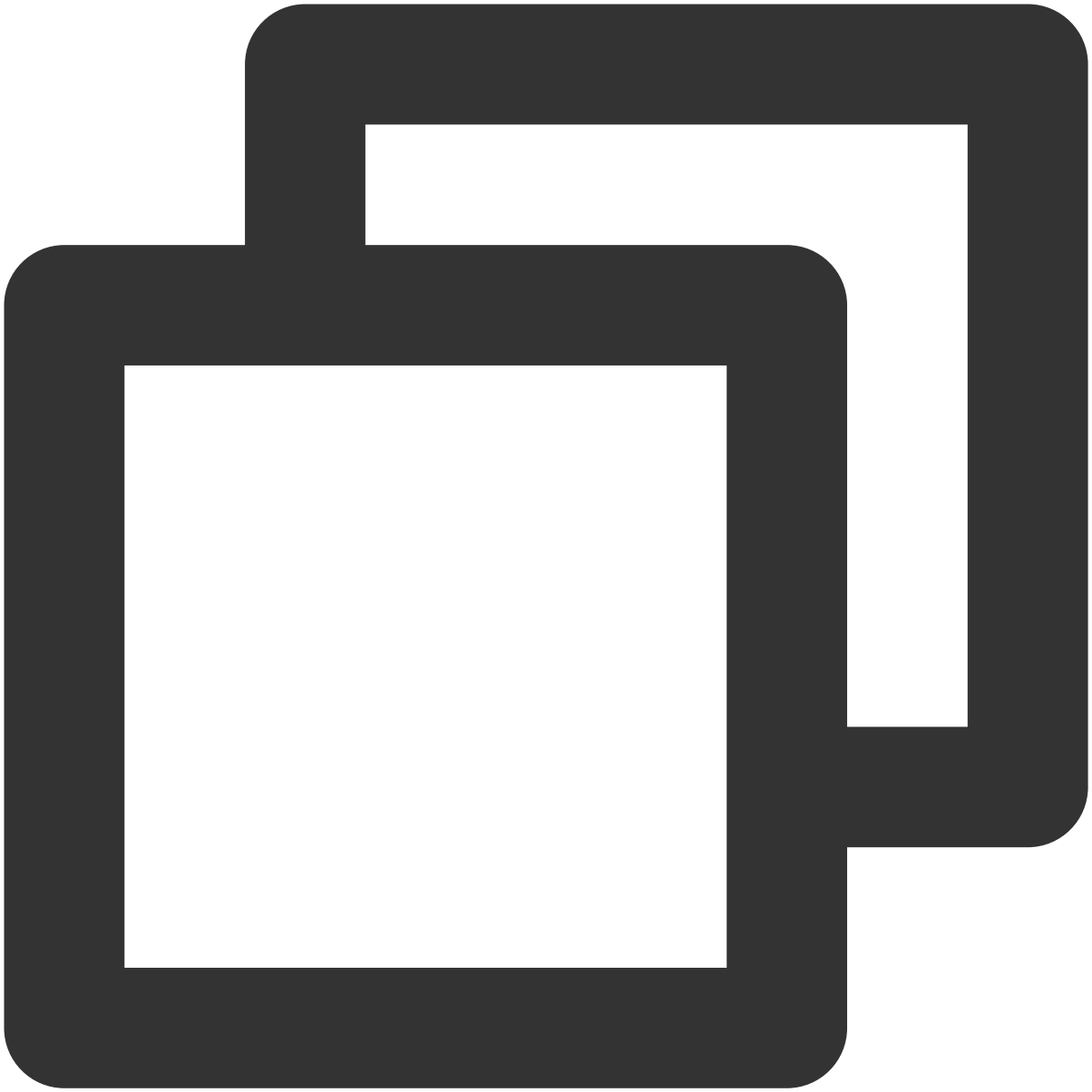
For more details about data binding, see Chapter 3.

Logical syntax

You can implement dynamic rending through {{variable name}}in WXML and implement simple operations in {{ }}.

Ternary operation:

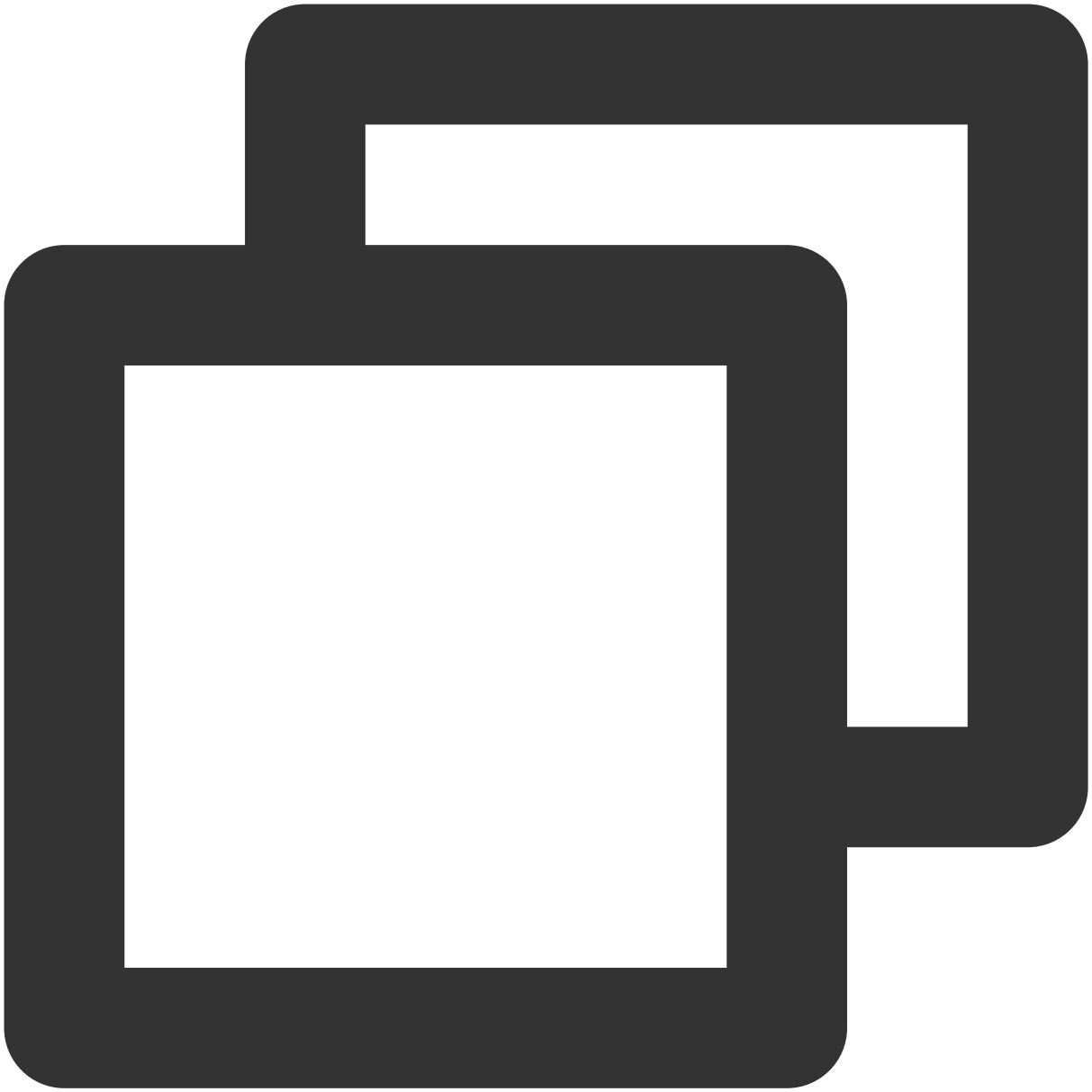
©2013-2022 Tencent Cloud. All rights reserved.



```
<!-- Display page content conditionally based on whether a is equal to 10 or not --
<text>{{ a === 10? "variable a is equal to 10": "variable a is not equal to 10"}}</
```

Arithmetic operation:

©2013-2022 Tencent Cloud. All rights reserved.



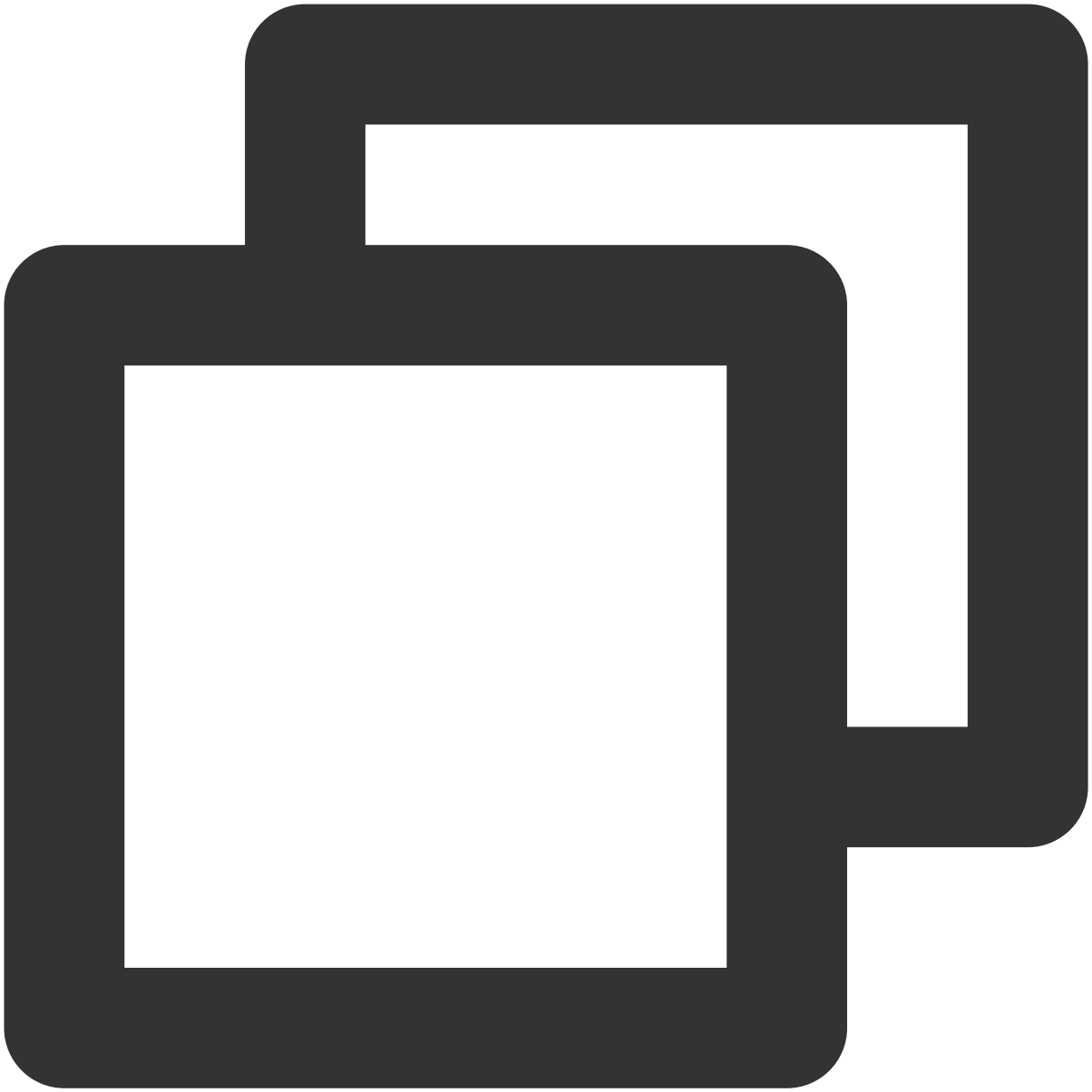
```
<!--
{ a: 1,  b: 2, c: 3 }
-->


<view> {{a + b}} + {{c}} + d </view>


<!-- Output 3 + 3 + d -->
```

Similar to arithmetic operation, it also supports string concatenation, as shown below:

©2013-2022 Tencent Cloud. All rights reserved.



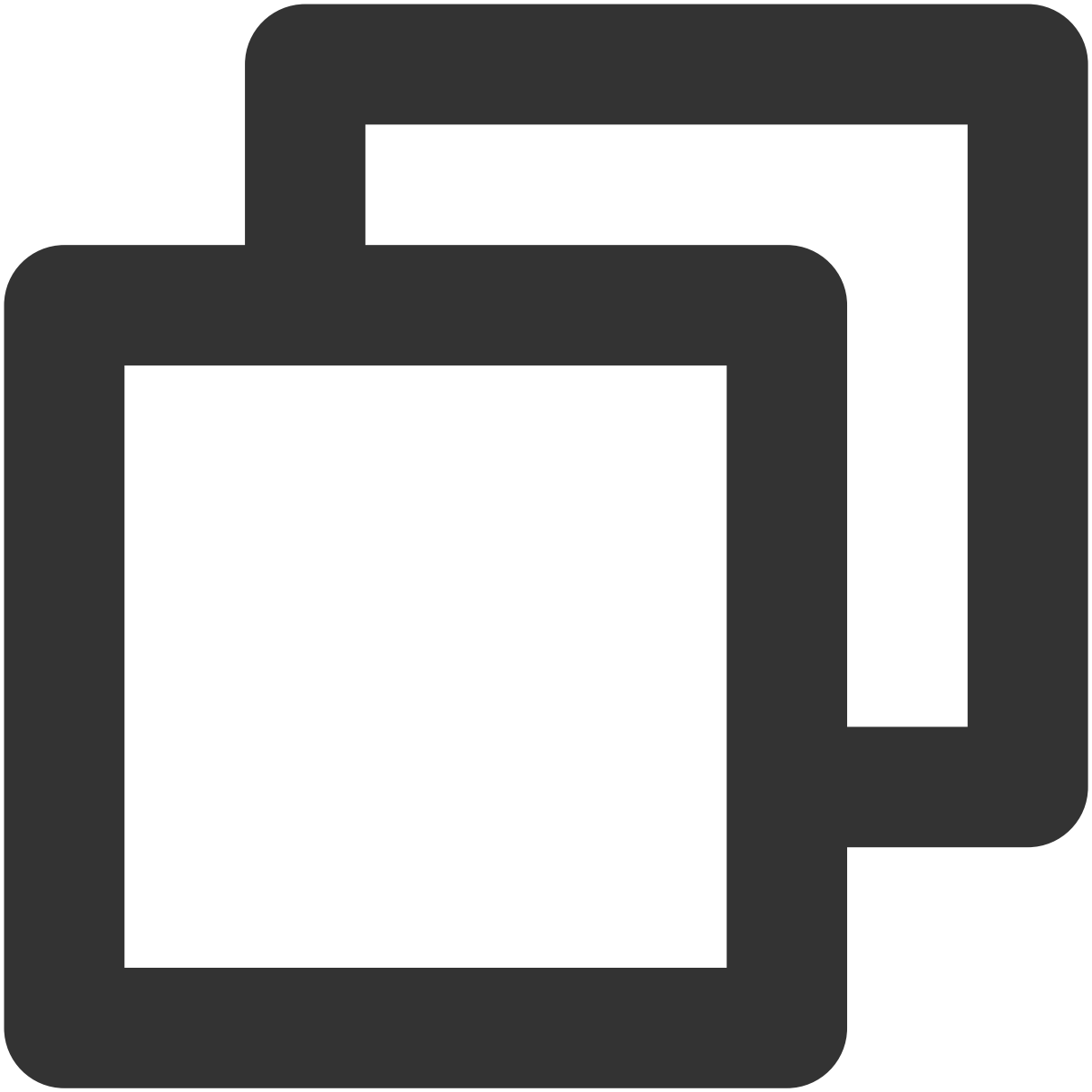
```
<!--
{ name: 'world' }
-->


<view>{{"hello " + name}}</view>


<!-- Output hello world -->
```

You can also place numbers, strings or arrays in the {{ }}, as shown below:

©2013-2022 Tencent Cloud. All rights reserved.



```
<text>{{[1,2,3]}}</text>

<!-- Output 1,2,3 -->



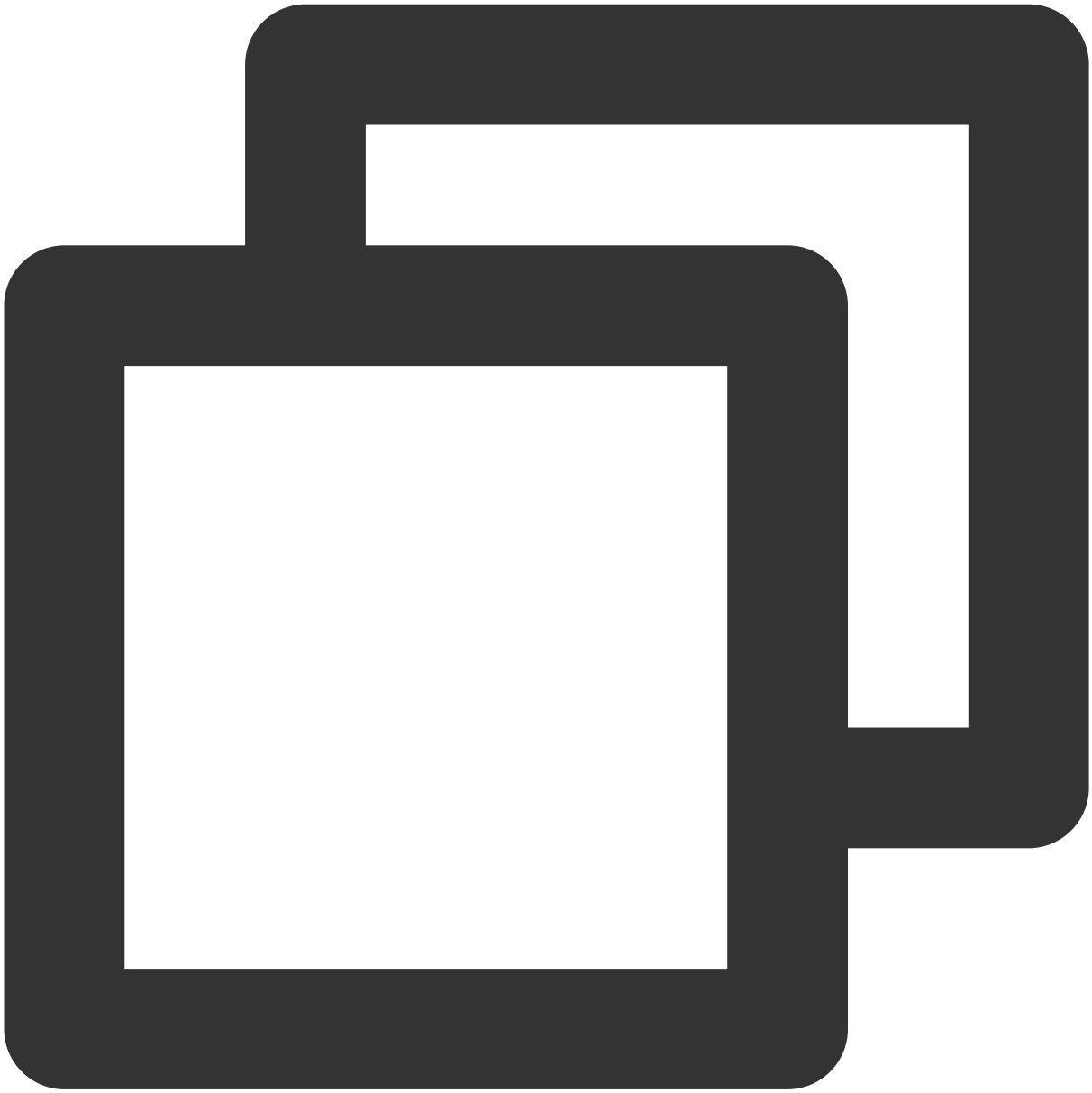
<text>{{"hello world"}}</text>

<!-- Output hello world -->
```

©2013-2022 Tencent Cloud. All rights reserved.

Logical judgement

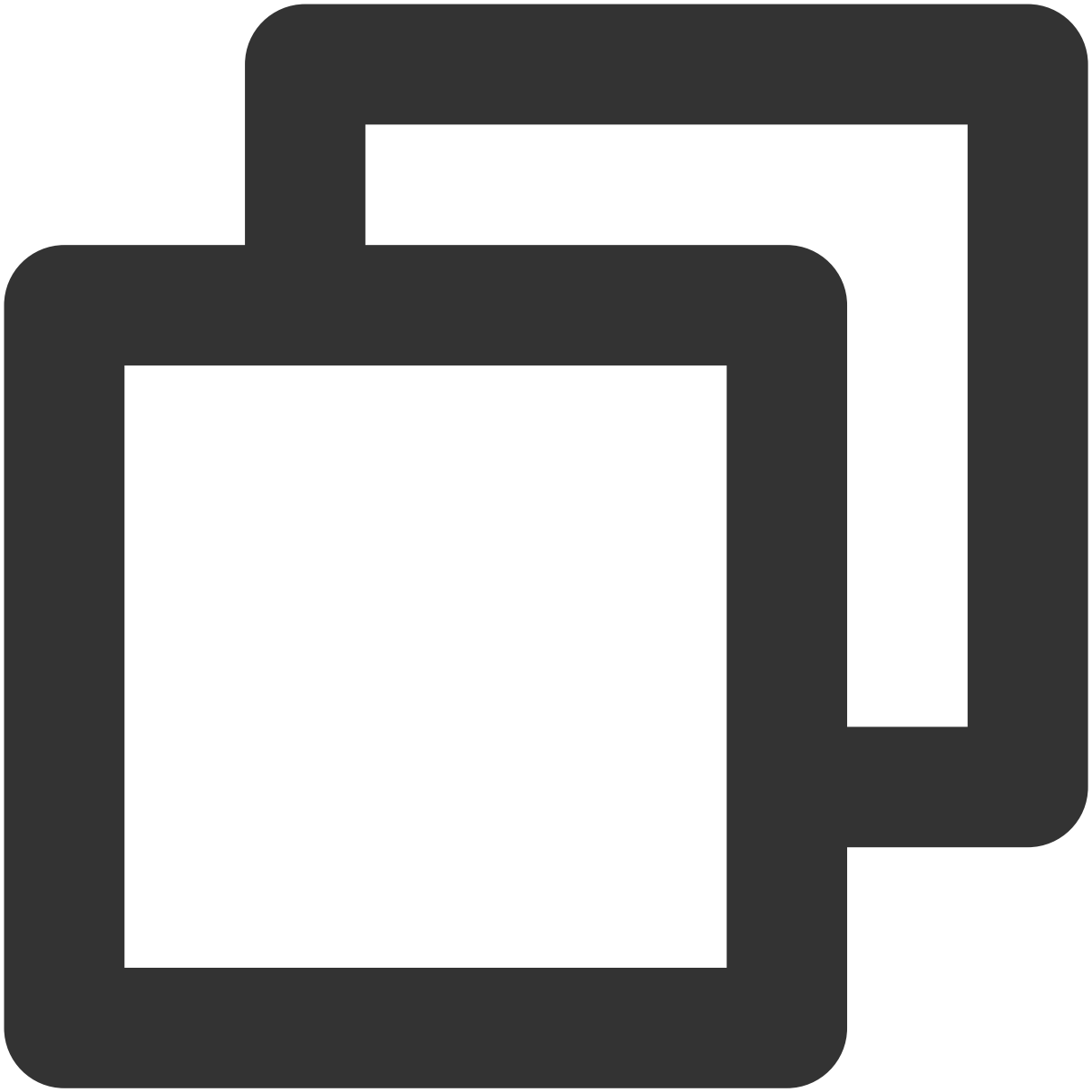
In WXML, use `wx:if="{{condition}}"` to determine whether or not to render this code block:



```
<view wx:if="{{condition}}"> True </view>
```

Use wx:elif and wx:else to add an "else" block:

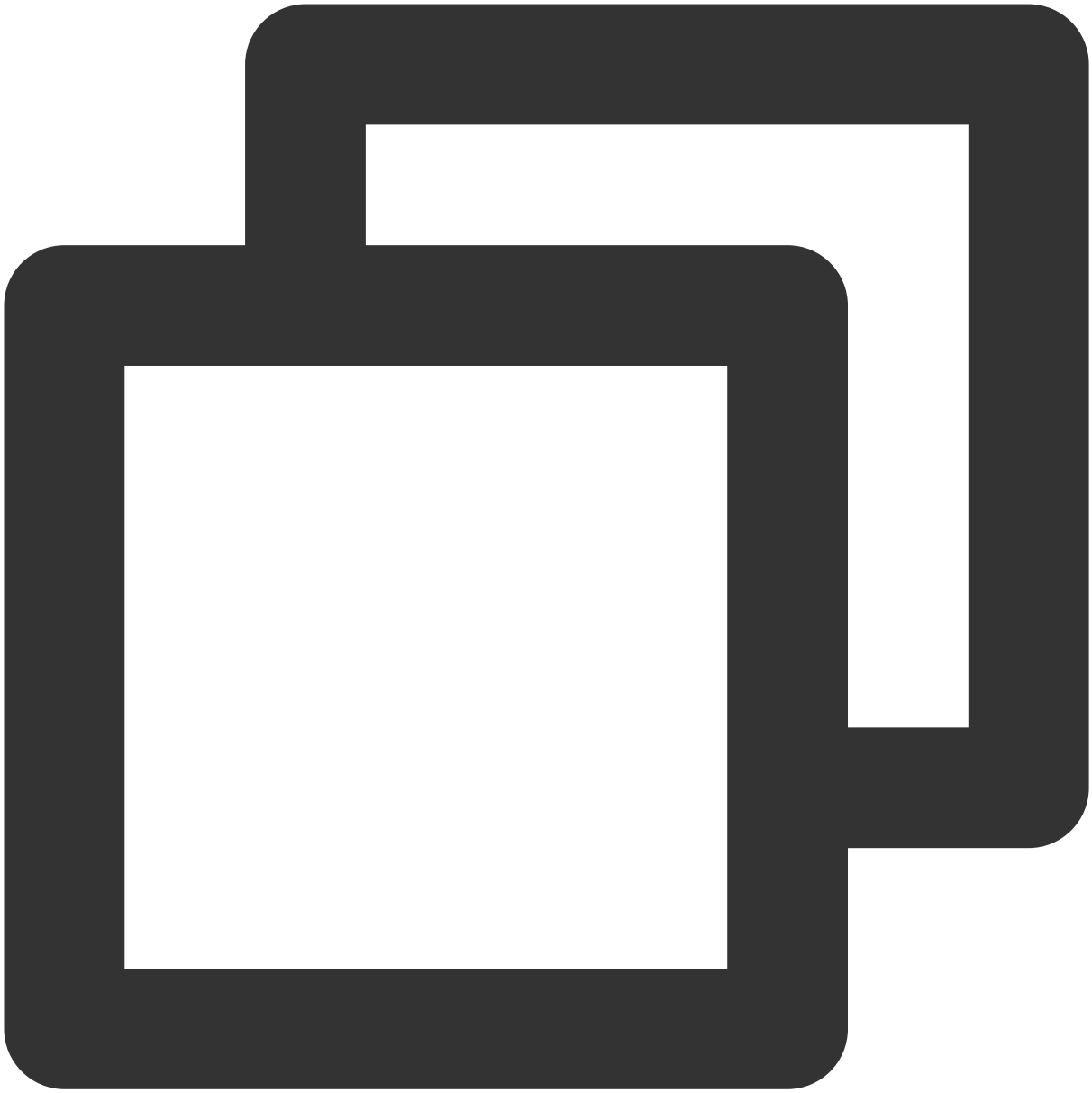
©2013-2022 Tencent Cloud. All rights reserved.



```
<view wx:if="{{length > 5}}"> 1 </view>
<view wx:elif="{{length > 2}}"> 2 </view>
<view wx:else> 3 </view>
```

Because `wx:if` is a control attribute, you must add it to a tag. To judge multiple component tags at once, you can use a `<block/>` tag to package multiple components together and use the `wx:if` control attribute above.

©2013-2022 Tencent Cloud. All rights reserved.

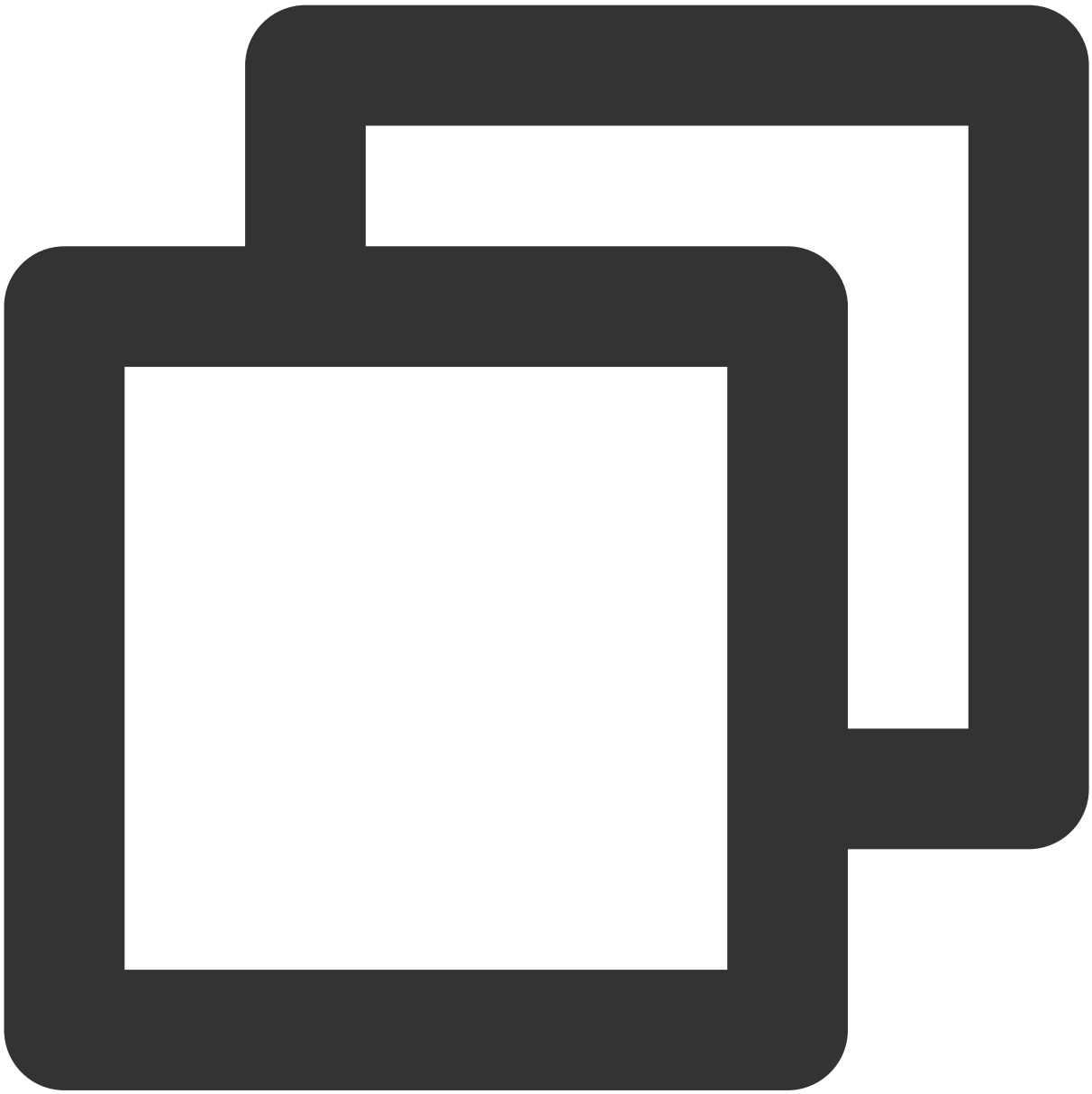


```
<block wx:if="{{true}}">
  <view> view1 </view>
  <view> view2 </view>
</block>
```

List rendering

©2013-2022 Tencent Cloud. All rights reserved.

In a component, by binding an array using the `wx:for` control attribute, you can use the data of the array items to re-render this component. The subscript variable name of the current item of the default array defaults to `index`, and the variable name of the current item of the array defaults to `item`:



```
<!-- `array` is an array -->
<view wx:for="{{array}}">
  {{index}}: {{item.message}}
</view>

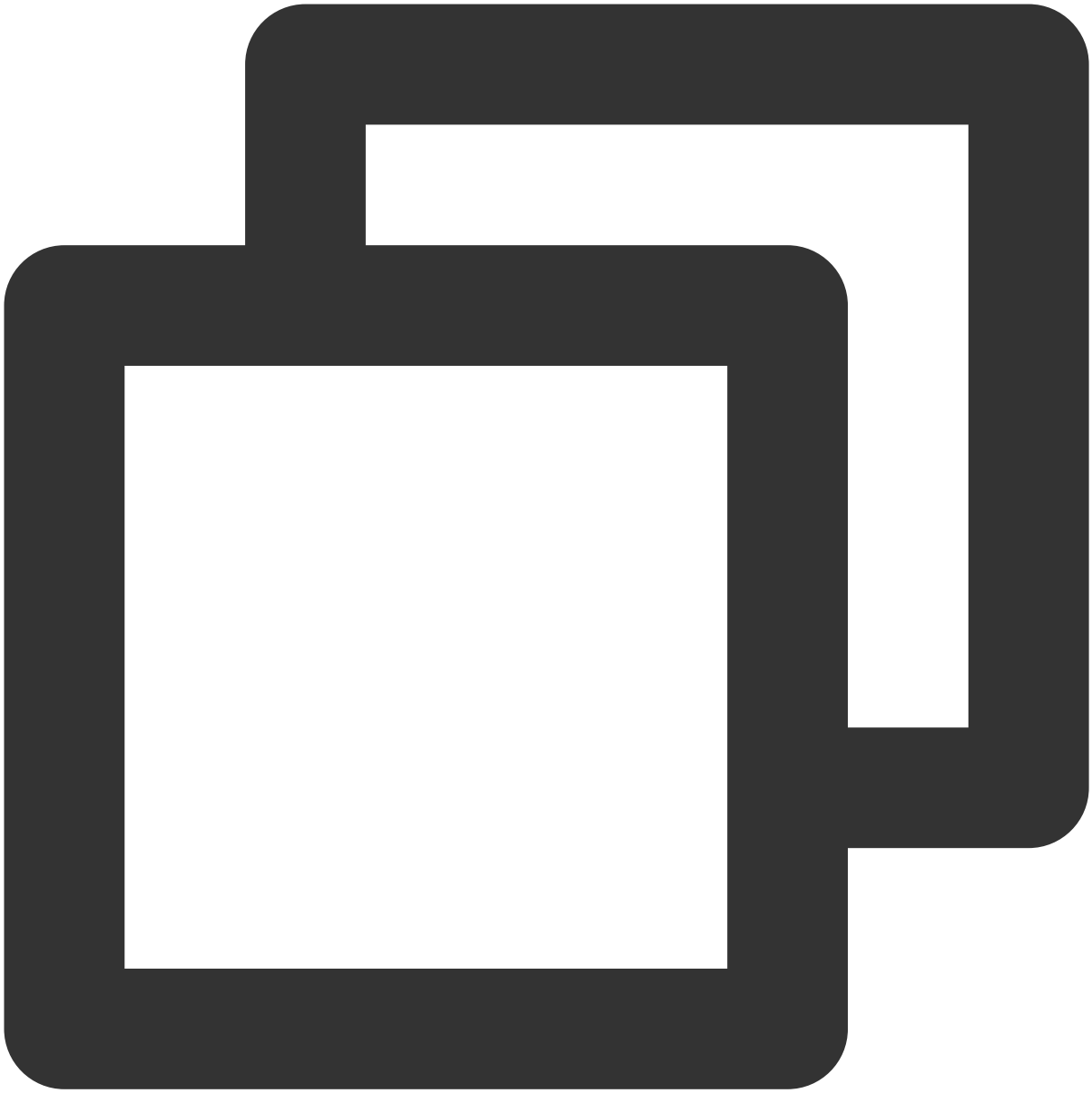
<!-- Corresponding script file
Page({
```

©2013-2022 Tencent Cloud. All rights reserved.

```
    data: {
      array: [{
        message: 'foo',
      }, {
        message: 'bar'
      }]
    }
  })
  -->
```

You can use `wx:for-item` to specify the variable name of the array's current element and use `wx:for-index` to specify the variable name of the array's current subscript:

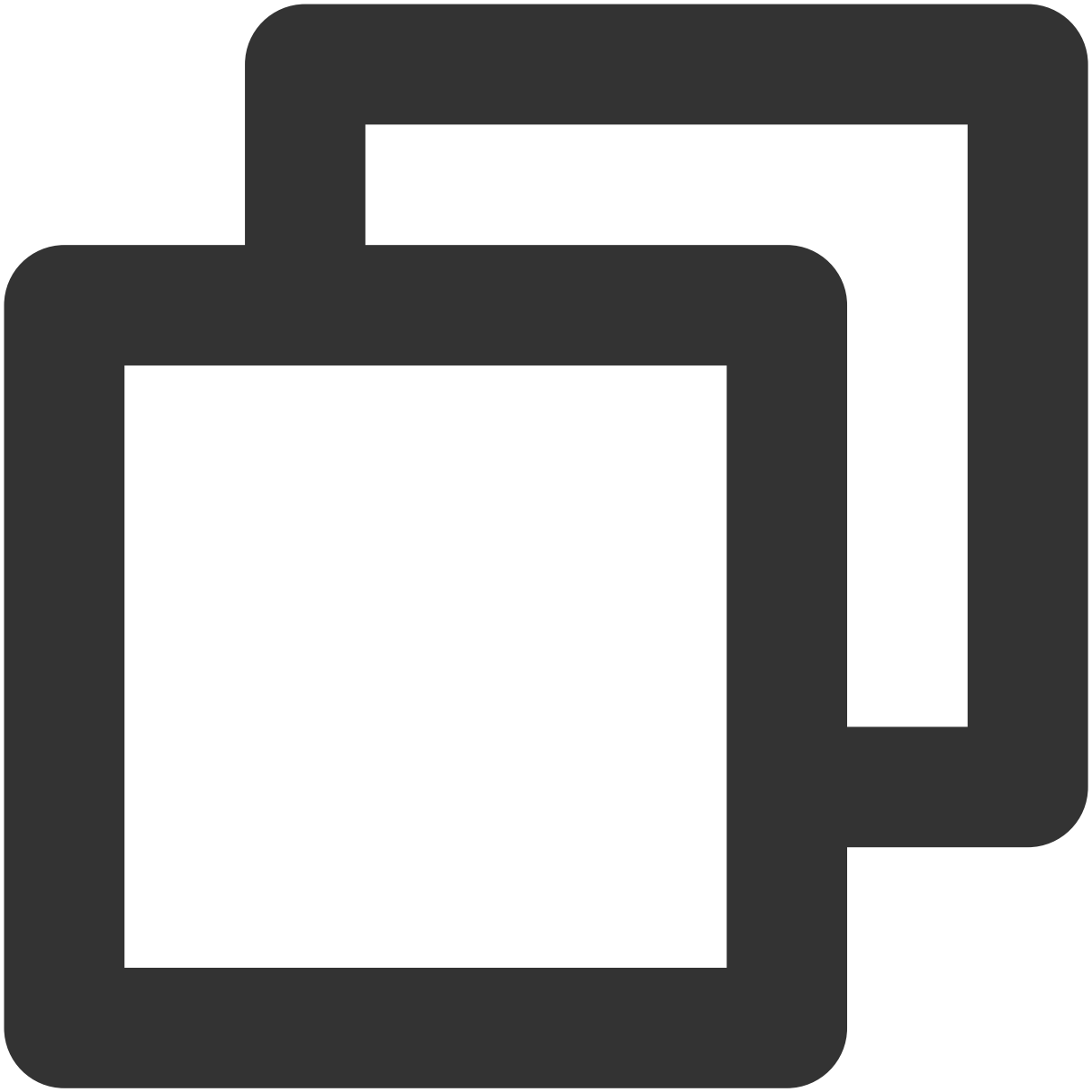
©2013-2022 Tencent Cloud. All rights reserved.



```
<view wx:for="{{array}}" wx:for-index="idx" wx:for-item="itemName">
  {{idx}}: {{itemName.message}}
</view>
```

Similar to `block wx:if` , you can use `wx:for` on a `<block/>` tag to render a structural block containing multiple nodes:

©2013-2022 Tencent Cloud. All rights reserved.



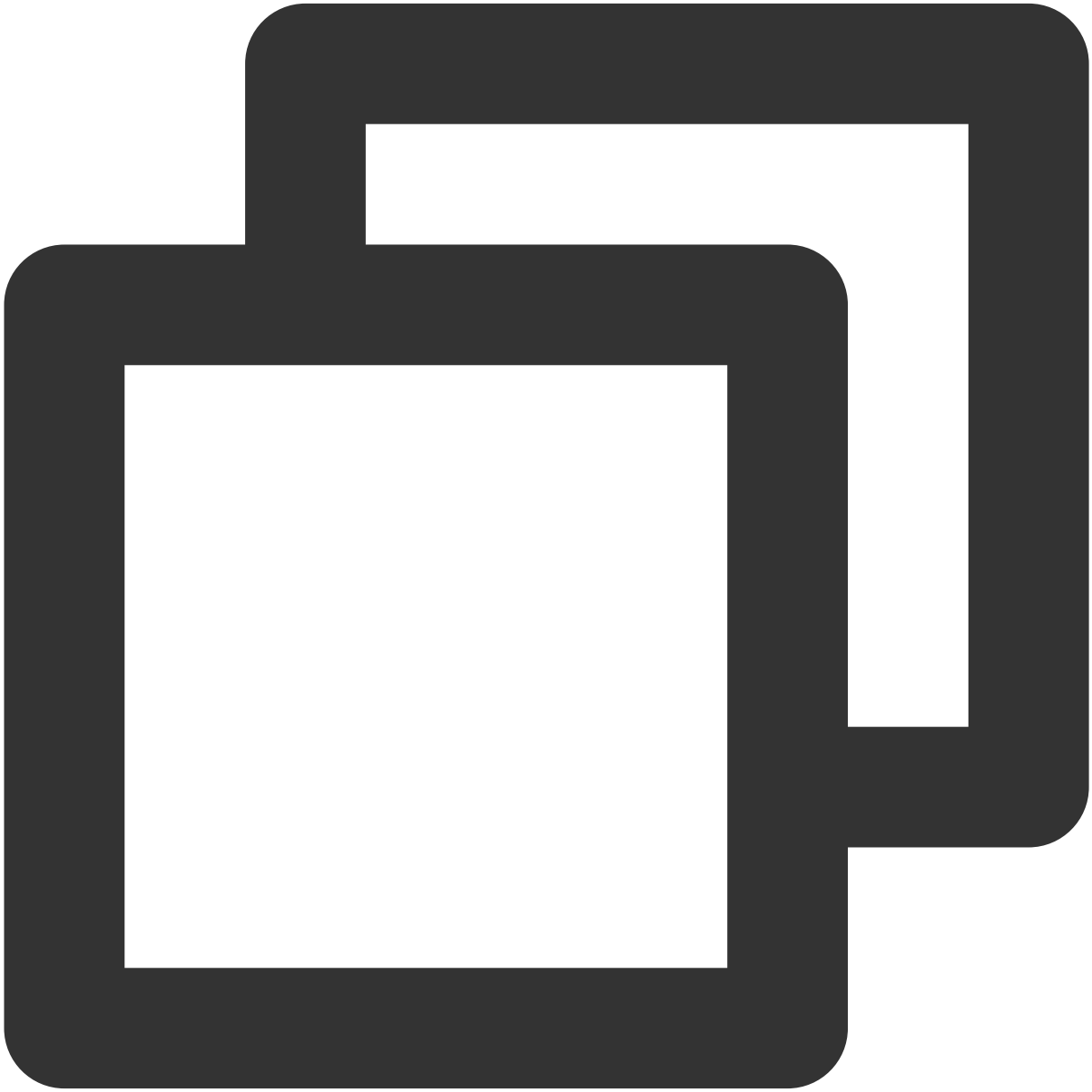
```
<block wx:for="{{[1, 2, 3]}}">
  <view> {{index}}: </view>
  <view> {{item}} </view>
</block>
```

If the positions of list items dynamically change or new items are added to the list and you want the items in the list to retain their features and statuses (such as the content input in <input/> and the selection status of <switch/>), you must use `wx:key` to specify the unique identifiers of the items in the list.

The `wx:key` value is provided in either of the following formats:

©2013-2022 Tencent Cloud. All rights reserved.

1. String: Represents an attribute of an item in the for loop array. The value of this attribute must be a unique string or number in the list and cannot dynamically change.
2. Reserved keyword `this` : Represents the item itself in the for loop. This indicates that the item itself is a unique string or number. For example:
When a data change triggers re-rendering at the rendering layer, the components that have keys are corrected. The framework ensures that they are reordered, rather than recreated. This ensures the components retain their statuses and improves the efficiency of list rendering.

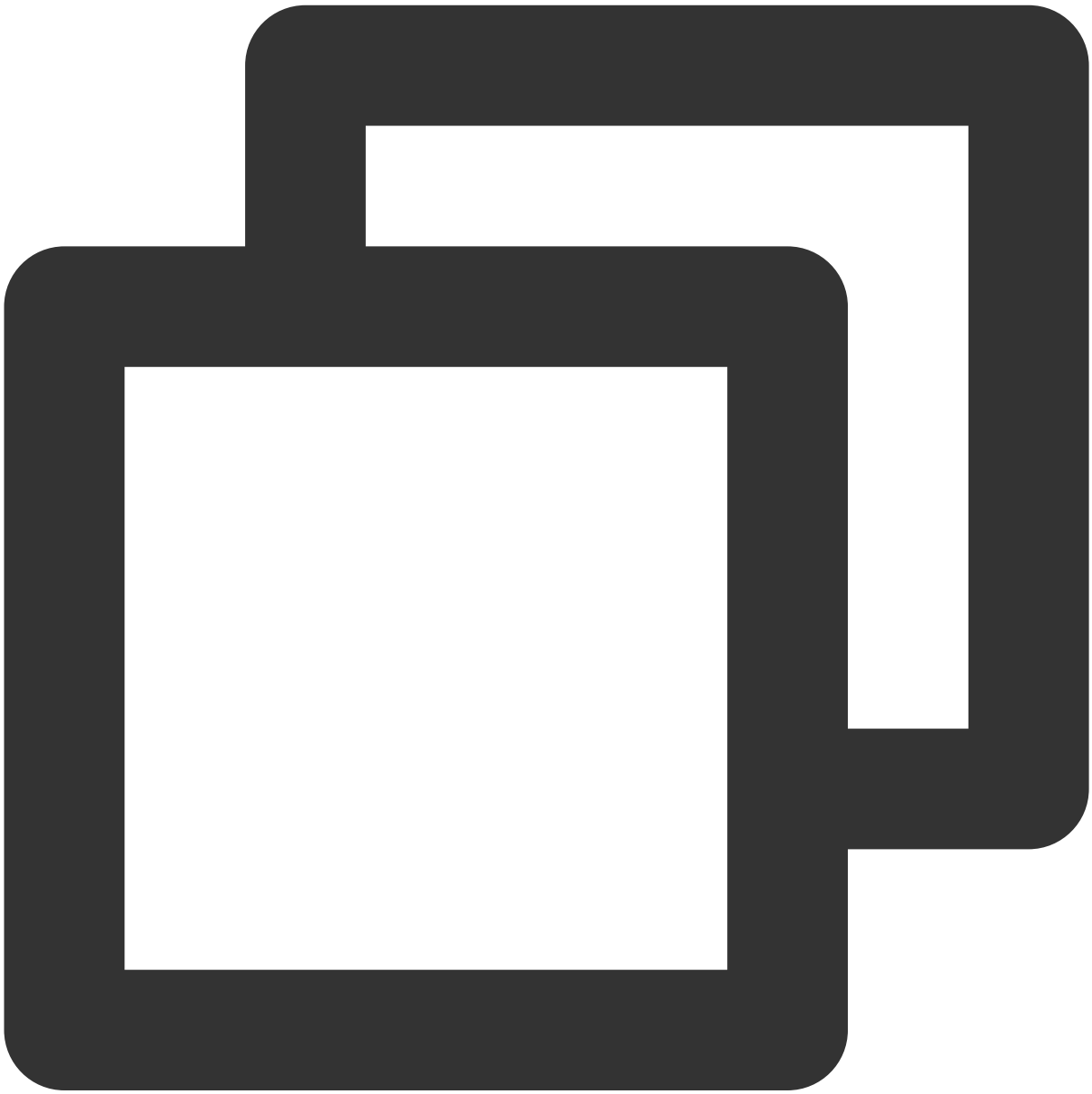


```
<switch wx:for="{{objectArray}}" wx:key="unique" > {{item.id}} </switch>
<button bindtap="switch"> Switch </button>
```

©2013-2022 Tencent Cloud. All rights reserved.

```
<button bindtap="addToFront"> Add to the front </button>


<switch wx:for="{{numberArray}}" wx:key="*this" > {{item}} </switch>
<button bindtap="addNumberToFront"> Add Number to the front </button>
```



```
Page({
  data: {
    objectArray: [
      {id: 5, unique: 'unique_5'},
      {id: 4, unique: 'unique_4'},
```

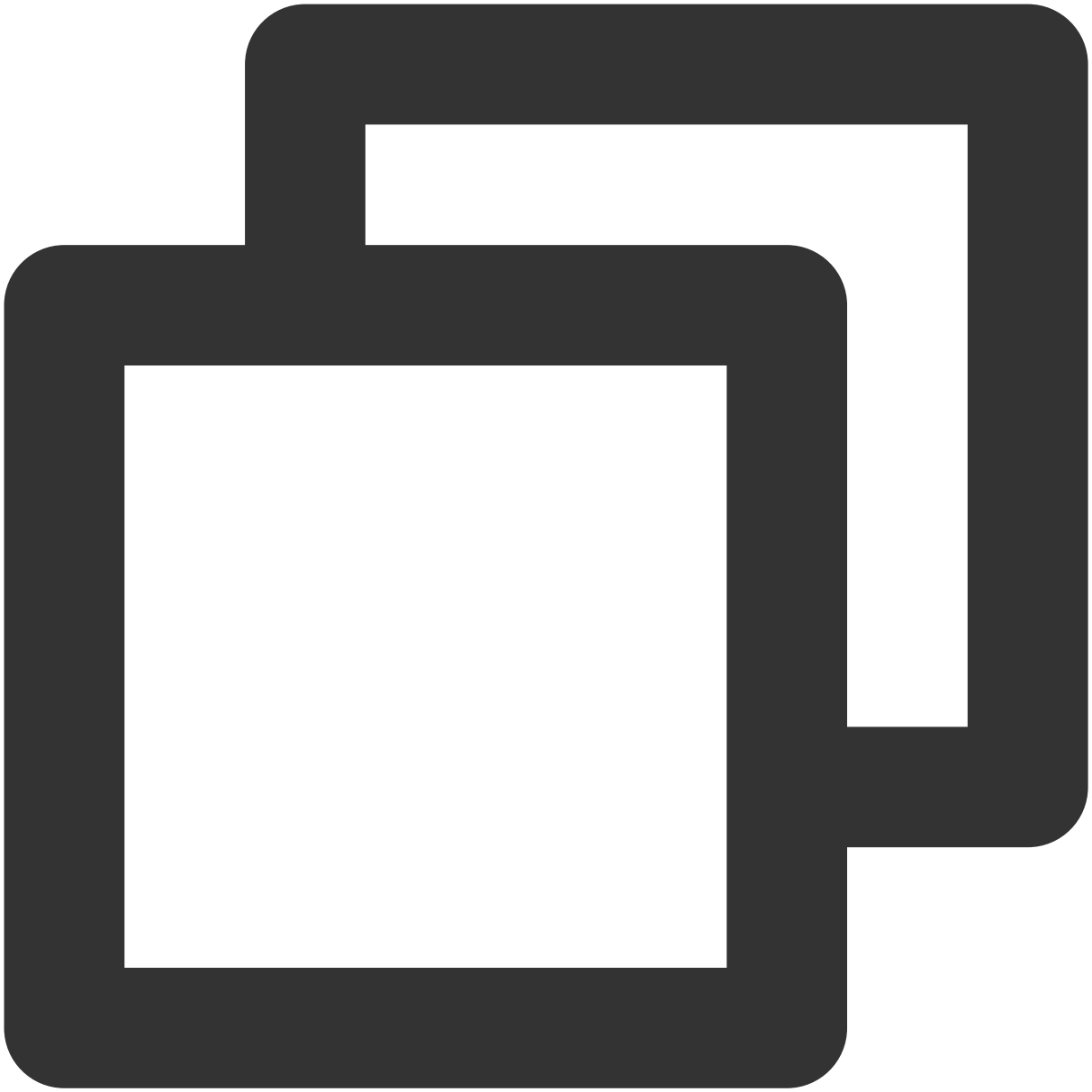
©2013-2022 Tencent Cloud. All rights reserved.

```
      {id: 3, unique: 'unique_3'},
      {id: 2, unique: 'unique_2'},
      {id: 1, unique: 'unique_1'},
      {id: 0, unique: 'unique_0'},
    ],
    numberArray: [1, 2, 3, 4]
  },
  switch: function(e) {
    const length = this.data.objectArray.length
    for (let i = 0; i < length; ++i) {
      const x = Math.floor(Math.random() * length)
      const y = Math.floor(Math.random() * length)
      const temp = this.data.objectArray[x]
      this.data.objectArray[x] = this.data.objectArray[y]
      this.data.objectArray[y] = temp
    }
    this.setData({
      objectArray: this.data.objectArray
    })
  },
  addToFront: function(e) {
    const length = this.data.objectArray.length
    this.data.objectArray = [{id: length, unique: 'unique_' + length}].concat(this.
    this.setData({
      objectArray: this.data.objectArray
    })
  },
  addNumberToFront: function(e){
    this.data.numberArray = [ this.data.numberArray.length + 1 ].concat(this.data.n
    this.setData({
      numberArray: this.data.numberArray
    })
  }
})
```

Templates

WXML provides templates, where you can define code snippets and then call them in different places. Use the `name` attribute to specify the name of the template. Then define that code snippet inside <template/>:

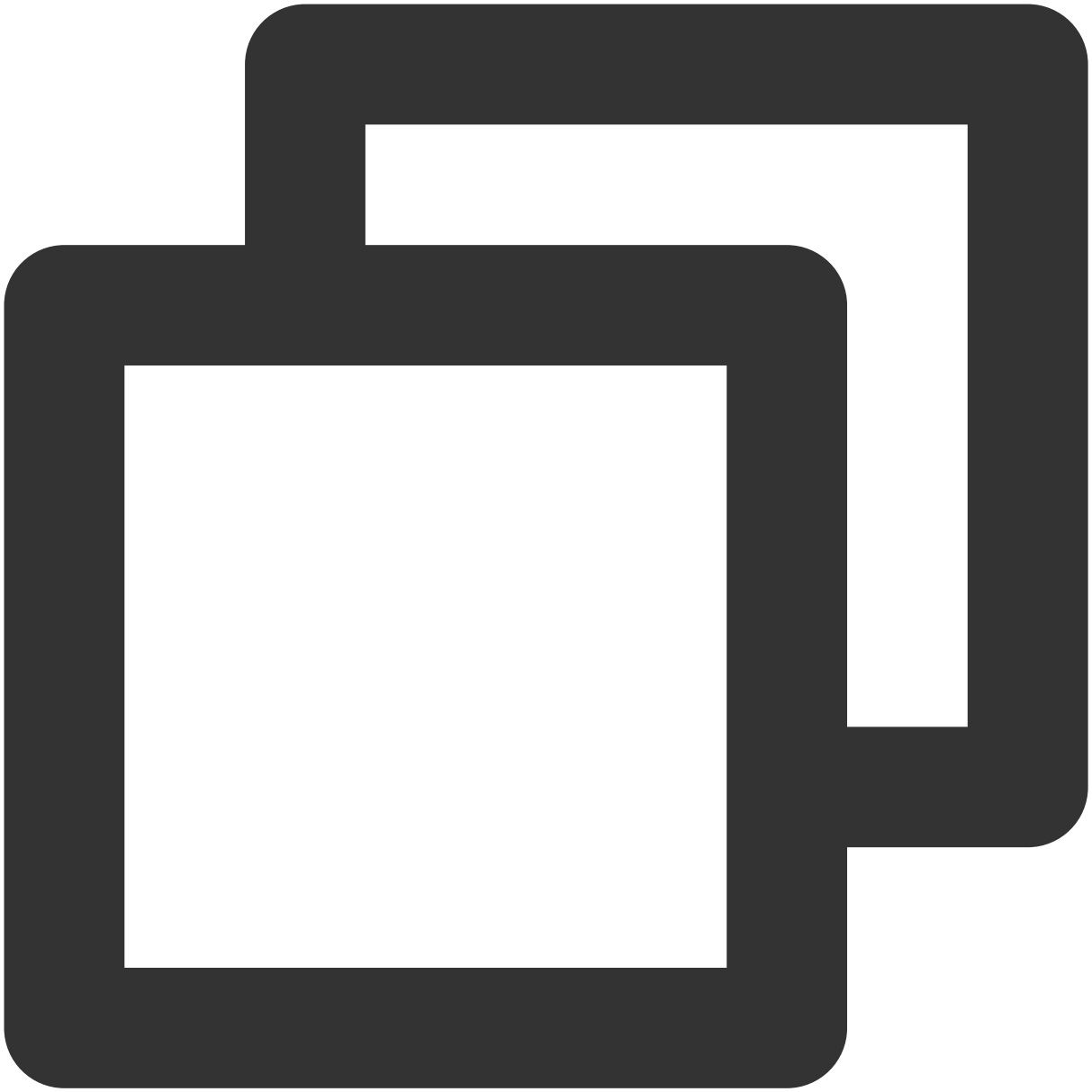
©2013-2022 Tencent Cloud. All rights reserved.



```
<template name="msgItem">
  <view>
    <text> {{index}}: {{msg}} </text>
    <text> Time: {{time}} </text>
  </view>
</template>
```

Use the `is` attribute to declare the template to be used. Then, pass the data needed by the template. For example:

©2013-2022 Tencent Cloud. All rights reserved.



```
<!--
item: {
  index: 0,
  msg: 'this is a template',
  time: '2016-06-18'
}
-->


<template name="msgItem">
  <view>
```

©2013-2022 Tencent Cloud. All rights reserved.

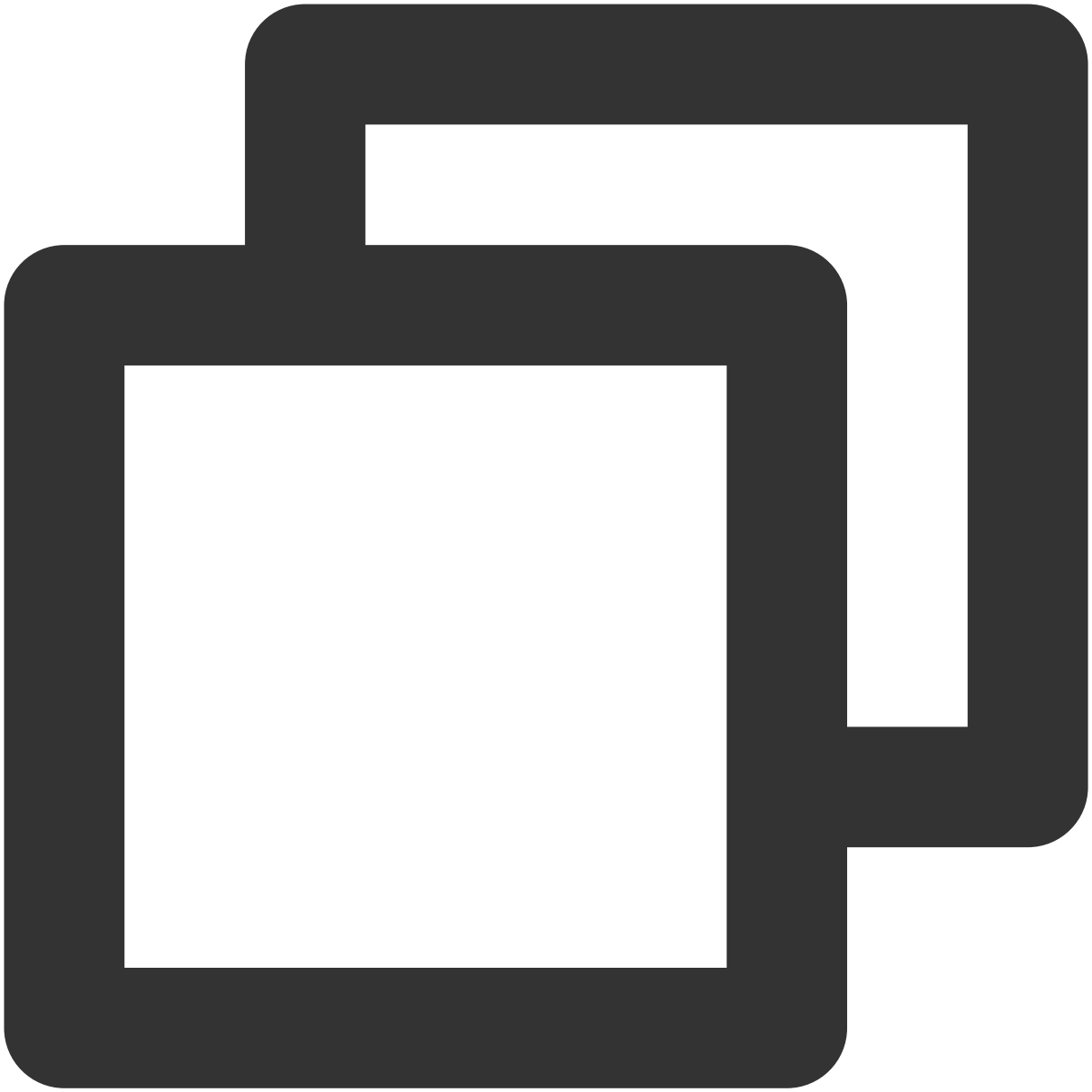
```
    <text> {{index}}: {{msg}} </text>
    <text> Time: {{time}} </text>
  </view>
</template>


<template is="msgItem" data="{{...item}}"/>

<!-- Output
0: this is a template Time: 2016-06-18
-->
```

Use `is` to dynamically decide which template to be rendered, as shown below:

©2013-2022 Tencent Cloud. All rights reserved.



```
<template name="odd">
  <view> odd </view>
</template>


<template name="even">
  <view> even </view>
</template>


<block wx:for="{{[1, 2, 3, 4, 5]}}">
```

©2013-2022 Tencent Cloud. All rights reserved.

```
  <template is="{{item % 2 == 0 ? 'even' : 'odd'}}"/>
</block>



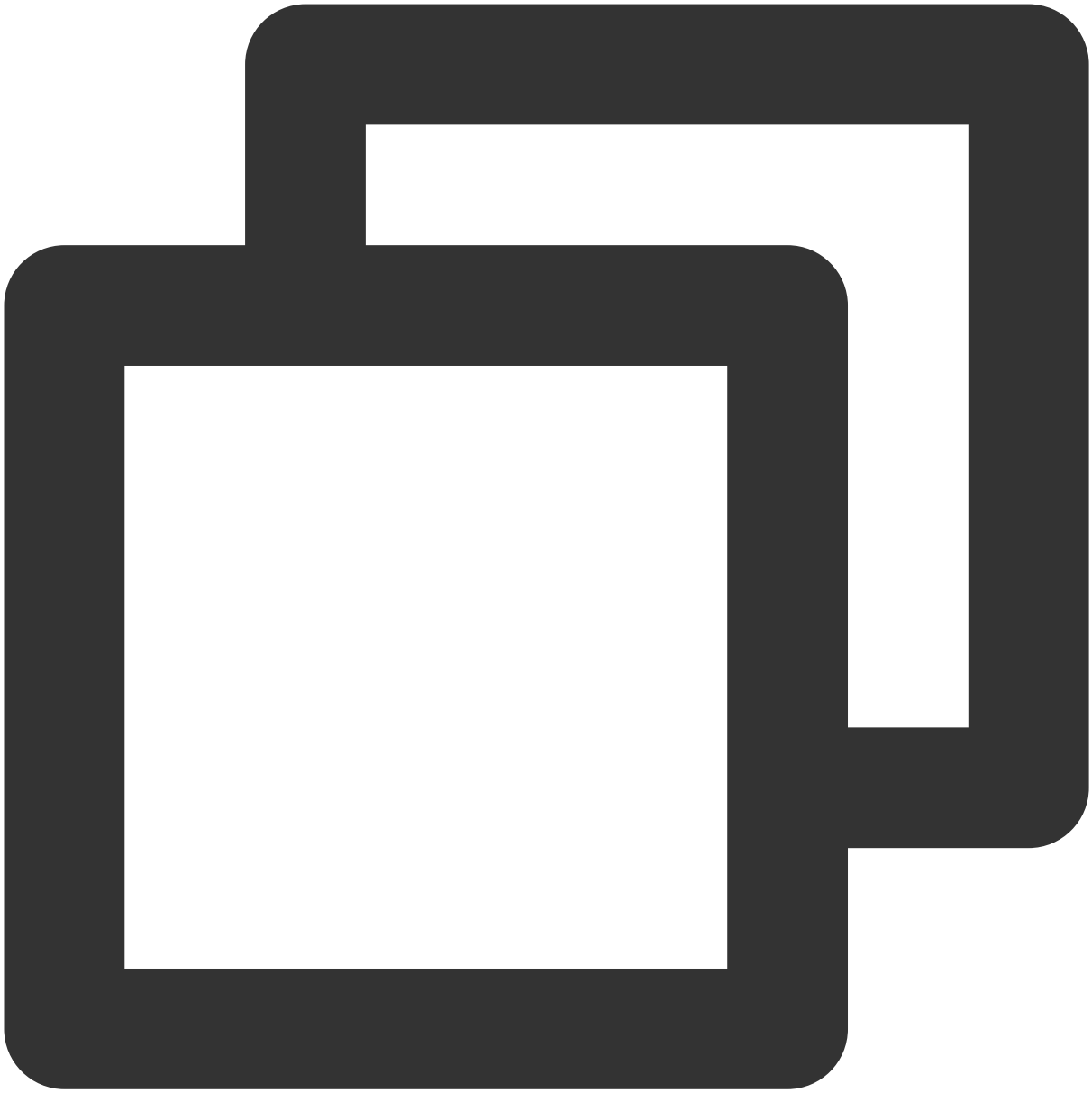
<!-- Output
odd
even
odd
even
odd
-->
```

Reference

WXML provides two referencing methods `import` and `include`.

`import` can use the template defined by the target file in the current file, for example, a template named `item` is defined in `item.wxml`:

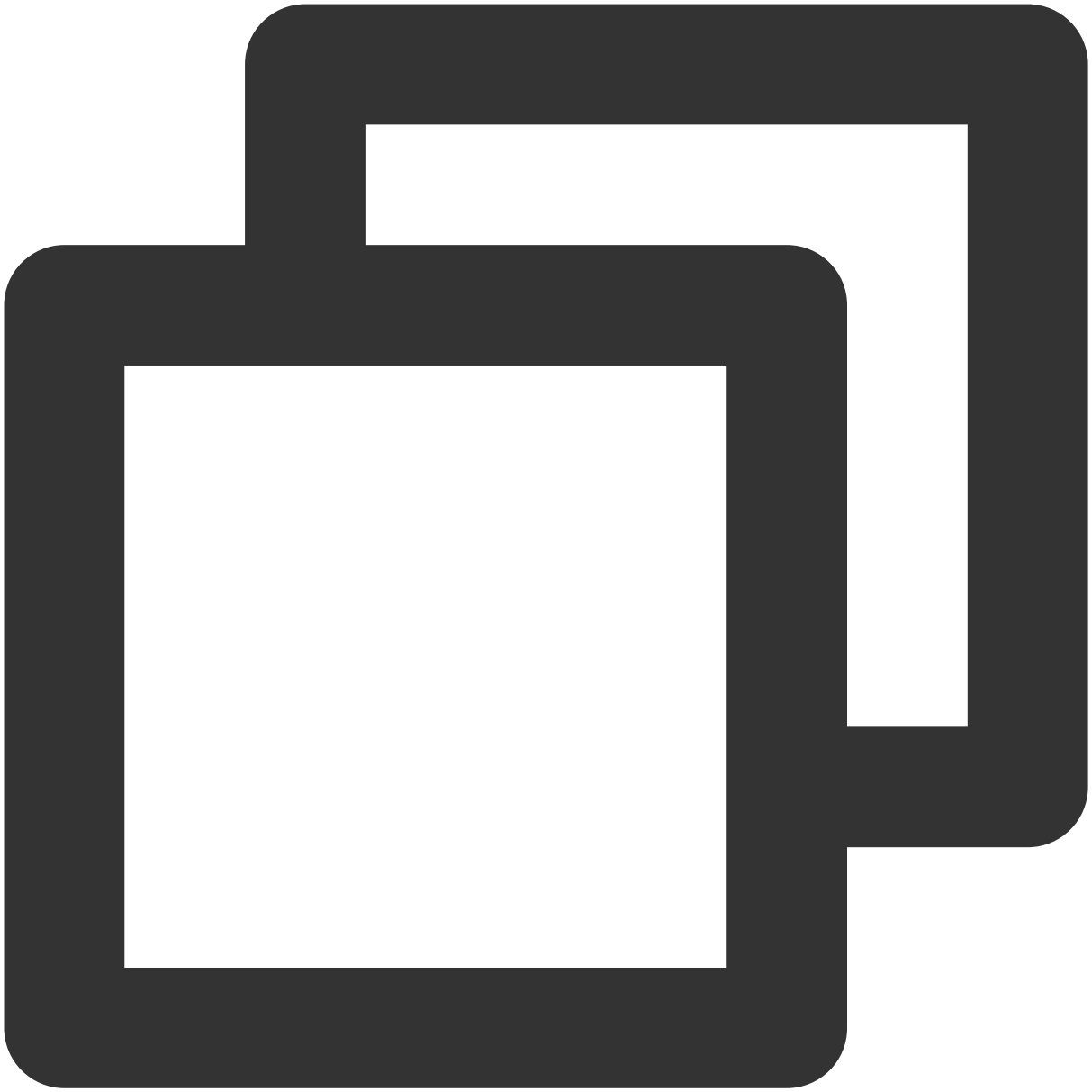
©2013-2022 Tencent Cloud. All rights reserved.



```
<!-- item.wxml -->
<template name="item">
  <text>{{text}}</text>
</template>
```

By referencing `item.wxml` in `index.wxml`, you can use the `item` template:

©2013-2022 Tencent Cloud. All rights reserved.



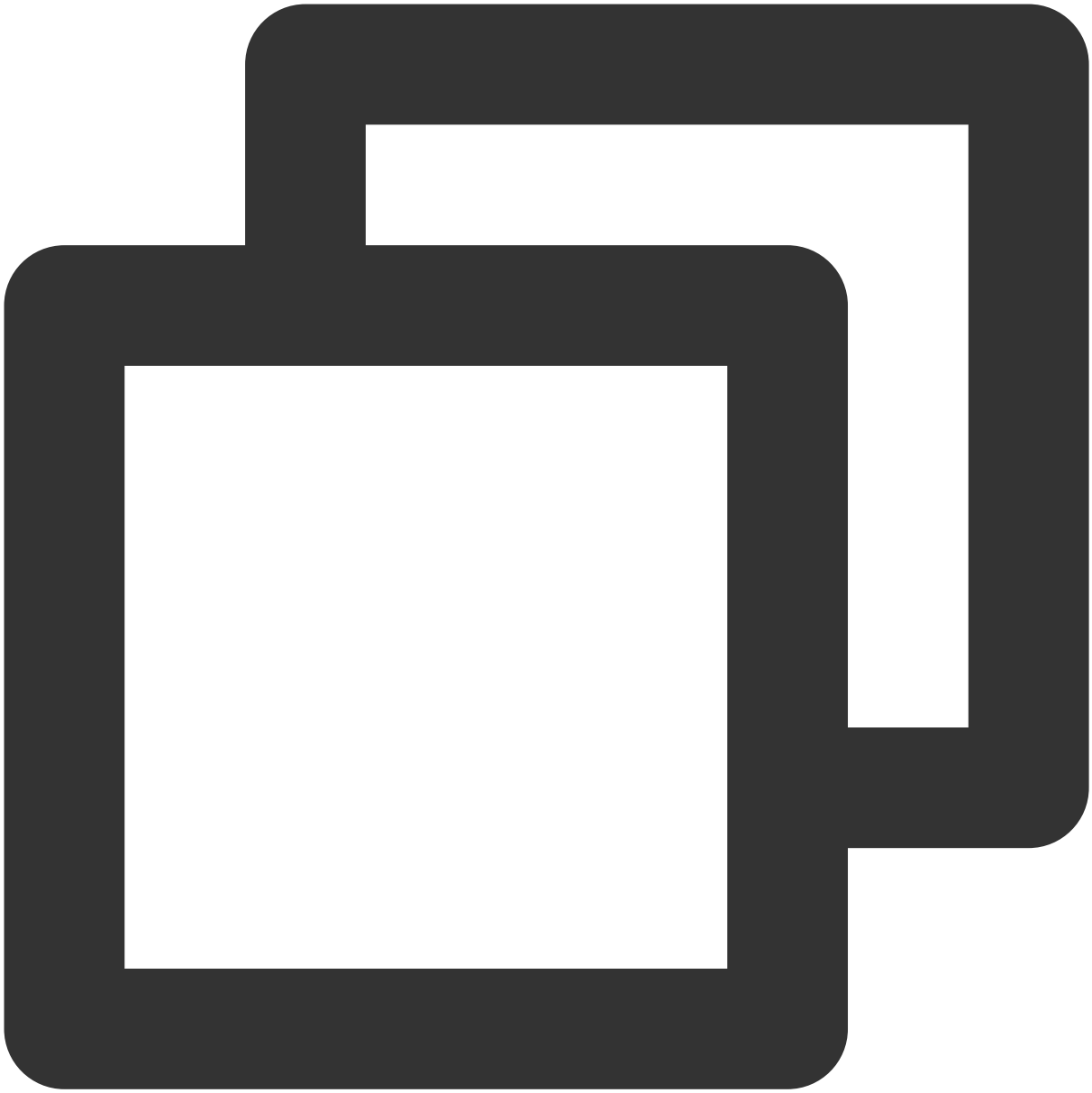
```
<import src="item.wxml"/>

<template is="item" data="{{text: 'forbar'}}"/>
```

The concept of "scope" is used for `import` references. This means that the reference will import the template defined in the target file but not the template imported by the target file.

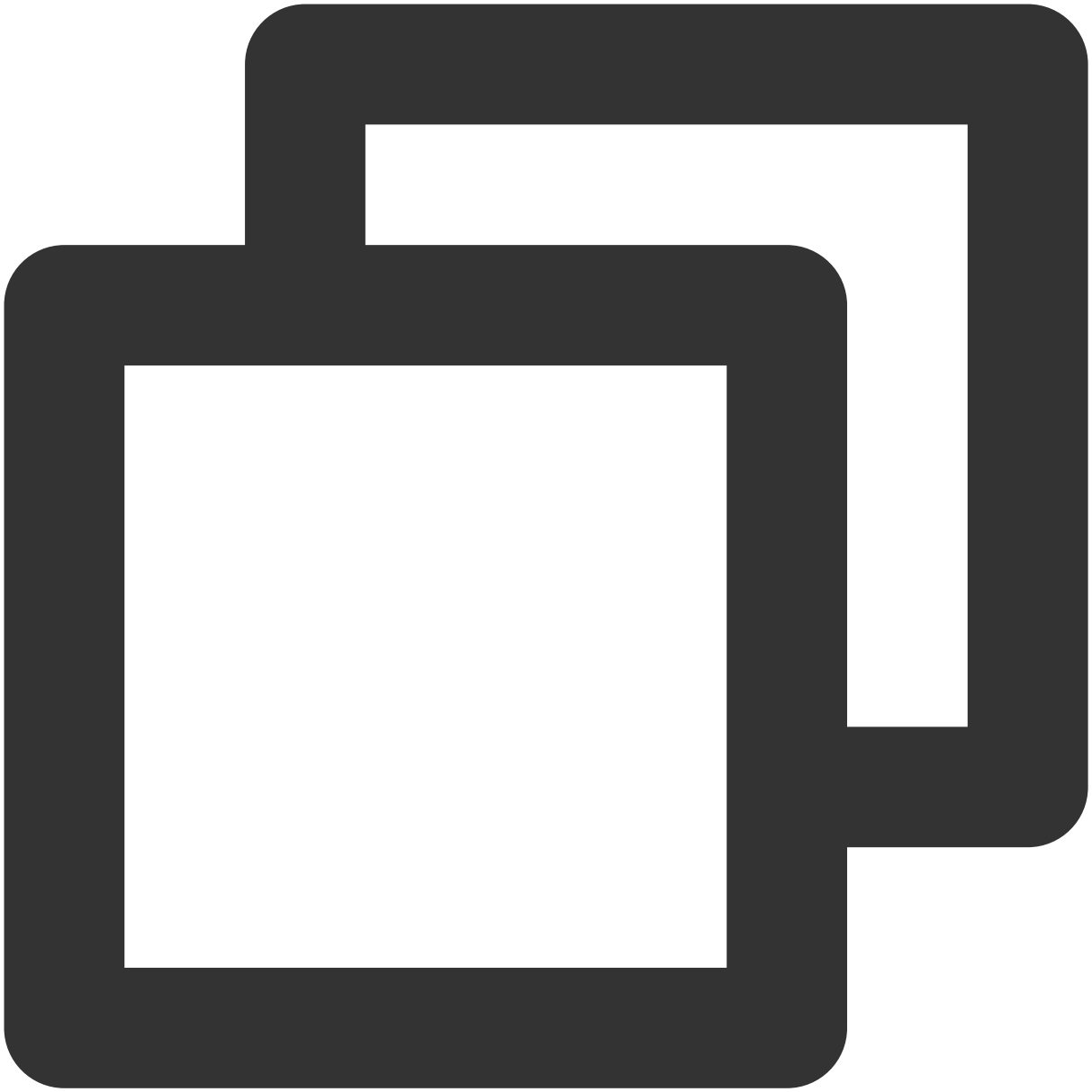
For example, C references B and B references A, and the template defined by B can be used in C, and the template defined by A can be used in B, but C cannot use the template defined by A, as shown below:

©2013-2022 Tencent Cloud. All rights reserved.



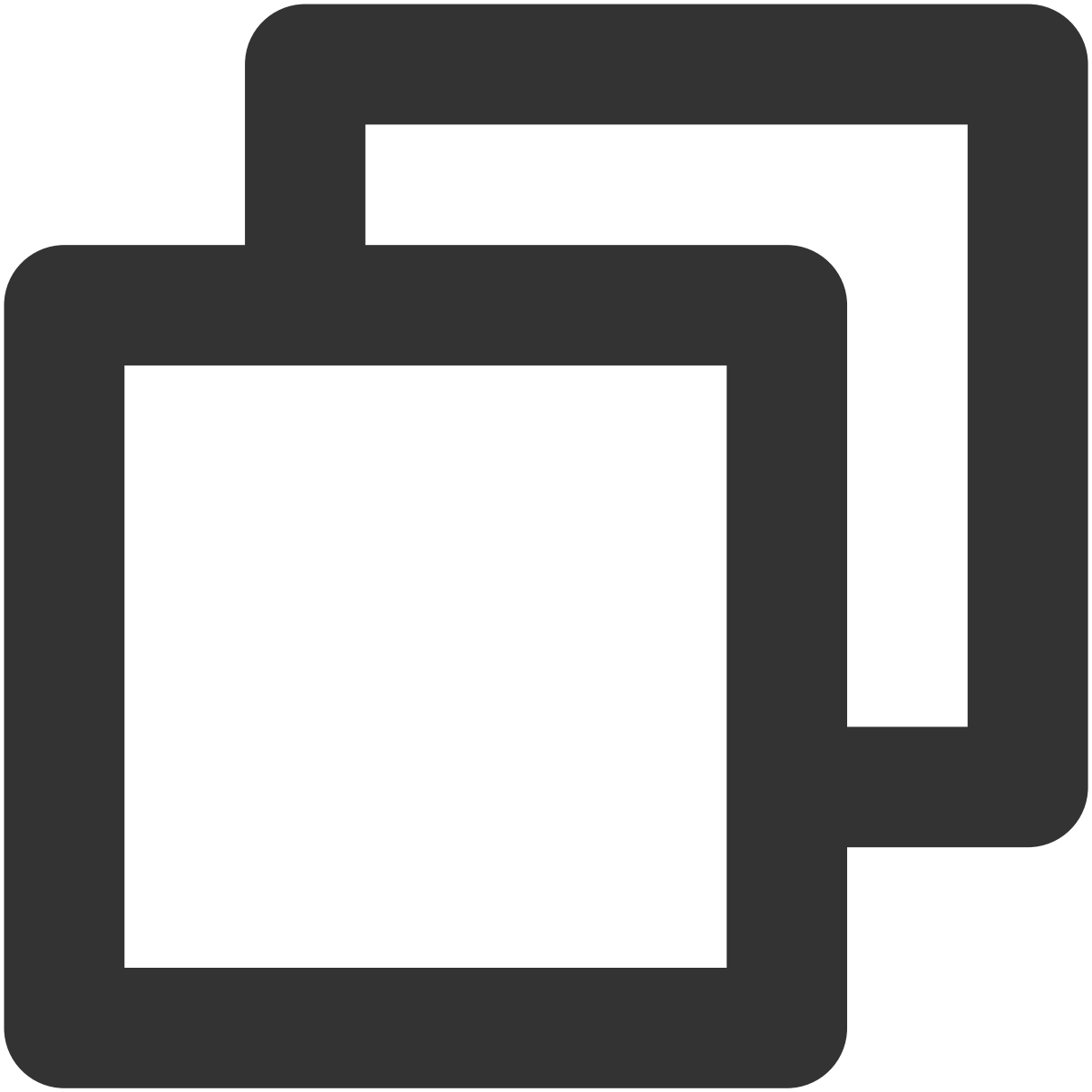
```
<!-- A.wxml -->
<template name="A">
  <text> A template </text>
</template>
```

©2013-2022 Tencent Cloud. All rights reserved.



```
<!-- B.wxml --><import src="a.wxml"/><template name="B">  <text> B template </text>
```

©2013-2022 Tencent Cloud. All rights reserved.



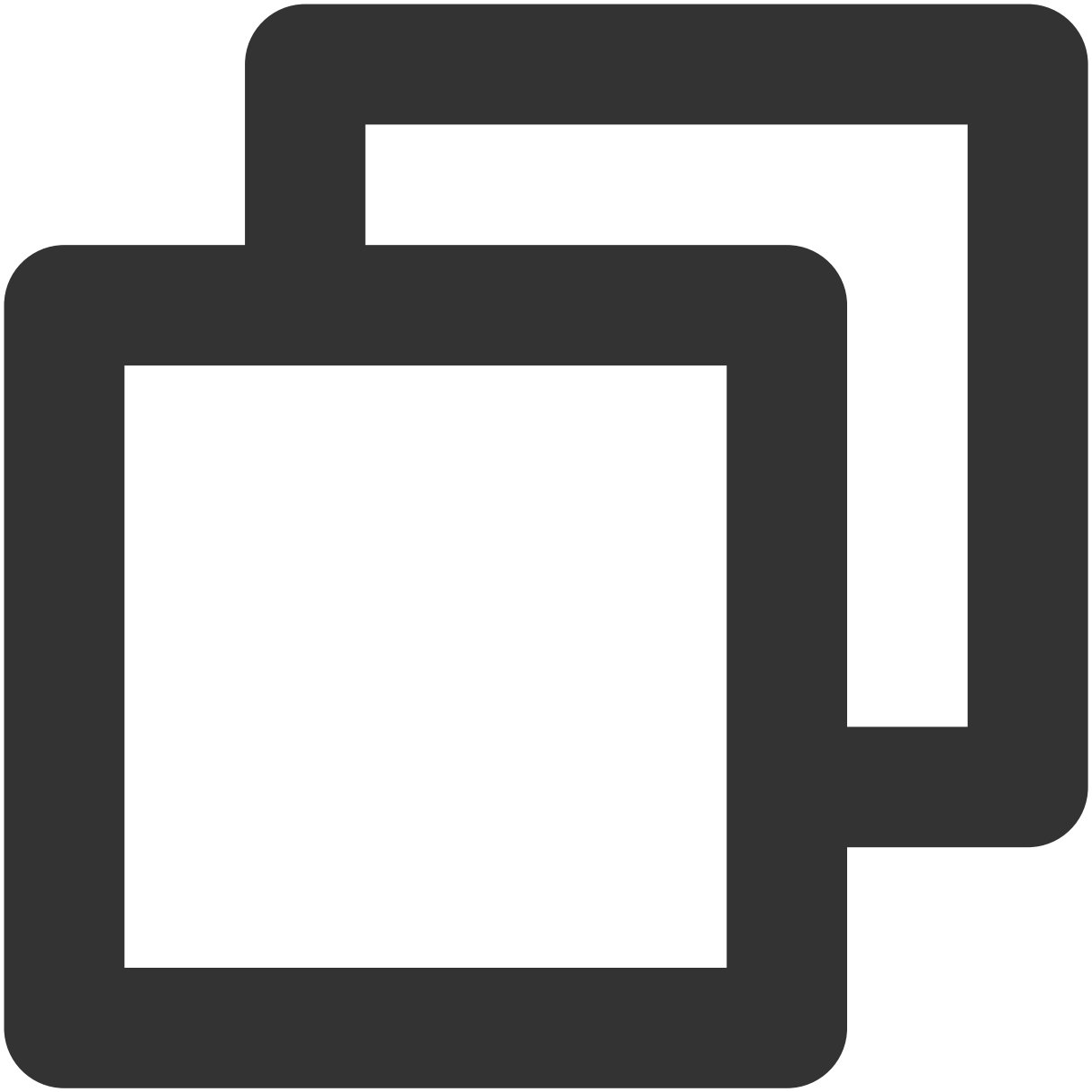
```
<!-- C.wxml -->
<import src="b.wxml"/>

<template is="A"/>  <!-- An alarm will be triggered, as template A is not defined i

<template is="B"/>
```

`include` can introduce the entire code excluding `<template/> <wxs/>` . This is equivalent to copying the code to the location of `include` . For example:

©2013-2022 Tencent Cloud. All rights reserved.

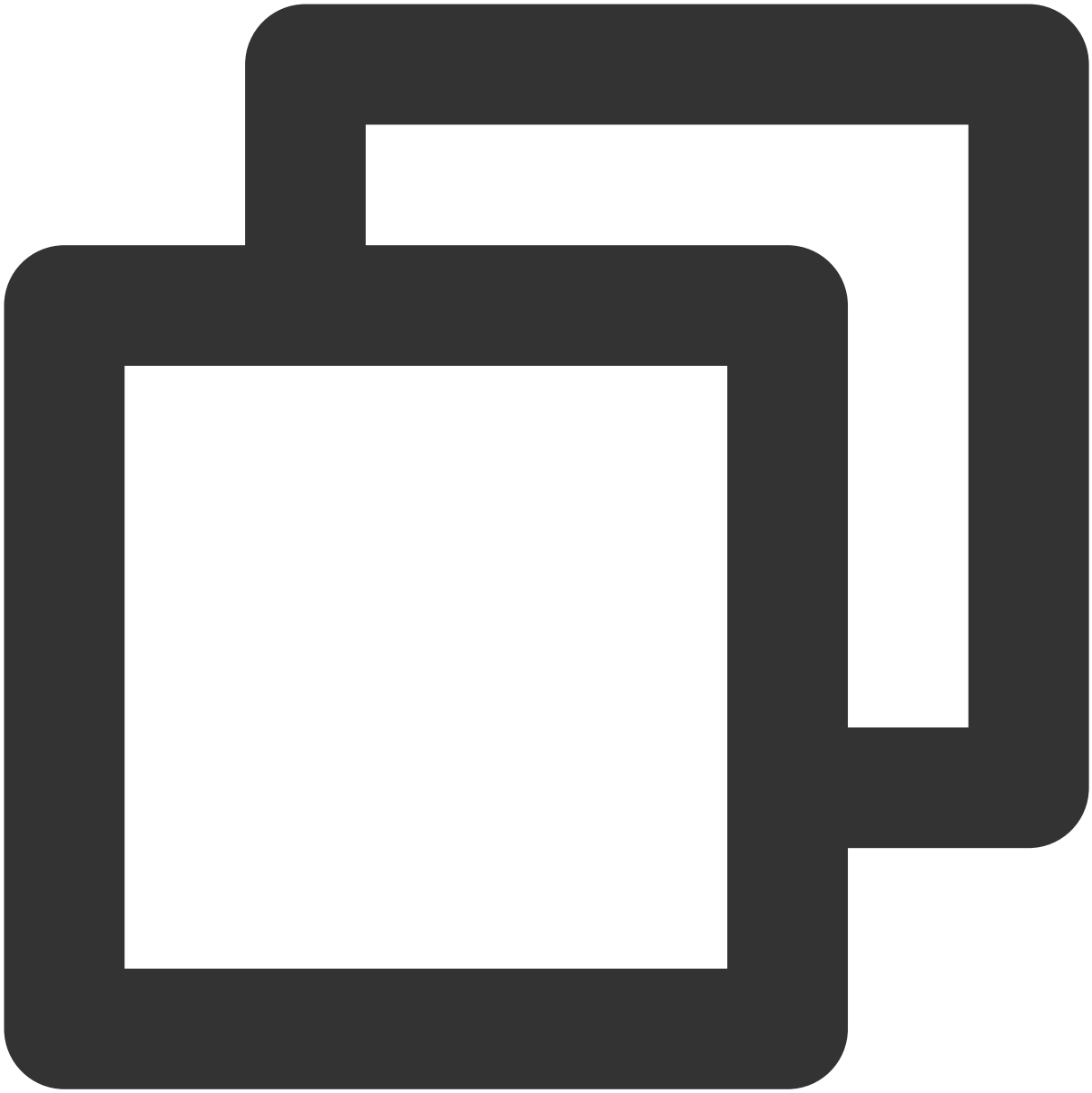


```
<!-- index.wxml -->
<include src="header.wxml"/>

<view> body </view>

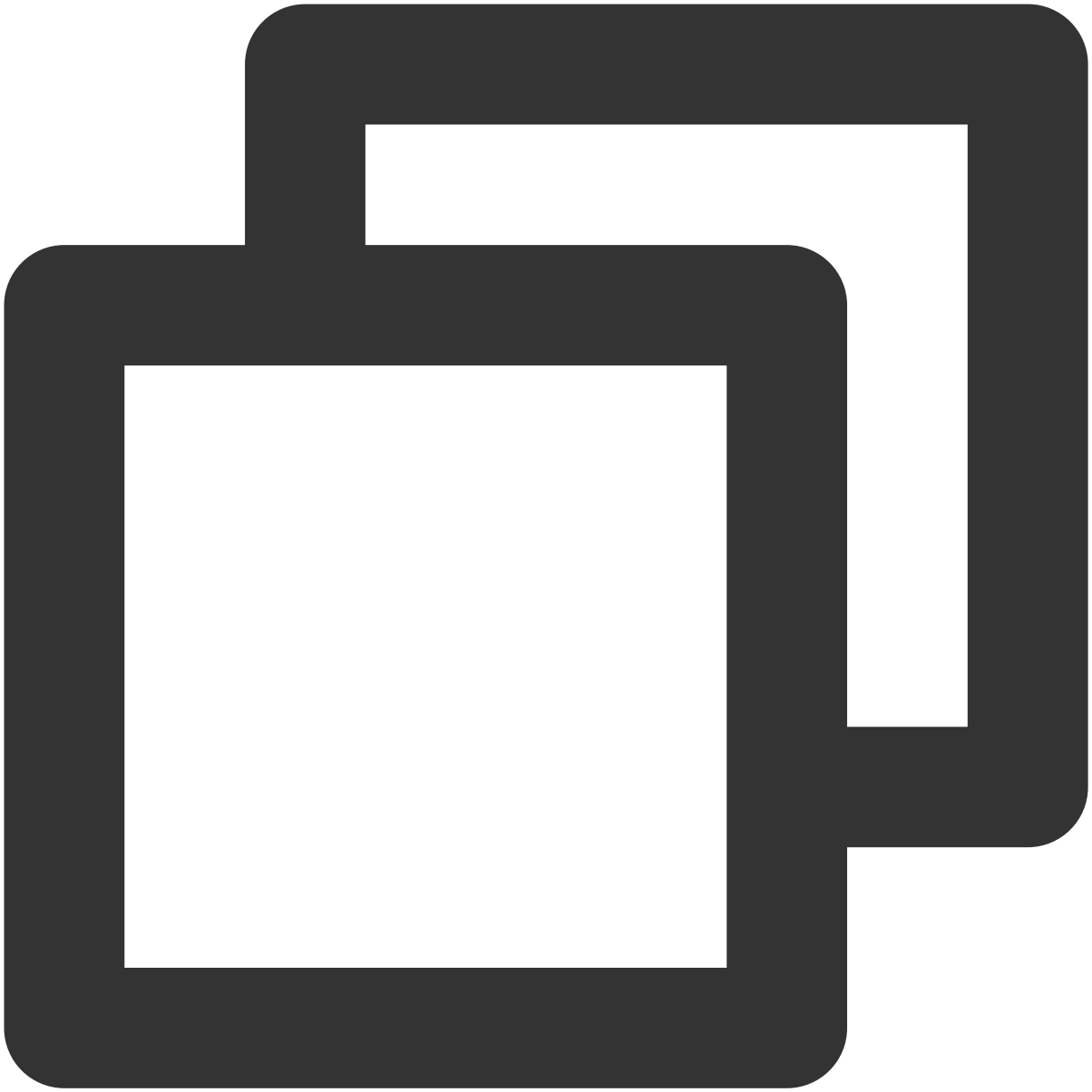
<include src="footer.wxml"/>
```

©2013-2022 Tencent Cloud. All rights reserved.



```
<!-- header.wxml -->
<view> header </view>
```

©2013-2022 Tencent Cloud. All rights reserved.



```
<!-- footer.wxml -->
<view> footer </view>
```

Common attributes

The attributes that are supported by all wxml tags are common attributes.

Attribute	Type	Definition	Description

©2013-2022 Tencent Cloud. All rights reserved.

id	String	Unique identifier of the component	Unique for entire page
class	String	Style class for the component	The style class defined in the corresponding WXSS
style	String	Inline style of the component	The inline style that can be set dynamically
hidden	Boolean	Whether the component is displayed	Default display for all components
data-*	Any	Custom attribute	When an event is triggered on the component, it will be sent to the event handler function.
bind*/catch*	EventHandler	Component Event	-

©2013-2022 Tencent Cloud. All rights reserved.

WXSS Style

Last updated：2024-05-10 11:25:23

WeChat Style Sheets (WXSS) is a style language used to describe the style of WXML component styles.
WXSS is similar to CSS in Web development. In order to be more suitable for mini program development, WXSS has been extended and modified based on CSS.

File composition

app.json

```
{
  "pages":[
    "pages/rpx/index",
    "pages/box/index",
    "pages/inlinestyle/index",
    "pages/flex/index"
  ],
  "window":{
    "backgroundTextStyle":"light",
    "navigationBarBackgroundColor": "#fff",
    "navigationBarTitleText": " ",
    "navigationBarTextStyle":"black"
  }
}
```

Project common style: The `app.wxss` in the root directory is the project common style and will be injected into every page of mini programs. Page style: A WXSS file with the same name and at the same level as the page registered with app.json. In the above example, `pages/rpx/index` is registered and the file `pages/rpx/index.wxss` is the style for page `pages/rpx/index.wxml` .
Other styles: Other styles can be referenced by project common styles and page styles. For referencing methods, see WXSS Reference section in this document.
Unlike web development, you don't need to optimize the number of style files requested for mini program development. You just focus on the code organization. Style files will go through compilation optimization and detailed compilation will be introduced later.

©2013-2022 Tencent Cloud. All rights reserved.

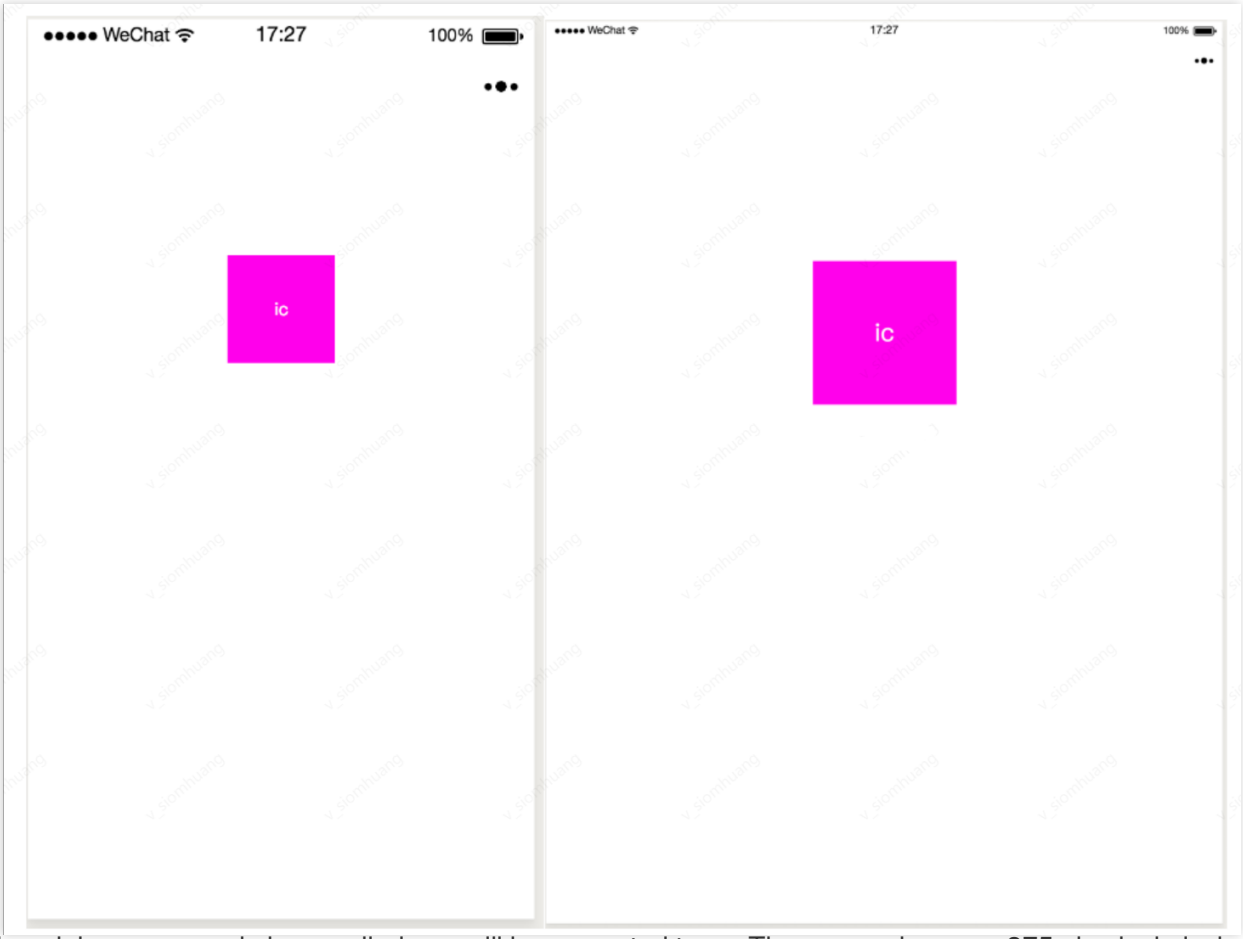
Dimension unit

In WXSS, the rpx (responsive pixel) was introduced. You can use this to adapt to different screen width to make development easier.
For example, if px is used, a page will have too much white space for the same element displayed under different screen width.



The following shows that the rpx is used:

©2013-2022 Tencent Cloud. All rights reserved.



After the mini program code is compiled, rpx will be converted to px. The conversion uses 375 physical pixels as a standard, which means 1 rpx = 1 px on a screen with 375 physical pixels width.

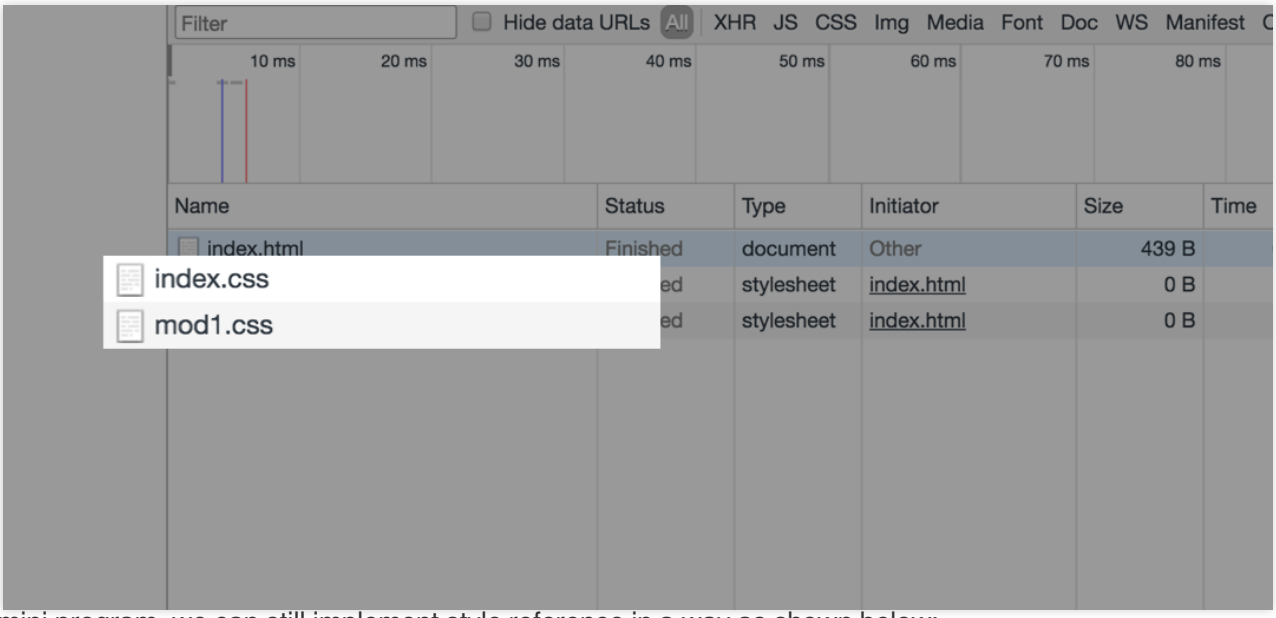
For example, on iPhone 6, the screen width is 375 px, and there are 750 physical pixels in total, then 1 rpx = 375 / 750 physical pixels = 0.5 px.

WXSS reference

In CSS, developers can reference another style file like this: @import url ('./ test_0.css')

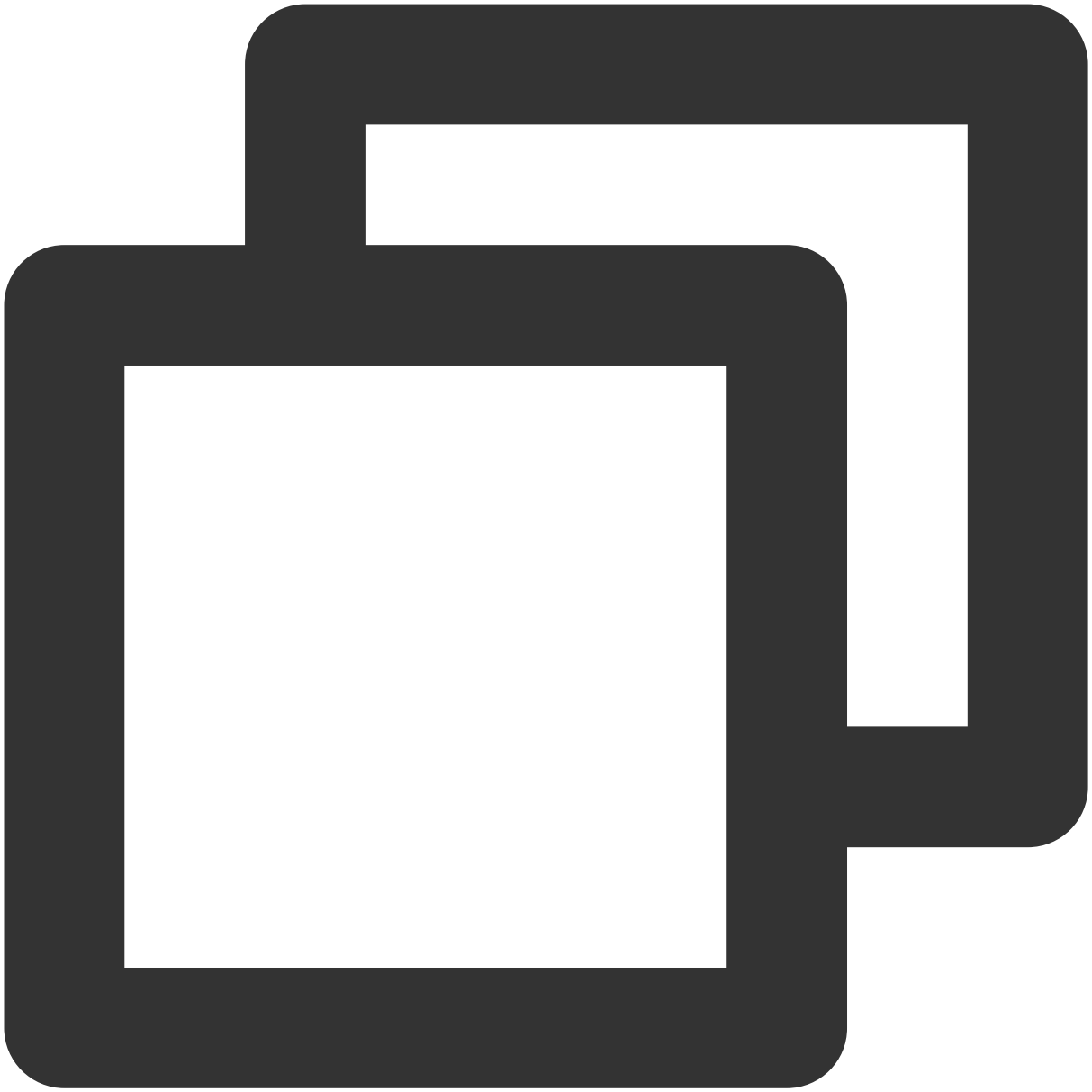
This does not merge test_0.css into index.css on the request, which means that when index.css is requested, there will be an additional request for test_0.css.

©2013-2022 Tencent Cloud. All rights reserved.



In the mini program, we can still implement style reference in a way as shown below:

©2013-2022 Tencent Cloud. All rights reserved.



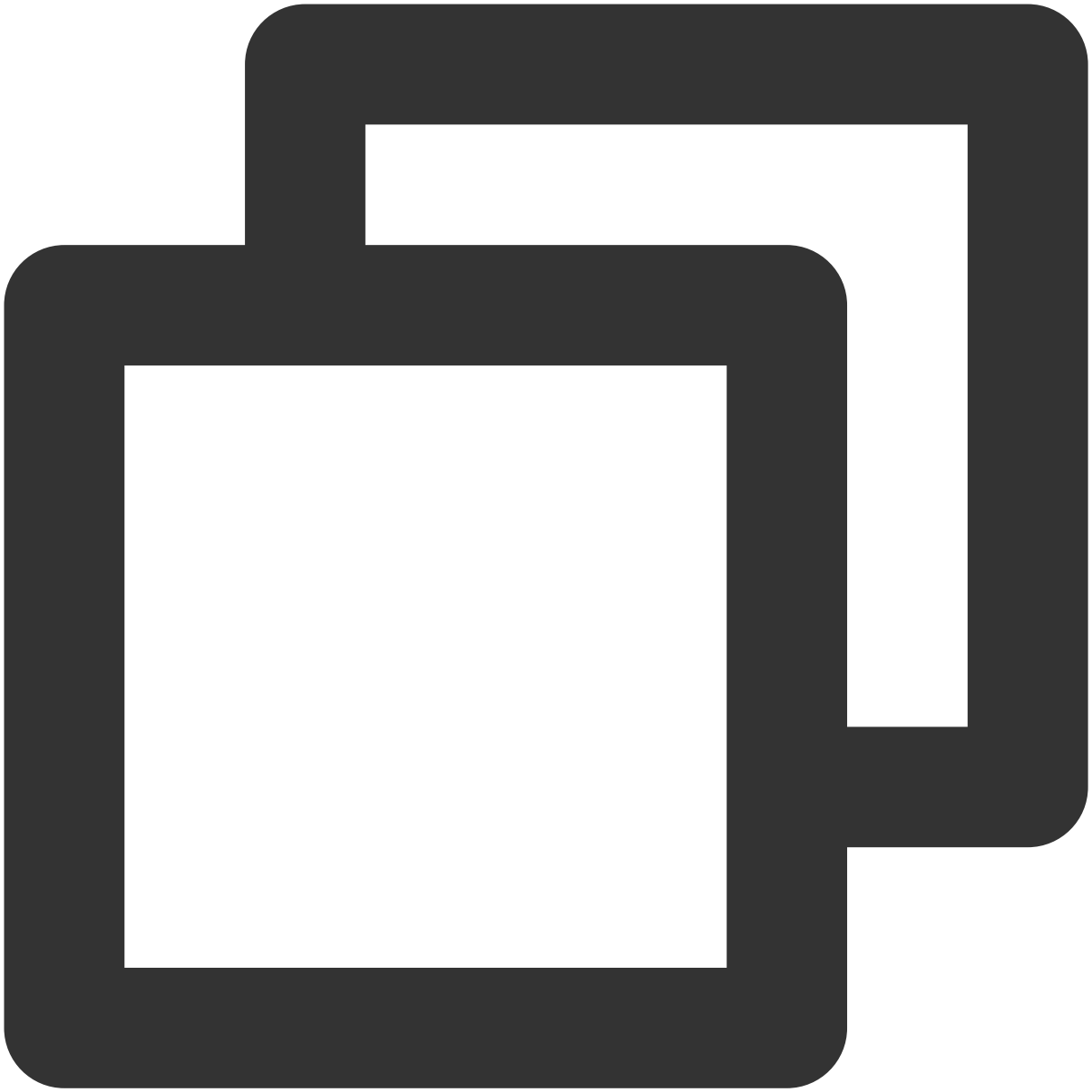
```
@import './test_0.wxss'
```

Since WXSS will be compiled and packaged into the object file, users only need to download it once. And there will be no redundant file requests due to style references during usage.

Inline style

WXSS inline style is consistent with that in Web development:

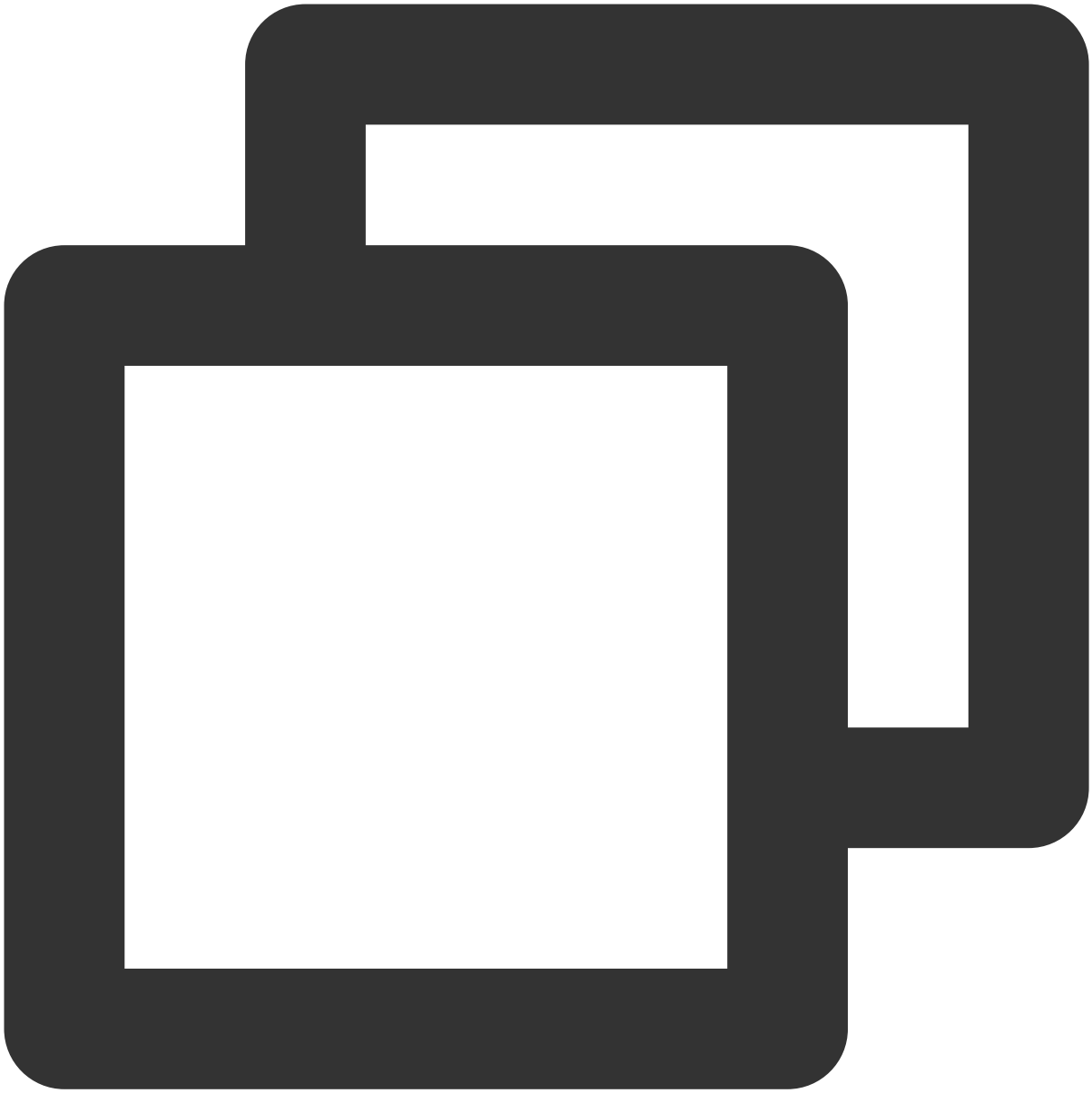
©2013-2022 Tencent Cloud. All rights reserved.



```
<!--index.wxml-->
<!--Inline style-->
<view style="color: red; font-size: 48rpx"></view>
```

Mini programs support dynamically updating inline styles:

©2013-2022 Tencent Cloud. All rights reserved.



```
<!--index.wxml-->

<!-- Dynamically changing inline styles-->
<!--
{
  eleColor: 'red',
  eleFontsize: '48rpx'
}
-->
<view style="color: {{eleColor}}; font-size: {{eleFontsize}}"></view>
```

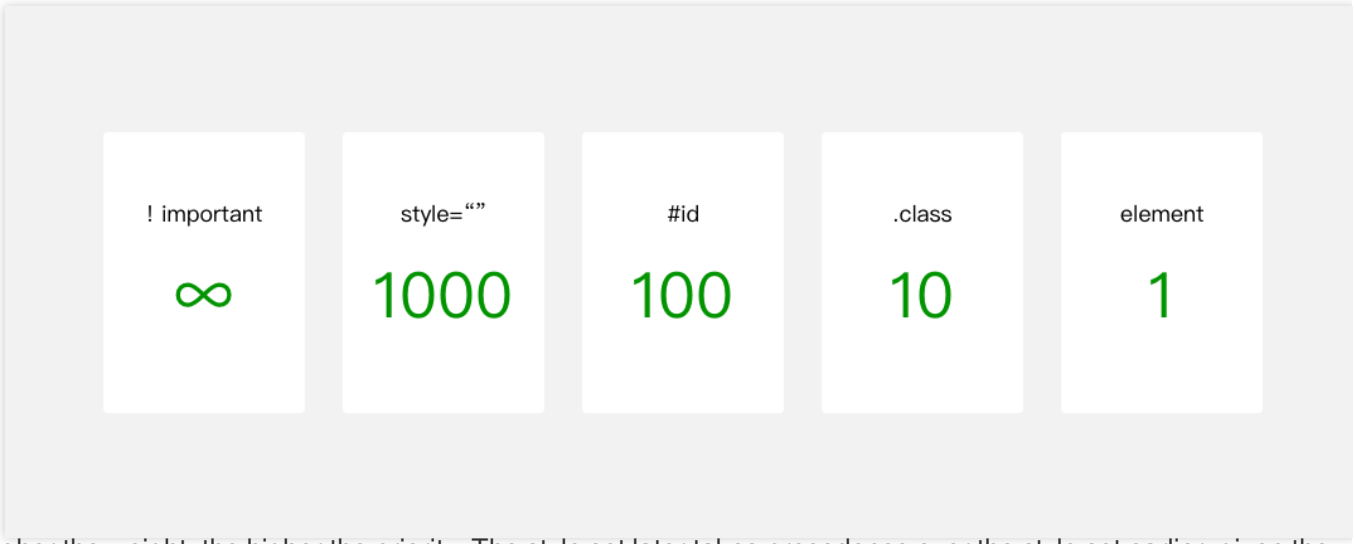
©2013-2022 Tencent Cloud. All rights reserved.

Selectors

The following selectors are supported:

Type	Selector	Example	Description
class selector	.class	.intro	Selects all components with class = "intro"
ID selector	#id	#firstname	Selects the components with id = "firstname"
element selector	element	view checkbox	Selects `view` components of all documents and all `checkbox` components
pseudo-element selector	::after	view::after	Insert content after the `view` component
pseudo-element selector	::before	view::before	Insert content before the `view` component

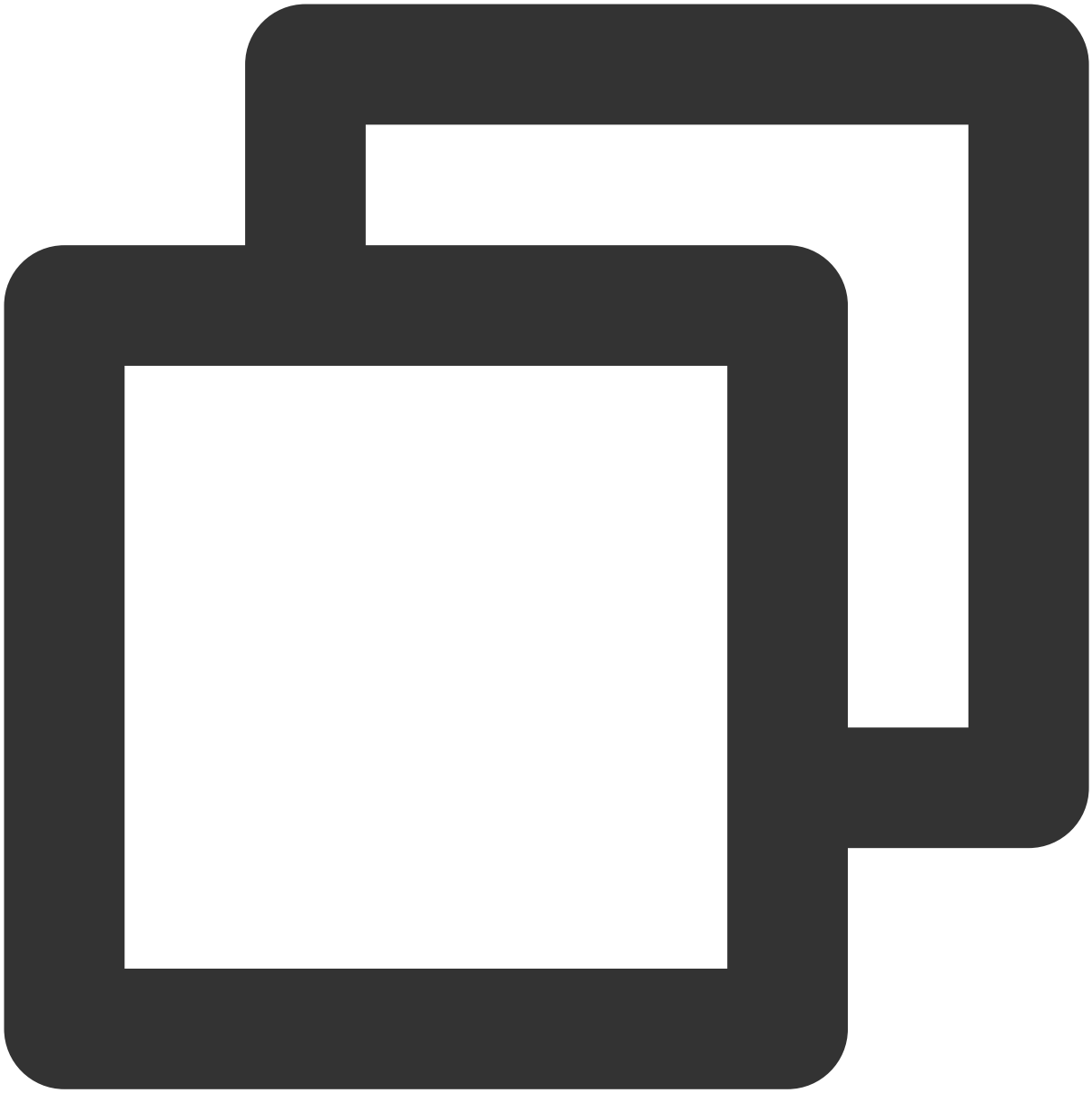
The priority of WXSS is similar to that of CSS, with weights as shown below:



The higher the weight, the higher the priority. The style set later takes precedence over the style set earlier, given the same priority.

The priority weights of WXSS selectors:

©2013-2022 Tencent Cloud. All rights reserved.



```
view{ // Weight is 1
  color: blue
}

.ele{ // Weight is 10
  color: red
}

#ele{ // Weight is 100
  color: pink
}
```

©2013-2022 Tencent Cloud. All rights reserved.

```
view#ele{ //Weight 1 + 100 = 101, highest priority, and element color is orange
  color: orange
}

view.ele{ // Weight is 1 + 10 = 11
  color: green
}
```

Official style library

For most developers, ECMAScript and JavaScript are the same thing, but strictly speaking, they are different. ECMAScript is a standard for scripting language and it is standardized by Ecma International in the document ECMA-262. JavaScript is an implementation of ECMAScript. Knowing this helps you understand that JavaScript in mini program development is different from JavaScript in browsers and in NodeJS.
ECMA-262 specifies components of the ECMAScript language.

©2013-2022 Tencent Cloud. All rights reserved.

JavaScript

Last updated：2024-05-10 11:26:09

The primary programming language for mini programs is JavaScript. You can use JavaScript to develop business logic and call APIs of mini programs to fulfill business requirements.

ECMAScript

For most developers, ECMAScript and JavaScript are the same thing, but strictly speaking, they are different. ECMAScript is a standard for scripting language and it is standardized by Ecma International in the document ECMA-262. JavaScript is an implementation of ECMAScript. Knowing this helps you understand that JavaScript in mini program development is different from JavaScript in browsers and in NodeJS.
ECMA-262 specifies components of the ECMAScript language:
1. Syntax
2. Types
3. Statements
4. Keywords
5. Operators
6. Objects
JavaScript components in web browsers:

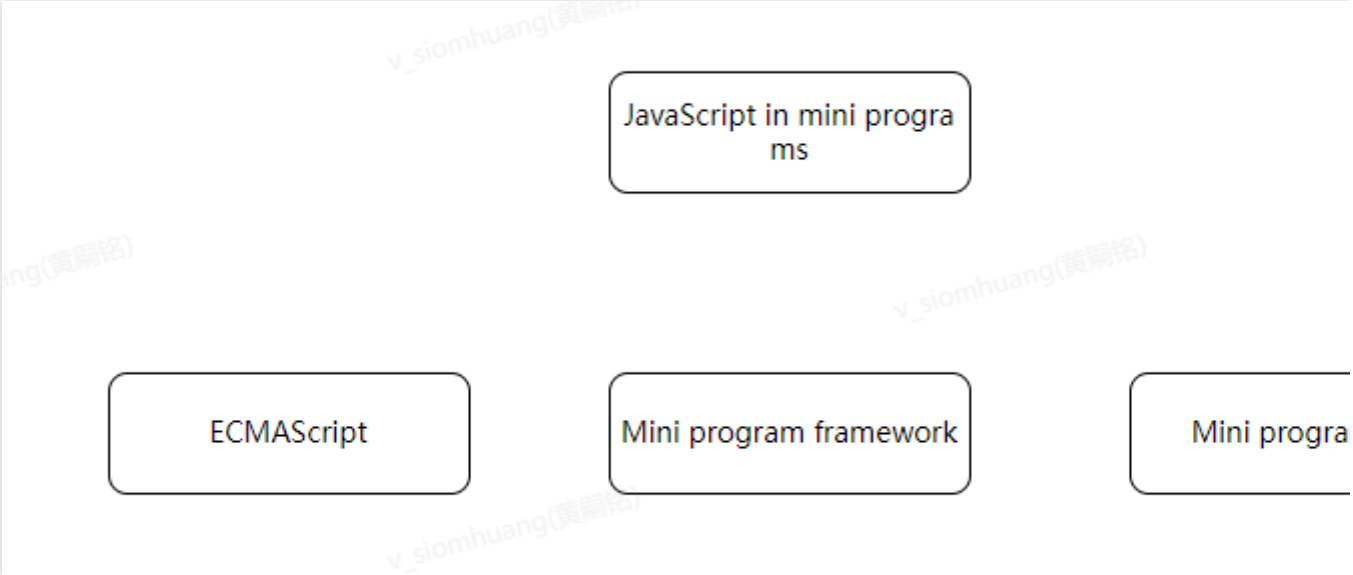


JavaScript in browsers has three important components: ECMAScript, Browser Object Model (BOM) and Document Object Model (DOM). These two object models, which are very familiar to web developers, allow developers to manipulate browser performances, such as modify URLs, modify page rendering, and record data. The JavaScript composition in NodeJS is shown in the following figure:

©2013-2022 Tencent Cloud. All rights reserved.



JavaScript in NodeJS is composed of ECMAScript, NPM and Native modules. NodeJS developers are very familiar with the NPM, the package manager. They can use various packages to implement features and use some native modules, such as FS, HTTP, OS, to have some capabilities that the language itself does not have.
However, different from the above mentioned environment, the mini program JavaScript compositions are as follows:



The JavaScript in mini programs is implemented by ECMAScript, as well as the mini program framework and APIs. Compared with JavaScript in browsers, there are no BOM or DOM objects, so JavaScript libraries such as JQuery and Zepto cannot run in mini programs. Similarly, due to the lack of Native modules and NPM package manager, neither native libraries can be loaded in mini programs nor most NPM packages can be directly used.

Mini program runtime environment

After understanding that JavaScript in mini programs is different from that in browsers and NodeJS, you also should be aware that the script execution environment for mini programs on different platforms is also different.
Mini programs can now run on three platforms:
iOS, including iOS9,10 and 11
Android
Mini program IDE

©2013-2022 Tencent Cloud. All rights reserved.

The ECMAScript standards are different for mini program implementation in these three platforms. Up to now, there are a total of seven versions of ECMAScript standard. Currently, most developers use ECMAScript 5 and ECMAScript 6 standards. However, for mini programs, the runtime environment used by iOS9 and iOS10 is not fully compatible with ECMAScript 6 standard. Some syntax and keywords specified in ECMAScript 6 are not available or different from the standard, such as:

Arrow functions

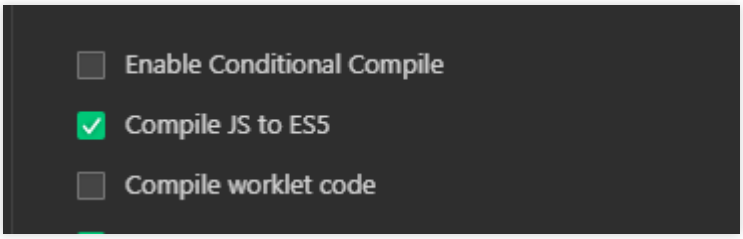
let const

Template strings

…

You may find that some code has syntax errors on mobile phones with earlier operating systems. In order to solve this problem, the mini program IDE provides syntax transcoding tools to help you convert ECMAScript 6 code into ECMAScript 5 code.

You need to check converting ES6 to ES5 in the project settings to use this feature.

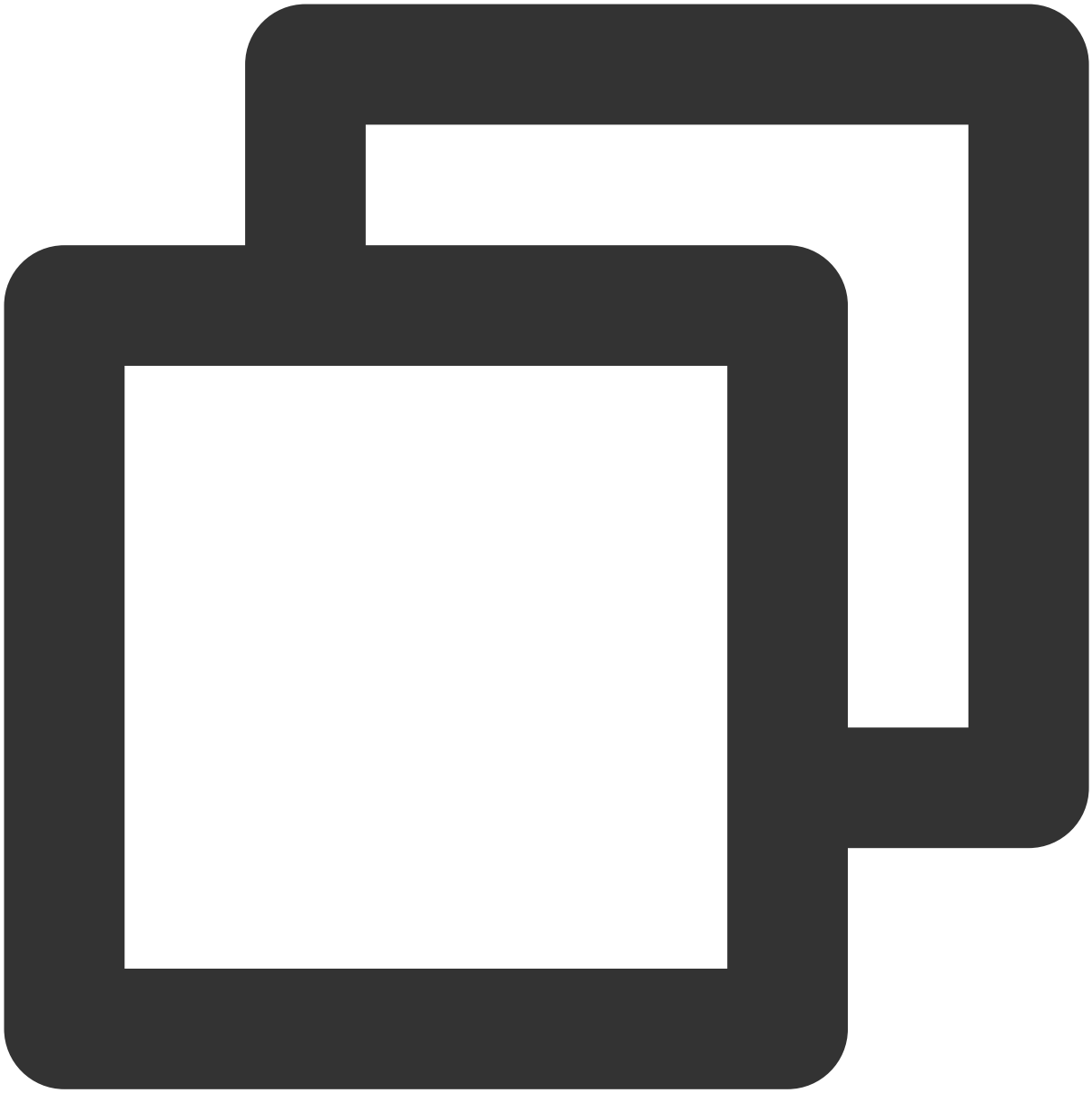


Modularization

In browsers, JavaScript is running in the same scope, and the defined parameters or methods can be accessed or overridden by subsequently loaded scripts. Unlike browsers, mini programs can use any JavaScript file as a module and expose APIs through `module.exports` or `exports` .

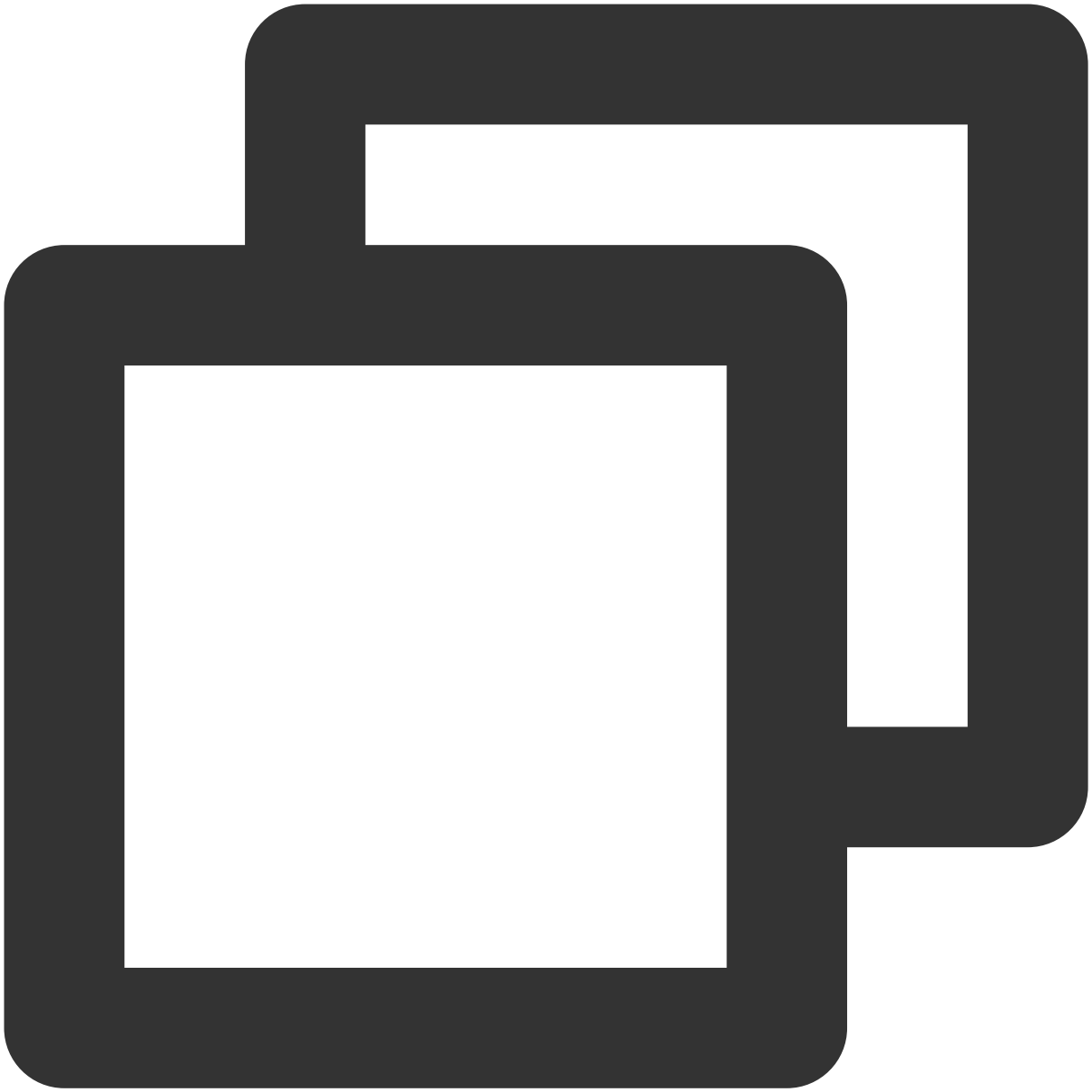
For example, `B.js` references module A and uses the `multiplyBy2` method exposed by A to implement an operation multiplying a variable by 2, as shown below:

©2013-2022 Tencent Cloud. All rights reserved.



```
// moduleA.js
module.exports = function( value ){
  return value * 2;
}
```

©2013-2022 Tencent Cloud. All rights reserved.

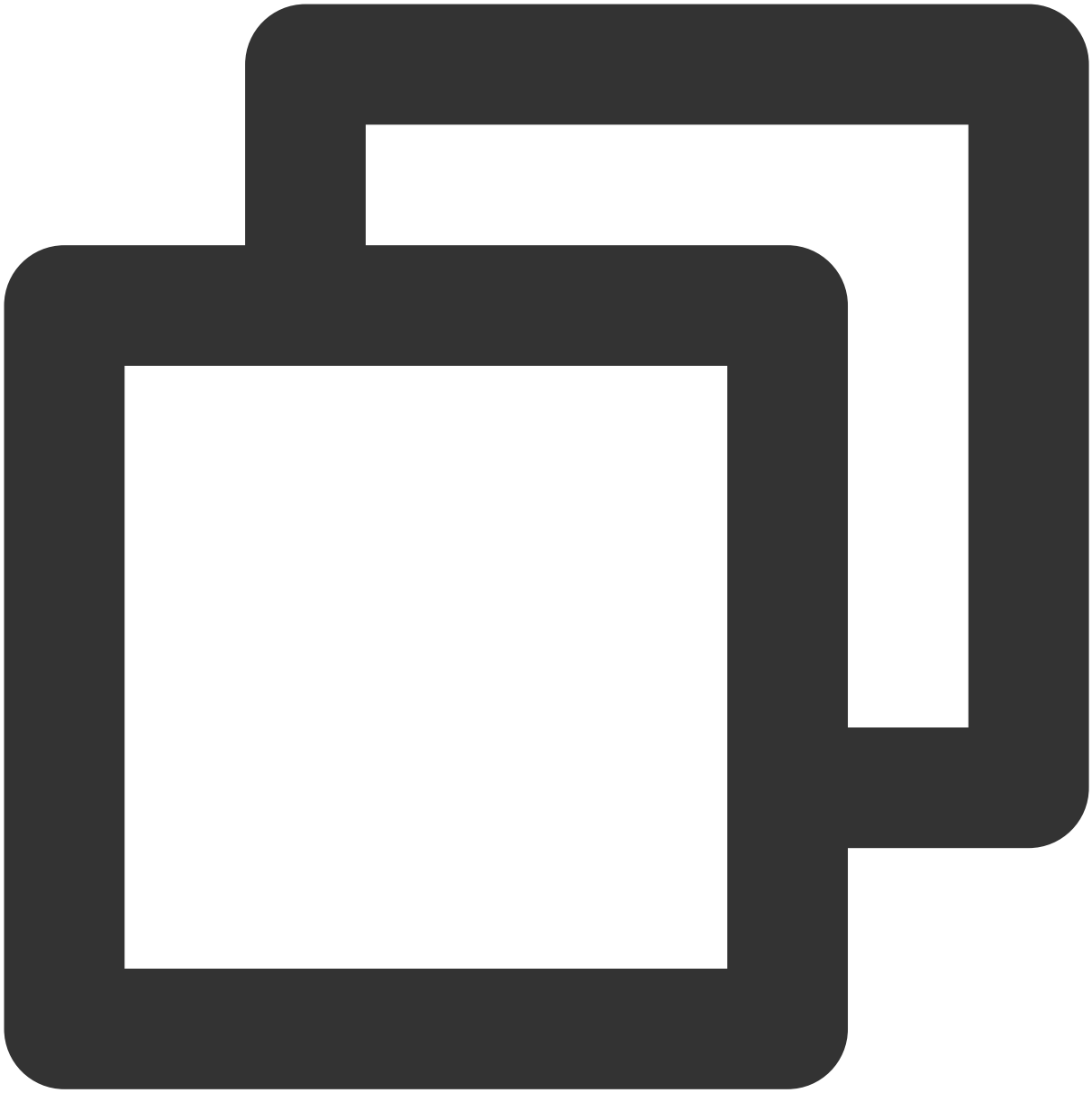


```
// B.js

//Reference module A in B.js
var multiplyBy2 = require('./moduleA')
var result = multiplyBy2(4)
```

Use `require(path)` to import the common code in the files that need to use these modules:

©2013-2022 Tencent Cloud. All rights reserved.

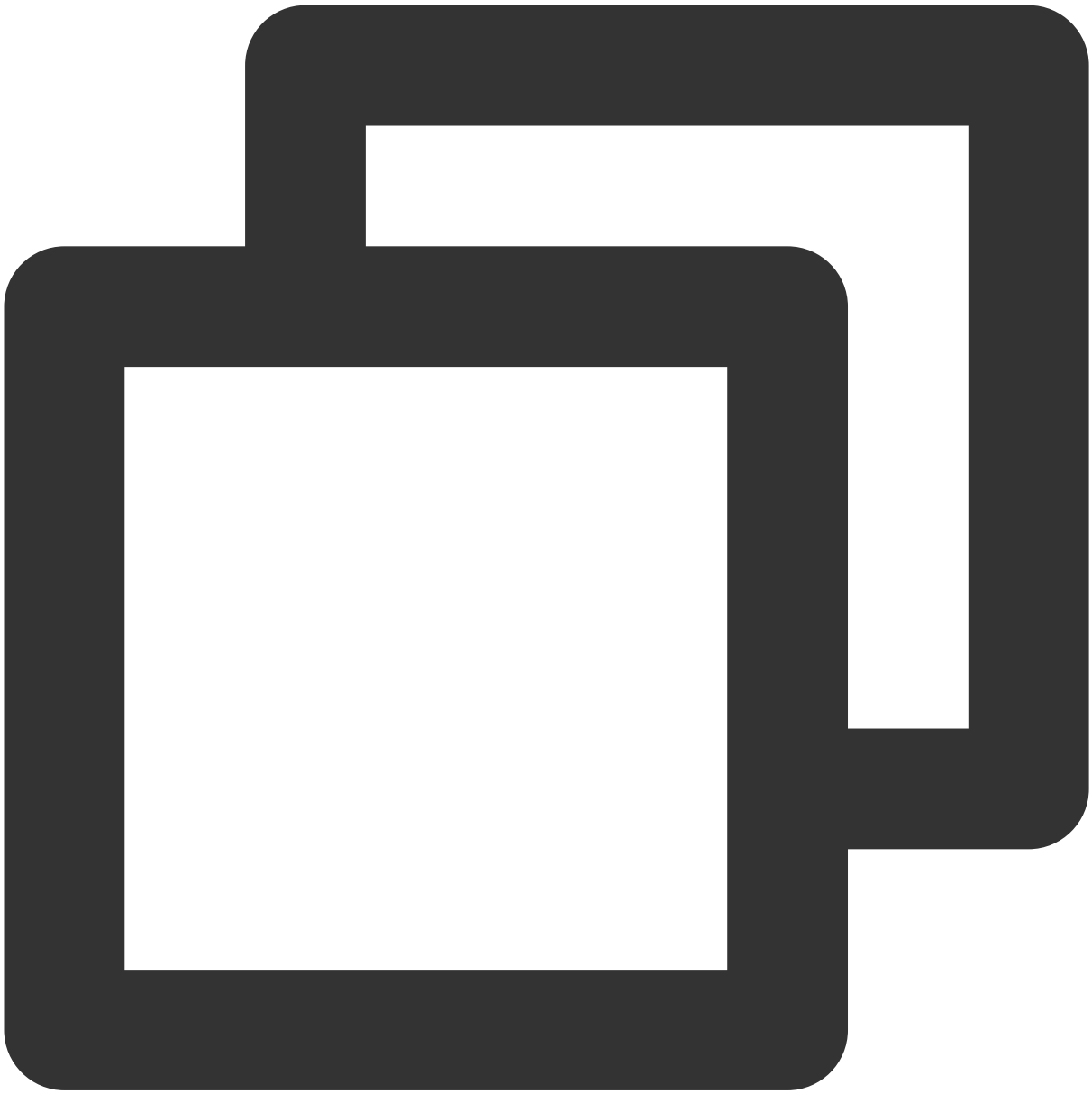


```
var common = require('common.js')
Page({
  helloMINA: function() {
    common.sayHello('MINA')
  },
  goodbyeMINA: function() {
    common.sayGoodbye('MINA')
  }
})
```

©2013-2022 Tencent Cloud. All rights reserved.

Script execution order

In browsers, the scripts are executed in strict accordance with the order of loading, as shown below:



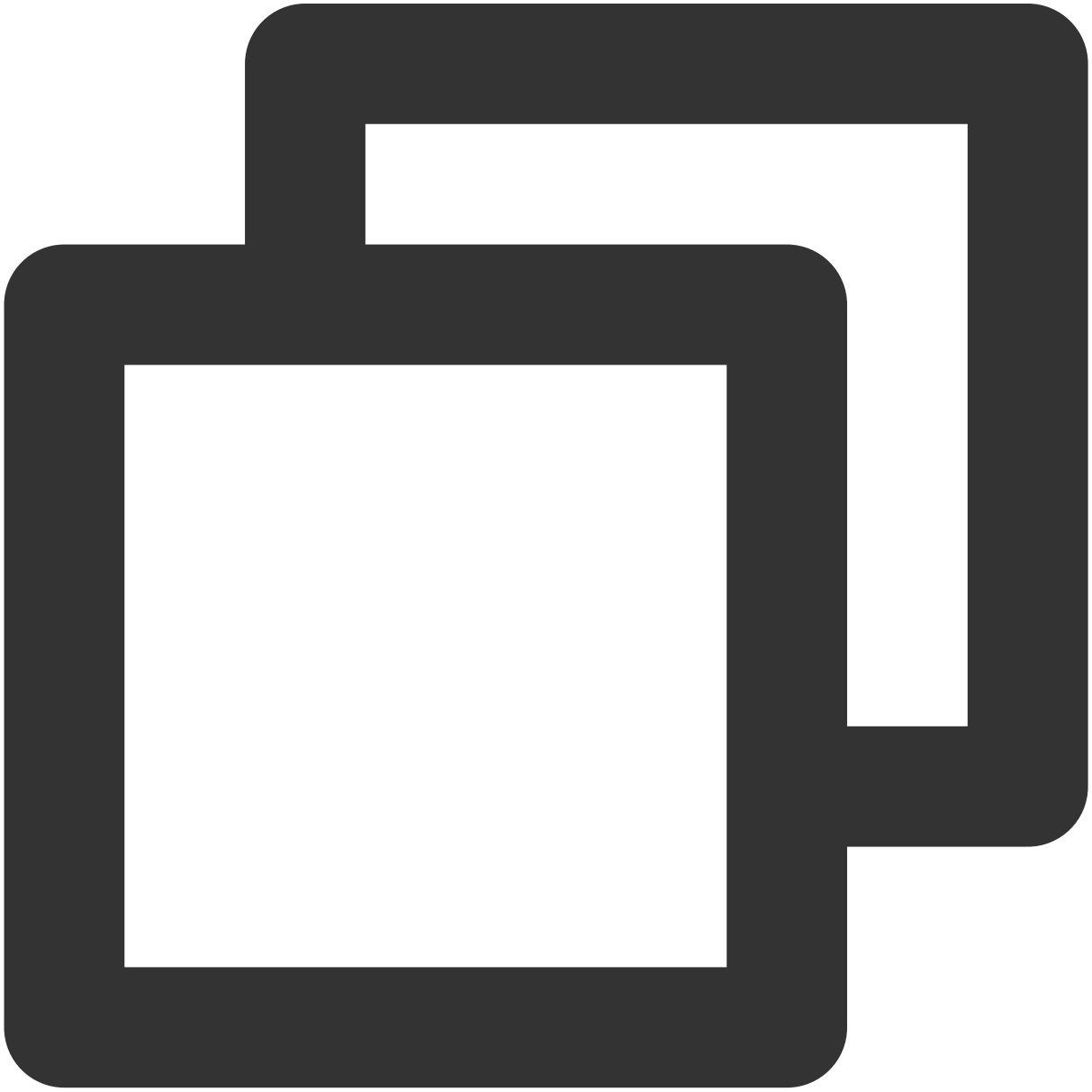
```
<html>
<head>
  <!-- a.js
  console.log('a.js')
   -->
  <script src ="a.js"></script>
  <script>
```

©2013-2022 Tencent Cloud. All rights reserved.

```
    console.log('inline script')
  </script>

  <!-- b.js
  console.log('b.js')
   -->
  <script src ="b.js"></script>
</head>
</html>
```

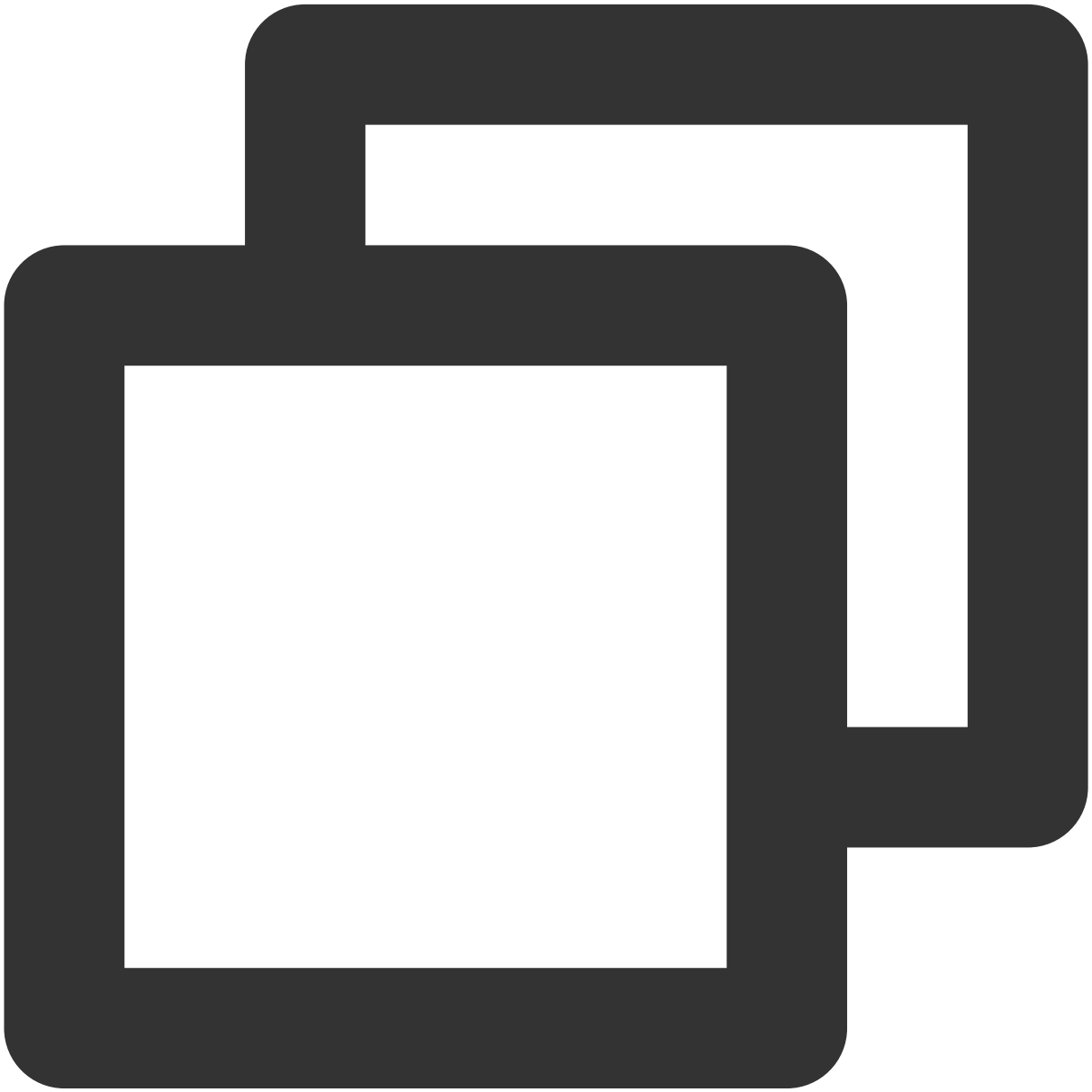
The output of the above code is:



©2013-2022 Tencent Cloud. All rights reserved.

```
a.js
inline script
b.js
```

The script execution order is different in mini programs. The entry point for the mini program execution is `app.js` . And the mini program execution order will follow the order defined in `require` in app.js file. For example:



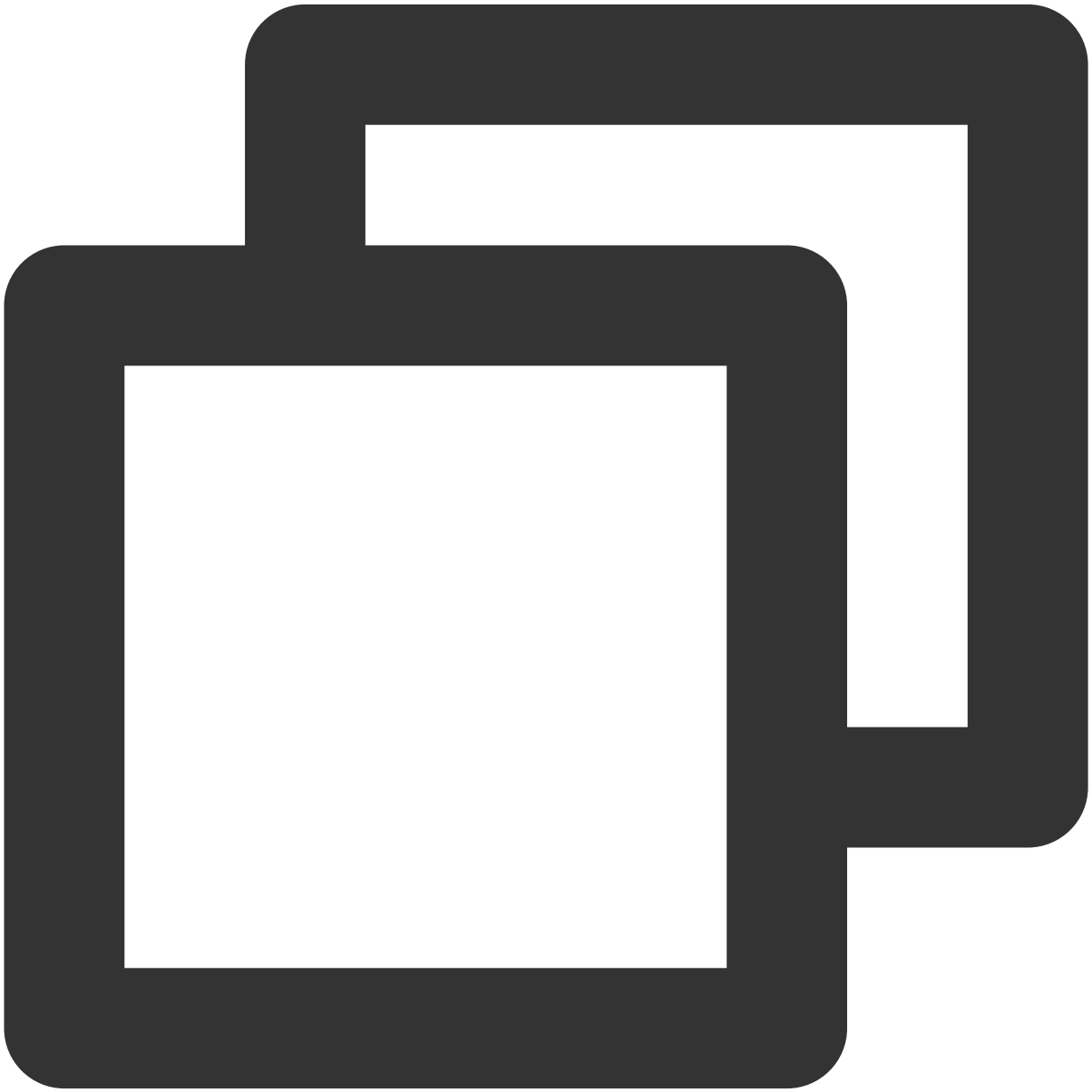
```
/* a.js
console.log('a.js')
*/
```

©2013-2022 Tencent Cloud. All rights reserved.

```
var a = require('./a.js')
console.log('app.js')

/* b.js
console.log('b.js')
*/
var b = require('./b.js')
```

The output order of the above code is:

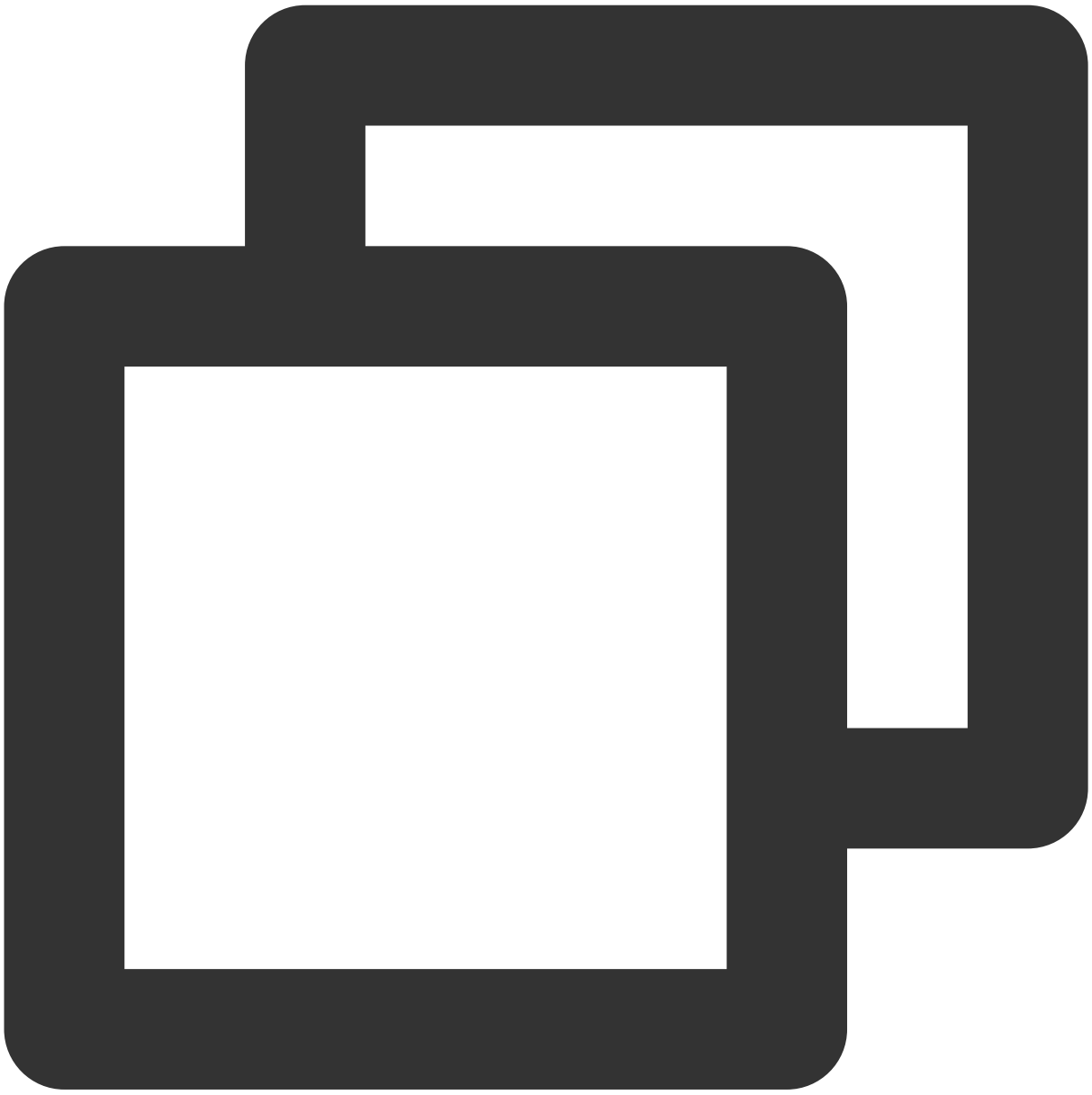


```
a.js
app.js
```

©2013-2022 Tencent Cloud. All rights reserved.

```
b.js
```

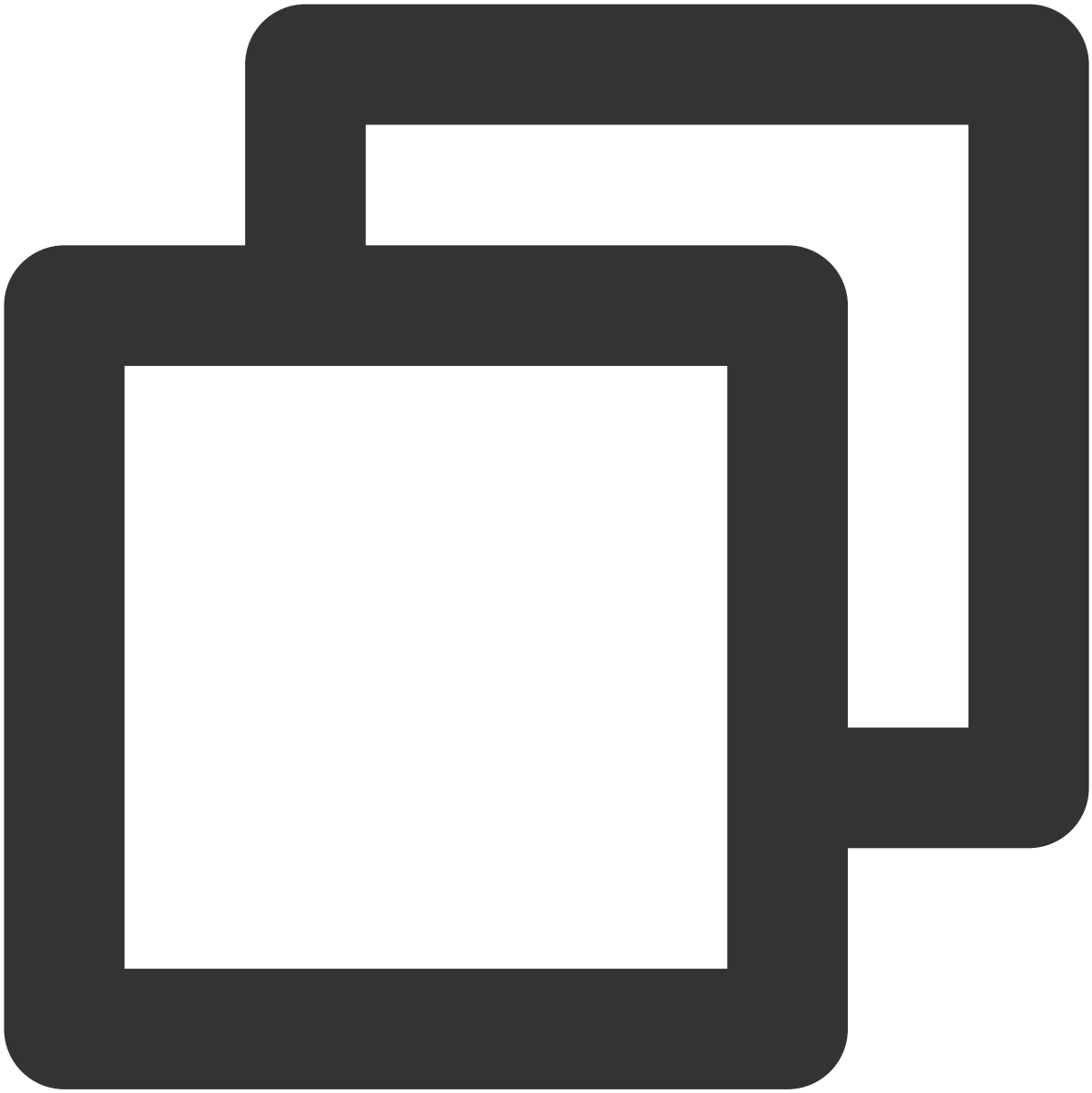
When the `app. js` is executed, the mini program will be executed in accordance with the order of pages defined by developers in the app. json, one by one, as shown below:



```
{
  "pages": [
    "pages/index/index",
    "pages/log/log",
    "pages/result/result"
  ],
```

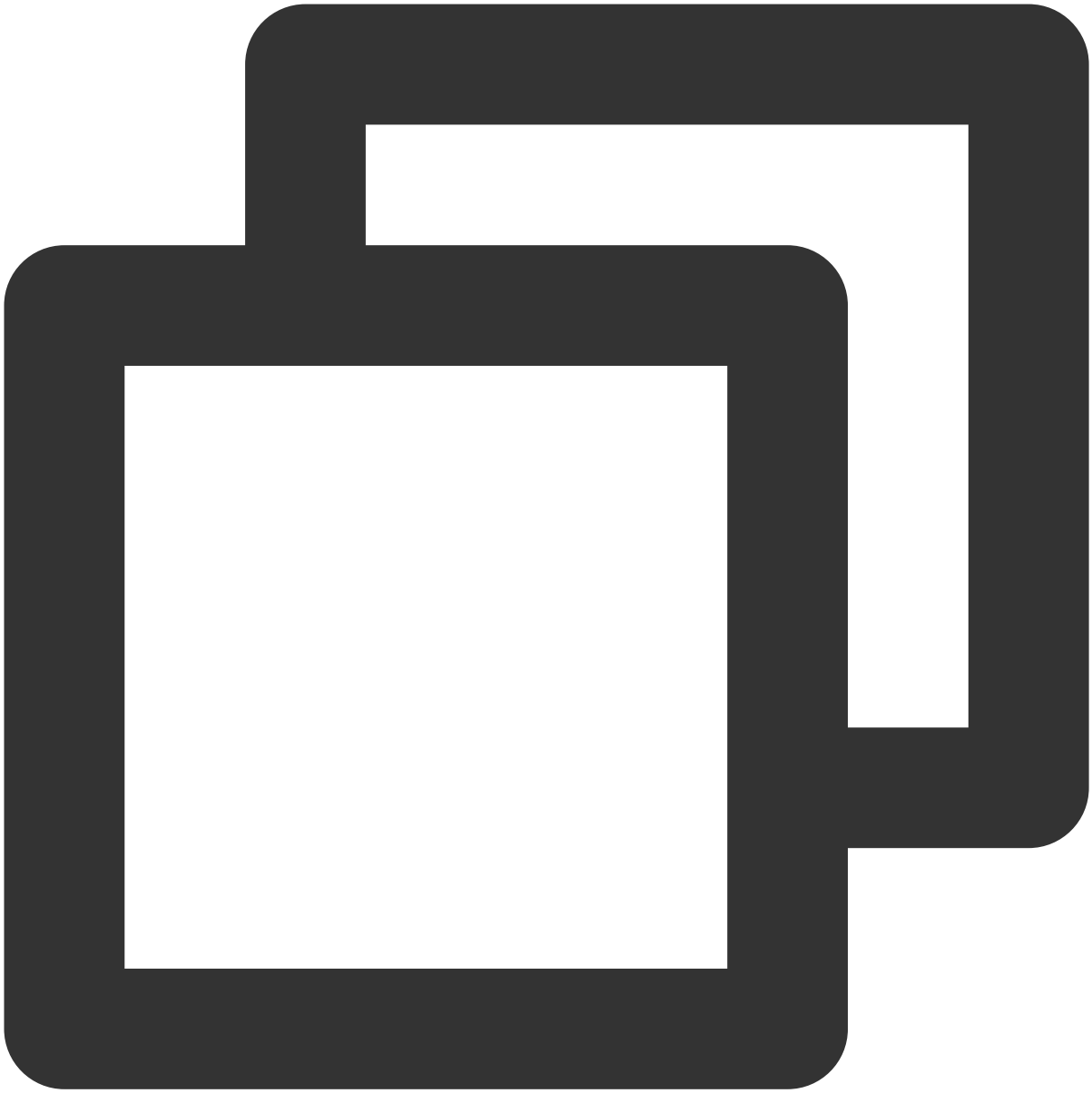
©2013-2022 Tencent Cloud. All rights reserved.

```
  "window": {}
}
```



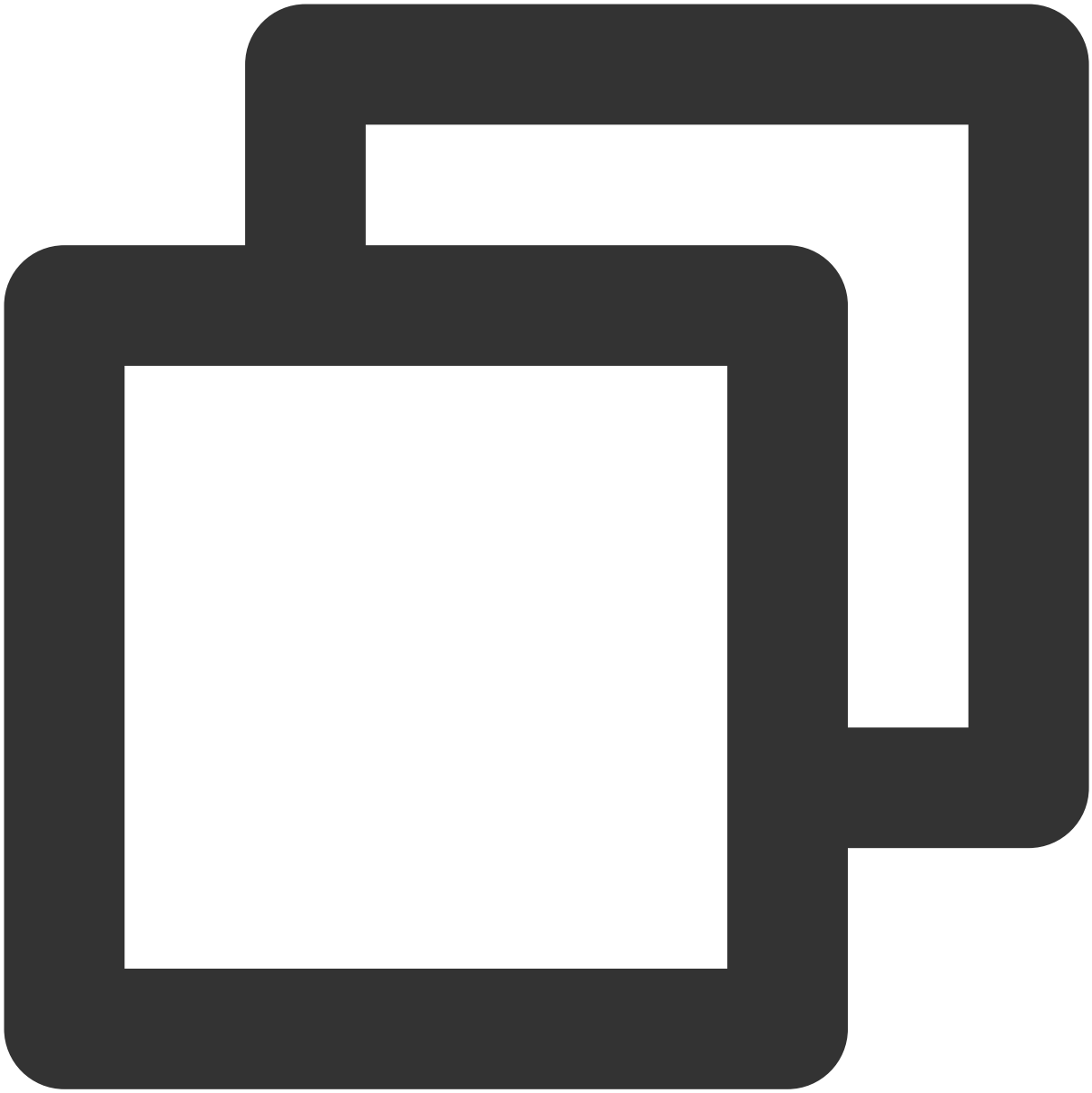
```
// app.js
console.log('app.js')
```

©2013-2022 Tencent Cloud. All rights reserved.



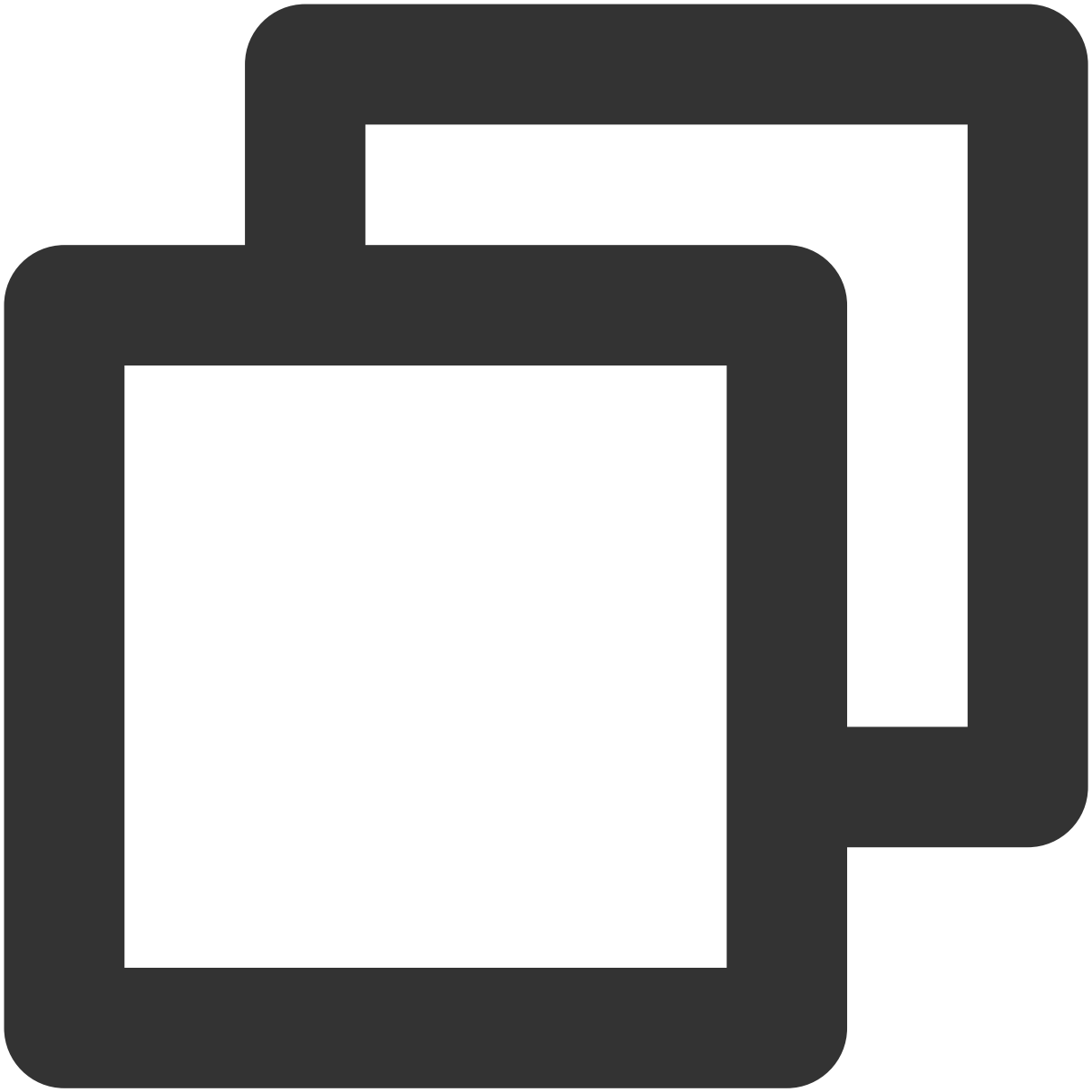
```
// pages/index/index
console.log('pages/index/index')
```

©2013-2022 Tencent Cloud. All rights reserved.



```
// pages/log/log
console.log('pages/log/log')
```

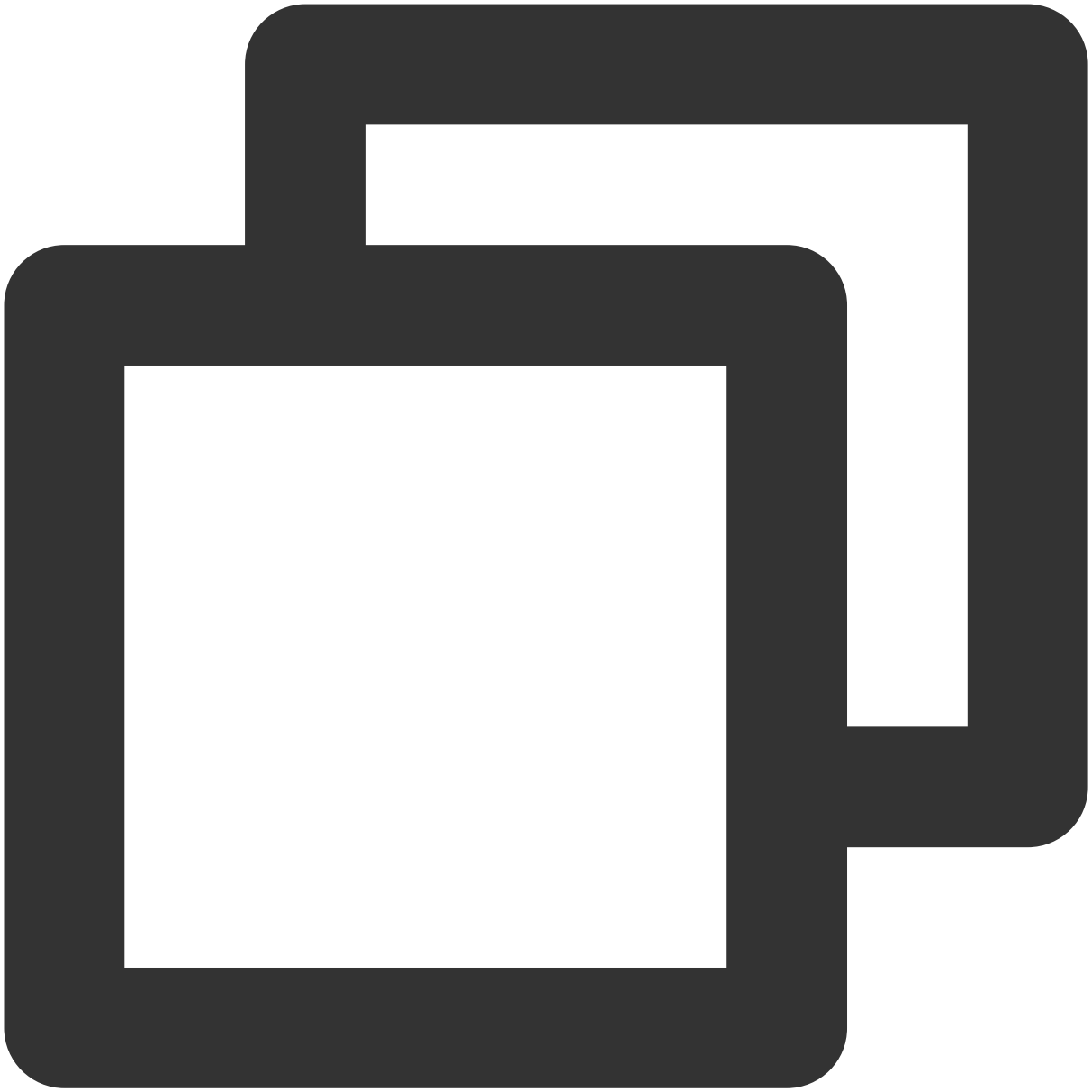
©2013-2022 Tencent Cloud. All rights reserved.



```
// pages/result/result
console.log('pages/result/result')
```

The output of above file execution:

©2013-2022 Tencent Cloud. All rights reserved.



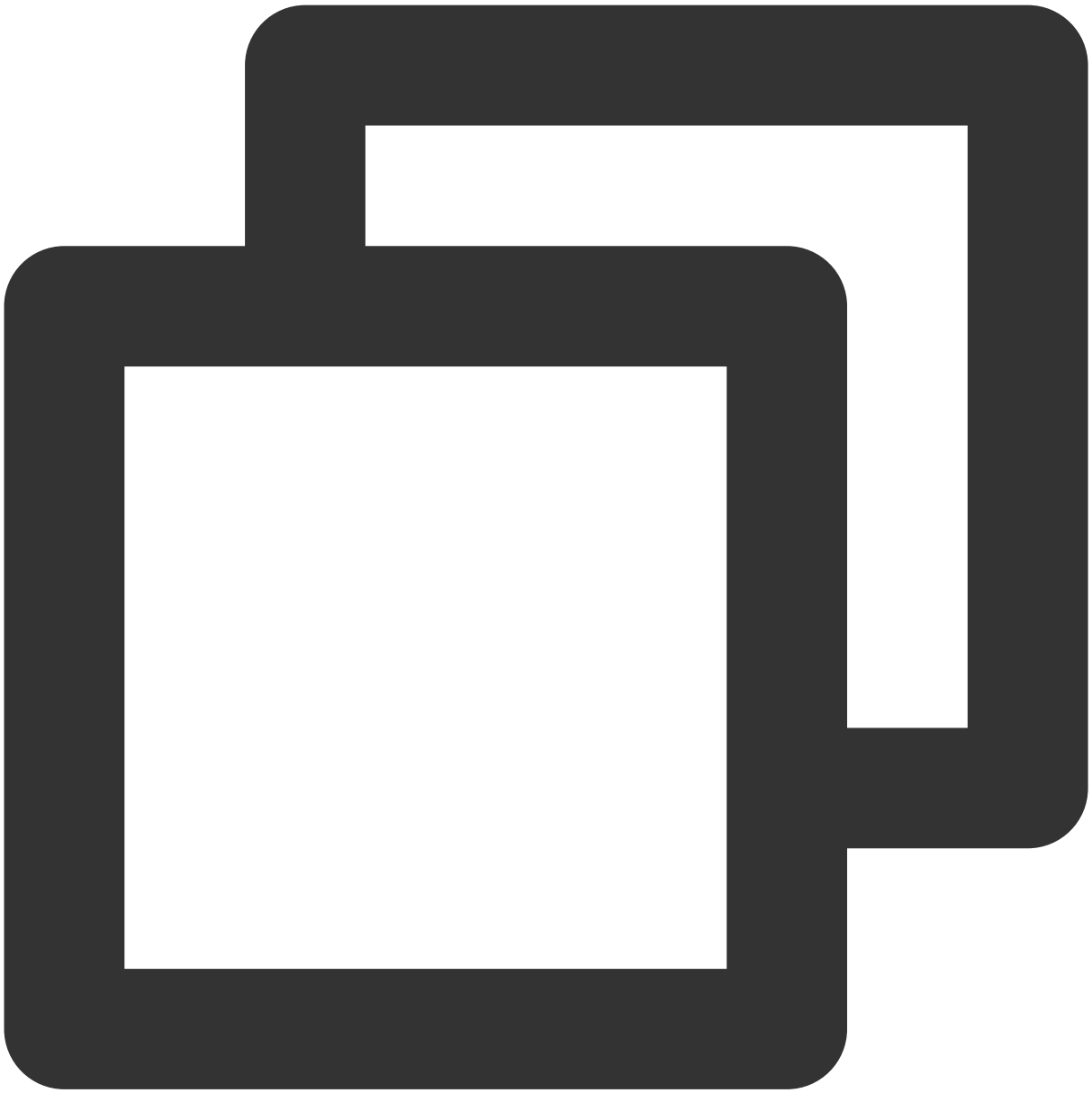
```
app.js
pages/index/index
pages/log/log
pages/result/result
```

Scope

Unlike script files that run in a browser, mini program scripts have a scope that is more similar to NodeJS.

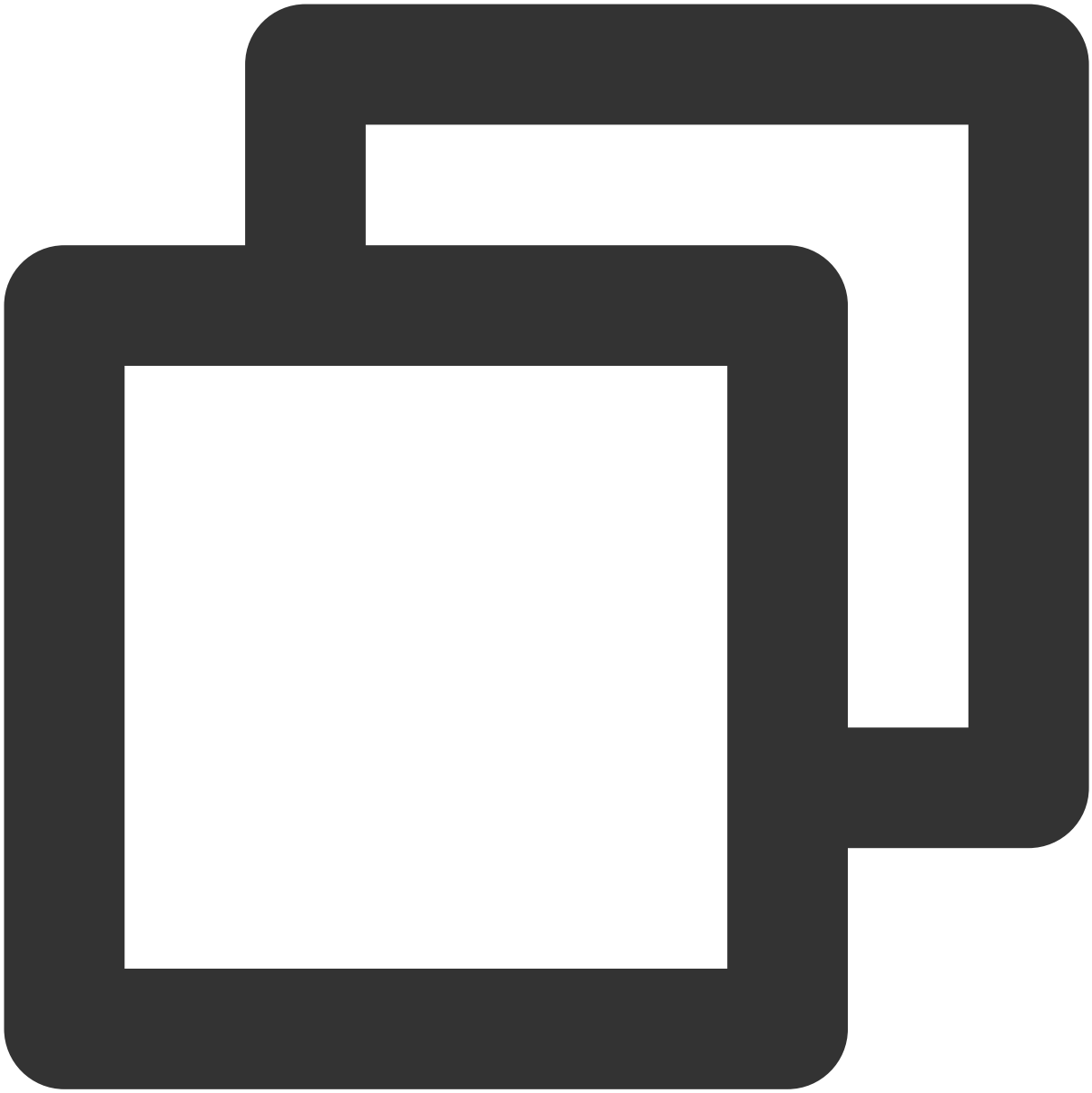
©2013-2022 Tencent Cloud. All rights reserved.

Variables and functions declared in a JavaScript file are only valid in the file. Variables and functions with the same name can be declared in different files without affecting each other.



```
// a.js
//Define local variables
var localValue = 'a'
```

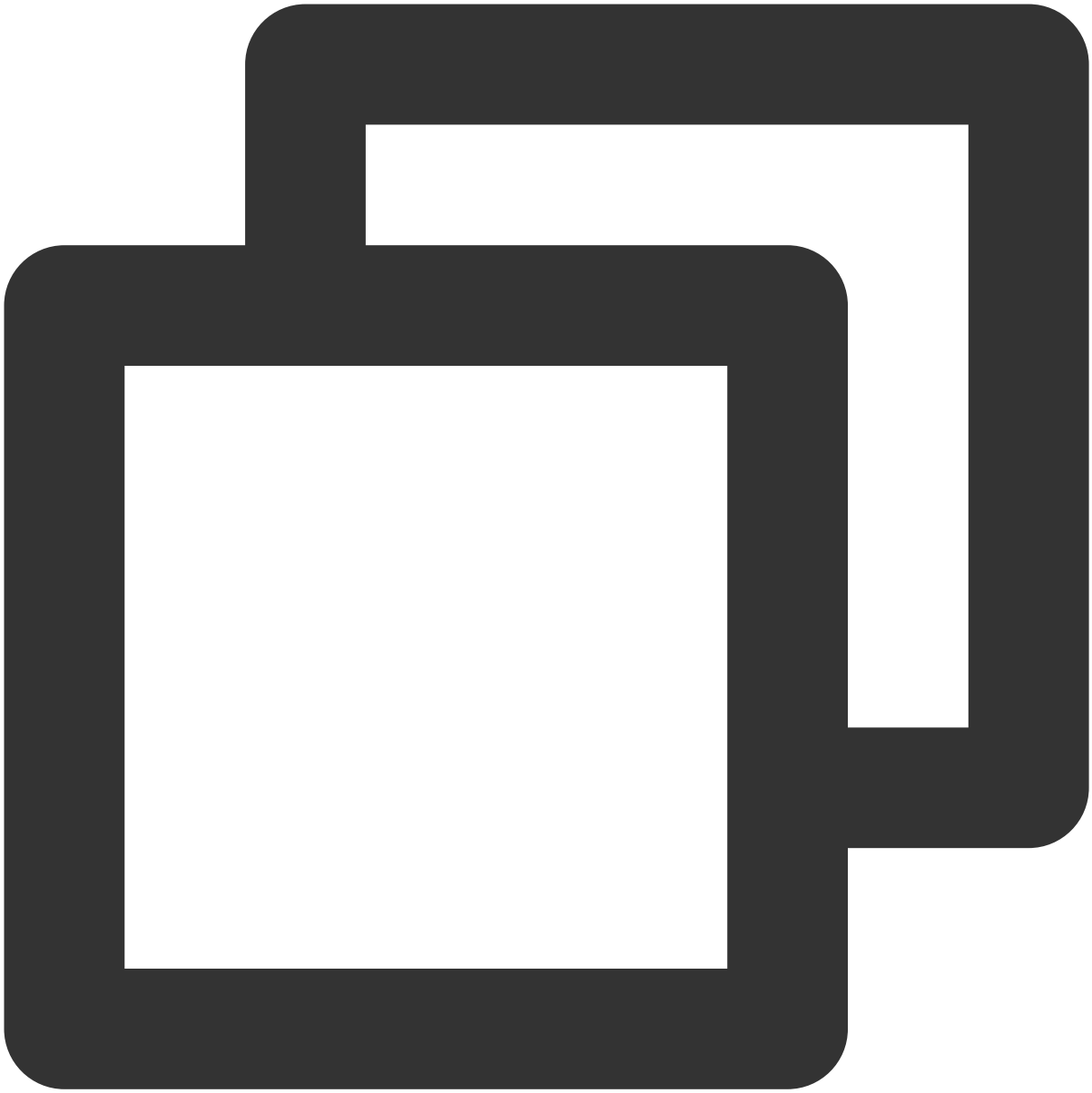
©2013-2022 Tencent Cloud. All rights reserved.



```
// b.js
// b.js cannot access the variables defined in a.js file:
//Define local variable console.log (localValue) //An error triggered: b.js cannot
```

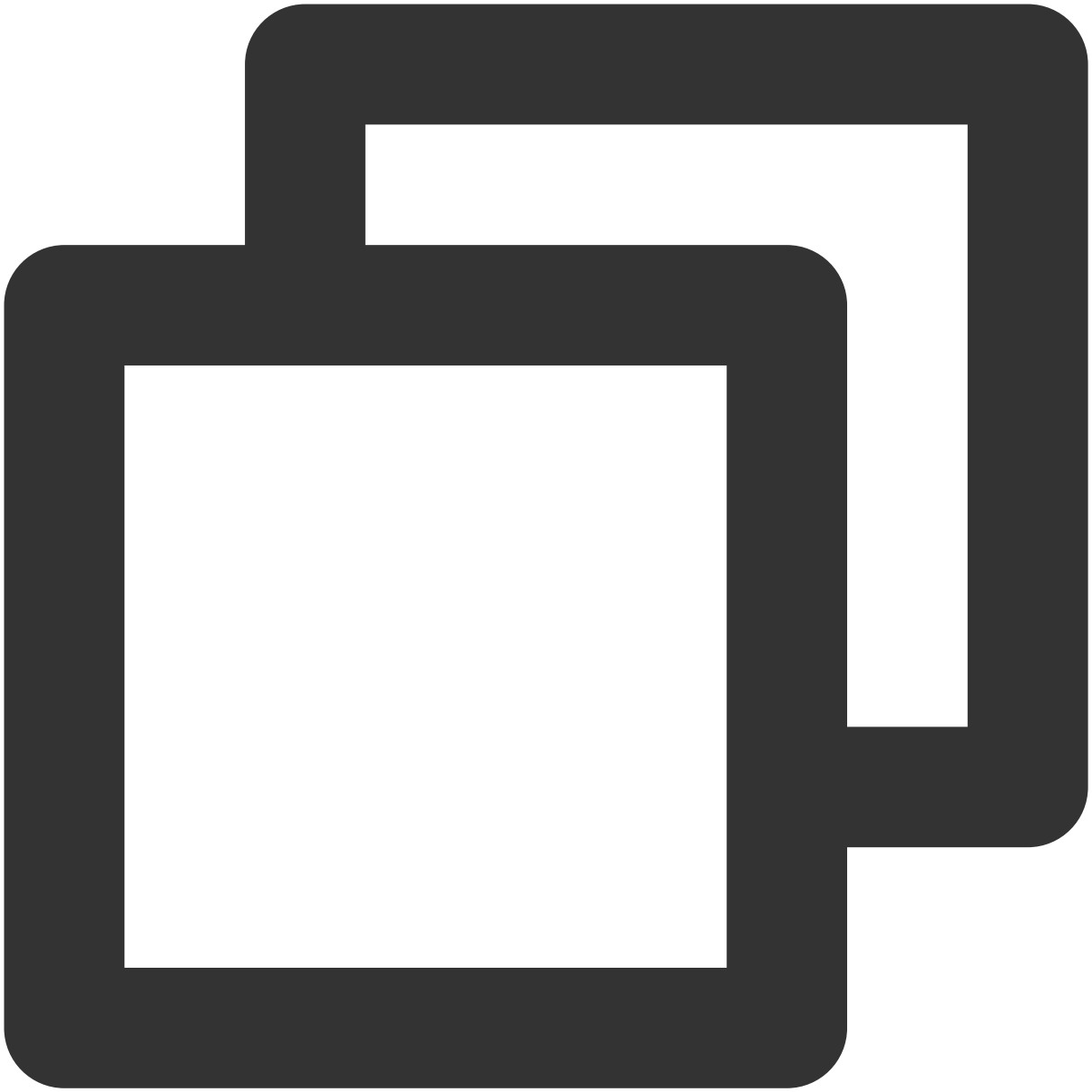
When you need to use a global variable, you can use global function `getApp()` to obtain a global instance and set the related attribute values, setting the global variable:

©2013-2022 Tencent Cloud. All rights reserved.



```
// a.js
//Access global variable
var global = getApp()
global.globalValue = 'globalValue'
```

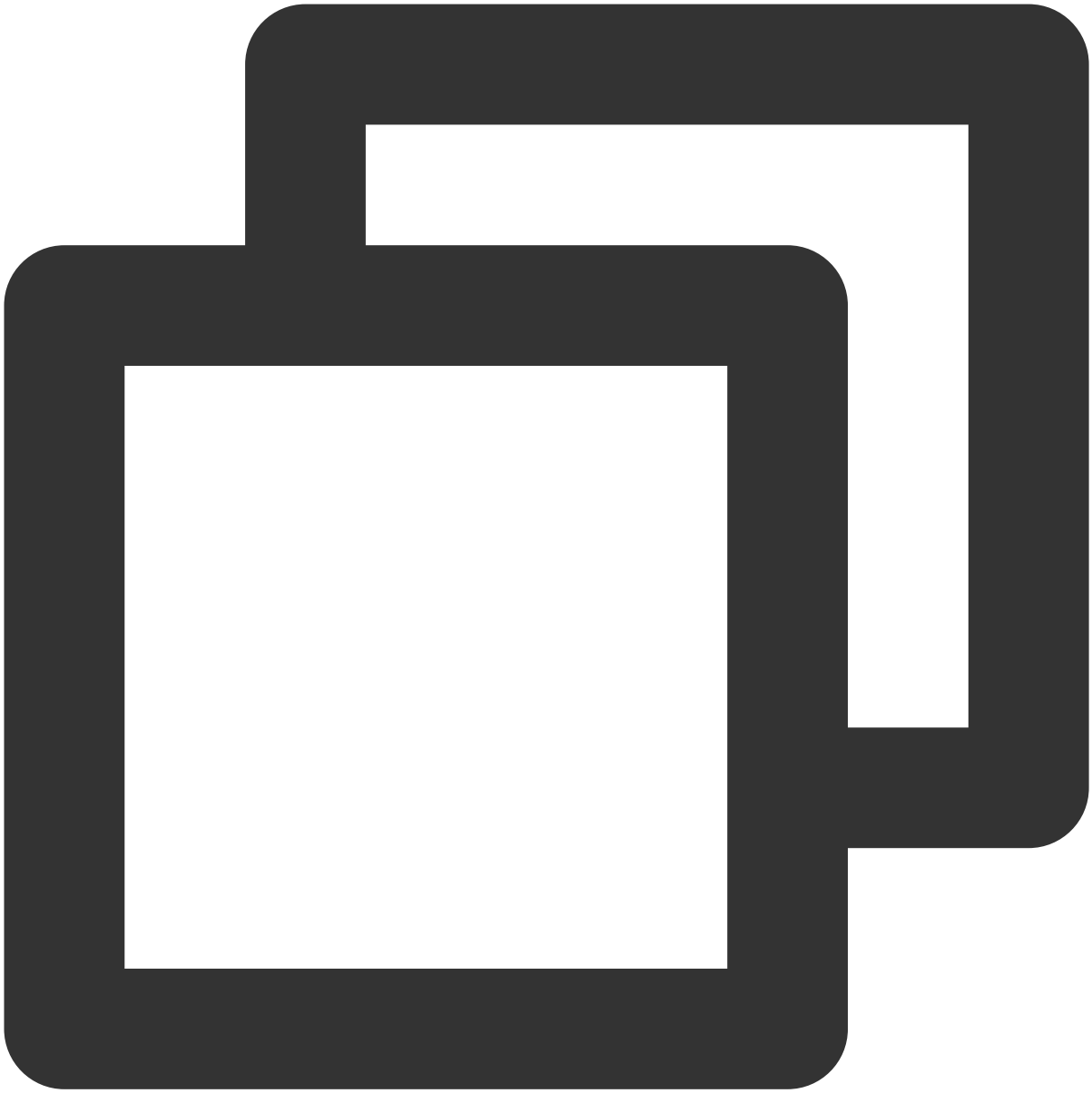
©2013-2022 Tencent Cloud. All rights reserved.



```
// b.js
//Access global variable
var global = getApp()
console.log(global.globalValue)
// Output globalValue
```

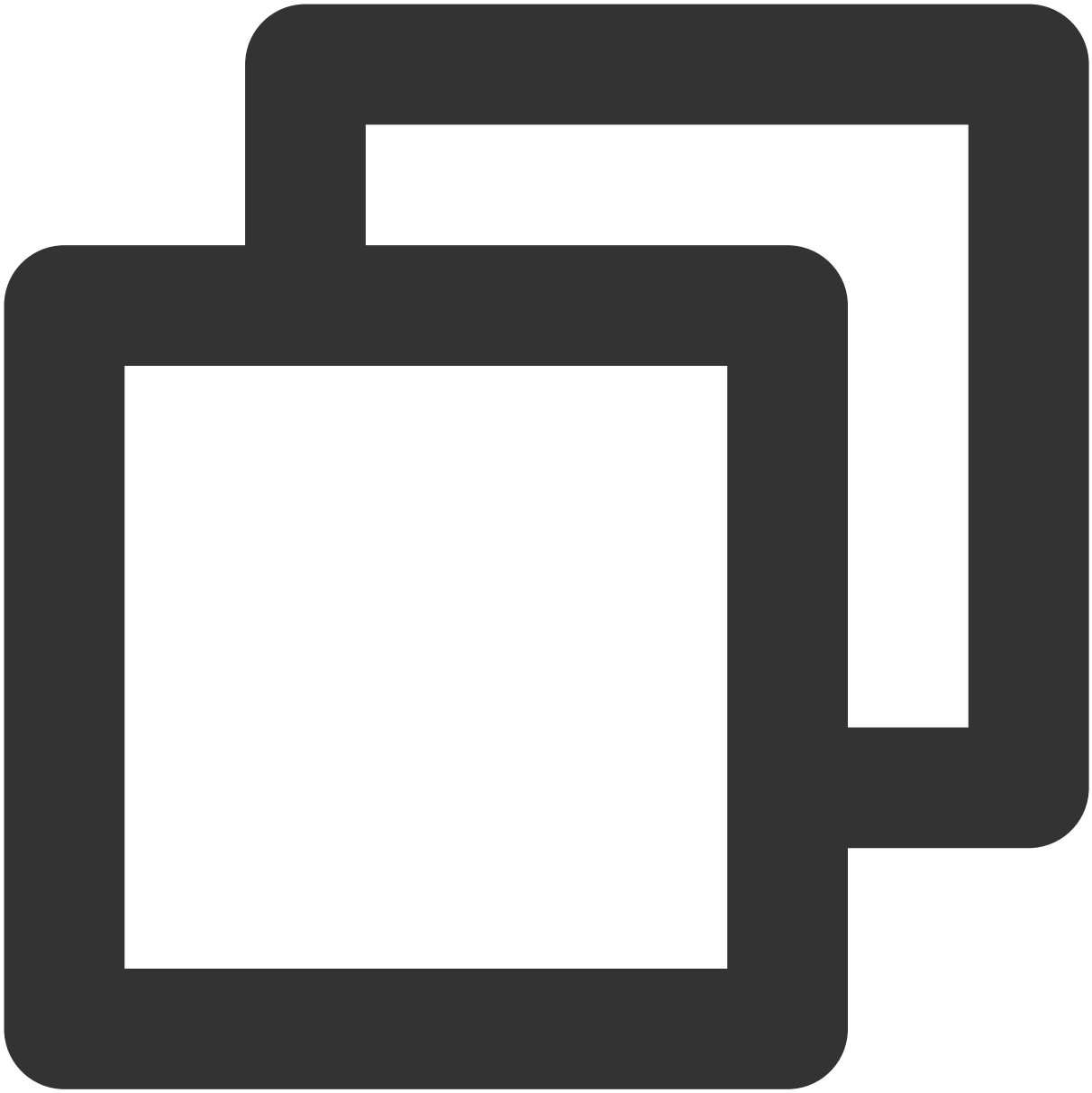
Please note that the example only works when the a.js is executed first and b.js is executed later. When you need to ensure that the global data can be used in any file securely, you can set in the `App()` :

©2013-2022 Tencent Cloud. All rights reserved.



```
// app.js
App({
  globalData: 1
})
```

©2013-2022 Tencent Cloud. All rights reserved.

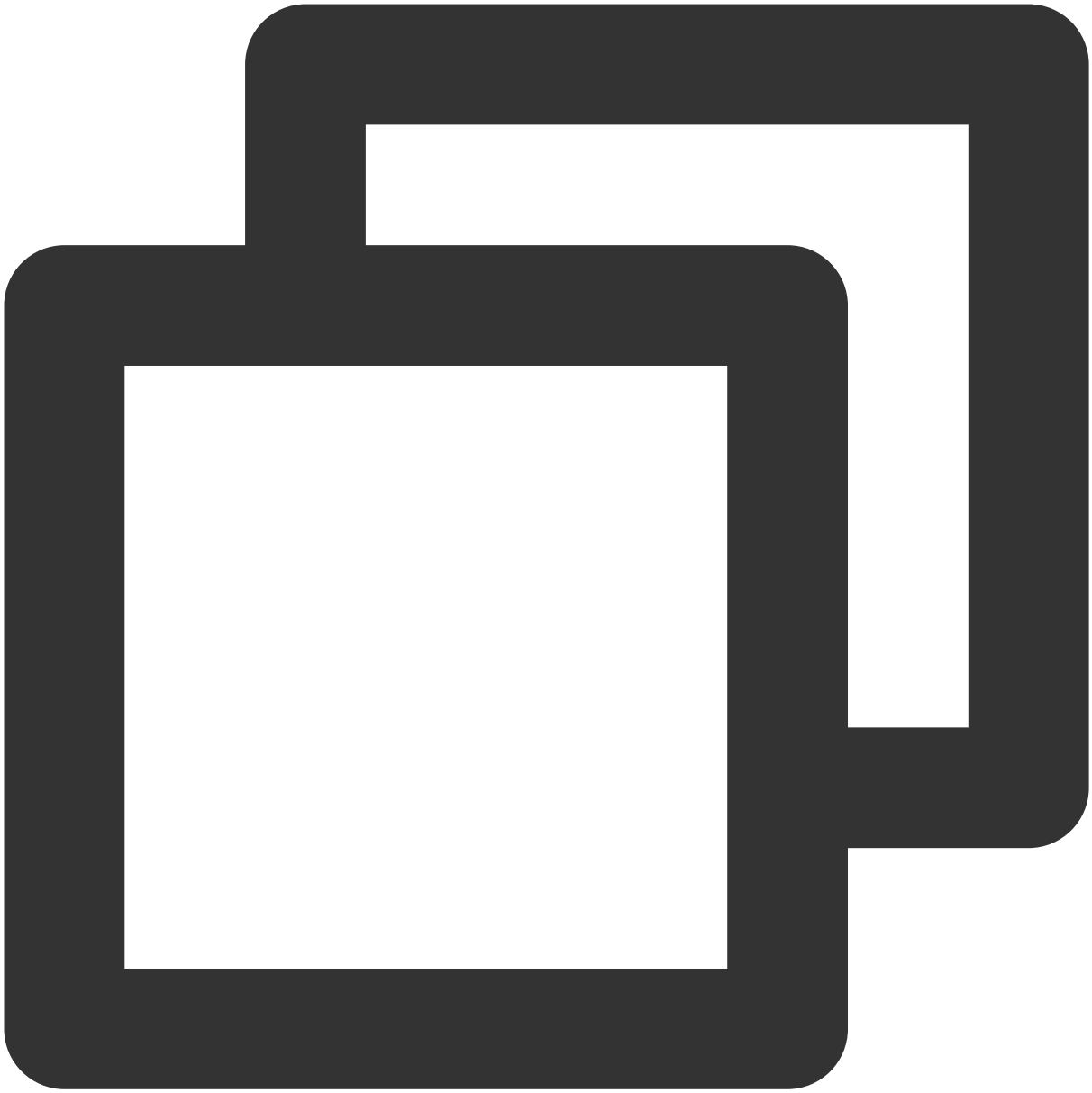


```
// a.js
//Local variable
var localValue = 'a'

//Get global variable
var app = getApp()

//Modify global variable
app.globalData++  // The value of globalData is 2 after execution
```

©2013-2022 Tencent Cloud. All rights reserved.



```
// b.js
//Define other local variables, which will not affect the variables in a.js
var localValue = 'b'

// If a.js is executed first, the value outputted should be 2
console.log(getApp().globalData)
```

©2013-2022 Tencent Cloud. All rights reserved.

Guidance
Learn about mini programs
Mini Program Host Environment

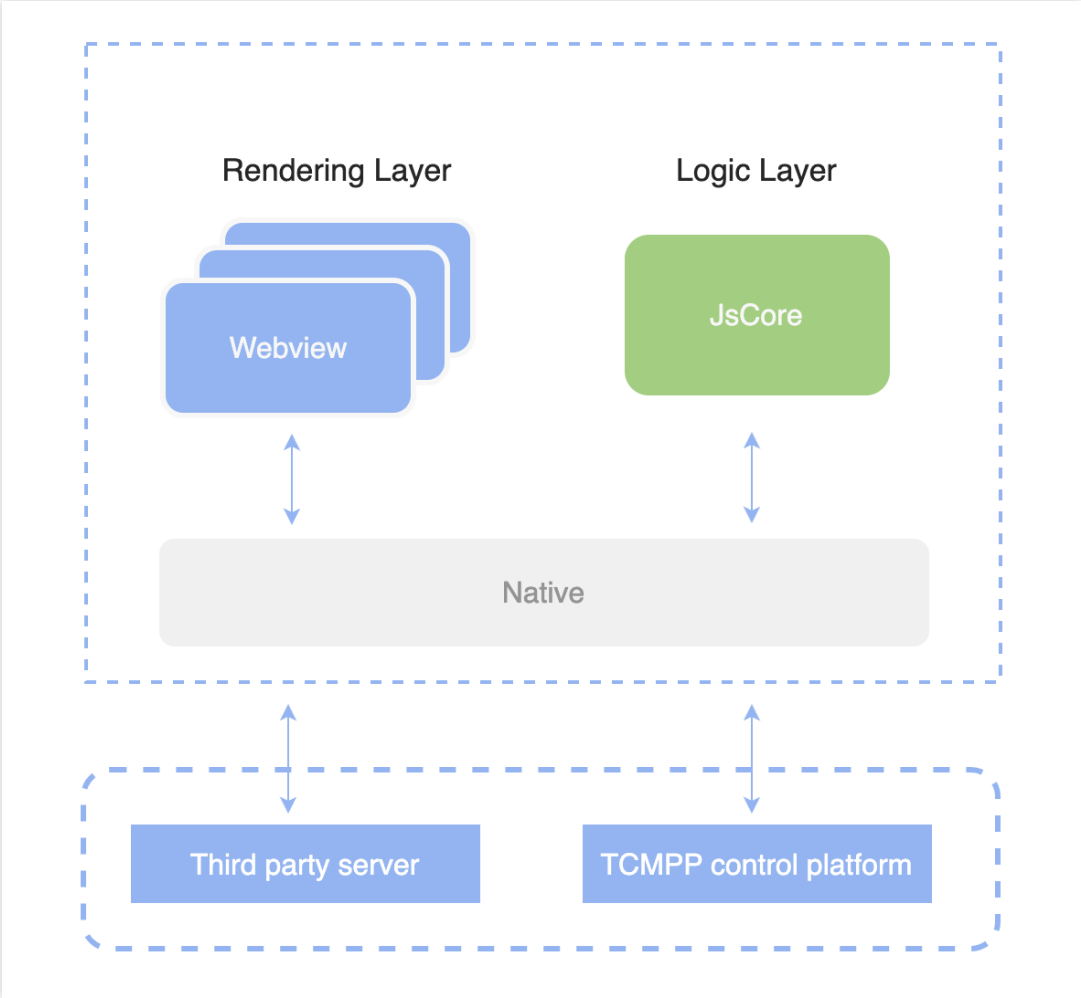
Last updated：2024-07-12 18:00:18

TCMPP mini programs can run in all types of mobile applications. When loading and launching a mini program, the required running environment and capabilities need to be provided to the mini program by the client application. We call this set of environment " `host environment (host application)` ". With the capabilities provided by the host environment, mini programs can complete many functions that cannot be achieved by ordinary web pages.

1. Rendering Layer and Logic Layer

Firstly, let's have a brief understanding of the runtime environment of the Mini Program, which is divided into a `render layer` and a `logic layer` . WXML templates and WXSS styles are used in the render layer and JS scripts in the logic layer.
These two layers are managed via two threads: the render layer renders its interfaces using WebView threads, while the logic layer runs the JS scripts with JsCore threads. As more than one interface exists in the Mini Program, there are multiple WebView threads in the render layer. The communication between the WebView threads and JsCore threads as well as the network requests sent from the logic layer are transferred via the Weixin app (hereinafter also referred to as Native). The communication model of the Mini Program is shown below.

©2013-2022 Tencent Cloud. All rights reserved.



The TCMPP mini program runs on multiple platforms: iOS/iPadOS client, Android client, and TCMPP developer tools for debugging.

The script execution environment and the environment used for component rendering are different for different running environments, and the performance varies:

On iOS, iPadOS, the JavaScript code of the logic layer of the mini program runs in JavaScriptCore, and the view layer is rendered by WKWebView, with environments such as iOS 14, iPad OS 14, and so on.

On Android, the JavaScript code of the logic layer of mini programs runs in V8, and the view layer is rendered by the self-developed XWeb engine based on Mobile Chromium kernel.

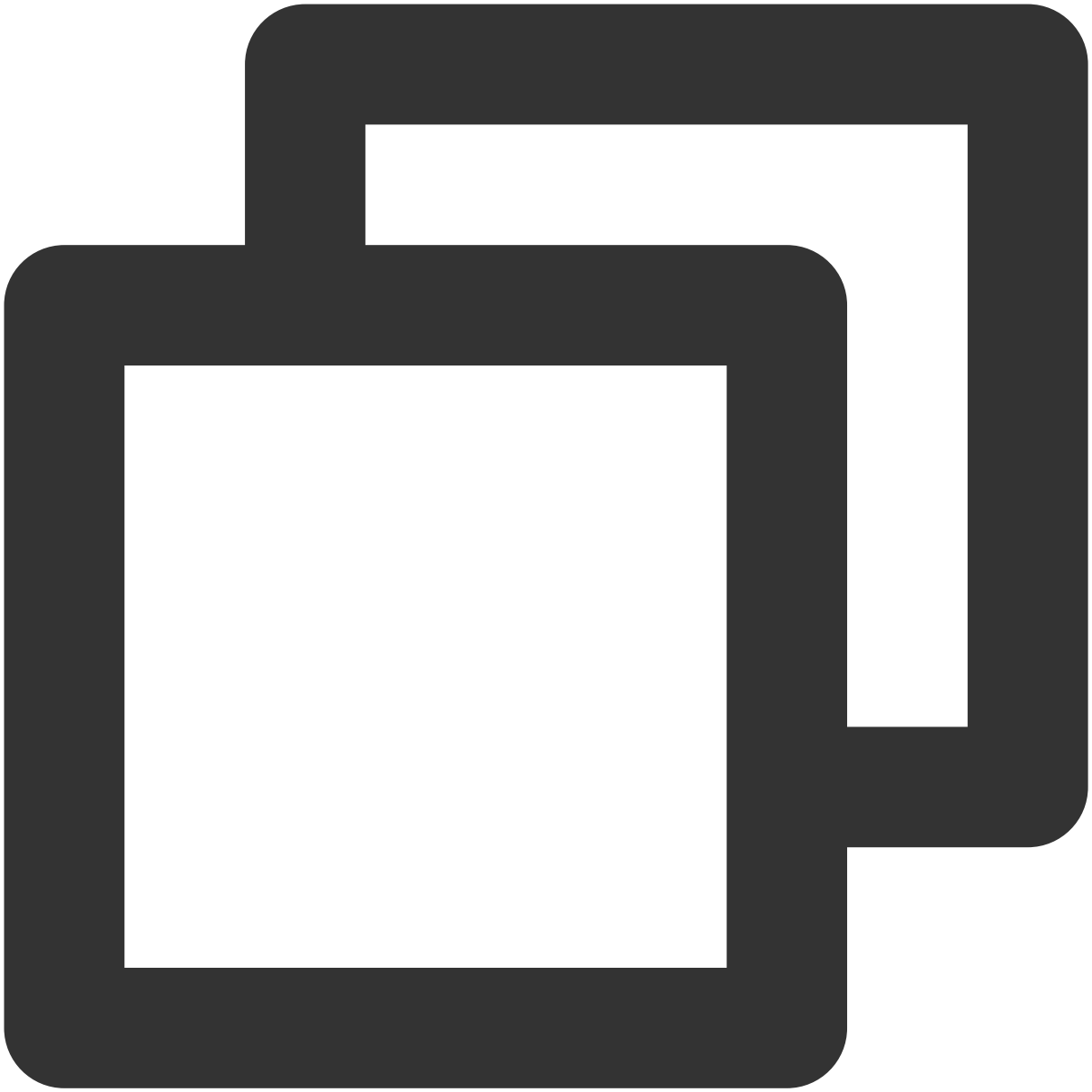
On development tools, the JavaScript code of the logic layer of the mini program runs in NW.js, and the view layer is rendered by Chromium Webview. JavaScriptCore cannot enable Just-In-Time Compiler, and the performance is significantly lower than other platforms under the same conditions.

2. Programs and Pages

The host app downloads the code package of the Mini Program to the local device before opening it.

This allows you to get all the page paths of the Mini Program from the pages field of app.json:

©2013-2022 Tencent Cloud. All rights reserved.

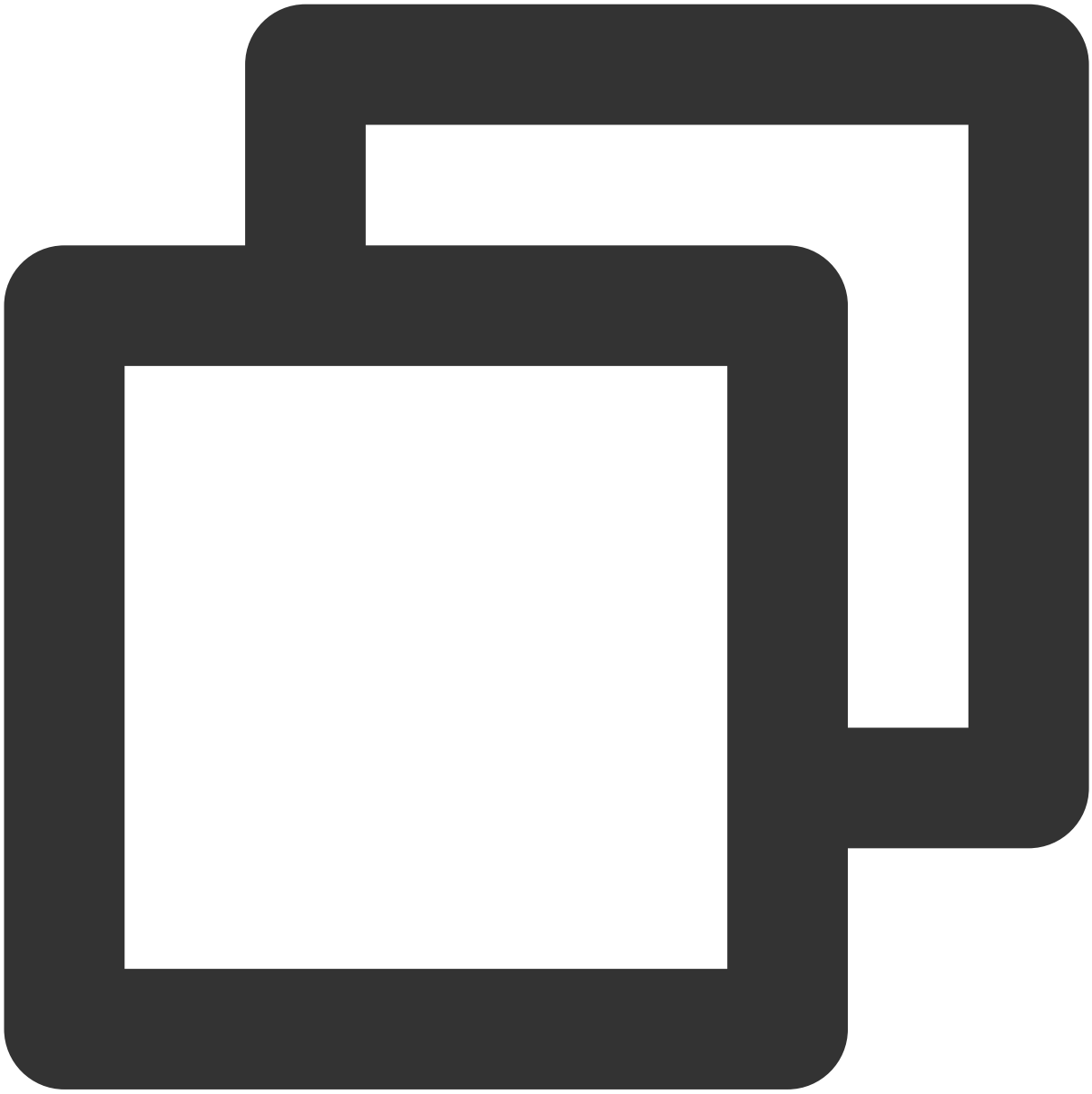


```
{
  "pages":[
     "pages/index/index",
     "pages/logs/logs"
   ]
}
```

This configuration note defines two pages in the project, pages/index/index and pages/logs/logs, and the first page written in the pages field is the home page of the mini program (the first page you see when you open the mini program).

©2013-2022 Tencent Cloud. All rights reserved.

The path to each page can be found in the package. For example, pages/index/index, under this path, there are `WXML` , `WXSS` , `JS` files and `JSON` configuration files corresponding to this page.
After the mini program is launched, the onLaunch callback of the App instance defined in app.js is executed.



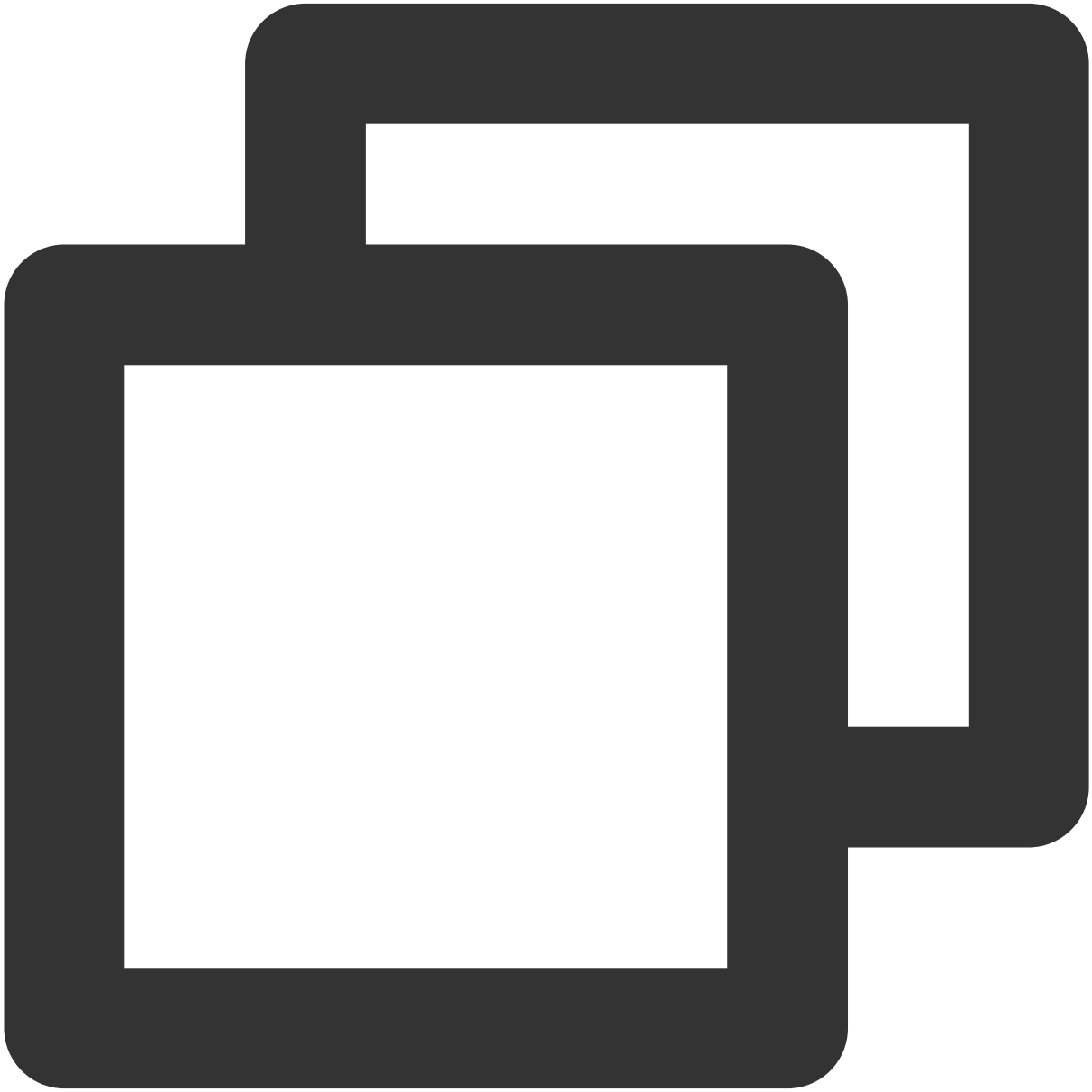
```
App({
  onLaunch() {
   // Triggered after the mini program is launched.
  }
})
```

©2013-2022 Tencent Cloud. All rights reserved.

The entire mini program has only one App instance, which is shared by all the pages, and more event callbacks refer to Registering a Mini Program Account.

Next, let's take a brief look at how a page of a mini program is written.

You can observe that under `pages/logs/logs`, there are actually 4 kinds of files. The host client will first generate an interface according to the `logs.json` configuration, and you can define the top colour and text in this json file. Immediately after that, the client loads the WXML structure and WXSS styles for the page. Finally, the client loads `logs.js`, and you can see that `logs.js` looks like this.



```
Page({
  data: { // Data involved in page rendering
```

©2013-2022 Tencent Cloud. All rights reserved.

```
    logs: []
  },
  onLoad() {
    // After the page is rendered, execute the
  }
})
```

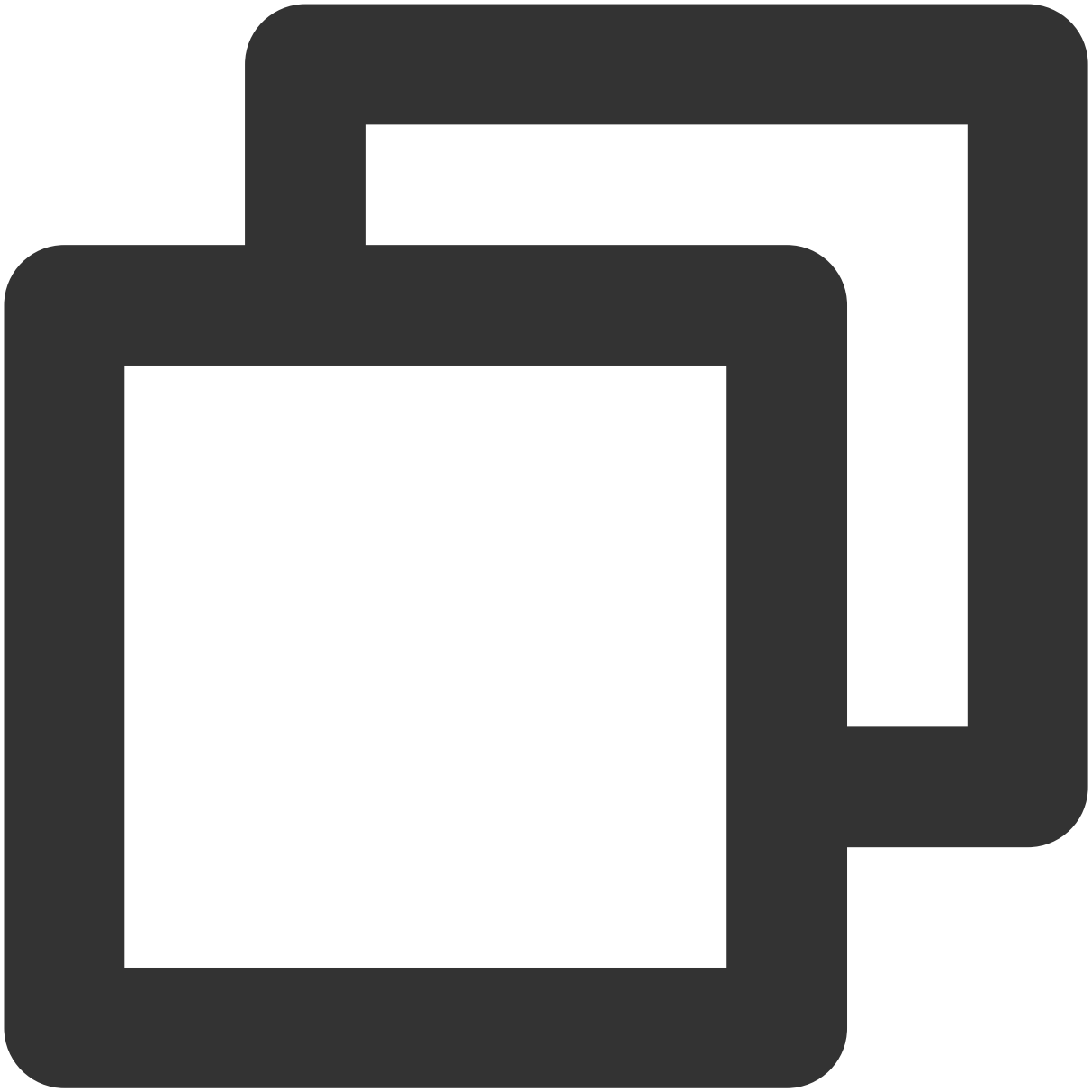
Page is a page constructor that generates a page. When the page is generated, the Mini Program framework renders the final structure using `data` and `index.wxml`, so that the Mini Program appears as it is now.
After the interface is rendered, the page instance receives an onLoad callback, where you can process your logic.

> For more information on the Page constructor, see Registering Page.

3. Components

Components are provided by mini programs for developers to create page UI and customise `WXML`. Just like HTML tags such as div, p, etc., in mini programs, you only need to write the name of the corresponding component tag in WXML to display the component on the interface. For example.

©2013-2022 Tencent Cloud. All rights reserved.



```
<view>Welcome to TCMPP</view>
```

The page displays a line of text: Welcome to TCMPP. The developer can control the style of the component by using style and class to specify the width, height, colours, and so on.

> More components can be found in the Component Overview.

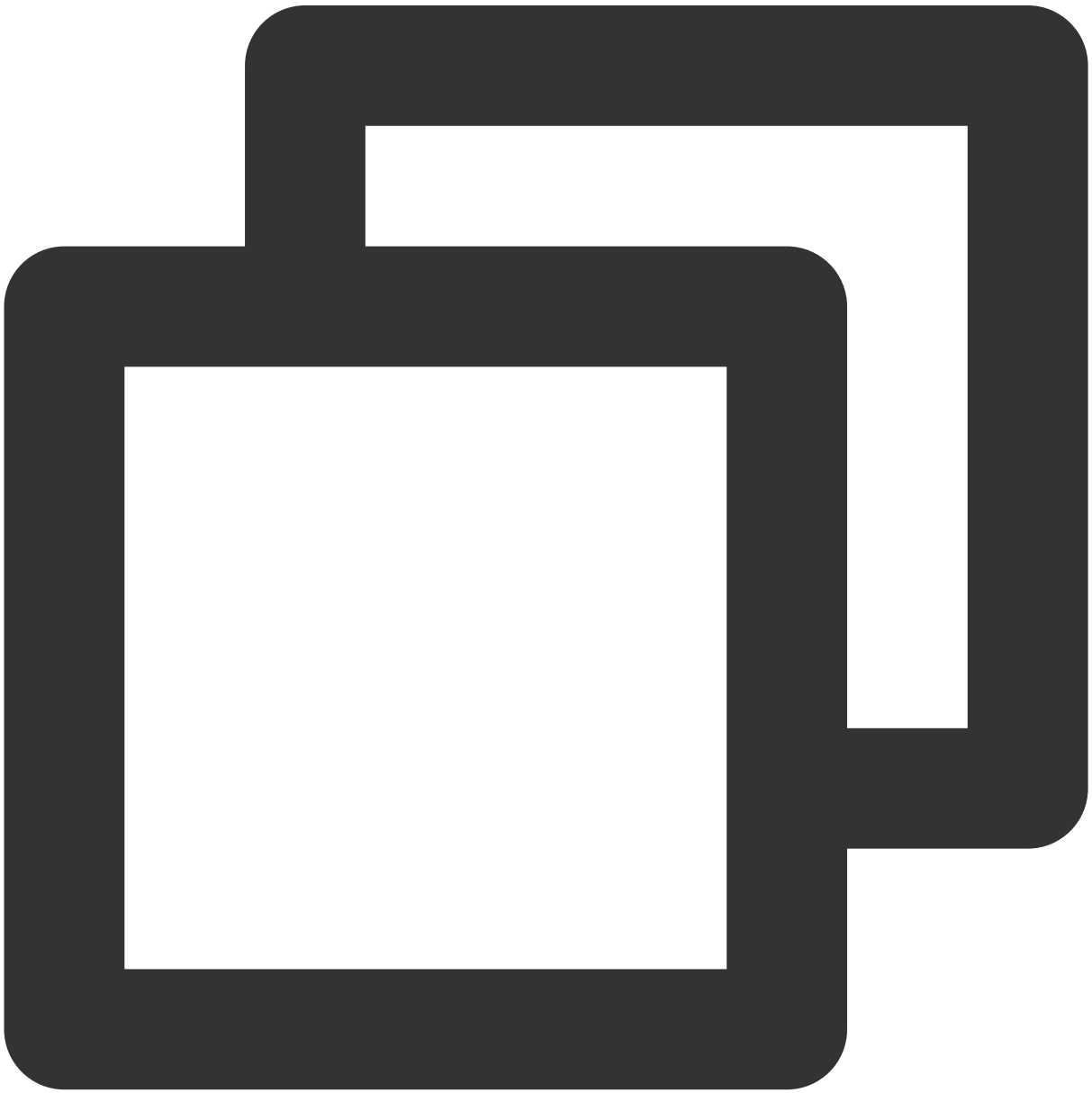
©2013-2022 Tencent Cloud. All rights reserved.

4. API

In order to make it easier for developers to use the capabilities provided by mini programs, TCMPP provides a variety of APIs for developers to use, which can be used to modify the interface of the mini program, get the current location of the device, play video, audio and so on.

There are two types of APIs, synchronous and asynchronous, the synchronous APIs will return the result directly, while the asynchronous APIs will return the result through callbacks. The developer needs to process the logic according to the callback method of the API and pass in the correct parameters to process the business.

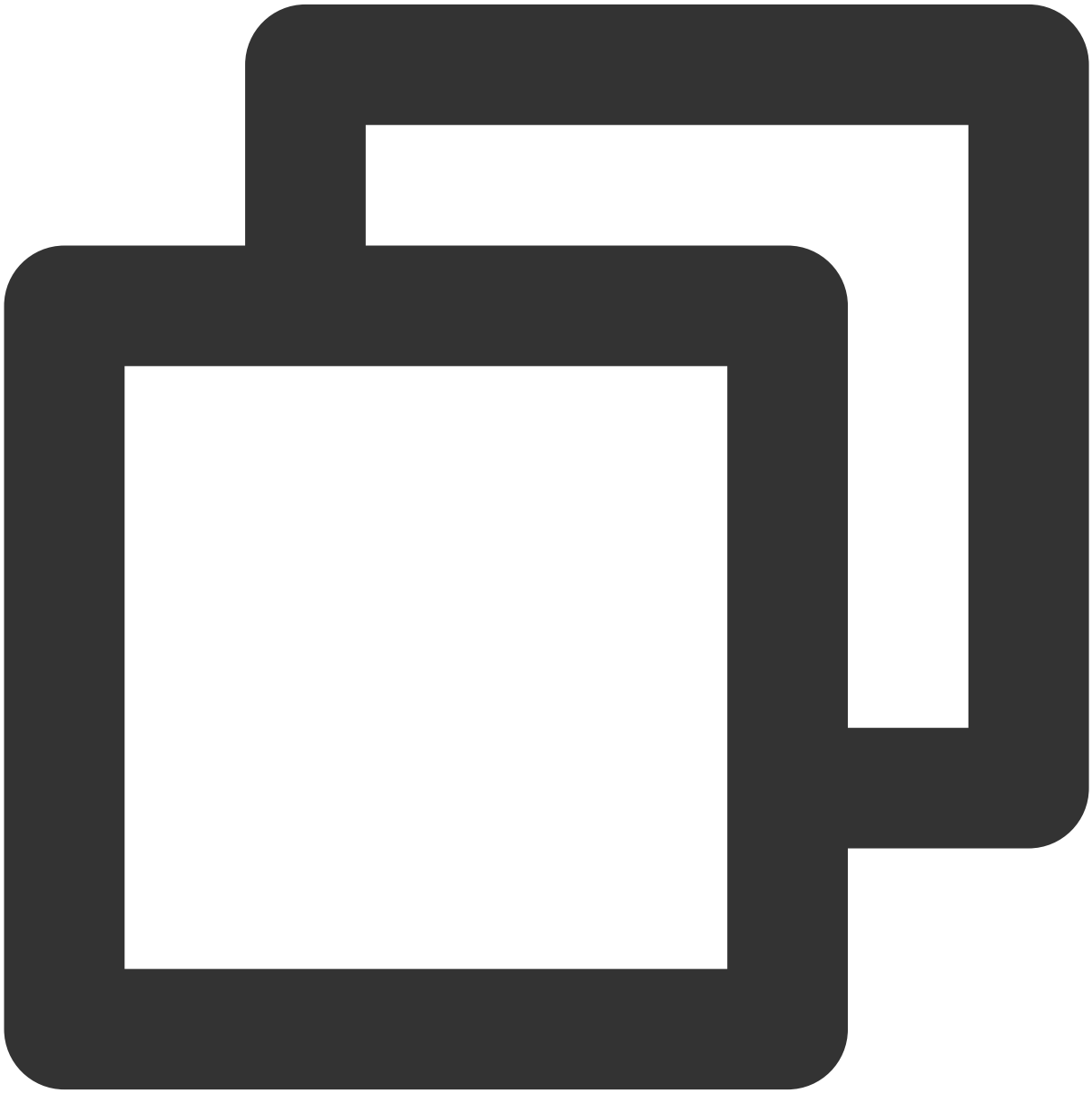
To get the user's geographic location, just call `wx.getLocation` asynchronously:



©2013-2022 Tencent Cloud. All rights reserved.

```
wx.getLocation({
  type: 'wgs84',
  success: (res) => {
    var latitude = res.latitude // latitude
    var longitude = res.longitude // longitude
  }
})
```

To retrieve device information, simply call wx.getSystemInfo synchronously or asynchronously: wx.getSystemInfo.



```
// asynchronous
```

©2013-2022 Tencent Cloud. All rights reserved.

```
wx.getSystemInfo({
  success (res) {
    console.log(res.model)
    console.log(res.pixelRatio)
    console.log(res.windowWidth)
    console.log(res.windowHeight)
    console.log(res.language)
    console.log(res.version)
    console.log(res.platform)
  }
})

// synchronous
try {
  wx.getSystemInfoSync();
} catch(e) {}
```

For more API capabilities see API Overview.

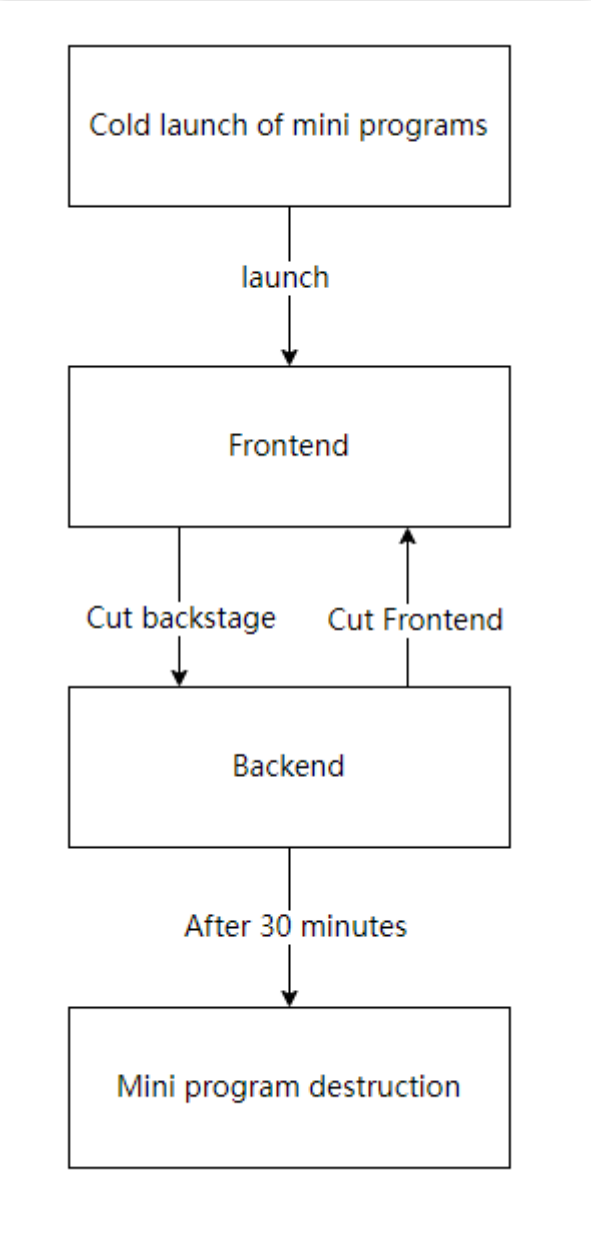
©2013-2022 Tencent Cloud. All rights reserved.

Operation mechanism

Last updated：2024-07-12 17:58:08

1. Mini programme life cycle

From the time a mini program is started to the time it is finally destroyed, it will go through a variety of states and behave differently in different states.



1.1 Mini program startup

There will be two kinds of mini programs, one is cold start and the other is hot start;

©2013-2022 Tencent Cloud. All rights reserved.

Hot start: the process of switching a mini program running in the background to the foreground when the user wants to reopen the opened mini program after a certain period of time;
Cold start: when the user opens the mini program for the first time or reopens the mini program after it has been actively destroyed by the host application, at this time, the mini program needs to be reloaded and started, i.e. cold start.

1.2 Frontend/Backend State

After the mini program starts, the user can directly see the interface of the mini program, at this time the mini program is in the foreground state.
When the user clicks the app control button in the upper right corner to "close" the mini program or presses the Home button on the mobile phone to leave the host application, the mini program is not destroyed but runs in the background, at this time, the mini program can still be run briefly for a short period of time, but the use of some APIs will be restricted.
The ways to cut the background include but not limited to the following:
Tap the capsule button in the upper right corner to leave the mini program;
iOS left from the left side of the screen right slide to leave the mini program;
Android tap the back button to leave the mini program;
Cut the host app directly to the background when the mini program is running in the foreground (gesture or Home button);
Locking the screen directly when the mini program is running in the foreground.
When the user enters the client again and opens the mini program, the mini program will run from the background to the foreground.

1.3 Mini program destruction mechanism

If the user has not used the mini program for a long time, or the system resources are strained, the mini program will be "destroyed", i.e. completely terminated. Specifically, it includes the following situations:
When the mini program enters the background and is "hung", if it does not enter the foreground again for a long time (currently 30 minutes), the mini program will be destroyed.
When a mini program takes up too much system resources, it may be destroyed by the system or recycled by the host app.
On iOS system, when the host app continuously receives system memory alarms within a certain time interval, it will actively destroy the mini program according to a certain strategy and prompt the user to re-open the mini program for lack of running memory.
Specific strategies will be continuously adjusted and optimised.

2. Mini program opening logic

©2013-2022 Tencent Cloud. All rights reserved.

2.1 Cold start opening page

When a mini program is cold started, the page will be opened in the following cases;

If the startup scene does not have path, it goes to the home page;

If the starting scene has a path, it will go to the page with the corresponding path.

2.2 Open page in hot launch

When the mini program is launched, the page will be opened in the following cases;

If there is no path in the scene, it will keep the last viewed page;

If the scene is with path, reLaunch to the page with corresponding path.

©2013-2022 Tencent Cloud. All rights reserved.

Code structure

Last updated：2024-07-12 17:56:51

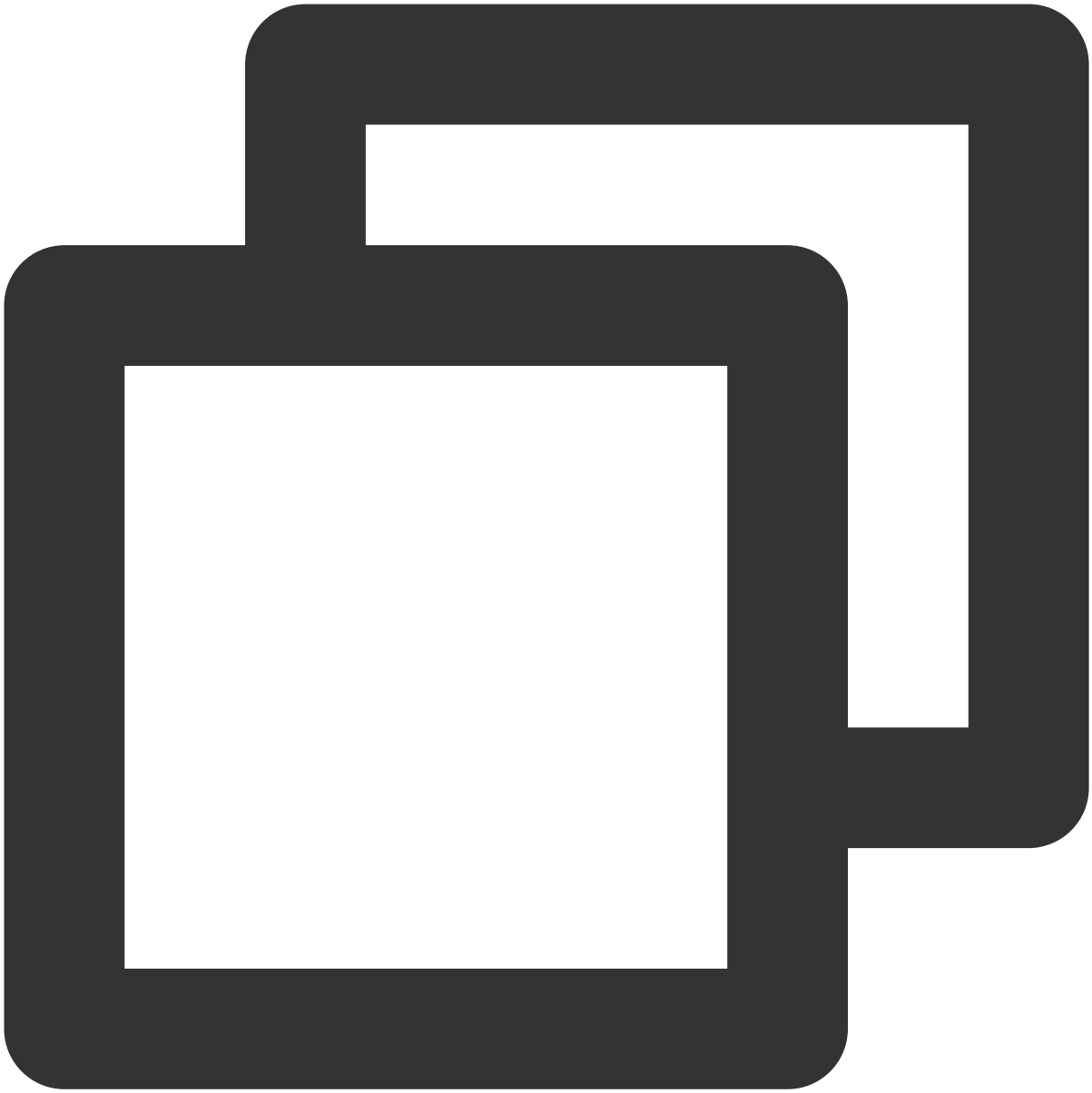
1. JSON configuration

JSON is a data format, rather than a programming language. In the Mini Program, JSON is used for static configuration.
The following sections describes how to use `app.json` and `project.config.json` in the root directory of the project as well as `logs.json` in the `pages/logs` directory.

1.1 app.json for Mini Program Configuration

`app.json` is used for the global configuration of the Mini Program, setting all the page paths, interface behaviors, network timeouts and bottom tabs. The app.json is configured in the QuickStart project as below:

©2013-2022 Tencent Cloud. All rights reserved.



```
{
  "pages":[
    "pages/index/index",
    "pages/logs/logs"
  ],
  "window":{
    "backgroundTextStyle":"light",
    "navigationBarBackgroundColor": "#fff",
    "navigationBarTitleText": "TCMPP",
    "navigationBarTextStyle":"black"
  }
```

©2013-2022 Tencent Cloud. All rights reserved.

```
}
```

Meaning of each configuration item:

The pages field - describes all the page paths of the Mini Program, so that the Weixin app knows the directory where your Mini Program page is defined.

The window field - defines the background color and text color at the top of all pages in the Mini Program.

> For more information on other configuration items, see app.json for Mini Program Configuration.

1.2 project.config.json for Tool Configuration

Usually when you use a tool, you do some personalised configurations for your own preferences, such as compilation configurations, and you have to reconfigure them when you change to another computer and reinstall the tool.

With this in mind, the mini program developer tool generates a project.config.json in the root directory of each project, and any configurations you make on the developer tool are written to this file. When you reinstall the tool or change computers to work on it, you can just load the code package of the same project, and the developer tool will automatically help you restore the personalised configurations to the ones that were used at the time when you were developing the project, which will include the name of the project, the name of the project, and the name of the developer. This will include the name of the project, the option to automatically compress the code when uploading it, and a number of other options.

1.3 page.json for Page Configuration

`page.json` is used for page-related configuration of the Mini Program, for example, the logs.json under the `pages/logs` directory.

If you want the entire Mini Program to be blue, you can specify the top color as blue in `app.json` . However, this may not be the case in reality. You may want to use different colors for different pages to indicate different function modules. Therefore, `page.json` is provided to allow developers to define the properties of each page separately, such as the top color, the option to enable "Pull Down to Refresh", etc.

For more information on other configuration items, see Page Configuration.

1.4 JSON Syntax

Note the following regarding JSON configuration in the Mini Program.

JSON files are wrapped in curly brackets ({}), containing data in the form of key-value pairs. JSON keys must be enclosed in double quotes.

Note：

1. Values in JSON files must be in one of the following data formats. Any other format will trigger an error, such as "undefined" in JavaScript:

Numbers, including floating-point numbers and integers;

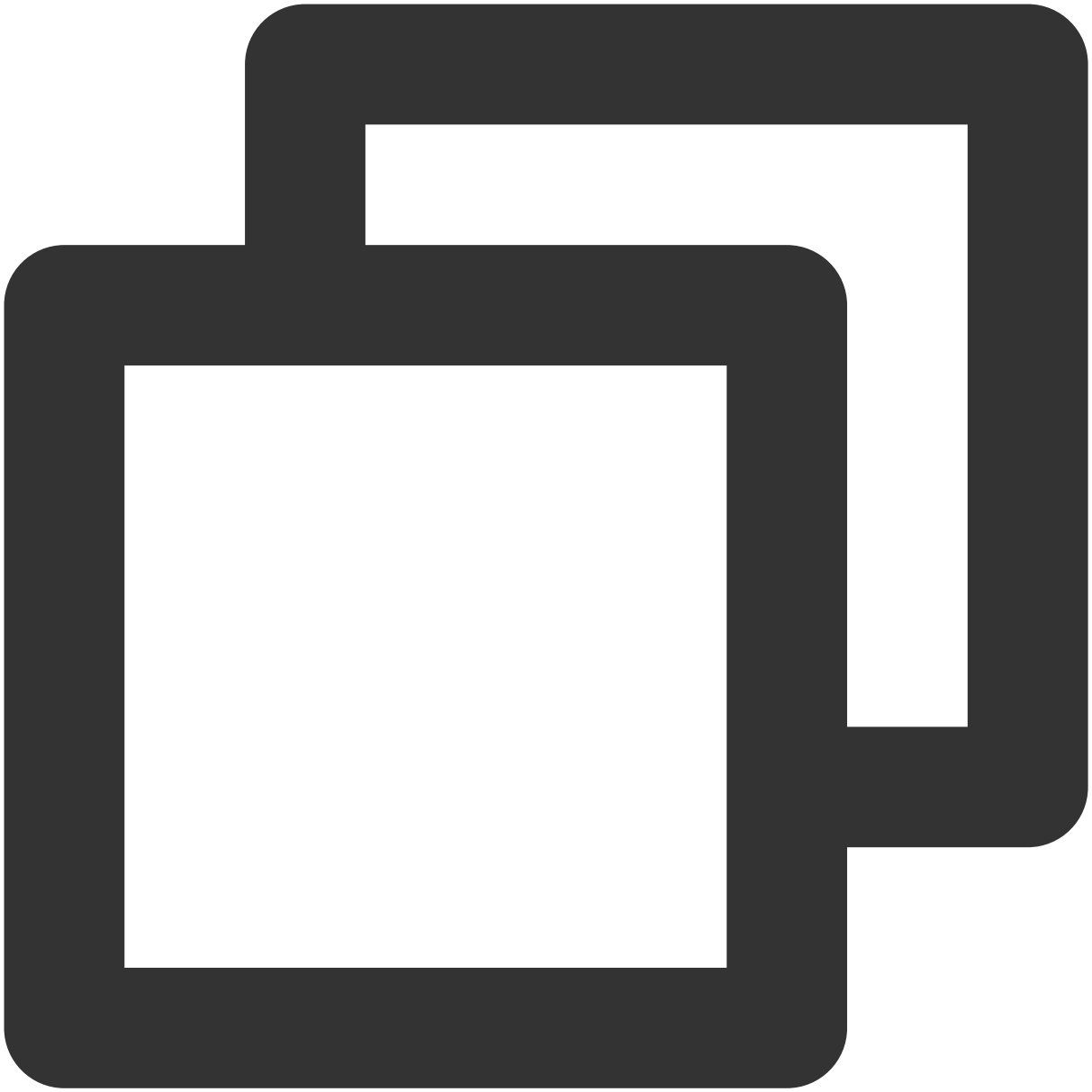
©2013-2022 Tencent Cloud. All rights reserved.

Strings enclosed in double quotes;
Booleans with a value of true or false;
Arrays enclosed in square brackets ([]);
Objects enclosed in curly brackets ({});
Null.
2. Also note that JSON files do not support comments. Adding comments will trigger an error.

2. WXML Template

All web page programmers know that web pages are built using `HTML` , `CSS` and `JavaScript` , where HTML describes the structure of the page, CSS determines the appearance of the page, and JS defines the interaction between the page and the user.
Similar roles exist in the Mini Program, where `WXML` is the equivalent of `HTML` . By opening `pages/index/index.wxml` , you will see the following:

©2013-2022 Tencent Cloud. All rights reserved.



```
<view class="container">
  <view class="userinfo">
    <button wx:if="{{!hasUserInfo && canIUse}}"> getUserInfo </button>
    <block wx:else>
      <image src="{{userInfo.avatarUrl}}" background-size="cover"></image>
      <text class="userinfo-nickname">{{userInfo.nickName}}</text>
    </block>
  </view>
  <view class="usermotto">
    <text class="user-motto">{{motto}}</text>
  </view>
```

©2013-2022 Tencent Cloud. All rights reserved.

```
</view>
```

Similar to HTML, WXML consists of tags and properties. However, there are also some differences, shown as below:

1. Tag names are different

div, p and span are three most commonly used tags when writing HTML, which can be combined in different ways to create different components, such as calendars and pop-ups. Since these common components are required in almost all scenarios, we can put them into packages to improve development efficiency.

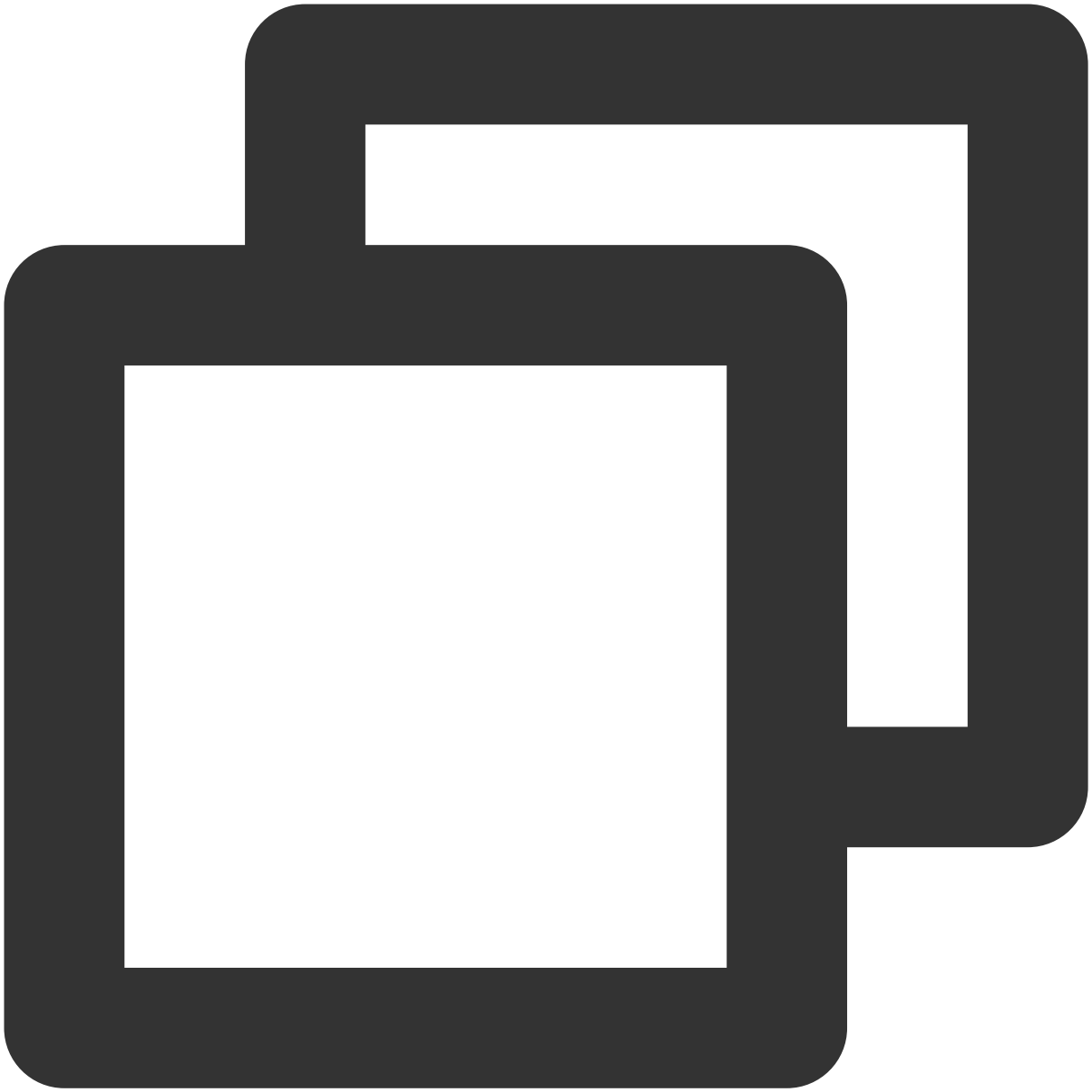
In the above example, the WXML of the Mini Program uses view, button and text tags. In addition to these native basic capabilities, we also provide packaged component capabilities like map, video and audio.

2. Properties like wx:if and expressions like are added

When developing web pages, we generally manipulate DOM (the tree generated from the description of HTML) with JS to allow the interface to respond to users' behaviors. For example, when a user clicks a button, JS records the states in JS variables and manipulates DOM properties or behaviors via the DOM API to trigger changes in the interface. However, as the project gets more and more complex, your code will be full of interface interaction logic and program state variables, which is not conducive to development. Therefore, the MVVM developer mode (e.g., React, Vue) is created to separate rendering from logic. To put it simply, instead of manipulating DOM directly, JS only needs to manage the states, and describes the relationship between states and interface structure via another template syntax.

This also applies to the framework of the Mini Program. If you want to display the Hello World string in the interface: Write the WXML as follows:

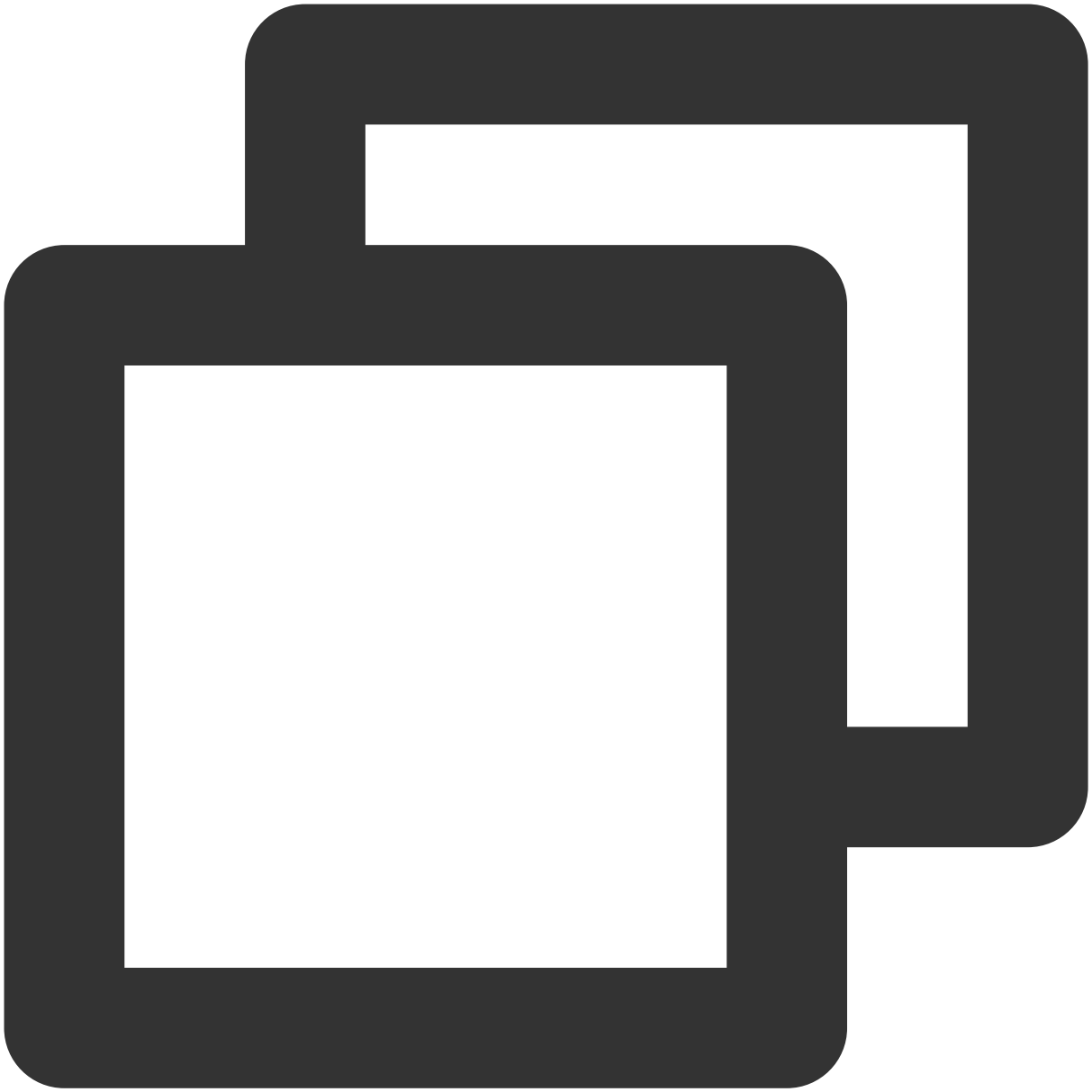
©2013-2022 Tencent Cloud. All rights reserved.



```
<text>{{msg}}</text>
```

Use JS to manage the state only:

©2013-2022 Tencent Cloud. All rights reserved.



```
this.setData({ msg: "Hello World" })
```

The process that binds a variable to the interface via the syntax is called data binding. Data binding is not enough to describe the relationship between states and interface. Control capabilities such as if/else and for are required, which are expressed with properties that start with wx: in the Mini Program.

Refer to WXML Syntax Reference for more details.

©2013-2022 Tencent Cloud. All rights reserved.

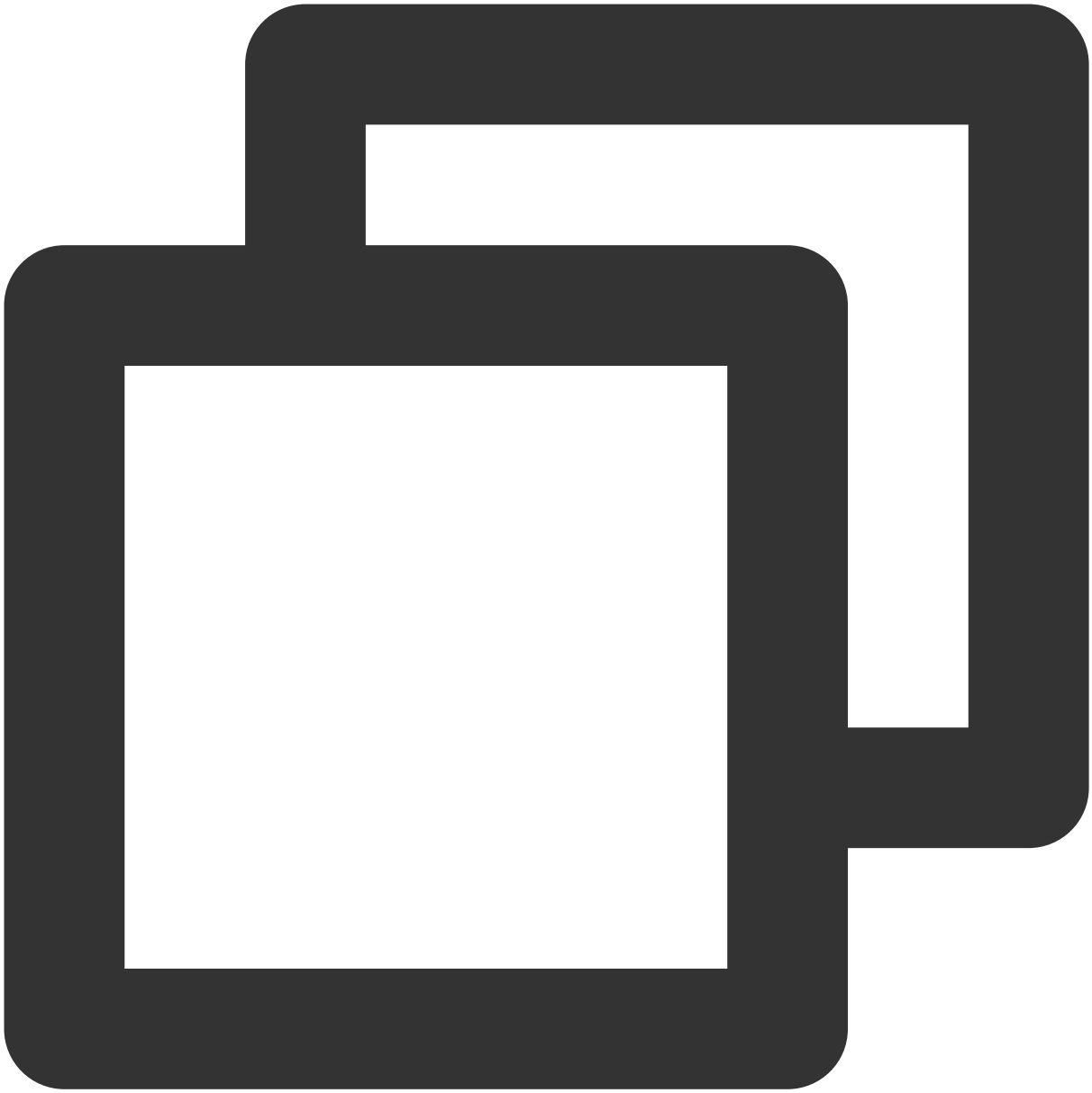
3. WXSS Style

WXSS has most of the features of CSS, but incorporates some new features and modifications for the Mini Program.
1. New size unit is added. When writing the CSS style, developers need to convert the pixel units to adapt to different mobile device screens with different widths and pixel ratios. WXSS supports the new rpx unit at its underlying layer, allowing the Mini Program to take over the job from developers and convert the units at the underlying layer. Since the units are converted using floating-point operations, the result may deviate slightly from the expected one.
2. Global and local styles are provided. Similar to the above app.json and page.json, app.wxss can be written as the global style, which applies to all the pages of the Mini Program. The local page style page.wxss only applies to the current page.
3. WXSS only supports certain CSS selectors
Refer to WXSS for more details.

4. JS Logic Interaction

It's not enough for a service to just display the interface. Interaction with users is required, such as responding to user's clicks and obtaining user's location. In the Mini Program, we process user's operations by writing JS scripts.

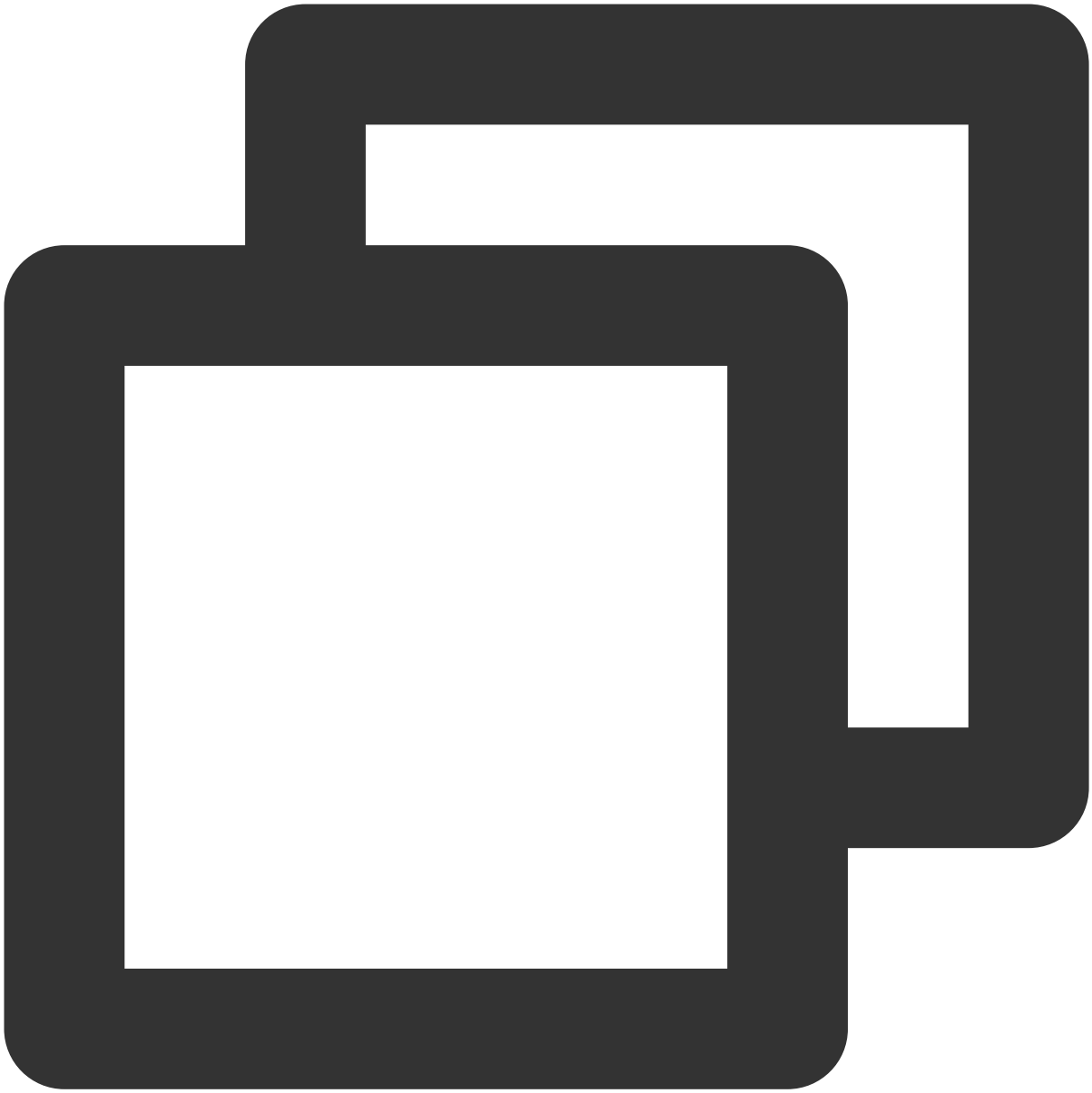
©2013-2022 Tencent Cloud. All rights reserved.



```
<view>{{ msg }}</view>
<button bindtap="clickMe">Click me</button>
```

When a button is clicked, we want to display msg as "Hello World" on the interface. To do this, we declare the bindtap property on the button and the clickMe method in the JS file to respond to this click:

©2013-2022 Tencent Cloud. All rights reserved.



```
Page({
  clickMe: function() {
    this.setData({ msg: "Hello World" })
  }
})
```

Responding to user actions is as simple as that. For more details on events, see View Layer - Event System.

©2013-2022 Tencent Cloud. All rights reserved.

Directory Structure

Last updated：2024-07-12 19:03:42

The mini program comprises an overarching `app` that defines the entire application, along with multiple `page` elements, each describing their respective pages.

1. The core of a mini program is constituted by three essential files, which must be situated in the root directory of the project, as follows:

File	Indispensable	Effect
app.js	Yes	Mini program logic
app.json	Yes	Mini program's public configuration
app.wxss	No	List of mini program's public styles

2. A mini program page is composed of four files, namely:

File	Indispensable	Effect
js	Yes	Page logic
wxml	Yes	Page Structure
json	No	Page Configuration
wxss	No	Page style list

Note:

To facilitate developers and reduce configuration items, the four files describing the page must possess identical paths and filenames.

Files allowed to be uploaded

Within the project directory, the following files undergo compilation and hence cannot be directly accessed after upload: *.js, app.json, *.wxml, *.wxss (wherein wxml and wxss files are only applicable to pages configured in app.json). Beyond this, only files with suffixes included in the allowlist can be uploaded. The specific allowlist is as follows:

1. qs
2. png

©2013-2022 Tencent Cloud. All rights reserved.

3. jpg
4. jpeg
5. gif
6. svg
7. json
8. cer
9. mp3
10. aac
11. m4a
12. mp4
13. wav
14. ogg
15. silk

©2013-2022 Tencent Cloud. All rights reserved.

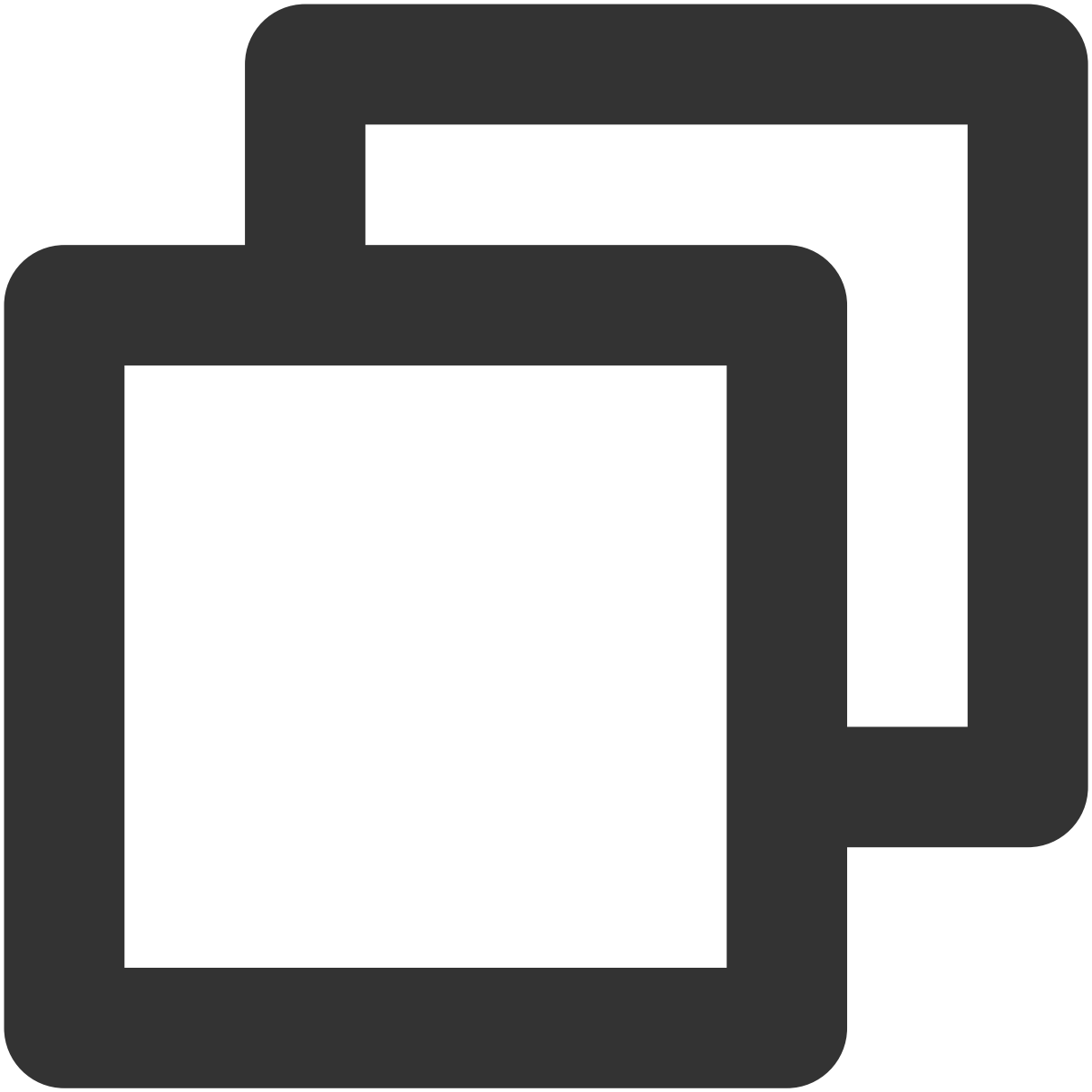
Configuring mini program
Global and Page Configuration

Last updated：2024-07-12 17:51:08

1. Global Configuration

The `app.json` file in the root directory of the mini program is used to configure the global configuration of the mini program, deciding the path of the page file, the window performance, setting the network timeout, setting multiple tabs and so on.
The following is an app.json file that contains some common configuration options:

©2013-2022 Tencent Cloud. All rights reserved.



```
{
  "pages": [
    "pages/index/index",
    "pages/logs/index"
  ],
  "window": {
    "navigationBarTitleText": "Demo"
  },
  "tabBar": {
    "list": [{
      "pagePath": "pages/index/index",
```

©2013-2022 Tencent Cloud. All rights reserved.

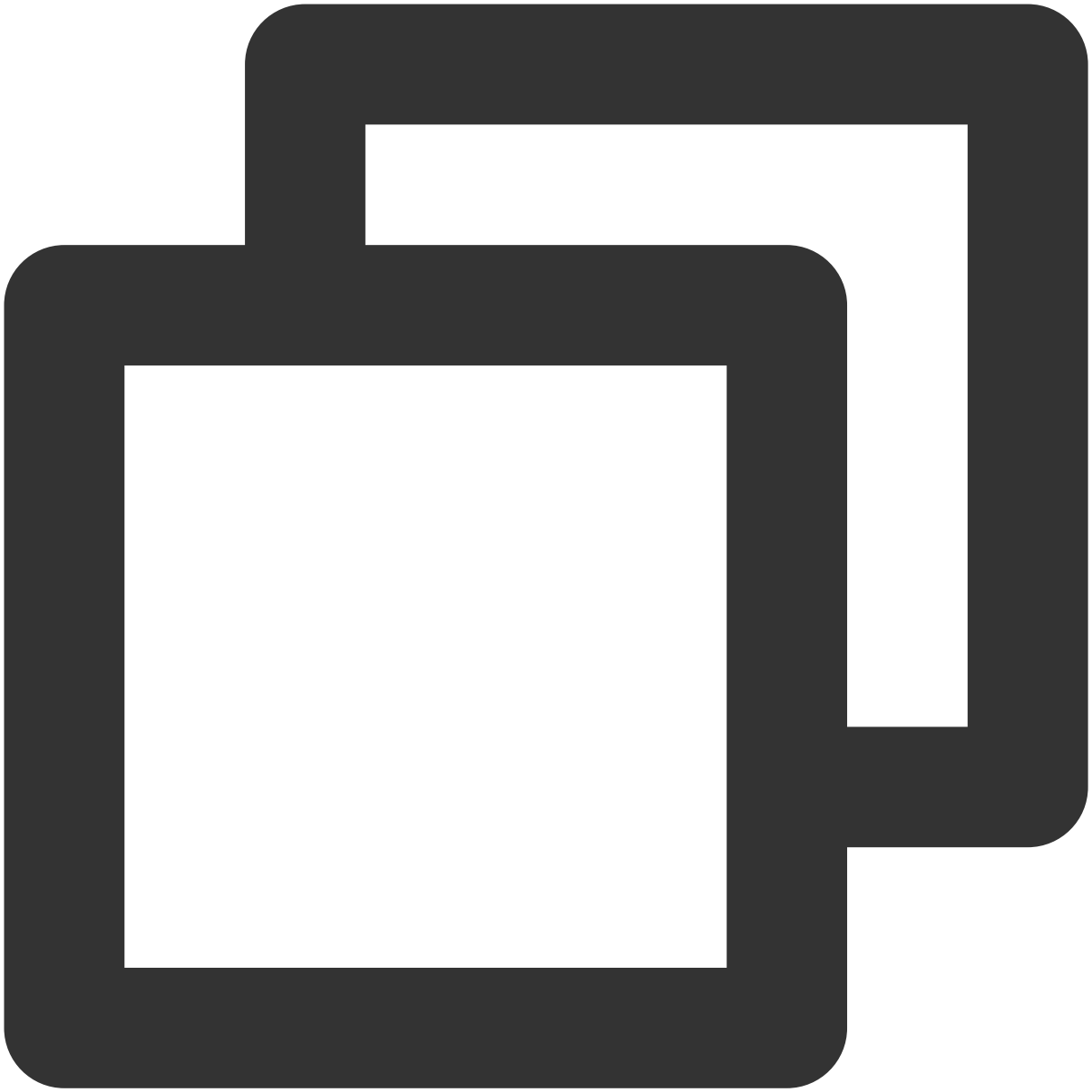
```
      "text": "index"
    }, {
      "pagePath": "pages/logs/index",
      "text": "log"
    }]
  },
  "debug": true
}
```

Please refer to global configuration for a complete description of the configuration items.

2. Page Configuration

Each mini program page can also use the same name .json file to configure the window of the page, the configuration items in the page will override the same configuration items in the window of app.json.
For example:

©2013-2022 Tencent Cloud. All rights reserved.



```
{
  "navigationBarBackgroundColor": "#ffffff",
  "navigationBarTextStyle": "black",
  "navigationBarTitleText": "title",
  "backgroundColor": "#eeeeee",
  "backgroundTextStyle": "light"
}
```

Please refer to Page Configuration for full configuration item descriptions.

©2013-2022 Tencent Cloud. All rights reserved.

©2013-2022 Tencent Cloud. All rights reserved.

Mini Program Framework Introduce

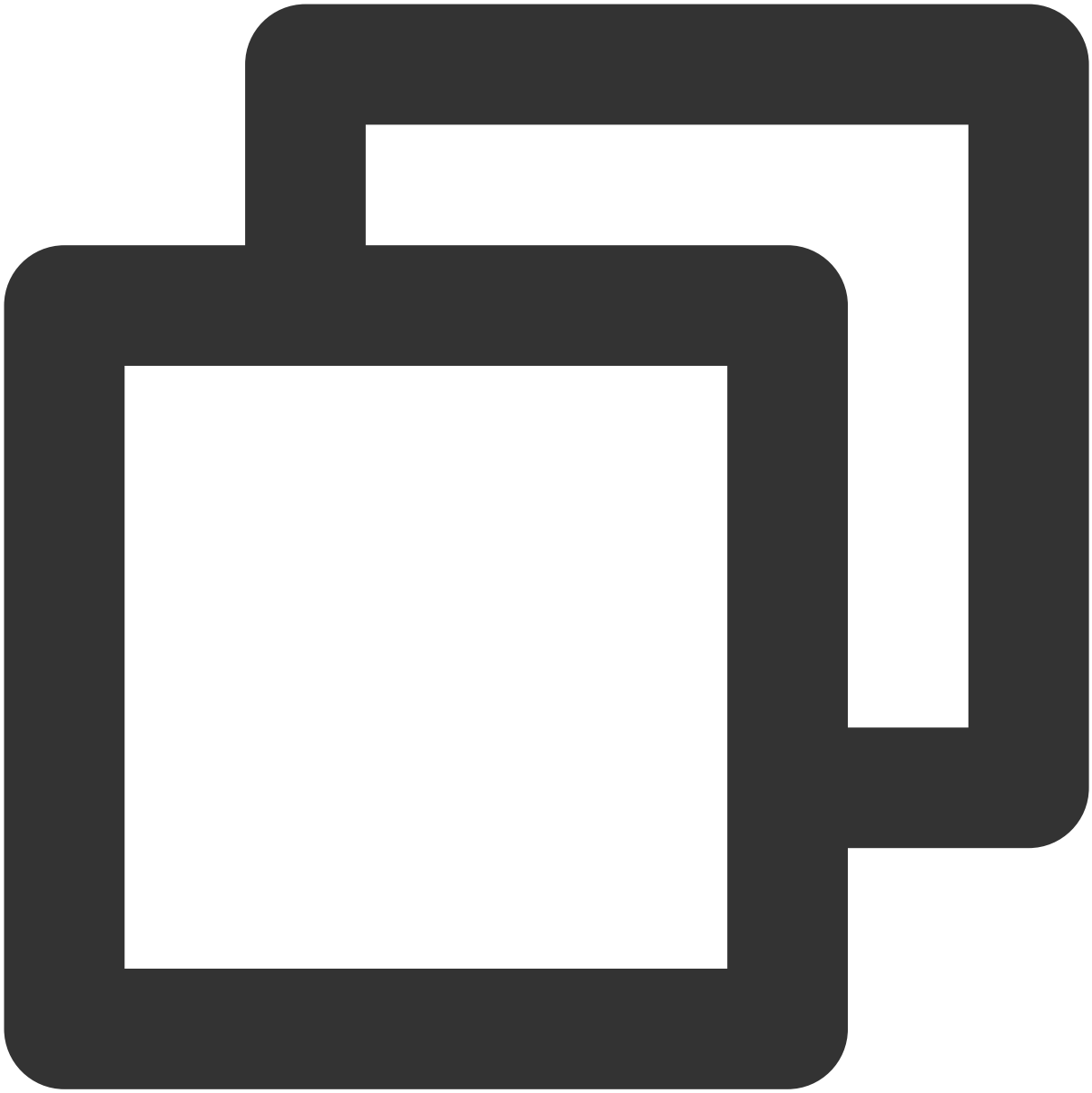
Last updated：2024-07-12 17:49:11

TCMPP development framework is designed to allow developers to develop services with a native app experience in Weixin by the simplest and most efficient means possible.
The Mini Program framework system includes Logic Layer (App Service) and View Layer (View). The Mini Program provides its own view layer description languages WXML and WXSS, and a logic layer framework based on JavaScript, as well as a data transfer and event system between the view layer and the logic layer, allowing developers to focus on data and logic.

Reactive Data Binding

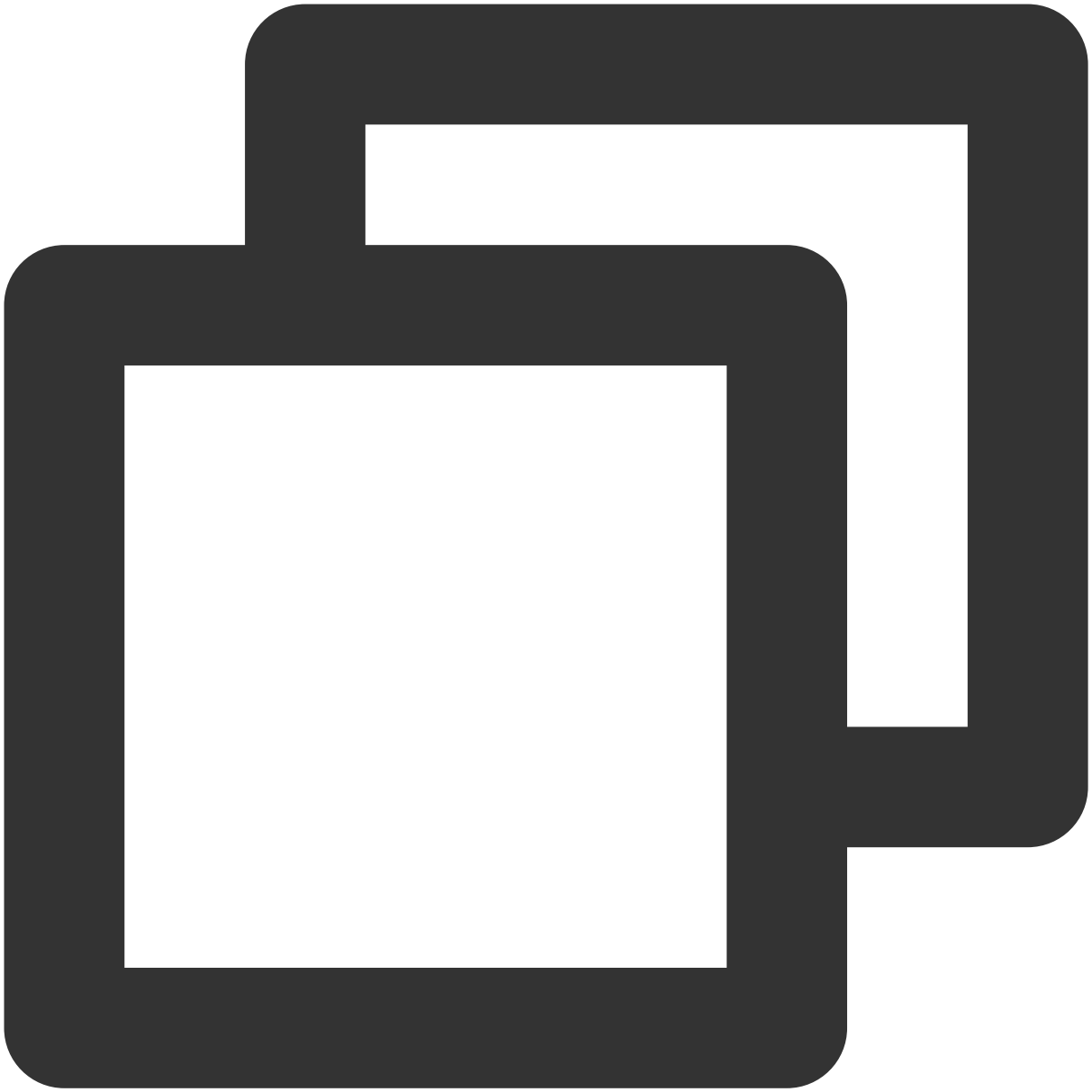
The reactive data binding system is the core of the framework. It provides an extremely simple way to ensure the synchronization of data and views. To modify data, you simply need to modify the data on the logic layer and it will be updated in the view layer.
Here is a simple example:

©2013-2022 Tencent Cloud. All rights reserved.



```
<!-- This is our View -->
<view> Hello {{name}}! </view>
<button bindtap="changeName"> Click me! </button>
```

©2013-2022 Tencent Cloud. All rights reserved.



```
// This is our App Service.
// This is our data.
var helloData = {
  name: 'TCMPP'
}

// Register a Page.
Page({
  data: helloData,
  changeName: function(e) {
    // sent data change to view
```

©2013-2022 Tencent Cloud. All rights reserved.

```
      this.setData({
        name: 'World'
      })
    }
  })
```

The developer uses the framework to bind `name` in the logic layer data to `name` in the view layer, so `Hello TCMPP!` appears once the page opens.

When the button is clicked, the view layer sends `changeName` events to the logic layer, which finds and executes the corresponding event handler.

After the callback function is triggered, the logic layer executes `setData` to change the name in `data` from TCMPP to `World`. Since the data is bound to the view layer, `Hello World!` is automatically applied to the view layer.

Page Management

The `framework` manages the page routing of the entire Mini Program, which enables seamless switching between pages and gives the pages a complete lifecycle. The developer only needs to register the data, methods, and lifecycle functions of the page to the `framework`. All other complex operations are handled by the `framework`.

Basic Components

The framework provides a set of basic components that come with Weixin styles and special logic. Developers can create powerful Weixin Mini Programs by combined use of these components.

Wide Range of APIs

The `framework` features a wide range of Weixin native APIs that can easily call the capabilities provided by Weixin, such as access to user information, local storage, and payment.

©2013-2022 Tencent Cloud. All rights reserved.

Logic Layer

Last updated：2024-07-12 17:48:00

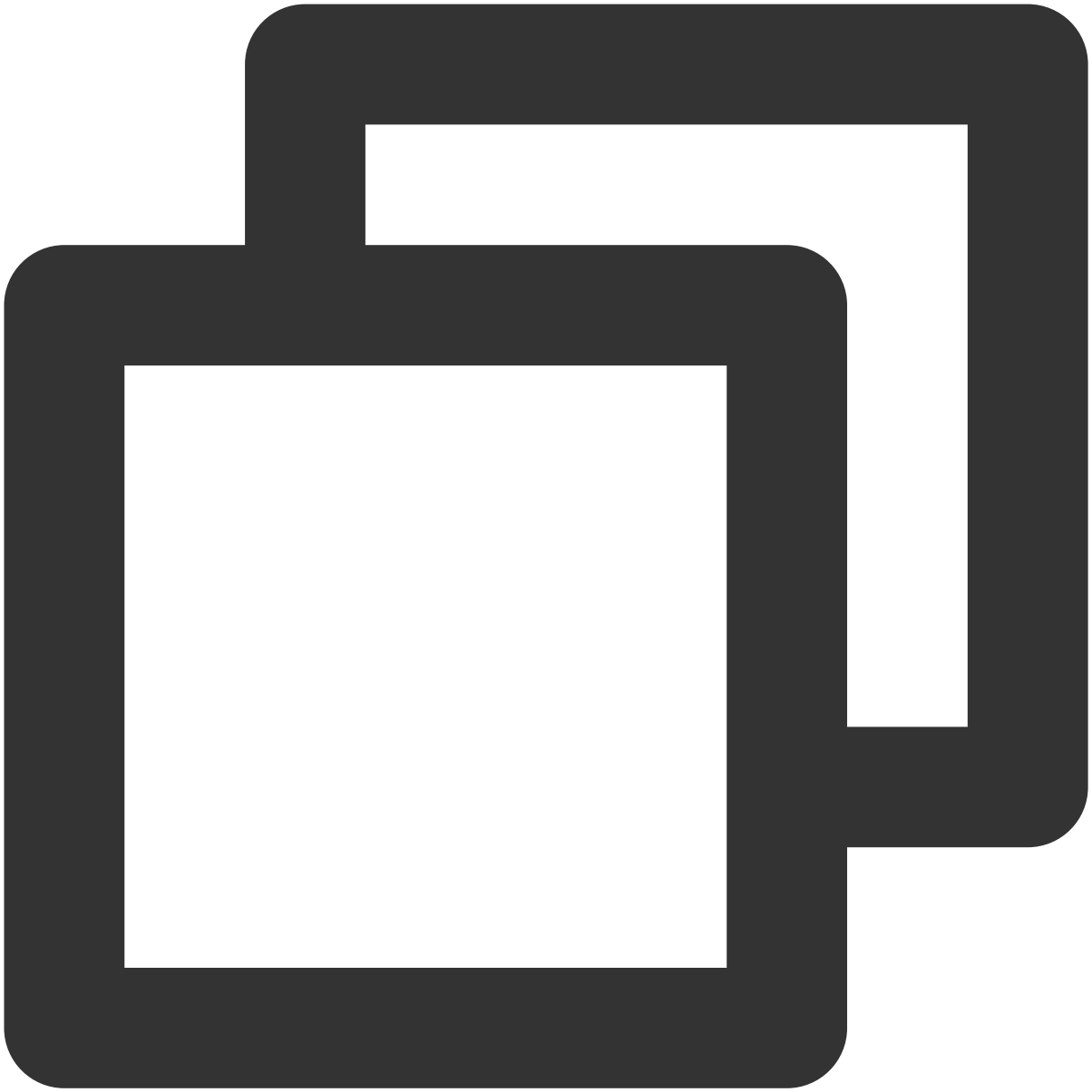
1. Introduce

At the logic layer of the Mini Program development framework, the `JavaScript` engine is used to provide a Mini Program with features specific to TCMPP Mini Programs and an environment for running JavaScript code written by developers.
The logic layer processes data and sends processed data to the view layer, and receives event feedback from the view layer.
All code written by developers will be packaged into a `JavaScript` file, which runs when the Mini Program is started until the Mini Program is destroyed. This behavior is similar to ServiceWorker. Therefore, the logic layer is also referred to as an `App Service` .
Based on `JavaScript` , we have added some features to facilitate the development of Mini Programs:
App and Page methods used for Program Registration and Registering Page;
getApp and getCurrentPages methods, respectively used to obtain the App instance and the current page stack;
Rich API, such as Weixin user data, Scan, WeChat Pay, and other Weixin-specific capabilities;
Modularization capability, allowing each page to have an independent [Scope](./module.md#File Scope).
Note：
The logic layer of the Mini Program development framework does not run on browsers. Therefore, web-based capabilities of JavaScript, such as window and document, are unavailable.

2. Registering a Mini Program Account

For each Mini Program, a Mini Program instance needs to be registered by calling the App method via app.js, to link the lifecycle callback function, error listening function, page not found listening function, and the like.
For specific definitions and usage of the parameters, refer to App.

©2013-2022 Tencent Cloud. All rights reserved.

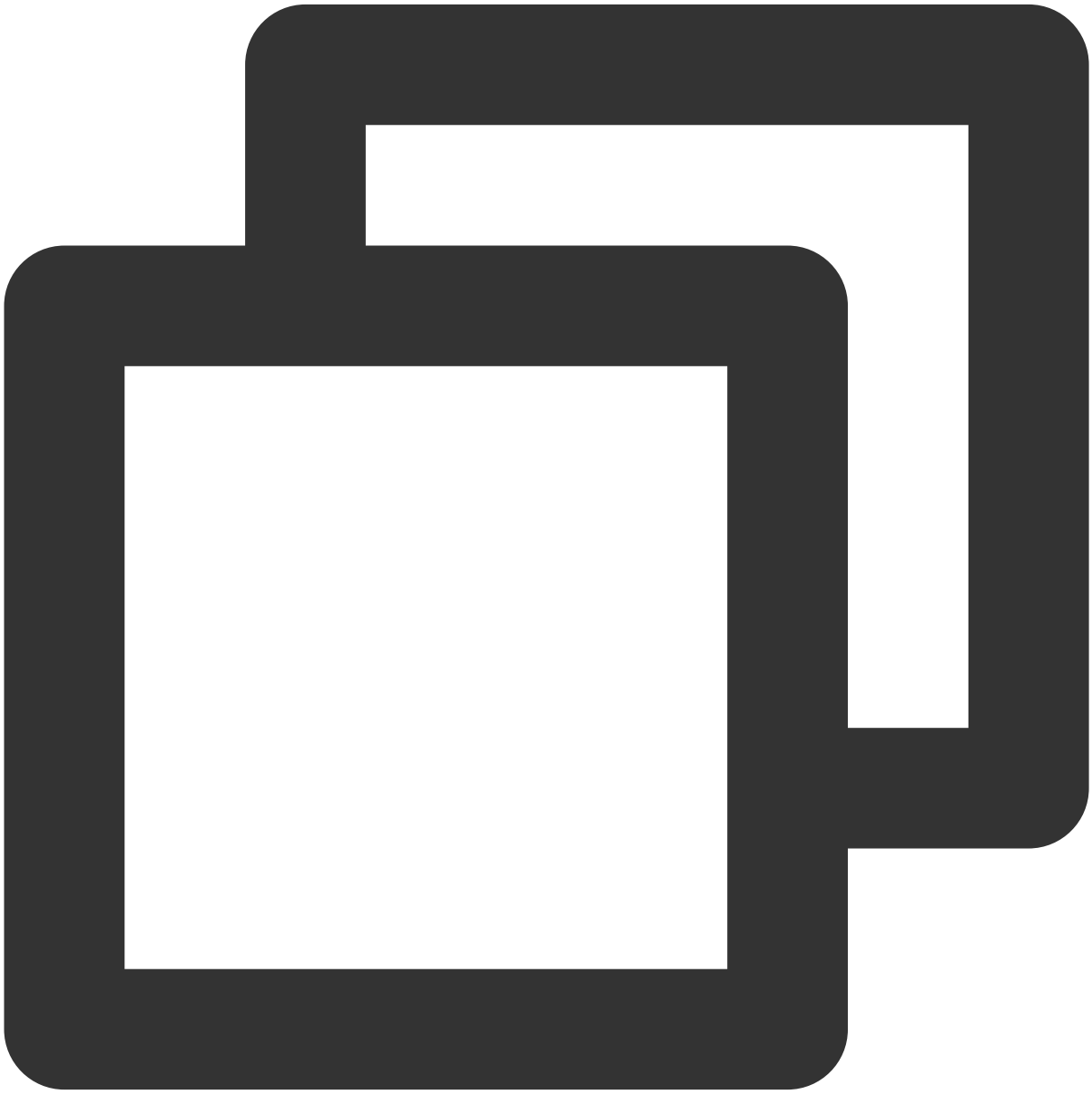


```
// app.js
App({
  onLaunch (options) {
    // Do something initial when launch.
  },
  onShow (options) {
    // Do something when show.
  },
  onHide () {
    // Do something when hide.
  },
```

©2013-2022 Tencent Cloud. All rights reserved.

```
  onError (msg) {
    console.log(msg)
  },
  globalData: 'I am global data'
})
```

A Mini Program has only one app instance, which is shared to all pages. Developers can obtain the globally unique app instance by using the `getApp` method, and then obtain data in the app or call a function registered with App by the developers.



```
// xxx.js
```

©2013-2022 Tencent Cloud. All rights reserved.

```
const appInstance = getApp()
console.log(appInstance.globalData) // I am global data
```

3. Registering Page

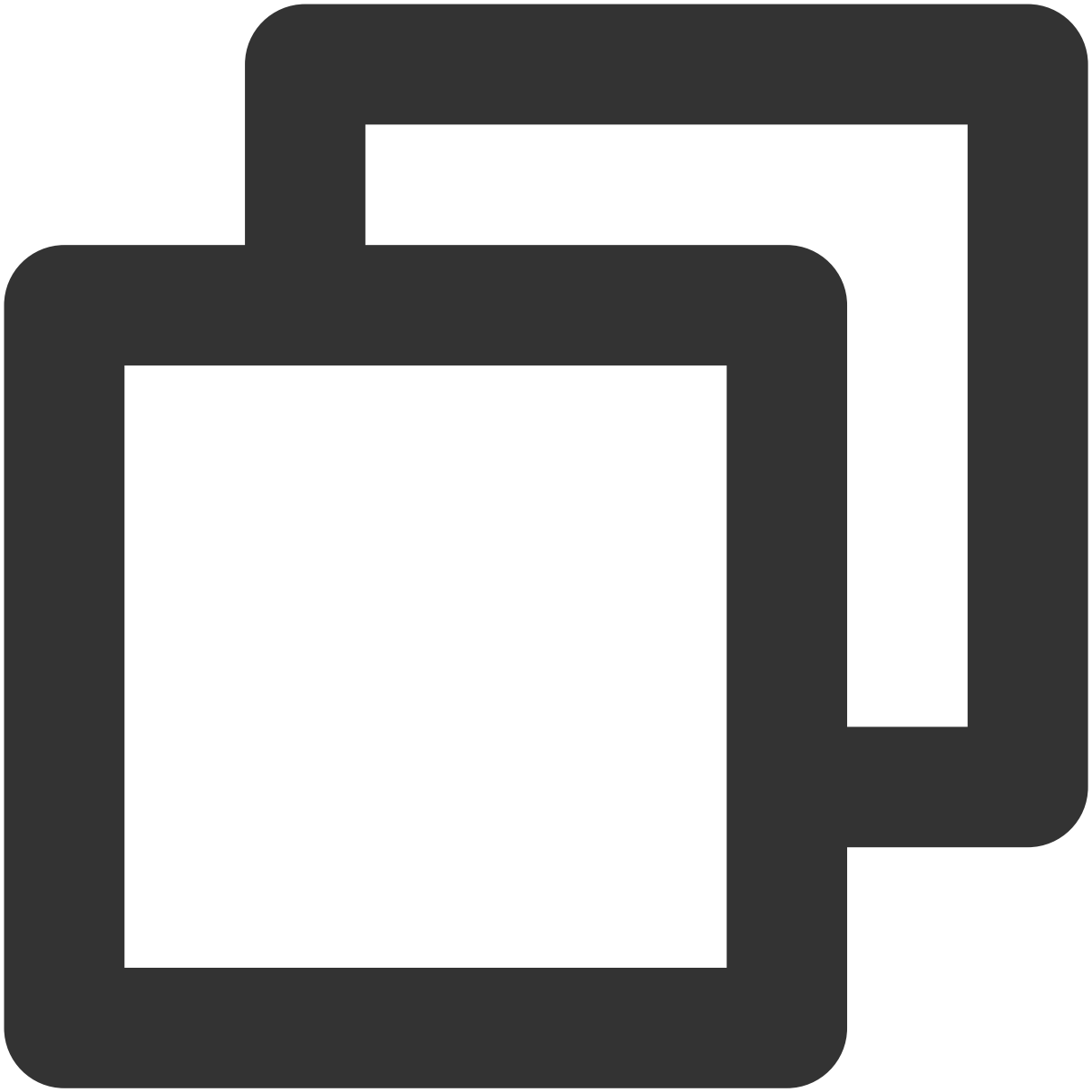
For each page of a Mini Program, a page instance needs to be registered by calling the Page method via the js file corresponding to the page, to specify the initial data of the page, the lifecycle callback function, the event processing function, and the like of the page.

3.1 Registering a Page with the Page Constructor

Simple pages can be constructed using Page().

Code example:

©2013-2022 Tencent Cloud. All rights reserved.



```
// index.js
Page({
  data: {
    text: "This is page data."
  },
  onLoad: function(options) {
    // Do some initialize when page load.
  },
  onReady: function() {
    // Do something when page ready.
  },
```

©2013-2022 Tencent Cloud. All rights reserved.

```
  onShow: function() {
    // Do something when page show.
  },
  onHide: function() {
    // Do something when page hide.
  },
  onUnload: function() {
    // Do something when page close.
  },
  onPullDownRefresh: function() {
    // Do something when pull down.
  },
  onReachBottom: function() {
    // Do something when page reach bottom.
  },
  onShareAppMessage: function () {
    // return custom share data when user share.
  },
  onPageScroll: function() {
    // Do something when page scroll
  },
  onResize: function() {
    // Do something when page resize
  },
  onTabItemTap(item) {
    console.log(item.index)
    console.log(item.pagePath)
    console.log(item.text)
  },
  // Event handler.
  viewTap: function() {
    this.setData({
      text: 'Set some data for updating view.'
    }, function() {
      // this is setData callback
    })
  },
  customData: {
    hi: 'MINA'
  }
})
```

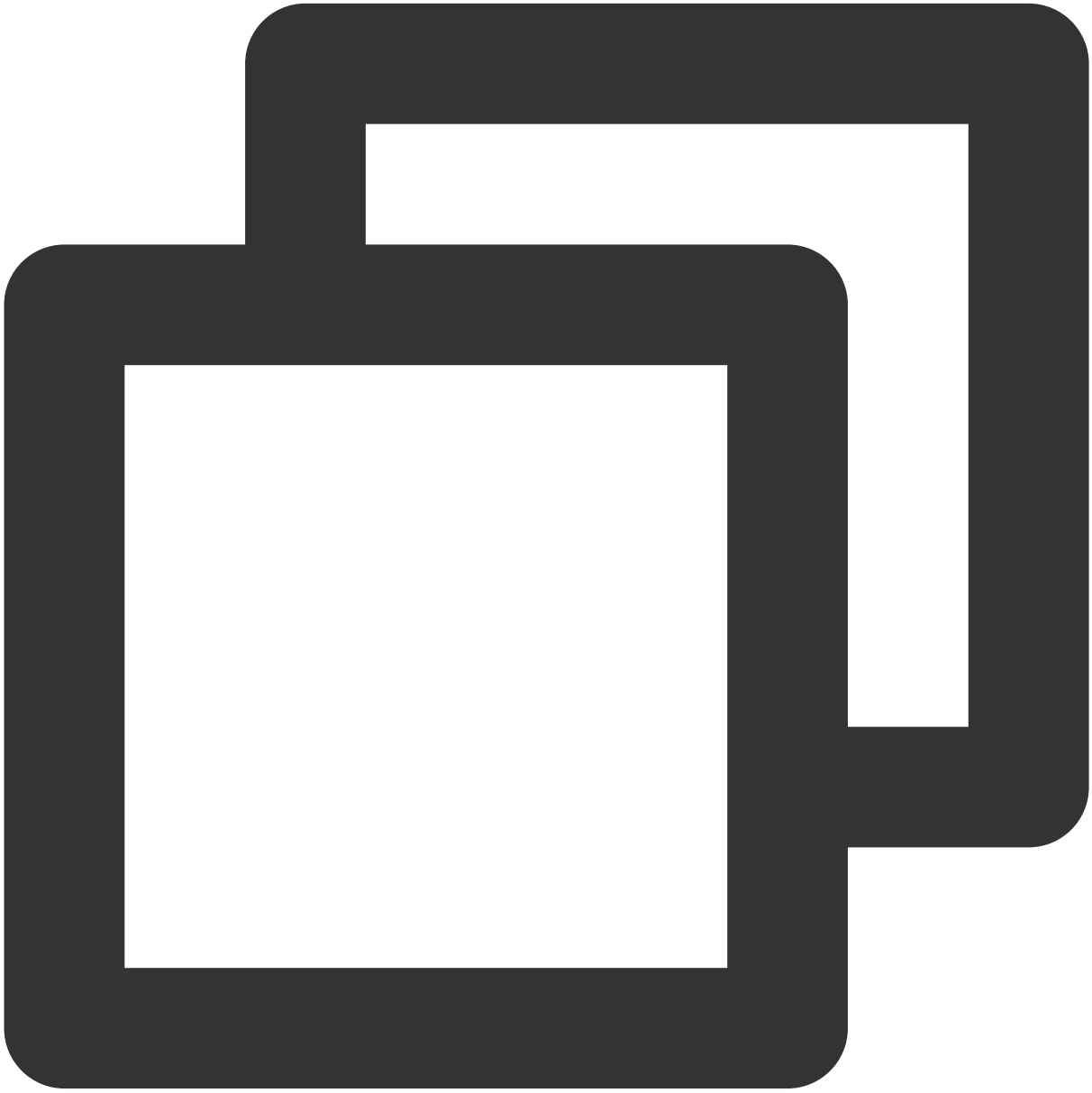
For specific definitions and usage of the parameters, refer to Page.

©2013-2022 Tencent Cloud. All rights reserved.

3.2 Constructing Pages with the Component Constructor

The Page constructor works for simple pages. However, for complex pages, the Page constructor may not work well. In this case, you can use the Component constructor to construct the page. The main difference of the Component constructor is that the methods need to be inside methods: { }.

Code example:



```
Component({
  data: {
    text: "This is page data."
  },
```

©2013-2022 Tencent Cloud. All rights reserved.

```
  methods: {
    onLoad: function(options) {
      // Executed when the page is created
    }, onPullDownRefresh: function() { // Execute on page creation.
    onPullDownRefresh: function() {
      // onPullDownRefresh: function() { // On page creation }, onPullDownRefresh:
    }, onPullDownRefresh.
    // Event response function
    viewTap: function() {
      // ...
    }
  }
})
```

This is very similar to a Custom Components in that you can use advanced features such as behaviors in the same way as a custom component.

> Read the Component Constructors section for more details.

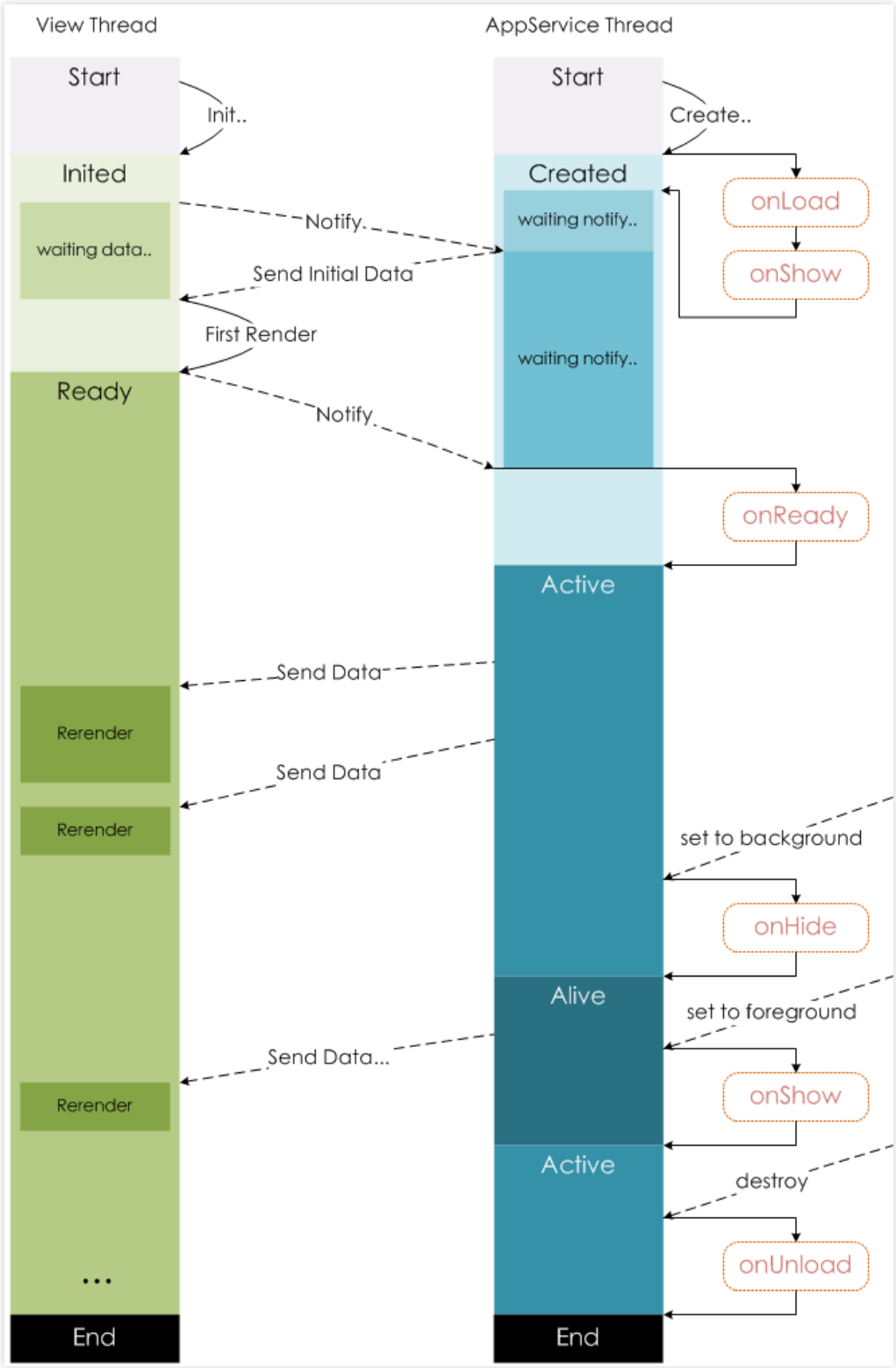
4. Page Lifecycle

Note

You don't need to fully understand the following content right away, but it will help in the future.

The following figure describes the lifecycle of a `Page` instance.

©2013-2022 Tencent Cloud. All rights reserved.



5. Page Routing

The framework manages all the routes to pages of a Mini Program.

5.1 Page Stack

©2013-2022 Tencent Cloud. All rights reserved.

The framework maintains all the existing pages via stacks. When a route is switched, the page stack reacts as follows:

Routing	Reaction on the Page Stack
Initialization	The new page is pushed onto the stack.
Opening a new page	The new page is pushed onto the stack.
Page redirection	The current page is popped out of the stack, and the new page is pushed onto the stack.
Page return	Pages are successively popped out of the stack until the destination return page is displayed.
Tab switching	All the pages are popped out of the stack, and only the new tab page is retained.
Reloading	All the pages are popped out of the stack, and only the new page is retained.

Developers can use the getCurrentPages() function to get the current page stack, given as an array in the order of the stack, with the first element being the home page and the last element being the current page.

Note：

Don't try to modify the page stack, it will cause routing and page state errors.

Don't call getCurrentPages() on App.onLaunch when the page has not been generated.

5.2 Routing

The following shows how page routing is triggered and the lifecycle functions for pages.

Routing	Trigger Condition	Source Page	Routed-To Page
Initialization	A page of the Mini Program is opened for the first time.	-	onLoad, onShow
Opening a new page	The API wx.navigateTo is called. The component <navigator open-type="navigateTo"/> is used.	onHide	onLoad, onShow
Page redirection	The API wx.redirectTo is called. The component <navigator open-type="redirectTo"/> is used.	onUnload	onLoad, onShow
Page return	The API wx.navigateBack is called. The component <navigator open-type="navigateBack"> is used. The user taps the return button on the top left.	onUnload	onShow

©2013-2022 Tencent Cloud. All rights reserved.

Tab switching	The API wx.switchTab is called. The component <navigator open-type="switchTab"/> is used. The user switches between tabs.	-	For details, refer to the following table.
Restart	The API wx.reLaunch is called. The component <navigator open-type="reLaunch"/> is used.	onUnload	onLoad, onShow

Lifecycle functions corresponding to tab switching (in the following example, pages A and B are Tabbar pages, page C is obtained after page A is opened, and page D is obtained after page C is opened):

Current Page	Routed-To Page	Triggered Lifecycle Function (In Order)
A	A	Nothing happend
A	B	A.onHide(), B.onLoad(), B.onShow()
A	B (Opened again)	A.onHide(), B.onShow()
C	A	C.onUnload(), A.onShow()
C	B	C.onUnload(), B.onLoad(), B.onShow()
D	B	D.onUnload(), C.onUnload(), B.onLoad(), B.onShow()
D (Entered due to redirection)	A	D.onUnload(), A.onLoad(), A.onShow()
D (Entered due to redirection)	B	D.onUnload(), B.onLoad(), B.onShow()

Note：

navigateTo and redirectTo can be used to open non-tabBar pages only.

switchTab can be used to open tabBar pages only.

reLaunch can be used to open any pages.

The tabBar on the bottom of a page is determined by the page. In other words, tabBar is displayed on the bottom of all pages defined as tabBar.

Parameters used to call page routing can be obtained via onLoad in the destination page.

6. Modules

6.1 Modularization

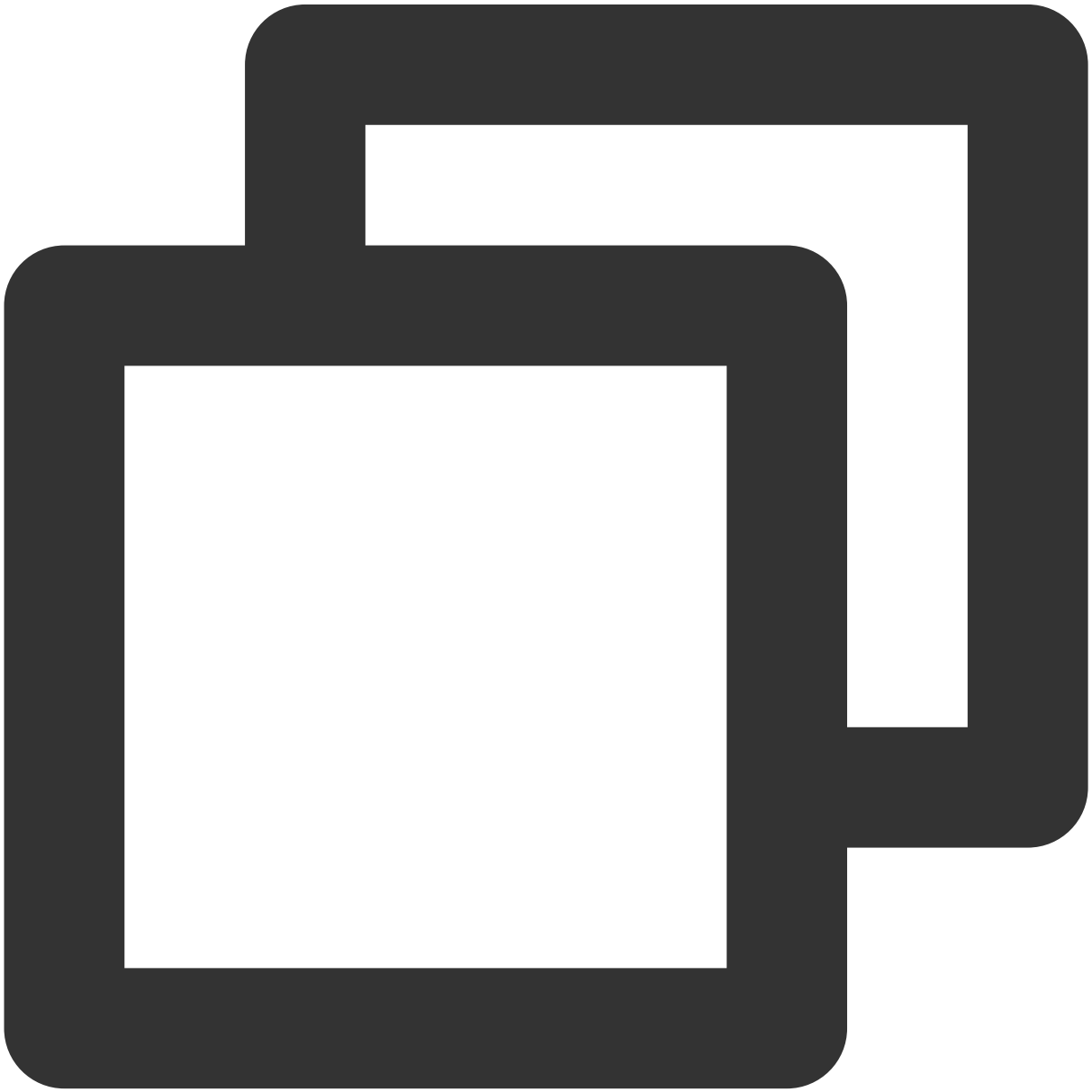
©2013-2022 Tencent Cloud. All rights reserved.

Some common codes can be extracted to create an independent js file as a module. APIs for the module can be exposed only via module.exports or exports.

Note：

exports is a reference to module.exports. Therefore, a random modification to the point of exports in the module may cause an unknown error. Developers are advised to expose module APIs via module.exports if they do not know the relationship between them.

Currently, node_modules cannot be directly passed into Mini Programs. To use `node_modules`, developers are advised to copy relevant code to the directory of the Mini Program, or use the `npm` feature supported by Mini Programs.



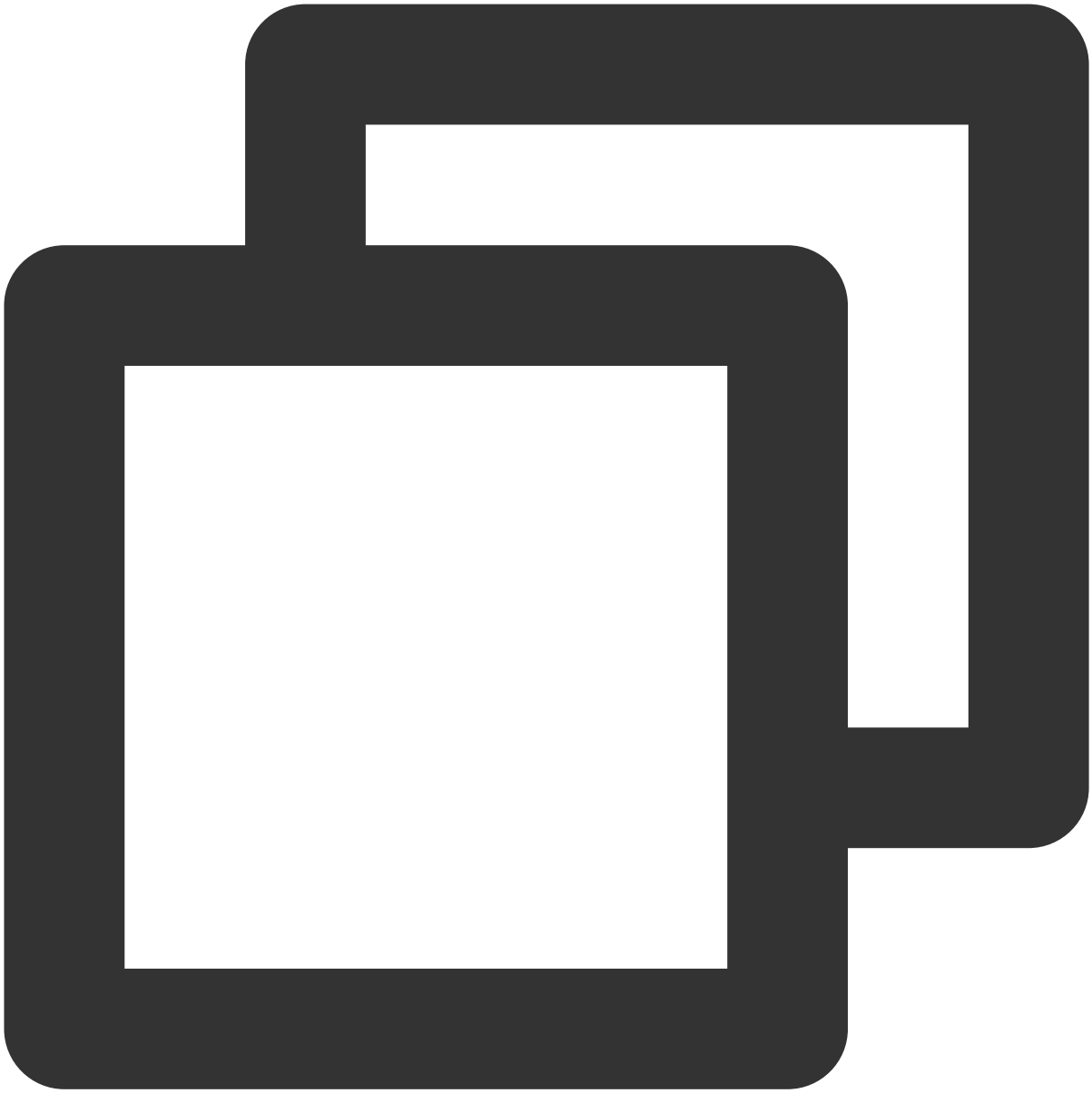
©2013-2022 Tencent Cloud. All rights reserved.

```
// common.js
function sayHello(name) {
  console.log(`Hello ${name} !`)
}
function sayGoodbye(name) {
  console.log(`Goodbye ${name} !`)
}

module.exports.sayHello = sayHello
exports.sayGoodbye = sayGoodbye
```

To pass common code into files that need to access these modules, use require.

©2013-2022 Tencent Cloud. All rights reserved.



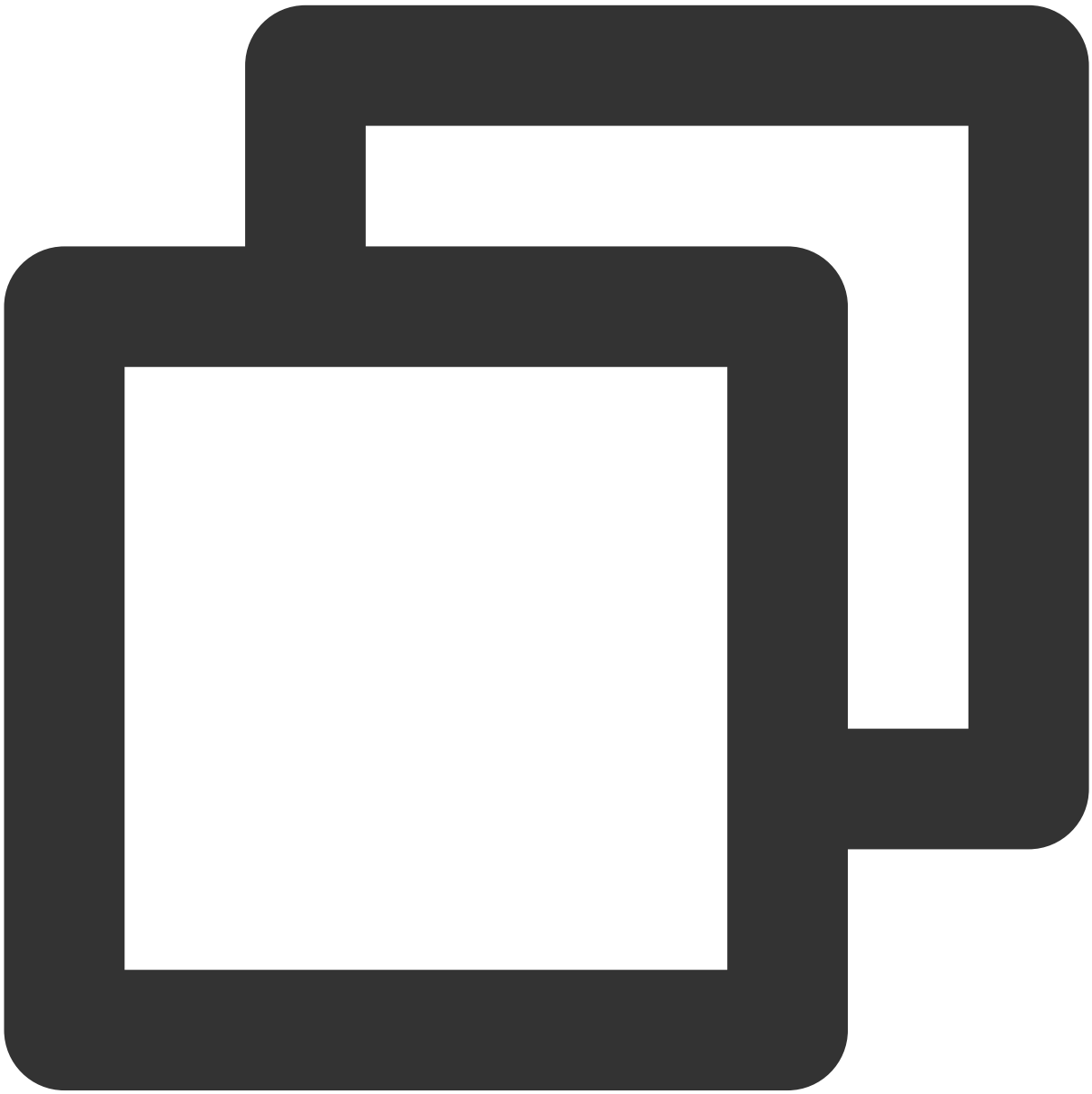
```
var common = require('common.js')
Page({
  helloMINA: function() {
    common.sayHello('MINA')
  },
  goodbyeMINA: function() {
    common.sayGoodbye('MINA')
  }
})
```

©2013-2022 Tencent Cloud. All rights reserved.

6.2 File Scope

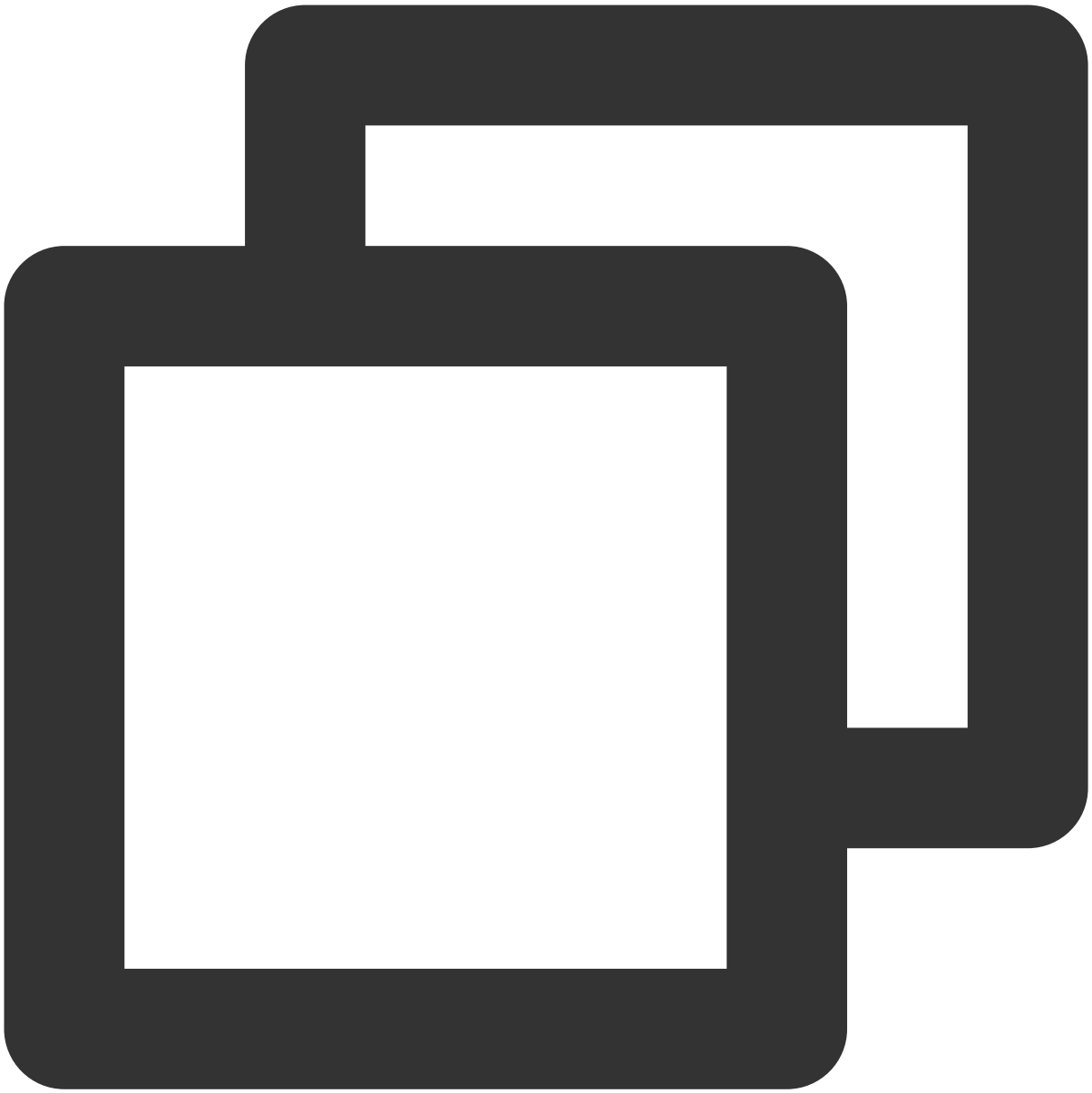
Variables and functions declared in the JavaScript file are valid only in the file. Variables and functions with the same names can be declared in different files without affecting each other.

The global application instance can be obtained via the global function `getApp` . To obtain the global data, you can set `App()` . For example:



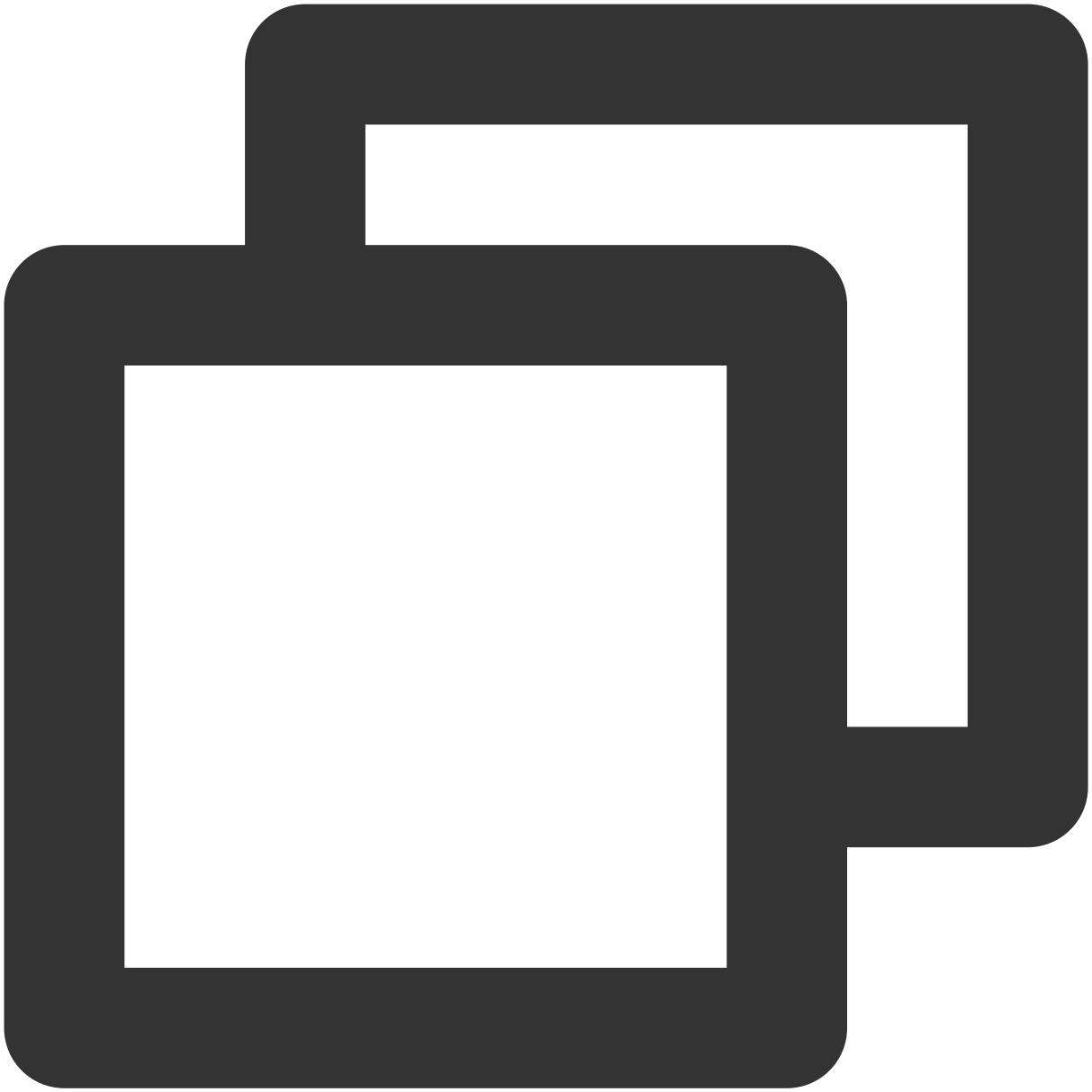
```
// app.js
App({
  globalData: 1
})
```

©2013-2022 Tencent Cloud. All rights reserved.



```
// a.js
// The localValue can only be used in file a.js.
var localValue = 'a'
// Get the app instance.
var app = getApp()
// Get the global data and change it.
app.globalData++
```

©2013-2022 Tencent Cloud. All rights reserved.



```
// a.js
// The localValue can only be used in file a.js.
var localValue = 'a'
// Get the app instance.
var app = getApp()
// Get the global data and change it.
app.globalData++
```

Note：

It's worth noting that require needs to use **relative paths** when introducing modules.

©2013-2022 Tencent Cloud. All rights reserved.

7. API

Mini Program development framework to provide rich micro-channel native API, you can easily tune up WeChat provides capabilities, such as access to user information, local storage, payment functions. Please refer to API Overview。

Usually, in Mini Programs API There are several types:

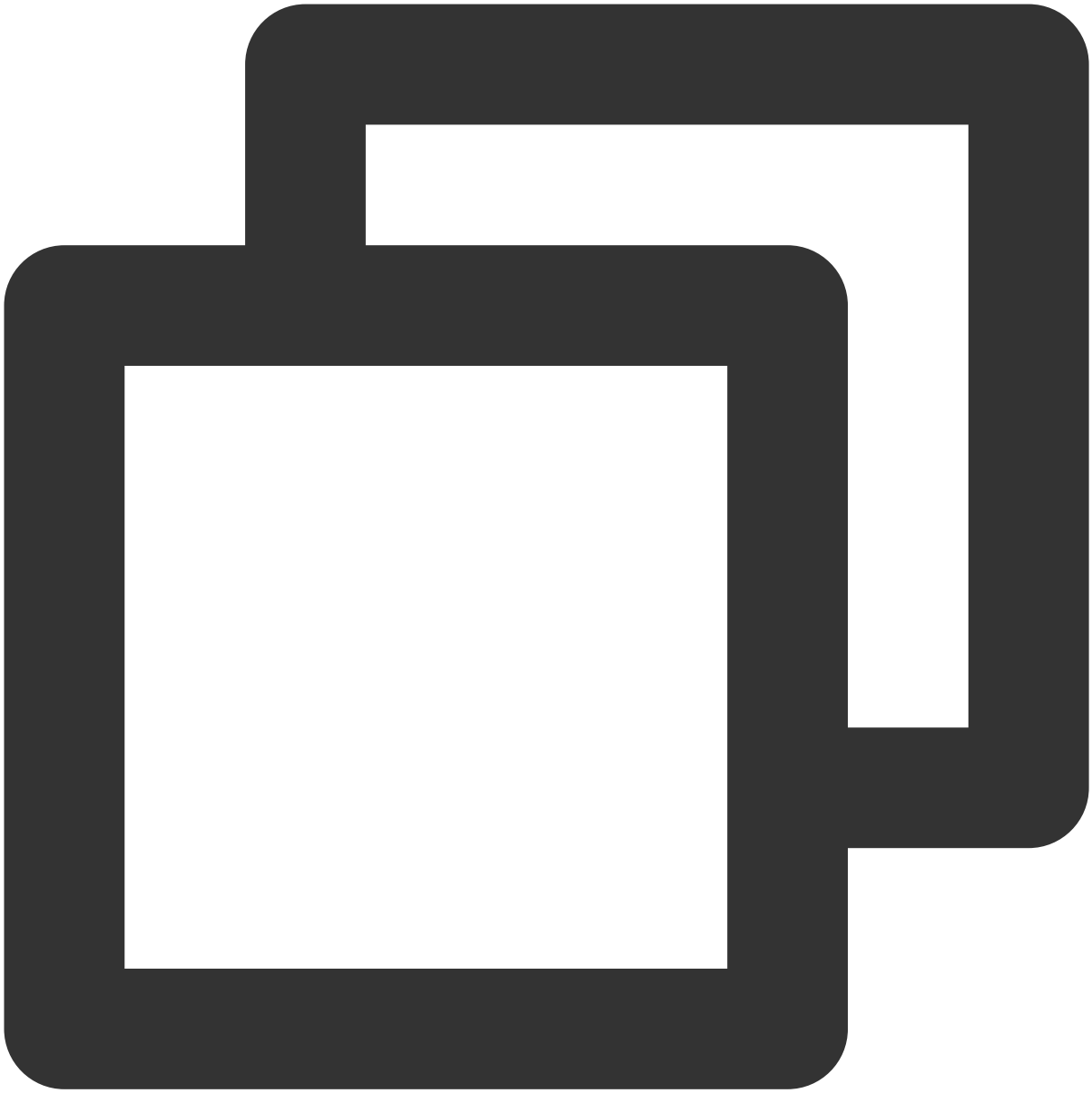
7.1 Event monitor API

We made a pact to on Initial API Used to listen if an event is triggered, such as:wx.onSocketOpen, wx.onCompassChange Wait.

this kind API Accepts a callback function as a parameter that is called when an event is triggered, and passes in the relevant data as parameters.

Code examples:

©2013-2022 Tencent Cloud. All rights reserved.



```
wx.onCompassChange(function (res) {
  console.log(res.direction)
})
```

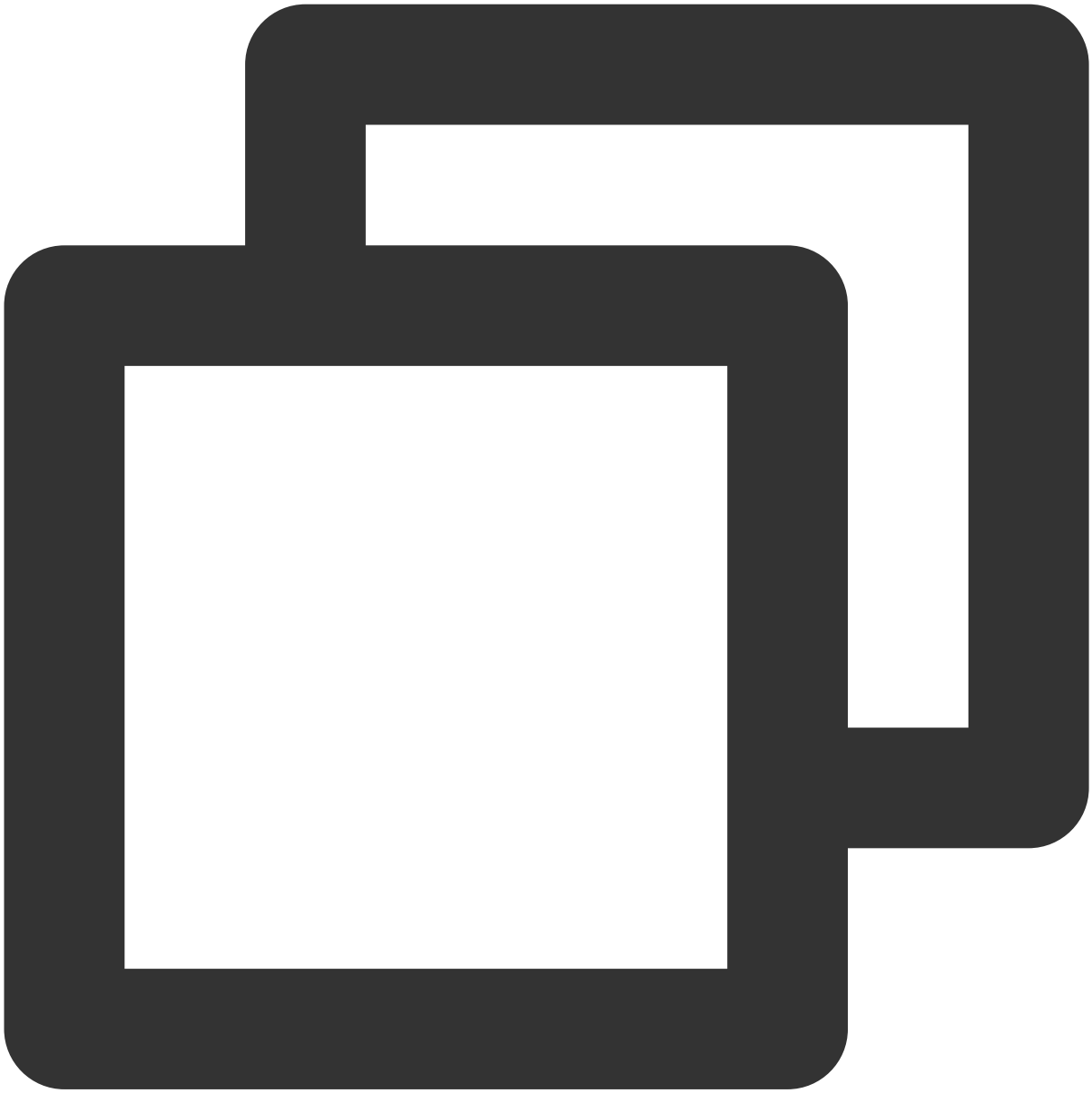
7.2 synchronization API

We made a pact to Sync Ending API It's all in sync. API, Such as wx.setStorageSync, wx.getSystemInfoSync Wait. In addition, there are some other synchronization API, such as wx.createWorker, wx.getBackgroundAudioManager For details, see API Instructions in the documentation.

©2013-2022 Tencent Cloud. All rights reserved.

synchronization API Can be obtained directly from the return value of the function, and an exception can be thrown if the execution goes wrong.

Code examples



```
try {
  wx.setStorageSync('key', 'value')
} catch (e) {
  console.error(e)
}
```

©2013-2022 Tencent Cloud. All rights reserved.

7.3 asynchronous API

Majority API It's all asynchronous. API, such as wx.request， wx.login Wait. this kind API Interfaces usually accept a Object Type, which supports specifying the following fields on demand to receive interface call results.

Object Parameter specification

Parameter name	type	Required	Introductions
success	function	no	Interface calls the successful callback function
fail	function	no	Interface calls failed callback functions
complete	function	no	Callback function at the end of an interface call (both successful and unsuccessful calls are executed)
Other	Any	-	Additional parameters defined by the interface

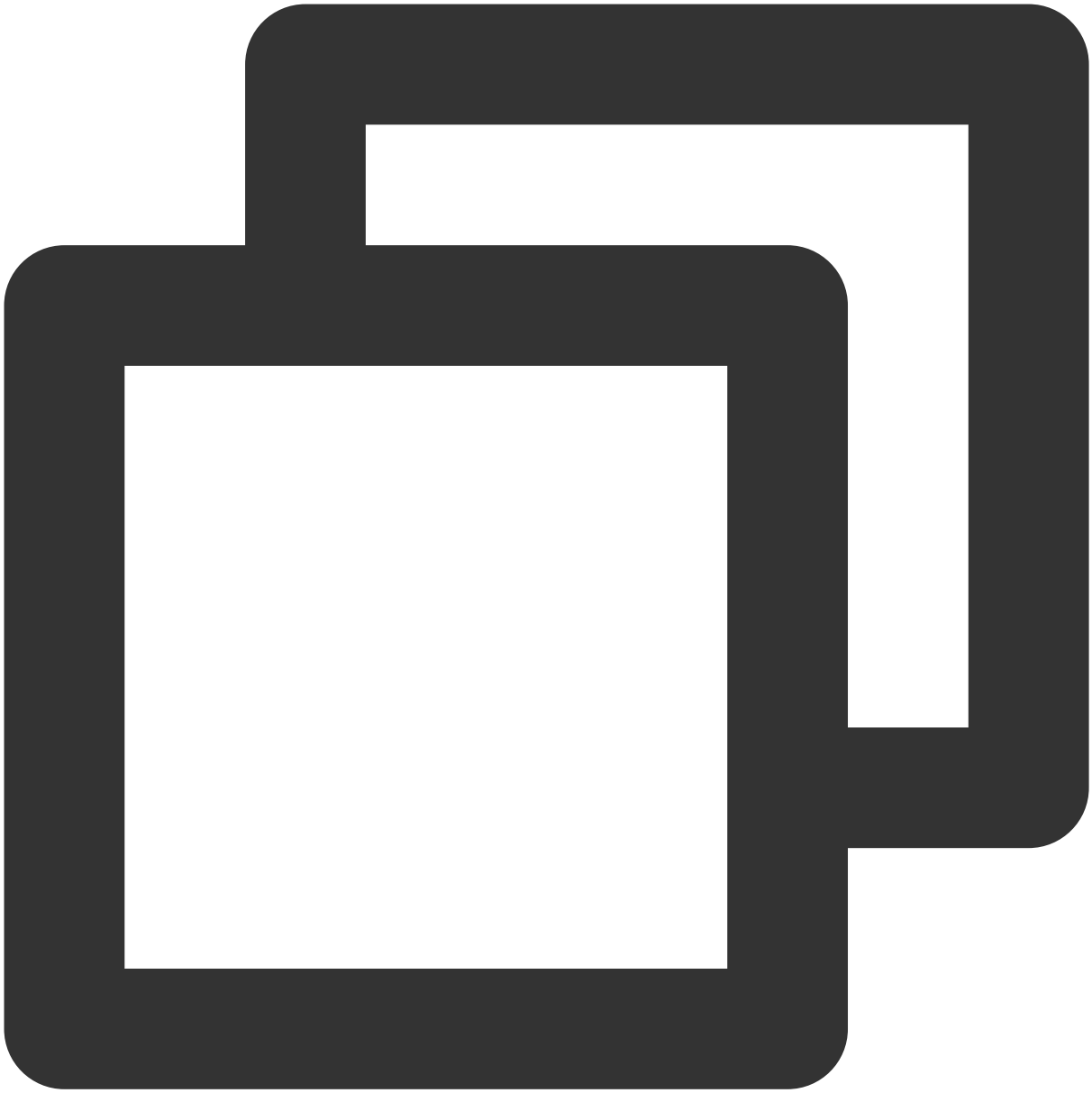
Parameters to the callback function

success， fail， complete When the function is called, it passes in a Object Type parameter with the following fields:

attribute	type	Introductions
errMsg	string	Error message if the call returns successfully ${apiName}:ok
errCode	number	Error code, partial only API Support, please refer to the corresponding meaning API Documentation, if successful 0。
Other	Any	Interface to return additional data

Code examples

©2013-2022 Tencent Cloud. All rights reserved.



```
wx.login({
  success(res) {
    console.log(res.code)
  }
})
```

©2013-2022 Tencent Cloud. All rights reserved.

View Layer

Last updated：2024-07-12 18:54:07

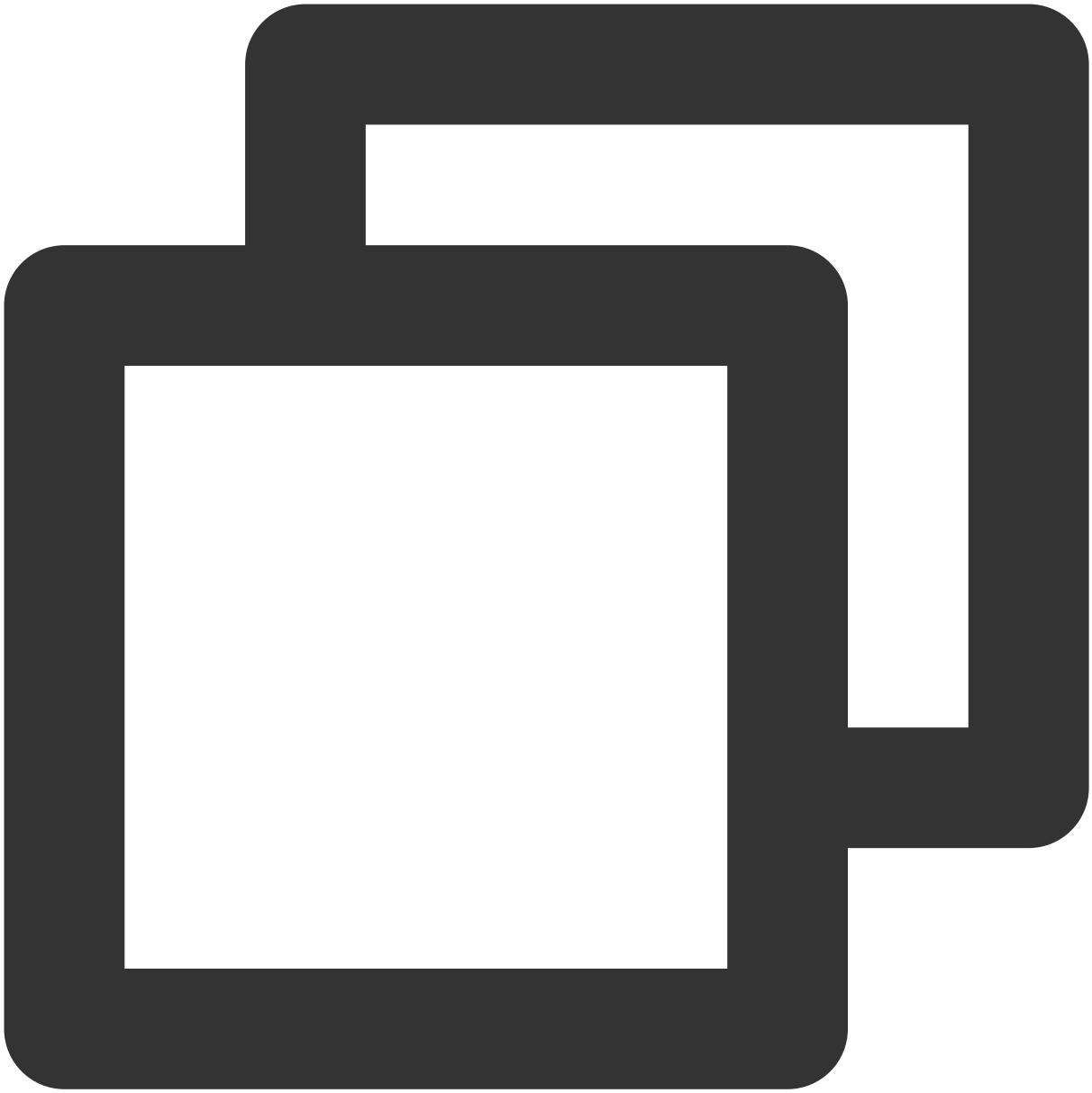
The view layer of the framework is written using WXML and WXSS and presented by components.
WXML (WeiXin Markup language) is used to describe the page structure;
WXS (WeiXin Script) is a scripting language for Mini Programs. Combined with WXML, it can be used to construct the page structure;
WXSS (WeiXin Style Sheet) is used to describe the page style.
Components are the building blocks of a view.

1. WXML

WXML (WeiXin Markup Language) is a markup language for framework design. It can be used to build the page structure when combined with base components and event system.
For more information on WXML syntax, see WXML syntax.
The following examples show what WXML can do:

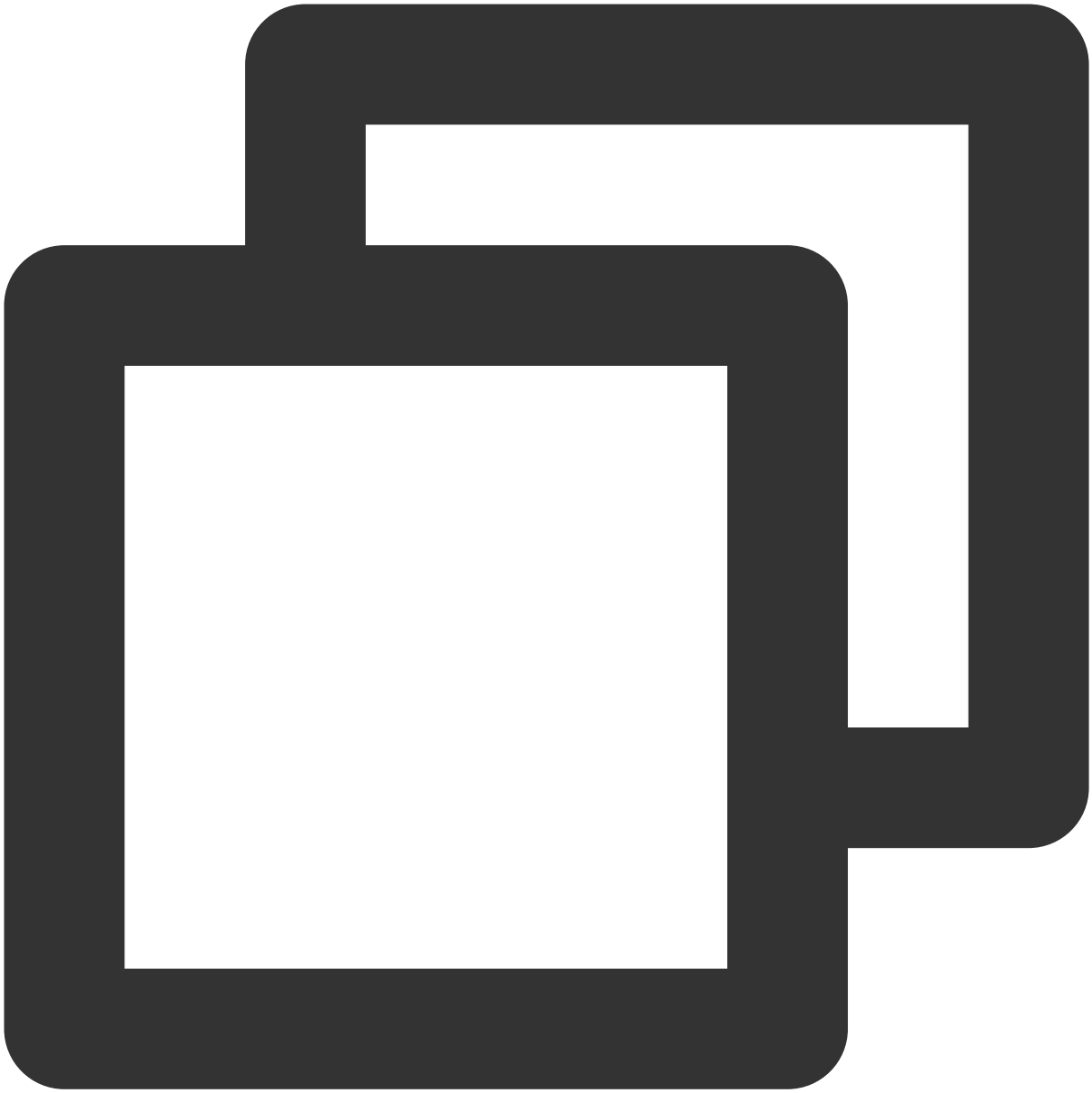
1.1 Data Binding

©2013-2022 Tencent Cloud. All rights reserved.



```
<!-- wxml -->
<view> {{message}} </view>
```

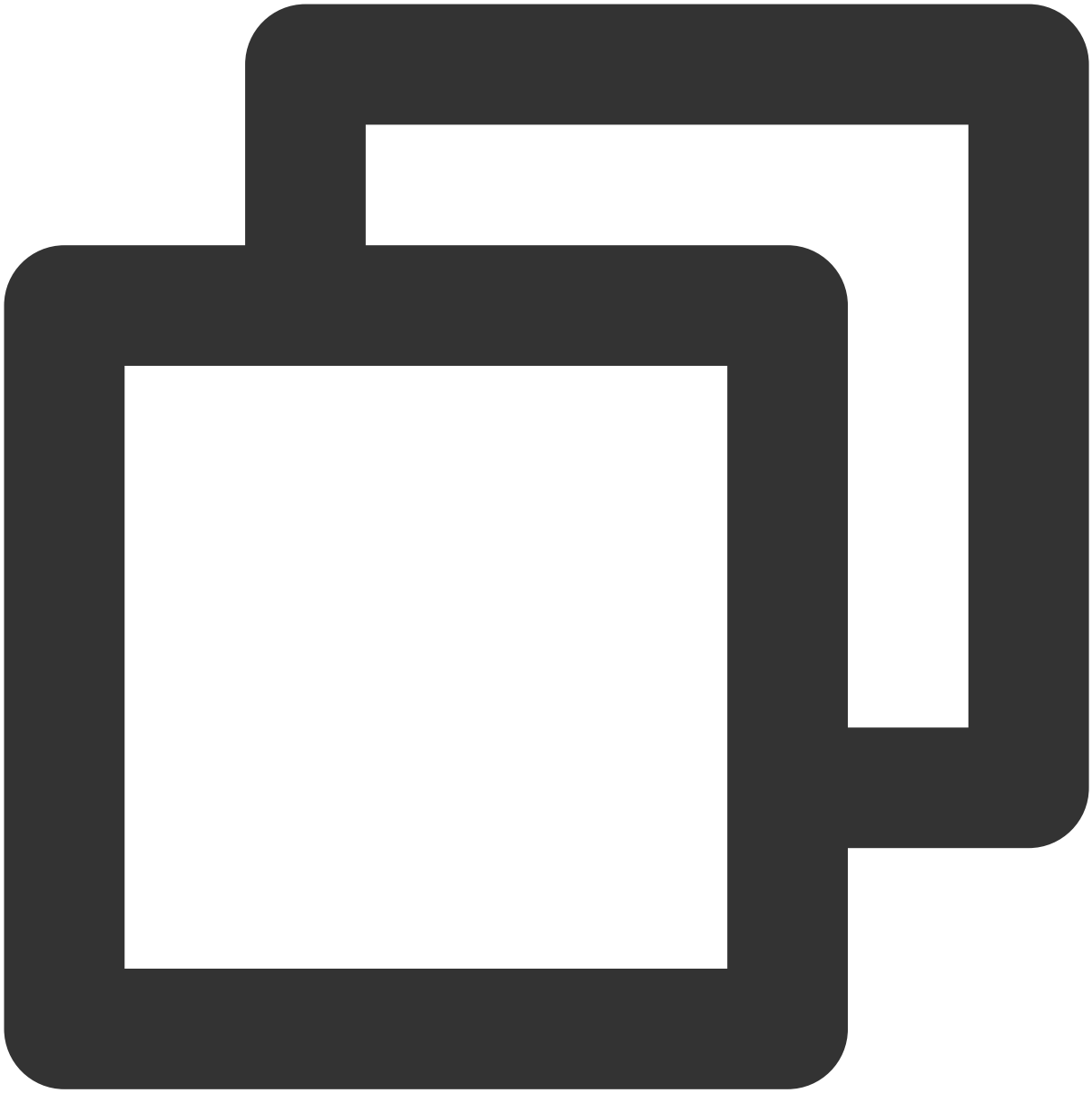
©2013-2022 Tencent Cloud. All rights reserved.



```
// page.js
Page({
  data: {
    message: 'Hello MINA!'
  }
})
```

1.2 List Rendering

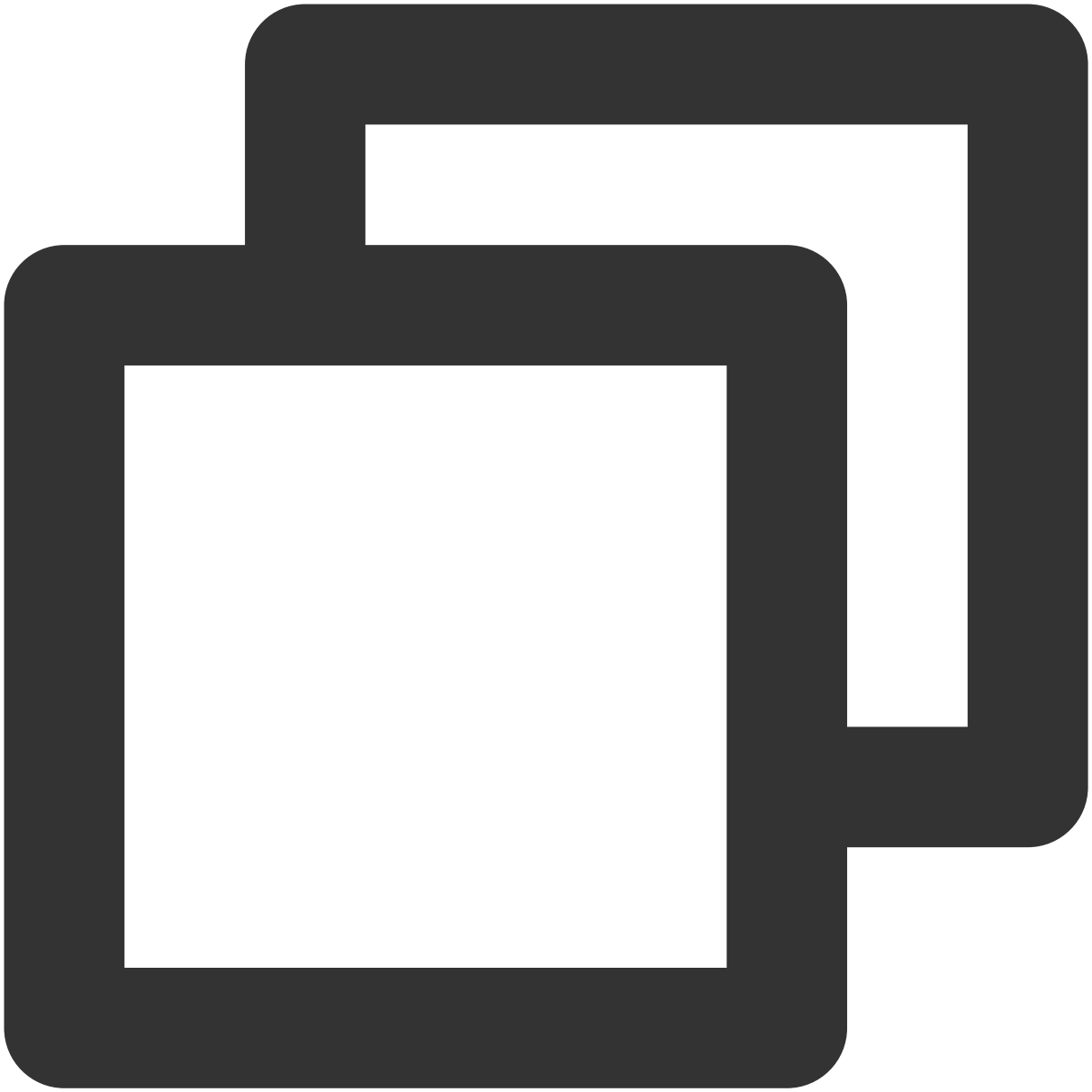
©2013-2022 Tencent Cloud. All rights reserved.



```
<!--wxml-->
<view wx:for="{{array}}"> {{item}} </view>
```

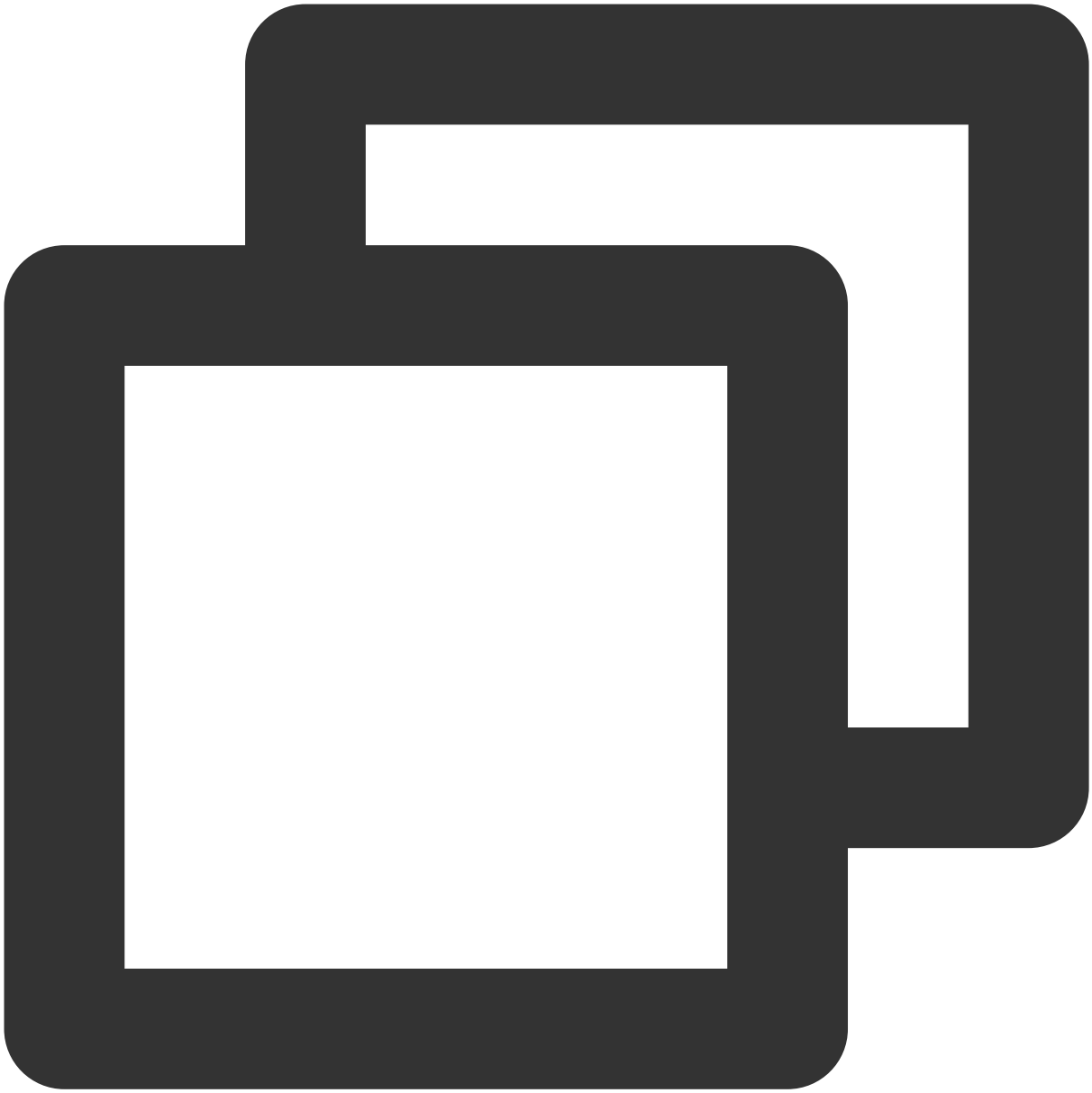
1.3 Condition Rendering

©2013-2022 Tencent Cloud. All rights reserved.



```
<!--wxml-->
<view wx:if="{{view == 'WEBVIEW'}}"> WEBVIEW </view>
<view wx:elif="{{view == 'APP'}}"> APP </view>
<view wx:else="{{view == 'MINA'}}"> MINA </view>
```

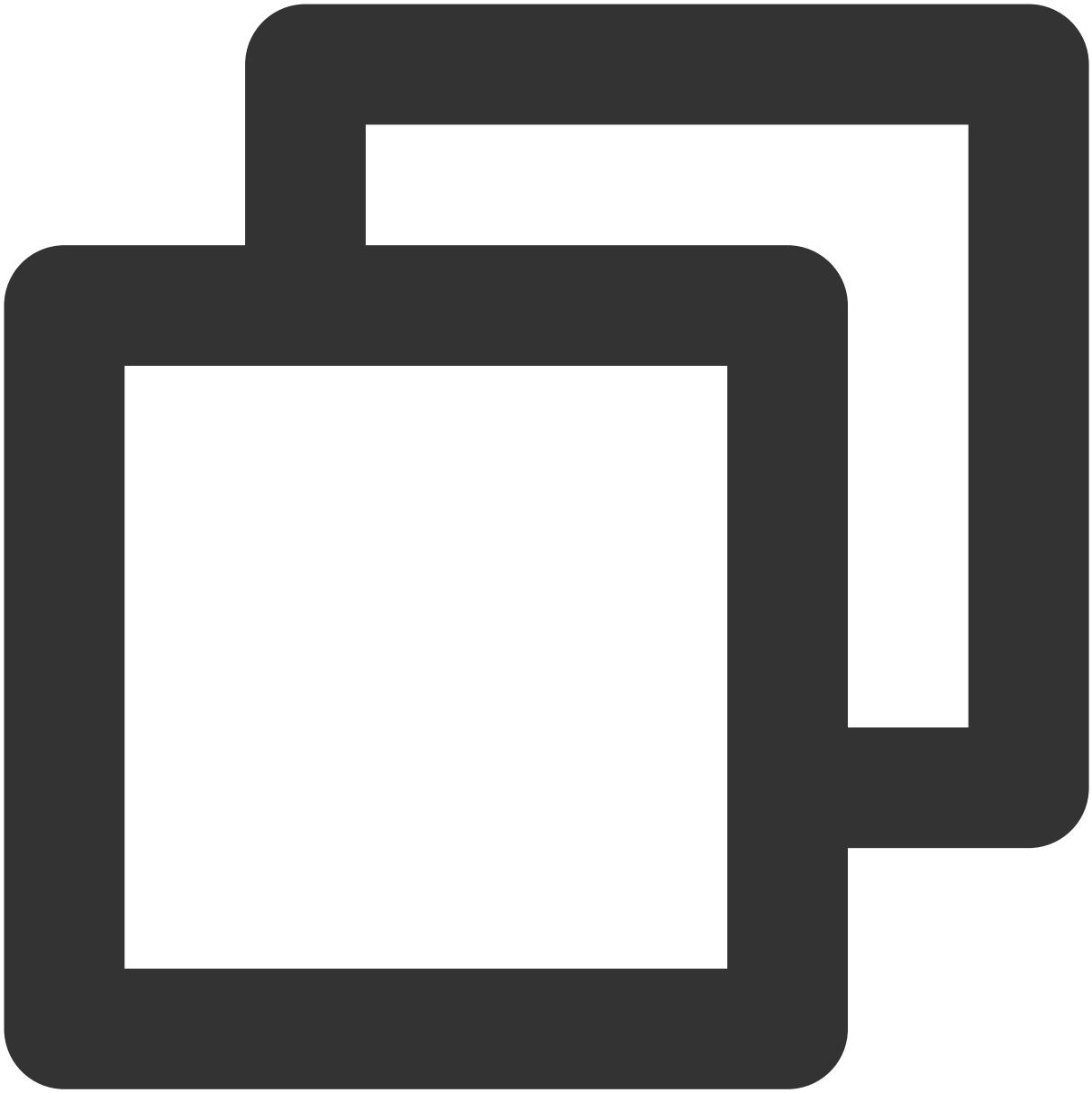
©2013-2022 Tencent Cloud. All rights reserved.



```
// page.js
Page({
  data: {
    view: 'MINA'
  }
})
```

1.4 Template

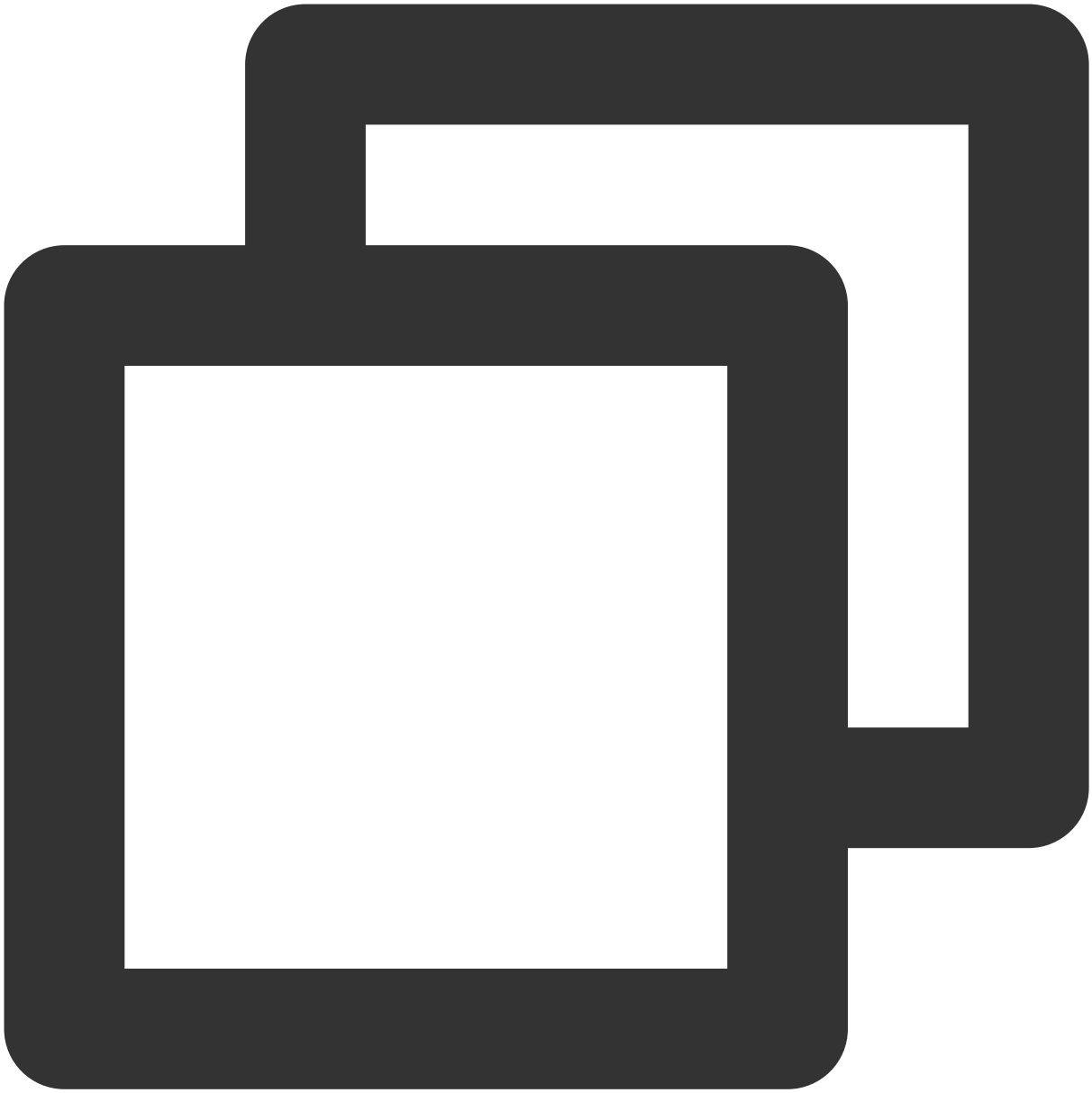
©2013-2022 Tencent Cloud. All rights reserved.



```
<!--wxml-->
<template name="staffName">
  <view>
    FirstName: {{firstName}}, LastName: {{lastName}}
  </view>
</template>

<template is="staffName" data="{{...staffA}}"></template>
<template is="staffName" data="{{...staffB}}"></template>
<template is="staffName" data="{{...staffC}}"></template>
```

©2013-2022 Tencent Cloud. All rights reserved.



```
// page.js
Page({
  data: {
    staffA: {firstName: 'Hulk', lastName: 'Hu'},
    staffB: {firstName: 'Shang', lastName: 'You'},
    staffC: {firstName: 'Gideon', lastName: 'Lin'}
  }
})
```

©2013-2022 Tencent Cloud. All rights reserved.

2. WXSS

WXSS (WeiXin Style Sheets) is a set of style languages that describe WXML component styles,
and is used to determine how WXML components are displayed.
WXSS has most of the features of CSS to cater for the needs of front-end developers, but incorporates some new features and modifications to adapt to the development of Weixin Mini Programs.
Compared with CSS, WXSS offers the following additional features:
Dimension unit
Style import;
Dimension Unit.

Note：

For iOS 8 and below, if you use flexbox layout, you need to add `display: -webkit-flex` property.

2.1 Dimension Unit

rpx (responsive pixel): Adaptable to the screen width. The specified screen width is 750 rpx. If the screen width on iPhone6 is 375 px (750 physical pixels), then 750 rpx = 375 px = 750 physical pixels, i.e. 1 rpx = 0.5 px = 1 physical pixel.

Device	rpx-to-px conversion (screen width/750)	px-to-rpx conversion (750/screen width)
iPhone5	1rpx = 0.42px	1px = 2.34rpx
iPhone6	1rpx = 0.5px	1px = 2rpx
iPhone6 Plus	1rpx = 0.552px	1px = 1.81rpx

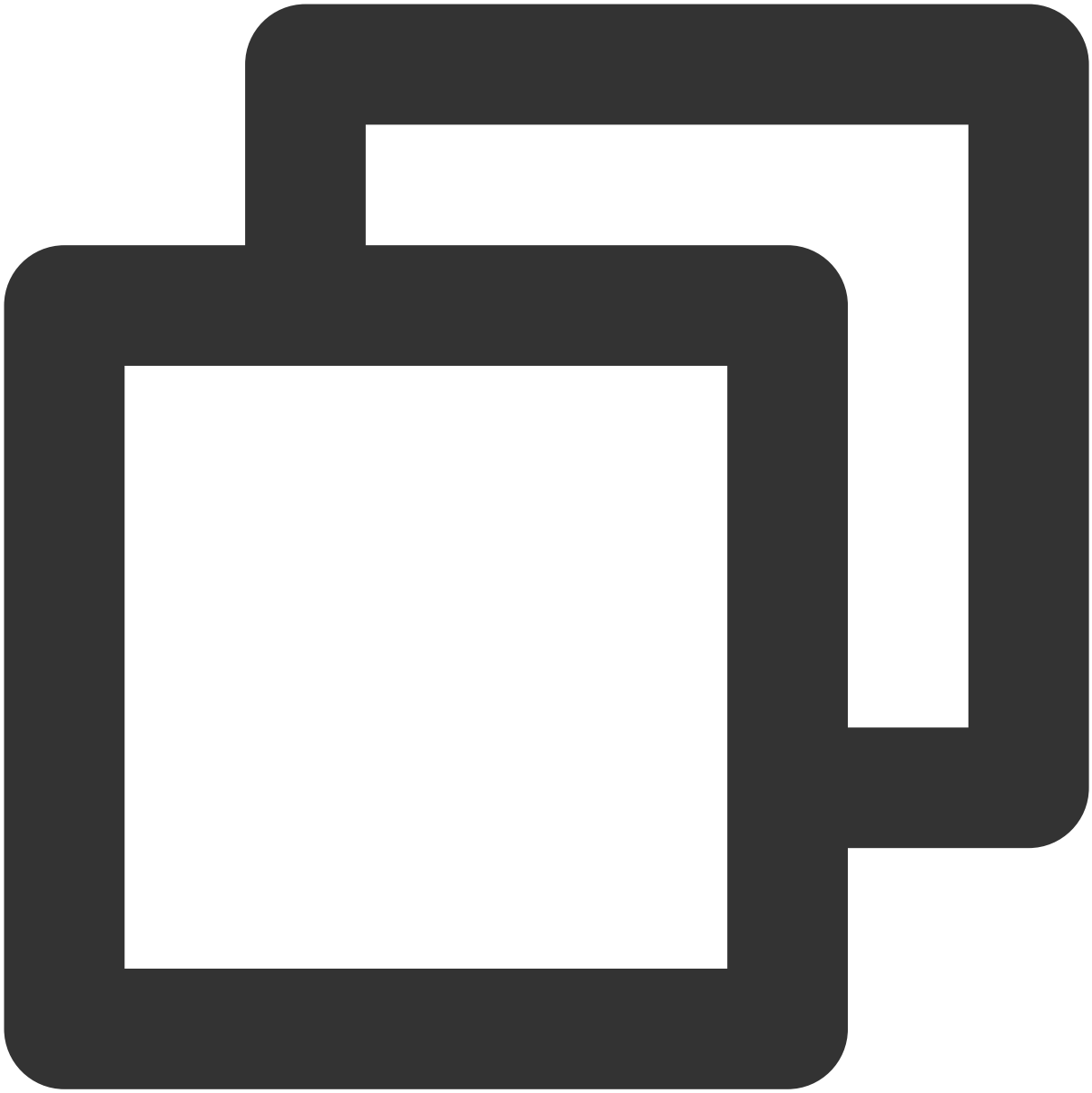
Note：

Tip: Designers can use iPhone6 as a standard in Visual Design when developing Weixin Mini Programs.
Note: It is inevitable to have some glitches on small screens. Please try to avoid this during development.

2.2 Style Import

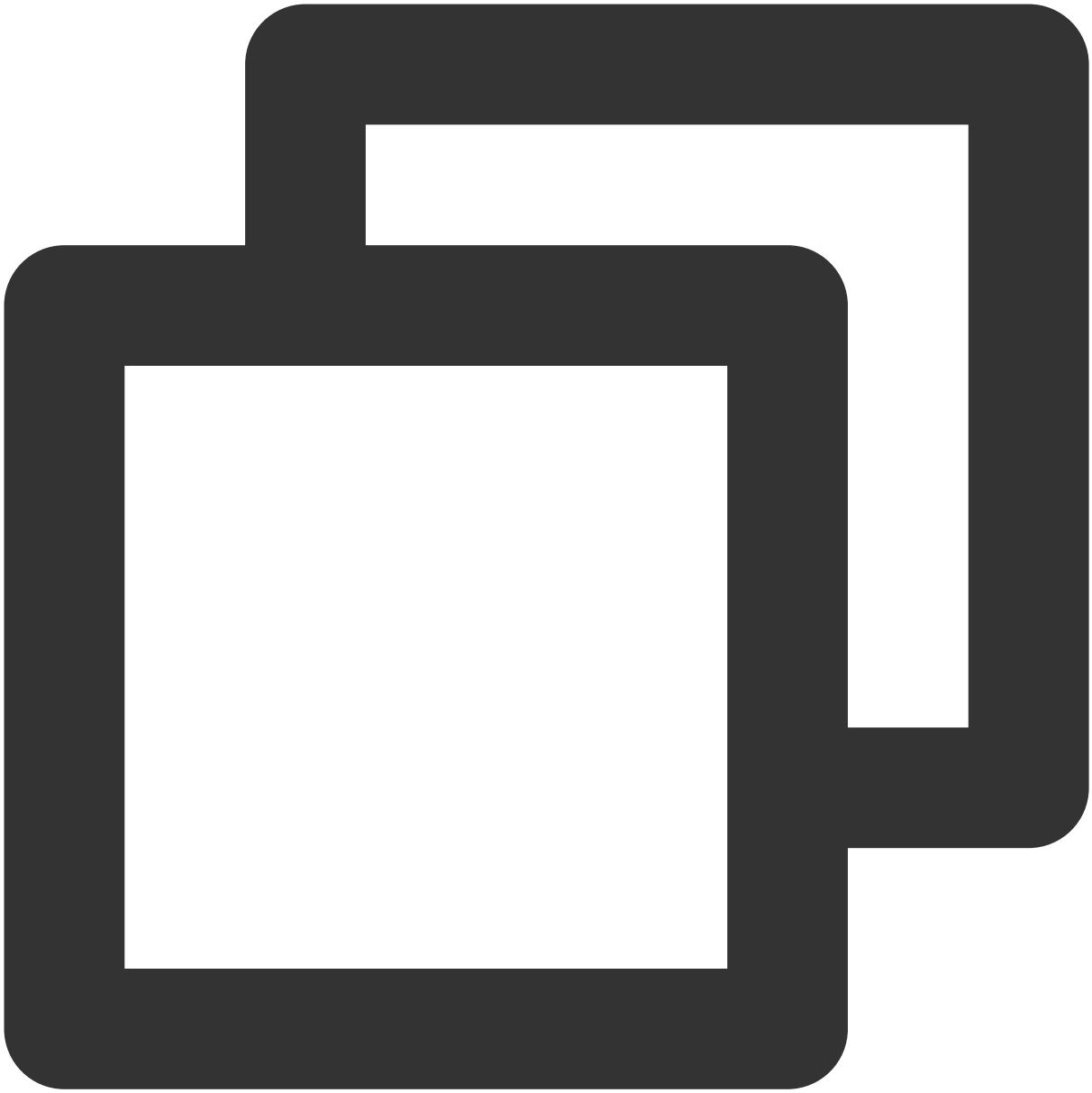
You can use the `@import` statement to import external style sheet. The `@import` statement should be followed by the relative path of the external style sheet and ends with ;.

©2013-2022 Tencent Cloud. All rights reserved.



```
/** common.wxss **/
.small-p {
  padding:5px;
}
```

©2013-2022 Tencent Cloud. All rights reserved.



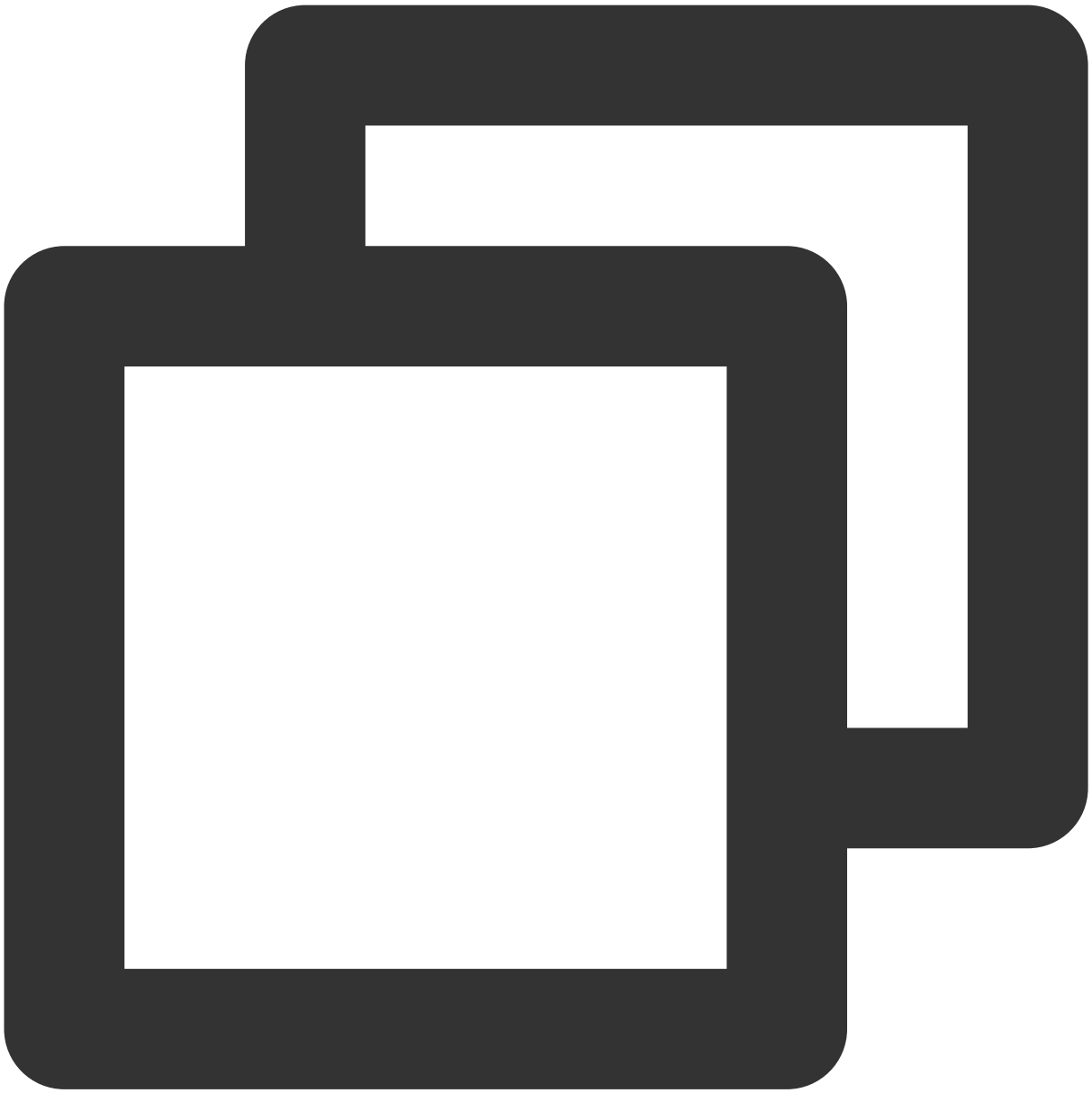
```
/** app.wxss **/
@import "common.wxss";
.middle-p {
  padding:15px;
}
```

2.3 Inline Style

You can use "style" and "class" properties for framework components to control the component style.

©2013-2022 Tencent Cloud. All rights reserved.

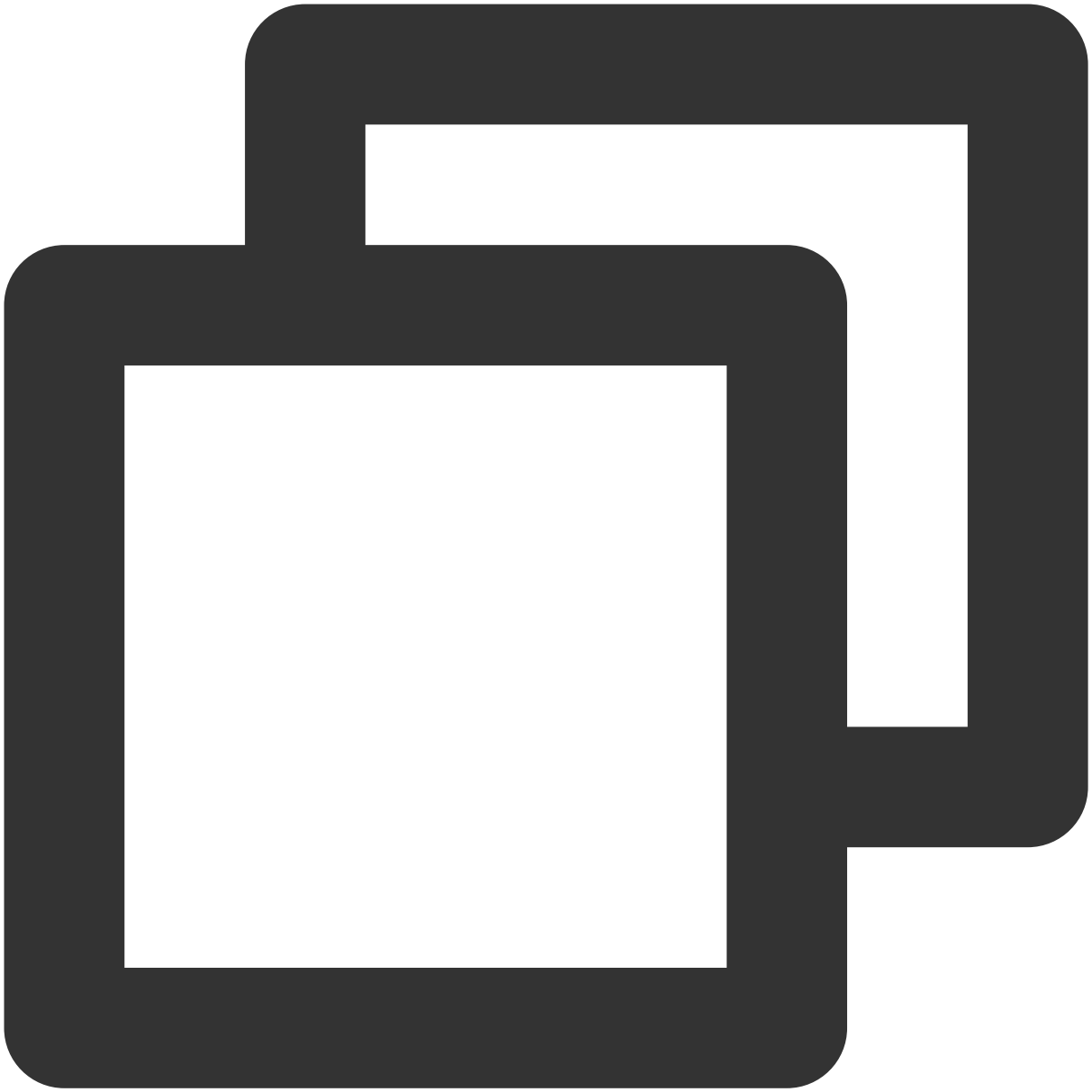
style: Static styles are all written to "class". "style" receives dynamic styles and parses them at runtime. It is not recommended to write static styles to "style" to avoid affecting the rendering speed.



```
/** common.wxss **/
.small-p {
  padding:5px;
}
```

class: Used to specify style rules. Its value is a collection of class selector names (style class names) in the style rules. The style class names do not contain . and are separated with spaces.

©2013-2022 Tencent Cloud. All rights reserved.



```
<view class="normal_view" />
```

2.4 Selectors

The following selectors are supported:

Selector	Example	Description
.class	.intro	Selects all components with class="intro"
#id	#firstname	Selects the component with id="firstname"

©2013-2022 Tencent Cloud. All rights reserved.

element	view	Selects all "view" components
element, element	view, checkbox	Selects all "checkbox" components and the "view" components of all documents
::after	view::after	Inserts content after the "view" component
::before	view::before	Inserts content before the "view" component

2.5 Global and Local Styles

A global style is defined in app.wxss and applies to every page, while a local one is defined in the page's wxss file and only applies to the current page (overrides the same selector in app.wxss).

3. WXS

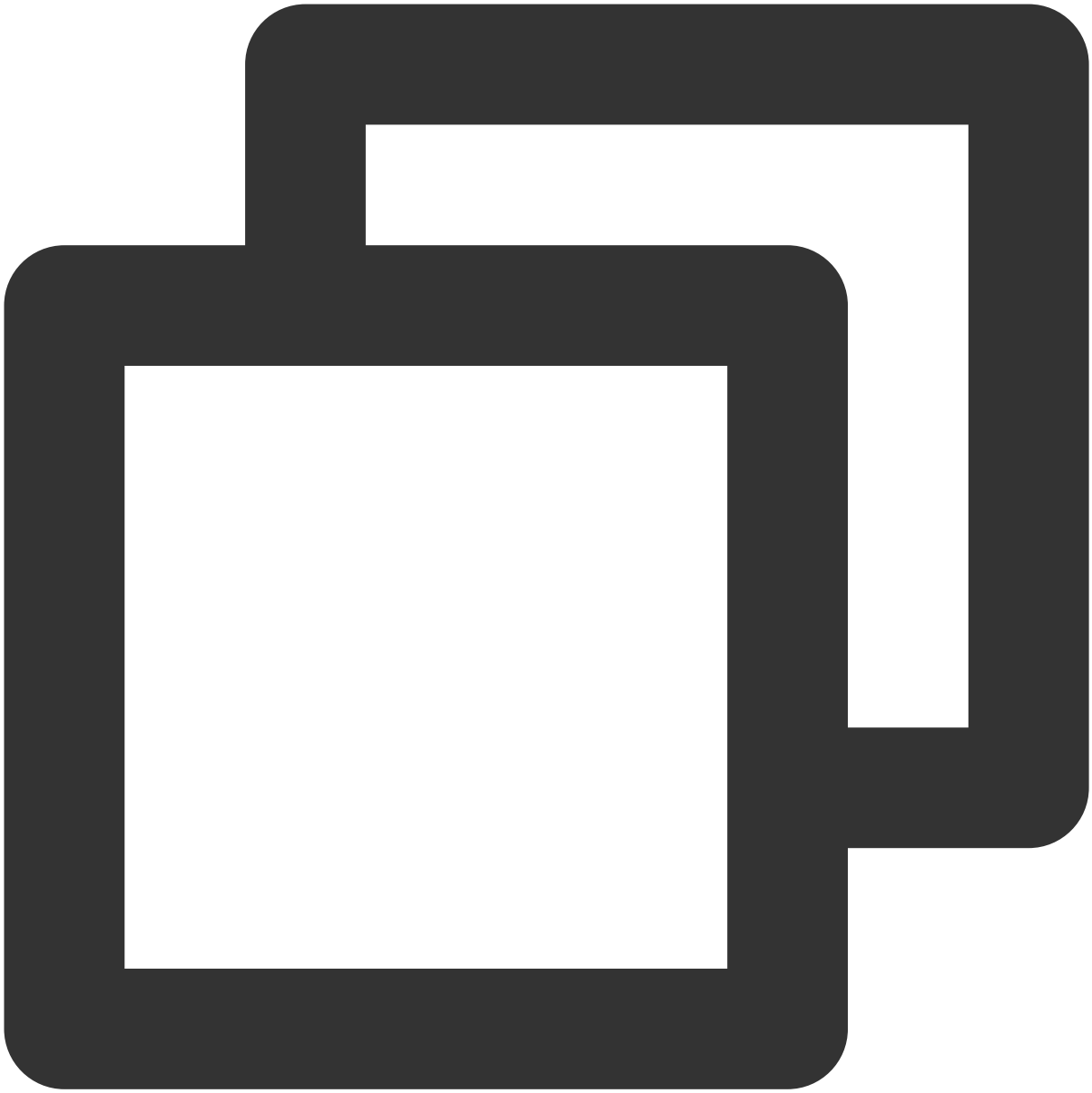
WXS (WeiXin Script) is a script segment inlined in WXML. WXS can be used to inline a small number of processing scripts in the template to enrich the data preprocessing capability of the template. In addition, WXS can also be used to write simple WXS event response functions.

Syntactically, WXS is similar to JavaScript with a few limitations. For a complete overview of the WXS syntax, see WXS - Introduction.

Here are some simple examples of using WXS.

Page Rendering

©2013-2022 Tencent Cloud. All rights reserved.



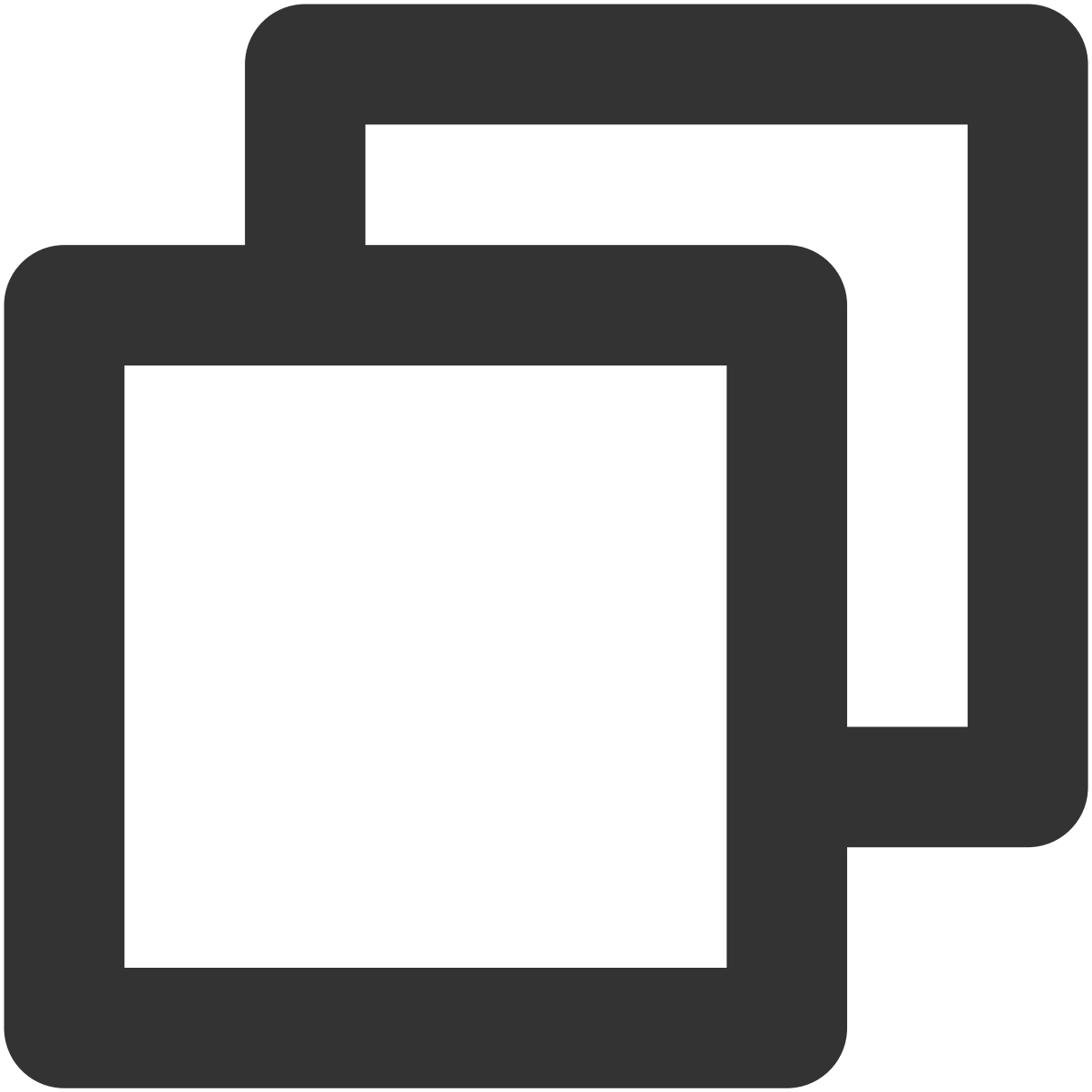
```
<!--wxml-->
<wxs module="m1">
var msg = "hello world";

module.exports.message = msg;
</wxs>

<view> {{m1.message}} </view>
```

Page output:

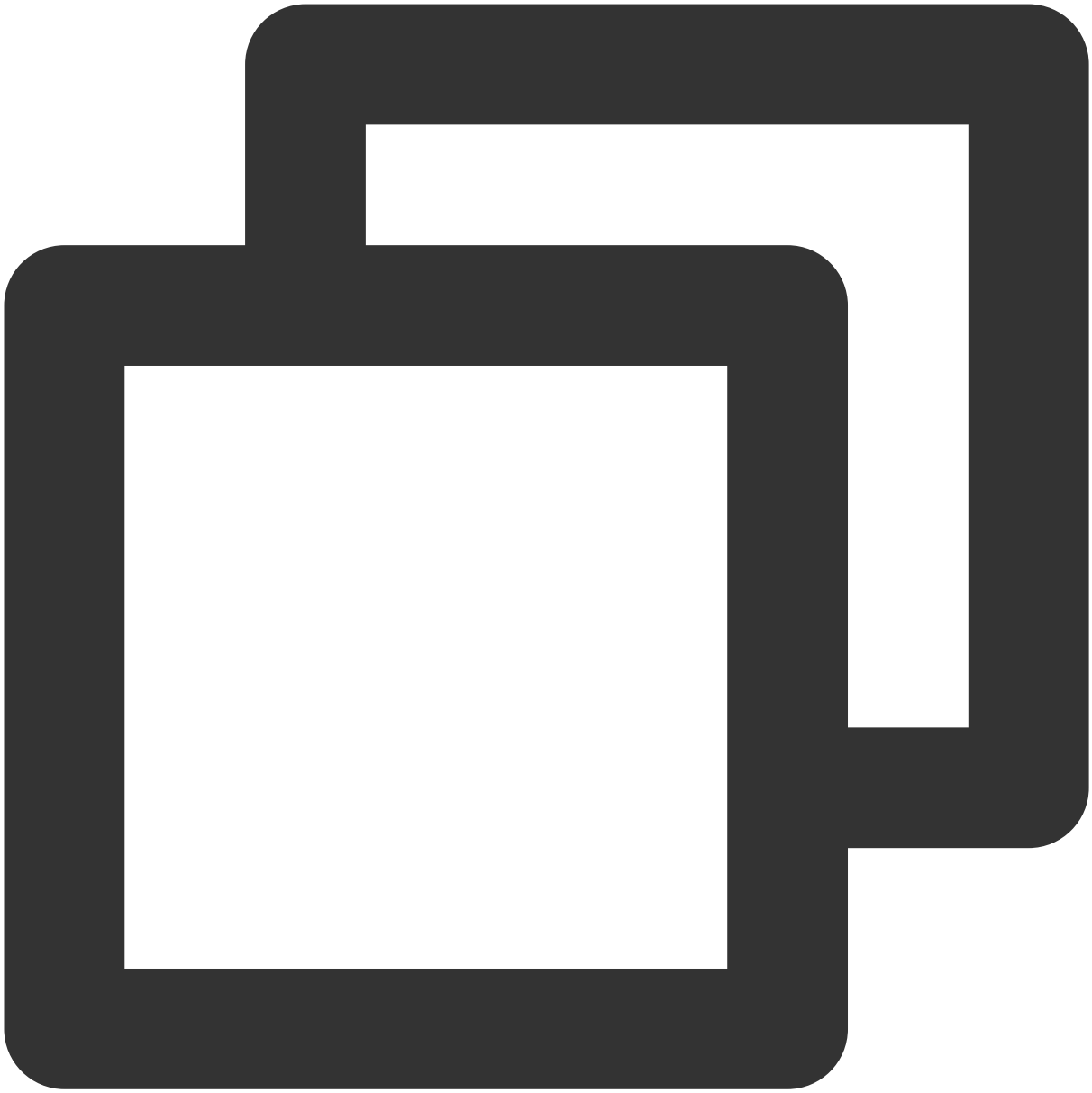
©2013-2022 Tencent Cloud. All rights reserved.



```
hello world
```

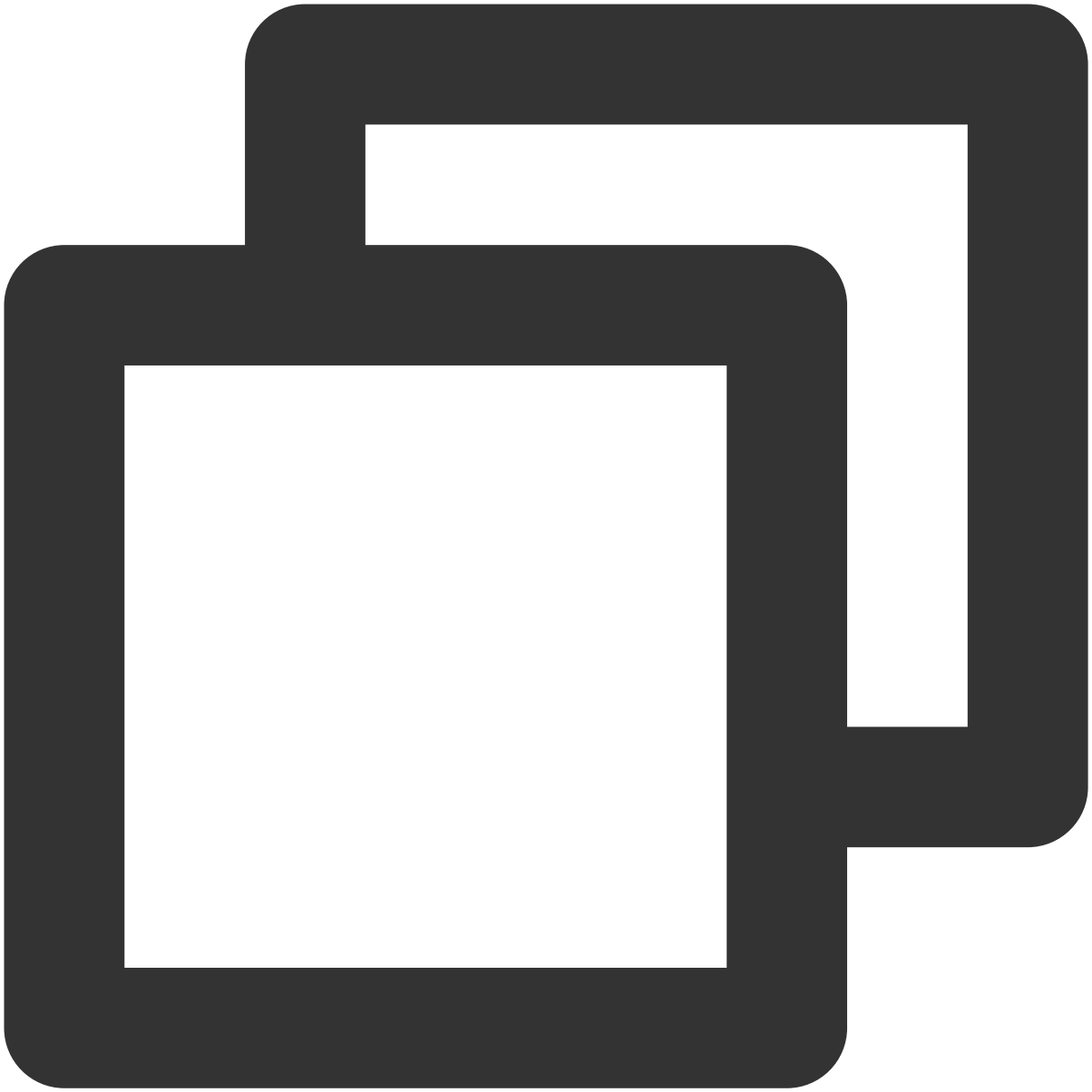
Data Processing

©2013-2022 Tencent Cloud. All rights reserved.



```
// page.js
Page({
  data: {
    array: [1, 2, 3, 4, 5, 1, 2, 3, 4]
  }
})
```

©2013-2022 Tencent Cloud. All rights reserved.



```
<!--wxml-->
<!-- The following getMax function takes an array and returns the value of the larg
<wxs module="m1">
var getMax = function(array) {
  var max = undefined;
  for (var i = 0; i < array.length; ++i) {
    max = max === undefined ?
      array[i] :
      (max >= array[i] ? max : array[i]);
  }
  return max;
```

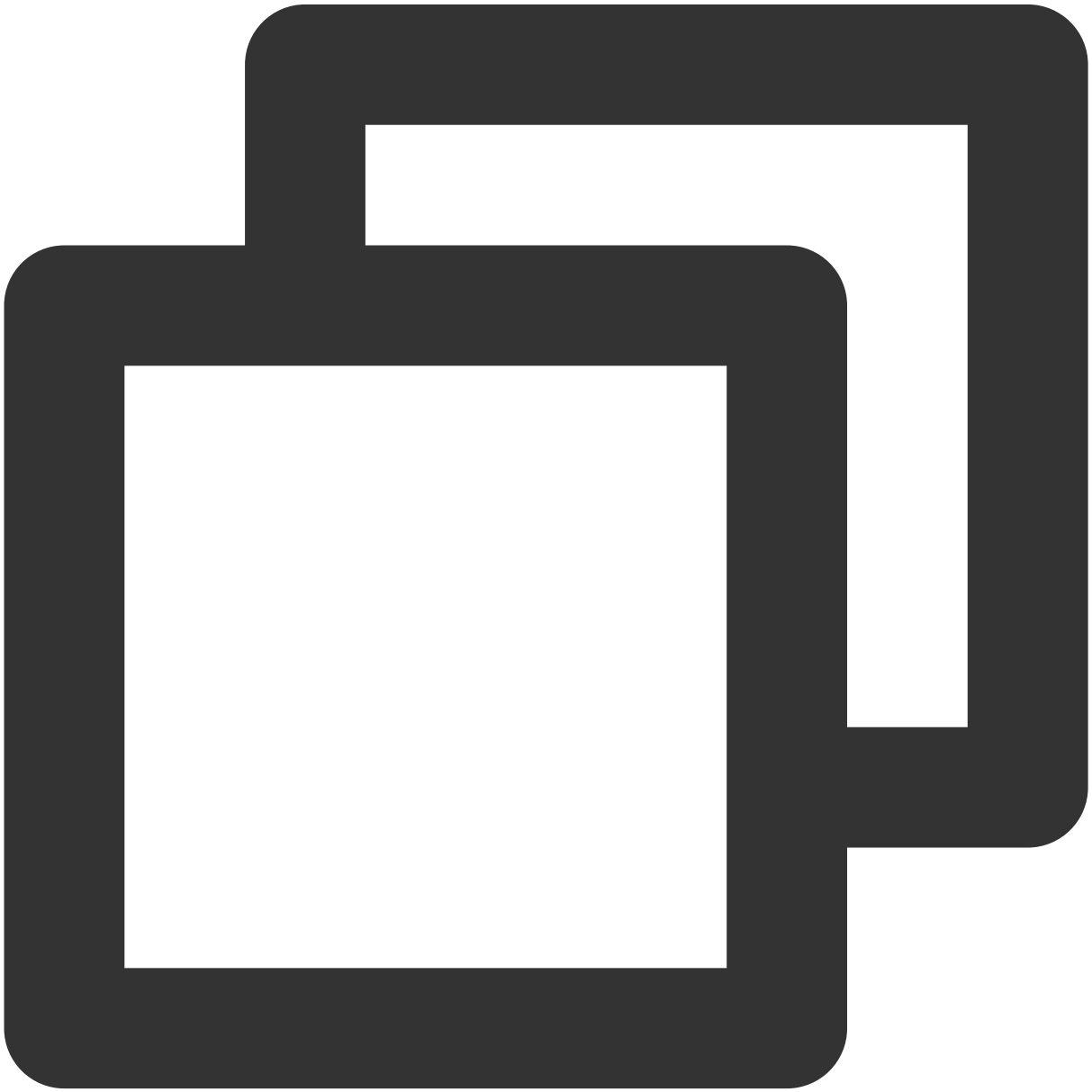
©2013-2022 Tencent Cloud. All rights reserved.

```
}

module.exports.getMax = getMax;
</wxs>

<!-- Calls the getMax function in WXS with the "array" in page.js as the parameter
<view> {{m1.getMax(array)}} </view>
```

Page output:



```
5
```

©2013-2022 Tencent Cloud. All rights reserved.

4. Event System

4.1 What Is an Event?

An event is a channel through which the view layer communicants with the logical layer;

An event can pass user's action on to the logical layer for processing;

An event can be bound to a component. When the event is triggered, the corresponding event handler is executed at the logic layer;

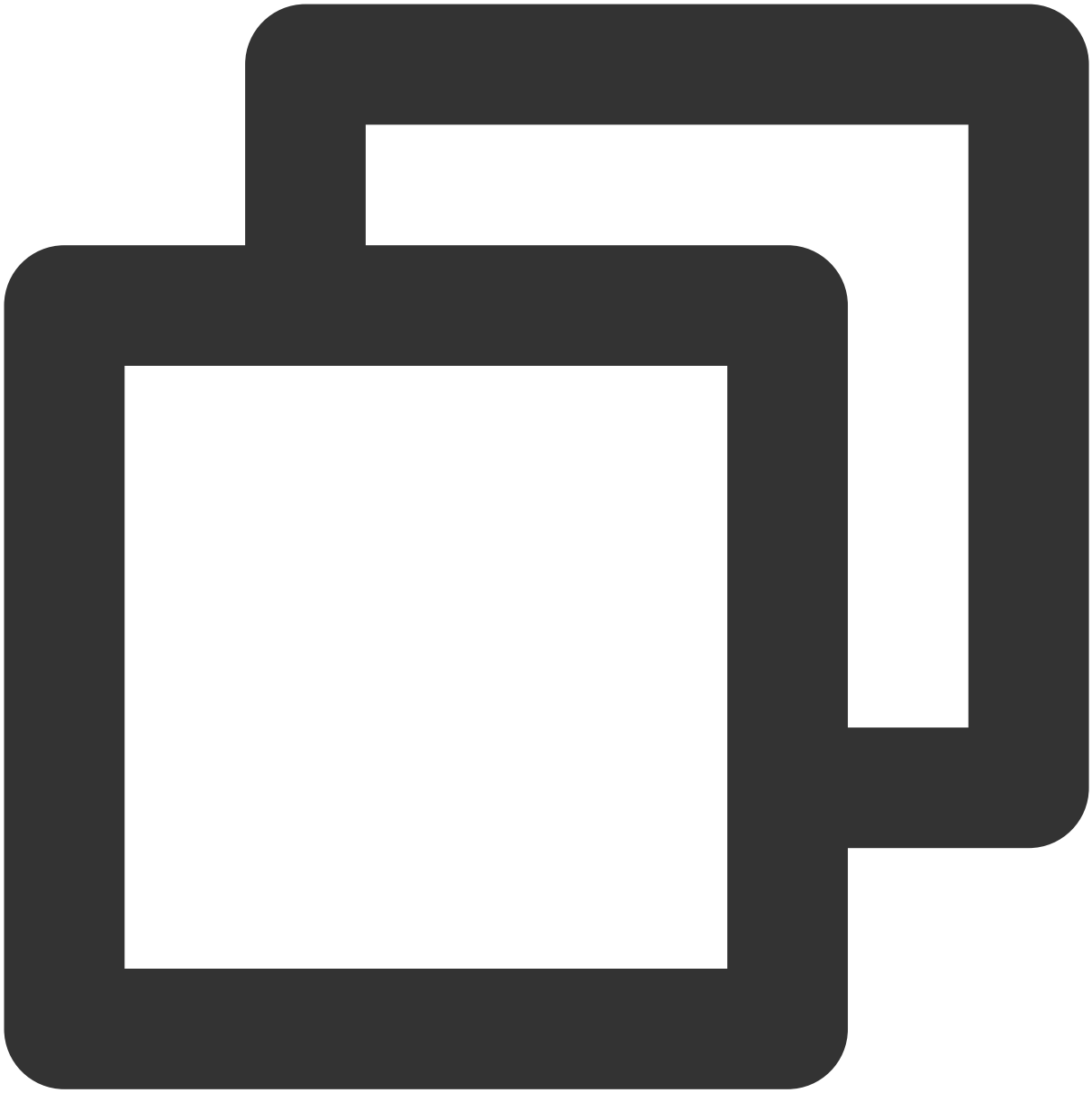
An event object can have additional information such as ID, dataset, touches.

4.2 How to Use an Event

Bind an event handler to the component.

For example, if the event is bindtap, the event handler can be found in the page when a user taps the component.

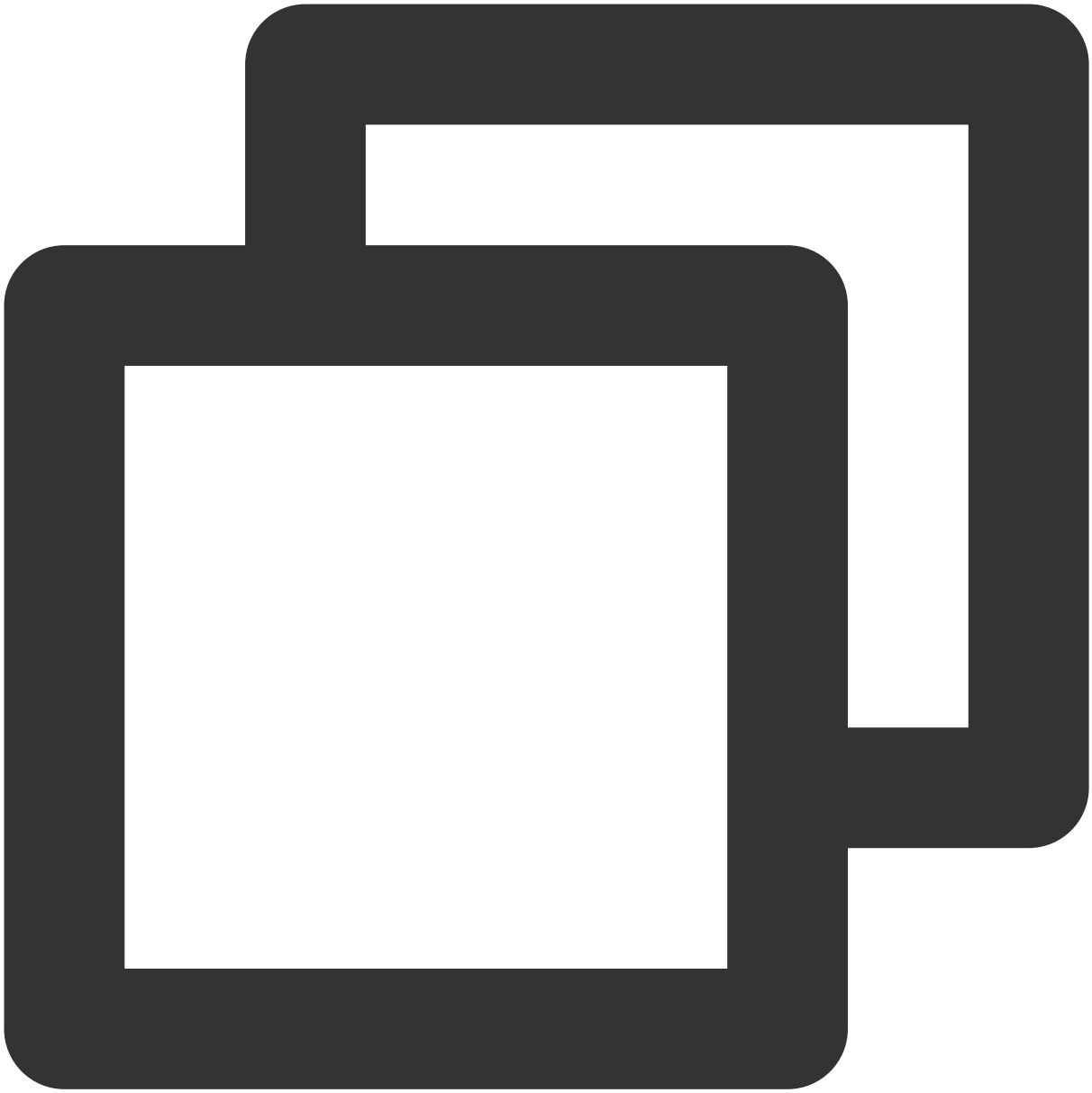
©2013-2022 Tencent Cloud. All rights reserved.



```
<view id="tapTest" data-hi="Weixin" bindtap="tapName"> Click me! </view>
```

Write the event handler in the Page definition with " `event` " as the parameter.

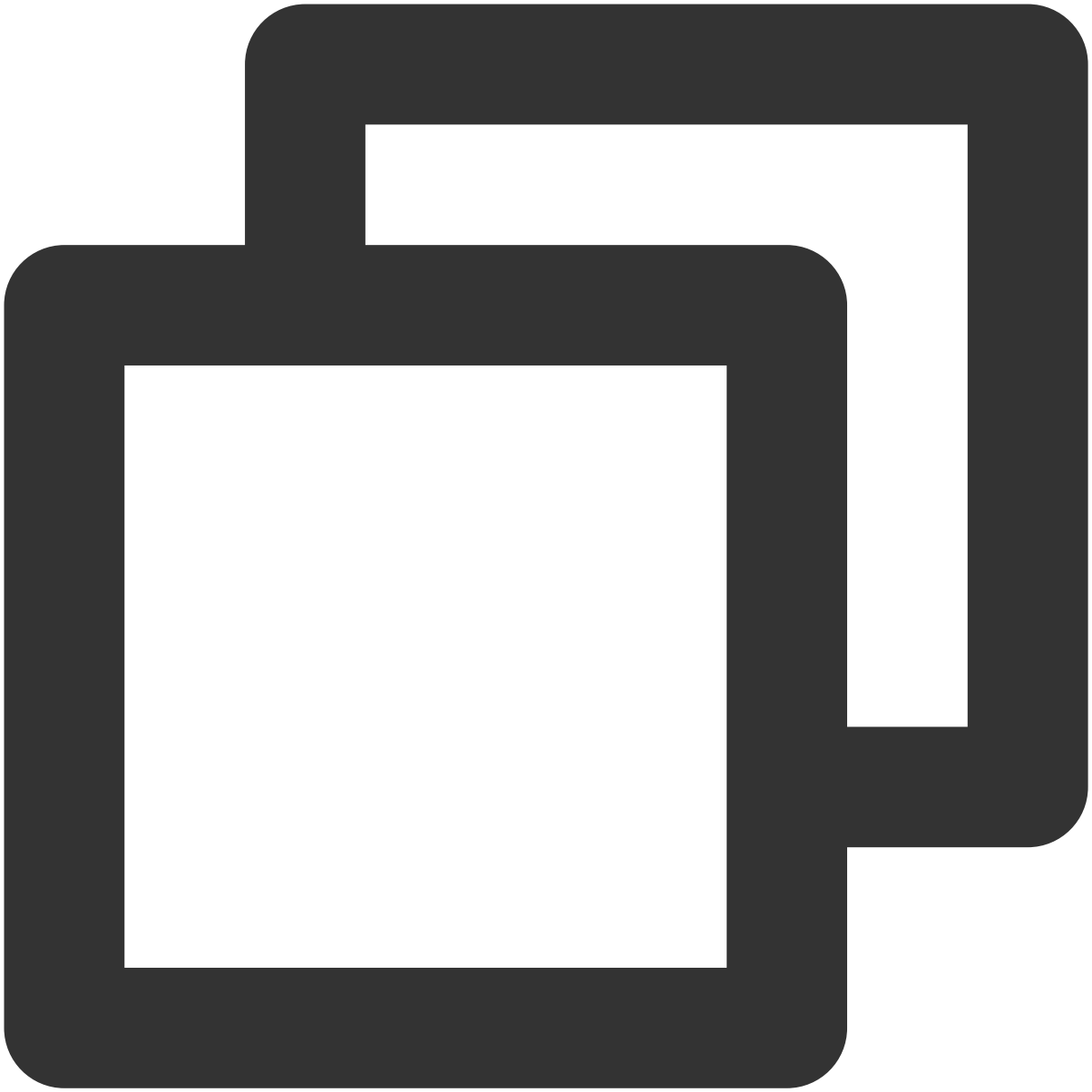
©2013-2022 Tencent Cloud. All rights reserved.



```
Page({
  tapName: function(event) {
    console.log(event)
  }
})
```

The log content is as follows:

©2013-2022 Tencent Cloud. All rights reserved.



```
{
  "type":"tap",
  "timeStamp":895,
  "target": {
    "id": "tapTest",
    "dataset":  {
      "hi":"Weixin"
    }
  },
  "currentTarget":  {
    "id": "tapTest",
```

©2013-2022 Tencent Cloud. All rights reserved.

```
      "dataset": {
        "hi":"Weixin"
      }
    },
    "detail": {
      "x":53,
      "y":14
    },
    "touches":[{
      "identifier":0,
      "pageX":53,
      "pageY":14,
      "clientX":53,
      "clientY":14
    }],
    "changedTouches":[{
      "identifier":0,
      "pageX":53,
      "pageY":14,
      "clientX":53,
      "clientY":14
    }]
  }
```

4.3 Respond to Events with WXS Function

4.3.1 Background

The effect of frequent user interactions on mini programs is laggy, for example, if there are two elements A and B on a page, and the user makes a touchmove gesture on A, and asks B to follow the gesture, movable-view is a typical example. The response to a touchmove event is as follows:

The touchmove event is thrown from the rendering layer (Webview) to the logical layer (App Service);

The logical layer (App Service) handles the touchmove event and then changes the position of B through setData.

A touchmove response requires two communications between the logic layer and the rendering layer and one rendering, which is time-consuming. In addition, the setData rendering will also block the execution of other scripts, resulting in a delay in the animation process of the whole user interaction.

4.3.2 Solution

The basic idea of this solution is to reduce the number of communications and let the events respond in the rendering layer (Webview). The framework of the mini program is divided into the rendering layer (Webview) and the logic layer (App Service), the purpose of this layering is to control, the developer's code can only run in the logic layer (App Service), and this idea must allow the developer's code to run in the rendering layer (Webview).

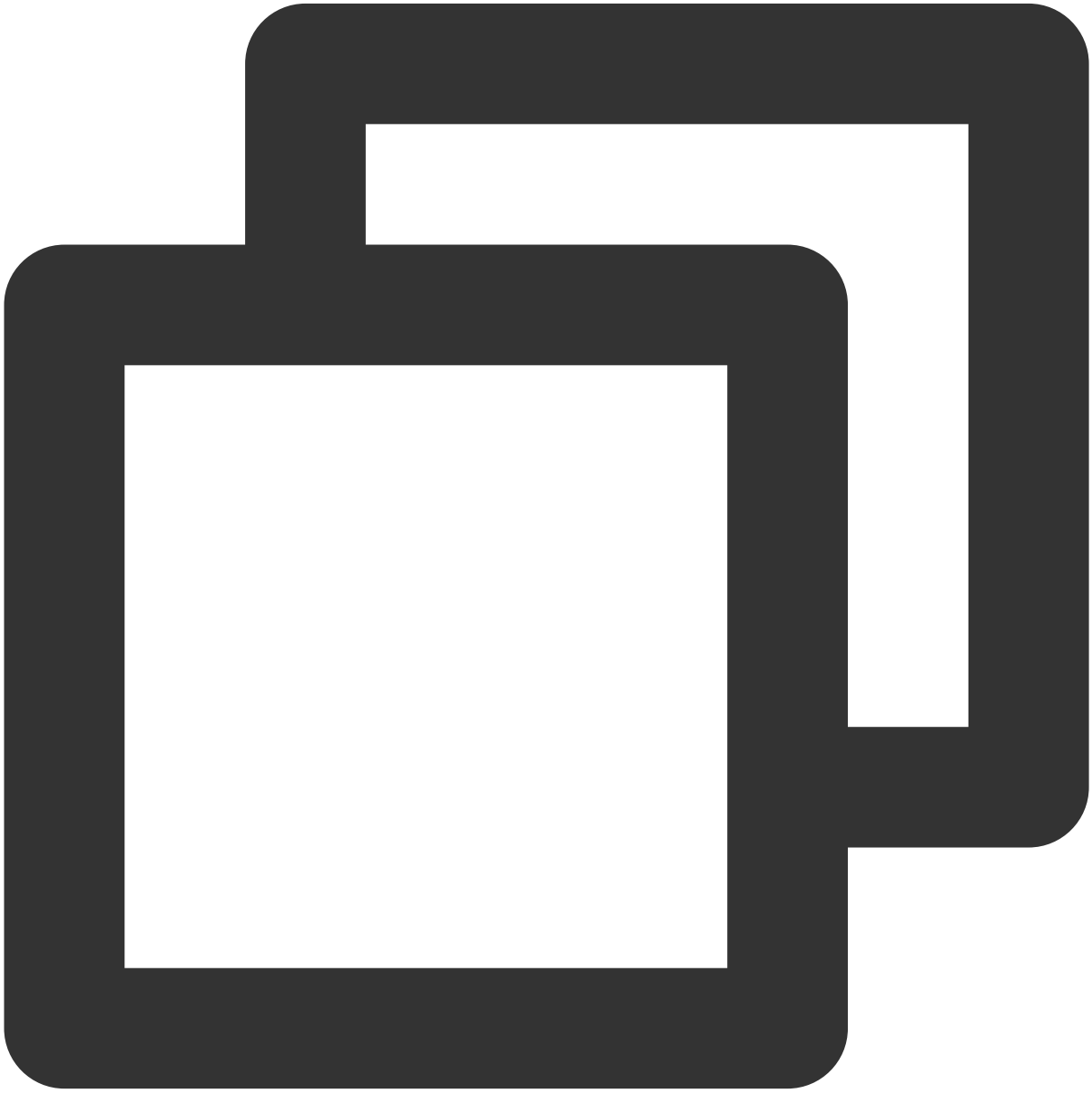
©2013-2022 Tencent Cloud. All rights reserved.

WXS functions used to respond to mini program events, can only respond to the built-in component events, does not support custom component events.

WXS function in addition to the pure logic of the operation, but also through the encapsulated ComponentDescriptor instance to access and set the component class and style, for interactive animation, set the style and class enough.

WXS function accepts two parameters, the first is event, in the original event based on the `event.instance` object, the second parameter is `ownerInstance`, and `event.instance` is a `ComponentDescriptor` object. The specific use is as follows:

WXS functions for binding and registering event handling in a component

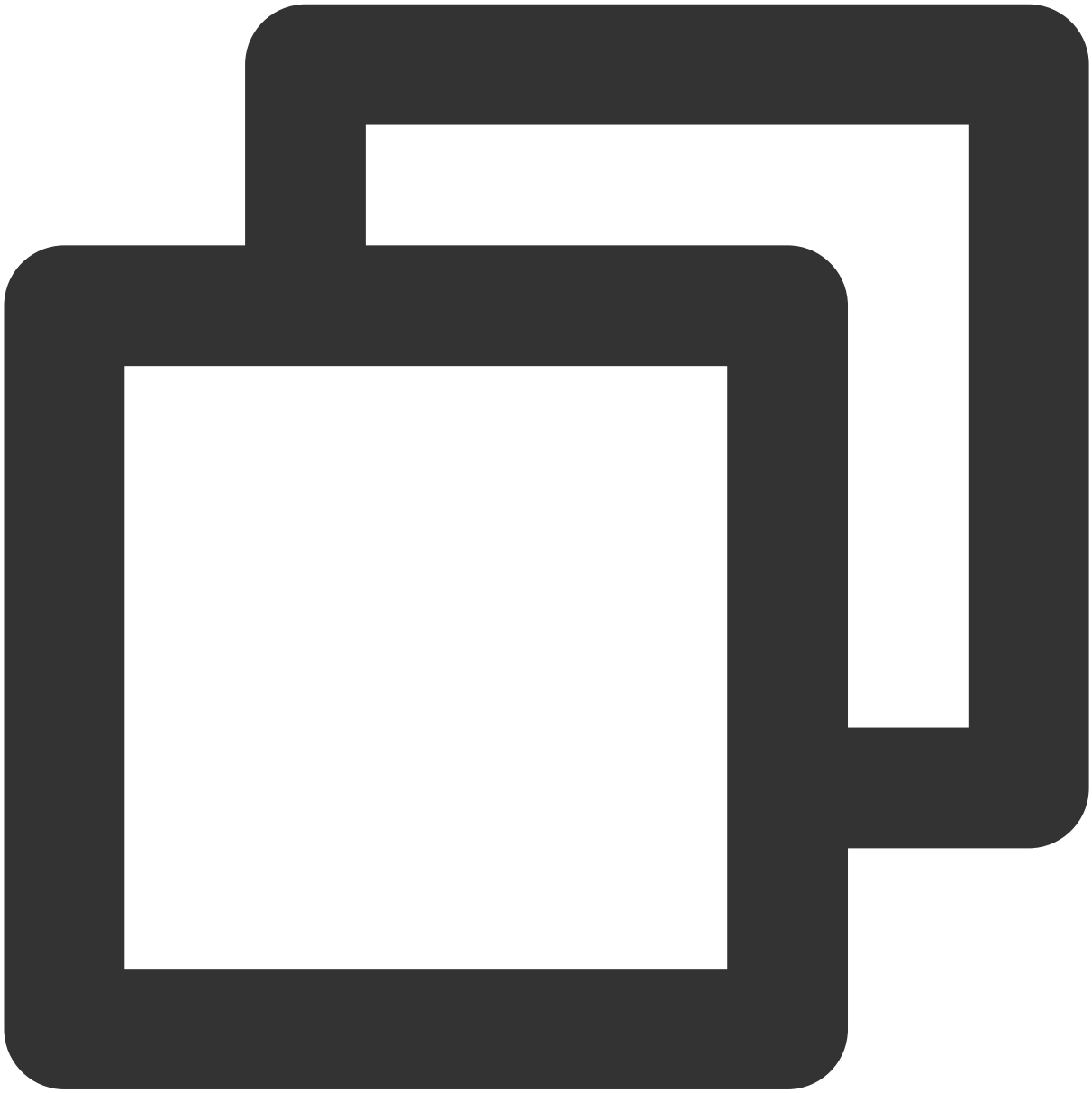


```
<wxs module="wxs" src="./test.wxs"></wxs>
```

©2013-2022 Tencent Cloud. All rights reserved.

```
<view id="tapTest" data-hi="Weixin" bindtap="{{wxs.tapName}}"> Click me! </view>
**Note: Bound WXS functions must be enclosed in {{}}**
```

The test.wxs file implements the tapName function.

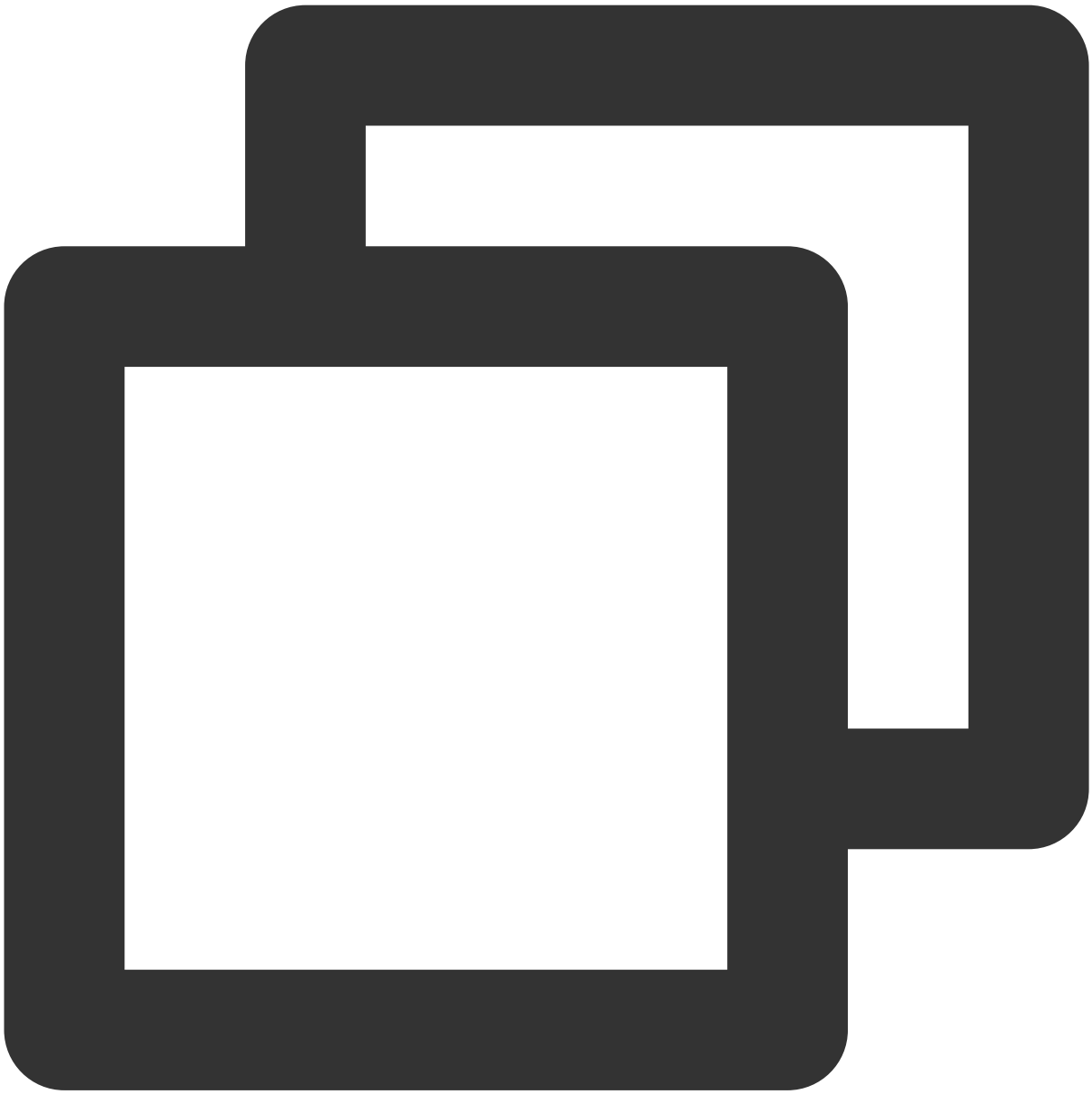


```
function tapName(event, ownerInstance) {
  console.log('tap Weixin', JSON.stringify(event))
}
module.exports = {
  tapName: tapName
}
```

©2013-2022 Tencent Cloud. All rights reserved.

ownerInstance contains methods to set the style and class of the component.

An example of a WXS function is shown below:=



```
var wxsFunction = function(event, ownerInstance) {
  var instance = ownerInstance.selectComponent('.classSelector') // Returns an inst
  instance.setStyle({
      "font-size": "14px" // Support for rpx
  })
  instance.getDataset()
  instance.setClass(className)
  // ...
```

©2013-2022 Tencent Cloud. All rights reserved.

```
    return false // does not bubble up, which is equivalent to calling both stopPropa
}
```

The input parameter `event` is the ComponentDescriptor instance of the component that triggered the event, which is the ComponentDescriptor instance of the component that triggered the event, and the ownerInstance is the instance of the page if the component that triggered the event is inside the page.

The ownerInstance represents the `ComponentDescriptor` instance of the component that triggered the event. If the component that triggered the event is inside a page, the ownerInstance represents the page instance.

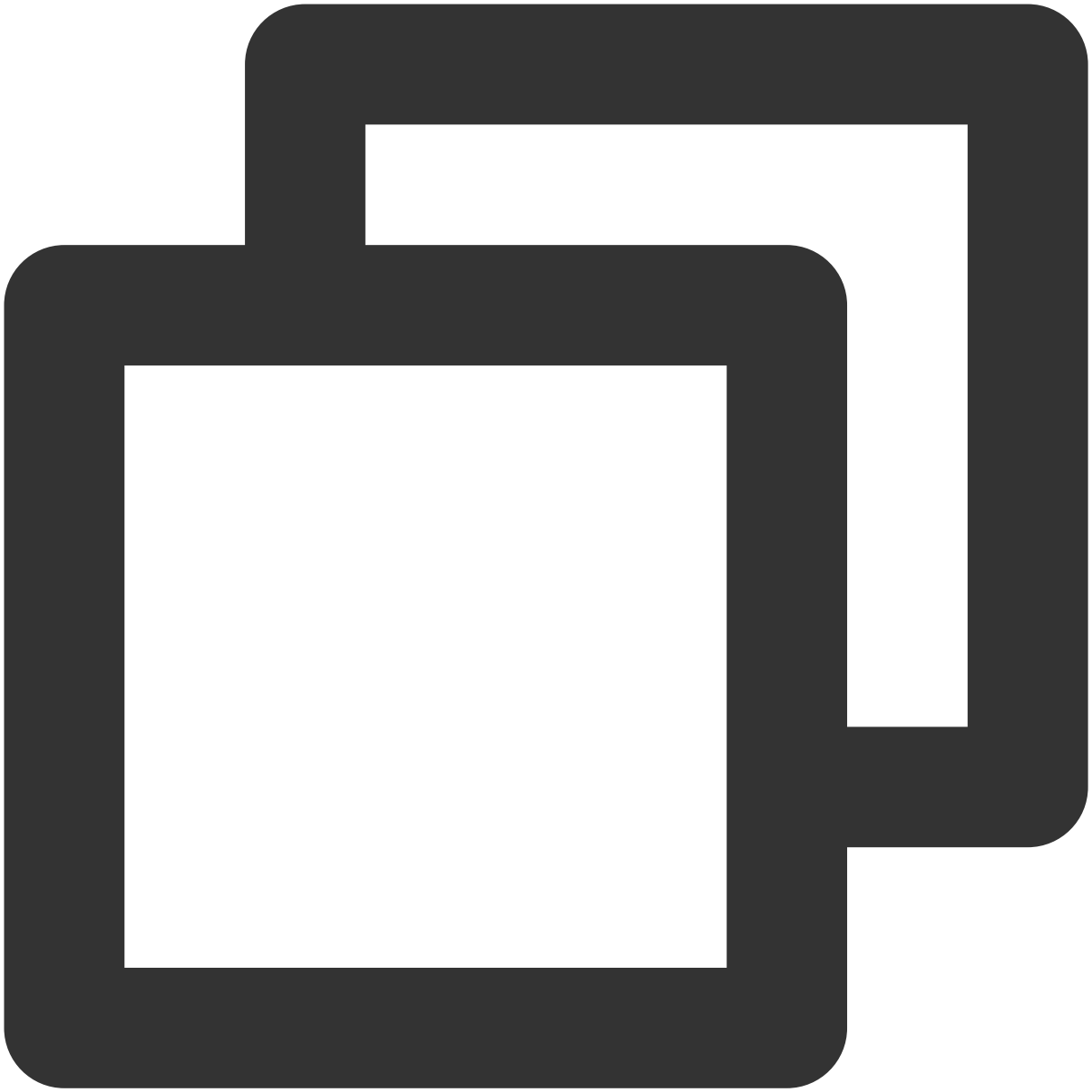
The `ComponentDescriptor` currently supports the following APIs.

Method	Parameter	Description
selectComponent	selector objects	Returns the `ComponentDescriptor` instance of the component
selectAllComponents	Array of selector objects	Returns an array of `ComponentDescriptor` instances of the component
setStyle	Object/string	Set the style of the component, supports rpx, the style set has higher priority than the style defined inside the component wxml. Cannot set the style of the topmost page
addClass/removeClass/hasClass	string	Set the class of the component, the class priority is higher than the class defined inside the component wxml. Cannot set the class of the top level page
getDataset	-	Return the dataset of the current component/page
callMethod	(funcName:string, args:object)	Call the function defined in the logic layer (App Service) of the current component/page. funcName is the name of the function, args is the parameter of the function
getState	-	Returns an object, use this method when there are local variables that need to be stored for subsequent use
triggerEvent	(eventName, detail)	Same as the triggerEvent of the component

4.3.3 Usage

WXML defines the event:

©2013-2022 Tencent Cloud. All rights reserved.



```
<wxs module="test" src="./test.wxs"></wxs>
<view
  change:prop="{{test.propObserver}}"
  prop="{{propValue}}"
  bindtouchmove="{{test.touchmove}}"
  class="movable">
</view>
```

The change:prop above (with the change: prefix in front of the property) triggers the WXS function when the prop property is set, and the value must be enclosed in `{{}}` . Similarly the observer property inside properties defined

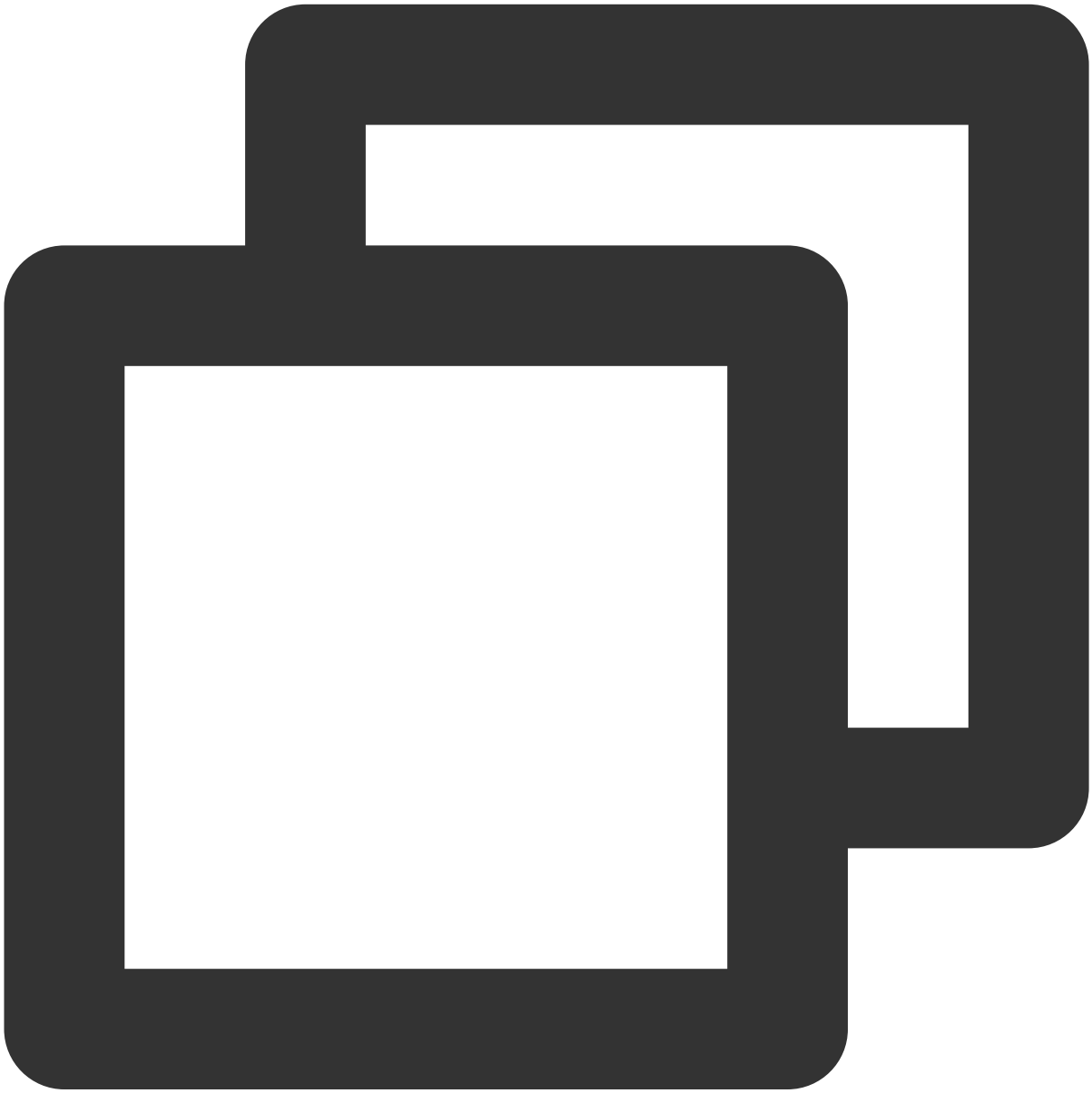
©2013-2022 Tencent Cloud. All rights reserved.

by Component is triggered after a `setData({propValue: newValue})` call.

Note：

WXS functions must be enclosed in `{{}}` . The WXS function is triggered when the value of the prop is set, not just when the value is changed, so the WxsPropObserver function is called once when the page is initialised.

The WXS file `test.wxs` defines and exports event handlers and functions triggered by property changes:



```
module.exports = {
  touchmove: function(event, instance) {
      console.log('log event', JSON.stringify(event))
  },
```

©2013-2022 Tencent Cloud. All rights reserved.

```
  propObserver: function(newValue, oldValue, ownerInstance, instance) {
      console.log('prop observer', newValue, oldValue)
  }
}
```

4.4Events

4.4.1 Event Types

Events are divided into bubbling events and non-bubbling ones.

Bubbling event: When an event is triggered on a component, this event is propagated to the component's parent node;

Non-bubbling event: When an event is triggered on a component, this event is not propagated to the component's parent node.

WXML bubbling events:

Type	Trigger Condition
touchstart	Finger touch starts
touchmove	Finger moves after touch
touchcancel	Finger touch is interrupted by call reminder, pop-up window, etc.
touchend	Finger touch ends
tap	The finger leaves the screen after touch
longpress	The finger leaves the screen after it taps and holds on the screen for more than 350 ms. If an event callback function is specified and this event is triggered, the tap event is not triggered.
longtap	The finger leaves the screen after it taps and holds on the screen for more than 350 ms (it is recommended to use longpress event instead).
transitionend	Triggered when a WXSS transition or wx.createAnimation animation ends
animationstart	Triggered when a WXSS animation starts
animationiteration	Triggered after an iteration of a WXSS animation ends
animationend	Triggered when a WXSS animation completes
touchforcechange	Triggered when the screen is tapped again on an iPhone supporting 3D Touch.

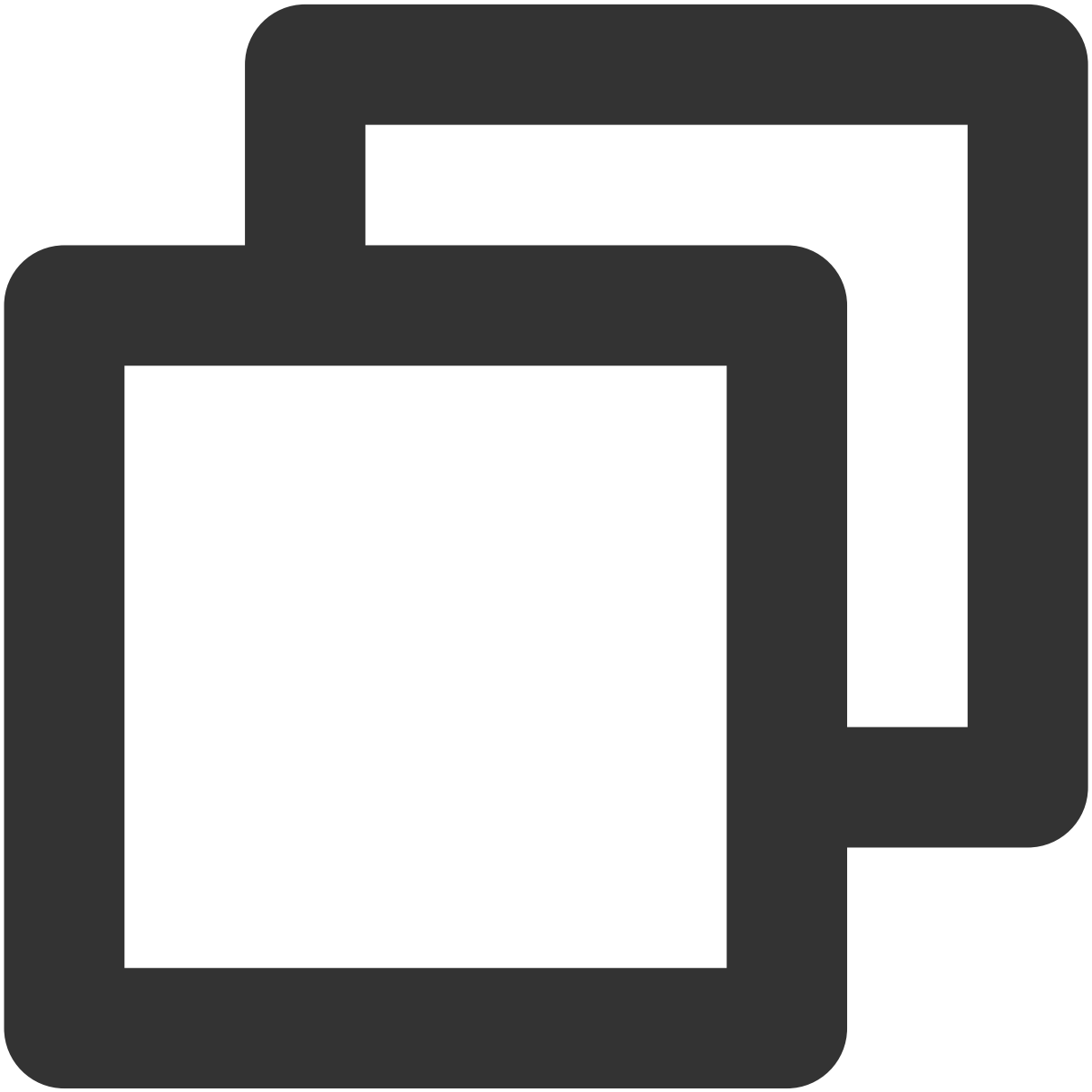
Note：

©2013-2022 Tencent Cloud. All rights reserved.

Unless otherwise specified, the custom events of components other than those listed above are non-bubbling events, such as the submit event of [form]((form), input event of [input]((input), and scroll event of scroll-view. For more information, see Component Overview.

4.4.2 Event Binding and Bubbling

Event bindings are written similarly to component properties, e.g.:

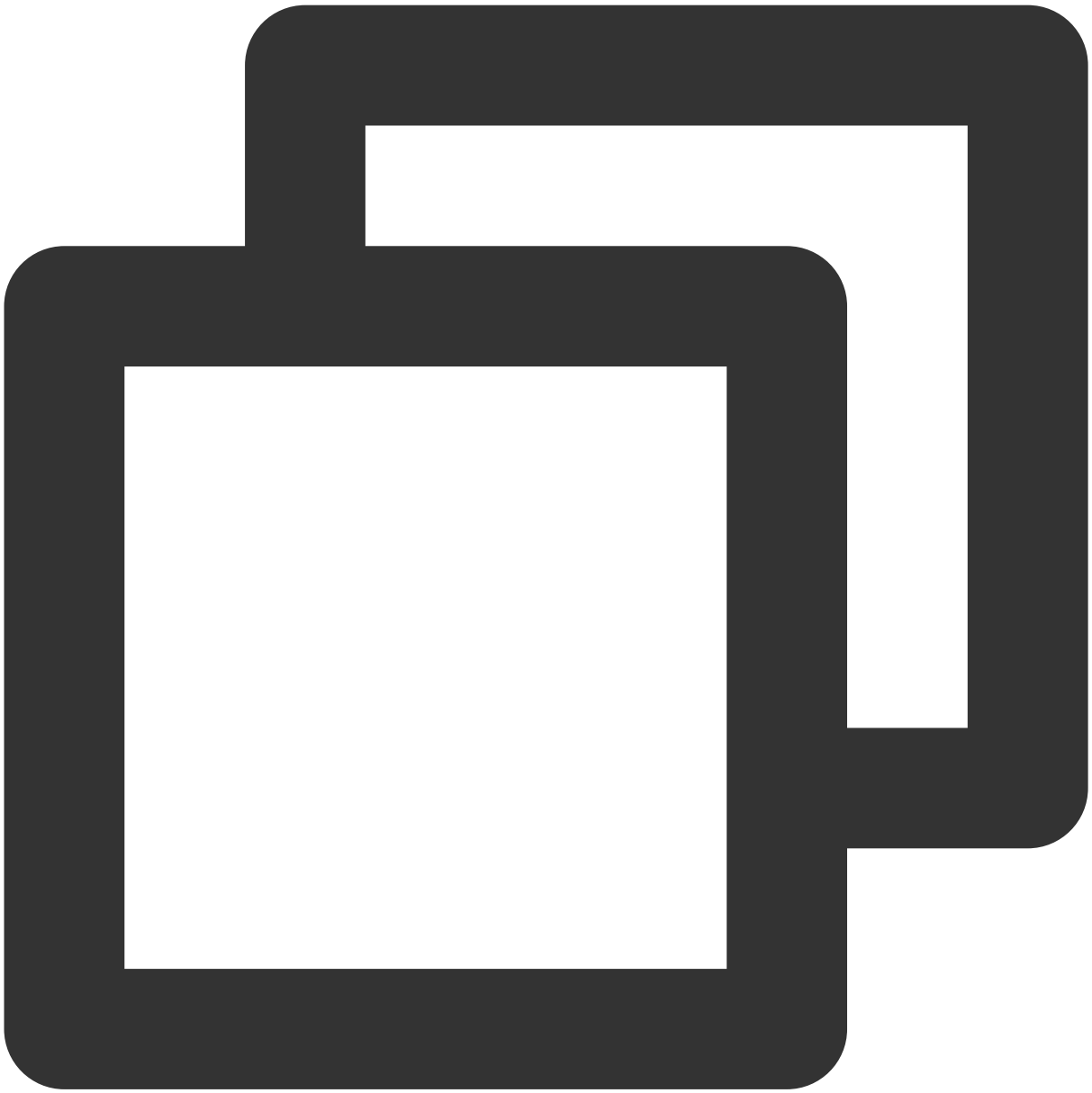


```
<view bindtap="handleTap">
    Click here!
</view>
```

©2013-2022 Tencent Cloud. All rights reserved.

If the user clicks on this view, the page's handleTap is called.

The event binding function can be a data binding, e.g.:



```
<view bindtap="{{ handlerName }}">
  Click here!
</view>
```

In this case, the page's this.data.handlerName must be a string specifying the name of the event handler; if it's an empty string, the binding is disabled (you can use this feature to temporarily disable some events).

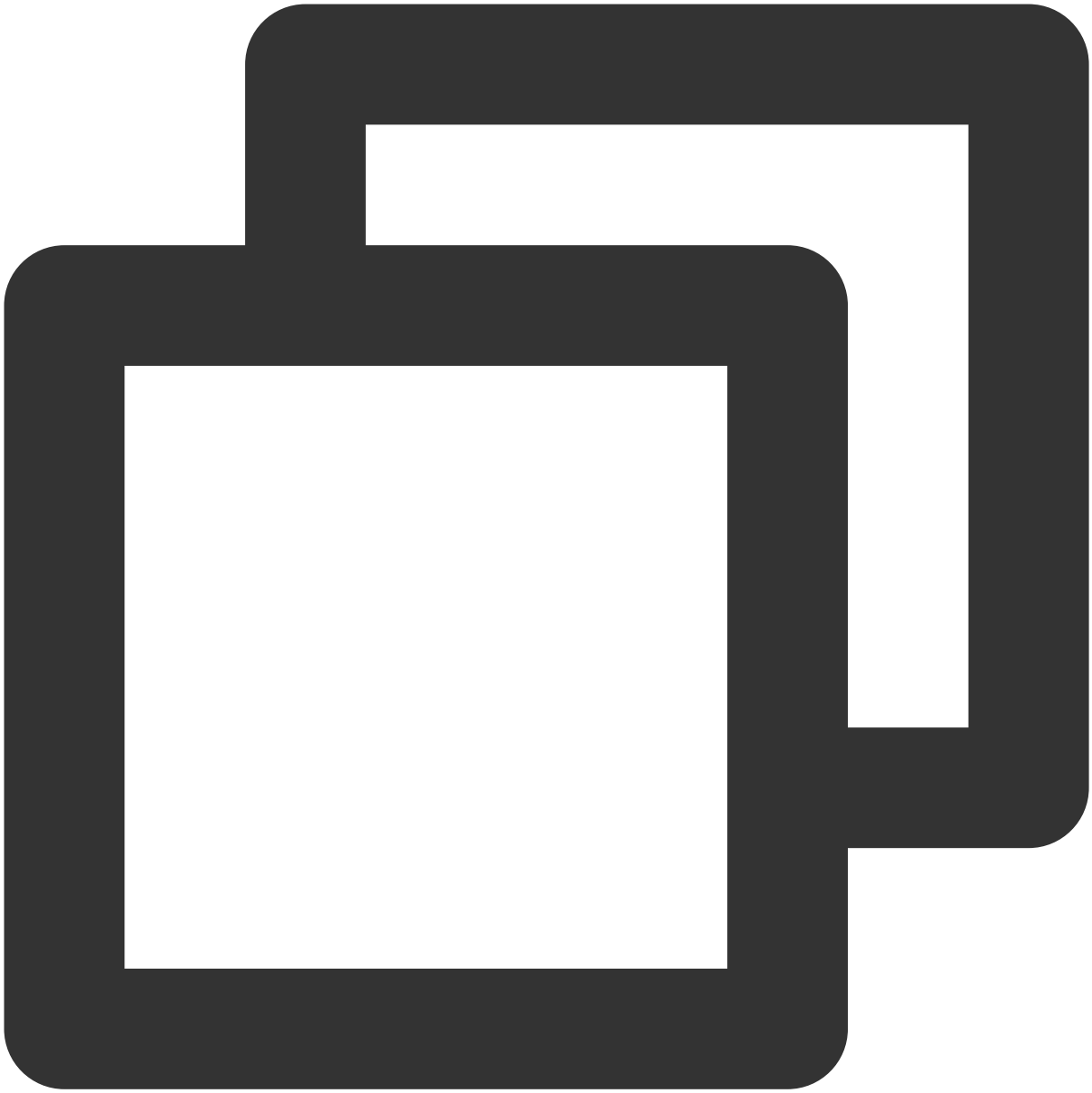
©2013-2022 Tencent Cloud. All rights reserved.

In most components and custom components, bind can be immediately followed by a colon, and the meaning remains the same, e.g. bind:tap

4.4.3 Binding and stopping event bubbling

In addition to bind, events can also be bound with catch. Unlike bind, catch prevents events from bubbling up.
For example, in the following example, clicking on the inner view calls handleTap3 and handleTap2 successively (because the tap event bubbles up to the middle view, which prevents the tap event from bubbling up to the parent node), and clicking on the middle view triggers handleTap2, which triggers handleTap2, which triggers handleTap2, which triggers handleTap2, which triggers handleTap2, which triggers handleTap2. Click outer view will trigger handleTap1.

©2013-2022 Tencent Cloud. All rights reserved.



```
<view id="outer" bindtap="handleTap1">
  outer view
  <view id="middle" catchtap="handleTap2">
    middle view
    <view id="inner" bindtap="handleTap3">
      inner view
    </view>
  </view>
</view>
```

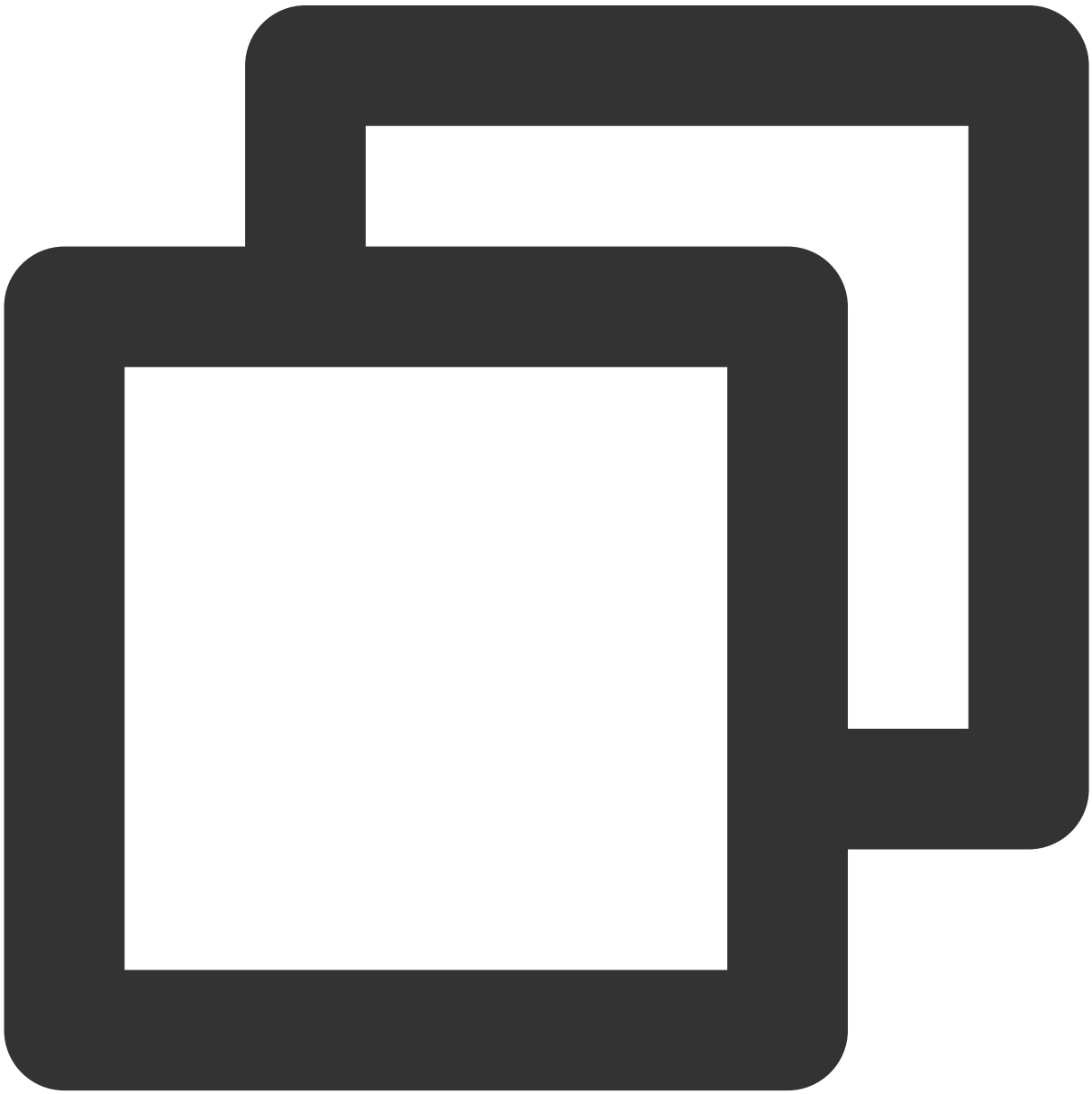
©2013-2022 Tencent Cloud. All rights reserved.

4.4.4 Mutually exclusive event binding

As of Base Library version 1.5.44, you can use mut-bind to bind events in addition to bind and catch. When a mut-bind is triggered, if the event bubbles to another node, the mut-bind bindings on the other node will not be triggered, but the bind and catch bindings will still be triggered.

In other words, all mut-binds are "mutually exclusive" and only one of them will be triggered. At the same time, it doesn't affect the binding effect of bind and catch at all.

For example, in the example below, clicking inner view will call handleTap3 and handleTap2, and clicking middle view will call handleTap2 and handleTap1.



©2013-2022 Tencent Cloud. All rights reserved.

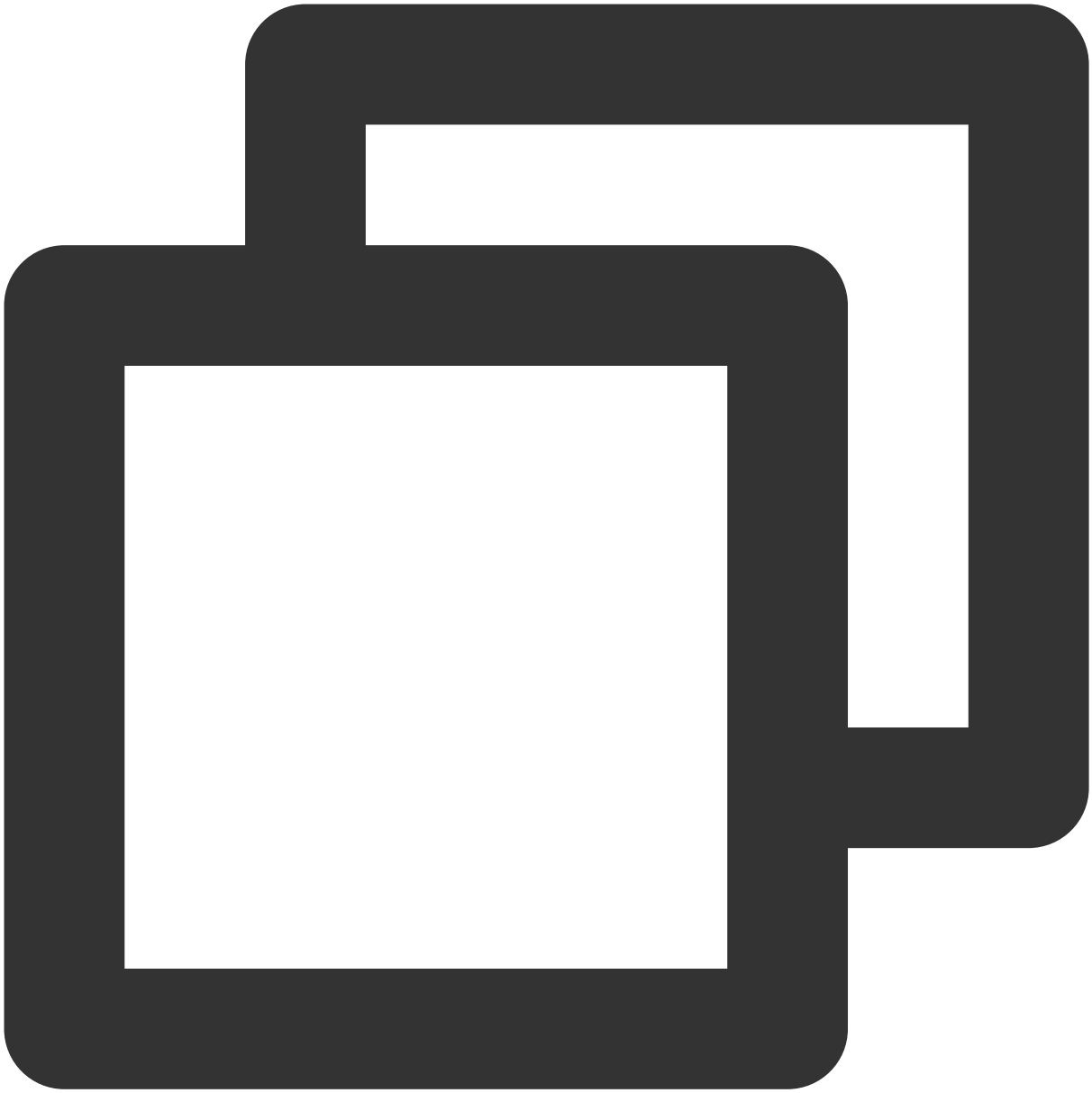
```
<view id="outer" mut-bind:tap="handleTap1">
  outer view
  <view id="middle" bindtap="handleTap2">
    middle view
    <view id="inner" mut-bind:tap="handleTap3">
      inner view
    </view>
  </view>
</view>
```

4.4.5 Event Capturing

touch-related events support capturing. In event capturing, which precedes bubbling, events arrive at nodes in an order reverse of that in which they do during bubbling. To listen to events during capturing, use `capture-bind` or `capture-catch` as the keyword. The latter interrupts the capturing and cancels bubbling.

In the code below, tap "inner view" to call `handleTap2` , `handleTap4` , `handleTap3` , and `handleTap1` in turn.

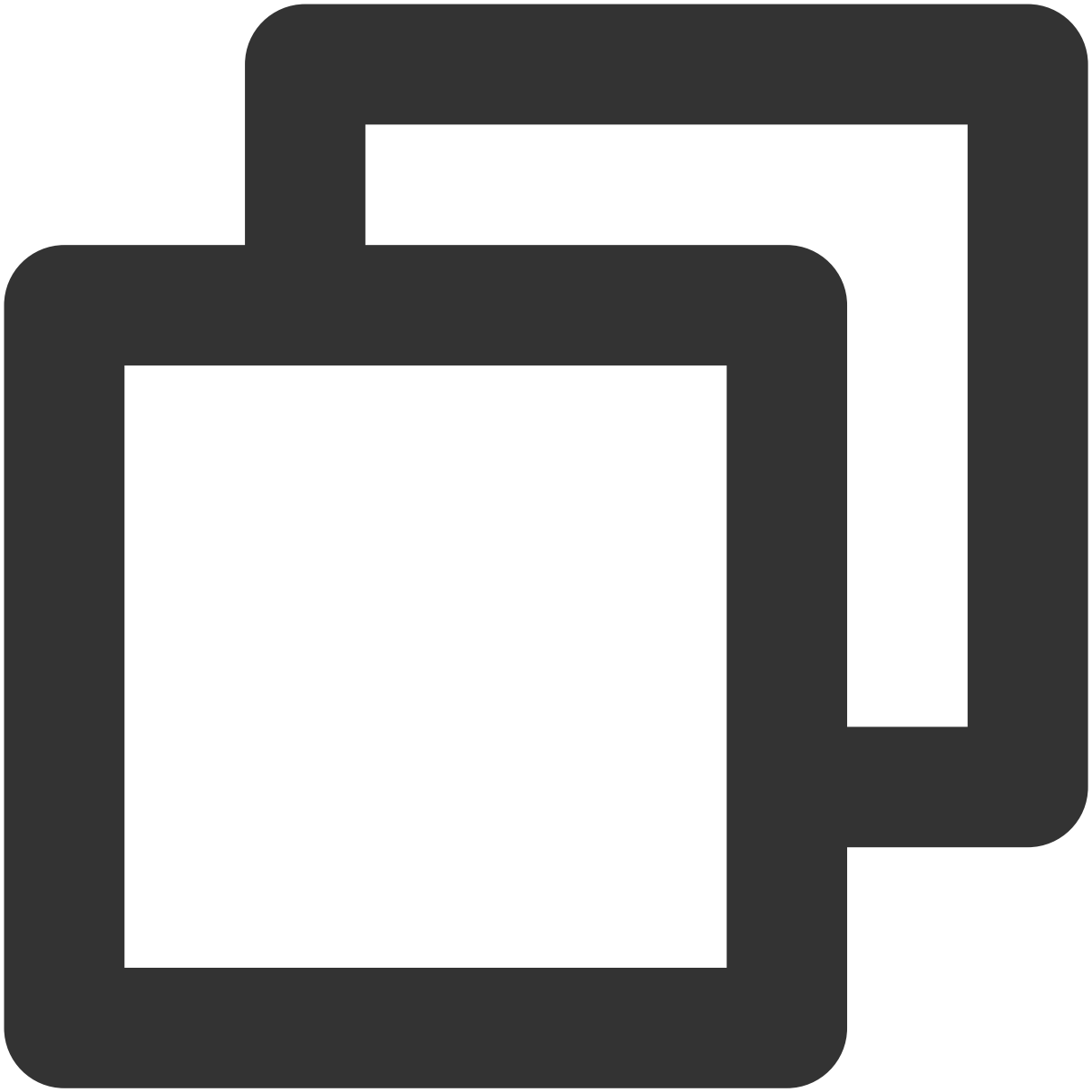
©2013-2022 Tencent Cloud. All rights reserved.



```
<view id="outer" bind:touchstart="handleTap1" capture-bind:touchstart="handleTap2">
  outer view
  <view id="inner" bind:touchstart="handleTap3" capture-bind:touchstart="handleTap4
    inner view
  </view>
</view>
```

If the first capture-bind in the above code is changed to capture-catch, only handleTap2 is triggered.

©2013-2022 Tencent Cloud. All rights reserved.



```
<view id="outer" bind:touchstart="handleTap1" capture-catch:touchstart="handleTap2"
  outer view
  <view id="inner" bind:touchstart="handleTap3" capture-bind:touchstart="handleTap4
    inner view
  </view>
</view>
```

4.4.6 Event Objects

©2013-2022 Tencent Cloud. All rights reserved.

Unless otherwise specified, when the component triggers an event, the handler bound to the event at logic layer receives an event object.

BaseEvent object properties:

Property	Type	Description
type	String	Event Type
timeStamp	Integer	Timestamp when the event is generated
target	Object	A collection of property valuesfor the component triggering the event
currentTarget	Object	A collection of property valuesfor the current component
mark	Object	Event flag data

CustomEvent object properties (inherit from BaseEvent):

Property	Type	Description
detail	Object	Additional information

TouchEvent object properties (inherit from BaseEvent):

Property	Type	Description
touches	Array	Touch event. An array of information of touch points staying in the screen.
changedTouches	Array	Touch event. An array of information of touch points involving changes in the screen

Special events: The touch events in canvas are non-bubbling events and thus have no currentTarget.

4.4.7 type

Represents the type of an event.

4.4.8 timeStamp

Number of milliseconds from the moment when the page is opened until the event is triggered.

4.4.9 target

The source component triggering the event.

Property	Type	Description
id	String	ID of the event source component

©2013-2022 Tencent Cloud. All rights reserved.

dataset	Object	A collection of custom properties starting with data- on the event source component

4.4.10 currentTarget

The current component bound to the event.

Property	Type	Description
id	String	Current component ID
dataset	Object	A collection of custom properties starting with data- on the current component

Note：

You can refer to the above example for "target" and "currentTarget". When you tap "inner view", the event objects "target" and "currentTarget" received by handleTap3 are both "inner", but the "target" and "currentTarget" received by handleTap2 are "inner" and "middle", respectively.

4.4.11 dataset

You can add some custom data to the component node so as to obtain the custom node data from the event for logical processing.

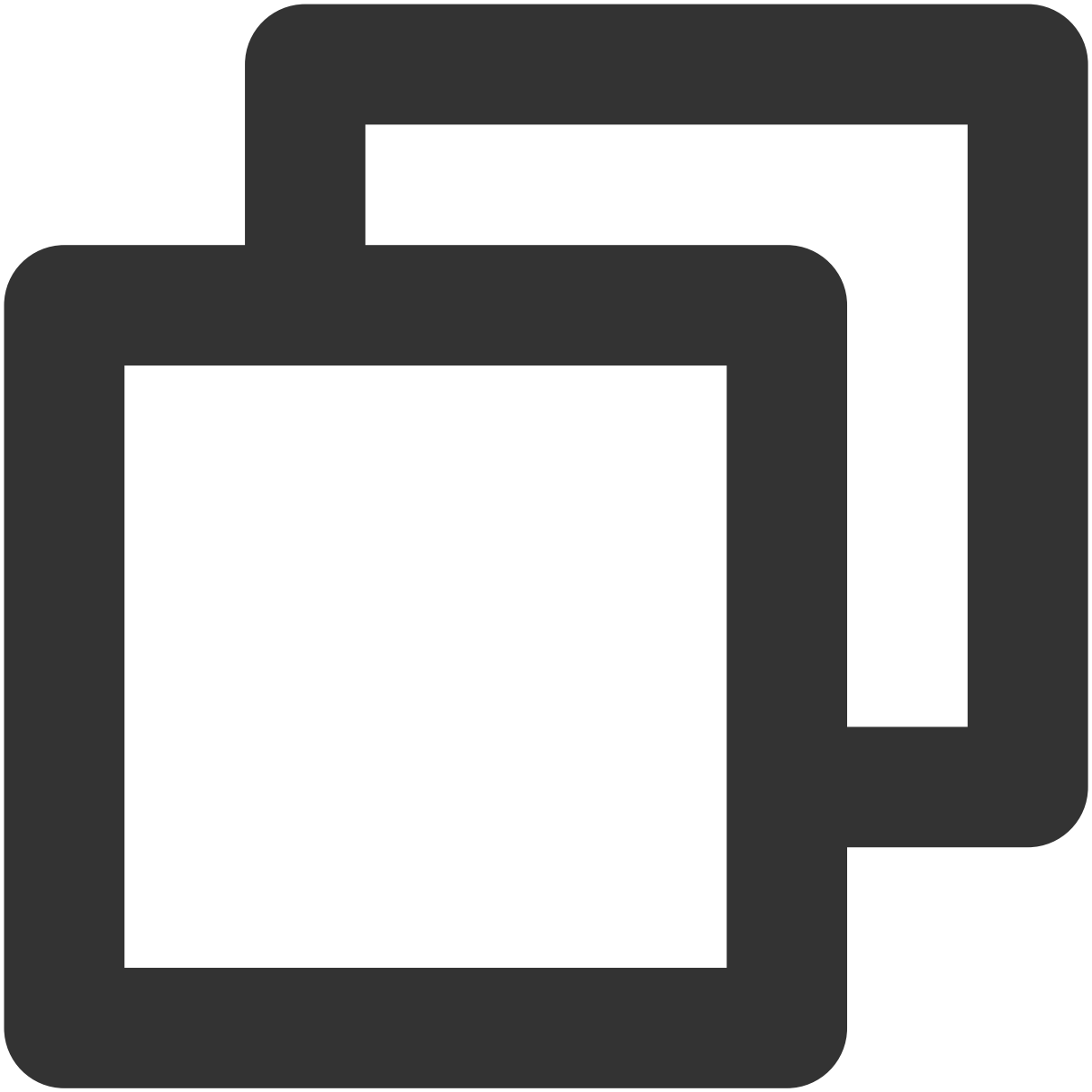
In WXML, these custom data begins with data- and multiple words are joined with hyphen -. In the code, a hyphenation is converted to camel case, and the uppercase characters are automatically converted to lowercase ones. For example:

data-element-type is converted to event.currentTarget.dataset.elementType;

data-elementType is converted to event.currentTarget.dataset.elementtype.

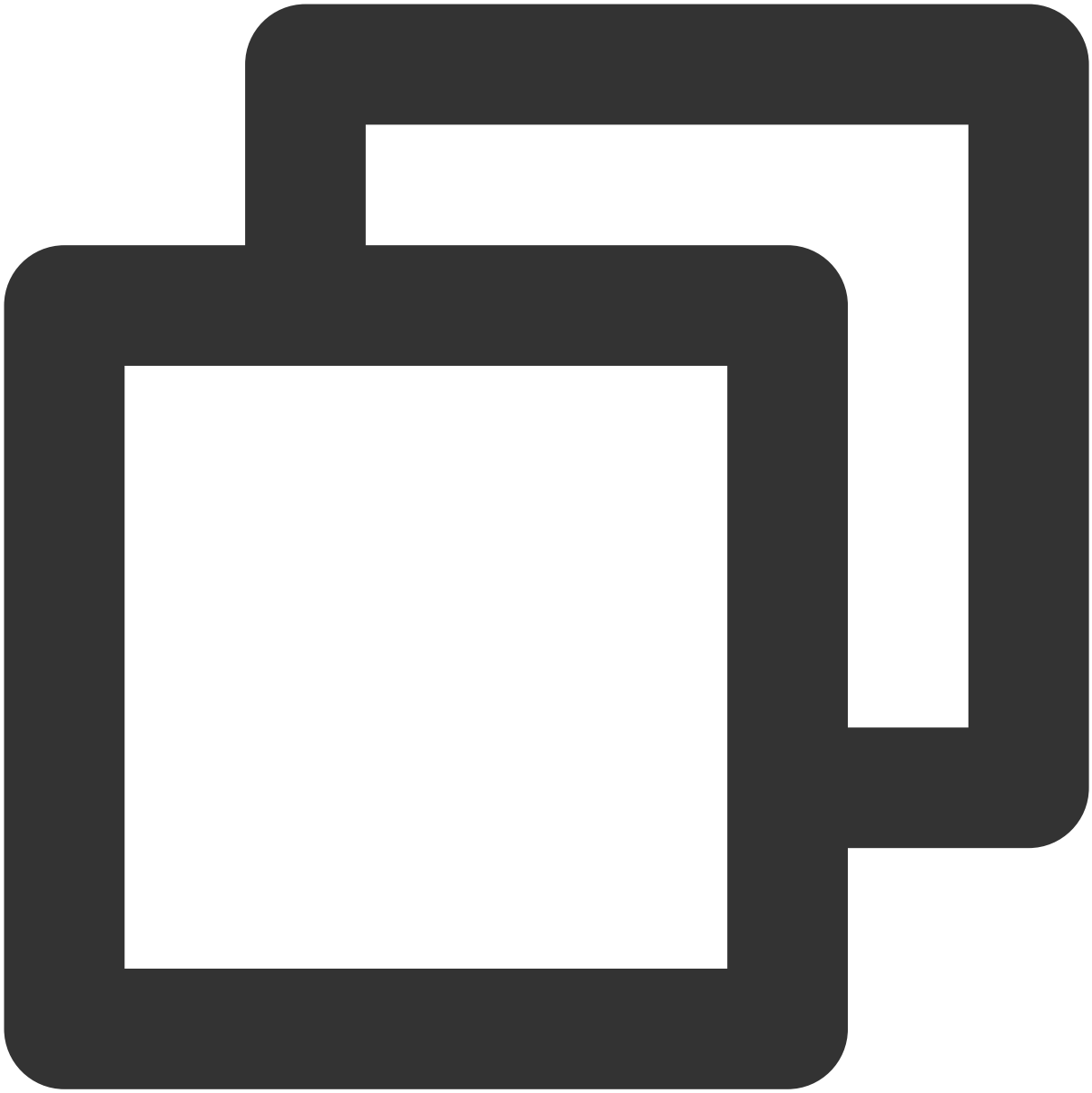
Example:

©2013-2022 Tencent Cloud. All rights reserved.



```
<view data-alpha-beta="1" data-alphaBeta="2" bindtap="bindViewTap"> DataSet Test </
```

©2013-2022 Tencent Cloud. All rights reserved.



```
Page({
  bindViewTap:function(event){
    event.currentTarget.dataset.alphaBeta === 1 // - will convert to camel case
    event.currentTarget.dataset.alphaBeta === 2 // uppercase will be converted to l
  }
})
```

4.4.12 touches

©2013-2022 Tencent Cloud. All rights reserved.

"touches" is an array, in which each element is a Touch object ("touches" in the canvas touch event is a CanvasTouch array). Touch points staying in the screen.

Touch objects

Property	Type	Description
identifier	Number	Touch point identifier
pageX, pageY	Number	Distance from the upper left corner of the document. The top left corner is the origin, the horizontal line is X-axis, and the vertical line is Y-axis.
clientX, clientY	Number	Distance from the upper left corner of the page display area (the area on the screen without navigation pane). The horizontal line is X-axis and the vertical line is Y-axis.

CanvasTouch objects

Property	Type	Description
identifier	Number	Touch point identifier
x, y	Number	Distance from the upper left corner of Canvas. Canvas's upper left corner is the origin, the horizontal line is X-axis, and the vertical line is Y-axis.

4.4.13 changedTouches

Indicates the touch points involving changes (in the same format as "touches"), including touchstart (appearance of touch point), touchmove (movement of touch point), and touchend/touchcancel (disappearance of touch point).

4.4.14 detail

The data carried by the custom event. For example, the submission event of a form component carries user's input, and the error event of media carries the error message. For more information, see the event definitions in Component Overview.

Similar to pageX and page Y, the x and y in detail of the tap event represent the distance from the upper left corner of the document.

5. Basic Components

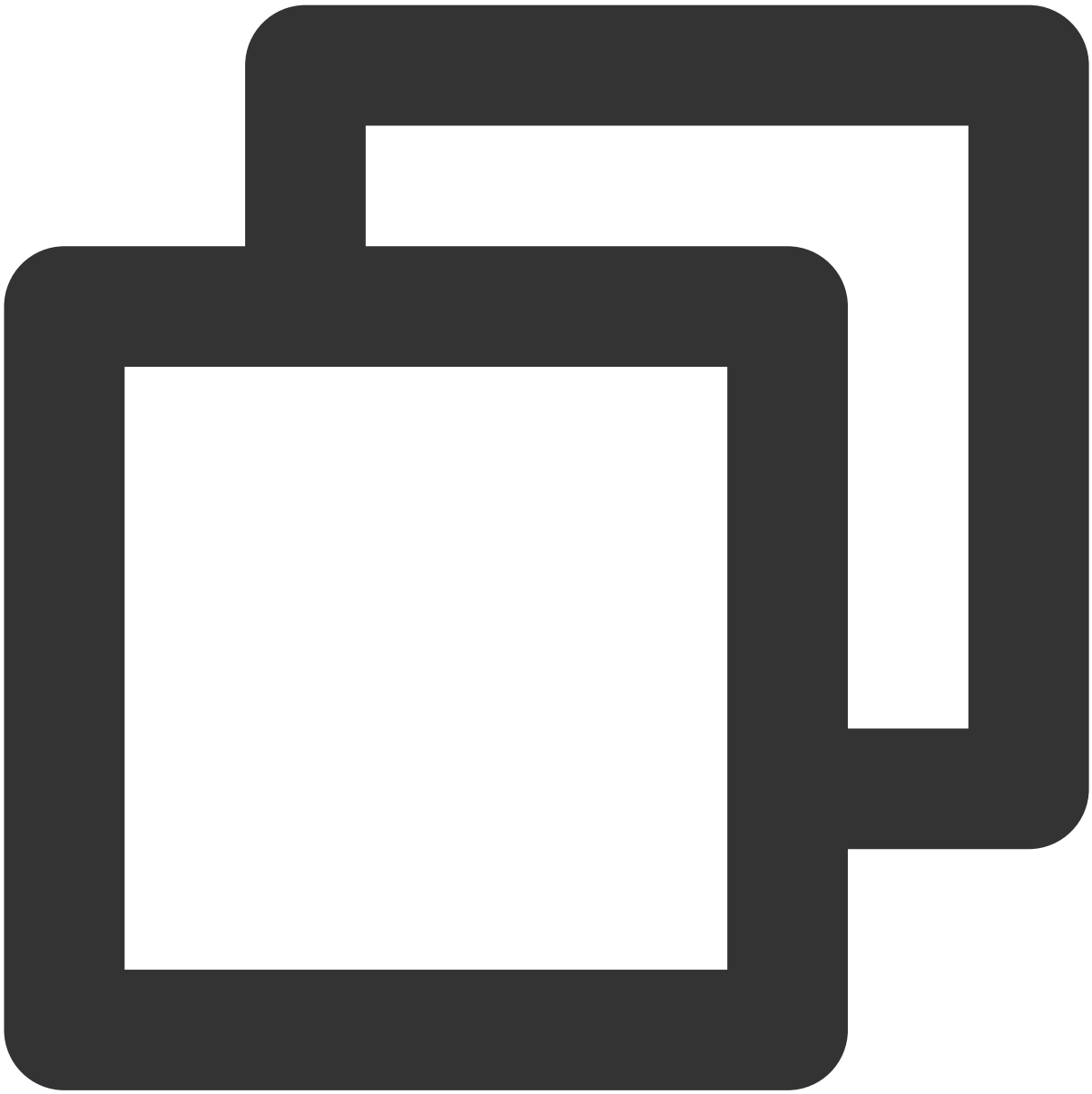
The framework provides developers with a set of basic components that can be combined to achieve quick development. For more information, see Component Overview.

What is component?

A component is the basic building block of a view layer;

©2013-2022 Tencent Cloud. All rights reserved.

A component includes a `start tag` , `end tag` , property (modifies the component), and content (between the two tags).



```
<tagname property="value">
Content goes here ...
</tagname>
```

Note：

All components and properties are in lowercase and joined with a hyphen -

Property Types

©2013-2022 Tencent Cloud. All rights reserved.

Type	Description	Comment
Boolean	Boolean value	If a component has this property, the property value is always considered true regardless of what the actual value is. The property value is false only when the component has no this property. If the property value is a variable, the variable is converted to a Boolean value.
Number	Number	1, 2.5
String	String	"string"
Array	Array	[1, "string"]
Object	Object	{ key: value }
EventHandler	Event handler	"handlerName" is the event handler name defined in Page
Any	Any property	-

Common Properties

All components have the following properties:

Property	Type	Description	Comment
id	String	A unique identifier of a component	Remains unique across the entire page.
class	String	Component style class	The style class defined in the WXSS
style	String	Component inline style	This is an inline style that can be dynamically set
hidden	Boolean	Whether to display the component	All components are displayed by default
data-*	Any	Custom property	When an event is triggered on the component, it is sent to the event handler
bind* / catch*	EventHandler	Component event	See Event

Special Properties

Almost all components have their own custom properties. You can modify the features or style of the component. See the definition of each Component Overview.

©2013-2022 Tencent Cloud. All rights reserved.

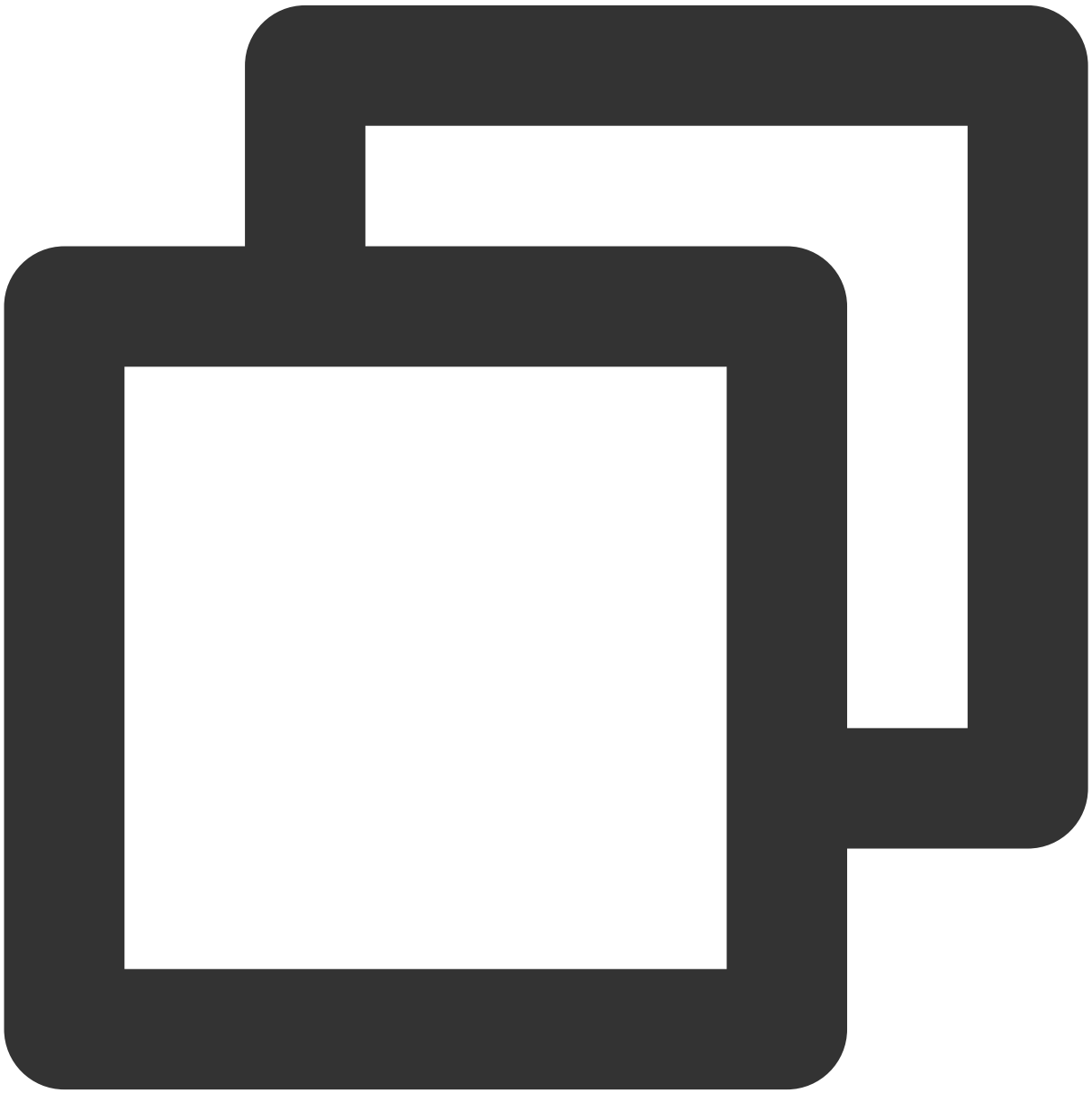
6. Getting Node Information on Interface

6.1 WXML Node Information

The API for querying node information is used to get information of a node such as its properties, style, and position in the interface.

This API is mostly used to query the current position and scroll position of a node in the interface.

Sample code:



```
const query = wx.createSelectorQuery()
```

©2013-2022 Tencent Cloud. All rights reserved.

```
query.select('#the-id').boundingClientRect(function(res){
  res.top // Coordinates of the upper boundary of the #the-id node (relative to the
})
query.selectViewport().scrollOffset(function(res){
  res.scrollTop // the vertical scroll position of the display area
})
query.exec()
```

In the above example, #the-id is a node selector, which is slightly different from the CSS selector. For more information, see SelectorQuery.select.

In custom components or pages that contain custom components, it is recommended to use this.createSelectorQuery, instead of wx.createSelectorQuery, to ensure nodes are selected within the correct area.

6.2 WXML Node Layout Intersection

The APIs for node layout intersection is used to listen to the layout intersection between two or more component nodes. This set of APIs are often used to determine whether certain nodes are visible to users and the percentage of the visible area.

The set of APIs involve the following concepts:

Reference node: The reference node for listening, whose layout area is taken as the reference area. If there are multiple reference nodes, the intersection of their layout areas is taken as the reference area. The page display area can also be used as a reference area;

Target node: The target to be listened to. Only one target node is allowed by default (you can listen to multiple nodes at the same time using the selectAll option);

Intersection area: The intersection area between the layout area of the target node and the reference area;

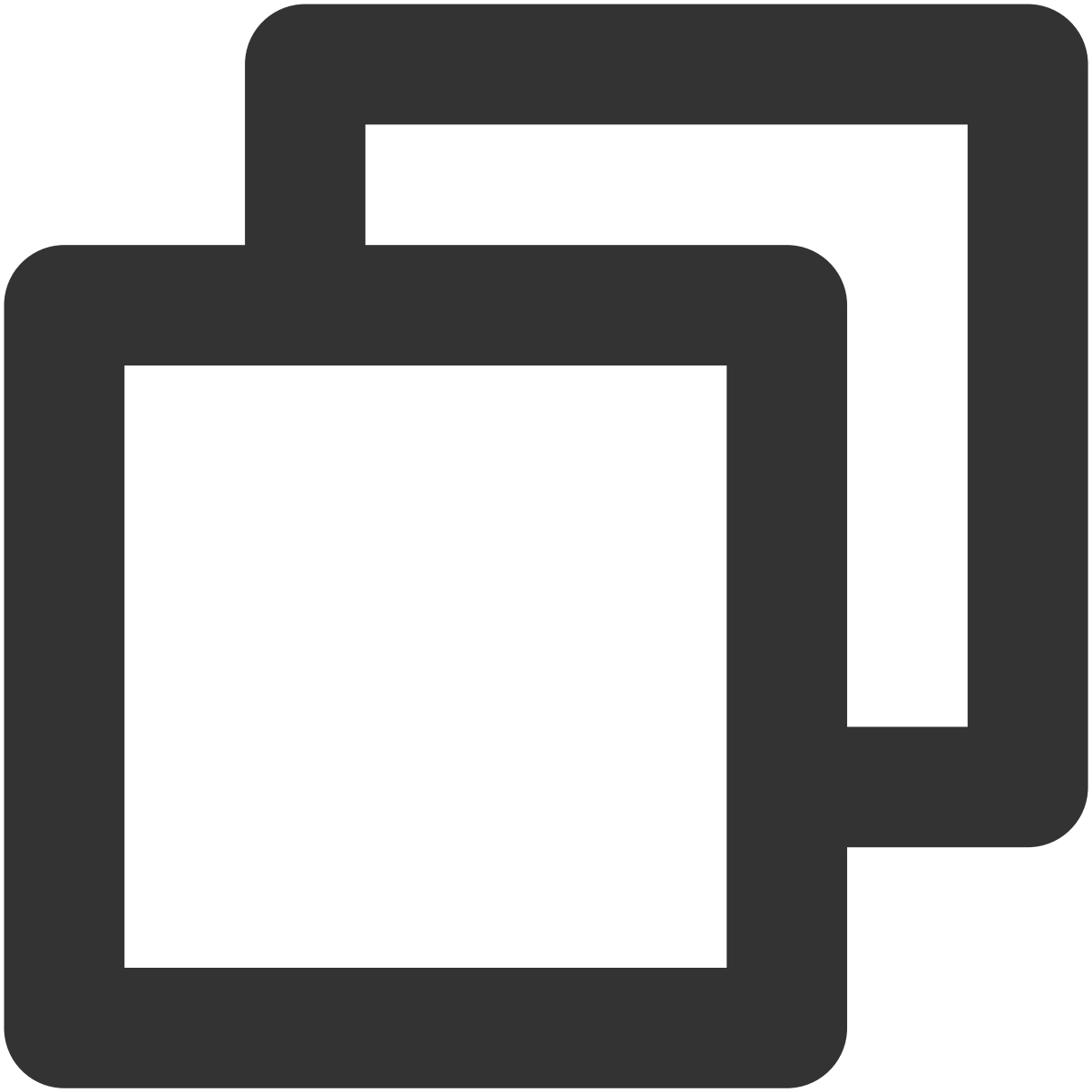
Intersection percentage: The percentage of the intersection area in the reference area

Threshold: If the intersection percentage reaches the threshold, the callback function of the listener is triggered. Multiple thresholds can be set;

In the following sample code, a callback function is triggered each time the target node (specified using the selector .target-class) enters or exits the page display area.

Sample code:

©2013-2022 Tencent Cloud. All rights reserved.



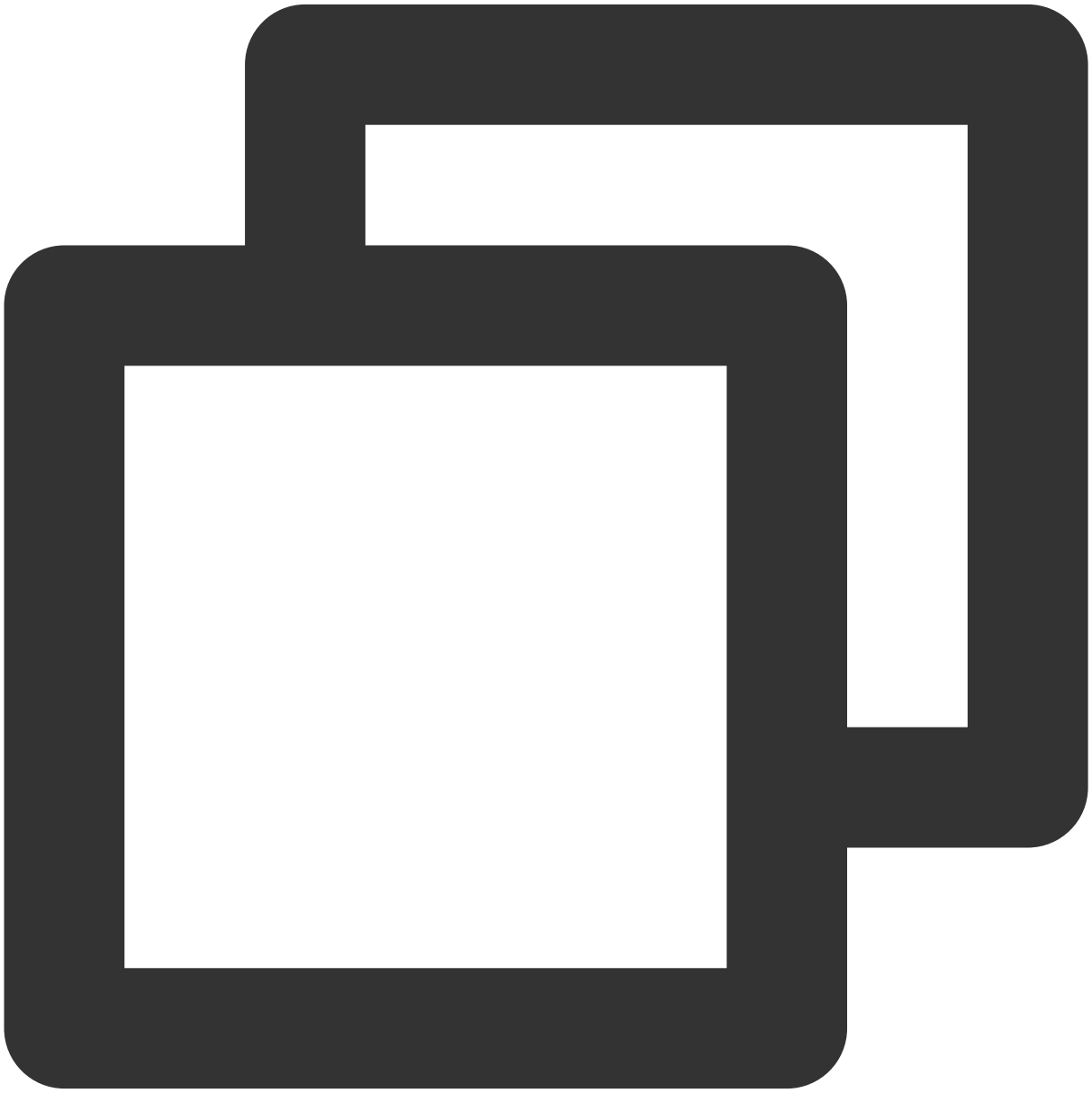
```
Page({
  onLoad: function(){
    wx.createIntersectionObserver().relativeToViewport().observe('.target-class', (
      res.id // target node id
      res.dataset // target node dataset
      res.intersectionRatio // the proportion of the intersection area to the layou
      res.intersectionRect // The intersection region
      res.intersectionRect.left // the coordinates of the left boundary of the inte
      res.intersectionRect.top // the upper boundary of the intersection region
      res.intersectionRect.width // width of the intersection region
      res.intersectionRect.height // height of the intersection region
```

©2013-2022 Tencent Cloud. All rights reserved.

```
      })
    }
})
```

The following sample code can be in the target node (specified with the selector .target-class) and the reference node (specified with the selector .relative-class) in the page display area of the intersection or separation, and the degree of intersection or separation to the extent of the layout of the target node of the area of the 20% and 50%, trigger the callback function.

Sample code:



```
Page({
```

©2013-2022 Tencent Cloud. All rights reserved.

```
  onLoad: function(){
    wx.createIntersectionObserver(this, {
      thresholds: [0.2, 0.5]
    }).relativeTo('.relative-class').relativeToViewport().observe('.target-class',
      res.intersectionRatio // the proportion of the intersection area to the layou
      res.intersectionRect // The intersection region
      res.intersectionRect.left // the coordinates of the left boundary of the inte
      res.intersectionRect.top // the upper boundary of the intersection region
      res.intersectionRect.width // width of the intersection region
      res.intersectionRect.height // height of the intersection region
    })
  }
})
```

Note:

The intersection area with the page display area does not accurately represent the area visible to users, because the area involved in the calculation is the "layout area", which may be cropped or hidden by, for example, a node whose ancestor node has the overflow style "hidden", or obscured by, for example, a node with fixed position during drawing.

In custom components or pages that contain custom components, it is recommended to use this.createIntersectionObserver, instead of wx.createIntersectionObserver, to ensure nodes are selected within the correct area.

7. Respond to Changes in Display Area

7.1 Display Area Size

The display area refers to the area in Mini Program interface where you can arrange elements freely. By default, the size of the display area remains unchanged after the page initialization. But you can change the default size by either of the following two ways.

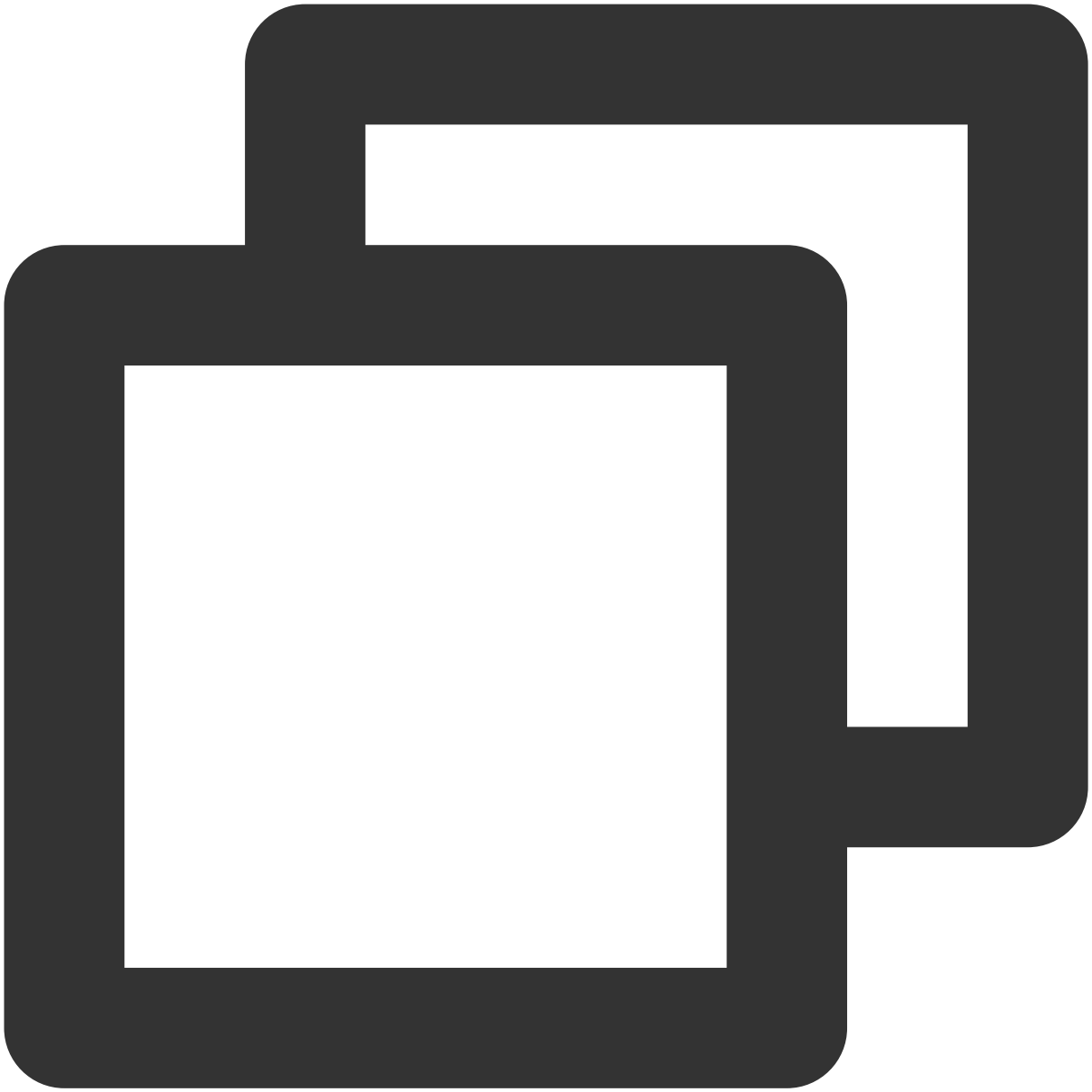
Enable Screen Rotation on a Phone

To allow the Mini Program pages to support screen rotation, set "pageOrientation": "auto" in the `window` segment of `app.json`, or configure `"pageOrientation": "auto"` in the page json file.

The following example shows how to enable screen rotation in a single page json file.

Code example:

©2013-2022 Tencent Cloud. All rights reserved.



```
{
"pageOrientation": "auto"
}
```

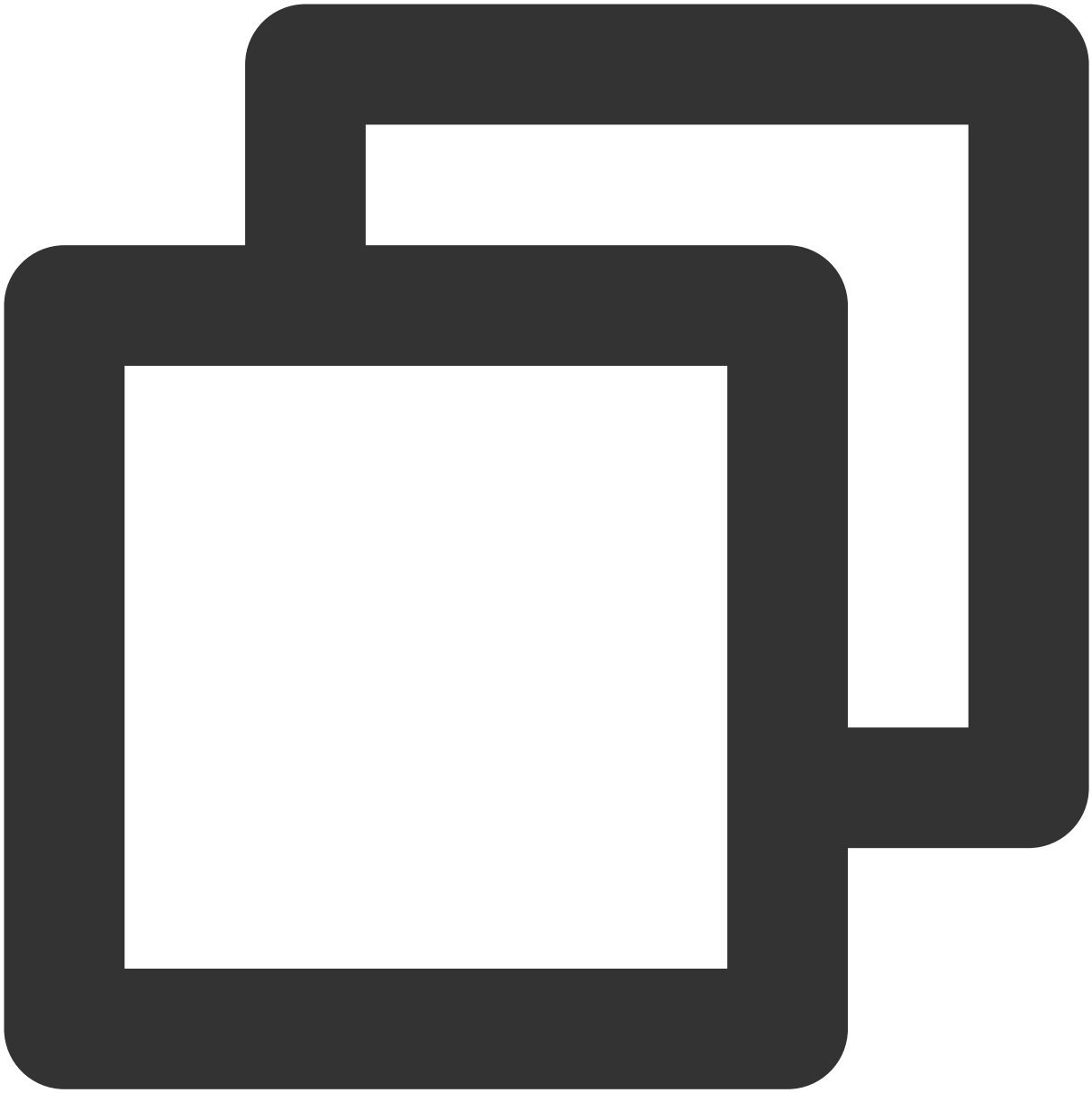
If the above declaration is added to the page, the page rotates and the display area is resized as the screen rotates. `pageOrientation` can be set to landscape, which locks the screen orientation to "Landscape".

Enable Screen Rotation on an iPad

To allow the Mini Program to support screen rotation, add "resizable": true to app.json.

Code example:

©2013-2022 Tencent Cloud. All rights reserved.



```
{
"resizable": true
}
```

If the above declaration is added to the Mini Program, the Mini Program page rotates and the display area is resized as the screen rotates.

Note：

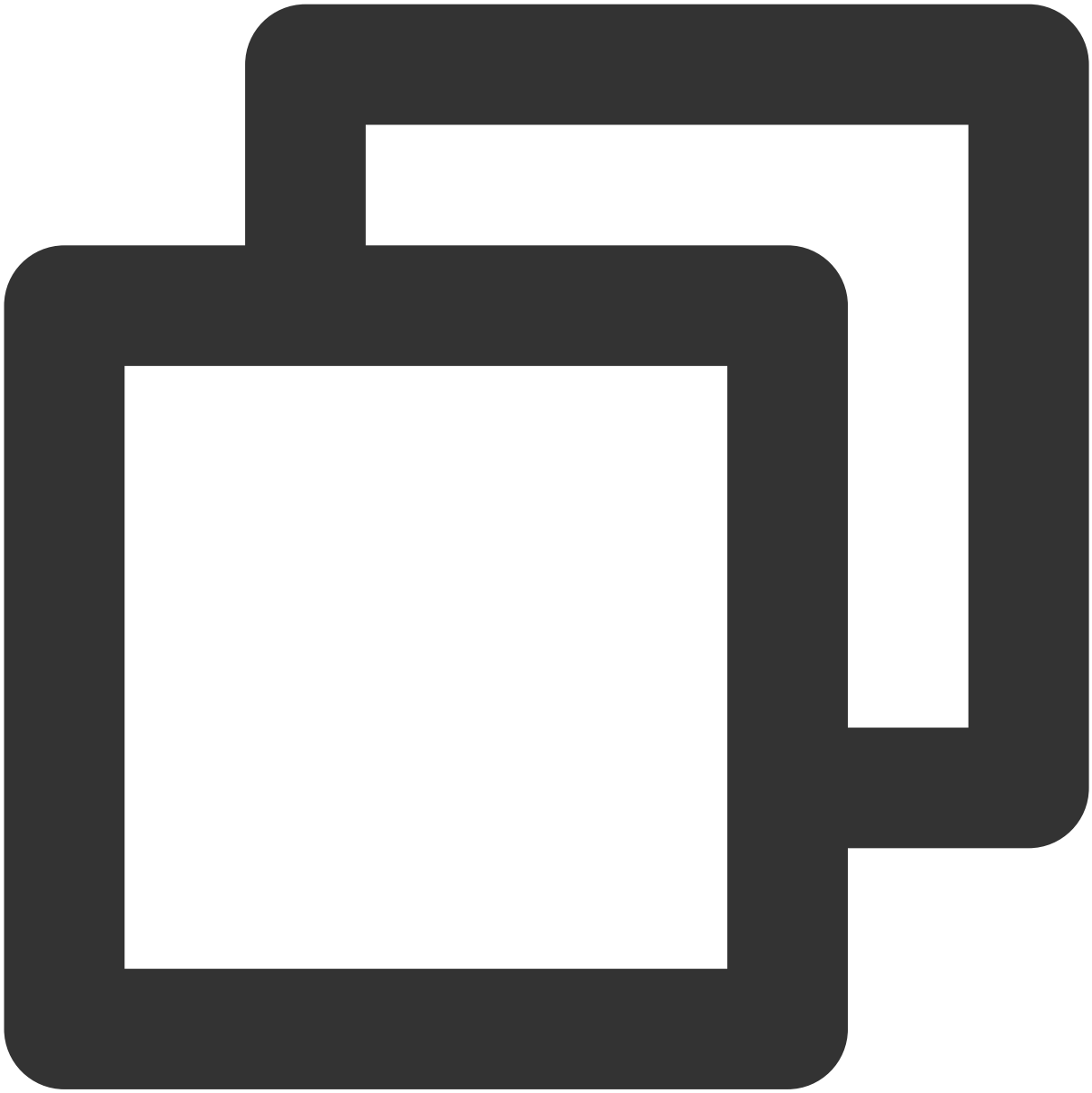
You cannot enable or disable screen rotation for a single page.

7.2 Media Query

©2013-2022 Tencent Cloud. All rights reserved.

Sometimes, the page layout varies with the size of the display area. You can use "media query" to solve most of the problems.

Code example:



```
.my-class {
  width: 40px;
}

@media (min-width: 480px) {
  /* Style rules that only work on screens 480px or wider */
  .my-class {
```

©2013-2022 Tencent Cloud. All rights reserved.

```
    width: 200px;
  }
}
```

7.3 Screen Rotation Event

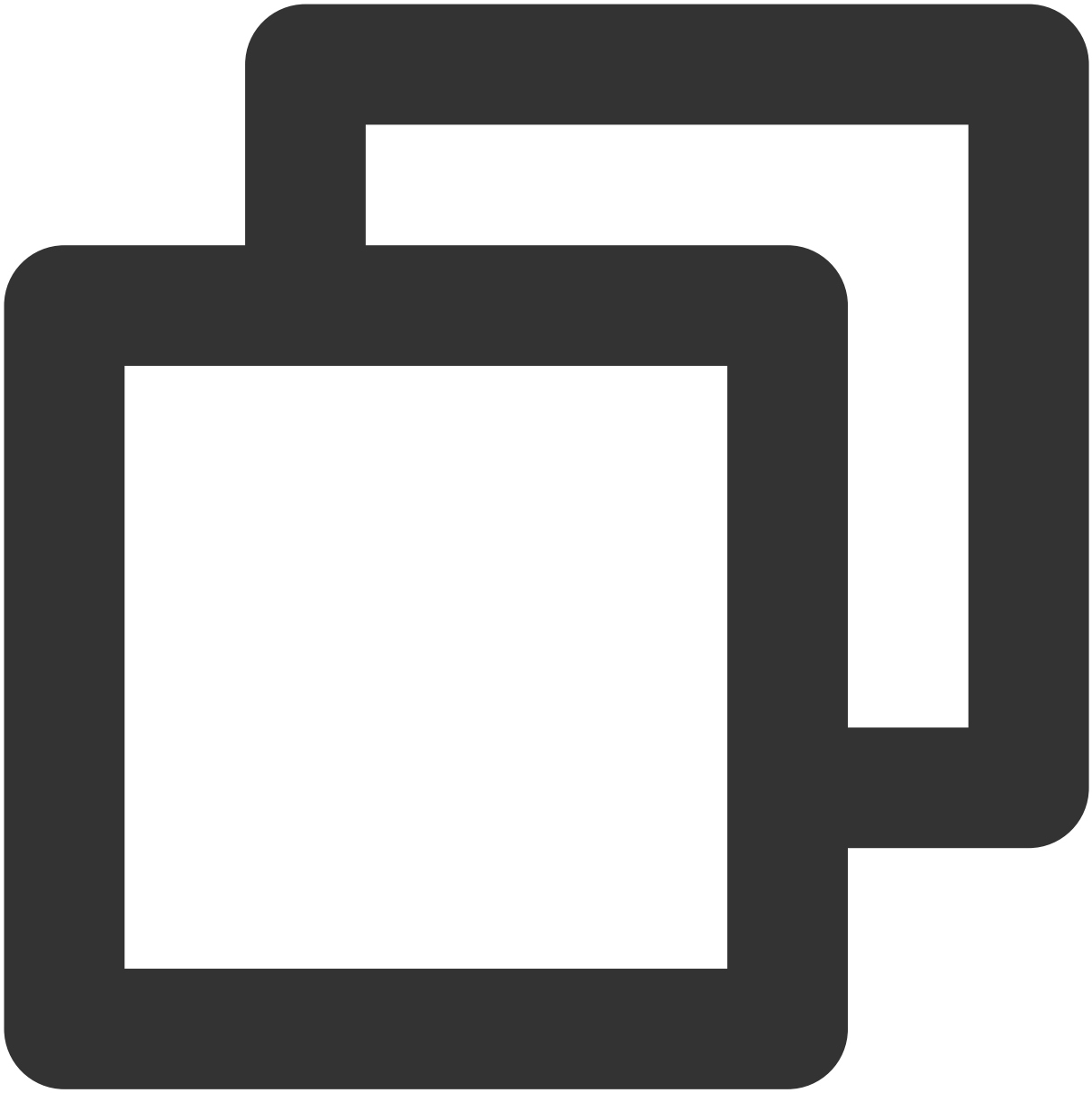
Sometimes, subtle layout changes cannot be controlled by using "media query" alone. In this case, js can be used.

You can use selectorQuery.selectViewport to read the size of the display area of the page in js.

The onResize of the page can be used to listen to the page resizing event. For custom components, you can use the "resize lifecycle" function to listen to the event. The callback function returns the size of the display area.

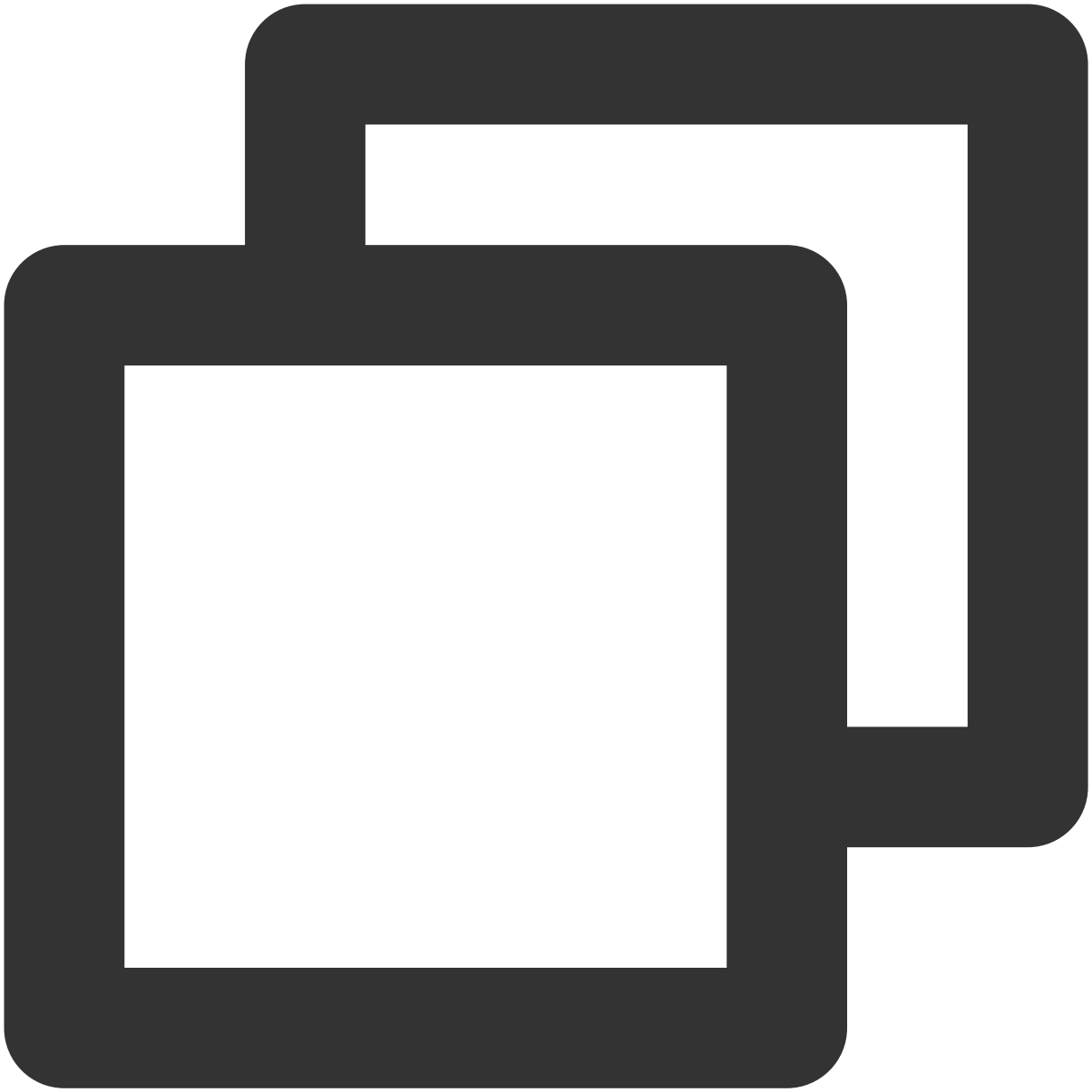
Code example:

©2013-2022 Tencent Cloud. All rights reserved.



```
Page({
  onResize(res) {
    res.size.windowWidth // The width of the new display area.
    res.size.windowHeight // the height of the new display area
  }
})
```

©2013-2022 Tencent Cloud. All rights reserved.



```
Component({
  pageLifetimes: {
    resize(res) {
    res.size.windowWidth // The width of the new display area.
    res.size.windowHeight // the height of the new display area
    }
  }
})
```

In addition, you can also use wx.onWindowResize to listen to the event (not recommended).

©2013-2022 Tencent Cloud. All rights reserved.

8. Animation

8.1 Common User Interface Animations

In Mini Programs, you can use CSS transitions and CSS animations to create simple UI animations.
During the animation creation, you can use `bindtransitionend`, `bindanimationstart`, `bindanimationioteration`, and `bindanimationend` to listen to animation events.

Event	Description
transitionend	A CSS transition ends or a step of wx.createAnimation ends
animationstart	CSS animation starts
animationiteration	A step of CSS animation ends
animationend	A CSS animation ends

Note：
These events are not bubbling events and need to be bound to the node that is being animated to take effect.
It is also possible to use the wx.createAnimation interface to dynamically create simple animation effects (the keyframe animation interface described below is recommended in the new version of the mini program base library instead).

8.2 Advanced Animations

The above animations may not be applicable to complicated scenarios.
You can dynamically adjust the "style" property of a node by responding to events with WXS. Changing the value of "style" continuously can produce animation effects. And this method can be used to dynamically generate animations based on users' touch events.
Animation effects can also be achieved by using setData to change the user interface continuously. This can change the interface arbitrarily, but usually produces a large delay or stutter, and even causes the Mini Program to freeze. In this case, you can improve performance by changing the setData of the page to the setData in Custom Components.
The following example shows how to generate a stopwatch animation using setData.

©2013-2022 Tencent Cloud. All rights reserved.

Custom Components

Last updated：2024-07-12 18:41:11

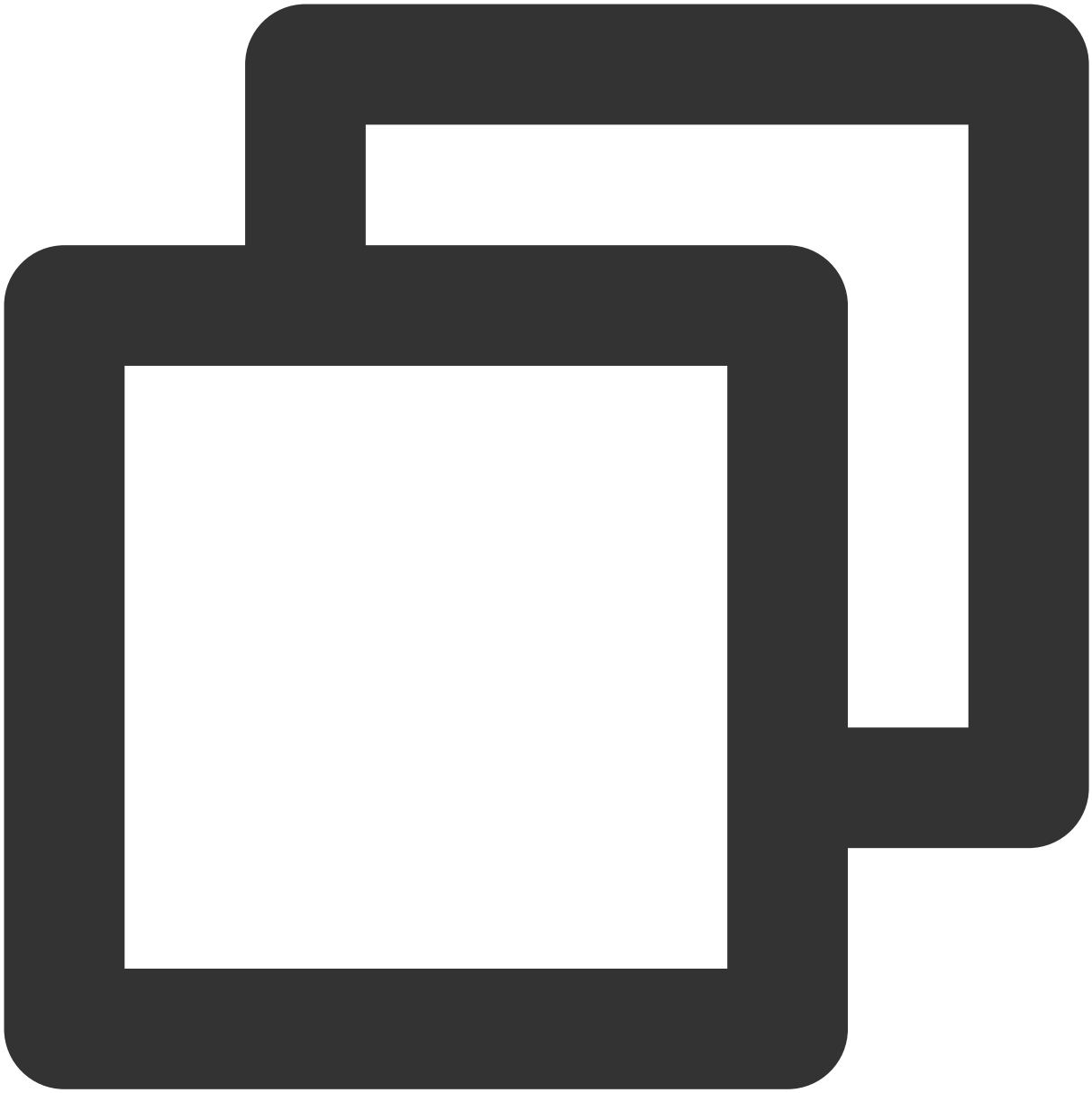
1. Introduce

Developers can abstract feature modules in a page into custom components to reuse them in different pages, or split a page into multiple low-coupling modules for easy code maintenance. Custom components are used in almost the same way as base components.

1.1 Creating Custom Component

Similar to a page, a custom component consists of `json`, `wxml`, `wxss`, and `js` files. To write a custom component, you first need to make a custom component declaration in the json file (set the files as custom `components` by setting the component field to `true`):

©2013-2022 Tencent Cloud. All rights reserved.

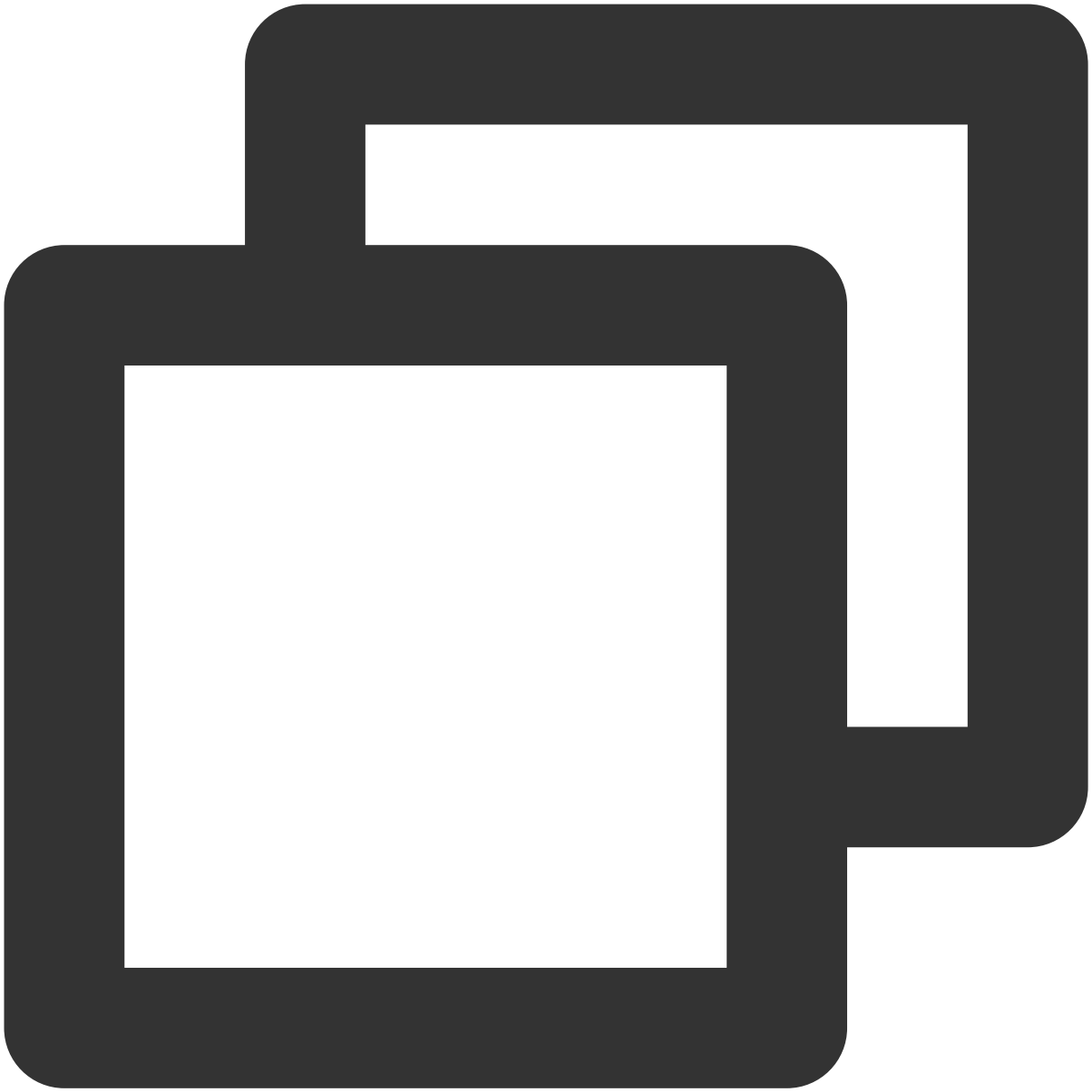


```
{
  "component": true
}
```

In the meanwhile, write a component template in the wxml file and a component style in the wxss file in a same way that you write a page. For details and considerations, see Component Template and Style.

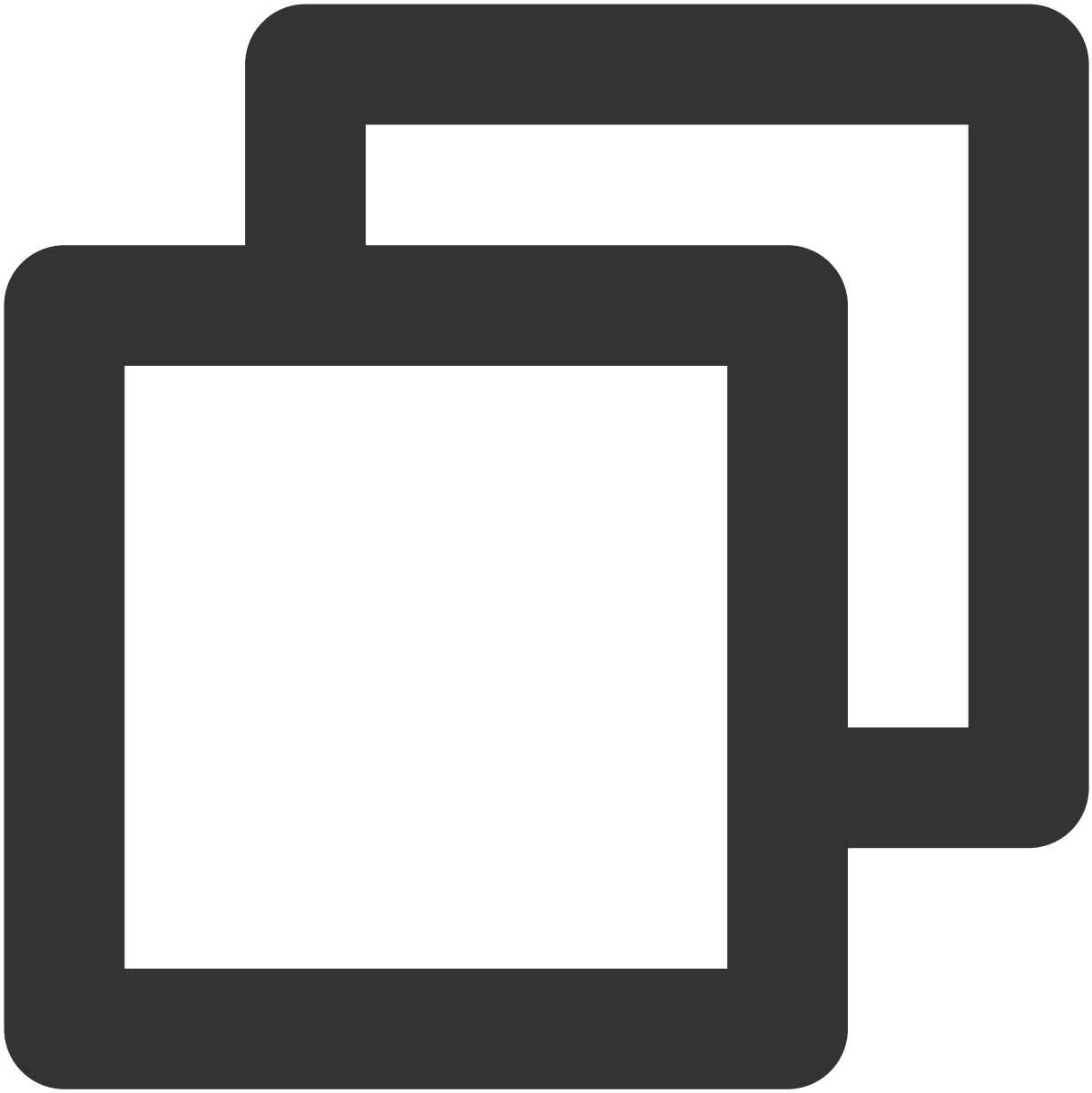
Code example:

©2013-2022 Tencent Cloud. All rights reserved.



```
<!-- This is the internal WXML structure of the custom component -->
<view class="inner">
  {{innerText}}
</view>
<slot></slot>
```

©2013-2022 Tencent Cloud. All rights reserved.



```
/*  This style only applies to this custom component  */
.inner {
  color: red;
}
```

Note：

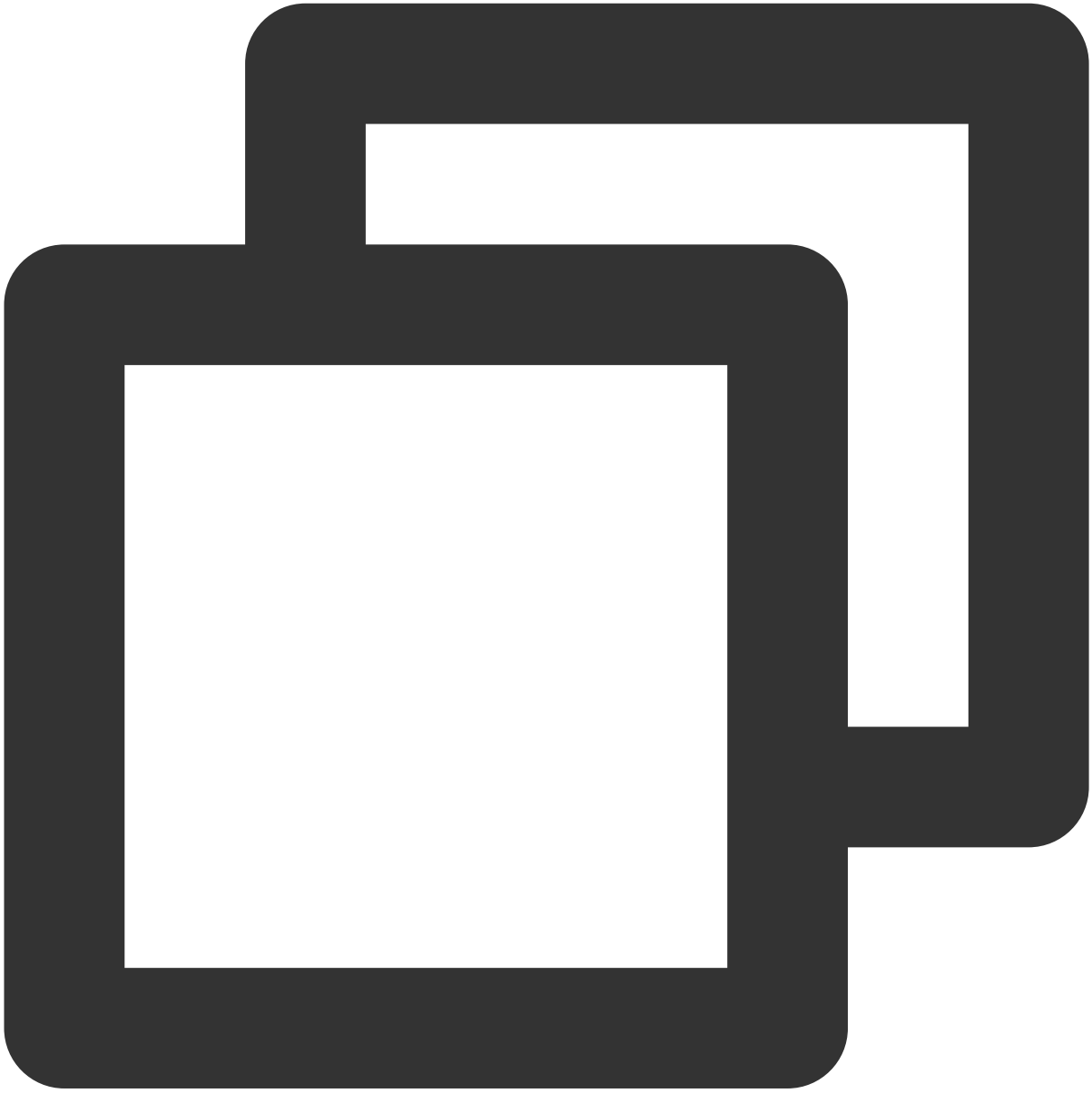
ID selectors, attribute selectors, and tag name selectors should not be used in the component's wxss.

In the custom component's js file, you need to register the component using `Component()` and provide the component's property definitions, internal data, and custom methods.

©2013-2022 Tencent Cloud. All rights reserved.

The property values and internal data of the component are used to render the component's `wxml` . The property values are passed in from outside the component. For details, see Component Constructor.

Code example:



```
Component({
  properties: {
    // The innerText property is defined here, and the property value can be specif
    innerText: {
      type: String,
      value: 'default value',
    }
```

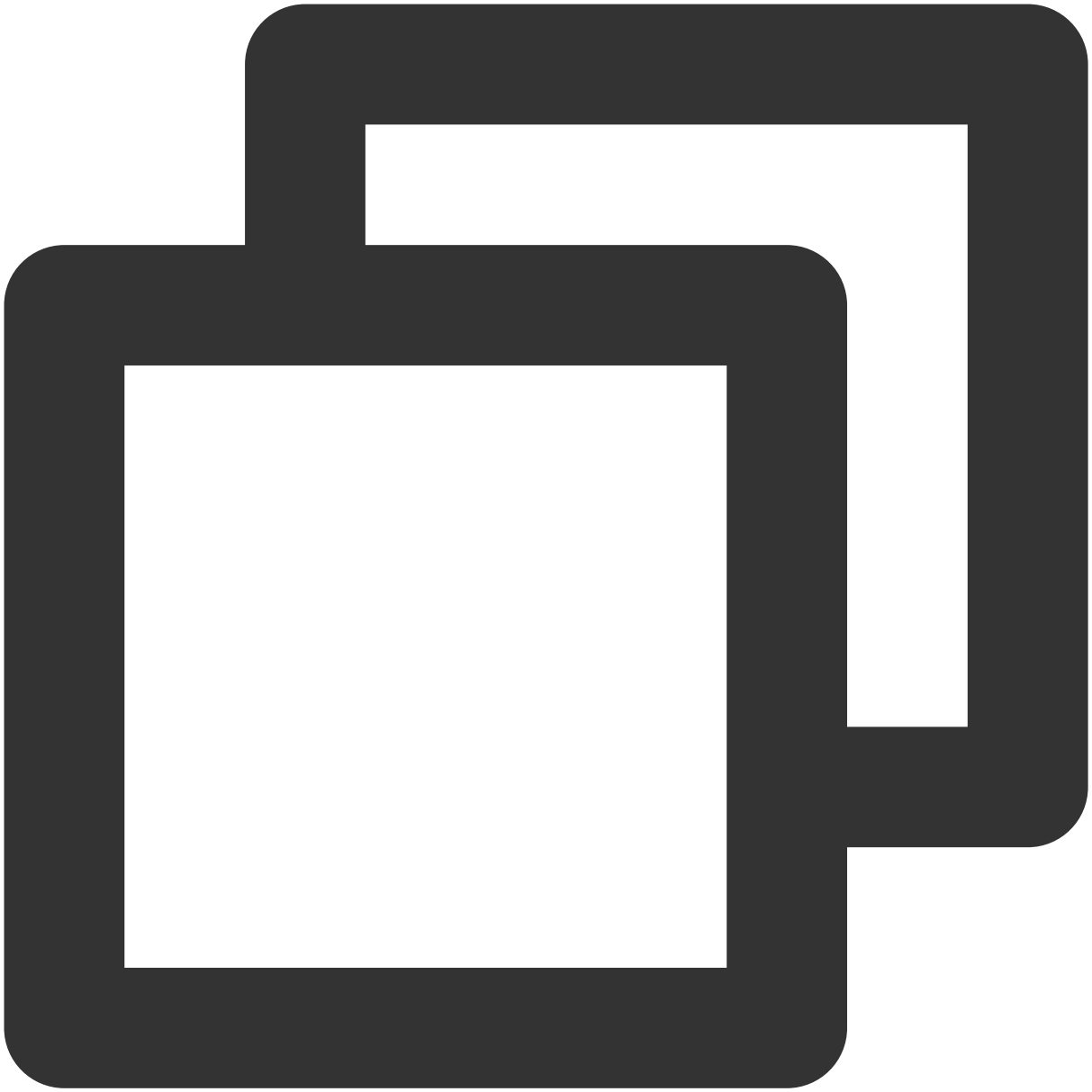
©2013-2022 Tencent Cloud. All rights reserved.

```
  },
  data: {
    // Here are some component internal data
    someData: {}
  },
  methods: {
    // Here is a custom method
    customMethod: function(){}
  }
})
```

1.2 Using Custom Component

Before using a registered custom component, make a reference declaration in the json file of the page. You need to provide the tag name of each custom component and the file path of the corresponding custom component:

©2013-2022 Tencent Cloud. All rights reserved.



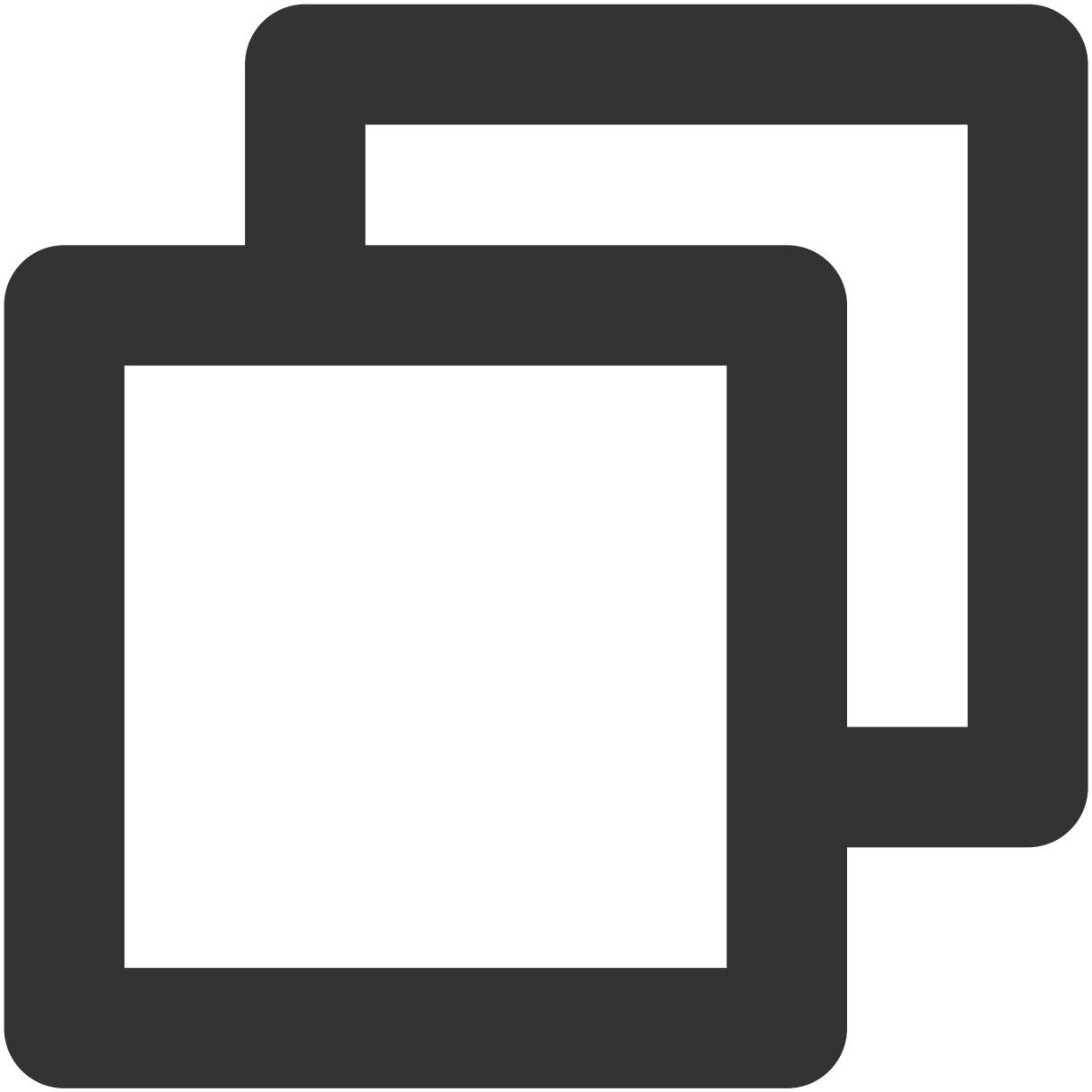
```
{
  "usingComponents": {
    "component-tag-name": "path/to/the/custom/component"
  }
}
```

As such, you can use custom components like you would with base components in the `wxml` of the page. The node name is the tag name of the custom component, and the node property is the property value passed to the component.

Note：

©2013-2022 Tencent Cloud. All rights reserved.

The usingComponents field declared in app.json is considered to be a global custom component and can be used directly in pages or custom components within the mini program without having to declare it again.
Code example:



```
<view>
  <!-- The following is a reference to a custom component -->
  <component-tag-name inner-text="Some text"></component-tag-name>
</view>
```

The custom component's `wxml` node structure, after being combined with the data, will be inserted into the reference location.

©2013-2022 Tencent Cloud. All rights reserved.

1.3 Notes

Please note the following:

Since the tag name of a WXML node can only be a combination of **lowercase letters**, **hyphens (-)**, and **underscores (_)**, the tag name of a custom component can contain these characters only.

Custom components can also reference other custom components in a similar way to referencing custom components by pages (the usingComponents field is used).

The name of the project's root directory where the custom components and pages are located cannot be prefixed with "wx-". Otherwise, an error will occur.

Note that using `usingComponents` in the page file will make the prototype of the page's `this` object slightly different, including:

The prototype that uses the `usingComponents` page differs from the prototype that does not use `usingComponents`, i.e. the result of `Object.getPrototypeOf(this)` is different.

More methods, such selectComponent, are available when `usingComponents` is used.

For performance reasons, when usingComponents is used, the content of setData is not deep copied directly, that is, this.data.field === obj after this.setData({ field: obj }) is called. (Deep copy occurs when this value is passed between components.)

For complicated pages, retest is required if the usingComponents definition field is added or removed.

2. Component Template and Style

Similar to a page, each custom component has its own `wxml` template and `wxss` style.

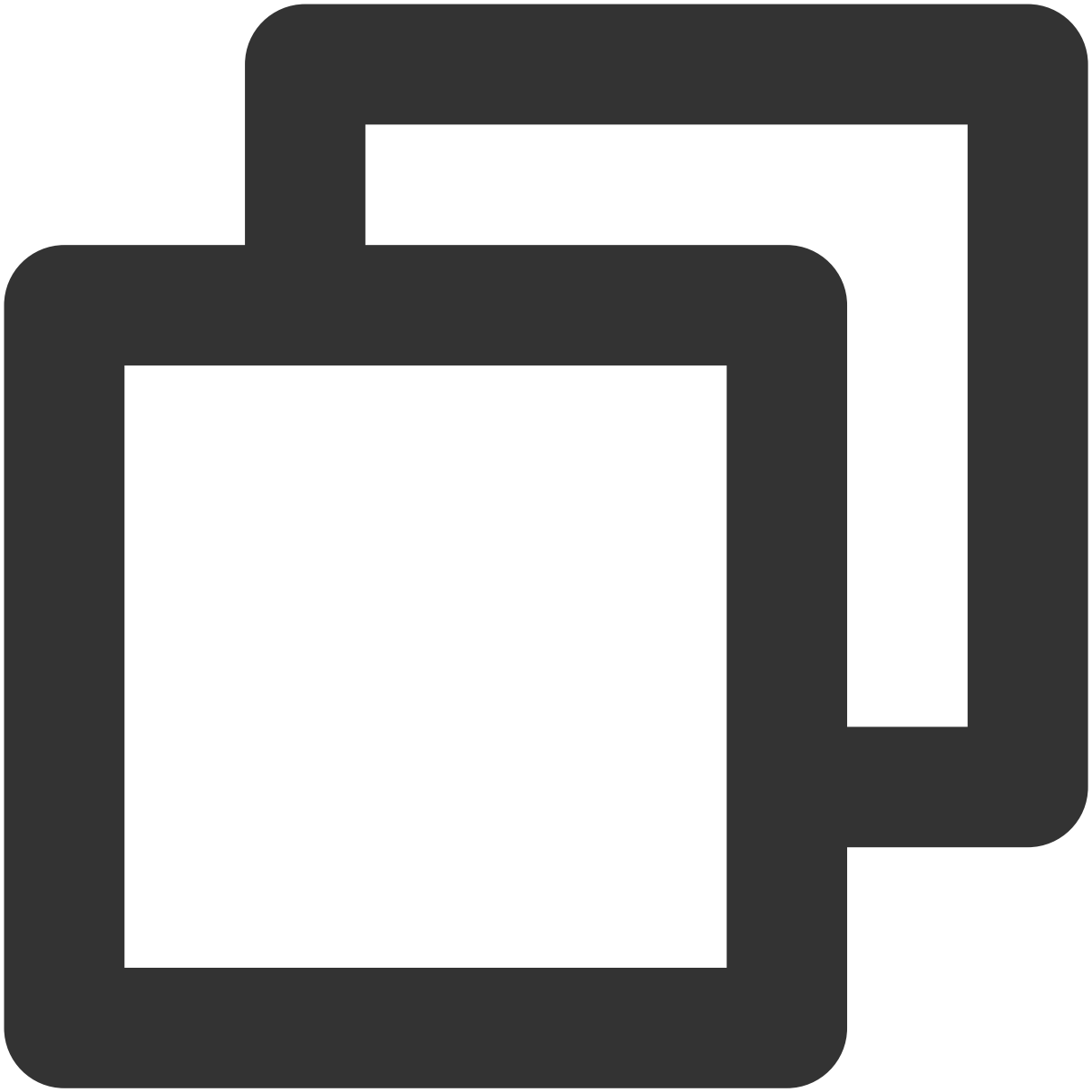
2.1 Component Template

Component templates are written in the same way as page templates. The node tree generated by binding the component template with the component data will be inserted into the location where the component is referenced.

A `<slot>` node can be provided in the component template to host the child nodes provided via component reference.

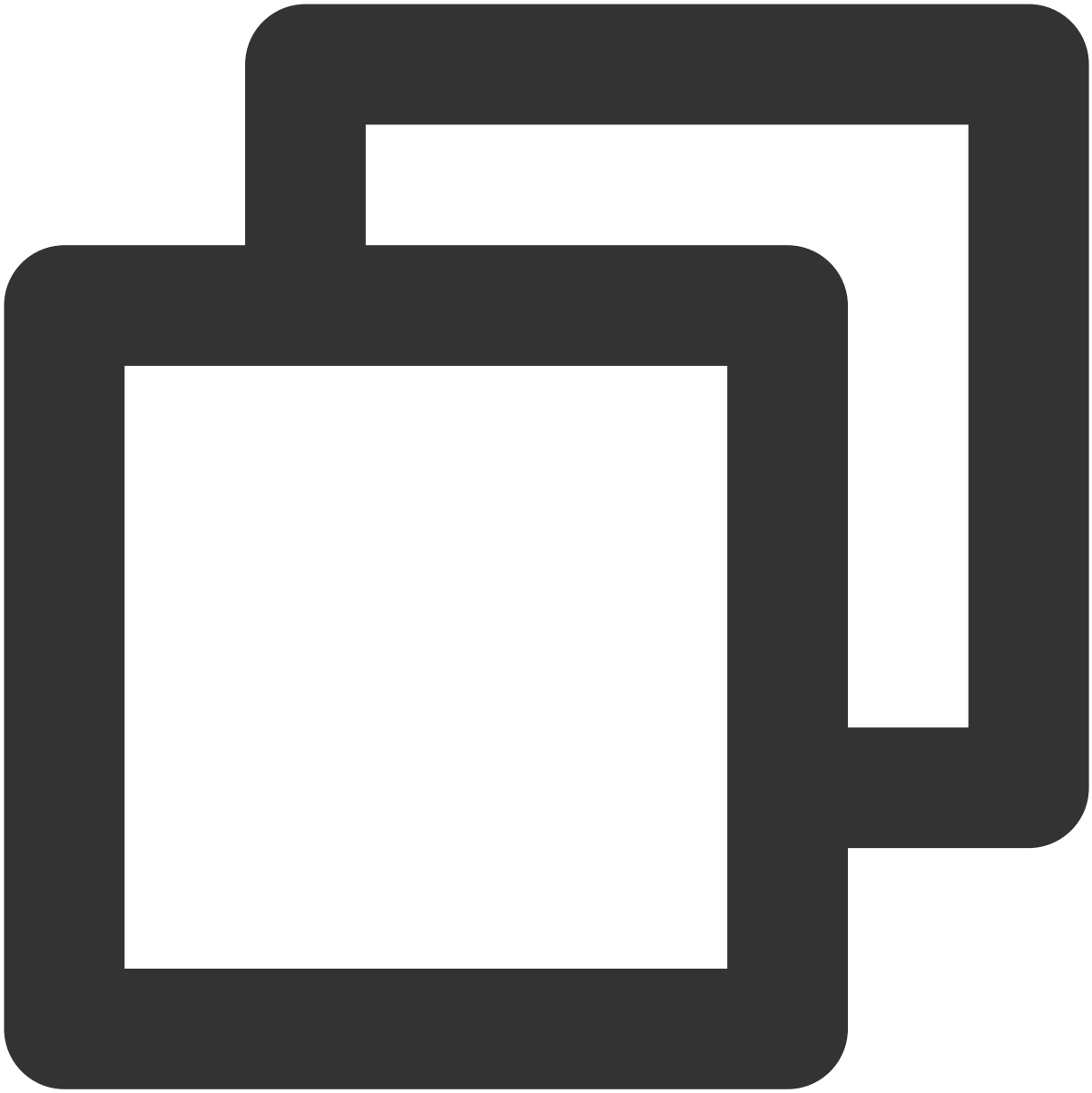
Code example:

©2013-2022 Tencent Cloud. All rights reserved.



```
<!-- Component template -->
<view class="wrapper">
  <view>This indicates the internal node of the component</view>
  <slot></slot>
</view>
```

©2013-2022 Tencent Cloud. All rights reserved.



```
<!--Page template where the component is referenced-->
<view>
  <component-tag-name>
    <!-- This part is added to the location of the  <slot>  component -->
    <view>This is inserted in the slot component</view>
  </component-tag-name>
</view>
```

Note：

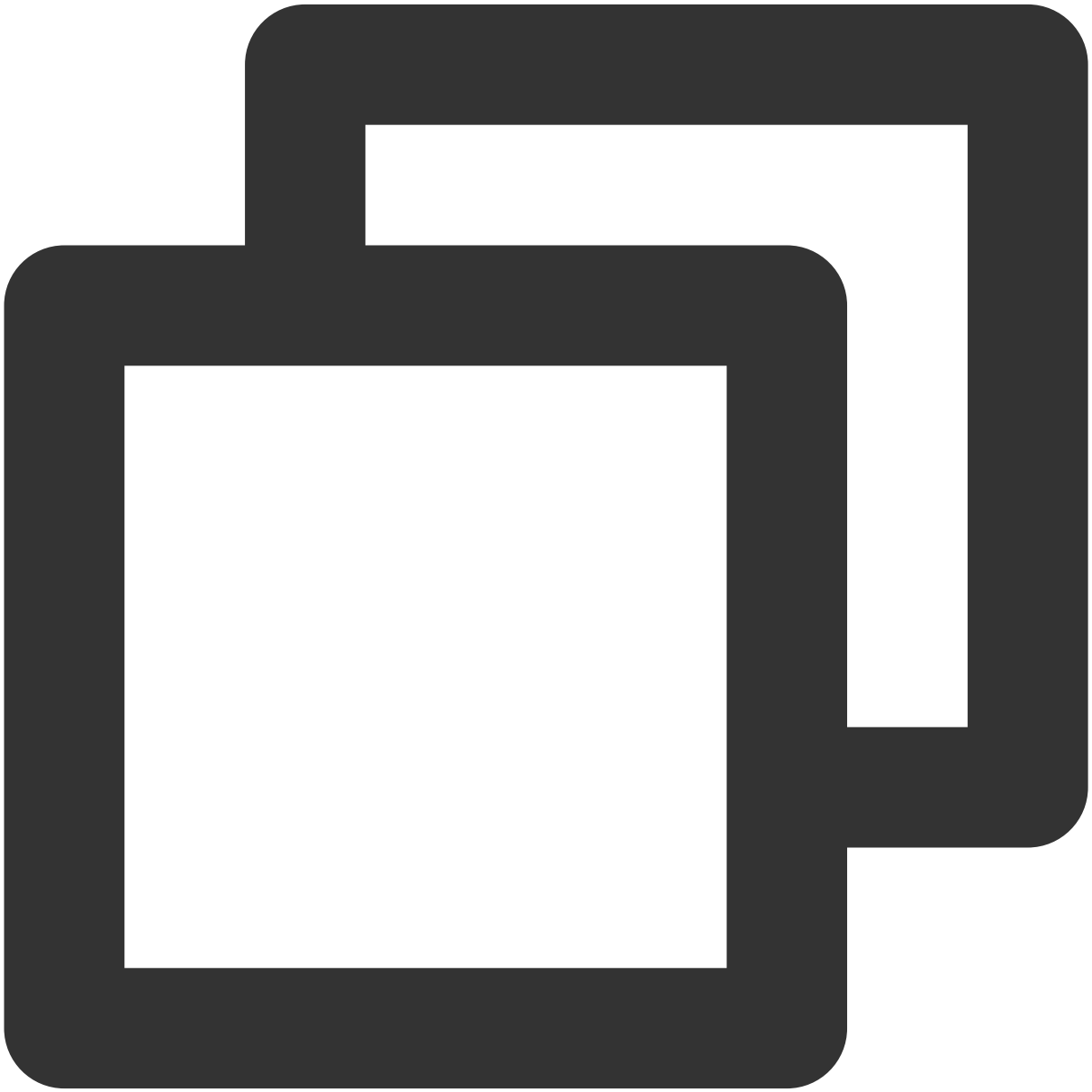
©2013-2022 Tencent Cloud. All rights reserved.

the custom component referenced in the template and the name of its corresponding node should be explicitly defined in the json file, otherwise the node will be deemed as meaningless. Besides, the node name can also be declared as an Abstract Node.

2.2 Template Data Binding

Like an ordinary WXML template, data binding can also be employed to pass dynamic data to child component properties.

Code example:



```
<!--Page template where the component is referenced-->
```

©2013-2022 Tencent Cloud. All rights reserved.

```
<view>
  <component-tag-name prop-a="{{dataFieldA}}" prop-b="{{dataFieldB}}">
    <!-- This part is added to the location of the  <slot>  component -->
    <view>This is inserted in the slot component</view>
  </component-tag-name>
</view>
```

In the above example, the component properties `propA` and `propB` receive data passed from the page, and the page can change linked data field via `setData` .

Note：

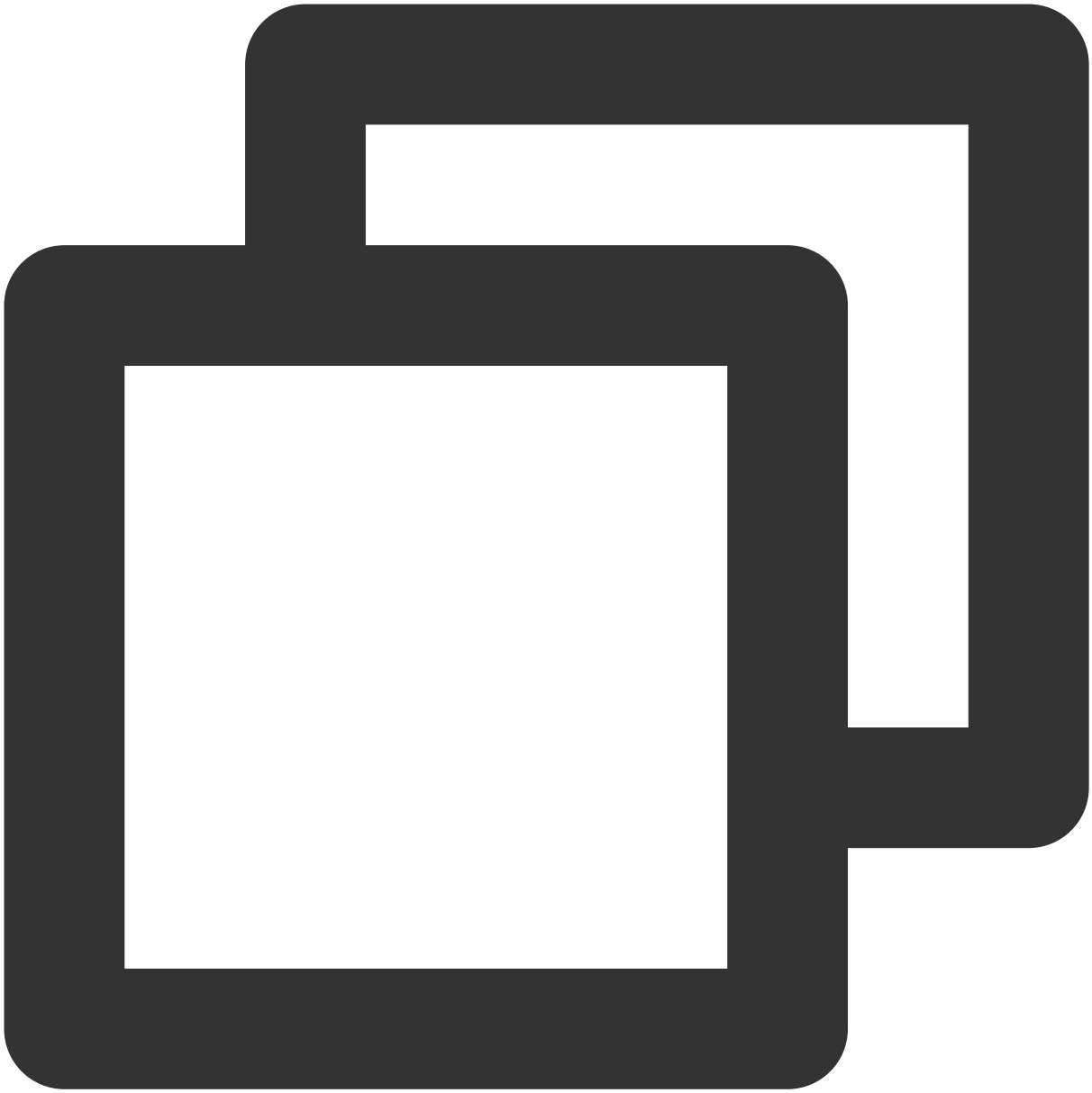
Only JSON compatible data can be passed via the above data binding.

2.3 slot of Component's WXML

The `slot` node can be included in the component's wxml to host the wxml structure provided by the component user.

By default, a component's wxml can contain only one `slot` . Multiple slots can be enabled by declaring in the component js if needed.

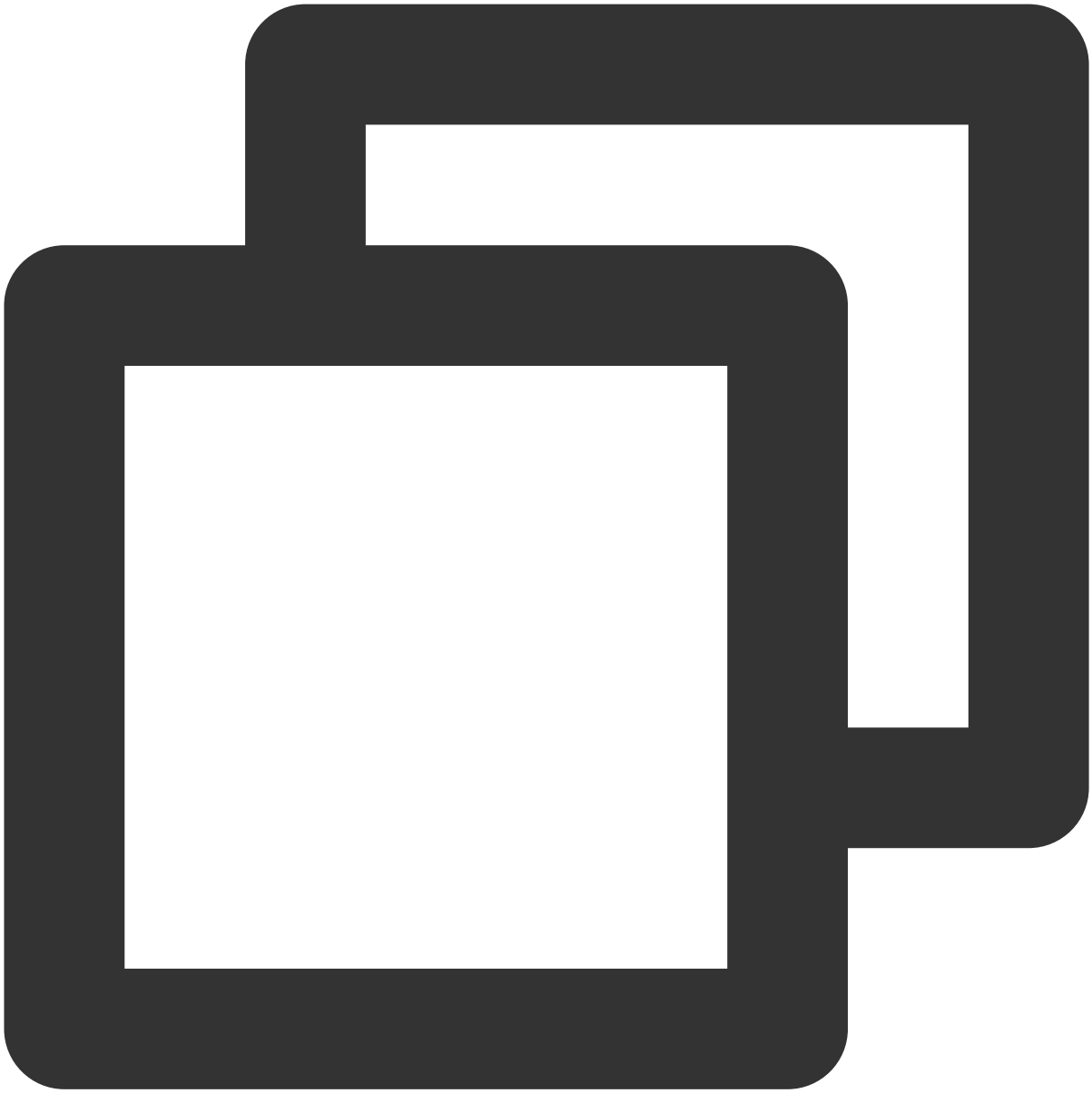
©2013-2022 Tencent Cloud. All rights reserved.



```
Component({
  options: {
    multipleSlots: true // Enable multiple slots in the options of component defini
  },
  properties: { /* ... */ },
  methods: { /* ... */ }
})
```

In this case, multiple `slots` can be used in the wxml of this component, each with a unique `name` .

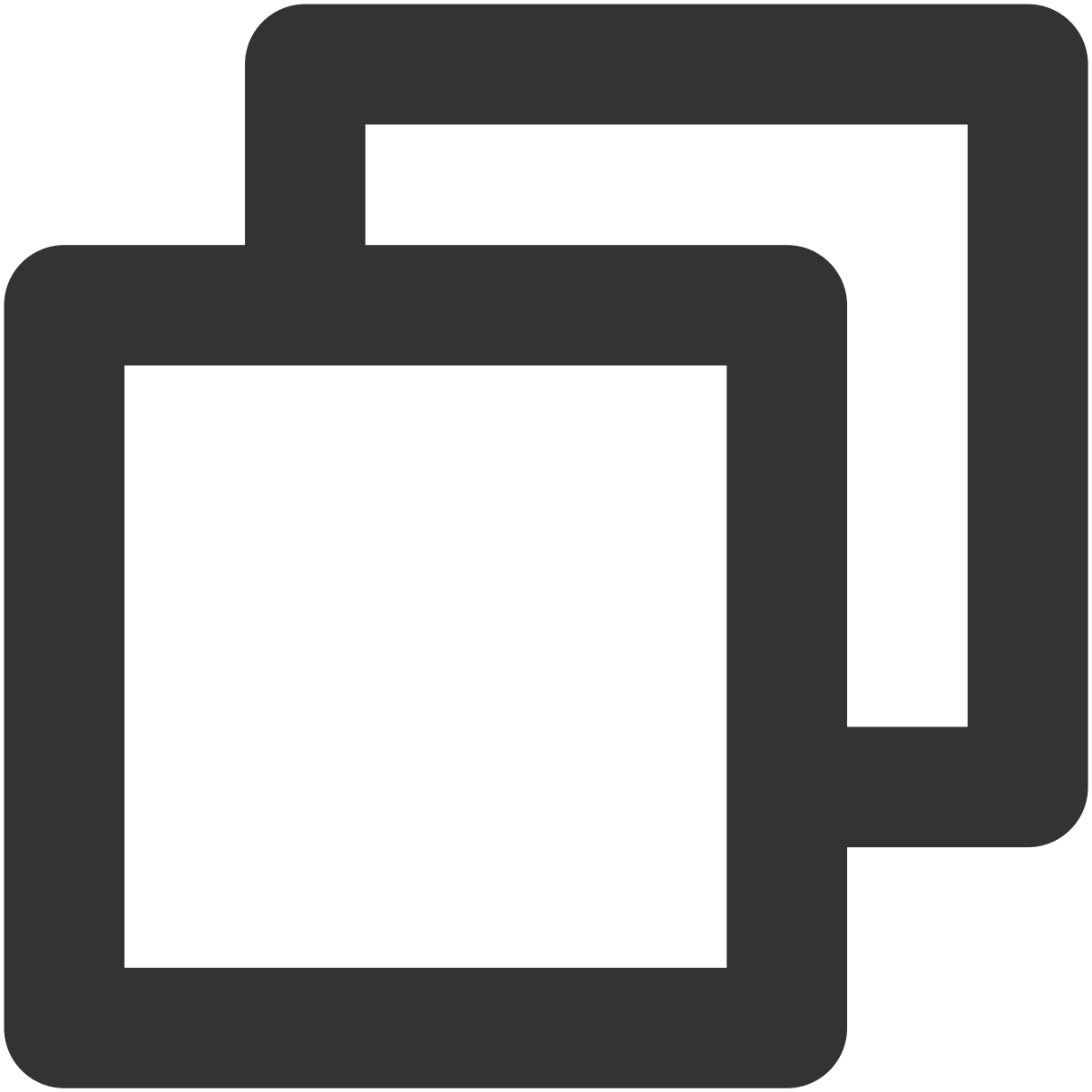
©2013-2022 Tencent Cloud. All rights reserved.



```
<!-- Component template -->
<view class="wrapper">
  <slot name="before"></slot>
  <view>This indicates the internal details of the component</view>
  <slot name="after"></slot>
</view>
```

You can use the slot property to insert nodes into different slots.

©2013-2022 Tencent Cloud. All rights reserved.



```
<!--Page template where the component is referenced-->
<view>
  <component-tag-name>
    <!-- This part is added to the location of the  <slot name="before">  component
    <view slot="before">This is inserted in the slot name="before" component</view>
    <!-- This part is added to the location of the  <slot name="after">  component
    <view slot="after">This is inserted in the slot name="after" component</view>
  </component-tag-name>
</view>
```

©2013-2022 Tencent Cloud. All rights reserved.

2.4 Component Style

The style of a component for the wxss file is only valid for the nodes in the component's wxml. Note the following when writing a component style:

For the components and the pages that reference the components, the class selector, instead of the id selector (`#a`), property selector (`[a]`) and tag name selector, should be used;

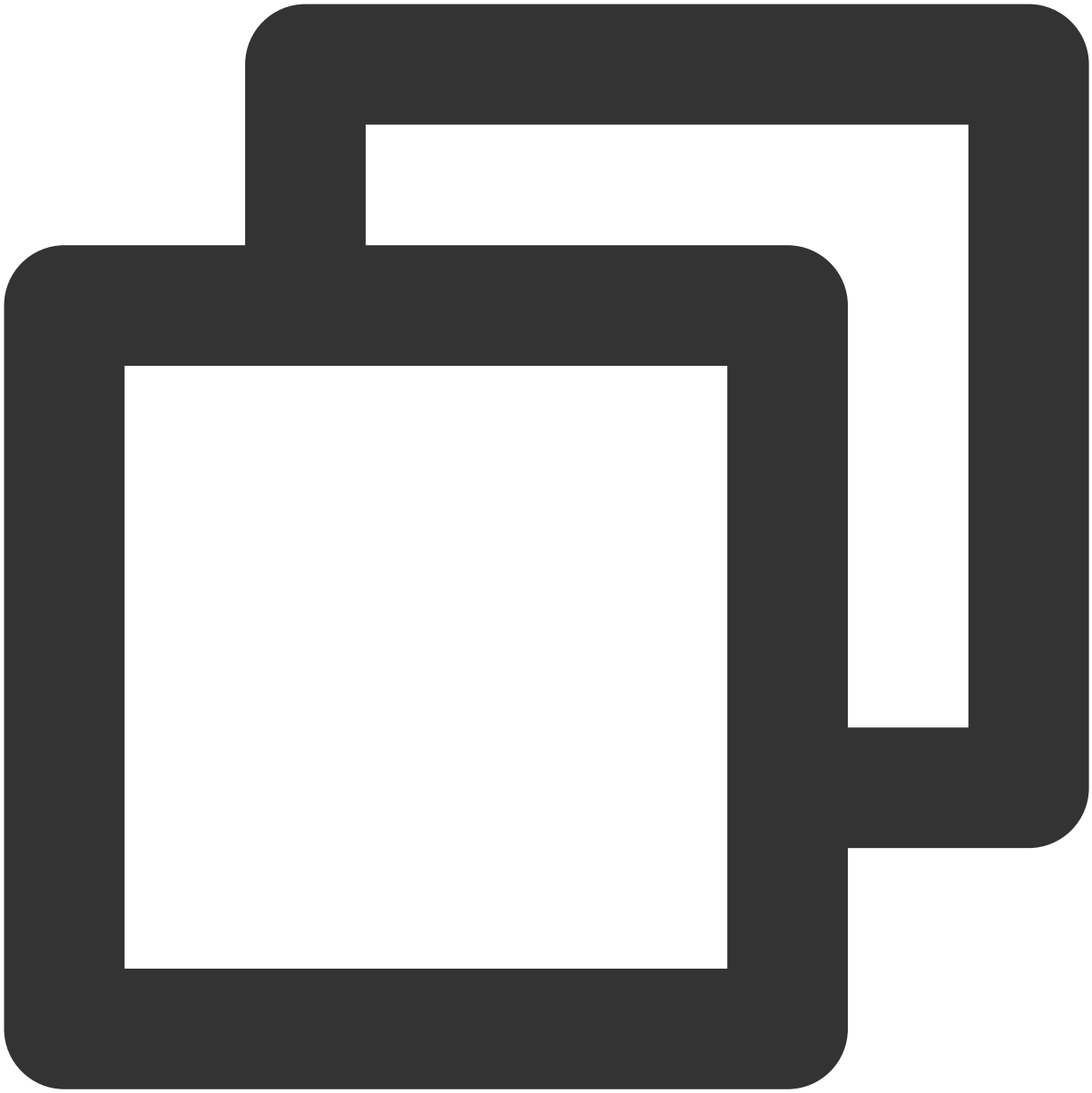
Unexpected behaviors may occur with the descendant selector (`.a .b`) used in the components and the pages that reference the components in extreme cases. If so, avoid using it;

The child selector (`.a>.b`) can be only used for the view component and its child nodes; unexpected behaviors may occur if it is used in other components;

Inheritance styles, such as `font` and `color` , are inherited into the components from outside the components;

In addition to inheritance styles, the styles in `app.wxss` and the styles of the pages that reference the components are invalid for custom components (unless the component style isolation option is changed).

©2013-2022 Tencent Cloud. All rights reserved.

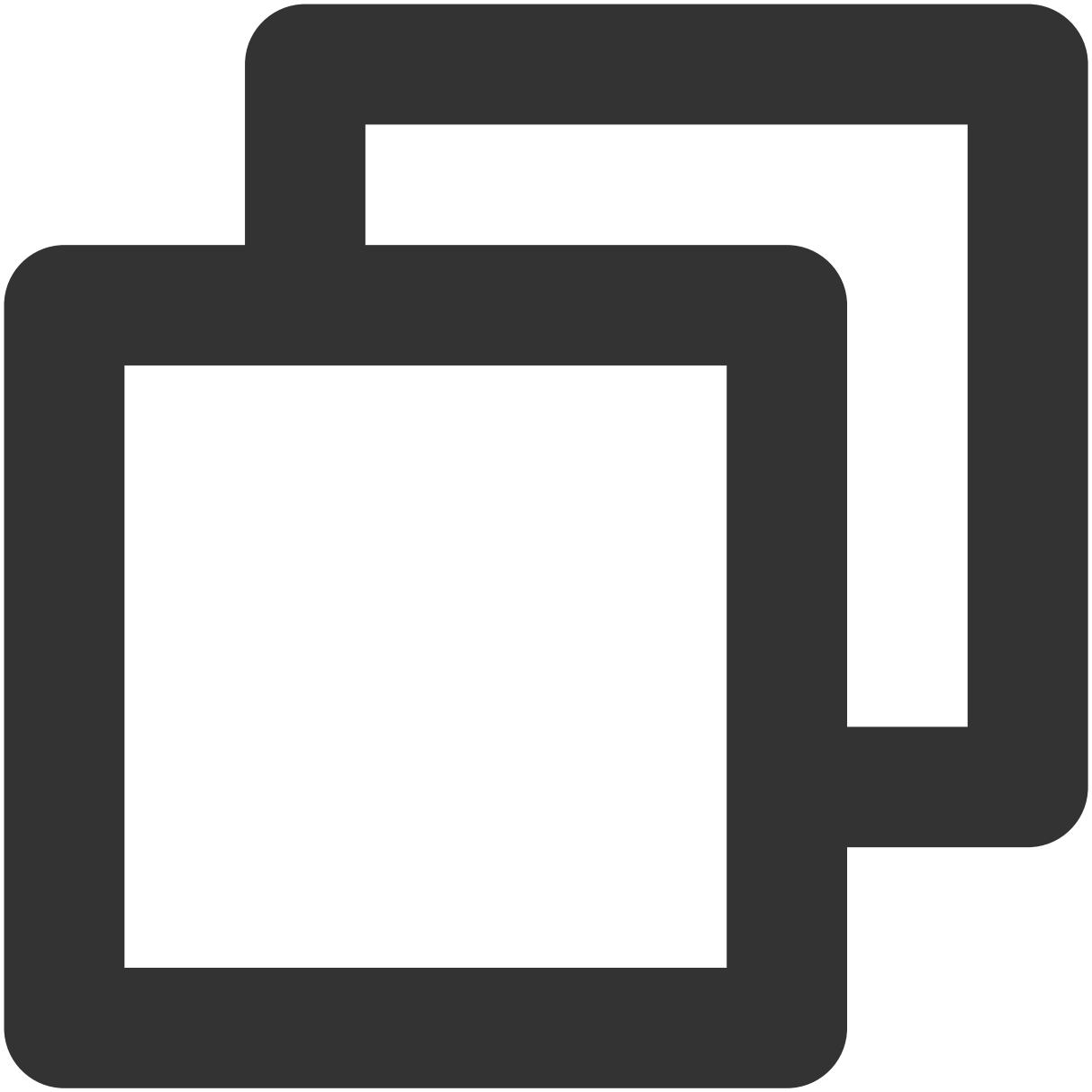


```
#a { } /* Cannot be used in components */
[a] { } /* Cannot be used in components */
button { } /* Cannot be used in components */
.a > .b { } /* Invalid unless  .a  is  view  component node */
```

In addition, a default style can be specified for the node where the component resides by using the `:host` selector

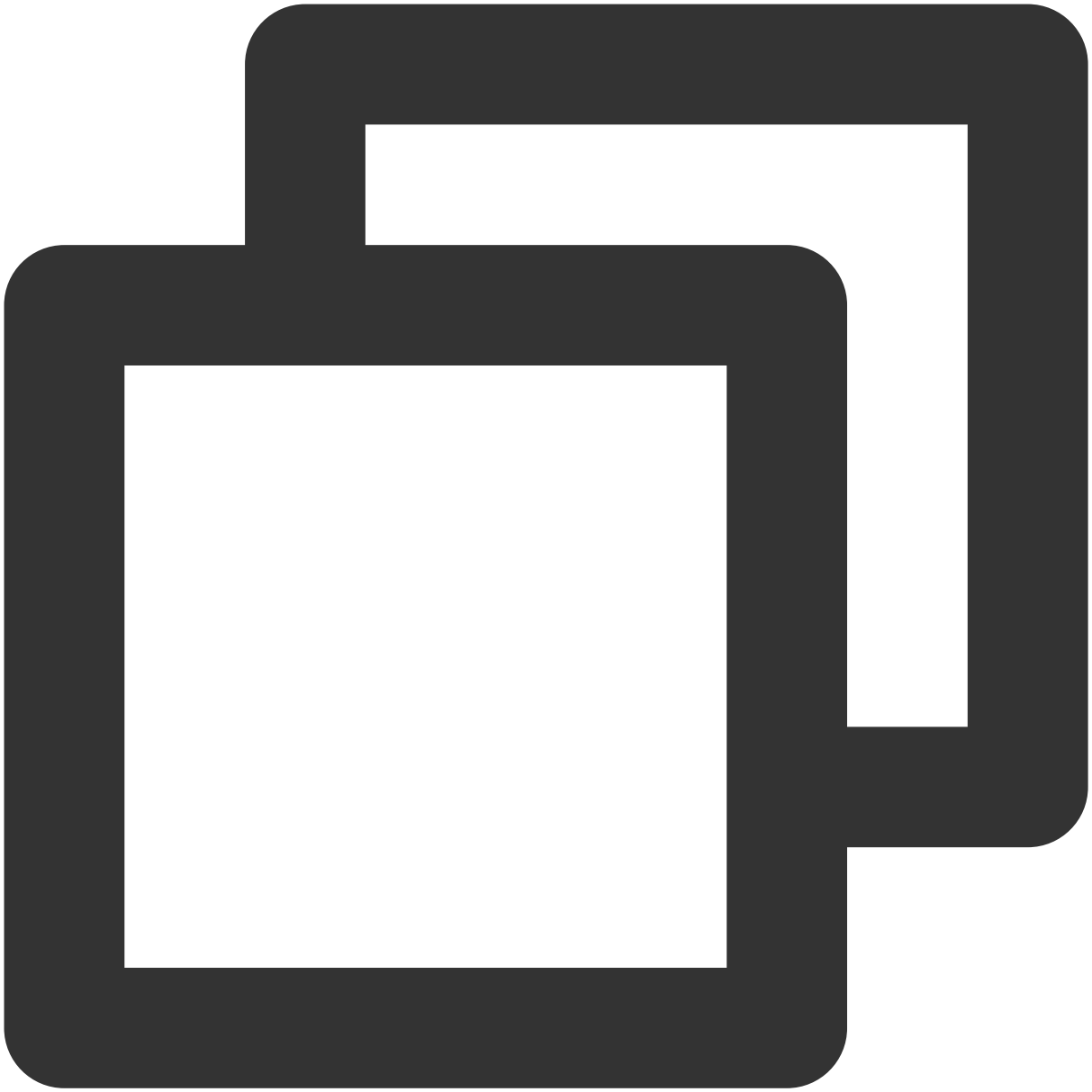
Code example:

©2013-2022 Tencent Cloud. All rights reserved.



```
/* Component custom-component.wxss */
:host {
  color: yellow;
}
```

©2013-2022 Tencent Cloud. All rights reserved.



```
<!-- page's WXML -->
<custom-component>The text here is highlighted in yellow</custom-component>
```

2.5 Component Style Isolation

Sometimes a component wants to accept incoming style classes. In this case, several external style classes can be defined in the Component using the externalClasses definition segment.
This feature can be used to implement a hover-class property similar to that of the view component: the page can provide a style class that gives the view its hover-class, which is itself written in the page rather than in the view

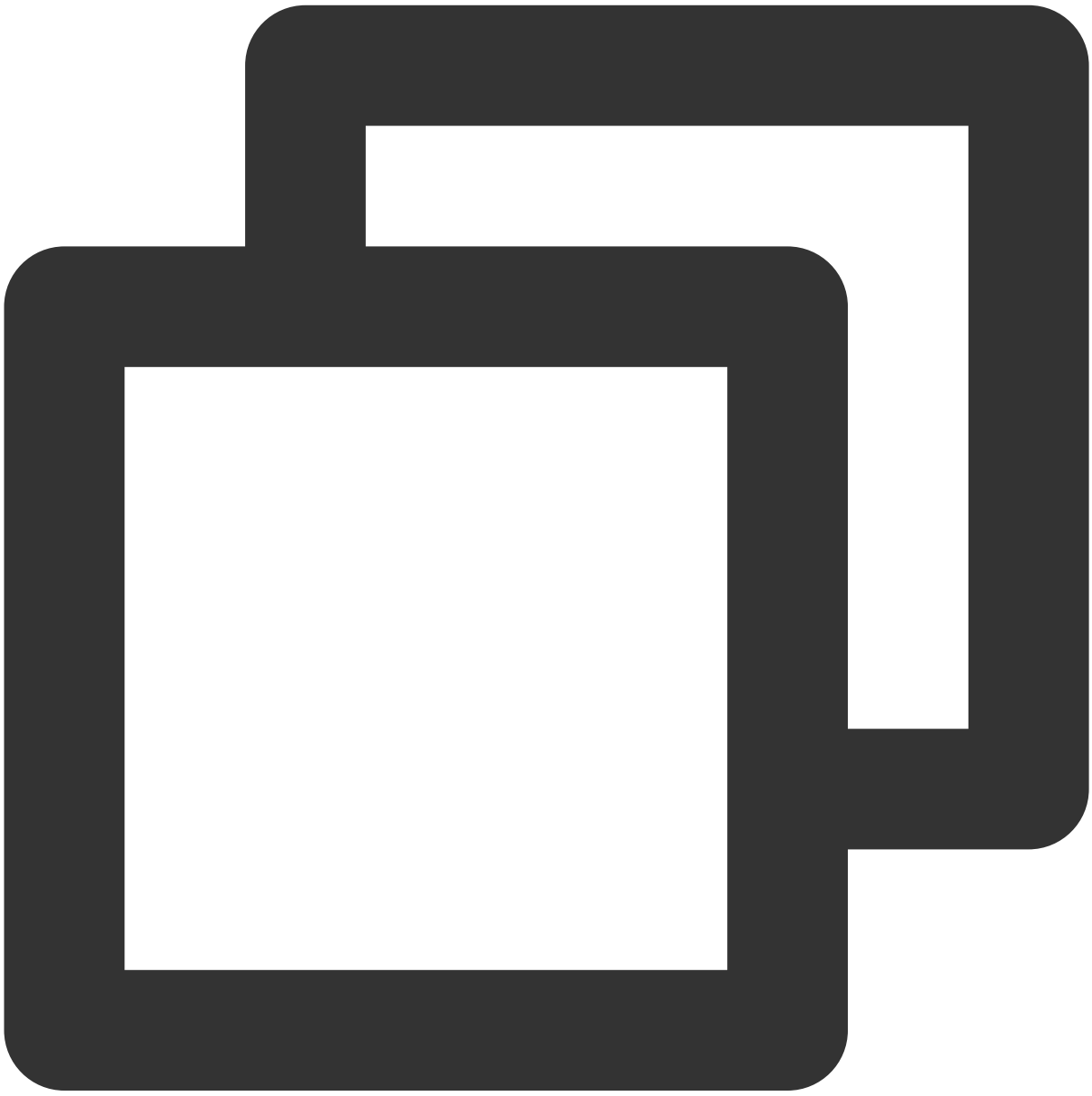
©2013-2022 Tencent Cloud. All rights reserved.

component's implementation.

Note：

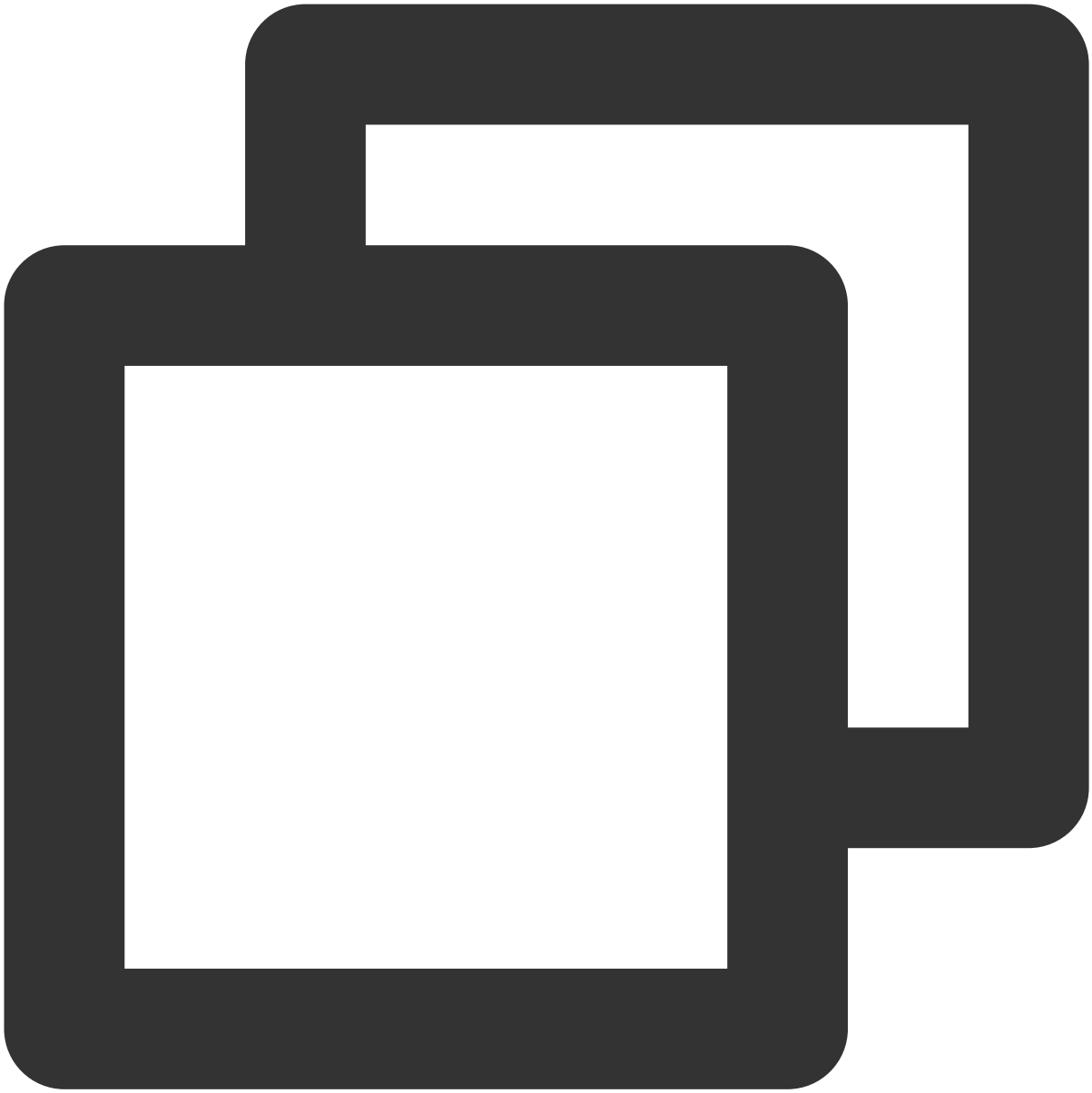
When using a normal style class and an external style class on the same node, the priority of both classes is undefined, so it is best to avoid this situation.

Code example:



```
/* Component custom-component.js */
Component({
  externalClasses: ['my-class']
})
```

©2013-2022 Tencent Cloud. All rights reserved.

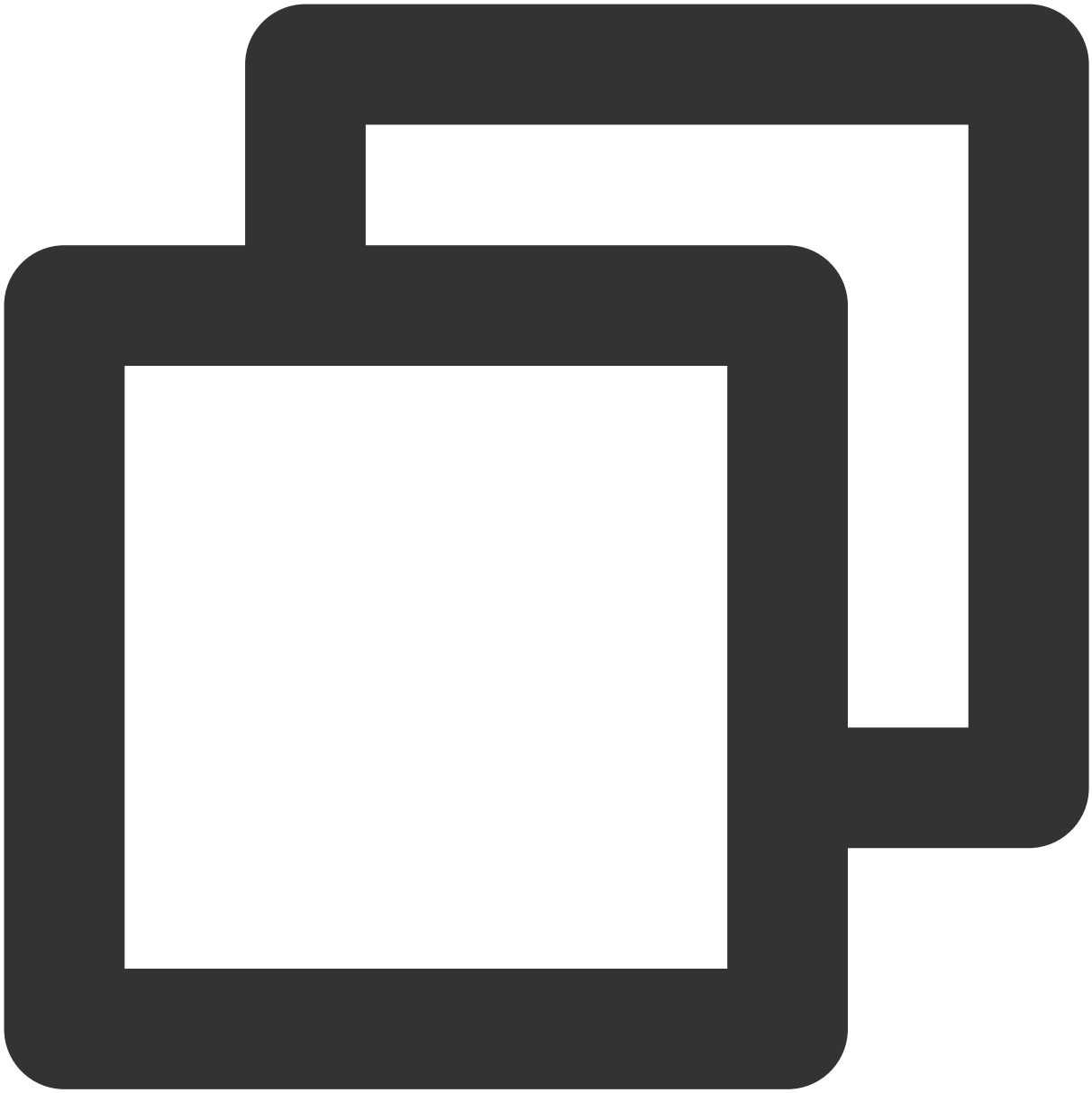


```
<!-- Component  custom-component.wxml -->
<custom-component class="my-class">The colour of this text is determined by the cla
```

In this way, the user of the component can specify the class corresponding to this style class, just like with normal properties.

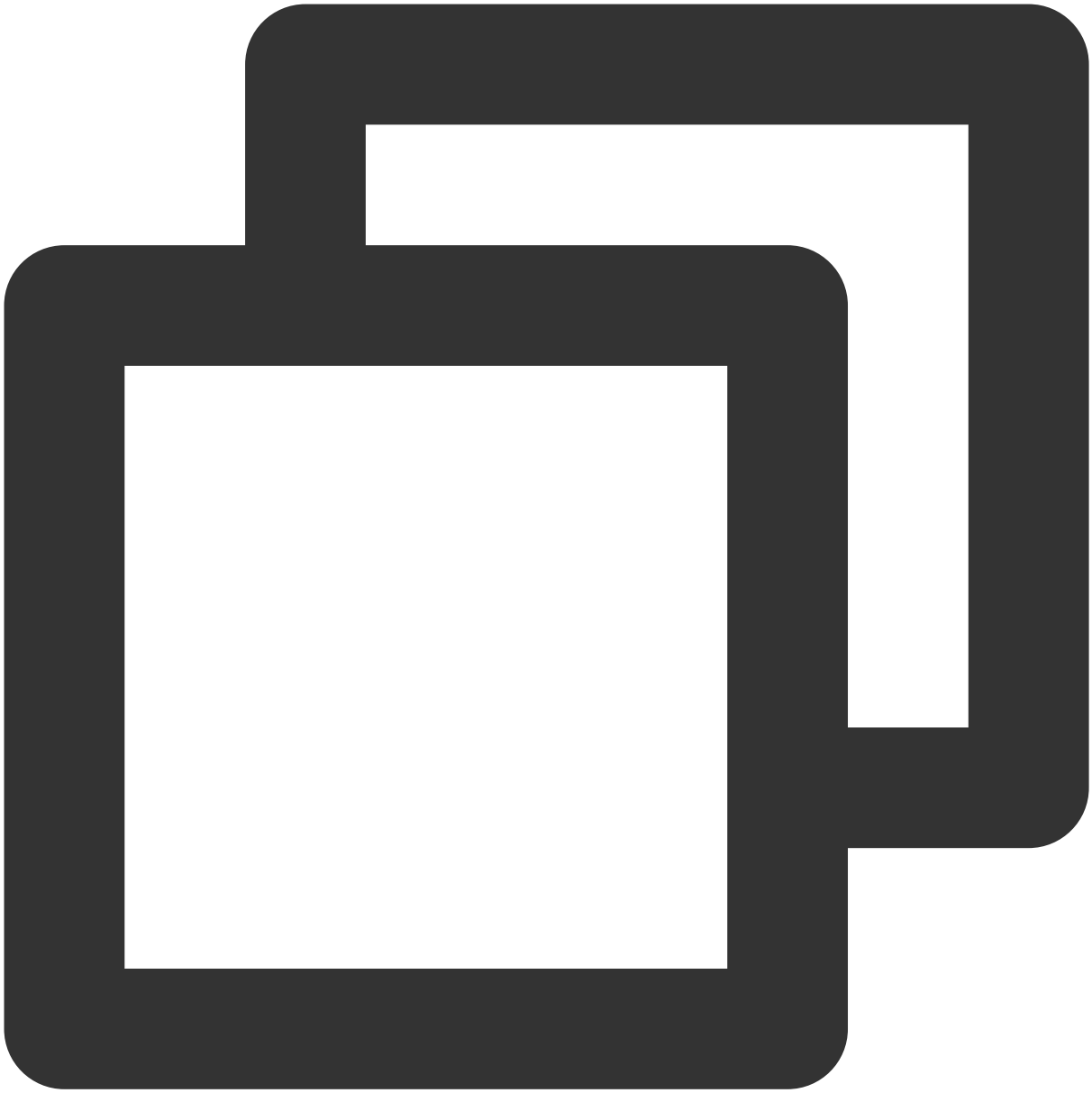
Code example:

©2013-2022 Tencent Cloud. All rights reserved.



```
<!-- page's WXML -->
<custom-component my-class="red-text" />
<custom-component my-class="large-text" />
```

©2013-2022 Tencent Cloud. All rights reserved.



```
.red-text {
  color: red;
}
.large-text {
  font-size: 1.5em;
}
```

2.6 Making components accept global styles

©2013-2022 Tencent Cloud. All rights reserved.

By default, the styling of a custom component is only affected by the custom component wxss. Except in the following two cases:

The app.wxss or the page's wxss uses tag name selectors (or some other special selectors) to directly specify styles that affect the page and the component in its entirety. This is generally not a recommended practice.

Activating the addGlobalClass option on a specific custom component makes it possible for that custom component to be affected by all style definitions in app.wxss or the page's wxss.

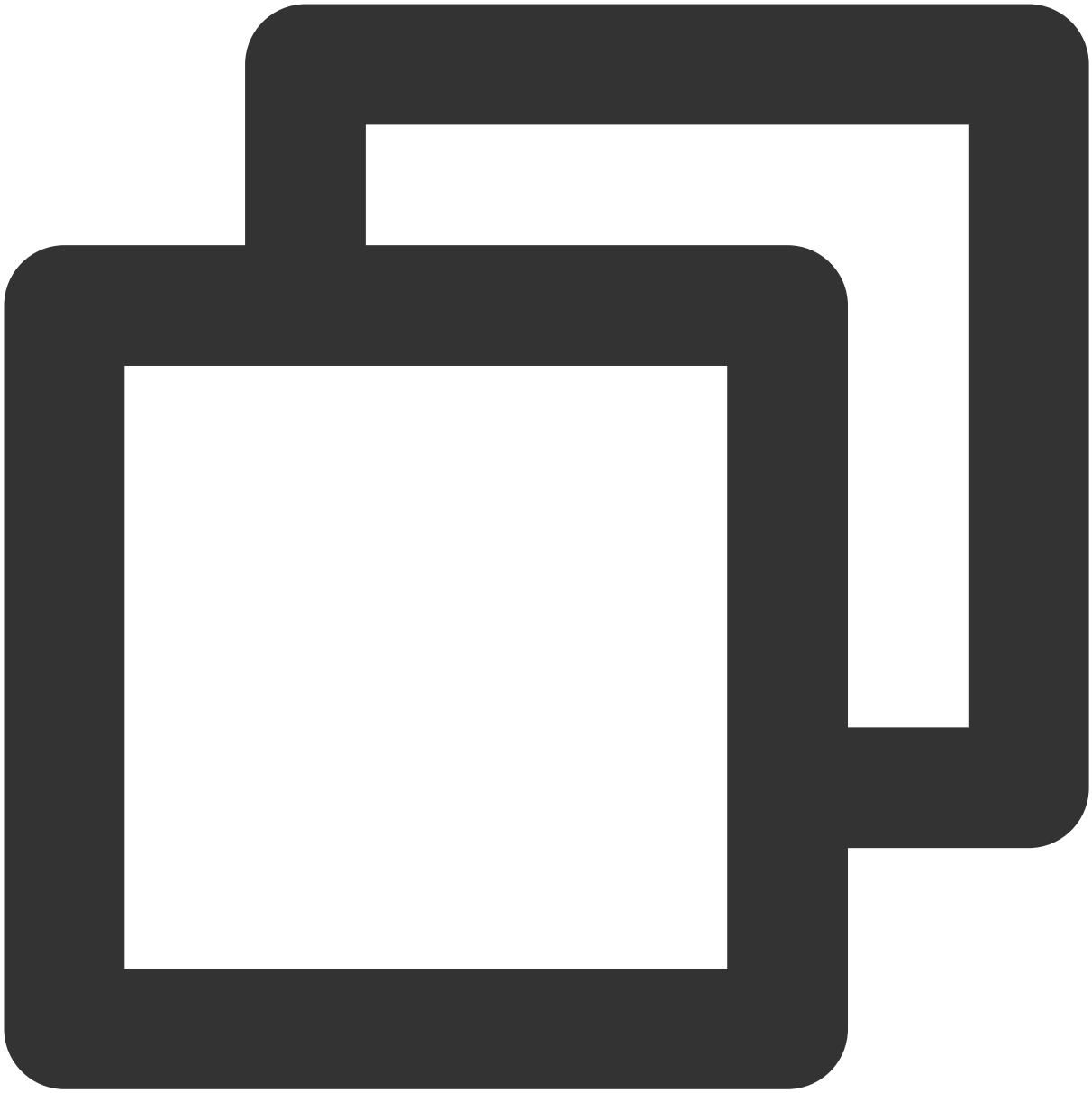
To activate the addGlobalClass option, simply set the options.addGlobalClass field to true in the Component constructor.

Note：

When the `addGlobalClass` option is activated, there is a risk of external styles contaminating the component style, so please choose carefully.

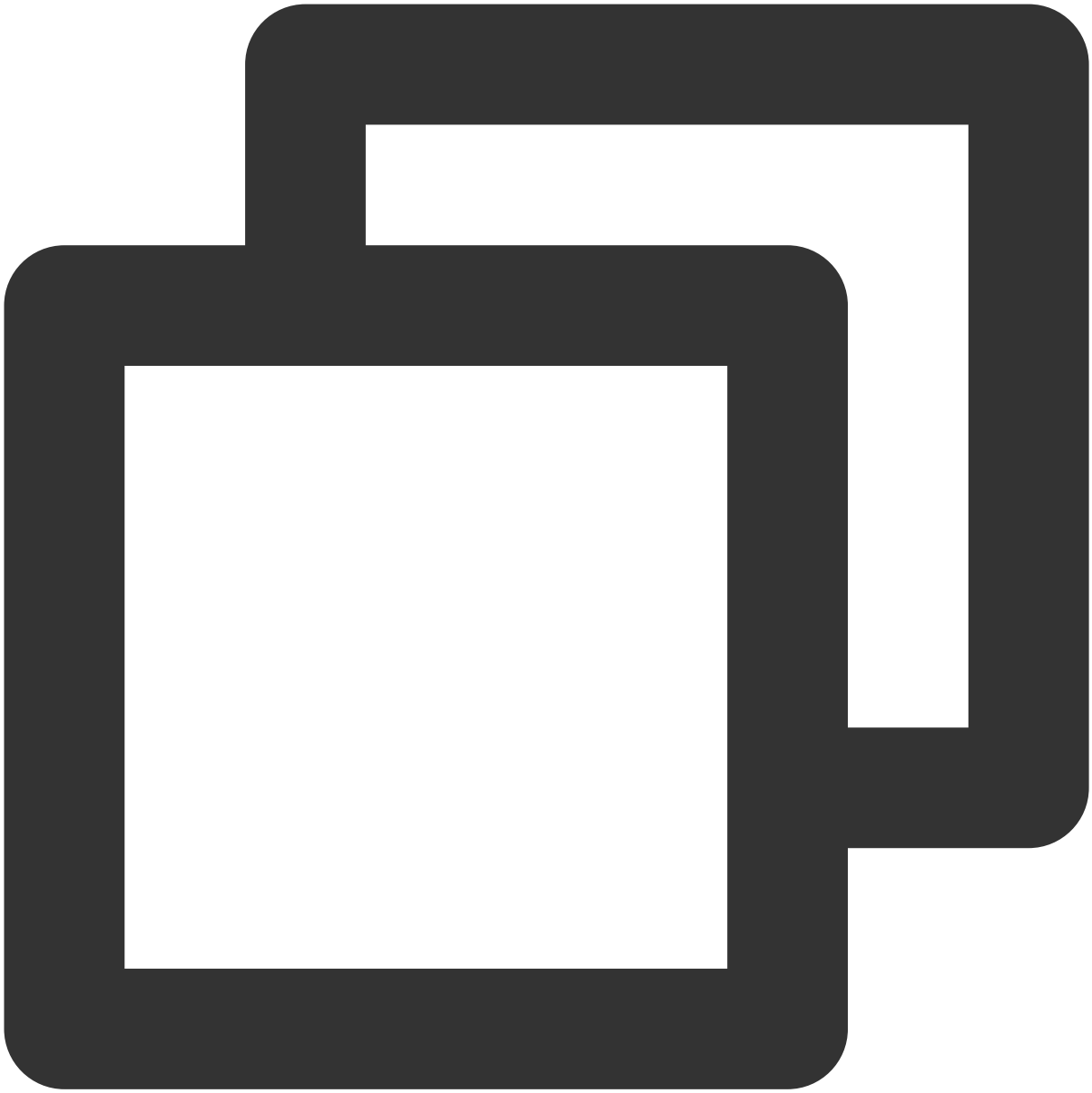
Code example:

©2013-2022 Tencent Cloud. All rights reserved.



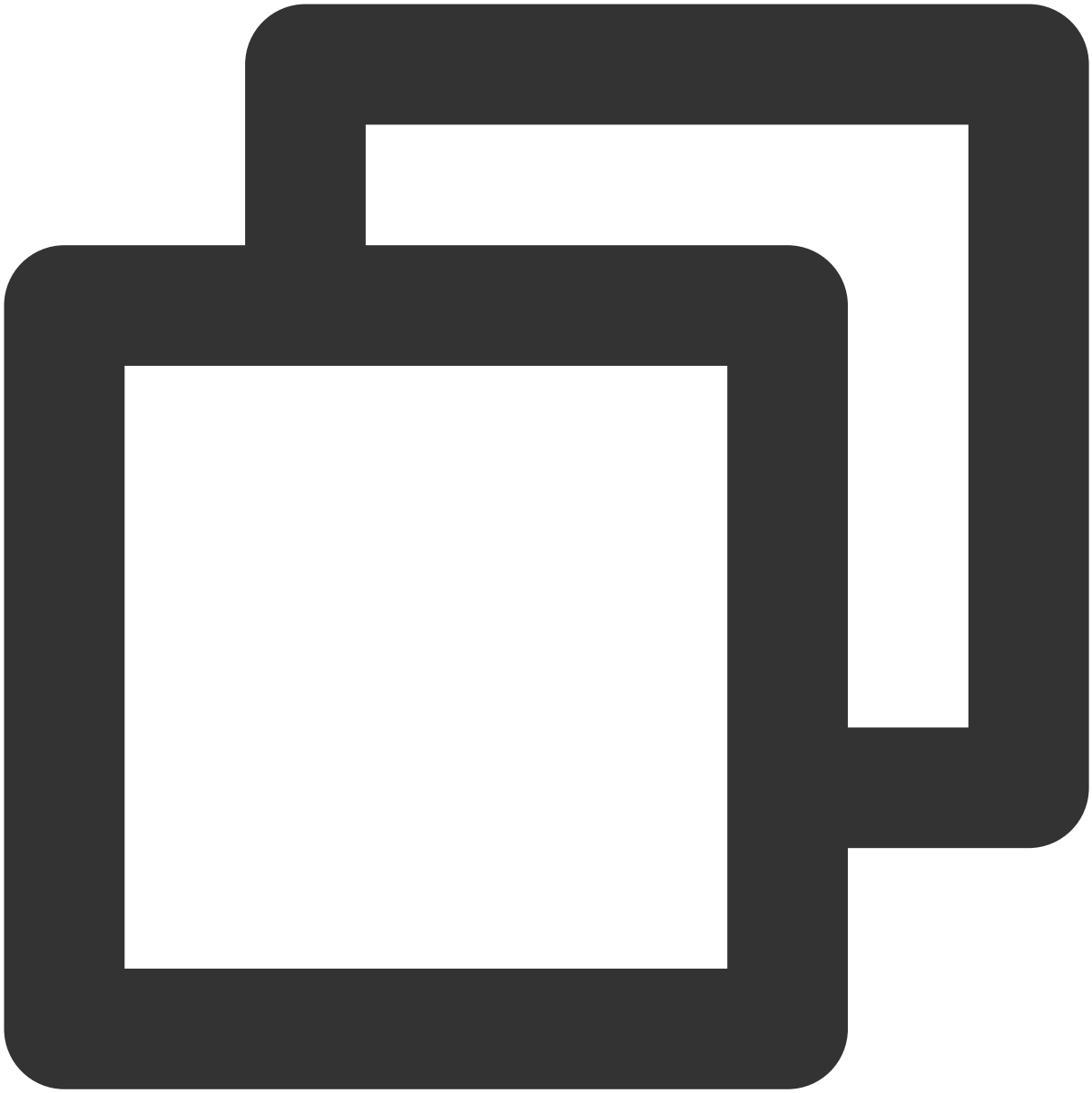
```
/* Component  custom-component.js */
Component({
  options: {
    addGlobalClass: true,
  }
}
```

©2013-2022 Tencent Cloud. All rights reserved.



```
<!-- Component  custom-component.wxml -->
<text class="red-text">
  The colour of this text is determined by the style definitions in `app.wxss` and
</text>
```

©2013-2022 Tencent Cloud. All rights reserved.

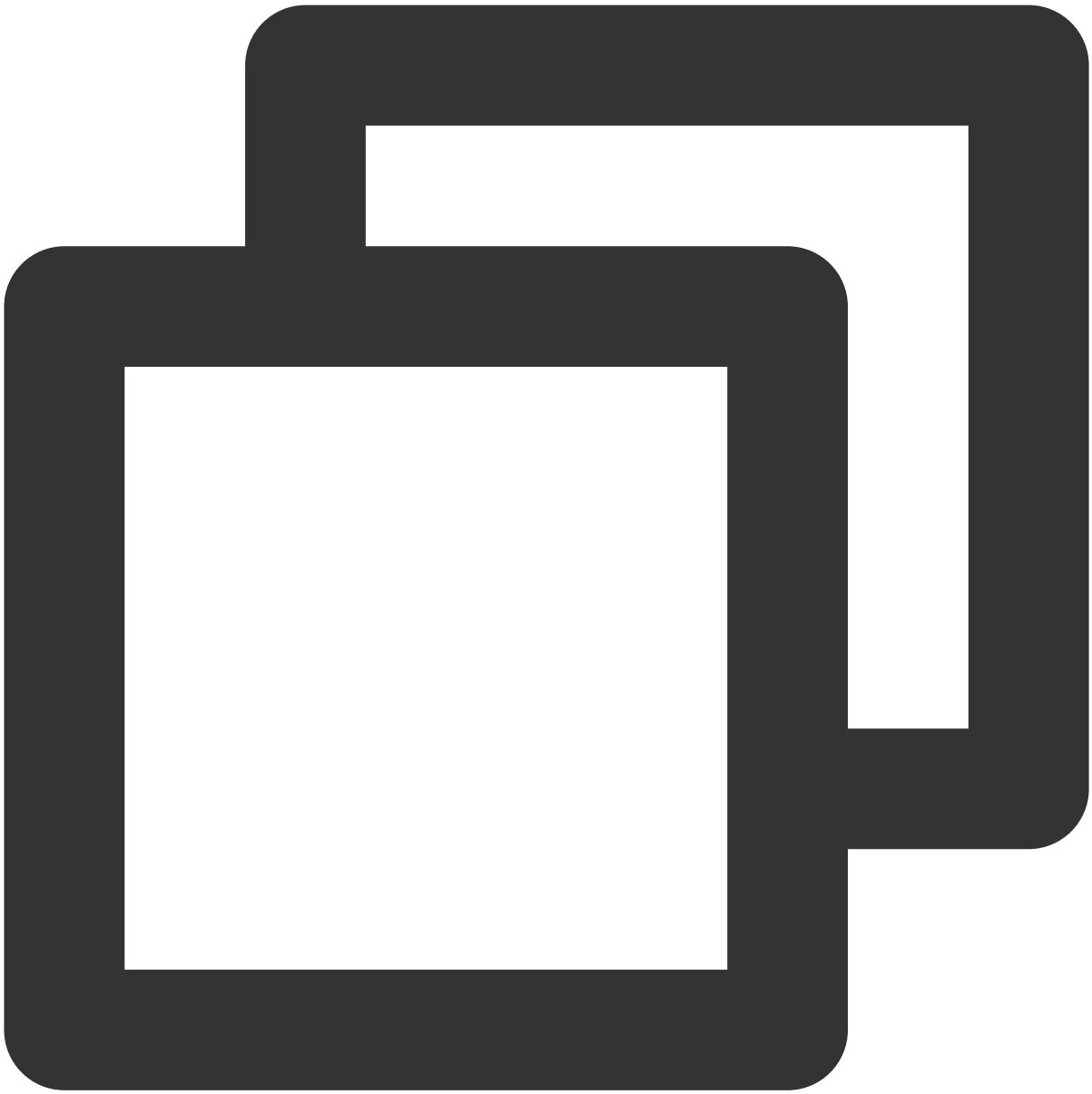


```
/* app.wxss */
.red-text {
  color: red;
}
```

3. Component Constructor

©2013-2022 Tencent Cloud. All rights reserved.

The Component constructor is used to define components. You can specify component properties, data, and methods when calling the Custom Components.



```
Component({

  behaviors: [],

  properties: {
    myProperty: { // Property name
      type: String,
      value: ''
```

©2013-2022 Tencent Cloud. All rights reserved.

```
    },
    myProperty2: String // Simplified definition
  },

  data: {}, // Private data, used for template rendering

  lifetimes: {
    // Lifecycle function, which can be a function or a method name defined in the
    attached: function () { },
    moved: function () { },
    detached: function () { },
  },

  // Lifecycle function, which can be a function or a method name defined in the me
  attached: function () { }, // The declaration in the attached field is overwritte
  ready: function() { },

  pageLifetimes: {
    // Component page lifecycle function
    show: function () { },
    hide: function () { },
    resize: function () { },
  },

  methods: {
    onMyButtonTap: function(){
      this.setData({
        // The properties and data are updated in the same way as with the page dat
      })
    },
    // The internal method starts with an underscore
    _myPrivateMethod: function(){
      // Here, data.A[0].B is set to 'myPrivateData'
      this.setData({
        'A[0].B': 'myPrivateData'
      })
    },
    _propertyChange: function(newVal, oldVal) {

    }
  }

})
```

In the properties definition segment, the property name is written in camel case (propertyName); in wxml, the corresponding property value is written in hyphenated case (component-tag-name property-name="attr value") when

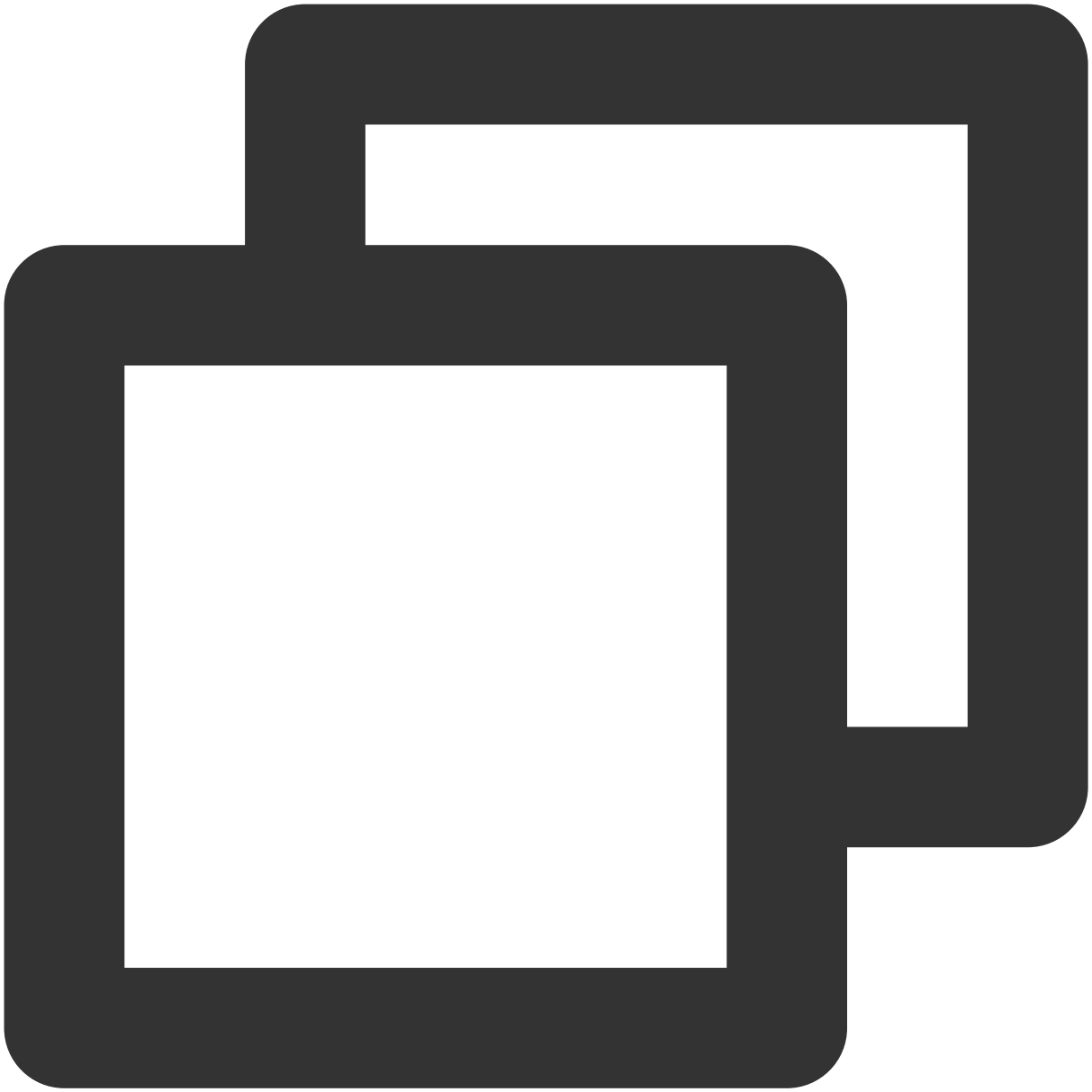
©2013-2022 Tencent Cloud. All rights reserved.

specifying the property value, and in camel case when applying it to data binding (attr=" {{propertyName}}").

3.1 Construct a Page via Component Constructor

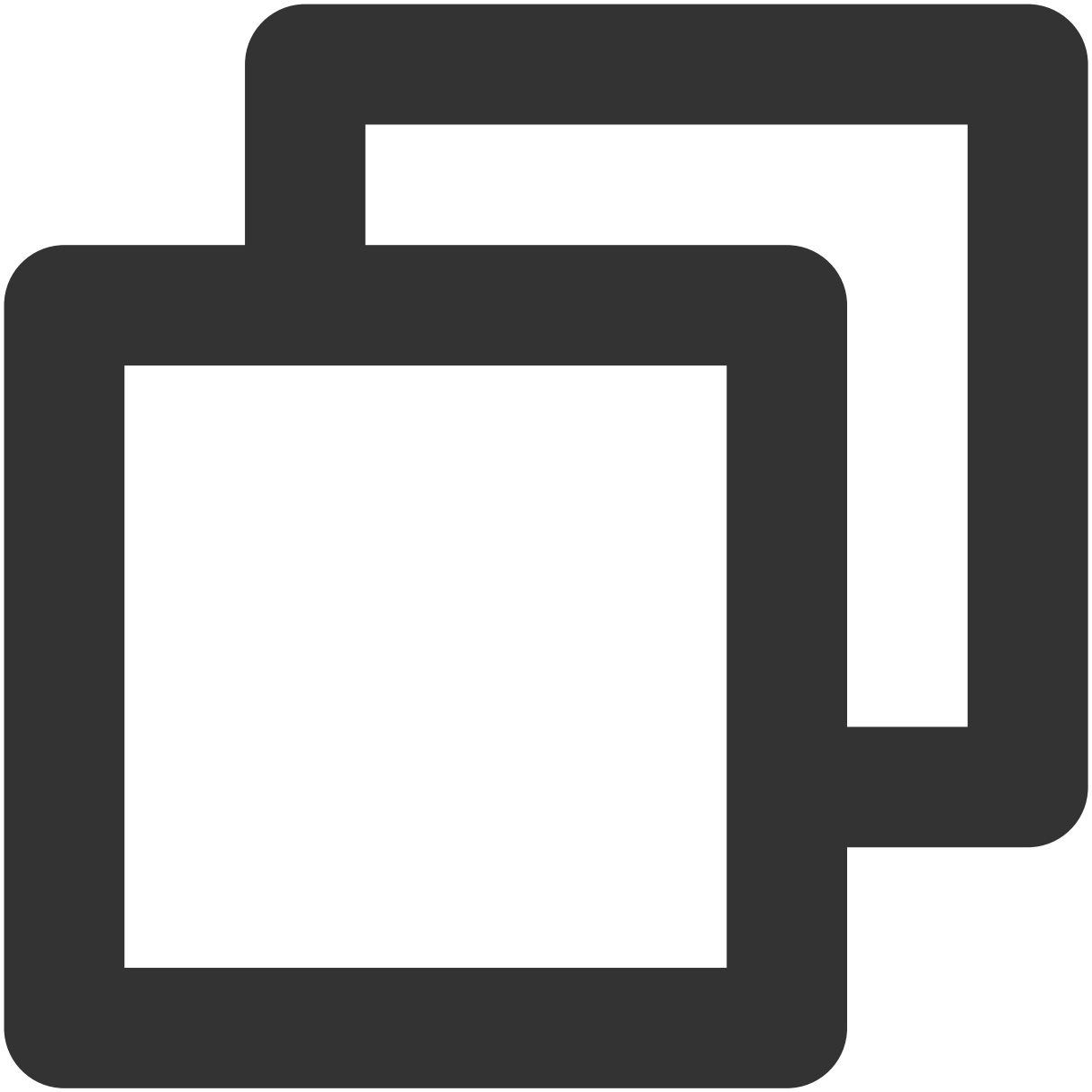
In fact, the Mini Program's page can also be considered as a custom component. Thus, its page can also be constructed using the Component constructor, as it has the same definition field and instance method as a general component. However, you must include the usingComponents definition field in the corresponding json file.
In this case, the properties of the component can be used to receive page parameters, such as visited pages /pages/index/index?paramA=123&paramB=xyz. If the property paramA or paramB is declared, they will be assigned 123 or xyz.
The page's lifecycle method (i.e., the method starting with on) must be written in the methods definition field.
Code example:

©2013-2022 Tencent Cloud. All rights reserved.



```
{
  "usingComponents": {}
}
```

©2013-2022 Tencent Cloud. All rights reserved.



```
Component({
  properties: {
    paramA: Number,
    paramB: String,
  },

  methods: {
    onLoad: function() {
      this.data.paramA // Value of page parameter paramA
      this.data.paramB // Value of page parameter paramB
    }
```

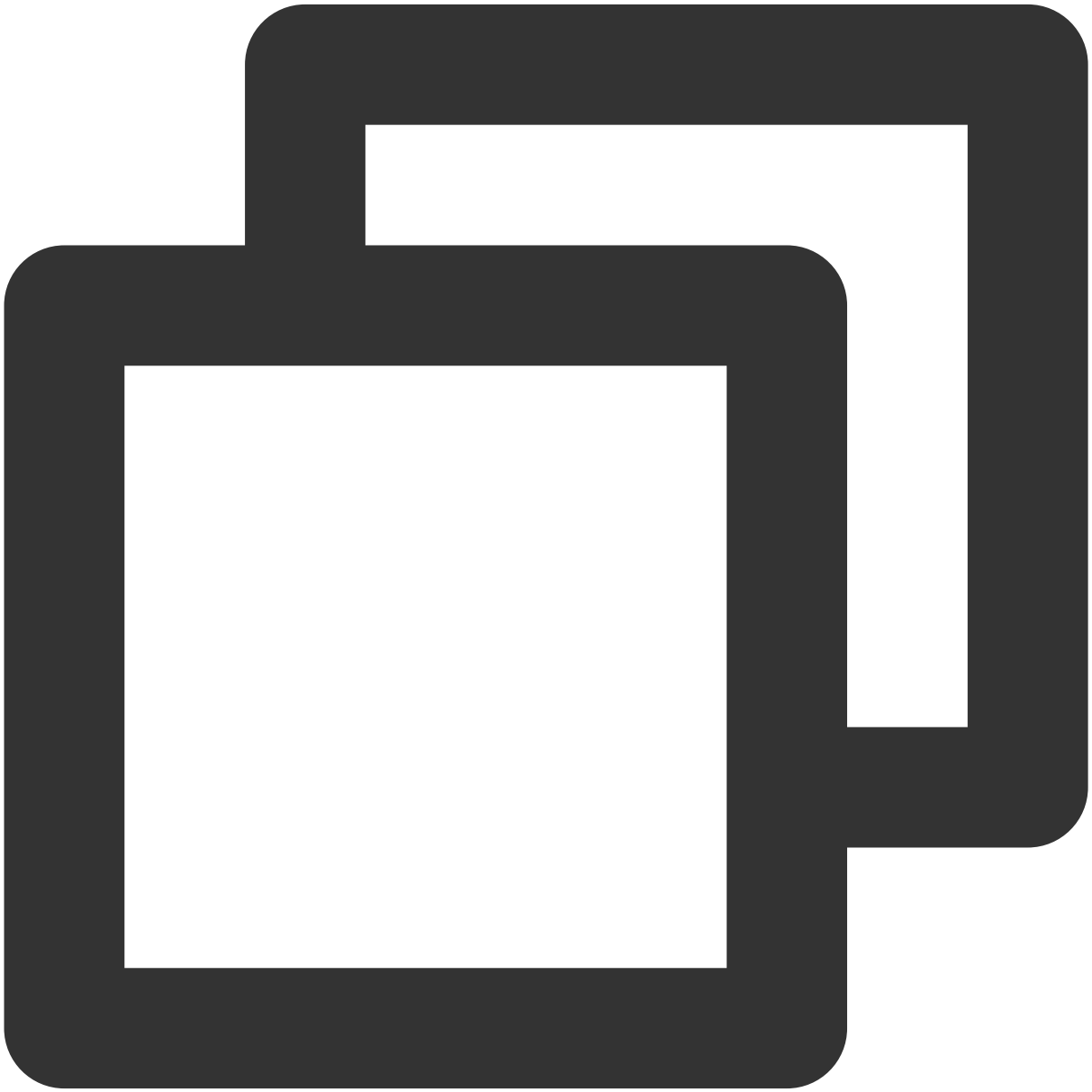
©2013-2022 Tencent Cloud. All rights reserved.

```
  }
})
```

One of the benefits of using the Component constructor to construct pages is that you can use behaviors to extract code segments that are common to all pages.

For example, if you want to execute the same piece of code when all pages are created and destroyed, you can extract that code into behaviors.

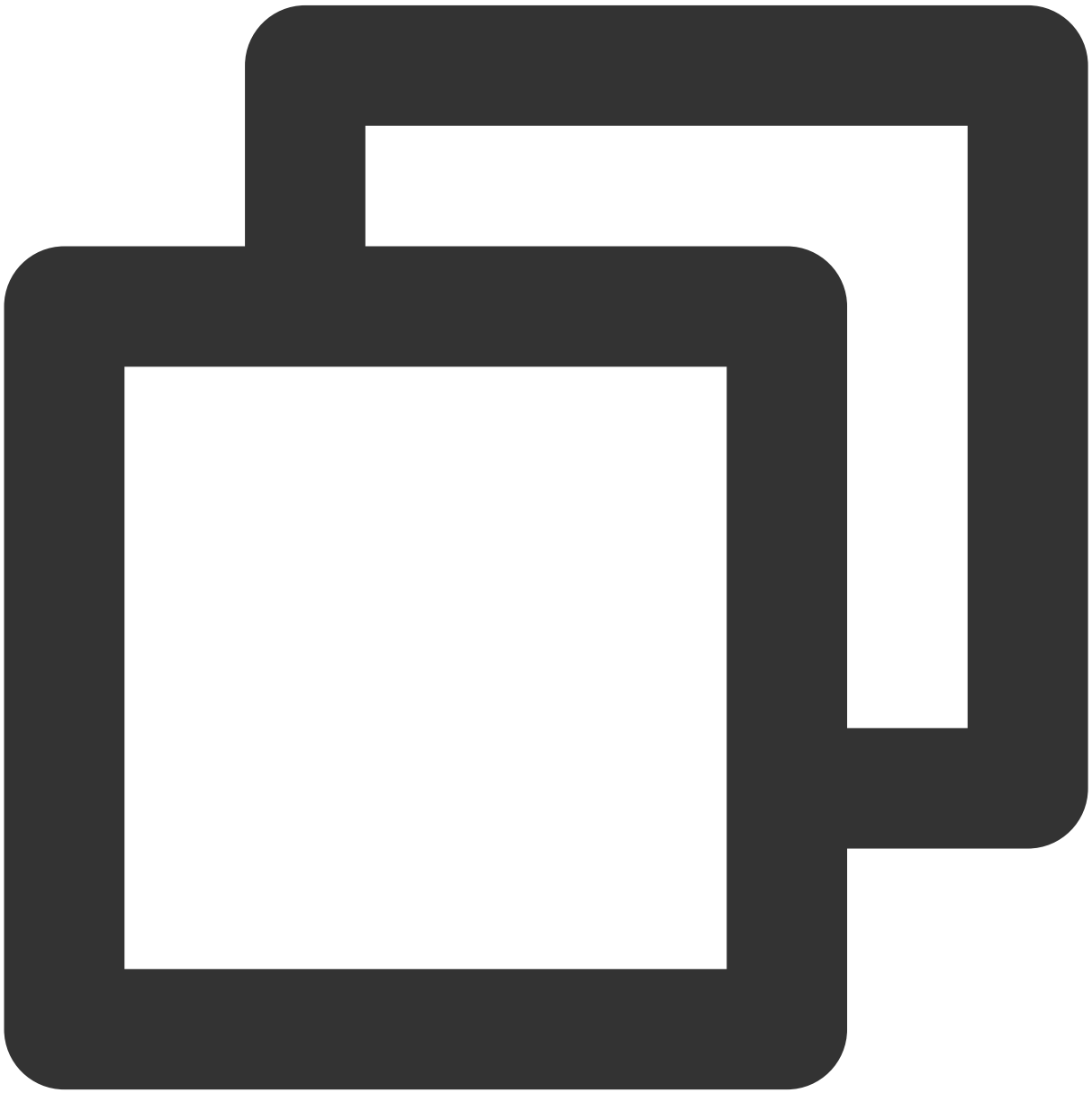
Code example:



```
// page-common-behavior.js
module.exports = Behavior({
```

©2013-2022 Tencent Cloud. All rights reserved.

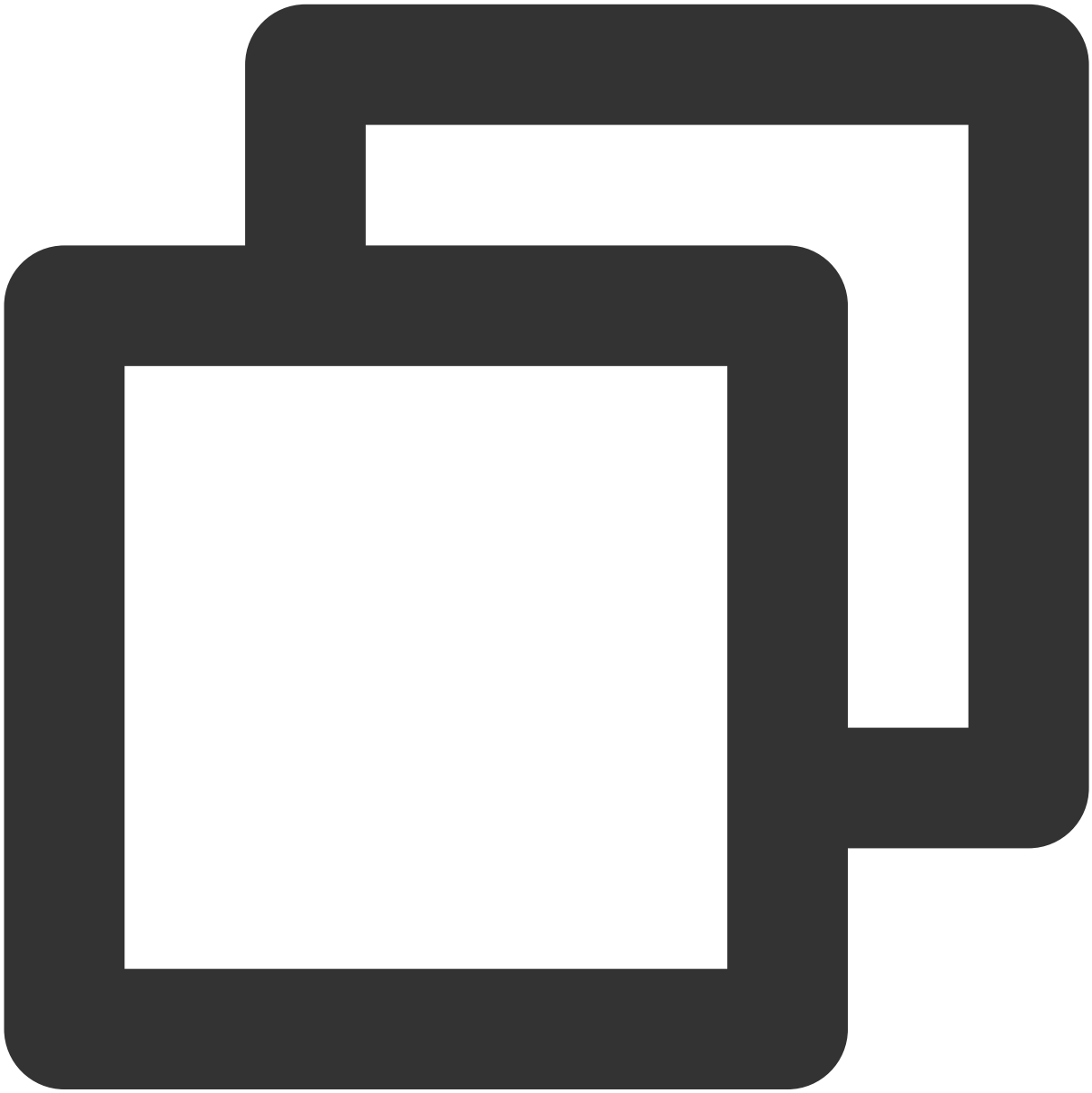
```
  attached: function() {
    // On page creation
    console.info('Page loaded!')
  },
  detached: function() {
    // When the page is destroyed
    console.info('Page unloaded!')
  }
})
```



```
// page A
```

©2013-2022 Tencent Cloud. All rights reserved.

```
var pageCommonBehavior = require('./page-common-behavior')
Component({
  behaviors: [pageCommonBehavior],
  data: { /* ... */ },
  methods: { /* ... */ },
})
```



```
// page B
var pageCommonBehavior = require('./page-common-behavior')
Component({
  behaviors: [pageCommonBehavior],
```

©2013-2022 Tencent Cloud. All rights reserved.

```
  data: { /* ... */ },
  methods: { /* ... */ },
})
```

4. Inter-component Communication and Events

4.1 Inter-component Communication

The following methods are used for communication between components.

WXML data binding: This is used by a parent component to set data for the specified property of a child component. Only JSON-compatible data is supported. For more information, see Component Template and Style.

Event: This is used by a child component to send any type of data to a parent component.

If the two methods above cannot meet your needs, the parent component can also get the instance objects of the child component via the this.selectComponent method. This allows you to directly access any component data or method.

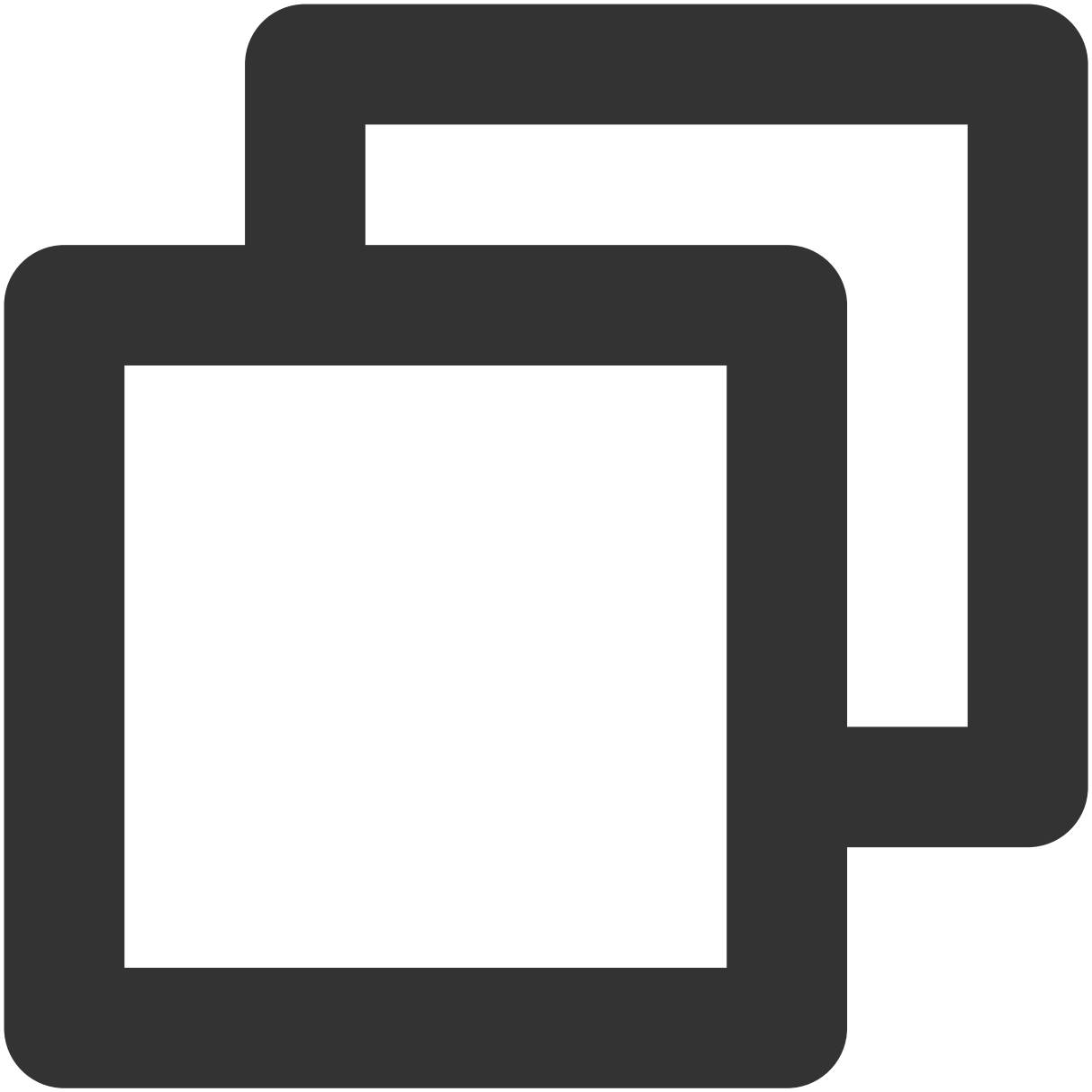
4.2 Monitoring Events

The event system is one of the methods of communication between components. Custom components can trigger any events, and pages that references the components can monitor these events. For the basic concept and usage of events, see Events.

Customer component events are monitored in the same way as with basic component events:

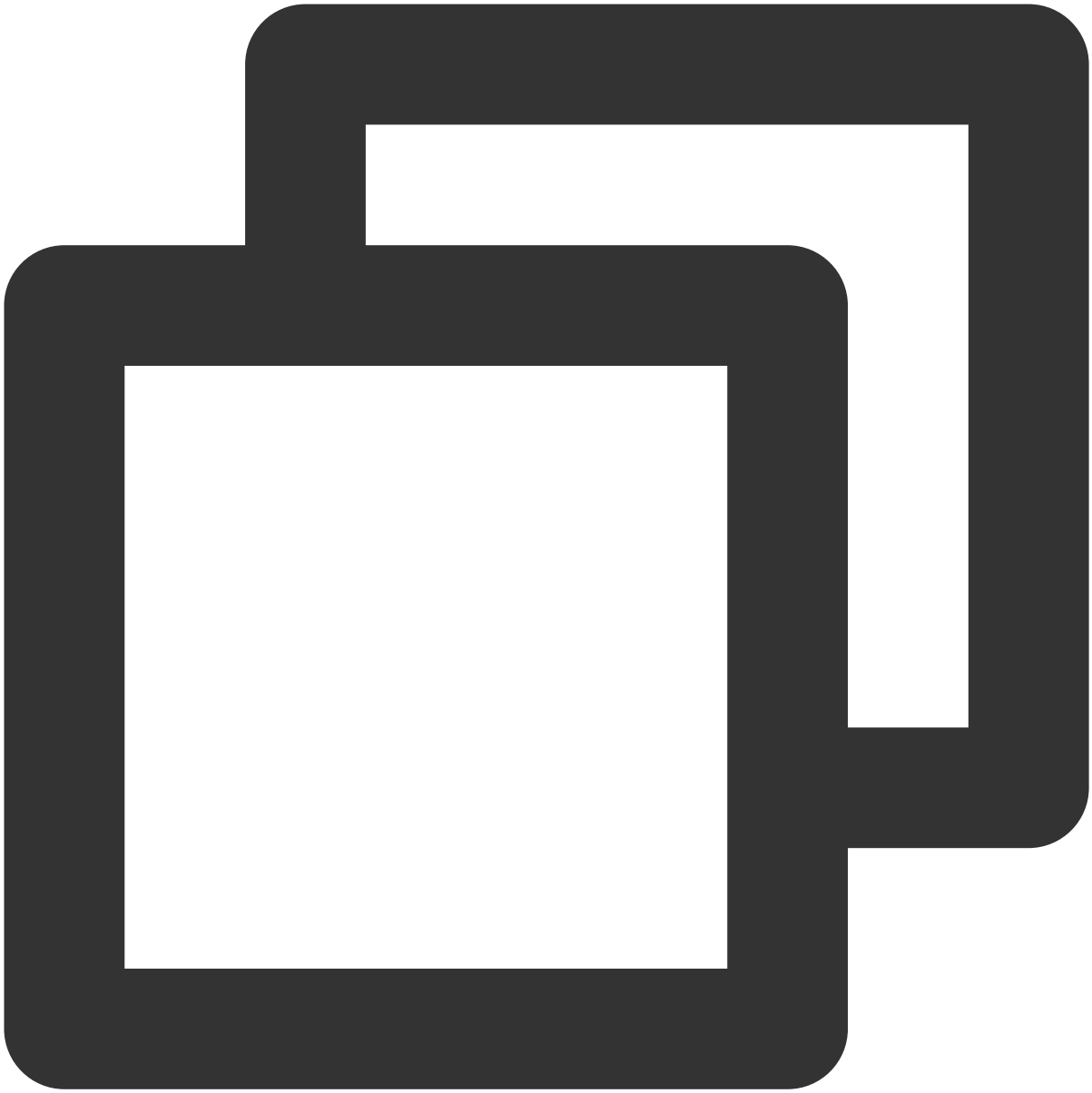
Code example:

©2013-2022 Tencent Cloud. All rights reserved.



```
<!-- The "onMyEvent" method is called when the custom component triggers "myevent"
<component-tag-name bindmyevent="onMyEvent" />
<!-- Or write it like this -->
<component-tag-name bind:myevent="onMyEvent" />
```

©2013-2022 Tencent Cloud. All rights reserved.



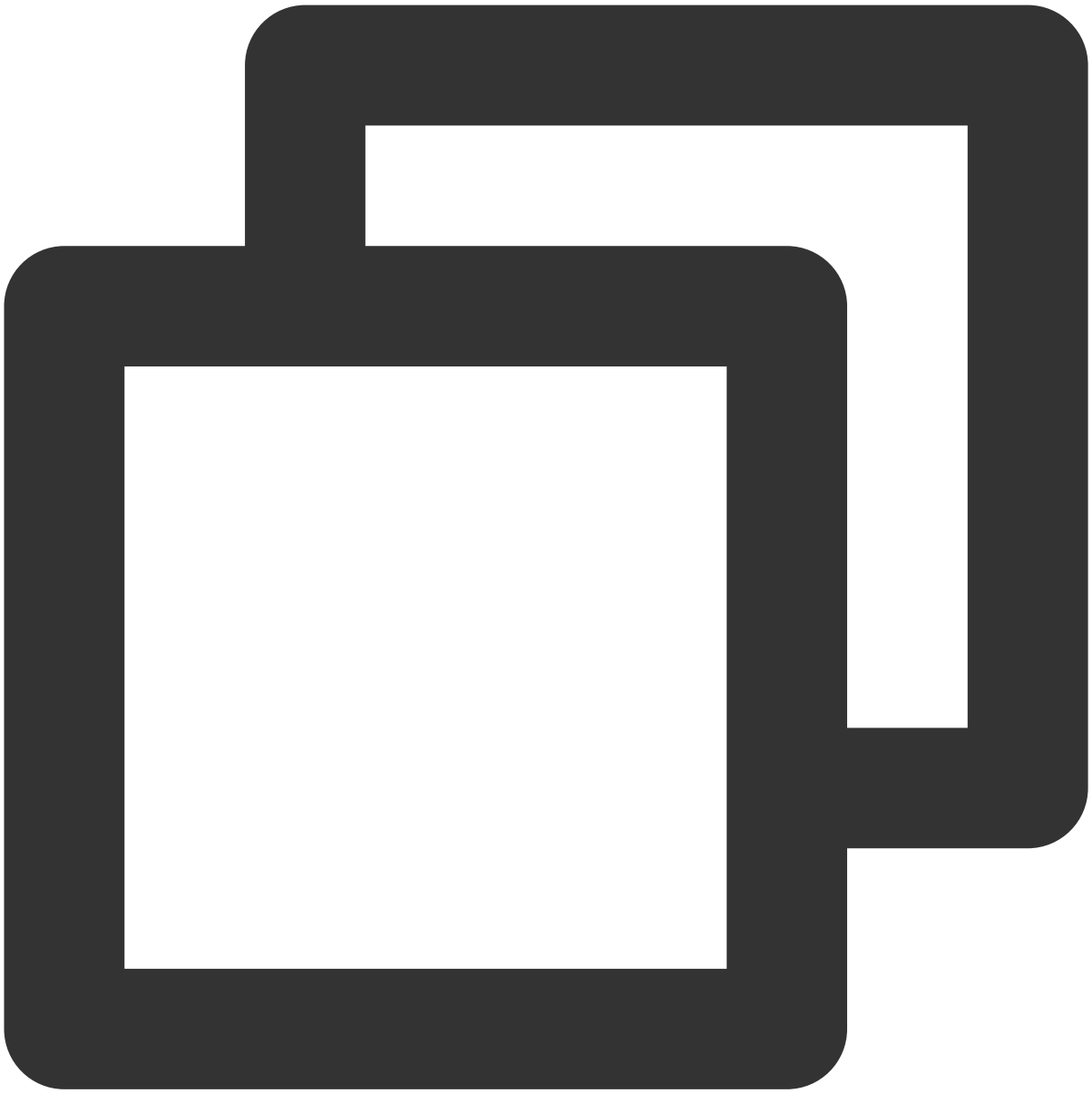
```
Page({
  onMyEvent: function(e){
    e.detail // The detail object provided when the custom component triggers an ev
  }
})
```

4.3 Triggering Event

When a custom component triggers an event, the `triggerEvent` method should be used to specify the event name, detail object, and event options:

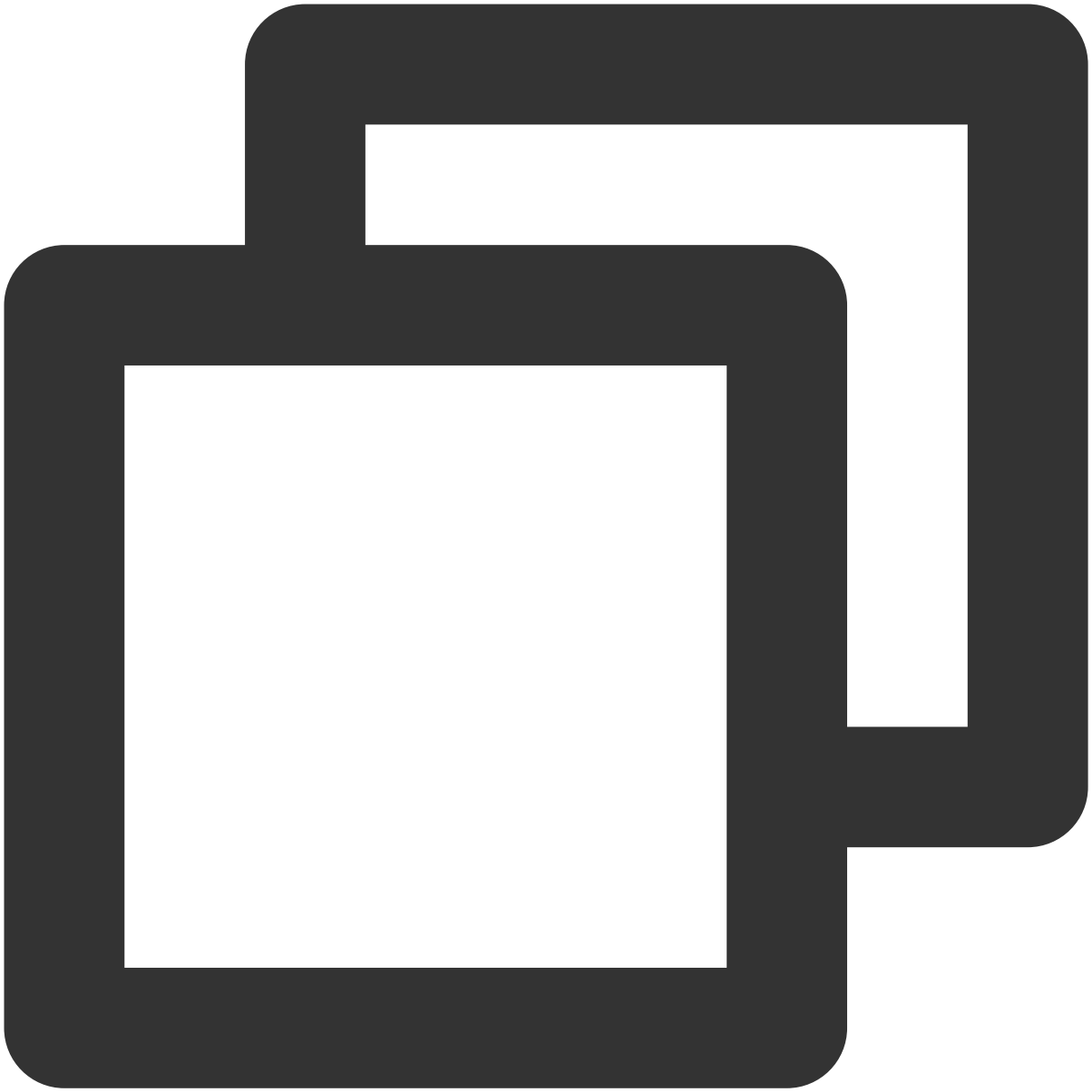
©2013-2022 Tencent Cloud. All rights reserved.

Code example:



```
<!-- In the custom component -->
<button bindtap="onTap">Click this button to trigger the "myevent" event</button>
```

©2013-2022 Tencent Cloud. All rights reserved.



```
Component({
  properties: {},
  methods: {
    onTap: function(){
      var myEventDetail = {} // detail object, provided to the event monitoring fun
      var myEventOption = {} // Event triggering options
      this.triggerEvent('myevent', myEventDetail, myEventOption)
    }
  }
})
```

©2013-2022 Tencent Cloud. All rights reserved.

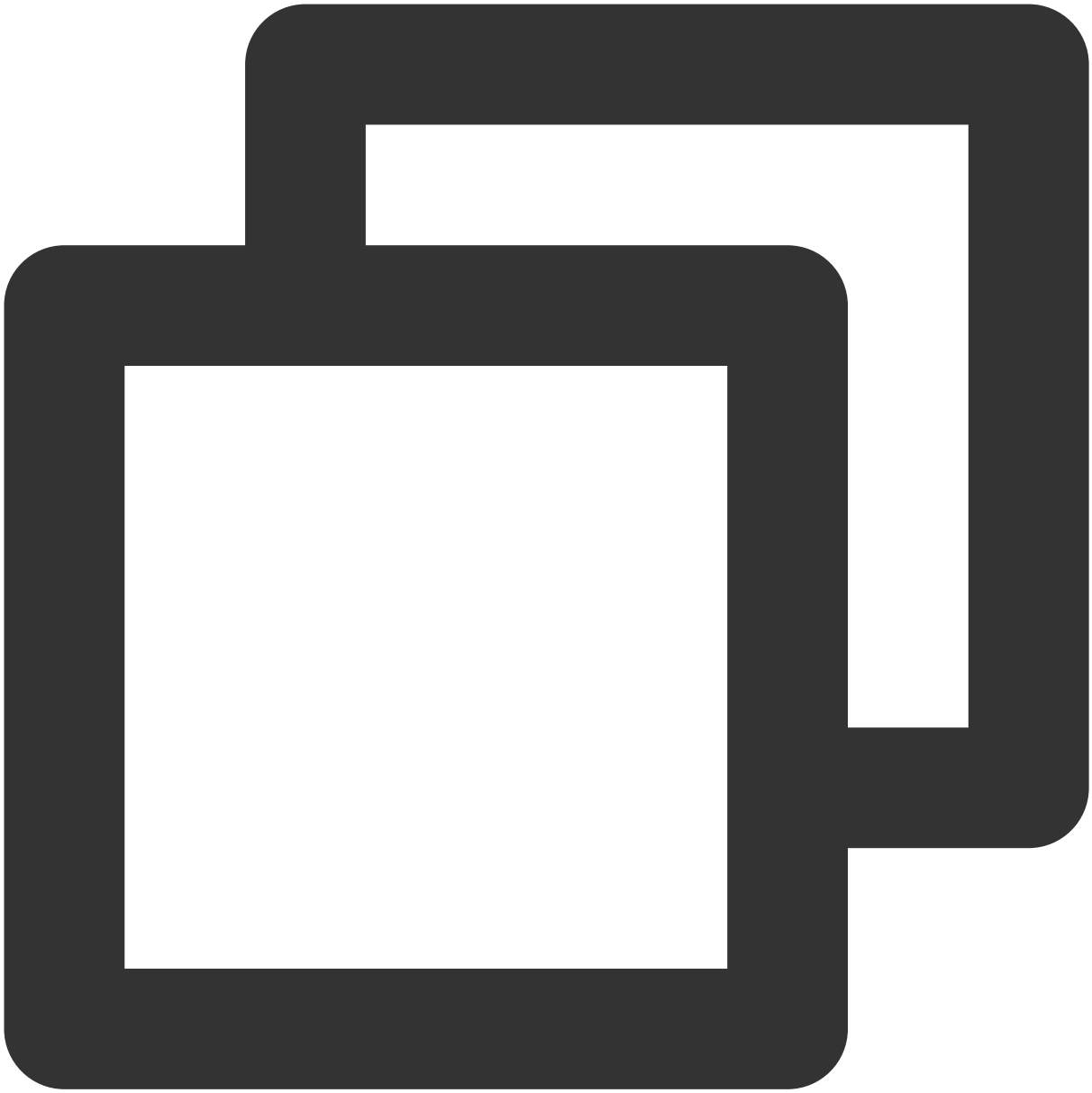
Event triggering options include:

Option	Type	Required	Default	Description
bubbles	Boolean	No	false	Indicates whether the event is a bubbling event
composed	Boolean	No	false	Indicates whether the event can cross component boundaries. If it is false, the event is only triggered in the node tree of the referenced component and will not enter other components.
capturePhase	Boolean	No	false	Indicates whether the event has a capture phase

For more information on bubbling and capture phase, see Events.

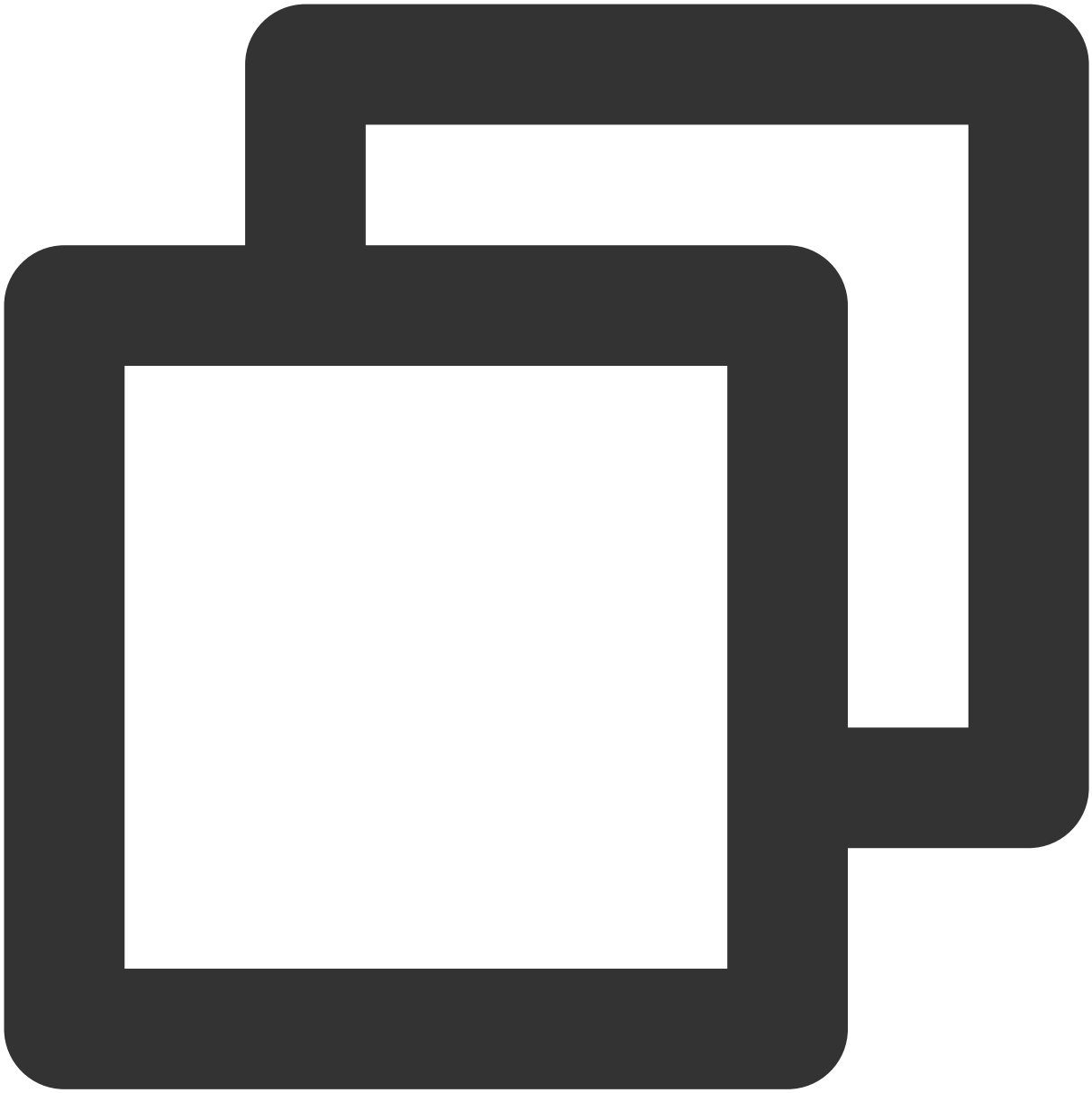
Code example:

©2013-2022 Tencent Cloud. All rights reserved.



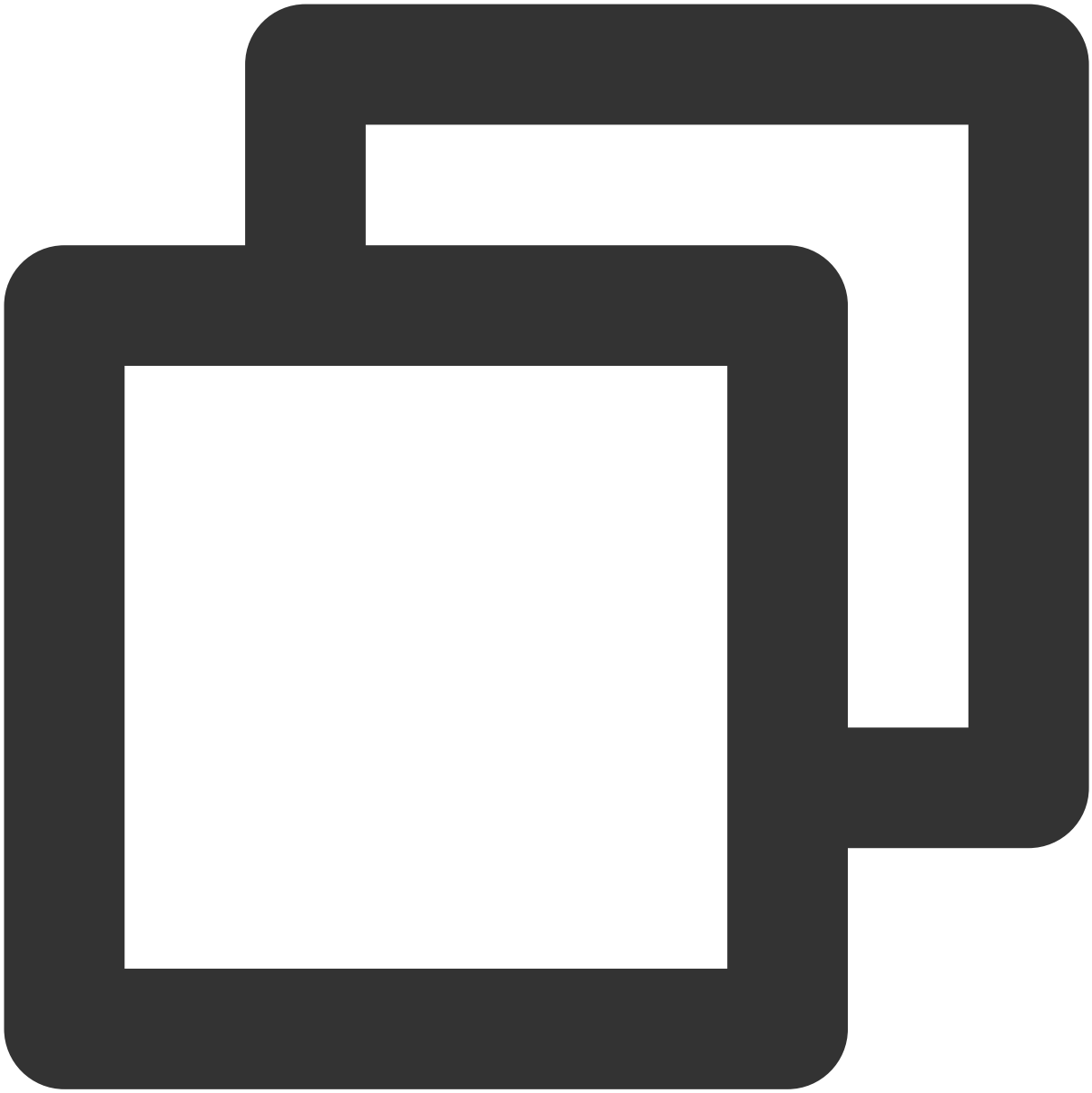
```
// page page.wxml
<another-component bindcustomevent="pageEventListener1">
  <my-component bindcustomevent="pageEventListener2"></my-component>
</another-component>
```

©2013-2022 Tencent Cloud. All rights reserved.



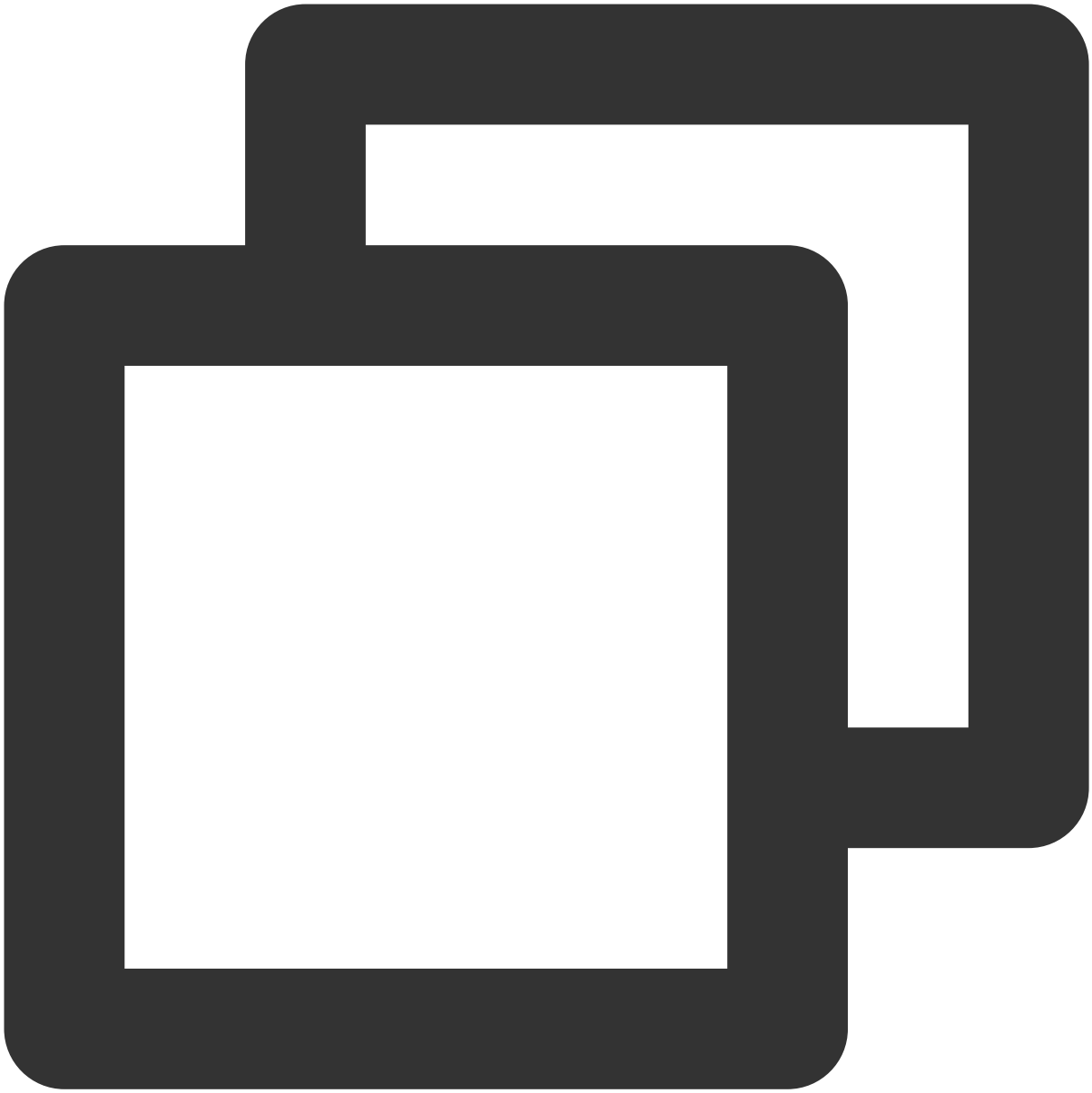
```
// Component  another-component.wxml
<view bindcustomevent="anotherEventListener">
  <slot />
</view>
```

©2013-2022 Tencent Cloud. All rights reserved.



```
// Component  my-component.wxml
<view bindcustomevent="myEventListener">
  <slot />
</view>
```

©2013-2022 Tencent Cloud. All rights reserved.



```
// Component  my-component.js
Component({
  methods: {
    onTap: function(){
      this.triggerEvent('customevent', {}) // Only pageEventListener2 is triggered
      this.triggerEvent('customevent', {}, { bubbles: true }) // pageEventListener2
      this.triggerEvent('customevent', {}, { bubbles: true, composed: true }) // pa
    }
  }
})
```

©2013-2022 Tencent Cloud. All rights reserved.

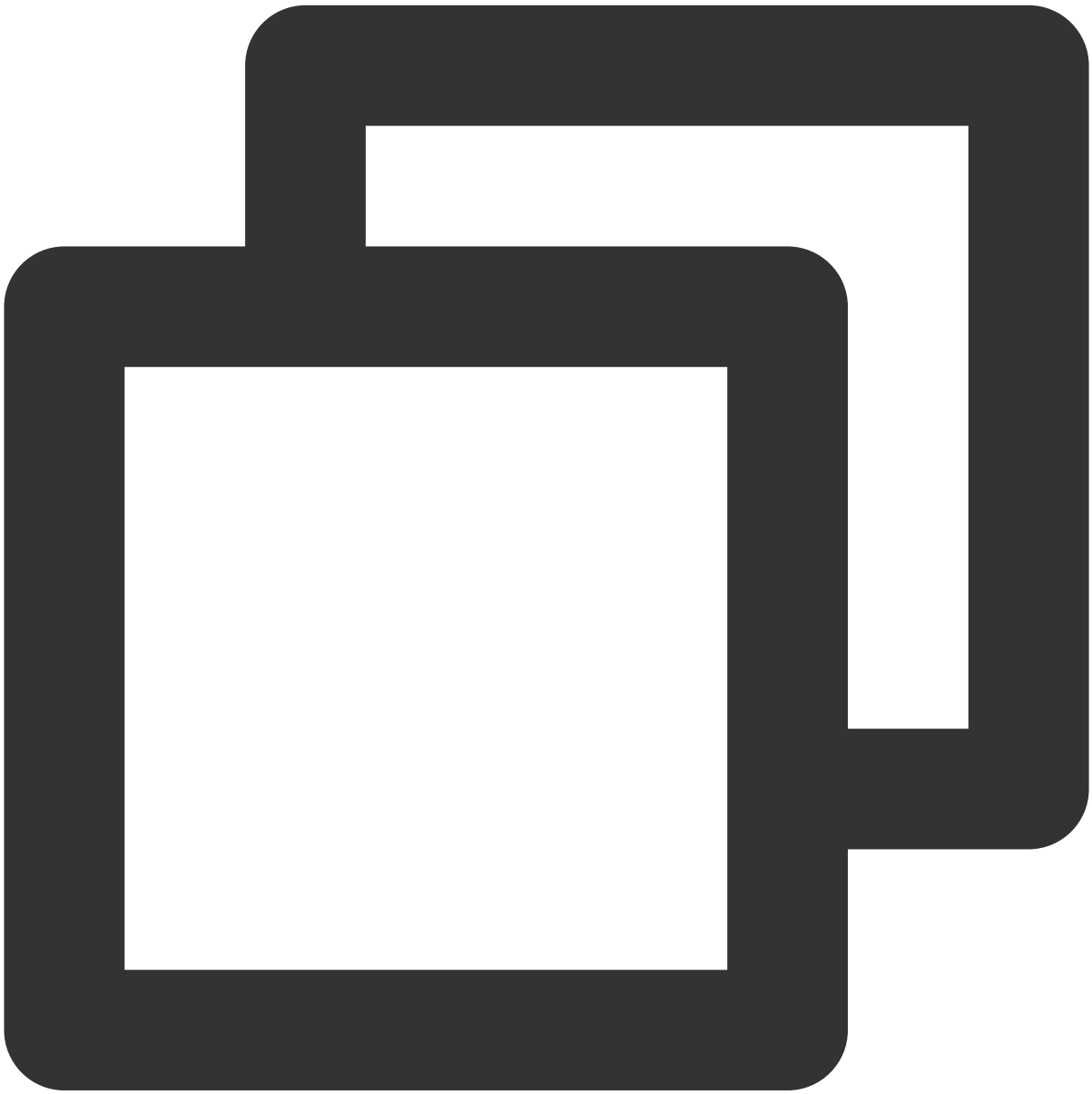
4.4 Getting component instances

This.selectComponent can be called from the parent component to get the instance of the child component.

You need to pass a match selector, e.g. `this.selectComponent(".my-component").`

The detailed syntax of selector can be found in the selector.

Code example:



```
// parent component
Page({
  data: {},
  getChildComponent: function () {
```

©2013-2022 Tencent Cloud. All rights reserved.

```
    const child = this.selectComponent('.my-component');
    console.log(child)
  }
})
```

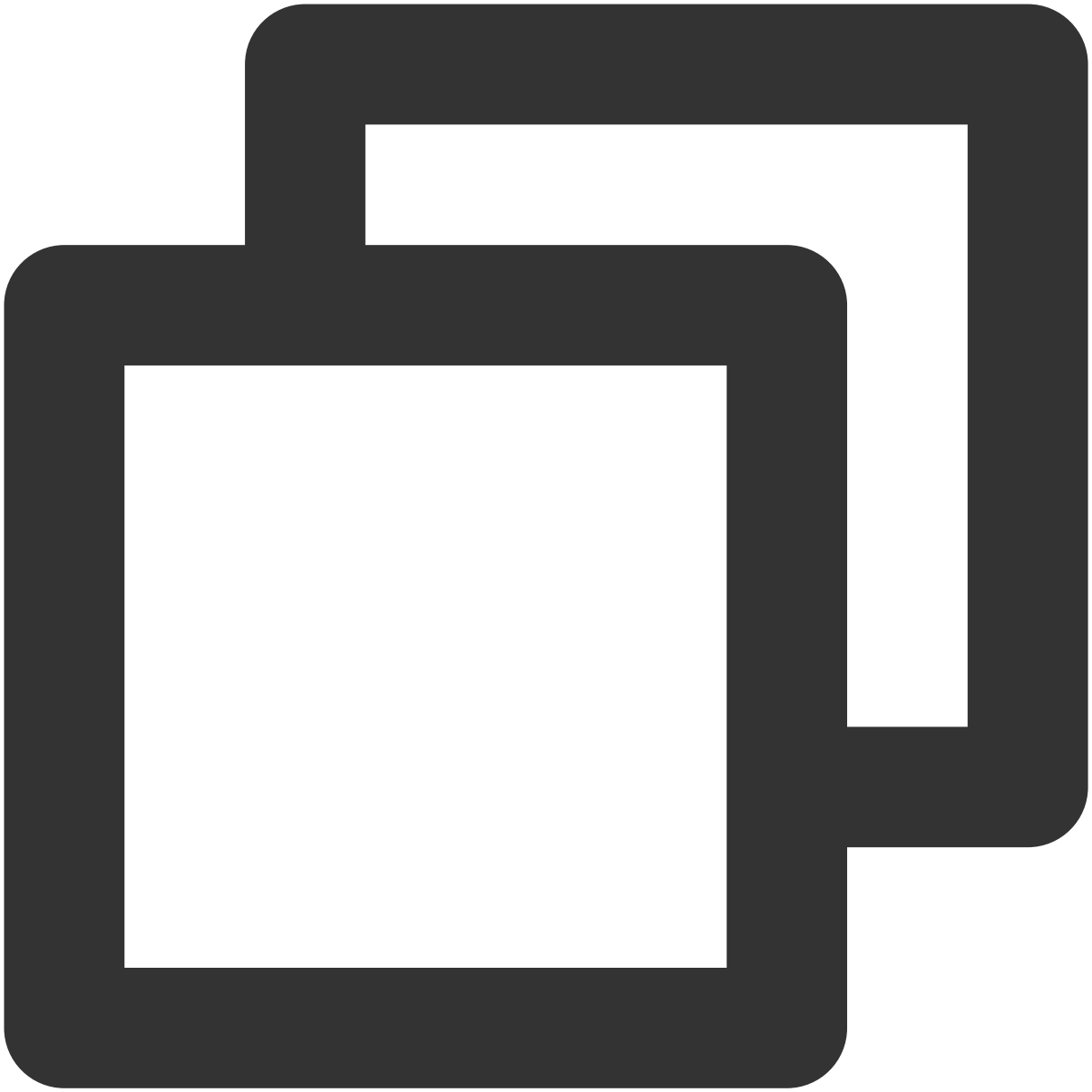
In the above example, the parent component will get the instance object of the child component with class my-component, i.e. this of the child component.

If you want to customise the data returned by selectComponent, you can use the built-in behavior: wx://component-export

When this behaviour is used, the export definition field in the custom component will be used to specify the return value of the component when it is called by selectComponent.

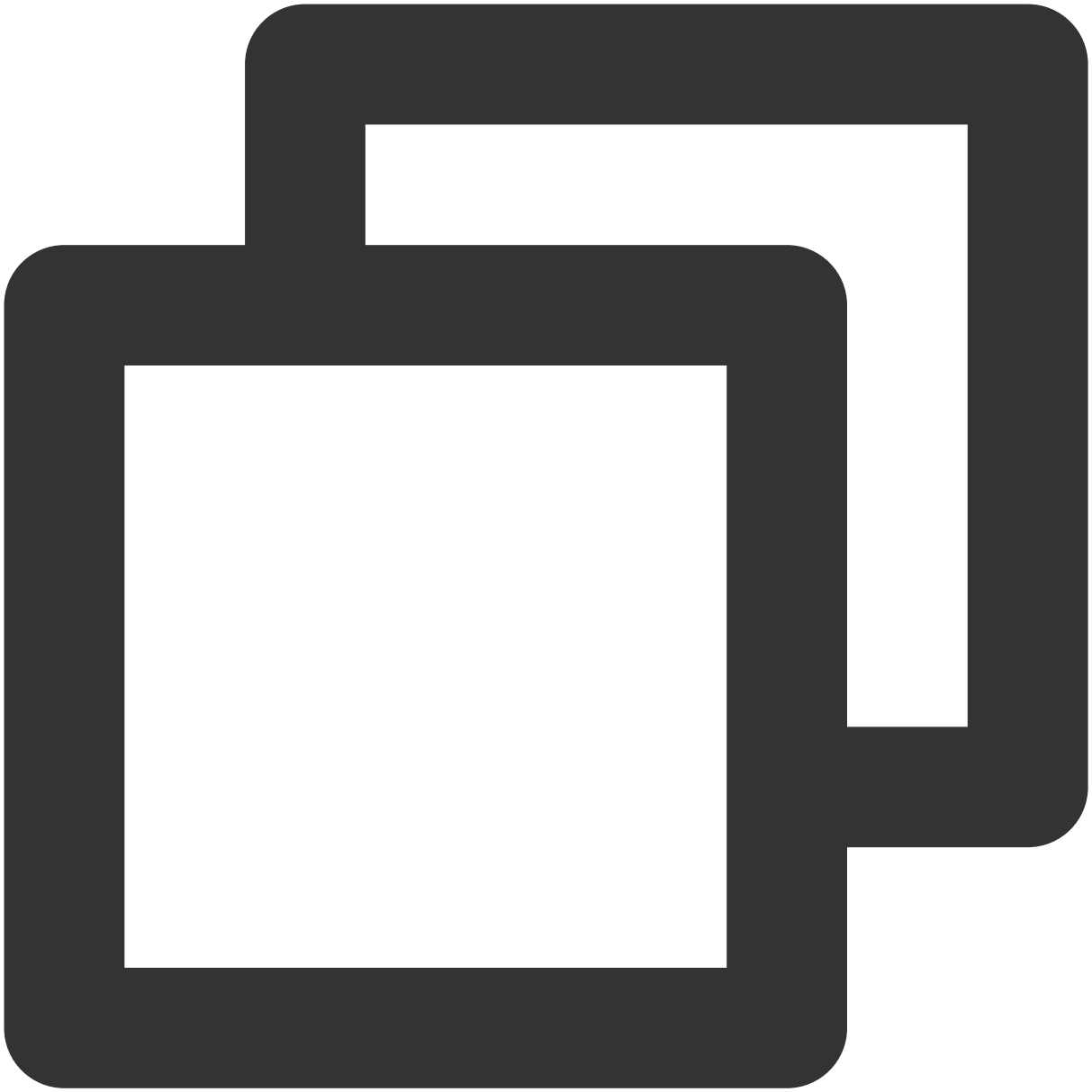
Code example:

©2013-2022 Tencent Cloud. All rights reserved.



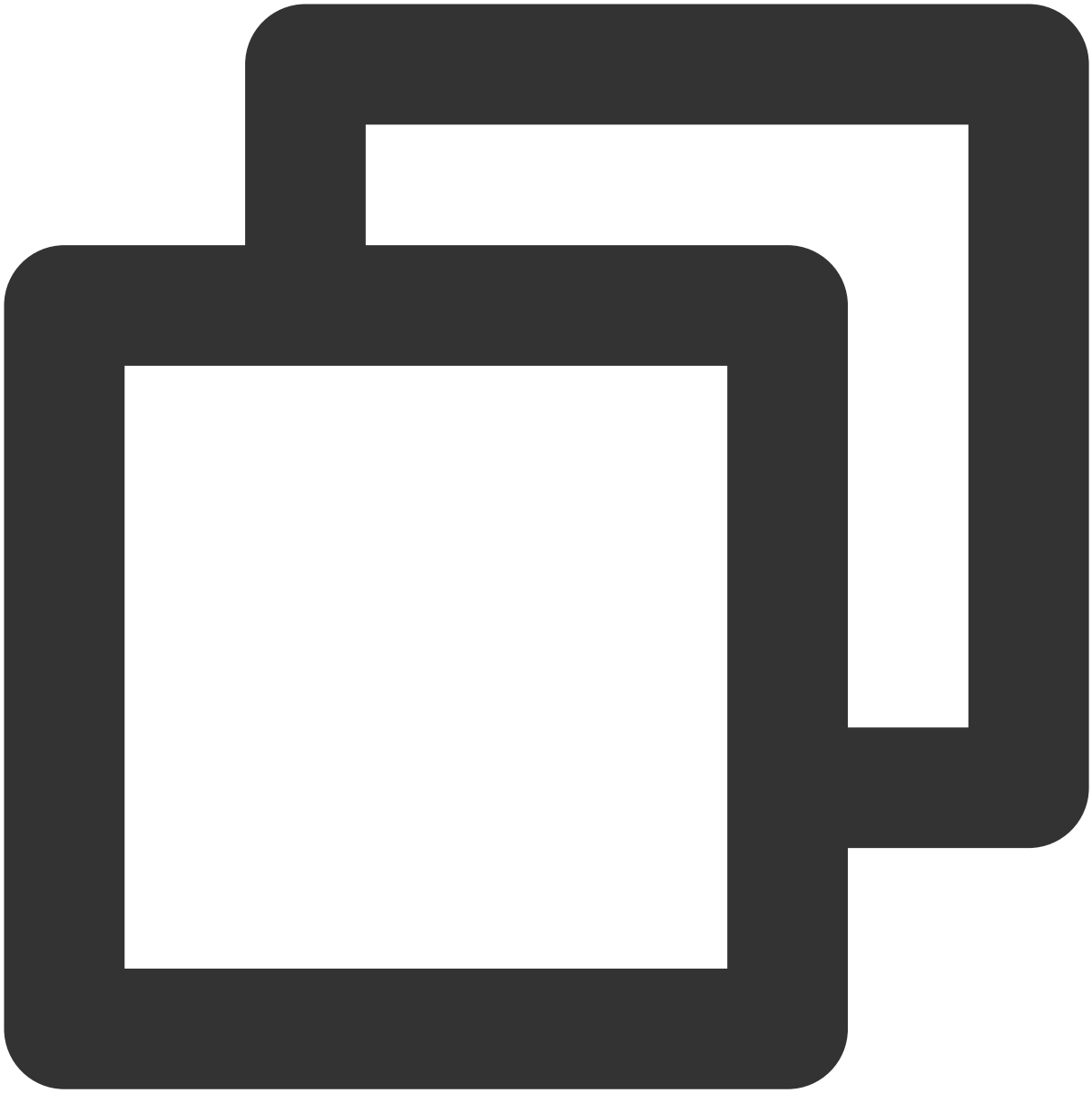
```
// Custom Components my-component Internal
Component({
  behaviors: ['wx://component-export'],
  export() {
    return { myField: 'myValue' }
  }
})
```

©2013-2022 Tencent Cloud. All rights reserved.



```
<! -- When using custom components -->
<my-component id="the-id" />
```

©2013-2022 Tencent Cloud. All rights reserved.



```
// Called by the parent component
const child = this.selectComponent('#the-id') // be tantamount to { myField: 'myVal
```

In the above example, the parent component gets the object `{ myField: 'myValue' }` when it gets the child component instance with `id` `the-id` .

5. Component Lifecycle

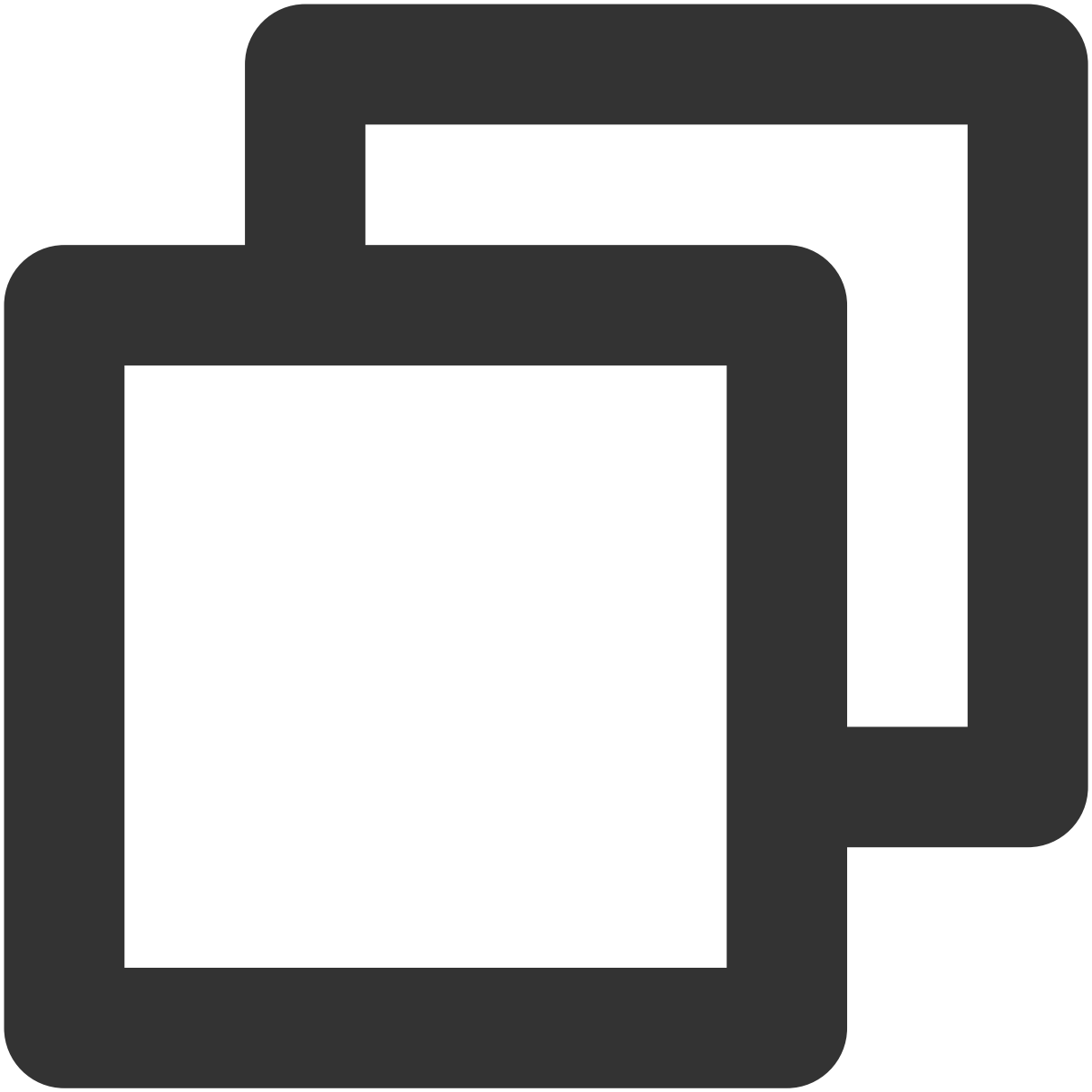
©2013-2022 Tencent Cloud. All rights reserved.

Component lifecycles refer to the functions in the component itself. These functions are automatically triggered at certain timepoints or when specific framework events occur.
The most important lifecycles are `created` , `attached` , and `detached` , which include the main time points in the entire lifecycle process of a component instance.
When a component instance is `created` , the created lifecycle is triggered. In `this case` , the `component data` , **this.data is defined as data in the Component constructor**. You cannot call setData yet. Generally, this lifecycle should only be used to add custom property fields to the component this.
After the component has been completely initialized and entered the page node tree, the `attached` lifecycle is triggered. In this case, `this.data` has been initialized to the current value of the component. This lifecycle is very useful, where most initialization work can be completed.
After the component leaves the page node tree, the `detached` lifecycle is triggered. When you exit a page, if the component is still in the page node tree, detached is triggered.

5.1 Defining Lifecycle Methods

Lifecycle methods can be directly defined in level-one parameters of the Component constructor.
component lifecycles can also be declared in the lifetimes field. We recommend this method as it has the highest priority.
Code example:

©2013-2022 Tencent Cloud. All rights reserved.



```
Component({
  lifetimes: {
    attached: function() {
      // Executed when the component instance enters the page node tree
    },
    detached: function() {
      // Executed when the component instance is removed from the page node tree
    },
  },
  // The older definition methods below can be used for compatibility with base lib
  attached: function() {
```

©2013-2022 Tencent Cloud. All rights reserved.

```
    // Executed when the component instance enters the page node tree
  },
  detached: function() {
    // Executed when the component instance is removed from the page node tree
  },
  // ...
})
```

You can also write lifecycle methods in behaviors, which will not be overwritten by lifecycles of the same name in other behaviors. Note that, if a component directly or indirectly references the same behavior multiple times, the lifecycle function in this behavior is only executed once during a single execution time.

All available lifecycles are listed below:

Lifecycle	Parameter	Description
created	None	Executed when the component instance is created
attached	None	Executed when the component instance enters the page node tree
ready	None	Executed when the component layout is completed in the view layer
moved	None	Executed when the component instance is moved to another node tree
detached	None	Executed when the component instance is removed from the page node tree
error	Object Error	Executed each time the component method throws an exception

5.2 Component Page Lifecycles

There are also some special lifecycles that are not strongly associated with components but must sometimes be known by components for internal processing. These lifecycles are called "component page lifecycles" and defined in the `pageLifetimes` definition field. The available component page lifecycles include:

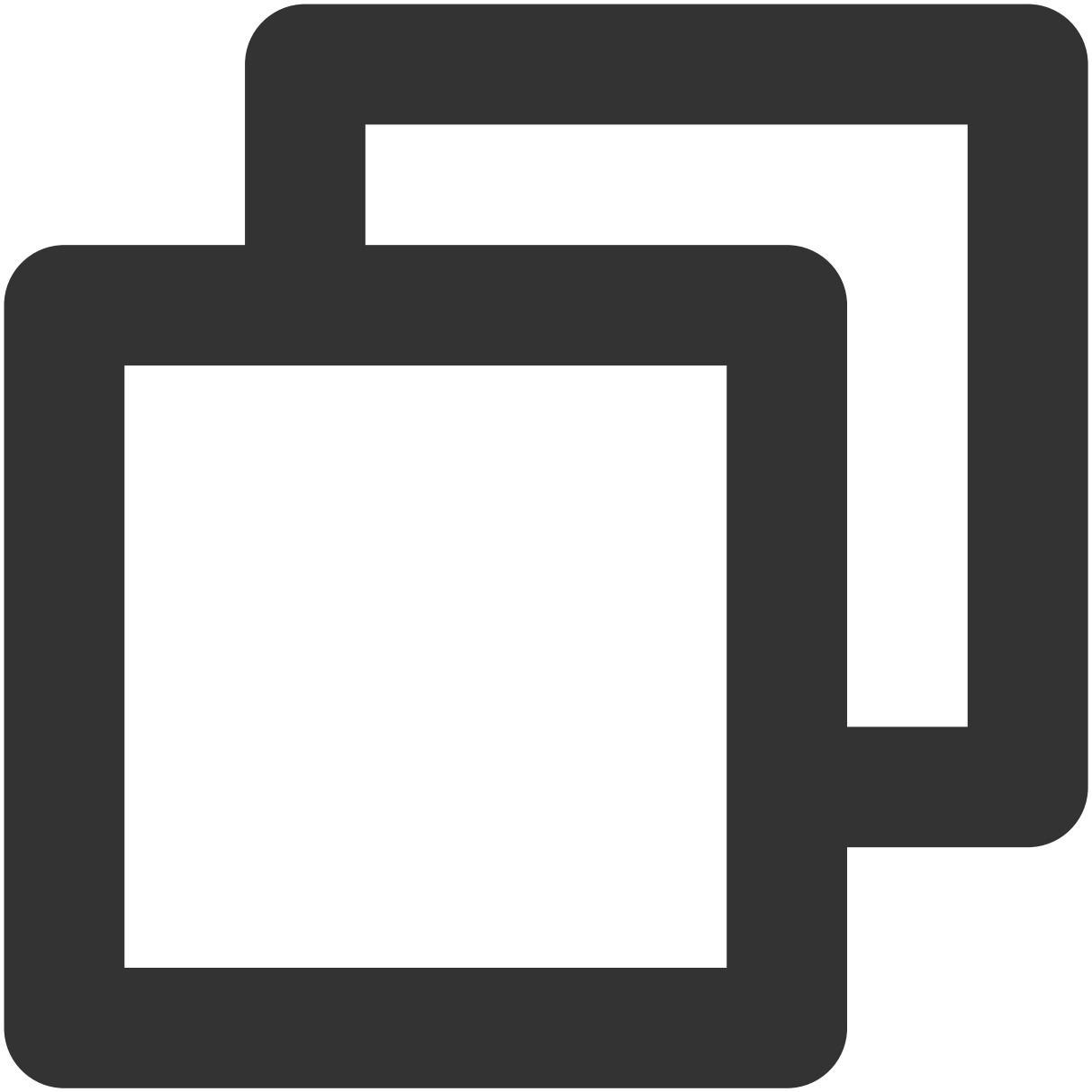
Lifecycle	Parameter	Description
show	None	Executed when the component's page is displayed
hide	None	Executed when the component's page is hidden
resize	Object Size	Executed when the component's page is resized

Note：

The pageLifetime of the custom tabBar will not be triggered.

Code example:

©2013-2022 Tencent Cloud. All rights reserved.



```
Component({
  pageLifetimes: {
    show: function() {
      // Page displayed
    },
    hide: function() {
      // Page hidden
    },
    resize: function(size) {
      // Page resized
    }
```

©2013-2022 Tencent Cloud. All rights reserved.

```
    }
})
```

6. behaviors

The `behaviors` feature is used to share code between components, similar to "mixins" or "traits" in certain programing languages.

A `behavior` can include a set of properties, data, lifecycle functions, and methods. **When referenced by a component, the behavior's properties, data, and methods are merged into the component. Each lifecycle function is also called at the appropriate time**. A component can reference more than one behavior and a behavior can reference another behavior.

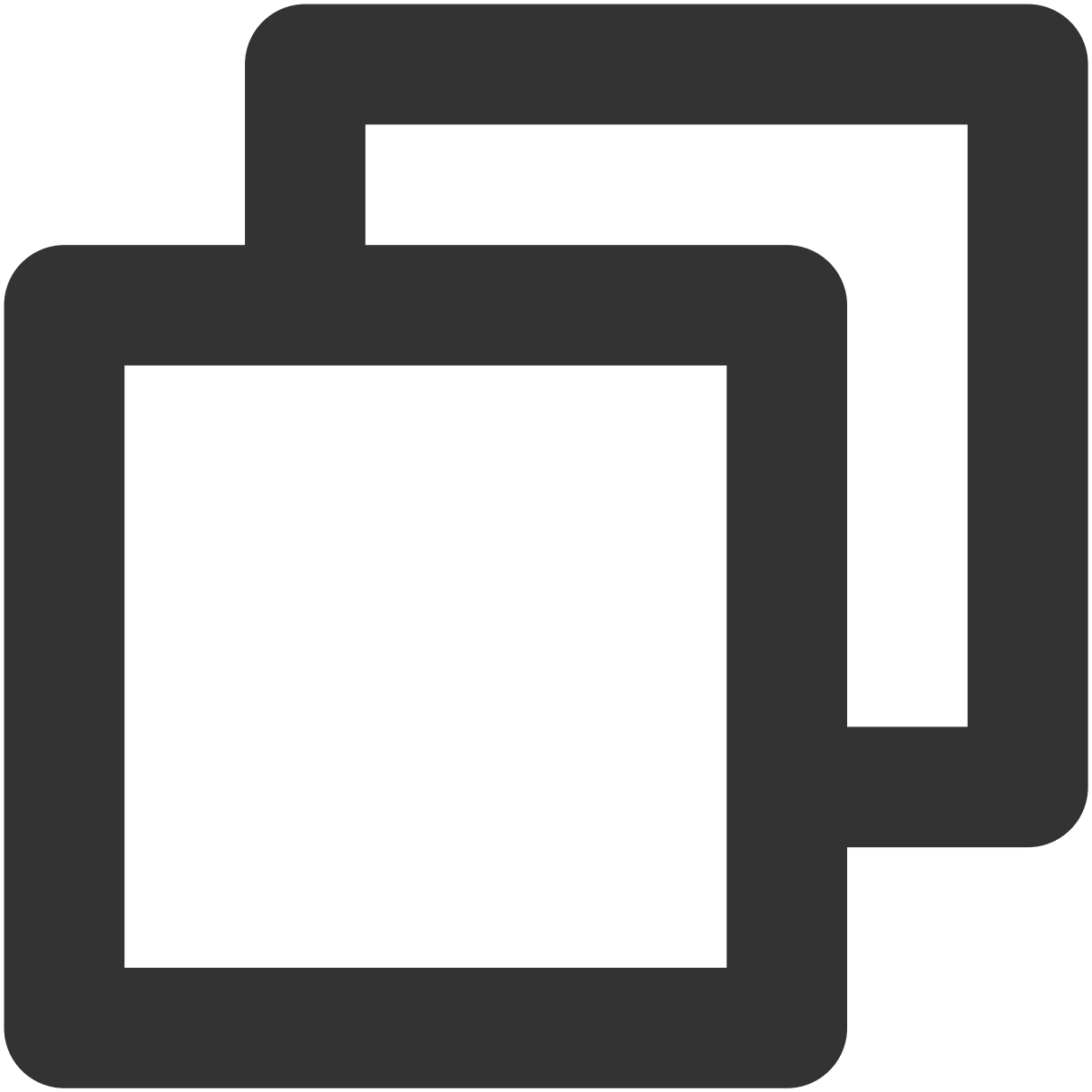
For specific definitions and usage of the parameters, see the Behavior.

6.1 Use in Components

When referenced by a component, behaviors are listed in the behaviors definition field.

Code example:

©2013-2022 Tencent Cloud. All rights reserved.

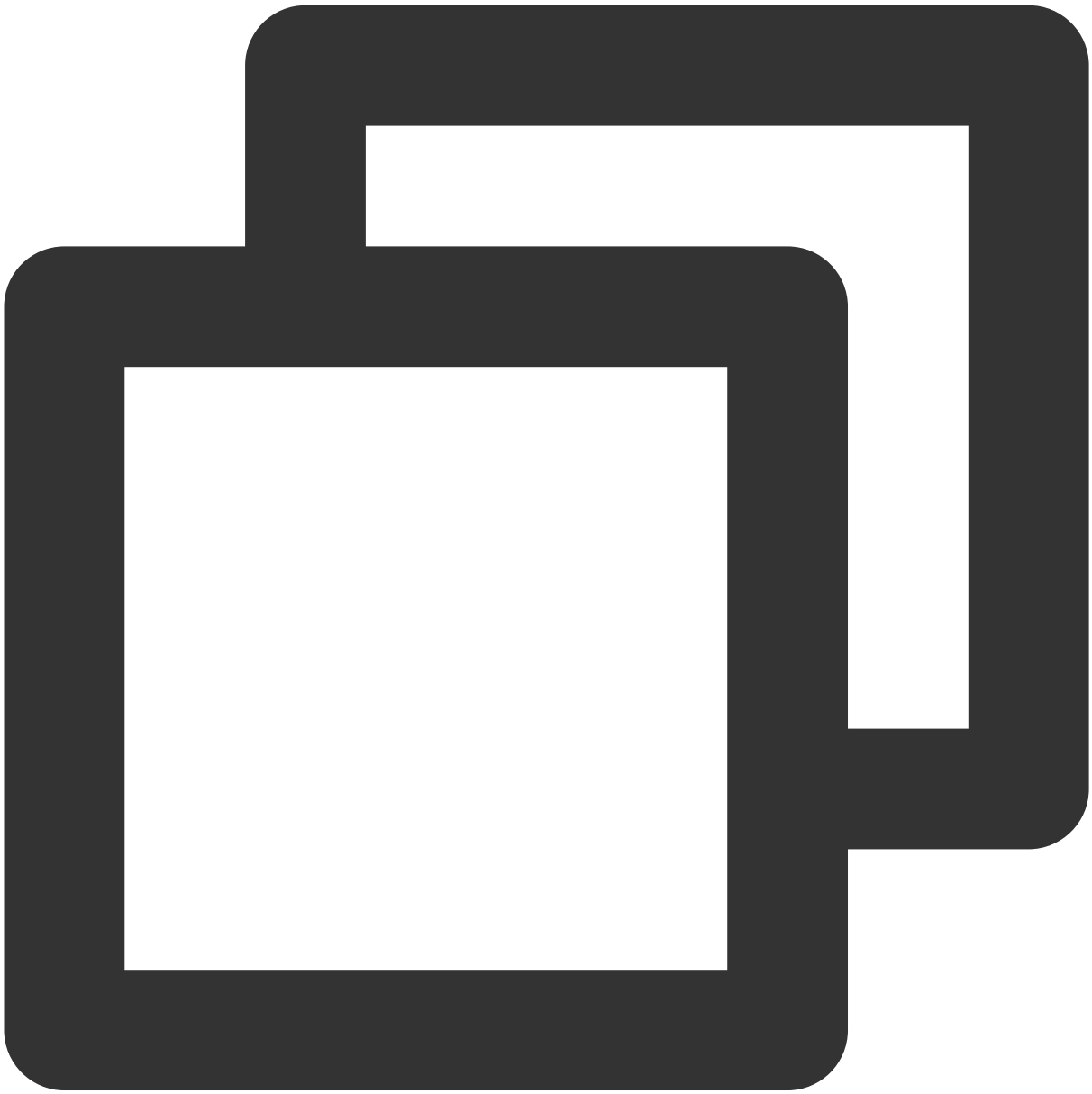


```
// my-behavior.js
module.exports = Behavior({
  behaviors: [],
  properties: {
    myBehaviorProperty: {
      type: String
    }
  },
  data: {
    myBehaviorData: {}
  },
```

©2013-2022 Tencent Cloud. All rights reserved.

```
  attached() {},
  methods: {
    myBehaviorMethod() {}
  }
})
```

When referencing components, just list them one by one in the behaviors definition section.



```
// my-component.js
var myBehavior = require('my-behavior')
Component({
  behaviors: [myBehavior],
```

©2013-2022 Tencent Cloud. All rights reserved.

```
  properties: {
    myProperty: {
      type: String
    }
  },
  data: {
    myData: 'my-component-data'
  },
  created: function () {
    console.log('[my-component] created')
  },
  attached: function () {
    console.log('[my-component] attached')
  },
  ready: function () {
    console.log('[my-component] ready')
  },
  methods: {
    myMethod: function () {
      console.log('[my-component] log by myMethod')
    },
  }
})
```

In the above example, my-behavior is added to the definition of the my-component component.

The structure of my-behavior is:

Property: myBehaviorProperty;

Data field: myBehaviorData;

method: myBehaviorMethod;

Lifecycle functions: attached, created, ready;

This will give my-component a final structure of:

Properties: myBehaviorProperty, myProperty;

Data fields: myBehaviorData, myData;

Methods: myBehaviorMethod, myMethod;

Lifecycle functions: attached, created, ready;

When the component triggers the lifecycle, the above example lifecycle functions are executed in the order:

1. [my-behavior] created;
2. [my-component] created;
3. [my-behavior] attached;
4. [my-component] attached;
5. [my-behavior] ready;
6. [my-component] ready;

For detailed rules, see Field Coverage and Combination Rules.

©2013-2022 Tencent Cloud. All rights reserved.

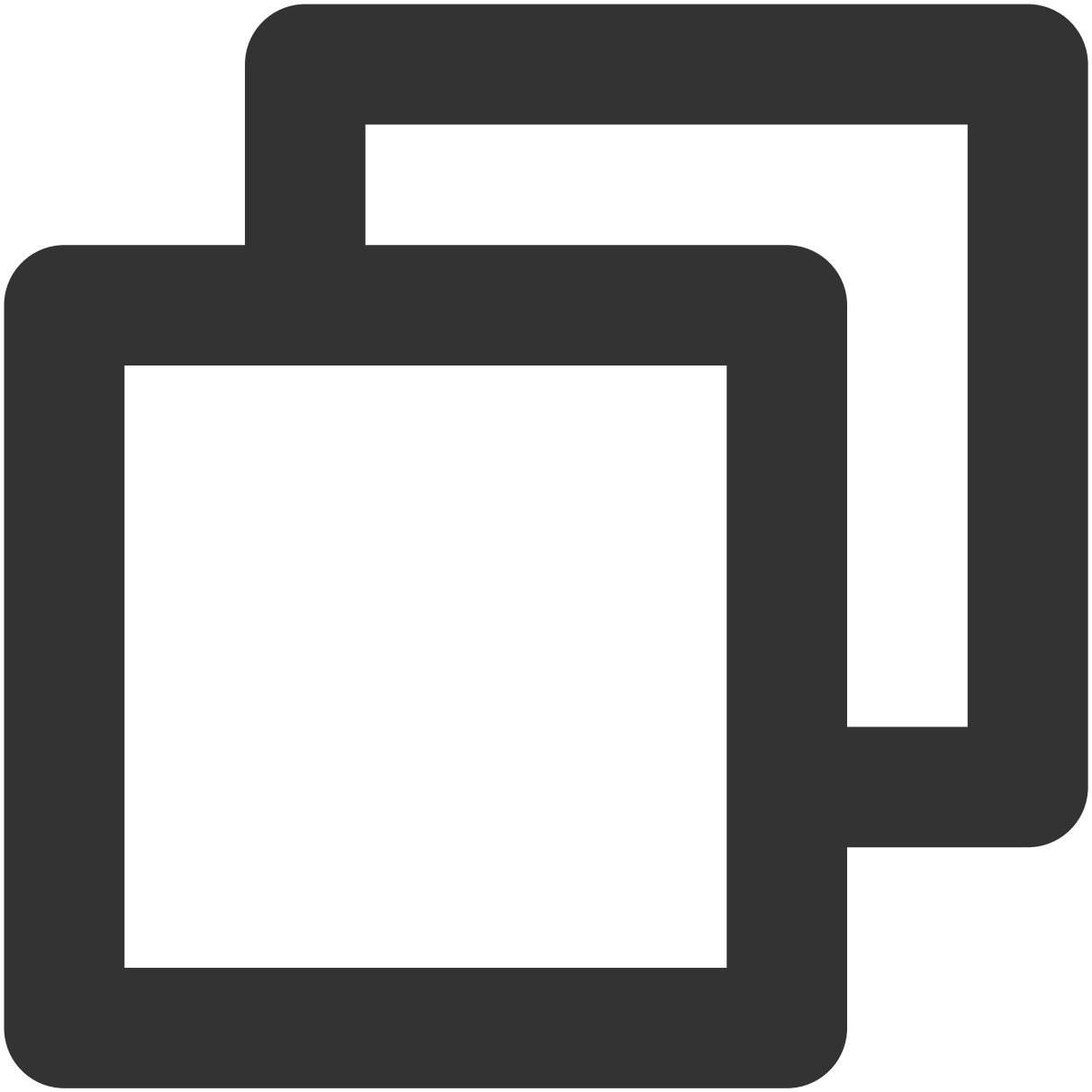
6.2 Field Coverage and Combination Rules

The component and the behaviour it references can contain fields with the same name, which are handled as follows:
If there are properties or methods with the same name:
If the component itself has the property or method, the property or method of the component overrides the property or method with the same name in the behavior;
If the component itself does not have this property or method, the property or method of the behaviour defined next in the behaviors field of the component overrides the property or method of the same name defined before it;
If there are nested references to behaviours in addition to 2, the rule is that the referencing behaviour overrides the property or method of the same name in the referenced behaviour.
If there are data fields with the same name:
If the data fields with the same name are both object types, object merging is performed;
In the rest of the cases, the data will be overwritten, and the overwriting rules are: referrer behaviour > referenced behaviour, backward behaviour > forward behaviour (higher priority overwrites lower priority, and the largest is the highest priority).
Lifecycle functions and observers do not override each other, but are called one by one at the corresponding trigger time:
For different lifecycle functions, the execution order of the component lifecycle functions is followed;
For the same kind of lifecycle function and the same field observers, the following rules are followed:
behaviour takes precedence over component execution;
The referenced behaviour takes precedence over the referrer behaviour;
The preceding behaviour takes precedence over the following behaviour;
If the same behaviour is referenced multiple times by a component, the lifecycle functions and observers defined by it will not be executed repeatedly.

6.3 Built-in Behaviors

Custom components can reference built-in behaviors to get some behaviors of built-in components.
Code example:

©2013-2022 Tencent Cloud. All rights reserved.



```
Component({
  behaviors: ['wx://form-field']
})
```

In the above example, `wx://form-field` represents a built-in behavior. It makes the custom component behave like a form control.

A built-in behavior generally adds properties to a component. Unless otherwise specified, components can overwrite these properties to change their type or add an observer.

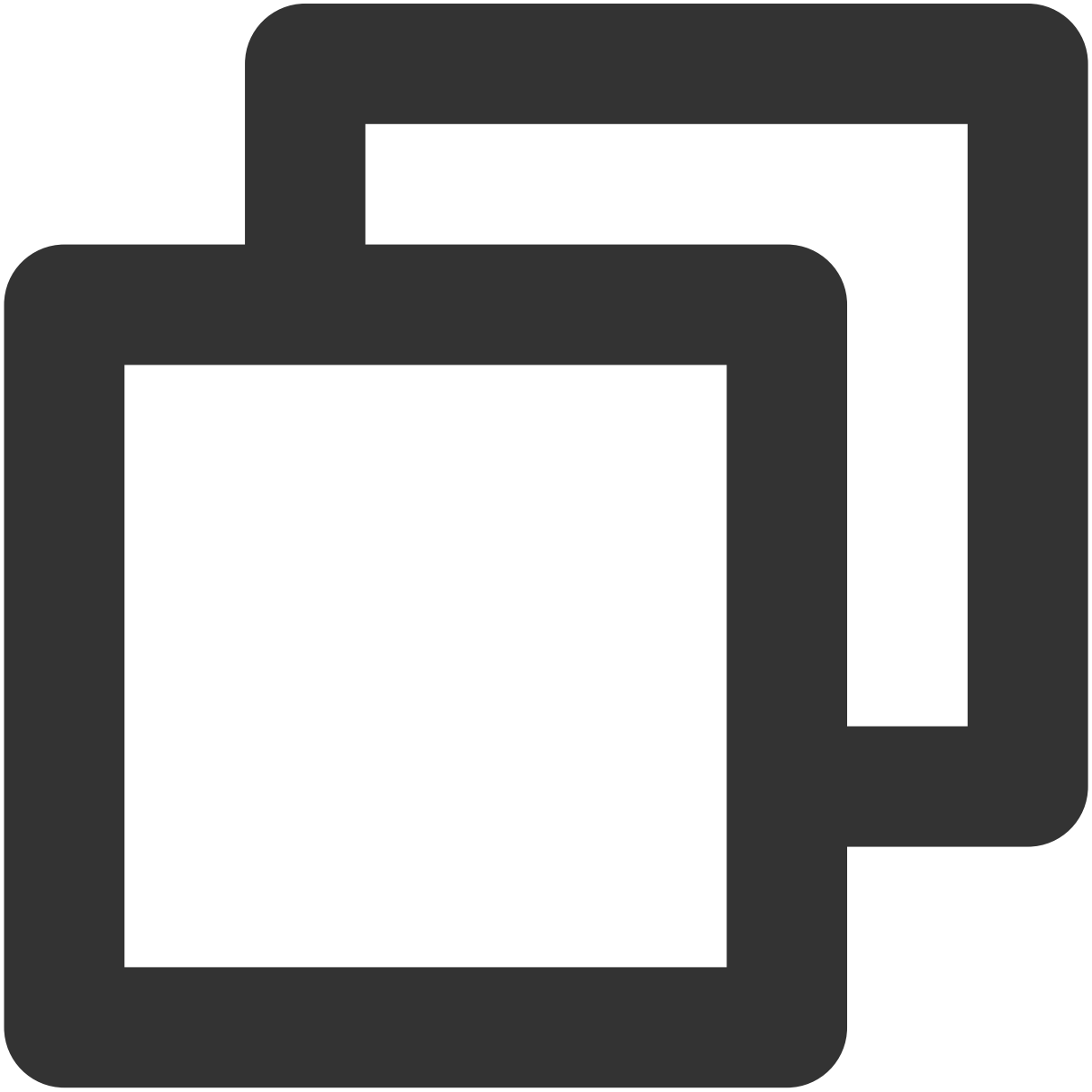
6.3.1 wx://form-field

©2013-2022 Tencent Cloud. All rights reserved.

Make custom components behave like form controls. The form component recognises these custom components and returns the component's field name and its corresponding field value in the submit event.

Property	Type	Description
name	String	Field name in the form
value	Any	Field value in the form

Code sample:



©2013-2022 Tencent Cloud. All rights reserved.

```
// custom-form-field.js
Component({
  behaviors: ['wx://form-field'],
  data: {
    value: ''
  },
  methods: {
    onChange: function (e) {
      this.setData({
        value: e.detail.value,
      })
    }
  }
 })
```

6.3.2 wx://component-export

Enables custom components to support the export definition field. This field can be used to specify the return value of the component when it is called by selectComponent.

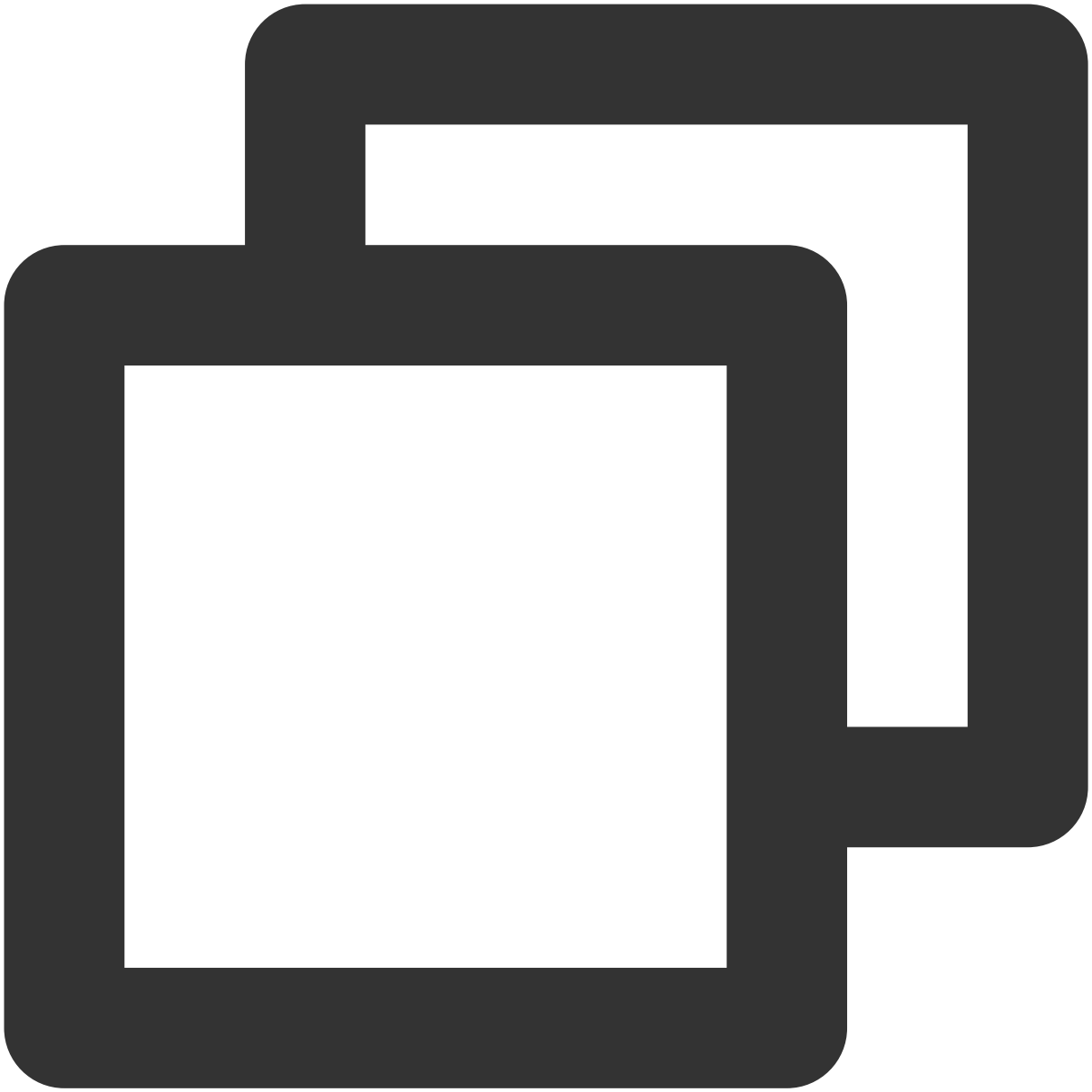
Detailed usage and code examples can be found in the selectComponent reference documentation.

7. Inter-component Relationship

7.1 Defining and Using Inter-component Relationship

Sometimes, you may need to implement components like this:

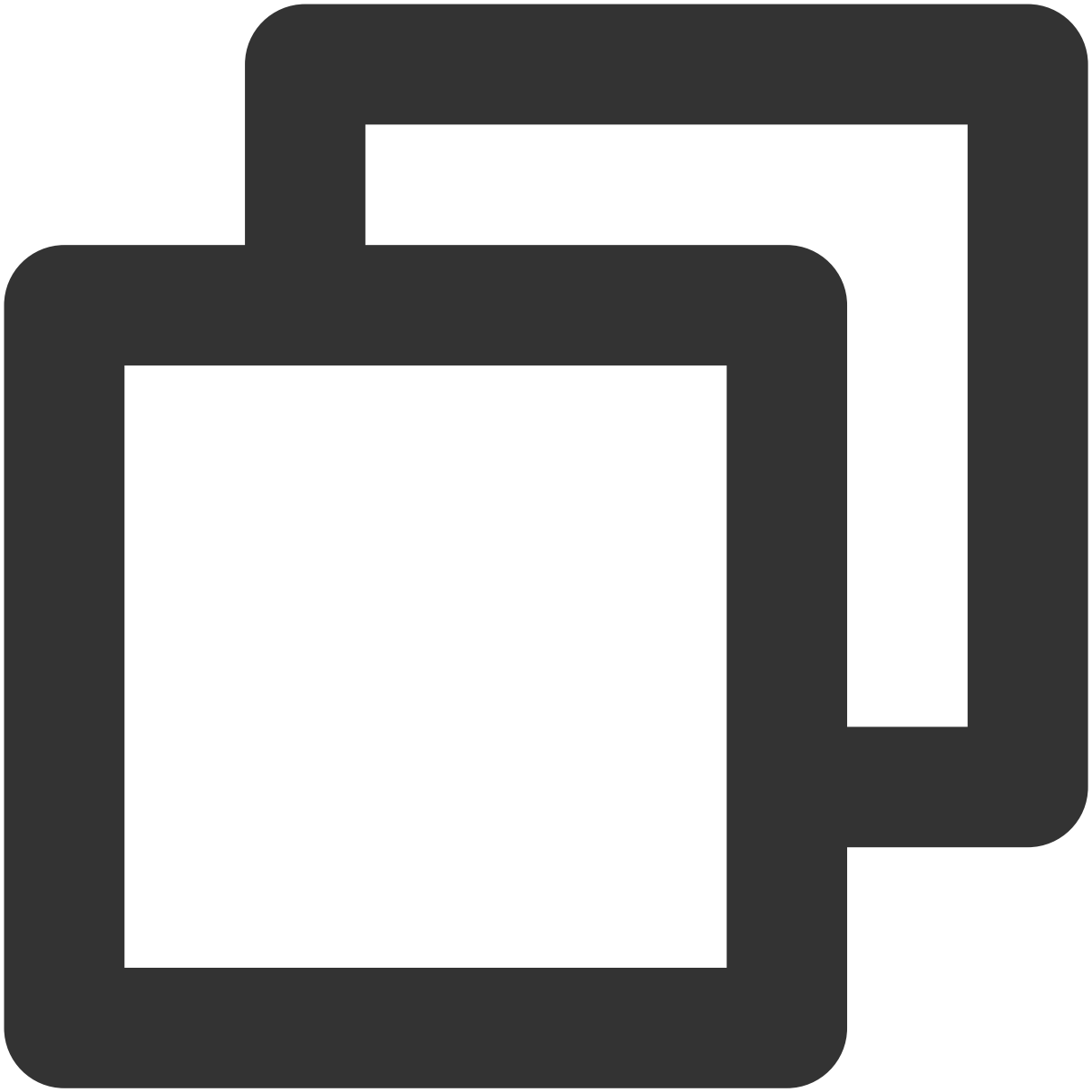
©2013-2022 Tencent Cloud. All rights reserved.



```
<custom-ul>
  <custom-li> item 1 </custom-li>
  <custom-li> item 2 </custom-li>
</custom-ul>
```

In this example, custom-ul and custom-li are custom components that are mutually related and communicate with each other in a complex manner. In this case, we can add the relations definition field when defining components. Example:

©2013-2022 Tencent Cloud. All rights reserved.

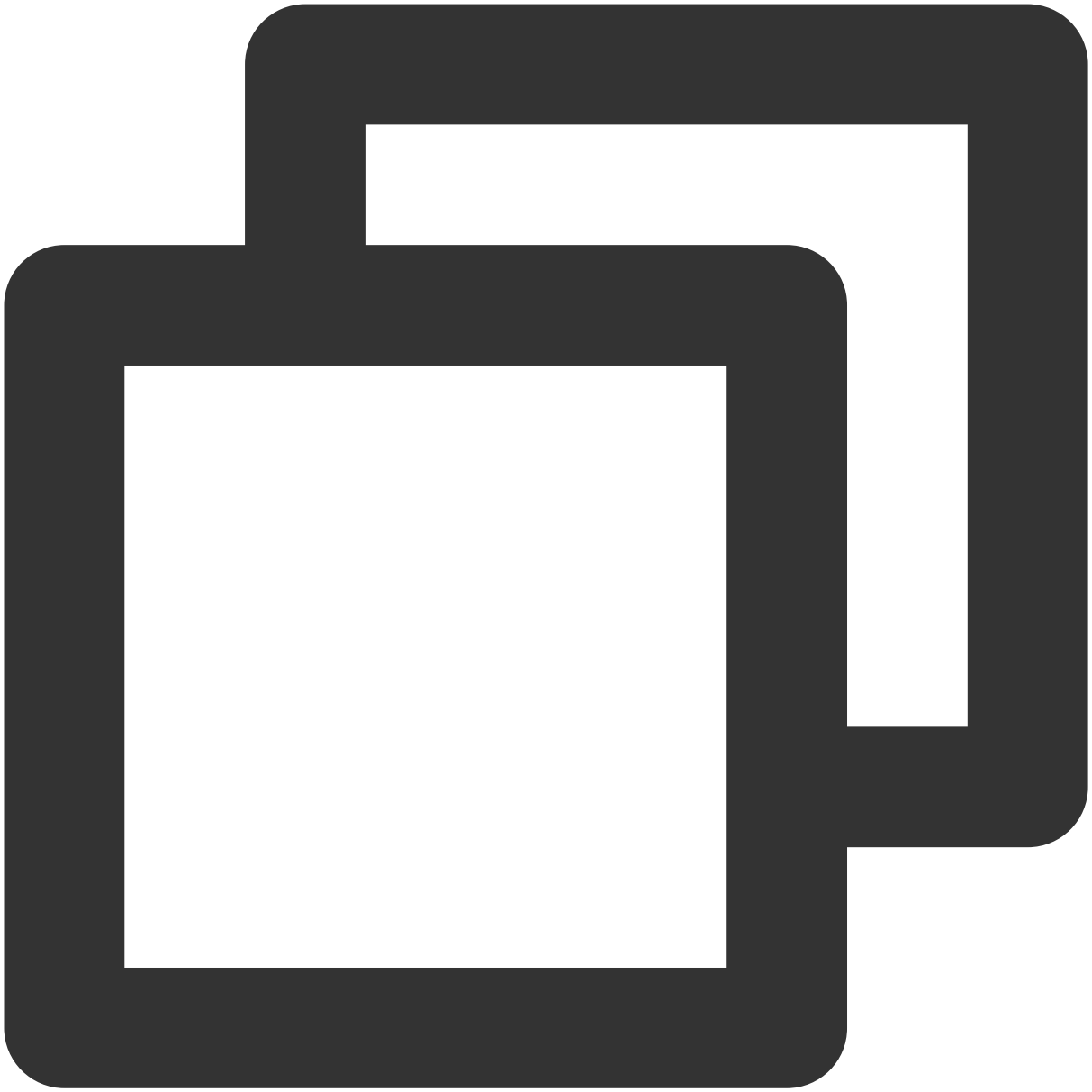


```
// path/to/custom-ul.js
Component({
  relations: {
    './custom-li': {
      type: 'child', // The linked target node should be a child node
      linked: function(target) {
        // Executed each time custom-li is inserted. "target" is the instance objec
      },
      linkChanged: function(target) {
        // Executed each time custom-li is moved. "target" is the instance object o
      },
```

©2013-2022 Tencent Cloud. All rights reserved.

```
      unlinked: function(target) {
        // Executed each time custom-li is removed. "target" is the instance object
      }
    }
  },
  methods: {
    _getAllLi: function(){
      // You can get the nodes array via getRelationNodes, including all linked cus
      var nodes = this.getRelationNodes('path/to/custom-li')
    }
  },
  ready: function(){
    this._getAllLi()
  }
})
```

©2013-2022 Tencent Cloud. All rights reserved.



```
// path/to/custom-li.js
Component({
  relations: {
    './custom-ul': {
      type: 'parent', // The linked target node should be a parent node
      linked: function(target) {
        // Executed after each insertion to custom-ul. "target" is the instance obj
      },
      linkChanged: function(target) {
        // Executed after each move. "target" is the instance object of the custom-
      },
```

©2013-2022 Tencent Cloud. All rights reserved.

```
        unlinked: function(target) {
          // Executed after each removal. "target" is the instance object of the cust
        }
      }
    }
  })
```

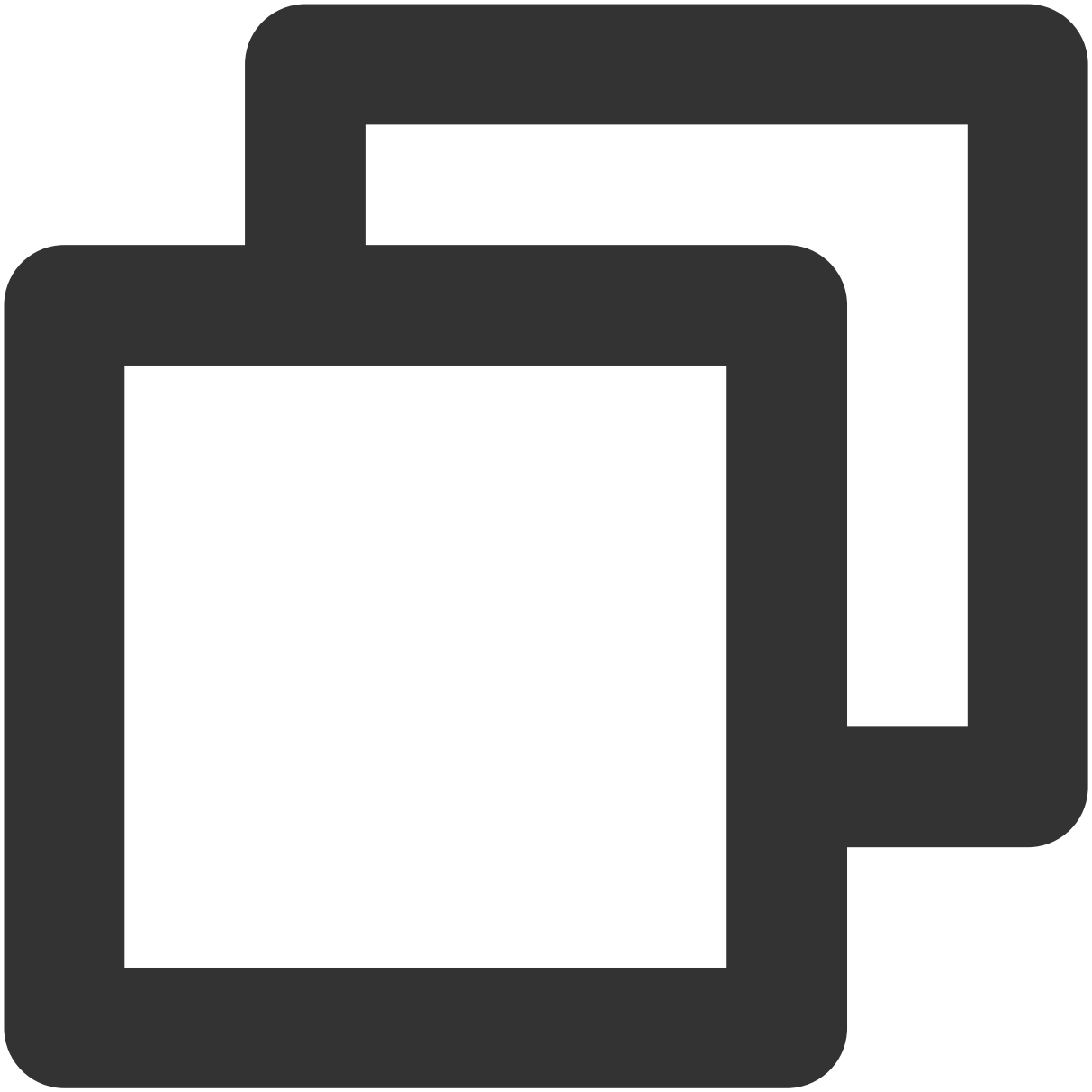
Note：

The relations definition must be added in the definitions of both components. Otherwise, it will be invalid.

7.2 Linking Components of the Same Class

Sometimes, you need to link components of the same class. For example:

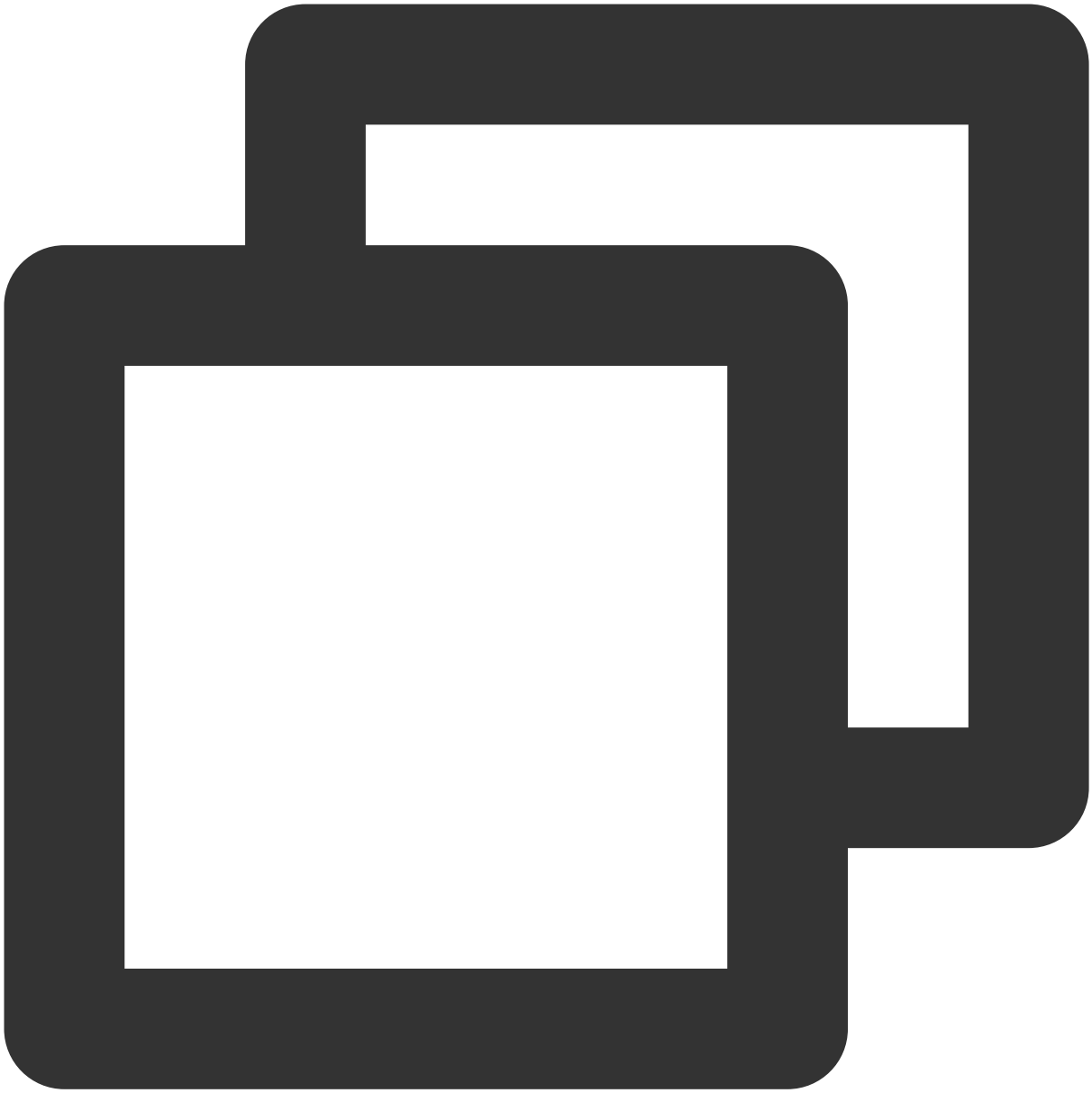
©2013-2022 Tencent Cloud. All rights reserved.



```
<custom-form>
  <view>
    input
    <custom-input></custom-input>
  </view>
  <custom-submit> submit </custom-submit>
</custom-form>
```

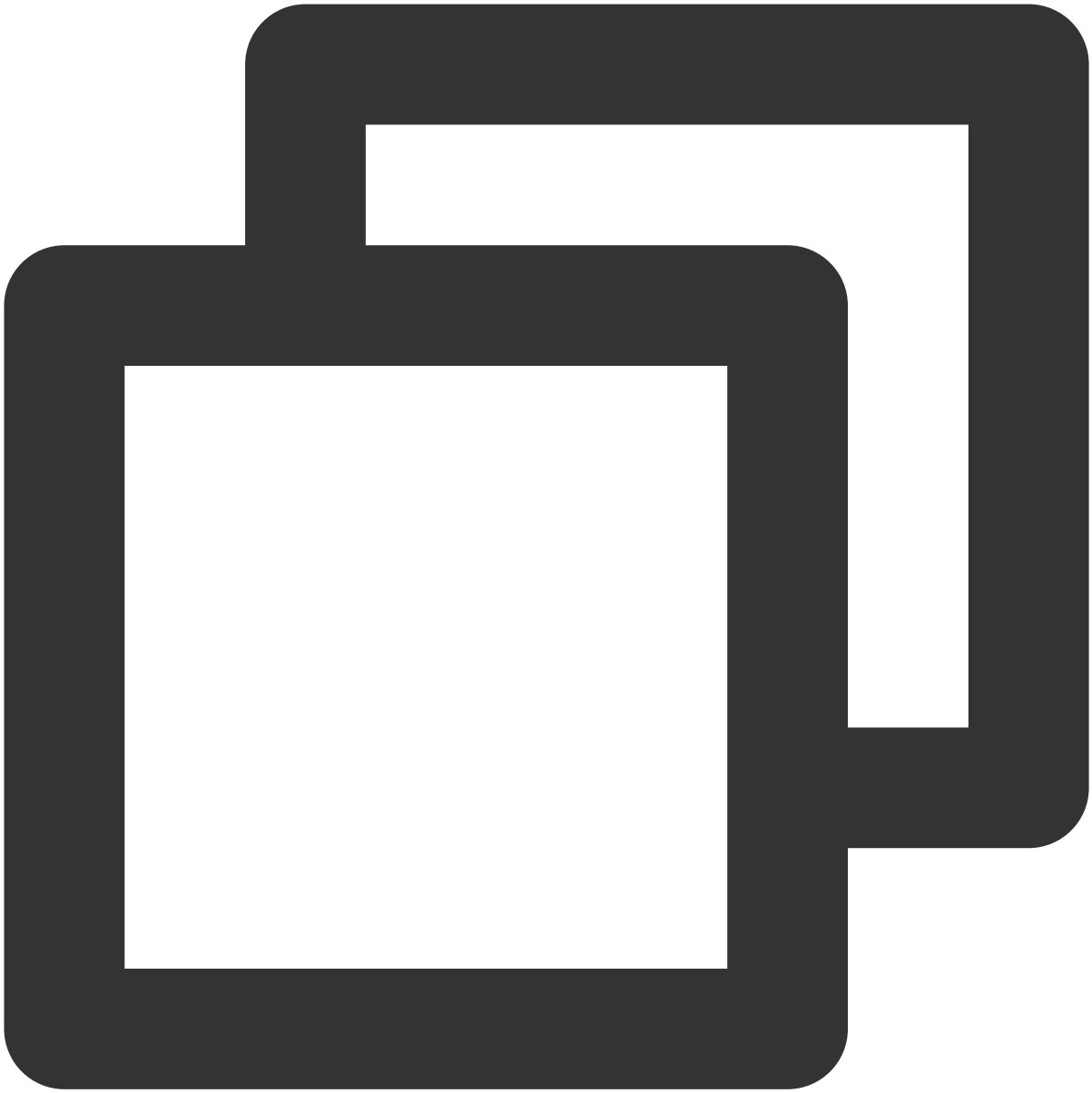
The custom-form component wants to link the custom-input and custom-submit components. In this case, if the two components have the same behavior:

©2013-2022 Tencent Cloud. All rights reserved.



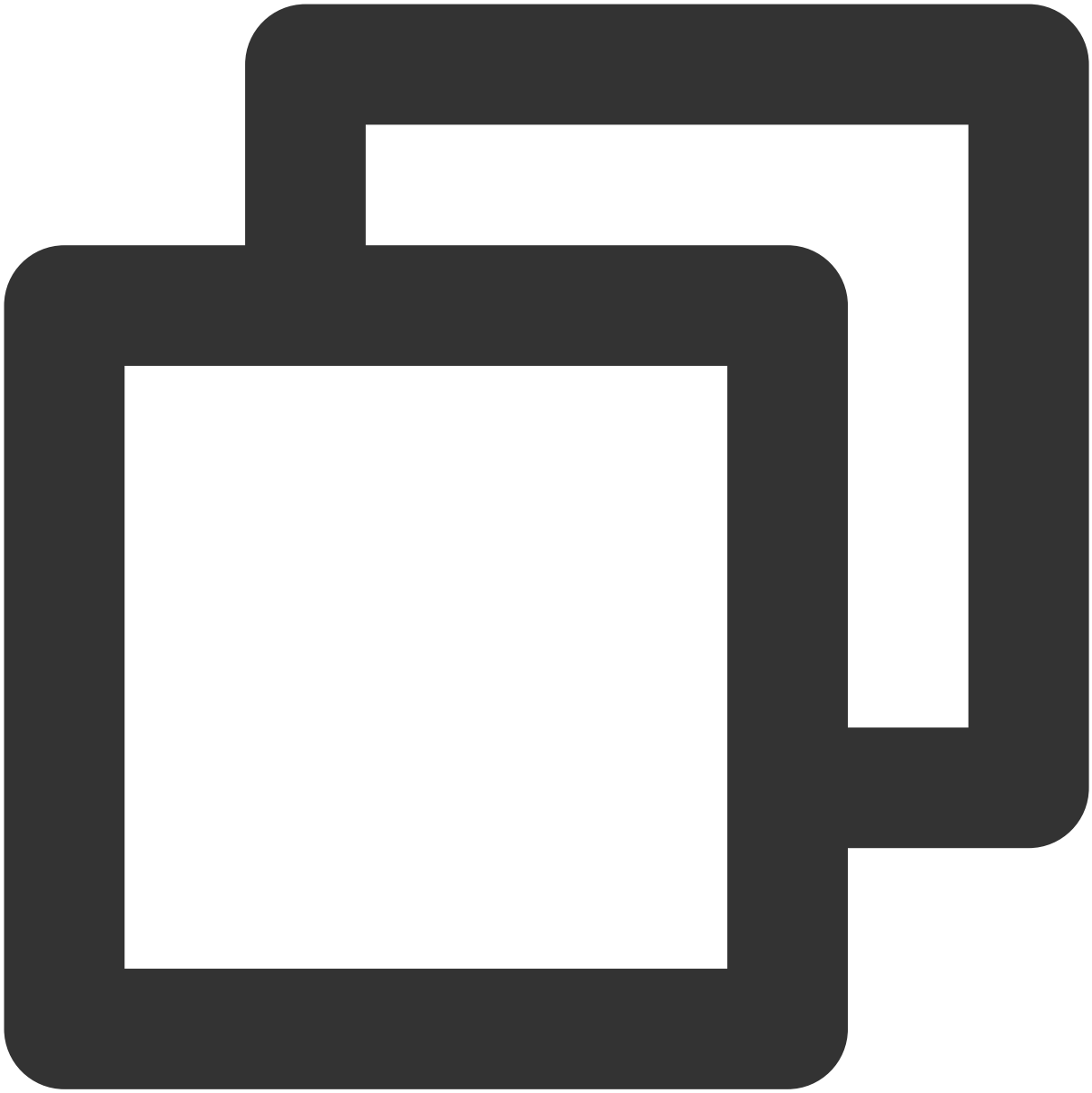
```
// path/to/custom-form-controls.js
module.exports = Behavior({
  // ...
})
```

©2013-2022 Tencent Cloud. All rights reserved.



```
// path/to/custom-input.js
var customFormControls = require('./custom-form-controls')
Component({
  behaviors: [customFormControls],
  relations: {
    './custom-form': {
      type: 'ancestor', // he linked target node should be an ancestor node
    }
  }
})
```

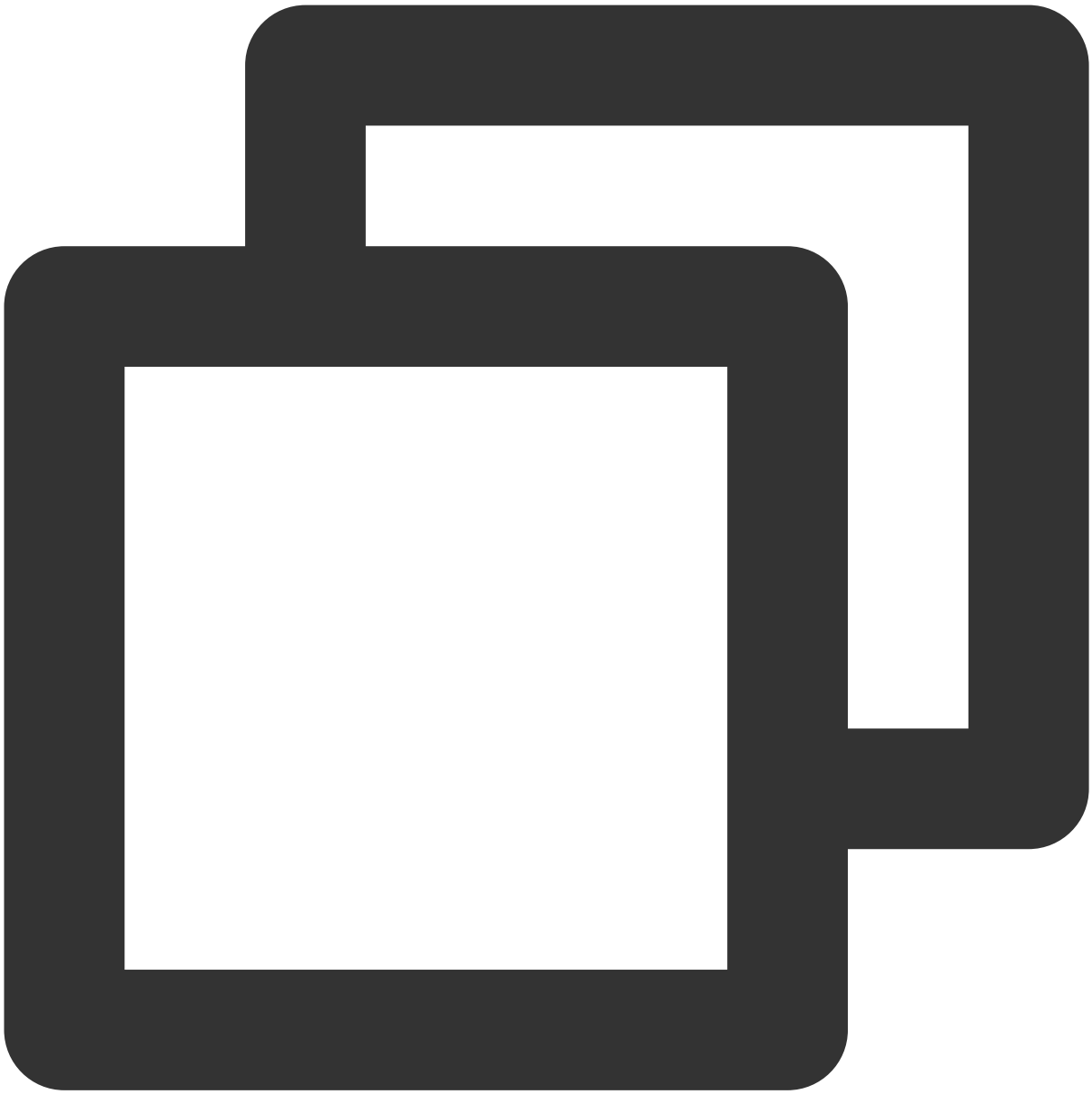
©2013-2022 Tencent Cloud. All rights reserved.



```
// path/to/custom-submit.js
var customFormControls = require('./custom-form-controls')
Component({
  behaviors: [customFormControls],
  relations: {
    './custom-form': {
      type: 'ancestor', // The linked target node should be an ancestor node
    }
  }
})
```

©2013-2022 Tencent Cloud. All rights reserved.

Then, in the relations definition, this behavior can be used to replace the component path as the linked target node:



```
// path/to/custom-form.js
var customFormControls = require('./custom-form-controls')
Component({
  relations: {
    'customFormControls': {
      type: 'descendant', // The linked target node should be a descendant node
      target: customFormControls
    }
  }
```

©2013-2022 Tencent Cloud. All rights reserved.

```
})
```

7.2 relations Definition Field

The `relations` definition field includes the target component path and its corresponding options. Available options are listed below.

Option	Type	Required	Description
type	String	Yes	The relative relationship of the target component, including parent, child, ancestor, and descendant.
linked	Function	No	A relationship lifecycle function, triggered when the relationship is created in the page node tree after the component's attached lifecycle.
linkChanged	Function	No	A relationship lifecycle function, triggered when the relationship is changed in the page node tree after the component's moved lifecycle.
unlinked	Function	No	A relationship lifecycle function, triggered when the relationship is removed from the page node tree after the component's detached lifecycle.
target	String	No	If this option is set, it indicates a behavior that should be possessed by the linked target node. All component nodes that have this behavior will be linked.

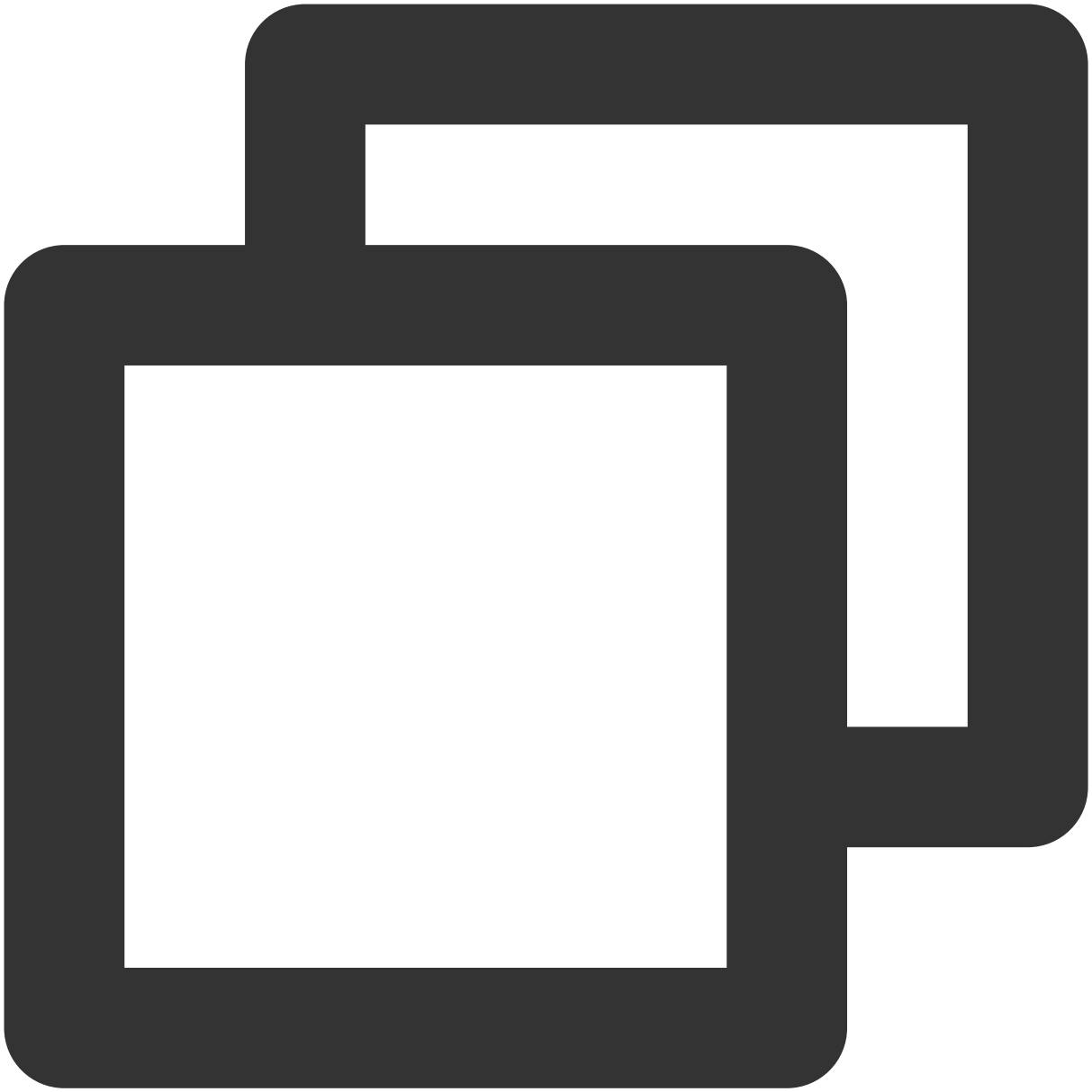
8. Data Listener

Data listener is used to monitor and respond to changes in any properties or data fields.

8.1 Using Data Listener

When certain data fields are set via setData, some operations must be executed.
For example, this.data.sum is always the sum of this.data.numberA and this.data.numberB. In this case, you can use a data listener to implement the following operation.

©2013-2022 Tencent Cloud. All rights reserved.



```
Component({
  attached: function() {
    this.setData({
      numberA: 1,
      numberB: 2,
    })
  },
  observers: {
    'numberA, numberB': function(numberA, numberB) {
      // Execute this function when numberA or numberB is set
      this.setData({
```

©2013-2022 Tencent Cloud. All rights reserved.

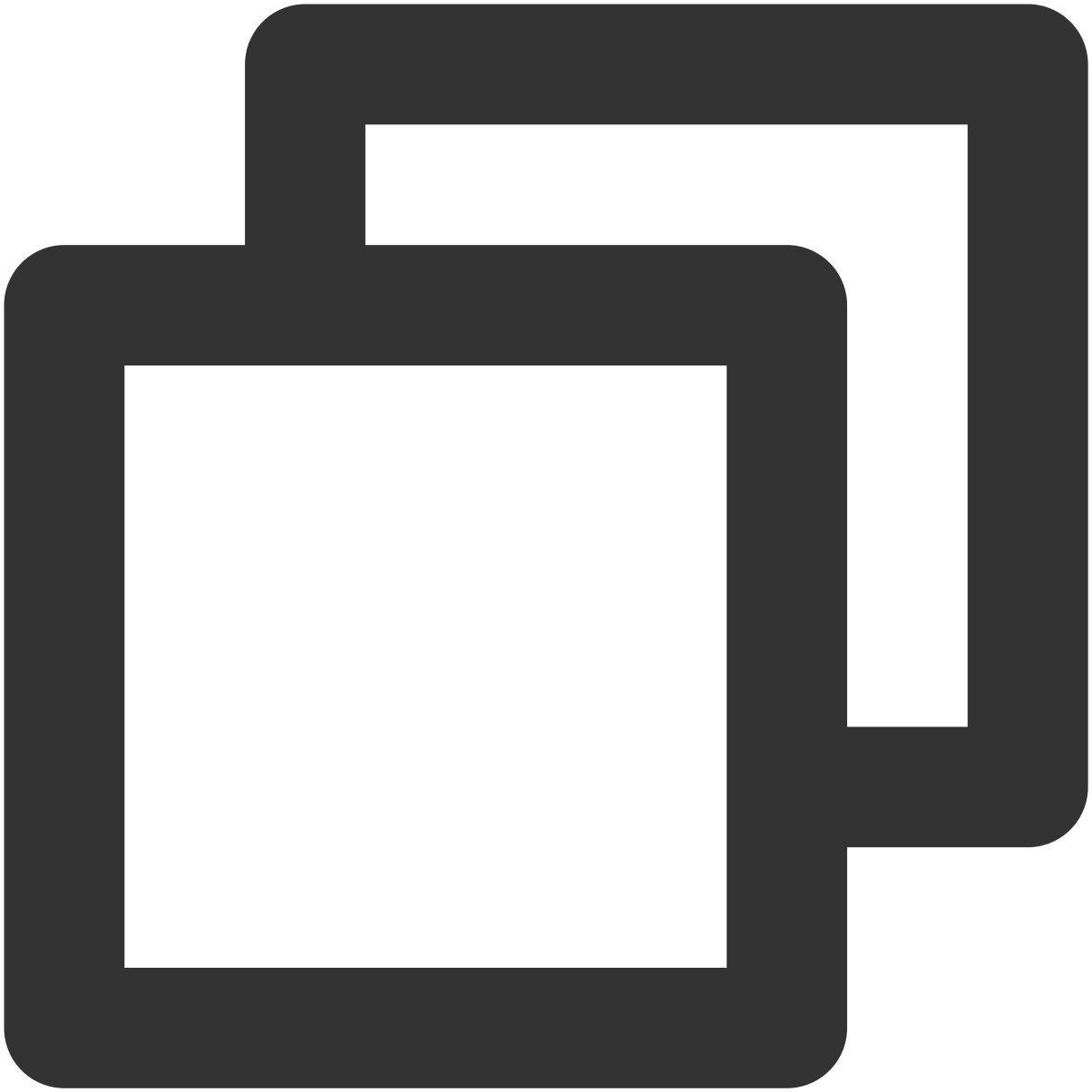
```
          sum: numberA + numberB
        })
      }
    }
})
```

8.2 Field Listening Syntax

Data listeners can monitor multiple changes to properties or internal data simultaneously. A listener can only be triggered one time by a setData operation.

Listeners can also monitor sub-data fields, as shown below.

©2013-2022 Tencent Cloud. All rights reserved.

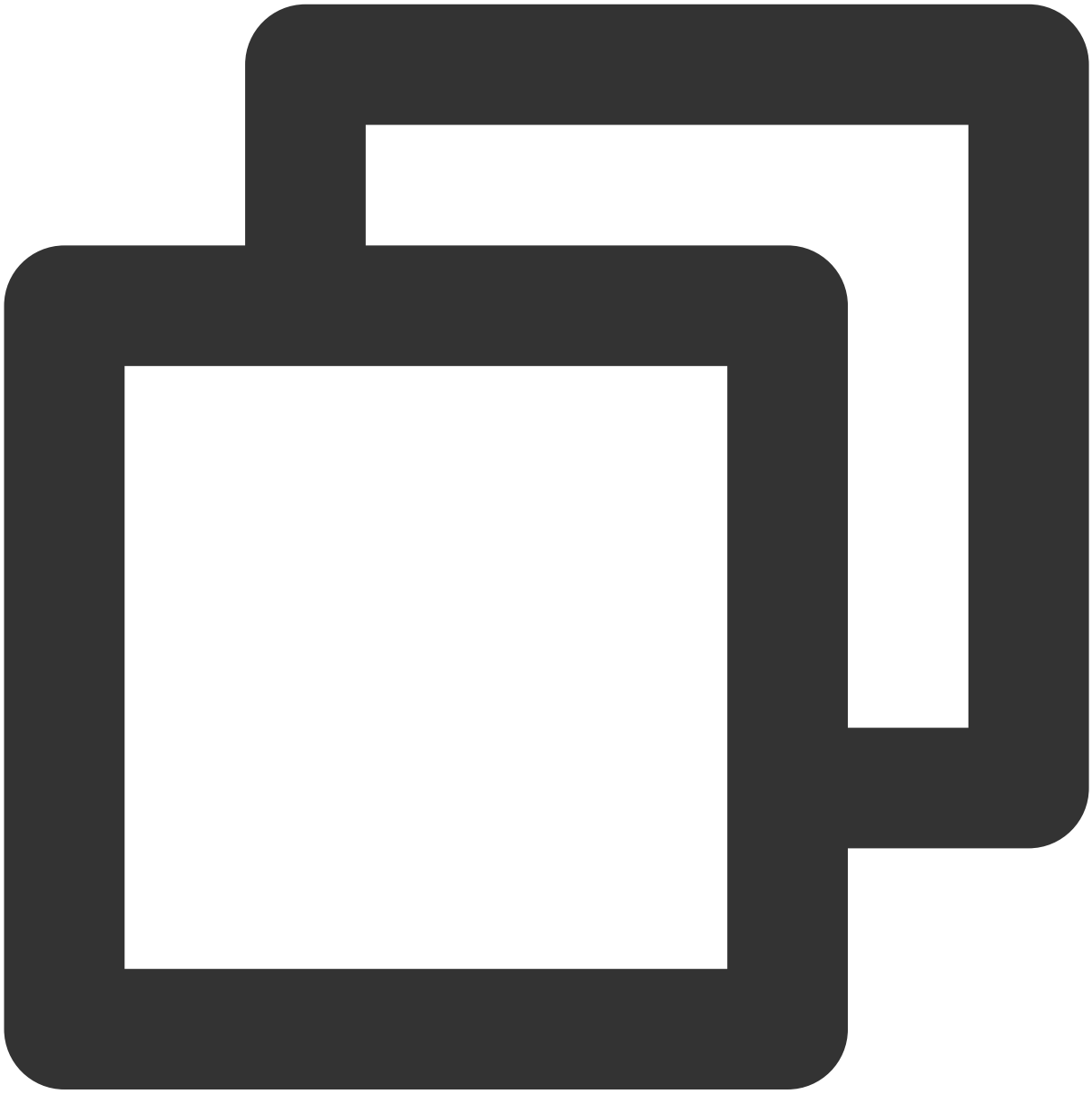


```
Component({
  observers: {
    'some.subfield': function(subfield) {
      // Triggered when this.data.some.subfield is set via setData
      // (Also triggered when this.data.some is set via setData)
      subfield === this.data.some.subfield
    },
    'arr[12]': function(arr12) {
      // Triggered when this.data.arr[12] is set via setData
      // (Also triggered when this.data.arr is set via setData)
      arr12 === this.data.arr[12]
```

©2013-2022 Tencent Cloud. All rights reserved.

```
    },
  }
})
```

To monitor changes to all sub-data fields, you can use a wildcard `**` .



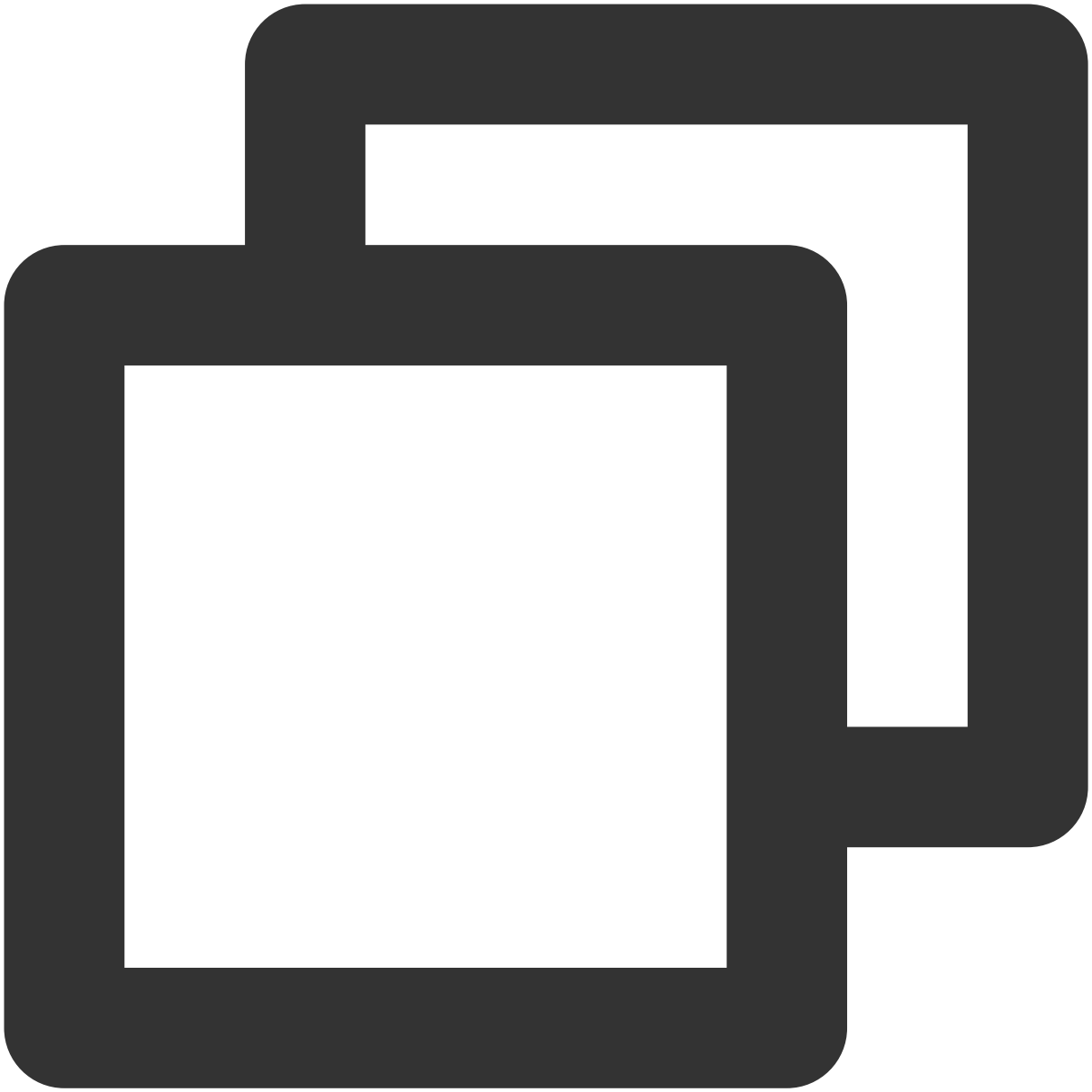
```
Component({
  observers: {
    'some.field.**': function(field) {
      // Triggered when setData is used to set this.data.some.field itself or any o
      // (Also triggered when this.data.some is set via setData)
```

©2013-2022 Tencent Cloud. All rights reserved.

```
        field === this.data.some.field
      },
    },
    attached: function() {
      // This will trigger the above observer
      this.setData({
        'some.field': { /* ... */ }
      })
      // This will also trigger the above observer
      this.setData({
        'some.field.xxx': { /* ... */ }
      })
      // This will still trigger the above observer
      this.setData({
        'some': { /* ... */ }
      })
    }
  })
```

You can simply use the wildcard `**` to monitor all setData.

©2013-2022 Tencent Cloud. All rights reserved.



```
Component({
  observers: {
    '**': function() {
      // Triggered upon each setData operation
    },
  },
})
```

8.3 Bugs & Tips:

©2013-2022 Tencent Cloud. All rights reserved.

Data listeners monitor the data fields set via setData. Even if the values of these data fields do not change, the data listeners will still be triggered;
Be aware that, if you use setData to set the monitored data fields in the data listener function, this can create an infinite loop;
Compared to a property observer, a data listener is more powerful with better performance.

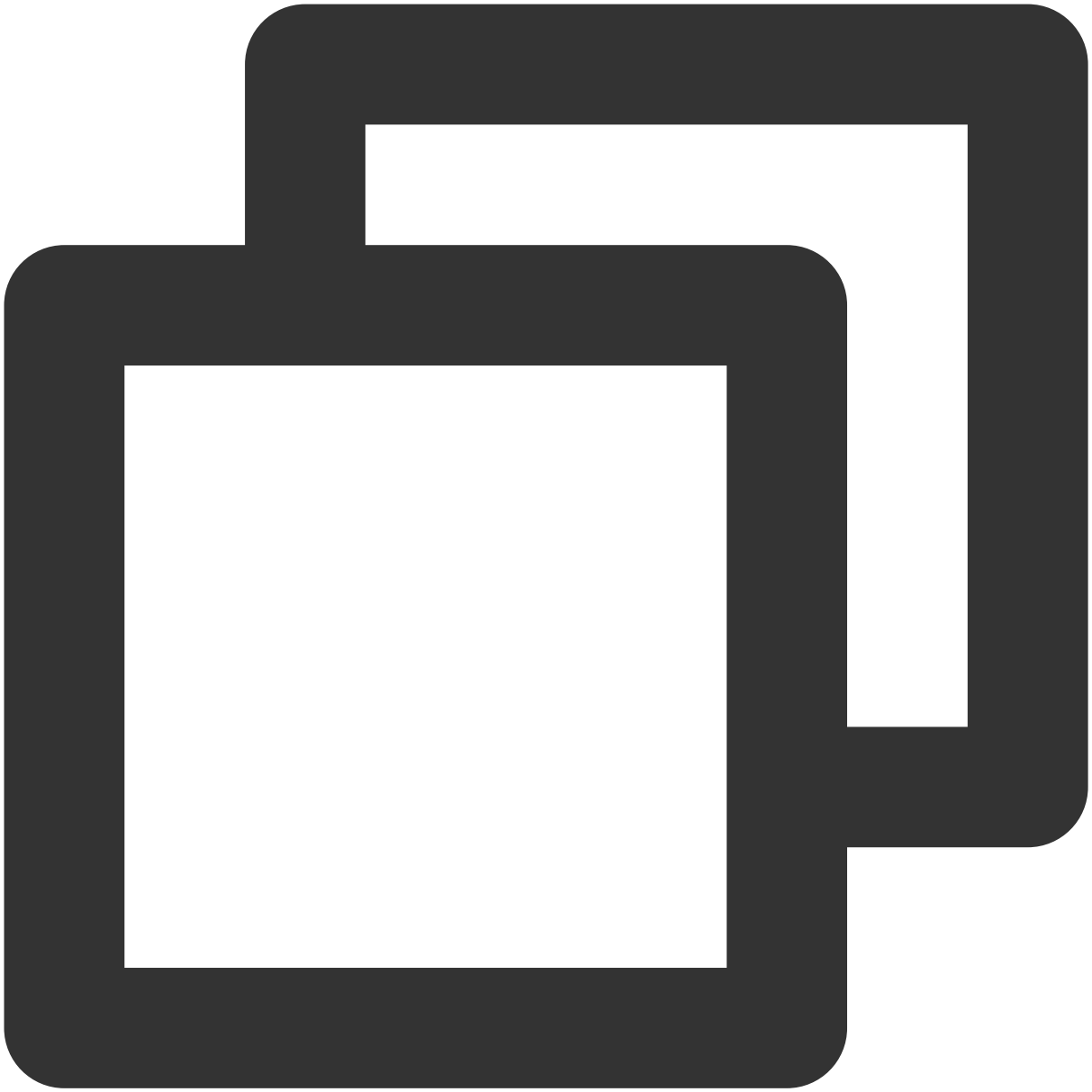
9. Data-only field

Pure data fields are data fields that are not used for interface rendering and can be used to improve page update performance.

9.1 Plain data fields in component data

In some cases, some fields in data (including those set by setData) are neither displayed on the interface nor passed to other components, but are only used internally by the current component.
In this case, you can specify such data fields as "pure data fields", which will only be recorded in this.data and will not be involved in any interface rendering process, which can help improve the page update performance.
The way to specify "pure data fields" is to specify pureDataPattern as a regular expression in the options definition section of the Component constructor, and fields whose names match this regular expression will become pure data fields.
Code example:

©2013-2022 Tencent Cloud. All rights reserved.

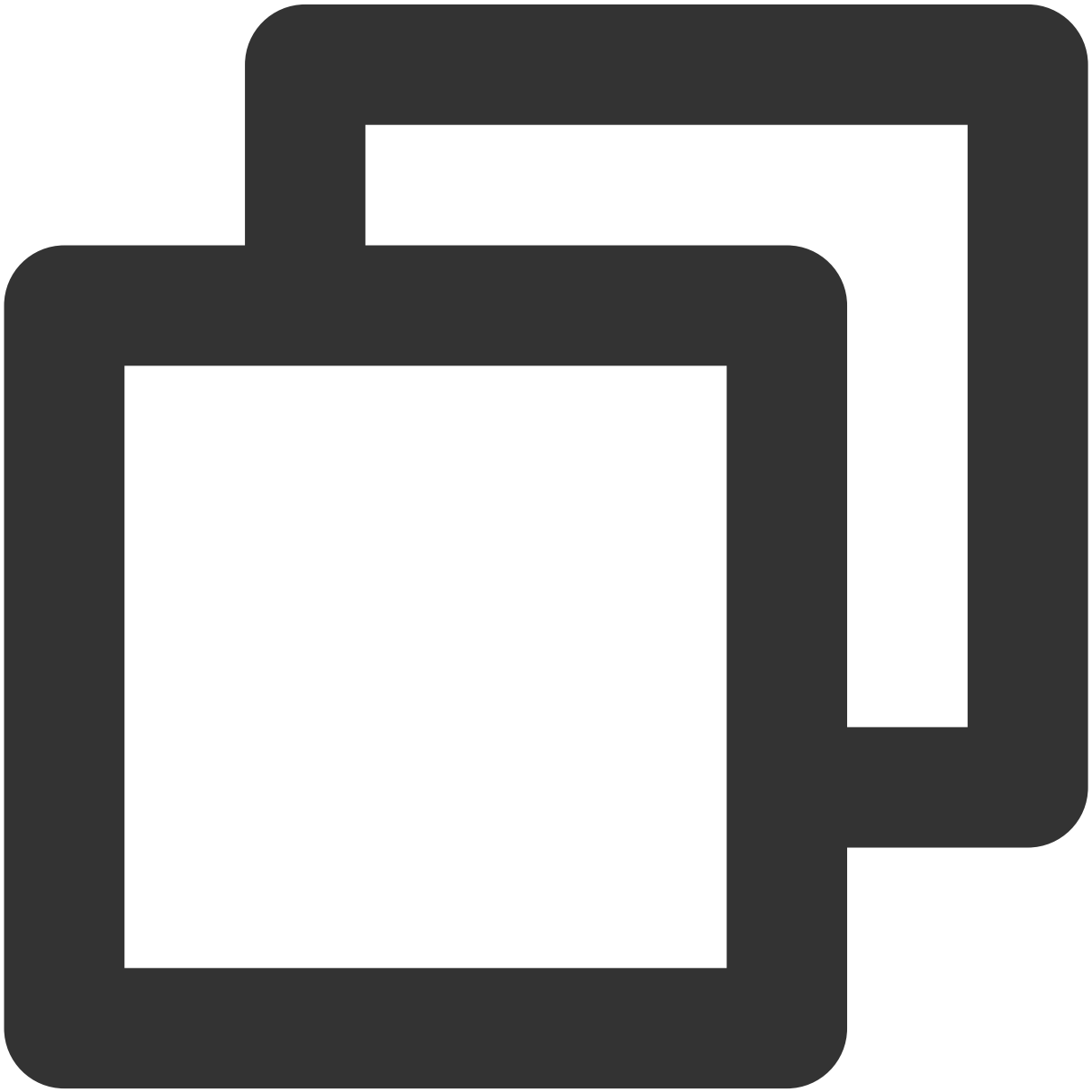


```
Component({
  options: {
    pureDataPattern: /^_/ // Specify all data fields starting with _ as pure data f
  },
  data: {
    a: true, // Plain data field
    _b: true, // Plain data field
  }, methods: { a: true, // Normal data field _b: true, // Plain data field
  methods: {
    myMethod() {
      this.data._b // Plain data fields can be retrieved in this.data
```

©2013-2022 Tencent Cloud. All rights reserved.

```
      this.setData({
        c: true, // Plain data field
        _d: true, // Plain data field
      })
    }
  }
})
```

Plain data fields in the above components will not be applied to WXML:



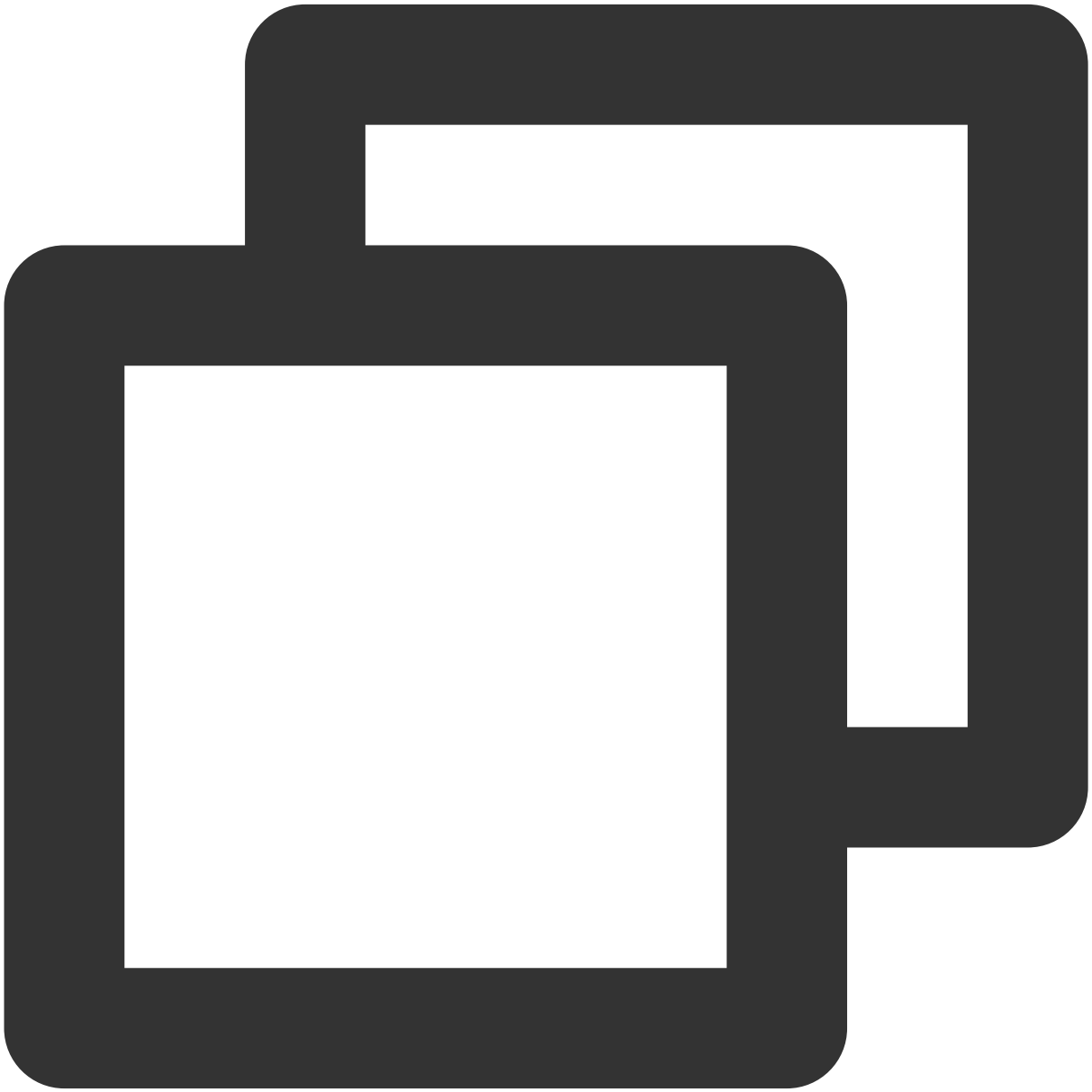
```
<view wx:if="{{a}}"> This line will be displayed </view>
<view wx:if="{{_b}}"> This line will not be shown </view>
```

©2013-2022 Tencent Cloud. All rights reserved.

9.2 Pure Data Fields in Component Properties

Properties can also be specified as pure data fields (regular expressions that follow the pureDataPattern).
A pure data field in a property can receive external incoming property values just like a normal property, but it cannot be used directly in the component's own WXML.

Code Example:



```
Component({
  options: {
```

©2013-2022 Tencent Cloud. All rights reserved.

```
    pureDataPattern: /^_/
  },
  properties: {
    a: Boolean,
    _b: {
      type: Boolean,
      observer() {
        // Don't do this! This observer will never be triggered
      }
    },
  }
})
```

Note：

The property observer is never triggered for data-only fields in properties! If you want to listen for property value changes, use a data listener instead.

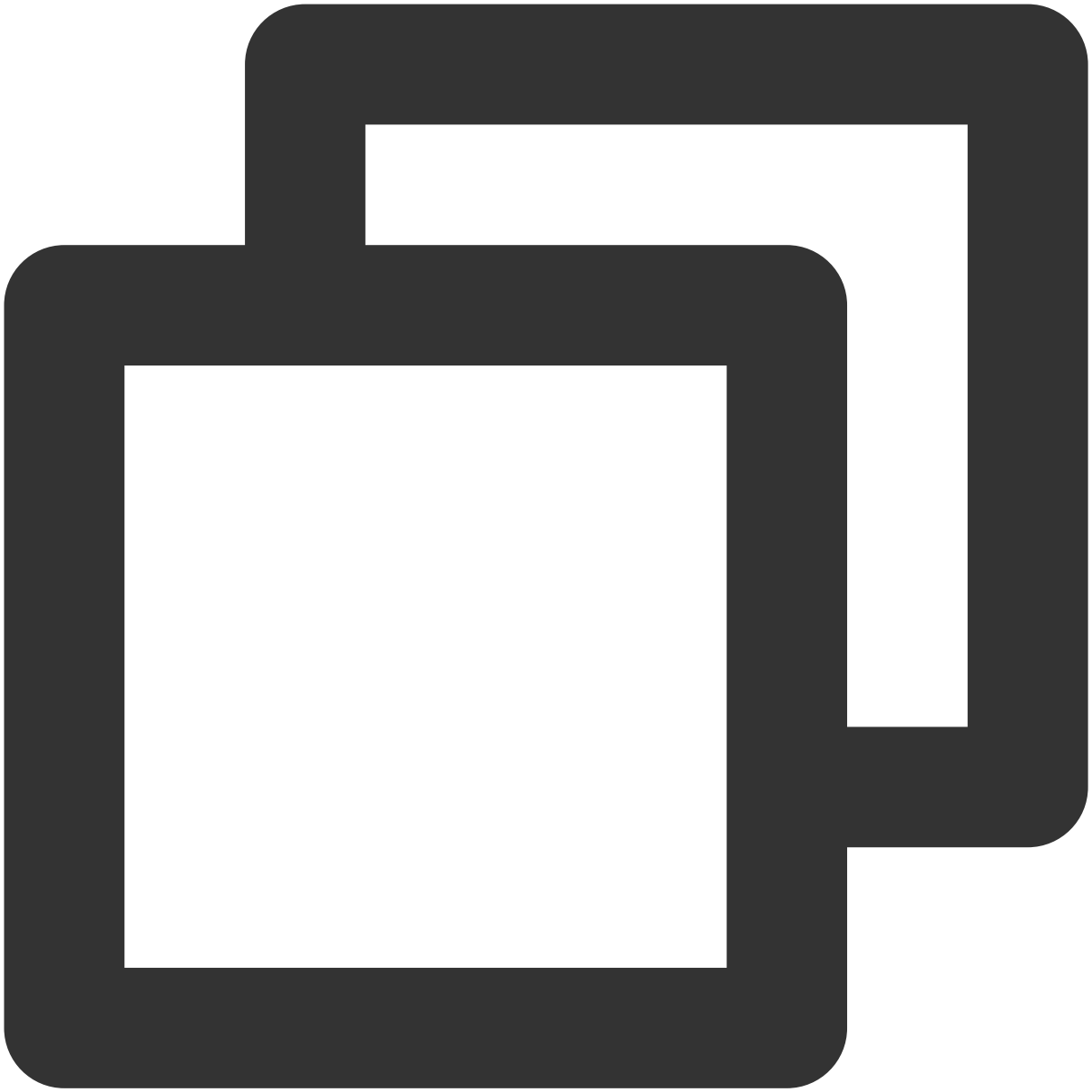
9.3 Listening to Pure Data Fields with Data Listeners

data listener can be used to listen to plain data fields (as with normal data fields). In this way, the interface can be changed by listening and responding to changes in plain data fields.

The following example is a custom component that converts a JavaScript timestamp to a readable time.

Code example:

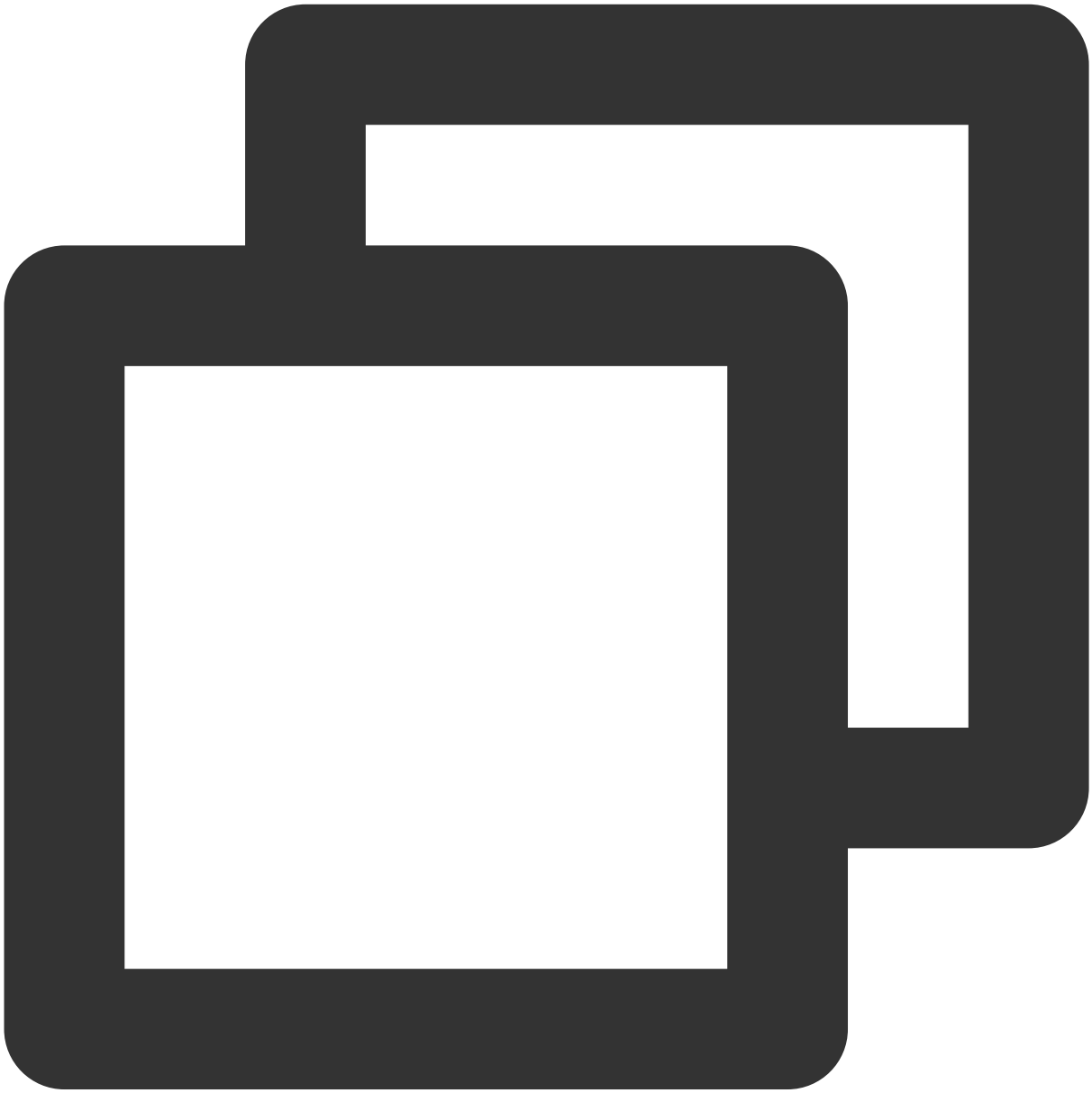
©2013-2022 Tencent Cloud. All rights reserved.



```
Component({
  options: {
    pureDataPattern: /^timestamp$/ // Specifying the timestamp attribute as a data-
  },
  properties: {
    timestamp: Number,
  },
  observers: {
    timestamp: function () {
      // When timestamp is set, display it as a readable time string.
      var timeString = new Date(this.data.timestamp).toLocaleString()
```

©2013-2022 Tencent Cloud. All rights reserved.

```
        this.setData({
          timeString: timeString
        })
      }
    }
})
```



```
<view>{{timeString}}</view>
```

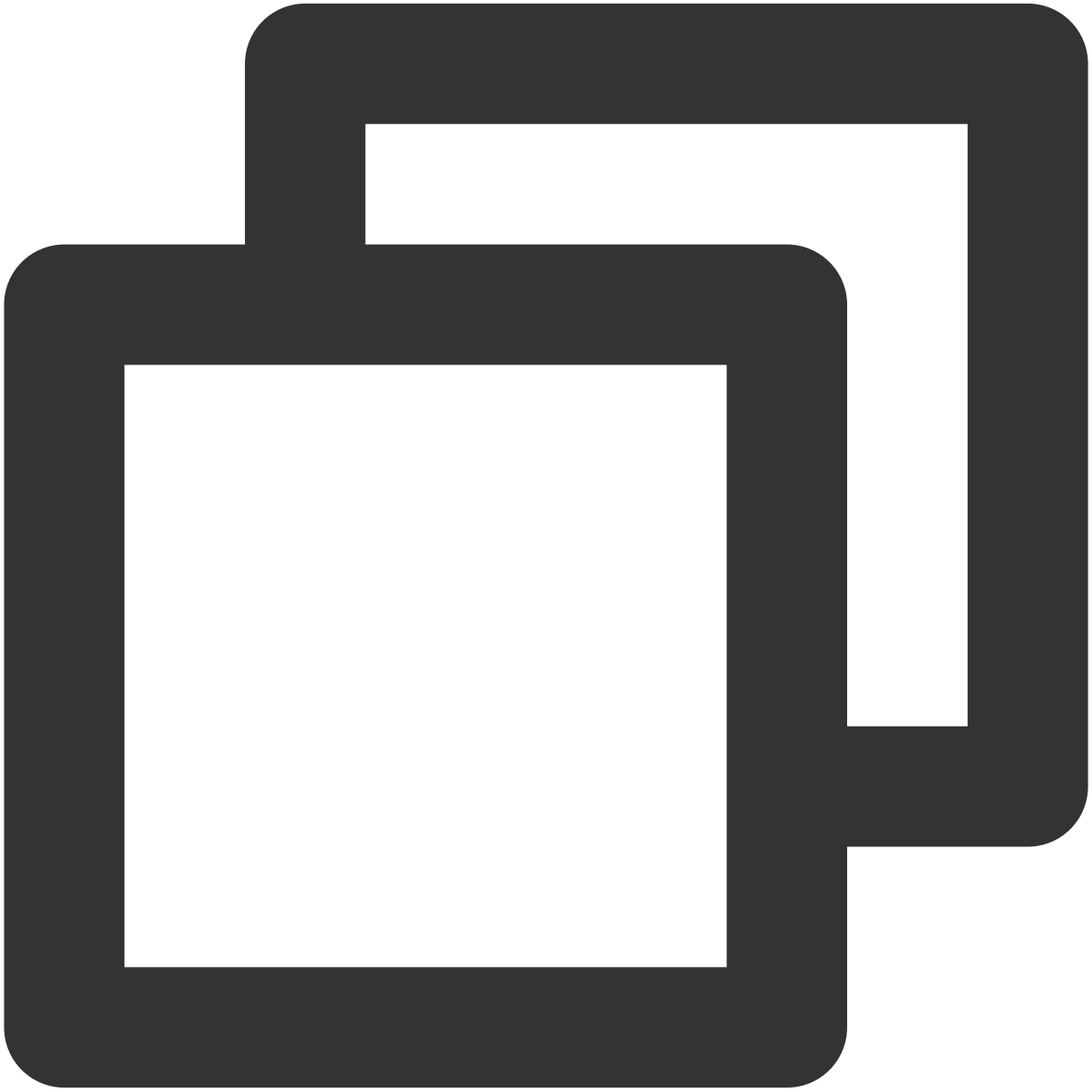
©2013-2022 Tencent Cloud. All rights reserved.

10. Abstract Node

10.1 Using Abstract Nodes in Components

For a node in a custom component template, its corresponding custom component is not determined by the component itself, but the custom component caller. In this case, you can declare that this node is an "abstract node". For example, we can implement a selectable-group component by adding custom-radio or custom-checkbox to it. The wxml of this component is written as follows:

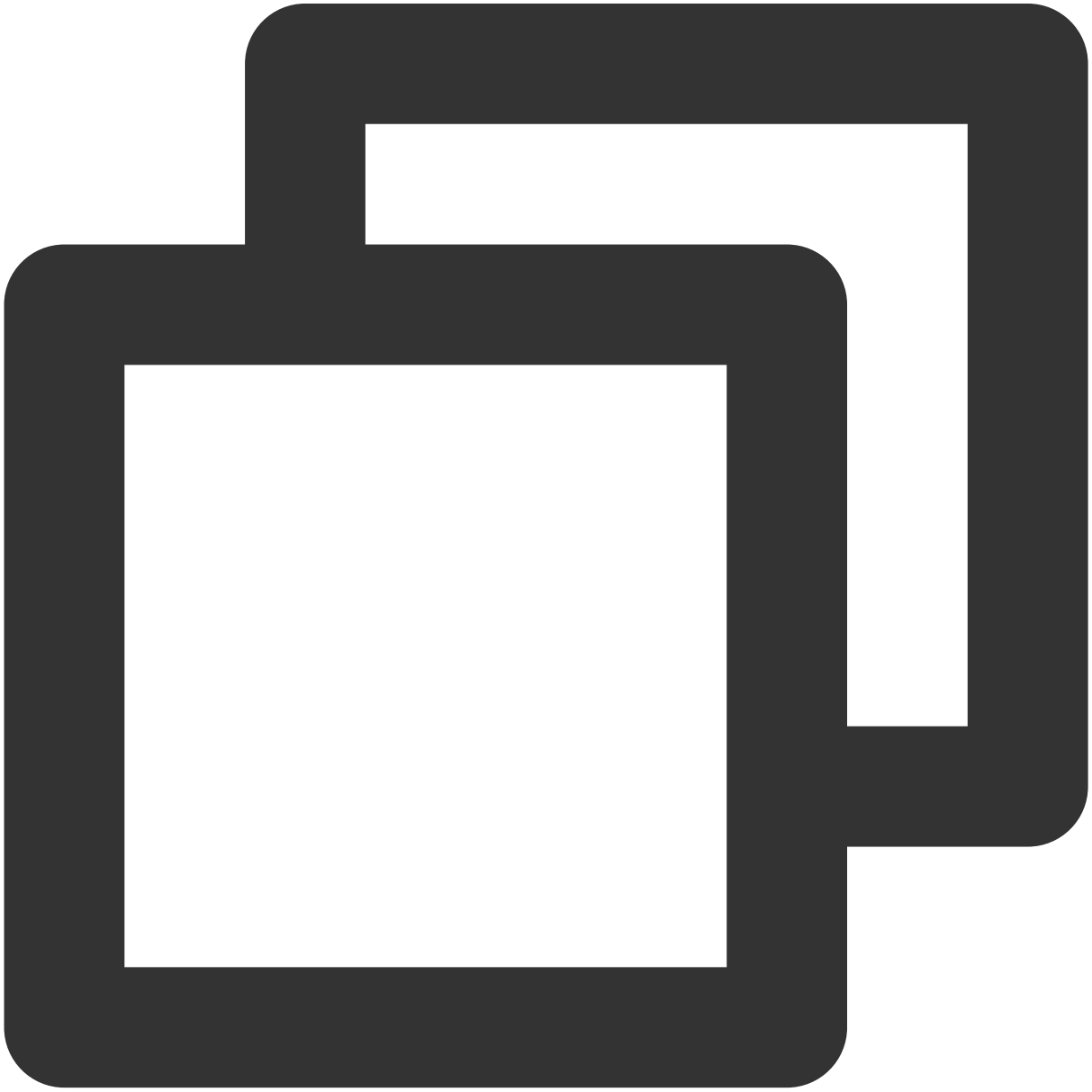
Code example:



©2013-2022 Tencent Cloud. All rights reserved.

```
<!-- selectable-group.wxml -->
<view wx:for="{{labels}}">
  <label>
    <selectable disabled="{{false}}"></selectable>
    {{item}}
  </label>
</view>
```

Where, "selectable" is not any component declared in the usingComponents field of the json file, but an abstract node. It must be declared in the componentGenerics field:

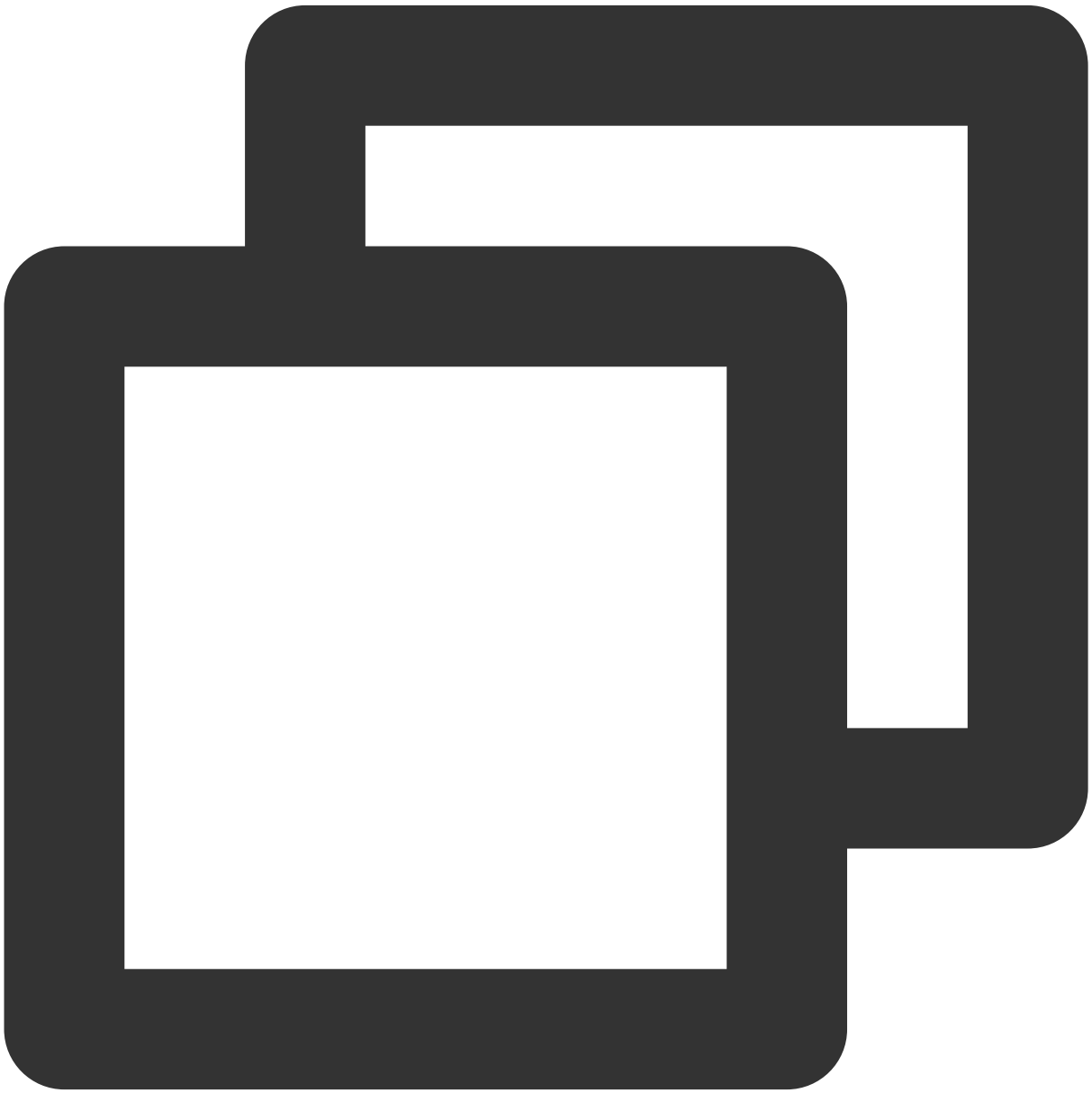


©2013-2022 Tencent Cloud. All rights reserved.

```
{
  "componentGenerics": {
    "selectable": true
  }
}
```

10.2 Using Components with Abstract Nodes

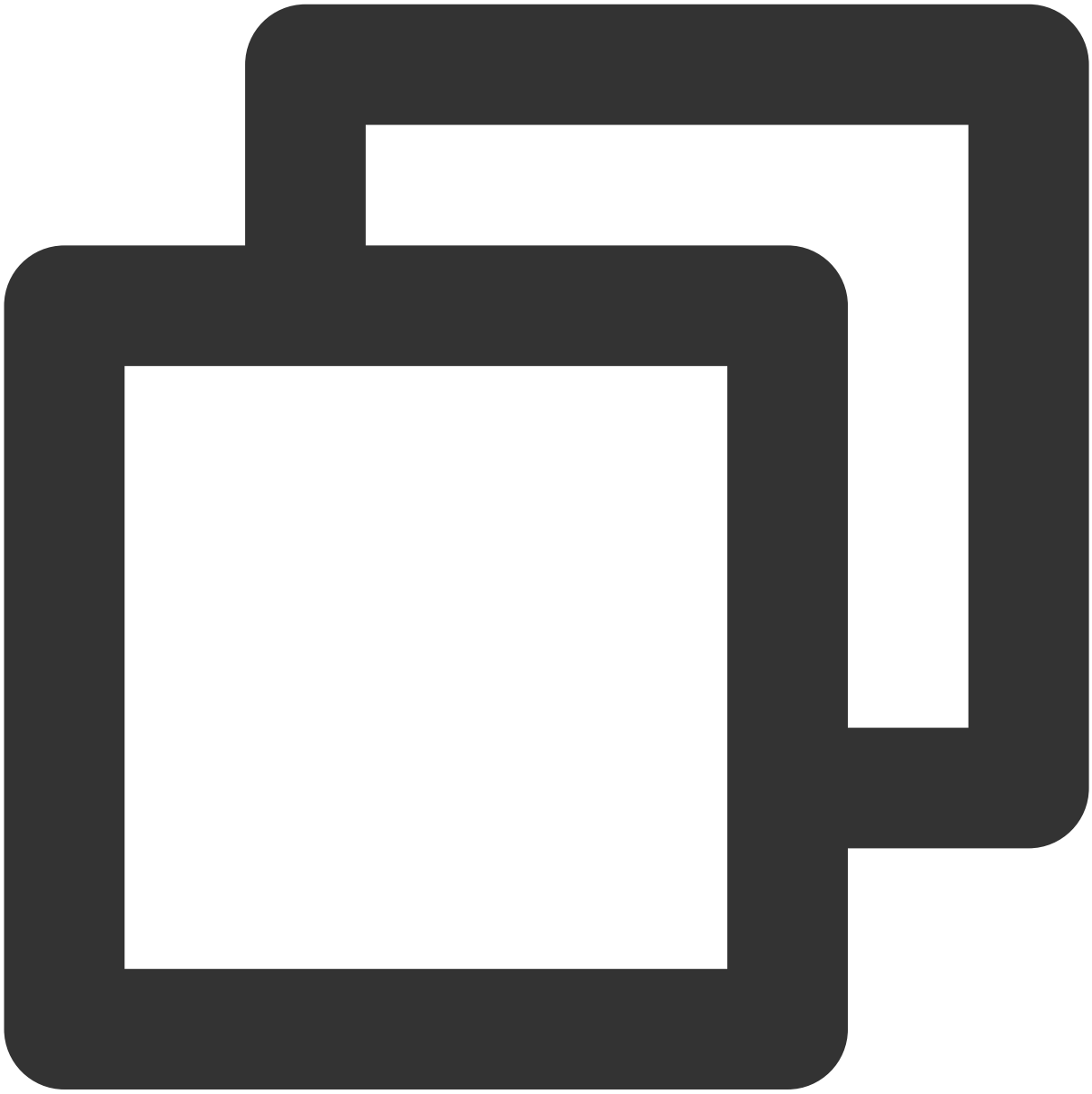
When using the selectable-group component, you must specify the specific component as "selectable":



©2013-2022 Tencent Cloud. All rights reserved.

```
<selectable-group generic:selectable="custom-radio" />
```

As such, when a selectable-group component instance is generated, a "custom-radio" component instance is generated for the "selectable" node. Likewise, if you do the following:



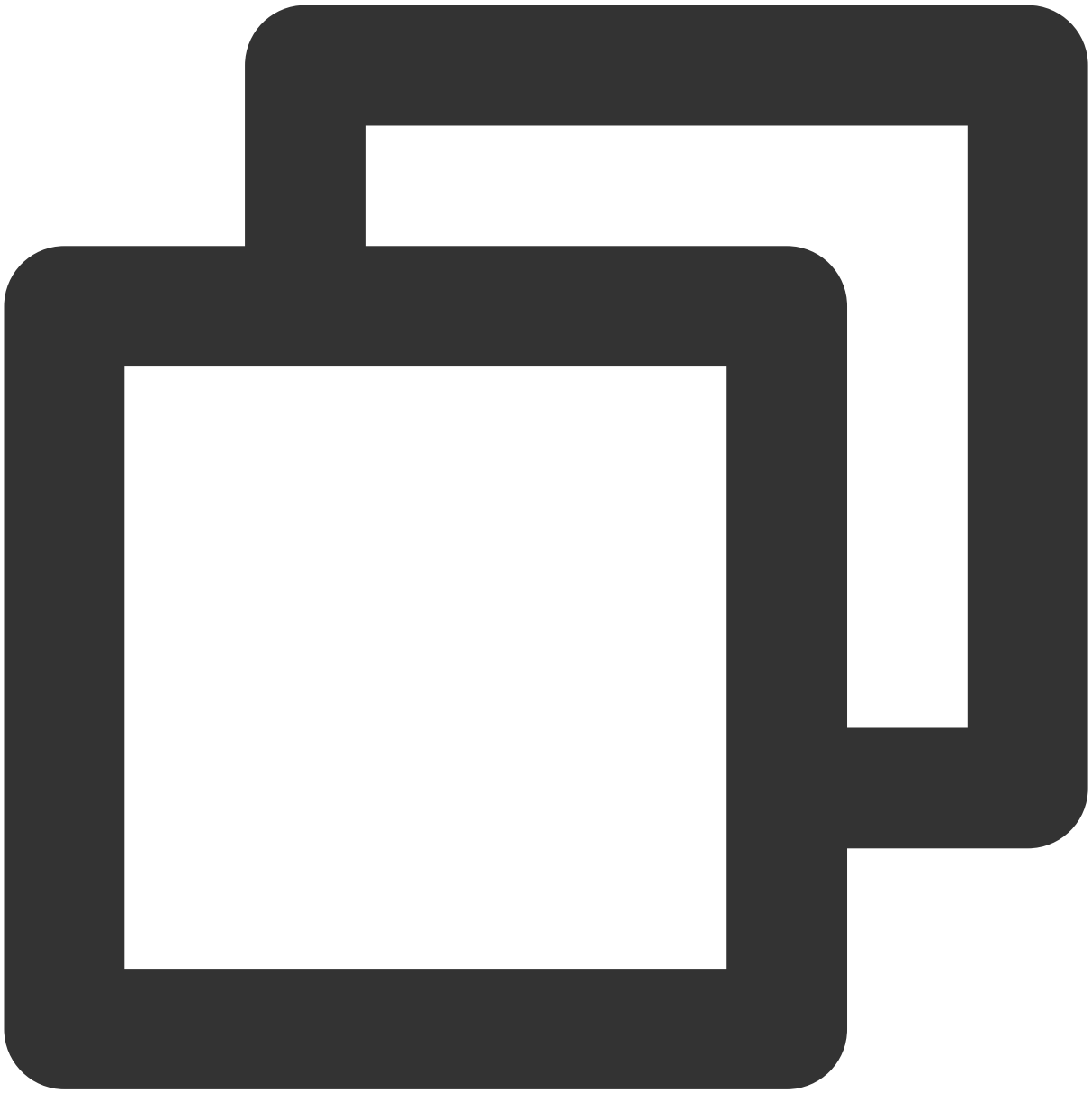
```
<selectable-group generic:selectable="custom-checkbox" />
```

A "custom-checkbox" component instance is generated for the "selectable" node.

Note：

©2013-2022 Tencent Cloud. All rights reserved.

The above-mentioned custom-radio and custom-checkbox must be included in the usingComponents definition field of the json file corresponding to this wxml.

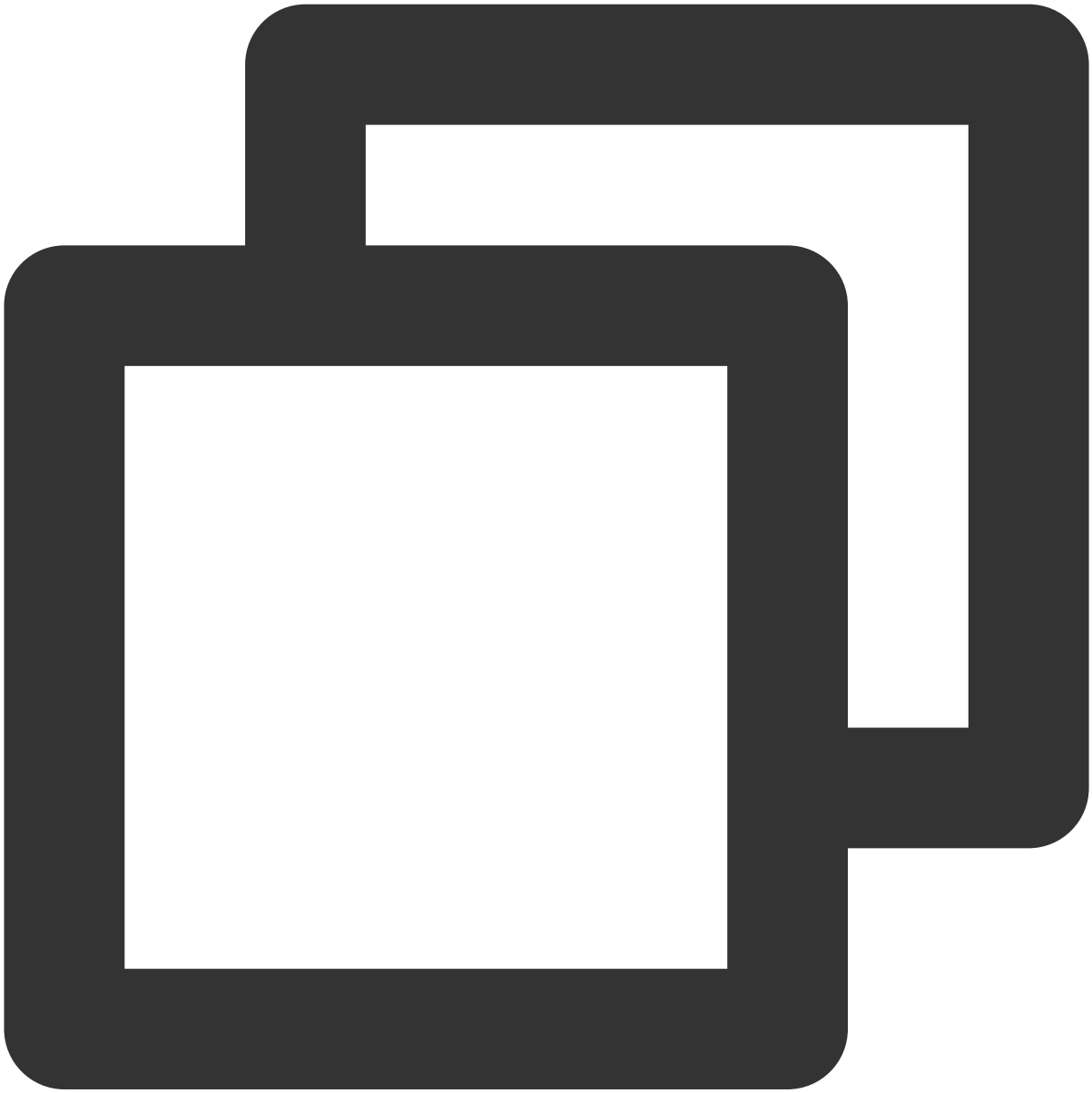


```
{
  "usingComponents": {
    "custom-radio": "path/to/custom/radio",
    "custom-checkbox": "path/to/custom/checkbox"
  }
}
```

©2013-2022 Tencent Cloud. All rights reserved.

10.3 Default Components of Abstract Nodes

You can specify a default component for an abstract node. When no specific component is specified, an instance of the default component is created. The default component can be specified in the componentGenerics field:



```
{
  "componentGenerics": {
    "selectable": {
      "default": "path/to/default/component"
    }
  }
}
```

©2013-2022 Tencent Cloud. All rights reserved.

10.4 Tips

yyy in generic:xxx="yyy" referenced by the node's generic field must be a static value. Therefore, abstract nodes are not suitable for scenarios where the node name is determined dynamically.

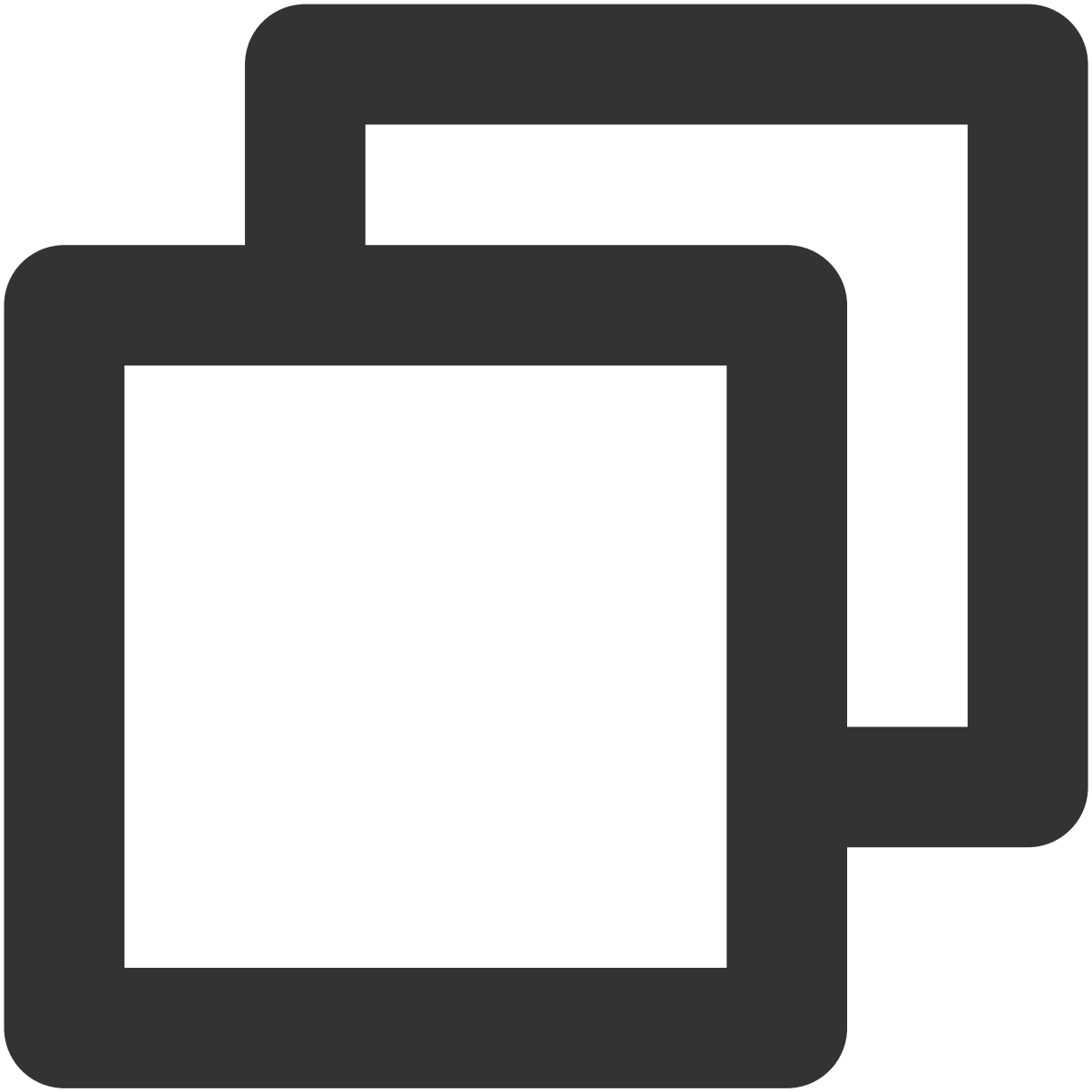
11. Custom Component Extension

To better customize the functionality of custom components, you can use the custom component extension mechanism.

11.1 Effect of Extension

In order to better understand the effect of the extension, look at the example below:

©2013-2022 Tencent Cloud. All rights reserved.



```
// behavior.js
module.exports = Behavior({
  definitionFilter(defFields) {
    defFields.data.from = 'behavior'
  },
})

// component.js
Component({
  data: {
    from: 'component'
```

©2013-2022 Tencent Cloud. All rights reserved.

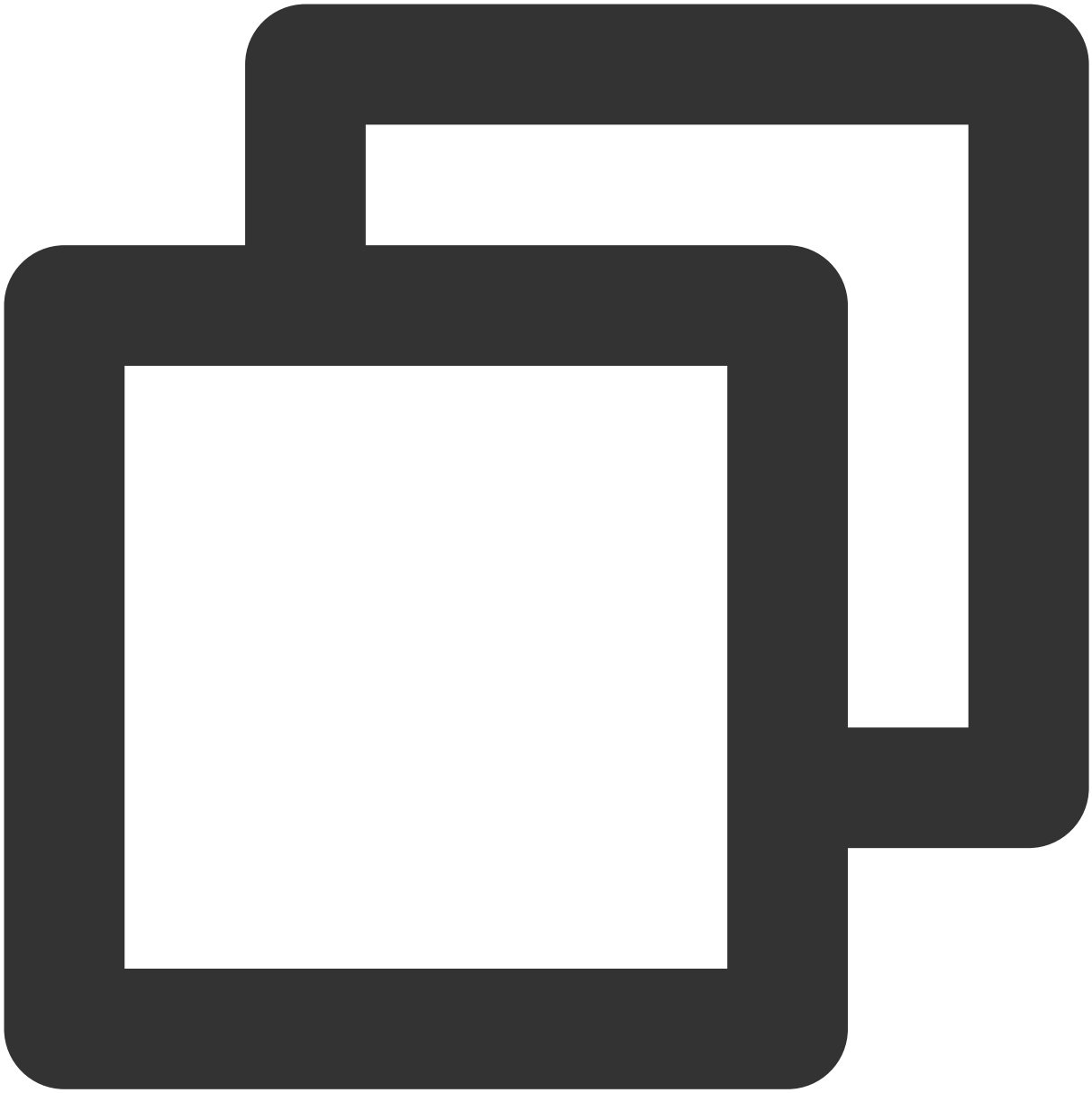
```
  },
  behaviors: [require('behavior.js')],
  ready() {
    console.log(this.data.from) // You can see that the output here is behavior ins
  }
})
```

As you can see from the example, extensions to custom components provide the ability to modify the data definition field in custom component.

11.2 Using Extensions

The Behavior() constructor provides a new definition field, definitionFilter to support the custom component extension. The definitionFilter function is injected with the following two parameters when it is called: the component/behavior definition object that uses this behavior, and the definitionFilter function list of the behavior used by this behavior. Here is an example:

©2013-2022 Tencent Cloud. All rights reserved.



```
// behavior3.js
module.exports = Behavior({
  definitionFilter(defFields, definitionFilterArr) {},
})

// behavior2.js
module.exports = Behavior({
behaviors: [require('behavior3.js')],
definitionFilter(defFields, definitionFilterArr) {
  // definitionFilterArr[0](defFields)
},
```

©2013-2022 Tencent Cloud. All rights reserved.

```
})

// behavior1.js
module.exports = Behavior({
behaviors: [require('behavior2.js')],
definitionFilter(defFields, definitionFilterArr) {},
})

// component.js
Component({
behaviors: [require('behavior1.js')],
})
```

The code above declares 1 custom component and 3 behaviors. Each behavior uses a definitionFilter definition field. Then, the following events occur in the order of the declaration:

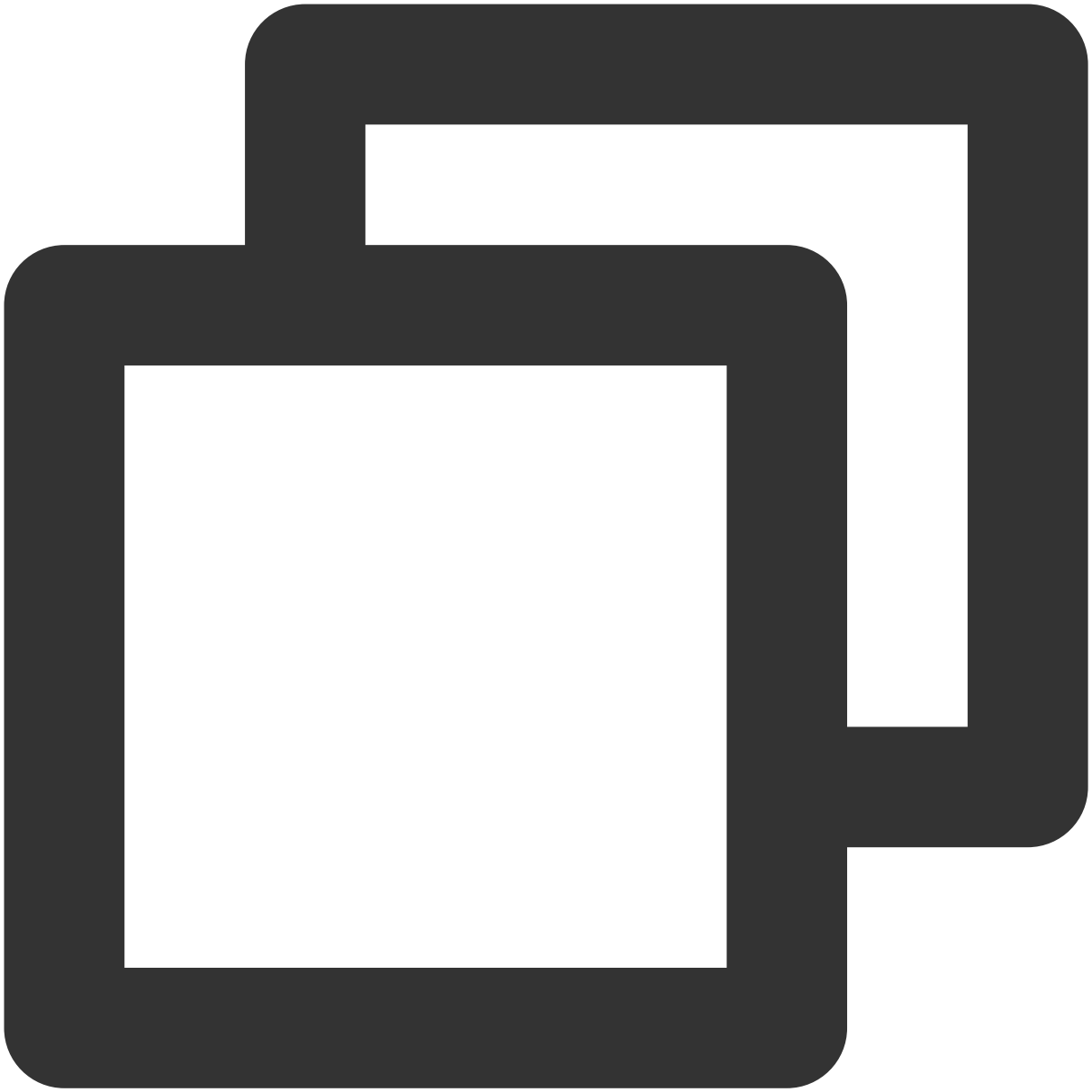
1. When the behavior2 declaration is made, the definitionFilter function of behavior3 is called. The defFields parameter is the definition field of behavior2, and the definitionFilterArr parameter is an empty array because behavior3 does not use any other behavior.
2. When the behavior1 declaration is made, the definitionFilter function of behavior2 is called. The defFields parameter is the definition field of behavior1, the definitionFilterArr parameter is an array with a length of 1, and definitionFilterArr[0] is the definitionFilter function of behavior3 because behavior3 is used by behavior2. You can decide whether to call the definitionFilter function of behavior3 when the behavior1 declaration is made. If the call is needed, add the code definitionFilterArr[0](defFields) here and the definitionFilterArr parameter will be passed by the base library.
3. Similarly, when component is declared, the definitionFilter function of behavior1 is called.

To put it simply, the definitionFilter function can be interpreted as the fact that when A uses B, the A declaration calls the definitionFilter function of B and passes the definition object of A for B to filter. If B also uses C and D, it can decide whether to call the definitionFilter function of C and D to filter the defined object of A.

Code example:

The following is a simple implementation of the Calculated Properties feature of the custom component using extensions:

©2013-2022 Tencent Cloud. All rights reserved.



```
// behavior.js
module.exports = Behavior({
  lifetimes: {
    created() {
      this._originalSetData = this.setData // Original setData
      this.setData = this._setData // Packaged setData
    }
  },
  definitionFilter(defFields) {
    const computed = defFields.computed || {}
    const computedKeys = Object.keys(computed)
```

©2013-2022 Tencent Cloud. All rights reserved.

```
    const computedCache = {}

    // Calculate computed
    const calcComputed = (scope, insertToData) => {
      const needUpdate = {}
      const data = defFields.data = defFields.data || {}

      for (let key of computedKeys) {
        const value = computed[key].call(scope) // Calculate new value
        if (computedCache[key] !== value) needUpdate[key] = computedCache[key] = va
        if (insertToData) data[key] = needUpdate[key] // Insert the value directly
      }

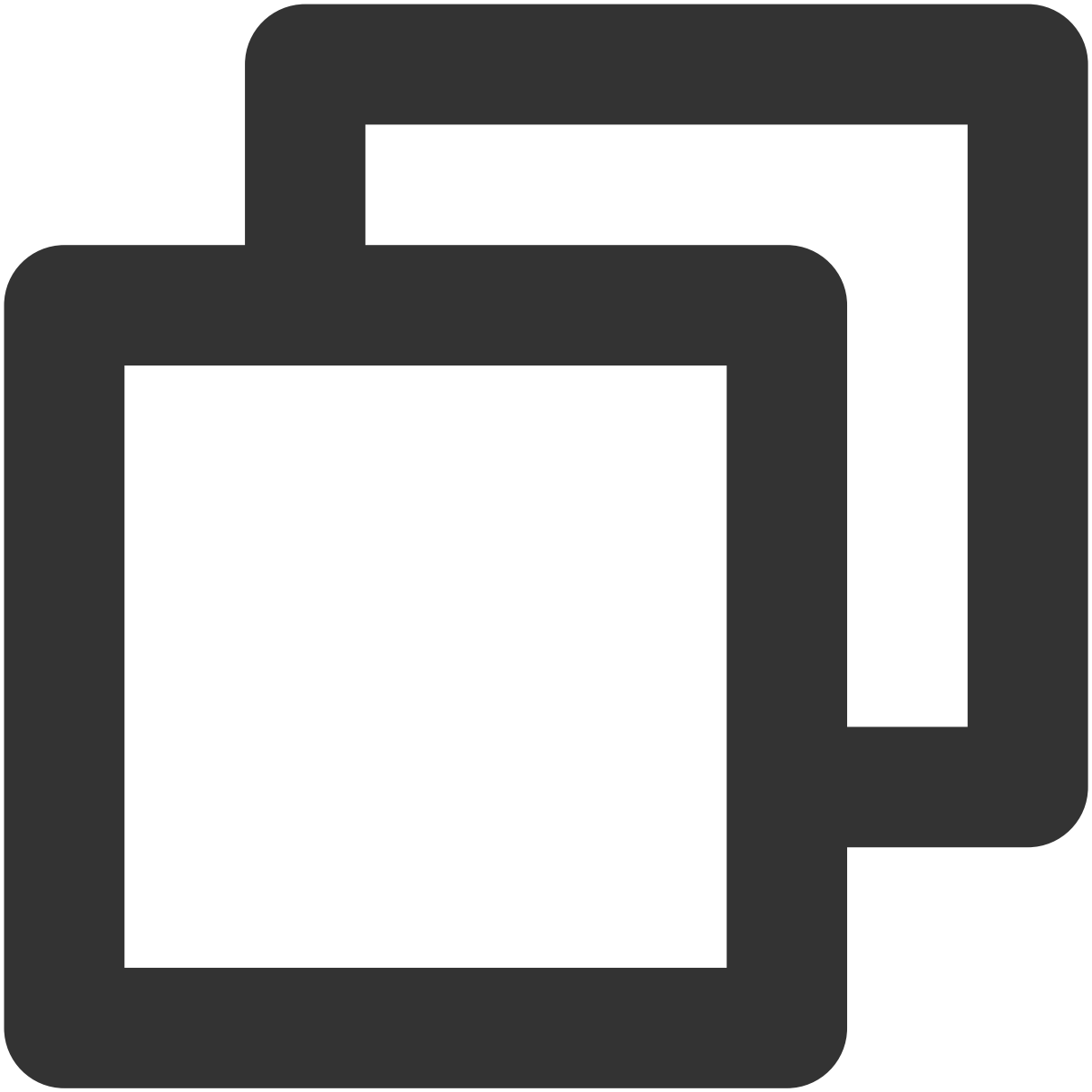
      return needUpdate
    }

    // Override the setData method
    defFields.methods = defFields.methods || {}
    defFields.methods._setData = function (data, callback) {
      const originalSetData = this._originalSetData // Original setData
      originalSetData.call(this, data, callback) // Perform setData for data
      const needUpdate = calcComputed(this) // Calculate computed
      originalSetData.call(this, needUpdate) // Perform setData for computed
    }

    // Initialize computed
    calcComputed(defFields, true) // Calculate computed
  }
})
```

Used in components:

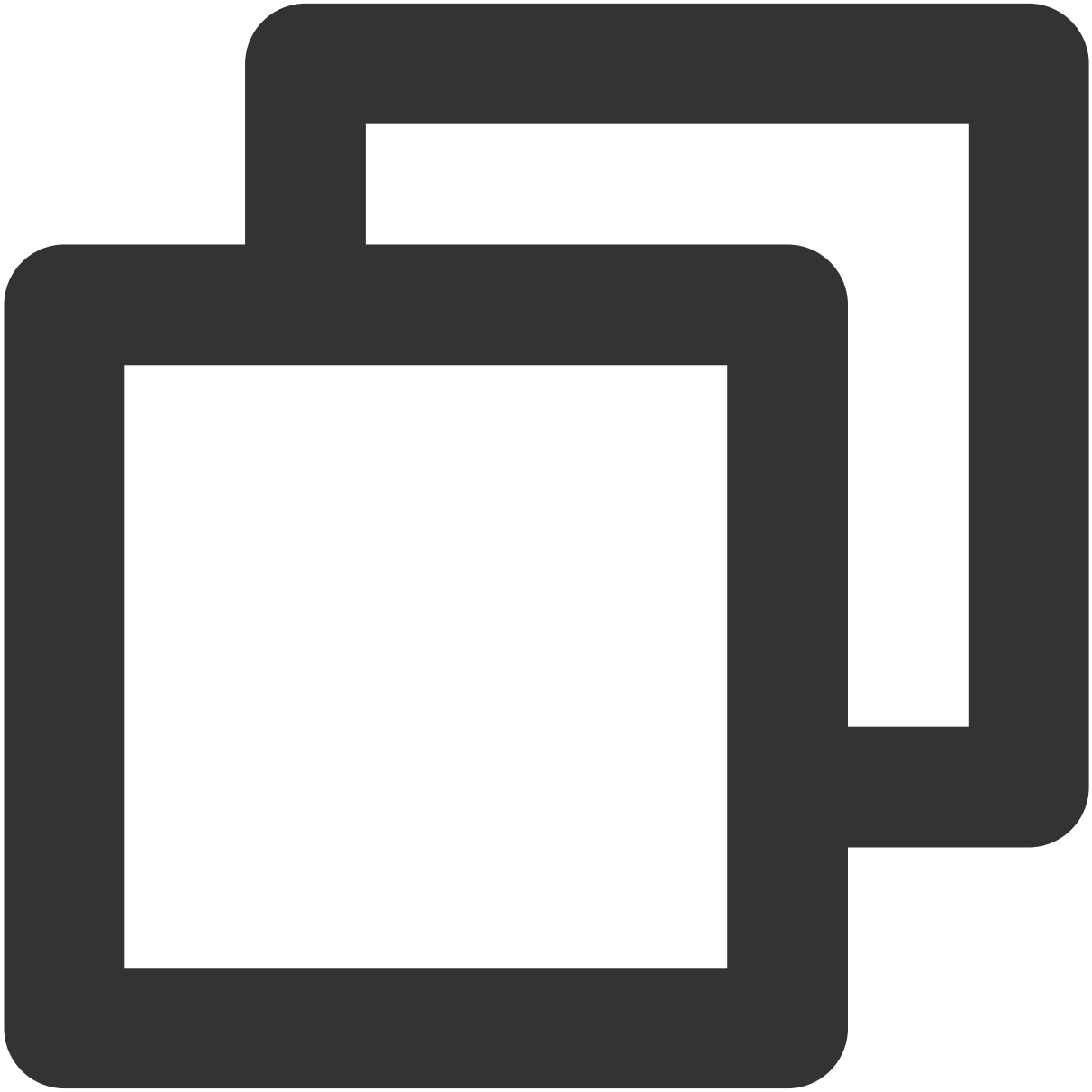
©2013-2022 Tencent Cloud. All rights reserved.



```
const beh = require('./behavior.js')
Component({
  behaviors: [beh],
  data: {
    a: 0,
  },
  computed: {
    b() {
      return this.data.a + 100
    },
  },
```

©2013-2022 Tencent Cloud. All rights reserved.

```
  methods: {
    onTap() {
      this.setData({
        a: ++this.data.a,
      })
    }
  }
})
```



```
<view>data: {{a}}</view>
<view>computed: {{b}}</view>
```

©2013-2022 Tencent Cloud. All rights reserved.

```
<button bindtap="onTap">click</button>
```

The implementation principle is simple: Perform a secondary encapsulation for an existing setData, calculate the value of each field in computed each time you perform setData, and then set it into the data to implement computed property.

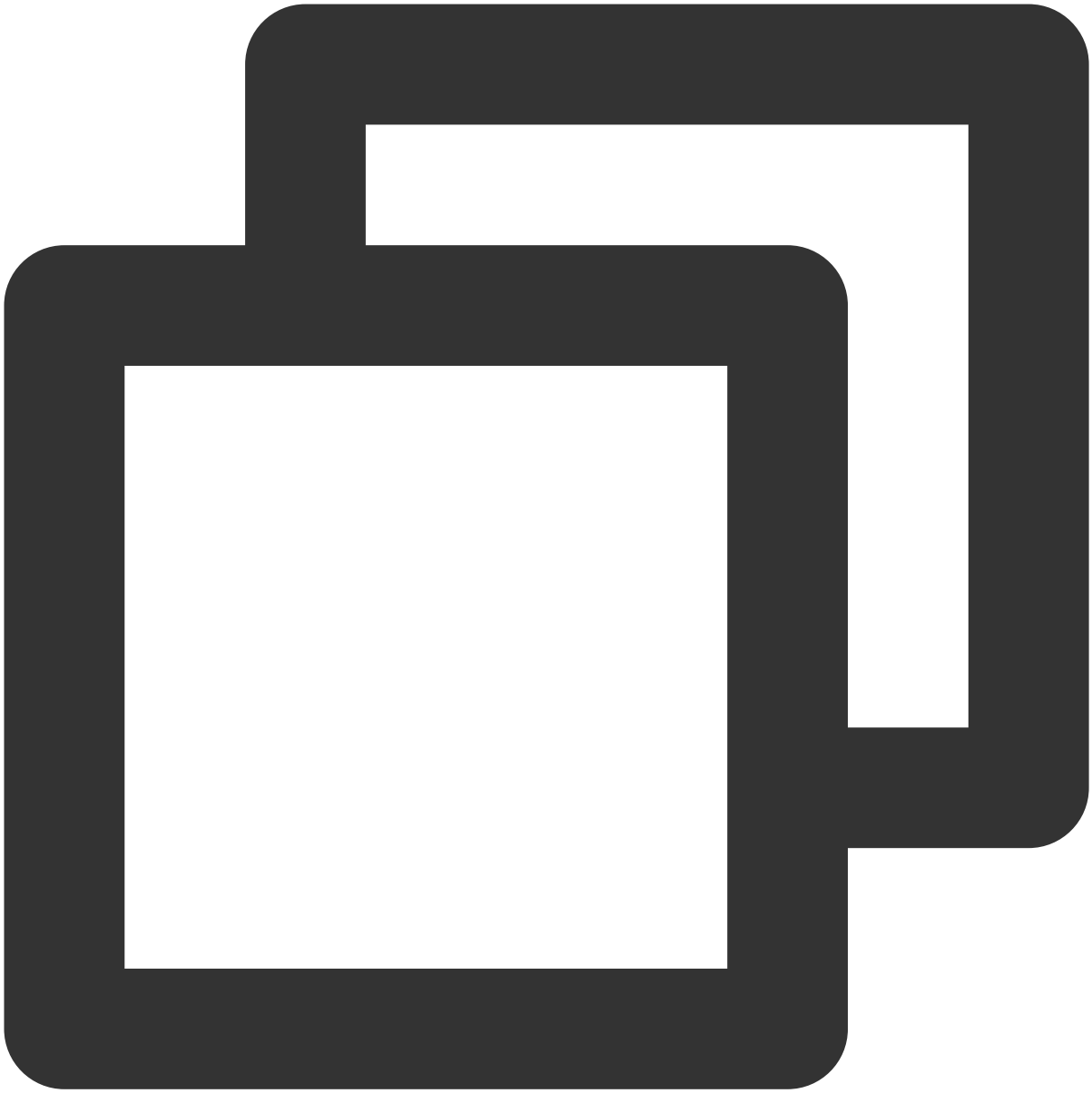
12. Placeholder

When using features such as the Packet Asynchronisation feature, other custom components referenced by the custom component may be unavailable at the start of the rendering process. In this case, in order for the rendering process not to be blocked, the unavailable custom component needs a Component placeholder. The base library renders the unavailable component with the placeholder, and then replaces the placeholder with the component when it becomes available.
The placeholder for a custom component can be another custom component, or a built-in component.

12.1 configure

The componentPlaceholder field in the JSON configuration corresponding to the page or custom component is used to specify the placeholder component, for example:

©2013-2022 Tencent Cloud. All rights reserved.



```
{
  "usingComponents": {
    "comp-a": "../comp/compA",
    "comp-b": "../comp/compB",
    "comp-c": "../comp/compC"
  },
  "componentPlaceholder": {
    "comp-a": "view",
    "comp-b": "comp-c"
  }
}
```

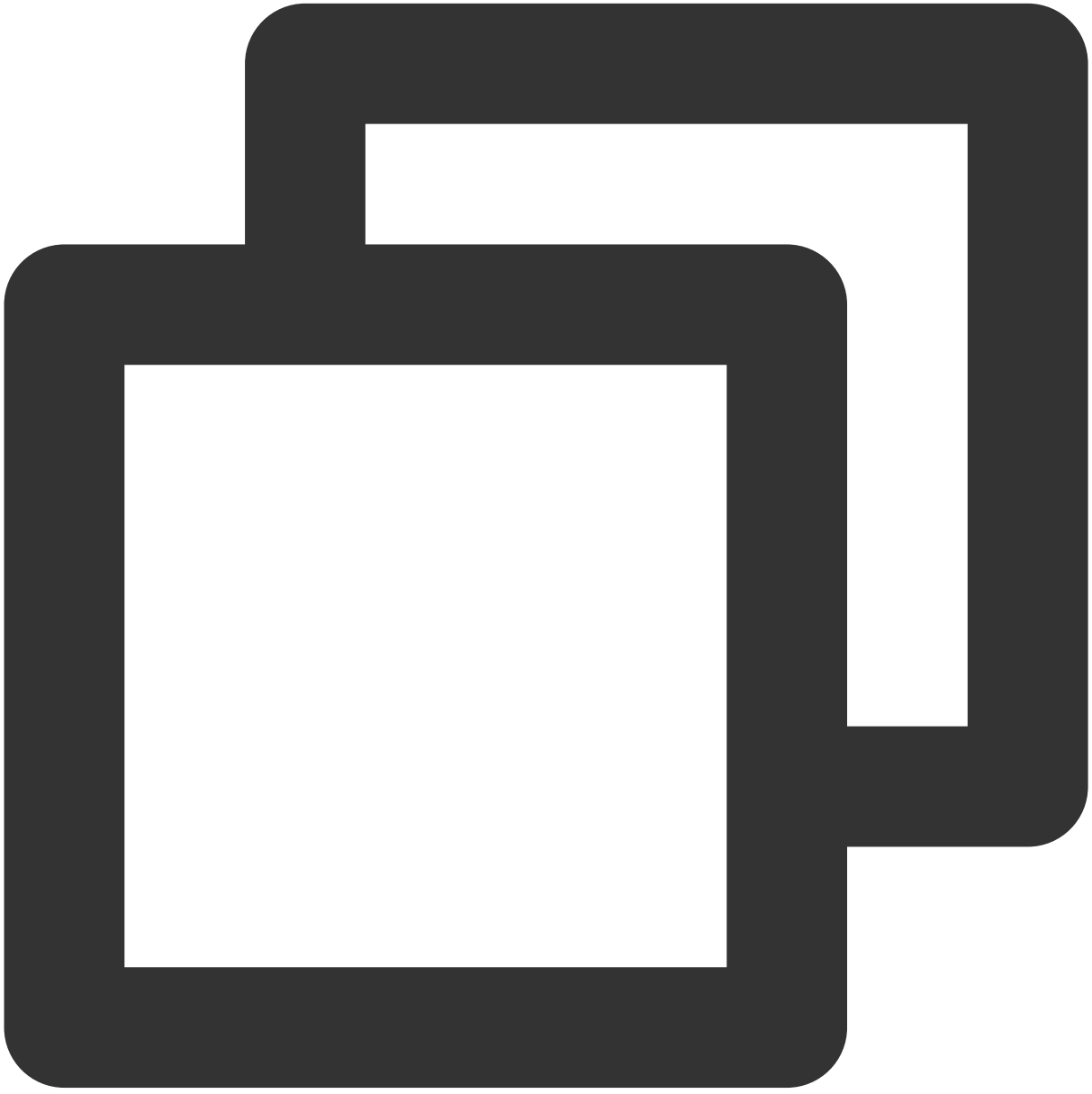
©2013-2022 Tencent Cloud. All rights reserved.

This configuration indicates:

the placeholder component for component comp-a is the built-in component view;

The placeholder component for component comp-b is the custom component comp-c (the path to which is configured in usingComponents).

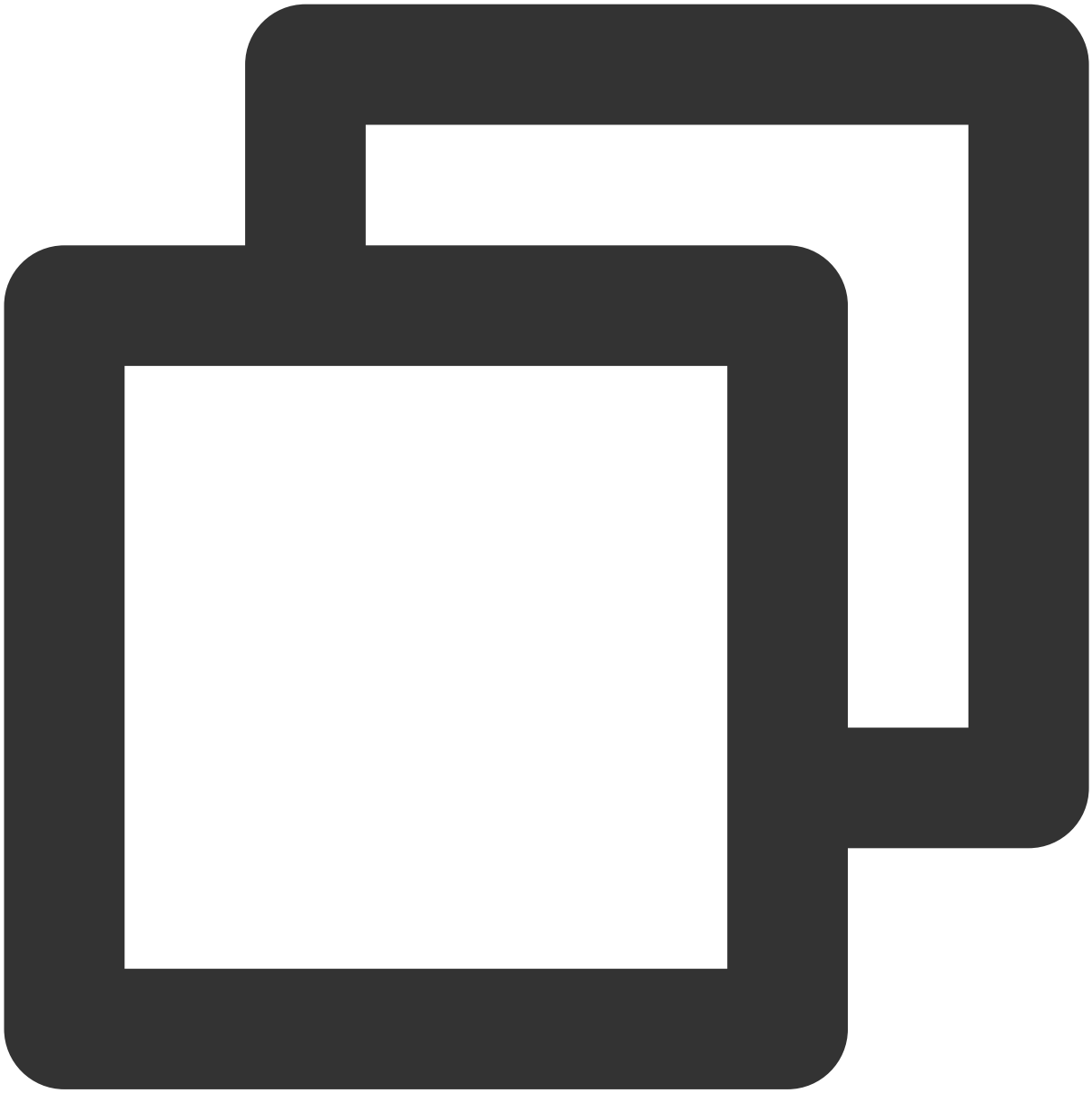
Assume that the template corresponding to this configuration is as follows:



```
<button ontap="onTap">display component</button>
<comp-a wx-if="{{ visible }}">
  <comp-b prop="{{ p }}">text in slot</comp-b>
</comp-a>
```

©2013-2022 Tencent Cloud. All rights reserved.

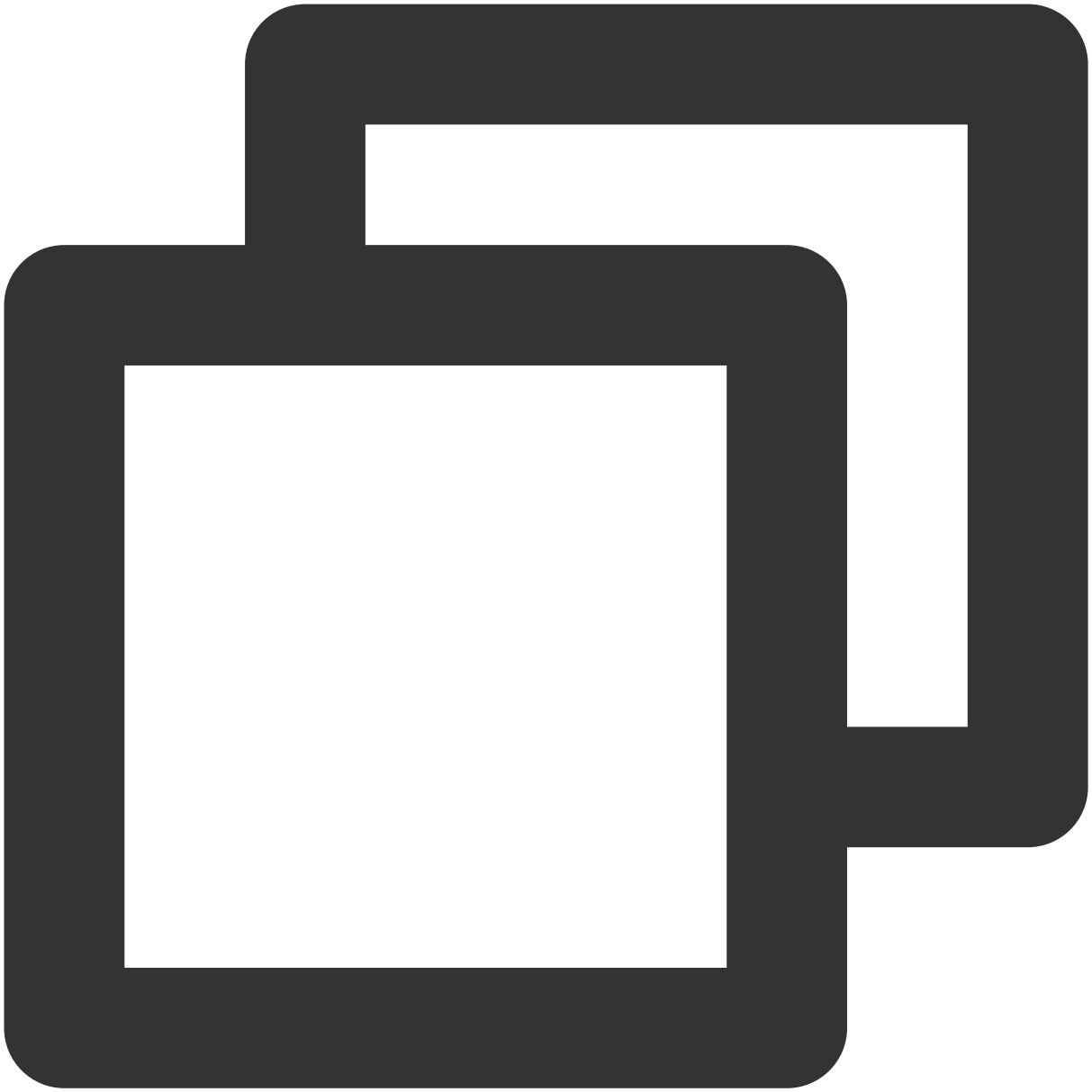
If visible is false when the mini program starts, only the button will be rendered; when the button is clicked, this.setData({ visible: true }) is executed, and if comp-a, comp-b are not available, the page will be rendered as:



```
<button>display component</button>
<view>
  <comp-c prop="{{ p }}">text in slot</comp-c>
</view>
```

When comp-a and comp-b are ready, the page is replaced:

©2013-2022 Tencent Cloud. All rights reserved.



```
<button>display component</button>
<comp-a>
  <comp-b prop="{{ p }}">text in slot</comp-b>
</comp-a>
```

12.2 Cautions

1. When a component is specified as a placeholder component (such as comp-c in the above example), it is not valid to specify a placeholder component for it. It is understood that if a component needs to be used as a placeholder component for other components, it must be available in the first place;

©2013-2022 Tencent Cloud. All rights reserved.

2. Currently custom components unavailable include:
2.1 In the case of using the subpackage asynchronisation feature, a component from another subpackage is referenced and the corresponding subpackage has not yet been downloaded;
2.2 the component has not yet been injected in the case of the use of the inject-when-injected feature;
3. If a component is not available and its placeholder component does not exist, an error will be reported and thrown when rendering;
4. if a component does not exist, but an available placeholder component is specified for it, the placeholder component can be rendered normally, but subsequent attempts to prepare a replacement will report an error and throw.

Attachment: Rendering flow with placeholder components involved

When the base library tries to render a component, it first recursively checks usingComponents to collect information about all the components it will use; in this process, if a component being used is not available, the base library will first check if it has a corresponding placeholder component. If not, the library interrupts the rendering and throws an error; if it does, it flags the unavailable component and replaces it with a placeholder component in the subsequent rendering process. The unavailable component will try to prepare (download a subpackage or inject code, etc.) at the end of the current rendering process; when the preparation process is complete, it will try to render the component again (in effect, performing the same process) and replace the previously rendered placeholder component.

©2013-2022 Tencent Cloud. All rights reserved.

Basic competencies
Network

Last updated：2024-07-12 17:42:57

Note：
When using web-related APIs in Mini Programs/Mini Games, developers should note the following issues in advance.

Configuration of Server Domain Name

Each mini program needs to set a communication domain name in advance, and the mini program can only communicate with the specified domain name. This includes normal HTTPS requests (wx.requestt), uploading files (wx.uploadFile), downloading files (wx.downloadFile) and Websocket communication (wx.connectSocket), UDP communication (wx.createUDPSocket), TCP communication (wx.createTCPSocket).
The network interface only allows communication with non-local IPs and configured server domains on the same LAN.

Configuration process

Configure the server domain name in Mini Program Console > Settings > Development Settings > Server Domain Name. Configuration notes:
The domain name supports the https (wx.request, wx.uploadFile,wx.downloadFile) protocol, the wss (wx.connectSocket) protocol, and the tcp and udp protocols;
The domain name cannot use an IP address or localhost.
Ports can be configured, such as https://miniApp.com:8080, but when configured, requests can only be made to https://miniApp.com:8080. Requests to URLs such as https://miniApp.com, https://miniApp.com:9091, etc. will fail.
If no port is configured. For example, https://miniApp.com, then the request URL cannot contain a port, not even the default port 443. Requests to https://miniApp.com:443 will fail.

Network Request

Timeout

The default timeout and maximum timeout are both 60s;
The timeout can be configured in app.json;
It is also possible to specify a timeout when the interface is called, e.g. wx.request({ timeout: 5000 }) in ms. The timeout configuration for the interface call takes precedence over the configuration in app.json.

Use Limits

The referer header of a network request is optional. Its format is always https://servicewechat.com/{appid}/{version}/page-frame.html, in which {appid} is the Mini Program's appid and {version} is the version number of the Mini Program. A version number of 0 means it is a developer version, trial

©2013-2022 Tencent Cloud. All rights reserved.

version, or review version. A version number of devtools means it is the Weixin DevTools. Those with a version number other than the above ones are formal versions;
The maximum concurrency of wx.request, wx.uploadFile,wx.downloadFile is 10;
The maximum concurrency limit for wx.connectSocket is 5;
When the Mini Program is running in the background, if a network request does not end within 5s, an error message of fail interrupted will be called back. The network request API cannot be called until the Mini Program client is re-opened.

Return value encoding

The values returned by the server should be UTF-8 encoded. The Mini Program will try to convert non-UTF-8 encoded return values but may fail;
The Mini Program automatically filters the BOM header (only one BOM header is filtered).

Callback function

As long the value returned by the server is received, the success callback is triggered regardless of statusCode. Determine the return value based on the business logic.

FAQs

HTTPS certificate

Mini Programs must use HTTPS/WSS to initiate a network request. During the request, the system verifies the HTTPS certificate used by the server domain name. If the verification fails, the request is not successfully initiated. Due to system limitations, different platforms have different levels of stringency for certificates. To ensure the compatibility of Mini Programs, we recommend that developers configure the certificate according to the highest standard and use the relevant tools to check whether the existing certificate meets the requirements.
The requirements for certificates are as follows:
The HTTPS certificate must be valid;
Use a trusted certificate, i.e., the root certificate has been built into the system;
The domain name of the website where the SSL certificate is deployed must be the same as the domain name for which the certificate was issued;
The certificate must not have expired;
The certificate's trust chain must be complete (server configuration is required);
Self-signed certificates is not supported for iOS;
Certificates in iOS must meet Apple's App Transport Security (ATS) requirements;
TLS 1.2 or above is required. However, TLS 1.2 is not supported on some older Android models, so make sure HTTPS server supports TLS 1.2 or below;
Some CAs may not be trusted by the operating system. Take note of the announcements of the Mini Program and other systems when selecting certificates.
Note：

©2013-2022 Tencent Cloud. All rights reserved.

The validity of the certificate can be verified using the openssl s_client -connect example.com:443 command, or you can use other online tools;
In addition to the web request API, if there is any abnormality in other HTTPS requests in the mini program, please check it according to the above process. For example, https images can't be loaded, audio and video can't be played, and so on.

Skip domain name verification

In the Weixin DevTools, the Do not verify the request domain name, TLS version, and HTTPS certificate in development environment option can be enabled to skip verification of the server domain name. In this case, the server domain name will not be verified in the Weixin DevTools or when debugging mode is enabled on your phone.
After the server domain name is configured, we recommend you disable this option during development and test the domain name on each platform to ensure it is correct.
Note：
If you see the phenomenon of "can send request with debugging mode on, can't send request with debugging mode off" on your mobile phone, please make sure that the domain name verification has been skipped, and make sure that the server domain name and certificate configuration are correct.
If you encounter "internal server error", please try the following methods to check yourself ① Change the browser ② Refresh the web page ③ Check whether the file is placed in the correct server root directory. Checking method: Splicing the URL according to the following rules, and try to access whether it is successful or not: https://m.da9c.cn/Filename.

©2013-2022 Tencent Cloud. All rights reserved.

Storage

Last updated：2024-07-12 17:42:28

Each Weixin Mini Program has its own local cache. You can read, write, and clear the local cache via wx.setStorage/wx.setStorageSync, wx.getStorage/wx.getStorageSync, wx.clearStorage/wx.clearStorageSync, and wx.removeStorage/wx.removeStorageSync.

Isolation Policy

The storage limit of a Mini Program is 10 MB for a Weixin user. Storage is isolated based on users. On the same device, user A cannot read the data of user B. A Mini Program's user cannot read or write data of another Mini Program.

Erasion Rules

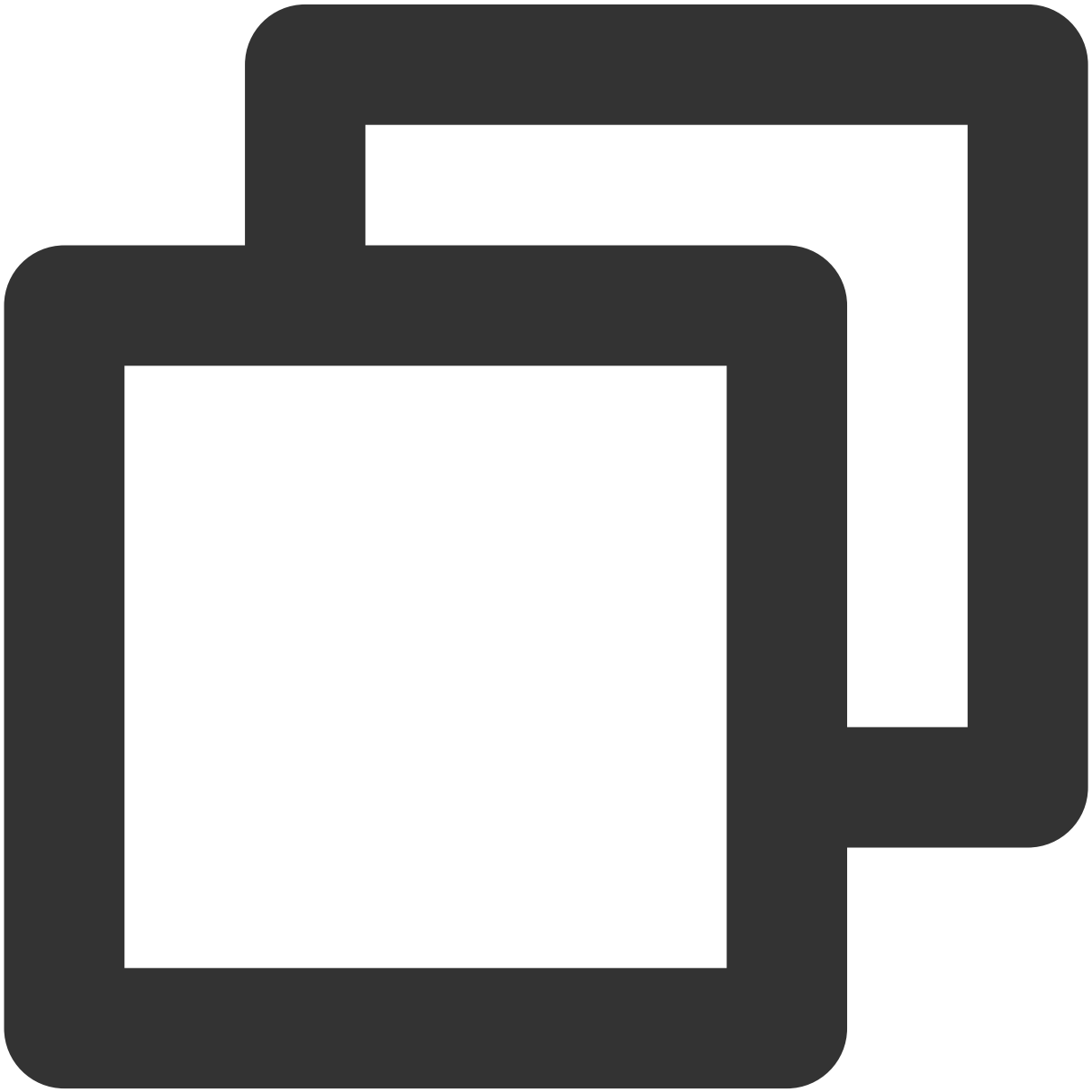
Local cache files are erased only when the code package is erased.

©2013-2022 Tencent Cloud. All rights reserved.

File System

Last updated：2024-07-12 17:41:32

The file system consists of a set of storage areas isolated based on Mini Programs and users and a set of appropriate management APIs. A globally unique file system manager is obtained via wx.getFileSystemManager(). The management operations for all file systems are called via FileSystemManager.



```
var fs = wx.getFileSystemManager()
```

©2013-2022 Tencent Cloud. All rights reserved.

Files are divided into two categories:
Code package file: A file added in the project directory;
Local file: A file generated locally by calling an API, or downloaded over the network and stored locally;
Local files are divided into three types:
Local temporary file: A file that is temporarily generated and will be recycled at any time. The storage space of local temporary files is unlimited;
Local cache file: A file generated by a Mini Program after caching a local temporary file via an API. The directory and file name cannot be customized. Normal Mini Programs can store up to 10 MB of local cache and local user files, while game-type Mini Programs can store up to 50 MB;
Local user file: A file generated by a Mini Program after caching a local temporary file via an API. The directory and file name can be customized. Normal Mini Programs can store up to 10 MB of local user and local cache files, while game-type Mini Programs can store up to 50 MB.

1. Code Package File

Due to limited size, code package files are those needed for the first load. Large files and those that require dynamic replacement should not be added to the code package, but should be downloaded to the local device via the download API after the Mini Game is launched.

1.1 Access code package files

Code package files are accessed by writing the file path from the project root directory, relative path writing is not supported. For example: `/a/b/c` , `a/b/c` are legal, `. /a/b/c` , `. /a/b/c` are not legal.

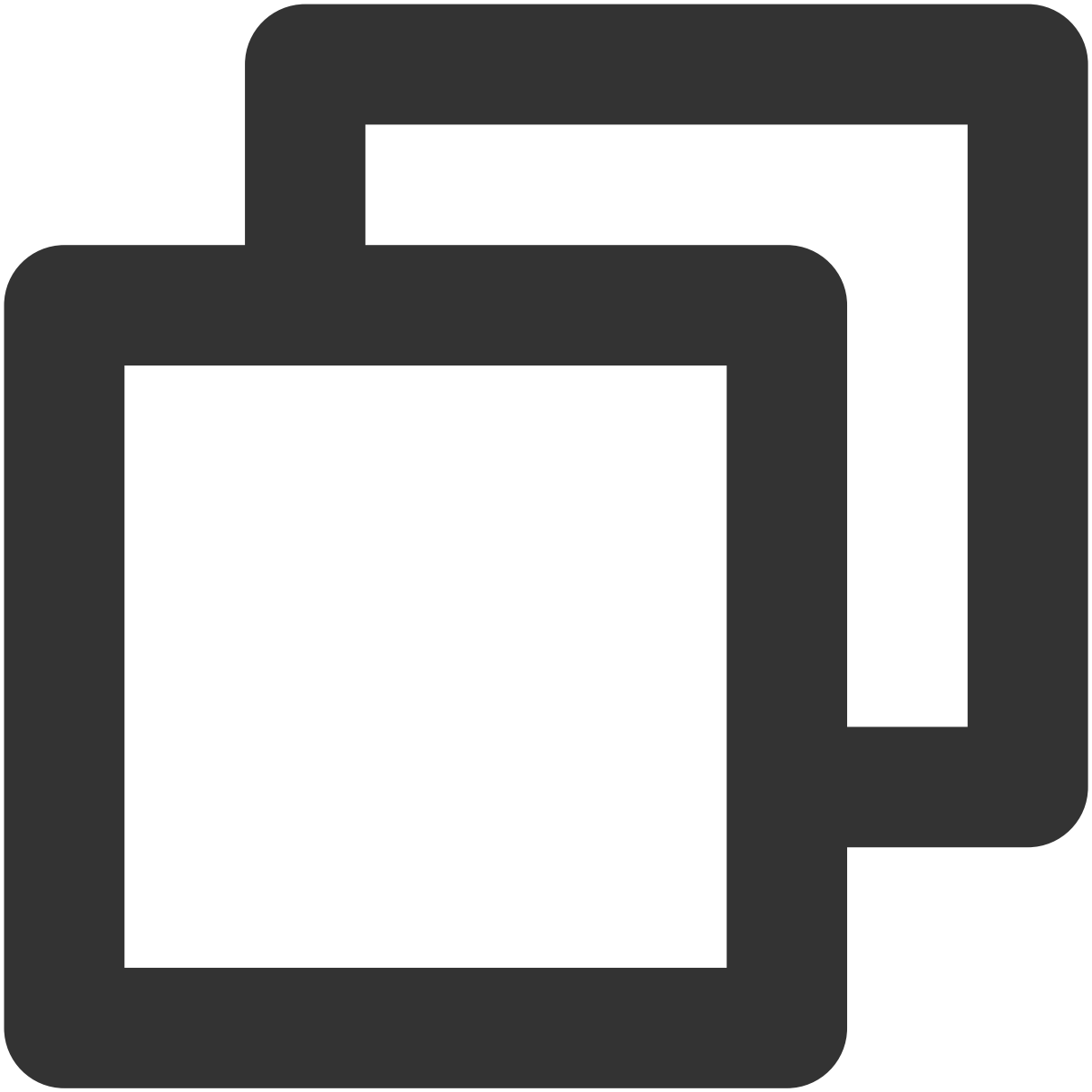
1.2 Modify code package files

Files in the code package cannot be dynamically modified or deleted after running. Modifying the code package files requires a version re-release.

2. Local File

Local file refers to an individual file storage isolated based on users after the Mini Program is added to the mobile device by the user. A Weixin user cannot access the files of other users who have logged in on the same phone, and a user who have multiple appIds cannot access files under one appId using another appId.
The file path of a local file follows this format:

©2013-2022 Tencent Cloud. All rights reserved.



```
{{Protocol Name}}://File Path
```

Note：

The protocol name is "wxfile" in the Weixin app for iOS/Android and "http" in the Weixin DevTools. Developers can ignore this difference, and should not hard code a complete file path in the code.

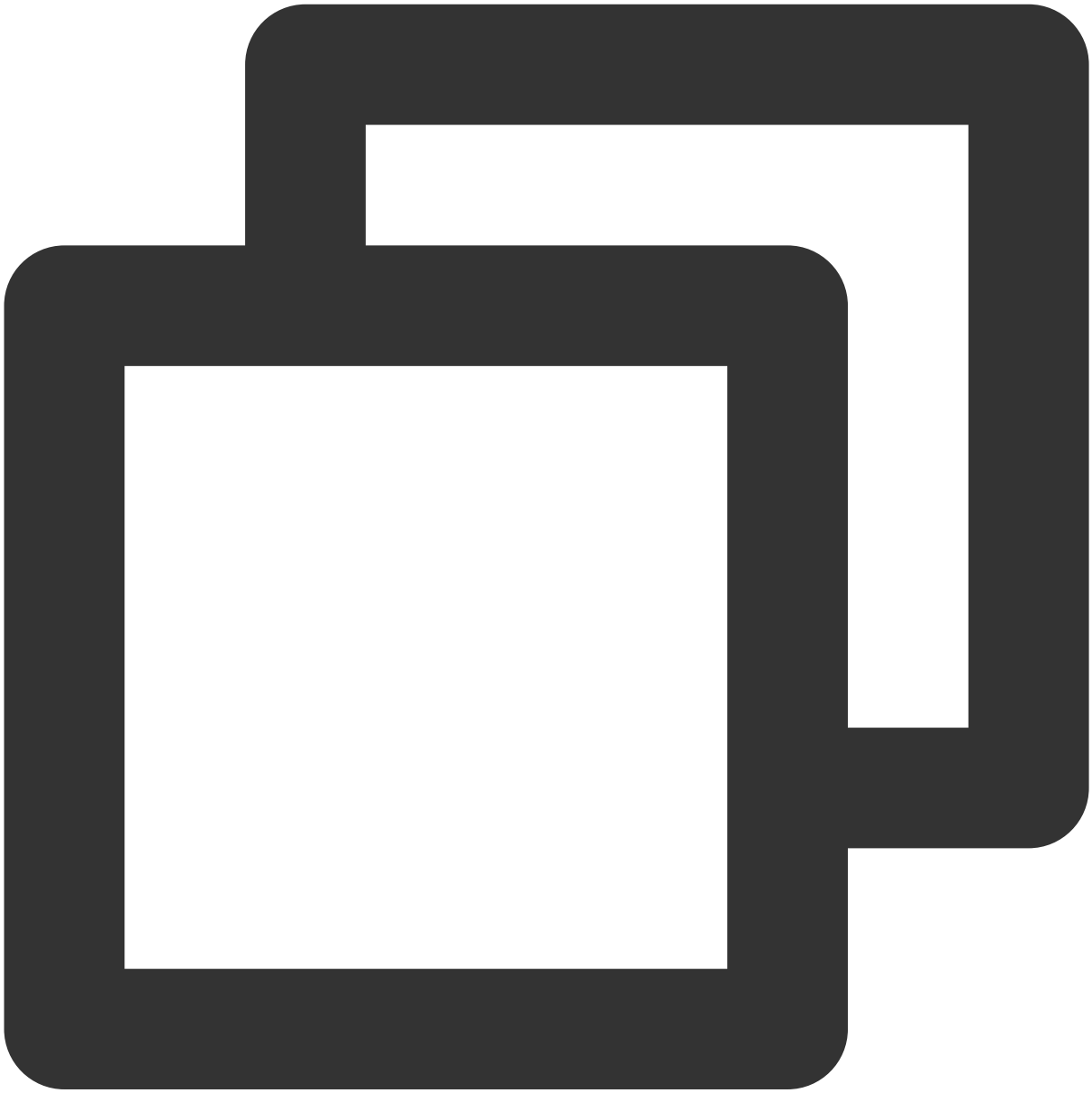
2.1 Local temporary file

Local temporary file can only be generated by calling a specific API. However, you do not have the permission to write to the file directly. After a local temporary file is generated, it is valid only during the Mini Program's current lifecycle

©2013-2022 Tencent Cloud. All rights reserved.

and is not available after the Mini Program restarts. Therefore, you cannot save the local temporary file path for later use. To use a local temporary file later, you can convert it to a local cache file or a local user file via the FileSystemManager.saveFile() or FileSystemManager.copyFile() API.

Code example:



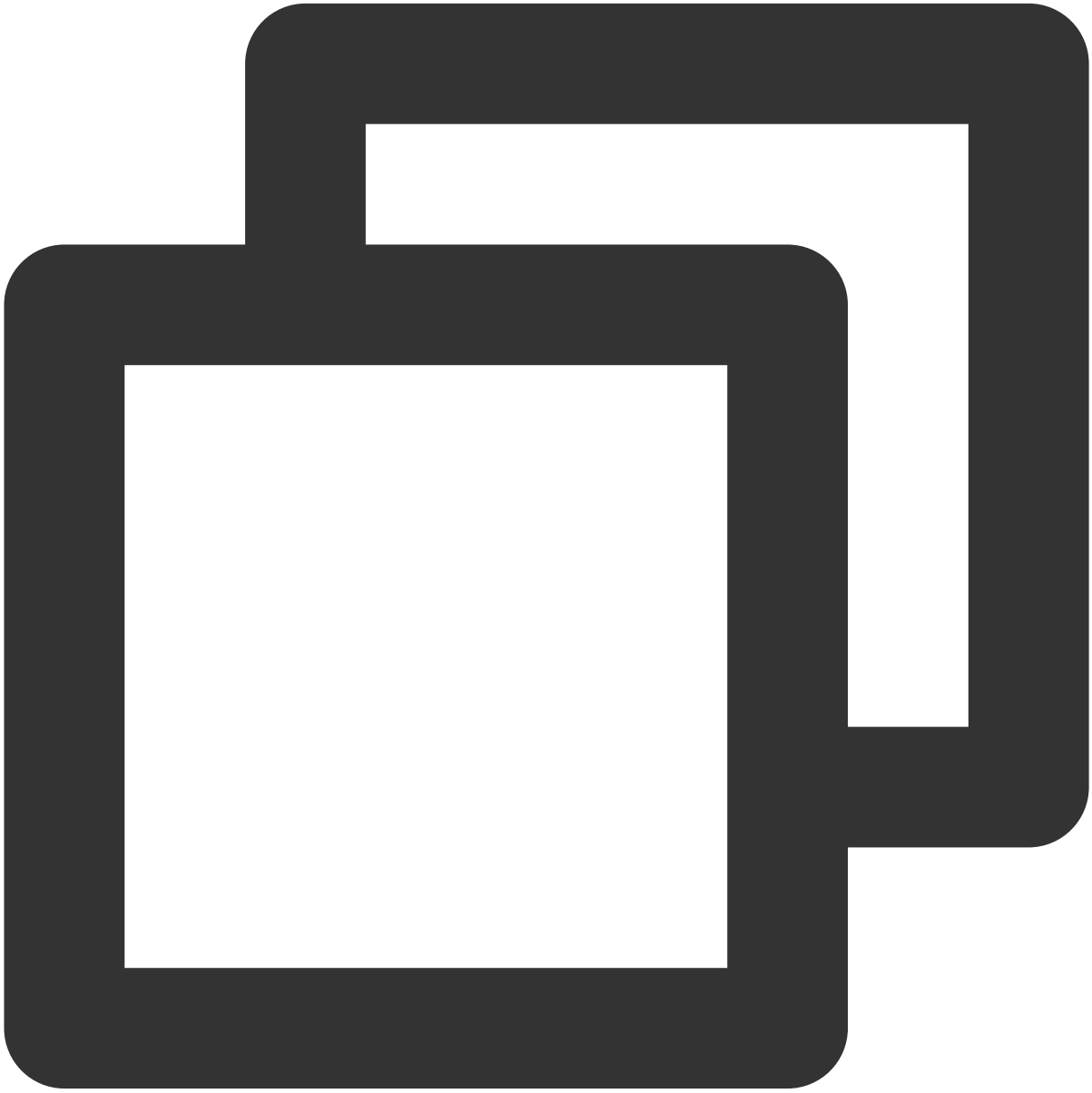
```
wx.chooseImage({
  success: function (res) {
    var tempFilePaths = res.tempFilePaths // Each entry of tempFilePaths is a local
})
```

©2013-2022 Tencent Cloud. All rights reserved.

2.2 Local cache file

Local cache file can only be generated by calling a specific API and cannot be written directly. After a local cache file is generated, it is still available after the Mini Program restarts. Local cache files can only be obtained by saving the local temporary files via the FileSystemManager.saveFile() API.

Code example:



```
fs.saveFile({
  tempFilePath: '', // Pass a local temporary file path
  success(res) {
    console.log(res.savedFilePath) // res.savedFilePath is a local cache file path
```

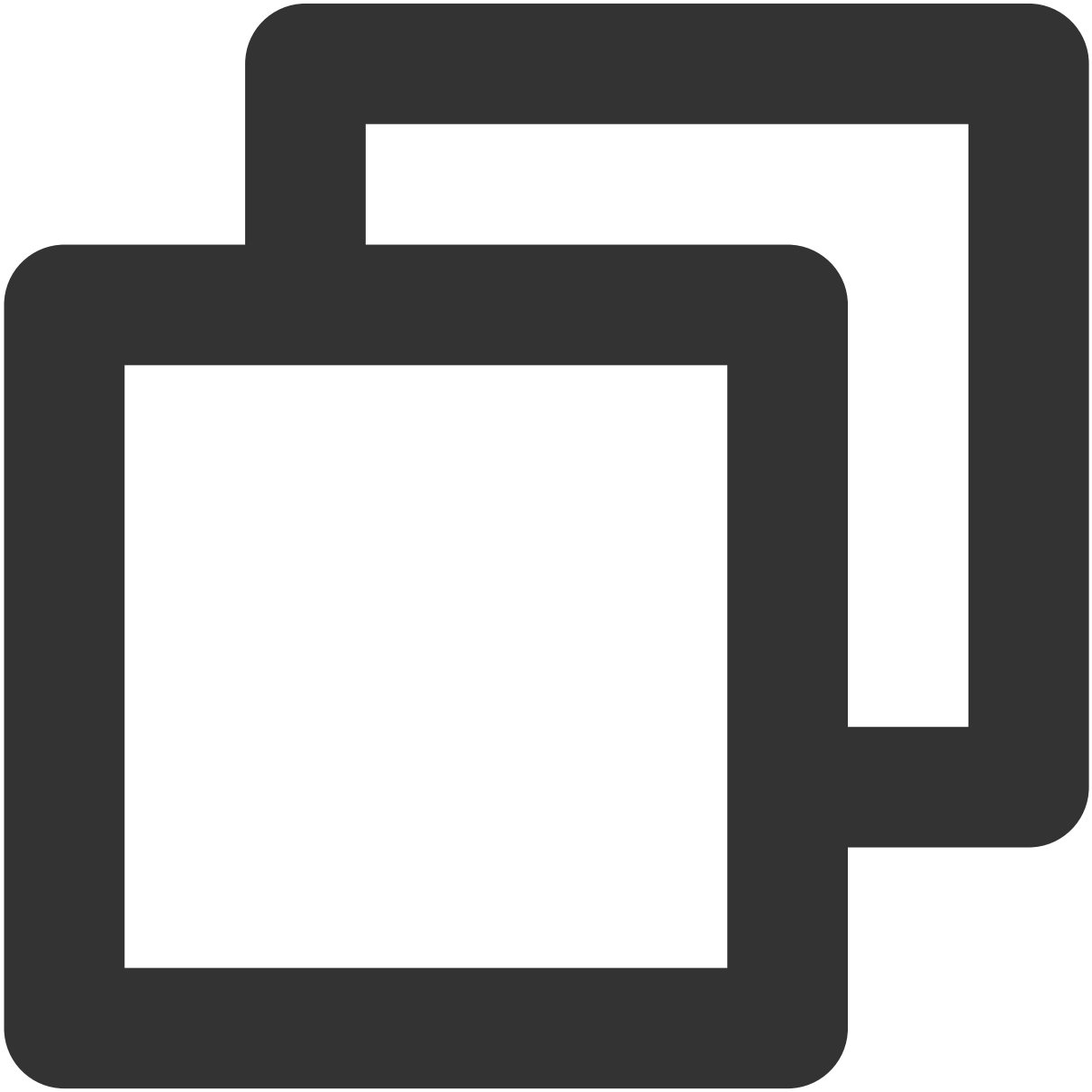
©2013-2022 Tencent Cloud. All rights reserved.

```
    }
})
```

2.3 Local user file

Local user file is a new concept that has been added since version 1.7.0. We provide a user file directory for developers, who have completely free read/write access to this directory. The path to this directory is available via `wx.env.USER_DATA_PATH` .

Code example:



©2013-2022 Tencent Cloud. All rights reserved.

```
// Create a hello.txt file in the local user file directory and write "hello, world
const fs = wx.getFileSystemManager()
fs.writeFileSync(`${wx.env.USER_DATA_PATH}/hello.txt`,  'hello, world', 'utf8')
```

2.4 Read/Write Permission

API/Component	Read	Write
Code package file	Yes	No
Local temporary file	Yes	No
Local cache file	Yes	No
Local user file	Yes	Yes

2.5 Erasion Rules

Local temporary files are only guaranteed to persist during the Mini Program's current lifecycle. Once the Mini Program is closed, these files can be erased and may not be available at the next cold start;
Local cache files and local user files are erased only when the code package is erased.

©2013-2022 Tencent Cloud. All rights reserved.

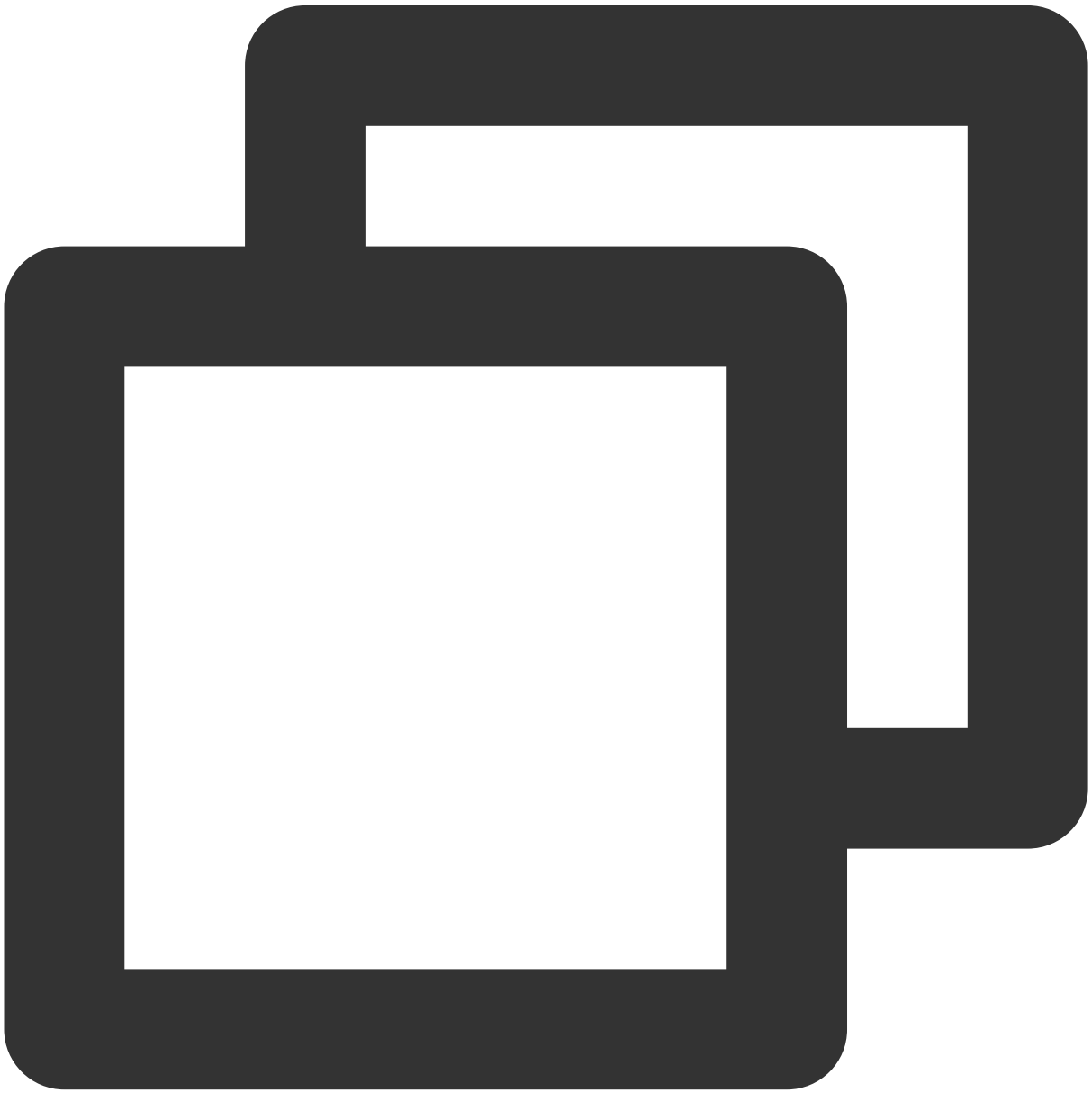
Subpackage Loading

Last updated：2024-07-12 18:33:47

1. Using Subpackage

1.1 Configuration

Assume that the directory structure of a Mini Program supporting subpackages is as follows:



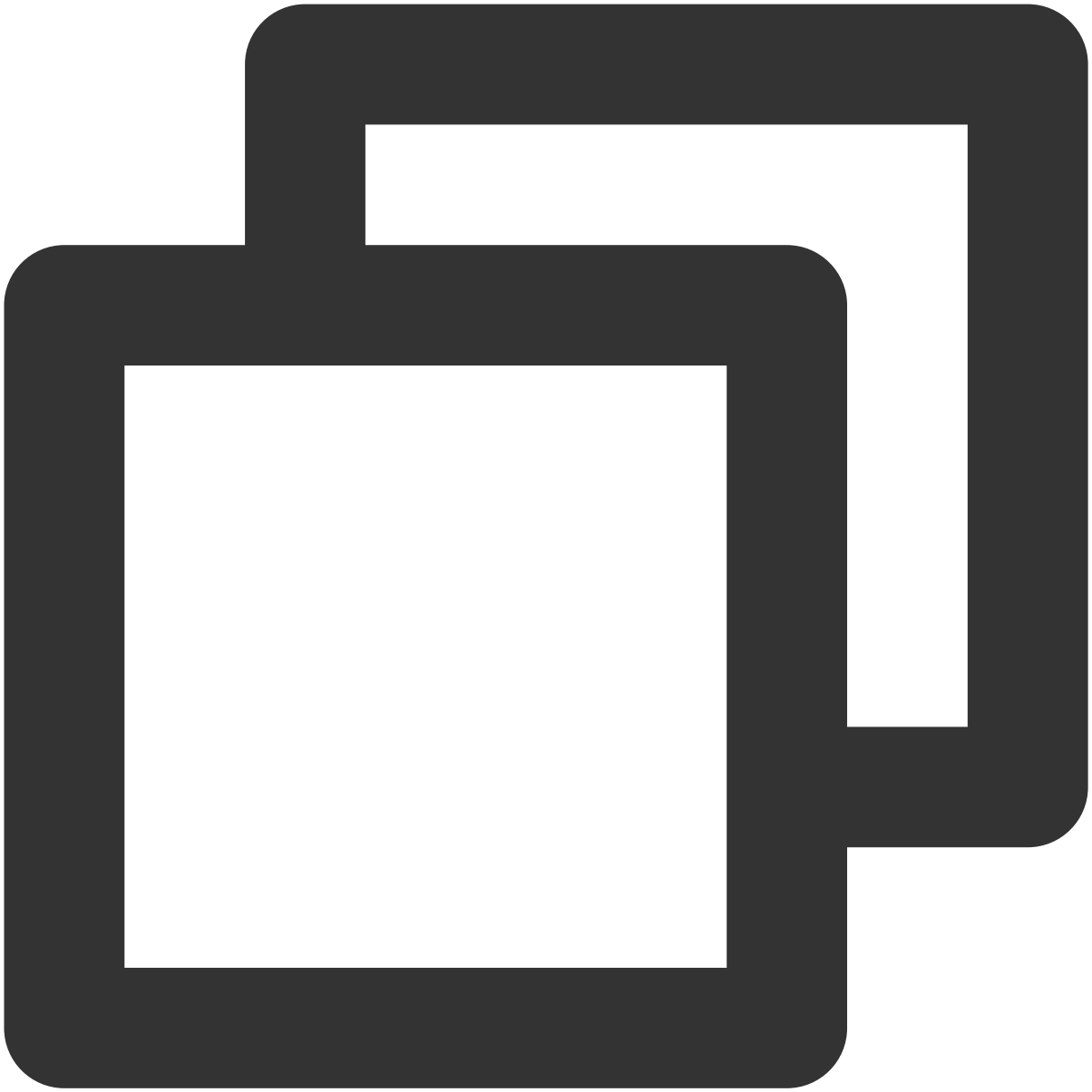
©2013-2022 Tencent Cloud. All rights reserved.

```
├── app.js
├── app.json
├── app.wxss
├── packageA
│   └── pages
│       ├── cat
│       └── dog
├── packageB
│   └── pages
│       ├── apple
│       └── banana
├── pages
│   ├── index
│   └── logs
└── utils
```

The developer declares the subpackage structure of a project in the subpackages field via app.json:

> The field can also be written as subPackages.

©2013-2022 Tencent Cloud. All rights reserved.



```
{
  "pages":[
    "pages/index",
    "pages/logs"
  ],
  "subpackages": [
    {
      "root": "packageA",
      "pages": [
        "pages/cat",
        "pages/dog"
```

©2013-2022 Tencent Cloud. All rights reserved.

```
        ]
      }, {
        "root": "packageB",
        "name": "pack2",
        "pages": [
          "pages/apple",
          "pages/banana"
        ]
      }
    ]
}
```

In `subpackages`, each subpackage are configured with the following fields:

Field	Type	Description
root	String	Root directory of subpackage
name	String	Alias of subpackage, which can be used in Subpackage Preload
pages	StringArray	Subpackage page path, which is relative to the root directory of the subpackage
independent	Boolean	Indicates whether a subpackage is an Independent Subpackage

1.2 Packaging Rules

After subpackages is declared, files are packaged based on the paths configured in subpackages, and files in the directories outside the paths in subpackages are packaged into the app (main package);
The app (main package) can also have its own pages (i.e. the outermost pages field);
The root directory of subpackage cannot be a subdirectory configured in another subpackage;
The tabBar page must be in the app (main package).

1.3 Referencing Rules

packageA cannot require JS files in packageB, but can require JS files in app and packageA;
packageA cannot import templates in packageB, but can import templates in app and packageA;
packageA cannot use resources in packageB, but can use resources in app and packageA.

2. Independent Subpackage

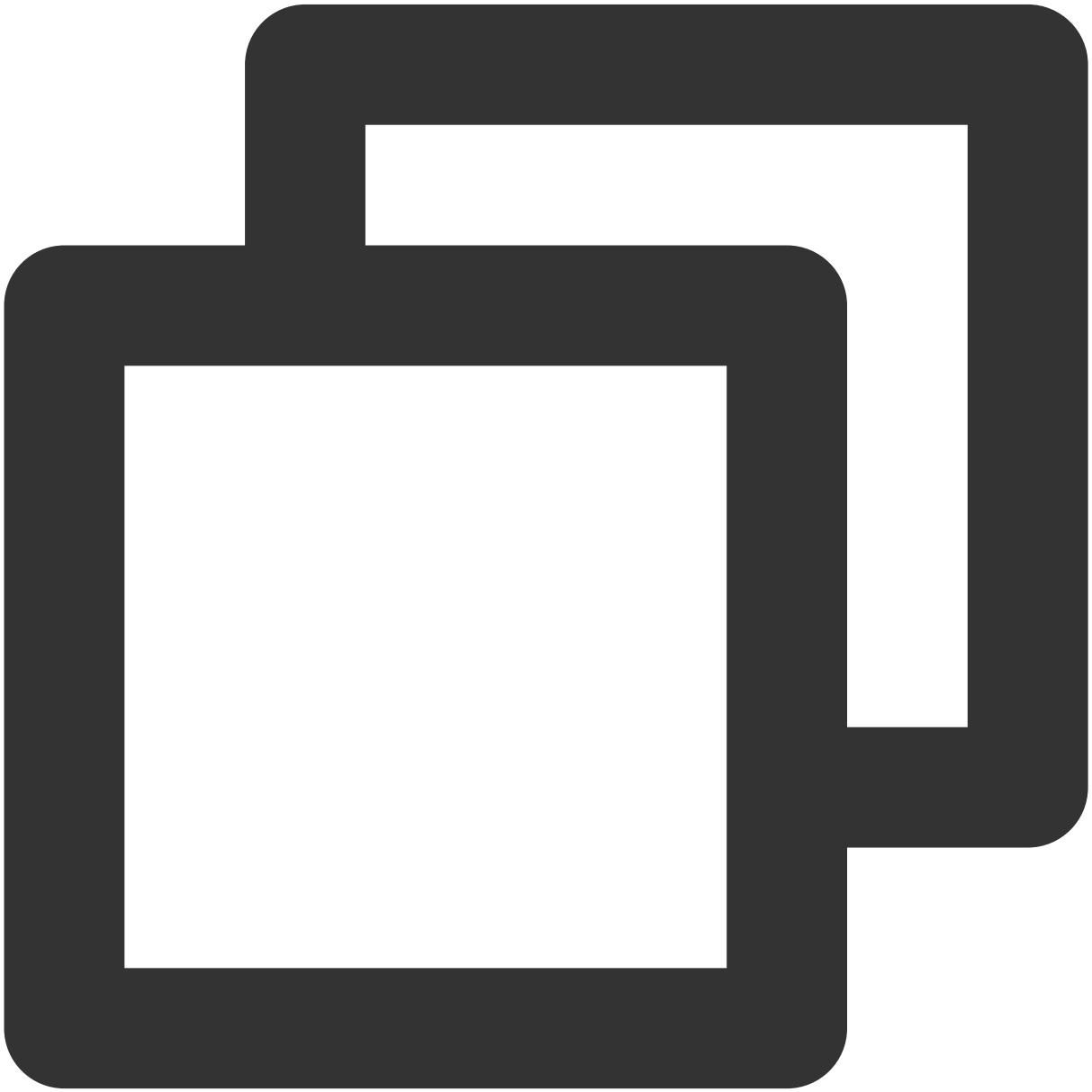
An independent subpackage is a special type of subpackage in Mini Programs. It can run independently of the main package and other subpackages. When a user enters a Mini Program from a page of an independent subpackage, the main package will not be downloaded, unless the user enters a page of an ordinary subpackage or a main package.

©2013-2022 Tencent Cloud. All rights reserved.

Developers can add some functionally independent pages to an independent subpackage as needed. When a Mini Program is launched from a page of an ordinary subpackage, the main package should be downloaded first. However, an independent subpackage can run independently of the main package. Therefore, subpackages greatly improve the startup speed of Mini Programs.
A Mini Program can have multiple independent subpackages.

2.1 Configuration

Assume that the directory structure of a Mini Program is as follows:



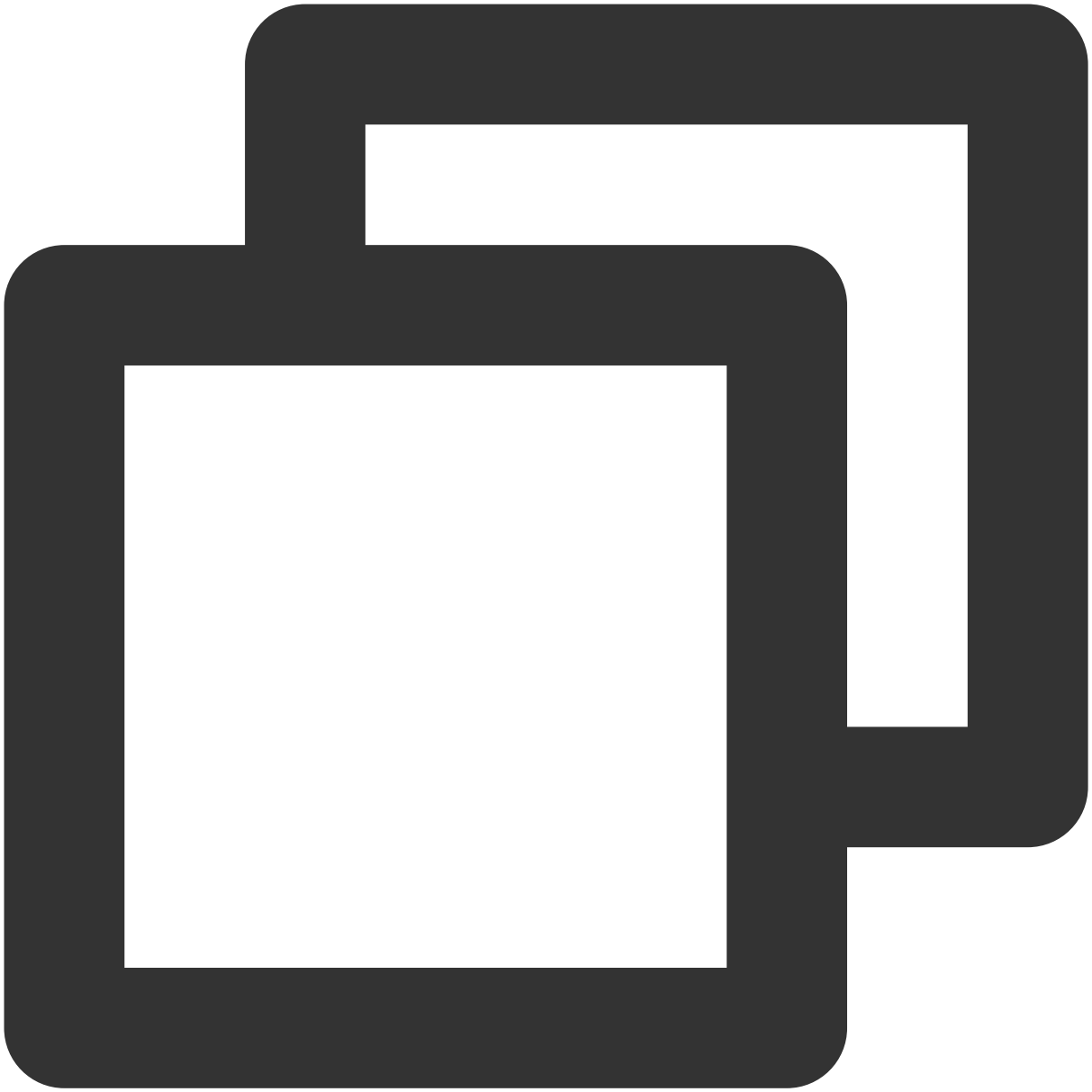
```
├── app.js
```

©2013-2022 Tencent Cloud. All rights reserved.

```
├── app.json
├── app.wxss
├── moduleA
│   └── pages
│       ├── rabbit
│       └── squirrel
├── moduleB
│   └── pages
│       ├── pear
│       └── pineapple
├── pages
│   ├── index
│   └── logs
└── utils
```

The developer declares a subpackage as an independent subpackage by defining the independent field in the subpackage configuration item in the `subpackages` field of `app.json` .

©2013-2022 Tencent Cloud. All rights reserved.



```
{
  "pages": [
    "pages/index",
    "pages/logs"
  ],
  "subpackages": [
    {
      "root": "moduleA",
      "pages": [
        "pages/rabbit",
        "pages/squirrel"
```

©2013-2022 Tencent Cloud. All rights reserved.

```
        ]
      }, {
        "root": "moduleB",
        "pages": [
          "pages/pear",
          "pages/pineapple"
        ],
        "independent": true
      }
    ]
}
```

2.2 Limits

The independent subpackage is a type of subpackage. All limits on ordinary subpackages are applicable to independent subpackages. Plug-ins and custom components in an independent subpackage are processed in the same way as with those in an ordinary subpackage:
Also note the following when using independent subpackages:
The content of an independent subpackage must be independent of the main package and other subpackages, including JS files, template, wxss, custom components, and plug-ins. app.wxss in the main package does not take effect for independent subpackages. Therefore, the styles in app.wxss should be avoided on pages of independent subpackages;
App can be defined only in the main package. Defining App in independent subpackages will cause unexpected behaviors;
Plug-ins are not supported in independent subpackages.

2.3 Notes

2.3.1 About getApp()

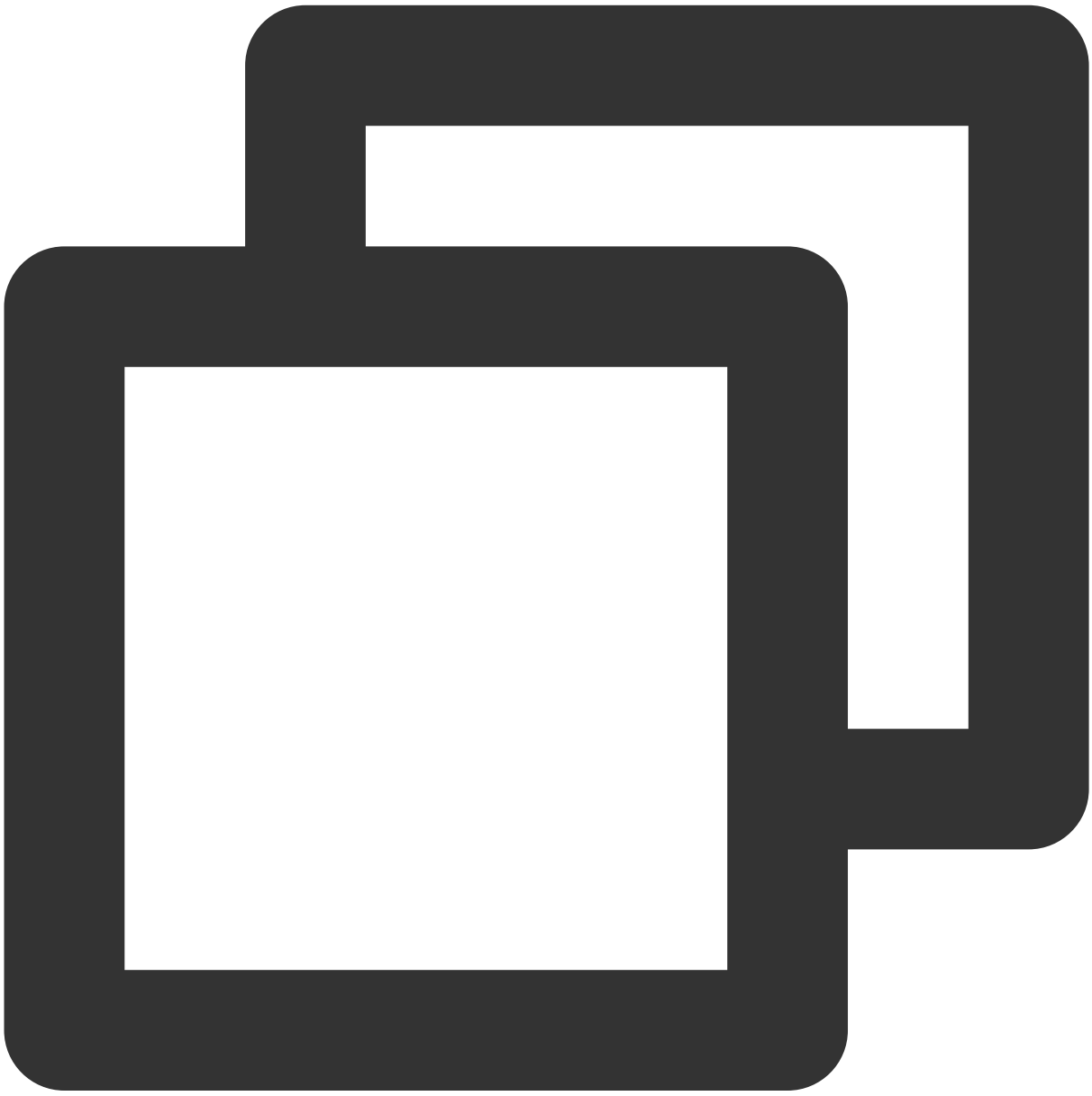
Unlike ordinary sub-packages, when an independent subpackage is running, App is not necessarily registered. Therefore, an App object can not necessarily be obtained via getApp():
When a user starts a Mini Program from a page of an independent subpackage, neither main package nor App exists. In this case, undefined is obtained when getApp() is called. The main package is downloaded and App can be registered only when the user opens a page of an ordinary subpackage or a main package.
When a user is redirected to a page of an independent subpackage from a page of an ordinary subpackage or the main package, the main package already exists, and the actual App can be obtained by calling getApp().
Because of this restriction, developers can not share global variables between independent subpackages and other parts of a Mini Program by using the App object.

©2013-2022 Tencent Cloud. All rights reserved.

To satisfy this requirement in a standalone sub-package, getApp supports the [allowDefault] parameter, which returns a default implementation when the App is undefined. When the main package is loaded and the App is registered, the properties defined in the default implementation will be overridden and merged into the real App.

Sample code:

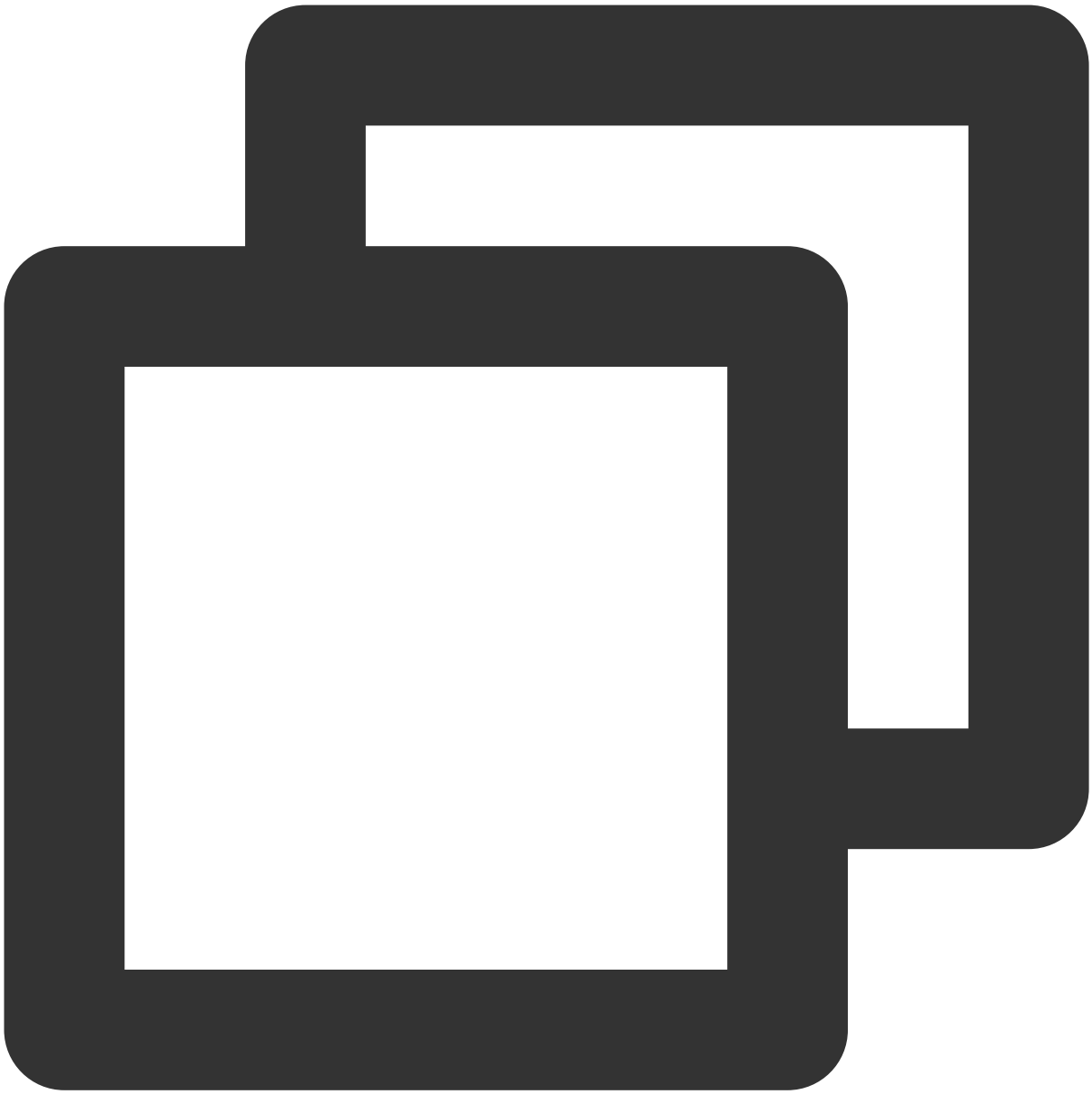
In the independent subpackage



```
const app = getApp({allowDefault: true}) // {}
app.data = 456
app.global = {}
```

©2013-2022 Tencent Cloud. All rights reserved.

In the app.js file



```
App({
  data: 123,
  other: 'hello'
})

console.log(getApp()) // {global: {}, data: 456, other: 'hello'}
```

2.3.2 About the lifecycle of App

©2013-2022 Tencent Cloud. All rights reserved.

If the Mini Program is launched from a page of an independent subpackage, onLaunch and first-time onShow of `App` in the main package are called when the user is redirected from the page of the independent subpackage to a page of the main package or an ordinary subpackage for the first time.
Because App cannot be defined in an independent subpackage, the lifecycle of a Mini Program can be listened via wx.onAppShow and wx.onAppHide. Other events on App can be listened via wx.onError and wx.onPageNotFound.

3. Subpackage Preload

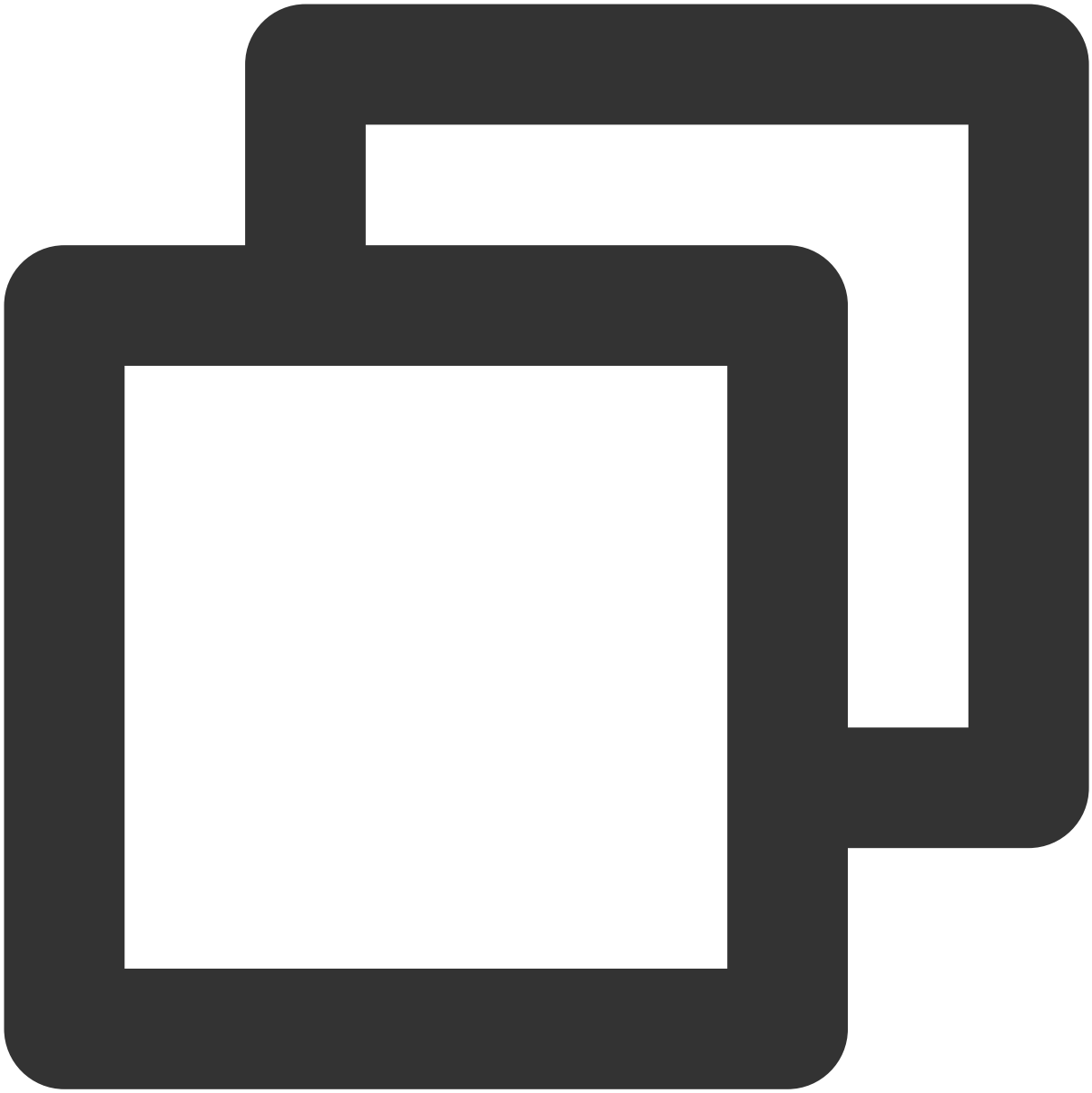
Developers can configure the framework to automatically preload required subpackages when a user enters a Mini Program page, to speed up the subsequent startup of subSubcontracting Asynchronisationpackaged pages. For Independent Subpackages, the main package can also be preloaded.
Subpackages can be preloaded only via configuration.

> Log prefixed with preloadSubpackages is available in vConsole to verify the preload status

3.1 Configuration

Subpackage preload is triggered when a user enters a specific page, and is controlled by adding `preloadRule` configuration items via `app.json` .

©2013-2022 Tencent Cloud. All rights reserved.



```
{
  "pages": ["pages/index"],
  "subpackages": [
    {
      "root": "important",
      "pages": ["index"],
    },
    {
      "root": "sub1",
      "pages": ["index"],
    },
```

©2013-2022 Tencent Cloud. All rights reserved.

```
    {
      "name": "hello",
      "root": "path/to",
      "pages": ["index"]
    },
    {
      "root": "sub3",
      "pages": ["index"]
    },
    {
      "root": "indep",
      "pages": ["index"],
      "independent": true
    }
  ],
  "preloadRule": {
    "pages/index": {
      "network": "all",
      "packages": ["important"]
    },
    "sub1/index": {
      "packages": ["hello", "sub3"]
    },
    "sub3/index": {
      "packages": ["path/to"]
    },
    "indep/index": {
      "packages": ["__APP__"]
    }
  }
}
```

In `preloadRule`, `key` is the page path, and `value` is the preload configuration for entering the page. Each configuration includes the following items:

Field	Type	Required	Default	Description
packages	StringArray	Yes	No	Indicates root or name of a subpackage preloaded after a user enters a page. __APP__ indicates the main package.
network	String	No	wifi	Indicates a specified network over which a subpackage is preloaded. Values: all: Any network wifi: Wi-Fi only

©2013-2022 Tencent Cloud. All rights reserved.

3.2 Limits

Pages of the same subpackage share the same maximum preload size of 2 MB, which is verified when the pages are packaged via the DevTools.
For example, Page A and Page B are in the same subpackage. If a subpackage with a total size of 0.5 MB is preloaded on Page A, a maximum of 1.5 MB subpackages can be preloaded on Page B.

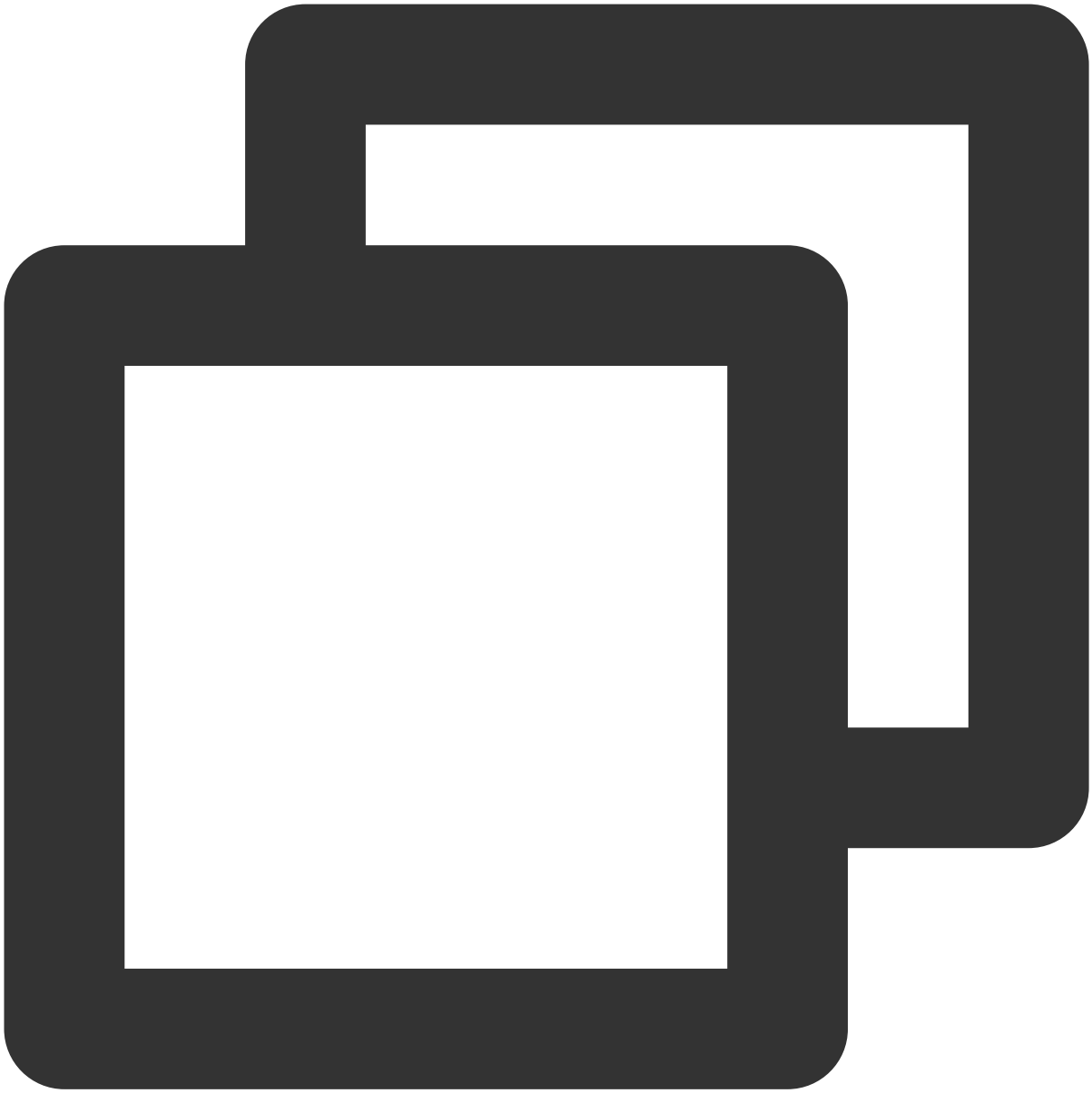
4. Subcontracting Asynchronisation

In mini programs, different sub-packages correspond to different download units; therefore, except for non-independent sub-packages that can rely on the main package, sub-packages cannot use custom components or require each other. the "sub-package asynchronisation" feature will allow this limitation to be solved to a certain extent by allowing, through some configurations and new interfaces, some cross-sub-packages to wait for the download and then use the content asynchronously.

4.1 Cross-subcontract custom component references

When a sub-package uses a custom component from another sub-package, the other sub-package's component is in an unavailable state because the other sub-package has not yet downloaded or injected it. By setting a placeholder component for the custom component of the other sub-package, we can render the placeholder component as a replacement first, and then replace it after the sub-package has finished downloading. Example:

©2013-2022 Tencent Cloud. All rights reserved.



```
// subPackageA/pages/index.json
{
  "usingComponents": {
    "button": "../../commonPackage/components/button",
    "list": "../../subPackageB/components/full-list",
    "simple-list": "../components/simple-list",
    "plugin-comp": "plugin://pluginInSubPackageB/comp"
  },
  "componentPlaceholder": {
    "button": "view",
    "list": "simple-list",
```

©2013-2022 Tencent Cloud. All rights reserved.

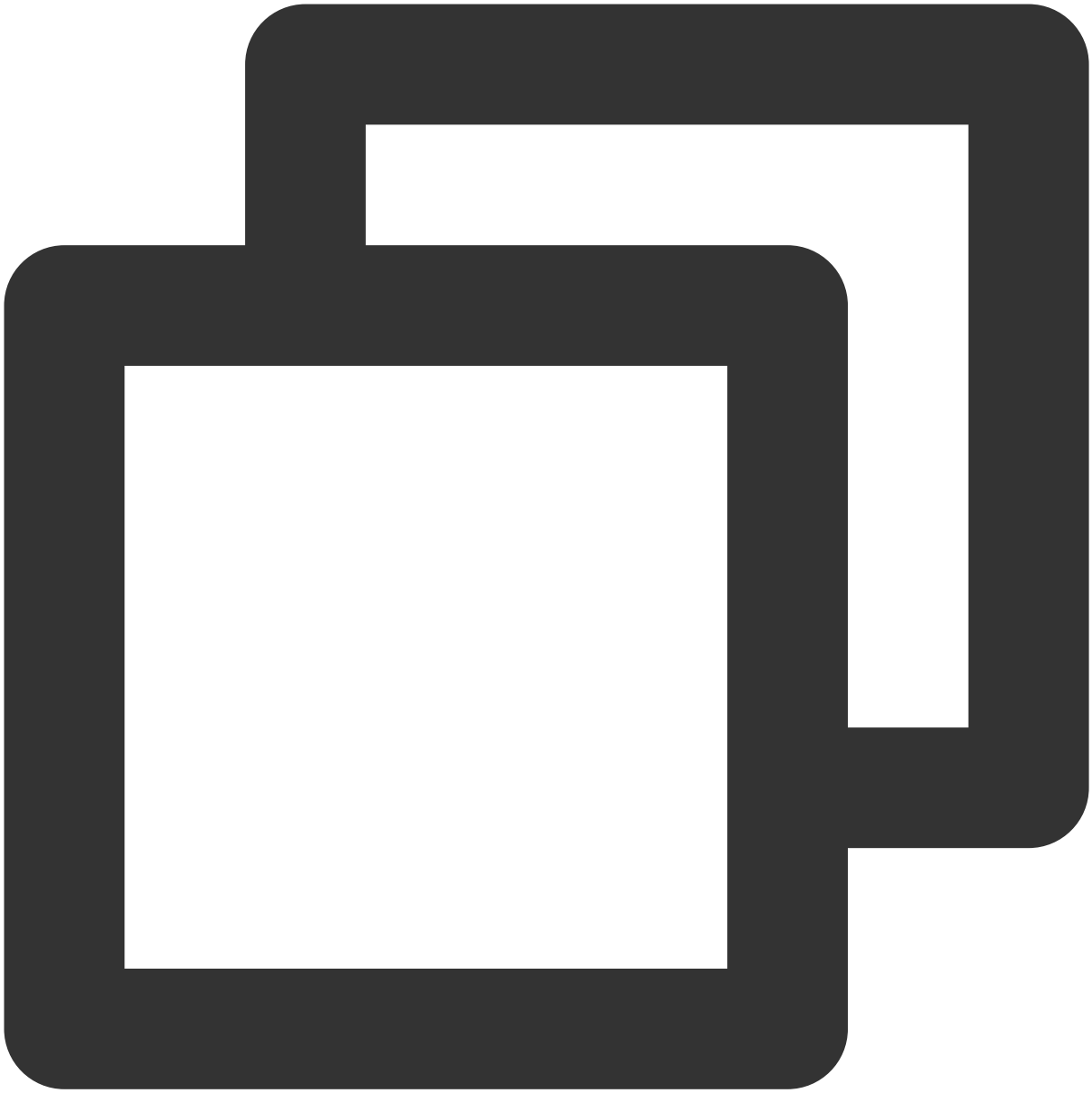
```
    "plugin-comp": "view"
  }
}
```

In this configuration, the button and list custom components are cross-package references, where the button is rendered using the built-in component view as a replacement, and the list is rendered using the current sub-package's custom component simple-list as a replacement; after these two sub-packages have been downloaded, the placeholder component is replaced with the corresponding cross-package component After both packages are downloaded, the placeholder component is replaced with the corresponding cross-package component.

4.2 Cross-Subpackage JS Code References

When code in a sub-package references code in another sub-package, we need to asynchronously get the result of the reference in order to prevent the download from blocking the code. For example:

©2013-2022 Tencent Cloud. All rights reserved.



```
// subPackageA/index.js
// Use a callback function style call
require('../subPackageB/utils.js', utils => {
  console.log(utils.whoami) // Wechat MiniProgram
}, ({mod, errMsg}) => {
  console.error(`path: ${mod}, ${errMsg}`)
})
// Or use a Promise-style call
require.async('../commonPackage/index.js').then(pkg => {
  pkg.getPackageName() // 'common'
}).catch(({mod, errMsg}) => {
```

©2013-2022 Tencent Cloud. All rights reserved.

```
  console.error(`path: ${mod}, ${errMsg}`)
})
```

See the Modularity - require section for more details.

©2013-2022 Tencent Cloud. All rights reserved.

Canvas

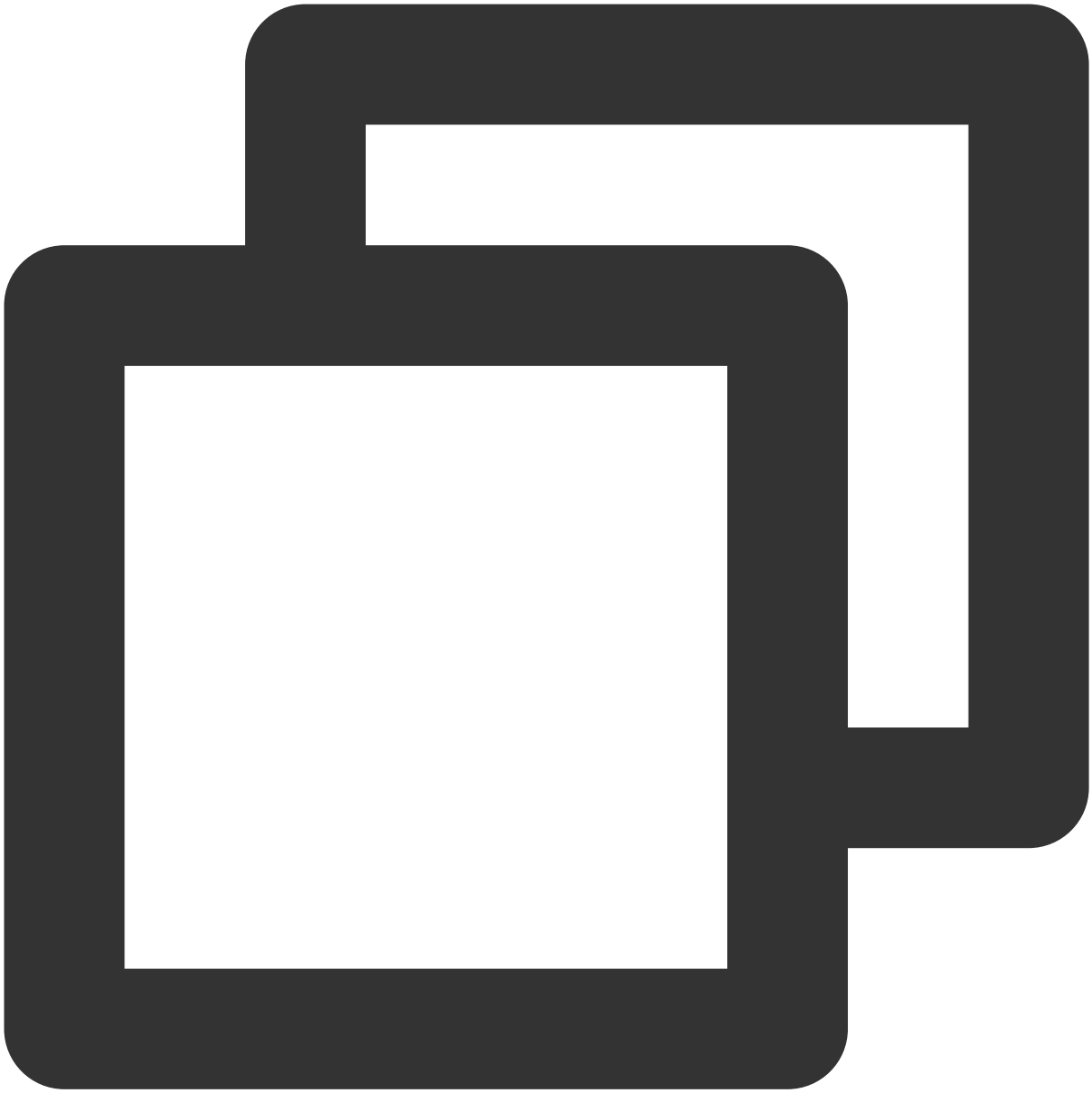
Last updated：2024-07-12 17:25:05

The Canvas provides a drawing interface on which you can draw anything you want.

1. Basic use

Step 1：Create a Canvas drawing context

©2013-2022 Tencent Cloud. All rights reserved.



```
<!-- 2d type canvas -->
<canvas id="myCanvas" type="2d" style="border: 1px solid; width: 300px; height: 150
```

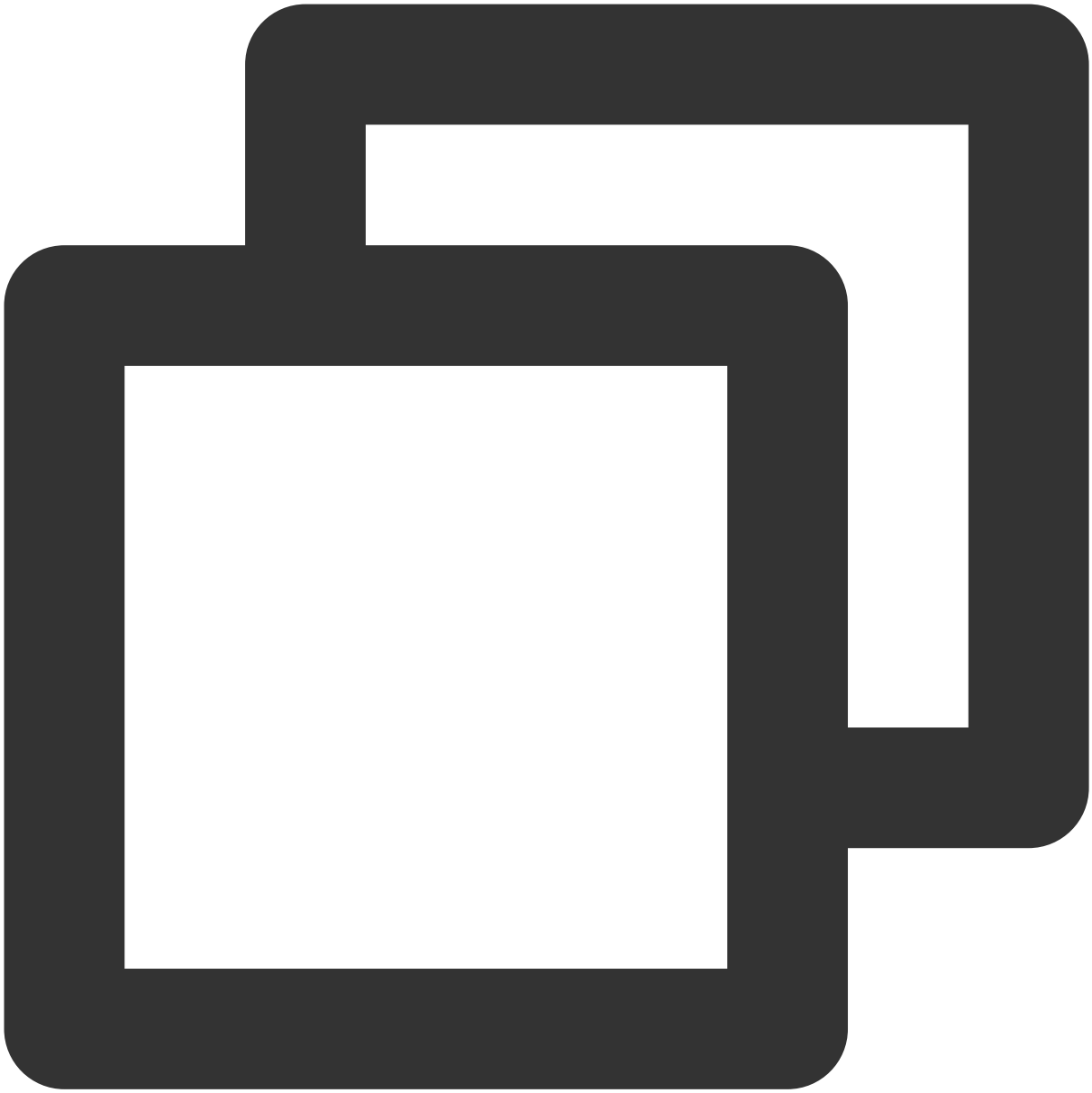
First you need to add the Canvas to the WXML.

Specify `id="myCanvas"` to uniquely identify a canvas, which will be used to get the Canvas object later.

Specify `type` to define the type of the canvas, for this example, `type="2d"` is used.

Step 2: Use the Canvas drawing context to describe the drawing

©2013-2022 Tencent Cloud. All rights reserved.



```
wx.createSelectorQuery()
    .select('#myCanvas') // id in WXML
    .fields({ node: true, size: true })
    .exec((res) => {
        // Canvas Object
        const canvas = res[0].node
        // rendering context
        const ctx = canvas.getContext('2d')
    })
```

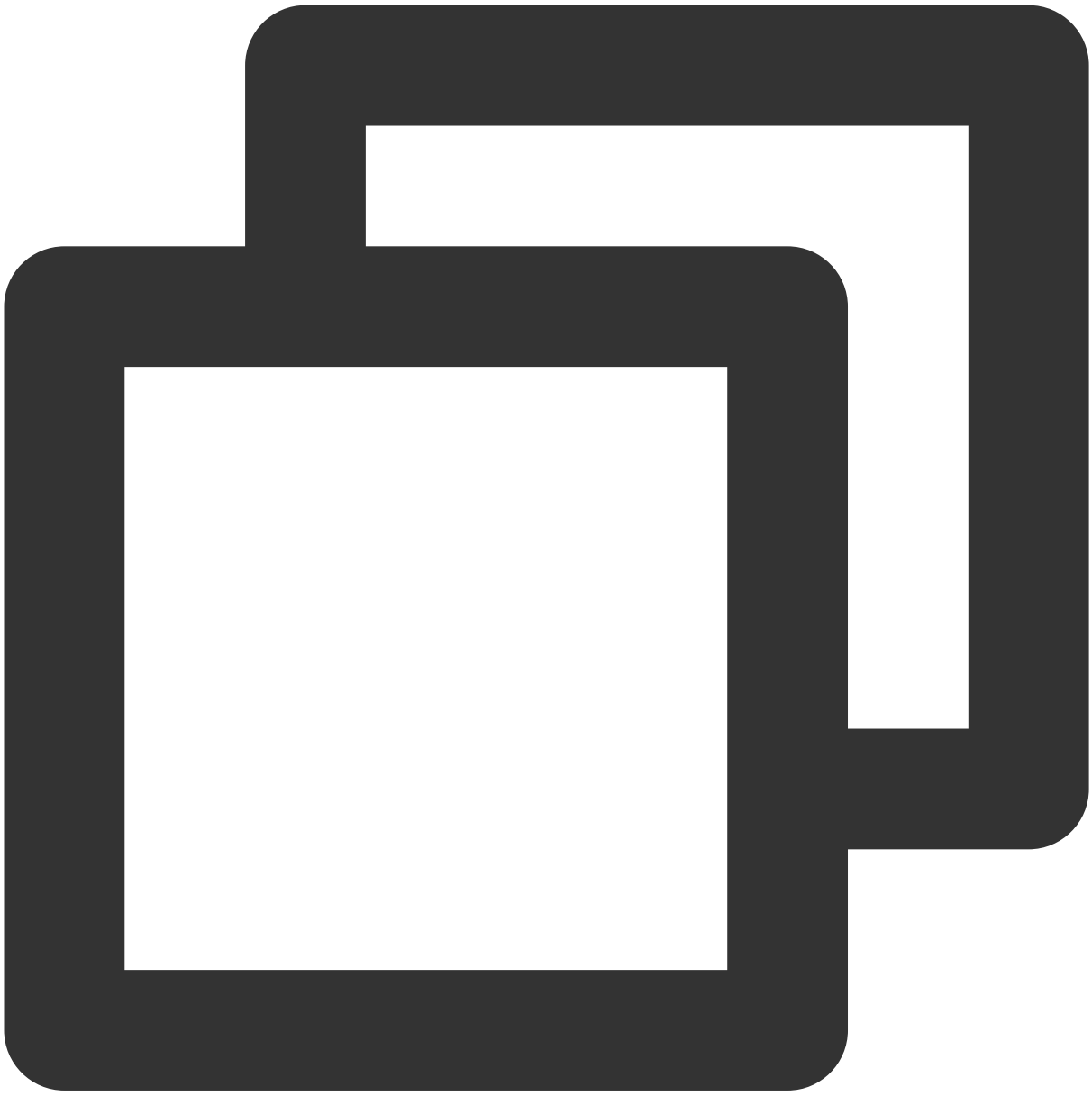
By selecting the canvas from the previous step via SelectorQuery, we can get the Canvas.

©2013-2022 Tencent Cloud. All rights reserved.

And through Canvas.getContext, we can get the rendering context RenderingContext.

Subsequent canvas operations and rendering operations need to be implemented through these two objects.

Step 3: Initialise the Canvas



```
wx.createSelectorQuery()
    .select('#myCanvas') // id in WXML
    .fields({ node: true, size: true })
    .exec((res) => {
        // Canvas Object
        const canvas = res[0].node
```

©2013-2022 Tencent Cloud. All rights reserved.

```
        // rendering context
        const ctx = canvas.getContext('2d')


        // The actual drawing width and height of the Canvas.
        const width = res[0].width
        const height = res[0].height


        // Initialise canvas size
        const dpr = wx.getWindowInfo().pixelRatio
        canvas.width = width * dpr
        canvas.height = height * dpr
        ctx.scale(dpr, dpr)
    })
```

The canvas width is divided into a rendered width and a logical width:

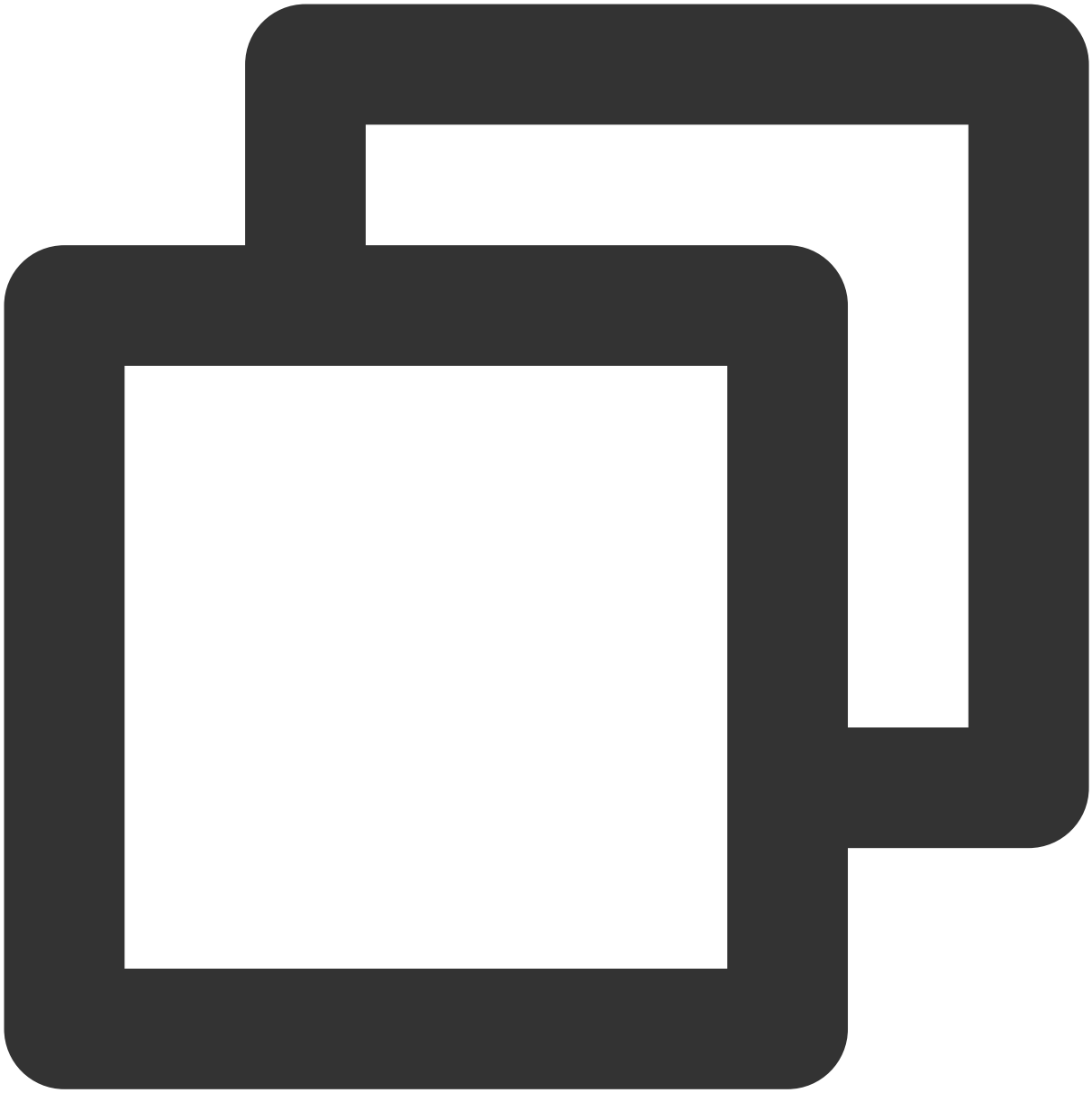
Rendered aspect is the actual width and height of the canvas canvas on the page, i.e. the size obtained by boundingClientRect requests to the nodes.

The logical width is the size of the logical width of the canvas during the rendering process, e.g. if you draw a rectangle with the same logical width, the rectangle will eventually fill the entire canvas. The logical width defaults to 300 * 150.

On different devices, there are cases where the physical pixels and logical pixels are not equal, so generally we need to use wx.getWindowInfo to get the pixel ratio of the device, multiply it by the rendered size of the canvas, and use it as the logical size of the canvas.

Step 4: Drawing

©2013-2022 Tencent Cloud. All rights reserved.



```
// Omit the initialisation steps above, the canvas object and ctx rendering context

// Empty the canvas
ctx.clearRect(0, 0, width, height)

// Drawing a red square
ctx.fillStyle = 'rgb(200, 0, 0)';
ctx.fillRect(10, 10, 50, 50);

// Drawing Blue Translucent Squares
ctx.fillStyle = 'rgba(0, 0, 200, 0.5)';
```

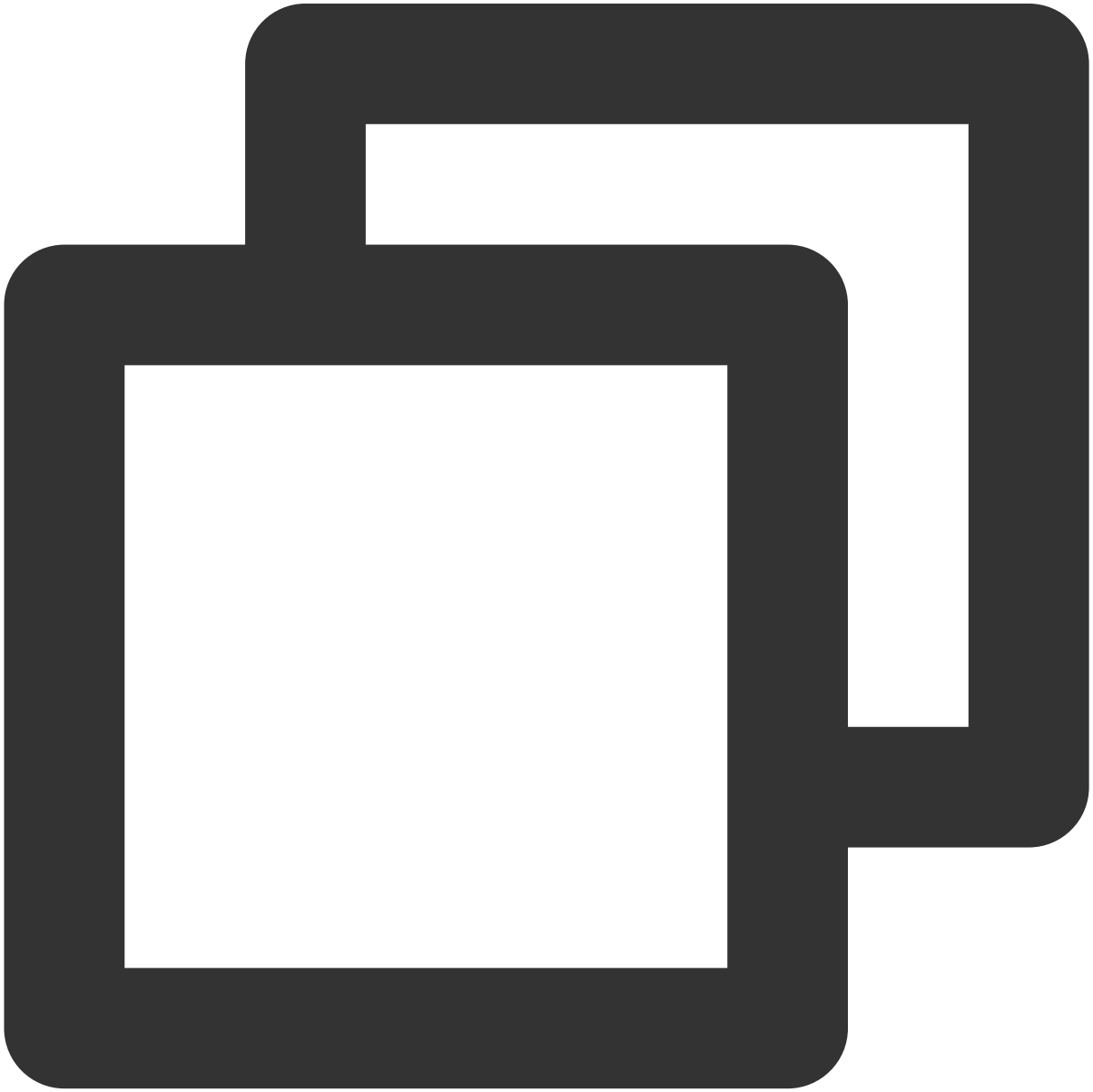
©2013-2022 Tencent Cloud. All rights reserved.

```
ctx.fillRect(30, 30, 50, 50);
```

The drawing api on the render context allows us to do arbitrary drawing on the canvas.

2. Advanced use

2.1 Drawing pictures



```
// Omit the initialisation steps above, the canvas object and ctx rendering context
```

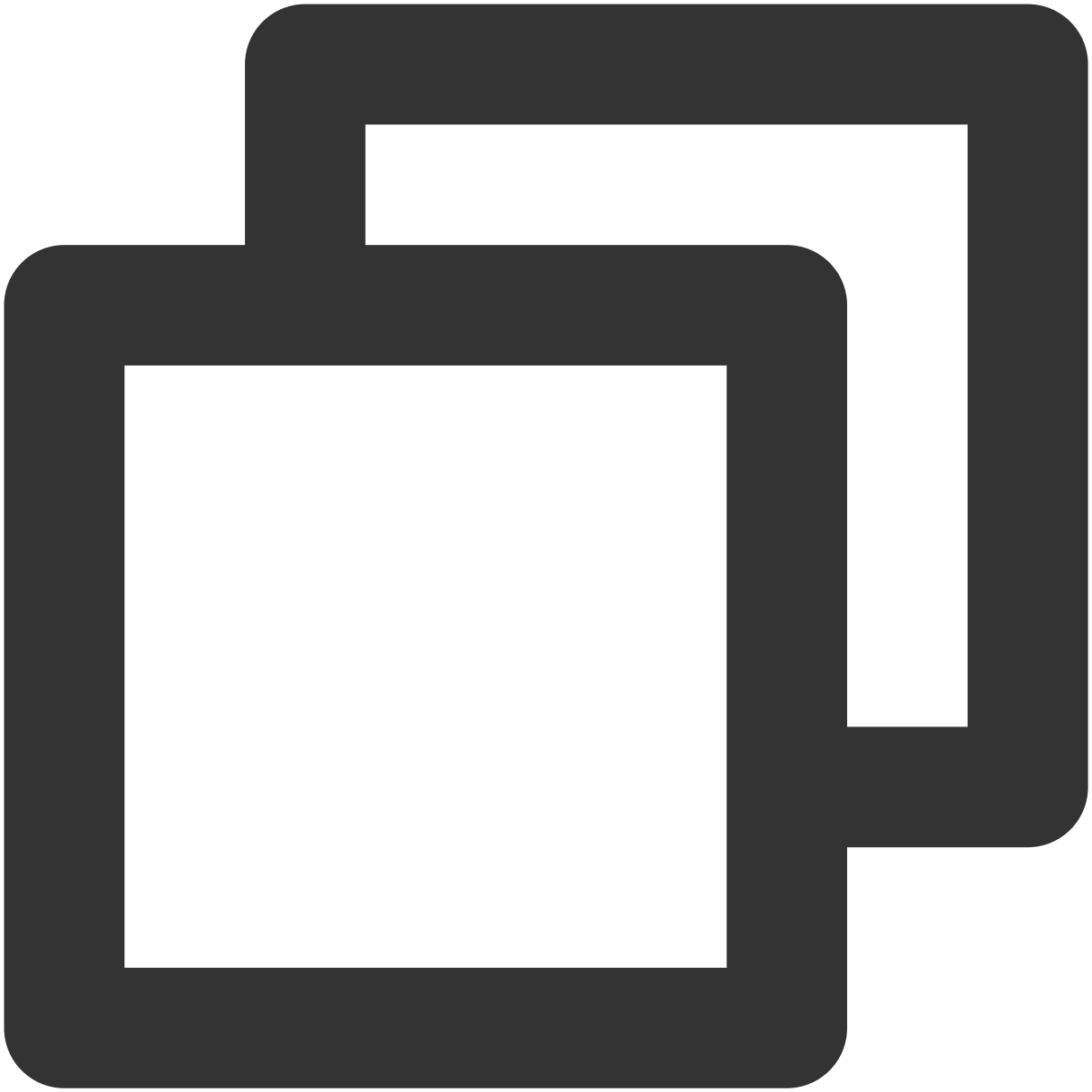
©2013-2022 Tencent Cloud. All rights reserved.

```
// The image object
const image = canvas.createImage()
// Image load completion callback
image.onload = () => {
    // Draw the image to the canvas
    ctx.drawImage(image, 0, 0)
}
// Set the image src
image.src = 'https://open.weixin.qq.com/zh_CN/htmledition/res/assets/res-design-dow
```

With Canvas.createImage we can create an image object and load the image. Once the image is loaded and the onload callback is triggered, use ctx.drawImage to draw the image to the canvas.

2.2 Generate Image

©2013-2022 Tencent Cloud. All rights reserved.



```
// Omitting the initialisation steps above, the canvas object and the ctx rendering


// Drawing a red square
ctx.fillStyle = 'rgb(200, 0, 0)';
ctx.fillRect(10, 10, 50, 50);


// Drawing Blue Translucent Squares
ctx.fillStyle = 'rgba(0, 0, 200, 0.5)';
ctx.fillRect(30, 30, 50, 50);
```

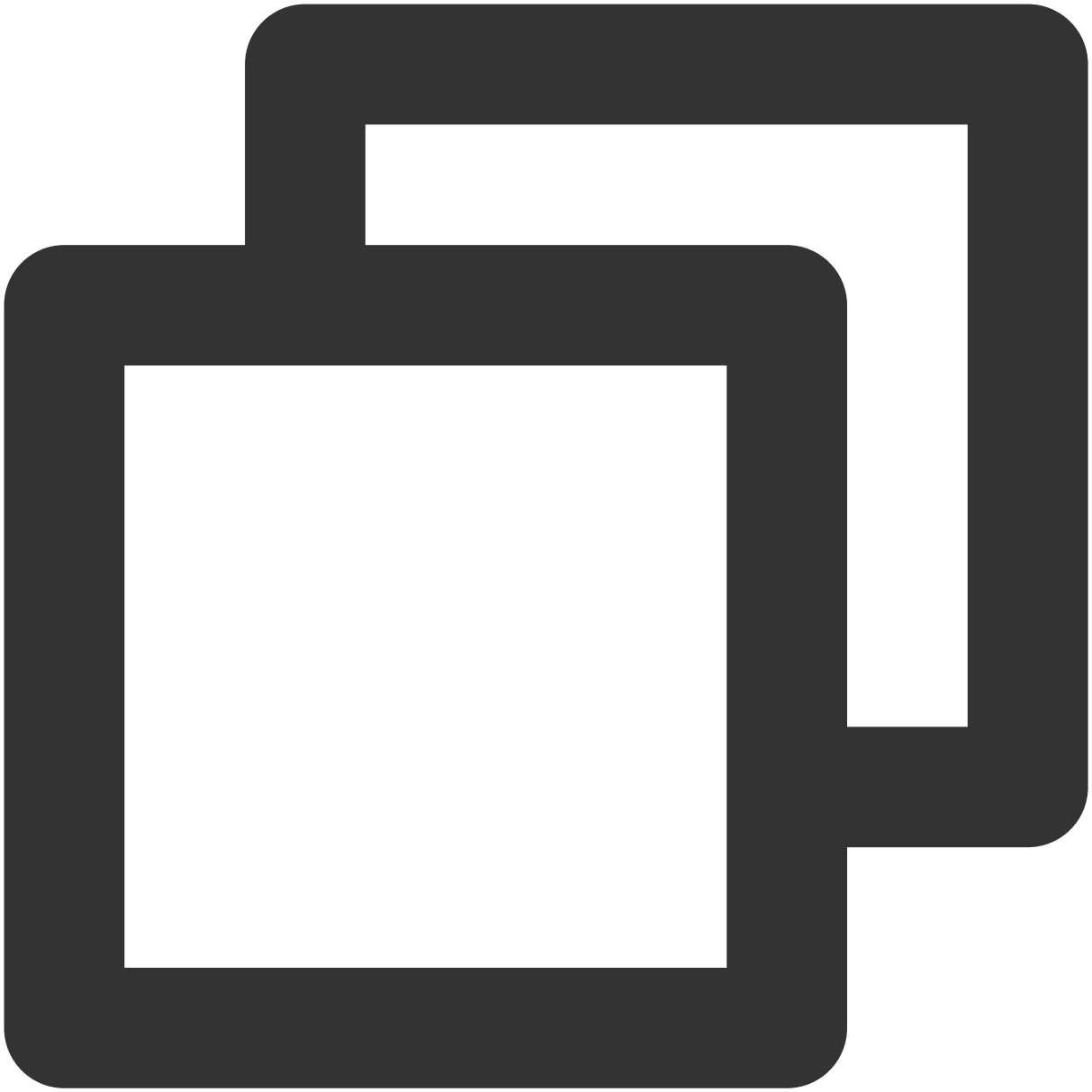
©2013-2022 Tencent Cloud. All rights reserved.

```
// Generate Image
wx.canvasToTempFilePath({
    canvas,
    success: res => {
        // Path to the generated image temporary file
        const tempFilePath = res.tempFilePath
    },
})
```

The wx.canvasToTempFilePath interface allows you to generate a temporary image file from the contents of a canvas.

2.3 Frame animation

©2013-2022 Tencent Cloud. All rights reserved.



```
// Omit the initialisation steps above, the canvas object and the ctx rendering con

const startTime = Date.now()


// Frame rendering callbacks
const draw = () => {
  const time = Date.now()
  // Calculate the elapsed time
  const elapsed = time - startTime
```

©2013-2022 Tencent Cloud. All rights reserved.

```
  // Calculate the animation position
  const n = Math.floor(elapsed / 3000)
  const m = elapsed % 3000
  const dx = (n % 2 ? 0 : 1) + (n % 2 ? 1 : -1) * (m < 2500 ? easeOutBounce(m / 250
  const x = (width - 50) * dx


  // Rendering
  ctx.clearRect(0, 0, width, height)
  ctx.fillStyle = 'rgb(200, 0, 0)';
  ctx.fillRect(x, height / 2 - 25, 50, 50);


  // Register for next frame rendering
  canvas.requestAnimationFrame(draw)
}

draw()
```

With Canvas.requestAnimationFrame, you can register an animation frame callback and draw the animation frame by frame within the callback.

2.4 Custom Fonts

Custom fonts can be loaded for the Canvas via wx.loadFontFace.

©2013-2022 Tencent Cloud. All rights reserved.

Custom tabBar

Last updated：2024-07-12 17:21:49

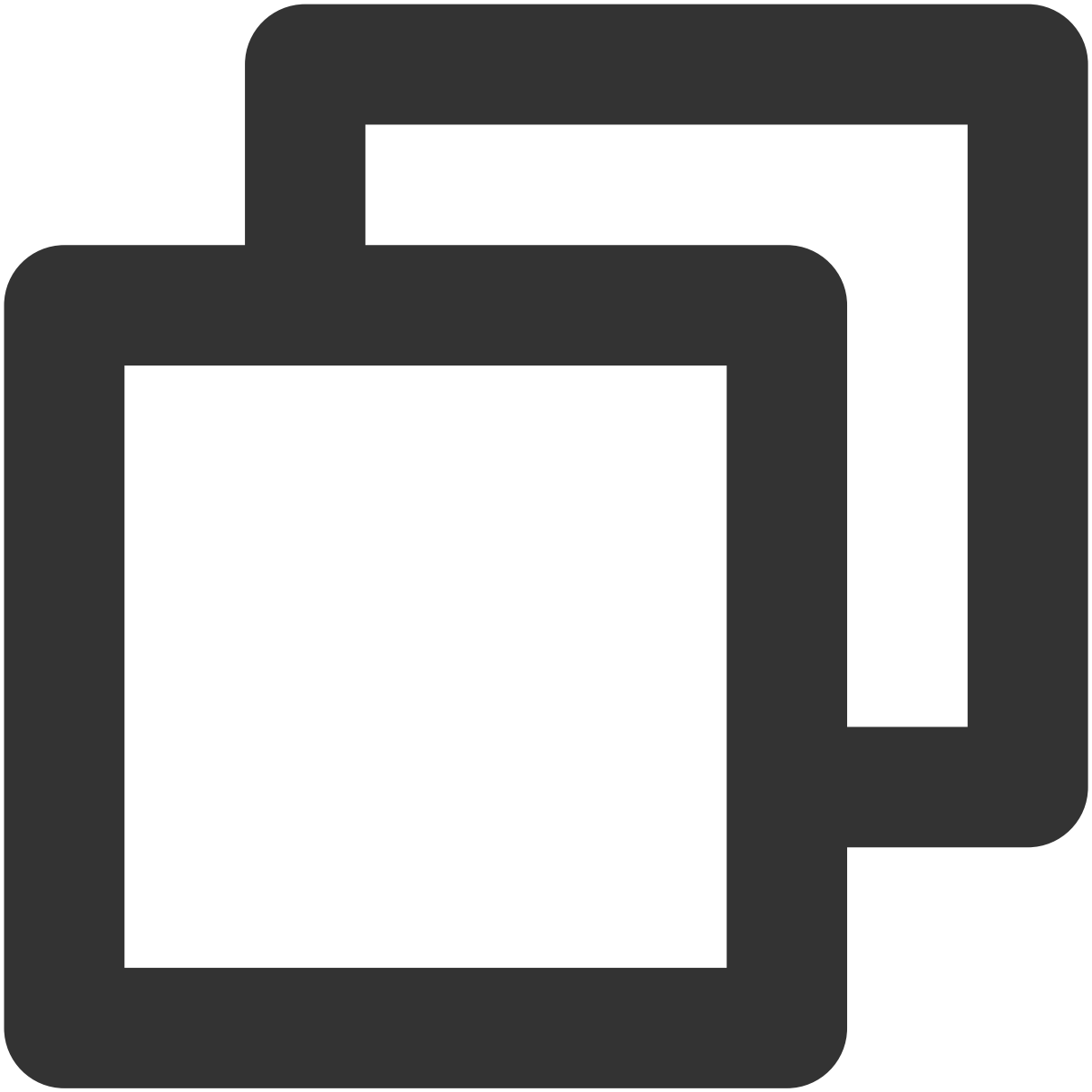
Custom tabBar allows developers to flexibly set tabBar styles and achieve greater personalization.
In custom tabBar Mode:
In order to ensure compatibility with older versions and identify pages that are tab pages, the relevant tabBar configuration items must be completely declared. However, these fields cannot be used to render custom tabBars.
Developers must provide a custom component to render the tabBar, and all tabBar styles must be rendered via the custom component. To ensure a higher tabBar level, we recommend using cover-view and cover-image components which are fixed at the bottom of the page to render styles.
APIs related to tabBar styles, such as wx.setTabBarItem, will be deactivated.
The custom tabBar component instances vary with different tab pages. You can use the getTabBar API under a custom component to get the custom tabBar component instance of the current page.
Note：
To put a tab in the selected status, on the current page, use the getTabBar API to get the component instance and call setData to update the selected status. See the code sample at the bottom for details.

How to use

Configure information

Specify the custom field of the tabBar item in app.json and set the other configurations related to tabBar;
The usingComponents item must be declared in json of all tab pages. You can also enable this option globally in app.json.
Code example:

©2013-2022 Tencent Cloud. All rights reserved.



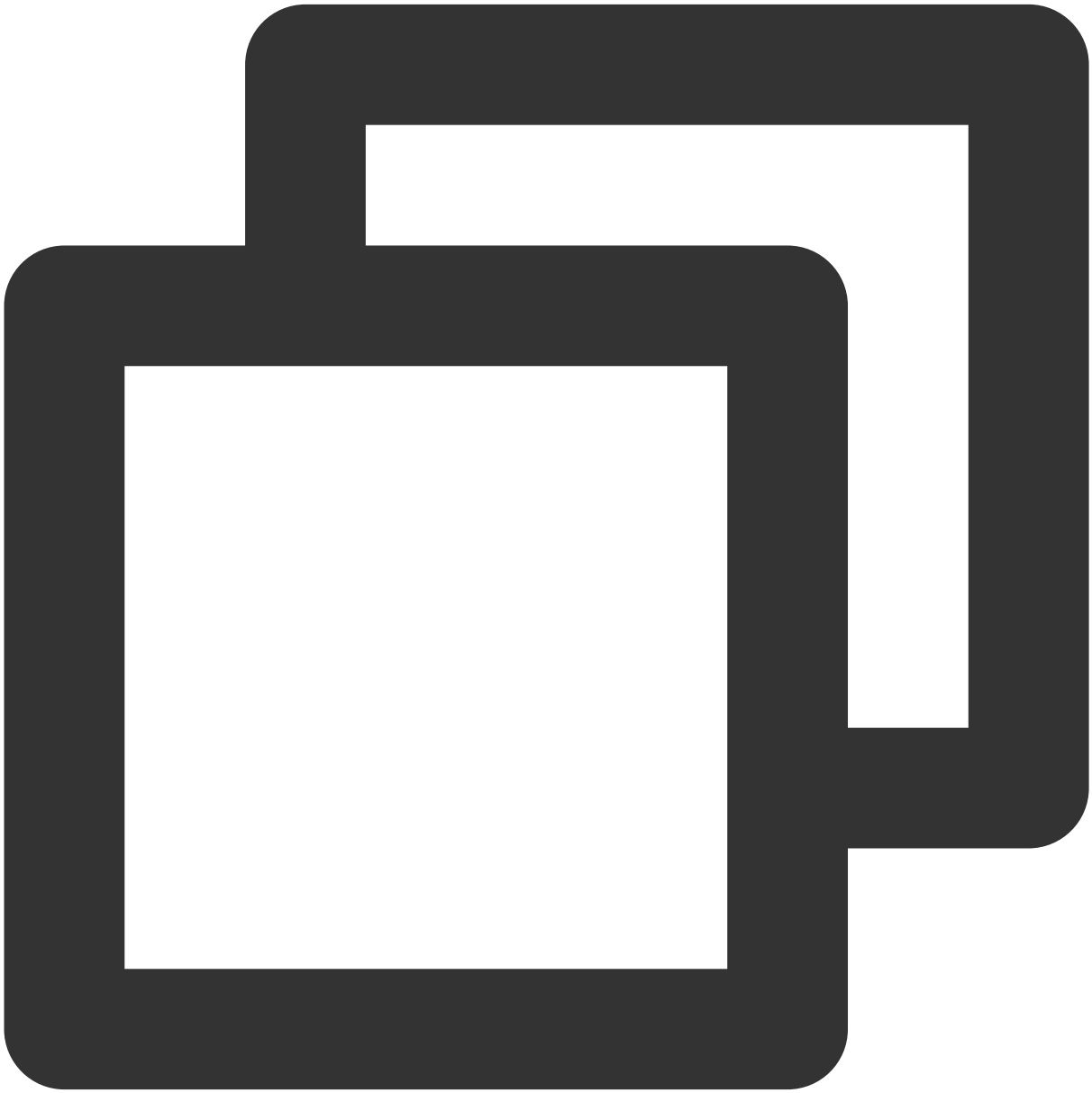
```
{
  "tabBar": {
    "custom": true,
    "color": "#000000",
    "selectedColor": "#000000",
    "backgroundColor": "#000000",
    "list": [{
      "pagePath": "page/component/index",
      "text": "Component"
    }, {
      "pagePath": "page/API/index",
```

©2013-2022 Tencent Cloud. All rights reserved.

```
        "text": "API"
      }]
    },
    "usingComponents": {}
}
```

Add tabBar code files

Add the entry files in the code root directory:



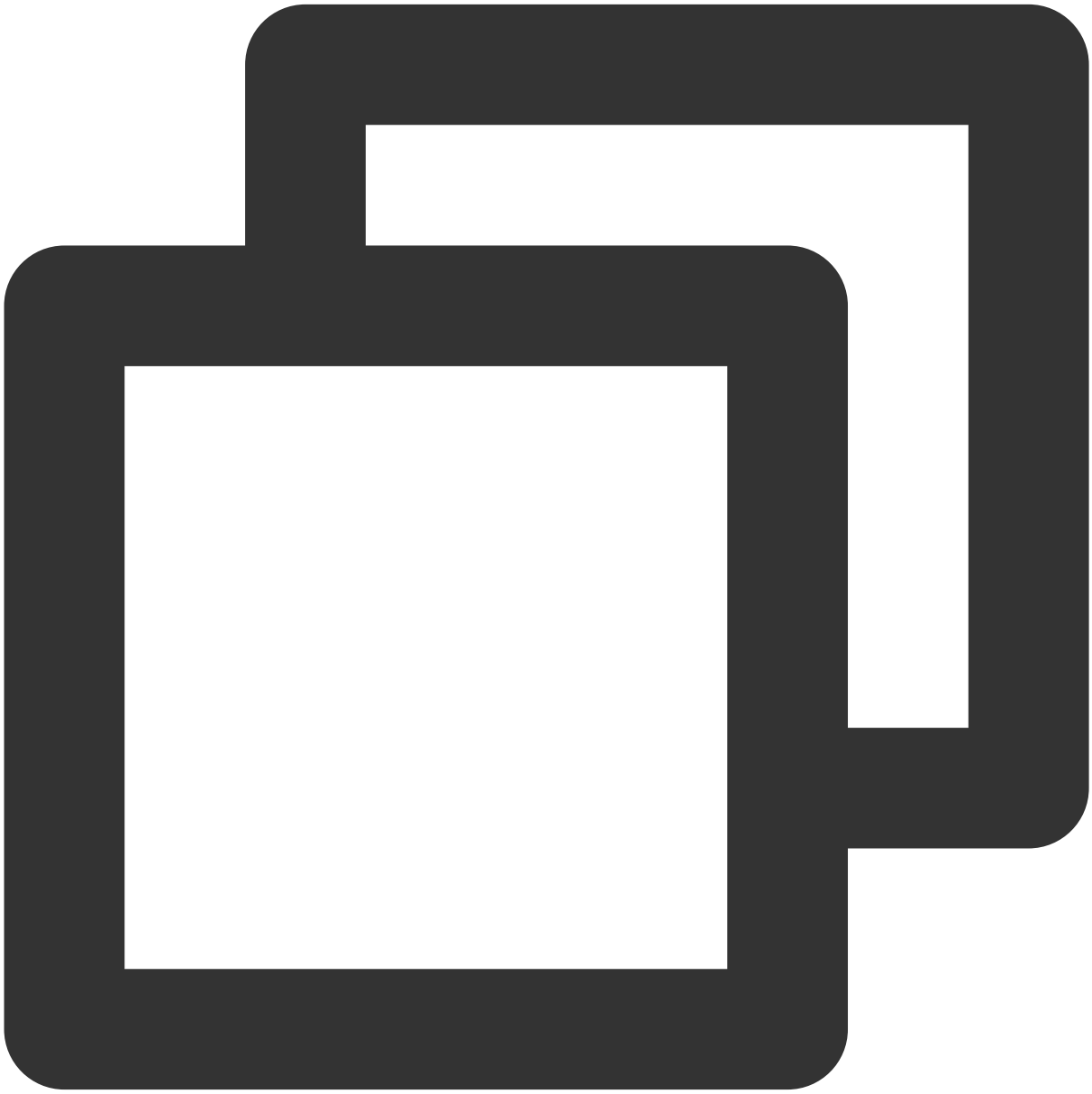
```
custom-tab-bar/index.js
```

©2013-2022 Tencent Cloud. All rights reserved.

```
custom-tab-bar/index.json
custom-tab-bar/index.wxml
custom-tab-bar/index.wxss
```

Write tabBar code

You can write the code using a custom component to render tabBar. In addition, the getTabBar API is added to the custom component to get the custom tabBar component instance of the current page.



```
Component({
  pageLifetimes: {
```

©2013-2022 Tencent Cloud. All rights reserved.

```
    show() {
      if (typeof this.getTabBar === 'function' &&
        this.getTabBar()) {
        this.getTabBar().setData({
          selected: 0
        })
      }
    }
  }
})
```

©2013-2022 Tencent Cloud. All rights reserved.

Multi-threaded Worker

Last updated：2024-07-12 18:29:08

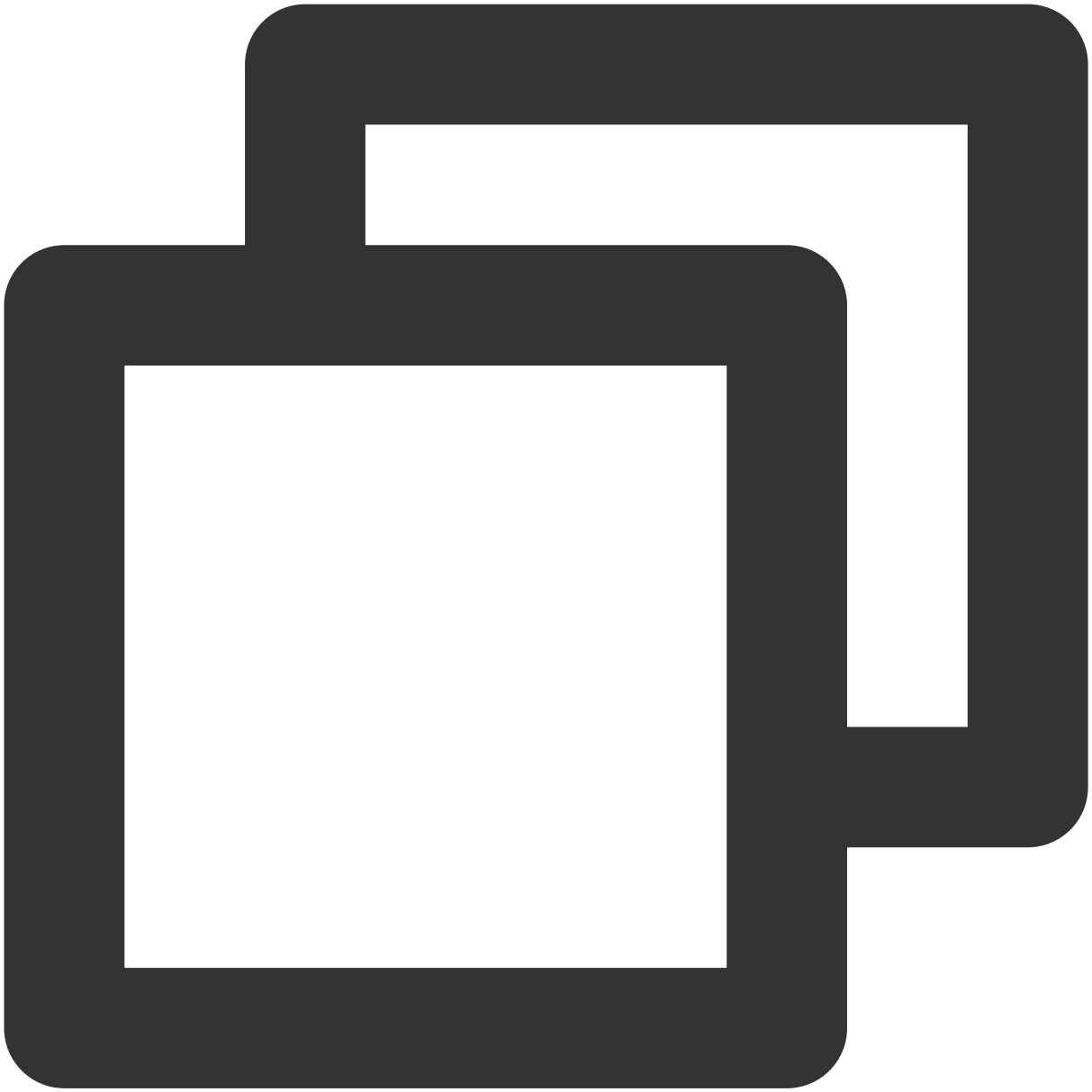
Some async tasks can be run in a worker. At the end of run, the results are returned to the main thread of the Mini Program. The worker runs in a separate global context and thread, and cannot directly call the methods of the main thread.
Data transfer between the worker and the main thread is accomplished using Worker.postMessage() to send data and Worker.onMessage() to receive data. The data to be transferred is not directly shared but copied.

Procedures

Configure Worker Information

In app.json, you can configure the directory where Worker code is placed. The code in the directory is packaged into a file:
Configuration example:

©2013-2022 Tencent Cloud. All rights reserved.



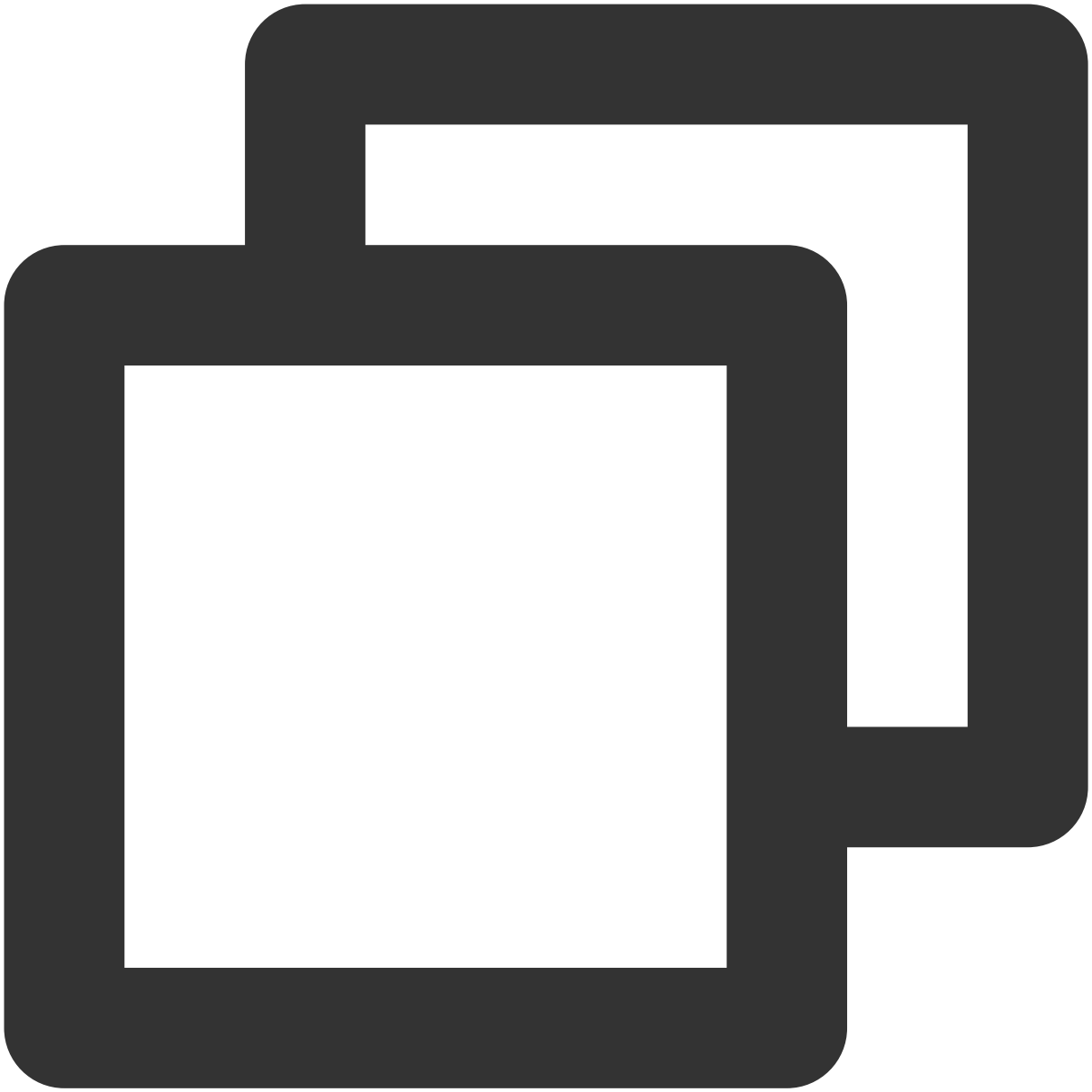
```
{
  "workers": "workers"
}
```

By configuring it in the above way, all the JS files under the workers directory will be packaged into one JS file and be part of the first package of the mini program.

Add Worker Code Files

Based on the configuration in Step 1, create the following two entry files:

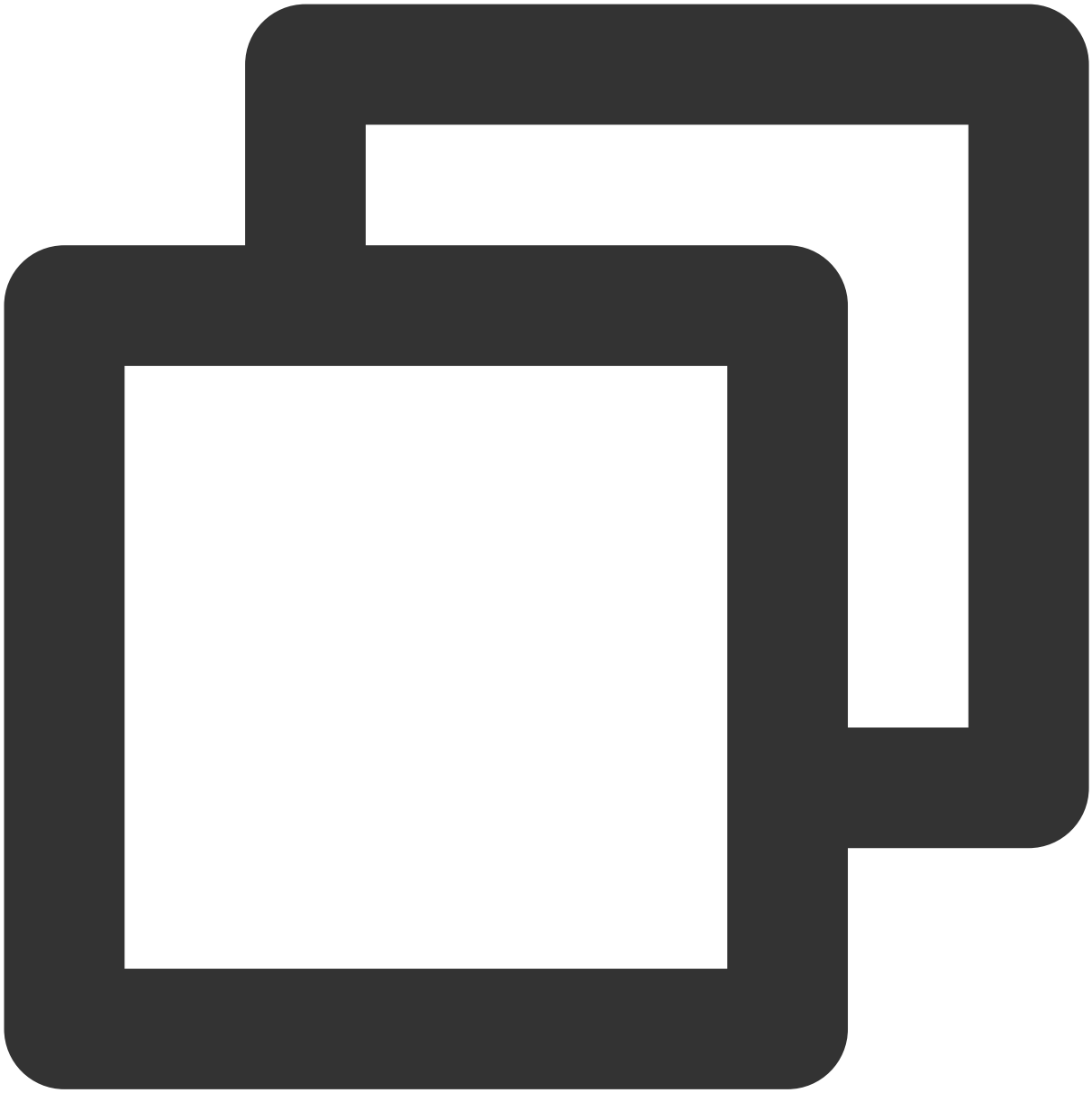
©2013-2022 Tencent Cloud. All rights reserved.



```
workers/request/index.js
workers/request/utils.js
workers/response/index.js
```

After the files are added, the directory structure is shown as below:

©2013-2022 Tencent Cloud. All rights reserved.

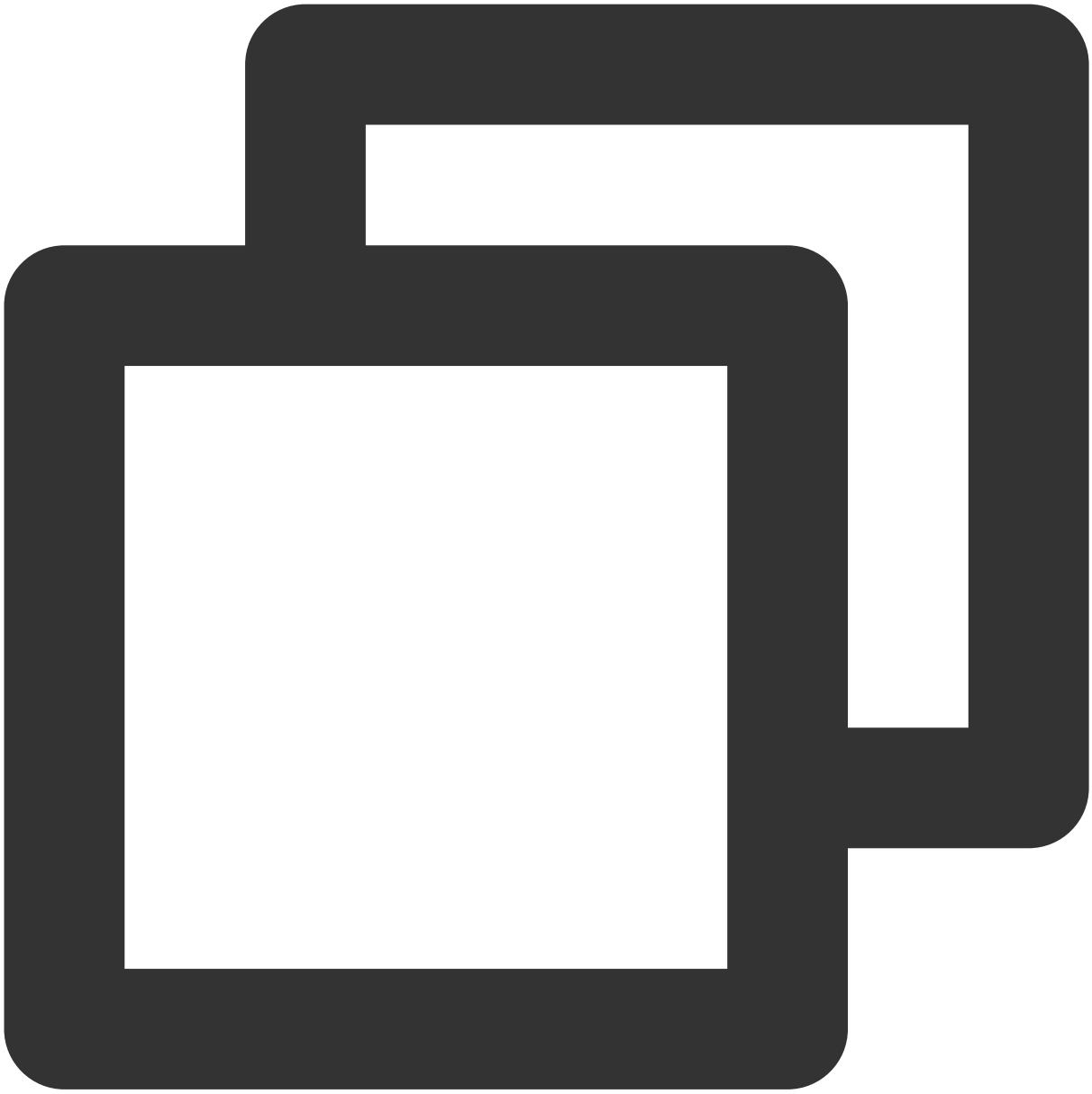


```
├── app.js
├── app.json
├── project.config.json
└── workers
    ├── request
    │   ├── index.js
    │   └── utils.js
    └── response
        └── index.js
```

©2013-2022 Tencent Cloud. All rights reserved.

Write Worker Code

Write the worker response code in `workers/request/index.js` .



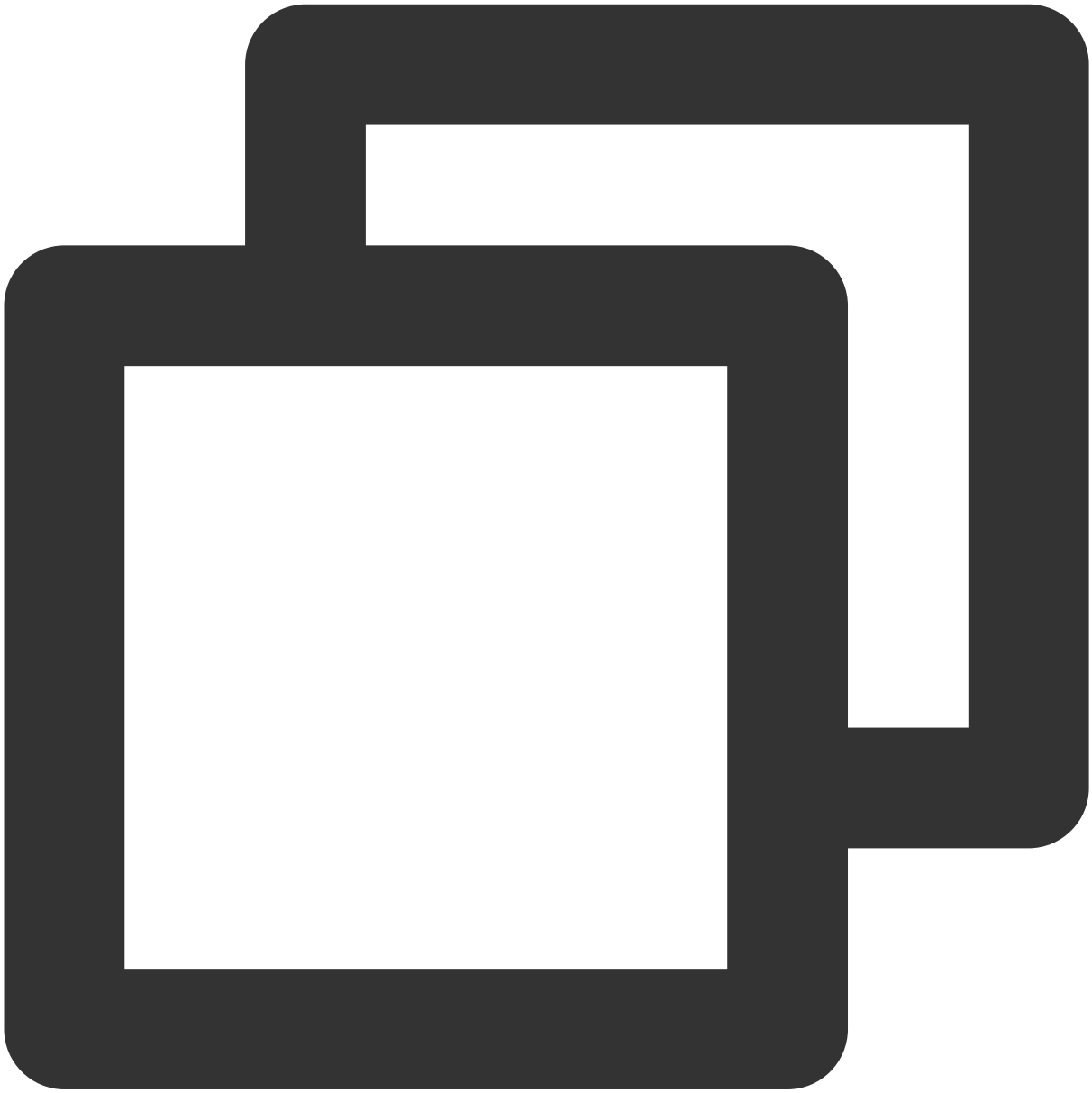
```
const utils = require('./utils')

// In the worker thread execution context, a worker object is globally exposed and
worker.onMessage(function (res) {
  console.log(res)
})
```

©2013-2022 Tencent Cloud. All rights reserved.

Initialize Worker in Main Thread

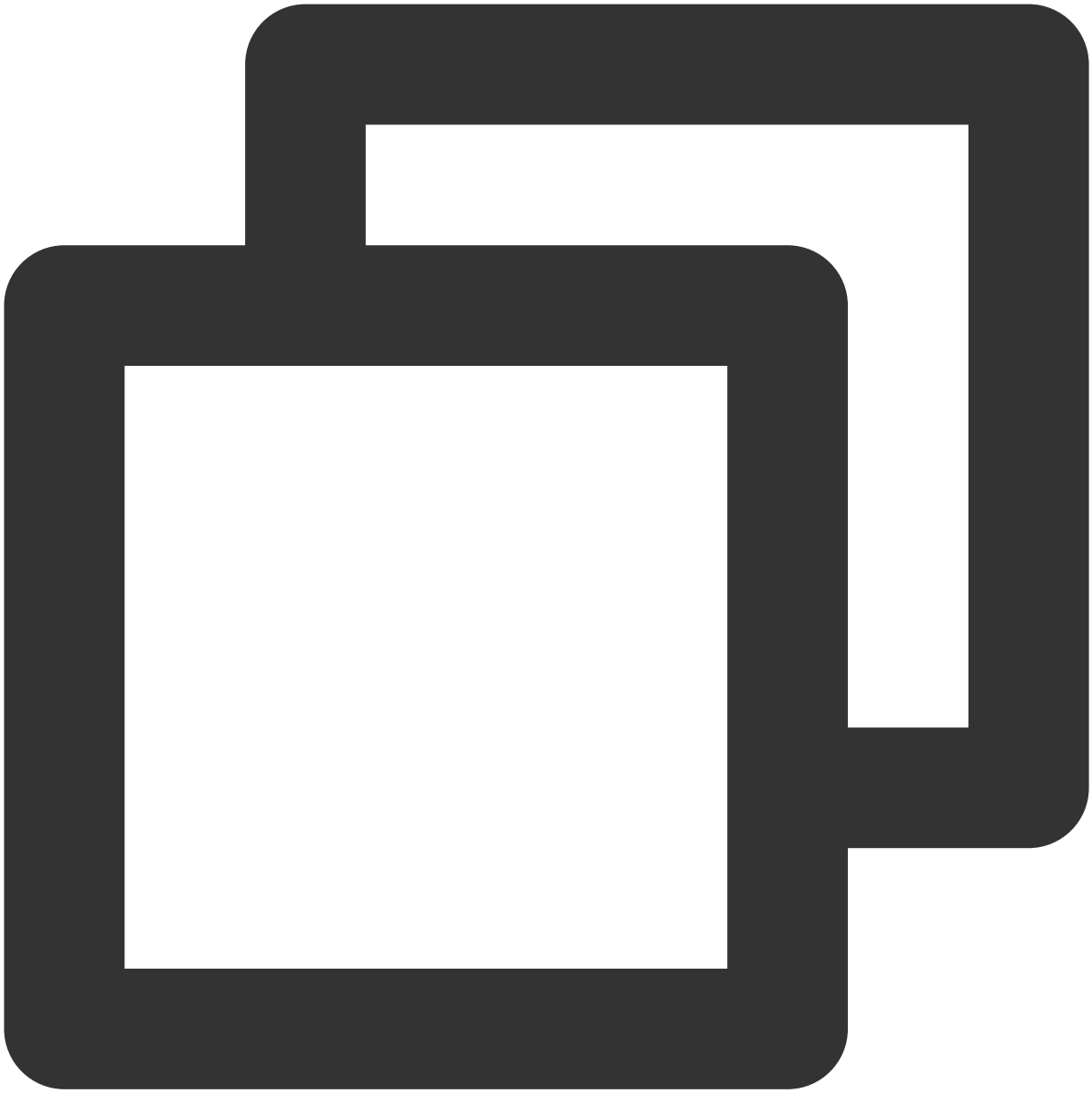
Initialize worker in app.js of the main thread.



```
const worker = wx.createWorker('workers/request/index.js') // The filename specifie
```

Main Thread Sends a Message to Worker

©2013-2022 Tencent Cloud. All rights reserved.



```
worker.postMessage({
  msg: 'hello worker'
})
```

Notes

The maximum concurrency of worker is one. Use Worker.terminate() to terminate the current worker before creating the next one;

©2013-2022 Tencent Cloud. All rights reserved.

The code in the worker can only require files in the specified worker path. It cannot reference other paths;
The worker entry files are specified when calling wx.createWorker(). Developers can dynamically specify the worker entry files;
wx APIs are not supported in workers;
Workers cannot send messages to each other.

©2013-2022 Tencent Cloud. All rights reserved.

DarkMode Adaptation Guide

Last updated：2024-07-12 17:18:38

1. Enable DarkMode

Configure darkmode as true in `app.json`, that means the current mini program has been adapted to DarkMode, all the basic components will display different default styles according to the system theme, and the navigation bar and tab bar will be automatically switched according to the following configuration.

2. Related Configuration

When `app.json` is configured with `darkmode` as `true`, some of the configuration items of mini program can be configured in the form of variables, defining the colour or icon under different themes in the variable configuration file, as follows:
1. Configure `themeLocation` in app.json to specify the path of variable configuration file theme.json, for example: add `theme.json` in the root directory, you need to configure `"themeLocation": "theme.json"`;
2. Define the relevant variables in theme.json;
3. Referencing variables in app.json starting with `@`;
Properties configured via variables are supported.
Globally configured window properties and properties under page configuration
navigationBarBackgroundColor;
navigationBarTextStyle;
backgroundColor;
backgroundTextStyle;
backgroundColourTop;
backgroundColorBottom.
Global configuration of window.tabBar properties
colour;
selectedColor;
backgroundColor;
borderStyle;
list;
iconPath;
selectedIconPath.

©2013-2022 Tencent Cloud. All rights reserved.

2.1 Variable configuration file theme.json

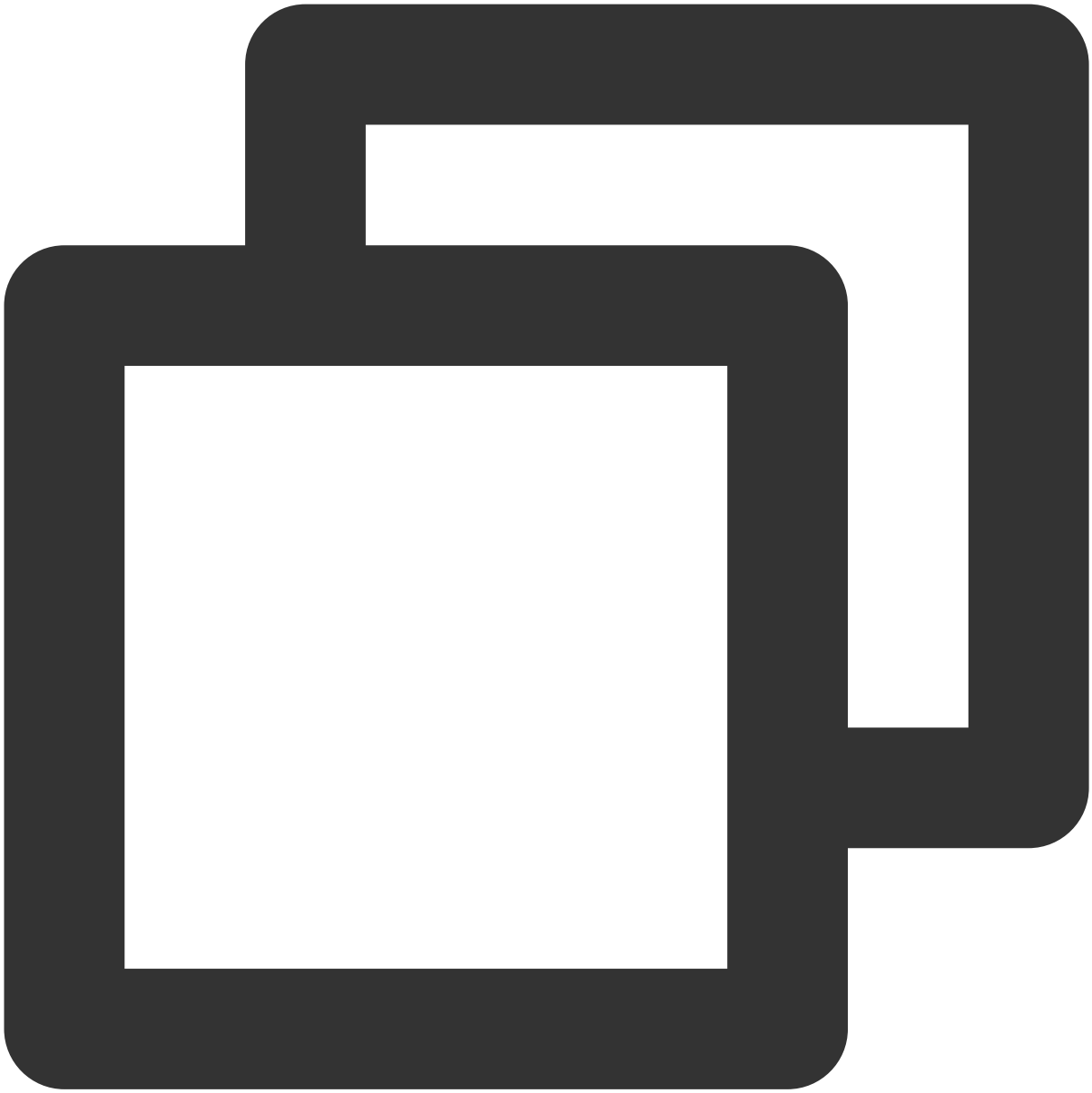
`theme.json` is used to define the variables related to the colour theme, you need to configure the path of `theme.json` in `themeLocation` first, otherwise you can't read the variable configuration.

The configuration file must contain the following properties:

Attributes	Type	Required	Descripción
light	object	yes	Definition of the variable in light mode
dark	object	yes	Definition of the variable in dark mode

Both `light` and `dark` can define variable names and values as key: value, for example:

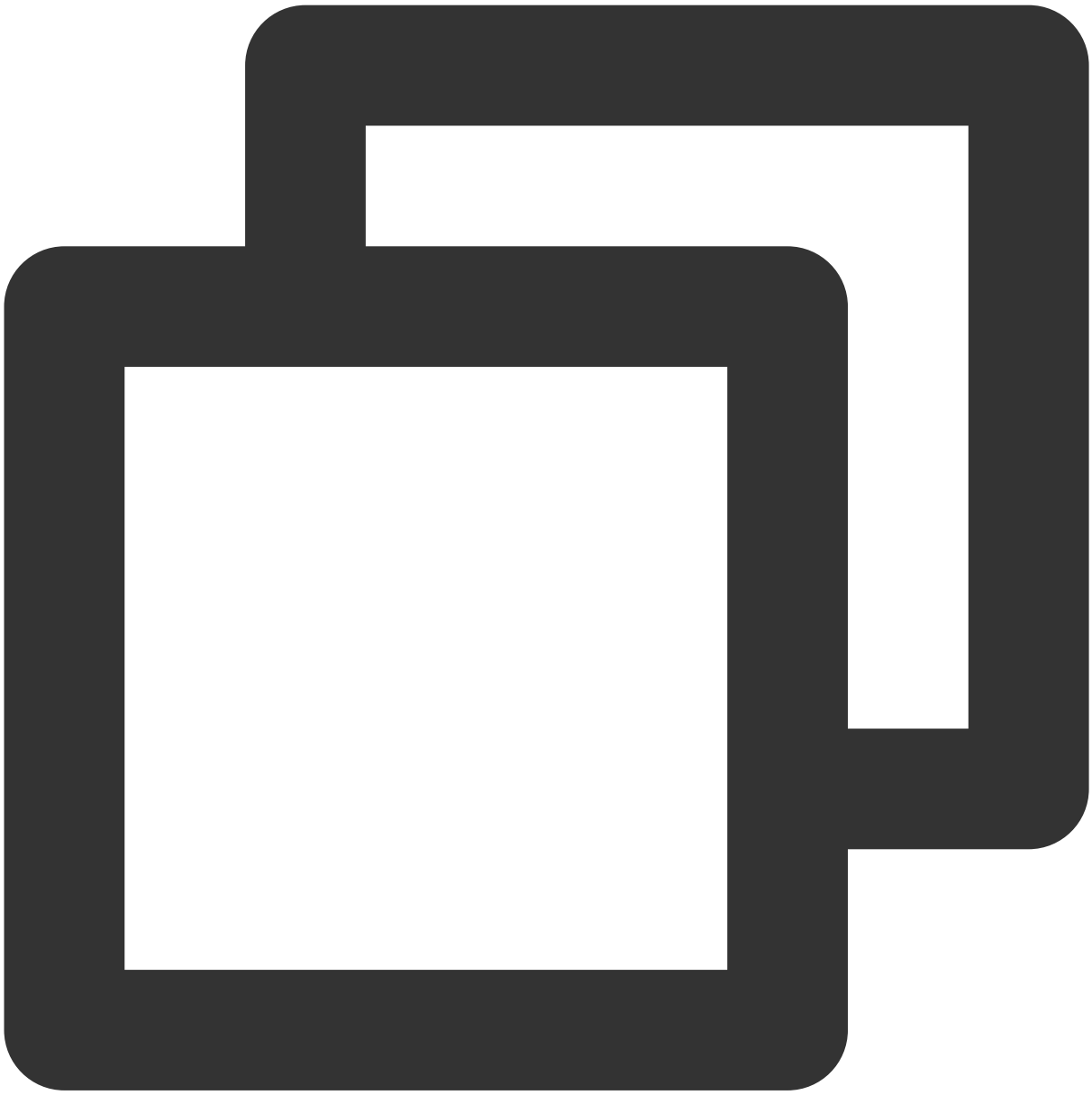
©2013-2022 Tencent Cloud. All rights reserved.



```
{
  "light": {
    "navBgColor": "#f6f6f6",
    "navTxtStyle": "black"
  },
  "dark": {
    "navBgColor": "#191919",
    "navTxtStyle": "white"
  }
}
```

©2013-2022 Tencent Cloud. All rights reserved.

After completing the definition, it can be referenced in the global configuration or in the relevant properties of the page configuration starting with `@`, for example:



```
// Global Configuration
{
  "window": {
    "navigationBarBackgroundColor": "@navBgColor",
    "navigationBarTextStyle": "@navTxtStyle"
  }
}
```

©2013-2022 Tencent Cloud. All rights reserved.

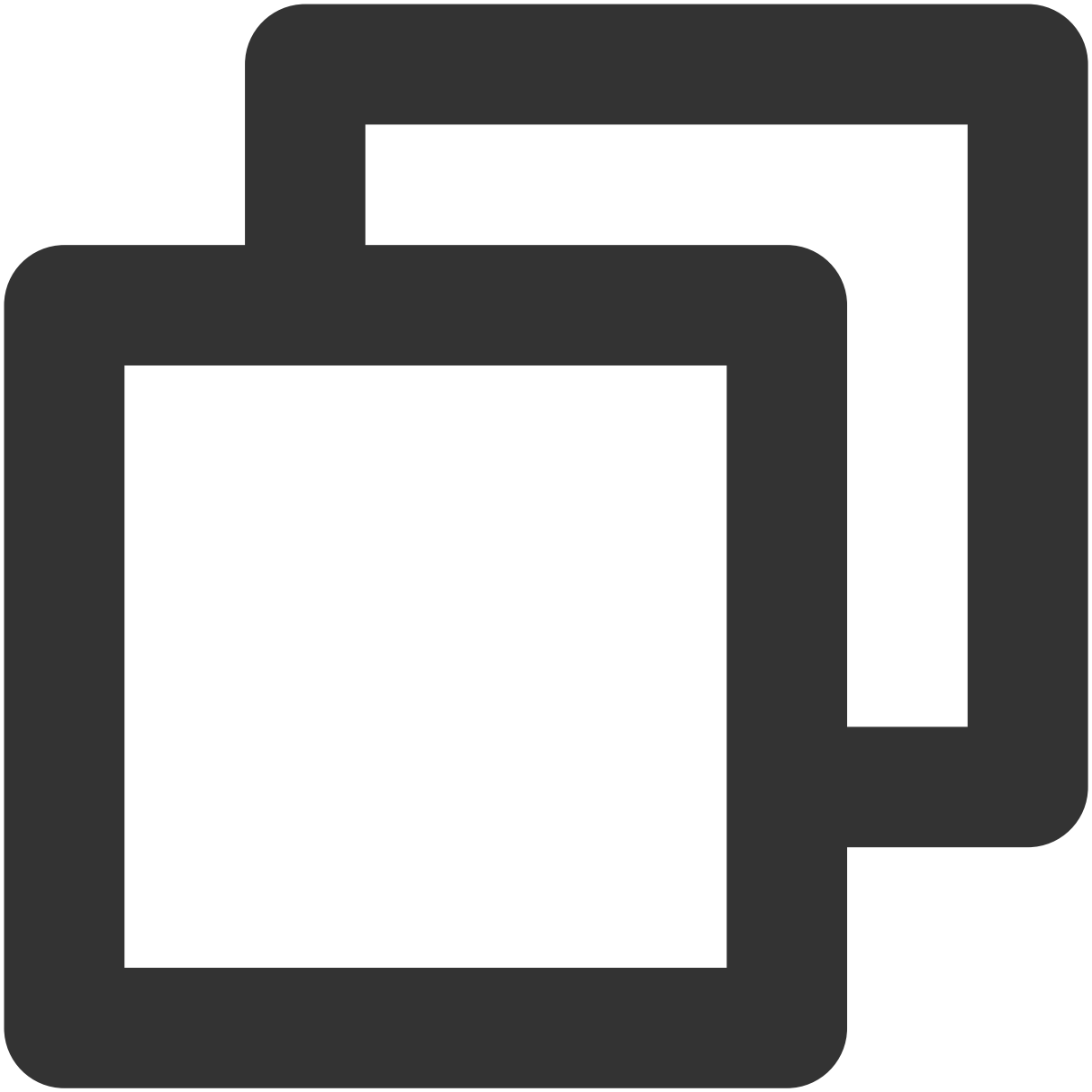
```
// Page Configuration
{
  "navigationBarBackgroundColor": "@navBgColor",
  "navigationBarTextStyle": "@navTxtStyle"
}
```

After the configuration is completed, the mini program framework will automatically display the colour under the corresponding theme for the mini program according to the system theme.

2.2 Configuration Example

app.json (the example omits configuration items other than those related to the theme)

©2013-2022 Tencent Cloud. All rights reserved.



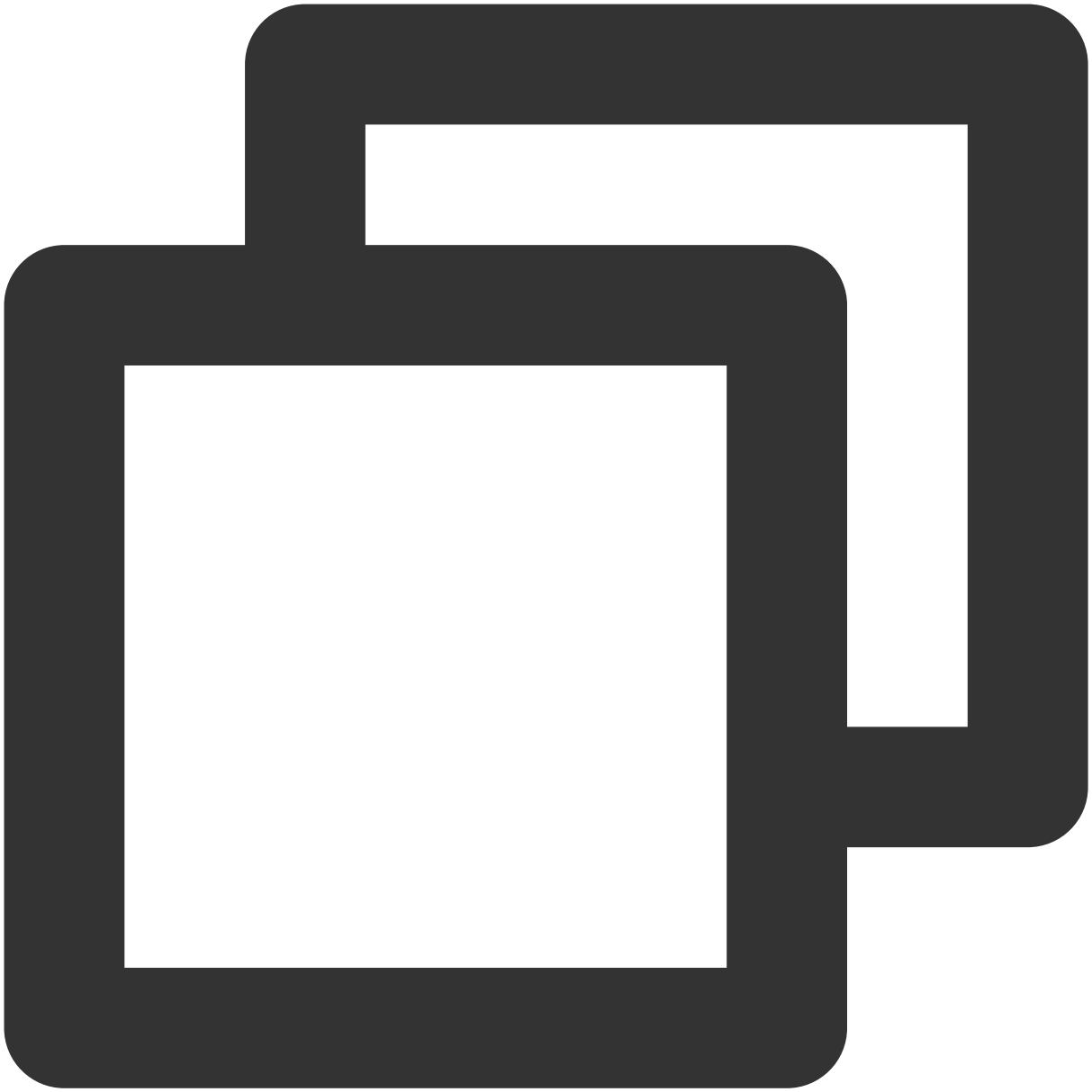
```
{
  "window": {
      "navigationBarBackgroundColor": "@navBgColor",
      "navigationBarTextStyle": "@navTxtStyle",
      "backgroundColor": "@bgColor",
      "backgroundTextStyle": "@bgTxtStyle",
      "backgroundColorTop": "@bgColorTop",
      "backgroundColorBottom": "@bgColorBottom"
  },
  "tabBar": {
      "color": "@tabFontColor",
```

©2013-2022 Tencent Cloud. All rights reserved.

```
        "selectedColor": "@tabSelectedColor",
        "backgroundColor": "@tabBgColor",
        "borderStyle": "@tabBorderStyle",
        "list": [{
            "iconPath": "@iconPath1",
            "selectedIconPath": "@selectedIconPath1"
        }, {
            "iconPath": "@iconPath2",
            "selectedIconPath": "@selectedIconPath2"
        }]
    }
}
```

theme.json

©2013-2022 Tencent Cloud. All rights reserved.



```
{
  "light": {
      "navBgColor": "#f6f6f6",
      "navTxtStyle": "black",
      "bgColor": "#ffffff",
      "bgTxtStyle": "light",
      "bgColorTop": "#eeeeee",
      "bgColorBottom": "#efefef",
      "tabFontColor": "#000000",
      "tabSelectedColor": "#3cc51f",
      "tabBgColor": "#ffffff",
```

©2013-2022 Tencent Cloud. All rights reserved.

```
        "tabBorderStyle": "black",
        "iconPath1": "image/icon1_light.png",
        "selectedIconPath1": "image/selected_icon1_light.png",
        "iconPath2": "image/icon2_light.png",
        "selectedIconPath2": "image/selected_icon2_light.png",
    },
    "dark": {
        "navBgColor": "#191919",
        "navTxtStyle": "white",
        "bgColor": "#1f1f1f",
        "bgTxtStyle": "dark",
        "bgColorTop": "#191919",
        "bgColorBottom": "#1f1f1f",
        "tabFontColor": "#ffffff",
        "tabSelectedColor": "#51a937",
        "tabBgColor": "#191919",
        "tabBorderStyle": "white",
        "iconPath1": "image/icon1_dark.png",
        "selectedIconPath1": "image/selected_icon1_dark.png",
        "iconPath2": "image/icon2_dark.png",
        "selectedIconPath2": "image/selected_icon2_dark.png",
    }
}
```

3. Get current system theme

If app.json declares `"darkmode": true`, `wx.getSystemInfo` or `wx.getSystemInfoSync` will return a `theme` attribute with a value of light or dark.

If app.json does not declare `"darkmode": true`, the theme property is not available (i.e. `theme` is `undefined`).

4. Listening for theme switching events

There are 2 ways to do this:

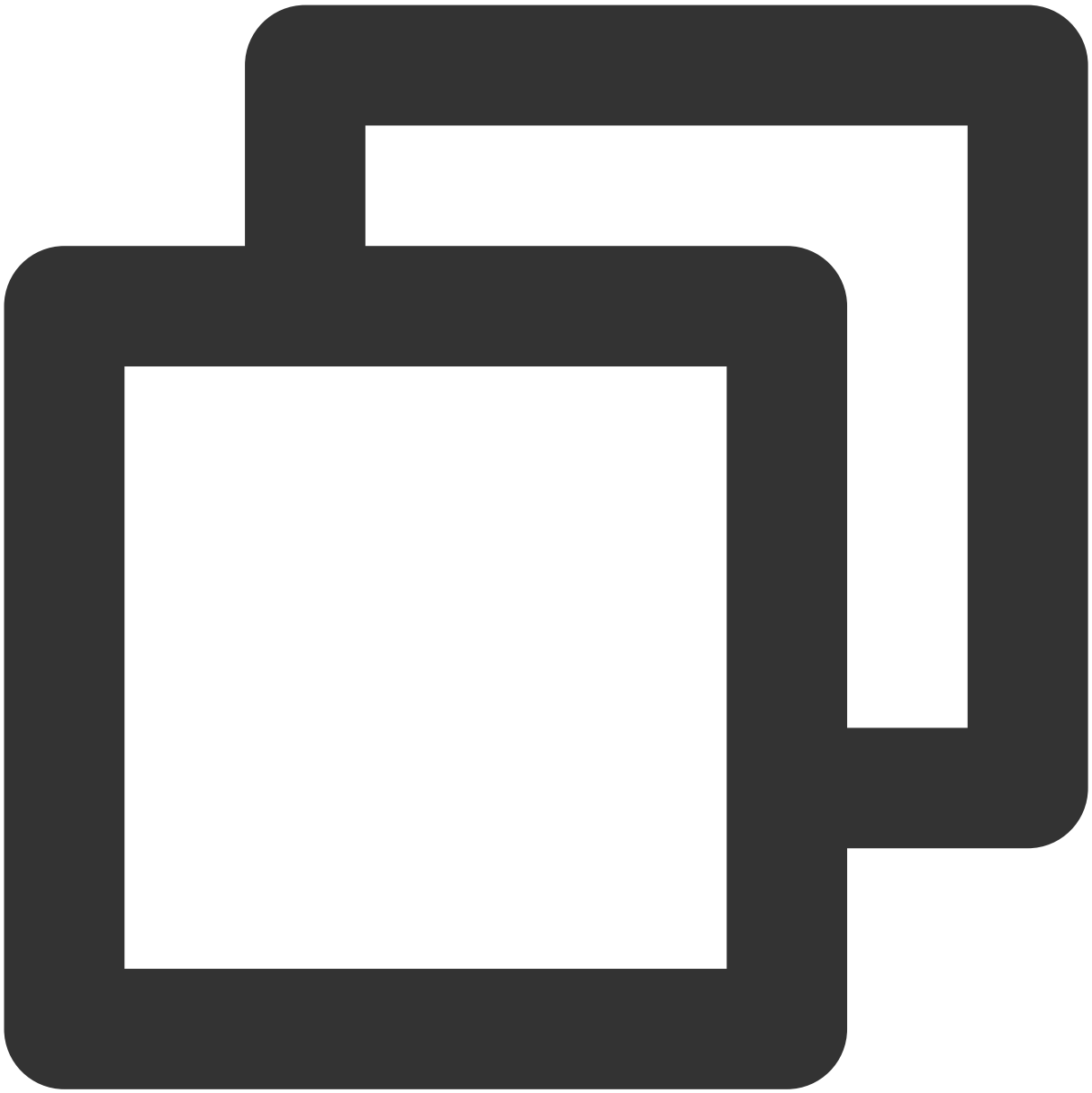
Pass onThemeChange callback method in App(), the callback will be triggered when the theme is switched;

Listen to theme change by wx.onThemeChange, wx.offThemeChange cancels the listening.

5. WXSS Adaptation

©2013-2022 Tencent Cloud. All rights reserved.

WXSS supports adapting different themes via prefers-color-scheme media query, which is the same as the web adapting method, for example:



```
/* General style begin */
.some-background {
  background: white;
}
.some-text {
  color: black;
}
/* General style end */
```

©2013-2022 Tencent Cloud. All rights reserved.

```
@media (prefers-color-scheme: dark) {
  /* Style start under DarkMode */
  .some-background {
      background: #1b1b1b;
  }
  .some-text {
      color: #ffffff;
  }
  /* Style end under DarkMode */
}
```

6. Tips

1. Note that the media query in WXSS is not affected by the darkmode switch in app.json. As long as the client supports DarkMode, the media query will take effect when the system is switched to DarkMode, no matter "darkmode": true is configured or not;
2. Theme switch event needs to be triggered when "darkmode": true is configured.

©2013-2022 Tencent Cloud. All rights reserved.

Framework
Mini program configuration
Project Configuration

Last updated：2024-07-12 18:29:49

The `project.config.json` file in the project root directory allows you to configure the project.
The `project.config.json` file is mainly used in mini programs to configure the behaviour and characteristics of the mini program development tool. This file allows developers to adjust some of the default settings of the development tool to better suit the development process and environment.

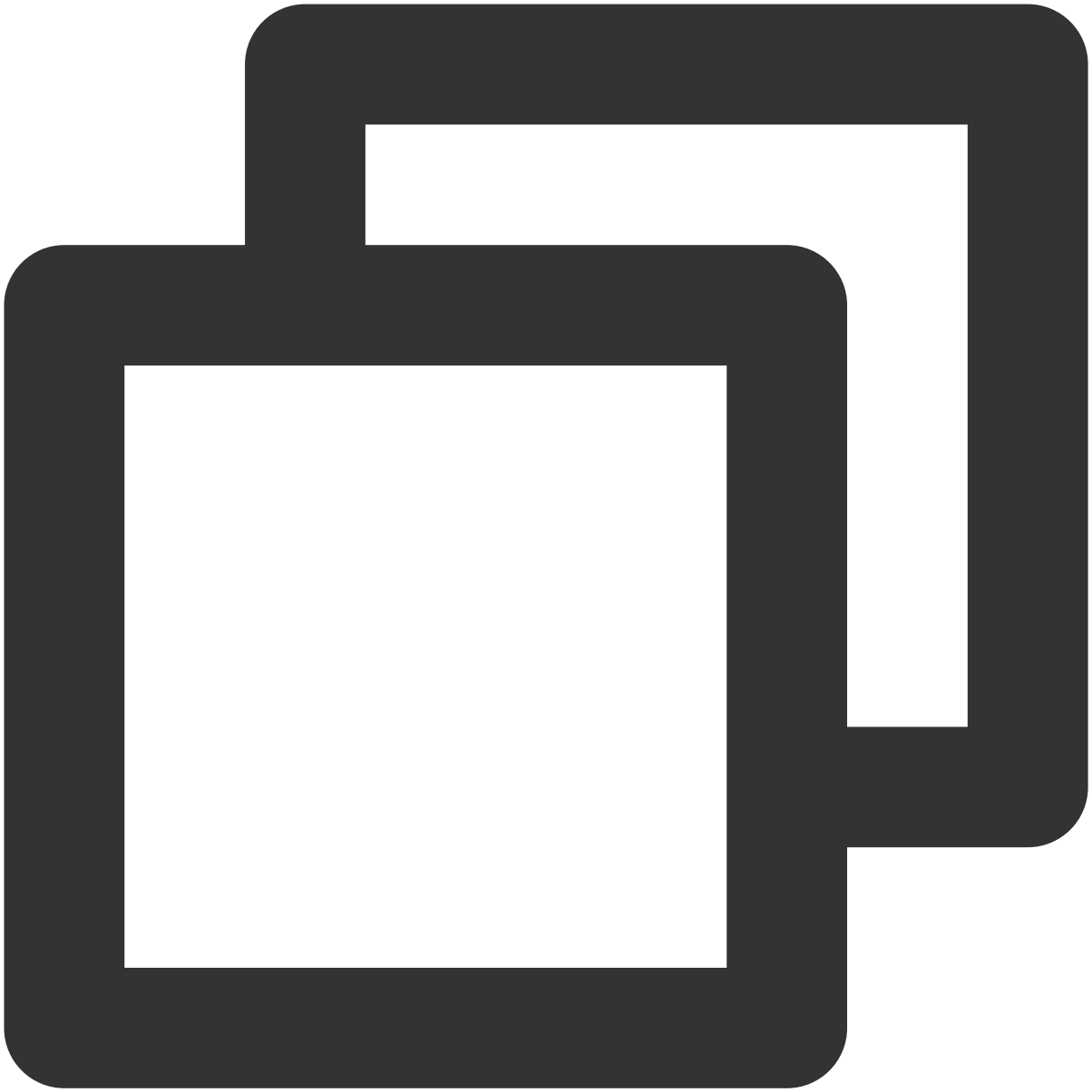
configuration item

attribute	type	describe
miniprogramRoot	Path String	Specify the directory of the source code of the mini program (must be a relative path).
compileType	String	Compile type, default is miniprogram.
TCMPPappid	String	The appid of the project.
projectname	String	Project name
setting	Object	Project setting
libVersion	String	Base library version
condition	Object	Conditional Compilation Configuration
mockConfig	Object	IDE mockAPI Configuration

miniprogramRoot

The directory of the mini program source code (needs to be a relative path).

©2013-2022 Tencent Cloud. All rights reserved.

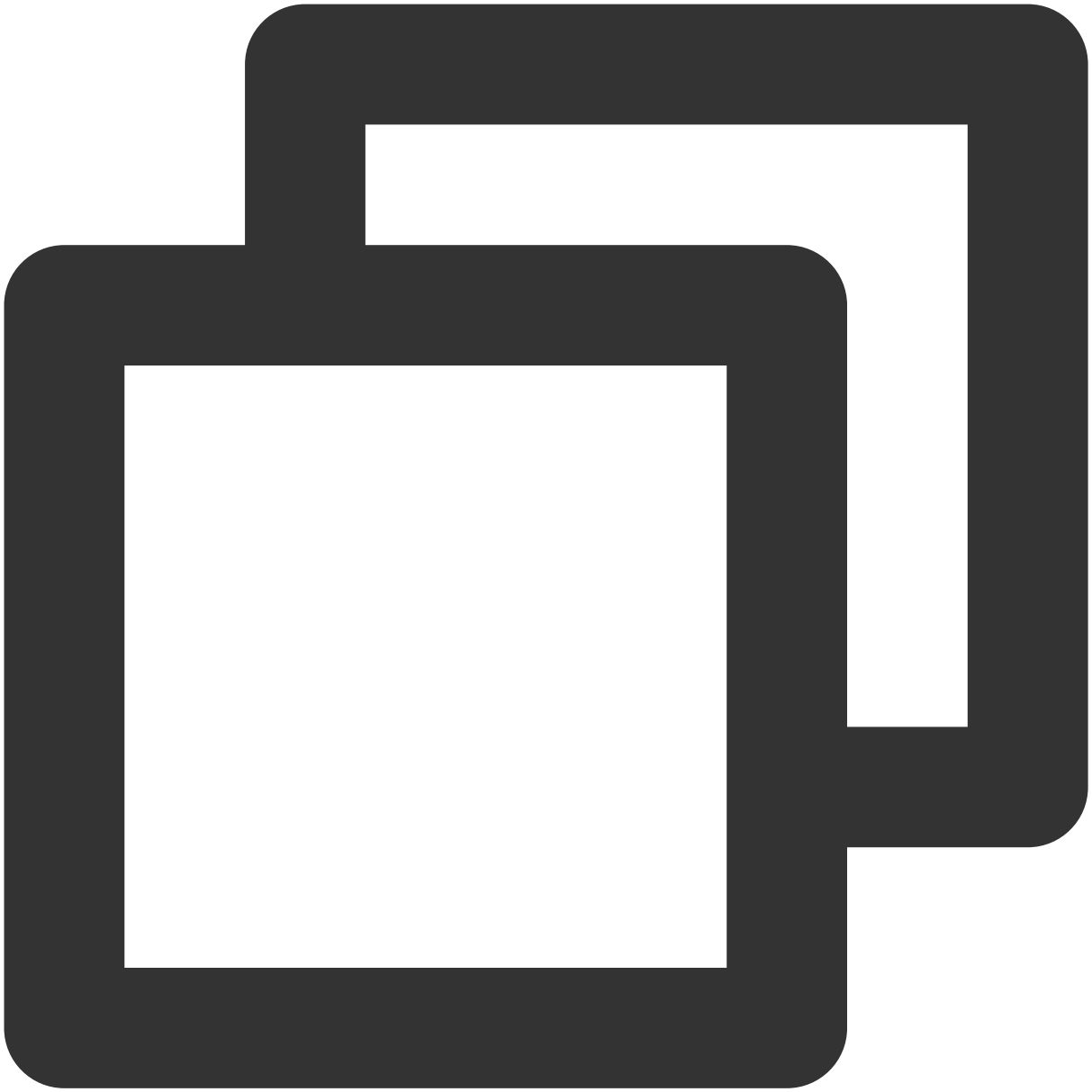


```
{
  "miniprogramRoot": "./src"
}
```

compileType

Compile type, defaults to miniprogram.

©2013-2022 Tencent Cloud. All rights reserved.

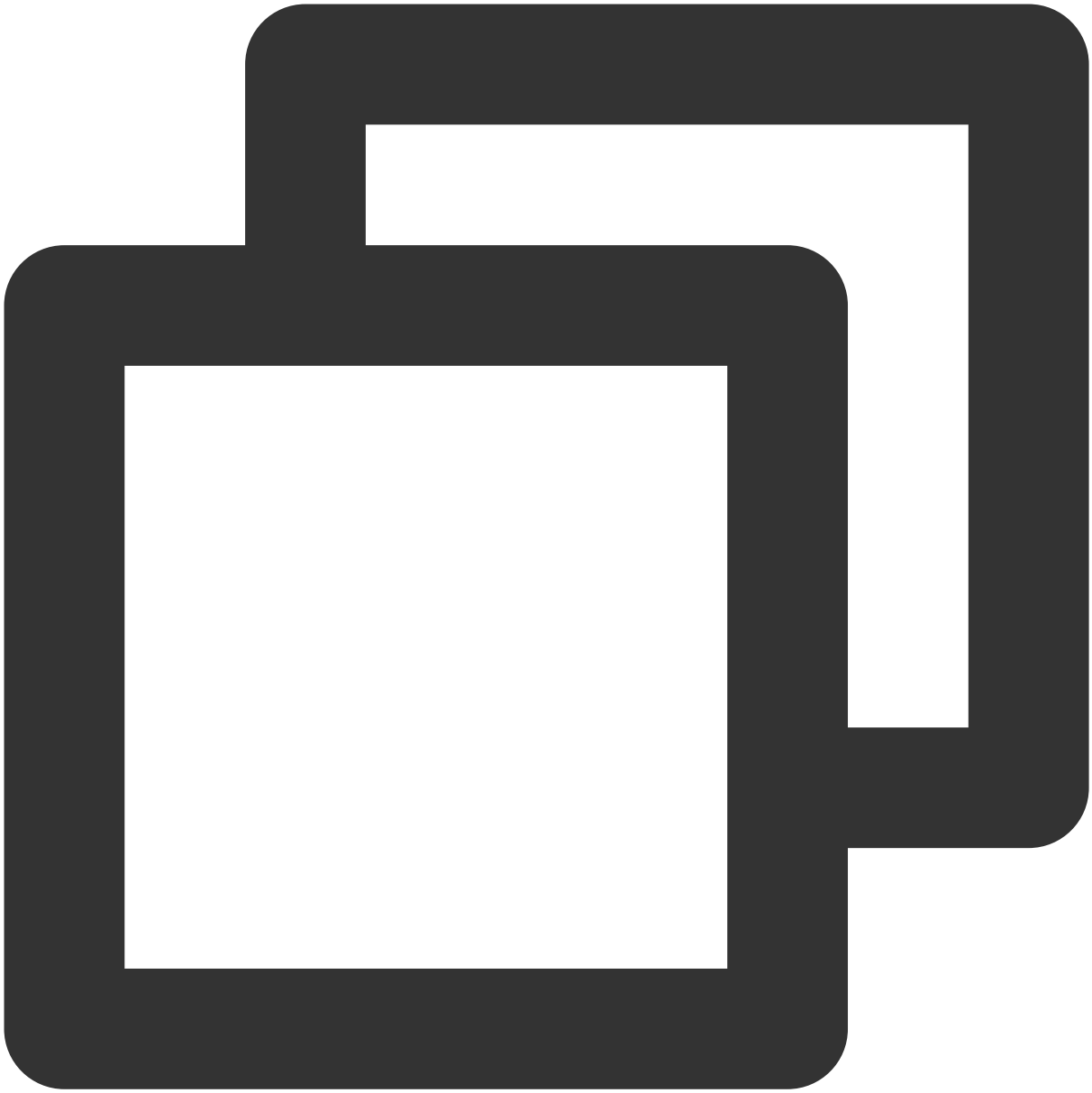


```
{
  "compileType": "miniprogram"
}
```

TCMPPappid

The appId corresponding to the mini program on the TCMPP console.

©2013-2022 Tencent Cloud. All rights reserved.

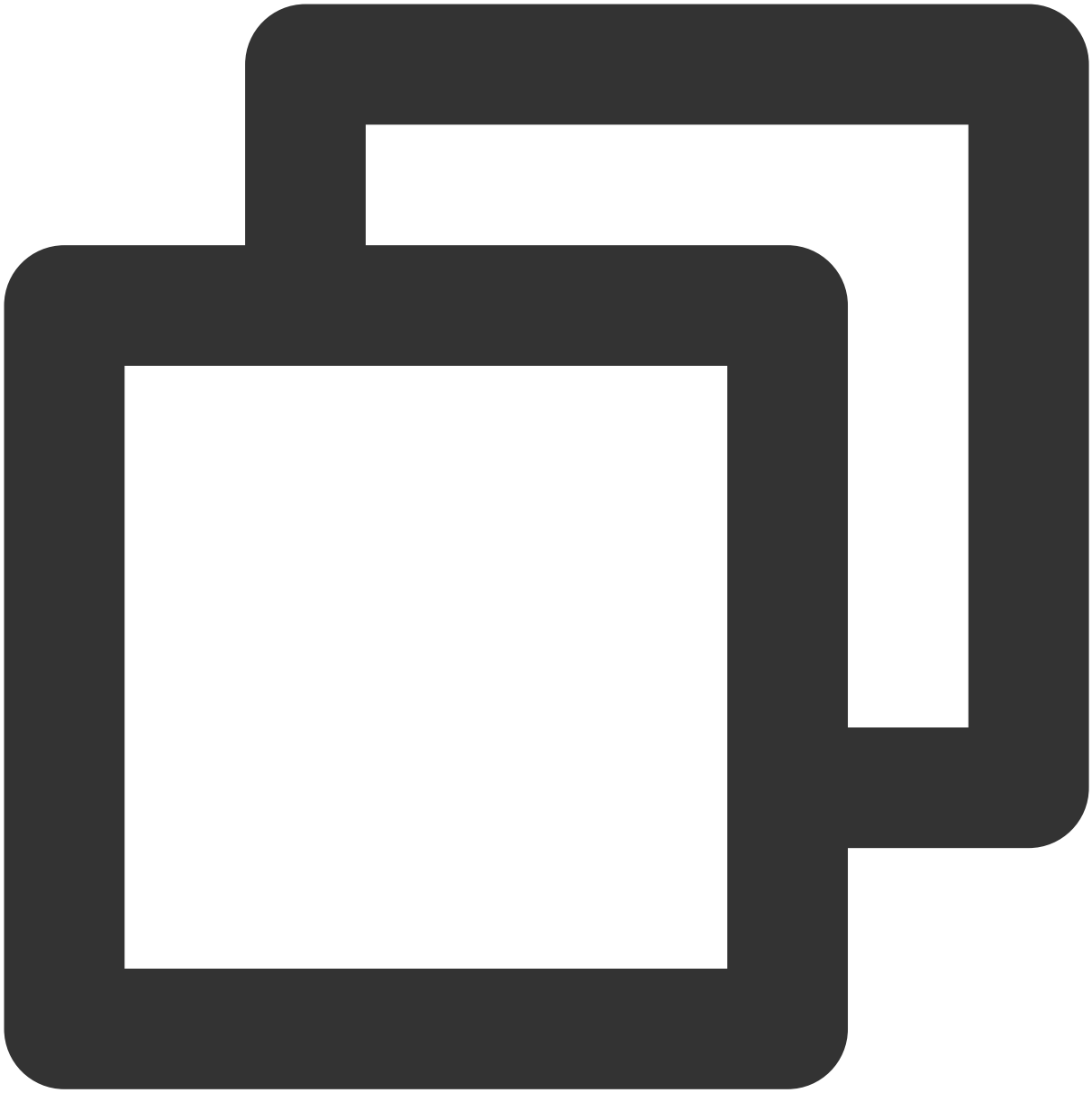


```
{
  "TCMPPappid": "mpz5g4hpzd5p3k2l"
}
```

projectname

Project Title.

©2013-2022 Tencent Cloud. All rights reserved.



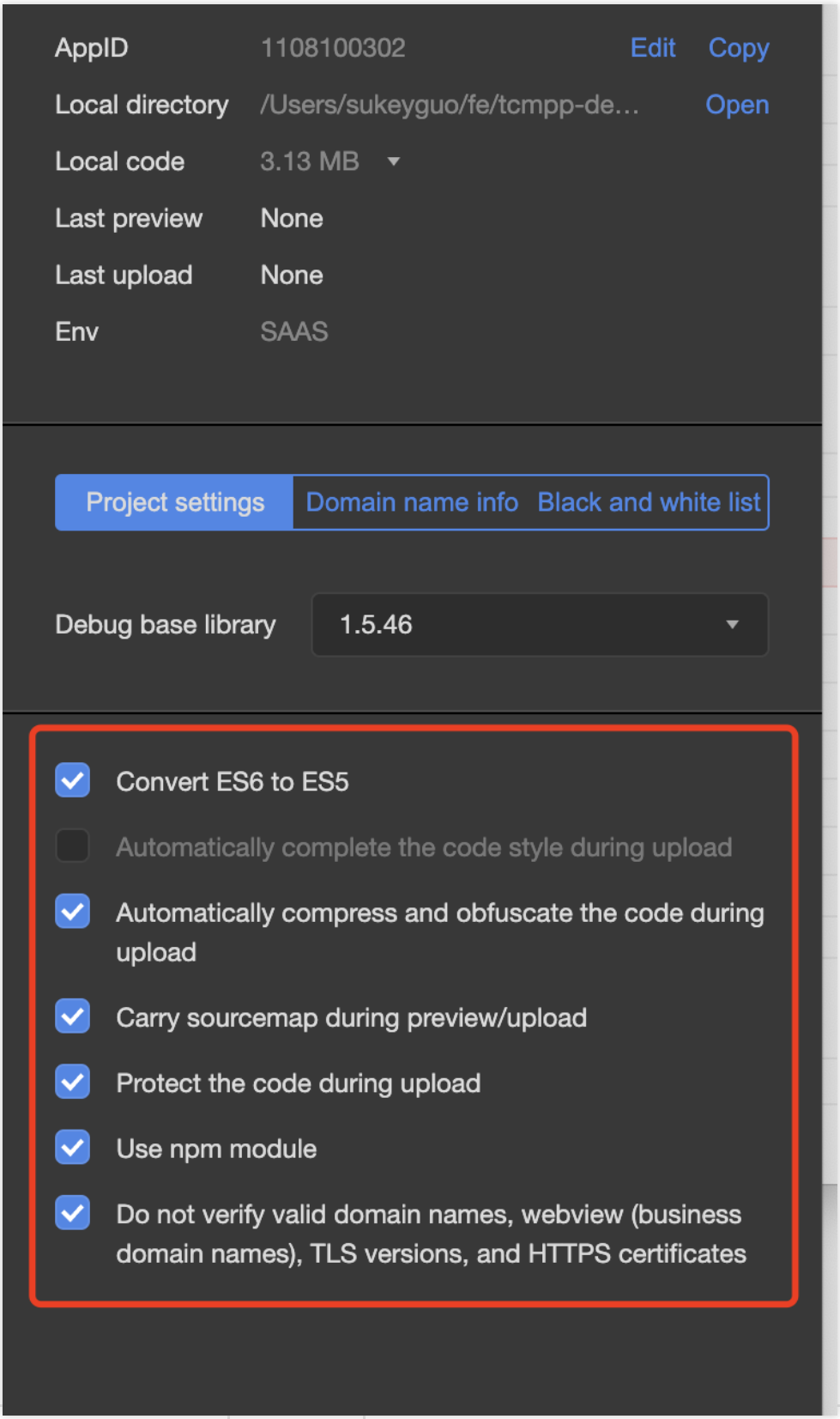
```
{
  "projectname": "main-demo"
}
```

setting

You can specify the following settings for the compilation settings of the project.

The following fields can be found in Developer Tools > Details > Local Settings.

©2013-2022 Tencent Cloud. All rights reserved.



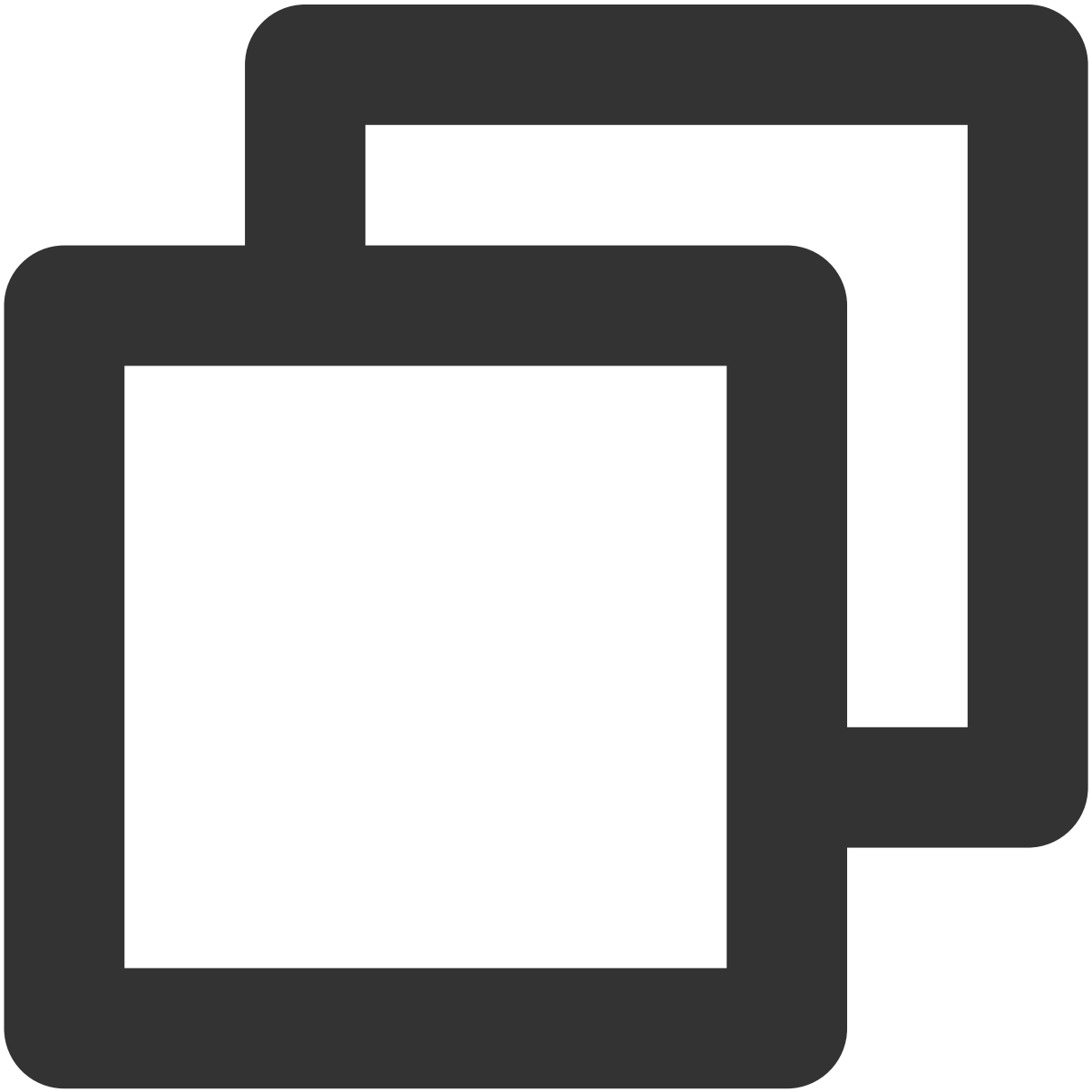
attribute	type	describe
es6	Boolean	Whether to enable es6 to es5 conversion
postcss	Boolean	Whether to auto-complete the style when uploading code

©2013-2022 Tencent Cloud. All rights reserved.

minified	Boolean	If or not the script file will be automatically compressed when uploading code.
uglifyFileName	Boolean	Protect code when uploading
urlCheck	Boolean	Whether to check secure domain name and TLS version and HTTPS certificate

libVersion

The version number of the base library currently in use.

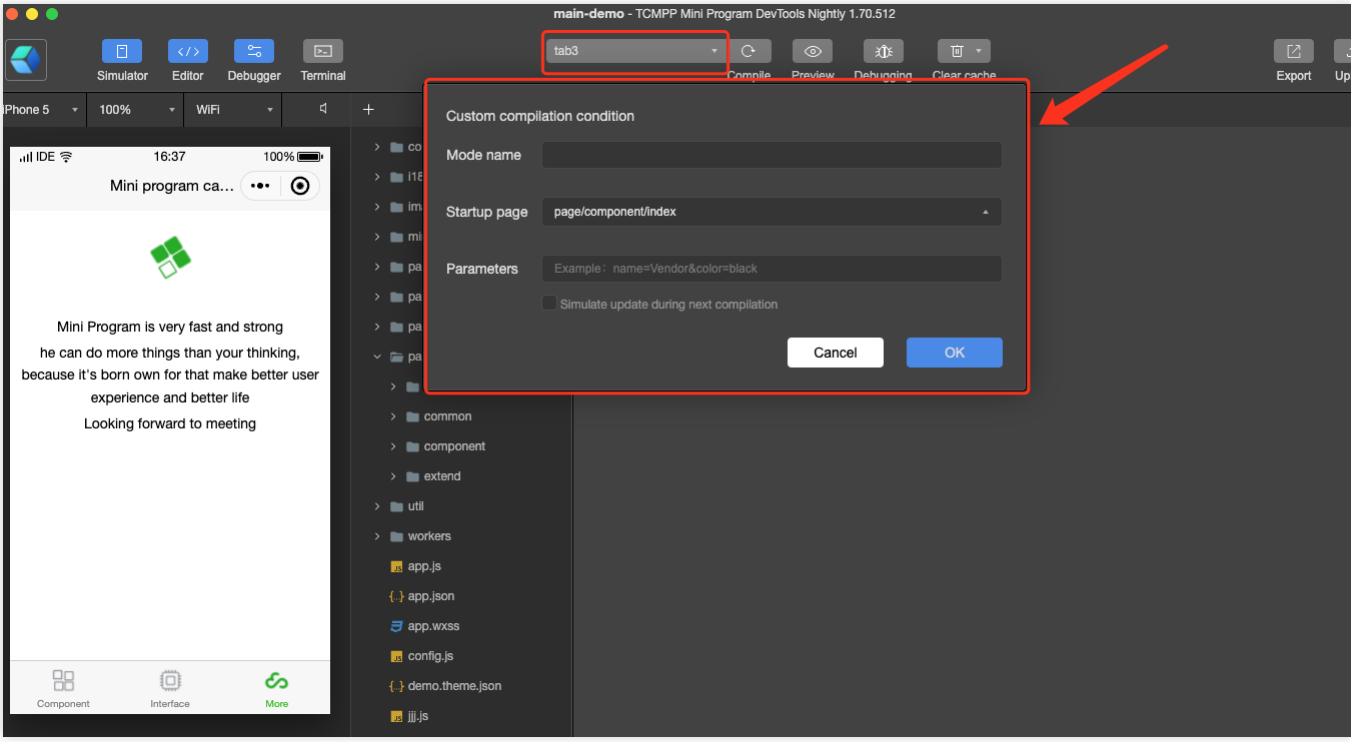


©2013-2022 Tencent Cloud. All rights reserved.

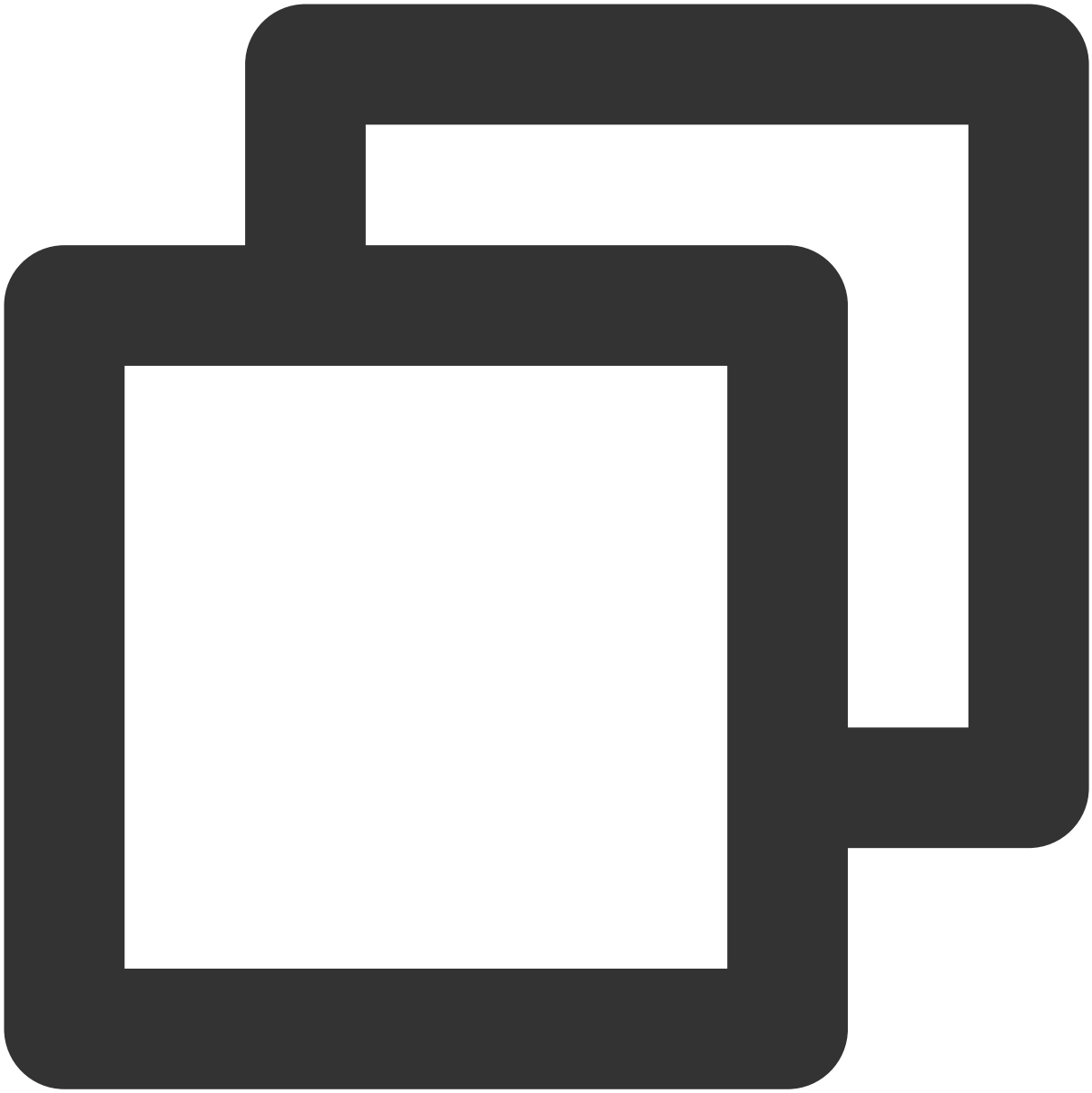
```
{
  "libVersion": "1.5.0"
}
```

condition

Corresponds to the conditional compilation information set on the IDE.



©2013-2022 Tencent Cloud. All rights reserved.



```
{
  "condition": {
    "miniprogram": {
      "current": 0,
      "list": [
        {
          "id": 0,
          "name": "tab3",
          "pathName": "page/extend/index",
          "query": "",
          "scene": null
```

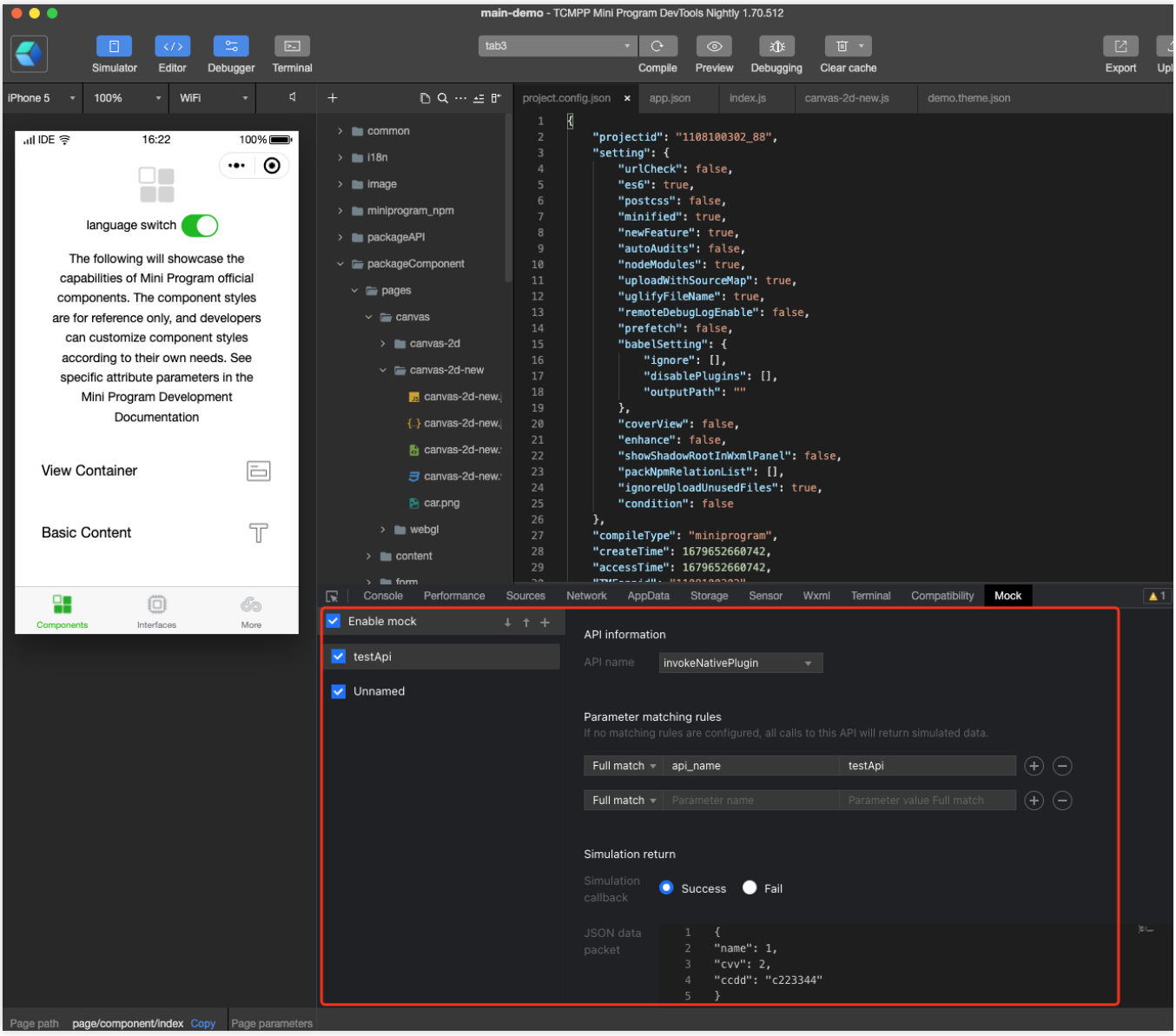
©2013-2022 Tencent Cloud. All rights reserved.

```
          },
          {
            "id": -1,
            "name": "tab4",
            "pathName": "page/extend/index",
            "query": "name=1",
            "scene": null
          }
        ]
      }
    }
  }
```

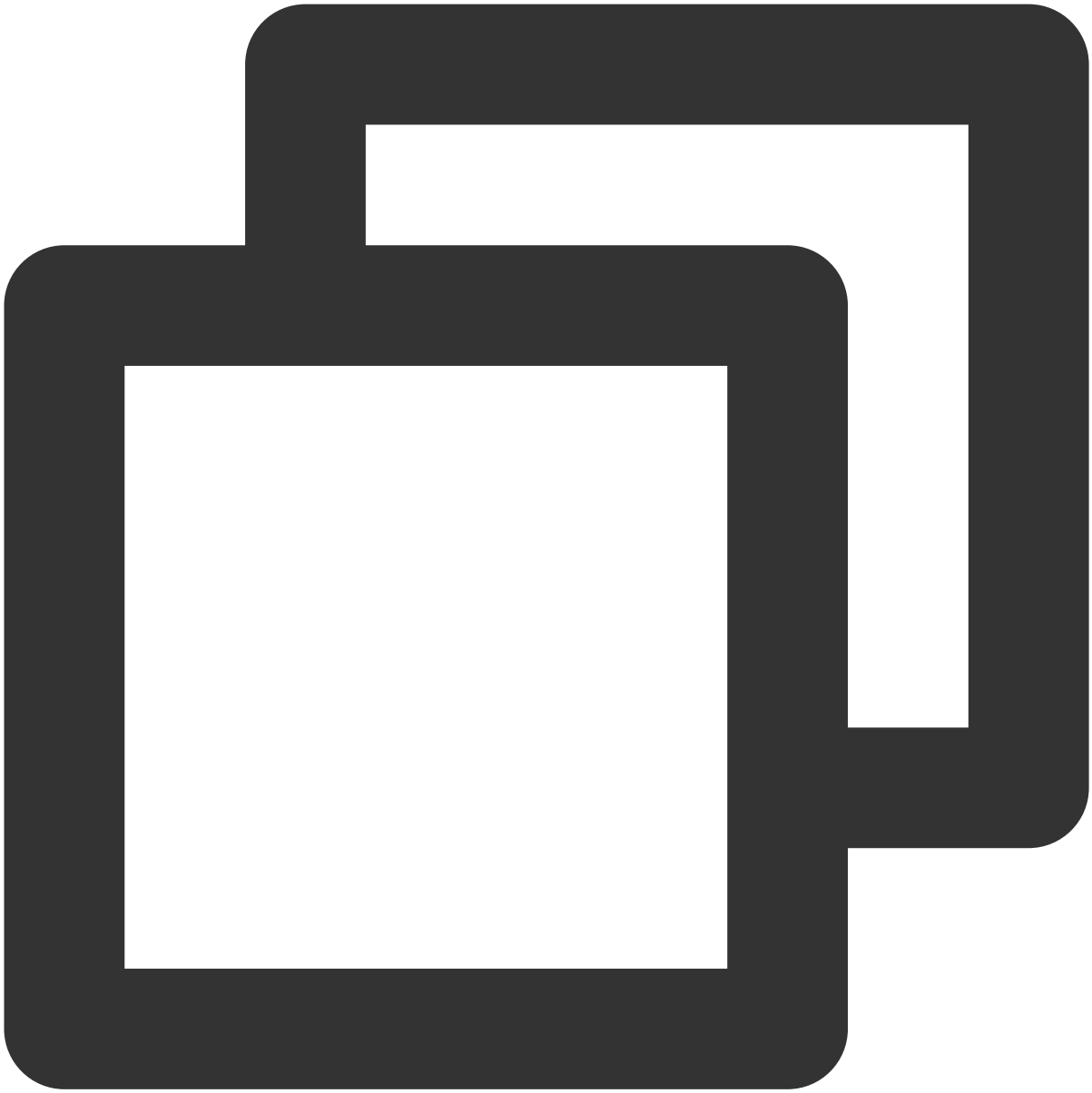
mockConfig

Configuration information that is automatically generated when the corresponding project mocks the API in the IDE debug panel.

©2013-2022 Tencent Cloud. All rights reserved.



©2013-2022 Tencent Cloud. All rights reserved.




```
{
  "globalOpen": true,
  "rules": [
    {
      "ruleName": "testApi",
      "ruleId": "48b83a70-84fd-400d-b53d-097518cfa515",
      "apiName": "invokeNativePlugin",
      "enable": true,
      "mockType": "API",
      "returnType": "succ",
      "resStr": "{\\n\\"name\\": 1,\\n\\"cvv\\": 2,\\n\\"ccdd\\": \\"c223344\\"\\n}
```


©2013-2022 Tencent Cloud. All rights reserved.

```
      "filterList": [
        {
          "propName": "api_name",
          "propRegString": "testApi",
          "matchType": "string"
        },
        {
          "propName": "",
          "propRegString": "",
          "matchType": "string"
        }
      ],
      "verifyFail": false
    }
  ]
}
```

©2013-2022 Tencent Cloud. All rights reserved.

Global Configuration

Last updated：2024-07-12 18:34:32

Mini Program root directory app.json The file is used for global configuration of the WeChat Mini Program. The file content is a JSON Object has the following properties.

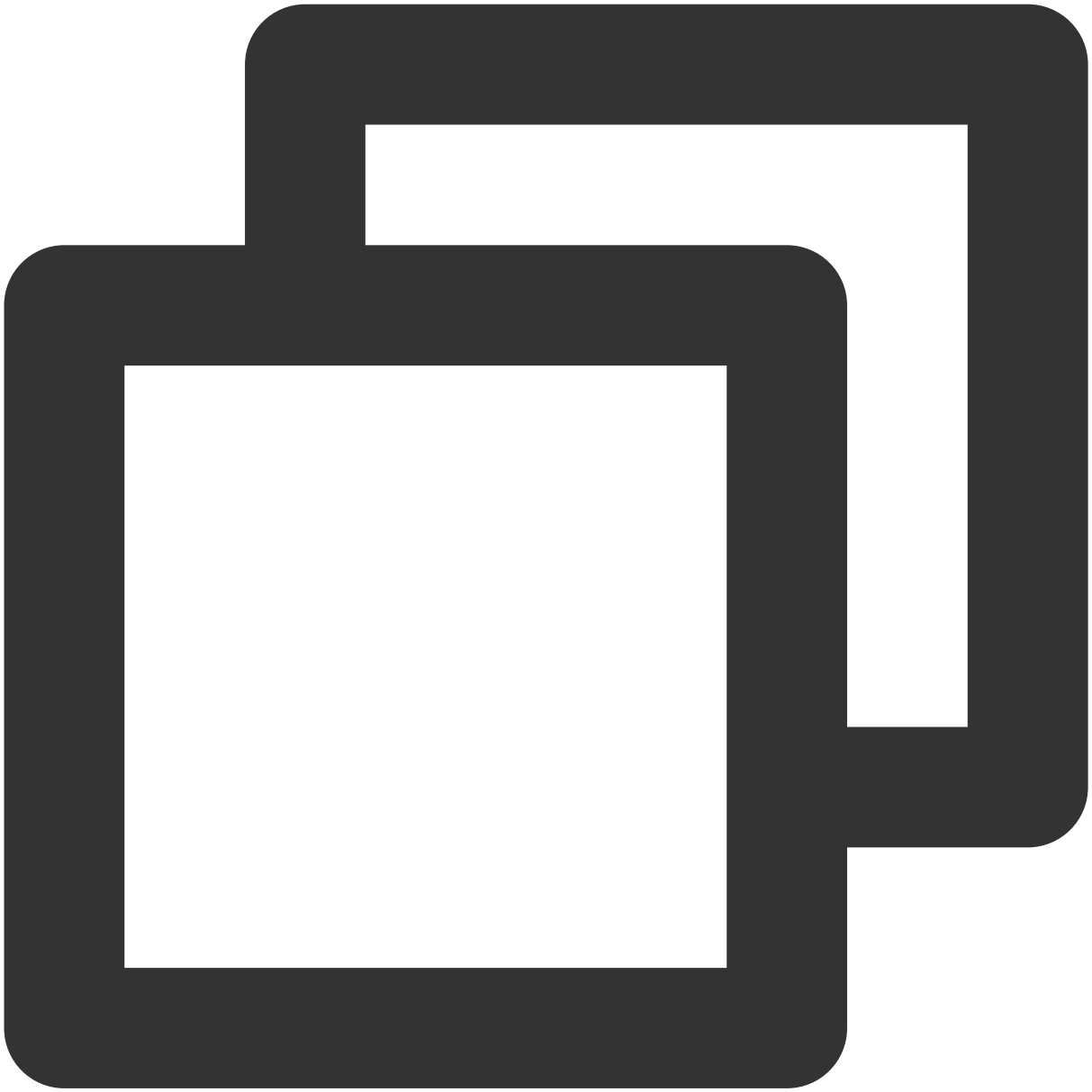
Configuration item

attribute	type	required	describe
entryPagePath	string	no	Mini Program default start home page
pages	string[]	yes	Page Path List
window	Object	no	Global default window representation
tabBar	Object	no	BOTTOM tab Column performance
debug	boolean	no	Whether or not to open debug Mode, off by default
subpackages	Object[]	no	Subcontract structure configuration
workers	string	no	Worker Directory for code placement
navigateToMiniProgramAppIdList	string[]	no	The list of mini programs to jump to, see wx.navigateToMiniProgram for details.
preloadRule	Object	no	Subcontract Predownload Rules
usingComponents	Object	no	overall situation Custom Components configuration
permission	Object	no	Small Program Interface Permissions Related Settings
darkmode	boolean	no	Mini Program support DarkMode
themeLocation	string	no	Specify theme.json And dark mode true is required

entryPagePath

©2013-2022 Tencent Cloud. All rights reserved.

Specify the default start path (home page) of the Mini Program, common scenario is from the WeChat chat list page drop-down start, Mini Program list start. If not, the default is pages The first item in the list. Page path parameters are not supported.

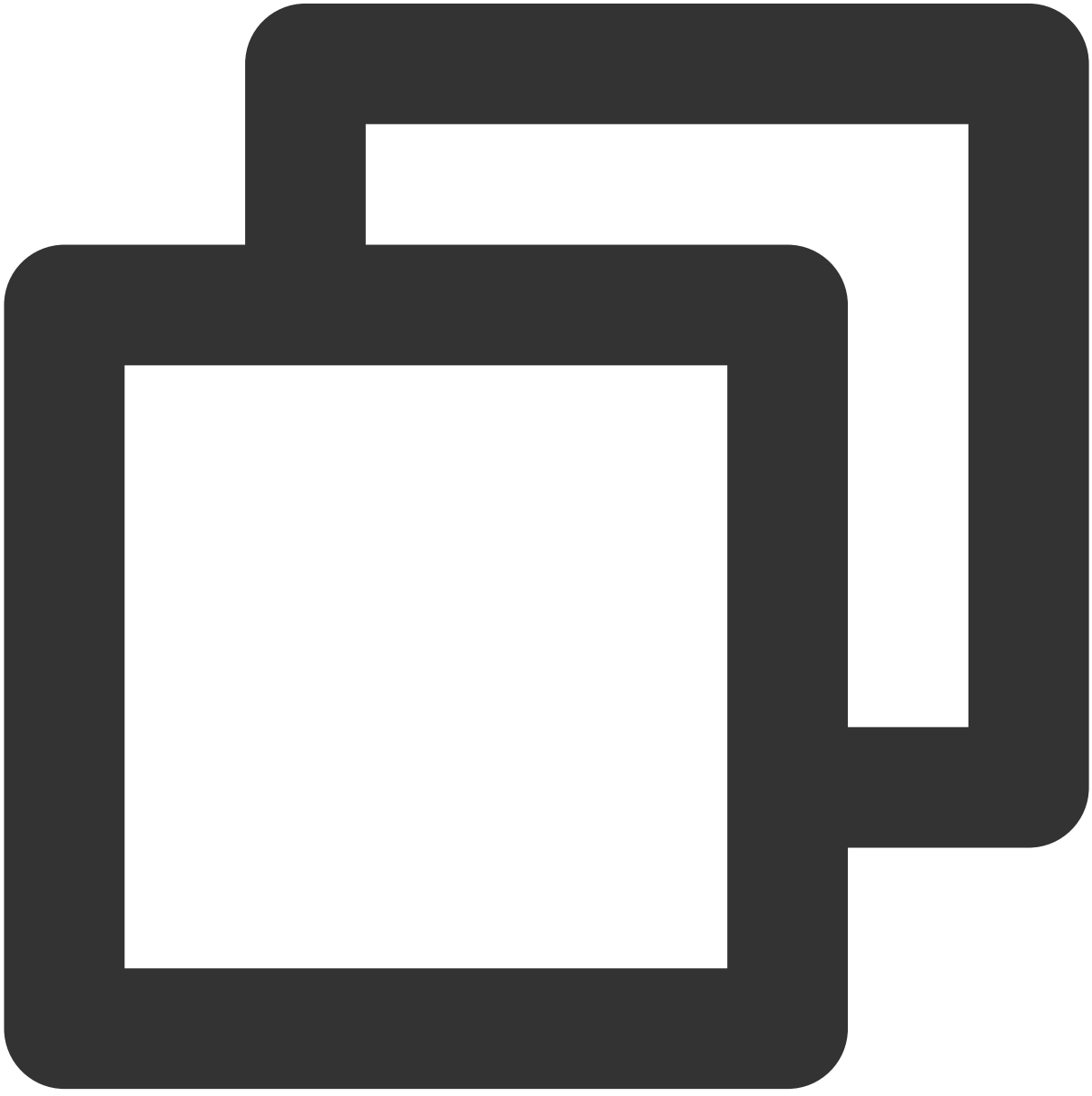


```
{
  "entryPagePath": "pages/index/index"
}
```

pages

©2013-2022 Tencent Cloud. All rights reserved.

That specifies which pages the Mini Program consists of, with each item corresponding to a page Path (including file name) Information. Filename does not need to write file suffix, framework will automatically find the corresponding location .json, .js, .wxml, .wxss Four files for processing.
Not specified entryPagePath The first item in the array represents the initial page (home page) of the Mini Program.
Mini Program Added/Reduce pages, all need to be on pages Array to modify.
If the development directory is:



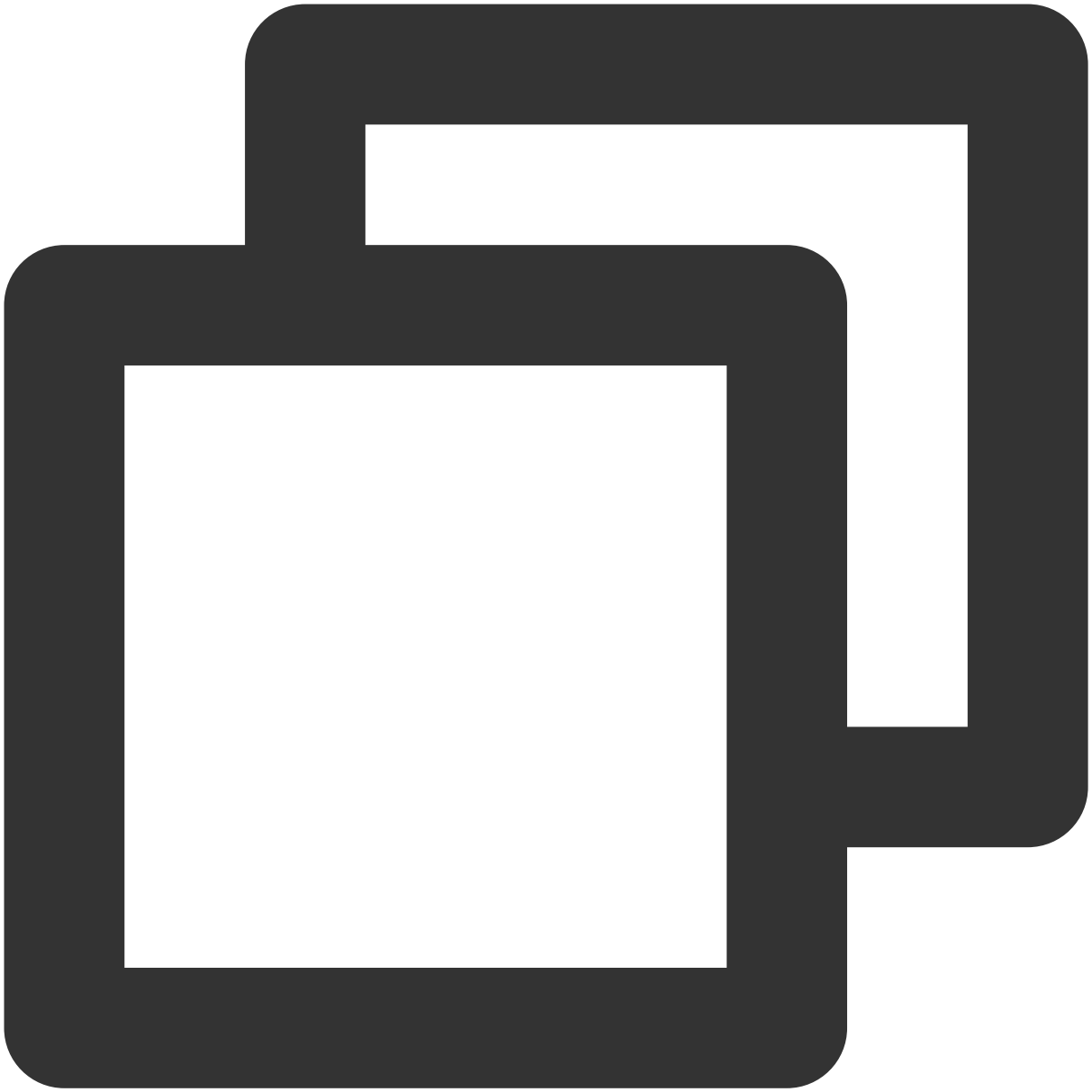
```
├── app.js
├── app.json
├── app.wxss
```

©2013-2022 Tencent Cloud. All rights reserved.

```
├── pages
│   ├── index
│   │   ├── index.wxml
│   │   ├── index.js
│   │   ├── index.json
│   │   └── index.wxss
│   └── logs
│       ├── logs.wxml
│       └── logs.js
└── utils
```

You need to app.json Middle write

©2013-2022 Tencent Cloud. All rights reserved.



```
{
  "pages": ["pages/index/index", "pages/logs/logs"]
}
```

window

Used to set Mini Program status bar, navigation bar, title, window background color.

attribute	type	default value	describe

©2013-2022 Tencent Cloud. All rights reserved.

navigationBarBackgroundColor	HexColor	#000000	Navigation bar background color, such as #000000
navigationBarTextStyle	string	white	Navigation bar header color, support only black / white
navigationBarTitleText	string	-	Navigation Title Text Content
navigationStyle	string	default	Navigation bar style supports only the following values default Default Style custom Customize the navigation bar, leaving only the capsule on in the upper right corner. See note 2.
backgroundColor	HexColor	#ffffff	The background color of the window
backgroundTextStyle	string	dark	Pull down loading Style, only supports dark / light
backgroundColorTop	string	#ffffff	The background color of the top window, only iOS Support
backgroundColorBottom	string	#ffffff	The background color of the bottom window, only iOS Support
enablePullDownRefresh	boolean	false	Whether to enable global drop-down refresh. See for details Page.onPullDownRefresh
onReachBottomDistance	number	50	The distance from the bottom of the page when the event is triggered, in px. See for details Page.onReachBottom
customNavigateBack	Boolean	false	Whether or not you want to intercept the default return of the page (tabBar return or side-swipe or back key return) is used in conjunction with Page.onCustomBack.
pageOrientation	string	portrait	Screen rotation settings, support car / portrait / landscape See for details Response to display area changes

notes 1: Hex Color (hex color value), as in "#ff00ff"

notes 2: About

navigationStyle

©2013-2022 Tencent Cloud. All rights reserved.

iOS/Android client 7.0.0 The following version,navigationStyle Only in app.json In effect;
iOS/Android client 6.7.2 Version starts,navigationStyle: custom Yes web-view Invalid component;

tabBar

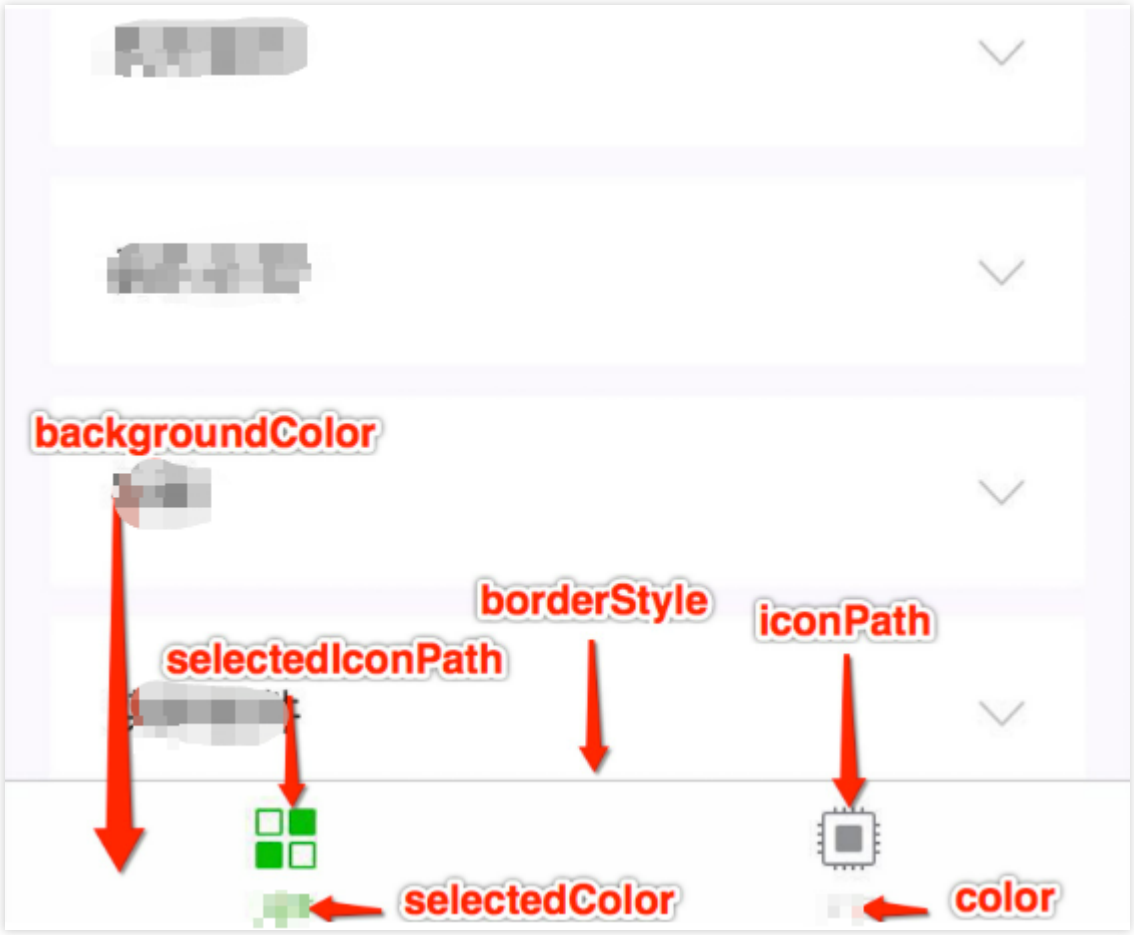
If the Mini Program is more than one tab Application (the bottom or top of the client window has tab Column to switch pages), you can click through the tabBar Configuration item designation tab Column performance, and tab The corresponding page to display when toggling.

attribute	type	required	default	describe
color	HexColor	yes	-	tab Default color for text on, only hexadecimal color is supported
selectedColor	HexColor	yes	-	tab Supports hexadecimal colors only
backgroundColor	HexColor	yes	-	tab Supports only hexadecimal colors
borderStyle	string	no	black	Tabs The color of the top border, Support only black / white
list	Array	yes	-	tab For a list of list Property explaination, minimum 2 Most 5 individual tab
position	string	no	bottom	tabBar Only supports the location of bottom / top
custom	boolean	no	false	custom Tab Bar, see you Custom tabBar

among list Take an array,**Only minimal configuration 2 Most 5 individual tab.** tab Sorted in the order of the array, each item is an object with the following property values.

attribute	type	required	introductions
pagePath	string	yes	Page path, must be in the pages First definition
text	string	yes	tab Up on text
iconPath	string	no	Picture path, icon Size limit is 40kb, recommended size 81px * 81px, does not support web pictures. when position for top Does not display when icon.
selectedIconPath	string	no	Selected picture path, icon Size limit is 40kb, recommended size 81px * 81px, does not support web pictures. when position for top Does not display when icon.

©2013-2022 Tencent Cloud. All rights reserved.



debug

Can be turned on in the developer tools debug Mode, in the developer tools console panel, debugging information to info Is given in the form of Page Registration, page routing, data update, event triggering, etc. Can help developers quickly locate some common problems.

subpackages

Enable Subpackage Loading, declare the project sub-packageing structure.

Note：

Written subPackages Also support.

workers

Use Worker When working with multithreaded tasks, setting the Worker Directory for code placement.

navigateToMiniProgramAppIdList

When a mini program needs to use navigateToMiniProgram interface to jump to other mini programs, you need to declare the list of appId of mini programs you need to jump to in the configuration file, at most 10 appId are allowed.

preloadRule

statementSubcontract pre-download.

©2013-2022 Tencent Cloud. All rights reserved.

usingComponents

Custom components declared in app.json are considered global custom components and can be used directly on pages or custom components within the mini program without being declared again. It is recommended to declare only those custom components that will be used on almost all pages.

Note：

Global custom components are considered to be relied upon by all pages and will be initialised when all pages are launched, which affects startup performance and takes up the size of the main package. Custom components that are only referenced by individual pages or sub-packages should be declared in the page configuration as much as possible.

permission

Mini Program Interface Permissions Related settings. Field type is Object, structured as:

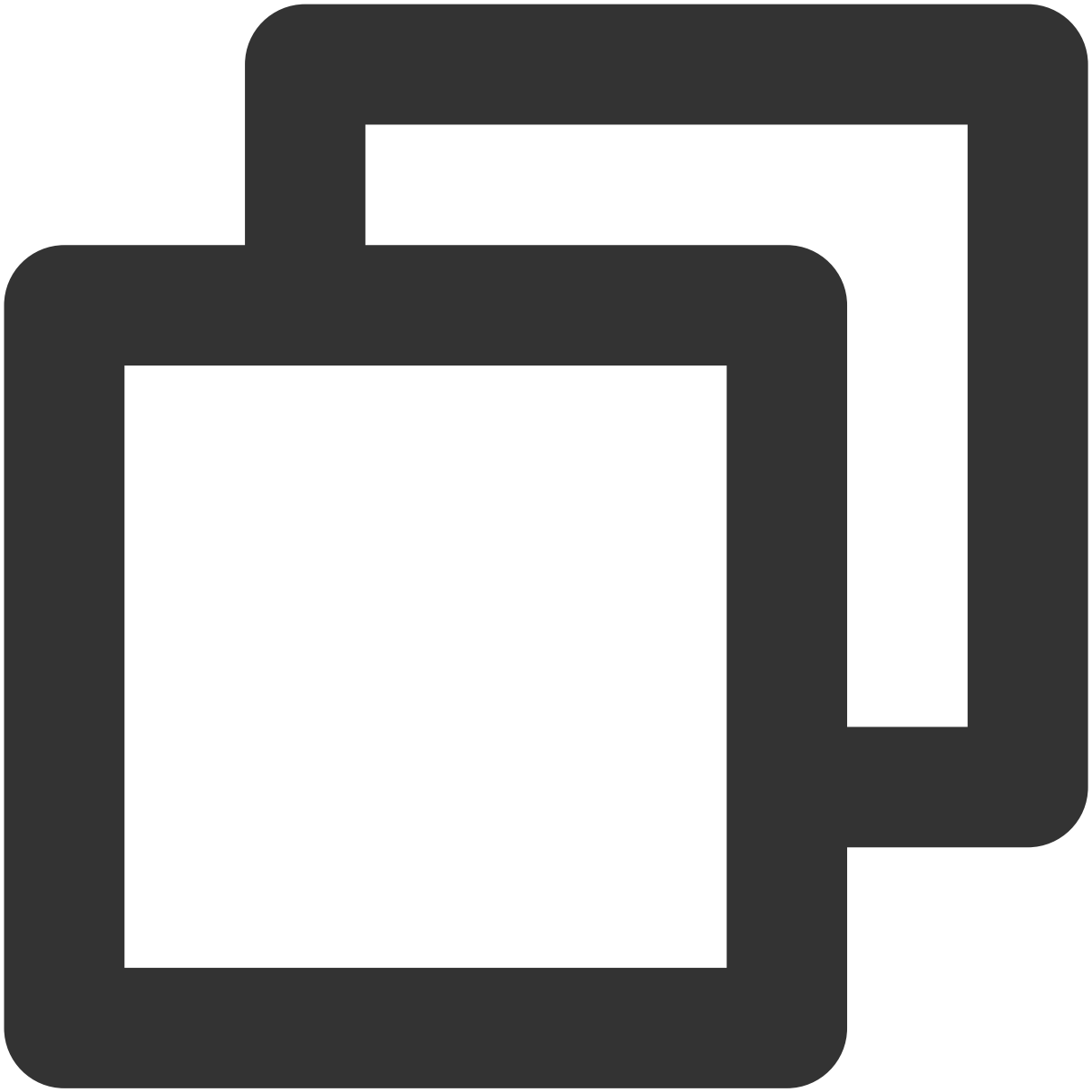
attribute	type	required	default	describe
scope.userLocation	PermissionObject	no	-	Location-dependent permission statement

Permission Object structure

attribute	type	Required	Default	Introductions
desc	string	yes	-	Description of the Chinese usage of the interface displayed by the mini program when obtaining permissions. Maximum 80 characters
desc-en	string	no	-	Description of the English usage of the interface displayed by the mini program when obtaining permissions. Maximum 80 characters
desc-adesc-ar	string	no	-	A description of the Arabic usage of the interface displayed when the mini program gets permissions. Maximum 80 characters

Such as:

©2013-2022 Tencent Cloud. All rights reserved.

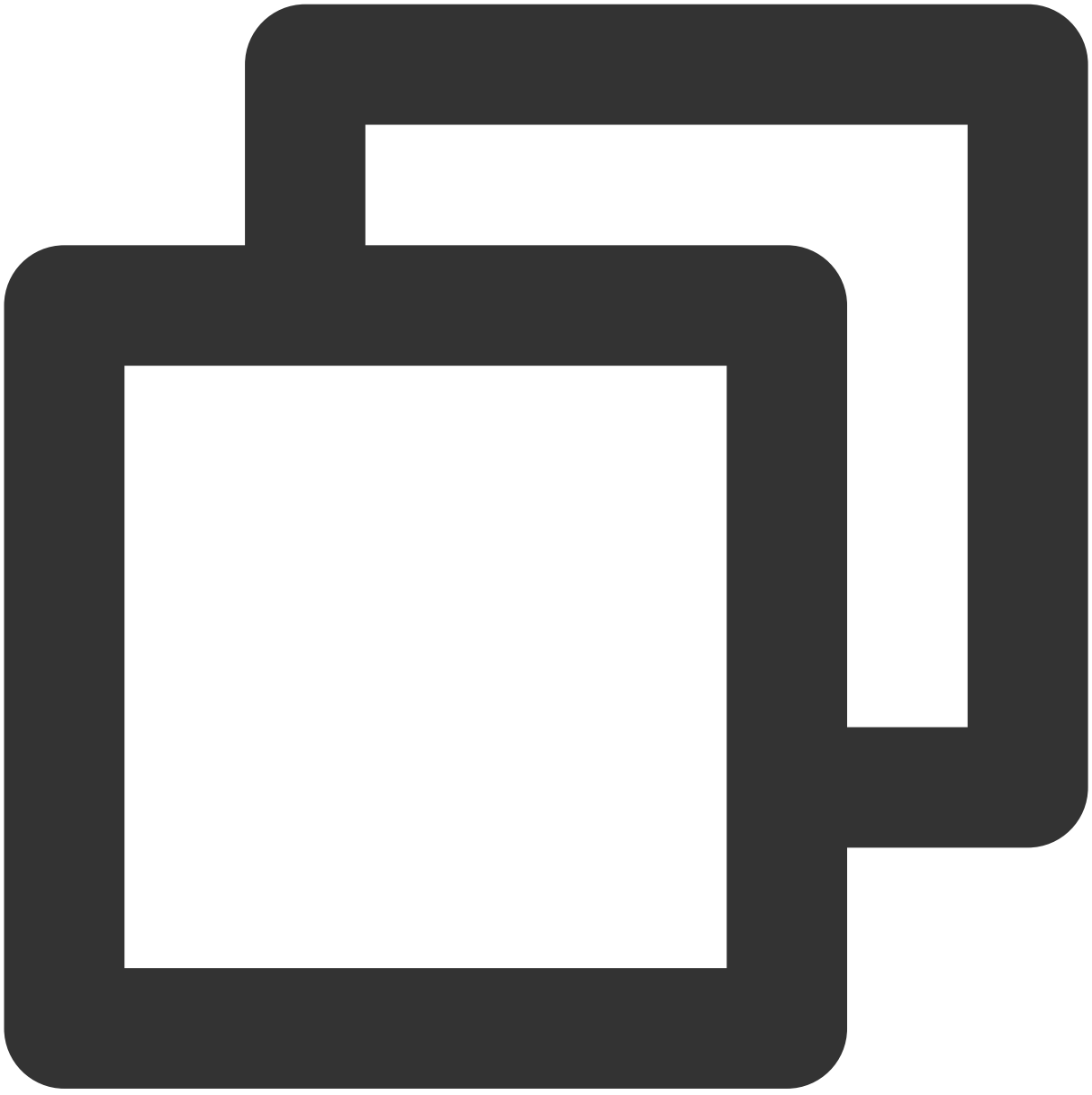


```
{
  "pages": ["pages/index/index"],
  "permission": {
    "scope.userLocation": {
      "desc": "Your location information will be used to display the effects of the
    }
  }
}
```

darkmode

©2013-2022 Tencent Cloud. All rights reserved.

You can configure `"darkmode": true` to indicate that the current mini program can be adapted to DarkMode, all the basic components will display different default styles according to the system theme, navigation bar and tab bar will also be automatically switched according to the developer's configuration.
After configuration, please follow the DarkMode Adaptation Guide to do your own adaptation beyond the basic style.



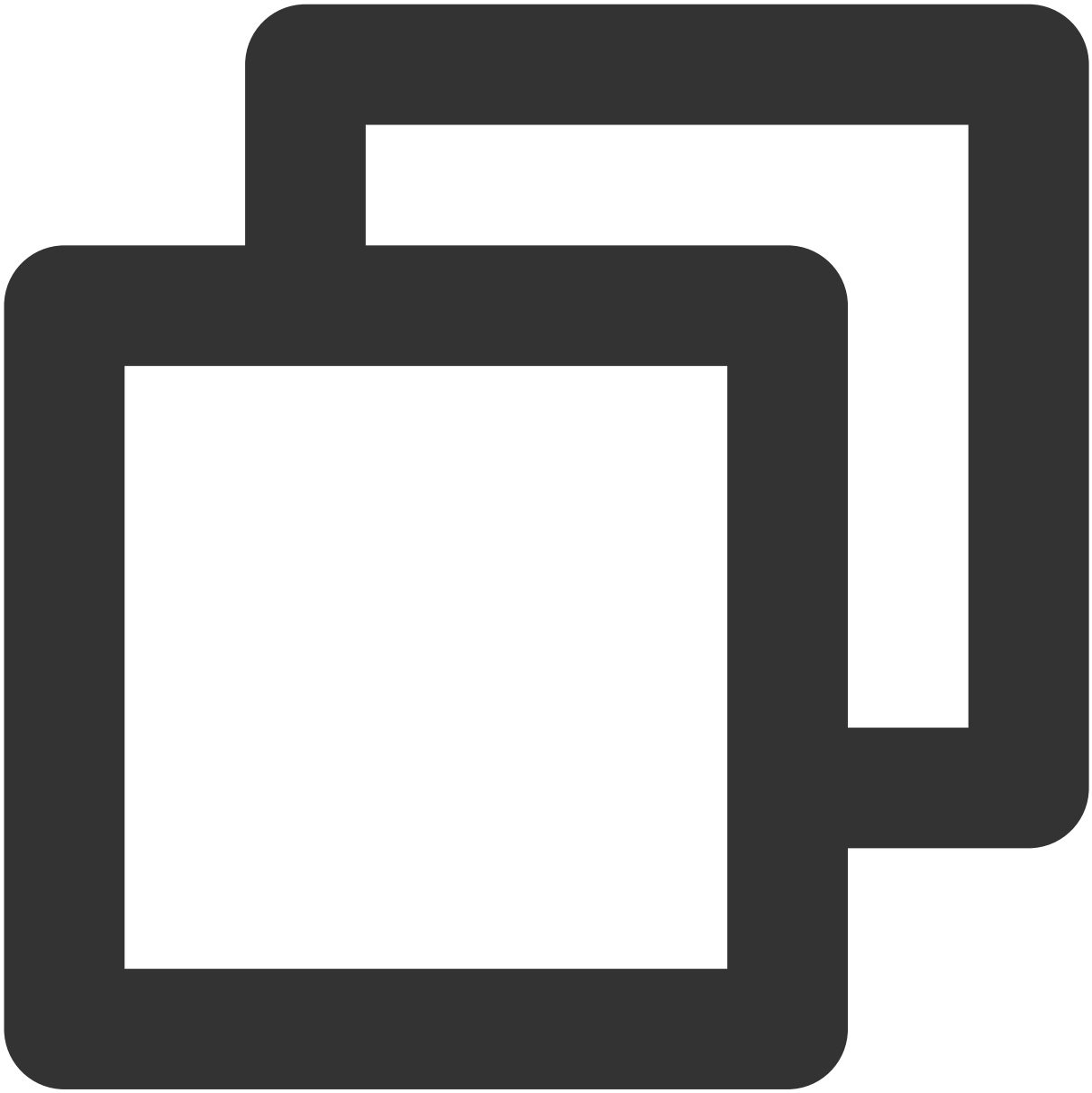
```
{
  "darkmode": true
}
```

themeLocation

©2013-2022 Tencent Cloud. All rights reserved.

custom theme.json Path, when configured `'darkmode': true` ,The current configuration file is required.

Configuration example



```
{
  "pages": ["pages/index/index", "pages/logs/index"],
  "window": {
    "navigationBarTitleText": "Demo"
  },
```

©2013-2022 Tencent Cloud. All rights reserved.

```
  "tabBar": {
    "list": [
      {
        "pagePath": "pages/index/index",
        "text": "index"
      },
      {
        "pagePath": "pages/logs/logs",
        "text": "logs"
      }
    ]
  },
  "debug": true
}
```

©2013-2022 Tencent Cloud. All rights reserved.

Page Configuration

Last updated：2024-07-12 18:34:42

Some of the configurations in app.json can also be configured for individual pages. You can configure the performance of the page in the corresponding .json file.
The configurations in the page will override the same configurations in app.json on the current page (style-related configurations are part of the window attribute in app.json, but you don't need to specify the window field here), and you can refer to the global configuration for the values and meanings of the configurations.
The content of the file is a JSON object with the following properties:

Configuration item

attribute	type	default	describe
navigationBarBackgroundColor	HexColor	#000000	Navigation bar background color, such as #000000
navigationBarTextStyle	string	white	Navigation bar header color, support only black / white
navigationBarTitleText	string		Navigation Title Text Content
navigationStyle	string	default	Navigation bar style supports only the following values default Default Style custom Customize the navigation bar, leaving only the capsule on in the upper right corner. See note 1。
backgroundColor	HexColor	#ffffff	The background color of the window
backgroundTextStyle	string	dark	Pull down loading Style, only supports dark / light
backgroundColorTop	string	#ffffff	The background color of the top window, only iOS Support
backgroundColorBottom	string	#ffffff	The background color of the bottom window, only iOS Support
enablePullDownRefresh	boolean	false	Whether to enable the current page drop-down refresh.

©2013-2022 Tencent Cloud. All rights reserved.

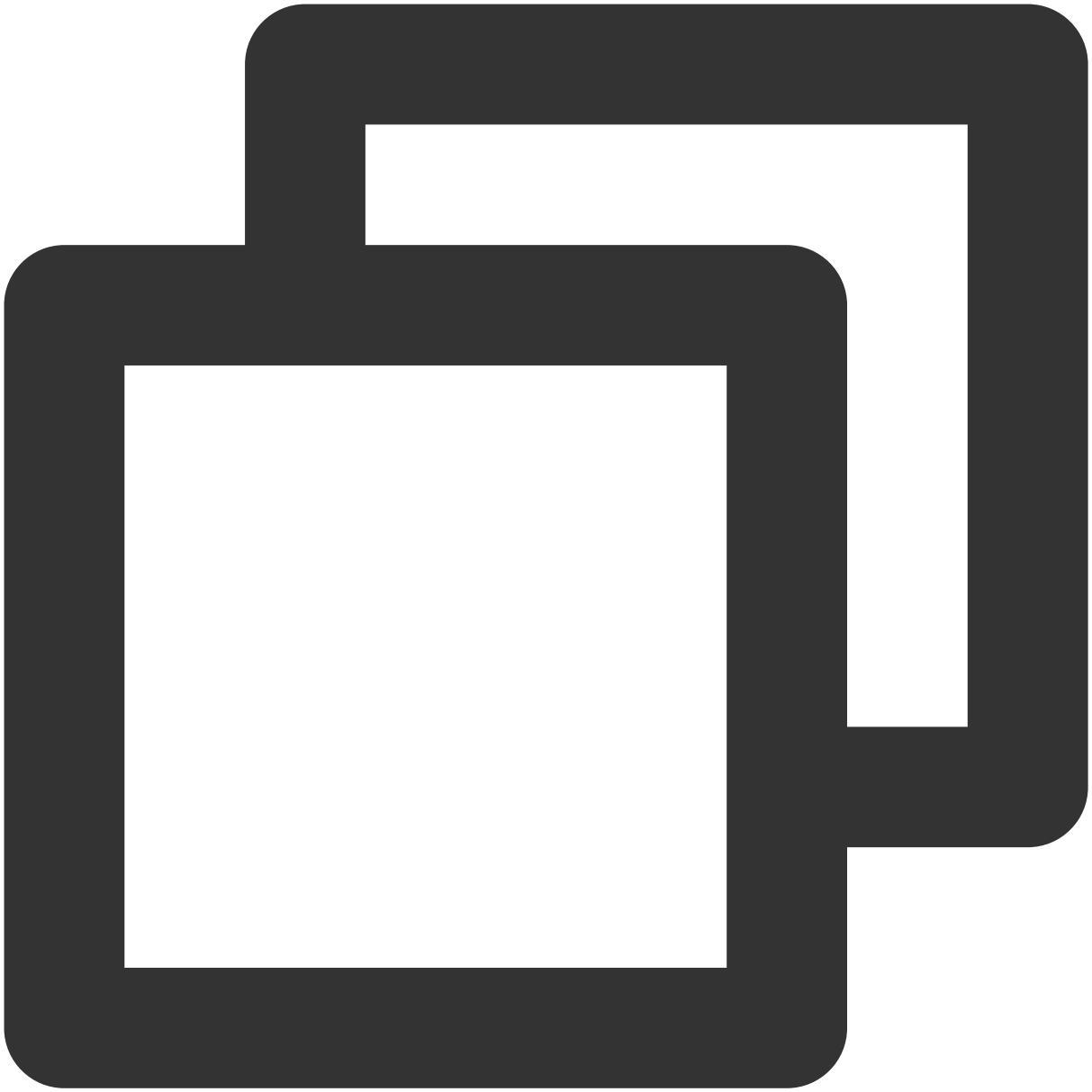
			See for details Page.onPullDownRefresh
onReachBottomDistance	number	50	The distance, in px, from the bottom of the page when the pull-down event is triggered. See for details Page.onReachBottom
pageOrientation	string	portrait	Screen rotation settings, support car / portrait / landscape See for details Response to display area changes
disableScroll	boolean	false	Set to true The page cannot scroll up and down as a whole. Valid only in page configuration, not in app.json Set in
usingComponents	Object	no	page Custom Components configuration

Note：

Not all configurations in app.json can be overridden or specified separately on the page, limited to the options included in this document.

Configuration example

©2013-2022 Tencent Cloud. All rights reserved.



```
{
  "navigationBarBackgroundColor": "#ffffff",
  "navigationBarTextStyle": "black",
  "navigationBarTitleText": "WeChat interface function demonstration,"
  "backgroundColor": "#eeeeee",
  "backgroundTextStyle": "light"
}
```

©2013-2022 Tencent Cloud. All rights reserved.

Framework Interface

App

Last updated：2024-07-12 18:40:19

App(Object object)

Sign up for the Mini Program. Take one. Object Parameter that specifies the lifecycle callbacks of Mini Programs, and so on.
App() Must be app.js Must be called and can be called only once. Or there will be unforeseen consequences.

parameter

Object object

attribute	type	default	required	Introductions
onLaunch	function	-	no	Lifecycle callbackMonitor Mini Program initialization.
onShow	function	-	no	Lifecycle callbackMonitor Mini Program start or cut the foreground.
onHide	function	-	no	Lifecycle callbackListen to the Mini Program cut background.
onError	function	-	no	Error listener.
onPageNotFound	function	-	no	There is no listener function on the page.
onUnhandledRejection	function	-	no	Untreated Promise Rejects event listener functions.
onThemeChange	function	-	no	Monitor subject changes
Other	any	-	no	Developers can add any function or data variable to Object Parameter, with the this Can access

For the definition of Mini Programs and the mechanism for running Mini Programs, see Operation mechanism.

©2013-2022 Tencent Cloud. All rights reserved.

onLaunch(Object object)

Mini Program initialization is completed when the trigger, the global trigger only once. Parameters can also be used wx.getLaunchOptionsSync Get.

parameter: With wx.getLaunchOptionsSync Agreement

onShow(Object object)

Mini Program start, or from the background into the foreground triggered when the display. You can also use wx.onAppShow Bind listening.

parameter: With wx.onAppShow Agreement

onHide()

Triggered when the Mini Program enters the background from the foreground. You can also use wx.onAppHide Bind listening.

onError(String error)

Script error or API Triggered on an error call. You can also use wx.onError Bind listening.

parameter: With wx.onError Agreement

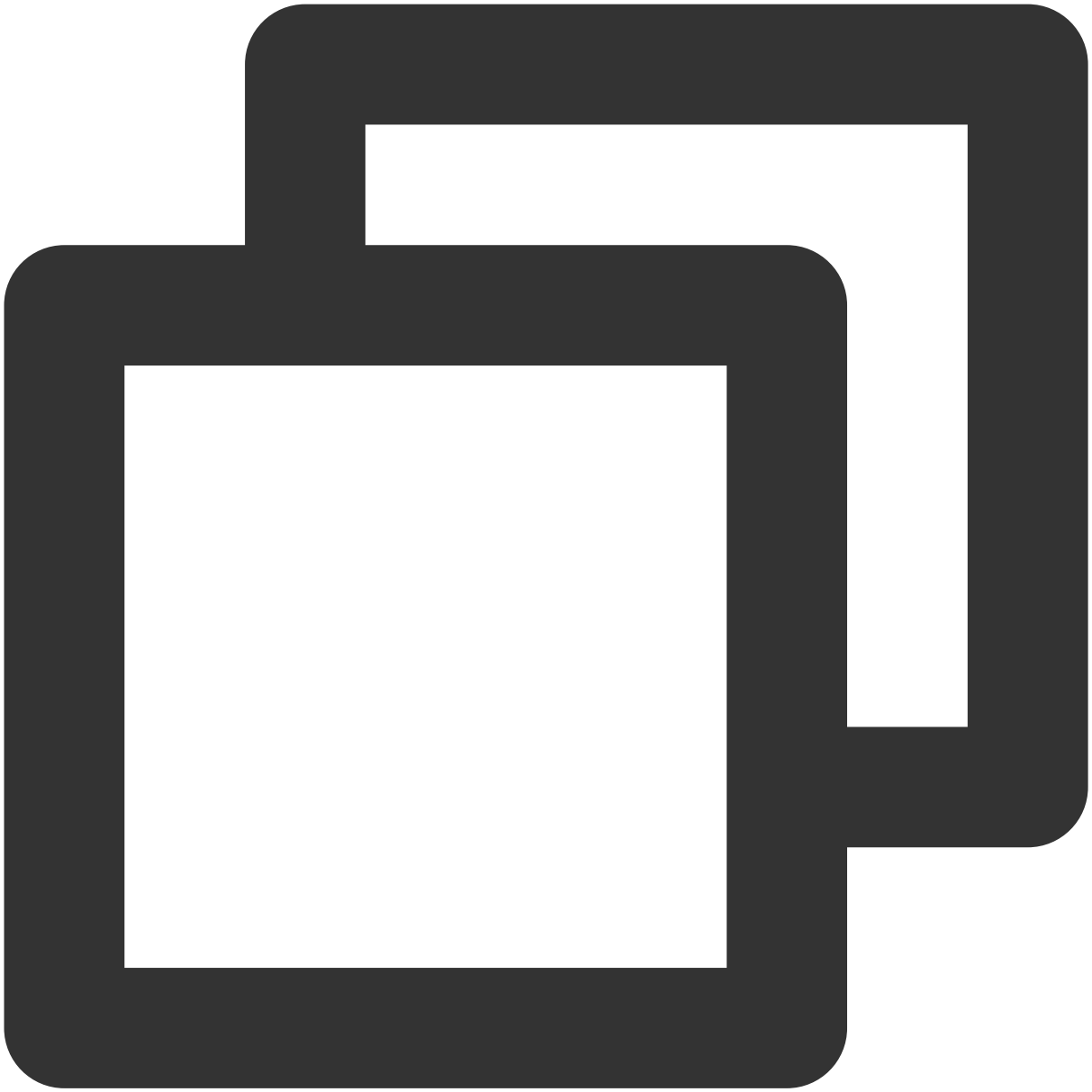
onPageNotFound(Object object)

Triggered when the page the Mini Program is trying to open does not exist. You can also use wx.onPageNotFound Bind listening. Attention please refer to wx.onPageNotFound.

parameter: With wx.onPageNotFound Agreement

Sample code:

©2013-2022 Tencent Cloud. All rights reserved.



```
App({
  onPageNotFound(res) {
    wx.redirectTo({
      url: 'pages/...'
    }) // If it were Tabs Page, use the wx.switchTab
  }
})
```

onUnhandledRejection(Object object)

©2013-2022 Tencent Cloud. All rights reserved.

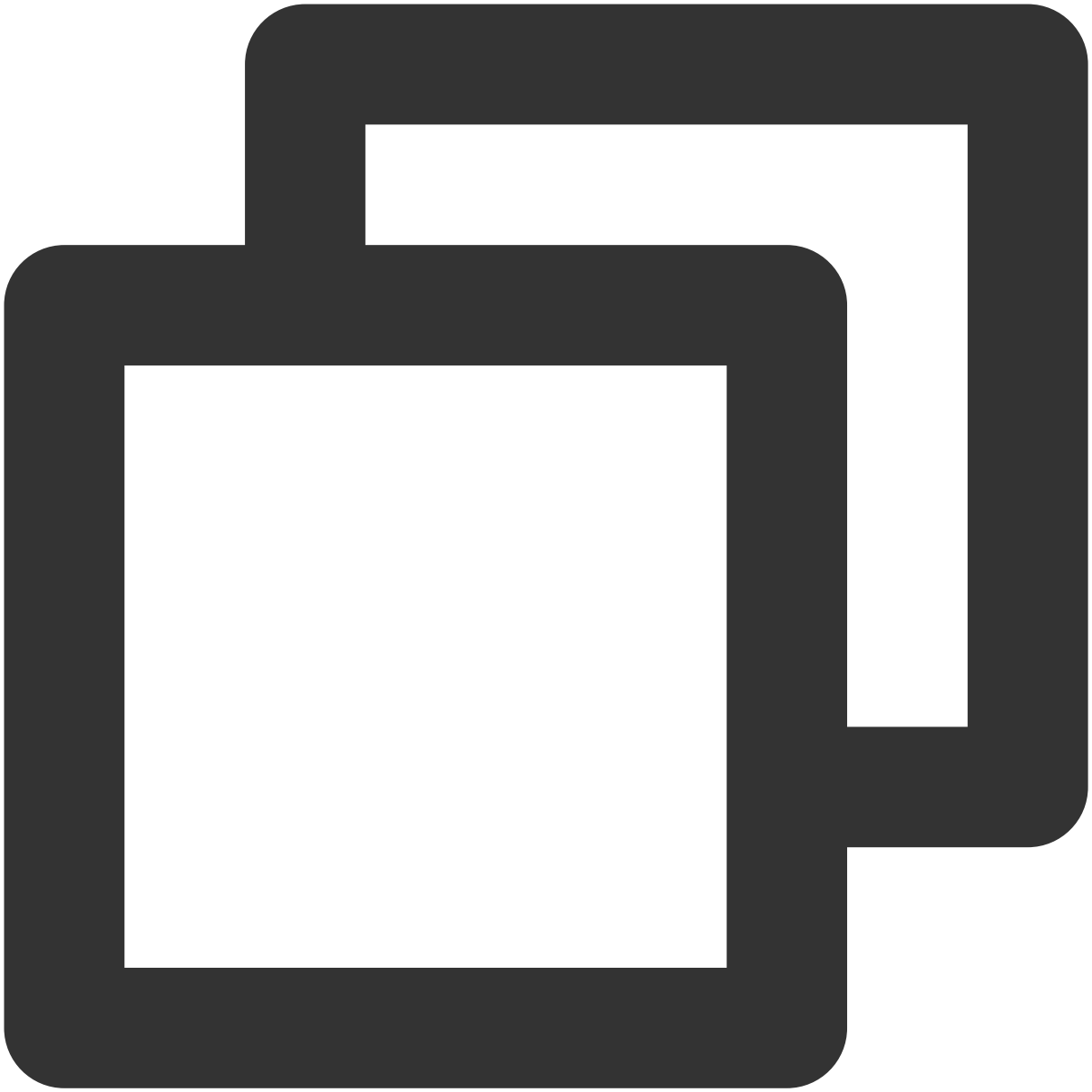
Mini Programs have unhandled Promise Triggered when rejected. You can also use wx.onUnhandledRejection Bind listening. Attention please refer to wx.onUnhandledRejection.
parameter: With wx.onUnhandledRejection Agreement

onThemeChange(Object object)

Triggered when the system switches topics. You can also use wx.onThemeChange Bind listening.
parameter: With wx.onThemeChange Agreement
The translations are provided by WeChat Translation and are for reference only. In case of any inconsistency and discrepancy between the Chinese version and the English version, the Chinese version shall prevail.Incorrect translation.

Sample code:

©2013-2022 Tencent Cloud. All rights reserved.



```
App({
  onLaunch (options) {
    // Do something initial when launch.
  },
  onShow (options) {
    // Do something when show.
  },
  onHide () {
    // Do something when hide.
  },
  onError (msg) {
```

©2013-2022 Tencent Cloud. All rights reserved.

```
    console.log(msg)
  },
  globalData: 'I am global data'
})
```

AppObject getApp(Object object)

Gets the Mini Programs globally unique App Instance.

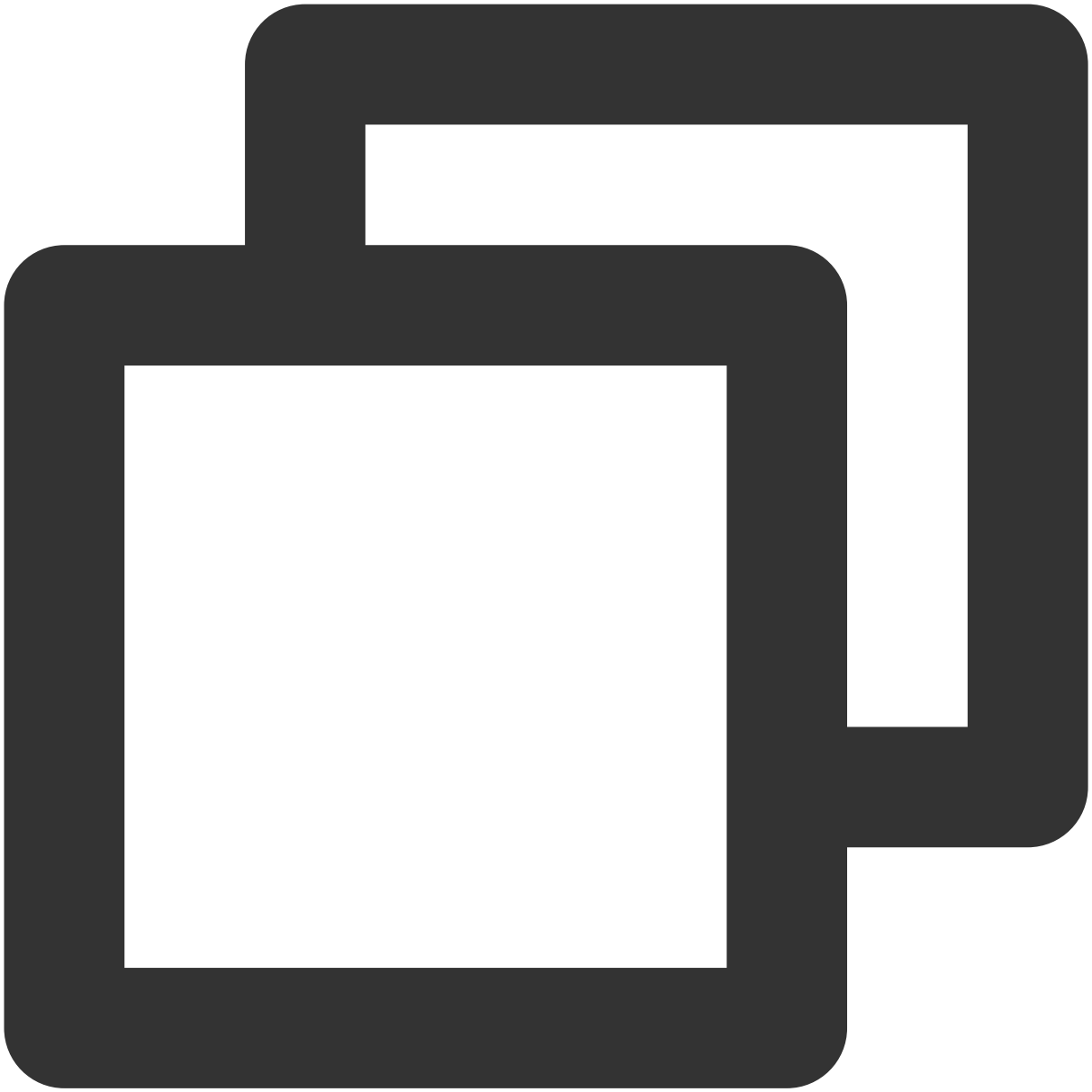
parameter

Object object

attribute	type	default	required	Introductions
allowDefault	boolean	false	no	in App Returns the default implementation when undefined. When the app is invoked, properties defined in the default implementation are overridden and incorporated into the app. Generally usedIndependent subcontracting

Sample code

©2013-2022 Tencent Cloud. All rights reserved.



```
// other.js
var appInstance = getApp()
console.log(appInstance.globalData) // I am global data
```

Be careful

Don't define App() Or call the App Precall getApp() , using this You can ge it. app Instance;

adopt getApp() Do not privately call the lifecycle function after obtaining the instance.

©2013-2022 Tencent Cloud. All rights reserved.

Page

Last updated：2024-07-12 18:45:48

Page(Object object)

Register a page in the Mini Program. Take one. Object Type parameters that specify initial data for the page, lifecycle callbacks, event handlers, and so on.

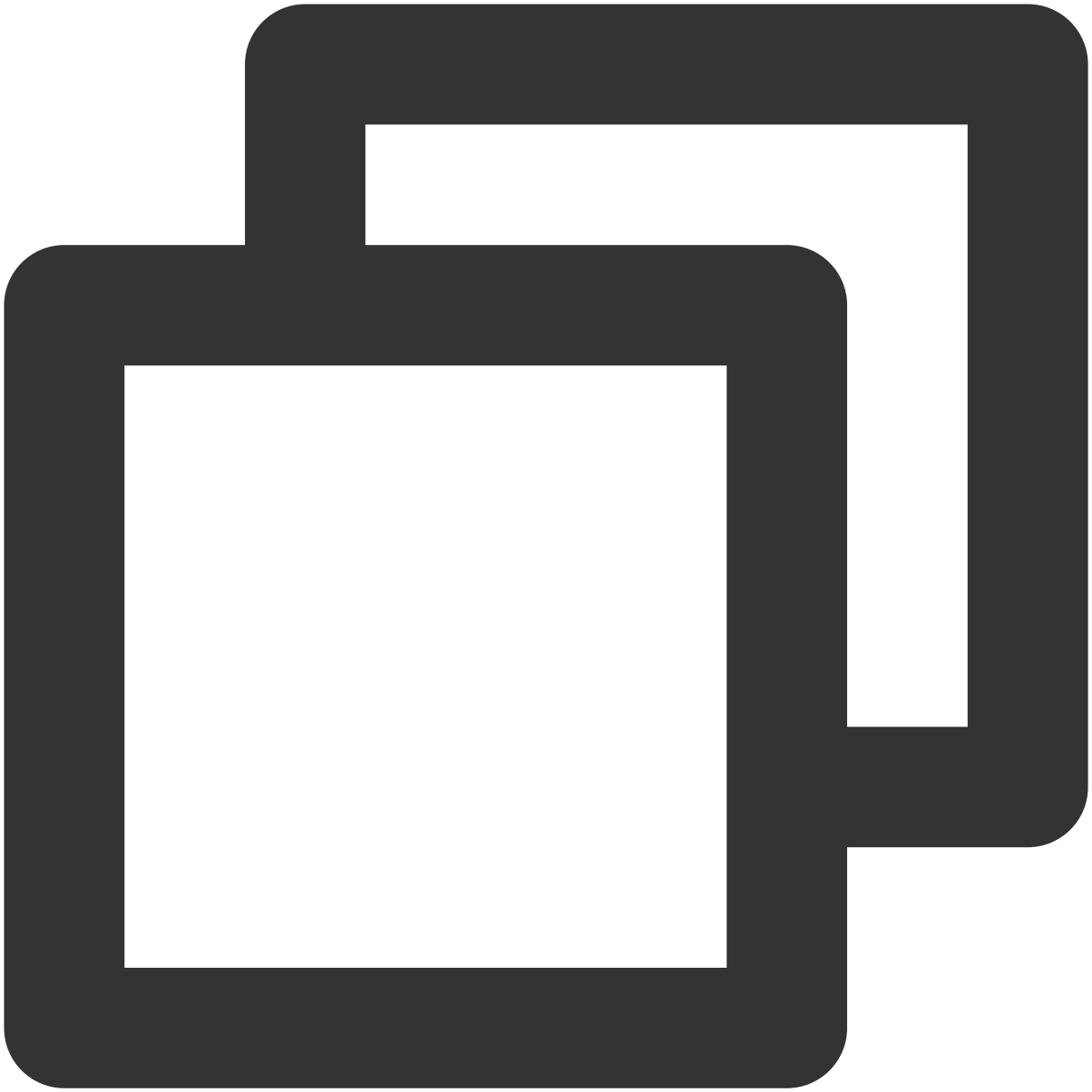
parameter

Object object

attribute	type	default	required	Introductions
data	Object	-	-	Initial data for the page
onLoad	function	-	-	Lifecycle callbackMonitor page loading
onShow	function	-	-	Lifecycle callbackMonitor page display
onReady	function	-	-	Lifecycle callbackMonitor Page Initial Rendering Completed
onHide	function	-	-	Lifecycle callbackMonitor Page Hiding
onUnload	function	-	-	Lifecycle callbackListen Page Uninstall
onPullDownRefresh	function	-	-	Listen for user drop-down actions
onReachBottom	function	-	-	Handle function for bottoming event on page
onShareAppMessage	function	-	-	User click on the upper right to forward
onPageScroll	function	-	-	Handler for page scrolling trigger event
onTabItemTap	function	-	-	At present tab Page, click tab Time trigger
Other	any-	-	-	Other developers can add arbitrary functions or data to the Object parameter, which can be accessed in the page's function with this. This portion of the property is made a deep copy when the page instance is created.

sample code

©2013-2022 Tencent Cloud. All rights reserved.



```
//index.js
Page({
  data: {
    text: "This is page data."
  },
  onLoad: function(options) {
    // Do some initialize when page load.
  },
  onShow: function() {
    // Do something when page show.
  },
```

©2013-2022 Tencent Cloud. All rights reserved.

```
  onReady: function() {
    // Do something when page ready.
  },
  onHide: function() {
    // Do something when page hide.
  },
  onUnload: function() {
    // Do something when page close.
  },
  onPullDownRefresh: function() {
    // Do something when pull down.
  },
  onReachBottom: function() {
    // Do something when page reach bottom.
  },
  onShareAppMessage: function () {
    // return custom share data when user share.
  },
  onPageScroll: function() {
    // Do something when page scroll
  },
  onResize: function() {
    // Do something when page resize
  },
  onTabItemTap(item) {
    console.log(item.index)
    console.log(item.pagePath)
    console.log(item.text)
  },
  // Event handler.
  viewTap: function() {
    this.setData({
      text: 'Set some data for updating view.'
    }, function() {
      // this is setData callback
    })
  },
  customData: {
    hi: 'MINA'
  }
})
```

data

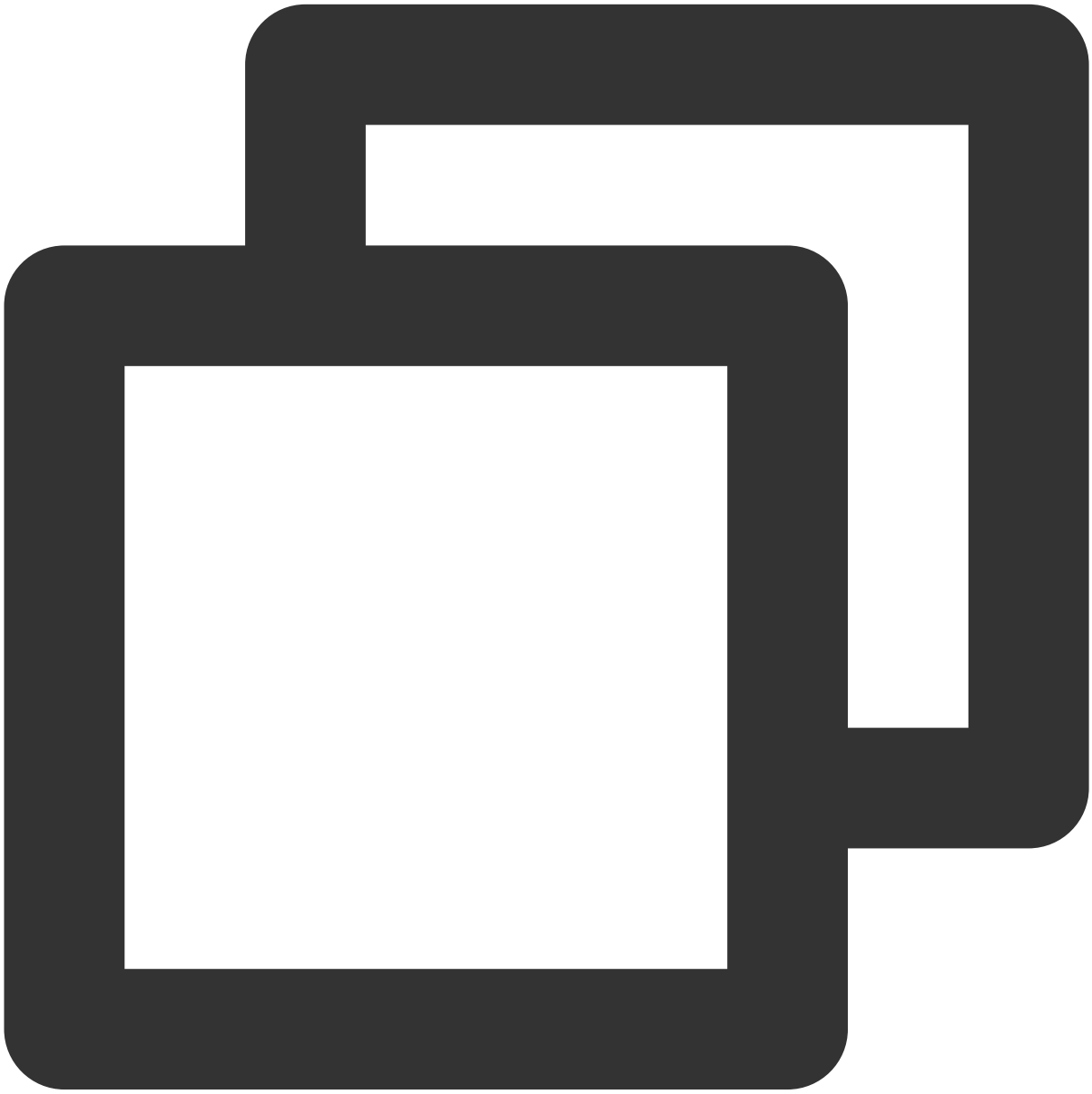
`data` is used in the first rendering of the pageInitial data。

©2013-2022 Tencent Cloud. All rights reserved.

When the page loads, `data` Will be `JSON` The string form is passed from the logical layer to the render layer, so `data` in must be converted toJSONType: string, number, boolean, object, array.

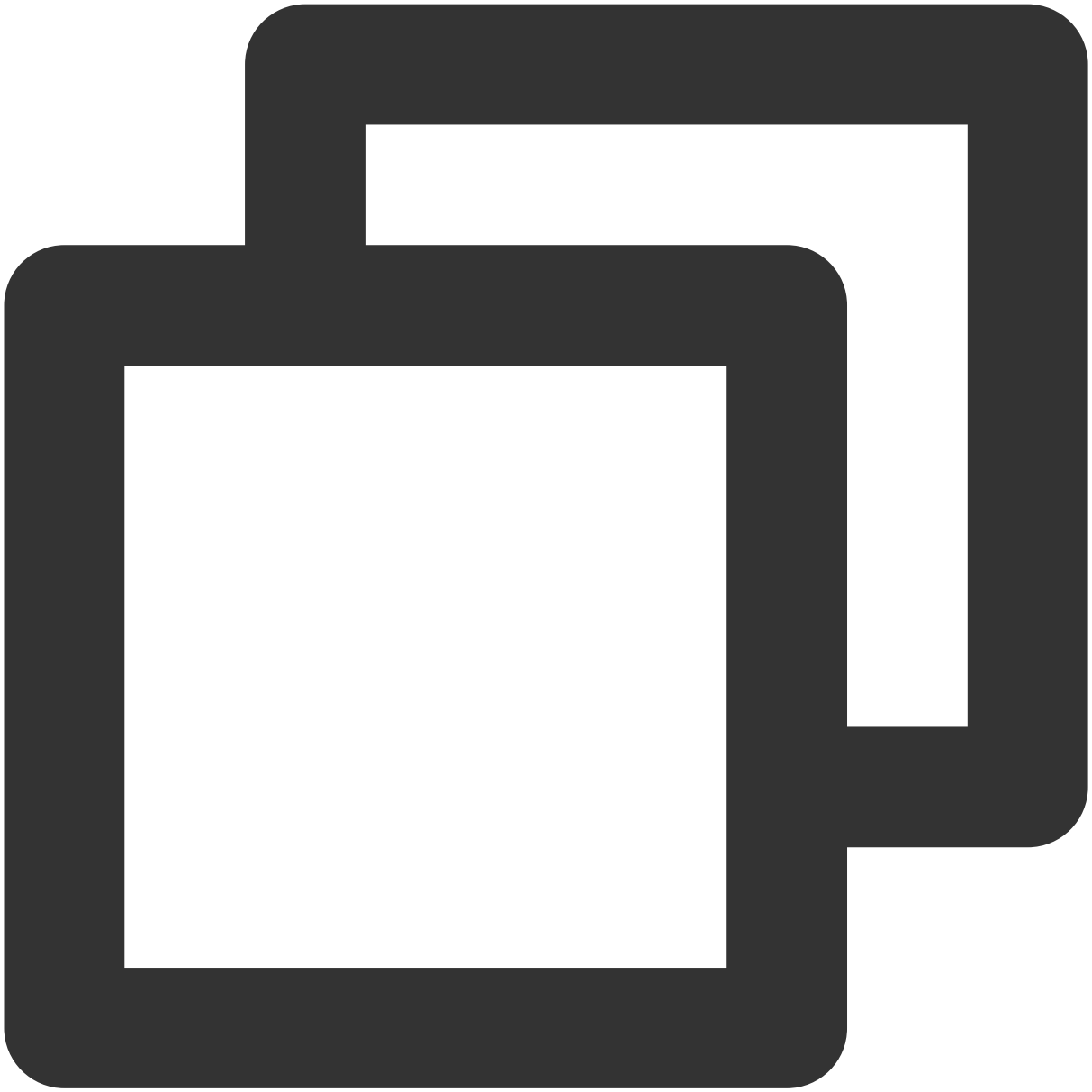
The render layer can be created by WXML Bind data.

Sample code:



```
<view>{{text}}</view>
<view>{{array[0].msg}}</view>
```

©2013-2022 Tencent Cloud. All rights reserved.



```
Page({
  data: {
    text: 'init data',
    array: [{msg: '1'}, {msg: '2'}]
  }
})
```

Lifecycle callback function

Triggering of the lifecycle and routing of the page Logic Layer.

©2013-2022 Tencent Cloud. All rights reserved.

onLoad(Object query)

Triggered when the page loads. A page will be called only once, and can be called in the onLoad Gets the parameters in the path to open the current page.

Parameters:

Name	type	Introductions
query	Object	Open parameters in the current page path

onShow()

Page display/Triggered when entering the front desk.

onReady()

Triggered when the initial rendering of the page is complete. A page is called only once, meaning the page is ready to interact with the view layer.

Note: to set the interface content API Such as wx.setNavigationBarTitle, in theonReadyAfter that. See for details Life cycle.

onHide()

Page concealment/Triggered when you cut into the background. Such as wx.navigateTo Or bottom tab Switch to other pages, Mini Programs into the background, etc.

onUnload()

Triggered when the page is unloaded. Such as wx.redirectTo or wx.navigateBack to another page.

Page event handler

onPullDownRefresh()

Listen for user drop-down refresh events;

Need to app.json of window option, or Page Configuration enablePullDownRefresh;

Can be passed wx.startPullDownRefresh triggers a drop-down refresh, and then triggers a pull-down refresh animation. The effect is consistent with the user's manual pulldown refresh;

After processing the data refresh , wx.stopPullDownRefresh can stop the drop-down refresh of the current page.

onReachBottom()

Listen for user bottom-up events.

Can be app.json of window option, or Page Configuration onReachBottomDistance;

This event will be triggered only once during slippage within the trigger distance.

onPageScroll(Object object)

©2013-2022 Tencent Cloud. All rights reserved.

Listens for user sliding page events.

parameter Object object:

attribute	type	Introductions
scrollTop	Number	Distance a page has been scrolled vertically (in px)

Note:

Please only be present when needed. page Do not define an empty method. To reduce the impact of unnecessary event dispatch on rendering layer - logic layer communication.

Please avoid onPageScroll Too frequent execution in setData Equal cause Logic Layer and Render Layer Communication operations. In particular, each transmission of large amounts of data will affect the communication time.

onShareAppMessage(Object object)

Listen to the user click on the in-page forward on button assembly open-type="share"Or the upper-right menu "Forward" on behavior, and customize the forward content.

Note:

The Forward on is displayed on the upper-right menu only if this event handler is defined.

parameter Object object:

parameter	type	Introductions
from	String	Forward the source of the event. button:: In-page forward on menu:: Retweet menu on top right
target	Object	if from The value is buttonThen target That triggered this retweet. button, otherwise undefined
WebViewUrl	String	The page contains web-view, returns the current web-view
shareTarget	string	Where users click to share

This event handler function needs to return an Object, which is used to customise the forwarding content, the return content is as follows:

fields	type	description	default
title	string	Forwarding Title	Current mini program name
path	string	Forwarding Path	Current page path, must be a full path starting

©2013-2022 Tencent Cloud. All rights reserved.

			with /.
imageUrl	string	Custom image url, can be a local file url, a package file url or a network image url. Support PNG and JPG, display image aspect ratio is 5:4	Use default screenshot
entryDataHash	string	Listen to the user clicking the in-page forward button, only with this parameter, fast sharing will be supported	-
shareType	string	Specify the type of share	"miniapp"

Description of the shareTarget parameter:

Value	Description
0	Share to QQ friends
1	Share to QQ space
2	Open from current chat window, quickly share to current chat window
3	Share to WeChat friends
4	Share to WeChat Friends
5	Share panel to recent contacts
6	Share to Quick Share Friends List

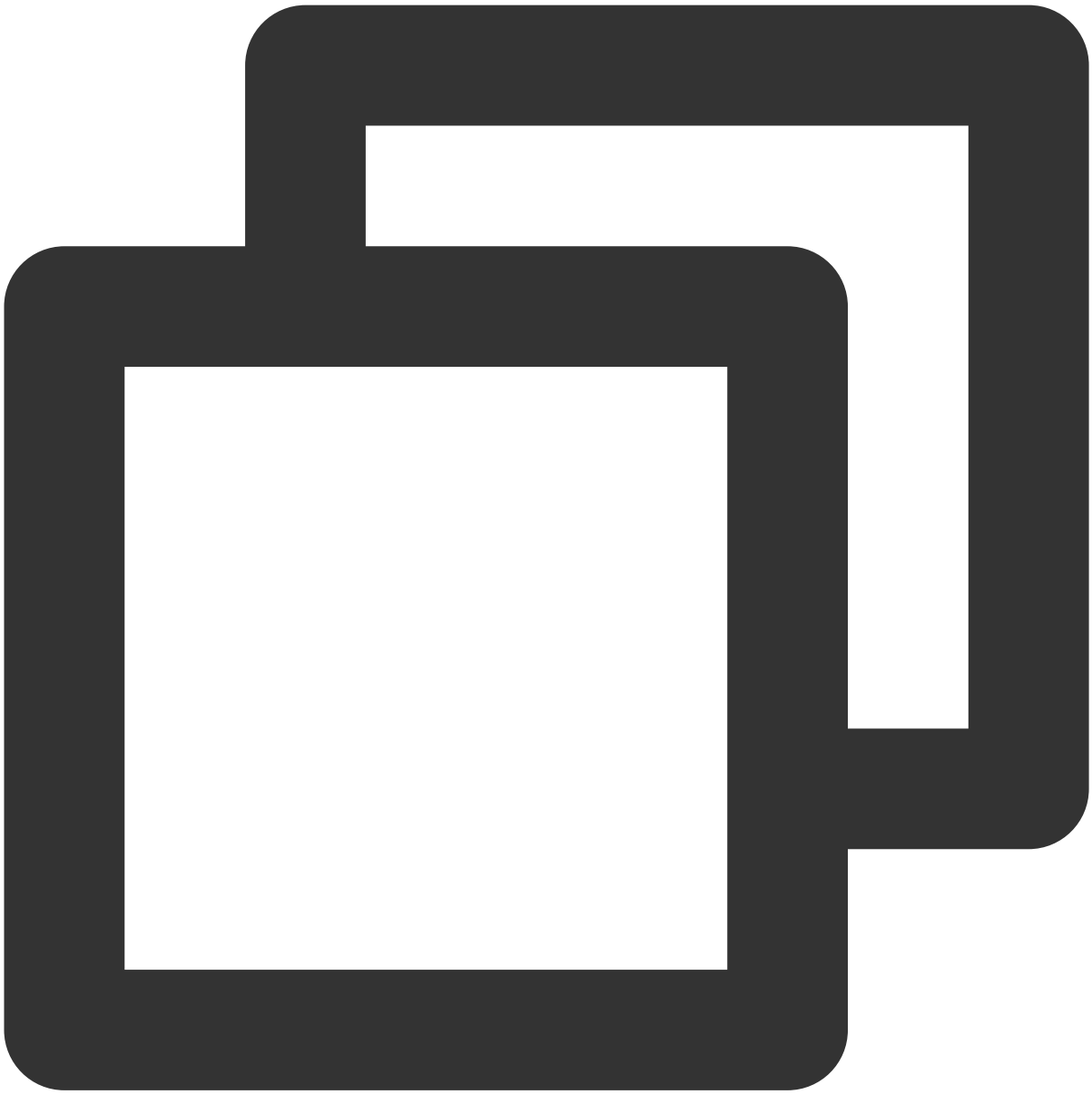
Description of the shareType parameter:

Used to specify the type of sharing, currently there are two types of sharing, one is to share a mini program and the other is to share an image.

Value	Description
"miniapp"	Shared as a mini program, the title, path, imageUrl, entryDataHash parameters take effect
"picture"	Shared as a picture, imageUrl, entryDataHash parameters will take effect

©2013-2022 Tencent Cloud. All rights reserved.

Sample code:



```
Page({
  onShareAppMessage(res) {
    if (res.from === 'button') {
      // From the in-page forwarding button
      console.log(res.target)
    }
    return {
      title: 'Custom Forwarding Title',
      path: '/page/user?id=123'
```

©2013-2022 Tencent Cloud. All rights reserved.

```
      }
    }
})
```

onCustomBack()

Listen to the event of user back to previous page.

You need to enable customNavigateBack in the window option of `app.json` or in the Page Configuration;

This event will be triggered if you click tabBar to go back, or if you swipe or back key to go back.

onTabItemTap(Object object)

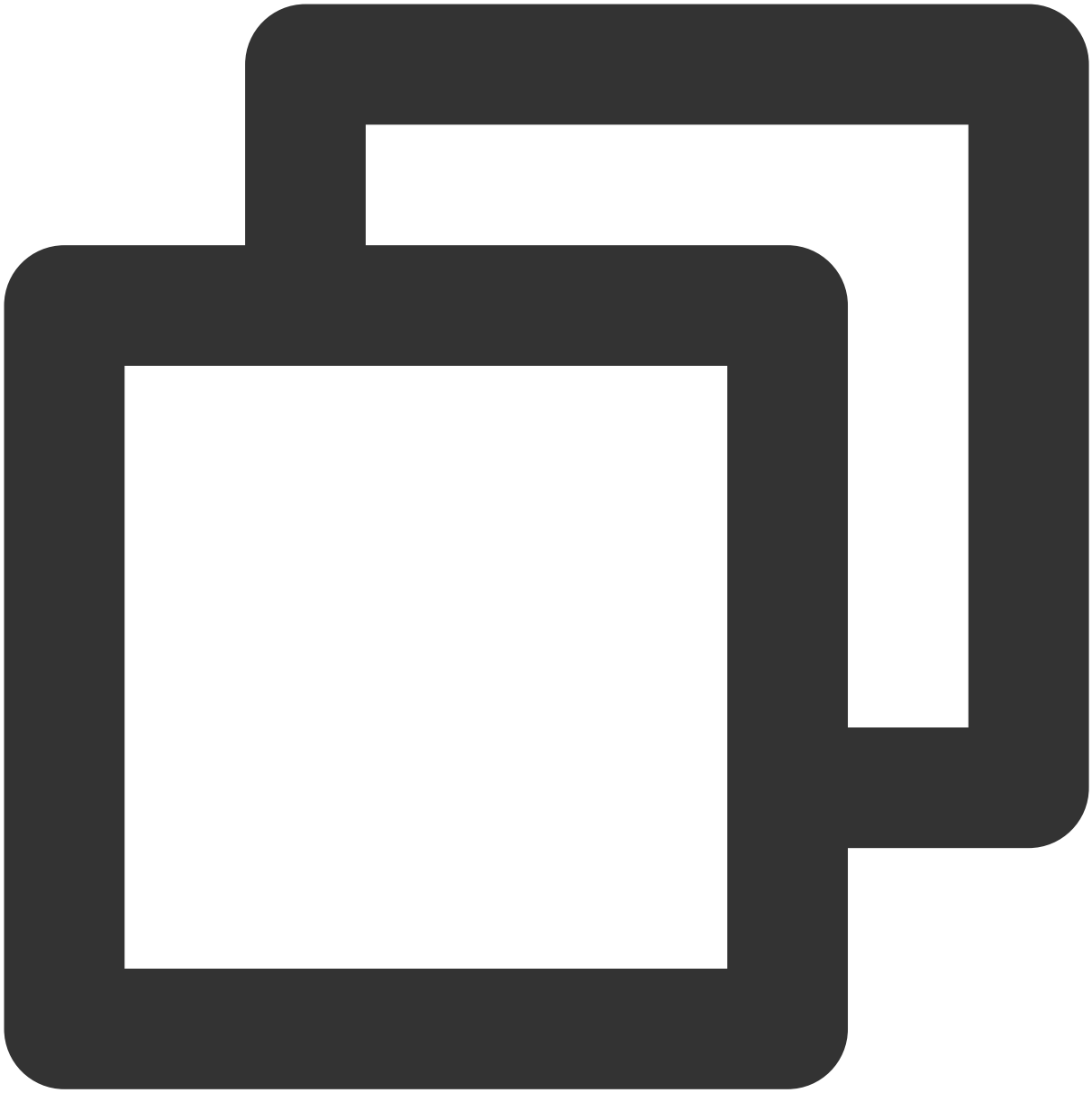
to hit tab Time trigger.

Object Parameter explaination:

parameter	type	Introductions
index	String	Number of tabItem clicked, starting with 0
pagePath	String	Page path of tabItem clicked
text	String	Click tabItem on text

Sample code:

©2013-2022 Tencent Cloud. All rights reserved.



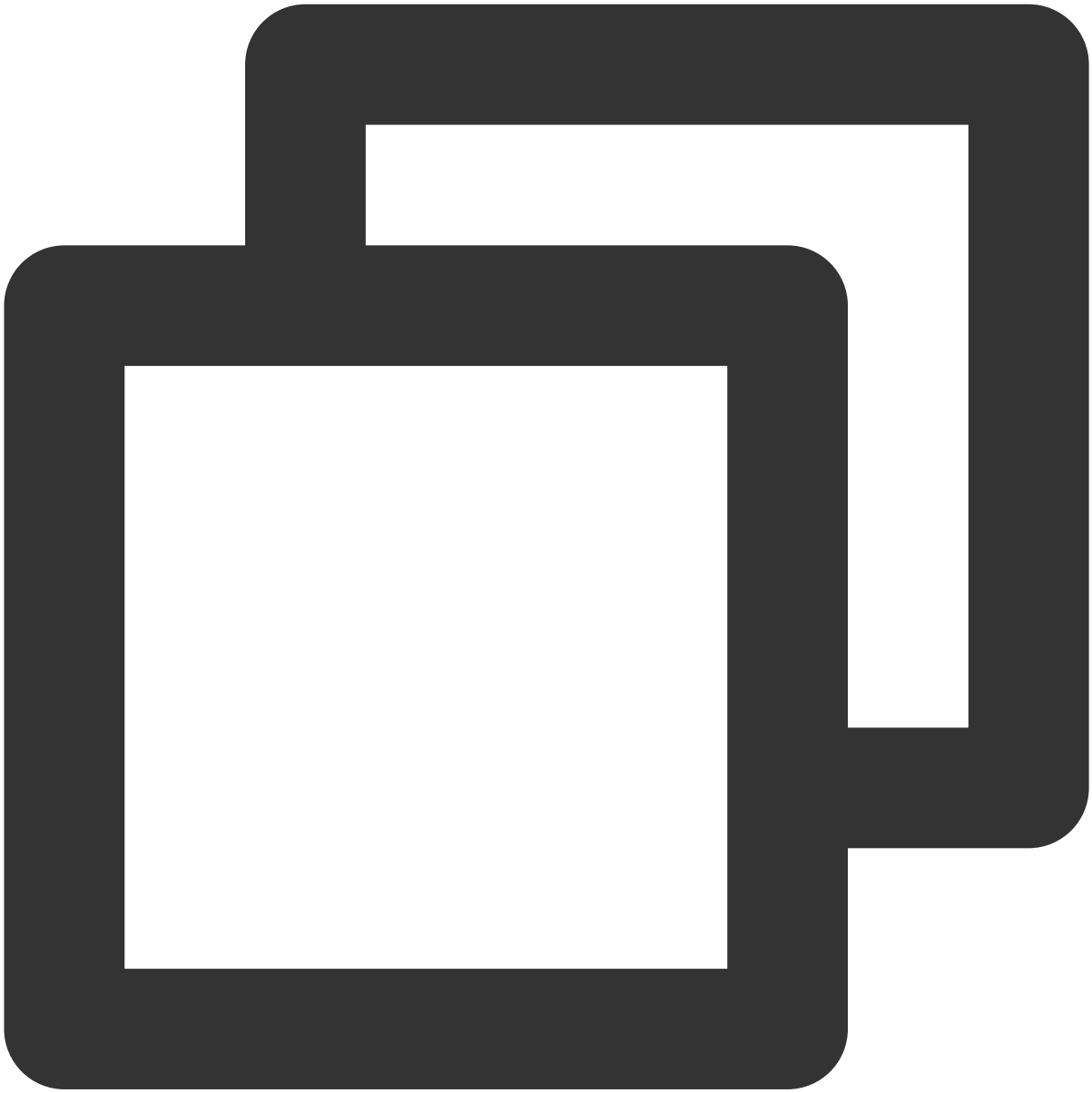
```
Page({
  onTabItemTap(item) {
    console.log(item.index)
    console.log(item.pagePath)
    console.log(item.text)
  }
})
```

Component event handler

©2013-2022 Tencent Cloud. All rights reserved.

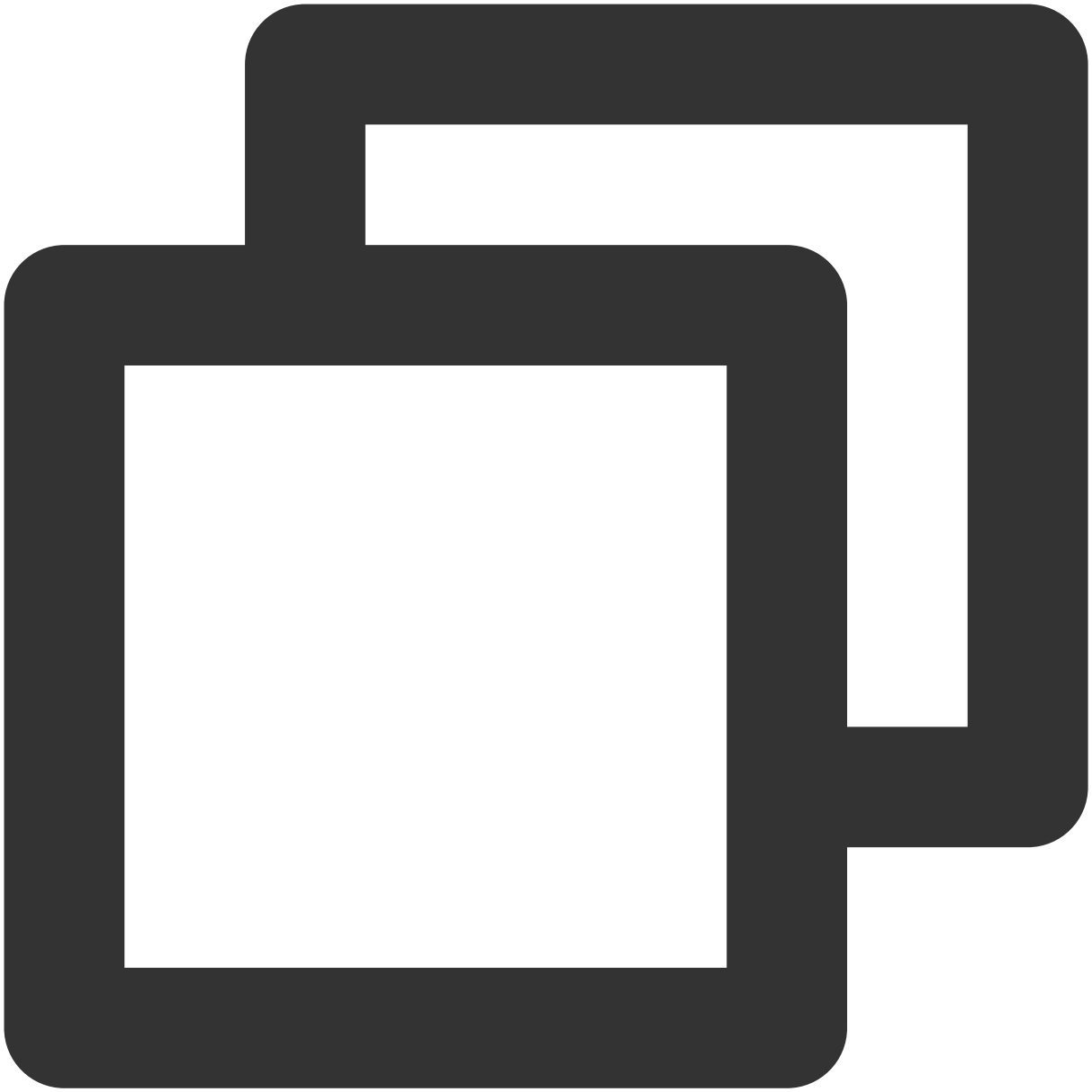
Page Component event handlers can also be defined in. In the components of the render layer Event binding is executed when an event is triggered Page The event handler defined in the.

Sample code:



```
<view bindtap="viewTap"> click me </view>
```

©2013-2022 Tencent Cloud. All rights reserved.

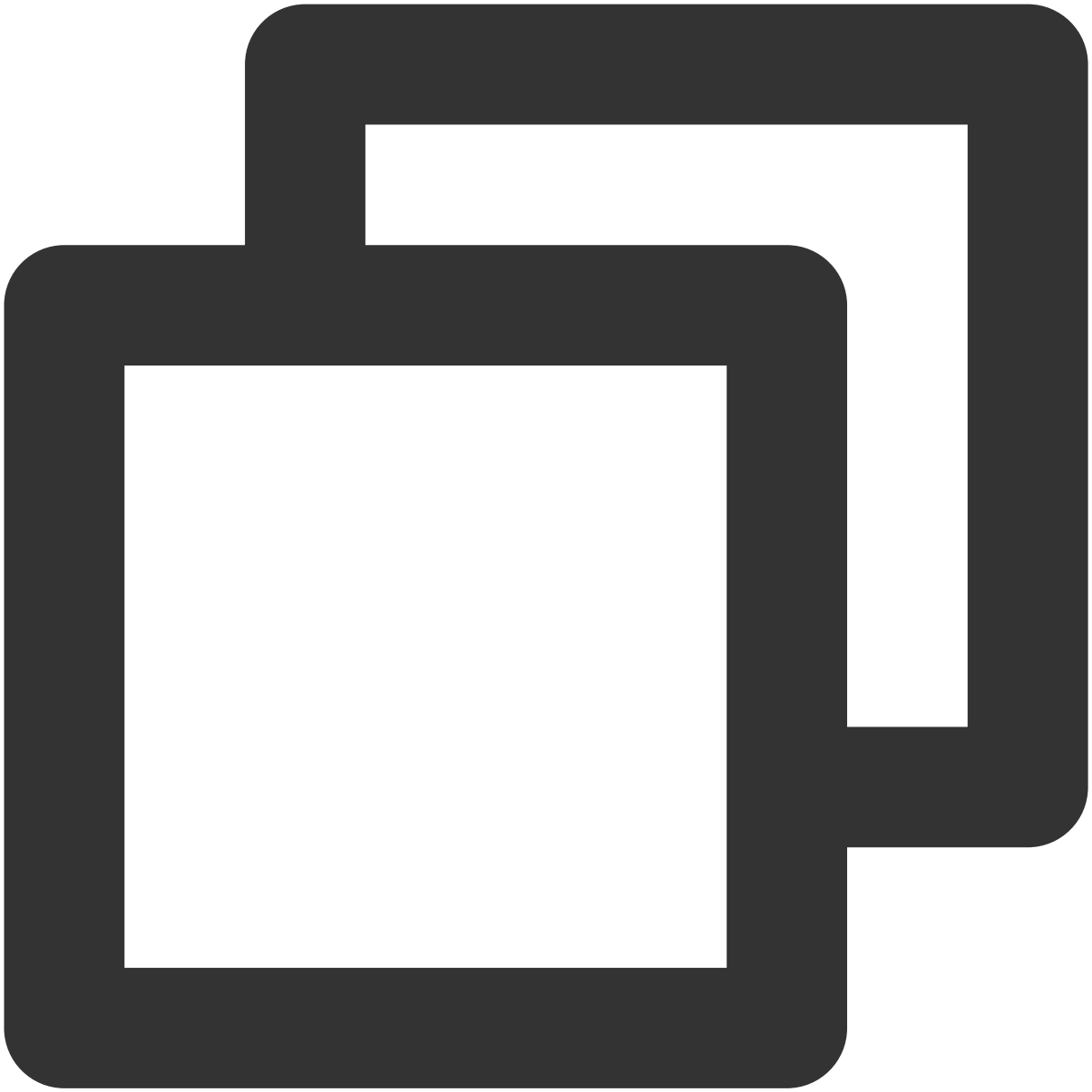


```
Page({
  viewTap: function() {
    console.log('view tap')
  }
})
```

Page.route

Path to the current page of typeString.

©2013-2022 Tencent Cloud. All rights reserved.



```
Page({
  viewTap: function() {
    console.log('view tap')
  }
})
```

Page.prototype.setData(Object data, Function callback)

setData Function is used to send data from the logical layer to the view layer (asynchronously), changing the corresponding this.data Is synchronized.

©2013-2022 Tencent Cloud. All rights reserved.

Parameter specification

field	type	Required	describe
data	Object	yes	The data to be changed this time
callback	Function	no	Interface update caused by setData after rendering

Object with key: value In the form of this.data to hit the target key The corresponding value is changed to value。among key Can be given in the form of a data path, supporting changing an item in an array or a property of an object, such as array[2].message， a.b.c.dAnd does not require this.data Predefined in.

Note：

Direct modification this.data Without calling this.setData Can not change the state of the page, but also cause data inconsistencies;

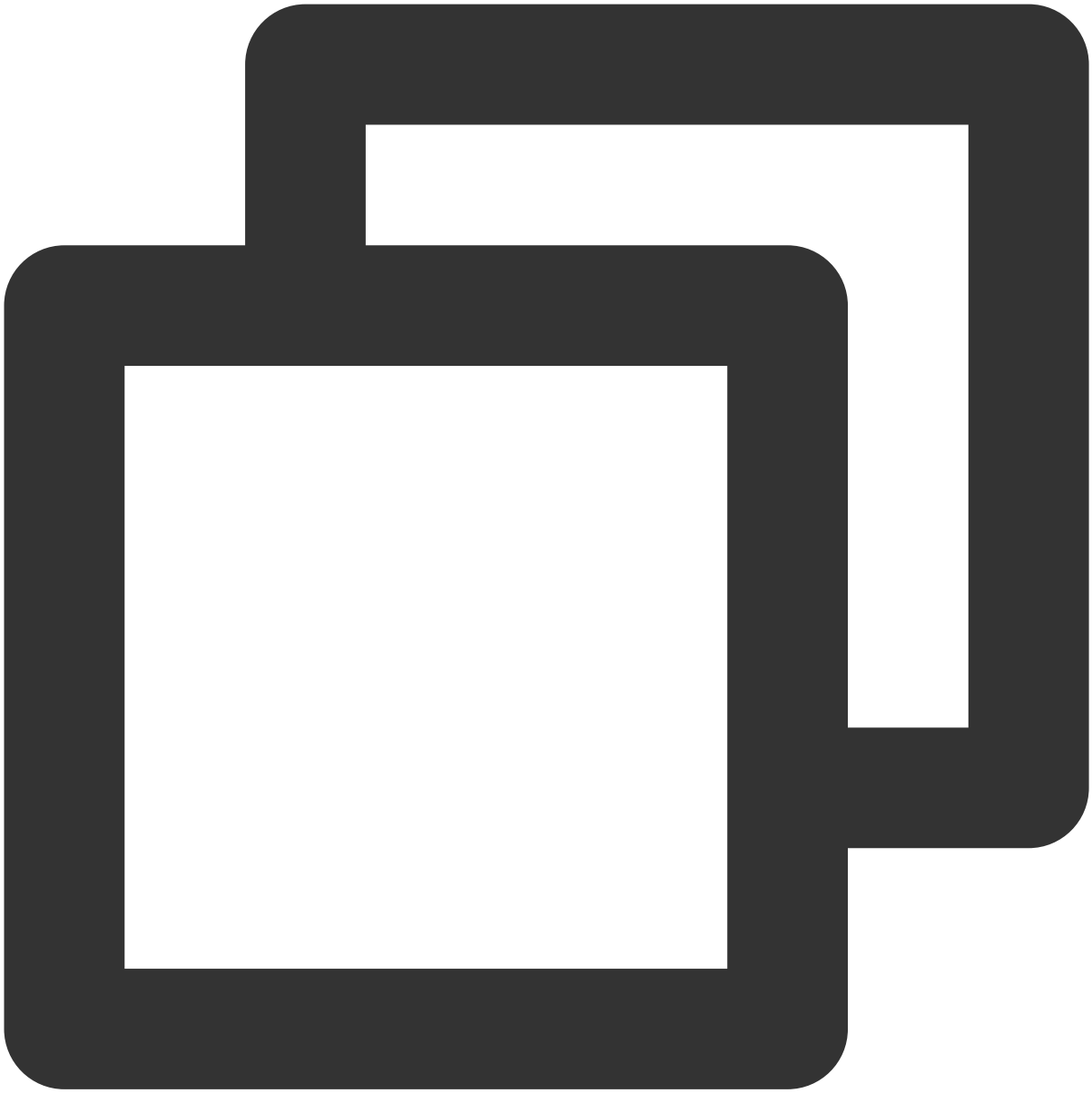
Supports settings only JSON Data;

A single set of data can not exceed 1024kB, try to avoid a set of too much data;

Please don't data Any of the value Set to undefined Otherwise this item will not be set and may leave some potential problems.

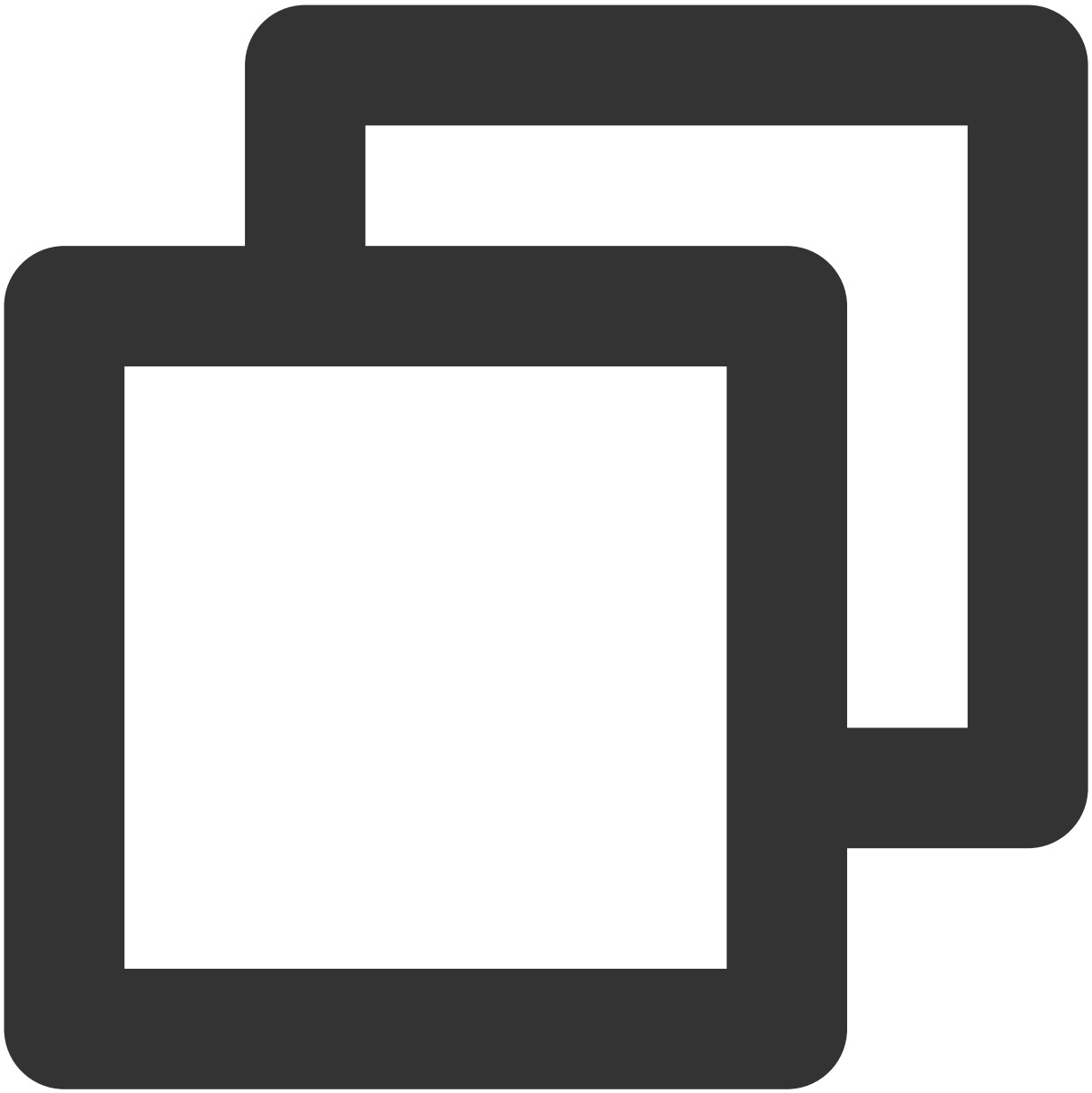
Sample code:

©2013-2022 Tencent Cloud. All rights reserved.



```
<!--index.wxml-->
<view>{{text}}</view>
<button bindtap="changeText"> Change normal data </button>
<view>{{num}}</view>
<button bindtap="changeNum"> Change normal num </button>
<view>{{array[0].text}}</view>
<button bindtap="changeItemInArray"> Change Array data </button>
<view>{{object.text}}</view>
<button bindtap="changeItemInObject"> Change Object data </button>
<view>{{newField.text}}</view>
<button bindtap="addNewField"> Add new data </button>
```

©2013-2022 Tencent Cloud. All rights reserved.



```
// index.js
Page({
  data: {
    text: 'init data',
    num: 0,
    array: [{text: 'init data'}],
    object: {
      text: 'init data'
    }
  },
```

©2013-2022 Tencent Cloud. All rights reserved.

```
  changeText: function() {
    // this.data.text = 'changed data' // Don't change this.data directly.
    // You should use setData
    this.setData({
      text: 'changed data'
    })
  },
  changeNum: function() {
    // Alternatively, you can modify this.data and then immediately set the modifie
    this.data.num = 1
    this.setData({
      num: this.data.num
    })
  },
  changeItemInArray: function() {
    // For object or array fields, you can modify a subfield directly under it, whi
    this.setData({
      'array[0].text':'changed data'
    })
  },
  changeItemInObject: function(){
    this.setData({
      'object.text': 'changed data'
    });
  },
  addNewField: function() {
    this.setData({
      'newField.text': 'new data'
    })
  }
})
```

PageObject[] getCurrentPages()

Get the current pages stack. The first element in the array is the first page and the last element is the current page.

Cautions

Don't try to modify the page stack, it will lead to routing and page state errors;

Don't call `getCurrentPages()` when `App.onLaunch` is running, the page has not been generated yet.

©2013-2022 Tencent Cloud. All rights reserved.

Custom Components

Last updated：2024-07-12 18:38:34

Component

Create a custom component that accepts a Object Parameter of type.

parameter

Object object

definition segment	type	Is it necessary to fill out	describe
properties	Object Map	no	External properties of a component are a map of property names to property settings
data	Object	no	Component's internal data, and properties Template rendering for components together
observers	Object	no	Component data field listener for listening properties and data See the changes in Data listener
methods	Object	no	Component, including event response functions and arbitrary custom methods. For the use of event response function, see Inter-component communication and events
behaviors	String Array	no	Inter-component code reuse mechanisms similar to mixins and traits, see behaviors
created	Function	no	Component lifecycle function - executes when the component instance has just been created, noting that it cannot be called at this time setData
attached	Function	no	Component life cycle function - executed when component instance enters page node tree
ready	Function	no	Component Life Cycle Function - Executed after component layout is complete
detached	Function	no	Component Lifecycle Function - Executes when component instance is removed from page node tree
relations	Object	no	Relationship definitions between components, see Inter-

©2013-2022 Tencent Cloud. All rights reserved.

			component relations
externalClasses	String Array	no	External style classes accepted by the component, see External style class
options	Object Map	no	Some options (specific option settings are covered when the features are described in the documentation, not listed here)
lifetimes	Object	no	Component lifecycle declaration object, see Component life cycle
pageLifetimes	Object	no	The lifecycle declaration object for the page on which the component resides, see Component life cycle

Generated component instances can be found in component methods, lifecycle functions, and properties `observer` Through this Access. Components contain common properties and methods.

Attribute name	type	describe
is	String	Component file path
id	String	Node id
dataset	String	Node dataset
data	Object	Component data,Including internal data and attribute values
properties	Object	Component data,Including internal data and attribute values(data Consistent)

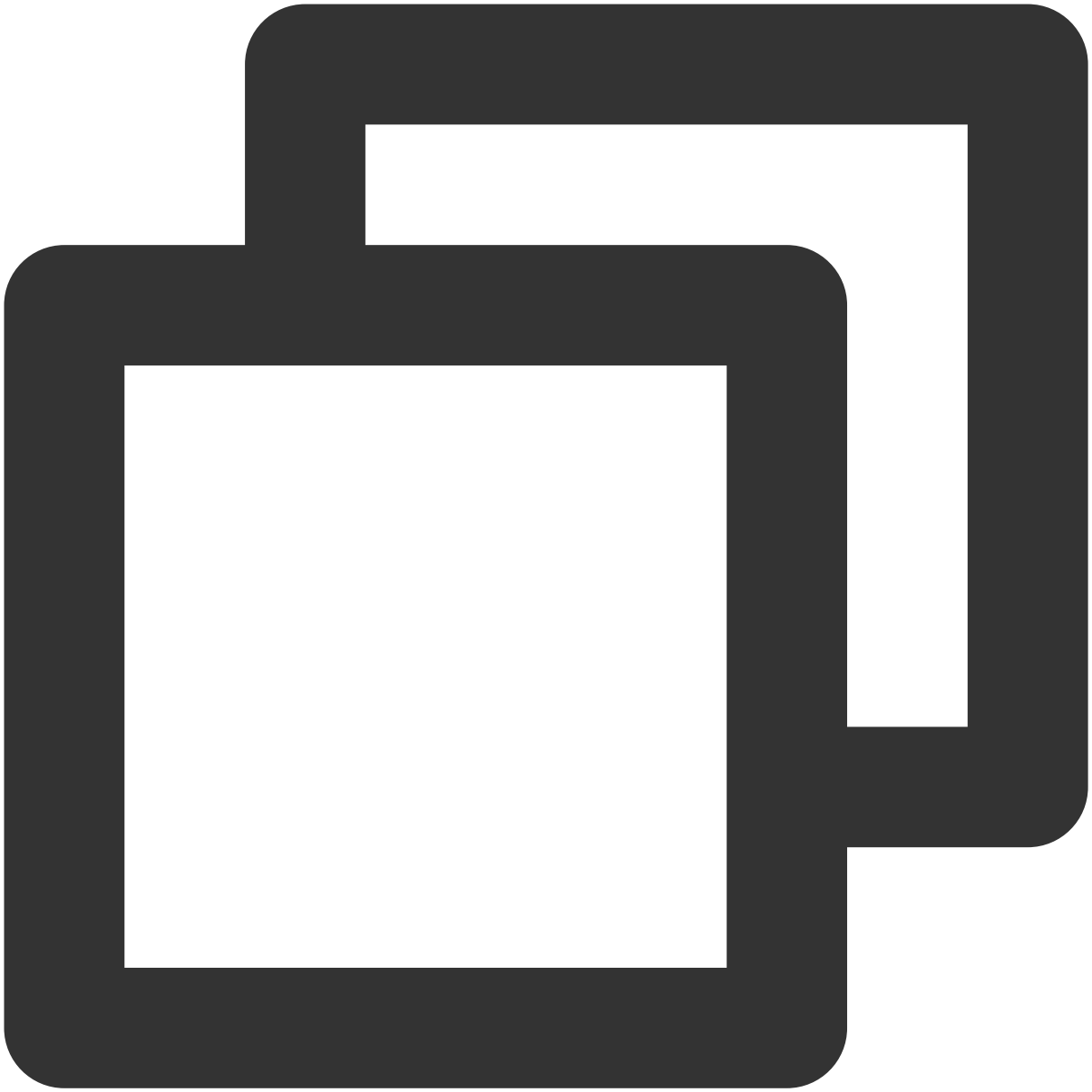
Method name	parameter	describe
setData	Object newData	Set up data and perform view-layer rendering
triggerEvent	String name, Object detail, Object options	Trigger events, see Inter-component communication and events
createSelectorQuery	-	Create a SelectorQuery Object, the selector selects from within this component instance
createIntersectionObserver	-	Create a IntersectionObserver Object, the selector selects from within this component instance
SelecComponent	String selector	Select the component instance node using the selector and return the first component instance object that matches wx://component-export Impact

©2013-2022 Tencent Cloud. All rights reserved.

selectAllComponents	String selector	Select the component instance node using the selector and return an array of matched component instance objects wx://component-export Impact
selectOwnerComponent	String selector	Selects the component instance of the current component node (that is, the reference to the component) and returns its component instance object wx://component-export Impact
getRelationNodes	String relationKey	Gets all the associated nodes corresponding to this relationship, see Inter-component relations
hasBehavior	Object `behavior`	Checks if the component has a behaviour (the check recursively checks all behaviours that have been introduced directly or indirectly).

Code example:

©2013-2022 Tencent Cloud. All rights reserved.



```
Component({

  behaviors: [],

  // Property definition (see below for details)
  properties: {
    myProperty: { // Attribute name
      type: String,
      value: ''
    },
    myProperty2: String // Simplified definition
```

©2013-2022 Tencent Cloud. All rights reserved.

```
  },

  data:  {}, // Private data, available for template rendering

  lifetimes: {
    // Life cycle function, can be a function, or a method name defined in the meth
    attached: function () { },
    moved: function () { },
    detached: function () { },
  },

  // Life cycle function, can be a function, or a method name defined in the method
  attached: function () { }, // The attached declaration here is overwritten by the
  ready: function() { },

  pageLifetimes: {
    // Life Cycle Function for the Page on which the component is located
    show: function () { },
    hide: function () { },
    resize: function () { },
  },

  methods: {
    onMyButtonTap: function(){
      this.setData({
        // The method of updating properties and data is similar to that of updatin
      })
    },
    // The internal method recommends starting with an underscore
    _myPrivateMethod: function(){
      // Here will be data.A[0].B Set to 'myPrivateData'
      this.setData({
        'A[0].B': 'myPrivateData'
      })
    },
    _propertyChange: function(newVal, oldVal) {

    }
  }

})
```

properties definition

Definition segment	type	Is it necessary to fill out	describe

©2013-2022 Tencent Cloud. All rights reserved.

type	-	yes	Type of attribute
value	-	no	The initial value of the
observer	Function	no	Callback function when attribute value changes

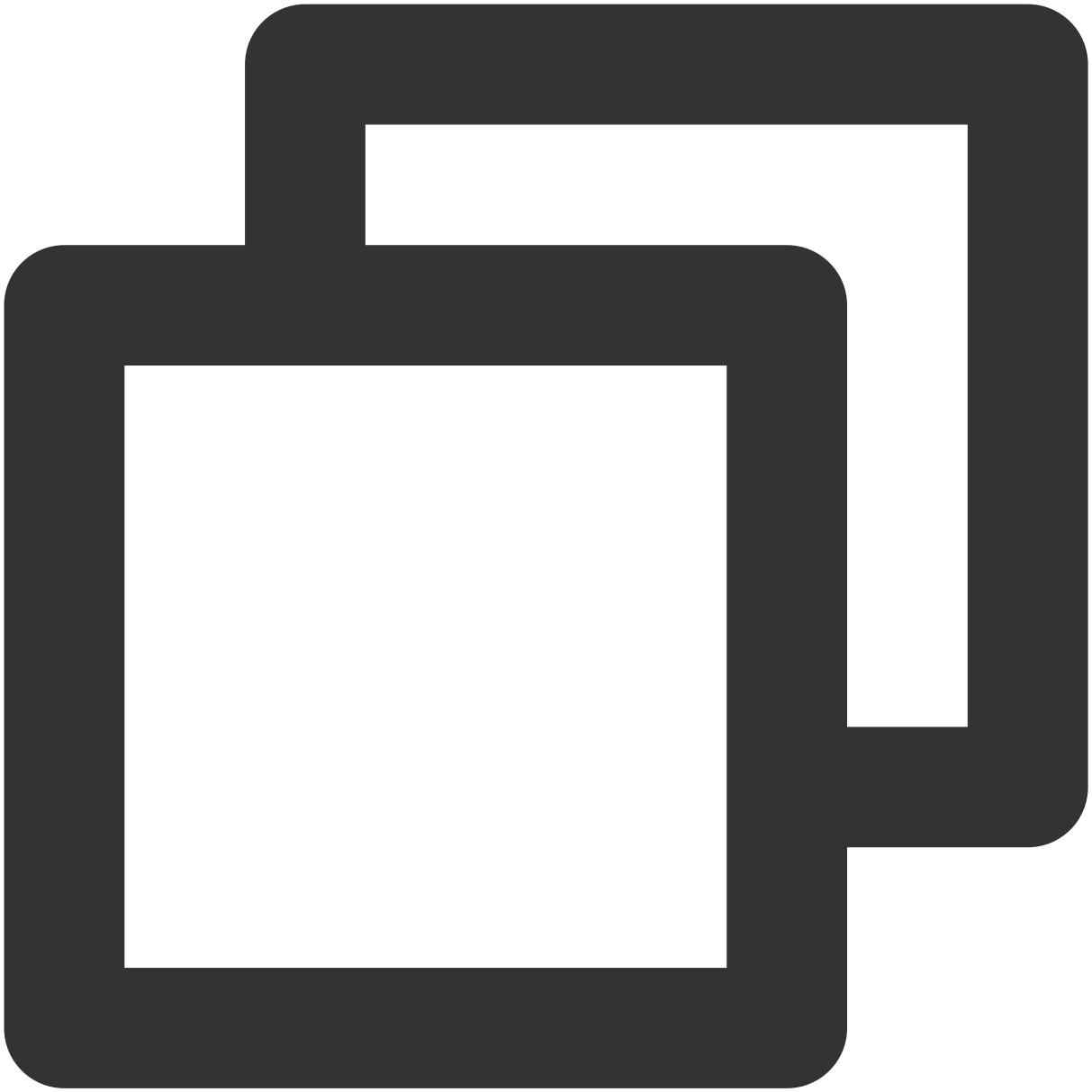
Property values can be changed using the observer To monitor. Currently, this field is not recommended in the new version of the base library; instead, use the Component Constructor observers Field, which is more powerful and better performance.

Note：

The type field in the definition field is mandatory.

Code example:

©2013-2022 Tencent Cloud. All rights reserved.



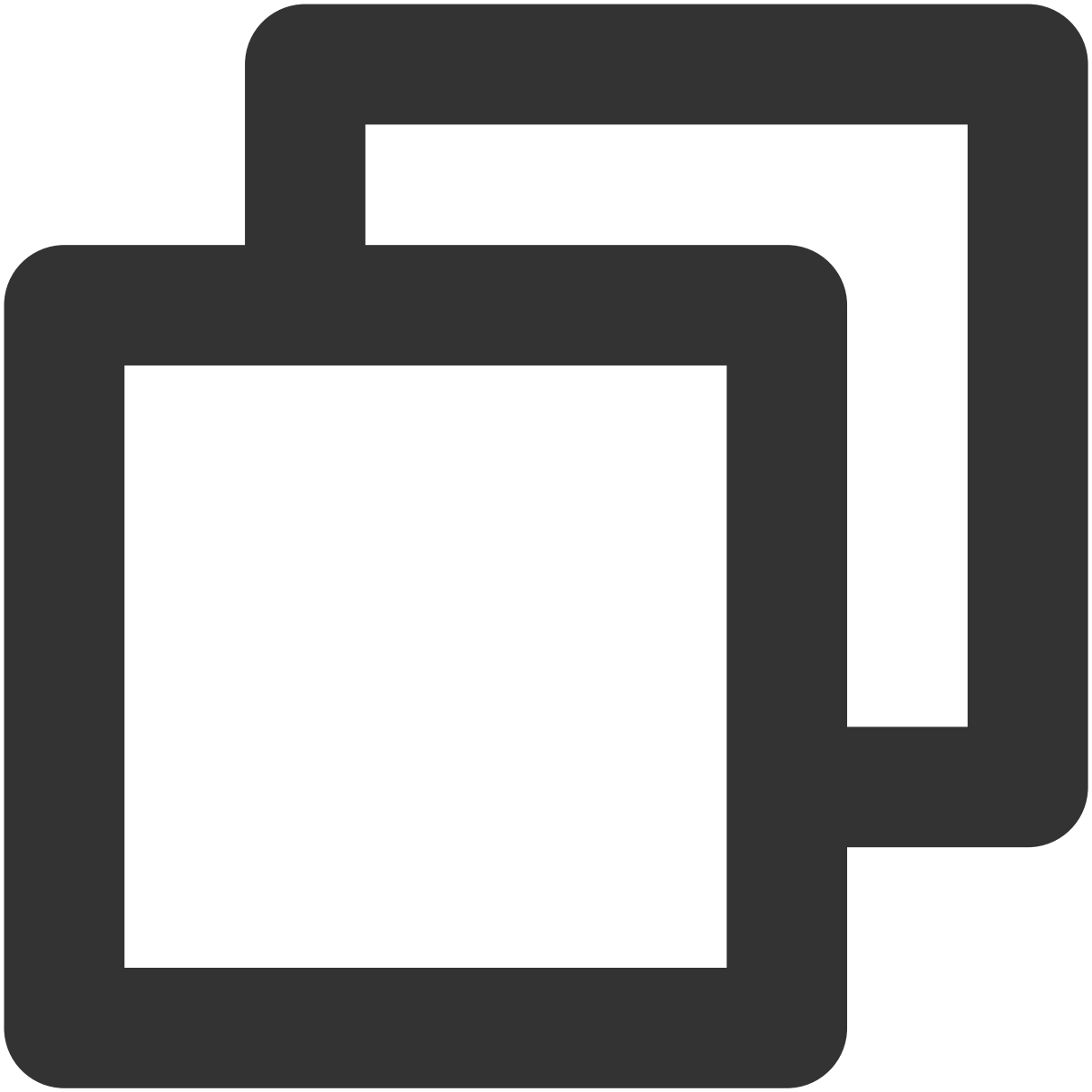
```
Component({
  properties: {
    min: {
      type: Number,
      value: 0
    },
    min: {
      type: Number,
      value: 0,
      observer: function(newVal, oldVal) {
        // When property values change
```

©2013-2022 Tencent Cloud. All rights reserved.

```
      }
    },
    lastLeaf: {
      // This property can be Number 、 String 、 Boolean One of three types.
      type: Number,
      optionalTypes: [String, Object],
      value: 0
    }
  }
})
```

Property can be of type String Number Boolean Object Array One, can also be null Represents an unrestricted type. In most cases, it is best to specify an exact type for a property. So, in... WXML When an attribute value is specified literarily in, the value can obtain an exact type, such as:

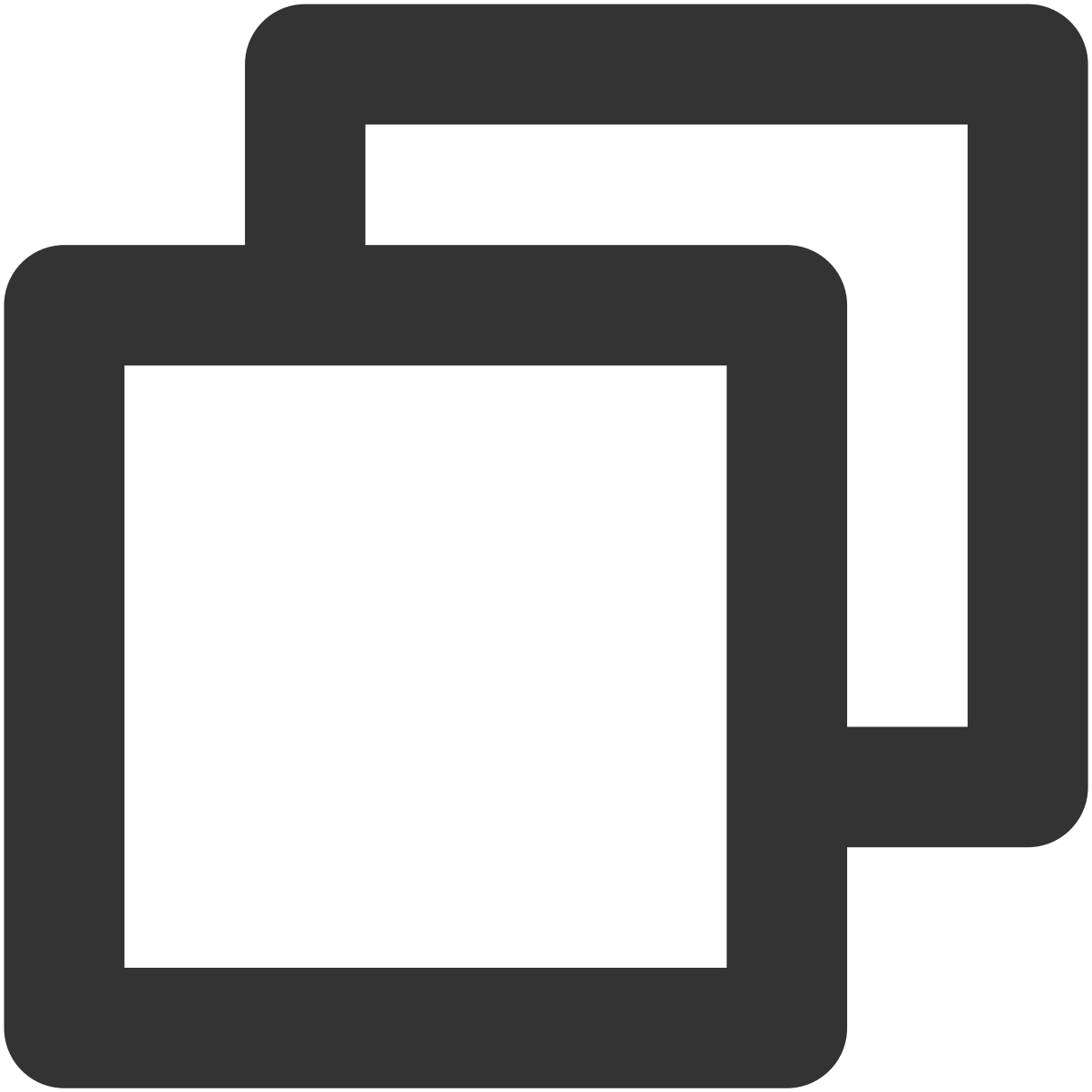
©2013-2022 Tencent Cloud. All rights reserved.



```
<custom-comp min="1" max="5" />
```

At this point, because the corresponding property of the custom component is specified as Number Type, min and Max Is going to be assigned to 1 and 5 Instead of "1" and "5" , namely:

©2013-2022 Tencent Cloud. All rights reserved.



```
this.data.min === 1 //
truethis.data.Max === 5 // true
```

Tips:

Use this.data Can get internal data and attribute valuesBut modifying it directly does not apply changes to the interface, and you should use the setData Modify;
Lifecycle functions cannot be passed through component methods this Access to;

©2013-2022 Tencent Cloud. All rights reserved.

Property names should be avoided with data Beginning, that is, do not name dataXyz This form, because in WXML In, data-xyz="" Will be used as nodes dataset Instead of component properties;
In the definition and use of a component, the component's attribute name and data Fields cannot conflict with each other (although they are in different definition segments).

Behavior

Register a behaviorAccept one. Object Parameter of type.

parameter

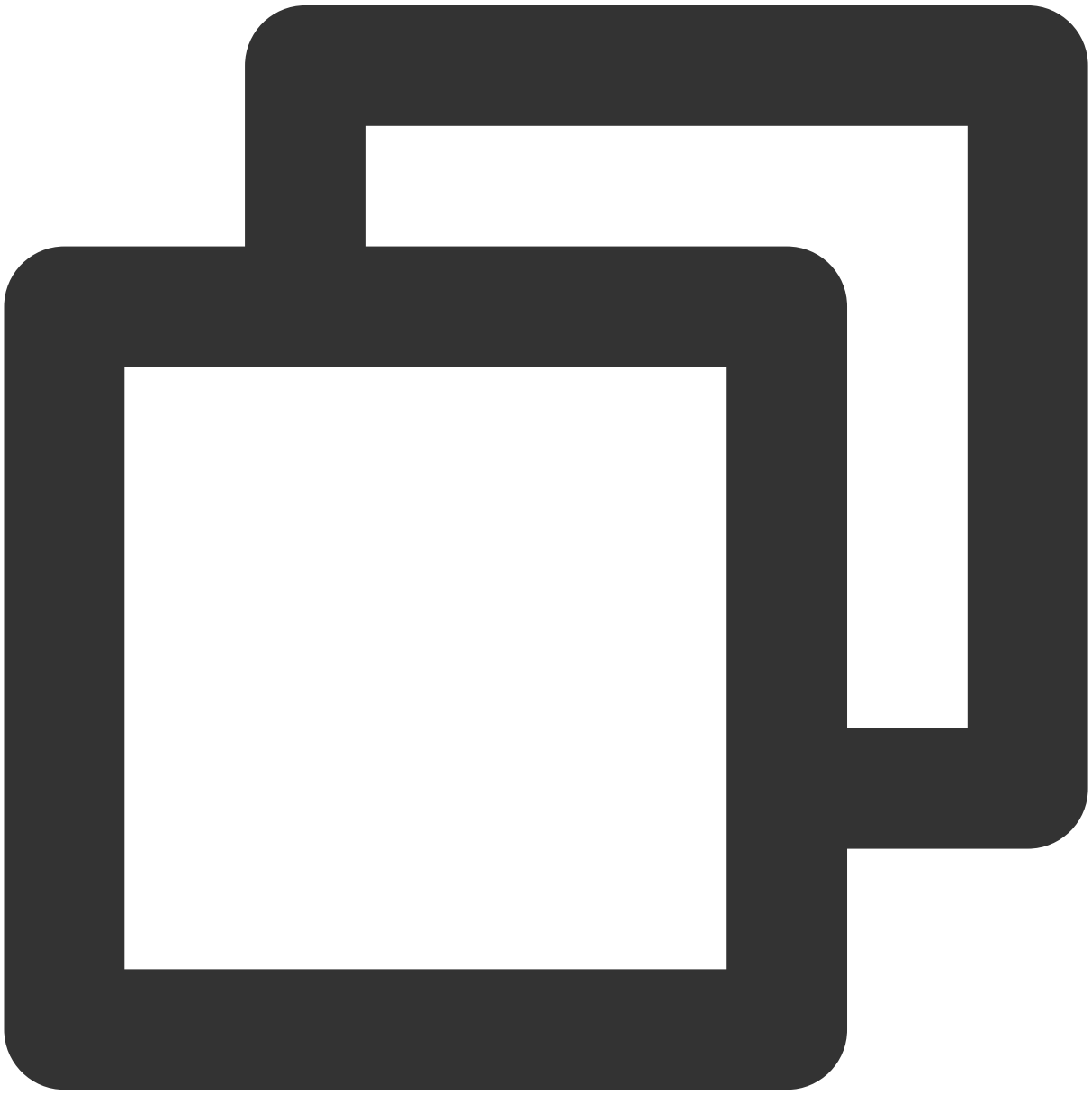
Object object

Definition segment	type	Is it necessary to fill out	describe
properties	Object Map	No	Properties of the same component
data	Object	No	Properties of the same component
methods	Object	No	Same method as custom components
behaviors	String Array	No	For a mechanism for reusing code between components similar to mixins and traits, see behaviors.
created	Function	No	Component Lifecycle Functions - executed when the component instance has just been created, note that setData cannot be called at this time.
attached	Function	No	Component Lifecycle Functions - executed when the component instance enters the page node tree
ready	Function	No	Component Lifecycle Functions - executed after the component layout is complete
detached	Function	No	Component Lifecycle Functions - executed when the component instance is removed from the page node tree
relations	Object	No	For definitions of inter-component relationships, see Inter-component

©2013-2022 Tencent Cloud. All rights reserved.

			Relationships
lifetimes	Object	No	Component Lifecycle Declaration Object, see Component Lifecycle
pageLifetimes	Object	No	The lifecycle declaration object for the page on which the component resides, see Component Lifecycle

Code example:



©2013-2022 Tencent Cloud. All rights reserved.

```
// my-behavior.js
module.exports = Behavior({
  behaviors: [],
  properties: {
    myBehaviorProperty: {
      type: String
    }
  },
  data: {
    myBehaviorData: {}
  },
  attached: function(){},
  methods: {
    myBehaviorMethod: function(){}
  }
})
```

©2013-2022 Tencent Cloud. All rights reserved.

modularity

Last updated：2024-07-12 17:09:41

require

Introduces the module. Returns the interface exposed by the module via module.exports or exports.
If you need to introduce a module from another subpackage, you can configure the callback function to fetch the module asynchronously. If the asynchronous fetch fails, the error callback function will be triggered.

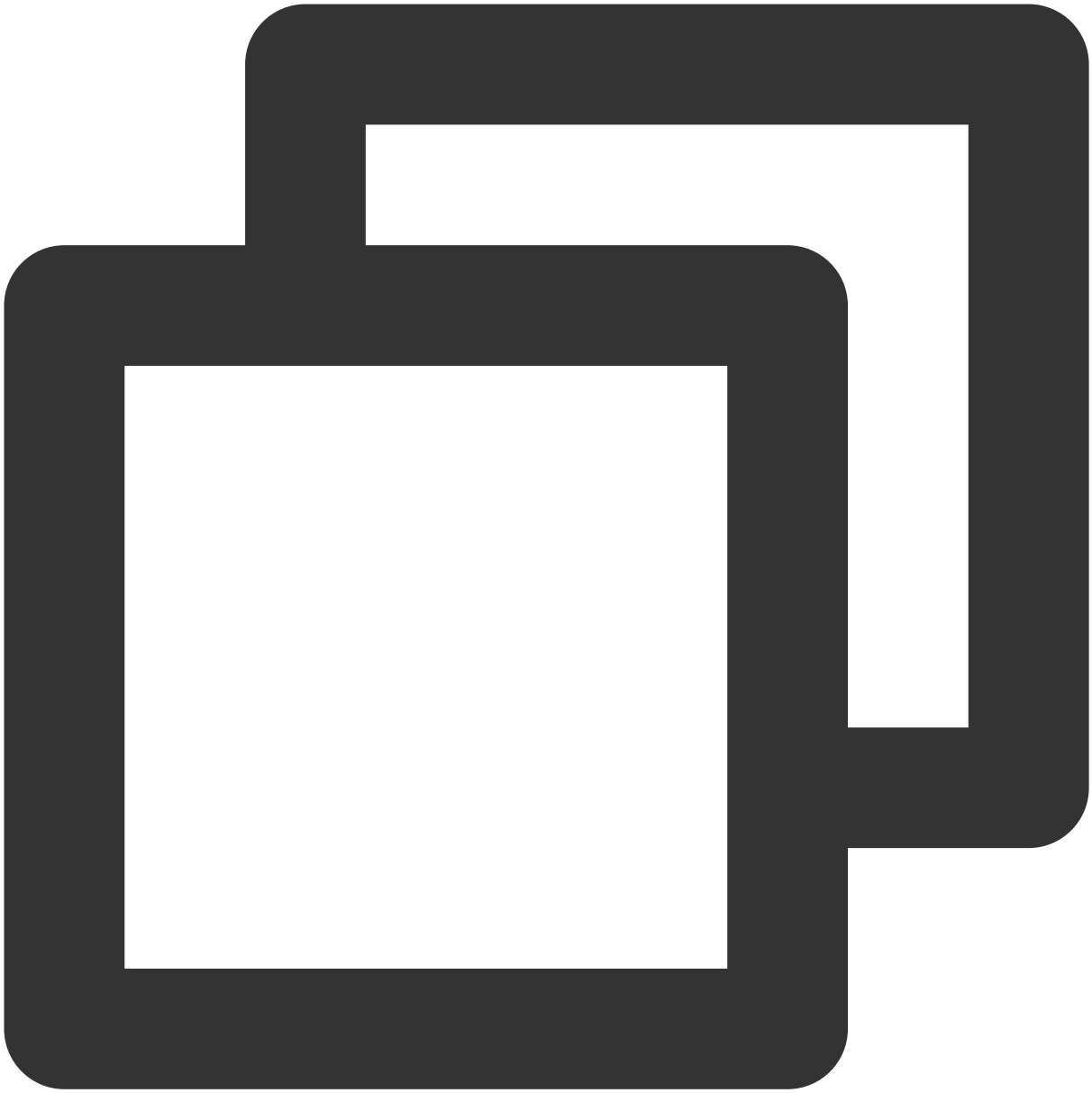
parameter

Name	type	mandatory field	Introductions	Note
path	string	yes	You need to introduce the relative path of the module file relative to the current file, or the npm module name, or npm modules path. Absolute path is not supported	You need to introduce a relative path to the module file relative to the current file, or the npm module name, or the npm module path. Absolute paths are not supported by default, you can customise the path mapping by configuring resolveAlias.
callback	function	no	Asynchronous loading success callback function, the callback function parameters for the successful loading of the module	Asynchronous loading success callback function, the callback function parameter is the successfully loaded module.
error	function	no	Asynchronous load failure callback function, the parameters of this callback function are the error message and module name	Asynchronous load failure callback function, the parameters of this callback function are error message and module name.

require.async Chained calls

It can be used by chaining calls.

©2013-2022 Tencent Cloud. All rights reserved.

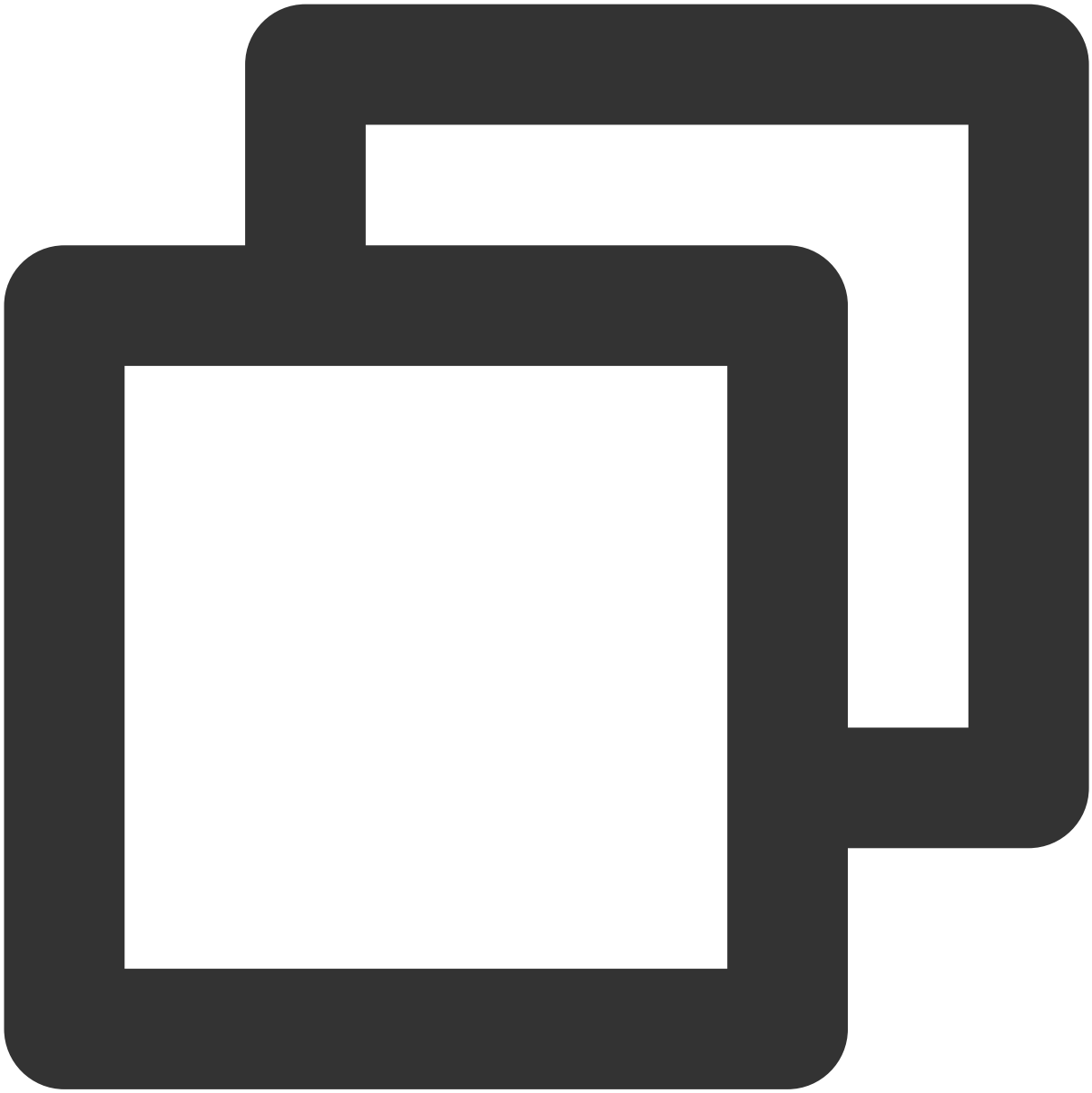


```
require
    .async('path/to/mod')
    .then((mod) => {
        console.log(mod)
    })
    .catch(({ errMsg, mod }) => {
        console.error(`path: ${mod}, ${errMsg}`)
    })
```

Sample Code

©2013-2022 Tencent Cloud. All rights reserved.

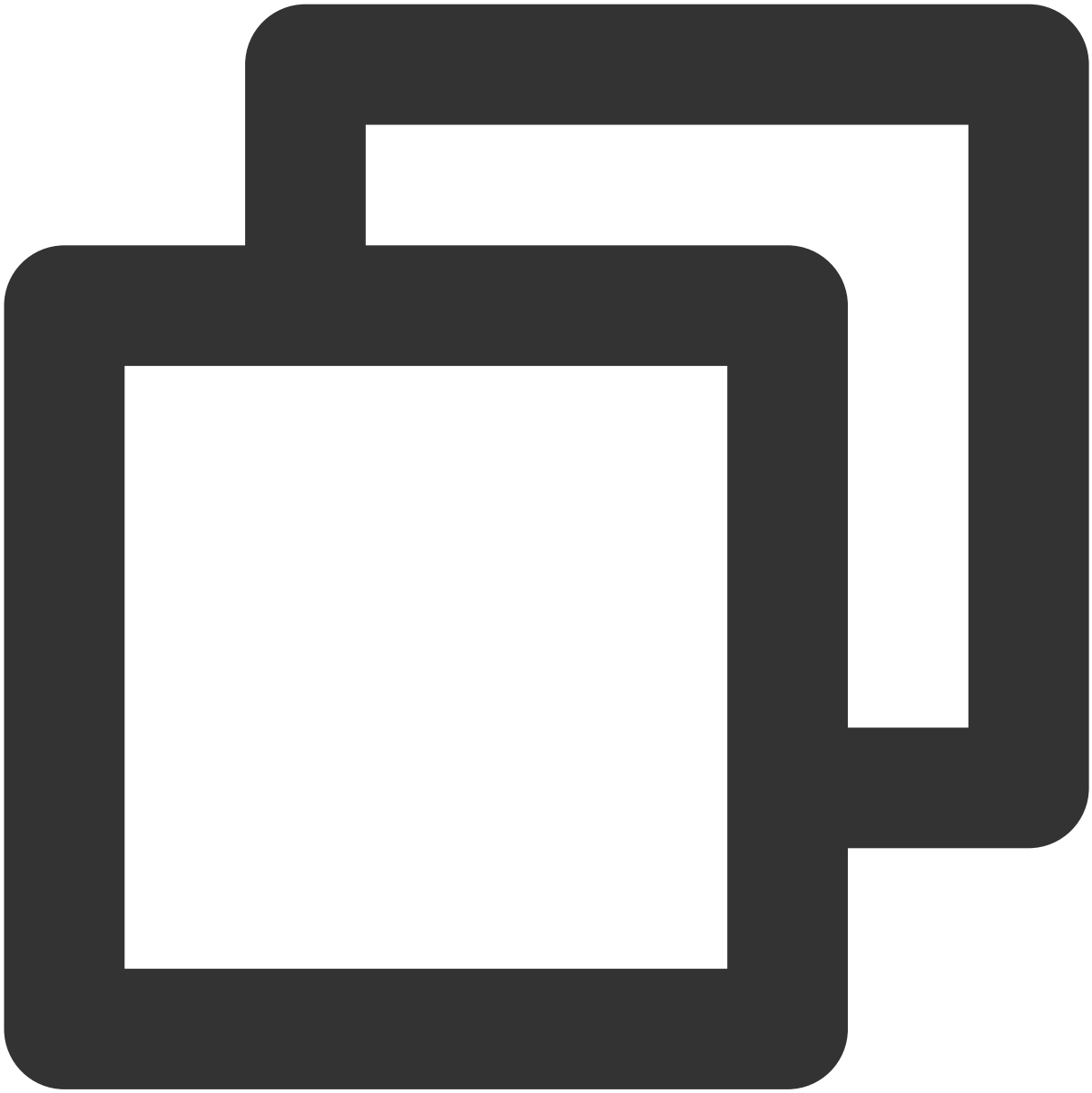
Calls within the same package



```
// common.js
function sayHello(name) {
  console.log(`Hello ${name} !`)
}
function sayGoodbye(name) {
  console.log(`Goodbye ${name} !`)
}
```

©2013-2022 Tencent Cloud. All rights reserved.

```
module.exports.sayHello = sayHello
exports.sayGoodbye = sayGoodbye
```

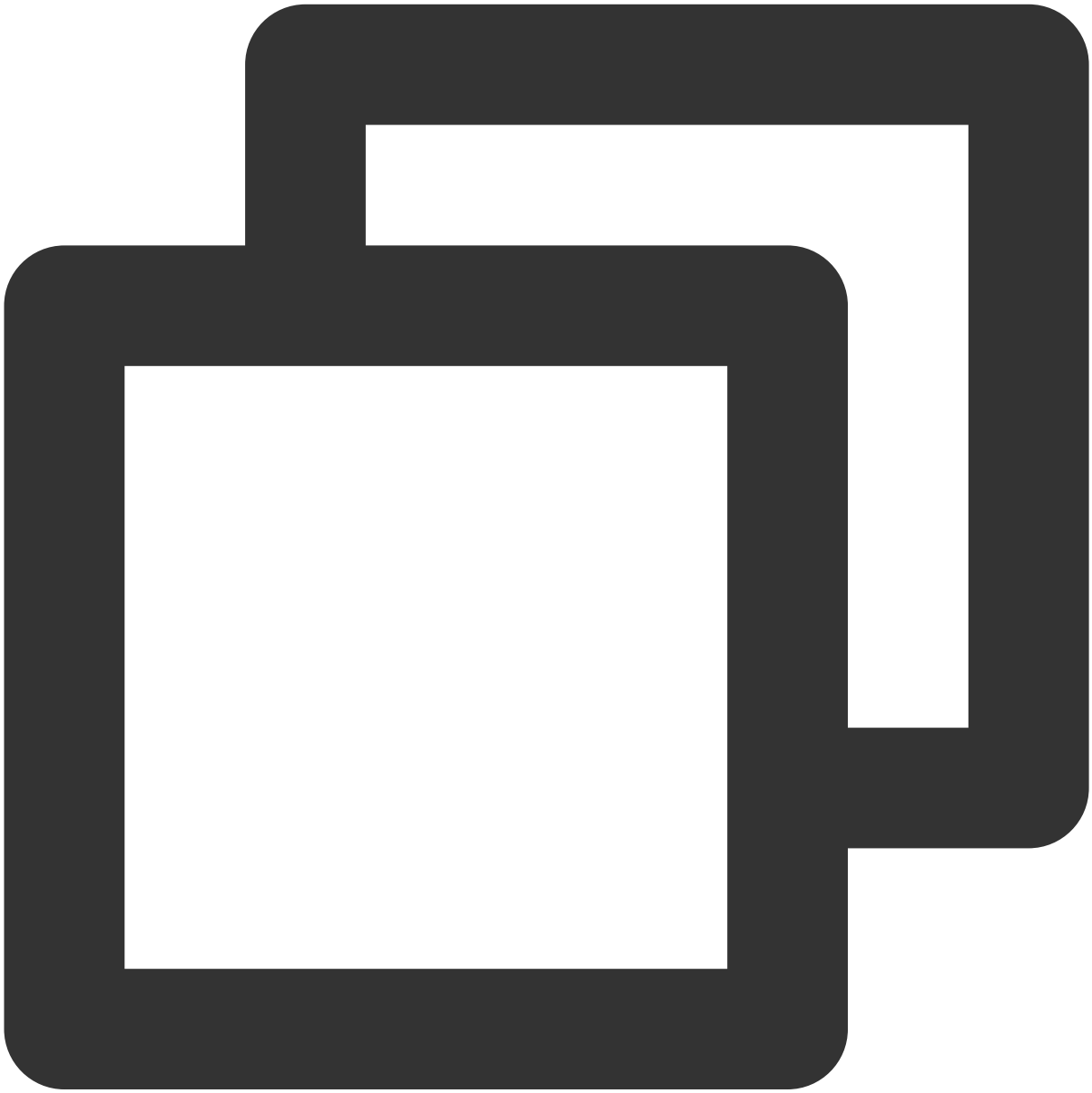


```
var common = require('common.js')
Page({
  helloMINA: function() {
    common.sayHello('MINA')
  },
  goodbyeMINA: function() {
    common.sayGoodbye('MINA')
  }
```

©2013-2022 Tencent Cloud. All rights reserved.

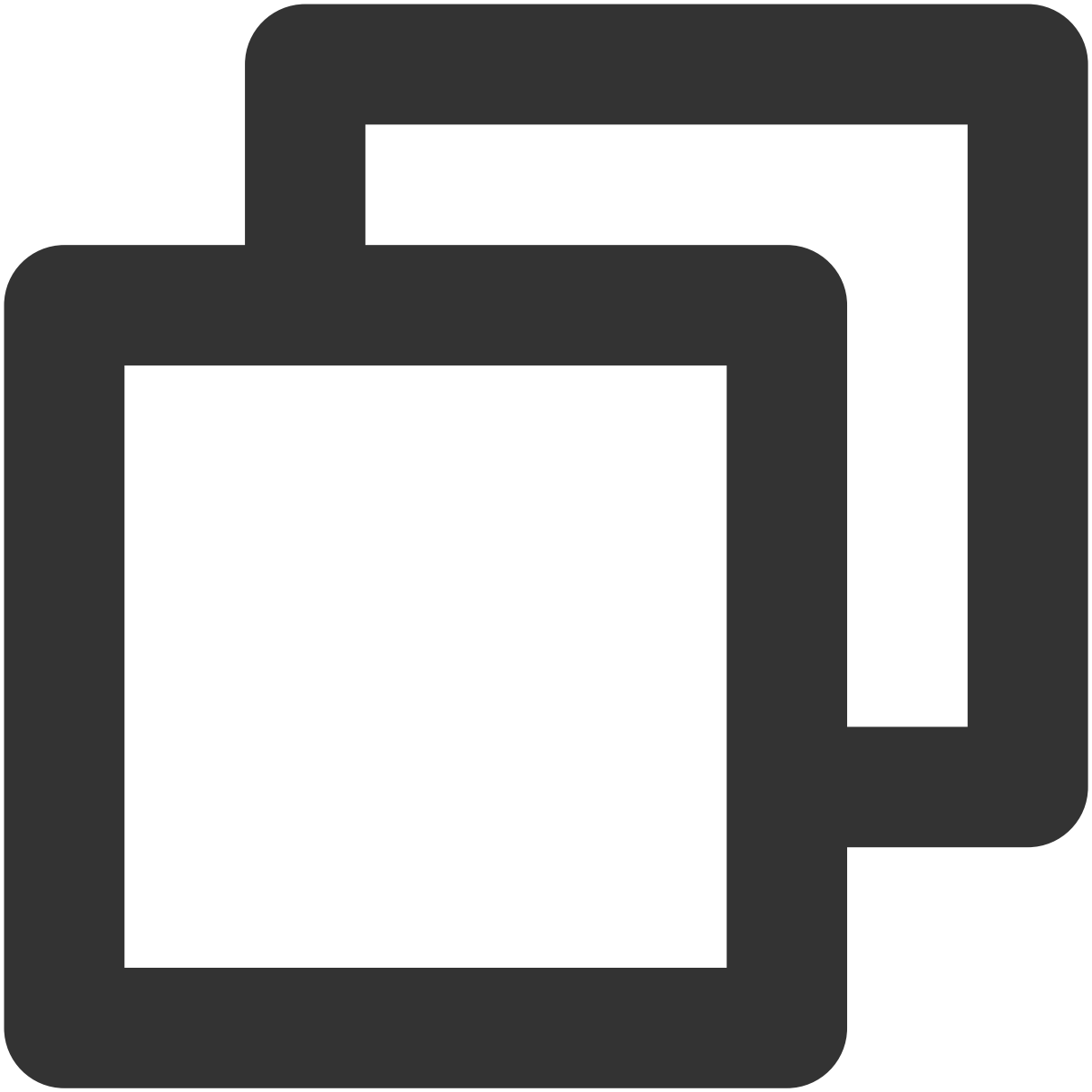
```
})
```

Asynchronous Calls Across Subpackages



```
// subpackage/common.js Subcontracting common files
export const sayHello = () => console.log("hello")
```

©2013-2022 Tencent Cloud. All rights reserved.



```
// pages/index.js main package page
let common;
require('../../subpackage/common.js', (mod) => {
    common = mod
}, ({ errMsg, mod }) => {
    console.error(`path: ${mod}, ${errMsg}`)
})

Page({
    sayHello() {
        common && common.sayHello()
```

©2013-2022 Tencent Cloud. All rights reserved.

```
    }
})
```

module

Object module

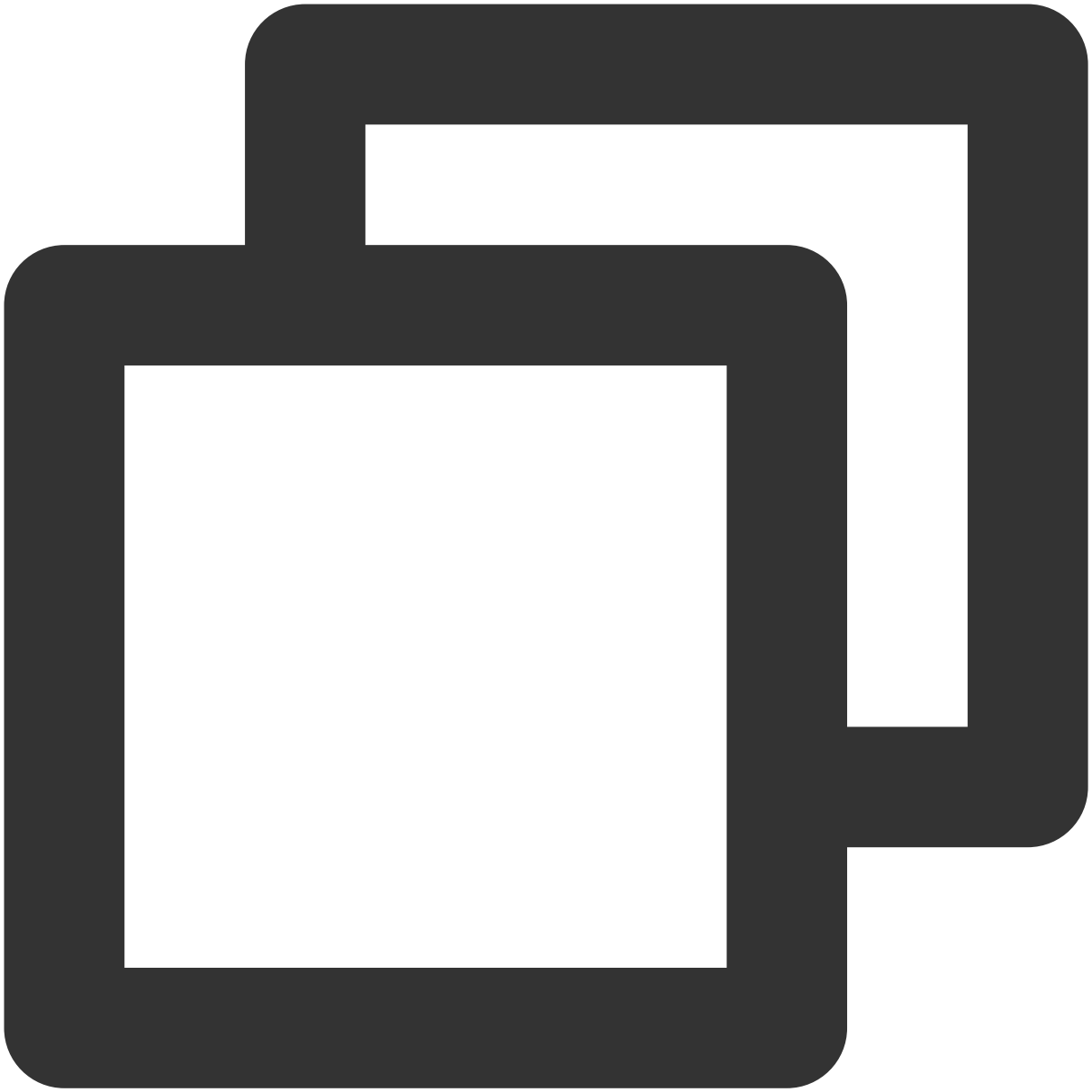
Current Module Object.

attribute

attribute	type	Introductions
exports	Object	Module exposes objects using therequireWhen referencing a module, you can get the

sample code

©2013-2022 Tencent Cloud. All rights reserved.



```
// common.js
function sayHello(name) {
  console.log(`Hello ${name} !`)
}
function sayGoodbye(name) {
  console.log(`Goodbye ${name} !`)
}


module.exports.sayHello = sayHello
exports.sayGoodbye = sayGoodbye
```

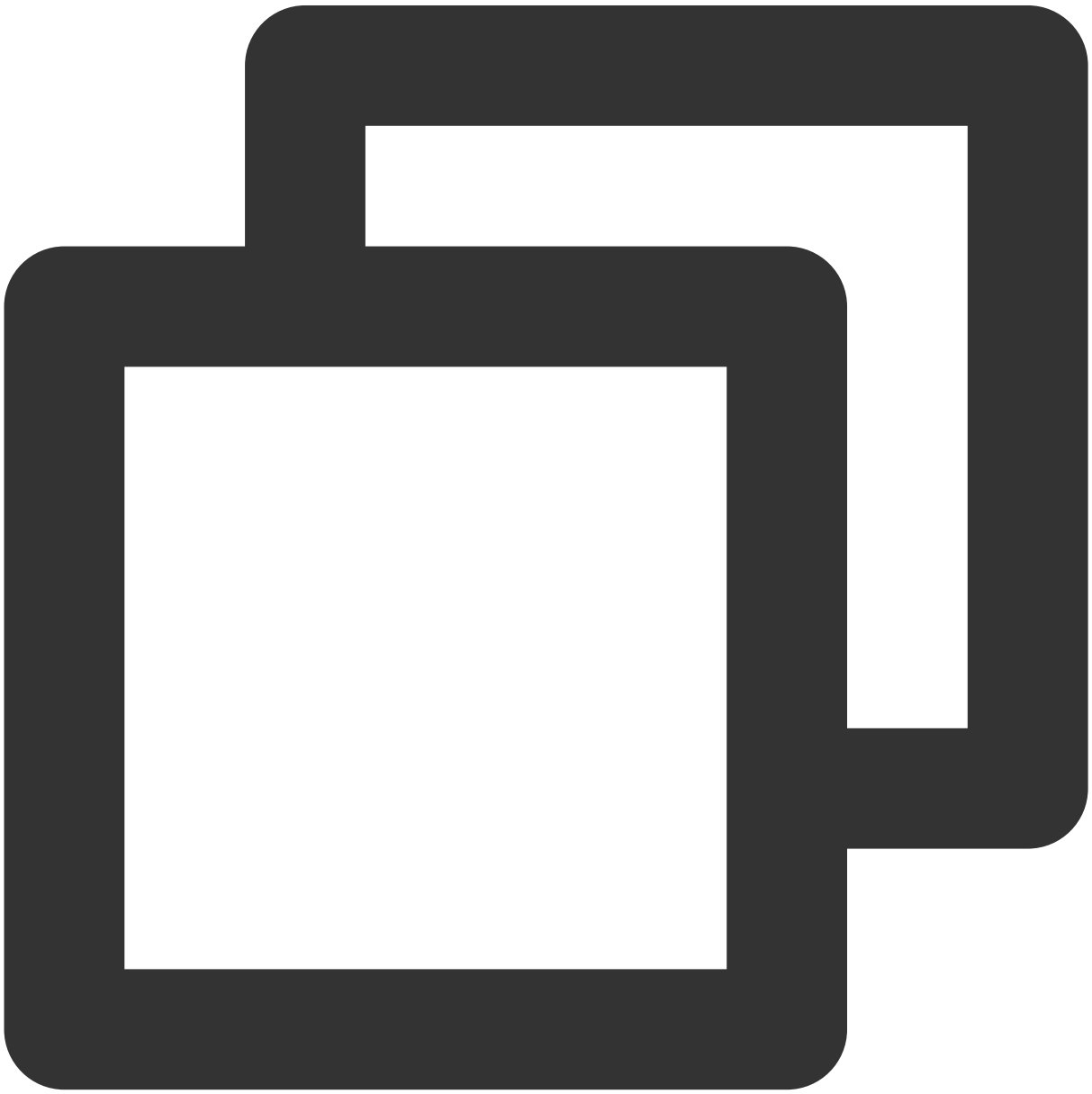
©2013-2022 Tencent Cloud. All rights reserved.

exports

Object exports

module.exports A reference to

sample code



©2013-2022 Tencent Cloud. All rights reserved.

```
// common.js
function sayHello(name) {
  console.log(`Hello ${name} !`)
}
function sayGoodbye(name) {
  console.log(`Goodbye ${name} !`)
}


module.exports.sayHello = sayHello
exports.sayGoodbye = sayGoodbye
```

©2013-2022 Tencent Cloud. All rights reserved.

Basic function

Last updated：2024-07-12 17:08:44

console

Print the log to the debug panel. console Is a global object that can be accessed directly. In the WeChat client, to vConsole Output log in.

method

console.debug()
Print to Debug Panel debug journal;
console.log()
Print to Debug Panel log journal;
console.info()
Print to Debug Panel info journal;
console.warn()
Print to Debug Panel warn journal;
console.error()
Print to Debug Panel error journal;
console.group(string label)
Create a new grouping in the debug panel. Any subsequent output is indented to indicate that it belongs to the current group. call console.groupEnd after the group;
console.groupEnd()
Ending by console.group Group created.

Be careful

Due to the limited functionality of vConsole and the variations in support for console methods across clients, it is recommended that developers only use the methods provided in this document in their mini programs.

Timer

setTimeout

This API uses number setTimeout(function callback, number delay, any rest)

©2013-2022 Tencent Cloud. All rights reserved.

Function: Set a timer. Execute the registered callback function after the timer expires.
Parameters and description:
function callback, callback function;
number delay, the delay time, the function will be called after the delay, in ms;
any rest, param1, param2, ... , paramN, etc., which are passed as arguments to the callback function.
Return value: number, the number of the timer, this value can be passed to clearTimeout to cancel the timer.

clearTimeout

> The API method is clearTimeout(number timeoutID).

Function: cancels the timer set by setTimeout.
Arguments and description: number timeoutID, the ID of the timer to be cancelled.

setInterval

> The API uses number setInterval(function callback, number delay, any rest)

Function: Set a timer. Execute the registered callback function according to the specified period (in milliseconds).
Parameters and description:
function callback, the callback function;
number delay, the time interval between the execution of the callback function, in ms;
any rest, param1, param2, ... , paramN and so on, which will be passed as parameters to the callback function.
Return value: number, the number of the timer, this value can be passed to clearInterval to cancel the timer.

clearInterval

> The API method is clearInterval(number intervalID)

Function: cancels the timer set by setInterval.
Parameters and description: number intervalID, the ID of the timer to be cancelled.

©2013-2022 Tencent Cloud. All rights reserved.

WXML Syntax Reference

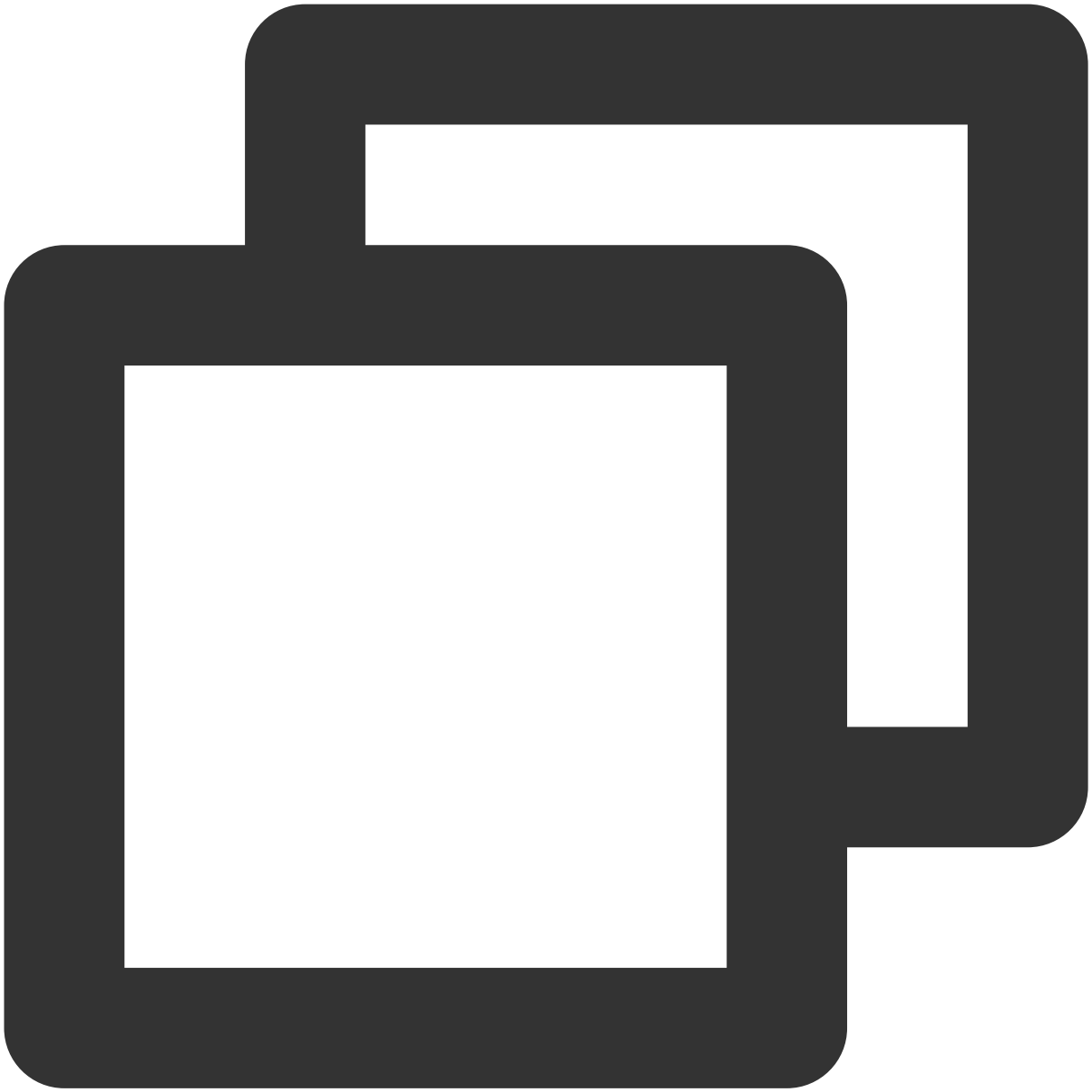
Introduce

Last updated：2024-07-12 18:53:13

WXML is a markup language for framework design. It can be used to build the page structure when combined with API Overview and an event system.

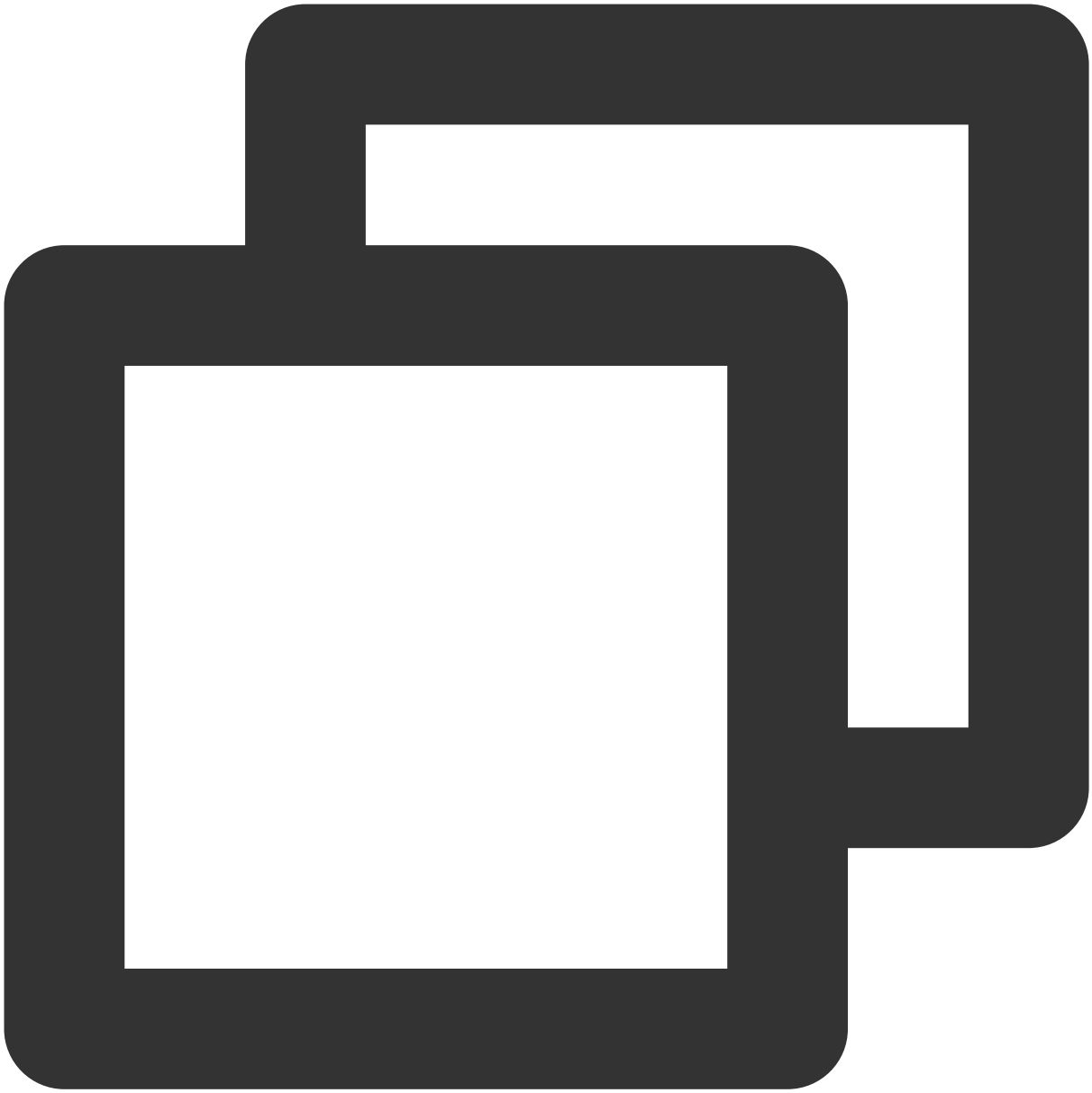
Data Binding

©2013-2022 Tencent Cloud. All rights reserved.



```
<!--wxml-->
<view> {{message}} </view>
```

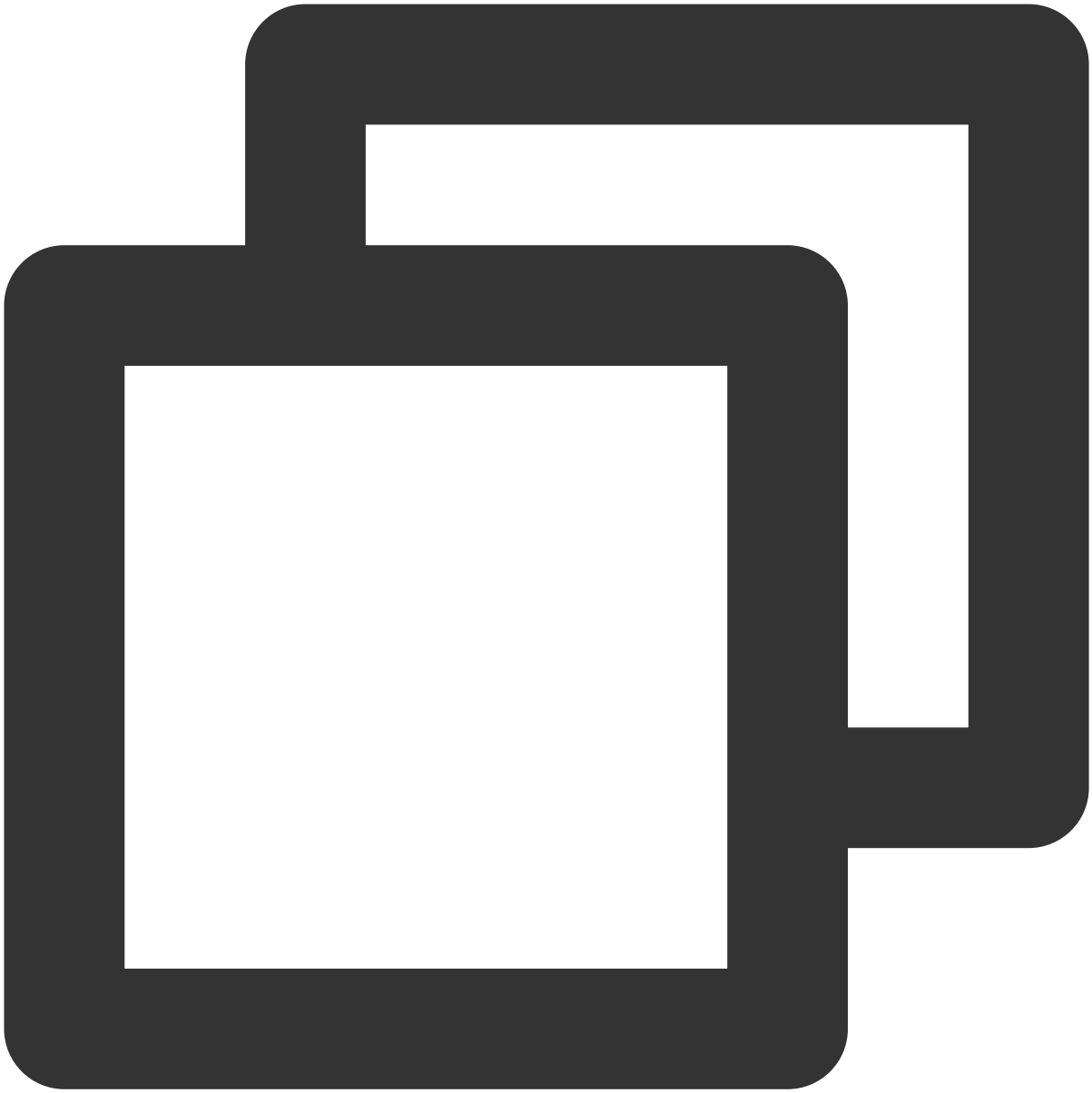
©2013-2022 Tencent Cloud. All rights reserved.



```
// page.js
Page({
  data: {
    message: 'Hello MINA!'
  }
})
```

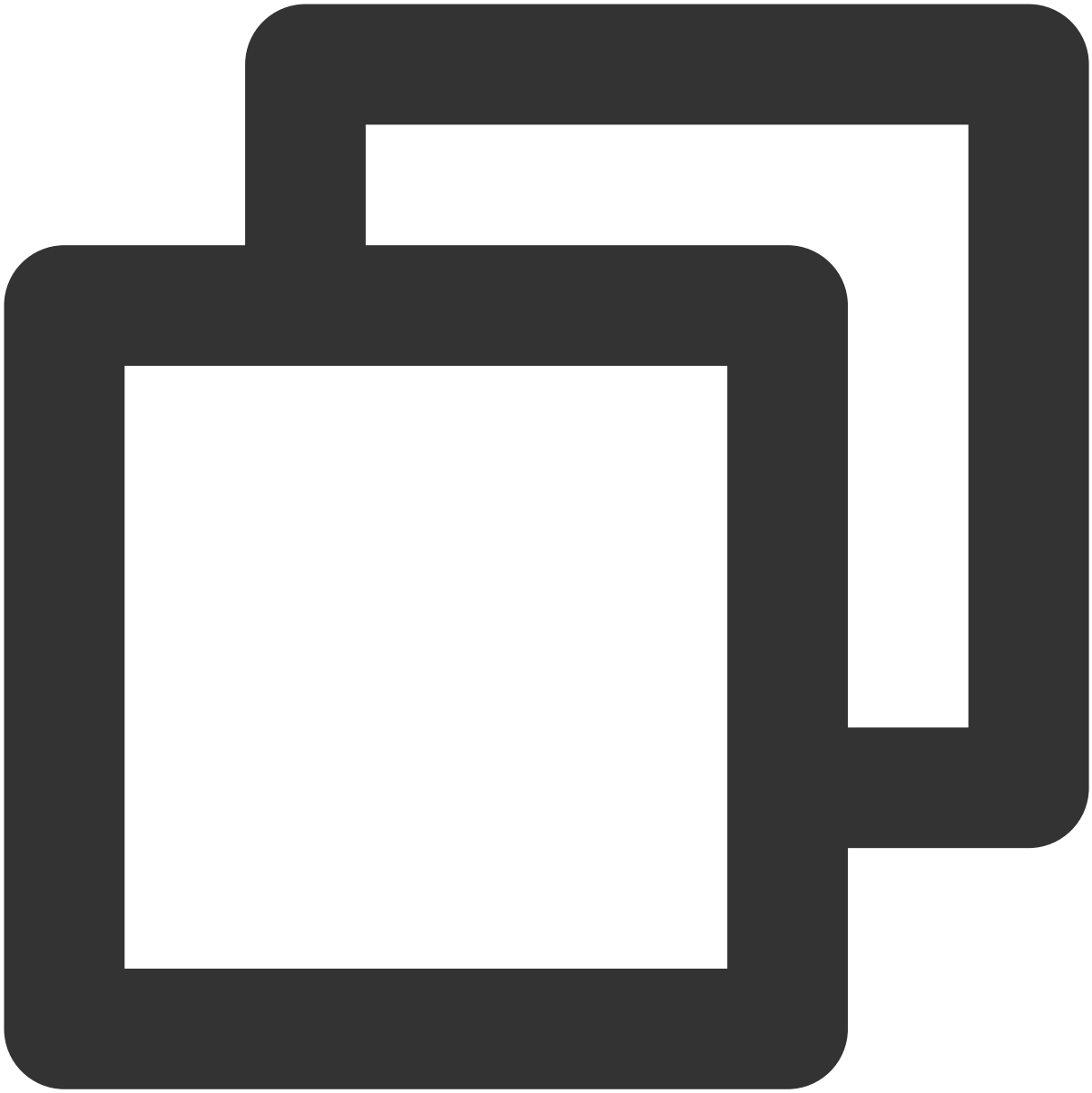
List Rendering

©2013-2022 Tencent Cloud. All rights reserved.



```
<!--wxml-->
<view wx:for="{{array}}"> {{item}} </view>
```

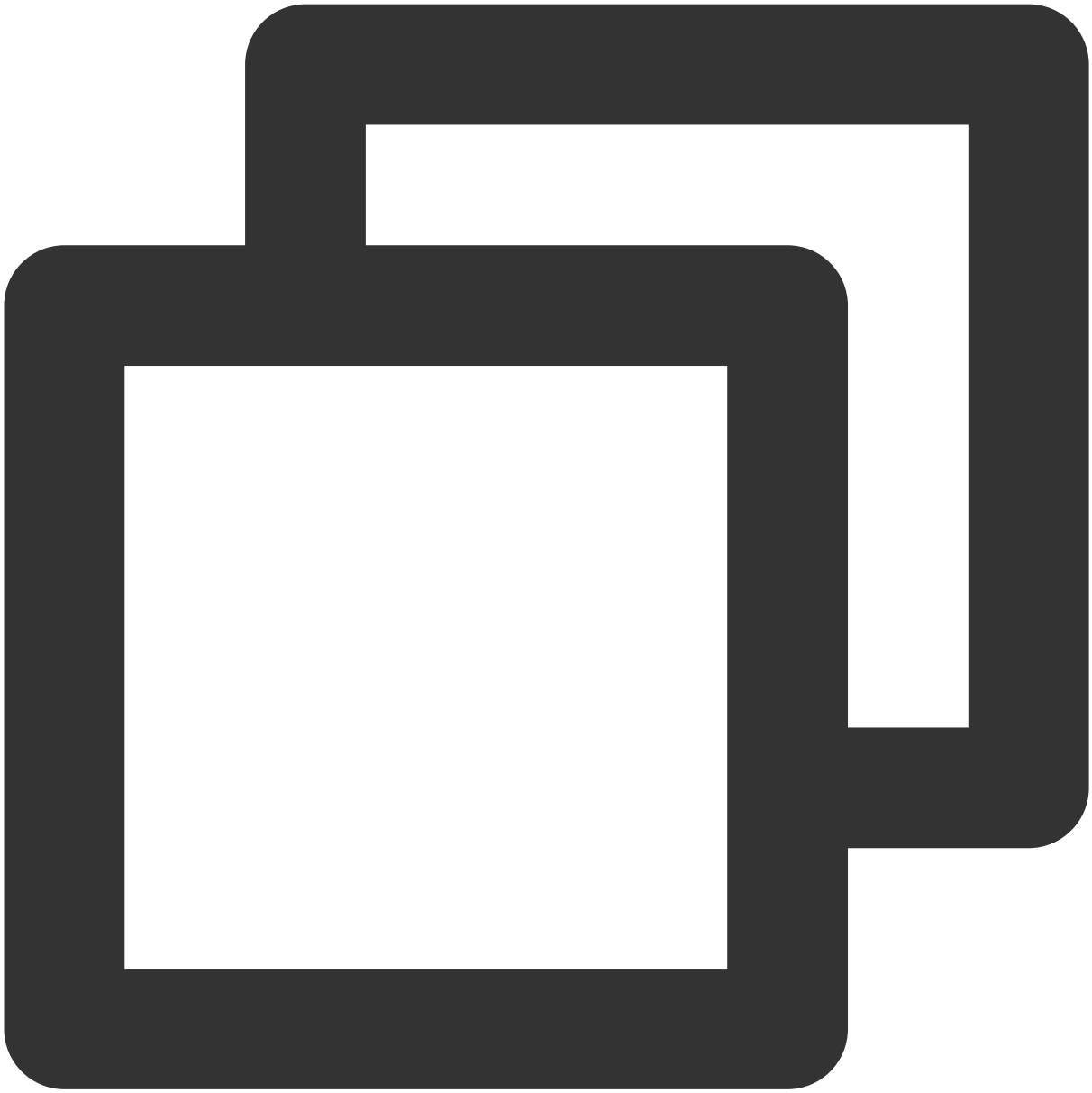
©2013-2022 Tencent Cloud. All rights reserved.



```
// page.js
Page({
  data: {
    array: [1, 2, 3, 4, 5]
  }
})
```

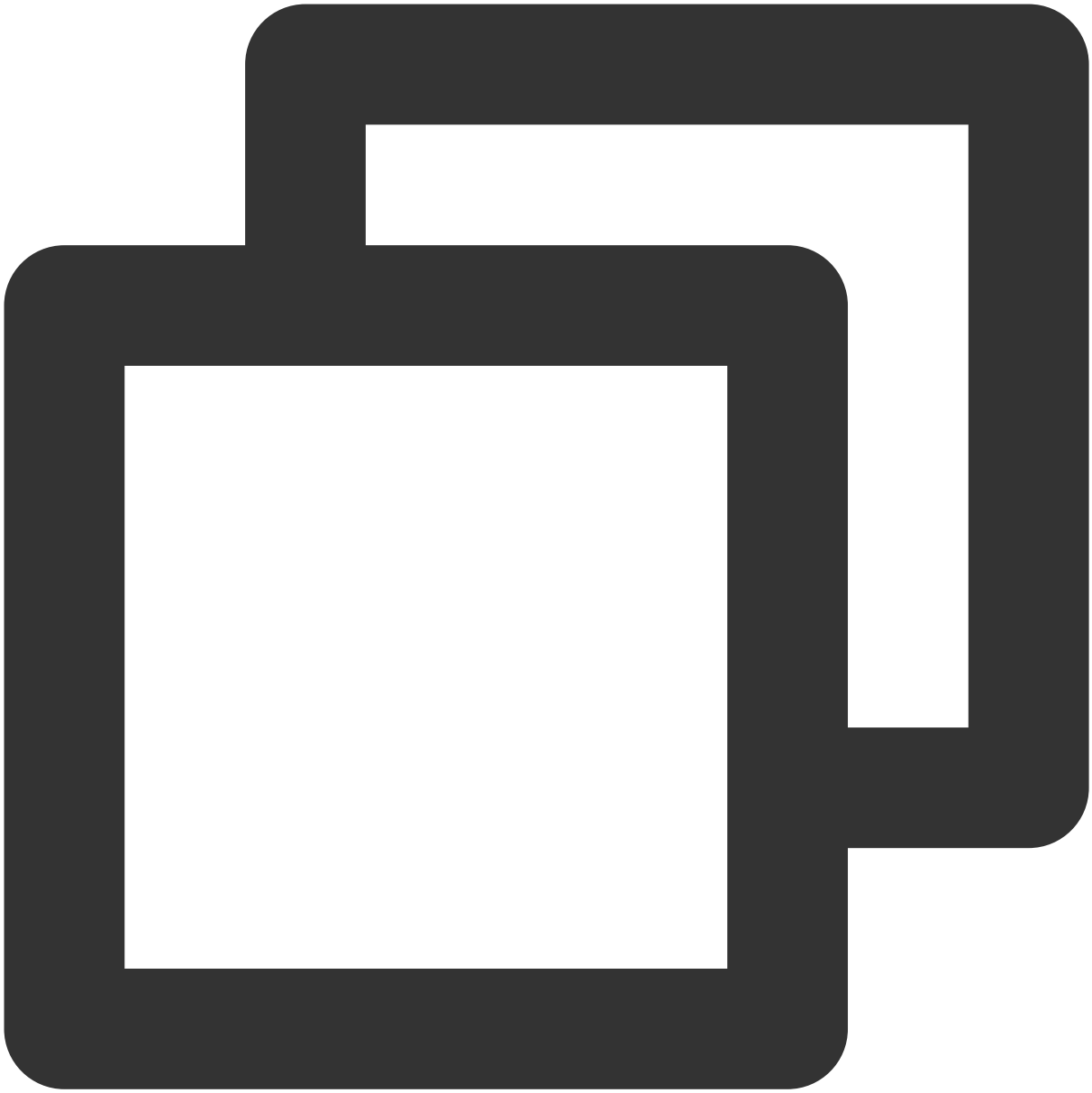
Conditional Rendering

©2013-2022 Tencent Cloud. All rights reserved.



```
<!--wxml-->
<view wx:if="{{view == 'WEBVIEW'}}"> WEBVIEW </view>
<view wx:elif="{{view == 'APP'}}"> APP </view>
<view wx:else="{{view == 'MINA'}}"> MINA </view>
```

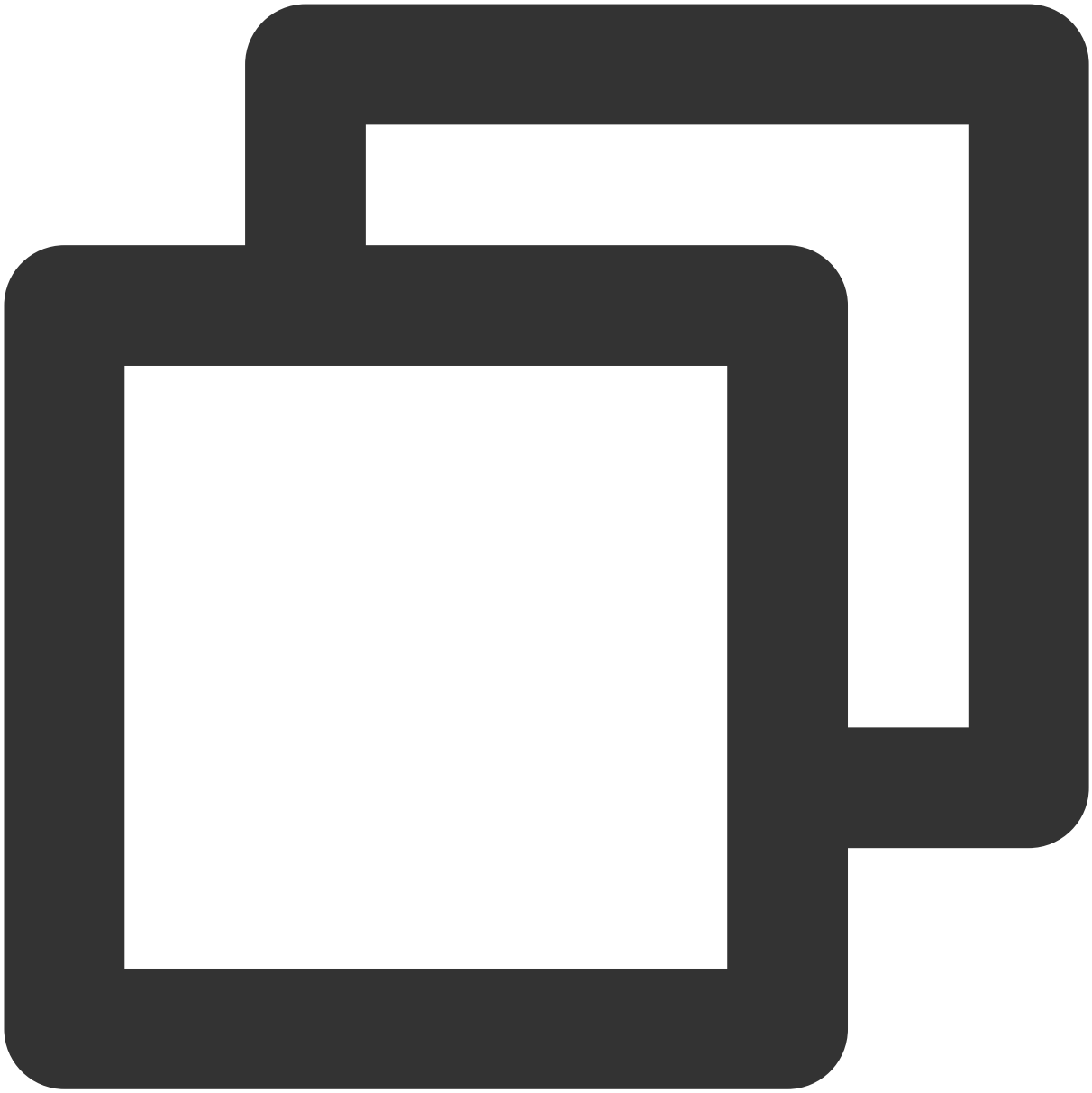
©2013-2022 Tencent Cloud. All rights reserved.



```
// page.js
Page({
  data: {
    view: 'MINA'
  }
})
```

Template

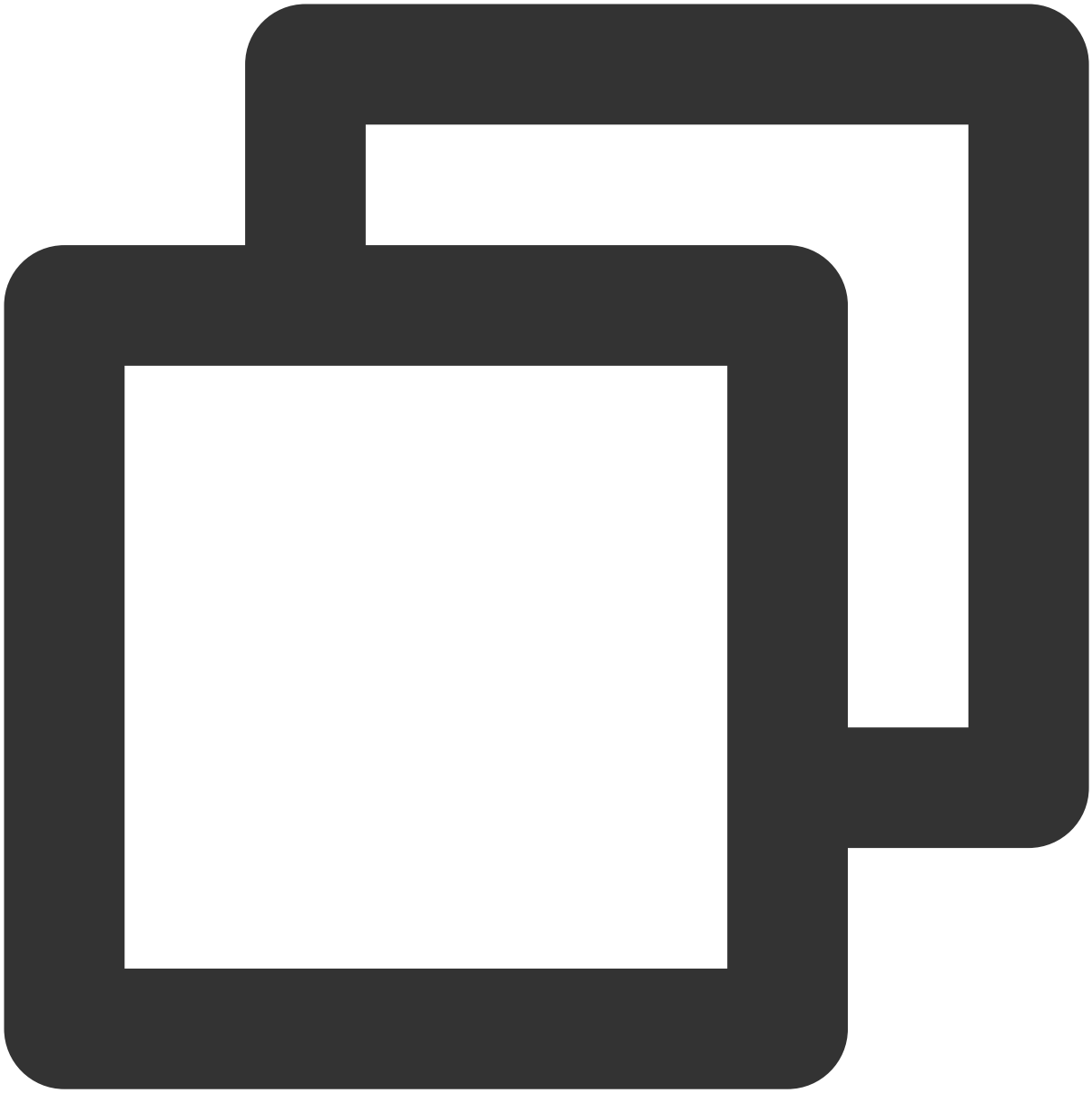
©2013-2022 Tencent Cloud. All rights reserved.



```
<!--wxml-->
<template name="staffName">
  <view>
    FirstName: {{firstName}}, LastName: {{lastName}}
  </view>
</template>

<template is="staffName" data="{{...staffA}}"></template>
<template is="staffName" data="{{...staffB}}"></template>
<template is="staffName" data="{{...staffC}}"></template>
```

©2013-2022 Tencent Cloud. All rights reserved.



```
// page.js
Page({
  data: {
    staffA: {firstName: 'Hulk', lastName: 'Hu'},
    staffB: {firstName: 'Shang', lastName: 'You'},
    staffC: {firstName: 'Gideon', lastName: 'Lin'}
  }
})
```

©2013-2022 Tencent Cloud. All rights reserved.

©2013-2022 Tencent Cloud. All rights reserved.

Data Binding

Last updated：2024-07-12 17:06:10

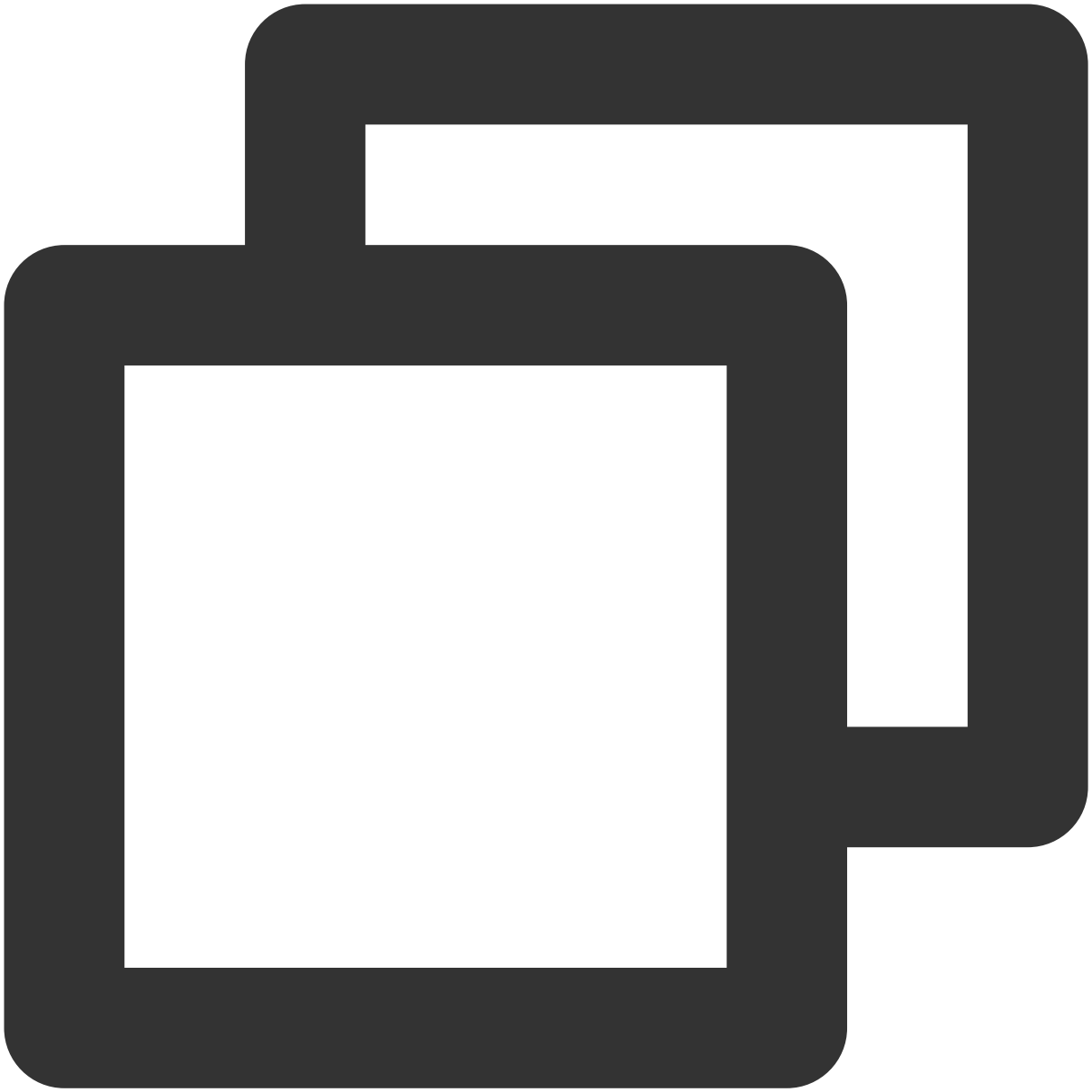
The dynamic data in WXML comes from the data of the corresponding Page.

Simple Binding

Data binding uses the Mustache syntax (double curly brackets) to wrap the variables. It can be used for the following:

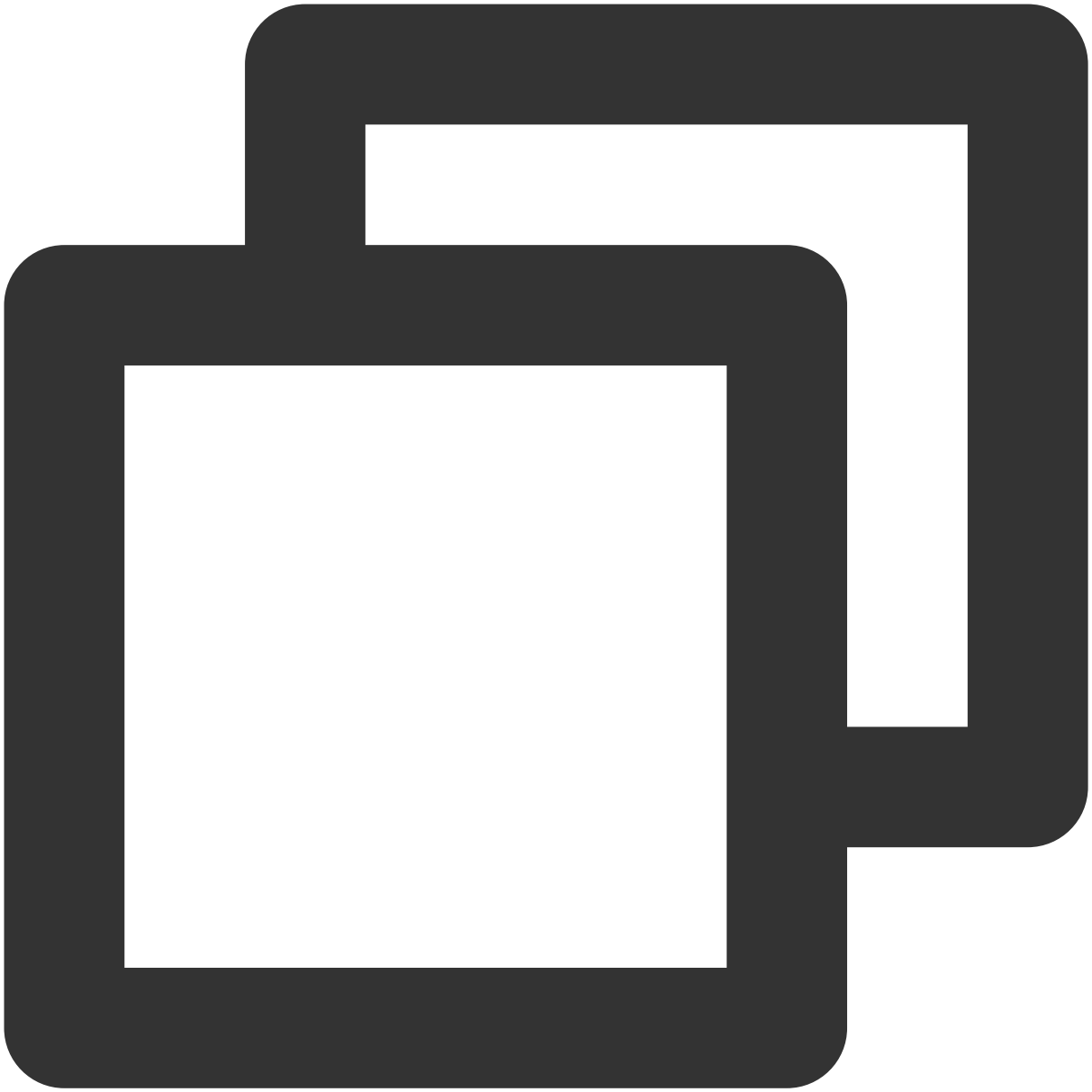
Content

©2013-2022 Tencent Cloud. All rights reserved.



```
<view> {{ message }} </view>
```

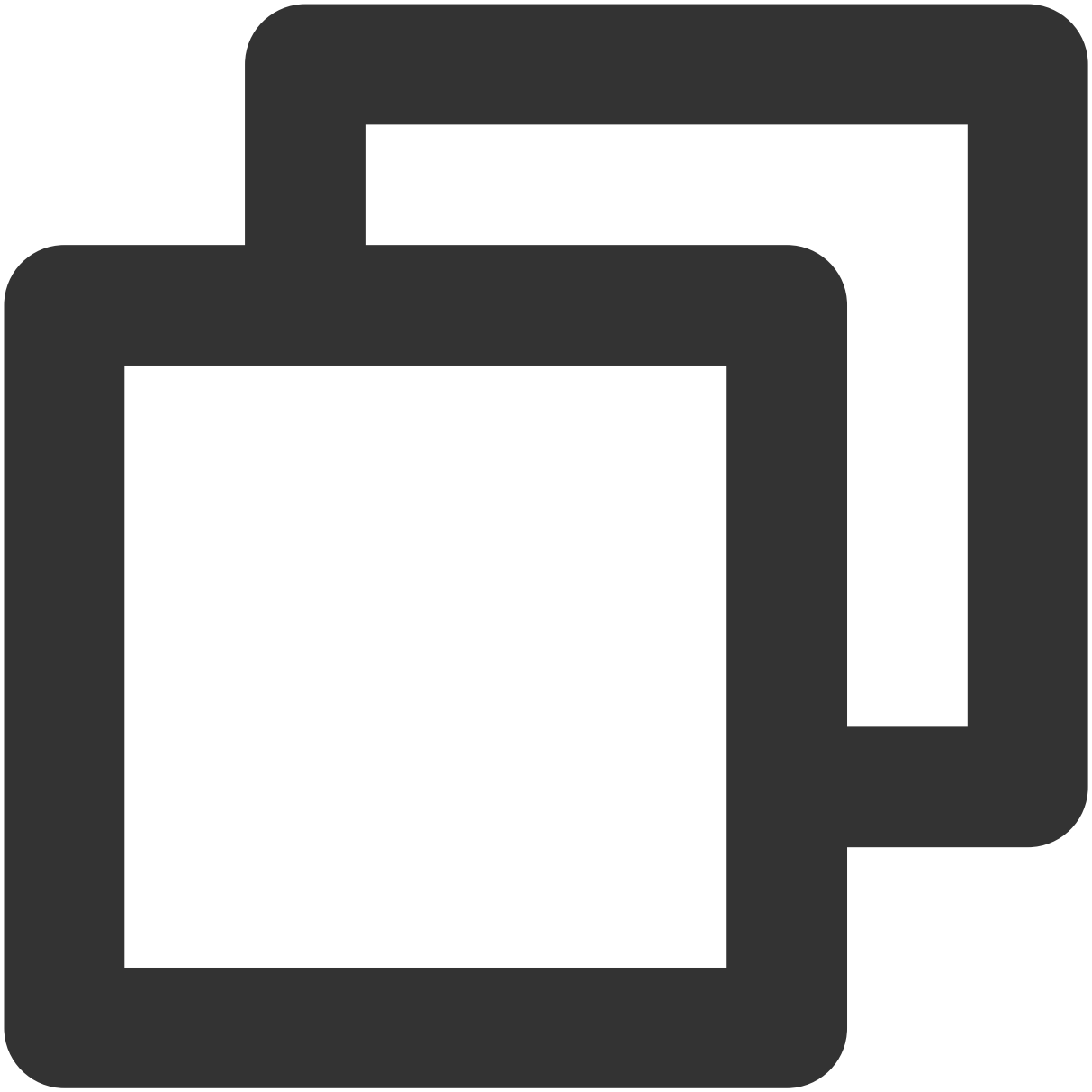
©2013-2022 Tencent Cloud. All rights reserved.



```
Page({
  data: {
    message: 'Hello MINA!'
  }
})
```

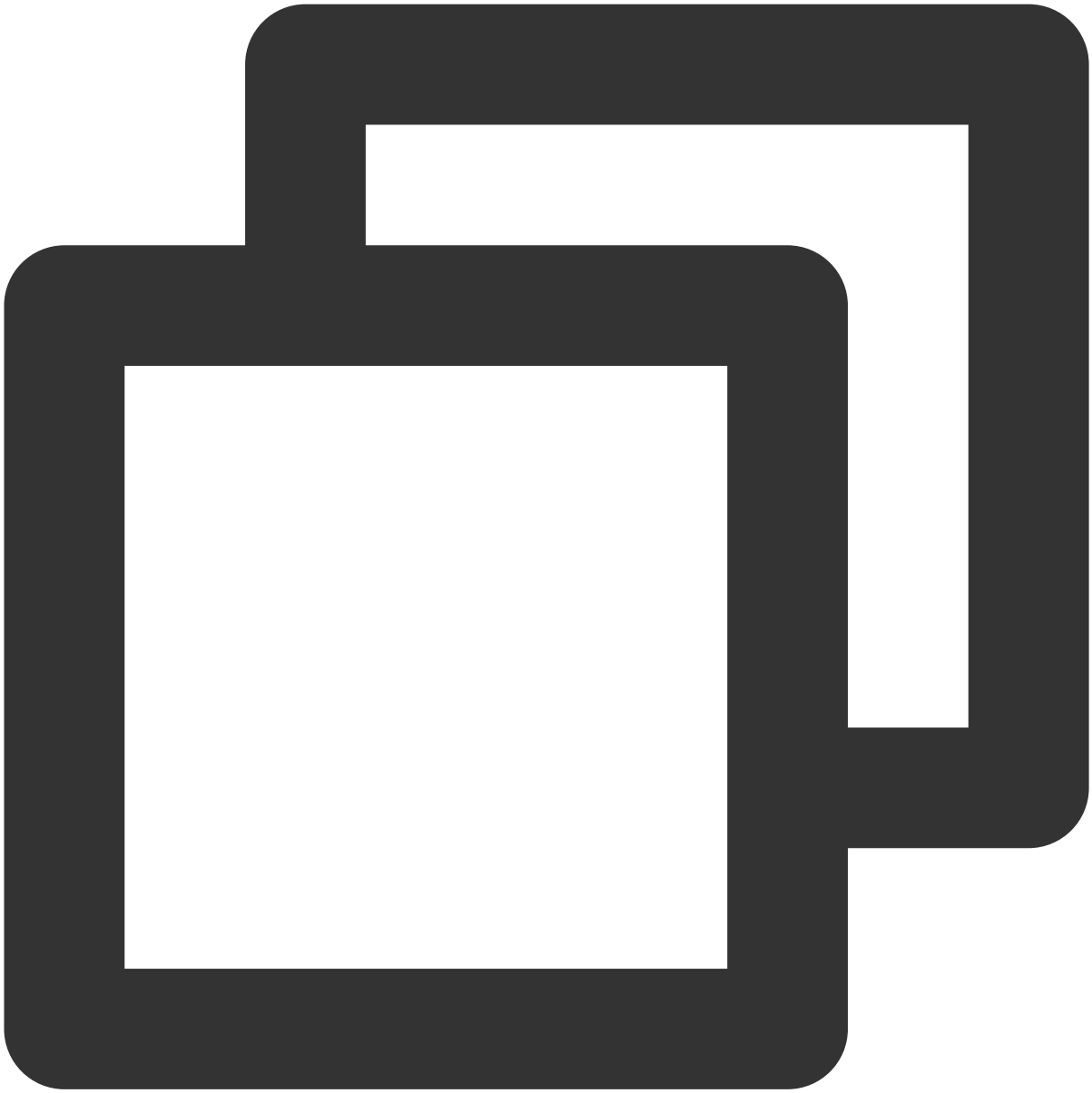
Component Properties (must be enclosed in double quotes)

©2013-2022 Tencent Cloud. All rights reserved.



```
<view id="item-{{id}}"> </view>
```

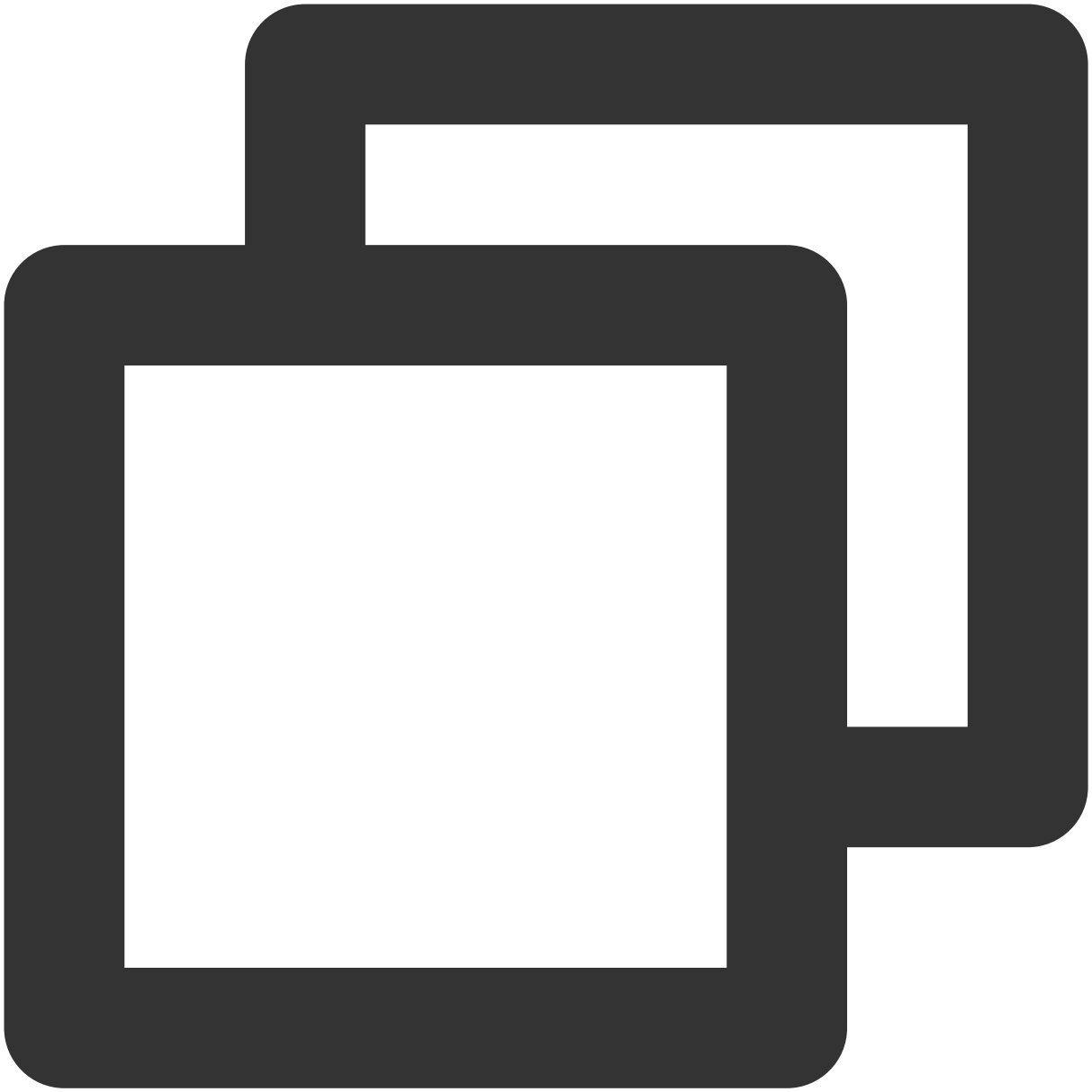
©2013-2022 Tencent Cloud. All rights reserved.



```
Page({
  data: {
    id: 0
  }
})
```

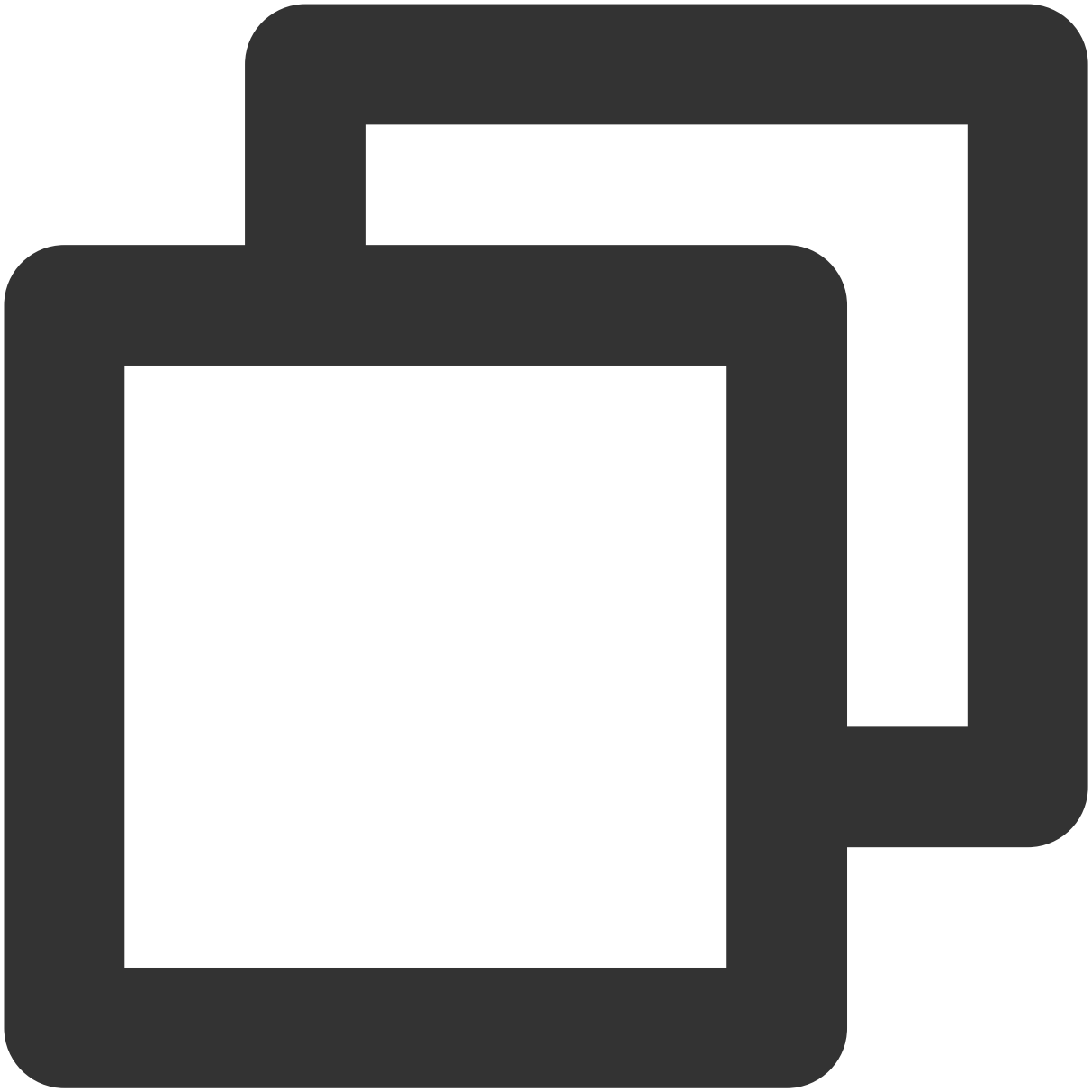
Control Properties (must be enclosed in double quotes)

©2013-2022 Tencent Cloud. All rights reserved.



```
<view wx:if="{{condition}}"> </view>
```

©2013-2022 Tencent Cloud. All rights reserved.



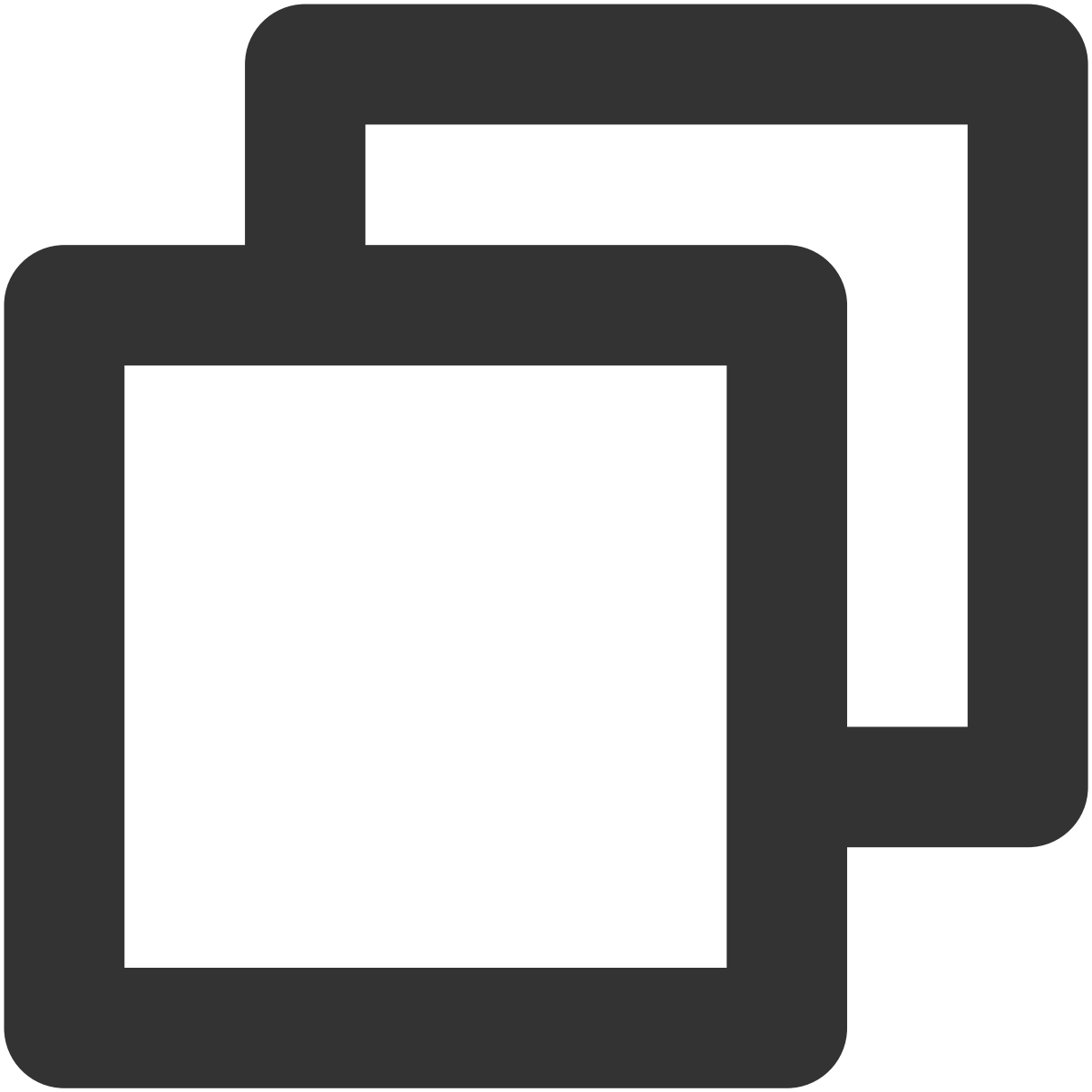
```
Page({
  data: {
    condition: true
  }
})
```

Keywords (must be enclosed in double quotes)

True: Boolean-type true.

false: Boolean-type false.

©2013-2022 Tencent Cloud. All rights reserved.



```
<checkbox checked="{{false}}"> </checkbox>
```

Note：

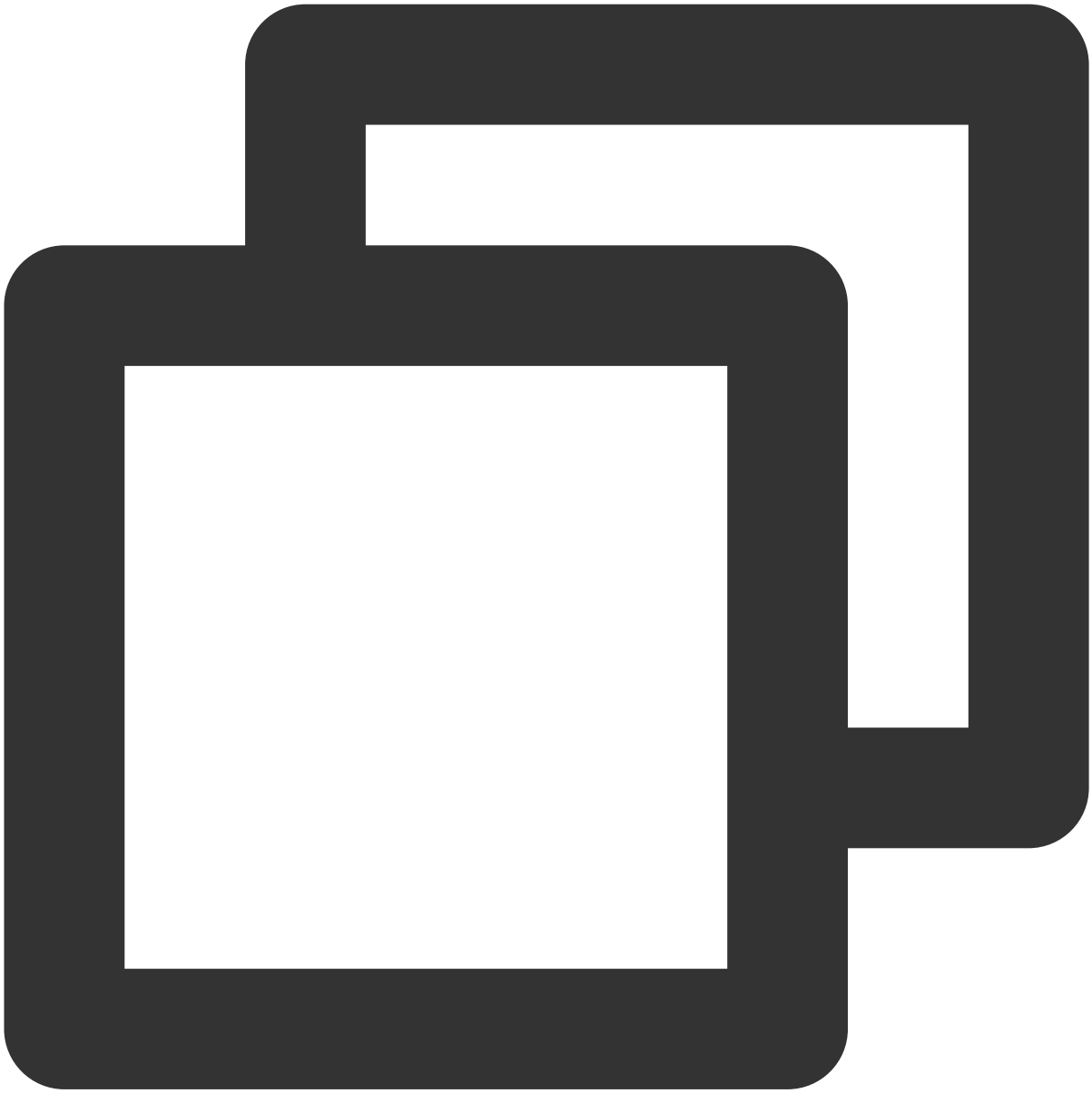
Do not directly write **checked="false"**. Its computing result is a string which represents a true value when converted to a Boolean value.

Operations

©2013-2022 Tencent Cloud. All rights reserved.

You can implement simple operations in {{}}. This syntax supports the following methods:

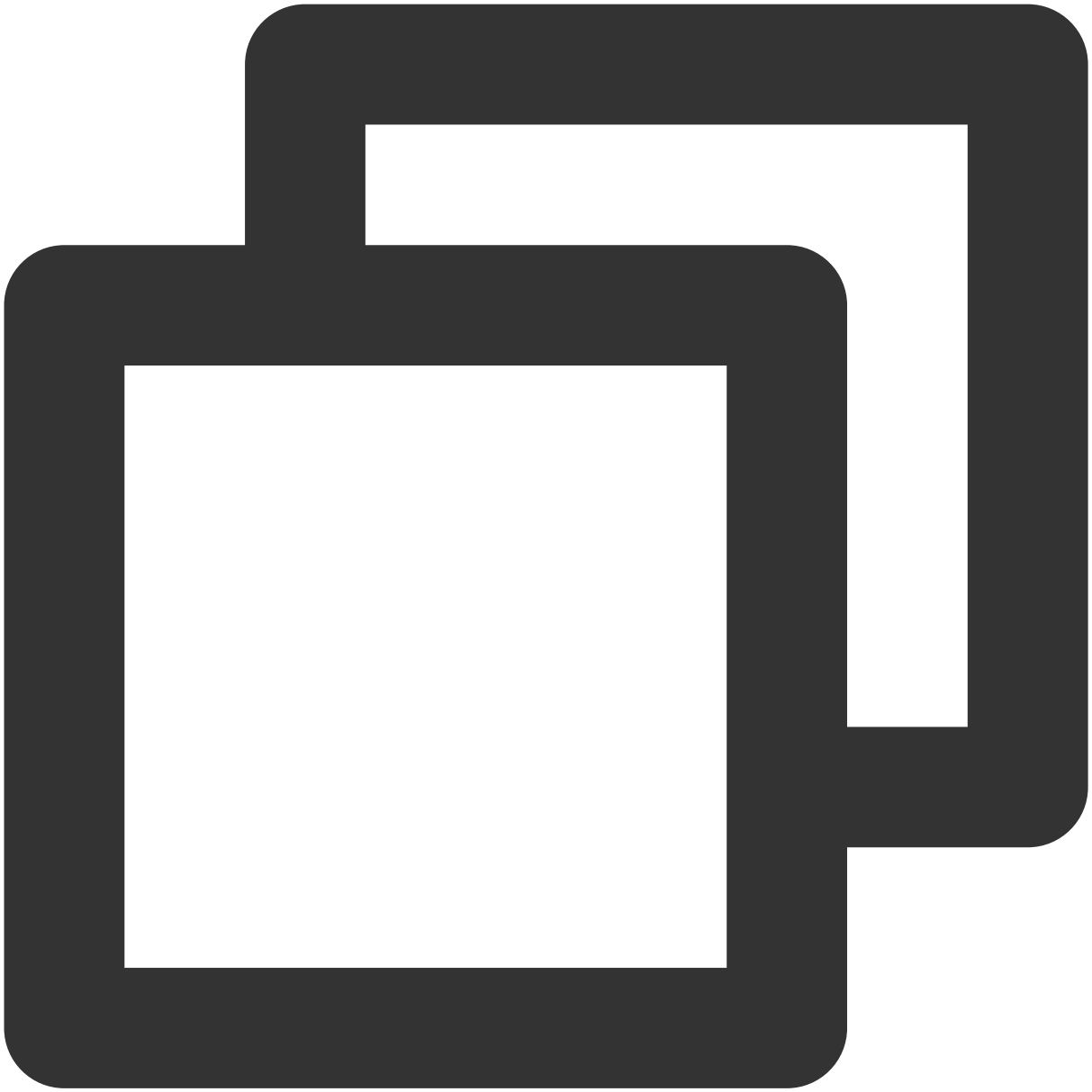
Ternary Operations



```
<view hidden="{{flag ? true : false}}"> Hidden </view>
```

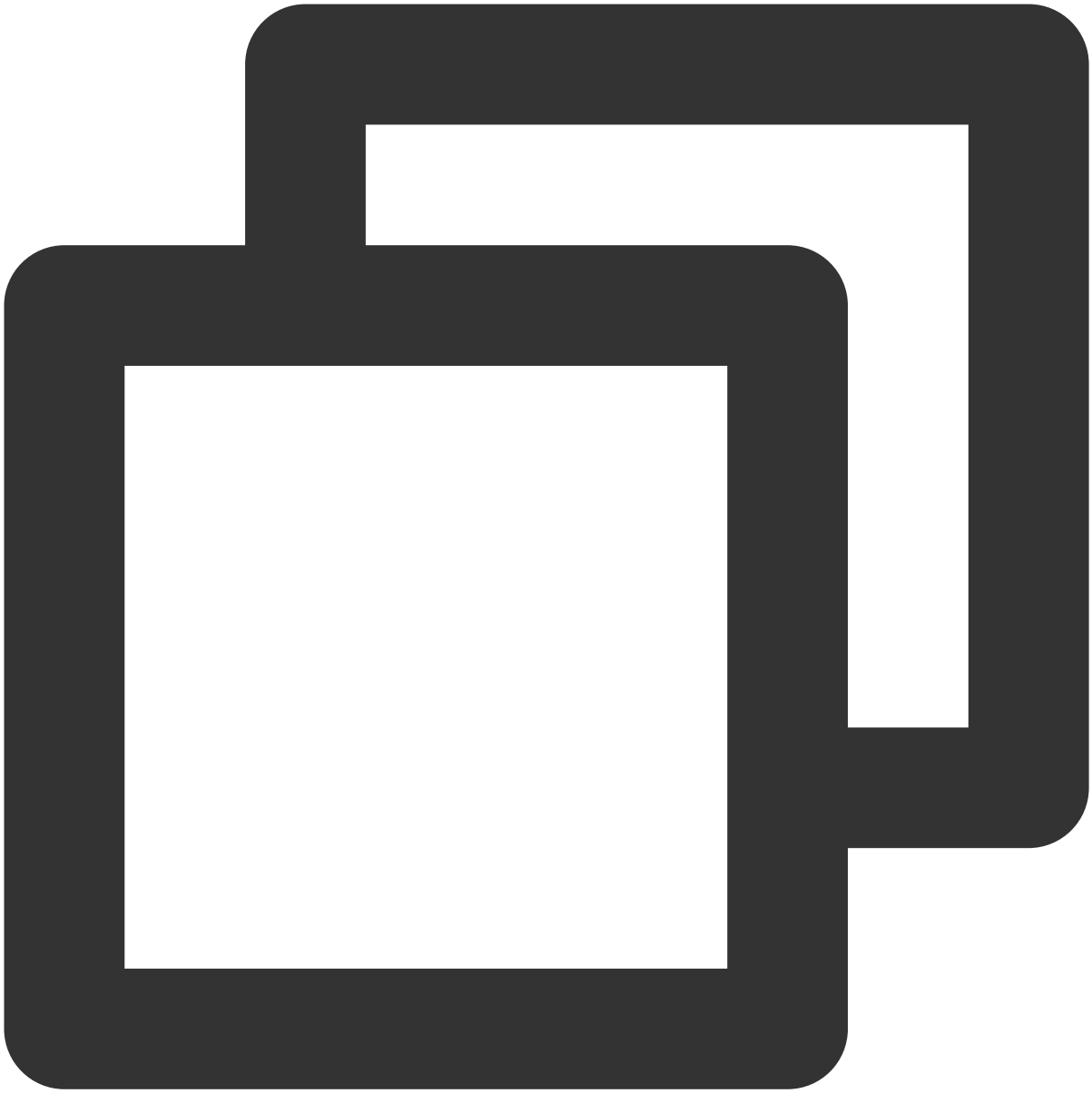
Arithmetic Operations

©2013-2022 Tencent Cloud. All rights reserved.



```
<view> {{a + b}} + {{c}} + d </view>
```

©2013-2022 Tencent Cloud. All rights reserved.

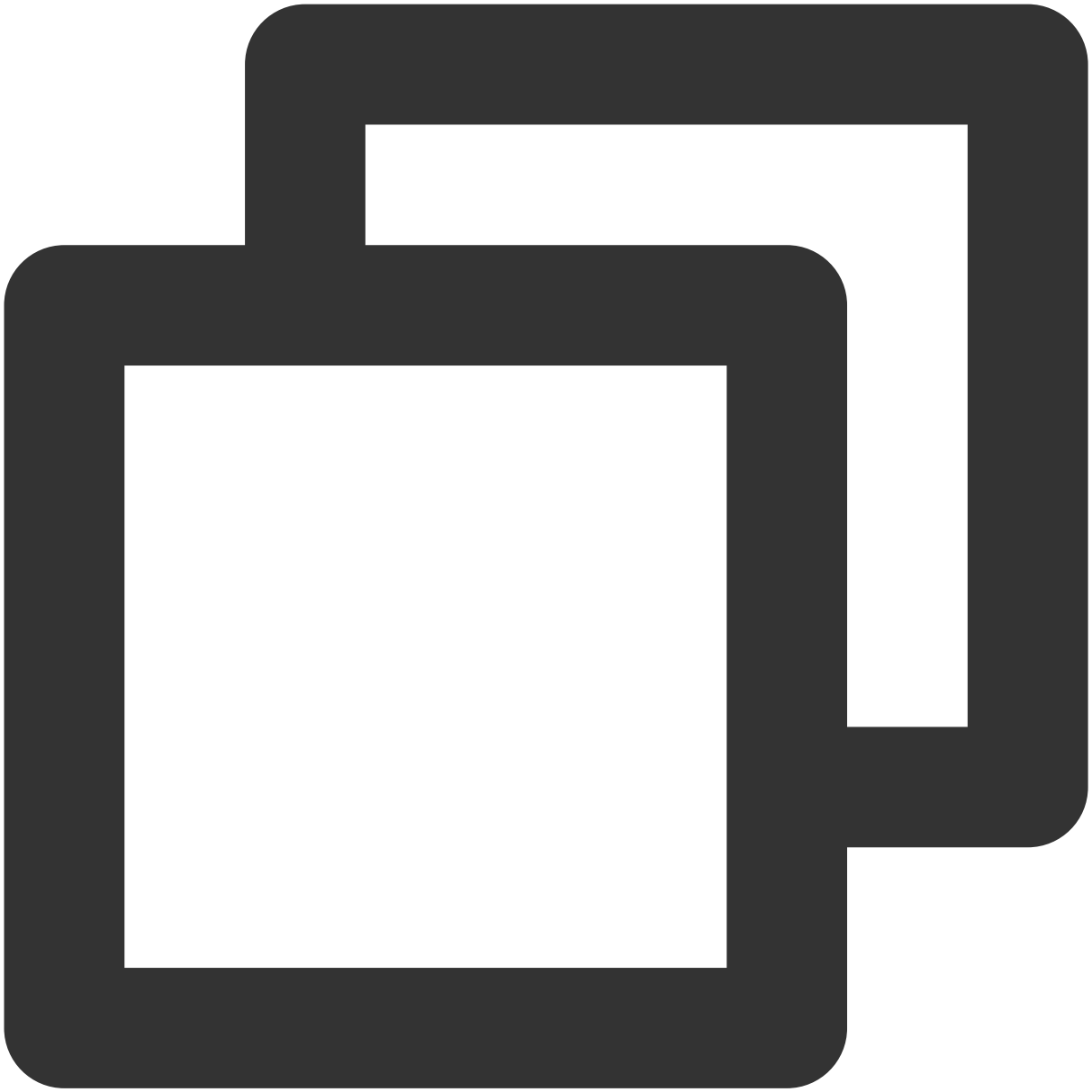


```
Page({
  data: {
    a: 1,
    b: 2,
    c: 3
  }
})
```

The content in view is 3 + 3 + d.

Logical Judgments

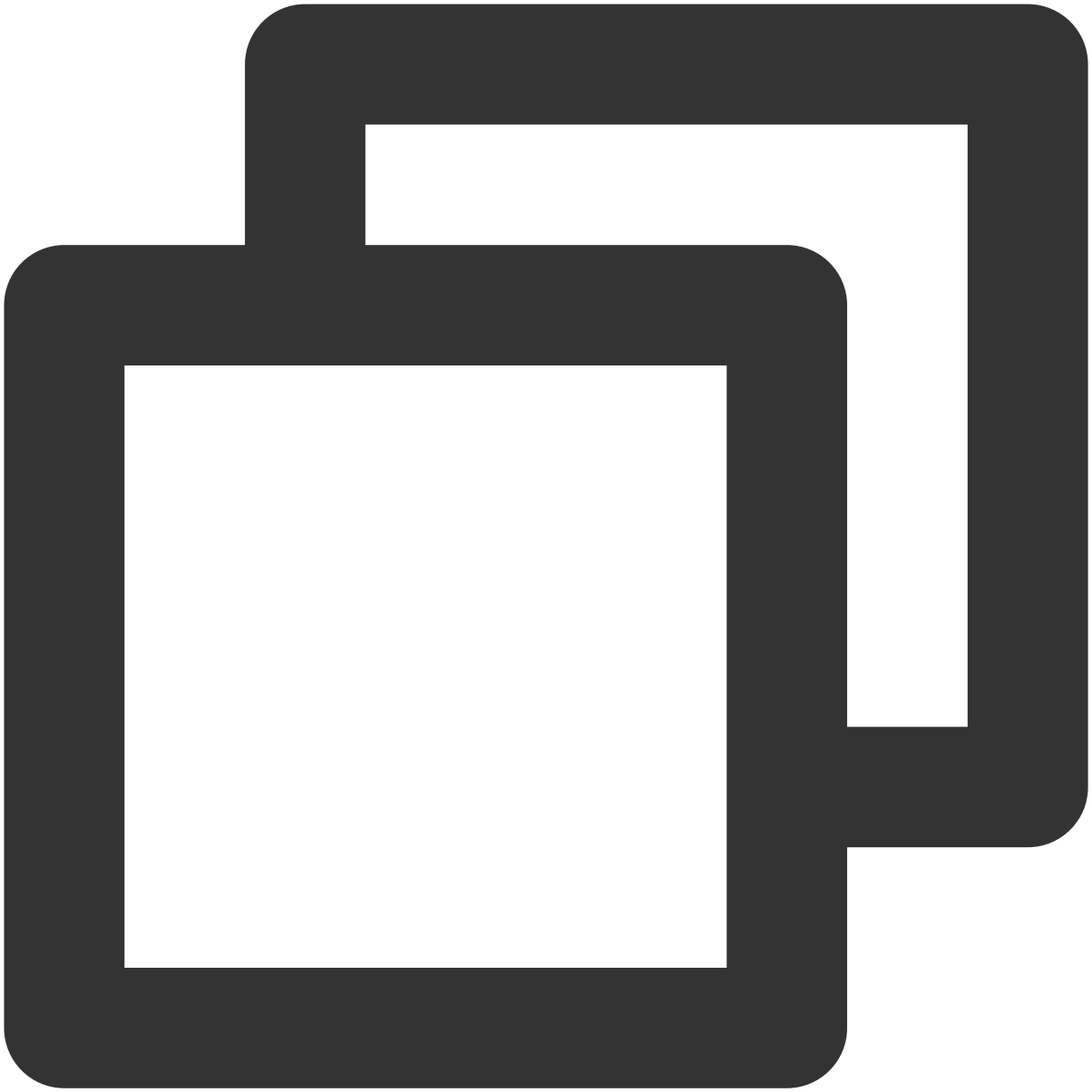
©2013-2022 Tencent Cloud. All rights reserved.



```
<view wx:if="{{length > 5}}"> </view>
```

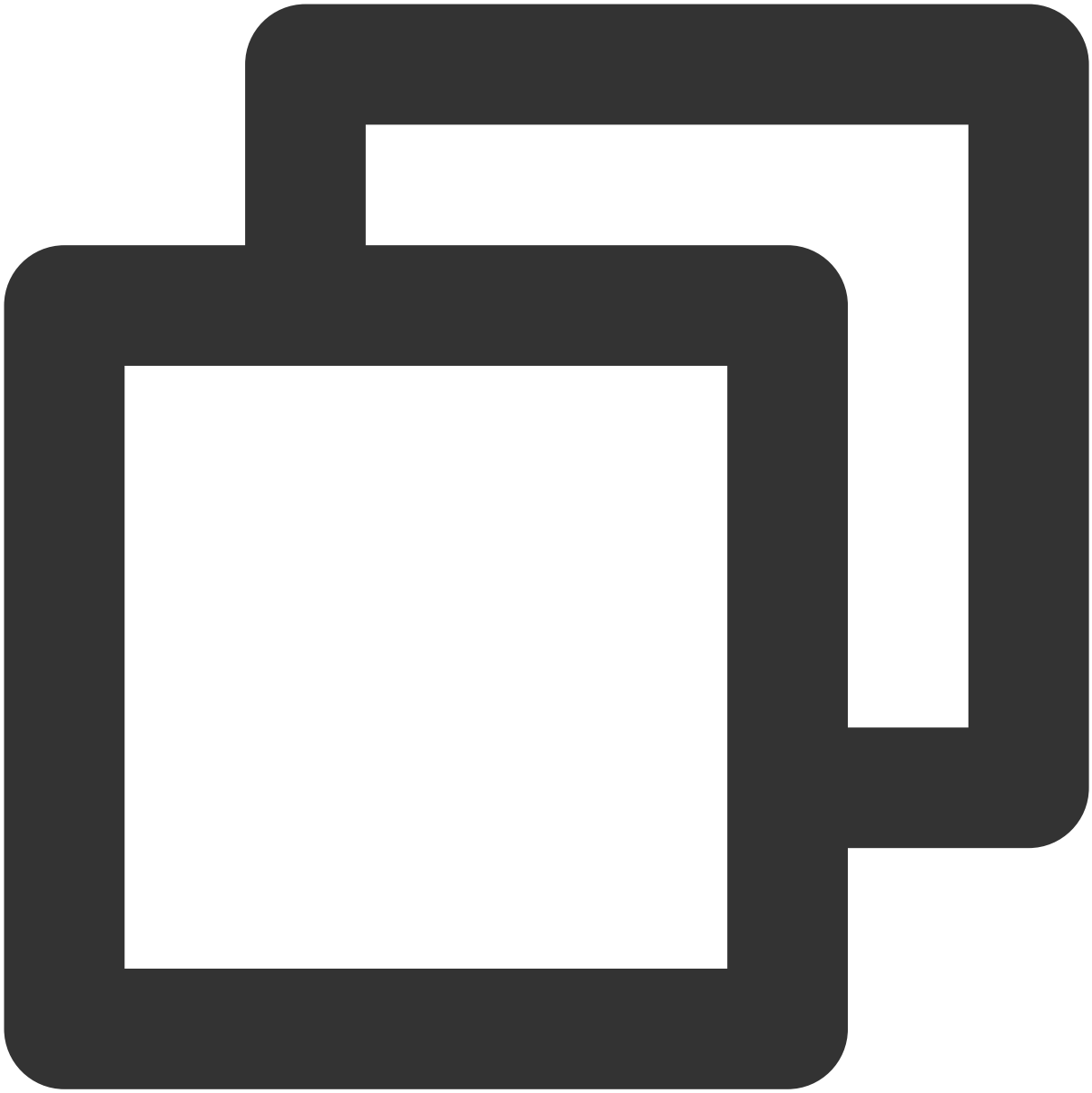
String Operations

©2013-2022 Tencent Cloud. All rights reserved.



```
<view>{{"hello" + name}}</view>
```

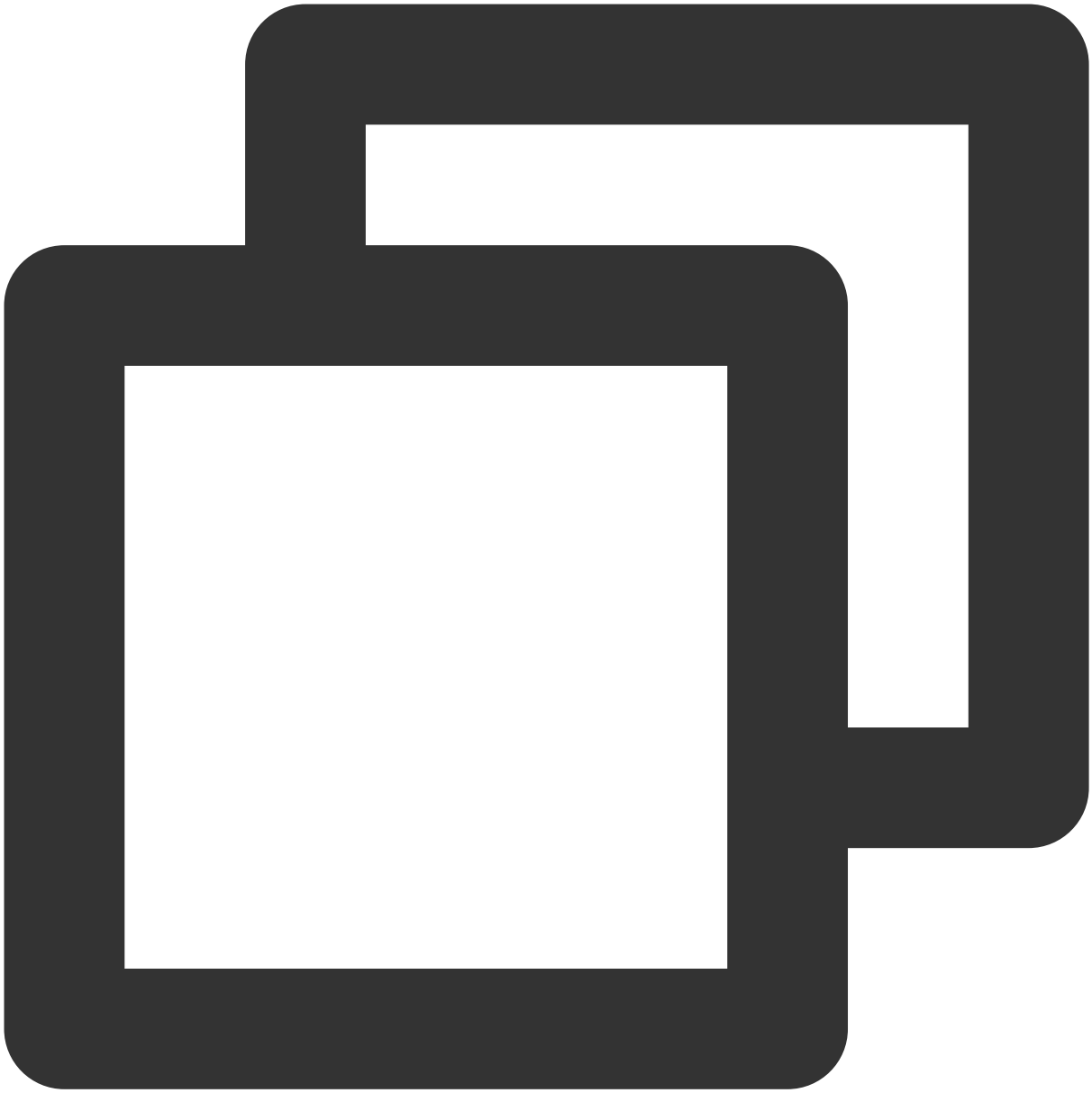
©2013-2022 Tencent Cloud. All rights reserved.



```
Page({
  data:{
    name: 'MINA'
  }
})
```

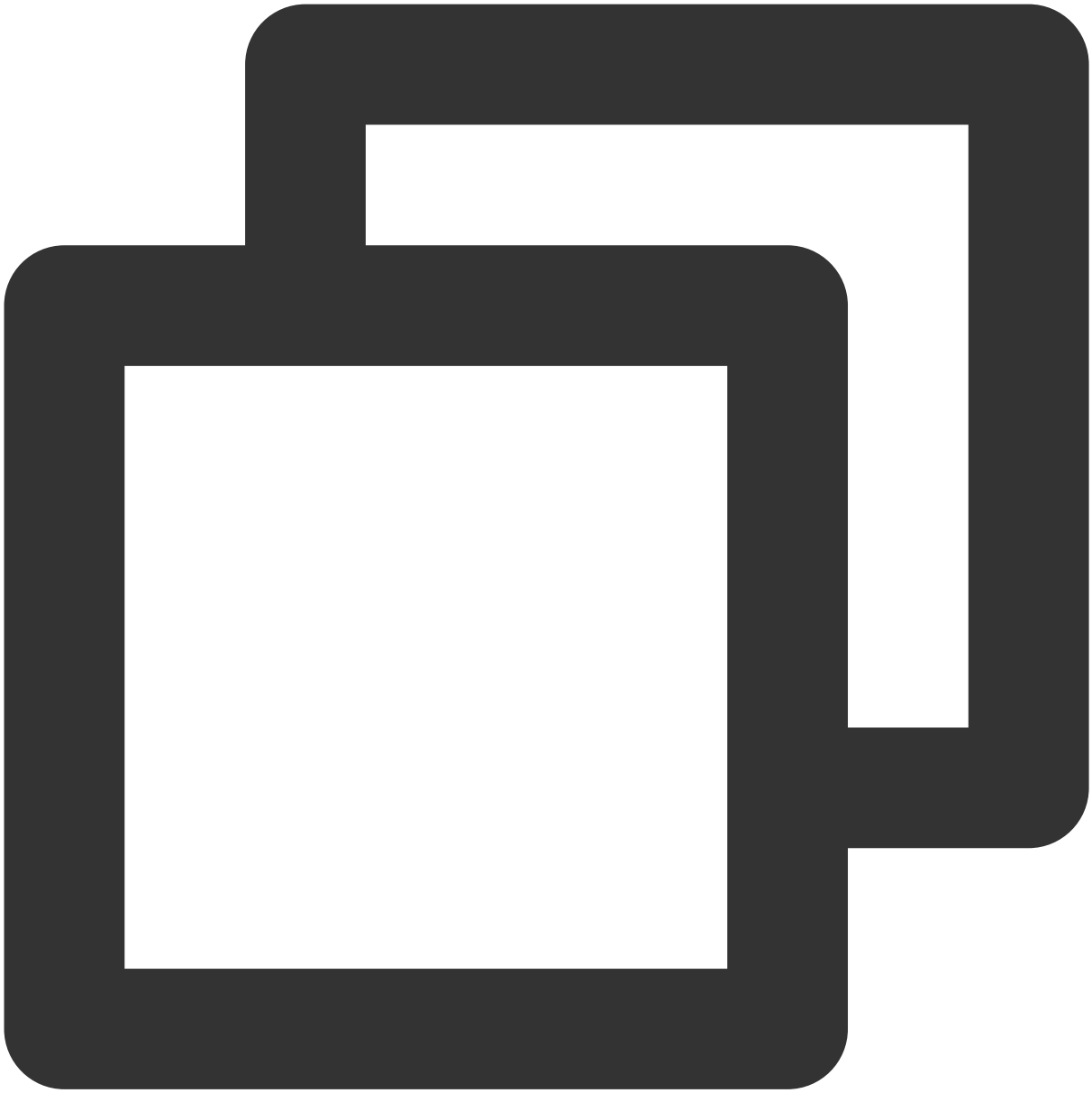
Data path Operations

©2013-2022 Tencent Cloud. All rights reserved.



```
<view>{{object.key}} {{array[0]}}</view>
```

©2013-2022 Tencent Cloud. All rights reserved.



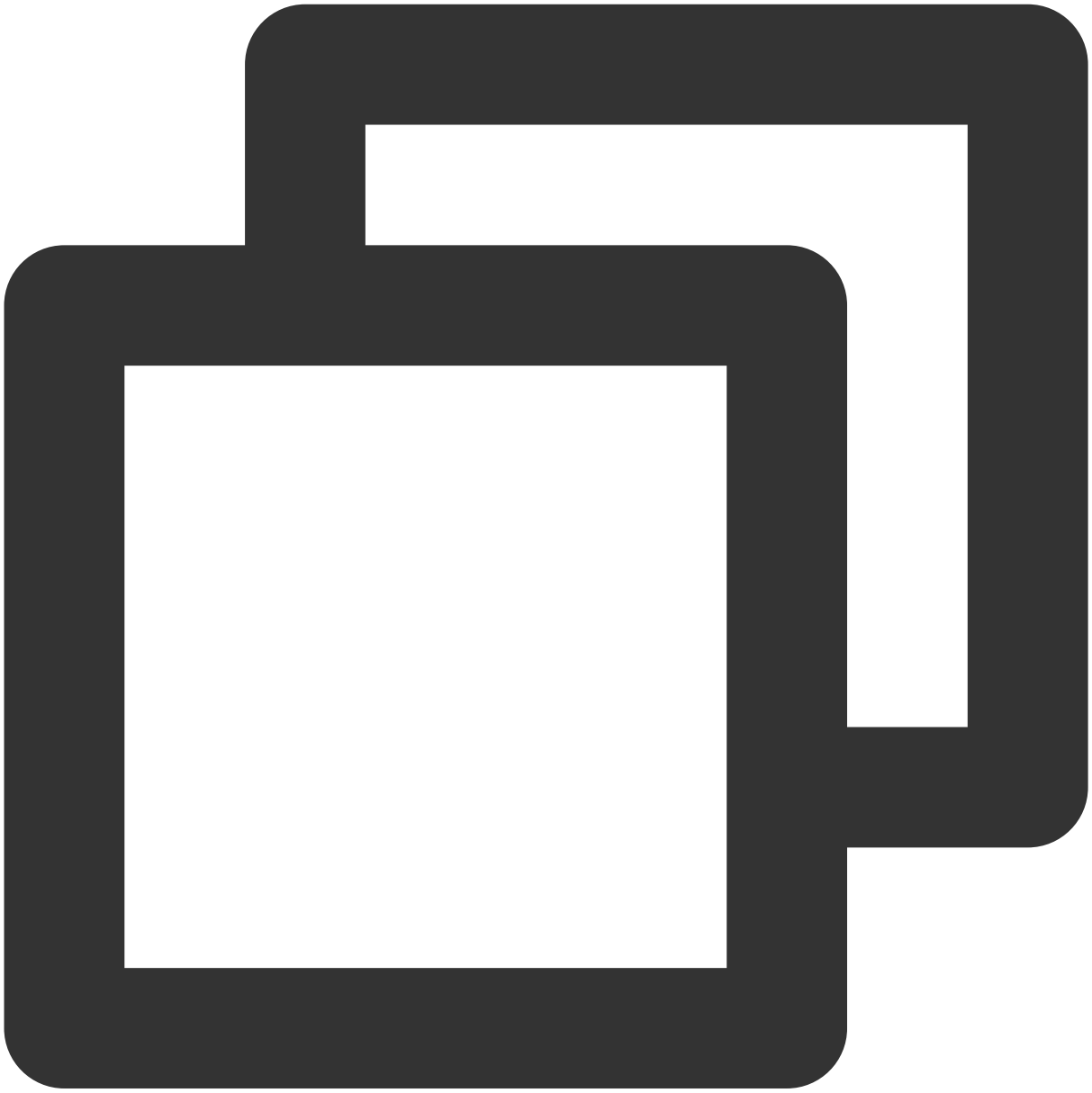
```
Page({
  data: {
    object: {
      key: 'Hello '
    },
    array: ['MINA']
  }
})
```

©2013-2022 Tencent Cloud. All rights reserved.

Combinations

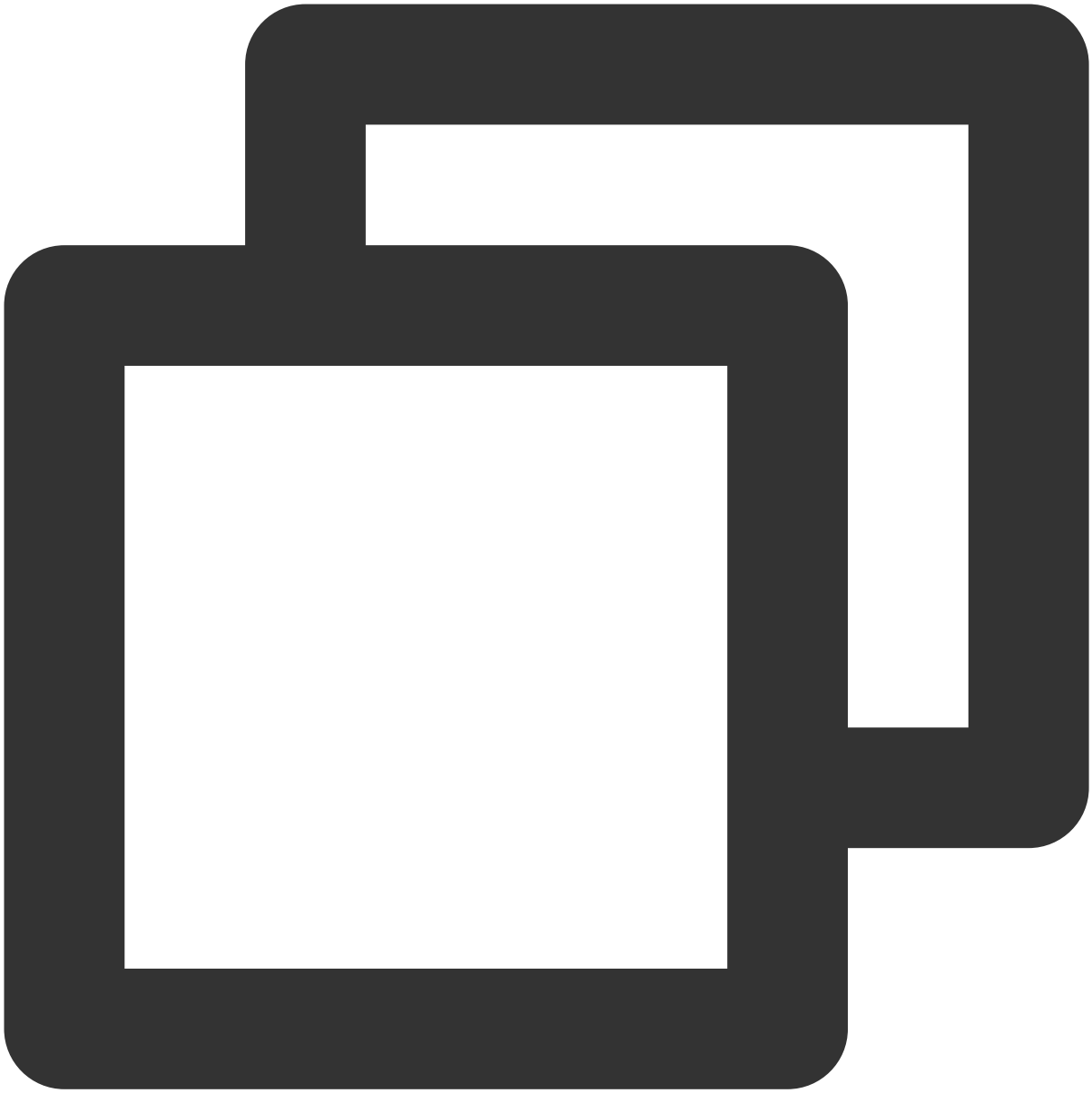
You can also directly implement combinations in Mustache syntax to build new objects and arrays.

Arrays



```
<view wx:for="{{[zero, 1, 2, 3, 4]}}"> {{item}} </view>
```

©2013-2022 Tencent Cloud. All rights reserved.

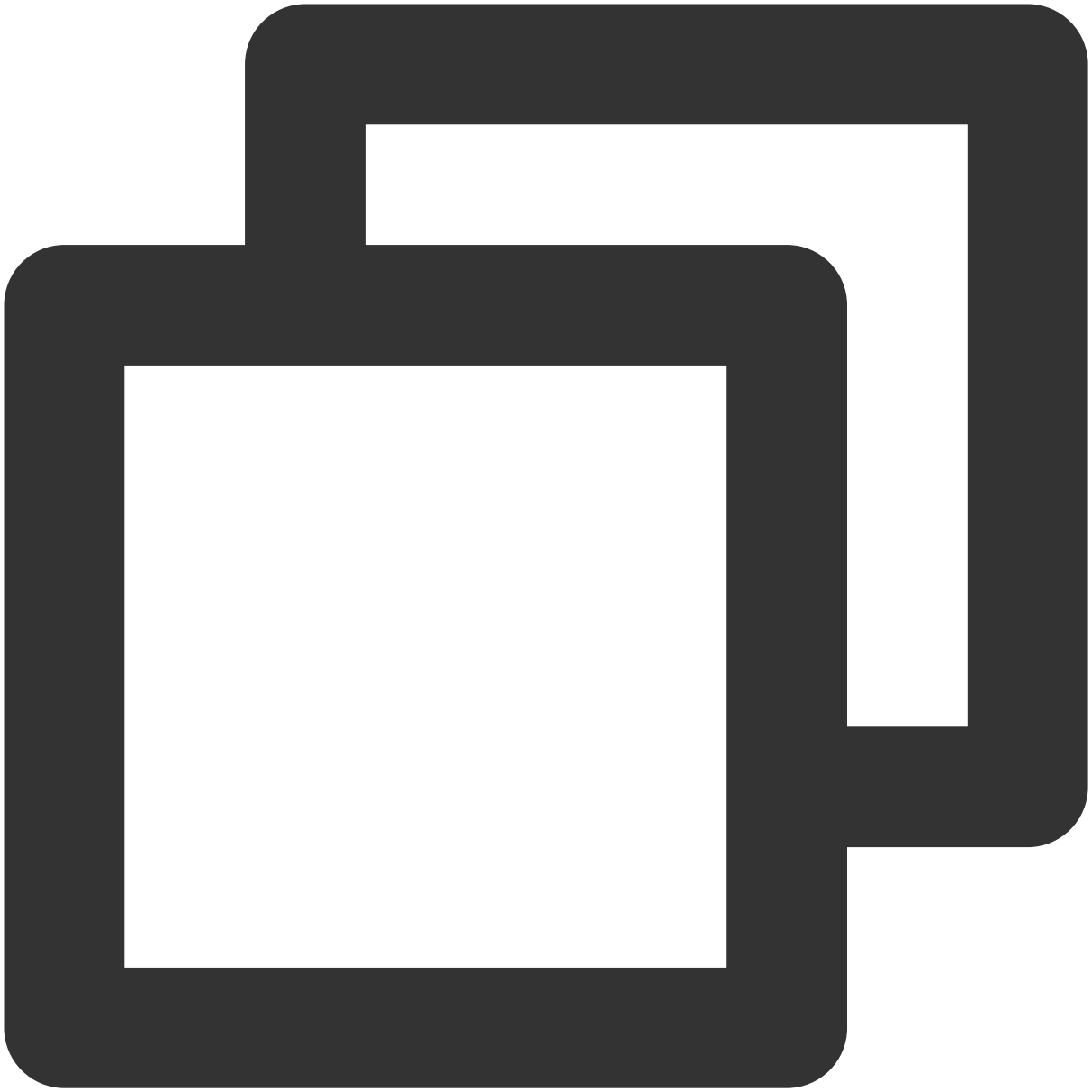


```
Page({
  data: {
    zero: 0
  }
})
```

The end result is the array [0, 1, 2, 3, 4].

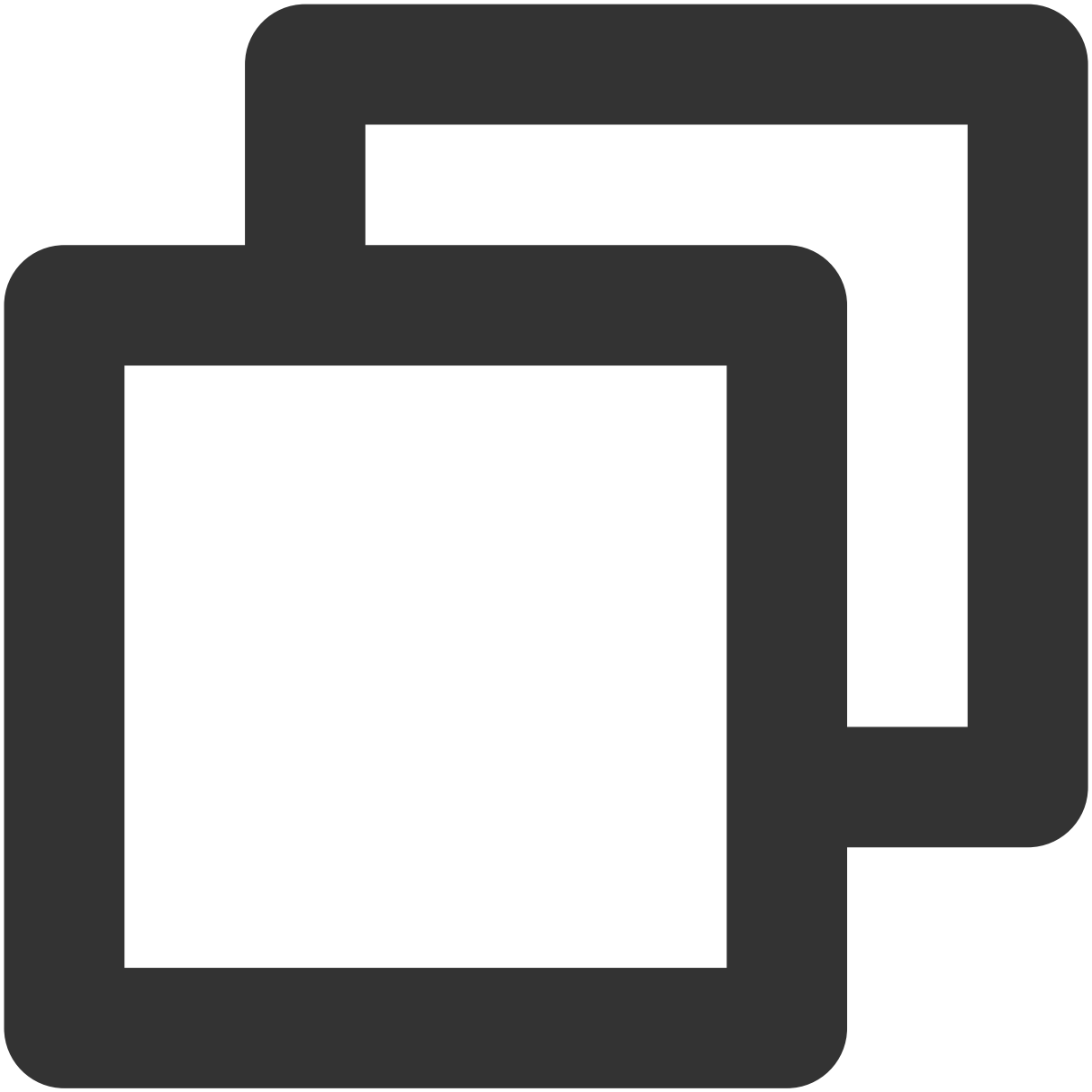
Objects

©2013-2022 Tencent Cloud. All rights reserved.



```
<template is="objectCombine" data="{{for: a, bar: b}}"></template>
```

©2013-2022 Tencent Cloud. All rights reserved.

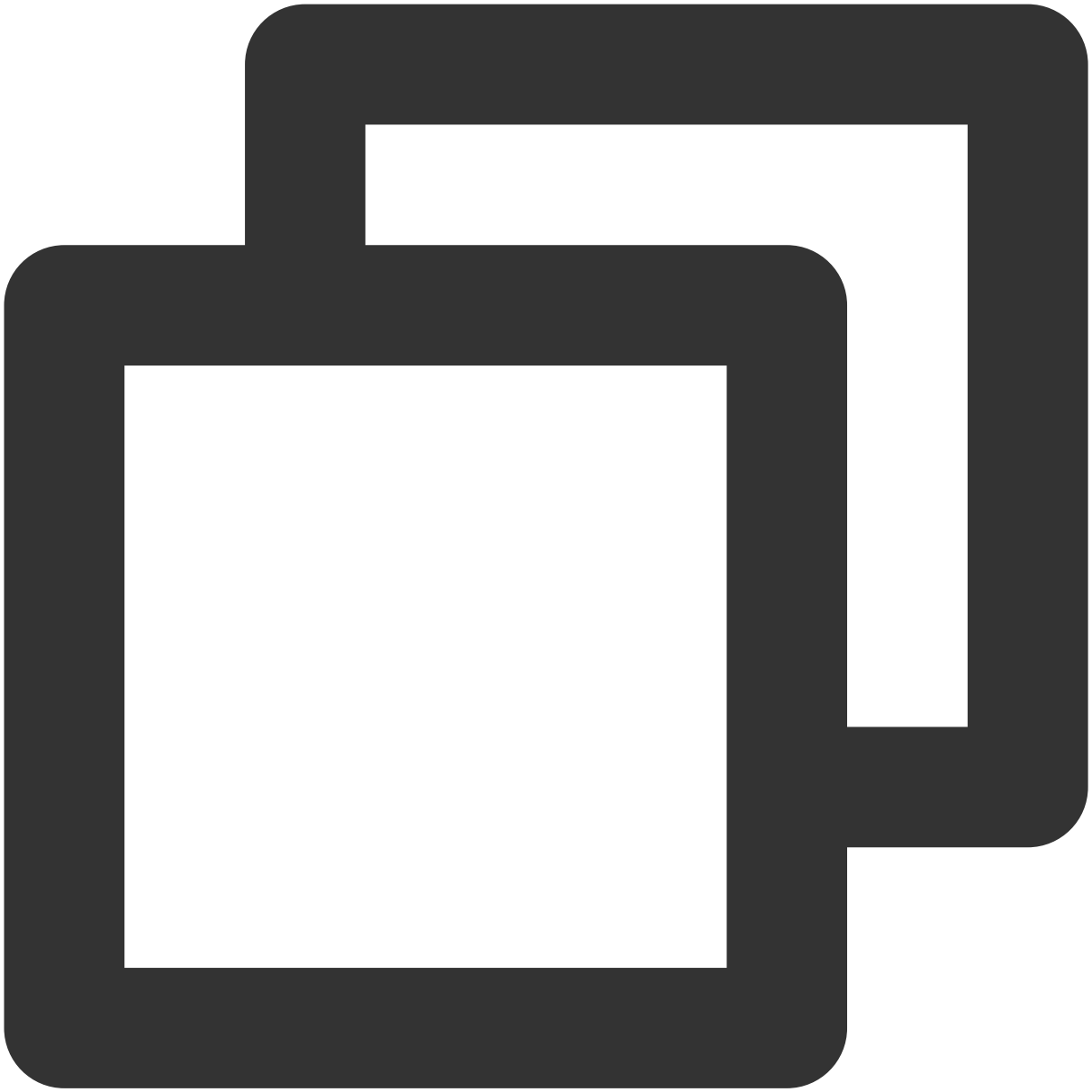


```
Page({
  data: {
    a: 1,
    b: 2
  }
})
```

The end result is the object {for: 1, bar: 2}.

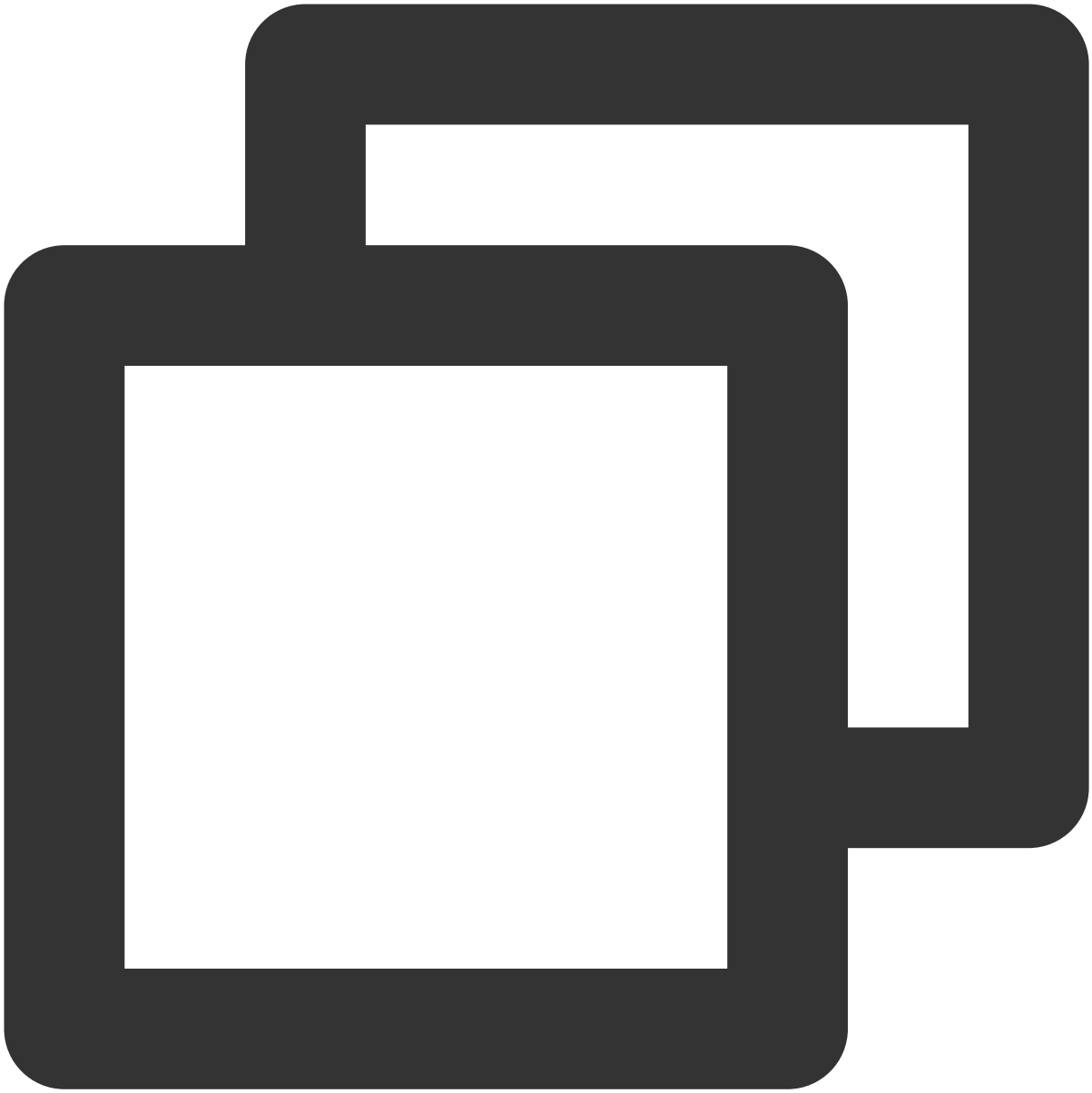
You can also use the extended operator ... to extend the object.

©2013-2022 Tencent Cloud. All rights reserved.



```
<template is="objectCombine" data="{{...obj1, ...obj2, e: 5}}"></template>
```

©2013-2022 Tencent Cloud. All rights reserved.



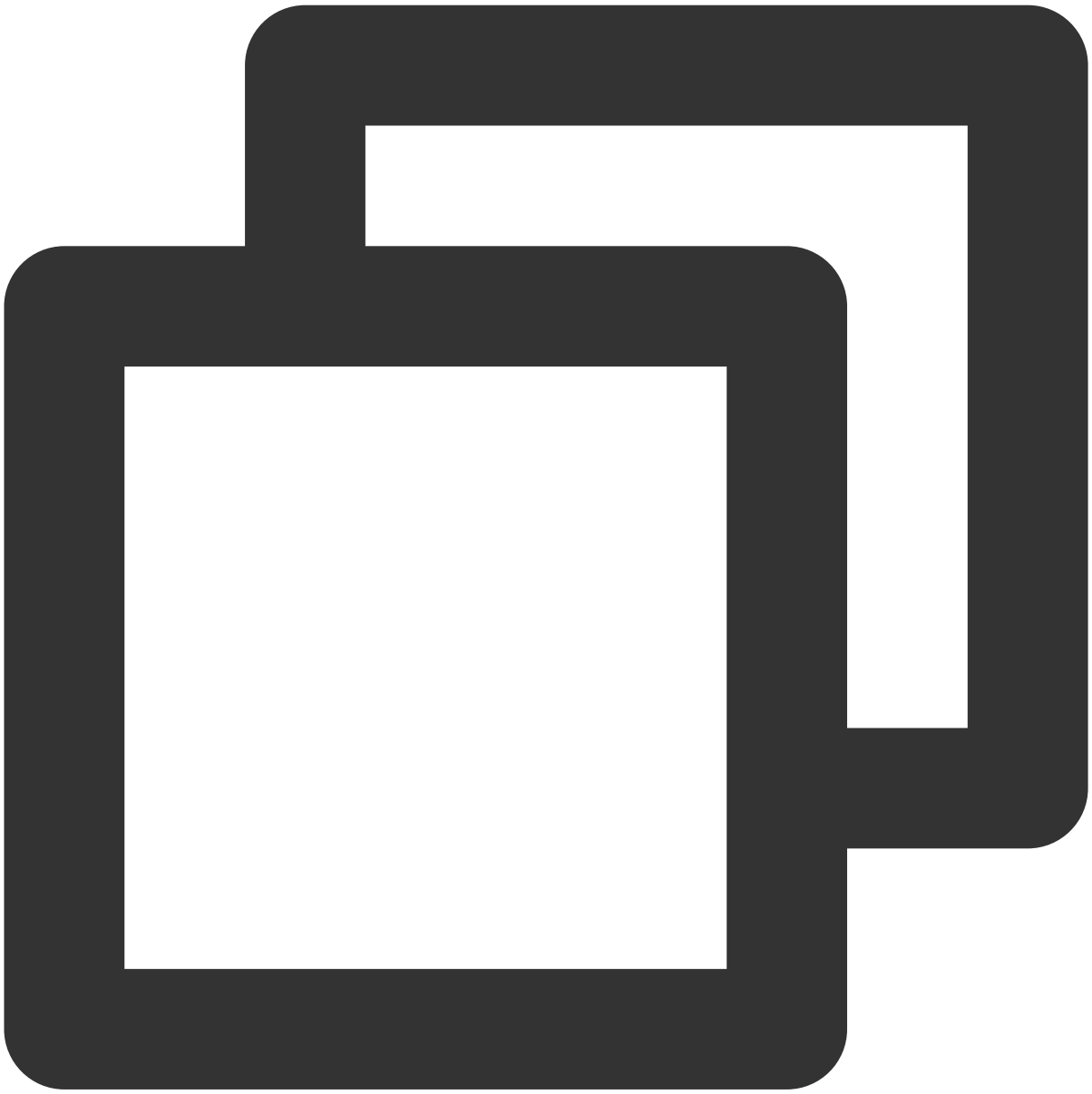
```
Page({
  data: {
    obj1: {
      a: 1,
      b: 2
    },
    obj2: {
      c: 3,
      d: 4
    }
  }
```

©2013-2022 Tencent Cloud. All rights reserved.

```
})
```

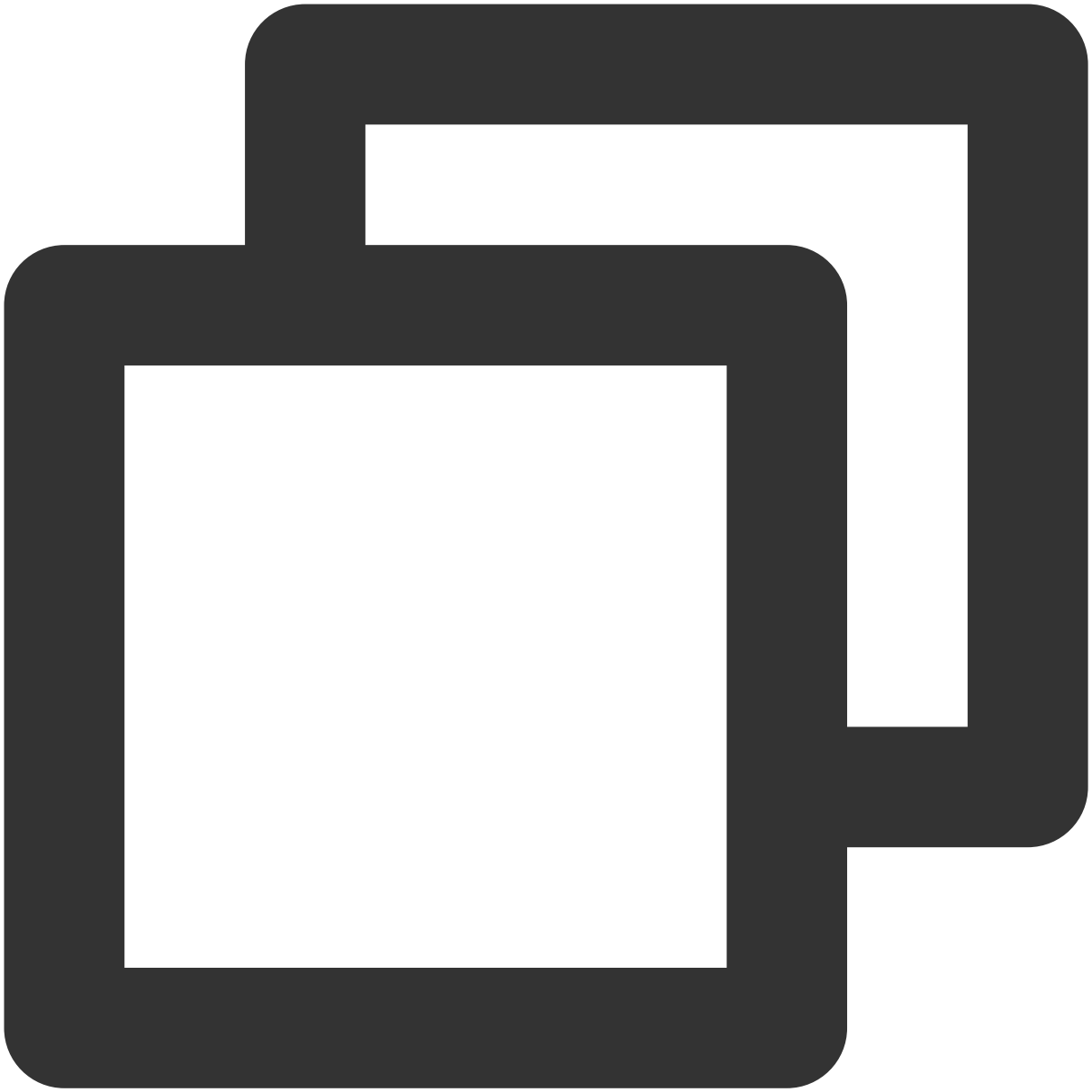
The end result is the object {a: 1, b: 2, c: 3, d: 4, e: 5}.

If the object's key and value are identical, this can be indirectly expressed.



```
<template is="objectCombine" data="{{foo, bar}}"></template>
```

©2013-2022 Tencent Cloud. All rights reserved.

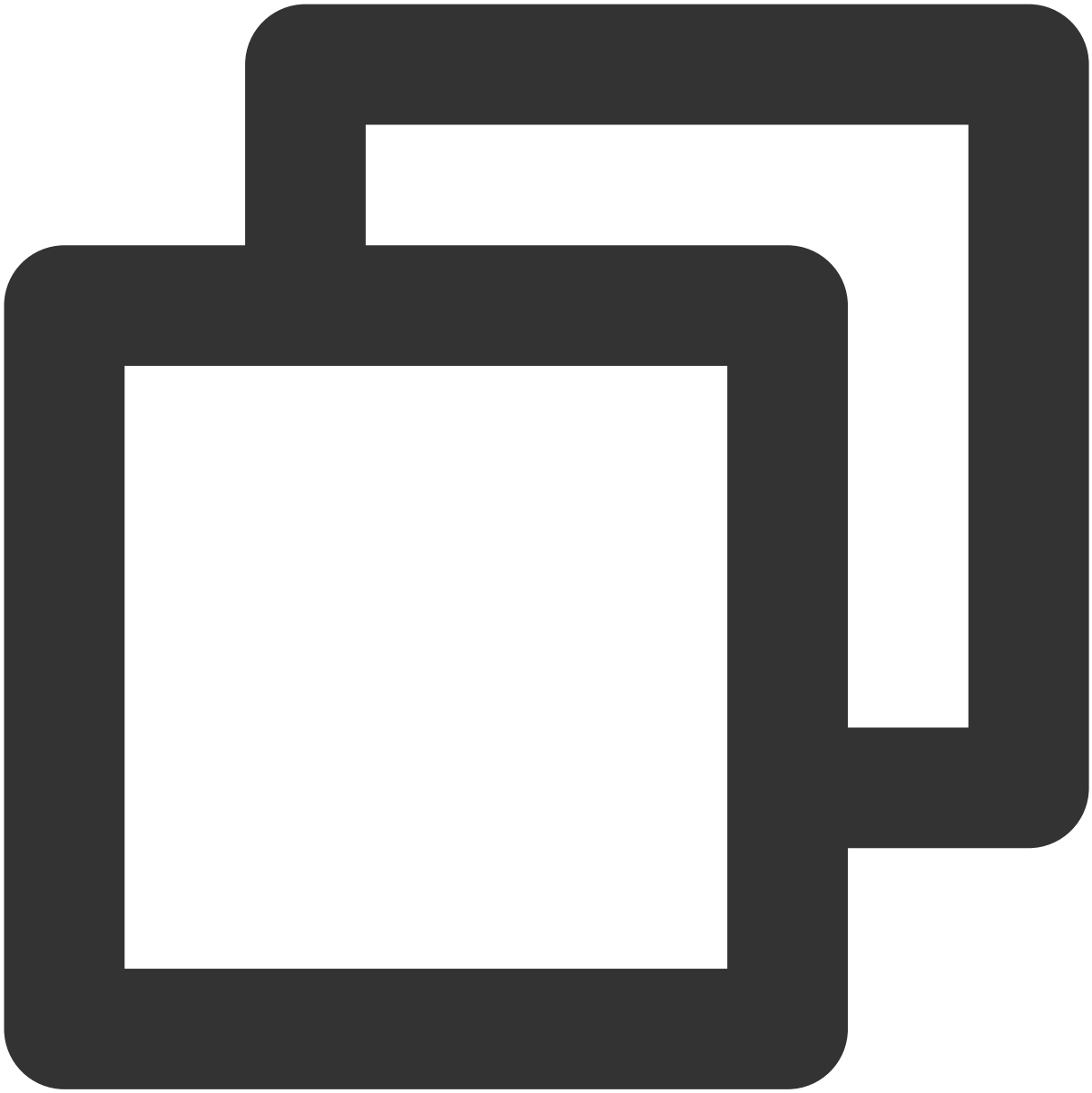


```
Page({
  data: {
    foo: 'my-foo',
    bar: 'my-bar'
  }
})
```

The end result is the object {foo: 'my-foo', bar:'my-bar'}.

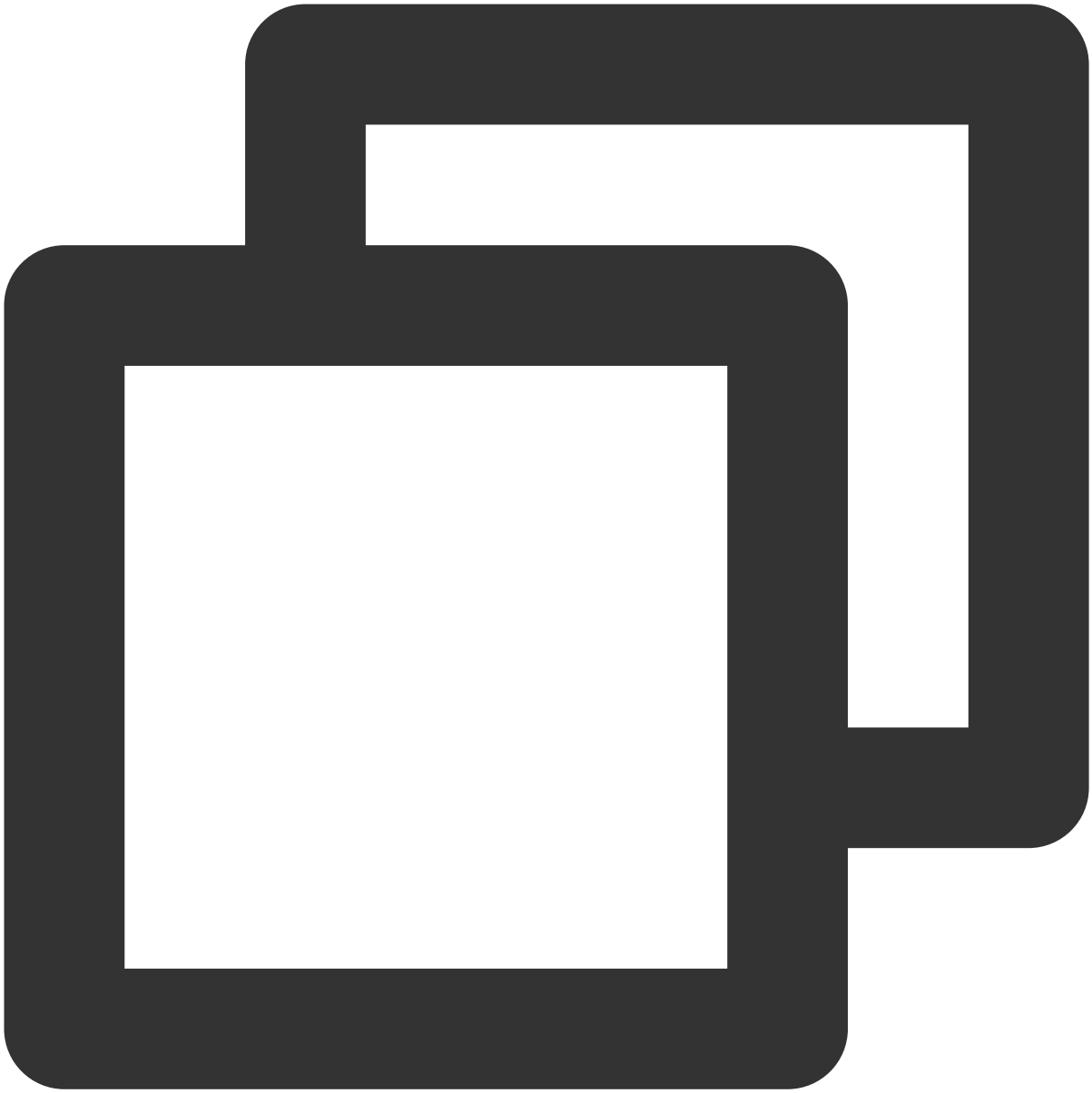
Any combination of the above methods is allowed, but if there are cases where identical variable names occur, the latter will overwrite the former, such as:

©2013-2022 Tencent Cloud. All rights reserved.



```
<template is="objectCombine" data="{{...obj1, ...obj2, a, c: 6}}"></template>
```

©2013-2022 Tencent Cloud. All rights reserved.



```
Page({
  data: {
    obj1: {
      a: 1,
      b: 2
    },
    obj2: {
      b: 3,
      c: 4
    },
    a: 5
```

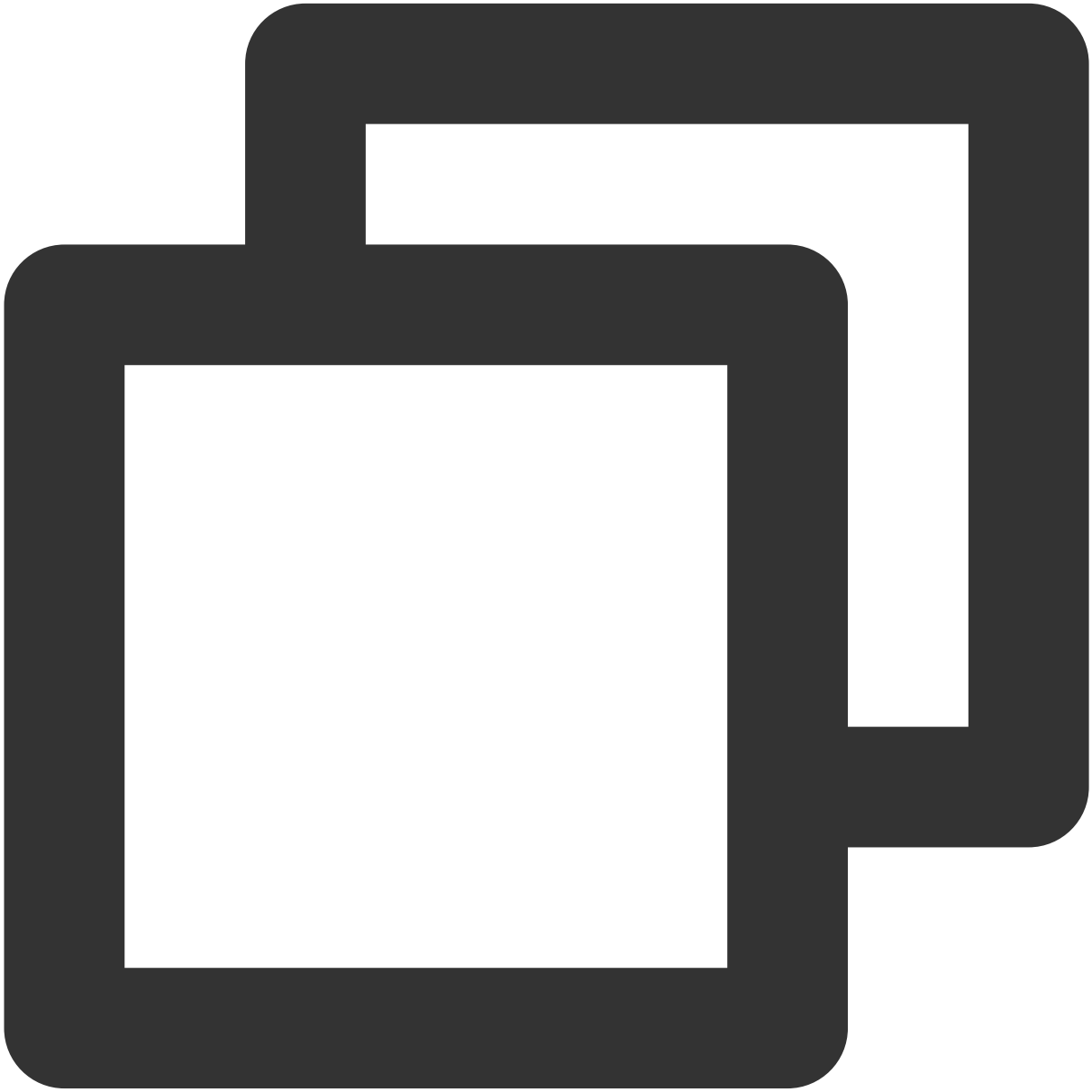
©2013-2022 Tencent Cloud. All rights reserved.

```
  }
})
```

The end result is the object {a: 5, b: 3, c: 6}.

Note:

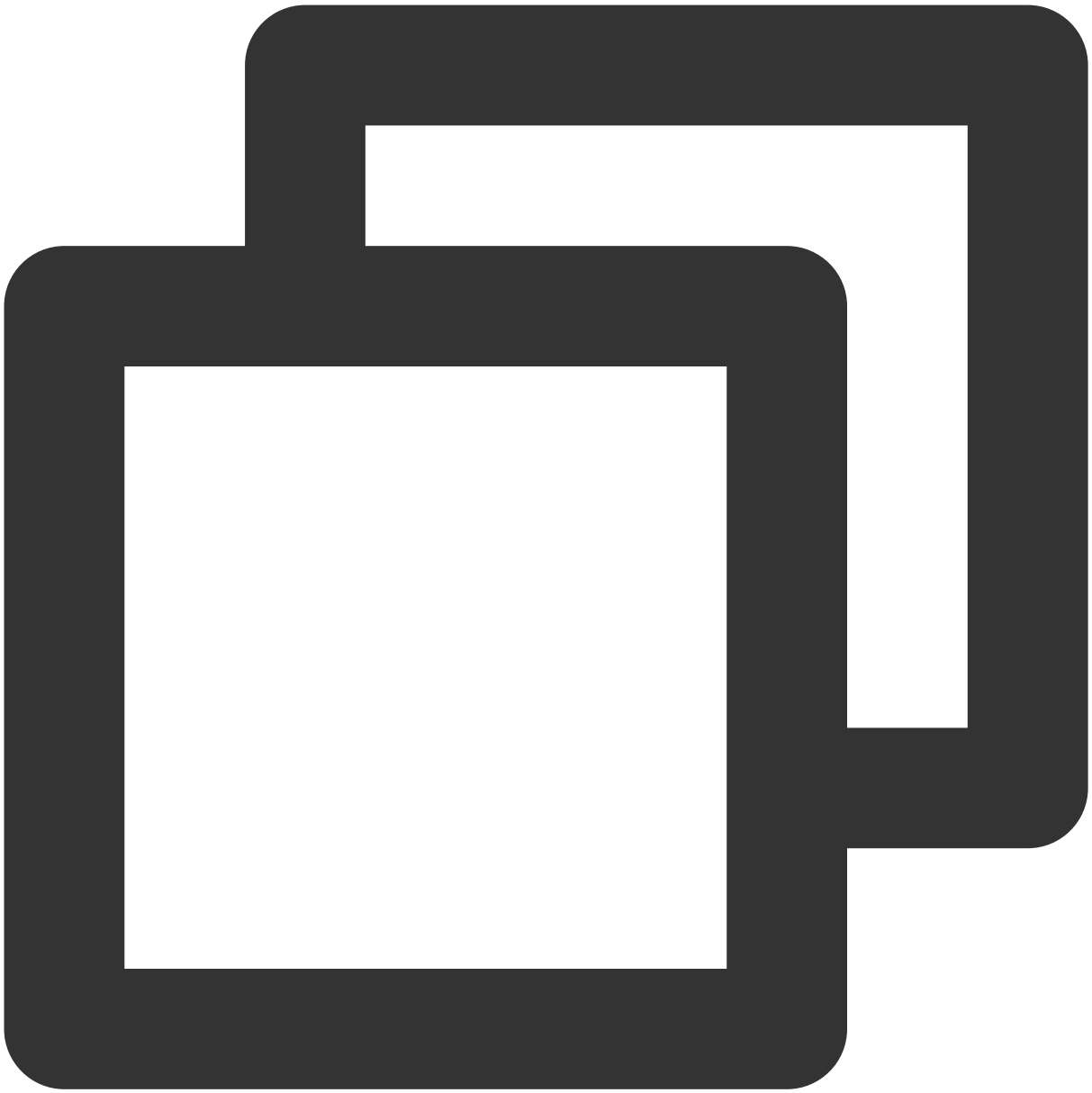
If there are spaces between curly brackets and quotes, the content is ultimately parsed into a string.



```
<view wx:for="{{[1,2,3]}} ">
  {{item}}
</view>
```

©2013-2022 Tencent Cloud. All rights reserved.

equivalent to



```
<view wx:for="{{[1,2,3] + ' '}}">
  {{item}}
</view>
```

©2013-2022 Tencent Cloud. All rights reserved.

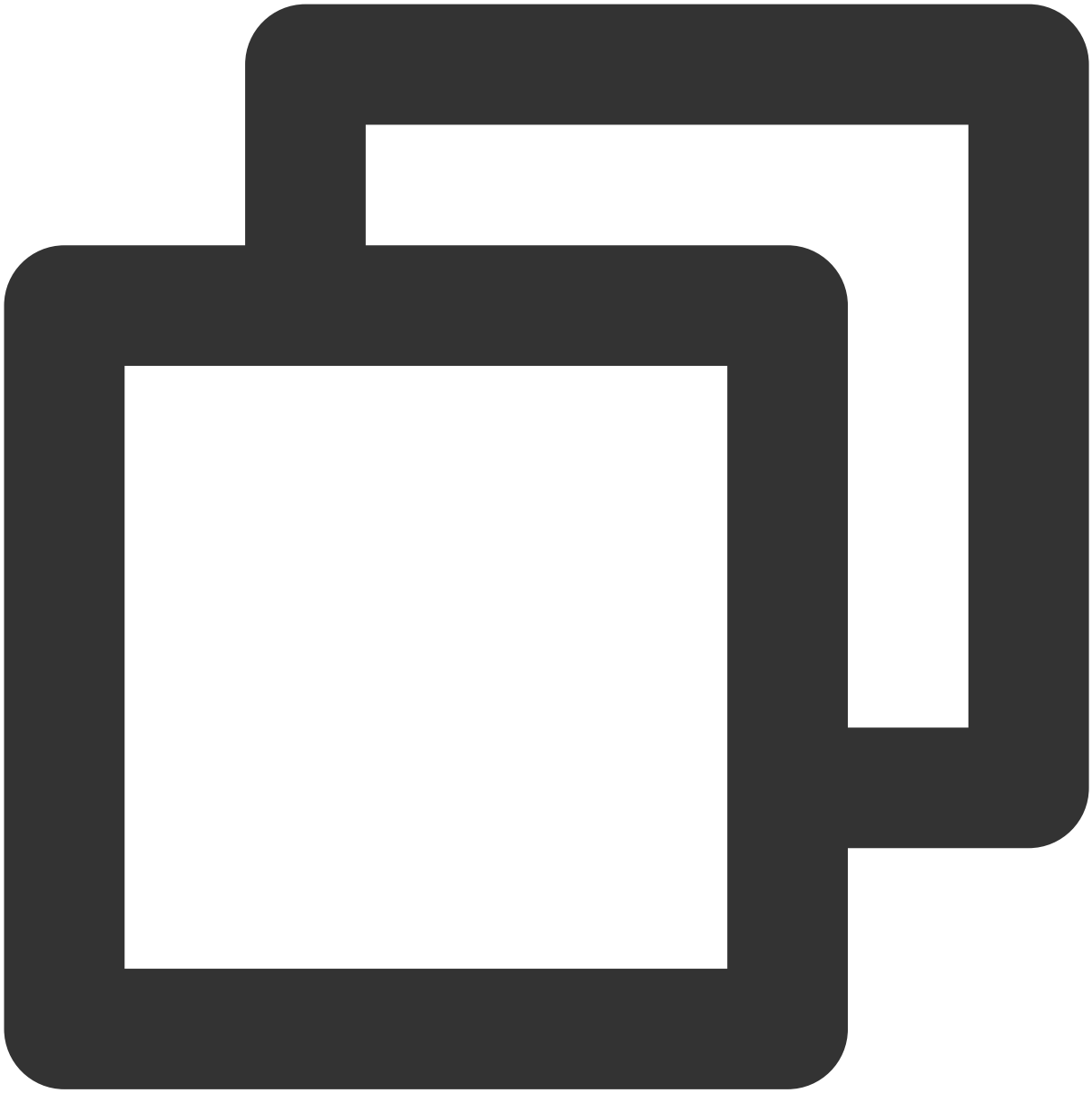
List Rendering

Last updated：2024-07-12 17:05:50

wx:for

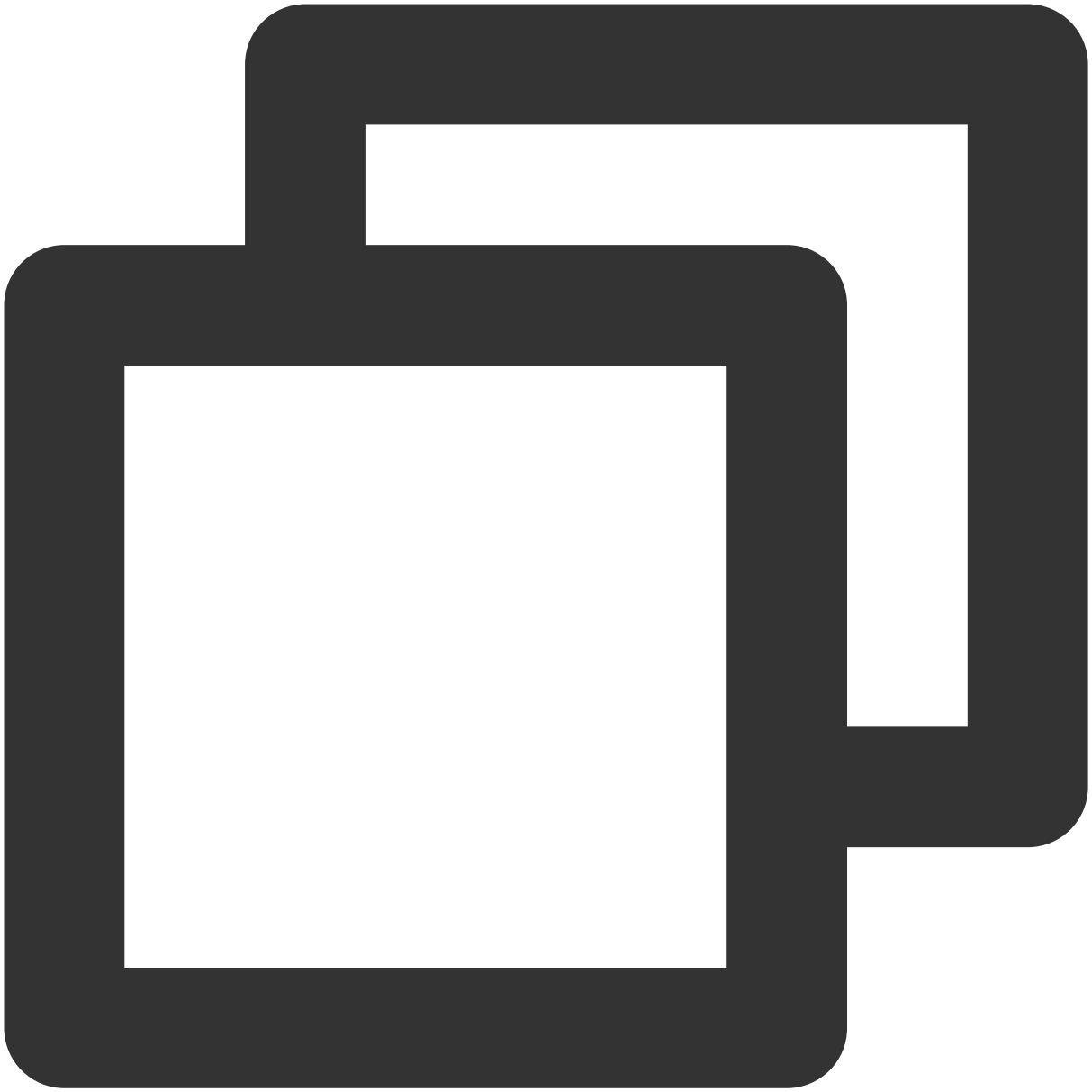
In a component, by binding an array using the wx:for control property, you can use the data of the array items to re-render this component.
The subscript variable name of the current item of the default array defaults to index, and the variable name of the current item of the array defaults to item.

©2013-2022 Tencent Cloud. All rights reserved.



```
<view wx:for="{{array}}">
  {{index}}: {{item.message}}
</view>
```

©2013-2022 Tencent Cloud. All rights reserved.

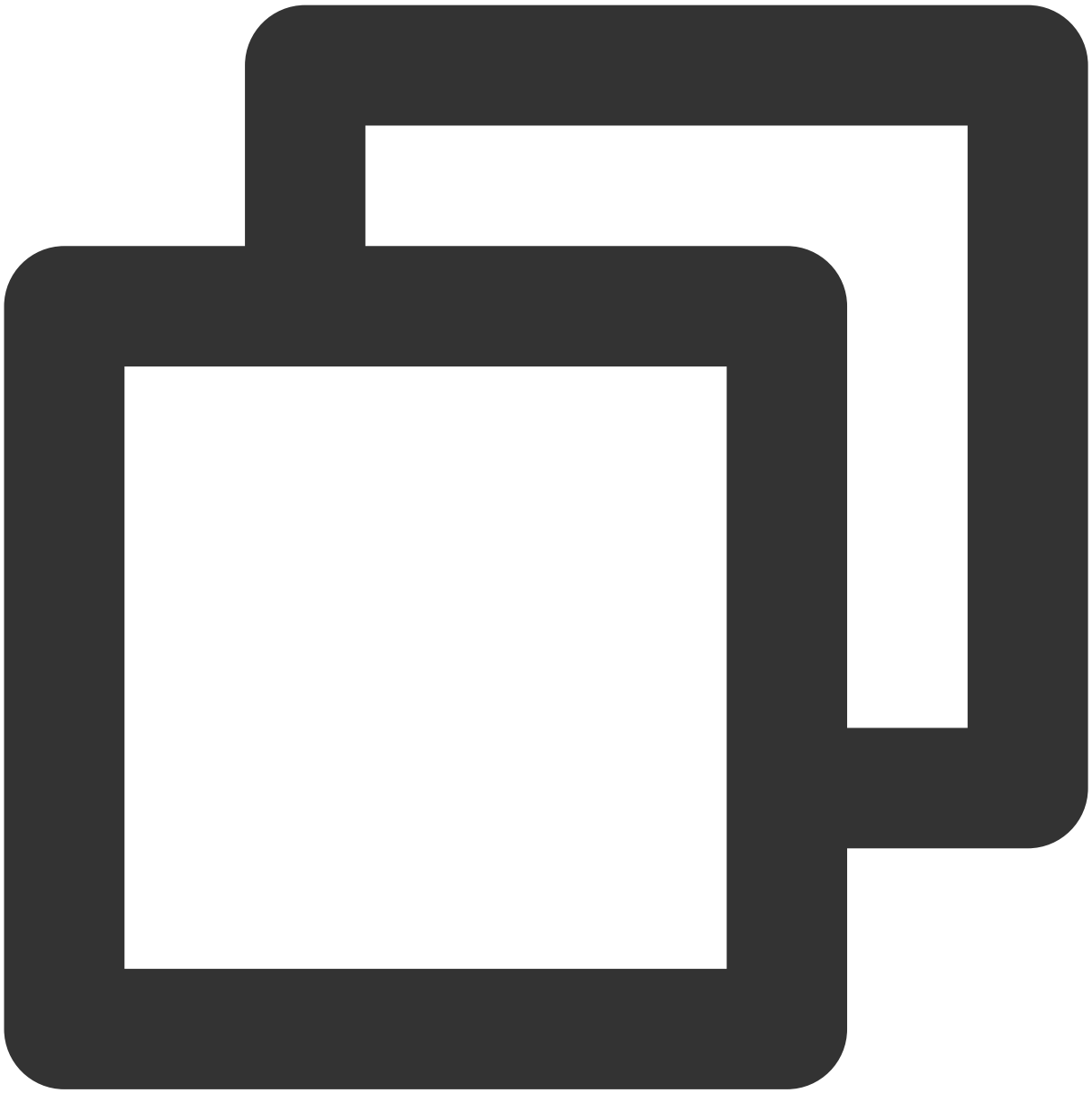


```
Page({
  data: {
    array: [{
      message: 'foo',
    }, {
      message: 'bar'
    }]
  }
})
```

You can use wx:for-item to specify the variable name of the array's current element.

©2013-2022 Tencent Cloud. All rights reserved.

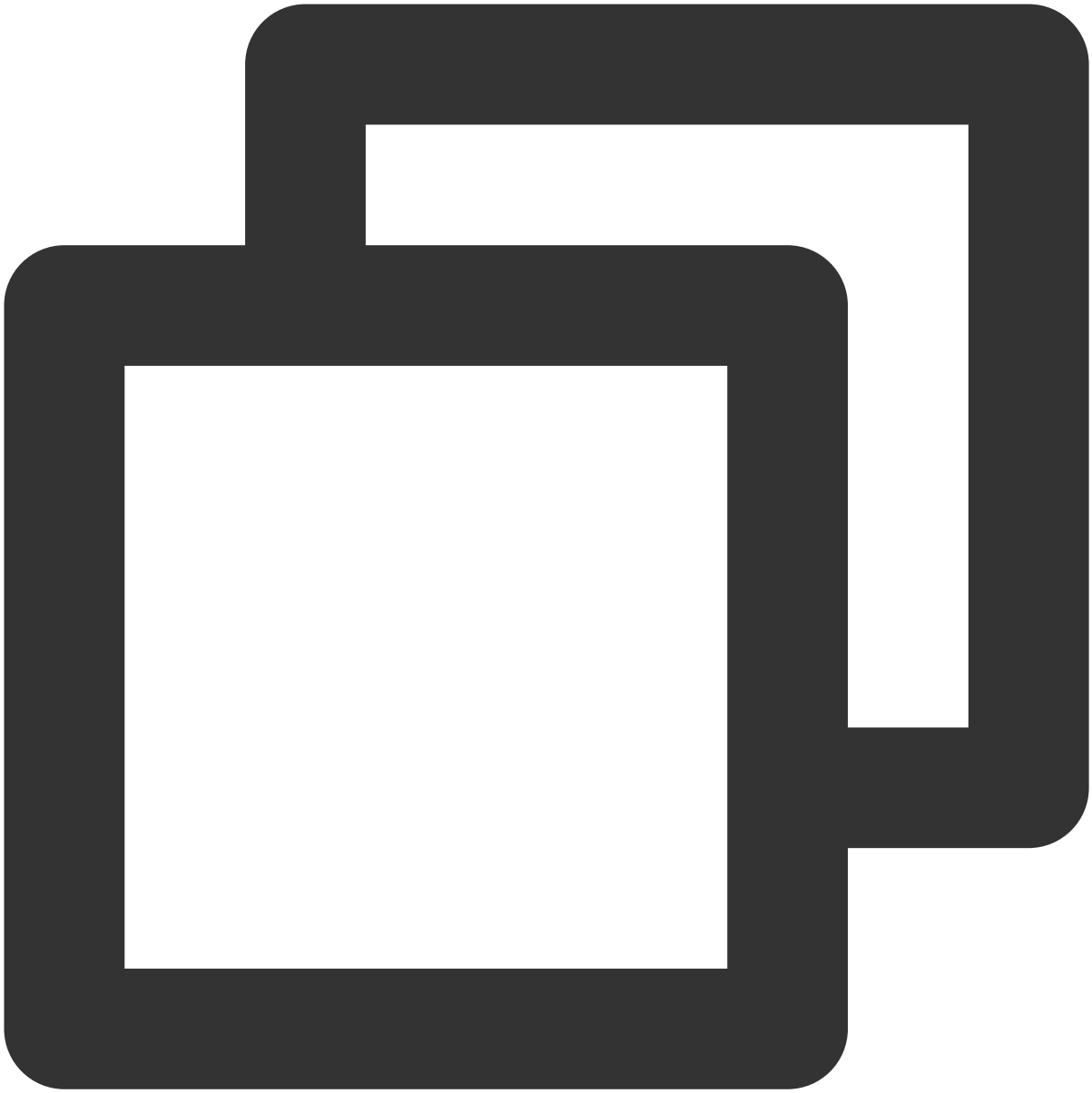
You can use wx:for-index to specify the variable name of the array's current subscript:



```
<view wx:for="{{array}}" wx:for-index="idx" wx:for-item="itemName">
  {{idx}}: {{itemName.message}}
</view>
```

wx:for can also be embedded. Below is the Chinese multiplication table:

©2013-2022 Tencent Cloud. All rights reserved.

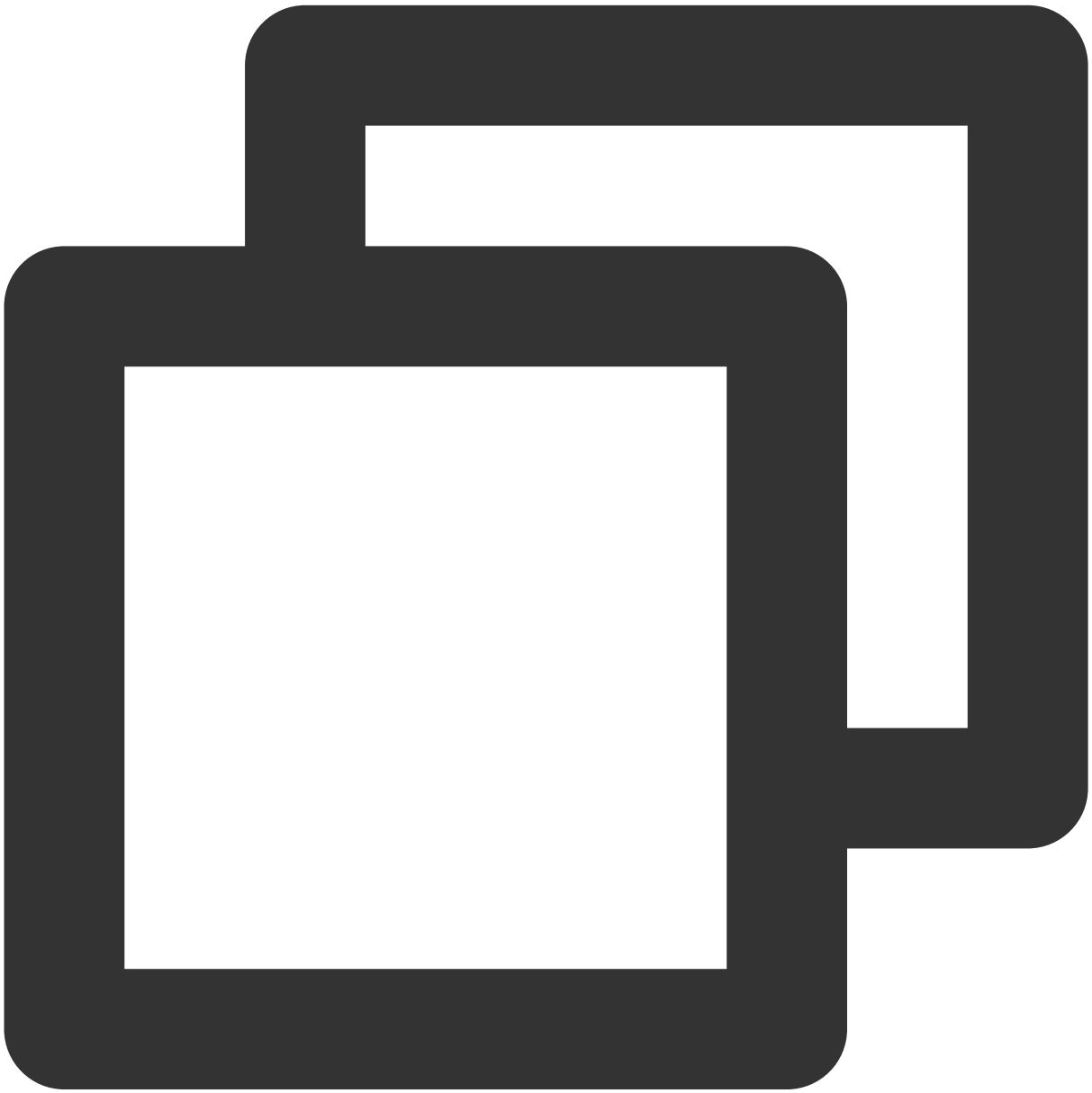


```
<view wx:for="{{[1, 2, 3, 4, 5, 6, 7, 8, 9]}}" wx:for-item="i">
  <view wx:for="{{[1, 2, 3, 4, 5, 6, 7, 8, 9]}}" wx:for-item="j">
    <view wx:if="{{i <= j}}">
      {{i}} * {{j}} = {{i * j}}
    </view>
  </view>
</view>
```

©2013-2022 Tencent Cloud. All rights reserved.

block wx:for

Similar to block wx:if, you can use wx:for on a <block/> tag to render a structural block containing multiple nodes. For example:



```
<block wx:for="{{[1, 2, 3]}}">
  <view> {{index}}: </view>
  <view> {{item}} </view>
</block>
```

©2013-2022 Tencent Cloud. All rights reserved.

wx:key

If the positions of list items dynamically change or new items are added to the list and you want the items in the list to retain their features and statuses (such as the content input in input and the selection status of switch), you must use wx:key to specify the unique identifiers of the items in the list.

The wx:key is provided in two formats:

String: Represents a property of an item in the for loop array. The value of this property must be a unique string or number in the list and cannot dynamically change.

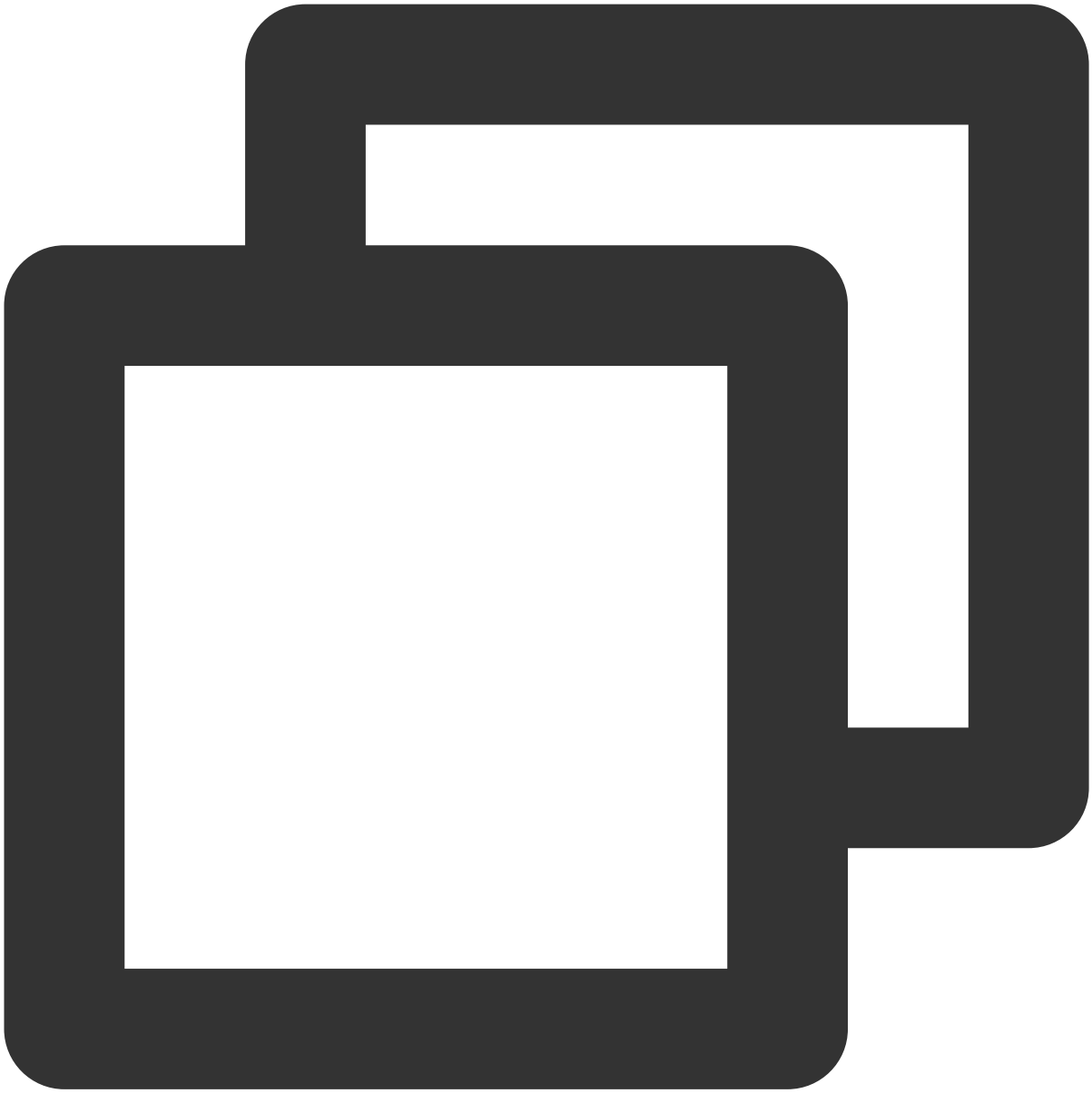
Reserved keyword `*this` : Represents the item itself in the for loop. This expresses that the item itself is a unique string or number. For example:

When a data change triggers re-rending at the rendering layer, the components that have keys are corrected. The framework ensures that they are reordered, rather than recreated. This ensures the components retain their statuses and improves the efficiency of list rendering.

If wx:key, is not provided, a warning is reported. If you know that this list is static or the order is not important, you can ignore the warning.

Sample code

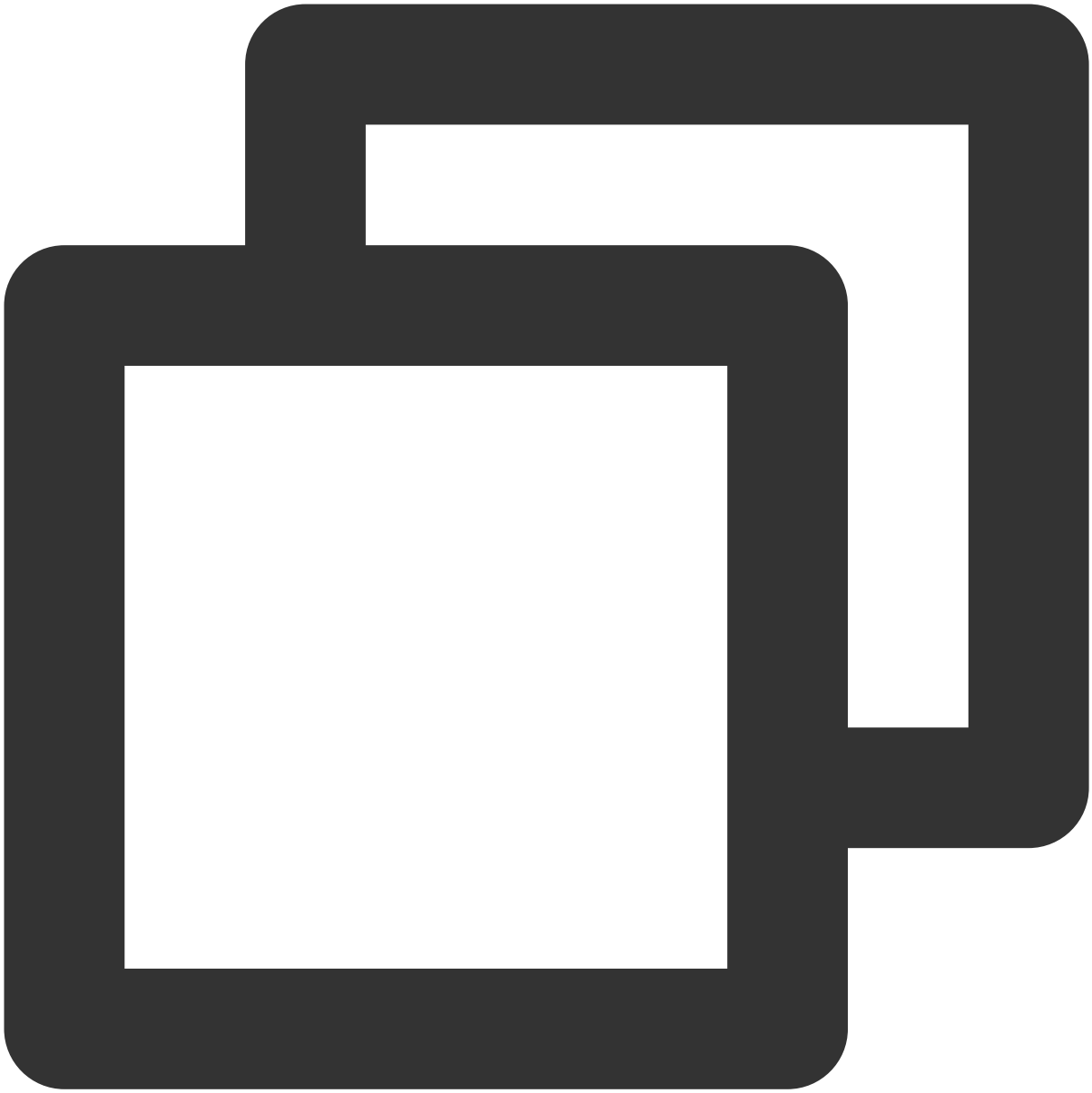
©2013-2022 Tencent Cloud. All rights reserved.



```
<switch wx:for="{{objectArray}}" wx:key="unique" style="display: block;"> {{item.id
<button bindtap="switch"> Switch </button>
<button bindtap="addToFront"> Add to the front </button>

<switch wx:for="{{numberArray}}" wx:key="*this" style="display: block;"> {{item}} <
<button bindtap="addNumberToFront"> Add to the front </button>
```

©2013-2022 Tencent Cloud. All rights reserved.



```
Page({
  data: {
    objectArray: [
      {id: 5, unique: 'unique_5'},
      {id: 4, unique: 'unique_4'},
      {id: 3, unique: 'unique_3'},
      {id: 2, unique: 'unique_2'},
      {id: 1, unique: 'unique_1'},
      {id: 0, unique: 'unique_0'},
    ],
    numberArray: [1, 2, 3, 4]
```

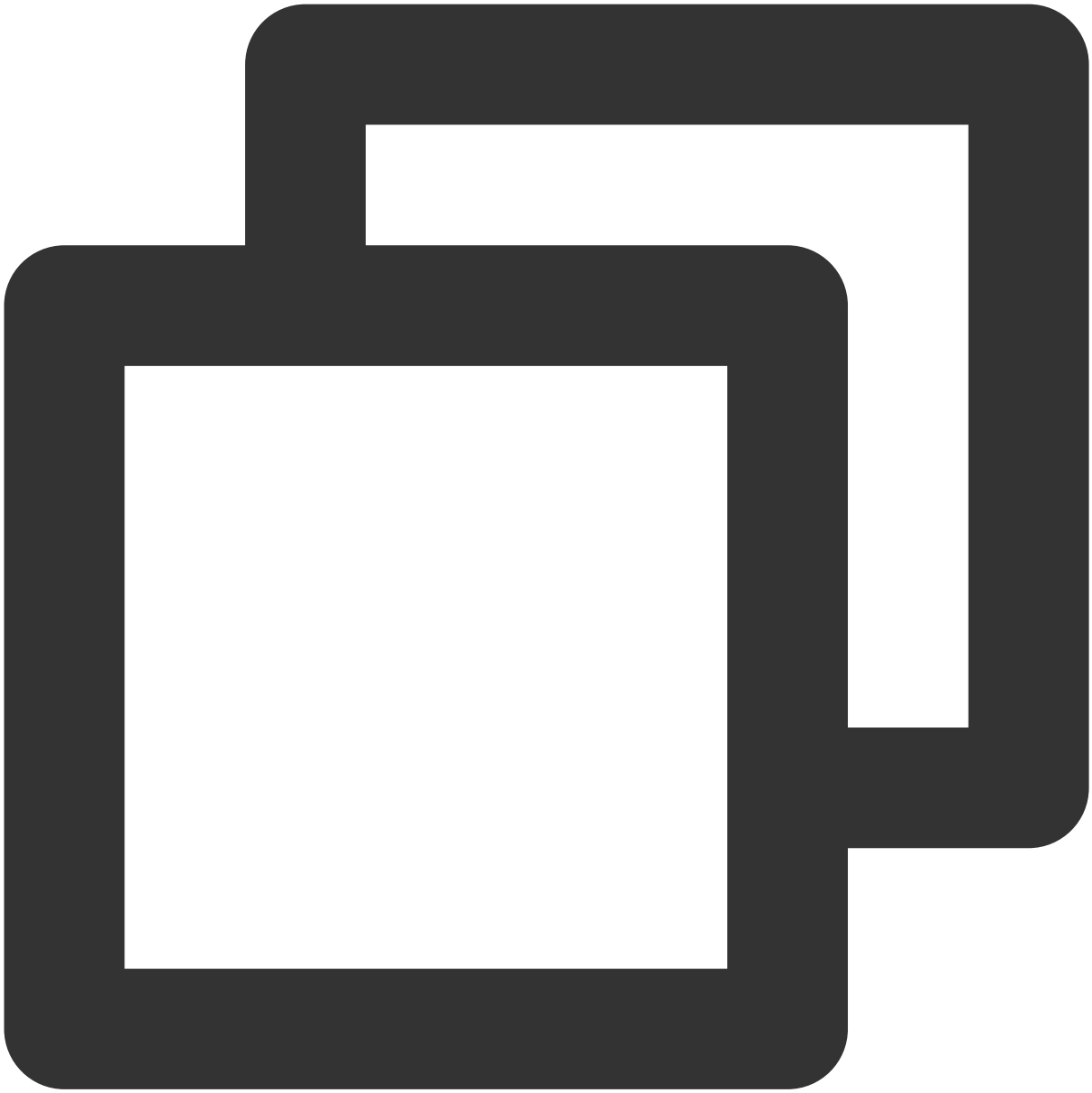
©2013-2022 Tencent Cloud. All rights reserved.

```
  },
  switch: function(e) {
    const length = this.data.objectArray.length
    for (let i = 0; i < length; ++i) {
      const x = Math.floor(Math.random() * length)
      const y = Math.floor(Math.random() * length)
      const temp = this.data.objectArray[x]
      this.data.objectArray[x] = this.data.objectArray[y]
      this.data.objectArray[y] = temp
    }
    this.setData({
      objectArray: this.data.objectArray
    })
  },
  addToFront: function(e) {
    const length = this.data.objectArray.length
    this.data.objectArray = [{id: length, unique: 'unique_' + length}].concat(this.
    this.setData({
      objectArray: this.data.objectArray
    })
  },
  addNumberToFront: function(e){
    this.data.numberArray = [ this.data.numberArray.length + 1 ].concat(this.data.n
    this.setData({
      numberArray: this.data.numberArray
    })
  }
})
```

Note

When the wx:for value is a string, the string is parsed into a string array.

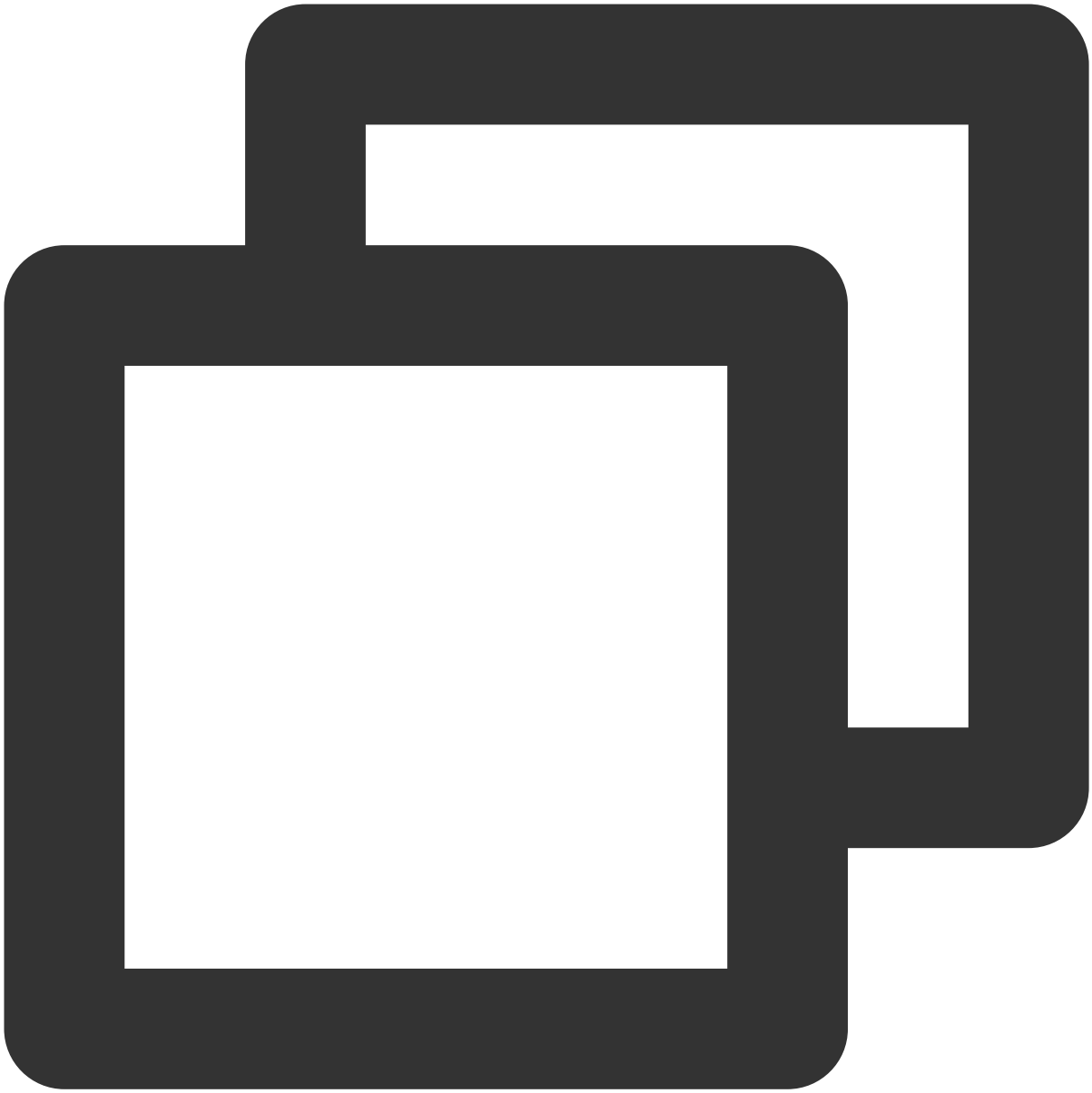
©2013-2022 Tencent Cloud. All rights reserved.



```
<view wx:for="array">
  {{item}}
</view>
```

equivalent to

©2013-2022 Tencent Cloud. All rights reserved.

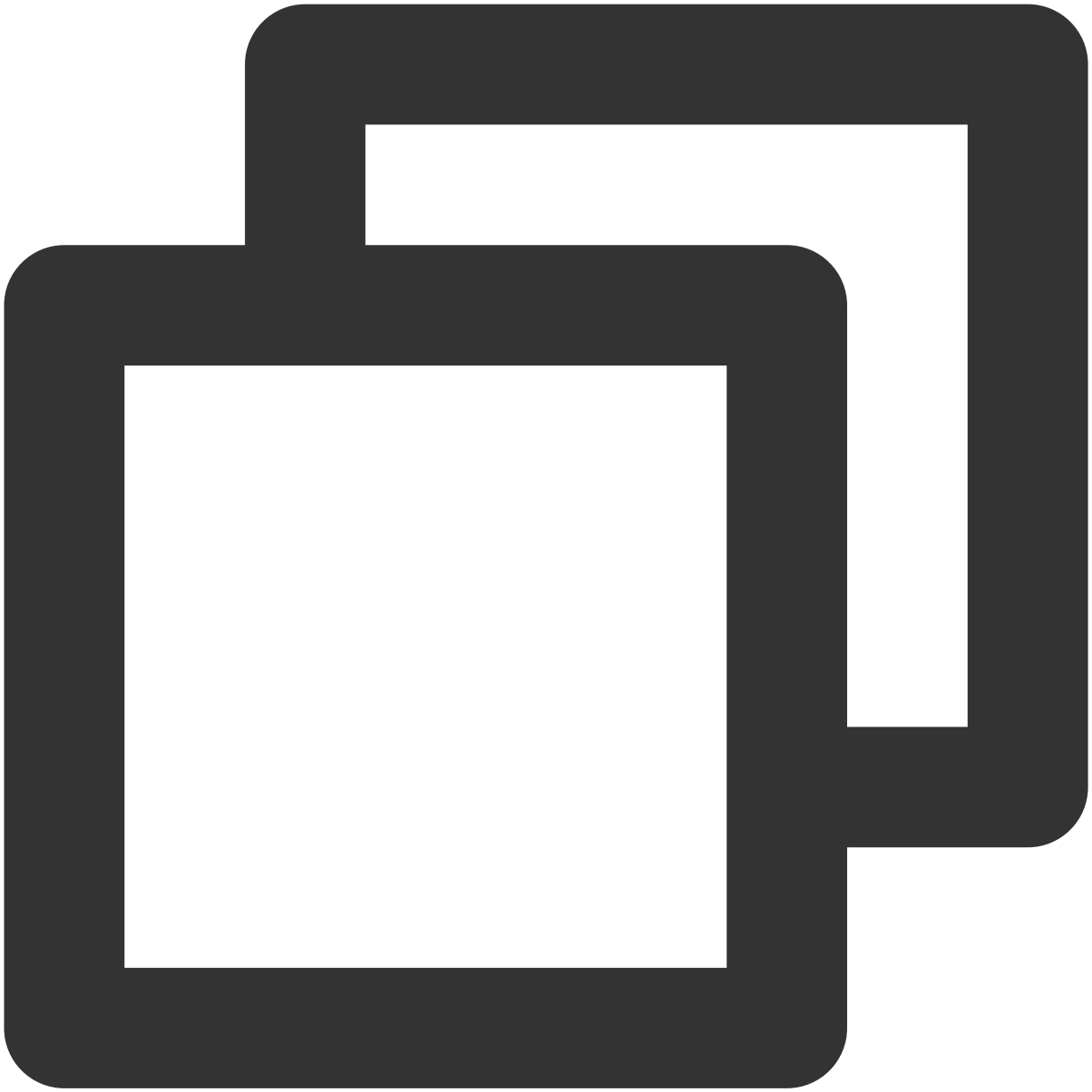


```
<view wx:for="{{['a','r','r','a','y']}}">
  {{item}}
</view>
```

Note:

If there are spaces between curly brackets and quotes, the content is ultimately parsed into a string.

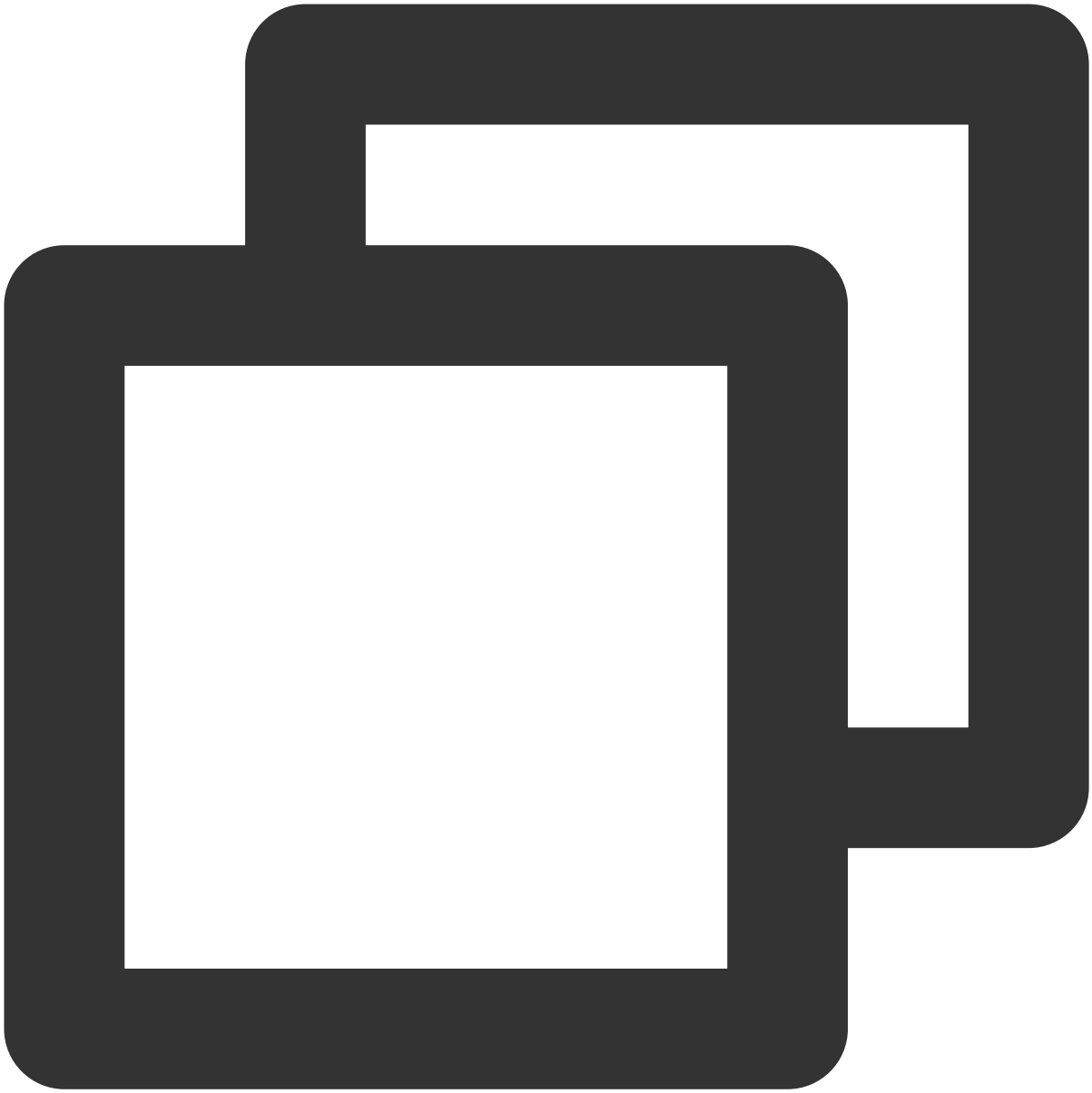
©2013-2022 Tencent Cloud. All rights reserved.



```
<view wx:for="{{[1,2,3]}} ">
  {{item}}
</view>
```

equivalent to

©2013-2022 Tencent Cloud. All rights reserved.



```
<view wx:for="{{[1,2,3] + ' '}}" >
  {{item}}
</view>
```

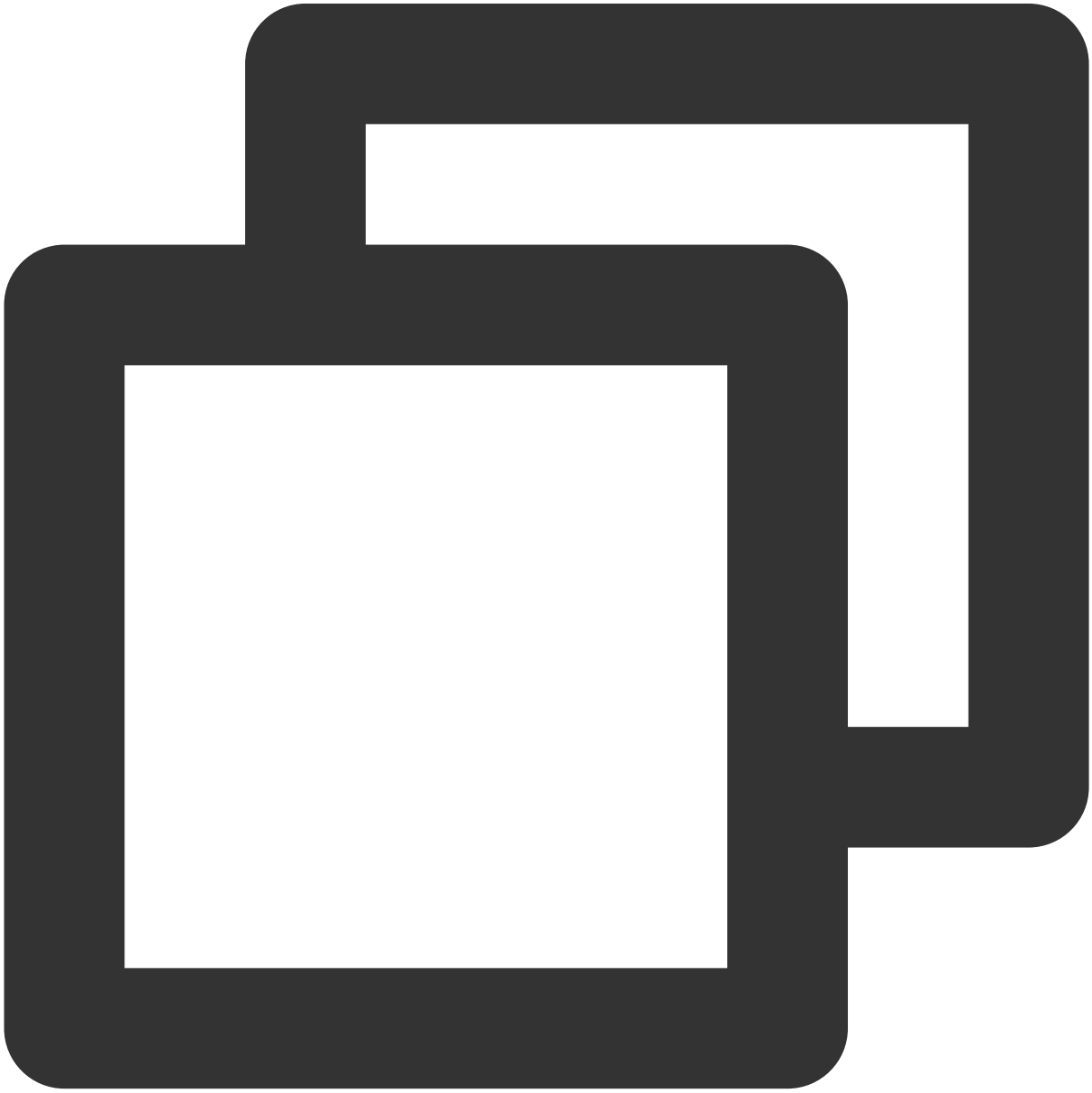
©2013-2022 Tencent Cloud. All rights reserved.

Conditional Rendering

Last updated：2024-07-12 17:02:09

wx:if

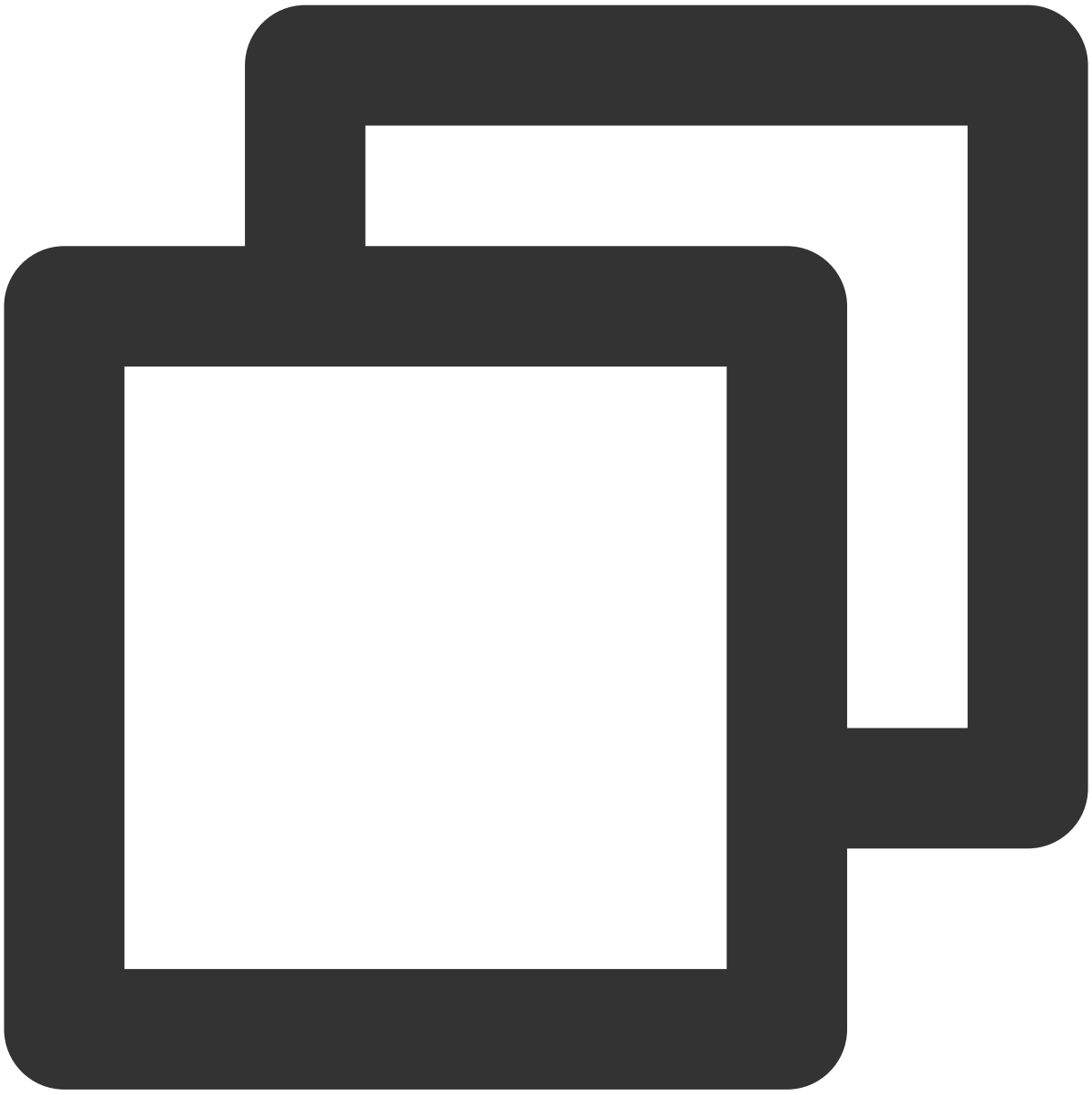
In the framework, use `wx:if=""` to determine whether or not to render this code block:



```
<view wx:if="{{condition}}"> True </view>
```

©2013-2022 Tencent Cloud. All rights reserved.

You can also use `wx:elif` and `wx:else` to add an “else” block:

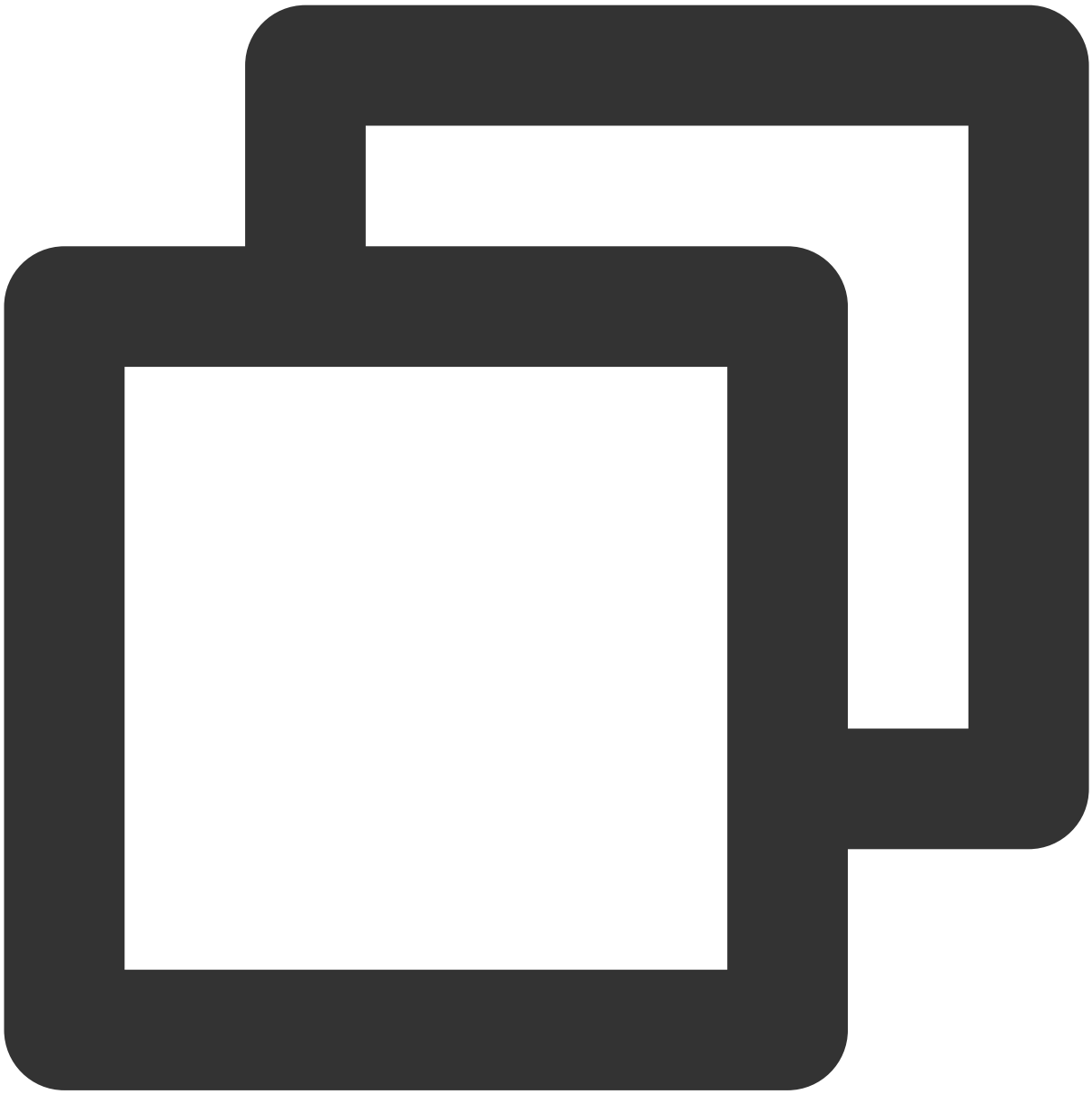


```
<view wx:if="{{length > 5}}"> 1 </view>
<view wx:elif="{{length > 2}}"> 2 </view>
<view wx:else> 3 </view>
```

block wx:if

©2013-2022 Tencent Cloud. All rights reserved.

Because `wx:if` is a control property, you must add it to a tag. To judge multiple component tags at once, you can use a `<block/>` tag to package multiple components together and use the wx:if control property above.



```
<block wx:if="{{true}}">
  <view> view1 </view>
  <view> view2 </view>
</block>
```

Note:

©2013-2022 Tencent Cloud. All rights reserved.

`<block/>` is not a component, just a package element. It does not perform any rendering on the page and only accepts control properties.

wx:if vs hidden

Because the template in `wx:if` can also contain data bindings, when the `wx:if` condition value changes, the framework has a local rendering process that will ensure the conditional block is destroyed or re-rendered when the value changes.
At the same time, `wx:if` is also inert. If the condition is false during initial rendering, the framework does not render it. When the condition changes to true for the first time, local rendering starts.
In contrast, hidden is much simpler. The component is always rendered and this property simply controls whether the content is displayed or hidden.
Generally, `wx:if` has higher switching costs, while hidden has higher initial rendering costs. Therefore, in situations with frequent condition changes, it is better to use hidden. If conditions are unlikely to change during runtime, `wx:if` is the better choice.

©2013-2022 Tencent Cloud. All rights reserved.

Templates

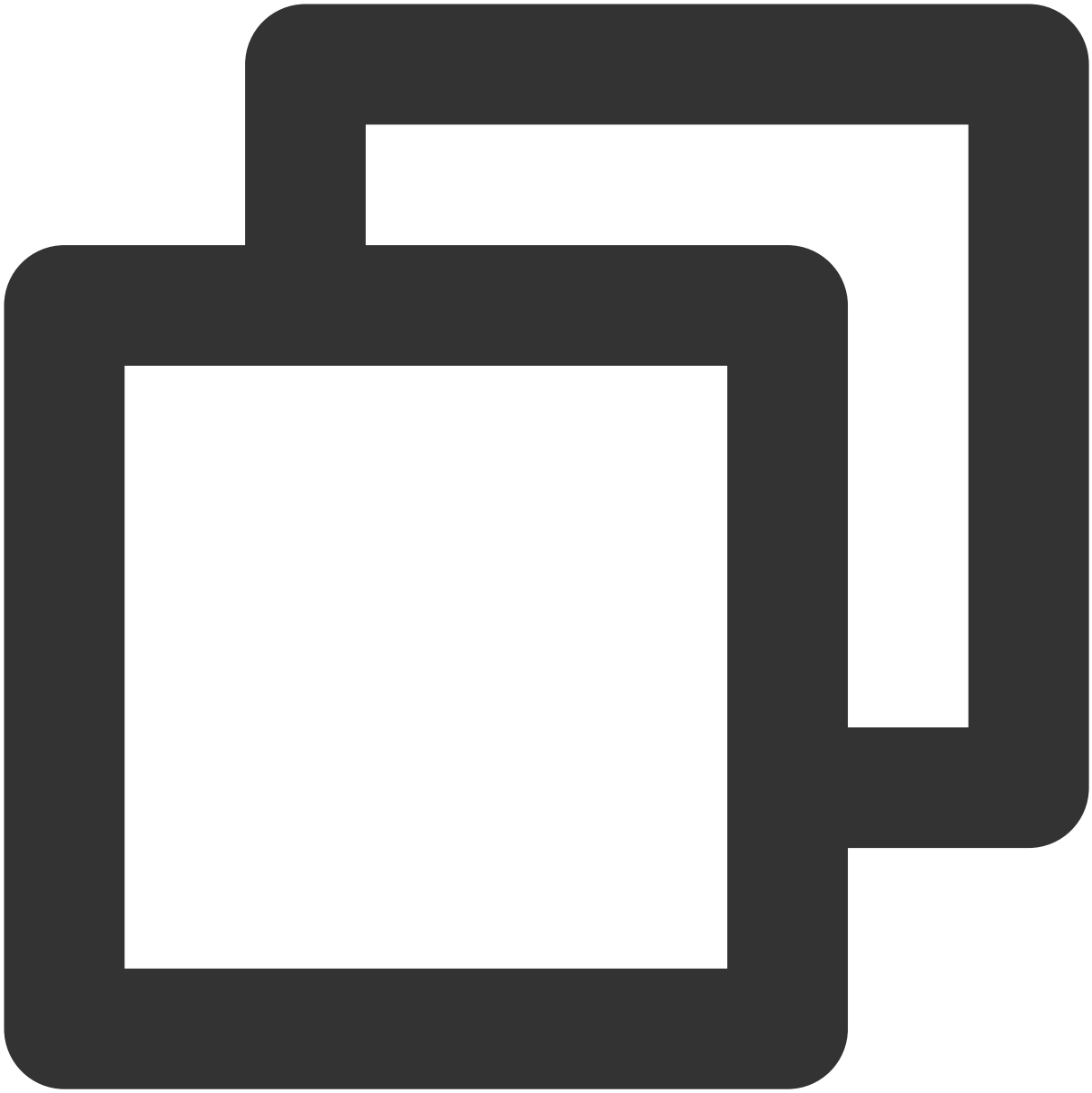
Last updated：2024-07-12 17:02:48

WXML provides templates, in which you can define code snippets and then call them in different places.

Defining a Template

Use the name property to specify the name of the `template`. Then, define a code snippet in <template/>. For example:

©2013-2022 Tencent Cloud. All rights reserved.

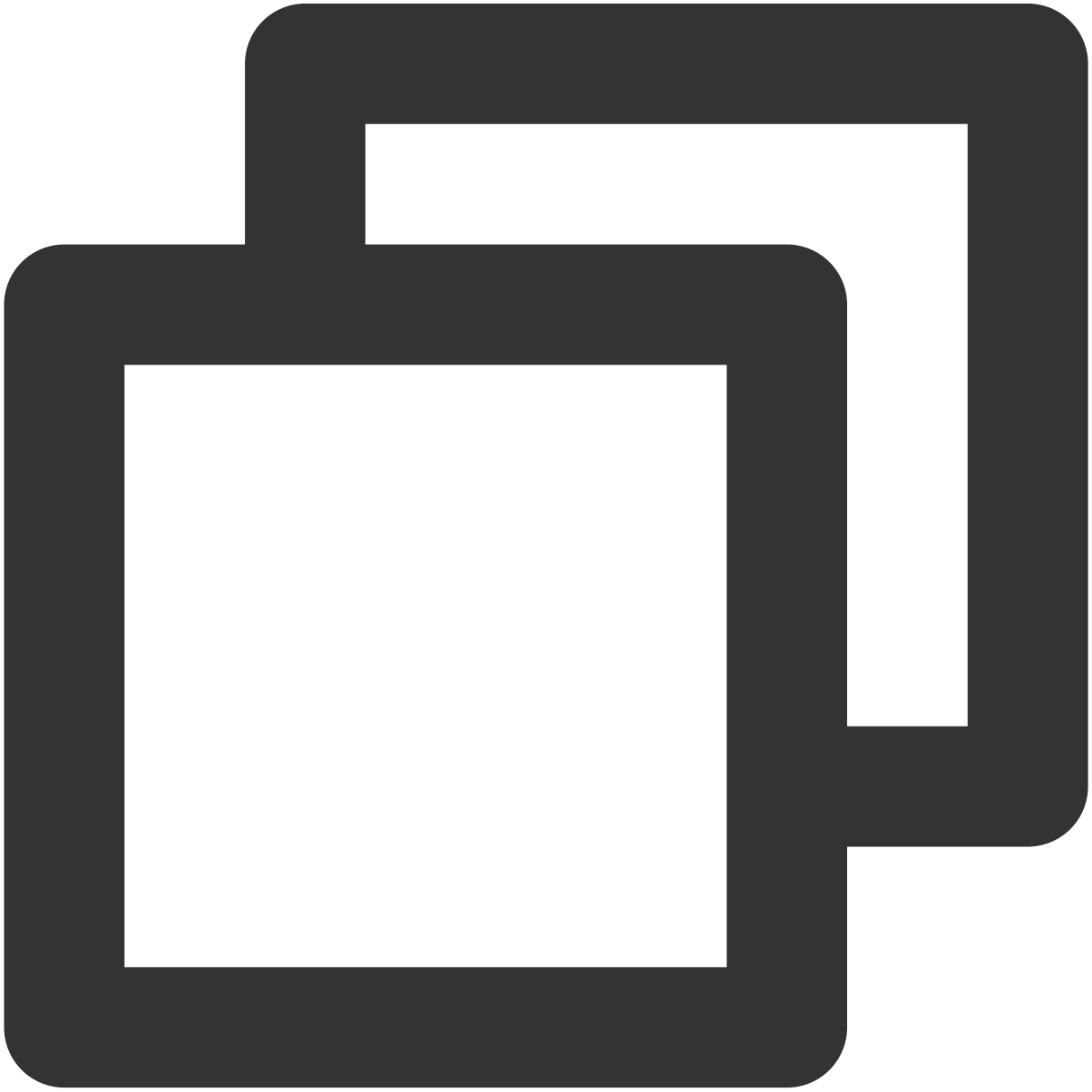


```
<!--
  index: int
  msg: string
  time: string
-->
<template name="msgItem">
  <view>
    <text> {{index}}: {{msg}} </text>
    <text> Time: {{time}} </text>
  </view>
</template>
```

©2013-2022 Tencent Cloud. All rights reserved.

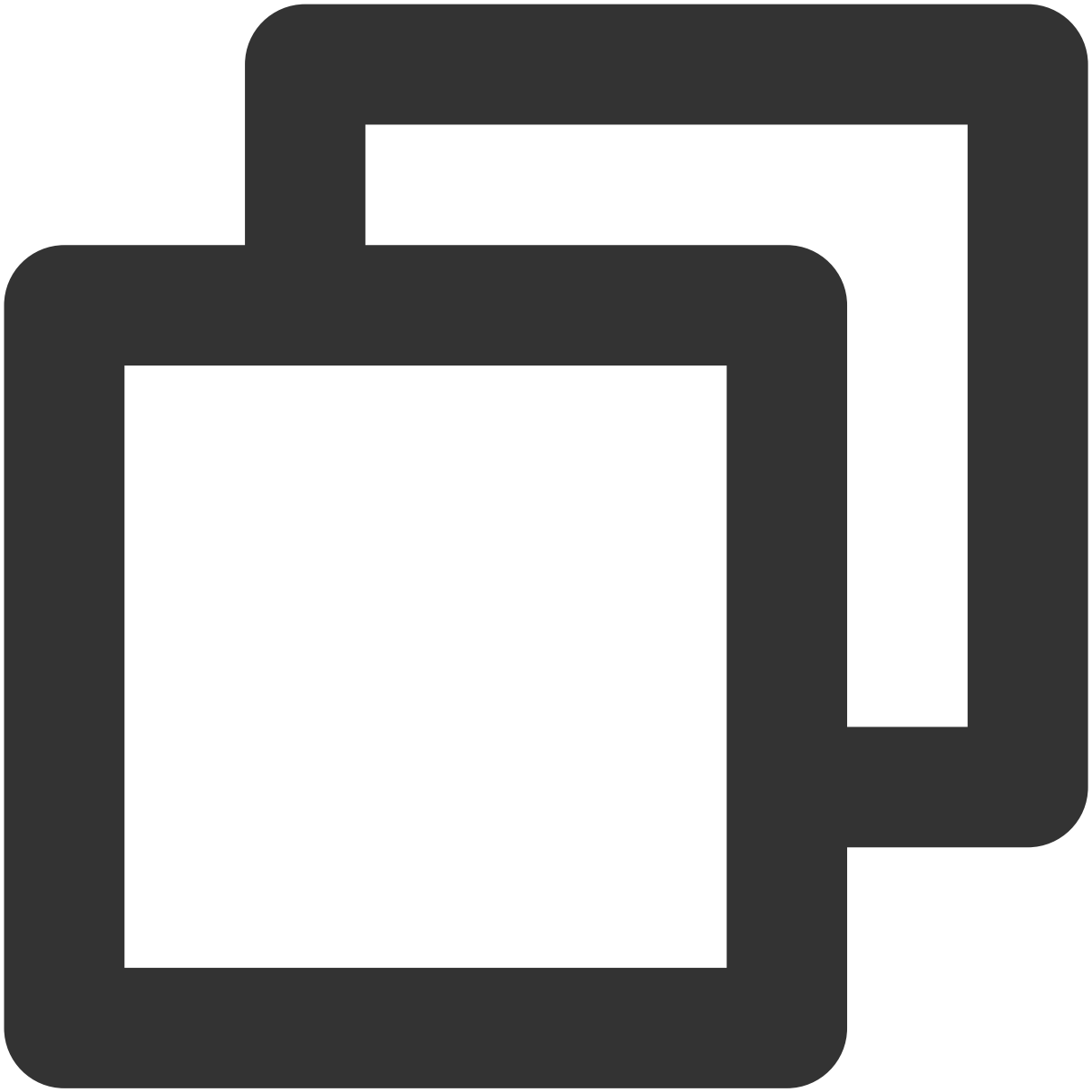
Using the Template

Using the is attribute, declare the template you want to use, and then pass in the data the template needs, for example:



```
<template is="msgItem" data="{{...item}}"/>
```

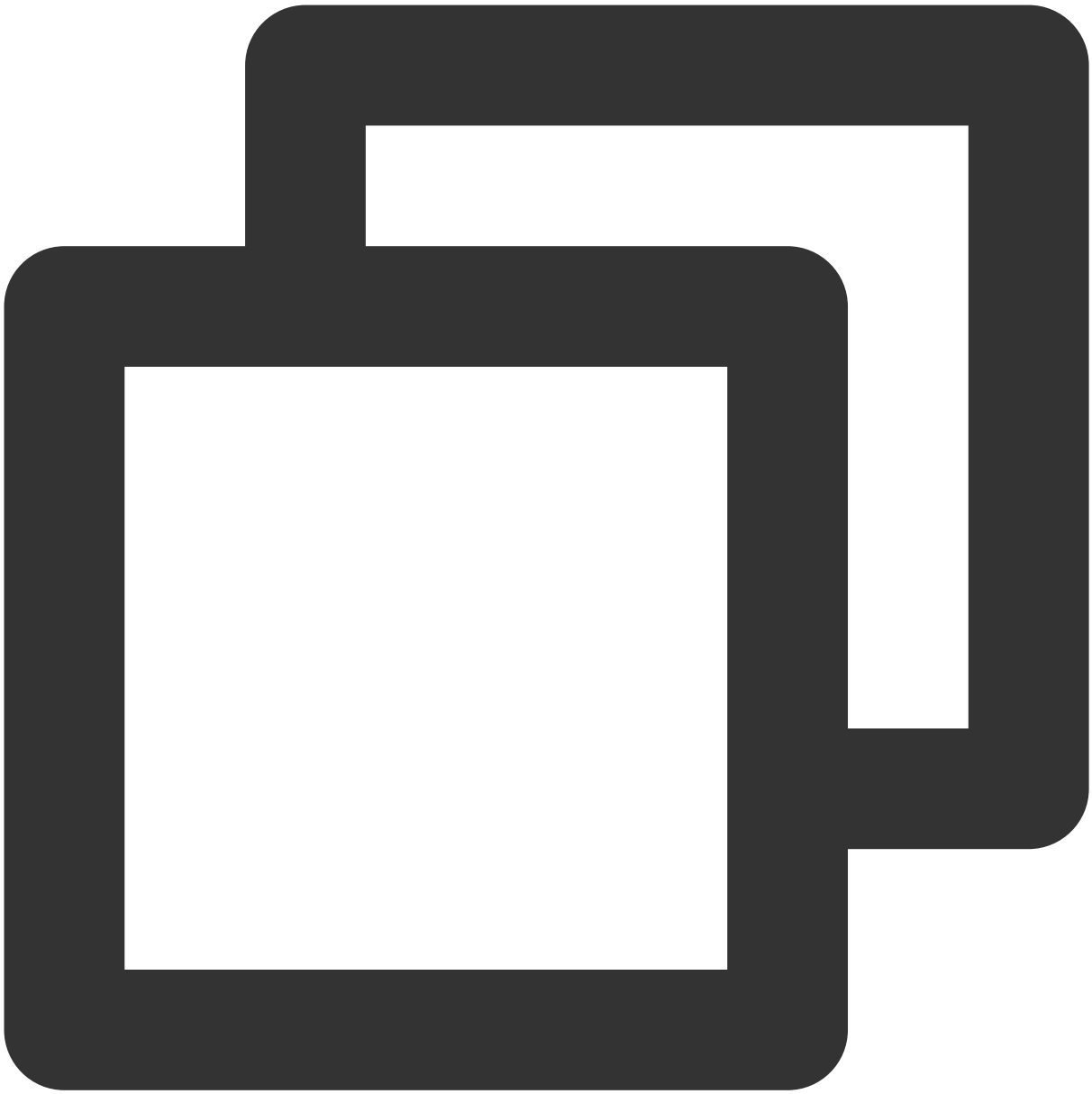
©2013-2022 Tencent Cloud. All rights reserved.



```
Page({
  data: {
    item: {
      index: 0,
      msg: 'this is a template',
      time: '2016-09-15'
    }
  }
})
```

©2013-2022 Tencent Cloud. All rights reserved.

The `is` property can use Mustache syntax to dynamically determine the template to be rendered:



```
<template name="odd">
  <view> odd </view>
</template>
<template name="even">
  <view> even </view>
</template>

<block wx:for="{{[1, 2, 3, 4, 5]}}">
  <template is="{{item % 2 == 0 ? 'even' : 'odd'}}"/>
```

©2013-2022 Tencent Cloud. All rights reserved.

```
</block>
```

Template Scope

Templates have their own scopes. They can only use the data passed by data and the <wxs /> module defined in the file defined by the template.

©2013-2022 Tencent Cloud. All rights reserved.

Referencing

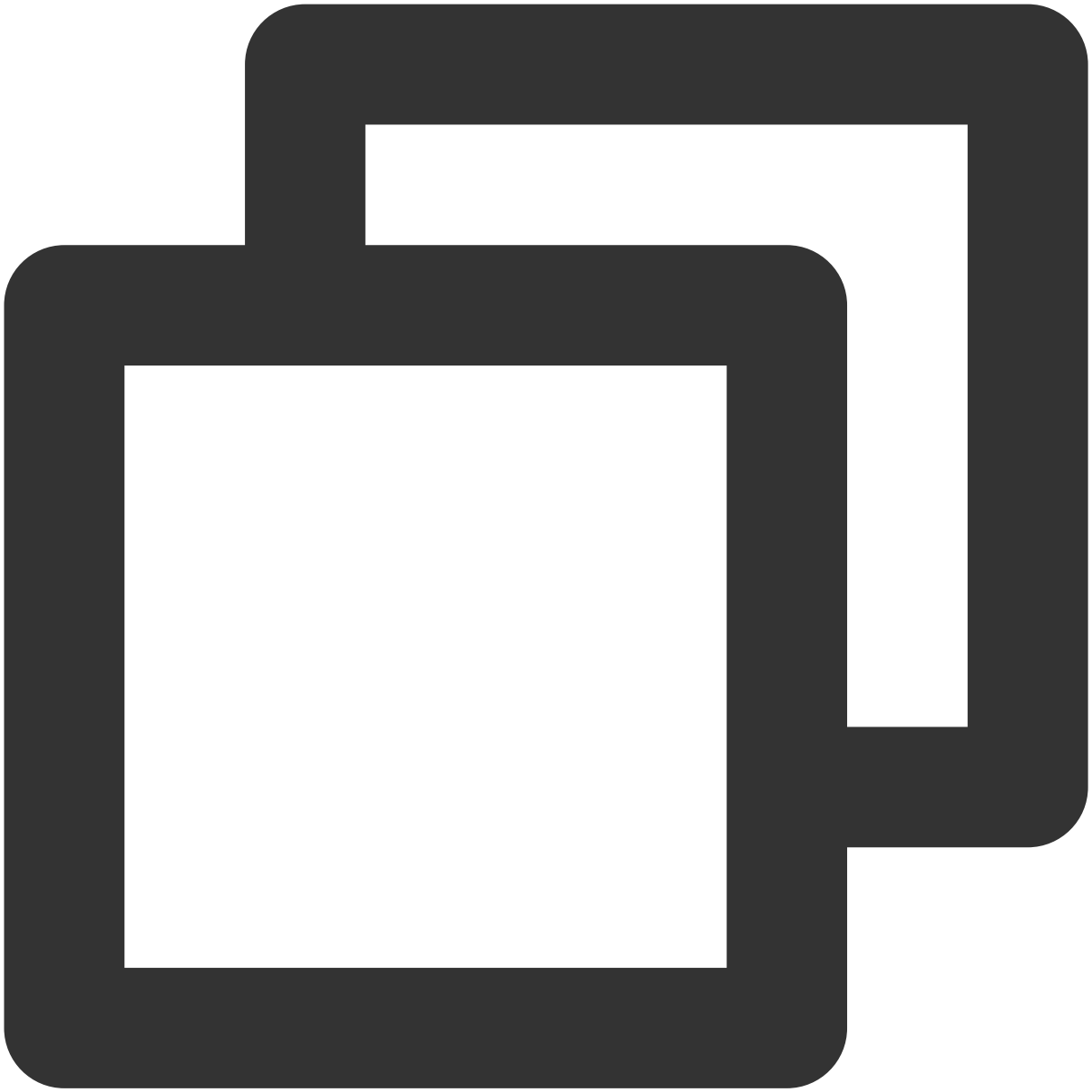
Last updated：2024-07-12 17:00:21

WXML provides two referencing methods `import` and `include`.

import

import can use the template defined by the target file in the current file, for example:
A template named item is defined in item.wxml:

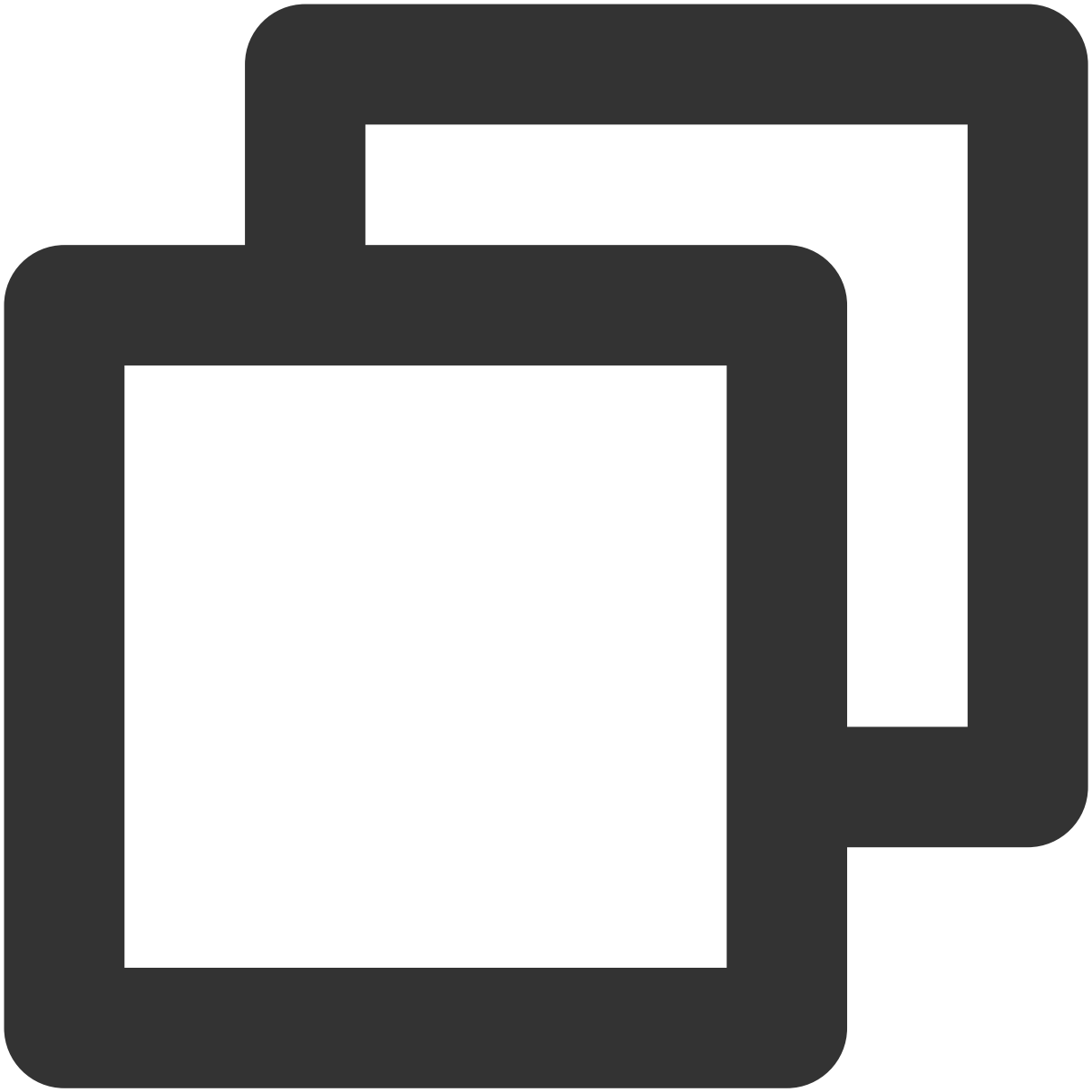
©2013-2022 Tencent Cloud. All rights reserved.



```
<!-- item.wxml -->
<template name="item">
  <text>{{text}}</text>
</template>
```

By referencing item.wxml in index.wxml, you can use the item template:

©2013-2022 Tencent Cloud. All rights reserved.



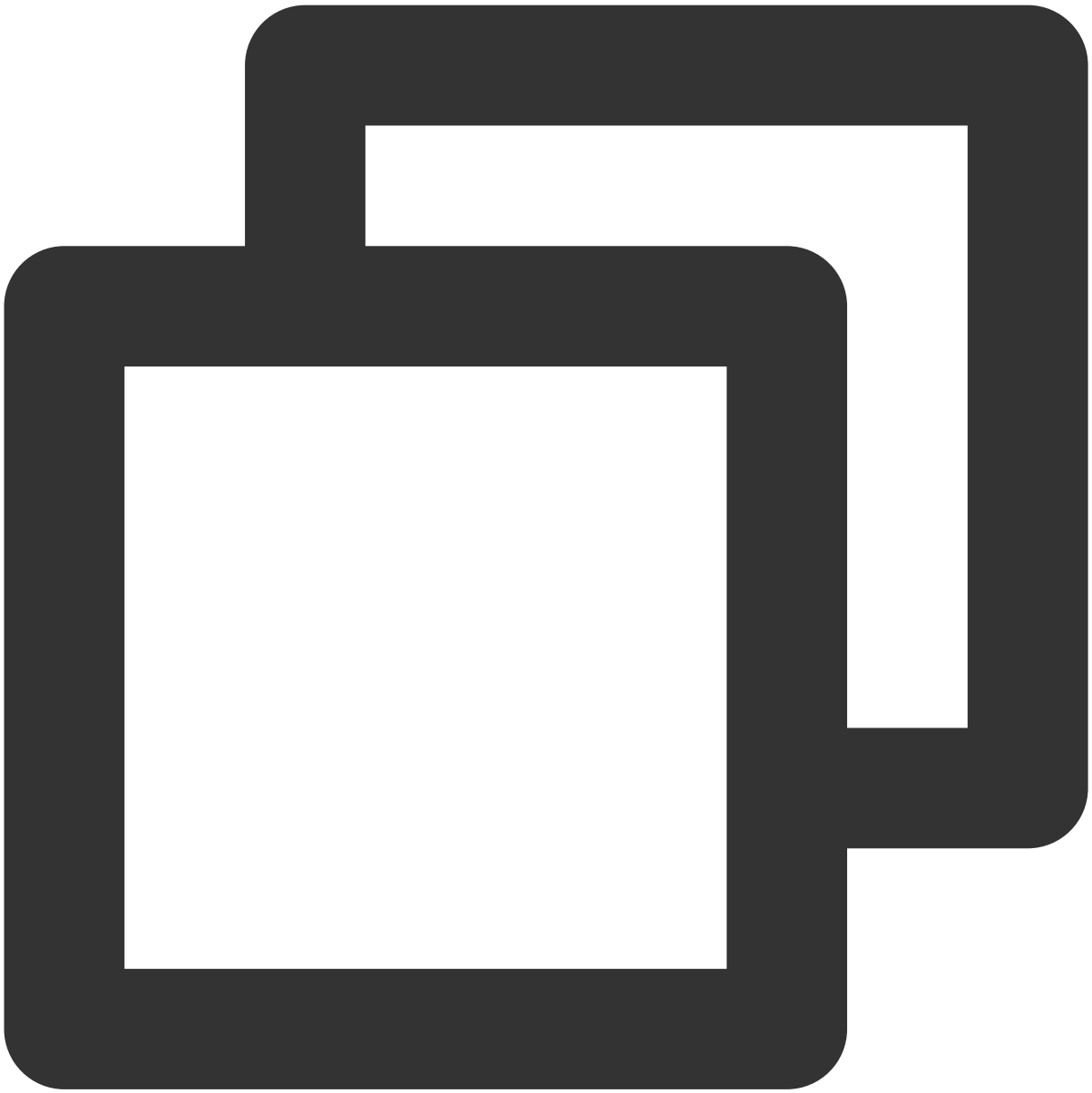
```
<import src="item.wxml"/>
<template is="item" data="{{text: 'forbar'}}"/>
```

import Scope

The concept of “scope” is used for import references. This means that the reference will import the template defined in the target file, but will not import any import template in the target file.

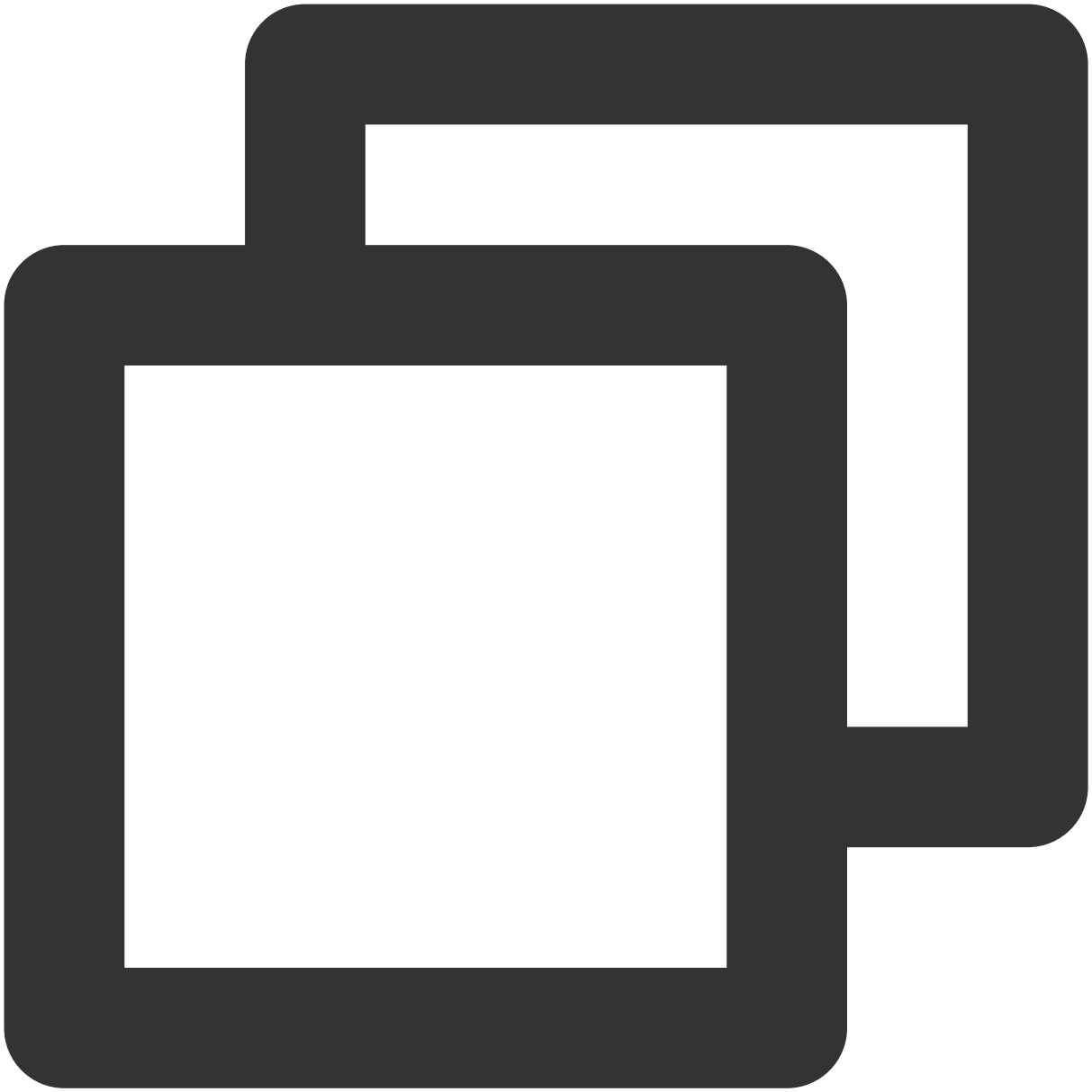
©2013-2022 Tencent Cloud. All rights reserved.

For example: In the case C import B, B import A, the template defined by B can be used in C and the template defined by A can be used in B. However, C cannot use the template defined by A.



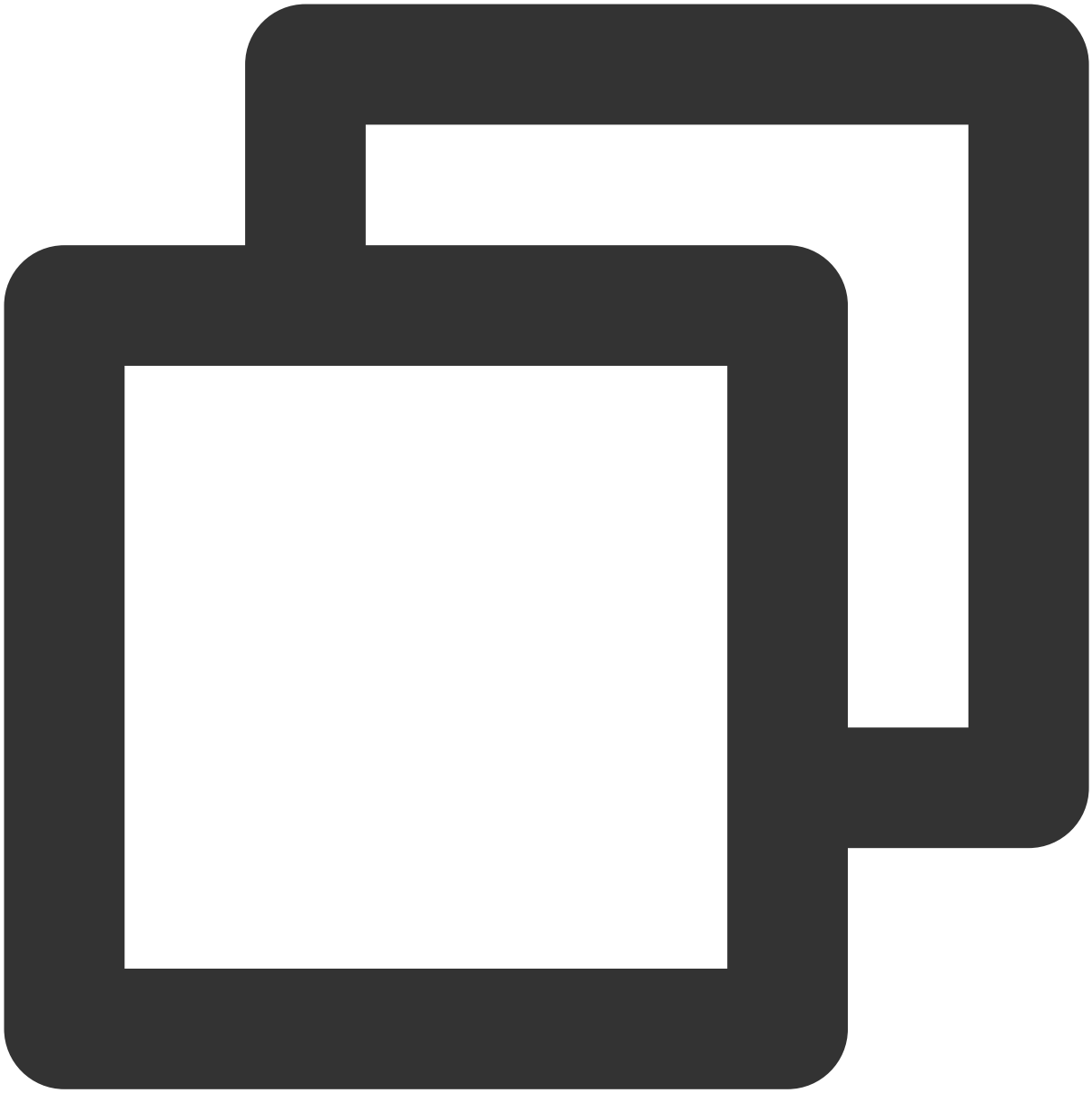
```
<!-- A.wxml -->
<template name="A">
  <text> A template </text>
</template>
```

©2013-2022 Tencent Cloud. All rights reserved.



```
<!-- B.wxml -->
<import src="a.wxml"/>
<template name="B">
  <text> B template </text>
</template>
```

©2013-2022 Tencent Cloud. All rights reserved.

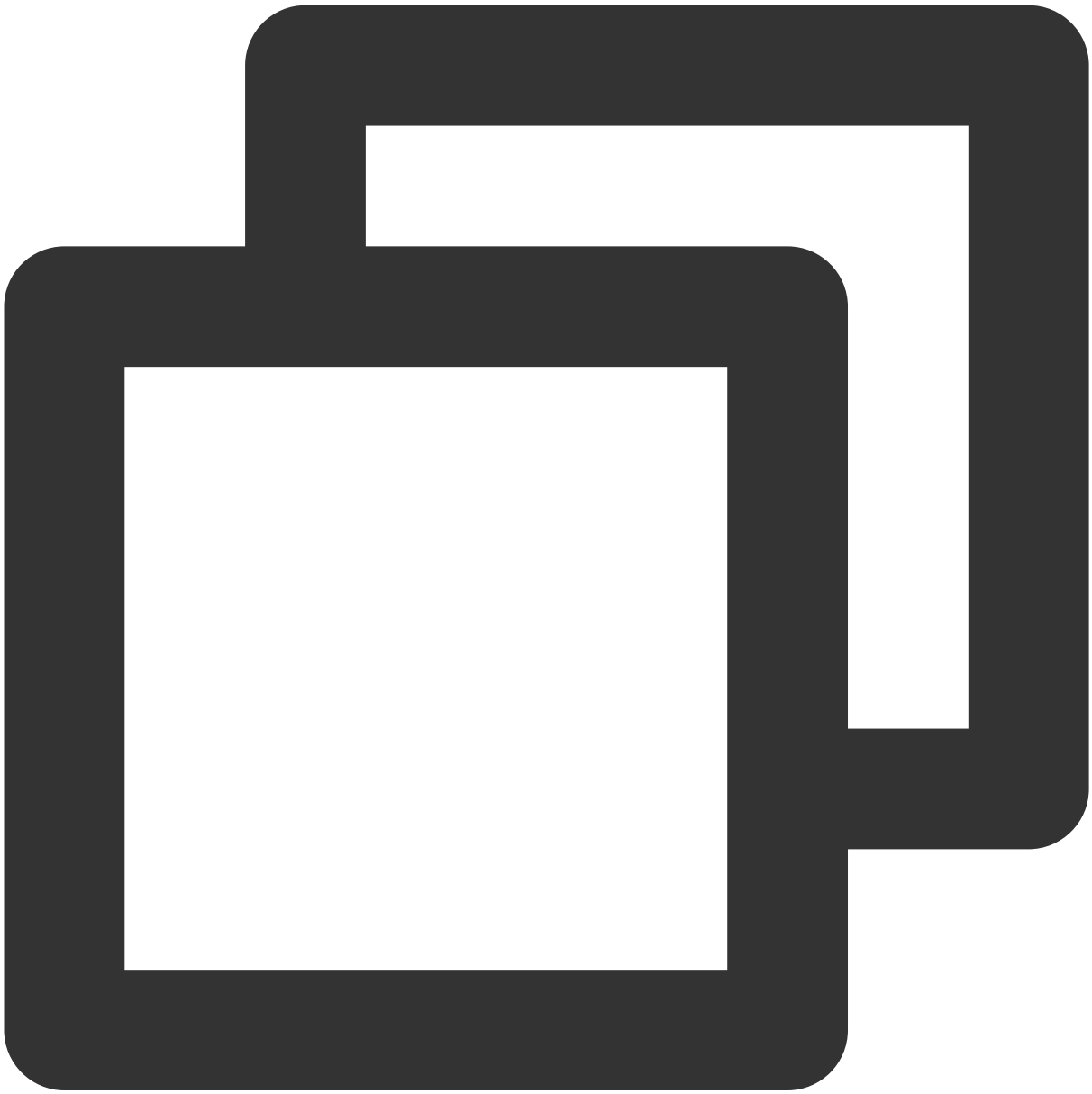


```
<!-- C.wxml -->
<import src="b.wxml"/>
<template is="A"/>  <!-- Error! Can not use tempalte when not import A. -->
<template is="B"/>
```

include

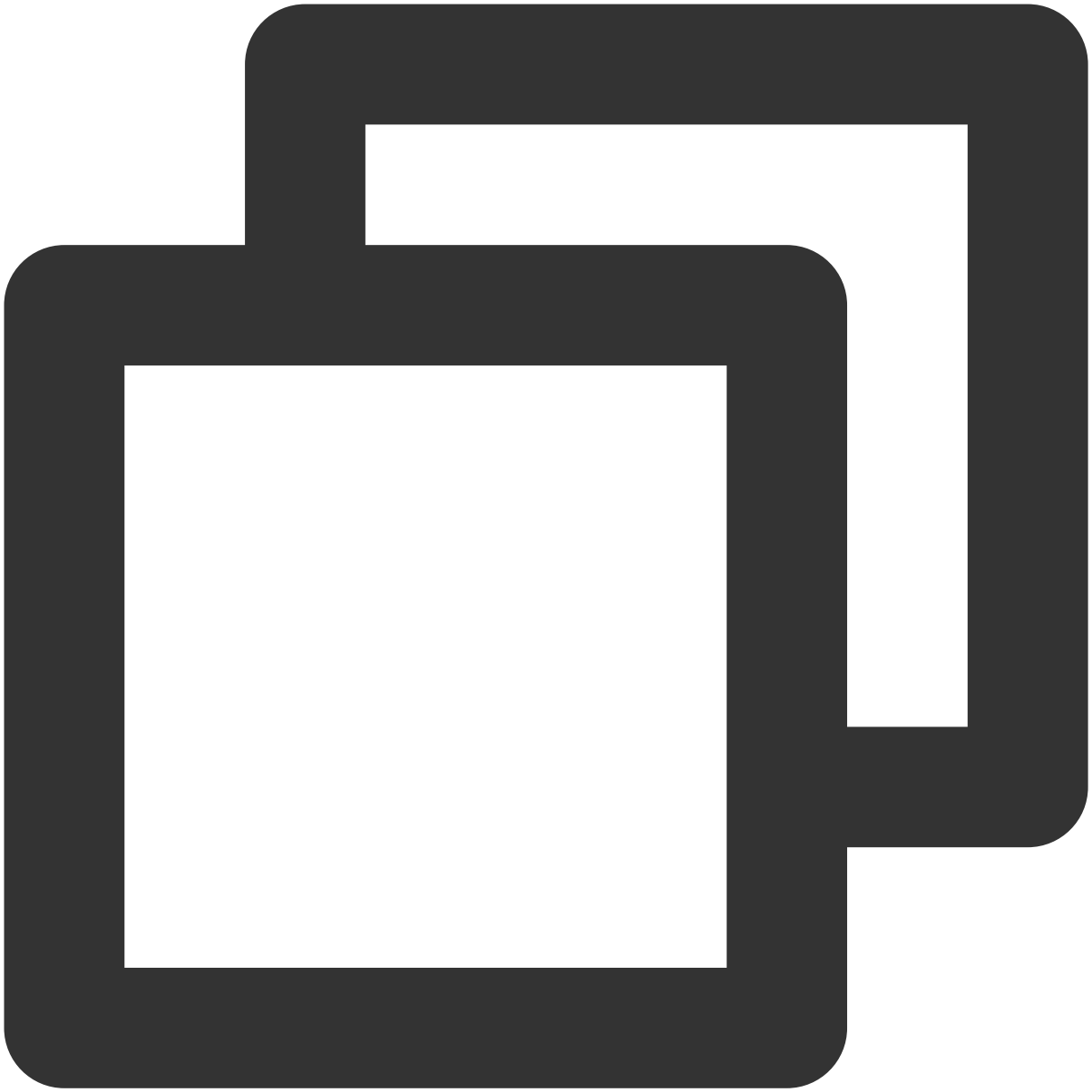
©2013-2022 Tencent Cloud. All rights reserved.

include can introduce the entire code excluding <template/> <wxs/>. This is equivalent to copying the code to the location of include. For example:



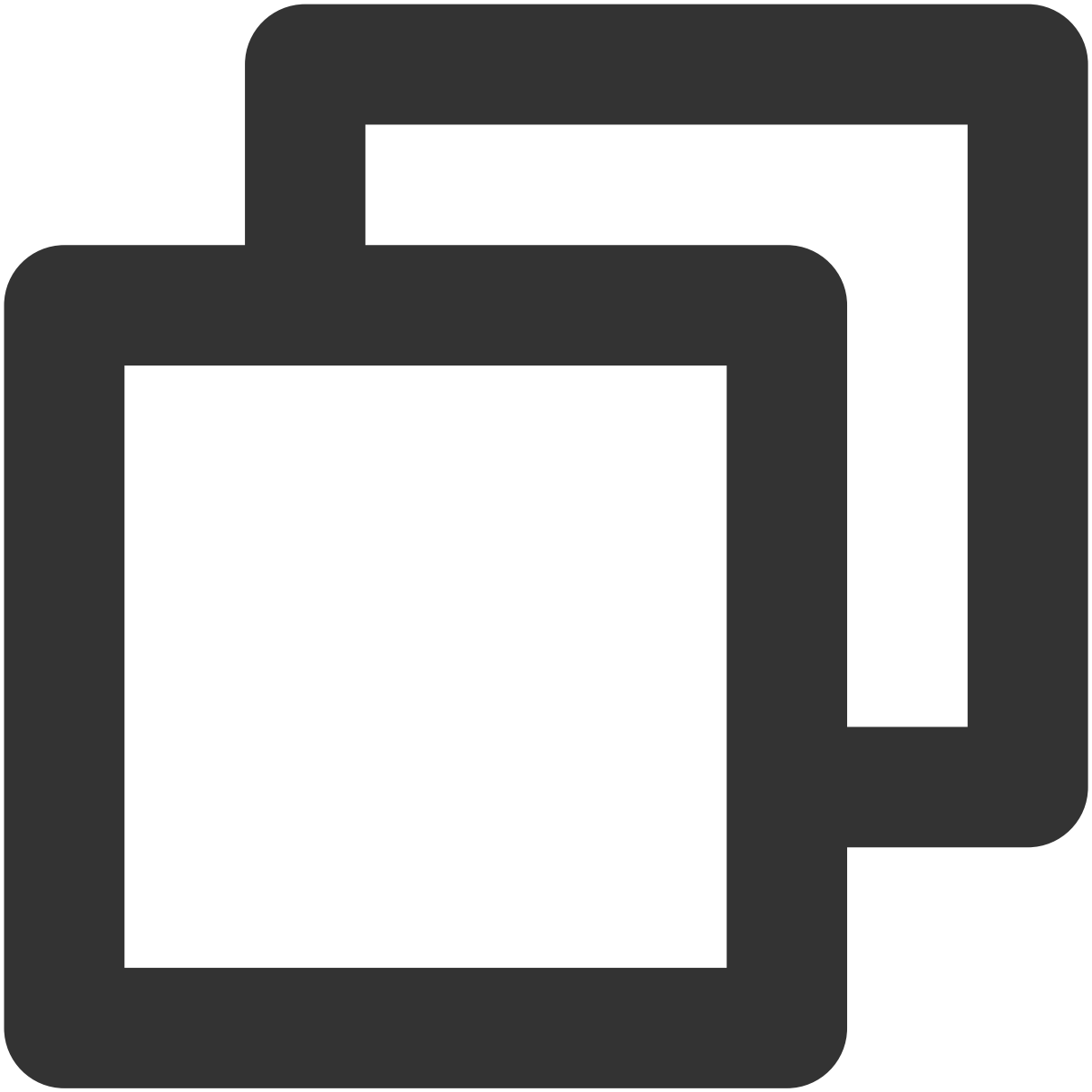
```
<!-- index.wxml -->
<include src="header.wxml"/>
<view> body </view>
<include src="footer.wxml"/>
```

©2013-2022 Tencent Cloud. All rights reserved.



```
<!-- header.wxml -->
<view> header </view>
```

©2013-2022 Tencent Cloud. All rights reserved.



```
<!-- footer.wxml -->
<view> footer </view>
```

©2013-2022 Tencent Cloud. All rights reserved.

WXS Syntax Reference

Introduce

Last updated：2024-07-12 16:58:59

WXS is a set of scripting languages for mini programs that, in combination with WXML, can build the structure of a page.

WXS is a different language from JavaScript and has its own syntax, which is not consistent with JavaScript.

WXS Modules
Variables
Annotations
Operators
Statements
Data Types
Base Library

©2013-2022 Tencent Cloud. All rights reserved.

WXS Modules

Last updated：2024-07-12 16:58:40

WXS code can be written in `<wxs>` tags in a wxml file or in files with the extension .wxs.

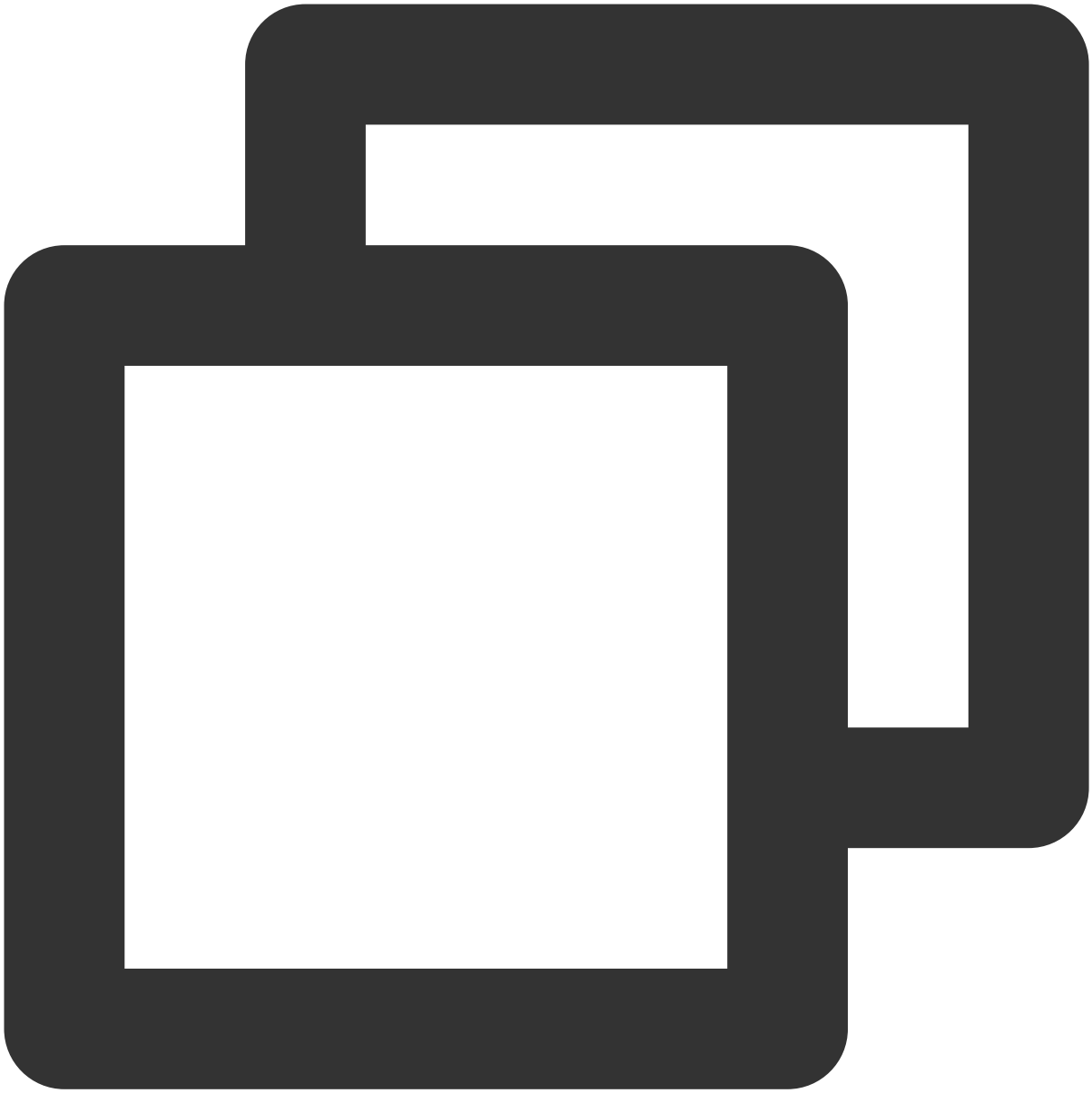
Modules

Each `.wxs` file or `<wxs>` tag is an independent module.
Each module has an independent scope. This means the variables and functions in a module are private by default and not visible to other modules.
A module can only externally expose its private variables and functions through the `module.exports` implementation.

wxs Files

In Weixin DevTools, right-click to create a `.wxs` file. You can write WXS scripts directly in the file.
Sample code:

©2013-2022 Tencent Cloud. All rights reserved.



```
// /pages/comm.wxs

var foo = "'hello world' from comm.wxs";
var bar = function(d) {
  return d;
}
module.exports = {
  foo: foo,
  bar: bar
};
```

©2013-2022 Tencent Cloud. All rights reserved.

In the above example, WXS code is written in the /**pages**/**comm.wxs** file. This **.wxs** file can be referenced by other **.wxs** files or **<wxs>** tags in WXML.

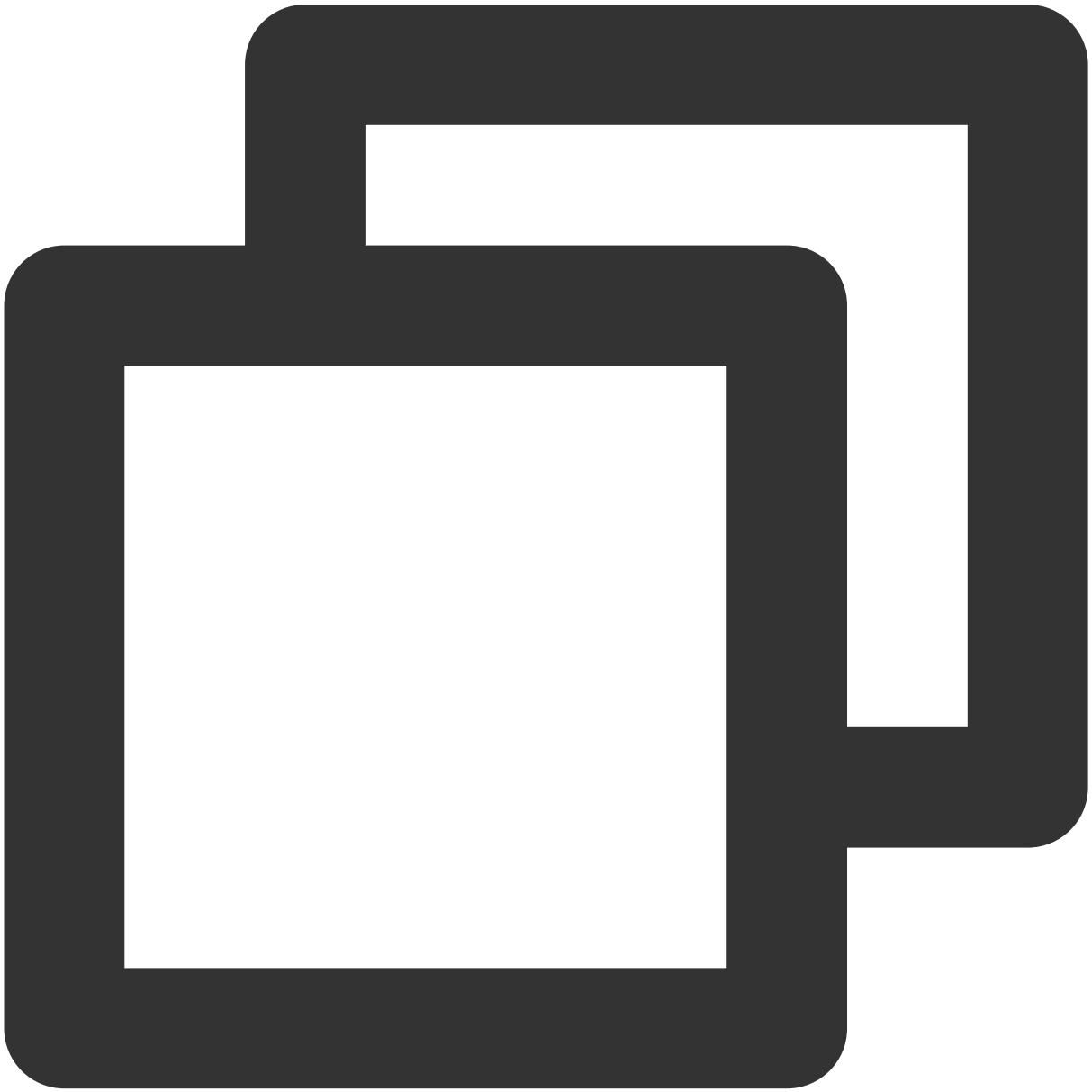
module Objects

Each `wxs` module contains one built-in `module` object.

Properties

exports: This property is used to share the module's private variables and functions externally.

Sample code:

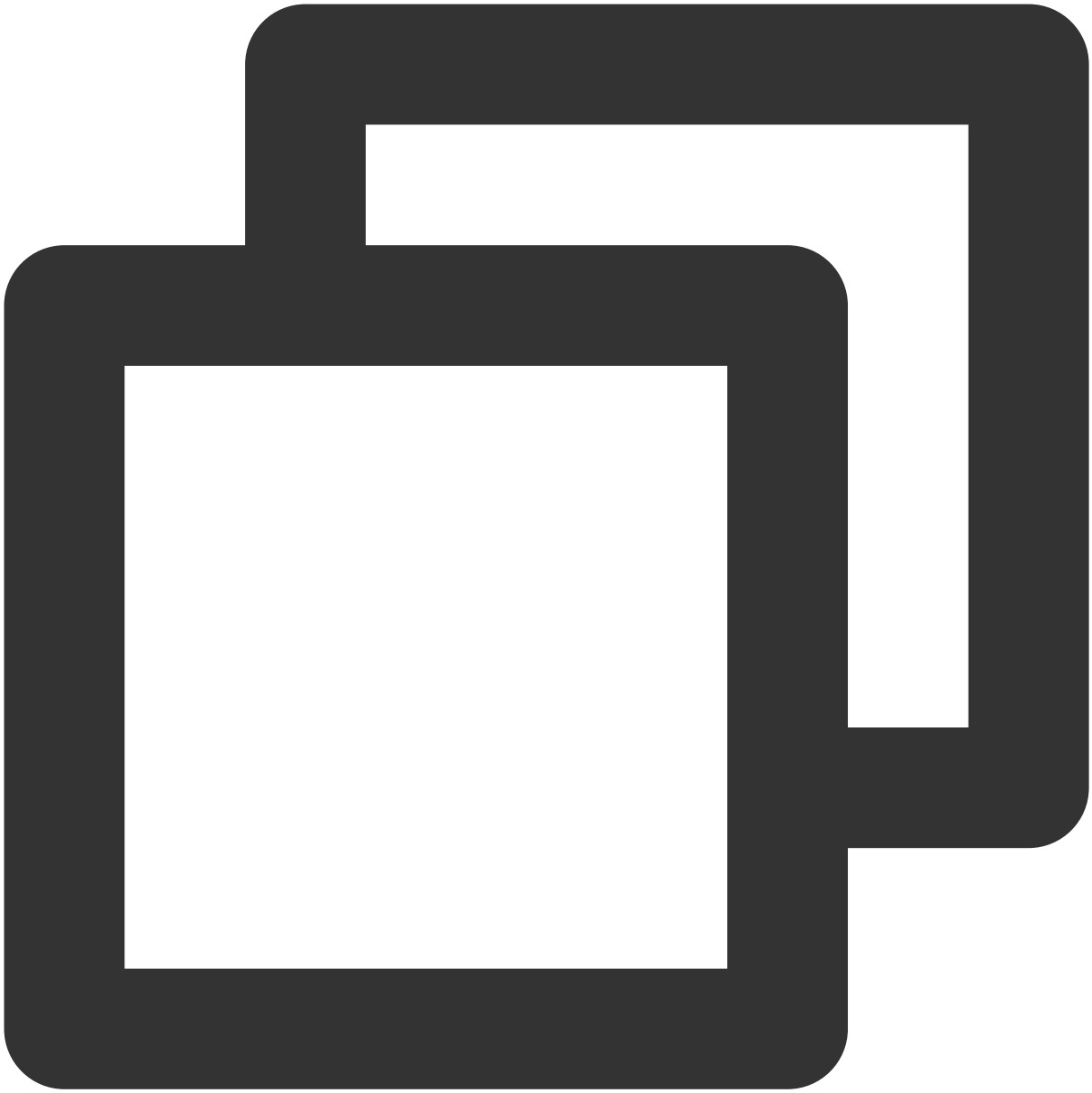


©2013-2022 Tencent Cloud. All rights reserved.

```
// /pages/tools.wxs

var foo = "'hello world' from tools.wxs";
var bar = function (d) {
  return d;
}
module.exports = {
  FOO: foo,
  bar: bar,
};
module.exports.msg = "some msg";
```

©2013-2022 Tencent Cloud. All rights reserved.

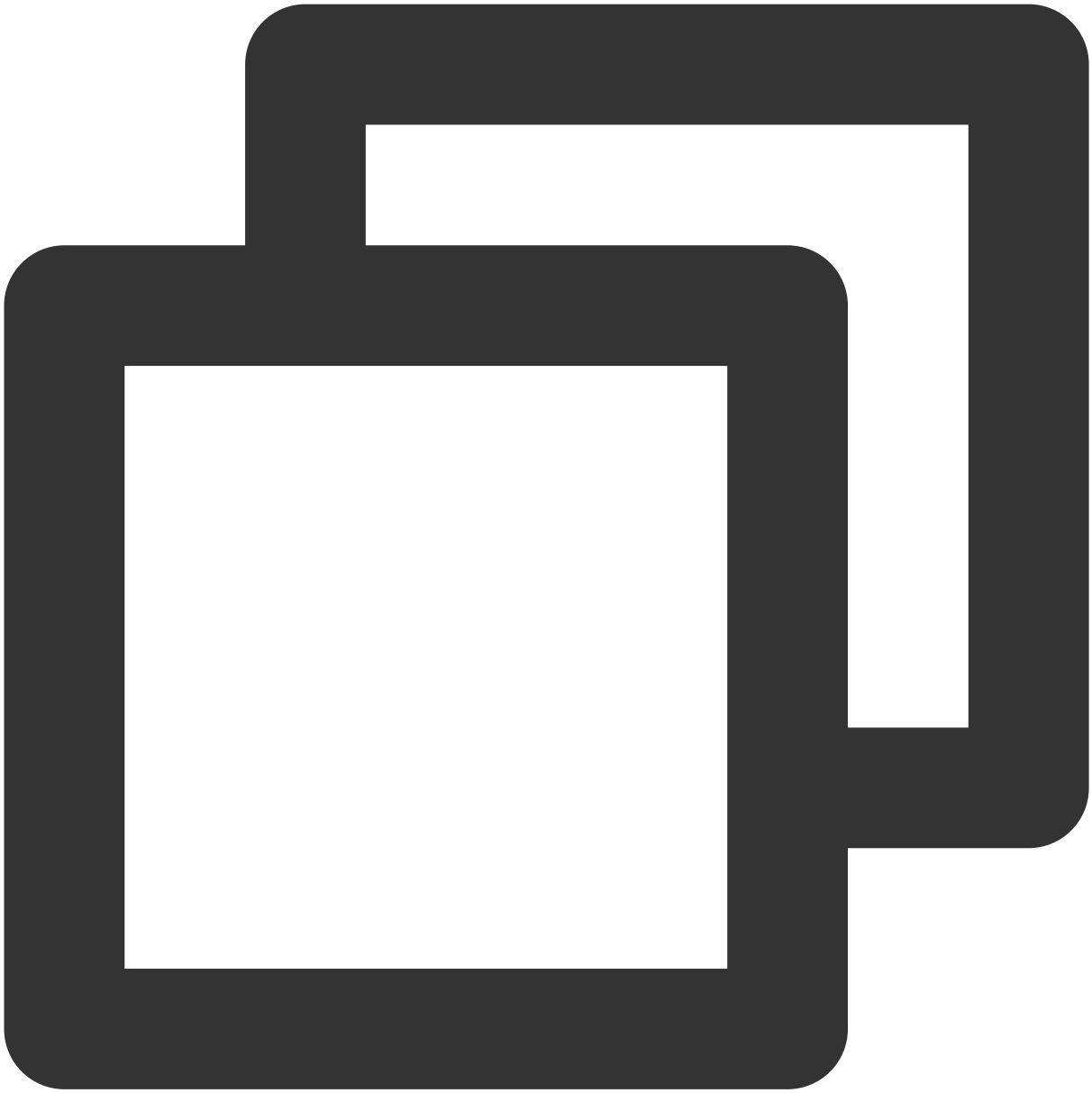


```
<!-- page/index/index.wxml -->

<wxs src="./../tools.wxs" module="tools" />
<view> {{tools.msg}} </view>
<view> {{tools.bar(tools.FOO)}} </view>
```

Page output:

©2013-2022 Tencent Cloud. All rights reserved.



```
some msg
'hello world' from tools.wxs
```

require Functions

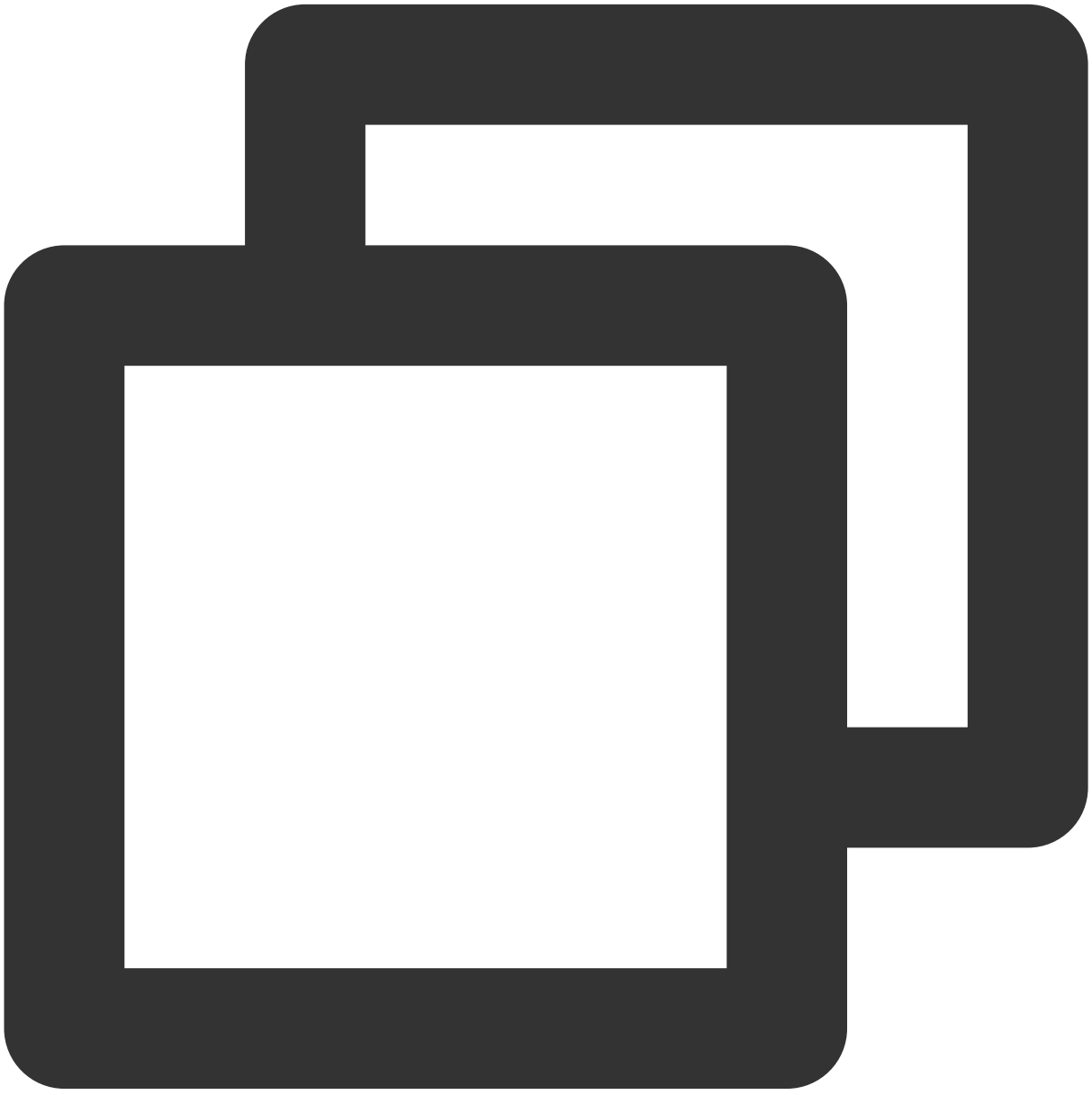
You can use a `require` function to reference another `wxs` file module in a `.wxs` module.

When referencing another `wxs` file module, you must note the following:

You can only reference `.wxs` files modules and must use relative paths;

©2013-2022 Tencent Cloud. All rights reserved.

All `wxs` modules are single instances. The first time a wxs module is referenced, it is automatically initialized as a single-instance object. Multiple pages, multiple regions, and multiple references all use the same wxs module object; If a wxs module is never referenced after it is defined, the module is not parsed or run.
Sample code:

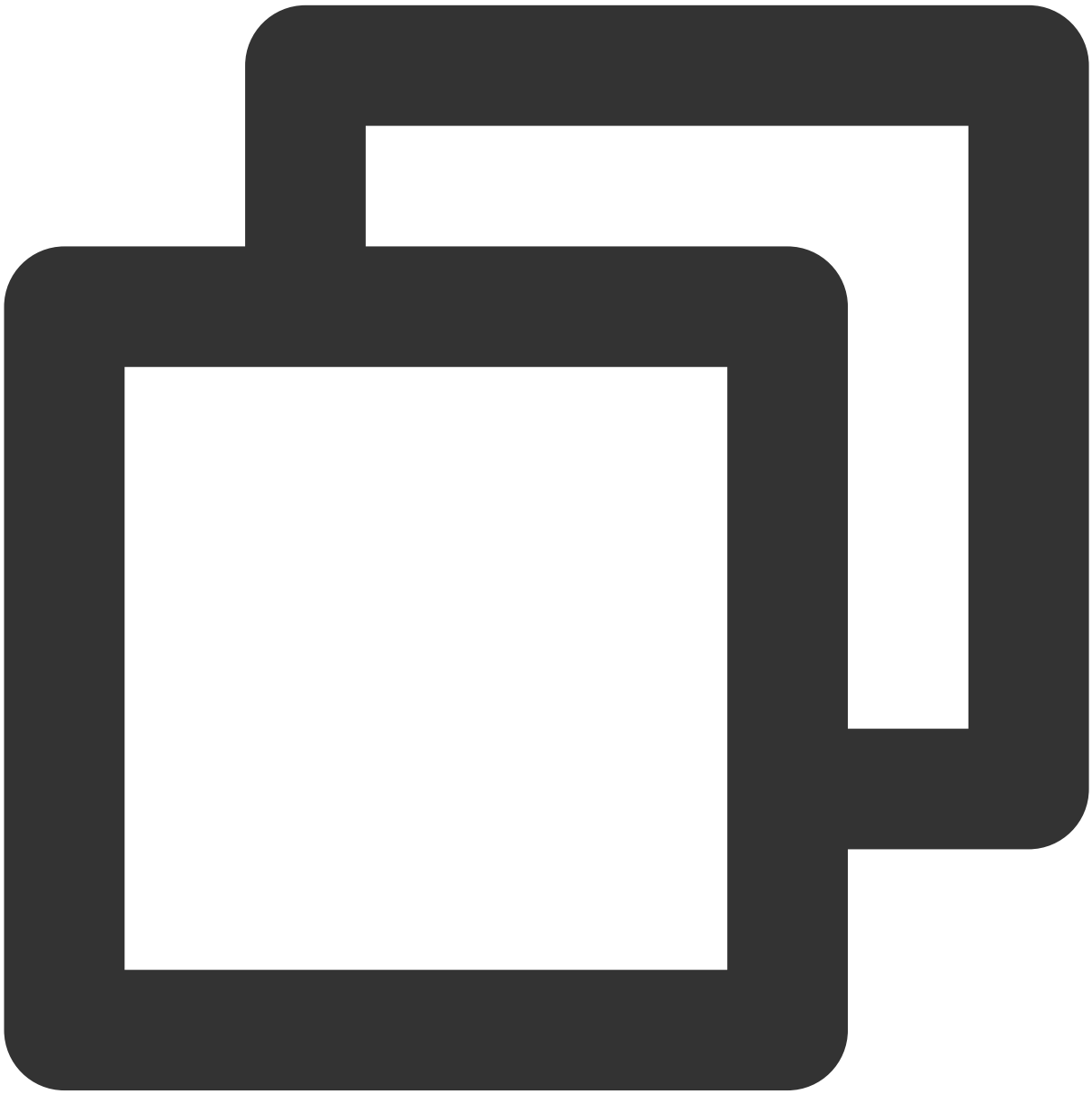


```
// /pages/tools.wxs

var foo = "'hello world' from tools.wxs";
var bar = function (d) {
  return d;
}
```

©2013-2022 Tencent Cloud. All rights reserved.

```
module.exports = {
  FOO: foo,
  bar: bar,
};
module.exports.msg = "some msg";
```



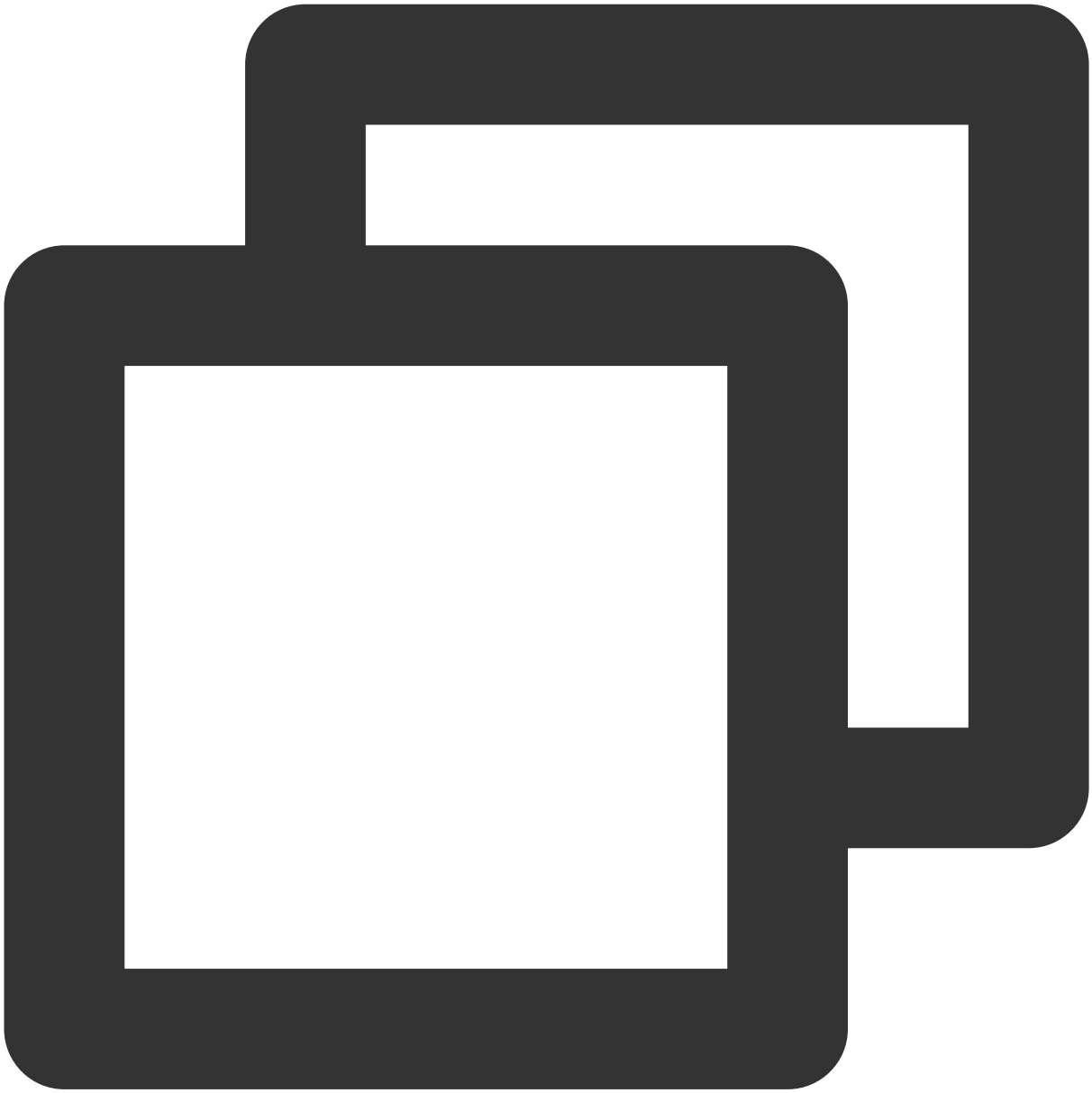
```
// /pages/logic.wxs

var tools = require("./tools.wxs");

console.log(tools.FOO);
```

©2013-2022 Tencent Cloud. All rights reserved.

```
console.log(tools.bar("logic.wxs"));
console.log(tools.msg);
```

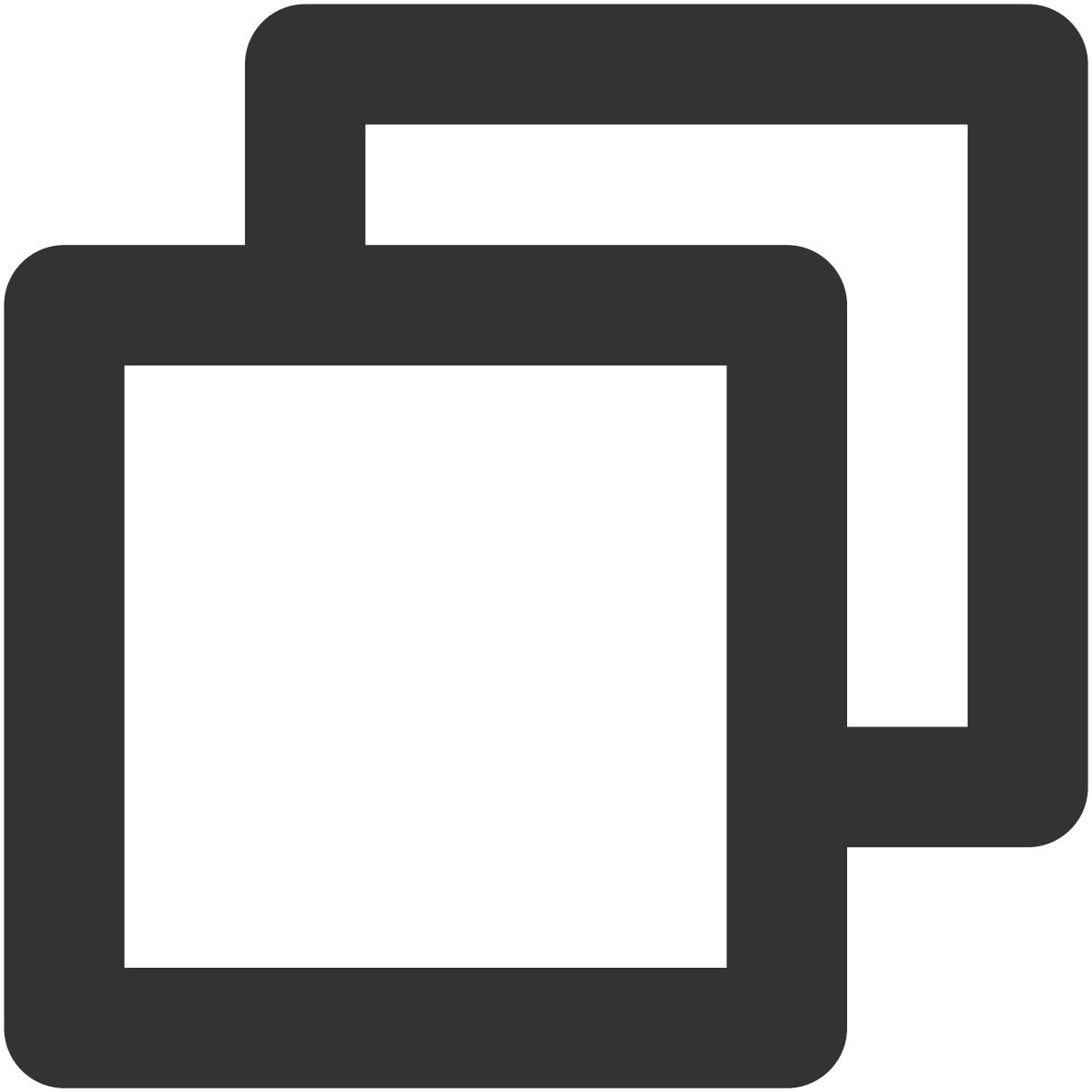


```
<!-- /page/index/index.wxml -->

<wxs src="./../logic.wxs" module="logic" />
```

Console output:

©2013-2022 Tencent Cloud. All rights reserved.



```
'hello world' from tools.wxs
logic.wxs
some msg
```

<wxs> Tags

Property Name	Type	Default Value	Description
module	String	-	The module name of the current <wxs> tag. This is a required

©2013-2022 Tencent Cloud. All rights reserved.

			field.
src	String	-	The relative path of the referenced .wxs file, only valid when this tag is a single closed tag or the tag content is empty.

module Property

The module property is the module name of the current `<wxs>` tag. In a single wxml file, we recommend that this value be unique. In the case of duplicate module names, the value is overwritten based on the order (the latter overwrites the former). The names of wxs modules are not overwritten across different files.

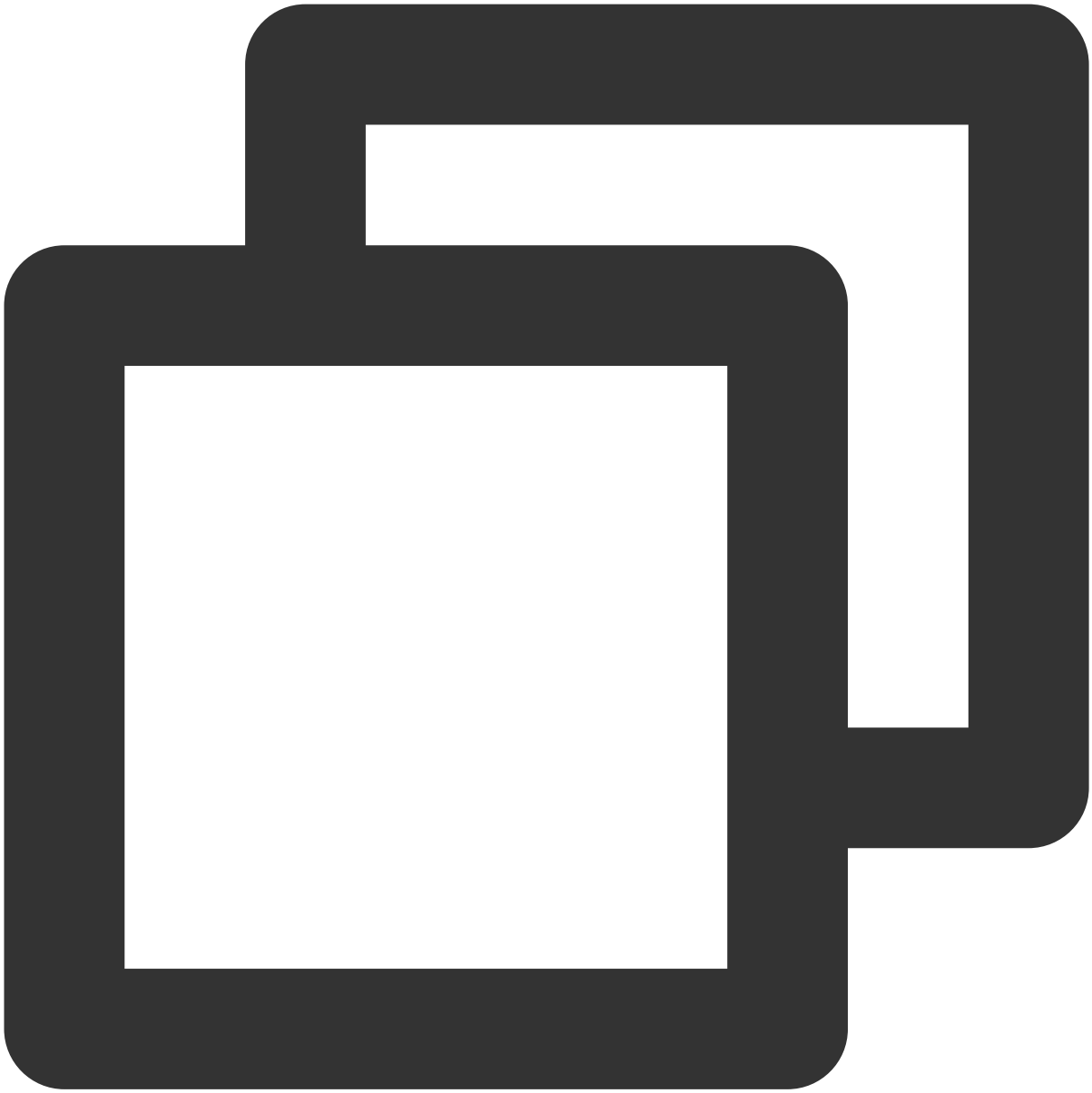
The name specified by the module property value must comply with the following rules:

The first character must be an English letter (a-z, A-Z) or underscore (_).

The remaining characters can be English letters (a-z, A-Z), underscores (_), or numbers (0-9).

Sample code:

©2013-2022 Tencent Cloud. All rights reserved.

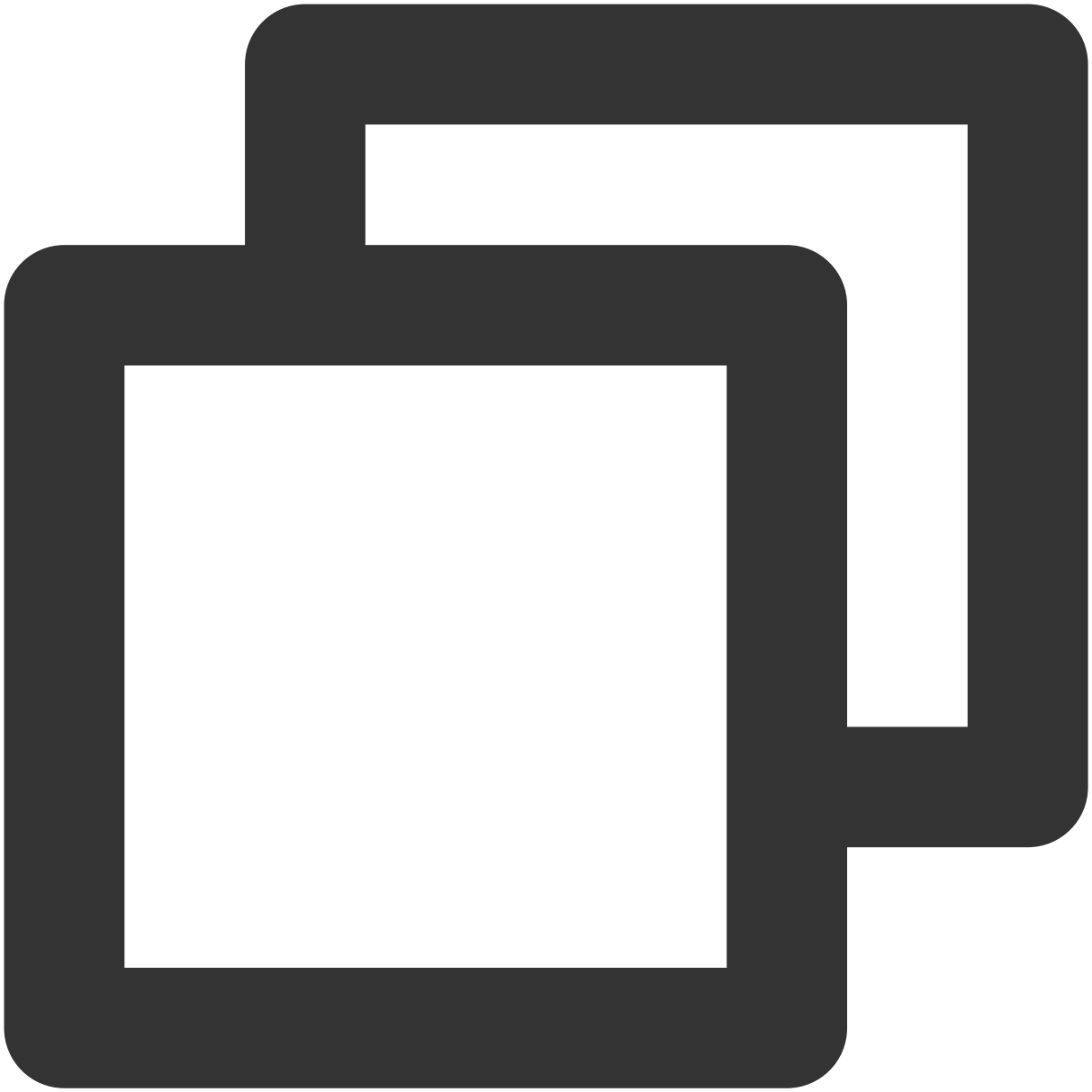


```
<!--wxml-->

<wxs module="foo">
var some_msg = "hello world";
module.exports = {
  msg : some_msg,
}
</wxs>
<view> {{foo.msg}} </view>
```

Page output:

©2013-2022 Tencent Cloud. All rights reserved.



```
hello world
```

The above example declares a module with the name foo that exposes the some_msg variable for use on the current page.

src Property

The src property is used to reference another wxs file module.

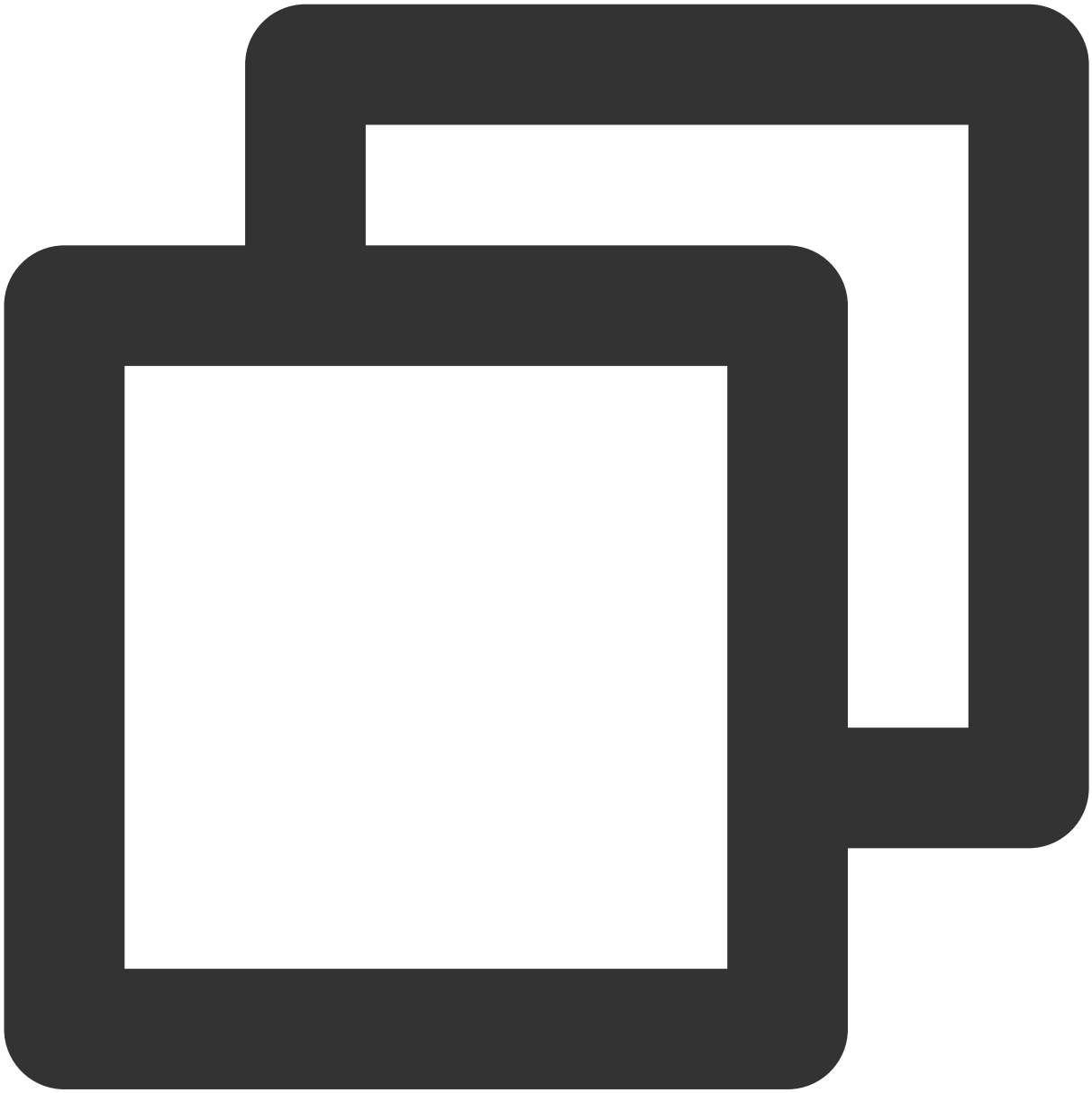
When referencing another wxs file module, you must note the following:

You can only reference .wxs files modules and must use relative paths.

©2013-2022 Tencent Cloud. All rights reserved.

All wxs modules are single instances. The first time a wxs module is referenced, it is automatically initialized as a single-instance object. Multiple pages, multiple regions, and multiple references all use the same wxs module object. If a wxs module is never referenced after it is defined, the module is not parsed or run.

Sample code:

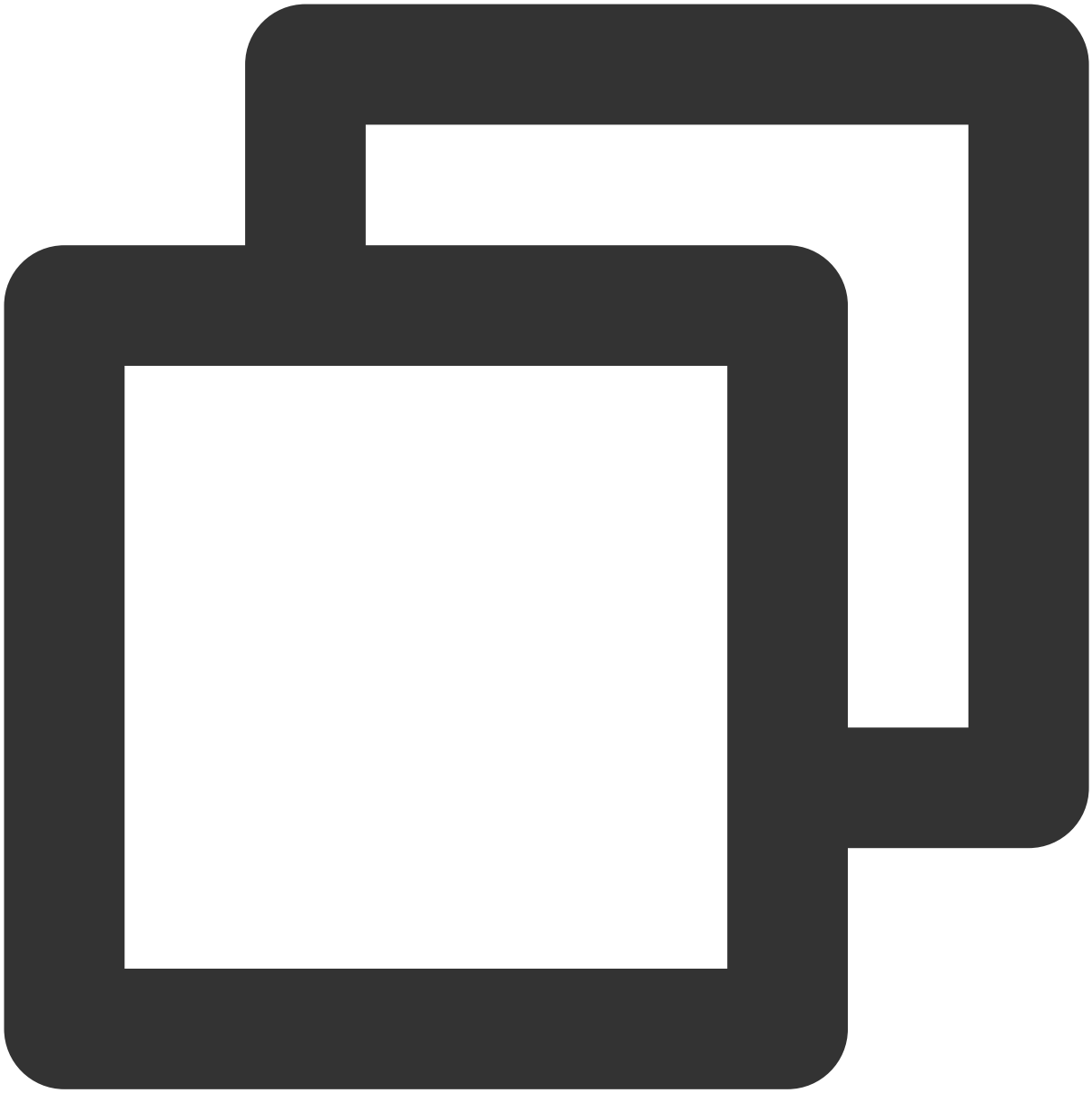


```
// /pages/index/index.js

Page({
  data: {
    msg: "'hello wrold' from js",
  }
```

©2013-2022 Tencent Cloud. All rights reserved.

```
})
```



```
<!-- /pages/index/index.wxml -->

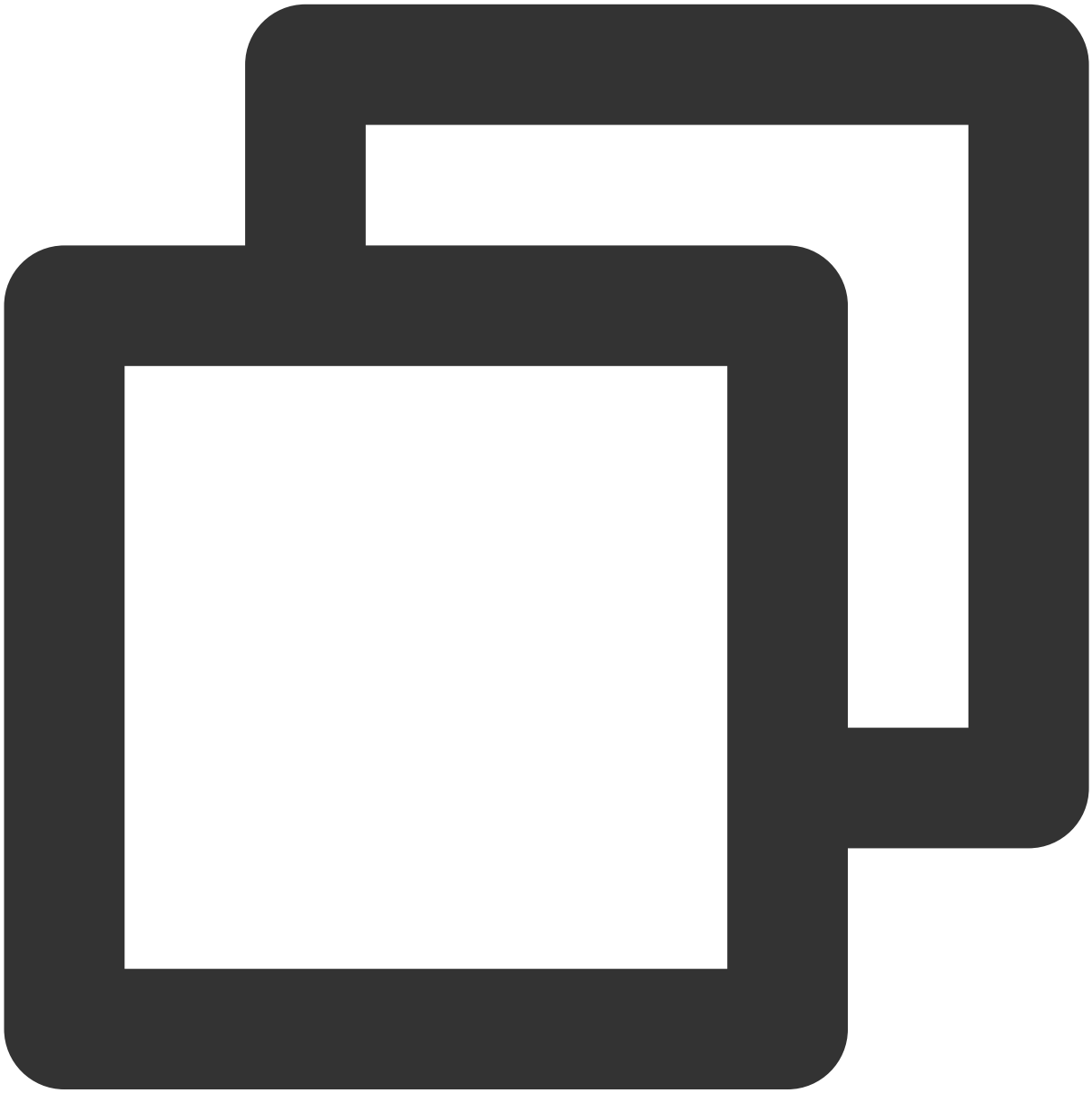
<wxs src="./../comm.wxs" module="some_comms"></wxs>
<!-- You can also directly use single tag closed syntax.
<wxs src="./../comm.wxs" module="some_comms" />
-->

<!-- Calls the bar function in the some_comms module, and the parameter is foo in t
<view> {{some_comms.bar(some_comms.foo)}} </view>
```

©2013-2022 Tencent Cloud. All rights reserved.

```
<!-- Calls the bar function in the some_comms module, and the parameter is msg in p
<view> {{some_comms.bar(msg)}} </view>
```

Page output:



```
'hello world' from comm.wxs
'hello wrold' from js
```

In the above example, the <wxs> tag is used to reference /page/comm.wxs in the file /page/index/index.wxml.

©2013-2022 Tencent Cloud. All rights reserved.

Note

A `<wxs>` module can only be accessed in the WXML file that defines the module. If you use `<include>` or `<import>`, `<wxs>` modules are not introduced into the corresponding WXML files.

In a `<template>` tag, you can only use the `<wxs>` module defined in the WXML file that defines the `<template>`.

©2013-2022 Tencent Cloud. All rights reserved.

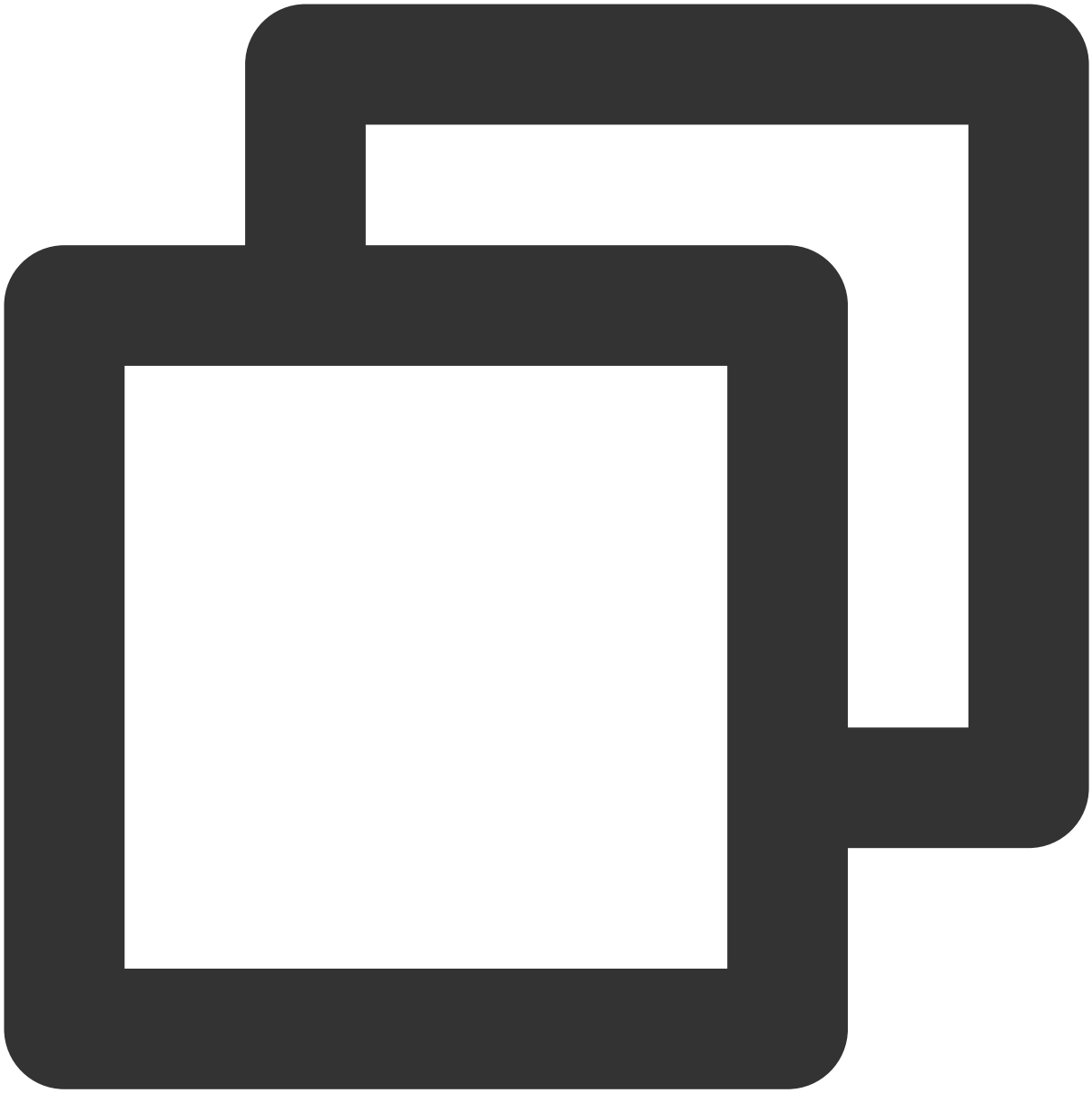
Variable

Last updated：2024-07-12 16:57:40

Concepts

Variables in WXS are references to values;
Variables that are not declared and used by direct assignment are defined as global variables;
If a variable is declared without being assigned a value, the default value is undefined;
var behaves in the same way as javascript, there will be variable lifting.

©2013-2022 Tencent Cloud. All rights reserved.



```
var foo = 1;
var bar = "hello world";
var i; // i === undefined
```

In the above code, three variables foo, bar and i are declared. Then, foo is assigned the value 1 and bar is assigned the string "hello world".

Variable Naming

©2013-2022 Tencent Cloud. All rights reserved.

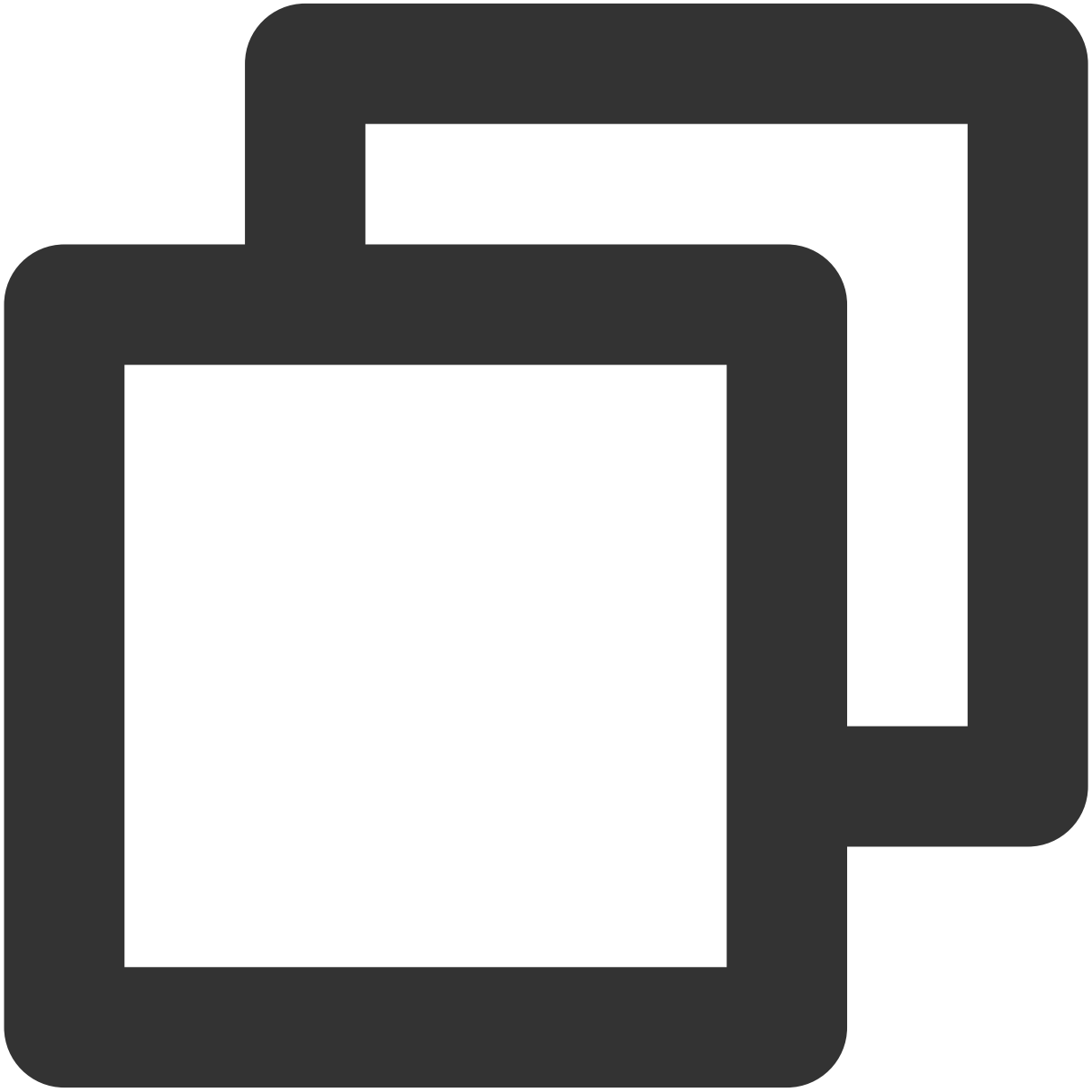
Variable naming must conform to the following two rules:

The first character must be: letter (a-zA-Z), underscore (_);

The remaining characters can be: letters (a-zA-Z), underscore (_), numbers (0-9).

Reserved Identifier

The following identifiers cannot be used as variable names:



```
delete
```

©2013-2022 Tencent Cloud. All rights reserved.

```
void
typeof

null
undefined
NaN
Infinity
var

if
else

true
false

require

this
function
arguments
return

for
while
do
break
continue
switch
case
default
```

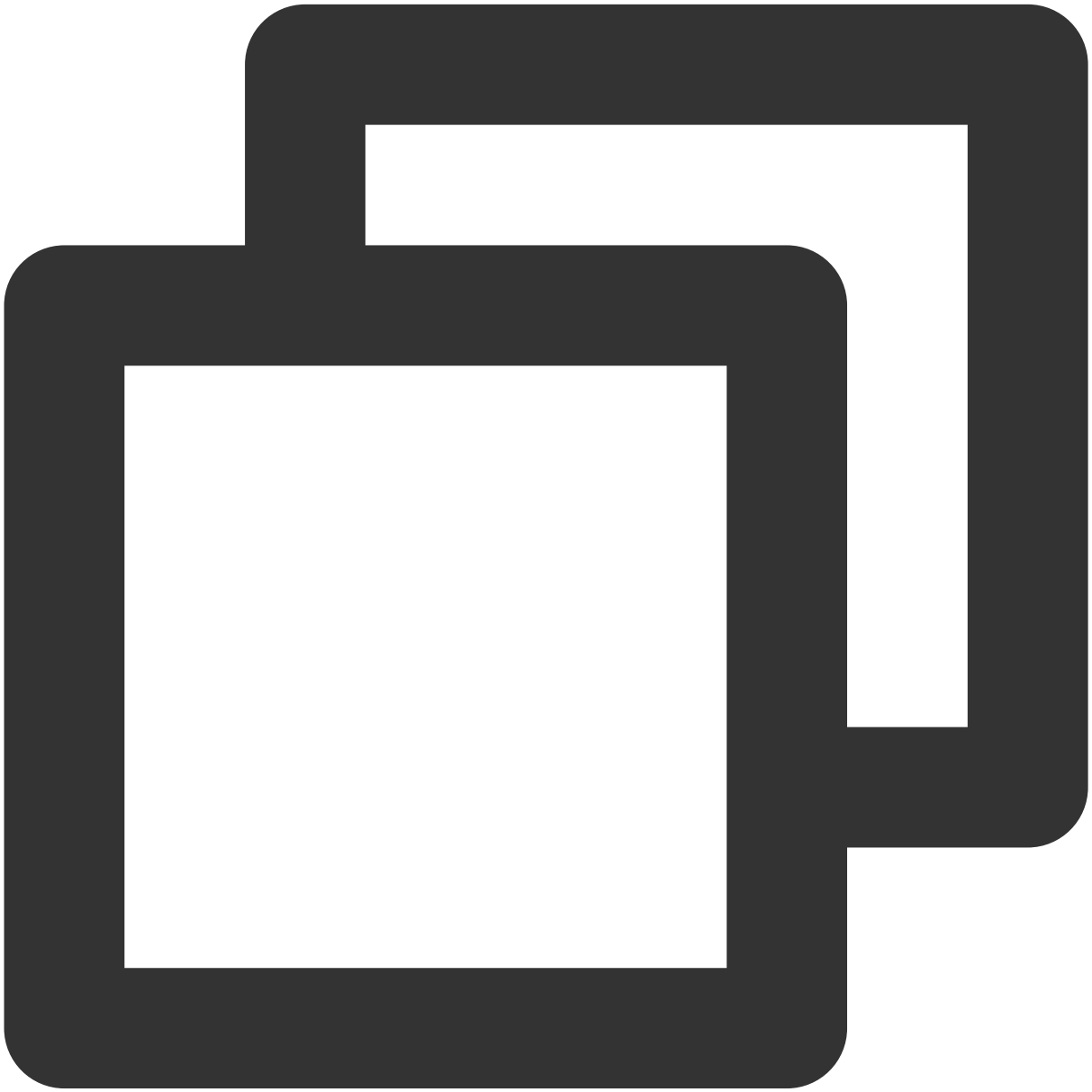
©2013-2022 Tencent Cloud. All rights reserved.

Annotations

Last updated：2024-07-12 16:57:17

WXS uses three main annotation methods.

Sample code:



```
<!-- wxml -->
<wxs module="sample">
// Method 1: Single-line annotations
```

©2013-2022 Tencent Cloud. All rights reserved.

```
/*
Method 2: Multi-line annotations
*/

/*
Method 3: End annotations. This annotates all the  WXS  code starting from  /* .

var a = 1;
var b = 2;
var c = "fake";

</wxs>
```

In the above example, all WXS code is annotated.

Note:

The only difference between method 3 and method 2 is the lack of the */ terminator.

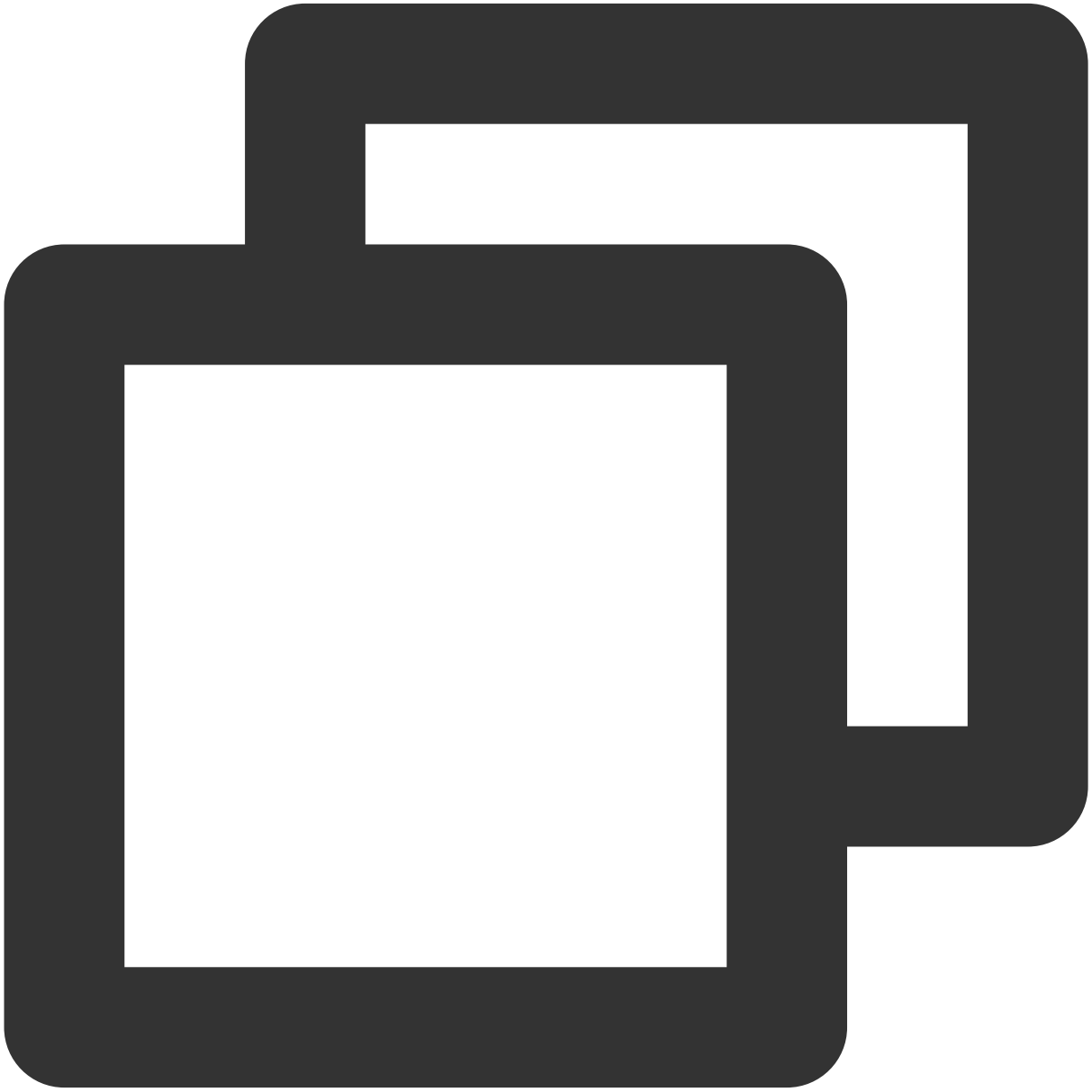
©2013-2022 Tencent Cloud. All rights reserved.

Operators

Last updated：2024-07-12 16:56:14

Basic Operators

Sample code:



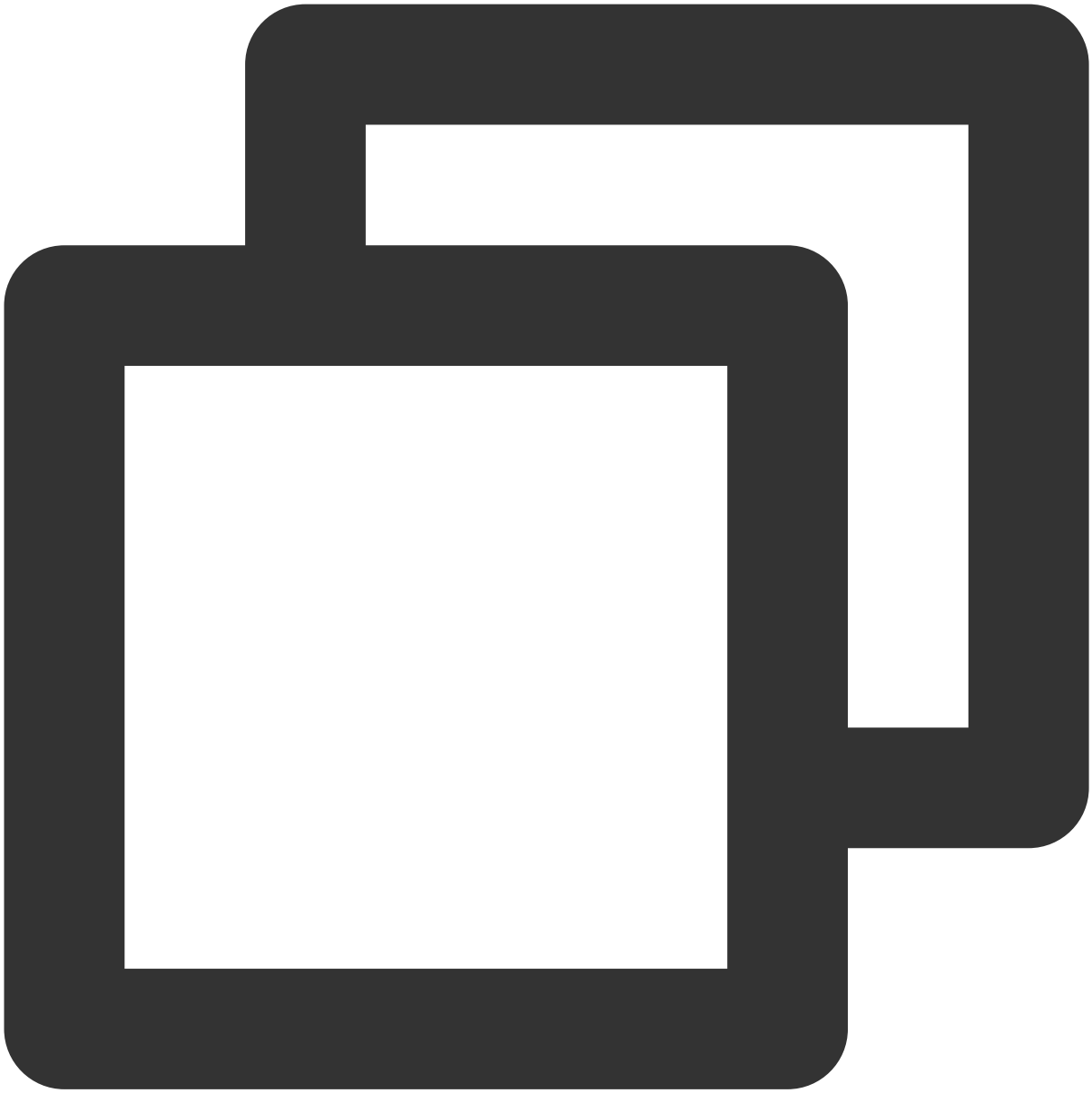
```
var a = 10, b = 20;
```

©2013-2022 Tencent Cloud. All rights reserved.

```
// Addition
console.log(30 === a + b);
// Subtraction
console.log(-10 === a - b);
// Multiplication
console.log(200 === a * b);
// Division
console.log(0.5 === a / b);
// Complementation
console.log(10 === a % b);
```

The addition operation (+) can also be used to concatenate strings.

©2013-2022 Tencent Cloud. All rights reserved.



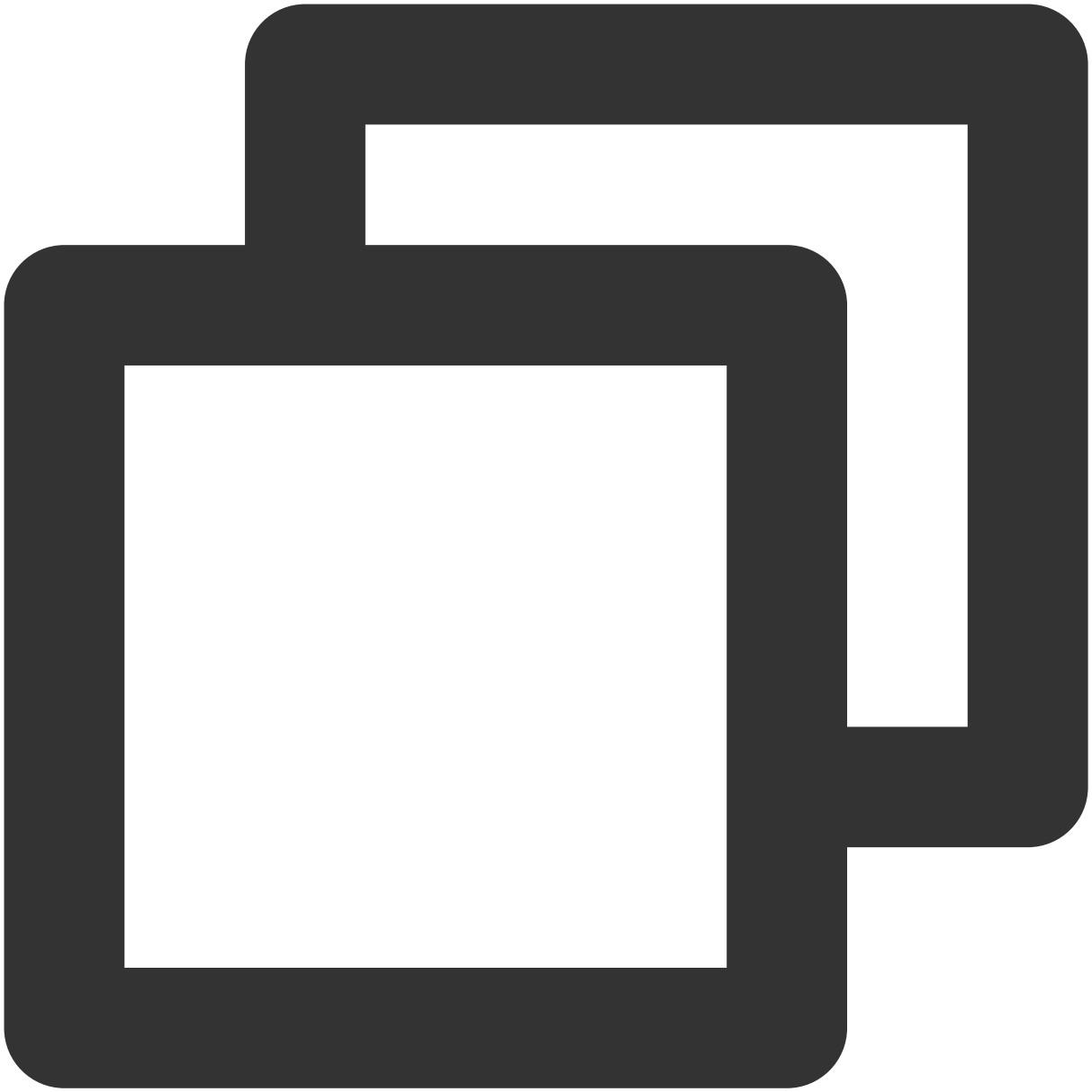
```
var a = '.w' , b = 'xs';

// String concatenation
console.log('.wxs' === a + b);
```

Unary Operators

Sample code:

©2013-2022 Tencent Cloud. All rights reserved.



```
var a = 10, b = 20;

// Auto increment
console.log(10 === a++);
console.log(12 === ++a);
// Auto decrement
console.log(12 === a--);
console.log(10 === --a);
// Positive value
console.log(10 === +a);
// Negative value
```

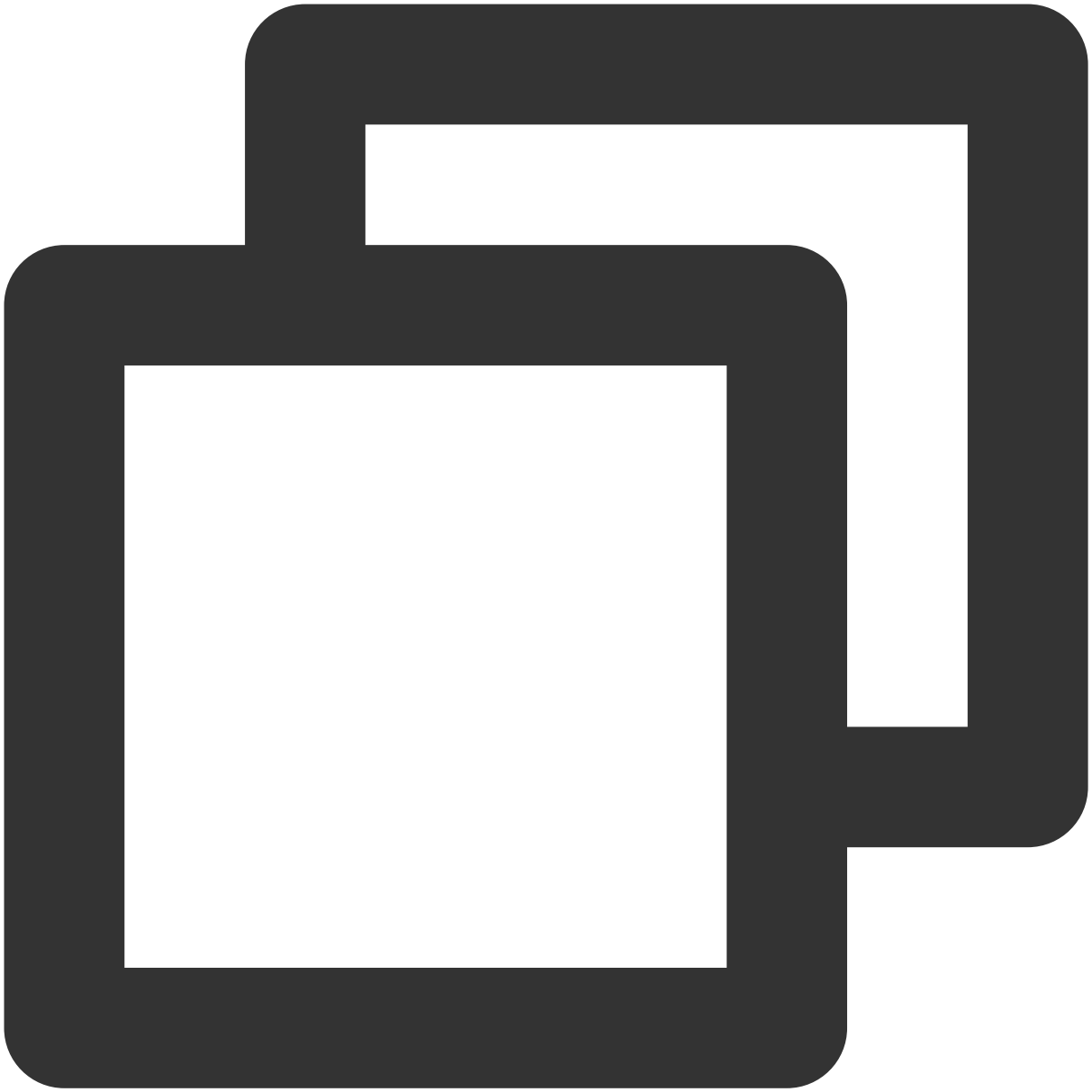
©2013-2022 Tencent Cloud. All rights reserved.

```
console.log(0-10 === -a);
// Not
console.log(-11 === ~a);
// Negate
console.log(false === !a);
// Delete
console.log(true === delete a.fake);
// Void
console.log(undefined === void a);
// Typeof
console.log("number" === typeof a);
```

Bit Operators

Sample code:

©2013-2022 Tencent Cloud. All rights reserved.



```
var a = 10, b = 20;

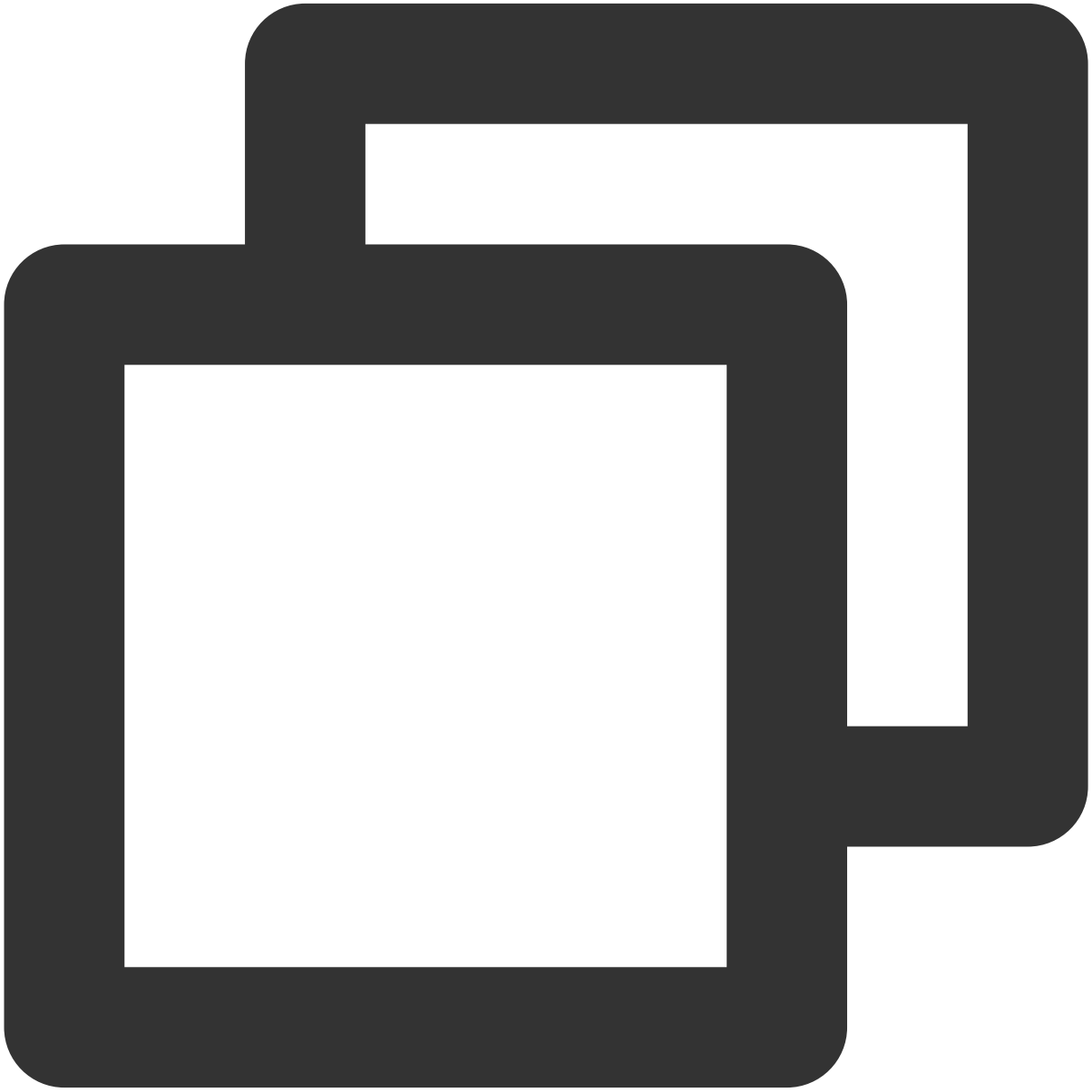
// Left shift
console.log(80 === (a << 3));
// Unsigned right shift
console.log(2 === (a >> 2));
// Signed right shift
console.log(2 === (a >>> 2));
// AND
console.log(2 === (a & 3));
// XOR
```

©2013-2022 Tencent Cloud. All rights reserved.

```
console.log(9 === (a ^ 3));
// OR
console.log(11 === (a | 3));
```

Relational Operators

Sample code:



```
var a = 10, b = 20;
```

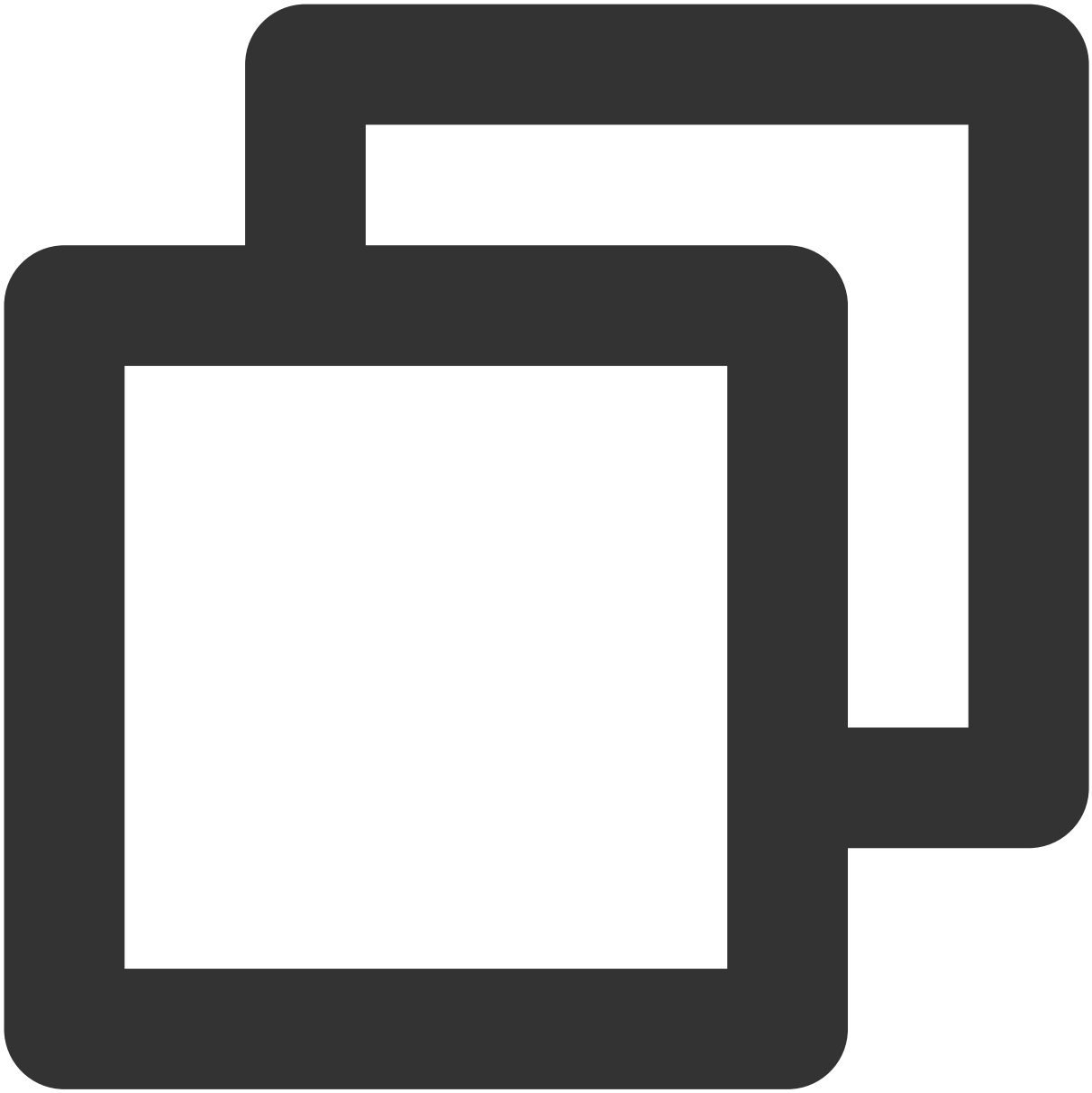
©2013-2022 Tencent Cloud. All rights reserved.

```
// Less than
console.log(true === (a < b));
// Greater than
console.log(false === (a > b));
// Less than or equal to
console.log(true === (a <= b));
// Greater than or equal to
console.log(false === (a >= b));
```

Equivalence Operators

Sample code:

©2013-2022 Tencent Cloud. All rights reserved.



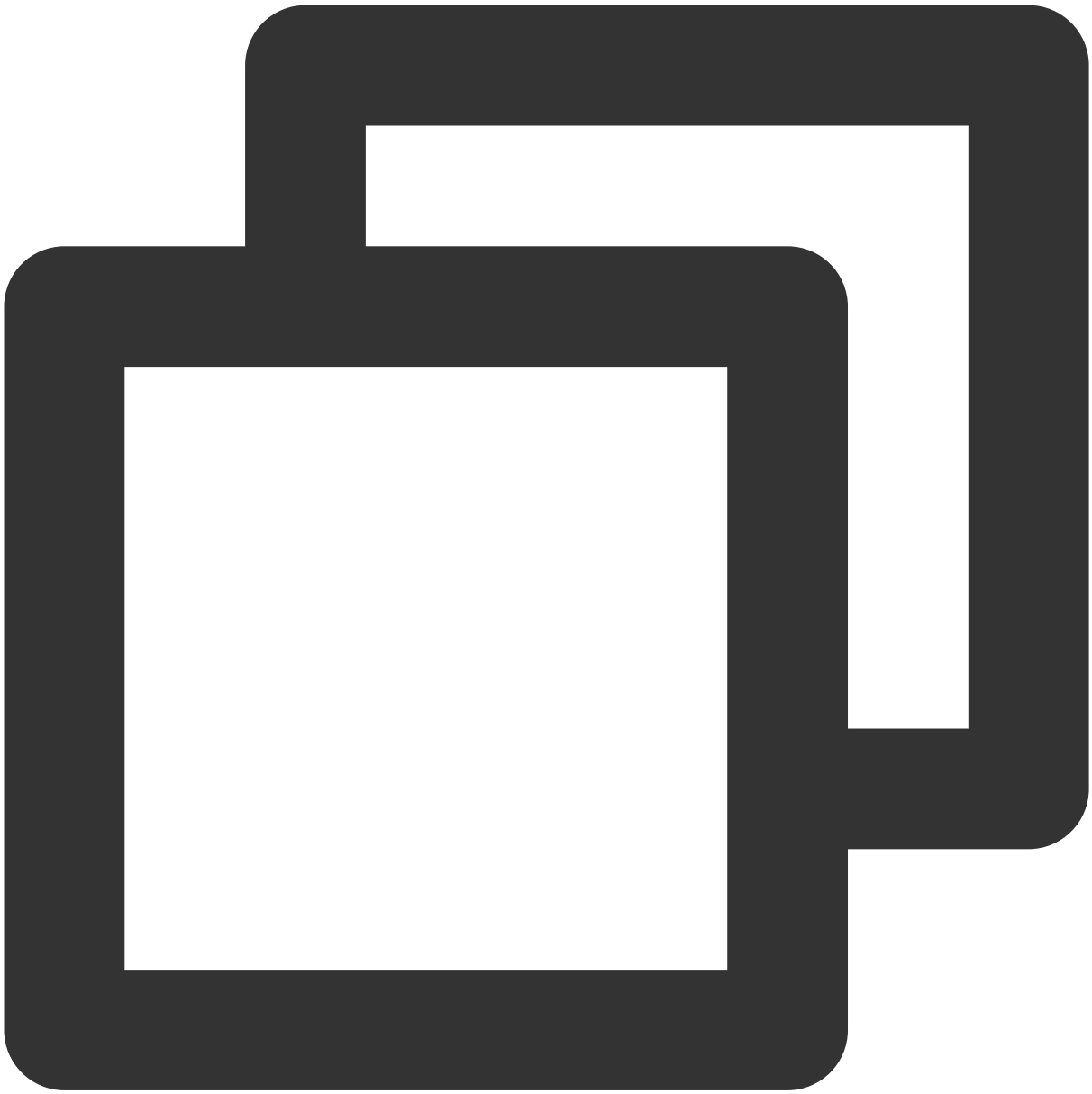
```
var a = 10, b = 20;

// Equal sign
console.log(false === (a == b));
// Not equal sign
console.log(true === (a != b));
// Equivalent sign
console.log(false === (a === b));
// Not equivalent sign
console.log(true === (a !== b));
```

©2013-2022 Tencent Cloud. All rights reserved.

Assignment Operators

Sample code:



```
var a = 10;

a = 10; a *= 10;
console.log(100 === a);
a = 10; a /= 5;
console.log(2 === a);
a = 10; a %= 7;
```

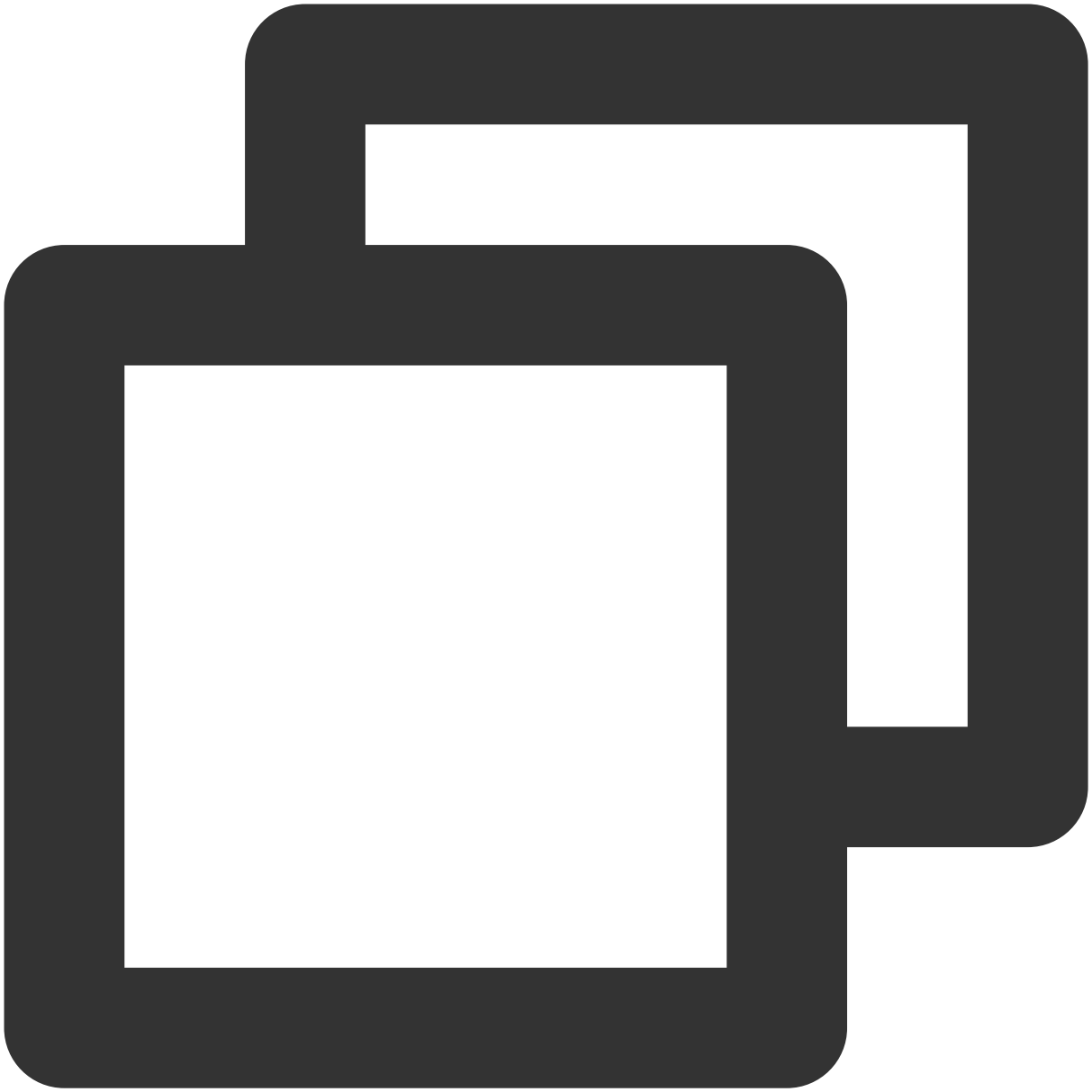
©2013-2022 Tencent Cloud. All rights reserved.

```
console.log(3 === a);
a = 10; a += 5;
console.log(15 === a);
a = 10; a -= 11;
console.log(-1 === a);
a = 10; a <<= 10;
console.log(10240 === a);
a = 10; a >>= 2;
console.log(2 === a);
a = 10; a >>>= 2;
console.log(2 === a);
a = 10; a &= 3;
console.log(2 === a);
a = 10; a ^= 3;
console.log(9 === a);
a = 10; a |= 3;
console.log(11 === a);
```

Binary Logical Operators

Sample code:

©2013-2022 Tencent Cloud. All rights reserved.



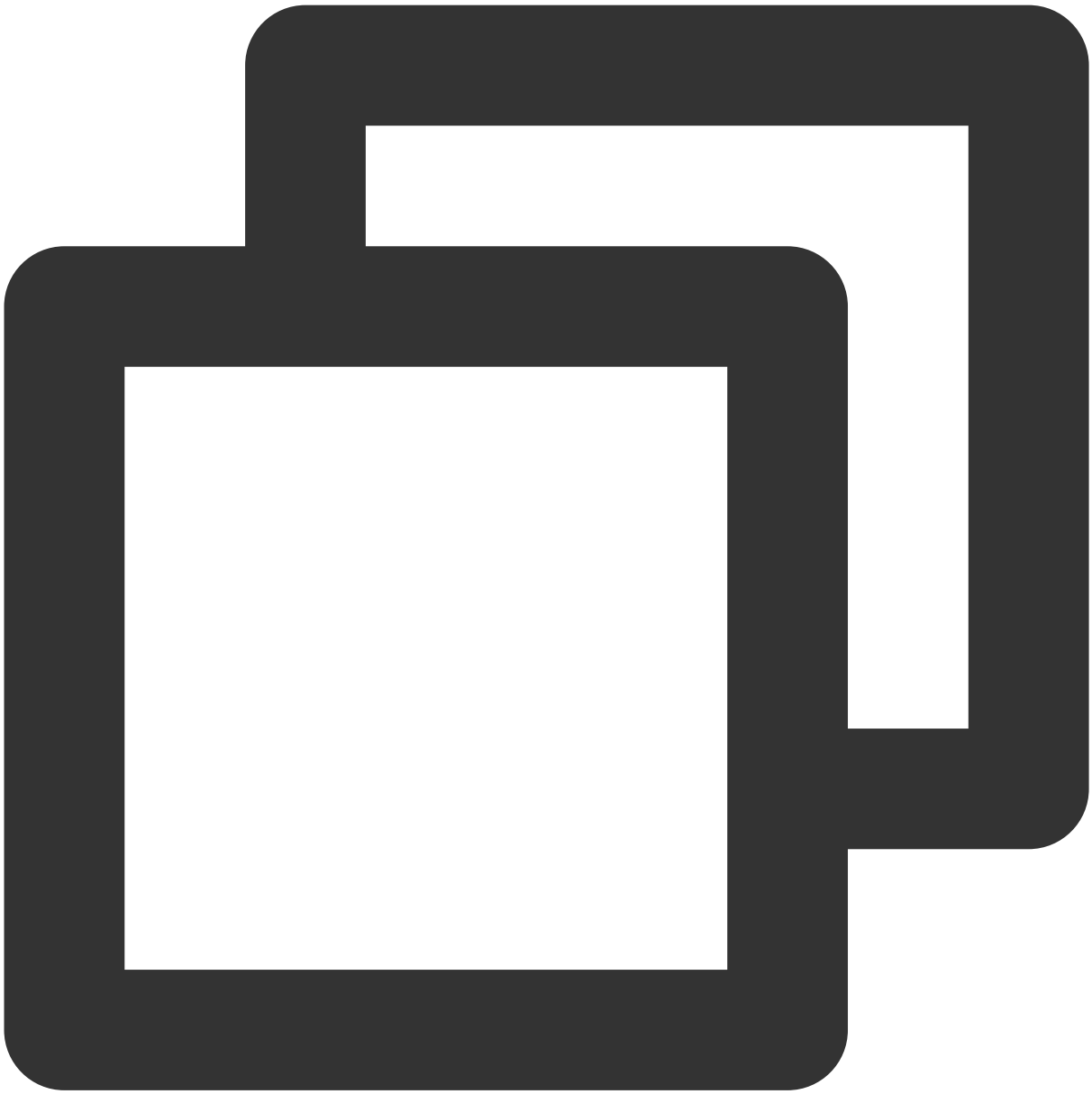
```
var a = 10, b = 20;

// Logical AND
console.log(20 === (a && b));
// Logical OR
console.log(10 === (a || b));
```

Other Operators

©2013-2022 Tencent Cloud. All rights reserved.

Sample code:



```
var a = 10, b = 20;

//Conditional operator
console.log(20 === (a >= 10 ? a + 10 : b + 10));
//Comma operator
console.log(20 === (a, b));
```

©2013-2022 Tencent Cloud. All rights reserved.

Operator Priority

Priority	Operator	Description	Associativity
20	(...)	Brackets	n/a
19	... .	Member access	Left to right
	... [...]	Member access	Left to right
	... (...)	Function call	Left to right
17	... ++	Postfix increment	n/a
	... --	Postfix decrement	n/a
16	! ...	Logical not	Right to left
	~ ...	Bitwise not	Right to left
	+ ...	Unary addition	Right to left
	- ...	Unary subtraction	Right to left
	++ ...	Prefix increment	Right to left
	-- ...	Prefix decrement	Right to left
	typeof ...	typeof	Right to left
	void ...	void	Right to left
	delete ...	delete	Right to left
14	... * ...	Multiplication	Left to right
	... / ...	Division	Left to right
	... % ...	Modulo	Left to right
13	... + ...	Addition	Left to right
	... - ...	Subtraction	left to right
12	... << ...	Bitwise left shift	Left to right
	... >> ...	Bitwise right shift	Left to right
	... >>> ...	Unsigned right shift	Left to right

©2013-2022 Tencent Cloud. All rights reserved.

11	... < ...	Less than	Left to right
	... <= ...	Less than or equal to	Left to right
	... > ...	Greater than	Left to right
	... >= ...	Greater than or equal to	Left to right
10	... == ...	Equal sign	Left to right
	... != ...	Not equal sign	Left to right
	... === ...	Equivalent sign	Left to right
	... !== ...	Not equivalent sign	Left to right
9	... & ...	Bitwise AND	Left to right
8	... ^ ...	Bitwise XOR	Left to right
7	... | ...	Bitwise OR	Left to right
6	... && ...	Logical AND	Left to right
5	... || ...	Local OR	Left to right
4	... ? ... : ...	Conditional operator	Left to right
3	... = ...	Value assignment	Left to right
	... += ...	Value assignment	Left to right
	... -= ...	Value assignment	Left to right
	... *= ...	Value assignment	Left to right
	... /= ...	Value assignment	Left to right
	... %= ...	Value assignment	Left to right
	... <<= ...	Value assignment	Left to right
	... >>= ...	Value assignment	Left to right
	... >>>= ...	Value assignment	Left to right
	... &= ...	Value assignment	Left to right
	... ^= ...	Value assignment	Left to right

©2013-2022 Tencent Cloud. All rights reserved.

	... | = ...	Value assignment	Right to left
0	... , ...	Comma	Left to right

©2013-2022 Tencent Cloud. All rights reserved.

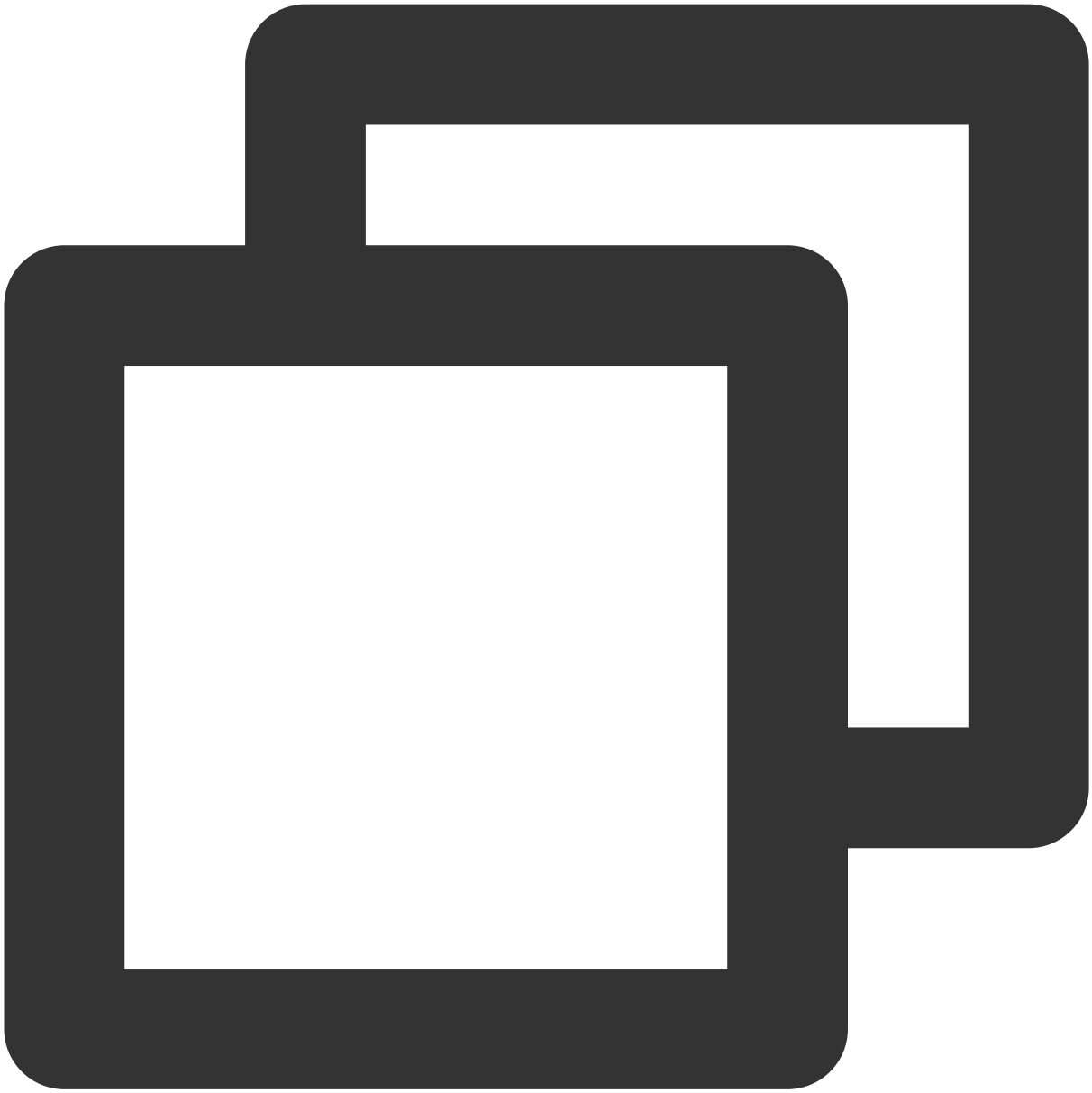
Statements

Last updated：2024-07-12 16:55:12

if Statements

In WXS, you can use if statements in the following formats:
if (expression) statement : Execute statement when expression is truthy.
if (expression) statement1 else statement2 : Execute statement1 when expression is truthy. Otherwise, execute statement2.
if ... else if ... else statementN Using this statement, you can choose one of the executions from statement1 to statementN.
Example Syntax:

©2013-2022 Tencent Cloud. All rights reserved.



```
// if ...

if (Expression) Statement;

if (Expression)
  Statement;

if (Expression) {
  Code block;
}
```

©2013-2022 Tencent Cloud. All rights reserved.

```
// if ... else

if (Expression) Statement;
else Statement;

if (Expression)
  Statement;
else
  Statement;

if (Expression) {
  Code block;
} else {
  Code block;
}

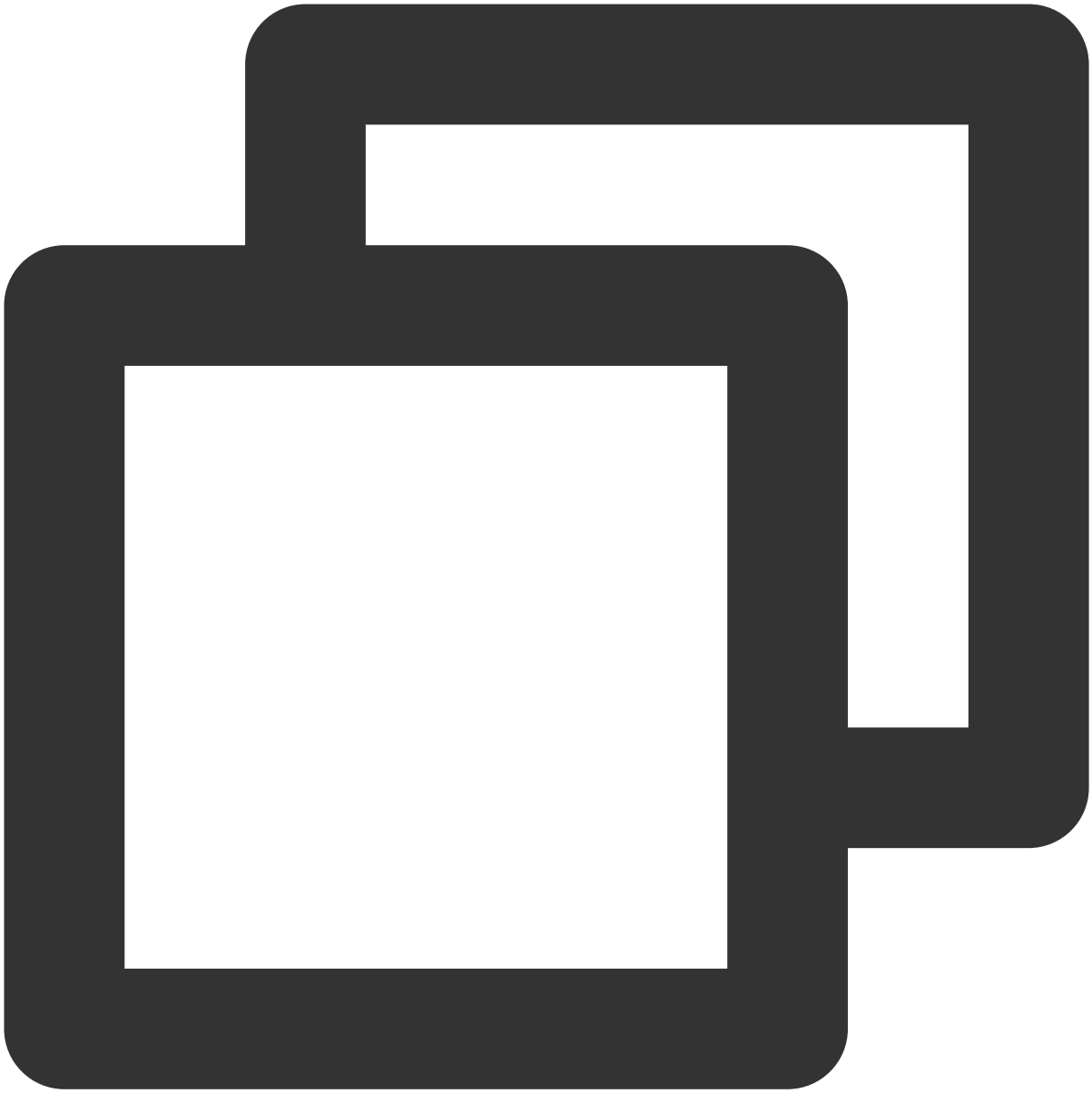
// if ... else if ... else ...

if (Expression) {
  Code block;
} else if (Expression) {
  Code block;
} else if (Expression) {
  Code block;
} else {
  Code block;
}
```

switch Statements

Example Syntax:

©2013-2022 Tencent Cloud. All rights reserved.



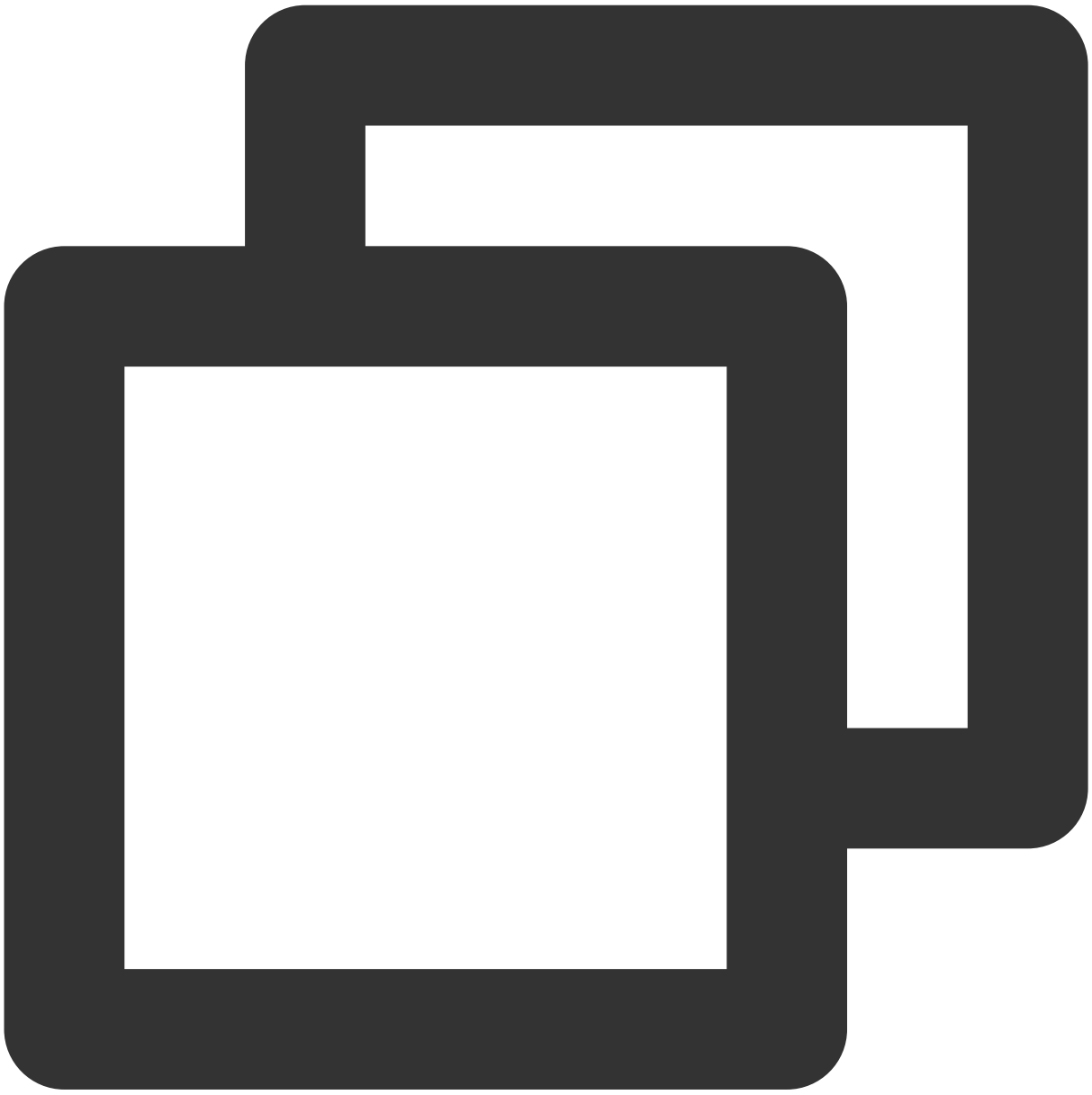
```
switch (Expression) {
  case Variable:
    Statement;
  case Number:
    Statement;
    break;
  case String:
    Statement;
  default:
    Statement;
}
```

©2013-2022 Tencent Cloud. All rights reserved.

default branches can be omitted.

After the case keyword, you can only use a variable, number, or string.

Sample code:



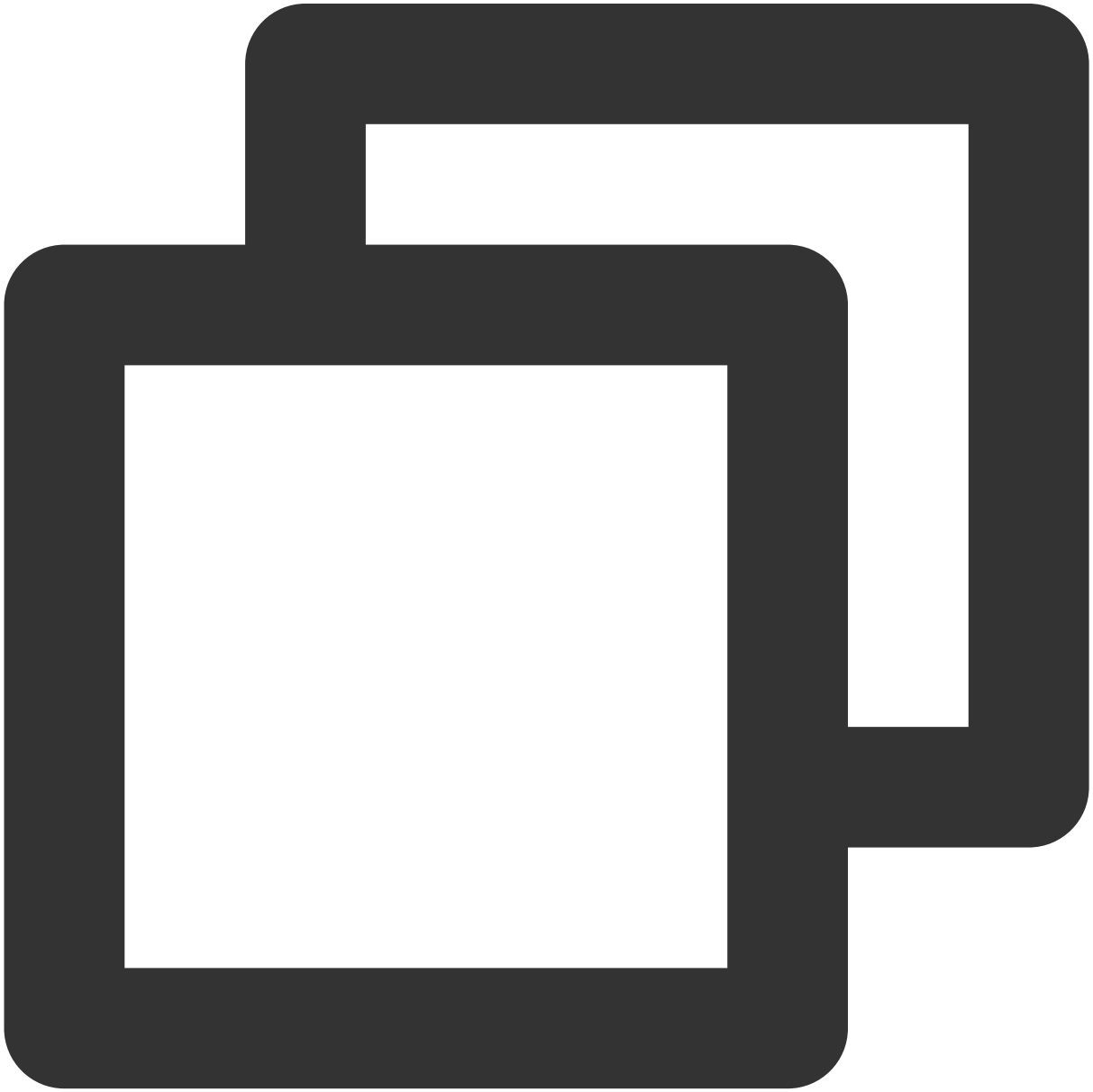
```
var exp = 10;

switch ( exp ) {
case "10":
  console.log("string 10");
  break;
```

©2013-2022 Tencent Cloud. All rights reserved.

```
case 10:
  console.log("number 10");
  break;
case exp:
  console.log("var exp");
  break;
default:
  console.log("default");
}
```

Output:

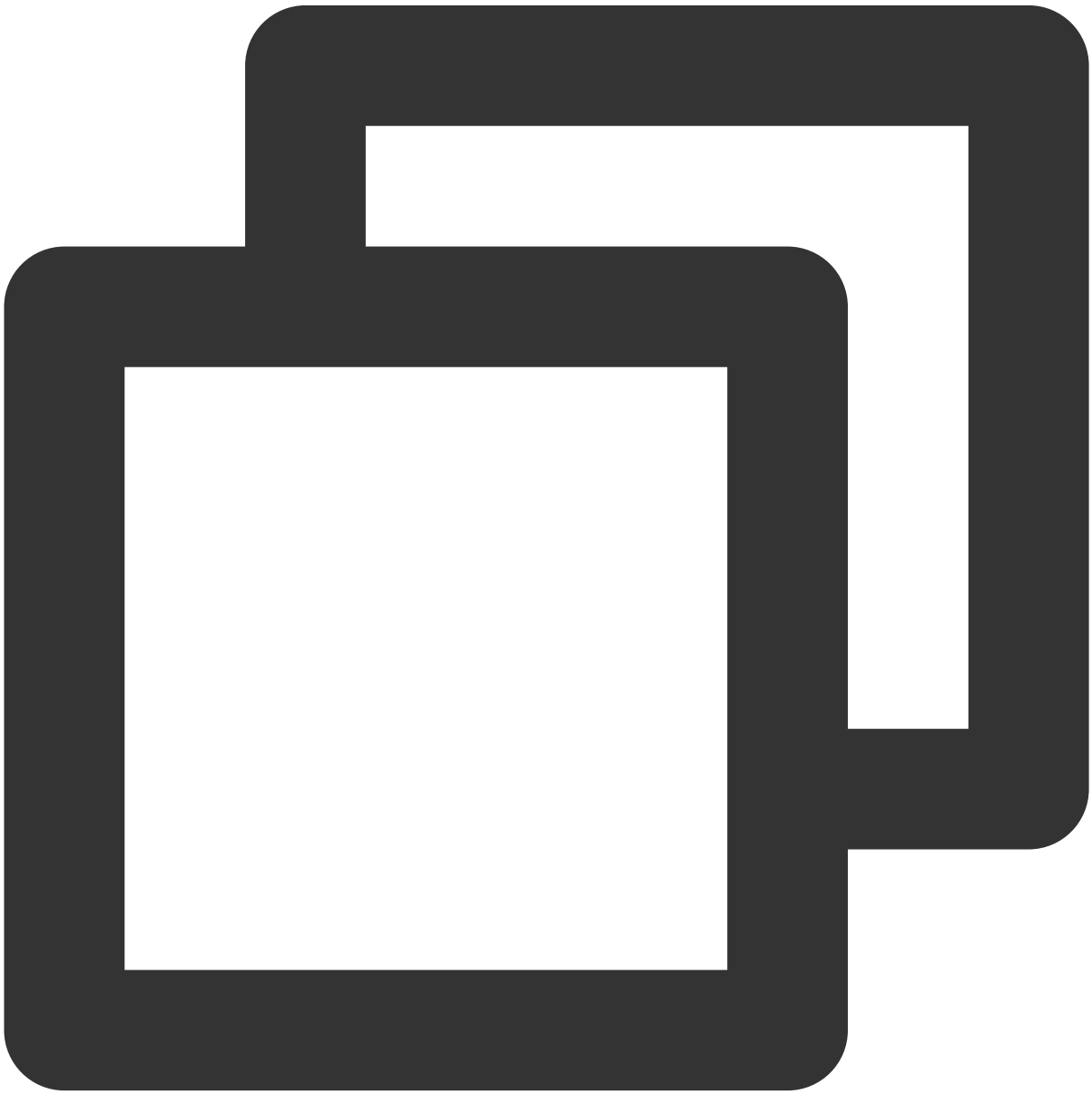


©2013-2022 Tencent Cloud. All rights reserved.

```
number 10
```

for Statements

Example Syntax:



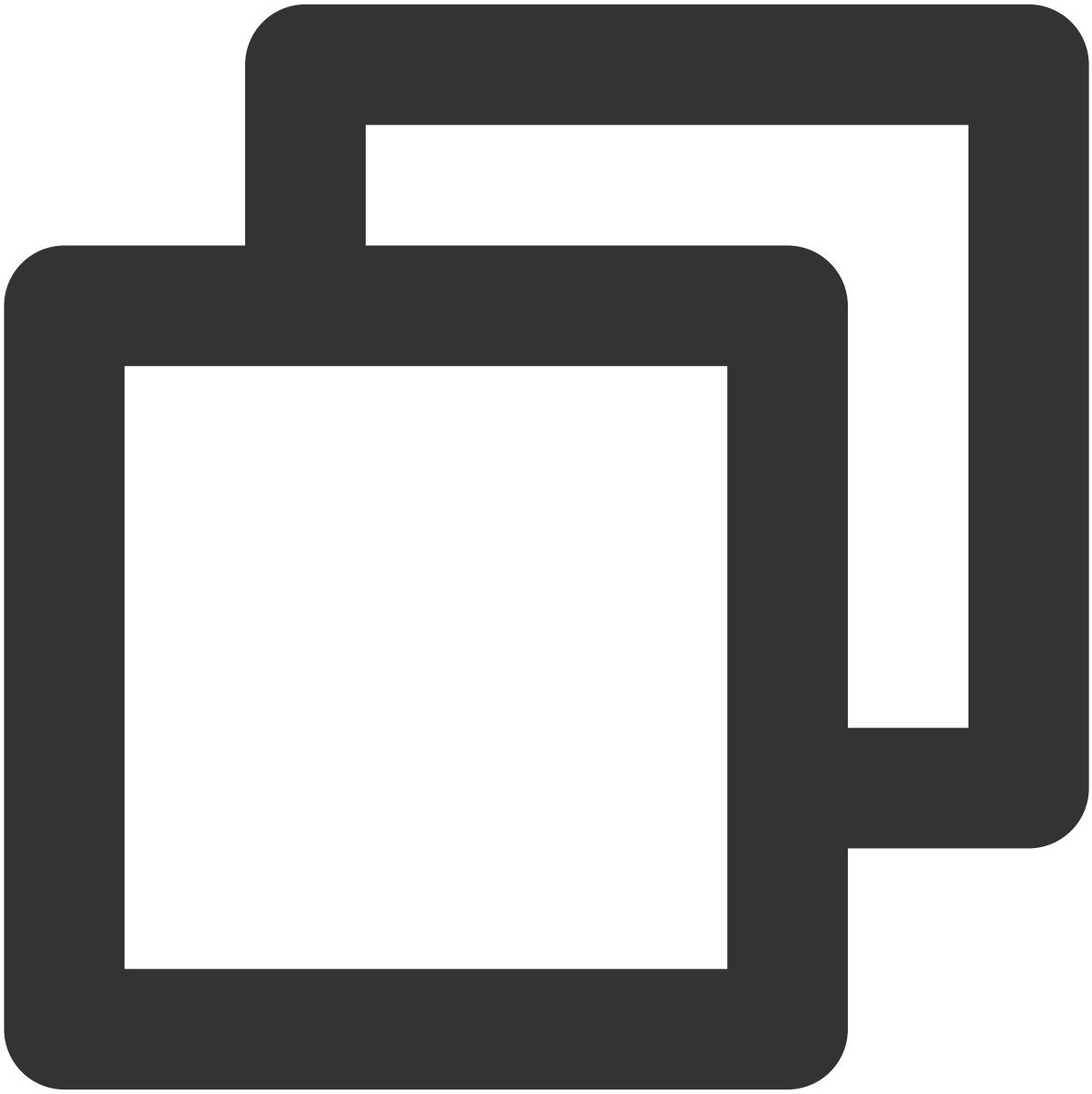
```
for (Statement; Statement; Statement)
  Statement;
```

©2013-2022 Tencent Cloud. All rights reserved.

```
for (Statement; Statement; Statement) {
  Code block;
}
```

Supports the use of break and continue keywords.

Sample code:

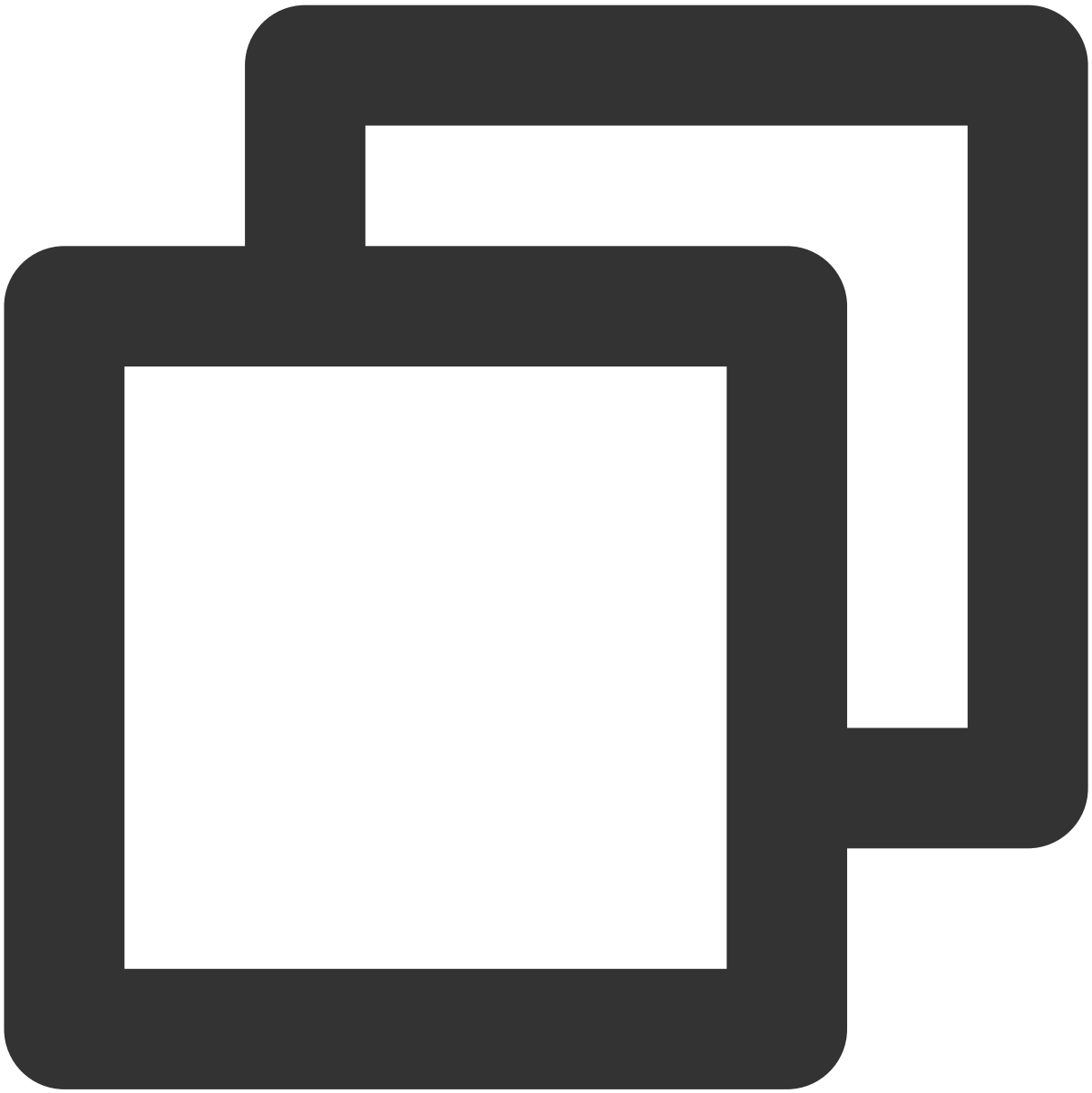


```
for (var i = 0; i < 3; ++i) {
  console.log(i);
  if( i >= 1) break;
```

©2013-2022 Tencent Cloud. All rights reserved.

```
}
```

Output:

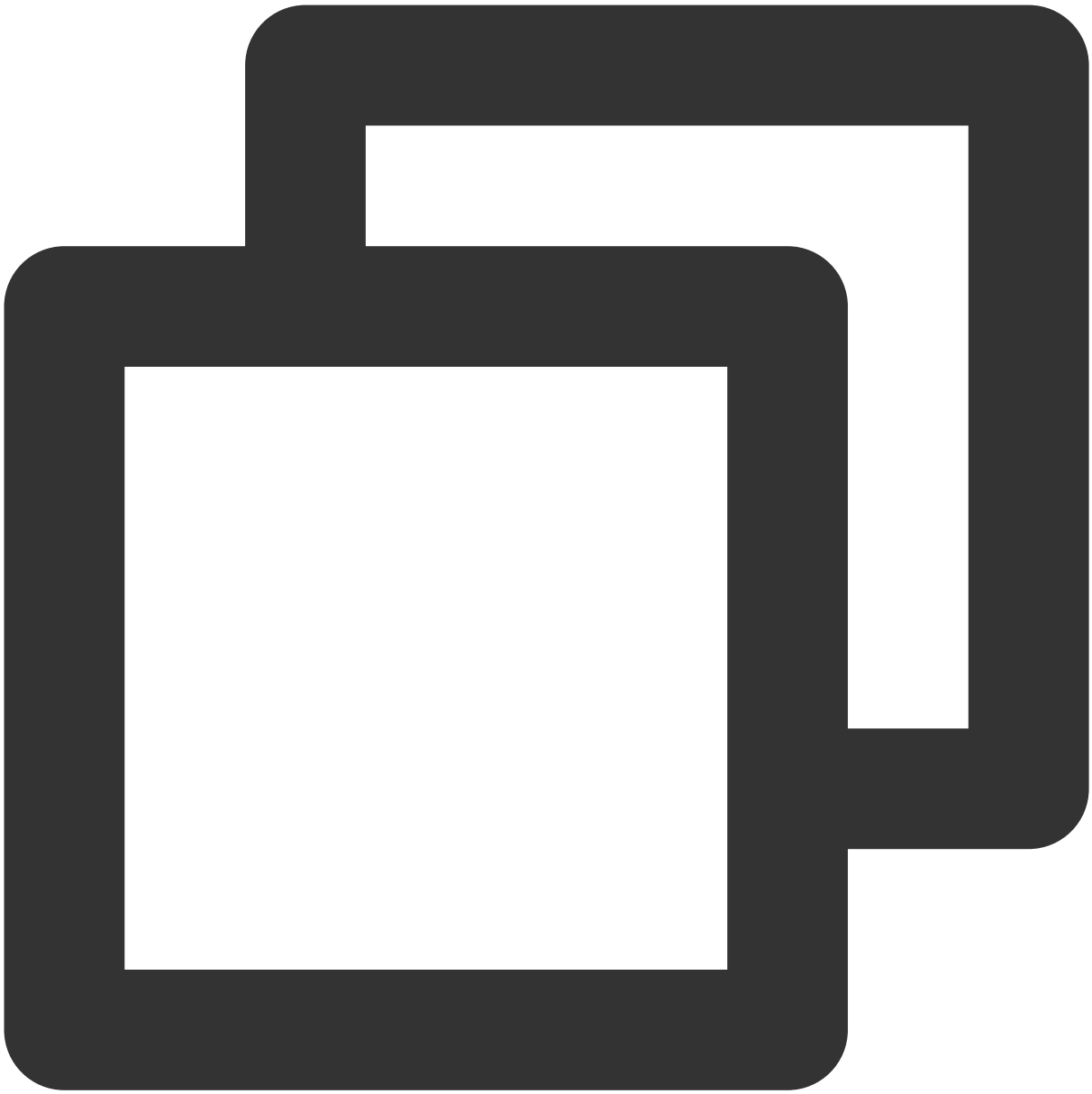


```
0
1
```

while Statements

©2013-2022 Tencent Cloud. All rights reserved.

Example Syntax:



```
while (Expression)
  Statement;

while (Expression){
  Code block;
}

do {
  Code block;
```

©2013-2022 Tencent Cloud. All rights reserved.

```
} while (Expression)
```

When expression is true, loop execute statement or code block.

Supports the use of break and continue keywords.

©2013-2022 Tencent Cloud. All rights reserved.

Data Types

Last updated：2024-07-12 16:55:44

The WXS language currently supports the following data types:

number;

string;

boolean;

object;

function;

array;

date;

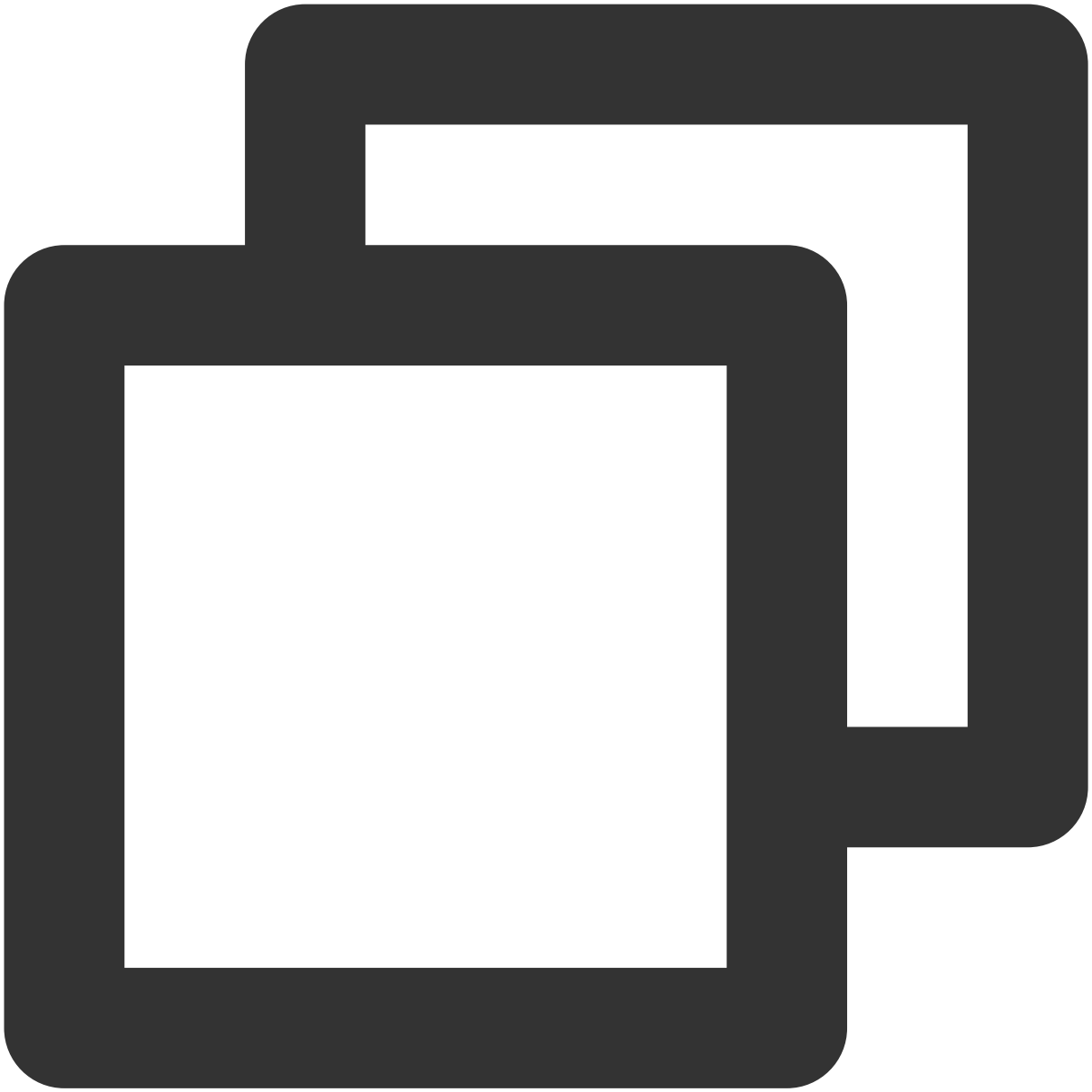
regexp;

number

Syntax

There are two types of numbers: integers and decimals.

©2013-2022 Tencent Cloud. All rights reserved.



```
var a = 10;
var PI = 3.141592653589793;
```

Properties

constructor: Returns the string "Number"

Methods

`toString` ;

©2013-2022 Tencent Cloud. All rights reserved.

`toLocaleString` ;
`valueOf` ;
`toFixed` ;
`toExponential` ;
`toPrecision` 。

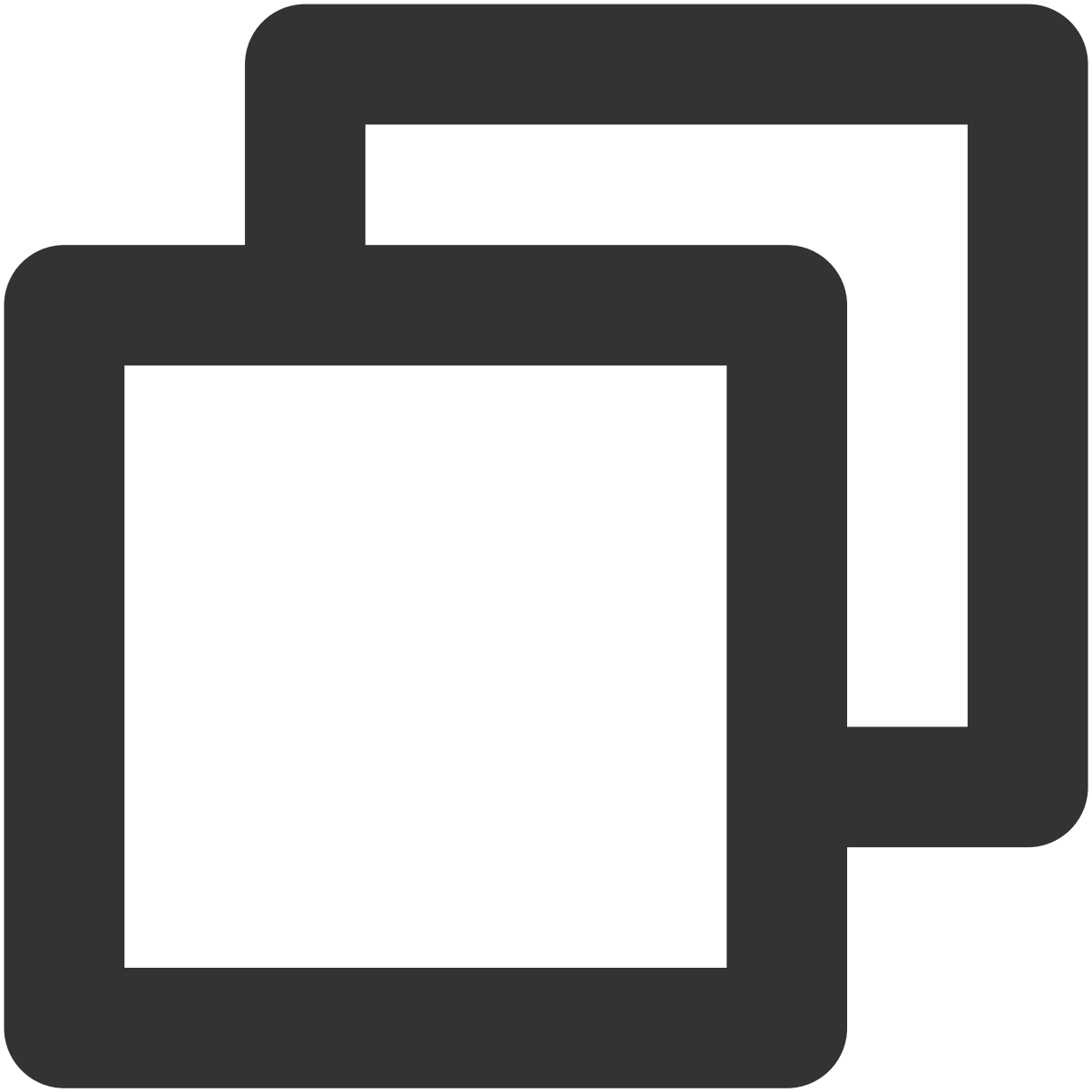
For information on using the above methods, refer to the ES5 standard.

string

Syntax

here are two ways to write a string:

©2013-2022 Tencent Cloud. All rights reserved.



```
'hello world';
"hello world";
```

Properties

`constructor` : Returns the string "String";

`length` 。

For explanations of properties other than constructor, refer to the ES5 standard.

©2013-2022 Tencent Cloud. All rights reserved.

Methods

`toString` ;
`valueOf` ;
`charAt` ;
`charCodeAt` ;
`concat` ;
`indexOf` ;
`lastIndexOf` ;
`localeCompare` ;
`match` ;
`replace` ;
`search` ;
`slice` ;
`split` ;
`substring` ;
`toLowerCase` ;
`toLocaleLowerCase` ;
`toUpperCase` ;
`toLocaleUpperCase` ;
`trim` 。

For information on using the above methods, refer to the ES5 standard.

boolean

Syntax

Boolean values can take one of two values: true or false.

Properties

`constructor` : Returns the string "Boolean".

Methods

`toString` ;

©2013-2022 Tencent Cloud. All rights reserved.

`valueOf` 。

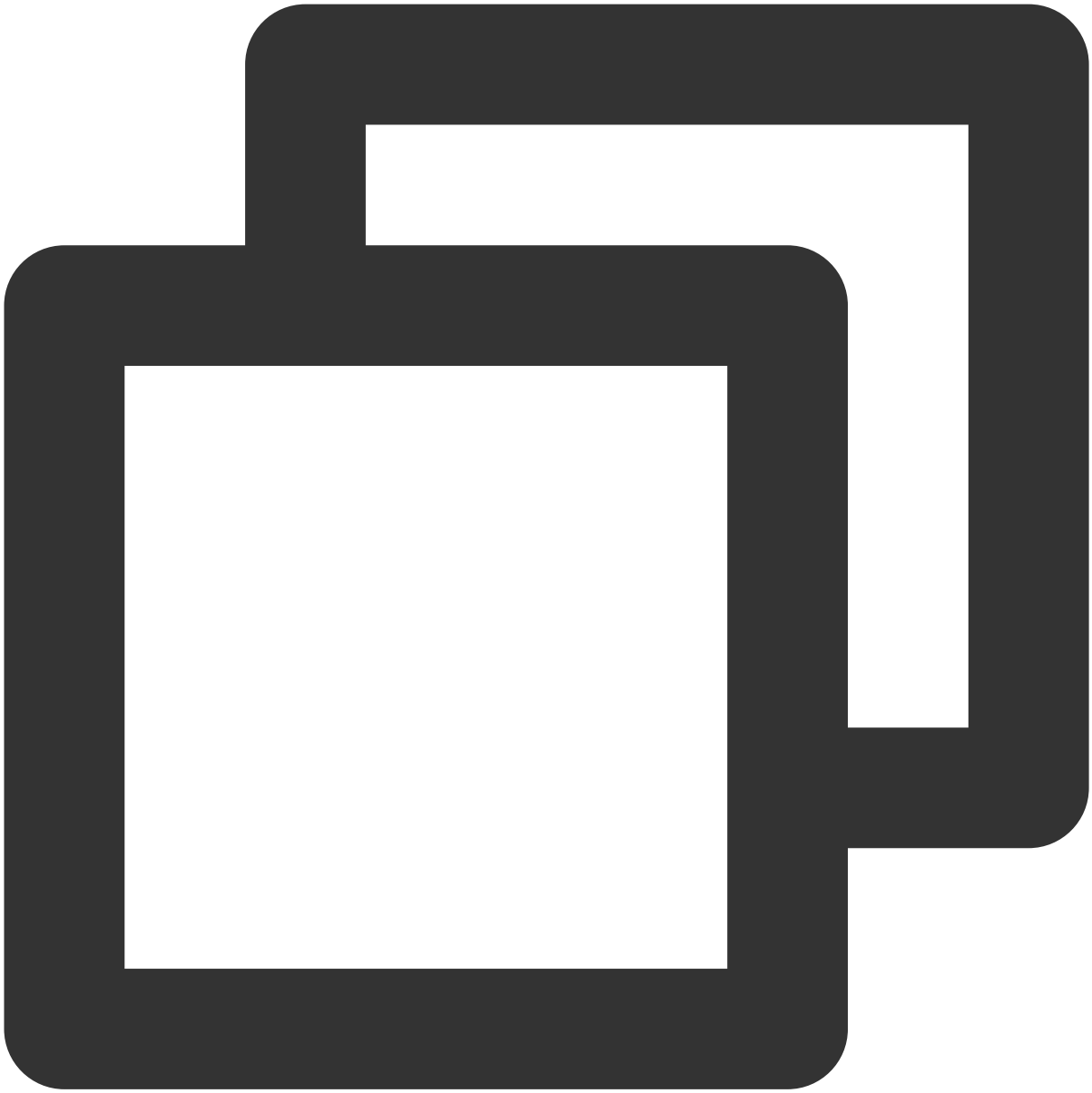
For information on using the above methods, refer to the ES5 standard.

object

Syntax

An object is a type of unordered key-value pair, which can be used in the following ways:

©2013-2022 Tencent Cloud. All rights reserved.



```
var o = {} // Generates a new empty object.

// Generates a new non-empty object.
o = {
  'string'   : 1,  //The object's key can be a string.
  const_var : 2,  //The object's key can also be an identifier that conforms to the
  func       : {}, //The object's value can be of any type.
};

//Object property read operation
console.log(1 === o['string']);
```

©2013-2022 Tencent Cloud. All rights reserved.

```
console.log(2 === o.const_var);

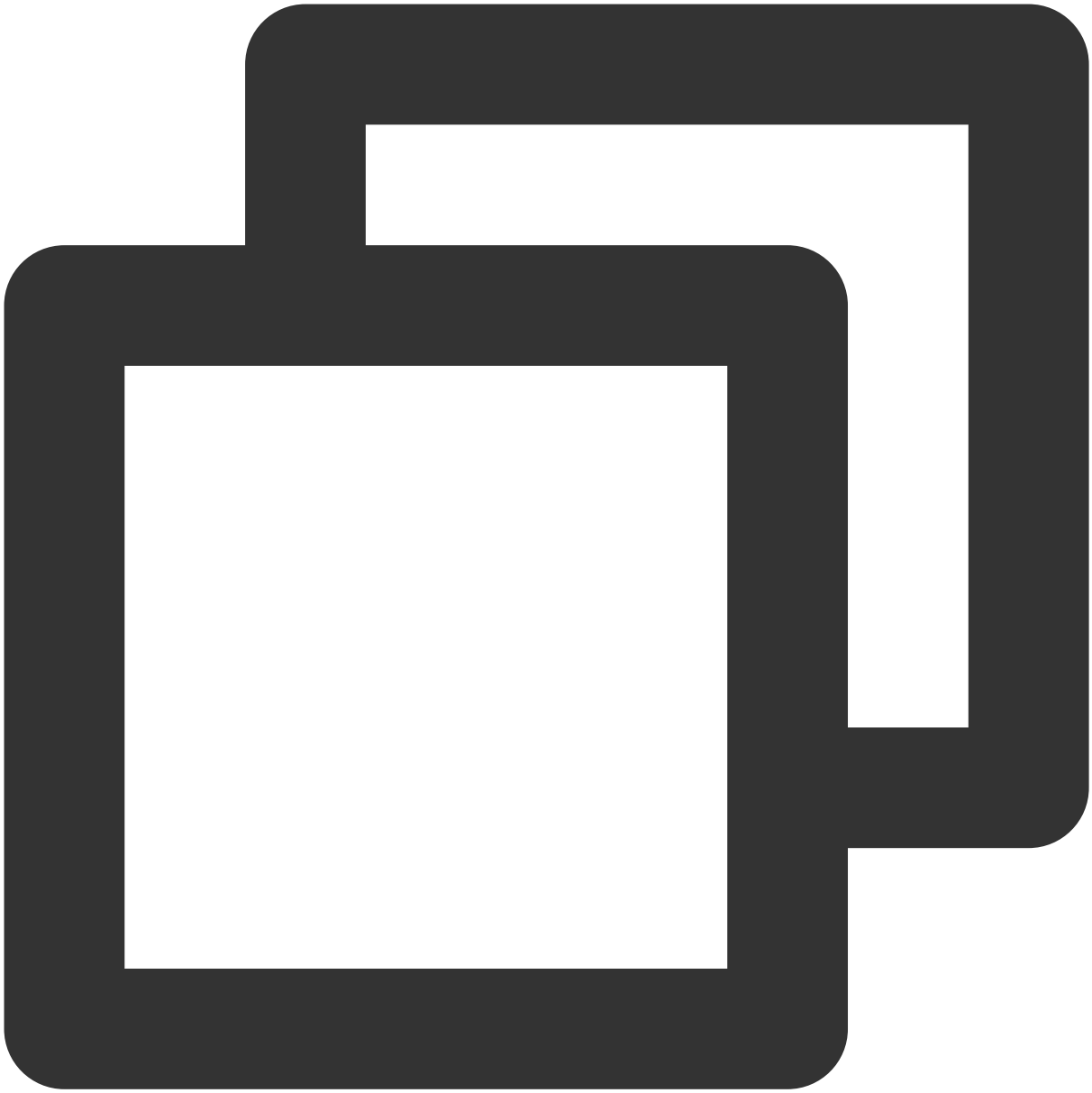
//Object property write operation
o['string']++;
o['string'] += 10;
o.const_var++;
o.const_var += 10;

//Object property read operation
console.log(12 === o['string']);
console.log(13 === o.const_var);
```

Properties

constructor: Returns the string "Object".

©2013-2022 Tencent Cloud. All rights reserved.



```
console.log("Object" === {k:"1",v:"2"}.constructor)
```

Methods

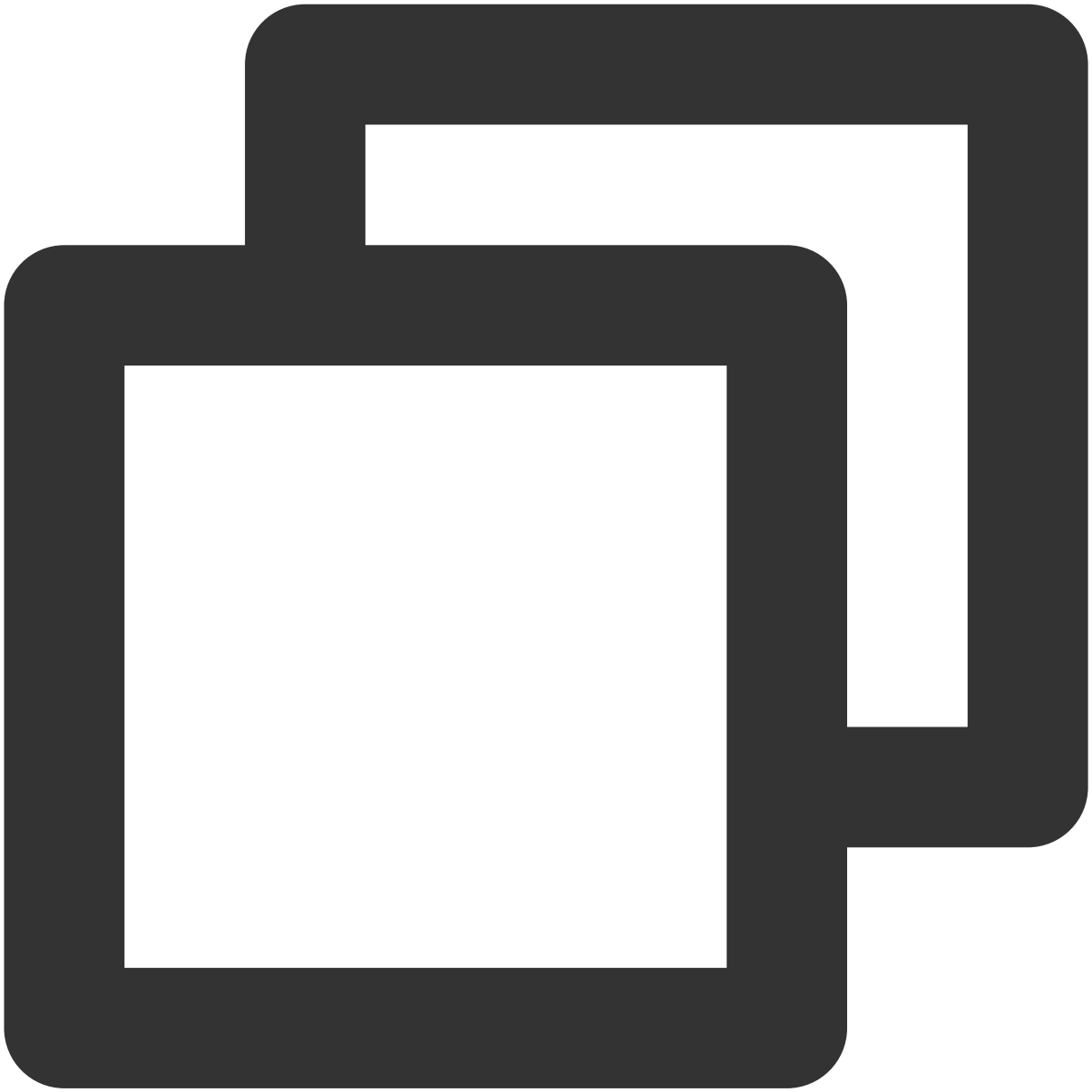
toString: Returns the string "[object Object]".

function

©2013-2022 Tencent Cloud. All rights reserved.

Syntax

function supports the following definition methods:



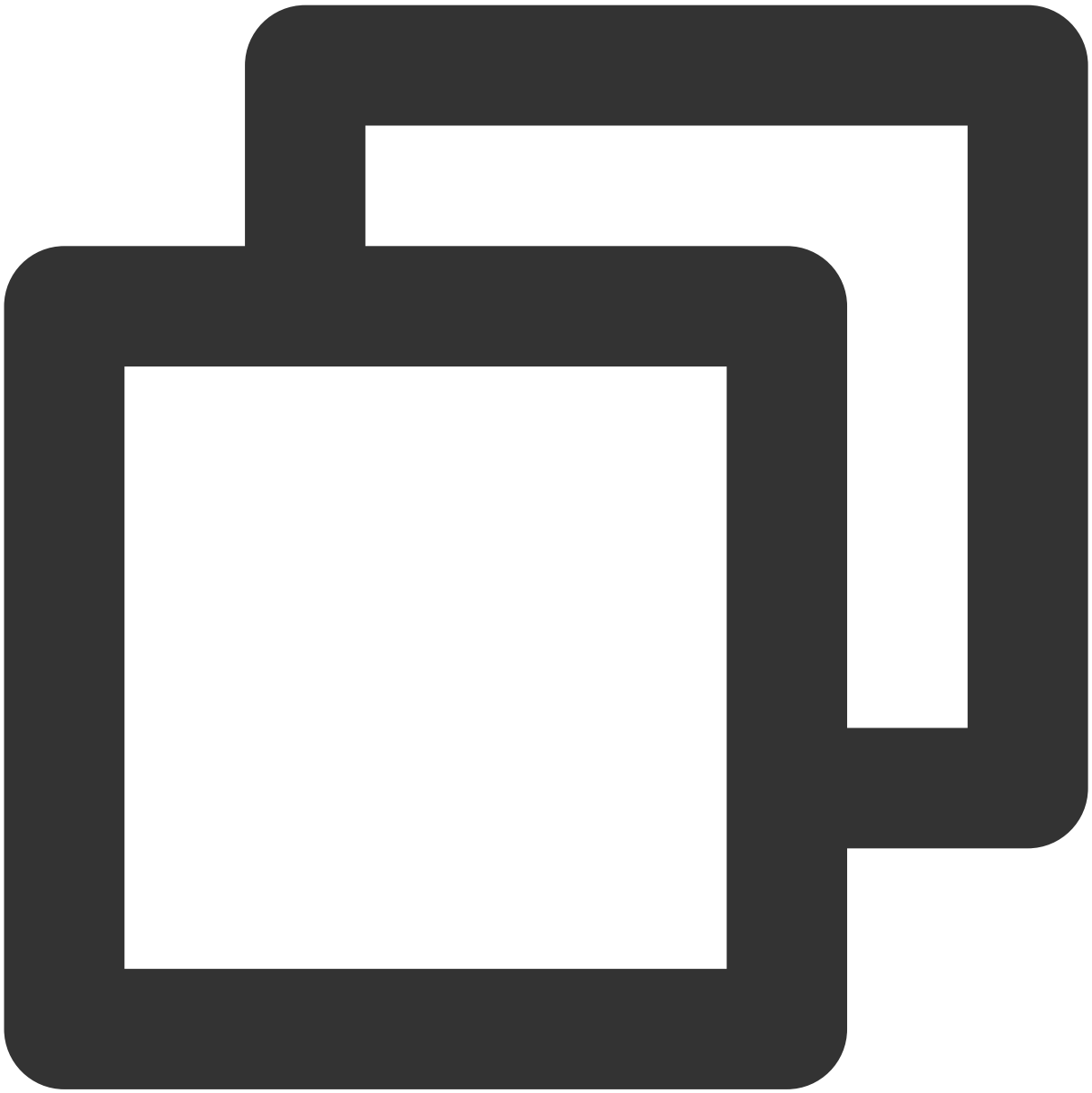
```
//Method 1
function a (x) {
  return x;
}

//Method 2
var b = function (x) {
  return x;
```

©2013-2022 Tencent Cloud. All rights reserved.

```
}
```

function also supports the following syntax (anonymous functions, closures, etc.):



```
var a = function (x) {
  return function () { return x;}
}

var b = a(100);
console.log( 100 === b() );
```

©2013-2022 Tencent Cloud. All rights reserved.

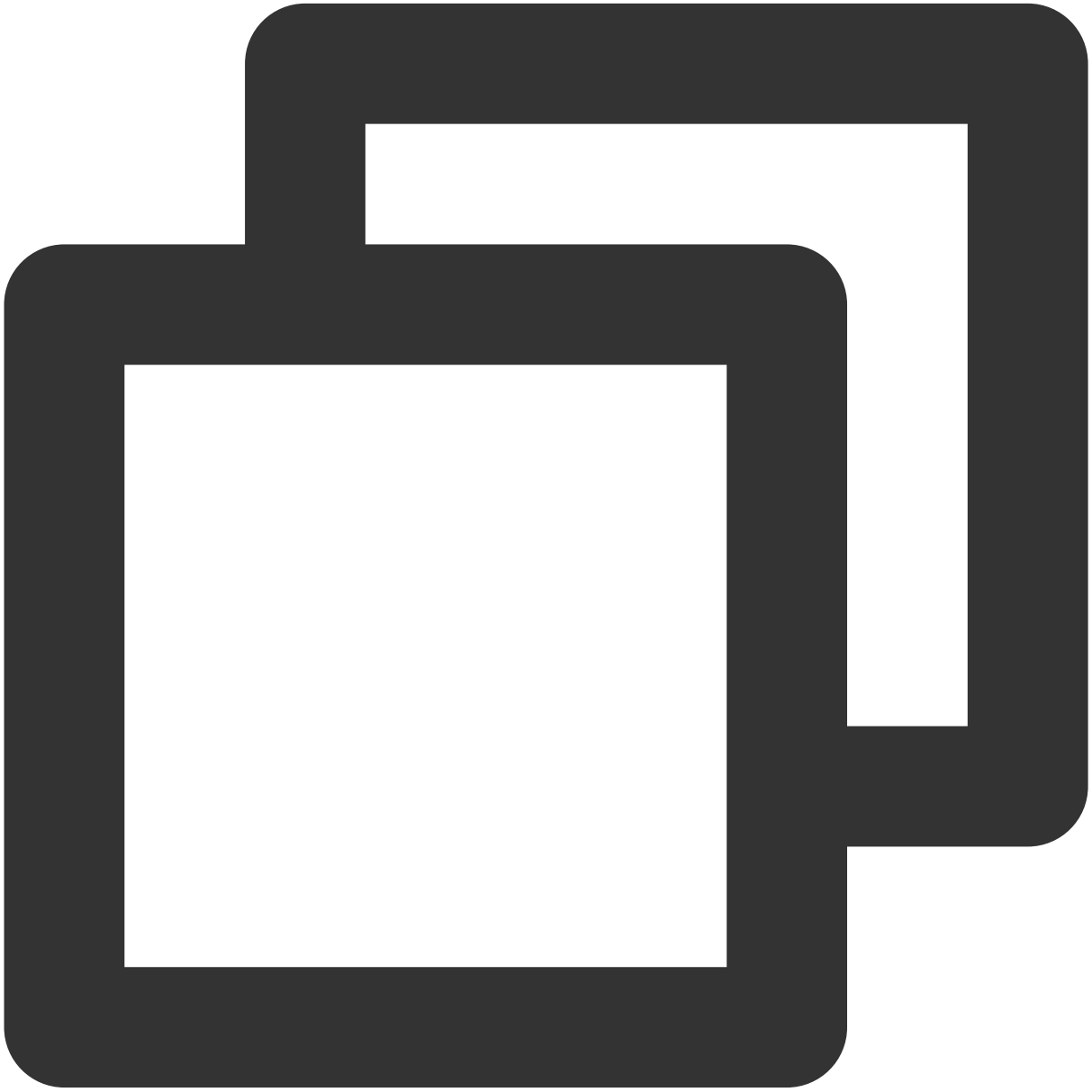
arguments

In function, you can use the arguments keyword. This keyword only has the following properties at present:

length: The number of arguments passed to the function.

[index]: Each argument passed to the function can be traversed by the index subscript.

Sample code:



```
var a = function(){
  console.log(3 === arguments.length);
  console.log(100 === arguments[0]);
  console.log(200 === arguments[1]);
```

©2013-2022 Tencent Cloud. All rights reserved.

```
  console.log(300 === arguments[2]);
};
a(100,200,300);
```

Properties

constructor: Returns the string "Function".

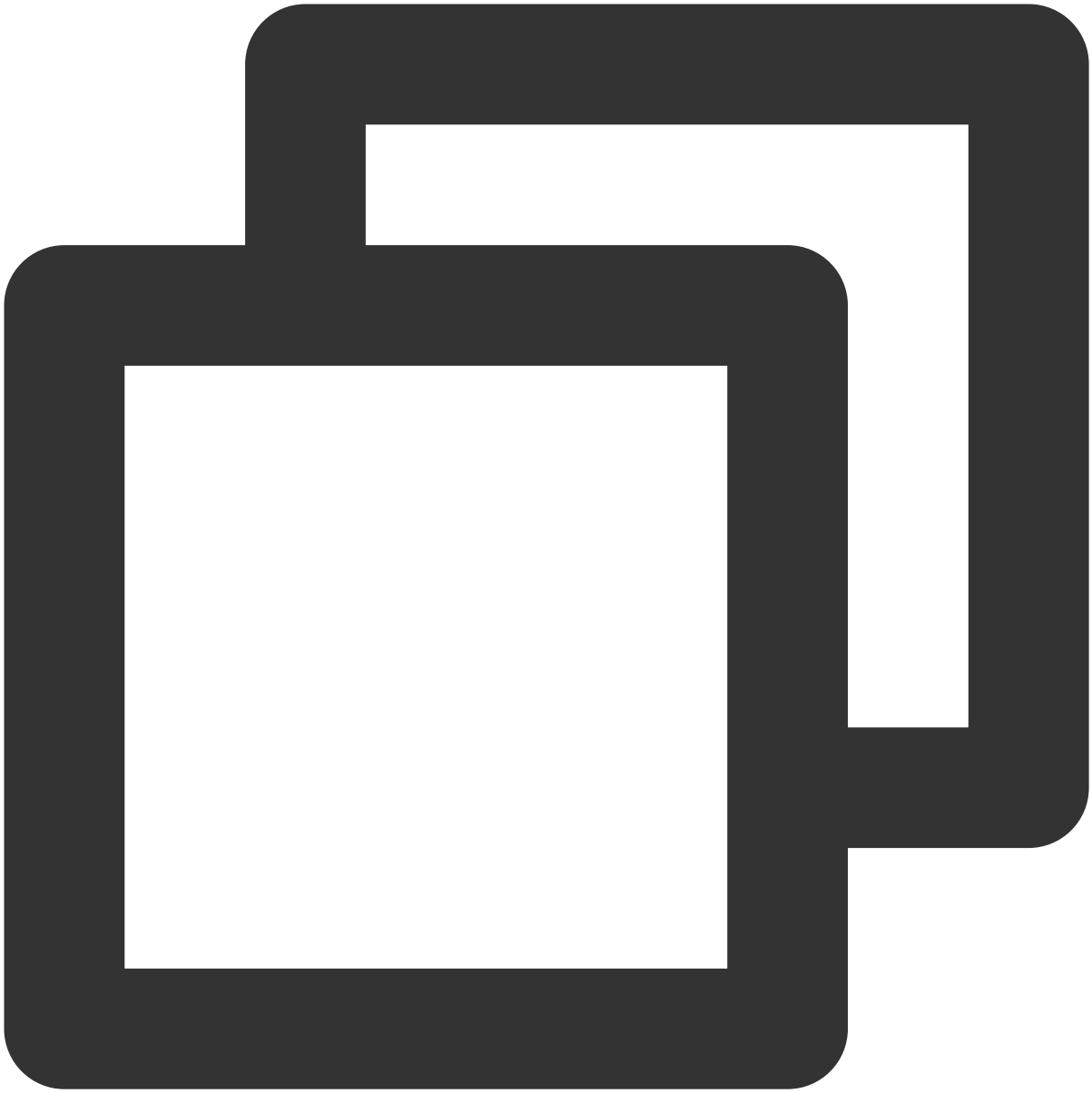
length: Returns the number of formal parameters of the function

Methods

toString: Returns the string "[function Function]".

Sample code:

©2013-2022 Tencent Cloud. All rights reserved.



```
var func = function (a,b,c) { }

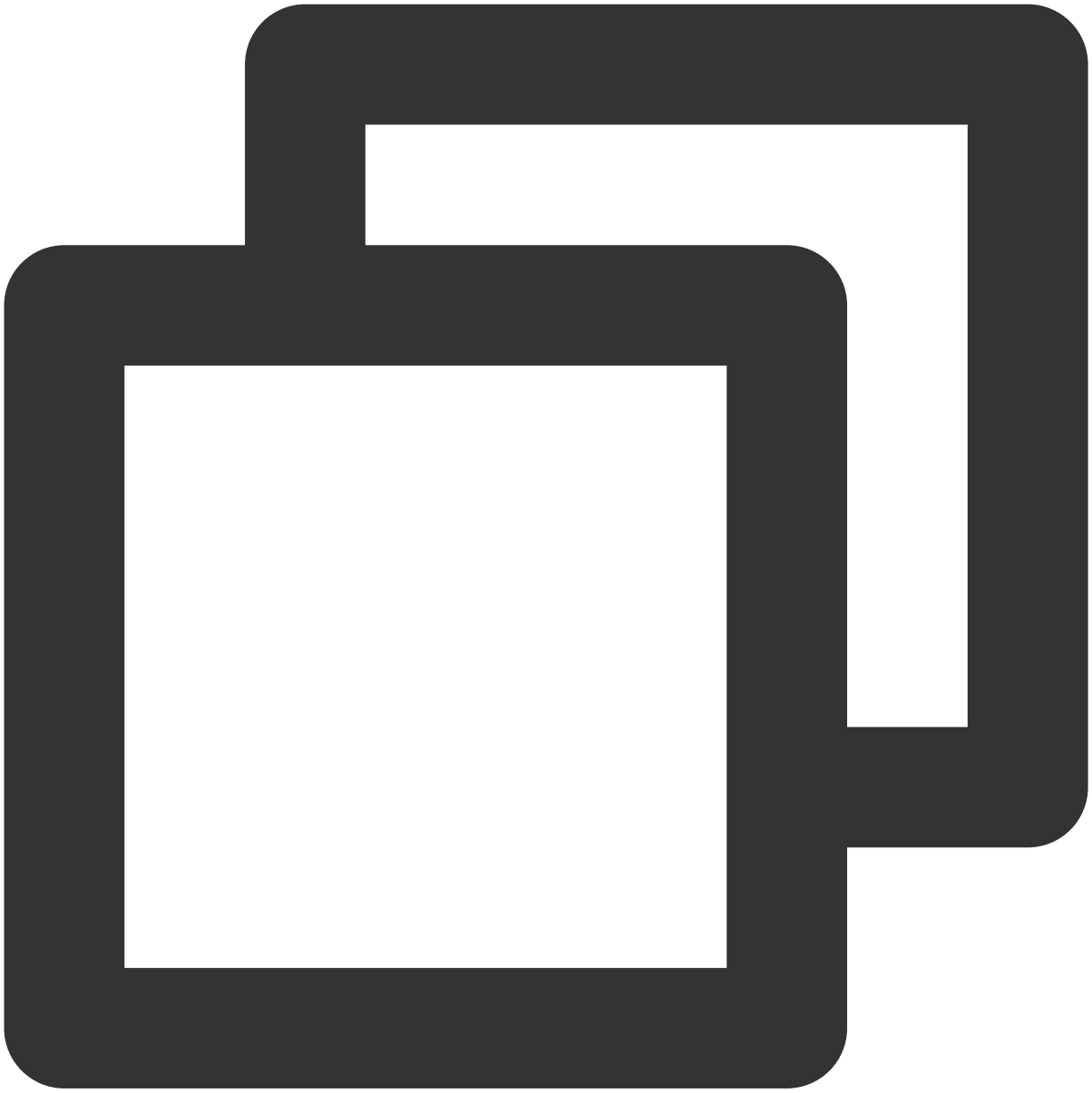
console.log("Function" === func.constructor);
console.log(3 === func.length);
console.log("[function Function]" === func.toString());
```

array

©2013-2022 Tencent Cloud. All rights reserved.

Syntax

array supports the following definition methods:



```
var a = [];       // Generates a new empty array.

a = [1,"2",{},function(){}];  // Generates a new non-empty array, with array elemen
```

Properties

`constructor` : Returns the string "Array";

©2013-2022 Tencent Cloud. All rights reserved.

`length` 。

For explanations of properties other than constructor, refer to the ES5 standard.

Methods

`toString` ;
`concat` ;
`join` ;
`pop` ;
`push` ;
`reverse` ;
`shift` ;
`slice` ;
`sort` ;
`splice` ;
`unshift` ;
`indexOf` ;
`lastIndexOf` ;
`every` ;
`some` ;
`forEach` ;
`map` ;
`filter` ;
`reduce` ;
`reduceRight` 。

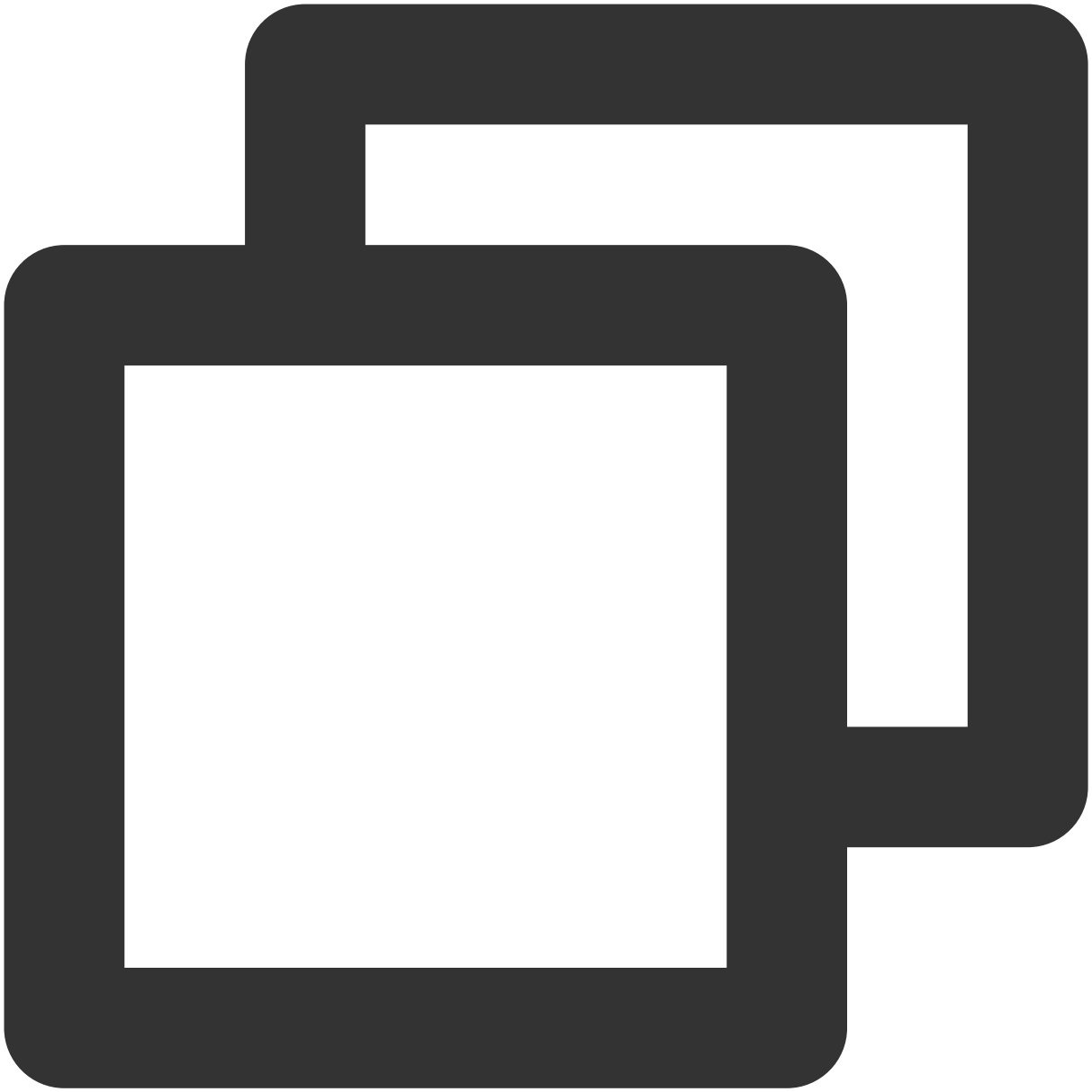
For information on using the above methods, refer to the ES5 standard.

date

Syntax

Generates the getDate function required by the date object and returns a current time object.

©2013-2022 Tencent Cloud. All rights reserved.



```
getDate()
getDate(milliseconds)
getDate(datestring)
getDate(year, month[, date[, hours[, minutes[, seconds[, milliseconds]]]]])
```

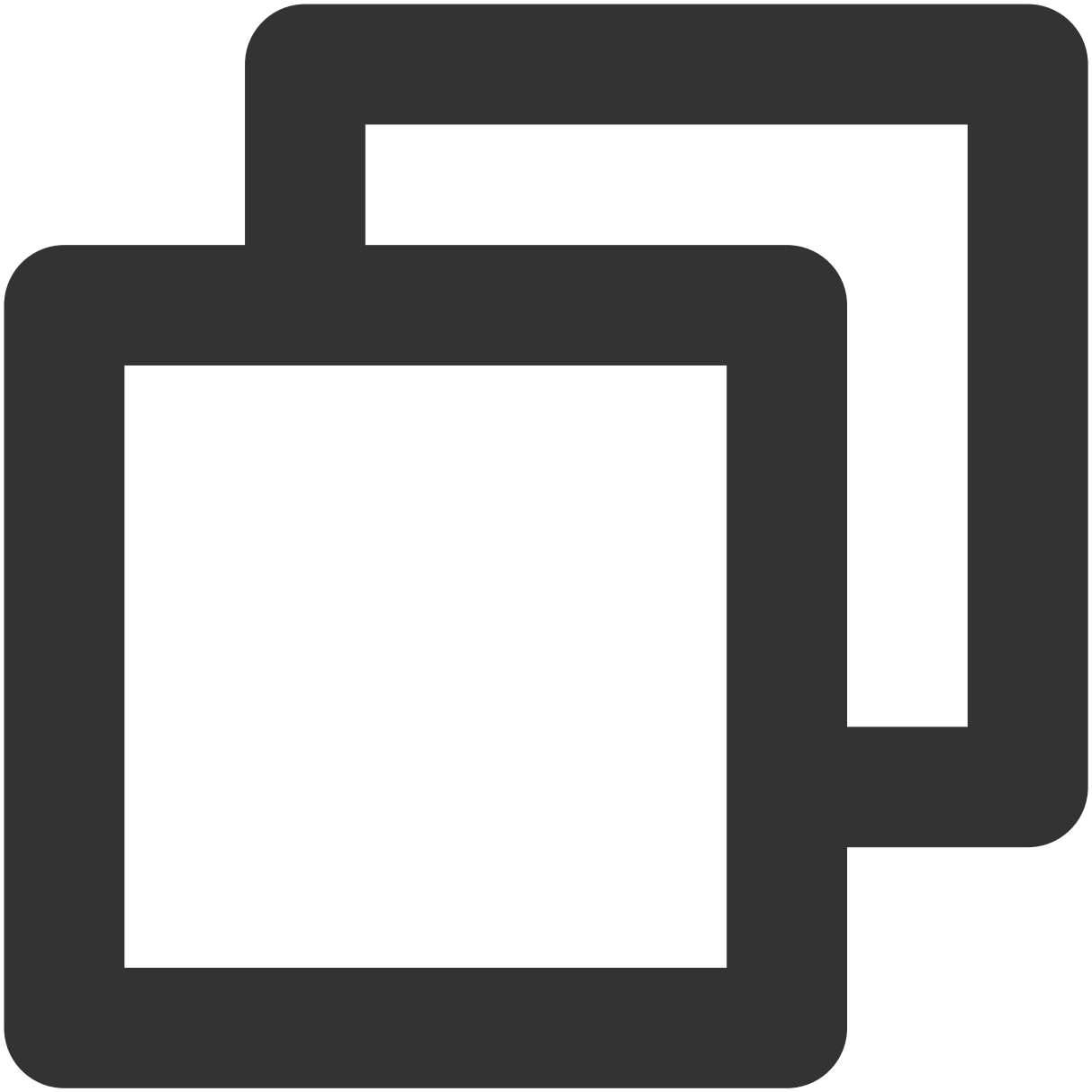
Parameters

milliseconds: The number of milliseconds from 00:00:00 UTC January 1, 1970.

datestring: The date string, in the format: "month day, year hours:minutes:seconds".

Sample code:

©2013-2022 Tencent Cloud. All rights reserved.



```
var date = getDate(); //Returns the current time object.

date = getDate(1500000000000);
// Fri Jul 14 2017 10:40:00 GMT+0800 (China Standard Time)
date = getDate('2017-7-14');
// Fri Jul 14 2017 00:00:00 GMT+0800 (China Standard Time)
date = getDate(2017, 6, 14, 10, 40, 0, 0);
// Fri Jul 14 2017 10:40:00 GMT+0800 (China Standard Time)
```

Properties

©2013-2022 Tencent Cloud. All rights reserved.

constructor: Returns the string "Date".

Methods

`toString` ;
`toDateString` ;
`toTimeString` ;
`toLocaleString` ;
`toLocaleDateString` ;
`toLocaleTimeString` ;
`valueOf` ;
`getTime` ;
`getFullYear` ;
`getUTCFullYear` ;
`getMonth` ;
`getUTCMonth` ;
`getDate` ;
`getUTCDate` ;
`getDay` ;
`getUTCDay` ;
`getHours` ;
`getUTCHours` ;
`getMinutes` ;
`getUTCMinutes` ;
`getSeconds` ;
`getUTCSeconds` ;
`getMilliseconds` ;
`getUTCMilliseconds` ;
`getTimezoneOffset` ;
`setTime` ;
`setMilliseconds` ;
`setUTCMilliseconds` ;
`setSeconds` ;
`setUTCSeconds` ;
`setMinutes` ;
`setUTCMinutes` ;
`setHours` ;

©2013-2022 Tencent Cloud. All rights reserved.

`setUTCHours` ;
`setDate` ;
`setUTCDate` ;
`setMonth` ;
`setUTCMonth` ;
`setFullYear` ;
`setUTCFullYear` ;
`toUTCString` ;
`toISOString` ;
`toJSON` 。

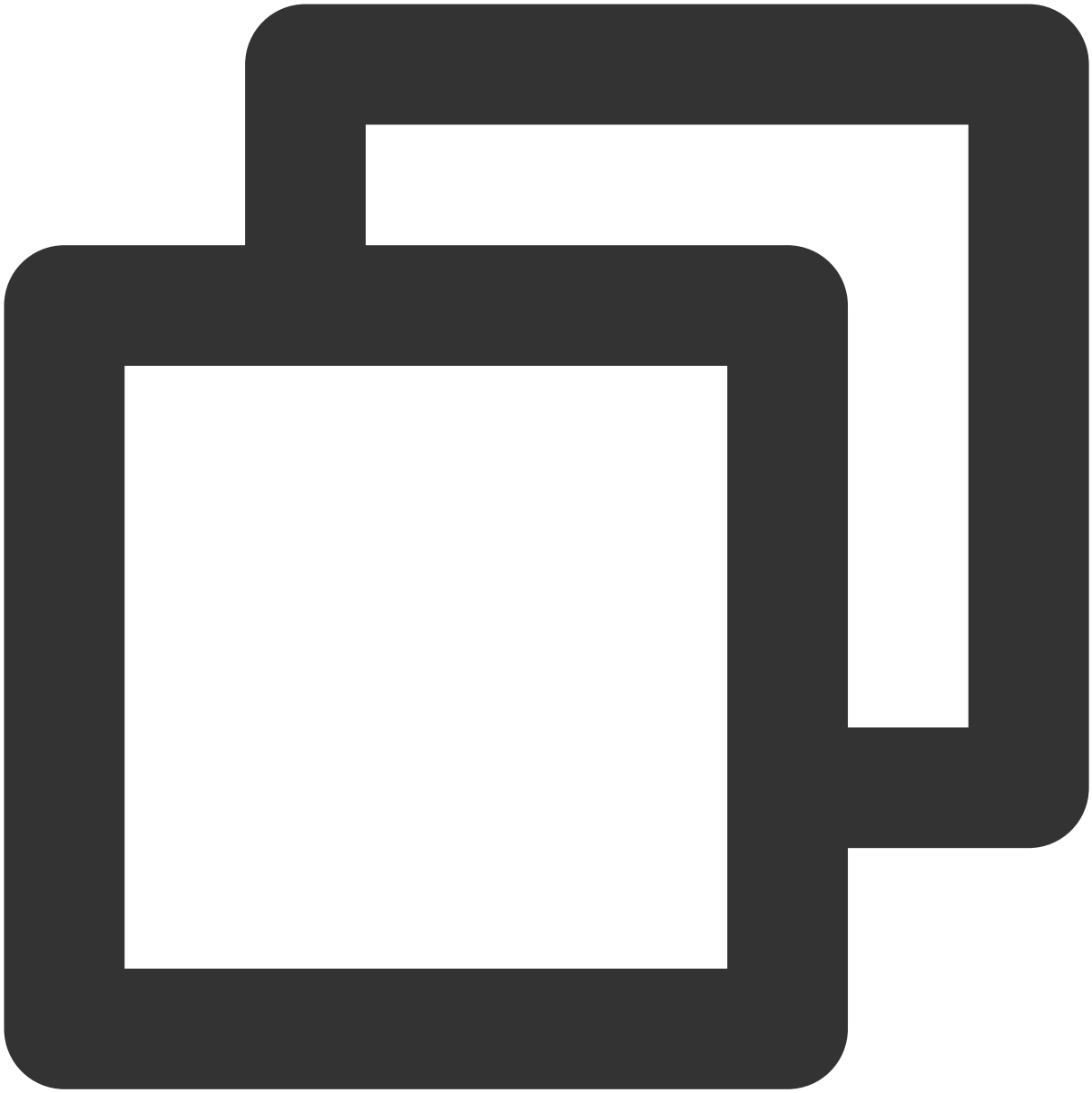
For information on using the above methods, refer to the ES5 standard.

regexp

Syntax

Generates the getRegExp function to be used by the regexp object.

©2013-2022 Tencent Cloud. All rights reserved.



```
getRegExp(pattern[, flags])
```

Parameters:

pattern: The content of the regular expression.

flags: The modifiers. This field can only include the following characters:

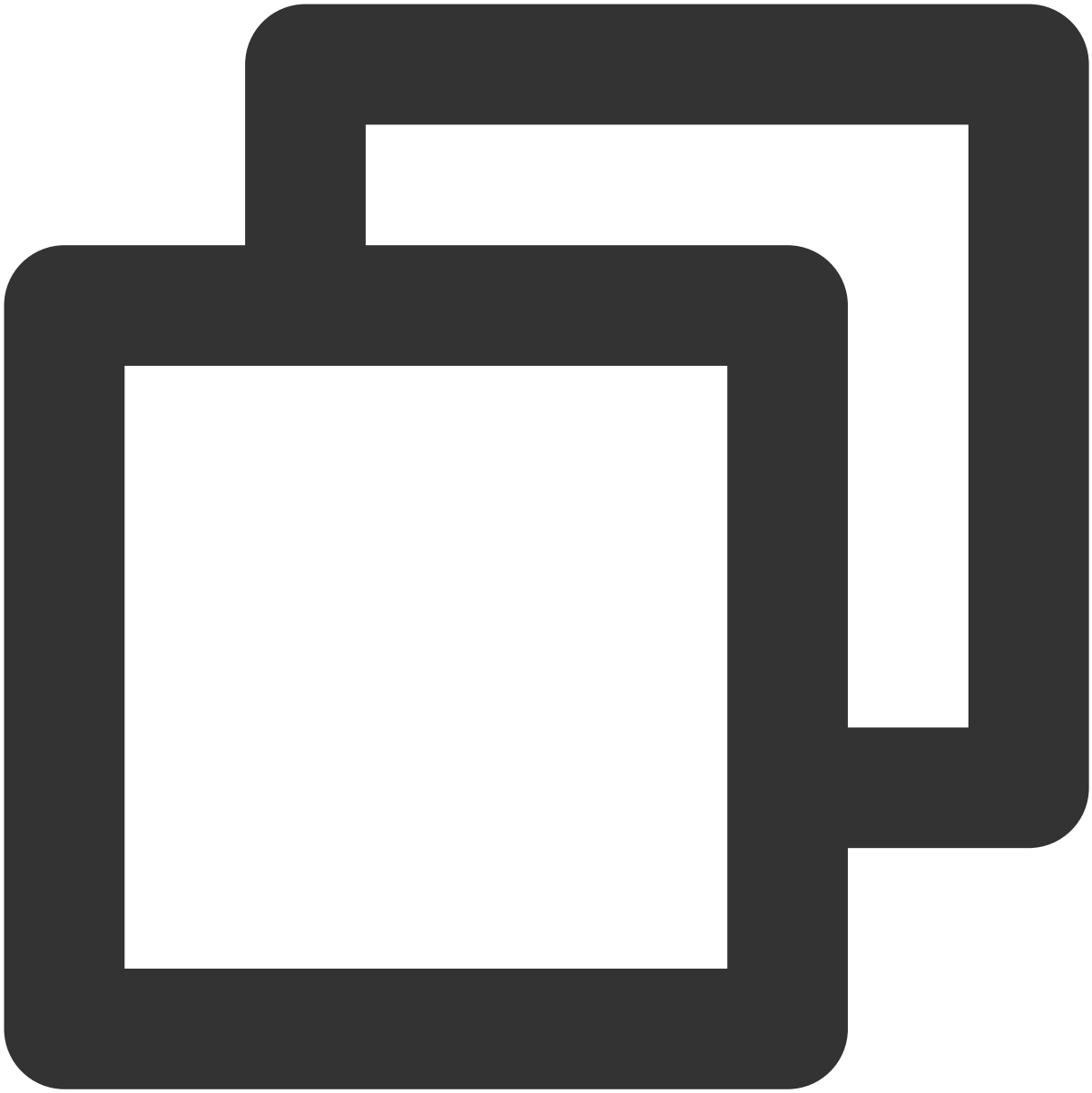
g: global

i: ignoreCase

m: multiline

Sample code:

©2013-2022 Tencent Cloud. All rights reserved.



```
var a = getRegExp("x", "img");
console.log("x" === a.source);
console.log(true === a.global);
console.log(true === a.ignoreCase);
console.log(true === a.multiline);
```

Properties

`constructor` : Returns the string `"RegExp"` ;

`source` ;

©2013-2022 Tencent Cloud. All rights reserved.

`global` ;
`ignoreCase` ;
`multiline` ;
`lastIndex` 。

For explanations of properties other than constructor, refer to the ES5 standard.

Methods

`exec` ;
`test` ;
`toString` 。

For information on using the above methods, refer to the ES5 standard.

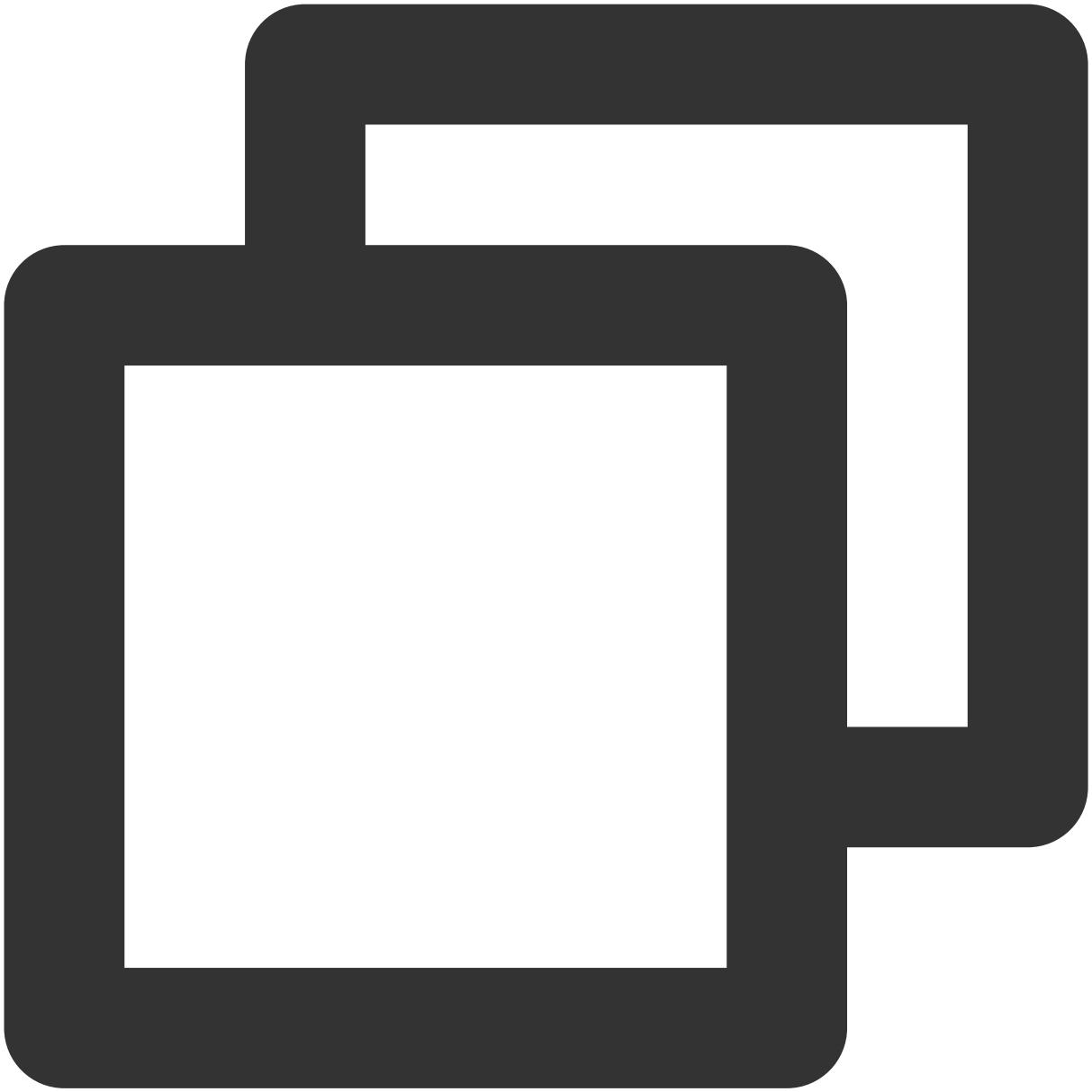
9. Data Type Determination

constructor Property

You can use the constructor property to determine the data type.

Sample code:

©2013-2022 Tencent Cloud. All rights reserved.



```
var number = 10;
console.log( "Number" === number.constructor );

var string = "str";
console.log( "String" === string.constructor );

var boolean = true;
console.log( "Boolean" === boolean.constructor );

var object = {};
console.log( "Object" === object.constructor );
```

©2013-2022 Tencent Cloud. All rights reserved.

```
var func = function(){};
console.log( "Function" === func.constructor );

var array = [];
console.log( "Array" === array.constructor );

var date = getDate();
console.log( "Date" === date.constructor );

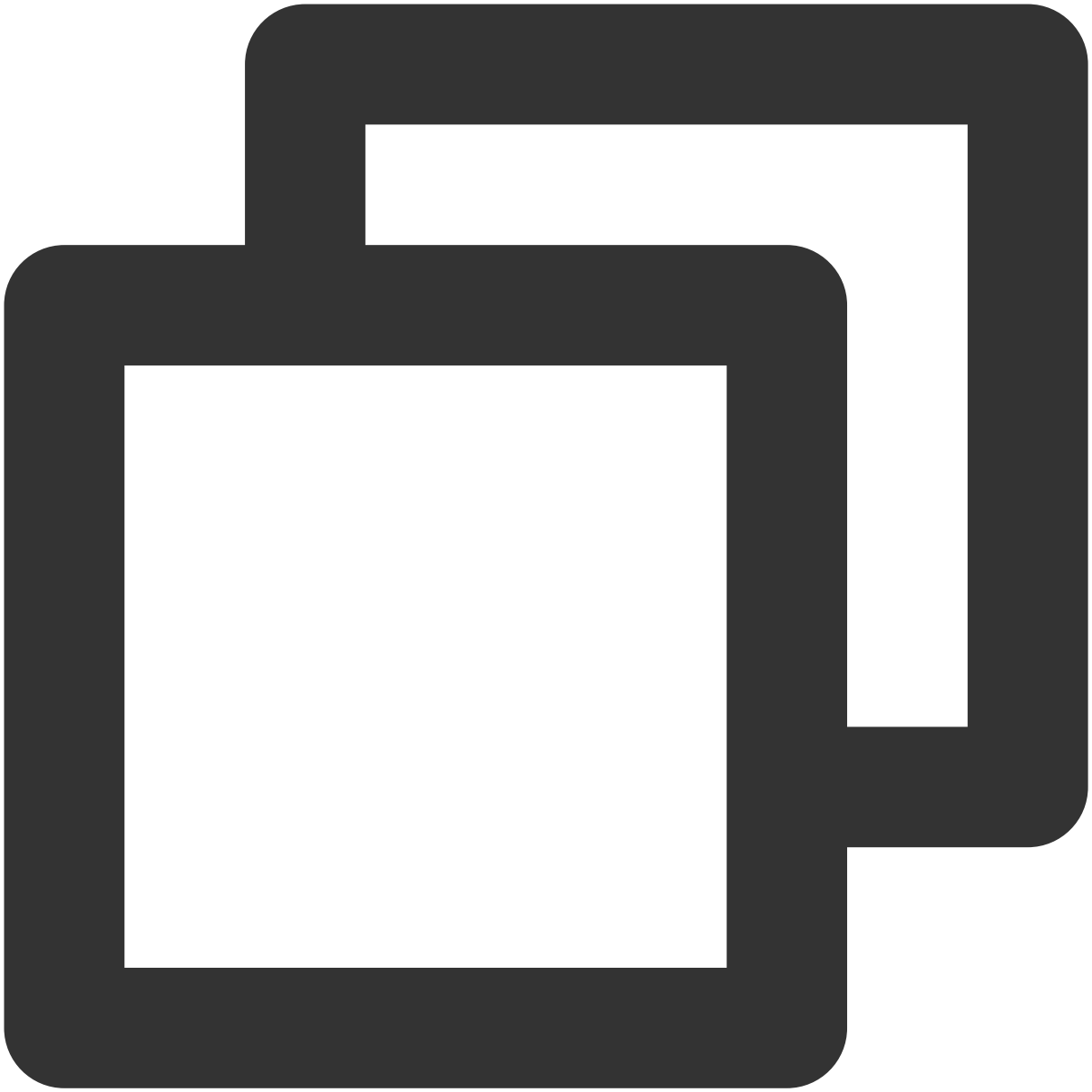
var regexp = getRegExp();
console.log( "RegExp" === regexp.constructor );
```

typeof

You can also use typeof to distinguish certain data types.

Sample code:

©2013-2022 Tencent Cloud. All rights reserved.



```
var number = 10;
var boolean = true;
var object = {};
var func = function(){};
var array = [];
var date = getDate();
var regexp = getRegExp();

console.log( 'number' === typeof number );
console.log( 'boolean' === typeof boolean );
console.log( 'object' === typeof object );
```

©2013-2022 Tencent Cloud. All rights reserved.

```
console.log( 'function' === typeof func );
console.log( 'object' === typeof array );
console.log( 'object' === typeof date );
console.log( 'object' === typeof regexp );

console.log( 'undefined' === typeof undefined );
console.log( 'object' === typeof null );
```

©2013-2022 Tencent Cloud. All rights reserved.

Base Library

Last updated：2024-07-12 16:53:47

console

The `console.log` method is used to output messages in the console window. This method can accept multiple parameters and concatenate their results for output.

Math

Properties

`E`
`LN10`
`LN2`
`LOG2E`
`LOG10E`
`PI`
`SQRT1_2`
`SQRT2`

> For information on using the above properties, refer to the ES5 standard.。

Methods

`abs`
`acos`
`asin`
`atan`
`atan2`
`ceil`
`cos`
`exp`
`floor`
`log`

©2013-2022 Tencent Cloud. All rights reserved.

`max`
`min`
`pow`
`random`
`round`
`sin`
`sqrt`
`tan`

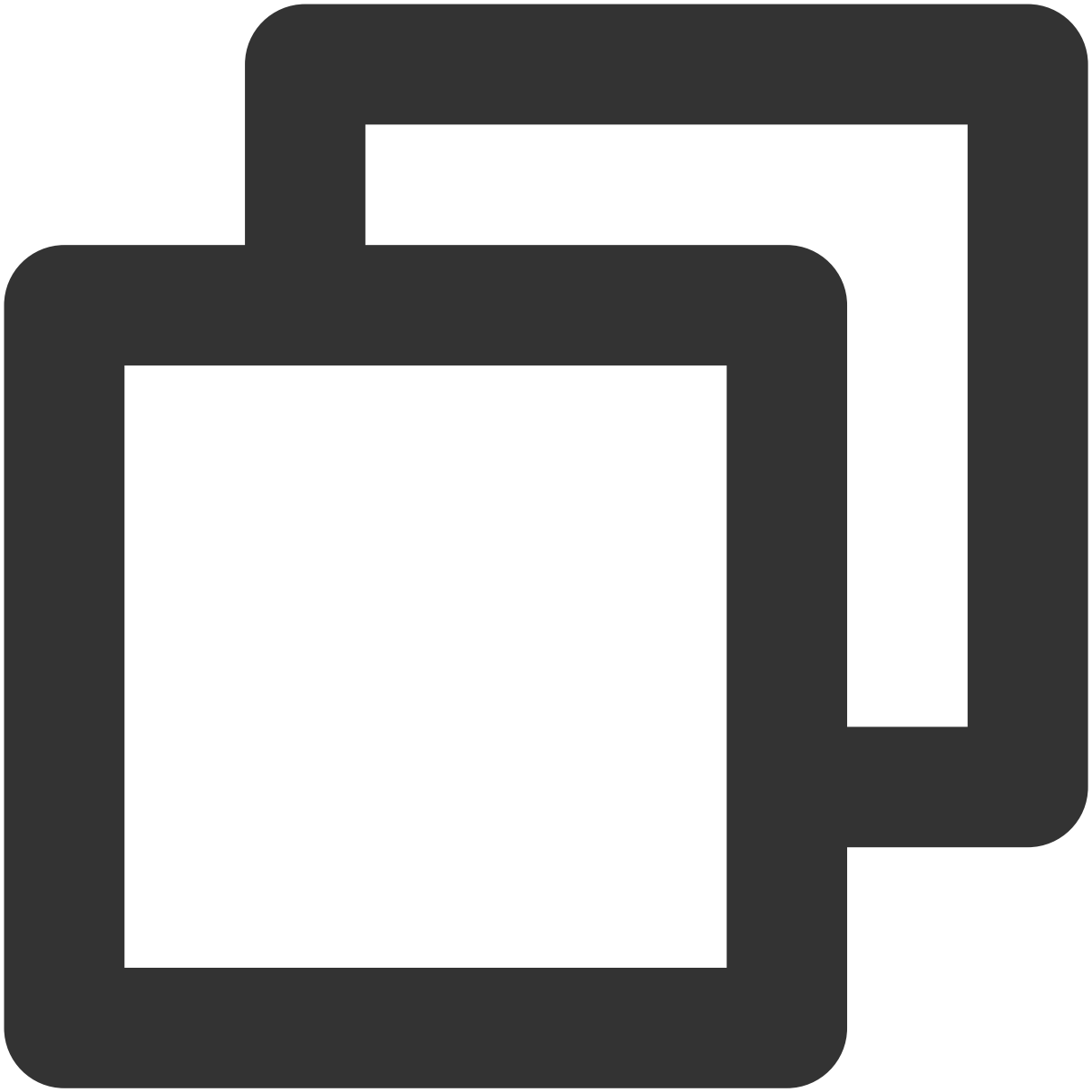
For information on using the above properties, refer to the ES5 standard.

JSON

Methods

stringify(object): Converts the `object` object into a `JSON` string and returns the string.
parse(string): Converts the `JSON` string into an object and returns the object.
Sample code:

©2013-2022 Tencent Cloud. All rights reserved.



```
console.log(undefined === JSON.stringify());
console.log(undefined === JSON.stringify(undefined));
console.log("null"===JSON.stringify(null));

console.log("111"===JSON.stringify(111));
console.log('"111"'===JSON.stringify("111"));
console.log("true"===JSON.stringify(true));
console.log(undefined===JSON.stringify(function(){}));


console.log(undefined===JSON.parse(JSON.stringify()));
```

©2013-2022 Tencent Cloud. All rights reserved.

```
console.log(undefined===JSON.parse(JSON.stringify(undefined)));
console.log(null===JSON.parse(JSON.stringify(null)));

console.log(111===JSON.parse(JSON.stringify(111)));
console.log("111"===JSON.parse(JSON.stringify("111")));
console.log(true===JSON.parse(JSON.stringify(true)));

console.log(undefined===JSON.parse(JSON.stringify(function(){})));
```

Number

Properties

`MAX_VALUE`

`MIN_VALUE`

`NEGATIVE_INFINITY`

`POSITIVE_INFINITY`

For information on using the above properties, refer to the ES5 standard.

Date

Properties

`parse`

`UTC`

now

For information on using the above properties, refer to the ES5 standard.

Global

Properties

`NaN`

©2013-2022 Tencent Cloud. All rights reserved.

`Infinity`
`undefined`

For information on using the above properties, refer to the ES5 standard.

Methods

`parseInt`
`parseFloat`
`isNaN`
`isFinite`
`decodeURI`
`decodeURIComponent`
`encodeURI`
`encodeURIComponent`

For information on using the above properties, refer to the ES5 standard.

©2013-2022 Tencent Cloud. All rights reserved.

Components

Component Overview

Last updated：2024-07-12 18:23:32

View Container

Name	Feature Description	Minimum Version
scroll-view	Scrollable View Region	1.5.0
swiper	Slider View Container	1.5.0
swiper-item	Can only be placed within the Swiper component, with width and height automatically set to 100%.	1.5.0
view	View Container	1.5.0
movable-area	Movable Area of the Movable-View	1.5.0
movable-view	A movable view container that can be dragged and slid across the page.	1.5.0
cover-image	An image view overlaying the native components.	1.5.0
cover-view	A text view overlaying the native components.	1.5.0

Fundamental Content

Name	Feature Description	Minimum Version
icon	Icon Component	1.5.0
progress	Progress Bar	1.5.0
text	Text	1.5.0
rich-text	rich text	1.5.0

©2013-2022 Tencent Cloud. All rights reserved.

Form Component

Name	Feature Description	Minimum Version
button	Button	1.5.0
checkbox	Multiple Selection Items	1.5.0
checkbox-group	Multi-item selector, internally composed of multiple checkboxes.	1.5.0
form	Form	1.5.0
input	Input Box	1.5.0
label	Enhancing the usability of form components.	1.5.0
picker	A scrolling selector that pops up from the bottom.	1.5.0
radio	Single selection item.	1.5.0
radio-group	Single-item selector, composed of multiple radio buttons.	1.5.0
slider	Sliding Selector	1.5.0
switch	Toggle Selector	1.5.0
textarea	Multi-line input field.	1.5.0

Navigation

Name	Feature Description	Minimum Version
navigator	Page Link	1.5.0

Media Component

Name	Feature Description	
image	Image	1.5.0

©2013-2022 Tencent Cloud. All rights reserved.

video	Video	1.5.0
animation-video	Transparent Video	1.5.0
camera	System Camera	1.5.0
live-player	Real-Time Audio-Video Playback	1.5.0
live-pusher	Real-Time Audio-Video Recording	1.5.0

Map

Name	Feature Description	Minimum Version
map	Map	1.5.0

canvas

Name	Feature Description	Minimum Version
Canvas	canvas	1.5.0

Open Capabilities

Name	Feature Description	Minimum Version
web-view	Container for Hosting Web Pages	1.5.0

©2013-2022 Tencent Cloud. All rights reserved.

Native Components

Last updated：2024-07-12 18:23:06

Native Components

Within the development framework of TCMPP, certain components are native, created by the client-side. These native components, implemented by the Mini Program client, boast superior performance and a richer array of features.
Native Components include:
camera: A camera component, utilized for invoking the camera to capture photographs or record videos.
Canvas: A canvas component, employed for the creation of graphics, animations, and the like.
input (Manifests as a native component only when in focus)
textarea: A multiline input box component, designed for the input of multiline text.
video: A video component, designed for video playback.
live-player: A live streaming player component, designed for the playback of live streams.
map: A map component, utilized for the display of maps along with markers, routes, and other features on the map.

Usage restrictions for the native components

Given that native components exist outside of the WebView rendering process, the following limitations apply when using them:
The hierarchy of native components is of the `highest` order. Therefore, regardless of the `z-index` set for other components on the page, they cannot overlay the native components. Subsequently inserted native components can overlay the previously inserted ones.
Native components cannot be used within the `<picker-view>` .
Certain CSS styles cannot be applied to native components, for example:
CSS animations cannot be applied to native components.
Defining native components as `position: fixed` is not possible.
The display area of native components cannot be cropped by using `overflow: hidden` on parent nodes.
Event listeners for native components do not support the `bind:eventname` syntax, only `bindeventname` is supported. Native components also do not support event binding methods such as `catch` and `capture` .
Native components will obscure the debug panel popped up by vConsole.
Note:
On the tool, native components are simulated with web components. Hence, in many cases, they cannot accurately replicate the performance on a real device. It is recommended that developers debug on real devices whenever possible when using native components.

©2013-2022 Tencent Cloud. All rights reserved.

©2013-2022 Tencent Cloud. All rights reserved.

View Container

Last updated：2024-07-12 18:20:58

scroll-view

Feature Description: A scrollable view area. When using vertical scrolling, it is necessary to assign a fixed height to the `<scroll-view>` through the application of WXSS height settings.

Parameter and Description:

Attribute name	Type	Default value	Description
scroll-x	boolean	false	Allows horizontal scrolling
scroll-y	boolean	false	Allows vertical scrolling
upper-threshold	number / string	50	Distance from the top/left at which the scrolltoupper is triggered
lower-threshold	number/tring	50	Distance from the bottom/right at which the scrolltolower is triggered.
scroll-top	number / string	-	Set the position of the vertical scrollbar
scroll-left	number / string	-	Set the position of the horizontal scrollbar
scroll-into-view	string	-	The value should be the id of a certain sub-element (the id cannot start with a number). Set which direction can be scrolled, then scroll to that element in that direction.
scroll-with-animation	boolean	false	Use animation transition when setting the scrollbar position
enable-back-to-top	boolean	false	When the top status bar is clicked on iOS, or the title bar is double-clicked on Android, the scrollbar returns to the top. This only applies to the vertical orientation.
bindscrolltoupper	eventhandle	-	Triggered when scrolling to the top/left side
bindscrolltolower	eventhandle	-	Triggered when scrolling to the bottom/right side
bindscroll	eventhandle	-	Triggered during scrolling, event.detail = {scrollLeft, scrollTop, scrollHeight, scrollWidth,

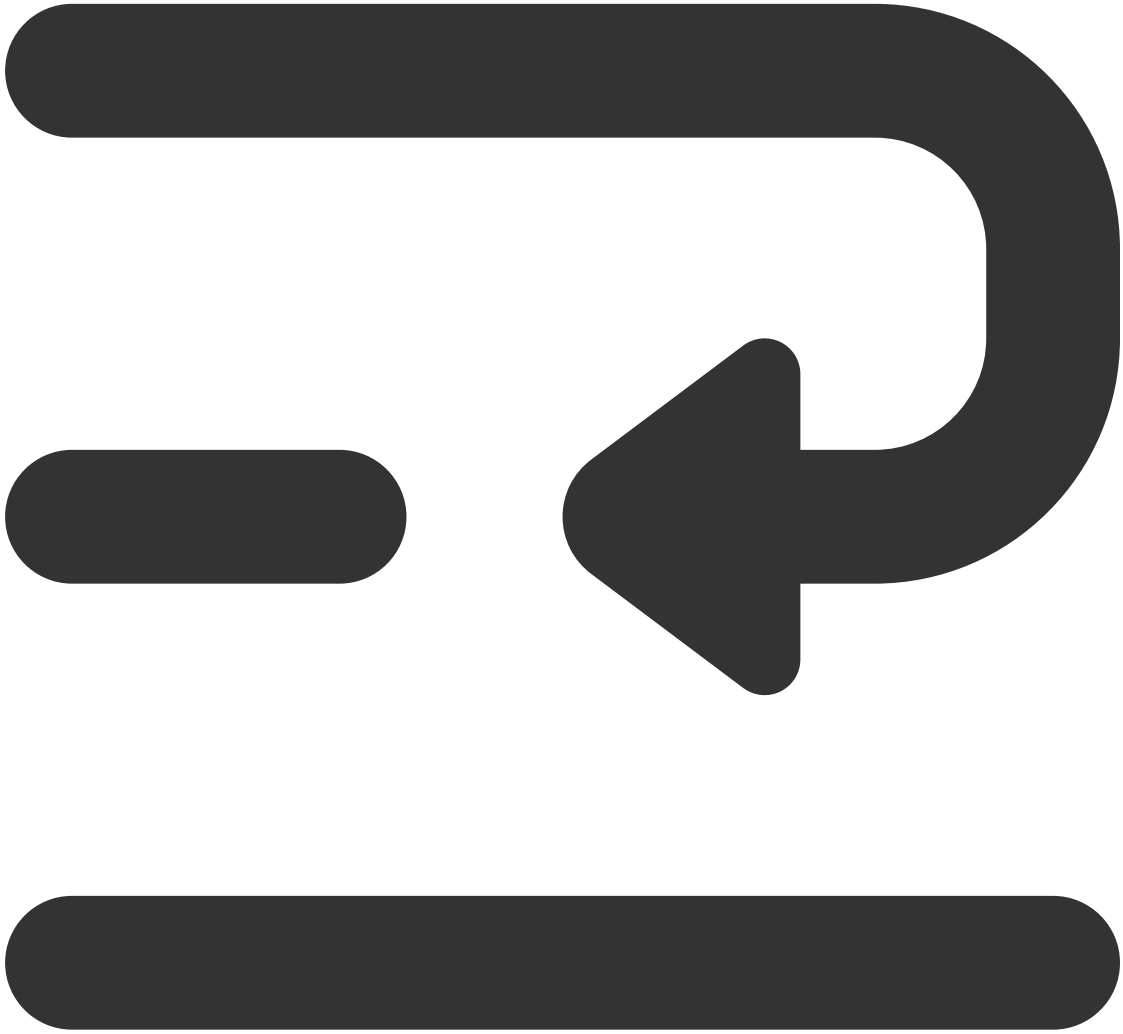
©2013-2022 Tencent Cloud. All rights reserved.

			deltaX, deltaY}
aria-label	string	false	-
enhanced	boolean	true	-
bounces	boolean	true	-
show-scrollbar	boolean	false	-
paging-enabled	boolean	false	-
fast-deceleration	boolean	false	-

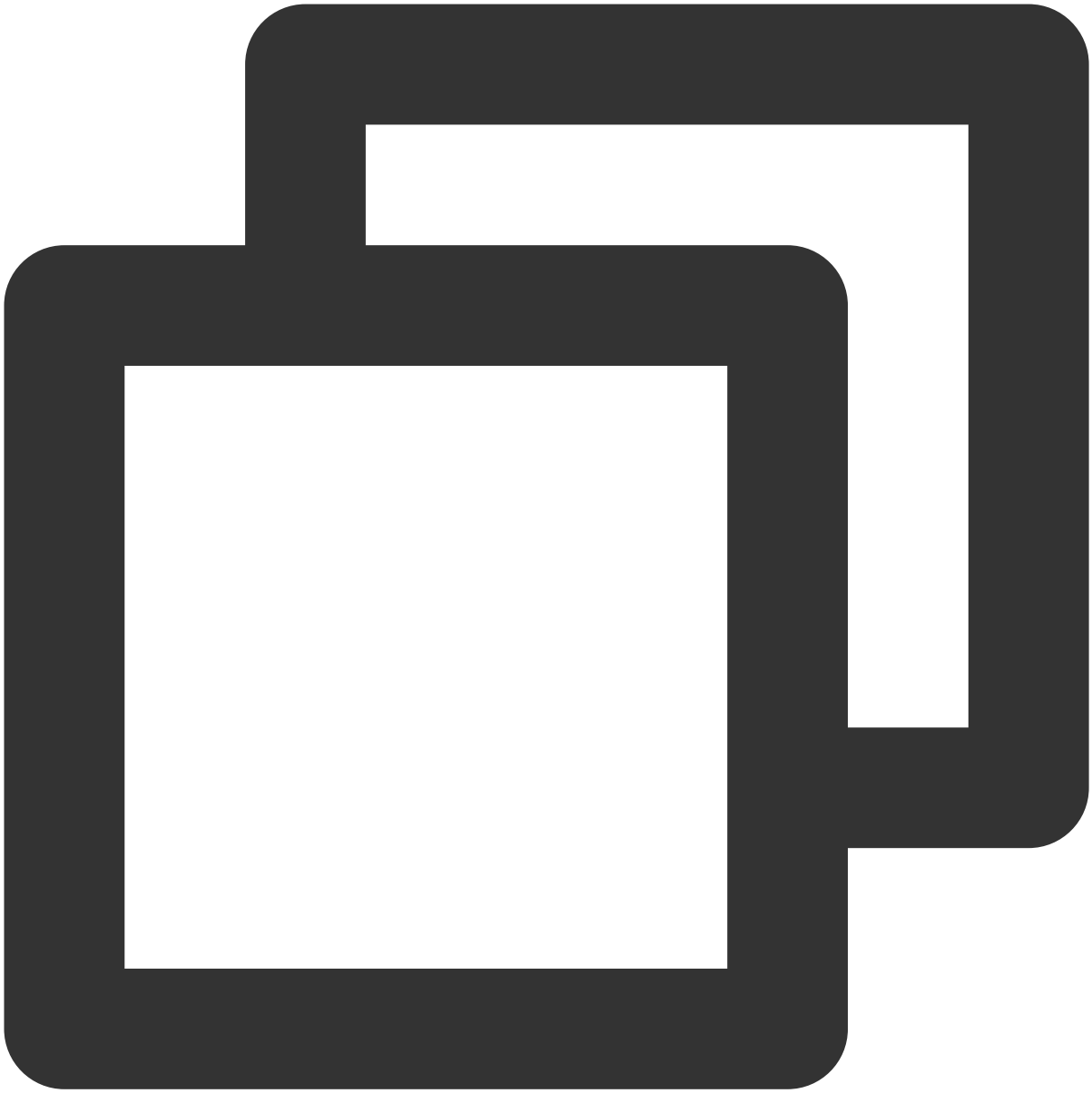
Sample Code

Corresponding WXML file

©2013-2022 Tencent Cloud. All rights reserved.



©2013-2022 Tencent Cloud. All rights reserved.



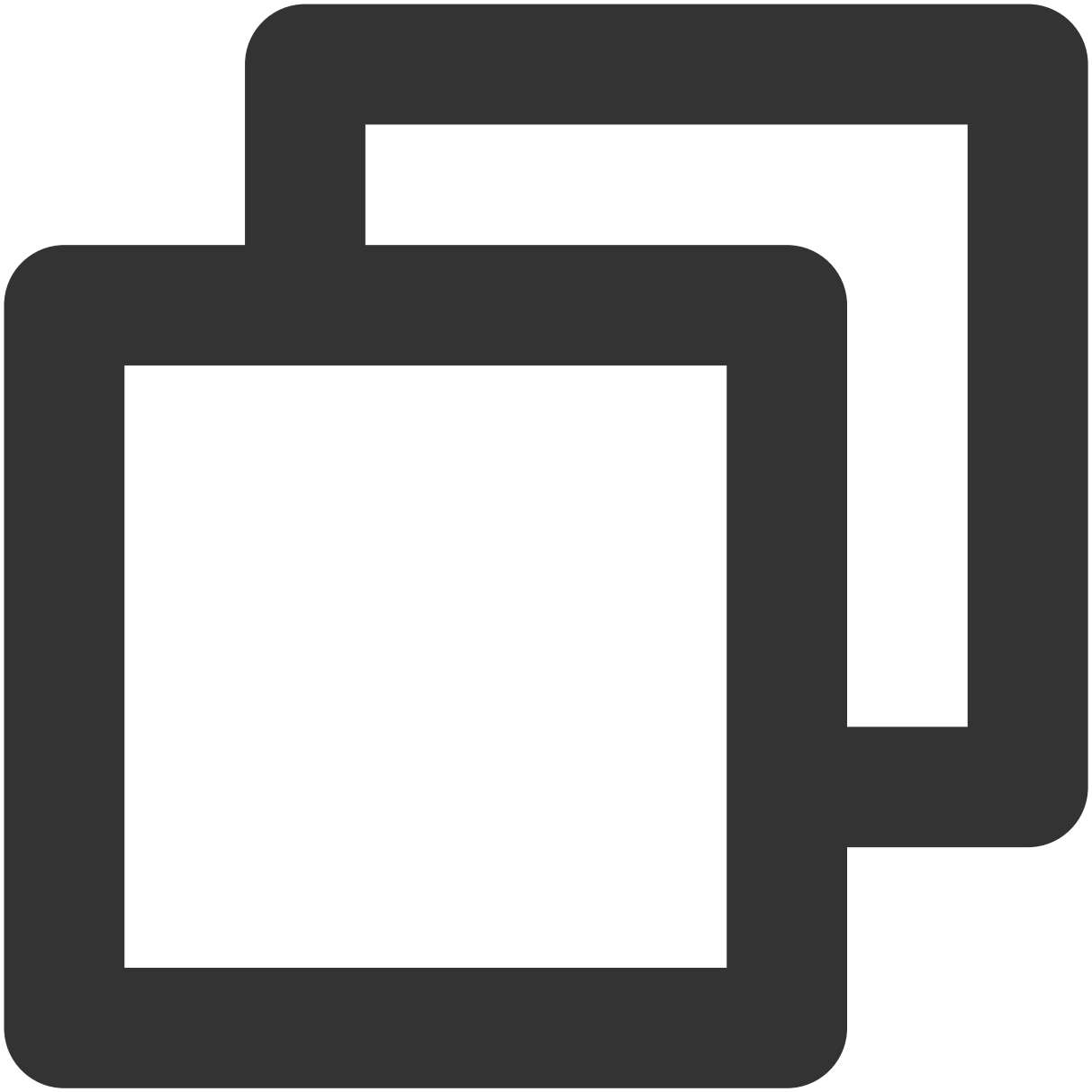
```
<view class="section">
  <view class="section__title">vertical scroll</view>
  <view class="section__title">Vertical Scrolling</view>
  <scroll-view
    scroll-y
    style="height: 200px;"
    bindscrolltoupper="upper"
    bindscrolltolower="lower"
    bindscroll="scroll"
    scroll-into-view="{{toView}}"
    scroll-top="{{scrollTop}}"
```

©2013-2022 Tencent Cloud. All rights reserved.

```
    >
      <view id="green" class="scroll-view-item bc_cyanblue">A</view>
      <view id="red" class="scroll-view-item bc_blue">B</view>
    </scroll-view>
  </view>
  <view class="section section_gap">
    <view class="section__title">horizontal scroll</view>
    <view class="section__title">Horizontal Scrolling</view>
    <scroll-view class="scroll-view_H" scroll-x style="width: 100%">
      <view id="green" class="scroll-view-item_H bc_cyanblue"></view>
      <view id="red" class="scroll-view-item_H bc_blue"></view>
    </scroll-view>
  </view>
```

Corresponding js file

©2013-2022 Tencent Cloud. All rights reserved.

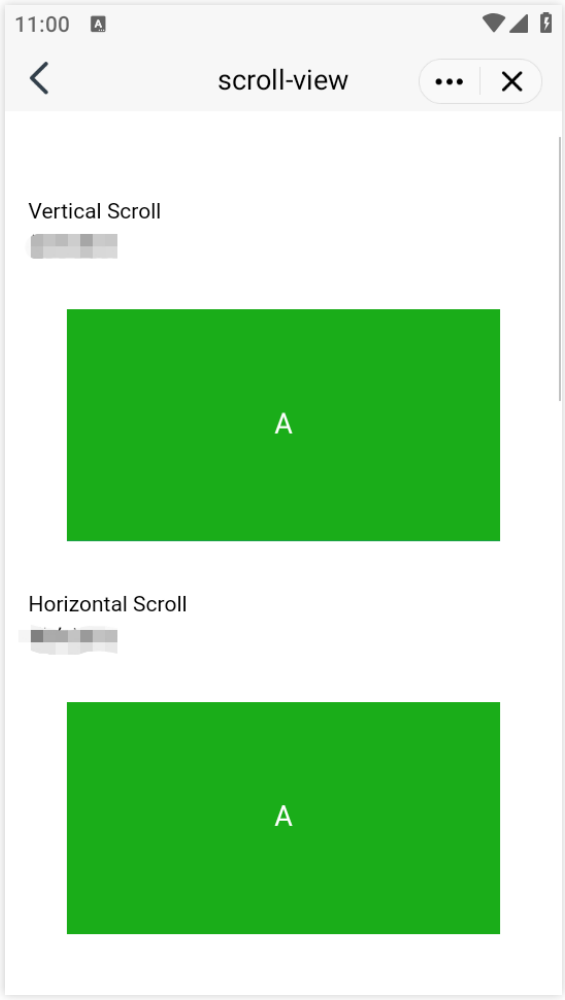


```
const order = ['red', 'yellow', 'blue', 'green', 'red']
Page({
  data: {
    toView: 'red',
    scrollTop: 100
  },
  upper(e) {
    console.log(e)
  },
  lower(e) {
    console.log(e)
```

©2013-2022 Tencent Cloud. All rights reserved.

```
  },
  scroll(e) {
    console.log(e)
  },
  tap(e) {
    for (let i = 0; i < order.length; ++i) {
      if (order[i] === this.data.toView) {
        this.setData({
          toView: order[i + 1]
        })
        break
      }
    }
  },
  tapMove(e) {
    this.setData({
      scrollTop: this.data.scrollTop + 10
    })
  }
})
```

©2013-2022 Tencent Cloud. All rights reserved.



Note:

Refrain from using textarea, map, canvas, video components within the scroll-view.

The priority of scroll-into-view supersedes that of scroll-top.

Scrolling within the scroll-view will prevent page bounce-back. Hence, it is impossible to trigger onPullDownRefresh while scrolling in the scroll-view.

To use the pull-down refresh feature, employ the page's scrolling instead of the scroll-view. This method also allows returning to the top of the page by clicking on the status bar.

swiper

Feature Description: Slider view container. The `change` event returns a `detail` that includes a `source` field, indicating the cause of the change. Possible values are as follows:

`autoplay` triggers automatic playback, causing changes in the swiper.

`touch` User sliding causes changes in the swiper.

Any other reasons will be represented by an empty string.

Note:

It can only accommodate the `<swiper-item/>` component; otherwise, it may lead to undefined behavior.

©2013-2022 Tencent Cloud. All rights reserved.

Parameter and Description:

Attribute name	Type	Default value	Description
indicator-dots	boolean	false	Whether to display panel indicator points.
indicator-color	color	rgba(0, 0, 0, .3)	Indicator point color
indicator-active-color	color	#000000	Color of the currently selected indicator point.
autoplay	coolean	false	Whether to switch automatically.
current	number	0	Index of the current slider.
interval	number	5000	Time interval of automatic switching
duration	number	500	Duration of the sliding animation
circular	boolean	false	Whether to adopt seamless sliding
vertical	boolean	false	Is the sliding direction vertical?
previous-margin	string	"0px"	Front margin, which can be used to reveal a small part of the previous item, supporting px and rpx values.
next-margin	string	"0px"	Rear margin, which can be used to reveal a small part of the subsequent item, supporting px and rpx values.
display-multiple-items	number	1	The number of sliders displayed simultaneously.
skip-hidden-item-layout	boolean	false	Whether to skip the layout of sliders not displayed. Setting this to true can optimize sliding performance in complex situations, but will lose layout information of sliders in hidden status.
bindchange	eventhandle	-	The change event will be triggered when 'current' changes, with event.detail = {current: current, source: source}.
bindtransition	eventhandle	-	The transition event will be triggered when the position of the swiper-item changes, with event.detail = {dx: dx, dy: dy}.
bindanimationfinish	eventhandle	-	The animationfinish event will be triggered

©2013-2022 Tencent Cloud. All rights reserved.

			when the animation ends, with event.detail as previously described.

swiper-item

Feature Description: Can only be placed within the `<swiper>` component, with width and height automatically set to 100%.

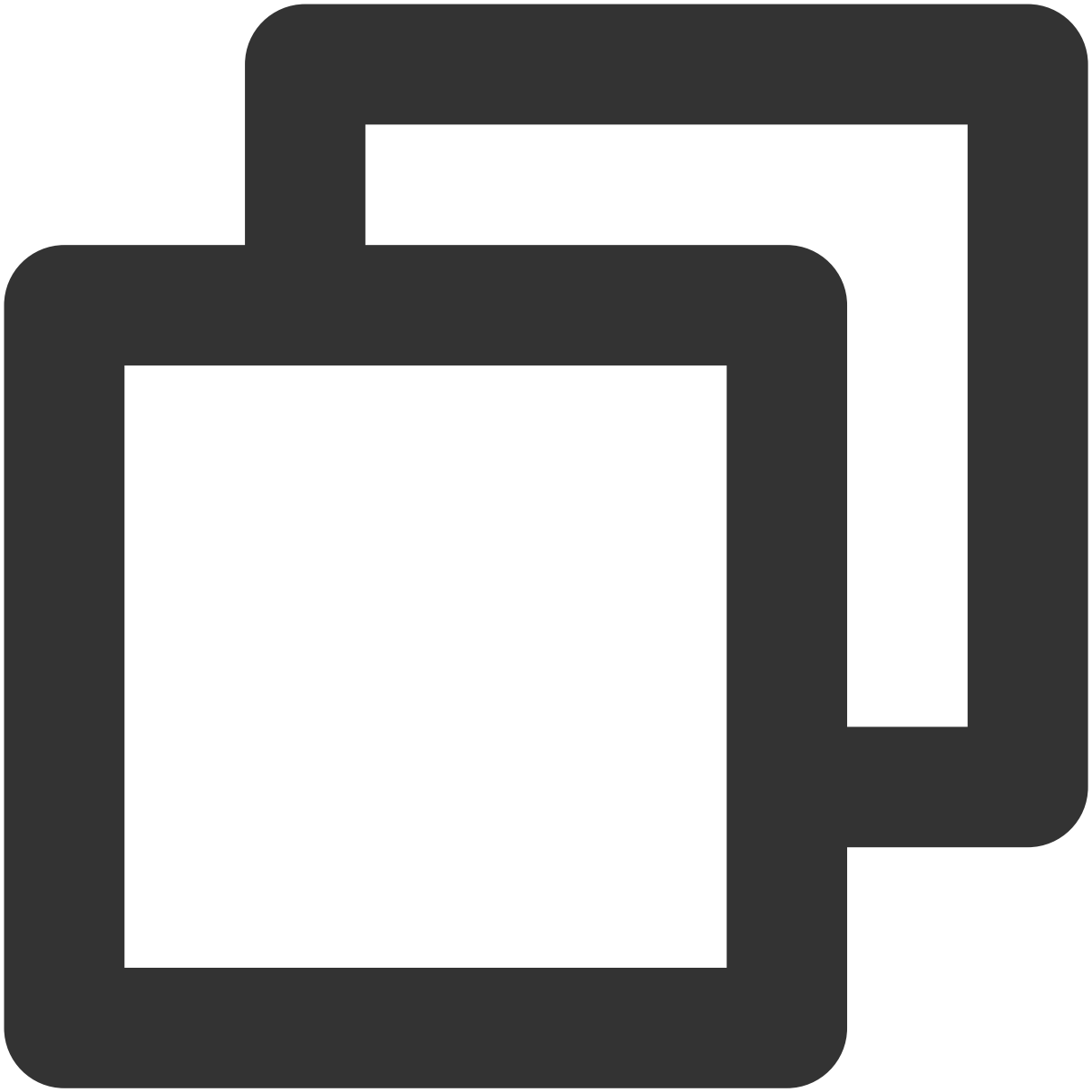
Parameter and Description:

Attribute name	Type	Default value	Description
item-id	string	“”	The identifier of this swiper-item.

Sample Code

Corresponding WXML file

©2013-2022 Tencent Cloud. All rights reserved.

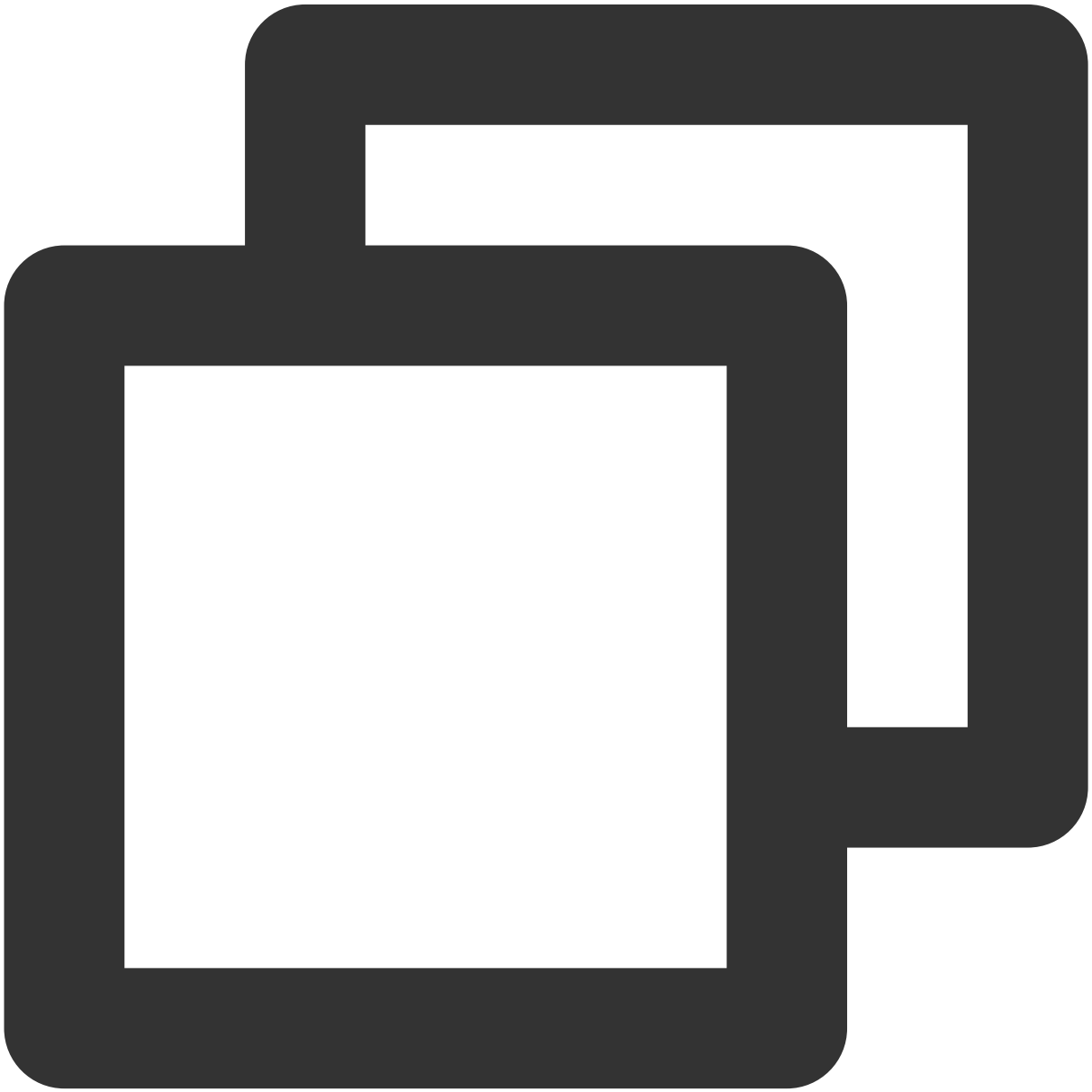


```
<swiper
  indicator-dots="{{indicatorDots}}"
  autoplay="{{autoplay}}"
  interval="{{interval}}"
  duration="{{duration}}"
>
  <block wx.for="{{imgUrls}}">
    <swiper-item>
      <image src="{{item}}" class="slide-image" width="355" height="150" />
    </swiper-item>
  </block>
```

©2013-2022 Tencent Cloud. All rights reserved.

```
</swiper>
<button bindtap="changeIndicatorDots">indicator-dots</button>
<button bindtap="changeAutoplay">autoplay</button>
<slider bindchange="intervalChange" show-value min="500" max="2000" />
interval
<slider bindchange="durationChange" show-value min="1000" max="10000" />
duration
```

Corresponding js file



```
Page({
  data: {
```

©2013-2022 Tencent Cloud. All rights reserved.

```
    imgUrls: [
      'https://img02.tooopen.com/images/20150928/tooopen_sy_143912755726.jpg',
      'https://img06.tooopen.com/images/20160818/tooopen_sy_175866434296.jpg',
      'https://img06.tooopen.com/images/20160818/tooopen_sy_175833047715.jpg'
    ],
    indicatorDots: false,
    autoplay: false,
    interval: 5000,
    duration: 1000
  },
  changeIndicatorDots(e) {
    this.setData({
      indicatorDots: !this.data.indicatorDots
    })
  },
  changeAutoplay(e) {
    this.setData({
      autoplay: !this.data.autoplay
    })
  },
  intervalChange(e) {
    this.setData({
      interval: e.detail.value
    })
  },
  durationChange(e) {
    this.setData({
      duration: e.detail.value
    })
  }
})
```

view

Feature Description: A view container.

Parameter and Description:

Attribute name	Type	Default value	Description
hover-class	string	none	Specifies the style class when pressed. When `hover-class="none"` , there is no click status effect.
hover-stop-	boolean	false	Specifies whether to prevent the ancestor nodes of

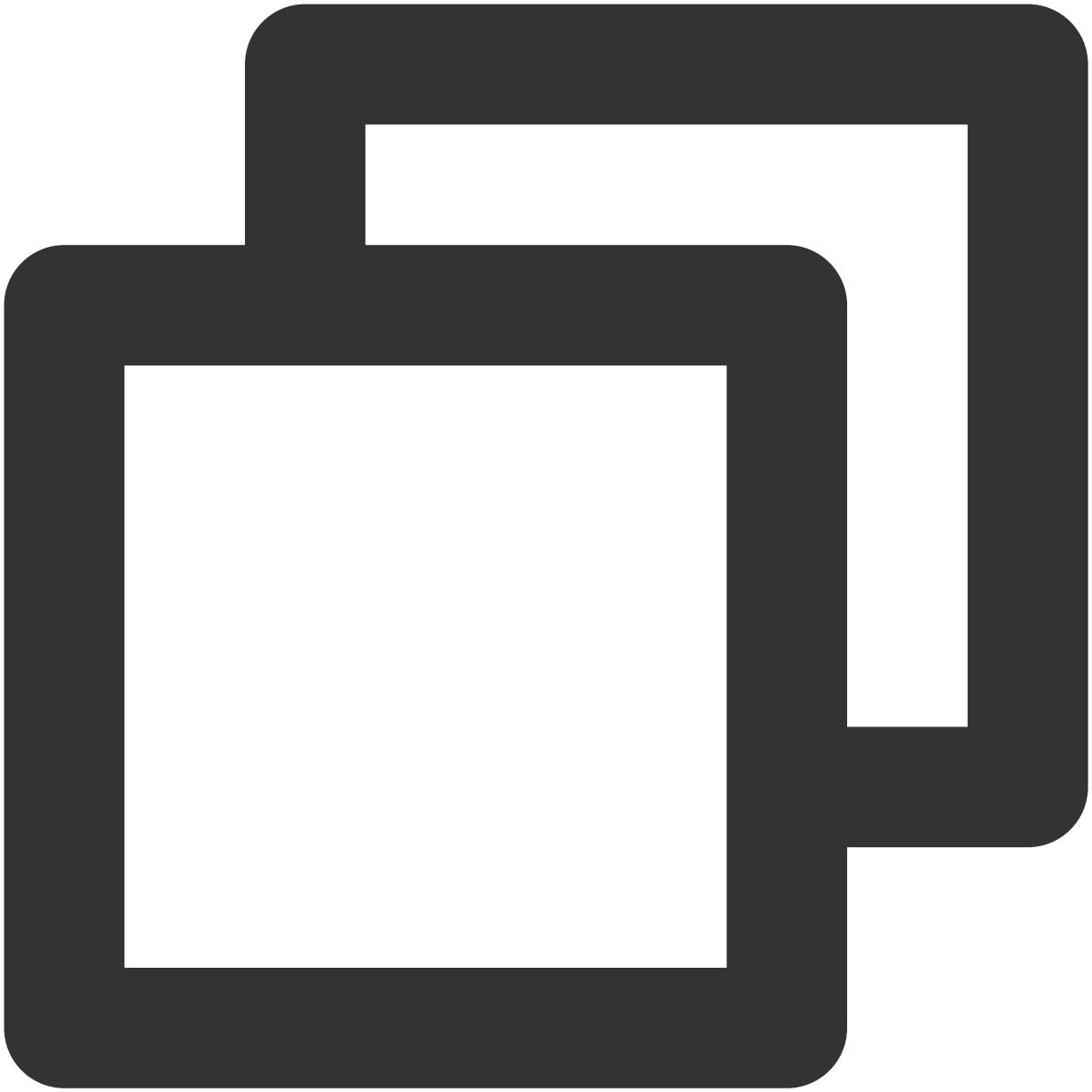
©2013-2022 Tencent Cloud. All rights reserved.

propagation			this node from appearing in the click status.
hover-start-time	number	50	The duration before the click status appears after pressing, measured in milliseconds.
hover-start-time	number	400	The retention time of the click status after the finger is released, measured in milliseconds.
aria-role	string	-	Accessibility, (Role) identifies the role of the element.
aria-label	string	-	Accessibility, (Attribute) Additional description of the element.

Sample Code

Corresponding WXML file

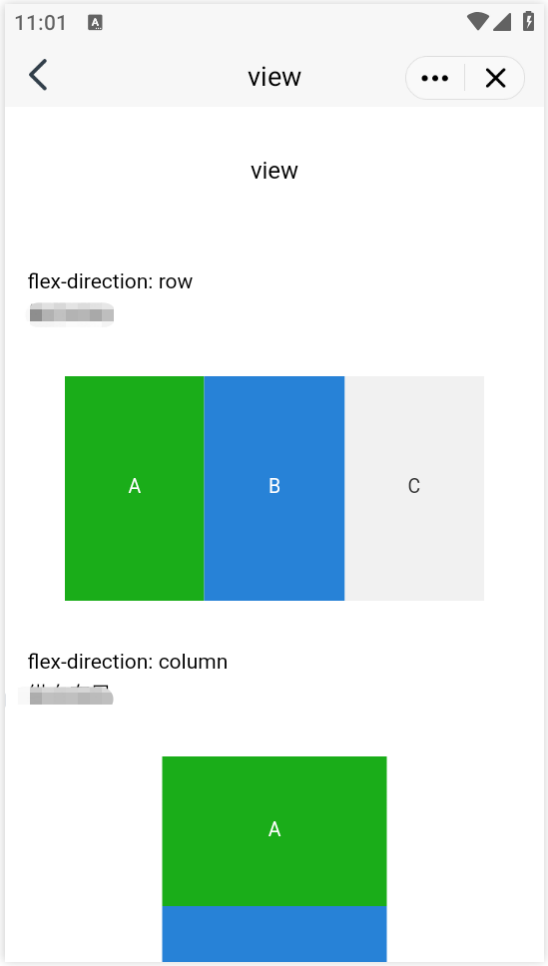
©2013-2022 Tencent Cloud. All rights reserved.



```
<view class="section">
  <view class="section__title">flex-direction: row</view>
  <view class="flex-wrp" style="flex-direction:row;">
    <view class="flex-item bc_cyanblue">A</view>
    <view class="flex-item bc_blue">B</view>
    <view class="flex-item bc_grey">C</view>
  </view>
</view>
<view class="section">
  <view class="section__title">flex-direction: column</view>
  <view class="flex-wrp" style="height: 300px;flex-direction:column;">
```

©2013-2022 Tencent Cloud. All rights reserved.

```
    <view class="flex-item bc_cyanblue">A</view>
    <view class="flex-item bc_blue">B</view>
    <view class="flex-item bc_grey">C</view>
  </view>
</view>
```



Note:

If a scrolling view is required, use scroll-view.

movable-area

Feature Description: The movable area of the movable-view.

Parameter and Description:

Attribute name	Type	Default value	Description
scale-area	boolean	false	When the movable-view within is set to support two-finger scaling, setting this value can modify the effective area of the scaling gesture to the entire movable-area.

©2013-2022 Tencent Cloud. All rights reserved.

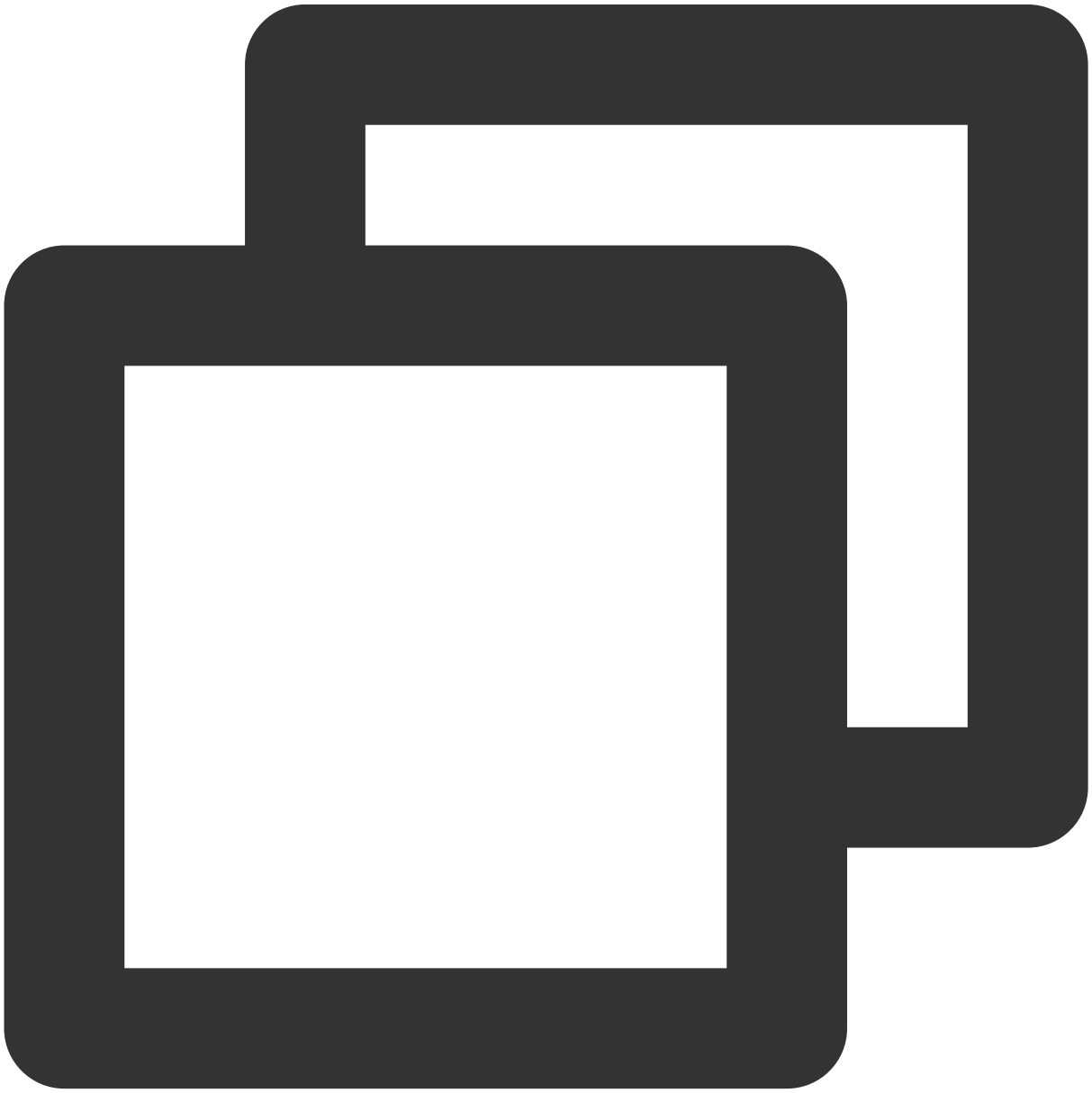
Note:

When the movable-view is smaller than the movable-area, the range of movement for the movable-view is within the movable-area.

When the movable-view is larger than the movable-area, the range of movement for the movable-view must encompass the movable-area, considering the x-axis and y-axis directions separately.

Sample Code

Corresponding WXML file



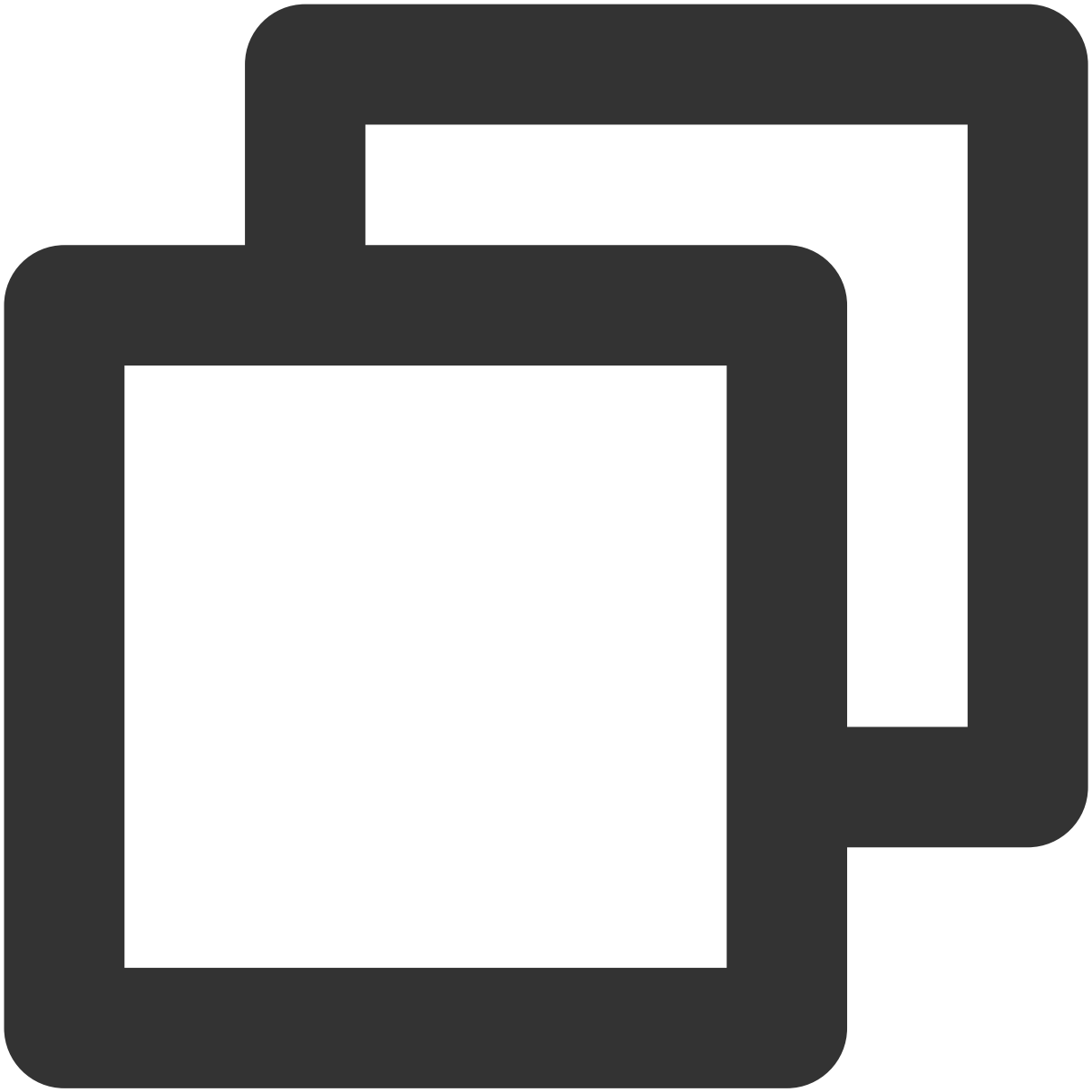
```
<view class="section">
<view class="section__title">When the movable-view area is smaller than the movab
```

©2013-2022 Tencent Cloud. All rights reserved.

```
  <movable-area style="height: 200px; width: 200px; background: red;">
    <movable-view
      style="height: 50px; width: 50px; background: blue;"
      x="{{x}}"
      y="{{y}}"
      direction="all"
    ></movable-view>
  </movable-area>
  <view class="btn-area">
    <button size="mini" bindtap="tap">click me to move to (30px, 30px)</button>
</view>
  <view class="section__title">When the movable-view area is larger than the movabl
  <movable-area style="height: 100px; width: 100px; background: red;">
    <movable-view
      style="height: 200px; width: 200px; background: blue;"
      direction="all"
    ></movable-view>
  </movable-area>
  <view class="section__title">Scalable</view>
  <movable-area
    style="height: 200px; width: 200px; background: red;"
    scale-area
  >
    <movable-view
      style="height: 50px; width: 50px; background: blue;"
      direction="all"
      bindchange="onChange"
      bindscale="onScale"
      scale
      scale-min="0.5"
      scale-max="4"
      scale-value="2"
    ></movable-view>
  </movable-area>
</view>
```

Corresponding js file

©2013-2022 Tencent Cloud. All rights reserved.



```
Page({
  data: {
    x: 0,
    y: 0
  },
  tap(e) {
    this.setData({
      x: 30,
      y: 30
    })
  },
```

©2013-2022 Tencent Cloud. All rights reserved.

```
  onChange(e) {
    console.log(e.detail)
  },
  onScale(e) {
    console.log(e.detail)
  }
})
```

movable-view

Feature Description: A movable view container that can be dragged and slid across the page.

Parameter and Description:

Attribute name	Type	Default value	Description
direction	string	none	The direction of movement for the movable-view, with attribute values including all, vertical, horizontal, and none.
inertia	boolean	false	Whether the movable-view has inertia.
out-of-bounds	boolean	false	Whether the movable-view can still be moved after exceeding the movable area.
x	number / string	-	Defines the offset in the x-axis direction. If the value of x is not within the movable range, it will automatically move to the movable range; changing the value of x will trigger an animation.
y	number / string	-	Defines the offset in the y-axis direction. If the value of y is not within the movable range, it will automatically move to the movable range; changing the value of y will trigger an animation.
damping	number	20	The damping coefficient, used to control the animation when x or y changes and the animation of the boundary rebound. The larger the value, the faster the movement.
friction	number	2	The friction coefficient, used to control the animation of inertial sliding. The larger the value, the greater the friction, and the quicker the sliding stops. The friction coefficient must be greater than 0. Otherwise, it will be set to the default value.

©2013-2022 Tencent Cloud. All rights reserved.

disabled	boolean	false	Whether to disable
scale	boolean	false	Whether to support two-finger zooming. The default effective area for the zoom gesture is within the movable-view.
scale-min	number	0.5	Defines the minimum value of the scaling factor.
scale-max	number	10	Defines the maximum value of the scaling factor.
scale-value	number	1	Defines the scaling factor, with a value range of 0.5 - 10.
animation	boolean	true	Defines the scaling factor, with a value range of 0.5 - 10.
bindchange	eventhandle	-	Event triggered during the dragging process, event.detail = {x: x, y: y, source: source}, where source indicates the cause of the movement. The value can be touch (dragging), touch-out-of-bounds (exceeding the moving range), out-of-bounds (bounce back after exceeding the moving range), friction (inertia), and an empty string (setData).
bindscale	eventhandle	-	Event triggered during the scaling process, event.detail = {x: x, y: y, scale: scale}.

In addition to basic events, movable-view offers two special events.

Type	Trigger condition
htouchmove	The initial movement after the finger touches is horizontal. If this event is caught, it implies that the touchmove event has also been caught.
vtouchmove	The initial movement after the finger touches is vertical. If this event is caught, it implies that the touchmove event has also been caught.

Note:

The movable-view must have the width and height attributes set. Otherwise, they will default to 10 px.

By default, movable-view is absolutely positioned, with top and left attributes set to 0 px.

The movable-view must be within the <movable-area/> component and must be a direct child node. Otherwise, it will not be movable.

©2013-2022 Tencent Cloud. All rights reserved.

cover-image

Feature Description: An image view that overlays native components. The native components that can be overlaid are the same as cover-view, and can be nested within cover-view.

Parameter and Description:

Attribute name	Type	Default value	Description
src	string	-	Icon path, supporting temporary paths, network addresses, and cloud file IDs. Base64 format is currently not supported.
bindload	eventhandle	-	Triggered when the image fails to load.
binderror	eventhandle	-	Triggered when the image fails to load.
aria-role	string	-	Accessibility, (Role) Identifies the function of the element.
aria-lable	string	-	Accessibility, (Role) Identifies the function of the element.

Note:

The aria-role of <cover-view> and <cover-image>can only be set to button, which allows for clicking and reading out "button" in screen reading mode; when empty, it can be focused but not clicked.

The event model follows the bubbling model, but it does not bubble up to native components.

It is recommended to enclose all text within the cover-view tag to prevent layout errors.

Only basic positioning, layout, and text styles are supported. Setting individual border, background-image, shadow, overflow: visible, etc., are not supported.

It is advised that child nodes should not overflow their parent nodes.

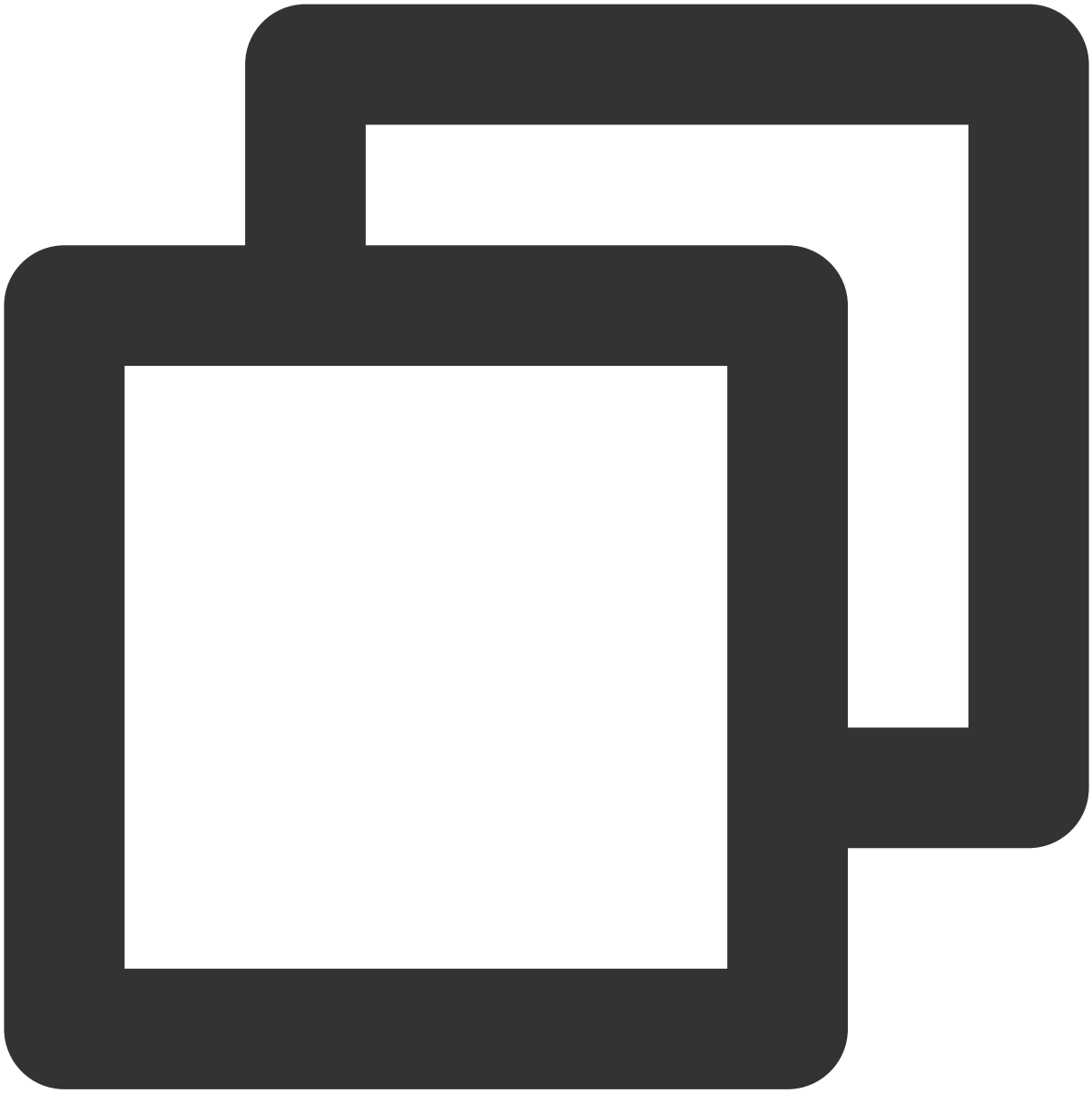
The default style settings are: white-space: nowrap; line-height: 1.2; display: block.

When custom components are nested within a cover-view, the custom component's slot and its parent node do not currently support visibility control through wx:if, as this may result in the cover-view not being displayed.

Sample Code

Corresponding WXML file

©2013-2022 Tencent Cloud. All rights reserved.

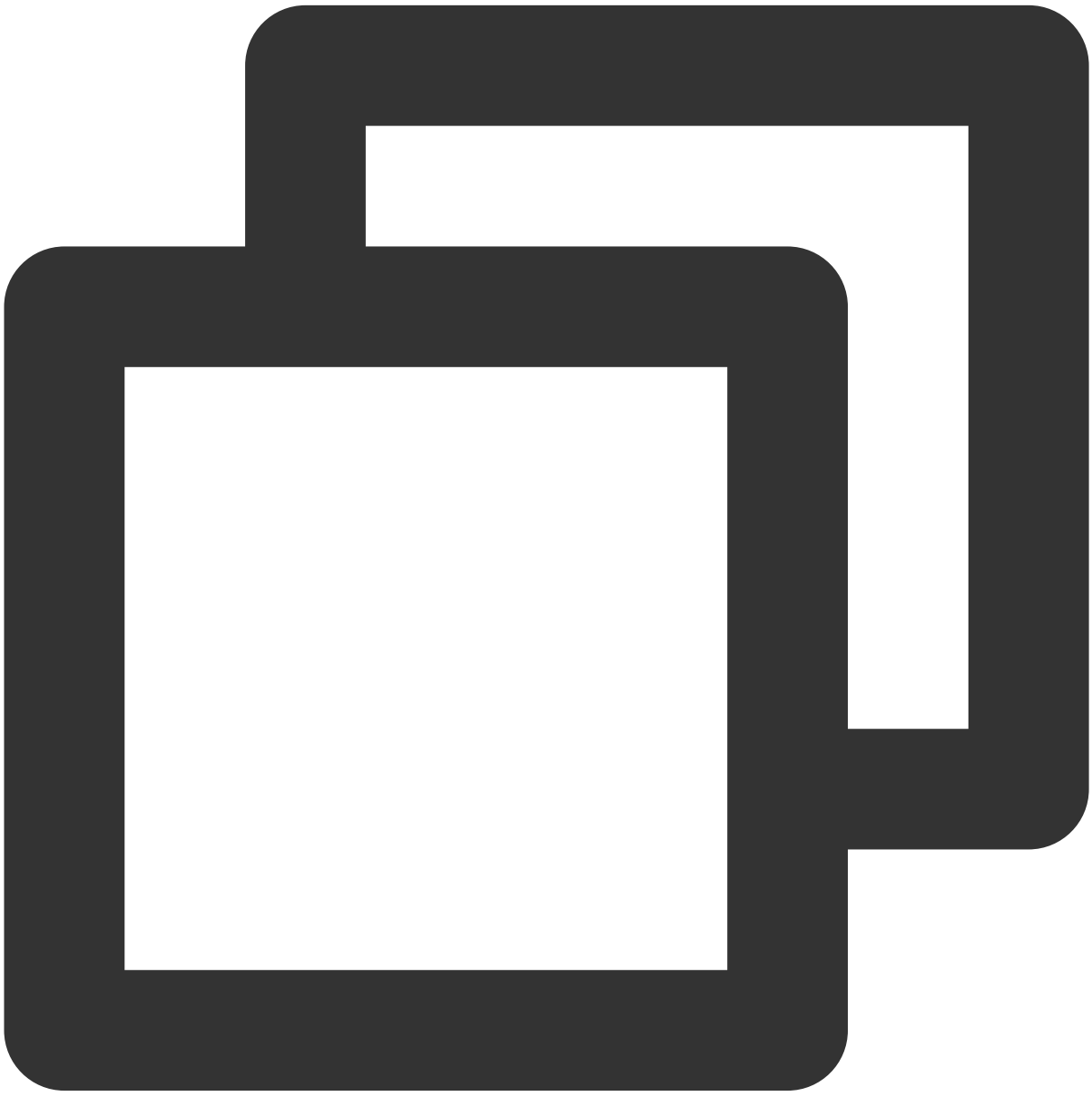


```
<video
  id="myVideo"
  src="https://qzonestyle.gtimg.cn/qzone/qzact/act/external/qq-video/qq-video.mp4"
  controls="{{false}}"
  event-model="bubble"
>
  <cover-view class="controls">
    <cover-view class="play" bindtap="play">
      <cover-image class="img" src="/path/to/icon_play" />
    </cover-view>
    <cover-view class="pause" bindtap="pause">
```

©2013-2022 Tencent Cloud. All rights reserved.

```
      <cover-image class="img" src="/path/to/icon_pause" />
    </cover-view>
    <cover-view class="time">00:00</cover-view>
  </cover-view>
</video>
```

Corresponding WXSS file



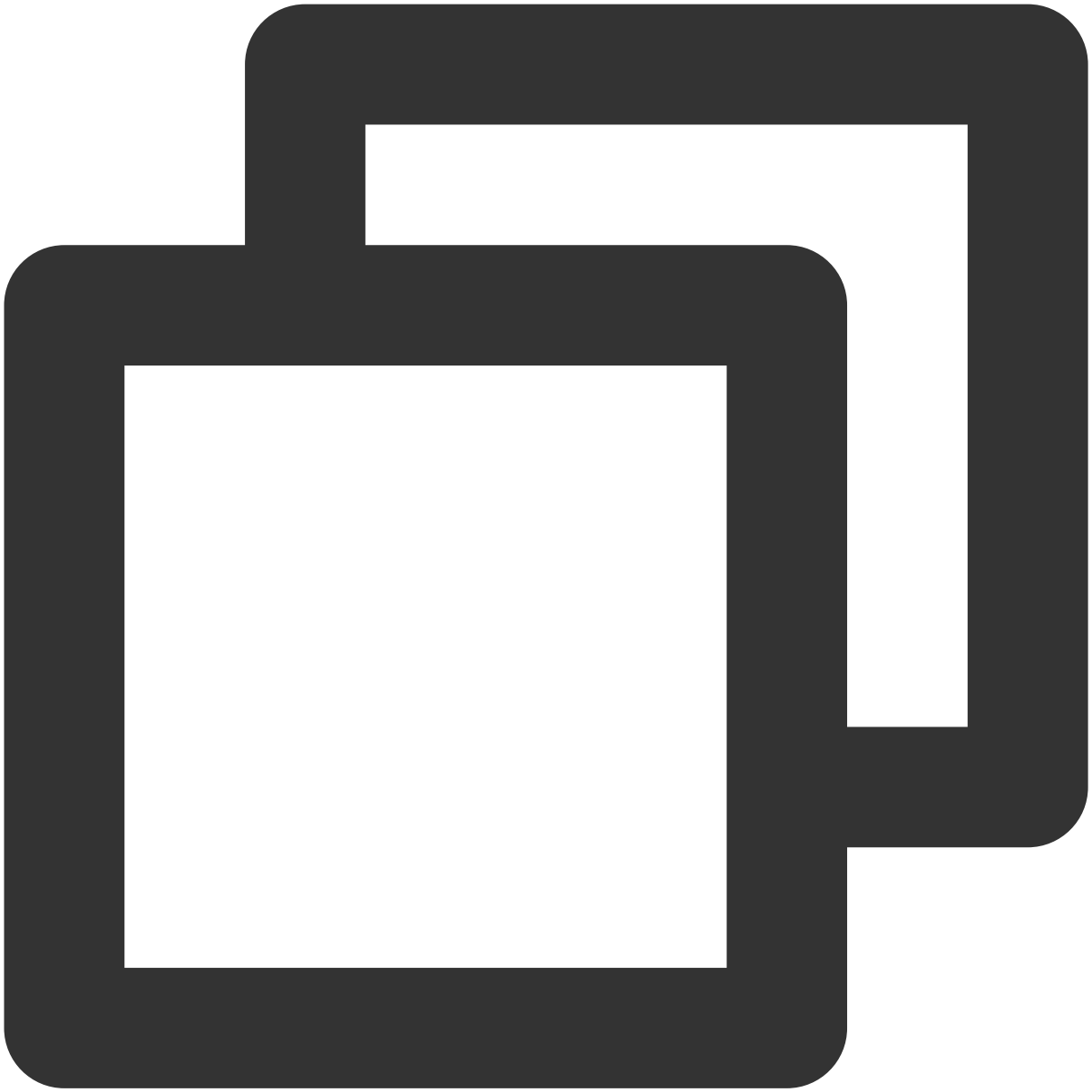
```
.controls {
  position: relative;
  top: 50%;
  height: 50px;
```

©2013-2022 Tencent Cloud. All rights reserved.

```
  margin-top: -25px;
  display: flex;
}
.play,
.pause,
.time {
  flex: 1;
  height: 100%;
}
.time {
  text-align: center;
  background-color: rgba(0, 0, 0, 0.5);
  color: white;
  line-height: 50px;
}
.img {
  width: 40px;
  height: 40px;
  margin: 5px auto;
}
```

Corresponding js file

©2013-2022 Tencent Cloud. All rights reserved.



```
Page({
  onReady() {
    this.videoCtx = wx.createVideoContext('myVideo')
  },
  play() {
    this.videoCtx.play()
  },
  pause() {
    this.videoCtx.pause()
  }
})
```

©2013-2022 Tencent Cloud. All rights reserved.

cover-view

Function Description: A text view that overlays native components, including `map`, `video`, `canvas`, `camera`, `live-player`, `live-pusher`. It only supports nesting `cover-view`, `cover-image`, and `button` can be used within `cover-view`.

Parameter and Description:

Attribute name	Type	Default value	Description
scroll-top	number / string	-	Set the top scroll offset, which only takes effect after overflow-y: scroll is set to become a scrolling element.
aria-role	string	-	Accessibility, (Role) Identifies the function of the element.
aria-label	string	-	Accessibility, (Attribute) Additional description of the element.

©2013-2022 Tencent Cloud. All rights reserved.

Fundamentals

Last updated：2024-07-12 18:18:10

icon

Feature Description: Icon Component.

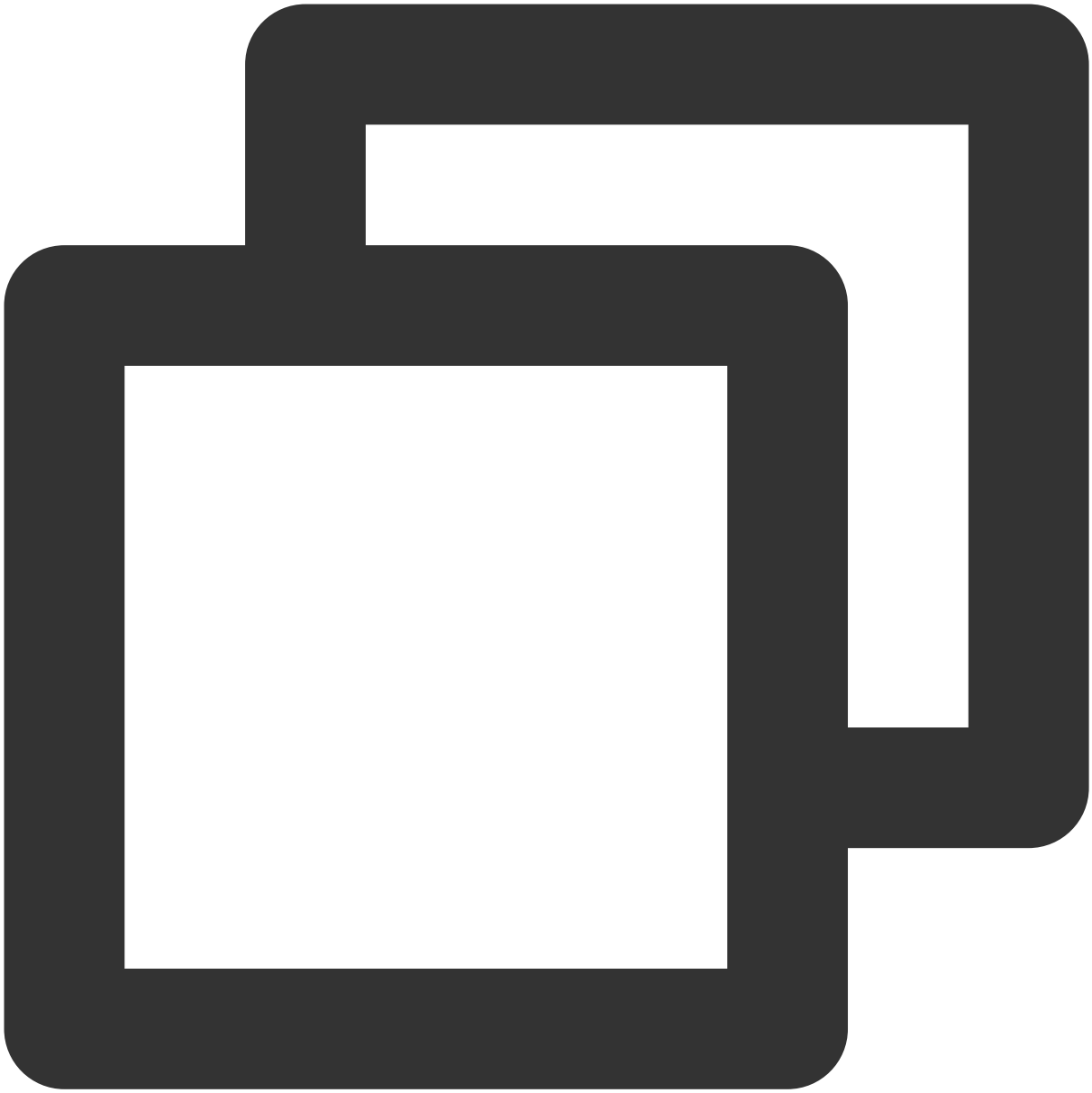
Parameter and Description:

Attribute	Type	Default value	Description
type	string	-	The type of icon, whose valid values are: success, success_no_circle, info, warn, waiting, cancel, download, search, clear.
size	umber / string	23	The size of the icon, with the default unit in pixels.
color	string	-	The color of the icon, same as the CSS color.
aria-label	string	-	Accessibility, (Attribute) Additional description of the element.

Sample Code

Corresponding WXML file

©2013-2022 Tencent Cloud. All rights reserved.



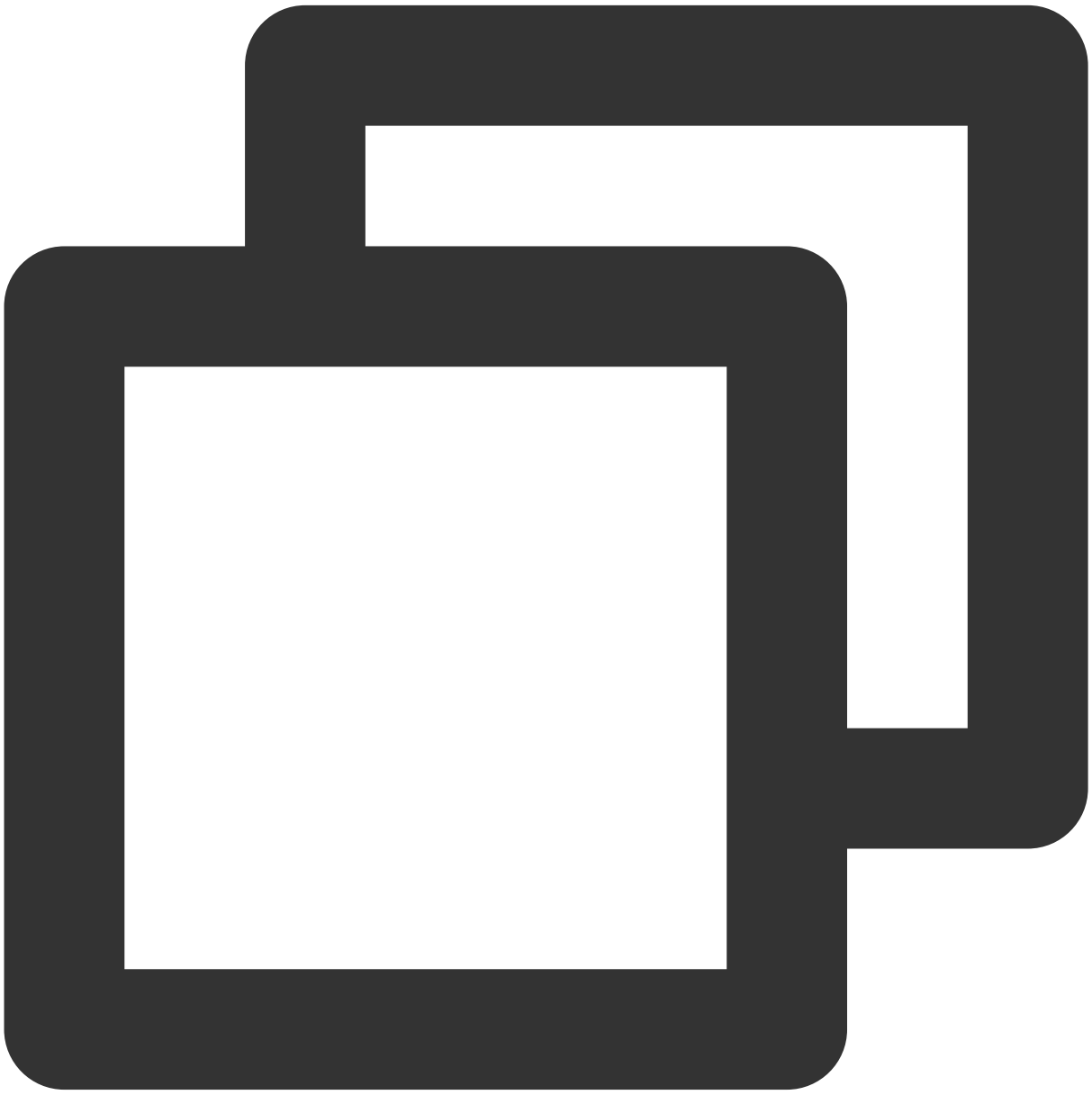
```
<view class="group">
  <block wx.for="{{iconSize}}">
    <icon type="success" size="{{item}}" />
  </block>
</view>

<view class="group">
  <block wx.for="{{iconType}}">
    <icon type="{{item}}" size="40" />
  </block>
</view>
```

©2013-2022 Tencent Cloud. All rights reserved.

```
<view class="group">
  <block wx.for="{{iconColor}}">
    <icon type="success" size="40" color="{{item}}" />
  </block>
</view>
```

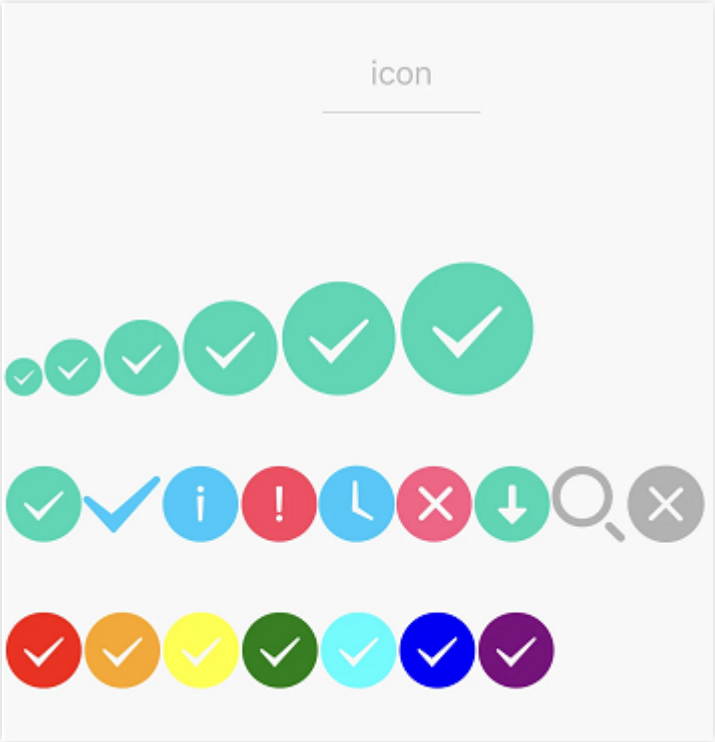
Corresponding js file



```
Page({
  data: {
    iconSize: [20, 30, 40, 50, 60, 70],
```

©2013-2022 Tencent Cloud. All rights reserved.

```
    iconColor: [
      'red', 'orange', 'yellow', 'green', 'rgb(0,255,255)', 'blue', 'purple'
    ],
    iconType: [
      'success', 'success_no_circle', 'info', 'warn', 'waiting', 'cancel', 'downloa
    ]
  }
})
```



progress

Feature Description: Progress Bar.

Parameter and Description:

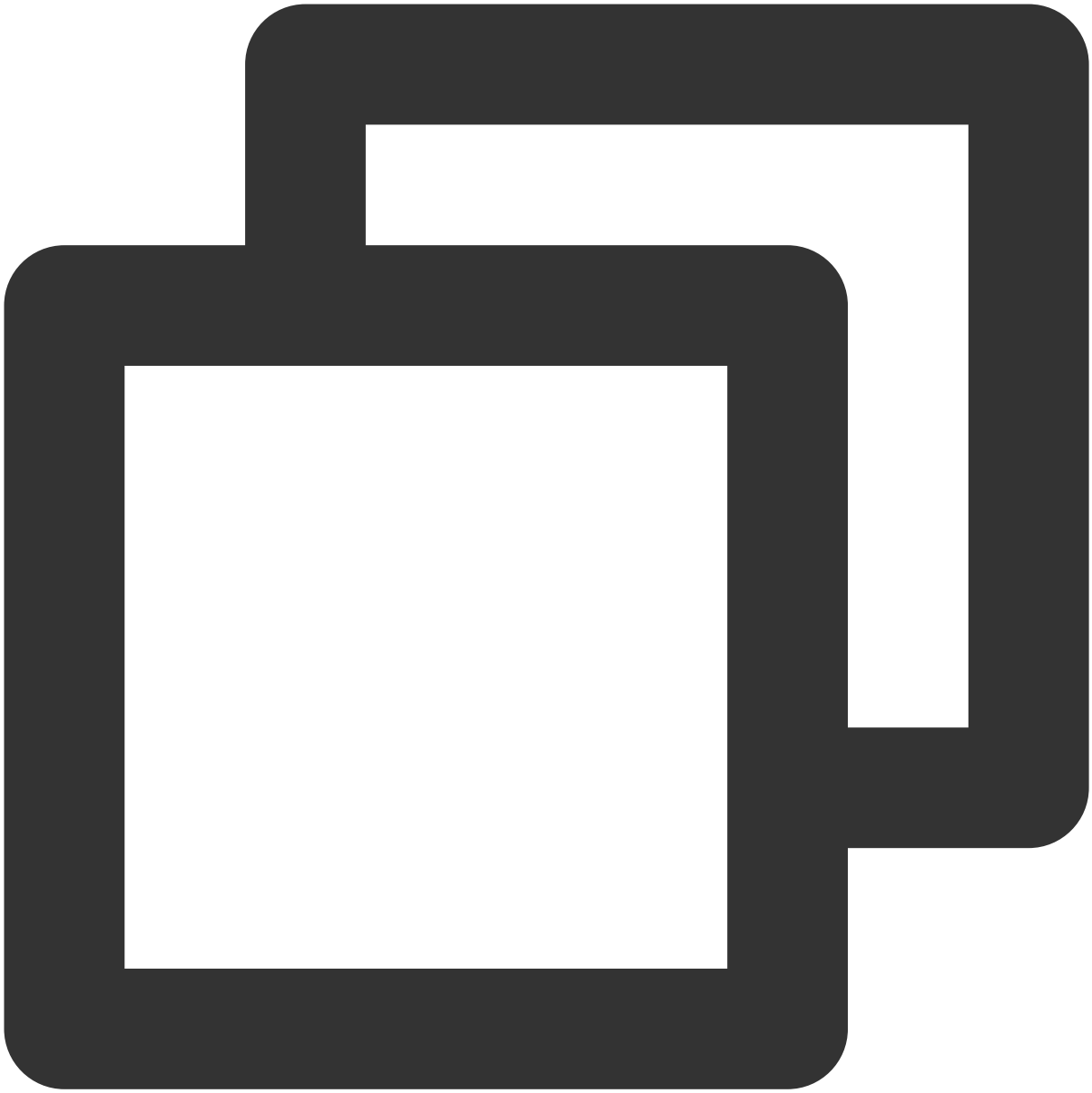
Attribute	Type	Default value	Description
percent	float	-	Percentage range: 0-100
show-info	boolean	false	Display percentage on the right side of the progress bar
border-radius	number / string	0	Chamfer size, unit: px
font-size	number / string	16	Font size of the percentage on the right, unit: px

©2013-2022 Tencent Cloud. All rights reserved.

stroke-width	number / string	6	Width of the progress bar line
color	color	#09BB07	Progress bar color (use activeColor)
activeColor	color	-	Color of the selected progress bar
backgroundColor	color	-	Color of the unselected progress bar
active	boolean	false	Animation of the progress bar moving from left to right
active-mode	string	backwards	Backwards: Animation plays from the beginning Forwards: The animation continues from the point where it last ended.
bindactiveend	eventhandle	-	Animation completion event
aria-label	string	-	Accessibility, (Attribute) Additional description of the element.

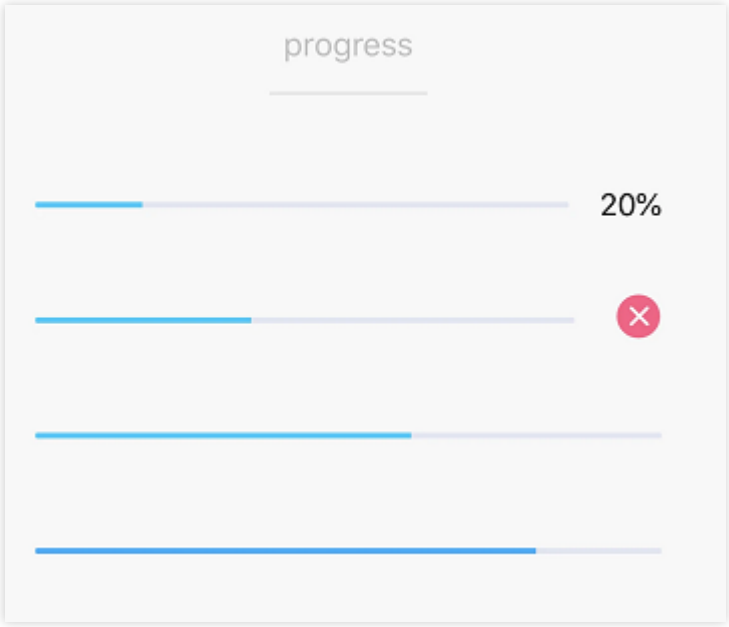
Sample Code

©2013-2022 Tencent Cloud. All rights reserved.



```
<progress percent="20" show-info />
<progress percent="40" stroke-width="12" />
<progress percent="60" color="blue" />
<progress percent="80" active />
```

©2013-2022 Tencent Cloud. All rights reserved.



text

Feature Description: Text.

Parameter and Description:

Attribute	Type	Default value	Description
selectable	boolean	false	Text selection availability
space	string	-	Display consecutive spaces
decode	boolean	false	Whether to decode

Value	Description
ensp	Half-size space for Chinese characters
emsp	Full-size space for Chinese characters
nbsp	Space size set based on the font

Note:

`decode` can parse the following: ` ` `<` `>` `&` `'` ` ` ` `.

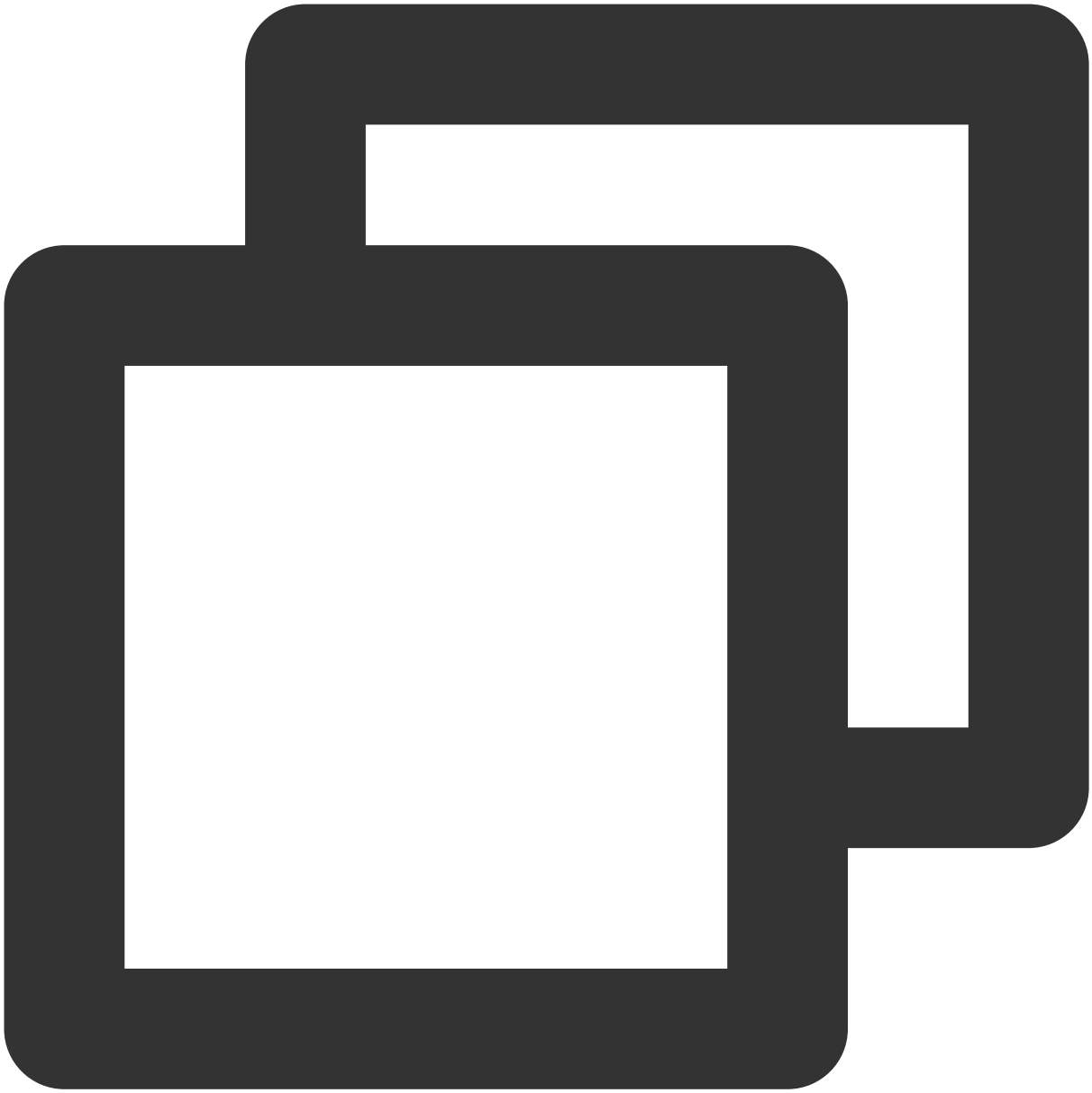
The space standards across various operating systems are not consistent.

The `<text>` component only supports nesting `<text>`.

Nodes other than text nodes cannot be selected by long pressing.

Corresponding WXML file

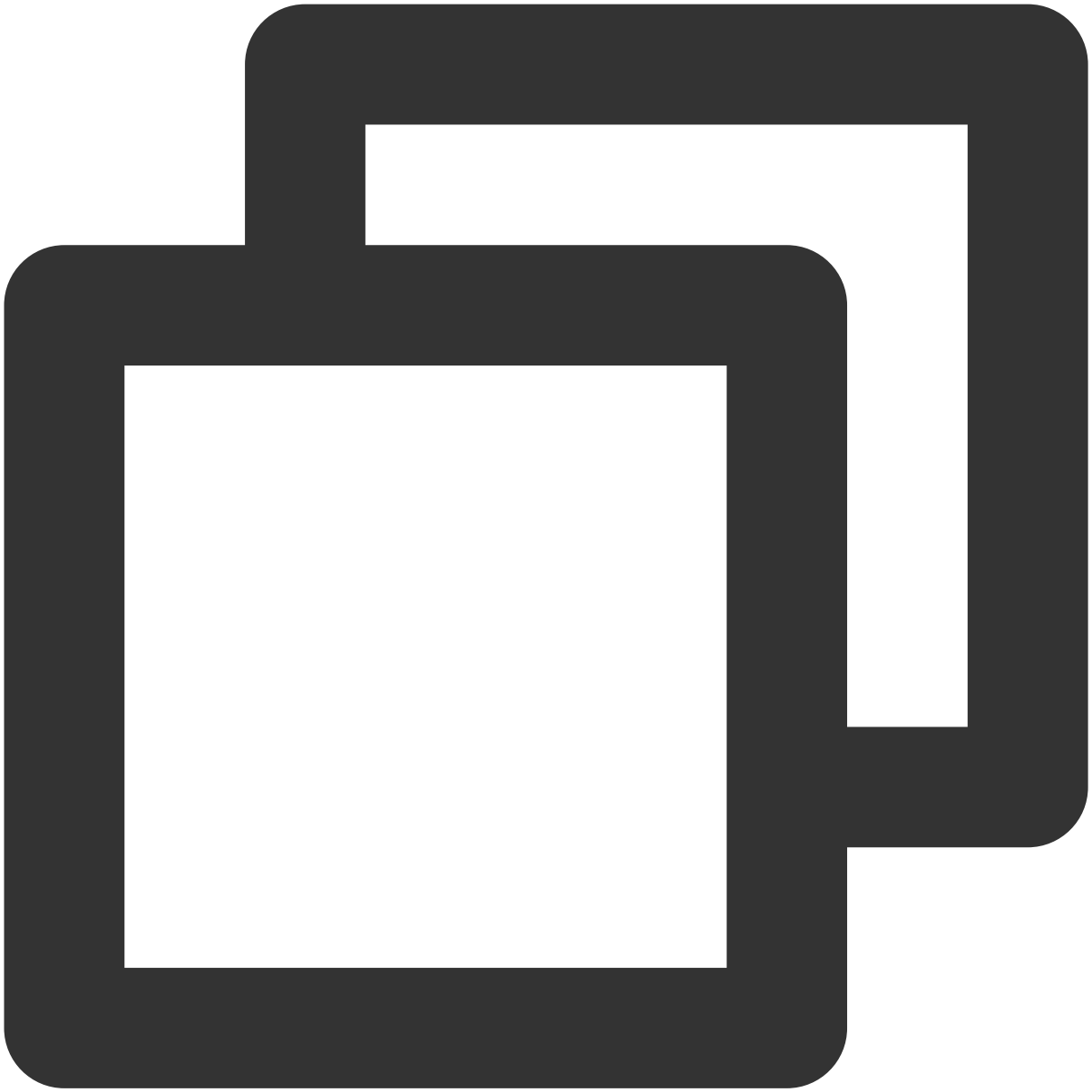
©2013-2022 Tencent Cloud. All rights reserved.



```
<view class="btn-area">
  <view class="body-view">
    <text>{{text}}</text>
    <button bindtap="add">add line</button>
    <button bindtap="remove">remove line</button>
  </view>
</view>
```

Corresponding js file

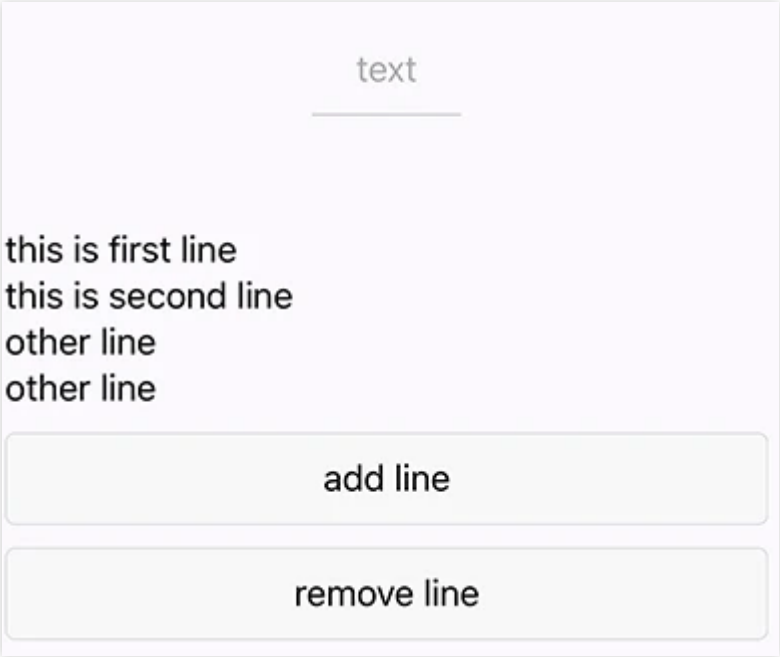
©2013-2022 Tencent Cloud. All rights reserved.



```
const initData = 'this is first line\\nthis is second line'
const extraLine = []
Page({
  data: {
    text: initData
  },
  add(e) {
    extraLine.push('other line')
    this.setData({
      text: initData + '\\n' + extraLine.join('\\n')
    })
```

©2013-2022 Tencent Cloud. All rights reserved.

```
  },
  remove(e) {
    if (extraLine.length > 0) {
      extraLine.pop()
      this.setData({
        text: initData + '\\n' + extraLine.join('\\n')
      })
    }
  }
})
```



rich-text

Feature Description: Rich text.

Parameter and Description:

Attribute	Type	Default value	Description
nodes	array / string	-	Node List / HTML String
space	string	-	Display consecutive spaces

Valid values of space:

Value	Description
ensp	Half-size space for Chinese characters

©2013-2022 Tencent Cloud. All rights reserved.

emsp	Full-size space for Chinese characters
nbsp	Space size set based on the font

Supports default events, including: `tap` , `touchstart` , `touchmove` , `touchcancel` , `touchend` , and `longtap` .

nodes

It is recommended to use the Array type for the nodes attribute, as the component will convert the String type into the Array type, resulting in a decrease in performance. Two types of nodes are now supported, distinguished by type, namely element nodes and text nodes. The default is the element node, which is the HTML node displayed in the rich text area.

Element Node: type = node

Attribute	Description	Type	Required	Remarks
name	Tag name	string	Yes	Supports a subset of trusted HTML nodes
attrs	Attribute	object	No	Supports a subset of recipient attributes, adhering to Pascal naming conventions.
children	List of child nodes	array	No	Consistent with the structure and nodes

Text node: type=text

Attribute	Description	Type	Required	Remarks
text	Text	string	Yes	Supports entities

Trusted HTML nodes and attributes

Globally supports class and style attributes and **does not support ID attributes.**

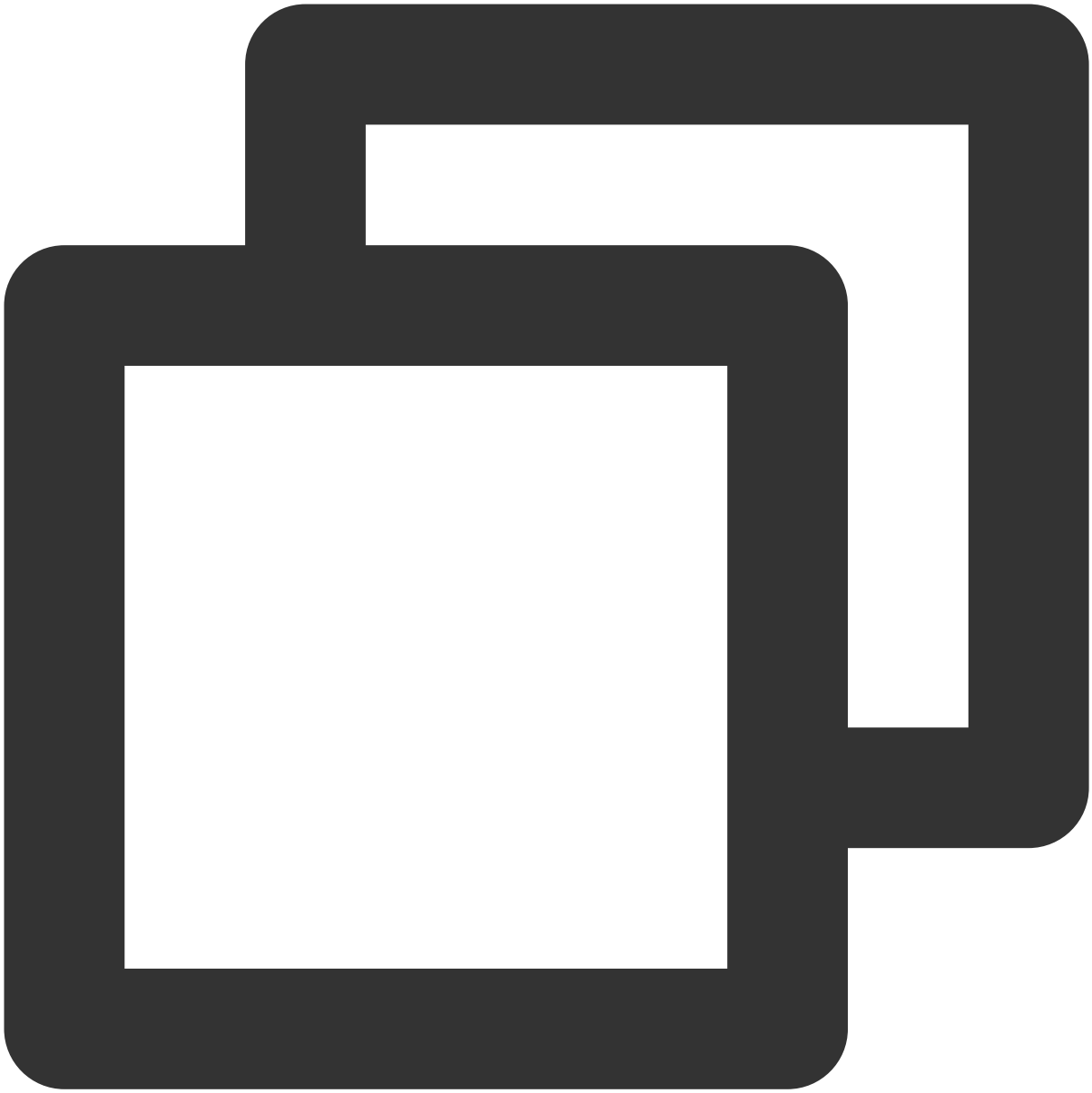
Node	Attribute	Node	Attribute
a	-	img	alt,src,height,width
abbr	-	ins	-
b	-	label	-
br	-	legend	-
code	-	li	-
col	span,width	ol	start,type

©2013-2022 Tencent Cloud. All rights reserved.

colgroup	span,width	p	-
dd	-	q	-
del	-	span	-
div	-	strong	-
dl	-	sub	-
dt	-	sup	-
em	-	table	width
fieldset	-	tbody	-
h1	-	td	colspan,height,rowspan,width
h2	-	tfoot	-
h3	-	th	colspan,height,rowspan,width
h4	-	thead	-
h5	-	tr	-
h6	-	ul	-
hr	-	-	-

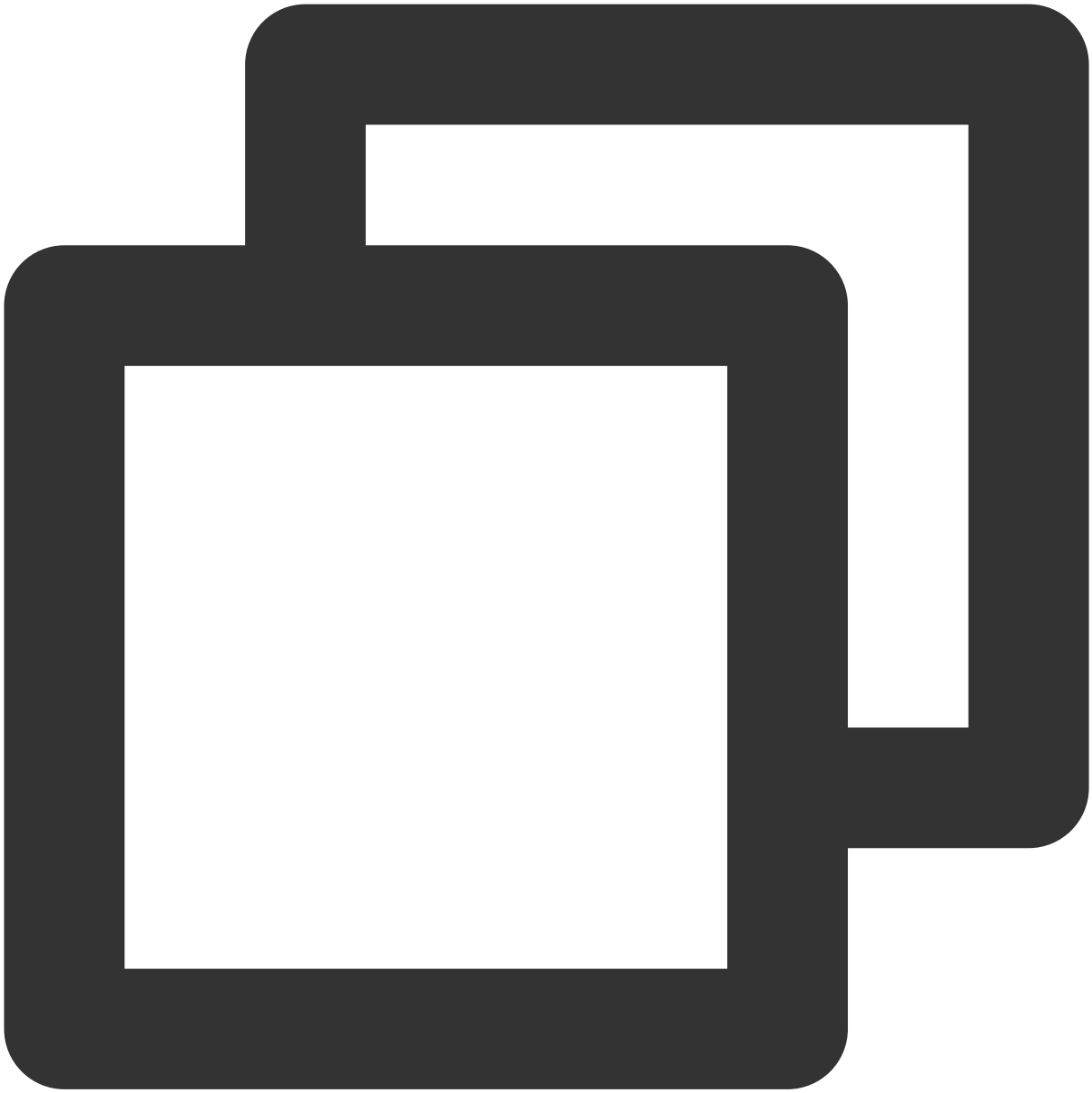
Sample Code

©2013-2022 Tencent Cloud. All rights reserved.



```
<!-- rich-text.wxml -->
<rich-text nodes="{{nodes}}" bindtap="tap"></rich-text>
```

©2013-2022 Tencent Cloud. All rights reserved.



```
// rich-text.js
Page({
  data: {
    nodes: [{
      name: 'div',
      attrs: {
        class: 'div_class',
        style: 'line-height: 60px; color: red;'
      },
      children: [{
        type: 'text',
```

©2013-2022 Tencent Cloud. All rights reserved.

```
          text: 'Hello World!'
        }]
      }]
    },
    tap() {
      console.log('tap')
    }
  })
```

Note:

It is not recommended to use String type for nodes, as it may result in performance degradation.

The rich-text component blocks all node events.

The attrs attribute does not support id, but supports class.

The name attribute is case-insensitive.

If untrusted HTML nodes are used, the node and all its child nodes will be removed.

The img tag only supports network images.

If the rich-text component is used within a custom component, only the wxss style of the custom component will apply to the class in the rich-text. It is not recommended to use the String type for `nodes` , as it may result in performance degradation.

©2013-2022 Tencent Cloud. All rights reserved.

Map Component

Last updated：2024-07-12 18:17:31

map

Feature Description: Map. This component is a native component, so mind the associated limitations when using it. For related APIs, see createMapContext.

Compatibility Description:

System Map: Only supported on iOS.

Google Maps: Supported on Android and IDE.

Huawei Map: Only supported on Android.

Tencent Map: Supported on Android and IDE.

Attribute	Type	Default value	Required	Description	iOS Compatibility	Android Compatibility
type	string	tencent	No	Map Type: Tencent or Google	No	No
longitude	number	-	Yes	Central Longitude	Yes	Yes
latitude	number	-	Yes	Central Latitude	Yes	Yes
scale	number	16	No	Scaling level, ranging from 3 to 20.	Yes	Yes
min-scale	number	3	No	Minimum scaling level		Yes
max-scale	number	20	No	Maximum scaling level		Yes
markers	array.<marker>	-	No	Marker Point	Yes	Yes
polyline	array.<polyline>	-	No	Route	Yes	Yes

©2013-2022 Tencent Cloud. All rights reserved.

circles	array.<circle>	-	No	Circle	Yes	Yes
controls	array.<control>	-	No	Widget	No	No
include-points	array.<point>	-	No	Scale the view to contain all given coordinate points.	Yes	Yes
show-location	boolean	false	No	Display the current location point with direction.	Yes	Yes
polygons	array.<polygon>	-	No	Polygon	Yes	Yes
subkey	string	-	No	Key used for personalized map customization.	No	No
layer-style	number	1	No	The style configuration of the personalized map does not support dynamic modifications.	No	No
rotate	number	0	No	Rotation angle, ranging from 0 to 360, represents the angle between the map's true north and the device's Y-axis.	No	Yes
skew	number	0	No	Inclination	System	Yes

©2013-2022 Tencent Cloud. All rights reserved.

				angle, ranging from 0 to 40, represents the tilt angle about the Z-axis.	Map: No Tencent Map: Yes Baidu Map: Yes Amap: Yes	
enable-3D	boolean	false	No	Display 3D building blocks	System Map: No Tencent Map: Yes Baidu Map: No Amap: No	Google Map: Yes Huawei Map: Yes Tencent Map: Yes Baidu Map: Yes Amap: No
show-compass	boolean	false	No	Display the compass	Yes	Yes
show-scale	boolean	false	No	Display the scale bar	Yes	Google Map: No Huawei Map: No Tencent Map: Yes Baidu Map: Yes Amap: Yes
enable-overlooking	boolean	false	No	Enable overhead view	System Map: No Tencent Map: Yes Baidu Map: Yes Amap: Yes	Yes
enable-zoom	boolean	true	No	Whether scaling is supported	Yes	Yes
enable-scroll	boolean	true	No	Whether dragging is supported	Yes	Yes
enable-rotate	boolean	-	No	Whether	Yes	

©2013-2022 Tencent Cloud. All rights reserved.

				rotation is supported		Yes
enable-satellite	boolean	false	No	Whether satellite imagery is enabled	Yes	Google Map: Yes Huawei Map: No Tencent Map: Yes Baidu Map: Yes Amap: Yes
enable-traffic	boolean	false	No	Whether the real-time traffic feature is enabled	Yes	Yes
enable-poi	boolean	true	No	Whether display POI points	System Map: No Tencent Map: Yes Baidu Map: No Amap: No	Google Map: Yes Huawei Map: No Tencent Map: Yes Baidu Map: Yes Amap: Yes
enable-building	boolean	true	No	Whether display buildings	System Map: No Tencent Map: Yes Baidu Map: Yes Amap: Yes	Google Map: Yes Huawei Map: No Tencent Map: Yes Baidu Map: No Amap: Yes
setting	object	-	No	Configuration	No	No
bindtap	eventhandle	-	No	Triggered upon clicking the map, returning latitude and	Yes	Yes

©2013-2022 Tencent Cloud. All rights reserved.

				longitude information.		
bindmarkertap	eventhandle	-	No	Triggered upon clicking the marker point, e.detail = {markerId}	Yes	Yes
bindlabeltap	eventhandle	-	No	Triggered upon clicking the label, e.detail = {markerId}	Yes	No
bindcontroltap	eventhandle	-	No	Triggered upon clicking the control, e.detail = {controlId}	No	No
bindcallouttap	eventhandle	-	No	Triggered upon clicking the bubble corresponding to the marker point, e.detail = {markerId}	No	No
bindupdated	eventhandle	-	No	Triggered upon the completion of the map's rendering update.	No	Yes
bindregionchange	eventhandle	-	No	Triggered twice when the field of view changes, returning type values of 'begin' and 'end' respectively.	Yes	Yes
bindpoitap		-	No	Triggered	No	Yes

©2013-2022 Tencent Cloud. All rights reserved.

	eventhandle			upon clicking the map's POI point, e.detail = {name, longitude, latitude}		
bindanchorpointtap	eventhandle	-	No	Triggered upon clicking the location marker, e.detail = {longitude, latitude}	No	No

Return value of 'regionchange'

Feature Description: When the field of view changes, 'regionchange' is triggered twice, returning type values of 'begin' and 'end' respectively.

During the 'begin' phase, 'causedBy' is returned, with valid values being 'gesture' (triggered by gesture) and 'update' (triggered by interface).

During the 'end' phase, 'causedBy' is returned, with valid values being 'drag' (caused by dragging), 'scale' (caused by scaling), and 'update' (caused by calling the update interface).

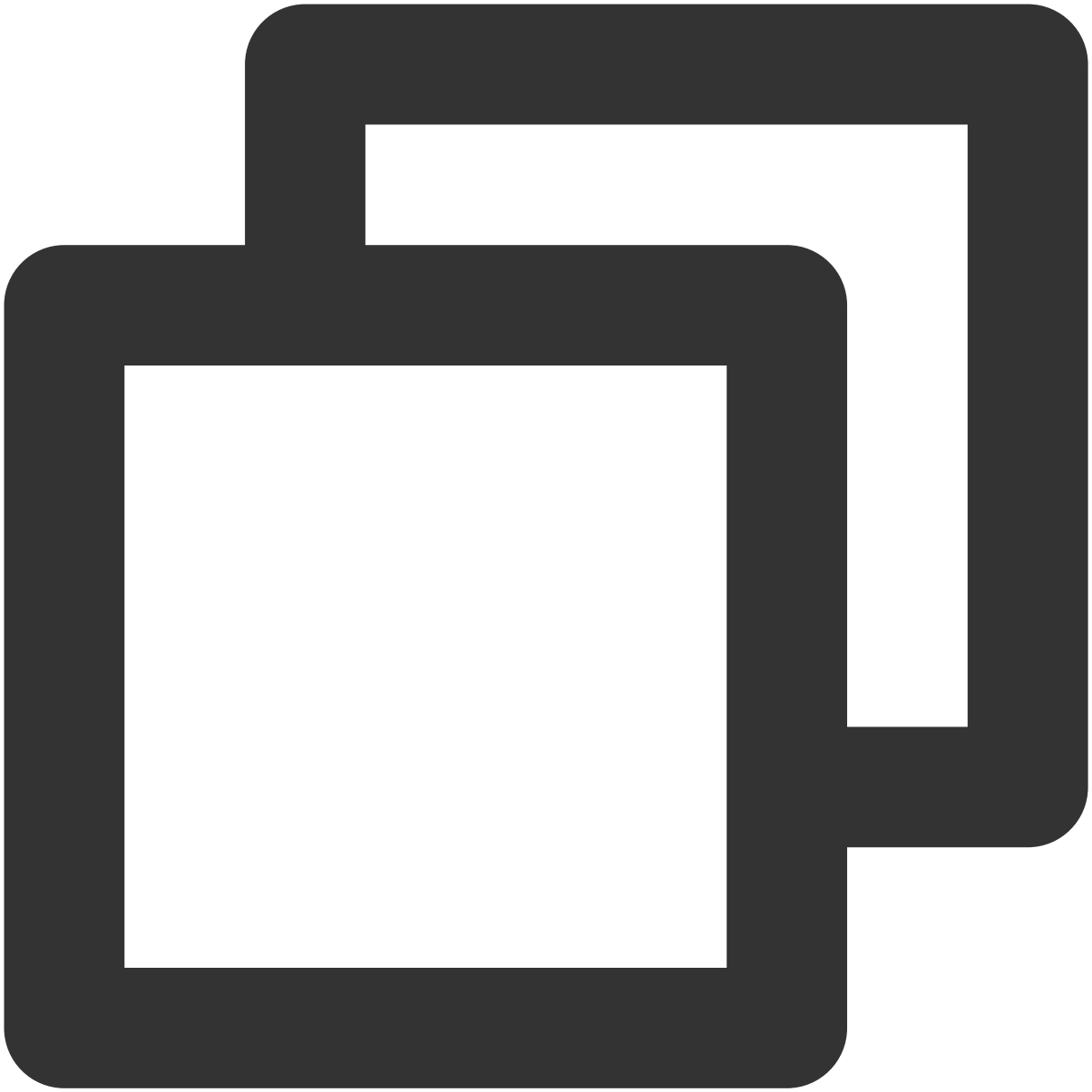
```
e = {causedBy, type, detail: {rotate, skew, scale, centerLocation, region}}
```

setting

Feature Description: Provides a 'setting' object to uniformly configure the map.

Sample Code

©2013-2022 Tencent Cloud. All rights reserved.



```
// Default Value
const setting = {
  rotate: 0,
  showLocation: false,
  subKey: '',
  layerStyle: 1,
  enableZoom: true,
  enableScroll: true,
  enableRotate: false,
  showCompass: false,
  enableSatellite: false,
```

©2013-2022 Tencent Cloud. All rights reserved.

```
}

this.setData({
  // Only the attributes that are set will take effect, and others will remain unaf
  setting: {
    enableZoom: false,
    enableScroll: false,
  }
})
```

maker

Feature Description: The marker point is used to display the marked location on the map.

Attribute	Description	Type	Required	Remarks	Supported on iOS	Supported on Android
id	Marker Point ID	number	No	The callback of a marker click event will return this ID.	Yes	Yes
clusterId	Cluster ID	number	No	Used when customizing the effect of point aggregation clusters.	No	No
joinCluster	Whether participate in point aggregation	boolean	No	No participation by default	No	Google Map: Yes Huawei Map: Yes Tencent Map: Yes Baidu Map: No Amap: No
latitude	Latitude	number	Yes	Number of floating points, ranging from -90 to 90.	Yes	Yes

©2013-2022 Tencent Cloud. All rights reserved.

longitude	Longitude	number	Yes	Number of floating points, ranging from -90 to 90.	Yes	Yes
title	Marker point name	string	No	Displayed upon clicking, will ignored if callout exists.	Yes	Yes
zIndex	Display Level	number	No	-	No	Yes
iconPath	Displayed Icon	string	Yes	Image path under the project directory, supporting network path, local path, and code package path.	Yes	Yes
rotate	Rotation angle	number	No	The angle of clockwise rotation, ranging from 0 to 360, with a default of 0.	No	Yes
alpha	Marked transparency	number	No	Ranges from 0 to 1, with a default value of 1 that indicates no transparency.	No	Yes
width	Mark icon width	number/string	No	Defaults to the actual image width.	No	Yes
height	Mark icon height	number/string	No	Defaults to the actual	No	Yes

©2013-2022 Tencent Cloud. All rights reserved.

				height of the image.		
callout	Bubble window above the marker point.	object	No	See the table below for supported attributes. Newline characters are recognizable.	Yes	Yes
customCallout	Customize the bubble window.	object	No	See the table below for supported attributes.	No	No
label	Add labels adjacent to the marker points.	object	No	See the table below for supported attributes. Newline characters are recognizable.	No	No
anchor	The latitude and longitude are at the anchor point of the marker icon, defaulting to the midpoint of the bottom edge.	object	No	{x, y}, where x represents the horizontal direction (0-1) and y represents the vertical direction (0-1). {x: .5, y: 1} denotes the midpoint of the bottom edge.	No	Yes
aria-label	Accessibility, (Attribute) Additional description of the element.	string	No	-	No	No

©2013-2022 Tencent Cloud. All rights reserved.

Note：
It is recommended to assign a numerical type id to each marker to ensure optimal performance when updating markers.

The callout bubble on the marker.

Attribute	Description	Type	iOS Compatibility	Android Compatibility	IDE Compatibility
content	Text	string	Yes	Yes	Yes
color	Text Color	string	No	Yes	Yes
fontSize	Font Size	number	No	Yes	Yes
borderRadius	Border chamfer	number	No	Yes	Yes
borderWidth	Border Width	number	No	Yes	Yes
borderColor	Outline color	string	No	Yes	Yes
bgColor	Background Color	string	No	Yes	Yes
padding	Text Padding	number	No	Yes	Yes
display	'BYCLICK': Display on Click; 'ALWAYS': Always Visible	string	No	No	Yes
textAlign	Text Alignment Method. Valid values: left, right, center	string	No	Yes	No
anchorX	Horizontal offset, with positive values indicating a shift to the right.	number	No	Google Map: Yes Huawei Map: Yes Tencent Map: Yes Baidu Map: No Amap: Yes	No
anchorY	Vertical offset, with positive	number	No	Google Map: Yes	No

©2013-2022 Tencent Cloud. All rights reserved.

	values indicating a downward shift.			Huawei Map: Yes Tencent Map: Yes Baidu Map: No Amap: Yes	

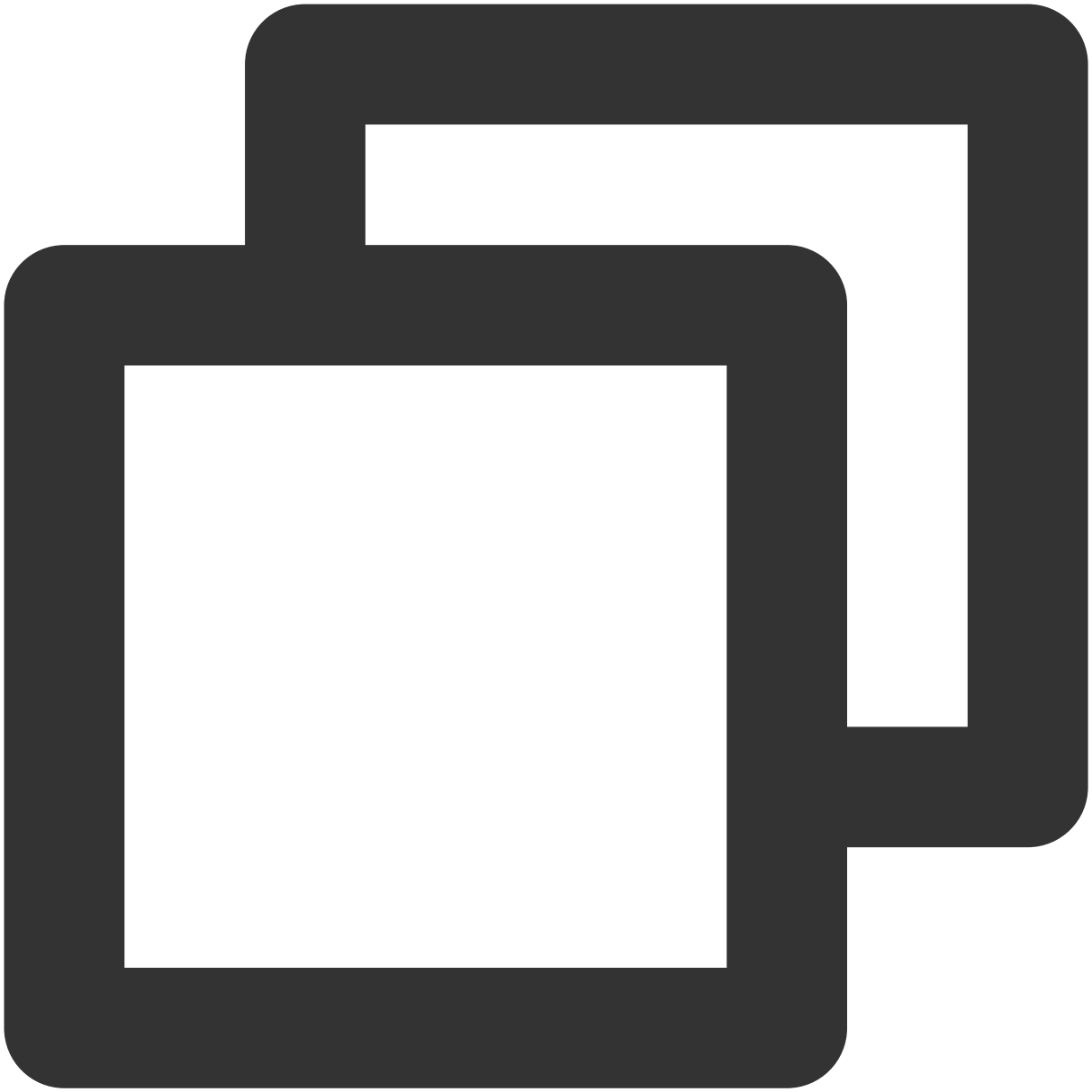
cumtomCallout, custom bubble on the marker

Function Description: When `customCallout` is present, the `callout` and `title` attributes will be ignored. The custom bubble uses `cover-view` for customization, offering greater flexibility.

Attribute	Description	Type	iOS Compatibility	Android Compatibility	IDE Compatibility
display	'BYCLICK': Display on Click; 'ALWAYS': Always Visible	string	No	No	No
anchorX	Horizontal offset, with positive values indicating a shift to the right.	number	No	No	No
anchorY	Vertical offset, with positive values indicating a downward shift.	number	No	No	No

The usage is as follows: Add a slot node named callout under the `map` component. The `cover-view` inside is bound to the `marker` through the `marker-id` attribute. When the `marker` is created, the content displayed by this `cover-view` will be displayed as a `callout` above the marker point.

©2013-2022 Tencent Cloud. All rights reserved.



```
<map>
  <cover-view slot="callout">
    <cover-view marker-id="1"></cover-view>
    <cover-view marker-id="2"></cover-view>
  </cover-view>
</map>
```

Bubble label on the marker

Attribute	Description	Type	iOS	Android	IDE

©2013-2022 Tencent Cloud. All rights reserved.

			Compatibility	Compatibility	Compatibility
content	Text	string	No	No	Yes
color	Text Color	string	No	No	Yes
fontSize	Font Size	number	No	No	Yes
anchorX	The coordinates of the label, whose origin is the point defined by the longitude and latitude corresponding to the marker.	number	No	No	Yes
anchorY	The coordinates of the label, whose origin is the point defined by the longitude and latitude corresponding to the marker.	number	No	No	Yes
borderWidth	Border Width	number	No	No	Yes
borderColor	Outline color	string	No	No	Yes
borderRadius	Border chamfer	number	No	No	Yes
bgColor	Background Color	string	No	No	Yes
padding	Text padding	number	No	No	Yes
textAlign	Text alignment method. Valid values: left, right, center.	string	No	No	Yes

Point Aggregation

Feature Description: When there are too many markers to be displayed on the map, it may cause overlapping of markers on the interface, incomplete display, and overall performance degradation. To address such issues, we have introduced the point aggregation capability.

System Map (Supported only on iOS)

The usage process is as follows:

MapContext.addMarkers specifies the markers involved in the aggregation.

(Optional) When `MapContext.on('markerClusterCreate', callback)` is triggered, the style of the aggregation cluster is updated through MapContext.addMarkers.

©2013-2022 Tencent Cloud. All rights reserved.

MapContext.removeMarkers removes the markers involved in the aggregation.

Google Map (Supported only on Android)

The usage method is to specify joinCluster when adding a marker. Markers that meet the conditions will be automatically aggregated.

polyline

Feature Description: Specifies a series of coordinate points, connecting from the first item in the array to the last. When drawing a rainbow line, it is necessary to specify the colors of different segments. For instance, if "points" contain 5 points, then colorList should pass in 4 color values; if the length of colorList is less than points.length - 1, then the color of the remaining segments will remain consistent with the last item.

Attribute	Description	Type	Required	Remarks	iOS Compatibility	Andr Com
points	Array of latitude and longitude coordinates	Array	Yes	[{latitude: 0, longitude: 0}]	Yes	Yes
color	Color of the line	string	No	Hexadecimal	Yes	Yes
colorList	Rainbow Line	Array	No	Ignore the color value when it exists	No	Goo No Hua Map Ten Map Baid Yes Ama
width	Line width	number	No		Yes	Yes
dottedLine	Is it a dashed line?	boolean	No	Defaults to "false"	Yes	Yes
arrowLine	Line with an arrow	boolean	No	-	No	Yes
arrowIconPath	Replace the arrow icon	string	No	-	No	Yes

©2013-2022 Tencent Cloud. All rights reserved.

borderColor	Line border color	string	No	-	No	Goo No Hua Map Ten Map Baid No Ama
borderWidth	Line thickness	number	No	-	No	Goo No Hua Map Ten Map Baid No Ama
level	Overlay relationship	string	No	Defaults to "abovelabels"	No	Goo No Hua Map Ten Map Baid No Ama
textStyle	Font Style	TextStyle	No	Text style on the polyline	No	Goo No Hua Map Ten Map Baid No Ama
segmentTexts	Segmented text	Array<SegmentText>	No	Content and position of text on the polyline	No	Goo No Hua Map Ten Map

©2013-2022 Tencent Cloud. All rights reserved.

						Baid No Ama

SegmentText

Attribute	Description	Type
name	Name	string
startIndex	Starting Point	number
endIndex	End point	number

TextStyle

Attribute	Description	Type	Default value
textColor	Text Color	string	#000000
strokeColor	Outline color	number	#ffffff
fontSize	Font Size	number	14

The `level` field represents the overlay relationship with other map elements, with the following optional values:

Value	Description
abovelabels	Displayed above all POIs
abovebuildings	Displayed above building blocks and below POIs
aboveroads	Displayed above the road and below the building blocks

polygon

Feature Description: Specifies a series of coordinate points, generating a closed polygon based on points coordinate data.

Attribute	Description	Type	Required	Remarks	Supported on iOS	Supported on Android	
dashArray	Dotted boundary line	Array<number>	No	The default value [0, 0] represents a	No	Google Map: No	

©2013-2022 Tencent Cloud. All rights reserved.

				solid line, [10, 10] signifies a dashed line composed of ten pixels of solid line and ten pixels of blank space (repeated in this manner).		Huawei Map: No Tencent Map: Yes Baidu Map: No Amap: No	
points	Array of latitude and longitude coordinates	array	Yes	[{latitude: 0, longitude: 0}]	Yes	Yes	
strokeWidth	Outline width	number	No	-	Yes	Yes	
strokeColor	Outline color	string	No	Hexadecimal	Yes	Yes	
fillColor	Filling Color	string	No	Hexadecimal	Yes	Yes	
zIndex	Set the Z-axis value for the polygon.	number	No	-	No	Yes	
level	Overlay relationship	string	No	Defaults to "abovelabels"	No	Google Map: No Huawei Map: No Tencent Map: Yes Baidu Map: No Amap: No	

circle

Feature Description: Displays images on the map.

Attribute	Description	Type	Required	Remarks	iOS Compatibility	Android Compatibility	IDE Cor

©2013-2022 Tencent Cloud. All rights reserved.

latitude	Latitude	number	Yes	Number of floating points, ranging from -90 to 90.	Yes	Yes	Yes
longitude	Longitude	number	Yes	Number of floating points, ranging from -180 to 180.	Yes	Yes	Yes
color	Outline color	string	No	Number of floating points, ranging from -180 to 180.	Yes	Yes	Yes
fillColor	Filling Color	string	No	Hexadecimal	Yes	Yes	Yes
radius	Radius	number	Yes	-	Yes	Yes	Yes
strokeWidth	Outline width	number	No	-	Yes	Yes	No
level	Overlay relationship	string	No	Defaults to "abovelabels"	No	Google Map: No Huawei Map: No Tencent Map: Yes Baidu Map: No Amap: No	No

control

Feature Description: Displays controls on the map, the controls not moving with the map. This feature is soon to be deprecated, and the use of cover-view is recommended.

Attribute	Description	Type	Required	Remarks	iOS Compatibility	Android Compatibility	IDE Compatib
id	Control id	number	No	This id will be returned	No	No	No

©2013-2022 Tencent Cloud. All rights reserved.

				in the callback of the control click event.			
position	Spatial position on the map	object	Yes	Spatial position relative to the map	No	No	No
iconPath	Displayed Icon	string	Yes	Image path in the project directory, supporting local path and code package path.	No	No	No
clickable	Whether it can be clicked	boolean	No	By default, it cannot be clicked.	No	No	No

position

Attribute	Description	Type	Required	Remarks	iOS Compatibility	Android Compatibility	IDE Compatibilit
left	Distance to the left boundary of the map	number	No	Defaults to 0.	No	No	No
top	Distance to the top boundary of the map	number	No	Defaults to 0.	No	No	No
width	Spatial width	number	No	Defaults to the image width.	No	No	No

©2013-2022 Tencent Cloud. All rights reserved.

height	Spatial height	number	No	Defaults to the image height.	No	No	No

Return value of "bindregionchange"

Attribute	Description	Type	Remarks
type	Triggered at the onset and conclusion of the changes to the field.	string	The onset of the change to the field of view is denoted as 'begin', and its conclusion as 'end'.
causedBy	The cause leading to the change to the field of view.	string	Induced by dragging the map (drag), scaling (scale), or invoking the interface (update).

Scale bar

scale	3	4	5	6	7	8	9	10	11
Proportion	1000km	500km	200km	100km	50km	50km	20km	10km	5km
scale	12	13	14	15	16	17	18	19	20
Proportion	2km	1km	500m	200m	100m	50m	50m	20m	10m

Note:

Personalized maps do not currently support debugging within the toolset. Please conduct tests using the client first.
The color values in the map, such as color/borderColor/bgColor, must be represented in six (or eight) digit hexadecimal format. In the case of eight digits, the last two represent the alpha value, for example: #000000AA.
The latitude and longitude of the map component are mandatory. If these are not provided, the default values will correspond to the coordinates of Beijing.
The latitude and longitude used by the map component are based on the Mars coordinate system. When invoking the wx.getLocation interface, it is necessary to specify the type as gcj02.
Please mind the usage restrictions for the native components.
If the page containing the current component, or the global configuration, has enabled the enablePassiveEvent option, this built-in component may exhibit unexpected behavior.

©2013-2022 Tencent Cloud. All rights reserved.

Media Component

Last updated：2024-03-12 11:22:25

image

Feature Description: Image.

Parameter and Description

Attribute	Type	Default value	Description
src	string	-	Image resource address, supporting cloud file ID (from version 2.2.3 onwards)
mode	string	'scaleToFill'	Image cropping and scaling mode
lazy-load	boolean	false	Image lazy loading. This is only effective for images under 'page' and 'scroll-view'.
binderror	handleevent	-	In the event of an error, the event name is published to the AppService, with the event object event.detail = {errMsg: 'something wrong'}.
bindload	handleevent	-	Upon the completion of image loading, the event name is published to the AppService, with the event object event.detail = {height:'image height in px', width:'image width in px'}.
aria-lable	string	-	Accessibility, (Attribute) Additional description of the element.

Note:

The default dimensions of the image component are 300 px in width and 225 px in height.

The image component does not support long-press recognition of QR codes/mini program codes. Long-press recognition is only supported within the wx.previewImage feature.

Valid values for mode: There are 13 modes, with 4 scaling modes and 9 cropping modes.

Mode	Value	Description
Scaling	scaleToFill	Scale the image without maintaining the aspect ratio, stretching the width and height of the image to completely fill the image element.
Scaling	aspectFit	Scale the image while maintaining the aspect ratio, ensuring the longer side of the image is fully displayed. This allows the image to be shown in its entirety.

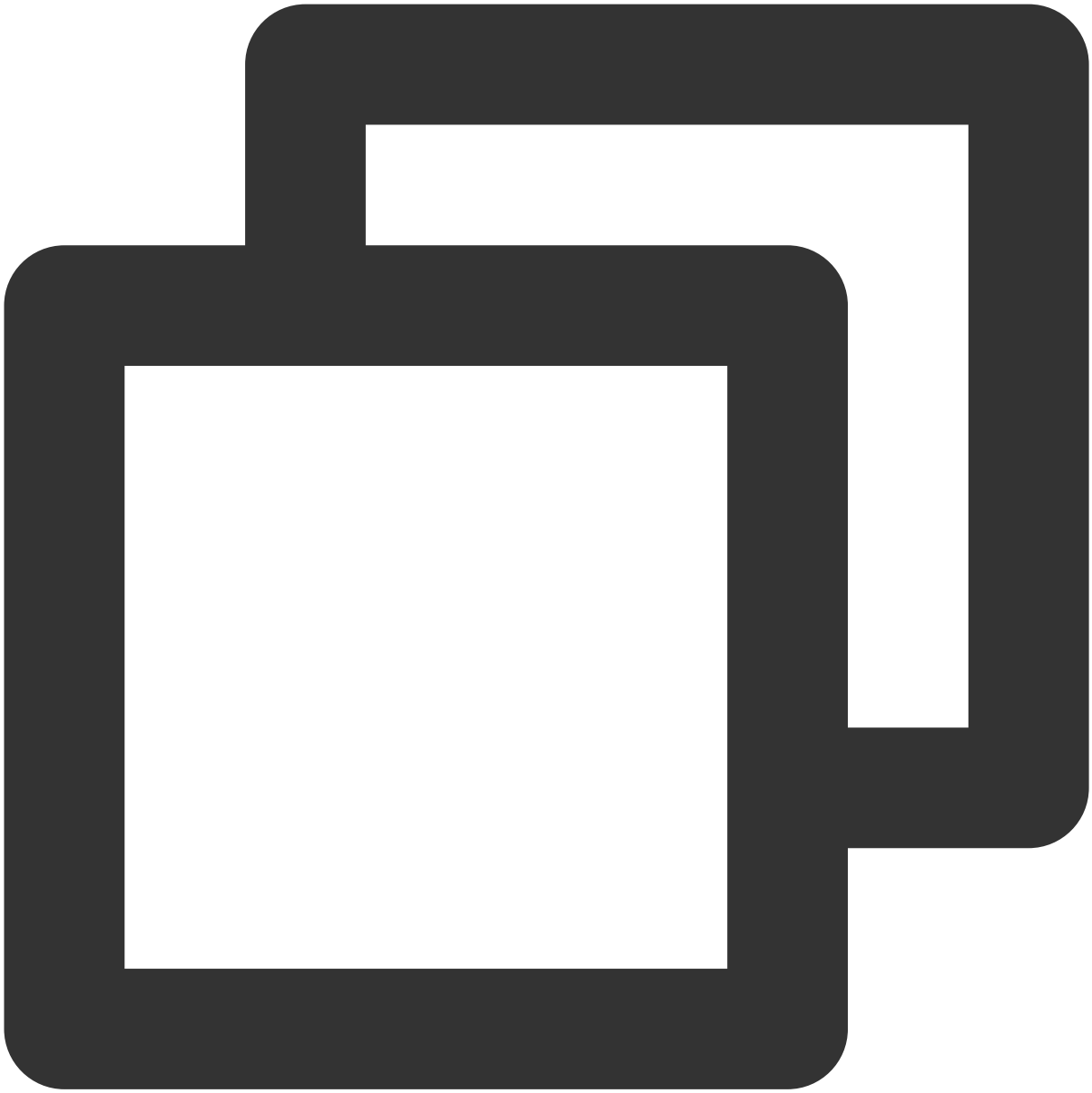
©2013-2022 Tencent Cloud. All rights reserved.

Scaling	aspectFill	Scale the image while preserving the aspect ratio, ensuring only the shorter side of the image is fully displayed. This typically results in the image being complete in either the horizontal or vertical direction, with the other direction being cropped.
Scaling	widthFix	Maintain the width constant while the height adjusts automatically, preserving the original aspect ratio of the image.
Cropping	top	Only display the top region of the image without scaling the image.
Cropping	bottom	Only display the bottom region of the image without scaling the image.
Cropping	center	Only display the central region of the image without scaling the image.
Cropping	left	Only display the left region of the image without scaling the image.
Cropping	right	Only display the right region of the image without scaling the image.
Cropping	top left	Only display the top left corner region of the image without scaling the image.
Cropping	top right	Only display the top right region of the image without scaling the image.
Cropping	bottom left	Only display the bottom left region of the image without scaling the image.
Cropping	bottom right	Only display the bottom right region of the image without scaling the image.

Sample Code

Corresponding WXML file

©2013-2022 Tencent Cloud. All rights reserved.

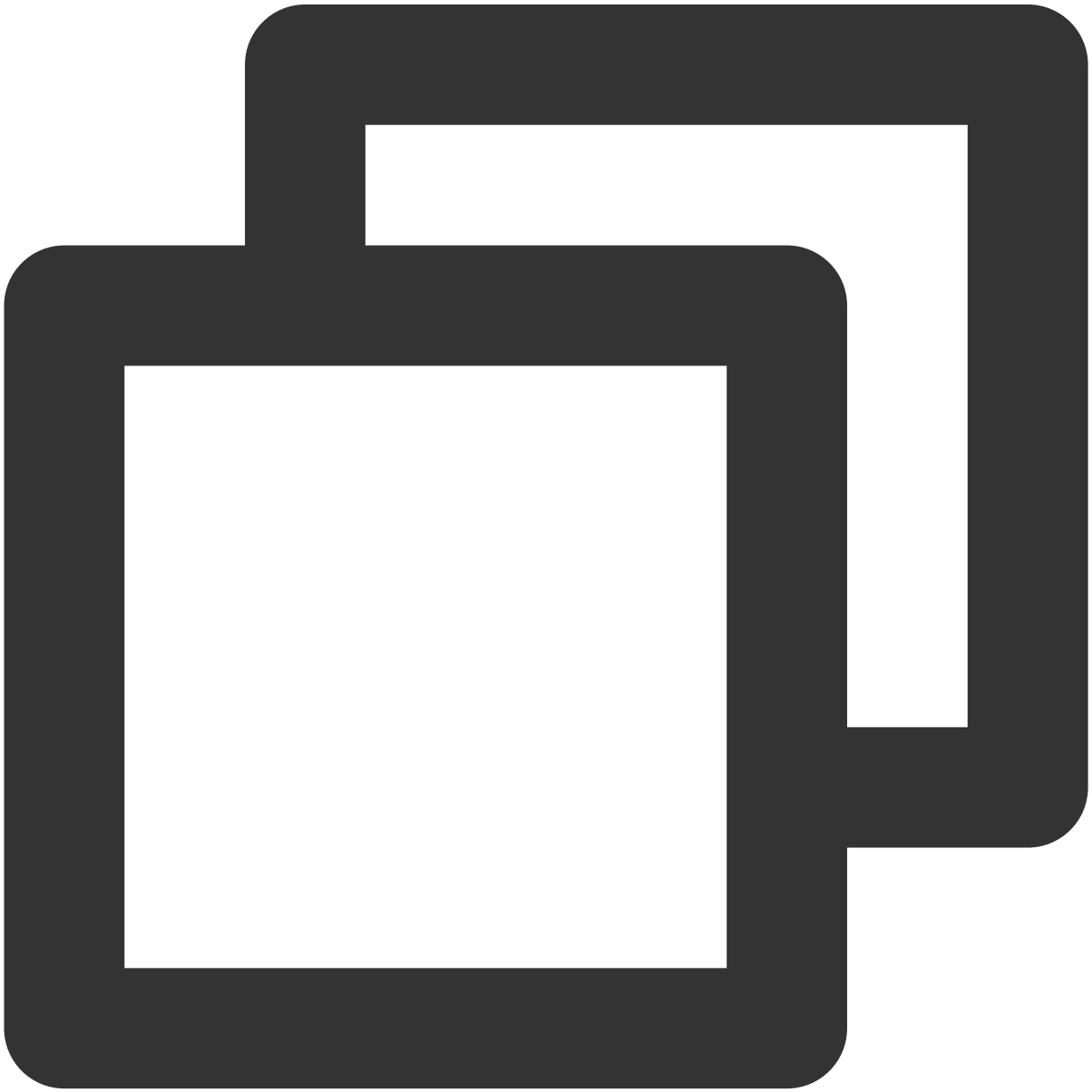


```
<view class="page">
  <view class="page__hd">
    <text class="page__title">image</text>
    <text class="page__desc">Image</text>
  </view>
  <view class="page__bd">
    <view class="section section_gap" wx.for="{{array}}" wx.for-item="item">
      <view class="section__title">{{item.text}}</view>
      <view class="section__ctn">
        <image
          style="width: 200px; height: 200px; background-color: #eeeeee;"
```

©2013-2022 Tencent Cloud. All rights reserved.

```
          mode="{{item.mode}}"
          src="{{src}}"
        ></image>
      </view>
    </view>
  </view>
</view>
```

Corresponding js file



```
Page({
  data: {
```

©2013-2022 Tencent Cloud. All rights reserved.

```
    array: [{
      mode: 'scaleToFill',
      text: 'scaleToFill: Do not maintain aspect ratio when scaling the image, ensu
    }, {
      mode: 'aspectFit',
      text: 'aspectFit: Maintain aspect ratio when scaling the image, ensuring the
    }, {
      mode: 'aspectFill',
      text: 'aspectFill: Maintain aspect ratio when scaling the image, ensuring the
    }, {
      mode: 'top',
      text: 'top: Only display the top region of the image without scaling the imag
    }, {
      mode: 'bottom',
      text: 'bottom: Only display the bottom region of the image without scaling th
    }, {
      mode: 'center',
      text: 'center: Only display the central region of the image without scaling t
    }, {
      mode: 'left',
      text: 'left: Only display the left region of the image without scaling the im
    }, {
      mode: 'right',
      text: 'right: Only display the right region of the image without scaling the
    }, {
      mode: 'top left',
      text: 'top left: Only display the top left region of the image without scalin
    }, {
      mode: 'top right',
      text: 'top right: Only display the top right region of the image without scal
    }, {
      mode: 'bottom left',
      text: 'bottom left: Only display the bottom left region of the image without
    }, {
      mode: 'bottom right',
      text: 'bottom right: Only display the bottom right region of the image withou
    }],
    src: '../resources/cat.jpg'
  },
  imageError(e) {
    console.log('Error event occurring in image3, carrying the value', e.detail.err
  }
})
```

©2013-2022 Tencent Cloud. All rights reserved.

video

Feature Description: Video. The earlier version is a native component. Remember the native component usage restrictions.

Attribute	Type	Default value	Description
src	string	-	The resource address for playing the video, supporting cloud file ID.
duration	number	-	Specify the video duration
controls	boolean	true	Whether to display the default playback controls (play/pause button, playback progress, time)
danmu-list	Object Array	-	Barrage list
danmu-btn	boolean	false	Whether to display the barrage button, which is effective only at initialization and cannot be dynamically changed.
enable-danmu	boolean	false	Whether to display the barrage, which is effective only at initialization and cannot be dynamically changed.
autoplay	boolean	false	Whether to enable auto-play
loop	boolean	false	Whether to enable loop playback
muted	boolean	false	Whether to play in mute
initial-time	number	-	Specify the initial playback position of the video.
page-gesture	boolean	false	Whether to enable brightness and volume adjustment gestures in non-full screen mode.
direction	number	-	Sets the orientation of the video in full screen mode. If not specified, it will be automatically determined based on the aspect ratio. The valid value is 0 (normal vertical orientation). The valid value is 90 (screen rotated 90 degrees counterclockwise). The valid value is -90 (screen rotated 90 degrees clockwise).
show-progress	boolean	true	If not set, it will only display when the width exceeds 240.

©2013-2022 Tencent Cloud. All rights reserved.

show-fullscreen-btn	boolean	true	Whether to display the full-screen button.
show-play-btn	boolean	true	Whether to display the play button on the bottom control bar of the video.
show-center-play-btn	boolean	true	Whether to display the play button in the middle of the video.
enable-progress-gesture	boolean	true	Whether to activate the gesture control for progress.
object-fit	string	contain	The video's display mode when the video size is inconsistent with the video container size. Contain: To encompass or include. Fill: To populate or saturate. Cover: To overlay or envelop.
poster	string	-	The online resource address or cloud file ID for the video cover image. If the controls attribute value is false, then the poster setting is invalid.
show-mute-btn	boolean	false	Whether to display the mute button.
title	string	-	The title of the video, displayed at the top when in full-screen mode.
play-btn-position	string	bottom	The position of the play button. Valid value is 'bottom' (on the controls bar). Valid value is 'center' (in the middle of the video).
enable-play-gesture	boolean	false	Whether to enable play gestures, i.e., double-tap to toggle between play/pause.
auto-pause-if-navigate	boolean	true	Whether to automatically pause the video on this page when redirecting to other mini program pages.
auto-pause-if-open-native	boolean	true	Whether to automatically pause the video on this page when redirecting to other native pages.
bindplay	eventhandle	-	The 'play' event is triggered when playback starts or resumes.
bindpause	eventhandle	-	The 'pause' event is triggered when playback pauses.
bindended	eventhandle	-	The 'ended' event is triggered when playback reaches the end.

©2013-2022 Tencent Cloud. All rights reserved.

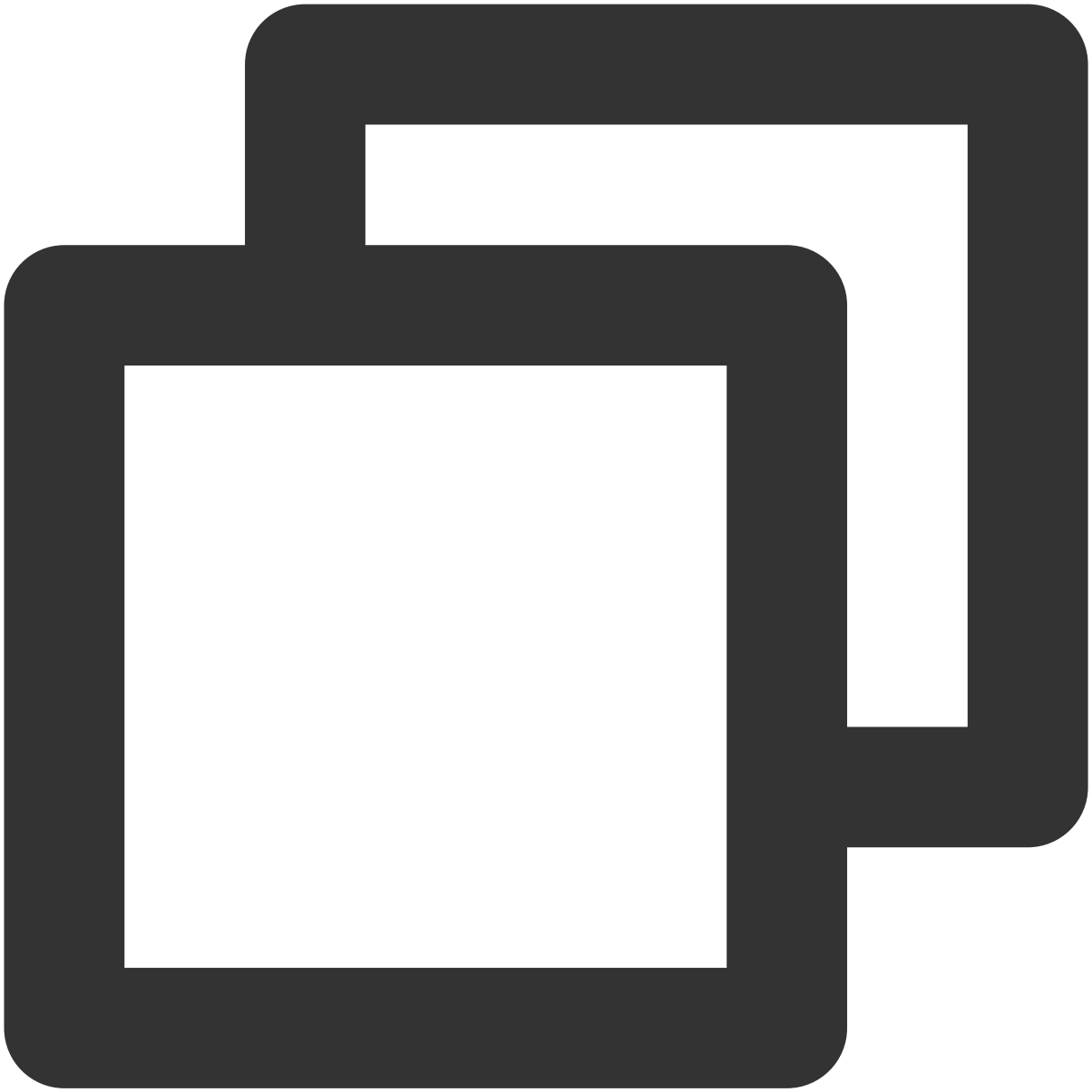
bindtimeupdate	eventhandle	-	The event is triggered when the playback progress changes, with event.detail = {currentTime, duration}. The triggering frequency is once every 250 ms.
bindfullscreenchange	eventhandle	-	The event is triggered when the video enters or exits full screen, with event.detail = {fullScreen, direction}. The valid values for direction are 'vertical' or 'horizontal'.
bindwaiting	eventhandle	-	The event is triggered when the video experiences buffering.
binderror	eventhandle	-	The event is triggered when an error occurs during video playback.
bindprogress	eventhandle	-	The event is triggered when the loading progress changes, supporting only loading of a section. The event.detail = {buffered}, expressed as a percentage.

The default dimensions of `<video>` are 300px in width and 225px in height, which can be adjusted via wxss.

Sample Code

Corresponding WXML file

©2013-2022 Tencent Cloud. All rights reserved.



```
<view class="section tc">
  <video src="{{src}}" controls></video>
  <view class="btn-area">
    <button bindtap="bindButtonTap">Get video</button>
  </view>
</view>

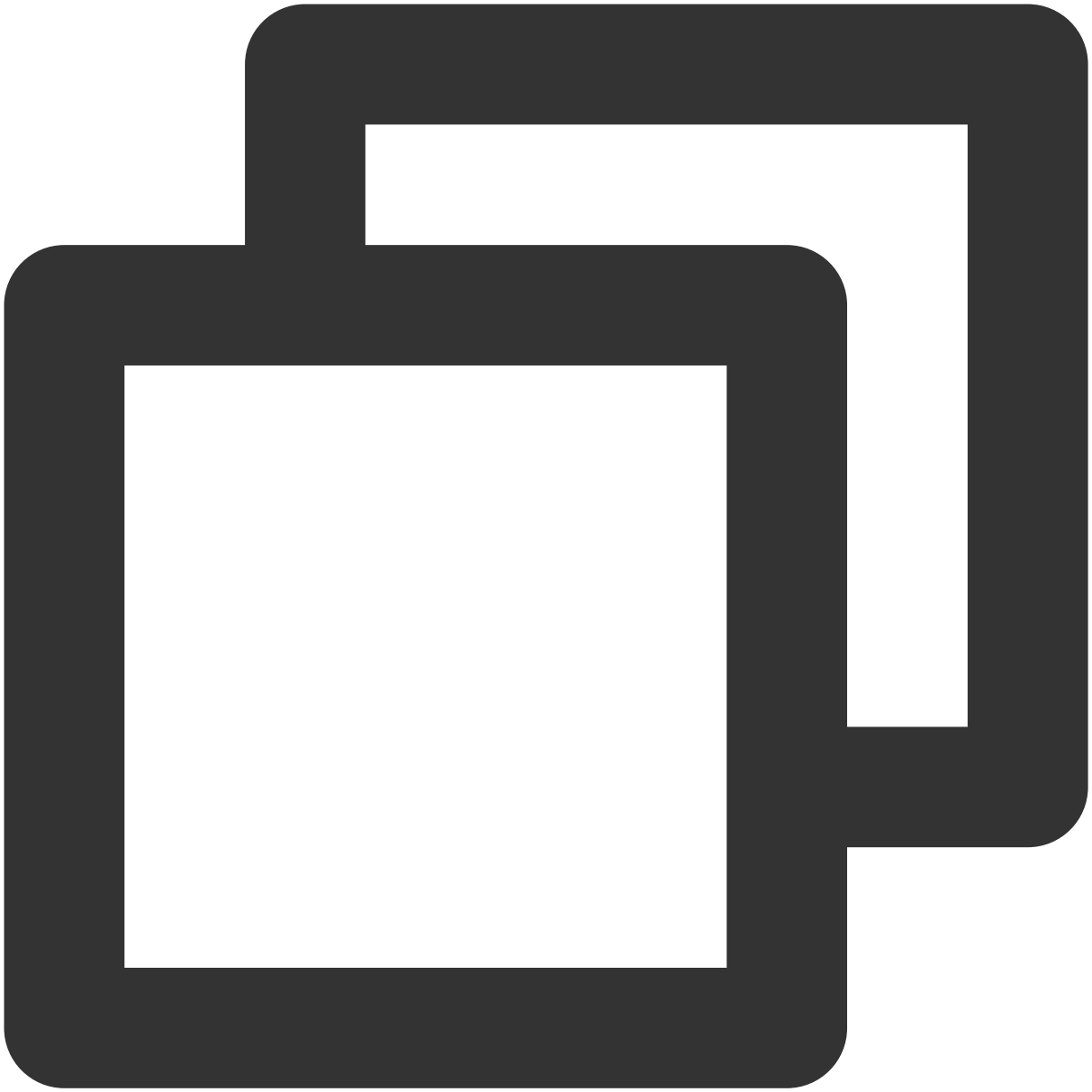
<view class="section tc">
  <video
    id="myVideo"
    src="https://qzonestyle.gtimg.cn/qzone/qzact/act/external/qq-video/qq-video.mp4
```

©2013-2022 Tencent Cloud. All rights reserved.

```
    danmu-list="{{danmuList}}"
    enable-danmu
    danmu-btn
    controls
  ></video>
  <view class="btn-area">
    <button bindtap="bindButtonTap">Get video</button>
    <input bindblur="bindInputBlur" />
    <button bindtap="bindSendDanmu">Dispatch the barrage</button>
  </view>
</view>
```

Corresponding js file

©2013-2022 Tencent Cloud. All rights reserved.



```
function getRandomColor() {
  const rgb = []
  for (let i = 0; i < 3; ++i) {
    let color = Math.floor(Math.random() * 256).toString(16)
    color = color.length == 1 ? '0' + color : color
    rgb.push(color)
  }
  return '#' + rgb.join('')
}

Page({
```

©2013-2022 Tencent Cloud. All rights reserved.

```
  onReady(res) {
    this.videoContext = wx.createVideoContext('myVideo')
  },
  inputValue: '',
  data: {
    src: '',
    danmuList: [
      {
        text: 'Barrage appearing at the 1st second',
        color: '#ff0000',
        time: 1
      },
      {
        text: 'Barrage appearing at the 3rd second',
        color: '#ff00ff',
        time: 3
      }]
  },
  bindInputBlur(e) {
    this.inputValue = e.detail.value
  },
  bindButtonTap() {
    const that = this
    wx.chooseVideo({
      sourceType: ['album', 'camera'],
      maxDuration: 60,
      camera: ['front', 'back'],
      success(res) {
        that.setData({
          src: res.tempFilePath
        })
      }
    })
  },
  bindSendDanmu() {
    this.videoContext.sendDanmu({
      text: this.inputValue,
      color: getRandomColor()
    })
  }
})
```

animation-video

©2013-2022 Tencent Cloud. All rights reserved.

Feature Description: The animation-video is a frontend component that provides the ability to render specific video resources into transparent background animations for mini programs. This can assist developers in achieving more immersive and rich animation effects at a low cost. The animation-video component also offers a wealth of APIs to control the animation's playback, pause, and jump to a specified position, etc. For related APIs, see wx.createAnimationVideo.

Parameter and Description:

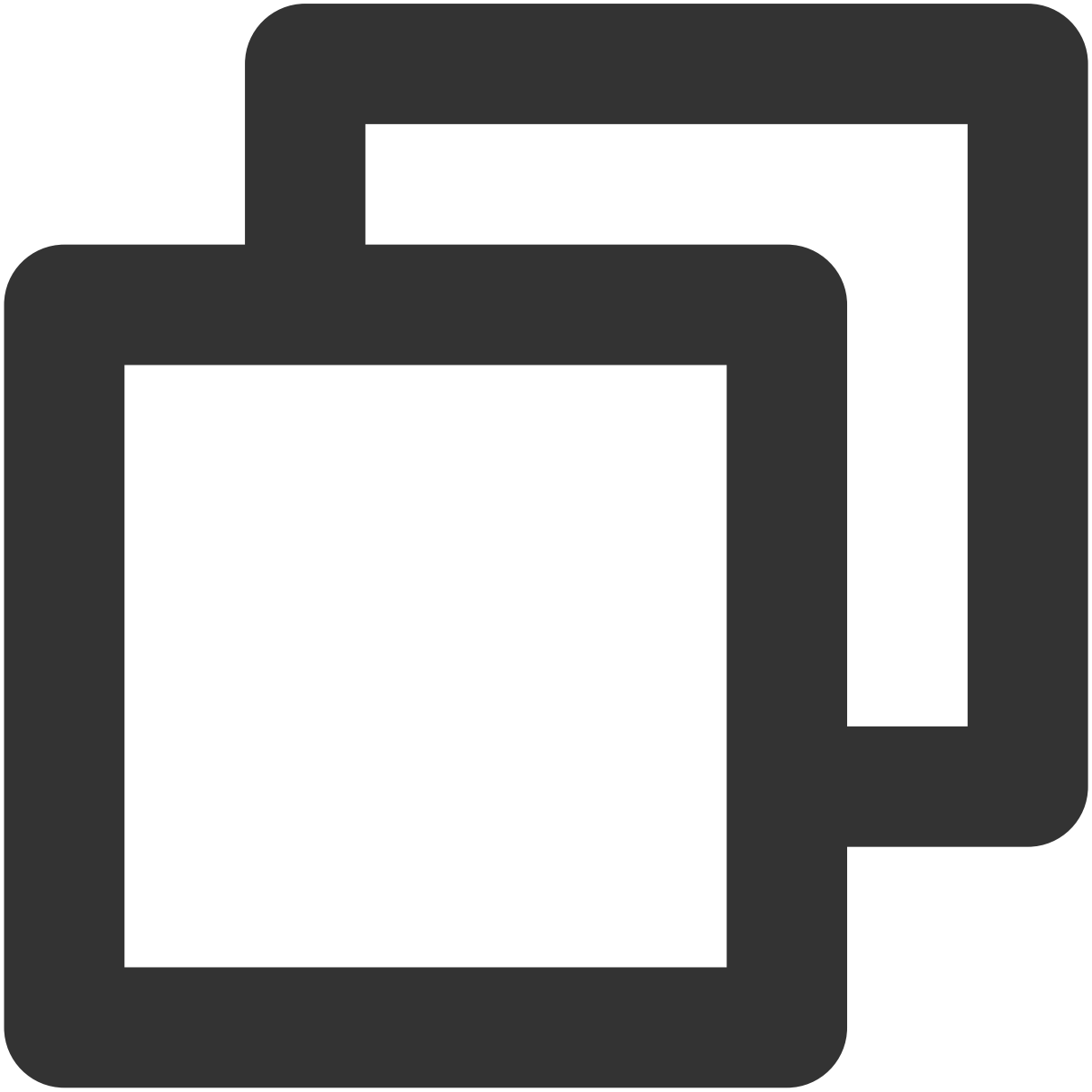
Attribute	Type	Default value	Required	Description	Earliest Version
resource-width	number	800	No	The width of the video resource used by the component (unit: px)	1.4.99
resource-height	number	400	No	The height of the video resource used by the component (unit: px)	1.4.99
canvas-style	string	width: 400px;height: 400px;	No	Used to set the CSS style of the animation canvas	1.4.99
path	string		Yes	The animation resource address, supporting both relative paths and remote addresses. If it is a remote address, the	1.4.99
loop	boolean	false	No	Whether to enable loop playback of animation.	1.4.99
autoplay	boolean	false	No	Whether the animation is automatic.	1.4.99
alpha-direction	string	left	No	The direction of the alpha channel in the video resource; 'left' indicates that the alpha channel is on the left side of the resource, while 'right' signifies that the alpha channel is on the right side of the resource.	1.4.99
bindstarted	eventhandle	-	No	Callback for the commencement of the animation playback	1.4.99

©2013-2022 Tencent Cloud. All rights reserved.

bindended	eventhandle	-	No	The 'ended' event is triggered when playback reaches the end (a callback is triggered upon natural playback completion, but not when looped playback ends or the animation pauses).	1.4.99

Sample Code

Corresponding WXML file



©2013-2022 Tencent Cloud. All rights reserved.

```
<view class="wrap">
    <view class="card-area">
        <view class="top-description border-bottom">
            <view>The animation's alpha channel information is on the right side
            <view>alpha-direction='right'</view>
        </view>
        <view class="video-area">
            <animation-video
                path="{{rightAlphaSrcPath}}"
                loop="{{loop}}"
                resource-width="800"
                resource-height="400"
                canvas-style="width:200px;height:200px"
                autoplay="{{autoplay}}"
                bindstarted="onStarted"
                bindended="onEnded"
                alpha-direction='right'
            ></animation-video>
        </view>
    </view>

    <view class="card-area">
        <view class="top-description border-bottom">
            <view>The animation's alpha channel information is on the left side o
            <view>alpha-direction='left'</view>
        </view>
        <view class="video-area">
            <animation-video
                path="{{leftAlphaSrcPath}}"
                loop="{{loop}}"
                resource-width="800"
                resource-height="400"
                canvas-style="width:200px;height:200px"
                autoplay="{{autoplay}}"
                bindstarted="onStarted"
                bindended="onEnded"
                alpha-direction='left'
            ></animation-video>
        </view>
    </view>
</view>
Page({
  data: {
    loop: true,
    leftAlphaSrcPath: 'https://efe-h2.cdn.bcebos.com/ceug/resource/res/2020-1/157
    rightAlphaSrcPath: 'https://b.bdstatic.com/miniapp/assets/docs/alpha-right-ex
    autoplay: true
```

©2013-2022 Tencent Cloud. All rights reserved.

```
    },
    onStarted() {
      console.log('===onStarted');
    },
    onEnded() {
      console.log('===onEnded');
    }
  })
```

camera

Feature Description: System Camera. This component is a native component, so mind the relevant restrictions when using it.

Requires "User Authorization": scope.camera.

For related APIs, see wx.createCameraContext.

Note:

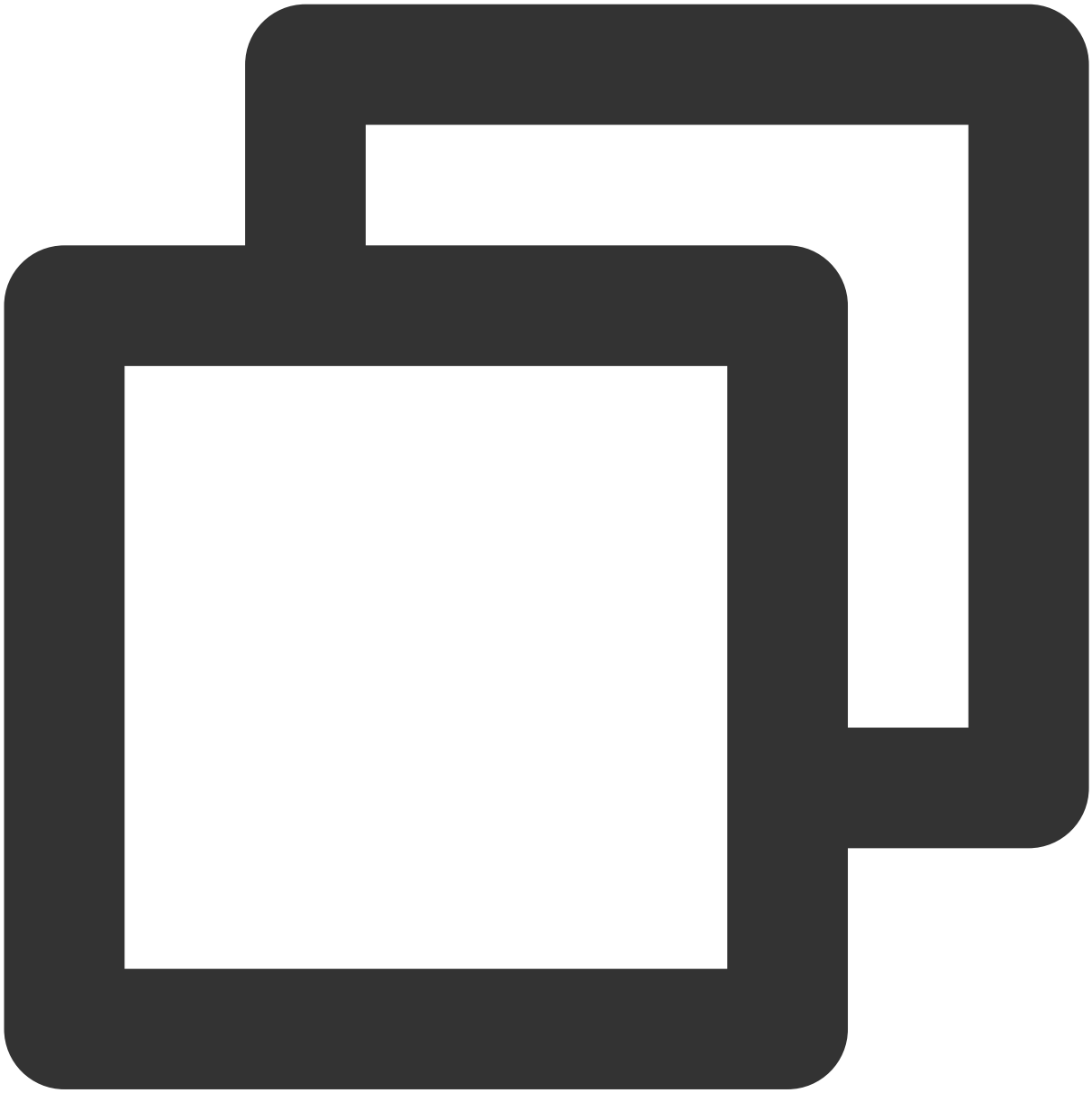
Only one camera component can be inserted on the same page.

Parameter and Description:

Attribute	Type	Default value	Description
mode	string	normal	Valid values are 'normal', 'scanCode'.
device-position	string	back	Front or rear, with values being 'front', 'back'.
flash	string	auto	Flashlight, with values being 'auto', 'on', 'off'.
bindstop	eventhandle	-	The camera is triggered in the event of an abnormal termination, such as exiting the backend and other situations.
binderror	eventhandle	-	Triggered when the user does not permit the use of the camera.
bindscancode	eventhandle	-	Triggered upon successful scan code recognition, effective only when mode is set to 'scanCode'.

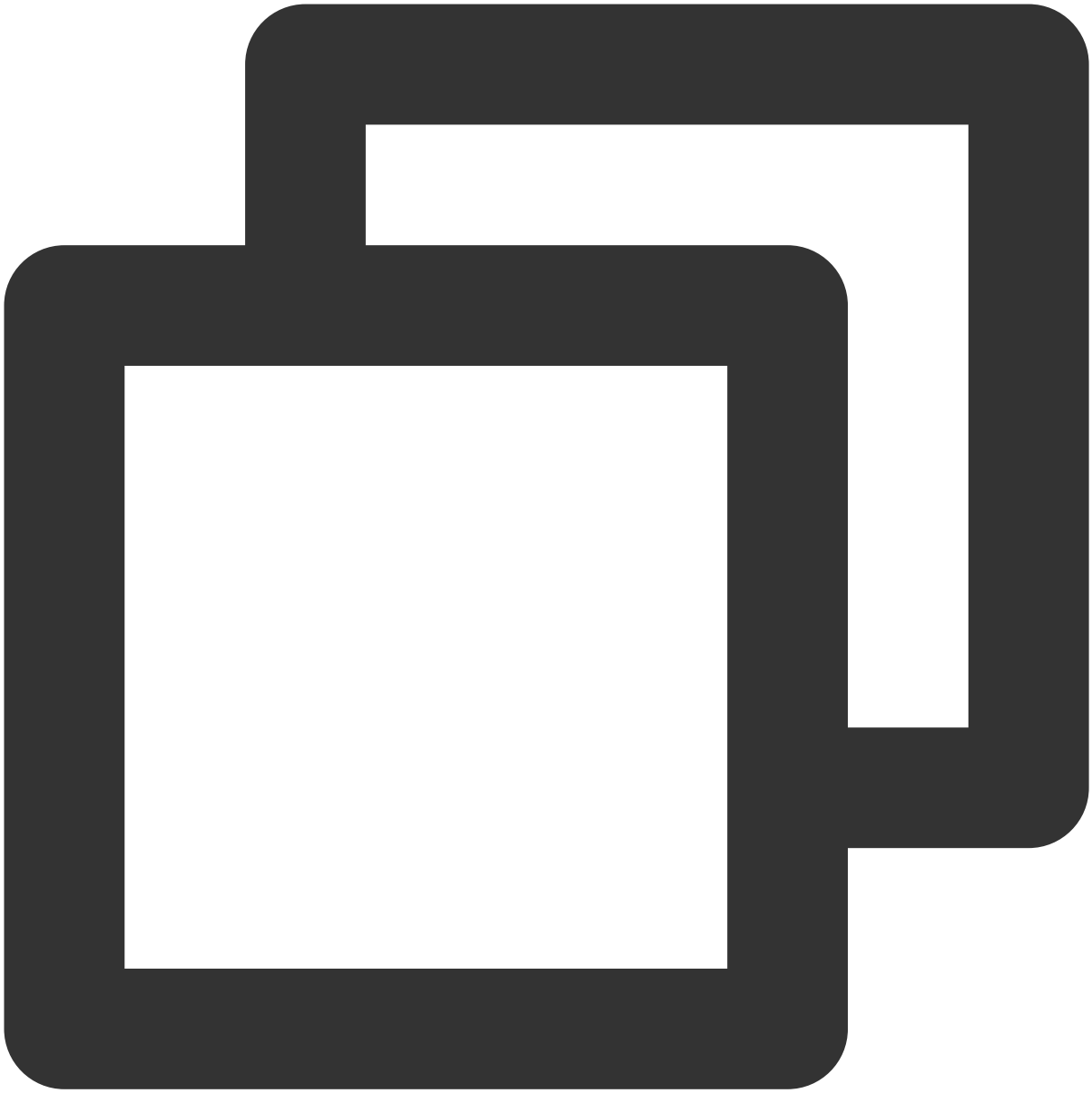
Sample Code

©2013-2022 Tencent Cloud. All rights reserved.



```
<!-- camera.wxml -->
<camera
  device-position="back"
  flash="off"
  binderror="error"
  style="width: 100%; height: 300px;"
></camera>
<button type="primary" bindtap="takePhoto">Take Photo</button>
<view>Preview</view>
<image mode="widthFix" src="{{src}}"></image>
```

©2013-2022 Tencent Cloud. All rights reserved.



```
// camera.js
Page({
  takePhoto() {
    const ctx = wx.createCameraContext()
    ctx.takePhoto({
      quality: 'high',
      success: (res) => {
        this.setData({
          src: res.tempImagePath
        })
      }
```

©2013-2022 Tencent Cloud. All rights reserved.

```
    })
  },
  error(e) {
    console.log(e.detail)
  }
})
```

live-player

Feature Description: Real-time audio and video playback. This component is a native component, so mind the limitations of native components when using it.

Parameter and Description:

Attribute	Type	Default value	Required	Description	Earliest Version
src	string	-	No	Audio and video address. Currently, only flv and rtmp formats are supported.	1.4.96
mode	string	live	No	Mode	1.4.96
autoplay	boolean	false	No	Auto play	1.4.96
muted	boolean	false	No	Whether to mute	1.4.96
orientation	string	vertical	No	Screen Orientation	1.4.96
object-fit	string	contain	No	Filling mode, with optional values being "contain" and "fillCrop".	1.4.96
min-cache	number	1	No	Minimum buffer zone, measured in seconds.	1.4.96
max-cache	number	3	No	Maximum buffer zone, measured in seconds.	1.4.96
sound-mode	string	speaker	No	Audio Output Method	1.4.96
auto-pause-if-navigate	boolean	true	No	When redirecting to another mini program page, should the real-time audio and video	1.4.96

©2013-2022 Tencent Cloud. All rights reserved.

				playback on the current page be automatically paused?	
auto-pause-if-open-native	boolean	true	No	When redirecting to another QQ native page, should the real-time audio and video playback on the current page be automatically paused?	1.4.96
enable-metadata	boolean	false	No	Should metadata be called back?	1.4.96
bindstatechange	eventhandle	-	No	Playback status change event, detail = {code}	1.4.96
bindfullscreenchange	eventhandle	-	No	Full screen change event, detail = {direction, fullScreen}	1.4.96
bindnetstatus	eventhandle	-	No	Network status notification, detail = {info}	1.4.96
bindmetadatachange	eventhandle	-	No	Metadata notification, detail = {info}	1.4.96

Valid values of mode

Value	Description	Earliest Version
live	Live stream	1.4.96

Valid values of orientation

Value	Description	Earliest Version
vertical	Vertical	1.4.96
horizontal	Horizontal	1.4.96

Valid values of object-fit

©2013-2022 Tencent Cloud. All rights reserved.

Value	Description	Earliest Version
contain	The longer dimension of the image fills the screen, while the shorter dimension is filled with black.	1.4.96
fillCrop	The image spans the entire screen, with portions exceeding the display area being cropped.	1.4.96

Valid values of sound-mode

Value	Description	Earliest Version
speaker	Speaker	1.4.96

Status Code

Code	Description
2001	Successfully connected to the server.
2002	Successfully connected to the server, initiating stream pull.
2003	The network has received the first video data packet (IDR).
2004	Video playback has commenced.
2006	Video playback has concluded.
2007	Video playback is loading.
2008	The decoder has been activated.
2009	The video resolution has been altered.
-2301	The network has been disconnected, and multiple reconnection attempts have proven futile. For further attempts, please manually restart the playback.
-2302	Acquisition of the accelerated streaming address has failed.
2101	The current video frame has failed to decode.
2102	The current audio frame has failed to decode.
2103	The network connection was disrupted. Automatic reconnection has been initiated.

©2013-2022 Tencent Cloud. All rights reserved.

2104	The network packet reception is unstable. This could be due to insufficient downstream bandwidth, or uneven streaming from the broadcaster's end.
2105	The current video playback is experiencing lag.
2106	Hardware decoding initiation has failed, and software decoding is being used.
2107	The current video frame is discontinuous, likely resulting in frame loss.
2108	The hardware decoding of the first I-frame in the current stream has failed. The SDK has automatically switched to software decoding.
3001	RTMP - DNS resolution has failed.
3002	Failed to connect to the RTMP server.
3003	Failed to establish a handshake with the RTMP server.
3005	RTMP read/write operation has failed.

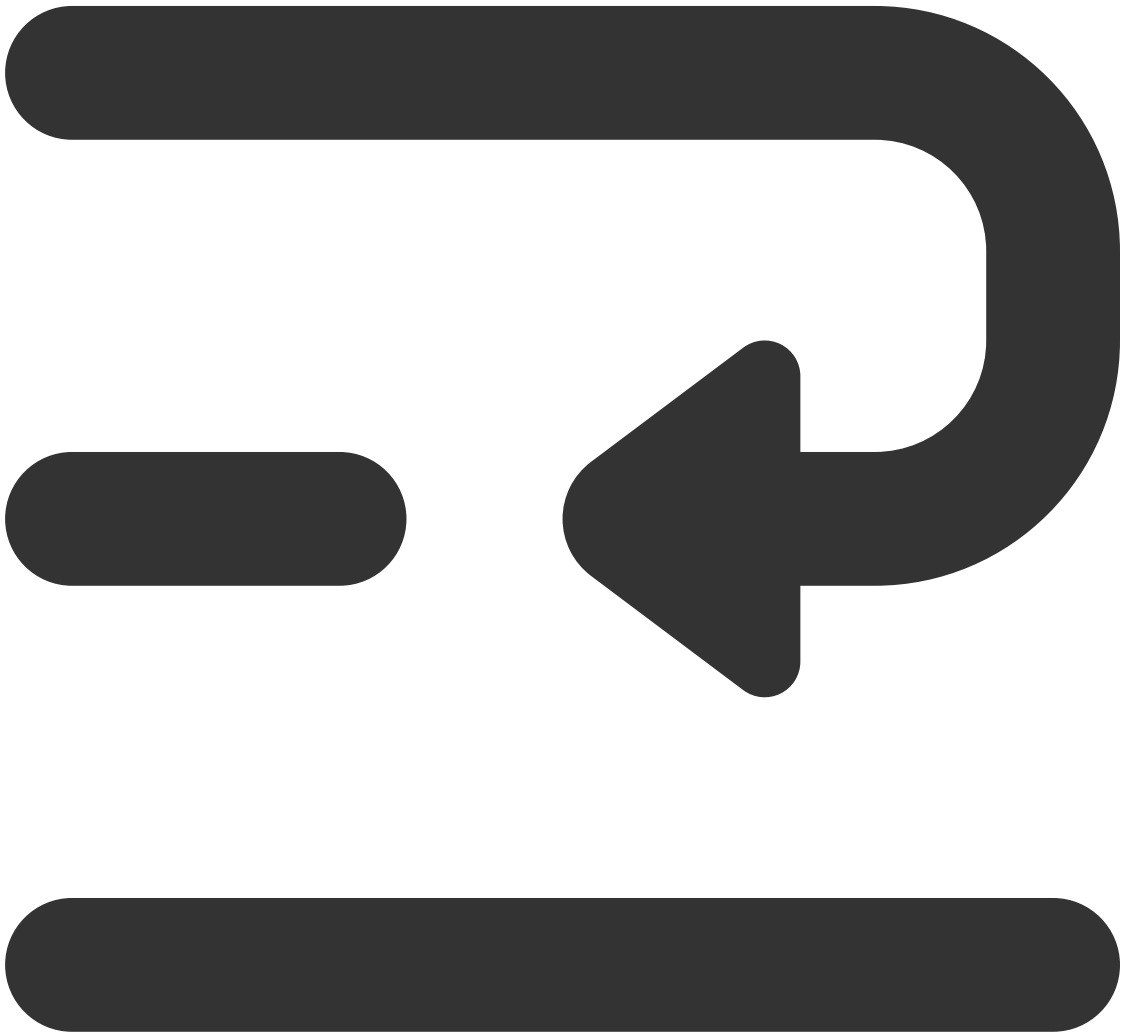
Network Status Data

Key Name	Description
videoBitrate	Current video data reception bitrate, measured in kilobits per second (kbps).
audioBitrate	Current audio data reception bitrate, measured in kilobits per second (kbps).
videoFPS	Current video frame rate
videoGOP	Current video GOP, which refers to the duration between two keyframes (I-frames), measured in seconds (s).
netSpeed	Current transmission/reception speed
netJitter	Network jitter conditions. The greater the jitter, the more unstable the network.
videoWidth	Video frame width
videoHeight	Video frame height

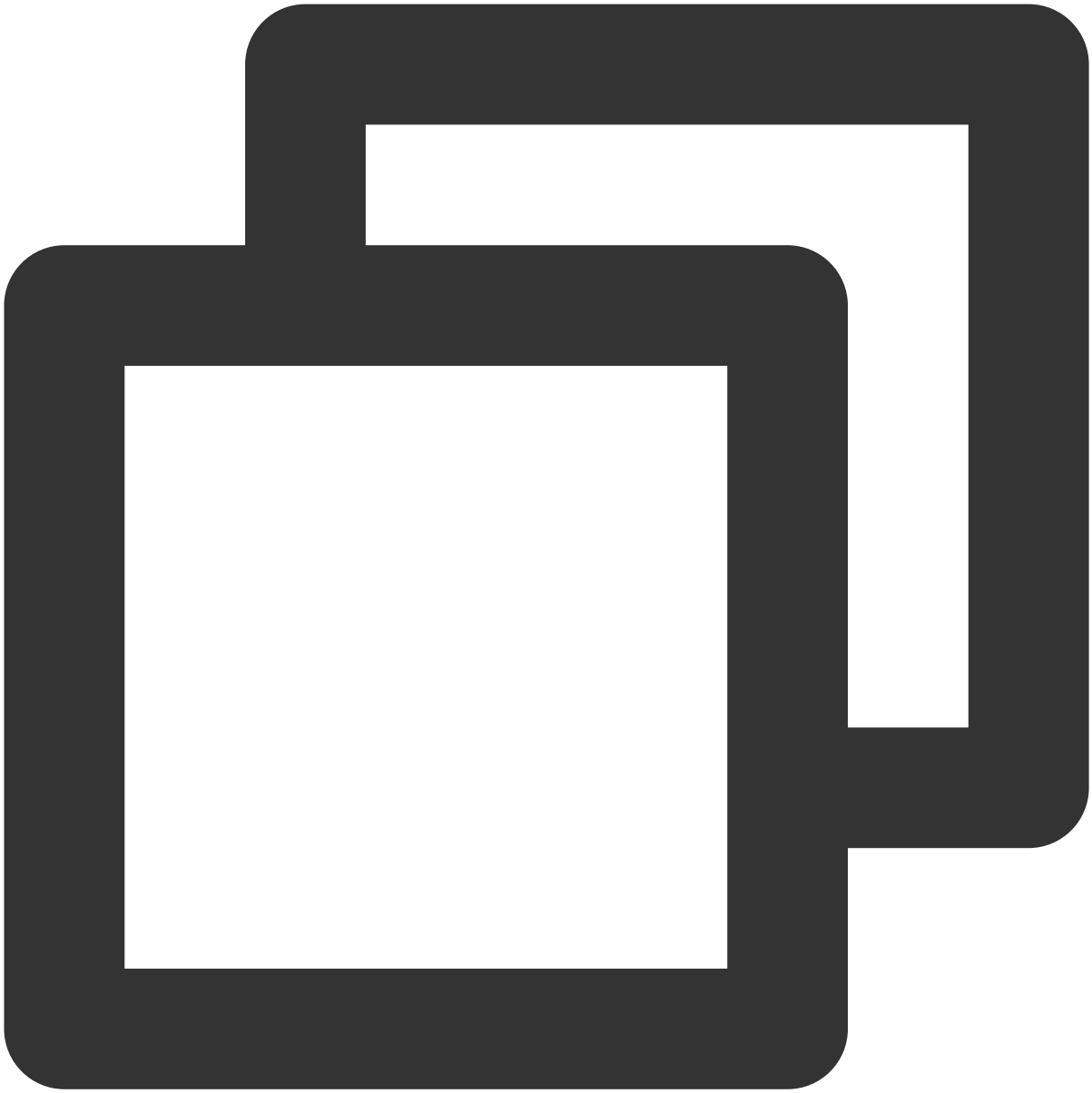
©2013-2022 Tencent Cloud. All rights reserved.

Sample Code

Corresponding WXML file



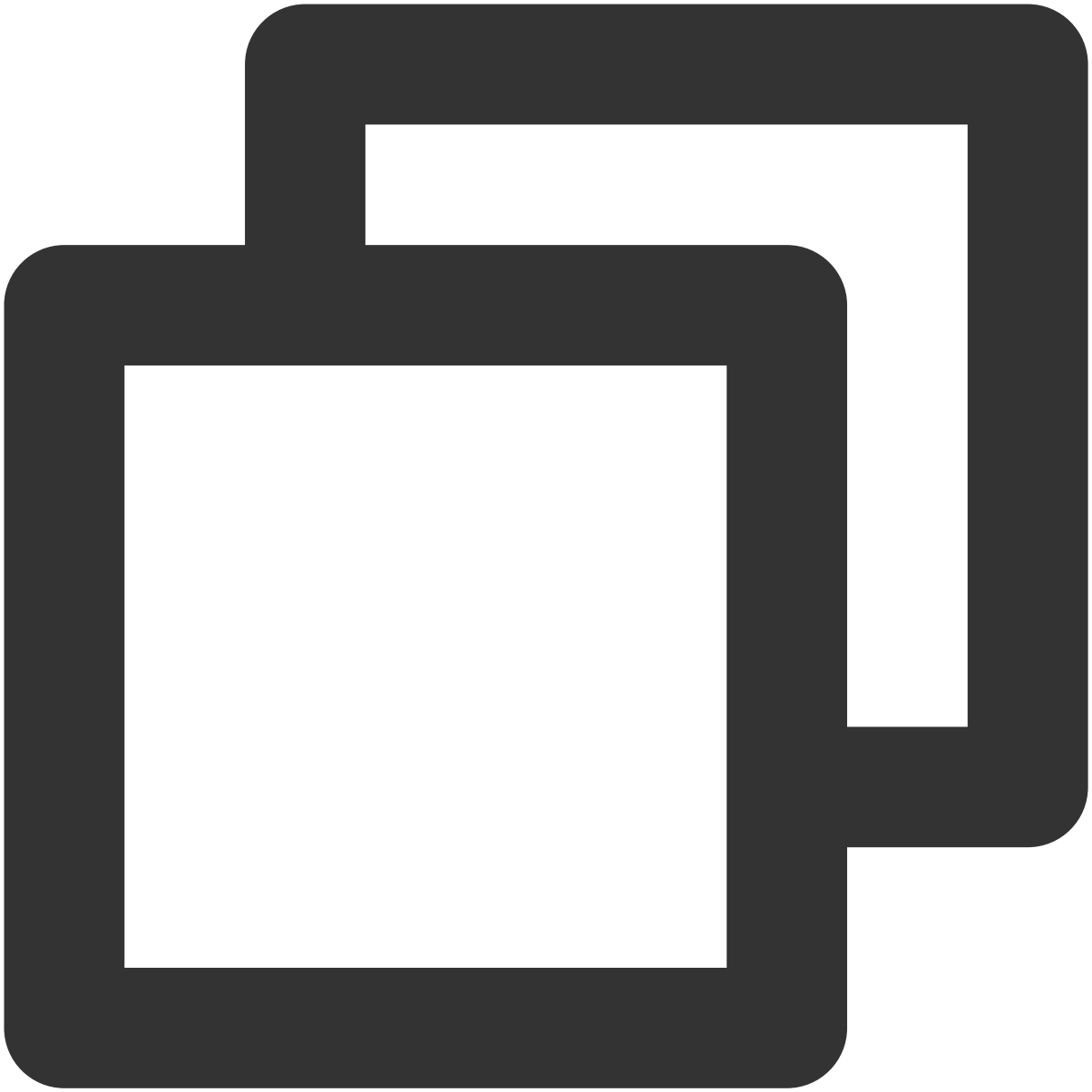
©2013-2022 Tencent Cloud. All rights reserved.



```
<live-player id="playerid" src="https://domain/pull_stream" mode="live" autoplay bi
```

Corresponding js file

©2013-2022 Tencent Cloud. All rights reserved.



```
Page({
  statechange(e) {
    console.log('live-player code:', e.detail.code)
  },
  error(e) {
    console.error('live-player error:', e.detail.errMsg)
  }
})
```

Note:

©2013-2022 Tencent Cloud. All rights reserved.

The default width and height of `live-player` are 300px and 225px respectively, which can be adjusted via wxss. Currently not supported on the developer tools.

live-pusher

Feature Description: Real-time audio and video recording (currently does not support same-layer rendering) requires "user authorization" for `scope.camera` and `scope.record` .

Parameter and Description:

Attribute	Type	Default value	Required	Description	Earliest Version
url	string	-	No	Streaming address. Currently, only rtmp is supported.	1.4.96
mode	string	RTC	No	SD (Standard Definition), HD (High Definition), FHD (Full High Definition), RTC (Real-Time Communication)	1.4.96
autopush	boolean	false	No	Automatic streaming	1.4.96
muted	boolean	false	No	Whether to mute	1.4.96
enable-camera	boolean	true	No	Turn on camera	1.4.96
auto-focus	boolean	true	No	Automatic aggregation	1.4.96
orientation	string	vertical	No	Screen Orientation	1.4.96
beauty	number	0	No	Beauty filter, with a value range of 0-9, where 0 indicates it is turned off.	1.4.96
whiteness	number	0	No	Whitening effect, with a value range of 0-9, where 0 indicates it is turned off.	1.4.96
aspect	string	9：16	No	Aspect ratio, with optional values of 3:4 and 9:16.	1.4.96
min-bitrate	number	200	No	Minimum bitrate	1.4.96
max-bitrat	number	1000	No	Maximum bitrate	1.4.96

©2013-2022 Tencent Cloud. All rights reserved.

audio-quality	string	height	No	High-quality audio (48KHz) or low-quality audio (16KHz), with values being 'high' or 'low'.	1.4.96
waiting-image	string	-	No	The standby screen displayed when streaming is initiated in the backend.	1.4.96
waiting-image-hash	string	-	No	MD5 value of the standby screen resource.	1.4.96
zoom	boolean	false	No	Focal length adjustment	1.4.96
device-position	string	-	No	Front or rear, with values being 'front' and 'back'.	1.4.96
background-mute	boolean	false	No	Mute status when entering the backend.	1.4.96
mirror	boolean	false	No	Setting whether the streaming screen is mirrored, with the effect reflected in the live-player.	1.4.96
bindstatechange	eventhandle	-	No	Status change event, detail = {code}	1.4.96
bindnetstatus	eventhandle	-	No	Network status notification, detail = {info}	1.4.96
binderror	eventhandle	-	No	Rendering error event, detail = {errMsg, errCode}.	1.4.96
bindbgmstart	eventhandle	-	No	Triggered when the background sound begins to play.	1.4.96
bindbgmprogress	eventhandle	-	No	Triggered when the background sound progress changes, detail = {progress, duration}.	1.4.96
bindbgmcomplete	eventhandle	-	No	Triggered when the play of background sound completes.	1.4.96

©2013-2022 Tencent Cloud. All rights reserved.

audio-reverb-type	number	0	No	Reverb mode (0-6): "Reverb off", "KTV", "Small room", "Large hall", "Deep", "Resonant", "Magnetic", respectively.	1.4.96

Valid values of orientation

Value	Description	Earliest Version
vertical	Vertical	1.4.96
horizontal	Horizontal	1.4.96

Error code (errCode)

Code	Description
10001	User has prohibited the use of the camera.
10002	User has prohibited the use of audio recording.
10003	Background sound resource (BGM) failed to load.
10004	Failed to load the waiting page resource (waiting-image).

Status code

Code	Description
1001	Successfully connected to the streaming server.
1002	Handshake with the server has been successfully completed, initiating streaming.
1003	Successfully activated the camera.
1004	Screen recording has been successfully initiated.
1005	Dynamically adjust streaming resolution.
1006	Dynamically adjust streaming bitrate.
1007	The first frame has been successfully captured.
1008	The encoder has been activated.
-1301	Failed to activate the camera.

©2013-2022 Tencent Cloud. All rights reserved.

-1302	Failed to activate the microphone.
-1303	Video encoding failed.
-1304	Audio encoding failed.
-1305	Unsupported video resolution.
-1306	Unsupported audio sampling rate.
-1307	Network disconnection has occurred, and multiple attempts to reconnect have proven ineffective. For further attempts, please manually restart the streaming.
-1308	Screen recording initiation has failed, possibly due to user rejection.
-1309	Screen recording has failed due to an unsupported Android system version. A system version of 5.0 or newer is required.
-1310	Screen recording has been interrupted by another application.
-1311	The Android Mic has been successfully activated. However, it is unable to record audio data.
-1312	The dynamic switch between landscape and portrait modes during screen recording has failed.
1101	Poor network conditions: Insufficient upstream bandwidth is causing obstruction to data upload.
1102	The network connection was disrupted. Automatic reconnection has been initiated.
1103	Hardware encoding initiation has failed, and software encoding is being used.
1104	Video encoding failed.
1105	The initiation of the new beauty software encoding has failed, and the previous software encoding is being used.
1106	The initiation of the new beauty software encoding has failed, and the previous software encoding is being used.
3001	RTMP - DNS resolution has failed.
3002	Failed to connect to the RTMP server.
3003	Failed to establish a handshake with the RTMP server.
3004	The RTMP server has actively disconnected. Please verify the legitimacy of the streaming address or the validity period of the anti-leech protection.

©2013-2022 Tencent Cloud. All rights reserved.

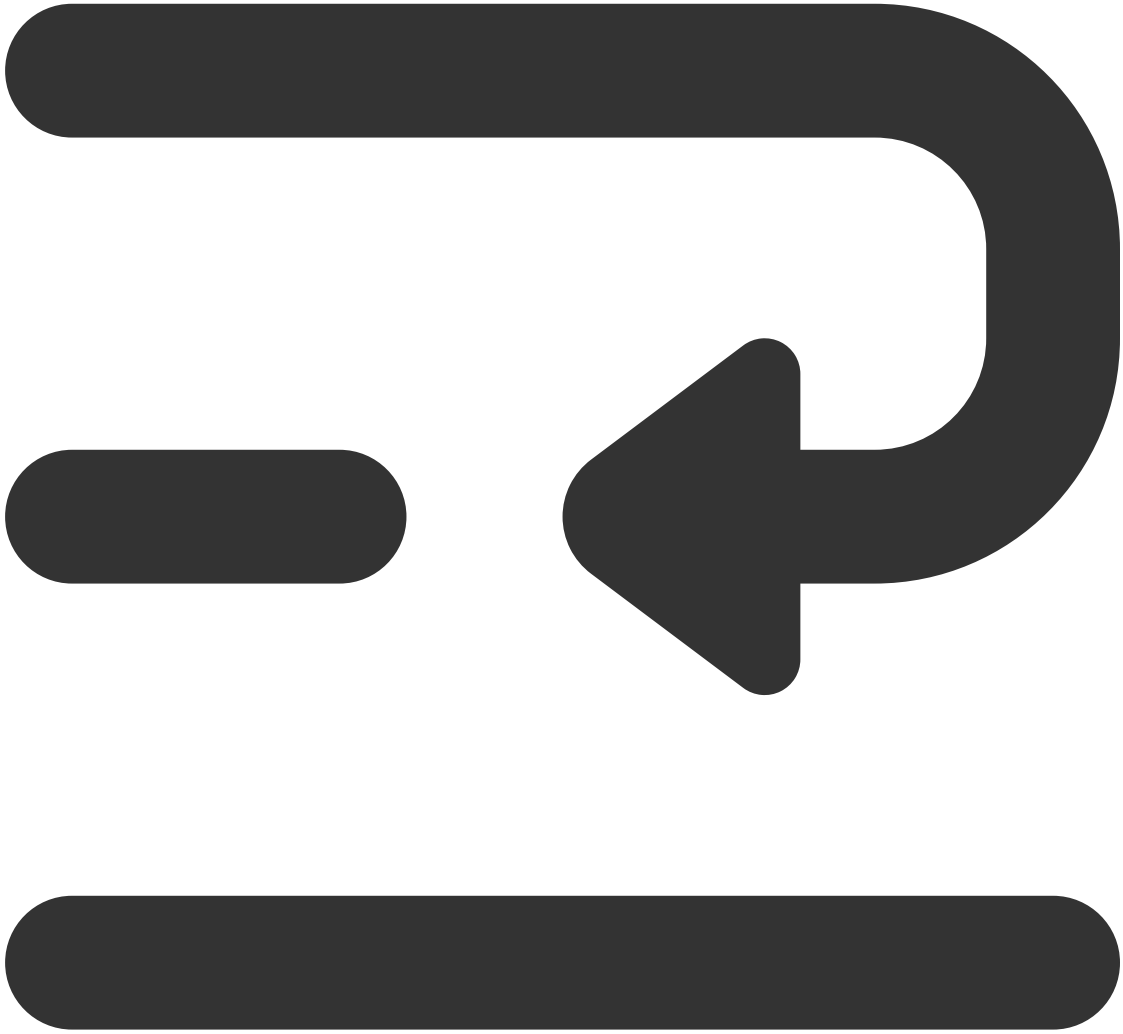
3005	RTMP read/write operation has failed.

Network Status Data (info)

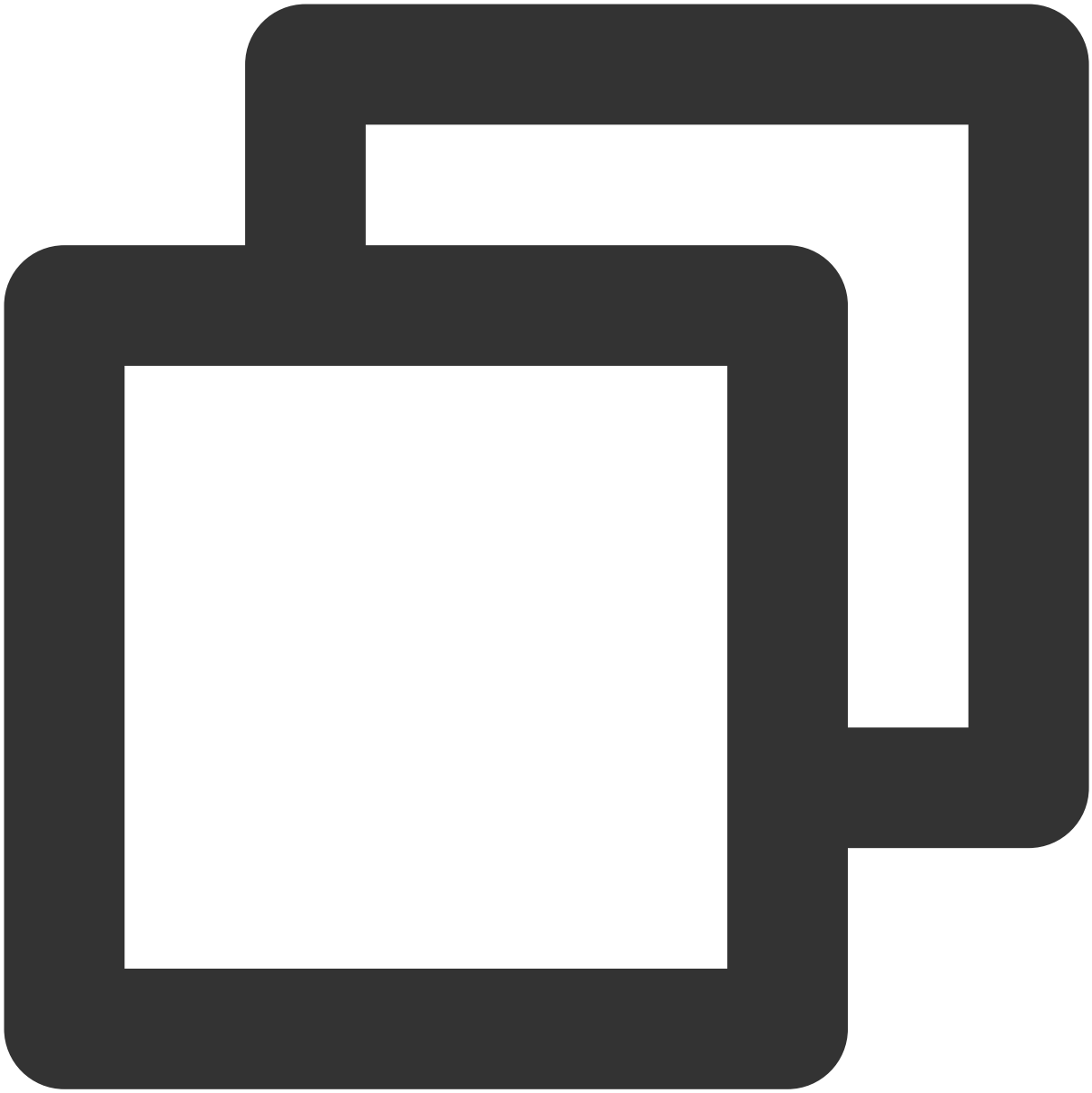
Key Name	Description
videoBitrate	The current output bitrate of the video encoder/decoder, measured in kilobits per second (kbps).
audioBitrate	The current output bitrate of the audio encoder/decoder, measured in kilobits per second (kbps).
videoFPS	Current video frame rate
videoGOP	Current video GOP, which refers to the duration between two keyframes (I-frames), measured in seconds (s).
netSpeed	Current transmission/reception speed
netJitter	Network jitter conditions. The greater the jitter, the more unstable the network.
videoWidth	Video frame width
videoHeight	Video frame height

Sample Code

©2013-2022 Tencent Cloud. All rights reserved.



©2013-2022 Tencent Cloud. All rights reserved.



```
  <live-pusher url="https://domain/push_stream" mode="RTC" autopush bindstatechange
Page({
  statechange(e) {
    console.log('live-pusher code:', e.detail.code)
  }
})
```

©2013-2022 Tencent Cloud. All rights reserved.

Open Capabilities

Last updated：2024-07-12 18:13:52

web-view

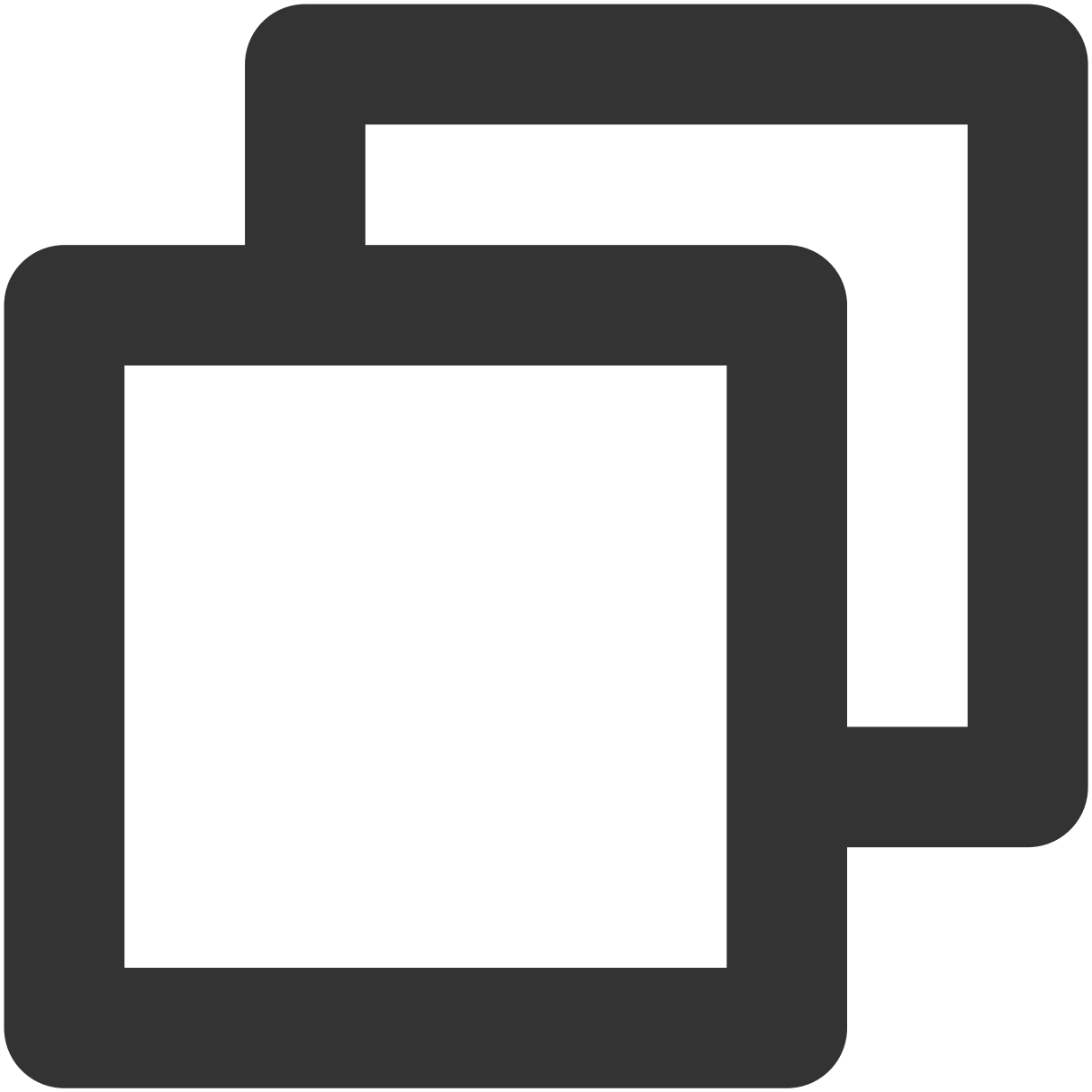
Feature Description: The web-view component serves as a container designed to accommodate web pages. It automatically fills the entire mini program page. Currently, this feature is not available for individual type mini programs. Corporate developers are required to select the service category on the developer platform first.

Parameter and Description:

Attribute	Type	Default value	Description
src	string	-	The webview points to a webpage link. The webpage requires configuration of the business domain after logging into the TMF Mini Program Management Backend.
bindload	eventhandler	-	When a webpage posts a message to the mini program, it triggers and receives messages at specific times (when the mini program backs out, the component is destroyed, or shared). e.detail = { data }, where data is an array composed of parameters from multiple postMessage instances.
binderror	eventhandler	-	This event is triggered when the webpage loads successfully. e.detail = { src }
binderror	eventhandler	-	This event is triggered when the webpage fails to load. e.detail = { src }

Sample Code

©2013-2022 Tencent Cloud. All rights reserved.



```
 <!-- wxml -->
<web-view src="https://www.qq.com/"></web-view>
```

Relevant interface 1

Within the `web-view`, the interface provided by JSSDK can be used to return to the mini program page. The supported interfaces include:

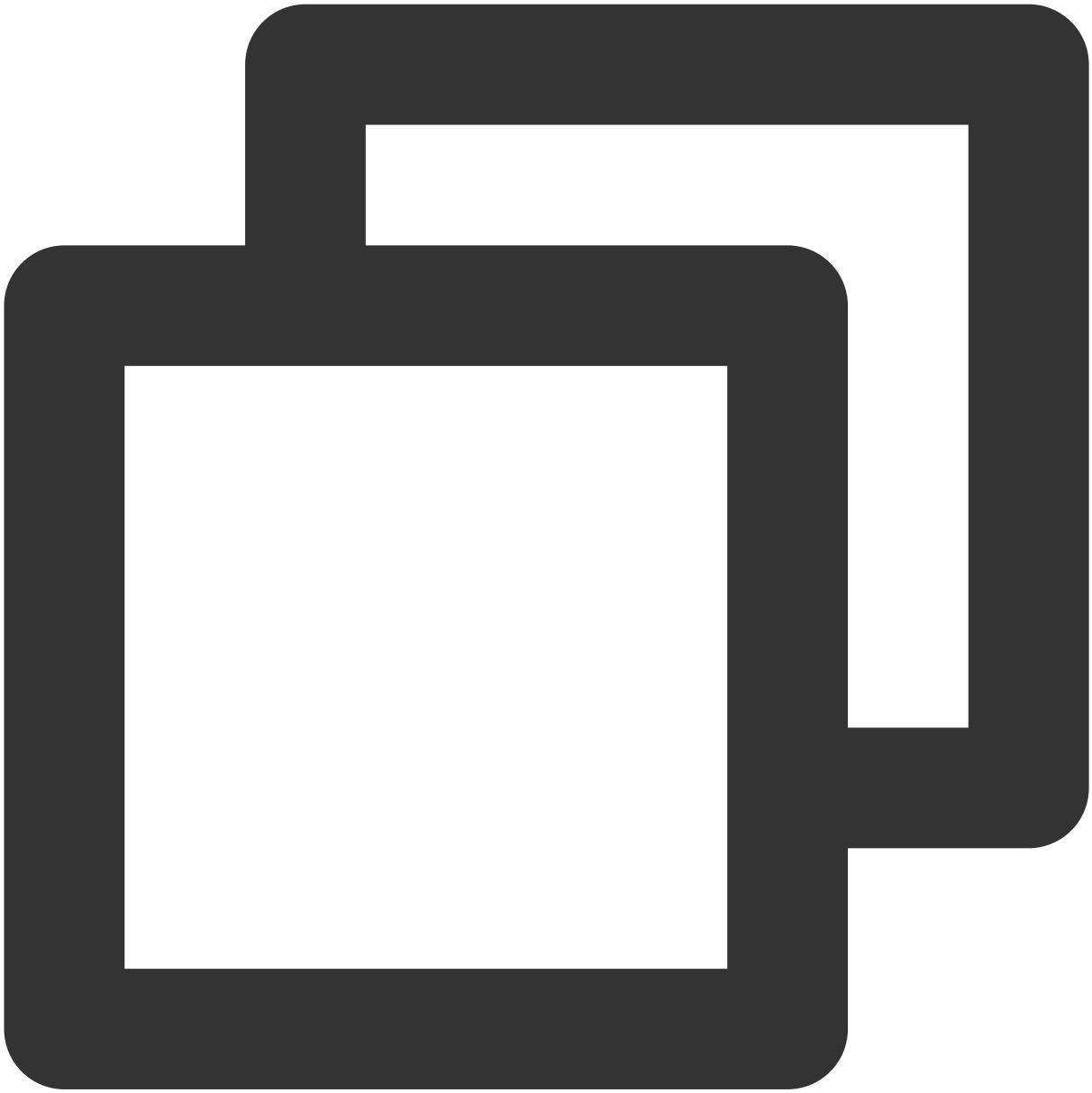
Interface Name	Description
wx.miniProgram.navigateTo	The parameters are consistent with the mini program interface.

©2013-2022 Tencent Cloud. All rights reserved.

wx.miniProgram.navigateBack	The parameters are consistent with the mini program interface.
wx.miniProgram.switchTab	The parameters are consistent with the mini program interface.
wx.miniProgram.reLaunch	The parameters are consistent with the mini program interface.
wx.miniProgram.redirectTo	The parameters are consistent with the mini program interface.
wx.miniProgram.postMessage	Sending a message to the mini program will trigger the component's message event at the following specific times: when the mini program is backed out, when the component is terminated, when sharing, and when copying the link.
wx.miniProgram.getEnv	Obtaining the current environment

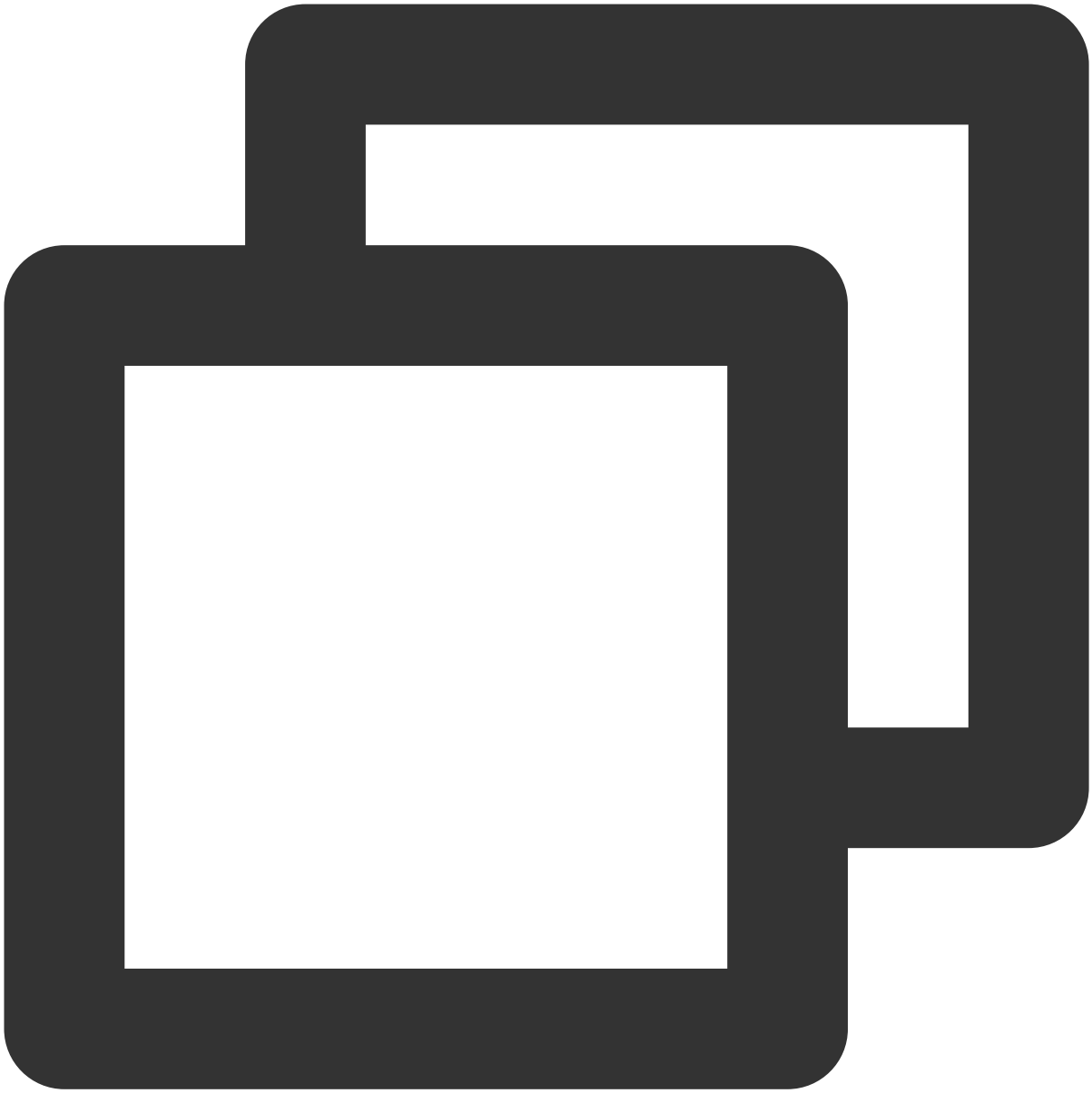
Sample Code

©2013-2022 Tencent Cloud. All rights reserved.



```
<!-- html -->
<script type="text/javascript" src="https://qqq.gtimg.cn/miniprogram/webview_jssdk/
```

©2013-2022 Tencent Cloud. All rights reserved.



```
wx.miniProgram.navigateTo({url: '/path/to/page'})
wx.miniProgram.postMessage({ data: 'foo' })
wx.miniProgram.postMessage({ data: {foo: 'bar'} })
wx.miniProgram.getEnv(function(res) { console.log(res.miniprogram) })
```

Relevant interface 2

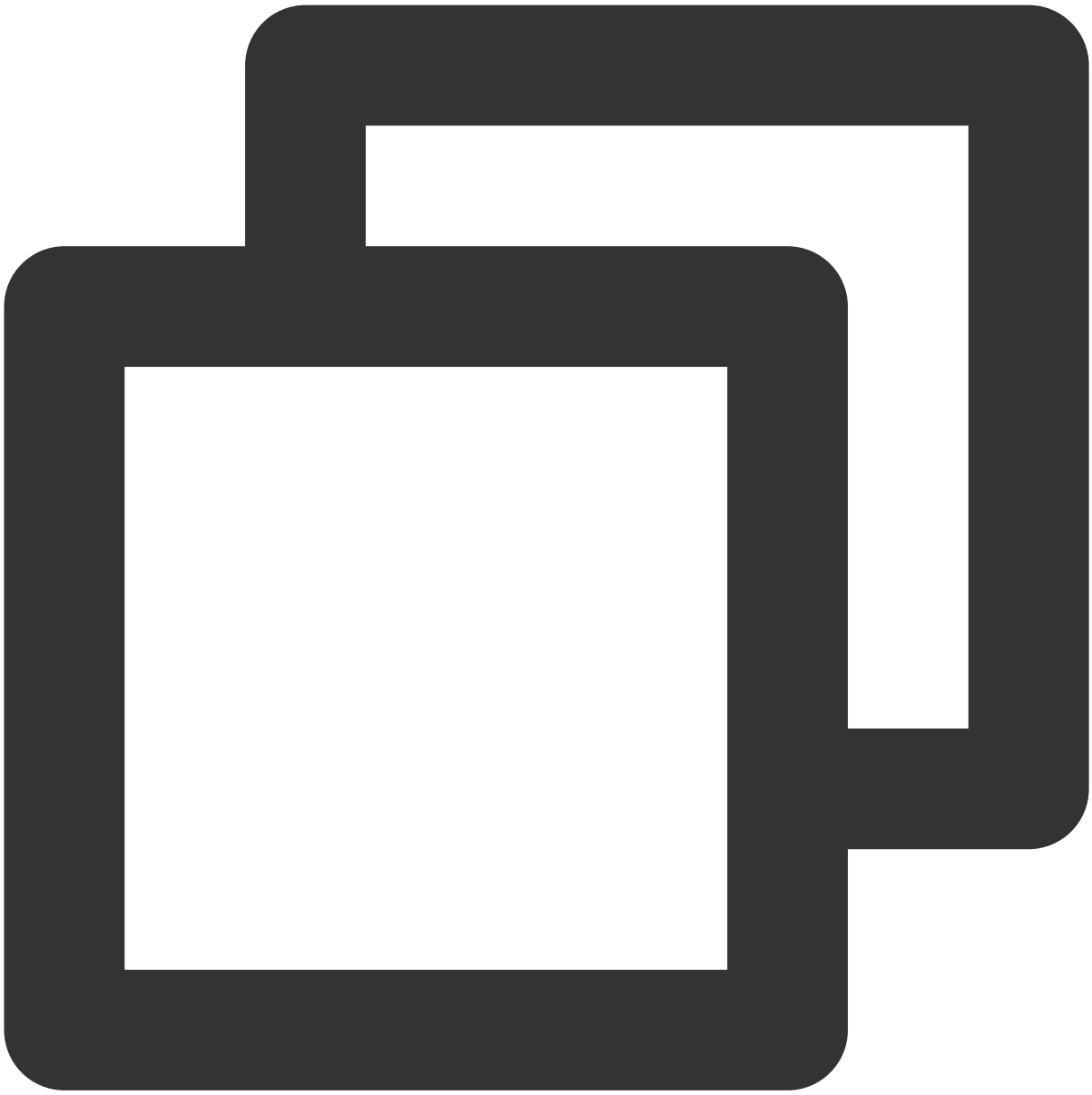
`<web-view>` page currently does not support other media-related interfaces.

Relevant interface 3

©2013-2022 Tencent Cloud. All rights reserved.

During user sharing, the current `<web-view>` URL can be obtained, that is, the `webViewUrl` parameter is returned in the `onShareAppMessage` callback.

Sample Code



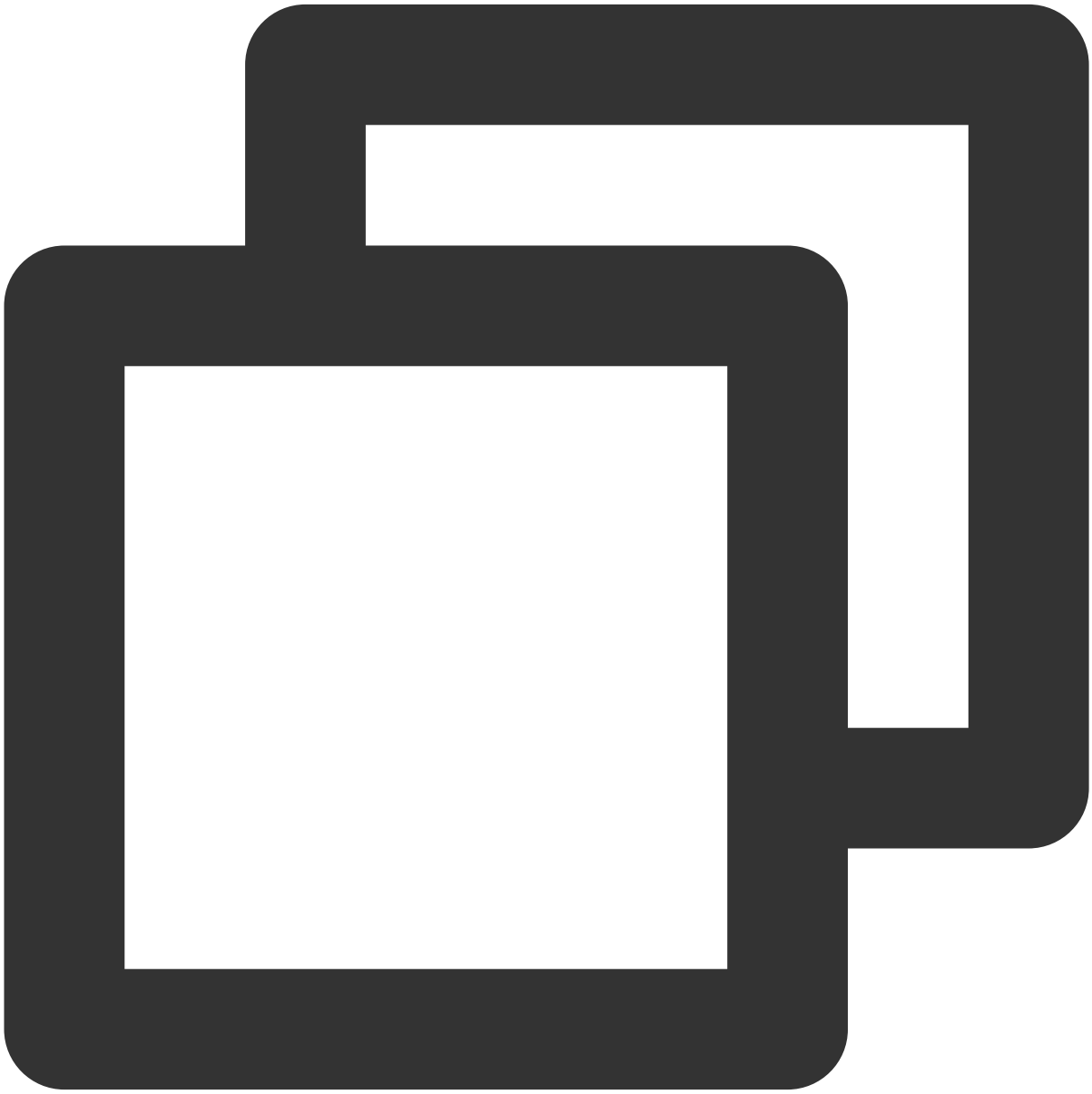
```
Page({
  onShareAppMessage(options) {
    console.log(options.webViewUrl)
  },
})
```

Relevant interface 4

©2013-2022 Tencent Cloud. All rights reserved.

Within the webpage, the `window.__wxjs_environment` variable can be used to determine whether it is in a mini program environment. It is recommended to use it in the `WeixinJSBridgeReady` callback, or the `getEnv` interface provided by JSSDK can also be used.

Sample Code



```
// Within the web-view page
function ready() {
  console.log(window.__wxjs_environment === 'miniprogram') // true
}
if (!window.QQJSBridge || !QQJSBridge.invoke) {
  document.addEventListener('WeixinJSBridgeReady', ready, false)
```

©2013-2022 Tencent Cloud. All rights reserved.

```
} else {
  ready()
}

// Or
wx.miniProgram.getEnv(function(res) {
  console.log(res.miniprogram) // true
})
```

Relevant interface 5

The mini program web-view environment can be determined by checking if the term 'miniProgram' is included in the userAgent.

Note:

The domain of the iframe within the webpage also needs to be configured into the domain allowlist.

On the developer tool, debugging can be initiated on the <web-view> component by right-clicking and selecting 'Debug'.

Each page can only contain one <web-view>. The <web-view> will automatically fill the entire page and overlay other components.

Communication between the <web-view> webpage and the mini program is not supported, except for the interfaces provided by JSSDK.

To prevent blank screen issues when opening on iOS, it is recommended to avoid including Chinese characters in the link and to use encodeURIComponent.

When the prompt 'insertHTMLWebView:fail no permission' appears, it indicates that there is no permission to use <web-view>. Currently, individual developers and corporate developers who have not selected a service category cannot be called.

©2013-2022 Tencent Cloud. All rights reserved.

Canvas

Last updated：2024-07-12 18:13:37

Feature Description: Canvas. This component is a native component, so mind the relevant restrictions when using it.

Note:

For related APIs, see wx.createcameracontext.

Avoid setting excessively large width and height values, as this may cause crash issues in the Android environment.

Attribute name	Type	Default value	Note
canvas-id	string	-	Unique identifier of the canvas component
disable-scroll	boolean	false	When moving within the canvas and a gesture event is bound, screen scrolling and pull-down refresh are prevented.
bindtouchstart	eventhandle	-	Initiation of finger touch action
bindtouchmove	eventhandle	-	Movement after finger touch
bindtouchend	eventhandle	-	Termination of finger touch action
bindtouchcancel	eventhandle	-	Finger touch action is interrupted, such as by an incoming call alert.
bindlongtap	eventhandle	-	Upon a finger's prolonged contact of 500 ms, a long-press event is triggered. Following this, movement will not trigger screen scrolling.
binderror	eventhandle	-	Upon encountering an error, an 'error' event is triggered, with detail = {errMsg: 'something wrong'}.

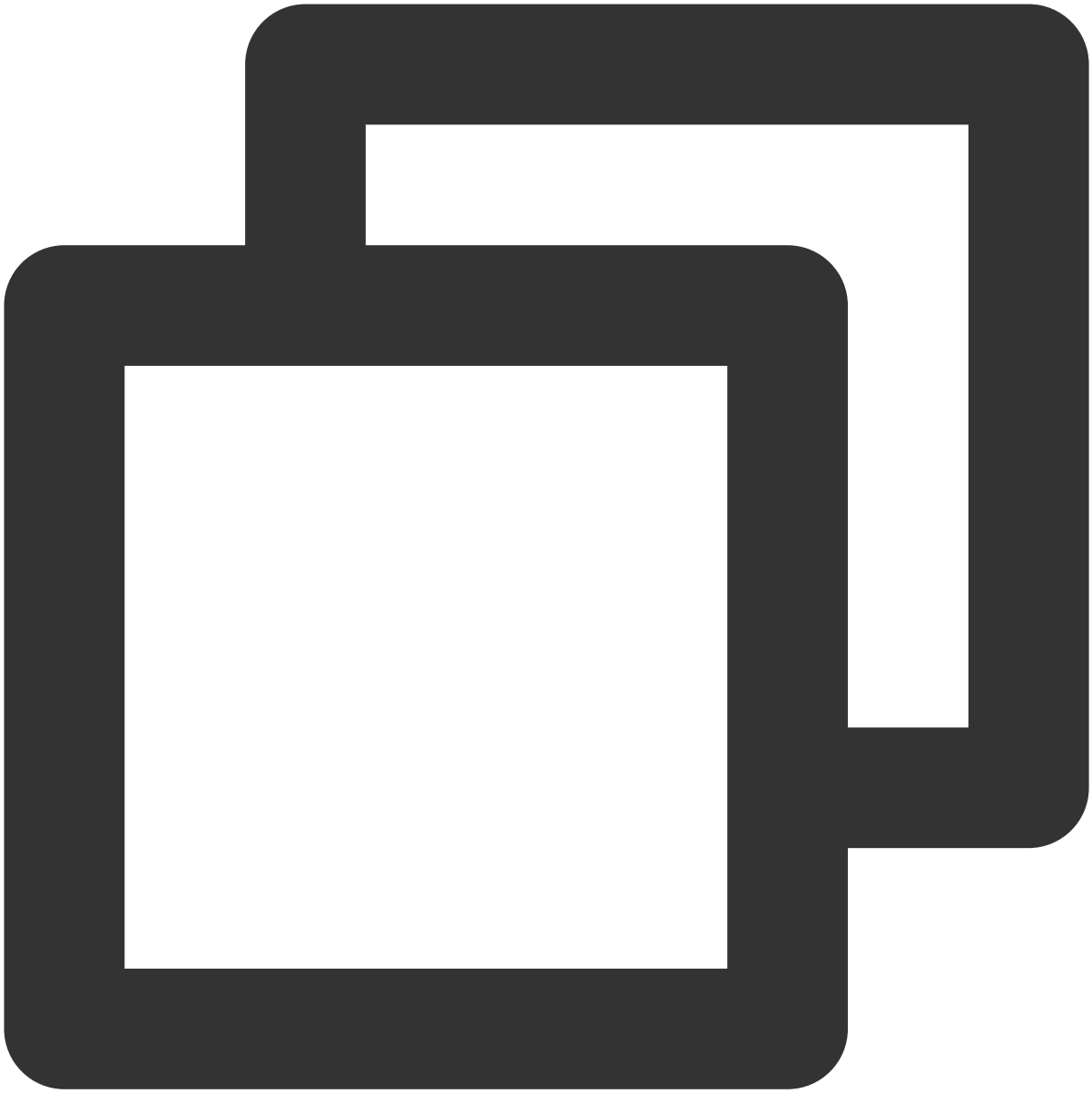
Note:

The default dimensions of the canvas tag are 300 px in width and 150 px in height.

Within the same page, the canvas-id must be unique. If a previously used canvas-id is employed, the corresponding canvas tag's canvas will be hidden and cease to function properly.

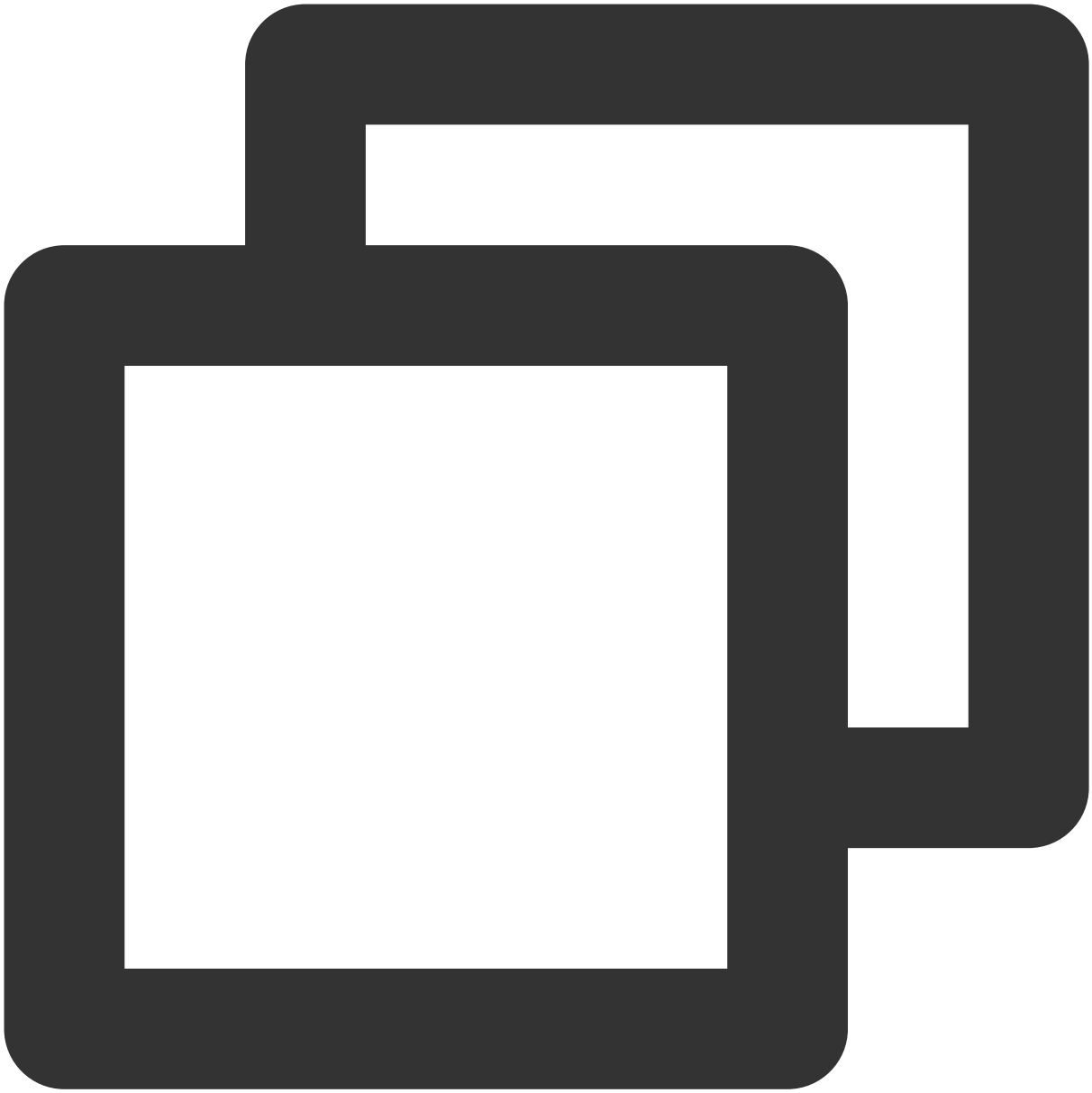
Sample Code

©2013-2022 Tencent Cloud. All rights reserved.



```
<!-- canvas.wxml -->
<canvas style="width: 300px; height: 200px;" canvas-id="firstCanvas"></canvas>
<!-- When absolute positioning is used, the display level of the canvas following t
<canvas style="width: 400px; height: 500px;" canvas-id="secondCanvas"></canvas>
<!-- As the canvas-id is duplicated from the previous canvas, this canvas will not
<canvas
  style="width: 400px; height: 500px;"
  canvas-id="secondCanvas"
  binderror="canvasIdErrorCallback"
></canvas>
```

©2013-2022 Tencent Cloud. All rights reserved.



```
// canvas.js
Page({
  canvasIdErrorCallback(e) {
    console.error(e.detail.errMsg)
  },
  onReady(e) {
    // Use wx.createContext to obtain the drawing context
    const context = wx.createCanvasContext('firstCanvas')

    context.setStrokeStyle('#00ff00')
    context.setLineWidth(5)
```

©2013-2022 Tencent Cloud. All rights reserved.

```
    context.rect(0, 0, 200, 200)
    context.stroke()
    context.setStrokeStyle('#ff0000')
    context.setLineWidth(2)
    context.moveTo(160, 100)
    context.arc(100, 100, 60, 0, 2 * Math.PI, true)
    context.moveTo(140, 100)
    context.arc(100, 100, 40, 0, Math.PI, false)
    context.moveTo(85, 80)
    context.arc(80, 80, 5, 0, 2 * Math.PI, true)
    context.moveTo(125, 80)
    context.arc(120, 80, 5, 0, 2 * Math.PI, true)
    context.stroke()
    context.draw()
  }
})
```

©2013-2022 Tencent Cloud. All rights reserved.

Navigation Component

Last updated：2024-07-12 18:13:33

navigator

Feature Description: Page link

Parameter and Description

Attribute	Type	Default value	Required	Description
target	string	self	-	Specifies the target where the redirection occurs, with the default being the current mini program. The optional values are 'self' or 'miniProgram'.
url	string	-	No	Redirection within the current mini program
open-type	string	navigate	No	Redirection method
path	string	-	-	When 'target' is set to 'miniProgram', this specifies the path of the page to be opened. If left blank, the home page will be opened.
extra-data	object	-	-	When 'target' is set to 'miniProgram', this represents the data to be passed to the target mini program. The target mini program can get this data through **App.onLaunch()** or **App.onShow().**
hover-class	string	navigator-hover	-	Specifies the style class when clicked. When
hover-stop-propagation	boolean	false	-	Specifies whether to prevent the ancestor nodes of this node from appearing in the click status.
hover-start-time	number	50	-	The duration before the click status appears after pressing, measured in milliseconds.
hover-stay-time	number	600	-	The retention time of the click status after the finger is released, measured in milliseconds.
bindsuccess	string	-	-	When 'target' is set to 'miniProgram', this indicates a successful transition to the mini

©2013-2022 Tencent Cloud. All rights reserved.

				program.
bindfail	string	-	-	When 'target' is set to 'miniProgram', this indicates a failed transition to the mini program.
bindcomplete	string	-	-	When 'target' is set to 'miniProgram', this indicates a completed transition to the mini program.
aria-label	string	-	-	Accessibility, (Attribute) Additional description of the element.

Valid values of 'open-type':

Valid Values	Description
navigate	Corresponds to the functionality of 'wx.navigateTo' or 'wx.navigateToMiniProgram'.
redirect	Corresponds to the functionality of 'wx.redirectTo'.
switchTab	Corresponds to the functionality of 'wx.switchTab'.
reLaunch	Corresponds to the functionality of 'wx.reLaunch'.
navigateBack	Corresponds to the functionality of 'wx.navigateBack'.

Usage Limitations: User confirmation is required for redirection.

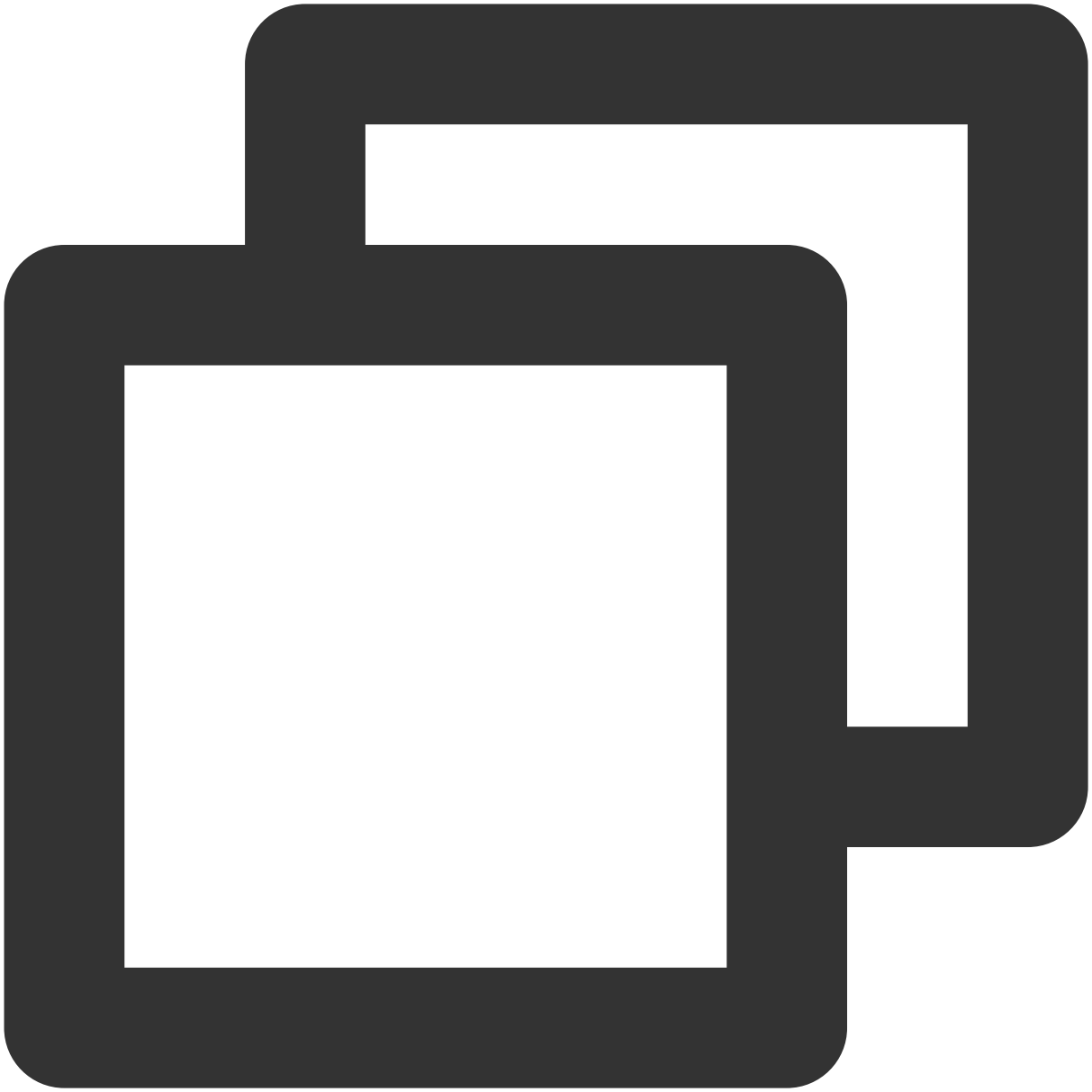
Before redirecting to another mini program, a unified pop-up window will be added, asking whether to redirect. The user can only redirect to another mini program after confirmation. If the user clicks cancel, the callback `fail cancel` will be triggered.

Note:

`navigator-hover` defaults to `{background-color: rgba(0, 0, 0, 0.1); opacity: 0.7;}`. The background color of the <navigator></2> child nodes should be transparent.

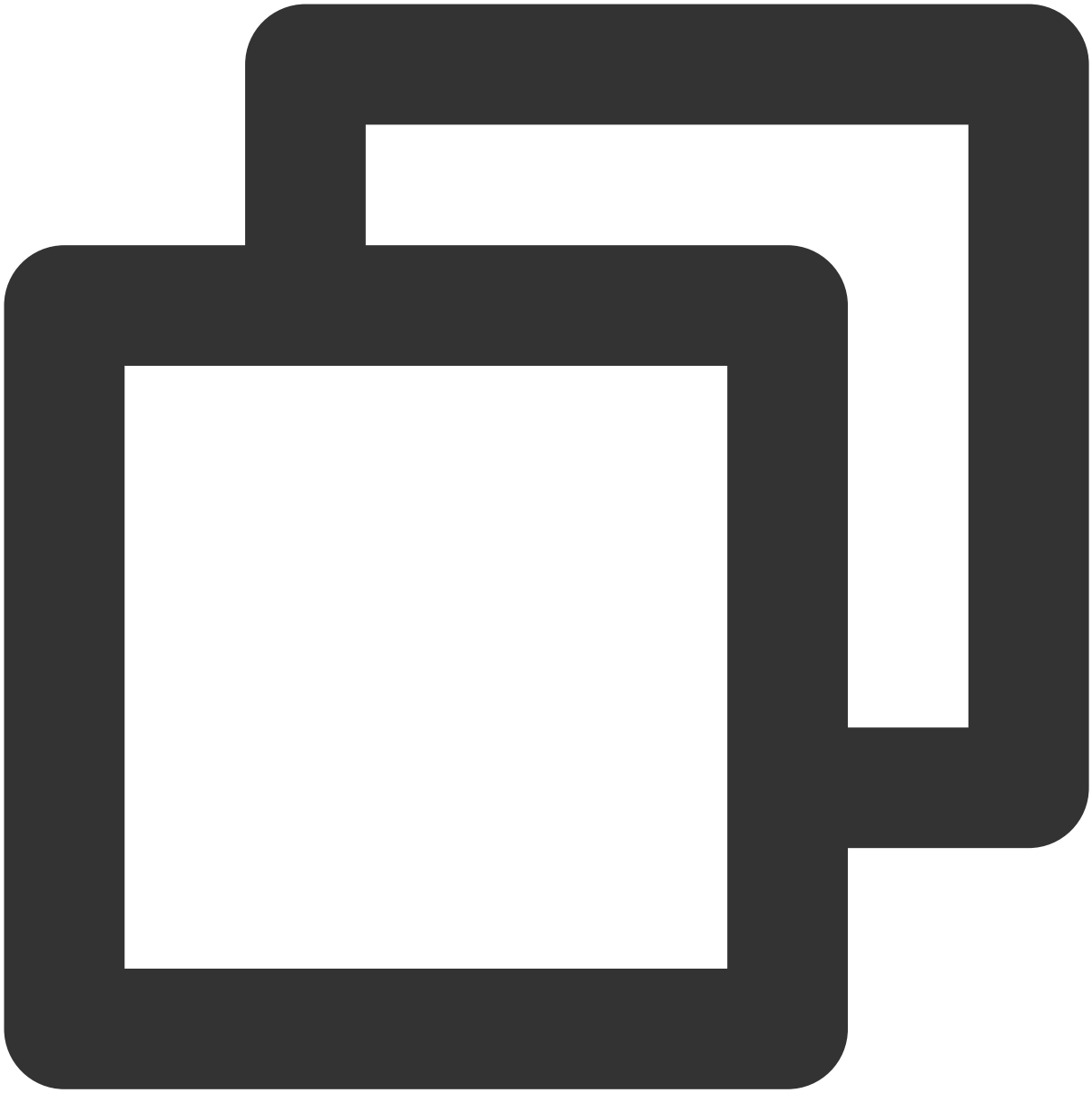
Sample Code

©2013-2022 Tencent Cloud. All rights reserved.



```
/ wxss /
/ Modify the default navigator click status /
.navigator-hover {
  color: blue;
}
/ Customize other click status style classes /
.other-navigator-hover {
  color: red;
}
```

©2013-2022 Tencent Cloud. All rights reserved.

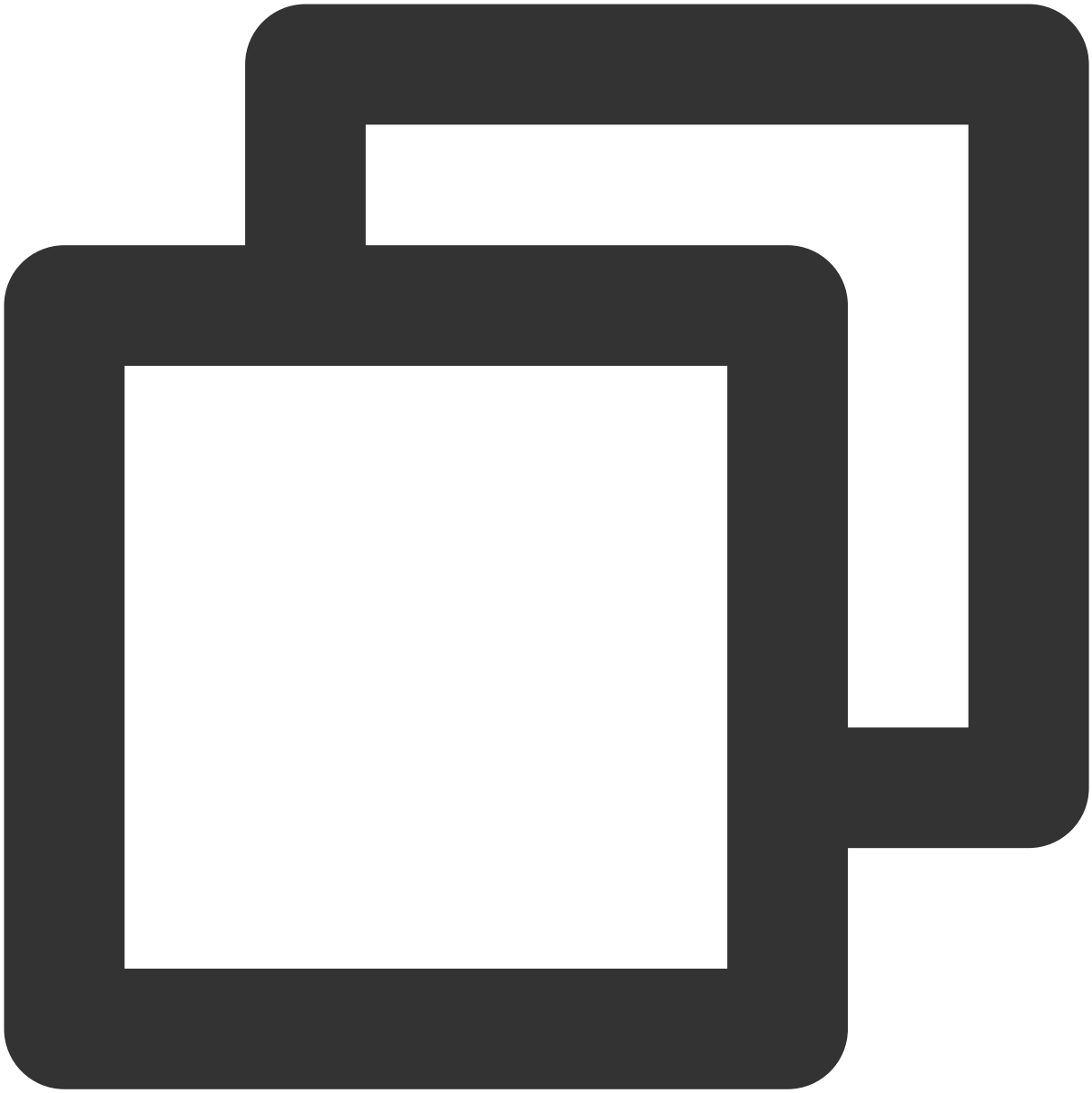


```
<!-- sample.wxml -->
<view class="btn-area">
  <navigator
    url="/page/navigate/navigate?title=navigate"
    hover-class="navigator-hover"
  >
    Redirect to a new page
  </navigator>
  <navigator
    url="../../redirect/redirect/redirect?title=redirect"
    open-type="redirect"
```

©2013-2022 Tencent Cloud. All rights reserved.

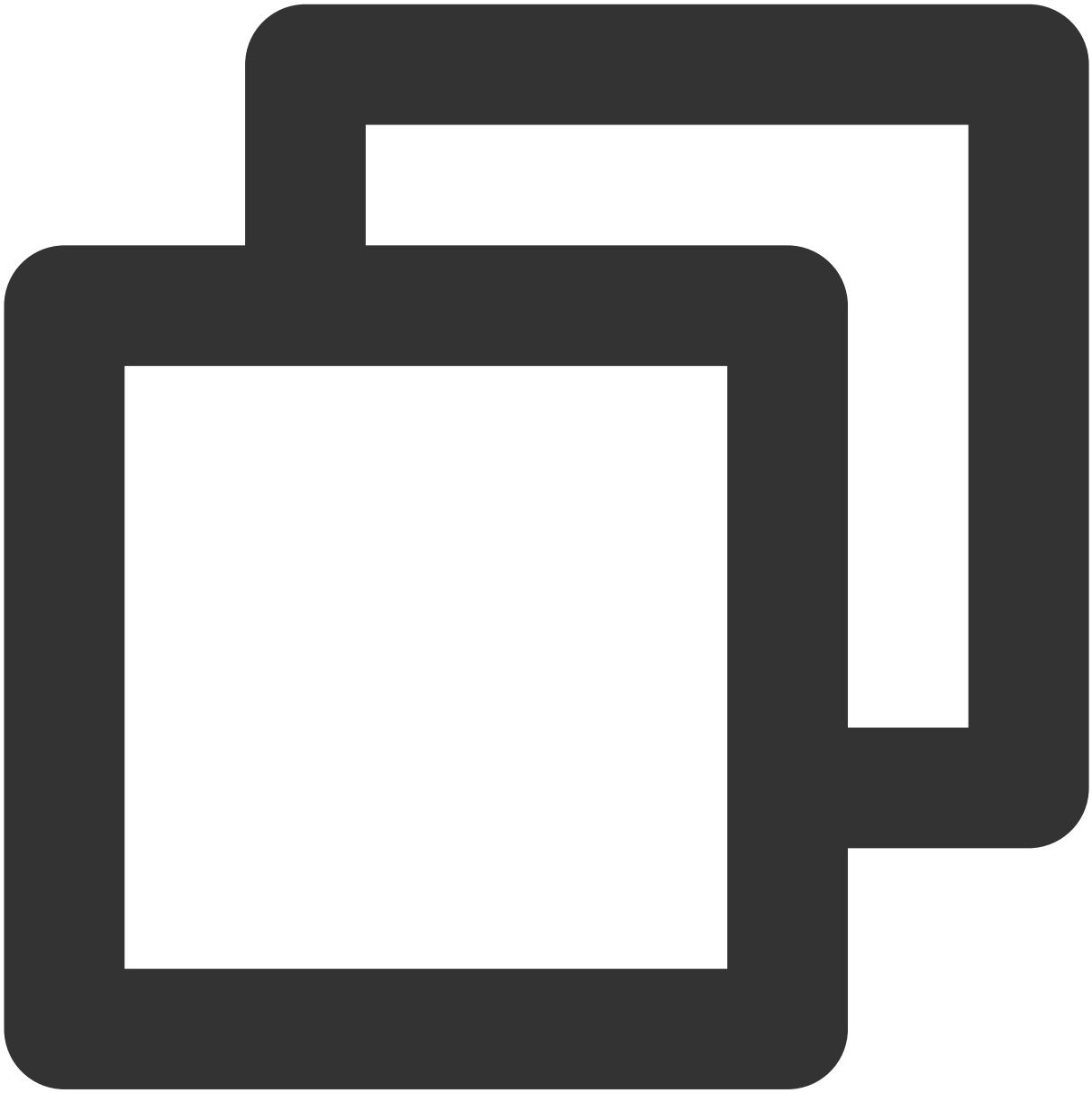
```
    hover-class="other-navigator-hover"
  >
    Open on the current page
  </navigator>
  <navigator
    url="/page/index/index"
    open-type="switchTab"
    hover-class="other-navigator-hover"
  >
    Switch Tab
  </navigator>
</view>
```

©2013-2022 Tencent Cloud. All rights reserved.



```
<!-- navigator.wxml -->
<view style="text-align:center">{{title}}</view>
<view>Click the top left corner to return to the previous page</view>
```

©2013-2022 Tencent Cloud. All rights reserved.



```
// redirect.js navigator.js
Page({
  onLoad(options) {
    this.setData({
      title: options.title
    })
  }
})
```

©2013-2022 Tencent Cloud. All rights reserved.

Form Component

Last updated：2024-07-12 18:13:28

button

Feature Description: Button.

Parameter and Description:

Attribute name	Type	Valid Values and Descriptions	Default value	Required	Note
size	string	Default: Default size Mini: Small dimensions	default	No	Dimensions of the button
type	string	Primary: Green Default: White Warn: Red	default	No	Aesthetic design of buttons
plain	boolean	-	false	No	The button's hollow design and background color
disabled	boolean	-	false	No	Whether to disable
loading	boolean	-	false	No	Presence of a loading icon preceding the name
form-type	string	Submit: Submit the form Reset: Reset the form	-	No	Used within the form component, a click will respectively trigger the submit/reset events of the form component.
hover-class	string	-	button-hover	No	Specifies the style class when pressing the button. When `hover-class="none"` , there is no click status effect.
hover-start-time	number	-	20	No	The duration, in milliseconds, before the click-status appears upon holding down.

©2013-2022 Tencent Cloud. All rights reserved.

hover-stay-time	number	-	70	No	The duration, in milliseconds, for which the click-status is retained after the finger is released.

checkbox

Feature Description: Multiple selection items.

Parameter and Description:

Attribute name	Type	Default value	Required	Note
value	string	-	No	Checkbox identifier, when selected, triggers the change event of the checkbox-group and carries the value of the checkbox.
disabled	boolean	false	No	Whether to disable
checked	boolean	false	No	Indicates whether it is currently selected, which can be used to set the default selection.
color	string	#09BB07	No	The color of the checkbox, identical to the color in CSS.

checkbox-group

Feature Description: A multi-item selector, internally composed of multiple checkboxes.

Attribute description:

Attribute name	Type	Default value	Required	Note
bindchange	eventhandle	-	No	When the selected item in the checkbox-group changes, it triggers the change event, detail = {value: [array of the selected checkbox's value]}.

form

©2013-2022 Tencent Cloud. All rights reserved.

Feature Description: Form, used to submit the user's inputs within the component, namely the switch, input, checkbox, slider, radio, and picker.

Upon clicking the button component in the form with a form-type of "submit", the value in the form component will be submitted, necessitating the addition of a name in the form component to serve as a key.

Parameter and Description

Attribute name	Type	Default value	Required	Note
bindsubmit	eventhandle	-	No	Carries the data in the form to trigger the "submit" event, event.detail = {value : {'name': 'value'}, formId: ''}.
bindreset	eventhandle	-	No	The reset event is triggered when the form is reset.

input

Feature Description: Input box. This component is a native component, so mind the relevant restrictions when using it.

Parameter and Description

Attribute name	Type	Valid Values and Descriptions	Default value	Required	Note
value	string	-	-	Yes	The initial content of the input box.
type	string	Text: Text input keyboard. Number: Numeric input keyboard. ID Card: ID number input keyboard. Digit: Numeric keyboard with decimal point.	text	No	Type of input.
password	boolean	-	false	No	Is it a password type?

©2013-2022 Tencent Cloud. All rights reserved.

placeholder	string	-		Yes	Placeholder when the input field is empty.
placeholder-style	string	-	-	Yes	Specify the style of the placeholder.
placeholder-class	string	-	input-placeholder	No	Specify the style class of the placeholder.
disabled	boolean	-	false	No	Whether to disable
maxlength	number	-	140	No	Maximum input length, unrestricted when set to -1.
cursor-spacing	number	-	0	No	Specify the distance between the cursor and the keyboard, taking the minimum value of the distance from the input to the bottom and the distance specified by cursor-spacing as the distance between the cursor and the keyboard.
auto-focus	boolean	-	false	No	(About to be deprecated, please use focus directly) Auto-focus, pull up the keyboard.
focus	boolean		false	No	Get focus.
bindinput	eventhandle	-	-	Yes	Triggered upon keyboard inputs, event.detail = {value, cursor, keyCode}. keyCode is the key value, supported from version 2.1.0 onwards. The handler function can directly return a string, which will replace the content of the input box.
bindfocus	eventhandle	-	-	Yes	Triggered when the input box is focused, event.detail = { value, height }. Height represents the keyboard height, supported from base library version 1.9.90 onwards.

©2013-2022 Tencent Cloud. All rights reserved.

bindblur	eventhandle	-	-	Yes	Triggered when the input box loses focus, event.detail = { value, encryptedValue, encryptError }.
bindconfirm	eventhandle	-	-	Yes	Triggered when **Done** is clicked, event.detail = { value }.

label

Feature Description: Used to enhance the usability of form components.
Use the 'for' attribute to find the corresponding id, or place the control under this tag. When clicked, it will trigger the corresponding control. 'For' has a higher priority than internal controls. When there are multiple controls internally, the first control is triggered by default. The controls that can currently be bound are: button, checkbox, radio, switch, input.

Parameter and Description:

Attribute name	Type	Default value	Required	Note
for	string	-	No	Bind with control ID

picker

Feature Description: A scrolling selector that pops up from the bottom.

Attribute description:

Attribute name	Type	Valid Values and Descriptions	Default value	Required	Note
mode	string	Selector: Regular selector. multiSelector: Multi-column selector. Time: Time selector. Date: Date selector. Region: Province, City, and District selector.	selector	No	Selector Type
disabled	boolean	-	false	No	Whether to disable

©2013-2022 Tencent Cloud. All rights reserved.

In addition to the aforementioned universal attributes, the `picker` possesses distinct attributes for different modes.

Regular Selector: mode = selector.

Attribute name	Type	Default value	Note
range	array/object array	[]	The range is valid when the mode is either selector or multiSelector.
range-key	string	-	When the range is an Object Array, the key value within the Object is specified as the selector display content via the range-key.
value	number	0	This represents the selection of the nth (starting from 0) element in the range.
bindchange	eventhandle	-	The change event is triggered when the value alters, with event.detail = {value}.

Multi-column selector: mode = multiSelector

Attribute name	Type	Default value	Note
range	array/object array	[]	The range is valid when the mode is either selector or multiSelector.
range-key	string	-	When the range is an Object Array, the key value within the Object is specified as the selector display content via the range-key.
value	array	[]	This represents the selection of the nth (starting from 0) element in the range.
bindchange	eventhandle	-	The change event is triggered when the value alters, with event.detail = {value}.
bindcolumnchange	eventhandle	-	Triggered when the column changes.

Time selector: mode = time

Attribute name	Type	Default value	Note
value	string	-	This represents the selected time, formatted as "hh:mm".

©2013-2022 Tencent Cloud. All rights reserved.

start	string	-	This signifies the commencement of the valid time range, formatted as a string "hh:mm".
end	string	-	This signifies the termination of the valid time range, formatted as a string "hh:mm".
bindchange	eventhandle	-	The change event is triggered when the value alters, with event.detail = {value}.

Date selector: mode = date

Attribute name	Type	Valid values and their descriptions	Default value	Note
value	string	-	The present day	This represents the selected date, formatted as "YYYY-MM-DD".
start	string	-	-	This denotes the commencement of the valid date range, formatted as a string "YYYY-MM-DD".
end	string	-	-	This signifies the termination of the valid date range, formatted as a string "YYYY-MM-DD".
fields	string	Year: The selector granularity is set to annual. Month: The selector granularity is set to monthly. Day: The selector granularity is set to daily.	day	Valid values are 'year', 'month', 'day', indicating the granularity of the selector.
bindchange	eventhandle	-	-	The change event is triggered when the value alters, with event.detail = {value}.

Province, city, and district selector: mode = region

Attribute name	Type	Default value	Note

©2013-2022 Tencent Cloud. All rights reserved.

value	Array	[]	This represents the selected province, city, and district, with the default being the first value in each column.
custom-item	string	-	A custom item can be added to the top of each column.
bindchange	evenhandle	-	When the 'value' changes, it triggers the 'change' event, with event.detail = {value: value, code: code, postcode: postcode}. Here, the 'code' field is the statistical division code, and 'postcode' is the postal code.
bindcancel	evenhandle	-	Triggered when a selection is canceled.
disabled	boolean	false	Whether to disable

radio

Feature Description: Single selection item.

Parameter and Description

Attribute name	Type	Default value	Required	Note
value	string	-	No	Radio identifier. When this radio is selected, the 'change' event of the radio-group will carry the value of the radio.
checked	boolean	false	No	Whether it is currently selected.
disabled	boolean	false	No	Whether to disable
color	string	#09BB07	No	The color of the radio, consistent with the CSS color.

radio-group

Feature Description: Single-item selector, internally composed of multiple radio elements.

Parameter and Description

Attribute name	Type	Default value	Required	Note
bindchange	eventhandle	-	No	When the selected item in the radio-group changes, it triggers the 'change' event, with detail = {value: [array of the selected radio's value]}.

©2013-2022 Tencent Cloud. All rights reserved.

slider

Feature Description: Slide selector.

Parameter and Description

Attribute name	Type	Default value	Required	Note
min	number	0	No	Minimum value
max	number	100	No	Maximum value
step	number	1	No	Step size, whose value must be greater than 0 and divisible by (max - min).
disabled	boolean	false	No	Whether to disable
value	number	0	No	Current value
color	color	#e9e9e9	No	Color of the background bar (use the backgroundColor)
selected-color	color	#1aad19	No	Selected color (use the activeColor)
activeColor	color	#1aad19	No	Selected color
backgroundColor	color	#e9e9e9	No	Color of the background bar
show-value	boolean	false	No	Whether to display the current value
bindchange	eventhandle	-	No	Event triggered upon completion of a single drag, event.detail = {value}
block-size	number	28	No	Slider size, ranging from 12 to 28
block-color	color	#ffffff	No	Slider color

switch

Feature Description: Switch selector.

Parameter and Description

Attribute name	Type	Default	Required	Note

©2013-2022 Tencent Cloud. All rights reserved.

		value		
checked	boolean	false	No	Selection status
disabled	boolean	false	No	Whether to disable
type	string	switch	No	Style, whose valid values include switch and checkbox.
color	string	#04BE02	No	Color of the switch, the same as the CSS color
bindchange	eventhandle	-	No	Triggering the change event when a click results in a modification of the checked status, event.detail={ value}

textarea

Feature Description: Multi-line input box. This component is a native component, so mind the relevant restrictions when using it.

Parameter and Description:

Attribute name	Type	Valid Values and Descriptions	Default value	Required	Note
value	string	-	-	No	The content of the input box.
placeholder	string	-	-	No	Placeholder when the input field is empty.
placeholder-style	string	-	-	No	Specify the style of the placeholder, currently only supporting color, font-size, and font-weight.
placeholder-class	string	-	textarea-placeholder	No	Specify the style class of the placeholder.
disabled	boolean	-	false	No	Whether to disable
maxlength	number	-	140	No	Maximum input length, unrestricted when set to -1.

©2013-2022 Tencent Cloud. All rights reserved.

auto-focus	boolean	-	false	No	Automatic focus, invoking the keyboard.
focus	boolean	-	false	No	Get focus.
auto-height	boolean	-	false	No	Whether to automatically increase in height, when auto-height is set, style.height will not take effect.
cursor	number	-	-1	No	Specify the cursor position when focus is applied.
selection-start	number	-	-1	No	Cursor starting position, effective when automatically focused, which must be used in conjunction with `selection-end` .
selection-end	number	-	-1	No	Cursor ending position, effective when automatically focused, which must be used in conjunction with `selection-start` .
confirm-type	string	Send: The button in the bottom right corner is labeled "Send". Search: The button in the bottom right corner is labeled "Search".	return	No	Set the text of the button in the bottom right corner of the keyboard.

©2013-2022 Tencent Cloud. All rights reserved.

		Next: The button in the bottom right corner is labeled "Next". Go: The button in the bottom right corner is labeled "Go". Done: The button in the bottom right corner is labeled "Done". Return: The button in the bottom right corner is labeled "Return".			
confirm-hold	boolean	-	false	No	Should the keyboard remain open when the button in the bottom right corner is clicked?

©2013-2022 Tencent Cloud. All rights reserved.

API

API Overview

Last updated：2024-07-12 19:07:20

1. Fundamentals

1.1. System Capabilities

Name	Feature Description	Earliest Version
canIUse	Determine whether the Mini Program's API, callbacks, parameters, components, etc., are usable in the current version.	1.5.0
env	Environment Variables	1.5.1
base64ToArrayBuffer	Convert Base64 string into an ArrayBuffer object.	1.5.1
arrayBufferToBase64	Convert ArrayBuffer object into a Base64 string.	1.5.1

1.2 System

Name	Feature Description	Earliest Version
getSystemInfo	Asynchronously call and get system information.	1.5.0
getSystemInfoSync	Synchronously call and get system information.	1.5.1
getSystemInfoAsync	Asynchronously get system information.	1.5.1
getWindowInfo	Get window information.	1.5.1
getSystemSetting	Get device settings.	1.5.0
openSystemBluetoothSetting	Redirect to the system's Bluetooth settings page.	1.5.1
openAppAuthorizeSetting	Redirect to the system's host client authorization management page.	1.5.1
getRendererUserAgent	Get the UserAgent of the Webview mini program.	1.5.1
getDeviceInfo	Get basic device information.	1.5.1

©2013-2022 Tencent Cloud. All rights reserved.

getAppBaseInfo	Get basic information of the host client.	1.5.1
getAppAuthorizeSetting	Get the host client's authorization settings.	1.5.1

1.3 Update

Name	Feature Description	Earliest Version
getUpdateManager	Get the globally unique version update manager for managing mini program updates.	1.5.1
UpdateManager.applyUpdate	Enforce a mini program to restart and use the new version.	1.5.0
UpdateManager.onCheckForUpdate	Monitor events of requesting an update check result from the WeChat backend.	1.5.0
UpdateManager.onUpdateFailed	Monitor mini program update failure events.	1.5.0
UpdateManager.onUpdateReady	Monitor mini program version update events.	1.5.0

1.4 Mini Program

Name	Feature Description	Earliest Version
getLaunchOptionsSync	Get the parameters at the initiation of the mini program.	1.5.0
getEnterOptionsSync	Get the parameters at the initiation of this mini program session.	1.5.1
onError	Monitor mini program error events.	1.5.0
offError	Remove the listener function for mini program error events.	1.5.0
onThemeChange	Monitor system theme change events.	1.5.0
offThemeChange	Remove the listener function for system theme change events.	1.5.0
onPageNotFound	Monitor events "the page to be opened in the mini program does not exist".	1.5.1
offPageNotFound	Remove the listener function for events when the page to be opened in the mini program does not exist.	1.5.1

©2013-2022 Tencent Cloud. All rights reserved.

onAppShow	Monitor events of mini program switch to frontend.	1.5.1
offAppShow	Remove the listener function for events of mini program switch to frontend.	1.5.1
onAppHide	Monitor events of mini program switch to backend.	1.5.1
offAppHide	Remove the listener function for events of mini program switch to frontend.	1.5.1
onUnhandledRejection	Monitor unprocessed Promise rejection events.	1.5.1
onUnhandledRejection	Remove the listener function for unprocessed Promise rejection events.	1.5.1

1.5 Debugging

Name	Feature Description	Earliest Version
setEnableDebug	Set whether to activate the debugging switch.	1.5.0
getLogManager	Get the log manager object.	1.5.1
console.debug	Print debug logs into the debugging panel.	1.5.0
console.error	Print error logs into the debugging panel.	1.5.0
console.group	Create a new group in the debugging panel.	1.5.0
console.groupEnd	Terminate the group created by console.group.	1.5.0
console.info	Print info logs into the debugging panel.	1.5.0
console.log	Print log entries into the debugging panel.	1.5.0
console.warn	Print warn logs into the debugging panel.	1.5.0
LogManager.debug	Write debug logs.	1.5.0
LogManager.info	Write info logs.	1.5.0
LogManager.log	Write log entries.	1.5.0
LogManager.warn	Write warn logs.	1.5.0
RealtimeLogManager.addFilterMsg	Set the page where the real-time log page parameter is located	1.5.16

©2013-2022 Tencent Cloud. All rights reserved.

RealtimeLogManager.error	Write error log	1.5.16
RealtimeLogManager.getCurrentState	Able to get the remaining space in the current cache.	1.5.16
RealtimeLogManager.in	Use the page where you set the real-time log page parameter	1.5.16
RealtimeLogManager.info	Write info logs	1.5.16
RealtimeLogManager.setFilterMsg	Setting up filter keywords	1.5.16
RealtimeLogManager.warn	Write warn logs	1.5.16
RealtimeTagLogManageraddFilterMsg	Add keywords for filtering	1.5.16
RealtimeTagLogManager.error	Write error logs.	1.5.16
RealtimeTagLogManager.info	Write info logs.	1.5.16
RealtimeTagLogManager.setFilterMsg	Set keywords for filtering	1.5.16
RealtimeTagLogManager.warn	Write warn logs.	1.5.16

2. Routing

Name	Feature Description	Earliest Version
switchTab	Redirect to the tabBar page and close all other non-tabBar pages.	1.5.0
reLaunch	Close all pages and navigate to a specific page within the application.	1.5.0
redirectTo	Close the current page and redirect to a specific page within the application.	1.5.0
navigateTo	Retain the current page and redirect to a specific page within the application.	1.5.1
navigateBack	Close the current page and return to the previous page or multiple levels up.	1.5.0
EventChannel	Inter-page event communication channel	1.5.1

©2013-2022 Tencent Cloud. All rights reserved.

3. Redirection

Name	Feature Description	Earliest Version
exitMiniProgram	Exit the current mini program.	1.5.0
navigateToMiniProgram	Open another mini program.	1.5.0
navigateBackMiniProgram	Return to the previous mini program. This feature can only be successfully called when the current mini program has been launched by another mini program.	1.5.1

4. Forwarding

Name	Feature Description	Earliest Version
hideShareMenu	Hide the forwarding button on the current page.	1.5.0
showShareMenu	Display the forwarding button on the current page.	1.5.0
updateShareMenu	Update forwarding attributes.	1.5.0
showShareImageMenu	Open the share image pop-up window	1.5.35
onCopyUrl	Listen to the event triggered when the user clicks the "Copy Link" button in the menu in the upper right corner	1.5.35
offCopyUrl	Remove all listening functions for events triggered when the user clicks the "Copy Link" button in the menu in the upper right corner	1.5.35

©2013-2022 Tencent Cloud. All rights reserved.

5. Interface

5.1 Interaction

Name	Feature Description	Earliest Version
hideLoading	Hide the loading notification box.	1.5.0
hideToast	Hide the message notification box.	1.5.0
showActionSheet	Display the operation menu.	1.5.0
showLoading	Display the loading notification box.	1.5.0
showModal	Display the modal dialog box.	1.5.0
showToast	Display the message notification box.	1.5.0
enableAlertBeforeUnload	Initiate the inquiry dialog box for returning to the mini program page.	1.5.1
disableAlertBeforeUnload	Terminate the inquiry dialog box for returning to the mini program page.	1.5.1

5.2 Navigation Bar

Name	Feature Description	Earliest Version
setNavigationBarTitle	Dynamically set the title of the current page.	1.5.0
setNavigationBarColor	Set the color scheme for the page navigation bar.	1.5.0
hideHomeButton	Hide the button leading back to the homepage.	1.5.0
showNavigationBarLoading	Display the navigation bar loading animation on the current page.	1.5.1
hideNavigationBarLoading	Hide the navigation bar loading animation on the current page.	1.5.1

5.3 Background

Name	Feature Description	Earliest Version

©2013-2022 Tencent Cloud. All rights reserved.

setBackgroundColor	Dynamically set the background color of the window.	1.5.0
setBackgroundTextStyle	Dynamically set the style of the dropdown background text and loading image.	1.5.0

5.4、TabBar

Name	Feature Description	Earliest Version
showTabBar	Display the tabBar.	1.5.0
hideTabBar	Hide the tabBar.	1.5.0
setTabBarStyle	Dynamically set the overall style of the tabBar.	1.5.0
setTabBarItem	Dynamically set the content of a specific tabBar item. As of version 2.7.0, images can be temporary files and network files.	1.5.0
showTabBarRedDot	Display the red dot in the top right corner of a specific tabBar item.	1.5.0
hideTabBarRedDot	Hide the red dot in the top right corner of a specific tabBar item.	1.5.0
setTabBarBadge	Append text to the top right corner of a specific tabBar item.	1.5.0
removeTabBarBadge	Remove the text in the top right corner of a specific tabBar item.	1.5.0

5.5 Implement pull-to-refresh functionality

Name	Feature Description	Earliest Version
startPullDownRefresh	Initiate pull-to-refresh action.	1.5.0
stopPullDownRefresh	Stop the pull-to-refresh action on the current page.	1.5.0

5.6 Scrolling

Name	Feature Description	Earliest Version

©2013-2022 Tencent Cloud. All rights reserved.

pageScrollTo	Scroll the page to the desired location, with two methods of positioning offered: selector and scroll distance.	1.5.0
ScrollViewContext.scrollIntoView	Scroll to the specified location.	1.5.0
ScrollViewContext.scrollTo	Scroll to the specified location.	1.5.0

5.7 Animation

Name	Feature Description	Earliest Version
createAnimation	Create an animation instance, termed as "animation".	1.5.0
Animation.backgroundColor	Set the background color.	1.5.0
Animation.bottom	Set the "bottom" value.	1.5.0
Animation.export	Export the animation queue.	1.5.0
Animation.height	Set the height.	1.5.0
Animation.left	Set the "left" value.	1.5.0
Animation.matrix	In accordance with transform-function matrix.	1.5.0
Animation.matrix3d	In accordance with transform-function matrix3d.	1.5.0
Animation.opacity	Set the transparency.	1.5.0
Animation.right	Set the "right" value.	1.5.0
Animation.rotate	Rotate an angle clockwise from the origin.	1.5.0
Animation.rotate3d	Rotate an angle clockwise from a fixed axis.	1.5.0
Animation.rotateX	Rotate an angle clockwise from the X-axis.	1.5.0
Animation.rotateY	Rotate an angle clockwise from the Y-axis.	1.5.0
Animation.rotateZ	Rotate an angle clockwise from the Z-axis.	1.5.0
Animation.scale	Scaling	1.5.0
Animation.scale3d	Scaling	1.5.0
Animation.scaleX	Scale the X-axis.	1.5.0

©2013-2022 Tencent Cloud. All rights reserved.

Animation.scaleY	Scale the Y-axis.	1.5.0
Animation.scaleZ	Scale the Z-axis.	1.5.0
Animation.skew	Skew the coordinates of the X and Y axes.	1.5.0
Animation.skewX	Skew the coordinates of the X-axis.	1.5.0
Animation.skewY	Skew the coordinates of the Y-axis.	1.5.0
Animation.step	Indicate the completion of a set of animations.	1.5.0
Animation.top	Set the top value.	1.5.0
Animation.translate	Translation transformation	1.5.0
Animation.translate3d	Perform a translation transformation on the xyz coordinates.	1.5.0
Animation.translateX	Translate along the X-axis.	1.5.0
Animation.translateY	Translate along the Y-axis.	1.5.0
Animation.translateZ	Translate along the Z-axis.	1.5.0
Animation.width	Set the width.	1.5.0

5.8 Custom Component

Name	Feature Description	Earliest Version
nextTick	Defer part of operations to be executed in the subsequent time slice.	1.5.0

5.9 Menu

Name	Feature Description	Earliest Version
getMenuButtonBoundingClientRect	Get the layout position information of the menu button (the capsule button in the top right corner).	1.5.0

5.10 Window

Name	Feature Description	Earliest

©2013-2022 Tencent Cloud. All rights reserved.

		Version
onWindowResize	Monitor window size change events.	1.5.0
offWindowResize	Remove the listener function for window size change events.	1.5.1

5.11 Font.

Name	Feature Description	Earliest Version
loadFontFace	Dynamically load web fonts.	1.5.0

6. Network

6.1 Initiate Request

Name	Feature Description	Earliest Version
request	Initiate an HTTPS network request.	1.5.0
RequestTask.abort	Interrupt the request task.	1.5.0
RequestTask.onChunkReceived	Monitor Transfer-Encoding Chunk Received events.	1.5.1
RequestTask.offChunkReceived	Remove Transfer-Encoding Chunk Received events.	1.5.1
RequestTask.onHeadersReceived	Monitor HTTP Response Header events.	1.5.0
RequestTask.offHeadersReceived	Remove the listener function for HTTP Response Header events.	1.5.0

6.2 Upload

Name	Feature Description	Earliest Version
uploadFile	Upload local resources to the server.	1.5.0
UploadTask.abort	Interrupt the upload task.	1.5.0

©2013-2022 Tencent Cloud. All rights reserved.

UploadTask.onProgressUpdate	Monitor upload progress change events.	1.5.0
UploadTask.offProgressUpdate	Stop monitoring upload progress change events.	1.5.0
UploadTask.onHeadersReceived	Monitor HTTP Response Header events.	1.5.0
UploadTask.offHeadersReceived	Remove the listener function for HTTP Response Header events.	1.5.0

6.3 Download

Name	Feature Description	Earliest Version
downloadFile	Download file resources to local storage. The client initiates a direct HTTPS GET request, returning the local temporary path of the file. The maximum file size allowed for a single download is 200 MB.	1.5.1
DownloadTask.abort	Interrupt the download task.	1.5.1
DownloadTask.onProgressUpdate	Monitor download progress change events.	1.5.1
DownloadTask.offProgressUpdate	Remove the listener function for download progress change events.	1.5.1
DownloadTask.onHeadersReceived	Monitor HTTP Response Header events.	1.5.1
DownloadTask.offHeadersReceived	Remove the listener function for HTTP Response Header events.	1.5.1

6.4、WebSocket

Name	Feature Description	Earliest Version
sendSocketMessage	Send data through a WebSocket connection. It requires an initial wx.connectSocket, and data can only be sent after the wx.onSocketOpen callback.	1.5.1
onSocketOpen	Monitor WebSocket connection opening events.	1.5.1
onSocketMessage	Monitor events WebSocket receiving a message from the server.	1.5.1
onSocketError	Monitor WebSocket error events.	1.5.1

©2013-2022 Tencent Cloud. All rights reserved.

onSocketClose	Monitor WebSocket connection closing events.	1.5.1
connectSocket	Create a WebSocket connection.	1.5.1
closeSocket	Disable WebSocket connection.	1.5.1
SocketTask.close	Disable WebSocket connection.	1.5.0
SocketTask.onClose	Monitor WebSocket connection closing events.	1.5.0
SocketTask.onError	Monitor WebSocket error events.	1.5.0
SocketTask.onMessage	Monitor events WebSocket receiving a message from the server.	1.5.0
SocketTask.onOpen	Monitor WebSocket connection opening events.	1.5.0
SocketTask.send	Transmit data via the WebSocket connection.	1.5.0

6.5 UDP Communication

Name	Feature Description	Earliest Version
createUDPSocket	Creates a UDP Socket instance.	1.5.1
UDPSocket.bind	Bind to a system-assigned random available port, or bind to a specified port number.	1.5.1
UDPSocket.close	Terminates a UDP Socket instance.	1.5.1
UDPSocket.connect	Connect to the specified IP and port in advance, which shall be used in conjunction with the write method.	1.5.1
UDPSocket.onClose	Monitor closing events.	1.5.1
UDPSocket.offClose	Remove the listener function for closing events.	1.5.1
UDPSocket.onError	Monitor error events.	1.5.1
UDPSocket.offError	Remove the listener function for error events.	1.5.1
UDPSocket.onListening	Monitor events where the monitoring of packet messages is initiated.	1.5.1
UDPSocket.offListening	Remove the listener function for events where the monitoring of packet messages is initiated.	1.5.1

©2013-2022 Tencent Cloud. All rights reserved.

UDPSocket.onMessage	Monitor message receipt events.	1.5.1
UDPSocket.offMessage	Remove the listener function for message receipt events.	1.5.1
UDPSocket.send	Send messages to the designated IP and port.	1.5.1
UDPSocket.setTTL	Configure the IP_TTL socket option, which sets the maximum number of hops permitted for an IP packet during transmission.	1.5.16
UDPSocket.write	The usage is the same as the send method.	1.5.16

6.6 TCP Communication

Name	Feature Description	Earliest Version
createTCPSocket	Creates a TCP Socket instance.	1.5.1
TCPSocket.bindWifi	Bind the TCP Socket to the current Wi-Fi network, which will, upon successful execution, trigger the onBindWifi event (supported only on Android).	1.5.1
TCPSocket.close	Disable Connection	1.5.1
TCPSocket.connect	Initiate a connection on the specified socket.	1.5.1
TCPSocket.onClose	Monitor for events that are emitted once the socket is completely closed.	1.5.1
TCPSocket.offClose	Remove the listener function for events that are emitted once the socket is completely closed.	1.5.1
TCPSocket.onConnect	Monitor events that are triggered when a socket connection is successfully established.	1.5.1
TCPSocket.offConnect	Remove the listener function for events that are triggered once the socket connection is successfully established.	1.5.1
TCPSocket.onError	Monitor is triggered when an error occurs.	1.5.1
TCPSocket.offError	Remove the listener function that is triggered when an error occurs.	1.5.1
TCPSocket.onMessage	Monitor events that are triggered when data is received.	1.5.1
TCPSocket.offMessage	Remove the listener function for events that are triggered when data is received.	1.5.1

©2013-2022 Tencent Cloud. All rights reserved.

TCPSocket.onBindWifi	Monitor events that are triggered when a socket successfully binds to the current Wi-Fi network.	1.5.1
TCPSocket.offBindWifi	Remove the listener function for events that are triggered when a socket successfully binds to the current Wi-Fi network.	1.5.1
TCPSocket.write	Transmit data over the socket.	1.5.1

6.7、mDNS

Name	Feature Description	Earliest Version
stopLocalServiceDiscovery	Cease the mDNS searching services.	1.5.1
startLocalServiceDiscovery	Initiate the search for mDNS services within the local area network.	1.5.1
offLocalServiceResolveFail	Remove the listener function for events of mDNS service resolution failure.	1.5.1
onLocalServiceResolveFail	Monitor events mDNS service resolution failure.	1.5.1
offLocalServiceLost	Remove the listener function for mDNS service departure events.	1.5.1
onLocalServiceLost	Monitor mDNS service departure events.	1.5.1
offLocalServiceFound	Remove the listener function for mDNS service discovery events.	1.5.1
onLocalServiceFound	Monitor mDNS service discovery events.	1.5.1
offLocalServiceDiscoveryStop	Remove the listener function for events of mDNS service searching cessation.	1.5.1
onLocalServiceDiscoveryStop	Monitor events of mDNS service searching cessation.	1.5.1

7. Data Caching

Name	Feature Description	Earliest

©2013-2022 Tencent Cloud. All rights reserved.

		Version
setStorage	Store the data in the specified key within the local cache.	1.5.0
setStorageSync	Store the data in the specified key within the local cache.	1.5.0
revokeBufferURL	Destroy data stored in memory based on URL	1.5.35
removeStorage	Remove the specified key from the local cache.	1.5.0
removeStorageSync	Synced version of removeStorage.	1.5.0
getStorage	Asynchronously get the content of the specified key from the local cache.	1.5.0
createBufferURL	Create a unique URL based on the incoming buffer and store it in memory	1.5.35
getStorageSync	Synchronously get the content of the specified key from the local cache.	1.5.0
getStorageInfo	Asynchronously get relevant information about the current storage.	1.5.0
getStorageInfoSync	Synced version of getStorageInfo.	1.5.0
clearStorage	Clear local data cache.	1.5.0
clearStorageSync	Synced version of clearStorage.	1.5.0
batchSetStorage	Batch store the data in the specified key within the local cache.	1.5.16
batchSetStorageSync	Synced version of batchSetStorage.	1.5.16
batchGetStorage	Asynchronously batch get the content of the specified key from the local cache.	1.5.16
batchGetStorageSync	Synced version of batchGetStorage.	1.5.16

8. Data Analysis

©2013-2022 Tencent Cloud. All rights reserved.

Name	Feature Description	Earliest Version
reportEvent	Report Event	1.5.16

9. Canvas

9.1 Canvas Overview

Name	Feature Description	Earliest Version
createCanvasContext	Create a CanvasContext object.	1.5.0
canvasToTempFilePath	Export the content of the specified area of the current canvas to generate an image of the designated size.	1.5.0
canvasPutImageData	Draw pixel data onto the canvas.	1.5.0
canvasGetImageData	Get the implicit pixel data from the canvas area.	1.5.0
Image	Image Object	1.5.16
ImageData	ImageData Object	1.5.16
RenderingContext	Canvas Drawing Context	1.5.16

9.2、canvasContext

Name	Feature Description	Earliest Version
CanvasContext.arc	Construct an arc line.	1.5.0
CanvasContext.arcTo	Sketch an arc path based on control points and radius.	1.5.0
CanvasContext.beginPath	Start creating a path.	1.5.0
CanvasContext.bezierCurveTo	Formulate a cubic Bezier curve path.	1.5.0
CanvasContext.clearRect	Erase the content within this rectangular area on the canvas.	1.5.0

©2013-2022 Tencent Cloud. All rights reserved.

CanvasContext.clip	Excise any shape and size from the original canvas.	1.5.0
CanvasContext.closePath	Close a path.	1.5.0
CanvasContext.createCircularGradient	Craft a circular gradient hue.	1.5.0
CanvasContext.createLinearGradient	Craft a linear gradient hue.	1.5.0
CanvasContext.createPattern	The method of fabricating a pattern for a specified image can replicate the original image in the designated direction.	1.5.0
CanvasContext.draw	Draw the prior descriptions (path, transformation, style) from the drawing context onto the canvas.	1.5.0
CanvasContext.drawImage	Draw images onto the canvas.	1.5.0
CanvasContext.fill	Implement filling within the current path's content.	1.5.0
CanvasContext.fillRect	Fill a rectangle.	1.5.0
CanvasContext.fillText	Draw filled text upon the canvas.	1.5.0
CanvasContext.lineTo	Add a new point and then generate a line from the previously specified point to the target point.	1.5.0
CanvasContext.measureText	Gauge the dimensions of the text.	1.5.0
CanvasContext.moveTo	Move the path to a specified point within the canvas, without generating any lines.	1.5.0
CanvasContext.quadraticCurveTo	Formulate a quadratic Bezier curve path.	1.5.0
CanvasContext.rect	Create a rectangular path.	1.5.0
CanvasContext.restore	Restore the previously stored drawing context.	1.5.0
CanvasContext.rotate	Rotate the current coordinate axis clockwise, using the origin as the pivot.	1.5.0
CanvasContext.save	Save the drawing context.	1.5.0
CanvasContext.scale	Upon calling, the horizontal and vertical coordinates of the subsequently created paths will be scaled.	1.5.0
CanvasContext.setFillStyle	Set the filling color.	1.5.0

©2013-2022 Tencent Cloud. All rights reserved.

CanvasContext.setFontSize	Set the font size.	1.5.0
CanvasContext.setGlobalAlpha	Set the global brush transparency.	1.5.0
CanvasContext.setLineCap	Set the endpoint style of the line.	1.5.0
CanvasContext.setLineDash	Set the style of the dashed line.	1.5.0
CanvasContext.setLineJoin	Set the intersection style of the line.	1.5.0
CanvasContext.setLineWidth	Set the line width.	1.5.0
CanvasContext.setMiterLimit	Set the maximum miter length.	1.5.0
CanvasContext.setShadow	Set the shade style.	1.5.0
CanvasContext.setStrokeStyle	Set the color of the outline.	1.5.0
CanvasContext.setTextAlign	Set the alignment of the text.	1.5.0
CanvasContext.setTextBaseline	Set the vertical alignment of the text.	1.5.0
CanvasContext.setTransform	Use the matrix to reset (overwrite) the current transformation method.	1.5.0
CanvasContext.stroke	Draw the border of the current path.	1.5.0
CanvasContext.strokeRect	Draw a rectangle (not a filled one).	1.5.0
CanvasContext.strokeText	The method of sketching text outlines at the given (x, y) position.	1.5.0
CanvasContext.transform	Use the matrix to repeatedly overlay the current transformation.	1.5.0
CanvasContext.translate	Transform the origin (0, 0) of the current coordinate system.	1.5.0

9.3、CanvasGradient

Name	Feature Description	Earliest Version
CanvasGradient.addColorStop	Add gradient points of color.	1.5.0

9.4、Canvas

©2013-2022 Tencent Cloud. All rights reserved.

Name	Feature Description	Earliest Version
Canvas.cancelAnimationFrame	Cancel the animation request scheduled by requestAnimationFrame.	1.5.16
Canvas.createImageData	Create an ImageData object.	1.5.16
Canvas.createImage	Create an image object.	1.5.16
Canvas.createPath2D	Create a Path2D object.	1.5.16
Canvas.getContext	This method returns the canvas drawing context.	1.5.16
Canvas.requestAnimationFrame	To be executed upon the next redraw.	1.5.16
Canvas.toDataURL	Return a data URL that contains the image display.	1.5.16

9.5、OffscreenCanvas

Name	Feature Description	Earliest Version
OffscreenCanvas.createImage	This method returns the OffscreenCanvas drawing context.	1.5.16
OffscreenCanvas.getContext	Create an image object.	1.5.16

9.6、Path2D

Name	Feature Description	Earliest Version
Path2D.addPath	Add a path into the existing one.	1.5.16
Path2D.arc	Add a segment of an arc path.	1.5.16
Path2D.arcTo	Add a segment of an arc path by providing control points.	1.5.16
Path2D.bezierCurveTo	Add a cubic Bezier curve path.	1.5.16
Path2D.closePath	Seal the path back to its origin.	1.5.16
Path2D.ellipse	Add an elliptical arc path.	1.5.16
Path2D.lineTo	Add a linear path.	1.5.16

©2013-2022 Tencent Cloud. All rights reserved.

Path2D.moveTo	Move the starting point of the path.	1.5.16
Path2D.quadraticCurveTo	Add a quadratic Bezier curve path.	1.5.16
Path2D.rect	Add a square path.	1.5.16

10. Media

10.1. Map

Name	Feature Description	Earliest Version
createMapContext	Create a MapContext object.	1.5.0
MapContext.addArc	Add an arc line.	1.5.16
MapContext.addCustomLayer	Add a personalized layer.	1.5.16
MapContext.addGroundOverlay	Create a custom image layer.	1.5.16
MapContext.addMarkers	Add a marker.	1.5.0
MapContext.eraseLines	Erase or gray out the line segments that have been integrated into the map.	1.5.16
MapContext.fromScreenLocation	Get the corresponding longitude and latitude of the point on the screen, with the coordinate origin situated at the top left corner of the map.	1.5.0
MapContext.getCenterLocation	Get the longitude and latitude of the current map center.	1.5.0
MapContext.getRegion	Get the view field range of the current map.	1.5.0
MapContext.getRotate	Get the rotation angle of the current map.	1.5.0
MapContext.getScale	Get the scaling level of the current map.	1.5.0
MapContext.getSkew	Get the tilt angle of the current map.	1.5.0
MapContext.includePoints	Scale the field of view to display all longitudes and latitudes.	1.5.0

©2013-2022 Tencent Cloud. All rights reserved.

MapContext.initMarkerCluster	Initialize the configuration of point aggregation.	1.5.16
MapContext.moveAlong	Move the marker along the specified path, suitable for scenarios such as trajectory playback.	1.5.0
MapContext.moveToLocation	Shift the map center to the current location point, when it is necessary to set the map component show-location to true.	1.5.0
MapContext.on	Monitor map events.	1.5.16
MapContext.openMapApp	Launch the map APP and select navigation.	1.5.0
MapContext.removeArc	Erase the arc line.	1.5.16
MapContext.removeCustomLayer	Remove the personalized layer.	1.5.16
MapContext.removeGroundOverlay	Remove the custom image layer.	1.5.16
MapContext.removeMarkers	Remove the marker.	1.5.0
MapContext.setBoundary	Constrain the display scope of the map.	1.5.16
MapContext.setCenterOffset	Set the offset of the map's central point.	1.5.16
MapContext.setLocMarkerIcon	Set the icon for the location point.	1.5.16
MapContext.toScreenLocation	Get the screen coordinates corresponding to the latitude and longitude.	1.5.16
MapContext.translateMarker	Translate the marker, along with animation.	1.5.0
MapContext.updateGroundOverlay	Update the custom image layer.	1.5.16

10.2 Image.

Name	Feature Description	Earliest Version
chooseImage	Select an image from the local album or use the camera to capture one.	1.5.0
compressImage	Image compression interface, with optional compression quality.	1.5.1
getImageInfo	Get image information	1.5.0
previewImage	Preview the image in full screen on a new page.	1.5.0

©2013-2022 Tencent Cloud. All rights reserved.

previewMedia	Preview of images and videos.	1.5.1
saveImageToPhotosAlbum	Save the image to the system's album.	1.5.1
chooseMessageFile	Selecting files from the client session.	1.5.1

10.3 Real-time Audio and Video.

Name	Features	Earliest Version
createLivePusherContext	Create a LivePusherContext object.	1.5.0
createLivePlayerContext	Create a LivePlayerContext object.	1.5.0
LivePlayerContext.exitfullscreen	Exit Full Screen	1.5.0
LivePlayerContext.mute	Mute	1.5.0
LivePlayerContext.pause	Pause	1.5.0
LivePlayerContext.play	Play	1.5.0
LivePlayerContext.requestFullScreen	Full Screen	1.5.0
LivePlayerContext.resume	Restore	1.5.0
LivePlayerContext.snapshot	Screenshot	1.5.0
LivePlayerContext.stop	Stop	1.5.0
LivePusherContext.start	Initiate streaming while simultaneously enabling camera preview.	1.5.0
LivePusherContext.stop	Cease streaming while concurrently disabling camera preview.	1.5.0
LivePusherContext.pause	Pause streaming	1.5.0
LivePusherContext.resume	Restore streaming	1.5.0
LivePusherContext.switchCamera	Switch between front and rear cameras.	1.5.0
LivePusherContext.snapshot	Snapshot	1.5.0
LivePusherContext.toggleTorch	Toggle flashlight.	1.5.0
LivePusherContext.playBGM	Play background music	1.5.0

©2013-2022 Tencent Cloud. All rights reserved.

LivePusherContext.stopBGM	Cease background music.	1.5.0
LivePusherContext.pauseBGM	Pause background music.	1.5.0
LivePusherContext.resumeBGM	Restore background music.	1.5.0
LivePusherContext.setBGMPosition	Set the background music progress.	1.5.0
LivePusherContext.setMICVolume	Set the microphone volume.	1.5.0
LivePusherContext.setBGMVolume	Set the background music volume.	1.5.0
LivePusherContext.setAudioReverbType	Set audio mixing type.	1.5.0
LivePusherContext.startPreview	Enable camera preview.	1.5.0
LivePusherContext.stopPreview	Disable camera preview.	1.5.0
LivePusherContext.startAudioRecord	Start audio recording.	1.5.0
LivePusherContext.stopAudioRecord	Stop audio recording.	1.5.0

10.4 Audio

Name	Feature Description	Earliest Version
stopVoice	Stop audio playback.	1.5.1
playVoice	Start audio playback.	1.5.1
pauseVoice	Pause the ongoing audio playback.	1.5.1
setInnerAudioOption	Set the playback options for InnerAudioContext.	1.5.0
getAvailableAudioSources	Get the currently supported audio input sources.	1.5.1
createInnerAudioContext	Create an InnerAudioContext object.	1.5.0
AudioContext.pause	Pause audio	1.5.1
AudioContext.play	Start audio playback.	1.5.1
AudioContext.seek	Redirect to the specified location.	1.5.1
AudioContext.setSrc	Set the audio address.	1.5.1
InnerAudioContext.destroy	Terminate current instance.	1.5.0

©2013-2022 Tencent Cloud. All rights reserved.

InnerAudioContext.offCanplay	Remove the listener function for events of audio entering the playable status.	1.5.0
InnerAudioContext.offEnded	Remove the listener function for events of audio naturally progressing to its conclusion.	1.5.0
InnerAudioContext.offError	Remove the listener function for audio playback error events.	1.5.0
InnerAudioContext.offPause	Remove the listener function for audio playback pause events.	1.5.0
InnerAudioContext.offPlay	Remove the listener function for audio playback events.	1.5.0
InnerAudioContext.offSeeked	Remove the listener function for events of audio completing a skip operation.	1.5.0
InnerAudioContext.offSeeking	Remove the listener function for events of audio executing a skip operation.	1.5.0
InnerAudioContext.offStop	Remove the listener function for audio playback stop events.	1.5.0
InnerAudioContext.offTimeUpdate	Remove the listener function for events of audio playback progress update.	1.5.0
InnerAudioContext.offWaiting	Remove the listener function for audio loading events.	1.5.0
InnerAudioContext.onCanplay	Monitor events of audio entering the playable status.	1.5.0
InnerAudioContext.onEnded	Monitor events of audio naturally playing to its conclusion.	1.5.0
InnerAudioContext.onError	Monitor audio playback error events.	1.5.0
InnerAudioContext.onPause	Monitor audio pausing events.	1.5.0
InnerAudioContext.onPlay	Monitor audio playback events.	1.5.0
InnerAudioContext.onSeeked	Monitor events of audio completing a skip operation.	1.5.0
InnerAudioContext.onSeeking	Monitor events of audio executing a skip operation.	1.5.0
InnerAudioContext.onStop	Monitor audio playback stop events.	1.5.0
InnerAudioContext.onTimeUpdate	Monitor events of audio playback progress update.	1.5.0
InnerAudioContext.onWaiting	Monitor audio loading events.	1.5.0

©2013-2022 Tencent Cloud. All rights reserved.

InnerAudioContext.pause	Pause	1.5.0
InnerAudioContext.play	Play	1.5.0
InnerAudioContext.seek	Redirect to the specified location.	1.5.0
InnerAudioContext.stop	Stop	1.5.0

10.5. Video

Name	Note	Earliest Version
saveVideoToPhotosAlbum	Save the video to the system's album.	1.5.0
createVideoContext	Create a VideoContext object.	1.5.0
chooseVideo	Capture a video or select one from the mobile phone's album.	1.5.0
compressVideo	Video compression interface	1.5.1
chooseMedia	Capture or select an image or video from the mobile phone's album.	1.5.1
VideoContext.exitFullScreen	Exit Full Screen	1.5.0
VideoContext.exitBackgroundPlayback	Exit the backend audio playback mode.	1.5.16
VideoContext.exitPictureInPicture	Exit the mini-window, a method that can be invoked on any page.	1.5.16
VideoContext.hideStatusBar	Hide the status bar, which is effective only in iOS full-screen mode.	1.5.0
VideoContext.pause	Pause video	1.5.0
VideoContext.play	Play video	1.5.0
VideoContext.requestFullScreen	Full Screen	1.5.0
VideoContext.requestBackgroundPlayback	Enter the backend audio playback mode.	1.5.16
VideoContext.seek	Redirect to the specified location.	1.5.0
VideoContext.sendDanmu	Dispatch the barrage.	1.5.0
VideoContext.showStatusBar	Display the status bar, which is effective only in	1.5.0

©2013-2022 Tencent Cloud. All rights reserved.

	iOS full-screen mode.	
VideoContext.stop	Stop video	1.5.0

10.6 Transparent Video

Name	Note	Earliest Version
createAnimationVideo	Create an AnimationVideoContext object.	1.5.0
AnimationVideoContext.play	Play video	1.5.0
AnimationVideoContext.pause	Pause video	1.5.0
AnimationVideoContext.seek	Redirect to the specified location.	1.5.0

10.7 Camera

Name	Feature Description	Earliest Version
createCameraContext	Create a CameraContext object.	1.5.0
CameraContext.onCameraFrame	Get real-time frame data from the Camera.	1.5.0
CameraContext.setZoom	Set the scaling level.	1.5.0
CameraContext.startRecord	Start video recording.	1.5.0
CameraContext.stopRecord	Stop video recording.	1.5.0
CameraContext.takePhoto	Capture a photo.	1.5.0
CameraFrameListener.start	Start monitoring frame data.	1.5.0
CameraFrameListener.stop	Stop monitoring frame data.	1.5.0

10.8 Audio Recording

Name	Features	Earliest Version
startRecord	Start audio recording.	1.5.1
stopRecord	Stop audio recording	1.5.1

©2013-2022 Tencent Cloud. All rights reserved.

getRecorderManager	Get the globally unique RecorderManager.	1.5.0
RecorderManager.onError	Monitor audio recording error events.	1.5.0
RecorderManager.onFrameRecorded	Monitor events where recording of files with the specified frame size has completed.	1.5.0
RecorderManager.onInterruptionBegin	Monitor events where the audio recording is interrupted due to system occupation.	1.5.0
RecorderManager.onInterruptionEnd	Monitor events where the audio recording interruption has ended.	1.5.0
RecorderManager.onPause	Monitor audio recording pause events.	1.5.0
RecorderManager.onResume	Monitor audio recording continuation events.	1.5.0
RecorderManager.onStart	Monitor audio recording start events.	1.5.0
RecorderManager.onStop	Monitor audio recording ending events.	1.5.0
RecorderManager.pause	Pause audio recording.	1.5.0
RecorderManager.resume	Continue audio recording.	1.5.0
RecorderManager.start	Start audio recording.	1.5.0
RecorderManager.stop	Pause audio recording.	1.5.0

10.9 Background Audio

Name	Note	Earliest Version
stopBackgroundAudio	Cease music playback.	1.5.1
seekBackgroundAudio	Regulate music playback progression.	1.5.1
playBackgroundAudio	Use the background player for music playback.	1.5.1
pauseBackgroundAudio	Pause monitoring of music playback stop events.	1.5.1
onBackgroundAudioStop	Monitor music playback stop events.	1.5.1
onBackgroundAudioPlay	Monitor music playback events.	1.5.1
onBackgroundAudioPause	Monitor music pause events.	1.5.1

©2013-2022 Tencent Cloud. All rights reserved.

getBackgroundAudioPlayerState	Get the status of background music playback.	1.5.1
getBackgroundAudioManager	Get the globally unique background audio manager.	1.5.1
BackgroundAudioManager.onCanplay	Monitor events of background audio entering the playable status.	1.5.16
BackgroundAudioManager.onEnded	Monitor events of background audio naturally playing to its conclusion.	1.5.16
BackgroundAudioManager.onError	Monitor background audio playback error events.	1.5.16
BackgroundAudioManager.onNext	Monitor events where the user clicks the "next track" button in the system music playback panel (iOS only).	1.5.16
BackgroundAudioManager.onPause	Monitor background audio pausing events.	1.5.16
BackgroundAudioManager.onPlay	Monitor background audio playback events.	1.5.16
BackgroundAudioManageronPrev	Monitor events where the user clicks the "previous track" button in the system music playback panel (iOS only).	1.5.16
BackgroundAudioManager.onSeeked	Monitor events of background audio completing a skip operation.	1.5.16
BackgroundAudioManager.orSeeking	Monitor events of background audio starting a skip operation.	1.5.16
BackgroundAudioManager.onStop	Monitor background audio playback stop events.	1.5.16
BackgroundAudioManager.onTimeUpdate	Monitor events of background audio playback progress update.	1.5.16
BackgroundAudioManager.onWaiting	Monitor audio loading events.	1.5.16
BackgroundAudioManager.pause	Pause music playback.	1.5.16
BackgroundAudioManager.play	Play the music.	1.5.16
BackgroundAudioManager.seek	Redirect to the specified location.	1.5.16
BackgroundAudioManager.stop	Stop music playback.	1.5.16

©2013-2022 Tencent Cloud. All rights reserved.

10.10 Rich Text

Name	Note	Earliest Version
EditorContext	EditorContext Instance	1.5.16
EditorContext.blur	The editor loses focus, concurrently retracting the keyboard.	1.5.16
EditorContext.clear	Empty the contents of the editor.	1.5.16
EditorContext.format	Alter the style.	1.5.16
EditorContext.getContents	Get the contents of the editor.	1.5.16
EditorContext.getSelectionText	Get the plain text content within the selected area of the editor.	1.5.16
EditorContext.insertDivider	Insert a dividing line.	1.5.16
EditorContext.insertImage	Insert image	1.5.16
EditorContext.insertText	Overwrite the current selected area, setting a segment of text.	1.5.16
EditorContext.redo	Restore	1.5.16
EditorContext.removeFormat	Clear the current selected area's style.	1.5.16
EditorContext.scrollIntoView	Ensure the editor's cursor scrolls into the window's visible area.	1.5.16
EditorContext.setContents	Initialize the editor's content.	1.5.16
EditorContext.undo	Cancel	1.5.1

10.11 Screen Recorder

Name	Note	Earliest Version
createMediaRecorder	Create a WebGL Screen Recorder.	1.5.16
MediaRecorder.destroy	Terminate the Recorder.	1.5.16
MediaRecorder.off	Cancel the monitoring of recording events.	1.5.16

©2013-2022 Tencent Cloud. All rights reserved.

MediaRecorder.on	Register a callback function to monitor recording events.	1.5.16
MediaRecorder.pause	Pause recording	1.5.16
MediaRecorder.requestFrame	Request the next frame for recording.	1.5.16
MediaRecorder.resume	Restore recording.	1.5.16
MediaRecorder.start	Start recording.	1.5.16
MediaRecorder.stop	Pause recording	1.5.16

11. File

11.1 File Overview

Name	Note	Earliest Version
saveFile	Save the file locally.	1.5.0
removeSavedFile	Delete local cache files.	1.5.0
openDocument	Open the document in a new page.	1.5.0
getSavedFileList	Get the list of locally cached files saved under this mini program.	1.5.0
getSavedFileInfo	Get the file information of the local file.	1.5.0
getFileSystemManager	Get the globally unique file manager.	1.5.0
getFileInfo	Get file information.	1.5.0
ReadResult	File reading result	1.5.16
WriteResult	File writing result	1.5.16

11.2、FileSystemManager

API	Note	Earliest Version

©2013-2022 Tencent Cloud. All rights reserved.

FileSystemManager.access	Determine if the file/directory exists.	1.5.0
FileSystemManager.accessSync	Synced version of FileSystemManager.access.	1.5.0
FileSystemManager.appendFile	Append Content at the End of the File.	1.5.0
FileSystemManager.appendFileSync	Synced version of FileSystemManager.appendFile.	1.5.0
FileSystemManager.close	Close the file.	1.5.16
FileSystemManager.closeSync	Synchronously close the file.	1.5.16
FileSystemManager.copyFile	Copy the file.	1.5.0
FileSystemManager.copyFileSync	Synced version of FileSystemManager.copyFile.	1.5.0
FileSystemManager.fstat	Get the status information of the file.	1.5.16
FileSystemManager.fstatSync	Synchronously get the status information of the file.	1.5.16
FileSystemManager.ftruncate	Truncate the content of the file using FileSystemManager.open.	1.5.16
FileSystemManager.ftruncateSync	Truncate the file content.	1.5.16
FileSystemManager.getFileInfo	Get information on local temporary files or local cache files under this mini program.	1.5.0
FileSystemManager.getSavedFileList	Get the list of locally cached files saved under this mini program.	1.5.0
FileSystemManager.mkdir	Create a directory.	1.5.0
FileSystemManager.mkdirSync	Synced version of FileSystemManager.mkdir.	1.5.0
FileSystemManager.open	Open the file and return the file descriptor.	1.5.16
FileSystemManager.openSync	Synchronously open the file and return the file descriptor.	1.5.16
FileSystemManager.read	Read the file.	1.5.16
FileSystemManager.readSync	Read the file.	1.5.16

©2013-2022 Tencent Cloud. All rights reserved.

FileSystemManager.readCompressedFile	Read the content of a local file with a specified compression type.	1.5.16
FileSystemManager.readCompressedFileSync	Synchronously read the content of a local file with a specified compression type.	1.5.16
FileSystemManager.readdir	Get the list of files within a directory.	1.5.0
FileSystemManager.readdirSync	Synced version of FileSystemManager.readdir.	1.5.0
FileSystemManager.readFile	Get the content of a local file.	1.5.0
FileSystemManager.readFileSync	Synced version of FileSystemManager.readFile.	1.5.0
FileSystemManager.readZipEntry	Read the files within a compressed package.	1.5.16
FileSystemManager.removeSavedFile	Delete the locally saved cache files under this mini program.	1.5.0
FileSystemManager.rename	Rename a file.	1.5.0
FileSystemManager.renameSync	Synced version of FileSystemManager.rename.	1.5.0
FileSystemManager.rmdir	Delete a directory.	1.5.0
FileSystemManager.rmdirSync	Synced version of FileSystemManager.rmdir.	1.5.0
FileSystemManager.saveFile	Save the temporary file locally.	1.5.0
FileSystemManager.saveFileSync	Synced version of FileSystemManager.saveFile.	1.5.0
FileSystemManager.stat	Get the Stats object of the file.	1.5.0
FileSystemManager.statSync	Synced version of FileSystemManager.stat.	1.5.0
FileSystemManager.truncate	Truncate the file content.	1.5.16
FileSystemManager.truncateSync	Truncate the file content (synced version of truncate).	1.5.16
FileSystemManager.unlink	Delete files.	1.5.0
FileSystemManager.unlinkSync	Synced version of FileSystemManager.unlink.	1.5.0

©2013-2022 Tencent Cloud. All rights reserved.

FileSystemManager.unzip	Decompress files.	1.5.0
FileSystemManager.write	Write in files.	1.5.16
FileSystemManager.writeSync	Synchronously write in the files.	1.5.16
FileSystemManager.writeFile	Write files.	1.5.0
FileSystemManager.writeFileSync	Synced version of FileSystemManager.writeFile.	1.5.0

11.3、Stats

API	Note	Earliest Version
Stats.isDirectory	Determines if the current file is a directory.	1.5.0
Stats.isFile	Determines if the current file is a common file.	1.5.0

12. Open Interfaces

12.1. Login

Name	Feature Description	Earliest Version
login	Call the interface to get the login credentials (code).	1.5.1
checkSession	Verify if the login status has expired.	1.5.1

12.2 Account Information

Name	Feature Description	Earliest Version
getAccountInfoSync	Get the current account information.	1.5.1

12.3 User Information

Name	Feature Description	Earliest Version

©2013-2022 Tencent Cloud. All rights reserved.

getUserProfile	Get user information.	1.5.1
getUserInfo	Get user information, which needs the user to authorize scope.userInfo.	1.5.1
userInfo	User information	1.5.0

12.4. Settings

Name	Feature Description	Earliest Version
AuthSetting	User settings authorization information	1.5.0
getSetting	Get the current settings of the user.	1.5.0
openSetting	Call the client-side mini program settings interface, returning the results of the user's configuration actions.	1.5.0

12.5 Biometric Authentication

Name	Feature Description	Earliest Version
checkIsSoterEnrolledInDevice	Verifies whether biometric information has been entered into the device	1.5.0
checkIsSupportSoterAuthentication	Verifies whether the device supports biometric authentication	1.5.0
startSoterAuthentication	Initiates biometric authentication	1.5.0

12.6. Authorization

Name	Feature Description	Earliest Version
authorize	Initiate an authorization request to the user in advance.	1.5.0

13. Location

©2013-2022 Tencent Cloud. All rights reserved.

Name	Feature Description	Earliest Version
getLocation	Get the current geographical location and velocity.	1.5.0
choosePoi	Open the POI list to select a location, which supports both vague positioning (accurate to the city) and precise positioning in a mixed selection.	1.5.0
chooseLocation	Open the map to select a location.	1.5.0
stopLocationUpdate	Cease monitoring real-time location changes, stopping message reception in both the frontend and backend.	1.5.0
startLocationUpdateBackground	Enable the mini program to receive location messages when entering the frontend and backend, which requires guiding the user to give "authorization".	1.5.0
startLocationUpdate	Enable the mini program to receive location messages upon entering the frontend.	1.5.0
openLocation	Use the built-in We Chat map to view the location.	1.5.0
onLocationChangeError	Triggered when the continuous positioning interface being monitored fails to return a response.	1.5.0
onLocationChange	Monitor real-time geographical location change events, necessitating the combined use of startLocationUpdateBackground and startLocationUpdate.	1.5.0
offLocationChangeError	Triggered when a failure message is returned in removing the continuous positioning interface.	1.5.0
offLocationChange	Stop monitoring real-time geographical location change events.	1.5.0
getFuzzyLocation	Get the current approximate geographical location.	1.5.0

14. Device

14.1 General Bluetooth

©2013-2022 Tencent Cloud. All rights reserved.

Name	Feature Description	Earliest Version
stopBluetoothDevicesDiscovery	Cease the search for nearby peripheral Bluetooth devices.	1.5.0
startBluetoothDevicesDiscovery	Start the search for nearby peripheral Bluetooth devices.	1.5.0
openBluetoothAdapter	Initialize the Bluetooth module.	1.5.1
getConnectedBluetoothDevices	Get connected Bluetooth devices based on the primary service UUID.	1.5.1
getBluetoothDevices	Get all Bluetooth devices detected during the effective period of the Bluetooth module.	1.5.1
getBluetoothAdapterState	Get the status of the local Bluetooth adapter.	1.5.1
closeBluetoothAdapter	Deactivate the Bluetooth module.	1.5.1
onBluetoothDeviceFound	Monitor events of finding new devices.	1.5.0
offBluetoothDeviceFound	Remove all listener functions for events of finding new devices.	1.5.0
onBluetoothAdapterStateChange	Monitor events of Bluetooth adapter status changes.	1.5.0
offBluetoothAdapterStateChange	Remove all listener functions for events of the Bluetooth adapter status changes.	1.5.0
makeBluetoothPair	Bluetooth pairing interface, supported exclusively on Android.	1.5.0
isBluetoothDevicePaired	Inquire if the Bluetooth device is paired, supported exclusively on Android.	1.5.0

14.2. Bluetooth - Low Energy Central Device

Name	Feature Description	Earliest Version
writeBLECharacteristicValue	Write binary data into the characteristic value of low-energy Bluetooth devices.	1.5.1
readBLECharacteristicValue	Read the binary data of the characteristic value from low-energy Bluetooth devices.	1.5.1
setBLEMTU	Negotiate the setting of the maximum unit for low-energy Bluetooth transmission.	1.5.1

©2013-2022 Tencent Cloud. All rights reserved.

getBLEMTU	Get the maximum unit for low-energy Bluetooth transmission.	1.5.1
onBLEMTUChange	Monitor events of maximum unit changes for low-energy Bluetooth transmission (triggered on Android only).	1.5.1
offBLEMTUChange	Remove the listener function for events of maximum unit changes for low-energy Bluetooth transmission.	1.5.1
onBLEConnectionStateChange	Monitor events of changes to low-energy Bluetooth connection status.	1.5.1
offBLEConnectionStateChange	Remove the listener function for events of changes to low-energy Bluetooth connection status.	1.5.1
onBLECharacteristicValueChange	Monitor events of characteristic value changes for low-energy Bluetooth devices.	1.5.1
offBLECharacteristicValueChange	Remove all listener functions for events of characteristic value changes for low-energy Bluetooth devices.	1.5.1
notifyBLECharacteristicValueChange	Enable the "notify" feature when the characteristic value of low-energy Bluetooth device changes, subscribing to the characteristics.	1.5.1
getBLEDeviceServices	Get all services from low-energy Bluetooth devices.	1.5.1
getBLEDeviceRSSI	Get the signal strength of low-energy Bluetooth devices.	1.5.1
getBLEDeviceCharacteristics	Get all characteristics within a specific service of low-energy Bluetooth devices.	1.5.1
createBLEConnection	Establish connection with a low-energy Bluetooth device.	1.5.1
closeBLEConnection	Disconnect with the low-energy Bluetooth device.	1.5.1

14.3 Bluetooth - Low Energy Peripheral Device

Name	Feature Description	Earliest Version

©2013-2022 Tencent Cloud. All rights reserved.

onBLEPeripheralConnectionStateChanged	Monitor the current peripheral device for connection or disconnection events.	1.5.1
offBLEPeripheralConnectionStateChanged	Remove the listener function for the connection or disconnection events of the current peripheral device.	1.5.1
createBLEPeripheralServer	Establish a local server as a low-energy Bluetooth peripheral device.	1.5.1
BLEPeripheraServer.addService	Add service	1.5.1
BLEPeripheraServer.removeService	Remove service	1.5.1
BLEPeripheraServer.startAdvertising	Initiate broadcasting of locally created peripheral devices.	1.5.1
BLEPeripheraServer.stopAdvertising	Stop broadcasting	1.5.1
BLEPeripheraServer..writeCharacteristicValue	Write binary data values into the specified characteristic and notify the connected host.	1.5.1
BLEPeripheraServer.onCharacteristicWriteRequest	Monitor events where the connected device requests to write the characteristic value of the current peripheral device.	1.5.1
BLEPeripheraServer.offCharacteristicWriteRequest	Remove the listener function for events associated with characteristic value changes of the currently connected peripheral device.	1.5.1
BLEPeripheraServer.onCharacteristicReadRequest	Monitor events where the connected device requests to read the characteristic value of the current peripheral device.	1.5.1
BLEPeripheraServer.offCharacteristicReadRequest	Remove the listener function for events where the connected device requests to read the characteristic value of the current peripheral device.	1.5.1
BLEPeripheraServer.onCharacteristicSubscribed	Monitor characteristic subscription events, supported exclusively on iOS.	1.5.1
BLEPeripheraServer.offCharacteristicSubscribed	Remove the listener function for characteristic subscription events.	1.5.1

©2013-2022 Tencent Cloud. All rights reserved.

BLEPeripheraServer.onCharacteristicUnsubscribed	Monitor characteristic unsubscription events, supported exclusively on iOS.	1.5.1
BLEPeripheraServer.offCharacteristicUnsubscribed	Remove the listener function for characteristic unsubscription events.	1.5.1

14.4 Bluetooth - Beacon

Name	Feature Description	Earliest Version
stopBeaconDiscovery	Stop the search for nearby Beacon devices.	1.5.1
startBeaconDiscovery	Start the search for nearby Beacon devices.	1.5.1
onBeaconUpdate	Monitor Beacon device update events, with only one listener registration permissible.	1.5.1
offBeaconUpdate	Remove all listener functions for Beacon device update events.	1.5.1
onBeaconServiceChange	Monitor events of Beacon service status changes, with only one listener registration permissible.	1.5.1
offBeaconServiceChange	Remove all listener functions for events of Beacon service status changes.	1.5.1
getBeacons	Get all Beacon devices that have been detected.	1.5.1
BeaconInfo	Beacon devices	1.5.1

14.5、NFC

14.5.1、getNFCAdapter

Name	Feature Description	Earliest Version
getNFCAdapter	Get the NFC instance.	1.5.0

14.5.2、NFCAdapter

Name	Feature Description	Earliest Version
NFCAdapter.startDiscovery	Turn on the device.	1.5.0
NFCAdapter.stopDiscovery	Turn off the device.	1.5.0

©2013-2022 Tencent Cloud. All rights reserved.

NFCAdapter.getNdef	Get the Ndef instance, which supports reading and writing NDEF data on NFC tags in NDEF format.	1.5.0
NFCAdapter.getNfcA	Get the NfcA instance, which supports reading and writing in accordance with the NFC-A (ISO 14443-3A) standard.	1.5.0
NFCAdapter.getNfcB	Get the NfcB instance, which supports reading and writing in accordance with the NFC-B (ISO 14443-3B) standard.	1.5.0
NFCAdapter.getNfcF	Get the NfcF instance, which supports reading and writing in accordance with the NFC-F (JIS 6319-4) standard.	1.5.0
NFCAdapter.getNfcV	Get the NfcV instance, which supports reading and writing in accordance with the NFC-V (ISO 15693) standard.	1.5.0
NFCAdapter.getIsoDep	Get the IsoDep instance, which supports reading and writing in accordance with the ISO-DEP (ISO 14443-4) standard.	1.5.0
NFCAdapter.getMifareClassic	Get the MifareClassic instance, which supports reading and writing of MIFARE Classic tags.	1.5.0
NFCAdapter.getMifareUltralight	Get the MifareUltralight instance, which supports reading and writing of MIFARE Ultralight tags.	1.5.0
NFCAdapter.onDiscovered	Monitor NFC Tags.	1.5.0
NFCAdapter.offDiscovered	Remove the listener function of NFC Tags.	1.5.0

14.5.3、IsoDep

Name	Feature Description	Earliest Version
IsoDep.connect	Connect with NFC tags.	1.5.0
IsoDep.close	Disconnect	1.5.0
IsoDep.setTimeout	Set a timeout period.	1.5.0
IsoDep.isConnected	Verify whether the connection has been established.	1.5.0
IsoDep.getMaxTransceiveLength	Determine the maximum transmission length.	1.5.0
IsoDep.transceive	Transmit data,	1.5.0

©2013-2022 Tencent Cloud. All rights reserved.

IsoDep.getHistoricalBytes	Get reset information.	1.5.0

14.5.4、MifareClassic

Name	Feature Description	Earliest Version
MifareClassic.connect	Connect with NFC tags.	1.5.0
MifareClassic.close	Disconnect	1.5.0
MifareClassic.setTimeout	Set a timeout period.	1.5.0
MifareClassic.isConnected	Verify whether the connection has been established.	1.5.0
MifareClassic.getMaxTransceiveLength	Determine the maximum transmission length.	1.5.0
MifareClassic.transceive	Transmit data,	1.5.0

14.5.5、MifareUltralifght

Name	Feature Description	Earliest Version
MifareUltralight.connect	Connect with NFC tags.	1.5.0
MifareUltralight.close	Disconnect	1.5.0
MifareUltralight.setTimeout	Set a timeout period.	1.5.0
MifareUltralight.isConnected	Verify whether the connection has been established.	1.5.0
MifareUltralight.getMaxTransceiveLength	Determine the maximum transmission length.	1.5.0
MifareUltralight.transceive	Transmit data,	1.5.0

14.5.6、Ndef

Name	Feature Description	Earliest Version
Ndef.connect	Connect with NFC tags.	1.5.0
Ndef.close	Disconnect	1.5.0
Ndef.setTimeout	Set a timeout period.	1.5.0

©2013-2022 Tencent Cloud. All rights reserved.

Ndef.isConnected	Verify whether the connection has been established.	1.5.0
Ndef.offNdefMessage	Stop monitoring of Ndef messages.	1.5.0
Ndef.onNdefMessage	Monitor Ndef messages.	1.5.0
Ndef.writeNdefMessage	Re-write the content of the Ndef tag.	1.5.0

14.5.7、NfcA

Name	Feature Description	Earliest Version
NfcA.connect	Connect with NFC tags.	1.5.0
NfcA.close	Disconnect	1.5.0
NfcA.setTimeout	Set a timeout period.	1.5.0
NfcA.isConnected	Verify whether the connection has been established.	1.5.0
NfcA.getMaxTransceiveLength	Determine the maximum transmission length.	1.5.0
NfcA.transceive	Transmit data,	1.5.0
NfcA.getAtqa	Get ATQA information.	1.5.0
NfcA.getSak	Get SAK information.	1.5.0

14.5.8、NfcB

Name	Feature Description	Earliest Version
NfcB.connect	Connect with NFC tags.	1.5.0
NfcB.close	Disconnect	1.5.0
NfcB.setTimeout	Set a timeout period.	1.5.0
NfcB.isConnected	Verify whether the connection has been established.	1.5.0
NfcB.getMaxTransceiveLength	Determine the maximum transmission length.	1.5.0
NfcB.transceive	Transmit data,	1.5.0

14.5.9、NfcF

Name	Feature Description	Earliest Version

©2013-2022 Tencent Cloud. All rights reserved.

NfcF.connect	Connect with NFC tags.	1.5.0
NfcF.close	Disconnect	1.5.0
NfcF.setTimeout	Set a timeout period.	1.5.0
NfcF.isConnected	Verify whether the connection has been established.	1.5.0
NfcF.getMaxTransceiveLength	Determine the maximum transmission length.	1.5.0
NfcF.transceive	Transmit data,	1.5.0

14.5.10、NfcV

Name	Feature Description	Earliest Version
NfcV.connect	Connect with NFC tags.	1.5.0
NfcV.close	Disconnect	1.5.0
NfcV.setTimeout	Set a timeout period.	1.5.0
NfcV.isConnected	Verify whether the connection has been established.	1.5.0
NfcV.getMaxTransceiveLength	Determine the maximum transmission length.	1.5.0
NfcV.transceive	Transmit data,	1.5.0

14.6、WiFi

Name	Feature Description	Earliest Version
onWifiConnected	Monitor events of establishing Wi-Fi connection.	1.5.1
offWifiConnected	Remove the listener function for events of establishing Wi-Fi connection.	1.5.1
stopWifi	Disable the Wi-Fi module.	1.5.0
startWifi	Initialize the Wi-Fi module.	1.5.0
onGetWifiList	Monitor events of receiving Wi-Fi list data.	1.5.0
offGetWifiList	Remove the listener function for events of receiving Wi-Fi list data.	1.5.0
getConnectedWifi	Get information about the currently connected Wi-Fi.	1.5.0

©2013-2022 Tencent Cloud. All rights reserved.

WifiInfo	Wi-Fi Information	1.5.1

14.7. Calendar

Name	Feature Description	Earliest Version
addPhoneRepeatCalendar	Add recurring events to the system calendar.	1.5.1
addPhoneCalendar	Add events to the system calendar.	1.5.1

14.8 Clipboard

Name	Feature Description	Earliest Version
setClipboardData	Set the content of the system clipboard.	1.5.1
getClipboardData	Get the content of the system clipboard.	1.5.1

14.9. Network

Name	Feature Description	Earliest Version
onNetworkStatusChange	Monitor network status change events.	1.5.1
getNetworkType	Get network type information.	1.5.1
offNetworkStatusChange	Remove the listener function for network status change events.	1.5.1

14.10 Encryption

Name	Feature Description	Earliest Version
getRandomValues	Gets cryptographically secure random numbers	1.5.1

14.11 Keyboard

Name	Feature Description	Earliest Version
onKeyboardHeightChange	Monitor keyboard height change events.	1.5.1
hideKeyboard	After the keyboard is triggered by focusing on input, textarea, etc., manually invoke this interface to retract the keyboard.	1.5.1

©2013-2022 Tencent Cloud. All rights reserved.

offKeyboardHeightChange	Remove the listener function for keyboard height change events.	1.5.1
getSelectedTextRange	After focusing on input, textarea, etc., get the cursor position in the input box.	1.5.1

14.12 Phone Call

Name	Feature Description	Earliest Version
makePhoneCall	Make a phone call.	1.5.1

14.13 Accelerometer

Name	Feature Description	Earliest Version
startAccelerometer	Begin monitoring acceleration data.	1.5.1
stopAccelerometer	Stop monitoring acceleration data.	1.5.1
onAccelerometerChange	Monitor acceleration data events.	1.5.1
offAccelerometerChange	Remove the listener function for acceleration data events.	1.5.1

14.14 Compass

Name	Feature Description	Earliest Version
startCompass	Begin monitoring compass data.	1.5.0
stopCompass	Stop monitoring compass data.	1.5.0
onCompassChange	Monitor compass data change events.	1.5.0
offCompassChange	Remove the listener function for compass data change events.	1.5.0

14.15 Gyroscope

Name	Feature Description	Earliest Version
startGyroscope	Start monitoring gyroscope data.	1.5.1
stopGyroscope	Stop monitoring gyroscope data.	1.5.1

©2013-2022 Tencent Cloud. All rights reserved.

onGyroscopeChange	Monitor gyroscope data change events.	1.5.1
offGyroscopeChange	Remove the listener function for gyroscope data change events.	1.5.1

14.16 Scanning Code

Name	Feature Description	Earliest Version
scanCode	Invoke the client's scanning interface for code scanning.	1.5.0

14.17 SMS

Name	Feature Description	Earliest Version
sendSms	Initiate the mobile SMS sending interface.	1.5.1

14.18 Vibration

Name	Feature Description	Earliest Version
vibrateShort	Induce a brief vibration in the mobile phone (15 ms).	1.5.1
vibrateLong	Induce an extended vibration in the mobile phone (400 ms).	1.5.1

14.19 Contact

Name	Feature Description	Earliest Version
chooseContact	Initiate the phone's contact list and select a contact.	1.5.1
addPhoneContact	Add a contact to the mobile phone directory.	1.5.1

14.20 Battery Level

Name	Feature Description	Earliest Version
getBatteryInfoSync	Synced version of wx.getBatteryInfo.	1.5.1

©2013-2022 Tencent Cloud. All rights reserved.

getBatteryInfo	Get the device's battery level information.	1.5.1

14.21 Screen

Name	Feature Description	Earliest Version
setVisualEffectOnCapture	Configure screen performance during screenshot/screen recording.	1.5.1
setScreenBrightness	Adjust the screen brightness.	1.5.1
setKeepScreenOn	Set whether to maintain a constant brightness.	1.5.1
onUserCaptureScreen	Monitor user-initiated screenshot events.	1.5.1
onScreenRecordingStateChanged	Monitor user-initiated screen recording events.	1.5.1
offUserCaptureScreen	User-initiated screenshot events.	1.5.1
offScreenRecordingStateChanged	Remove the listener function for user-initiated screen recording events.	1.5.1
getScreenRecordingState	Inquire if the user is currently recording the screen.	1.5.1
getScreenBrightness	Get the screen brightness information.	1.5.1

14.22. Device Orientation

Name	Feature Description	Earliest Version
stopDeviceMotionListening	Stop monitoring changes in device orientation.	1.5.1
startDeviceMotionListening	Start monitoring changes in device orientation.	1.5.1
onDeviceMotionChange	Monitor device orientation change events.	1.5.1
offDeviceMotionChange	Remove the listener function for device orientation change events.	1.5.1

14.23 Memory

Name	Feature Description	Earliest Version
onMemoryWarning	Monitor insufficient memory alarm events.	1.5.1
offMemoryWarning	Remove the listener function for insufficient memory alarm events.	1.5.1

©2013-2022 Tencent Cloud. All rights reserved.

14.24 Accessibility

Name	Feature Description	Earliest Version
checkIsOpenAccessibility	Detect whether the visual accessibility feature is enabled.	1.5.1

15、WXML

15.1 WXML Overview

Name	Feature Description	Earliest Version
createSelectorQuery	Return an instance of the SelectorQuery object.	1.5.0
createIntersectionObserver	Create and return an instance of the IntersectionObserver object.	1.5.0

15.2、IntersectionObserve

Name	Feature Description	Earliest Version
IntersectionObserve.relativeTo	Use a selector to designate a node as one of the reference areas.	1.5.0
IntersectionObserve.relativeToViewport	Designate the page display area as one of the reference areas.	1.5.0
IntersectionObserve.disconnect	Stop monitoring and the callback function will no longer be triggered.	1.5.0
IntersectionObserve.observe	Designate the target node and start monitoring changes in intersection status.	1.5.0

15.3、NodesRef

Name	Feature Description	Earliest Version

©2013-2022 Tencent Cloud. All rights reserved.

NodesRef.fields	Get pertinent information related to the node.	1.5.0
NodesRef.boundingClientRect	Add a query request for the layout position of the node.	1.5.0
NodesRef.scrollOffset	Add a query request for the scrolling position of the node.	1.5.0
NodesRef.context	Add a query request for the Context object of the node.	1.5.0
NodesRef.node	Get the Node instance.	1.5.0

15.4、SelectorQuery

Name	Feature Description	Earliest Version
SelectorQuery.exec	Execute all the requests.	1.5.0
SelectorQuery.in	Change the selection range of the selector to be within the custom component.	1.5.0
SelectorQuery.select	Select the first node that matches the selector within the current page.	1.5.0
SelectorQuery.selectAll	Select all nodes that match the selector within the current page.	1.5.0
SelectorQuery.selectViewport	Select the display area.	1.5.0

15.5、meidaQueryObserver

Name	Feature Description	Earliest Version
MediaQueryObserver.disconnect	Stop monitoring.	1.5.16
MediaQueryObserver.observe	Start monitoring media query changes.	1.5.16

16. Custom APIs

©2013-2022 Tencent Cloud. All rights reserved.

Name	Feature Description	Earliest Version
invokeNativePlugin	Custom APIs	1.5.0

©2013-2022 Tencent Cloud. All rights reserved.

Fundamentals

Last updated：2024-07-12 19:39:04

canIUse

This API is used via Boolean wx.canIUse(string schema).

Feature Description: Determines whether the API, callbacks, parameters, components, etc., of the mini program are usable in the current version.

Parameter and Description: String schema:

Invoke using the `${API}.${method}.${param}.${options}` or `${component}.${attribute}.${option}` method.

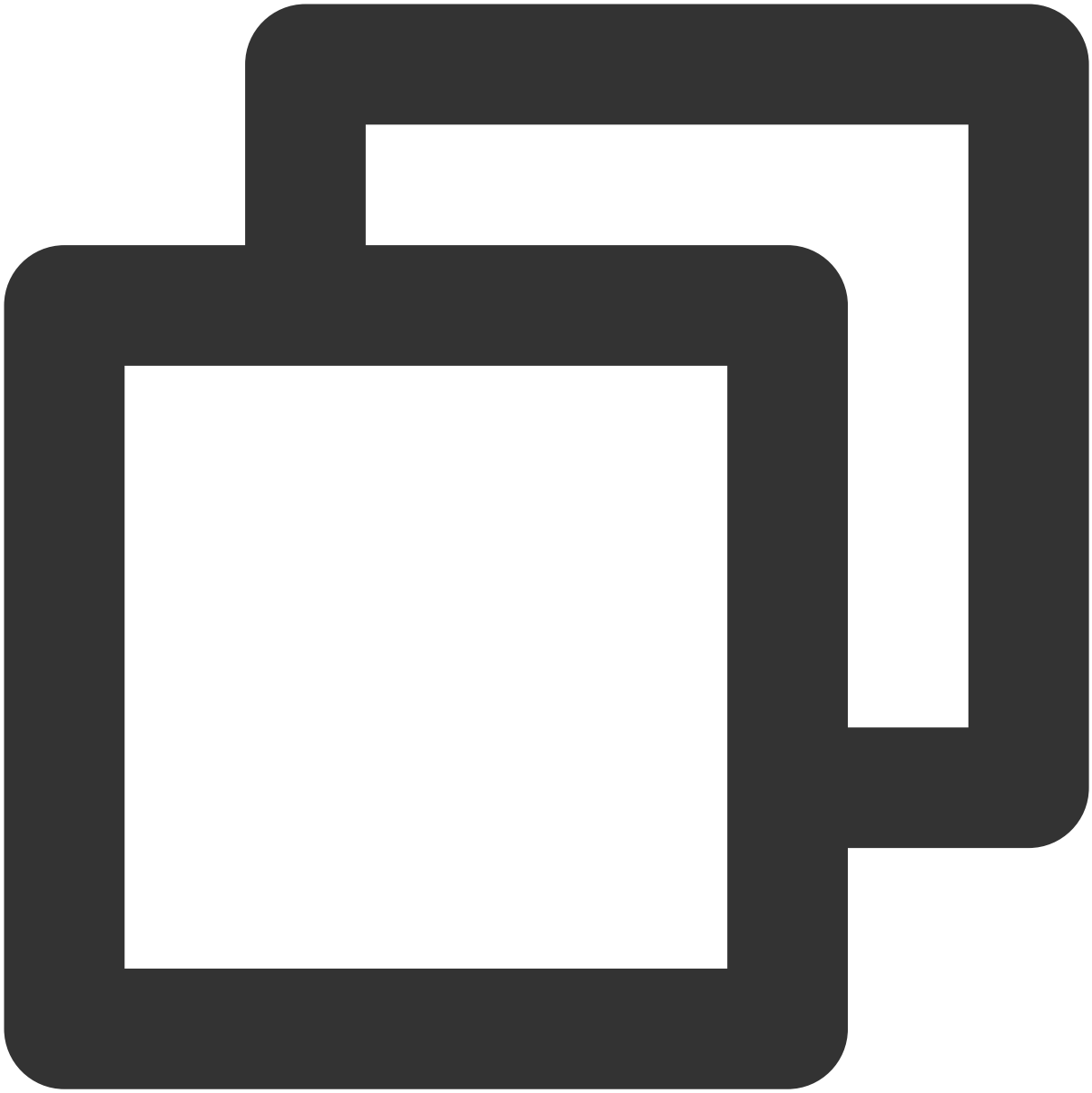
Description:

Value	Description
${API}	Represents the API name.
${method}	Represents the invocation method, with valid values being return, success, object, callback.
${param}	Represents parameters or return values.
${options}	Represents the optional values of the parameters.
${component}	Represents the component name.
${attribute}	Represents the component attribute.
${option}	Represents the optional values of the component attribute.

Return Value: Boolean, indicating whether the current version is available.

Sample Code

©2013-2022 Tencent Cloud. All rights reserved.



```
wx.canIUse('openBluetoothAdapter')
wx.canIUse('getSystemInfoSync.return.screenWidth')
wx.canIUse('getSystemInfo.success.screenWidth')
wx.canIUse('showToast.object.image')
wx.canIUse('onCompassChange.callback.direction')
wx.canIUse('request.object.method.GET')

wx.canIUse('live-player')
wx.canIUse('text.selectable')
wx.canIUse('button.open-type.contact')
```

©2013-2022 Tencent Cloud. All rights reserved.

env

This API is used via wx.env.

Feature Description: Environment variable.

Parameter and Description: string USER_DATA_PATH, the user directory path in the file system (local path).

base64ToArrayBuffer

This API is used via ArrayBuffer wx.base64ToArrayBuffer(string base64).

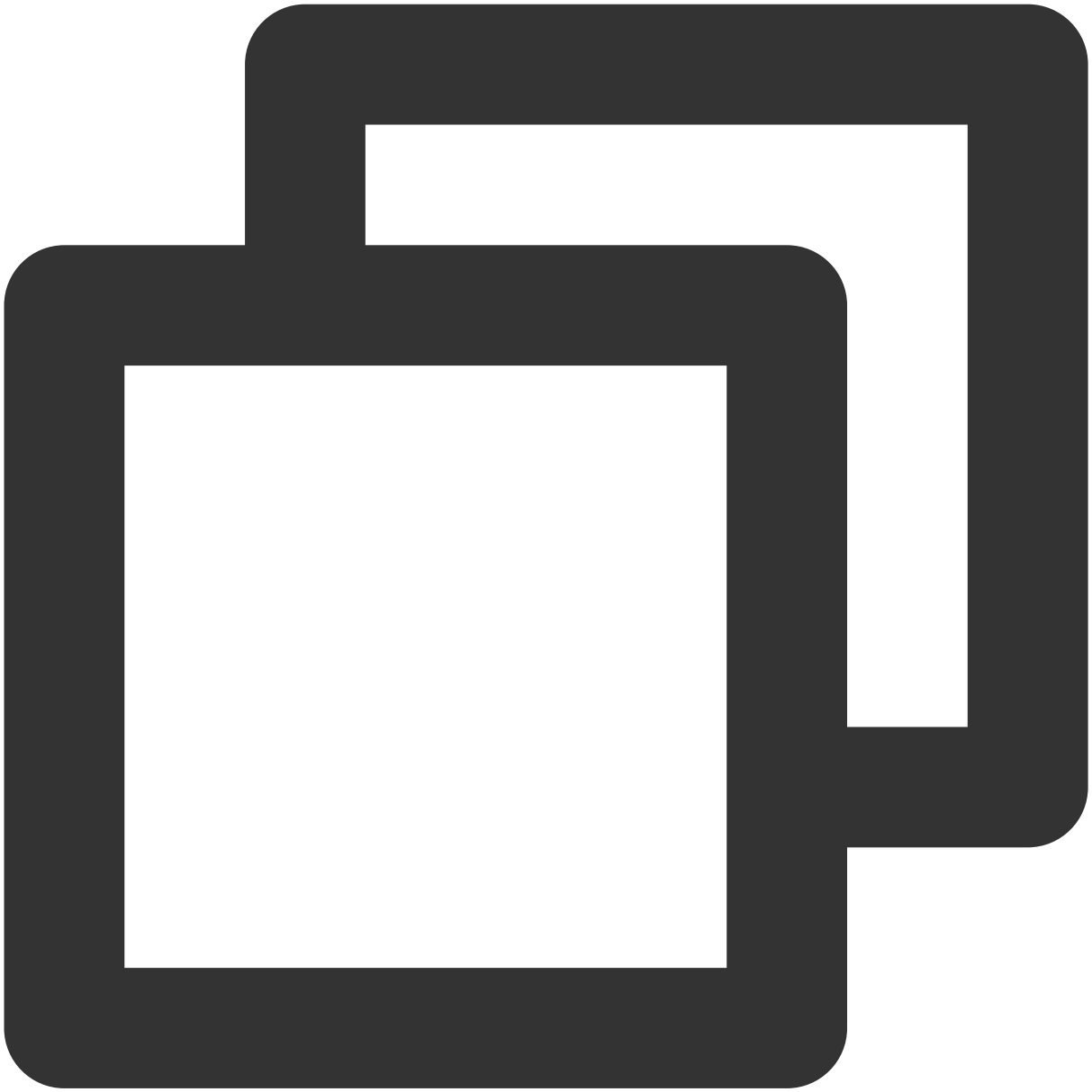
Feature Description: Converts a Base64 string into an ArrayBuffer object.

Parameter and Description: string base64, the Base64 string to be converted into an ArrayBuffer object.

Return Value: ArrayBuffer, an ArrayBuffer object.

Sample Code

©2013-2022 Tencent Cloud. All rights reserved.



```
const base64 = 'CxYh'
const arrayBuffer = wx.base64ToArrayBuffer(base64)
```

arrayBufferToBase64

This API is used via string wx.arrayBufferToBase64(ArrayBuffer arrayBuffer).

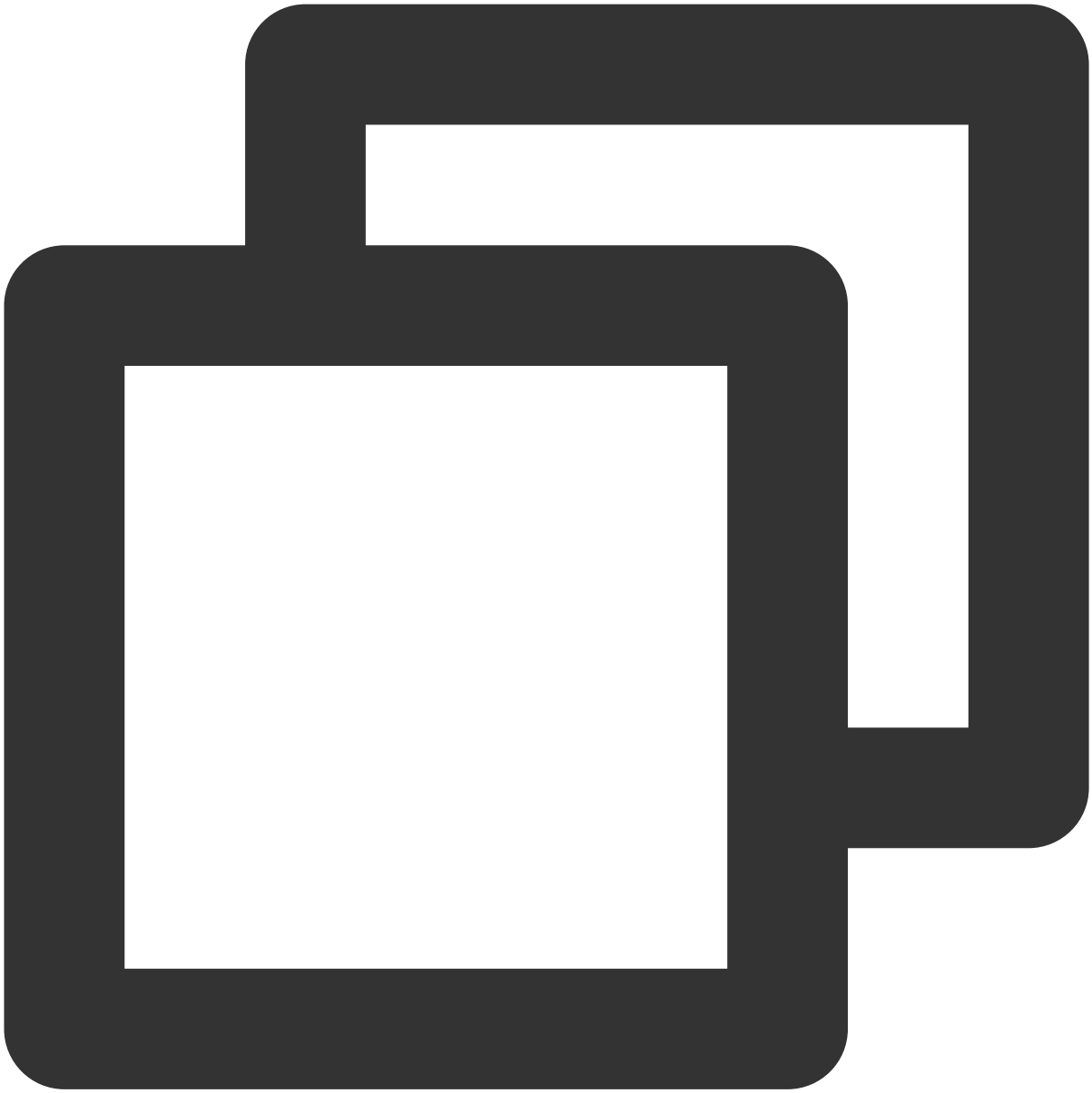
©2013-2022 Tencent Cloud. All rights reserved.

Feature Description: Converts an ArrayBuffer object into a Base64 string.

Parameter and Description: ArrayBuffer, the ArrayBuffer object to be converted into a Base64 string.

Return Value: string, a Base64 string.

Sample Code



```
const arrayBuffer = new Uint8Array([11, 22, 33])
const base64 = wx.arrayBufferToBase64(arrayBuffer)
```

©2013-2022 Tencent Cloud. All rights reserved.

System

getSystemInfo

This API is used via wx.getSystemInfo(Object object).

Feature Description: Gets system information.

Parameter and Description: Object.

Attribute	Type	Default value	Required	Description
success	Function	-	No	Callback Function of Successful Interface Call
fail	Function	-	No	Callback Function of Failing Interface Call
complete	Function	-	No	Callback function executed upon the completion of the interface invocation (regardless of success or failure of the invocation)

Callback result of Object.success's res

Attribute	Type	Description
brand	String	Device Brand
Device Model	String	Device Model
pixelRatio	Number	Device Pixel Ratio
screenWidth	Number	Screen width, measured in pixels (px)
screenHeight	Number	Screen height, measured in pixels (px)
windowWidth	Number	Usable window width, measured in pixels (px)
windowHeight	Number	Usable window height, measured in pixels (px)
statusBarHeight	Number	Status bar height, measured in pixels (px)
language	String	Language
version	String	Version No.
system	String	Operating System and Version

©2013-2022 Tencent Cloud. All rights reserved.

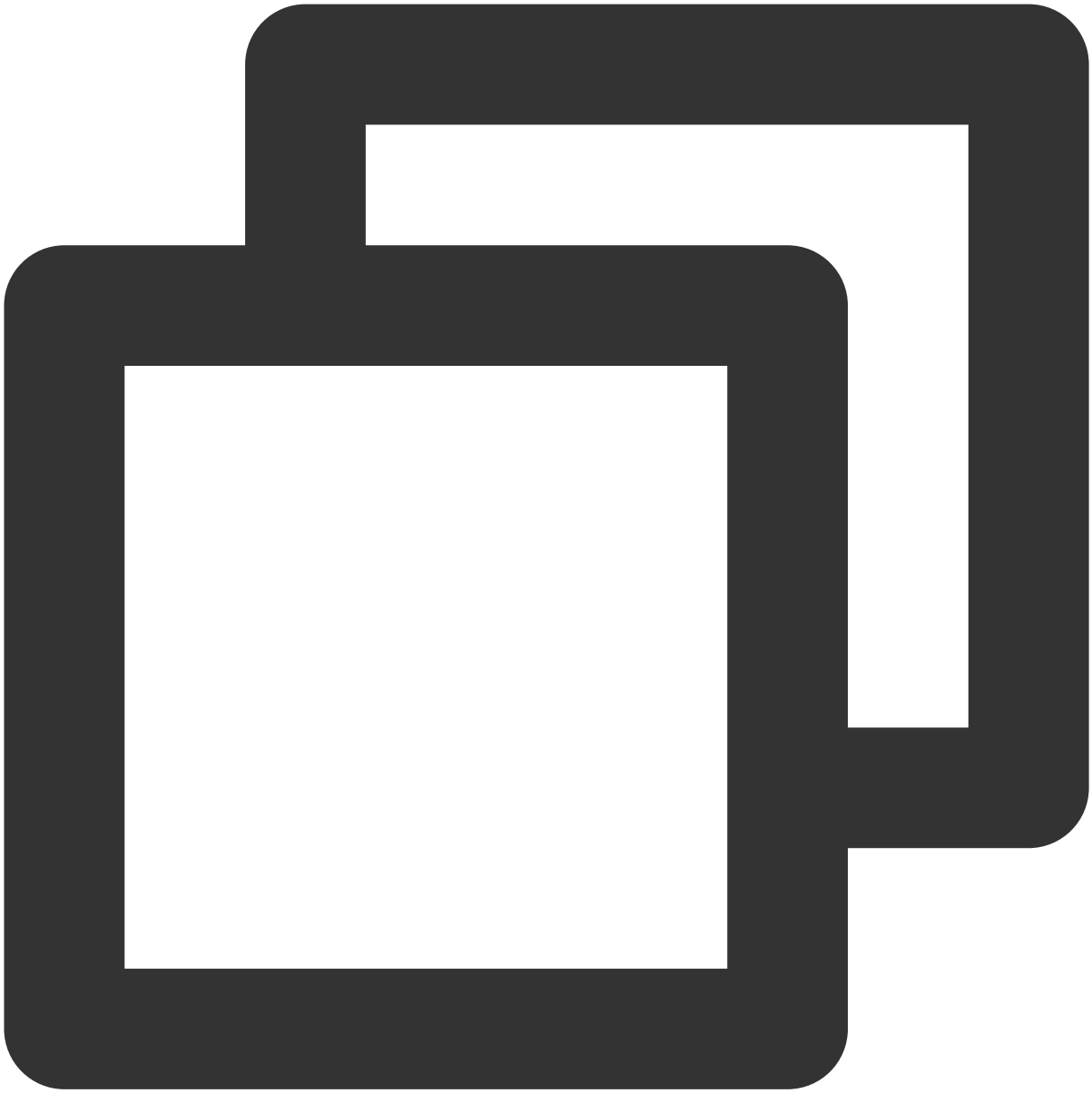
platform	String	User font size in px
SDKVersion	String	Client Base Library Version
App Platform	String	App Platform
safeArea	Object	Safe area in the upright orientation
theme	String	The current system theme, `values can be either light or dark`. It can only be obtained when the global configuration `"darkmode"` is set to "true". Otherwise, it is undefined.

Structure of res.safeArea:

Attribute	Type	Description
left	Number	Horizontal coordinate of the top left corner of the safe area
right	Number	Horizontal coordinate of the bottom right corner of the safe area
top	Number	Vertical coordinate of the top left corner of the safe area
bottom	Number	Vertical coordinate of the bottom right corner of the safe area
width	Number	Width of the safe area, measured in logical pixels
height	Number	Height of the safe area, measured in logical pixels

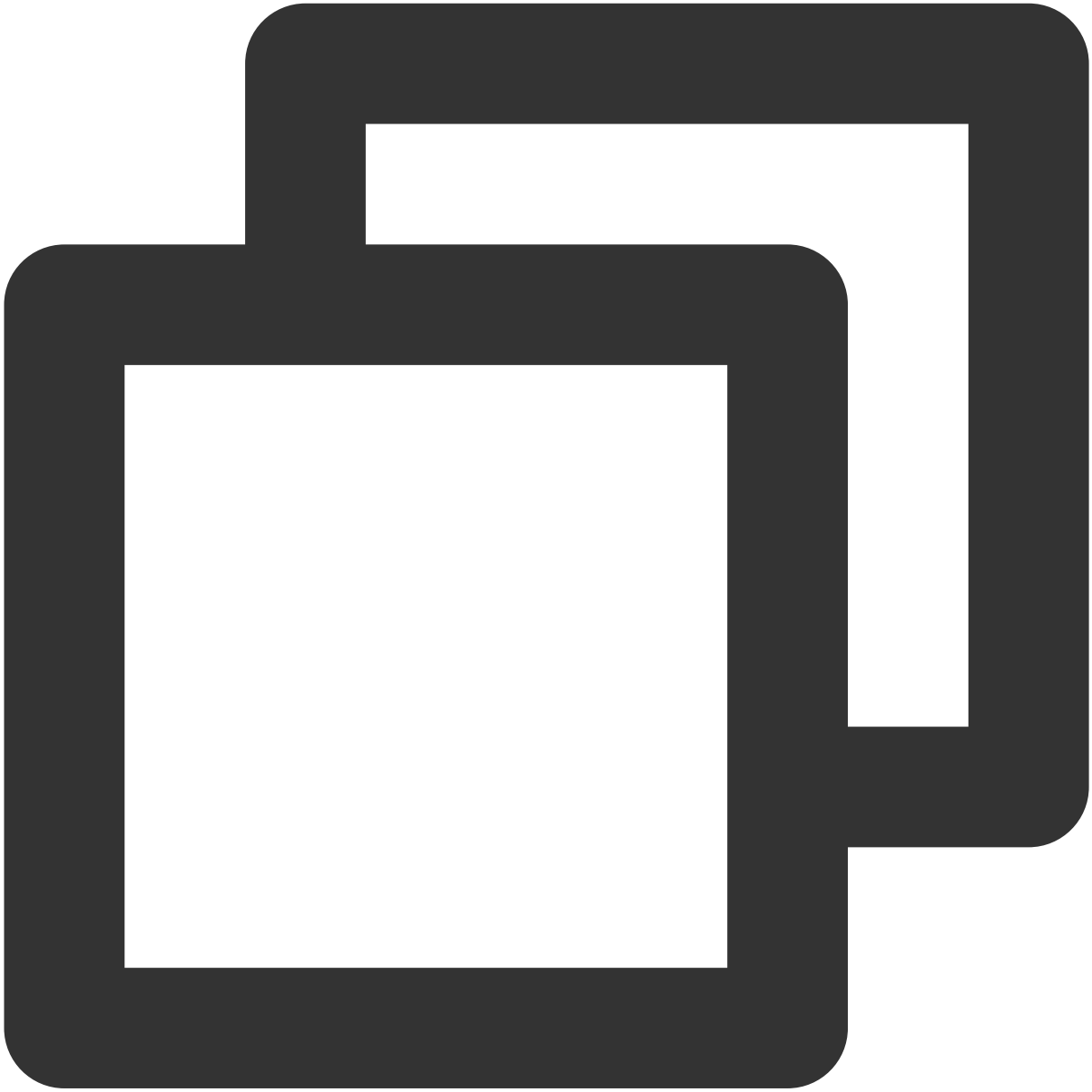
Sample Code

©2013-2022 Tencent Cloud. All rights reserved.



```
wx.getSystemInfo({
  success(res) {
    console.log(res.model)
    console.log(res.pixelRatio)
    console.log(res.windowWidth)
    console.log(res.windowHeight)
    console.log(res.language)
    console.log(res.version)
    console.log(res.platform)
  }
})
```

©2013-2022 Tencent Cloud. All rights reserved.



```
try {
  const res = wx.getSystemInfoSync()
  console.log(res.model)
  console.log(res.pixelRatio)
  console.log(res.windowWidth)
  console.log(res.windowHeight)
  console.log(res.language)
  console.log(res.version)
  console.log(res.platform)
} catch (e) {
```

©2013-2022 Tencent Cloud. All rights reserved.

```
  // Do something when catch error
}
```

getSystemInfoSync

This API is used via Object wx.getSystemInfo().

Feature Description: Synced version of wx.getSystemInfo.

Callback result of Object.success's res

Attribute	Type	Description
brand	String	Device Brand
Device Model	String	Device Model
pixelRatio	Number	Device Pixel Ratio
screenWidth	Number	Screen width, measured in pixels (px)
screenHeight	Number	Screen height, measured in pixels (px)
windowWidth	Number	Usable window width, measured in pixels (px)
windowHeight	Number	Usable window height, measured in pixels (px)
statusBarHeight	Number	Status bar height, measured in pixels (px)
language	String	Language
version	String	Version No.
system	String	Operating System and Version
platform	String	Client platform, with the following valid values: ios: iOS client (including iPhone, iPad) Android: Android client devtools: TCMPP Developer Tools
App Platform	String	App Platform
safeArea	Object	Safe area in the upright orientation
theme	String	The current system theme, `values can be either light or dark` . It can only be obtained when the global configuration

©2013-2022 Tencent Cloud. All rights reserved.

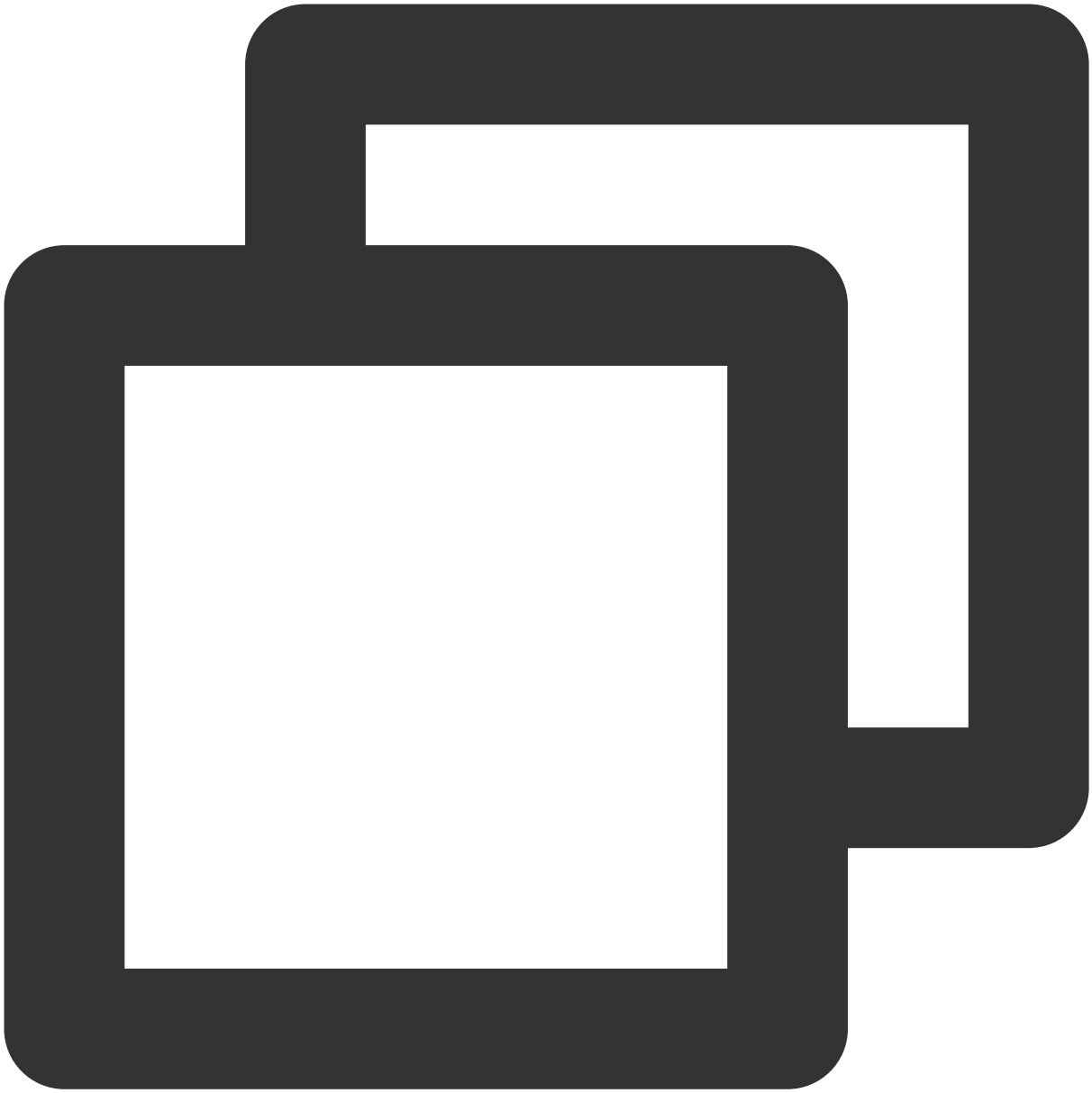
		`"darkmode"` is set to "true". Otherwise, it is undefined.

Structure of res.safeArea:

Attribute	Type	Description
left	Number	Horizontal coordinate of the top left corner of the safe area
right	Number	Horizontal coordinate of the bottom right corner of the safe area
top	Number	Vertical coordinate of the top left corner of the safe area
bottom	Number	Vertical coordinate of the bottom right corner of the safe area
width	Number	Width of the safe area, measured in logical pixels
height	Number	Height of the safe area, measured in logical pixels

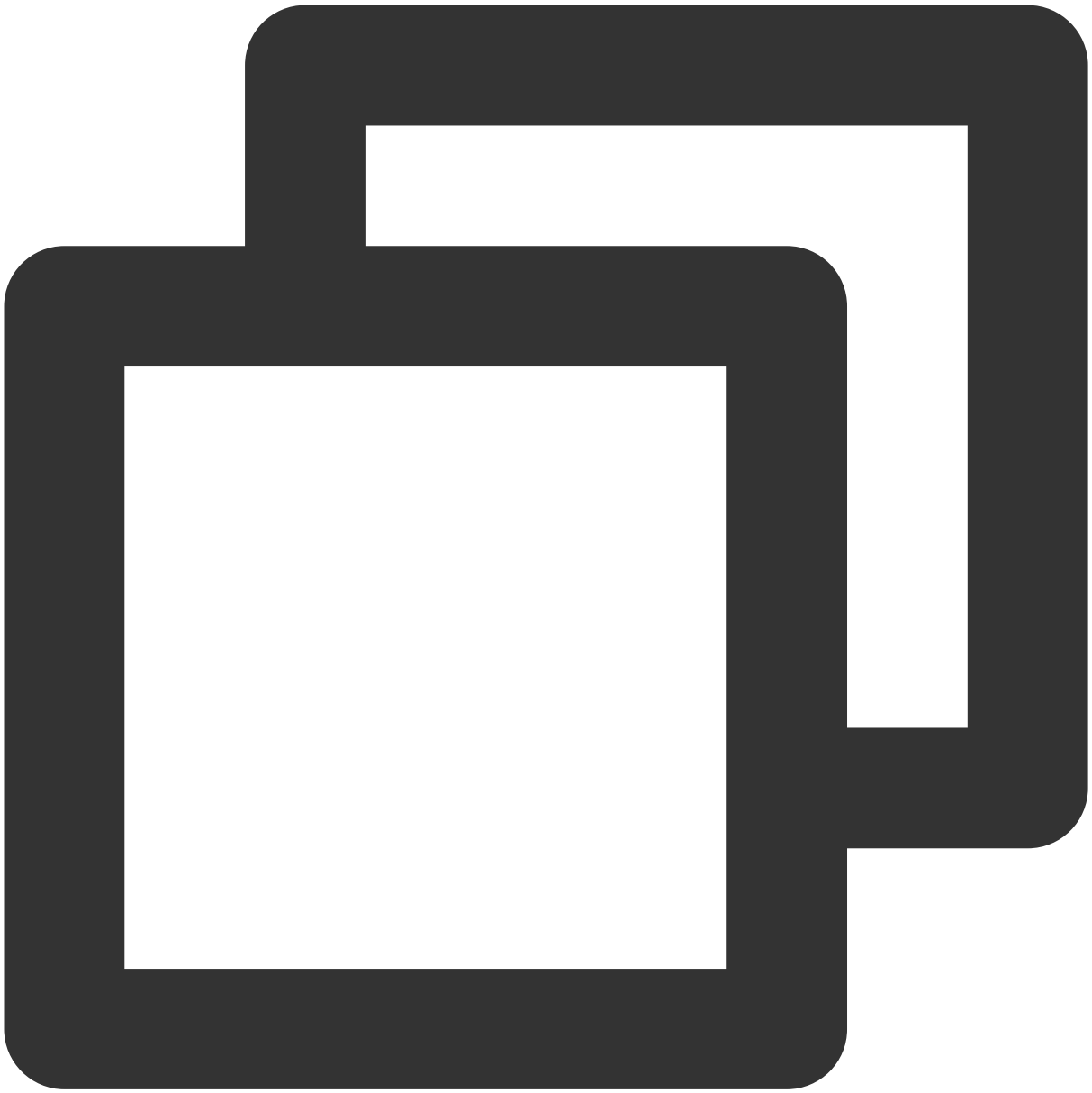
Sample Code

©2013-2022 Tencent Cloud. All rights reserved.



```
wx.getSystemInfo({
  success(res) {
    console.log(res.model)
    console.log(res.pixelRatio)
    console.log(res.windowWidth)
    console.log(res.windowHeight)
    console.log(res.language)
    console.log(res.version)
    console.log(res.platform)
  }
})
```

©2013-2022 Tencent Cloud. All rights reserved.



```
try {
  const res = wx.getSystemInfoSync()
  console.log(res.model)
  console.log(res.pixelRatio)
  console.log(res.windowWidth)
  console.log(res.windowHeight)
  console.log(res.language)
  console.log(res.version)
  console.log(res.platform)
} catch (e) {
```

©2013-2022 Tencent Cloud. All rights reserved.

```
  // Do something when catch error
}
```

getSystemInfoAsync

This API is used via wx.getSystemInfoAsync(Object object).

Feature Description: Asynchronously gets system information. It requires support from a certain host client version. On unsupported clients, a synchronous implementation will be used for the return.

Parameter and Description: Object.

Attribute	Type	Default value	Required	Description
success	function	-	No	Callback Function of Successful Interface Call
fail	function	-	No	Callback Function of Failing Interface Call
complete	function	-	No	Callback function executed upon the completion of the interface invocation (both successful and unsuccessful invocations)

Parameters for object.success callback function: Object res.

Attribute	Type	Description
brand	string	Device Brand
Device Model	string	Device model. Newly launched models will display 'unknown' for a period of time. We will complete adaptation as quickly as possible.
pixelRatio	number	Device Pixel Ratio
screenWidth	number	Screen width, measured in pixels (px)
screenHeight	number	Screen height, measured in pixels (px)
windowWidth	number	Usable window width, measured in pixels (px)
windowHeight	number	Usable window height, measured in pixels (px)
statusBarHeight	number	Task status bar height, measured in pixels (px)
language	string	Language settings within the application.

©2013-2022 Tencent Cloud. All rights reserved.

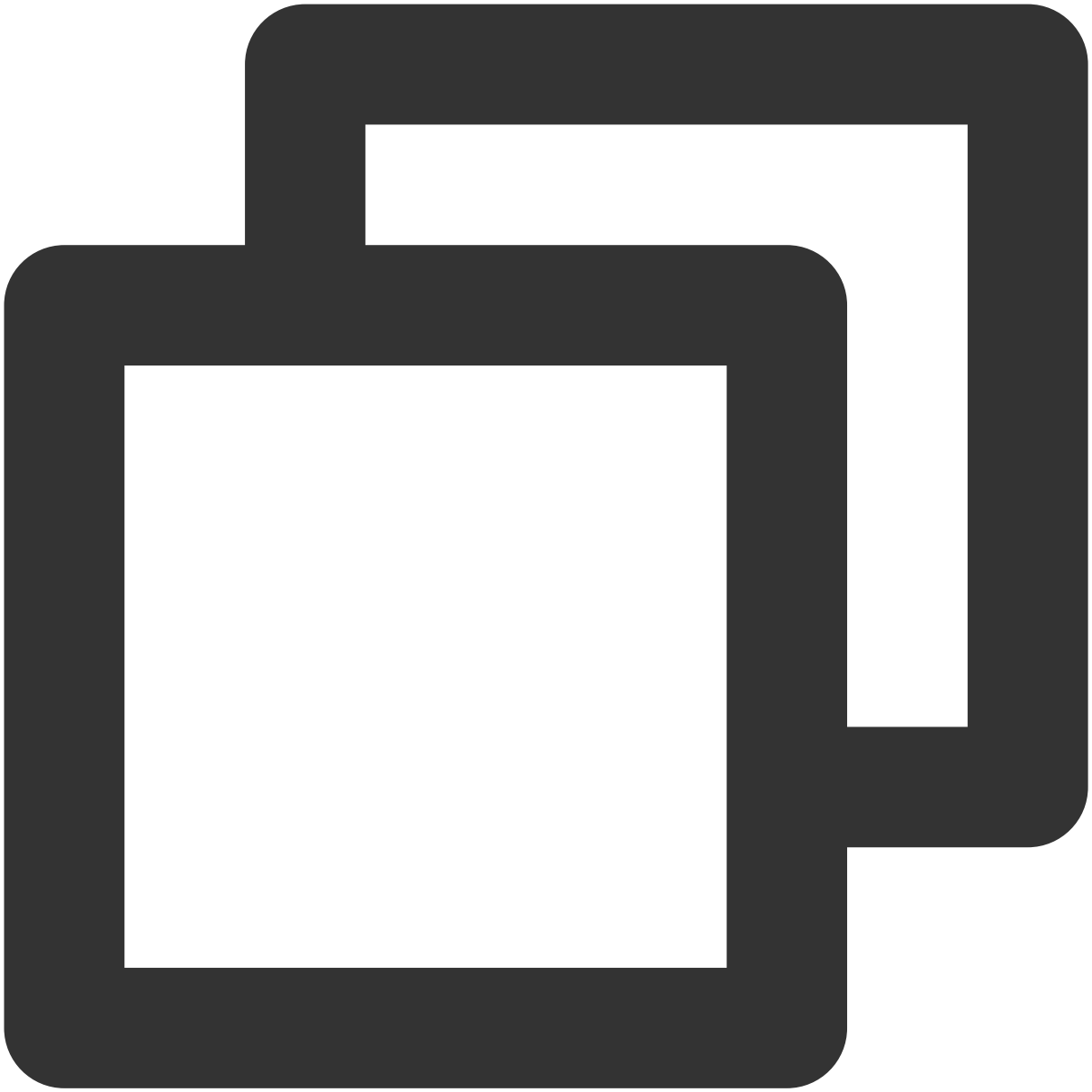
version	string	Application Version No.
system	string	Operating System and Version
platform	string	Client platform, with the following valid values: ios: iOS client (including iPhone, iPad) Android: Android client devtools: IDE Developer Tools
fontSizeSetting	number	User font size, measured in pixels (px).
SDKVersion	string	Client Base Library Version
safeArea	object	Safe area in the upright orientation of the screen. Some models do not have the concept of a safe area and will not return the 'safeArea' field, requiring developers to ensure compatibility. See the table below: **Return Values of 'safeArea'**.
locationReducedAccuracy	boolean	"True" signifies approximate location, while "false" indicates precise location, supported exclusively on iOS.
theme	string	The current system theme can only be obtained when the global configuration 'darkmode' is set to "true". Otherwise, it will be undefined. The valid values are: Light Dark

Return value of safeArea

Structure attributes	Type	Description
left	number	Horizontal coordinate of the top left corner of the safe area
right	number	Horizontal coordinate of the bottom right corner of the safe area
top	number	Vertical coordinate of the top left corner of the safe area
bottom	number	Vertical coordinate of the bottom right corner of the safe area
width	number	Width of the safe area, measured in logical pixels
height	number	Height of the safe area, measured in logical pixels

Sample Code

©2013-2022 Tencent Cloud. All rights reserved.



```
wx.getSystemInfoAsync({
  success(res) {
    console.log(res.model)
    console.log(res.pixelRatio)
    console.log(res.windowWidth)
    console.log(res.windowHeight)
    console.log(res.language)
    console.log(res.version)
    console.log(res.platform)
  }
})
```

©2013-2022 Tencent Cloud. All rights reserved.

getWindowInfo

The usage method of this API is Object wx.getWindowInfo()

Feature Description: Gets window information.

Parameter and Description: Object.

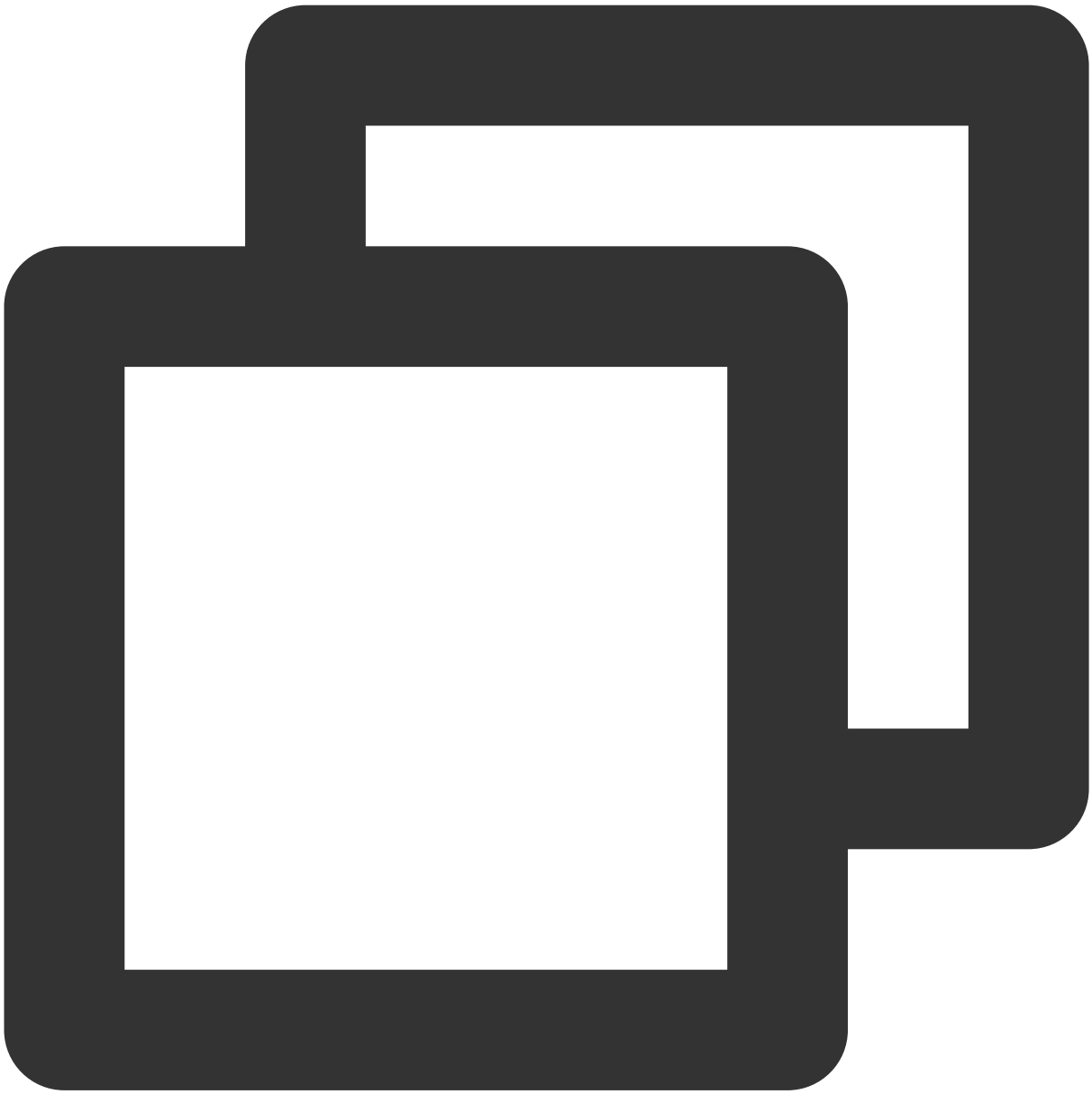
Attribute	Type	Description
pixelRatio	number	Device Pixel Ratio
screenWidth	number	Screen width, measured in pixels (px)
screenHeight	number	Screen height, measured in pixels (px)
windowWidth	number	Usable window width, measured in pixels (px)
windowHeight	number	Usable window height, measured in pixels (px)
statusBarHeight	number	Status bar height, measured in pixels (px)
safeArea	object	Safe area in the upright orientation of the screen. Some models do not have the concept of a safe area and will not return the 'safeArea' field, requiring developers to ensure compatibility.
screenTop	number	The y-value of the window's upper edge

Return value of safeArea

Structure attributes	Type	Description
left	number	Horizontal coordinate of the top left corner of the safe area
right	number	Horizontal coordinate of the bottom right corner of the safe area
top	number	Vertical coordinate of the top left corner of the safe area
bottom	number	Vertical coordinate of the bottom right corner of the safe area
width	number	Width of the safe area, measured in logical pixels
height	number	Height of the safe area, measured in logical pixels

©2013-2022 Tencent Cloud. All rights reserved.

Sample Code



```
const windowInfo = wx.getWindowInfo()

console.log(windowInfo.pixelRatio)
console.log(windowInfo.screenWidth)
console.log(windowInfo.screenHeight)
console.log(windowInfo.windowWidth)
console.log(windowInfo.windowHeight)
console.log(windowInfo.statusBarHeight)
console.log(windowInfo.safeArea)
```

©2013-2022 Tencent Cloud. All rights reserved.

```
console.log(windowInfo.screenTop)
```

getSystemSetting

The usage method of this API is Object wx.getSystemSetting()

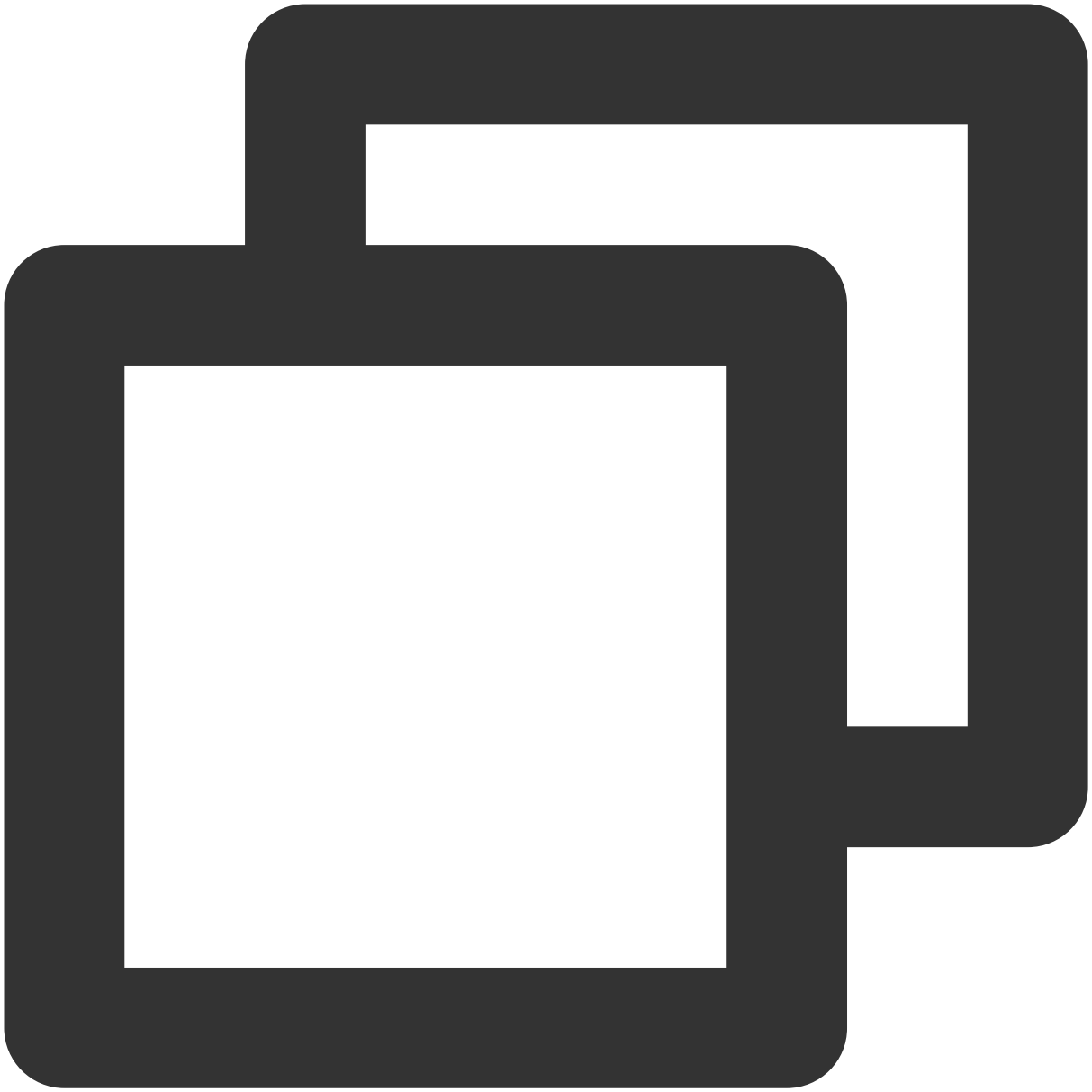
Feature Description: Gets device settings.

Parameter and Description: Object.

Attribute	Type	Description
bluetoothEnabled	boolean	System toggle for Bluetooth functionality.
locationEnabled	boolean	System toggle for geographical location functionality.
wifiEnabled	boolean	System toggle for Wi-Fi functionality.
deviceOrientation	string	Device orientation, whose valid values are: Portrait: Vertical Screen Landscape: Horizontal Screen

Sample Code

©2013-2022 Tencent Cloud. All rights reserved.



```
const systemSetting = wx.getSystemSetting()

console.log(systemSetting.bluetoothEnabled)
console.log(systemSetting.deviceOrientation)
console.log(systemSetting.locationEnabled)
console.log(systemSetting.wifiEnabled)
```

openSystemBluetoothSetting

©2013-2022 Tencent Cloud. All rights reserved.

The usage method of this API is wx.openSystemBluetoothSetting(Object object)

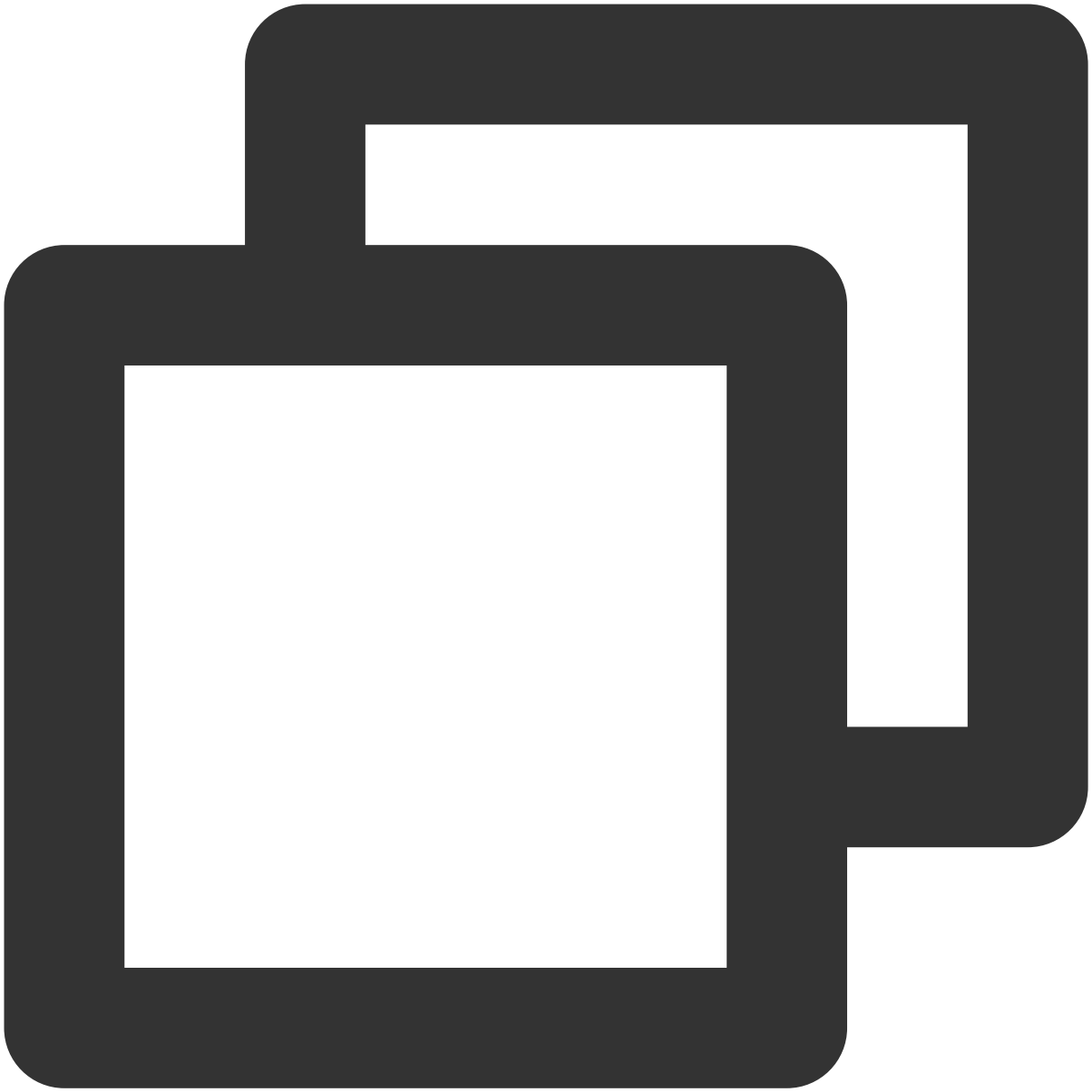
Feature Description: Redirects to the system's Bluetooth settings page. Only supported on Android.

Parameter and Description: Object.

Attribute	Type	Default value	Required	Description
success	function	-	No	Callback Function of Successful Interface Call
fail	function	-	No	Callback Function of Failing Interface Call
complete	function	-	No	Callback function executed upon the completion of the interface invocation (both successful and unsuccessful invocations)

Sample Code

©2013-2022 Tencent Cloud. All rights reserved.



```
wx.openSystemBluetoothSetting({
  success(res) {
    console.log(res)
  }
})
```

openAppAuthorizeSetting

The usage method of this API is wx.openAppAuthorizeSetting(Object object).

©2013-2022 Tencent Cloud. All rights reserved.

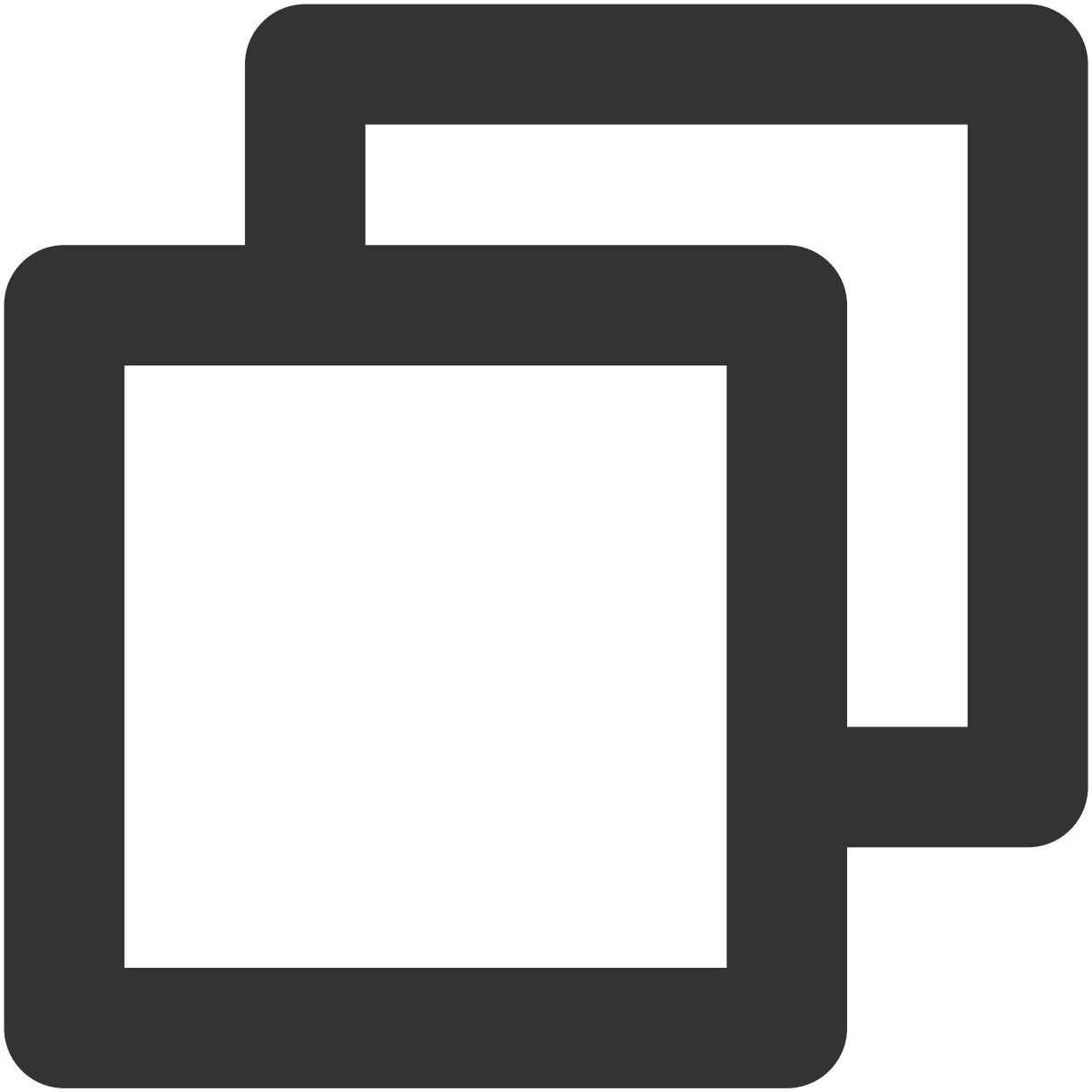
Feature Description: Redirects to the system host client's authorization management page.

Parameter and Description: Object.

Attribute	Type	Default value	Required	Description
success	function	-	No	Callback Function of Successful Interface Call
fail	function	-	No	Callback Function of Failing Interface Call
complete	function	-	No	Callback function executed upon the completion of the interface invocation (both successful and unsuccessful invocations)

Sample Code

©2013-2022 Tencent Cloud. All rights reserved.



```
wx.openAppAuthorizeSetting({
  success(res) {
    console.log(res)
  }
})
```

getRendererUserAgent

The usage method of this API is Promise wx.getRendererUserAgent(Object object).

©2013-2022 Tencent Cloud. All rights reserved.

Feature Description: Gets the UserAgent of the Webview mini program.

Parameter and Description: Object.

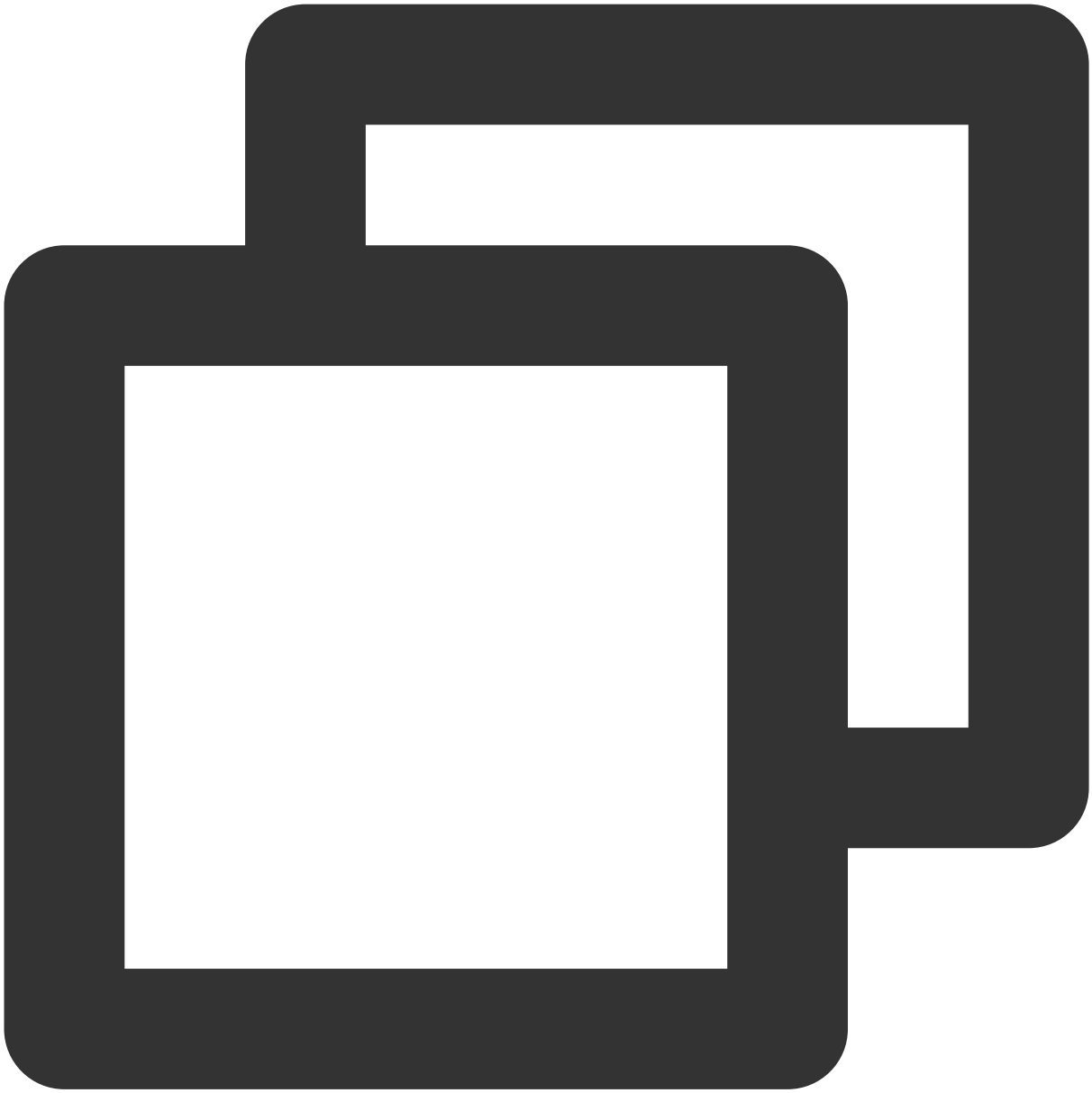
Attribute	Type	Default value	Required	Description
success	function	-	No	Callback Function of Successful Interface Call
fail	function	-	No	Callback Function of Failing Interface Call
complete	function	-	No	Callback function executed upon the completion of the interface invocation (both successful and unsuccessful invocations)

Parameters for object.success callback function: string userAgent, UserAgent.

Return Value: Promise.<string>

Sample Code

©2013-2022 Tencent Cloud. All rights reserved.



```
wx.getRendererUserAgent().then(userAgent => console.log(userAgent))
wx.getRendererUserAgent({
    success(res) {
      console.log(res.userAgent)
  }
})
```

getDeviceInfo

©2013-2022 Tencent Cloud. All rights reserved.

The usage method of this API is Object wx.getDeviceInfo()

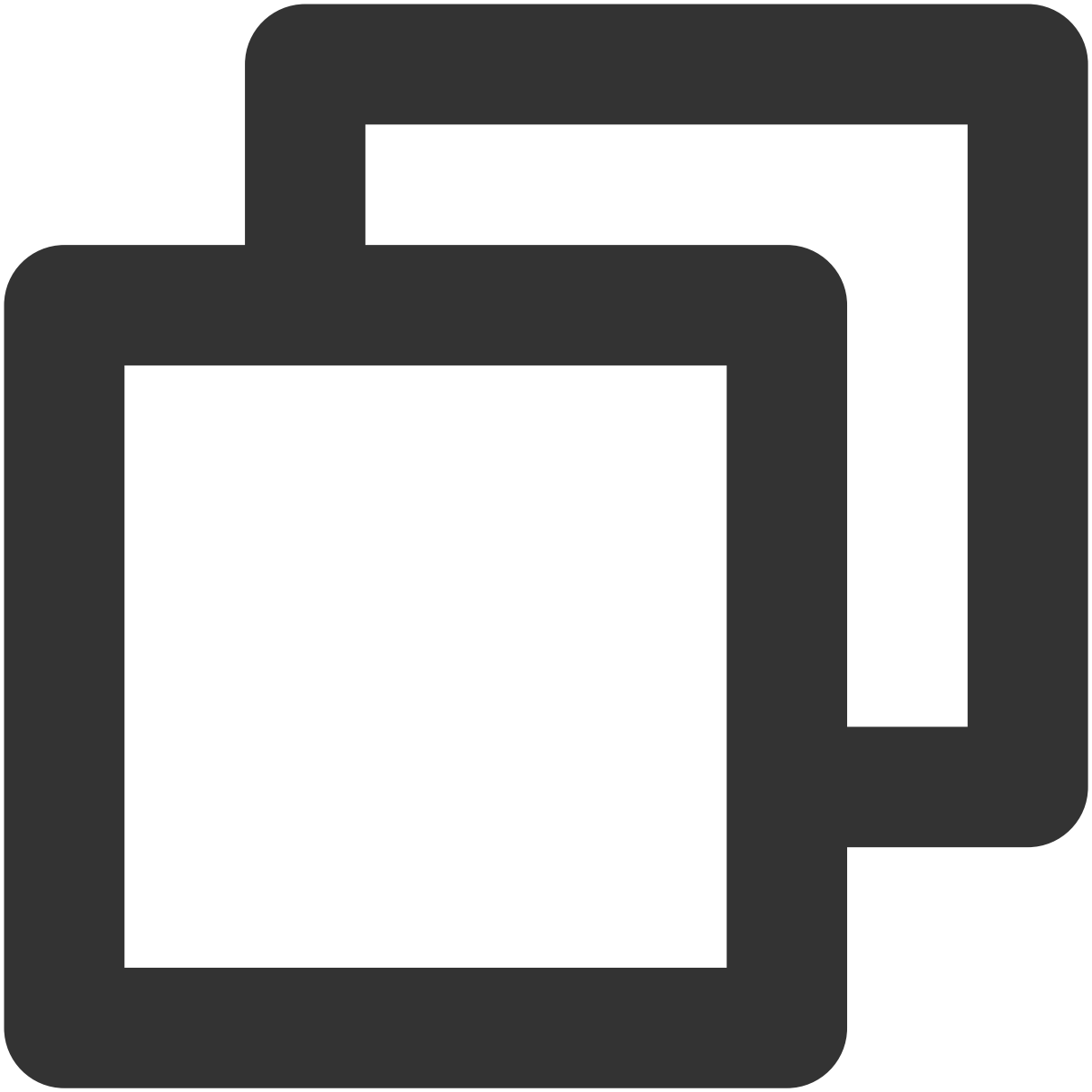
Feature Description: Gets the fundamental information of the device.

Return Value: Object.

Attribute	Type	Description
abi	string	Host App's binary interface type (supported only on Android)
deviceAbi	string	Device's binary interface type (supported only on Android)
benchmarkLevel	number	Device performance level (supported on Android only). Valid values include: -2 or 0: This device is incapable of running mini games. -1: Performance unknown. >=1: Device performance value (currently not exceeding 50). The higher the value, the better the device performance.
brand	string	Device Brand
Device Model	string	Device model. Newly launched models will display 'unknown' for a period of time. We will complete adaptation as quickly as possible.
system	string	Operating System and Version
platform	string	Client platform
cpuType	string	Device CPU model (supported only on Android) **GPU model can be obtained through WebGLRenderingContext.getExtension('WEBGL_debug_renderer_info')**
menorySize	string	Device memory size, measured in MB.

Sample Code

©2013-2022 Tencent Cloud. All rights reserved.



```
const deviceInfo = wx.getDeviceInfo()

console.log(deviceInfo.abi)
console.log(deviceInfo.benchmarkLevel)
console.log(deviceInfo.brand)
console.log(deviceInfo.model)
console.log(deviceInfo.platform)
console.log(deviceInfo.system)
```

getAppBaseInfo

©2013-2022 Tencent Cloud. All rights reserved.

This API is used via Object wx.getAppBaseInfo().

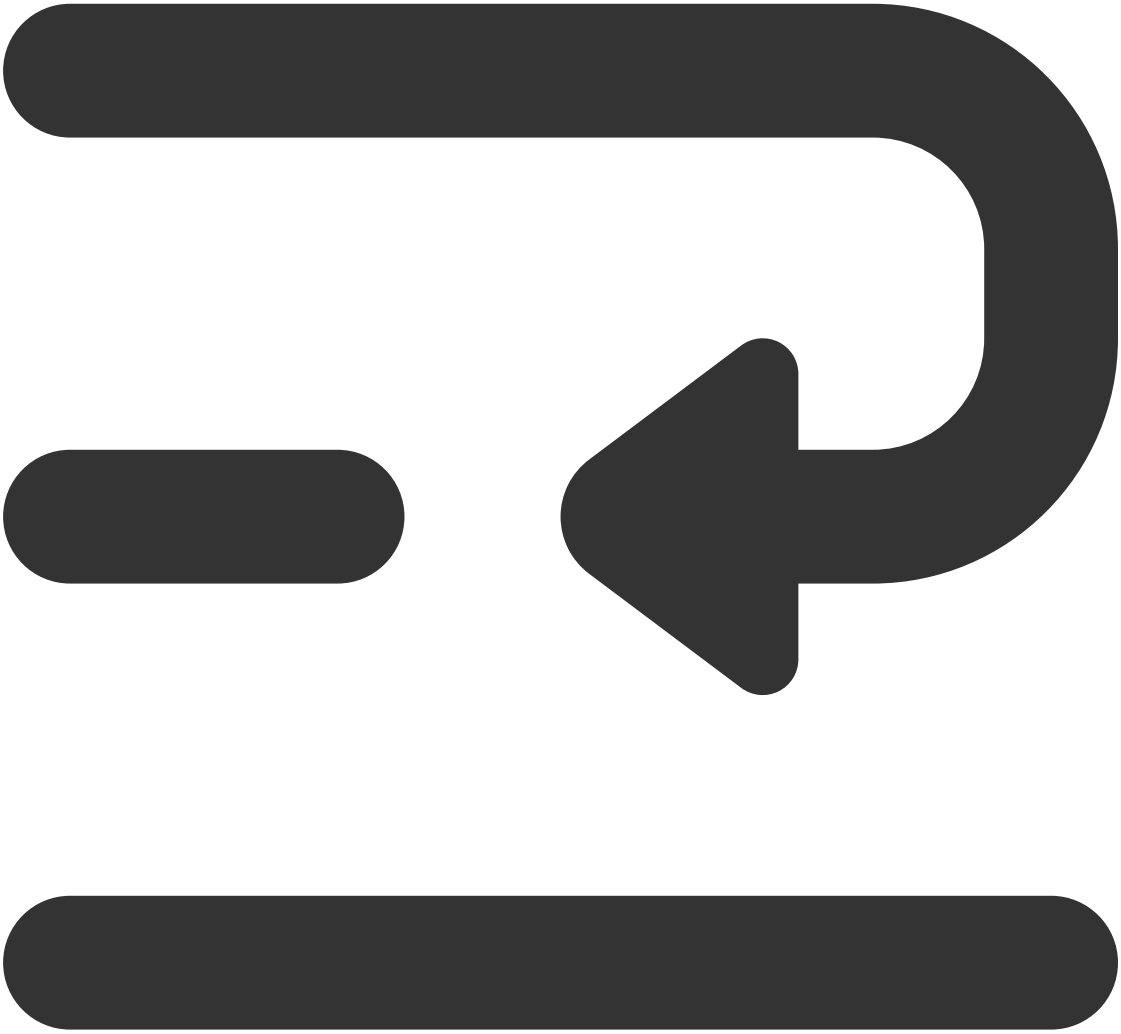
Feature Description: Gets basic information of the host client.

Parameter and Description: Object.

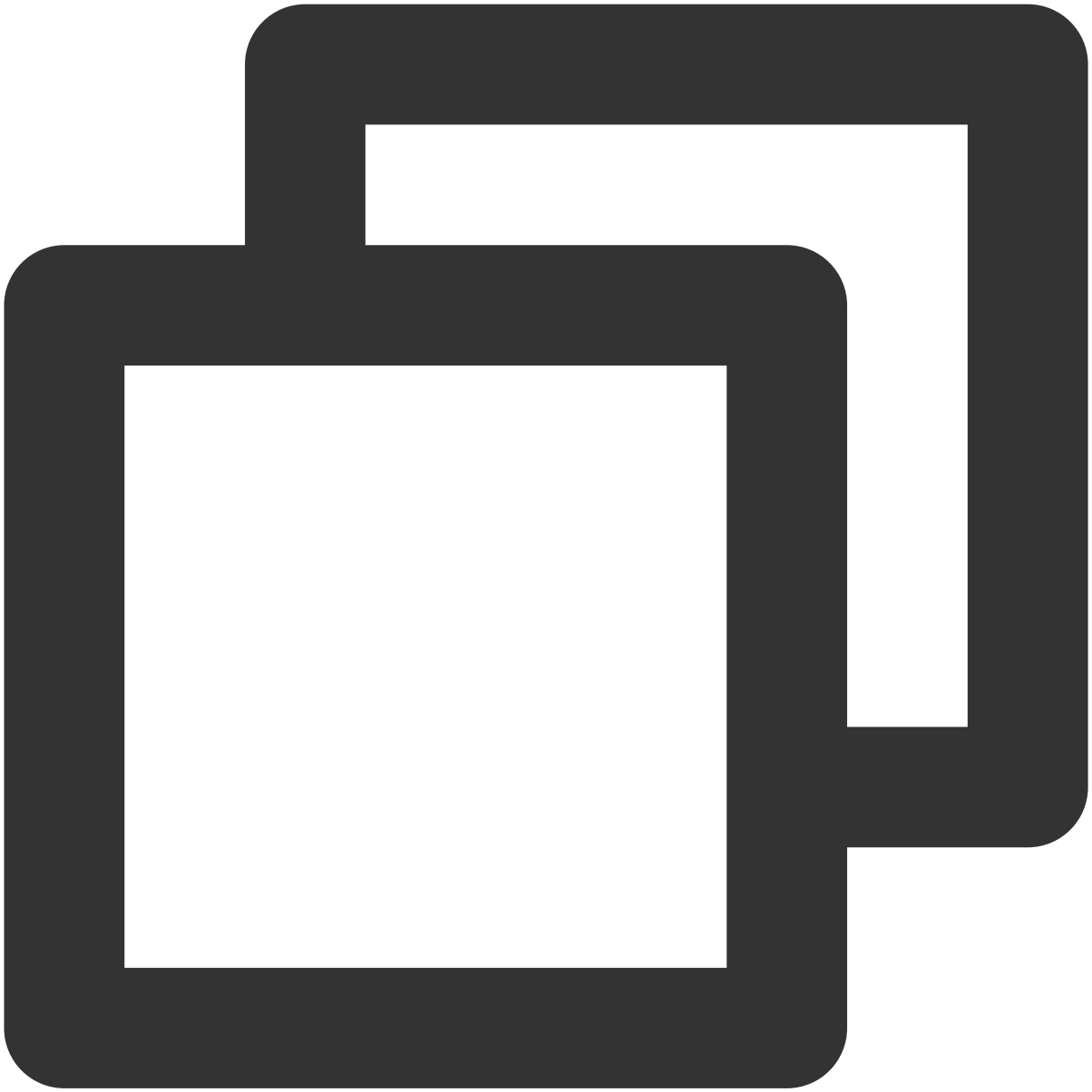
Attribute	Type	Description
SDKVersion	string	Host client's base library version
enableDebug	boolean	Whether debugging has been activated. Debugging can be enabled through the menu in the top right corner or via wx.setEnableDebug.
host	Object	The host environment in which the current mini program is running. The structure attribute is a string type appId, used for the corresponding appId of the host client (third-party App), which is returned only when the mini program is running in a third-party App environment.
language	string	The language set by the host client.
version	string	Host client version number
theme	string	The current system theme has a value of either "light" (bright mode) or "dark" (dark mode). It can only be obtained when the global configuration "darkmode" is set to "true". Otherwise, it is undefined. (Not supported in mini games)

Sample Code

©2013-2022 Tencent Cloud. All rights reserved.



©2013-2022 Tencent Cloud. All rights reserved.



```
const appBaseInfo = wx.getAppBaseInfo()

console.log(appBaseInfo.SDKVersion)
console.log(appBaseInfo.enableDebug)
console.log(appBaseInfo.host)
console.log(appBaseInfo.language)
console.log(appBaseInfo.version)
console.log(appBaseInfo.theme)
```

getAppAuthorizeSetting

©2013-2022 Tencent Cloud. All rights reserved.

The usage method of this API is Object wx.getAppAuthorizeSetting().

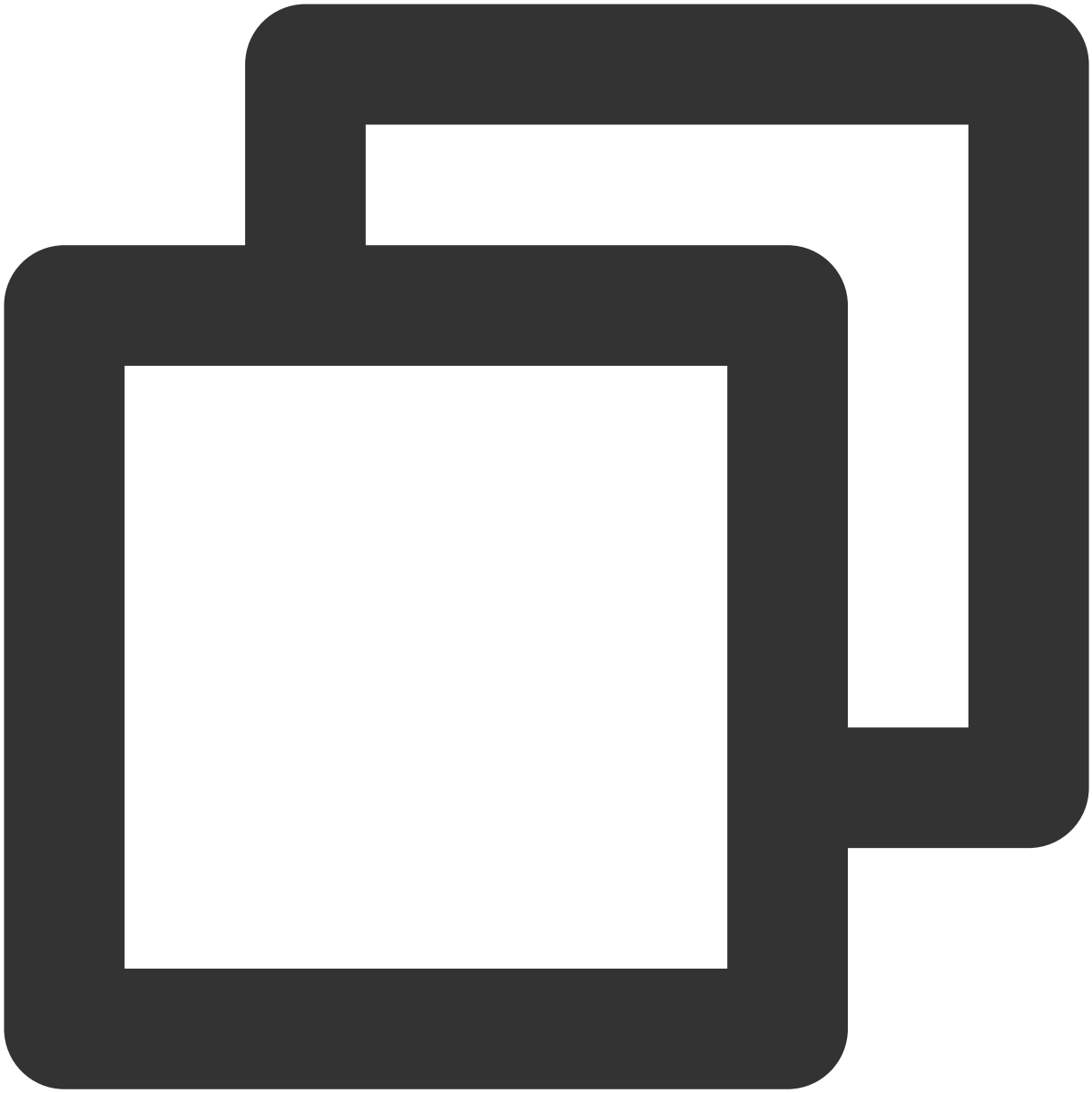
Feature Description: Gets the authorization settings of the host client.

Parameter and Description: Object.

Attribute	Type	Description
albumAuthorized	'authorized'/'denied'/'not determined'	Permission toggle for host client to access the album (effective on iOS only).
bluetoothAuthorized	'authorized'/'denied'/'not determined'	Permission toggle for host client to use the Bluetooth (effective on iOS only).
cameraAuthorized	'authorized'/'denied'/'not determined'	Permission toggle for host client to use the camera.
locationAuthorized	'authorized'/'denied'/'not determined'	Permission toggle for host client to use location services.
locationReducedAccuracy	boolean	"True" signifies approximate location, while "false" indicates precise location, supported exclusively on iOS.
microphoneAuthorized	'authorized'/'denied'/'not determined'	Permission toggle for host client to use the microphone.
notificationAuthorized	'authorized'/'denied'/'not determined'	Permission toggle for host client notifications.
notificationAlertAuthorized	'authorized'/'denied'/'not determined'	Permission host toggle for client notifications to carry a reminder (effective on iOS only).
notificationBadgeAuthorized	'authorized'/'denied'/'not determined'	Permission toggle for host client notifications to carry a marker (effective on iOS only).
notificationSoundAuthorized	'authorized'/'denied'/'not determined'	Permission toggle for host client notifications to have sound (effective on iOS only).
phoneCalendarAuthorized	'authorized'/'denied'/'not determined'	Permission toggle for host client to read and write the calendar.

Sample Code

©2013-2022 Tencent Cloud. All rights reserved.



```
const appAuthorizeSetting = wx.getAppAuthorizeSetting()

console.log(appAuthorizeSetting.albumAuthorized)
console.log(appAuthorizeSetting.bluetoothAuthorized)
console.log(appAuthorizeSetting.cameraAuthorized)
console.log(appAuthorizeSetting.locationAuthorized)
console.log(appAuthorizeSetting.locationReducedAccuracy)
console.log(appAuthorizeSetting.microphoneAuthorized)
console.log(appAuthorizeSetting.notificationAlertAuthorized)
console.log(appAuthorizeSetting.notificationAuthorized)
console.log(appAuthorizeSetting.notificationBadgeAuthorized)
```

©2013-2022 Tencent Cloud. All rights reserved.

```
console.log(appAuthorizeSetting.notificationSoundAuthorized)
console.log(appAuthorizeSetting.phoneCalendarAuthorized)
```

Update

getUpdateManager

This API is used via UpdateManager wx.getUpdateManager.

Feature Description: Gets the globally unique version update manager, used for managing mini program updates, with the usage method being UpdateManager wx.getUpdateManager.
Return Value: UpdateManager, the update manager object.

UpdateManager

.applyUpdate

This method is used via UpdateManager.applyUpdate().

Feature Description: Forces the mini program to restart and use the new version. This is invoked after the new version of the mini program has been downloaded (i.e., upon receipt of the `onUpdateReady` callback).

.onCheckForUpdate

This method is used via UpdateManager.onCheckForUpdate(function listener).

Feature Description: Monitors events of requesting the backend to check for update results. The mini program automatically checks for updates during a cold start, eliminating the need for the developer to manually trigger it.
Parameter and Description: function callback, the callback function for the event of requesting the backend to check for update results.

Attribute	Type	Description
hasUpdate	Boolean	Is there a new version?

©2013-2022 Tencent Cloud. All rights reserved.

.onUpdateFailed

This method is used via UpdateManager.onUpdateFailed(function listener).

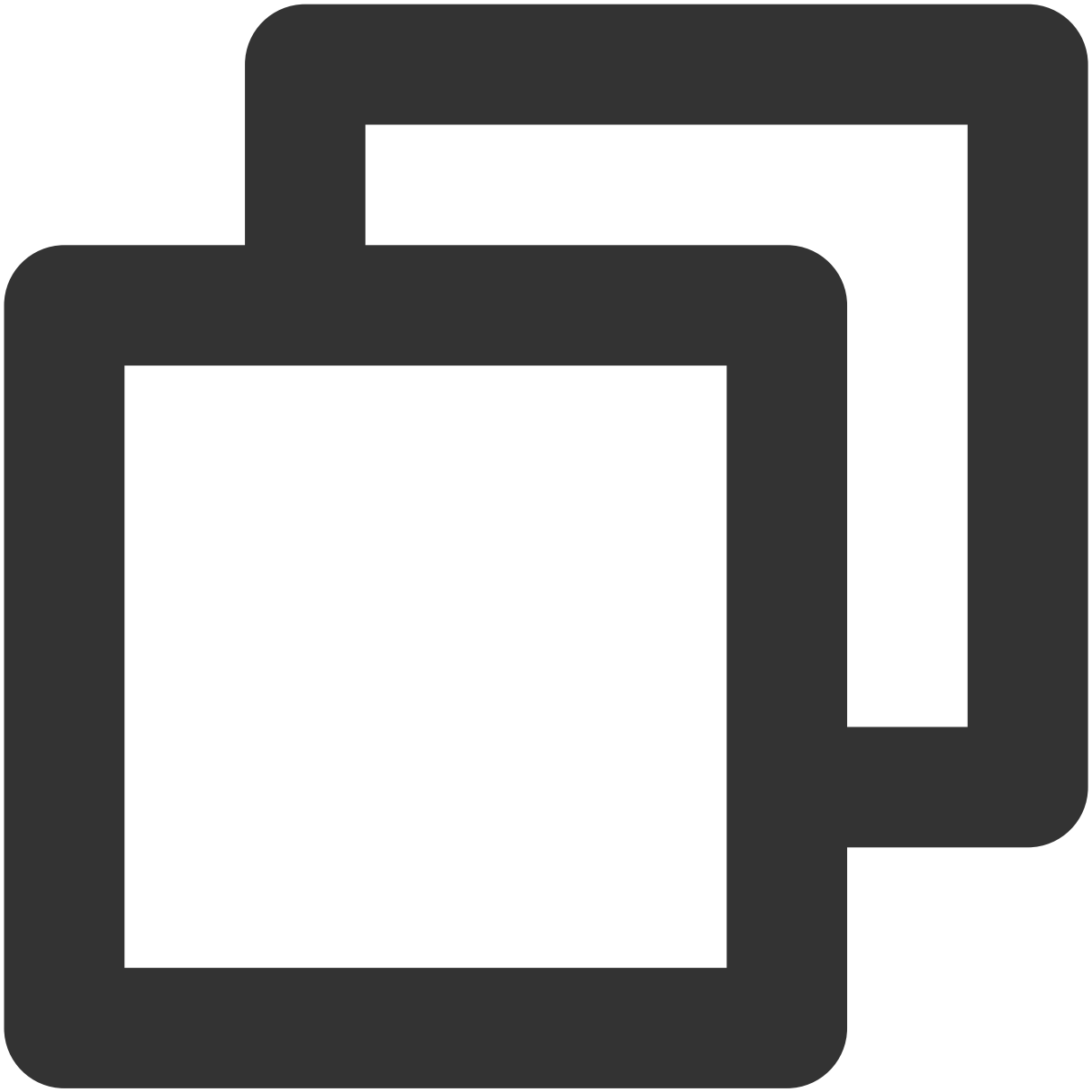
Feature Description: Monitors events of a version update for the mini program. The client initiates the download (no developer trigger required), and a callback is executed upon successful download.
Parameter and Description: function callback, the callback function for the event of a mini program update failure.

.onUpdateReady

This method is used via UpdateManager.onUpdateReady(function listener).

Feature Description: Monitors events of a mini program update failure. When a new version of the mini program is available, the client initiates the download (no developer trigger required), and a callback is executed if the download fails (possibly due to network issues, etc.).
Parameter and Description: function callback, the callback function for the event of a new version update for the mini program.
Sample Code

©2013-2022 Tencent Cloud. All rights reserved.



```
const updateManager = wx.getUpdateManager()

updateManager.onCheckForUpdate(function (res) {
  // Callback after the new version information request is completed
  console.log(res.hasUpdate)
})

updateManager.onUpdateReady(function () {
  wx.showModal({
    title: 'Update Prompt',
    content: 'The new version is ready. Would you like to restart the application?'
```

©2013-2022 Tencent Cloud. All rights reserved.

```
    success(res) {
      if (res.confirm) {
        // The new version has been downloaded. Call applyUpdate to apply the new v
        updateManager.applyUpdate()
      }
    }
  })
})

updateManager.onUpdateFailed(function () {
  // New version download failed.
})
```

Mini Program

Lifecycle

getLaunchOptionsSync

The usage method of this API is Object wx.getLaunchOptionsSync().

Note:

Some versions will return undefined when there is no referrerInfo, it is recommended to use options.referrerInfo && options.referrerInfo.appId to judge.

Feature Description: Gets the parameters at the launch of the mini program.

Return Value: Object, launch parameters.

Attribute	Type	Description
path	String	The path to launch the mini program.
scene	Number	Scenario value for launching the mini program, default is 1001
query	Object	The query parameters for launching the Mini Program.
referrerInfo	Object	Source information, returned when entering the mini program from another mini program or App. Otherwise returns {}
forwardMaterials	Array.<Object>	An array of information about the opened file, which is only carried if it is opened from a chat clip scene

©2013-2022 Tencent Cloud. All rights reserved.

The structure of referrerInfo.

Attribute	Type	Description
appId	String	The appId of the source mini program or app.
extraData	Object	Data passed by the source mini program

The structure of forwardMaterials.

Attribute	Type	Description
type	string	The mimetype type of the file.
name	Object	File Name
path	string	File path
size	number	File size

Scenes where valid `referrerInfo` is returned.

Attribute	Type	Significance of appId
1036	App message sharing card	Originating from App
1037	Open a mini program from another mini program	Originating from mini program
1038	Returning from another mini program	Originating from mini program

getEnterOptionsSync

This API is used via Object wx.getEnterOptionsSync().

Note:

Some versions will return undefined when there is no referrerInfo, it is recommended to use options.referrerInfo && options.referrerInfo.appId to judge.

Feature Description: Gets the parameters at the time of this mini program's initiation. If it is a cold start, the return value is consistent with the callback parameters of App.onLaunch; if it is a hot start, the return value is consistent with App.onShow.

Return Value: Object

©2013-2022 Tencent Cloud. All rights reserved.

Attribute	Type	Description
path	string	Path to launch the mini program (code package path)
scene	number	Scenario values for launching a mini program
query	Object	The query parameters for launching the Mini Program.
shareTicket	string	shareTicket
referrInfo	Object	Source information. The value is returned when entering the mini program from another mini program, We Chat Official Account, or App. Otherwise, it returns {}. For specifics, see the notes preceding the table.
forwardMaterials	Array.<Object>	Array of opened file information, a parameter that is only carried when opened from the chat material scene (scene is 1173).
chatType	number	When opening the mini program from a group chat or private chat in the host client, chatType indicates the specific type of group chat or private chat within the host client. The valid values are: 1: Single chat with host client contacts 2: Single chat with corporate host client contacts 3: Common host client group chat 4: Intercommunication group chat in corporate host client

The structure of referrerInfo.

Attribute	Type	Description
appId	String	The appId of the source mini program or app.
extraData	Object	Data passed by the source mini program

The structure of forwardMaterials.

Attribute	Type	Description
type	String	The mimetype type of the file.
name	Object	File Name
path	String	File path (or URL if it's a webview)
size	Number	File size

Scenarios returning valid referrerInfo

©2013-2022 Tencent Cloud. All rights reserved.

Attribute	Type	Significance of appId
1036	App message sharing card	Originating from App
1037	Open a mini program from another mini program	Originating from mini program
1038	Returning from another mini program	Originating from mini program

Application-level events

onError

The usage method of this API is wx.onError(function callback)

Feature Description: Monitors errors within the mini program, such as script errors or API call failures.
Parameter and Description: function callback, the callback function for mini program error events.
Return Parameter: string error, error information, including the stack.

offError

The usage method of this API is wx.offError(function callback)

Feature Description: Cancels the monitoring of mini program error events.
Parameter and Description: function callback, the callback function for mini program error events.

onThemeChange

The usage method of this API is wx.onThemeChange(function listener)

Note:
This event will only be triggered when the global configuration darkmode is set to "true".
Feature Description: Monitors system theme change events.
Parameter and Description: function listener, the monitoring function for system theme change events.

©2013-2022 Tencent Cloud. All rights reserved.

Return Parameter: Object res

Attribute	Valid Values and Descriptions	Type	Description
theme	Dark: Dark Theme Light: Light Theme	String	The current theme of the system, either light or dark.

offThemeChange

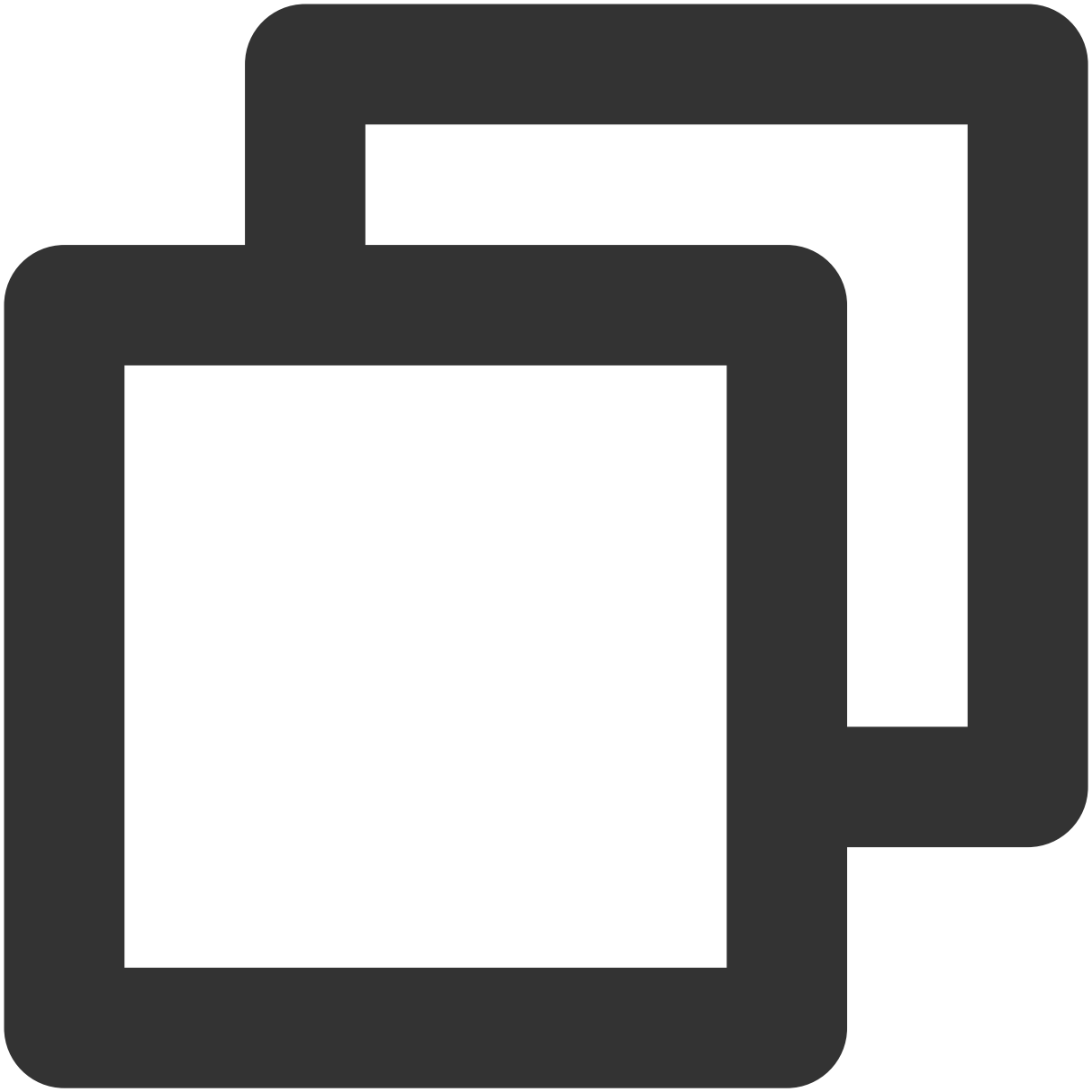
> The usage method of this API is wx.offThemeChange(function listener).

Function Description: Removes the listener function for system theme change events.

Parameter and Description: Function listener, the listener function passed in by onThemeChange. If this parameter is not provided, all listener functions will be removed.

Sample Code

©2013-2022 Tencent Cloud. All rights reserved.



```
const listener = function (res) { console.log(res) }

wx.onThemeChange(listener)
wx.offThemeChange(listener) // The same function object as the listener must be pas
```

onPageNotFound

The usage method of this API is wx.onPageNotFound(function listener).

©2013-2022 Tencent Cloud. All rights reserved.

Note:

Developers can redirect the page within the callback, but it must be handled synchronously within the callback. Asynchronous handling (such as asynchronous execution with setTimeout) is invalid.

If the developer does not call wx.onPageNotFound to bind the listener, and does not declare App.onPageNotFound, when the page to be redirected does not exist, it will push into the host client's native page-not-found prompt page.

If the callback redirects to another non-existent page, it will push into the host client's native page-not-found prompt page, without callback for a second time.

Feature Description: Monitors events where the page that the mini program wants to open does not exist. The timing of this event is consistent with the callback of App.onPageNotFound.

Parameter and Description: Object res parameter, function listener, which is the listener function for events where the page that the mini program wants to open does not exist.

Attribute	Type	Description
path	string	Path of the non-existent page (code package path)
query	Object	Query parameters for opening a non-existent page
isEntryPage	boolean	Whether it is the first page of this launch (for example, when entering from shares or other entrances, the first page is the share page configured by the developer)

offPageNotFound

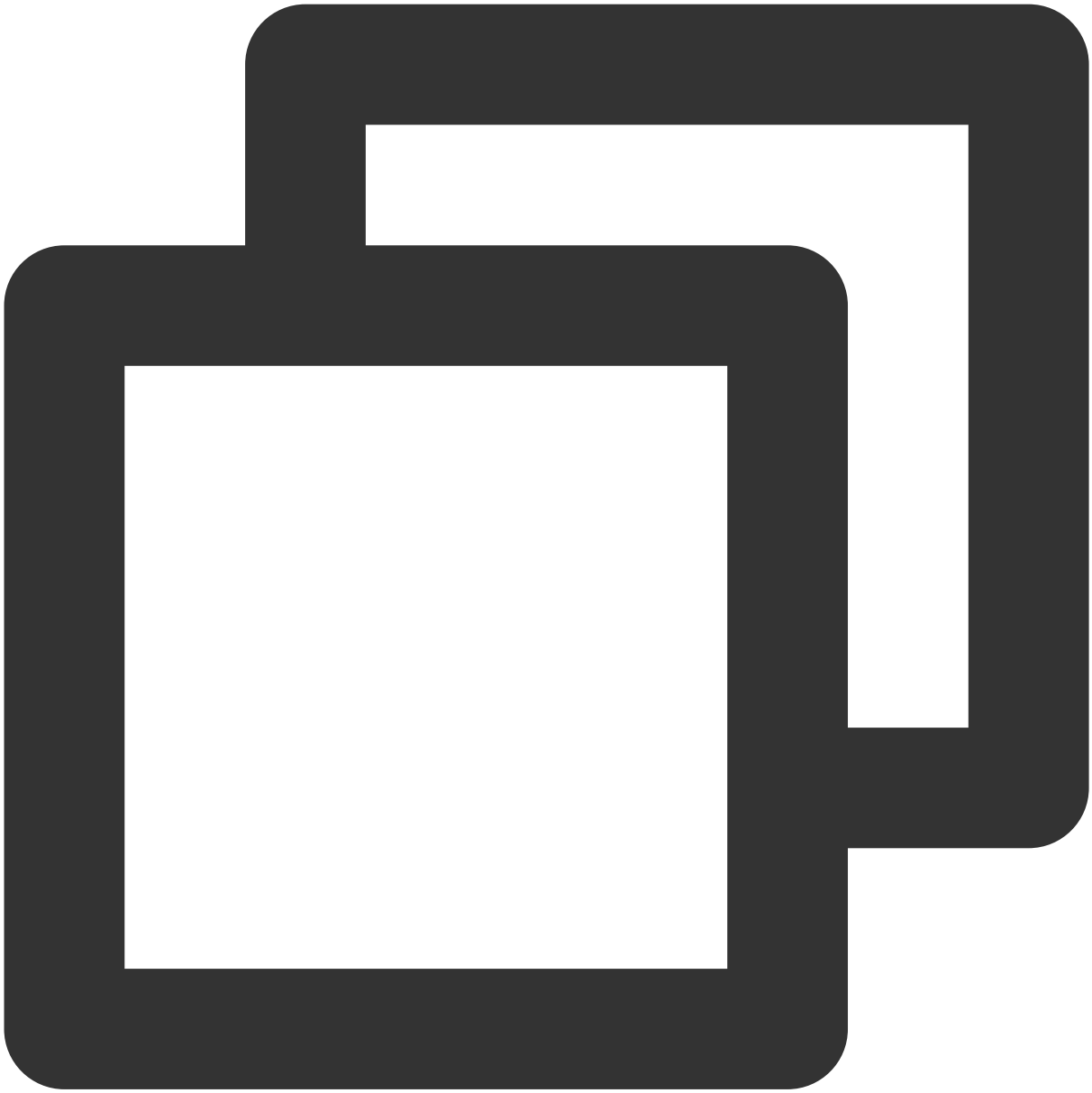
> The usage method of this API is wx.offPageNotFound(function listener).

Feature Description: Removes the listener function for events where the page that the mini program wants to open does not exist.

Parameter and Description: function listener, the listener function passed in by onPageNotFound. If this parameter is not passed, all listener functions will be removed.

Sample Code

©2013-2022 Tencent Cloud. All rights reserved.



```
const listener = function (res) { console.log(res) }

wx.onPageNotFound(listener)
wx.offPageNotFound(listener) // The same function object as the listener must be pa
```

onAppShow

The usage method of this API is wx.onAppShow(function listener)

©2013-2022 Tencent Cloud. All rights reserved.

Note:

Some versions will return undefined if there is no referrerInfo, it is recommended to use options.referrerInfo && options.referrerInfo.appId to judge.

Feature Description: Monitors events of mini program switching to the frontend. The callback parameters of these events are consistent with those of App.onShow.

Parameter and Description: Object res parameter, function listener, the listener function for events of mini program switching to the frontend.

Attribute	Type	Description
path	string	Path to launch the mini program (code package path)
scene	number	Scenario values for launching a mini program
query	Object	The query parameters for launching the Mini Program.
shareTicket	string	shareTicket
referrerInfo	Object	Source information. Returned when the mini programme is accessed from another mini program or an App. Otherwise returns {}, see the note in the previous section of the form.
forwardMaterials	Array.<Object>	An array of information about the opened file, which is only carried if it is opened from a chat clip scene
chatType	number	When opening the mini program from a group chat or private chat in the host client, chatType indicates the specific type of group chat or private chat within the host client. The valid values are: 1: Single chat contacts within the host client 2: Single chat with corporate contacts within the host client 3: Common group chat within the host client 4: Inter-company group chat within the host client

The structure of referrerInfo.

Attribute	Type	Description
appId	string	The appId of the source mini program or app.
extraData	Object	Data passed by the source mini program

The structure of forwardMaterials.

Attribute	Type	Description
type	String	The mimetype type of the file.

©2013-2022 Tencent Cloud. All rights reserved.

name	Object	File Name
path	String	File path (or URL if it's a webview)
size	Number	File size

Scenarios returning valid referrerInfo

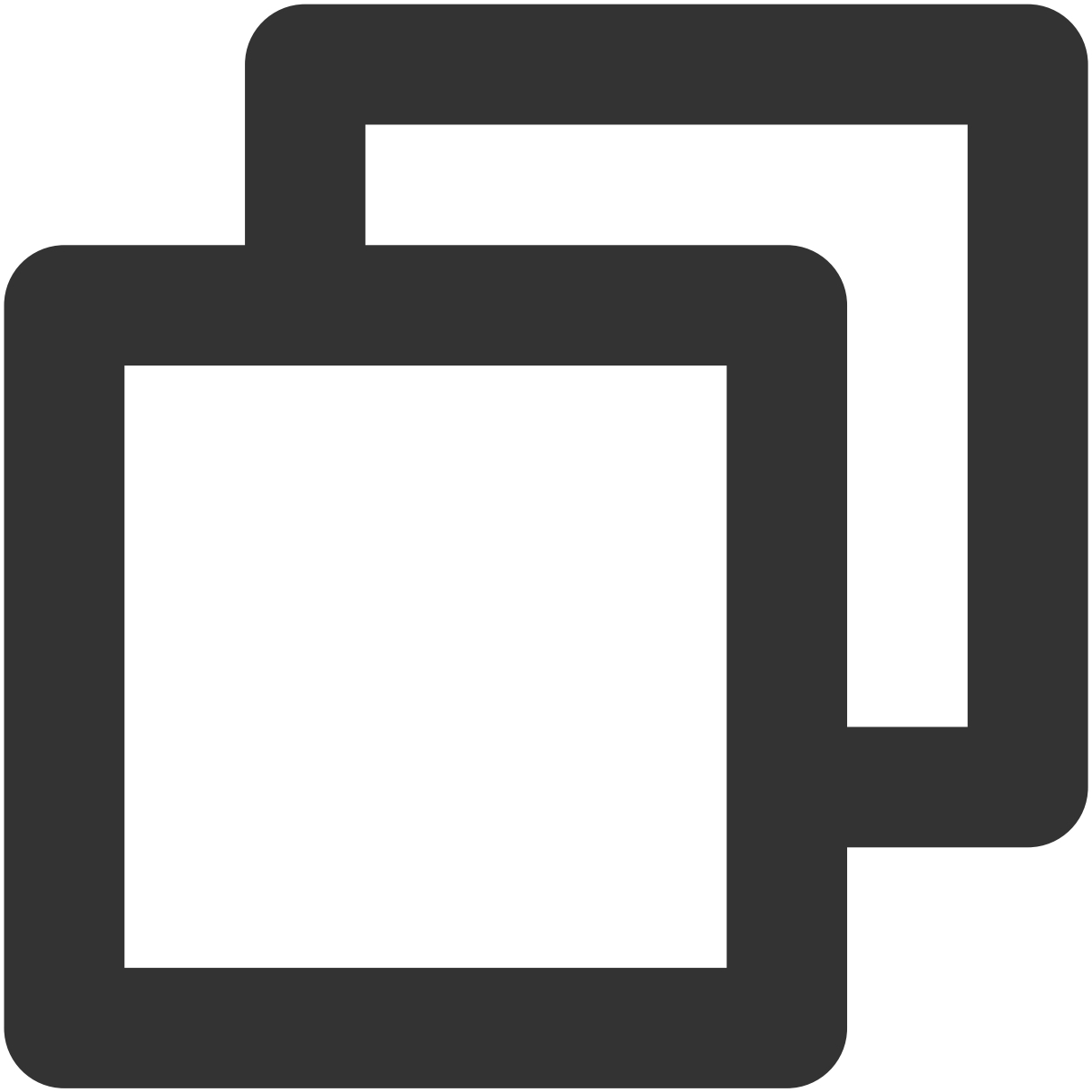
Scene Value	Scene	Significance of appId
1036	App message sharing card	Originating from App
1037	Open a mini program from another mini program	Originating from mini program
1038	Returning from another mini program	Originating from mini program

offAppShow

The usage method of this API is wx.offAppShow(function listener).

Feature Description: Removes the liThestener function for events of mini program switch to frontend.
Parameter and Description: function listener, the listener function passed in by onAppShow. If this parameter is not passed, all listener functions will be removed.
Sample Code

©2013-2022 Tencent Cloud. All rights reserved.



```
const listener = function (res) { console.log(res) }

wx.onAppShow(listener)
wx.offAppShow(listener) // The same function object as the listener must be passed
```

onAppHide

The usage method of this API is wx.onAppHide(function listener).

©2013-2022 Tencent Cloud. All rights reserved.

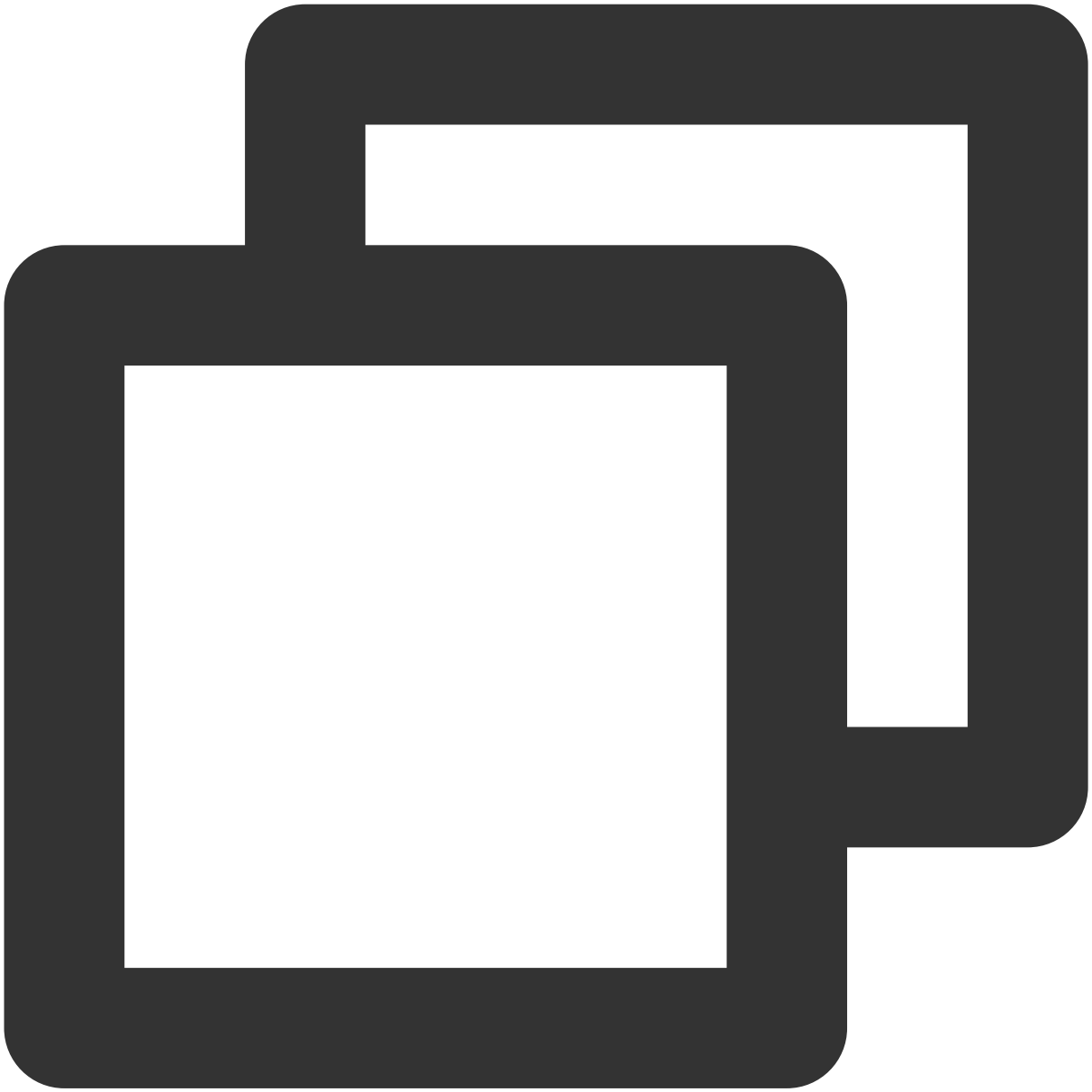
Feature Description: Monitors events of mini program switching to the backend. The timing of this event is consistent with the callback of App.onHide.
Parameter and Description: function listener, the listener function for events of mini program switching to the backend.

offAppHide

> The usage method of this API is wx.offAppHide(function listener)

Feature Description: Removes the listener function for events of mini program switch to frontend.
Parameter and Description: function listener, the listener function passed in by onAppHide. If this parameter is not passed, all listener functions will be removed.

©2013-2022 Tencent Cloud. All rights reserved.



```
const listener = function (res) { console.log(res) }

wx.onAppHide(listener)
wx.offAppHide(listener) // The same function object as the listener must be passed
```

onUnhandledRejection

The usage method of this API is wx.onUnhandledRejection(function listener).

©2013-2022 Tencent Cloud. All rights reserved.

Note:

All unhandledRejection events can be captured by this listener, but only those of the Error type will trigger an alarm in the mini program backend.

Feature Description: Monitors unhandled Promise rejection events.

Parameter and Description: Object res parameter, function listener, the listener function for unhandled Promise rejection events.

Attribute	Type	Description
reason	string	Rejection reason, typically an Error object.
promise	Promise.<any>	The rejected Promise object.

offUnhandledRejection

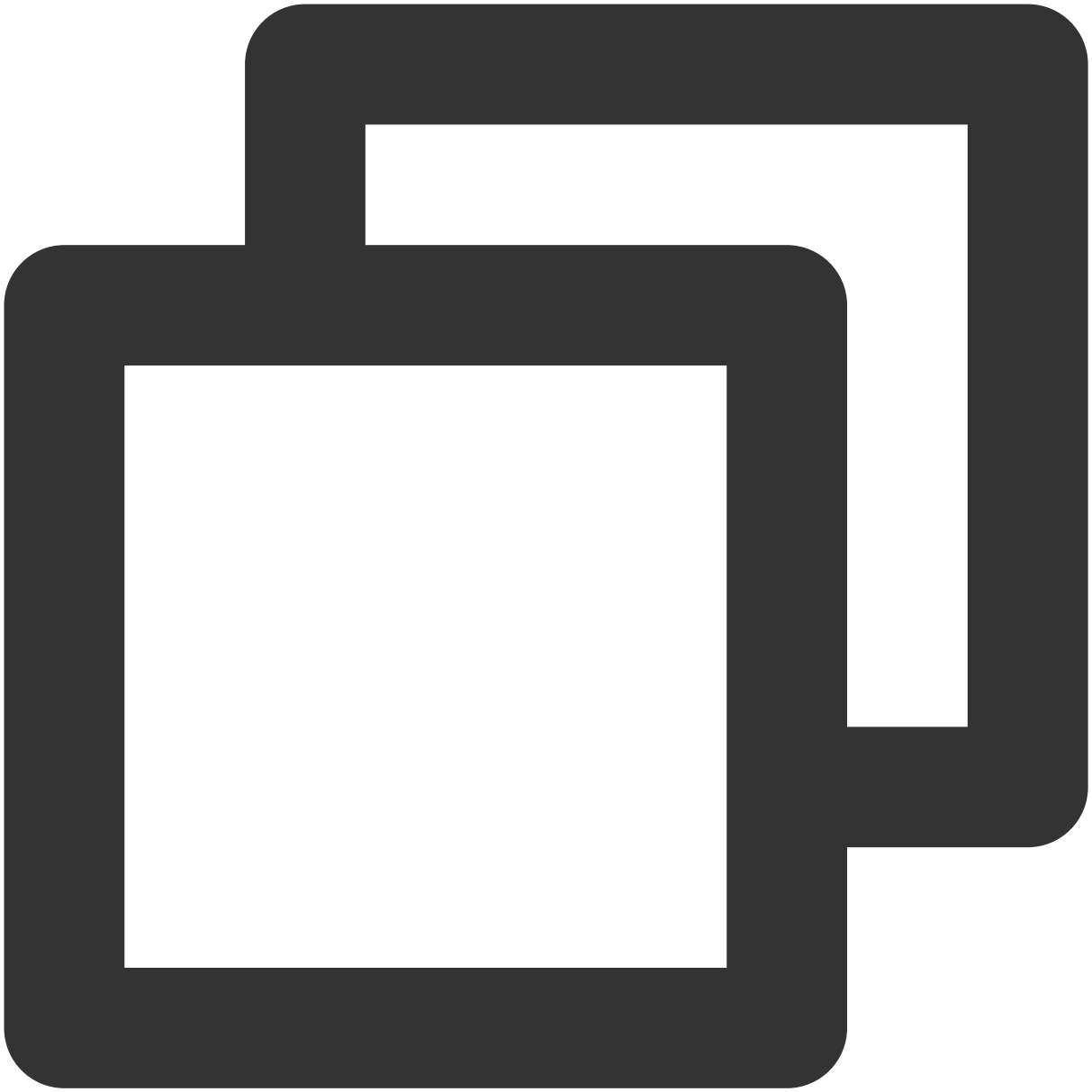
> The usage method of this API is wx.offUnhandledRejection(function listener).

Feature Description: Removes the listener function for unhandled Promise rejection events.

Parameter and Description: Function listener, the listener function passed in by onUnhandledRejection. If this parameter is not provided, all listener functions will be removed.

Sample Code

©2013-2022 Tencent Cloud. All rights reserved.



```
const listener = function (res) { console.log(res) }

wx.onUnhandledRejection(listener)
wx.offUnhandledRejection(listener) // The same function object as the listener must
```

Debug

setEnableDebug

©2013-2022 Tencent Cloud. All rights reserved.

The usage method of this API is wx.setEnableDebug(Object object).

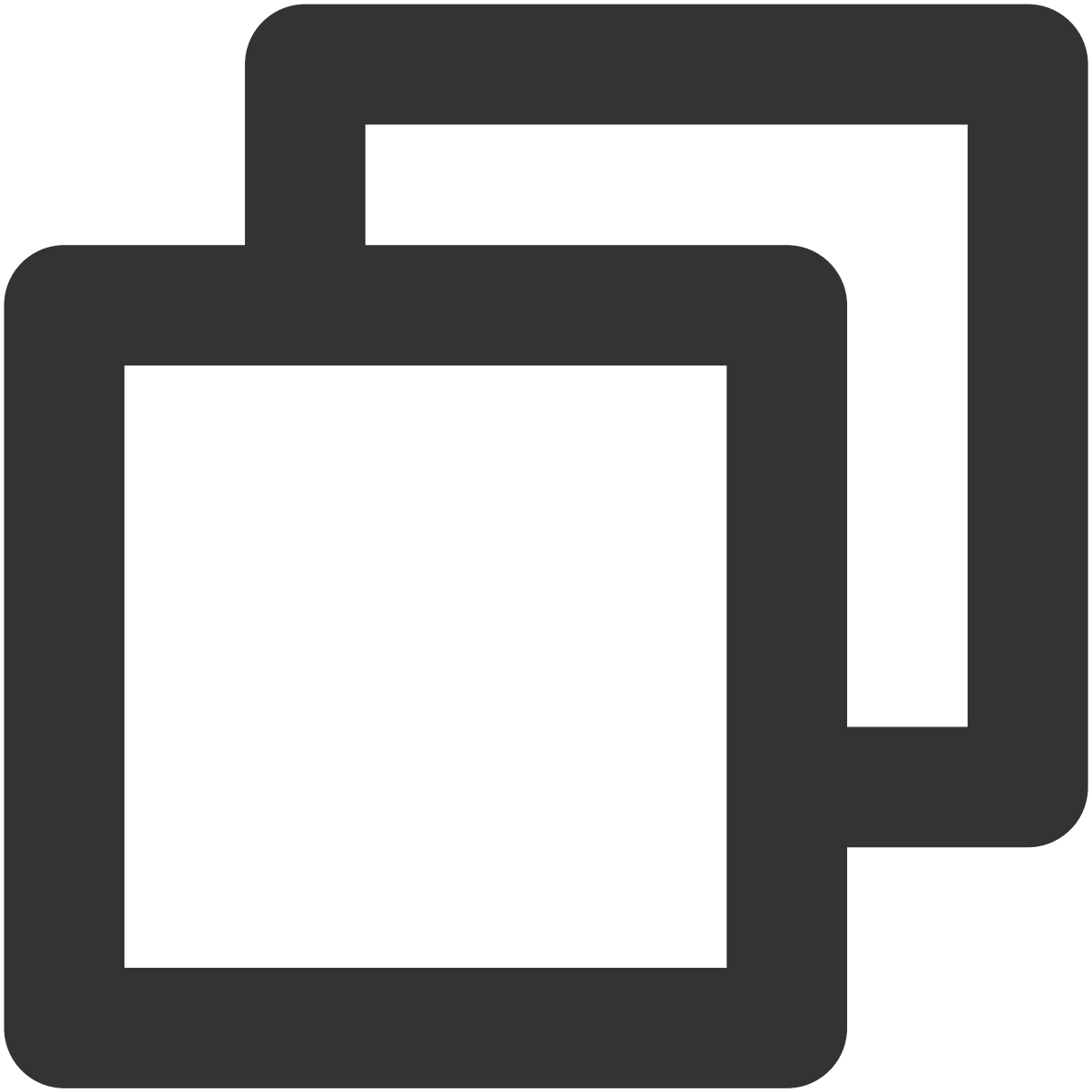
Feature Description: Sets whether to enable the debugging switch. This switch is also effective for the official version.

Parameter and Description: Object.

Attribute	Type	Default value	Required	Description
enableDebug	boolean	-	Yes	Whether to enable debugging
success	function	-	No	Callback Function of Successful Interface Call
fail	function	-	No	Callback Function of Failing Interface Call
complete	function	-	No	Callback function executed upon the completion of the interface invocation (both successful and unsuccessful invocations)

Sample Code

©2013-2022 Tencent Cloud. All rights reserved.



```
// Enable debugging
wx.setEnableDebug({
  enableDebug: true
})

// Disable debugging
wx.setEnableDebug({
  enableDebug: false
})
```

Note:

©2013-2022 Tencent Cloud. All rights reserved.

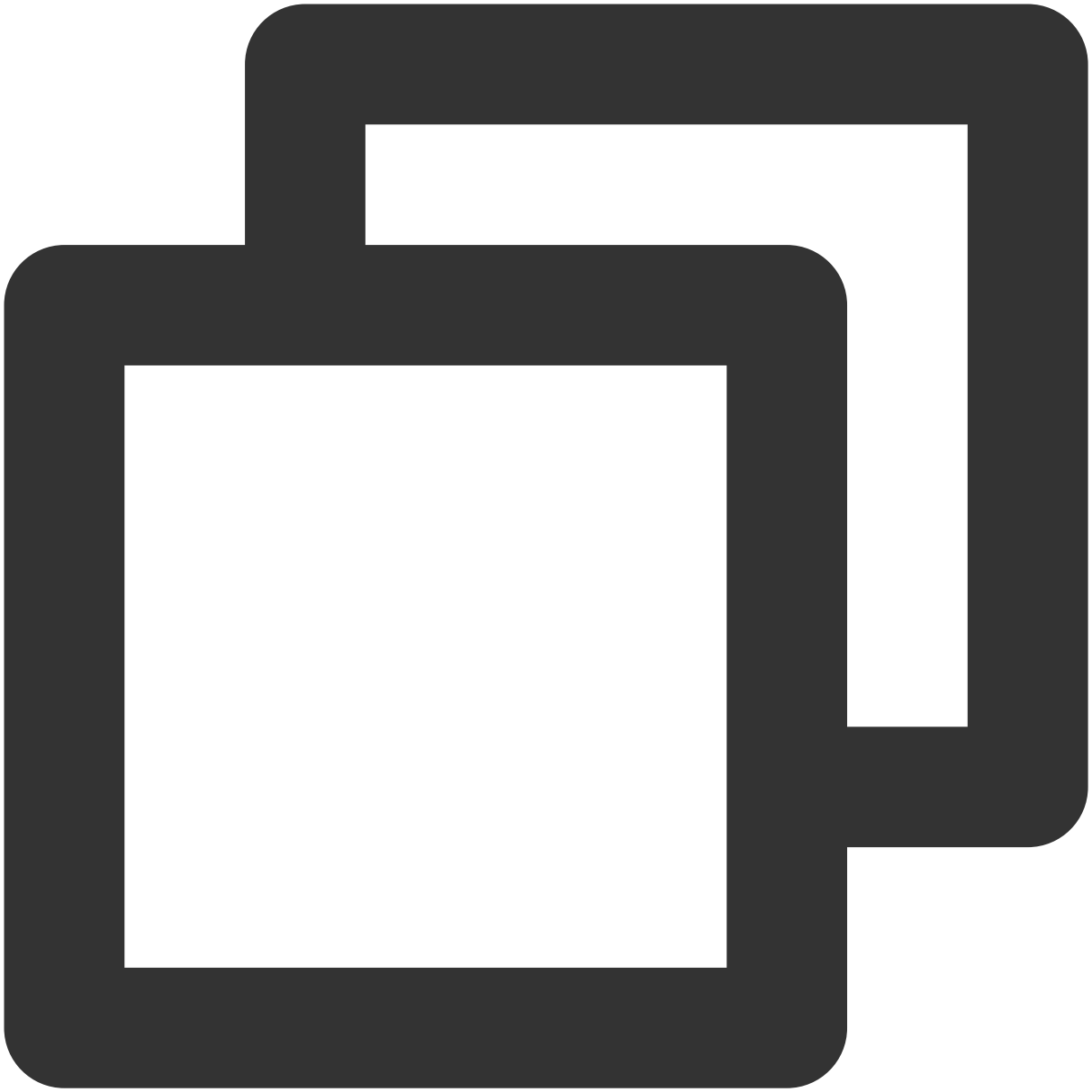
Open debugging in the official version can also open debugging in the development version or experience version first, and then cut to the official version to see the vConsole.

getRealtimeLogManager

This API is used via RealtimeLogManager wx.getRealtimeLogManager().

Feature Description: Gets the real-time log manager object.
Return Value: RealtimeLogManager.
Sample Code

©2013-2022 Tencent Cloud. All rights reserved.



```
// Mini Program End
const logger = wx.getRealtimeLogManager()
logger.info({str: 'hello world'}, 'info log', 100, [1, 2, 3])
logger.error({str: 'hello world'}, 'error log', 100, [1, 2, 3])
logger.warn({str: 'hello world'}, 'warn log', 100, [1, 2, 3])

// Plugin end, supported after the 2.16.0 base library version, only allows reporti
const logManager = wx.getRealtimeLogManager()
const logger = logManager.tag('plugin-log1')
logger.info('key1', 'value1')
logger.error('key2', {str: 'value2'})
```

©2013-2022 Tencent Cloud. All rights reserved.

```
logger.warn('key3', 'value3')
```

getLogManager

This API is used via LogManager wx.getLogManager(Object object).

Feature Description: Gets the log manager object.

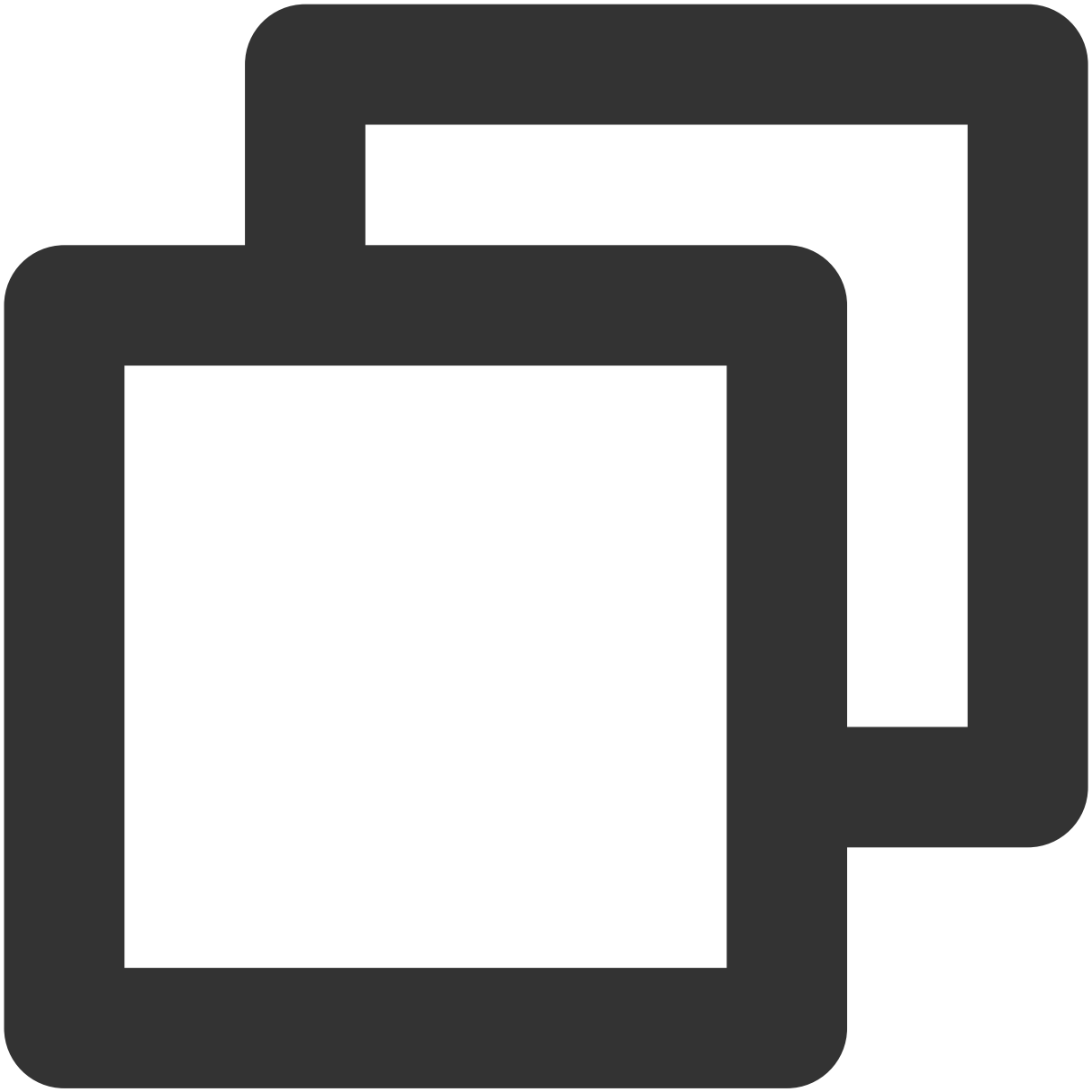
Parameter and Description: Object.

Attribute	Type	Default value	Required	Description
level	number	0	No	The value is 0/1, where 0 indicates whether the lifecycle functions of `App` and `Page` and function calls under the namespace, will be written into the log. A value of 1 means they will not be. The default value is 0.

Return Value: LogManager.

Sample Code

©2013-2022 Tencent Cloud. All rights reserved.



```
const logger = wx.getLogManager({level: 1})
logger.log({str: 'hello world'}, 'basic log', 100, [1, 2, 3])
logger.info({str: 'hello world'}, 'info log', 100, [1, 2, 3])
logger.debug({str: 'hello world'}, 'debug log', 100, [1, 2, 3])
logger.warn({str: 'hello world'}, 'warn log', 100, [1, 2, 3])
```

console

Console is a globally accessible object with a specific set of methods as follows. It can print logs in the IDE debugging panel and output logs in the client using vConsole.

©2013-2022 Tencent Cloud. All rights reserved.

debug

This method is used via console.debug()

Feature Description: Prints debug logs into the debugging panel.
Parameter and Description: any ... args, log content, with unlimited quantity.
error

This method is used via console.error()

Feature Description: Prints error logs into the debugging panel.
Parameter and Description: any ... args, log content, with unlimited quantity.
group

This method is used via console.group()

Caution:
This feature is only effective within the tool, and is implemented as an empty function in vConsole.
Feature Description: Creates a new group in the debugging panel. Subsequent outputs will be indented, indicating that they belong to the current group. The group ends after console.groupEnd is called.
Parameter and Description: string label, an optional group marker.
groupEnd

This method is used via console.groupEnd().

Caution:
This feature is only effective within the tool, and is implemented as an empty function in vConsole.
Feature Description: Ends the group created by console.group.
info

This method is used via console.info.

©2013-2022 Tencent Cloud. All rights reserved.

Feature Description: Prints info logs into the debugging panel.
Parameter and Description: any ... args, log content, with unlimited quantity.
log

> This method is used via console.log().

Feature Description: Prints logs into the debugging panel.
Parameter and Description: any ... args, log content, with unlimited quantity.
warn

> This method is used via console.warn().

Caution:
Given the limited functionality of vConsole and the varying support for console methods across different clients, it is recommended that developers only use the methods provided in this documentation when working with mini programs.
For limitations on some content displays, see Availability - Debugging.
Feature Description: Prints warn logs into the debugging panel.
Parameter and Description: any ... args, log content, with unlimited quantity.

LogManager

The LogManager instance can be obtained through getLogManager, with the specific set of methods outlined below.
debug

> This method is used via LogManager.debug().

Feature Description: Writes debug logs.
Parameter and Descriptions: Object|Array.\\ |number|string ...args, log content, with unlimited quantity. The total size of the parameters for each call should not exceed 100 Kb.
info

> This method is used via LogManager.info().

©2013-2022 Tencent Cloud. All rights reserved.

Feature Description: Writes info logs.
Parameter and Descriptions: Object|Array.\\ |number|string ...args, log content, with unlimited quantity. The total size of the parameters for each call should not exceed 100 Kb.

log

> This method is used via LogManager.log().

Feature Description: Writes log entries.
Parameter and Descriptions: Object|Array.\\ |number|string ...args, log content, with unlimited quantity. The total size of the parameters for each call should not exceed 100 Kb.

warn

> This method is used via LogManager.warn().

Caution:
A maximum of 5 MB of log content can be retained. Any content exceeding this limit will result in the deletion of older logs.
Users can upload printed logs by using the `<Button>` component's `open-type="feedback"` feature. Developers can view these through the "Feedback Management" page located in the left-hand menu of the mini program management backend.
By default, the base library writes the calls of lifecycle functions of `App` and `Page` and function calls under the namespace into the log.
Feature Description: Writes warn logs.
Parameter and Descriptions: Object|Array.\\ |number|string ...args, log content, with unlimited quantity. The total size of the parameters for each call should not exceed 100 Kb.

RealtimeLogManager

addFilterMsg

> This method is used via RealtimeLogManager.addFilterMsg(string msg).

Feature Description: Adds filter keywords. Currently, this feature does not support usage within plugins.
Parameter and Description: string msg, the addition interface for setFilterMsg. It is used to set multiple filter keywords.

©2013-2022 Tencent Cloud. All rights reserved.

error

This method is used via RealtimeLogManager.error().

Feature Description: Writes error logs. Currently, this feature does not support usage within plugins.
Parameter and Description: Object|Array.<any>|number|string ...args, log content, with unlimited quantity. The total size of the parameters for each call should not exceed 5 Kb.
getCurrentState

This method is used via Object RealtimeLogManager.getCurrentState().

Caution:
The base library internally supplements some structured data when reporting logs. If overflow occurs during reporting, warning logs will also be supplemented. Therefore, the current occupancy information obtained by this method may be larger than expected.
Feature Description: Real-time logs aggregate and report cached logs within a certain time interval. If the content cached within this time exceeds the limit, it will be discarded. This method can be used to obtain the remaining cache space information.
Return Value: Object

Attribute	Type	Description
size	number	The space already used in the current cache, measured in bytes.
maxSize	number	The maximum usable space in the current cache, measured in bytes.
logCount	number	The number of log entries in the current cache.
maxLogCount	number	The maximum number of log entries that can be stored in the current cache.

in

This method is used via RealtimeLogManager.in(Page pageInstance).

©2013-2022 Tencent Cloud. All rights reserved.

Feature Description: Sets the page where the real-time log page parameter is located. Currently, it does not support usage within plugins.
Parameter and Description: Page pageInstance, an instance of a page.
info

> This method is used via RealtimeLogManager.info().

Feature Description: Writes info logs. Currently, it does not support usage within plugins.
Parameter and Description: Object|Array.<any>|number|string ...args, log content, with unlimited quantity. The total size of the parameters for each call should not exceed 5 Kb.
setFilterMsg

> This method is used via RealtimeLogManager.setFilterMsg(string msg).

Feature Description: Sets the filter keywords. Currently, it does not support usage within plugins.
Parameter and Description: string msg, keywords for filtering, which should not exceed 1 Kb. You can search for the corresponding logs based on the set content in the mini program management backend.
tag

> This method is used via RealtimeTagLogManager RealtimeLogManager.tag(string tagName).

Feature Description: Gets the log manager instance for the given tag. Currently, it only supports usage within plugins.
Parameter and Description: string tagName, the tag name.
Return Value: RealtimeTagLogManager
warn

> This method is used via RealtimeLogManager.warn().

Feature Description: Writes warn logs, currently not supporting usage within plugins.
Parameter and Description: Object|Array.<any>|number|string ...args, log content, with unlimited quantity. The total size of the parameters for each call should not exceed 5 Kb.

©2013-2022 Tencent Cloud. All rights reserved.

RealtimeTagLogManager

addFilterMsg

This method is used via RealtimeTagLogManager.addFilterMsg(string msg).

Feature Description: Adds keywords for filtering.
Parameter and Description: string msg, the addition interface for setFilterMsg. It is used to set multiple filter keywords.

error

This method is used via RealtimeTagLogManager.error(string key, Object]Array.<any> number string value).

Feature Description: Writes error logs.
Parameter and Description:
string key, the key of the log.
Object|Array.<any>|number|string value: The value of the log. The total size of the parameters for each call should not exceed 5 Kb.

info

This method is used via RealtimeTagLogManager.info(string key, Object Array.<any> number string value).

Feature Description: Writes info logs.
Parameter and Description:
string key, the key of the log.
Object|Array.<any>|number|string value: The value of the log. The total size of the parameters for each call should not exceed 5 Kb.

setFilterMsg

This method is used via RealtimeTagLogManager.setFilterMsg(string msg).

Feature Description: Sets the keyword for filtering.

©2013-2022 Tencent Cloud. All rights reserved.

Parameter and Description: string msg, keywords for filtering, which should not exceed 1 Kb. You can search for the corresponding logs based on the set content in the mini program management backend.

warn

This method is used via RealtimeTagLogManager.warn(string key, ObjectJArray.< any> number string value)

Feature Description: Writes warn logs.

Parameter and Description:

string key, the key of the log.

Object|Array.<any>|number|string value: The value of the log. The total size of the parameters for each call should not exceed 5 Kb.

©2013-2022 Tencent Cloud. All rights reserved.

Routing

Last updated：2024-07-12 19:38:30

switchTab

This API is used via wx.switchTab(Object object).

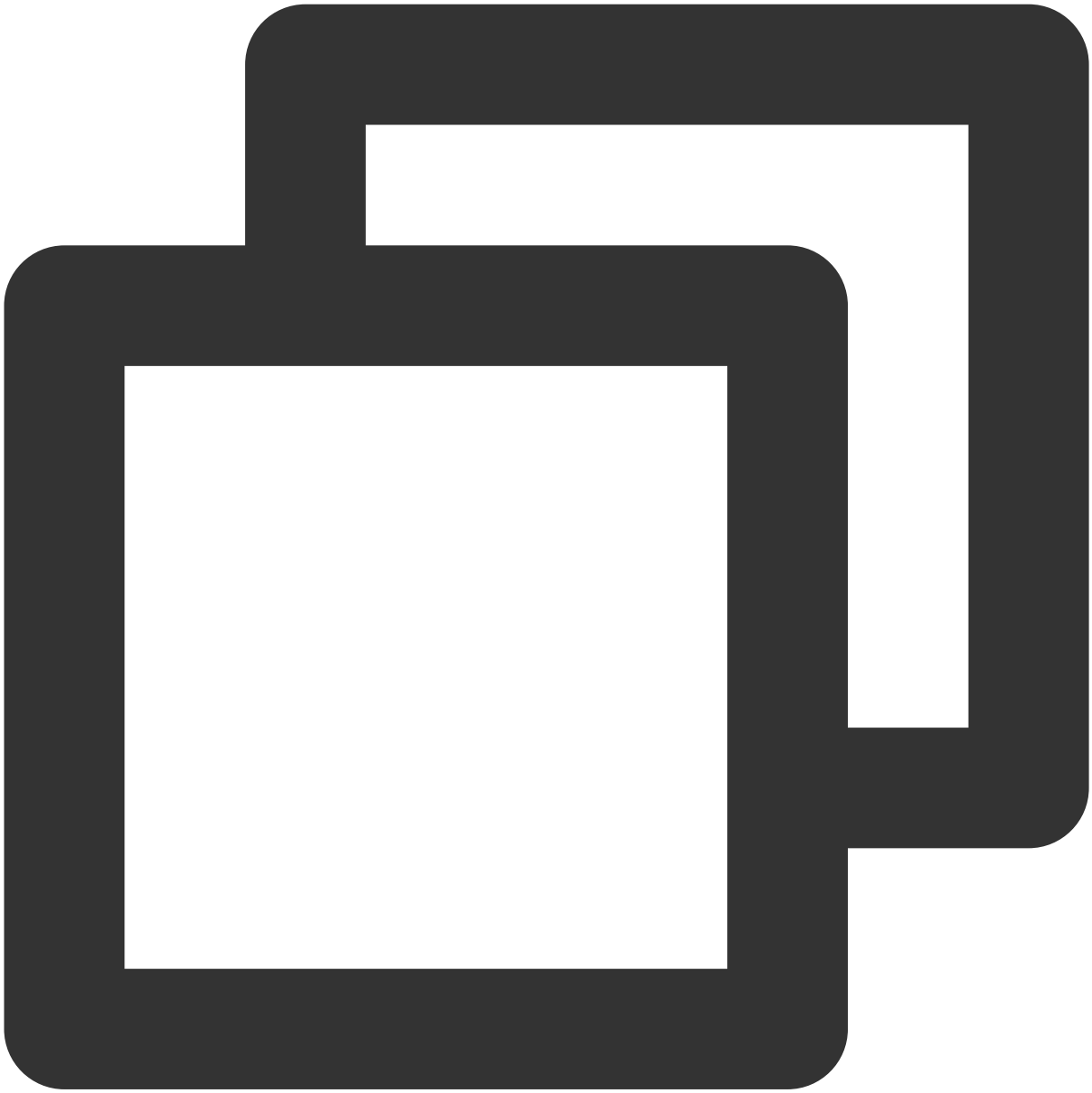
Feature Description: Redirects to the tabBar page, simultaneously closing all other non-tabBar pages.

Parameter and Description: Object.

Attribute	Type	Default value	Required	Description
url	string	-	Yes	The path to the tabBar page that needs to be redirected (code package path) (must be defined on the page in the tabBar field of app.json). The path cannot carry parameters.
success	function	-	No	Callback Function of Successful Interface Call
fail	function	-	No	Callback Function of Successful Interface Call
complete	function	-	No	Callback function executed upon the completion of the interface invocation (both successful and unsuccessful invocations)

Sample Code

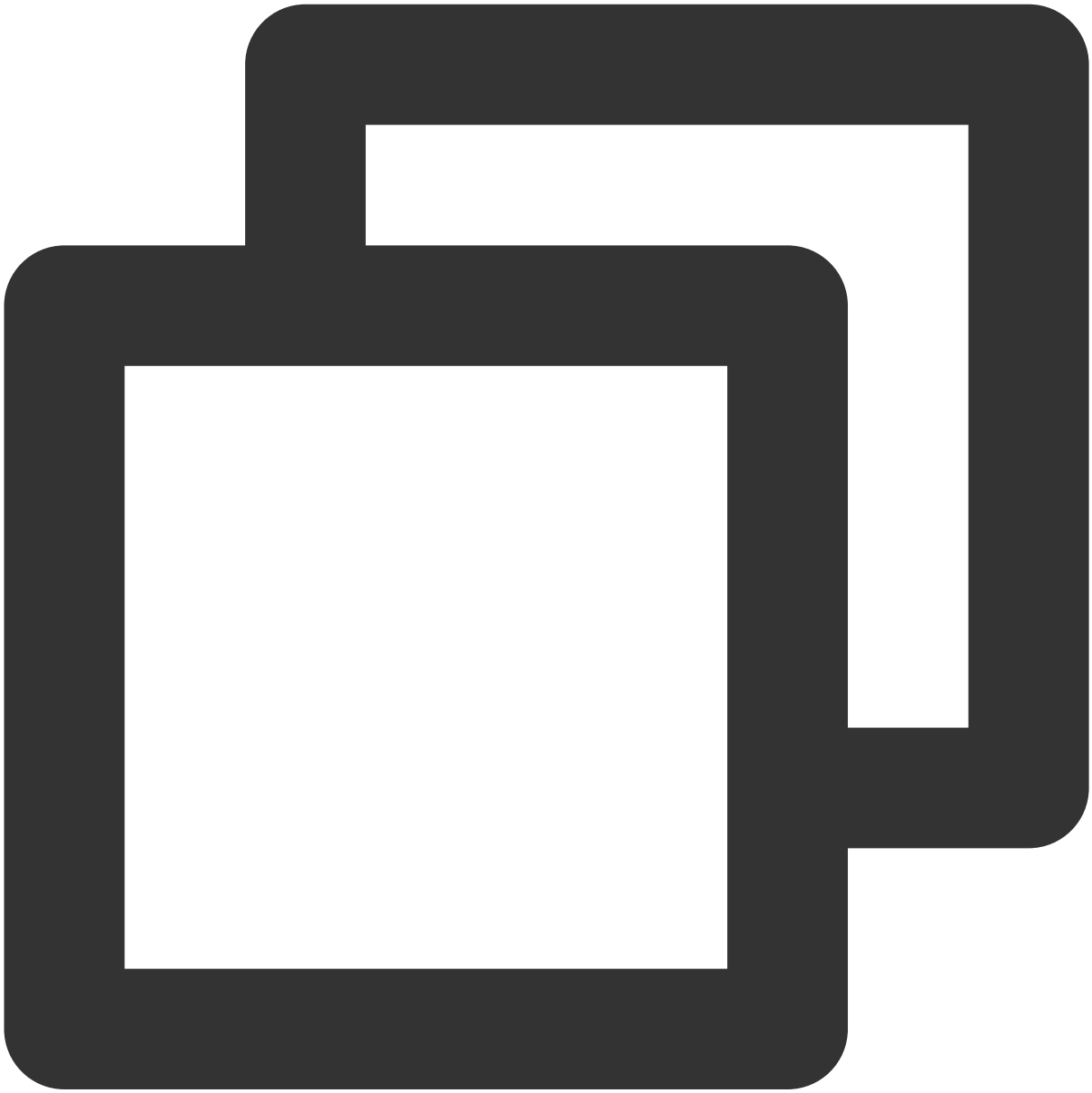
©2013-2022 Tencent Cloud. All rights reserved.



```
// app.json
{
  "tabBar": {
    "list": [{
      "pagePath": "index",
      "text": "Homepage"
    },{
      "pagePath": "other",
      "text": "Other"
    }]
  }
```

©2013-2022 Tencent Cloud. All rights reserved.

```
}
```



```
wx.switchTab({
  url: '/index'
})
```

reLaunch

©2013-2022 Tencent Cloud. All rights reserved.

This method is used via wx.reLaunch(Object object).

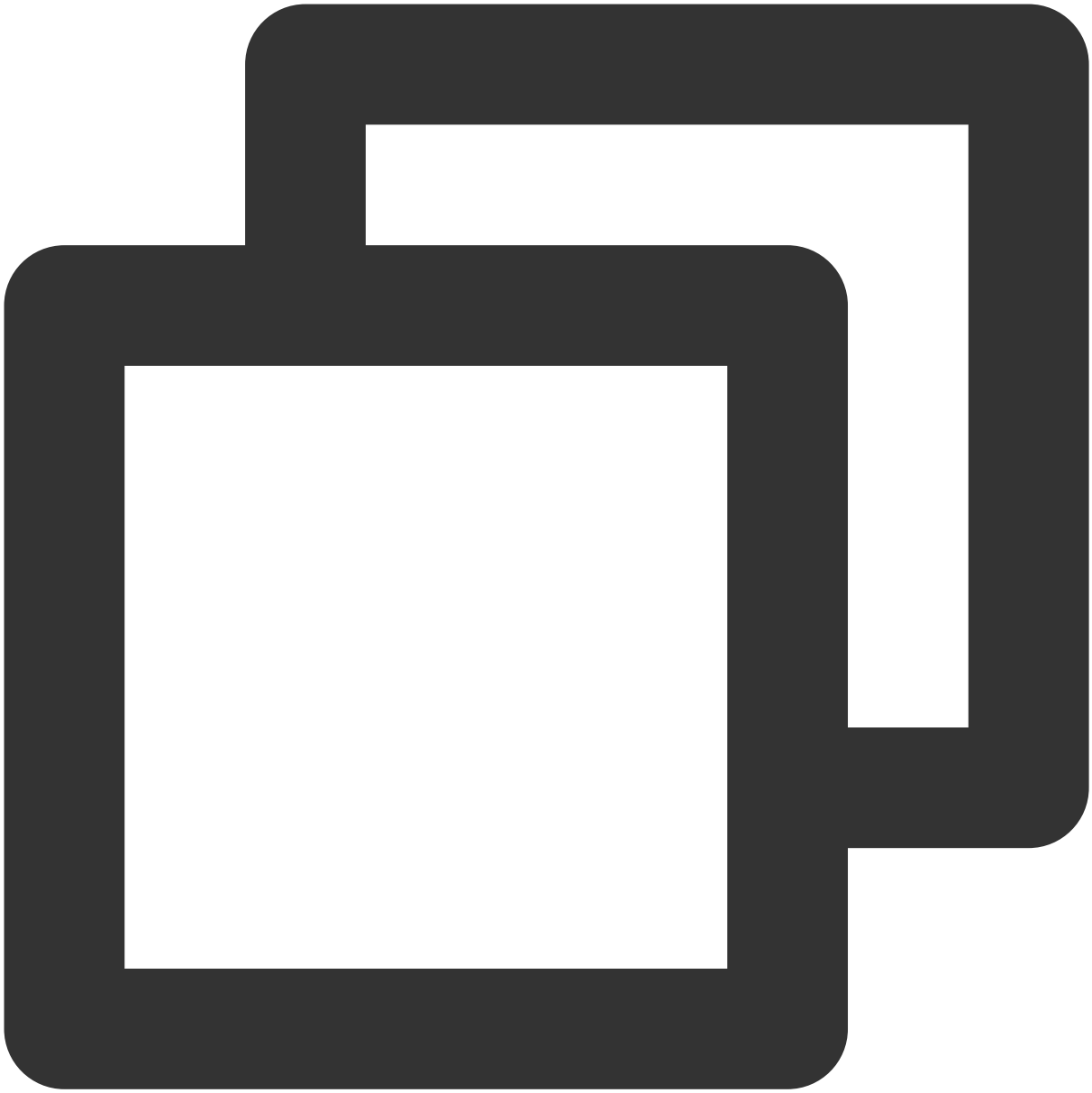
Feature Description: Closes all pages and opens a specific page within the application.

Parameter and Description: Object.

Attribute	Type	Default value	Required	Description
url	string	-	Yes	The path to a non-tabBar page within the application to which you want to redirect (code package path). This path can be followed by parameters. Parameters are separated from the path by a `?`, with parameter keys and values connected by a `=`, and different parameters separated by a `&`. For example, `'path?key=value&key2=value2'`
success	function	-	No	Callback Function of Successful Interface Call
fail	function	-	No	Callback Function of Failing Interface Call
complete	function	-	No	Callback function executed upon the completion of the interface invocation (both successful and unsuccessful invocations)

Sample Code

©2013-2022 Tencent Cloud. All rights reserved.



```
wx.reLaunch({
  url: 'test?id=1'
})
```

redirectTo

This API is used via wx.redirectTo(Object object).

©2013-2022 Tencent Cloud. All rights reserved.

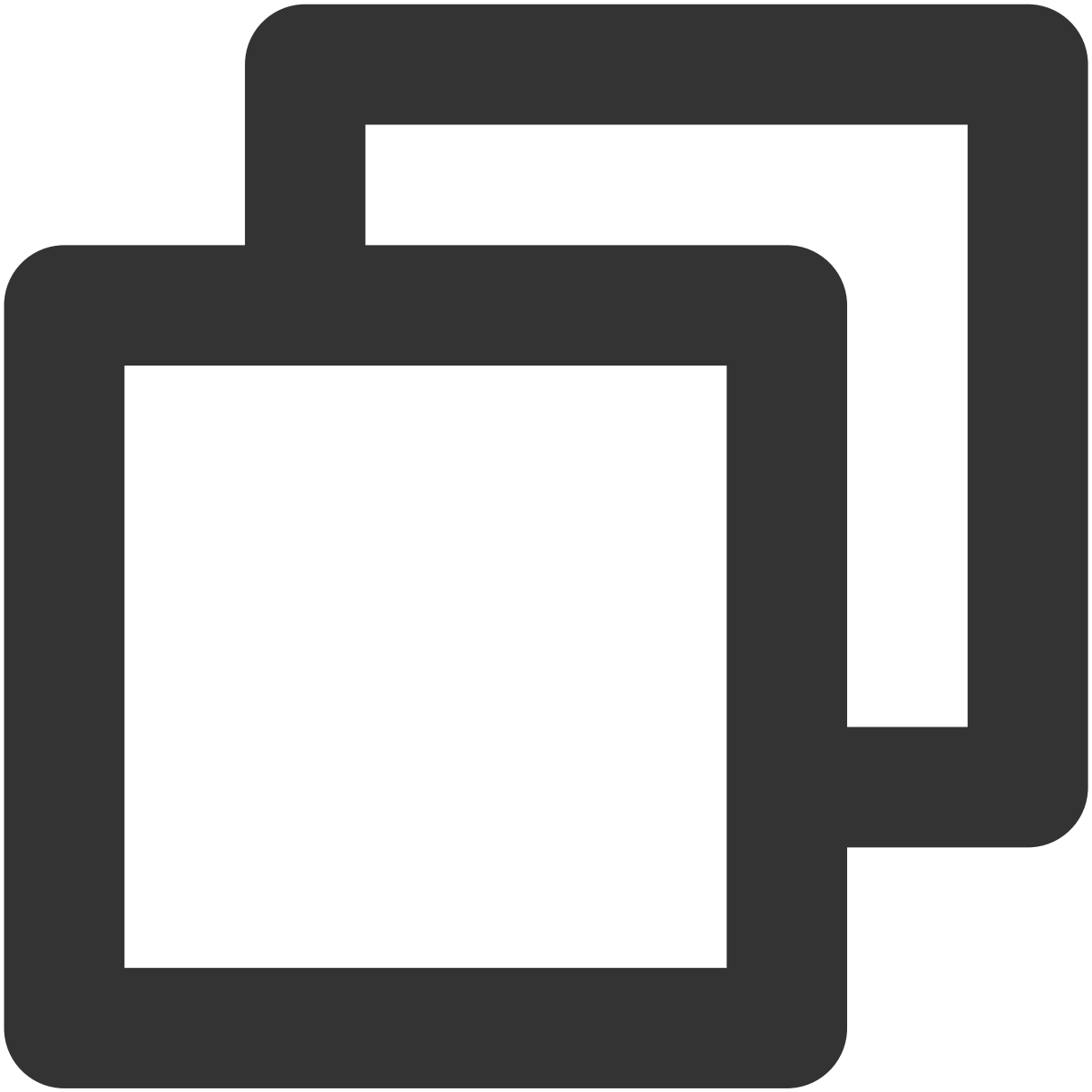
Feature Description: Close the current page and jump to a page within the app. However, it is not allowed to jump to a tabbar page.

Parameter and Description: Object.

Attribute	Type	Default value	Required	Description
url	string	-	Yes	The path to a page within the application to which you want to redirect (code package path). This path can be followed by parameters. Parameters are separated from the path by a `?`, with parameter keys and values connected by a =, and different parameters separated by a `&`. For example, `'path?key=value&key2=value2'`
success	function	-	No	Callback Function of Successful Interface Call
fail	function	-	No	Callback Function of Failing Interface Call
complete	function	-	No	Callback function executed upon the completion of the interface invocation (both successful and unsuccessful invocations)

Sample Code

©2013-2022 Tencent Cloud. All rights reserved.



```
wx.redirectTo({
  url: 'test?id=1'
})
```

navigateTo

This API is used via wx.navigateTo(Object object).

©2013-2022 Tencent Cloud. All rights reserved.

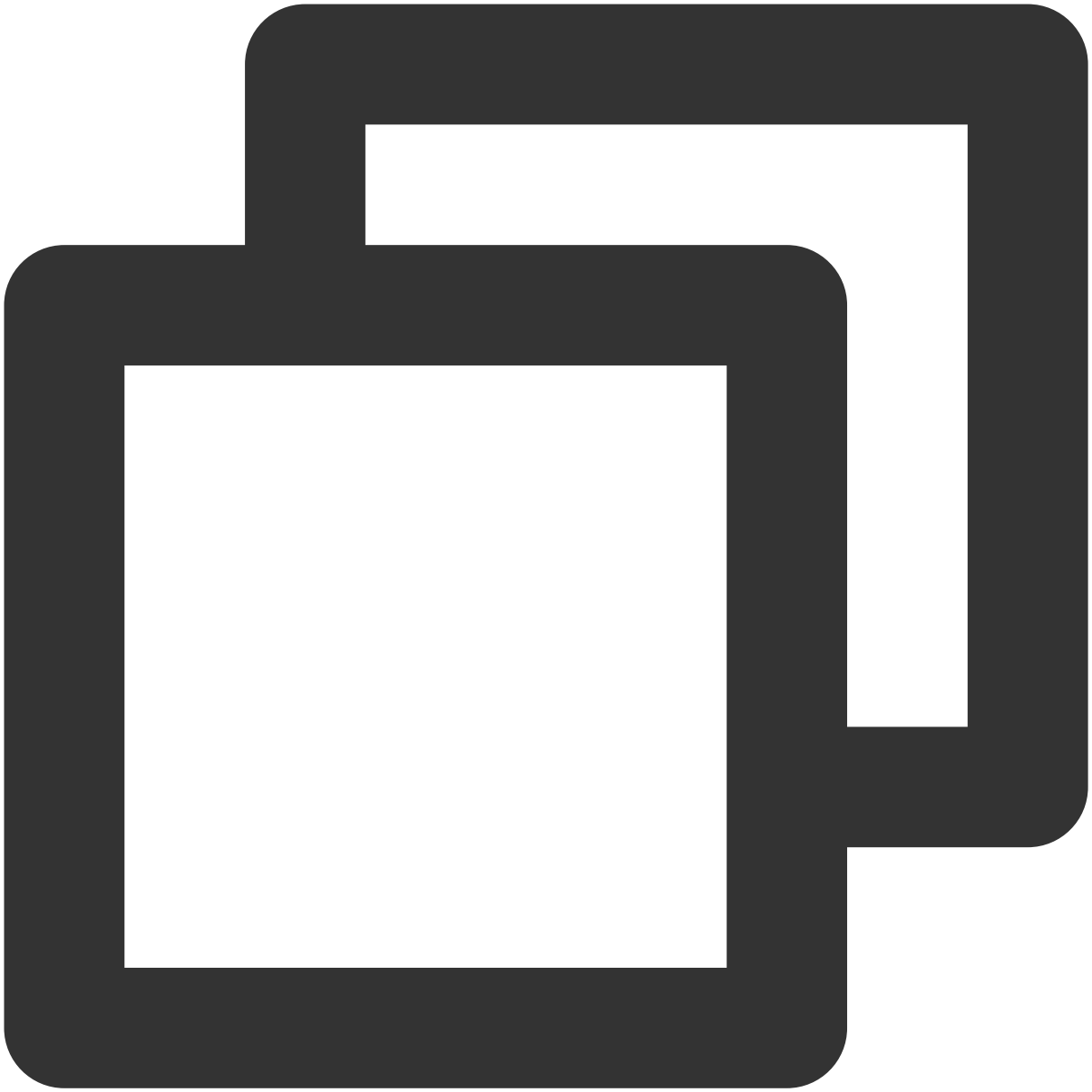
Feature Description: Retains the current page and redirects to a specific page within the application, but cannot redirect to a tabbar page. Using wx.navigateBack allows a return to the original page. The page stack in the mini program can be up to ten layers deep.

Parameter and Description: Object.

Attribute	Type	Default value	Required	Description
url	string	-	Yes	The path to the non-tabBar page within the application to which you want to navigate (code package path). This path can be followed by parameters. Parameters are separated from the path by a `?`, parameter keys and values are connected by an equals sign, and different parameters are separated by a `&`; for example, `'path?key=value&key2=value2'`
success	function	-	No	Callback Function of Successful Interface Call
fail	function	-	No	Callback Function of Failing Interface Call
complete	function	-	No	Callback function executed upon the completion of the interface invocation (both successful and unsuccessful invocations)

Sample Code

©2013-2022 Tencent Cloud. All rights reserved.



```
wx.navigateTo({
  url: 'test?id=1',
})


//test.js
Page({
  onLoad: (option){
      console.log(option.query)
    }
})
```

©2013-2022 Tencent Cloud. All rights reserved.

navigateBack

This API is used via wx.navigateBack(Object object).

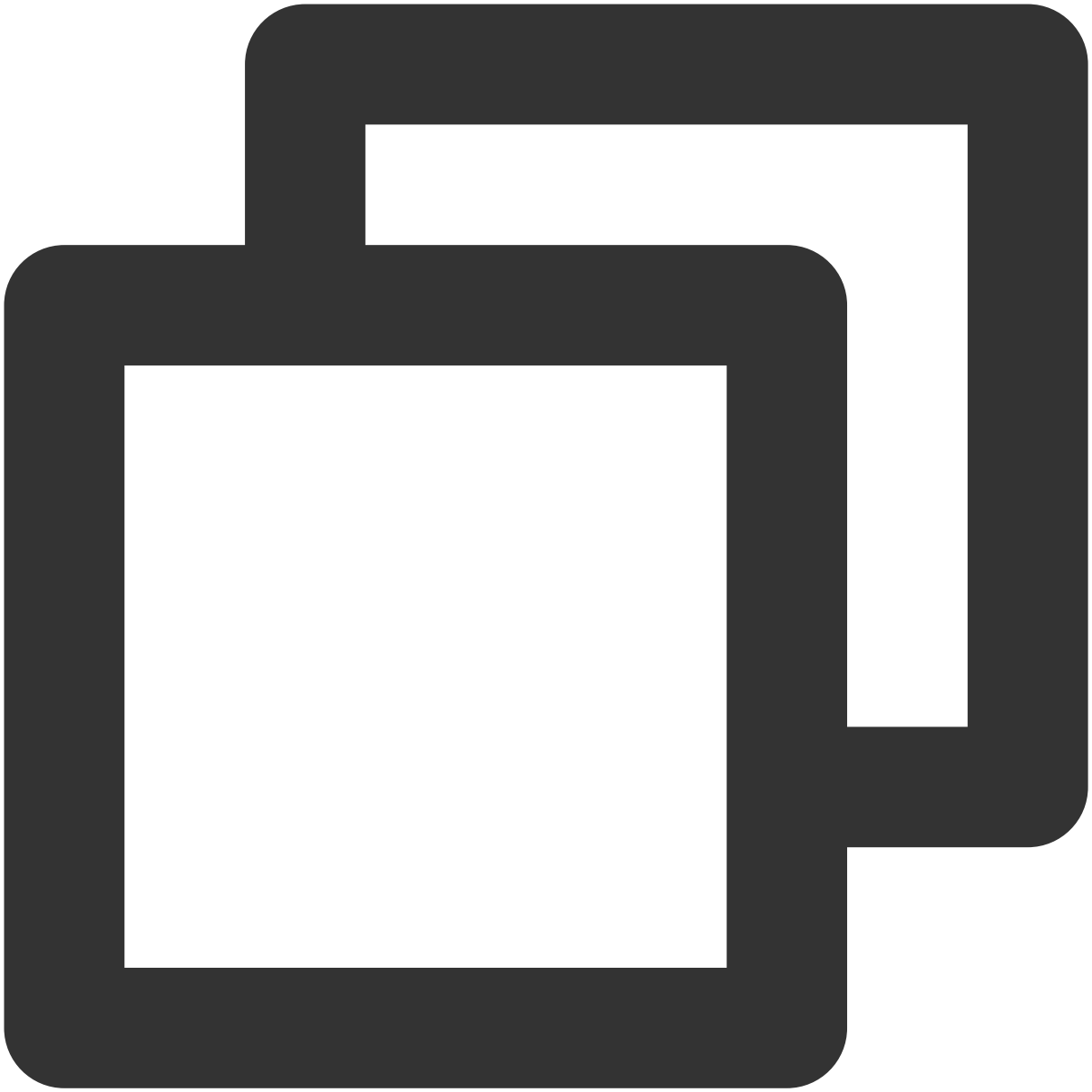
Feature Description: Close the current page, return to the previous page or multi-level page, you can get the current page stack by getCurrentPages, decide how many levels you need to return.

Parameter and Description: Object.

Attribute	Type	Default value	Required	Description
delta	number	1	No	The number of pages returned. If delta exceeds the number of existing pages, it will return to the home page.
success	function	-	No	Callback Function of Successful Interface Call
fail	function	-	No	Callback Function of Failing Interface Call
complete	function	-	No	Callback function executed upon the completion of the interface invocation (both successful and unsuccessful invocations)

Sample Code

©2013-2022 Tencent Cloud. All rights reserved.



```
// Note: When calling navigateTo for navigation, the page that calls this method wi

// This is page A
wx.navigateTo({
  url: 'B?id=1'
})

// This is page B
wx.navigateTo({
  url: 'C?id=1'
})
```

©2013-2022 Tencent Cloud. All rights reserved.

```
// Perform navigateback within page C will return to page A.
wx.navigateBack({
  delta: 2
})
```

EventChannel

Inter-page event communication channel.

EventChannel.emit

This method is used via EventChannel.emit(string eventName, any args).

Feature Description: Triggers an event.
Parameter and Description: string eventName, event name; any args, event parameter.

EventChannel.off

This method is used via EventChannel.off(string eventName, EventCallback fn).

Feature Description: Cancels the monitoring of an event. When a second parameter is provided, only the specified monitoring function is canceled; otherwise, all monitoring functions are canceled. The method is used via EventChannel.off(string eventName, function fn).
Parameter and Description: string eventName, event name; function fn, event listener function; any args, parameters of triggered event.

EventChannel.on

This method is used via EventChannel.on(string eventName, EventCallback fn).

Feature Description: Continuously monitors an event. This method is used via EventChannel.on(string eventName, function fn).
Parameter and Description: string eventName, event name; function fn, event listener function; any args, parameters of triggered event.

©2013-2022 Tencent Cloud. All rights reserved.

EventChannel.once

This method is used via EventChannel.once(string eventName, EventCallback fn)

Feature Description: Monitors an event once, becoming invalid after triggering. This method is used via EventChannel.once(string eventName, function fn).

Parameter and Description: string eventName, event name; function fn, event listener function; any args, parameters of triggered event.

©2013-2022 Tencent Cloud. All rights reserved.

Redirect

Last updated：2024-07-12 19:38:08

exitMiniProgram

The method of utilizing this API is wx.exitMiniProgram(Object object).

Function Description: This function facilitates the termination of the current mini program. A successful invocation necessitates a click action.

Parameters and Description: Object object.

Attributes	Types	Default value	Required	Note
success	function	-	Not required	Successful Interface Call Callback Function
fail	function	-	Not required	Callback function for unsuccessful interface invocation
complete	function	-	Not required	Callback function executed upon the completion of the interface invocation (both successful and unsuccessful invocations)

navigateToMiniProgram

The method for utilizing this API is wx.navigateToMiniProgram(Object object)

Function Description: Launches another mini program.

Parameters and Description: Object object.

Attributes	Types	Default value	Required	Note
appId	string	-	Supported	The appId of the mini program to be launched
path	string	-	Not	The path of the page to be opened, if it is empty then

©2013-2022 Tencent Cloud. All rights reserved.

			required	the first page will be opened, the part of path after ? The part after the path will become query, you can get the query data in the callback function of App.onLaunch, App.onShow and Page.onLoad of the mini program or wx.getLaunchOptionsSync.
extraData	object	-	Not required	Data to be passed to the target mini program; the target mini program can retrieve this data in `App.onLaunch` , `App.onShow`
envVersion	string	release	Not required	The version of the mini program to be launched. This parameter is only effective when the current mini program is in development or trial version. If the current mini program is in the official version, the mini program to be launched will definitely be the official version. The valid values are as follows: develop: Development Version trial: Trial Version release: Official Version
success	function	-	Not required	Callback function for successful interface invocation
fail	function	-	Not required	Callback function for unsuccessful interface invocation
complete	function	-	Not required	Callback function executed upon the completion of the interface invocation (both successful and unsuccessful invocations)

Usage Limits

Requires user-initiated redirection

If the user does not click on any location within the mini-program page, the developer will be unable to invoke this interface to automatically redirect to another mini-program.

User confirmation required for redirection

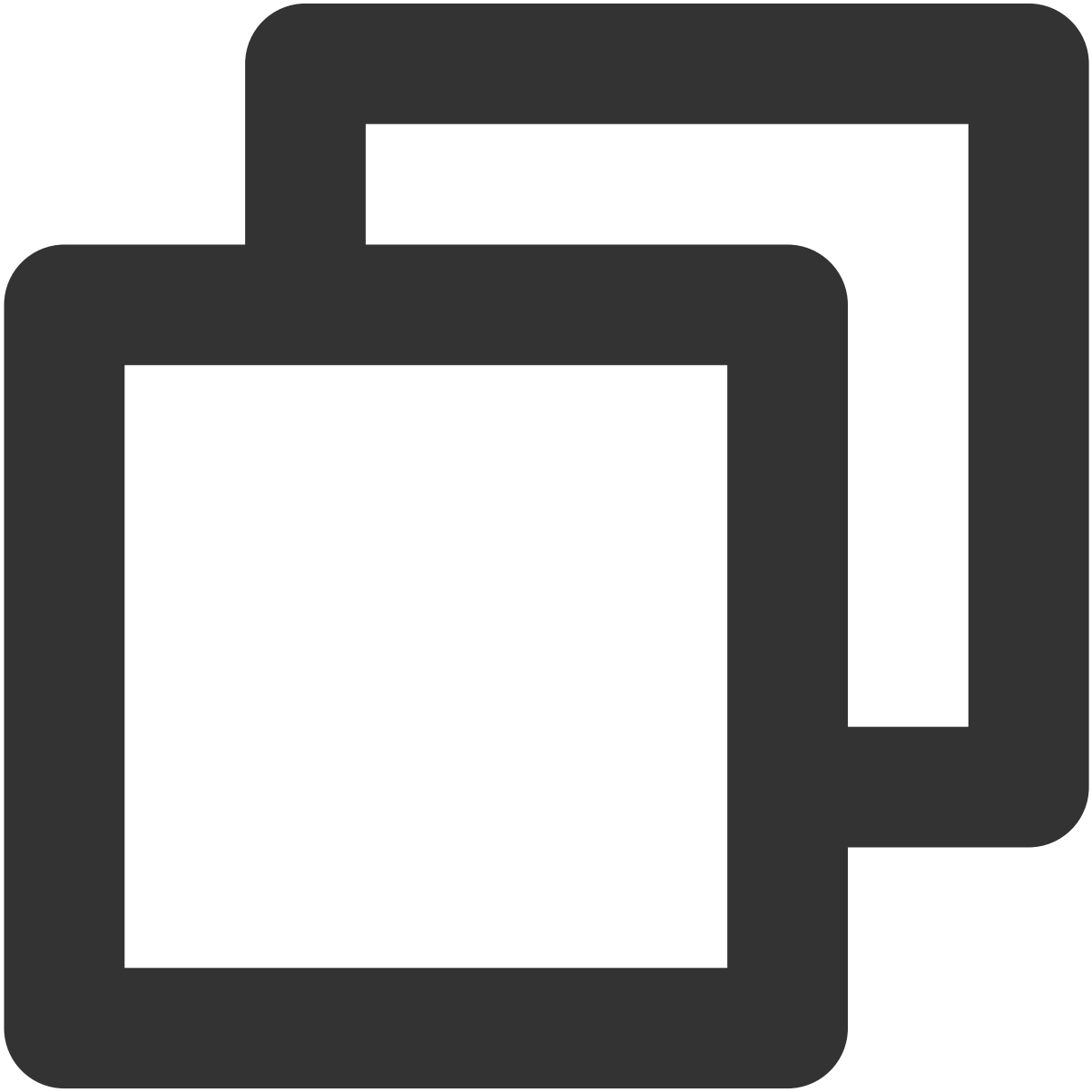
Before redirecting to another mini-program, a unified pop-up window will be added, asking whether to redirect. The user can only redirect to another mini-program after confirmation. If the user clicks on **cancel**, the callback `fail cancel` will be triggered.

Each mini-program is limited to redirecting to no more than 10 other mini-programs.

©2013-2022 Tencent Cloud. All rights reserved.

When developers submit a new version of the mini-program code, if the function to redirect to other mini-programs is used, it is necessary to declare in the code configuration the list of mini-programs to which it will redirect, limited to no more than 10, otherwise it will not pass the review. This list can be updated when a new version is released and does not support dynamic modification. For configuration methods, see Configuration. When invoking this interface, the appId to which it redirects must be in the configuration list, otherwise the callback `fail appId "${appId}" is not in navigateToMiniProgramAppIdList` will be triggered.

Sample Code:



```
wx.navigateToMiniProgram({
  appId: '',
```

©2013-2022 Tencent Cloud. All rights reserved.

```
  path: 'page/index/index?id=123',
  extraData: {
    foo: 'bar'
  },
  envVersion: 'develop',
  success(res) {
    // Successfully opened
  }
})
```

navigateBackMiniProgram

The usage method for this API is wx.navigateBackMiniProgram(Object object).

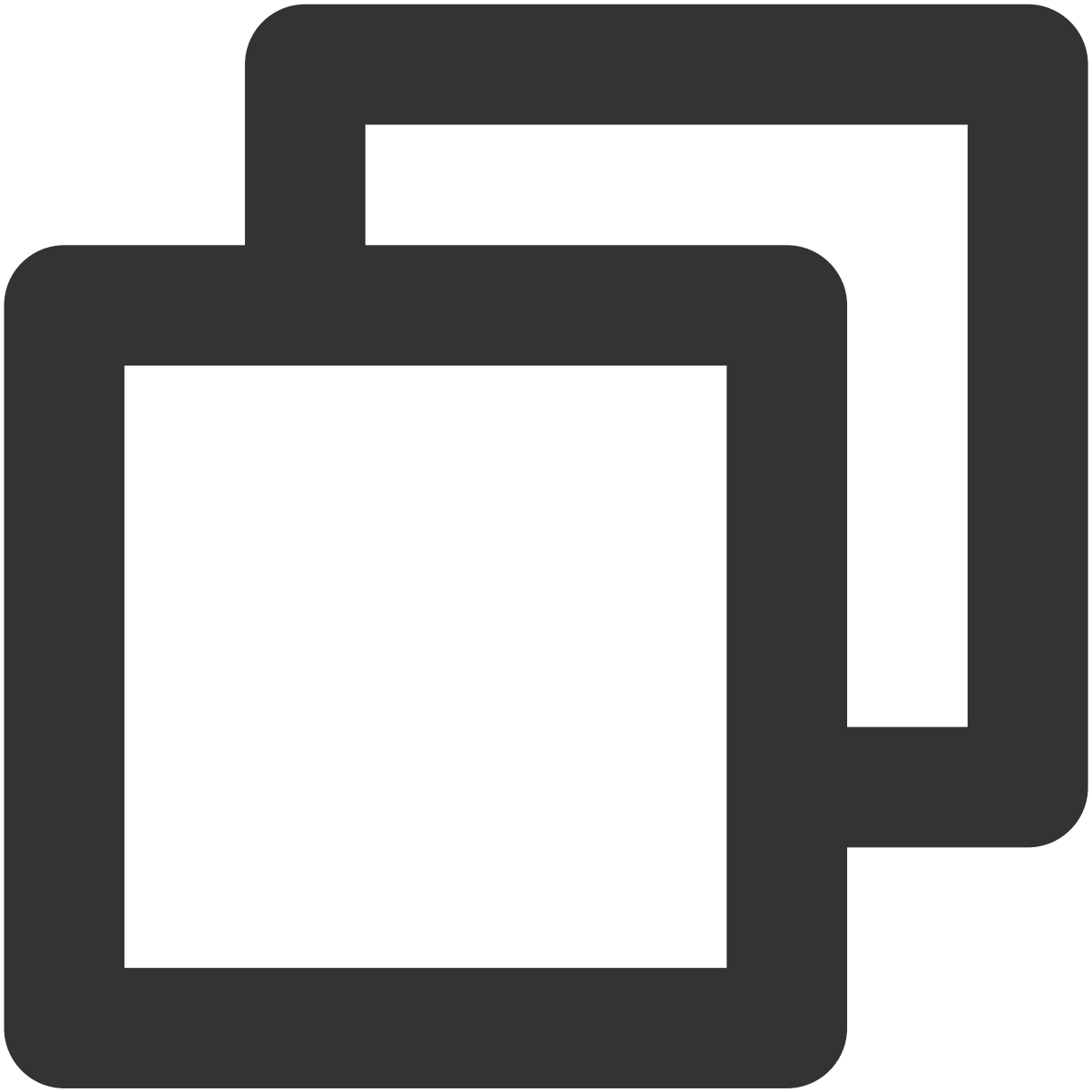
Function Description: Returns to the previous mini-program. This function can only be successfully invoked when the current mini-program is opened by another mini-program.

Parameters and Description: Object object

Attributes	Types	Default value	Required	Note
extraData	Object	{}	Not required	Data that needs to be returned to the previous mini-program can be obtained in the App.onShow of the previous mini-program.
success	function	-	Not required	Callback function for successful interface invocation
fail	function	-	Not required	Callback function for unsuccessful interface invocation
complete	function	-	Not required	Callback function executed upon the completion of the interface invocation (both successful and unsuccessful invocations)

Example Code

©2013-2022 Tencent Cloud. All rights reserved.



```
wx.navigateBackMiniProgram({
  extraData: {
    foo: 'bar'
  },
  success(res) {
    // Successful return
  }
})
```

©2013-2022 Tencent Cloud. All rights reserved.

Forwarding

Last updated：2024-07-12 19:37:43

hideShareMenu

This API is used via wx.hideShareMenu(Object object).

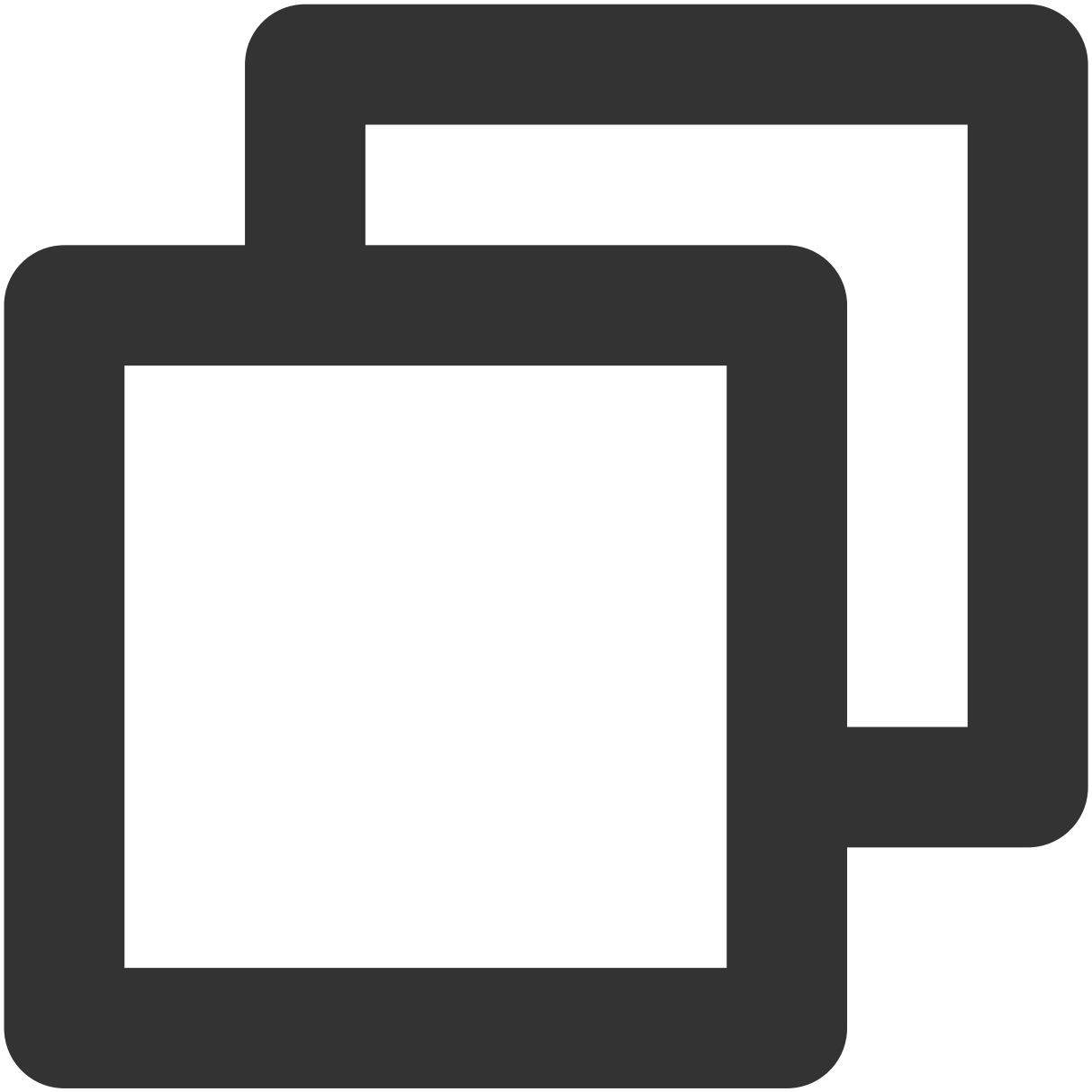
Feature Description: Hides the "Share with Friends" button when the top right corner menu is clicked.

Parameter and Description: Object.

Attribute	Type	Default value	Required	Description
hideShareItems	Array.<string>	['qq', 'qzone', 'wechatFriends', 'wechatMoment']	No	QQ: Control the visibility of the "Forward" option. Qzone: Manage the visibility of the "Share to Host Client QQ Space" option. WeChatFriends: Manage the visibility of the "Share to Host Client We Chat Friends" option. WeChatMoment: Manage the visibility of the "Share to Host Client We Chat Moments" option. Without the hideShareItems parameter, the "Forward", "Share to Host Client QQ Space", "Share to Host Client We Chat Friends", and "Control Visibility" options are all hidden by default.
success	function	-	No	Callback Function of Successful Interface Call
fail	function	-	No	Callback Function of Failing Interface Call
complete	function	-	No	Callback function executed upon the completion of the interface invocation (both successful and unsuccessful invocations)

Sample Code

©2013-2022 Tencent Cloud. All rights reserved.



```
wx.hideShareMenu({
    hideShareItems: ['qq', 'qzone', 'wechatFriends', 'wechatMoment']
})
```

showShareMenu

This API is used via wx.showShareMenu(Object object).

©2013-2022 Tencent Cloud. All rights reserved.

Feature Description: When clicking the top-right corner menu, the "Share with Friends", "Share to Host Client QQ Space", "Share to Host Client We Chat Friends", and "Share to Host Client We Chat Moments" buttons for the current page will be displayed.

Parameter and Description: Object.

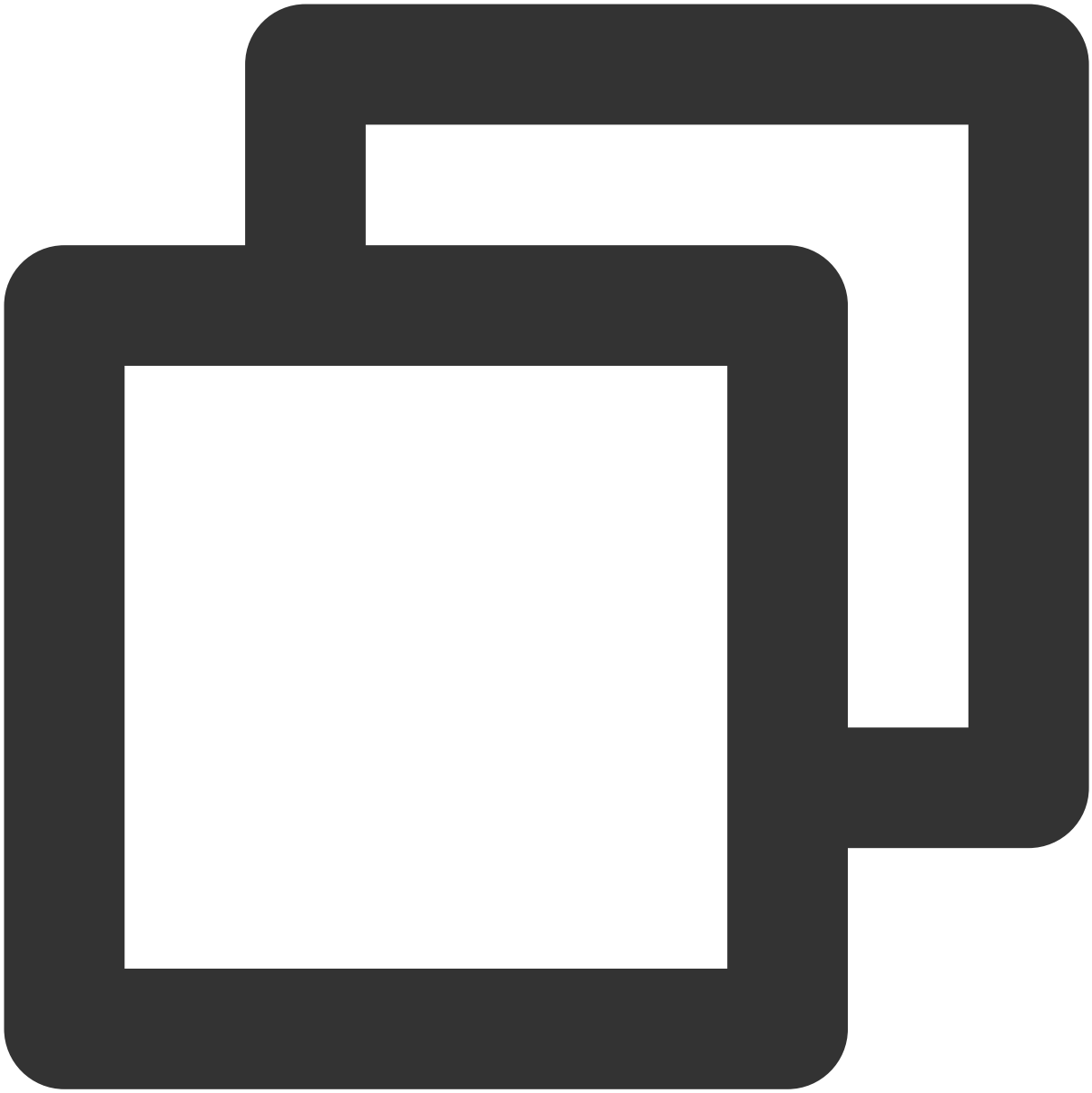
Attribute	Type	Default value	Required	Description
showShareItems	Array.<string>	['qq', 'qzone', 'wechatFriends', 'wechatMoment']	No	See the parameter descriptions in object.showShareItems as follows.
withShareTicket	boolean	false	No	Whether to use forwarding with shareTicket, supported from Base Library version 1.4.0 on Android and from Base Library version 1.4.0 on iOS.
success	function	-	No	Callback Function of Successful Interface Call
fail	function	-	No	Callback Function of Failing Interface Call
complete	function	-	No	Callback function executed upon the completion of the interface invocation (both successful and unsuccessful invocations)

object.showShareItems Parameter

Value	Description
'qq'	Control whether to display "Forwarding"
'qzone'	Control whether to display "Share to host client QQ Space"
'wechatFriends'	Control whether to display "Host Client We Chat Friends"
'wechatMoment'	Control whether to hide "Share to Host Client We Chat Moments". Without the hideShareItems parameter, "Forwarding", "Share to Host Client QQ Space", "Share to Host Client We Chat Friends", and "Control whether to hide" are all hidden by default.

Sample Code

©2013-2022 Tencent Cloud. All rights reserved.



```
wx.showShareMenu({
  showShareItems: ['qq', 'qzone', 'wechatFriends', 'wechatMoment']
})
```

updateShareMenu

This API is used via wx.updateShareMenu(Object object).

©2013-2022 Tencent Cloud. All rights reserved.

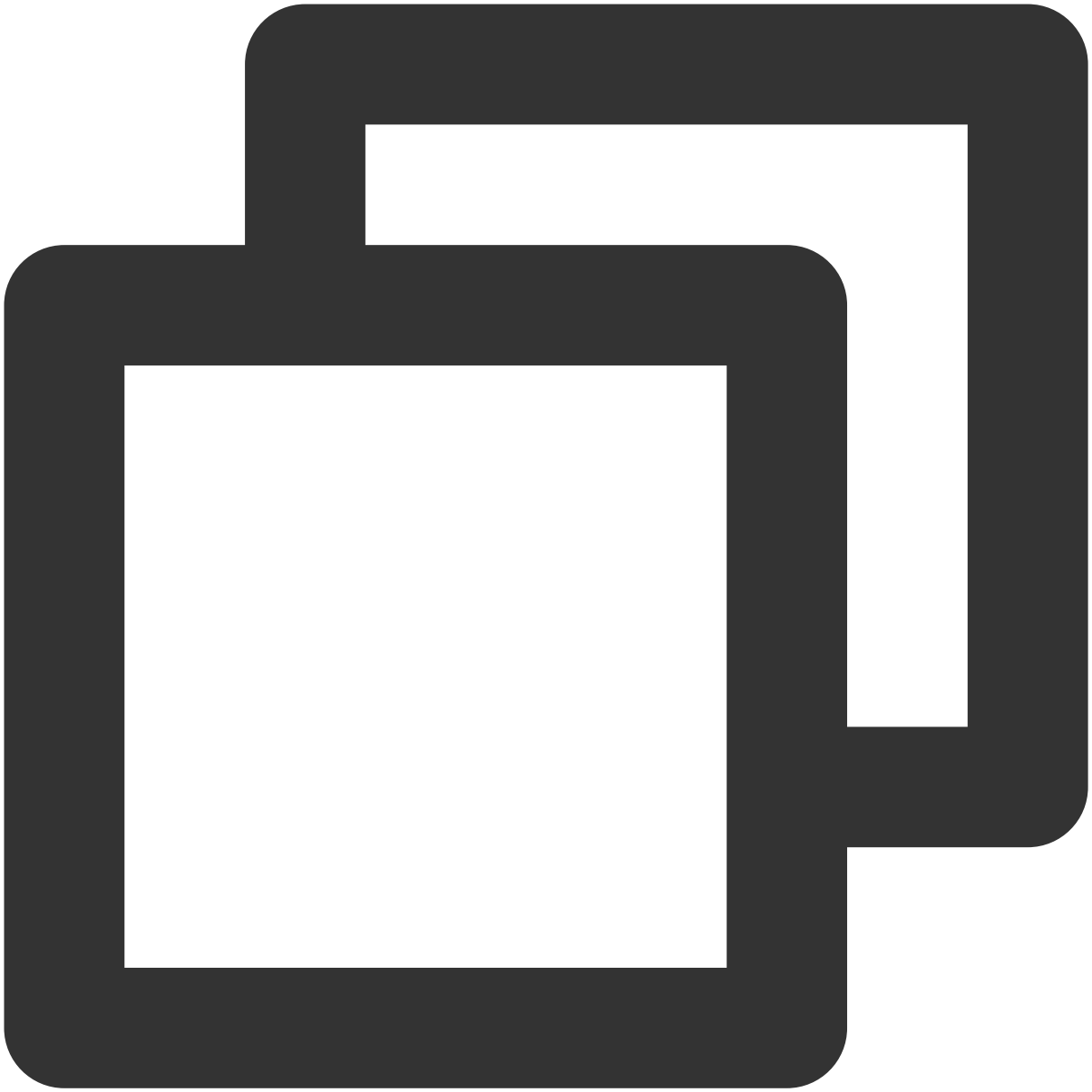
Feature Description: Updates forwarding attributes.

Parameter and Description: Object.

Attribute	Type	Default value	Required	Description
withShareTicket	boolean	false	No	Whether to use forwarding with shareTicket
success	function	-	No	Callback Function of Successful Interface Call
fail	function	-	No	Callback Function of Failing Interface Call
complete	function	-	No	Callback function executed upon the completion of the interface invocation (both successful and unsuccessful invocations)

Sample Code

©2013-2022 Tencent Cloud. All rights reserved.



```
wx.updateShareMenu({
  withShareTicket: true
})
```

showShareImageMenu

This API is used via wx.showShareImageMenu(Object object)

©2013-2022 Tencent Cloud. All rights reserved.

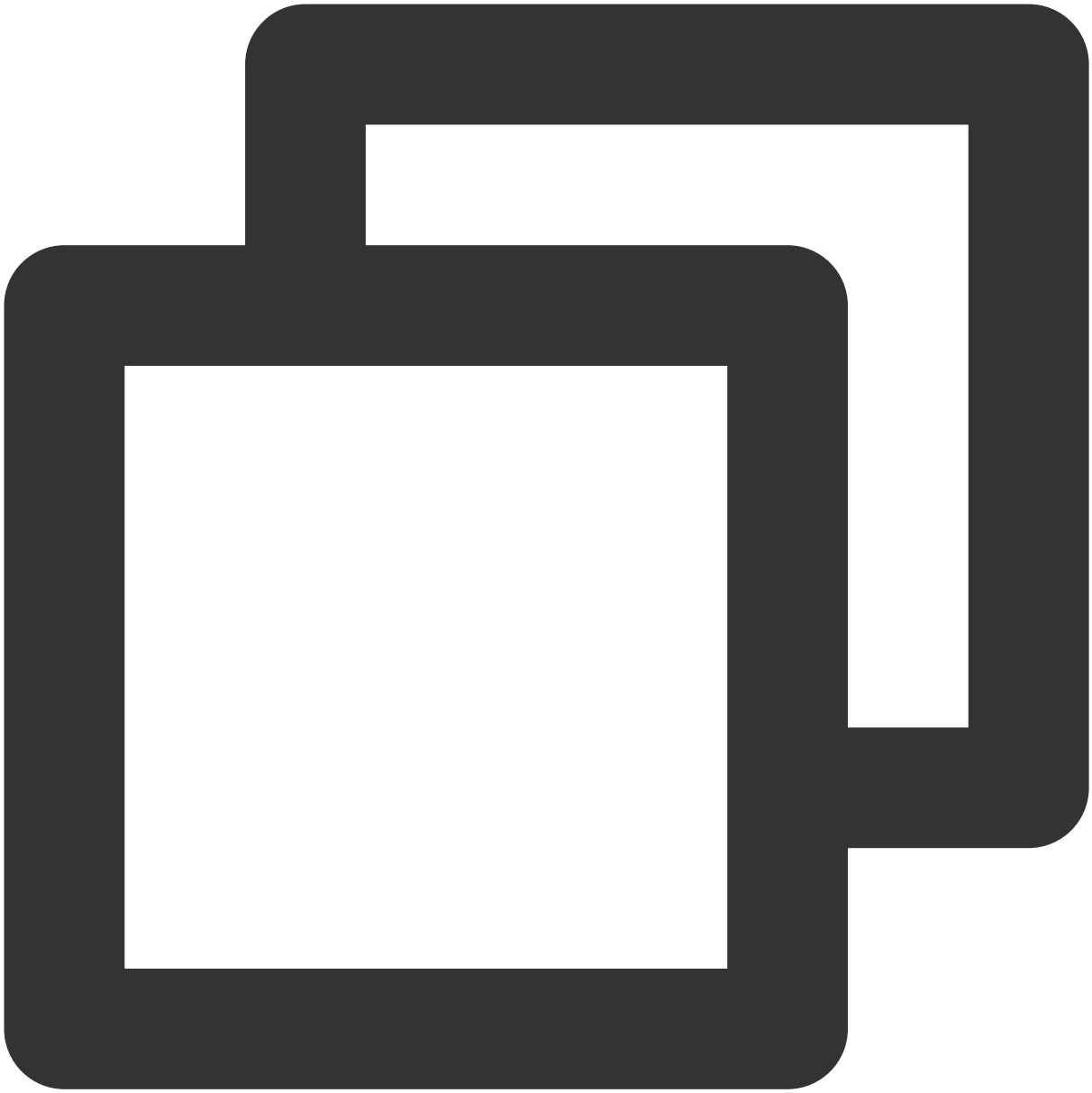
Function description: Open the picture sharing pop-up window, and you can send pictures to friends, collect them, or download them.

Parameters and description: Object object.

Attribute	Type	Default value	Required	Description
path	string	-	Yes	The image address to be shared must be a local path or a temporary path
style	string	default	No	Share styles
needShowEntrance	string	false	No	Whether the shared picture message needs to have a mini program entry (only some mini program categories are available)
entrance	string	false	No	Open the path of the mini program from the entrance of the message mini program. If the current page allows sharing with friends, the default is the path of the current page. Otherwise, the default is the home page of the mini program.
success	function	-	No	Callback Function of Successful Interface Call
fail	function	-	No	Callback Function of Failing Interface Call
complete	function	-	No	Callback function executed upon the completion of the interface invocation (both successful and unsuccessful invocations)

Sample Code

©2013-2022 Tencent Cloud. All rights reserved.



```
  wx.downloadFile({
  url: 'https://res.wx.qq.com/wxdoc/dist/assets/img/demo.ef5c5bef.jpg',
  success: (res) => {
    wx.showShareImageMenu({
      path: res.tempFilePath
    })
  }
})
```

©2013-2022 Tencent Cloud. All rights reserved.

onCopyUrl

This API is used via wx.onCopyUrl(function listener)

Function description：Listen to the event triggered when the user clicks the "Copy Link" button in the menu in the upper right corner. This interface is a Beta version and is currently only supported on the Android platform.

Parameters and description：

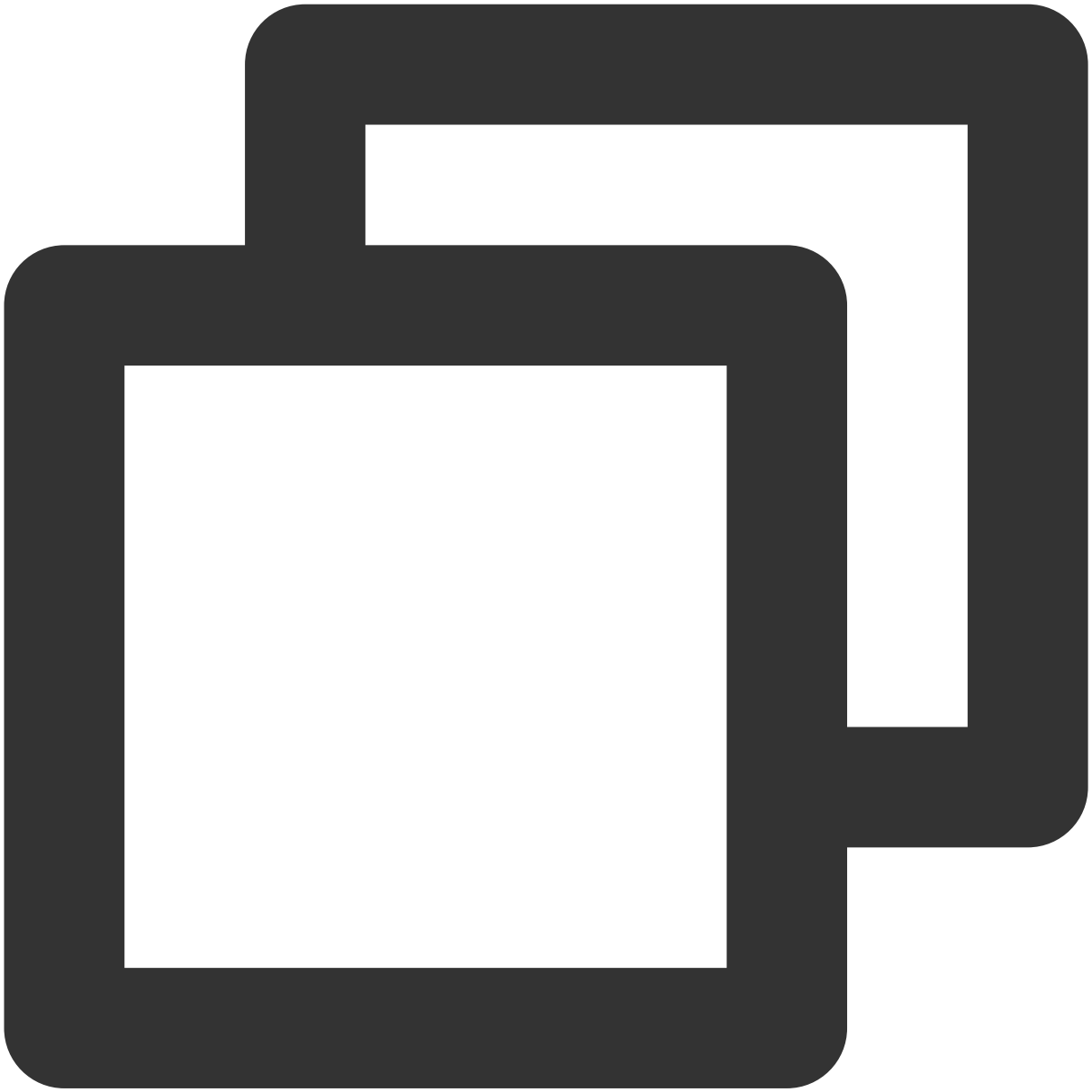
function listener， Listener function for events triggered when the user clicks the "Copy Link" button in the menu in the upper right corner。

Object res,

Attribute	Type	Description
query	string	The query string carried by the current page when opening the mini program with a short link. When used in a mini program, wx.onCopyUrl custom query should be called when entering the page, and wx.offCopyUrl should be called when exiting the page to prevent it from affecting other pages.

Sample Code

©2013-2022 Tencent Cloud. All rights reserved.



```
//Bind shared parameters
 wx.onCopyUrl(() => {
   return { query: 'a=1&b=2' }
 })


 // Unbind shared parameters
 wx.offCopyUrl()
```

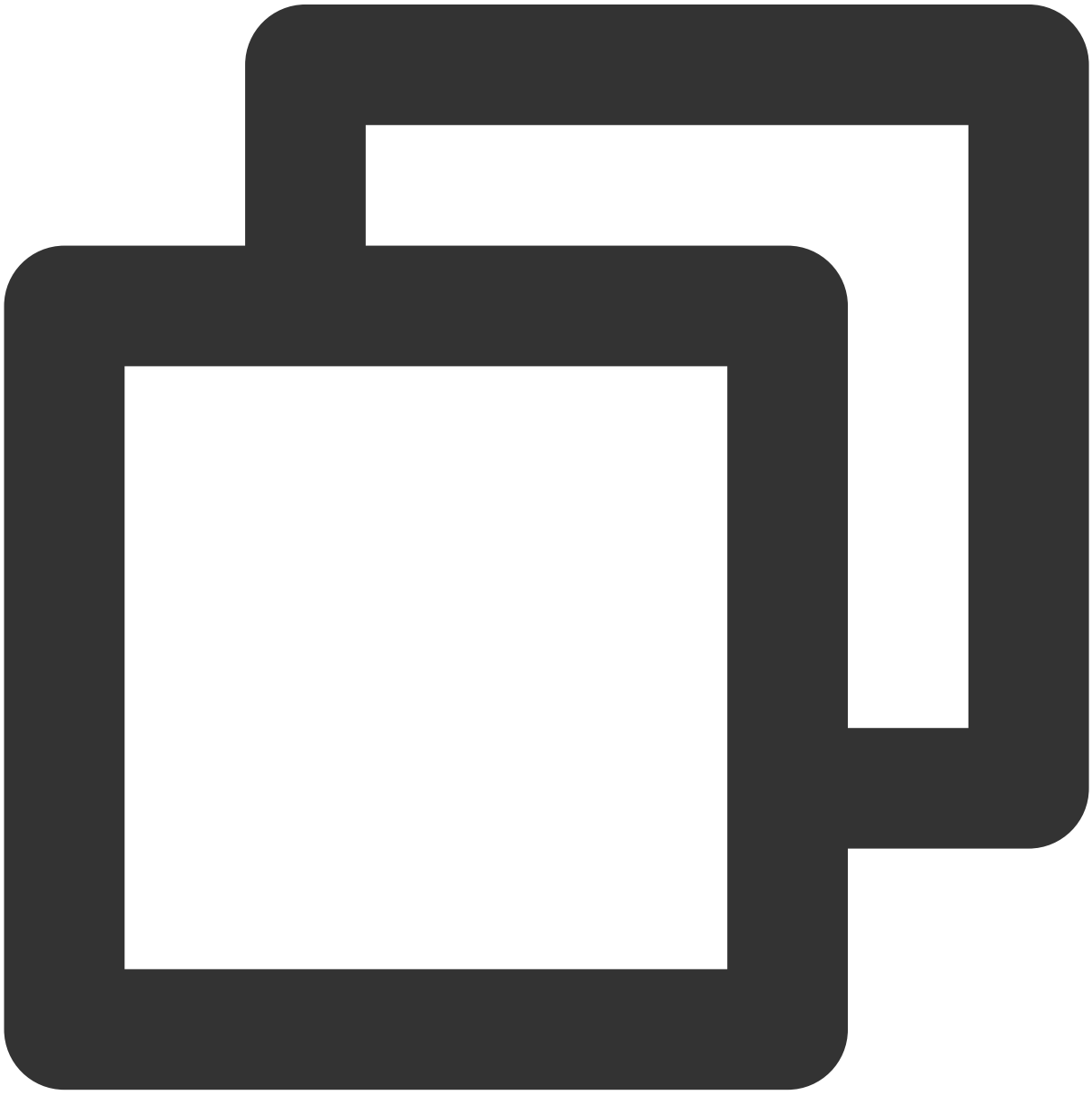
©2013-2022 Tencent Cloud. All rights reserved.

offCopyUrl

This API is used via wx.offCopyUrl()

Function description：Remove all listening functions for events triggered when the user clicks the "Copy Link" button in the menu in the upper right corner.

Sample Code



©2013-2022 Tencent Cloud. All rights reserved.

```
//Bind shared parameters
 wx.onCopyUrl(() => {
   return { query: 'a=1&b=2' }
 })


 // Unbind shared parameters
 wx.offCopyUrl()
```

©2013-2022 Tencent Cloud. All rights reserved.

Interface
Window

Last updated：2024-07-12 19:37:16

onWindowResize

This API is used via wx.onWindowResize(function listener).

Feature Description: Monitors window size change events.

Parameter and Description: function listener, the listener function for window size change events.

Attribute	Type
size	Object

Structure Attribute of Size

Structure attributes	Type	Description
windowWidth	number	The changed window width, measured in pixels.
windowHeight	number	The changed window height, measured in pixels (px).

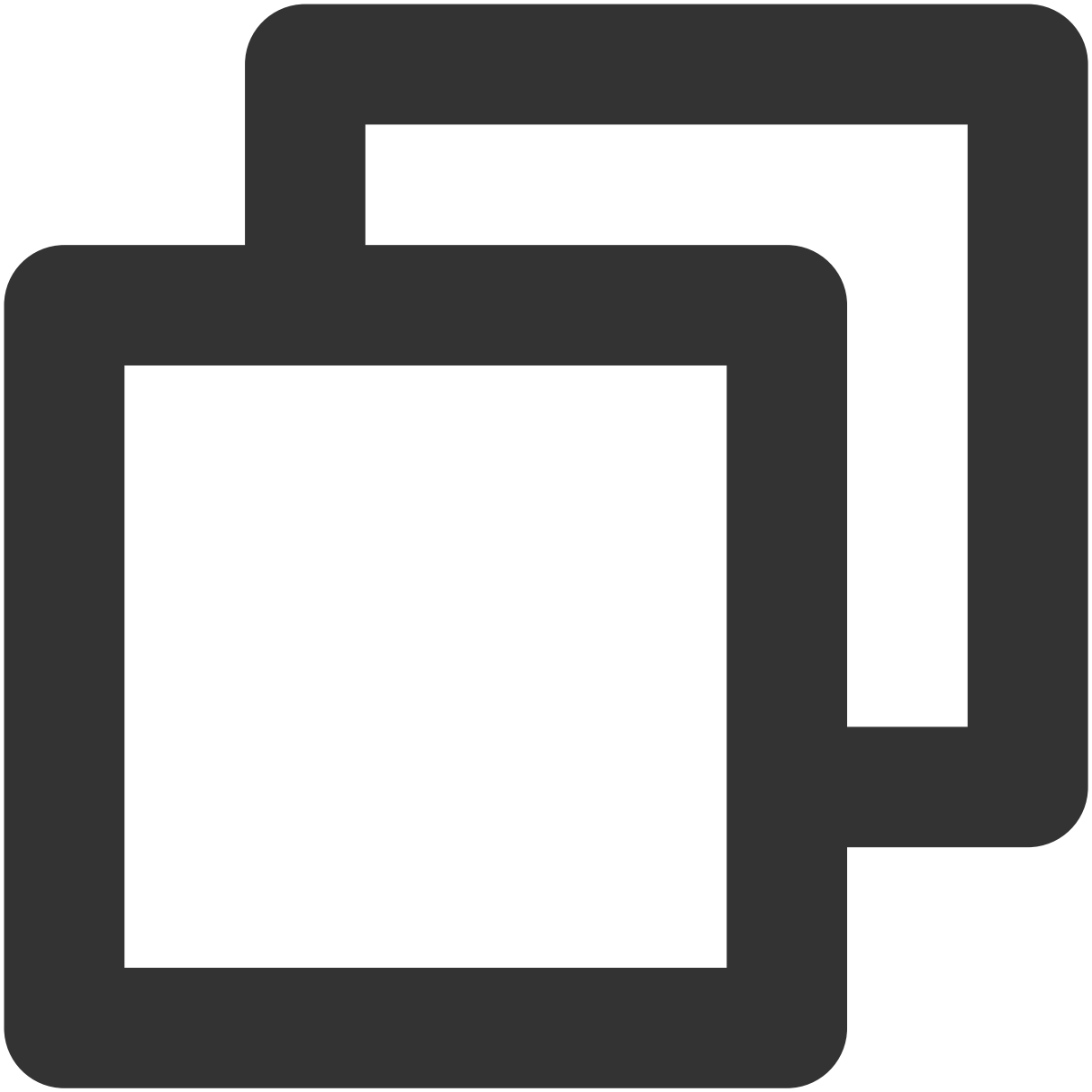
offWindowResize

This API is used via wx.offWindowResize(function listener).

Feature Description: Removes the listener function for window size change events.

Parameter and Description: function listener, the listener function passed in by function listener, onWindowResize. If this parameter is not passed, all listener functions will be removed.

©2013-2022 Tencent Cloud. All rights reserved.



```
const listener = function (res) { console.log(res) }

wx.onWindowResize(listener)
wx.offWindowResize(listener) // The same function object as the listener must be pa
```

©2013-2022 Tencent Cloud. All rights reserved.

Font

Last updated：2024-07-12 19:36:56

loadFontFace

This API is used via wx.loadFontFace(Object object).

Feature Description: Dynamically loads web fonts. The file address must be of download type, and it needs to be invoked within app.js.

Note:

The contet-type returned by the font file refers to the font, and will fail to parse if it is not in the correct format;

The font link must be https (iOS does not support http);

The link to the font must be under the same source, or enable cors support, the domain name of the mini program is servicewechat.com;

The tooltip Faild to load font can be ignored.

Parameter and Description: Object

Attribute	Type	Default value	Required	Description
global	boolean	false	No	Is it globally effective?
family	string	-	Yes	Defined font name
source	string	-	Yes	The address of the font resource. The recommended formats are TTF and WOFF, as WOFF2 may not be compatible with older versions of iOS.
desc	object	-	No	Optional font descriptors
scopes	Array	-	No	The scope of the font's application, with optional values being 'webview' or 'native'. The default is 'webview', but setting it to 'native' allows for use under Canvas 2D.
success	function	-	No	Callback Function of Successful Interface Call
fail	function	-	No	Callback Function of Successful Interface Call
complete	function	-	No	Callback function executed upon the completion of the interface invocation (both successful and unsuccessful

©2013-2022 Tencent Cloud. All rights reserved.

				invocations)

Structural value of 'desc'

Structure attributes	Type	Default value	Required	Description
style	string	normal	No	Font style, with optional values being 'normal', 'italic', or 'oblique'.
weight	string	normal	No	Font weight, with optional values being 'normal', 'bold', '100', '200', or '900'.
variant	string	normal	No	Sets the display text for small capital letters, with optional values being 'normal', 'small-caps', or 'inherit'.

Parameters for object.success callback function: Object res.

Attribute	Type	Description
status	string	Font Loading Results

Parameters for object.fail callback function: Object res

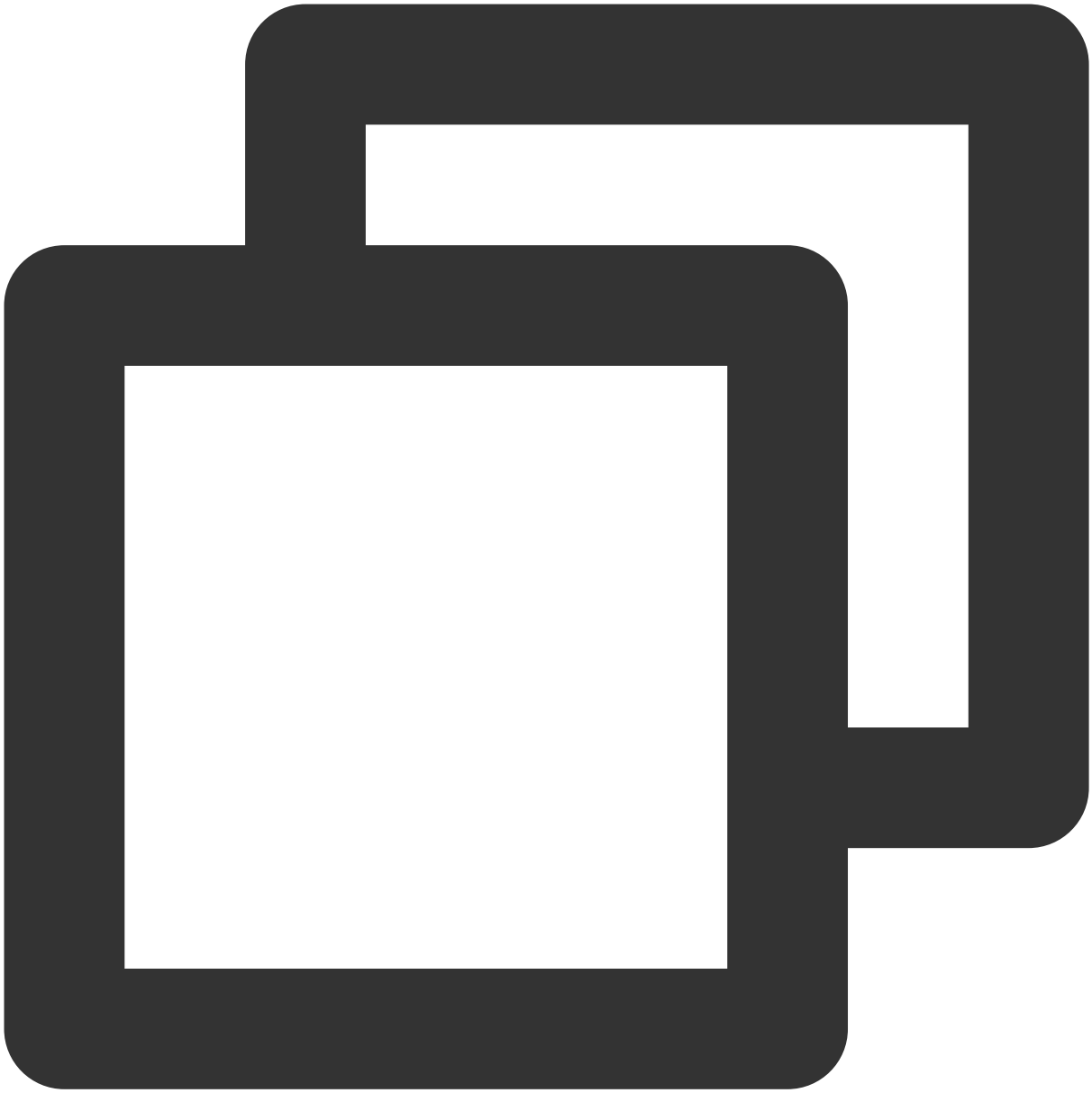
Attribute	Type	Description
status	string	Font Loading Results

object.complete Callback Function Parameters: Object res

Attribute	Type	Description
status	string	Font Loading Results

Sample Code

©2013-2022 Tencent Cloud. All rights reserved.



```
wx.loadFontFace({
  family: 'Bitstream Vera Serif Bold',
  source: 'url("https://sungd.github.io/Pacifico.ttf")',
  success: console.log
})
```

©2013-2022 Tencent Cloud. All rights reserved.

Interaction

Last updated：2024-07-12 19:36:34

hideLoading

This API is used via wx.hideLoading(Object object).

Feature Description: Hides the loading prompt box.

Parameter and Description: Object.

Attribute	Type	Default value	Required	Description
noConflict	boolean	false	No	Currently, toast and loading related interfaces can be confused with each other, this parameter can be used to disable the confusion feature.
success	function	-	No	Callback Function of Successful Interface Call
fail	function	-	No	Callback Function of Failing Interface Call
complete	function	-	No	Callback function executed upon the completion of the interface invocation (both successful and unsuccessful invocations)

hideToast

This API is used via wx.hideToast(Object object).

Feature Description: Hides the message prompt box.

Parameter and Description: Object.

Attribute	Type	Default value	Required	Description
noConflict	boolean	false	No	Currently, toast and loading related interfaces can be confused with each other, this parameter can be used

©2013-2022 Tencent Cloud. All rights reserved.

				to disable the confusion feature.
success	function	-	No	Callback Function of Successful Interface Call
fail	function	-	No	Callback Function of Failing Interface Call
complete	function	-	No	Callback function executed upon the completion of the interface invocation (both successful and unsuccessful invocations)

showActionSheet

This API is used via wx.showActionSheet(Object object).

Note:

For Android versions earlier than V6.7.2, upon clicking cancel or the mask layer, the callback fails, with errMsg being "fail cancel".

For Android versions 6.7.2 and later, as well as iOS, clicking on the mask layer will not close the modal popup.

Therefore, it is advisable to avoid implementing business logic within the 'cancel' branch.

Feature Description: Displays the action menu.

Parameter and Description: Object.

Attribute	Type	Default value	Required	Description
itemList	array.<string>	-	Yes	The array of button texts, with a maximum length of 6.
itemColor	string	#000000	No	The color of the button text.
success	function	-	No	Callback Function of Successful Interface Call
fail	function	-	No	Callback Function of Failing Interface Call
complete	function	-	No	Callback function executed upon the completion of the interface invocation (both successful and unsuccessful invocations)

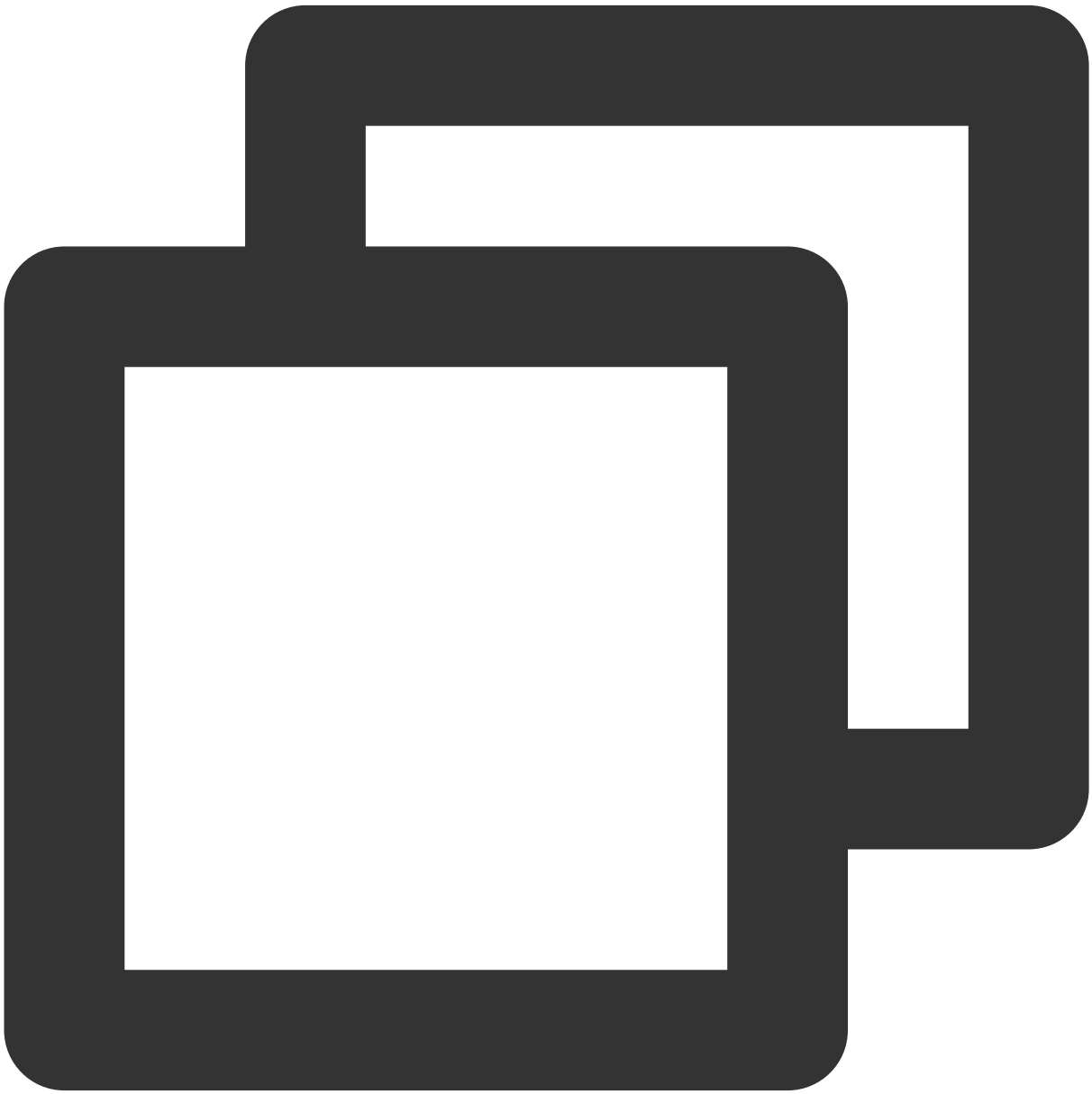
Parameters for object.success callback function: Object res.

Attribute	Type	Description

©2013-2022 Tencent Cloud. All rights reserved.

tapIndex	number	The button number clicked by the user, in descending order, starting from 0.

Sample Code



```
wx.showActionSheet({
  itemList: ['A', 'B', 'C'],
  success (res) {
    console.log(res.tapIndex)
  },
  fail (res) {
    console.log(res.errMsg)
```

©2013-2022 Tencent Cloud. All rights reserved.

```
    }
})
```

showLoading

This API is used via wx.showLoading(Object object).

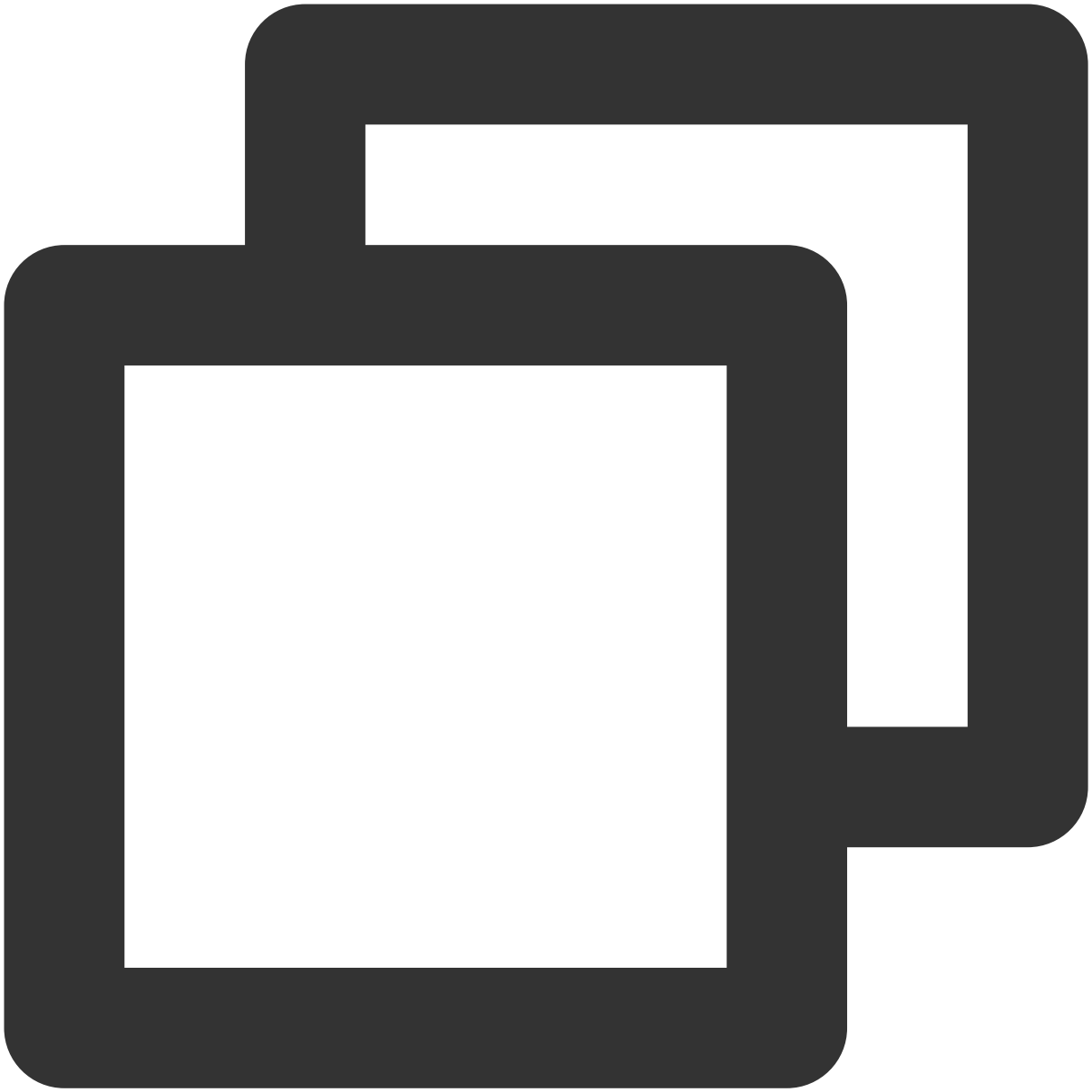
Feature Description: Displays a loading prompt box. It requires an active call to wx.hideLoading to close the prompt box.

Parameter and Description: Object.

Attribute	Type	Default value	Required	Description
title	string	-	Yes	The content of the prompt.
mask	boolean	false	No	Whether to display a transparent mask layer to prevent touch penetration.
success	function	-	No	Callback Function of Successful Interface Call
fail	function	-	No	Callback Function of Failing Interface Call
complete	function	-	No	Callback function executed upon the completion of the interface invocation (both successful and unsuccessful invocations)

Sample Code

©2013-2022 Tencent Cloud. All rights reserved.



```
wx.showLoading({
  title: 'Loading',
})

setTimeout(function () {
  wx.hideLoading()
}, 2000)
```

Note:

Only one of wx.showLoading and wx.showToast can be displayed at a time.

©2013-2022 Tencent Cloud. All rights reserved.

wx.showLoading should be used in conjunction with wx.hideLoading.

showModal

This API is used via wx.showModal(Object object).

Note:

For Android versions of V6.7.2 and earlier, upon clicking cancel or the mask layer, the callback fails, with errMsg being "fail cancel".

For Android versions 6.7.2 and later, as well as iOS, clicking on the mask layer will not close the modal popup. Therefore, it is advisable to avoid implementing business logic within the **cancel** branch.

Feature Description: Displays a modal dialog box.

Parameter and Description: Object.

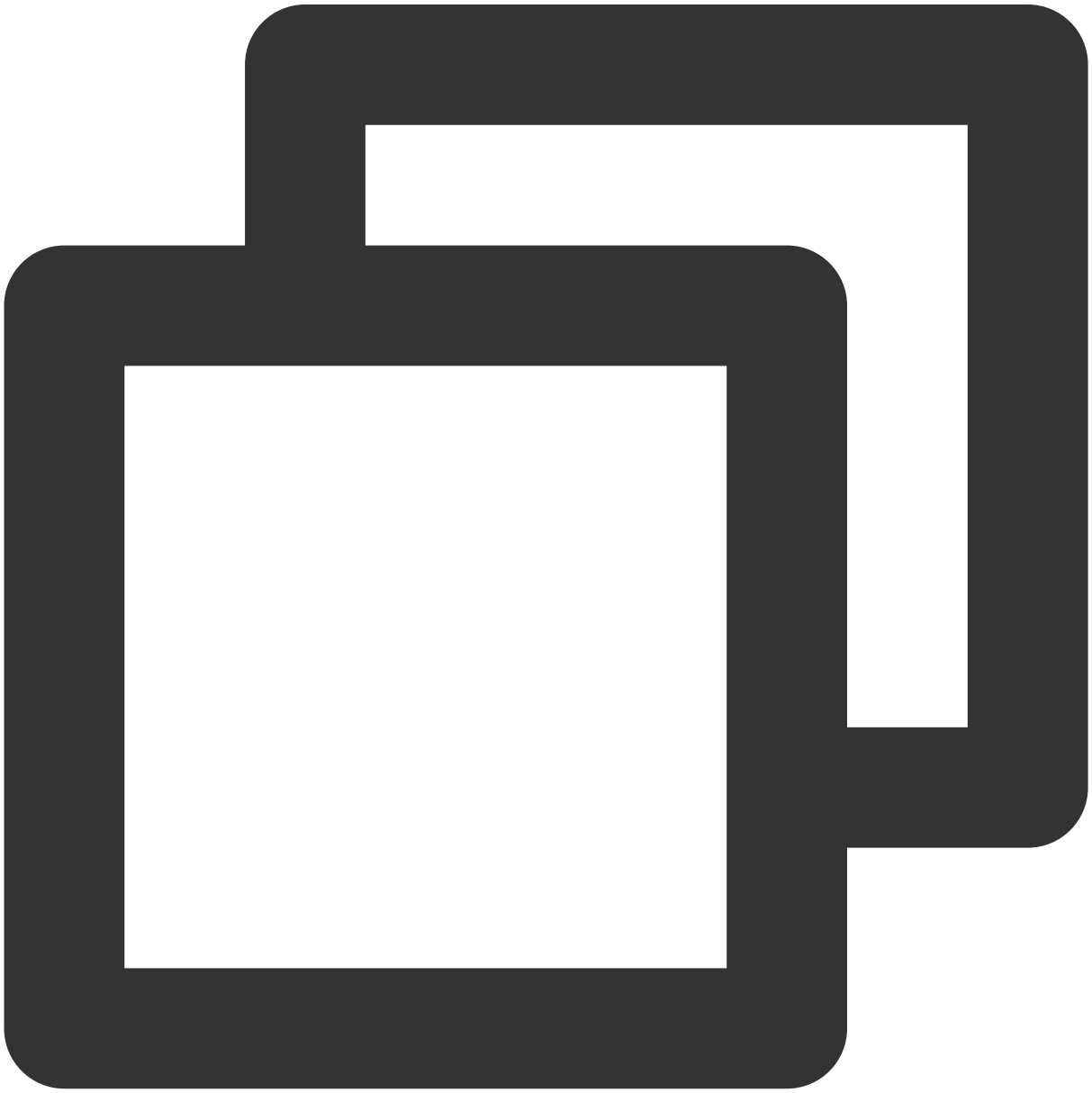
Attribute	Type	Default value	Required	Description
title	string	-	No	The title of the prompt.
content	string	-	No	The content of the prompt.
showCancel	boolean	true	No	Whether to display the "Cancel" button.
cancelText	string	Cancel	No	Text for the "Cancel" button, up to a maximum of 4 characters.
cancelColor	string	#000000	No	Text color of the "Cancel" button, which must be a hexadecimal color string.
confirmText	string	Confirm	No	Text for the "Confirm" button, limited to a maximum of 4 characters.
confirmColor	string	#576B95	No	Text color of the "Confirm" button, which must be a hexadecimal color string.
success	function	-	No	Callback Function of Successful Interface Call
fail	function	-	No	Callback Function of Failing Interface Call
complete	function	-	No	Callback function executed upon the completion of the interface invocation (both successful and unsuccessful invocations)

©2013-2022 Tencent Cloud. All rights reserved.

Parameters for object.success callback function: Object res.

Attribute	Type	Description
confirm	boolean	When set to true, it indicates that the user has clicked the "Confirm" button.
cancel	boolean	When set to true, it indicates that the user has clicked the "Cancel" button (used in the Android system to distinguish between clicking to close the mask layer or clicking the "Cancel" button to close).

Sample Code



©2013-2022 Tencent Cloud. All rights reserved.

```
wx.showModal({
  title: 'Prompt',
  content: 'This is a modal dialog box',
  success (res) {
    if (res.confirm) {
      console.log(User clicked "Confirm")
    } else if (res.cancel) {
      console.log(User clicked "Cancel")
    }
  }
})
```

showToast

This API is used via wx.showToast(Object object).

Feature Description: Displays a modal dialog box.

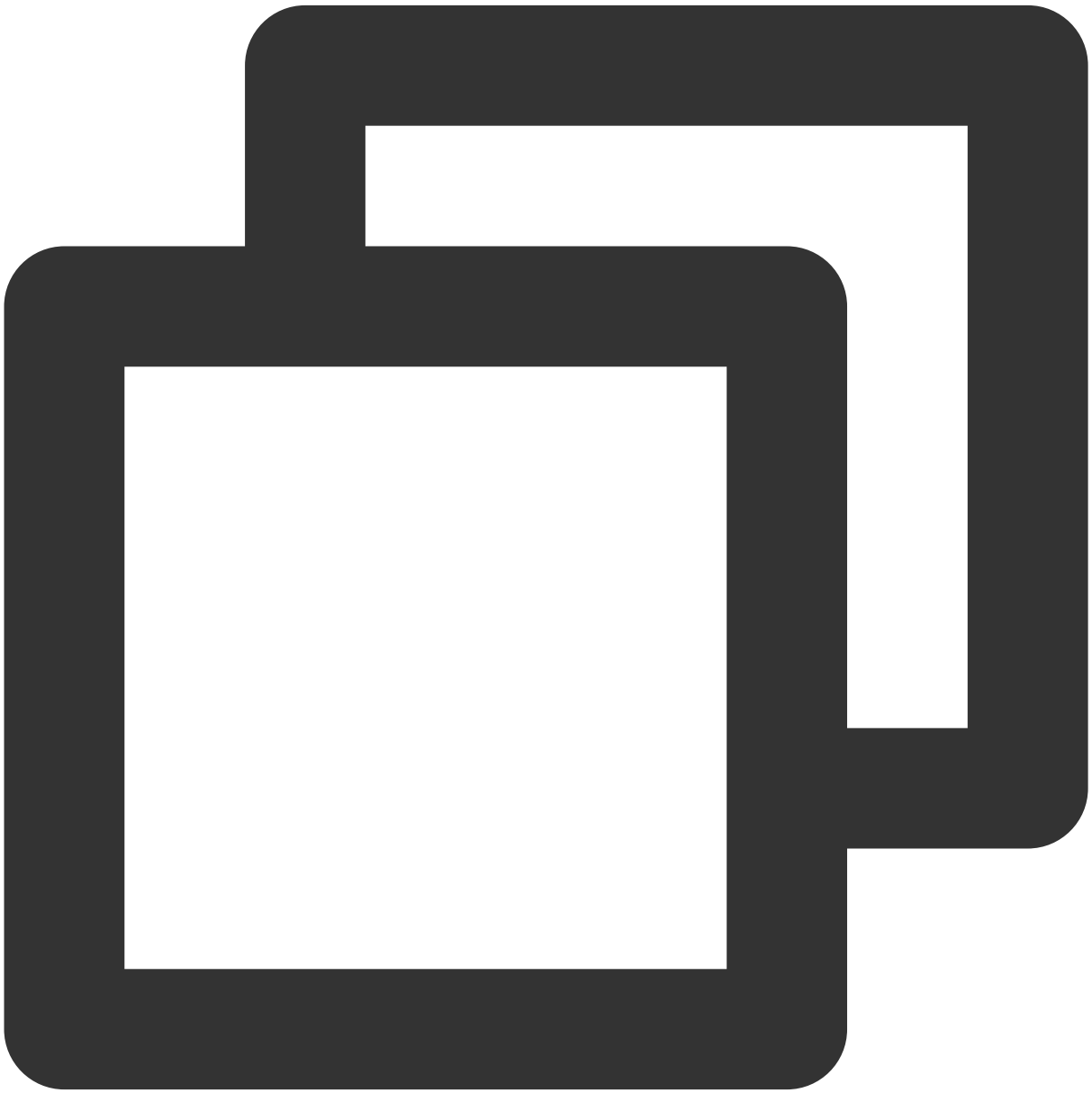
Parameter and Description: Object.

Attribute	Type	Default value	Required	Description
title	string	-	Yes	The content of the prompt.
icon	string	success	No	Icon, whose valid values are: success: Displays the failure icon, when the title text can display up to 7 Chinese characters in length. error: Displays the failure icon, when the title text can display up to 7 Chinese characters in length. loading: Displays the loading icon, when the title text can display up to 7 Chinese characters in length. none: Not to display icons, when the title text can display up to two lines.
image	string	-	No	The local path of the custom icon, with the priority of the image higher than that of the icon.
duration	number	1500	No	The delay duration of the prompt.
mask	boolean	false	No	Whether to display a transparent mask layer to prevent touch penetration.

©2013-2022 Tencent Cloud. All rights reserved.

success	function	-	No	Callback Function of Successful Interface Call
fail	function	-	No	Callback Function of Failing Interface Call
complete	function	-	No	Callback function executed upon the completion of the interface invocation (both successful and unsuccessful invocations)

Sample Code



```
wx.showToast({
```

©2013-2022 Tencent Cloud. All rights reserved.

```
  title: 'Success',
  icon: 'success',
  duration: 2000
})
```

Note:

Only one of wx.showLoading and wx.showToast can be displayed at a time.

wx.showLoading should be used in conjunction with wx.hideLoading.

enableAlertBeforeUnload

This API is used via wx.enableAlertBeforeUnload(Object object).

Feature Description: Enables the dialog box for asking about the return of the mini program page.

Popup Conditions:

When the user is on a non-homepage or bottom-level page within the mini program.

The return on the official navigation bar.

Custom return key in full-screen mode.

When the back key on the Android system is pressed.

Note:

No interception is made during gesture slide return.

Under any circumstances, this feature should not obstruct the user's action of exiting the mini program.

Parameter and Description: Object.

Attribute	Type	Default value	Required	Description
message	string	-	Yes	Content of the inquiry dialog box.
success	function	-	No	Callback Function of Successful Interface Call
fail	function	-	No	Callback Function of Failing Interface Call
complete	function	-	No	Callback function executed upon the completion of the interface invocation (both successful and unsuccessful invocations)

disableAlertBeforeUnload

©2013-2022 Tencent Cloud. All rights reserved.

This API is used via wx.disableAlertBeforeUnload(Object object).

Feature Description: Disables the dialog box for asking about the return of the mini program page.

Parameter and Description: Object.

Attribute	Type	Default value	Required	Description
success	function	-	No	Callback Function of Successful Interface Call
fail	function	-	No	Callback Function of Failing Interface Call
complete	function	-	No	Callback function executed upon the completion of the interface invocation (both successful and unsuccessful invocations)

©2013-2022 Tencent Cloud. All rights reserved.

Navigation Bar

Last updated：2024-07-12 19:36:04

setNavigationBarTitle

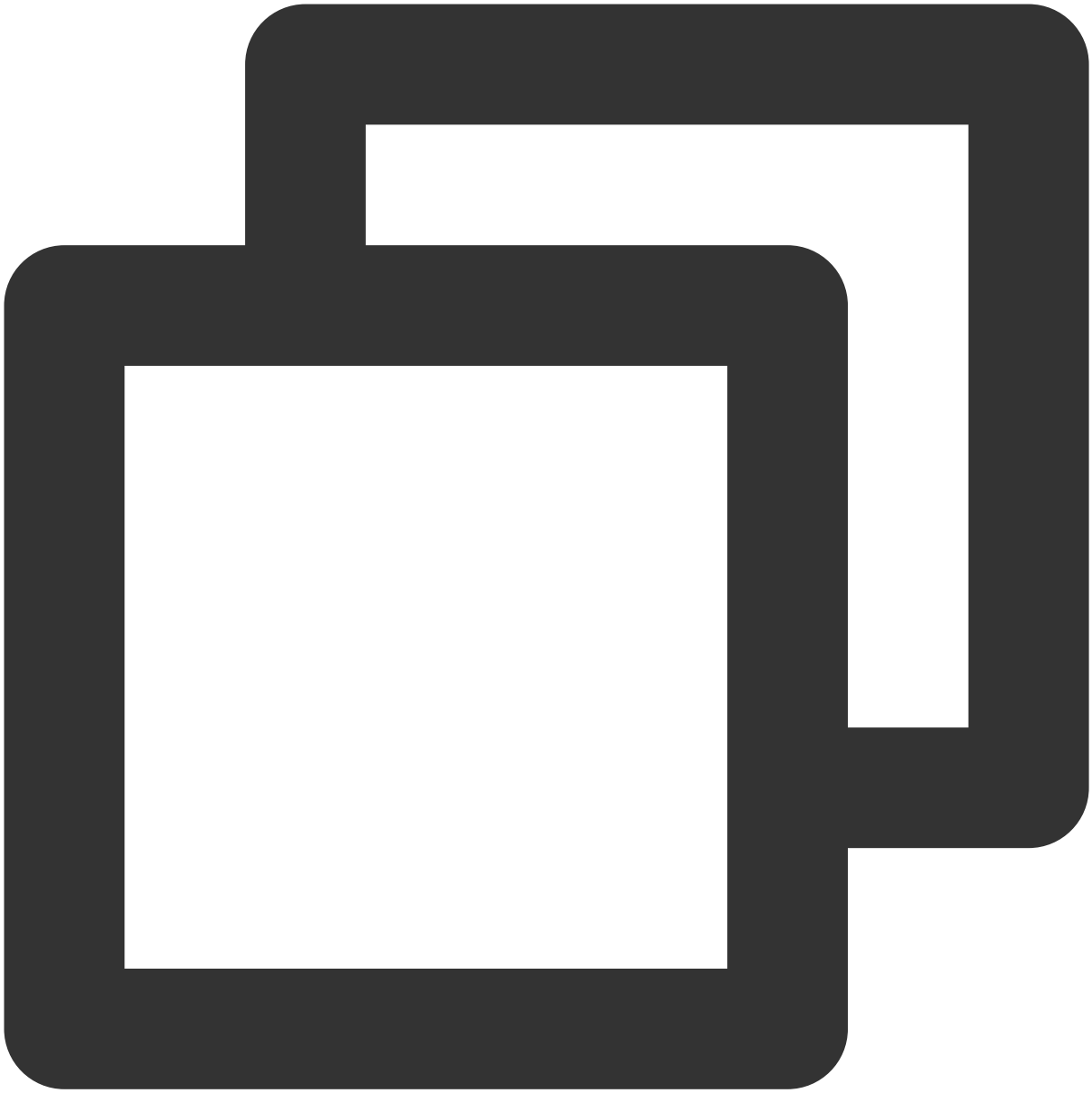
This API is used via wx.setNavigationBarTitle(Object object).

Feature Description: Dynamically sets the title of the current page.

Parameter and Description: Object.

Attribute	Type	Default value	Required	Description
title	string	-	Yes	Page title
success	function	-	No	Callback Function of Successful Interface Call
fail	function	-	No	Callback Function of Failing Interface Call
complete	function	-	No	Callback function executed upon the completion of the interface invocation (both successful and unsuccessful invocations)

©2013-2022 Tencent Cloud. All rights reserved.



```
wx.setNavigationBarTitle({
  title: 'Current Page'
})
```

setNavigationBarColor

This API is used via wx.setNavigationBarColor(Object object).

©2013-2022 Tencent Cloud. All rights reserved.

Feature Description: Set the color scheme for the page navigation bar.

Parameter and Description: Object.

Attribute	Type	Default value	Required	Description
frontColor	string	-	Yes	Foreground color value, including the color of buttons, title, and status bar, only supporting #ffffff and #000000.
backgroundColor	string	-	Yes	Background color value, with valid values being hexadecimal colors.
animation	object	-	No	Animation Effect
success	function	-	No	Callback Function of Successful Interface Call
fail	function	-	No	Callback Function of Failing Interface Call
complete	function	-	No	Callback function executed upon the completion of the interface invocation (both successful and unsuccessful invocations)

Structure attributes of object.animation

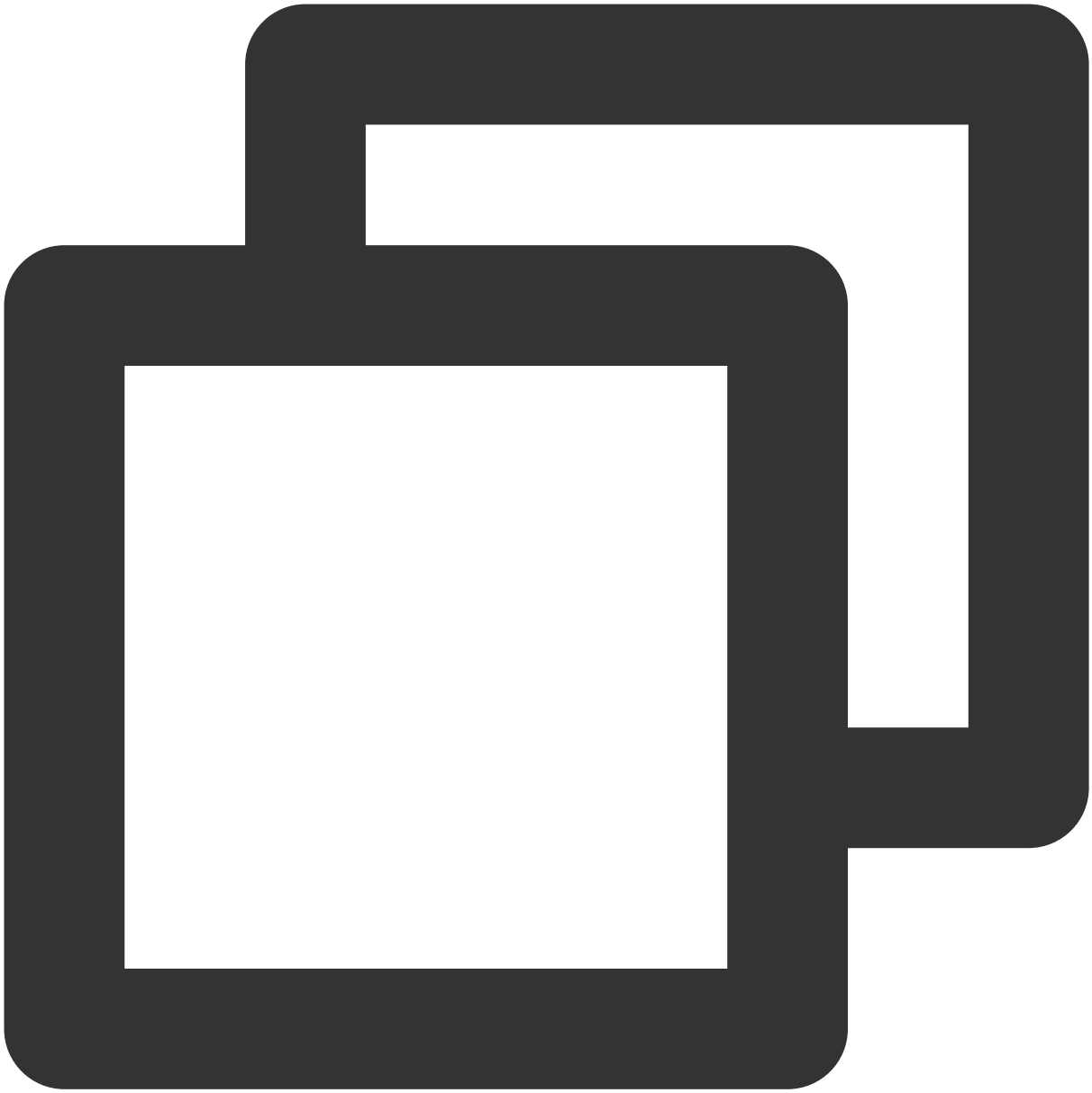
Attribute	Type	Default value	Required	Description
duration	number	0	No	Animation transition duration, measured in milliseconds.
timingFunc	string	'linear'	No	Animation transition method

Valid values for object.animation.timingFunc

Value	Description
'linear'	The animation maintains a consistent speed from beginning to end.
'easeIn'	The animation starts at a slow pace.
'easeOut'	The animation concludes at a slow pace.
'easeInOut'	The animation starts and concludes at a slow pace.

Sample Code

©2013-2022 Tencent Cloud. All rights reserved.



```
wx.setNavigationBarColor({
  frontColor: '#ffffff',
  backgroundColor: '#ff0000',
  animation: {
    duration: 400,
    timingFunc: 'easeIn'
  }
})
```

©2013-2022 Tencent Cloud. All rights reserved.

hideHomeButton

This API is used via wx.hideHomeButton(Object object).

Feature Description: Hide home button. When the bottom page of the mini program opened by the user is not the home page, the sharing panel will show the "Back to Home" button by default, and the developer can call hideHomeButton in the onShow of the page to hide it.

Parameter and Description: Object.

Attribute	Type	Default value	Required	Description
success	function	-	No	Callback Function of Successful Interface Call
fail	function	-	No	Callback Function of Failing Interface Call
complete	function	-	No	Callback function executed upon the completion of the interface invocation (both successful and unsuccessful invocations)

showNavigationBarLoading

This API is used via wx.showNavigationBarLoading(Object object).

Feature Description: Displays a navigation bar loading animation on the current page.

Parameter and Description: Object.

Attribute	Type	Default value	Required	Description
success	function	-	No	Callback Function of Successful Interface Call
fail	function	-	No	Callback Function of Failing Interface Call
complete	function	-	No	Callback function executed upon the completion of the interface invocation (both successful and unsuccessful invocations)

©2013-2022 Tencent Cloud. All rights reserved.

hideNavigationBarLoading

This API is used via wx.hideNavigationBarLoading(Object object).

Feature Description: Hides the navigation bar loading animation on the current page.

Parameter and Description: Object.

Attribute	Type	Default value	Required	Description
success	function	-	No	Callback Function of Successful Interface Call
fail	function	-	No	Callback Function of Failing Interface Call
complete	function	-	No	Callback function executed upon the completion of the interface invocation (both successful and unsuccessful invocations)

©2013-2022 Tencent Cloud. All rights reserved.

Background

Last updated：2024-07-12 19:35:43

setBackgroundColor

This API is used via wx.setBackgroundColor(Object object).

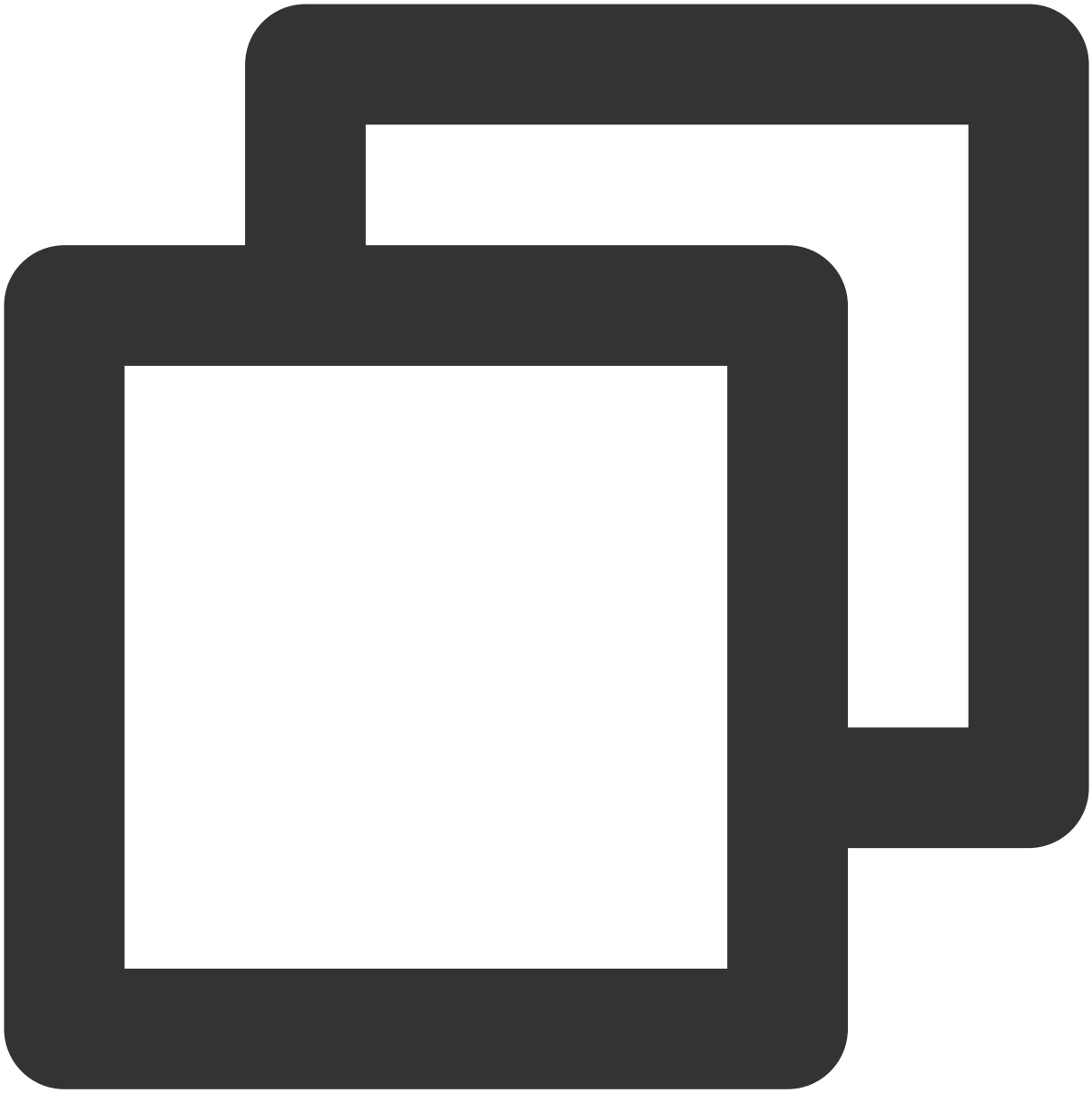
Feature Description: Dynamically sets the background color of the window.

Parameter and Description: Object.

Attribute	Type	Default value	Required	Note
backgroundColor	string	-	No	Background color value, which must be a hexadecimal color value.
backgroundColorTop	string	-	No	Background color of the top window, which must be a hexadecimal color value. This feature is supported only on iOS.
backgroundColorBottom	string	-	No	Background color of the bottom window, which must be a hexadecimal color value. This feature is supported only on iOS.
success	function	-	No	Callback Function of Successful Interface Call
fail	function	-	No	Callback Function of Failing Interface Call
complete	function	-	No	Callback function executed upon the completion of the interface invocation (both successful and unsuccessful invocations)

Sample Code

©2013-2022 Tencent Cloud. All rights reserved.



```
wx.setBackgroundColor({
  backgroundColor: '#ffffff', // The background color of the window is white.
})

wx.setBackgroundColor({
  backgroundColorTop: '#ffffff', // The background color of the top window is white
  backgroundColorBottom: '#ffffff', // The background color of the bottom window is
})
```

©2013-2022 Tencent Cloud. All rights reserved.

setBackgroundTextStyle

This API is used via wx.setBackgroundTextStyle(Object object).

Feature Description: Dynamically sets the style of the dropdown background font and loading image.

Parameter and Description: Object

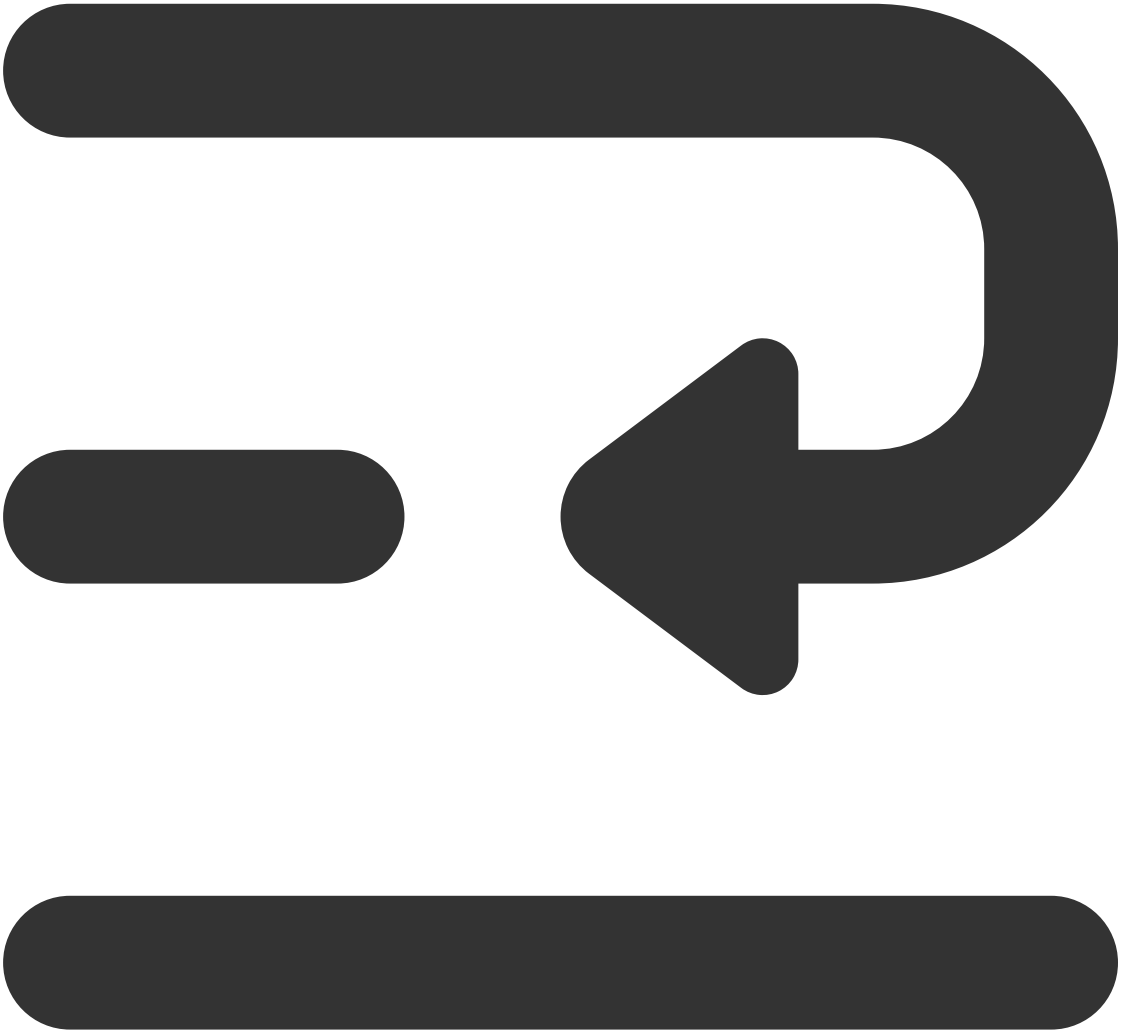
Attribute	Type	Default value	Required	Note
textStyle	string	-	Yes	The style of the dropdown background text and loading image.
success	function	-	No	Callback Function of Successful Interface Call
fail	function	-	No	Callback Function of Failing Interface Call
complete	function	-	No	Callback function executed upon the completion of the interface invocation (both successful and unsuccessful invocations)

Valid values of object.textStyle

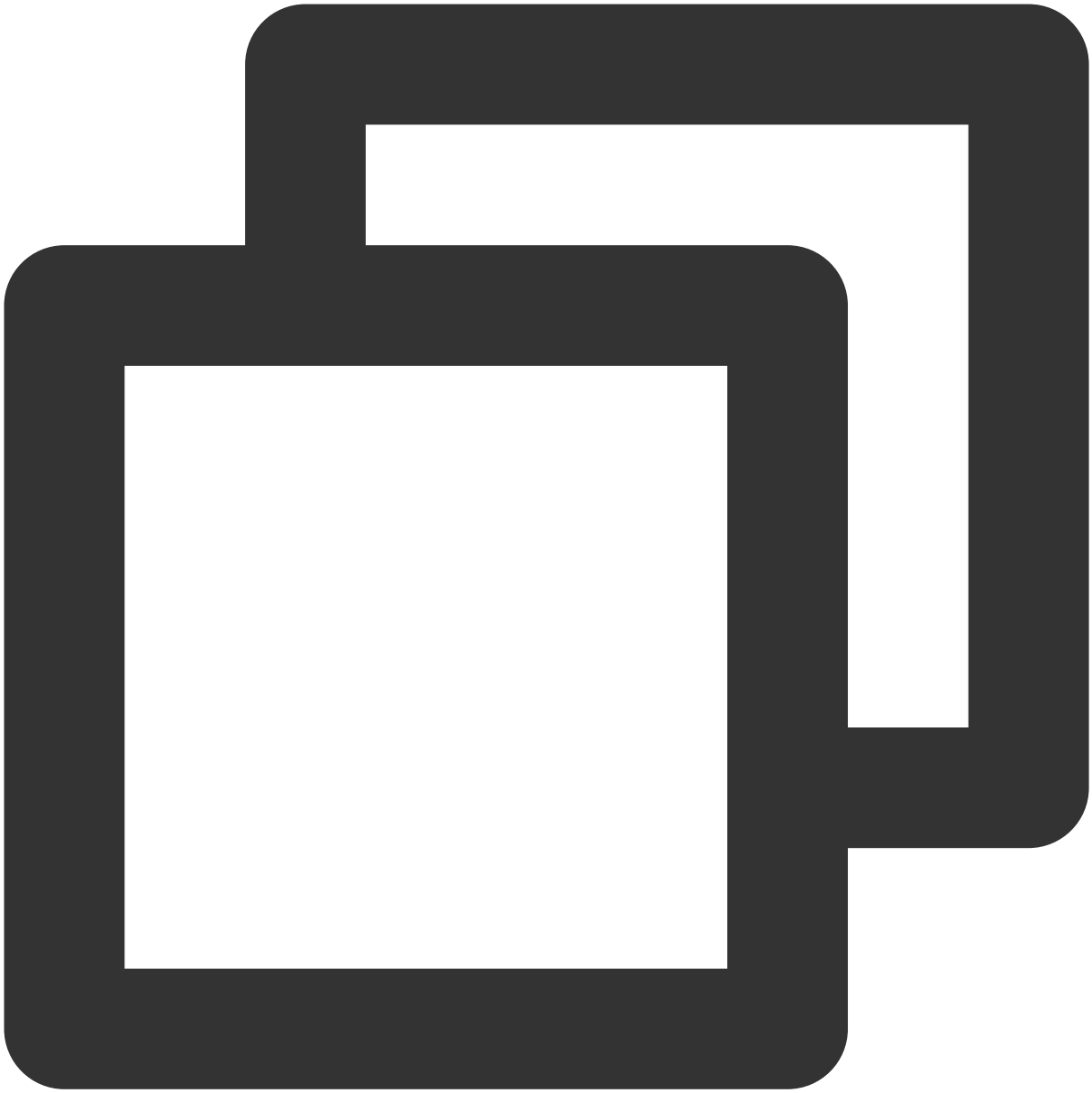
Value	Note
dark	Dark mode
light	Light mode

Sample Code

©2013-2022 Tencent Cloud. All rights reserved.



©2013-2022 Tencent Cloud. All rights reserved.



```
wx.setBackgroundTextStyle({
    textStyle: 'dark' // The dropdown background text and loading icon styles are s
})
```

©2013-2022 Tencent Cloud. All rights reserved.

TabBar

Last updated：2024-07-12 19:35:22

showTabBar

The method of utilizing this API is wx.showTabBar(Object object).

Function Description: Unveils the tabBar.

Parameters and Description: Object object.

Attributes	Types	Default value	Required	Note
animation	boolean	false	Not required	Is animation effect necessary?
success	function	-	Not required	Callback function for successful interface invocation
fail	function	-	Not required	Callback function for unsuccessful interface invocation
complete	function	-	Not required	Callback function executed upon the completion of the interface invocation (both successful and unsuccessful invocations)

hideTabBar

The usage method for this API is wx.hideTabBar(Object object)

Function Description: Conceals the tabBar.

Parameters and Description: Object object.

Attributes	Types	Default value	Required	Note
animation	boolean	false	Not	Is animation effect necessary?

©2013-2022 Tencent Cloud. All rights reserved.

			required	
success	function	-	Not required	Callback function for successful interface invocation
fail	function	-	Not required	Callback function for unsuccessful interface invocation
complete	function	-	Not required	Callback function executed upon the completion of the interface invocation (both successful and unsuccessful invocations)

setTabBarStyle

The usage method for this API is wx.setTabBarStyle(Object object)

Function Description: Dynamically configures the overall style of the tabBar.

Parameters and Description: Object object.

Attributes	Types	Default value	Required	Note
color	string	-	Not required	Default text color on the tab, HexColor
selectedColor	string	-	Not required	Text color on the tab when selected, HexColor
backgroundColor	string	-	Not required	Background color of the tab, HexColor
borderStyle	string	-	Not required	Color of the border on the tab, only supports black/white
success	function	-	Not required	Callback function for successful interface invocation
fail	function	-	Not required	Callback function for unsuccessful interface invocation
complete	function	-	Not required	Callback function executed upon the completion of the interface invocation (both successful and unsuccessful invocations)

©2013-2022 Tencent Cloud. All rights reserved.

setTabBarItem

The usage method for this API is wx.setTabBarItem(Object object)

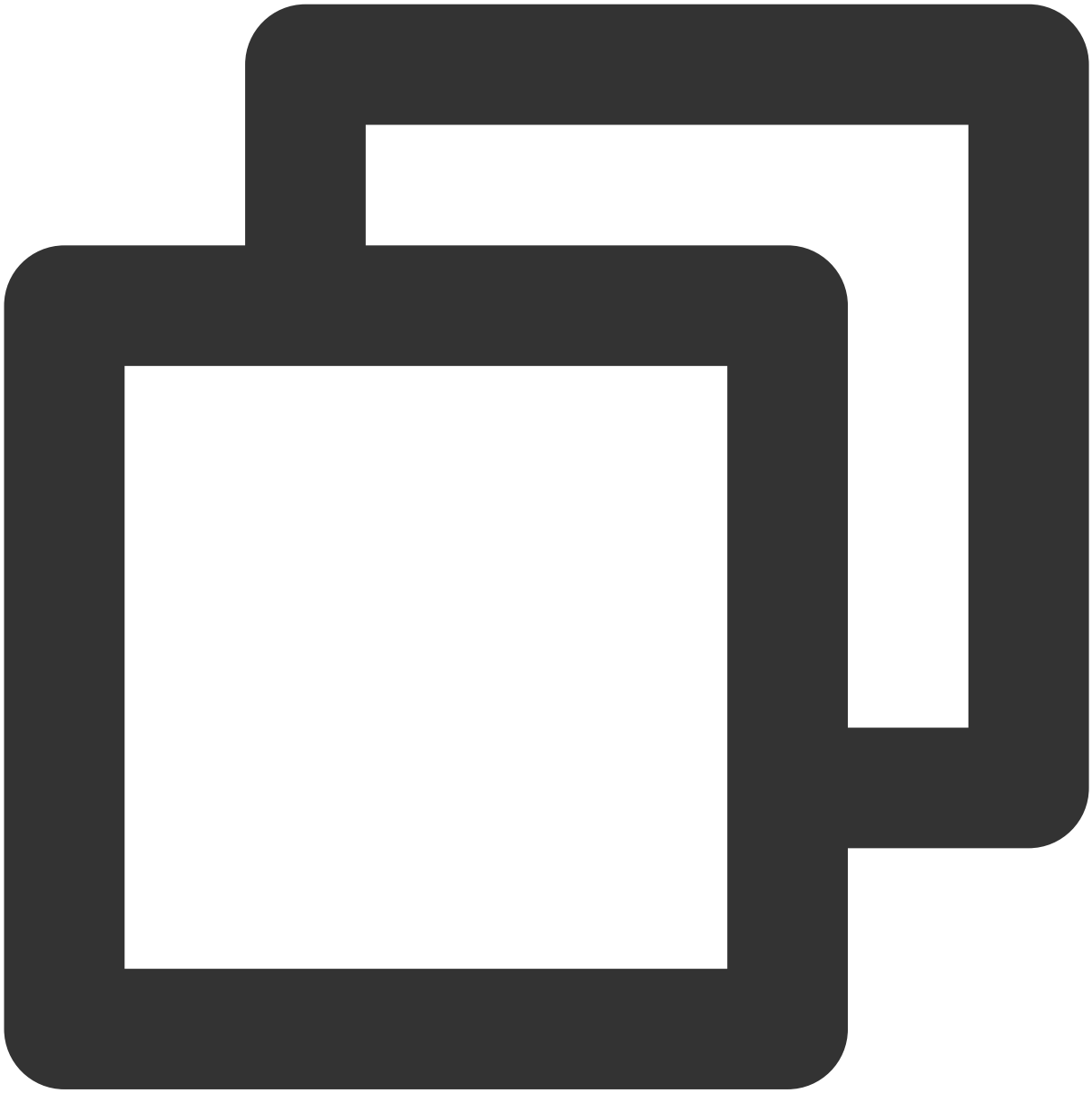
Function Description: Dynamically sets the content of a specific tabBar item.

Parameters and Description: Object object.

Attributes	Types	Default value	Required	Note
index	number	-	Supported	Which item of the tabBar, counting from the left
text	string	-	Not required	Text on the tab button
iconPath	string	-	Not required	Image path, the icon size is limited to 40kb, with a recommended dimension of 81px * 81px. This parameter is invalid when the position is set to 'top'.
selectedIconPath	string	-	Not required	Path of the image when selected, the icon size is limited to 40kb, with a recommended dimension of 81px * 81px. This parameter is invalid when the position is set to 'top'.
success	function	-	Not required	Callback function for successful interface invocation
fail	function	-	Not required	Callback function for unsuccessful interface invocation
complete	function	-	Not required	Callback function executed upon the completion of the interface invocation (both successful and unsuccessful invocations)

Example Code

©2013-2022 Tencent Cloud. All rights reserved.



```
wx.setTabBarItem({
  index: 0,
  text: 'text',
  iconPath: '/path/to/iconPath',
  selectedIconPath: '/path/to/selectedIconPath'
})
```

showTabBarRedDot

©2013-2022 Tencent Cloud. All rights reserved.

The usage of this API is wx.showTabBarRedDot(Object object)

Function Description: Display the red dot at the top right corner of a tabBar item.

Parameters and Description: Object object.

Attributes	Types	Default value	Required	Note
index	number	-	Supported	Which item of the tabBar, counting from the left
success	function	-	Not required	Callback function for successful interface invocation
fail	function	-	Not required	Callback function for unsuccessful interface invocation
complete	function	-	Not required	Callback function executed upon the completion of the interface invocation (both successful and unsuccessful invocations)

hideTabBarRedDot

The usage of this API is wx.hideTabBarRedDot(Object object)

Function Description: Hide the red dot at the top right corner of a tabBar item.

Parameters and Description: Object object.

Attributes	Types	Default value	Required	Note
index	number	-	Supported	Which item of the tabBar, counting from the left
success	function	-	Not required	Callback function for successful interface invocation
fail	function	-	Not required	Callback function for unsuccessful interface invocation
complete	function	-	Not required	Callback function executed upon the completion of the interface invocation (both successful and unsuccessful invocations)

©2013-2022 Tencent Cloud. All rights reserved.

setTabBarBadge

The usage of this API is wx.setTabBarBadge(Object object)

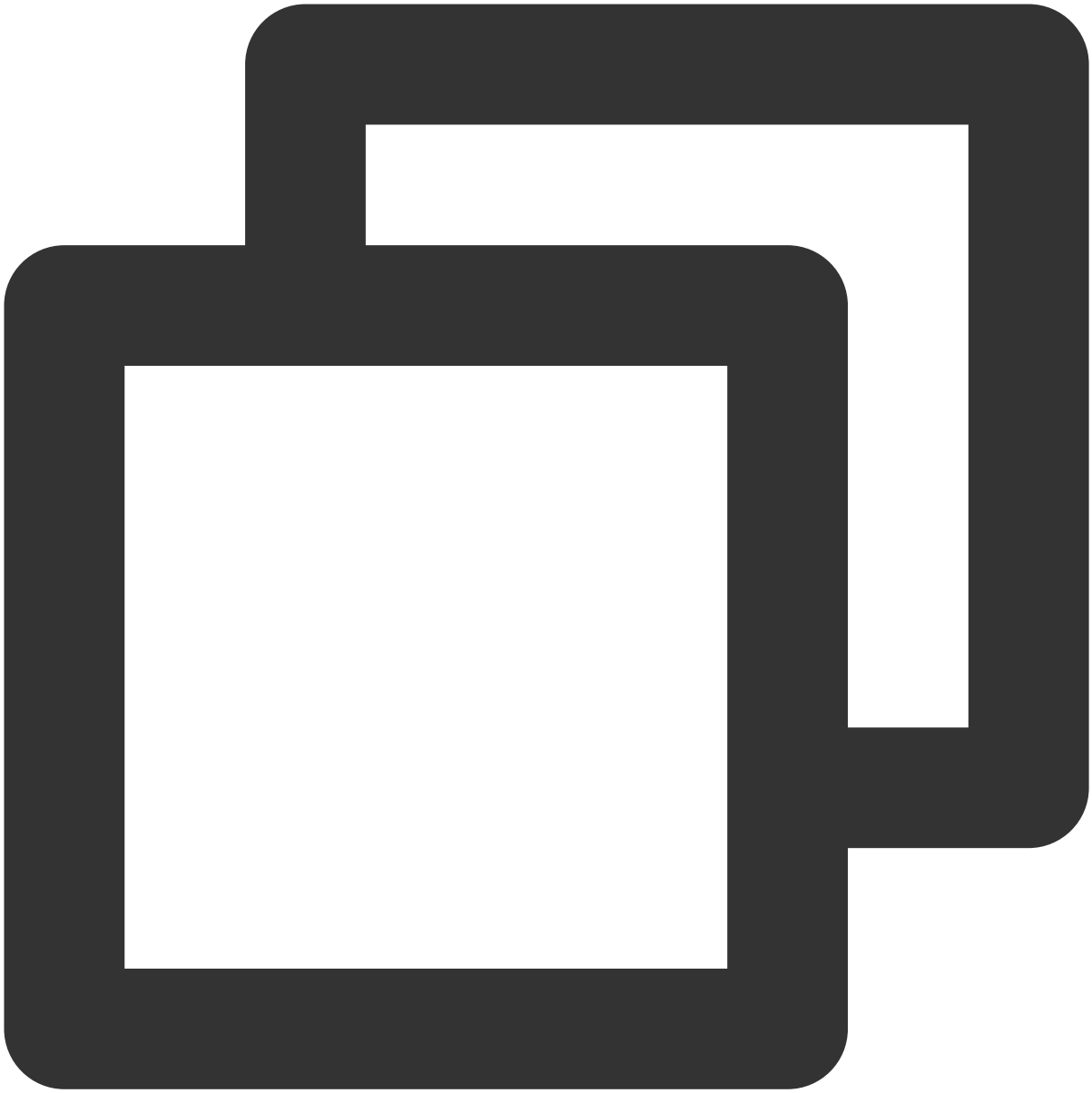
Function Description: Add text to the top right corner of a tabBar item.

Parameters and Description: Object object.

Attributes	Types	Default value	Required	Note
index	number	-	Supported	Which item of the tabBar, counting from the left
text	string	-	Supported	The displayed text truncates to '...' if it exceeds 4 characters.
success	function	-	Not required	Callback function for successful interface invocation
fail	function	-	Not required	Callback function for unsuccessful interface invocation
complete	function	-	Not required	Callback function executed upon the completion of the interface invocation (both successful and unsuccessful invocations)

Example Code

©2013-2022 Tencent Cloud. All rights reserved.



```
wx.setTabBarBadge({
  index: 0,
  text: '1'
})
```

removeTabBarBadge

©2013-2022 Tencent Cloud. All rights reserved.

The usage of this API is wx.removeTabBarBadge(Object object)

Function Description: Remove the text from the top right corner of a tabBar item.

Parameters and Description: Object object.

Attributes	Types	Default value	Required	Note
index	number	-	Supported	Which item of the tabBar, counting from the left
success	function	-	Not required	Callback function for successful interface invocation
fail	function	-	Not required	Callback function for unsuccessful interface invocation
complete	function	-	Not required	Callback function executed upon the completion of the interface invocation (both successful and unsuccessful invocations)

©2013-2022 Tencent Cloud. All rights reserved.

Pull-down refresh

Last updated：2024-07-12 19:34:55

startPullDownRefresh

This API is used via wx.startPullDownRefresh(Object object).

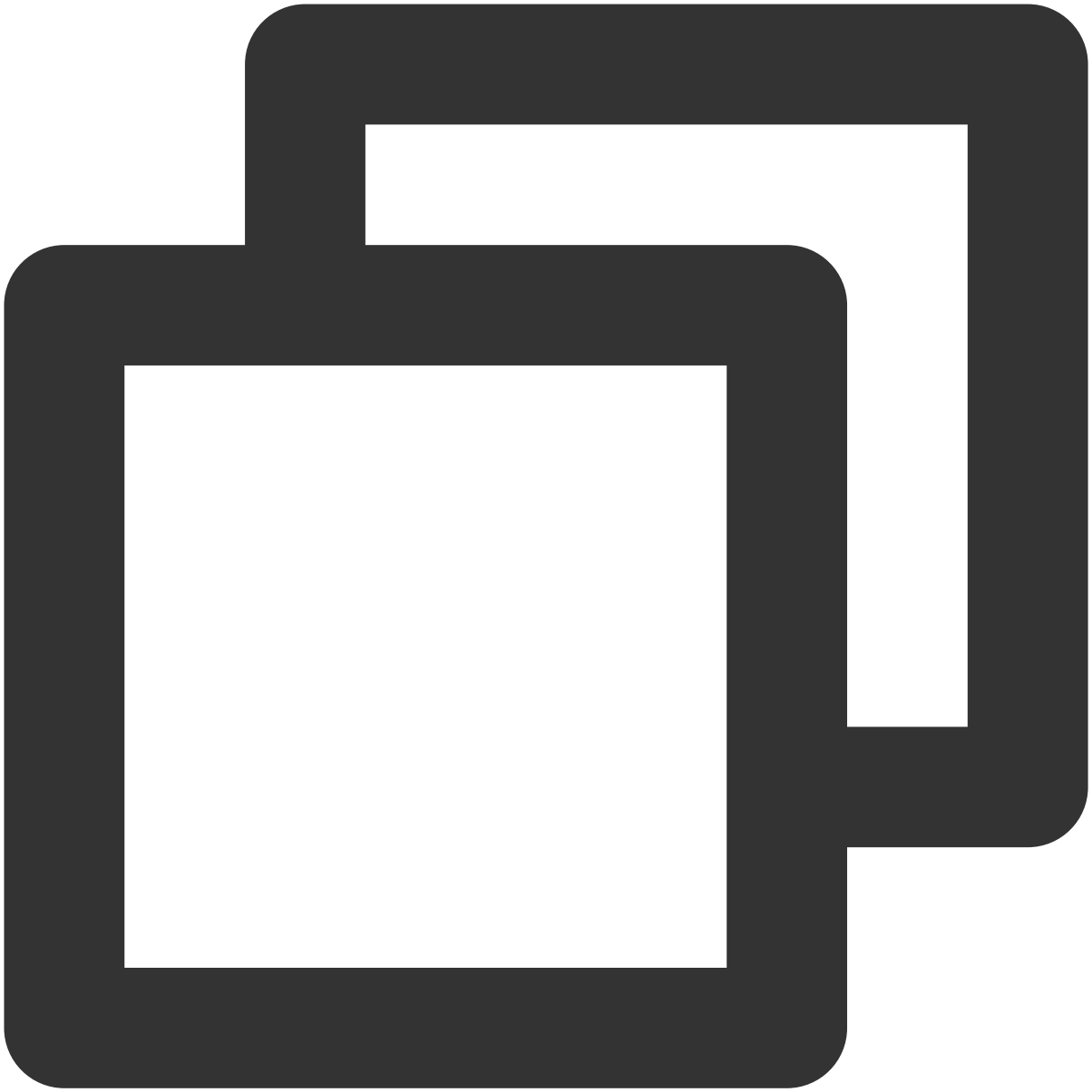
Feature Description: Starts pull-down refresh. Upon invocation, it triggers the pull-down refresh animation, mirroring the effect of a user-initiated pull-down refresh.

Parameter and Description: Object.

Attribute	Type	Default value	Required	Description
success	function	-	No	Callback Function of Successful Interface Call
fail	function	-	No	Callback Function of Failing Interface Call
complete	function	-	No	Callback function executed upon the completion of the interface invocation (both successful and unsuccessful invocations)

Sample Code

©2013-2022 Tencent Cloud. All rights reserved.



```
wx.startPullDownRefresh()
```

stopPullDownRefresh

This API is used via wx.stopPullDownRefresh(Object object).

©2013-2022 Tencent Cloud. All rights reserved.

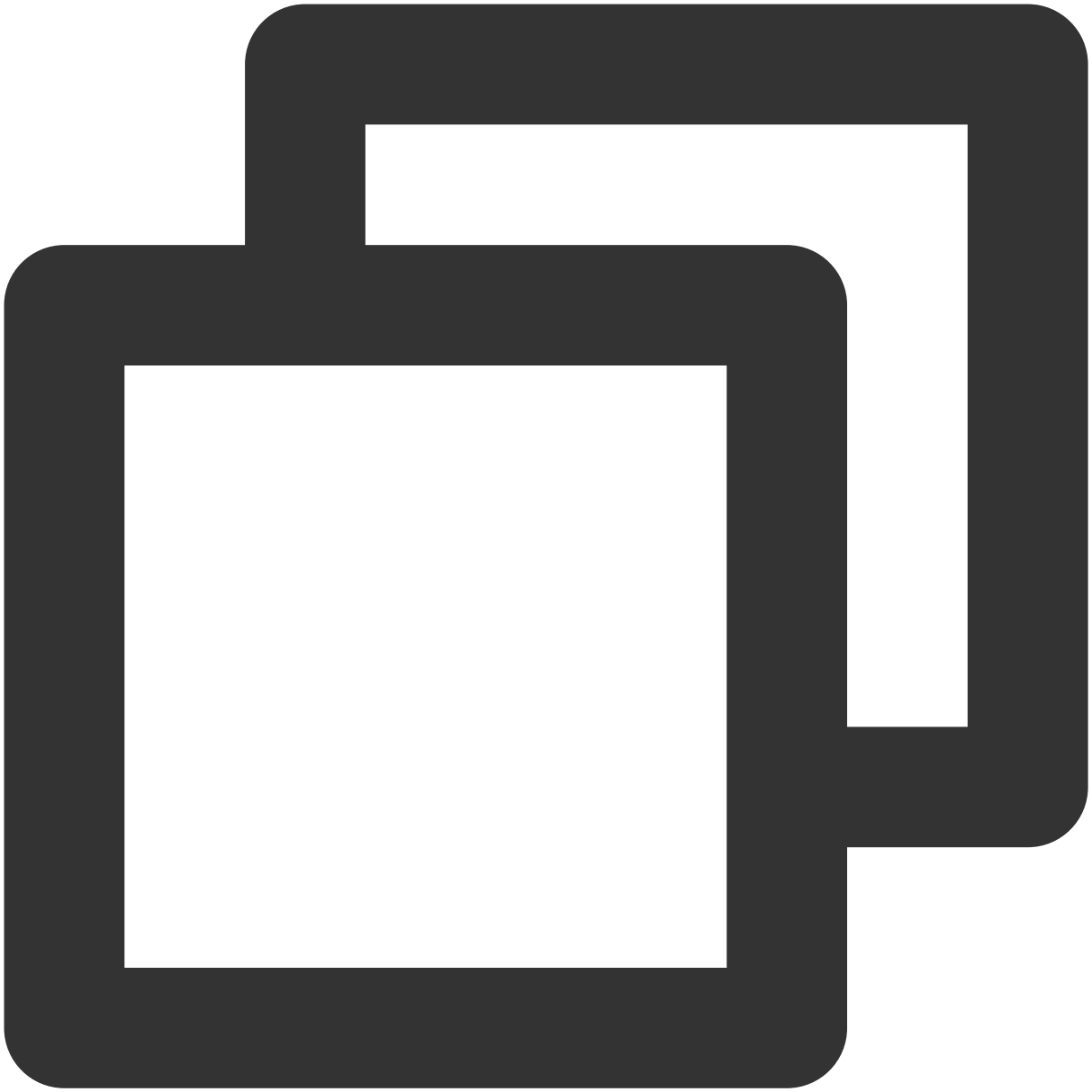
Feature Description: Disables pull-down refresh on the current page.

Parameter and Description: Object.

Attribute	Type	Default value	Required	Description
success	function	-	No	Callback Function of Successful Interface Call
fail	function	-	No	Callback Function of Failing Interface Call
complete	function	-	No	Callback function executed upon the completion of the interface invocation (both successful and unsuccessful invocations)

Sample Code

©2013-2022 Tencent Cloud. All rights reserved.



```
Page({
  onPullDownRefresh () {
    wx.stopPullDownRefresh()
  }
})
```

©2013-2022 Tencent Cloud. All rights reserved.

Scroll

Last updated：2024-07-12 19:33:49

pageScrollTo

This API is used via wx.pageScrollTo(Object object).

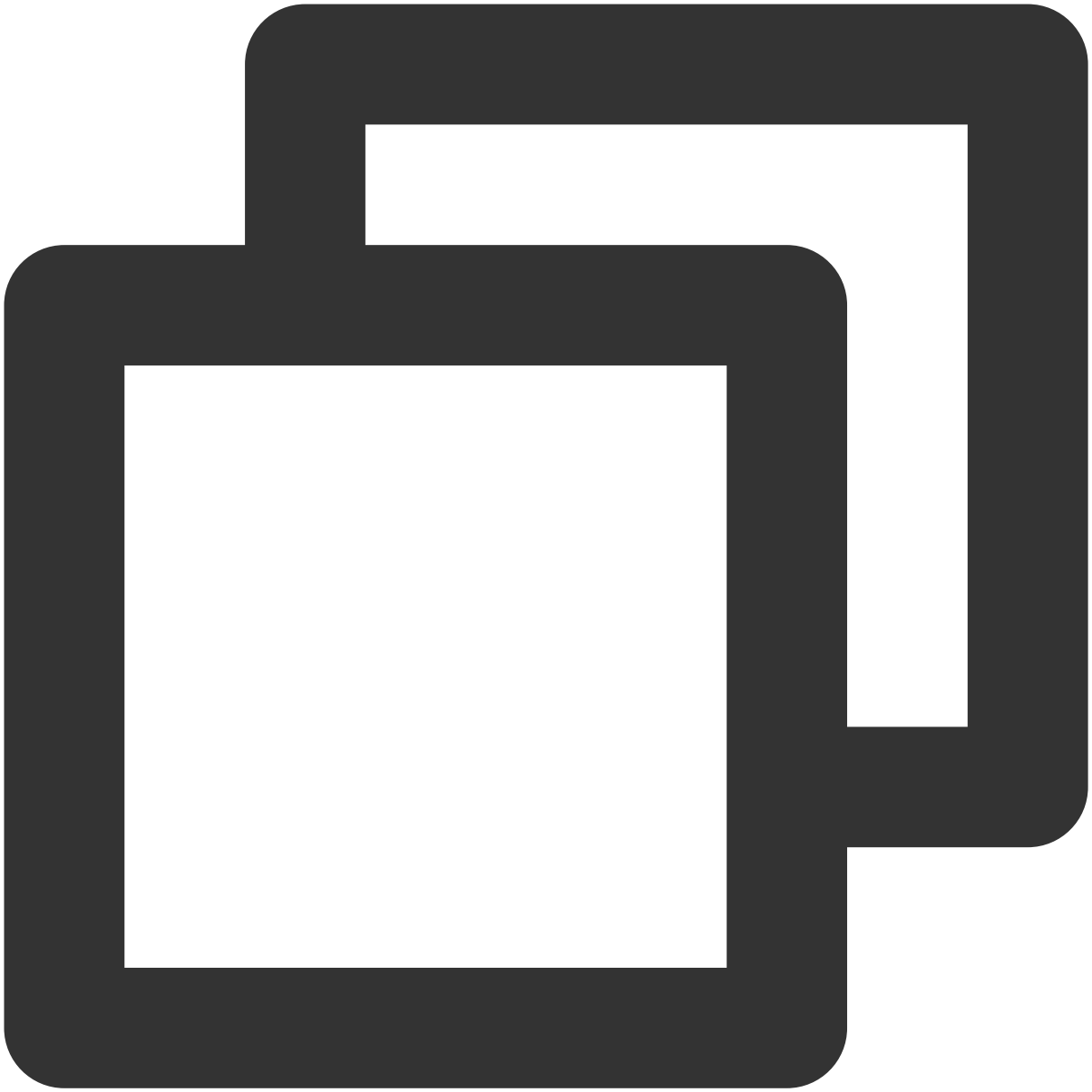
Feature Description: Scrolls the page to the desired location.

Parameter and Description: Object.

Attribute	Type	Default value	Required	Description
scrollTop	number	-	Yes	Scroll to the target position on the page, measured in pixels.
duration	number	300	No	Duration of the scrolling animation, measured in milliseconds.
success	function	-	No	Callback Function of Successful Interface Call
fail	function	-	No	Callback Function of Failing Interface Call
complete	function	-	No	Callback function executed upon the completion of the interface invocation (both successful and unsuccessful invocations)

Sample Code

©2013-2022 Tencent Cloud. All rights reserved.



```
wx.pageScrollTo({
  scrollTop: 0,
  duration: 300
})
```

ScrollViewContext

©2013-2022 Tencent Cloud. All rights reserved.

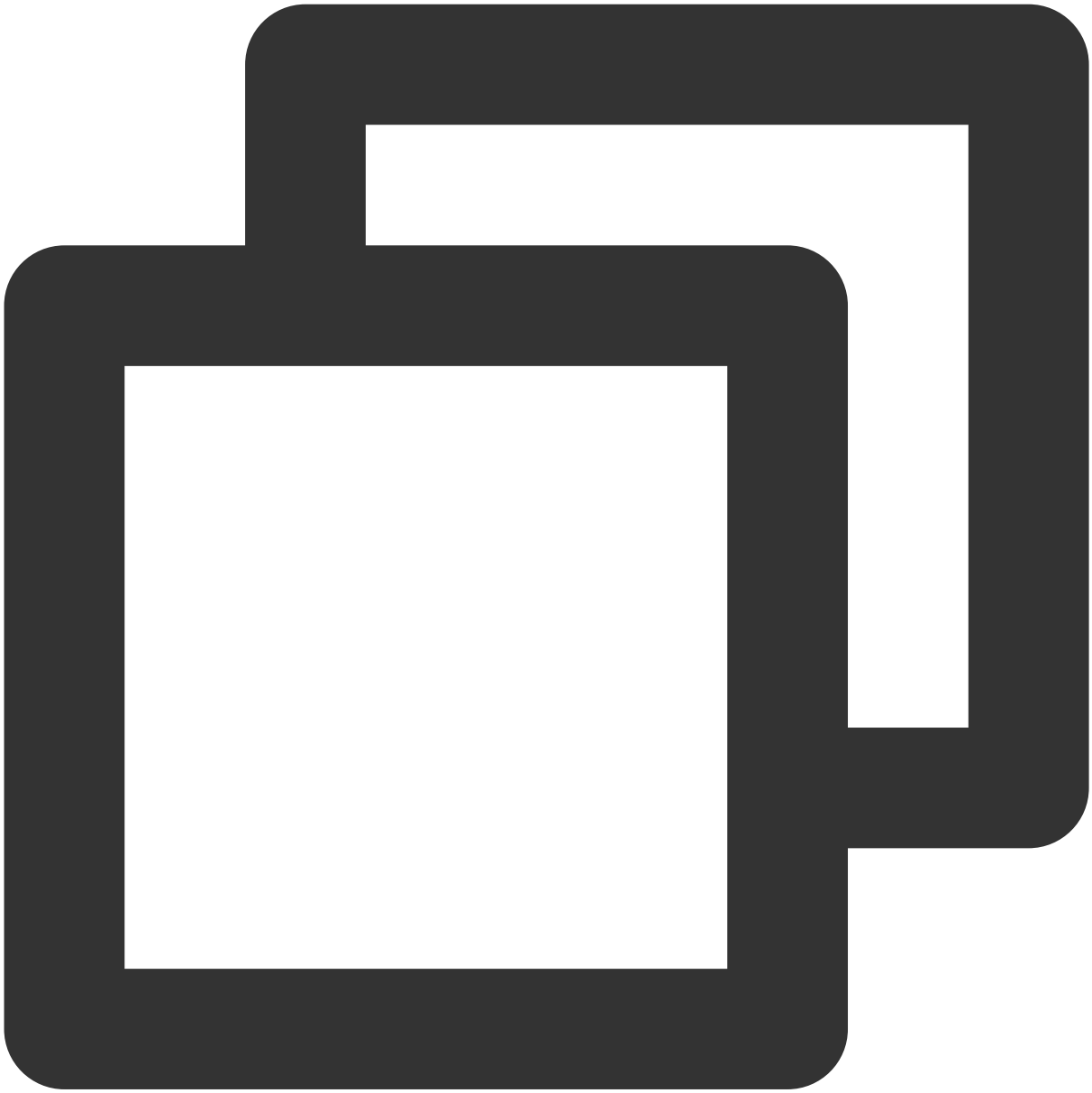
Feature Description: Enhances ScrollView instance, which can be obtained through the wx.createSelectorQuery's NodesRef.node method. Effective only when the scroll-view component's enhanced attribute is enabled.

Attribute and Descriptions

Attribute	Type	Description
scrollEnabled	boolean	Scrolling Switch
bounces	boolean	Set the scroll boundary elasticity (currently unsupported)
showScrollbar	boolean	Set the visibility of the scrollbar
pagingEnabled	boolean	Pagination Slide Switch
fastDeceleration	boolean	Set scroll deceleration rate (iOS only)
decelerationDisabled	boolean	Cancel the scroll inertia (Effective only on iOS)

Sample Code

©2013-2022 Tencent Cloud. All rights reserved.



```
wx.createSelectorQuery()
  .select('#scrollview')
  .node()
  .exec((res) => {
    const scrollView = res[0].node;
    scrollView.scrollEnabled = false;
  })
```

©2013-2022 Tencent Cloud. All rights reserved.

.scrollIntoView

The method is used via: ScrollViewContext.scrollIntoView(string selector, object ScrollIntoViewOptions)

Feature Description: Scrolls to the specified position.

Parameters and Description: string selector, element selector.

.scrollTo

The method is used via ScrollViewContext.scrollTo(Object object)

Feature Description: Scrolls to the specified position.

Parameter and Description: Object.

Attribute	Type	Default value	Required	Description
top	number	-	No	Top distance
left	number	-	No	Left Boundary Distance
velocity	number	-	No	Initial Speed (iOS only)
duration	number	-	No	Scroll animation length (iOS only)
animated	boolean	-	No	Enable scroll animation or not

©2013-2022 Tencent Cloud. All rights reserved.

Animation

Last updated：2024-07-12 19:33:26

createAnimation

This API is used via wx.createAnimation(Object object).

Feature Description: Creates an animation instance animation and invokes the instance's methods to depict the animation, before ultimately exporting the animation data through the instance's export method and passing it to the component's animation attribute.

Parameter and Description: Object.

Attribute	Type	Default value	Required	Note
duration	number	400	No	Animation duration, measured in milliseconds.
timingFunction	string	'linear'	No	Animation effect
delay	number	0	No	Animation delay duration, measured in milliseconds.
transformOrigin	string	'50% 50% 0'	No	-

Valid values of timingFunction

Value	Note
'linear'	The animation maintains a consistent speed from beginning to end.
'ease'	The animation starts at a slow pace, then accelerates, and finally decelerates before concluding.
'ease-in'	The animation starts at a slow pace.
'ease-in-out'	The animation starts and concludes at a slow pace.
'ease-out'	The animation concludes at a slow pace.
'step-start'	The animation leaps to its concluding state from the very first frame and maintains this until the end.

©2013-2022 Tencent Cloud. All rights reserved.

'step-end'	The animation consistently maintains its initial state, leaping to its concluding state only in the final frame.

Return Value：Animation.

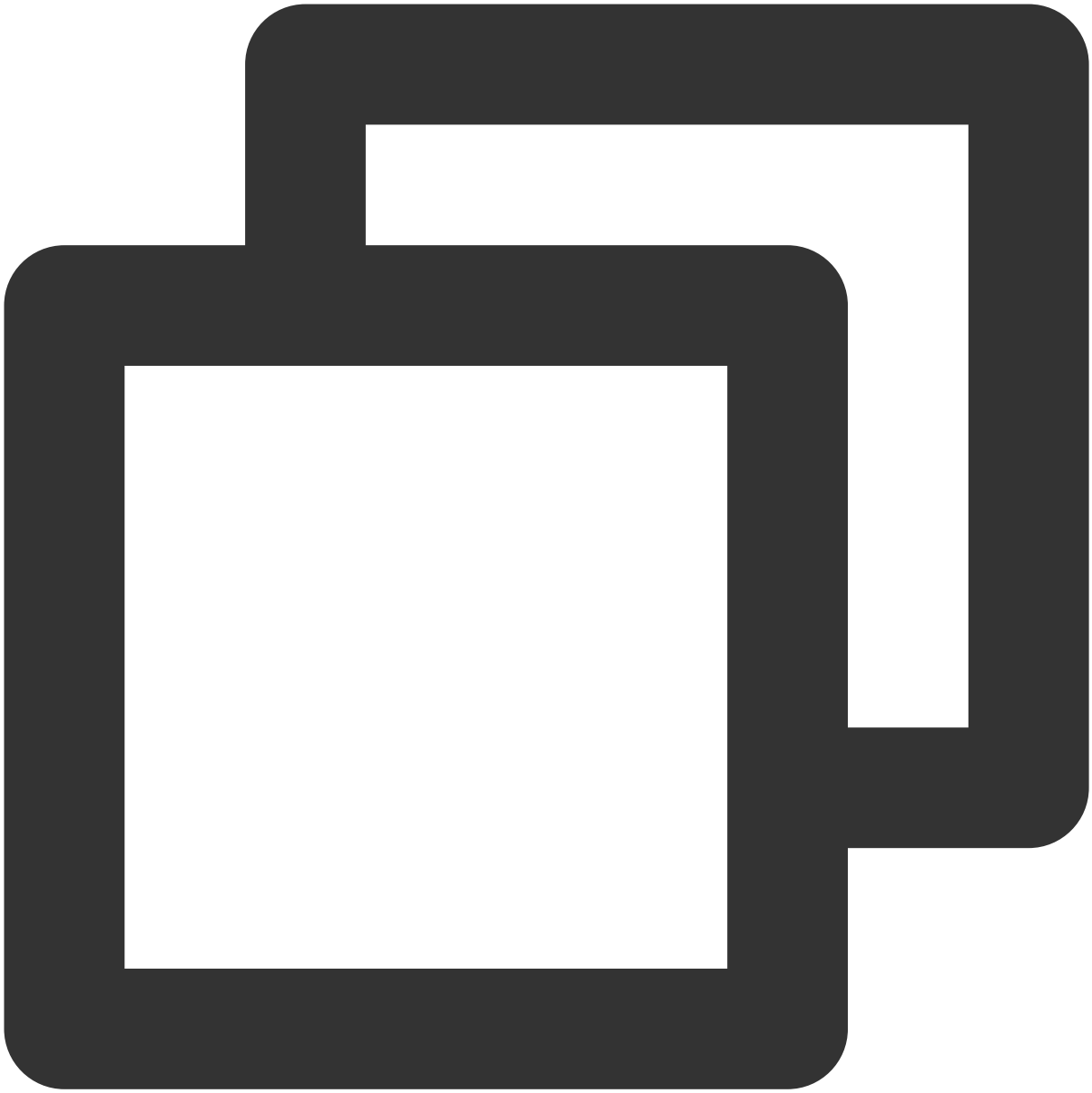
Animation

Animation object.

Sample code

Corresponding WXML file

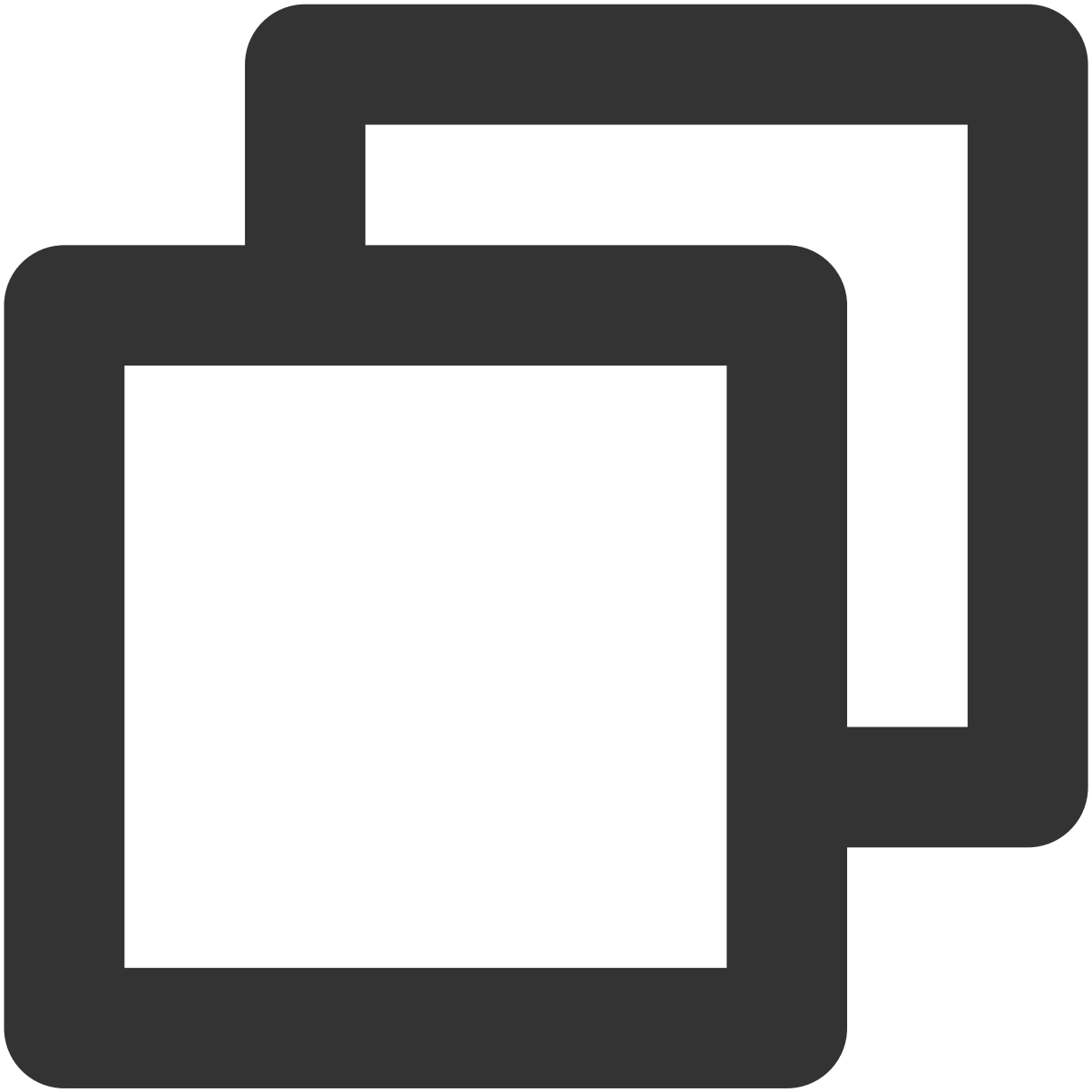
©2013-2022 Tencent Cloud. All rights reserved.



```
<view
  animation="{{animationData}}"
  style="background:red;height:100rpx;width:100rpx"
></view>
```

Corresponding js file

©2013-2022 Tencent Cloud. All rights reserved.



```
Page({
  data: {
    animationData: {}
  },
  onShow() {
    const animation = wx.createAnimation({
      duration: 1000,
      timingFunction: 'ease',
    })

    this.animation = animation
```

©2013-2022 Tencent Cloud. All rights reserved.

```
    animation.scale(2, 2).rotate(45).step()

    this.setData({
      animationData: animation.export()
    })

    setTimeout(function () {
      animation.translate(30).step()
      this.setData({
        animationData: animation.export()
      })
    }.bind(this), 1000)
  },
  rotateAndScale() {
    // Simultaneous rotation and enlargement
    this.animation.rotate(45).scale(2, 2).step()
    this.setData({
      animationData: this.animation.export()
    })
  },
  rotateThenScale() {
    // Rotate first, then enlarge.
    this.animation.rotate(45).step()
    this.animation.scale(2, 2).step()
    this.setData({
      animationData: this.animation.export()
    })
  },
  rotateAndScaleThenTranslate() {
    // Simultaneous rotation and enlargement before translation
    this.animation.rotate(45).scale(2, 2).step()
    this.animation.translate(100, 100).step({duration: 1000})
    this.setData({
      animationData: this.animation.export()
    })
  }
})
```

.backgroundColor

The method is used via Animation.Animation.backgroundColor(string value).

Feature Description: Sets the background color.

©2013-2022 Tencent Cloud. All rights reserved.

Parameter and Description: string value, color value.
Return Value：Animation.

.bottom

This method is used via Animation.Animation.bottom(number|string value).

Feature Description: Sets the bottom value.
Parameter and Description: number|string value, length value. If a number is passed in, px is used by default. Length values in other custom units can also be passed in.
Return Value：Animation.

.export

This API is used via Object Animation.export().

Feature Description: Exports the animation queue. The export method will clear the previous animation operations each time it is called.
Return Value: Object, animationData.

Attribute	Type
actions	Array.<Object>

.height

This method is used via Animation Animation.height(number|string value).

Feature Description: Sets height.
Parameter and Description: number|string value, length value. If a number is passed in, px is used by default. Length values in other custom units can also be passed in.
Return Value：Animation.

.left

©2013-2022 Tencent Cloud. All rights reserved.

This method is used via Animation Animation.left(number|string value).

Feature Description: Sets the left value.
Parameter and Description: number|string value, length value. If a number is passed in, px is used by default. Length values in other custom units can also be passed in.
Return Value：Animation.

.matrix

This API is used via Animation Animation.matrix().

Feature Description: Consistent with transform-function matrix.
Return Value：Animation.

.matrix3d

This API is used via Animation Animation.matrix3d()

Feature Description: Consistent with transform-function matrix.
Return Value：Animation.

.opacity

This method is used via Animation Animation.opacity(number value).

Feature Description: Sets the opacity.
Parameter and Description: number value, opacity, ranging from 0 to 1.
Return Value：Animation.

.right

This method is used via Animation Animation.right(number|string value).

©2013-2022 Tencent Cloud. All rights reserved.

Feature Description: Sets the right value.
Parameter and Description: number|string value, length value. If a number is passed in, px is used by default. Length values in other custom units can also be passed in.
Return Value：Animation.

.rotate

This API is used via Animation Animation.rotate(number angle).

Feature Description: Rotate one angle clockwise from the origin.
Parameter and Description: number angle, the degree of rotation. Range [-180, 180].
Return Value：Animation.

.rotate3d

This API is used via Animation Animation.rotate3d(number x, number y, number z, number angle).

Feature Description: Rotates a degree clockwise from a fixed axis.
Parameter and Description: number x, the x-coordinate of the rotation axis; number y, the y-coordinate of the rotation axis; number z, the z-coordinate of the rotation axis; number angle, the degree of rotation. Range [-180, 180].
Return Value：Animation.

.rotateX

This API is used via Animation Animation.rotateX(number angle).

Feature Description: Rotates a degree clockwise from the X-axis.
Parameter and Description: number angle, the degree of rotation. Range [-180, 180].
Return Value：Animation.

.rotateY

This method is used via Animation Animation.rotateY(number angle).

©2013-2022 Tencent Cloud. All rights reserved.

Feature Description: Rotates a degree clockwise from the Y-axis.
Parameter and Description: number angle, the degree of rotation. Range [-180, 180].
Return Value：Animation.

.rotateZ

This Method is used via Animation Animation.rotateZ(number angle).

Feature Description: Rotates a degree clockwise from the Z-axis.
Parameter and Description: number angle, the degree of rotation. Range [-180, 180].
Return Value：Animation.

.scale

This method is used via Animation Animation.scale(number sx, number sy).

Feature Description: Scaling.
Parameter and Description: number sx, when only sx parameter is present, it indicates scaling on the X and Y axes simultaneously by sx times; number sy, scaling on the Y axis by sy times.
Return Value：Animation.

.scale3d

This method is used via Animation Animation.scale3d(number sx, number sy, number sz).

Feature Description: Scaling.
Parameter and Description: number sx, the scaling factor for the x-axis; number sy, the scaling factor for the y-axis; number sz, the scaling factor for the z-axis.
Return Value：Animation.

.scaleX

This method is used via Animation Animation.scaleX(number scale).

©2013-2022 Tencent Cloud. All rights reserved.

Feature Description: Scaling on the X-axis.
Parameter and Description: number scale, the scaling factor on the X-axis.
Return Value：Animation.

scaleY

This method is used via Animation Animation.scaleY(number scale).

Feature Description: Scaling on the Y-axis.
Parameter and Description: number scale, the scaling factor on the Y-axis.
Return Value：Animation.

.scaleZ

This method is used via Animation Animation.scaleZ(number scale).

Feature Description: Scaling on the Z-axis.
Parameter and Description: number scale, the scaling factor on the Z-axis.
Return Value：Animation.

.skew

This method is used via Animation Animation.skew(number ax, number ay).

Feature Description: Skews the coordinates on the X and Y axes.
Parameter and Description: number ax, tilt angle to X-axis coordinates, range [-180, 180]; number ay, tilt angle to Y-axis coordinates, range [-180, 180].
Return Value：Animation.

.skewX

This Method is used via Animation Animation.skewX(number angle).

Feature Description: Skews the coordinates on the X-axis.

©2013-2022 Tencent Cloud. All rights reserved.

Parameter and Description: number angle, tilt angle, range [-180, 180].
Return Value：Animation.

.skewY

This method is used via Animation Animation.skewY(number angle).

Feature Description: Skews the coordinates on the Y-axis.
Parameter and Description: number angle, tilt angle, range [-180, 180].
Return Value：Animation.

.step

This API is used via Animation Animation.step(Object object).

Feature Description: Indicates the completion of a set of animations. Any number of animation methods can be called within a set of animations. All animations in a set will start simultaneously, and the next set of animations will only proceed after the completion of the current set.
Parameter and Description: Object.

Attribute	Type	Default value	Required	Note
duration	number	400	No	Animation duration, measured in milliseconds.
timingFunction	string	'linear'	No	Animation effect
delay	number	0	No	Animation delay duration, measured in milliseconds.
transformOrigin	string	'50% 50% 0'	No	-

Valid values of timingFunction

Value	Note
'linear'	The animation maintains a consistent speed from beginning to end.
'ease'	The animation starts at a slow pace, then accelerates, and finally decelerates before concluding.

©2013-2022 Tencent Cloud. All rights reserved.

'ease-in'	The animation starts at a slow pace.
'ease-in-out'	The animation starts and concludes at a slow pace.
'ease-out'	The animation concludes at a slow pace.
'step-start'	The animation leaps to its concluding state from the very first frame and maintains this until the end.
'step-end'	The animation consistently maintains its initial state, leaping to its concluding state only in the final frame.

Return Value：Animation.

.top

This method is used via Animation Animation.top(number|string value).

Feature Description: Sets the top value.
Parameter and Description: number|string value, length value. If a number is passed in, px is used by default. Length values in other custom units can also be passed in.
Return Value：Animation.

.translate

This method is used via Animation Animation.translate(number tx, number ty).

Feature Description: Translation and transformation.
Parameter and Description: number tx, when this is the only parameter, it represents the offset on the X-axis by tx, measured in pixels; number ty, the distance of translation on the Y-axis, measured in pixels.
Return Value：Animation.

.translate3d

This method is used via Animation Animation.translate3d(number tx, number ty, number tz).

Feature Description: Performs a translation transformation on the xyz coordinates.

©2013-2022 Tencent Cloud. All rights reserved.

Parameter and Description: number tx, when only this parameter is present, indicates an offset tx in the X-axis in px; number ty, the distance in the Y-axis in px; number tz, the distance in the Z-axis in px.
Return Value：Animation.

.translateX

This method is used via Animation Animation.translateX(number translation).

Feature Description: Performs a translation on the X-axis.
Parameter and Description: number translation, the distance of translation on the X-axis, measured in pixels.
Return Value：Animation.

.translateY

This method is used via Animation Animation.translateY(number translation).

Feature Description: Performs a translation on the Y-axis.
Parameter and Description: number translation, the distance of translation on the Y-axis, measured in pixels.
Return Value：Animation.

.translateZ

This method is used via Animation Animation.translateZ(number translation).

Feature Description: Performs a translation on the Z-axis.
Parameter and Description: number translation, the distance of translation on the Z-axis, measured in pixels.
Return Value：Animation.

.width

This method is used via Animation Animation.width(number|string value).

Feature Description: Sets the width.

©2013-2022 Tencent Cloud. All rights reserved.

Parameter and Description: number|string value, length value. If a number is passed in, px is used by default. Length values in other custom units can also be passed in.

Return Value： Animation.

©2013-2022 Tencent Cloud. All rights reserved.

Custom component

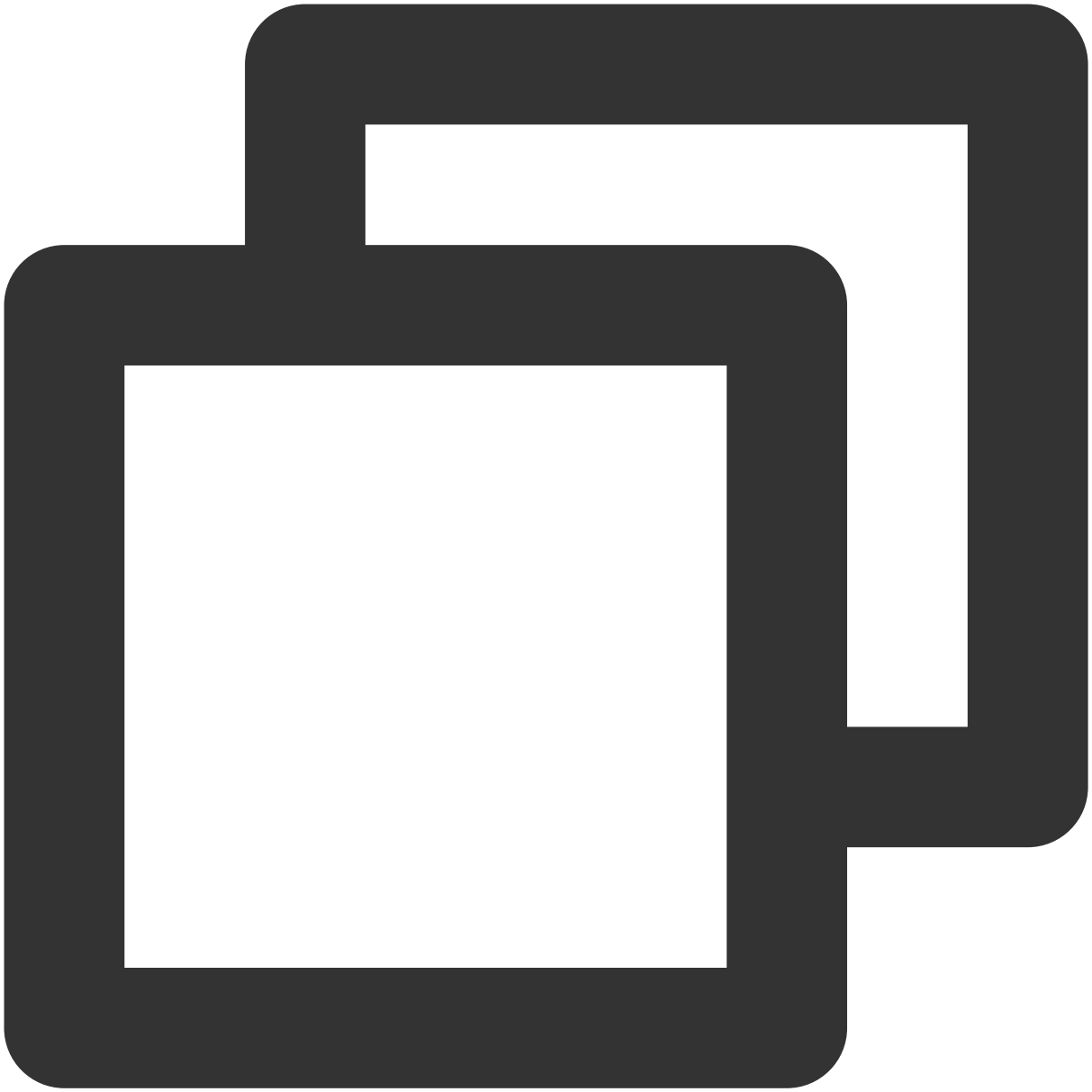
Last updated：2024-07-12 19:32:59

nextTick

The usage method of this API is wx.nextTick(function callback).

Feature Description: Defers certain operations to be executed in the next time slice, similar to the setTimeout feature.
Parameter and Description: The function callback is used because the setData and triggerEvent interfaces in custom components are inherently synchronous operations. When these interfaces are called consecutively, they are executed within a single synchronous process, which may lead to errors in the case of improper logic.
An extreme case: If the parent component's setData triggers the child component's triggerEvent, causing the parent component to perform setData again, and in the process, the child component is unloaded through the wx:if statement, this could potentially cause an error. Therefore, for logic that does not need to be completed within a single synchronous process, this interface can be used to defer execution to the next time slice.
Sample Code

©2013-2022 Tencent Cloud. All rights reserved.



```
Component({
  doSth() {
    this.setData({ number: 1 }) // Executed directly within the current synchronous

    wx.nextTick(() => {
      this.setData({ number: 3 }) // Executed after the current synchronous process
    })

    this.setData({ number: 2 }) // Executed directly within the current synchronous
  }
})
```

©2013-2022 Tencent Cloud. All rights reserved.

©2013-2022 Tencent Cloud. All rights reserved.

Menu

Last updated：2024-07-12 19:32:29

getMenuButtonBoundingClientRect

This API is used via wx.getMenuButtonBoundingClientRect().

Feature Description: Gets information about the layout position of the menu button (capsule button in the upper right corner). The coordinate information has the upper left corner of the screen as the origin .

Return Value: Object.

Layout Position Information of the Menu Buttons:

Attribute	Type	Description
width	number	Width, measured in pixels (px)
height	number	Height, measured in pixels (px)
top	number	Upper Boundary Coordinate, Unit: px
right	number	Right Boundary Coordinate, Unit: px
bottom	number	Lower Boundary Coordinate, Unit: px
left	number	Left Boundary Coordinate, Unit: px

©2013-2022 Tencent Cloud. All rights reserved.

Network
UDP Communication

Last updated：2024-07-12 19:32:04

createUDPSocket

This API is used via UDPSocket?wx.createUDPSocket().

Feature description: Creates a UDP Socket instance.
Return Value: UDPSocket, a UDP Socket instance.

UDPSocket

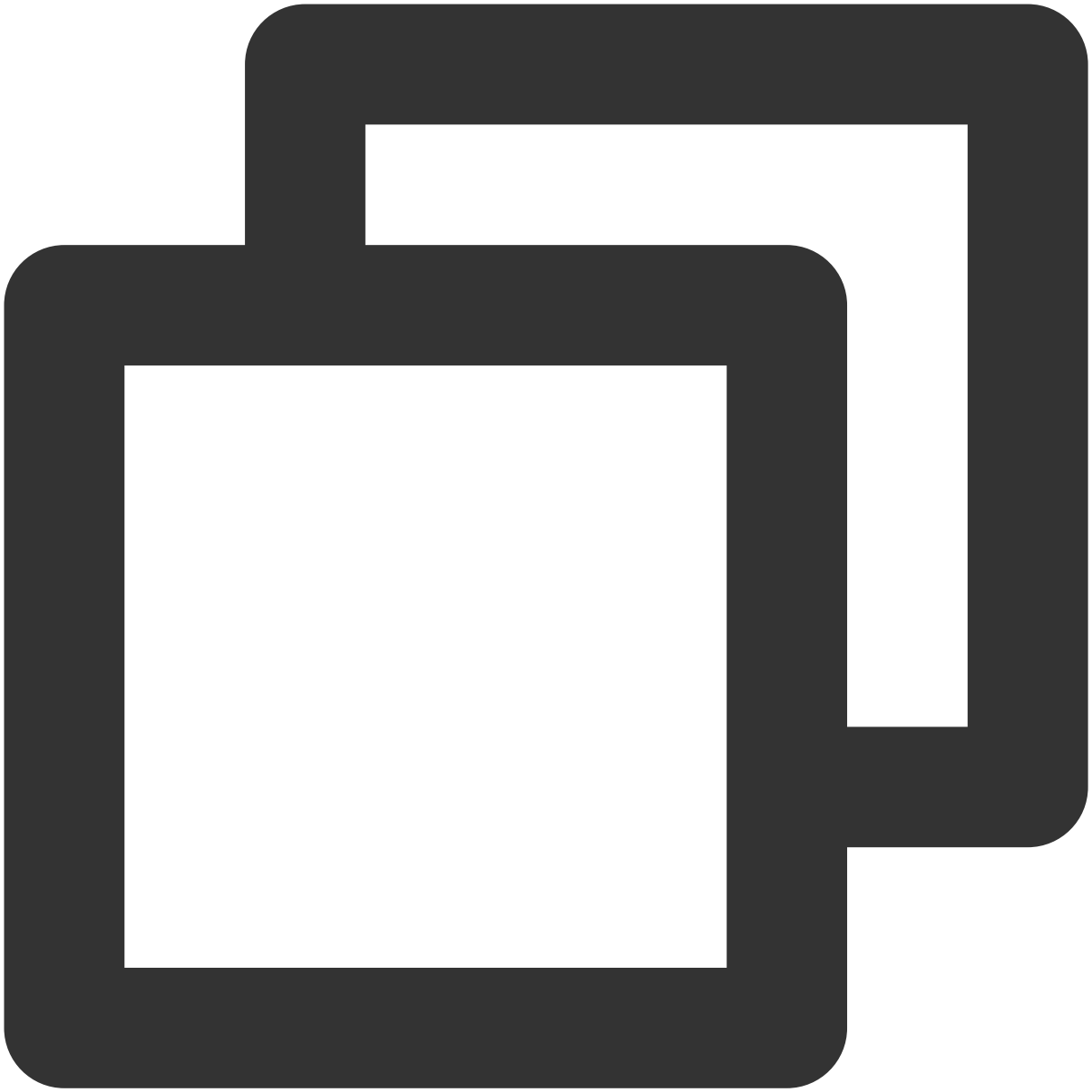
Note:
A UDP Socket instance, using IPv4 protocol by default.

.bind

This method is used via number UDPSocket.bind(number port).

Feature Description: Binds to a system-assigned random available port, or bind to a specified port number.
Parameter and Description: number port, No. of specified port to be bound. If this parameter is not passed in, the system randomly assigns an available port.
Return Value: number, the successfully bound port number.
Sample Code

©2013-2022 Tencent Cloud. All rights reserved.



```
const udp = wx.createUDPSocket()
const port = udp.bind()
```

.close

This method is used via UDPSocket.close().

©2013-2022 Tencent Cloud. All rights reserved.

Feature Description: Closes the UDP Socket instance, effectively terminating it. After closure, the UDP Socket instance can no longer send messages. Each call to UDPSocket.send will trigger an error event, and the message event callback function will no longer be executed. After a UDPSocket instance is created, it is strongly referenced by Native to ensure it is not garbage collected. After UDPSocket.close, the strong reference is removed, allowing the UDPSocket instance to be subject to garbage collection.

.connect

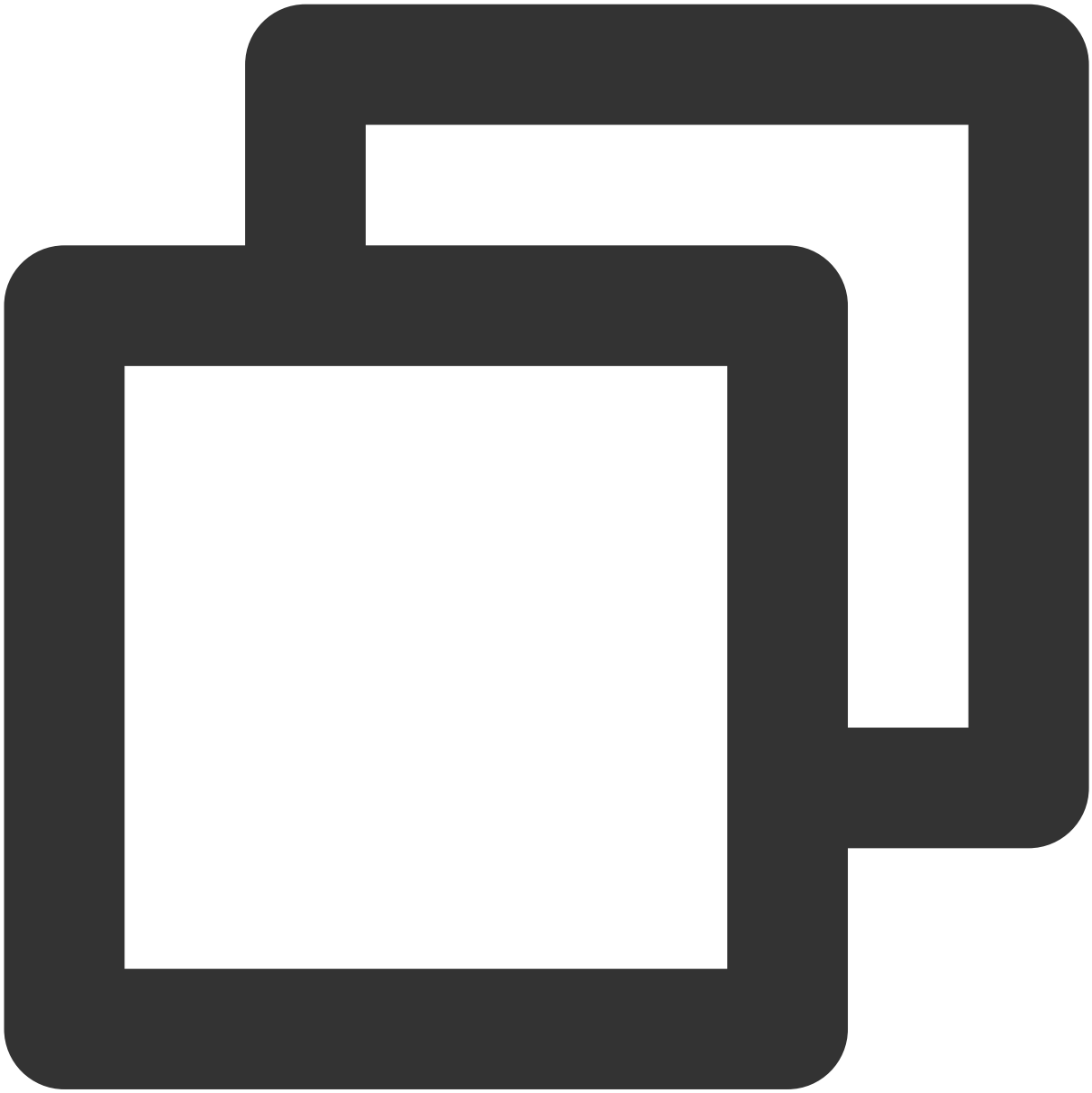
> This method is used via UDPSocket.connect(Object object).

Feature Description: Pre-connects to the specified IP and port, to be used in conjunction with the write method.
Parameter and Description: Object

Attribute	Type	Default value	Required	Note
address	string	-	Yes	Destination for message transmission
Port No.	number	-	Yes	Port number for message transmission

Sample Code

©2013-2022 Tencent Cloud. All rights reserved.



```
const udp = wx.createUDPSocket()
udp.bind()
udp.connect({
  address: '192.168.193.2',
  port: 8848,
})
udp.write({
  address: '192.168.193.2',
  port: 8848,
  message: 'hello, how are you'
})
```

©2013-2022 Tencent Cloud. All rights reserved.

.onClose

This method is used via UDPSocket.onClose(function listener).

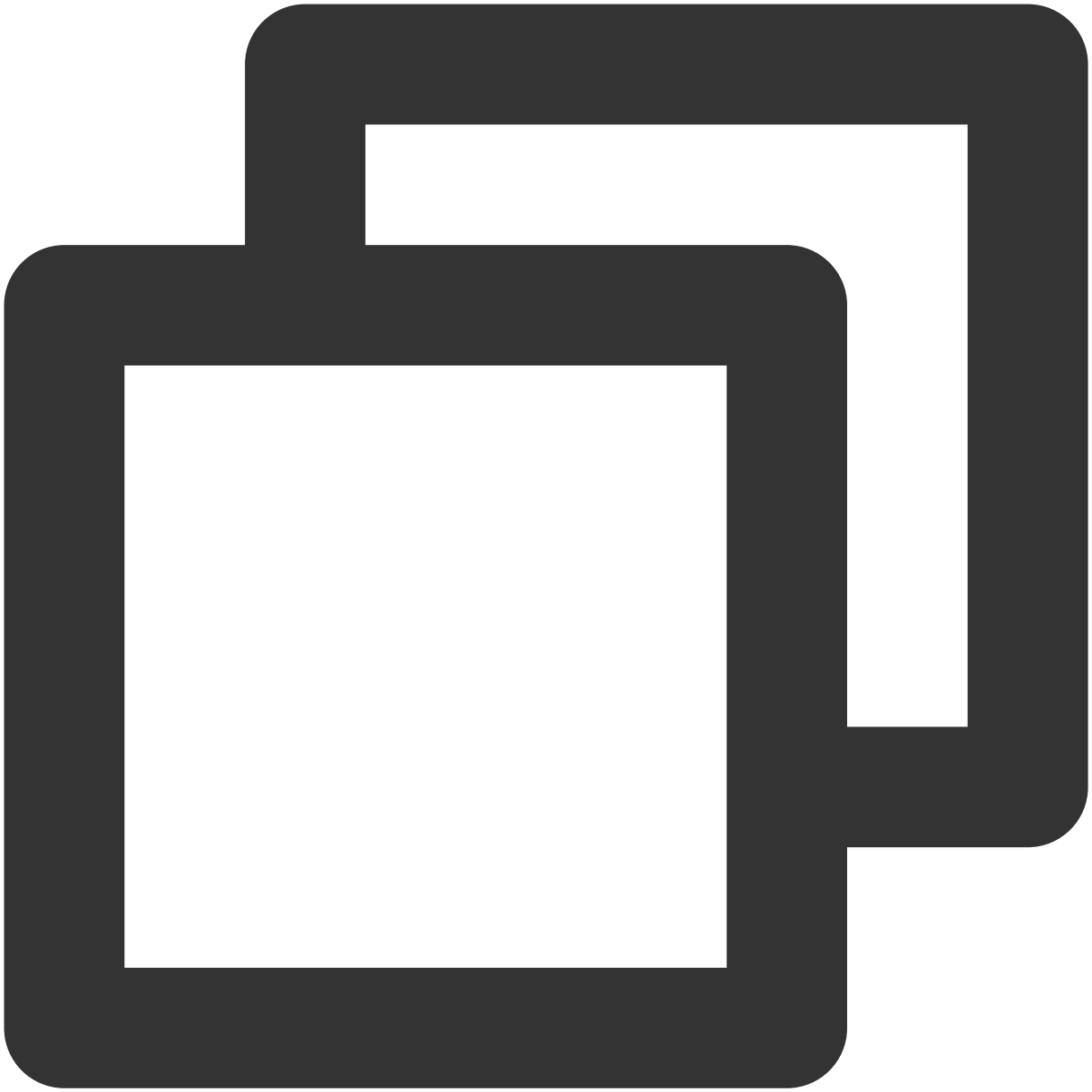
Feature Description: Monitors closure events.
Parameter and Description: The listener function of closure events.

.offClose

This method is used via UDPSocket.offClose(function listener).

Feature Description: Removes the listener function for closure events.
Parameter and Description: function listener, the listener function passed in by onClose. If this parameter is not passed in, all listener functions will be removed.
Sample Code

©2013-2022 Tencent Cloud. All rights reserved.



```
const listener = function (res) { console.log(res) }

UDPSocket.onClose(listener)
UDPSocket.offClose(listener) // The same function object as the listener must be pa
```

.onError

This method is used via UDPSocket.onError(function listener).

©2013-2022 Tencent Cloud. All rights reserved.

Feature Description: Monitors error events.

Parameter and Description: Object res parameter, function listener, the listener function for error events.

Attribute	Type	Note
errMsg	string	Error Message

.offError

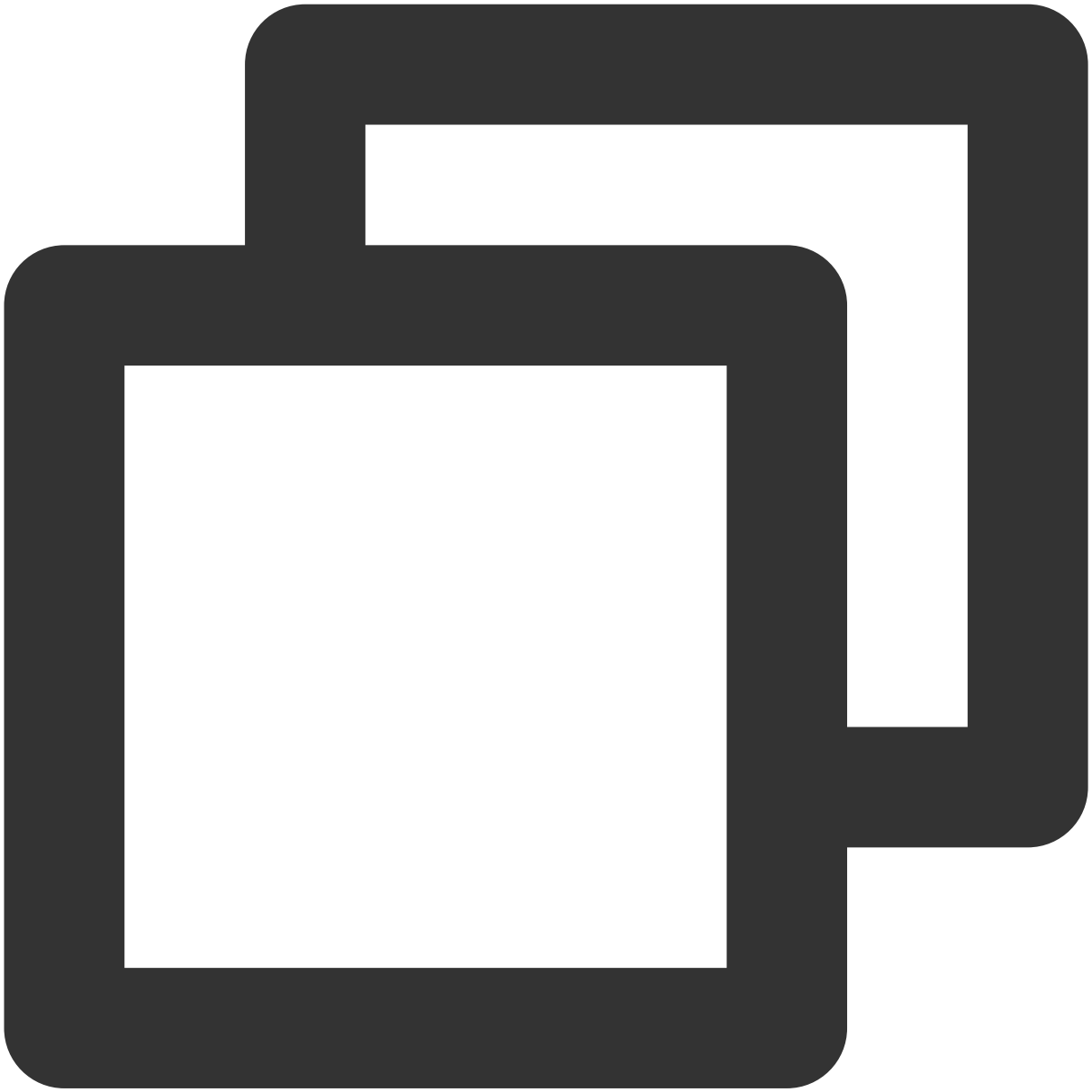
This method is used via UDPSocket.offError(function listener).

Feature Description: Removes the listener function for error events.

Parameter and Description: function listener, error event listener function, with the following arguments Object res:

Sample Code

©2013-2022 Tencent Cloud. All rights reserved.



```
const listener = function (res) { console.log(res) }

UDPSocket.onError(listener)
UDPSocket.offError(listener) // The same function object as the listener must be pa
```

.onListening

This method is used via UDPSocket.onListening(function listener).

©2013-2022 Tencent Cloud. All rights reserved.

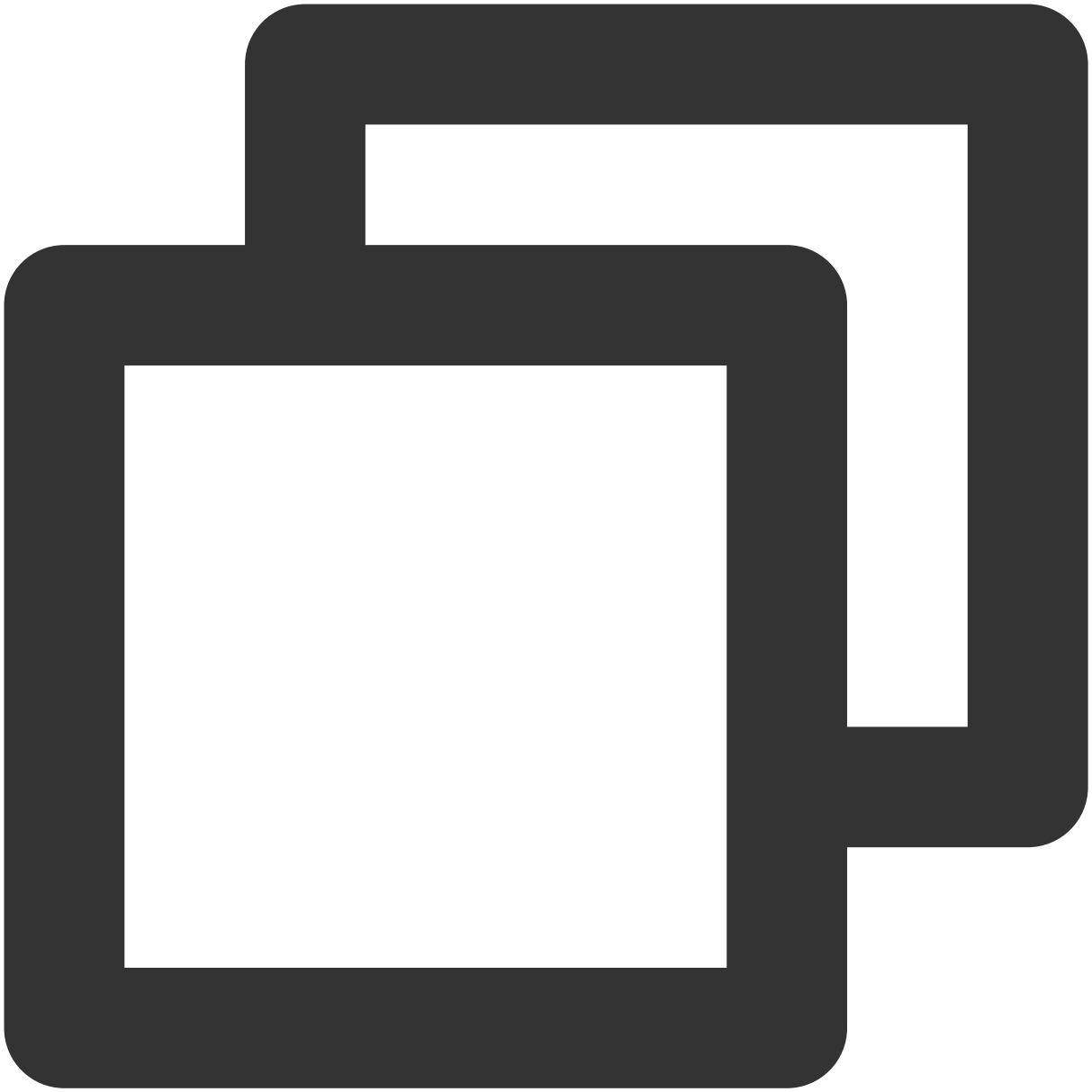
Feature Description: Monitors events of starting to monitor data packet messages.
Parameter and Description: function listener, the listener function for events of starting to monitor data packet messages.

.offListening

This method is used via UDPSocket.offListening(function listener).

Feature Description: Removes the listener function for events of starting to monitor data packet messages.
Parameter and Description: function listener, the listener function passed in by onListening. If this parameter is not passed in, all listener functions will be removed.
Sample Code

©2013-2022 Tencent Cloud. All rights reserved.



```
const listener = function (res) { console.log(res) }

UDPSocket.onListening(listener)
UDPSocket.offListening(listener) // The same function object as the listener must b
```

.onMessage

This method is used via UDPSocket.onMessage(function listener).

©2013-2022 Tencent Cloud. All rights reserved.

Feature Description: Monitors events of receiving a message.
Parameter and Description: function listener, the listener function for the event that received the message, with the following arguments Object res:

Attribute	Type	Note
message	ArrayBuffer	The received message. The length of the message must be less than 4096.
remoteInfo	Object	Sender's address information

Structure attributes of "remoteInfo"

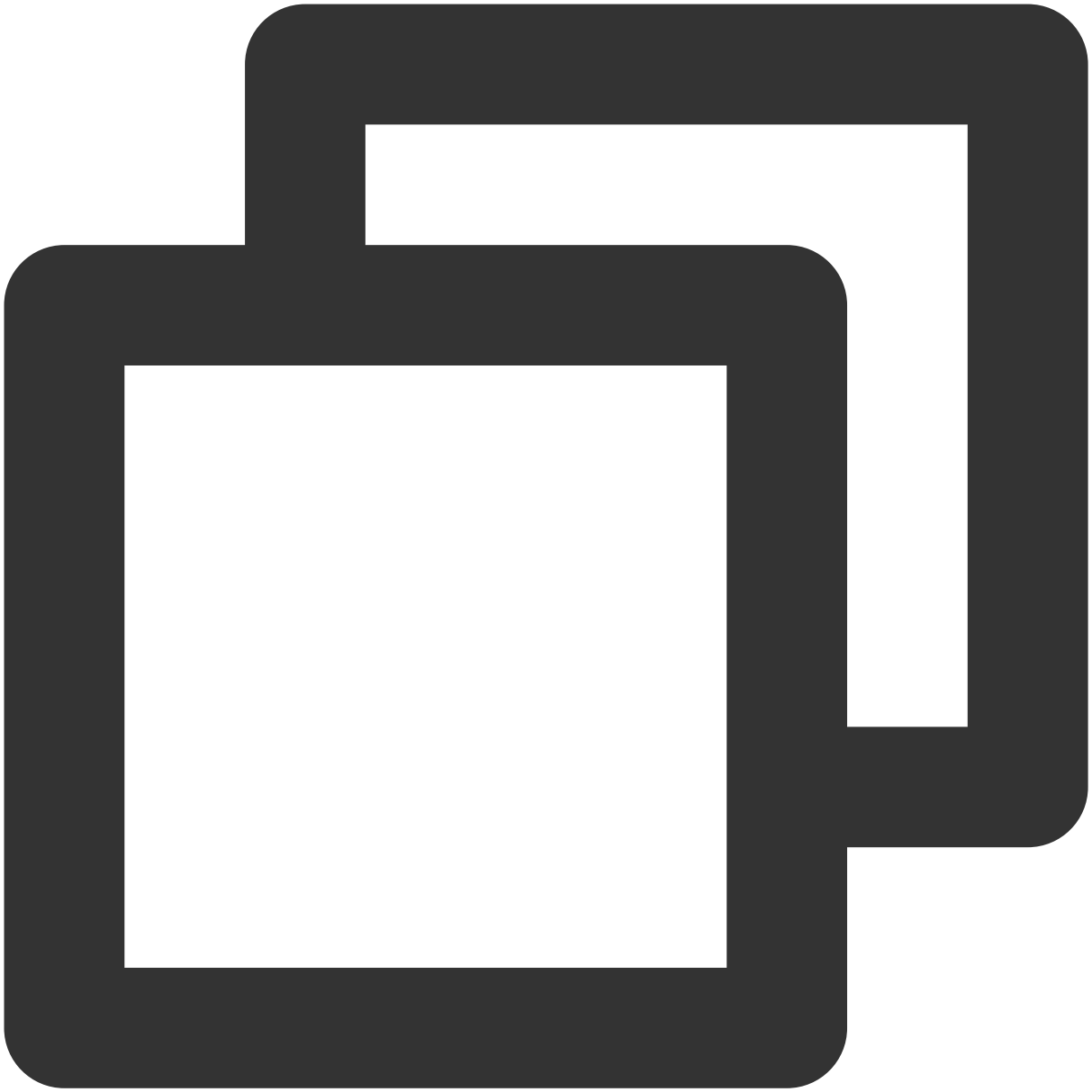
Structure attributes	Type	Note
address	string	The address of the socket sending the message
family	string	The protocol family in use, either IPv4 or IPv6.
Port No.	number	Port No.
size	number	The size of the message, measured in bytes.

.offMessage

This method is used via UDPSocket.offMessage(function listener).

Feature Description: Removes the listener function for events of receiving a message.
Parameter and Description: function listener, the listener function passed in by onMessage. If this parameter is not passed in, all listener functions will be removed.
Sample Code

©2013-2022 Tencent Cloud. All rights reserved.



```
const listener = function (res) { console.log(res) }

UDPSocket.onMessage(listener)
UDPSocket.offMessage(listener) // The same function object as the listener must be
```

.send

This method is used via UDPSocket.send(Object object).

©2013-2022 Tencent Cloud. All rights reserved.

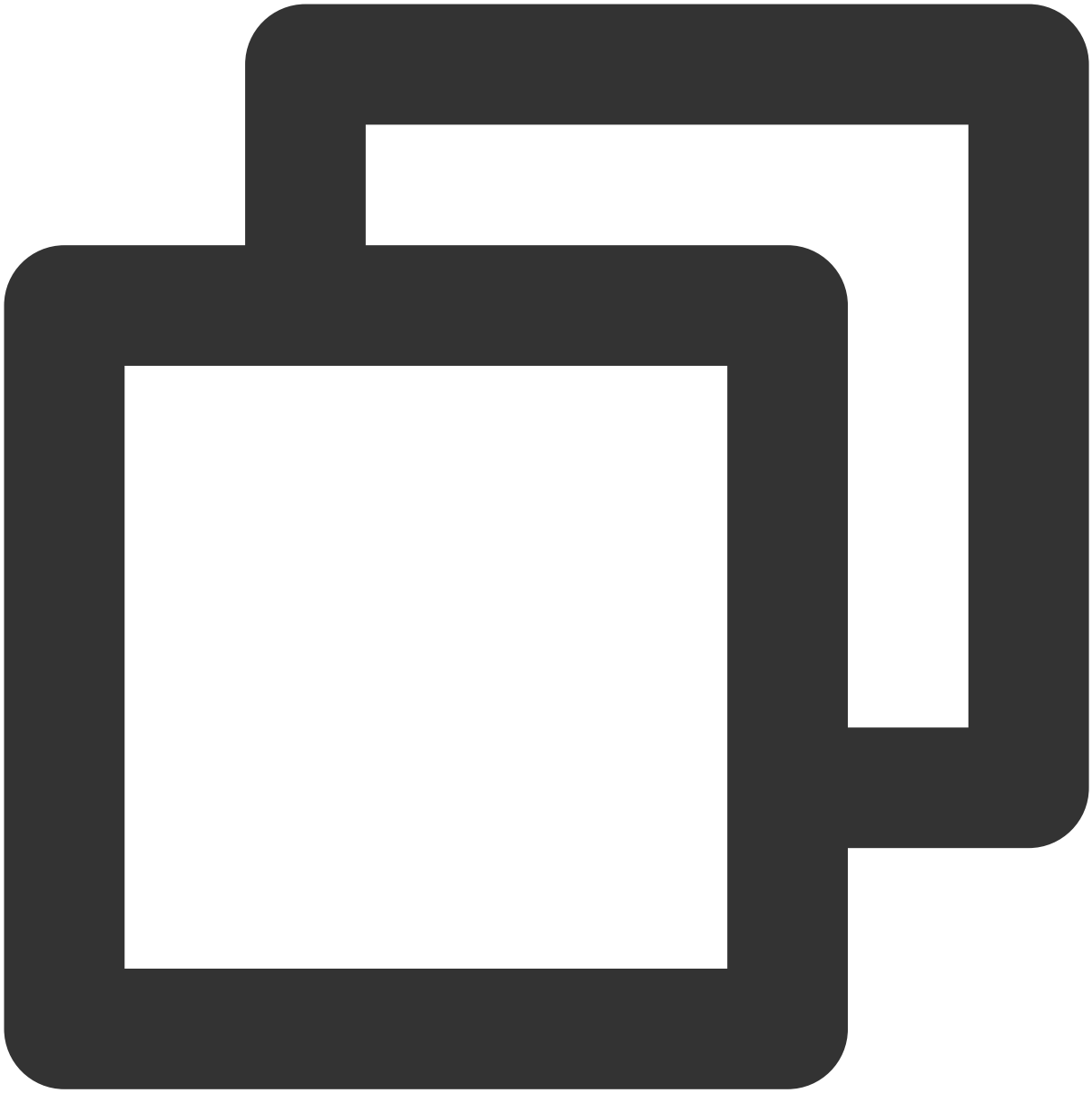
Feature Description: Sends messages to the specified IP and port.

Parameter and Description: Object.

Attribute	Type	Default value	Required	Note
address	string	-	Yes	The address to send the message to.
Port No.	number	-	Yes	Port number for message transmission
message	string/ArrayBuffer	-	Yes	Data to be transmitted
offset	number	0	No	The offset of the data to be sent, valid only when the message is of the ArrayBuffer type.
length	number	message.byteLength	No	The length of the data to be sent, valid only when the message is of the ArrayBuffer type.

Sample Code

©2013-2022 Tencent Cloud. All rights reserved.



```
const udp = wx.createUDPSocket()
  udp.bind()
  udp.send({
    address: '192.168.193.2',
    port: 8848,
    message: 'hello, how are you'
 })
```

.setTTL

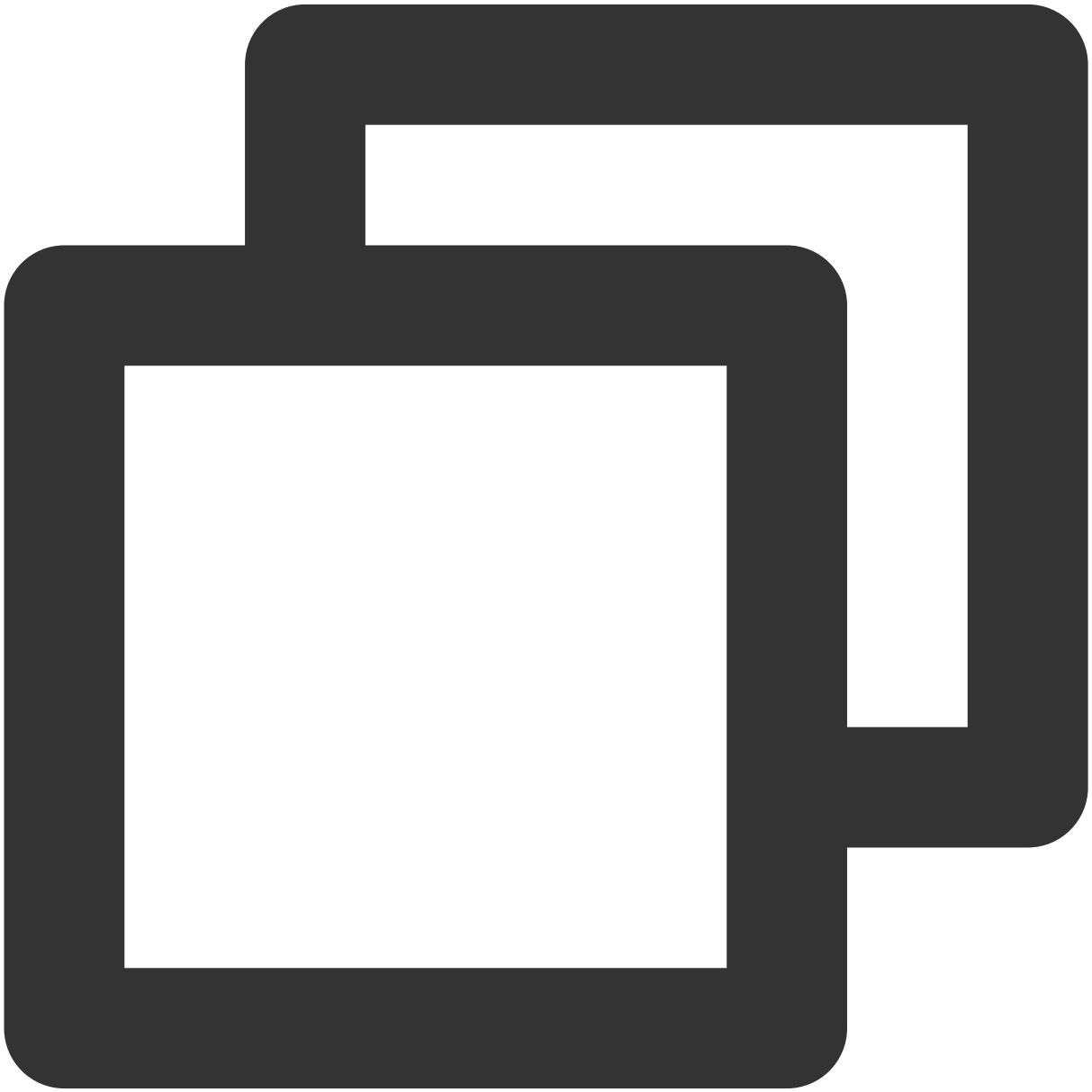
©2013-2022 Tencent Cloud. All rights reserved.

This method is used via UDPSocket.setTTL(number ttl).

Feature Description: Sets the IP_TTL socket option, which determines the maximum number of hops allowed for an IP packet during transmission.

Parameter and Description: number ttl, ranging between 0 and 255.

Sample Code



```
const udp = wx.createUDPSocket()
udp.onListening(function () {
```

©2013-2022 Tencent Cloud. All rights reserved.

```
  udp.setTTL(64)
})
udp.bind()
```

.write

This method is used via UDPSocket.write().

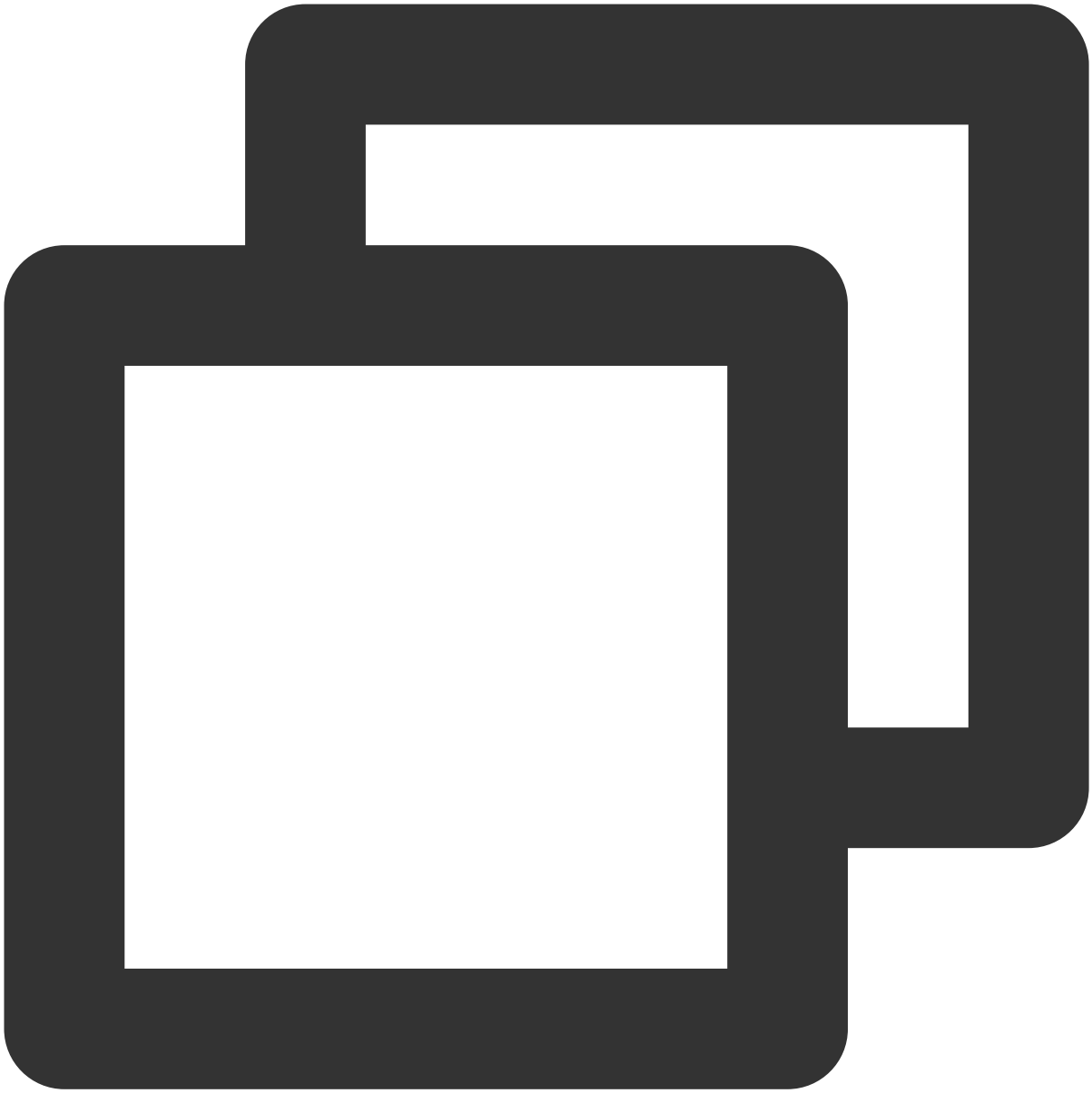
Note:

Even if "connect" is called, it is necessary to input the address and port parameters in this interface.

Feature Description: Usage is the same as the send method, but not different from send if connect is not called beforehand.

Sample Code

©2013-2022 Tencent Cloud. All rights reserved.



```
const udp = wx.createUDPSocket()
udp.bind()
udp.connect({
  address: '192.168.193.2',
  port: 8848,
})
udp.write({
  address: '192.168.193.2',
  port: 8848,
  message: 'hello, how are you'
})
```

©2013-2022 Tencent Cloud. All rights reserved.

©2013-2022 Tencent Cloud. All rights reserved.

mDNS

Last updated：2024-07-12 19:31:37

stopLocalServiceDiscovery

The method of utilizing this API is wx.stopLocalServiceDiscovery(Object object).

Function Description: Cease the exploration of mDNS services.

Parameters and Description: Object object.

Attributes	Types	Default value	Required	Note
success	function	-	Not required	Callback function for successful interface invocation
fail	function	-	Not required	Callback function for unsuccessful interface invocation
complete	function	-	Not required	Callback function executed upon the completion of the interface invocation (both successful and unsuccessful invocations)

object.fail Callback Function Parameters: Object res.

Attributes	Types	Note
errMsg	string	Error message, with the valid value being 'task not found': invoked when stopLocalServiceDiscovery is called while not currently in the search service.

startLocalServiceDiscovery

The usage method for this API is wx.startLocalServiceDiscovery(Object object)

Note:

For iOS, the integration of the TMFMiniAppExtMDNS extension library is required to use the mDNS related interfaces.

©2013-2022 Tencent Cloud. All rights reserved.

wx.startLocalServiceDiscovery is a performance-consuming action. It will automatically stop after 30 seconds and execute the callback function registered with wx.onLocalServiceDiscoveryStop.
After invoking wx.startLocalServiceDiscovery, the next wx.startLocalServiceDiscovery can only be initiated once this search action has ceased. Actions to halt this search include invoking wx.stopLocalServiceDiscovery and the system automatically stopping the search after 30 seconds.
Function Description: Initiates the search for mDNS services within the local area network. The results of the search will be returned via the wx.onLocalService* event.
Parameters and Description: Object object.

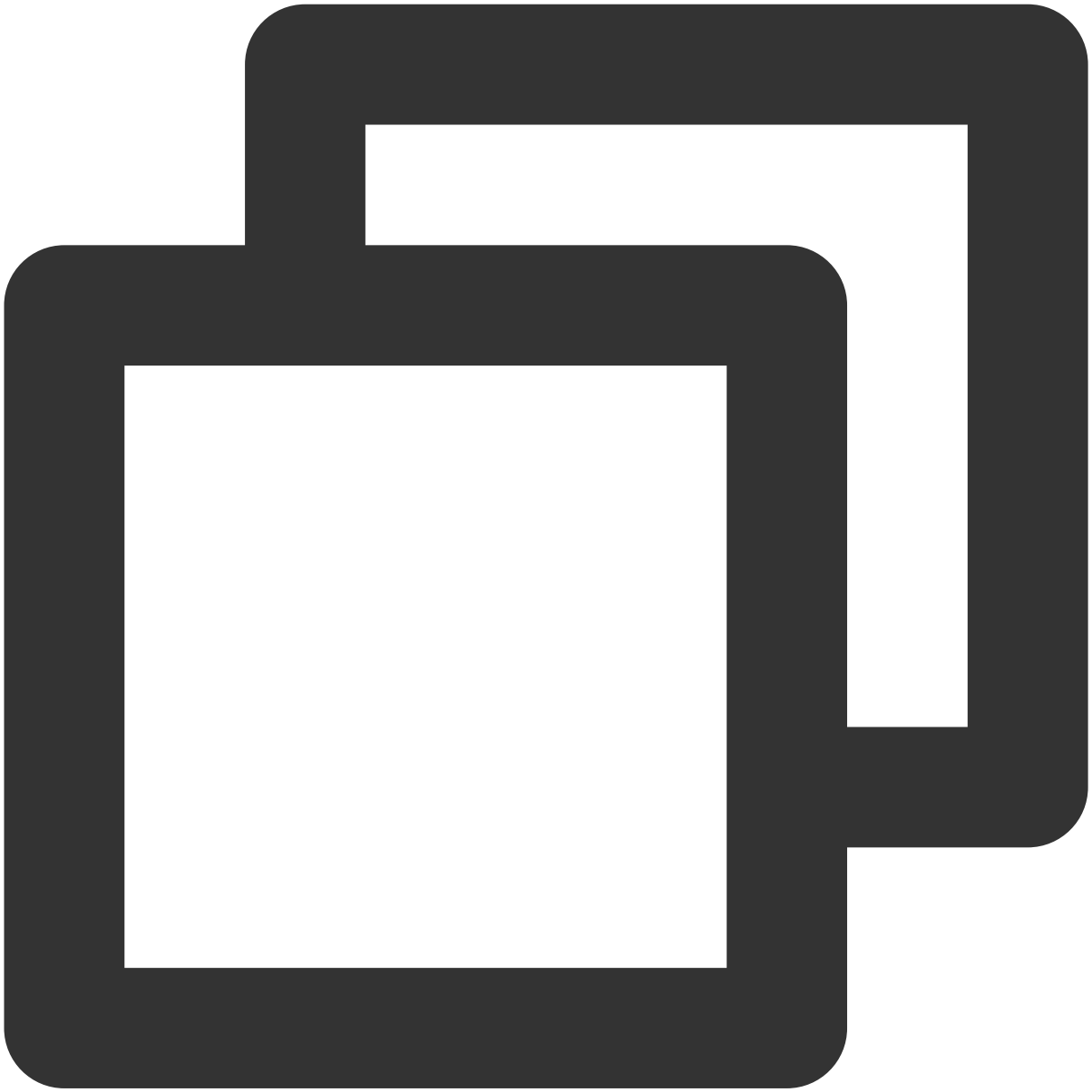
Attributes	Types	Default value	Required	Note
serviceType	string	-	Supported	The type of service to be searched for
success	function	-	Not required	Callback function for successful interface invocation
fail	function	-	Not required	Callback function for unsuccessful interface invocation
complete	function	-	Not required	Callback function executed upon the completion of the interface invocation (both successful and unsuccessful invocations)

object.fail callback function parameters: Object res.

Attributes	Types	Note
errMsg	string	Error message, valid values are Invalid parameter: serviceType is empty. Scan task already exists: startLocalServiceDiscovery is called again while the search initiated by the current startLocalServiceDiscovery has not stopped.

Example Code

©2013-2022 Tencent Cloud. All rights reserved.



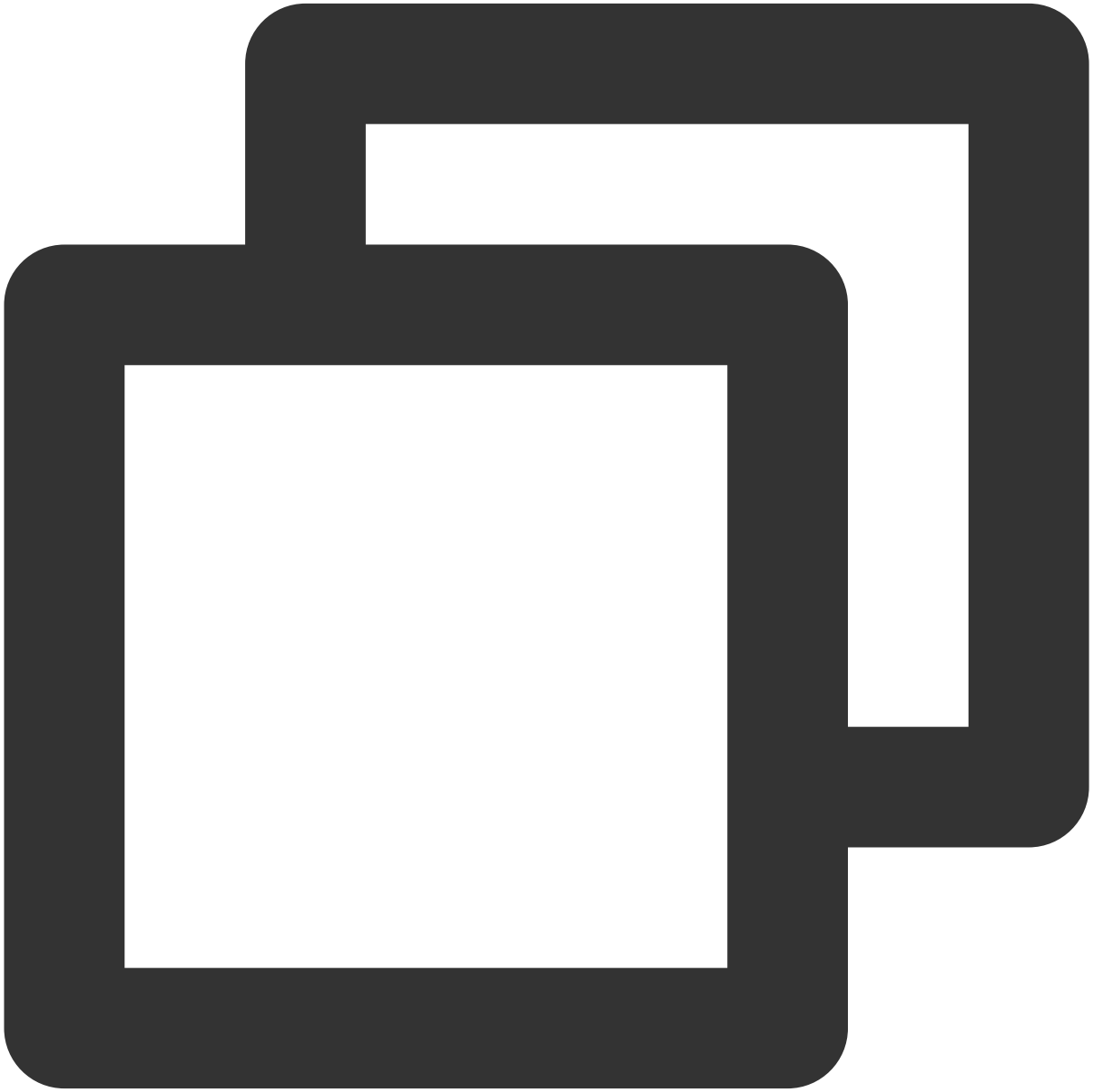
```
wx.startLocalServiceDiscovery({
    // There is a service of type _http._tcp. in the local area network connected t
    serviceType: '_http._tcp.',
    success: console.log,
    fail: console.log
})
```

offLocalServiceResolveFail

©2013-2022 Tencent Cloud. All rights reserved.

The usage method for this API is wx.offLocalServiceResolveFail(function listener).

Function Description: Removes the listener function for the event of mDNS service resolution failure.
Parameters and Description: function listener, the listening function passed in by onLocalServiceResolveFail. If this parameter is not passed, all listening functions will be removed.
Sample Code:



```
const listener = function (res) { console.log(res) }
```

©2013-2022 Tencent Cloud. All rights reserved.

```
wx.onLocalServiceResolveFail(listener)
wx.offLocalServiceResolveFail(listener) // The same function object as the listener
```

onLocalServiceResolveFail

The usage method for this API is wx.onLocalServiceResolveFail(function listener).

Function Description: Monitors the event of mDNS service resolution failure.
Parameters and Description: function listener, the listener function for the event that the mDNS service fails to resolve, with the following arguments Object res:

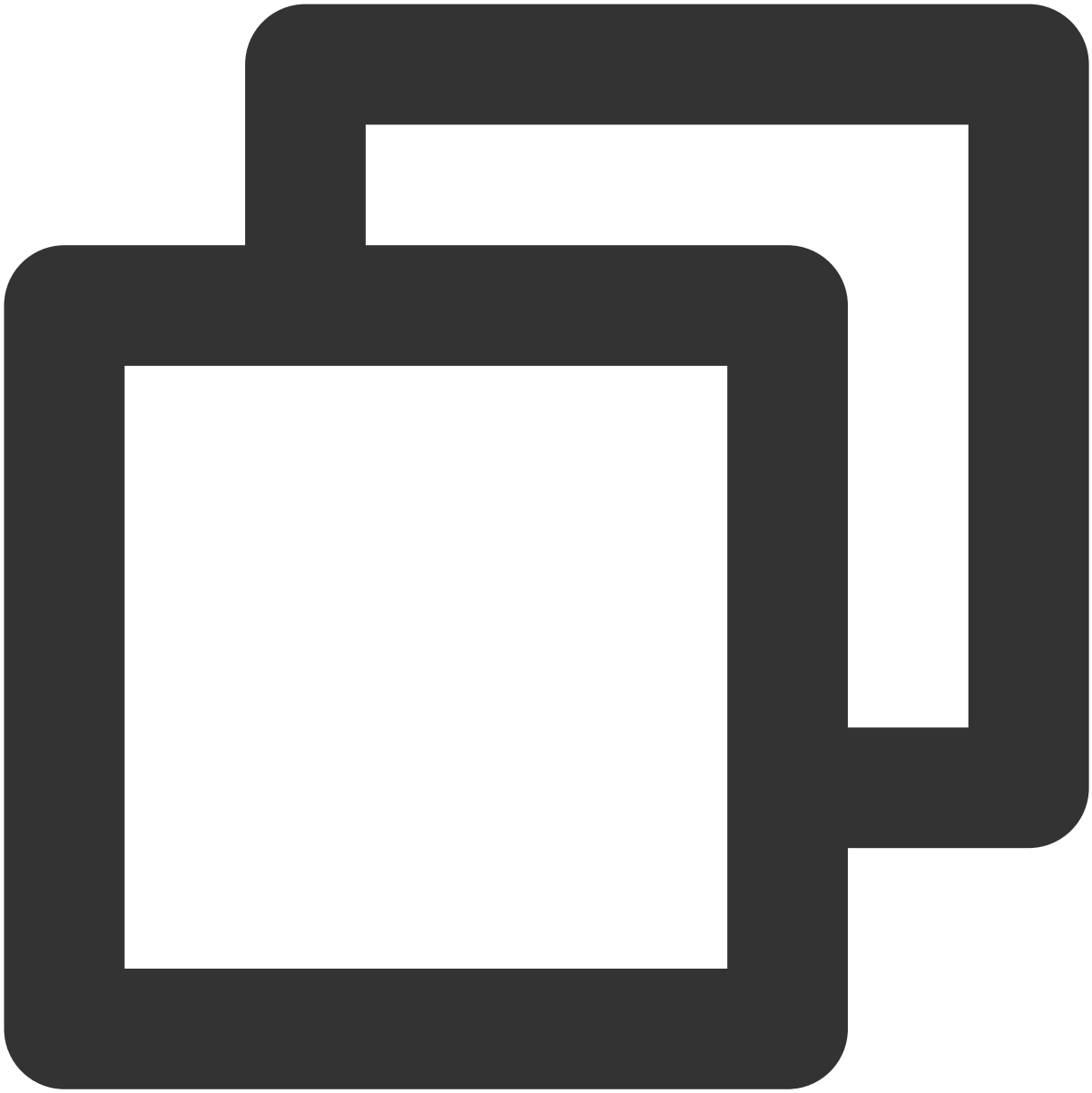
Attributes	Types	Note
serviceType	string	Service Type
serviceName	string	Service Name

offLocalServiceLost

The usage method for this API is wx.offLocalServiceLost(function listener).

Function Description: Removes the listening function for the event of mDNS service departure.
Parameters and Description: function listener, the listener passed in by onLocalServiceLost. If you don't pass this parameter, you can remove all the listener functions.
Sample Code:

©2013-2022 Tencent Cloud. All rights reserved.



```
const listener = function (res) { console.log(res) }

wx.onLocalServiceLost(listener)
wx.offLocalServiceLost(listener) // The same function objct as used during listenin
```

onLocalServiceLost

©2013-2022 Tencent Cloud. All rights reserved.

The usage method for this API is wx.onLocalServiceLost(function listener).

Function Description: Monitors the event of mDNS service departure.
Parameters and Description: function listener, the listener function for the event that the mDNS service leaves, with the following arguments Object res:

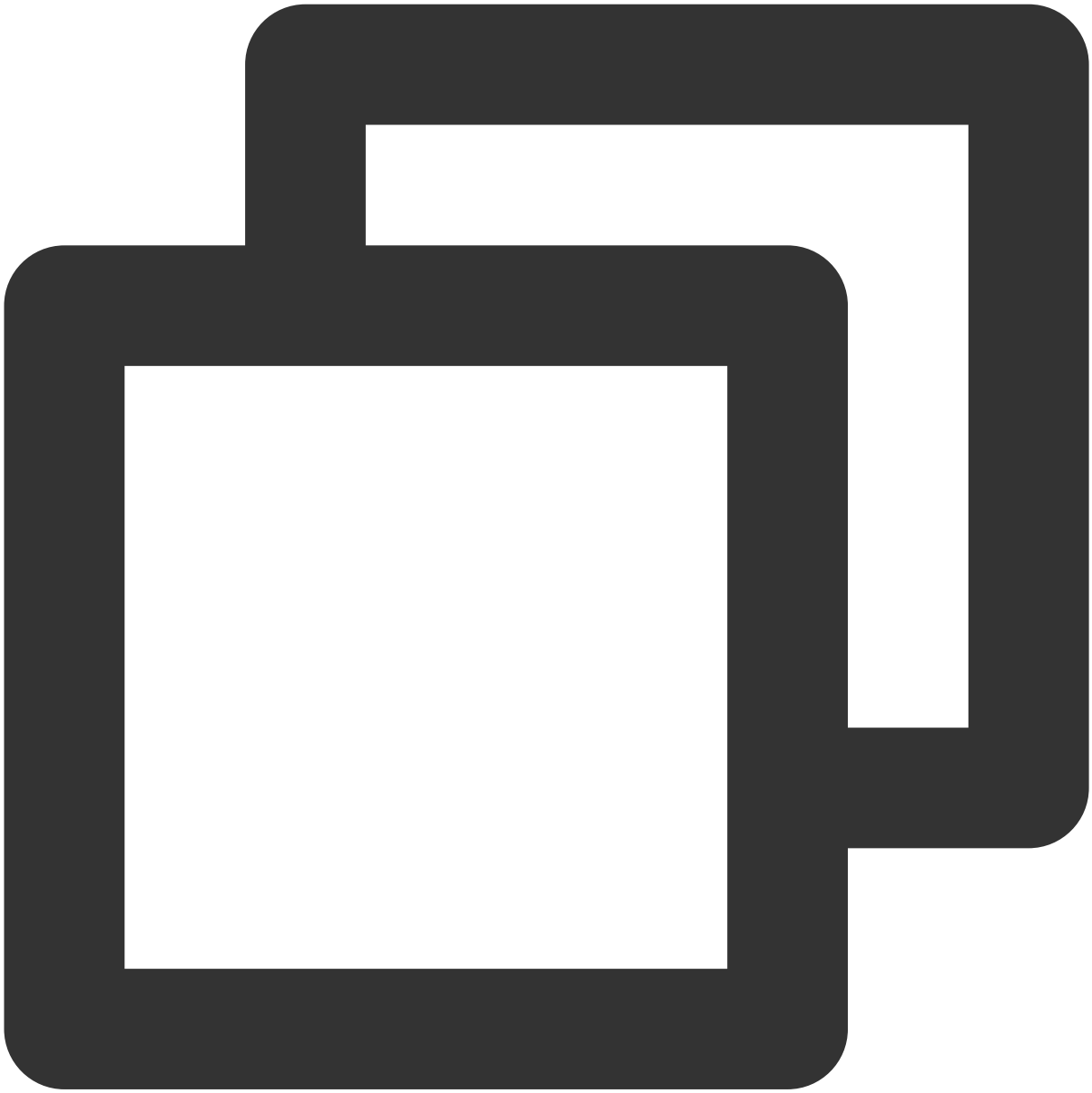
Attributes	Types	Note
serviceType	string	Service Type
serviceName	string	Service Name

offLocalServiceFound

The usage method for this API is wx.offLocalServiceFound(function listener).

Function Description: Removes the listening function for the event of mDNS service discovery.
Parameters and Description: function listener, onLocalServiceFound The listener function passed in. If this parameter is not passed, all listener functions are removed.
Sample Code:

©2013-2022 Tencent Cloud. All rights reserved.



```
const listener = function (res) { console.log(res) }

wx.onLocalServiceFound(listener)
wx.offLocalServiceFound(listener) // The same function object as the one used durin
```

onLocalServiceFound

©2013-2022 Tencent Cloud. All rights reserved.

The usage method for this API is wx.onLocalServiceFound(function listener).

Function Description: Monitors the event of mDNS service discovery.
Parameters and Description: function listener, the listener function for events discovered by the mDNS service, with the following arguments Object res:

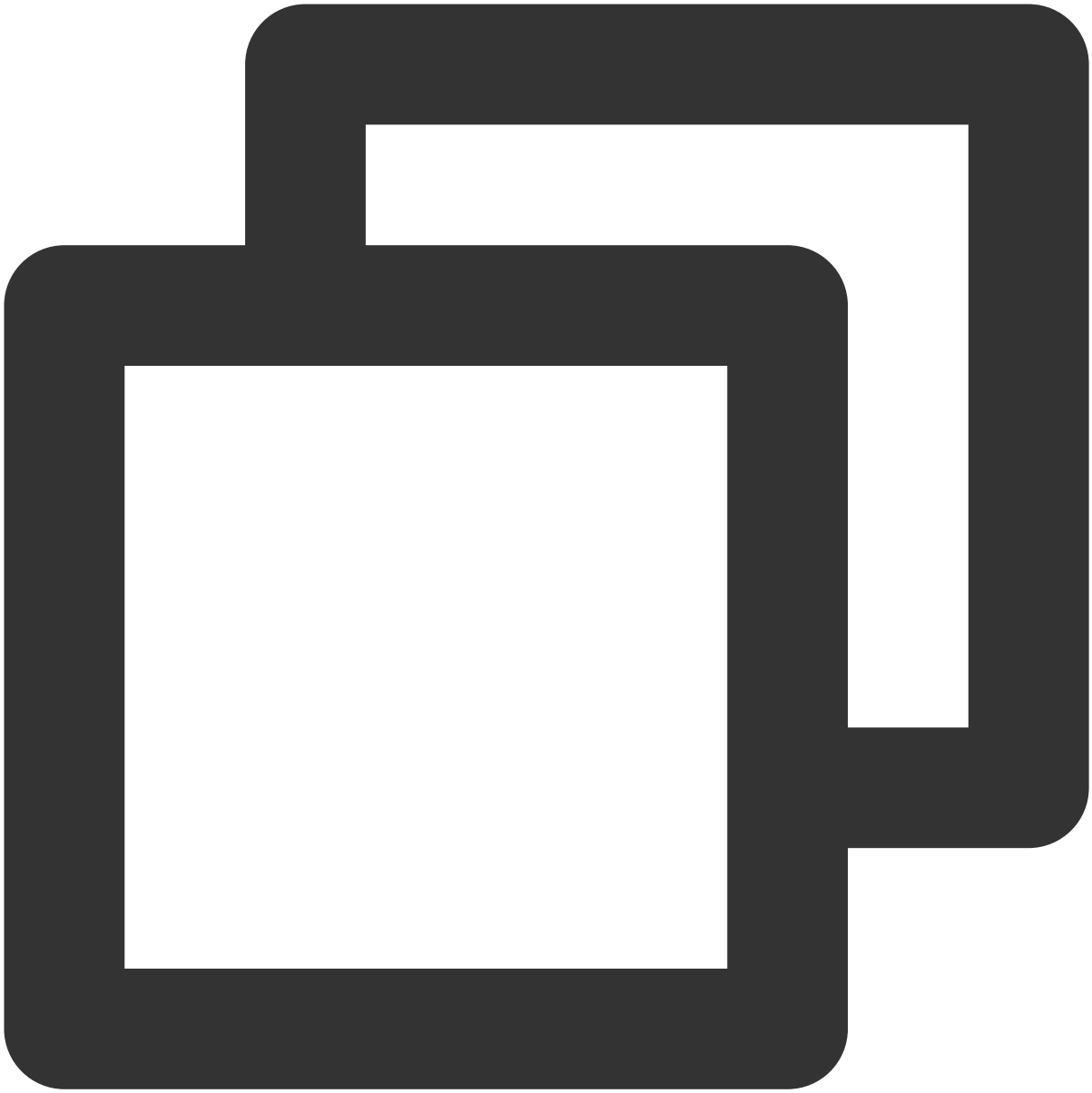
Attributes	Types	Note
serviceType	string	Service Type
serviceName	string	Service Name
ip	string	The IP address of the service.
port	number	The port of the service.

offLocalServiceDiscoveryStop

The usage method for this API is wx.offLocalServiceDiscoveryStop(function listener).

Function Description: Removes the listening function for the event of mDNS service stop discovery.
Parameters and Description: function listener, onLocalServiceDiscoveryStop The listener function passed in. If this parameter is not passed, all listener functions are removed.
Sample Code:

©2013-2022 Tencent Cloud. All rights reserved.



```
const listener = function (res) { console.log(res) }

wx.onLocalServiceDiscoveryStop(listener)
wx.offLocalServiceDiscoveryStop(listener) // The same function object as the listen
```

onLocalServiceDiscoveryStop

©2013-2022 Tencent Cloud. All rights reserved.

The usage method for this API is wx.onLocalServiceDiscoveryStop(function listener).

Function Description: Monitors the event of mDNS service stop discovery.

Parameters and Description: function listener, the listener function for the event that the mDNS service stops searching.

©2013-2022 Tencent Cloud. All rights reserved.

Initiate request

Last updated：2024-07-12 19:31:16

request

This API is used via RequestTask wx.request(Object object).

Feature Description: Initiates an HTTPS network request. Read relevant instructions before using this feature.

Parameter and Description: Object.

Attribute	Type	Default value	Required	Description
url	string	-	Yes	Developer Server Interface Address
data	string/object/ArrayBuffer	-	No	Parameters of the Request
header	Object	-	No	Sets the request's header, but Referer is not allowed to be set in the header. The default content-type is application/json.
method	string	GET	No	HTTP request method
dataType	string	json	No	Format of returned data
responseType	string	text	No	Response Data Type
success	function	-	No	Callback Function of Successful Interface Call
fail	function	-	No	Callback Function of Failing Interface Call
complete	function	-	No	Callback function executed upon the completion of the interface invocation (both successful and unsuccessful invocations)

Valid values of object.method

Value	Description

©2013-2022 Tencent Cloud. All rights reserved.

OPTIONS	HTTP Request OPTIONS
GET	HTTP Request GET
HEAD	HTTP Request HEAD
POST	HTTP Request POST
PUT	HTTP Request PUT
DELETE	HTTP Request DELETE
TRACE	HTTP Request TRACE
CONNECT	HTTP Request CONNECT

Valid values of object.dataType

Value	Description
json	The returned data is in JSON format. upon receipt, the data will undergo a JSON.parse operation.
Other	The data will not undergo a JSON.parse operation.

Valid values of object.responseType

Value	Description
text	Response data is text.
arraybuffer	The response data is in ArrayBuffer format.

object.success callback function:

Parameter: Object res

Attribute	Type	Description
data	string/Object/Arraybuffer	Data returned from the developer's server.
statusCode	number	HTTP status code returned from the developer's server.
header	Object	HTTP Response Header returned from the developer's server.
cookies	Array.<string>	Cookies returned by the developer server, formatted as an array of strings

©2013-2022 Tencent Cloud. All rights reserved.

object.fail callback function

Parameters: Object err

Attribute	Type	Description
errMsg	String	error message

Return Value: RequestTask, the request task object.

Data parameter description

The final data sent to the server is of String type. If the data passed in is not of String type, it will be converted to String. The conversion rules are as follows:

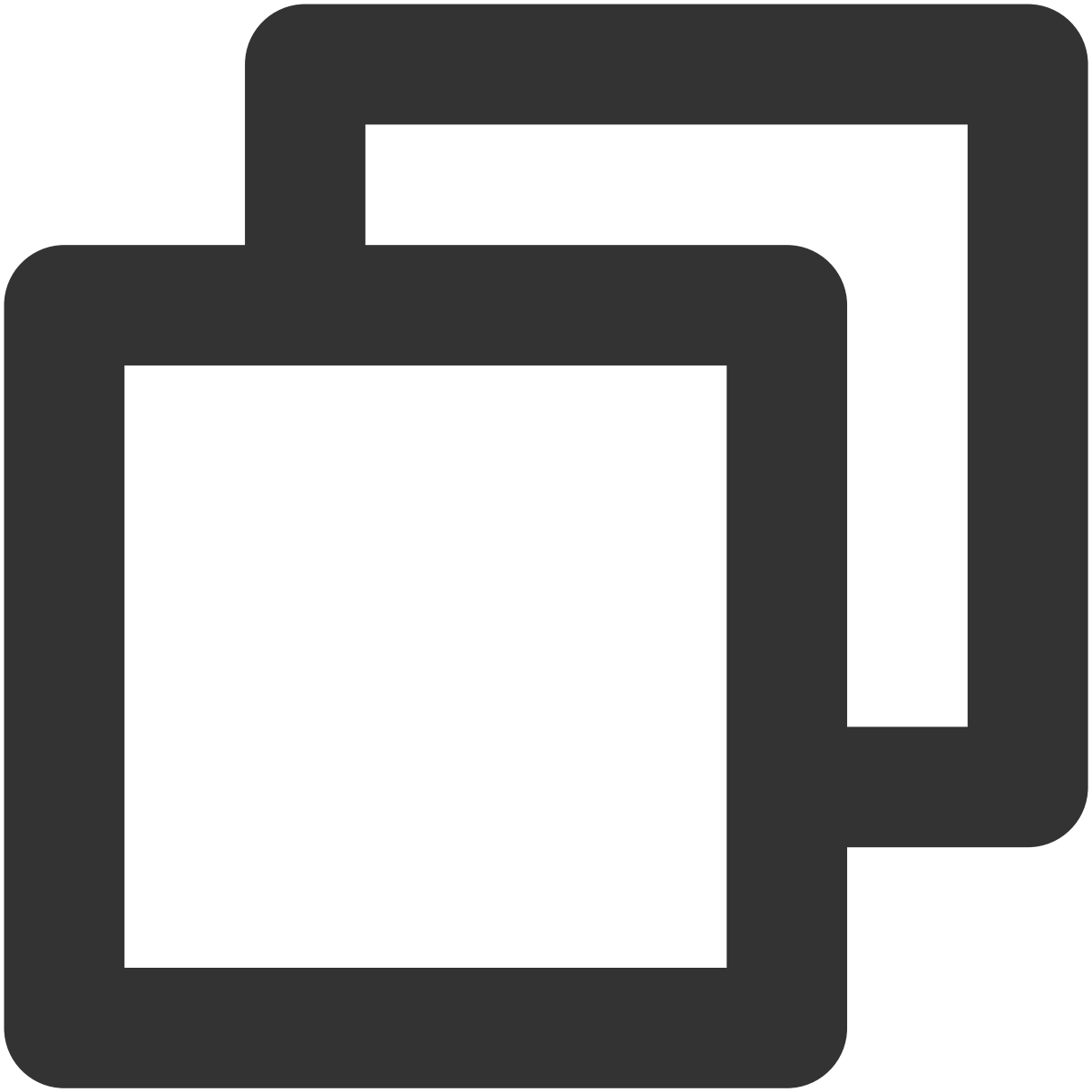
For data pertaining to the `GET` method, the data will be converted into a query string (`encodeURIComponent(k)=encodeURIComponent(v)&encodeURIComponent(k)=encodeURIComponent(v)...`).

For data associated with the `POST` method and where `header['content-type']` is `application/json` , the data will undergo JSON serialization.

For data associated with the `POST` method and where `header['content-type']` is `application/x-www-form-urlencoded` , the data will be converted into a query string `(encodeURIComponent(k)=encodeURIComponent(v)&encodeURIComponent(k)=encodeURIComponent(v)...)` .

Sample Code

©2013-2022 Tencent Cloud. All rights reserved.



```
wx.request({
  url: 'test.php', // This is merely an example and not an actual interface address
  data: {
    x: '',
    y: ''
  },
  header: {
    'content-type': 'application/json' // Default value
  },
  success(res) {
    console.log(res.data)
```

©2013-2022 Tencent Cloud. All rights reserved.

```
    }
})
```

RequestTask

.abort

This method is used via RequestTask.abort().

Feature Description: Aborts the request task.

.onChunkReceived

This method is used via RequestTask.onChunkReceived(function listener).

Feature Description: Monitors the Transfer-Encoding Chunk Received event. It is triggered when a new chunk is received.

Parameter and Description: function listener, the listener function for the Transfer-Encoding Chunk Received event, with the following arguments Object res:

Attribute	Type	Description
res	Object	The Response every time when a new chunk is returned from the developer's server.

Structure attribute of res

Structure attributes	Type	Description
data	ArrayBuffer	Returned chunk buffer

.offChunkReceived

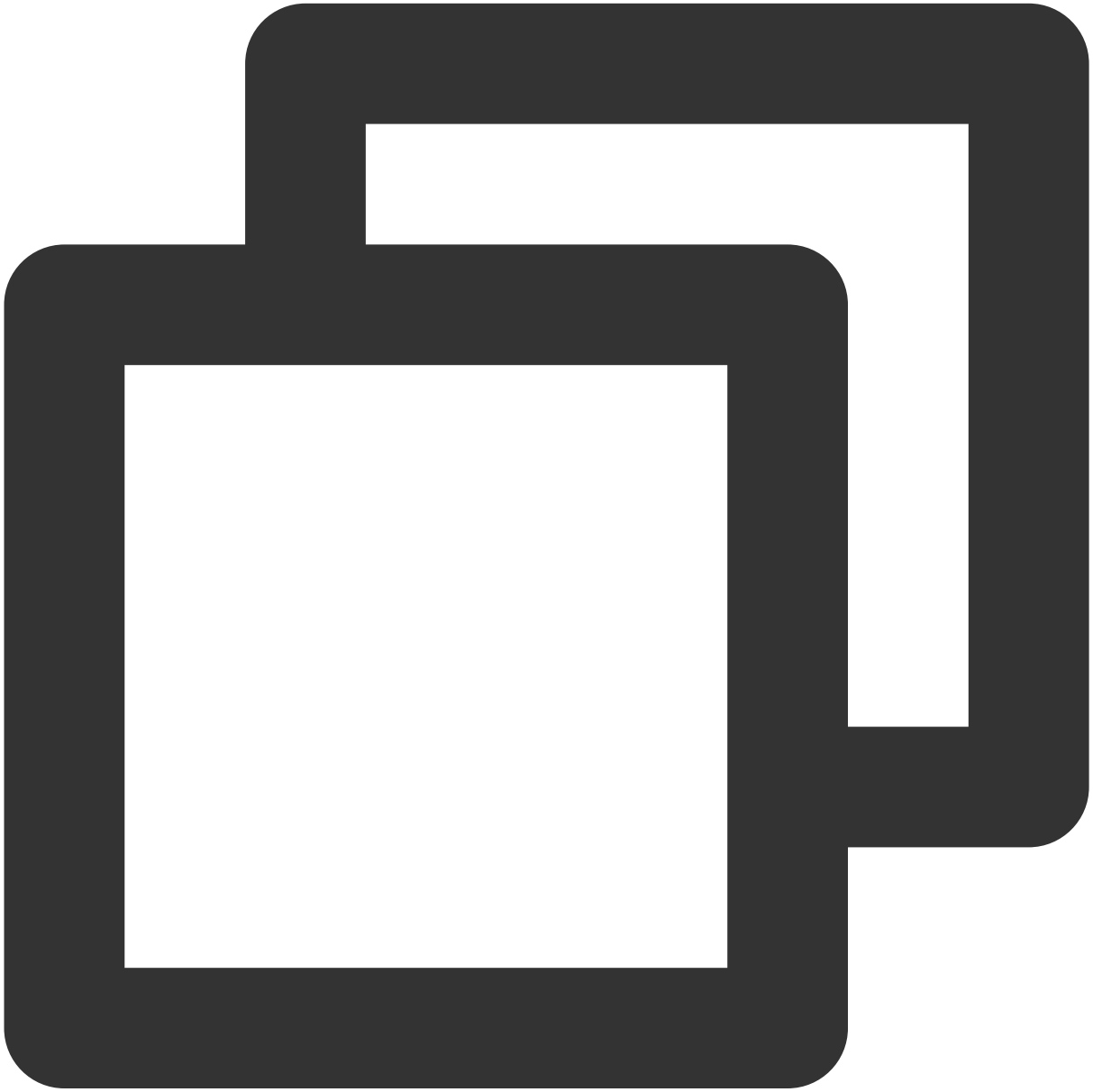
This method is used via RequestTask.offChunkReceived(function listener).

Feature Description: Removes the listener function for Transfer-Encoding Chunk Received events.

©2013-2022 Tencent Cloud. All rights reserved.

Parameter and Description: function listener, the listener function passed in by onChunkReceived. If this parameter is not passed in, all listener functions will be removed.

Sample Code



```
const listener = function (res) { console.log(res) }

RequestTask.onChunkReceived(listener)
RequestTask.offChunkReceived(listener) // The same function object as the listener
```

.onHeadersReceived

©2013-2022 Tencent Cloud. All rights reserved.

This method is used via RequestTask.onHeadersReceived(function listener).

Feature Description: Listens for HTTP Response Header events. It will precede the request completion event.
Parameter and Description: function listener, HTTP Response Header event listener function.

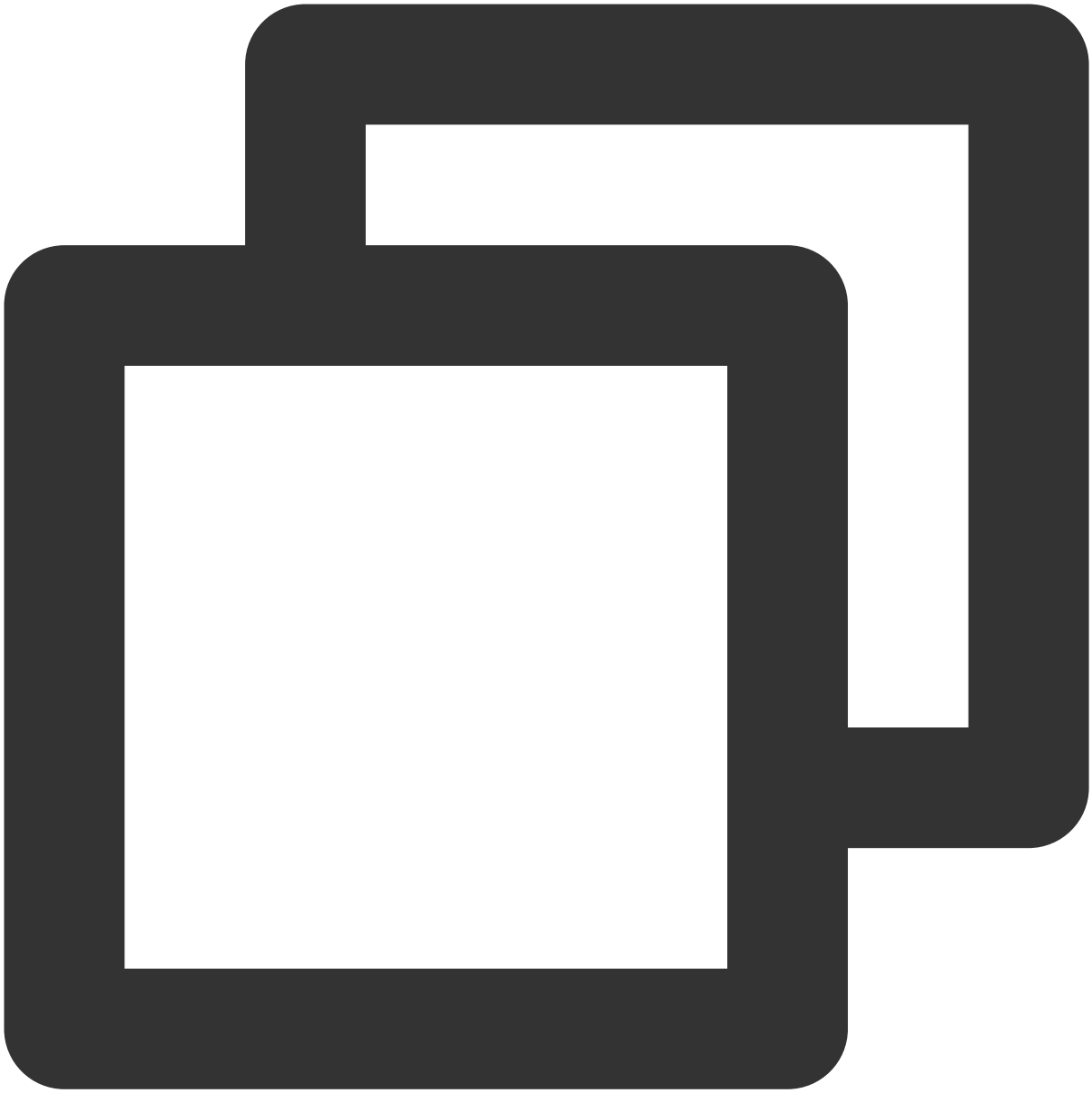
Attributes	Type	Description
header	Object	HTTP Response Header returned by the developer server
statusCode	Number	HTTP status code returned by the developer server (currently the statusCode field is not returned by the developer tools, you can view it on the real machine, it will be supported in the future)
cookies	Array.<string>	Cookies returned by the developer server, formatted as an array of strings

.offHeadersReceived

This method is used via RequestTask.offHeadersReceived(function listener).

Feature Description: Removes the listener function for the HTTP Response Header event.
Parameter and Description: function listener, onHeadersReceived The listener function passed in, if this parameter is not passed, then all listener functions will be removed.
Sample Code

©2013-2022 Tencent Cloud. All rights reserved.



```
const listener = function (res) { console.log(res) }

RequestTask.onHeadersReceived(listener)
RequestTask.offHeadersReceived(listener) // need to pass in the same function objec
```

©2013-2022 Tencent Cloud. All rights reserved.

Upload

Last updated：2024-07-12 19:29:35

uploadFile

This API is used via UploadTask wx.uploadFile(Object object).

Feature Description: Uploads local resources to the server. The client initiates an HTTPS POST request, where the `content-type` is `multipart/form-data`. Prior to usage, read relevant instructions.

Parameter and Description: Object.

Attribute	Type	Default value	Required	Description
url	string	-	Yes	Developer Server Address
filePath	string	-	Yes	Path to the file resource to be uploaded (local path)
name	string	-	Yes	The corresponding key for the file, through which developers can access the binary content of the file on the server side.
header	object	-	No	HTTP request's Header; setting Referer within the Header is not allowed.
formData	object	-	No	Additional form data in the HTTP request
success	function	-	No	Callback Function of Successful Interface Call
fail	function	-	No	Callback Function of Failing Interface Call
complete	function	-	No	Callback function executed upon the completion of the interface invocation (both successful and unsuccessful invocations)

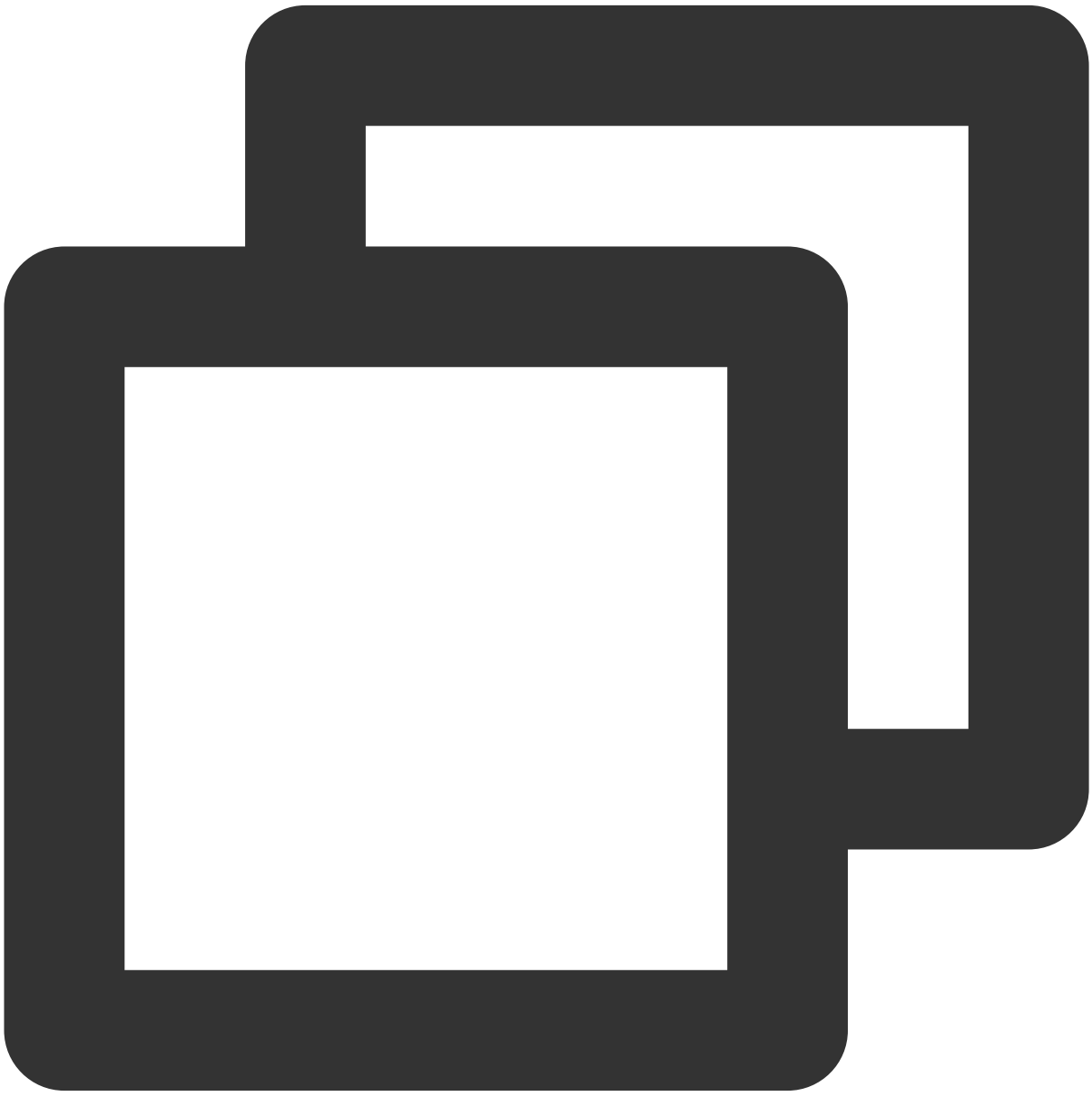
Parameters for object.success callback function: Object res.

Attribute	Type	Description
data	string	Data returned from the developer's server.
statusCode	number	HTTP status code returned from the developer's server.

©2013-2022 Tencent Cloud. All rights reserved.

Return Value: UploadTask, an object that can monitor the progress of the upload and cancel the upload.

Sample Code



```
wx.chooseImage({
  success (res){
    const tempFilePaths = res.tempFilePaths
    wx.uploadFile({
      url: 'https://example.weixin.qq.com/upload', // This is merely an example and
      filePath: tempFilePaths[0],
      name: 'file',
```

©2013-2022 Tencent Cloud. All rights reserved.

```
        formData:{
          'user': 'test'
        },
        success (res){
          const data = res.data
          //do something
        }
      })
    }
  })
```

UploadTask

.abort

This method is used via UploadTask.abort().

Feature Description: Aborts the upload task.

.onProgressUpdate

This method is used via UploadTask.onProgressUpdate(function listener).

Feature Description: Monitors upload progress change events.
Parameter and Description: function listener, listener for upload progress change events, arguments Object.res are as follows.

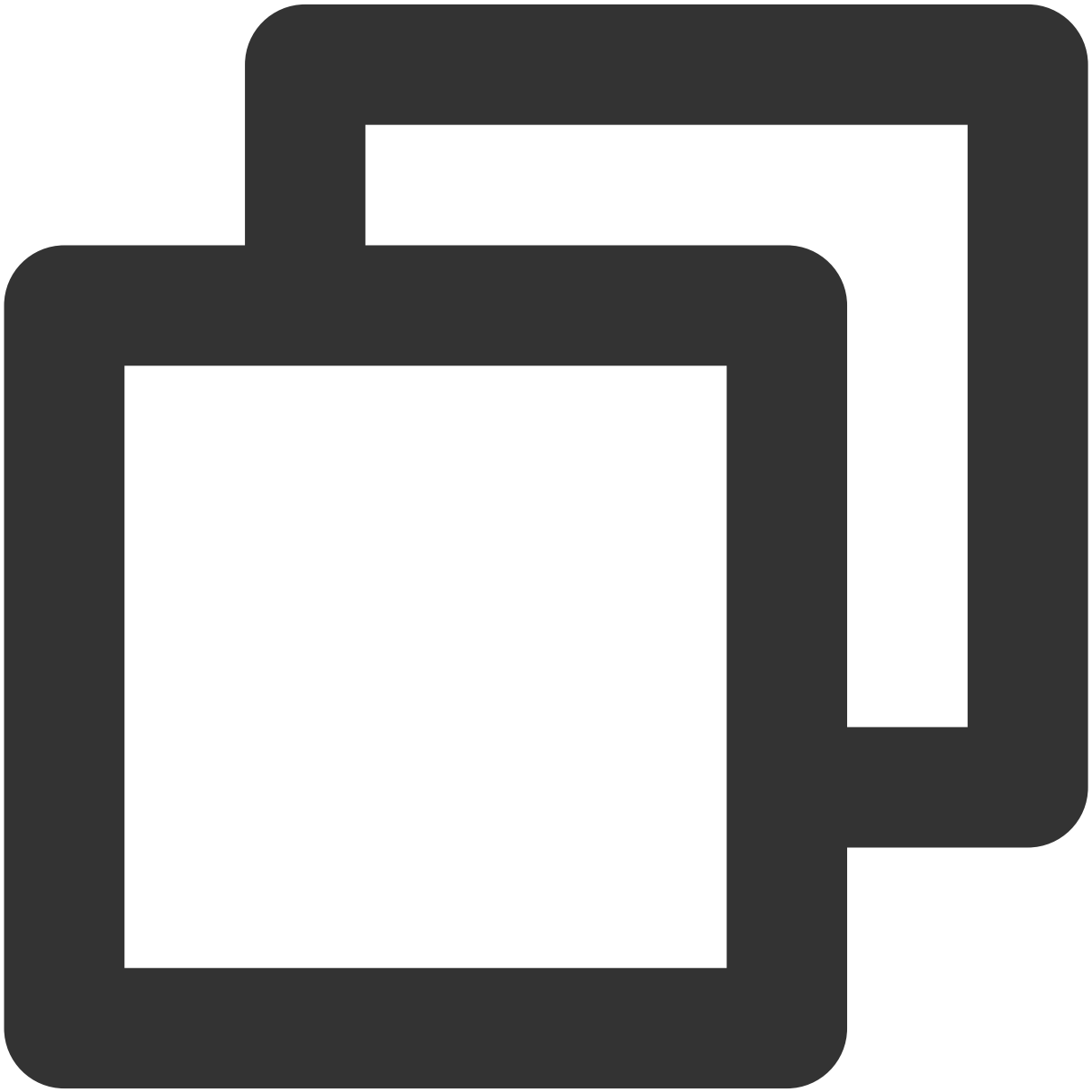
Attribute	Type	Description
progress	number	Download progress percentage
totalBytesWritten	number	Length of the data already downloaded, measured in Bytes.
totalBytesExpectedToWrite	number	Total length of the data expected to be downloaded, measured in bytes.

.offProgressUpdate

©2013-2022 Tencent Cloud. All rights reserved.

This method is used via UploadTask.offProgressUpdate(function listener).

Feature Description: Removes the listener function for upload progress change events.
Parameters and Description: function listener, the listener function passed in by onProgressUpdate. If this parameter is not passed in, all listener functions will be removed.
Sample Code



```
const listener = function (res) { console.log(res) }
```

©2013-2022 Tencent Cloud. All rights reserved.

```
DownloadTask.onProgressUpdate(listener)
DownloadTask.offProgressUpdate(listener) // The same function object as the listene
```

.onHeadersReceived

This method is used via UploadTask.onHeadersReceived(function listener).

Feature Description: Monitors HTTP Response Header events. This occurs earlier than the request completion event.

Parameter and Description: function listener, the listener function for HTTP Response Header events.

Attribute	Type	Description
header	Object	HTTP Response Header returned from the developer's server.

.offHeadersReceived

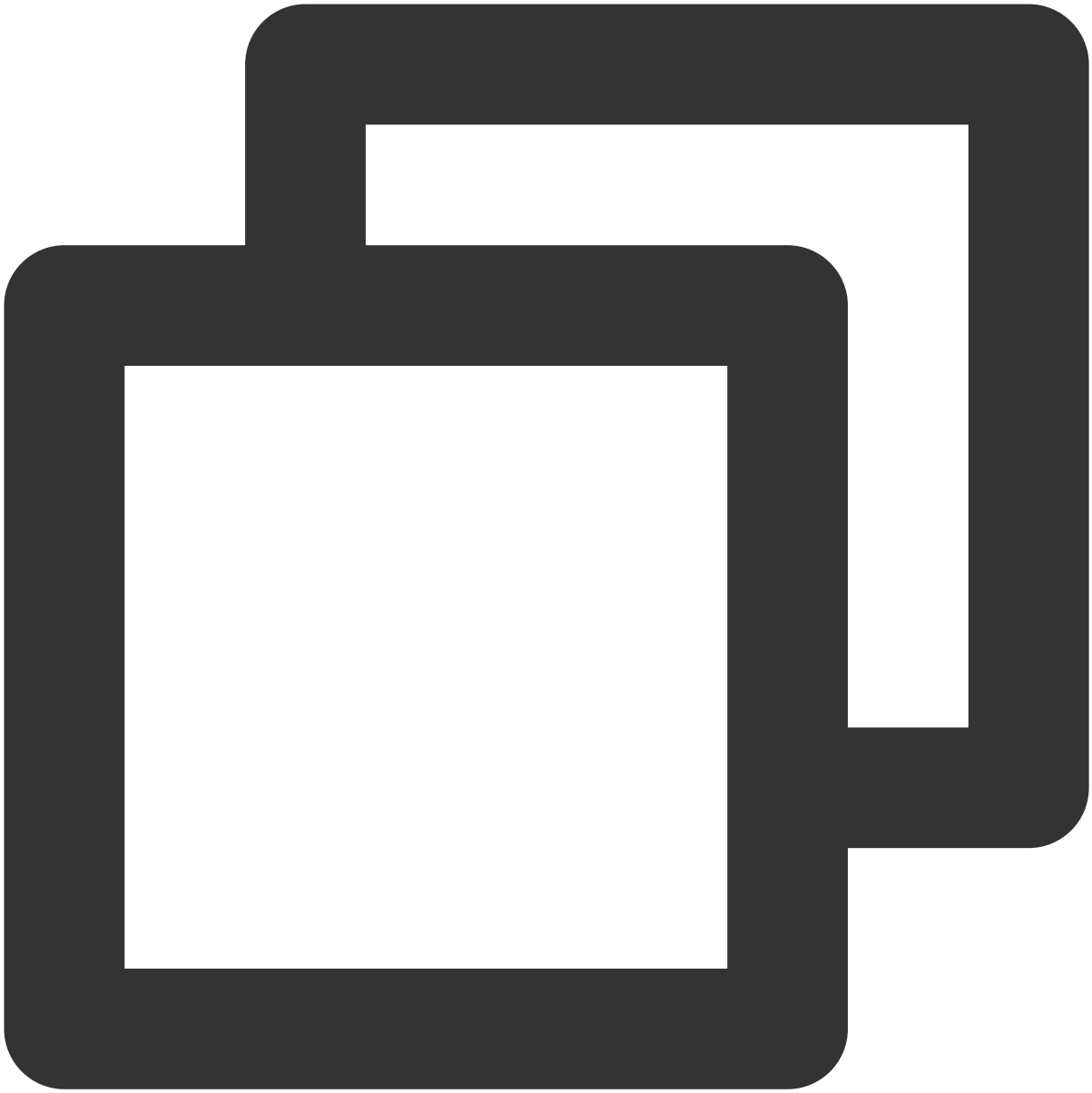
This method is used via UploadTask.offHeadersReceived(function listener).

Feature Description: Removes the listener function for HTTP Response Header events.

Parameters and Description: function listener, the listener function passed in by onHeadersReceived. If this parameter is not passed in, all listener functions will be removed.

Sample Code

©2013-2022 Tencent Cloud. All rights reserved.

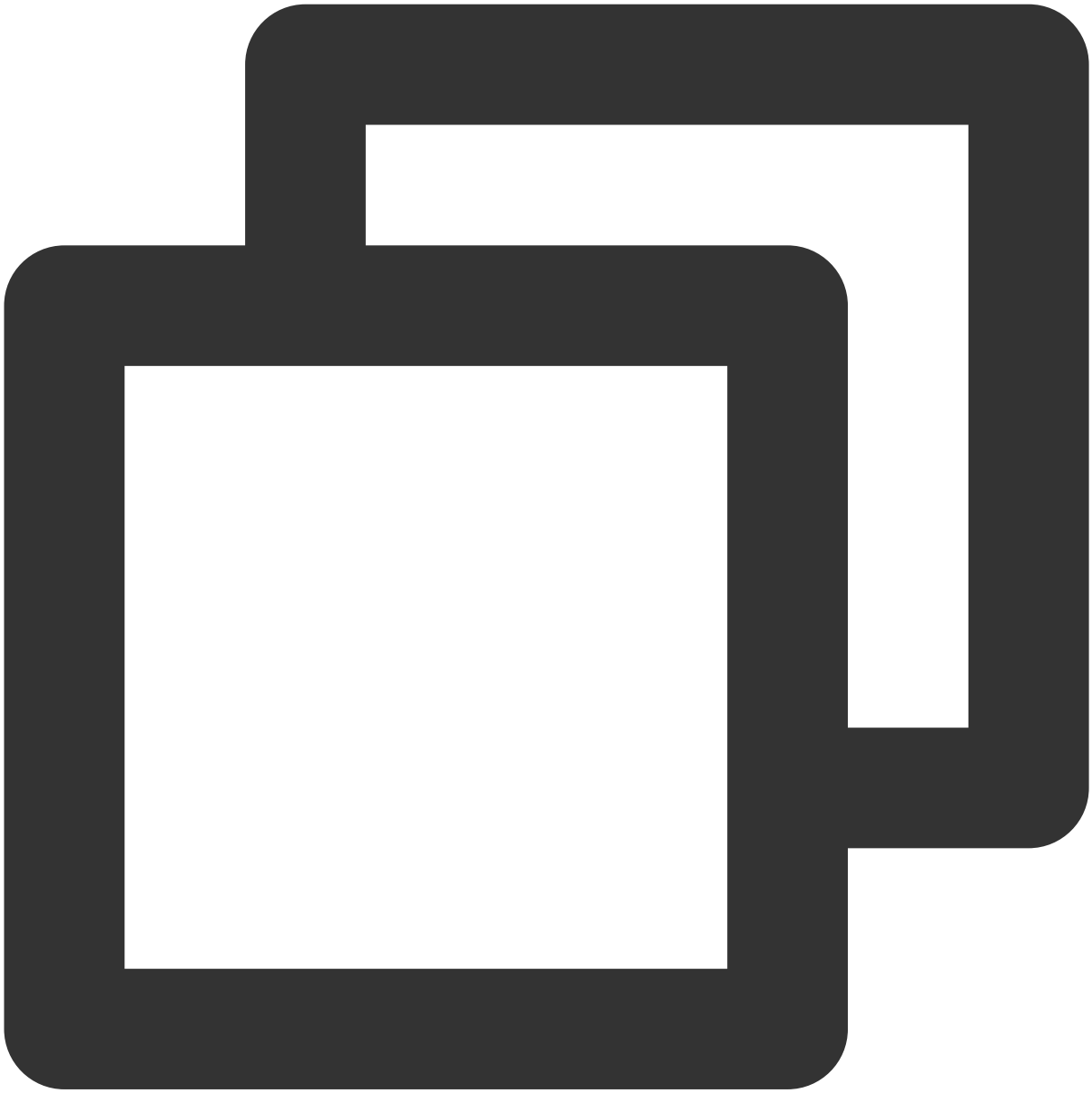


```
const listener = function (res) { console.log(res) }

UploadTask.onHeadersReceived(listener)
UploadTask.offHeadersReceived(listener) // The same function object as the listener
```

Sample code

©2013-2022 Tencent Cloud. All rights reserved.



```
const uploadTask = wx.uploadFile({
  url: 'http://example.weixin.qq.com/upload', // This is merely an example and not
  filePath: tempFilePaths[0],
  name: 'file',
  formData:{
    'user': 'test'
  },
  success (res){
    const data = res.data
    //do something
  }
```

©2013-2022 Tencent Cloud. All rights reserved.

```
})

uploadTask.onProgressUpdate((res) => {
  console.log('Upload progress', res.progress)
  console.log('Length of already uploaded data', res.totalBytesSent)
  console.log('Total length of data expected to be uploaded', res.totalBytesExpecte
})

uploadTask.abort() // Aborts upload task
```

©2013-2022 Tencent Cloud. All rights reserved.

Websocket

Last updated：2024-07-12 19:29:13

Note:
It is advisable to utilize the SocketTask method to manage WebSocket connections, as it provides greater control over the lifecycle of each link. In scenarios where multiple WebSocket connections exist, the use of methods with a wx prefix may result in situations that deviate from expectations.

sendSocketMessage

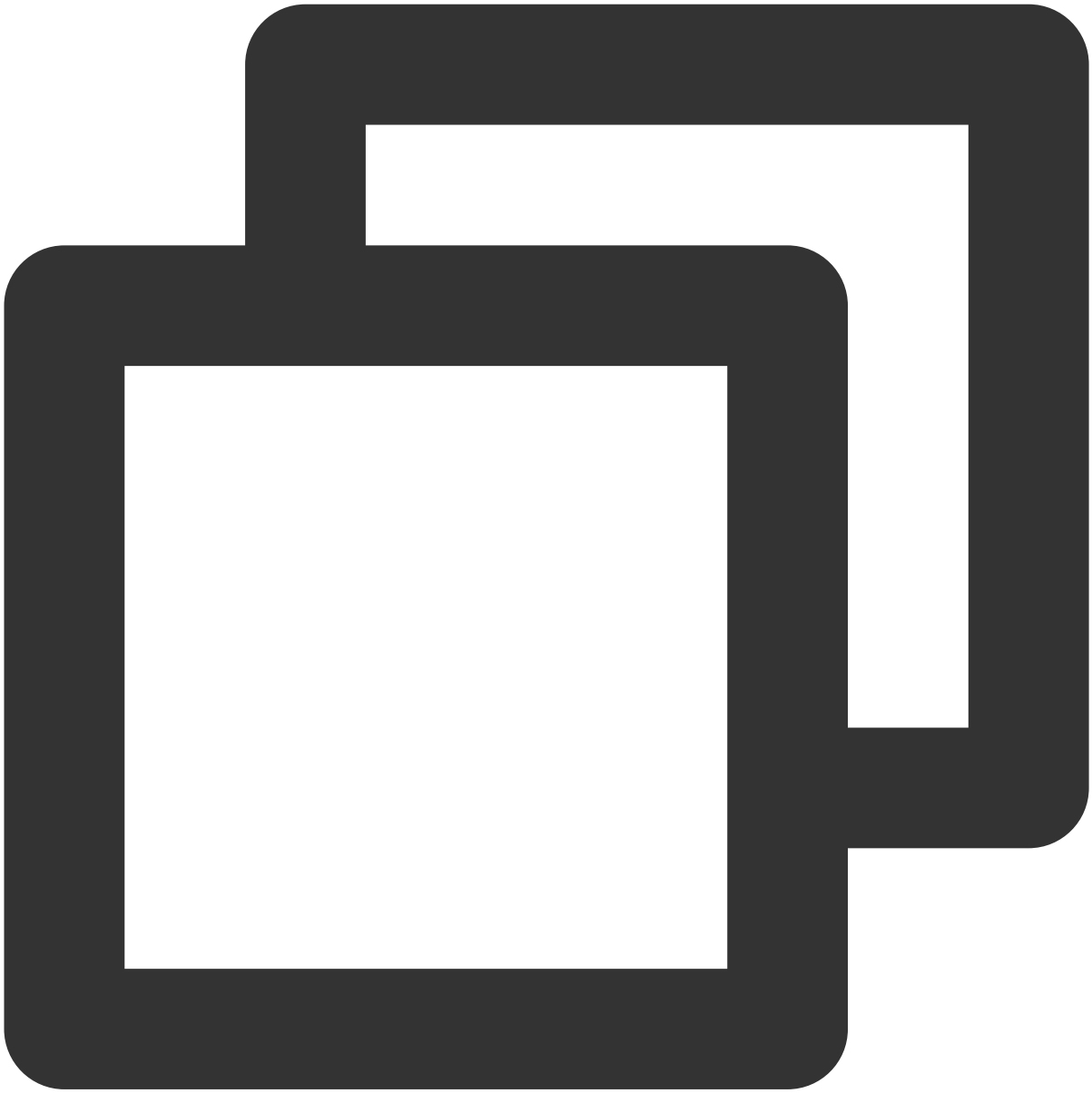
The usage of this API is as follows: wx.sendSocketMessage(Object object)

Function Description: This feature allows data transmission via WebSocket connection. It requires an initial wx.connectSocket, and can only transmit data after the wx.onSocketOpen callback.
Parameters and Description: Object object.

Attributes	Types	Default value	Required	Note
data	string/ArrayBuffer	-	Supported	Content to be dispatched
success	function	-	Not required	Callback function for successful interface invocation
fail	function	-	Not required	Callback function for unsuccessful interface invocation
complete	function	-	Not required	Callback function executed upon the completion of the interface invocation (both successful and unsuccessful invocations)

Sample Code:

©2013-2022 Tencent Cloud. All rights reserved.



```
let socketOpen = false
let socketMsgQueue = []
wx.connectSocket({
  url: 'test.php'
})

wx.onSocketOpen(function(res) {
  socketOpen = true
  for (let i = 0; i < socketMsgQueue.length; i++){
    sendSocketMessage(socketMsgQueue[i])
  }
```

©2013-2022 Tencent Cloud. All rights reserved.

```
  socketMsgQueue = []
})

function sendSocketMessage(msg) {
  if (socketOpen) {
    wx.sendSocketMessage({
      data:msg
    })
  } else {
    socketMsgQueue.push(msg)
  }
}
```

onSocketOpen

The usage method for this API is wx.onSocketOpen(function listener)

Function Description: Monitors the WebSocket connection opening event.
Parameters and Description: function listener, the monitoring function for the WebSocket connection opening event.

Attributes	Types	Note
header	object	HTTP response Header upon successful connection

onSocketMessage

The usage method for this API is wx.onSocketMessage(function listener)

Function Description: Monitors the event of WebSocket receiving a message from the server.
Parameters and Description: function listener, the monitoring function for the event of WebSocket receiving a message from the server.

Attributes	Types	Note
data	string/ArrayBuffer	Message returned by the server

©2013-2022 Tencent Cloud. All rights reserved.

onSocketError

The usage method for this API is wx.onSocketError(function listener)

Function Description: Monitors the WebSocket error event.

Parameters and Description: function listener, the monitoring function for the WebSocket error event.

Attributes	Types	Note
errMsg	string	Error Message

onSocketClose

The usage method for this API is wx.onSocketClose(function listener)

Function Description: Monitors the WebSocket connection closure event.

Parameters and Description: function listener, the monitoring function for the WebSocket connection closure event.

Attributes	Types	Note
code	number	A numerical value representing the status number of the closed connection, indicating the reason for the connection closure.
reason	string	A readable string indicating the reason for the connection closure.

connectSocket

The usage method for this API is SocketTask wx.connectSocket(Object object)

Function Description: Establishes a WebSocket connection. Please read the relevant instructions.

Parameters and Description: Object object.

Attributes	Types	Default value	Required	Note
url	string	-	Supported	Developer server's wss interface address

©2013-2022 Tencent Cloud. All rights reserved.

header	Object	-	Not required	HTTP Header, the Referer cannot be set in the Header.
protocols	Array.<string>	-	Not required	Array of sub-protocols.
tcpNoDelay	boolean	false	Not required	TCP_NODELAY setting during the establishment of a TCP connection.
timeout	number	-	Not required	Timeout duration, measured in milliseconds.
success	function	-	Not required	Callback function for successful interface invocation
fail	function	-	Not required	Callback function for unsuccessful interface invocation
complete	function	-	Not required	Callback function executed upon the completion of the interface invocation (both successful and unsuccessful invocations)

Return Value: SocketTask.

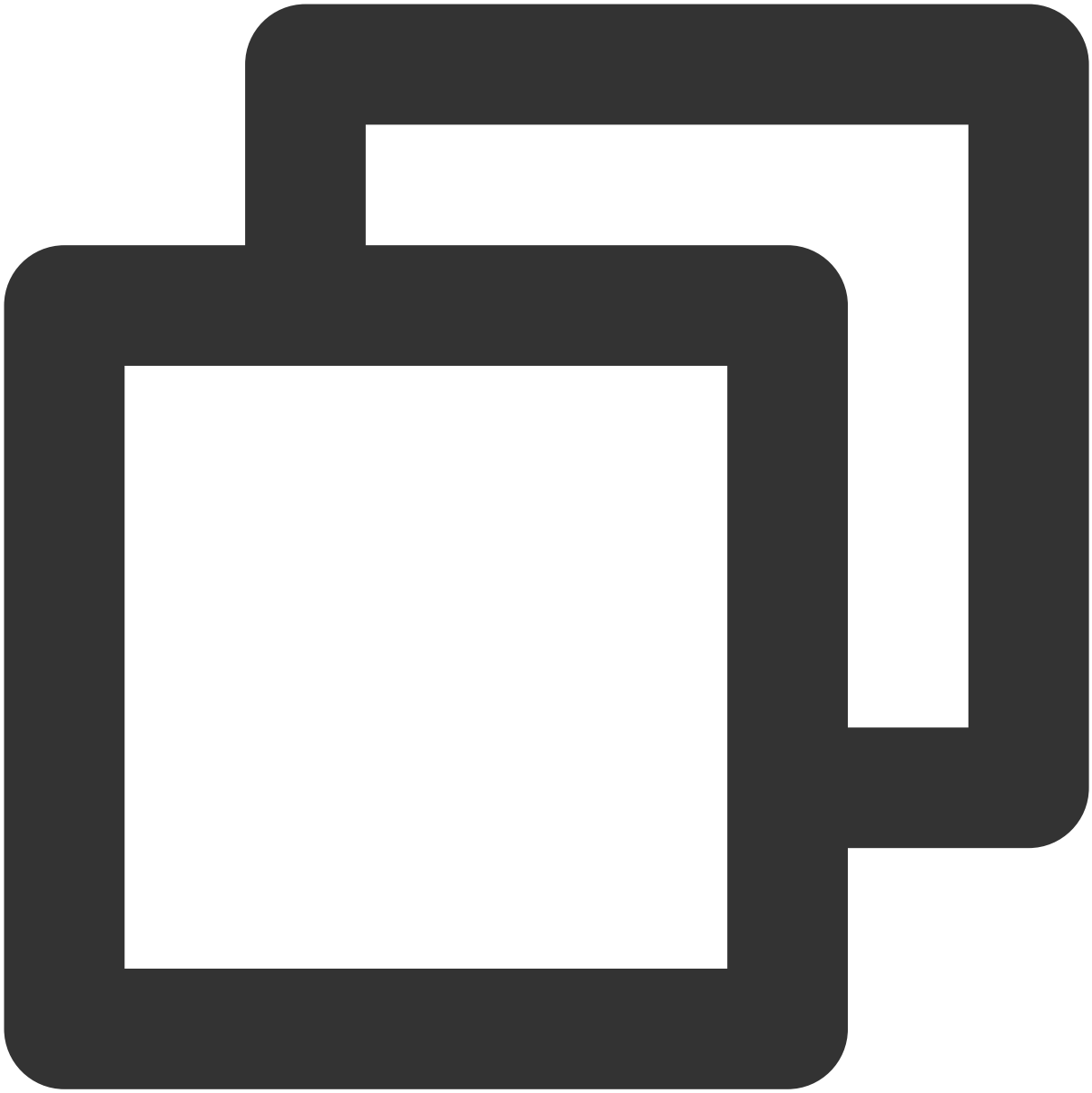
Concurrency Count

In versions 1.5.0 and above, a maximum of five WebSocket connections can coexist simultaneously.

In versions below 1.5.0, a mini program can only maintain a single WebSocket connection at a time. If a WebSocket connection already exists, it will be automatically closed and a new WebSocket connection will be established.

Example Code

©2013-2022 Tencent Cloud. All rights reserved.



```
wx.connectSocket({
  url: 'wss://example.qq.com',
  header:{
    'content-type': 'application/json'
  },
  protocols: ['protocol1']
})
```

©2013-2022 Tencent Cloud. All rights reserved.

closeSocket

The usage of this API is wx.closeSocket(Object object).

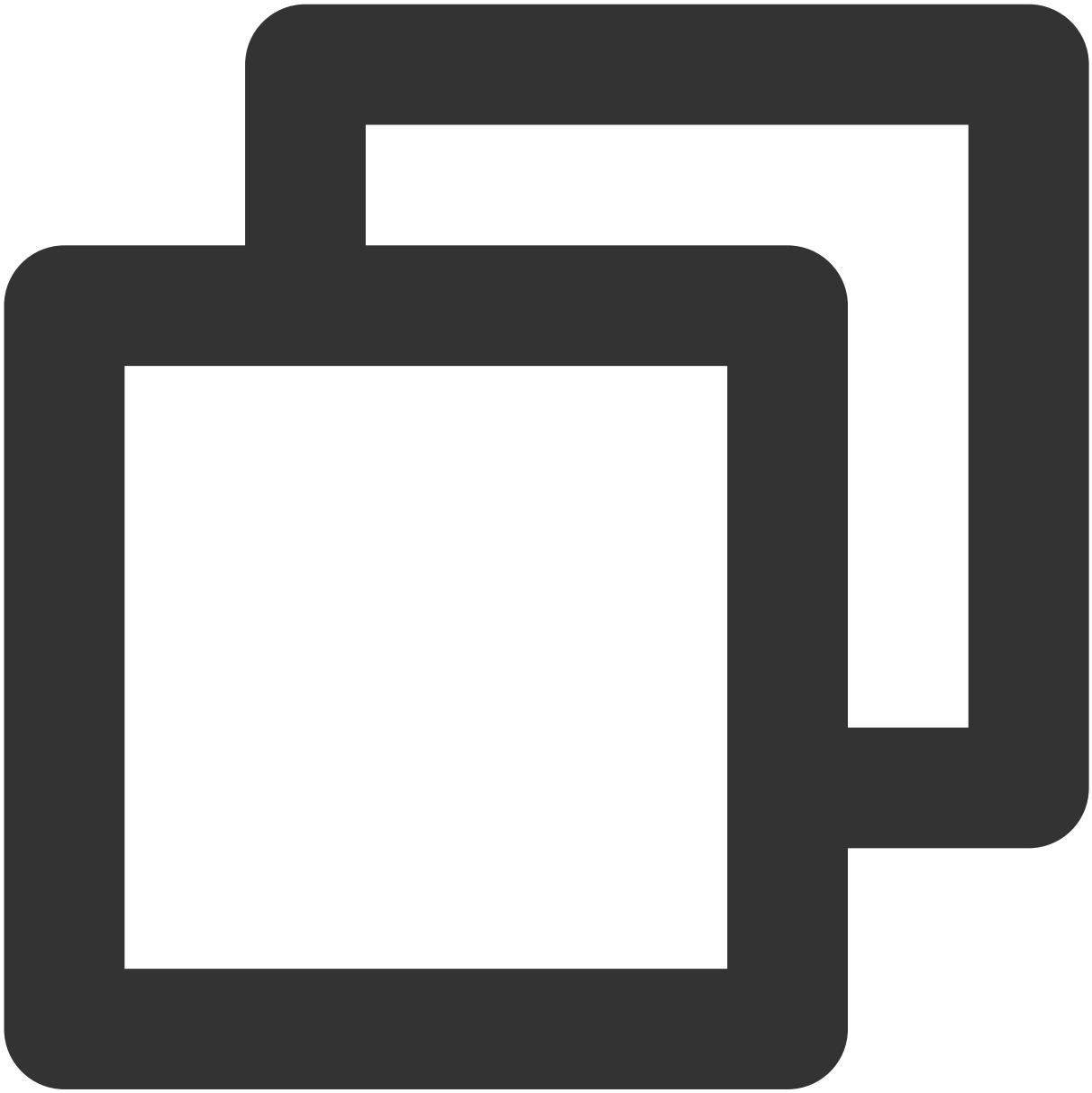
Function Description: This function closes the WebSocket connection. The method of use is as follows.

Parameters and Description: Object object.

Attributes	Types	Default value	Required	Note
code	number	1000 (indicating a normal closure of the connection)	Not required	A numerical value representing the status number of the closed connection, indicating the reason for the connection closure.
reason	string	-	Not required	A readable string indicating the reason for the connection closure. This string must be a UTF-8 text that does not exceed 123 bytes in length (not characters).
success	function	-	Not required	Callback function for successful interface invocation
fail	function	-	Not required	Callback function for unsuccessful interface invocation
complete	function	-	Not required	Callback function executed upon the completion of the interface invocation (both successful and unsuccessful invocations)

Sample Code:

©2013-2022 Tencent Cloud. All rights reserved.



```
wx.connectSocket({
  url: 'test.php'
})

//Please note that there may be timing issues here,
//If wx.connectSocket has not yet called back wx.onSocketOpen, and wx.closeSocket i
//It is imperative to call wx.closeSocket during the WebSocket's open period to eff
wx.onSocketOpen(function() {
  wx.closeSocket()
})
```

©2013-2022 Tencent Cloud. All rights reserved.

```
wx.onSocketClose(function(res) {
  console.log('WebSocket has been closed!')
})
```

SocketTask

.close

The method is utilized in the form of SocketTask.close(Object object).

Function Description: This function serves to close the WebSocket connection.

Parameters and Description: Object object.

Attributes	Types	Default value	Required	Note
code	number	1000 (indicating a normal closure of the connection)	Not required	A numerical value representing the status number of the closed connection, indicating the reason for the connection closure.
reason	string	-	Not required	A readable string indicating the reason for the connection closure. This string must be a UTF-8 text that does not exceed 123 bytes in length (not characters).
success	function	-	Not required	Callback function for successful interface invocation
fail	function	-	Not required	Callback function for unsuccessful interface invocation
complete	function	-	Not required	Callback function executed upon the completion of the interface invocation (both successful and unsuccessful invocations)

.onClose

The method is employed as SocketTask.onClose(function listener).

©2013-2022 Tencent Cloud. All rights reserved.

Function Description: Monitors the WebSocket connection closure event.

Parameters and Description: function listener, the monitoring function for the WebSocket connection closure event.

Attributes	Types	Note
code	number	A numerical value representing the status number of the closed connection, indicating the reason for the connection closure.
reason	string	A readable string indicating the reason for the connection closure.

.onError

The method is invoked as SocketTask.onError(function listener).

Function Description: Monitors the WebSocket error event.

Parameters and Description: function listener, the monitoring function for the WebSocket error event.

Attributes	Types	Note
errMsg	string	Error Message

.onMessage

The method is implemented as SocketTask.onMessage(function listener).

Function Description: Monitors the event of WebSocket receiving a message from the server.

Parameters and Description: function listener, the monitoring function for the event of WebSocket receiving a message from the server.

Attributes	Types	Note
data	string/ArrayBuffer	Message returned by the server

.onOpen

The method is executed as SocketTask.onOpen(function listener).

Function Description: Monitors the WebSocket connection opening event.

©2013-2022 Tencent Cloud. All rights reserved.

Parameters and Description: function listener, the monitoring function for the WebSocket connection opening event.

Attributes	Types	Note
header	Object	HTTP response Header upon successful connection

.send

> The usage of this method is SocketTask.send(Object object).

Function Description: Data is transmitted via a WebSocket connection.

Parameters and Description: Object object.

Attributes	Types	Default value	Required	Note
data	string/ArrayBuffer	-	Supported	Content to be dispatched
success	function	-	Not required	Callback function for successful interface invocation
fail	function	-	Not required	Callback function for unsuccessful interface invocation
complete	function	-	Not required	Callback function executed upon the completion of the interface invocation (both successful and unsuccessful invocations)

©2013-2022 Tencent Cloud. All rights reserved.

Download

Last updated：2024-07-12 19:29:02

downloadFile

This API is used via DownloadTask wx.downloadFile (Object object).

Note:

Please specify a reasonable Content-Type field in the server response header to ensure the client correctly processes the file type.

Feature Description: Downloads file resources to local storage. The client directly initiates an HTTPS GET request, returning the local temporary path of the file (local path). The maximum file allowed for a single download is 200 MB. See relevant instructions before use.

Parameter and Description: Object.

Attribute	Type	Required	Description
url	string	Yes	URL for resource download
header	Object	No	HTTP request's Header; setting Referer within the Header is not allowed.
filePath	string	No	Designated path for storing the downloaded files
success	function	No	Callback Function of Successful Interface Call
fail	function	No	Callback Function of Failing Interface Call
complete	function	No	Callback function executed upon the completion of the interface invocation (both successful and unsuccessful invocations)

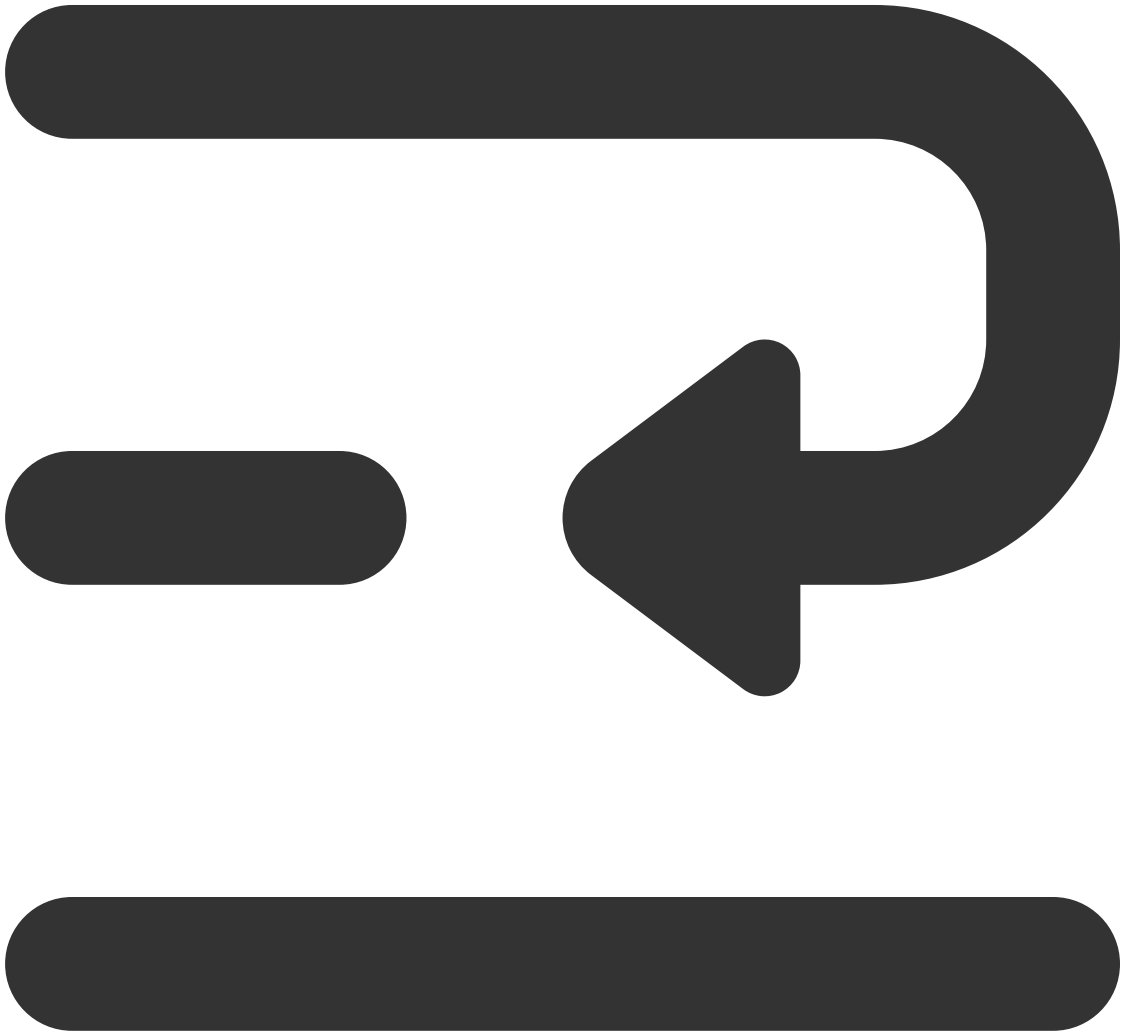
Parameters for object.success callback function: Object res.

Attribute	Type	Description
tempFilePath	string	Temporary file path (local path). If no filePath is specified for file storage, this will be returned. The downloaded files will be stored in a temporary folder.
filePath	string	User file path (local path). This will be returned when a filePath is passed in, and it will be consistent with the provided filePath.
statusCode	number	HTTP status code returned from the developer's server.

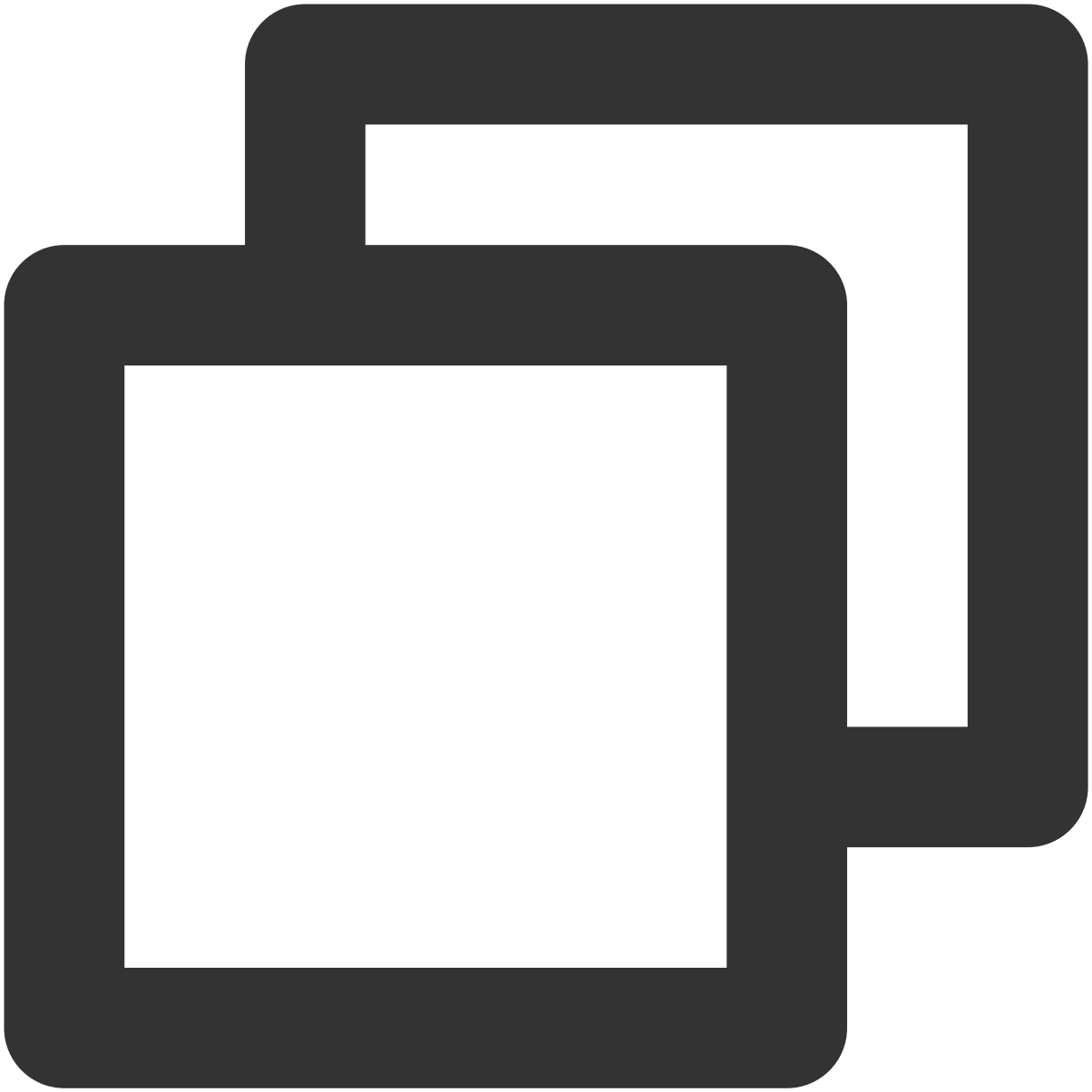
©2013-2022 Tencent Cloud. All rights reserved.

Return Value: DownloadTask

Sample Code



©2013-2022 Tencent Cloud. All rights reserved.



```
wx.downloadFile({
  url: 'https://example.com/audio/123', // This is merely an example, not an actual
  success (res) {
    // As long as the server has response data, the response content will be writte
    if (res.statusCode === 200) {
      wx.playVoice({
        filePath: res.tempFilePath
      })
    }
  }
})
```

©2013-2022 Tencent Cloud. All rights reserved.

DownloadTask

.abort

This method is used via DownloadTask.abort().

Function Description: Aborts the download task.

.onProgressUpdate

This method is used via DownloadTask.onProgressUpdate(function listener).

Function Description: Monitors the event of download progress changes.
Parameter and Description: function listener, the listener function for download progress change events.

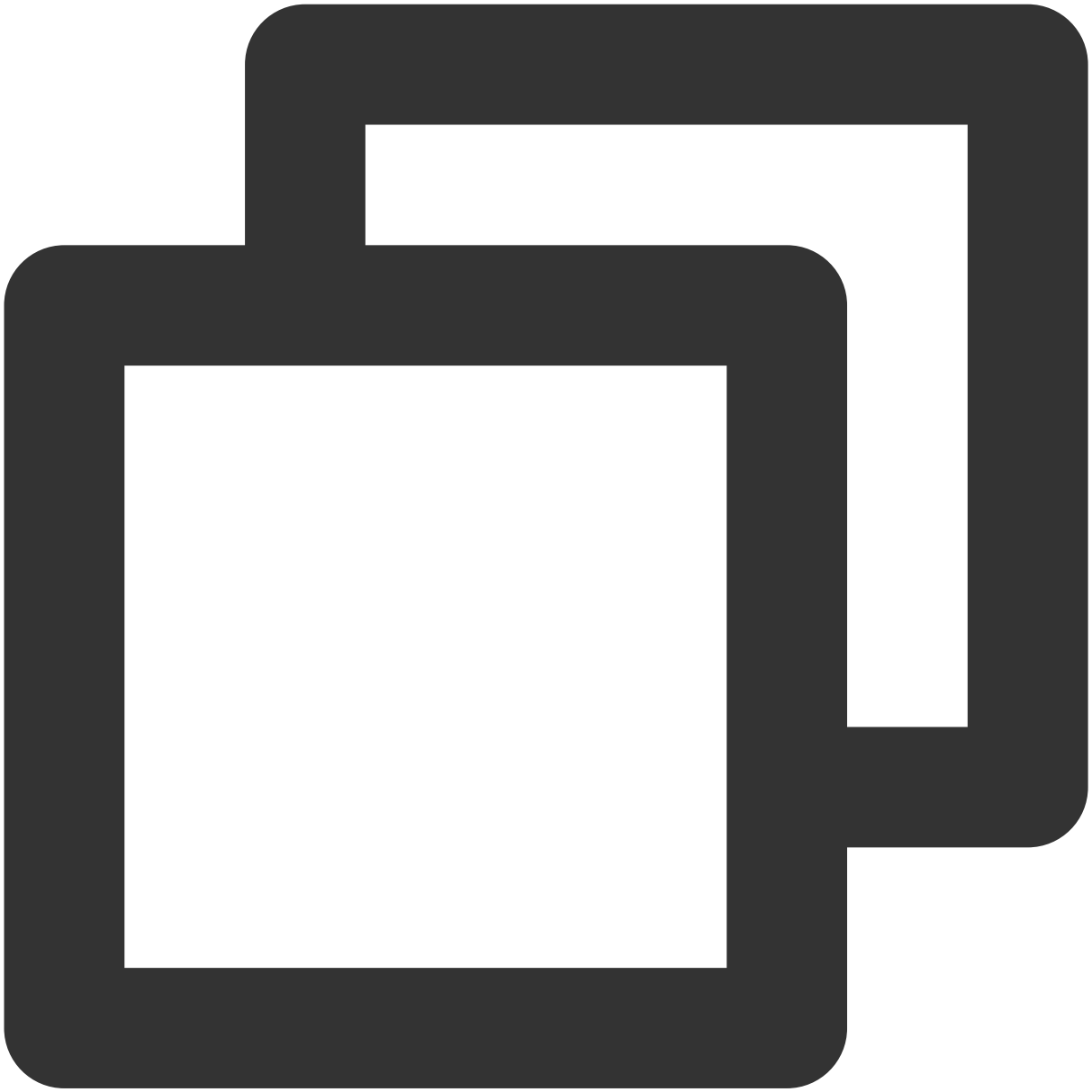
Attribute	Type	Description
progress	number	Download progress percentage
totalBytesWritten	number	Length of the data already downloaded, measured in Bytes.
totalBytesExpectedToWrite	number	Total length of the data expected to be downloaded, measured in bytes.

.offProgressUpdate

This method is used via DownloadTask.offProgressUpdate(function listener).

Feature Description: Removes the listener function for download progress change events.
Parameters and Description: function listener, the listener function passed in by onProgressUpdate. If this parameter is not passed in, all listener functions will be removed.
Sample Code

©2013-2022 Tencent Cloud. All rights reserved.



```
const listener = function (res) { console.log(res) }

DownloadTask.onHeadersReceived(listener)
DownloadTask.offHeadersReceived(listener) // The same function object as the listen
```

.onHeadersReceived

This method is used via DownloadTask.onHeadersReceived(function listener).

©2013-2022 Tencent Cloud. All rights reserved.

Feature Description: Monitors HTTP Response Header events. This occurs earlier than the request completion event.

Parameter and Description: function listener, the listener function for HTTP Response Header events.

Attribute	Type	Description
header	Object	HTTP Response Header returned from the developer's server.

.offHeadersReceived

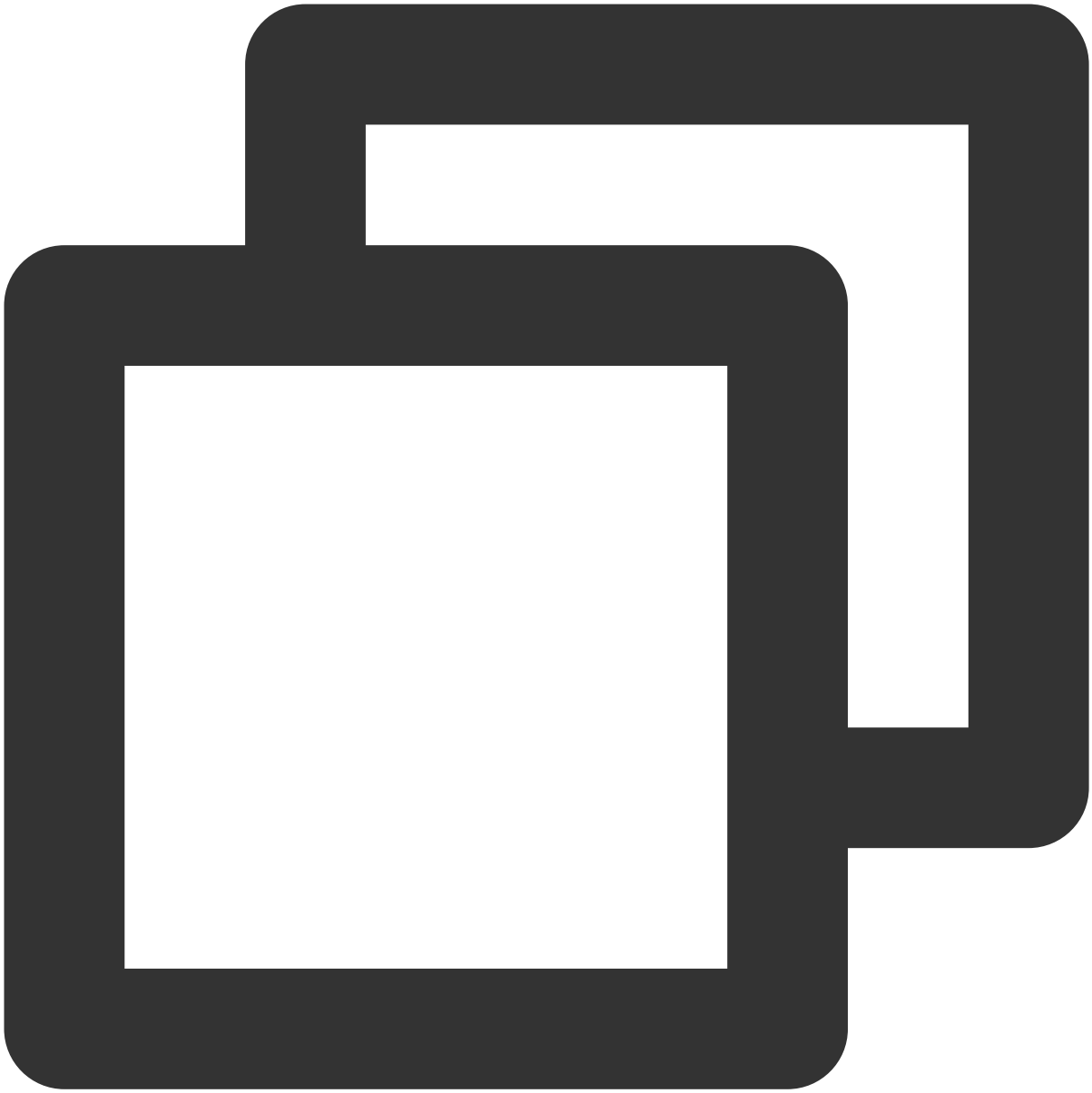
This method is used via DownloadTask.offHeadersReceived(function listener).

Feature Description: Removes the listener function for HTTP Response Header events.

Parameters and Description: function listener, the listener function passed in by onHeadersReceived. If this parameter is not passed in, all listener functions will be removed.

Sample Code

©2013-2022 Tencent Cloud. All rights reserved.

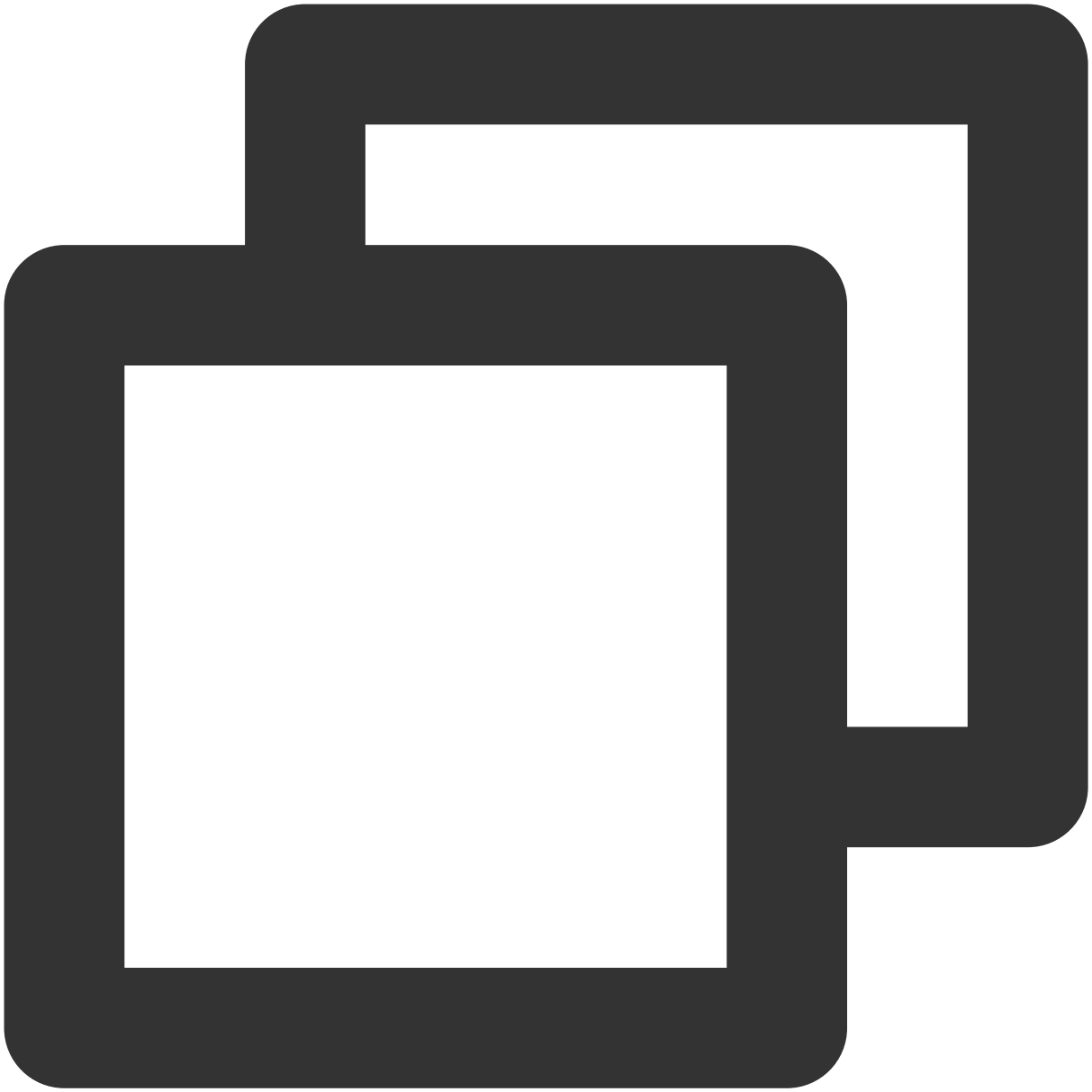


```
const listener = function (res) { console.log(res) }

DownloadTask.onHeadersReceived(listener)
DownloadTask.offHeadersReceived(listener) // The same function object as the listen
```

Sample code

©2013-2022 Tencent Cloud. All rights reserved.



```
const downloadTask = wx.downloadFile({
  url: 'http://example.com/audio/123', // This is merely an example, not an actual
  success (res) {
    wx.playVoice({
      filePath: res.tempFilePath
    })
  }
})

downloadTask.onProgressUpdate((res) => {
  console.log('Download progress', res.progress)
```

©2013-2022 Tencent Cloud. All rights reserved.

```
  console.log('Length of data already downloaded', res.totalBytesWritten)
    console.log(''Length of data expected to be downloaded', res.totalBytesExpected
})

downloadTask.abort() // Aborts the download task
```

©2013-2022 Tencent Cloud. All rights reserved.

TCP Communication

Last updated：2024-07-12 19:28:31

createTCPSocket

This API is used via TCPSocket?wx.createTCPSocket().

Feature description: Creates a TCP Socket instance.
Return Value: A TCP Socket instance.
Link limitations
Allows communication with non-local IP within the local area network.
Allows communication with configured server domain names.
Prohibits connections to the following port numbers: below 1024, 1099, 1433, 1521, 1719, 1720, 1723, 2049, 2375, 3128, 3306, 3389, 3659, 4045, 5060, 5061, 5432, 5984, 6379, 6000, 6566, 7001, 7002, 8000-8100, 8443, 8888, 9200, 9300, 10051, 10080, 11211, 27017, 27018, 27019.
A maximum of 20 TCPSockets can be created within every 5-minute interval.

TCPSocket

Note:
A TCP Socket instance, using IPv4 protocol by default.
When errCode is -2, the corresponding errno should be present in errMsg. Developers can examine the specific error message in the Linux code's errno-base.h and errno.h based on the errno.

.bindWifi

This method is used via TCPSocket.bindWifi(Object options).

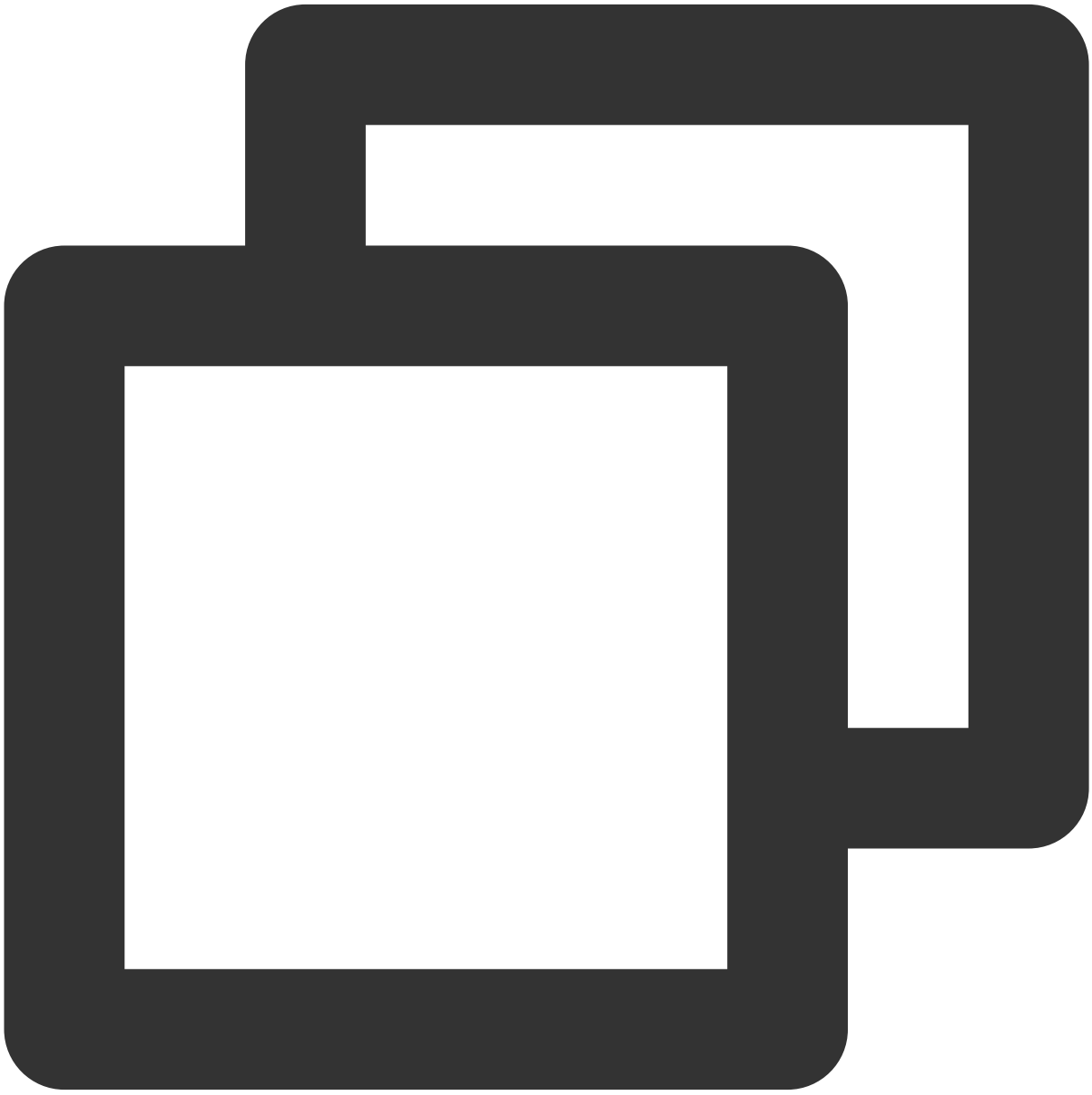
Feature Description: Binds the TCP Socket to the current Wi-Fi network, which will, upon successful execution, trigger the onBindWifi event (supported only on Android).
Parameter and Description: Object options.

Attribute	Type	Default value	Required	Note

©2013-2022 Tencent Cloud. All rights reserved.

BSSID	string	-	Yes	The BSSID of the current Wi-Fi network, which can be obtained through wx.getConnectedWifi.

Sample Code



```
const tcp = wx.createTCPSocket()
tcp.bindWifi({ BSSID: 'xxx' })
tcp.onBindWifi(() => {})
```

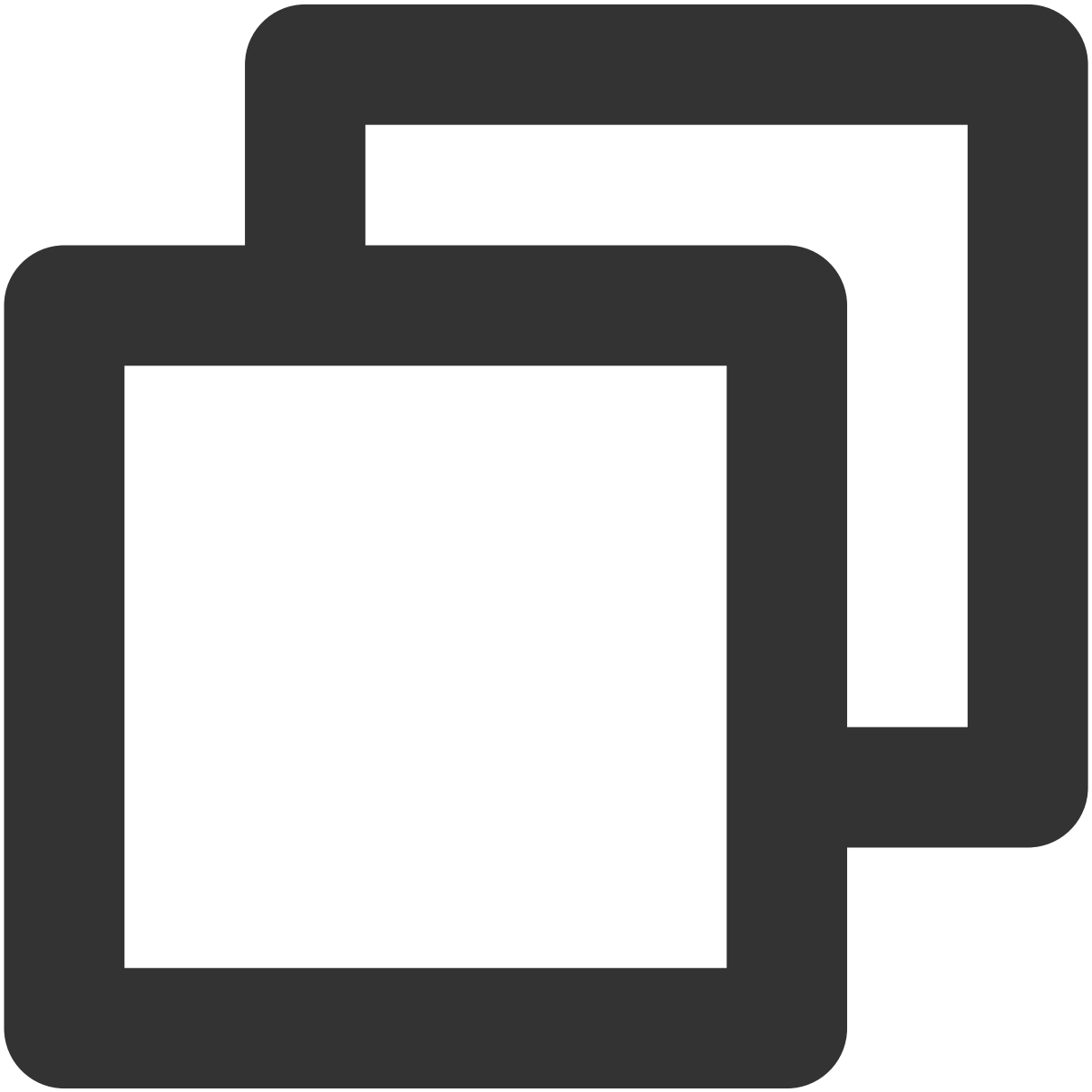
©2013-2022 Tencent Cloud. All rights reserved.

.close

This method is used via TCPSocket.close().

Feature Description: Disables connection.

Sample Code



```
const tcp = wx.createTCPSocket()
tcp.close()
```

©2013-2022 Tencent Cloud. All rights reserved.

.connect

This method is used via TCPSocket.connect(Object options).

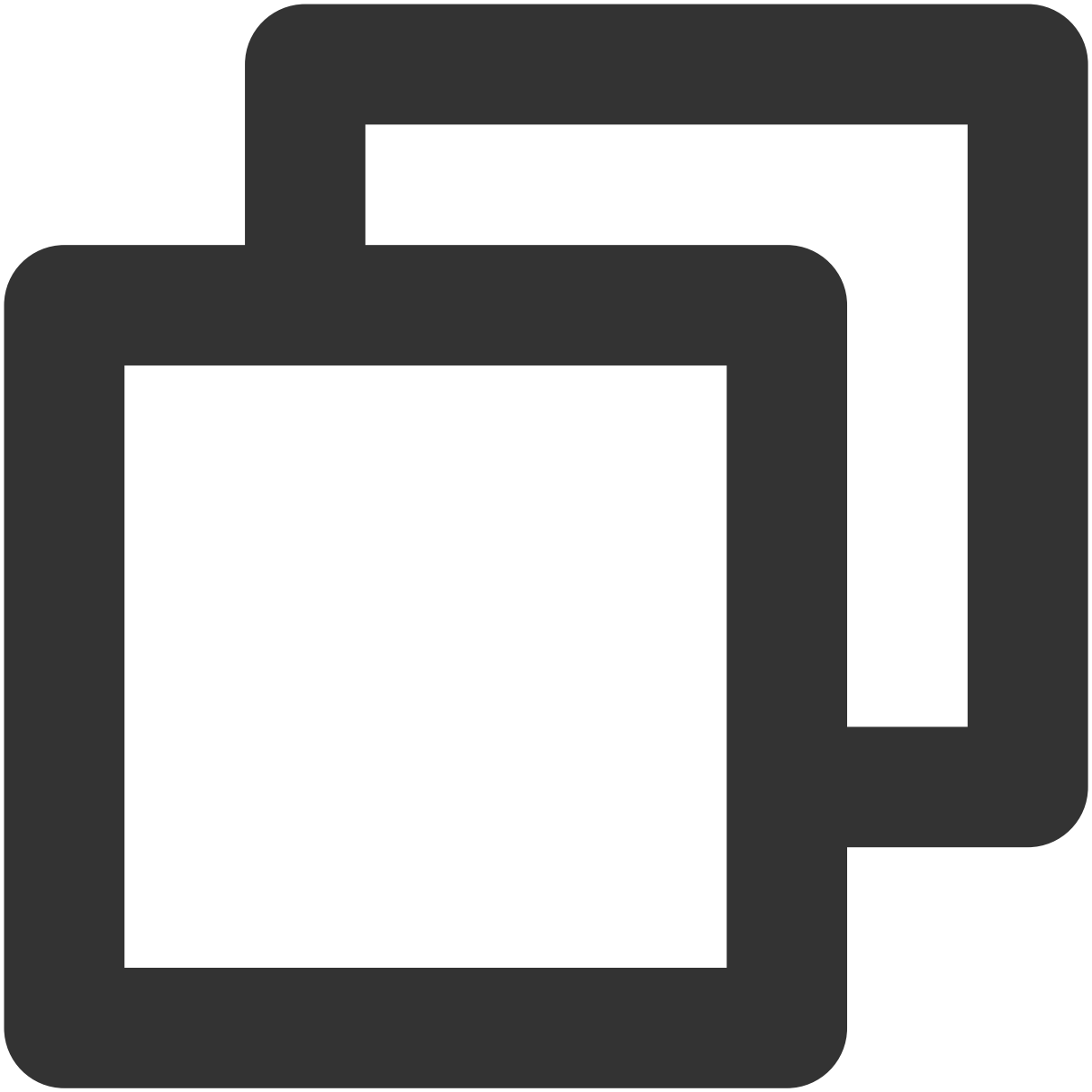
Feature Description: Initiates a connection on the specified socket.

Parameter and Description: Object options.

Attribute	Type	Default value	Required	Note
address	string	-	Yes	The address to which the socket is to be connected.
Port No.	number	-	Yes	The port to which the socket is to be connected.
timeout	number	2	No	The timeout duration for the socket connection, defaulting to 2 seconds.

Sample Code

©2013-2022 Tencent Cloud. All rights reserved.



```
const tcp = wx.createTCPSocket()
tcp.connect({address: '192.168.193.2', port: 8848})
```

.onClose

This method is used via TCPSocket.onClose(function listener).

©2013-2022 Tencent Cloud. All rights reserved.

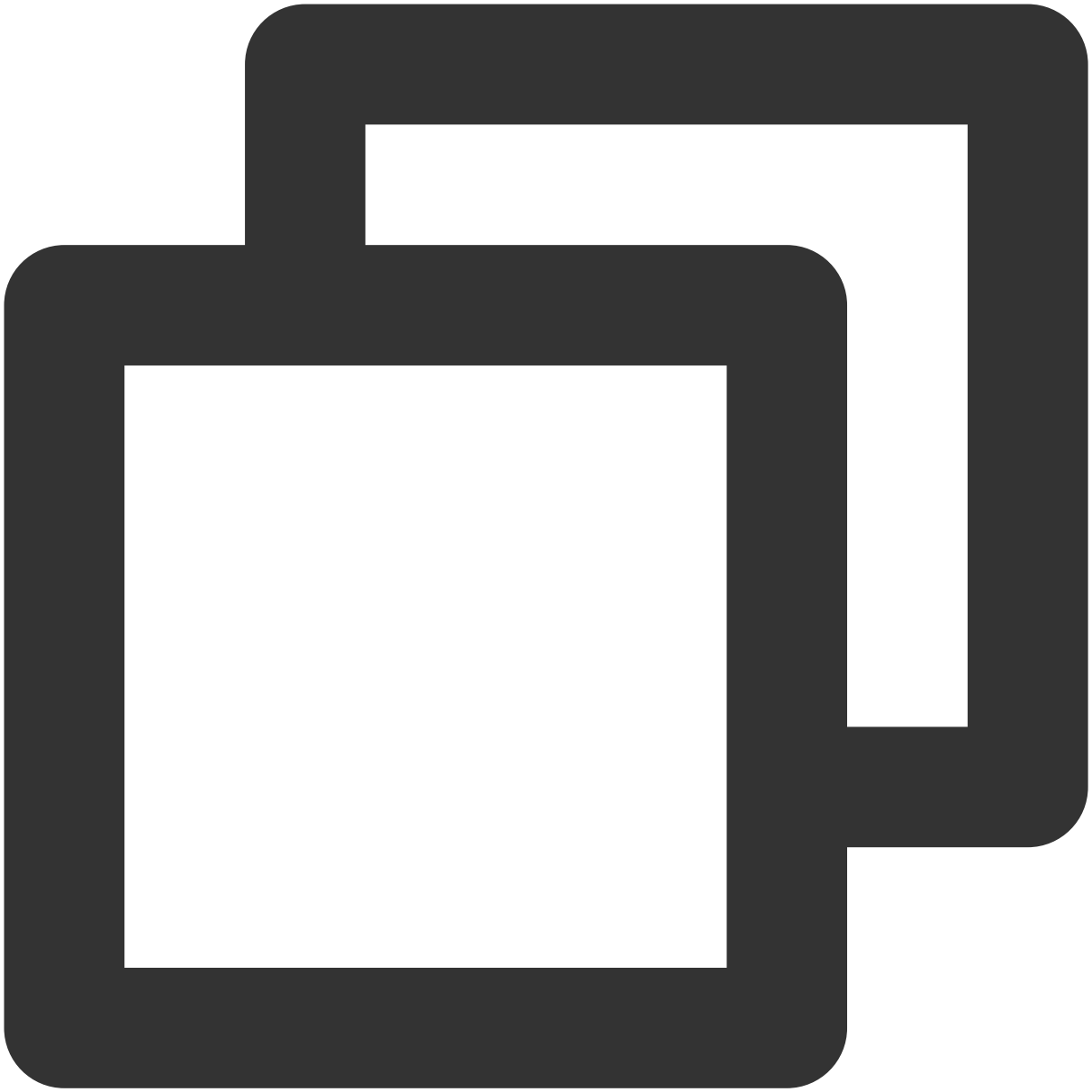
Feature Description: Monitors events that are emitted once the socket is fully closed.
Parameter and Description: function listener, the listener function for events triggered once the socket is fully closed.

.offClose

This method is used via TCPSocket.offClose(function listener).

Feature Description: Removes the listener function for events that are emitted once the socket is fully closed.
Parameter and Description: function listener, the listener function passed in by onClose. If this parameter is not passed in, all listener functions will be removed.
Sample Code

©2013-2022 Tencent Cloud. All rights reserved.



```
const listener = function (res) { console.log(res) }

TCPSocket.onConnect(listener)
TCPSocket.offConnect(listener) // The same function object as the listener must be
```

.onConnect

This method is used via TCPSocket.onConnect(function listener).

©2013-2022 Tencent Cloud. All rights reserved.

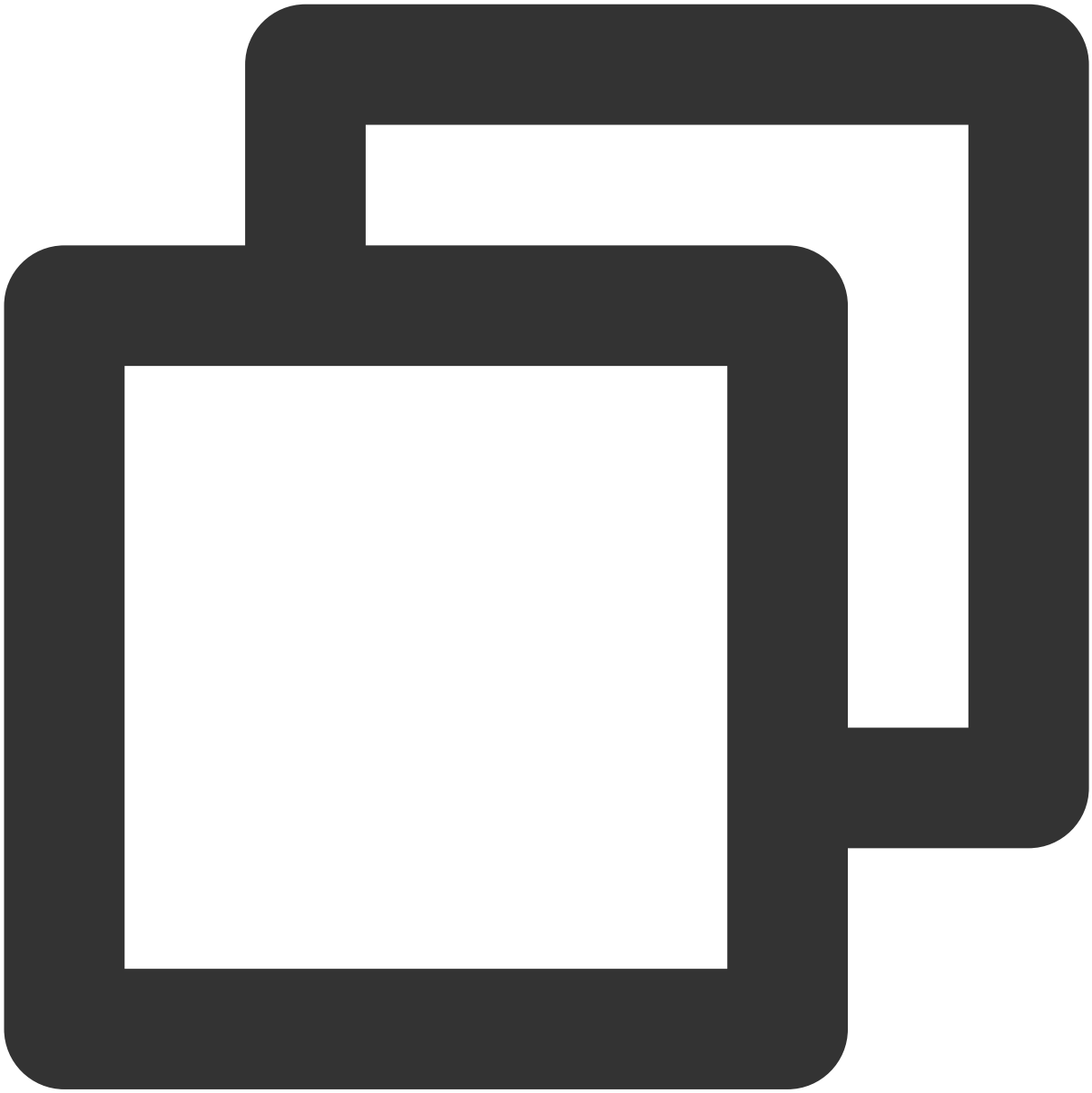
Feature Description: Monitors events that are triggered when a socket connection is successfully established.
Parameter and Description: function listener, the listener function for events triggered when a socket connection is successfully established.

.offConnect

This method is used via TCPSocket.offConnect(function listener).

Feature Description: Removes the listener function for events triggered when a socket connection is successfully established.
Parameter and Description: function listener, the listener function passed in by onConnect. If this parameter is not passed in, all listener functions will be removed.
Sample Code

©2013-2022 Tencent Cloud. All rights reserved.



```
const listener = function (res) { console.log(res) }

TCPSocket.onConnect(listener)
TCPSocket.offConnect(listener) // The same function object as the listener must be
```

.onError

This method is used via TCPSocket.onError(function listener).

©2013-2022 Tencent Cloud. All rights reserved.

Feature Description: Monitoring is triggered when an error occurs.

Parameters and Description: The listener function of function listener.

Attribute	Type	Note
errMsg	string	Error Message

.offError

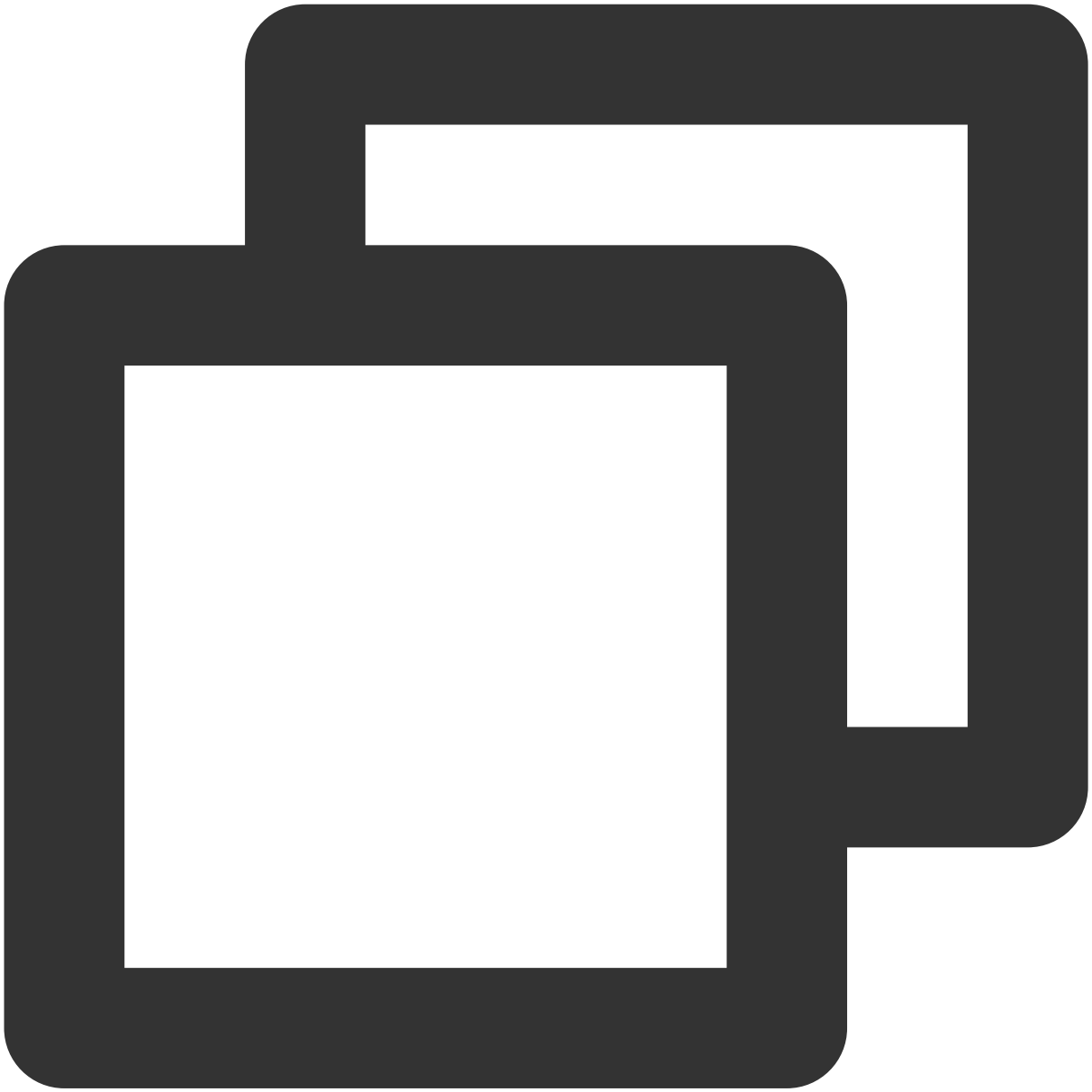
This method is used via TCPSocket.offError(function listener).

Feature Description: Removes the listener function that is triggered when an error occurs.

Parameter and Description: function listener, the listener function passed in by onError. If this parameter is not passed in, all listener functions will be removed.

Sample Code

©2013-2022 Tencent Cloud. All rights reserved.



```
const listener = function (res) { console.log(res) }

TCPSocket.onError(listener)
TCPSocket.offError(listener) // The same function object as the listener must be pa
```

.onMessage

This method is used via TCPSocket.onMessage(function listener).

©2013-2022 Tencent Cloud. All rights reserved.

Feature Description: Monitors events that are triggered when data is received.
Parameter and Description: function listener, when receiving data to trigger the event of the listener function, the parameter Object res as follows:

Attribute	Type	Note
message	ArrayBufffer	Received message
remoteInfo	Object	Sender's address information
localInfo	Object	Recipient's address information

Structure attributes of "remoteInfo"

Structure attributes	Type	Note
address	string	The address of the socket sending the message
family	string	The protocol family in use, either IPv4 or IPv6.
Port No.	number	Port No.
size	number	The size of the message, measured in bytes.

Structure attributes of "localInfo"

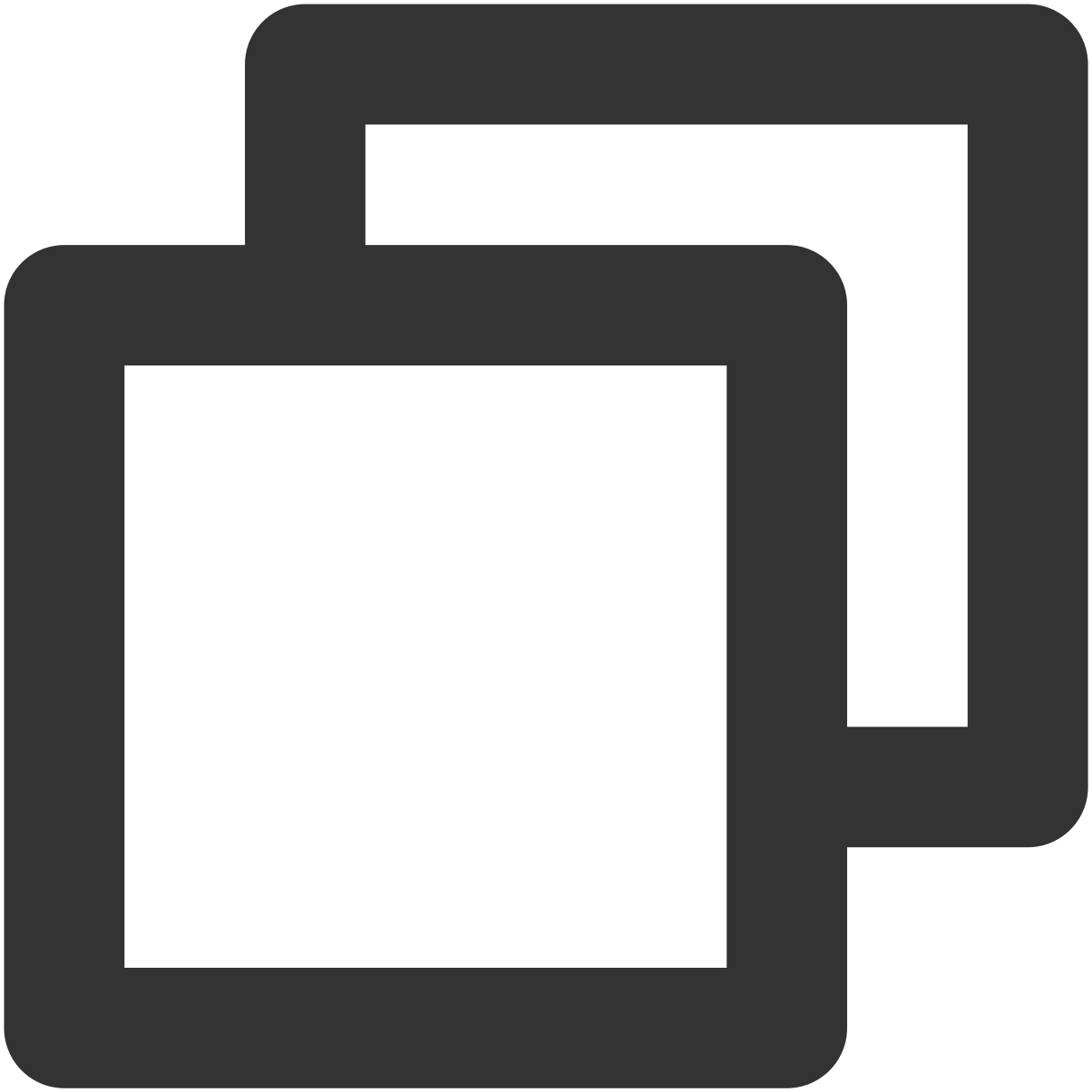
Structure attributes	Type	Note
address	string	The address of the socket sending the message
family	string	The protocol family in use, either IPv4 or IPv6.
Port No.	number	Port No.

.offMessage

This method is used via TCPSocket.offMessage(function listener).

Feature Description: Removes the listener function of the event that is triggered when data is received.
Parameter and Description: function listener, the listener function passed in by onMessage. If this parameter is not passed in, all listener functions will be removed.
Sample Code

©2013-2022 Tencent Cloud. All rights reserved.



```
const listener = function (res) { console.log(res) }

TCPSocket.onMessage(listener)
TCPSocket.offMessage(listener) // The same function object as the listener must be
```

.onBindWifi

This method is used via TCPSocket.onBindWifi(function listener).

©2013-2022 Tencent Cloud. All rights reserved.

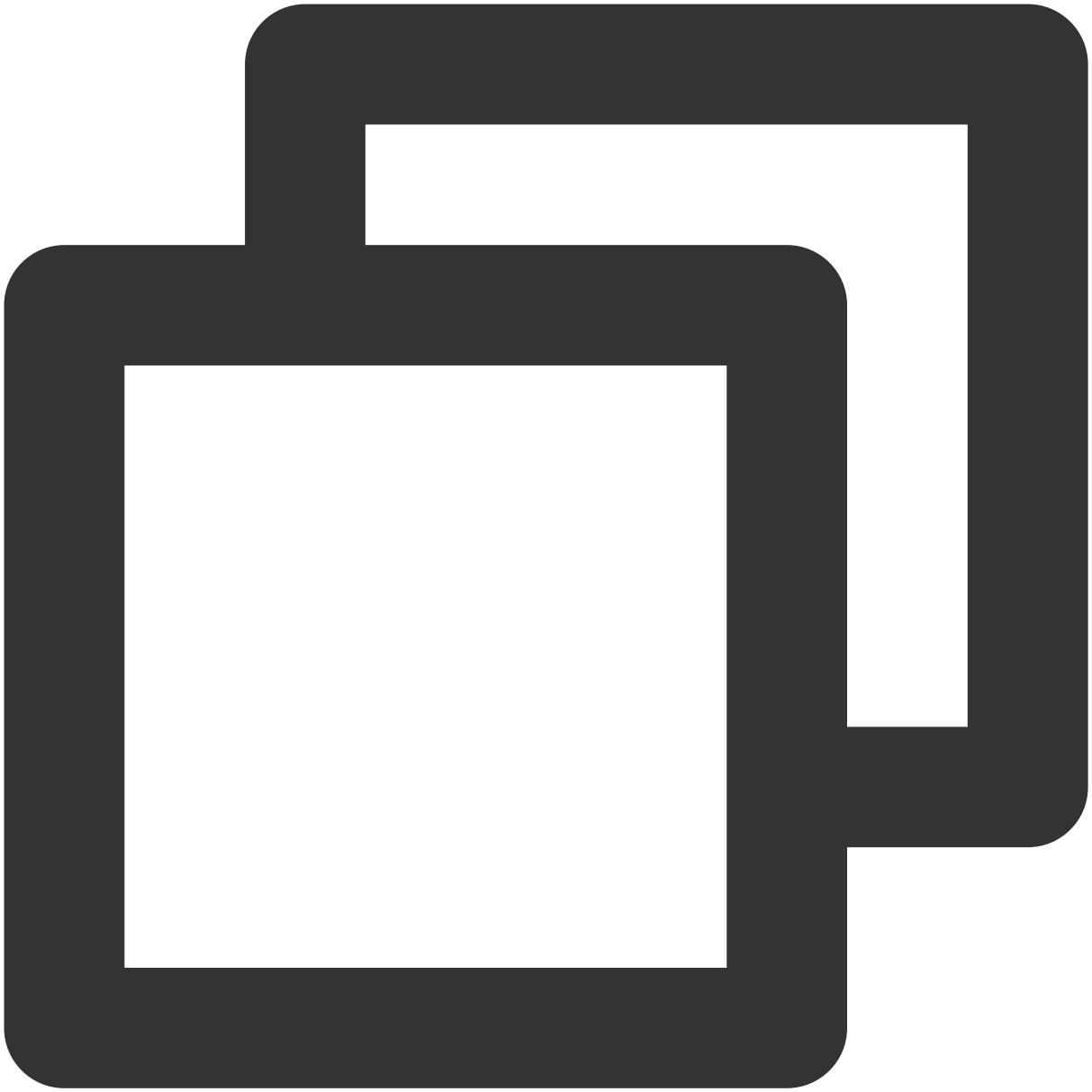
Feature Description: Monitors events triggered when a socket successfully binds to the current Wi-Fi network.
Parameter and Description: function listener, the listener function for events triggered when a socket successfully binds to the current Wi-Fi network.

.offBindWifi

This method is used via TCPSocket.offBindWifi(function listener).

Feature Description: Removes the listener function for events triggered when a socket successfully binds to the current Wi-Fi network.
Parameters and Description: function listener, the listener function passed in by onBindWifi. If this parameter is not passed in, all listener functions will be removed.
Sample Code

©2013-2022 Tencent Cloud. All rights reserved.



```
const listener = function (res) { console.log(res) }

TCPSocket.onBindWifi(listener)
TCPSocket.offBindWifi(listener) // The same function object as the listener must be
```

.write

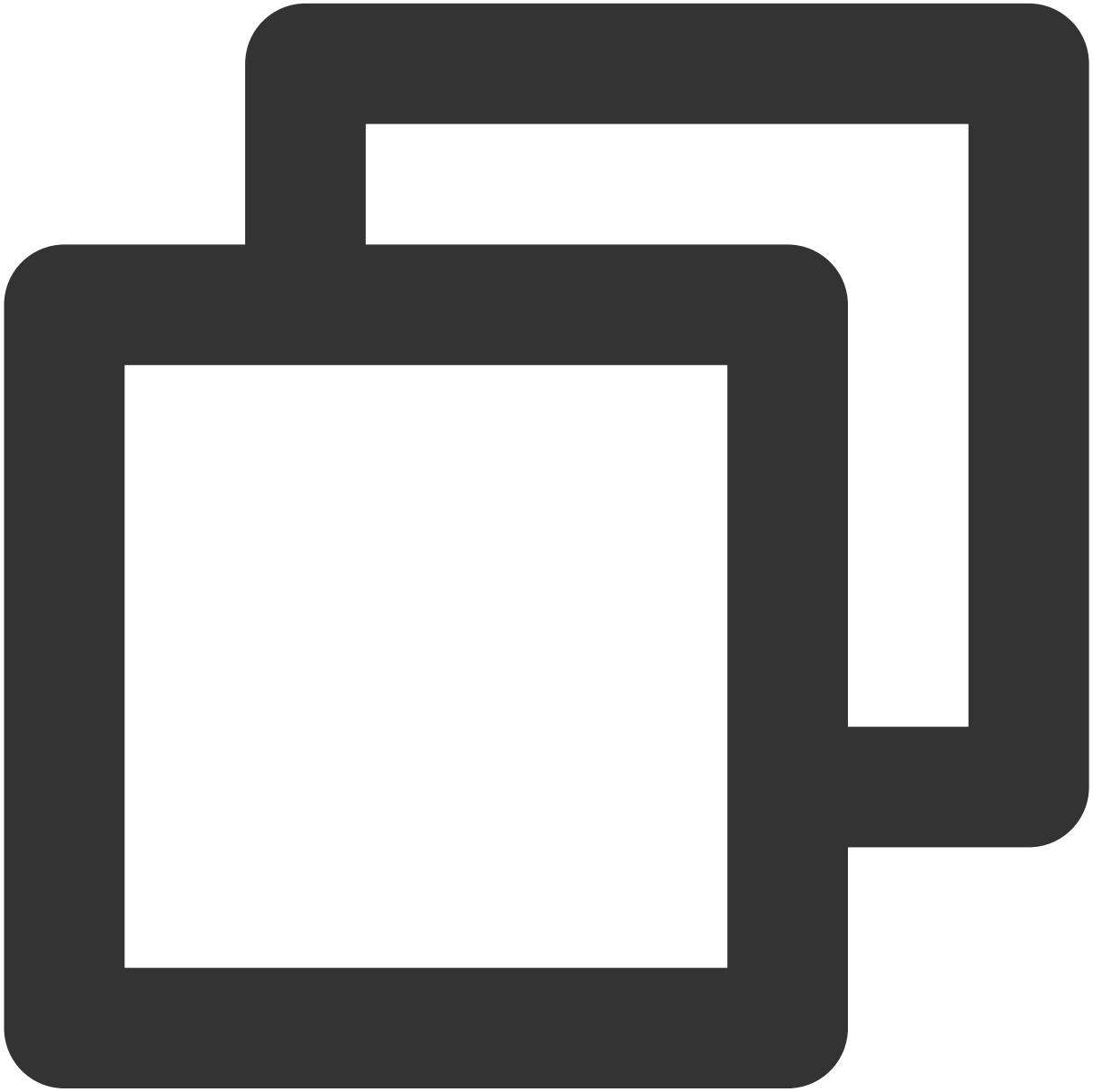
This method is used via TCPSocket.write(string|ArrayBuffer data).

©2013-2022 Tencent Cloud. All rights reserved.

Feature Description: Sends data over the socket.

Parameter and Description: string|ArrayBuffer data, the data to be sent.

Sample Code



```
const tcp = wx.createTCPSocket()
tcp.write('hello, how are you')
```

©2013-2022 Tencent Cloud. All rights reserved.

Data Caching

Last updated：2024-07-12 19:27:52

setStorage

This API is used via wx.setStorage(Object object).

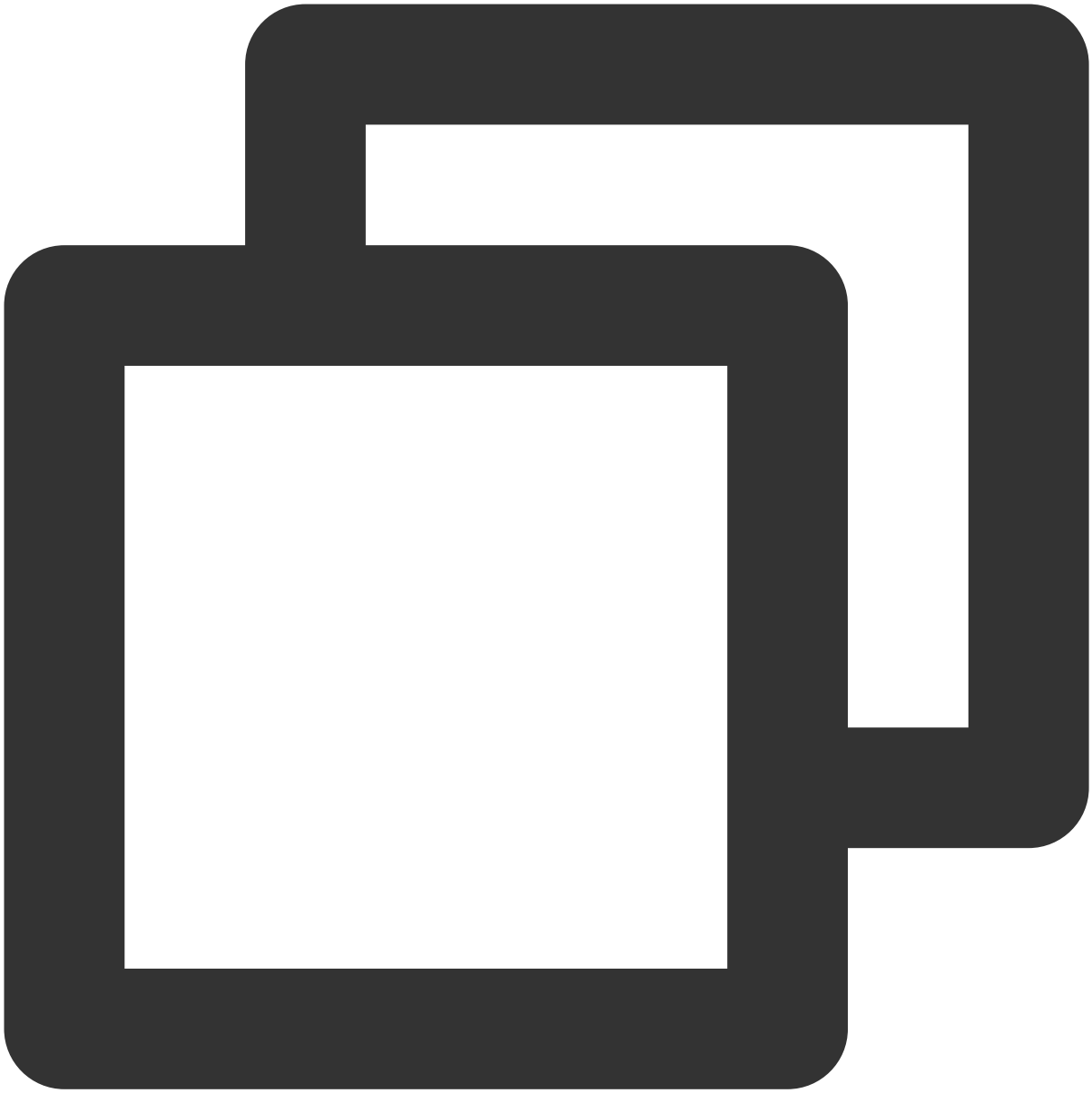
Feature Description: Stores data in the specified key within the local cache, overwriting any existing content associated with that key. The data remains available unless the user actively deletes it or it is cleared by the system due to storage space constraints. The maximum data length allowed for a single key is 1 MB, with an overall data storage limit of 10 MB.

Parameter and Description: Object.

Attribute	Type	Default value	Required	Description
key	string	-	Yes	Specified key in the local cache
data	any	-	Yes	The content that needs to be stored. Only native types, Date, and objects that can be serialized through JSON.stringify are supported.
success	function	-	No	Callback Function of Successful Interface Call
fail	function	-	No	Callback Function of Failing Interface Call
complete	function	-	No	Callback function executed upon the completion of the interface invocation (both successful and unsuccessful invocations)

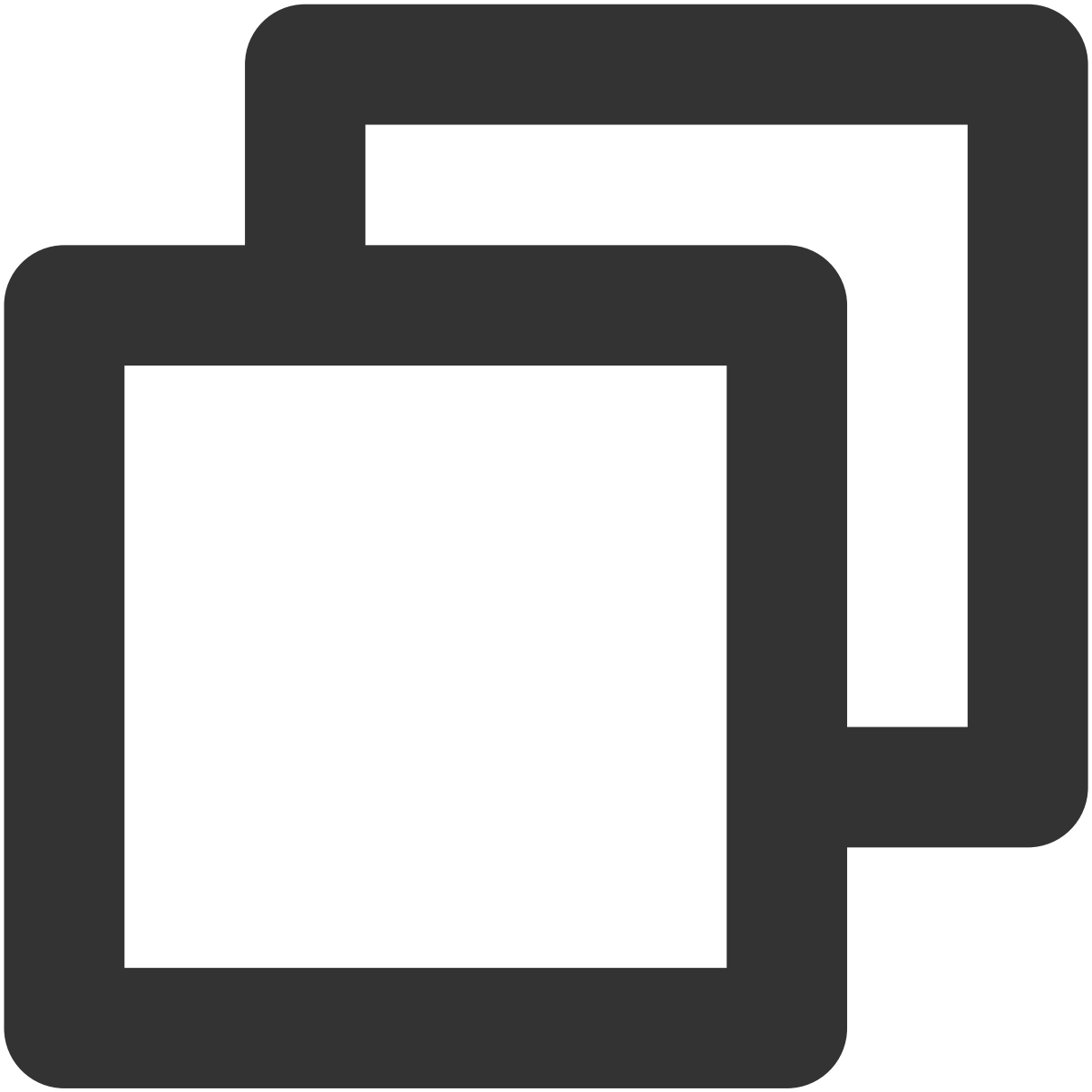
Sample Code

©2013-2022 Tencent Cloud. All rights reserved.



```
wx.setStorage({
  key:"key",
  data:"value"
})
```

©2013-2022 Tencent Cloud. All rights reserved.



```
wx.setStorage({
  key: 'key',
  success (res) {
    console.log(res.data)
  }
})
```

setStorageSync

©2013-2022 Tencent Cloud. All rights reserved.

This API is used via wx.setStorageSync(string key, any data).

Note:

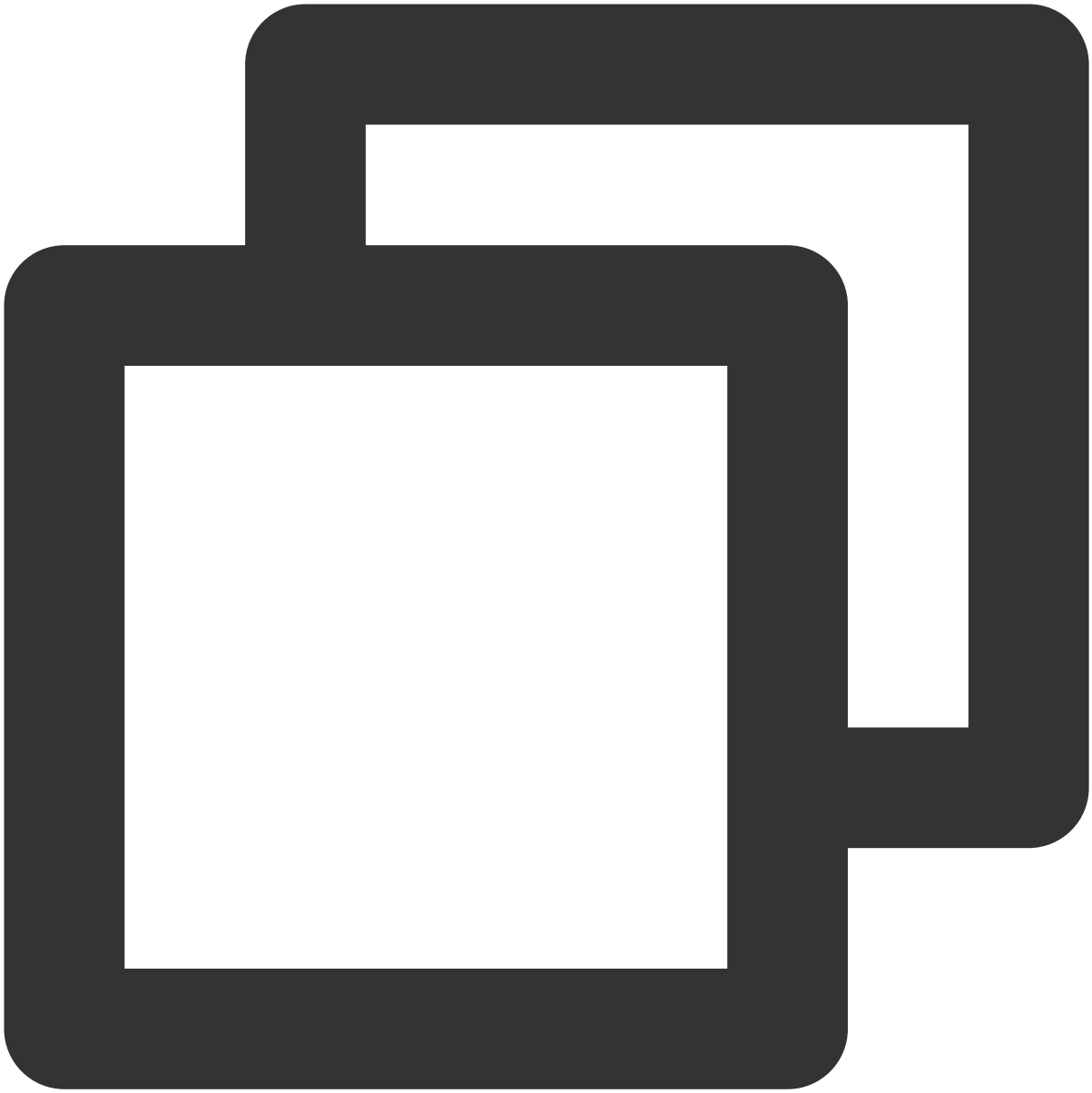
Storage should be used solely for persistent data storage, and not for runtime data transmission or global state management. Excessive synchronous read-write operations during the startup process can significantly increase the duration of startup time.

Feature Description: Stores data in the specified key within the local cache, overwriting any existing content associated with that key. The data remains available unless the user actively deletes it or it is cleared by the system due to storage space constraints. The maximum data length allowed for a single key is 1 MB, with an overall data storage limit of 10 MB.

Parameter and Description: string key, the specified key in local storage; any data, the content to be stored. Only supports native types, Date, and objects that can be serialized through `JSON.stringify` .

Sample Code

©2013-2022 Tencent Cloud. All rights reserved.



```
try {
  wx.setStorageSync('key', 'value')
} catch (e) { }
```

revokeBufferURL

This API is used via wx.revokeBufferURL(string url)

©2013-2022 Tencent Cloud. All rights reserved.

Function description: Destroy the data stored in memory according to the URL.

Parameters and description: string url, the binary data URL that needs to be destroyed.

removeStorage

This API is used via wx.removeStorage(Object object).

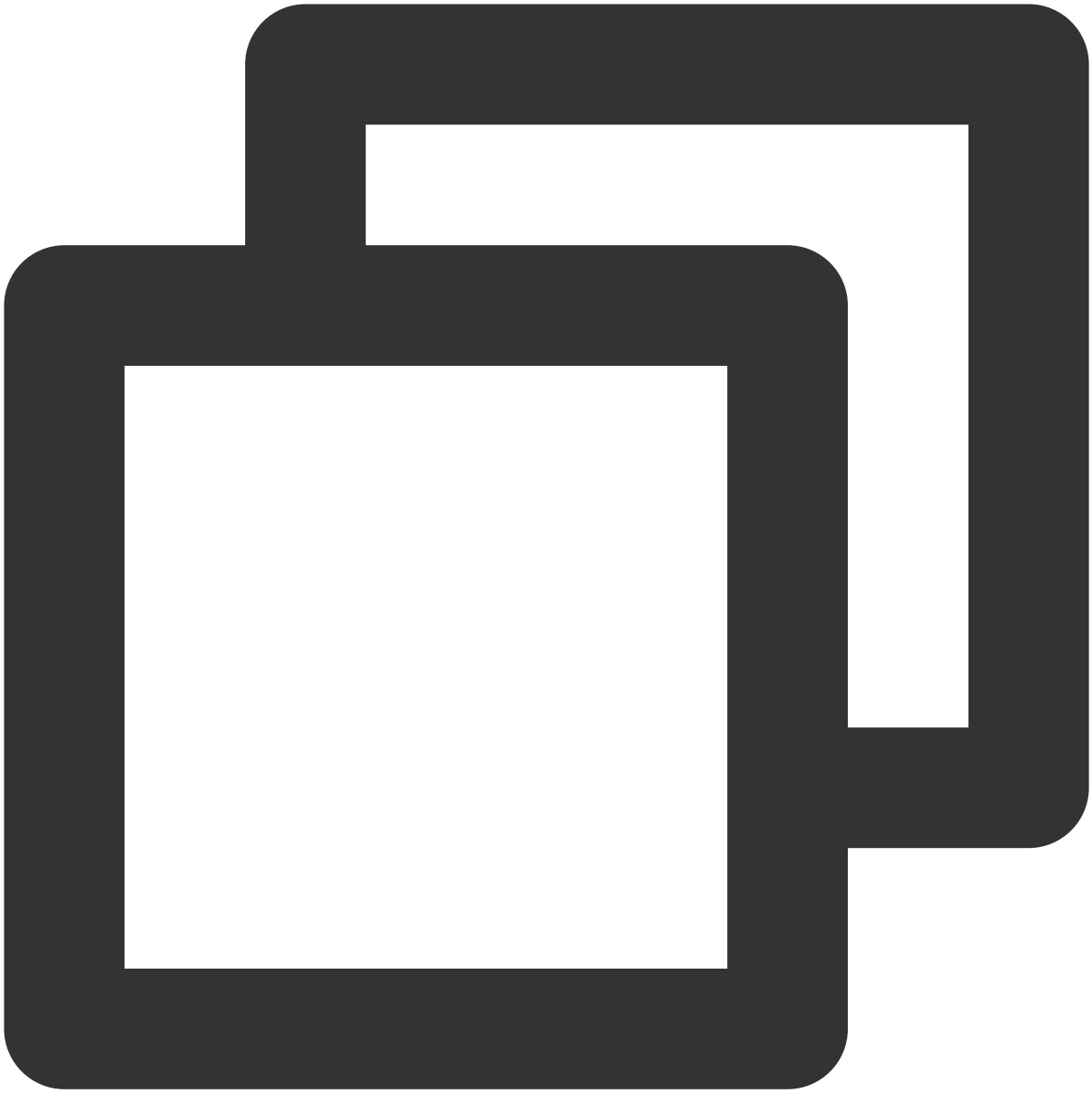
Feature Description: Removes the specified key from local storage.

Parameter and Description: Object.

Attribute	Type	Default value	Required	Description
key	string	-	Yes	Specified key in the local cache
success	function	-	No	Callback Function of Successful Interface Call
fail	function	-	No	Callback Function of Failing Interface Call
complete	function	-	No	Callback function executed upon the completion of the interface invocation (both successful and unsuccessful invocations)

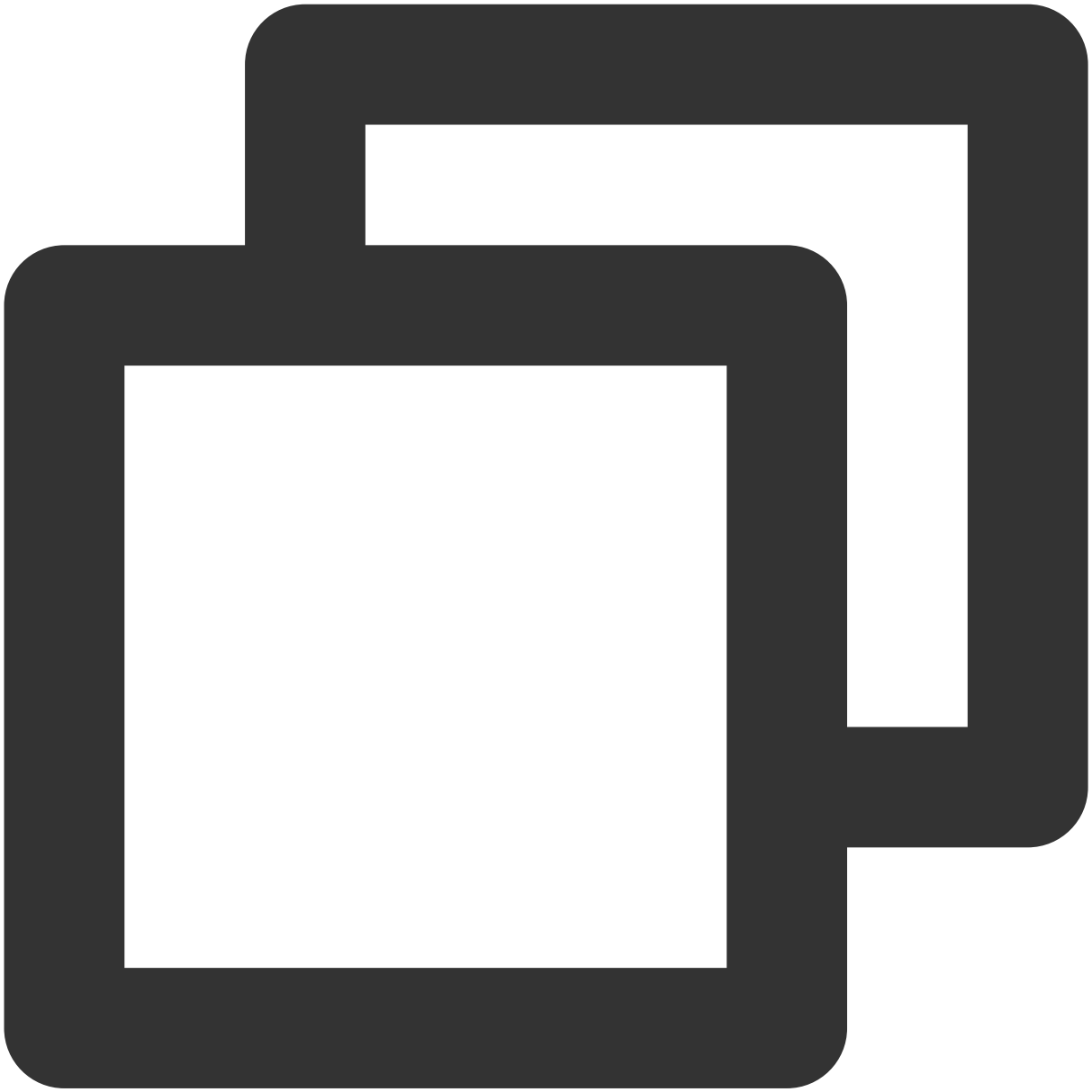
Sample Code

©2013-2022 Tencent Cloud. All rights reserved.



```
wx.removeStorage({
  key: 'key',
  success (res) {
    console.log(res)
  }
})
```

©2013-2022 Tencent Cloud. All rights reserved.



```
try {
  wx.removeStorageSync('key')
} catch(e) {
  // Do something when catch error
}
```

removeStorageSync

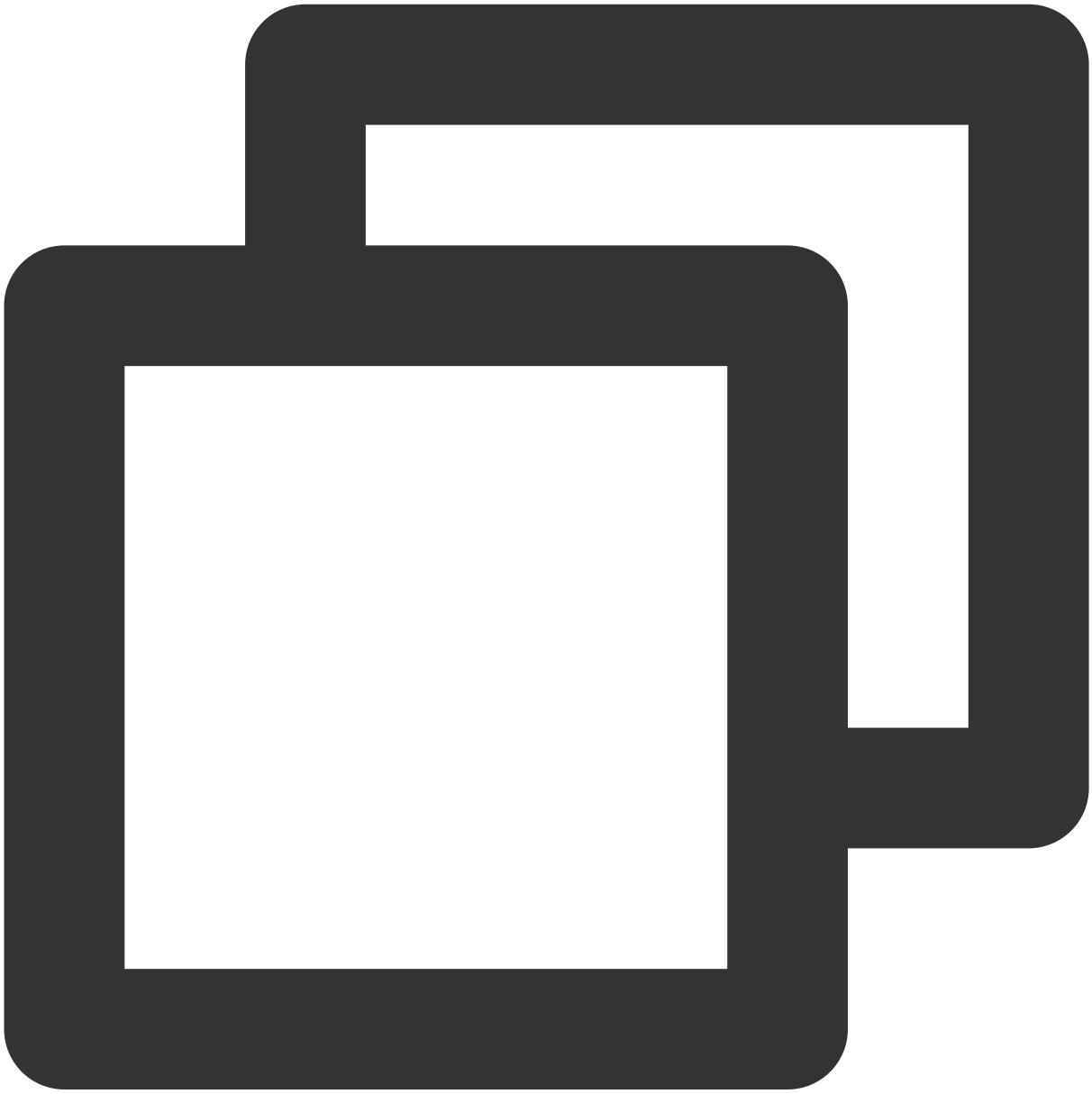
©2013-2022 Tencent Cloud. All rights reserved.

This API is used via wx.removeStorageSync(string key).

Feature Description: The synced version of wx.removeStorage.

Parameter and Description: string key, specified key in the local cache.

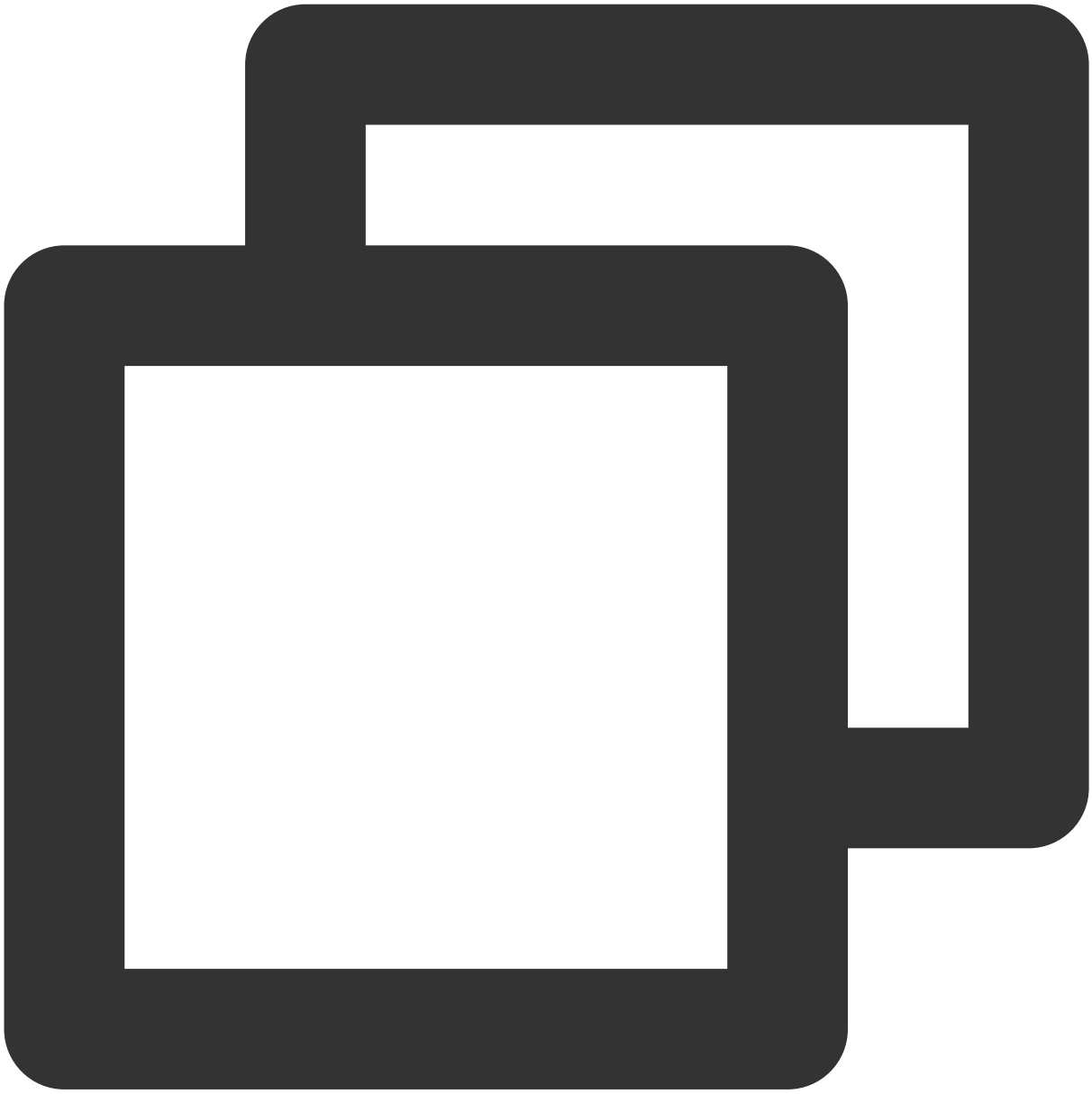
Sample Code



```
wx.removeStorage({
  key: 'key',
  success (res) {
```

©2013-2022 Tencent Cloud. All rights reserved.

```
    console.log(res)
  }
})
```



```
try {
  wx.removeStorageSync('key')
} catch(e) {
  // Do something when catch error
}
```

©2013-2022 Tencent Cloud. All rights reserved.

getStorage

This API is used via wx.getStorage(Object object).

Feature Description: Asynchronously gets the content of the specified key from local storage.

Parameter and Description: Object.

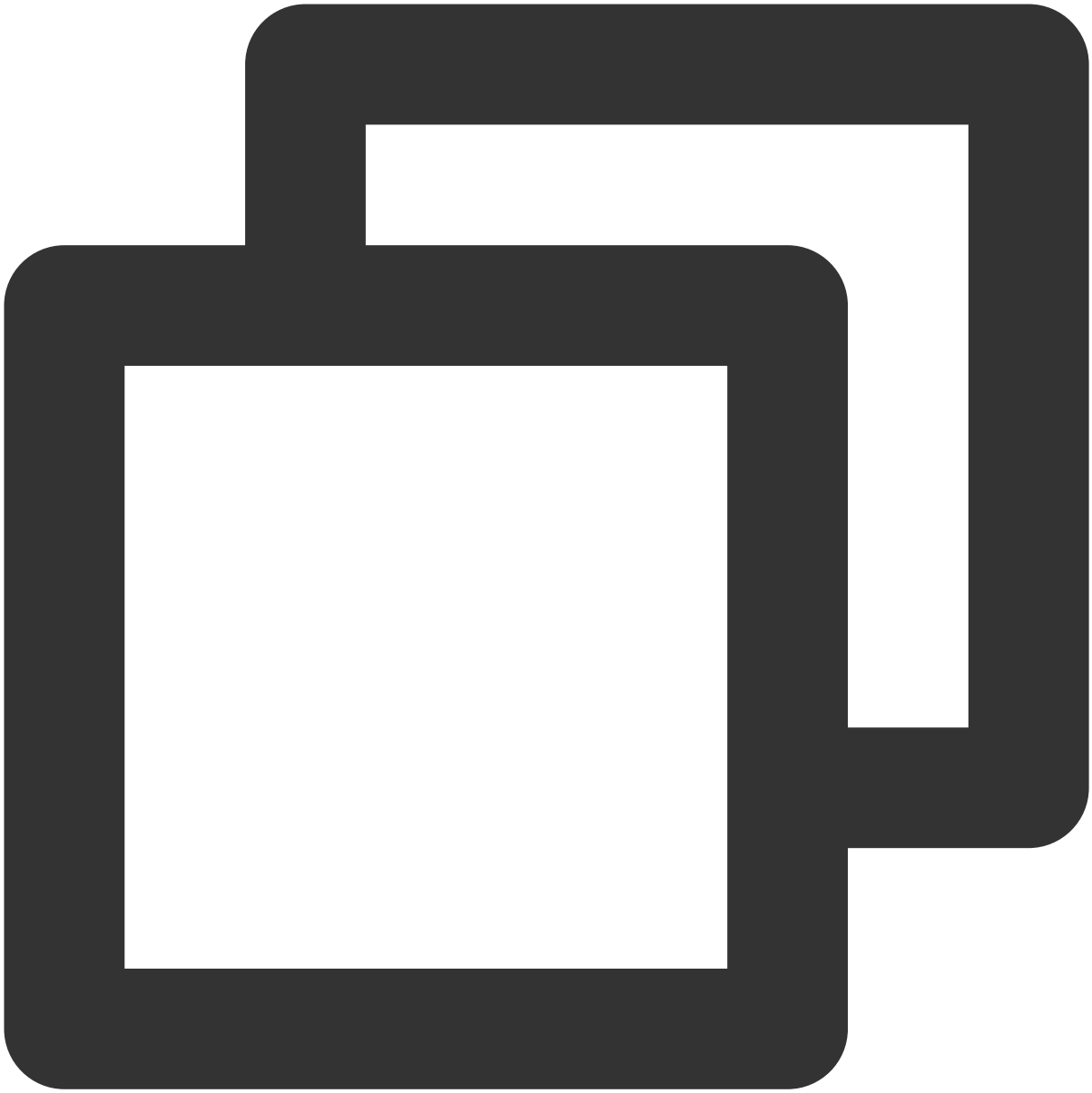
Attribute	Type	Default value	Required	Description
key	string	-	Yes	Specified key in the local cache
success	function	-	No	Callback Function of Successful Interface Call
fail	function	-	No	Callback Function of Failing Interface Call
complete	function	-	No	Callback function executed upon the completion of the interface invocation (both successful and unsuccessful invocations)

Parameters for object.success callback function: Object res.

Attribute	Type	Description
data	any	The content corresponding to the key.

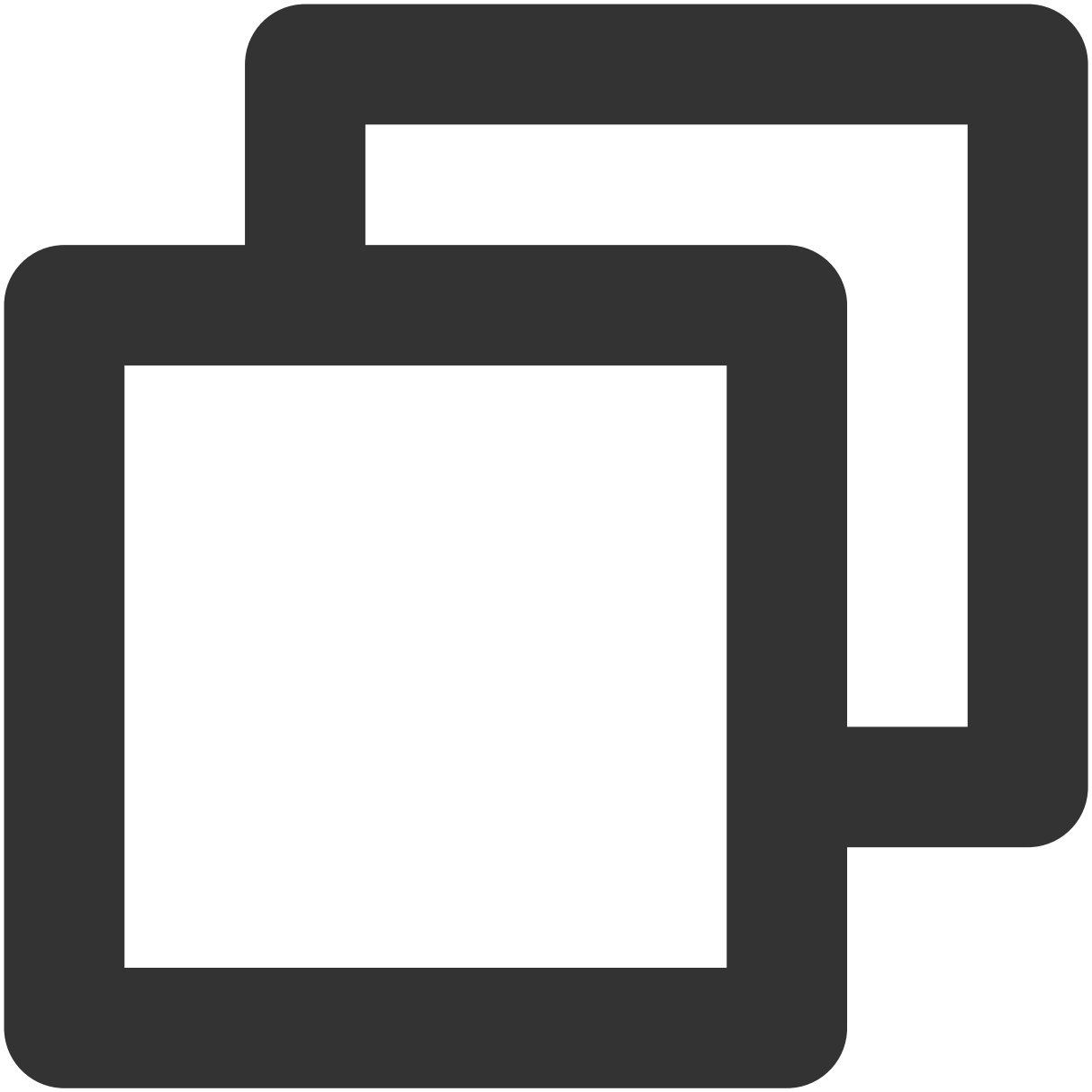
Sample Code

©2013-2022 Tencent Cloud. All rights reserved.



```
wx.getStorage({
  key: 'key',
  success (res) {
    console.log(res.data)
  }
})
```

©2013-2022 Tencent Cloud. All rights reserved.



```
wx.getStorage({
  key: 'key',
  success (res) {
    console.log(res.data)
  }
})
```

createBufferURL

©2013-2022 Tencent Cloud. All rights reserved.

This API is used via string wx.createBufferURL(ArrayBuffer|TypedArray buffer)

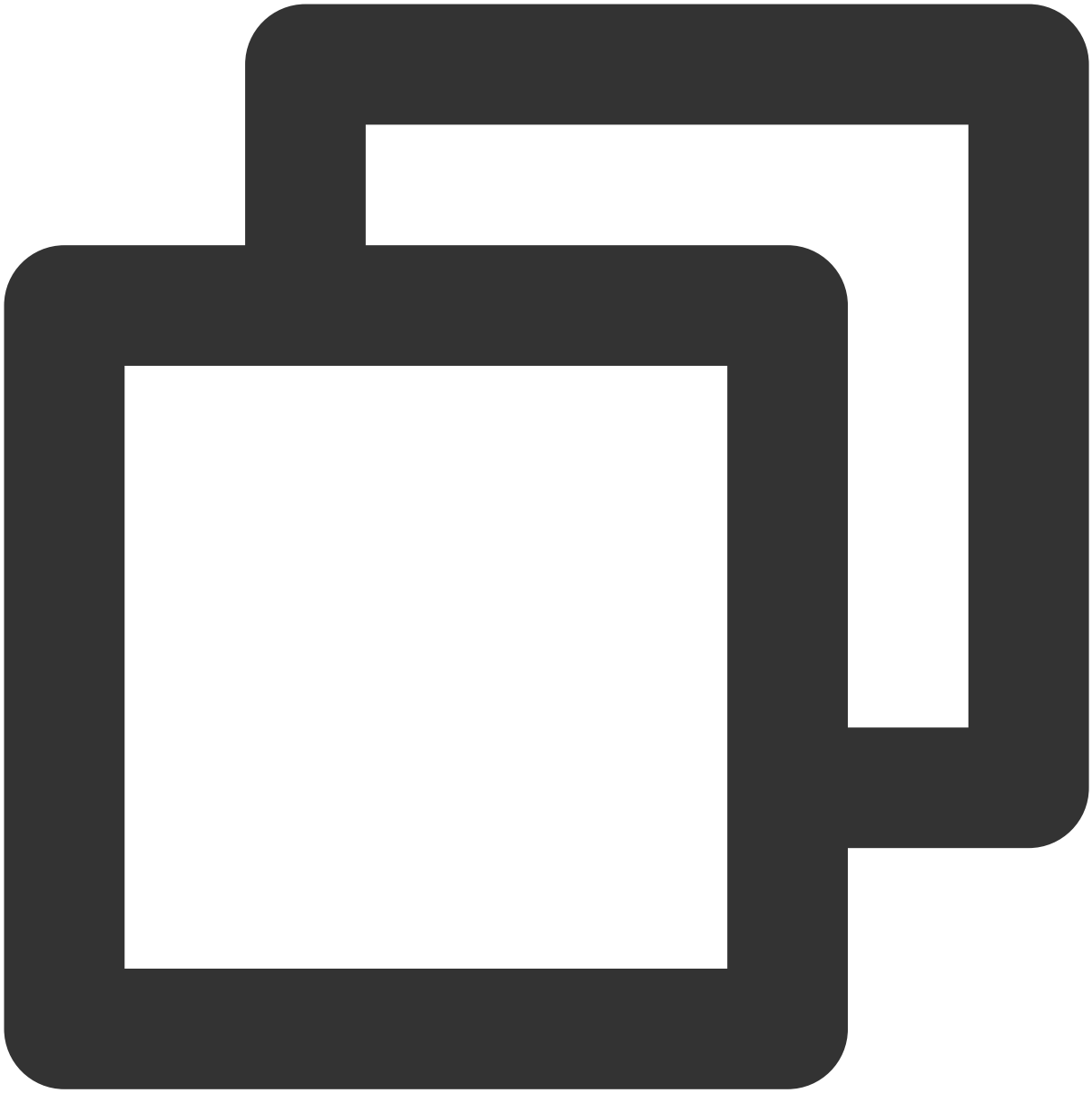
Function description: Create a unique URL based on the incoming buffer and store it in memory.
Parameters and description: ArrayBuffer|TypedArray buffer, binary data that needs to be stored in memory.
Return value: string.

getStorageSync

This API is used via any wx.getStorageSync(string key).

Note：
storage should only be used for persistent storage of data and should not be used for runtime data transfer or global state management. Excessive synchronization of read and write storage during startup can significantly impact startup elapsed time.
Feature Description: Synchronously gets the content of the specified key from local storage.
Parameter and Description: string key, specified key in the local cache.
Return Value: Any key, corresponding content.
Sample Code

©2013-2022 Tencent Cloud. All rights reserved.



```
try {
  var value = wx.getStorageSync('key')
  if (value) {
    // Do something with return value
  }
} catch(e) {
  // Do something when catch error
}
```

©2013-2022 Tencent Cloud. All rights reserved.

getStorageInfo

This API is used via wx.getStorageInfo(Object object).

Feature Description: Asynchronously obtains relevant information about the current storage.

Parameter and Description: Object.

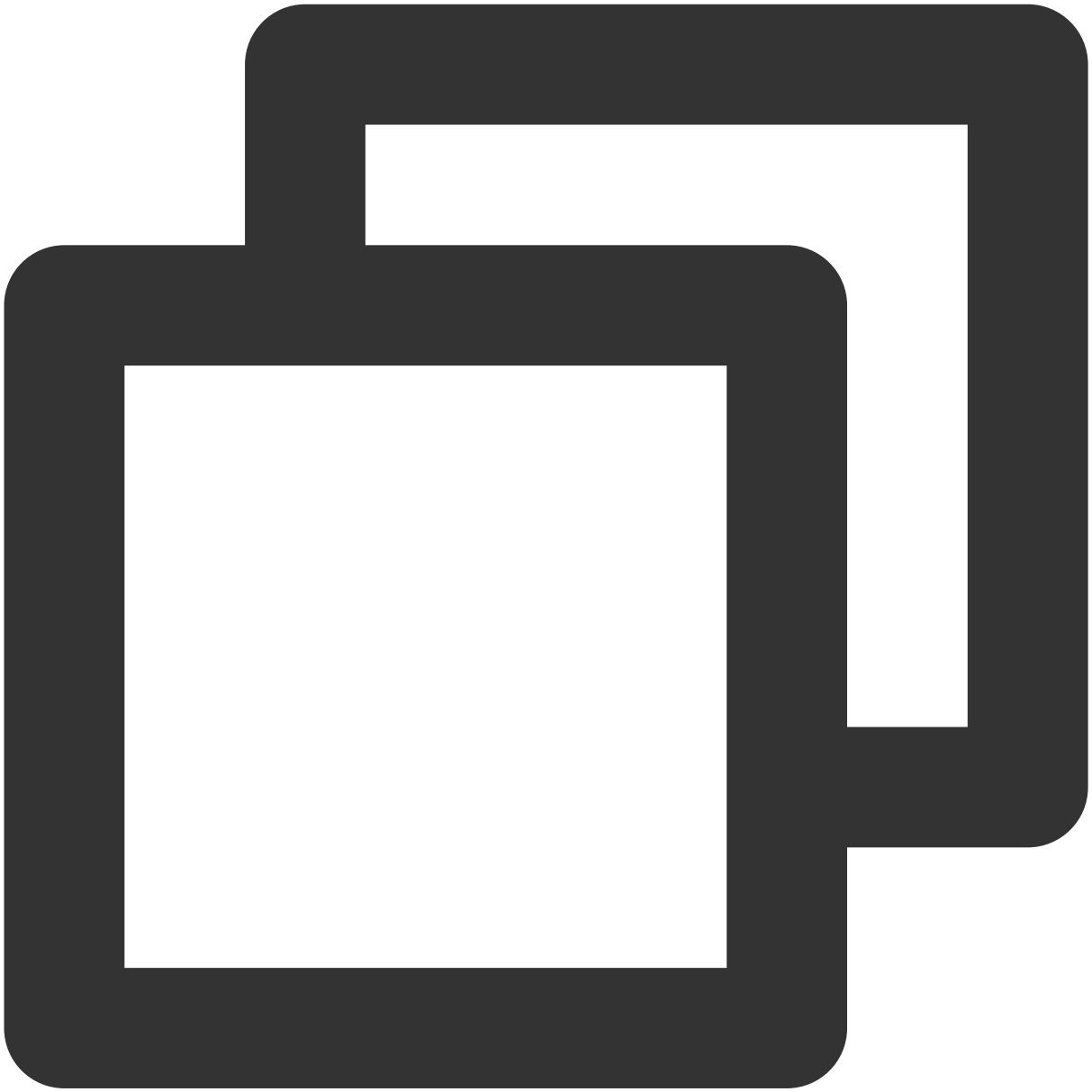
Attribute	Type	Default value	Required	Description
success	function	-	No	Callback Function of Successful Interface Call
fail	function	-	No	Callback Function of Failing Interface Call
complete	function	-	No	Callback function executed upon the completion of the interface invocation (both successful and unsuccessful invocations)

Parameters for object.success callback function: Object.

Attribute	Type	Description
keys	Array.<string>	All keys in the current storage.
currentSize	number	Current occupied space size, in kilobytes (KB).
limitSize	number	Restricted space size, in kilobytes (KB).

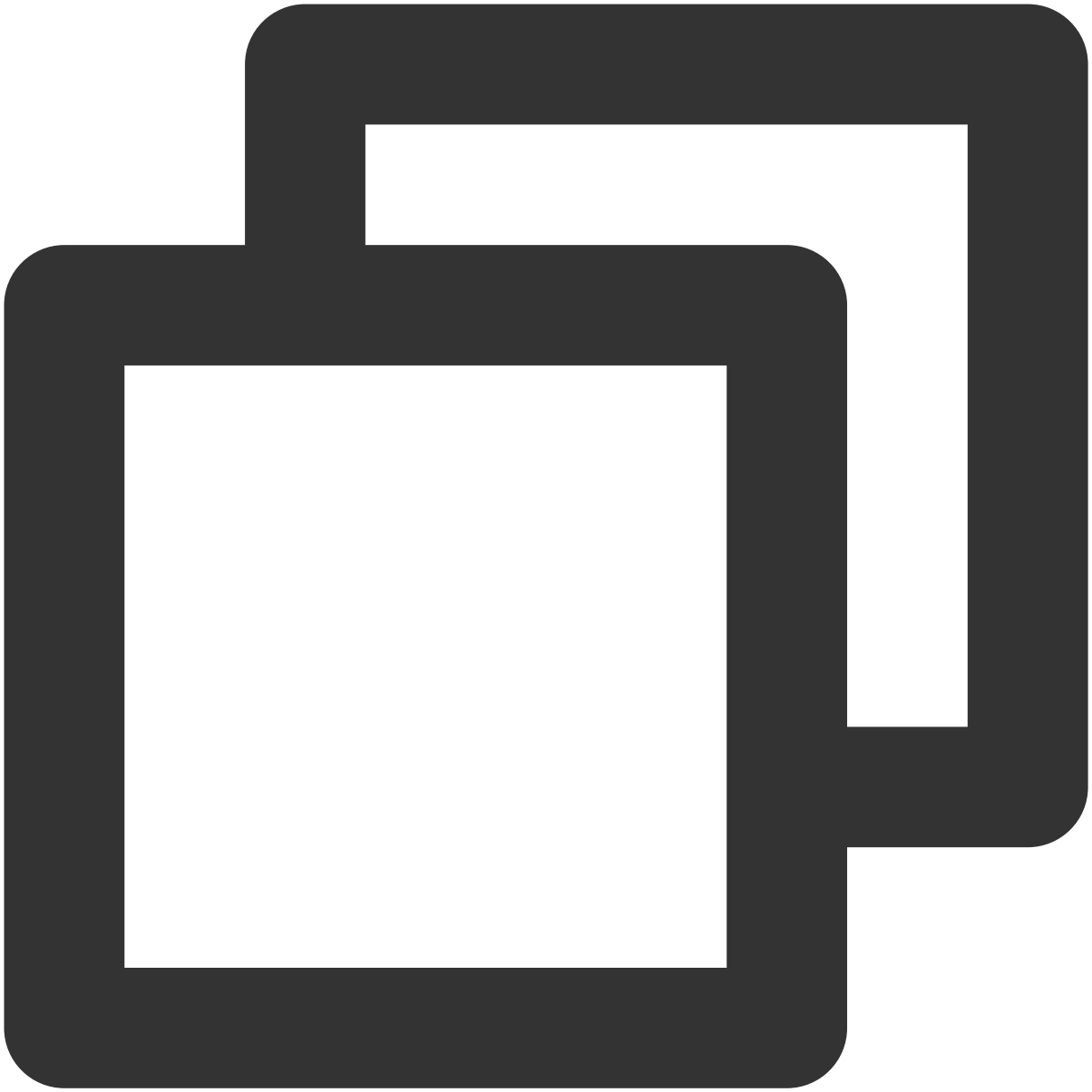
Sample Code

©2013-2022 Tencent Cloud. All rights reserved.



```
wx.getStorageInfo({
  success (res) {
    console.log(res.keys)
    console.log(res.currentSize)
    console.log(res.limitSize)
  }
})
```

©2013-2022 Tencent Cloud. All rights reserved.



```
wx.getStorageInfo({
  success (res) {
    console.log(res.keys)
    console.log(res.currentSize)
    console.log(res.limitSize)
  }
})
```

©2013-2022 Tencent Cloud. All rights reserved.

getStorageInfoSync

This API is used via Object wx.getStorageInfoSync().

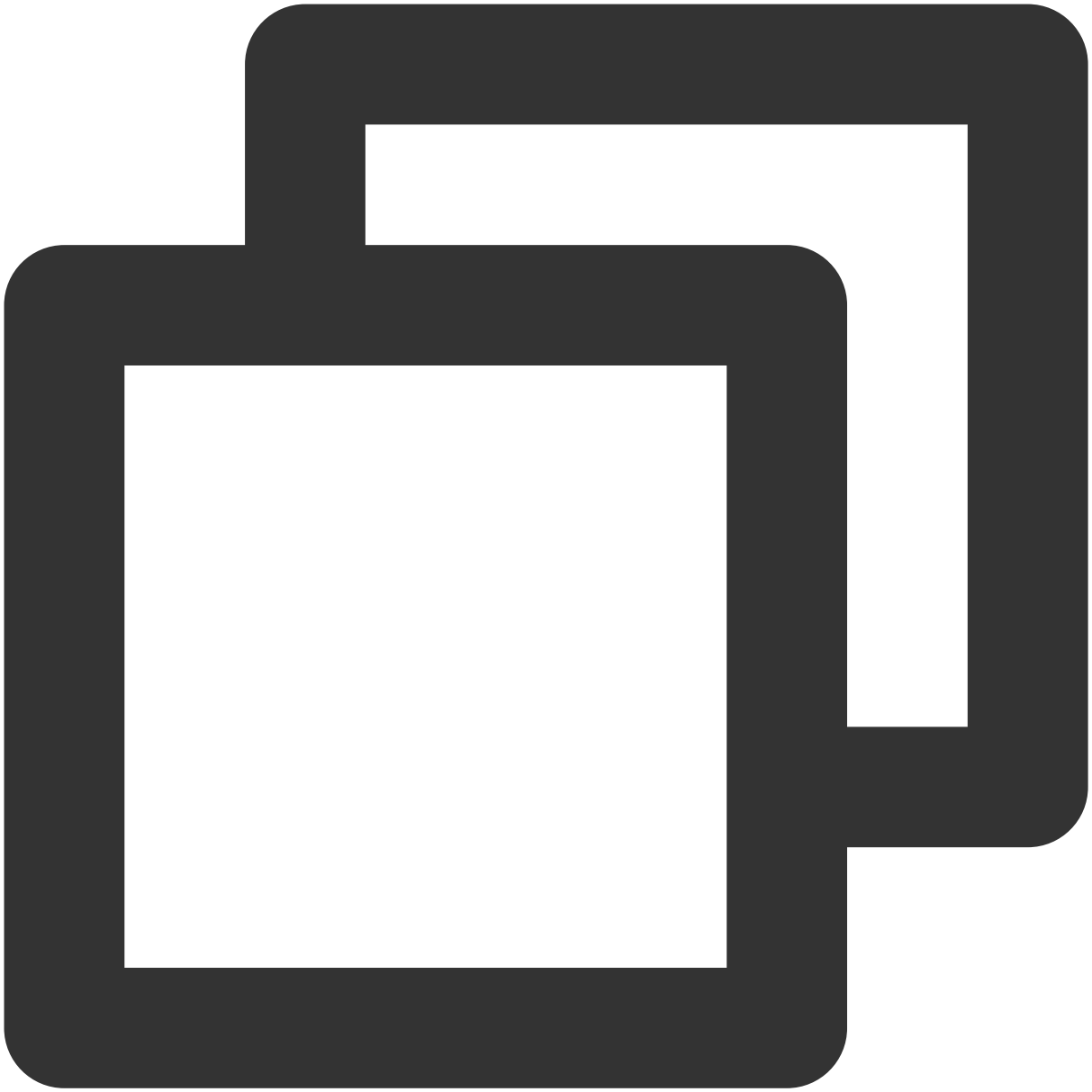
Function Description: The synced version of wx.getStorageInfo.

Return Value: Object.

Attribute	Type	Description
keys	Array.	All keys in the current storage.
currentSize	number	Current occupied space size, in kilobytes (KB).
limitSize	number	Restricted space size, in kilobytes (KB).

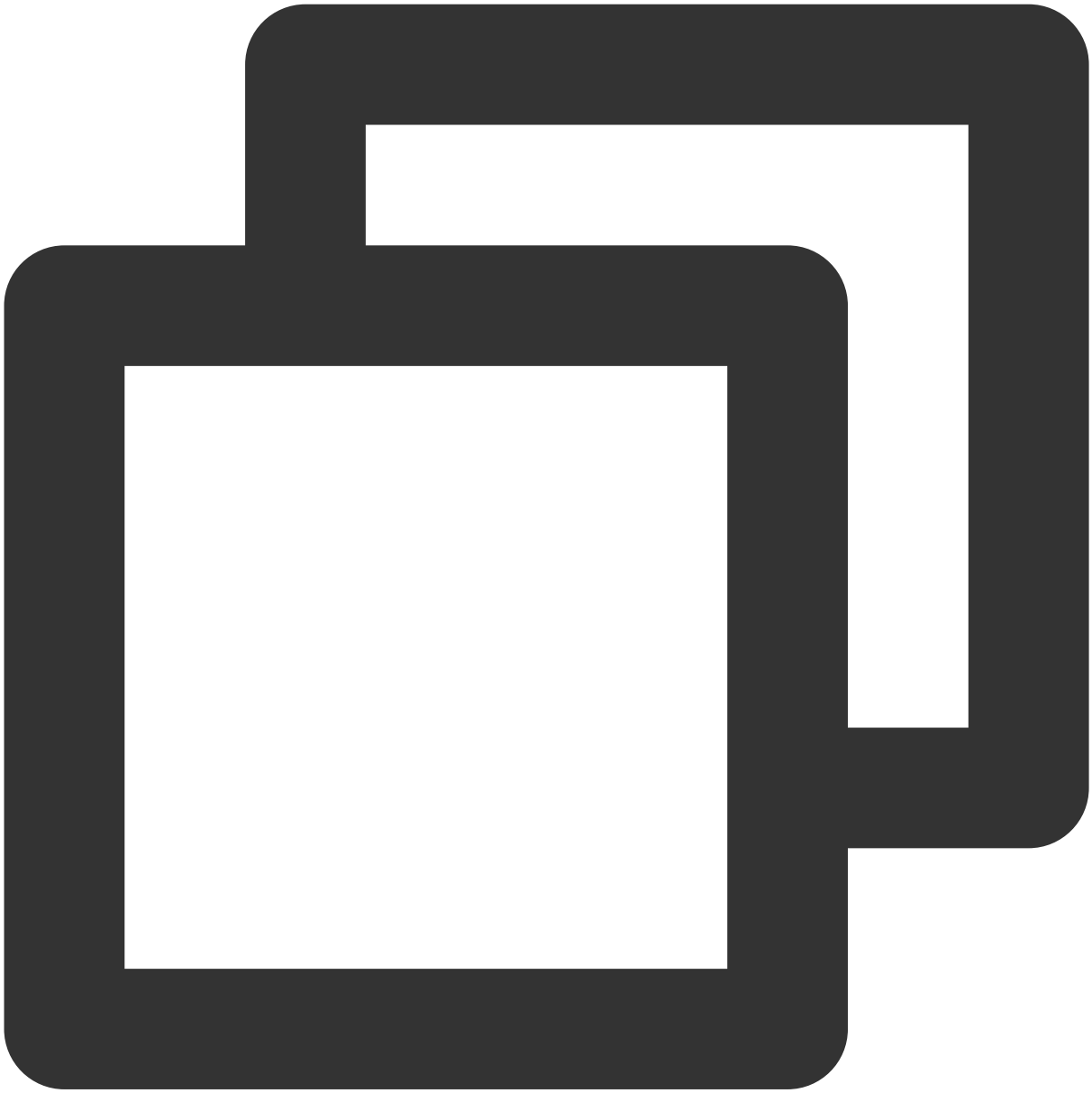
Sample Code

©2013-2022 Tencent Cloud. All rights reserved.



```
wx.getStorageInfo({
  success (res) {
    console.log(res.keys)
    console.log(res.currentSize)
    console.log(res.limitSize)
  }
})
```

©2013-2022 Tencent Cloud. All rights reserved.



```
try {
  const res = wx.getStorageInfoSync()
  console.log(res.keys)
  console.log(res.currentSize)
  console.log(res.limitSize)
} catch(e) {
  // Do something when catch error
}
```

©2013-2022 Tencent Cloud. All rights reserved.

clearStorage

This API is used via wx.clearStorage(Object object).

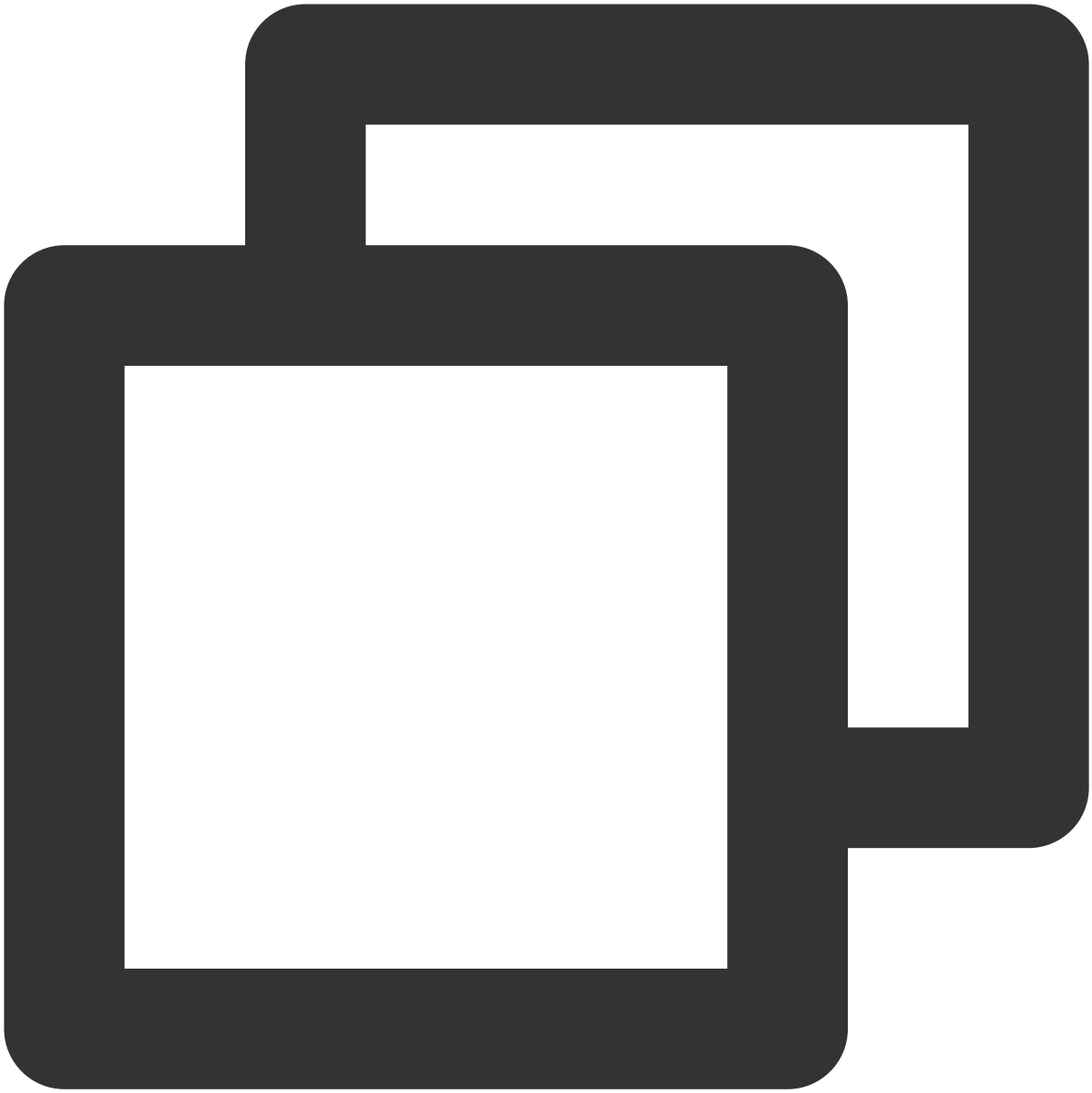
Feature Description: Clears local data cache.

Parameter and Description: Object.

Attribute	Type	Default value	Required	Description
success	function	-	No	Callback Function of Successful Interface Call
fail	function	-	No	Callback Function of Failing Interface Call
complete	function	-	No	Callback function executed upon the completion of the interface invocation (both successful and unsuccessful invocations)

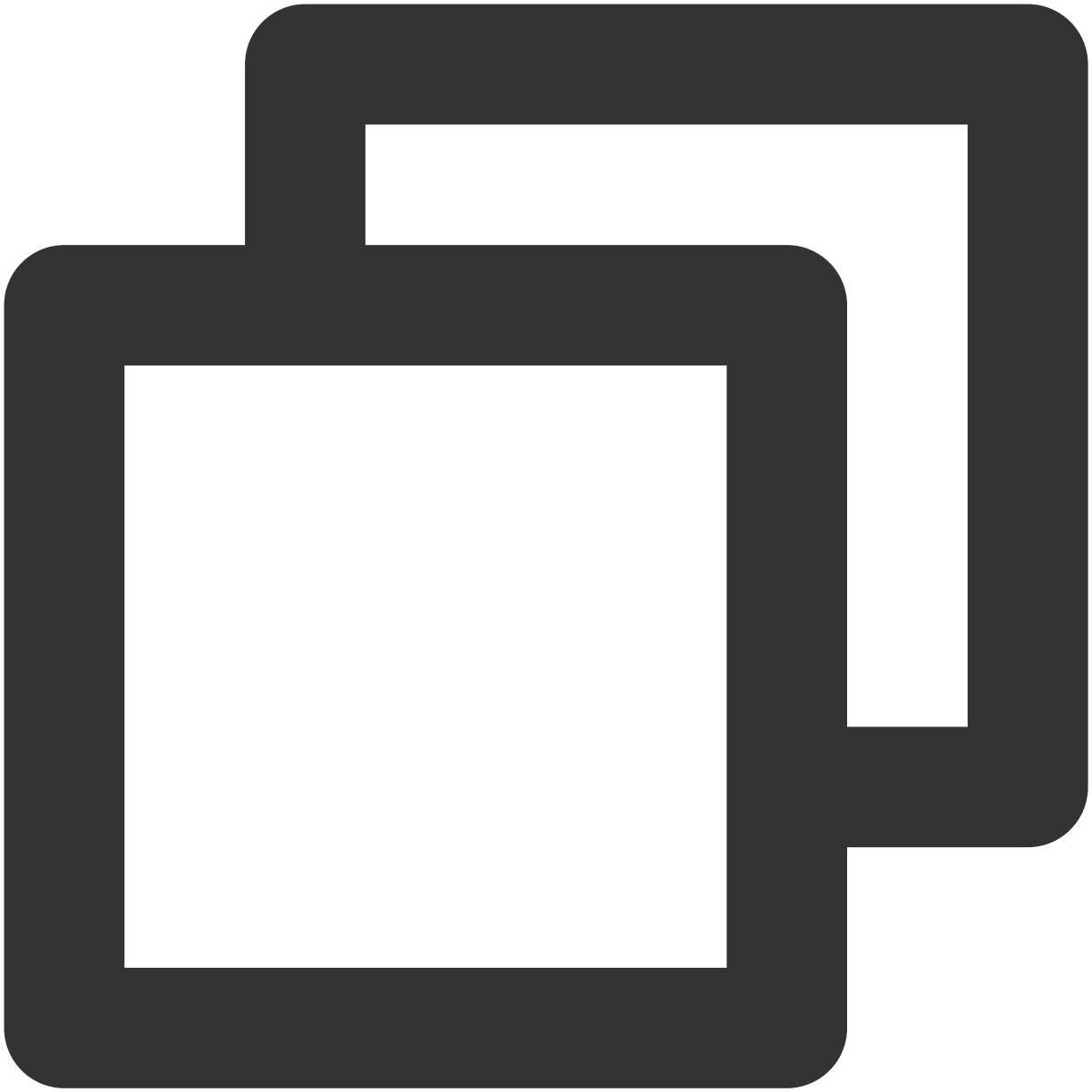
Sample Code

©2013-2022 Tencent Cloud. All rights reserved.



```
wx.clearStorage()
```

©2013-2022 Tencent Cloud. All rights reserved.



```
try {
  wx.clearStorageSync()
} catch(e) {
  // Do something when catch error
}
```

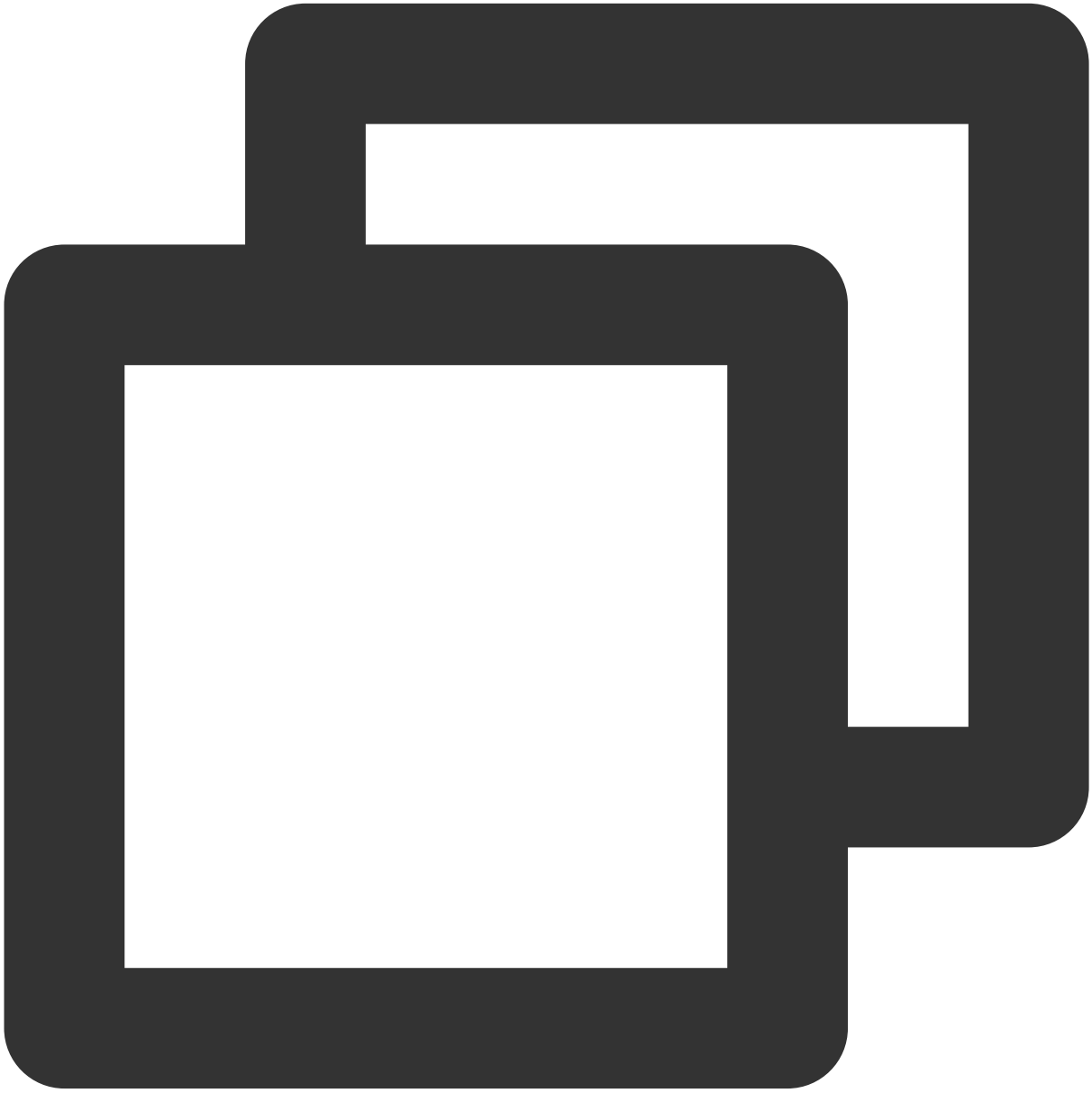
clearStorageSync

©2013-2022 Tencent Cloud. All rights reserved.

This API is used via wx.clearStorageSync().

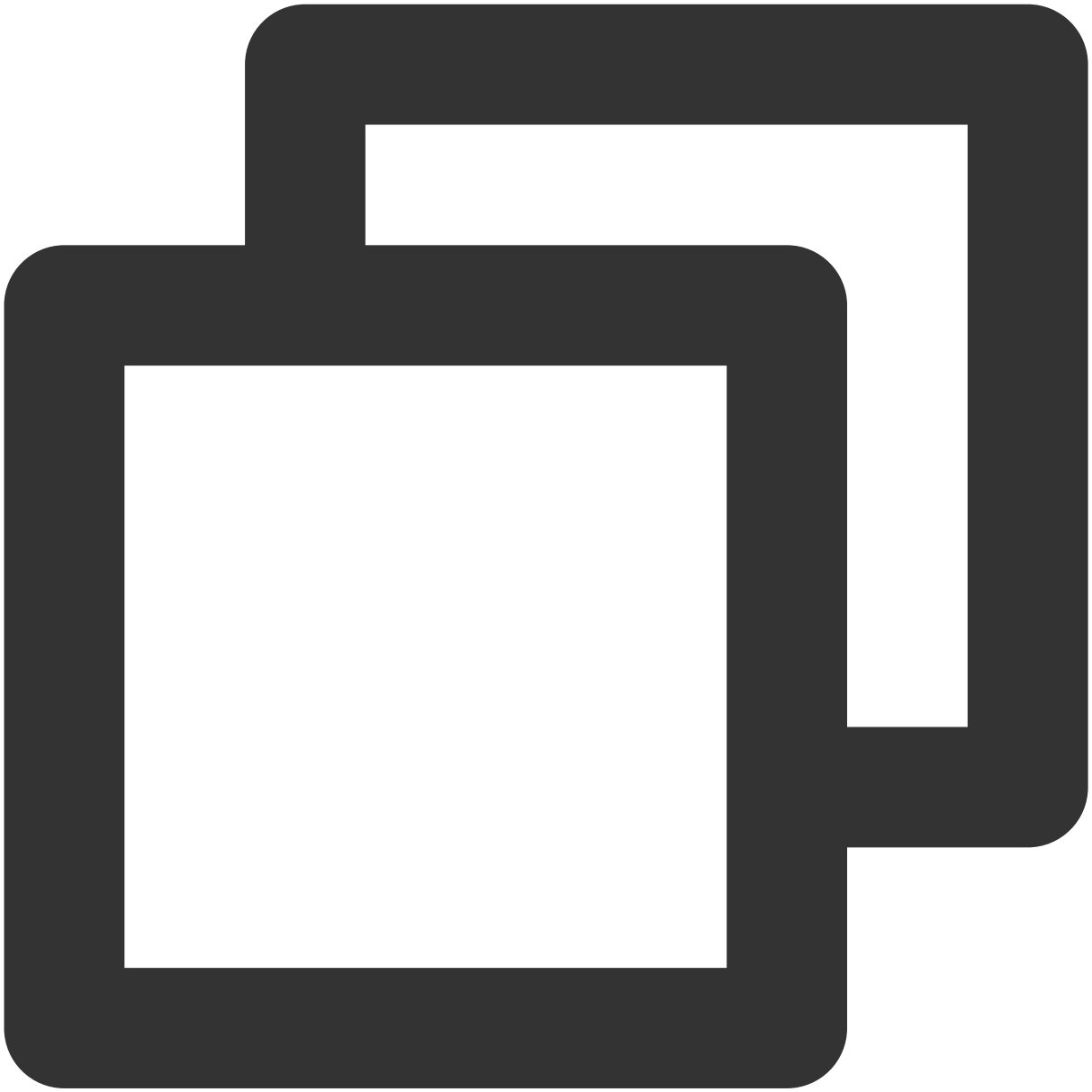
Function Description: The synced version of wx.clearStorage.

Sample Code



```
wx.clearStorage()
```

©2013-2022 Tencent Cloud. All rights reserved.



```
try {
  wx.clearStorageSync()
} catch(e) {
  // Do something when catch error
}
```

batchSetStorage

©2013-2022 Tencent Cloud. All rights reserved.

This API is used via wx.batchSetStorage(Object object).

Feature Description: Batch stores data in the specified key within the local cache, overwriting any existing content associated with that key. The data remains available unless the user actively deletes it or it is cleared by the system due to storage space constraints. The maximum data length allowed for a single key is 1 MB, with an overall data storage limit of 10 MB.

Parameter and Description: Object

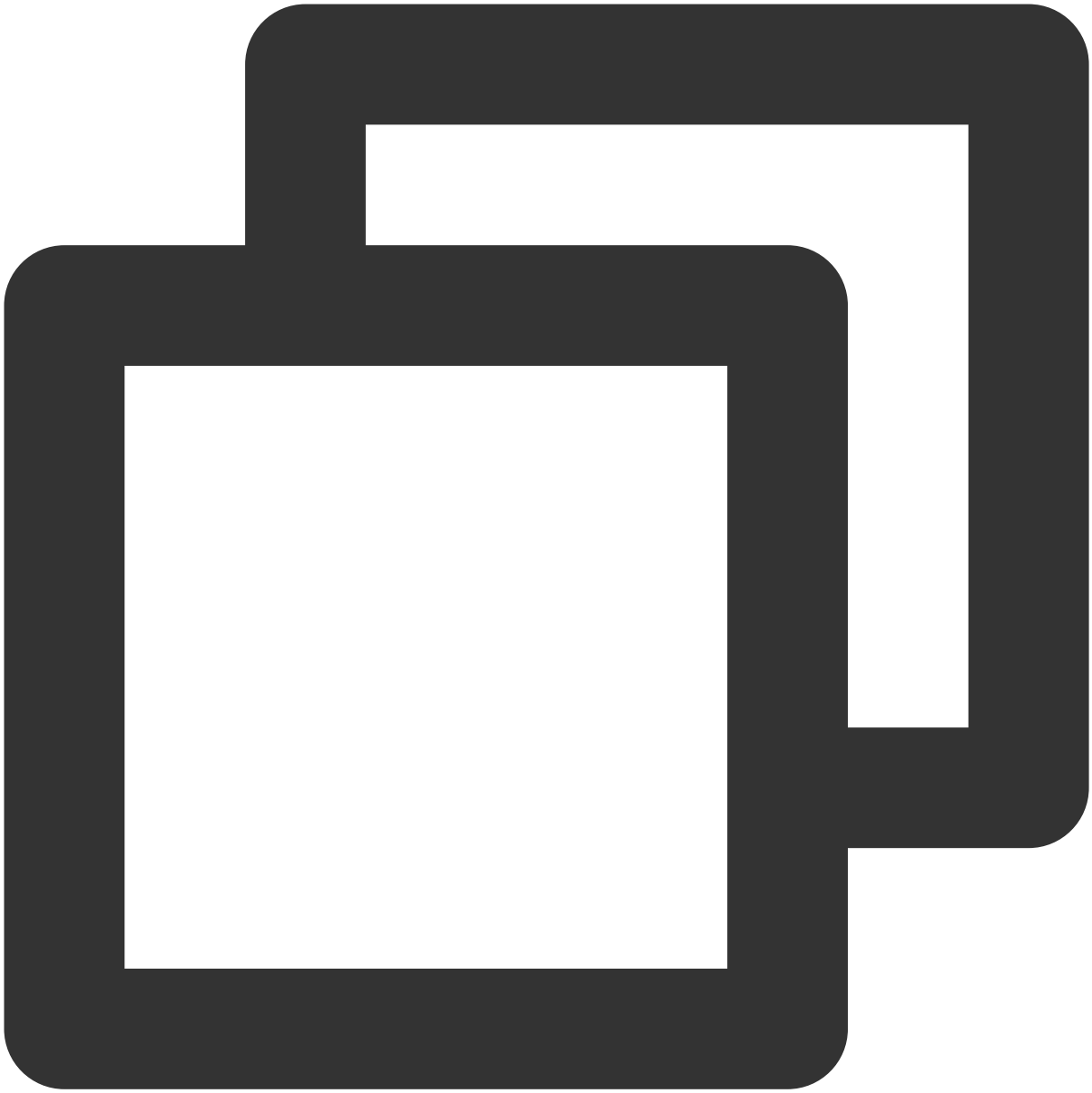
Attribute	Type	Default value	Required	Description
kvList	Array	-	Yes	[{ key, value }]
success	function	-	No	Callback Function of Successful Interface Call
fail	function	-	No	Callback Function of Failing Interface Call
complete	function	-	No	Callback function executed upon the completion of the interface invocation (both successful and unsuccessful invocations)

Array.<Object> kvList

Attribute	Type	Default value	Required	Description
key	string	-	Yes	key The key specified in the local cache.
value	any	-	Yes	data The content to be stored. Only native types, dates, and objects that can be serialized with JSON.stringify are supported.

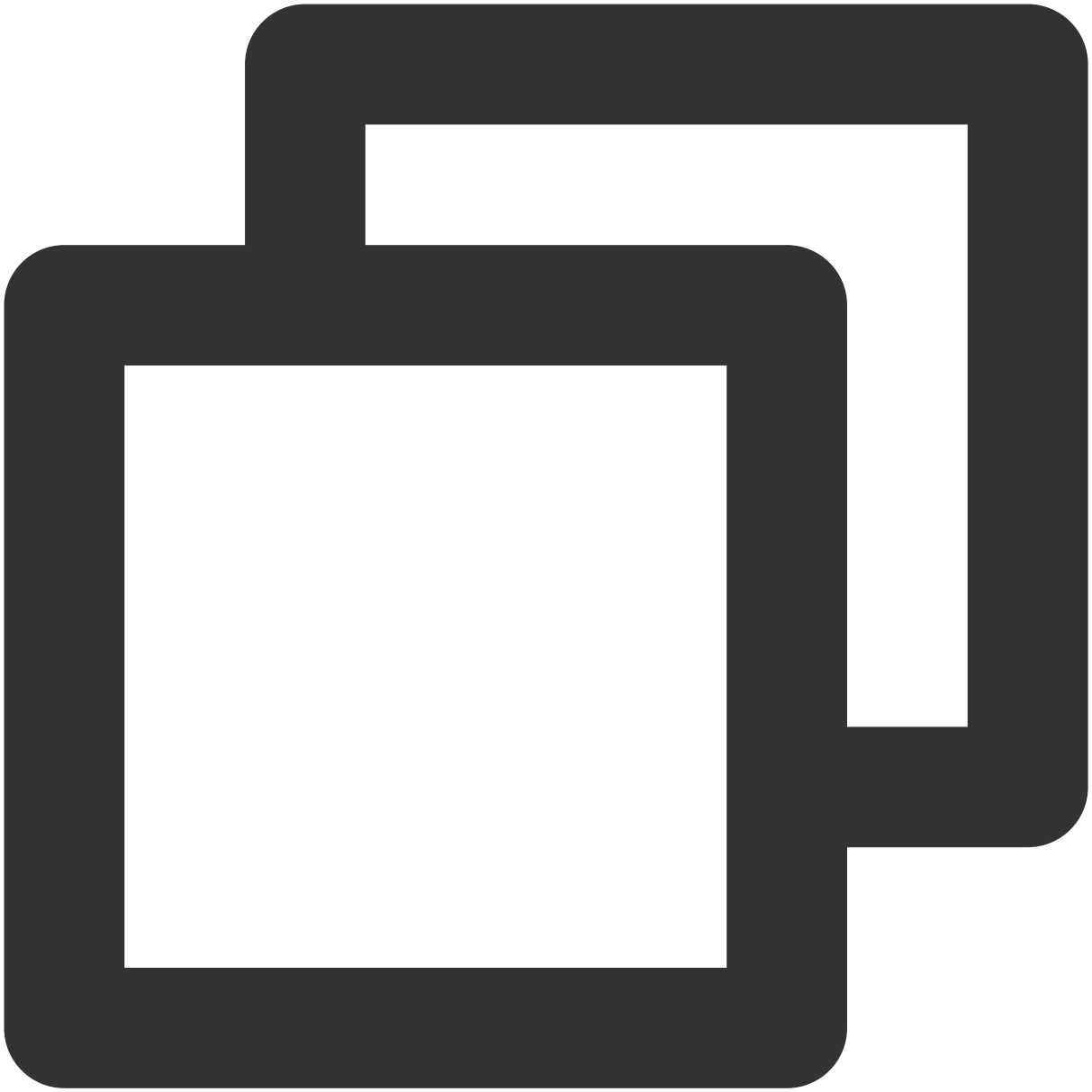
Sample Code

©2013-2022 Tencent Cloud. All rights reserved.



```
wx.setStorage({
  key:"key",
  data:"value"
})
```

©2013-2022 Tencent Cloud. All rights reserved.



```
// Enable encrypted storage
wx.batchSetStorage({
  kvList: [{
    key: 'key',
    value: 'value',
  }],
})
```

©2013-2022 Tencent Cloud. All rights reserved.

batchSetStorageSync

This API is used via Array.<any> wx.batchGetStorageSync(Array.<string> keyList).

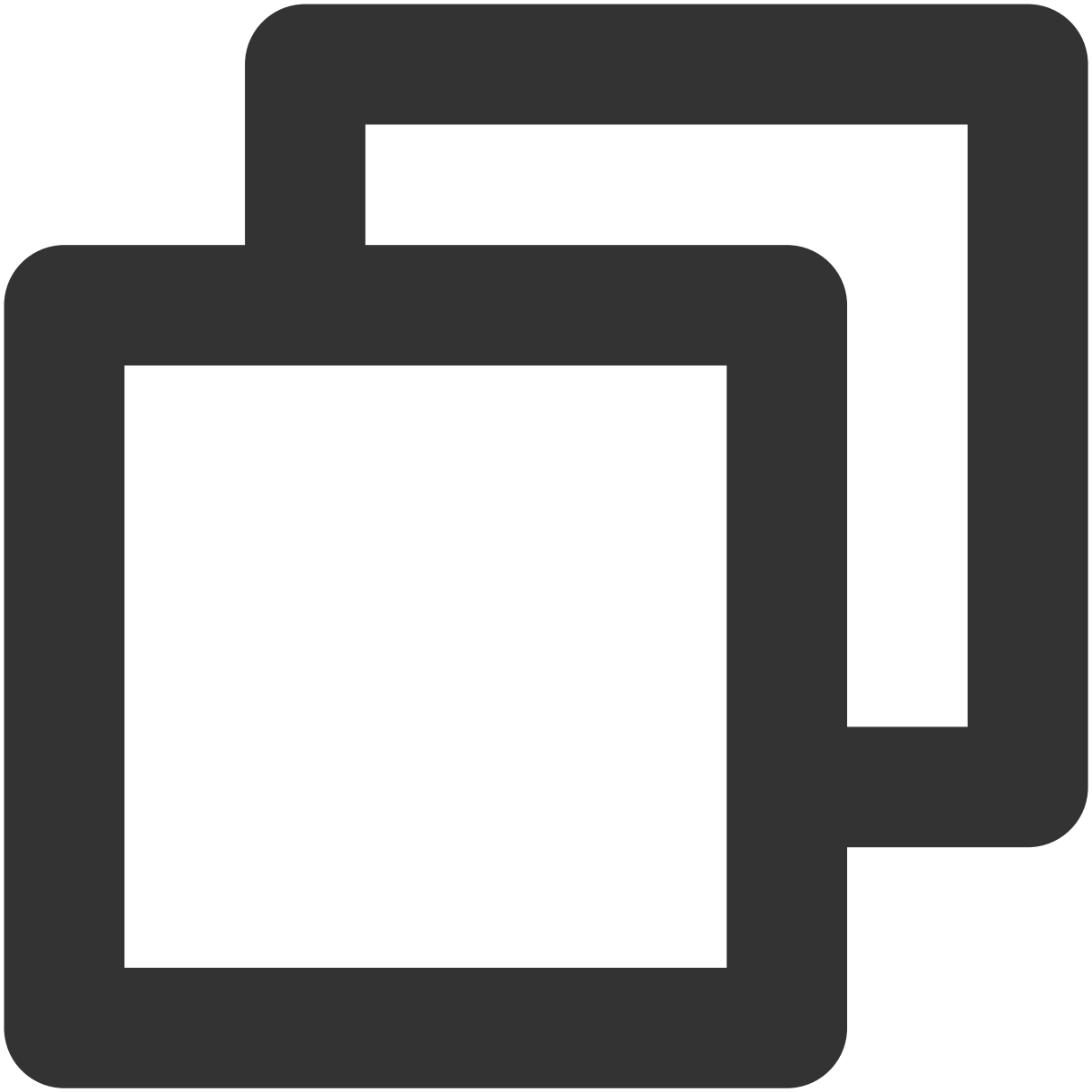
Feature Description: Batch stores data in the specified key within the local cache, overwriting any existing content associated with that key. The data remains available unless the user actively deletes it or it is cleared by the system due to storage space constraints. The maximum data length allowed for a single key is 1 MB, with an overall data storage limit of 10 MB.

Parameters and Description: Array.<Object> kvList.

Attribute	Type	Default value	Required	Description
key	string	-	Yes	Key: Specified key in the local cache.
value	any	-	Yes	The content that needs to be stored. Only native types, Date, and objects that can be serialized through JSON.stringify are supported.

Sample Code

©2013-2022 Tencent Cloud. All rights reserved.



```
try {
  wx.batchSetStorageSync([{key: 'key', value: 'value'}])
} catch (e) { }
```

batchGetStorage

This API is used via wx.batchGetStorage(Object object).

©2013-2022 Tencent Cloud. All rights reserved.

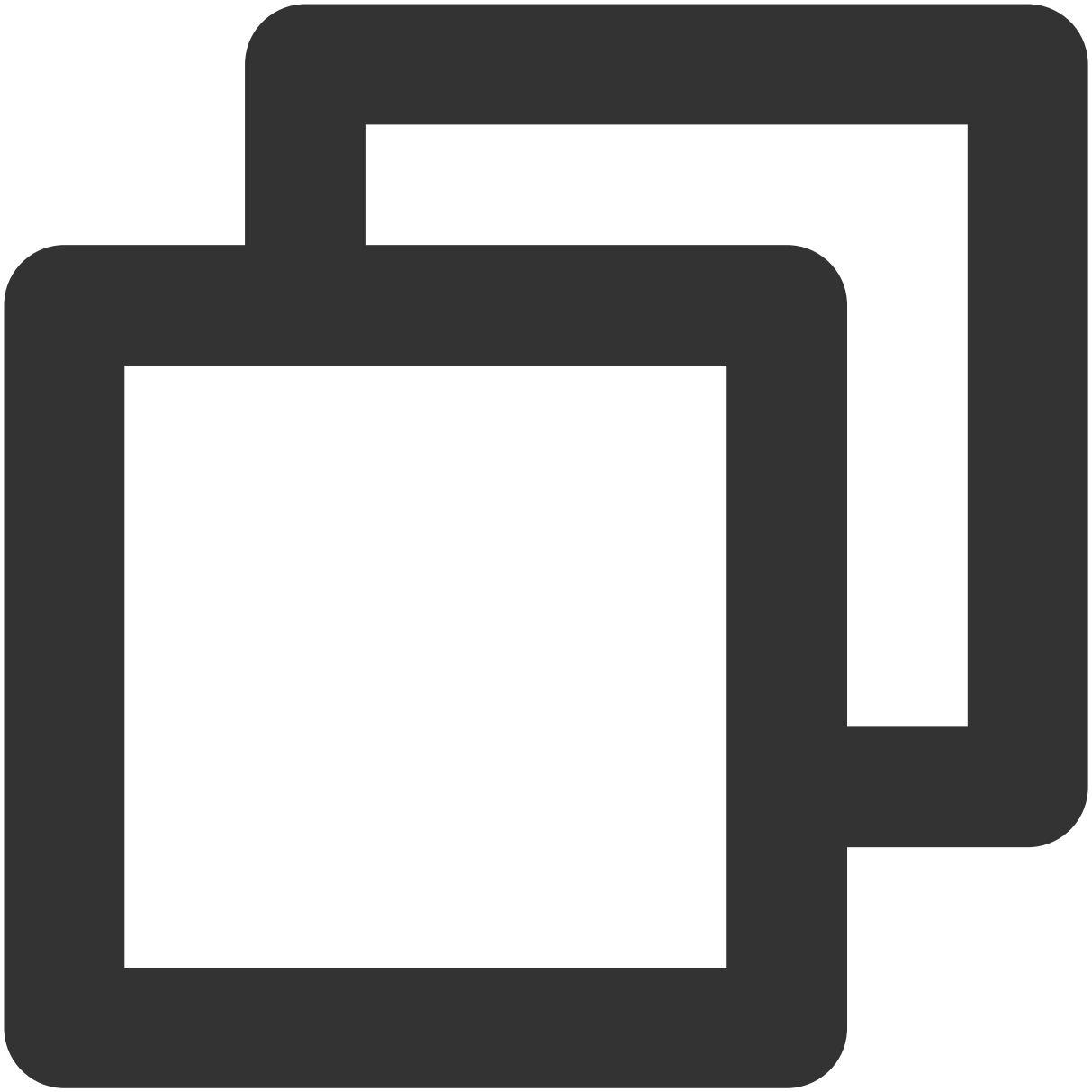
Feature Description: Asynchronously batch gets the content of the specified key from local storage.

Parameter and Description: Object.

Attribute	Type	Default value	Required	Description
keyList	Array.<string>	-	Yes	Specified keyList in the local cache
success	function	-	No	Callback Function of Successful Interface Call
fail	function	-	No	Callback Function of Failing Interface Call
complete	function	-	No	Callback function executed upon the completion of the interface invocation (both successful and unsuccessful invocations)

Sample Code

©2013-2022 Tencent Cloud. All rights reserved.



```
wx.batchGetStorage({
  keyList: ['key'],
  success (res) {
    console.log(res)
}
})
```

batchGetStorageSync

©2013-2022 Tencent Cloud. All rights reserved.

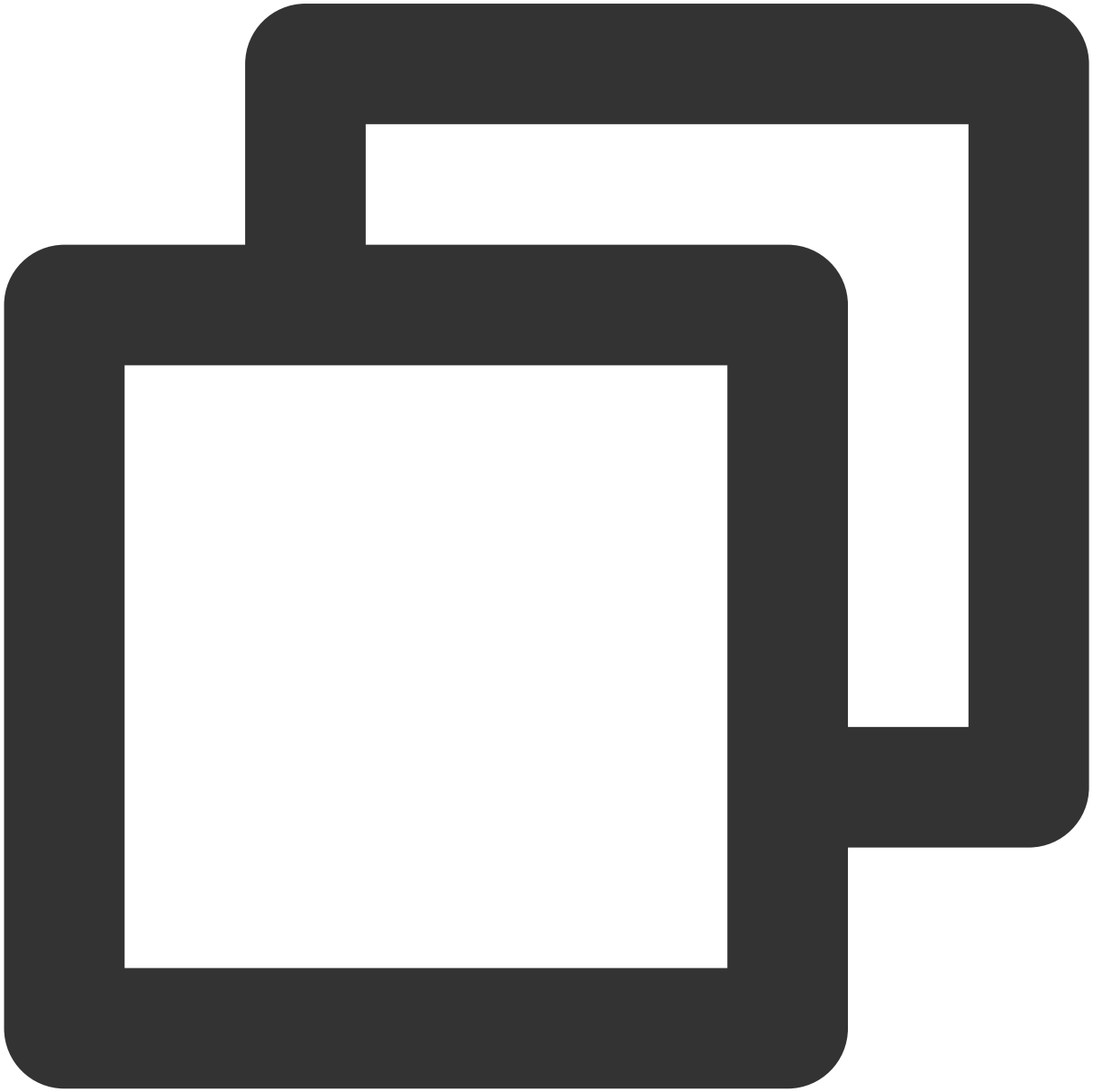
This API is used via Array.<any> wx.batchGetStorageSync(Array.<string> keyList).

Feature Description: Synchronously batch gets the content of the specified key from the local cache.

Parameter Description: Array.<string> keyList, specified array of keys in local cache.

Return value: Array.<any>, corresponding content of the key.

Sample Code



```
// For reading multiple keys, batch reading is superior in performance to multiple
```

©2013-2022 Tencent Cloud. All rights reserved.

```
try {
  var valueList = wx.batchGetStorageSync(['key'])
  if (valueList) {
    // Do something with return value
}
} catch (e) {
  // Do something when catch error
}
```

©2013-2022 Tencent Cloud. All rights reserved.

Data Analysis

Last updated：2024-07-12 19:26:18

reportEvent

This API is used via wx.reportEvent(string eventId, object data).

Feature Description: Event Reporting.

Parameter and Description:

string eventId: The English name of the event set in the mp experimental system.

object data: An object that can be processed by JSON.stringify, which will be reported to the system concurrently.

©2013-2022 Tencent Cloud. All rights reserved.

canvas

Canvas Overview

Last updated：2024-07-12 19:25:42

createOffscreenCanvas

This API is used via OffscreenCanvas wx.createOffscreenCanvas(object object, number width, number height, Object this).

Feature Description: Creates an instance of an offscreen canvas.

Parameter and Description:

number width: The width of the canvas.

number height: The height of the canvas.

object this: this of the current component instance under Custom Components

Object object

Attribute	Type	Default value	Required	Description
type	string	webgl	No	Type of created offscreen canvas
width	number		No	Canvas Width
height	number		No	Canvas Height
compInst	Component		No	"This" of the current component instance in the context of custom components

Object.type

Legal value	Description
webgl	Webgl type context
2d	2d type context

createCanvasContext

©2013-2022 Tencent Cloud. All rights reserved.

This API is used via CanvasContext wx.createCanvasContext(string canvasId, Object this).

Feature Description: Creates a canvasContext object for canvas drawing context.
Parameter and Description:
string canvasId: The `canvas-id` attribute of the `<canvas>` component for which the context is to be obtained.
Object this: In the context of custom components, "this" signifies the search for the `<canvas>` possessing the canvas-id within this custom component. If omitted, the search will not be conducted within any custom components.
Return Value: canvasContext.

canvasToTempFilePath

This API is used via wx.canvasToTempFilePath(Object object, Object this).

Feature Description: Exports the content of a specified area on the current canvas, generating an image of a specified size. Calling this method within the `draw()` callback ensures successful image export.
Parameter and Description: Object.

Attribute	Type	Default value	Required	Description
x	number	0	No	Horizontal coordinate of the top left corner of the specified canvas area
y	number	0	No	Vertical coordinate of the top left corner of the specified canvas area
width	number	Canvas width - x	No	The width of the specified canvas area.
height	number	Canvas height - y	No	The height of the specified canvas area.
destWidth	number	Width multiplied by screen pixel density.	No	The width of the output image.
destHeight	number	Height multiplied by	No	The height of the output image.

©2013-2022 Tencent Cloud. All rights reserved.

		screen pixel density.		
canvasId	string	-	Yes	Canvas identifier, canvas-id passed into the `<canvas>` component.
fileType	string	png	No	Type of the target file.
quality	number	-	Yes	The quality of the image, currently only effective for jpg. The value range is (0, 1]. If not within the range, it is treated as 1.0.
success	function	-	No	Callback Function of Successful Interface Call
fail	function	-	No	Callback Function of Failing Interface Call
complete	function	-	No	Callback function executed upon the completion of the interface invocation (both successful and unsuccessful invocations)

Valid values of object.fileType

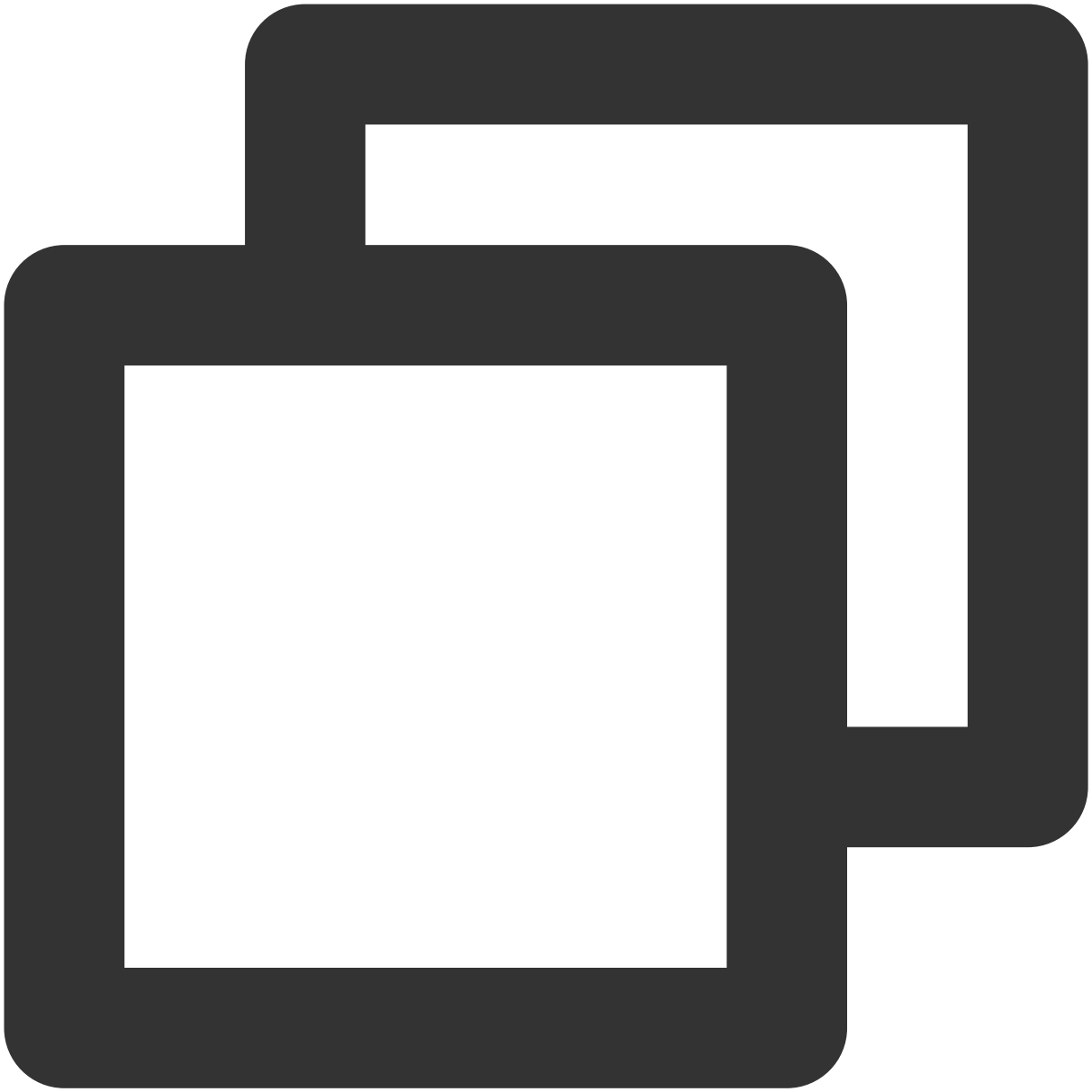
Value	Description
jpg	JPG image
png	PNG image

Result of res callback for object.success

Attribute	Type	Description
tempFilePath	string	Temporary path to the generated file (local path)

Sample Code

©2013-2022 Tencent Cloud. All rights reserved.



```
wx.canvasToTempFilePath({
  x: 100,
  y: 200,
  width: 50,
  height: 50,
  destWidth: 100,
  destHeight: 100,
  canvasId: 'myCanvas',
  success(res) {
    console.log(res.tempFilePath)
  }
```

©2013-2022 Tencent Cloud. All rights reserved.

```
})
```

canvasPutImageData

This API is used via wx.canvasPutImageData(Object object, Object this).

Feature Description: Draws pixel data onto the canvas. In the context of a custom component, the second parameter passes in the custom component instance "this", to manipulate the `canvas` component in the component.

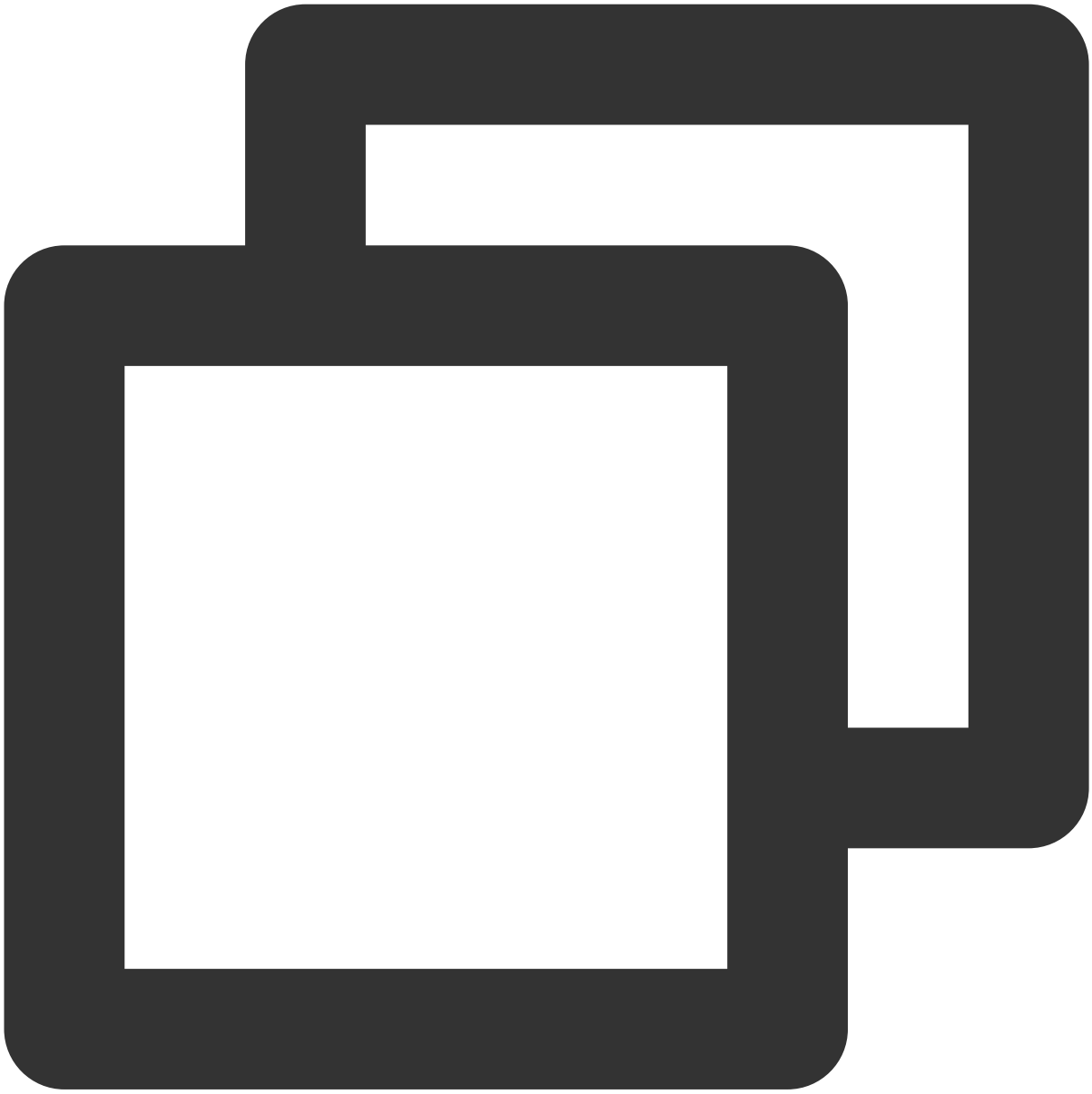
Parameter 1: Object.

Attribute	Type	Default value	Required	Description
canvasId	string	-	Yes	Canvas identifier, the attribute of the canvas-id passed into the `<canvas>` component.
data	Uint8ClampedArray	-	Yes	Image pixel data, a one-dimensional array, with every four items representing the RGBA of a pixel.
x	number	-	Yes	The position offset of the source image data in the target canvas (offset in the x-axis direction).
y	number	-	Yes	The position offset of the source image data in the target canvas (offset in the y-axis direction).
width	number	-	Yes	The width of the source image data's rectangular area.
height	number	-	Yes	The height of the source image data's rectangular area.
success	function	-	No	Callback Function of Successful Interface Call
fail	function	-	No	Callback Function of Failing Interface Call
complete	function	-	No	Callback function executed upon the completion of the interface invocation (both successful and unsuccessful invocations)

©2013-2022 Tencent Cloud. All rights reserved.

Parameter 2: Object this, under the custom component, the current component instance's this, to operate within the `<canvas>` component.

Sample Code



```
const data = new Uint8ClampedArray([255, 0, 0, 1])
wx.canvasPutImageData({
  canvasId: 'myCanvas',
  x: 0,
  y: 0,
  width: 1,
  data,
```

©2013-2022 Tencent Cloud. All rights reserved.

```
  success(res) {}
})
```

canvasGetImageData

This API is used via wx.canvasGetImageData(Object object, Object this).

Feature Description: Gets the implicit pixel data from the canvas area.

Parameter 1: Object.

Attribute	Type	Default value	Required	Description
canvasId	string	-	Yes	Canvas identifier, the attribute of the `canvas-id` passed into the component `<canvas>` .
x	number	-	Yes	The horizontal coordinate of the top-left corner of the rectangular area of the image data to be extracted.
y	number	-	Yes	The vertical coordinate of the top-left corner of the rectangular area of the image data to be extracted.
width	number	-	Yes	The width of the top-left corner of the rectangular area of the image data to be extracted.
height	number	-	Yes	The height of the top-left corner of the rectangular area of the image data to be extracted.
success	function	-	No	Callback Function of Successful Interface Call
fail	function	-	No	Callback Function of Failing Interface Call
complete	function	-	No	Callback function executed upon the completion of the interface invocation (both successful and unsuccessful invocations)

Parameters for object.success callback function: Object res.

Attribute	Type	Description
width	number	The width of the image data rectangle.
height	number	The height of the source image data rectangle.

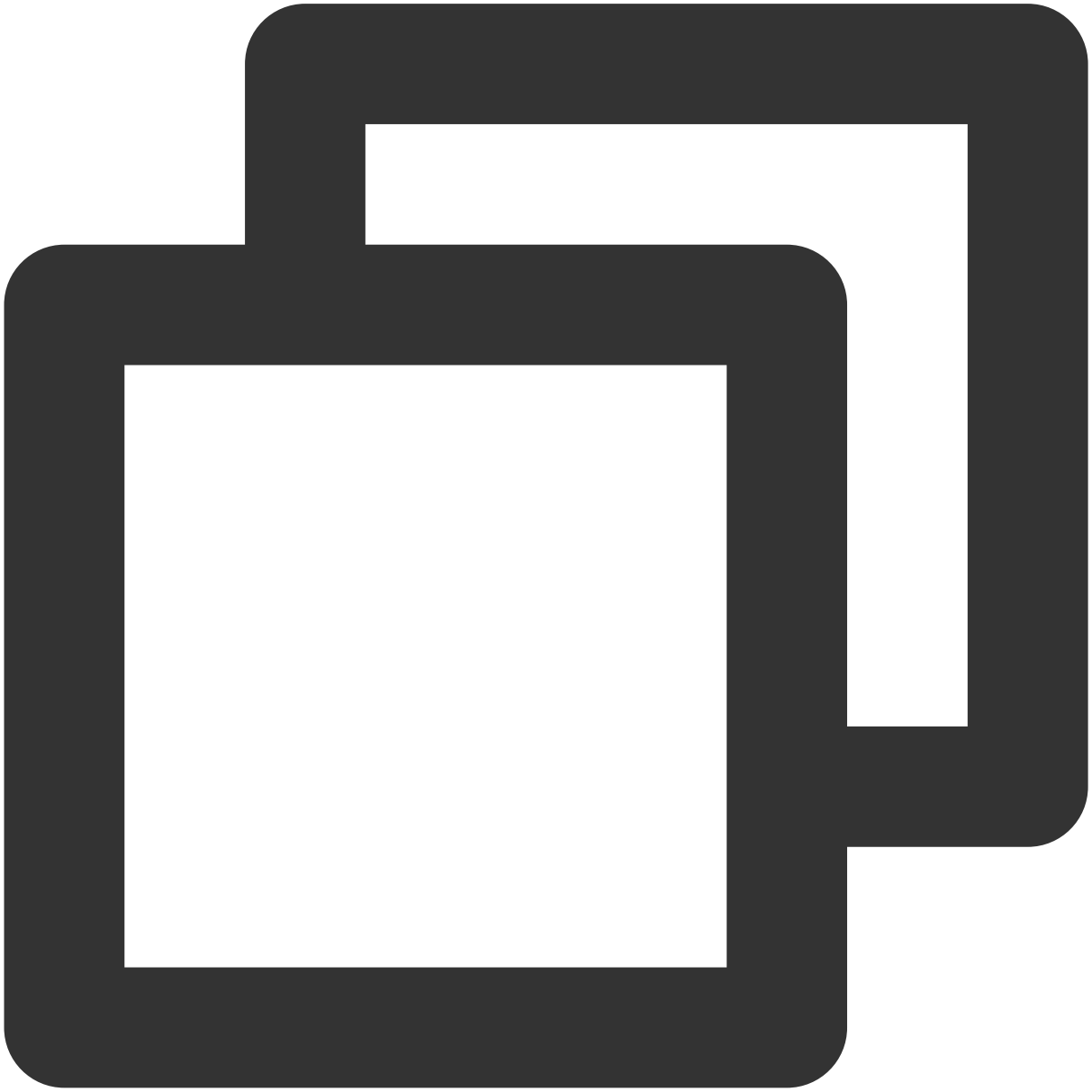
©2013-2022 Tencent Cloud. All rights reserved.

data	Uint8ClampedArray	Image pixel data, a one-dimensional array, with every four items representing the RGBA of a pixel.

Parameter 2：Object this, under Custom Components, this of the current component instance to manipulate <canvas> components within the component.

The parameter of the object.success callback function: Object this, "this" of the current component instance under custom components, used to operate the `<canvas>` component within.

Sample Code



©2013-2022 Tencent Cloud. All rights reserved.

```
wx.canvasGetImageData({
  canvasId: 'myCanvas',
  x: 0,
  y: 0,
  width: 100,
  height: 100,
  success(res) {
    console.log(res.width) // 100
    console.log(res.height) // 100
    console.log(res.data instanceof Uint8ClampedArray) // true
    console.log(res.data.length) // 100 * 100 * 4
  }
})
```

Image

Feature Description: Image object.

Parameter and Description

Attribute	Type	Description
src	string	The URL of the image
width	number	The actual width of the image
height	number	The actual height of the image
referrerPolicy	string	origin: Sends the complete referrer; no-referrer: Not to send. The format is fixed as https://servicewechat.com/{appid}/{version}/page-frame.html, where {appid} is the appid of the mini program, {version} is the version number of the mini program. A version number of 0 indicates a development version, trial version, or review version. A version number of devtools indicates a developer tool, and the rest are official versions.
onload	function	Callback function triggered upon the completion of image loading
onerror	function	Callback function triggered when an error occurs during image loading

ImageData

Feature Description: ImageData object.

Parameter and Description

©2013-2022 Tencent Cloud. All rights reserved.

Attribute	Type	Description
width	number	Describe the actual width of ImageData using pixels.
height	number	Describe the actual height of ImageData using pixels.
data	Uint8ClampedArray	A one-dimension array containing data in RGBA order, represented by integers ranging from 0 to 255 (inclusive).

canvasGradient

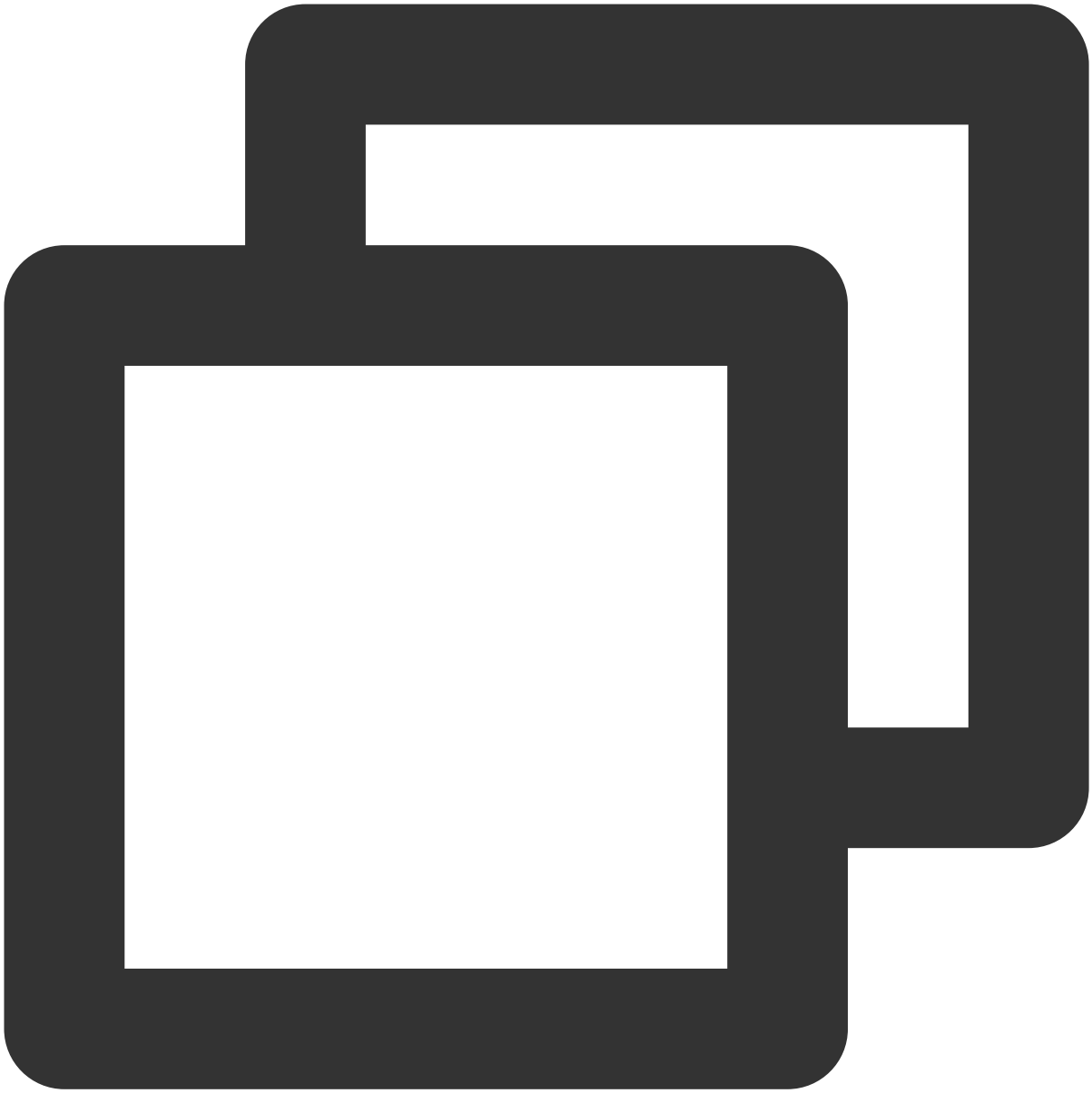
Feature Description: Gradient object.

.addColorStop

> This method is used via CanvasGradient.addColorStop(number stop, Color color).

Feature Description: Adds a gradient point of color. Parts less than the minimum stop value will be rendered according to the color of the minimum stop value, and parts greater than the maximum stop value will be rendered according to the color of the maximum stop value, equivalent to CanvasGradient.addColorStop(number stop, Color color).
Parameter 1: number stop, indicating the position between the start and end of the gradient, range from 0 to 1.
Parameter 2: Color color, the color of the gradient point.
Sample Code

©2013-2022 Tencent Cloud. All rights reserved.



```
const ctx = wx.createCanvasContext('myCanvas')

// Create circular gradient
const grd = ctx.createLinearGradient(30, 10, 120, 10)
grd.addColorStop(0, 'red')
grd.addColorStop(0.16, 'orange')
grd.addColorStop(0.33, 'yellow')
grd.addColorStop(0.5, 'green')
grd.addColorStop(0.66, 'cyan')
grd.addColorStop(0.83, 'blue')
grd.addColorStop(1, 'purple')
```

©2013-2022 Tencent Cloud. All rights reserved.

```
// Fill with gradient
ctx.setFillStyle(grd)
ctx.fillRect(10, 10, 150, 80)
ctx.draw()
```

RenderingContext

Feature Description: Canvas drawing context.
The CanvasRenderingContext2D object can be obtained through the Canvas.getContext('2d') interface, implementing the properties and methods defined by the HTML Canvas 2D Context.
The WebGLRenderingContext object can be obtained through the Canvas.getContext('webgl') or OffscreenCanvas.getContext('webgl') interface, implementing all properties, methods, and constants defined by WebGL 1.0.

CanvasContext

For more details, see canvasContext.

OffscreenCanvas

For more details, see OffscreenCanvas.

Path2D

For more details, see Path2D.

Canvas

For more details, see Canvas.

Color

For more details, see Color.

©2013-2022 Tencent Cloud. All rights reserved.

OffscreenCanvas

Last updated：2024-07-12 19:25:29

Function Description: Offscreen canvas instances can be created through wx.createOffscreenCanvas.

Attribute

Attributes	Types	Note
width	number	Canvas Width
height	number	Canvas Height

Method Set

createImage

The method is utilized as Image OffscreenCanvas.createImage()

Function Description: This creates an image object. It is compatible with both 2D Canvas and WebGL Canvas, however, it does not support the mixed use of 2D and WebGL methods.
Return Value: Image. Please note that the mixed use of image objects created by WebGL and 2D canvas is not permitted. When using, it is advisable to create image objects using the canvas's own createImage.

getContext

The method is utilized as RenderingContext OffscreenCanvas.getContext(string contextType)

Function Description: This method returns the drawing context of the OffscreenCanvas.
Parameters and Description: string contextType, the type of drawing context, which needs to be consistent with the type passed in when creating OffscreenCanvas.

Value	Note
webgl	WebGL type context
2d	2D type

©2013-2022 Tencent Cloud. All rights reserved.

©2013-2022 Tencent Cloud. All rights reserved.

Path2D

Last updated：2024-07-12 19:25:00

Function Description: The Path2D interface of the Canvas 2D API is utilized to declare paths, which will subsequently be employed by the CanvasRenderingContext2D object. The path methods of the CanvasRenderingContext2D interface are also present within the Path2D interface, permitting the creation of retainable and reusable paths in the Canvas as per necessity.

Attribute

Attributes	Types	Note
width	number	Describing the actual width of ImageData using pixels.
height	number	Describing the actual height of ImageData using pixels.
data	Uint8ClampedArray	A one-dimensional array containing data in RGBA order, represented by integers ranging from 0 to 255 (inclusive).

Method Set

addPath

The method is utilized as Path2D.addPath(Path2D path).

Function Description: Adds a path to the current path.
Parameters and Description: Path2D path, the Path2D path to be added.

arc

The method is employed as Path2D.arc(number x, number y, number radius, number startAngle, number endAngle, boolean counterclockwise).

Function Description: Incorporates an arc path segment.

©2013-2022 Tencent Cloud. All rights reserved.

Parameters and Descriptions:

number x: The x-coordinate of the circle's center.

number y: The y-coordinate of the circle's center.

number radius: The radius of the circle, which must be a positive number.

number startAngle: The starting angle of the arc.

number endAngle: The terminating angle of the arc.

boolean counterclockwise: Determines if the drawing is in a counterclockwise direction. If set to true, the drawing will commence from the endAngle and proceed to the startAngle.

arcTo

> The method is utilized as follows: Path2D.arcTo(number x1, number y1, number x2, number y2, number radius)

Function Description: Adds an arc path by specifying control points.

Parameters and Descriptions:

number x1: The x-coordinate of the first control point.

number y1: The y-coordinate of the first control point.

number x2: The x-coordinate of the second control point.

number y2: The y-coordinate of the second control point.

number radius: The radius of the circle, which must be a non-negative number.

bezierCurveTo

> The method is utilized as follows: Path2D.bezierCurveTo(number cp1x, number cp1y, number cp2x, number cp2y, number x, number y)

Function Description: Adds a cubic Bezier curve path.

Parameters and Descriptions:

number cp1x: The x-coordinate of the first control point.

number cp1y: The y-coordinate of the first control point.

number cp2x: The x-coordinate of the second control point.

number cp2y: The y-coordinate of the second control point.

number x: The x-coordinate of the endpoint.

number y: The y-coordinate of the endpoint.

©2013-2022 Tencent Cloud. All rights reserved.

closePath

The method is utilized as Path2D.closePath().

Function Description: Closes the path to the starting point.

ellipse

The method is utilized as Path2D.ellipse(number x, number y, number radiusX, number radiusY, number rotation, number startAngle, number endAngle, boolean counterclockwise).

Function Description: Adds an elliptical arc path.
Parameters and Descriptions:
number x: The x-coordinate of the ellipse's center.
number y: The y-coordinate of the ellipse's center.
number radiusX: The semi-major axis of the ellipse, must be a non-negative number.
number radiusY: The semi-minor axis of the ellipse, must be a non-negative number.
number rotation: The rotation angle of the ellipse.
number startAngle: The starting angle of the arc.
number endAngle: The terminating angle of the arc.
boolean counterclockwise: Determines if the drawing is in a counterclockwise direction. If set to true, the drawing will commence from the endAngle and proceed to the startAngle.

lineTo

The method is utilized as follows: Path2D.lineTo(number x, number y)

Function Description: Addition of a linear path.
Parameters and Descriptions:
number x: The x-coordinate of the endpoint.
number y: The y-coordinate of the endpoint.

moveTo

©2013-2022 Tencent Cloud. All rights reserved.

The method is utilized as follows: Path2D.moveTo(number x, number y)

Function Description: Movement of the path's starting point.

Parameters and Descriptions:

number x: The horizontal coordinate.

number y: The vertical coordinate.

quadraticCurveTo

The method is employed as follows: Path2D.quadraticCurveTo(number cpx, number cpy, number x, number y)

Function Description: Addition of a quadratic Bézier curve path.

Parameters and Descriptions:

number cpx: The horizontal coordinate of the control point.

number cpy: The vertical coordinate of the control point.

number x: The x-coordinate of the endpoint.

number y: The y-coordinate of the endpoint.

rect

The method is utilized as follows: Path2D.rect(number x, number y, number width, number height)

Function Description: Addition of a rectangular path.

Parameters and Descriptions:

number x: The horizontal coordinate of the starting point.

number y: The vertical coordinate of the starting point.

number width: The width of the rectangle, positive values extend to the right, negative values to the left.

number height: The height of the rectangle, positive values extend downwards, negative values upwards.

©2013-2022 Tencent Cloud. All rights reserved.

Canvas

Last updated：2024-07-12 19:24:24

Attributes

Canvas Width: The breadth of the canvas.
Canvas Height: The vertical extent of the canvas.

Method Set

cancelAnimationFrame

The method is utilized as follows: Canvas.cancelAnimationFrame(number requestID)

Function Description: Cancels the animation frame request scheduled by requestAnimationFrame. It is compatible with both 2D Canvas and WebGL Canvas, however, it does not support the concurrent use of 2D and WebGL methods.
Parameters and Description: Number requestID.

createImageData

The method is utilized as follows: Image Canvas.createImage()

Function Description: Creates an ImageData object. Only supported in 2D Canvas.
Parameters and Description: Image.

createImage

The method is utilized as follows: ImageData Canvas.createImageData()

Function Description: Creates an image object. Supports use under 2D Canvas and WebGL Canvas, but does not support mixing 2D and WebGL methods.

©2013-2022 Tencent Cloud. All rights reserved.

Parameters and Description: ImageData.

createPathe2D

The method is utilized as follows: Path2D Canvas.createPath2D(Path2D path)

Function Description: Creates a Path2D object.
Parameters and Description: Path2D path.
Return Value: Path2D.

getContext

The method is utilized as follows: RenderingContext Canvas.getContext(string contextType)

Function Description: This method returns the drawing context of the Canvas.
Parameters and Description: string contextType, the context type.

Value	Note
2d	2D drawing context
webgl	WebGL drawing context
webgl2	WebGL2 drawing context

Return Value: RenderingContext, facilitating the acquisition of 2D and WebGL drawing contexts.

requestAnimationFrame

The method is utilized as follows: number Canvas.requestAnimationFrame(function callback)

Function Description: Executes upon the next redraw. Compatible with both 2D Canvas and WebGL Canvas, however, the methods of 2D and WebGL cannot be used interchangeably.
Parameters and Description: function callback, the callback to be executed.
Return Value: number, the request ID.

toDataURL

©2013-2022 Tencent Cloud. All rights reserved.

The method is employed as follows: string Canvas.toDataURL(string type, number encoderOptions)

Function Description: Returns a data URI that displays the image. The type parameter can be used to specify its format, with PNG as the default.

Parameters and Descriptions:

string type: The image format, default is image/png.

number encoderOptions: When the image format is specified as image/jpeg or image/webp, the image quality can be selected within the range of 0 to 1. If it exceeds the value range, the default value of 0.92 will be used. Other parameters will be disregarded.

Return Value: string.

©2013-2022 Tencent Cloud. All rights reserved.

canvasContext

Last updated：2024-07-12 19:23:41

Function Description: The drawing context for the canvas component. canvasContext is the old interface, the new Canvas 2D interface is Web-compliant.

Property Description

Attributes	Types	Note
fillStyle	String	Fill color. Usage is the same as CanvasContext.setFillStyle()
strokeStyle	String	Border color. The usage aligns with CanvasContext.setFillStyle().
font	String	-
globalCompositeOperation	String	The type of compositing operation applied when drawing a new shape. The current Android version only applies compositing to fill fill blocks, compositing effects used for stroke line segments are source-over. The currently supported operations are: Android：xor, source-over, source-atop, destination-out, lighter, overlay, darken, lighten, hard-light iOS：xor, source-over, source-atop, destination-over, destination-out, lighter, multiply, overlay, darken, lighten, color-dodge, color-burn, hard-light, soft-light, difference, exclusion, saturation, luminosity
shadowOffsetX	Number	The horizontal offset of the shadow relative to the shape.
shadowOffsetY	Number	The vertical offset of the shadow relative to the shape.
shadowColor	Number	The color of the shadow.
shadowBlur	Number	The blur level of the shadow.
lineWidth	Number	The width of the line. The usage is the same as CanvasContext.setLineWidth().
lineCap	Number	The style of the line's endpoints. The usage is the same as CanvasContext.setLineCap().
lineJoin	Number	The style of the line's intersection. The usage is the same as

©2013-2022 Tencent Cloud. All rights reserved.

		CanvasContext.setLineJoin().
miterLimit	Number	The maximum miter length. The usage is the same as CanvasContext.setMiterLimit().
lineDashOffset	Number	The offset of the dashed line, with an initial value of zero.
globalAlpha	Number	The global brush opacity. The range is 0-1, where 0 indicates complete transparency and 1 indicates complete opacity.

Legal values for lineJoin

Value	Description
bevel	bevel
round	round
miter	miter

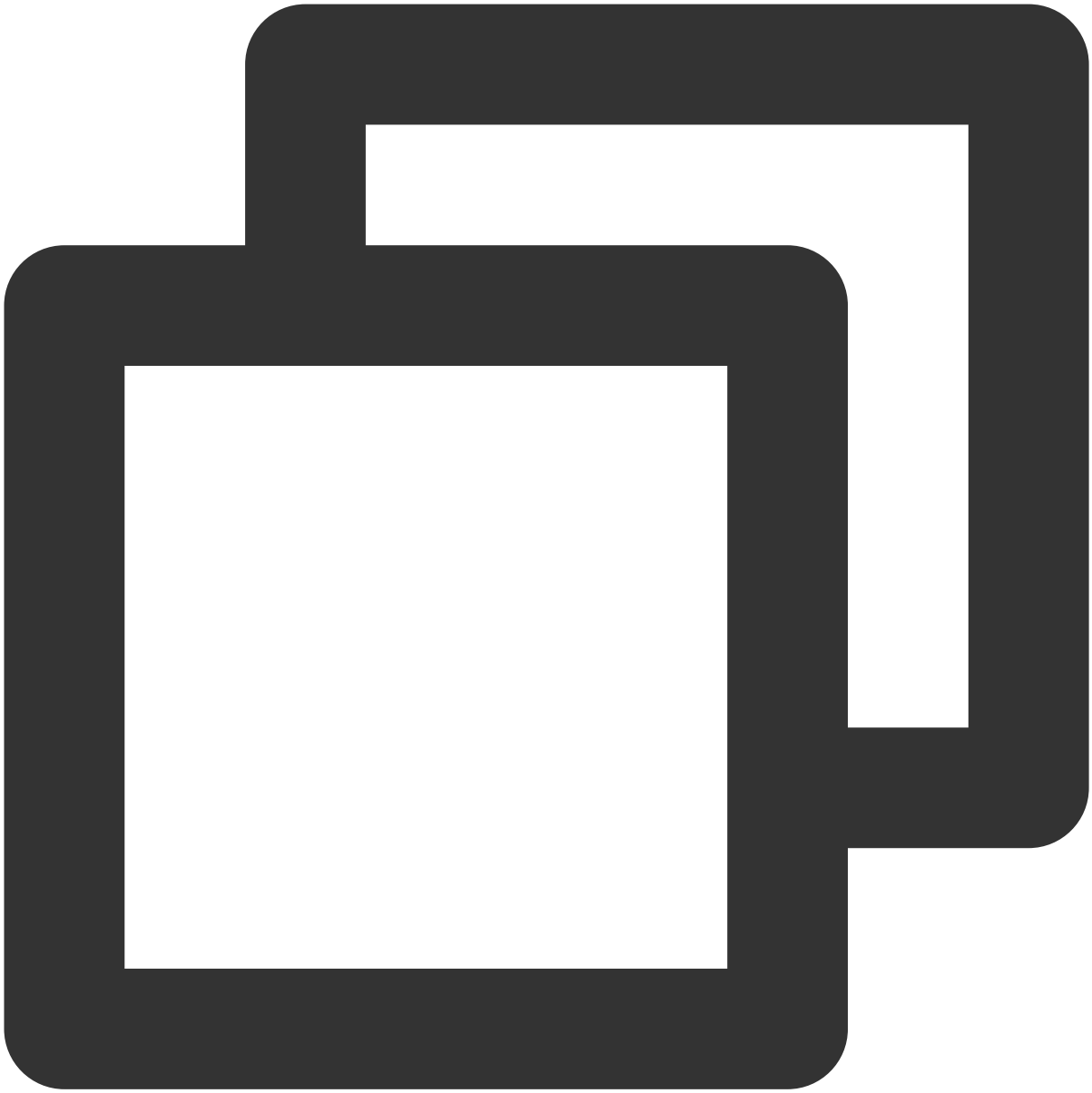
Method Set

arc

The method is used as follows: CanvasContext.arc(number x, number y, number r, number sAngle, number eAngle, boolean counterclockwise).

Function Description: Creates an arc. The creation of a circle can specify the starting radian as 0 and the ending radian as 2 * Math.PI. Use the stroke or fill method to draw an arc in the canvas.
Parameters and Descriptions:
number x, the x-coordinate of the circle's center;
number y, the y-coordinate of the circle's center;
number r, the radius of the circle;
number sAngle, the starting radian, in radians (in the 3 o'clock direction);
number eAngle, the ending radian;
number counterclockwise, whether the direction of the radian is counterclockwise.
Sample Code:

©2013-2022 Tencent Cloud. All rights reserved.



```
const ctx = wx.createCanvasContext('myCanvas')

// Draw coordinates
ctx.arc(100, 75, 50, 0, 2 * Math.PI)
ctx.setFillStyle('#EEEEEE')
ctx.fill()

ctx.beginPath()
ctx.moveTo(40, 75)
ctx.lineTo(160, 75)
ctx.moveTo(100, 15)
```

©2013-2022 Tencent Cloud. All rights reserved.

```
ctx.lineTo(100, 135)
ctx.setStrokeStyle('#AAAAAA')
ctx.stroke()

ctx.setFontSize(12)
ctx.setFillStyle('black')
ctx.fillText('0', 165, 78)
ctx.fillText('0.5*PI', 83, 145)
ctx.fillText('1*PI', 15, 78)
ctx.fillText('1.5*PI', 83, 10)

// Draw points
ctx.beginPath()
ctx.arc(100, 75, 2, 0, 2 * Math.PI)
ctx.setFillStyle('lightgreen')
ctx.fill()

ctx.beginPath()
ctx.arc(100, 25, 2, 0, 2 * Math.PI)
ctx.setFillStyle('blue')
ctx.fill()

ctx.beginPath()
ctx.arc(150, 75, 2, 0, 2 * Math.PI)
ctx.setFillStyle('red')
ctx.fill()

// Draw arc
ctx.beginPath()
ctx.arc(100, 75, 50, 0, 1.5 * Math.PI)
ctx.setStrokeStyle('#333333')
ctx.stroke()

ctx.draw()
```

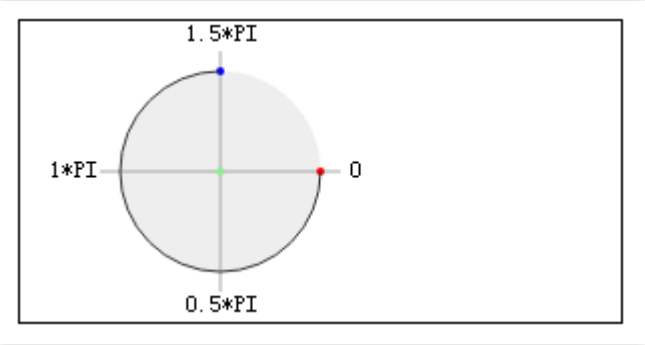
The three key coordinates for arc(100, 75, 50, 0, 1.5 * Math.PI) are as follows:

Green: Circle center (100, 75)

Red: Starting radian (0)

Blue: Ending radian (1.5 * Math.PI)

©2013-2022 Tencent Cloud. All rights reserved.



arcTo

The method is used as follows: CanvasContext.arcTo(number x1, number y1, number x2, number y2, number radius)

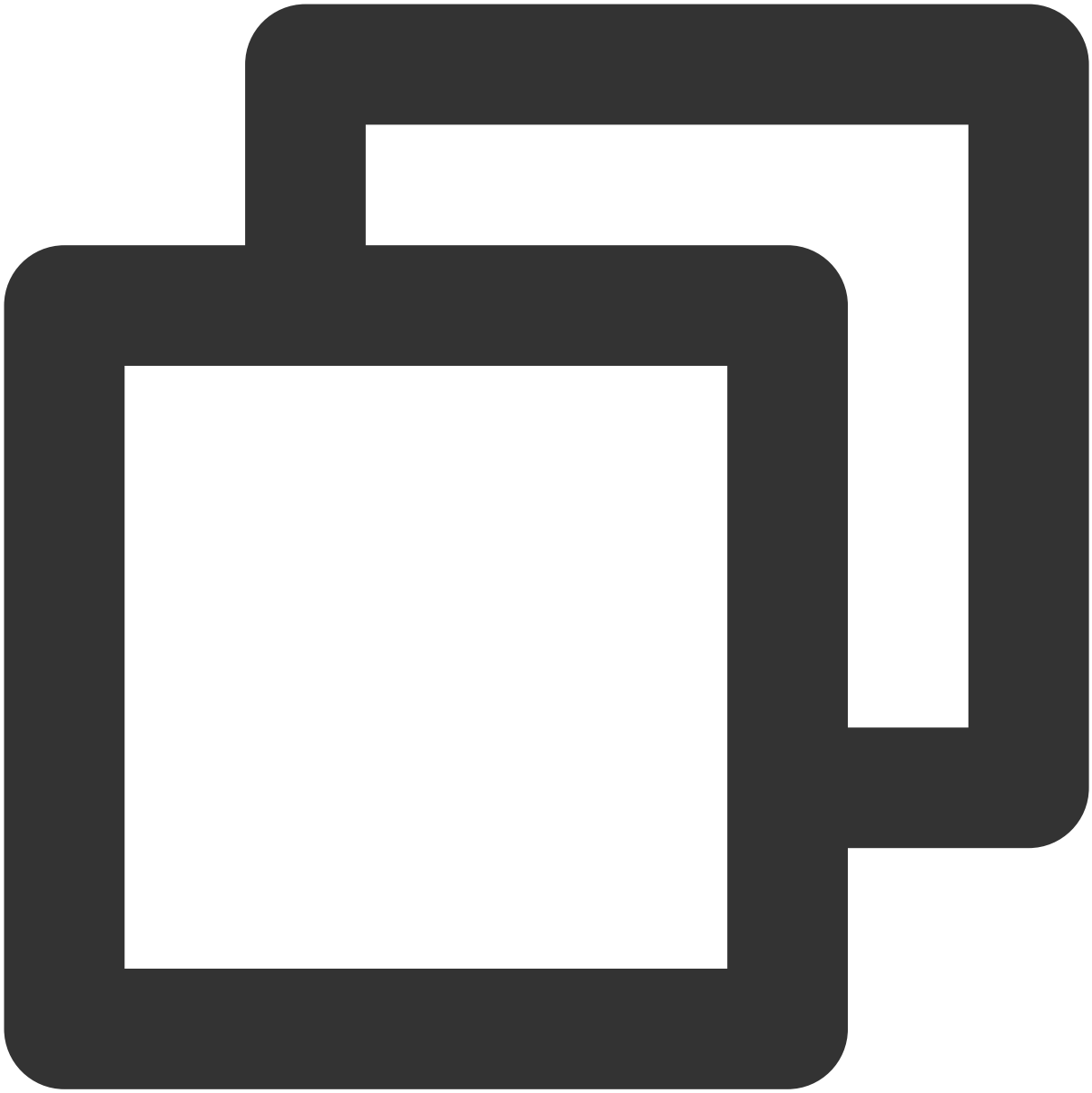
Function Description: Draws an arc path based on the control point and radius.
Parameters and Descriptions:
number x1, the x-axis coordinate of the first control point;
number y1, the y-axis coordinate of the first control point;
number x2, the x-axis coordinate of the second control point;
number y2, the y-axis coordinate of the second control point;
number radius, the radius of the arc.

draw

The method is used as follows: CanvasContext.draw(boolean reserve, function callback)

Function Description: Renders the previously described elements (paths, transformations, styles) in the drawing context onto the canvas.
Parameters and Descriptions:
boolean reserve: Determines whether the current drawing continues from the previous one. If the reserve parameter is set to false, the native layer will clear the canvas before proceeding with the drawing; if the reserve parameter is set to true, the current content on the canvas will be preserved, and the content drawn by the drawCanvas call will be overlaid on top, default is false.
function callback: The callback function executed upon completion of the drawing.
Sample Code: In the second draw() call, reserve is set to true. Therefore, the result of the previous drawing is preserved, and the fillStyle set in the context to 'red' has defaulted back to 'black'.

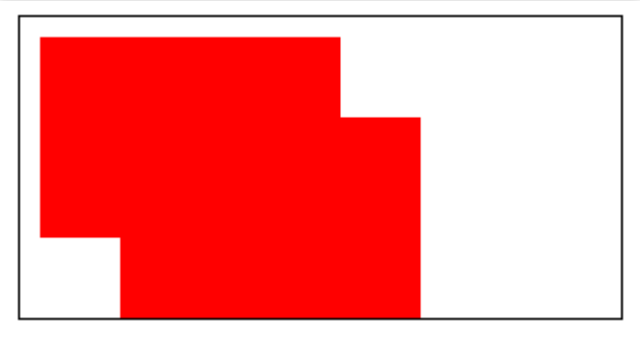
©2013-2022 Tencent Cloud. All rights reserved.



```
const ctx = wx.createCanvasContext('myCanvas')

ctx.setFillStyle('red')
ctx.fillRect(10, 10, 150, 100)
ctx.draw()
ctx.fillRect(50, 50, 150, 100)
ctx.draw(true)
```

©2013-2022 Tencent Cloud. All rights reserved.



drawImage

The method is utilized as follows: CanvasContext.drawImage(string imageResource, number sx, number sy, number sWidth, number sHeight, number dx, number dy, number dWidth, number dHeight)

Function Description: Draws an image onto the canvas.

Parameters and Descriptions:

string imageResource, the image resource to be drawn;

number sx, the x-coordinate of the top-left corner of the source image's rectangular selection box;

number sy, the y-coordinate of the top-left corner of the source image's rectangular selection box;

number sWidth, the width of the source image's rectangular selection box;

number sHeight, the height of the source image's rectangular selection box;

number dx, the x-axis position of the image's top-left corner on the target canvas;

number dy, the y-axis position of the image's top-left corner on the target canvas;

number dWidth, the width of the image drawn on the target canvas, allowing for scaling of the drawn image;

number dHeight, the height of the image drawn on the target canvas, allowing for scaling of the drawn image.

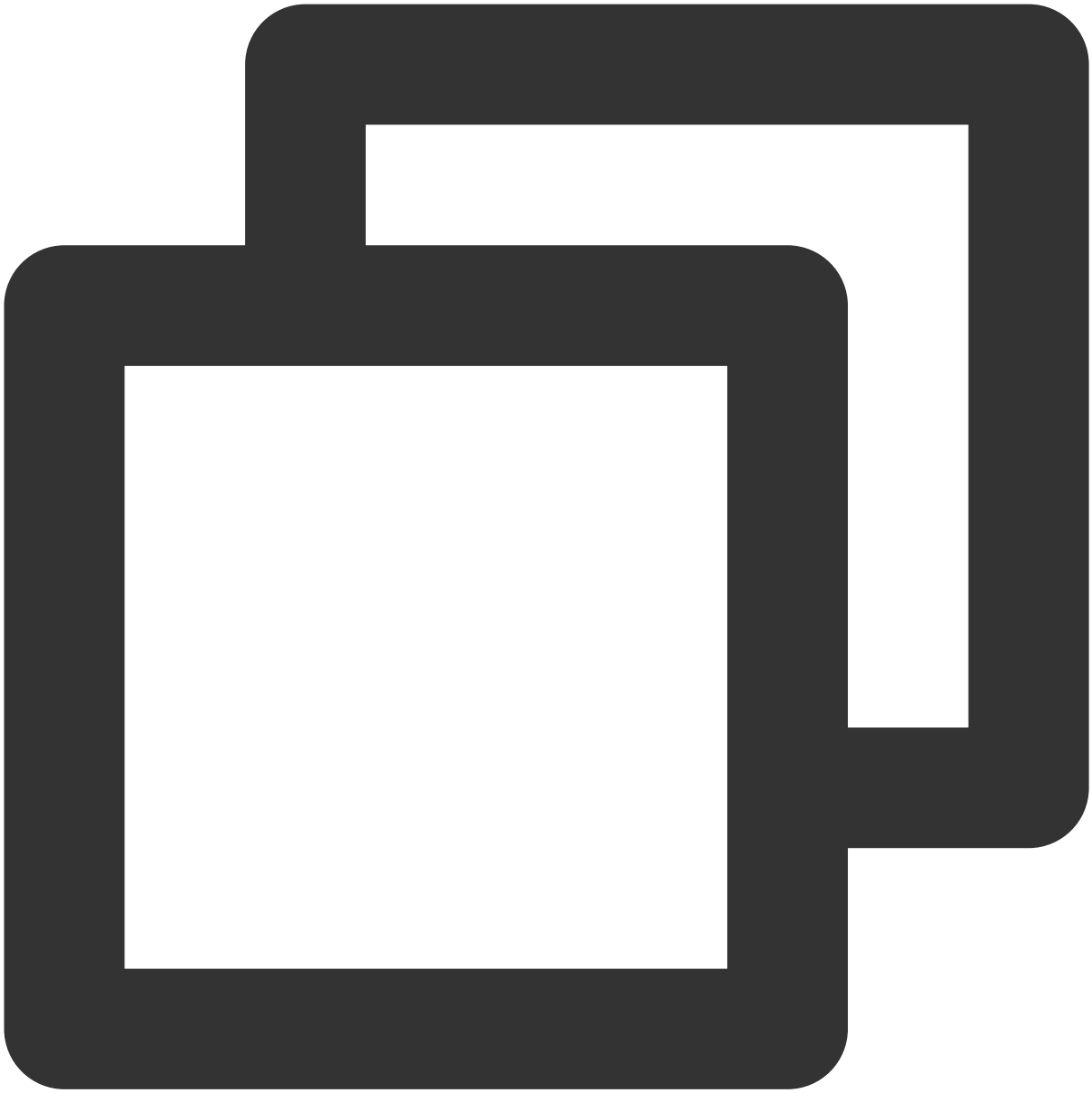
Sample Code: There are three versions of the syntax as follows.

drawImage(imageResource, dx, dy)

drawImage(imageResource, dx, dy, dWidth, dHeight)

drawImage(imageResource, sx, sy, sWidth, sHeight, dx, dy, dWidth, dHeight)

©2013-2022 Tencent Cloud. All rights reserved.



```
const ctx = wx.createCanvasContext('myCanvas')

wx.chooseImage({
  success(res) {
    ctx.drawImage(res.tempFilePaths[0], 0, 0, 150, 100)
    ctx.draw()
  }
})
```

©2013-2022 Tencent Cloud. All rights reserved.



createLinearGradient

The method is utilized as follows: CanvasGradient CanvasContext.createLinearGradient(number x0, number y0, number x1, number y1)

Function Description: Creates a linear gradient color. The returned CanvasGradient object requires the use of CanvasGradient.addColorStop() to specify gradient points, with at least two required.

Parameters and Descriptions:

number x0, the x-coordinate of the starting point;

number y0, the y-coordinate of the starting point;

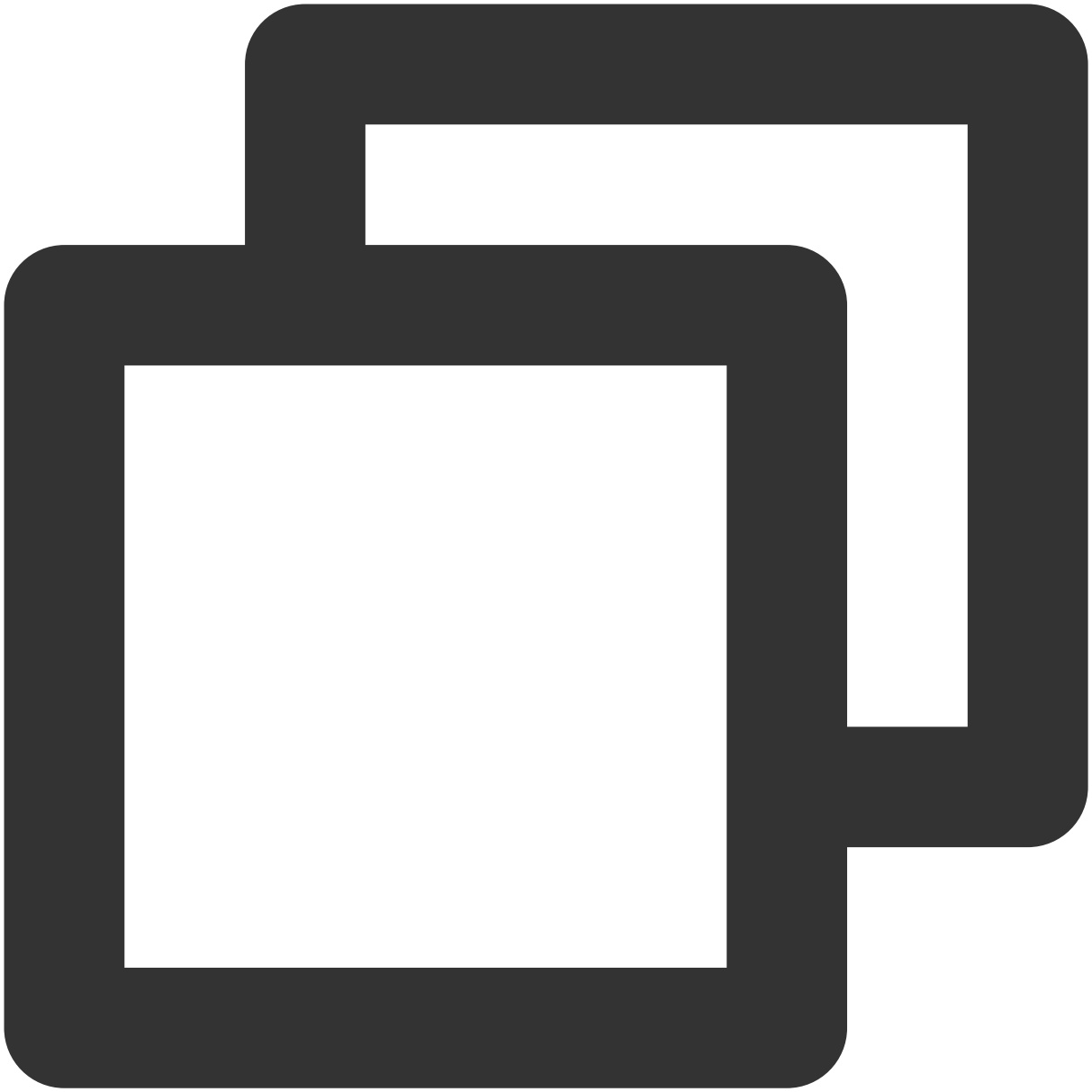
number x1, the x-coordinate of the endpoint;

number y1, the y-coordinate of the endpoint.

Return Value: CanvasGradient.

Sample Code:

©2013-2022 Tencent Cloud. All rights reserved.

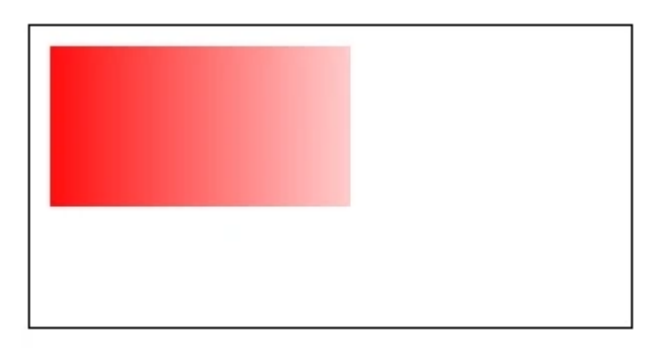


```
const ctx = wx.createCanvasContext('myCanvas')

// Create linear gradient
const grd = ctx.createLinearGradient(0, 0, 200, 0)
grd.addColorStop(0, 'red')
grd.addColorStop(1, 'white')

// Fill with gradient
ctx.setFillStyle(grd)
ctx.fillRect(10, 10, 150, 80)
ctx.draw()
```

©2013-2022 Tencent Cloud. All rights reserved.



createCircularGradient

The method is utilized as follows: CanvasGradient CanvasContext.createCircularGradient(number x, number y, number r)

Function Description: Creates a circular gradient color. The starting point is at the center of the circle, and the endpoint is on the circumference. The returned CanvasGradient object requires the use of CanvasGradient.addColorStop() to specify gradient points, with at least two required.

Parameters and Descriptions:

number x, the x-coordinate of the circle's center;

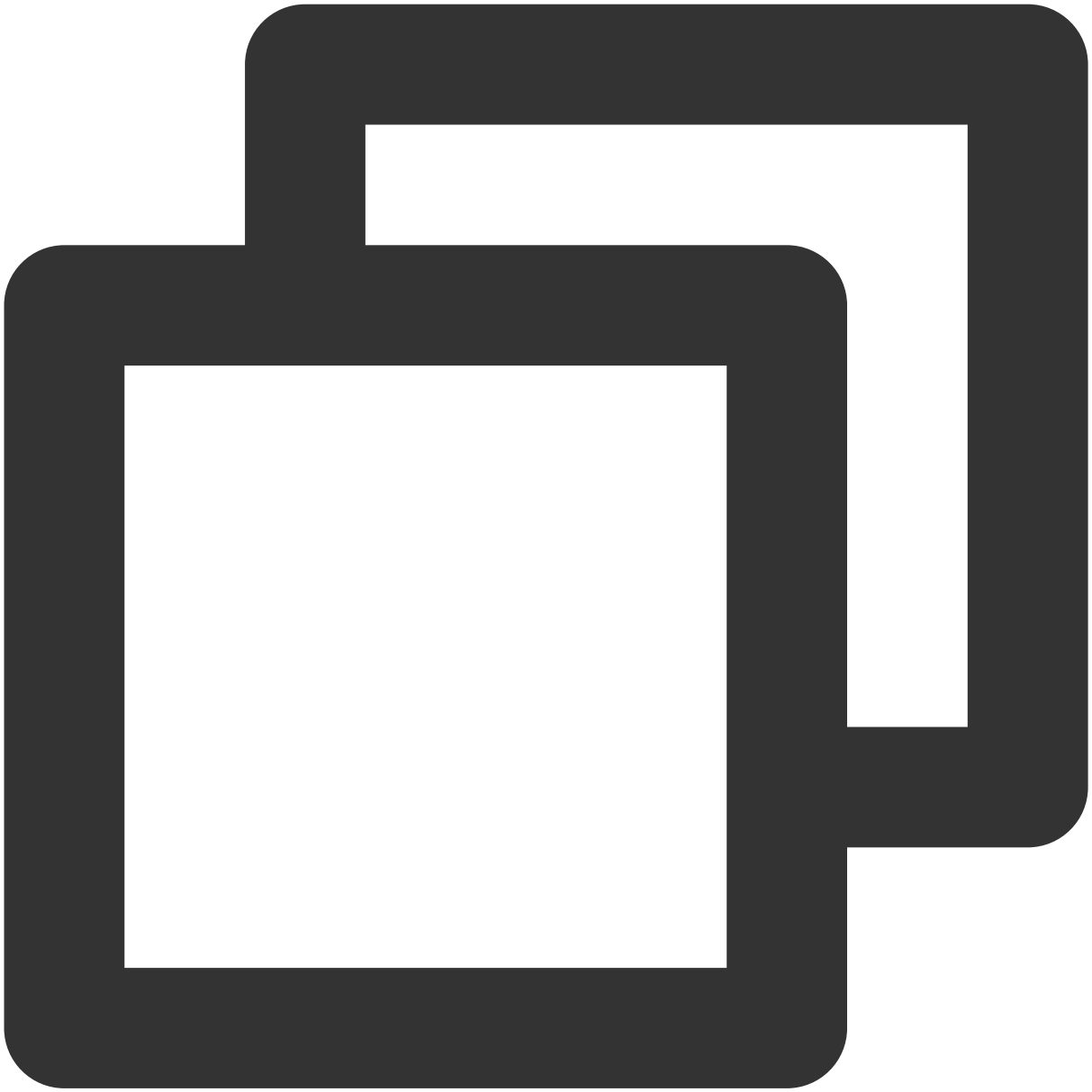
number y, the y-coordinate of the circle's center;

number r, the radius of the circle.

Return Value: CanvasGradient.

Sample Code:

©2013-2022 Tencent Cloud. All rights reserved.

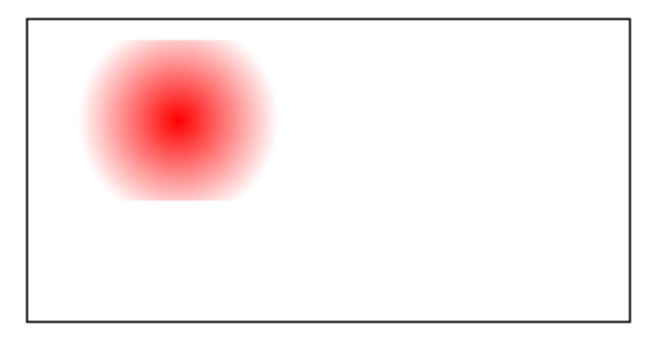


```
const ctx = wx.createCanvasContext('myCanvas')

// Create circular gradient
const grd = ctx.createCircularGradient(75, 50, 50)
grd.addColorStop(0, 'red')
grd.addColorStop(1, 'white')

// Fill with gradient
ctx.setFillStyle(grd)
ctx.fillRect(10, 10, 150, 80)
ctx.draw()
```

©2013-2022 Tencent Cloud. All rights reserved.



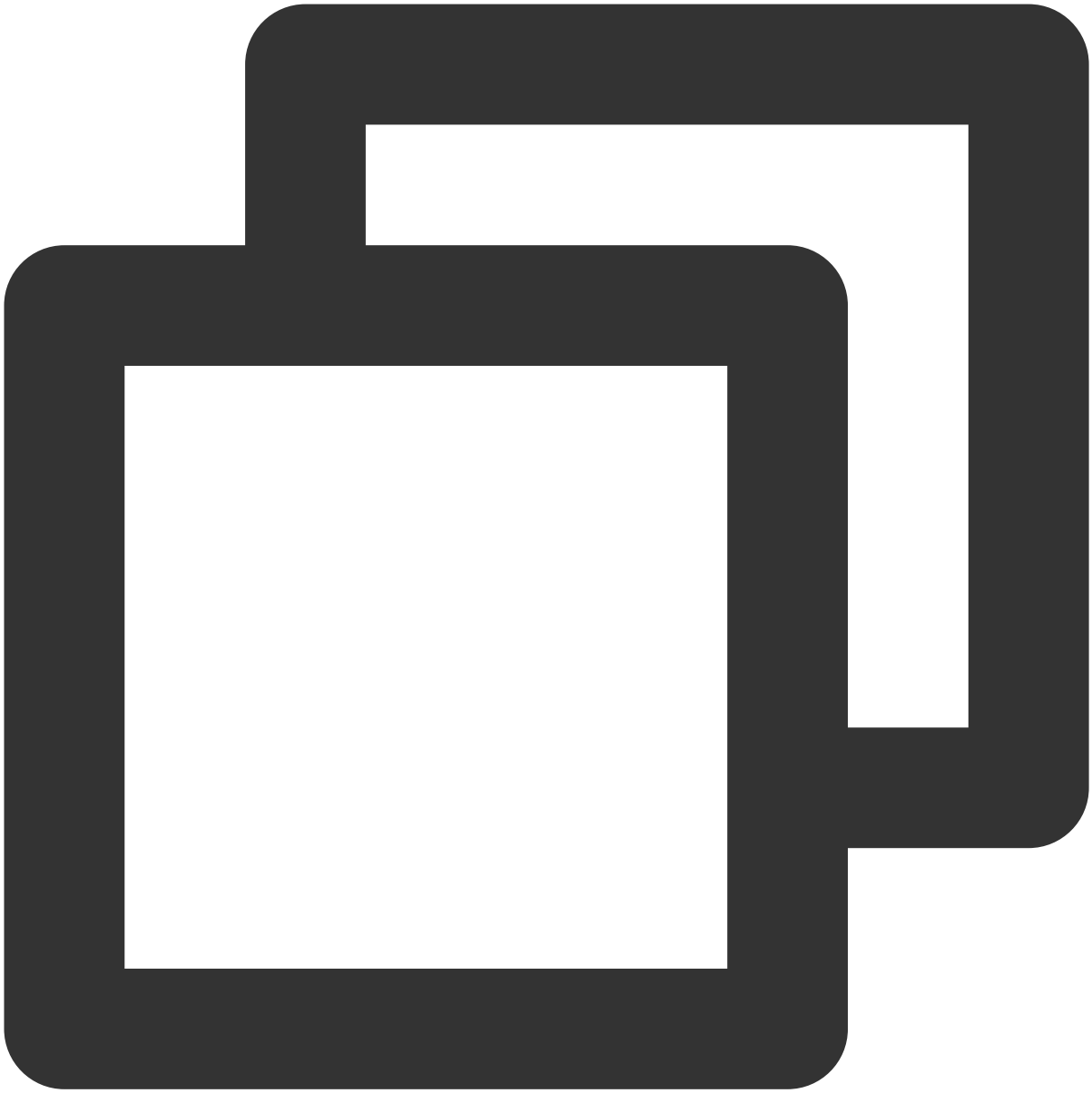
createPattern

Function Description: A method for creating a pattern for a specified image, which can repeat the original image in a specified direction. **The usage is as follows** CanvasContext.createPattern(string image, string repetition).
Parameters and Descriptions: string image, the source of the repeating image, supports code package paths and local temporary paths (local paths).

Value	Note
repeat	Repeats in both horizontal and vertical directions.
repeat-x	Repeats in the horizontal direction.
repeat-y	Repeats in the vertical direction.
no-repeat	Not repeating

Sample Code:

©2013-2022 Tencent Cloud. All rights reserved.



```
const ctx = wx.createCanvasContext('myCanvas')
constpattern=ctx.createPattern('/path/to/image','repeat-x')
ctx.fillStyle=pattern
ctx.fillRect(0,0,300,150)
ctx.draw()
```

measureText

The usage of this method is Object CanvasContext.measureText(string text).

©2013-2022 Tencent Cloud. All rights reserved.

Function Description: Measures the dimensions of the text. Currently, it only returns the width of the text, a synchronous interface.

Parameters and Descriptions:

string image: The source of the repeating image, supporting both the code package path and the local temporary path (local path).

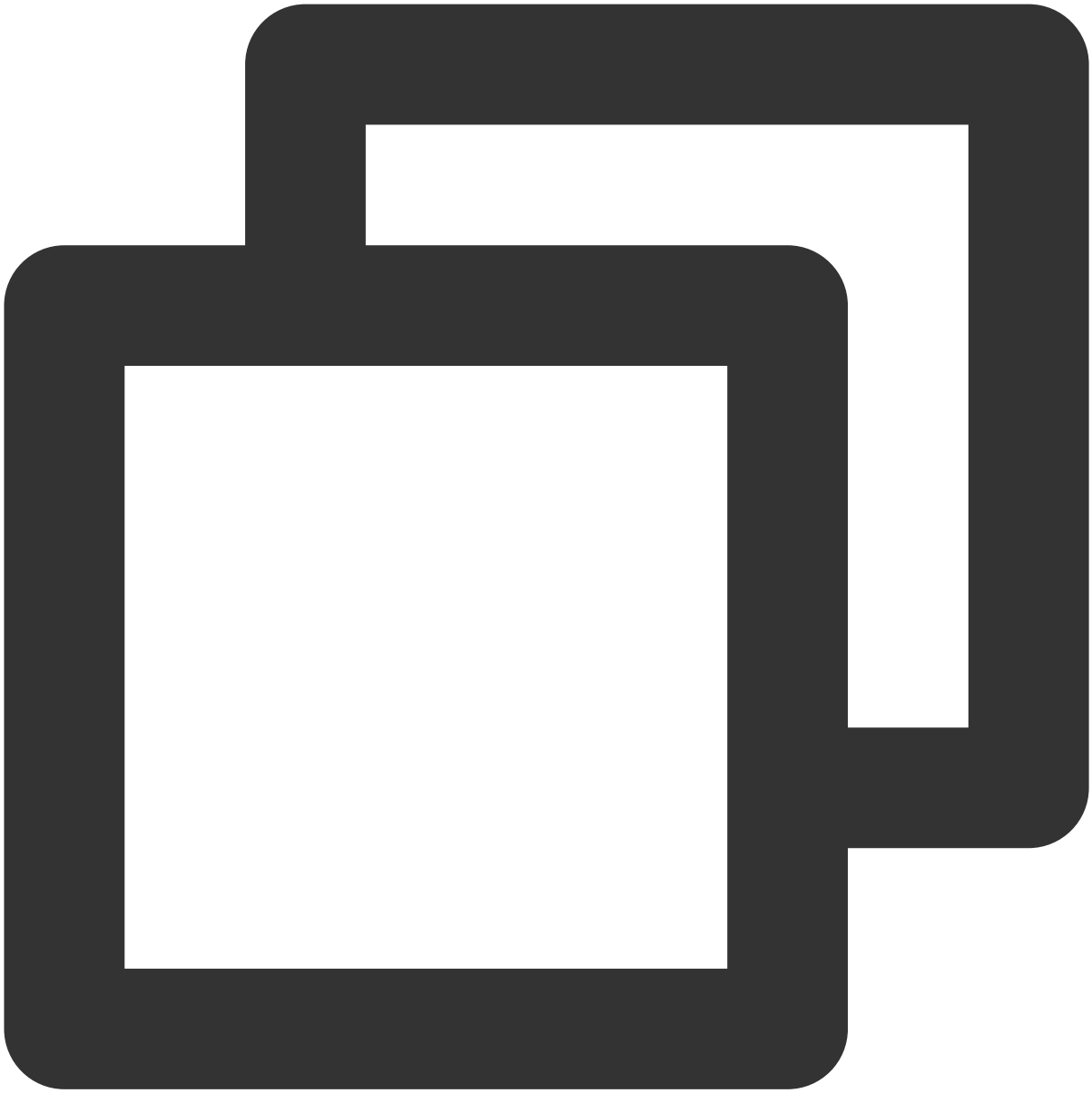
string repetition: How to repeat the image.

Legitimate values of repetition

Attributes	Types	Note
width	number	The width of the text.

Sample Code:

©2013-2022 Tencent Cloud. All rights reserved.



```
const ctx = wx.createCanvasContext('myCanvas')
ctx.font = 'italic bold 20px cursive'
const metrics = ctx.measureText('Hello World')
console.log(metrics.width)
```

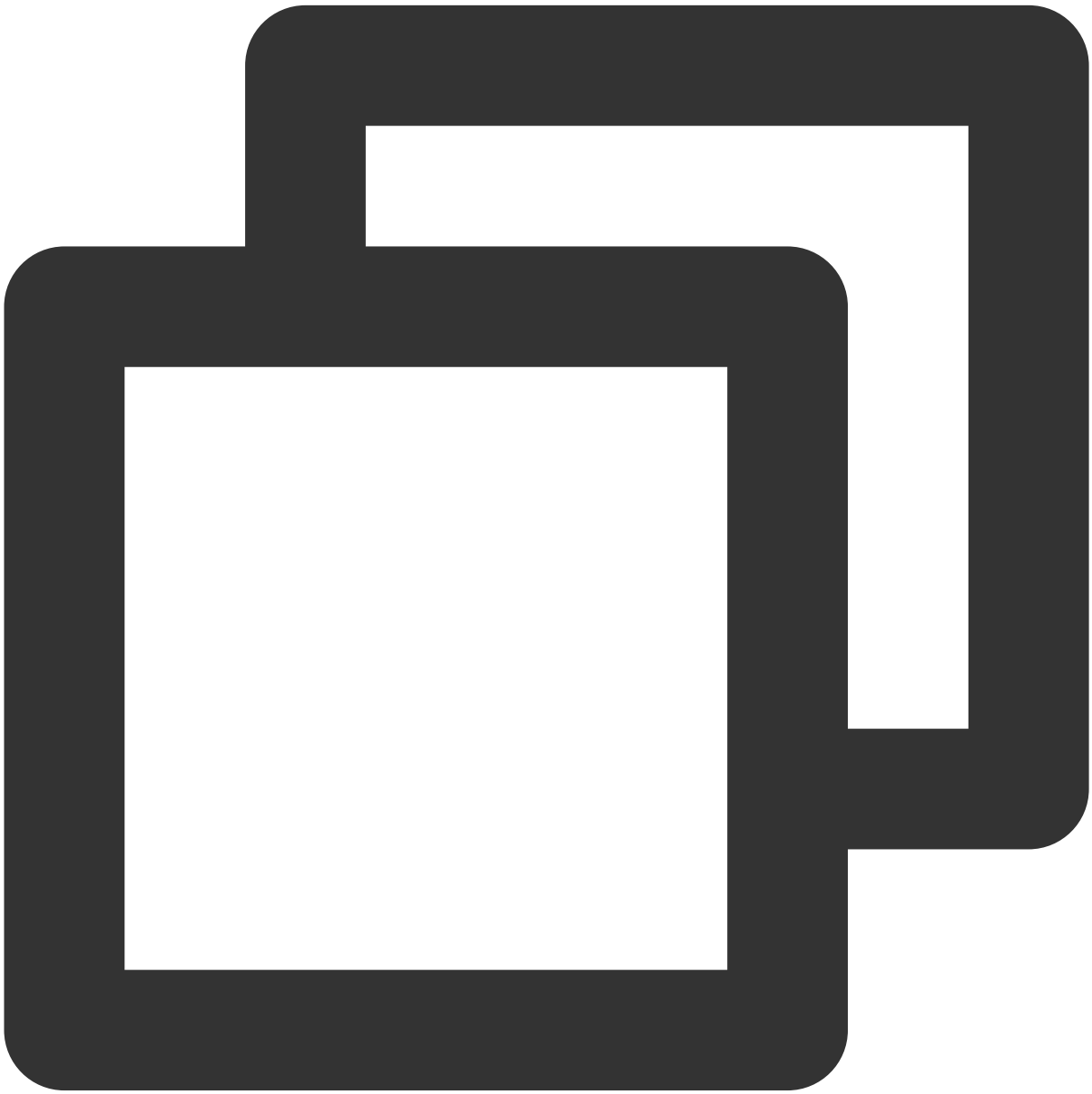
save

The method is utilized as CanvasContext.save()

©2013-2022 Tencent Cloud. All rights reserved.

Function Description: Saves the drawing context.

Sample Code:



```
const ctx = wx.createCanvasContext('myCanvas')

// save the default fill style
ctx.save()
ctx.setFillStyle('red')
ctx.fillRect(10, 10, 150, 100)
```

©2013-2022 Tencent Cloud. All rights reserved.

```
// restore to the previous saved state
ctx.restore()
ctx.fillRect(50, 50, 150, 100)

ctx.draw()
```



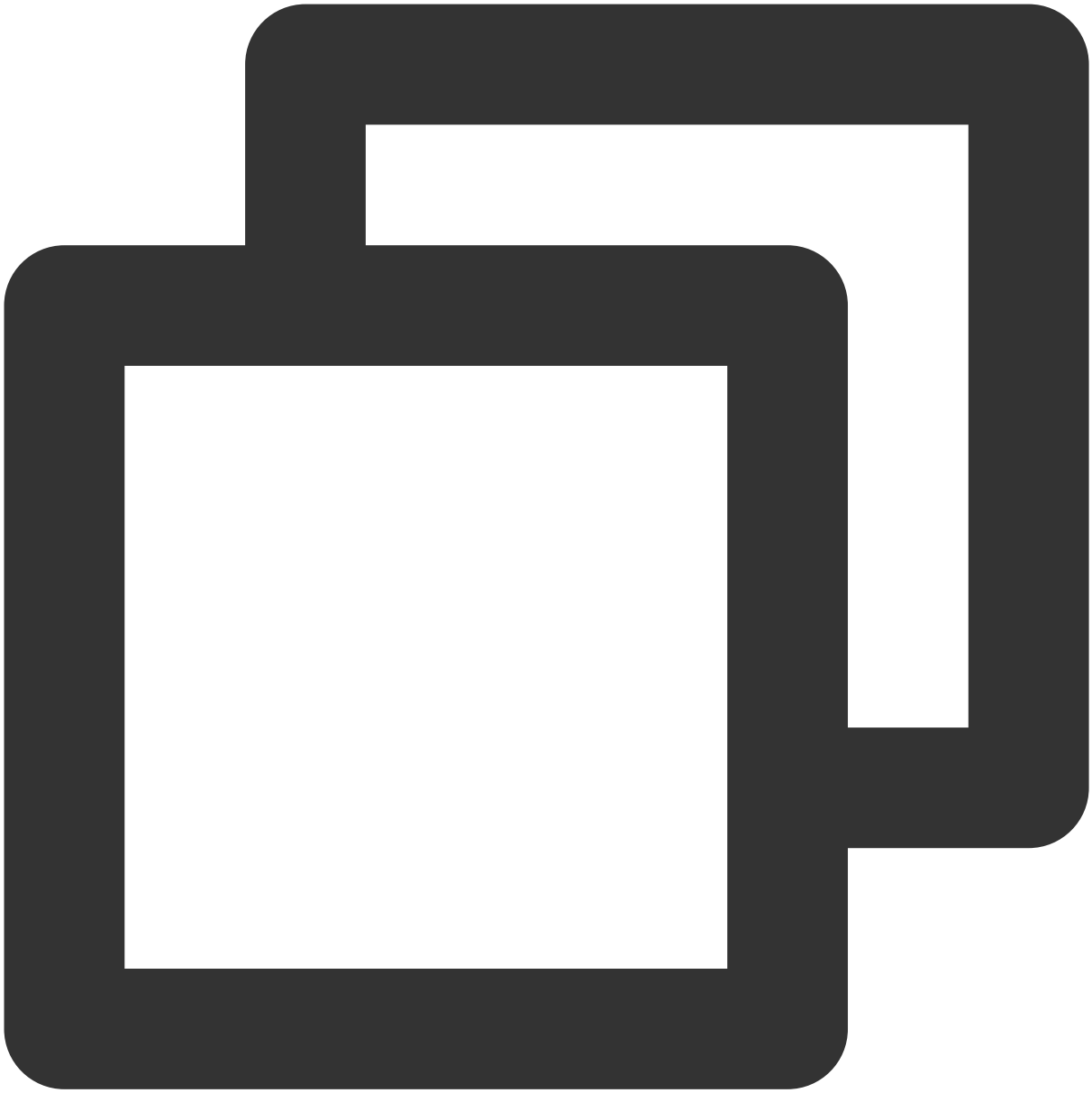
restore

The method is utilized as CanvasContext.restore()

Function Description: Restore the previously saved drawing context.

Sample Code:

©2013-2022 Tencent Cloud. All rights reserved.



```
const ctx = wx.createCanvasContext('myCanvas')

// save the default fill style
ctx.save()
ctx.setFillStyle('red')
ctx.fillRect(10, 10, 150, 100)

// restore to the previous saved state
ctx.restore()
ctx.fillRect(50, 50, 150, 100)
```

©2013-2022 Tencent Cloud. All rights reserved.

```
ctx.draw()
```

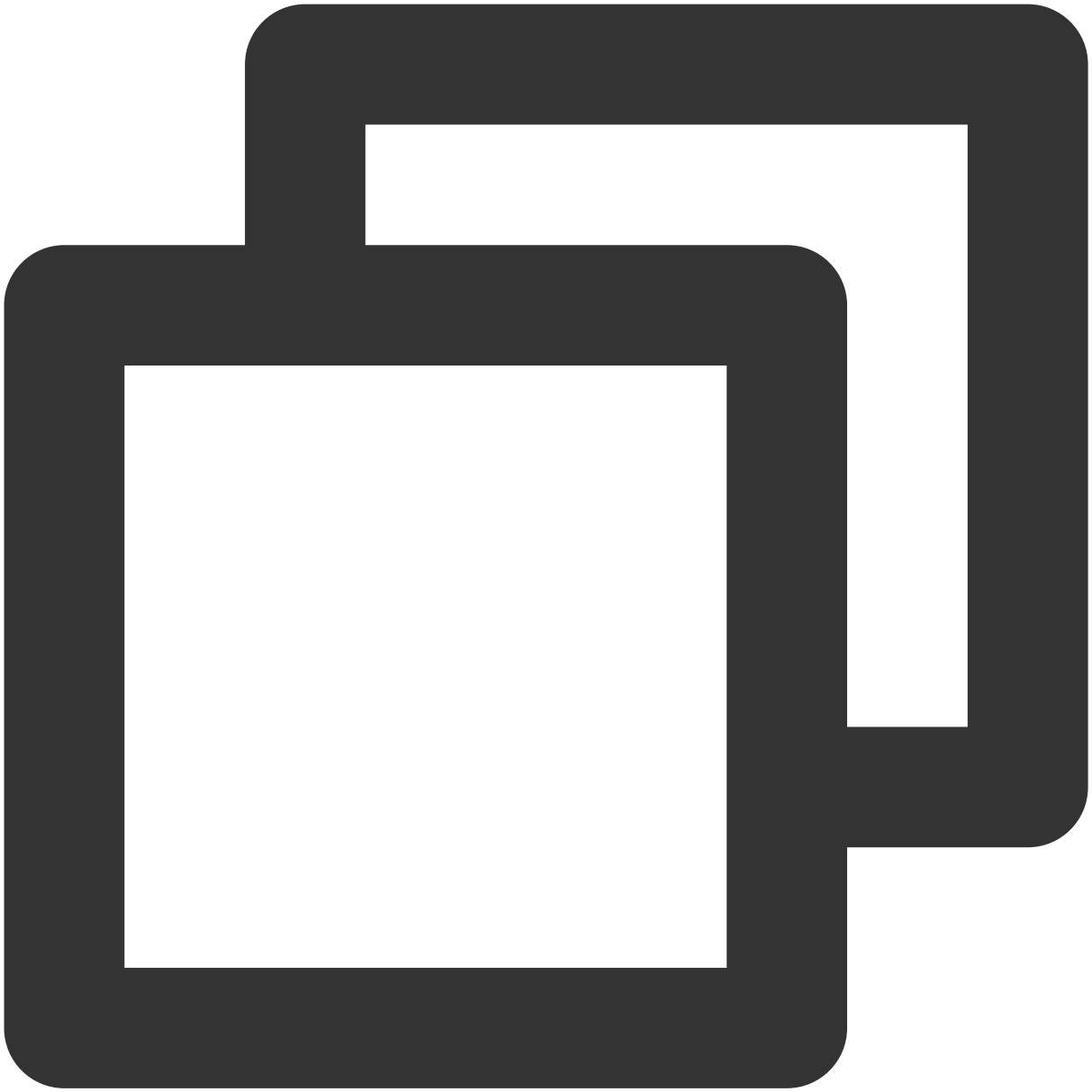


beginPath

> The method is utilized as CanvasContext.beginPath()

Function Description: Initiate the creation of a path. It necessitates the invocation of fill or stroke to utilize the path for filling or outlining.
At the outset, it is equivalent to invoking beginPath once.
Within the same path, multiple settings such as setFillStyle, setStrokeStyle, setLineWidth, etc., the final setting prevails.
Sample Code:

©2013-2022 Tencent Cloud. All rights reserved.



```
const ctx = wx.createCanvasContext('myCanvas')
// begin path
ctx.rect(10, 10, 100, 30)
ctx.setFillStyle('yellow')
ctx.fill()

// begin another path
ctx.beginPath()
ctx.rect(10, 40, 100, 30)

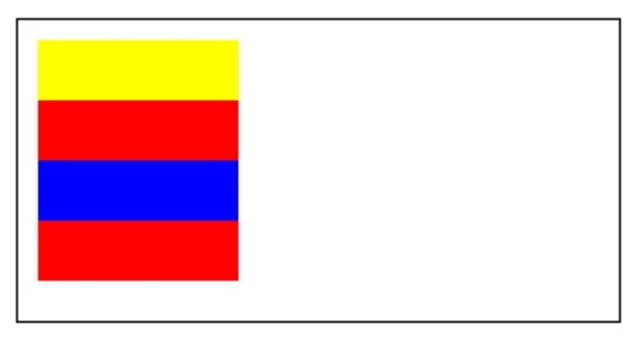
// only fill this rect, not in current path
```

©2013-2022 Tencent Cloud. All rights reserved.

```
ctx.setFillStyle('blue')
ctx.fillRect(10, 70, 100, 30)

ctx.rect(10, 100, 100, 30)

// it will fill current path
ctx.setFillStyle('red')
ctx.fill()
ctx.draw()
```



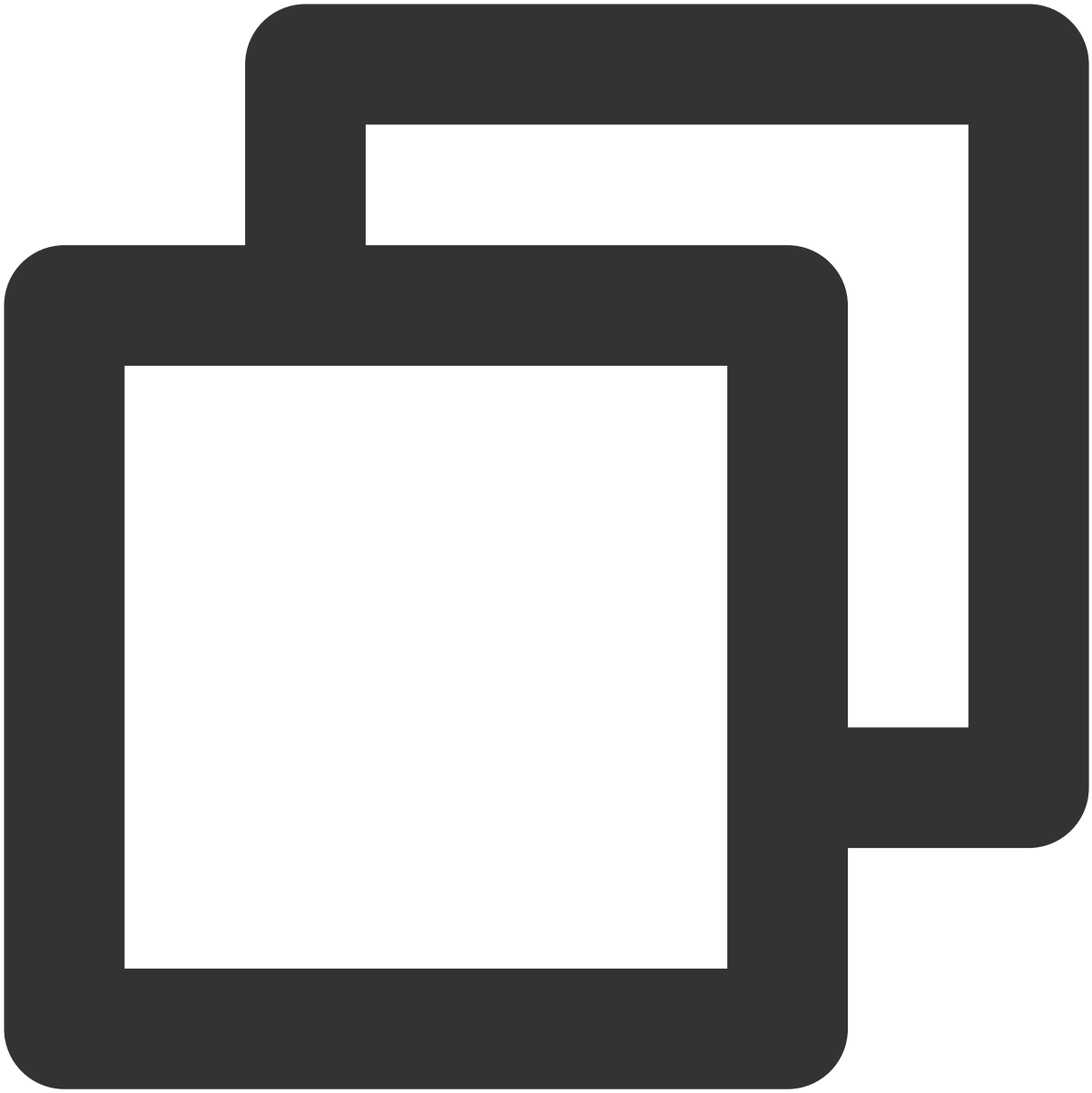
closePath

> The method is utilized as CanvasContext.closePath()

Function Description: Close a path. This will connect the start and end points. If the path is closed and fill or stroke is not invoked, and a new path is initiated, the previous path will not be rendered.

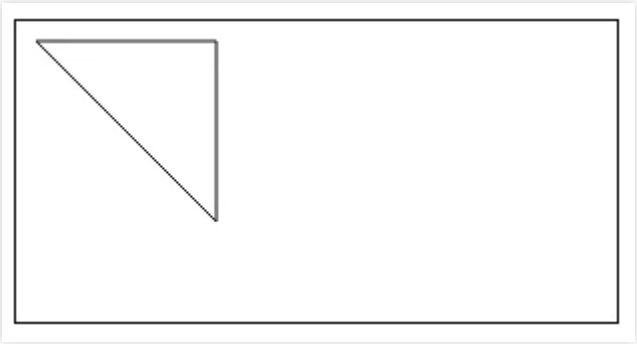
Sample Code:

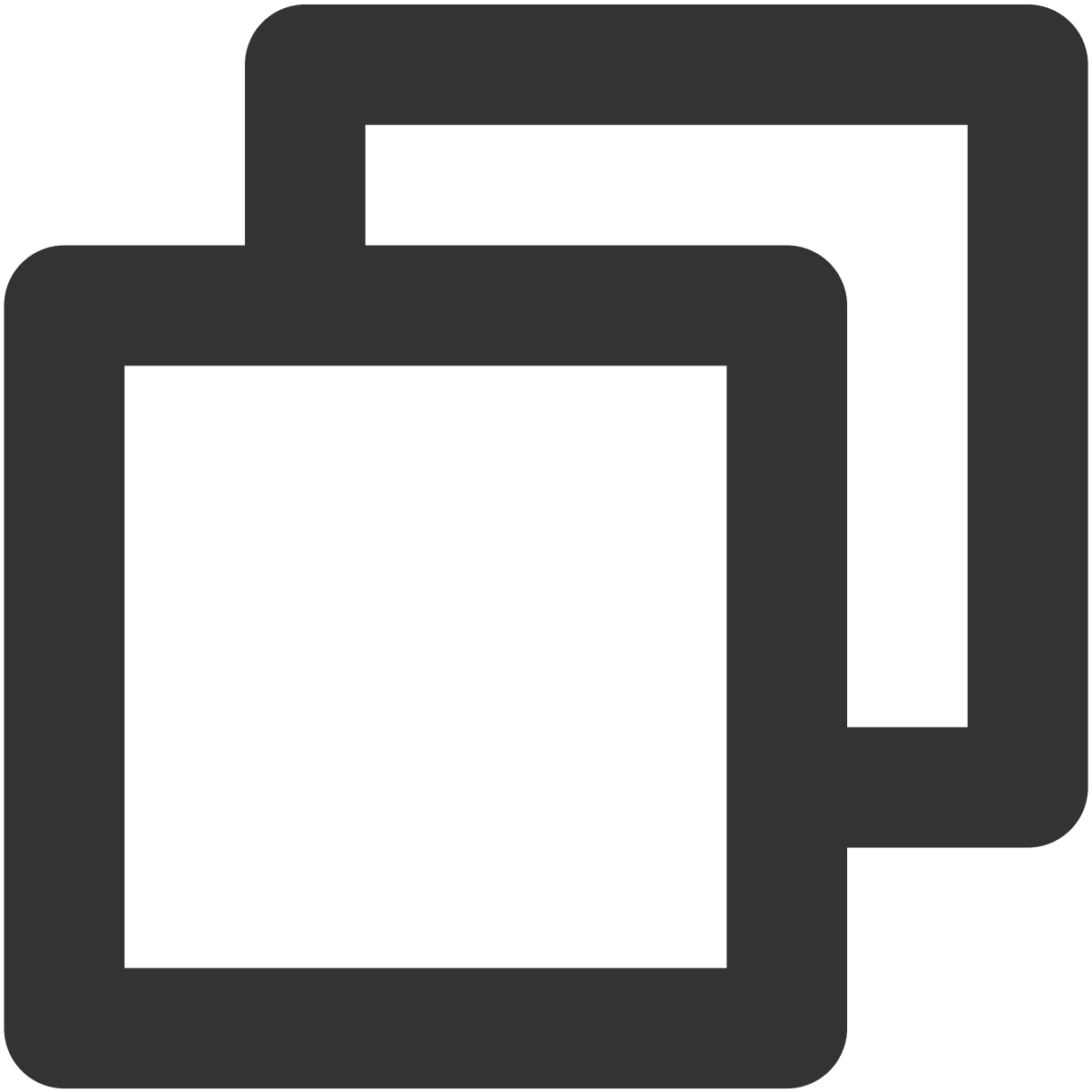
©2013-2022 Tencent Cloud. All rights reserved.



```
const ctx = wx.createCanvasContext('myCanvas')
ctx.moveTo(10, 10)
ctx.lineTo(100, 10)
ctx.lineTo(100, 100)
ctx.closePath()
ctx.stroke()
ctx.draw()
```

©2013-2022 Tencent Cloud. All rights reserved.





```
const ctx = wx.createCanvasContext('myCanvas')
// begin path
```

©2013-2022 Tencent Cloud. All rights reserved.

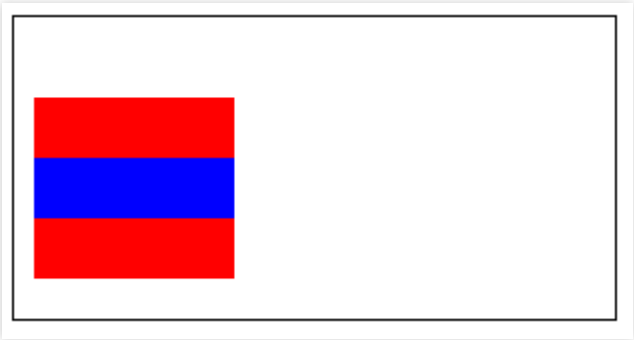
```
ctx.rect(10, 10, 100, 30)
ctx.closePath()

// begin another path
ctx.beginPath()
ctx.rect(10, 40, 100, 30)

// only fill this rect, not in current path
ctx.setFillStyle('blue')
ctx.fillRect(10, 70, 100, 30)

ctx.rect(10, 100, 100, 30)

// it will fill current path
ctx.setFillStyle('red')
ctx.fill()
ctx.draw()
```



moveTo

The method is utilized as CanvasContext.moveTo(number x, number y)

Function Description: Move the path to a specified point in the canvas, without the need to create a line. Utilize the stroke method to draw the line.

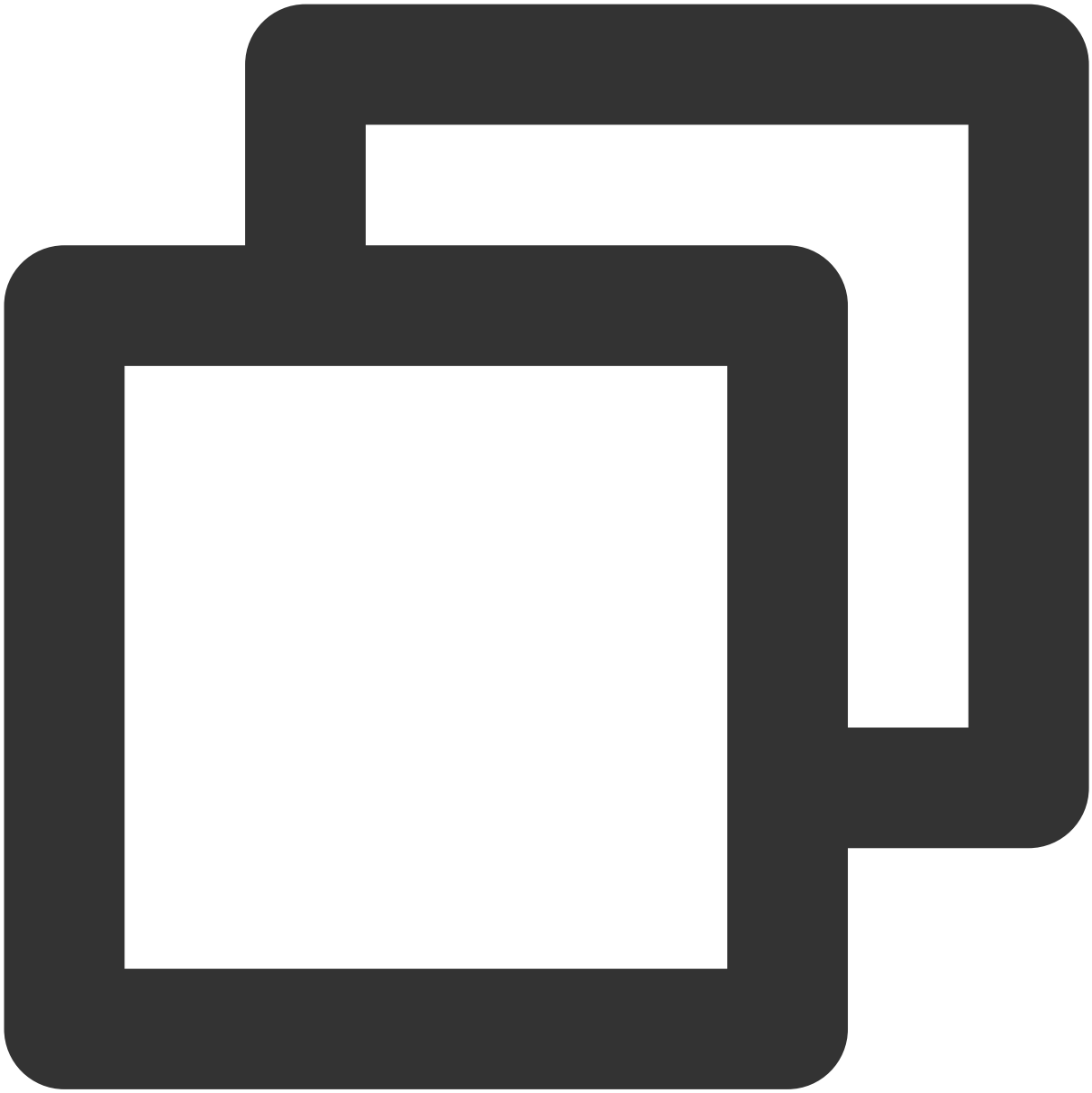
Parameters and Description:

number x, the x-coordinate of the target position;

number y, the y-coordinate of the target position.

Sample Code:

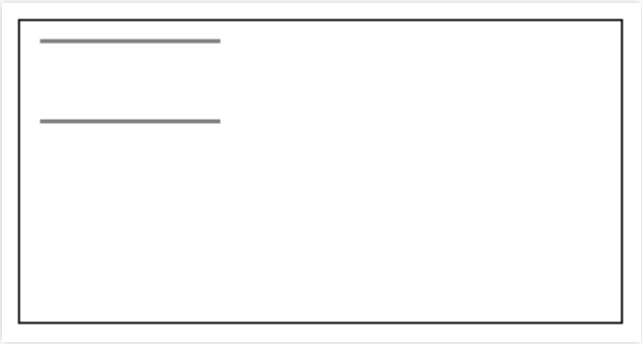
©2013-2022 Tencent Cloud. All rights reserved.



```
const ctx = wx.createCanvasContext('myCanvas')
ctx.moveTo(10, 10)
ctx.lineTo(100, 10)

ctx.moveTo(10, 50)
ctx.lineTo(100, 50)
ctx.stroke()
ctx.draw()
```

©2013-2022 Tencent Cloud. All rights reserved.



lineTo

The method is utilized as CanvasContext.lineTo(number x, number y)

Function Description: Add a new point, then create a line from the last specified point to the target point, using the stroke method to draw the line.

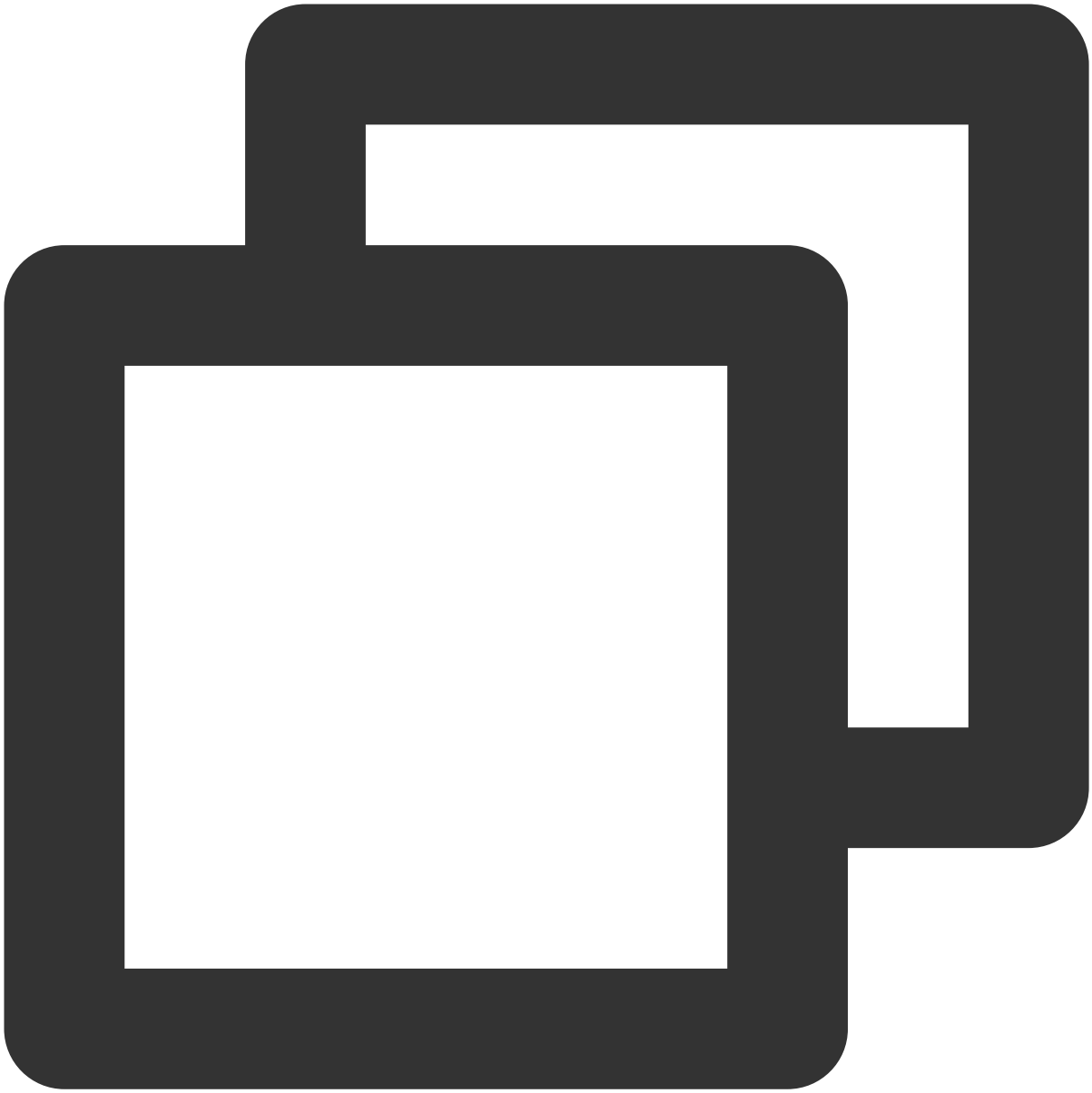
Parameters and Description:

number x, the x-coordinate of the target position;

number y, the y-coordinate of the target position.

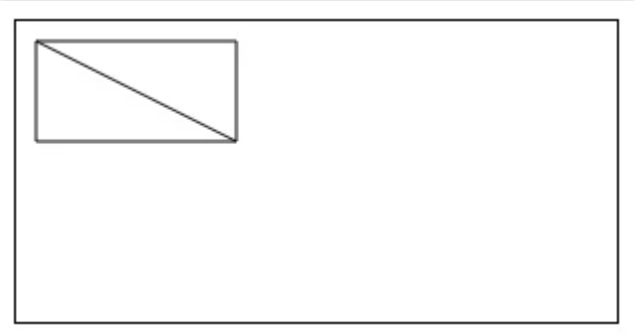
Sample Code:

©2013-2022 Tencent Cloud. All rights reserved.



```
const ctx = wx.createCanvasContext('myCanvas')
ctx.moveTo(10, 10)
ctx.rect(10, 10, 100, 50)
ctx.lineTo(110, 60)
ctx.stroke()
ctx.draw()
```

©2013-2022 Tencent Cloud. All rights reserved.



quadraticCurveTo

The method is utilized as CanvasContext.quadraticCurveTo(number cpx, number cpy, number x, number y)

Function Description: Create a quadratic Bézier curve path. The starting point of the curve is the previous point in the path.

Parameters and Description:

number cpx, the x-coordinate of the Bézier control point;

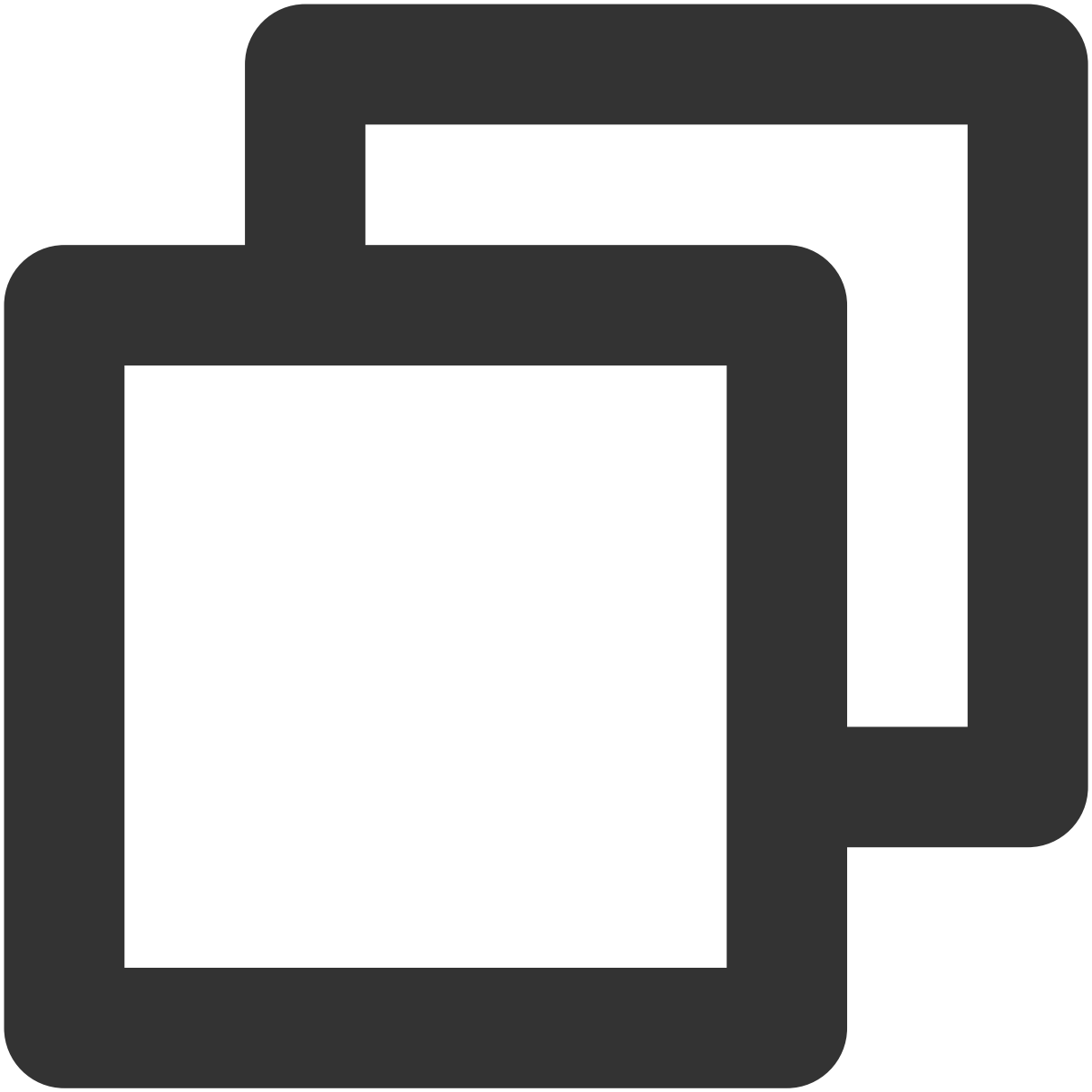
number cpy, the y-coordinate of the Bézier control point;

number x, the x-coordinate of the end point;

number y, the y-coordinate of the end point.

Sample Code:

©2013-2022 Tencent Cloud. All rights reserved.



```
const ctx = wx.createCanvasContext('myCanvas')

// Draw points
ctx.beginPath()
ctx.arc(20, 20, 2, 0, 2 * Math.PI)
ctx.setFillStyle('red')
ctx.fill()

ctx.beginPath()
ctx.arc(200, 20, 2, 0, 2 * Math.PI)
ctx.setFillStyle('lightgreen')
```

©2013-2022 Tencent Cloud. All rights reserved.

```
ctx.fill()

ctx.beginPath()
ctx.arc(20, 100, 2, 0, 2 * Math.PI)
ctx.setFillStyle('blue')
ctx.fill()

ctx.setFillStyle('black')
ctx.setFontSize(12)

// Draw guides
ctx.beginPath()
ctx.moveTo(20, 20)
ctx.lineTo(20, 100)
ctx.lineTo(200, 20)
ctx.setStrokeStyle('#AAAAAA')
ctx.stroke()

// Draw quadratic curve
ctx.beginPath()
ctx.moveTo(20, 20)
ctx.quadraticCurveTo(20, 100, 200, 20)
ctx.setStrokeStyle('black')
ctx.stroke()

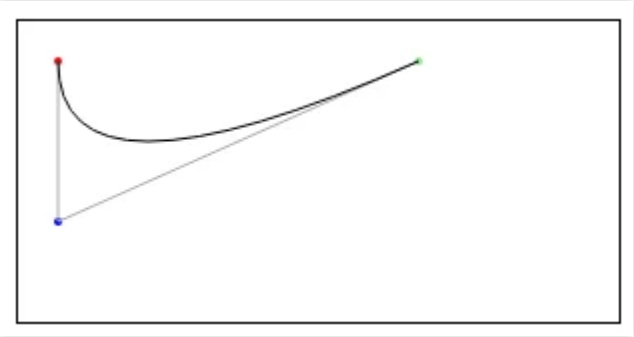
ctx.draw()
```

The three key coordinates for moveTo(20, 20) quadraticCurveTo(20, 100, 200, 20) are as follows:

Red: Starting point (20, 20)

Blue: Control point (20, 100)

Green: Termination point (200, 20)



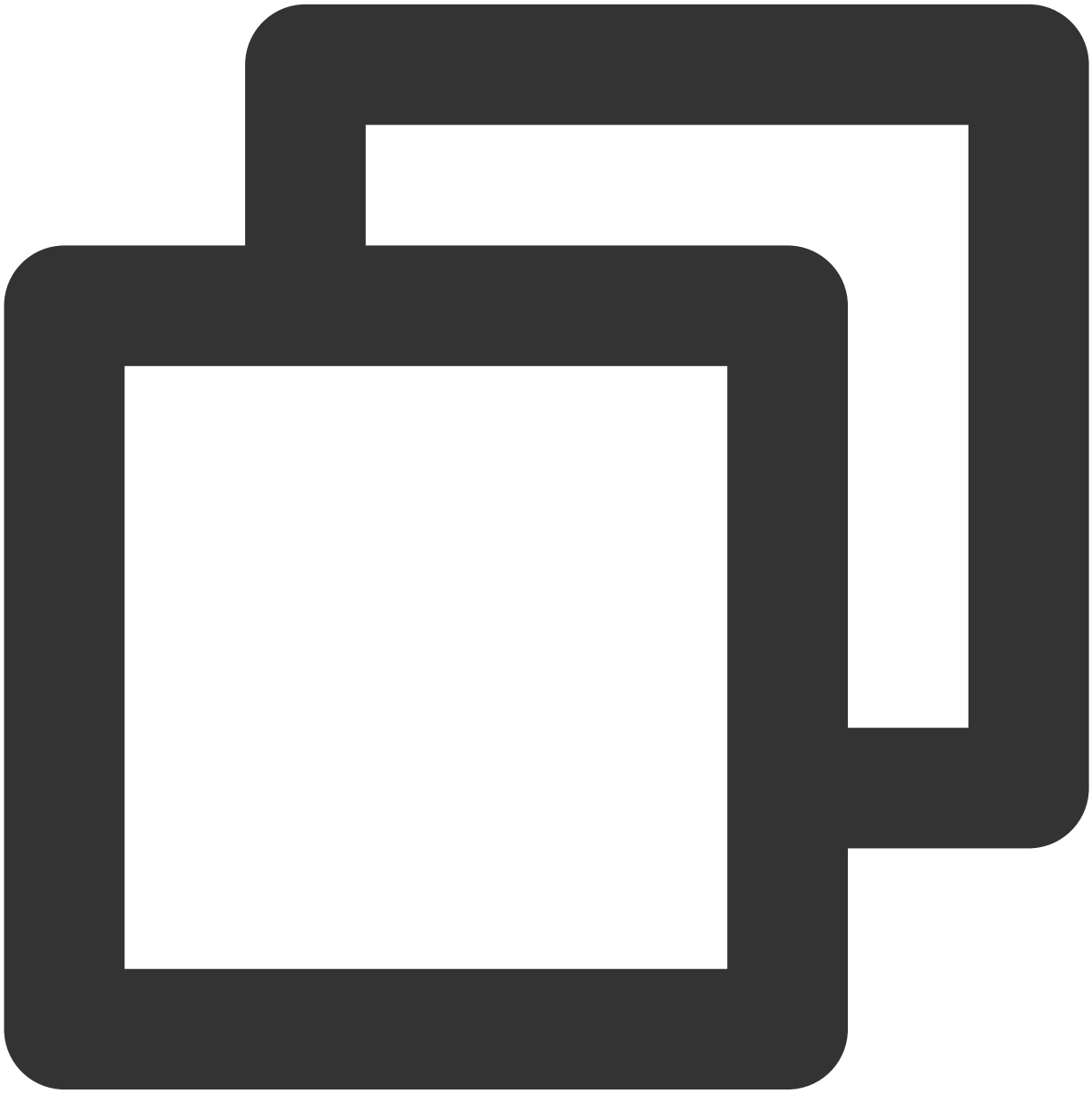
bezierCurveTo

The method is used as follows: CanvasContext.bezierCurveTo(number cp1x, number cp1y, number cp2x, number cp2y, number x, number y)

©2013-2022 Tencent Cloud. All rights reserved.

Function Description: Creates a cubic Bezier curve path. The starting point of the curve is the previous point in the path.

Sample Code:



```
const ctx = wx.createCanvasContext('myCanvas')

// Draw points
ctx.beginPath()
ctx.arc(20, 20, 2, 0, 2 * Math.PI)
```

©2013-2022 Tencent Cloud. All rights reserved.

```
ctx.setFillStyle('red')
ctx.fill()

ctx.beginPath()
ctx.arc(200, 20, 2, 0, 2 * Math.PI)
ctx.setFillStyle('lightgreen')
ctx.fill()

ctx.beginPath()
ctx.arc(20, 100, 2, 0, 2 * Math.PI)
ctx.arc(200, 100, 2, 0, 2 * Math.PI)
ctx.setFillStyle('blue')
ctx.fill()

ctx.setFillStyle('black')
ctx.setFontSize(12)

// Draw guides
ctx.beginPath()
ctx.moveTo(20, 20)
ctx.lineTo(20, 100)
ctx.lineTo(150, 75)

ctx.moveTo(200, 20)
ctx.lineTo(200, 100)
ctx.lineTo(70, 75)
ctx.setStrokeStyle('#AAAAAA')
ctx.stroke()

// Draw quadratic curve
ctx.beginPath()
ctx.moveTo(20, 20)
ctx.bezierCurveTo(20, 100, 200, 100, 200, 20)
ctx.setStrokeStyle('black')
ctx.stroke()

ctx.draw()
```

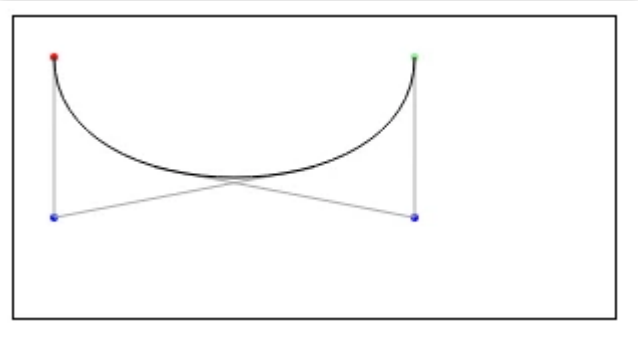
The three key coordinates for moveTo(20, 20) bezierCurveTo(20, 100, 200, 100, 200, 20) are as follows:

Red: Starting point (20, 20)

Blue: Two control points (20, 100) (200, 100)

Green: Termination point (200, 20)

©2013-2022 Tencent Cloud. All rights reserved.



rect

The method is used as follows: CanvasContext.rect(number x, number y, number width, number height)

Function Description: Creates a rectangular path. The rectangle must be actually drawn on the canvas using the fill or stroke method.

Parameters and Descriptions:

number x, the x-coordinate of the top-left corner of the rectangular path;

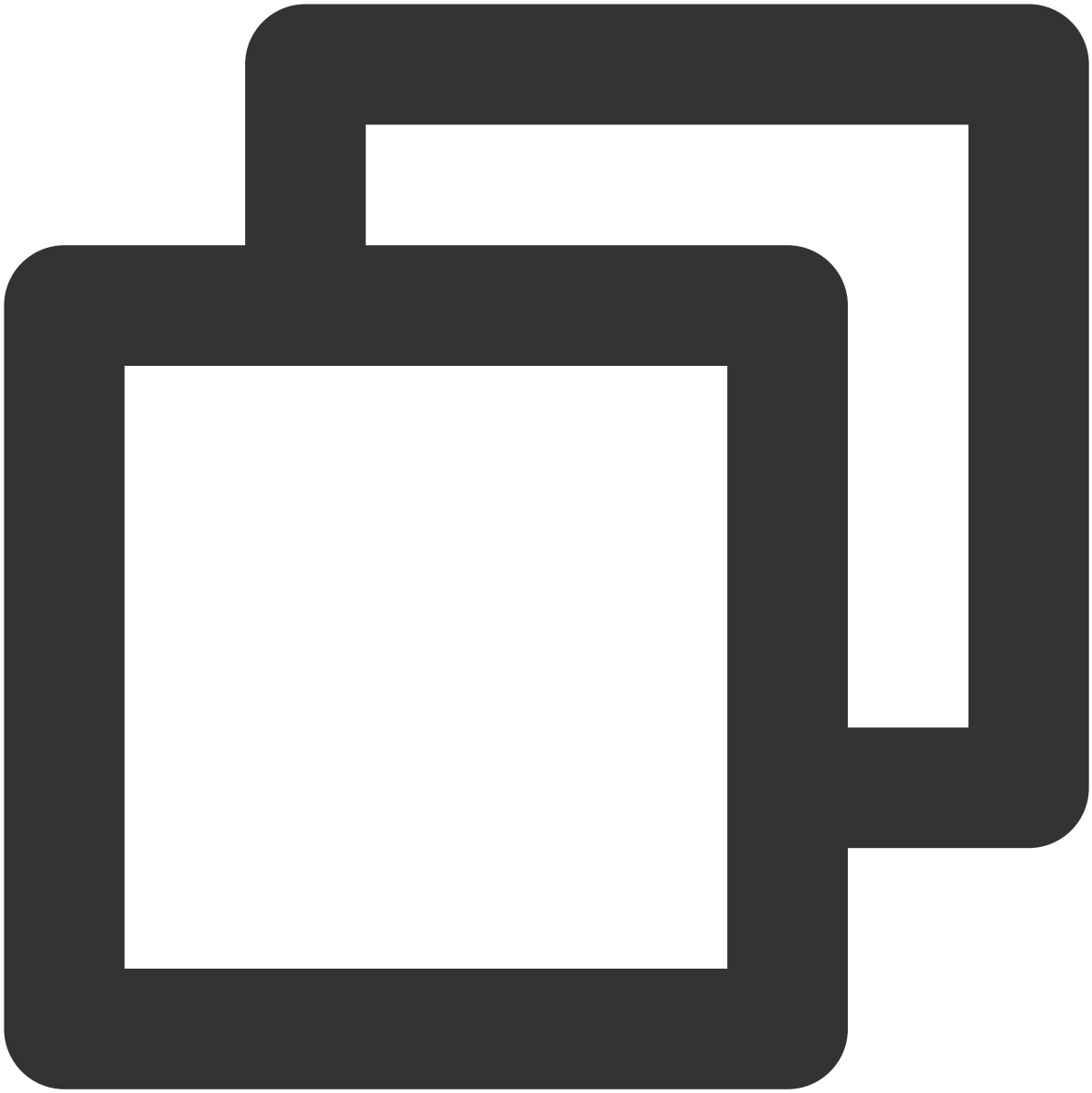
number y, the y-coordinate of the top-left corner of the rectangular path;

number width, the width of the rectangular path;

number height, the height of the rectangular path.

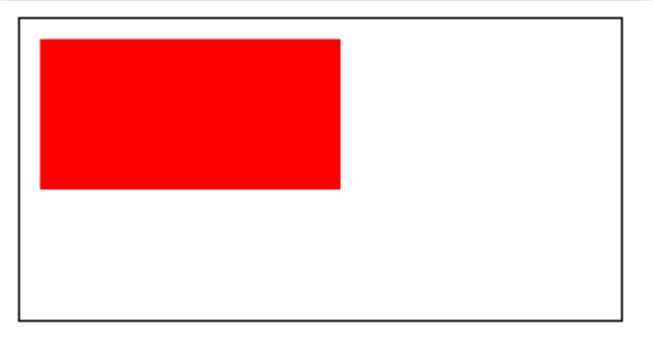
Sample Code:

©2013-2022 Tencent Cloud. All rights reserved.



```
const ctx = wx.createCanvasContext('myCanvas')
ctx.rect(10, 10, 150, 75)
ctx.setFillStyle('red')
ctx.fill()
ctx.draw()
```

©2013-2022 Tencent Cloud. All rights reserved.



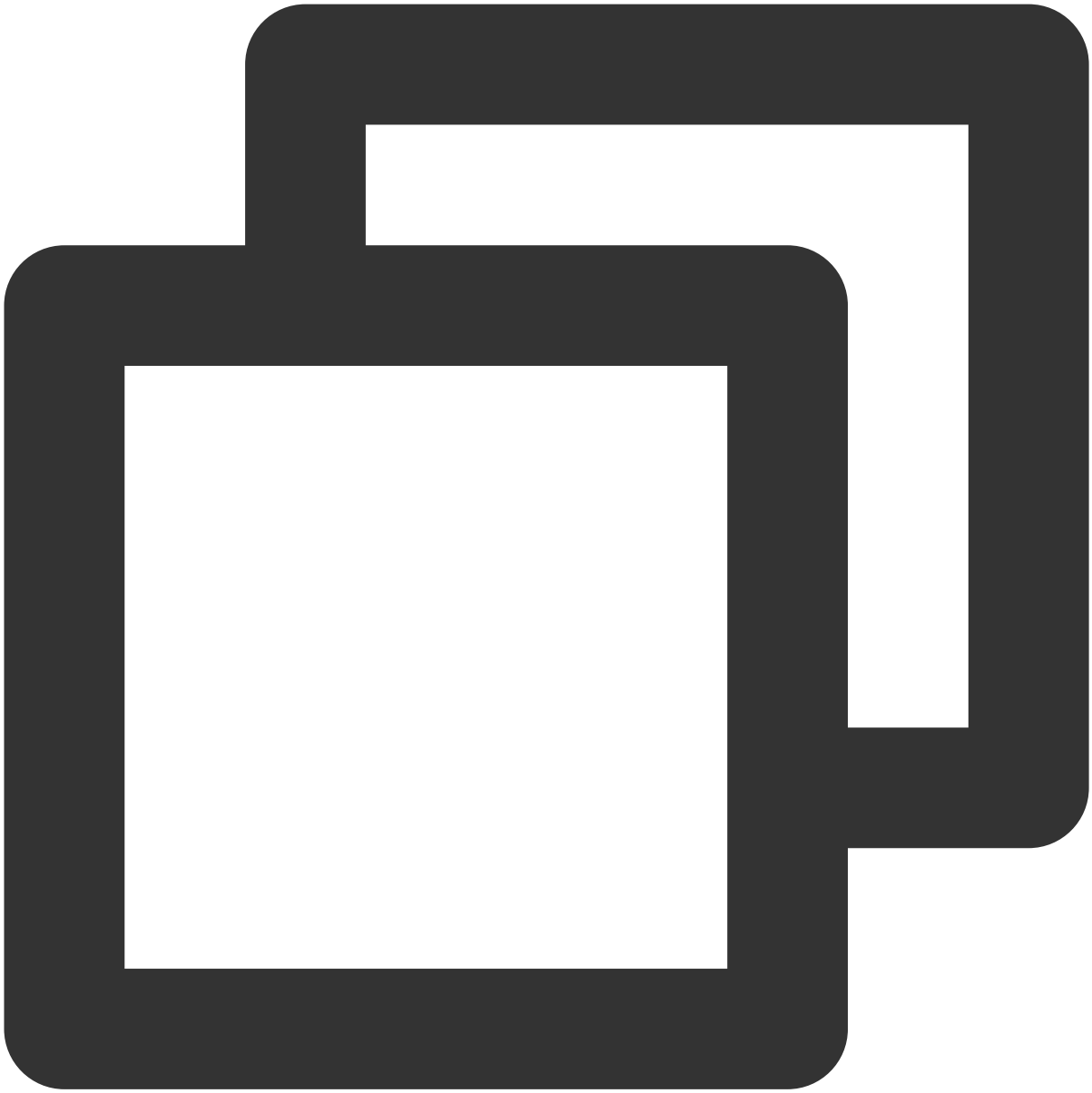
clip

The method is used as follows: CanvasContext.clip()

Function Description: Cuts any shape and size from the original canvas. Once an area is cut, all subsequent drawings will be confined within the cut area (unable to access other areas on the canvas). The current canvas area can be saved before using the clip method by using the save method, and can be restored at any time in the future using the restore method.

Example Code

©2013-2022 Tencent Cloud. All rights reserved.

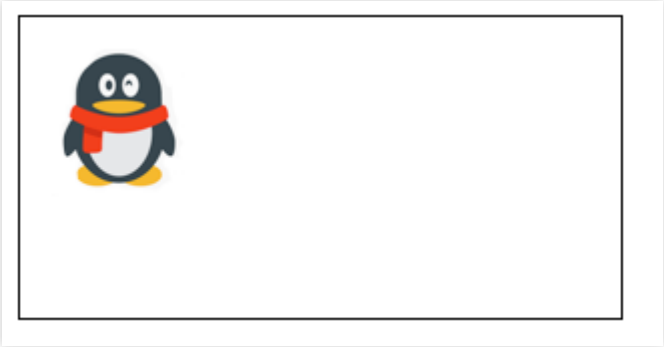


```
const ctx = wx.createCanvasContext('myCanvas')

wx.downloadFile({
  url: 'https://is5.mzstatic.com/image/thumb/Purple128/v4/75/3b/90/753b907c-b7fb-58
  success(res) {
    ctx.save()
    ctx.beginPath()
    ctx.arc(50, 50, 25, 0, 2 * Math.PI)
    ctx.clip()
    ctx.drawImage(res.tempFilePath, 25, 25)
    ctx.restore()
```

©2013-2022 Tencent Cloud. All rights reserved.

```
    ctx.draw()
  }
})
```



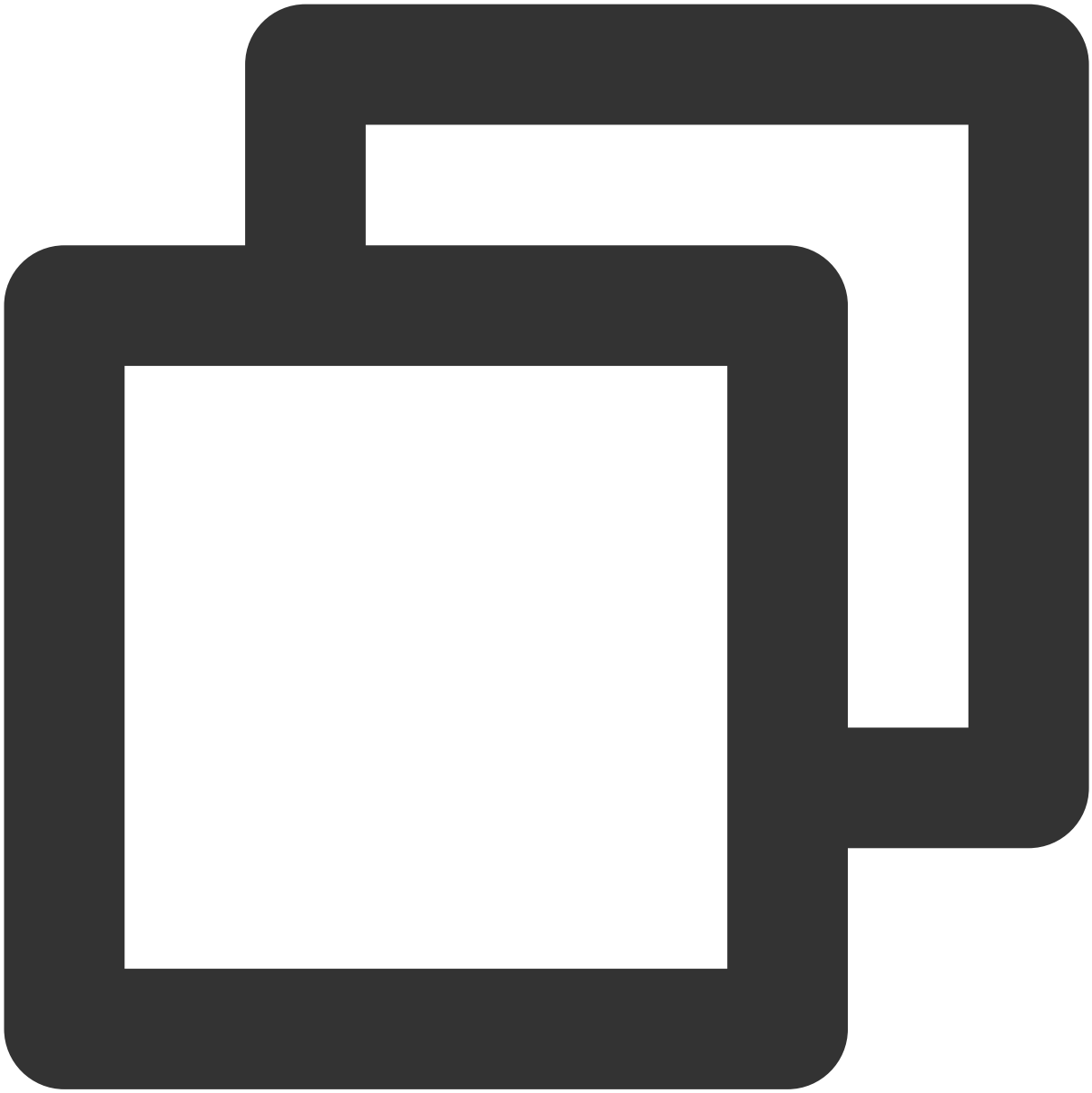
stroke

The method is used as follows: CanvasContext.stroke()

Function Description: Draws the border of the current path. The default color is black.

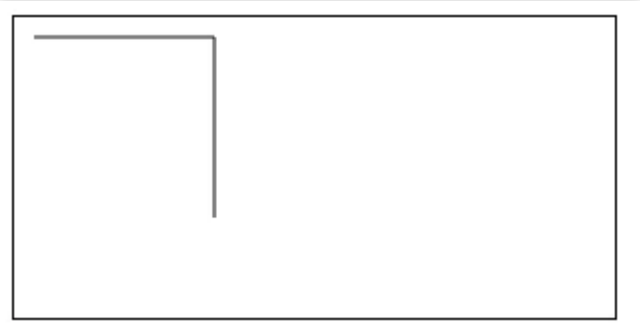
Sample Code:

©2013-2022 Tencent Cloud. All rights reserved.

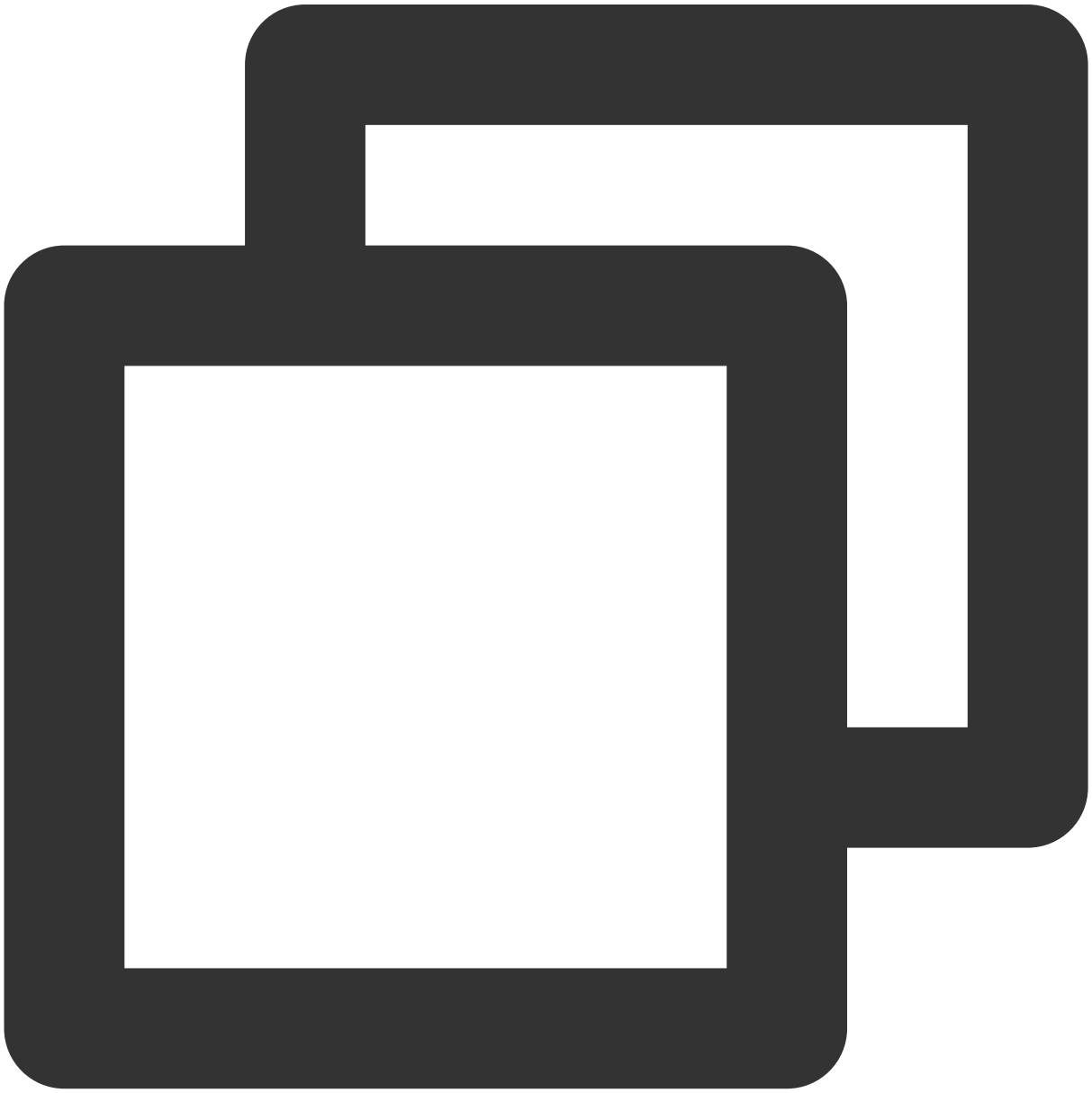


```
const ctx = wx.createCanvasContext('myCanvas')
ctx.moveTo(10, 10)
ctx.lineTo(100, 10)
ctx.lineTo(100, 100)
ctx.stroke()
ctx.draw()
```

©2013-2022 Tencent Cloud. All rights reserved.



The path depicted by stroke() is calculated from beginPath(), but it does not include strokeRect().



©2013-2022 Tencent Cloud. All rights reserved.

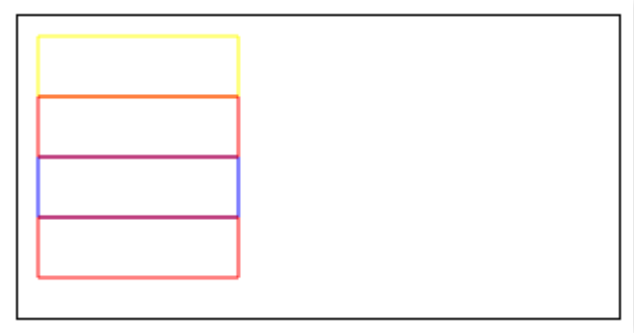
```
const ctx = wx.createCanvasContext('myCanvas')
// begin path
ctx.rect(10, 10, 100, 30)
ctx.setStrokeStyle('yellow')
ctx.stroke()

// begin another path
ctx.beginPath()
ctx.rect(10, 40, 100, 30)

// only stoke this rect, not in current path
ctx.setStrokeStyle('blue')
ctx.strokeRect(10, 70, 100, 30)

ctx.rect(10, 100, 100, 30)

// it will stroke current path
ctx.setStrokeStyle('red')
ctx.stroke()
ctx.draw()
```



strokeRect

The method is used as follows: CanvasContext.strokeRect(number x, number y, number width, number height)

Function Description: Draws a rectangle (non-filled). The color of the rectangle's outline is set using setStrokeStyle, if not set, the default color is black.

Parameters and Descriptions:

number x, the x-coordinate of the top-left corner of the rectangular path;

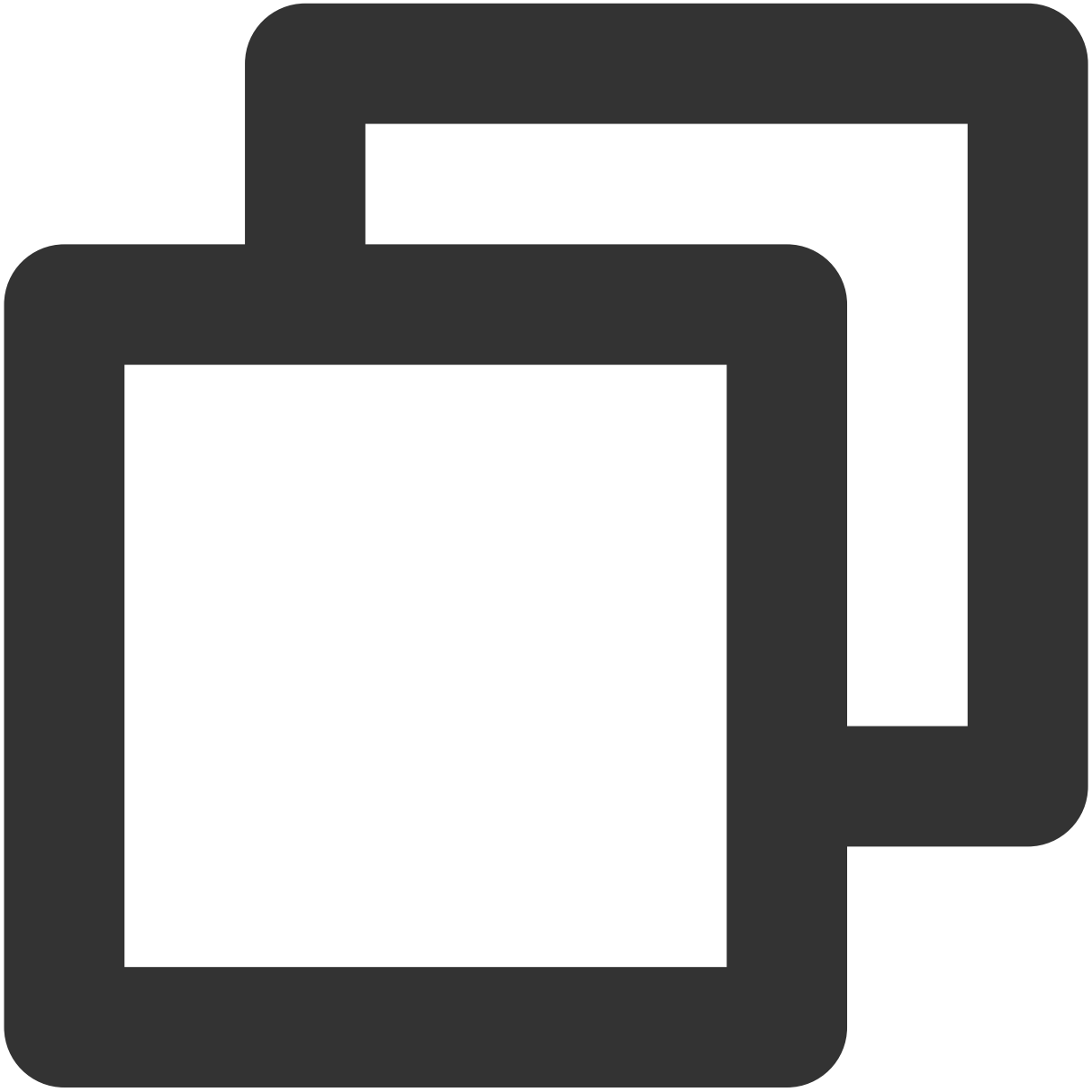
number y, the y-coordinate of the top-left corner of the rectangular path;

©2013-2022 Tencent Cloud. All rights reserved.

number width, the width of the rectangular path;

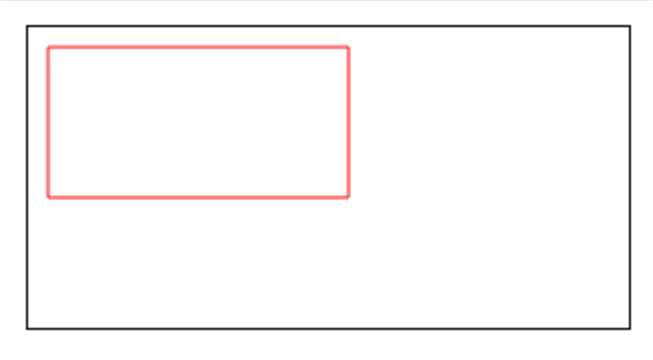
number height, the height of the rectangular path.

Sample Code:



```
const ctx = wx.createCanvasContext('myCanvas')
ctx.setStrokeStyle('red')
ctx.strokeRect(10, 10, 150, 75)
ctx.draw()
```

©2013-2022 Tencent Cloud. All rights reserved.



strokeText

The method is used as follows: CanvasContext.strokeText(string text, number x, number y, number maxWidth)

Function Description: A method to draw text outlines at the given (x, y) position.

Parameters and Description:

string text, the text to be drawn;

number x, the x-axis coordinate of the text's starting point;

number y, the y-axis coordinate of the text's starting point;

number maxWidth, the maximum width to be drawn, optional.

clearRect

The method is used as follows: CanvasContext.clearRect(number x, number y, number width, number height)

Function Description: Clears the content within the specified rectangular area on the canvas.

Parameters and Descriptions:

number x, the x-coordinate of the top-left corner of the rectangular path;

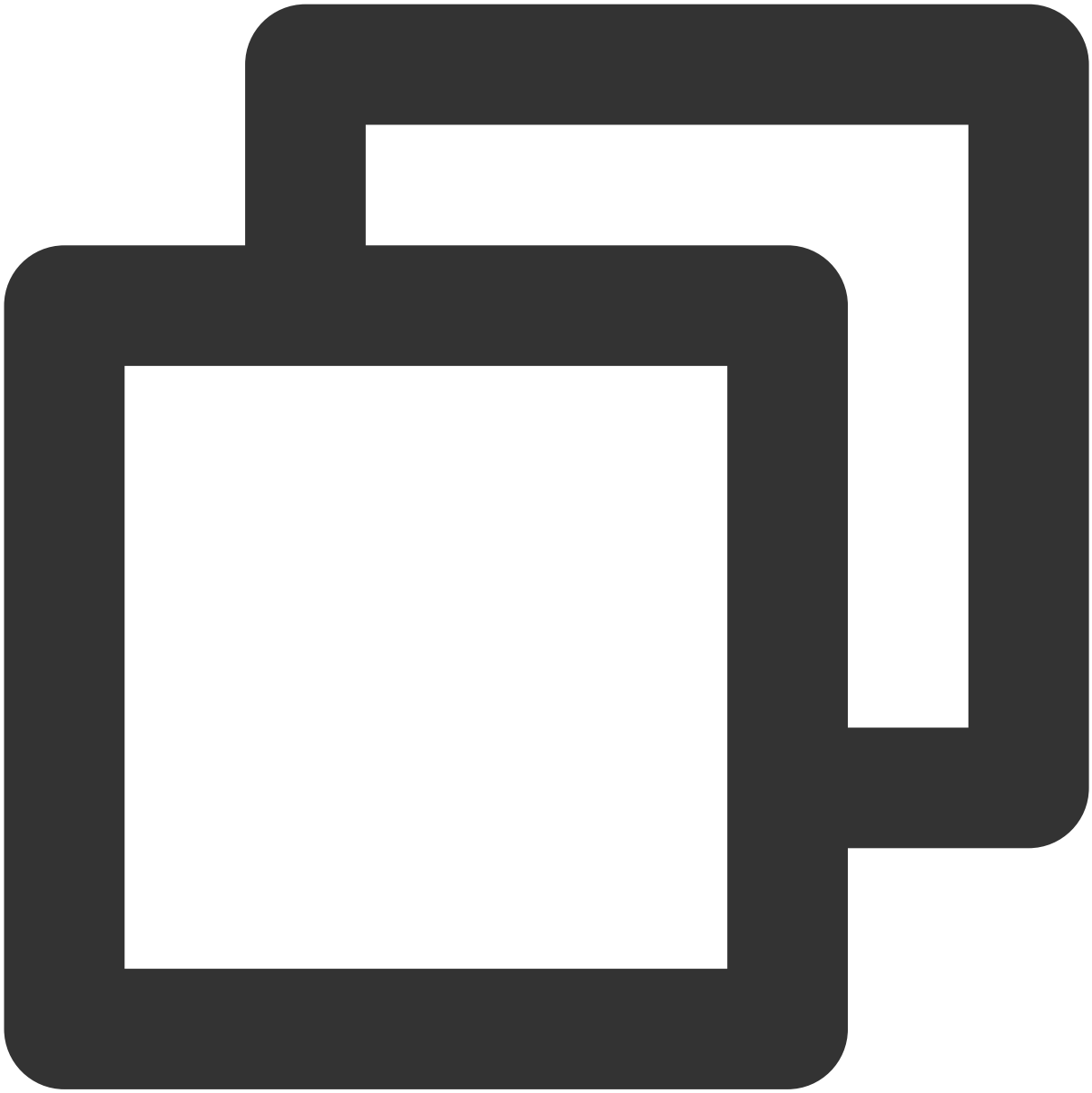
number y, the y-coordinate of the top-left corner of the rectangular path;

number width, the width of the rectangular path;

number height, the height of the rectangular path.

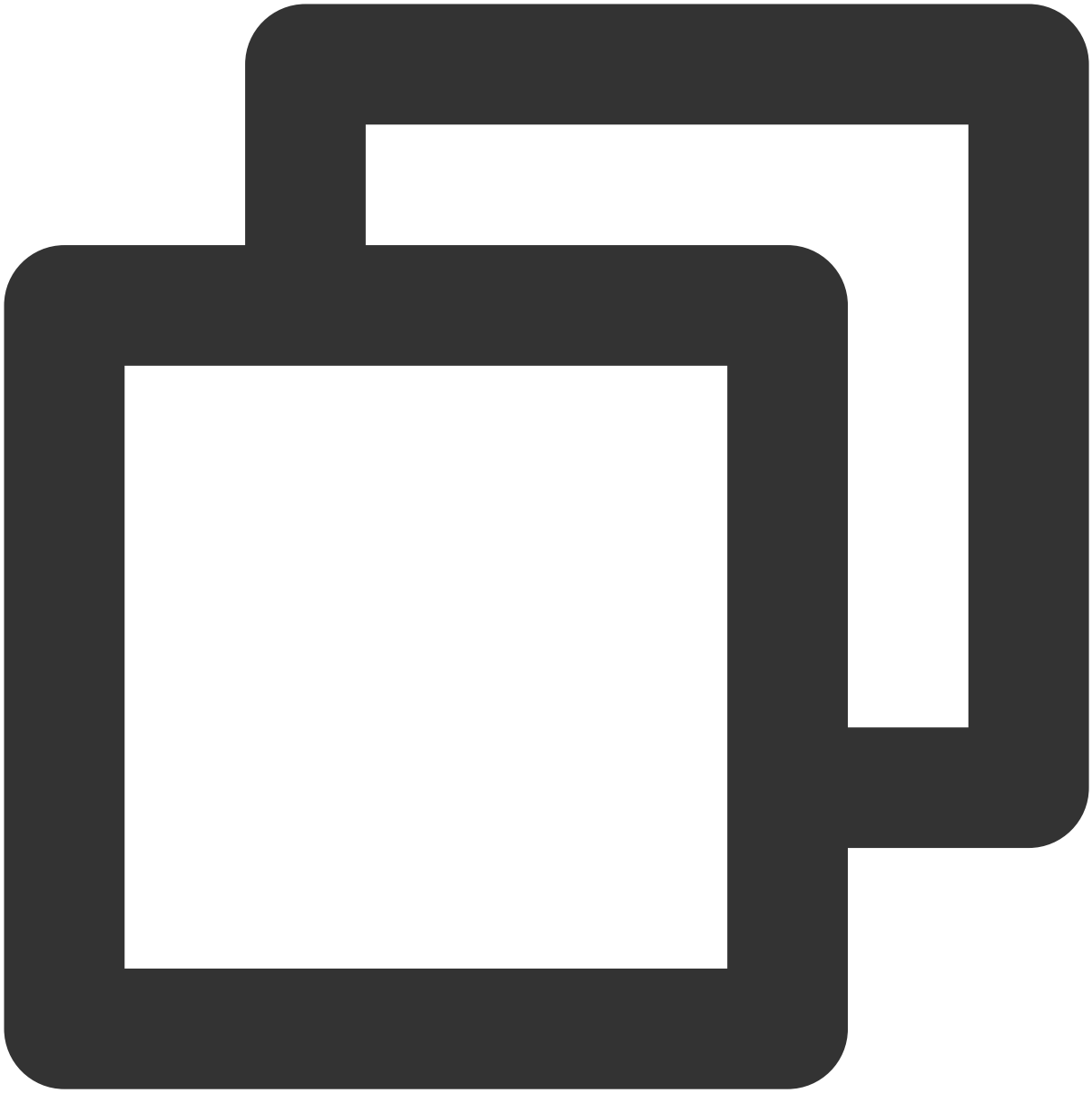
Example Code: clearRect does not draw a white rectangle in the specified area, but rather clears it. For a more intuitive understanding, a background color has been added to the canvas.

©2013-2022 Tencent Cloud. All rights reserved.



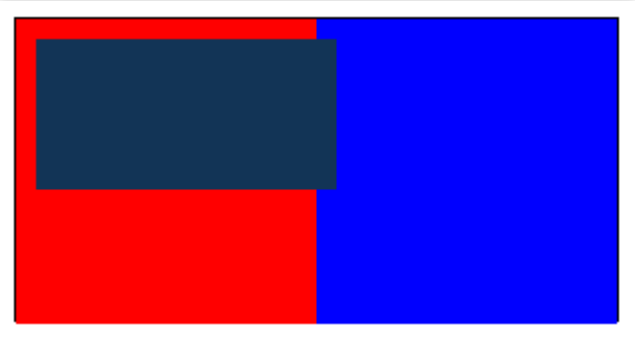
```
<canvas canvas-id="myCanvas" style="border: 1px solid; background: #123456;" />
```

©2013-2022 Tencent Cloud. All rights reserved.



```
const ctx = wx.createCanvasContext('myCanvas')
ctx.setFillStyle('red')
ctx.fillRect(0, 0, 150, 200)
ctx.setFillStyle('blue')
ctx.fillRect(150, 0, 150, 200)
ctx.clearRect(10, 10, 150, 75)
ctx.draw()
```

©2013-2022 Tencent Cloud. All rights reserved.



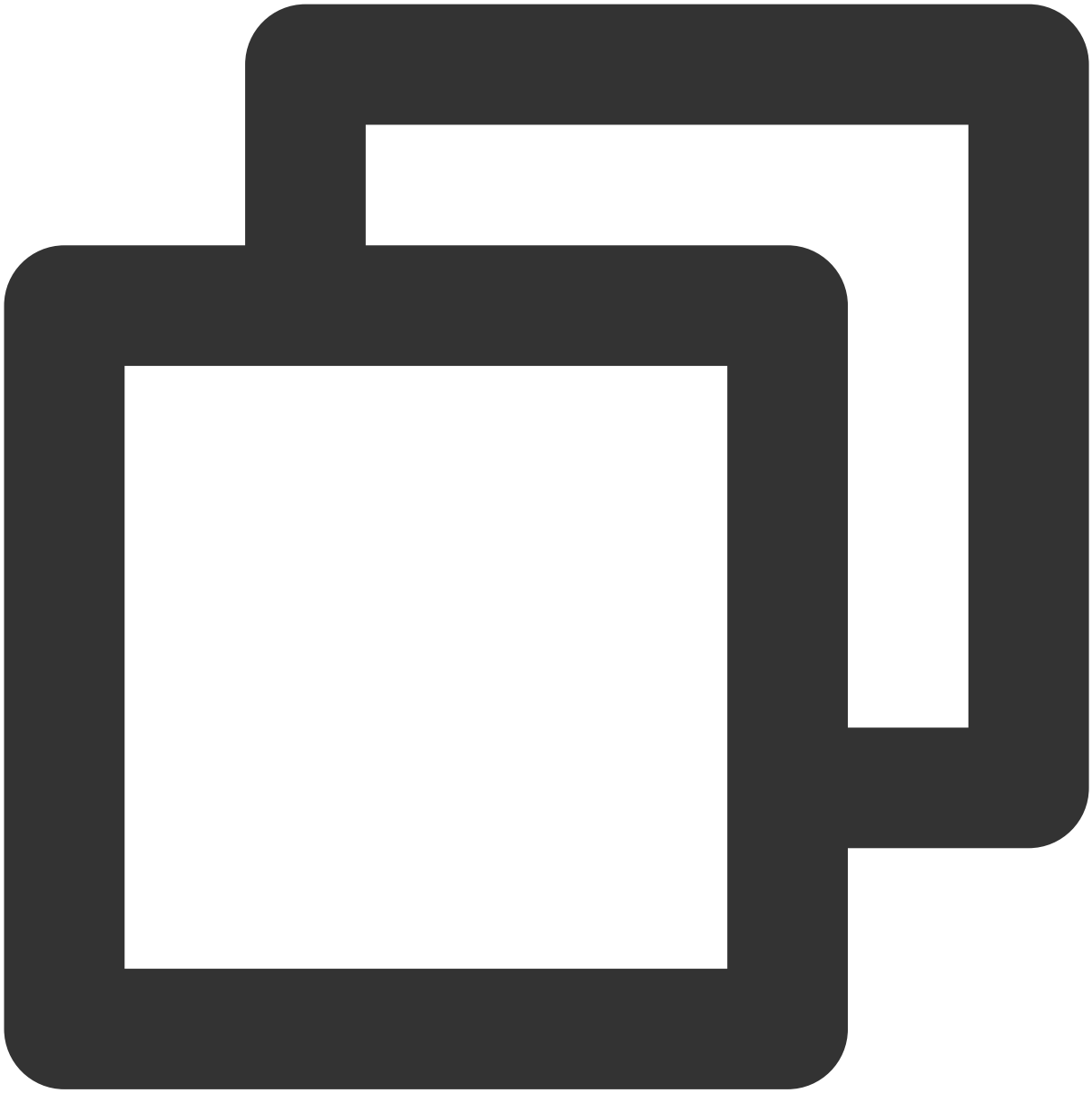
fill

The method is used as follows: CanvasContext.fill()

Function Description: Fills the content within the current path. The default fill color is black.

Sample Code: If the current path is not closed, the fill() method will connect the start and end points, and then fill it.

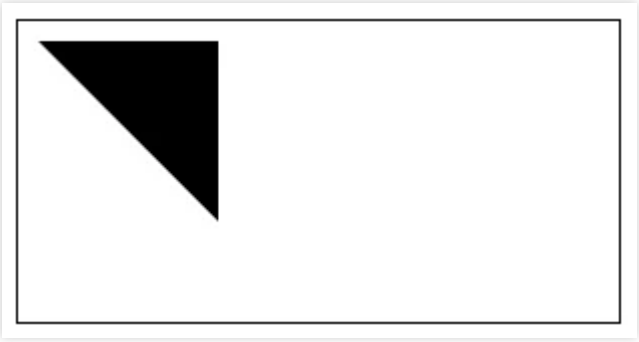
©2013-2022 Tencent Cloud. All rights reserved.

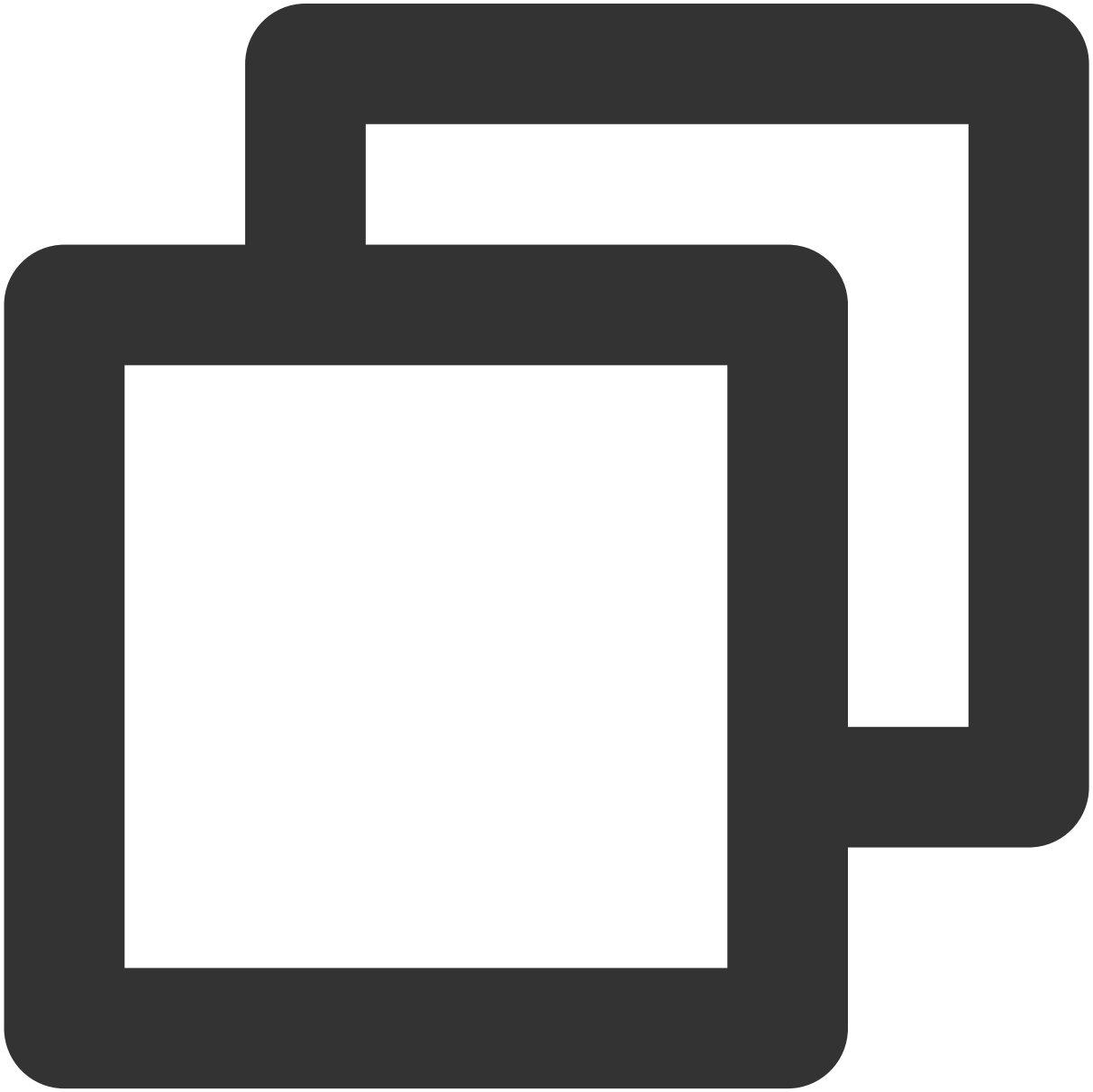


```
const ctx = wx.createCanvasContext('myCanvas')
ctx.moveTo(10, 10)
ctx.lineTo(100, 10)
ctx.lineTo(100, 100)
ctx.fill()
ctx.draw()
```

The path filled by fill() is calculated from beginPath(), but it does not include fillRect().

©2013-2022 Tencent Cloud. All rights reserved.





©2013-2022 Tencent Cloud. All rights reserved.

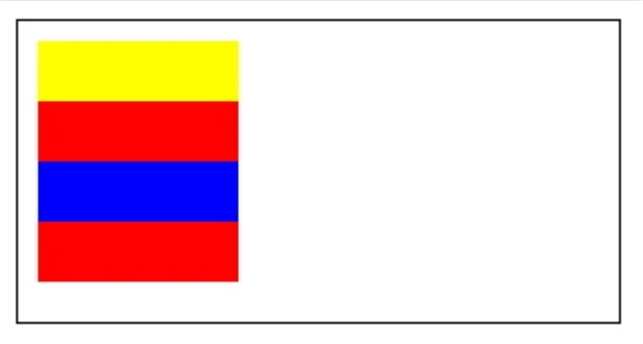
```
const ctx = wx.createCanvasContext('myCanvas')
// begin path
ctx.rect(10, 10, 100, 30)
ctx.setFillStyle('yellow')
ctx.fill()

// begin another path
ctx.beginPath()
ctx.rect(10, 40, 100, 30)

// only fill this rect, not in current path
ctx.setFillStyle('blue')
ctx.fillRect(10, 70, 100, 30)

ctx.rect(10, 100, 100, 30)

// it will fill current path
ctx.setFillStyle('red')
ctx.fill()
ctx.draw()
```



fillRect

The method is used as follows: CanvasContext.fillRect(number x, number y, number width, number height)

Function Description: Fills a rectangle. The fill color of the rectangle is set using setFillStyle, if not set, the default is black.

Parameters and Descriptions:

number x, the x-coordinate of the top-left corner of the rectangular path;

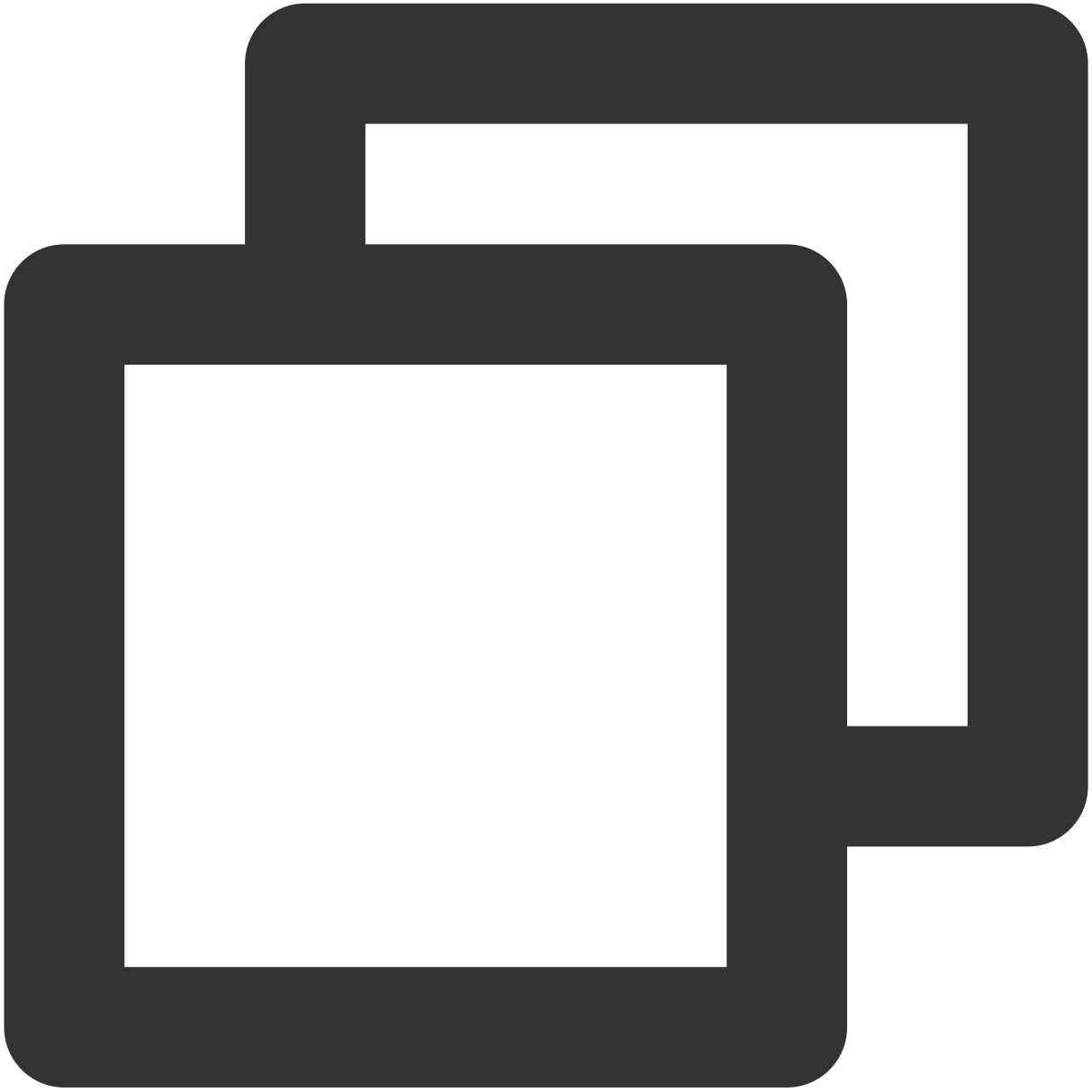
number y, the y-coordinate of the top-left corner of the rectangular path;

©2013-2022 Tencent Cloud. All rights reserved.

number width, the width of the rectangular path;

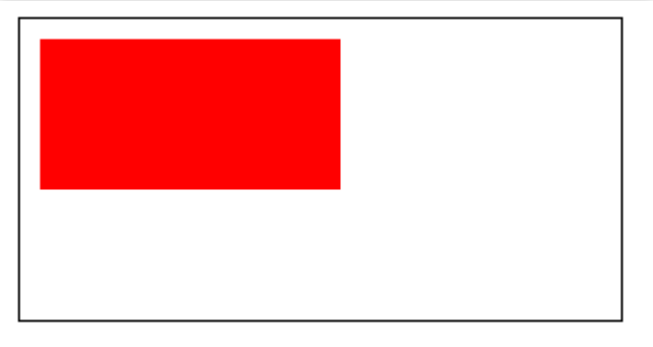
number height, the height of the rectangular path.

Sample Code:



```
const ctx = wx.createCanvasContext('myCanvas')
ctx.setFillStyle('red')
ctx.fillRect(10, 10, 150, 75)
ctx.draw()
```

©2013-2022 Tencent Cloud. All rights reserved.



fillText

The method is used as follows: CanvasContext.fillText(string text, number x, number y, number maxWidth)

Function Description: Draws filled text on the canvas.

Parameters and Description:

string text, the text output on the canvas;

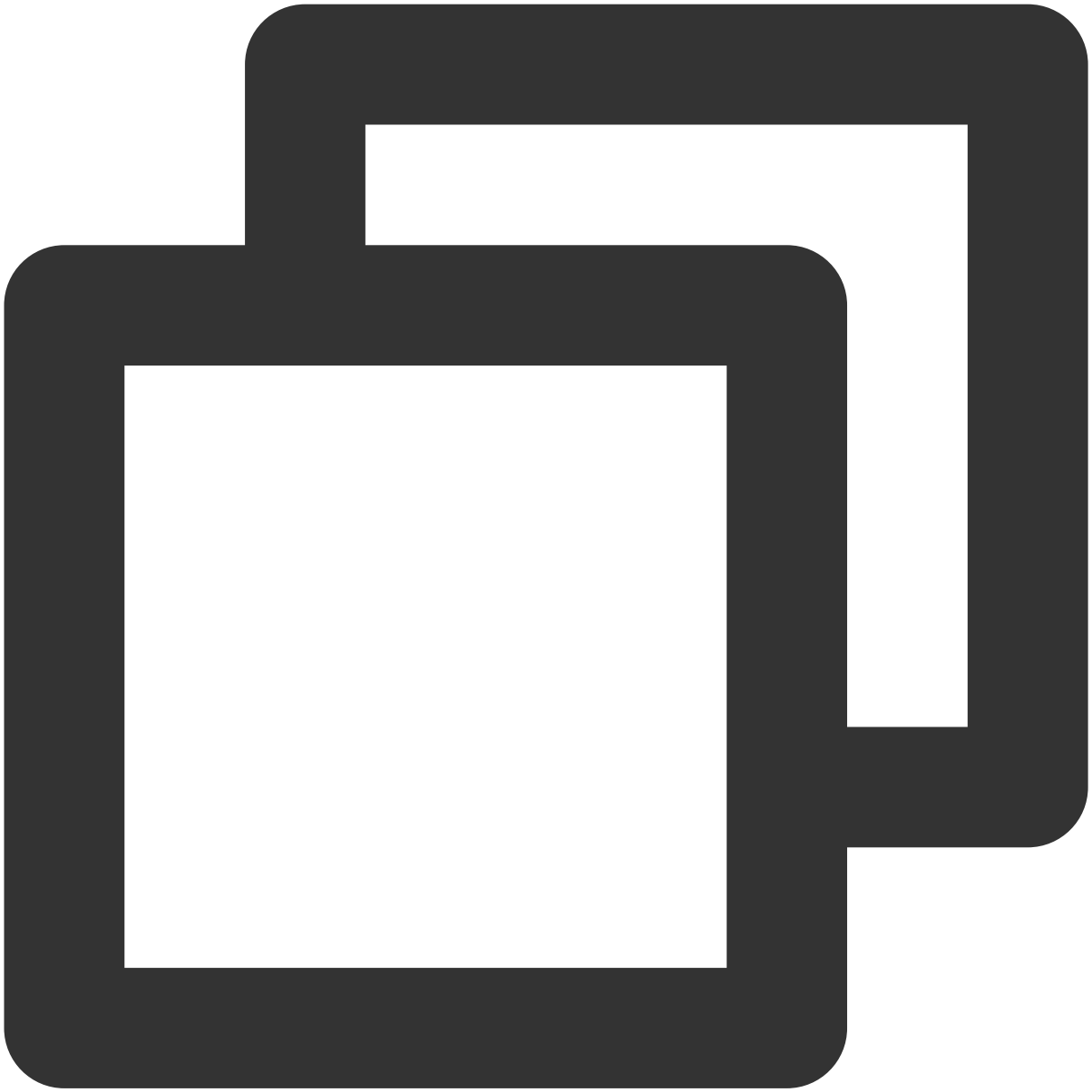
number x, the x-coordinate position of the top left corner of the drawn text;

number y, the y-coordinate position of the top left corner of the drawn text;

number maxWidth, the maximum width that needs to be drawn, optional.

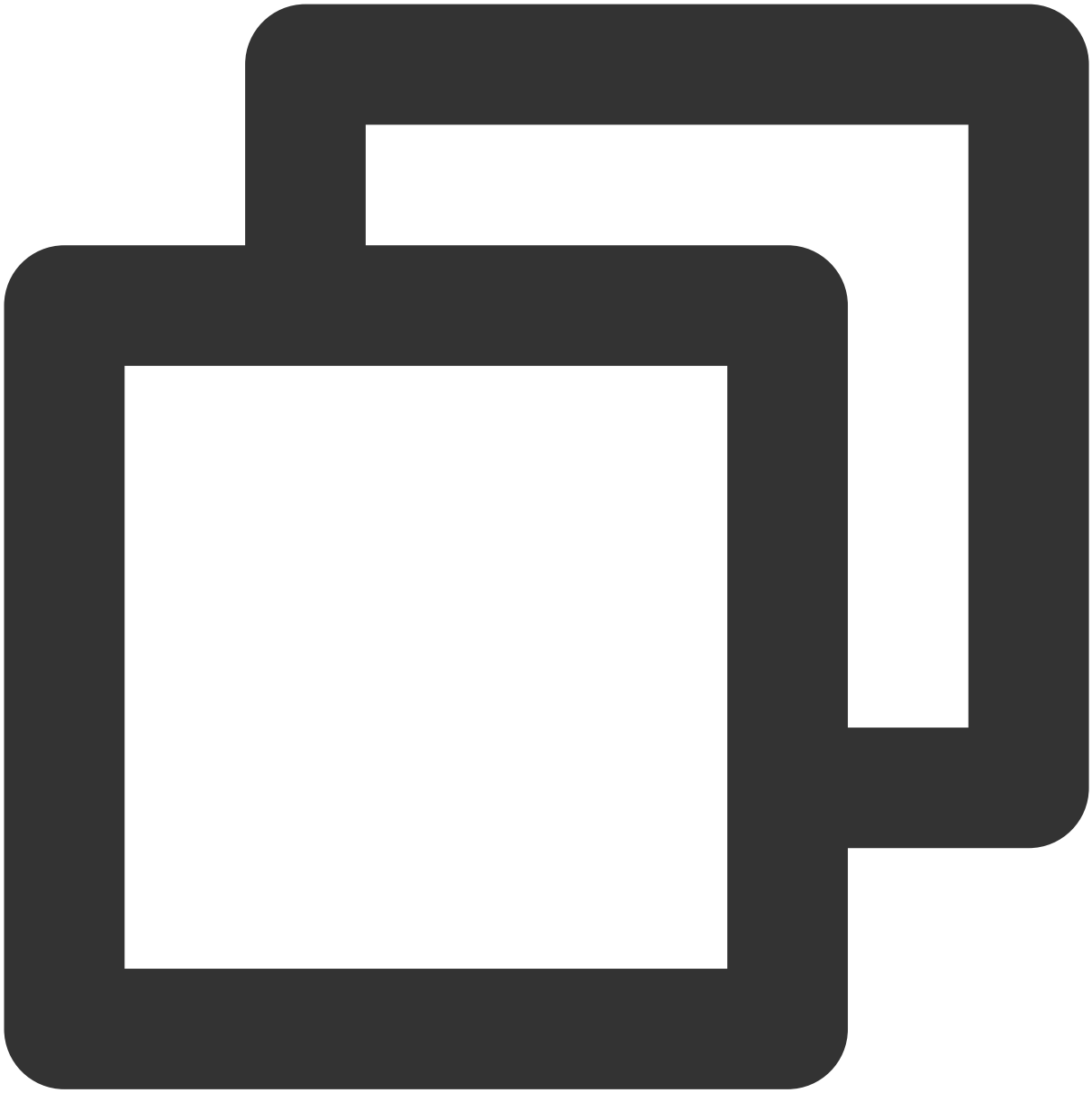
Sample Code:

©2013-2022 Tencent Cloud. All rights reserved.



```
const ctx = wx.createCanvasContext('myCanvas')
```

©2013-2022 Tencent Cloud. All rights reserved.



```
const ctx = wx.createCanvasContext('myCanvas')

ctx.setFontSize(20)
ctx.fillText('Hello', 20, 20)
ctx.fillText('MINA', 100, 100)

ctx.draw()
```

©2013-2022 Tencent Cloud. All rights reserved.



scale

The method is used as follows: CanvasContext.scale(number scaleWidth, number scaleHeight)

Function Description: After calling, the x and y coordinates of the path created thereafter will be scaled. Multiple calls will multiply the factors.

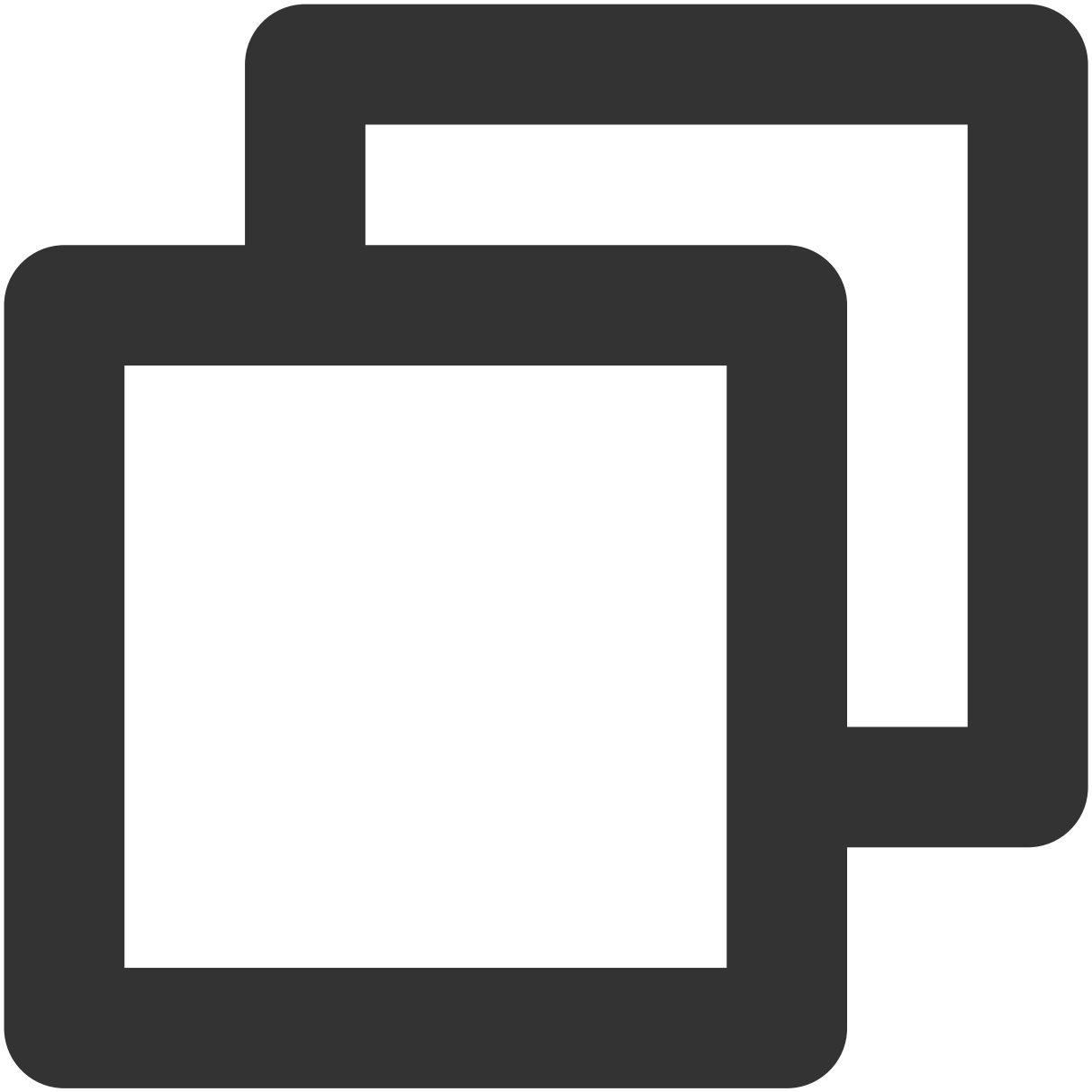
Parameters and Description:

number scaleWidth, the scaling factor for the x-coordinate (1 = 100%, 0.5 = 50%, 2 = 200%);

number scaleHeight, the scaling factor for the y-coordinate (1 = 100%, 0.5 = 50%, 2 = 200%).

Sample Code:

©2013-2022 Tencent Cloud. All rights reserved.

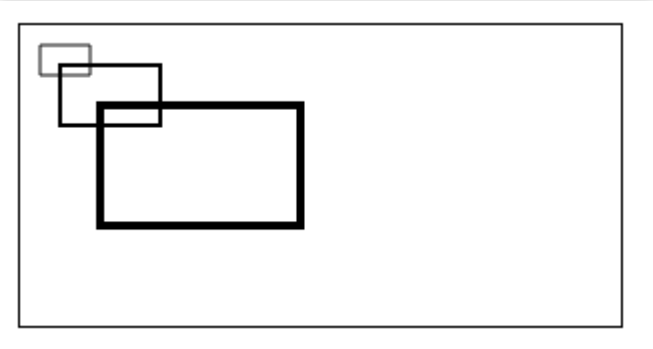


```
const ctx = wx.createCanvasContext('myCanvas')

ctx.strokeRect(10, 10, 25, 15)
ctx.scale(2, 2)
ctx.strokeRect(10, 10, 25, 15)
ctx.scale(2, 2)
ctx.strokeRect(10, 10, 25, 15)

ctx.draw()
```

©2013-2022 Tencent Cloud. All rights reserved.

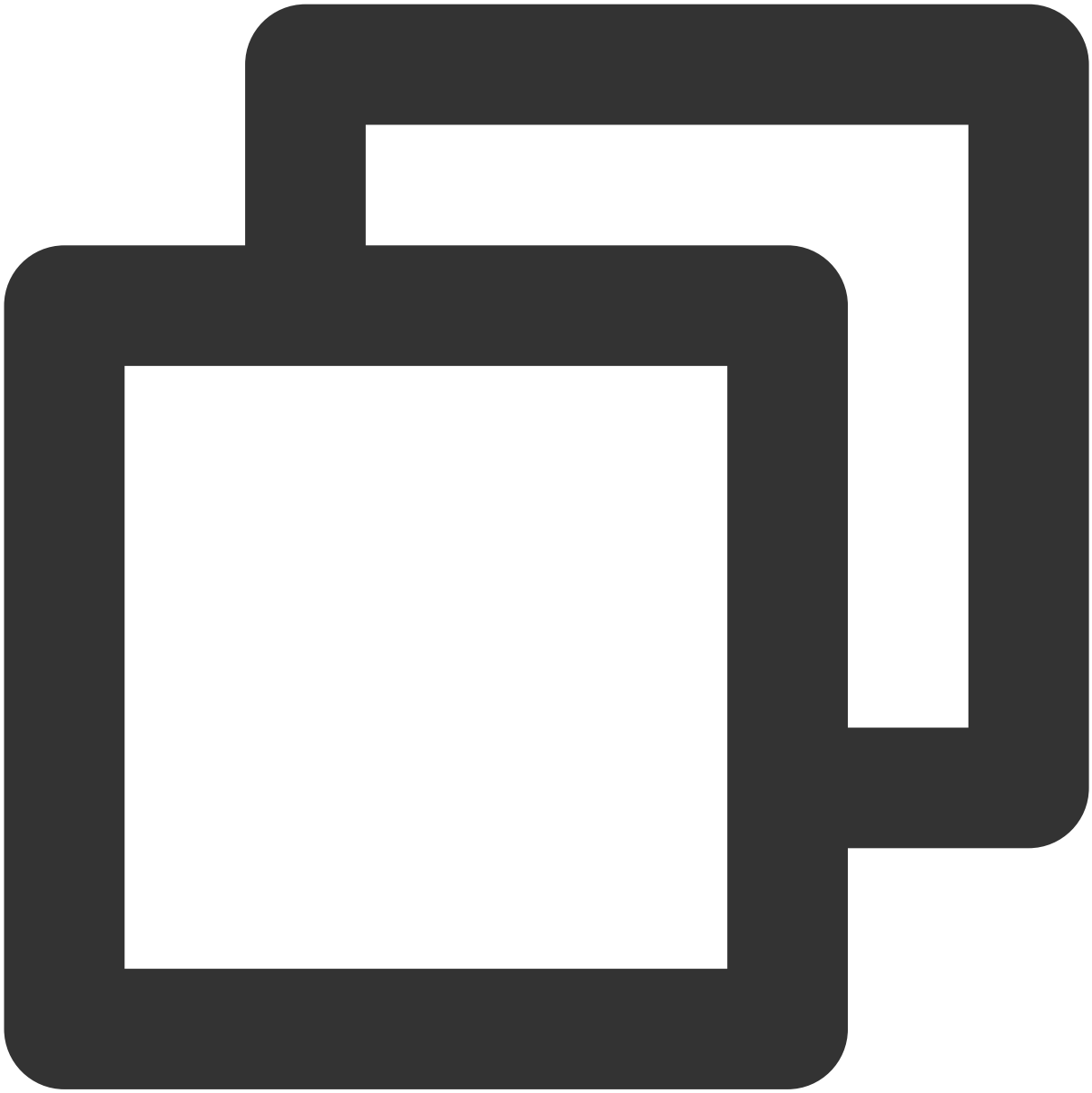


rotate

The method is used as follows: CanvasContext.rotate(number rotate)

Function Description: Rotates the current coordinate axis clockwise with the origin as the center. Multiple calls will accumulate the rotation angle. The origin can be modified using the translate method.
Parameters and Description: number rotate, rotation angle, calculated in radians as degrees * Math.PI/180; degrees range from 0-360.
Sample Code:

©2013-2022 Tencent Cloud. All rights reserved.

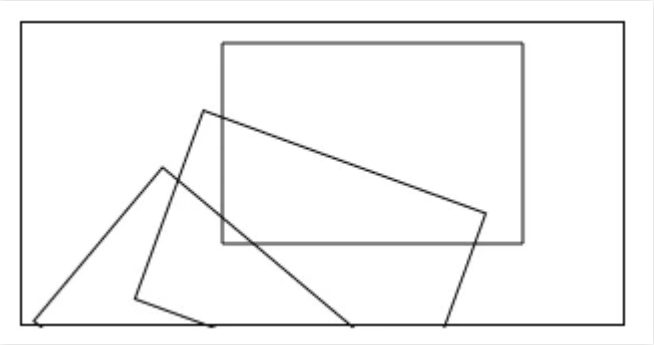


```
const ctx = wx.createCanvasContext('myCanvas')

ctx.strokeRect(100, 10, 150, 100)
ctx.rotate(20 * Math.PI / 180)
ctx.strokeRect(100, 10, 150, 100)
ctx.rotate(20 * Math.PI / 180)
ctx.strokeRect(100, 10, 150, 100)

ctx.draw()
```

©2013-2022 Tencent Cloud. All rights reserved.



transform

The method is used as follows: CanvasContext.transform(number scaleX, number skewX, number skewY, number scaleY, number translateX, number translateY)

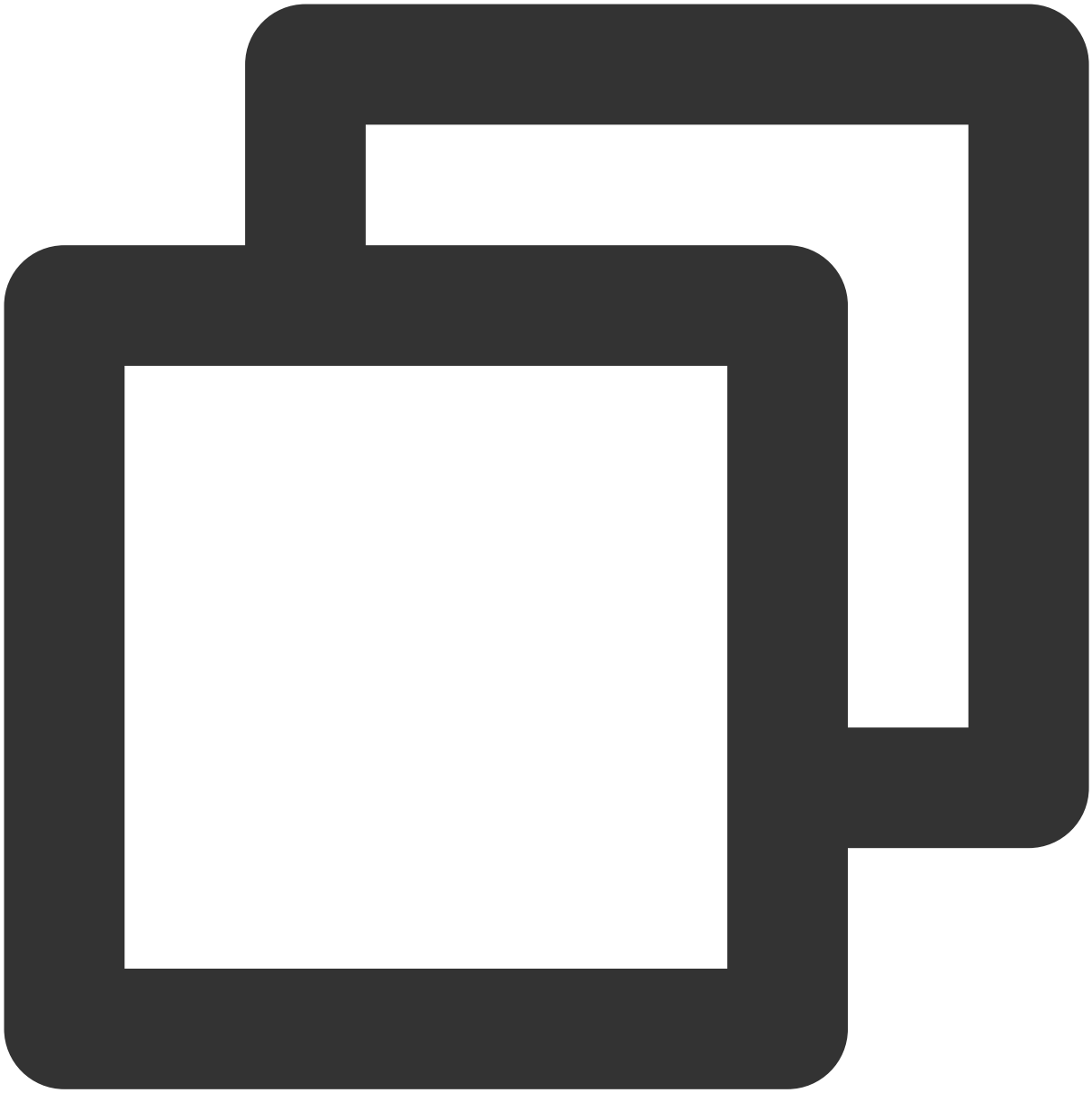
Function Description: A method to repeatedly overlay the current transformation using a matrix.
Parameters and Description:
number scaleX, horizontal scaling;
number scaleY, vertical scaling;
number skewX, horizontal skewing;
number skewY, vertical skewing;
number translateX, horizontal translation;
number translateY, vertical translation.

translate

The method is used as follows: CanvasContext.translate(number x, number y)

Function Description: Transforms the origin (0, 0) of the current coordinate system. The default coordinate system origin is the top left corner of the page.
Parameters and Description:
number x, horizontal coordinate translation;
number y, vertical coordinate translation.
Sample Code:

©2013-2022 Tencent Cloud. All rights reserved.

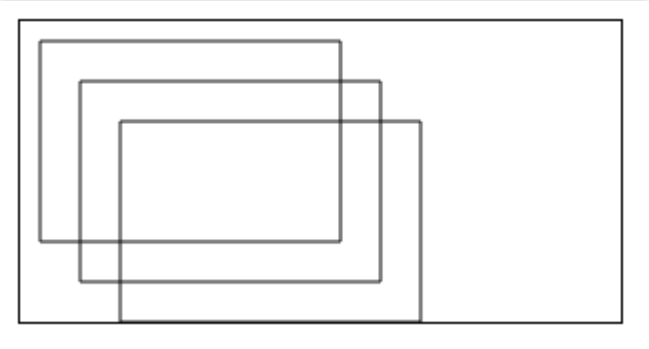


```
const ctx = wx.createCanvasContext('myCanvas')

ctx.strokeRect(10, 10, 150, 100)
ctx.translate(20, 20)
ctx.strokeRect(10, 10, 150, 100)
ctx.translate(20, 20)
ctx.strokeRect(10, 10, 150, 100)

ctx.draw()
```

©2013-2022 Tencent Cloud. All rights reserved.



setTransform

The method is used as follows: CanvasContext.setTransform(number scaleX, number skewX, number skewY, number scaleY, number translateX, number translateY)

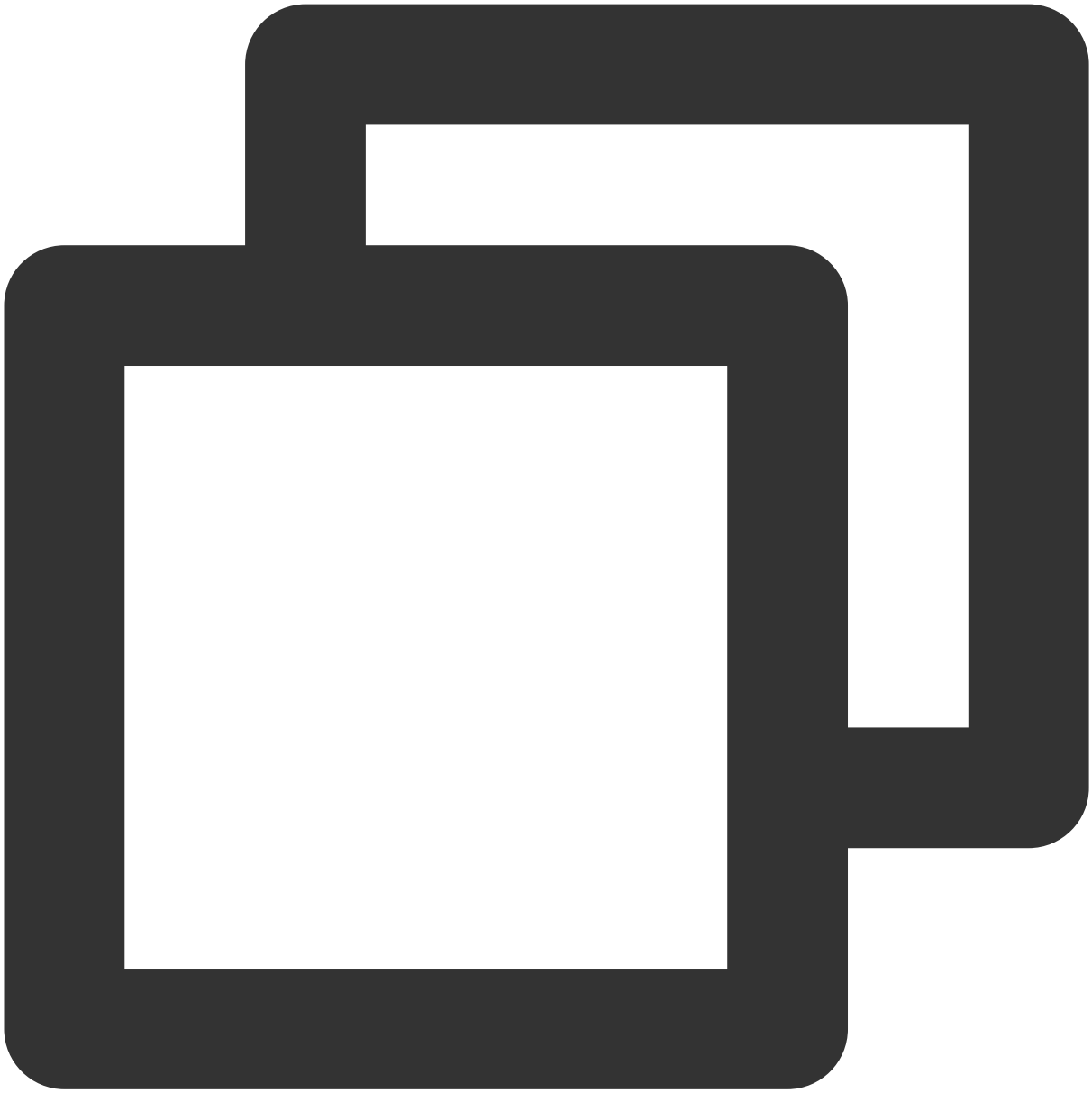
Function Description: A method to reset (overwrite) the current transformation using a matrix.
Parameters and Description:
number scaleX, horizontal scaling;
number scaleY, vertical scaling;
number skewX, horizontal skewing;
number skewY, vertical skewing;
number translateX, horizontal translation;
number translateY, vertical translation.

setFillStyle

The method is used as follows: CanvasContext.setFillStyle(string|CanvasGradient color)

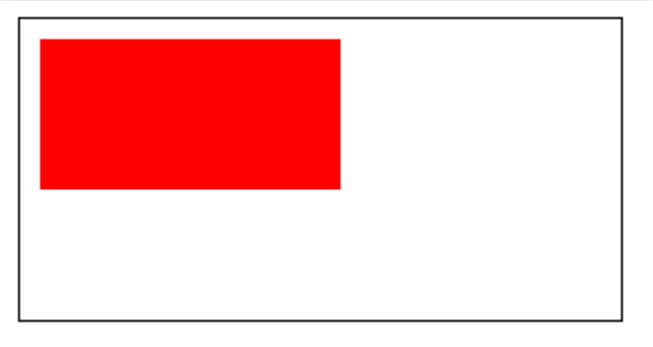
Function Description: Set the fill color.
Parameters and Description: string CanvasGradient color, the fill color, with the default color being black.
Sample Code:

©2013-2022 Tencent Cloud. All rights reserved.



```
const ctx = wx.createCanvasContext('myCanvas')
ctx.setFillStyle('red')
ctx.fillRect(10, 10, 150, 75)
ctx.draw()
```

©2013-2022 Tencent Cloud. All rights reserved.



setStrokeStyle

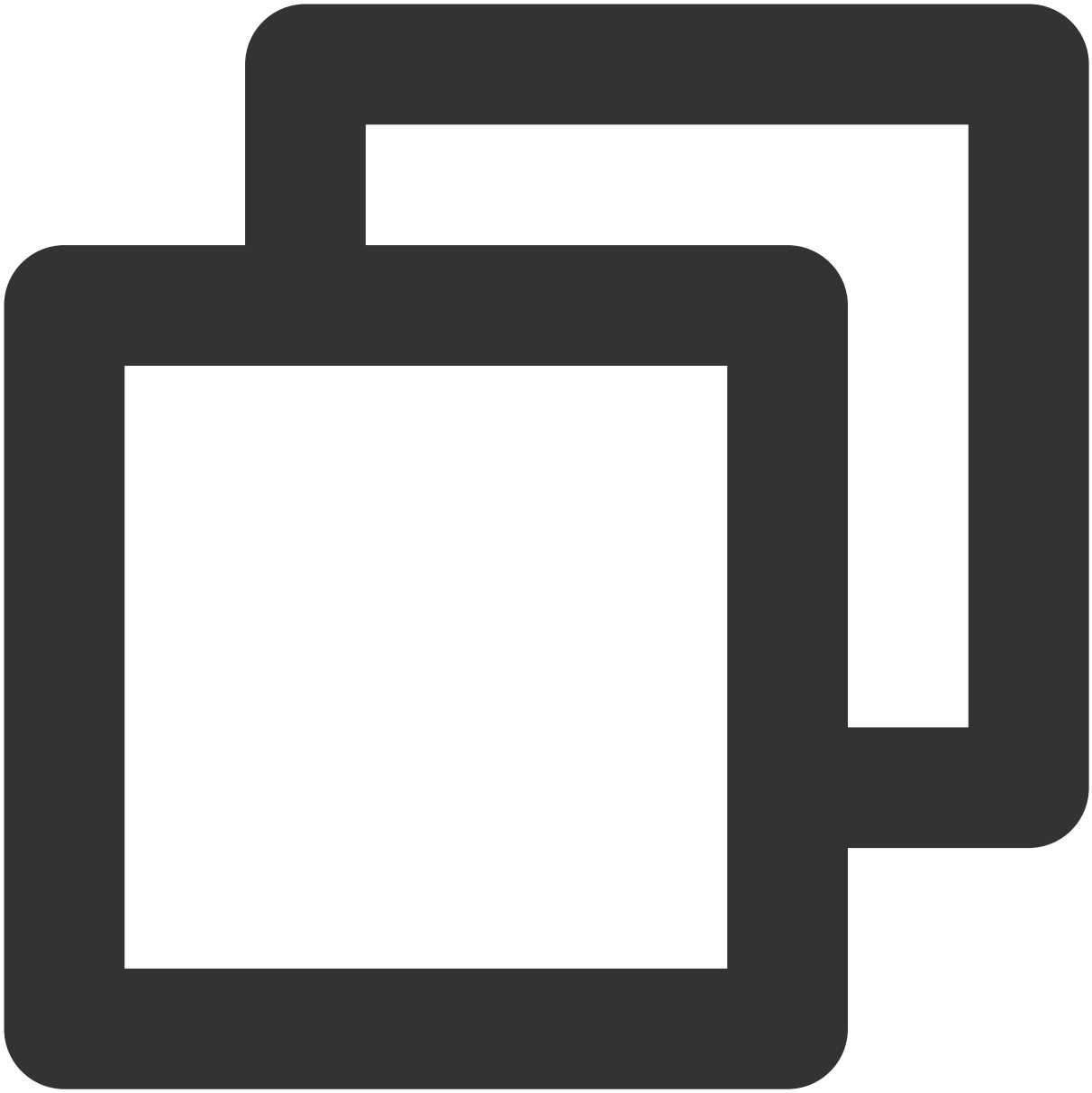
The method is used as follows: CanvasContext.setStrokeStyle(string|CanvasGradient color)

Function Description: Set the stroke color.

Parameters and Description: string CanvasGradient color, the fill color, with the default color being black.

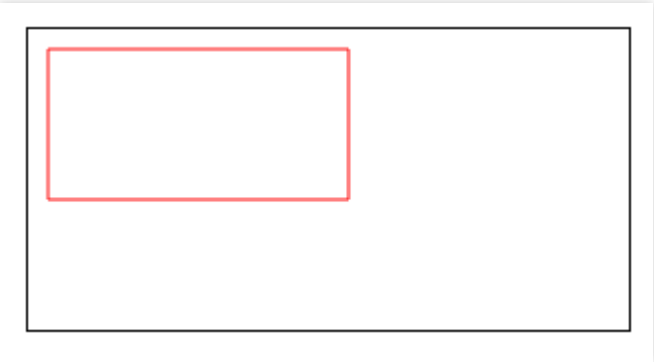
Sample Code:

©2013-2022 Tencent Cloud. All rights reserved.



```
const ctx = wx.createCanvasContext('myCanvas')
ctx.setStrokeStyle('red')
ctx.strokeRect(10, 10, 150, 75)
ctx.draw()
```

©2013-2022 Tencent Cloud. All rights reserved.

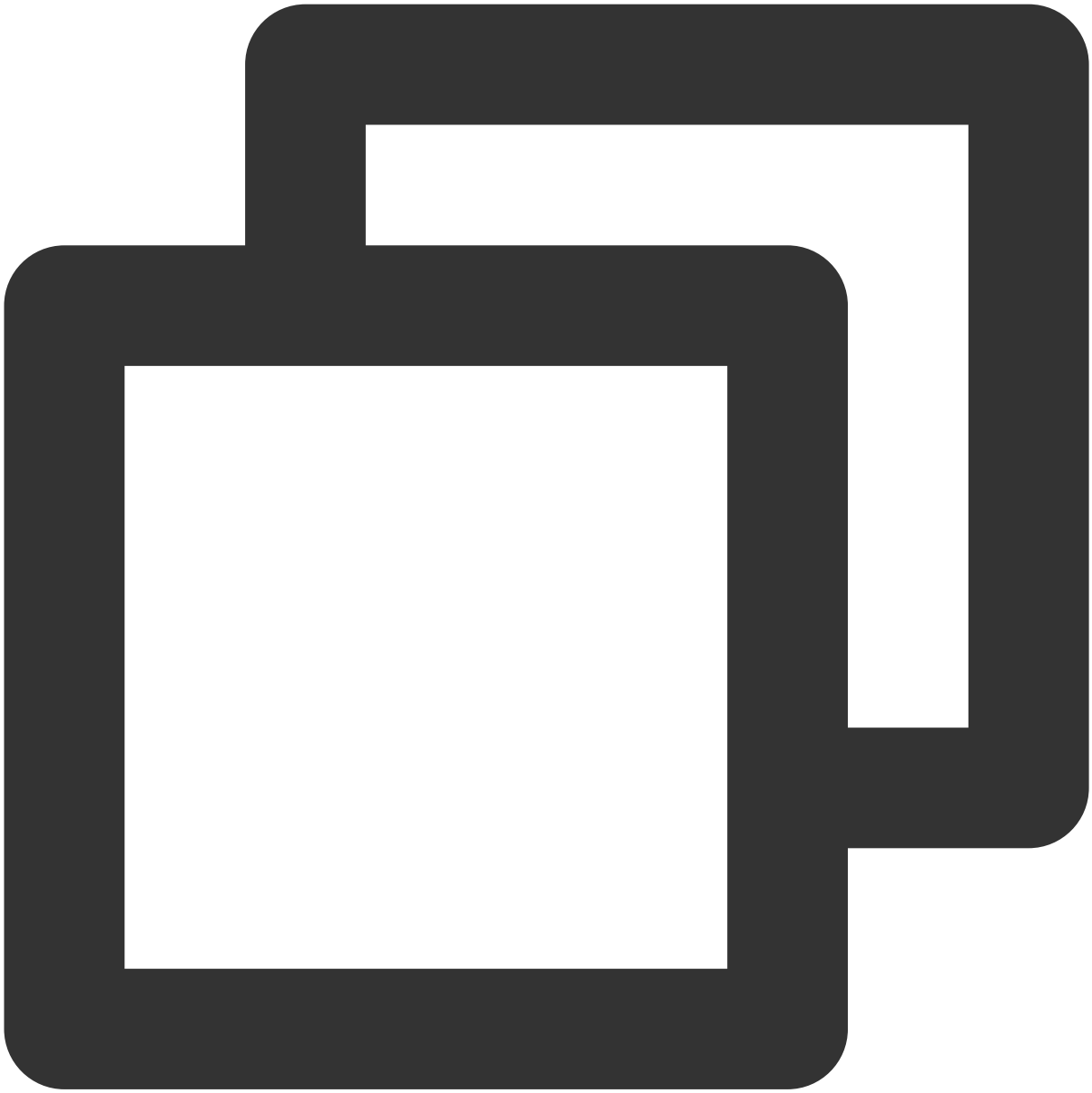


setShadow

The method is used as follows: CanvasContext.setShadow(number offsetX, number offsetY, number blur, string color)

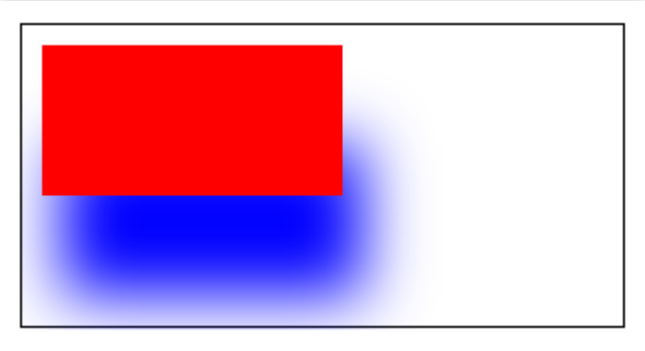
Function Description: Set the shadow style.
Parameters and Description:
number offsetX, the horizontal offset of the shadow relative to the shape, default value is 0;
number offsetY, the vertical offset of the shadow relative to the shape, default value is 0;
number blur, the blur level of the shadow, the larger the value, the more blurred it is. Range 0-100, default value is 0;
string color, the color of the shadow. Default value is black.
Sample Code:

©2013-2022 Tencent Cloud. All rights reserved.



```
const ctx = wx.createCanvasContext('myCanvas')
ctx.setFillStyle('red')
ctx.setShadow(10, 50, 50, 'blue')
ctx.fillRect(10, 10, 150, 75)
ctx.draw()
```

©2013-2022 Tencent Cloud. All rights reserved.



setGlobalAlpha

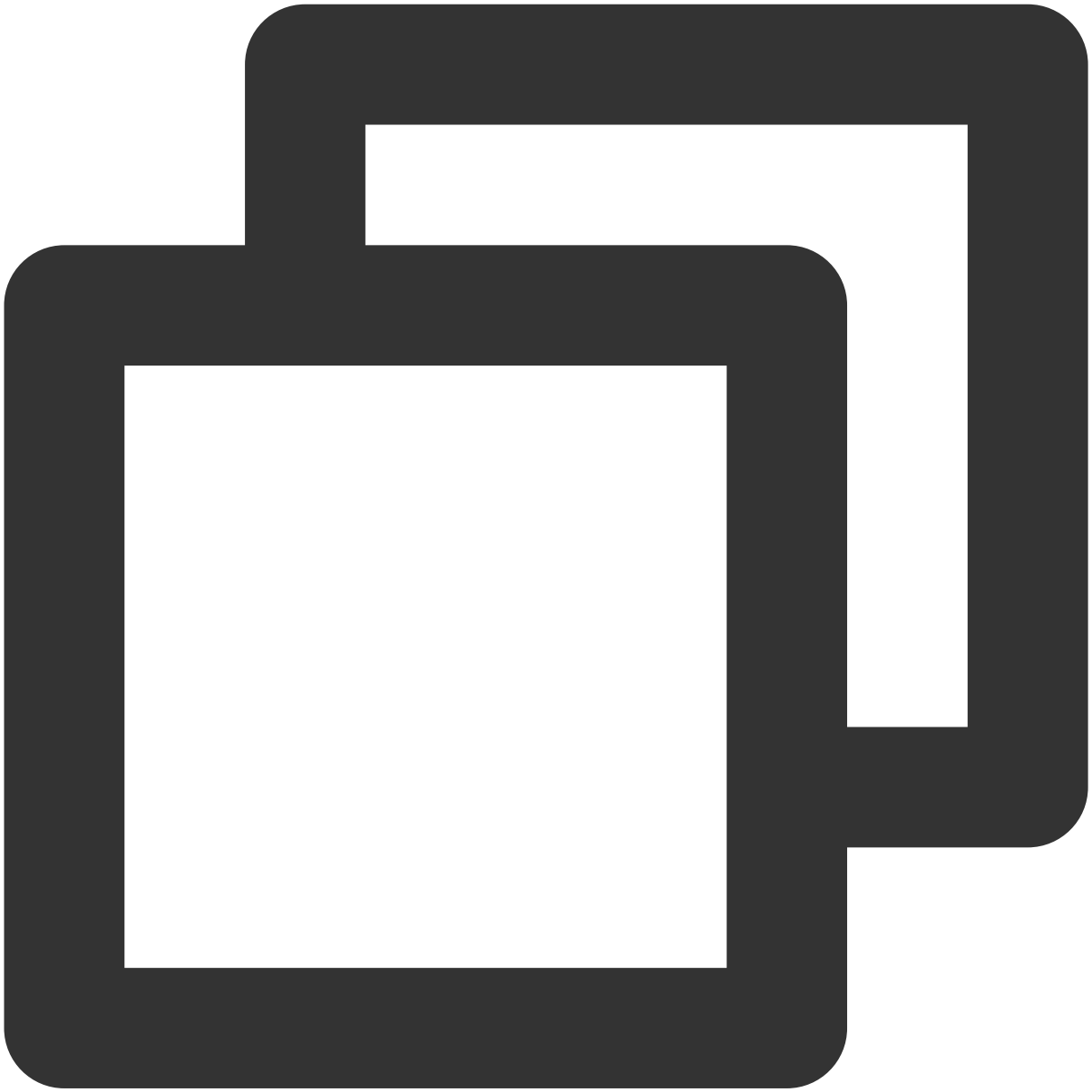
The method is used as follows: CanvasContext.setGlobalAlpha(number alpha)

Function Description: Set the global brush transparency.

Parameters and Description: number alpha, transparency. Range 0-1, 0 indicates complete transparency, 1 indicates complete opacity.

Sample Code:

©2013-2022 Tencent Cloud. All rights reserved.

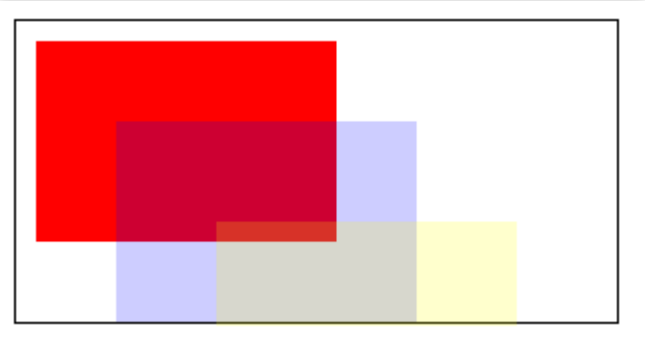


```
const ctx = wx.createCanvasContext('myCanvas')

ctx.setFillStyle('red')
ctx.fillRect(10, 10, 150, 100)
ctx.setGlobalAlpha(0.2)
ctx.setFillStyle('blue')
ctx.fillRect(50, 50, 150, 100)
ctx.setFillStyle('yellow')
ctx.fillRect(100, 100, 150, 100)

ctx.draw()
```

©2013-2022 Tencent Cloud. All rights reserved.



setLineWidth

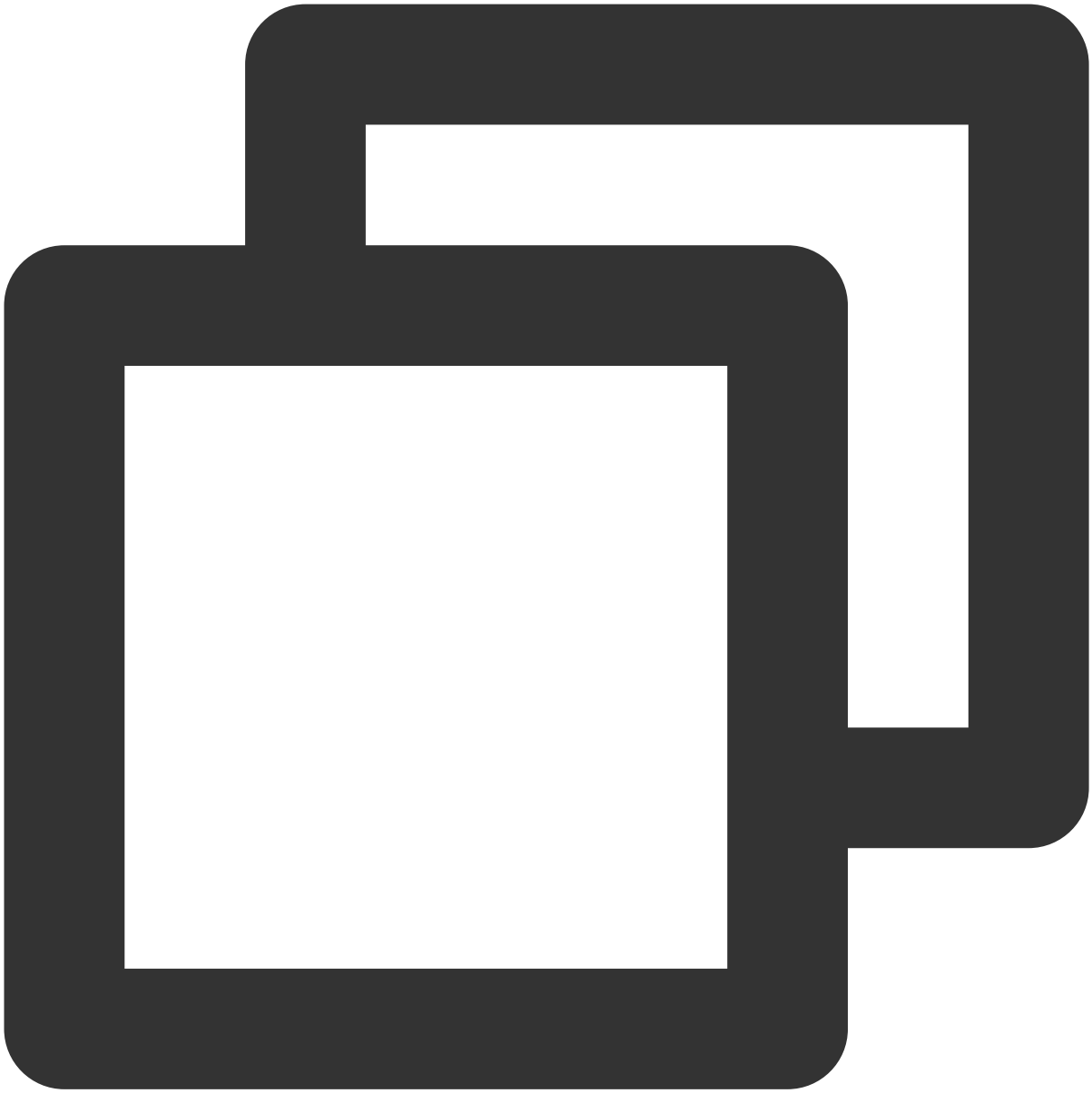
The method is used as follows: CanvasContext.setLineWidth(number lineWidth)

Function Description: Set the width of the line.

Parameters and Description: number lineWidth, the width of the line, unit: px.

Sample Code:

©2013-2022 Tencent Cloud. All rights reserved.



```
const ctx = wx.createCanvasContext('myCanvas')
ctx.beginPath()
ctx.moveTo(10, 10)
ctx.lineTo(150, 10)
ctx.stroke()

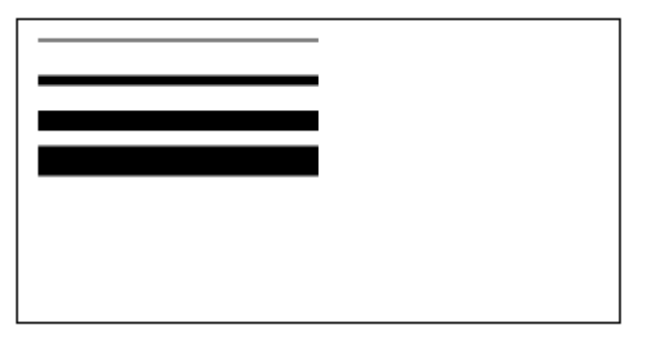
ctx.beginPath()
ctx.setLineWidth(5)
ctx.moveTo(10, 30)
ctx.lineTo(150, 30)
ctx.stroke()
```

©2013-2022 Tencent Cloud. All rights reserved.

```
ctx.beginPath()
ctx.setLineWidth(10)
ctx.moveTo(10, 50)
ctx.lineTo(150, 50)
ctx.stroke()

ctx.beginPath()
ctx.setLineWidth(15)
ctx.moveTo(10, 70)
ctx.lineTo(150, 70)
ctx.stroke()

ctx.draw()
```



setLineJoin

The method is used as follows: CanvasContext.setLineJoin(string lineJoin)

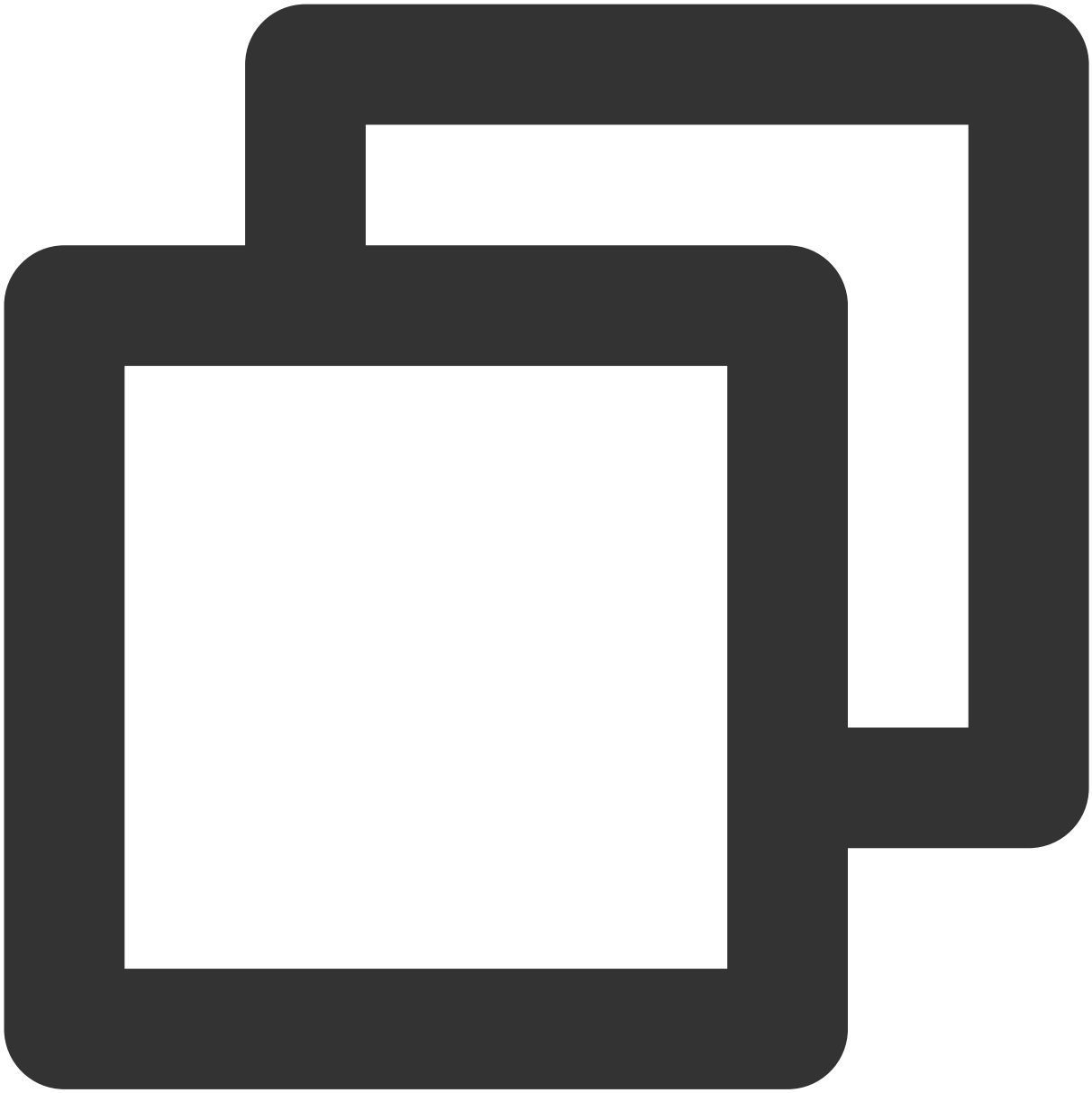
Function Description: Set the style of the line intersection.

Parameters and Description: string lineJoin, the style of the line's end intersection.

Value	Note
bevel	Bevel
round	Rounded corner
miter	Sharp corner

Sample Code:

©2013-2022 Tencent Cloud. All rights reserved.



```
const ctx = wx.createCanvasContext('myCanvas')
ctx.beginPath()
ctx.moveTo(10, 10)
ctx.lineTo(100, 50)
ctx.lineTo(10, 90)
ctx.stroke()

ctx.beginPath()
ctx.setLineJoin('bevel')
ctx.setLineWidth(10)
ctx.moveTo(50, 10)
```

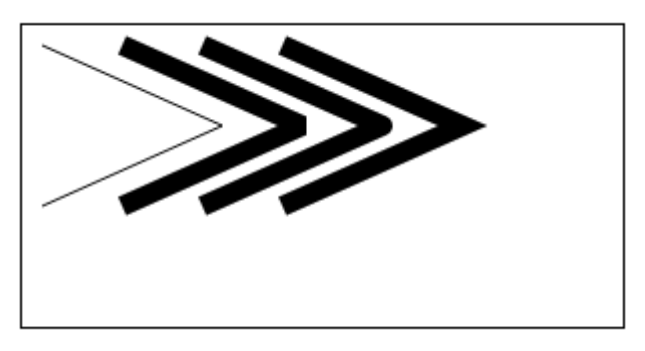
©2013-2022 Tencent Cloud. All rights reserved.

```
ctx.lineTo(140, 50)
ctx.lineTo(50, 90)
ctx.stroke()

ctx.beginPath()
ctx.setLineJoin('round')
ctx.setLineWidth(10)
ctx.moveTo(90, 10)
ctx.lineTo(180, 50)
ctx.lineTo(90, 90)
ctx.stroke()

ctx.beginPath()
ctx.setLineJoin('miter')
ctx.setLineWidth(10)
ctx.moveTo(130, 10)
ctx.lineTo(220, 50)
ctx.lineTo(130, 90)
ctx.stroke()

ctx.draw()
```



setLineCap

The method is used as follows: CanvasContext.setLineCap(string lineCap)

Function Description: Sets the style of the line's endpoint.

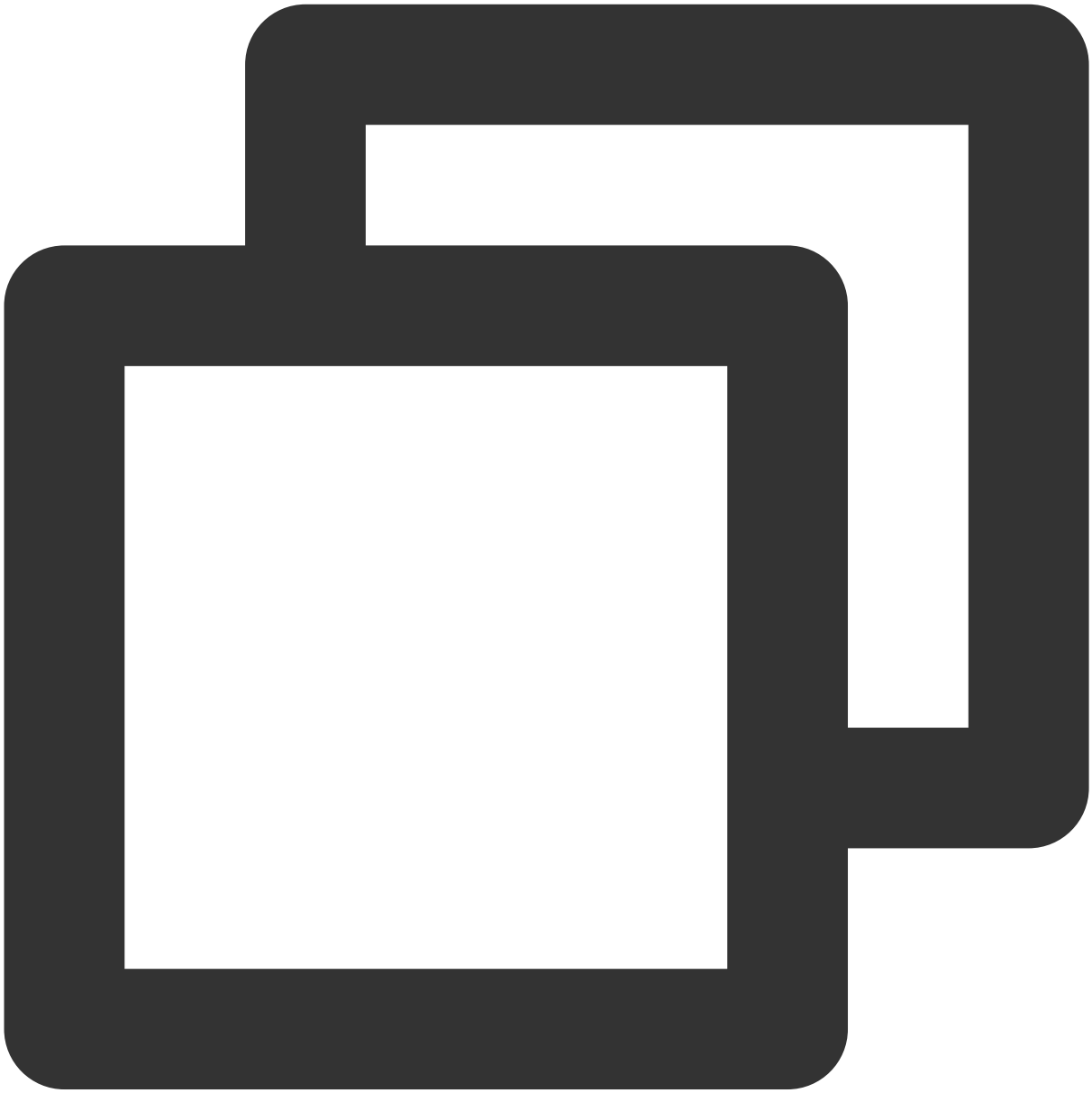
Parameters and Description: string lineGap, the style of the line's termination junction.

Value	Note

©2013-2022 Tencent Cloud. All rights reserved.

butt	Adding flat edges to each end of the line
round	Adding round caps to each end of the line
square	Adding square caps to each end of the line

Sample Code:



```
const ctx = wx.createCanvasContext('myCanvas')
ctx.beginPath()
ctx.moveTo(10, 10)
```

©2013-2022 Tencent Cloud. All rights reserved.

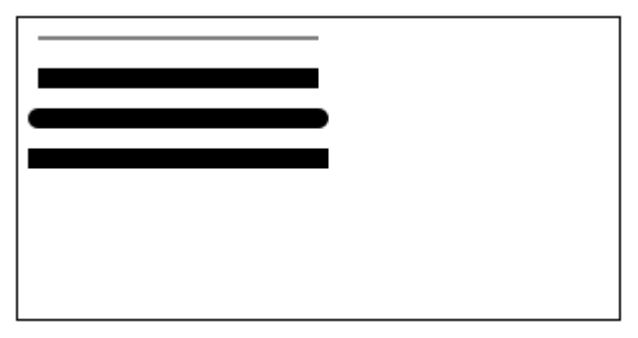
```
ctx.lineTo(150, 10)
ctx.stroke()

ctx.beginPath()
ctx.setLineCap('butt')
ctx.setLineWidth(10)
ctx.moveTo(10, 30)
ctx.lineTo(150, 30)
ctx.stroke()

ctx.beginPath()
ctx.setLineCap('round')
ctx.setLineWidth(10)
ctx.moveTo(10, 50)
ctx.lineTo(150, 50)
ctx.stroke()

ctx.beginPath()
ctx.setLineCap('square')
ctx.setLineWidth(10)
ctx.moveTo(10, 70)
ctx.lineTo(150, 70)
ctx.stroke()

ctx.draw()
```



setLineDash

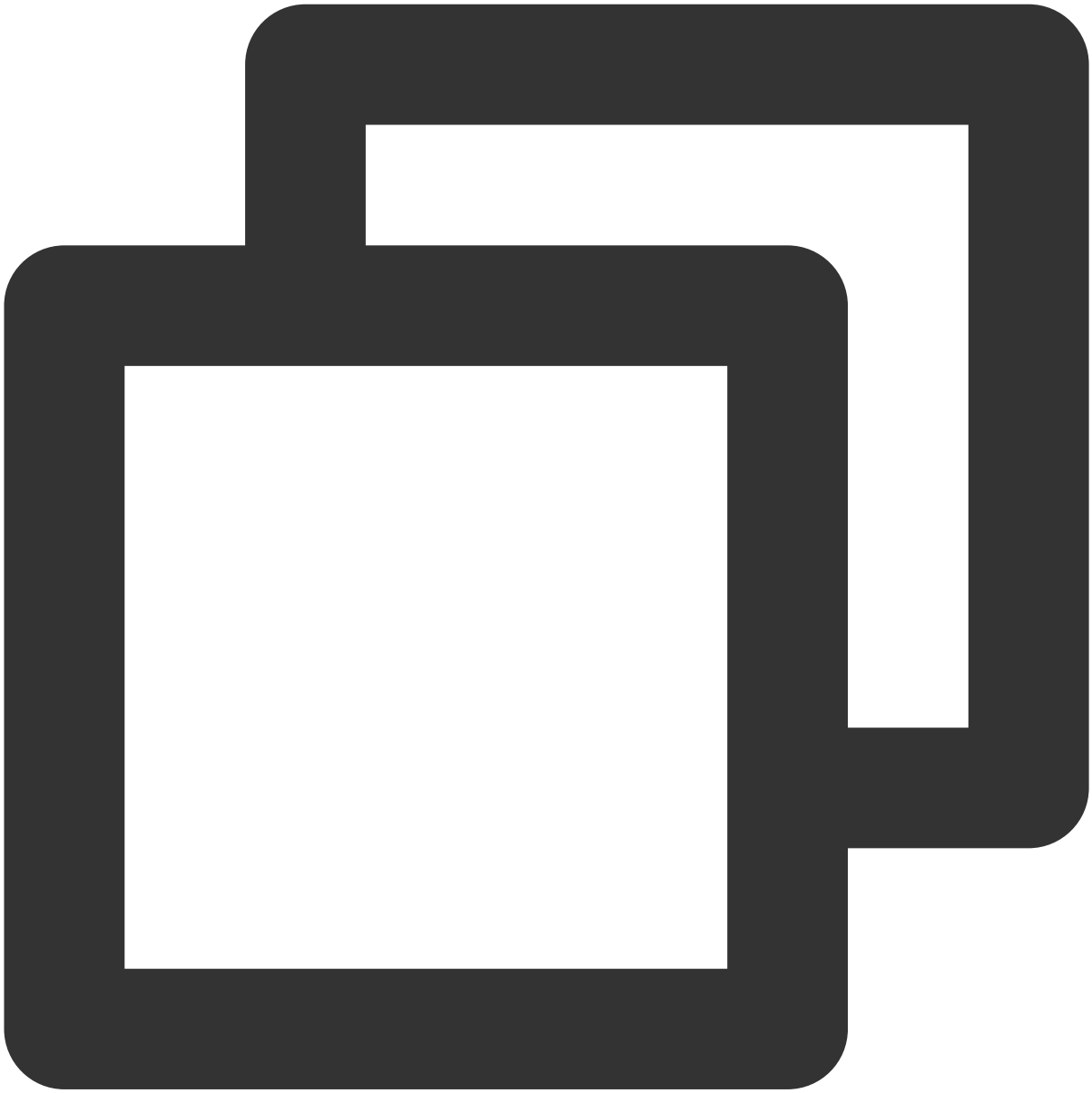
The method is used as follows: CanvasContext.setLineDash(Array.<number> pattern, number offset)

Function Description: Set the style of the dashed line.

Parameters and Description: Array.<number> pattern, a set of numbers describing the alternating lengths of the drawn segments and gaps (in coordinate space units); number offset, the offset of the dashed line.

Sample Code:

©2013-2022 Tencent Cloud. All rights reserved.



```
const ctx = wx.createCanvasContext('myCanvas')

ctx.setLineDash([10, 20], 5);

ctx.beginPath();
ctx.moveTo(0,100);
ctx.lineTo(400, 100);
ctx.stroke();

ctx.draw()
```

©2013-2022 Tencent Cloud. All rights reserved.

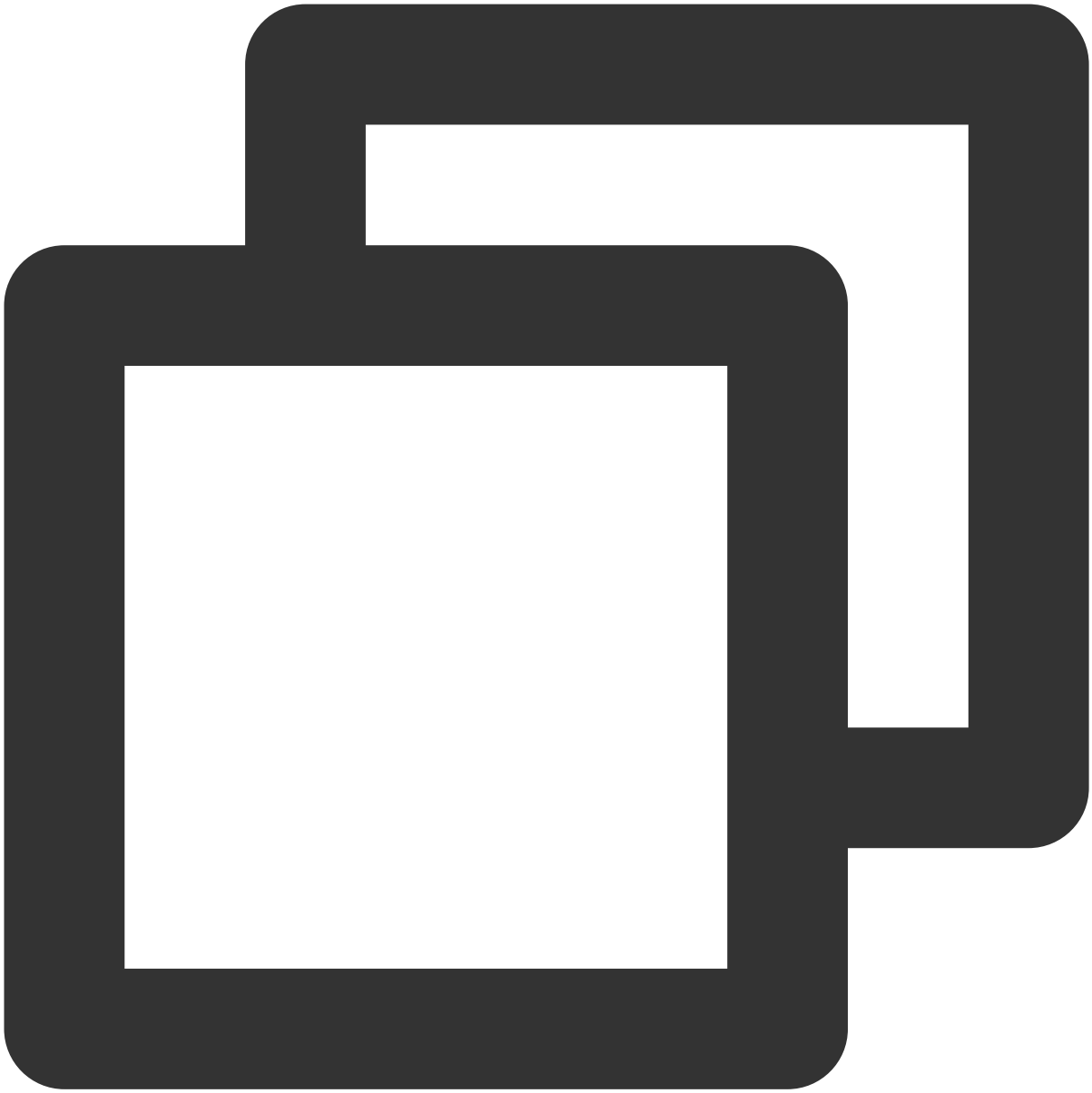


setMiterLimit

The method is used as follows: CanvasContext.setMiterLimit(number miterLimit)

Function Description: Set the maximum miter length. The miter length refers to the distance between the inner and outer angles at the intersection of two lines. It is only effective when CanvasContext.setLineJoin() is set to miter. If it exceeds the maximum miter length, the junction will be displayed as bevel in lineJoin.
Parameters and Description: number miterLimit, the maximum miter length.
Sample Code:

©2013-2022 Tencent Cloud. All rights reserved.



```
const ctx = wx.createCanvasContext('myCanvas')
ctx.beginPath()
ctx.setLineWidth(10)
ctx.setLineJoin('miter')
ctx.setMiterLimit(1)
ctx.moveTo(10, 10)
ctx.lineTo(100, 50)
ctx.lineTo(10, 90)
ctx.stroke()

ctx.beginPath()
```

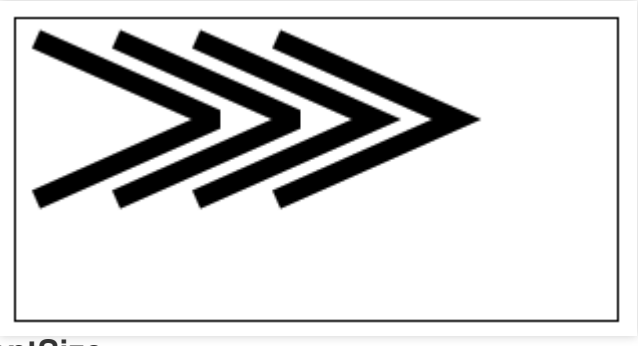
©2013-2022 Tencent Cloud. All rights reserved.

```
ctx.setLineWidth(10)
ctx.setLineJoin('miter')
ctx.setMiterLimit(2)
ctx.moveTo(50, 10)
ctx.lineTo(140, 50)
ctx.lineTo(50, 90)
ctx.stroke()

ctx.beginPath()
ctx.setLineWidth(10)
ctx.setLineJoin('miter')
ctx.setMiterLimit(3)
ctx.moveTo(90, 10)
ctx.lineTo(180, 50)
ctx.lineTo(90, 90)
ctx.stroke()

ctx.beginPath()
ctx.setLineWidth(10)
ctx.setLineJoin('miter')
ctx.setMiterLimit(4)
ctx.moveTo(130, 10)
ctx.lineTo(220, 50)
ctx.lineTo(130, 90)
ctx.stroke()

ctx.draw()
```



setFontSize

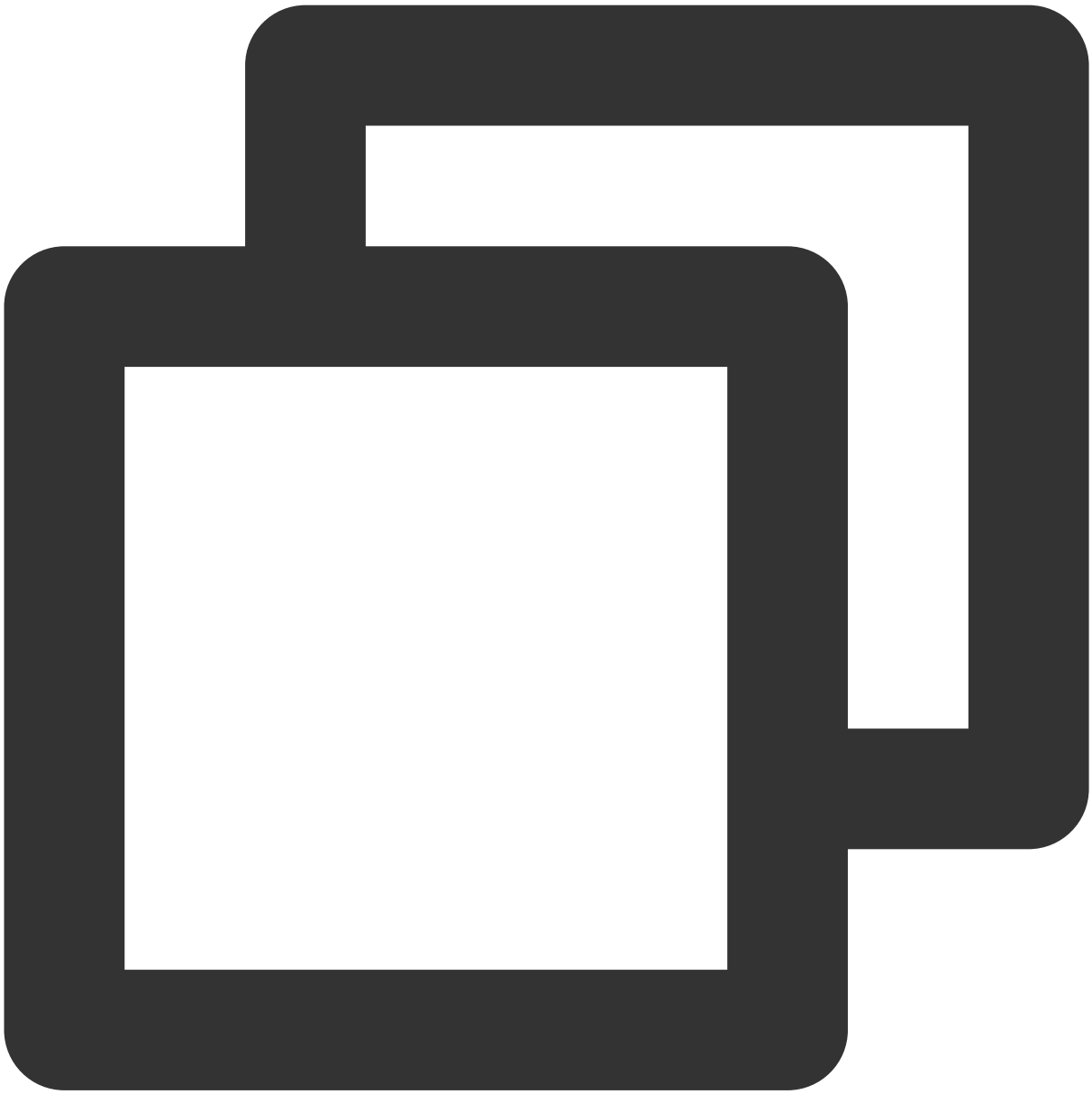
The method is used as follows: CanvasContext.setFontSize(number fontSize)

Function Description: Set the font size.

Parameters and Description: number fontSize, the size of the font.

©2013-2022 Tencent Cloud. All rights reserved.

Example Code



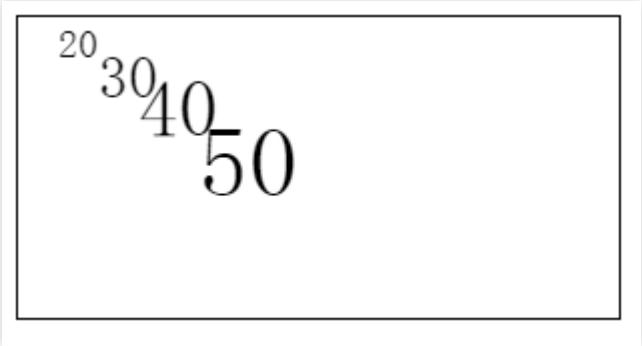
```
const ctx = wx.createCanvasContext('myCanvas')

ctx.setFontSize(20)
ctx.fillText('20', 20, 20)
ctx.setFontSize(30)
ctx.fillText('30', 40, 40)
ctx.setFontSize(40)
ctx.fillText('40', 60, 60)
ctx.setFontSize(50)
```

©2013-2022 Tencent Cloud. All rights reserved.

```
ctx.fillText('50', 90, 90)

ctx.draw()
```



setTextAlign

The method is used as follows: CanvasContext.setTextAlign(string align)

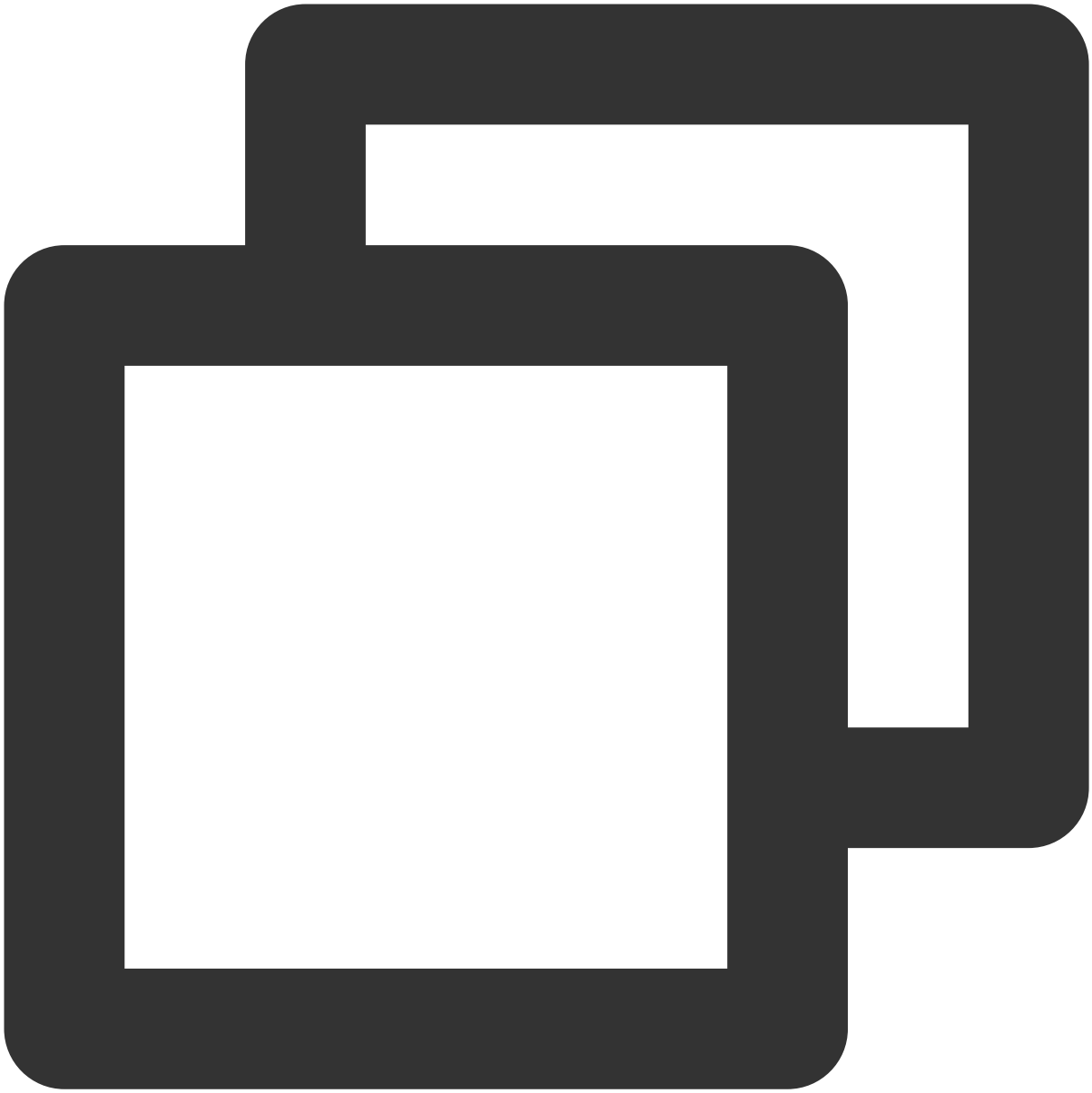
Function Description: Set the alignment of the text.

Parameters and Description: string align, the alignment of the text.

Value	Note
left	Align Left
center	Align Center
right	Align Right

Sample Code:

©2013-2022 Tencent Cloud. All rights reserved.



```
const ctx = wx.createCanvasContext('myCanvas')

ctx.setStrokeStyle('red')
ctx.moveTo(150, 20)
ctx.lineTo(150, 170)
ctx.stroke()

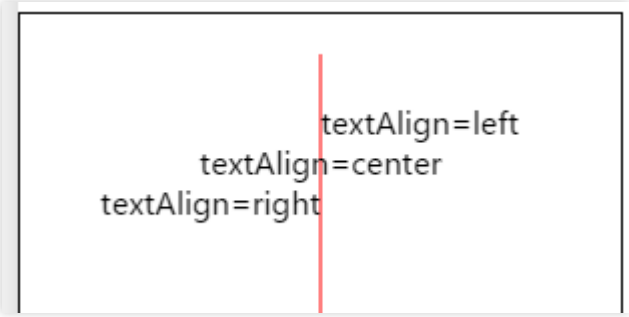
ctx.setFontSize(15)
ctx.setTextAlign('left')
ctx.fillText('textAlign=left', 150, 60)
```

©2013-2022 Tencent Cloud. All rights reserved.

```
ctx.setTextAlign('center')
ctx.fillText('textAlign=center', 150, 80)

ctx.setTextAlign('right')
ctx.fillText('textAlign=right', 150, 100)

ctx.draw()
```



setTextBaseline

The method is used as follows: CanvasContext.setTextBaseline(string textBaseline)

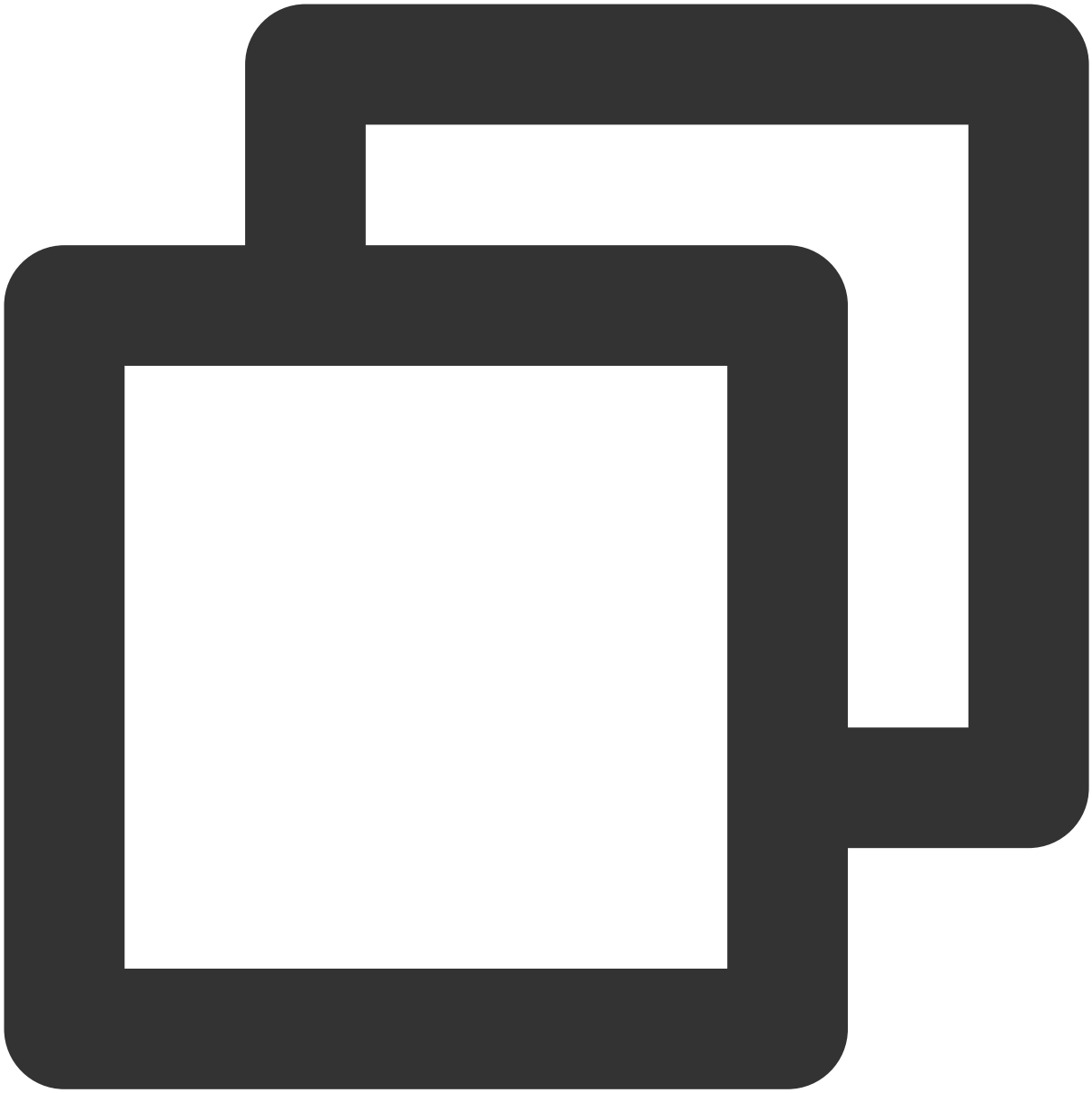
Function Description: Sets the vertical alignment of the text.

Parameters and Description: string textBaseline, the vertical alignment method of the text.

Value	Note
top	Align Top
bottom	Align Bottom
middle	Align Center
normal	Default Value

Sample Code:

©2013-2022 Tencent Cloud. All rights reserved.



```
const ctx = wx.createCanvasContext('myCanvas')

ctx.setStrokeStyle('red')
ctx.moveTo(5, 75)
ctx.lineTo(295, 75)
ctx.stroke()

ctx.setFontSize(20)

ctx.setTextBaseline('top')
ctx.fillText('top', 5, 75)
```

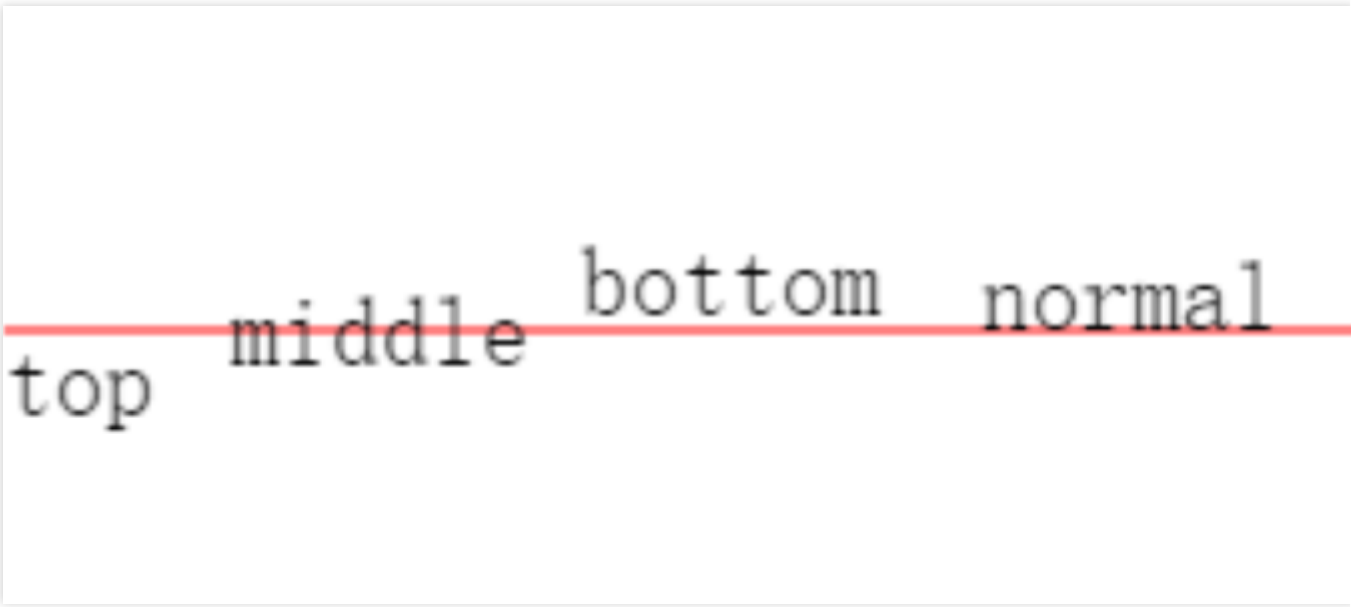
©2013-2022 Tencent Cloud. All rights reserved.

```
ctx.setTextBaseline('middle')
ctx.fillText('middle', 50, 75)

ctx.setTextBaseline('bottom')
ctx.fillText('bottom', 120, 75)

ctx.setTextBaseline('normal')
ctx.fillText('normal', 200, 75)

ctx.draw()
```



©2013-2022 Tencent Cloud. All rights reserved.

Color

Last updated：2024-07-12 19:23:08

Colors. The colors used in canvas can be represented in several ways:

RGB colors: e.g. 'rgb(255, 0, 0)'.

RGBA colors: e.g. 'rgba(255, 0, 0, 0, 0.3)'

hexadecimal colors: e.g. '#FF0000'

Predefined colors: e.g. 'red'

There are 148 predefined colors: Note*: Color Name is case insensitive.

Color Name	HEX
AliceBlue	#F0F8FF
AntiqueWhite	#FAEBD7
Aqua	#00FFFF
Aquamarine	#7FFFD4
Azure	#F0FFFF
Beige	#F5F5
Bisque	#FFE4C4
Black	#000000
BlanchedAlmond	#FFEBCD
Blue	#0000FF
BlueViolet	#8A2BE2
Brown	#A52A2A
BurlyWood	#DEB887
CadetBlue	#5F9EA0
Chartreuse	#7FFF00
Chocolate	#D2691E
Coral	#FF7F50

©2013-2022 Tencent Cloud. All rights reserved.

CornflowerBlue	#6495ED
Cornsilk	#FFF8DC
Crimson	#DC143C
Cyan	#00FFFF
DarkBlue	#00008B
DarkCyan	#008B8B
DarkGoldenRod	#B8860B
DarkGray	#A9A9A9
DarkGrey	#A9A9A9
DarkGreen	#006400
DarkKhaki	#BDB76B
DarkMagenta	#8B008B
DarkOliveGreen	#556B2F
DarkOrange	#FF8C00
DarkOrchid	#9932CC
DarkRed	#8B0000
DarkSalmon	#E9967A
DarkSeaGreen	#8FBC8F
DarkSlateBlue	#483D8B
DarkSlateGray	#2F4F4F
DarkSlateGrey	#2F4F4F
DarkTurquoise	#00CED1
DarkViolet	#9400D3
DeepPink	#FF1493
DeepSkyBlue	#00BFFF

©2013-2022 Tencent Cloud. All rights reserved.

DimGray	#696969
DimGrey	#696969
DodgerBlue	#1E90FF
FireBrick	#B22222
FloralWhite	#FFFAF0
ForestGreen	#228B22
Fuchsia	#FF00FF
Gainsboro	#DCDCDC
GhostWhite	#F8F8FF
Gold	#FFD700
GoldenRod	#DAA520
Gray	#808080
Grey	#808080
Green	#008000
GreenYellow	#ADFF2F
HoneyDew	#F0FFF0
HotPink	#FF69B4
IndianRed	#CD5C5C
Indigo	#4B0082
Ivory	#FFFFF0
Khaki	#F0E68C
Lavender	#E6E6FA
LavenderBlush	#FFF0F5
LawnGreen	#7CFC00
LemonChiffon	#FFFACD

©2013-2022 Tencent Cloud. All rights reserved.

LightBlue	#ADD8E6
LightCoral	#F08080
LightCyan	#E0FFFF
LightGoldenRodYellow	#FAFAD2
LightGray	#D3D3D3
LightGrey	#D3D3D3
LightGreen	#90EE90
LightPink	#FFB6C1
LightSalmon	#FFA07A
LightSeaGreen	#20B2AA
LightSkyBlue	#87CEFA
LightSlateGray	#778899
LightSlateGrey	#778899
LightSteelBlue	#B0C4DE
LightYellow	#FFFFE0
Lime	#00FF00
LimeGreen	#32CD32
Linen	#FAF0E6
Magenta	#FF00FF
Maroon	#800000
MediumAquaMarine	#66CDAA
MediumBlue	#0000CD
MediumOrchid	#BA55D3
MediumPurple	#9370DB
MediumSeaGreen	#3CB371

©2013-2022 Tencent Cloud. All rights reserved.

MediumSlateBlue	#7B68EE
MediumSpringGreen	#00FA9A
MediumTurquoise	#48D1CC
MediumVioletRed	#C71585
MidnightBlue	#191970
MintCream	#F5FFFA
MistyRose	#FFE4E1
Moccasin	#FFE4B5
NavajoWhite	#FFDEAD
Navy	#000080
OldLace	#FDF5E6
Olive	#808000
OliveDrab	#6B8E23
Orange	#FFA500
OrangeRed	#FF4500
Orchid	#DA70D6
PaleGoldenRod	#EEE8AA
PaleGreen	#98FB98
PaleTurquoise	#AFEEEE
PaleVioletRed	#DB7093
PapayaWhip	#FFEFD5
PeachPuff	#FFDAB9
Peru	#CD853F
Pink	#FFC0CB
Plum	#DDA0DD

©2013-2022 Tencent Cloud. All rights reserved.

PowderBlue	#B0E0E6
Purple	#800080
RebeccaPurple	#663399
Red	#FF0000
RosyBrown	#BC8F8F
RoyalBlue	#4169E1
SaddleBrown	#8B4513
Salmon	#FA8072
SandyBrown	#F4A460
SeaGreen	#2E8B57
SeaShell	#FFF5EE
Sienna	#A0522D
Silver	#C0C0C0
SkyBlue	#87CEEB
SlateBlue	#6A5ACD
SlateGray	#708090
SlateGrey	#708090
Snow	#FFFAFA
SpringGreen	#00FF7F
SteelBlue	#4682B4
Tan	#D2B48C
Teal	#008080
Thistle	#D8BFD8
Tomato	#FF6347
Turquoise	#40E0D0

©2013-2022 Tencent Cloud. All rights reserved.

Violet	#EE82EE
Wheat	#F5DEB3
White	#FFFFFF
WhiteSmoke	#F5F5F5
Yellow	#FFFF00
YellowGreen	#9ACD32

©2013-2022 Tencent Cloud. All rights reserved.

Worker

Last updated：2024-07-12 18:28:42

wx.CreateWorker

wx.createWorker(string scriptPath)

Function: Create a worker thread.

Parameters: string scriptPath, the absolute path of the worker entry file.

Return：Worker， Worker object

Notes:

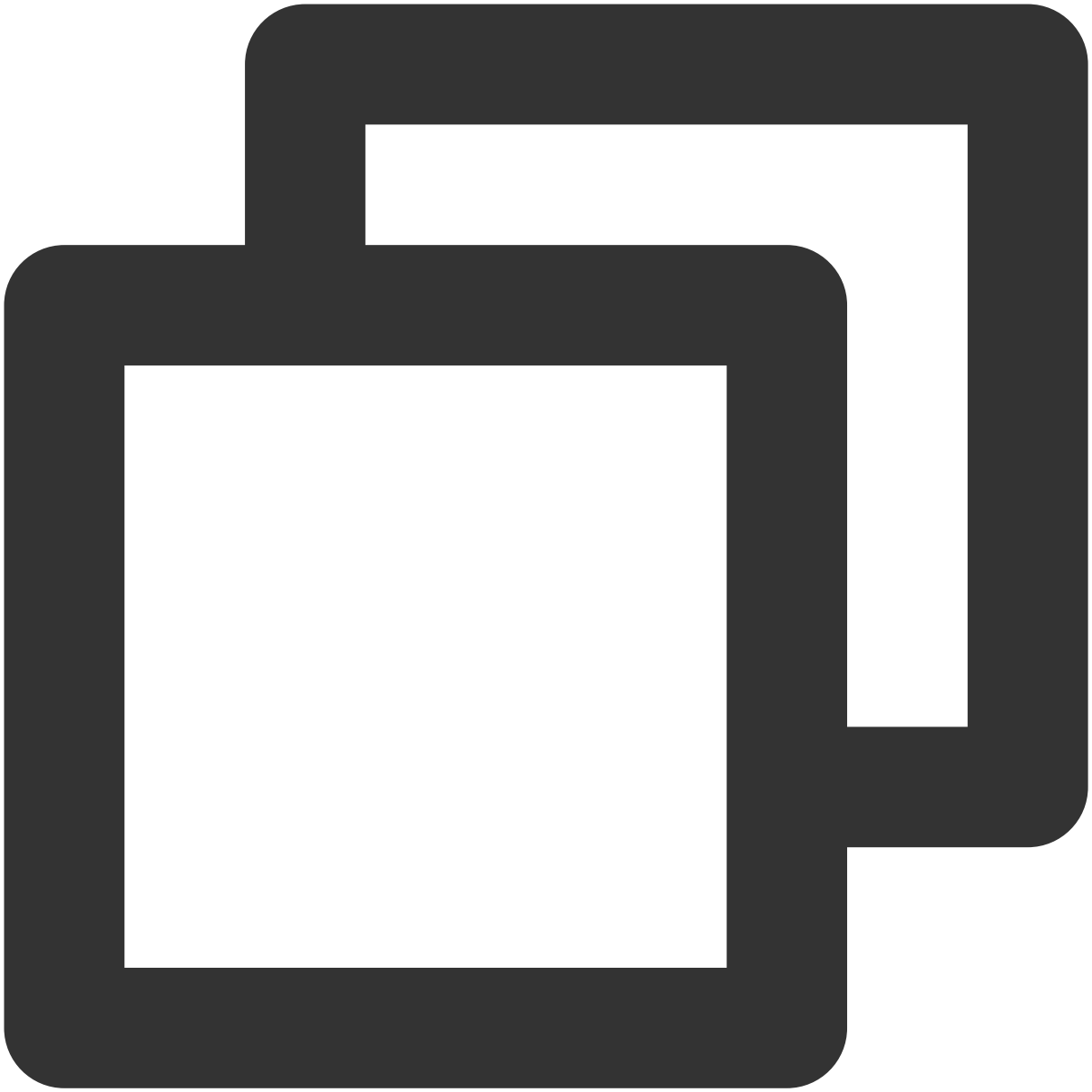
Before using the interface, you need to configure the workers field in app.json, it means the root directory of the worker code;

scriptPath is the absolute path of the entry file, and does not start with /;

Currently, only one worker can be created at most, please calll Worker.terminate before creating the next worker.

Sample code

©2013-2022 Tencent Cloud. All rights reserved.



```
// Create a regular worker
const worker = wx.createWorker('workers/index.js') // The filename specifies the pa


worker.onMessage(function (res) {
  console.log(res)
})


worker.postMessage({
  msg: 'hello worker'
```

©2013-2022 Tencent Cloud. All rights reserved.

```
})


worker.terminate()
```

Worker

Worker.onMessage

Worker.onMessage(function listener)

Feature Description：Listen to the event of the message sent by the main thread/worker thread to the current thread.

Parameter 1：function listener, the event listener for the message sent by the main thread/worker thread to the current thread.

Parameter 2：Object res

Attributes	Types	Description
message	Object	Message sent from the main/Worker thread to the current thread

Worker.postMessage

Worker.postMessage(Object message)

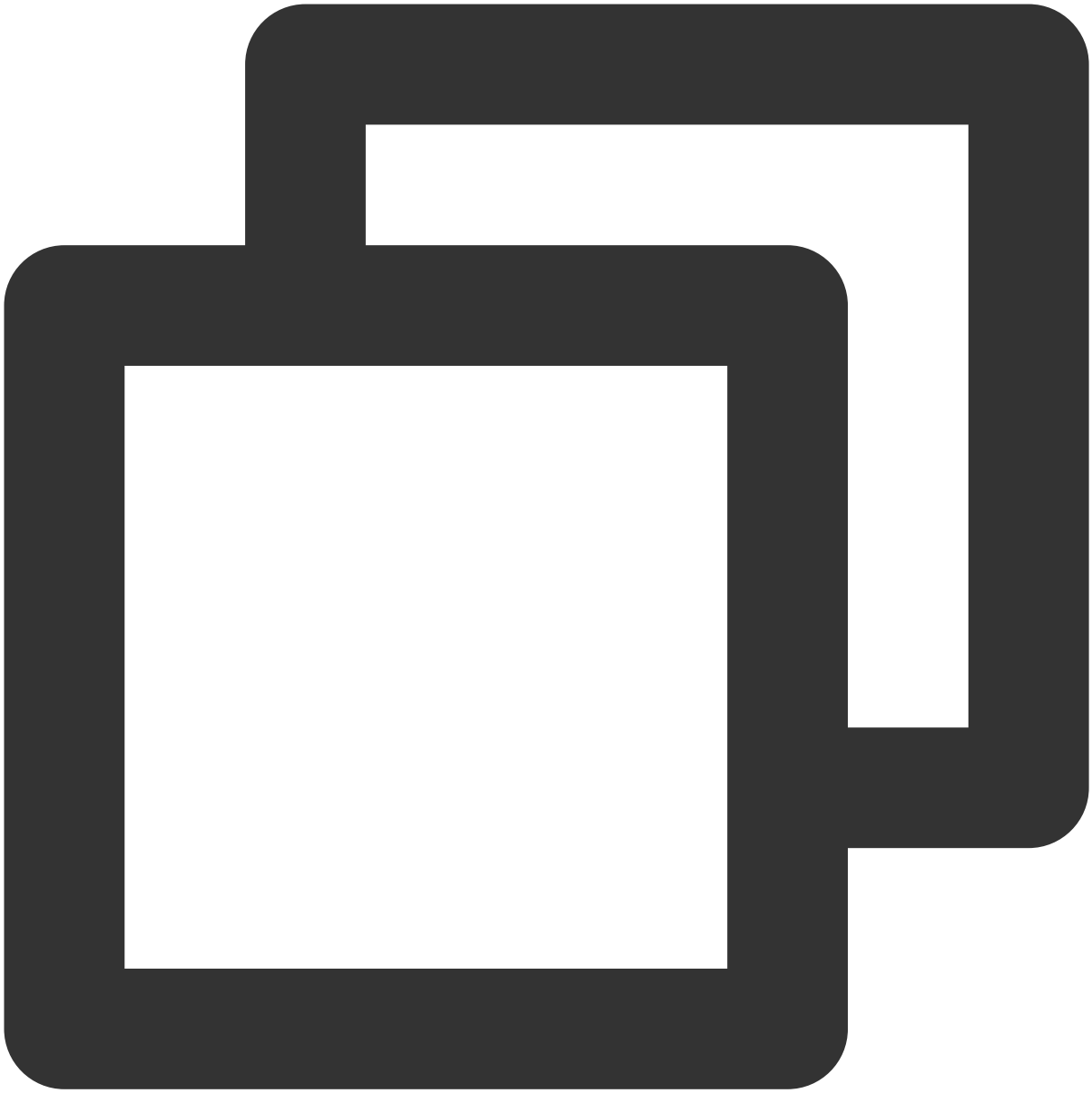
Feature Description：Message sent to the main/Worker thread.

Parameters：Object message, the message to be sent.

Sample code:

worker thread

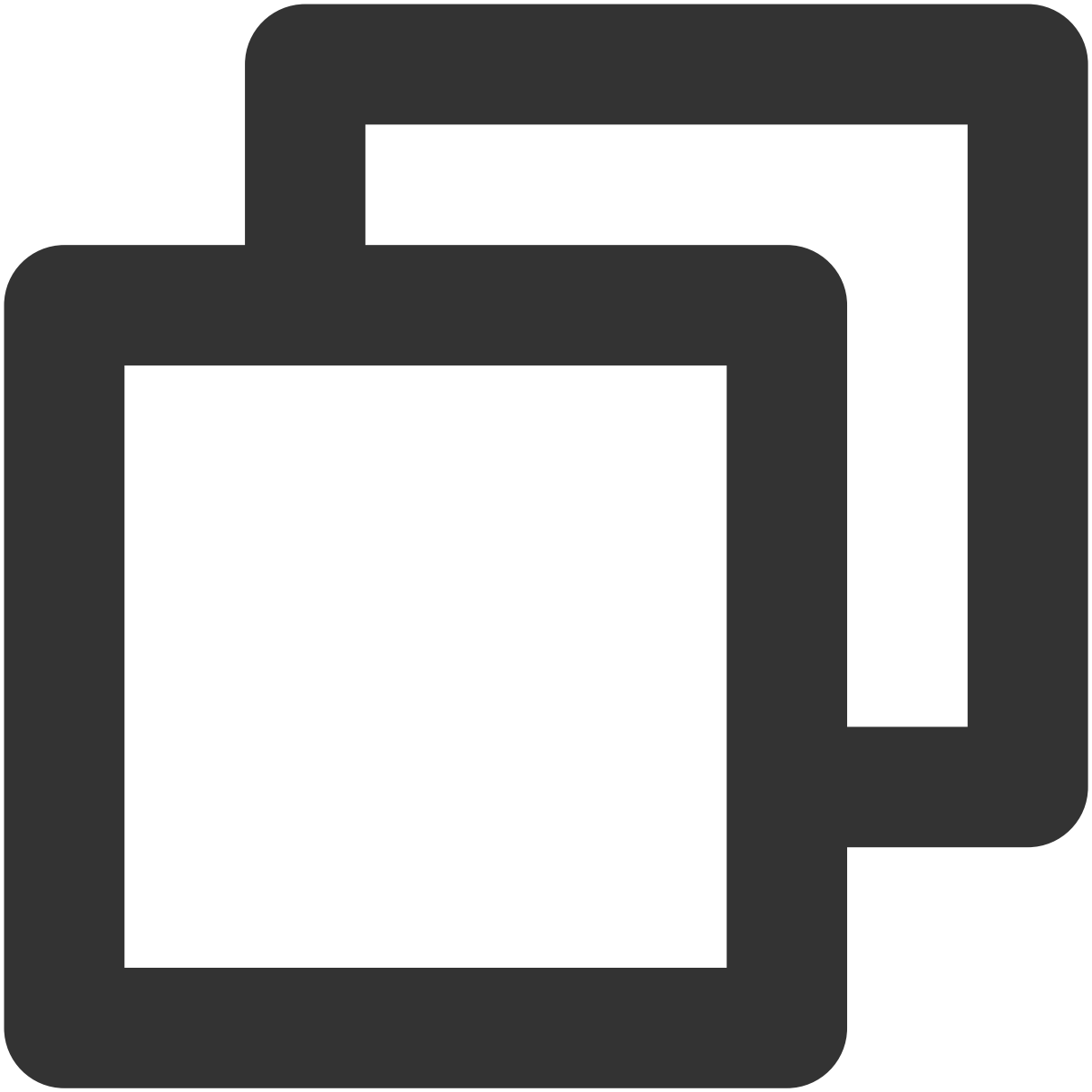
©2013-2022 Tencent Cloud. All rights reserved.



```
worker.postMessage({
  msg: 'hello from worker'
})
```

main thread

©2013-2022 Tencent Cloud. All rights reserved.



```
const worker = wx.createWorker('workers/request/index.js')

worker.postMessage({
  msg: 'hello from main'
})
```

Woker.terminate

Worker.terminate()

©2013-2022 Tencent Cloud. All rights reserved.

Feature Description：Ends the current Worker thread. Call only on the worker object of the main thread.

©2013-2022 Tencent Cloud. All rights reserved.

Media
rich text

Last updated：2024-07-12 19:22:31

EditorContext

Feature Description: An EditorContext instance can be obtained through wx.createSelectorQuery. The EditorContext is bound to an editor component through an id, thereby manipulating the corresponding editor component.

.blur

This method is used via EditorContext.blur(Object object).

Feature Description: The editor loses focus, concurrently retracting the keyboard.

Parameter and Description: Object

Attribute	Type	Default value	Required	Description
success	function	-	No	Callback Function of Successful Interface Call
fail	function	-	No	Callback Function of Failing Interface Call
complete	function	-	No	Callback function executed upon the completion of the interface invocation (both successful and unsuccessful invocations)

.clear

This method is used via EditorContext.clear(Object object)

Feature Description: Empties the content of the editor.

Parameter and Description: Object.

Attribute	Type	Default value	Required	Description

©2013-2022 Tencent Cloud. All rights reserved.

success	function	-	No	Callback Function of Successful Interface Call
fail	function	-	No	Callback Function of Failing Interface Call
complete	function	-	No	Callback function executed upon the completion of the interface invocation (both successful and unsuccessful invocations)

.format

This method is used via EditorContext.format(string name, string value).

Feature Description: Modifies the style.

Parameter and Description:

string name, the attribute;

string value, value.

A list of styles that support the setting:

name	value
bold	-
italic	-
underline	-
strike	-
ins	-
script	sub / super
header	H1 / H2 / h3 / H4 / h5 / H6
align	left / center / right / justify
direction	rtl
indent	-1 / +1
list	ordered / bullet / check
color	hex color

©2013-2022 Tencent Cloud. All rights reserved.

backgroundColor	hex color
margin/marginTop/marginBottom/marginLeft/marginRight	css style
padding/paddingTop/paddingBottom/paddingLeft/paddingRight	css style
font/fontSize/fontStyle/fontVariant/fontWeight/fontFamily	css style
lineHeight	css style
letterSpacing	css style
textDecoration	css style
textIndent	css style
wordWrap	css style
wordBreak	css style
whiteSpace	css style

Setting a style on a selection that has already been styled removes the style. css style represents the allowed values specified in css.

.getContents

> This method is used via EditorContext.getContents().

Feature Description: Gets the content of the editor.
Parameter and Description: Object.

.getSelectionText

> This method is used via EditorContext.getSelectionText(Object object).

Feature Description: Gets the plain text content within the selected area of the editor. When the editor loses focus or a section is not selected, the returned content is empty.
Parameter and Description: Object.

Attribute	Type	Default value	Required	Description

©2013-2022 Tencent Cloud. All rights reserved.

success	function	-	No	Callback Function of Successful Interface Call
fail	function	-	No	Callback Function of Failing Interface Call
complete	function	-	No	Callback function executed upon the completion of the interface invocation (both successful and unsuccessful invocations)

object.success callback function

Arguments Object.res

Attribute	Type	Description
text	string	Plain text content

.insertDivider

This method is used via EditorContext.insertDivider(Object object)

Feature Description: Inserts a divider.

Parameter and Description: Object.

Attribute	Type	Default value	Required	Description
success	function	-	No	Callback Function of Successful Interface Call
fail	function	-	No	Callback Function of Failing Interface Call
complete	function	-	No	Callback function executed upon the completion of the interface invocation (both successful and unsuccessful invocations)

.insertImage

This method is used via EditorContext.insertImage(Object object).

Feature Description: Inserts an image.

©2013-2022 Tencent Cloud. All rights reserved.

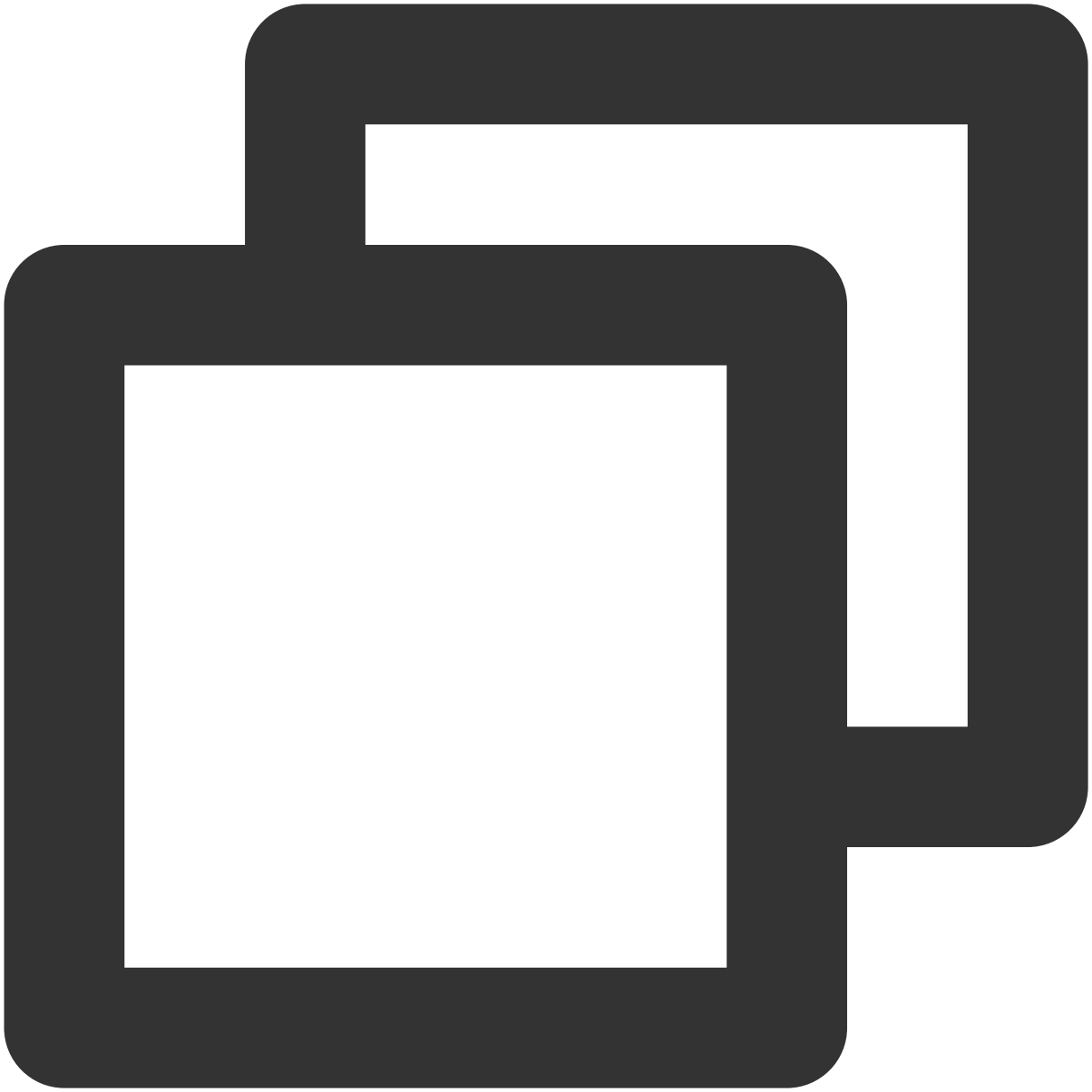
When the address is a temporary file, the HTML format content of the editor obtained adds the attribute data-local to the tag, and the data-local field is added to the image attributes attribute in the data format content, which is the address of the temporary file passed in.
Developers can opt to upload images to the server during the submission phase, replacing them once the network address is obtained. For HTML content, the src value should be replaced, while for delta content, the insert {image: abc} value should be substituted.

Parameter and Description: Object.

Attribute	Type	Default value	Required	Description
src	string		Yes	Image address, only http(s), base64, temp files are supported.
nowrap	boolean	false	No	Whether to automatically line feed after inserting a picture, default line feed
alt	string		No	Alternative text when images cannot be displayed
width	string		No	Image width (pixels/percentage)
height	string		No	Image height (pixels/percentage)
extClass	string		No	Class name added to the image img tag
data	Object		No	data is serialized as name=value;name1=value2 and hooked to the attribute data-custom.
success	function	-	No	Callback Function of Successful Interface Call
fail	function	-	No	Callback Function of Failing Interface Call
complete	function	-	No	Callback function executed upon the completion of the interface invocation (both successful and unsuccessful invocations)

Sample code

©2013-2022 Tencent Cloud. All rights reserved.



```
this.editorCtx.insertImage({
  src: 'xx',
  width: '100px',
  height: '50px',
  extClass: className
})
```

.insertText

©2013-2022 Tencent Cloud. All rights reserved.

This method is used via EditorContext.insertText(Object object).

Feature Description: Overwrites the current selected area, setting a segment of text.

Parameter and Description: Object.

Attribute	Type	Default value	Required	Description
text	string	-	No	Text Content
success	function	-	No	Callback Function of Successful Interface Call
fail	function	-	No	Callback Function of Failing Interface Call
complete	function	-	No	Callback function executed upon the completion of the interface invocation (both successful and unsuccessful invocations)

.redo

This method is used via EditorContext.redo(Object object).

Feature Description: Restoration.

Parameter and Description: Object.

Attribute	Type	Default value	Required	Description
success	function	-	No	Callback Function of Successful Interface Call
fail	function	-	No	Callback Function of Failing Interface Call
complete	function	-	No	Callback function executed upon the completion of the interface invocation (both successful and unsuccessful invocations)

.removeFormat

This method is used via EditorContext.removeFormat(Object object).

©2013-2022 Tencent Cloud. All rights reserved.

Feature Description: Clears the style of the current selected area.

Parameter and Description: Object.

Attribute	Type	Default value	Required	Description
success	function	-	No	Callback Function of Successful Interface Call
fail	function	-	No	Callback Function of Failing Interface Call
complete	function	-	No	Callback function executed upon the completion of the interface invocation (both successful and unsuccessful invocations)

.scrollIntoView

This method is used via EditorContext.scrollIntoView().

Feature Description: Enables the editor cursor to scroll within the visible area of the window.

.setContents

This method is used via EditorContext.setContents(Object object).

Feature Description: Initializes the editor content. When both html and delta exist, only delta takes effect.

Parameter and Description: Object.

Attribute	Type	Default value	Required	Description
html	string		No	HTML content with tags
delta	Object		No	A delta object representing the content
success	function	-	No	Callback Function of Successful Interface Call
fail	function	-	No	Callback Function of Failing Interface Call
complete	function	-	No	Callback function executed upon the completion of the interface invocation (both successful and unsuccessful invocations)

©2013-2022 Tencent Cloud. All rights reserved.

.undo

This method is used via EditorContext.undo(Object object).

Feature Description: Undo.

Parameter and Description: Object.

Attribute	Type	Default value	Required	Description
success	function	-	No	Callback Function of Successful Interface Call
fail	function	-	No	Callback Function of Failing Interface Call
complete	function	-	No	Callback function executed upon the completion of the interface invocation (both successful and unsuccessful invocations)

©2013-2022 Tencent Cloud. All rights reserved.

Video

Last updated：2024-07-12 19:20:45

saveVideoToPhotosAlbum

This API is used via wx.saveVideoToPhotosAlbum(Object object).

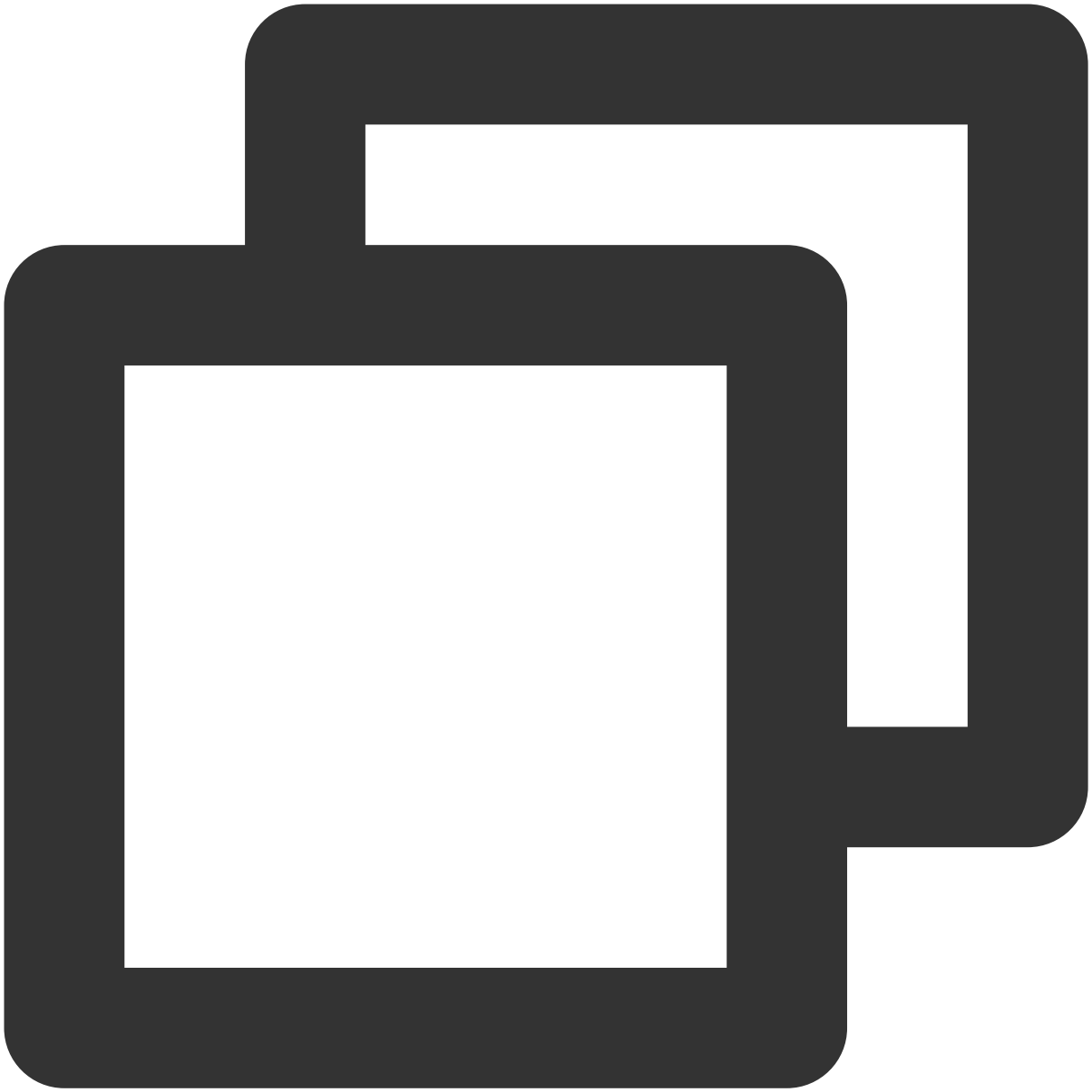
Feature Description: Save video system album. Support mp4 video format.

Parameter and Description: Object.

Attribute	Type	Default value	Required	Description
filePath	string	-	Yes	The path of the video file, which can either be a temporary file path or a permanent file path.
success	function	-	No	Callback Function of Successful Interface Call
fail	function	-	No	Callback Function of Failing Interface Call
complete	function	-	No	Callback function executed upon the completion of the interface invocation (both successful and unsuccessful invocations)

Sample Code

©2013-2022 Tencent Cloud. All rights reserved.



```
wx.saveVideoToPhotosAlbum({
  filePath: 'wxfile://xxx',
  success(res) {
    console.log(res.errMsg)
  },
})
```

createVideoContext

©2013-2022 Tencent Cloud. All rights reserved.

This API is used via VideoContext wx.createVideoContext(string id, Object this).

Feature Description: Creates a VideoContext object.

Parameter and Description:

string id: The id of the <video> component.

"this" of the current component instance under custom components, used to operate the <video> component within.

Return Value: VideoContext

chooseVideo

This API is used via wx.chooseVideo(Object object).

Feature Description: Captures a video or selects a video from the phone's album.

Parameter and Description: Object.

Attribute	Type	Default value	Required	Description
sourceType	Array.\\ <string>	['album', 'camera']	No	Video Selection Source
compressed	boolean	true	No	Whether to compress the selected video file
maxDuration	number	60	No	Maximum recording time of video, in seconds.
camera	string	‘back’	No	The default is to activate the front or rear camera. However, on some Android phones, this may not work because the system ROM does not support this feature.
success	function	-	No	Callback Function of Successful Interface Call
fail	function	-	No	Callback Function of Failing Interface Call
complete	function	-	No	Callback function executed upon the completion of the interface invocation (both successful and unsuccessful invocations)

Valid values of object.sourceType

©2013-2022 Tencent Cloud. All rights reserved.

Value	Description
album	Select video from the album
camera	Capture video using the camera

Valid values of object.camera

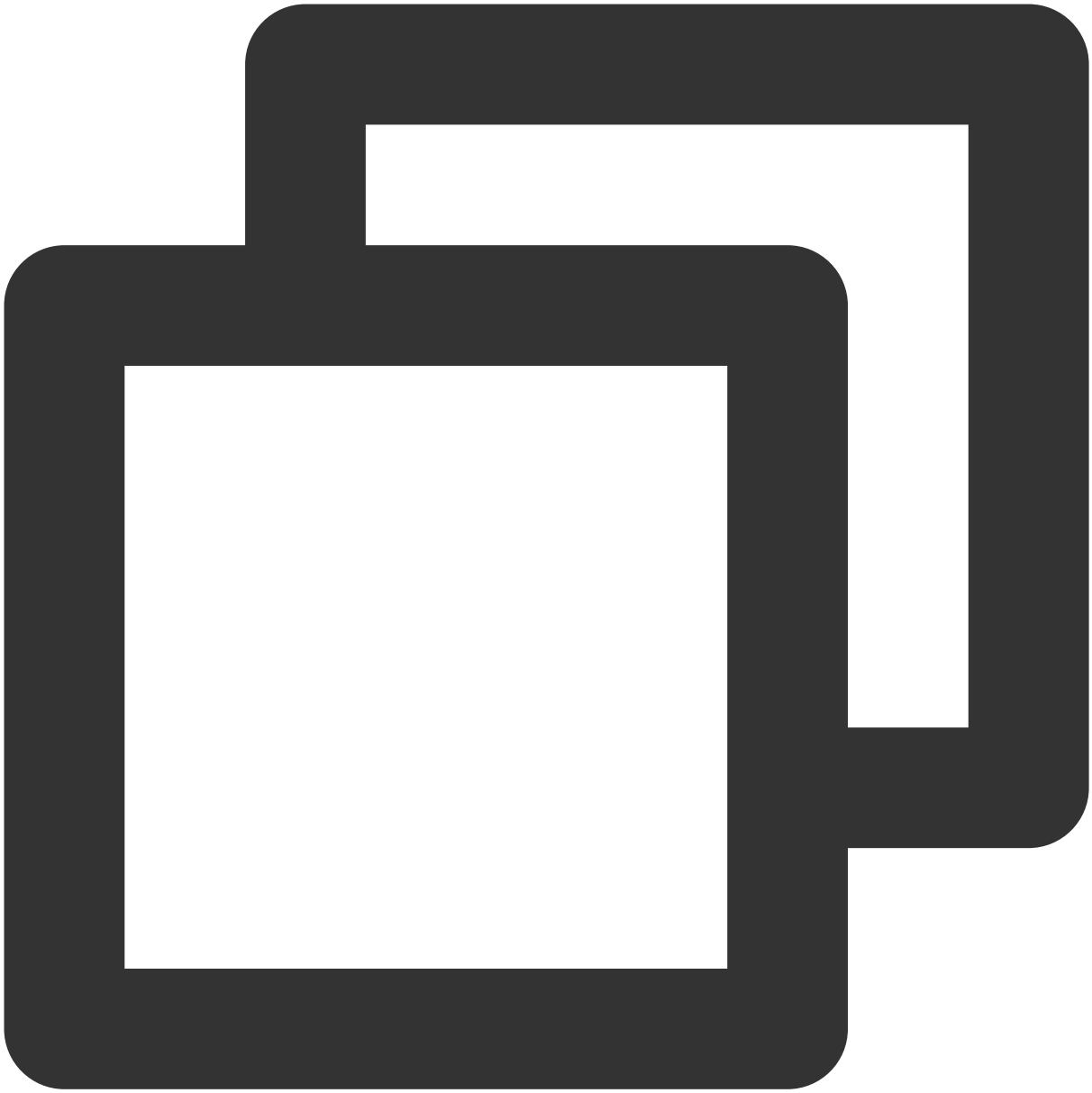
Value	Description
back	Default to activating the rear camera.
front	Default to activating the front camera.

Parameters for object.success callback function: Object res.

Attribute	Type	Description
tempFilePath	string	Temporary file path of the selected video.
duration	number	Duration of the selected video.
size	number	Data size of the selected video.
height	number	Return to selecting the height of the selected video.
width	number	Return to selecting the width of the selected video.

Sample Code

©2013-2022 Tencent Cloud. All rights reserved.



```
wx.chooseVideo({
  sourceType: ['album', 'camera'],
  maxDuration: 60,
  camera: 'back',
  success(res) {
    console.log(res.tempFilePath)
  },
})
```

©2013-2022 Tencent Cloud. All rights reserved.

compressVideo

This API is used via wx.compressVideo(Object object).

Feature Description: Compressed video interface. The developer can specify the quality of the compression for compression. When finer control is needed, bitrate, fps, and resolution can be specified, which are ignored when quality is passed in.

Parameter and Description: Object.

Attribute	Type	Required	Description
src	string	Yes	The path of the video file, which can either be a temporary file path or a permanent file path.
quality	string	Yes	Compression quality, whose valid values are: Low: Low power Medium: Moderate power High: High power
bitrate	number	Yes	Bitrate, measured in kbps.
fps	number	Yes	Frame rate
resolution	number	Yes	Resolution ratio relative to the original video, with the value range of (0, 1].
success	function	No	Callback Function of Successful Interface Call
fail	function	No	Callback Function of Failing Interface Call
complete	function	No	Callback function executed upon the completion of the interface invocation (both successful and unsuccessful invocations)

Parameters for object.success callback function: Object res.

Attribute	Type	Description
tempFilePath	string	Address of the temporary file after compression.
size	string	Size after compression, measured in kB.

chooseMedia

©2013-2022 Tencent Cloud. All rights reserved.

This method is used via wx.chooseMedia(Object object).

Feature Description: Captures or selects images or videos from the mobile phone's album.

Parameter and Description: Object

Attribute	Type	Default value	Required	Description
count	number	9	No	Number of files that can be selected, supports up to 15 files
mediaType	Array.<string>	['image', 'video']	No	File type, whose valid values are: Image: Only capture images or select images from the album. Video: Only capture videos or select videos from the album.
sourceType	Array.<string>	['album', 'camera']	No	The source of image and video selection, whose valid values are: Album: Select from album Camera: Uses the camera for photographing
maxDuration	number	10	No	Maximum recording time for shooting video, in seconds. The time range is between 3s and 60s. There is no limit to albums.
sizeType	Array.<string>	['original', 'compressed']	No	Only available when mediaType is image, whether to compress the selected file or not.
camera	string	'back'	No	Only valid when sourceType is camera, use front or back camera, legal value is: back: use rear camera front: use front camera
success	function	-	No	Callback Function of Successful Interface Call
fail	function	-	No	Callback Function of Failing Interface Call
complete	function	-	No	Callback function executed upon the completion of the interface invocation (both successful and unsuccessful invocations)

Parameters for object.success callback function: Object res.

Attribute	Type	Description

©2013-2022 Tencent Cloud. All rights reserved.

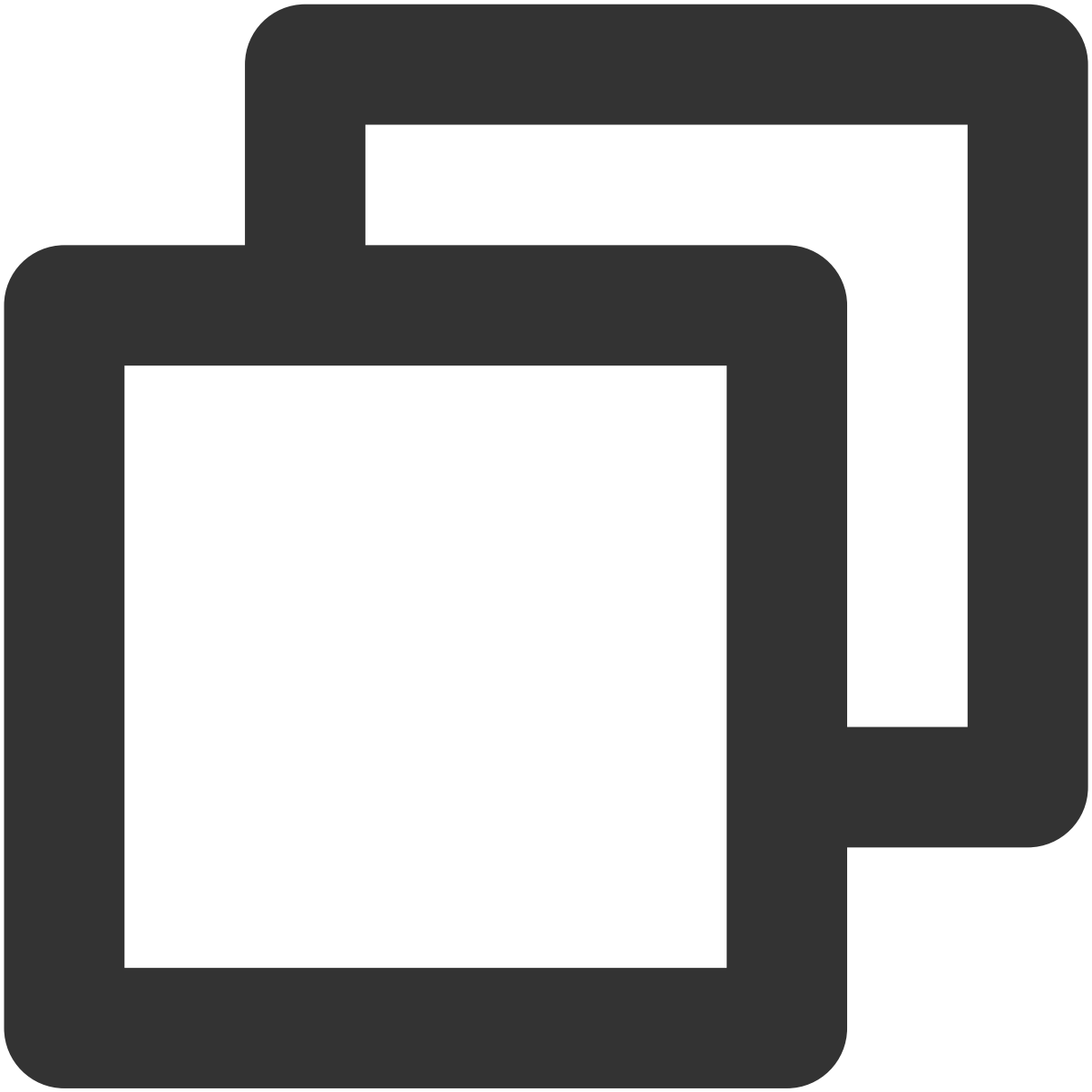
tempFiles	Array.<Object>	List of local temporary files
type	string	File type, whose valid values include image, video

Structure attribute of tempFiles

Structure attributes	Type	Description
tempFilePath	string	Local temporary file path (local direct)
size	number	Size of local temporary files, measured in bytes.
duration	number	Duration of video
height	number	Video height
width	number	Video width
fileType	string	File type, whose valid values are: Image Video

Sample Code

©2013-2022 Tencent Cloud. All rights reserved.



```
wx.chooseMedia({
  count: 9,
  mediaType: ['image','video'],
  sourceType: ['album', 'camera'],
  maxDuration: 30,
  camera: 'back',
  success(res) {
    console.log(res.tempFiles.tempFilePath)
    console.log(res.tempFiles.size)
  }
})
```

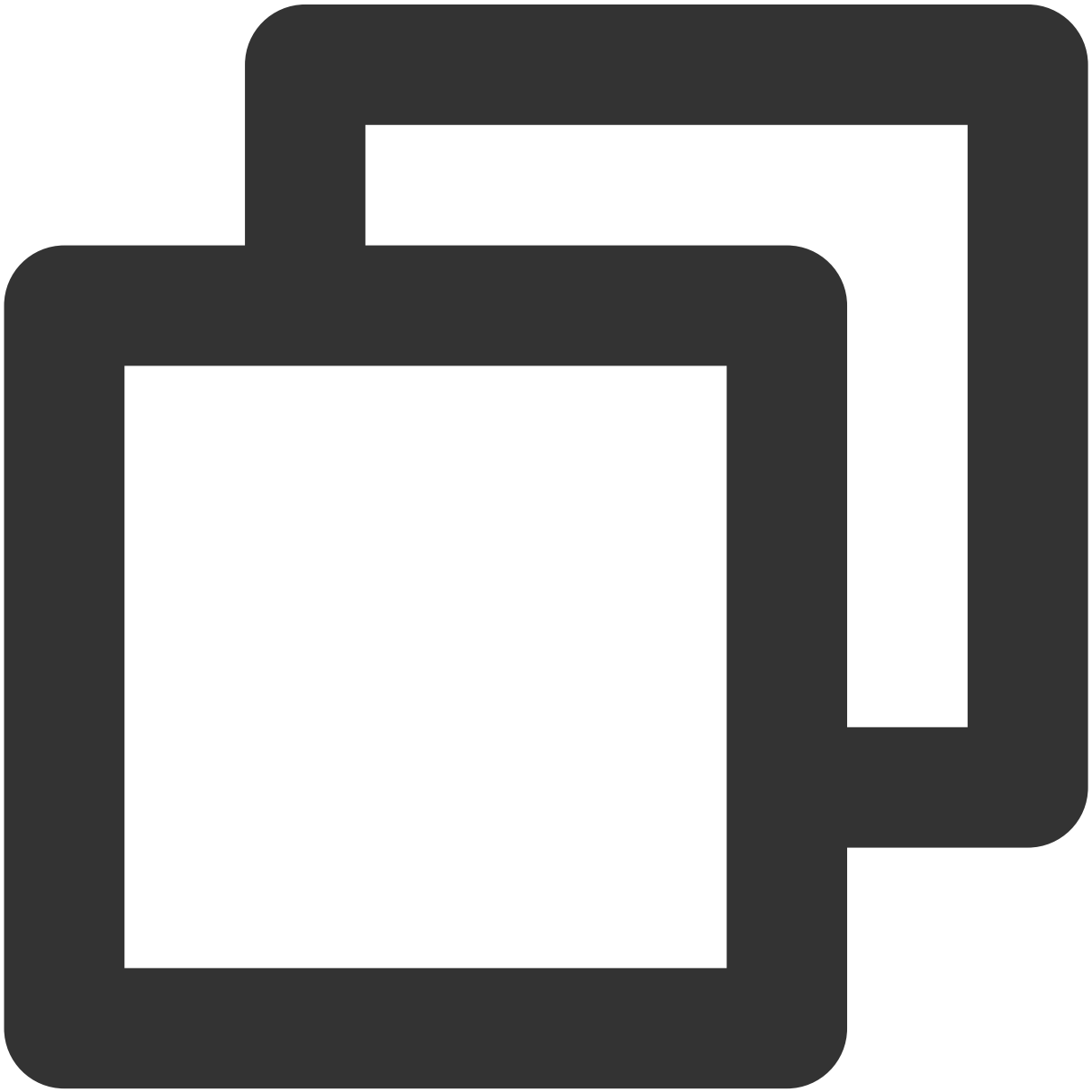
©2013-2022 Tencent Cloud. All rights reserved.

VideoContext

Feature Description: VideoContext instance, which can be obtained through wx.createVideoContext. VideoContext is bound to a <video> component through its id, and operates the corresponding <video> component.

Sample Code

Corresponding WXML file:

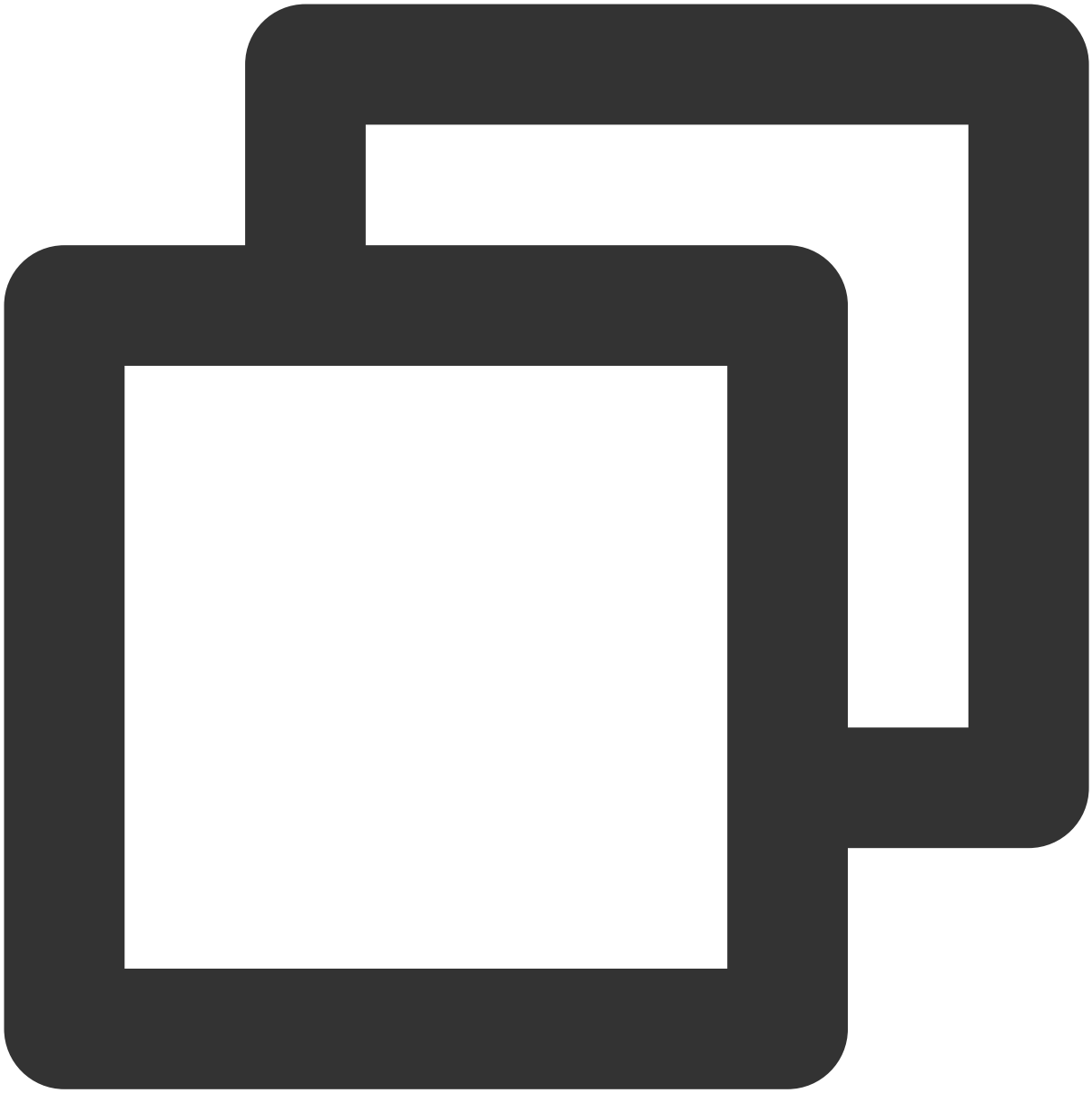


©2013-2022 Tencent Cloud. All rights reserved.

```
<view class="section tc">
  <video
    id="myVideo"
    src="http://wxsnsdy.tc.qq.com/105/20210/snsdyvideodownload?filekey=302802010104
    enable-danmu
    danmu-btn
    controls
  ></video>
  <view class="btn-area">
    <input bindblur="bindInputBlur" />
    <button bindtap="bindSendDanmu">Dispatch the barrage</button>
  </view>
</view>
```

Corresponding js file:

©2013-2022 Tencent Cloud. All rights reserved.



```
function getRandomColor() {
  const rgb = []
  for (let i = 0; i < 3; ++i) {
    let color = Math.floor(Math.random() * 256).toString(16)
    color = color.length == 1 ? '0' + color : color
    rgb.push(color)
  }
  return '#' + rgb.join('')
}

Page({
```

©2013-2022 Tencent Cloud. All rights reserved.

```
  onReady(res) {
    this.videoContext = wx.createVideoContext('myVideo')
  },
  inputValue: '',
  bindInputBlur(e) {
    this.inputValue = e.detail.value
  },
  bindSendDanmu() {
    this.videoContext.sendDanmu({
      text: this.inputValue,
      color: getRandomColor(),
    })
  },
})
```

Method Set

exitFullScreen

This method is used via VideoContext.exitFullScreen().

Feature Description: Exits full screen.

exitBackgroundPlayback

This method is used via VideoContext.exitBackgroundPlayback().

Feature Description: Exits backend audio playback mode.

exitPictureInPicture

This method is used via VideoContext.exitPictureInPicture(Object object).

Feature Description: Exits mini window, a method that can be invoked on any page.

Parameter and Description: Object

Attribute	Type	Default value	Required	Description
success	function	-	No	Callback Function of Successful Interface Call

©2013-2022 Tencent Cloud. All rights reserved.

fail	function	-	No	Callback Function of Failing Interface Call
complete	function	-	No	Callback function executed upon the completion of the interface invocation (both successful and unsuccessful invocations)

hideStatusBar

This method is used via VideoContext.hideStatusBar().

Feature Description: Hides the status bar, which is effective only in iOS full screen mode.

pause

This method is used via

Feature Description: Pauses the video.

play

This method is used via

Feature Description: Plays the video.

requestFullScreen

This method is used via VideoContext.requestFullScreen(Object object).

Feature Description: Enters full screen.

Parameter and Description: Object

Attribute	Type	Default value	Required	Description
direction	number	-	No	Sets the orientation of the video in full screen mode. If not specified, it will be

©2013-2022 Tencent Cloud. All rights reserved.

				automatically determined based on the aspect ratio.

Valid values of object.direction

Value	Description
0	Normal vertical orientation
90	Screen rotating 90 degrees counterclockwise
-90	Screen rotating 90 degrees clockwise

requestBackgroundPlayback

This method is used via VideoContext.requestBackgroundPlayback().

Feature Description: Enters the backend audio playback mode.

seek

This method is used via VideoContext.seek(number position).

Feature Description: Redirects to the specified position.

Parameter and Description: number position, the position to redirect to, unit: seconds.

sendDanmu

This method is used via VideoContext.sendDanmu(Object data).

Feature Description: Sends a barrage.

Parameter: Object data, barrage content.

Attribute	Type	Default value	Required	Description
text	string	-	Yes	Barrage text
color	string	-	No	Barrage color

©2013-2022 Tencent Cloud. All rights reserved.

showStatusBar

This method is used via VideoContext.showStatusBar().

Feature Description: Displays the status bar, which is effective only in iOS full screen.

stop

This method is used via VideoContext.stop().

Feature Description: Stops the video.

©2013-2022 Tencent Cloud. All rights reserved.

Audio recording

Last updated：2024-07-12 19:20:00

startRecord

This API is used via wx.startRecord(Object object).

Feature Description: Starts audio recording. The audio recording will automatically cease when wx.stopRecord is actively invoked, or when the audio recording exceeds one minute. This interface cannot be invoked when the user exits the mini program.

Parameter and Description: Object.

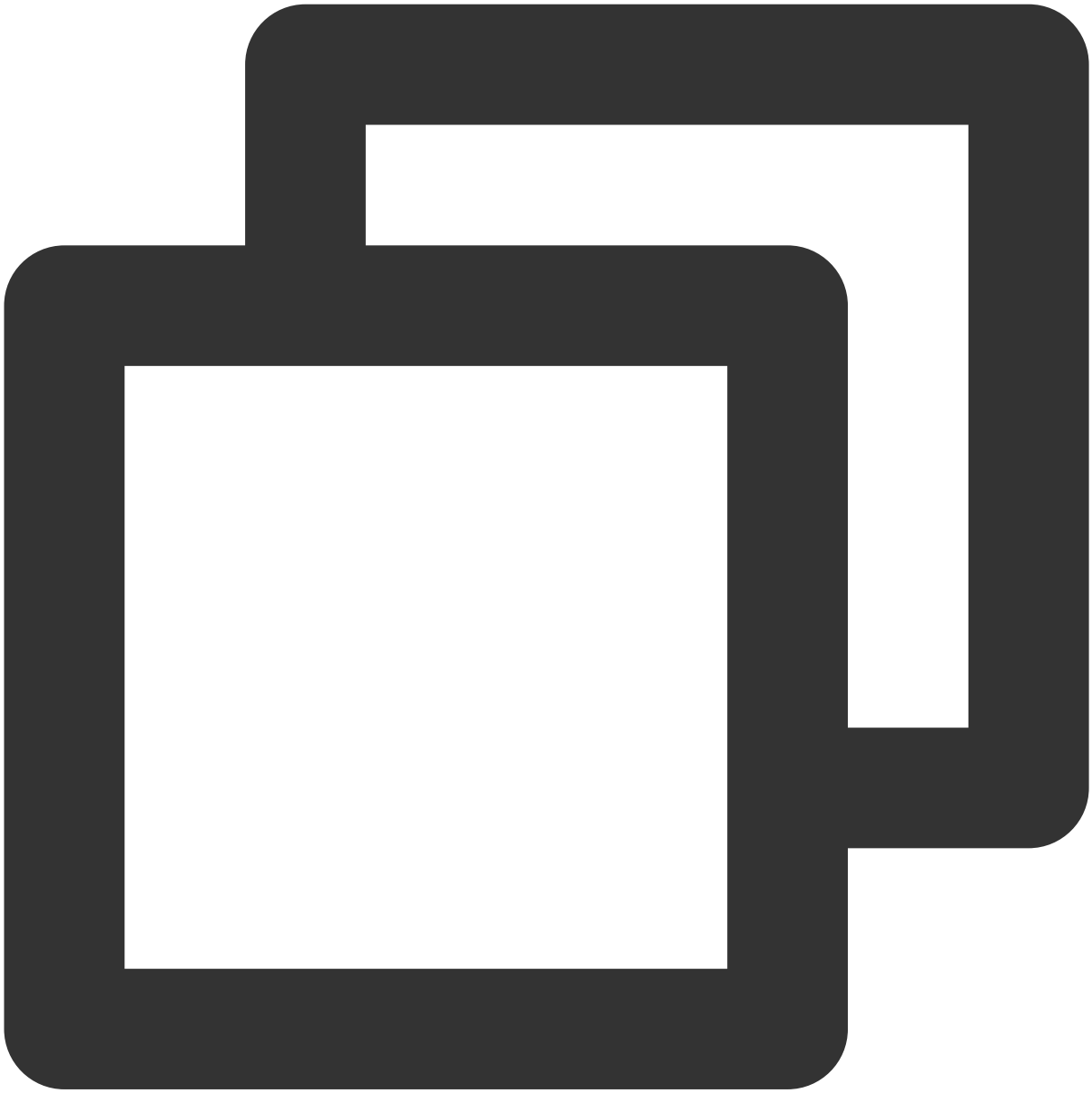
Attribute	Type	Default value	Required	Description
success	function	-	No	Callback Function of Successful Interface Call
fail	function	-	No	Callback Function of Failing Interface Call
complete	function	-	No	Callback function executed upon the completion of the interface invocation (both successful and unsuccessful invocations)

Parameters for object.success callback function: Object res.

Attribute	Type	Description
tempFilePath	string	Temporary path of audio recording file (local path)

Sample Code

©2013-2022 Tencent Cloud. All rights reserved.



```
wx.startRecord({
  success (res) {
    const tempFilePath = res.tempFilePath
  }
})
setTimeout(function () {
  wx.stopRecord() // Stop audio recording
}, 10000)
```

©2013-2022 Tencent Cloud. All rights reserved.

stopRecord

This API is used via wx.stopRecord(Object object).

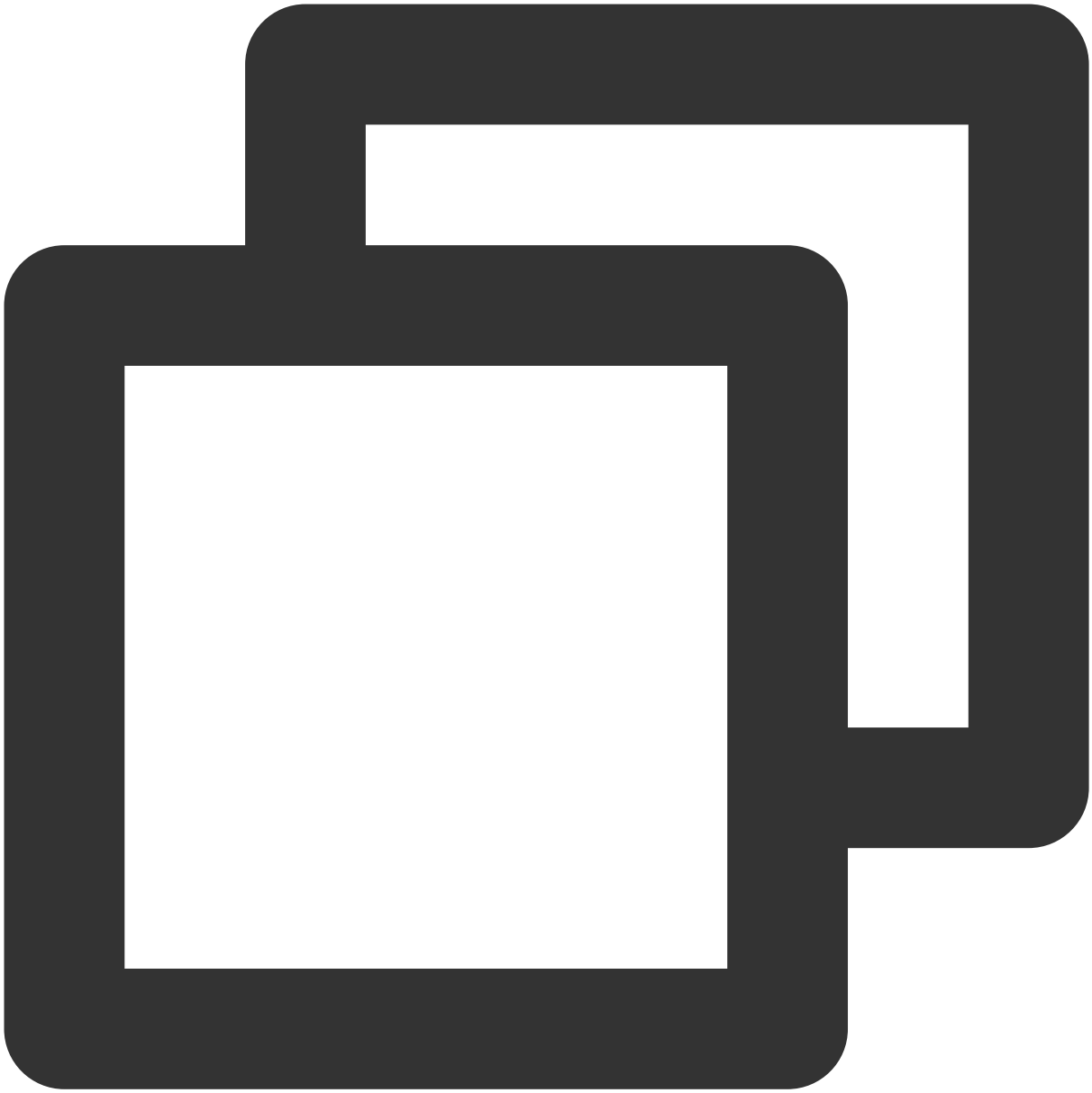
Feature Description: Stops audio recording.

Parameter and Description: Object.

Attribute	Type	Default value	Required	Description
success	function	-	No	Callback Function of Successful Interface Call
fail	function	-	No	Callback Function of Failing Interface Call
complete	function	-	No	Callback function executed upon the completion of the interface invocation (both successful and unsuccessful invocations)

Sample Code

©2013-2022 Tencent Cloud. All rights reserved.



```
wx.startRecord({
  success (res) {
    const tempFilePath = res.tempFilePath
  }
})
setTimeout(function () {
  wx.stopRecord() // Stop audio recording
}, 10000)
```

©2013-2022 Tencent Cloud. All rights reserved.

getRecorderManager

This API is used via RecorderManager wx.getRecorderManager().

Feature Description: Gets the globally unique RecorderManager.
Return Value: RecorderManager.

RecorderManager

.onError

This method is used via RecorderManager.onError(function listener).

Feature Description: Monitors audio recording error events.
Parameter and Description: function listener, the listener function for audio recording error events.

Attribute	Type	Description
errMsg	string	Error Message

.onFrameRecorded

This method is used via RecorderManager.onFrameRecorded(function listener).

Feature Description: Monitors events where files have been recorded to a specified frame size. If frameSize is set, this event will be called back.
Parameter and Description: function listener, the listener function for events where files have been recorded to a specified frame size.

Attribute	Type	Description
frameBuffer	ArrayBuffer	Audio recording shard data
isLastFrame	boolean	Is the current frame the last one before the normal end of the audio recording?

©2013-2022 Tencent Cloud. All rights reserved.

.onInterruptionBegin

This method is used via RecorderManager.onInterruptionBegin(function listener).

Feature Description: Monitors events where audio recording is interrupted due to system occupation. WeChat voice chat and WeChat video chat scenarios will trigger such events, and subsequently the audio recording will pause.
Parameter and Description: function listener, the listener function for events where audio recording is interrupted due to system occupation.

.onInterruptionEnd

This method is used via RecorderManager.onInterruptionEnd(function listener).

Feature Description: Monitors audio recording interruption ending events. After receiving the interruptionBegin event, all audio recordings within the mini program will pause, and only after receiving this event can audio recording be successfully resumed.
Parameter and Description: function listener, the listener function for audio recording interruption ending events.

.onPause

This method is used via RecorderManager.onPause(function listener).

Feature Description: Monitors audio recording pause events.
Parameter and Description: function listener, the listener function for audio recording pause events.

.onResume

This method is used via RecorderManager.onResume(function listener).

Feature Description: Monitors audio recording continuation events.
Parameter and Description: function listener, the listener function for audio recording continuation events.

.onStart

©2013-2022 Tencent Cloud. All rights reserved.

This method is used via RecorderManager.start(Object object).

Feature Description: Monitors audio recording start events.
Parameter and Description: function listener, the listener function for audio recording start events.

.onStop

This method is used via RecorderManager.onStop(function listener).

Feature Description: Monitors audio recording ending events.
Parameters and Description: function listener, the listener function for audio recording ending events. Object res parameter.

Attribute	Type	Description
tempFilePath	string	Temporary path of audio recording file (local path)
duration	number	Total audio recording duration, in milliseconds.
fileSize	number	Audio recording file size, in bytes.

.pause

This method is used via RecorderManager.pause().

Feature Description: Pauses audio recording.

.resume

This method is used via RecorderManager.resume().

Feature Description: Continues audio recording.

.start

©2013-2022 Tencent Cloud. All rights reserved.

This method is used via RecorderManager.start(Object object).

Feature Description: Starts audio recording.

Parameter and Description: Object

Attribute	Type	Default value	Required	Description
duration	number	60000	No	The duration of the audio recording, measured in milliseconds (ms), with a maximum value of 600,000 (equivalent to 10 minutes).
sampleRate	number	80000	No	Sampling rate (not supported on PC)
numberOfChannels	number	2	No	Number of audio recording channels
encodeBitRate	number	48000	No	Encoding bitrate, with valid values provided in the table below.
format	string	aac	No	Audio Format
frameSize	number	-	No	Specify the frame size in kilobytes (KB). Upon passing in the frameSize, the content of the files recorded to the specified frame size will be called back. If not specified, no callback will occur. Currently, only mp3 and pcm formats are supported.
audioSource	string	auto	No	Specify the audio input source for audio recording.

object.numberOfChannels Legal values

Legal value	Description
1	1 channel
2	2 channel

object.format Legal values

Legal value	Description
mp3	mp3

©2013-2022 Tencent Cloud. All rights reserved.

aac	aac
wav	wav
PCM	pcm

object.audioSource Legal values

Legal value	Description
auto	Automatically set up, the default use of cell phone microphone, plug in the headset automatically switch to use the headset microphone, all platforms applicable
buildInMic	Cellular microphone, iOS only
headsetMic	Wired headset microphone, iOS only
mic	Microphone (phone microphone when headset is not plugged in, headset microphone when headset is plugged in), Android only
camcorder	Same as mic for recording audio and video content, Android only
voice_communication	Same as mic, for real-time communication, Android only
voice_recognition	Same as mic, for speech recognition, Android only

Sampling rate and encoding bitrate limits

Each sampling rate corresponds to a valid range of encoding bitrates. Setting an invalid sampling rate or encoding bitrate will result in audio recording failure. The specific correspondences are as follows in the table.

Sampling rate.	Encoding bitrate
8000	16000 ~ 48000
11025	16000 ~ 48000
12000	24000 ~ 64000
16000	24000 ~ 96000
22050	32000 ~ 128000
24000	32000 ~ 128000
32000	48000 ~ 192000
44100	64000 ~ 320000

©2013-2022 Tencent Cloud. All rights reserved.

48000	64000 ~ 320000

.stop

This method is used via RecorderManager.stop().

Feature Description: Stop audio recording.

©2013-2022 Tencent Cloud. All rights reserved.

Background audio

Last updated：2024-07-12 19:16:26

stopBackgroundAudio

The method of utilizing this API is wx.stopBackgroundAudio(Object object).

Function Description: Halts the playback of music.

Parameters and Description: Object object.

Attributes	Types	Default value	Required	Note
success	function	-	Not required	Callback function for successful interface invocation
fail	function	-	Not required	Callback function for unsuccessful interface invocation
complete	function	-	Not required	Callback function executed upon the completion of the interface invocation (both successful and unsuccessful invocations)

seekBackgroundAudio

The method for utilizing this API is wx.seekBackgroundAudio(Object object)

Function Description: Controls the progression of music playback.

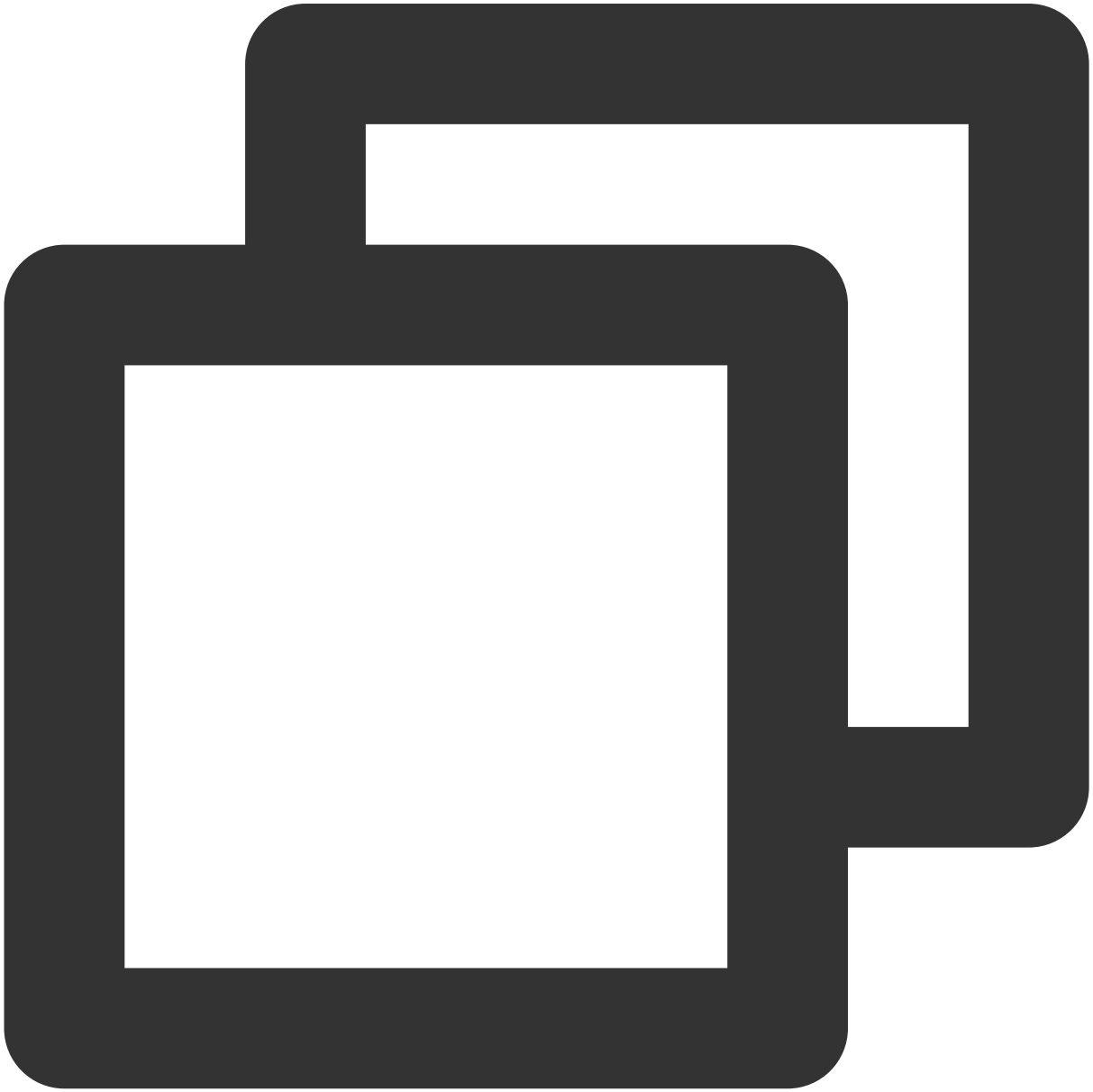
Parameters and Description: Object object.

Attributes	Types	Default value	Required	Note
position	number	-	Supported	Music position, measured in seconds
success	function	-	Not required	Callback function for successful interface invocation

©2013-2022 Tencent Cloud. All rights reserved.

fail	function	-	Not required	Callback function for unsuccessful interface invocation
complete	function	-	Not required	Callback function executed upon the completion of the interface invocation (both successful and unsuccessful invocations)

Example Code



```
wx.seekBackgroundAudio({
    position: 30
```

©2013-2022 Tencent Cloud. All rights reserved.

```
})
```

playBackgroundAudio

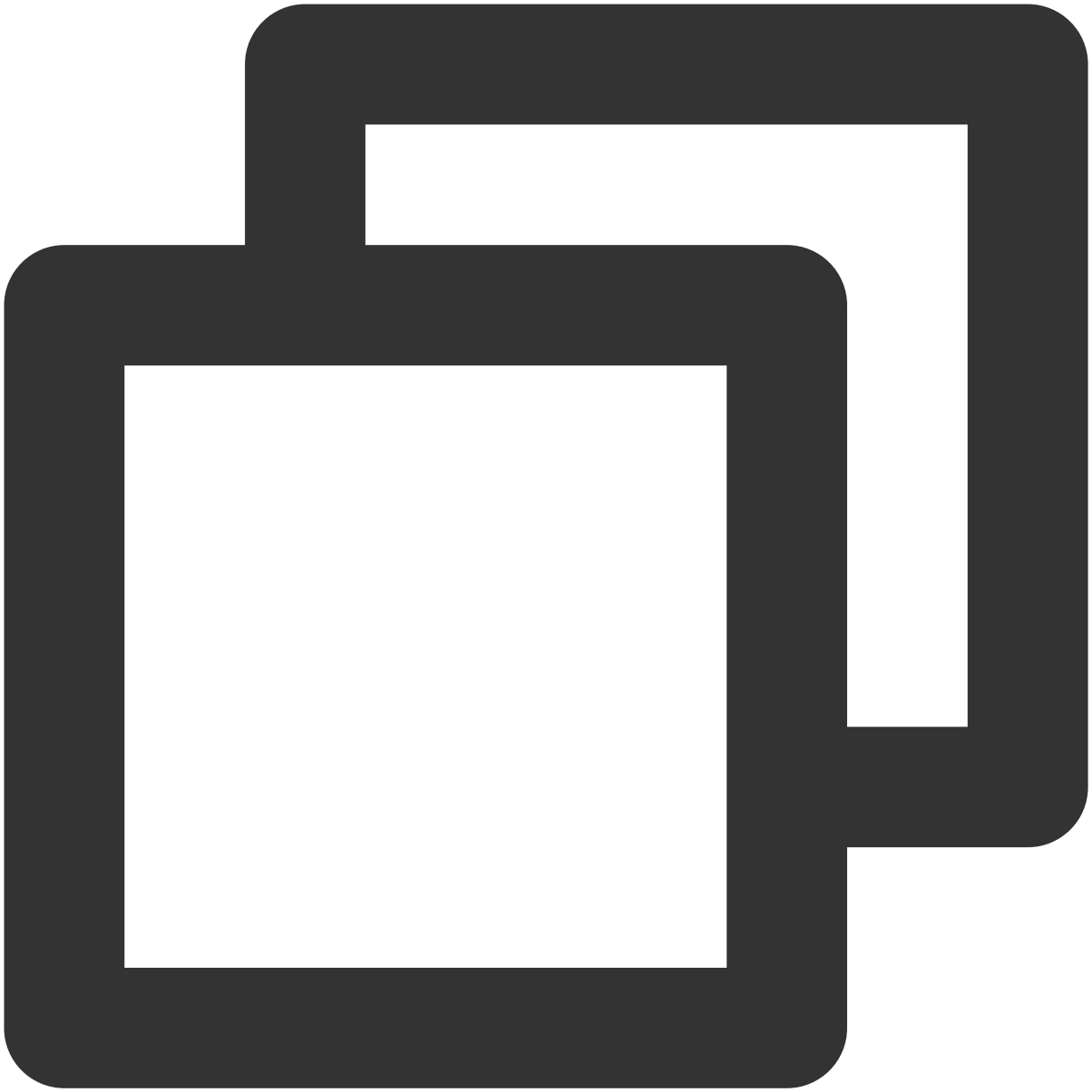
The method for employing this API is wx.playBackgroundAudio(Object object)

Function Description: Utilizes the background player for music playback. For the host client, only one background music can be played at a time. When the user exits the mini program, the music will pause; if the music player is occupied by another mini program, the music within the original mini program will cease to play.
Parameters and Description: Object object.

Attributes	Types	Default value	Required	Note
dataUrl	string	-	Supported	Music link, currently supporting formats include m4a, aac, mp3, wav
title	string	-	Not required	Music Title
coverImgUrl	string	-	Not required	Cover URL
success	function	-	Not required	Callback function for successful interface invocation
fail	function	-	Not required	Callback function for unsuccessful interface invocation
complete	function	-	Not required	Callback function executed upon the completion of the interface invocation (both successful and unsuccessful invocations)

Example Code

©2013-2022 Tencent Cloud. All rights reserved.



```
wx.playBackgroundAudio({
  dataUrl: '',
  title: '',
  coverImgUrl: ''
})
```

pauseBackgroundAudio

©2013-2022 Tencent Cloud. All rights reserved.

The method for utilizing this API is wx.pauseBackgroundAudio(Object object)

Function Description: Pauses music playback.

Parameters and Description: Object object.

Attributes	Types	Default value	Required	Note
success	function	-	Not required	Callback function for successful interface invocation
fail	function	-	Not required	Callback function for unsuccessful interface invocation
complete	function	-	Not required	Callback function executed upon the completion of the interface invocation (both successful and unsuccessful invocations)

onBackgroundAudioStop

The method for utilizing this API is wx.onBackgroundAudioStop(function listener)

Function Description: Monitors music stop events.

Parameters and Description: function listener, the monitoring function for music stop events.

onBackgroundAudioPlay

The method for utilizing this API is wx.onBackgroundAudioPlay(function listener)

Function Description: Monitors music playback events.

Parameters and Description: function listener, the monitoring function for music playback events.

onBackgroundAudioPause

©2013-2022 Tencent Cloud. All rights reserved.

The method for utilizing this API is wx.onBackgroundAudioPause(function listener)

Function Description: Monitors music pause events.

Parameters and Description: function listener, the monitoring function for music pause events.

getBackgroundAudioPlayerState

The method for utilizing this API is wx.getBackgroundAudioPlayerState(Object object)

Function Description: Retrieves the status of background music playback.

Parameters and Description: Object object.

Attributes	Types	Default value	Required	Note
success	function	-	Not required	Callback function for successful interface invocation
fail	function	-	Not required	Callback function for unsuccessful interface invocation
complete	function	-	Not required	Callback function executed upon the completion of the interface invocation (both successful and unsuccessful invocations)

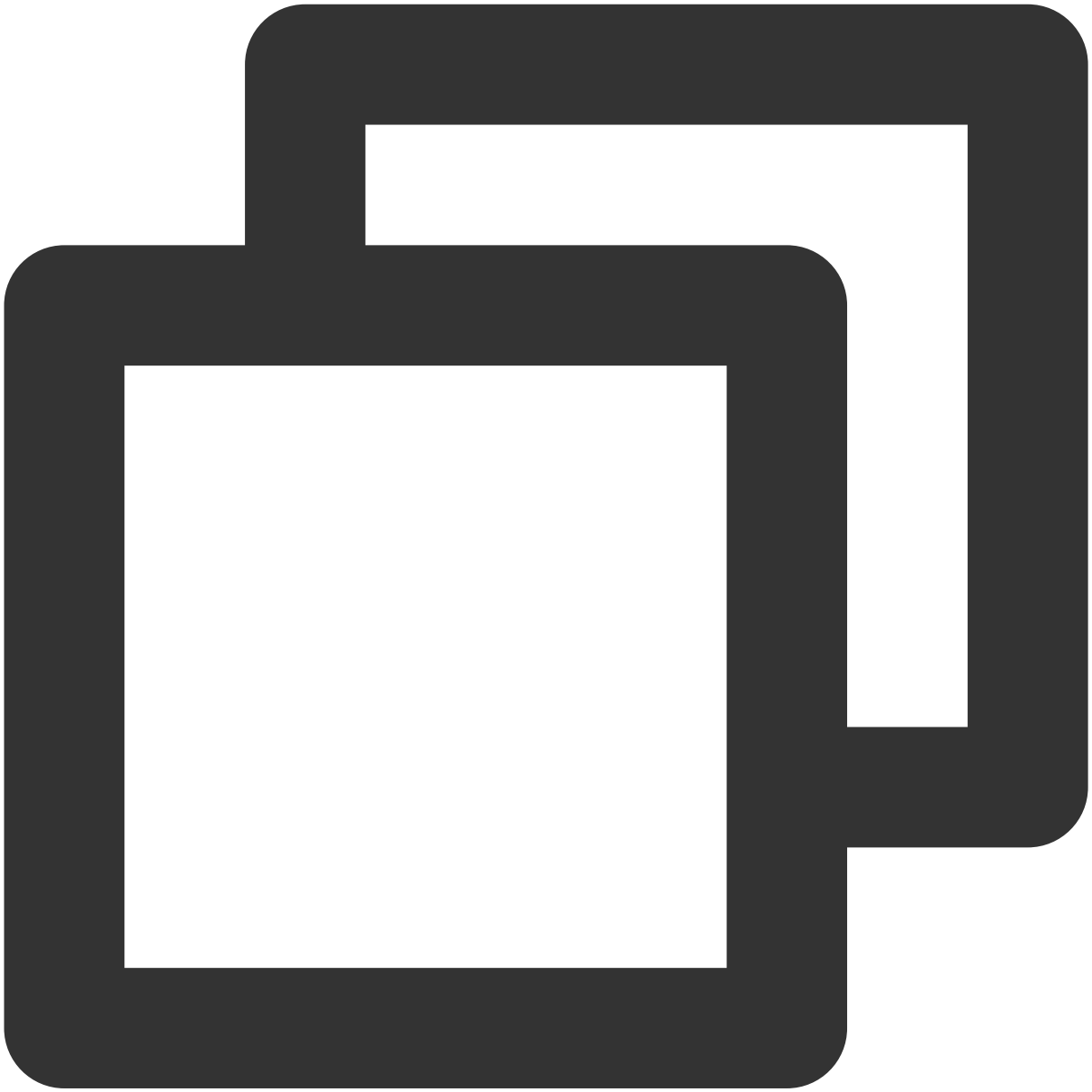
object.success Callback Function Parameters: Object res.

Attributes	Types	Note
duration	number	The length of the selected audio (in seconds), returned only during music playback.
currentPosition	number	Selects the playback position of the audio (in s), only returned when the music is playing
status	number	Playback status, valid values: 0: Paused 1: Playing 2: No music playing

©2013-2022 Tencent Cloud. All rights reserved.

downloadPercent	number	The percentage of audio download progress, returned only during music playback.
dataUrl	string	Song data link, returned only during music playback.

Example Code



```
wx.getBackgroundAudioPlayerState({
  success (res) {
    const status = res.status
    const dataUrl = res.dataUrl
```

©2013-2022 Tencent Cloud. All rights reserved.

```
    const currentPosition = res.currentPosition
    const duration = res.duration
    const downloadPercent = res.downloadPercent
  }
})
```

getBackgroundAudioManager

Function Description: Acquire the globally unique background audio manager. When the mini program switches to the background, if the audio is playing, it can continue to play. However, the background status cannot manipulate the audio playback status through API calls.

BackgroundAudioManager

BackgroundAudioManager instance, can be obtained through wx.getBackgroundAudioManager.

Attributes

Types	Note
string src	The data source of the audio. Default is empty string, when new src is set, it will start playing automatically, currently supported formats are m4a, aac, mp3, wav
number startTime	The position where the audio starts playing, unit: s.
string title	Audio title, used for the native audio player's audio title (required). The share function in the native audio player, the title of the shared card, will also use this value.
string epname	Album name, the share function in the native audio player, the description of the shared card, will also use this value.
string singer	Singer's name, the share function in the native audio player, the description of the shared card, will also use this value.
string coverImgUrl	Cover image URL, used as the background image for the native audio player. The share function in the native audio player, the accompanying image and background of the shared card will also use this image.
string webUrl	Page link, the share function in the native audio player, the description of the shared card, will also use this value.
string protocol	Audio protocol. The default value is 'http', setting 'hls' can support the playback of live audio using the HLS protocol.

©2013-2022 Tencent Cloud. All rights reserved.

number playbackRate	Playback speed. Range 0.5-2.0, default is 1 (Android requires version 6 or above).
number duration	The length of the current audio (unit: s), only returned when there is a valid src (read-only).
number currentTime	The current playback position of the audio (unit: s), only returned when there is a valid src (read-only).
boolean paused	Whether it is currently paused or stopped (read-only).
number buffered	The time the audio has been buffered, it only guarantees that the content from the current playback time to this point has been buffered (read-only).

Method Set

onCanplay

> The usage of this method is BackgroundAudioManager.onCanplay(function listener).

Function Description: Listens for the event when the background audio enters a playable state. However, it does not guarantee smooth playback afterwards.
Parameters and Description: function listener, the listening function for the event when the background audio enters a playable state.
onEnded

> The usage of this method is BackgroundAudioManager.onEnded(function listener).

Function Description: Listens for the event when the background audio ends naturally.
Parameters and Description: function listener, the listening function for the event when the background audio ends naturally.
onError

> The usage of this method is BackgroundAudioManager.onError(function listener).

Function Description: Listens for the event when the background audio playback encounters an error.

©2013-2022 Tencent Cloud. All rights reserved.

Parameters and Description: function listener, the listening function for the event when the background audio playback encounters an error.

onNext

The usage of this method is BackgroundAudioManager.onNext(function listener).

Function Description: Listens for the event when the user clicks on the next track in the system music playback panel (iOS only).

Parameters and Description: function listener, the listening function for the event when the user clicks on the next track in the system music playback panel.

onPause

The usage of this method is BackgroundAudioManager.onPause(function listener).

Function Description: Listens for the event when the background audio is paused.

Parameters and Description: function listener, the listening function for the event when the background audio is paused.

onPlay

The usage of this method is BackgroundAudioManager.onPlay(function listener).

Function Description: Listens for the event when the background audio is played.

Parameters and Description: function listener, the listening function for the event when the background audio is played.

onPrev

The usage of this method is BackgroundAudioManager.onPrev(function listener).

Function Description: Listens for the event when the user clicks on the previous track in the system music playback panel (iOS only).

Parameters and Description: function listener, the listening function for the event when the user clicks on the previous track in the system music playback panel.

onSeeked

©2013-2022 Tencent Cloud. All rights reserved.

The usage of this method is BackgroundAudioManager.onSeeked(function listener).

Function Description: Listens for the event when the background audio completes the jump operation.
Parameters and Description: function listener, the listening function for the event when the background audio completes the jump operation.
onSeeking

The usage of this method is BackgroundAudioManager.onSeeking(function listener).

Function Description: Listens for the event when the background audio begins the jump operation.
Parameters and Description: function listener, the listening function for the event when the background audio begins the jump operation.
onStop

The usage of this method is BackgroundAudioManager.onStop(function listener).

Function Description: Listens for the background audio stop event.
Parameters and Description: function listener, the listening function for the background audio stop event.
onTimeUpdate

The usage of this method is BackgroundAudioManager.onTimeUpdate(function listener).

Function Description: Listens for the background audio playback progress update event, only callbacks when the mini program is in the foreground.
Parameters and Description: function listener, the listening function for the background audio playback progress update event.
onWaiting

The usage of this method is BackgroundAudioManager.onWaiting(function listener).

©2013-2022 Tencent Cloud. All rights reserved.

Function Description: Listens for the audio loading event. It is triggered when the audio needs to stop and load due to insufficient data.
Parameters and Description: function listener, the listening function for the audio loading event.
pause

> The usage of this method is BackgroundAudioManager.pause().

Function Description: Pauses the music.
Error Code

Error Codes	Error Message	Note
10001	-	System error
10002	-	Network Error
10003	-	File error, please verify if the response header is missing the Content-Length.
10004	-	Invalid format
-1	-	An unknown error occurred.

play

> The method is utilized as BackgroundAudioManager.play().

Function Description: Plays music.
seek

> The method is employed as BackgroundAudioManager.seek(number currentTime).

Function Description: Navigate to a specified location.
Parameters and Description: number currentTime, the position to navigate to, measured in seconds. Accurate to three decimal places, thus supporting millisecond-level precision.
Error Code

©2013-2022 Tencent Cloud. All rights reserved.

Error Codes	Error Message	Note
10001	-	System error
10002	-	Network Error
10003	-	File error, please verify if the response header is missing the Content-Length.
10004	-	Invalid format
-1	-	An unknown error occurred.

stop

> The method is implemented as BackgroundAudioManager.stop().

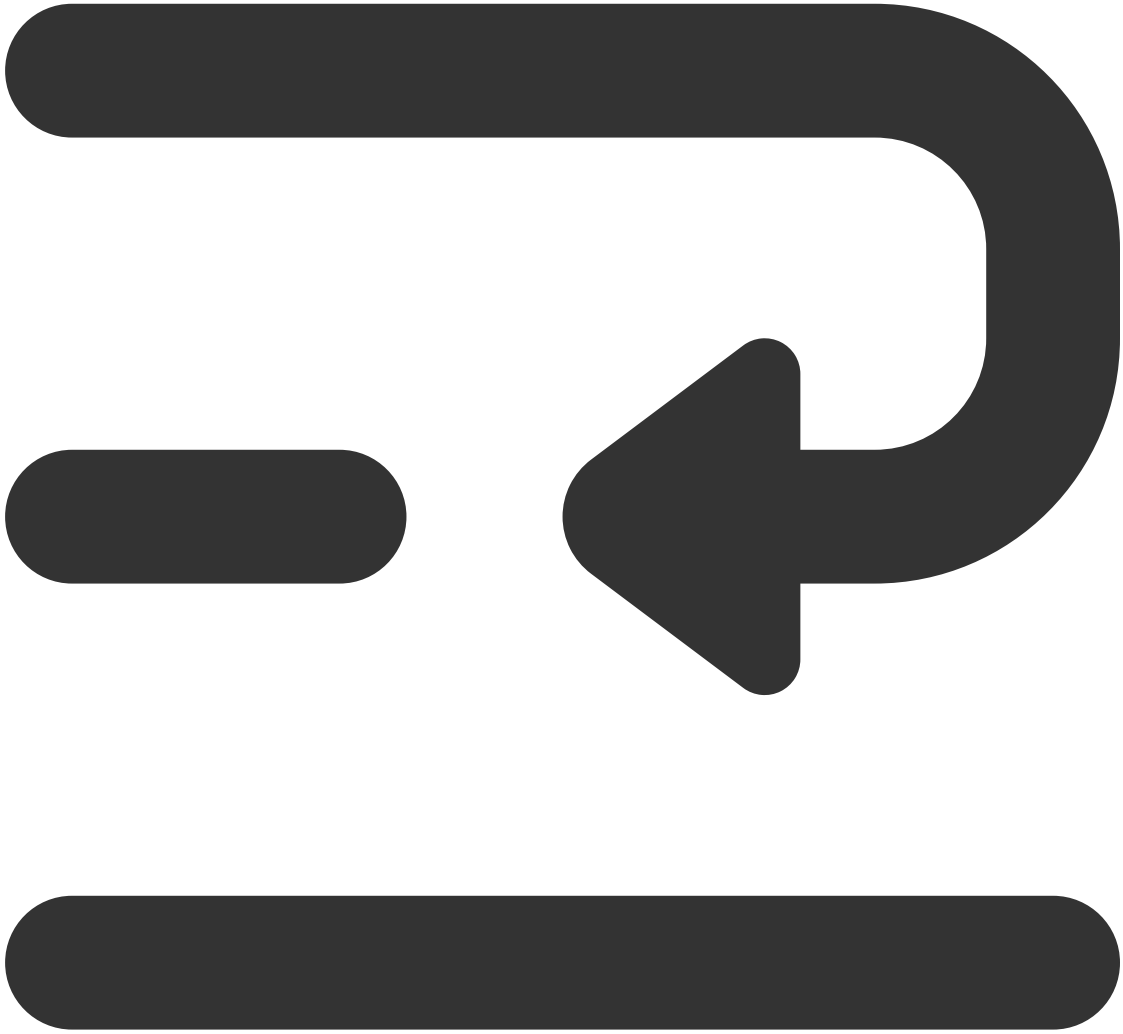
Function Description: Halts music.

Error Code

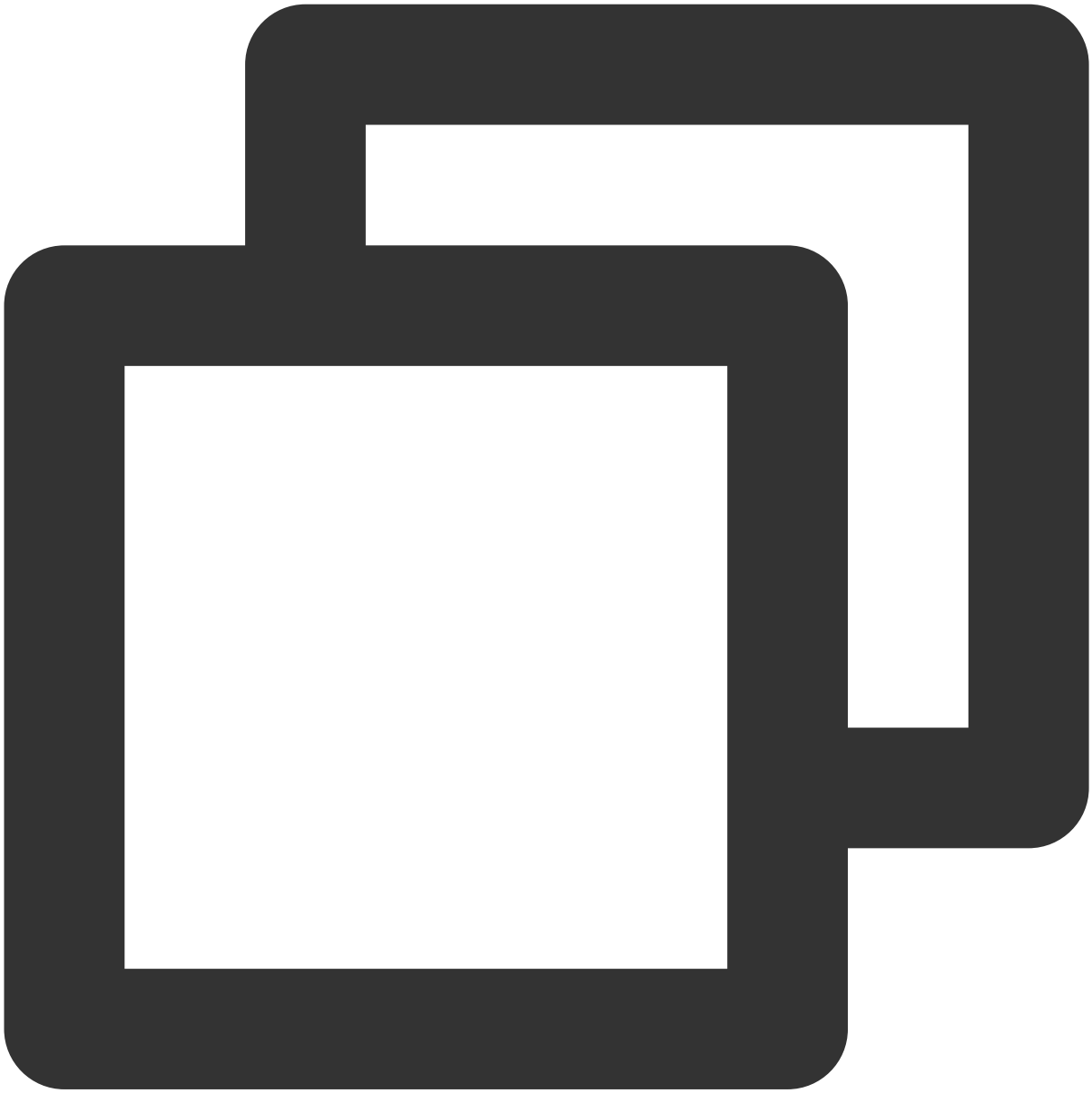
Error Codes	Error Message	Note
10001	-	System error
10002	-	Network Error
10003	-	File error, please verify if the response header is missing the Content-Length.
10004	-	Invalid format
-1	-	An unknown error occurred.

Sample code

©2013-2022 Tencent Cloud. All rights reserved.



©2013-2022 Tencent Cloud. All rights reserved.



```
const backgroundAudioManager = wx.getBackgroundAudioManager()

backgroundAudioManager.title = 'This Very Moment'
backgroundAudioManager.epname = 'This Very Moment'
backgroundAudioManager.singer = 'Xu Wei'
backgroundAudioManager.coverImgUrl = 'http://y.gtimg.cn/music/photo_new/T002R300x30
// After setting the src, it will play automatically.
backgroundAudioManager.src = 'http://ws.stream.qqmusic.qq.com/M500001VfvsJ21xFqb.mp
```

©2013-2022 Tencent Cloud. All rights reserved.

Map

Last updated：2024-07-12 19:15:28

createMapContext

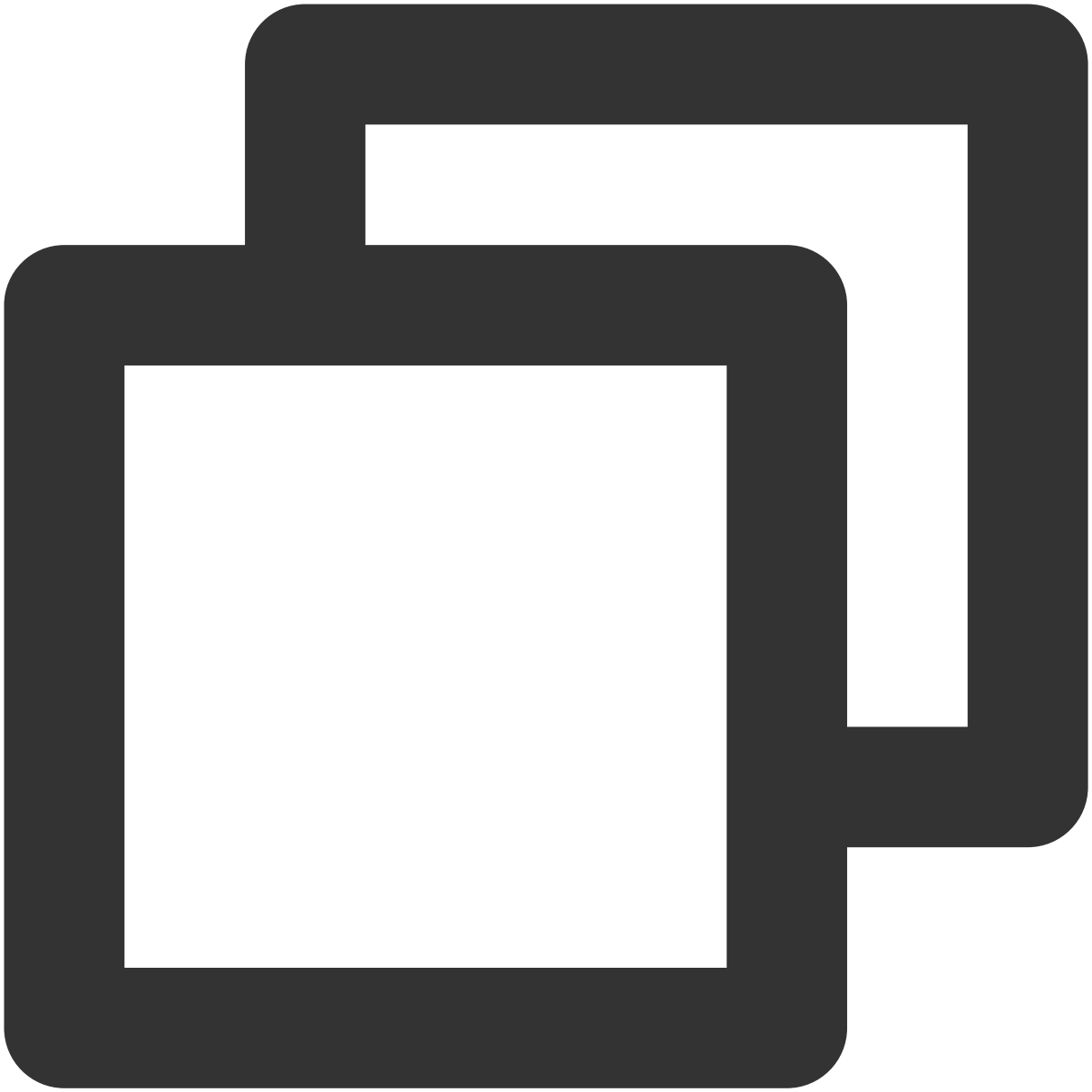
This API is used via wx.createMapContext(string mapId, Object this).

Feature Description: Creates a MapContext object.
Parameter and Description: string mapId, ID of the map component; Object this, "this" of the current component instance under custom components, used to operate the map component within.
Return Value: MapContext.

MapContext

Note:
Unless otherwise specified, the support level for MapContext methods by mobile manufacturers is assumed to be:
System Map (only supported by iOS): Yes;
Google Map (Supported by Android, IDE): Yes;
Huawei Map (only supported by Android): Yes;
Tencent Map: Yes;
Baidu Map: Yes;
Amap: Yes.
MapContext instance, obtainable via wx.createMapContext.
MapContext is bound to a map component via an id, thereby manipulating the corresponding <map> component.
Sample Code

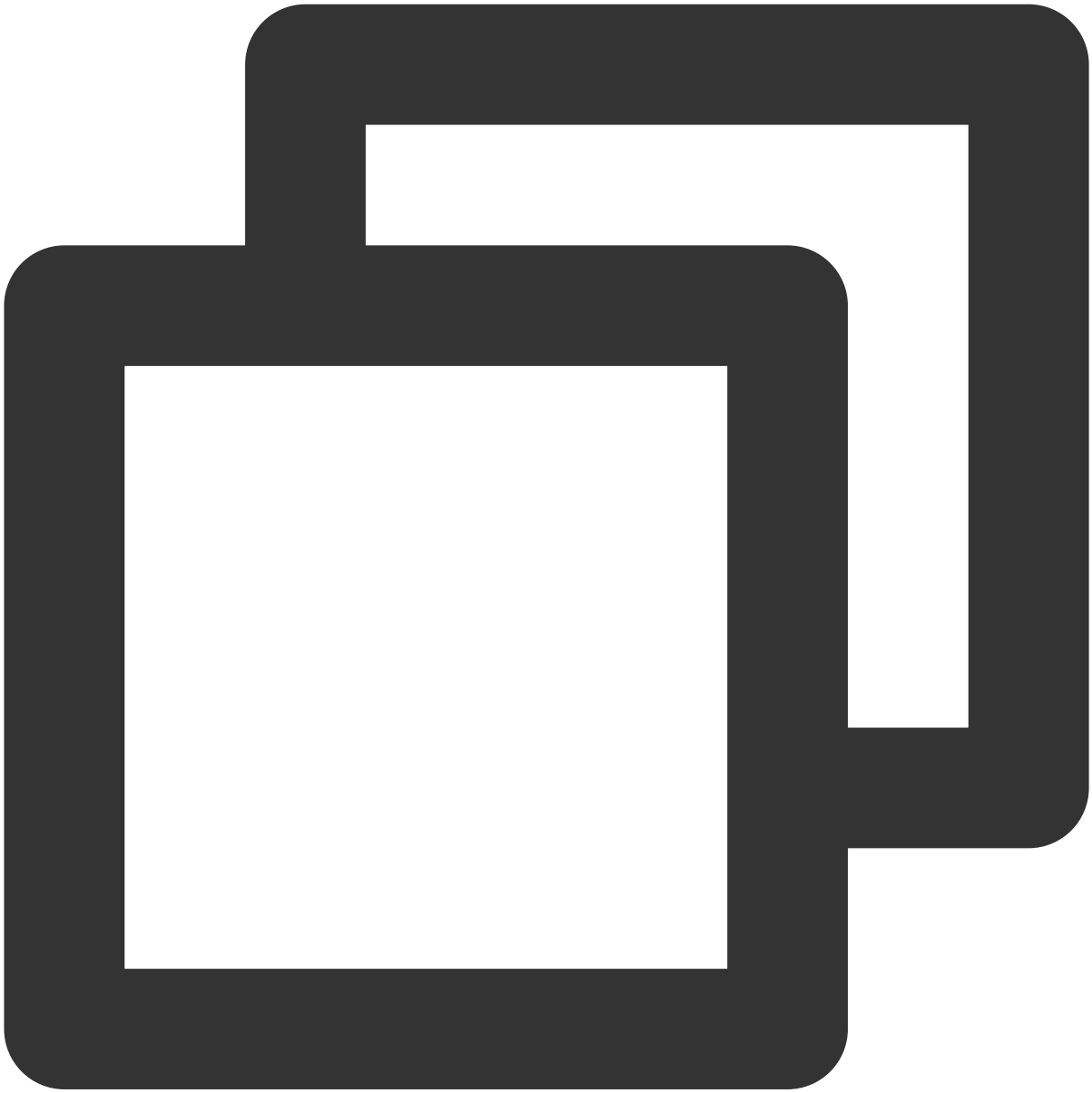
©2013-2022 Tencent Cloud. All rights reserved.



```
<!-- map.wxml -->
<map id="myMap" show-location />

<button type="primary" bindtap="getCenterLocation">Get location</button>
<button type="primary" bindtap="moveToLocation">Change location</button>
<button type="primary" bindtap="translateMarker">Move Marker</button>
<button type="primary" bindtap="includePoints">Scale the field of view to display a
```

©2013-2022 Tencent Cloud. All rights reserved.



```
// map.js
Page({
  onReady: function (e) {
    // Use wx.createMapContext to get map context.
    this.mapCtx = wx.createMapContext('myMap')
  },
  getCenterLocation: function () {
    this.mapCtx.getCenterLocation({
      success: function(res){
        console.log(res.longitude)
        console.log(res.latitude)
```

©2013-2022 Tencent Cloud. All rights reserved.

```
        }
      })
    },
    moveToLocation: function () {
      this.mapCtx.moveToLocation()
    },
    translateMarker: function() {
      this.mapCtx.translateMarker({
        markerId: 0,
        autoRotate: true,
        duration: 1000,
        destination: {
          latitude:23.10229,
          longitude:113.3345211,
        },
        animationEnd() {
          console.log('animation end')
        }
      })
    },
    includePoints: function() {
      this.mapCtx.includePoints({
        padding: [10],
        points: [{
          latitude:23.10229,
          longitude:113.3345211,
        }, {
          latitude:23.00229,
          longitude:113.3345211,
        }]
      })
    }
  })
```

.addArc

The method is used via MapContext.addArc(Object object).

Note：

Support: Not supported.

Feature Description: An arc line is added, and either a waypoint or an angle must be set. The waypoint must be within the valid coordinate range of the start and end points, otherwise a correct arc line cannot be generated. When

©2013-2022 Tencent Cloud. All rights reserved.

setting the angle, the angle is prioritized. The angle is defined as the angle rotated counterclockwise from the starting point to the end point, tangent to the starting point. This feature is temporarily unsupported on the tool side.

Parameter and Description: Object

Attribute	Type	Default value	Required	Description
id	number	-	Yes	Arc ID
start	Object	-	Yes	Start point
end	Object	-	Yes	End point
pass	Object	-	No	Waypoint
angle	number	0	No	Included Angle
width	number	5	No	Line Width
color	number	#000000	No	Color of the line
success	function	-	No	Callback Function of Successful Interface Call
fail	function	-	No	Callback Function of Failing Interface Call
complete	function	-	No	Callback function executed upon the completion of the interface invocation (both successful and unsuccessful invocations)

Structure attributes of "start"

Structure attributes	Type	Default value	Required	Description
longitude	number	-	Yes	Longitude
latitude	number	-	Yes	Latitude

Structure attributes of "end"

Structure attributes	Type	Default value	Required	Description
longitude	number	-	Yes	Longitude
latitude	number	-	Yes	Latitude

©2013-2022 Tencent Cloud. All rights reserved.

Structure attributes of "pass"

Structure attributes	Type	Default value	Required	Description
longitude	number	-	Yes	Longitude
latitude	number	-	Yes	Latitude

.addCustomLayer

The method is used via : MapContext.addCustomLayer(Object object)

Note：

Support: Not supported.

Feature Description: Adds a personalized layer. For details about layer creation, see the documentation.

Parameter and Description: Object

Attribute	Type	Default value	Required	Description
layerId	string	-	Yes	ID of personalized layer.
success	function	-	No	Callback Function of Successful Interface Call
fail	function	-	No	Callback Function of Failing Interface Call
complete	function	-	No	Callback function executed upon the completion of the interface invocation (both successful and unsuccessful invocations)

.addGroundOverlay

The method is used via MapContext.addGroundOverlay(Object object).

Note：

Support: supported by IDE, not supported by Android and ios.

Feature Description: Creates a custom image layer, which scales in accordance with the map's zoom level.

Parameter and Description: Object

Attribute	Type	Default	Required	Description

©2013-2022 Tencent Cloud. All rights reserved.

		value		
id	String	-	Yes	ID of image layer
src	String	-	Yes	Image path, supporting network images, temporary paths, and code package paths.
bounds	Object	-	Yes	Longitude and latitude range covered by the image
visible	Boolean	true	No	Visible or not
zIndex	Number	1	No	Layer Rendering Sequence
opacity	Number	1	No	Layer Transparency
success	function	-	No	Callback Function of Successful Interface Call
fail	function	-	No	Callback Function of Failing Interface Call
complete	function	-	No	Callback function executed upon the completion of the interface invocation (both successful and unsuccessful invocations)

Structure attributes of "bounds"

Structure attributes	Type	Default value	Required	Description
southwest	Object	-	Yes	Southwest corner's longitude and latitude
northeast	Object	-	Yes	Northeast corner's longitude and latitude

Structure attributes of "southwest"

Structure attributes	Type	Default value	Required	Description
longitude	number	-	Yes	Longitude
latitude	number	-	Yes	Latitude

©2013-2022 Tencent Cloud. All rights reserved.

Structure attributes of "northeast"

Structure attributes	Type	Default value	Required	Description
longitude	number	-	Yes	Longitude
latitude	number	-	Yes	Latitude

.addMarkers

The method is used via MapContext.addMarkers(Object object).

Feature Description: Adds a marker.

Parameter and Description: Object.

Attribute	Type	Default value	Required	Description
markers	array	-	Yes	Same as the marker attribute passed into the map component.
clear	boolean	false	No	Whether to clear all markers on the map first.
success	function	-	No	Callback Function of Successful Interface Call
fail	function	-	No	Callback Function of Failing Interface Call
complete	function	-	No	Callback function executed upon the completion of the interface invocation (both successful and unsuccessful invocations)

.eraseLines

The method is used via MapContext.eraseLines(Object object).

Feature Description: Erases or grays out the lines that have been added to the map.

Parameter and Description: Object

Attribute	Type	Default value	Required	Description

©2013-2022 Tencent Cloud. All rights reserved.

lines	Array.<Object>	-	Yes	Array of line segments to be erased. See polyline attribute for details.
success	function	-	No	Callback Function of Successful Interface Call
fail	function	-	No	Callback Function of Failing Interface Call
complete	function	-	No	Callback function executed upon the completion of the interface invocation (both successful and unsuccessful invocations)

Structure attributes of "lines"

Structure attributes	Type	Default value	Required	Description
id	number	-	Yes	Line segment ID
index	number	-	Yes	Specifies a section of the line segment, with the starting point index of the line being 0.
point	Object	-	Yes	Specifies a point within a section of the line segment.
clear	boolean	true	No	True for erasure, false for graying out.

Structural attributes of a "point".

Structure attributes	Type	Default value	Required	Description
longitude	number	-	Yes	Longitude
latitude	number	-	Yes	Latitude

.fromScreenLocation

The method is used via MapContext.fromScreenLocation(Object object).

Feature Description: Gets the corresponding latitude and longitude of a point on the screen, with the coordinate origin at the top left corner of the map.

Parameter and Description: Object

©2013-2022 Tencent Cloud. All rights reserved.

Attribute	Type	Default value	Required	Description
x	number	-	Yes	X-coordinate value.
y	number	-	Yes	Y-coordinate value.
success	function	-	No	Callback Function of Successful Interface Call
fail	function	-	No	Callback Function of Failing Interface Call
complete	function	-	No	Callback function executed upon the completion of the interface invocation (both successful and unsuccessful invocations)

Parameters for object.success callback function: Object res.

Attribute	Type	Description
latitude	number	Latitude
longitude	number	Longitude

.getCenterLocation

This method is used via MapContext.getCenterLocation(Object object).

Feature Description: Gets the latitude and longitude of the current map center. The return is in the gcj02 coordinate system, which can be used for wx.openLocation().

Parameter and Description: Object

Attribute	Type	Default value	Required	Description
iconPath	string	-	No	Image path, supporting network paths, local paths, and code package paths.
success	function	-	No	Callback Function of Successful Interface Call
fail	function	-	No	Callback Function of Failing Interface Call
complete	function	-	No	Callback function executed upon the completion of the interface invocation (both successful and unsuccessful invocations)

©2013-2022 Tencent Cloud. All rights reserved.

Parameters for object.success callback function: Object res.

Attribute	Type	Description
longitude	number	Longitude
latitude	number	Latitude

.getRegion

This method is used via MapContext.getRegion(Object object).

Feature Description: Gets the field of view range of the current map.

Parameter and Description: Object

Attribute	Type	Default value	Required	Description
success	function	-	No	Callback Function of Successful Interface Call
fail	function	-	No	Callback Function of Failing Interface Call
complete	function	-	No	Callback function executed upon the completion of the interface invocation (both successful and unsuccessful invocations)

Parameters for object.success callback function: Object res.

Attribute	Type	Description
southwest	object	Southwest corner's longitude and latitude
northeast	object	Northeast corner's longitude and latitude

Structure attributes of "southwest"

Structure attributes	Type	Description
longitude	number	Longitude
latitude	number	Latitude

©2013-2022 Tencent Cloud. All rights reserved.

Structure attributes of "northeast"

Structure attributes	Type	Description
longitude	number	Longitude
latitude	number	Latitude

.getRotate

This method is used via MapContext.getRotate(Object object).

Feature Description: Gets the rotation angle of the current map.

Parameter and Description: Object

Attribute	Type	Default value	Required	Description
success	function	-	No	Callback Function of Successful Interface Call
fail	function	-	No	Callback Function of Failing Interface Call
complete	function	-	No	Callback function executed upon the completion of the interface invocation (both successful and unsuccessful invocations)

Parameters for object.success callback function: Object res.

Attribute	Type	Description
rotate	number	Rotation Angle

.getScale

This method is used via MapContext.getScale(Object object).

Feature Description: Gets the scaling level of the current map.

Parameter and Description: Object

Attribute	Type	Default value	Required	Description

©2013-2022 Tencent Cloud. All rights reserved.

success	function	-	No	Callback Function of Successful Interface Call
fail	function	-	No	Callback Function of Failing Interface Call
complete	function	-	No	Callback function executed upon the completion of the interface invocation (both successful and unsuccessful invocations)

Parameters for object.success callback function: Object res.

Attribute	Type	Description
scale	number	Scaling value

.getSkew

This method is used via MapContext.getSkew(Object object).

Feature Description: Gets the tilt angle of the current map.

Parameter and Description: Object

Attribute	Type	Default value	Required	Description
success	function	-	No	Callback Function of Successful Interface Call
fail	function	-	No	Callback Function of Failing Interface Call
complete	function	-	No	Callback function executed upon the completion of the interface invocation (both successful and unsuccessful invocations)

Parameters for object.success callback function: Object res.

Attribute	Type	Description
skew	number	Tilt Angle

.includePoints

The method is used via MapContext.includePoints(Object object).

©2013-2022 Tencent Cloud. All rights reserved.

Feature Description: Scales the view field to display all latitudes and longitudes.

Parameter and Description: Object

Attribute	Type	Default value	Required	Description
points	array.	-	Yes	List of coordinate points to be displayed within the visible area.
padding	array.	-	No	The distance from the edges of the rectangle formed by the coordinate points to the edge of the map, measured in pixels. The format is [top, right, bottom, left]. On Android, only the first item in the array can be recognized, and the padding for top, bottom, left, and right is consistent. The developer tool does not currently support the padding parameter.
success	function	-	No	Callback Function of Successful Interface Call
fail	function	-	No	Callback Function of Failing Interface Call
complete	function	-	No	Callback function executed upon the completion of the interface invocation (both successful and unsuccessful invocations)

Structure attributes of "points"

Structure attributes	Type	Default value	Required	Description
longitude	number	-	Yes	Longitude
latitude	number	-	Yes	Latitude

.initMarkerCluster

The method is used via MapContext.initMarkerCluster(Object object).

Feature Description: Initializes the configuration for point clustering. The default configuration is used if this feature is not called.

Parameter and Description: Object

Attribute	Type	Default value	Required	Description

©2013-2022 Tencent Cloud. All rights reserved.

enableDefaultStyle	boolean	true	No	Enable the default clustering style.
zoomOnClick	boolean	true	No	Whether to implement cluster separation when a clustered marker point is clicked.
gridSize	number	60	No	The aggregatable distance (measured in pixels) for the clustering algorithm, that is, points closer than this value will be clustered together.
success	function	-	No	Callback Function of Successful Interface Call
fail	function	-	No	Callback Function of Failing Interface Call
complete	function	-	No	Callback function executed upon the completion of the interface invocation (both successful and unsuccessful invocations)

.moveAlong

This method is used via MapContext.moveAlong(Object object).

Feature Description: Moves the **marker** along a specified path, suitable for scenarios such as trajectory playback. A callback event is triggered upon animation completion. If the animation is in progress and the **moveAlong** method is called again for the same **marker**, the previous animation will be interrupted.

Parameter and Description: Object

Attribute	Type	Default value	Required	Description
markerId	number	-	Yes	Specify a marker.
path	array	-	Yes	The coordinate string of the moving path, with the coordinate point format being `{longitude, latitude}` .
autoRotate	boolean	true	No	Automatically adjust the rotation angle of the marker based on the path direction.
duration	number	-	Yes	The duration of smooth movement.
success	function	-	No	Callback Function of Successful Interface Call
fail	function	-	No	Callback Function of Failing Interface Call

©2013-2022 Tencent Cloud. All rights reserved.

complete	function	-	No	Callback function executed upon the completion of the interface invocation (both successful and unsuccessful invocations)

.moveToLocation

This method is used via MapContext.moveToLocation(Object object).

Feature Description: Moves the map center to the current location point, when the map component "show-location" must be set to true.

Parameter and Description: Object

Attribute	Type	Default value	Required	Description
longitude	number	-	No	Longitude
latitude	number	-	No	Latitude
success	function	-	No	Callback Function of Successful Interface Call
fail	function	-	No	Callback Function of Failing Interface Call
complete	function	-	No	Callback function executed upon the completion of the interface invocation (both successful and unsuccessful invocations)

.on

This method is used via MapContext.on(string event, function callback).

Feature Description: Monitors map events.

Parameter 1: visualLayerEvent, a unified callback exit for the visual layer, supported from version 1.5.16.

Parameter	Type	Description
layerId	String	Layer ID
eventType	String	Event type
eventInfo	String	Event information

©2013-2022 Tencent Cloud. All rights reserved.

Parameter 2: markerClusterCreate, triggered when a new cluster is generated due to scaling or dragging, only returning information about the newly created cluster.

Parameter	Type	Description
clusters	Array<ClusterInfo>	Cluster data

Parameter 3: markerClusterClick, click events of the cluster.

Parameter	Type	Description
cluster	ClusterInfo	Cluster

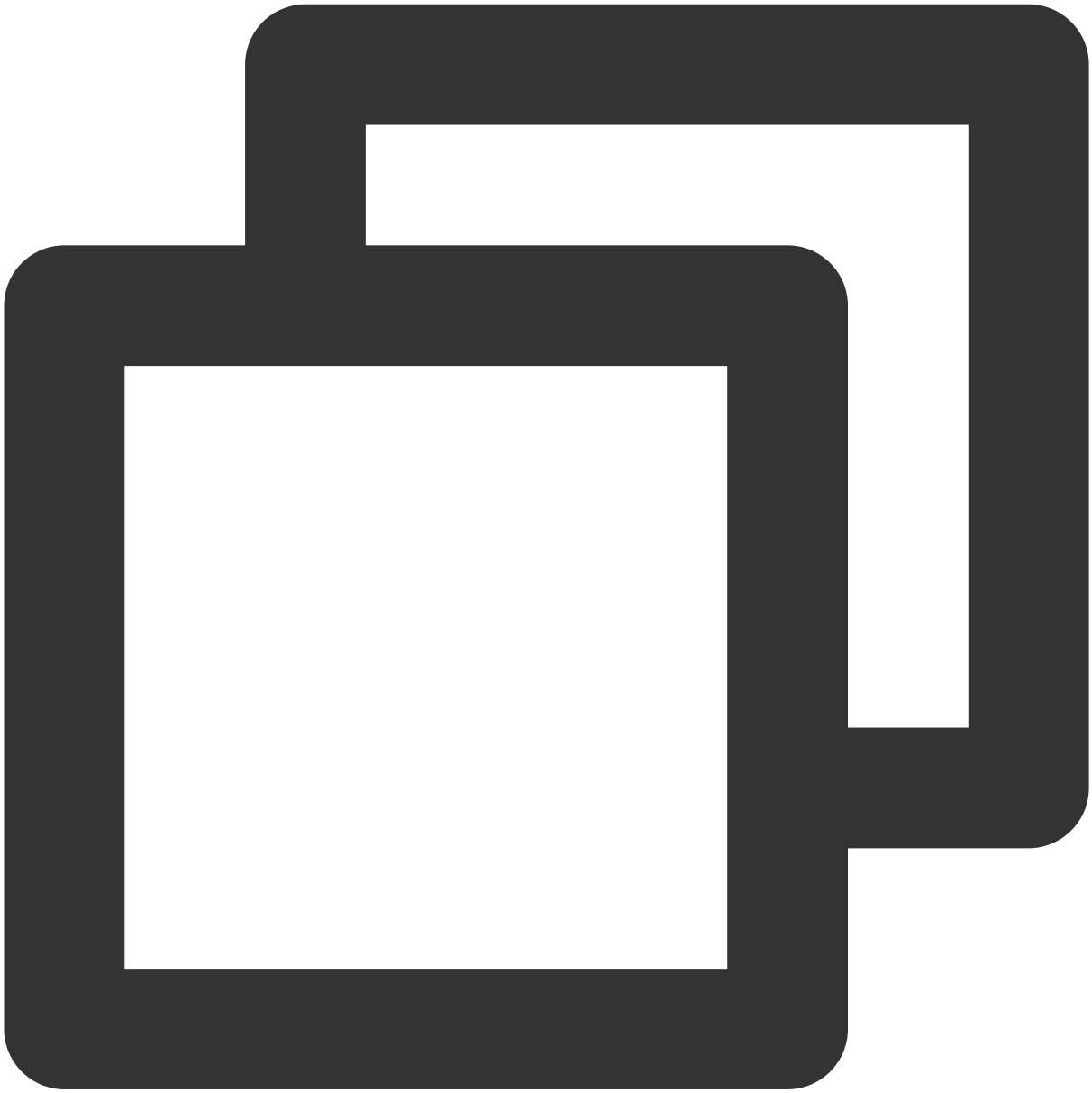
ClusterInfo Structure

Parameter	Type	Description
clusterId	Number	Cluster ID
center	LatLng	Cluster coordinates
markerIds	Array<Number>	Array of point marker data within the aggregation cluster

Parameter 4: string event, the name of the event; function callback, the callback function for the event.
The valid values for the event include markerClusterCreate, markerClusterClick, and visualLayerEvent.

Sample Code

©2013-2022 Tencent Cloud. All rights reserved.



```
MapContext.on('visualLayerEvent', (res) => {})
MapContext.on('markerClusterCreate', (res) => {})
MapContext.on('markerClusterClick', (res) => {})
```

.openMapApp

This method is used via MapContext.openMapApp(Object object).

©2013-2022 Tencent Cloud. All rights reserved.

Feature Description: Launches the map App for navigation.

Parameter and Description: Object

Attribute	Type	Default value	Required	Description
longitude	number	-	Yes	Destination Longitude
latitude	number	-	Yes	Destination Latitude
destination	string	-	Yes	Destination Name
success	function	-	No	Callback Function of Successful Interface Call
fail	function	-	No	Callback Function of Failing Interface Call
complete	function	-	No	Callback function executed upon the completion of the interface invocation (both successful and unsuccessful invocations)

.removeArc

This method is used via MapContext.removeArc(Object object).

Note：

Support: Not supported.

Feature Description: Removes the arc line.

Parameter and Description: Object

Attribute	Type	Default value	Required	Description
id	number	-	Yes	Arc ID
success	function	-	No	Callback Function of Successful Interface Call
fail	function	-	No	Callback Function of Failing Interface Call
complete	function	-	No	Callback function executed upon the completion of the interface invocation (both successful and unsuccessful invocations)

.removeCustomLayer

©2013-2022 Tencent Cloud. All rights reserved.

The method is used via MapContext.removeCustomLayer(Object object).

Note：

Support: Not supported.

Feature Description: Removes personalized image layers.

Parameter and Description: Object

Attribute	Type	Default value	Required	Description
layerId	string	-	Yes	ID of personalized layer.
success	function	-	No	Callback Function of Successful Interface Call
fail	function	-	No	Callback Function of Failing Interface Call
complete	function	-	No	Callback function executed upon the completion of the interface invocation (both successful and unsuccessful invocations)

.removeGroundOverlay

This method is used via MapContext.removeGroundOverlay(Object object).

Note：

Support: supported by IDE, not supported by Android and ios.

Feature Description: Removes custom image layers.

Parameter and Description: Object

Attribute	Type	Default value	Required	Description
id	String	-	Yes	ID of image layer
success	function	-	No	Callback Function of Successful Interface Call
fail	function	-	No	Callback Function of Failing Interface Call
complete	function	-	No	Callback function executed upon the completion of the interface invocation (both successful and unsuccessful invocations)

©2013-2022 Tencent Cloud. All rights reserved.

.removeMarkers

This method is used via MapContext.removeMarkers(Object object).

Feature Description: Removes markers.

Parameter and Description: Object

Attribute	Type	Default value	Required	Description
markerIds	array	-	Yes	Collection of marker IDs.
success	function	-	No	Callback Function of Successful Interface Call
fail	function	-	No	Callback Function of Failing Interface Call
complete	function	-	No	Callback function executed upon the completion of the interface invocation (both successful and unsuccessful invocations)

.setBoundary

This method is used via MapContext.setBoundary(Object object).

Feature Description: Restricts the map's display range. This interface will also limit the map's minimum scaling integer level.

Parameter and Description: Object

Attribute	Type	Default value	Required	Description
southwest	Object	-	Yes	Southwest corner's longitude and latitude
northeast	Object	-	Yes	Northeast corner's longitude and latitude
success	function	-	No	Callback Function of Successful Interface Call
fail	function	-	No	Callback Function of Failing Interface Call
complete	function	-	No	Callback function executed upon the completion of the interface invocation (both successful and unsuccessful invocations)

©2013-2022 Tencent Cloud. All rights reserved.

Structure attributes of "southwest"

Structure attributes	Type	Default value	Required	Description
longitude	number	-	Yes	Longitude
latitude	number	-	Yes	Latitude

Structure attributes of "northeast"

Structure attributes	Type	Default value	Required	Description
longitude	number	-	Yes	Longitude
latitude	number	-	Yes	Latitude

.setCenterOffset

This method is used via MapContext.setCenterOffset(Object object).

Feature Description: Sets the offset of the map center point, with growth towards the back and downwards, within a screen ratio range of (0.25 - 0.75), default offset being [0.5, 0.5].

Parameter and Description: Object

Attribute	Type	Default value	Required	Description
offset	Array.<number>	-	Yes	Offset value, a two-element array.
success	function	-	No	Callback Function of Successful Interface Call
fail	function	-	No	Callback Function of Failing Interface Call
complete	function	-	No	Callback function executed upon the completion of the interface invocation (both successful and unsuccessful invocations)

.setLocMarkerIcon

©2013-2022 Tencent Cloud. All rights reserved.

This method is used via MapContext.setLocMarkerIcon(Object object).

Feature Description: Sets the location point icon, supporting network paths, local paths, and code package paths.

Parameter and Description: Object

Attribute	Type	Default value	Required	Description
iconPath	string	-	No	Image path, supporting network paths, local paths, and code package paths.
success	function	-	No	Callback Function of Successful Interface Call
fail	function	-	No	Callback Function of Failing Interface Call
complete	function	-	No	Callback function executed upon the completion of the interface invocation (both successful and unsuccessful invocations)

.toScreenLocation

The method is used via MapContext.toScreenLocation(Object object).

Feature Description: Gets the screen coordinates corresponding to the latitude and longitude, with the coordinate origin at the top left corner of the map.

Parameter and Description: Object

Attribute	Type	Default value	Required	Description
latitude	Number	-	Yes	Latitude
longitude	Number	-	Yes	Longitude
success	function	-	No	Callback Function of Successful Interface Call
fail	function	-	No	Callback Function of Failing Interface Call
complete	function	-	No	Callback function executed upon the completion of the interface invocation (both successful and unsuccessful invocations)

©2013-2022 Tencent Cloud. All rights reserved.

Parameters for object.success callback function: Object res.

Attribute	Type	Description
x	number	X-coordinate value.
y	number	Y-coordinate value.

.translateMarker

The method is used via MapContext.translateMarker(Object object).

Feature Description: Translates the marker, with iOS and IDE supporting animation effects. Android platform does not currently support animation effects.

Parameter and Description: Object

Attribute	Type	Default value	Required	Description
markerId	number	-	Yes	Specify a marker.
destination	object	-	Yes	Specify the target point for the marker to move to.
animationEnd	function	-	No	Animation completion callback function
success	function	-	No	Callback Function of Successful Interface Call
fail	function	-	No	Callback Function of Failing Interface Call
complete	function	-	No	Callback function executed upon the completion of the interface invocation (both successful and unsuccessful invocations)

Structure attributes of "destination"

Structure attributes	Type	Default value	Required	Description
longitude	number	-	Yes	Longitude
latitude	number	-	Yes	Latitude

.updateGroundOverlay

©2013-2022 Tencent Cloud. All rights reserved.

This method is used via MapContext.updateGroundOverlay(Object object).

Note：

Support: supported by IDE, not supported by Android and ios.

Feature Description: Updates custom image layers.

Parameter and Description: Object

Attribute	Type	Default value	Required	Description
id	String	-	Yes	ID of image layer
src	String	-	Yes	Image path, supporting network images, temporary paths, and code package paths.
bounds	Object	-	Yes	Longitude and latitude range covered by the image
visible	Boolean	true	No	Visible or not
zIndex	Number	1	No	Layer Rendering Sequence
opacity	Number	1	No	Layer Transparency
success	function	-	No	Callback Function of Successful Interface Call
fail	function	-	No	Callback Function of Failing Interface Call
complete	function	-	No	Callback function executed upon the completion of the interface invocation (both successful and unsuccessful invocations)

Structure attributes of "bounds"

Structure attributes	Type	Default value	Required	Description
southwest	Object	-	Yes	Southwest corner's longitude and latitude
northeast	Object	-	Yes	Northeast corner's longitude and latitude

©2013-2022 Tencent Cloud. All rights reserved.

Structure attributes of "southwest"

Structure attributes	Type	Default value	Required	Description
longitude	number	-	Yes	Longitude
latitude	number	-	Yes	Latitude

Structure attributes of "northeast"

Structure attributes	Type	Default value	Required	Description
longitude	number	-	Yes	Longitude
latitude	number	-	Yes	Latitude

©2013-2022 Tencent Cloud. All rights reserved.

Real-time audio/video

Last updated：2024-07-12 19:14:36

createLivePusherContext

This method is used via LivePusherContext wx.createLivePusherContext().

Note:

Support for this feature starts from the base library 1.4.96. Compatibility measures are required for earlier versions.

Feature Description: Creates a LivePusherContext object.

Return Value: LivePusherContext.

LivePusherContext

LivePusherContext instance, obtainable via wx.createLivePusherContext.

The LivePusherContext is uniquely bound within the page to the live-pusher component, operating the corresponding live-pusher component.

.start

This method is used via LivePusherContext.start(Object object).

Feature Description: Starts streaming while simultaneously activating camera preview.

Parameter and Description: Object.

Attribute	Type	Default value	Required	Description
success	Function	-	No	Callback Function of Successful Interface Call
fail	Function	-	No	Callback Function of Failing Interface Call
complete	Function	-	No	Callback function executed upon the completion of the interface invocation (both successful and unsuccessful invocations)

©2013-2022 Tencent Cloud. All rights reserved.

.stop

This method is used via LivePusherContext.stop(Object object).

Feature Description: Stops streaming and concurrently camera preview.

Parameter and Description: Object.

Attribute	Type	Default value	Required	Description
success	Function	-	No	Callback Function of Successful Interface Call
fail	Function	-	No	Callback Function of Failing Interface Call
complete	Function	-	No	Callback function executed upon the completion of the interface invocation (both successful and unsuccessful invocations)

.pause

This method is used via LivePlayerContext.pause(Object object).

Feature Description: Pauses streaming.

Parameter and Description: Object.

Attribute	Type	Default value	Required	Description
success	Function	-	No	Callback Function of Successful Interface Call
fail	Function	-	No	Callback Function of Failing Interface Call
complete	Function	-	No	Callback function executed upon the completion of the interface invocation (both successful and unsuccessful invocations)

.resume

This method is used via LivePusherContext.resume(Object object).

©2013-2022 Tencent Cloud. All rights reserved.

Feature Description: Resumes streaming.

Parameter and Description: Object.

Attribute	Type	Default value	Required	Description
success	Function	-	No	Callback Function of Successful Interface Call
fail	Function	-	No	Callback Function of Failing Interface Call
complete	Function	-	No	Callback function executed upon the completion of the interface invocation (both successful and unsuccessful invocations)

.switchCamera

This method is used via LivePusherContext.switchCamera(Object object).

Feature Description: Switches between front and rear cameras.

Parameter and Description: Object.

Attribute	Type	Default value	Required	Description
success	Function	-	No	Callback Function of Successful Interface Call
fail	Function	-	No	Callback Function of Failing Interface Call
complete	Function	-	No	Callback function executed upon the completion of the interface invocation (both successful and unsuccessful invocations)

.snapshot

This method is used via LivePusherContext.snapshot(Object object).

Feature Description: Snapshot.

Parameter and Description: Object.

Attribute	Type	Default value	Required	Description

©2013-2022 Tencent Cloud. All rights reserved.

success	Function	-	No	Callback Function of Successful Interface Call
fail	Function	-	No	Callback Function of Failing Interface Call
complete	Function	-	No	Callback function executed upon the completion of the interface invocation (both successful and unsuccessful invocations)

object.success callback function: arguments Object res

Attribute	Type	Description
tempImagePath	string	Temporary path for image files
width	string	Width of the image
height	string	height of the image

.toggleTorch

This method is used via LivePusherContext.toggleTorch(Object object).

Feature Description: Switches to flashlight.

Parameter and Description: Object.

Attribute	Type	Default value	Required	Description
success	Function	-	No	Callback Function of Successful Interface Call
fail	Function	-	No	Callback Function of Failing Interface Call
complete	Function	-	No	Callback function executed upon the completion of the interface invocation (both successful and unsuccessful invocations)

.playBGM

This method is used via LivePusherContext.playBGM(Object object).

Feature Description: Plays background sound.

©2013-2022 Tencent Cloud. All rights reserved.

Parameter and Description: Object.

Attribute	Type	Default value	Required	Description
url	String	-	Yes	Resource address for adding background mix.
success	Function	-	No	Callback Function of Successful Interface Call
fail	Function	-	No	Callback Function of Failing Interface Call
complete	Function	-	No	Callback function executed upon the completion of the interface invocation (both successful and unsuccessful invocations)

.stopBGM

This method is used via LivePusherContext.stopBGM(Object object).

Feature Description: Stops background sound.

Parameter and Description: Object.

Attribute	Type	Default value	Required	Description
success	Function	-	No	Callback Function of Successful Interface Call
fail	Function	-	No	Callback Function of Failing Interface Call
complete	Function	-	No	Callback function executed upon the completion of the interface invocation (both successful and unsuccessful invocations)

.pauseBGM

This method is used via LivePusherContext.pauseBGM(Object object).

Feature Description: Pauses background sound.

Parameter and Description: Object.

Attribute	Type	Default value	Required	Description

©2013-2022 Tencent Cloud. All rights reserved.

success	Function	-	No	Callback Function of Successful Interface Call
fail	Function	-	No	Callback Function of Failing Interface Call
complete	Function	-	No	Callback function executed upon the completion of the interface invocation (both successful and unsuccessful invocations)

.resumeBGM

This method is used via LivePusherContext.resumeBGM(Object object).

Feature Description: Resumes background sound.

Parameter and Description: Object.

Attribute	Type	Default value	Required	Description
success	Function	-	No	Callback Function of Successful Interface Call
fail	Function	-	No	Callback Function of Failing Interface Call
complete	Function	-	No	Callback function executed upon the completion of the interface invocation (both successful and unsuccessful invocations)

.setBGMPosition

This method is used via LivePusherContext.setBGMPosition(Object object).

Feature Description: Sets the progress of the background sound.

Parameter and Description: Object.

Attribute	Type	Default value	Required	Description
position	Number	-	Yes	Background sound progress, measured in seconds.
success	Function	-	No	Callback Function of Successful Interface Call
fail	Function	-	No	Callback Function of Failing Interface Call

©2013-2022 Tencent Cloud. All rights reserved.

complete	Function	-	No	Callback function executed upon the completion of the interface invocation (both successful and unsuccessful invocations)

.setMICVolume

This method is used via LivePusherContext.setMICVolume(Object object).

Feature Description: Sets the microphone volume.

Parameter and Description: Object.

Attribute	Type	Default value	Required	Description
volume	String	-	Yes	Volume level, ranging from 0 to 1.
success	Function	-	No	Callback Function of Successful Interface Call
fail	Function	-	No	Callback Function of Failing Interface Call
complete	Function	-	No	Callback function executed upon the completion of the interface invocation (both successful and unsuccessful invocations)

.setBGMVolume

This method is used via LivePusherContext.setBGMVolume(Object object).

Feature Description: Sets the background volume.

Parameter and Description: Object.

Attribute	Type	Default value	Required	Description
volume	String	-	Yes	Volume level, ranging from 0 to 1.
success	Function	-	No	Callback Function of Successful Interface Call
fail	Function	-	No	Callback Function of Failing Interface Call
complete	Function	-	No	Callback function executed upon the completion of

©2013-2022 Tencent Cloud. All rights reserved.

				the interface invocation (both successful and unsuccessful invocations)

.setAudioReverbType

This method is used via LivePusherContext.setAudioReverbType(Object object).

Feature Description: Sets the mix type.

Parameter and Description: Object.

Attribute	Type	Default value	Required	Description
audioReverbType	Number	-	Yes	Mix type, 0 to 6 correspond to "Reverb off", "KTV", "Small room", "Large hall", "Deep", "Resonant", "Magnetic", respectively.
success	Function	-	No	Callback Function of Successful Interface Call
fail	Function	-	No	Callback Function of Failing Interface Call
complete	Function	-	No	Callback function executed upon the completion of the interface invocation (both successful and unsuccessful invocations)

.startPreview

This method is used via LivePusherContext.startPreview(Object object).

Feature Description: Enables camera preview.

Parameter and Description: Object.

Attribute	Type	Default value	Required	Description
success	function	-	No	Callback Function of Successful Interface Call
fail	function	-	No	Callback Function of Failing Interface Call
complete	function	-	No	Callback function executed upon the completion of the interface invocation (both successful and

©2013-2022 Tencent Cloud. All rights reserved.

				unsuccessful invocations)

.stopPreview

This method is used via LivePusherContext.stopPreview(Object object).

Feature Description: Disables camera preview.

Parameter and Description: Object.

Attribute	Type	Default value	Required	Description
success	Function	-	No	Callback Function of Successful Interface Call
fail	Function	-	No	Callback Function of Failing Interface Call
complete	Function	-	No	Callback function executed upon the completion of the interface invocation (both successful and unsuccessful invocations)

.startAudioRecord

This method is used via LivePusherContext.startAudioRecord(Object object).

Feature Description: Starts audio recording, which will automatically terminate when LivePusherContext.stopAudioRecord is actively invoked, or when the recording exceeds one minute.

Parameter and Description: Object.

Attribute	Type	Default value	Required	Description
success	Function	-	No	Callback Function of Successful Interface Call
fail	Function	-	No	Callback Function of Failing Interface Call
complete	Function	-	No	Callback function executed upon the completion of the interface invocation (both successful and unsuccessful invocations)

.stopAudioRecord

©2013-2022 Tencent Cloud. All rights reserved.

This method is used via LivePusherContext.stopAudioRecord(Object object).

Feature Description: Terminates audio recording.

Parameter and Description: Object.

Attribute	Type	Default value	Required	Description
success	Function	-	No	Callback Function of Successful Interface Call
fail	Function	-	No	Callback Function of Failing Interface Call
complete	Function	-	No	Callback function executed upon the completion of the interface invocation (both successful and unsuccessful invocations)

Parameters for object.success callback function: Object res.

Attribute	Type	Description
tempFilePath	String	Temporary path of audio recording file (local path)

createLivePlayerContext

This API is used via LivePlayerContext wx.createLivePlayerContext(string id, Object this).

Note:

Support for this feature starts from the base library 1.4.96. Compatibility measures are required for earlier versions.

Feature Description: Creates a LivePlayerContext object.

Parameter and Description:

string id, the ID of the live-player component;

Object this, "this" of the current component instance under custom components, used to operate the live-player component within.

Return Value: LivePlayerContext.

LivePlayerContext

©2013-2022 Tencent Cloud. All rights reserved.

.exitfullscreen

This method is used via LivePlayerContext.exitFullScreen(Object object).

Feature Description: Exits full screen.

Parameter and Description: Object.

Attribute	Type	Default value	Required	Description
success	Function	-	No	Callback Function of Successful Interface Call
fail	Function	-	No	Callback Function of Failing Interface Call
complete	Function	-	No	Callback function executed upon the completion of the interface invocation (both successful and unsuccessful invocations)

.mute

This method is used via LivePlayerContext.mute(Object object).

Feature Description: Mute.

Parameter and Description: Object.

Attribute	Type	Default value	Required	Description
success	Function	-	No	Callback Function of Successful Interface Call
fail	Function	-	No	Callback Function of Failing Interface Call
complete	Function	-	No	Callback function executed upon the completion of the interface invocation (both successful and unsuccessful invocations)

.pause

This method is used via LivePlayerContext.pause(Object object).

©2013-2022 Tencent Cloud. All rights reserved.

Feature Description: Pause.

Parameter and Description: Object.

Attribute	Type	Default value	Required	Description
success	Function	-	No	Callback Function of Successful Interface Call
fail	Function	-	No	Callback Function of Failing Interface Call
complete	Function	-	No	Callback function executed upon the completion of the interface invocation (both successful and unsuccessful invocations)

.play

This method is used via LivePlayerContext.play(Object object).

Feature Description: Play.

Parameter and Description: Object.

Attribute	Type	Default value	Required	Description
success	Function	-	No	Callback Function of Successful Interface Call
fail	Function	-	No	Callback Function of Failing Interface Call
complete	Function	-	No	Callback function executed upon the completion of the interface invocation (both successful and unsuccessful invocations)

.requestFullScreen

This method is used via LivePlayerContext.requestFullScreen(Object object).

Feature Description: Enters full screen.

Parameter and Description: Object.

Attribute	Type	Default value	Required	Description

©2013-2022 Tencent Cloud. All rights reserved.

direction	Number	0	No	Set the direction during full screen.
success	Function	-	No	Callback Function of Successful Interface Call
fail	Function	-	No	Callback Function of Failing Interface Call
complete	Function	-	No	Callback function executed upon the completion of the interface invocation (both successful and unsuccessful invocations)

Valid values of object.direction.

Value	Description
0	Normal vertical orientation
90	Screen rotating 90 degrees counterclockwise
-90	Screen rotating 90 degrees clockwise

.resume

This method is used via LivePlayerContext.resume(Object object).

Feature Description: Restoration.

Parameter and Description: Object.

Attribute	Type	Default value	Required	Description
success	Function	-	No	Callback Function of Successful Interface Call
fail	Function	-	No	Callback Function of Failing Interface Call
complete	Function	-	No	Callback function executed upon the completion of the interface invocation (both successful and unsuccessful invocations)

.snapshot

This method is used via LivePlayerContext.snapshot(Object object).

©2013-2022 Tencent Cloud. All rights reserved.

Feature Description: Screenshot.

Parameter and Description: Object.

Attribute	Type	Default value	Required	Description
success	Function	-	No	Callback Function of Successful Interface Call
fail	Function	-	No	Callback Function of Failing Interface Call
complete	Function	-	No	Callback function executed upon the completion of the interface invocation (both successful and unsuccessful invocations)

Parameters for object.success callback function: Object res.

Attribute	Type	Description
tempImagePath	String	Temporary path of the image file
width	String	Image width
height	String	Image height

.stop

This method is used via LivePlayerContext.stop(Object object).

Feature Description: Halt.

Parameter and Description: Object.

Attribute	Type	Default value	Required	Description
success	Function	-	No	Callback Function of Successful Interface Call
fail	Function	-	No	Callback Function of Failing Interface Call
complete	Function	-	No	Callback function executed upon the completion of the interface invocation (both successful and unsuccessful invocations)

©2013-2022 Tencent Cloud. All rights reserved.

©2013-2022 Tencent Cloud. All rights reserved.

Transparent Video

Last updated：2024-07-12 19:14:10

createAnimationVideo

This API is used via AnimationVideoContext wx.createAnimationVideo(string id, Object this)

Feature Description: Creates an AnimationVideoContext object.

Parameter and Description:

string id: The id of the <animation-video> component.

Object this, "this" of the current component instance under custom components, used to operate the <animation-video> component within.

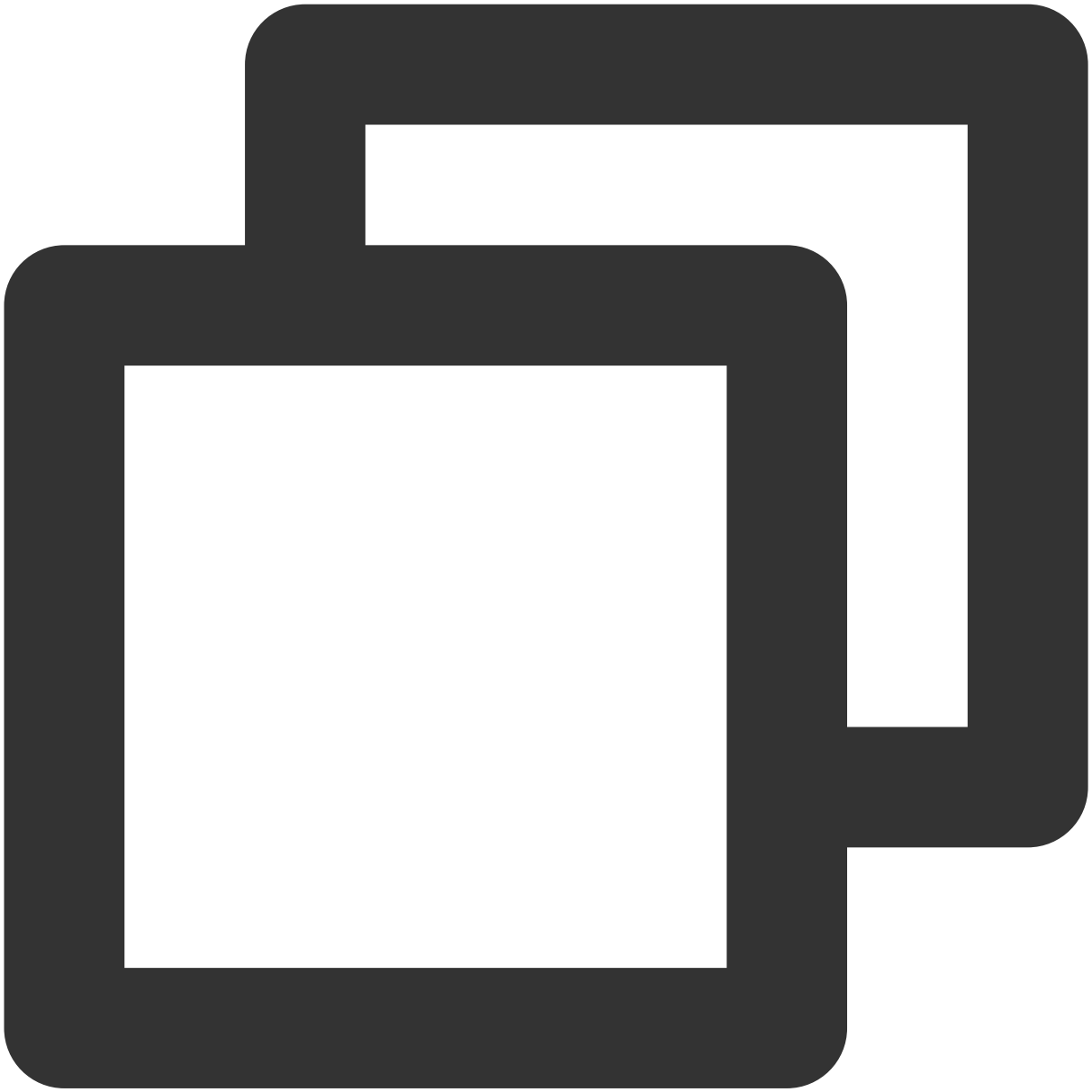
Return Value: AnimationVideoContext.

AnimationVideoContext

Feature Description: AnimationVideoContext instance, which can be obtained through wx.createAnimationVideo. AnimationVideoContext is bound to an <animation-video> component through its id, thereby manipulating the corresponding <animation-video> component.

Sample Code

©2013-2022 Tencent Cloud. All rights reserved.

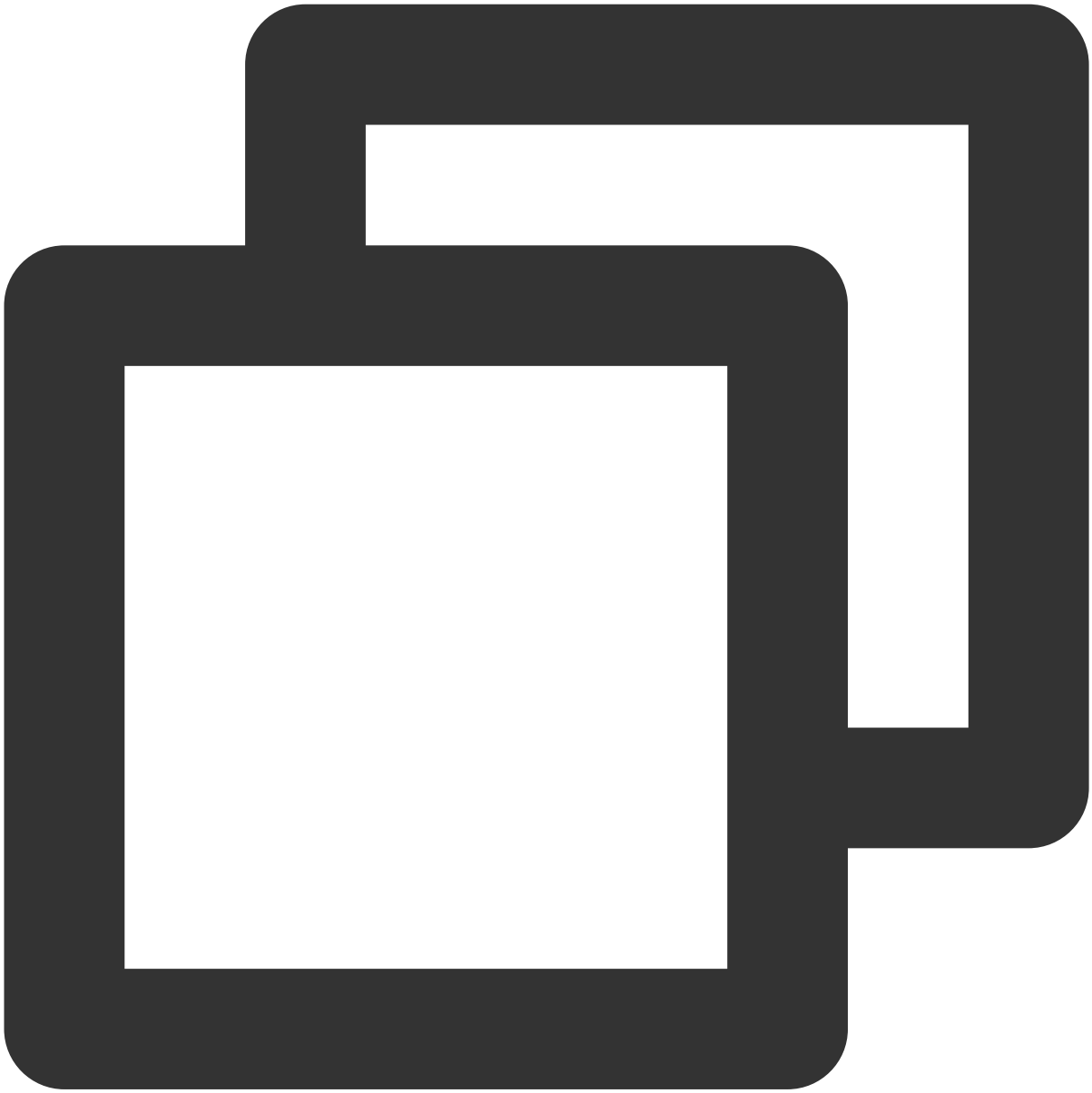


```
<view class="wrap">
  <view class="card-area">
    <view class="top-description border-bottom">
        <view>Feature</view>
        <view>play pause seek</view>
    </view>
    <view class="video-area">
        <animation-video
            id="myAnimationVideo"
            path="{{leftAlphaSrcPath}}"
            loop="{{loop}}"
```

©2013-2022 Tencent Cloud. All rights reserved.

```
            resource-width="800"
            resource-height="400"
            canvas-style="width:200px;height:200px"
            autoplay="{{autoplay}}"
            bindstarted="onStarted"
            bindended="onEnded"
        ></animation-video>
    </view>
    <button bindtap="play">
      Play
    </button>
    <button bindtap="pause">
      Pause
    </button>
    <button bindtap="seek">
      Navigate to the 2-second mark of the animation.
    </button>
  </view>
</view>
```

©2013-2022 Tencent Cloud. All rights reserved.



```
Page({
  onLoad() {
    this.createCtx();
  },
  createCtx() {
    this.myAnimationVideo = wx.createAnimationVideoContext('my-video');
  },
  play() {
    this.myAnimationVideo?.play();
  },
  pause() {
```

©2013-2022 Tencent Cloud. All rights reserved.

```
    this.myAnimationVideo?.pause();
  },
  seek() {
    this.myAnimationVideo?.seek(2);
  },
})
```

.play

This method is used via AnimationVideoContext.play().

Feature Description: Plays the video.

.pause

This method is used via AnimationVideoContext.pause().

Feature Description: Pauses the video.

.seek

This method is used via AnimationVideoContext.seek(number position).

Feature Description: Redirects to the specified position.

Parameter and Description: number position, the position to redirect to, unit: seconds.

©2013-2022 Tencent Cloud. All rights reserved.

Camera

Last updated：2024-07-12 19:13:57

createCameraContext

This method is used via CameraContext wx.createCameraContext().

Feature Description: Creates a CameraContext object.

Return Value: CameraContext.

CameraContext

Feature Description: A CameraContext instance, which can be obtained through wx.createCameraContext. The `cameraContext` is bound to the unique `<camera>` component within the page, operating the corresponding `<camera>` component.

.onCameraFrame

This method is used via CameraFrameListener CameraContext.onCameraFrame(onCameraFrameCallback callback).

Feature Description: Gets real-time frame data from the Camera.

Parameter and Description: Function callback.

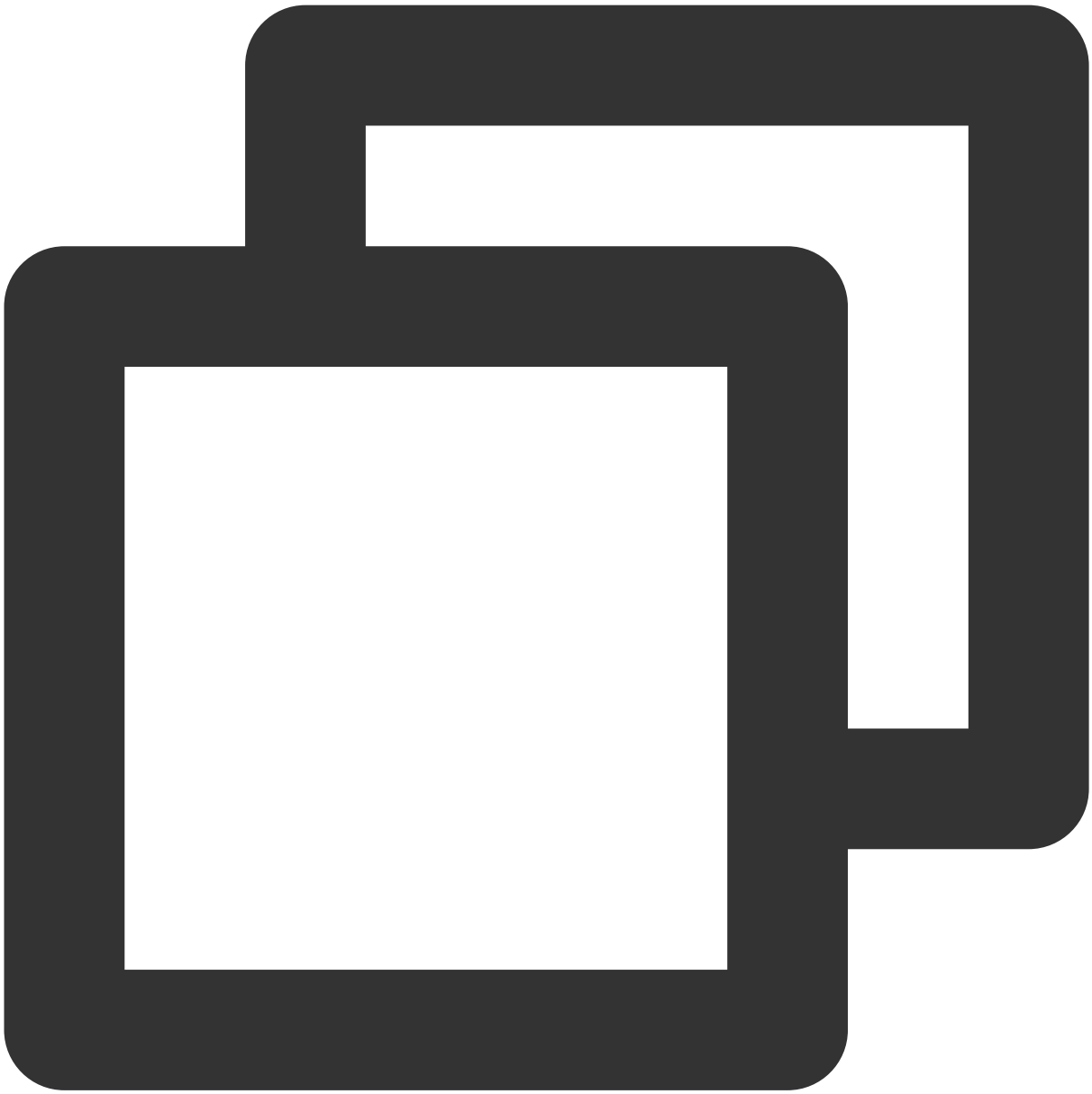
Callback function parameter: Object res.

Attribute	Type	Description
width	number	The width of the image data rectangle.
height	number	The height of the source image data rectangle.
data	ArrayBuffer	Image pixel data, a one-dimensional array, with every four items representing the RGBA of a pixel.

Return Value:CameraFrameListener。

©2013-2022 Tencent Cloud. All rights reserved.

Sample Code



```
const context = wx.createCameraContext()
const listener = context.onCameraFrame((frame) => {
  console.log(frame.data instanceof ArrayBuffer, frame.width, frame.height)
})
listener.start()
```

.setZoom

©2013-2022 Tencent Cloud. All rights reserved.

This method is used via CameraContext.setZoom(Object object).

Feature Description: Sets the scaling level.

Parameter and Description: Object.

Attribute	Type	Default value	Required	Description
zoom	number	-	Yes	Scaling level, ranging [1, maxZoom]. The zoom can be a decimal, accurate to one decimal place. maxZoom can be obtained from the return value of bindinitdone.
success	function	-	No	Callback Function of Successful Interface Call
fail	function	-	No	Callback Function of Failing Interface Call
complete	function	-	No	Callback function executed upon the completion of the interface invocation (both successful and unsuccessful invocations)

Parameters for object.success callback function: Object res.

Attribute	Type	Description
zoom	number	The actual set scaling level. Due to system constraints, some models may not be able to be set to the specified value, and will instead use the closest feasible value.

.startRecord

This method is used via CameraContext.startRecord(Object object).

Feature Description: Starts video recording.

Parameter and Description: Object.

Attribute	Type	Default value	Required	Description
timeoutCallback	function	-	No	The recording will terminate after 30 seconds or when the page `onHide` is triggered.
success	function	-	No	Callback Function of Successful Interface Call

©2013-2022 Tencent Cloud. All rights reserved.

fail	function	-	No	Callback Function of Failing Interface Call
complete	function	-	No	Callback function executed upon the completion of the interface invocation (both successful and unsuccessful invocations)

Parameter of the object.timeoutCallback callback function: Object res.

Attribute	Type	Description
tempThumbPath	string	Temporary path of the cover image file
tempVideoPath	string	Temporary path of the video file

.stopRecord

This method is used via CameraContext.startRecord(Object object).

Feature Description: Stops video recording.

Parameter and Description: Object.

Attribute	Type	Default value	Required	Description
success	function	-	No	Callback Function of Successful Interface Call
fail	function	-	No	Callback Function of Failing Interface Call
complete	function	-	No	Callback function executed upon the completion of the interface invocation (both successful and unsuccessful invocations)

Parameters for object.success callback function: Object res.

Attribute	Type	Description
tempThumbPath	string	Temporary path of the cover image file
tempVideoPath	string	Temporary path of the video file

.takePhoto

©2013-2022 Tencent Cloud. All rights reserved.

This method is used via CameraFrameListener.stop(Object object).

Feature Description: Captures photographs.

Parameter and Description: Object.

Attribute	Type	Default value	Required	Description
quality	string	normal	No	Image Quality
success	function	-	No	Callback Function of Successful Interface Call
fail	function	-	No	Callback Function of Failing Interface Call
complete	function	-	No	Callback function executed upon the completion of the interface invocation (both successful and unsuccessful invocations)

Valid values of object.quality

Value	Description
high	Higher quality
normal	Standard Quality
low	Lower quality

Parameters for object.success callback function: Object res.

Attribute	Type	Description
tempImagePath	string	Temporary path (local path) for photo files, Android is jpg image format, ios is png

CameraFrameListener

Note:

Listener Returned by CameraContext.onCameraFrame()

For related documentation, see camera.

.start

©2013-2022 Tencent Cloud. All rights reserved.

This method is used via CameraFrameListener.start(Object object).

Feature Description: Starts monitoring frame data.

Parameter and Description: Object.

Attribute	Type	Default value	Required	Description
success	function	-	No	Callback Function of Successful Interface Call
fail	function	-	No	Callback Function of Failing Interface Call
complete	function	-	No	Callback function executed upon the completion of the interface invocation (both successful and unsuccessful invocations)

.stop

This method is used via CameraFrameListener.stop(Object object).

Feature Description: Stops monitoring frame data.

Parameter and Description: Object.

Attribute	Type	Default value	Required	Description
success	function	-	No	Callback Function of Successful Interface Call
fail	function	-	No	Callback Function of Failing Interface Call
complete	function	-	No	Callback function executed upon the completion of the interface invocation (both successful and unsuccessful invocations)

©2013-2022 Tencent Cloud. All rights reserved.

Image

Last updated：2024-07-12 19:13:05

chooseImage

This API is used via wx.chooseImage(Object object).

Feature Description: Selects images from the local album or uses the camera to take photos.

Parameter and Description: Object.

Attribute	Type	Valid Values and Descriptions	Default value	Required	Description
count	number	-	9	No	Maximum selectable number of images
sizeType	Array.<string>	Original: Original image Compressed: Compressed image	['original', 'compressed']	No	Dimensions of the Selected Image
sourceType	Array.<string>	Album: Select image from album Camera: Uses the camera	['album', 'camera']	No	Select image source
success	function	-	-	No	Callback Function of Successful Interface Call
fail	function	-	-	No	Callback Function of Failing Interface Call
complete	function	-	-	No	Callback function executed upon the completion of the interface invocation (both successful and unsuccessful invocations)

©2013-2022 Tencent Cloud. All rights reserved.

Parameters for object.success callback function: Object res.

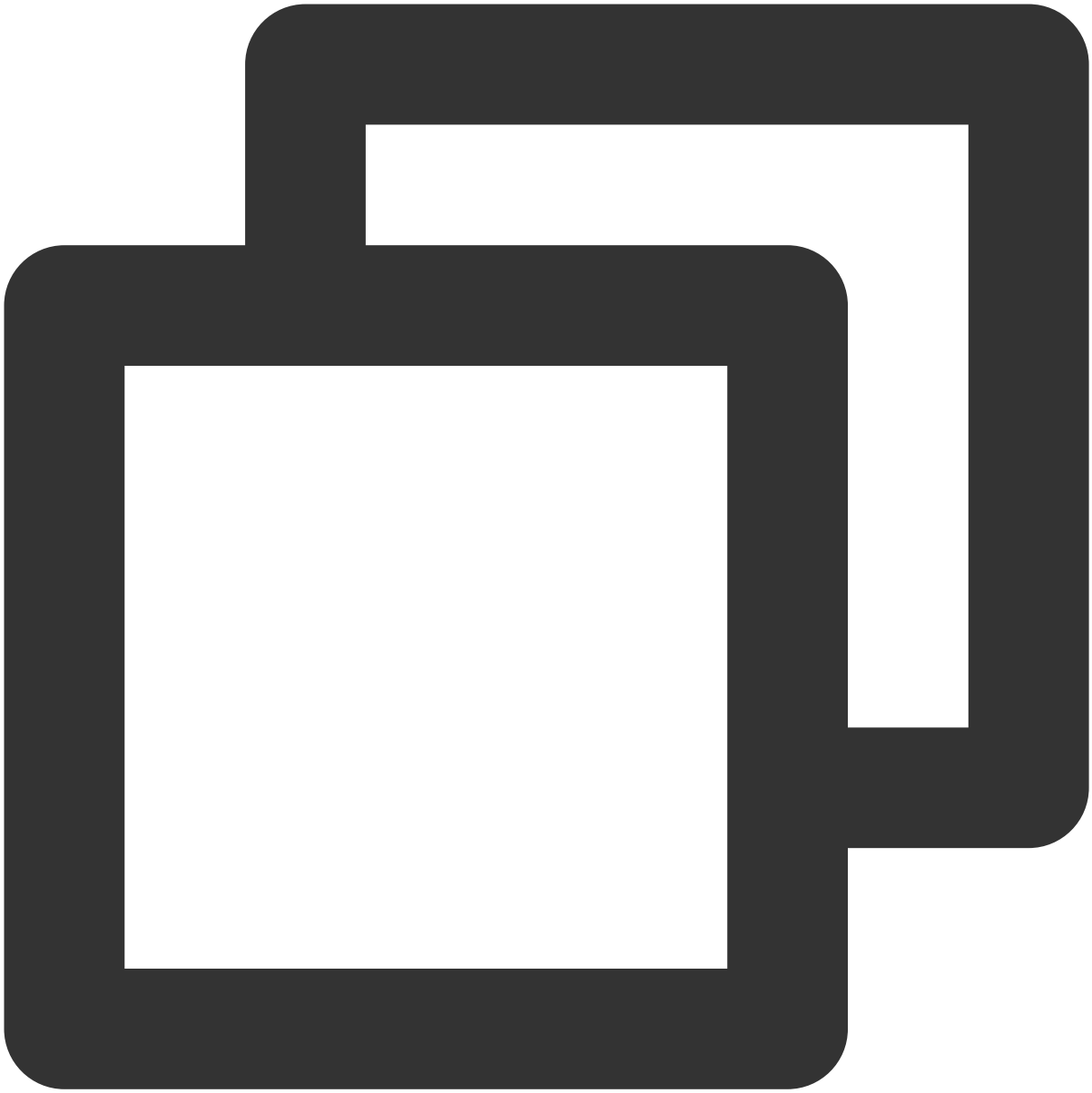
Attribute	Type	Description
tempFilePaths	Array.<string>	List of local temporary file paths for the image (Local Path)
tempFiles	Array.<string>	List of local temporary files for the image

Structure of res.tempFiles

Attribute	Type	Description
path	string	Local temporary file path
size	number	Size of local temporary files, measured in bytes.

Sample Code

©2013-2022 Tencent Cloud. All rights reserved.



```
wx.chooseImage({
  count: 1,
  sizeType: ['original', 'compressed'],
  sourceType: ['album', 'camera'],
  success(res) {
    // tempFilePath can be used as the src attribute in an img tag to display the i
    const tempFilePaths = res.tempFilePaths
  }
})
```

©2013-2022 Tencent Cloud. All rights reserved.

compressImage

This API is used via wx.compressImage(Object object).

Feature Description: Image compression interface, with optional compression quality.

Parameter and Description: Object.

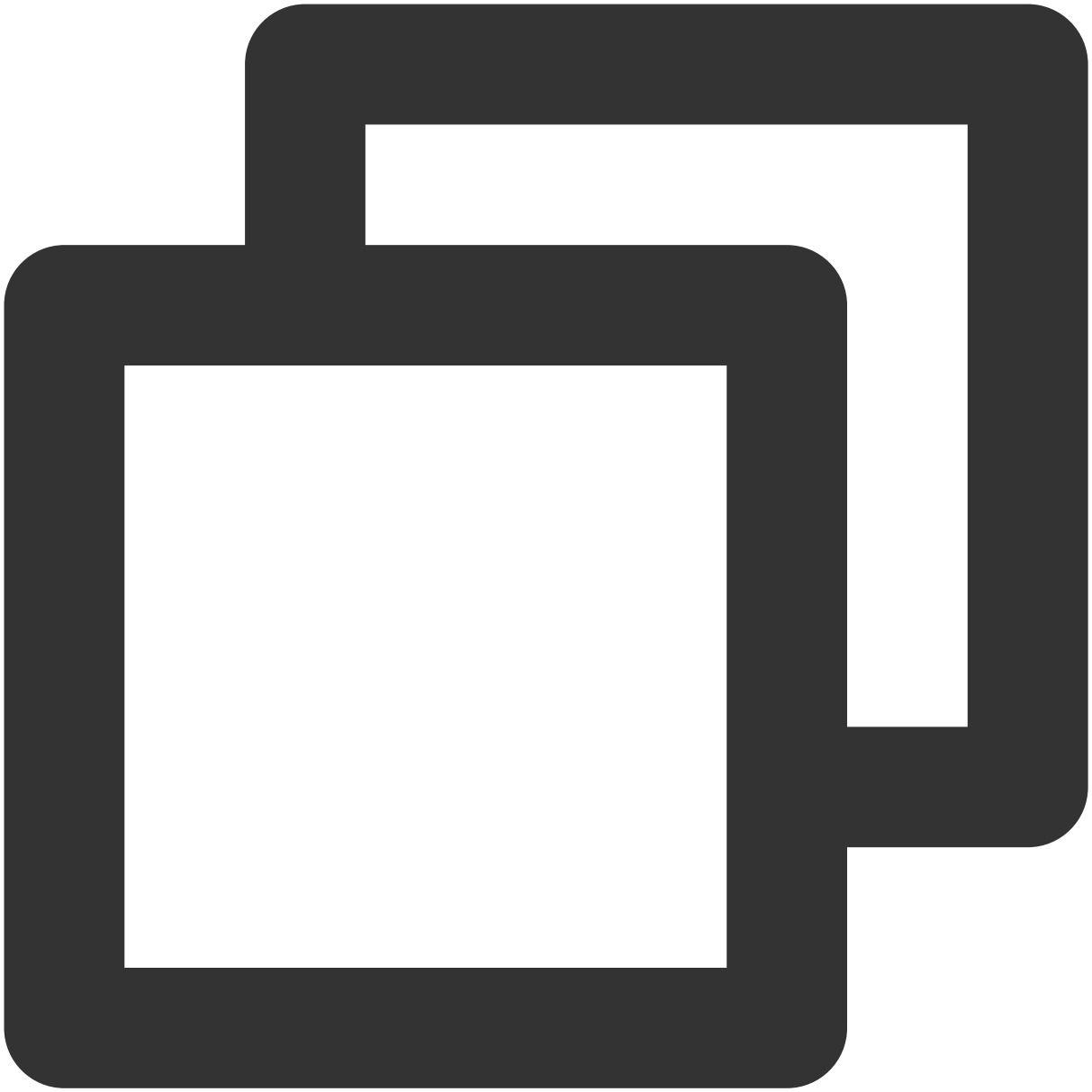
Attribute	Type	Default value	Required	Description
src	string	-	Yes	Image path, the path of the image, supporting local paths and code package paths.
quality	number	80	No	Compression quality, ranging from 0 to 100. The lower the value, the lower the quality and the higher the compression rate (only effective for jpg).
compressedWidth	number	-	No	The width of the compressed image, in pixels (px). If not specified, it will default to proportional scaling based on compressedHeight.
compressedHeight	number	-	No	The height of the compressed image, in pixels (px). If not specified, it will default to proportional scaling based on compressedWidth.
success	function	-	No	Callback Function of Successful Interface Call
fail	function	-	No	Callback Function of Failing Interface Call
complete	function	-	No	Callback function executed upon the completion of the interface invocation (both successful and unsuccessful invocations)

Parameters for object.success callback function: Object res.

Attribute	Type	Description
tempFilePath	string	The temporary file path of the compressed image (local path).

Sample Code

©2013-2022 Tencent Cloud. All rights reserved.



```
wx.compressImage({
  src: '', // Image path
  quality: 80 // Compression quality
})
```

getImageInfo

©2013-2022 Tencent Cloud. All rights reserved.

This API is used via wx.getImageInfo(Object object).

Feature Description: Gets image information. For network images to work, the download domain must be configured first.

Parameter and Description: Object.

Attribute	Type	Default value	Required	Description
src	string	-	Yes	The path of the image, which can be a relative path, temporary file path, storage file path, or network image path.
success	function	-	No	Callback Function of Successful Interface Call
fail	function	-	No	Callback Function of Failing Interface Call
complete	function	-	No	Callback function executed upon the completion of the interface invocation (both successful and unsuccessful invocations)

Parameters for object.success callback function: Object res.

Attribute	Type	Description
width	number	Original width of the image, in pixels (px). Rotation is not considered.
height	number	Original height of the image, in pixels (px). Rotation is not considered.
path	string	Local path of image
orientation	string	Device orientation during photography
type	string	Image format

Valid values for res.orientation

Value	Description
up	Default orientation (photographing horizontally with a mobile phone), corresponding to 1 in Exif, or without orientation information.
up-mirrored	Identical to 'up', but mirrored, corresponding to 2 in Exif.
down	Rotating 180 degrees, corresponding to 3 in Exif.

©2013-2022 Tencent Cloud. All rights reserved.

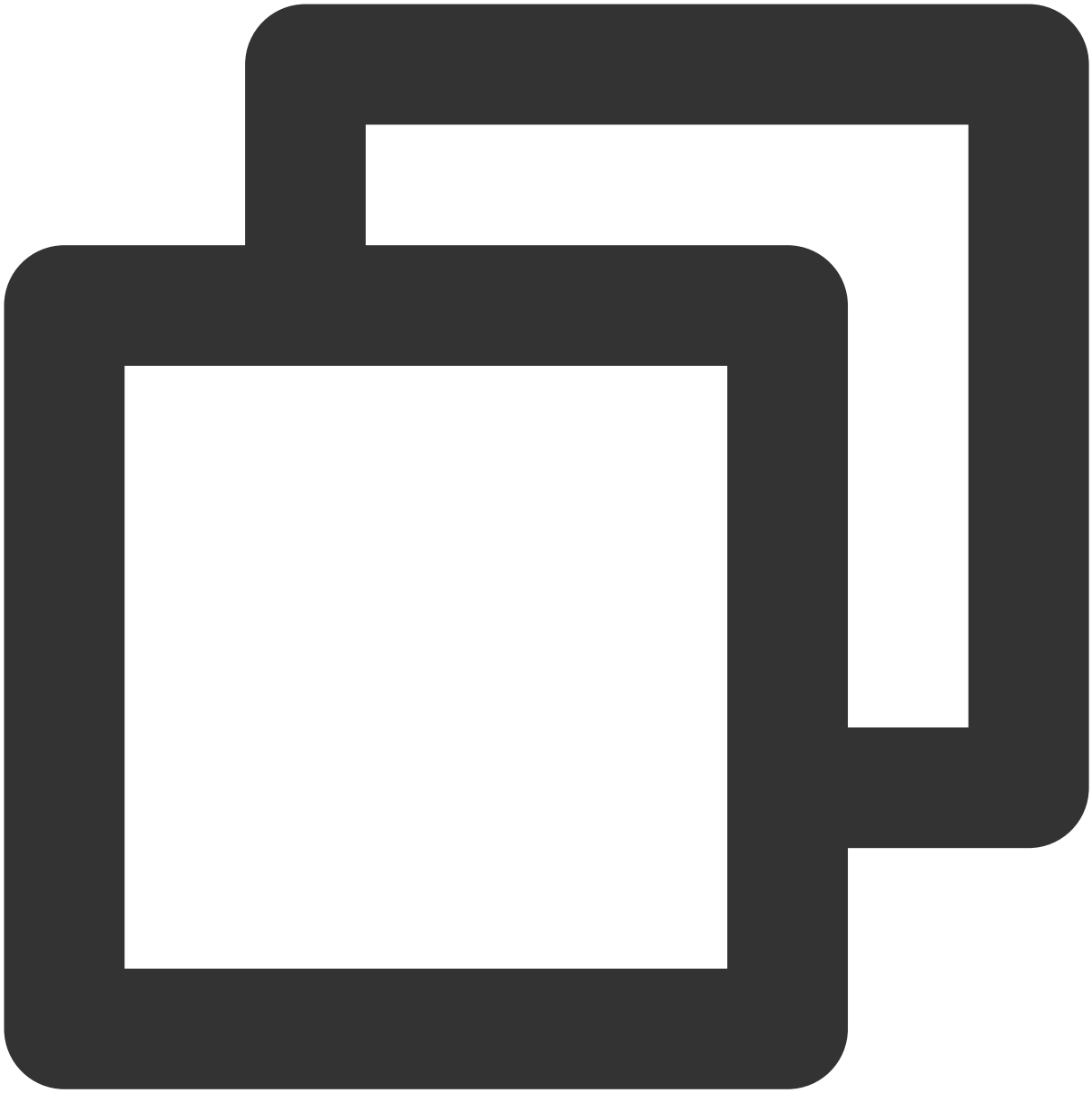
down-mirrored	Identical to 'down', but mirrored, corresponding to 4 in Exif.
left-mirrored	Identical to 'left', but mirrored, corresponding to 5 in Exif.
right	Rotating 90 degrees clockwise, corresponding to 6 in Exif.
right-mirrored	Identical to 'right', but mirrored, corresponding to 7 in Exif.
left	Rotating 90 degrees counterclockwise, corresponding to 8 in Exif.

The legal value of res.type

Legal value	Description
unknown	unknown format
jpeg	jpeg compression format
png	png compression format
gif	gif compression format
tiff	tiff compression format

Sample Code

©2013-2022 Tencent Cloud. All rights reserved.



```
wx.getImageInfo({
  src: 'images/a.jpg',
  success(res) {
    console.log(res.width)
    console.log(res.height)
  },
})

wx.chooseImage({
  success(res) {
    wx.getImageInfo({
```

©2013-2022 Tencent Cloud. All rights reserved.

```
        src: res.tempFilePaths[0],
        success(res) {
          console.log(res.width)
          console.log(res.height)
        },
      })
    },
})
```

previewImage

This API is used via wx.previewImage(Object object).

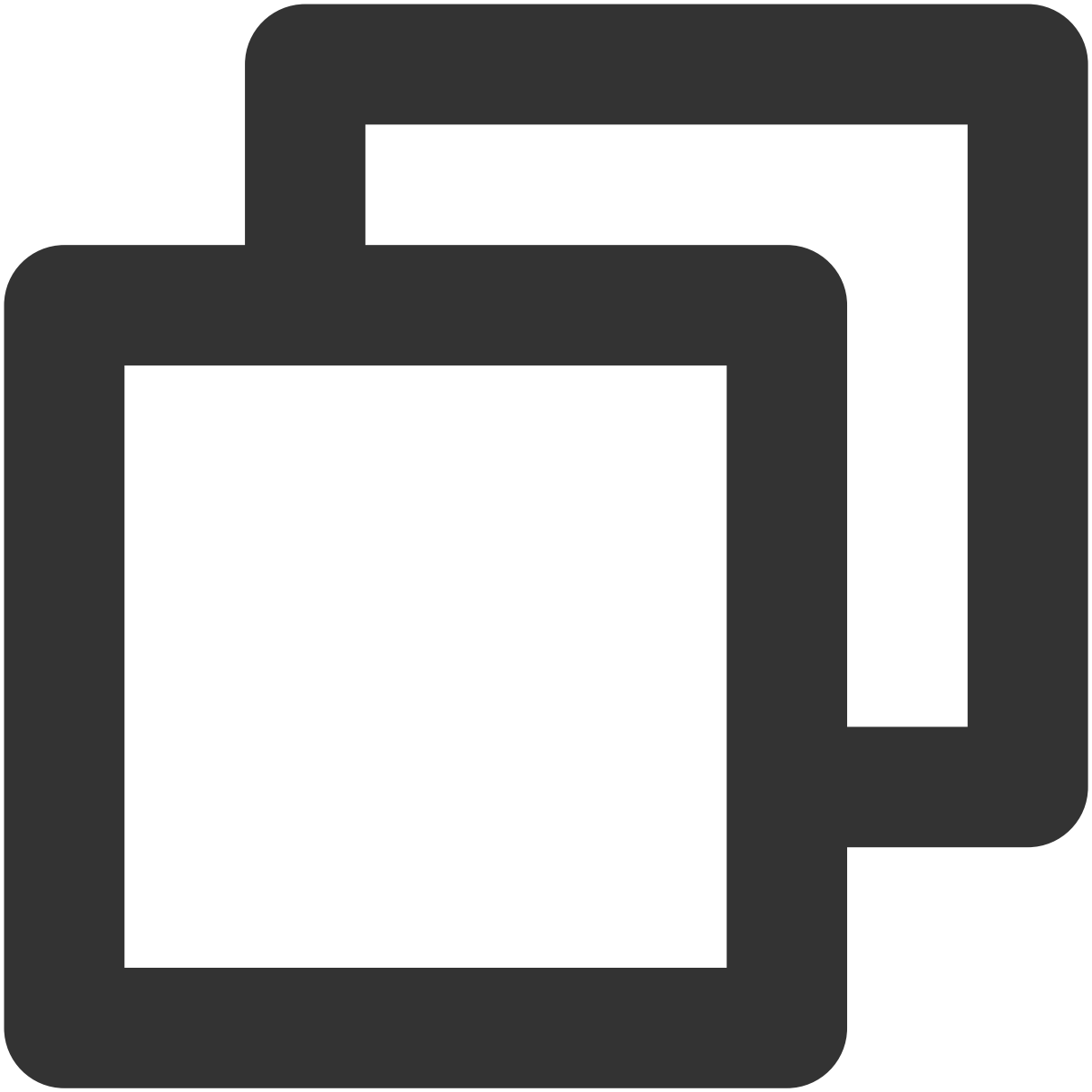
Feature Description: Full-screen image preview in a new page. During the preview, users can save the image, share it with friends, and perform other operations.

Parameter and Description: Object.

Attribute	Type	Default value	Required	Description
urls	Array.<string>	-	Yes	List of image links to be previewed. Cloud file ID is supported.
current	string	The first one in urls.	No	The link to the currently displayed image.
success	function	-	No	Callback Function of Successful Interface Call
fail	function	-	No	Callback Function of Failing Interface Call
complete	function	-	No	Callback function executed upon the completion of the interface invocation (both successful and unsuccessful invocations)

Sample Code

©2013-2022 Tencent Cloud. All rights reserved.



```
wx.previewImage({
  current: '', // The HTTP link to the currently displayed image.
  urls: [], // List of HTTP image links to be previewed.
})
```

previewMedia

©2013-2022 Tencent Cloud. All rights reserved.

This API is used via wx.previewMedia(Object object).

Feature Description: Previews images and videos.

Parameter and Description: Object.

Attribute	Type	Default value	Required	Description
sources	Array.<Object>		Yes	List of resources to preview
current	number	0	No	Serial number of the currently displayed resource
filePath	string	-	Yes	Image file path, can be temporary file path or permanent file path (local path), does not support network image paths
success	function	-	No	Callback Function of Successful Interface Call
fail	function	-	No	Callback Function of Failing Interface Call
complete	function	-	No	Callback function executed upon the completion of the interface invocation (both successful and unsuccessful invocations)

saveImageToPhotosAlbum

This API is used via wx.saveImageToPhotosAlbum(Object object).

Note:

Prior to invocation, user authorization "scope.writePhotosAlbum" is required.

Feature Description: Saves images to the system album.

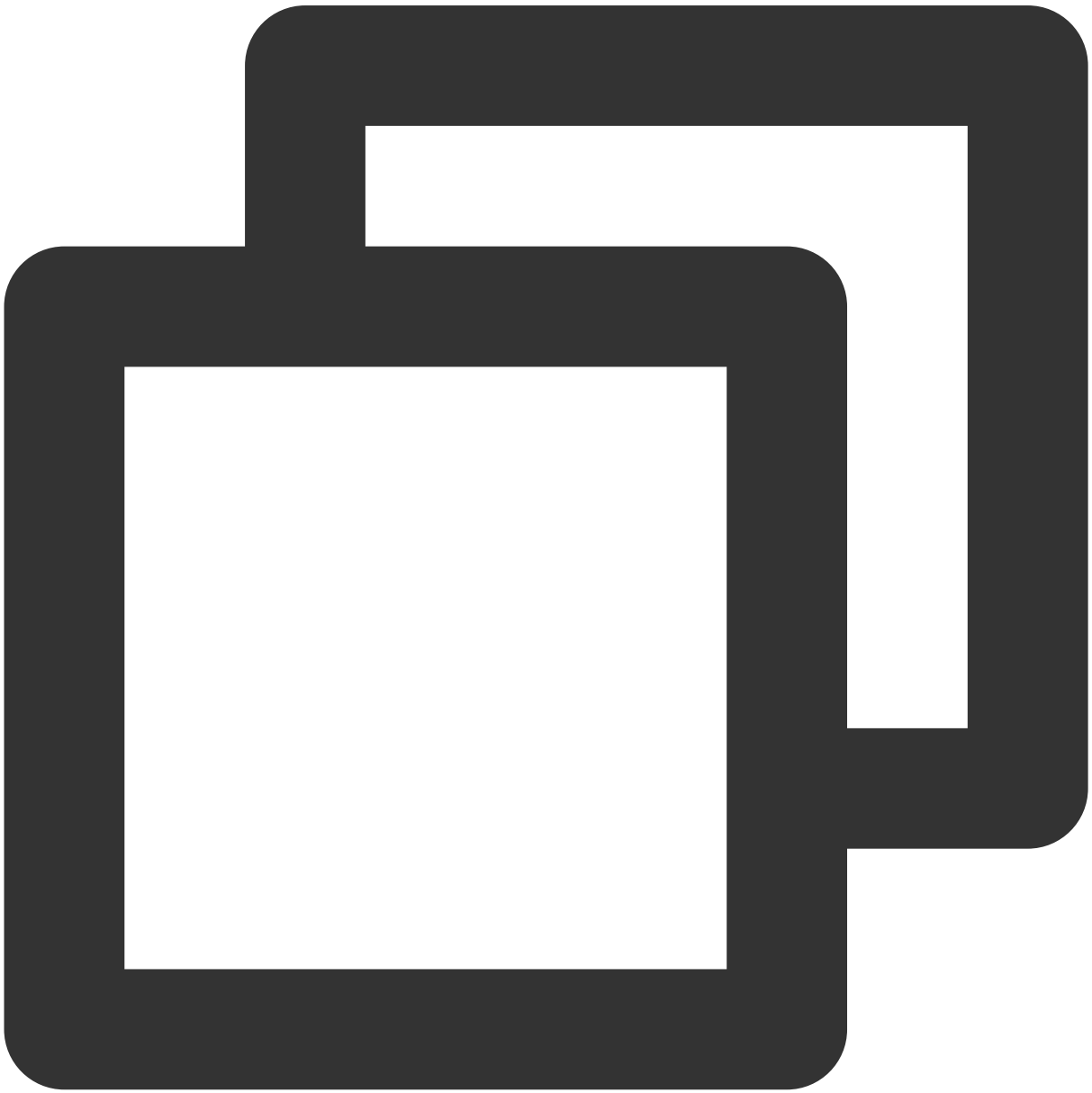
Parameter and Description: Object.

Attribute	Type	Default value	Required	Description
filePath	string	-	Yes	Image file path, can be temporary file path or permanent file path (local path), does not support network image paths

©2013-2022 Tencent Cloud. All rights reserved.

success	function	-	No	Callback Function of Successful Interface Call
fail	function	-	No	Callback Function of Failing Interface Call
complete	function	-	No	Callback function executed upon the completion of the interface invocation (both successful and unsuccessful invocations)

Sample Code



```
wx.saveImageToPhotosAlbum({
```

©2013-2022 Tencent Cloud. All rights reserved.

```
  success(res) {},
})
```

©2013-2022 Tencent Cloud. All rights reserved.

Audio

Last updated：2024-07-12 19:12:24

stopVoice

This API is used via wx.stopVoice(Object object).

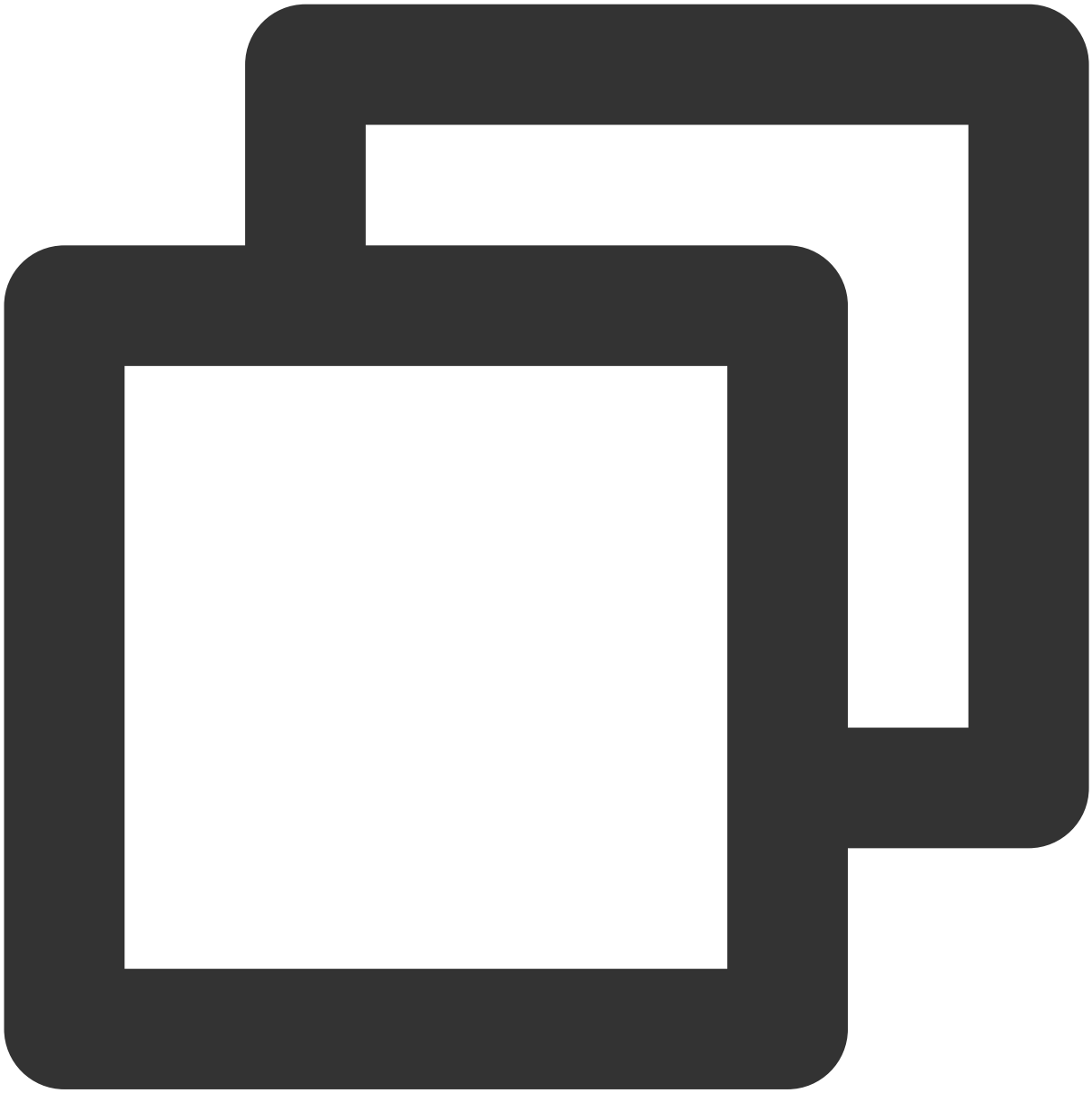
Feature Description: Terminates the playback of audio.

Parameter and Description: Object.

Attribute	Type	Default value	Required	Description
success	function	-	No	Callback Function of Successful Interface Call
fail	function	-	No	Callback Function of Failing Interface Call
complete	function	-	No	Callback function executed upon the completion of the interface invocation (both successful and unsuccessful invocations)

Sample Code

©2013-2022 Tencent Cloud. All rights reserved.



```
wx.startRecord({
  success (res) {
    const tempFilePath = res.tempFilePath
    wx.playVoice({
      filePath: tempFilePath,
    })

    setTimeout(() => { wx.stopVoice() }, 5000)
  }
})
```

©2013-2022 Tencent Cloud. All rights reserved.

playVoice

This API is used via wx.playVoice(Object object).

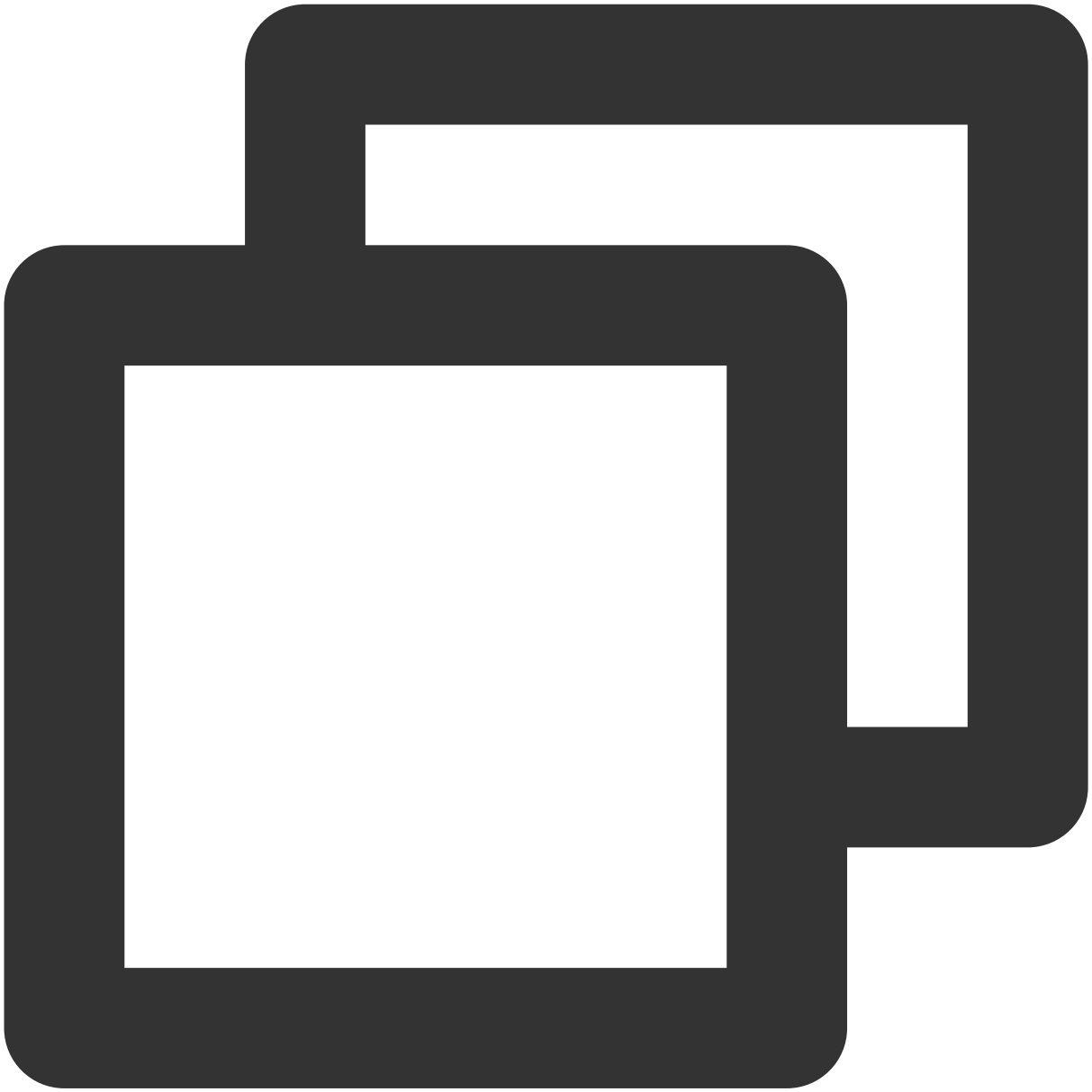
Feature Description: Starts the playback of voice. Only one voice file is permitted to play at a time. If a previous voice file is still playing, it will be interrupted.

Parameter and Description:

Attribute	Type	Default value	Required	Description
filePath	string	-	Yes	Path of the audio file to be played (local path)
duration	number	60	No	Specifies the playback duration. Upon reaching the specified duration, the playback will automatically cease. Unit: seconds.
success	function	-	No	Callback Function of Successful Interface Call
fail	function	-	No	Callback Function of Failing Interface Call
complete	function	-	No	Callback function executed upon the completion of the interface invocation (both successful and unsuccessful invocations)

Sample Code

©2013-2022 Tencent Cloud. All rights reserved.



```
wx.startRecord({
  success (res) {
    const tempFilePath = res.tempFilePath
    wx.playVoice({
      filePath: tempFilePath,
      complete () { }
    })
  }
})
```

©2013-2022 Tencent Cloud. All rights reserved.

pauseVoice

This API is used via wx.pauseVoice(Object object).

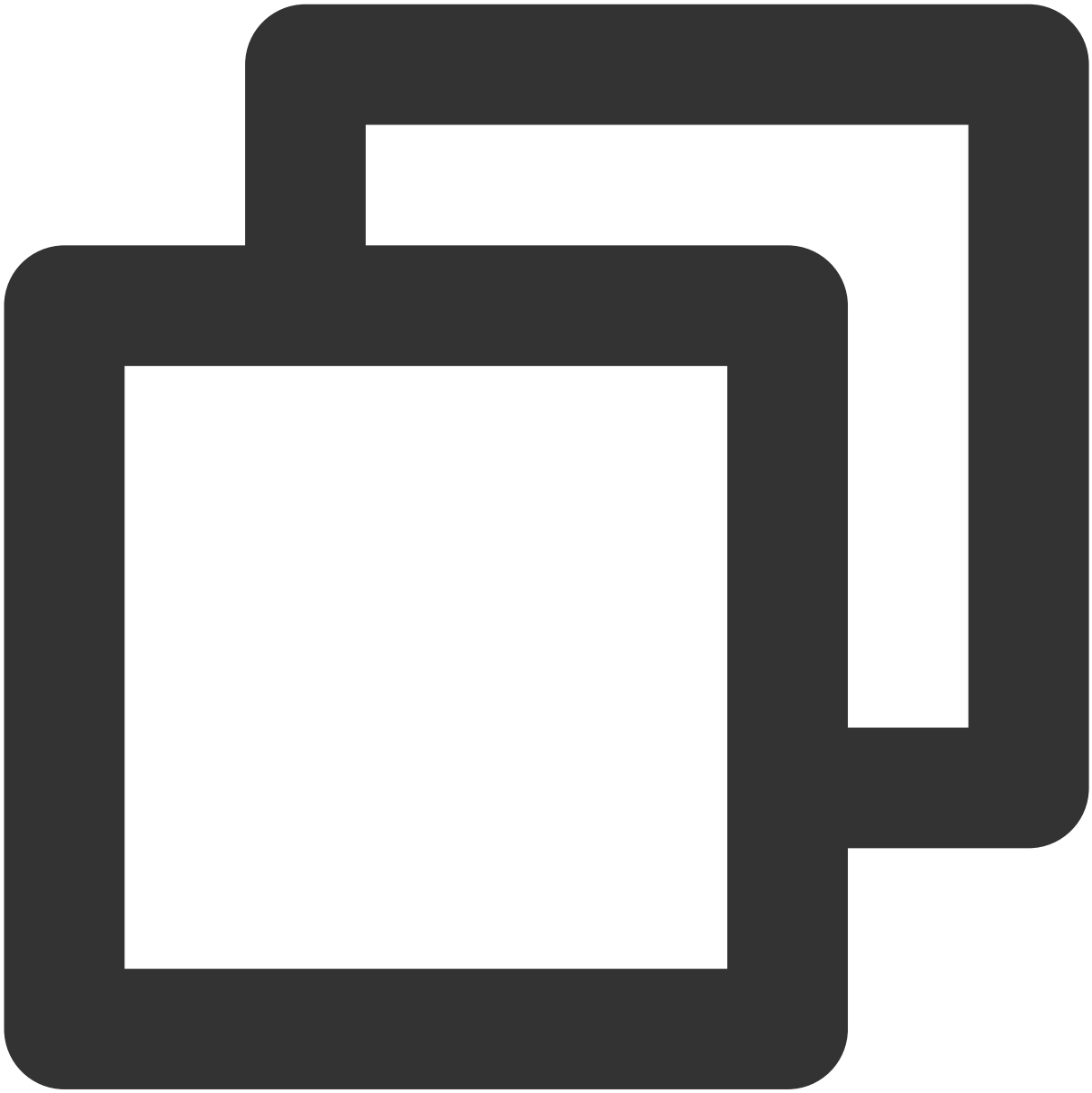
Feature Description: Pauses the currently playing audio. When wx.playVoice is called again to play the same file, it will resume from the point of pause. If playback is to start from the beginning, wx.stopVoice must be called first.

Parameter and Description: Object.

Attribute	Type	Default value	Required	Description
success	function	-	No	Callback Function of Successful Interface Call
fail	function	-	No	Callback Function of Failing Interface Call
complete	function	-	No	Callback function executed upon the completion of the interface invocation (both successful and unsuccessful invocations)

Sample Code

©2013-2022 Tencent Cloud. All rights reserved.



```
wx.startRecord({
  success (res) {
    const tempFilePath = res.tempFilePath
    wx.playVoice({
      filePath: tempFilePath
    })

    setTimeout(() => { wx.pauseVoice() }, 5000)
  }
})
```

©2013-2022 Tencent Cloud. All rights reserved.

setInnerAudioOption

This API is used via wx.setInnerAudioOption(Object object).

Feature Description: Sets the playback options for InnerAudioContext. Once set, it applies globally to the current mini program.

Parameter: Object.

Attribute	Type	Default value	Required	Description
mixWithOther	boolean	true	No	Whether to mix with other audio. When set to true, it will not terminate the music of other applications.
obeyMuteSwitch	boolean	true	No	(Effective only on iOS) Whether to adhere to the mute switch. When set to false, sound can be played even in mute mode.
speakerOn	boolean	true	No	true for playback from speakers, false for playback from earpiece, default value is true
success	function	-	No	Callback Function of Successful Interface Call
fail	function	-	No	Callback Function of Failing Interface Call
complete	function	-	No	Callback function executed upon the completion of the interface invocation (both successful and unsuccessful invocations)

getAvailableAudioSources

This API is used via wx.getAvailableAudioSources(Object object).

Feature Description: Gets the currently supported audio input sources.

Parameter and Description: Object.

Attribute	Type	Default value	Required	Description
success	function	-	No	Callback Function of Successful Interface Call

©2013-2022 Tencent Cloud. All rights reserved.

fail	function	-	No	Callback Function of Failing Interface Call
complete	function	-	No	Callback function executed upon the completion of the interface invocation (both successful and unsuccessful invocations)

Parameters for object.success callback function:

Attribute	Type	Description
audioSources	Array.<string>	List of supported audio input sources. See AudioSource for the definition of return values.

Valid values for audioSources

Valid Values	Description
auto	Automatically configured to default to the mobile microphone. Upon connecting a headset, it automatically switches to the headset microphone. Applicable across all platforms.
buildInMic	Mobile microphone, exclusive to iOS.
headsetMic	Headset microphone, exclusive to iOS.
mic	Microphone (defaults to mobile microphone when headset is not connected, switches to headset microphone when connected), exclusive to Android.
camcorder	Similar to the microphone, applicable for recording audio and video content, exclusive to Android.
voice_communication	Similar to the microphone, applicable for real-time communications, exclusive to Android.
voice_recognition	Similar to the microphone, applicable for recording audio content, exclusive to Android.

createAudioContext

This API is used via AudioContext wx.createAudioContext(string id, Object this).

Feature Description: Creates anAudioContext object.

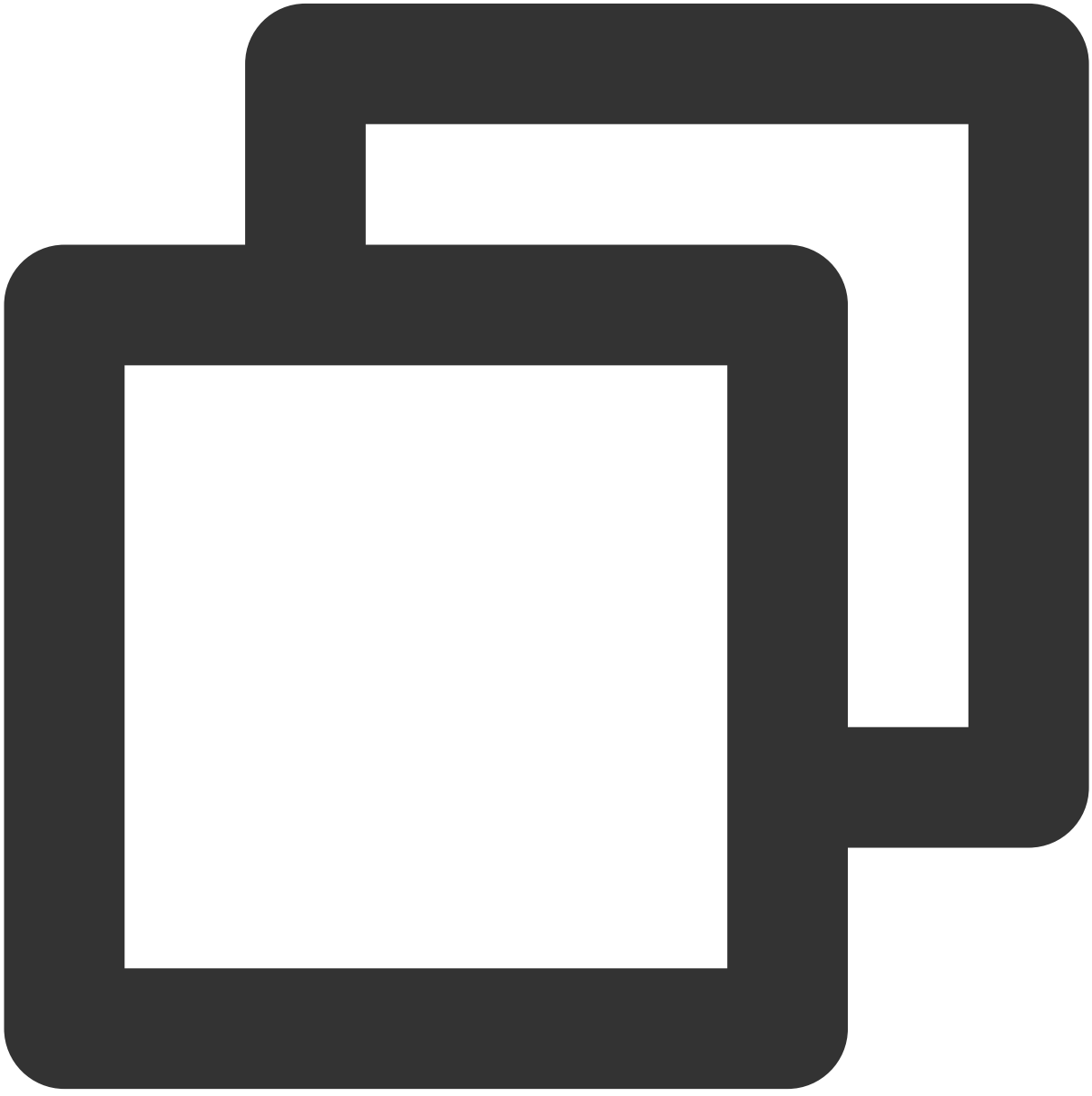
©2013-2022 Tencent Cloud. All rights reserved.

Parameter and Description: string id, ID of the audio component; Object this, "this" of the current component instance under custom components, used to operate the audio component within.

Return Value: AudioContext

Note: InnerAudioContext audio resources will not be automatically released, so if you no longer need to use the audio, please call InnerAudioContext.destroy() to release the resources in time to avoid memory leakage.

Sample code:



```
const innerAudioContext = wx.createInnerAudioContext()
innerAudioContext.src = 'http://ws.stream.qqmusic.qq.com/M500001VfvsJ21xFqb.mp3?gui
```

©2013-2022 Tencent Cloud. All rights reserved.

```
innerAudioContext.play() // play

innerAudioContext.pause() // pause

innerAudioContext.stop() // stops

innerAudioContext.destroy() // free the audio resource
```

AudioContext

An AudioContext instance can be obtained through wx.createAudioContext. It binds with an audio component through an id, allowing operations on the corresponding audio component.

pause

> This method is used via AudioContext.pause().

Feature Description: Pauses the audio.

play

> This method is used via AudioContext.play().

Feature Description: Plays the audio.

seek

> This method is used via AudioContext.seek(number position).

Feature Description: Redirects to the specified position.

Parameter and Description: number position, the position to redirect to, unit: seconds.

setSrc

©2013-2022 Tencent Cloud. All rights reserved.

This method is used via AudioContext.setSrc(string src).

Feature Description: Sets the audio address.
Parameter and Description: string src, audio address.

createInnerAudioContext

This API is used via InnerAudioContext wx.createInnerAudioContext(Object object).

Feature Description: Creates an internal `audio` context InnerAudioContext object, with the usage method being wx.createInnerAudioContext().
Return Value: InnerAudioContext.

InnerAudioContext

Attributes

string src: The address of the audio resource, used for direct playback. Cloud file ID is supported.
number startTime: The starting position for playback (in seconds), defaulting to 0.
boolean autoplay: Determines whether playback starts automatically, defaulting to "false".
boolean loop: Determines whether the playback circulates, defaulting to "false".
number volume: Volume level, ranging from 0 to 1, defaulting to 1.
number duration: The length of the current audio (in seconds). Only returned when there is a valid src (read-only).
number currentTime: The current playback position of the audio (in seconds). Only returned when there is a valid src, and the time is accurate to 6 decimal places (read-only).
boolean paused: Indicates whether the current status is paused or stopped (read-only).
number buffered: The time point that the audio is buffered to, ensuring content from the current playback point to this point has been buffered (read-only).

Supported format

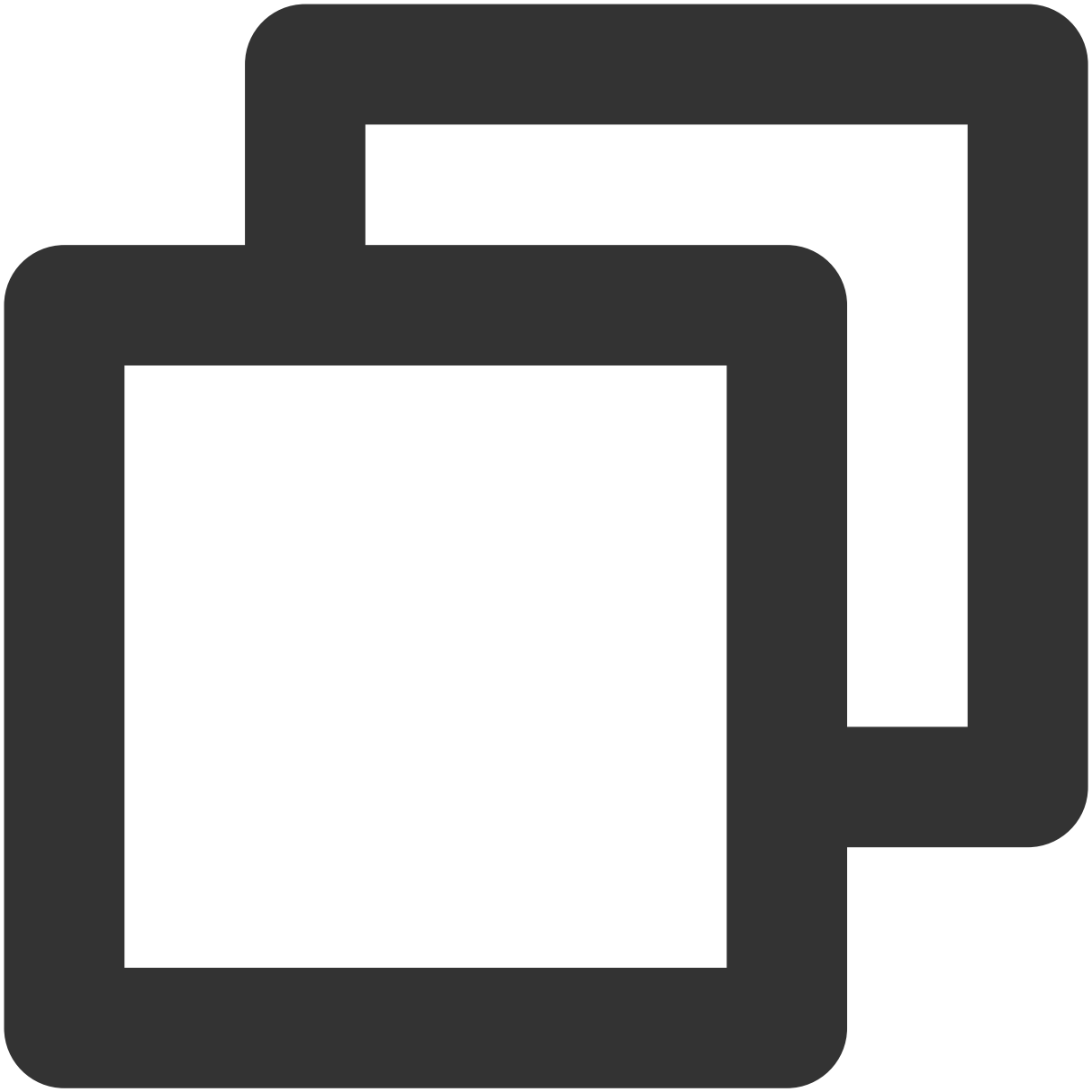
Format	iOS	Android
flac	x	√

©2013-2022 Tencent Cloud. All rights reserved.

m4a	√	√
ogg	x	√
ape	x	√
amr	x	√
wma	x	√
wav	√	√
mp3	√	√
mp4	x	√
aac	√	√
aiff	√	x
caf	√	x

Sample code

©2013-2022 Tencent Cloud. All rights reserved.



```
const innerAudioContext = wx.createInnerAudioContext()
innerAudioContext.autoplay = true
innerAudioContext.src = 'https://ws.stream.qqmusic.qq.com/M500001VfvsJ21xFqb.mp3?gu
innerAudioContext.onPlay(() => {
  console.log('Start playback')
})
innerAudioContext.onError((res) => {
  console.log(res.errMsg)
  console.log(res.errCode)
})
```

©2013-2022 Tencent Cloud. All rights reserved.

Method Set

play

This method is used via InnerAudioContext.play().

Feature Description: Play.

pause

This method is used via InnerAudioContext.pause().

Feature Description: Pause. Upon resumption, the audio will start playback from the point of pause.

stop

This method is used via InnerAudioContext.stop().

Feature Description: Stop. Upon resumption, the audio will start playback from the beginning.

seek

This method is used via InnerAudioContext.seek(number position).

Feature Description: Redirects to the specified position.

destroy

This method is used via InnerAudioContext.destroy().

Feature Description: Terminates the current instance.

onCanplay

This method is used via InnerAudioContext.onCanplay(function listener).

©2013-2022 Tencent Cloud. All rights reserved.

Feature Description: Monitors events where the audio enters a playable status. However, it does not guarantee smooth playback thereafter.
Parameter and Description: function listener, the listener function for events where the audio enters a playable status.
offCanplay

This method is used via InnerAudioContext.offCanplay(function listener).

Feature Description: Removes the listener function for events where the audio enters a playable status.
Parameter and Description: function listener, the listener function passed in by onCanplay. If this parameter is not passed in, all listener functions will be removed.
onPlay

This method is used via InnerAudioContext.onPlay(function listener).

Feature Description: Monitors audio playback events.
Parameters and Description: function listener, the listener function for audio playback events.
offPlay

This method is used via InnerAudioContext.offPlay(function listener).

Feature Description: Removes the listener function for audio playback events.
Parameter and Description: function listener, the listener function passed in by onPlay. If this parameter is not passed in, all listener functions will be removed.
onPuase

This method is used via InnerAudioContext.onPause(function listener).

Feature Description: Monitors audio pause events.
Parameter and Description: function listener, the listener function for audio pause events.
offPause

©2013-2022 Tencent Cloud. All rights reserved.

This method is used via InnerAudioContext.offPause(function listener).

Feature Description: Removes the listener function for audio pause events.
Parameter and Description: function listener, the listener function passed in by onPause. If this parameter is not passed in, all listener functions will be removed.
onStop

This method is used via InnerAudioContext.onStop(function listener).

Feature Description: Monitors audio stop events.
Parameter and Description: function listener, the listener function for audio stop events.
offStop
This method is used via InnerAudioContext.offStop(function listener).
Feature Description: Removes the listener function for audio stop events.
Parameter and Description: function listener, the listener function passed in by onStop. If this parameter is not passed in, all listener functions will be removed.
onEnded

This method is used via InnerAudioContext.onEnded(function listener).

Feature Description: Monitors events of audio naturally playing to its end.
Parameters and Description: function listener, the listener function for events of audio naturally playing to its end.
offEnded

This method is used via InnerAudioContext.offEnded(function listener).

Feature Description: Removes the listener function for events of audio naturally playing to its end.
Parameter and Description: function listener, the listener function passed in by onEnded. If this parameter is not passed in, all listener functions will be removed.
onTimeUpdate

This method is used via InnerAudioContext.onTimeUpdate(function listener).

©2013-2022 Tencent Cloud. All rights reserved.

Feature Description: Monitors events of audio playback progress updates.
Parameter and Description: function listener, the listener function for events of audio playback progress updates.
offTimeUpdate

This method is used via InnerAudioContext.offTimeUpdate(function listener).

Feature Description: Removes the listener function for events of audio playback progress updates.
Parameter and Description: function listener, the listener function passed in by onTimeUpdate. If this parameter is not passed in, all listener functions will be removed.
onError

This method is used via InnerAudioContext.onError(function listener).

Feature Description: Monitors audio playback error events.
Parameter and Description: function listener, the listener function for audio playback error events.
Object res

Attribute	Type	Description
errCode	number	

Valid values for errCode

Value	Description
10001	System error
10002	Network Error
10003	File Error
10004	Incorrect format.
-1	Unknown error.

offError

©2013-2022 Tencent Cloud. All rights reserved.

This method is used via InnerAudioContext.offError(function listener).

Feature Description: Removes the listener function for audio playback error events.
Parameter and Description: function listener, the listener function passed in by onError. If this parameter is not passed in, all listener functions will be removed.
onWaitting

This method is used via InnerAudioContext.onWaiting(function listener).

Feature Description: Monitors audio loading events. It is triggered when the audio needs to stop and load due to insufficient data.
Parameters and Description: function listener, the listener function passed in by onWaiting. If this parameter is not passed in, all listener functions will be removed.
offWaitting

This method is used via InnerAudioContext.offWaiting(function listener).

Feature Description: Removes the listener function for audio loading events.
Parameter and Description: function listener, the listener function for audio loading events.
onSeeking

This method is used via InnerAudioContext.onSeeking(function listener).

Feature Description: Monitors audio skipping operation events.
Parameters and Description: function listener, the listener function for audio skipping operation events.
offSeeking

This method is used via InnerAudioContext.offSeeking(function listener).

Feature Description: Removes the listener function for audio skipping operation events.
Parameter and Description: function listener, the listener function passed in by onSeeking. If this parameter is not passed in, all listener functions will be removed.

©2013-2022 Tencent Cloud. All rights reserved.

onSeeked

This method is used via InnerAudioContext.onSeeked(function listener).

Feature Description: Monitors events of completion of audio skipping operations.
Parameter and Description: function listener, the listener function for events of completion of audio skipping operations.
offSeeked

This method is used via InnerAudioContext.offSeeked(function listener).

Feature Description: Removes the listener function for events of completion of audio skipping operations.
Parameter and Description: function listener, the listener function passed in by onSeeked. If this parameter is not passed in, all listener functions will be removed.

©2013-2022 Tencent Cloud. All rights reserved.

WXML

Last updated：2024-07-12 19:11:27

createSelectorQuery

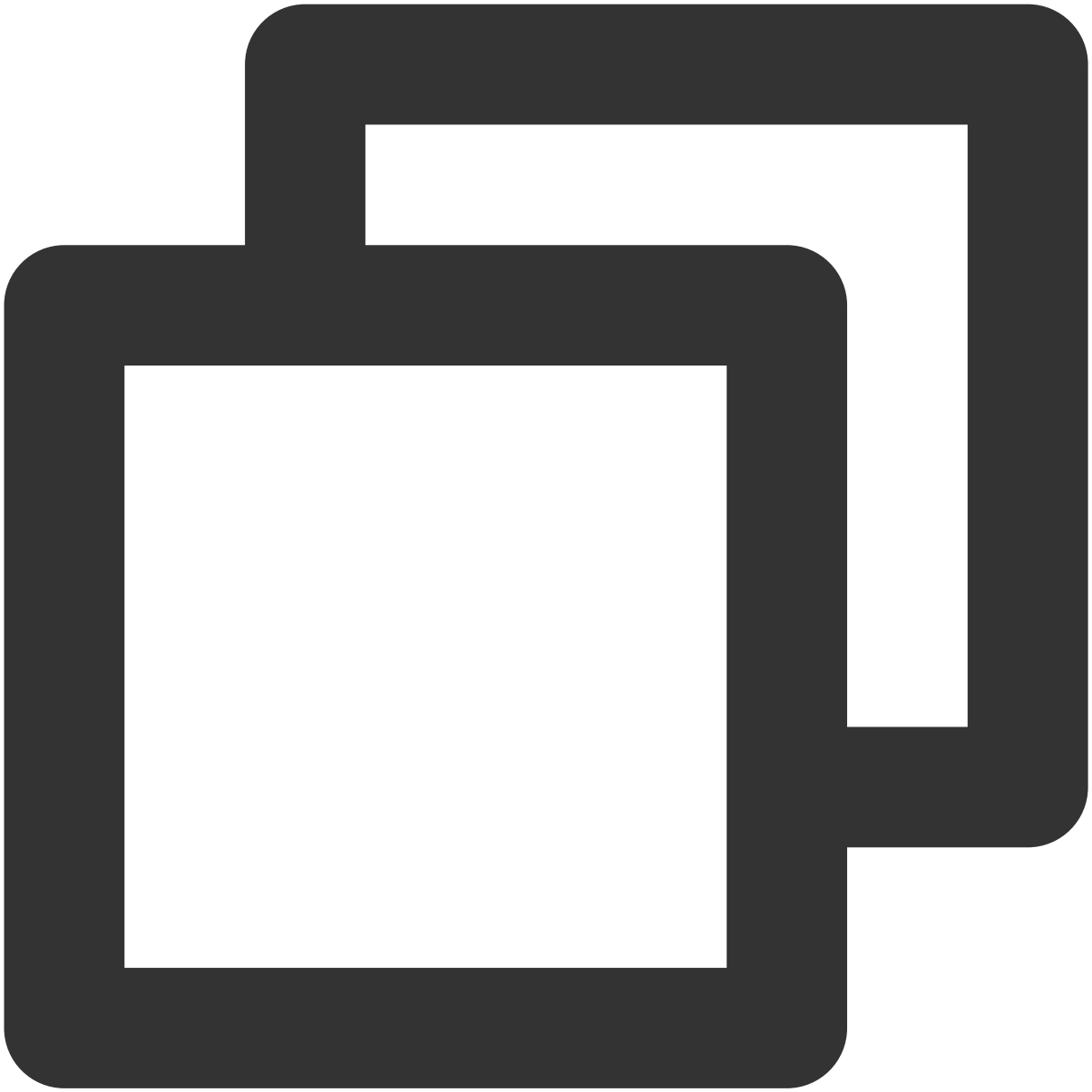
The application of this API is through SelectorQuery wx.createSelectorQuery().

Function Description: Returns an instance of the SelectorQuery object. In custom components or pages containing custom components, replace with `this.createSelectorQuery()` .

Return Value: SelectorQuery.

Sample Code:

©2013-2022 Tencent Cloud. All rights reserved.



```
const query = wx.createSelectorQuery()
query.select('#the-id').boundingClientRect()
query.selectViewport().scrollOffset()
query.exec(function (res) {
  res[0].top // The upper boundary coordinate of the #the-id node.
  res[1].scrollTop // The vertical scroll position of the display area.
})
```

©2013-2022 Tencent Cloud. All rights reserved.

createIntersectionObserver

The usage of this API is IntersectionObserver wx.createIntersectionObserver(Object component, Object options).

Function Description: Creates and returns an instance of the IntersectionObserver object. In custom components or pages containing custom components, `this.createIntersectionObserver([options])` should be used as a substitute.
Parameters and Description: Object this
Object component, a custom component instance;
Object options, with the following legal values:

Attributes	Types	Default value	Required	Note
thresholds	Array.<number>	[0]	Not required	An array of numerical values, encompassing all thresholds.
initialRatio	number	0	Not required	The initial intersection ratio, if the detected intersection ratio at the time of invocation is not equivalent to this value and reaches the threshold, it will trigger a callback function of the listener once.
observeAll	boolean	false	Not required	Whether to observe multiple target nodes simultaneously (rather than a single one). If set to true, the targetSelector of observe will select multiple nodes (be aware: selecting too many nodes at once may impact rendering performance).

Return Value: IntersectionObserver.

IntersectionObserver

IntersectionObserver object, used to infer whether certain nodes can be seen by the user and what percentage can be seen by the user.

.relativeTo

©2013-2022 Tencent Cloud. All rights reserved.

The method is utilized as follows: IntersectionObserver.relativeTo(string selector, Object margins)

Function Description: Utilize a selector to designate a node, serving as one of the reference areas.

Parameters and Description:

string selector, the selector;

Object margins, the boundaries used to expand (or contract) the layout area of the reference node.

Attributes	Types	Default value	Required	Note
left	number	-	Not required	The left boundary of the node's layout area.
right	number	-	Not required	The right boundary of the node's layout area.
top	number	-	Not required	The upper boundary of the node's layout area.
bottom	number	-	Not required	The lower boundary of the node's layout area.

.relativeToViewport

The method is utilized as IntersectionObserver.relativeToViewport(Object margins).

Function Description: Designates the page display area as one of the reference regions.

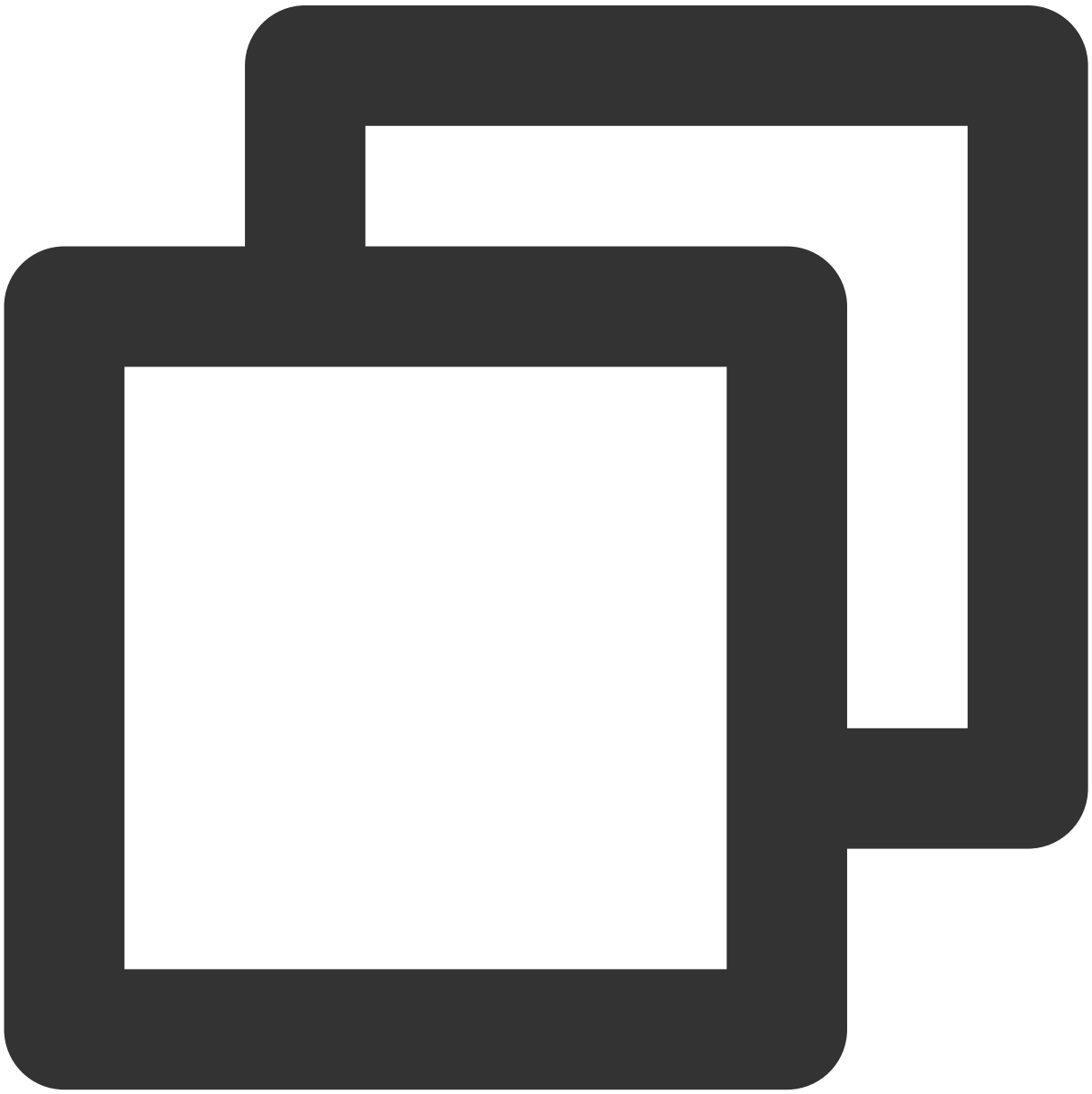
Parameters and Description: Object margins, utilized to expand (or contract) the boundaries of the reference node's layout area.

Attributes	Types	Default value	Required	Note
left	number	-	Not required	The left boundary of the node's layout area.
right	number	-	Not required	The right boundary of the node's layout area.
top	number	-	Not required	The upper boundary of the node's layout area.
bottom	number	-	Not required	The lower boundary of the node's layout area.

©2013-2022 Tencent Cloud. All rights reserved.

Example Code

In the following example code, if the target node (specified by the selector .target-class) enters 100px below the display area, the callback function will be triggered.



```
Page({
  onLoad() {
    wx.createIntersectionObserver().rela
  onLoad: function(){
    wx.createIntersectionObserver().relativeToViewport({bottom: 100}).observe('.tar
```

©2013-2022 Tencent Cloud. All rights reserved.

```
        res.intersectionRatio // the ratio of the intersection region to the layout r
        res.intersectionRect // intersection region
        res.intersectionRect.left // coordinates of the left boundary of the intersec
        res.intersectionRect.top // the upper boundary of the intersection region
        res.intersectionRect.width // width of the intersection region
        res.intersectionRect.height // height of the intersection region
      })
    }
})
```

.disconnect

The method is utilized as IntersectionObserver.disconnect().

Function Description: Ceases monitoring, the callback function will no longer be triggered.

.observe

The method is utilized as IntersectionObserver.observe(string targetSelector, function callback).

Function Description: Designates the target node and commences monitoring for changes in intersection status.
Parameters and Description:
string targetSelector, selector;
function callback, a callback function that listens for changes in the intersection state, with the following arguments Object res:
Parameters for the Callback Function Monitoring Intersection Status Changes: Object res.

Structural attributes	Types	Note
intersectionRatio	number	Intersection Ratio
intersectionRect	object	Boundary of the Intersection Area
boundingClientRect	object	Target Boundary
relativeRect	object	Boundary of the Reference Area
time	number	Timestamp during Intersection Detection

Structure of res.intersectionRect

©2013-2022 Tencent Cloud. All rights reserved.

Structural attributes	Types	Note
left	number	Left Boundary
right	number	Right Boundary
top	number	Upper Boundary
bottom	number	Lower Boundary

Structure of res. boundingClientRect

Structural attributes	Types	Note
left	number	Left Boundary
right	number	Right Boundary
top	number	Upper Boundary
bottom	number	Lower Boundary

Structure of res.relativeRect

Structural attributes	Types	Note
left	number	Left Boundary
right	number	Right Boundary
top	number	Upper Boundary
bottom	number	Lower Boundary

NodesRef

An object used to obtain information about a WXML node.

.fields

The method is utilized as NodesRef.fields(Object fields, function callback)

Function Description: Get information about the node. The fields to be fetched are specified in fields. The return value is the selectorQuery corresponding to nodesRef

Parameters and Description:

Object fields, the legal parameters are as follows

Attributes	Types	Default value	Required	Note
id	boolean	false	Not required	Whether to return the node id
dataset	boolean	false	Not required	Whether to return the node dataset
rect	boolean	false	Not required	Whether to return the layout position of the node (left, right, top, bottom)
size	boolean	false	Not required	Whether to return the dimensions of the node (width, height)
scrollOffset	boolean	false	Not required	Whether to return the `scrollLeft scrollTop` of the node, the node must be a `scroll-view` or `viewport`
properties	Array.<string>	-	Not required	Specify a list of attribute names to return the current attribute values of the corresponding node (only the regular attribute values marked in the component documentation can be obtained, the attribute values of id, class, style, and event binding cannot be retrieved)
computedStyle	Array.<string>	-	Not required	Specify a list of style names to return the current values of the corresponding style names for the node
context	boolean	false	Not required	Whether to return the corresponding Context object of the node

Note:

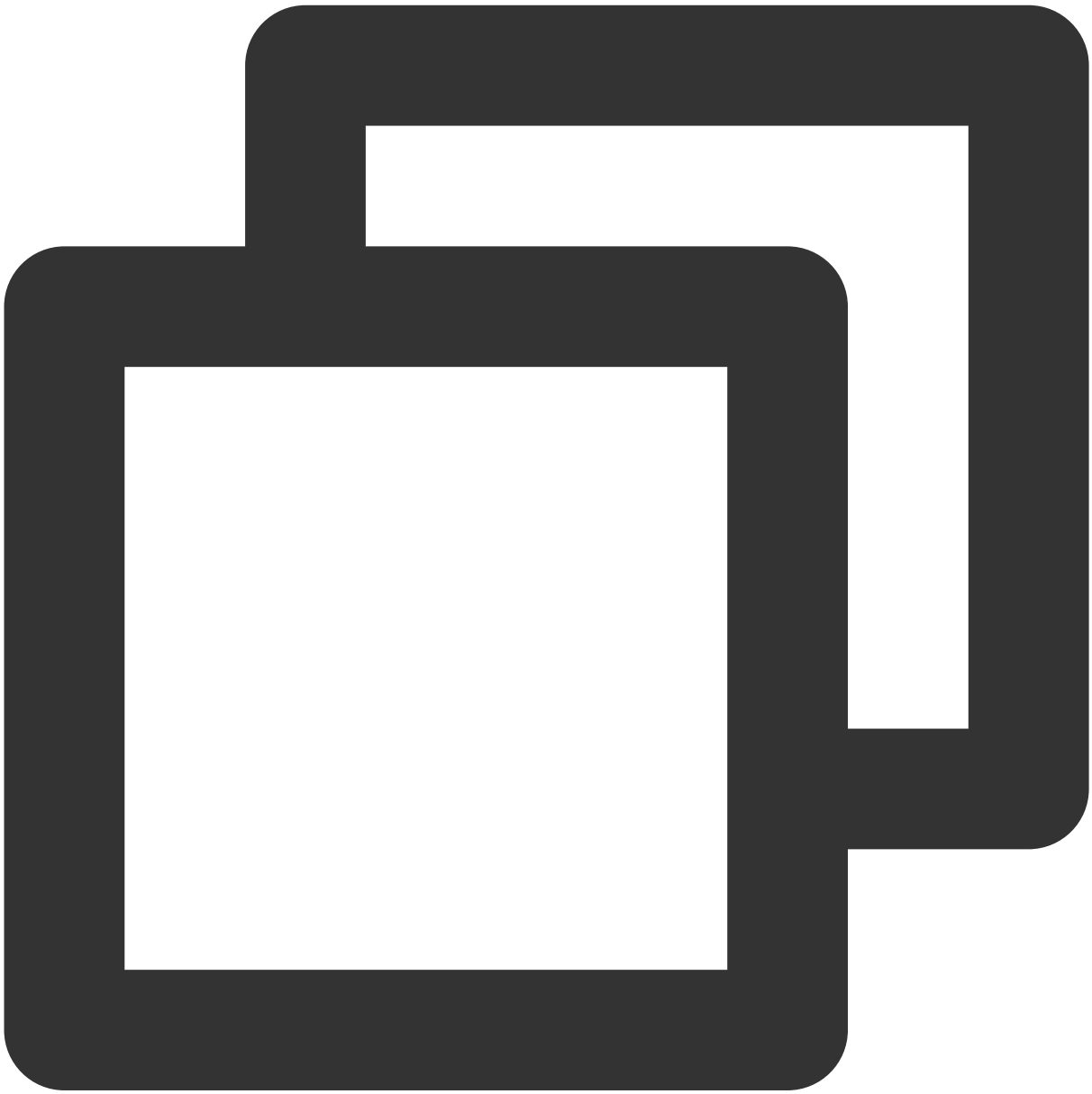
The priority of computedStyle is higher than size. If width/height is specified in computedStyle and size: true is passed in at the same time, the width/height obtained from computedStyle will be returned preferentially.

function callback, callback function, argument Object res is node related information.

Return Value: The selectorQuery corresponding to nodesRef.

©2013-2022 Tencent Cloud. All rights reserved.

Sample Code



```
Page({
getFields () {
  wx.createSelectorQuery().select('#the-id').fields({
    dataset: true,
    size: true,
    scrollOffset: true,
    properties: ['scrollX', 'scrollY'],
    computedStyle: ['margin', 'backgroundColor'],
    context: true, }, function (res) { {
```

©2013-2022 Tencent Cloud. All rights reserved.

```
    }, function (res) {
      res.dataset // the node's dataset
      res.width // width of the node
      res.height // the height of the node
      res.scrollLeft // The horizontal scroll position of the node.
      res.scrollTop // The vertical scroll position of the node.
      res.scrollX // The current value of the node's scroll-x property.
      res.scrollY // The current value of the node's scroll-y property.
      // This returns the name of the style to be returned.
      res.margin
      res.backgroundColor
      res.context // The node's corresponding Context object.
      res.ref // The Ref object for the node.
    res.ref // Ref object for the node }).exec()
  }
})
```

.boundingClientRect

The method is used as follows:SelectorQuery NodesRef.boundingClientRect(function callback)

Function Description: Adds a query request for the layout position of the node. Relative to the display area, in pixels. Its function is similar to the `getBoundingClientRect` of the DOM.

Parameters and Description: function callback**.** This is a callback function. After executing the SelectorQuery.exec method, the node information will be returned in the callback.

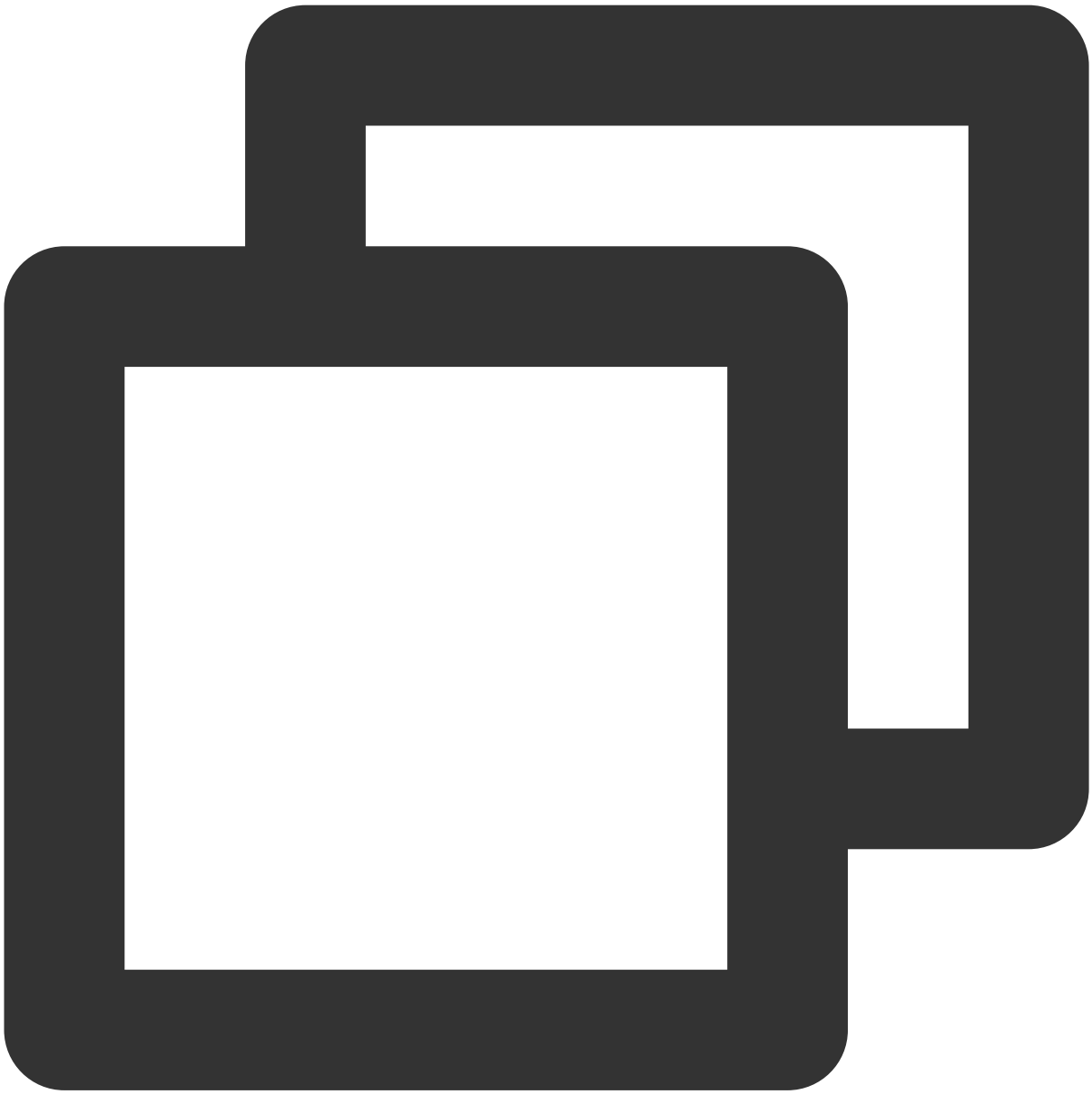
Object res

Attributes	Types	Note
id	string	Node's ID
dataset	Object	Node's dataset
left	number	Node's left boundary coordinate
right	number	Node's right boundary coordinate
top	number	Node's upper boundary coordinate
bottom	number	Node's lower boundary coordinate
width	number	Node's width
height	number	Node's height

©2013-2022 Tencent Cloud. All rights reserved.

Return Value: SelectorQuery.

Sample Code:



```
Page({
  getRect() {
    wx.createSelectorQuery().select('#the-id').boundingClientRect(function (rect) {
      rect.id // Node's ID
      rect.dataset // Node's dataset
      rect.left // Node's left boundary coordinate
      rect.right // Node's right boundary coordinate
      rect.top // Node's upper boundary coordinate
```

©2013-2022 Tencent Cloud. All rights reserved.

```
        rect.bottom // Node's lower boundary coordinate
        rect.width // Node's width
        rect.height // Node's height
      }).exec()
    },
    getAllRects() {
      wx.createSelectorQuery().selectAll('.a-class').boundingClientRect(function (rec
        rects.forEach(function (rect) {
          rect.id // Node's ID
          rect.dataset // Node's dataset
          rect.left // Node's left boundary coordinate
          rect.right // Node's right boundary coordinate
          rect.top // Node's upper boundary coordinate
          rect.bottom // Node's lower boundary coordinate
          rect.width // Node's width
          rect.height // Node's height
        })
      }).exec()
    }
  })
```

.scrollOffset

The method is utilized as SelectorQuery NodesRef.scrollOffset(function callback)

Function Description: Adds a query request for the node's scroll position. Measured in pixels. The node must be a `scroll-view` or a `viewport` .

Parameters and Description: function callback, a callback function, after executing the SelectorQuery.exec method, the node information will be returned in the callback**.**

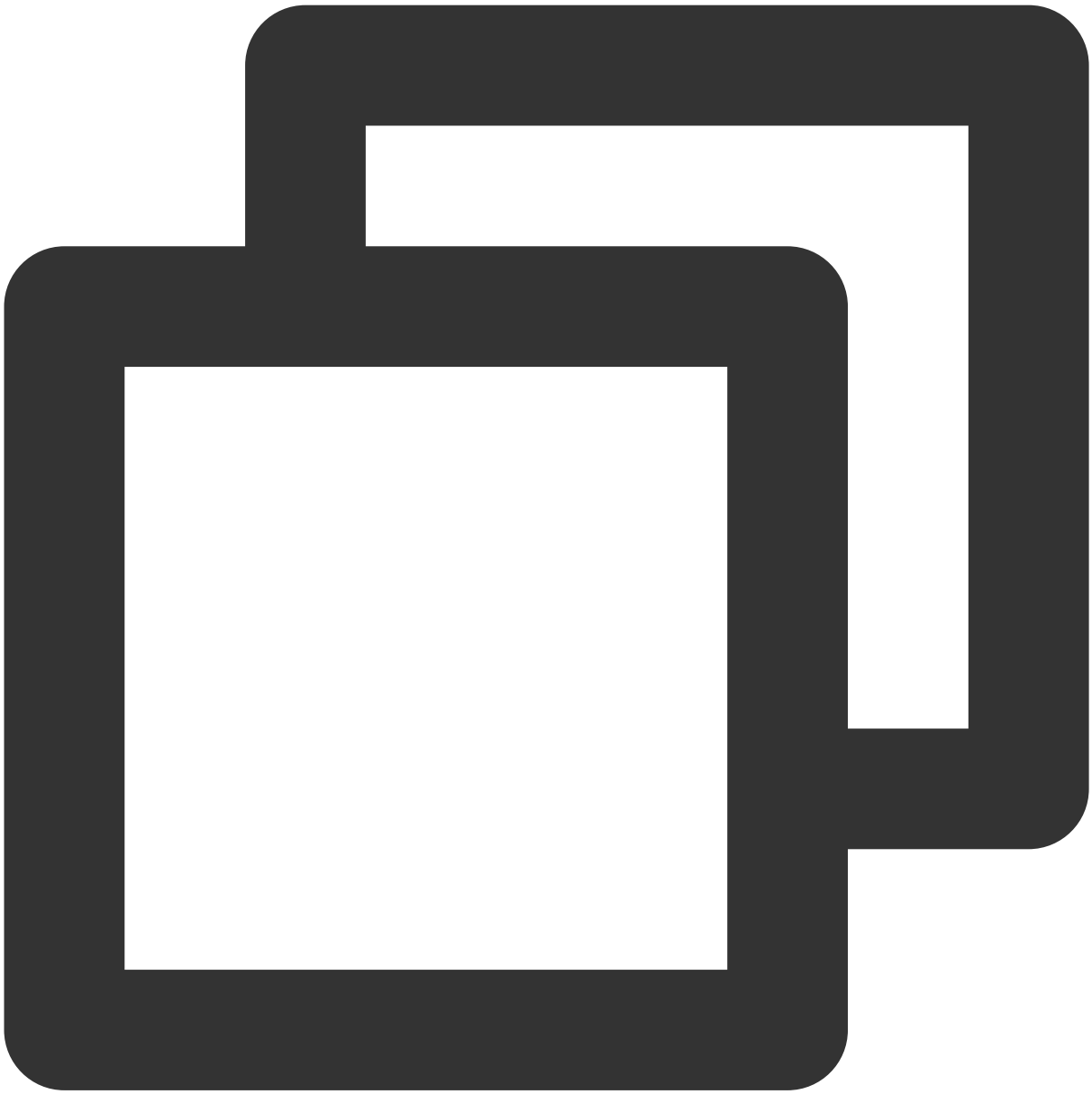
Object res

Attributes	Types	Note
id	string	Node's ID
dataset	Object	Node's dataset
scrollLeft	number	Node's horizontal scroll position
scrollTop	number	Node's vertical scroll position

Return Value: SelectorQuery.

©2013-2022 Tencent Cloud. All rights reserved.

Sample Code:



```
Page({
  getScrollOffset() {
    wx.createSelectorQuery().selectViewport().scrollOffset(function (res) {
      res.id // Node's ID
      res.dataset // Node's dataset
      res.scrollLeft // Node's horizontal scroll position
      res.scrollTop // Node's vertical scroll position
    }).exec()
  }
```

©2013-2022 Tencent Cloud. All rights reserved.

```
})
```

.context

The method is utilized as follows: SelectorQuery NodesRef.context(function callback)

Function Description: Adds a query request for the Context object of the node. Currently supports the acquisition of MapContext.

Parameters and Description: function callback, a callback function that returns node information after executing the SelectorQuery.exec method.

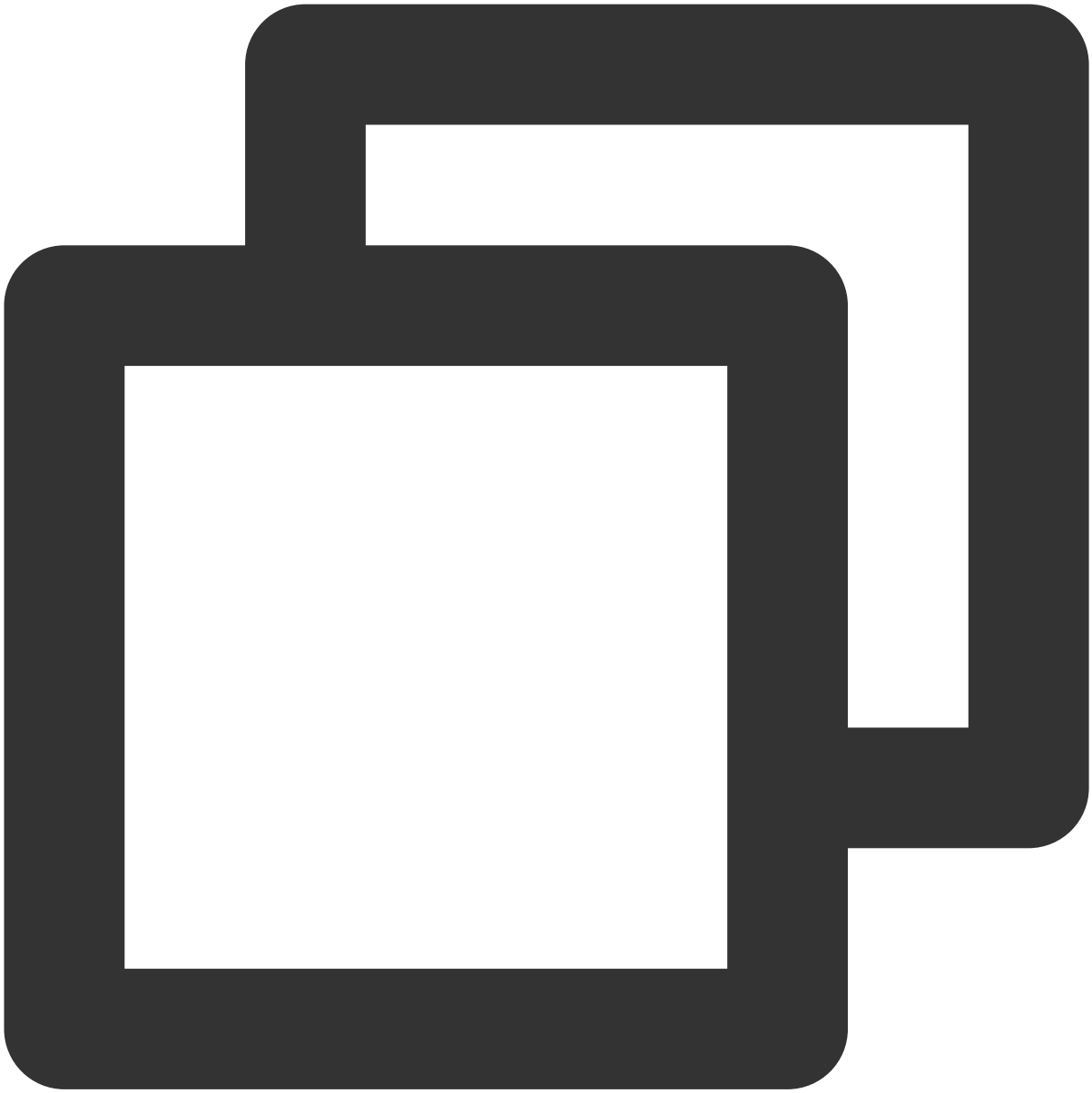
Object res。

Attributes	Types	Note
context	Object	Context object corresponding to the node

Return Value: SelectorQuery.

Sample Code:

©2013-2022 Tencent Cloud. All rights reserved.



```
Page({
  getContext() {
    wx.createSelectorQuery().select('.the-video-class').context(function (res) {
      console.log(res.context) // Context object corresponding to the node. For ins
    }).exec()
  }
})
```

.node

©2013-2022 Tencent Cloud. All rights reserved.

The method is employed as follows: SelectorQuery NodesRef.node(NodesRef.nodeCallback callback)

Function Description: Get Node node instance. Currently supports ScrollViewContext and Canvas.

Parameters and Description: function callback, a callback function that returns node information upon the execution of the SelectorQuery.exec method**.**

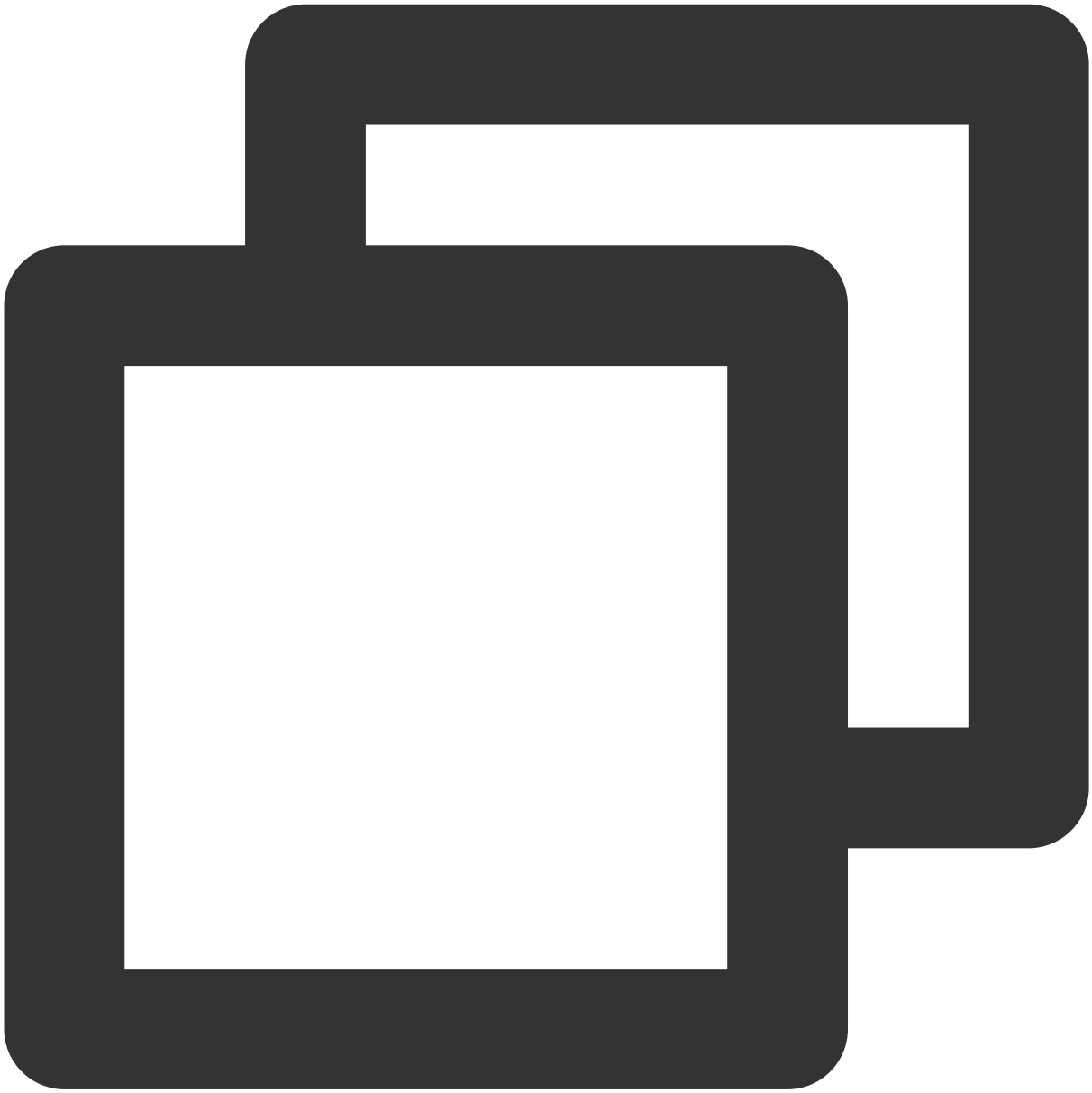
Object res

Attributes	Types	Note
node	Object	Node instance corresponding to the node

Return Value: SelectorQuery.

Sample Code:

©2013-2022 Tencent Cloud. All rights reserved.



```
Page({
  getNode() {
    wx.createSelectorQuery().select('.canvas').node(function(res){
      console.log(res.node) // Canvas instance corresponding to the node.
    }).exec()
  }
})
```

©2013-2022 Tencent Cloud. All rights reserved.

SelectorQuery

Object for querying node information.

.exec

The method is utilized as follows: NodesRef SelectorQuery.exec(function callback)

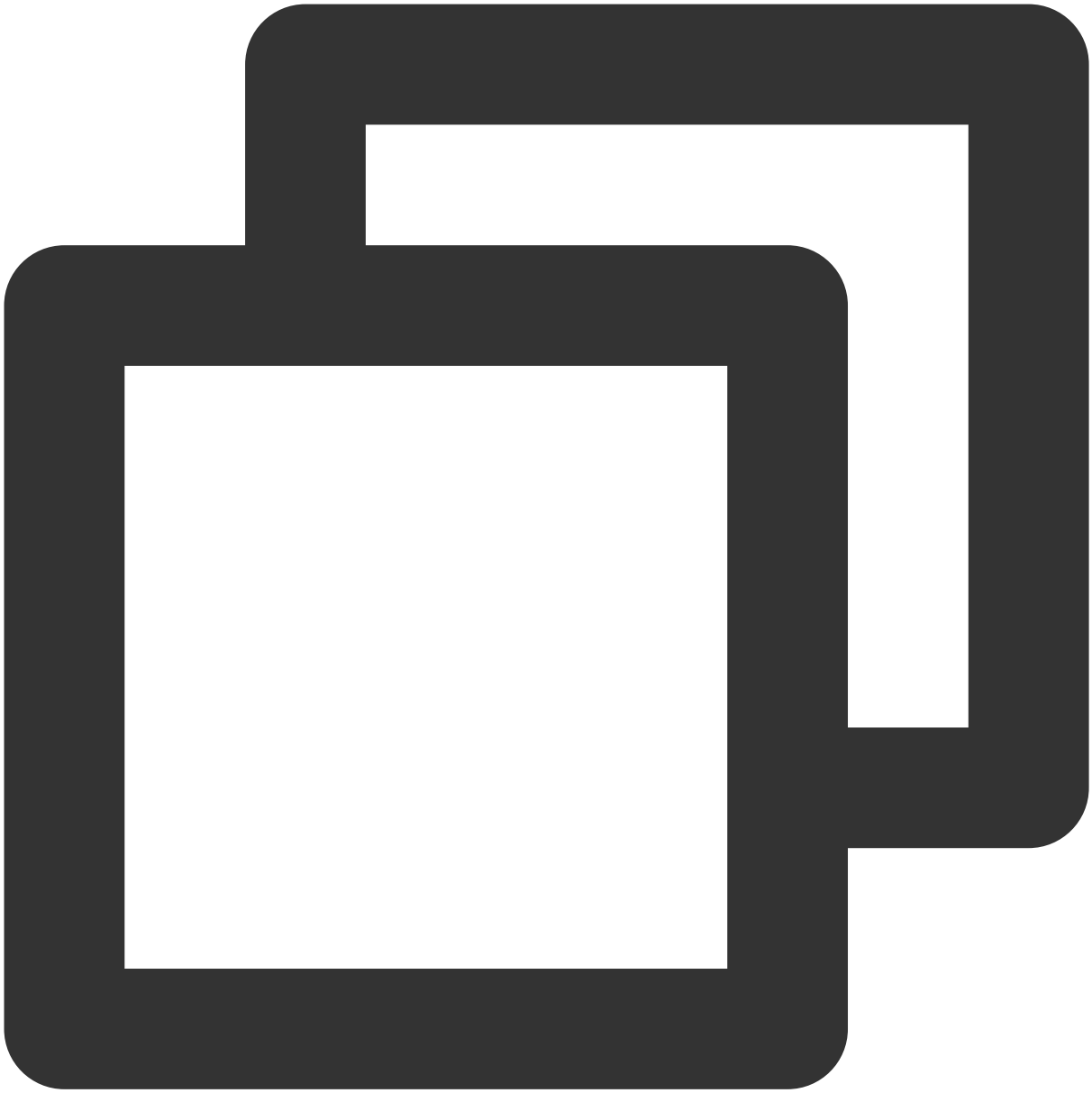
Function Description: Executes all requests. The results of the requests are arranged in an array according to the order of the requests and returned in the first parameter of the callback.
Parameters and Description: function callback, a callback function.
Return Value: NodesRef.

.in

The method is utilized as follows: SelectorQuery SelectorQuery.in(Component component)

Function Description: Modifies the selection range of the selector to be within the custom component, component. (Initially, the selector only selects nodes within the page range and does not select any nodes within custom components).
Parameters and Description: Component component, a custom component instance**.**
Return Value: SelectorQuery.
Sample Code:

©2013-2022 Tencent Cloud. All rights reserved.



```
Component({
  queryMultipleNodes() {
    const query = wx.createSelectorQuery().in(this)
    query.select('#the-id').boundingClientRect(function (res) {
      res.top // The upper boundary coordinate of the #the-id node within this comp
    }).exec()
  }
})
```

.select

©2013-2022 Tencent Cloud. All rights reserved.

The method is utilized as follows: NodesRef SelectorQuery.select(string selector)

Function Description: Selects the first node that matches the selector, selector, on the current page. Returns an instance of a NodesRef object, which can be used to retrieve node information.
Parameters and Description: string selector, the selector**.**
Return Value: NodesRef
Selector Syntax: The selector is similar to a CSS selector, but only supports the following syntax.
ID Selector: #the-id.
Class Selector (multiple can be specified consecutively): .a-class.another-class.
Child Selector: .the-parent > .the-child.
Descendant Selector: .the-ancestor .the-descendant.
Cross-component Descendant Selector: .the-ancestor >>> .the-descendant.
Union of Multiple Selectors: #a-node, .some-other-nodes.

.selectAll

The method is utilized as follows: NodesRef SelectorQuery.selectAll(string selector)

Function Description: Selects all nodes matching the selector 'selector' within the current page.
Parameters and Description: string selector, the selector**.**
Return Value: NodesRef
Selector Syntax: The selector is similar to a CSS selector, but only supports the following syntax.
ID Selector: #the-id.
Class Selector (multiple can be specified consecutively): .a-class.another-class.
Child Selector: .the-parent > .the-child.
Descendant Selector: .the-ancestor .the-descendant.
Cross-component Descendant Selector: .the-ancestor >>> .the-descendant.
Union of Multiple Selectors: #a-node, .some-other-nodes.

.selectViewport

The method is utilized as follows: NodesRef SelectorQuery.selectViewport()

©2013-2022 Tencent Cloud. All rights reserved.

Function Description: Selects the display area. This can be used to obtain information such as the size of the display area and the scroll position.
Return Value: NodesRef

MediaQueryObserver

MediaQueryObserver object, utilized for monitoring changes in the media query status, such as whether the interface's length and width are within a specified range.

.disconnect

The method is utilized as follows: MediaQueryObserver.disconnect()

Function Description: Ceases monitoring. The callback function will no longer be triggered.

.observe

The method is utilized as follows: MediaQueryObserver.observe(Object descriptor, function callback)

Function Description: Initiates monitoring of media query changes.
Parameters and Descriptions:
Object descriptor, a descriptor for the media query.

Attributes	Types	Default value	Required	Note
minWidth	number	-	Supported	Minimum page width (in pixels)
maxWidth	number	-	Supported	Maximum page width (in pixels)
width	number	-	Supported	Page width (in pixels)
minHeight	number	-	Supported	Minimum page height (in pixels)
maxHeight	number	-	Supported	Maximum page height (in pixels)
height	number	-	Supported	Page height (in pixels)
orientation	string	-	Supported	Screen orientation (landscape or portrait)

©2013-2022 Tencent Cloud. All rights reserved.

function callback, a callback function that listens for changes in the state of the media query. The arguments Object res are as follows:

Attributes	Types	Note
matches	boolean	Whether the current status of the page satisfies the specified media query.

©2013-2022 Tencent Cloud. All rights reserved.

File

Last updated：2024-07-12 19:10:17

saveFile

This method is used via wx.saveFile(Object object)

Note:

The storage limit for local files is set at 10 MB.

Feature Description: Saves files to local storage.

Note:

Upon successful invocation, saveFile will relocate the temporary file, rendering the input tempFilePath unusable.

Parameter and Description:

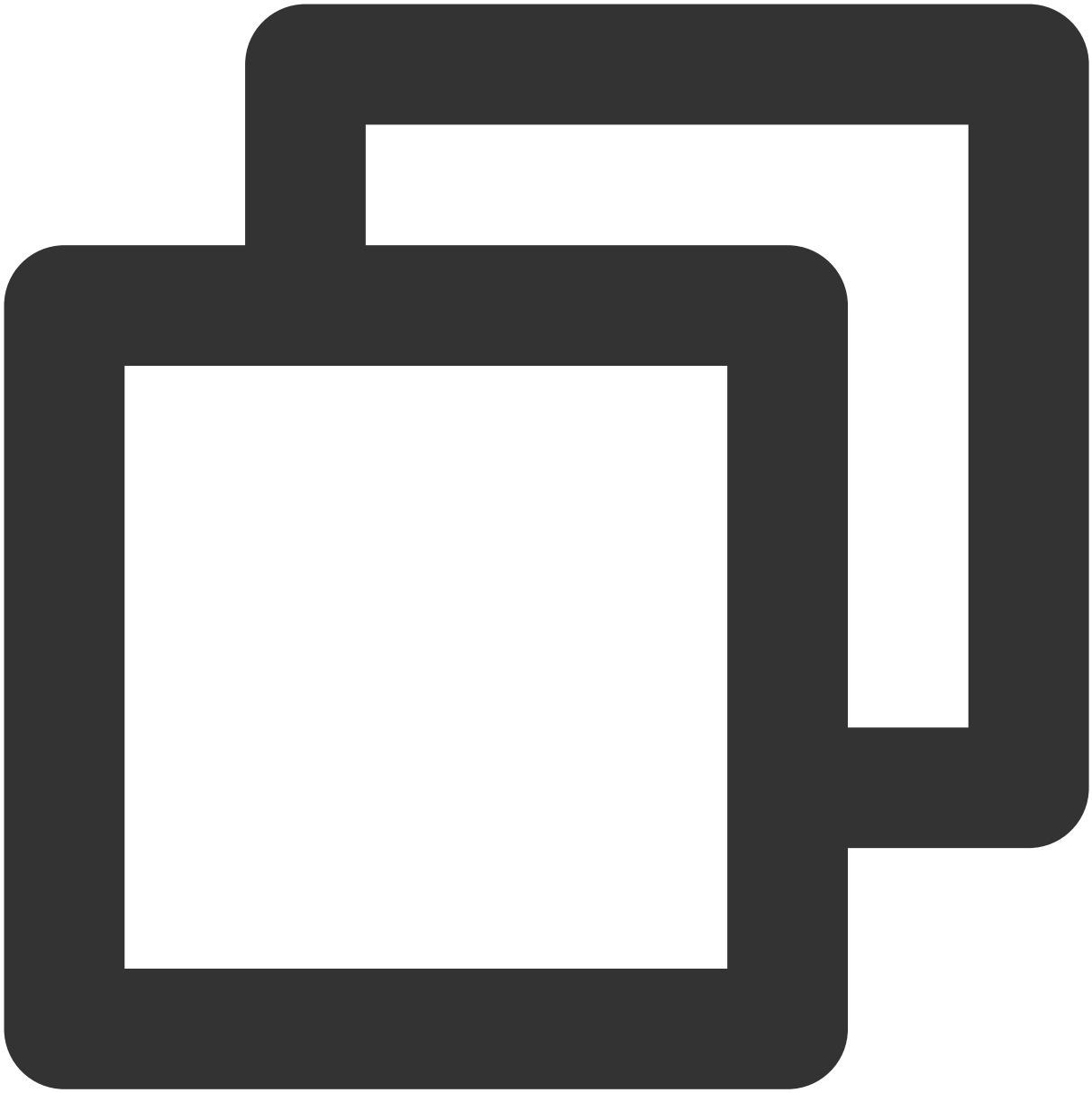
Attribute	Type	Default value	Required	Description
tempFilePath	string	-	Yes	Temporary path for the files that need to be stored
success	function	-	No	Callback Function of Successful Interface Call
fail	function	-	No	Callback Function of Failing Interface Call
complete	function	-	No	Callback function executed upon the completion of the interface invocation (both successful and unsuccessful invocations)

Parameters for object.success callback function: Object res.

Attribute	Type	Description
savedFilePath	string	File path after storage

Sample Code

©2013-2022 Tencent Cloud. All rights reserved.



```
  wx.chooseImage({
      success(res) {
    const tempFilePaths = res.tempFilePaths
    wx.saveFile({
      tempFilePath: tempFilePaths[0],
      success(res) {
        const savedFilePath = res.savedFilePath
  }
})
  }
})
```

©2013-2022 Tencent Cloud. All rights reserved.

removeSavedFile

This method is used via wx.removeSavedFile(Object object).

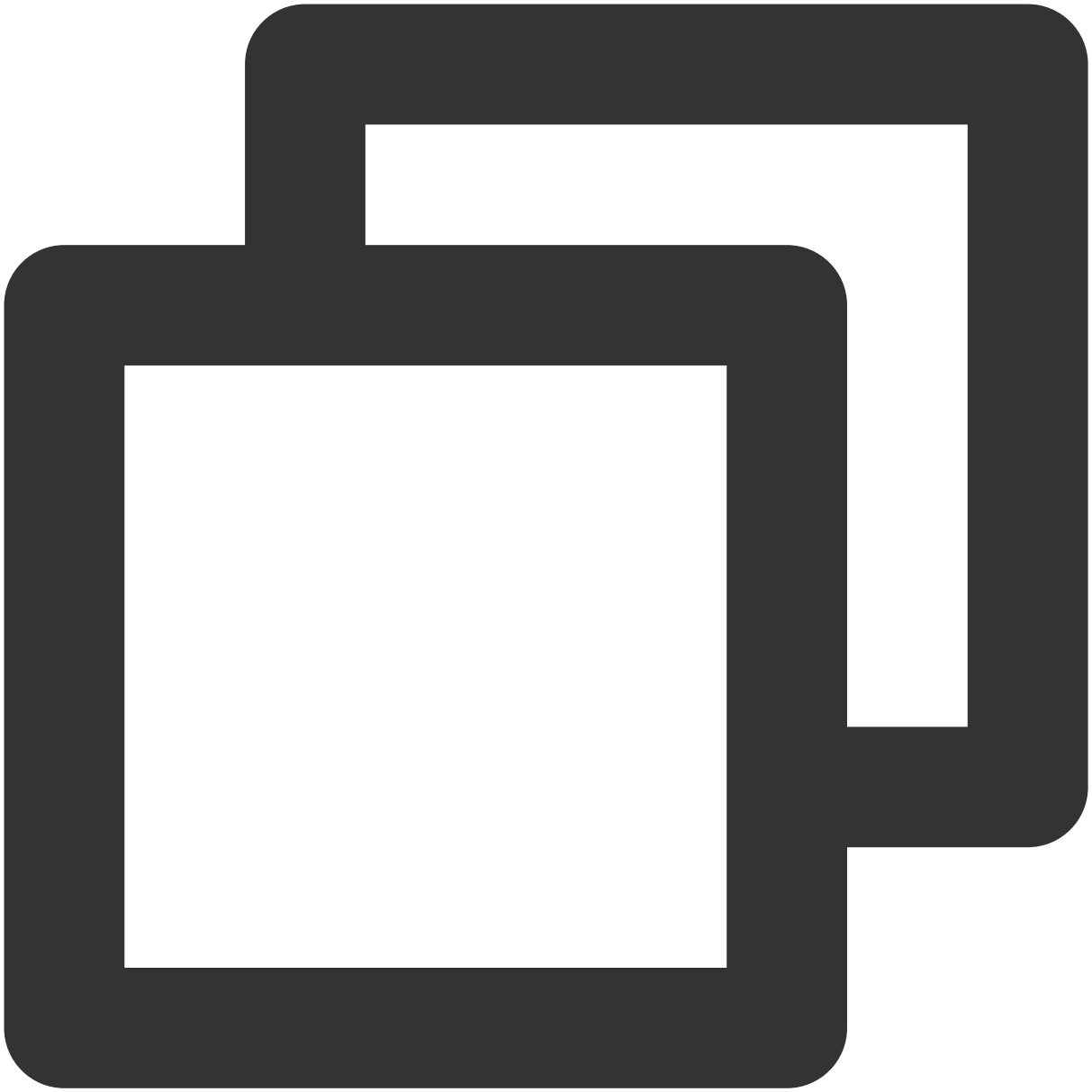
Feature Description: Deletes local cache files.

Parameter and Description:

Attribute	Type	Default value	Required	Description
filePath	string	-	Yes	Path for the files that need to be deleted.
success	function	-	No	Callback Function of Successful Interface Call
fail	function	-	No	Callback Function of Failing Interface Call
complete	function	-	No	Callback function executed upon the completion of the interface invocation (both successful and unsuccessful invocations)

Sample Code

©2013-2022 Tencent Cloud. All rights reserved.



```
wx.getSavedFileList({
  success(res) {
    if (res.fileList.length > 0) {
      wx.removeSavedFile({
        filePath: res.fileList[0].filePath,
        complete(res) {
          console.log(res)
        }
      })
    }
  }
```

©2013-2022 Tencent Cloud. All rights reserved.

```
})
```

openDocument

This API is used via wx.openDocument(Object object).

Feature Description: Opens a document on a new page.

Parameter and Description:

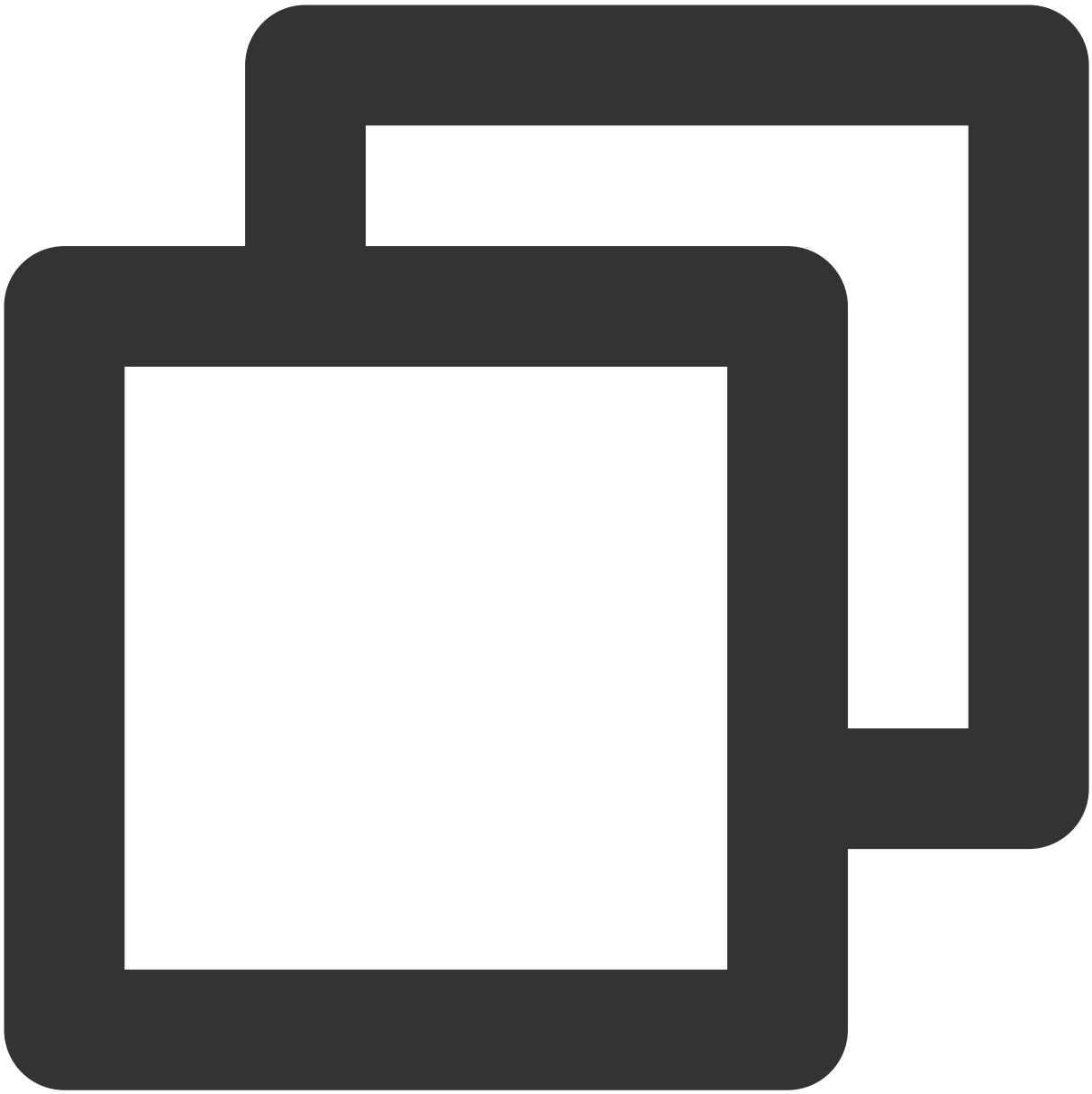
Attribute	Type	Default value	Required	Description
filePath	string	-	Yes	The path to the file (local path), which can be obtained from downloadFile.
fileType	string	-	No	File type, used to open the file of specified type.
success	function	-	No	Callback Function of Successful Interface Call
fail	function	-	No	Callback Function of Failing Interface Call
complete	function	-	No	Callback function executed upon the completion of the interface invocation (both successful and unsuccessful invocations)

Valid values of object.fileType

Value	Description
doc	DOC format
docx	DOCX format
xls	XLS format
xlsx	XLSX format
ppt	PPT format
pptx	PPTX format
pdf	PDF format

©2013-2022 Tencent Cloud. All rights reserved.

Sample Code



```
wx.downloadFile({
  // Example url, not real
  url: 'https://example.com/somefile.pdf',
  success(res) {
    const filePath = res.tempFilePath
    wx.openDocument({
      filePath,
      success(res) {
        console.log('Opened document successfully')
```

©2013-2022 Tencent Cloud. All rights reserved.

```
        }
      })
    }
})
```

getSavedFileList

This method is used via wx.getSavedFileList(Object object).

Feature Description: Gets the list of locally cached files saved under this mini program.

Parameter and Description:

Attribute	Type	Default value	Required	Description
success	function	-	No	Callback Function of Successful Interface Call
fail	function	-	No	Callback Function of Failing Interface Call
complete	function	-	No	Callback function executed upon the completion of the interface invocation (both successful and unsuccessful invocations)

Parameters for object.success callback function: Object res

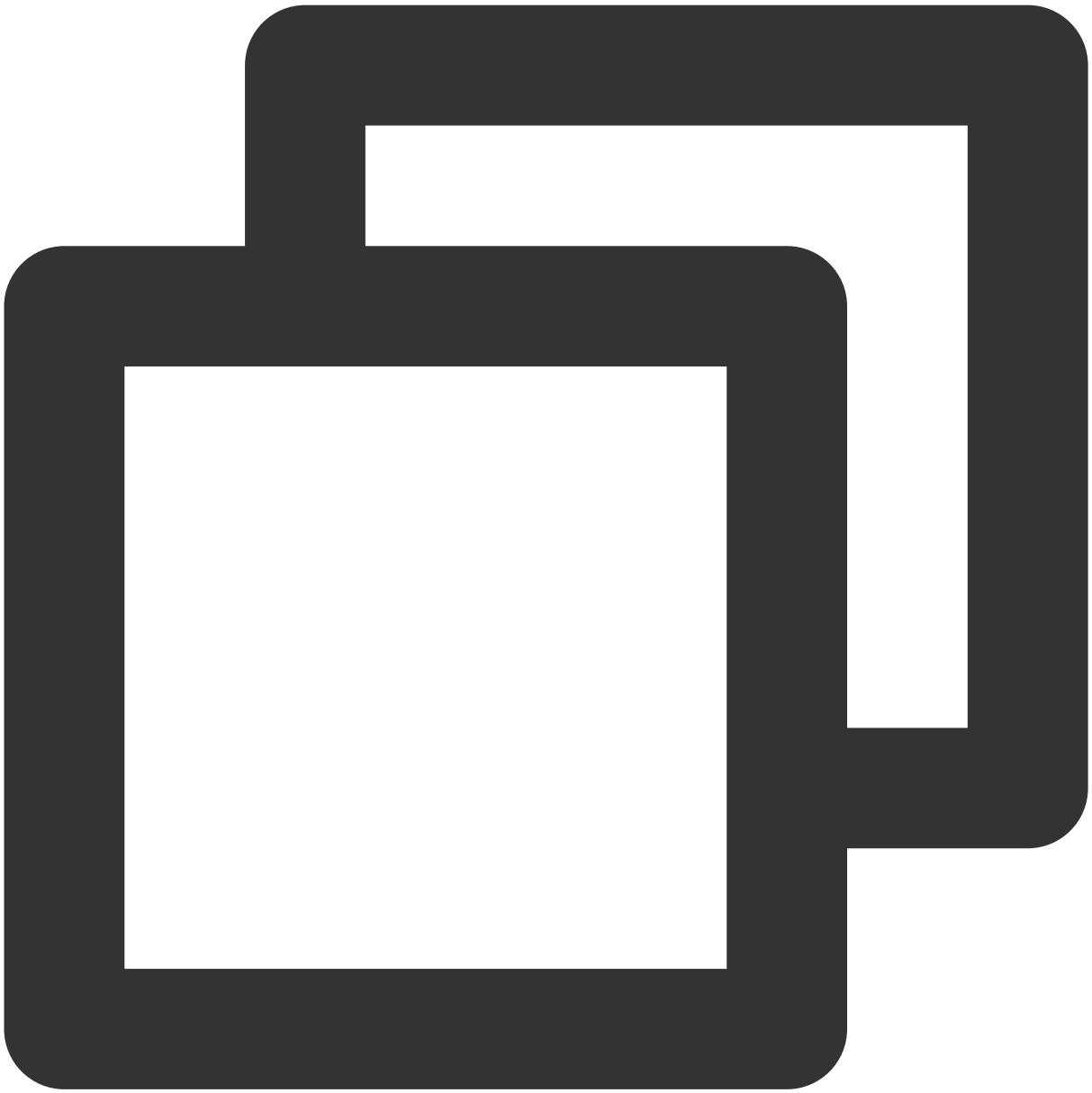
Attribute	Type	Description
fileList	Array.<Object>	File array, with each item being a FileItem.

Structure of res.fileList

Attribute	Type	Description
filePath	string	Local path
size	number	Local file size, measured in bytes.
createTime	number	Timestamp for file saving, denoting the number of seconds from 08:00:00 on Jan. 01, 1970 to the current time.

©2013-2022 Tencent Cloud. All rights reserved.

Sample Code



```
wx.getSavedFileList({
  success(res) {
    console.log(res.fileList)
  }
})
```

©2013-2022 Tencent Cloud. All rights reserved.

getSavedFileInfo

This method is used via wx.getSavedFileInfo(Object object).

Feature Description: Gets the file information of local files. This interface can only be used to obtain files that have been saved locally. If you need to obtain temporary file information, use the wx.getFileInfo() interface.

Parameter and Description:

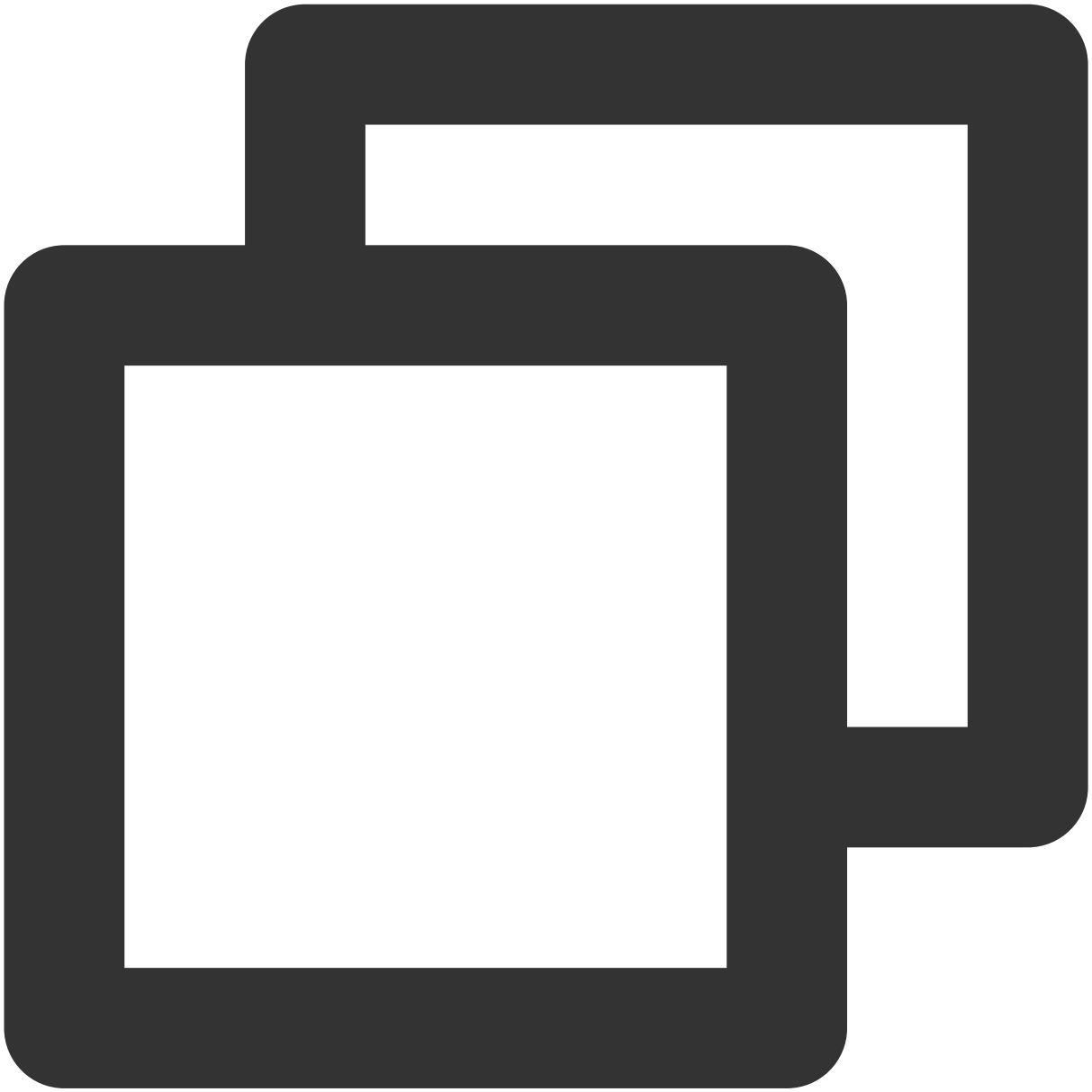
Attribute	Type	Default value	Required	Description
filePath	string	-	Yes	File path
success	function	-	No	Callback Function of Successful Interface Call
fail	function	-	No	Callback Function of Failing Interface Call
complete	function	-	No	Callback function executed upon the completion of the interface invocation (both successful and unsuccessful invocations)

Parameters for object.success callback function: Object res.

Attribute	Type	Description
size	number	File size measured in bytes.
createTime	number	Timestamp for file saving, denoting the number of seconds from 08:00:00 on Jan. 01, 1970 to the current time.

Sample Code

©2013-2022 Tencent Cloud. All rights reserved.



```
wx.getSavedFileList({
  success(res) {
    console.log(res.fileList)
  }
})
```

getFileInfo

©2013-2022 Tencent Cloud. All rights reserved.

The API is used via wx.getFileInfo(Object object).

Feature Description: Gets file information.

Parameter and Description: Object

Attribute	Type	Default value	Required	Description
filePath	string	-	Yes	Local File Path
digestAlgorithm	string	'md5'	No	Algorithm for calculating file digest.
success	function	-	No	Callback Function of Successful Interface Call
fail	function	-	No	Callback Function of Failing Interface Call
complete	function	-	No	Callback function executed upon the completion of the interface invocation (both successful and unsuccessful invocations)

Valid values of object.digestAlgorithm

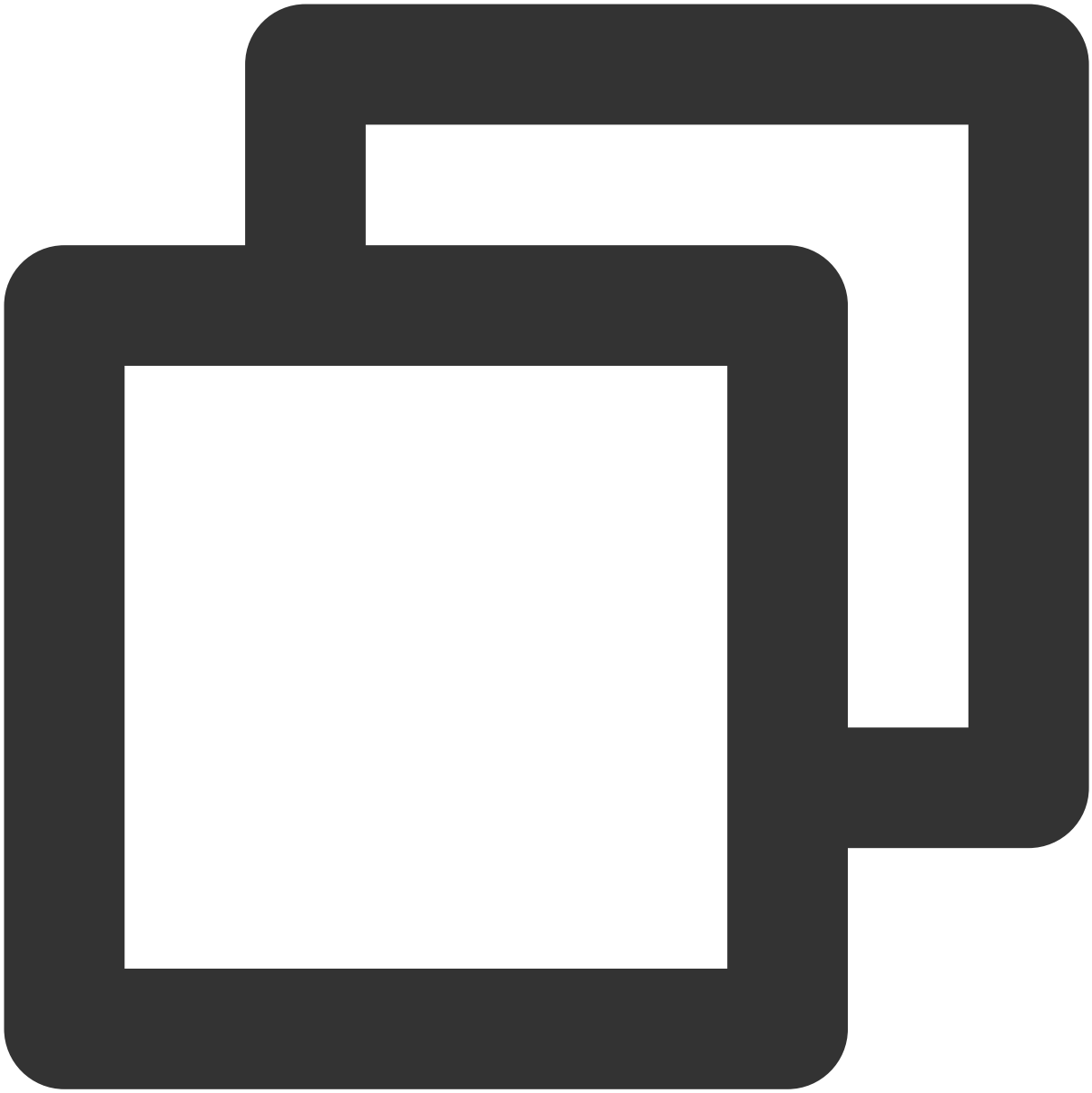
Value	Description
md5	MD5 Algorithm
sha1	SHA1 Algorithm

Parameters for object.success callback function: Object res.

Attribute	Type	Description
size	number	File size, measured in bytes.
digest	string	File digest calculated according to the provided digestAlgorithm

Sample Code

©2013-2022 Tencent Cloud. All rights reserved.



```
wx.getFileInfo({
  success(res) {
    console.log(res.size)
    console.log(res.digest)
  }
})
```

©2013-2022 Tencent Cloud. All rights reserved.

getFileSystemManager

The API is used via FileSystemManager wx.getFileSystemManager().

Feature Description: Gets the globally unique file manager.

Return Value： returning the FileSystemManager file manager.

FileSystemManager

File manager, available via wx.getFileSystemManager.

.access

This method is used via FileSystemManager.access(Object object).

Feature Description: Judges whether a file/directory exists.

Parameter and Description: Object

Attribute	Type	Default value	Required	Description
path	string	-	Yes	The file/directory path to be determined for existence.
success	function	-	No	Callback Function of Successful Interface Call
fail	function	-	No	Callback Function of Failing Interface Call
complete	function	-	No	Callback function executed upon the completion of the interface invocation (both successful and unsuccessful invocations)

Parameters for object.fail callback function: Object res

Attribute	Type	Description
errMsg	string	Error Message

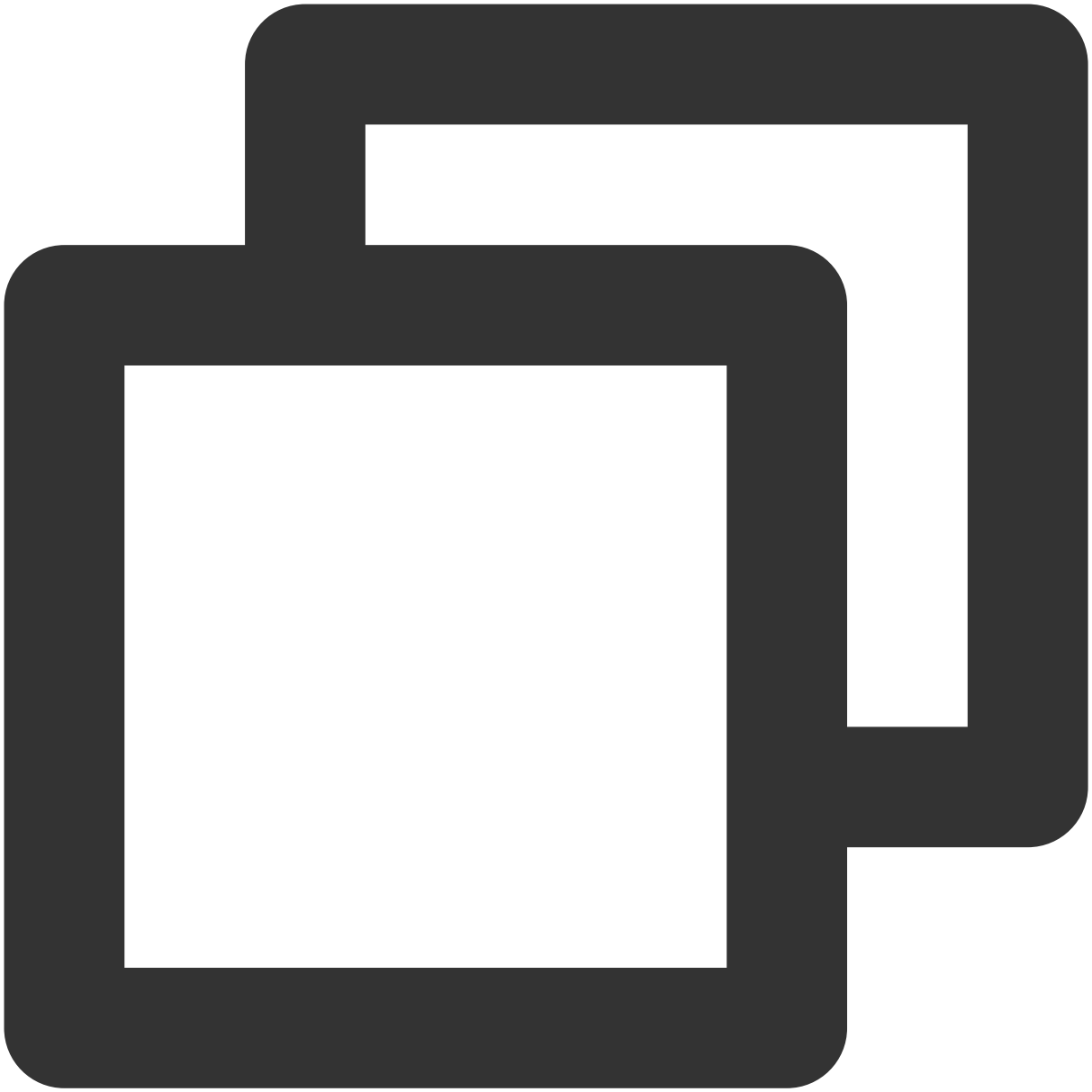
©2013-2022 Tencent Cloud. All rights reserved.

Valid values of res.errMsg

Value	Description
fail no such file or directory ${path}	The file/directory does not exist.
bad file descriptor	Invalid file descriptors
permission denied	Permission error, file is read-only or write-only
permission denied, cannot access file path	Destination path without access rights (usr directory)
not a directory	dirPath specifies that the path is not a directory, which is common when the parent path of the specified write path is a file.
Invalid argument	Invalid parameter to check if length or offset is out of bounds
directory not empty	directory not empty
the maximum size of the file storage limit is exceeded	the maximum size of the file storage limit is exceeded
base64 encode error	Character encoding conversion failure (e.g. base64 format error)
data to write is empty	data to write is empty
illegal operation on a directory	This must not be done for directories (e.g., the specified filePath is an existing directory)
file already exists ${dirPath}	A file or directory with the same name already exists
value of length is out of range	value of length is out of range
value of offset is out of range	value of offset is out of range
value of position is out of range	value of position is out of range

Sample Code

©2013-2022 Tencent Cloud. All rights reserved.



```
const fs = wx.getFileSystemManager()
// Determine if the file/directory exists
fs.access({
  path: `${wx.env.USER_DATA_PATH}/hello.txt`, success(res) {
  success(res) {
    // The file exists
    console.log(res)
  }, fail(res) { // File exists.
  fail(res) {
    // File does not exist or other error
    console.error(res)
```

©2013-2022 Tencent Cloud. All rights reserved.

```
  }
})

// Synchronization interface
try {
  fs.accessSync(`${wx.env.USER_DATA_PATH}/hello.txt`)
} catch(e) {
  console.error(e)
}
```

.accessSync

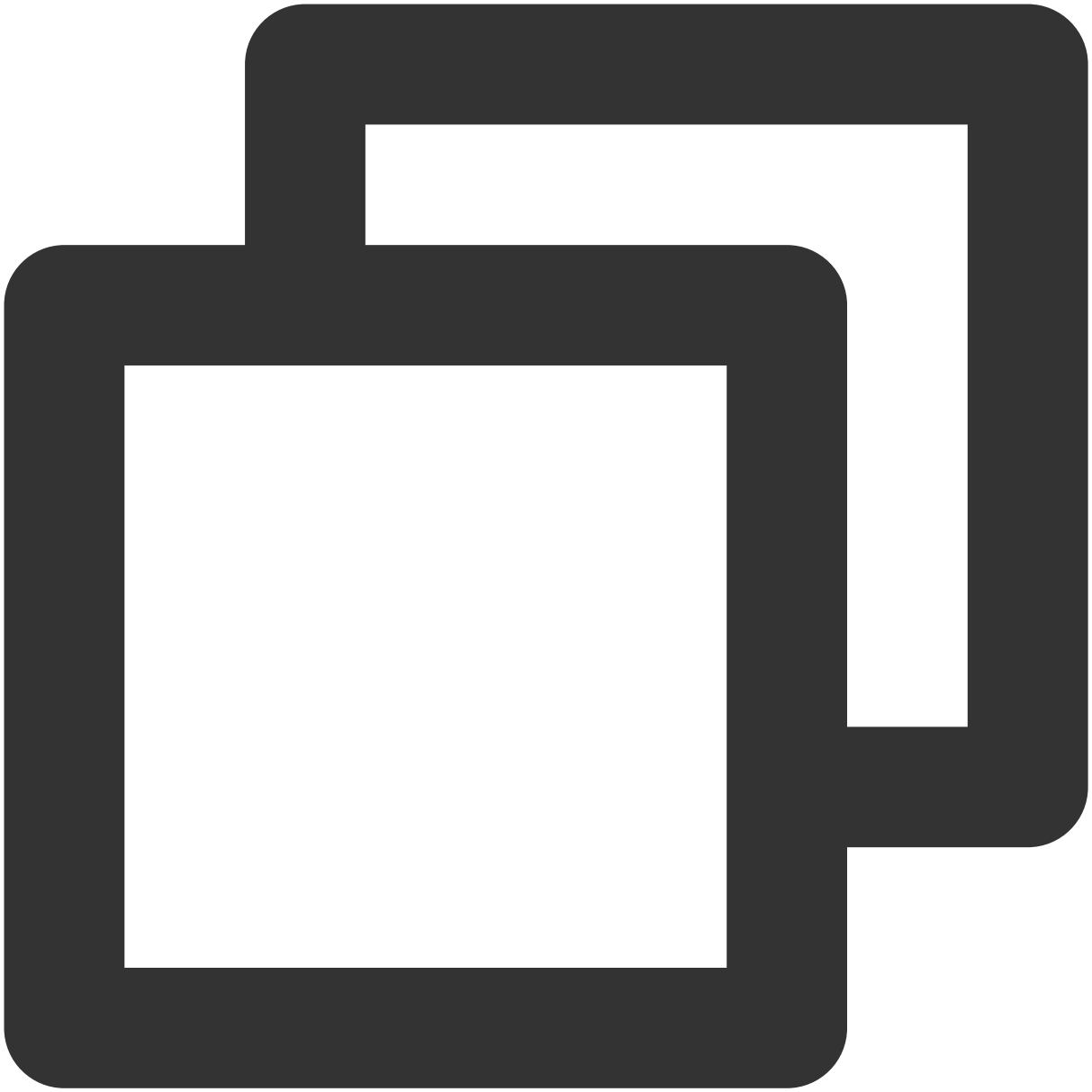
This method is used via FileSystemManager.accessSync(string path).

Feature Description: The synced version of FileSystemManager.access.

Parameter and Description: string path, the file/directory path to be judged for existence.

Sample Code

©2013-2022 Tencent Cloud. All rights reserved.



```
const fs = wx.getFileSystemManager()
// Determine if the file/directory exists
fs.access({
  path: `${wx.env.USER_DATA_PATH}/hello.txt`, success(res) {
  success(res) {
    // The file exists
    console.log(res)
  }, fail(res) { // File exists.
  fail(res) {
    // File does not exist or other error
    console.error(res)
```

©2013-2022 Tencent Cloud. All rights reserved.

```
  }
})

// Synchronization interface
try {
  fs.accessSync(`${wx.env.USER_DATA_PATH}/hello.txt`)
} catch(e) {
  console.error(e)
}
```

.appendFile

This method is used via FileSystemManager.appendFile(Object object).

Feature Description: Appends content at the end of the file.

Parameter and Description: Object

Attribute	Type	Default value	Required	Description
filePath	string	-	Yes	The path of file to which content is to be appended.
data	string/ArrayBuffer	-	Yes	The text or binary data to be appended.
encoding	string	utf-8	No	Specifies the character encoding for the file to be written.
success	function	-	No	Callback Function of Successful Interface Call
fail	function	-	No	Callback Function of Failing Interface Call
complete	function	-	No	Callback function executed upon the completion of the interface invocation (both successful and unsuccessful invocations)

Valid values for object.encoding

Value	Description
ascii	-
base64	-
binary	-

©2013-2022 Tencent Cloud. All rights reserved.

hex	-
ucs2/ucs-2/utf16le/utf-16le	Reads in little-endian order.
utf-8/utf8	-
latin1	-

Parameters for object.fail callback function: Object res

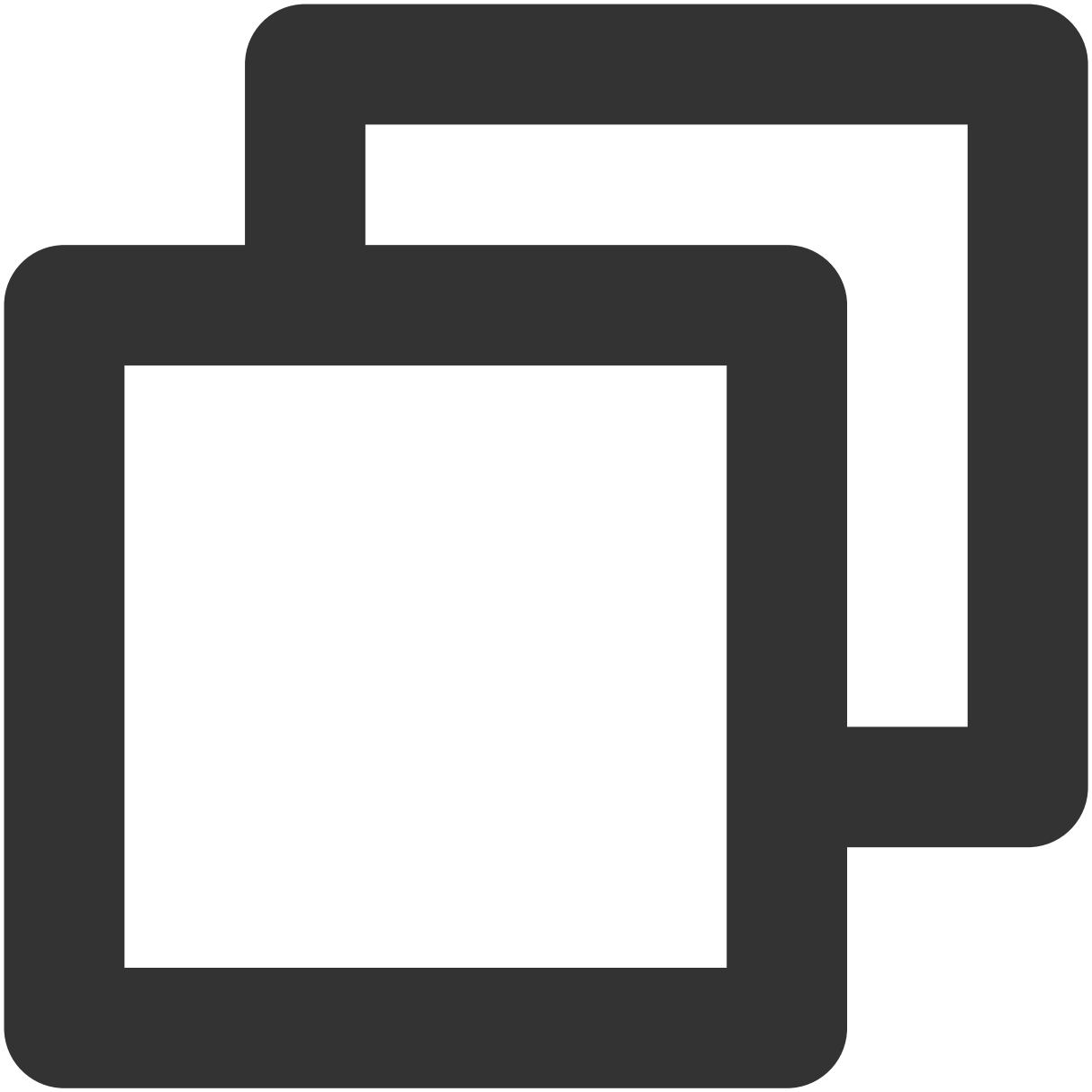
Attribute	Type	Description
errMsg	string	Error Message

Valid values of res.errMsg

Value	Description
fail no such file or directory, open ${filePath}	The specified filePath file does not exist.
fail illegal operation on a directory, open "${filePath}"	The specified filePath is an existing directory.
fail permission denied, open ${dirPath}	The specified filePath does not have write permissions.
fail sdcard not mounted	The specified filePath is an existing directory.

Sample Code

©2013-2022 Tencent Cloud. All rights reserved.



```
const fs = wx.getFileSystemManager()

fs.appendFile({
  filePath: `${wx.env.USER_DATA_PATH}/hello.txt`,
  data: 'some text',
  encoding: 'utf8',
  success(res) {
    console.log(res)
  },
  fail(res) {
    console.error(res)
```

©2013-2022 Tencent Cloud. All rights reserved.

```
  }
})

// Synchronization interface
try {
  fs.appendFileSync(`${wx.env.USER_DATA_PATH}/hello.txt`, 'some text', 'utf8')
} catch(e) {
  console.error(e)
}
```

.appendFileSync

This method is used via FileSystemManager.appendFileSync(string filePath, string|ArrayBuffer data, string encoding).

Feature Description: The synced version of FileSystemManager.appendFile.

Parameter and Description

string filePath: The path of file to which content is to be appended.

string|ArrayBuffer data: The text or binary data to be appended.

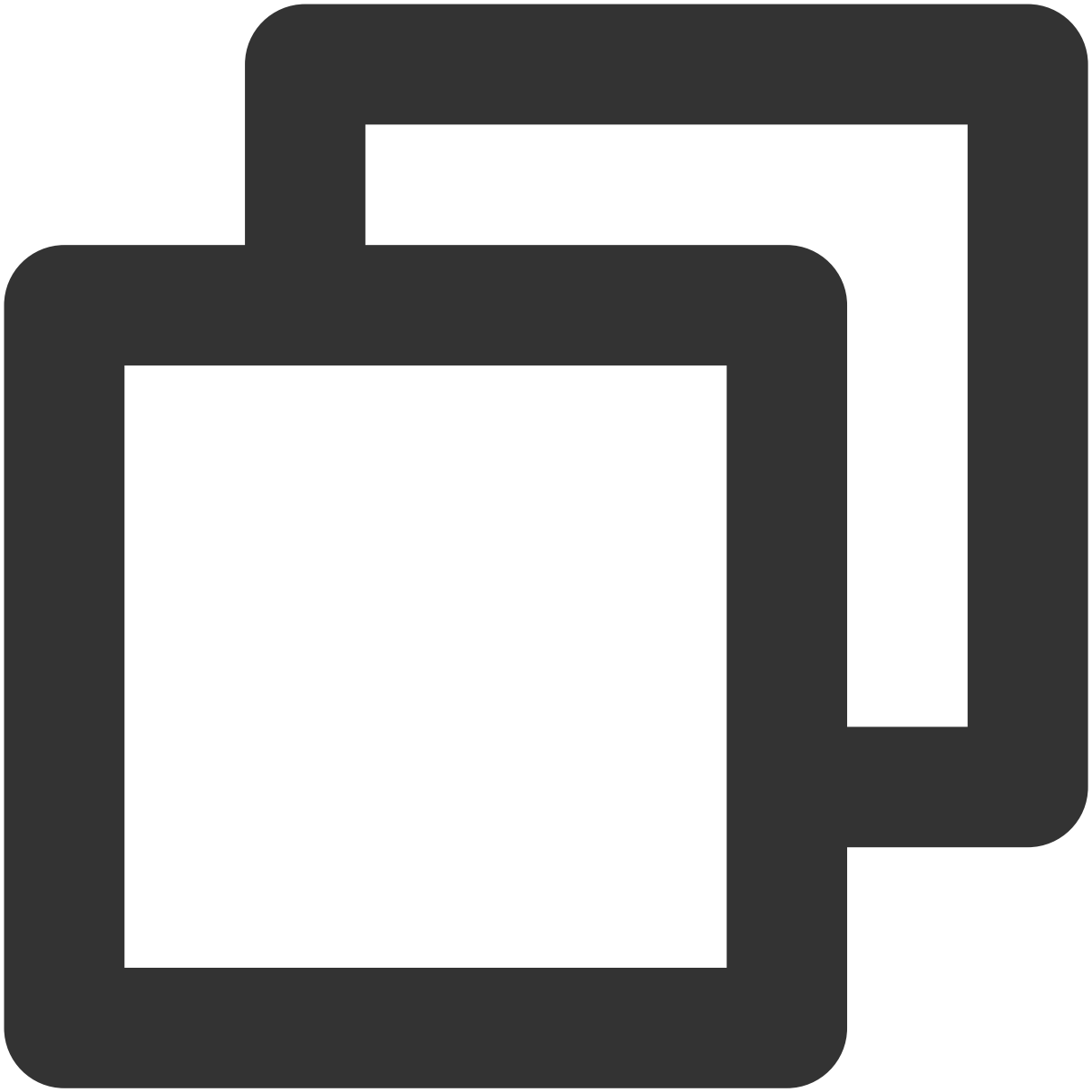
string encoding: Specifies the character encoding for writing to the file.

Valid Values for encoding

Value	Description
ascii	-
base64	-
binary	-
hex	-
ucs2/ucs-2/utf16le/utf-16le	Reads in little-endian order.
utf-8/utf8	-
latin1	-

Sample Code

©2013-2022 Tencent Cloud. All rights reserved.



```
const fs = wx.getFileSystemManager()

fs.appendFile({
  filePath: `${wx.env.USER_DATA_PATH}/hello.txt`,
  data: 'some text',
  encoding: 'utf8',
  success(res) {
    console.log(res)
  },
  fail(res) {
    console.error(res)
```

©2013-2022 Tencent Cloud. All rights reserved.

```
  }
})

// Synchronization interface
try {
  fs.appendFileSync(`${wx.env.USER_DATA_PATH}/hello.txt`, 'some text', 'utf8')
} catch(e) {
  console.error(e)
}
```

.close

This method is used via FileSystemManager.close(Object object)

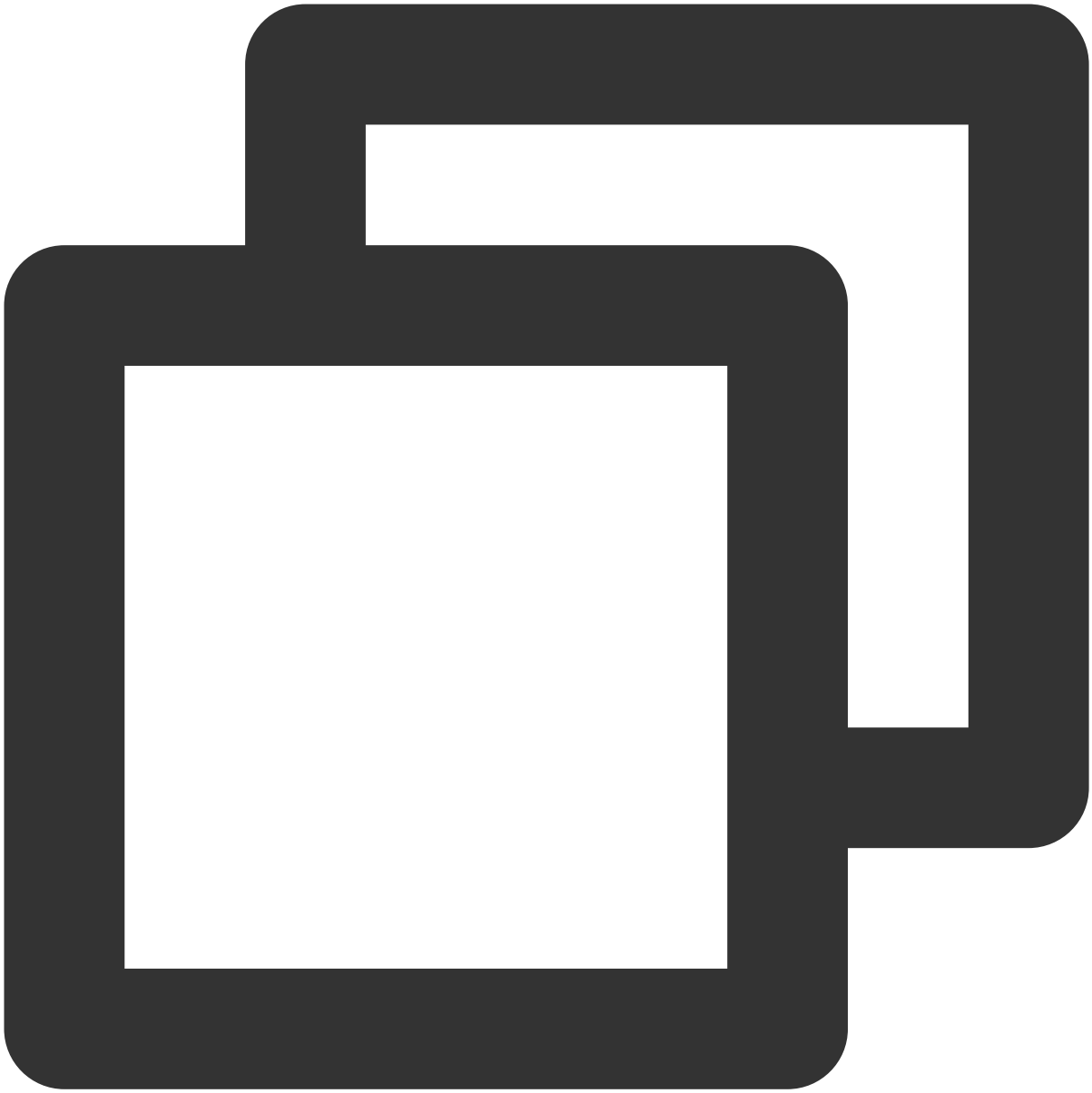
Feature Description: Closes the file.

Parameter and Description: Object

Attribute	Type	Default value	Required	Description
fd	string	-	Yes	The file descriptor that needs to be closed. The fd is obtained through the FileSystemManager.open or FileSystemManager.openSync interface.
success	function	-	No	Callback Function of Successful Interface Call
fail	function	-	No	Callback Function of Failing Interface Call
complete	function	-	No	Callback function executed upon the completion of the interface invocation (both successful and unsuccessful invocations)

Sample Code

©2013-2022 Tencent Cloud. All rights reserved.



```
const fs = wx.getFileSystemManager()
// Open the file
fs.open({
  filePath: ${wx.env.USER_DATA_PATH}/hello.txt,
  flag: 'a+',
      success(res) {
// Close the file
    fs.close({
      fd: res.fd
})
  }
```

©2013-2022 Tencent Cloud. All rights reserved.

```
})
```

.closeSync

This method is used via undefined FileSystemManager.closeSync(Object object).

Feature Description: Synchronously closes the file.

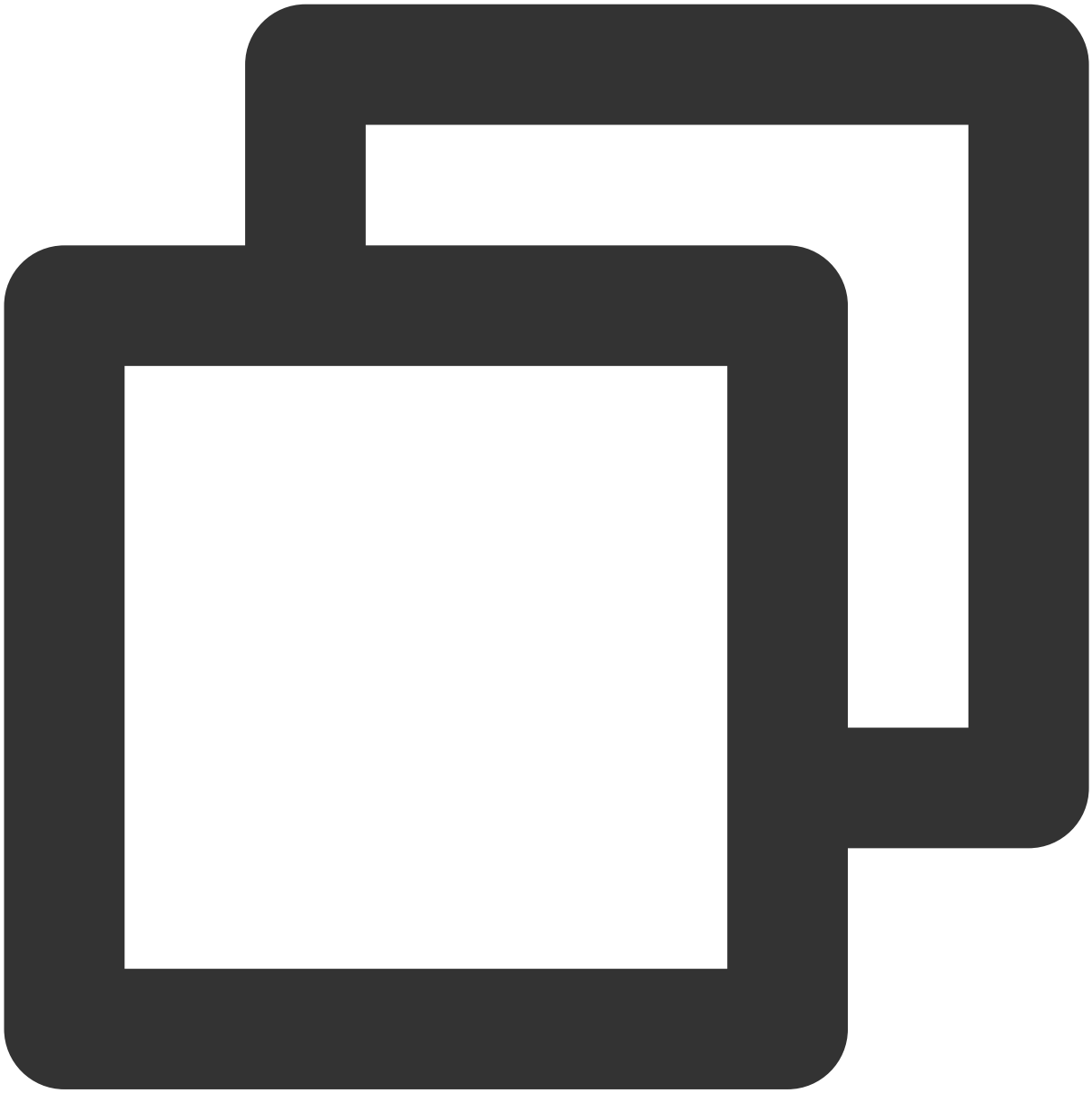
Parameter and Description: Object

Attribute	Type	Default value	Required	Description
fd	string	-	Yes	The file descriptor that needs to be closed. The fd is obtained through the FileSystemManager.open or FileSystemManager.openSync interface.

Return Value: undefined.

Sample Code

©2013-2022 Tencent Cloud. All rights reserved.



```
const fs = wx.getFileSystemManager()
const fd = fs.openSync({
  filePath: ${wx.env.USER_DATA_PATH}/hello.txt,
  flag: 'a+'
})

// Close the file
fs.closeSync({fd: fd})
```

.copyFile

©2013-2022 Tencent Cloud. All rights reserved.

This method is used via FileSystemManager.copyFile(Object object).

Feature Description: Copies the file.

Parameter and Description: Object

Attribute	Type	Default value	Required	Description
srcPath	string	-	Yes	Source file path, which can only be a regular file.
destPath	string	-	Yes	Destination file path, support local path
success	function	-	No	Callback Function of Successful Interface Call
fail	function	-	No	Callback Function of Failing Interface Call
complete	function	-	No	Callback function executed upon the completion of the interface invocation (both successful and unsuccessful invocations)

Parameters for object.fail callback function: Object res

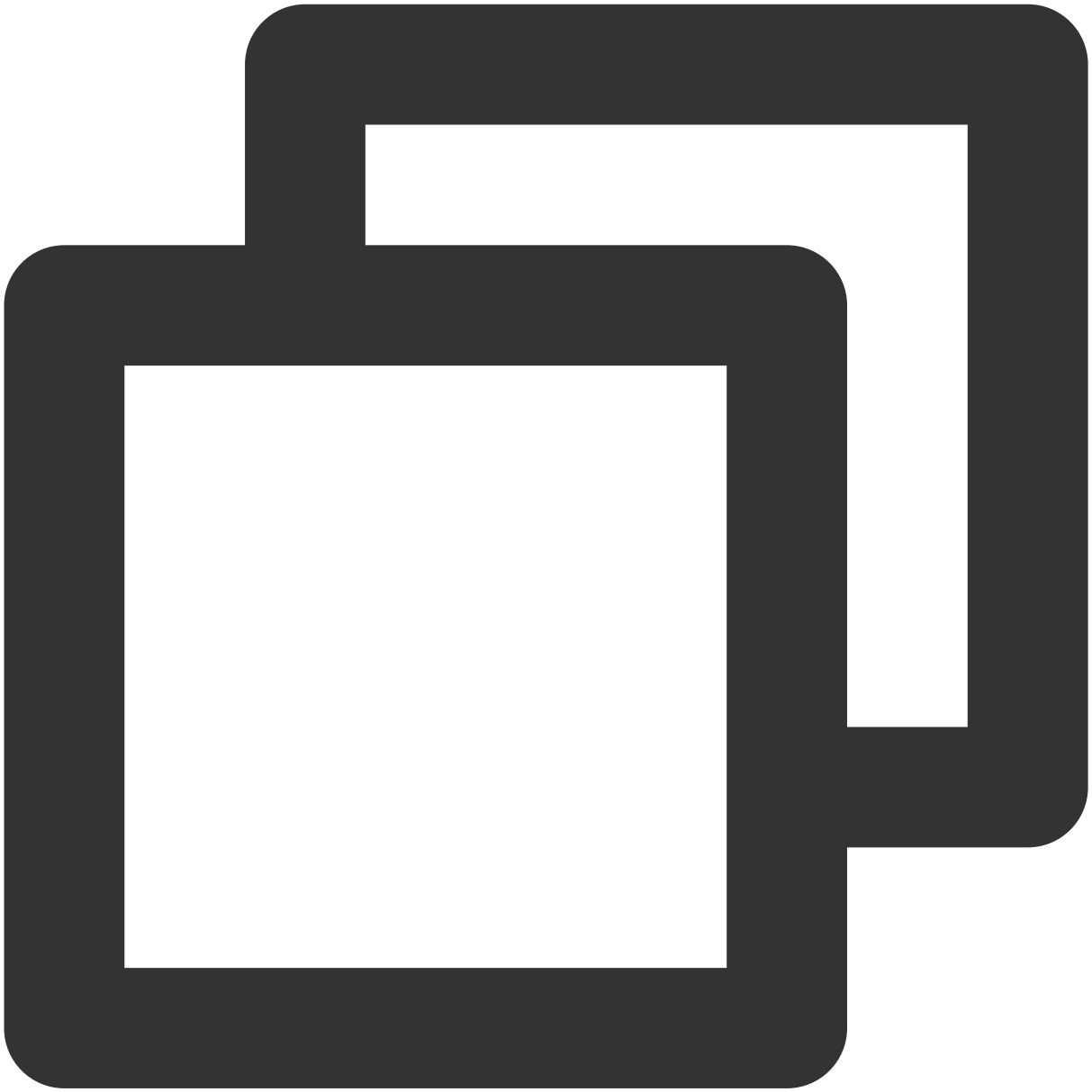
Attribute	Type	Description
errMsg	string	Error Message

Valid values of res.errMsg

Value	Description
fail permission denied, copyFile ${srcPath} -> ${destPath}	The specified destination file path does not have write permissions.
fail no such file or directory, copyFile ${srcPath} -> ${destPath}	The source file does not exist, or the parent directory of the destination file path does not exist.

Sample Code

©2013-2022 Tencent Cloud. All rights reserved.



```
const fs = wx.getFileSystemManager()
fs.copyFile({
  srcPath: `${wx.env_USER_DATA_PATH}/hello.txt`、
  destPath：`${wx.env.USER_DATA_PATH}/hello_copy.txt`。
  success(res) {
    console.log(res)
  },
  fail(res) {
    console.error(res)
  }
})
```

©2013-2022 Tencent Cloud. All rights reserved.

```
// Synchronization interface
try {
  fs.copyFileSync(
    `${wx.env.USER_DATA_PATH}/hello.txt`、`${wx.env.USER_DATA_PATH}/hello_copy.txt`
    ${wx.env.USER_DATA_PATH}/hello_copy.txt`.
  )
} catch(e) {
  console.error(e)
}
```

.copyFileSync

This method is used via FileSystemManager.copyFileSync(string srcPath, string destPath).

Feature Description: The synced version of FileSystemManager.copyFile.

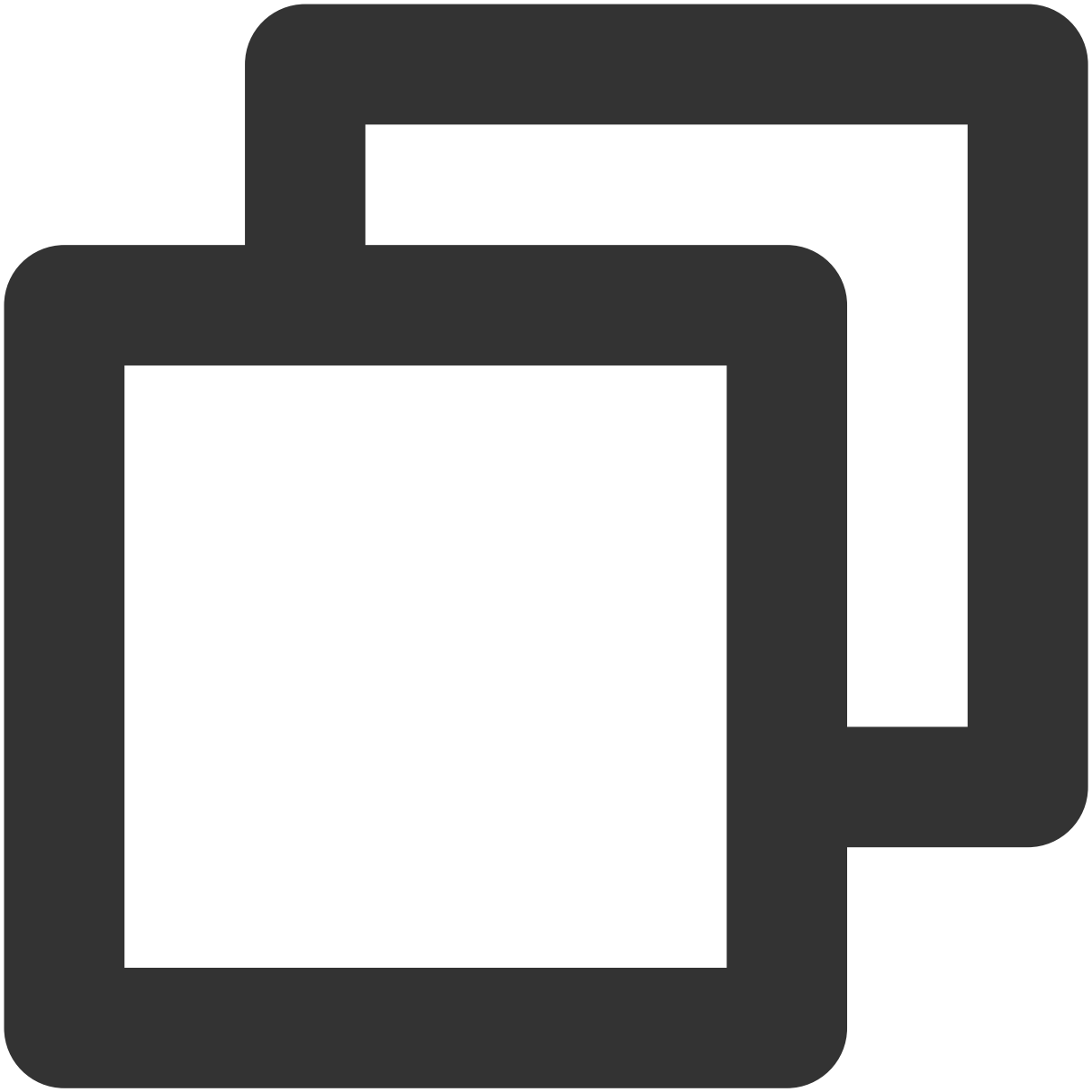
Parameter and description

string srcPath: Source file path, which can only be a regular file.

string destPath: Destination file path.

Sample Code

©2013-2022 Tencent Cloud. All rights reserved.



```
const fs = wx.getFileSystemManager()
fs.copyFile({
  srcPath: `${wx.env_USER_DATA_PATH}/hello.txt`、
  destPath：`${wx.env.USER_DATA_PATH}/hello_copy.txt`。
  success(res) {
    console.log(res)
  },
  fail(res) {
    console.error(res)
  }
})
```

©2013-2022 Tencent Cloud. All rights reserved.

```
// Synchronization interface
try {
  fs.copyFileSync(
    `${wx.env.USER_DATA_PATH}/hello.txt`、`${wx.env.USER_DATA_PATH}/hello_copy.txt`
    ${wx.env.USER_DATA_PATH}/hello_copy.txt`.
  )
} catch(e) {
  console.error(e)
}
```

.getFileInfo

This method is used via FileSystemManager.getFileInfo(Object object).

Feature Description: Gets information on local temporary files or local cache files under this mini program.

Parameter and Description: Object

Attribute	Type	Default value	Required	Description
filePath	string	-	Yes	The path of the file to be read.
digestAlgorithm	string	md5	No	Algorithms for calculating document summaries
success	function	-	No	Callback Function of Successful Interface Call
fail	function	-	No	Callback Function of Failing Interface Call
complete	function	-	No	Callback function executed upon the completion of the interface invocation (both successful and unsuccessful invocations)

Legal value of object.digestAlgorithm

Value	Description
md5	md5 algorithm
sha1	sha1 algorithm

Parameters for object.success callback function: Object res.

Attribute	Type	Description

©2013-2022 Tencent Cloud. All rights reserved.

size	number	File size, measured in bytes.
digest	string	File digest calculated according to the passed digestAlgorithm

Parameters for object.fail callback function: Object res

Attribute	Type	Description
errMsg	string	Error Message

Valid values of res.errMsg

Value	Description
fail file not exist	The file is not found in the specified filePath.
no such file or directory ${path}	File/directory does not exist, or the upper directory of the target file path does not exist
Input/output error	Input or output streams not available
permission denied	Permission error, file is read-only or write-only
Path permission denied	The incoming path has no permissions
not a directory	dirPath specifies that the path is not a directory, which is common when the parent path of the specified write path is a file.
Invalid argument	Invalid parameter to check if length or offset is out of bounds
excced max concurrent fd limit	The number of fd's has reached its limit

.fstat

This method is used via FileSystemManager.fstat(Object object).

Feature Description: Gets the status information of the file.

Parameter and Description: Object

Attribute	Type	Default value	Required	Description
fd	string	-	Yes	The file descriptor. The fd is obtained through the FileSystemManager.open or FileSystemManager.openSync

©2013-2022 Tencent Cloud. All rights reserved.

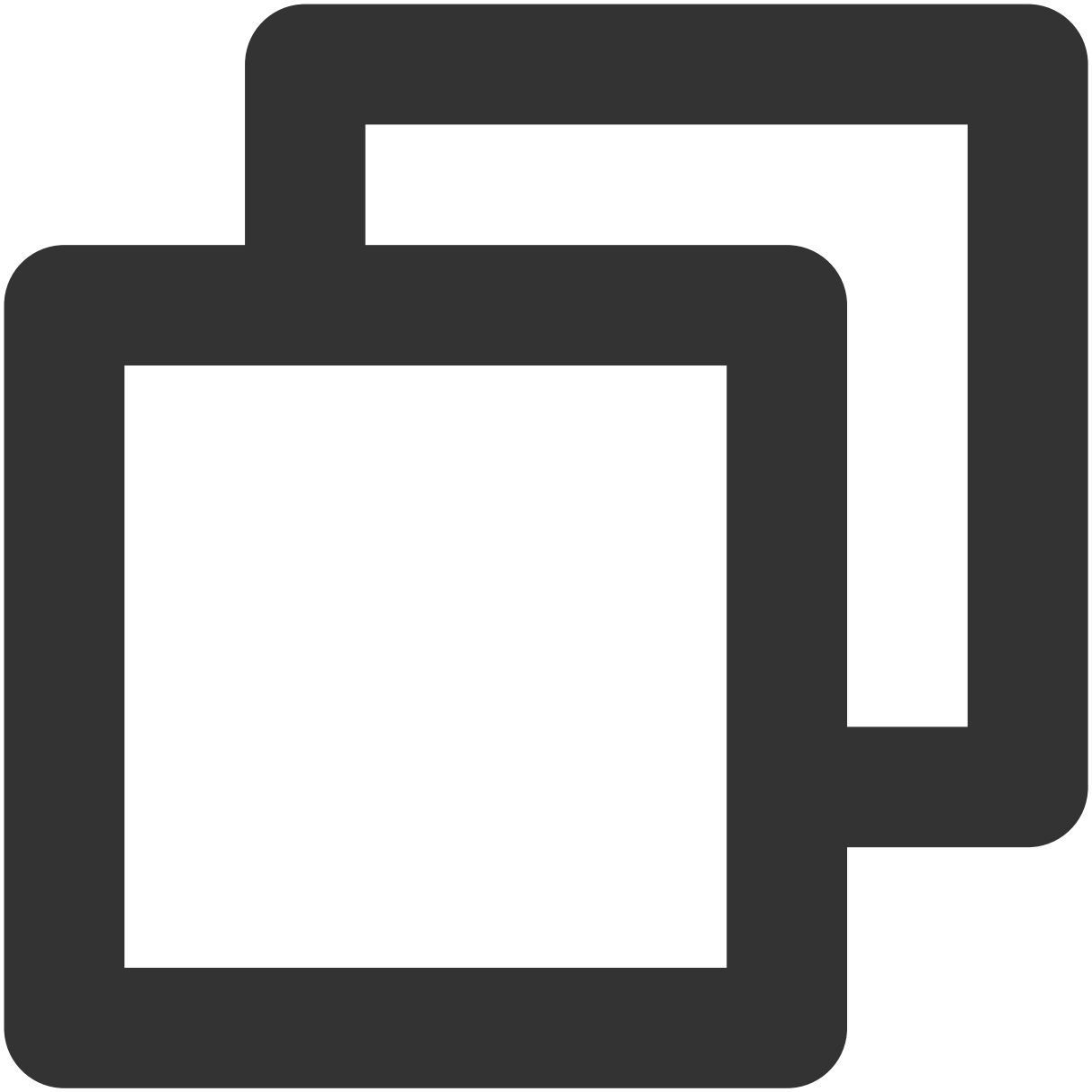
				interface.
success	function	-	No	Callback Function of Successful Interface Call
fail	function	-	No	Callback Function of Failing Interface Call
complete	function	-	No	Callback function executed upon the completion of the interface invocation (both successful and unsuccessful invocations)

Parameters for object.success callback function: Object res.

Attribute	Type	Description
stats	Stats	Get the Stats object of the file, which contains the status information of the file.

Sample Code

©2013-2022 Tencent Cloud. All rights reserved.



```
const fs = wx.getFileSystemManager()
// Open the file
fs.open({
  filePath: ${wx.env.USER_DATA_PATH}/hello.txt,
  flag: 'a+',
      success(res) {
// Get the status information of the file.
    fs.fstat({
      fd: res.fd,
      success(res) {
        console.log(res.stats)
```

©2013-2022 Tencent Cloud. All rights reserved.

```
  }
})
  }
})
```

.fstatSync

This method is used via Stats FileSystemManager.fstatSync(Object object).

Feature Description: Synchronously gets the status information of the file.

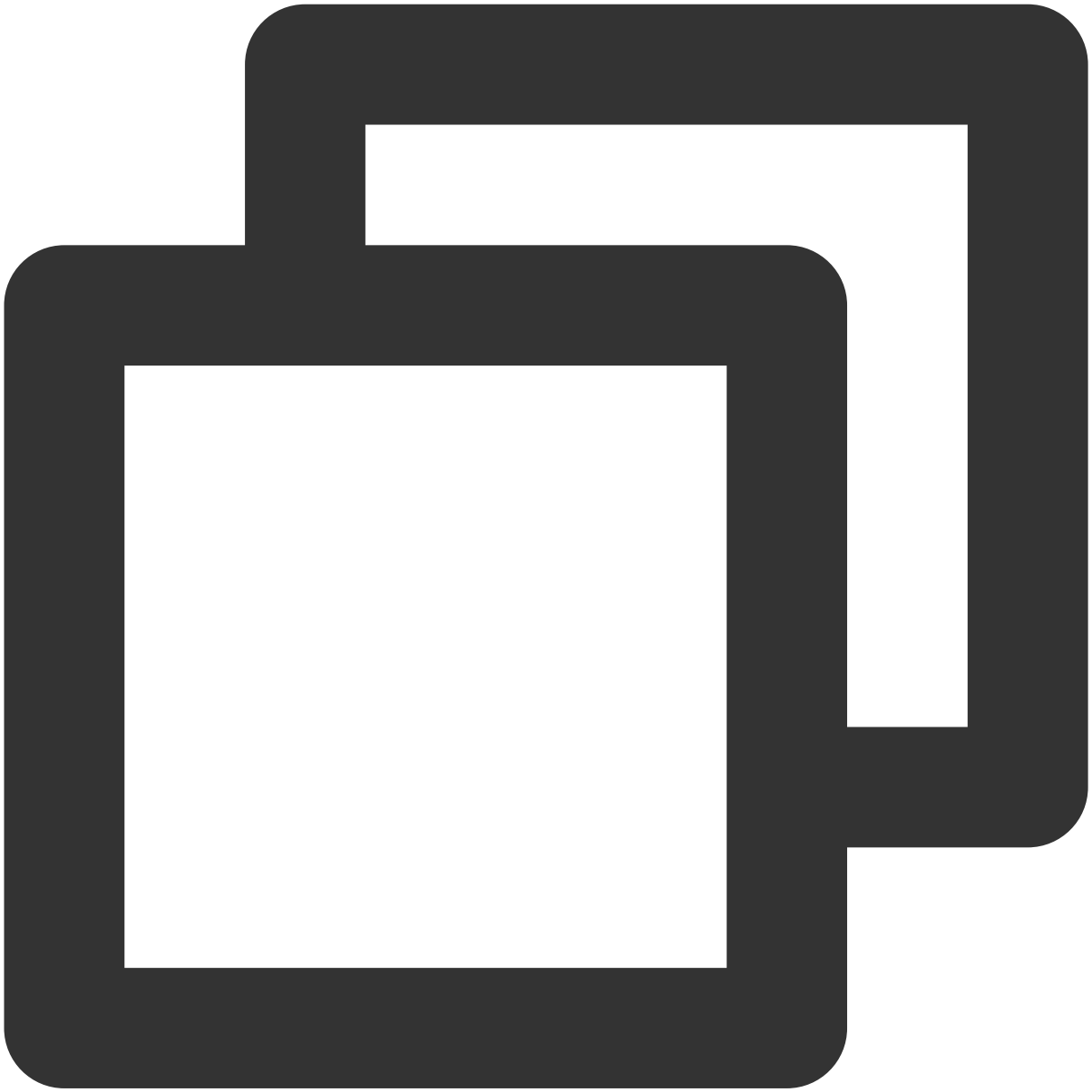
Parameter and Description: Object

Attribute	Type	Default value	Required	Description
fd	string	-	Yes	The file descriptor. The fd is obtained through the FileSystemManager.open or FileSystemManager.openSync interface.

Return Value: Stats, a Stats object, containing the status information of the file.

Sample Code

©2013-2022 Tencent Cloud. All rights reserved.



```
const fs = wx.getFileSystemManager()
const fd = fs.openSync({
  filePath: ${wx.env.USER_DATA_PATH}/hello.txt,
  flag: 'a+'
})
const stats = fs.fstatSync({fd: fd})
console.log(stats)
```

.ftruncate

©2013-2022 Tencent Cloud. All rights reserved.

This method is used via FileSystemManager.ftruncate(Object object).

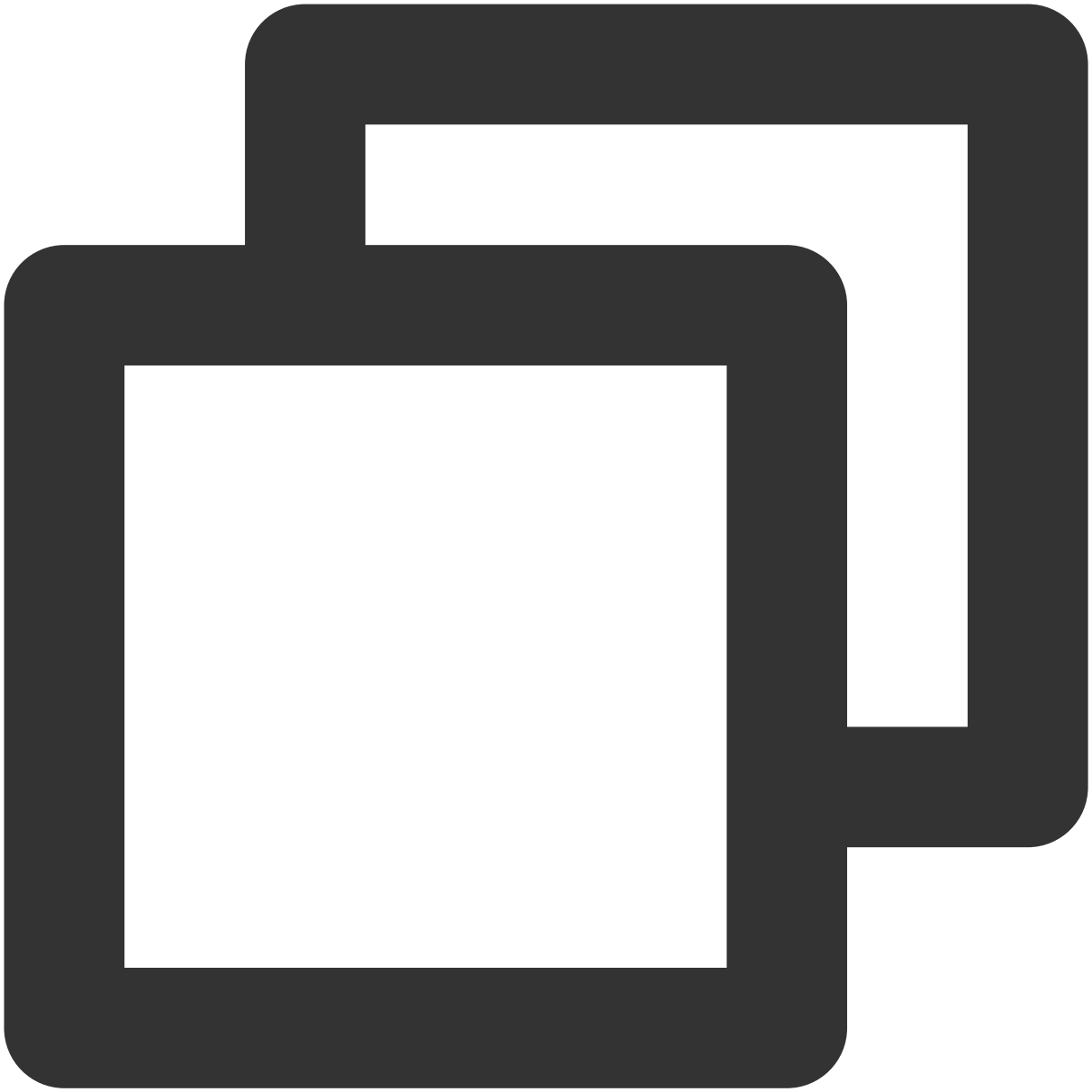
Feature Description: Executes a truncation operation on the file content.

Parameter and Description: Object

Attribute	Type	Default value	Required	Description
fd	string	-	Yes	The file descriptor. The fd is obtained through the FileSystemManager.open or FileSystemManager.openSync interface.
length	number	-	Yes	By default, the truncation position is 0. If the specified length value is less than the file length (measured in bytes), only the preceding number of bytes equal to the specified length will be retained in the file, and the remaining content will be deleted. If the specified length exceeds the file length, the file will be expanded, with the extended section being filled with null bytes ('\\0').
success	function	-	No	Callback Function of Successful Interface Call
fail	function	-	No	Callback Function of Failing Interface Call
complete	function	-	No	Callback function executed upon the completion of the interface invocation (both successful and unsuccessful invocations)

Sample Code

©2013-2022 Tencent Cloud. All rights reserved.



```
const fs = wx.getFileSystemManager()
// Open the file
fs.open({
  filePath: ${wx.env.USER_DATA_PATH}/hello.txt,
  flag: 'a+',
      success(res) {
//Truncate the file content.
    fs.ftruncate({
      fd: res.fd,
  length: 10, // Truncate the file starting from the 10th byte
      success(res) {
```

©2013-2022 Tencent Cloud. All rights reserved.

```
    console.log(res)
  }
})
  }
})
```

.ftruncateSync

This method is used via undefined FileSystemManager.ftruncateSync(Object object).

Feature Description: Executes a truncation operation on the file content.

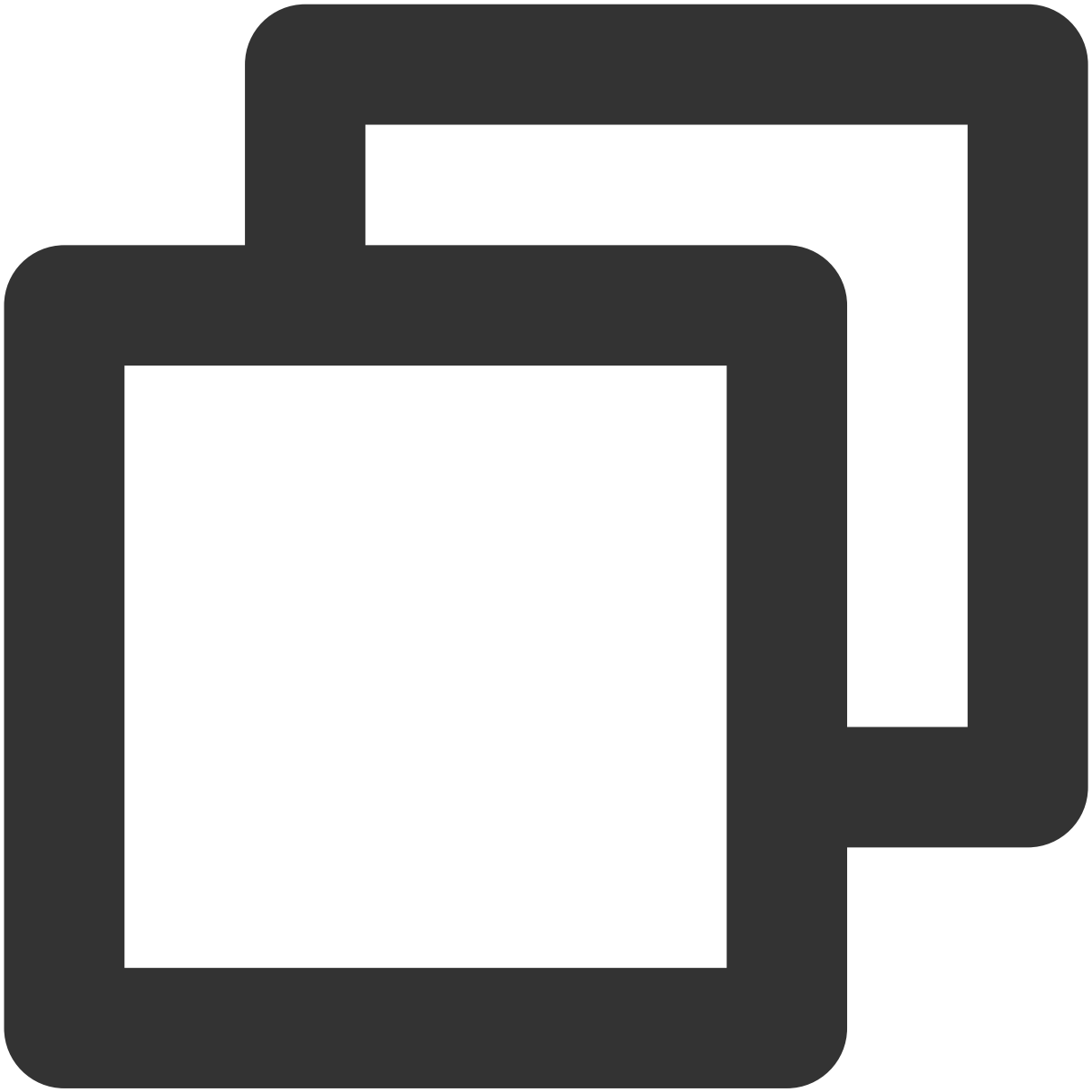
Parameter and Description: Object

Attribute	Type	Default value	Required	Description
fd	string	-	Yes	The file descriptor. The fd is obtained through the FileSystemManager.open or FileSystemManager.openSync interface.
length	number	-	Yes	By default, the truncation position is 0. If the specified length value is less than the file length (measured in bytes), only the preceding number of bytes equal to the specified length will be retained in the file, and the remaining content will be deleted. If the specified length exceeds the file length, the file will be expanded, with the extended section being filled with null bytes ('\\0').

Return Value: undefined.

Sample Code

©2013-2022 Tencent Cloud. All rights reserved.



```
const fs = wx.getFileSystemManager()
const fd = fs.openSync({
  filePath: ${wx.env.USER_DATA_PATH}/hello.txt,
  flag: 'a+'
})
fs.ftruncateSync({
  fd: fd,
length: 10// Truncate the file starting from the 10th byte
})
```

©2013-2022 Tencent Cloud. All rights reserved.

.getSavedFileList

This method is used via FileSystemManager.getSavedFileList(Object object).

Feature Description: Gets the list of locally cached files saved under this mini program.

Parameter and Description: Object

Attribute	Type	Default value	Required	Description
success	function	-	No	Callback Function of Successful Interface Call
fail	function	-	No	Callback Function of Failing Interface Call
complete	function	-	No	Callback function executed upon the completion of the interface invocation (both successful and unsuccessful invocations)

Parameters for object.success callback function: Object res.

Attribute	Type	Description
fileList	Array.<Object>	Array of files

Structure of res.fileList

Attribute	Type	Description
filePath	string	Local path
size	number	Local file size, measured in bytes.
createTime	number	Timestamp for file saving, denoting the number of seconds from 08:00:00 on Jan. 01, 1970 to the current time.

.mkdir

This method is used via FileSystemManager.mkdir(Object object).

Feature Description: Creates directory.

Parameter and Description: Object

©2013-2022 Tencent Cloud. All rights reserved.

Attribute	Type	Default value	Required	Description
dirPath	string	-	Yes	Path of the created directory.
recursive	boolean	false	No	Whether to create the directory after recursively creating its parent directory. If the corresponding parent directory already exists, this parent directory will not be created. For instance, if dirPath is a/b/c/d and recursive is true, it will create directory a, then create directory b under a, and so forth until directory d under a/b/c is created.
success	function	-	No	Callback Function of Successful Interface Call
fail	function	-	No	Callback Function of Failing Interface Call
complete	function	-	No	Callback function executed upon the completion of the interface invocation (both successful and unsuccessful invocations)

Parameters for object.fail callback function: Object res

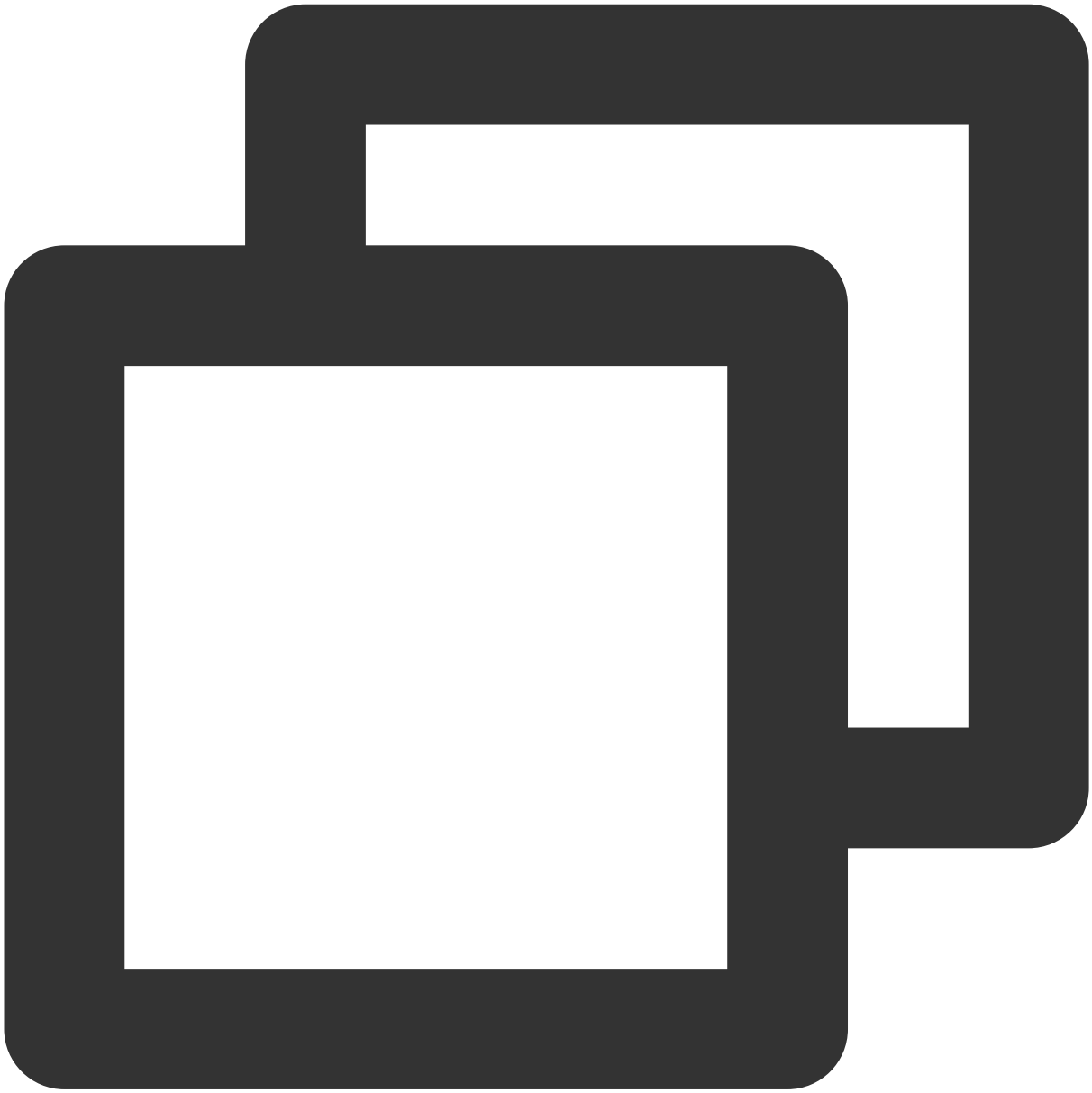
Attribute	Type	Description
errMsg	string	Error Message

Valid values of res.errMsg

Value	Description
fail no such file or directory ${dirPath}	Parent directory does not exist.
fail permission denied, open ${dirPath}	The specified filePath does not have write permissions.
fail file already exists ${dirPath}	There exists a file or directory with the same name.

Sample Code

©2013-2022 Tencent Cloud. All rights reserved.



```
const fs = wx.getFileSystemManager()
fs.mkdir({
  dirPath: `${wx.env.USER_DATA_PATH}/example`,
  recursive: false,
  success(res) {
    console.log(res)
  },
  fail(res) {
    console.error(res)
  }
})
```

©2013-2022 Tencent Cloud. All rights reserved.

```
// Synchronization interface
try {
  fs.mkdirSync(`${wx.env.USER_DATA_PATH}/example`, false)
} catch(e) {
  console.error(e)
}
```

.mkdirSync

This method is used via FileSystemManager.mkdirSync(string dirPath, boolean recursive).

Feature Description: The synced version of FileSystemManager.mkdir.

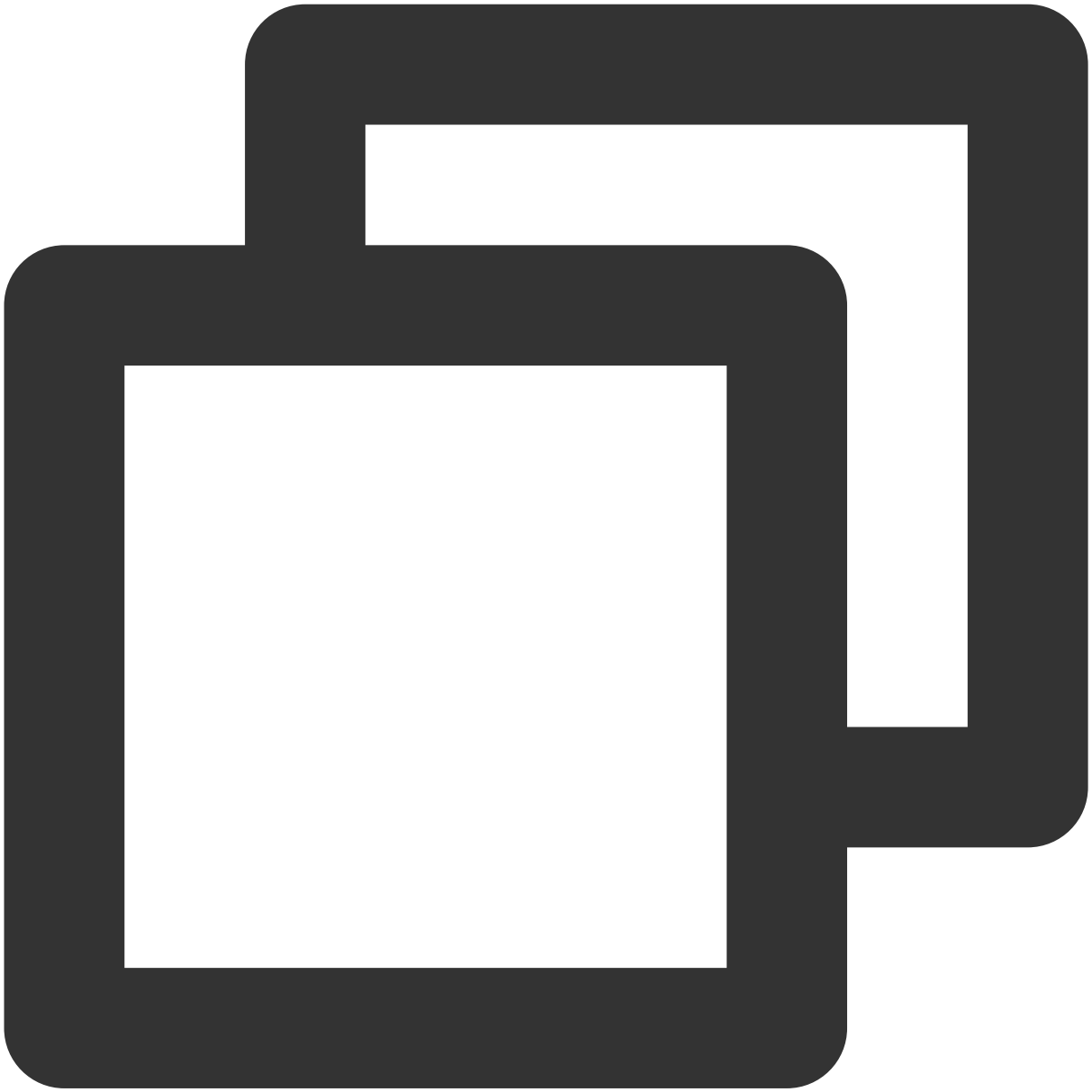
Parameter and Description:

string dirPath: The path of the directory to be created.

boolean recursive: Whether to create the directory after recursively creating its parent directory. If the corresponding parent directory already exists, this parent directory will not be created. For instance, if dirPath is a/b/c/d and recursive is true, it will create directory a, then create directory b under a, and so forth until directory d under a/b/c is created.

Sample Code

©2013-2022 Tencent Cloud. All rights reserved.



```
const fs = wx.getFileSystemManager()
fs.mkdir({
  dirPath: `${wx.env.USER_DATA_PATH}/example`,
  recursive: false,
  success(res) {
    console.log(res)
  },
  fail(res) {
    console.error(res)
  }
})
```

©2013-2022 Tencent Cloud. All rights reserved.

```
// Synchronization interface
try {
  fs.mkdirSync(`${wx.env.USER_DATA_PATH}/example`, false)
} catch(e) {
  console.error(e)
}
```

.open

This method is used via FileSystemManager.open(Object object).

Feature Description: Opens a file, returning a file descriptor.

Parameter and Description: Object

Attribute	Type	Default value	Required	Description
filePath	string	-	Yes	File path (local path)
flag	string	r	No	File system flag, default value: r
success	function	-	No	Callback Function of Successful Interface Call
fail	function	-	No	Callback Function of Failing Interface Call
complete	function	-	No	Callback function executed upon the completion of the interface invocation (both successful and unsuccessful invocations)

Valid values of flag.

Valid Values	Description
a	Opens a file for appending. If the file does not exist, it creates the file.
ax	Similar to 'a', but will fail if the path exists.
a+	Opens a file for reading and appending. If the file does not exist, it creates the file.
ax+	Similar + to 'a', but will fail if the path exists.
as	Opens a file for appending (in the syncing mode). If the file does not exist, it creates the file.

©2013-2022 Tencent Cloud. All rights reserved.

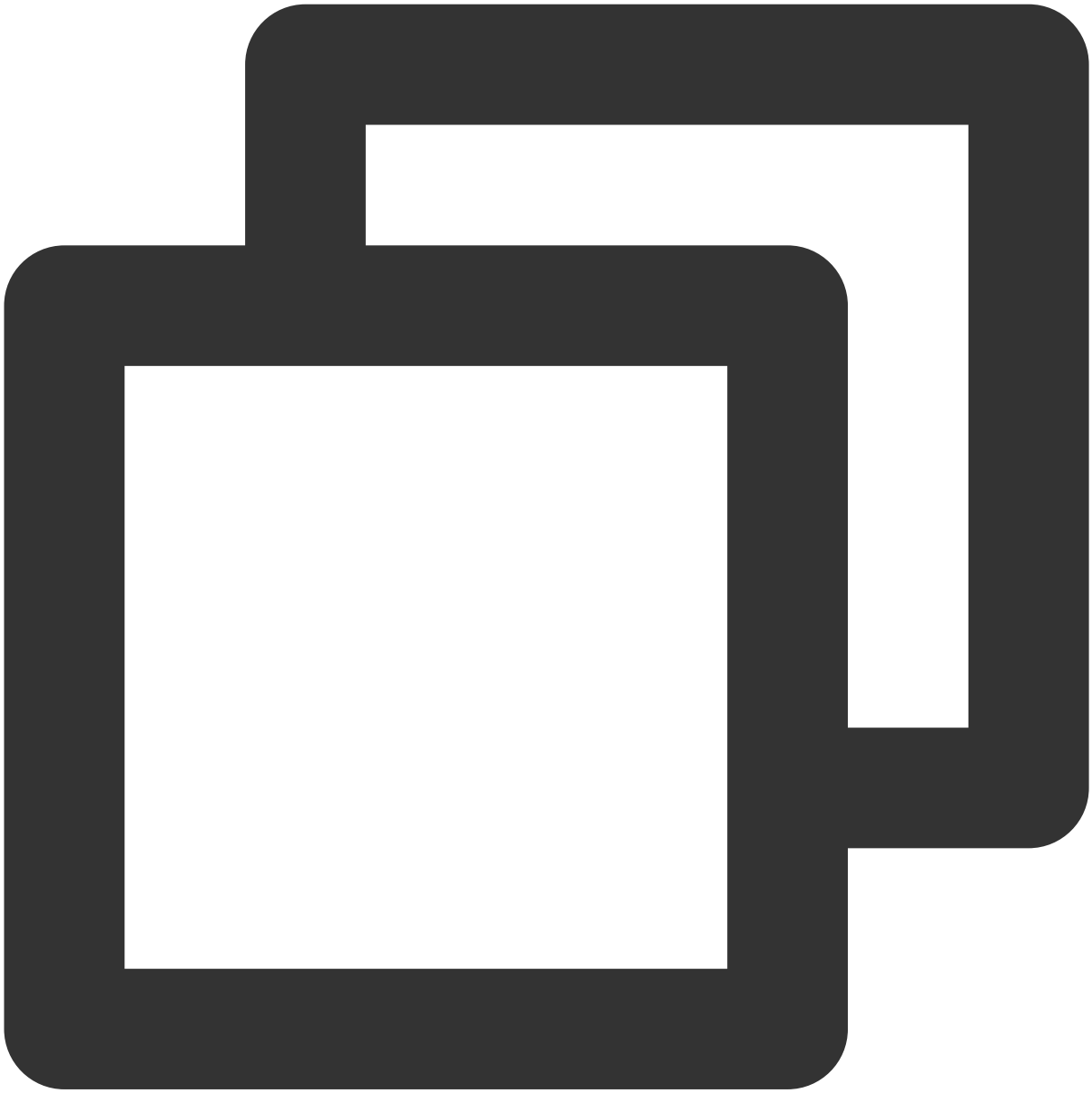
as+	Opens a file for both reading and appending (in the syncing mode). If the file does not exist, it creates the file.
r	Opens a file for reading. If the file does not exist, an exception occurs.
r+	Opens a file for reading and writing. If the file does not exist, an exception occurs.
w	Opens a file for writing. If the file does not exist, it creates the file; if the file exists, it truncates the file.
wx	Similar to 'w', but will fail if the path exists.
w+	Opens a file for reading and writing. If the file does not exist, it creates the file; if the file exists, it truncates the file.
wx+	Similar to 'w+', but will fail if the path exists.

Parameters for object.success callback function: Object res.

Attribute	Type	Description
fd	string	File descriptor

Sample Code

©2013-2022 Tencent Cloud. All rights reserved.



```
const fs = wx.getFileSystemManager()
fs.open({
  filePath: ${wx.env.USER_DATA_PATH}/hello.txt,
  flag: 'a+',
      success(res) {
    console.log(res.fd)
  }
})
```

.openSync

©2013-2022 Tencent Cloud. All rights reserved.

This method is used via string FileSystemManager.openSync(Object object).

Feature Description: Synchronously opens a file, returning a file descriptor.

Parameter and Description: Object

Attribute	Type	Default value	Required	Description
filePath	string	-	Yes	File path (local path)
flag	string	r	No	File system flag, default value: r

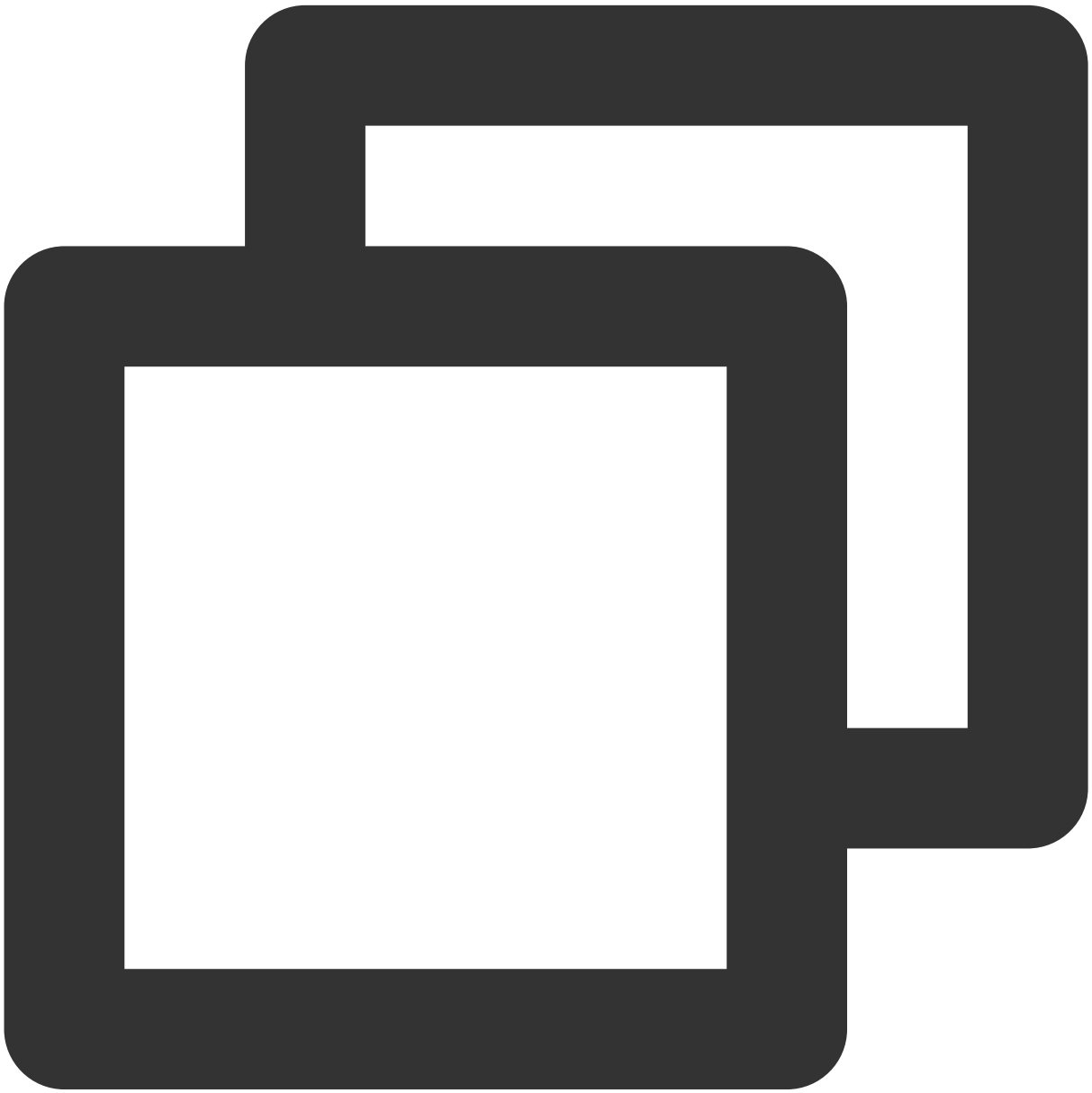
Valid values of flag.

Valid Values	Description
a	Opens a file for appending. If the file does not exist, it creates the file.
ax	Similar to 'a', but will fail if the path exists.
a+	Opens a file for reading and appending. If the file does not exist, it creates the file.
ax+	Similar + to 'a', but will fail if the path exists.
as	Opens a file for appending (in the syncing mode). If the file does not exist, it creates the file.
as+	Opens a file for both reading and appending (in the syncing mode). If the file does not exist, it creates the file.
r	Opens a file for reading. If the file does not exist, an exception occurs.
r+	Opens a file for reading and writing. If the file does not exist, an exception occurs.
w	Opens a file for writing. If the file does not exist, it creates the file; if the file exists, it truncates the file.
wx	Similar to 'w', but will fail if the path exists.
w+	Opens a file for reading and writing. If the file does not exist, it creates the file; if the file exists, it truncates the file.
wx+	Similar to 'w+', but will fail if the path exists.

Return Value: String, the file descriptor.

©2013-2022 Tencent Cloud. All rights reserved.

Sample Code



```
const fs = wx.getFileSystemManager()
const fd = fs.openSync({
  filePath: `${wx.env.USER_DATA_PATH}/hello.txt`,
  flag: 'a+'
})
console.log(fd)
```

.read

©2013-2022 Tencent Cloud. All rights reserved.

This method is used via FileSystemManager.read(Object object).

Feature Description: Reads a file.

Parameter and Description: Object

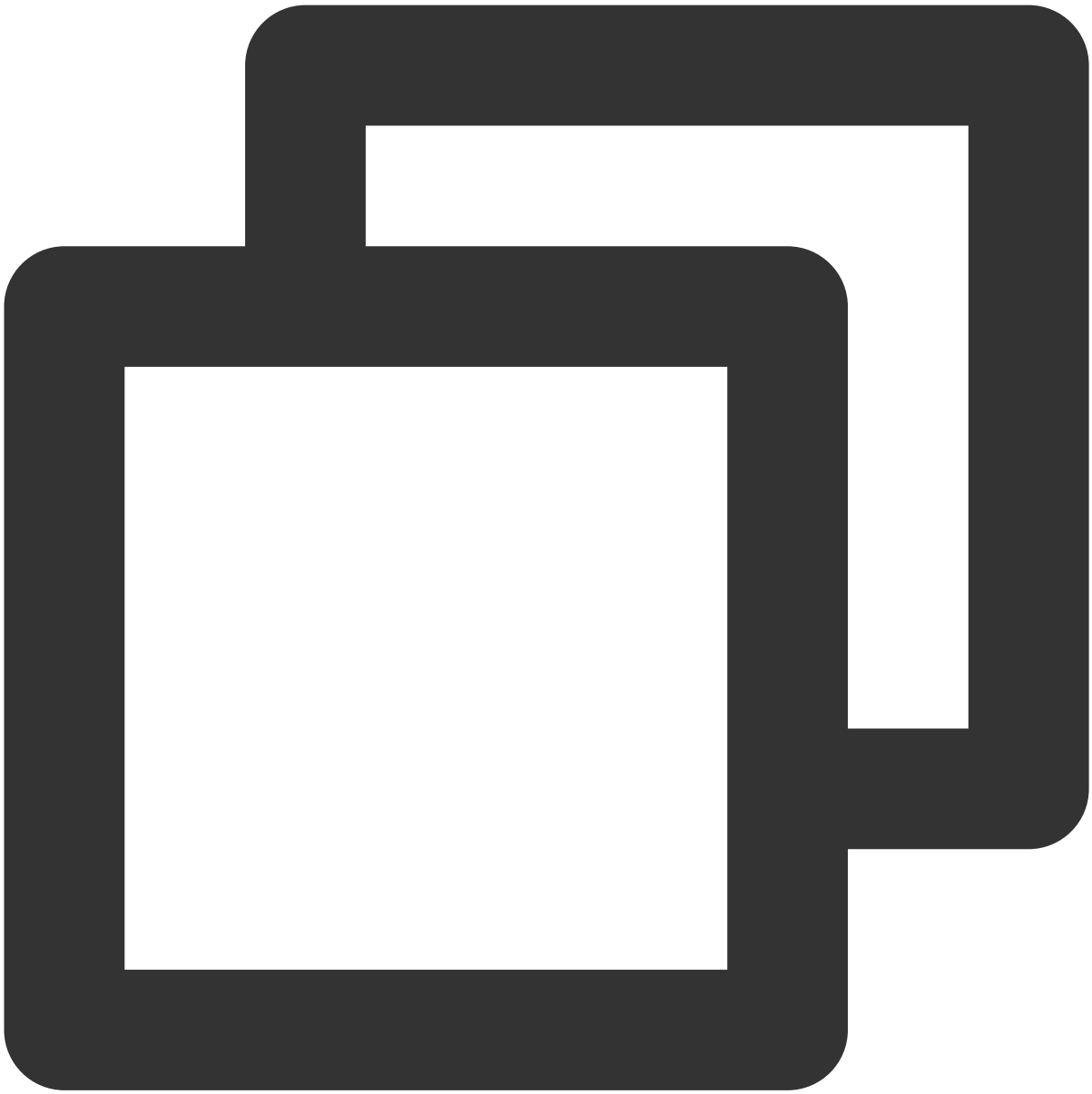
Attribute	Type	Default value	Required	Description
fd	string	-	Yes	The file descriptor. The fd is obtained through the FileSystemManager.open or FileSystemManager.openSync interface.
arrayBuffer	ArrayBuffer	-	Yes	The buffer for data writing must be an instance of ArrayBuffer.
offset	number	0	No	The write offset in the buffer, with the default value being 0.
length	number	0	No	The number of bytes to read from the file, with the default value being 0.
position	number	-	No	The starting position of the file to be read, if not passed or if null is passed, then the file will be read from the current position of the file pointer. If position is a positive integer, the position of the file pointer will remain unchanged and the file will be read from position.
success	function	-	No	Callback Function of Successful Interface Call
fail	function	-	No	Callback Function of Failing Interface Call
complete	function	-	No	Callback function executed upon the completion of the interface invocation (both successful and unsuccessful invocations)

Parameters for object.success callback function: Object res.

Attribute	Type	Description
bytesRead	number	The actual number of bytes read.
arrayBuffer	ArrayBuffer	The object of the buffer being written to, that is, the arrayBuffer parameter of the interface.

©2013-2022 Tencent Cloud. All rights reserved.

Sample Code



```
const fs = wx.getFileSystemManager()
const ab = new ArrayBuffer(1024)
// Open the file
fs.open({
  filePath: ${wx.env.USER_DATA_PATH}/hello.txt,
  flag: 'a+',
      success(res) {
//Read the file into the ArrayBuffer
    fs.read({
```

©2013-2022 Tencent Cloud. All rights reserved.

```
      fd: res.fd,
  arrayBuffer: ab,
      length: 10,
      success(res) {
    console.log(res)
  }
})
  }
})
```

.readSync

This method is used via ReadResult FileSystemManager.readSync(Object object).

Feature Description: Reads a file.

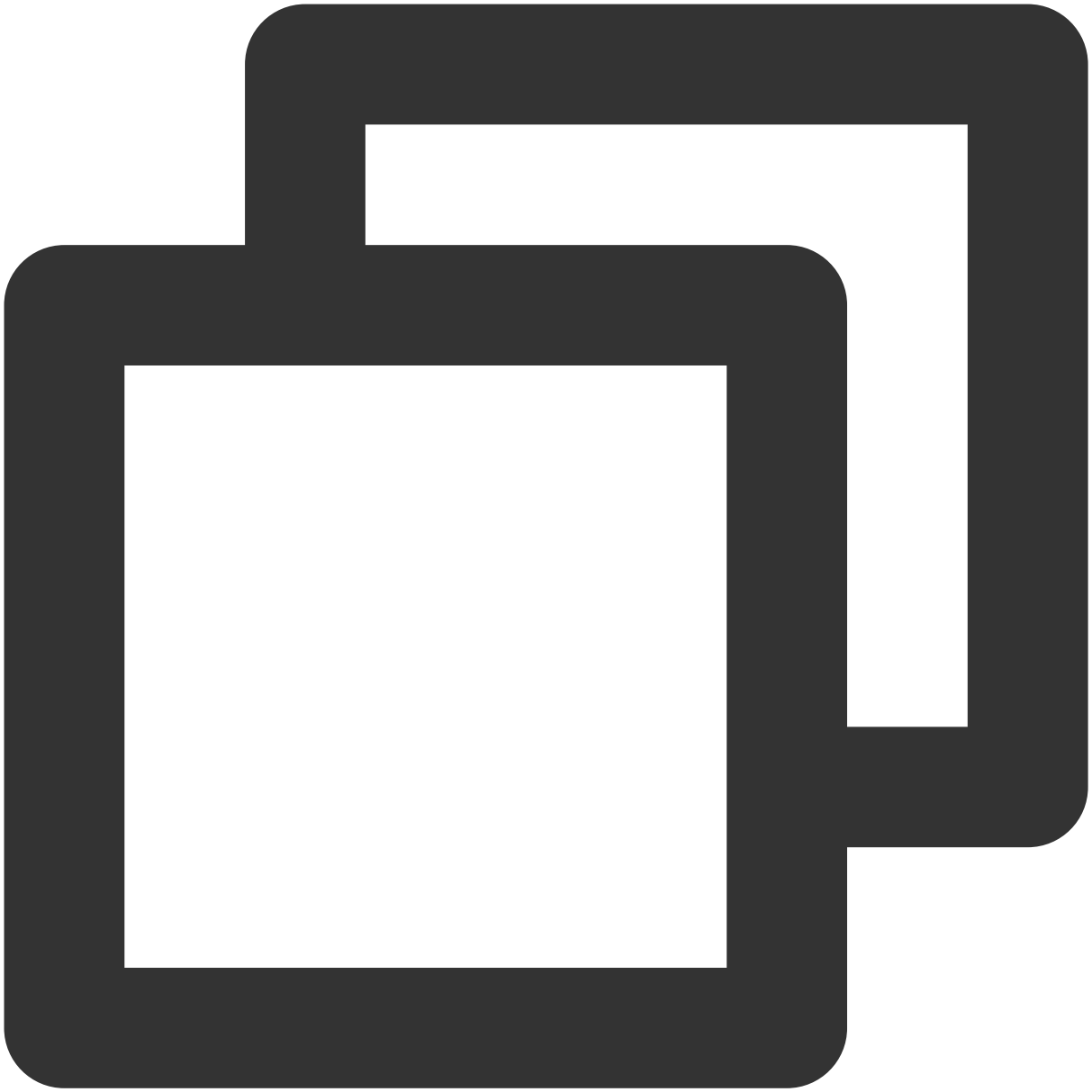
Parameter and Description: Object

Attribute	Type	Default value	Required	Description
fd	string	-	Yes	The file descriptor. The fd is obtained through the FileSystemManager.open or FileSystemManager.openSync interface.
arrayBuffer	ArrayBuffer	-	Yes	The buffer for data writing must be an instance of ArrayBuffer.
offset	number	0	No	The write offset in the buffer, with the default value being 0.
length	number	0	No	The number of bytes to read from the file, with the default value being 0.
position	number	-	No	The starting position for file reading. If not provided or null, the file will be read from the current file pointer position. If the position is a positive integer, the file pointer position will remain unchanged and the file will be read from the position.

Return Value: ReadResult, the result of the file read.

Sample Code

©2013-2022 Tencent Cloud. All rights reserved.



```
const fs = wx.getFileSystemManager()
const ab = new ArrayBuffer(1024)
const fd = fs.openSync({
  filePath: ${wx.env.USER_DATA_PATH}/hello.txt,
  flag: 'a+'
})
const res = fs.readSync({
  fd: fd,
  arrayBuffer: ab,
  length: 10
})
```

©2013-2022 Tencent Cloud. All rights reserved.

```
    console.log(res)
```

.readCompressedFile

This method is used via FileSystemManager.readCompressedFile(Object object).

Feature Description: Reads the content of a local file of a specified compression type.

Parameter and Description: Object

Attribute	Type	Default value	Required	Description
filePath	string	-	Yes	The path of the file to be read (local user file or code package file).
compressionAlgorithm	string	-	Yes	File compression type, currently only supported br
success	function	-	No	Callback Function of Successful Interface Call
fail	function	-	No	Callback Function of Failing Interface Call
complete	function	-	No	Callback function executed upon the completion of the interface invocation (both successful and unsuccessful invocations)

Legal values for object.compressionAlgorithm are as follows.

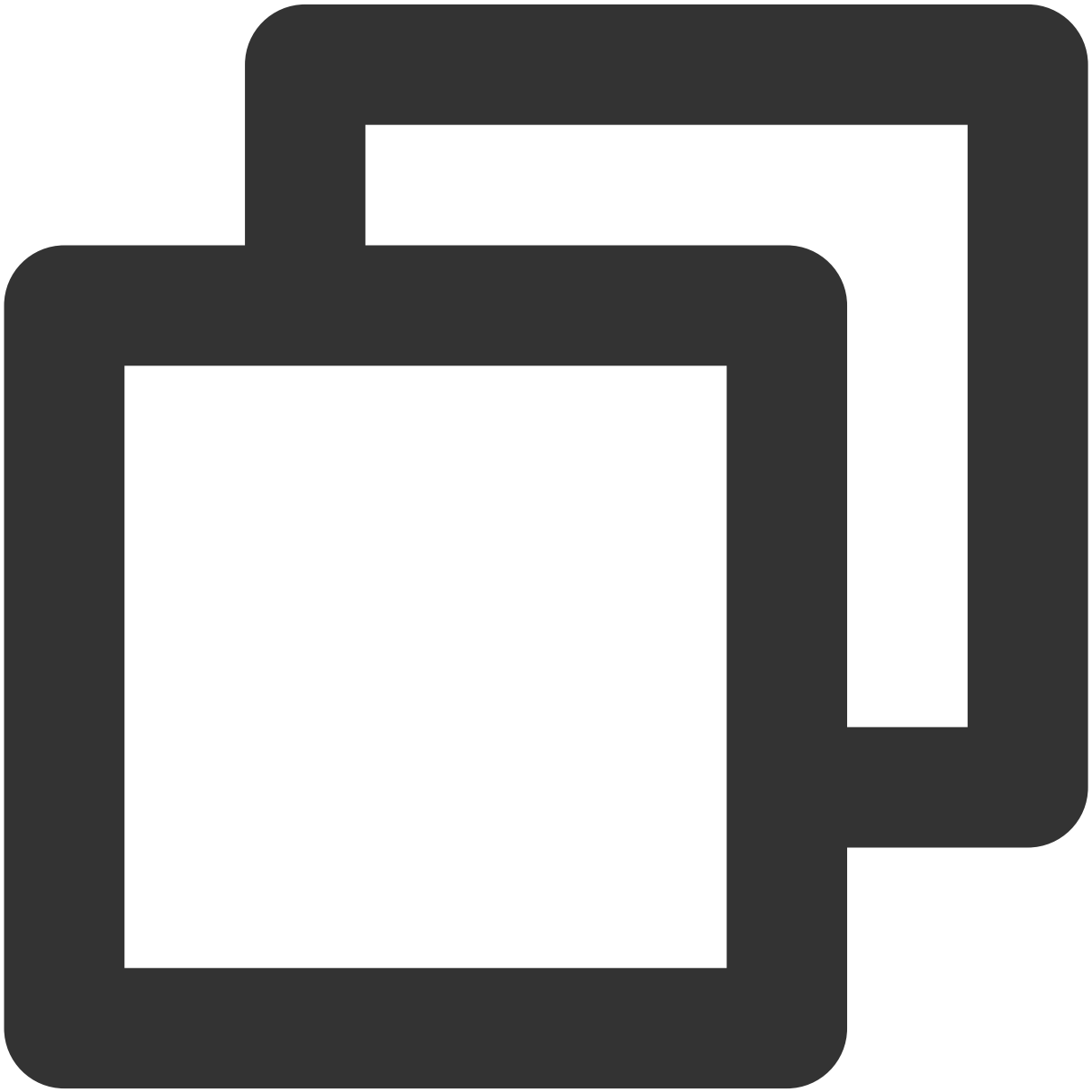
Legal values	Description
br	brotli zip file

Parameters for object.success callback function: Object res.

Attribute	Type	Description
data	ArrayBuffer	Version content

Sample Code

©2013-2022 Tencent Cloud. All rights reserved.



```
const fs = wx.getFileSystemManager()

//Asynchronous Interface
fs.readCompressedFile({
    filePath: '${wx.env.USER_DATA_PATH}/hello.br',
    compressionAlgorithm: 'br',
      success(res) {
    console.log(res.data)
  },
  fail(res) {
    console.log('readCompressedFile fail', res)
```

©2013-2022 Tencent Cloud. All rights reserved.

```
  }
})

//Synchronous Interface
  const data = fs.readCompressedFileSync({
    filePath: '${wx.env.USER_DATA_PATH}/hello.br',
    compressionAlgorithm: 'br',
})
  console.log(data)
```

.readCompressedFileSync

This method is used via ArrayBuffer FileSystemManager.readCompressedFileSync(Object object).

Feature Description: Synchronously reads the content of a local file of a specified compression type.

Parameter and Description: Object

Attribute	Type	Default value	Required	Description
filePath	string	-	Yes	The path of the file to be read (local user file or code package file).
compressionAlgorithm	string	-	Yes	File compression type, currently only supported br

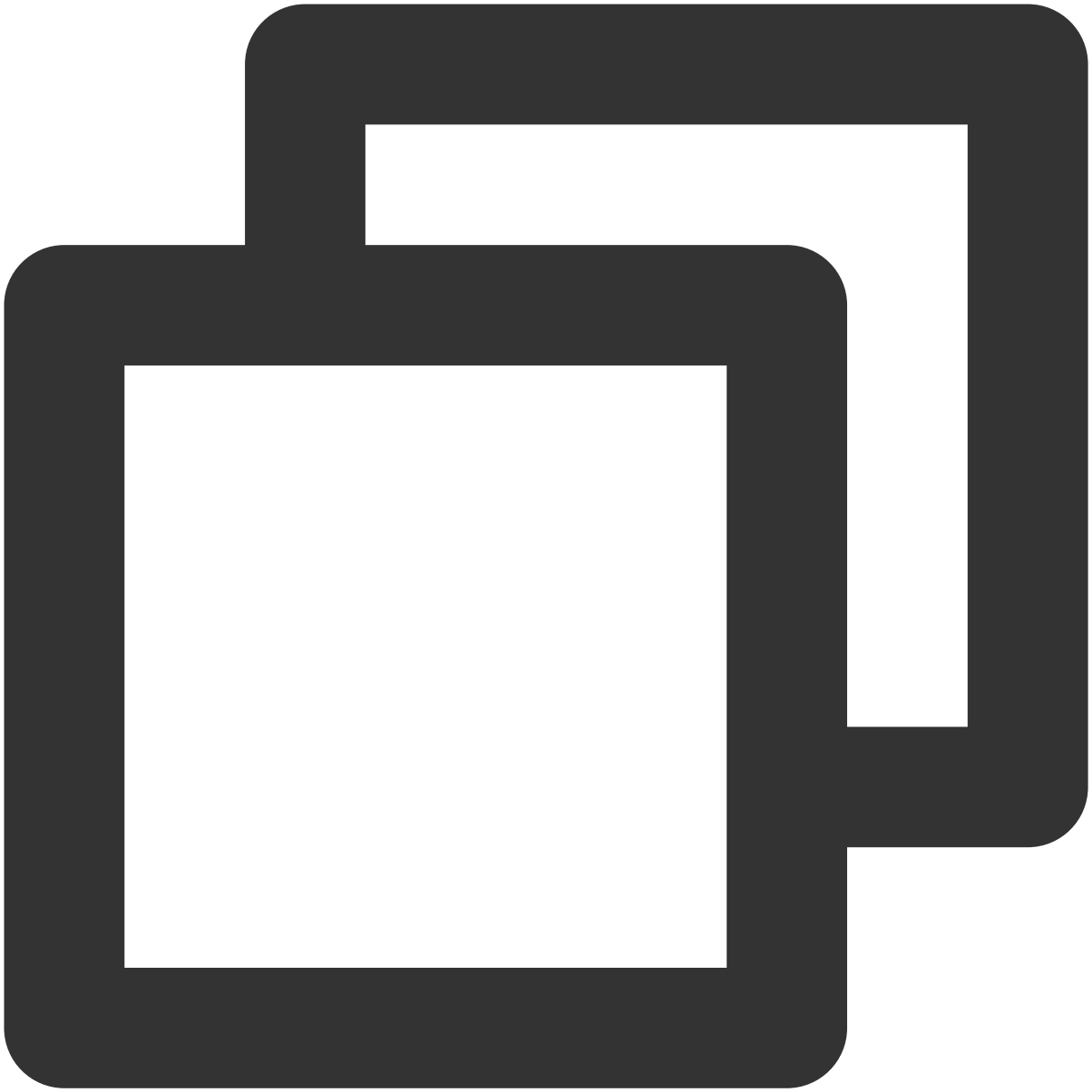
Legal values for object.compressionAlgorithm are as follows.

Legal values	Description
br	brotli zip file

Return Value: ArrayBuffer, the file reading result.

Sample Code

©2013-2022 Tencent Cloud. All rights reserved.



```
const fs = wx.getFileSystemManager()

//Asynchronous Interface
fs.readCompressedFile({
    filePath: '${wx.env.USER_DATA_PATH}/hello.br',
    compressionAlgorithm: 'br',
      success(res) {
    console.log(res.data)
  },
  fail(res) {
    console.log('readCompressedFile fail', res)
```

©2013-2022 Tencent Cloud. All rights reserved.

```
  }
})

//Synchronous Interface
try {
  const data = fs.readCompressedFileSync({
    filePath: '${wx.env.USER_DATA_PATH}/hello.br',
    compressionAlgorithm: 'br',
})
  console.log(data)
} catch (err) {
  console.log(err)
  }
```

.readdir

This method is used via FileSystemManager.readdir(Object object).

Feature Description: Reads the list of files in the directory.

Parameter and Description: Object

Attribute	Type	Default value	Required	Description
dirPath	string	-	Yes	The directory path to be read.
success	function	-	No	Callback Function of Successful Interface Call
fail	function	-	No	Callback Function of Failing Interface Call
complete	function	-	No	Callback function executed upon the completion of the interface invocation (both successful and unsuccessful invocations)

Parameters for object.success callback function: Object res.

Attribute	Type	Description
files	Array.\\<string>	An array of file names in the specified directory.

Parameters for object.fail callback function: Object res

Attribute	Type	Description

©2013-2022 Tencent Cloud. All rights reserved.

errMsg	string	Error Message

Valid values of res.errMsg

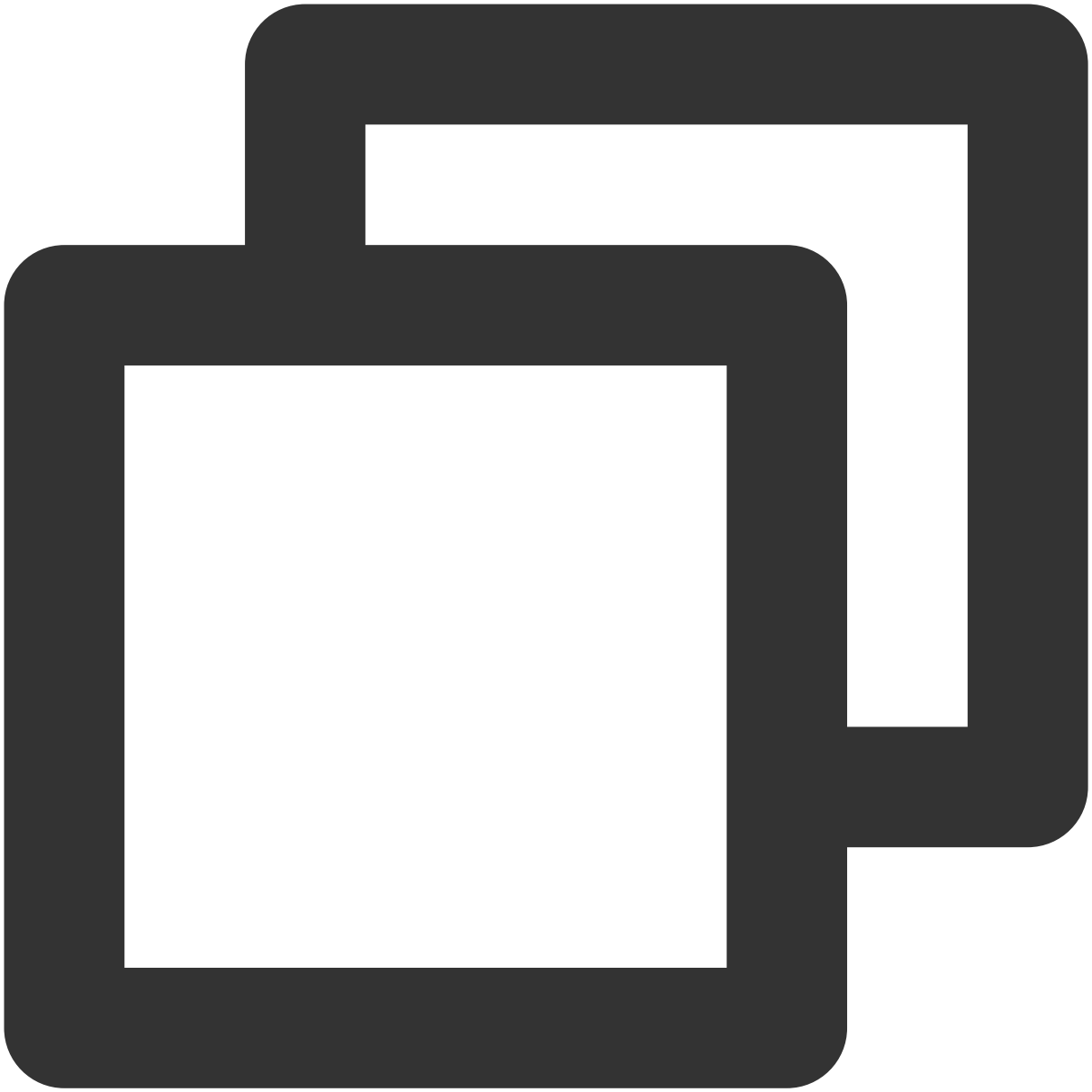
Value	Description
fail no such file or directory ${dirPath}	The directory does not exist.
fail not a directory ${dirPath}	dirPath is not a directory.
fail permission denied, open ${dirPath}	The specified filePath does not have read permissions.

Sample Code

Note：

The readdir interface could not access the file system root path (wxfile://).

©2013-2022 Tencent Cloud. All rights reserved.



```
const fs = wx.getFileSystemManager()
fs.readdir({
  dirPath: `${wx.env.USER_DATA_PATH}/example`,
  success(res) {
    console.log(res.files)
  },
  fail(res) {
    console.error(res)
  }
})
```

©2013-2022 Tencent Cloud. All rights reserved.

```
// Synchronization interface
try {
  const res = fs.readdirSync(`${wx.env.USER_DATA_PATH}/example`)
  console.log(res)
} catch(e) {
  console.error(e)
}
```

.readdirSync

This method is used via Array.<string> FileSystemManager.readdirSync(string dirPath).

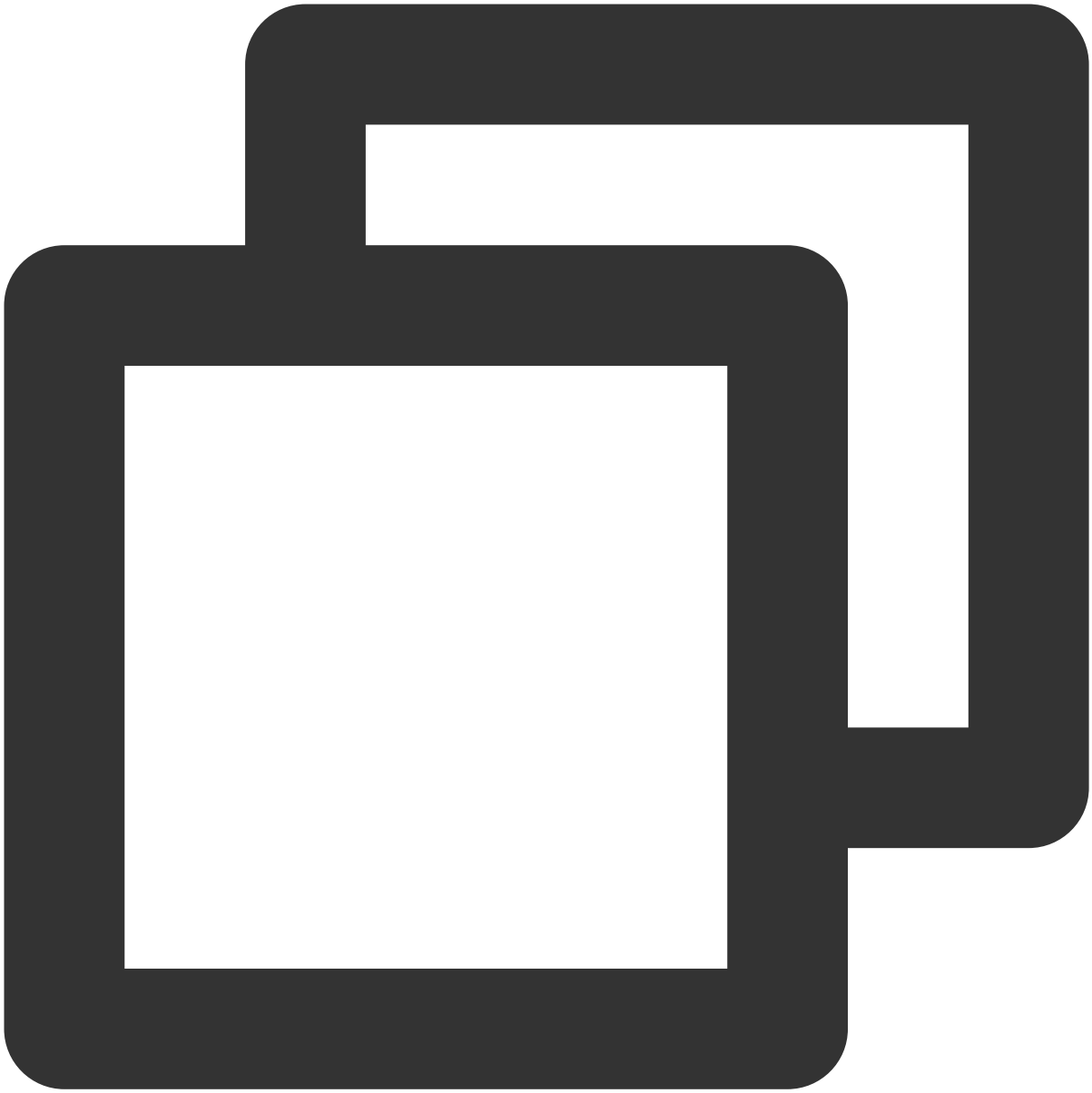
Feature Description: The synced version of FileSystemManager.readdir.

Parameters and Description: string dirPath, the directory path to be read.

Return Value: Array.<string> files, an array of file names in the specified directory.

Sample Code

©2013-2022 Tencent Cloud. All rights reserved.



```
const fs = wx.getFileSystemManager()
fs.readdir({
  dirPath: `${wx.env.USER_DATA_PATH}/example`,
  success(res) {
    console.log(res.files)
  },
  fail(res) {
    console.error(res)
  }
})
```

©2013-2022 Tencent Cloud. All rights reserved.

```
// Synchronization interface
try {
  const res = fs.readdirSync(`${wx.env.USER_DATA_PATH}/example`)
  console.log(res)
} catch(e) {
  console.error(e)
}
```

.readFile

This method is used via FileSystemManager.readFile(Object object).

Feature Description: Reads the content of a local file.The maximum size of a single file is 100M.

Parameter and Description: Object

Attribute	Type	Default value	Required	Description
filePath	string	-	Yes	The path of the file to be read.
encoding	string	-	No	Specifies the character encoding for reading the file. If encoding is not provided, the file's binary content is read in ArrayBuffer format.
success	function	-	No	Callback Function of Successful Interface Call
fail	function	-	No	Callback Function of Failing Interface Call
complete	function	-	No	Callback function executed upon the completion of the interface invocation (both successful and unsuccessful invocations)

Valid values for object.encoding

Value	Description
ascii	-
base64	-
binary	-
hex	-
ucs2/ucs-2/utf16le/utf-16le	Reads in little-endian order.

©2013-2022 Tencent Cloud. All rights reserved.

utf-8/utf8	-
latin1	-

Parameters for object.success callback function: Object res.

Attribute	Type	Description
data	string/ArrayBuffer	Version content

Parameters for object.fail callback function: Object res

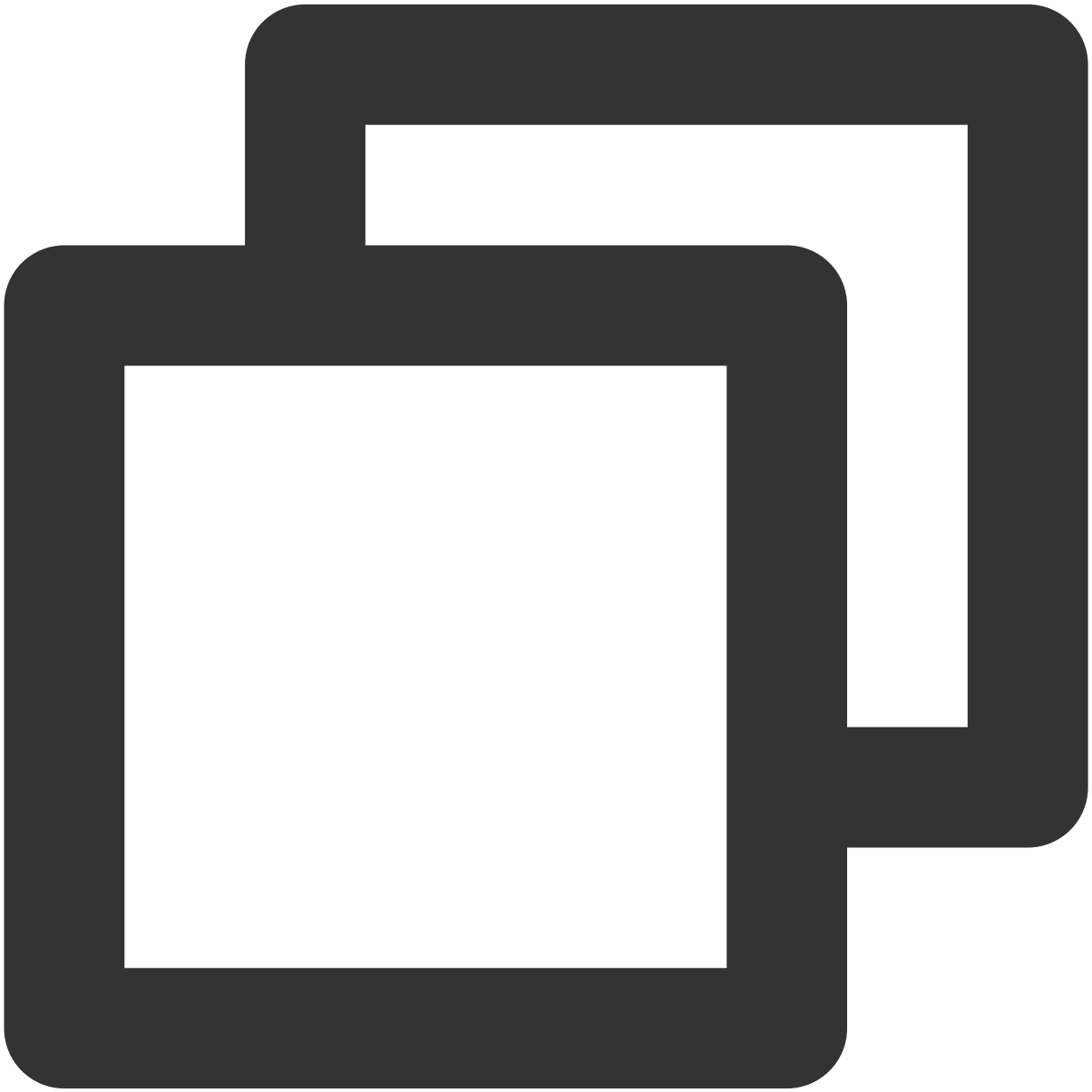
Attribute	Type	Description
errMsg	string	Error Message

Valid values of res.errMsg

Value	Description
fail no such file or directory, open ${filePath}	The directory of the specified filePath does not exist.
fail permission denied, open ${dirPath}	The specified filePath does not have read permissions.

Sample Code

©2013-2022 Tencent Cloud. All rights reserved.



```
const fs = wx.getFileSystemManager()
fs.readFile({
  filePath: `${wx.env.USER_DATA_PATH}/hello.txt`,
  encoding: 'utf8',
  position: 0,
  success(res) {
    console.log(res.data)
  },
  fail(res) {
    console.error(res)
  }
```

©2013-2022 Tencent Cloud. All rights reserved.

```
})

// Synchronization interface
try {
  const res = fs.readFileSync(`${wx.env.USER_DATA_PATH}/hello.txt`, 'utf8', 0)
  console.log(res)
} catch(e) {
  console.error(e)
}
```

.readFileSync

This method is used via string|ArrayBuffer FileSystemManager.readFileSync(string filePath, string encoding).

Feature Description: The synced version of FileSystemManager.readFile.

Parameter and Description:

string filePath: The path of the file to be read.

string encoding: Specifies the character encoding for reading the file. If encoding is not provided, the file's binary content is read in ArrayBuffer format.

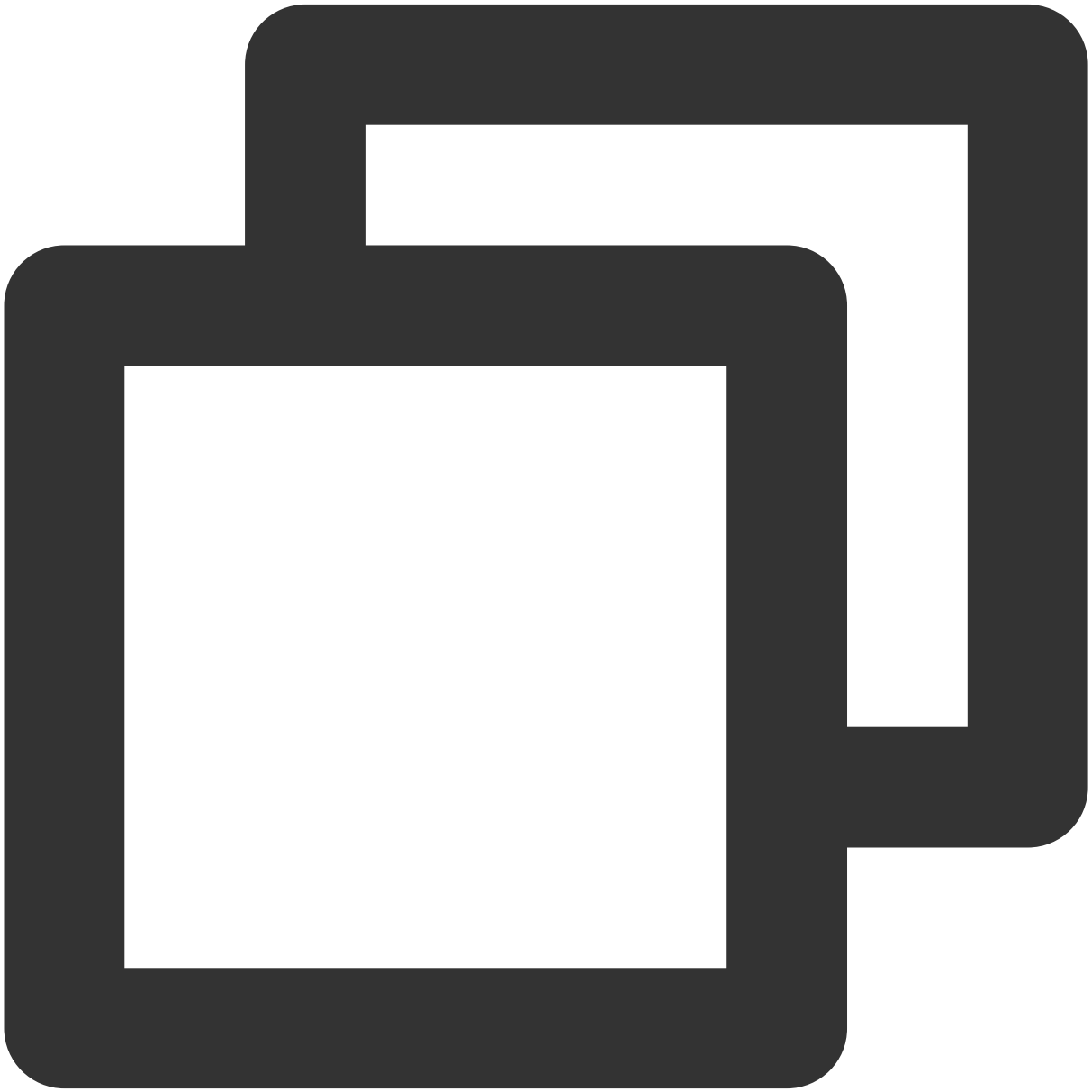
Valid Values for encoding

Value	Description
ascii	-
base64	-
binary	-
hex	-
ucs2/ucs-2/utf16le/utf-16le	Reads in little-endian order.
utf-8/utf8	-
latin1	-

Return Value: string|ArrayBuffer data, the file content.

Sample Code

©2013-2022 Tencent Cloud. All rights reserved.



```
const fs = wx.getFileSystemManager()
fs.readFile({
  filePath: `${wx.env.USER_DATA_PATH}/hello.txt`,
  encoding: 'utf8',
  position: 0,
  success(res) {
    console.log(res.data)
  },
  fail(res) {
    console.error(res)
  }
```

©2013-2022 Tencent Cloud. All rights reserved.

```
})

// Synchronization interface
try {
  const res = fs.readFileSync(`${wx.env.USER_DATA_PATH}/hello.txt`, 'utf8', 0)
  console.log(res)
} catch(e) {
  console.error(e)
}
```

.readZipEntry

This method is used via FileSystemManager.readZipEntry(Object object).

Feature Description: Reads a file within a compressed package.

Parameter and Description: Object

Attribute	Type	Default value	Required	Description
filePath	string	-	Yes	The path of the compressed package to be read (local path).
encoding	string	-	No	Uniformly specifies the character encoding for reading the file, effective only when the entries value is "all". If the entries value is "all" and encoding is not provided, the file's binary content is read in ArrayBuffer format.
entries	Array.<Object>/'all'	-	Yes	The list of files within the compressed package to be read (when "all" is passed in, it signifies reading all files within the compressed package).
success	function	-	No	Callback Function of Successful Interface Call
fail	function	-	No	Callback Function of Failing Interface Call
complete	function	-	No	Callback function executed upon the completion of the interface invocation (both successful and unsuccessful invocations)

Valid values for encoding

Value	Description

©2013-2022 Tencent Cloud. All rights reserved.

ascii	-
base64	-
binary	-
hex	-
ucs2/ucs-2/utf16le/utf-16le	Reads in little-endian order.
utf-8/utf8	-
latin1	-

Structure Attributes of entries

Structure attributes	Type	Default value	Required	Description
path	string	-	Yes	Path of the file within the compressed package.
encoding	string	-	No	Specify the character encoding of the file to be read, if encoding is not passed, the binary content of the file will be read in ArrayBuffer format, the legal value is the same as encoding.
position	number	-	No	Start reading from a specified position in the file; if not specified, reading begins from the start of the file. The reading range should be a left-closed and right-open interval [position, position+length). The valid range is [0, fileLength - 1] in bytes.
length	number	-	No	Specify the length of the file. If not specified, reading continues until the end of the file. The valid range is [1, fileLength] in bytes.

Parameters for object.success callback function: Object res.

Attribute	Type	Description
entries	Object	The result of the file reading. 'res.entries' is an object where the key is the file path and the value is an object 'FileItem', representing the reading result of that file. Each 'FileItem' includes 'data' (file content) and 'errMsg' (error information) attributes.

Structure Attributes of Entries

©2013-2022 Tencent Cloud. All rights reserved.

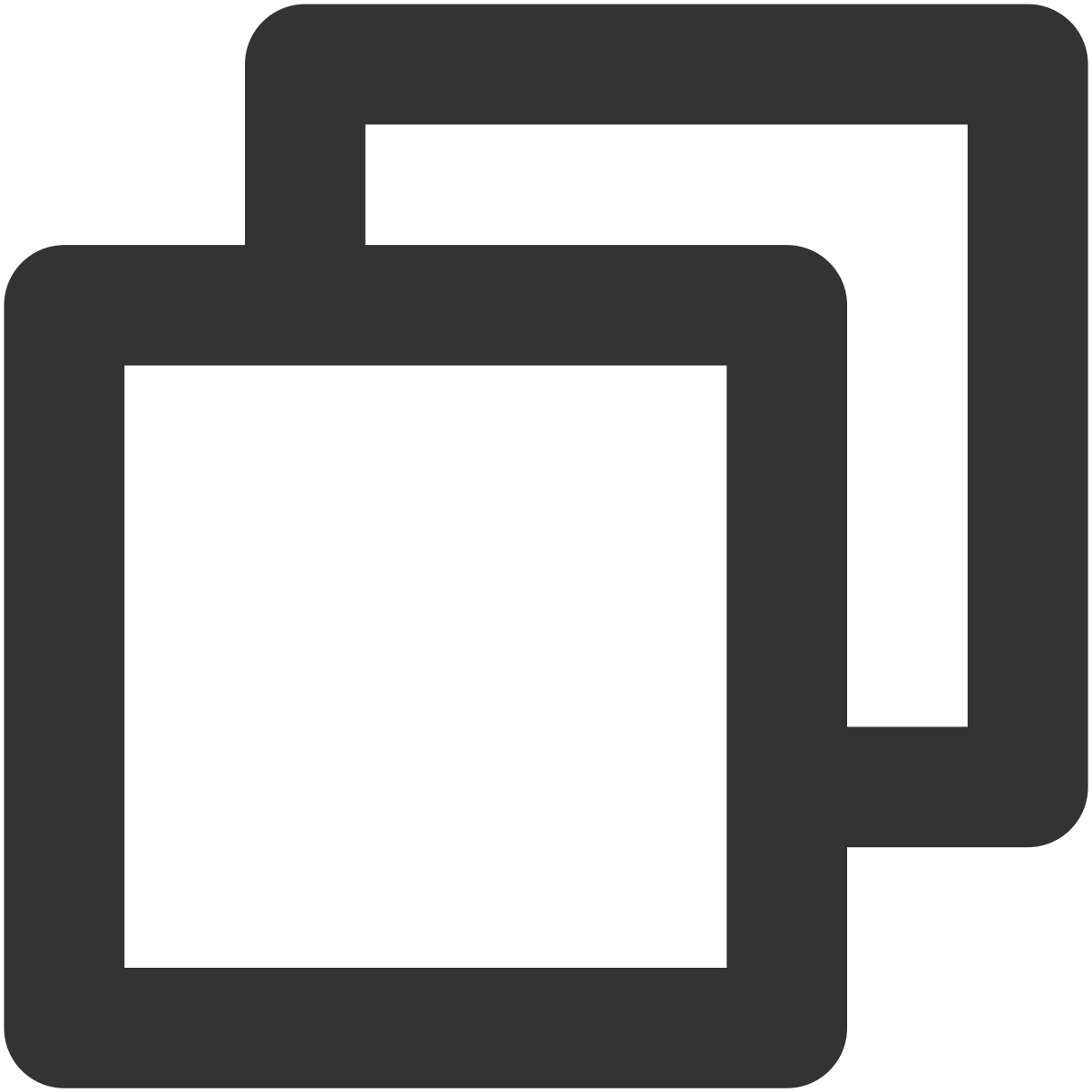
Structure attributes	Type	Description
path	string	File path

path Structure Attributes

Structure attributes	Type	Description
data	string/ArrayBuffer	Version content
errMsg	string	Error Message

Sample Code

©2013-2022 Tencent Cloud. All rights reserved.



```
const fs = wx.getFileSystemManager()
// Read one or multiple files within a zip.
fs.readZipEntry({
  filePath: 'wxfile://from/to.zip',
  entries: [{
 path: 'some_folder/my_file.txt', // File path within the zip
encoding: 'utf-8', // Specifies the character encoding for reading the file. If 'en
position: 0, // Start reading from a specified position in the file; if not specifi
    length: 10000, // Specify the length of the file. If not specified, reading con
  }, {
    path: 'other_folder/orther_file.txt', // File path within the zip
```

©2013-2022 Tencent Cloud. All rights reserved.

```
  }],
      success(res) {
    console.log(res.entries)
    // res.entries === {
    //     'some_folder/my_file.txt': {
    //         errMsg: 'readZipEntry:ok',
    //         data: 'xxxxxx'
    //     },
    //     'other_folder/orther_file.txt': {
    //         data: (ArrayBuffer)
    //  }
    //  }
  },
  fail(res) {
    console.log(res.errMsg)
  },
})

// Reads all files within the zip. A unified 'encoding' can be specified. 'Position
fs.readZipEntry({
  filePath: 'wxfile://from/to.zip',
  entries: 'all'
encoding: 'utf-8', // Uniformly specifies the character encoding for reading the fi
      success(res) {
    console.log(res.entries)
    // res.entries === {
    //     'some_folder/my_file.txt': {
    //         errMsg: 'readZipEntry:ok',
    //         data: 'xxxxxx'
    //     },
    //     'other_folder/orther_file.txt': {
    //         errMsg: 'readZipEntry:ok',
    //         data: 'xxxxxx'
    //  }
    //  }
  },
  fail(res) {
    console.log(res.errMsg)
  },
})
```

.removeSavedFile

This method is used via FileSystemManager.removeSavedFile(Object object).

©2013-2022 Tencent Cloud. All rights reserved.

Feature Description: Deletes the locally cached files saved under this mini program.

Parameter and Description: Object

Attribute	Type	Default value	Required	Description
filePath	string	-	Yes	Path for the files that need to be deleted.
success	function	-	No	Callback Function of Successful Interface Call
fail	function	-	No	Callback Function of Failing Interface Call
complete	function	-	No	Callback function executed upon the completion of the interface invocation (both successful and unsuccessful invocations)

Parameters for object.fail callback function: Object res

Attribute	Type	Description
errMsg	string	Error Message

Valid values of res.errMsg

Value	Description
fail file not exist	The file is not found in the specified tempFilePath.

.rename

This method is used via FileSystemManager.rename(Object object).

Feature Description: Renames the file. It allows the file to be moved from oldPath to newPath.

Parameter and Description: Object

Attribute	Type	Default value	Required	Description
oldPath	string	-	Yes	Source file path, can be a common file or directory, support local paths
newPath	string	-	Yes	New file path, local path support
success	function	-	No	Callback Function of Successful Interface Call

©2013-2022 Tencent Cloud. All rights reserved.

fail	function	-	No	Callback Function of Failing Interface Call
complete	function	-	No	Callback function executed upon the completion of the interface invocation (both successful and unsuccessful invocations)

Parameters for object.fail callback function: Object res

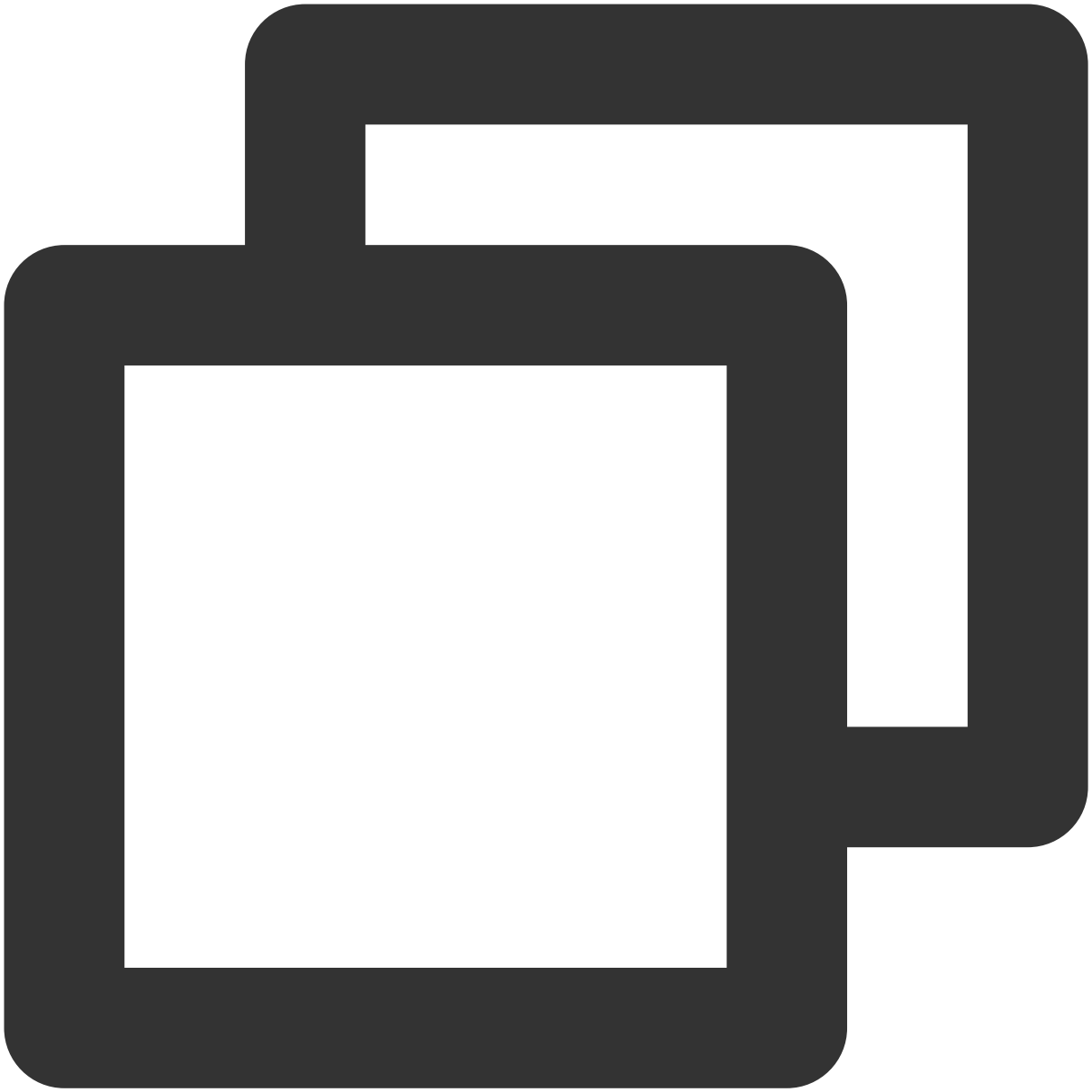
Attribute	Type	Description
errMsg	string	Error Message

Valid values of res.errMsg

Value	Description
fail permission denied, rename ${oldPath} -> ${newPath}	The specified source file or target file does not have write permissions.
fail no such file or directory, rename ${oldPath} -> ${newPath}	The source file does not exist, or the parent directory of the destination file path does not exist.

Sample Code

©2013-2022 Tencent Cloud. All rights reserved.



```
const fs = wx.getFileSystemManager()
fs.rename({
  oldPath: `${wx.env.USER_DATA_PATH}/hello.txt`,
  newPath: `${wx.env.USER_DATA_PATH}/hello_new.txt`,
  success(res) {
    console.log(res)
  },
  fail(res) {
    console.error(res)
  }
})
```

©2013-2022 Tencent Cloud. All rights reserved.

```
// Synchronization interface
try {
  const res = fs.renameSync(
    `${wx.env.USER_DATA_PATH}/hello.txt`,
    `${wx.env.USER_DATA_PATH}/hello_new.txt`
  )
  console.log(res)
} catch(e) {
  console.error(e)
}
```

.renameSync

This method is used via FileSystemManager.renameSync(string oldPath, string newPath).

Feature Description: The synced version of FileSystemManager.rename.

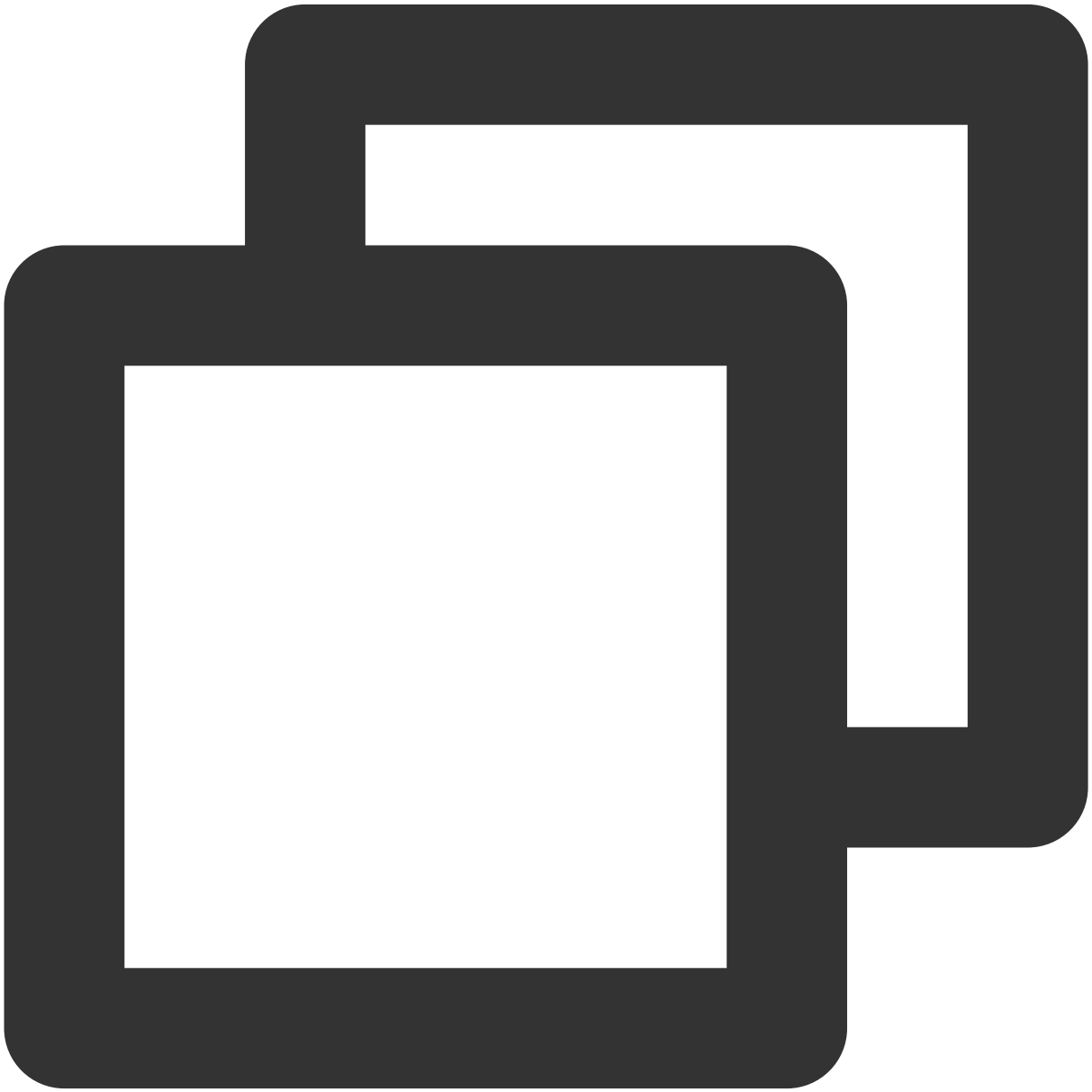
Parameter and Description

string oldPath: The source file path, which can be a regular file or a directory.

string newPath: The path for the new file.

Sample Code

©2013-2022 Tencent Cloud. All rights reserved.



```
const fs = wx.getFileSystemManager()
fs.rename({
  oldPath: `${wx.env.USER_DATA_PATH}/hello.txt`,
  newPath: `${wx.env.USER_DATA_PATH}/hello_new.txt`,
  success(res) {
    console.log(res)
  },
  fail(res) {
    console.error(res)
  }
})
```

©2013-2022 Tencent Cloud. All rights reserved.

```
// Synchronization interface
try {
  const res = fs.renameSync(
    `${wx.env.USER_DATA_PATH}/hello.txt`,
    `${wx.env.USER_DATA_PATH}/hello_new.txt`
  )
  console.log(res)
} catch(e) {
  console.error(e)
}
```

.rmdir

This method is used via FileSystemManager.rmdir(Object object).

Feature Description: Deletes directory.

Parameter and Description: Object

Attribute	Type	Default value	Required	Description
dirPath	string	-	Yes	The path of directory to be deleted.
recursive	boolean	false	No	Determines whether to recursively delete the directory. If set to true, it will delete the directory along with all its subdirectories and files.
success	function	-	No	Callback Function of Successful Interface Call
fail	function	-	No	Callback Function of Failing Interface Call
complete	function	-	No	Callback function executed upon the completion of the interface invocation (both successful and unsuccessful invocations)

Parameters for object.fail callback function: Object res

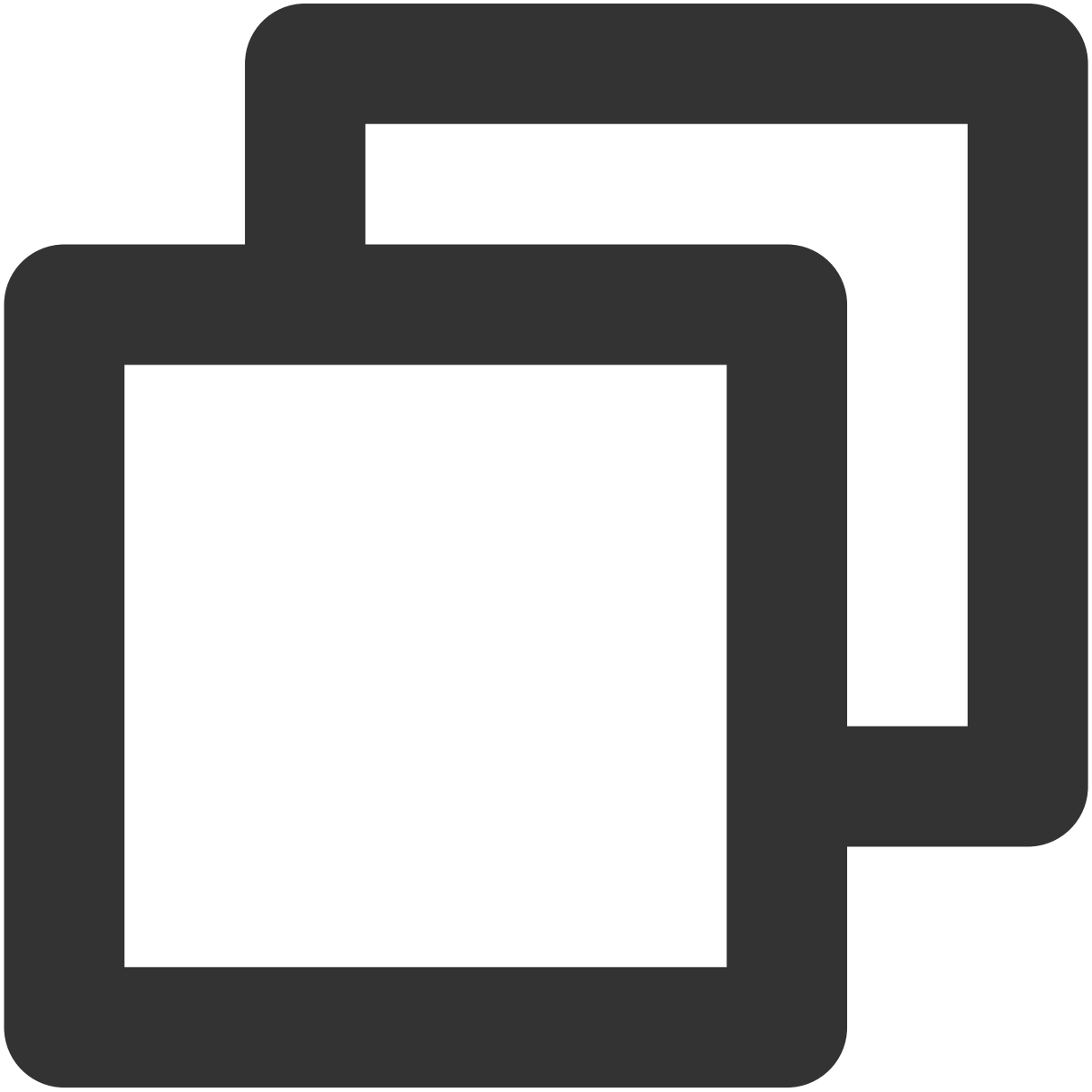
Attribute	Type	Description
errMsg	string	Error Message

Valid values of res.errMsg

©2013-2022 Tencent Cloud. All rights reserved.

Value	Description
fail no such file or directory ${dirPath}	The directory does not exist.
fail directory not empty	The directory is not empty.
fail permission denied, open ${dirPath}	The specified dirPath does not have write permissions.

Sample Code



©2013-2022 Tencent Cloud. All rights reserved.

```
const fs = wx.getFileSystemManager()
fs.rmdir({
  dirPath: `${wx.env.USER_DATA_PATH}/example`,
  recursive: false,
  success(res) {
    console.log(res)
  },
  fail(res) {
    console.error(res)
  }
})

// Synchronization interface
try {
  const res = fs.rmdirSync(`${wx.env.USER_DATA_PATH}/example`, false)
  console.log(res)
} catch(e) {
  console.error(e)
}
```

.rmdirSync

This method is used via FileSystemManager.rmdirSync(string dirPath, boolean recursive).

Feature Description: The synced version of FileSystemManager.rmdir.

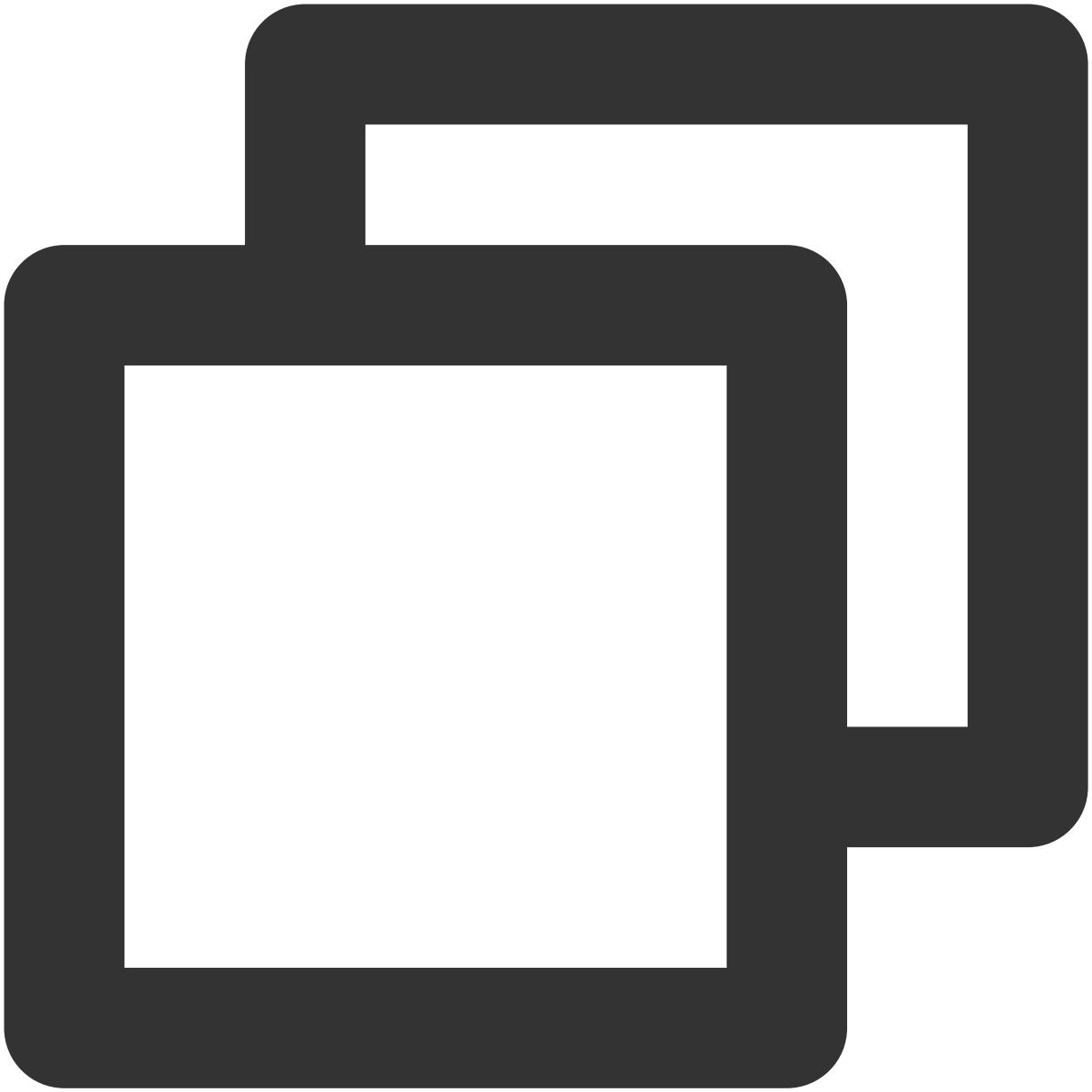
Parameter and Description:

string dirPath: The path of the directory to be deleted.

boolean recursive: Determines whether to recursively delete the directory. If set to true, it will delete the directory along with all its subdirectories and files.

Sample Code

©2013-2022 Tencent Cloud. All rights reserved.



```
const fs = wx.getFileSystemManager()
fs.rmdir({
  dirPath: `${wx.env.USER_DATA_PATH}/example`,
  recursive: false,
  success(res) {
    console.log(res)
  },
  fail(res) {
    console.error(res)
  }
})
```

©2013-2022 Tencent Cloud. All rights reserved.

```
// Synchronization interface
try {
  const res = fs.rmdirSync(`${wx.env.USER_DATA_PATH}/example`, false)
  console.log(res)
} catch(e) {
  console.error(e)
}
```

.saveFile

This method is used via FileSystemManager.saveFile(Object object).

Feature Description: Saves temporary files locally. Upon successful execution, the temporary file will be moved, rendering the tempFilePath unusable.

Parameter and Description: Object

Attribute	Type	Default value	Required	Description
tempFilePath	string	-	Yes	The path for temporary file storage.
filePath	string	-	No	Path of files to be stored.
success	function	-	No	Callback Function of Successful Interface Call
fail	function	-	No	Callback Function of Failing Interface Call
complete	function	-	No	Callback function executed upon the completion of the interface invocation (both successful and unsuccessful invocations)

Parameters for object.success callback function: Object res.

Attribute	Type	Description
savedFilePath	number	File path after storage

Parameters for object.fail callback function: Object res

Attribute	Type	Description
errMsg	string	Error Message

©2013-2022 Tencent Cloud. All rights reserved.

Valid values of res.errMsg

Value	Description
fail tempFilePath file not exist	The file is not found in the specified tempFilePath.
fail permission denied, open "${filePath}"	The specified filePath does not have write permissions.
fail no such file or directory "${dirPath}"	Parent directory does not exist.

.saveFileSync

This method is used via number FileSystemManager.saveFileSync(string tempFilePath, string filePath).

Feature Description: The synced version of FileSystemManager.saveFile.

Parameter and Description:

string tempFilePath: The path for temporary file storage.

string filePath: The path of the file to be stored.

Return Value: number savedFilePath, the path of the file after storage.

.stat

This method is used via FileSystemManager.stat(Object object).

Feature Description: Gets the Stats object of the file.

Parameter and Description: Object

Attribute	Type	Default value	Required	Description
path	string	-	Yes	Path of file/directory
recursive	boolean	false	No	Whether to recursively get the Stats information of each file in the directory.
success	function	-	No	Callback Function of Successful Interface Call
fail	function	-	No	Callback Function of Failing Interface Call
complete	function	-	No	Callback function executed upon the completion of the interface invocation (both successful and unsuccessful

©2013-2022 Tencent Cloud. All rights reserved.

				invocations)

Parameters for object.success callback function: Object res.

Attribute	Type	Description
stats	Stats/Object	When recursive is false, res.stats is a Stats object. When recursive is true and path is a directory path, res.stats is an Object, with the key being the relative path rooted at path, and the value being the Stats object corresponding to that path.

Parameters for object.fail callback function: Object res

Attribute	Type	Description
errMsg	string	Error Message

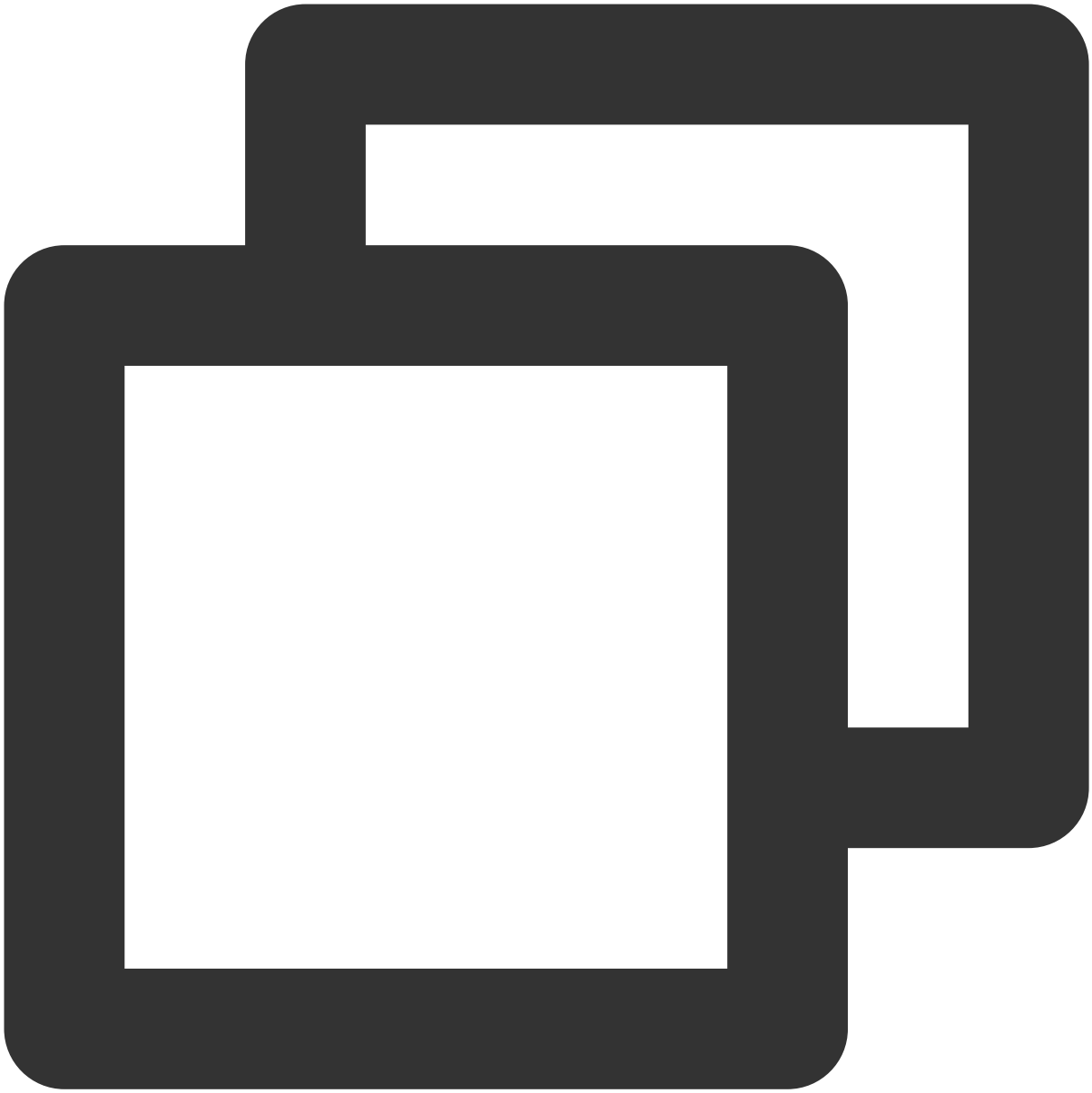
Valid values of res.errMsg

Value	Description
fail permission denied, open ${path}	The specified path does not have read permissions.
fail no such file or directory ${path}	The file does not exist.

Sample Code

When recursive is set to false.

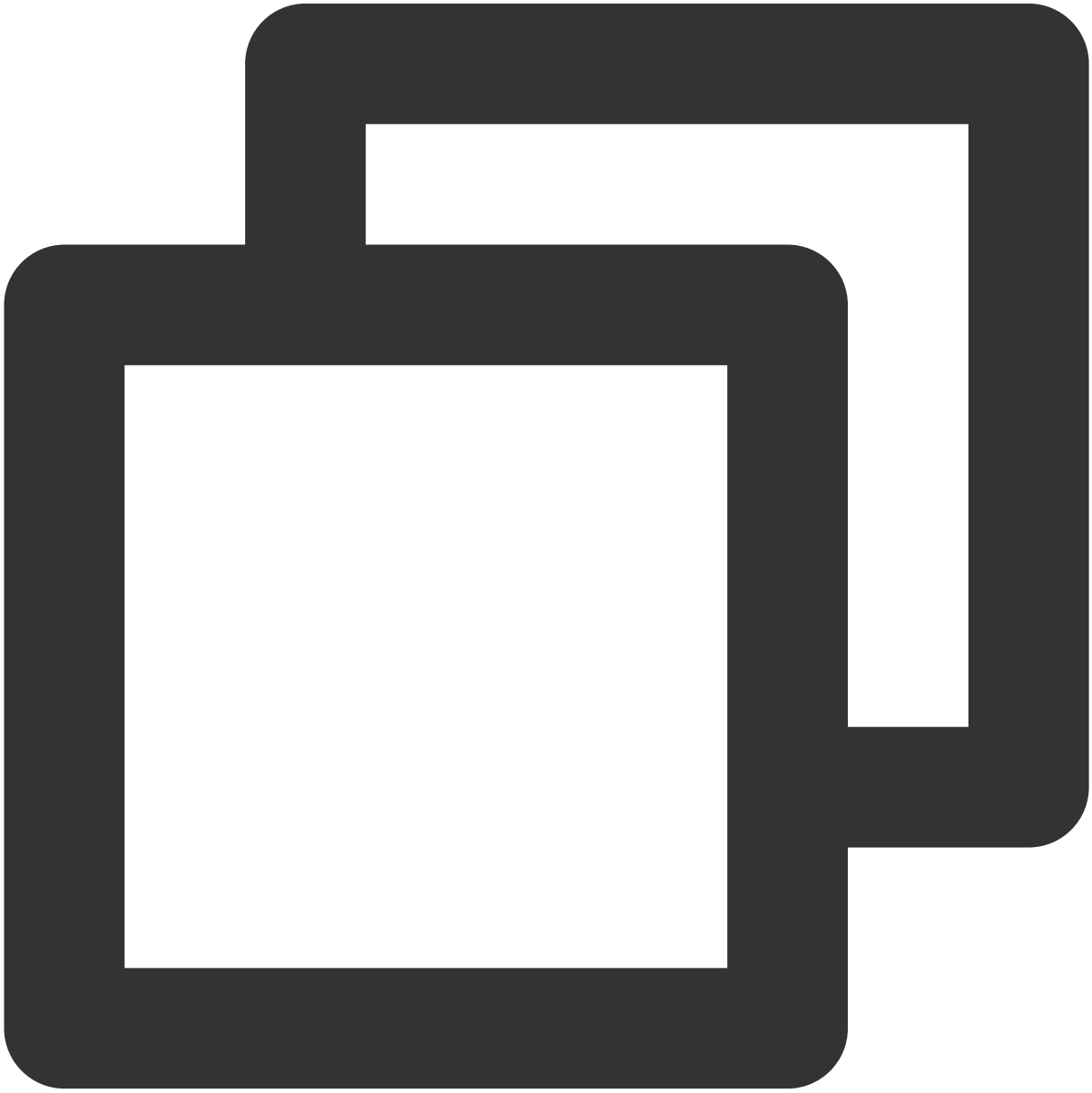
©2013-2022 Tencent Cloud. All rights reserved.



```
// Asynchronous version
let fs = wx.getFileSystemManager()
fs.stat({
  path: `${wx.env.USER_DATA_PATH}/testDir`,
  success: res => {
    console.log(res.stats.isDirectory())
  }
})
// Synchronise versions
fs.statSync(`${wx.env.USER_DATA_PATH}/testDir`, false)
```

©2013-2022 Tencent Cloud. All rights reserved.

When recursive is set to true.



```
let fs = wx.getFileSystemManager()
// Asynchronous version
fs.stat({
  path: `${wx.env.USER_DATA_PATH}/testDir`,
  recursive: true,
  success: res => {
    Object.keys(res.stats).forEach(path => {
      let stats = res.stats[path]
      console.log(path, stats.isDirectory())
```

©2013-2022 Tencent Cloud. All rights reserved.

```
    })
  }
})
// Synchronise versions
fs.statSync(`${wx.env.USER_DATA_PATH}/testDir`, true)
```

.statSync

This method is used via Stats|Object FileSystemManager.statSync(string path, boolean recursive).

Feature Description: The synced version of FileSystemManager.stat.

Parameter and Description:

string path: File/directory path.

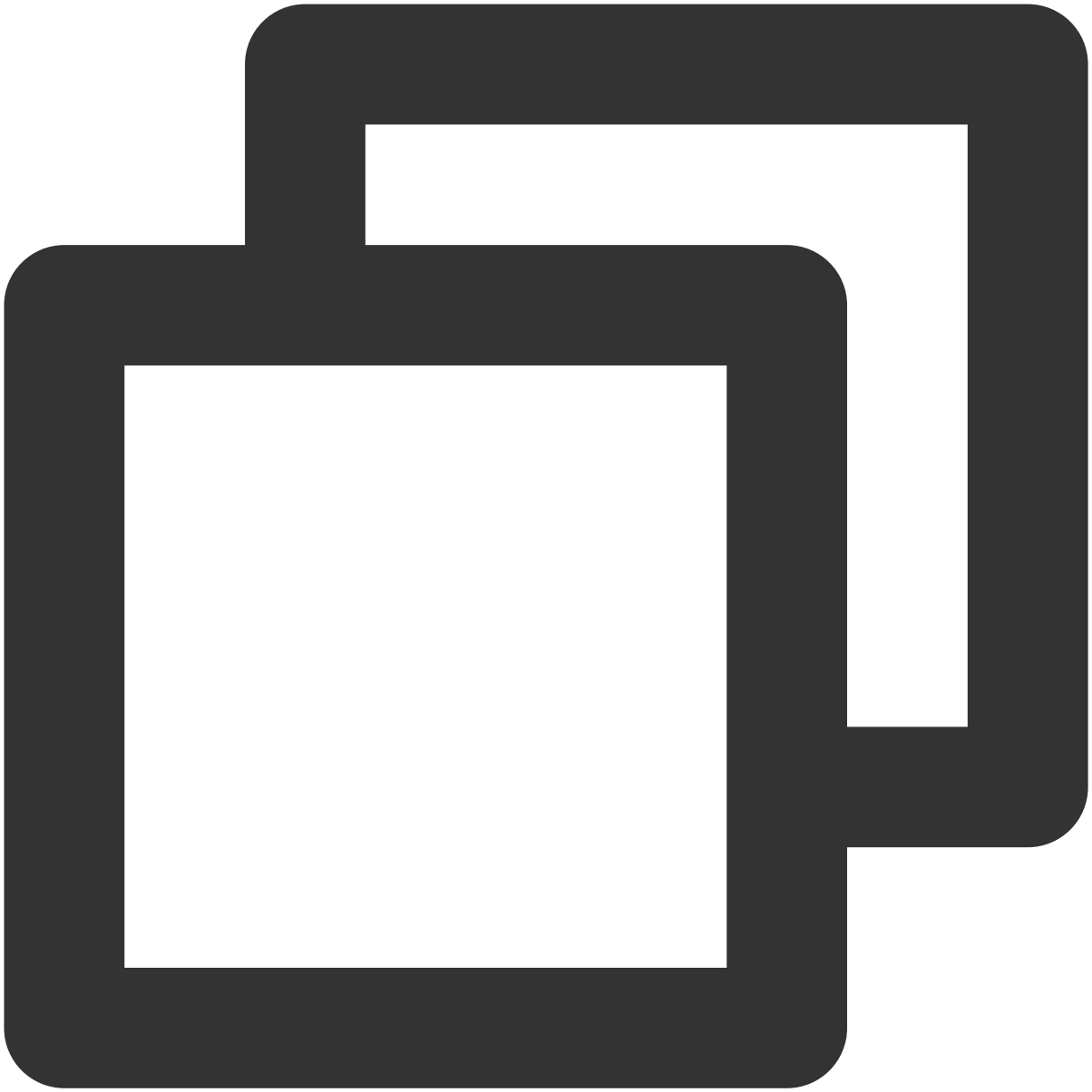
boolean recursive: Determines whether to recursively retrieve the Stats information for each file in the directory.

Return Value: Stats|Object stats. When recursive is false, res.stats is a Stats object. When recursive is true and path is a directory path, res.stats is an Object, with the key being the relative path rooted at path, and the value being the Stats object corresponding to that path.

Sample Code

When recursive is set to false.

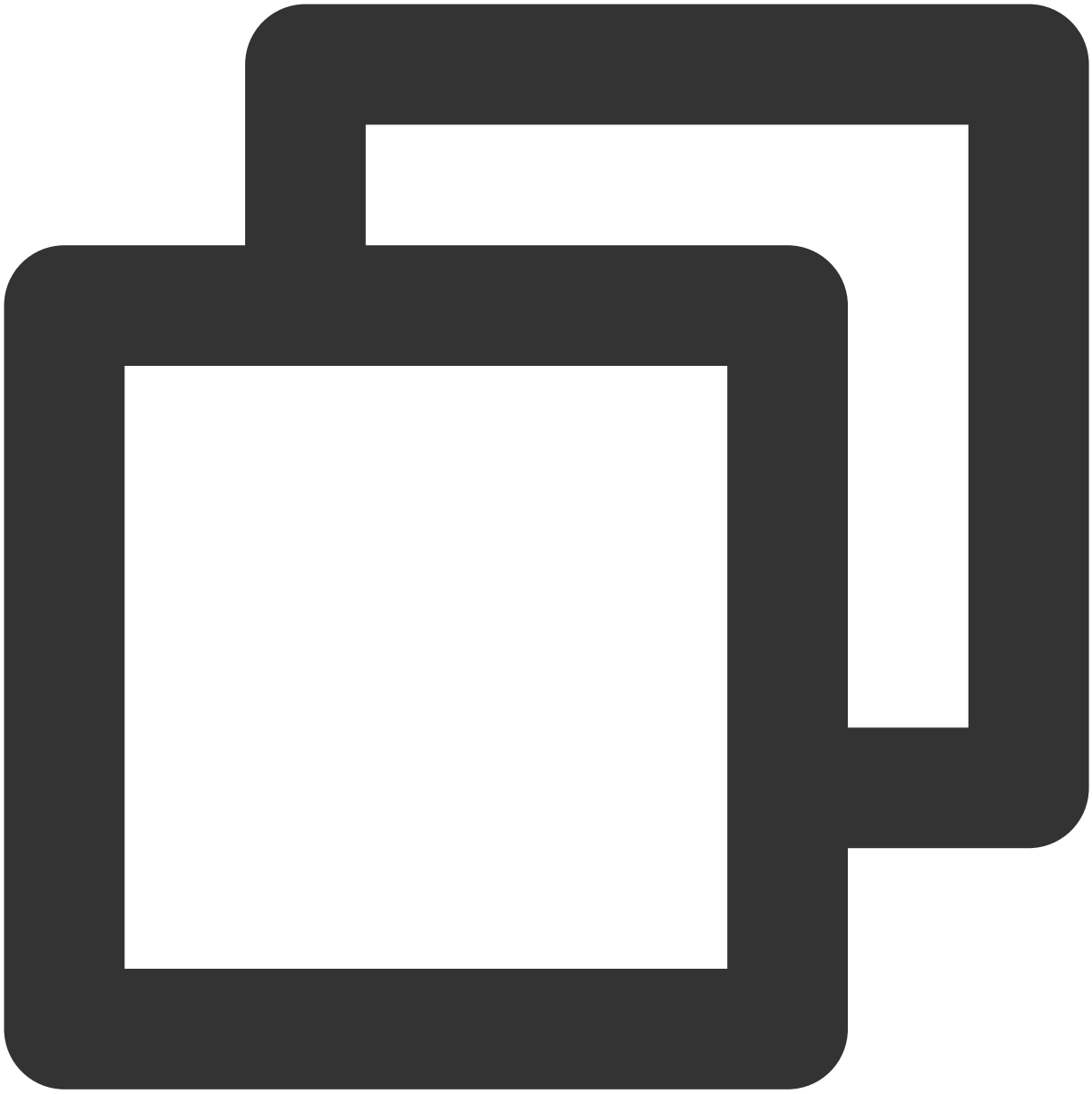
©2013-2022 Tencent Cloud. All rights reserved.



```
// Asynchronous version
let fs = wx.getFileSystemManager()
fs.stat({
  path: `${wx.env.USER_DATA_PATH}/testDir`,
  success: res => {
    console.log(res.stats.isDirectory())
  }
})
// Synchronise versions
fs.statSync(`${wx.env.USER_DATA_PATH}/testDir`, false)
```

©2013-2022 Tencent Cloud. All rights reserved.

When recursive is set to true.



```
let fs = wx.getFileSystemManager()
// Asynchronous version
fs.stat({
  path: `${wx.env.USER_DATA_PATH}/testDir`,
  recursive: true,
  success: res => {
    Object.keys(res.stats).forEach(path => {
      let stats = res.stats[path]
      console.log(path, stats.isDirectory())
```

©2013-2022 Tencent Cloud. All rights reserved.

```
    })
  }
})
// Synchronise versions
fs.statSync(`${wx.env.USER_DATA_PATH}/testDir`, true)
```

.truncate

This method is used via FileSystemManager.truncate(Object object).

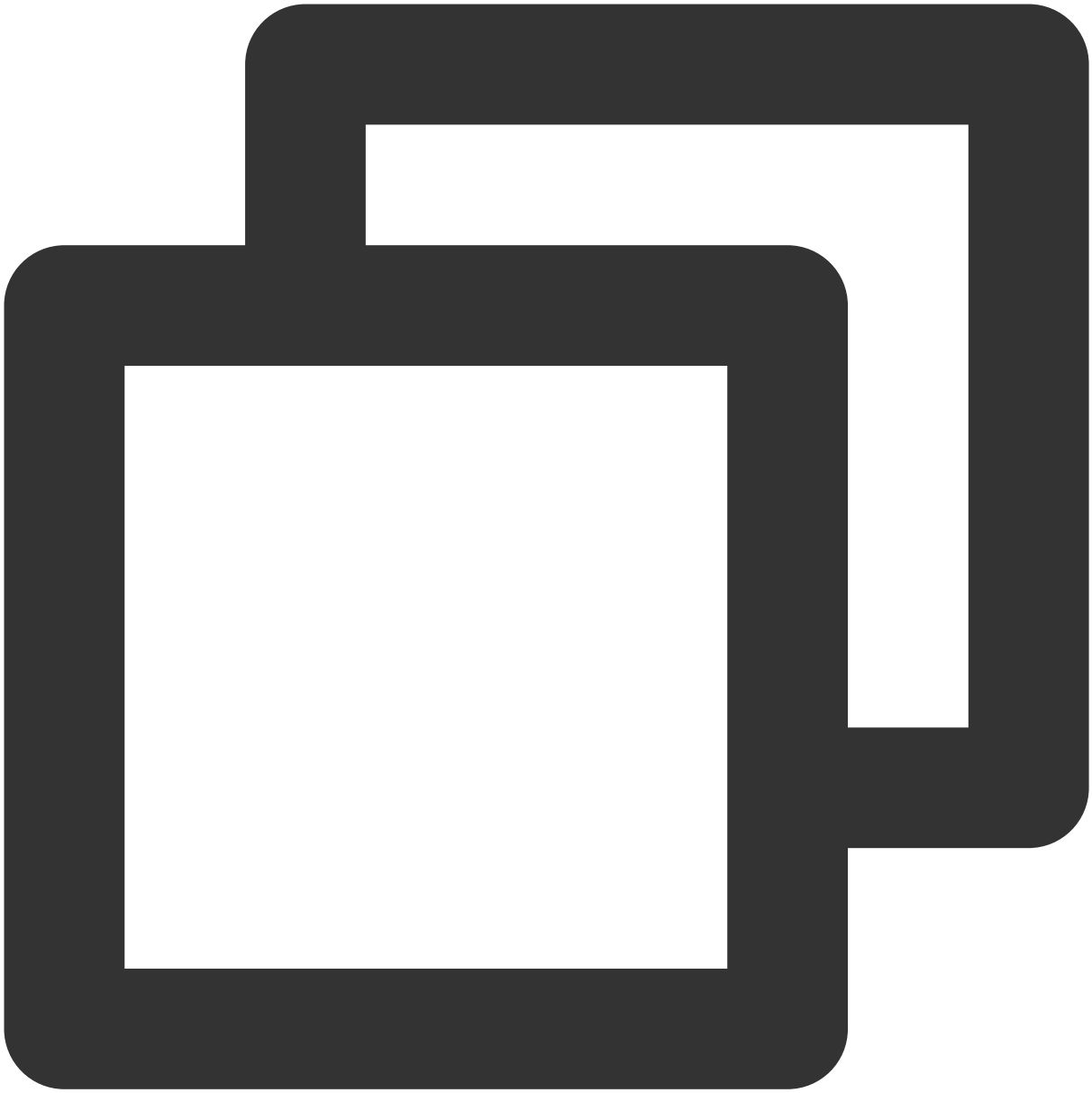
Feature Description: Executes a truncation operation on the file content.

Parameter and Description: Object

Attribute	Type	Default value	Required	Description
filePath	string	-	Yes	Path of the file to be truncated (local path)
length	number	0	No	By default, the truncation position is 0. If the specified length value is less than the file length (measured in bytes), only the preceding number of bytes equal to the specified length will be retained in the file, and the remaining content will be deleted. If the specified length exceeds the file length, the file will be expanded, with the extended section being filled with null bytes ('\\0').
success	function	-	No	Callback Function of Successful Interface Call
fail	function	-	No	Callback Function of Failing Interface Call
complete	function	-	No	Callback function executed upon the completion of the interface invocation (both successful and unsuccessful invocations)

Sample Code

©2013-2022 Tencent Cloud. All rights reserved.



```
const fs = wx.getFileSystemManager()
fs.truncate({
  filePath: ${wx.env.USER_DATA_PATH}/hello.txt,
  length: 10, // Truncate the file starting from the 10th byte
      success(res) {
    console.log(res)
  }
})
```

.truncateSync

©2013-2022 Tencent Cloud. All rights reserved.

This method is used via undefined FileSystemManager.truncateSync(Object object).

Feature Description: Performs truncation operation on file content (synced version of truncate).

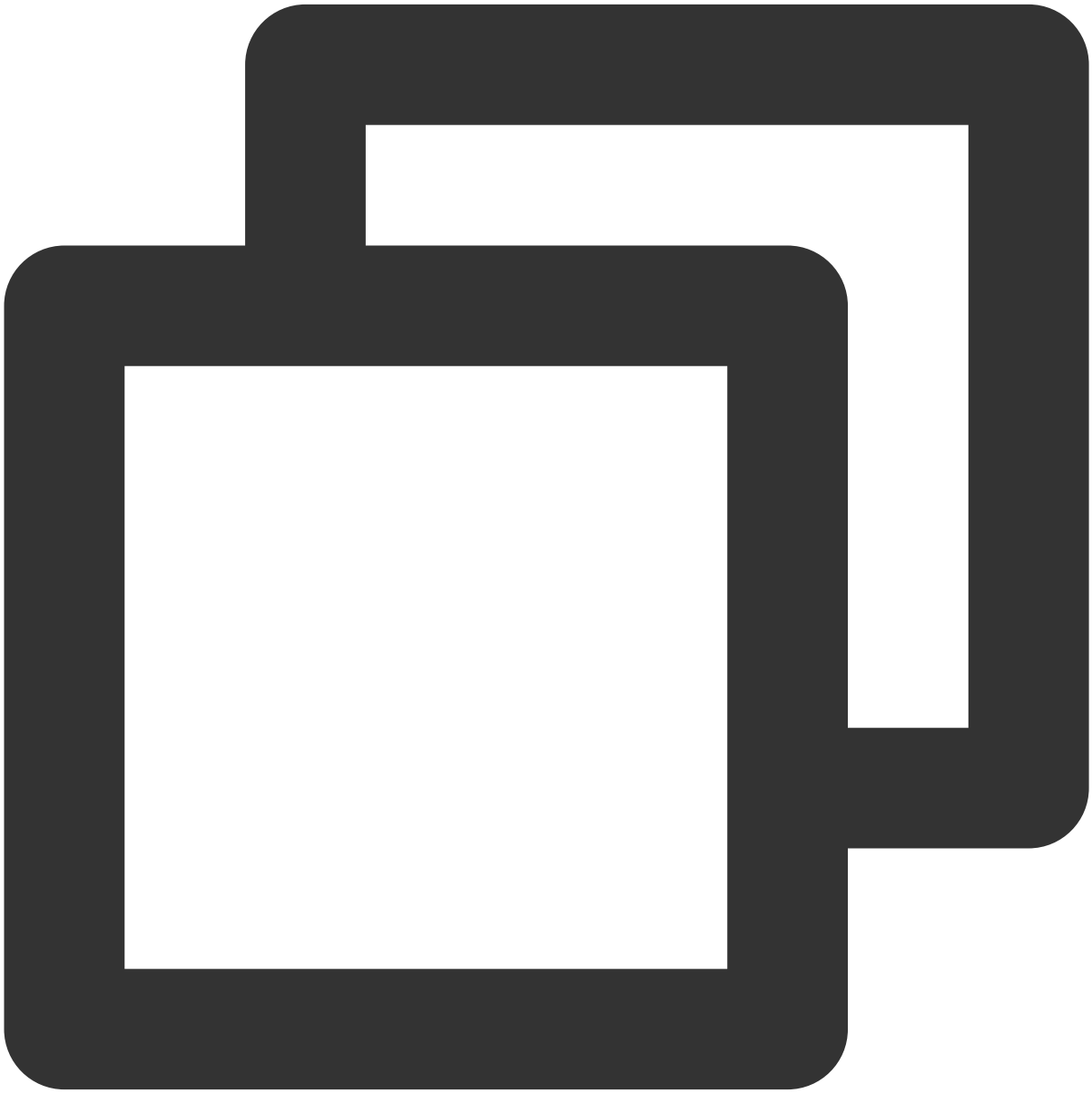
Parameter and Description: Object

Attribute	Type	Default value	Required	Description
filePath	string		Yes	Path of the file to be truncated (local path)
length	number	0	No	By default, the truncation position is 0. If the specified length value is less than the file length (measured in bytes), only the preceding number of bytes equal to the specified length will be retained in the file, and the remaining content will be deleted. If the specified length exceeds the file length, the file will be expanded, with the extended section being filled with null bytes ('\\0').

Return Value: undefined.

Sample Code

©2013-2022 Tencent Cloud. All rights reserved.



```
const fs = wx.getFileSystemManager()
fs.truncateSync({
  filePath: ${wx.env.USER_DATA_PATH}/hello.txt,
  length: 10, // Truncate the file starting from the 10th byte
})
```

.unlink

This method is used via FileSystemManager.unlink(Object object).

©2013-2022 Tencent Cloud. All rights reserved.

Feature Description: Deletes files.

Parameter and Description: Object

Attribute	Type	Default value	Required	Description
filePath	string	-	Yes	Path for the files that need to be deleted.
success	function	-	No	Callback Function of Successful Interface Call
fail	function	-	No	Callback Function of Failing Interface Call
complete	function	-	No	Callback function executed upon the completion of the interface invocation (both successful and unsuccessful invocations)

Parameters for object.fail callback function: Object res

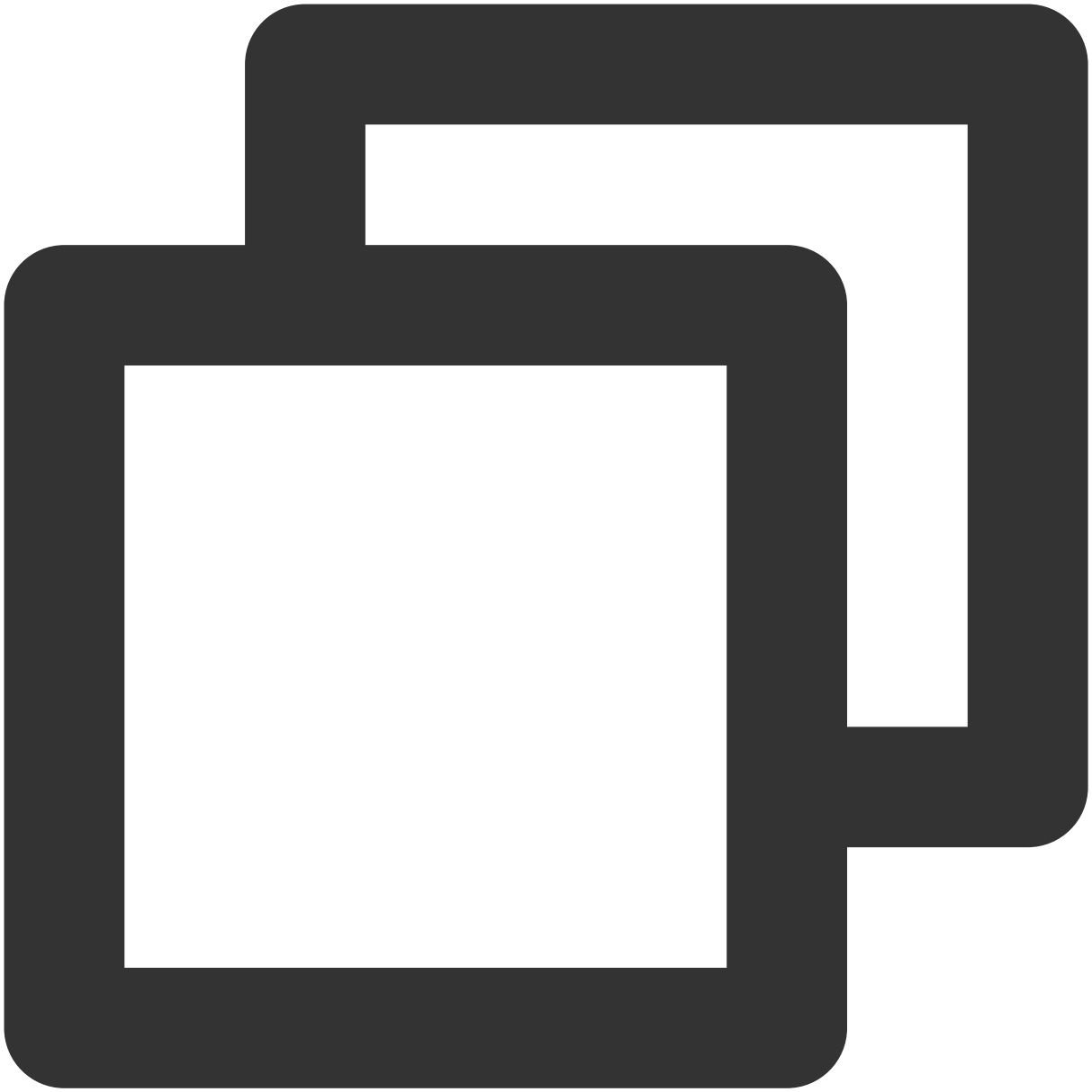
Attribute	Type	Description
errMsg	string	Error Message

Valid values of res.errMsg

Value	Description
fail permission denied, open ${path}	The specified path does not have read permissions.
fail no such file or directory ${path}	The file does not exist.
fail operation not permitted, unlink ${filePath}	The filePath passed in is a directory.

Sample Code

©2013-2022 Tencent Cloud. All rights reserved.



```
const fs = wx.getFileSystemManager()
fs.unlink({
  filePath: `${wx.env.USER_DATA_PATH}/hello.txt`,
  success(res) {
    console.log(res)
  },
  fail(res) {
    console.error(res)
  }
})
```

©2013-2022 Tencent Cloud. All rights reserved.

```
// Synchronization interface
try {
  const res = fs.unlinkSync(`${wx.env.USER_DATA_PATH}/hello.txt`)
  console.log(res)
} catch(e) {
  console.error(e)
}
```

.unlinkSync

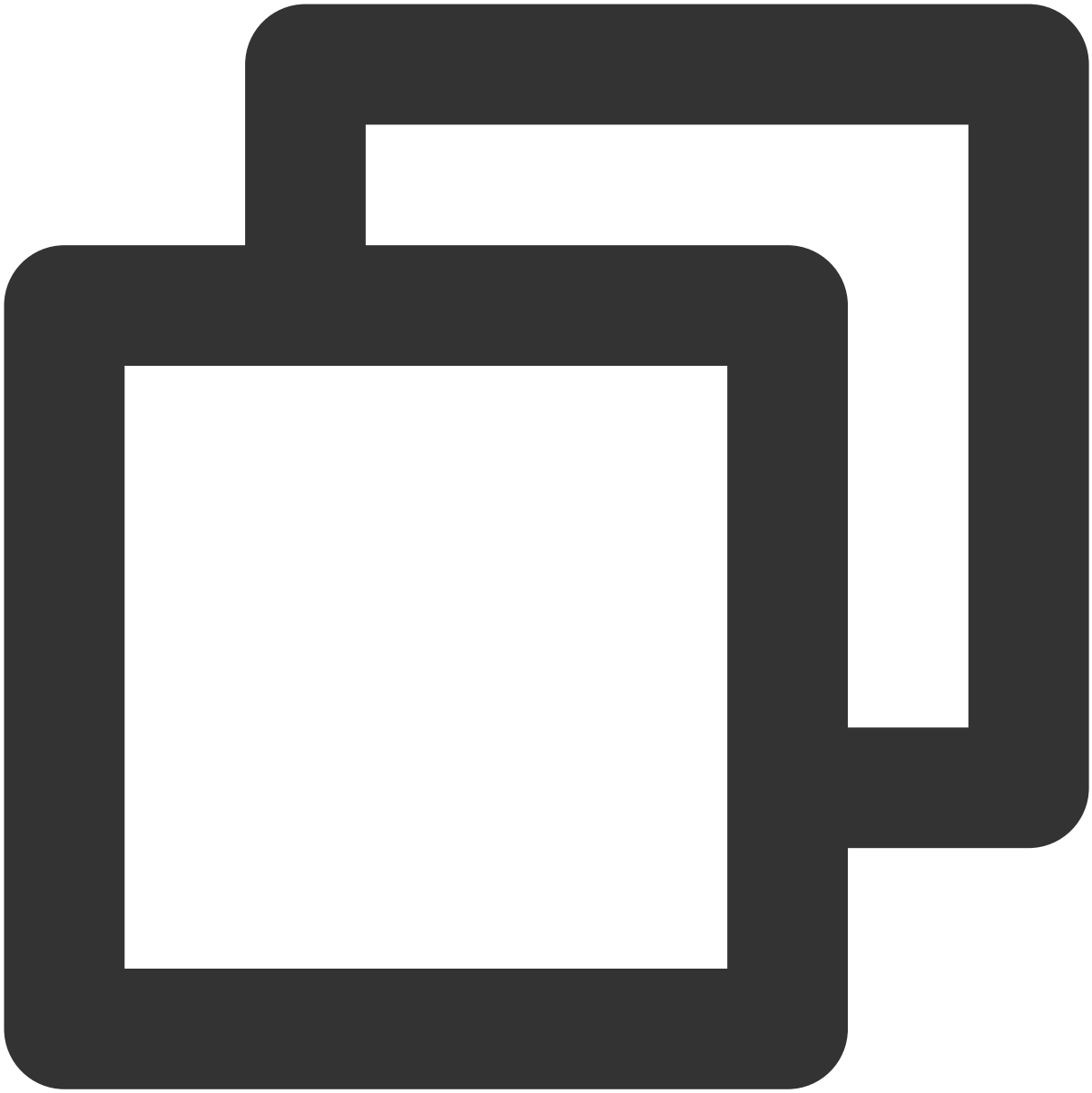
This method is used via FileSystemManager.unlinkSync(string filePath).

Feature Description: The synced version of FileSystemManager.unlink.

Parameter and Description: string filePath, the path of the file to be deleted.

Sample Code

©2013-2022 Tencent Cloud. All rights reserved.



```
const fs = wx.getFileSystemManager()
fs.unlink({
  filePath: `${wx.env.USER_DATA_PATH}/hello.txt`,
  success(res) {
    console.log(res)
  },
  fail(res) {
    console.error(res)
  }
})
```

©2013-2022 Tencent Cloud. All rights reserved.

```
// Synchronization interface
try {
  const res = fs.unlinkSync(`${wx.env.USER_DATA_PATH}/hello.txt`)
  console.log(res)
} catch(e) {
  console.error(e)
}
```

.unzip

This method is used via FileSystemManager.unzip(Object object).

Feature Description: Decompresses files.

Parameter and Description: Object

Attribute	Type	Default value	Required	Description
zipFilePath	string	-	Yes	Source file path, which can only be a zip compressed file.
targetPath	string	-	Yes	Path of destination directory.
success	function	-	No	Callback Function of Successful Interface Call
fail	function	-	No	Callback Function of Failing Interface Call
complete	function	-	No	Callback function executed upon the completion of the interface invocation (both successful and unsuccessful invocations)

Parameters for object.fail callback function: Object res

Attribute	Type	Description
errMsg	string	Error Message

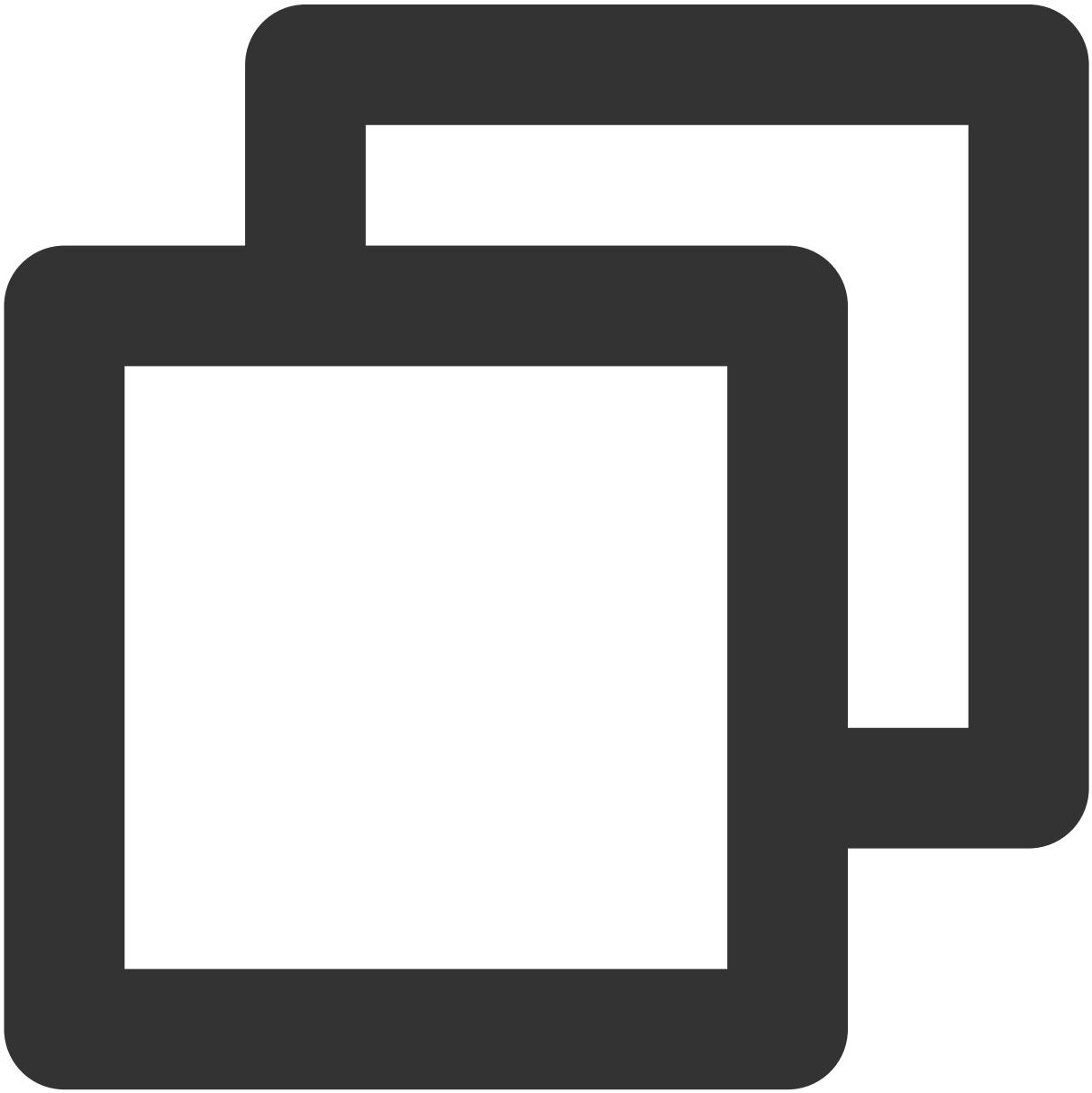
Valid values of res.errMsg

Value	Description
fail permission denied, unzip ${zipFilePath} -> ${destPath}	The specified destination file path does not have write permissions.
fail no such file or directory, unzip ${zipFilePath} ->	The source file does not exist, or the parent

©2013-2022 Tencent Cloud. All rights reserved.

${destPath}	directory of the destination file path does not exist.

Sample Code



```
const fs = wx.getFileSystemManager()
fs.unzip({
  zipFilePath: `${wx.env.USER_DATA_PATH}/example.zip`,
  targetPath: '${wx.env.USER_DATA_PATH}/example',
  success(res) {
    console.log(res)
```

©2013-2022 Tencent Cloud. All rights reserved.

```
  },
  fail(res) {
    console.error(res)
  }
})
```

.write

This method is used via FileSystemManager.write(Object object).

Feature Description: Writes in file.

Parameter and Description: Object

Attribute	Type	Default value	Required	Description
fd	string	-	Yes	The file descriptor. The fd is obtained through the FileSystemManager.open or FileSystemManager.openSync interface.
data	string/ArrayBuffer	-	Yes	The text or binary data to be written in.
offset	number	-	No	Only effective when the data type is ArrayBuffer, it determines the part to be written in the ArrayBuffer, that is, the index in the ArrayBuffer, with the default value being 0.
length	number	-	No	Only effective when the data type is ArrayBuffer, it specifies the number of bytes to be written, defaulting to the remaining bytes after offsetting the ArrayBuffer from 0 by the specified bytes at the offset value.
encoding	string	utf8	No	Specifies the character encoding for the file to be written.
position	number	-	No	Specifies the offset at the beginning of the file, that is, the position where the data is to be written. When position is not passed or a non-Number type value is passed, the data will be written at the current pointer location.
success	function	-	No	Callback Function of Successful Interface Call

©2013-2022 Tencent Cloud. All rights reserved.

fail	function	-	No	Callback Function of Failing Interface Call
complete	function	-	No	Callback function executed upon the completion of the interface invocation (both successful and unsuccessful invocations)

Valid values for encoding

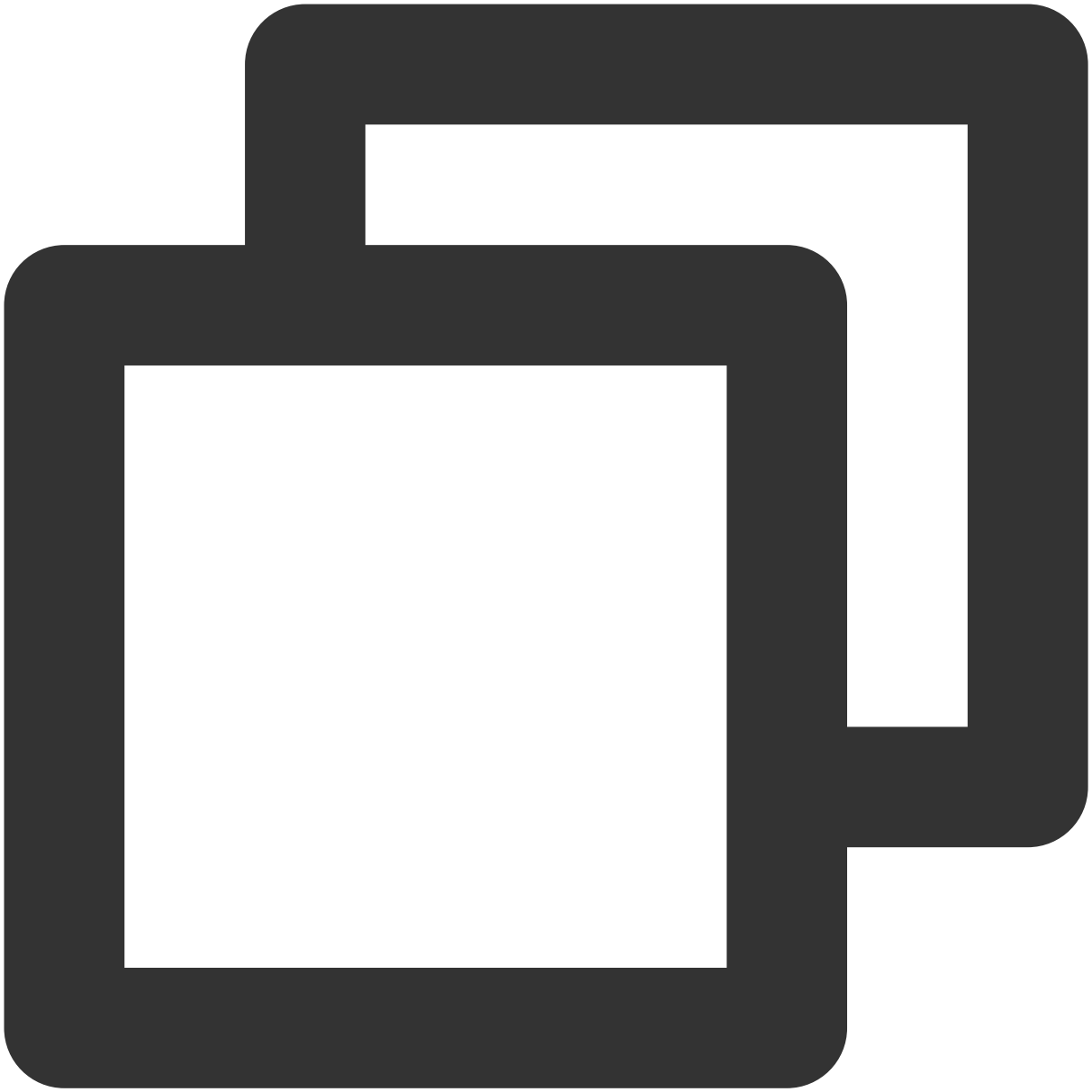
Value	Description
ascii	-
base64	-
binary	-
hex	-
ucs2/ucs-2/utf16le/utf-16le	Reads in little-endian order.
utf-8/utf8	-
latin1	-

Parameters for object.success callback function: Object res.

Attribute	Type	Description
bytesWritten	number	The actual number of bytes written into the file (note, the number of bytes written may not necessarily be the same as the number of characters in the written string).

Sample Code

©2013-2022 Tencent Cloud. All rights reserved.



```
const fs = wx.getFileSystemManager()
// Open the file
fs.open({
  filePath: ${wx.env.USER_DATA_PATH}/hello.txt,
  flag: 'a+',
      success(res) {
    // Write in files.
    fs.write({
      fd: res.fd,
      data: 'some text',
      success(res) {
```

©2013-2022 Tencent Cloud. All rights reserved.

```
console.log(res.bytesWritten)
  }
})
  }
})
```

.writeSync

This method is used via FileSystemManager.writeFile(Object object).

Feature Description: Synchronously writes in files.

Parameter and Description: Object

Attribute	Type	Default value	Required	Description
fd	string	-	Yes	The file descriptor. The fd is obtained through the FileSystemManager.open or FileSystemManager.openSync interface.
data	string/ArrayBuffer	-	Yes	The text or binary data to be written in.
offset	number	-	No	Only effective when the data type is ArrayBuffer, it determines the part to be written in the ArrayBuffer, that is, the index in the ArrayBuffer, with the default value being 0.
length	number	-	No	Only effective when the data type is ArrayBuffer, it specifies the number of bytes to be written, defaulting to the remaining bytes after offsetting the ArrayBuffer from 0 by the specified bytes at the offset value.
encoding	string	utf8	No	Specifies the character encoding for the file to be written.
position	number	-	No	Specifies the offset at the beginning of the file, that is, the position where the data is to be written. When position is not passed or a non-Number type value is passed, the data will be written at the current pointer location.
success	function	-	No	Callback Function of Successful Interface Call

©2013-2022 Tencent Cloud. All rights reserved.

fail	function	-	No	Callback Function of Failing Interface Call
complete	function	-	No	Callback function executed upon the completion of the interface invocation (both successful and unsuccessful invocations)

Valid values for encoding

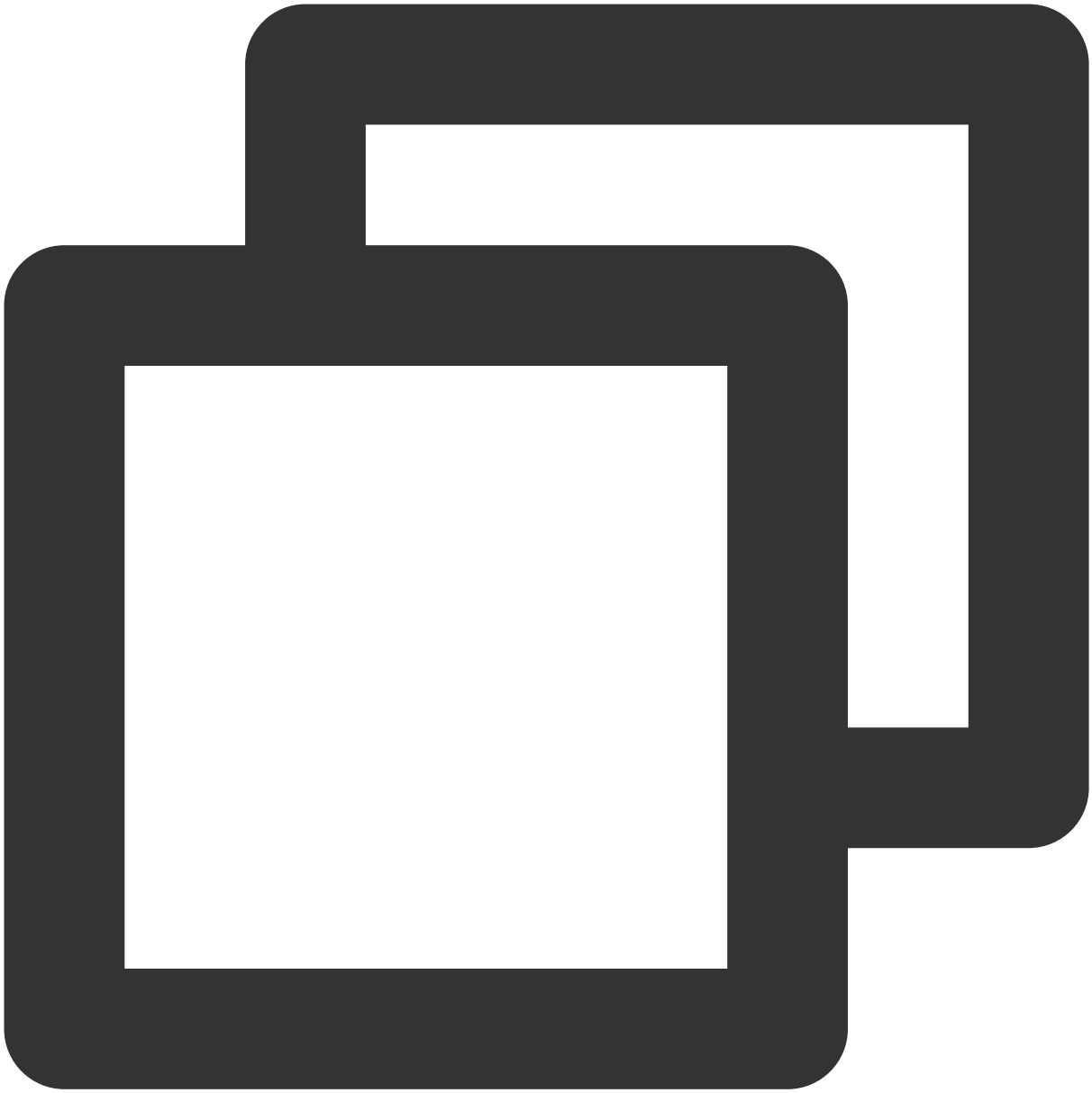
Value	Description
ascii	-
base64	-
binary	-
hex	-
ucs2/ucs-2/utf16le/utf-16le	Reads in little-endian order.
utf-8/utf8	-
latin1	-

Parameters for object.success callback function: Object res.

Attribute	Type	Description
bytesWritten	number	The actual number of bytes written into the file (note, the number of bytes written may not necessarily be the same as the number of characters in the written string).

Sample Code

©2013-2022 Tencent Cloud. All rights reserved.



```
const fs = wx.getFileSystemManager()
const fd = fs.openSync({
  filePath: ${wx.env.USER_DATA_PATH}/hello.txt,
  flag: 'a+'
})
const res = fs.writeSync({
  fd: fd,
  data: 'some text'
})
console.log(res.bytesWritten)
```

©2013-2022 Tencent Cloud. All rights reserved.

writeFile

This method is used via FileSystemManager.writeFile(Object object).

Feature Description: Writes files.

Parameter and Description: Object

Attribute	Type	Default value	Required	Description
filePath	string	-	Yes	Path of file to be written in.
data	string/ArrayBuffer	-	Yes	The text or binary data to be written in.
encoding	string	utf8	No	Specifies the character encoding for the file to be written.
success	function	-	No	Callback Function of Successful Interface Call
fail	function	-	No	Callback Function of Failing Interface Call
complete	function	-	No	Callback function executed upon the completion of the interface invocation (both successful and unsuccessful invocations)

Valid values for object.encoding

Value	Description
ascii	-
base64	-
binary	-
hex	-
ucs2/ucs-2/utf16le/utf-16le	Reads in little-endian order.
utf-8/utf8	-
latin1	-

Parameters for object.fail callback function: Object res

©2013-2022 Tencent Cloud. All rights reserved.

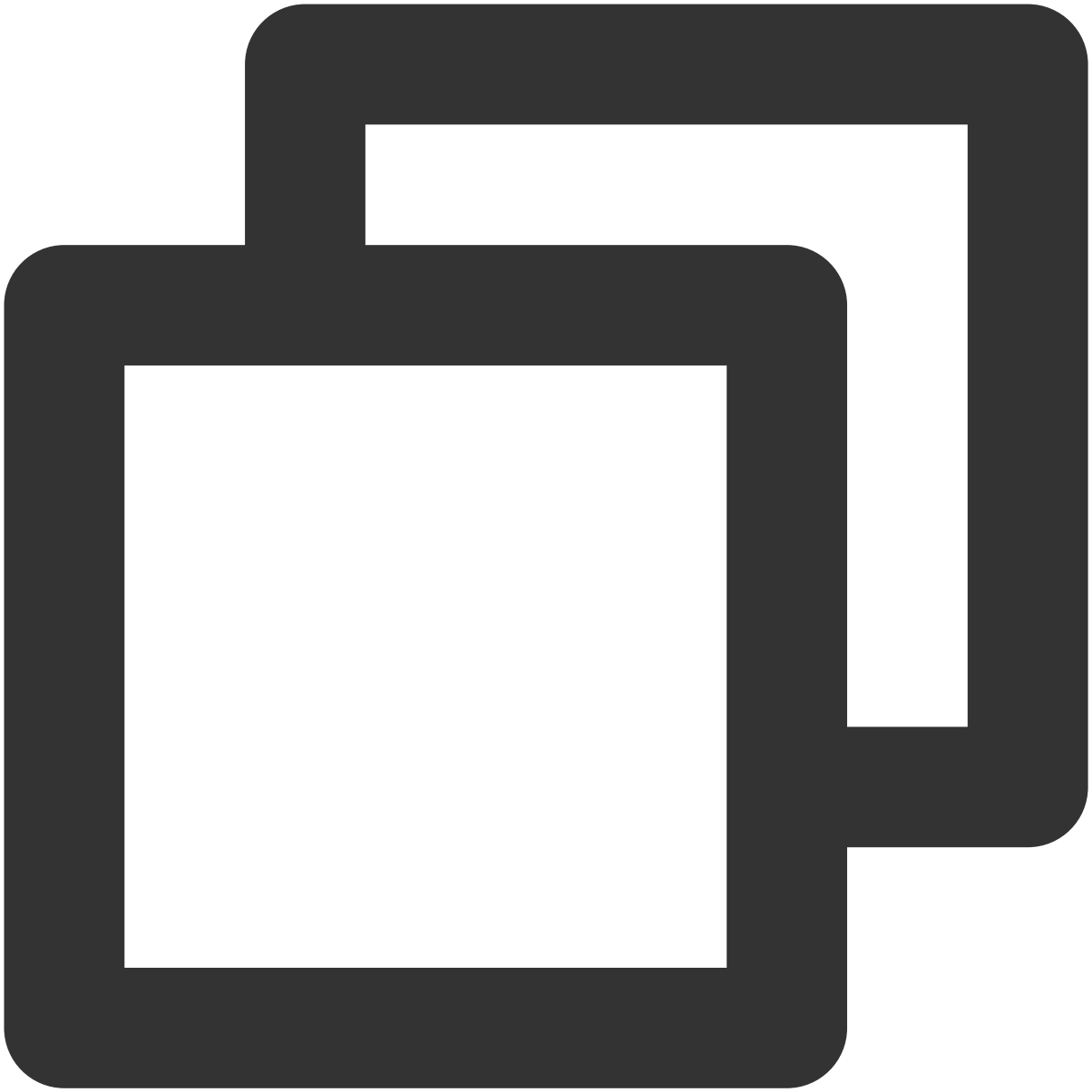
Attribute	Type	Description
errMsg	string	Error Message

Valid values of res.errMsg

Value	Description
fail no such file or directory, open ${filePath}	The directory of the specified filePath does not exist.
fail permission denied, open ${dirPath}	The specified filePath does not have write permissions.

Sample Code

©2013-2022 Tencent Cloud. All rights reserved.



```
const fs = wx.getFileSystemManager()
fs.writeFile({
  filePath: `${wx.env.USER_DATA_PATH}/hello.txt`,
  data: 'some text or arrayBuffer',
  encoding: 'utf8',
  success(res) {
    console.log(res)
  },
  fail(res) {
    console.error(res)
  }
```

©2013-2022 Tencent Cloud. All rights reserved.

```
})

// Synchronization interface
try {
  const res = fs.writeFileSync(
    `${wx.env.USER_DATA_PATH}/hello.txt`,
    'some text or arrayBuffer',
    'utf8'
  )
  console.log(res)
} catch(e) {
  console.error(e)
}
```

.writeFileSync

This method is used via FileSystemManager.writeFileSync(string filePath, string|ArrayBuffer data, string encoding).

Feature Description: The synced version of FileSystemManager.writeFile.

Parameter and Description:

string filePath: The path of file to be appended.

string|ArrayBuffer data: The text or binary data to be written in.

string encoding: Specifies the character encoding for writing to the file.

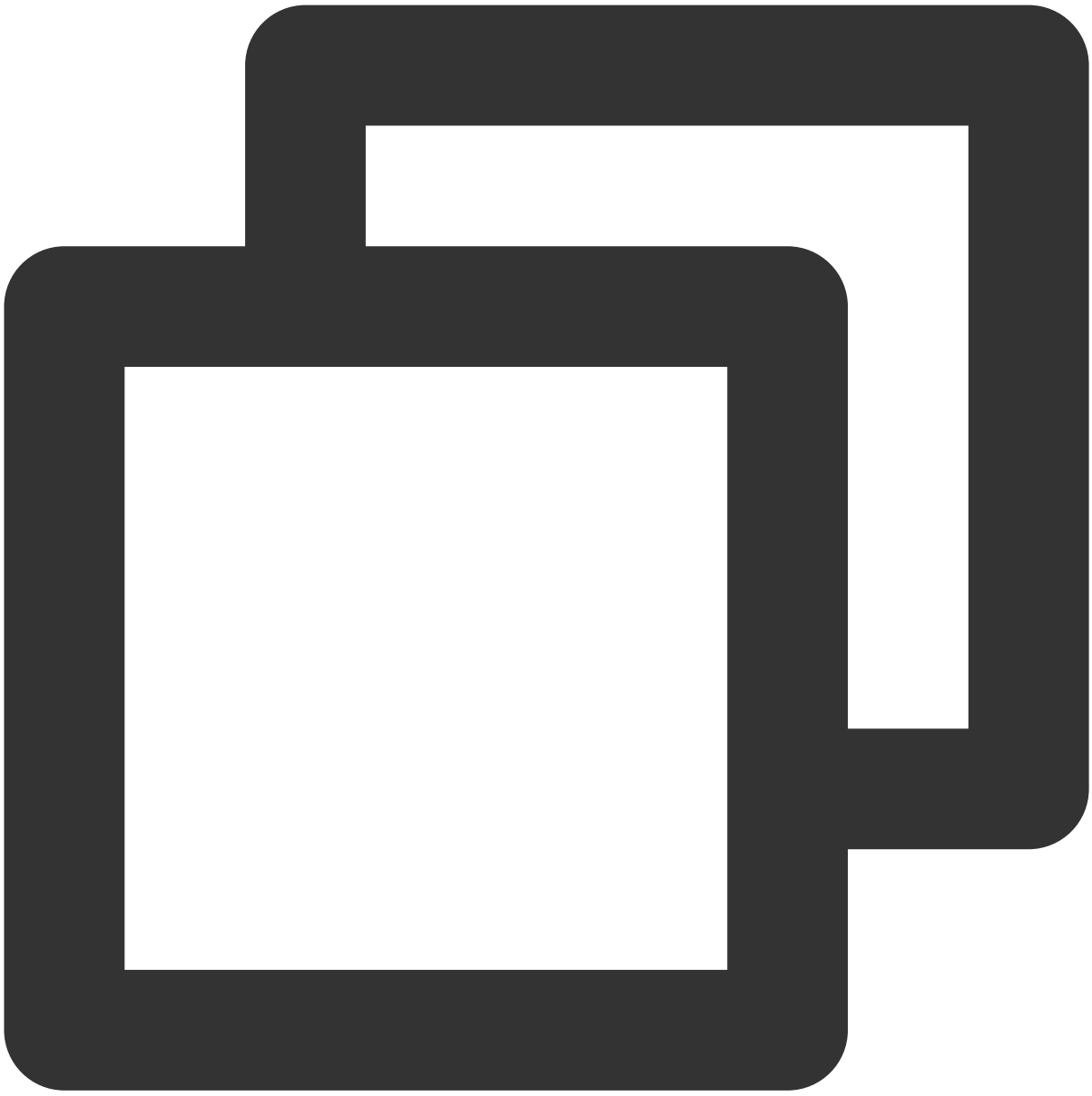
Valid Values for encoding

Value	Description
ascii	-
base64	Note that if you choose base64 encoding, you only need to pass the base64 content itself as data, not the Data URI prefix, otherwise you will get a fail base64 encode error. For example, pass aGVsbG8= instead of .
binary	-
hex	-
ucs2/ucs-2/utf16le/utf-16le	Reads in little-endian order.
utf-8/utf8	-

©2013-2022 Tencent Cloud. All rights reserved.

latin1	-

Sample Code



```
const fs = wx.getFileSystemManager()
fs.writeFile({
  filePath: `${wx.env.USER_DATA_PATH}/hello.txt`,
  data: 'some text or arrayBuffer',
  encoding: 'utf8',
  success(res) {
    console.log(res)
```

©2013-2022 Tencent Cloud. All rights reserved.

```
  },
  fail(res) {
    console.error(res)
  }
})

// Synchronization interface
try {
  const res = fs.writeFileSync(
    `${wx.env.USER_DATA_PATH}/hello.txt`,
    'some text or arrayBuffer',
    'utf8'
  )
  console.log(res)
} catch(e) {
  console.error(e)
}
```

Error Code

Caution:

Unless otherwise specified, the error codes adhere to the following table.

Error code	Error Message	Description
1300001	operation not permitted	Operation not permitted (for instance, filePath is expected to receive a file but a directory is actually passed in).
1300002	no such file or directory ${path}	The file/directory does not exist, or the parent directory of the target file path does not exist.
1300005	Input/output error	Input/output stream is unavailable.
1300009	bad file descriptor	Invalid file descriptor.
1300013	permission denied	Permission error. The file is read-only or write-only.
1300014	Path permission denied	The specified path does not have permissions.
1300020	not a directory	The specified dirPath is not a directory, a case commonly seen when the parent path of the designated write path is a file.
1300021	Is a directory	The specified path is a directory.
1300022	Invalid argument	Invalid parameters. Consider checking if length or offset has

©2013-2022 Tencent Cloud. All rights reserved.

		exceeded their limits.
1300036	File name too long	File name is too long.
1300066	directory not empty	The directory is not empty.
1300201	system error	The system interface call has failed.
1300202	the maximum size of the file storage limit is exceeded	Insufficient storage space, or the file size has exceeded the limit (maximum 100 M).
1300203	base64 encode error	Character encoding conversion has failed (for instance, base64 format error).
1300300	sdcard not mounted	Android SD card mounting has failed.
1300301	unable to open as fileType	Unable to open the file with the specified fileType.
1301000	permission denied, cannot access file path	No access permission to the target path (usr directory).
1301002	data to write is empty	The data written in is empty.
1301003	illegal operation on a directory	This operation cannot be performed on a directory (for example, the specified filePath is an existing directory).
1301004	illegal operation on a package directory	This operation cannot be performed on the code package directory.
1301005	file already exists ${dirPath}	A file or directory with the same name already exists.
1301006	value of length is out of range	The length passed in is invalid.
1301007	value of offset is out of range	The offset passed in is invalid.
1301009	value of position is out of range	The position value is out of limits.
1301100	store directory is empty	The "store" directory is empty.
1301102	unzip open file fail	Failed to open the compressed file.
1301103	unzip entry fail	De-compression of individual files failed.
1301104	unzip fail	De-compression failed.
1301111	brotli decompress fail	Brotli de-compression failed (for instance, the specified compression algorithm does not match the actual compression format of the file).

©2013-2022 Tencent Cloud. All rights reserved.

1301112	tempFilePath file not exist	The file is not found in the specified tempFilePath.
1302001	fail permission denied	The specified fd path does not have read/write permissions.
1302002	excced max concurrent fd limit	The number of file descriptors has reached its maximum limit.
1302003	invalid flag	Invalid flag.
1302004	permission denied when open using flag	Unable to open the file using the flag indicator.
1302005	array buffer does not exist	ArrayBuffer has not been passed in.
1302100	array buffer is readonly	The ArrayBuffer is read-only.

Stats

An object that describes the status of the file.

Attribute

Number mode: The type and access permissions of the file, corresponding to POSIX stat.st_mode.
Number size: The size of the file, measured in bytes (B), corresponding to POSIX stat.st_size.
Number lastAccessedTime: The most recent time the file was accessed or executed, represented as a UNIX timestamp, corresponding to POSIX stat.st_atime.
Number lastModifiedTime: The most recent time the file was modified, represented as a UNIX timestamp, corresponding to POSIX stat.st_mtime.

Method Set

.isDirectory

This method is used via boolean Stats.isDirectory().

Feature Description: Determines whether the current file is a directory.
Return Value: Boolean, indicating whether the current file is a directory.

.isFile

This method is used via boolean Stats.isFile().

©2013-2022 Tencent Cloud. All rights reserved.

Feature Description: Determines whether the current file is a regular file.

Return Value: Boolean, indicating whether the current file is a regular file.

ReadResult

Feature Description: File reading results. Returned via the FileSystemManager.readSync interface.

Attributes:

Number bytesRead, the actual number of bytes read.

ArrayBuffer, the object of the buffer area being written, that is, the arrayBuffer of the interface parameter.

WriteResult

Feature Description: File writing results. Returned via the FileSystemManager.writeSync interface.

Attributes: Number bytesWritten, the actual number of bytes written into the file (note, the number of bytes written may not necessarily be the same as the number of characters in the written string).

©2013-2022 Tencent Cloud. All rights reserved.

Device
Contact person

Last updated：2024-07-16 17:43:56

chooseContact

This API is used via wx.chooseContact(Object object).

Feature Description: Initiates the mobile phone's contact list, allowing the selection of a contact.

Parameter and Description: Object.

Attribute	Type	Default value	Required	Description
success	function	-	No	Callback Function of Successful Interface Call
fail	function	-	No	Callback Function of Failing Interface Call
complete	function	-	No	Callback function executed upon the completion of the interface invocation (both successful and unsuccessful invocations)

object.success callback function: Object.

Attribute	Type	Description
phoneNumber	string	Mobile number
displayName	string	Contact name
phoneNumberList	string	All mobile numbers of the selected contact (some Android systems only allow selecting contacts, not specific phone numbers)

addPhoneContact

This API is used via wx.addPhoneContact(Object object).

©2013-2022 Tencent Cloud. All rights reserved.

Feature Description: Adds mobile phone contacts. Users can choose to write this form into the mobile phone system's contact list by either **adding a new contact** or **adding to an existing contact**.
Parameter and Description: Object.

Attribute	Type	Default value	Required	Description
firstName	string	-	Yes	Name
photoFilePath	string	-	No	Local file path of avatar
nickName	string	-	No	Nickname
lastName	string	-	No	Surname
middleName	string	-	No	Middle Name
remark	string	-	No	Remarks
mobilePhoneNumber	string	-	No	Mobile number
weChatNumber	string	-	No	WeChat ID
addressCountry	string	-	No	Country of address
addressState	string	-	No	Province of address
addressCity	string	-	No	City of address
addressStreet	string	-	No	Sub-district of address
addressPostalCode	string	-	No	Postcode
organization	string	-	No	Company
title	string	-	No	Position
workFaxNumber	string	-	No	Office fax number
workPhoneNumber	string	-	No	Office telephone number
hostNumber	string	-	No	Company phone number
email	string	-	No	Email Address
url	string	-	No	Website
workAddressCountry	string	-	No	Country of office address

©2013-2022 Tencent Cloud. All rights reserved.

workAddressState	string	-	No	Province of office address
workAddressCity	string	-	No	City of office address
workAddressStreet	string	-	No	Sub-district of office address
workAddressPostalCode	string	-	No	Postcode of office address
homeFaxNumber	string	-	No	Residential Fax
homePhoneNumber	string	-	No	Residential phone number
homeAddressCountry	string	-	No	Country of Residential Address
homeAddressState	string	-	No	Province of Residential Address
homeAddressCity	string	-	No	City of Residential Address
homeAddressStreet	string	-	No	Sub-district of residential address
homeAddressPostalCode	string	-	No	Postcode of residential address
success	function	-	No	Callback Function of Successful Interface Call
fail	function	-	No	Callback Function of Failing Interface Call
complete	function	-	No	Callback function executed upon the completion of the interface invocation (both successful and unsuccessful invocations)

©2013-2022 Tencent Cloud. All rights reserved.

Vibration

Last updated：2024-07-16 17:44:57

vibranteShort

This API is used via wx.vibrateShort(Object object).

Feature Description: Triggers a brief vibration (15 ms) on the mobile device. It is only effective on iPhone 7/7 Plus and newer models, as well as on Android devices.

Parameter and Description: Object.

Attribute	Type	Default value	Required	Description
success	function	-	No	Callback Function of Successful Interface Call
fail	function	-	No	Callback Function of Failing Interface Call
complete	function	-	No	Callback function executed upon the completion of the interface invocation (both successful and unsuccessful invocations)

vibranteLong

This API is used via wx.vibrateLong(Object object).

Feature Description: Triggers a prolonged vibration on the mobile phone (400 ms).

Parameter and Description: Object.

Attribute	Type	Default value	Required	Description
success	function	-	No	Callback Function of Successful Interface Call
fail	function	-	No	Callback Function of Failing Interface Call
complete	function	-	No	Callback function executed upon the completion of the

©2013-2022 Tencent Cloud. All rights reserved.

				interface invocation (both successful and unsuccessful invocations)

©2013-2022 Tencent Cloud. All rights reserved.

Battery Level

Last updated：2024-07-16 17:45:18

getBatteryInfoSync

The utilization method for this API is Object wx.getBatteryInfoSync().

Function Description: The synchronous version of wx.getBatteryInfo.

Return Value: Object res.

Attributes	Types	Note
level	number	Device battery level, ranging from 1 - 100.
isCharging	boolean	Is the device currently charging?

getBatteryInfo

The method for utilizing this API is wx.getBatteryInfo(Object object).

Function Description: This function retrieves the device's battery level. The synchronous API wx.getBatteryInfoSync is not available on iOS.

Parameters and Description: Object object.

Attributes	Types	Default value	Required	Note
success	function	-	Not required	Callback function for successful interface invocation
fail	function	-	Not required	Callback function for unsuccessful interface invocation
complete	function	-	Not required	Callback function executed upon the completion of the interface invocation (both successful and unsuccessful invocations)

©2013-2022 Tencent Cloud. All rights reserved.

object.success Callback Function Parameters: Object res.

Attributes	Types	Note
level	number	Device battery level, ranging from 1 - 100.
isCharging	boolean	Is the device currently charging?

©2013-2022 Tencent Cloud. All rights reserved.

Screen

Last updated：2024-07-16 17:45:38

setVisualEffectOnCapture

This API is used via wx.setVisualEffectOnCapture(Object object).

Feature Description: Sets the screen display performance during screenshot/screen recording, which is only supported for calls on the Android platform.

Parameter and Description: Object.

Attribute	Type	Default value	Required	Description
visualEffect	string	none	No	The display performance during screenshot/screen recording only supports 'none' or 'hidden'. If 'hidden' is passed in, it indicates that the screen will be hidden during screenshot/screen recording.
success	function	-	No	Callback Function of Successful Interface Call
fail	function	-	No	Callback Function of Failing Interface Call
complete	function	-	No	Callback function executed upon the completion of the interface invocation (both successful and unsuccessful invocations)

setScreenBrightness

This API is used via wx.setScreenBrightness(Object object).

Feature Description: Adjusts the screen brightness.

Parameter and Description: Object.

Attribute	Type	Default value	Required	Description

©2013-2022 Tencent Cloud. All rights reserved.

value	number	-	Yes	Screen brightness value, ranging from 0 to 1, where 0 is the darkest and 1 is the brightest. On the Android platform, a special value of -1 is supported, indicating that the screen brightness will adjust according to system changes.
success	function	-	No	Callback Function of Successful Interface Call
fail	function	-	No	Callback Function of Failing Interface Call
complete	function	-	No	Callback function executed upon the completion of the interface invocation (both successful and unsuccessful invocations)

setKeepScreenOn

This API is used via wx.setKeepScreenOn(Object object).

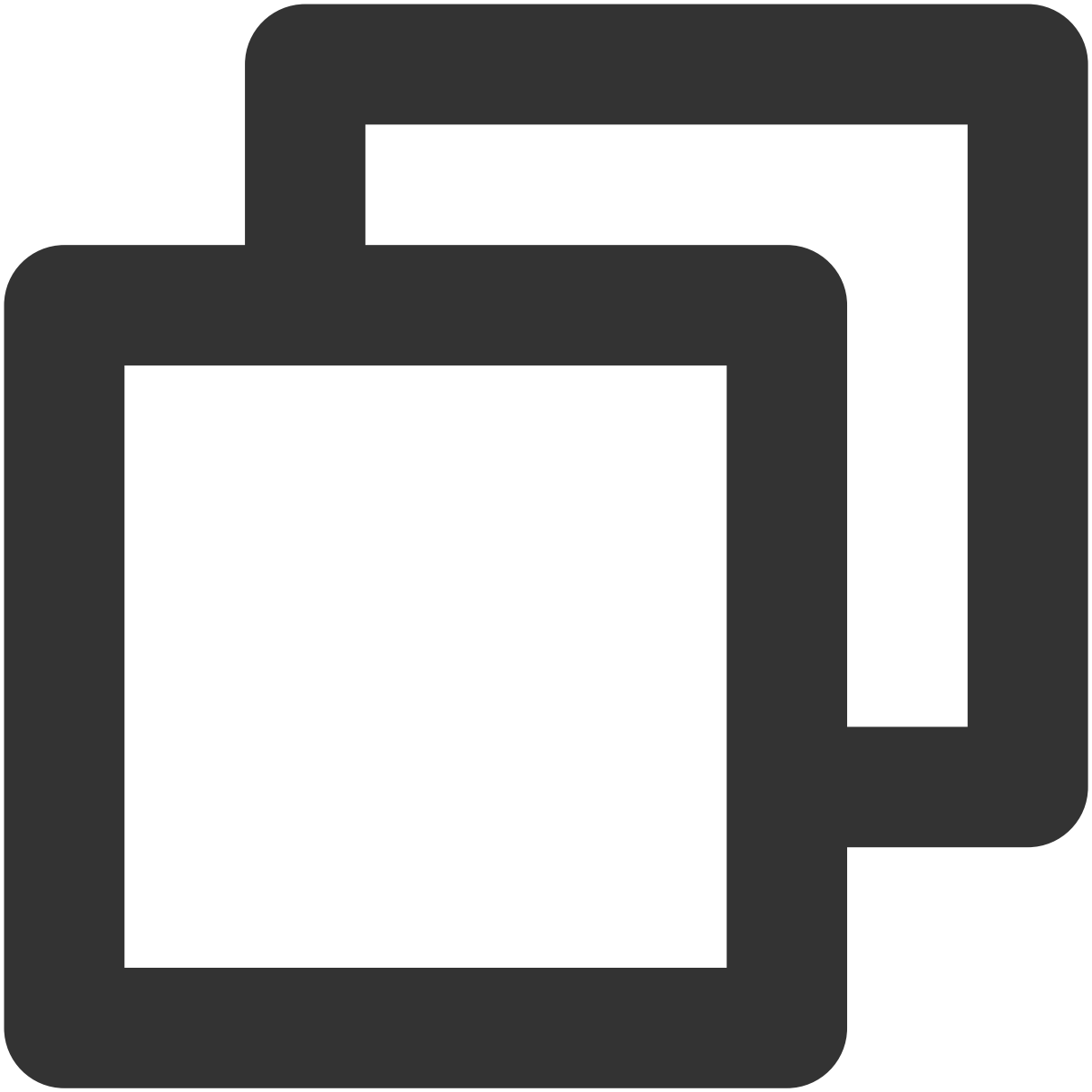
Feature Description: Sets whether to maintain a constant brightness. This setting is only effective within the current mini program and will be invalidated upon exiting.

Parameter and Description: Object.

Attribute	Type	Default value	Required	Description
keepScreenOn	boolean	-	Yes	Whether to maintain constant screen brightness
success	function	-	No	Callback Function of Successful Interface Call
fail	function	-	No	Callback Function of Failing Interface Call
complete	function	-	No	Callback function executed upon the completion of the interface invocation (both successful and unsuccessful invocations)

Sample Code

©2013-2022 Tencent Cloud. All rights reserved.



```
wx.setKeepScreenOn({
  keepScreenOn: true
})
```

onUserCaptureScreen

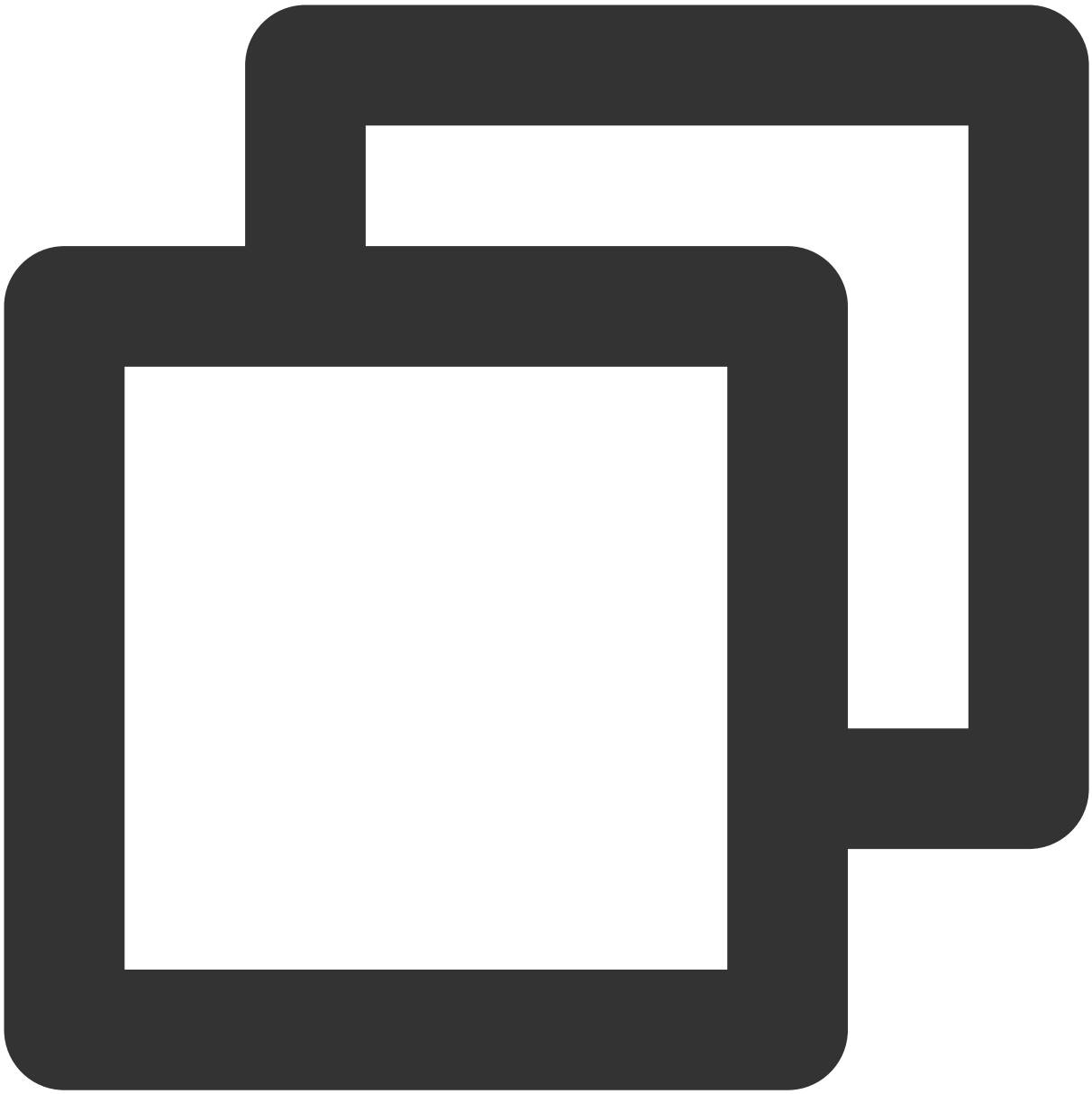
This API is used via wx.onUserCaptureScreen(function listener).

©2013-2022 Tencent Cloud. All rights reserved.

Feature Description: Monitors user-initiated screenshot events. This is triggered when the user uses the system screenshot key, and only one listener can be registered.

Parameter and Description: function listener, the monitoring function for user-initiated screenshot events.

Sample Code



```
wx.onUserCaptureScreen(function (res) {
  console.log('User has taken a screenshot')
})
```

©2013-2022 Tencent Cloud. All rights reserved.

onScreenRecordingStateChanged

This API is used via wx.onScreenRecordingStateChanged(function listener).

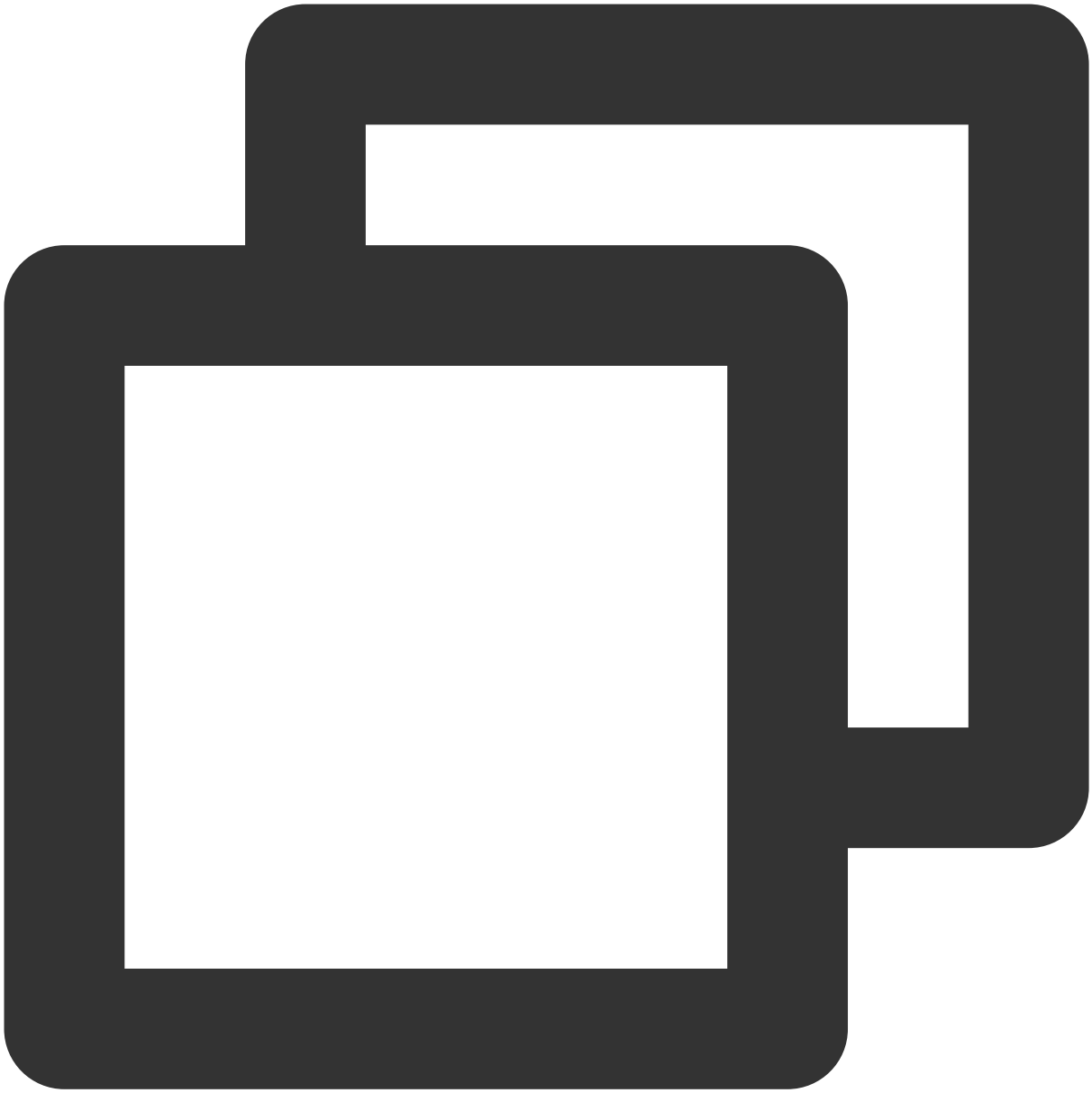
Feature Description: Monitors user-initiated screen recording events.

Parameter and Description: function listener, the monitoring function for user-initiated screen recording events.

Attribute	Type	Description
state	string	Screen recording status, whose valid values include: start: Start screen recording stop: Stop screen recording

Sample Code

©2013-2022 Tencent Cloud. All rights reserved.



```
// Monitor user-initiated screen recording events.
const handler = function (res) {
  console.log(res.state)
}
wx.onScreenRecordingStateChanged(handler)

// Cancel the monitoring of user-initiated screen recording events.
wx.offScreenRecordingStateChanged(handler)
```

©2013-2022 Tencent Cloud. All rights reserved.

offUserCaptureScreen

This API is used via wx.offUserCaptureScreen(function callback).

Feature Description: Cancels monitoring of user-initiated screenshot events.

Parameter and Description: function callback, the callback function of "user-initiated screenshot events".

offScreenRecordingStateChanged

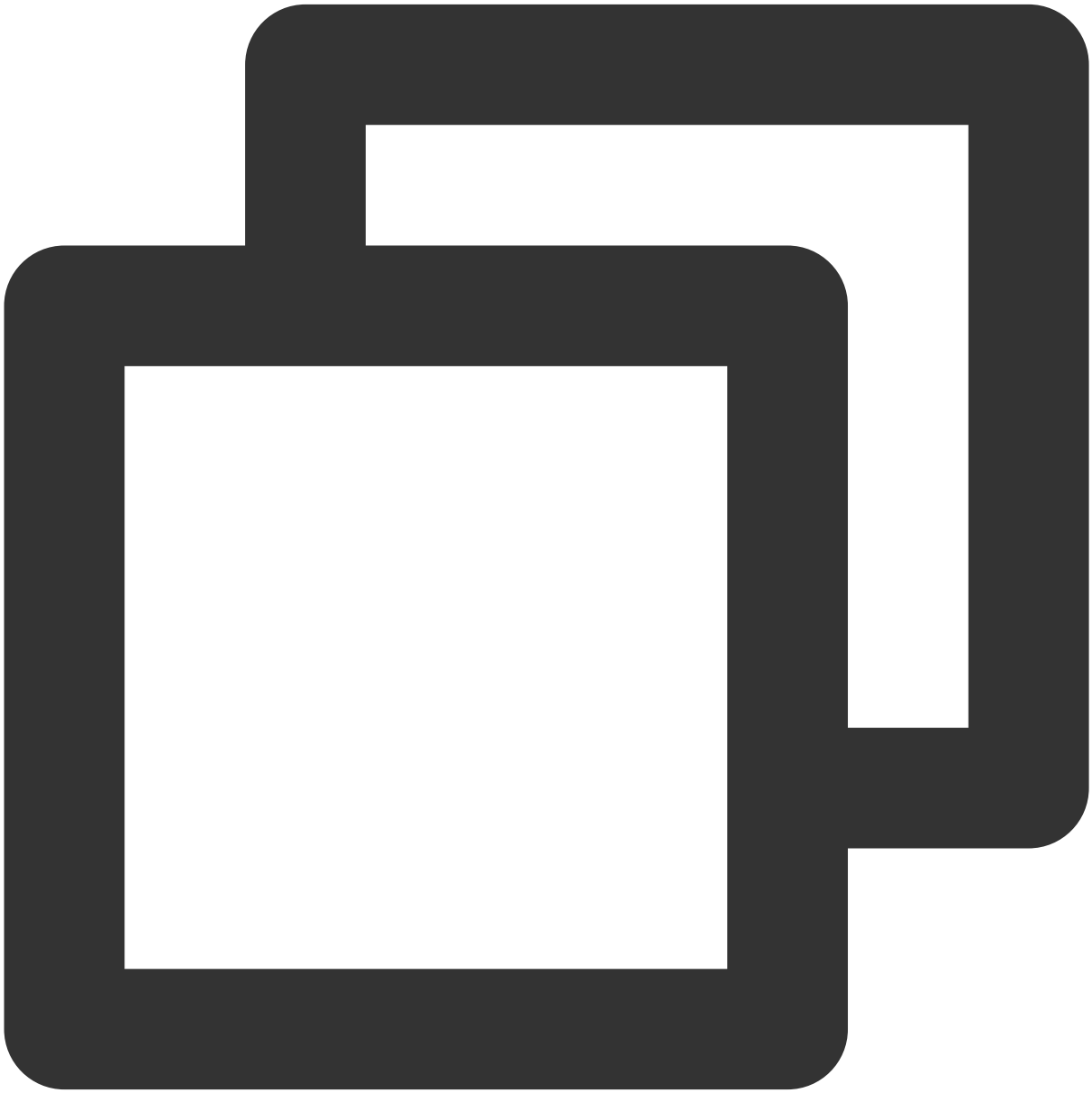
This API is used via wx.offScreenRecordingStateChanged(function listener).

Feature Description: Removes the listener function for user-initiated screenshot events.

Parameter and Description: function listener, the listener function passed in by onScreenRecordingStateChanged. If this parameter is not passed in, all listener functions will be removed.

Sample Code

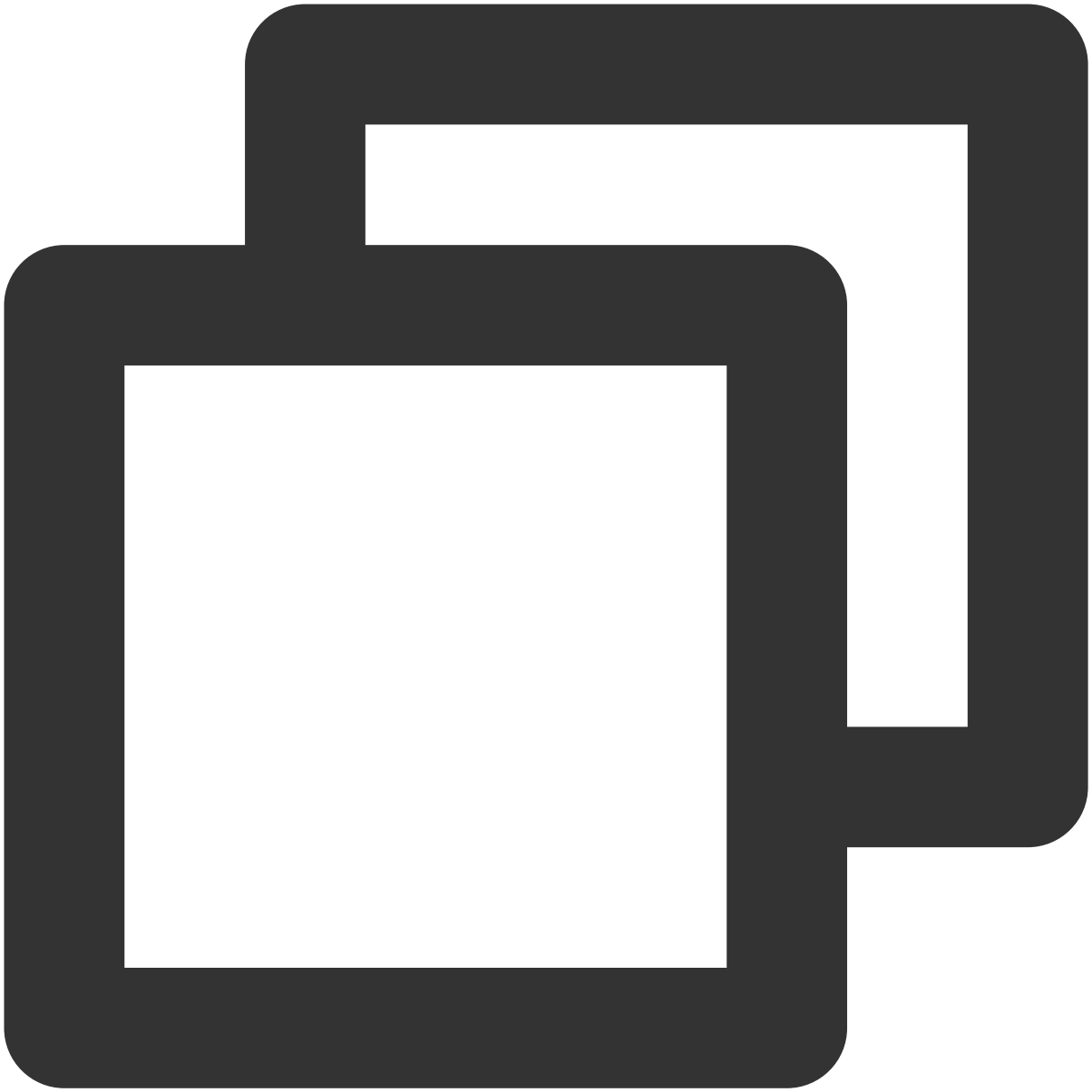
©2013-2022 Tencent Cloud. All rights reserved.



```
// Monitor user-initiated screen recording events.
const handler = function (res) {
  console.log(res.state)
}
wx.onScreenRecordingStateChanged(handler)

// Cancel the monitoring of user-initiated screen recording events.
wx.offScreenRecordingStateChanged(handler)
```

©2013-2022 Tencent Cloud. All rights reserved.



```
const listener = function (res) { console.log(res) }

wx.onScreenRecordingStateChanged(listener)
wx.offScreenRecordingStateChanged(listener) // The same function object as the list
```

getScreenRecordingState

©2013-2022 Tencent Cloud. All rights reserved.

This API is used via wx.getScreenRecordingState(Object object).

Feature Description: Inquires the user is currently recording the screen.

Parameter and Description: Object.

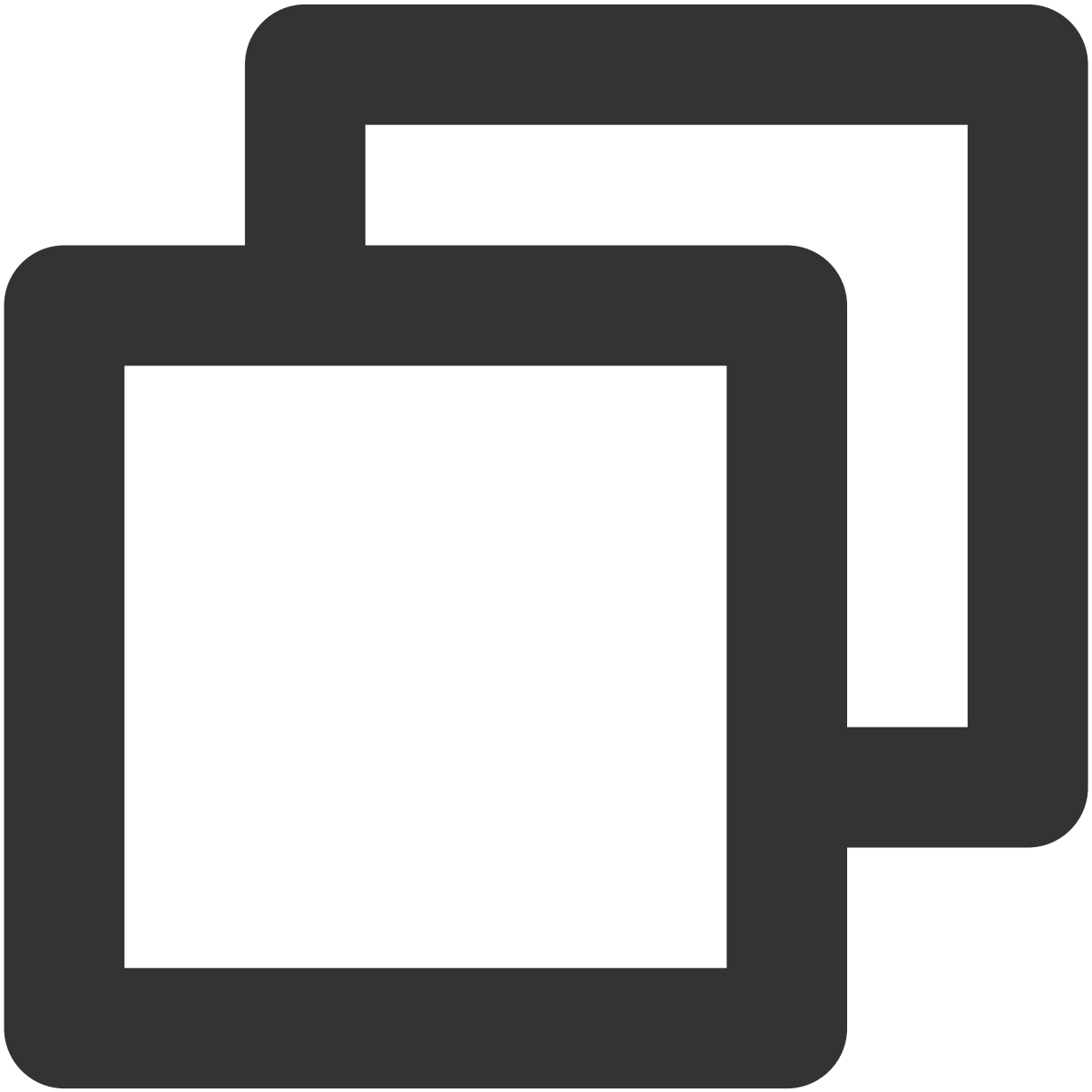
Attribute	Type	Default value	Required	Description
success	function	-	No	Callback Function of Successful Interface Call
fail	function	-	No	Callback Function of Failing Interface Call
complete	function	-	No	Callback function executed upon the completion of the interface invocation (both successful and unsuccessful invocations)

Parameters for object.success callback function: Object res.

Attribute	Type	Description
state	string	Screen recording status, whose valid values include: on: Enable off: Disable

Sample Code

©2013-2022 Tencent Cloud. All rights reserved.



```
wx.getScreenRecordingState({
  success: function (res) {
    console.log(res.state)
  },
})
```

getScreenBrightness

©2013-2022 Tencent Cloud. All rights reserved.

This API is used via wx.getScreenBrightness(Object object).

Note:

If the automatic brightness adjustment feature is enabled in the Android system settings, the screen brightness will adjust automatically according to the light. This interface can only get the value prior to the automatic brightness adjustment, not the real-time brightness value.

Feature Description: Gets screen brightness.

Parameter and Description: Object.

Attribute	Type	Default value	Required	Description
success	function	-	No	Callback Function of Successful Interface Call
fail	function	-	No	Callback Function of Failing Interface Call
complete	function	-	No	Callback function executed upon the completion of the interface invocation (both successful and unsuccessful invocations)

Parameters for object.success callback function: Object.

Attribute	Type	Description
value	number	Screen brightness value, ranging from 0 to 1, where 0 is the darkest and 1 is the brightest.

©2013-2022 Tencent Cloud. All rights reserved.

Memory

Last updated：2024-07-16 17:46:02

onMemoryWarning

This API is used via wx.onMemoryWarning(function listener).

Feature Description: Monitors low memory warning events. When iOS/Android issues a memory warning to the mini program process, these events are triggered. The triggering of such events does not imply that the mini program process has been cleared. In most cases, it is merely a warning. Developers can recycle some unnecessary resources to avoid further exacerbating memory tension after receiving the notification.
Parameter and Description: function listener, the monitoring function for low memory warning events.

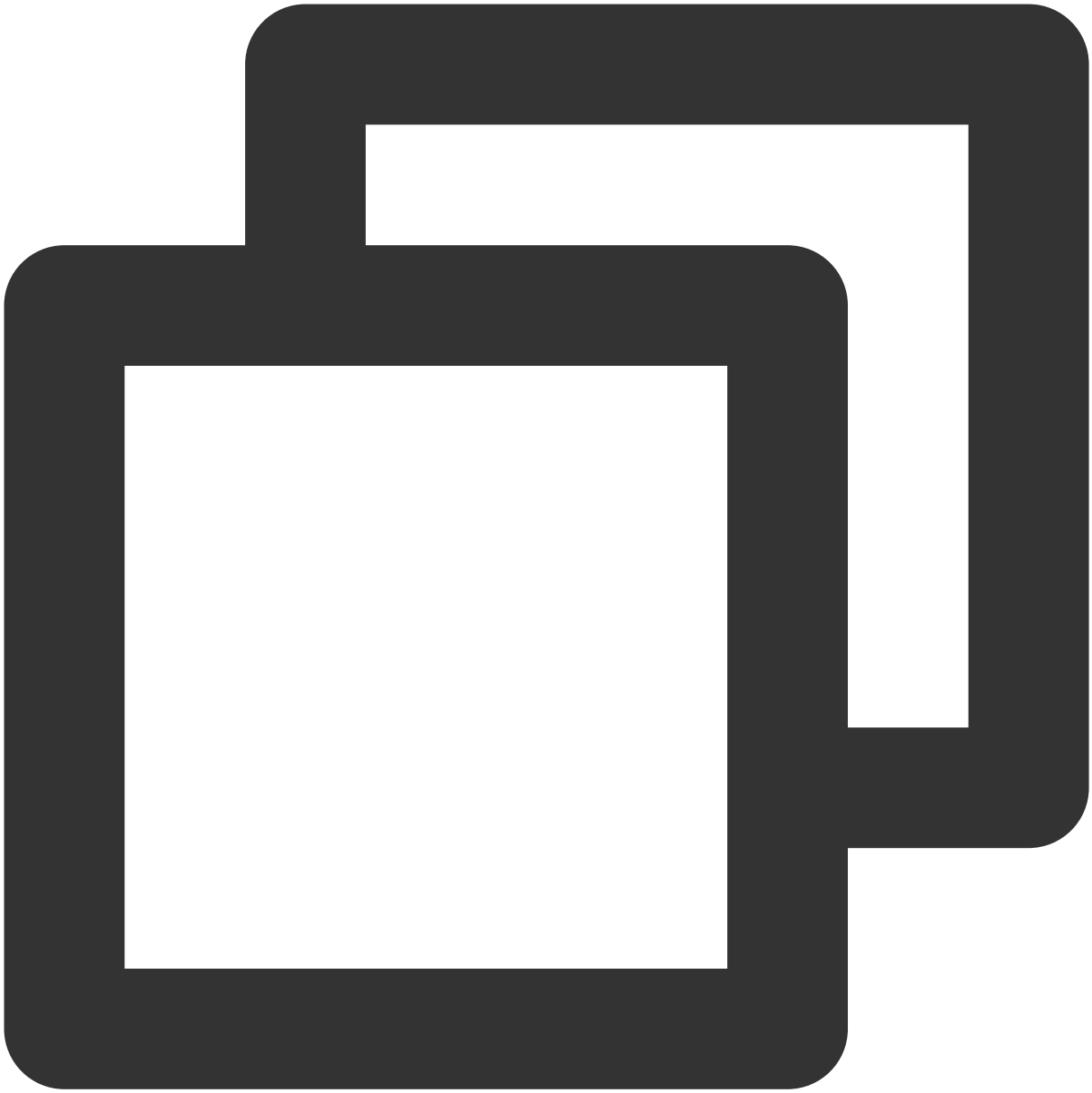
Attribute	Type	Description
level	number	Memory warning level, only available on Android, corresponds to system macro definition.

Valid Values for Level

Valid Values	Description
5	TRIM_MEMORY_RUNNING_MODERATE
10	TRIM_MEMORY_RUNNING_LOW
15	TRIM_MEMORY_RUNNING_CRITICAL

Sample Code

©2013-2022 Tencent Cloud. All rights reserved.



```
wx.onMemoryWarning(function () {
  console.log('onMemoryWarningReceive')
})
```

offMemoryWarning

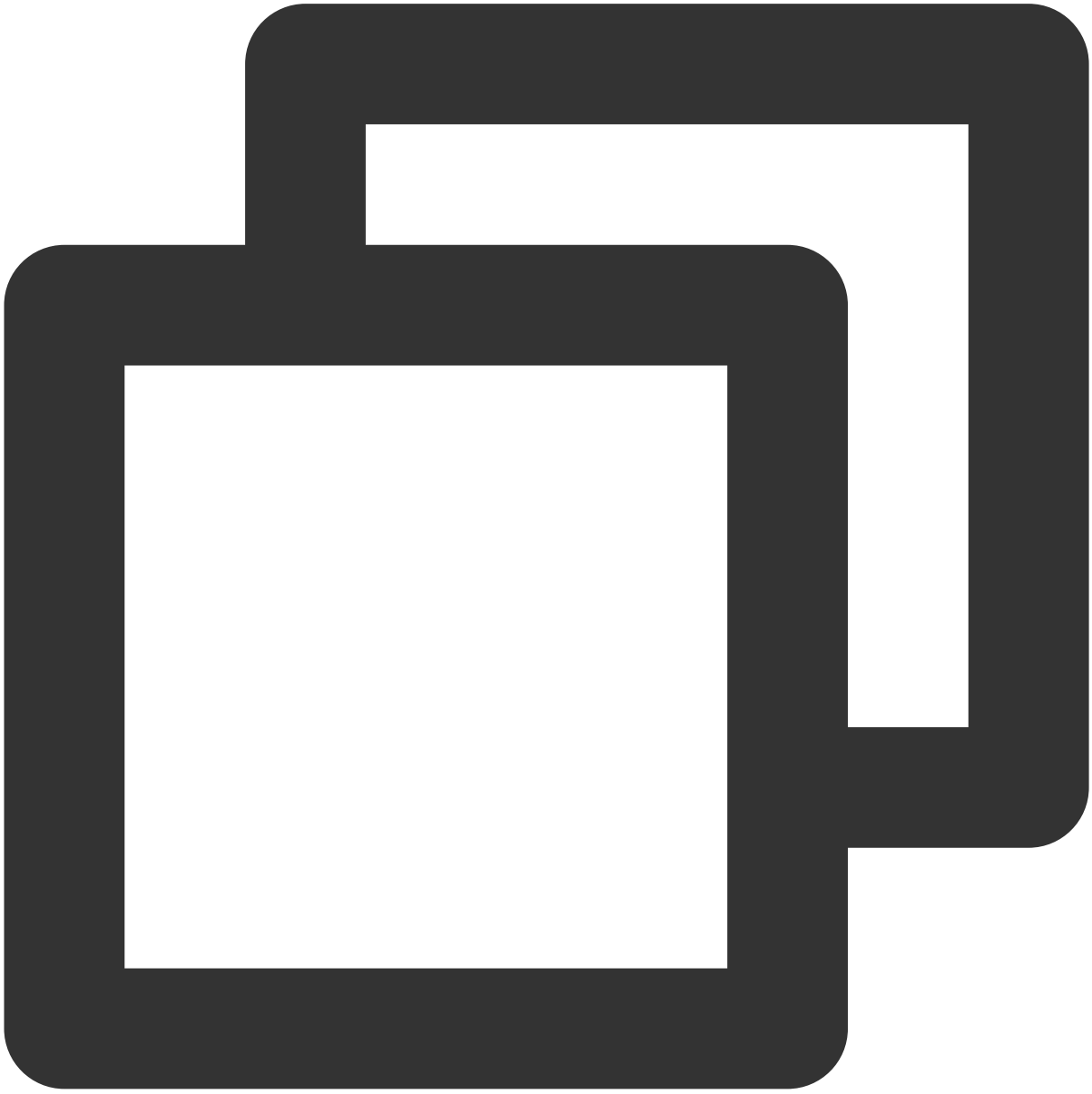
This API is used via wx.offMemoryWarning(function listener).

©2013-2022 Tencent Cloud. All rights reserved.

Feature Description: Removes the listener function for low memory warning events.

Parameters and Description: function listener, the listener function passed in by onMemoryWarning. If this parameter is not passed in, all listener functions will be removed.

Sample Code



```
const listener = function (res) { console.log(res) }

wx.onMemoryWarning(listener)
wx.offMemoryWarning(listener) // The same function object as the listener must be p
```

©2013-2022 Tencent Cloud. All rights reserved.

©2013-2022 Tencent Cloud. All rights reserved.

Device Orientation

Last updated：2024-07-16 17:46:23

stopDeviceMotionListening

The method of utilizing this API is wx.stopDeviceMotionListening(Object object).

Function Description: Cease monitoring alterations in device orientation.

Parameters and Description: Object object.

Attributes	Types	Default value	Required	Note
success	function	-	Not required	Callback function for successful interface invocation
fail	function	-	Not required	Callback function for unsuccessful interface invocation
complete	function	-	Not required	Callback function executed upon the completion of the interface invocation (both successful and unsuccessful invocations)

startDeviceMotionListening

The method for utilizing this API is wx.startDeviceMotionListening(Object object)

Function Description: Initiates monitoring of device orientation changes.

Parameters and Description: Object object.

Attributes	Types	Default value	Required	Note
interval	string	normal	Not required	The execution frequency of the callback function monitoring device orientation changes, valid values are: game: Suitable for updating the callback frame rate of the game, approximately every 20ms.

©2013-2022 Tencent Cloud. All rights reserved.

				ui: Suitable for updating the callback frame rate of the UI, approximately every 60ms. normal: Regular callback frequency, approximately every 200ms.
success	function	-	Not required	Callback function for successful interface invocation
fail	function	-	Not required	Callback function for unsuccessful interface invocation
complete	function	-	Not required	Callback function executed upon the completion of the interface invocation (both successful and unsuccessful invocations)

onDeviceMotionChange

The usage method for this API is wx.onDeviceMotionChange(function listener)

Function Description: Monitors device orientation change events. The frequency depends on the interval parameter of wx.startDeviceMotionListening(). You can use wx.stopDeviceMotionListening() to stop monitoring.

Parameters and Description: function listener, the monitoring function for device orientation change events.

Attributes	Types	Note
alpha	number	When the X/Y coordinates of the mobile device coincide with the X/Y coordinates of the Earth, the angle of rotation around the Z-axis is alpha, with a range of [0, 2*PI). Counterclockwise rotation is positive.
beta	number	When the Y/Z coordinates of the mobile device coincide with the Y/Z coordinates of the Earth, the angle of rotation around the X-axis is beta, with a range of [-1*PI, PI). Rotation with the top facing the Earth's surface is positive. It may also be positive when facing the user.
gamma	number	When the X/Z coordinates of the mobile device coincide with the X/Z coordinates of the Earth, the angle of rotation around the Y-axis is gamma, with a range of [-1*PI/2, PI/2). Rotation with the right side facing the Earth's surface is positive.

offDeviceMotionChange

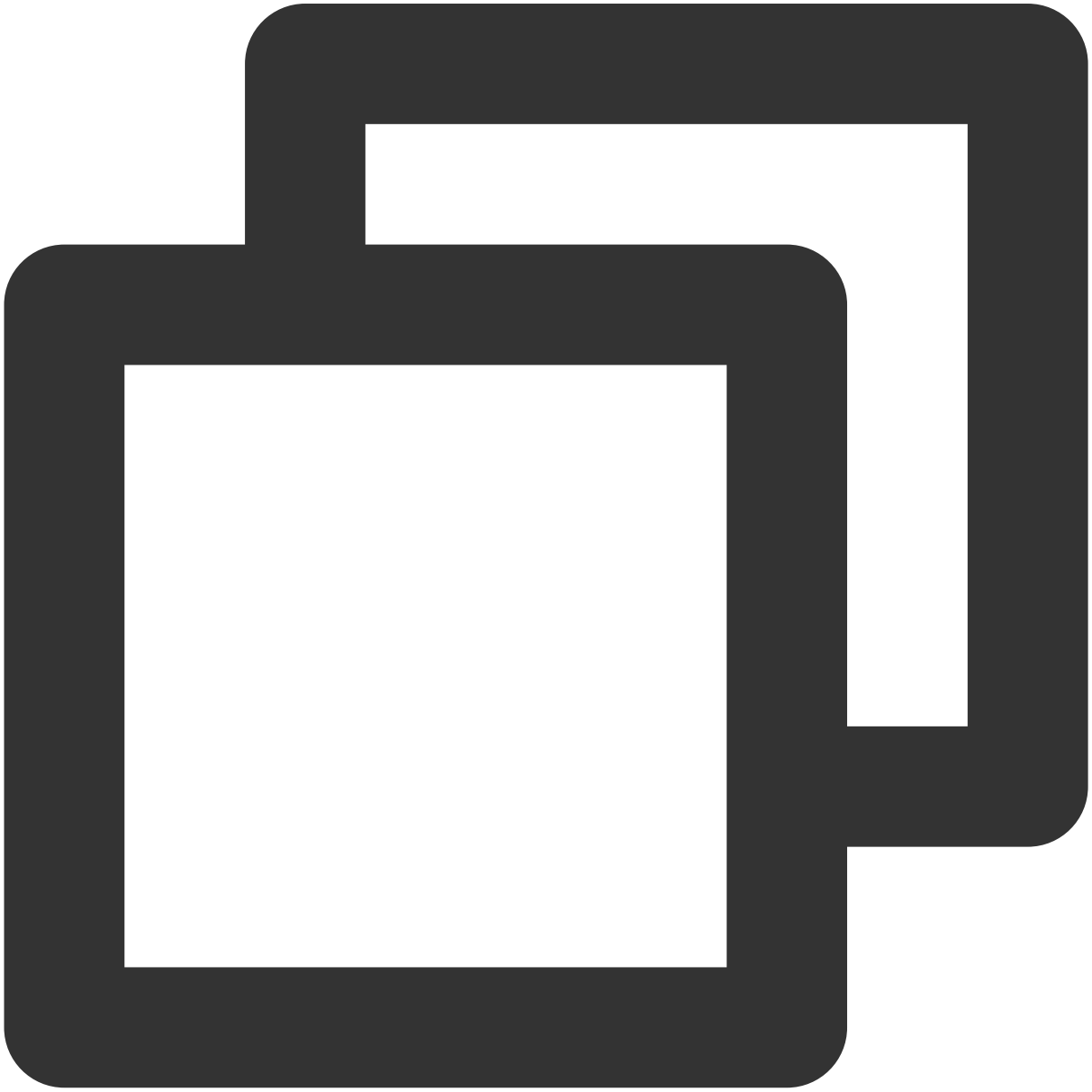
©2013-2022 Tencent Cloud. All rights reserved.

The usage of this API is wx.offDeviceMotionChange(function listener).

Function Description: Removes the monitoring function for device orientation change events.

Parameters and Description: function listener, the monitoring function passed into onDeviceMotionChange. If this parameter is not provided, all monitoring functions will be removed.

Sample Code:



```
const listener = function (res) { console.log(res) }
```

©2013-2022 Tencent Cloud. All rights reserved.

```
wx.onDeviceMotionChange(listener)
wx.offDeviceMotionChange(listener) // The same function object as used during monit
```

©2013-2022 Tencent Cloud. All rights reserved.

Accessibility

Last updated：2024-07-16 17:51:28

checkIsOpenAccessibility

This API is used via wx.checkIsOpenAccessibility(Object object).

Feature Description: Detects whether the visual accessibility feature has been activated.

Parameter and Description: Object.

Attribute	Type	Default value	Required	Description
success	function	-	No	Callback Function of Successful Interface Call
fail	function	-	No	Callback Function of Failing Interface Call
complete	function		No	Callback function executed upon the completion of the interface invocation (both successful and unsuccessful invocations)

Parameters for object.success callback function: Object.

Attribute	Type	Note ·
open	Boolean	Returns "true" when the auxiliary function narrative is enabled on iOS, or when talkback is enabled on Android.

©2013-2022 Tencent Cloud. All rights reserved.

Network

Last updated：2024-07-16 17:51:48

onNetworkStatusChange

This API is used via wx.onNetworkStatusChange(function callback).

Feature Description: function callback, a callback function for network state change events.

Parameter and Description: Function callback.

Callback function parameters for network status change events: Object res.

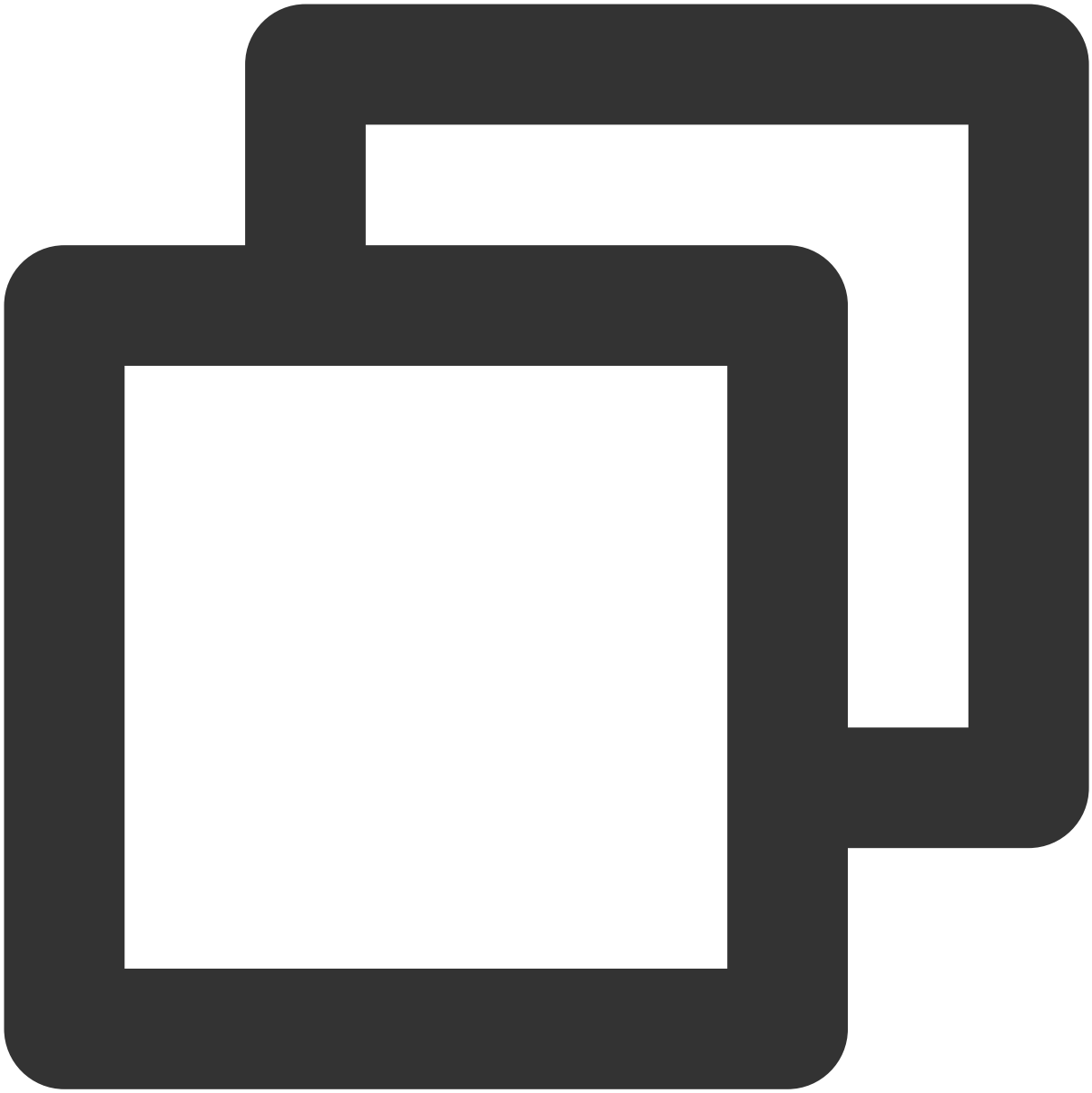
Attribute	Type	Description
isConnected	boolean	Current Network Connection Status
networkType	string	Network type

Legitimate Values of Legitimate

Value	Description
wifi	Wi-Fi Network
2g	2G Network
3g	3G Network
4g	4G Network
unknown	Uncommon Network Types on Android
none	No network

Sample Code

©2013-2022 Tencent Cloud. All rights reserved.



```
wx.onNetworkStatusChange(function (res) {
  console.log(res.isConnected)
  console.log(res.networkType)
})
```

getNetworkType

©2013-2022 Tencent Cloud. All rights reserved.

This API is used via wx.getNetworkType(Object object).

Feature Description: Gets the network type.

Parameter and Description: Object.

Attribute	Type	Default value	Required	Description
success	function	-	No	Callback Function of Successful Interface Call
fail	function	-	No	Callback Function of Failing Interface Call
complete	function	-	No	Callback function executed upon the completion of the interface invocation (both successful and unsuccessful invocations)

Object.success callback function: Object res.

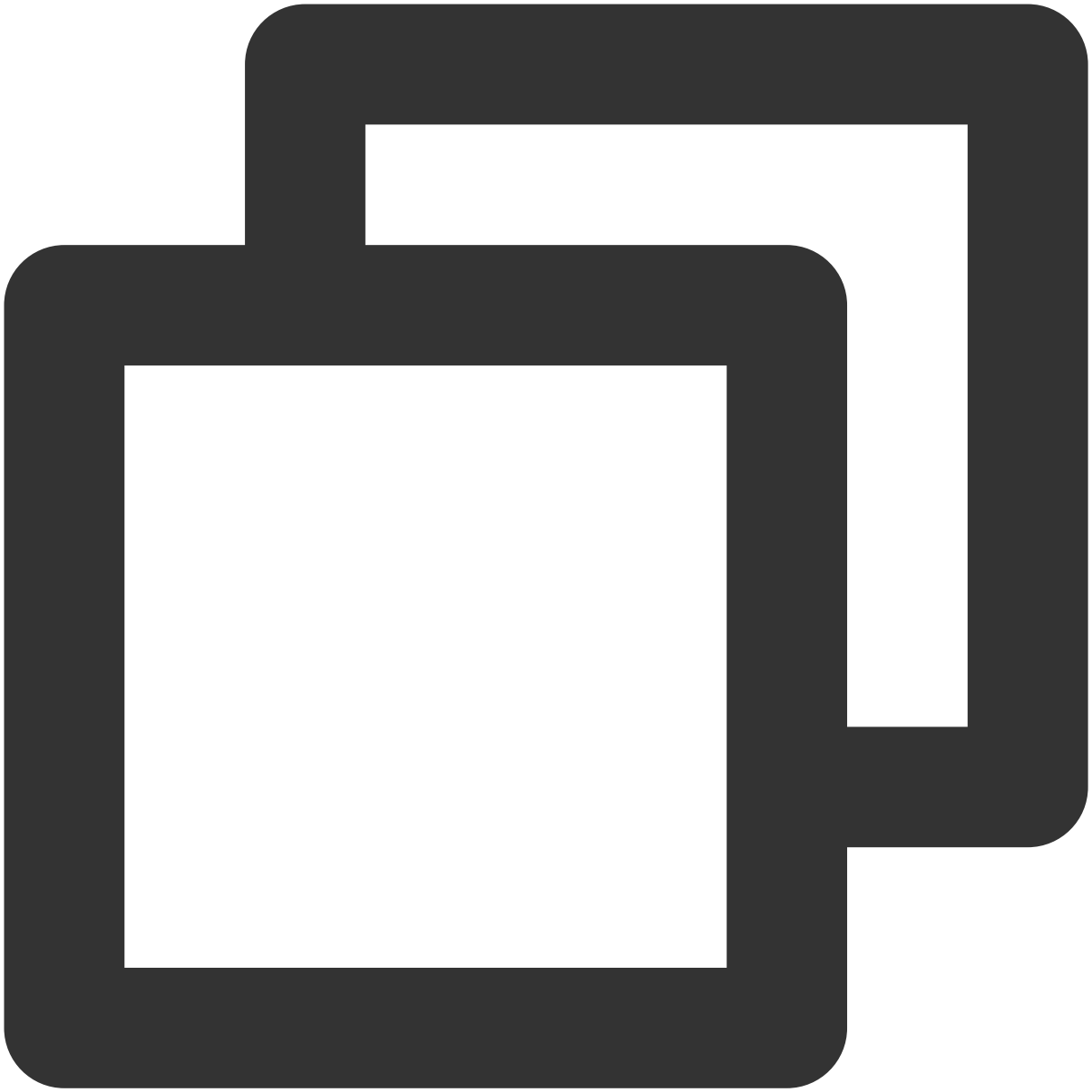
Attribute	Type	Description
networkType	string	Network type

Valid values of res.networkType

Value	Description
wifi	Wi-Fi Network
2g	2G Network
3g	3G Network
4g	4G Network
unknown	Uncommon Network Types on Android
none	No network

Sample Code

©2013-2022 Tencent Cloud. All rights reserved.



```
wx.getNetworkType({
  success(res) {
    const networkType = res.networkType
  }
})
```

offNetworkStatusChange

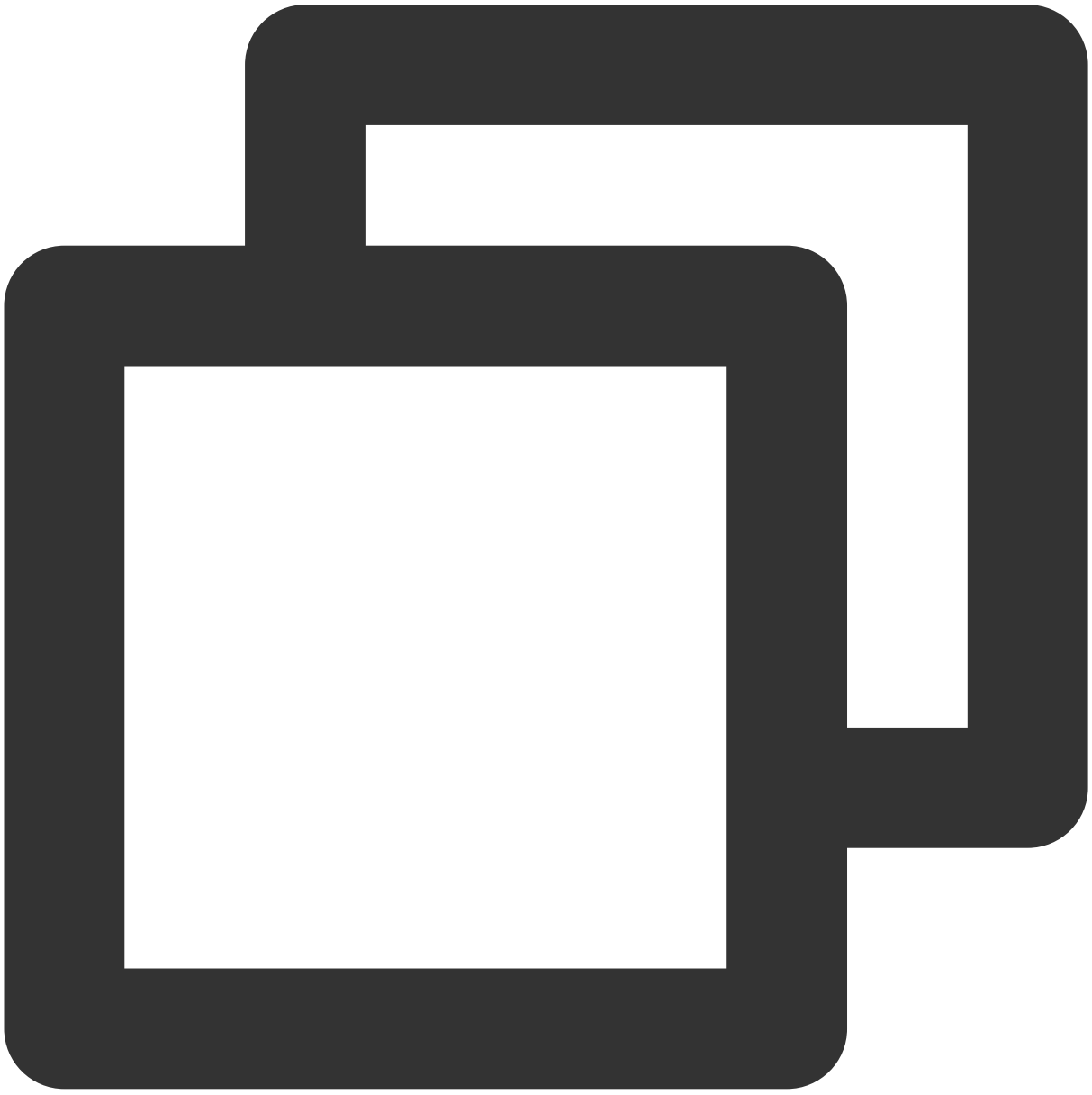
©2013-2022 Tencent Cloud. All rights reserved.

This API is used via wx.offNetworkStatusChange(function listener).

Feature Description: Removes the listener function for network status change events.

Parameter and Description: function listener, the listener function passed in by onNetworkStatusChange. If this parameter is not passed in, all listener functions will be removed.

Sample Code



```
const listener = function (res) { console.log(res) }
```

©2013-2022 Tencent Cloud. All rights reserved.

```
wx.onNetworkStatusChange(listener)
wx.offNetworkStatusChange(listener) // The same function object as the listener mus
```

©2013-2022 Tencent Cloud. All rights reserved.

Keyboard

Last updated：2024-07-16 17:52:14

onKeyboardHeightChange

This API is used via wx.onKeyboardHeightChange(function listener)

Feature Description: Listen for keyboard height change events.

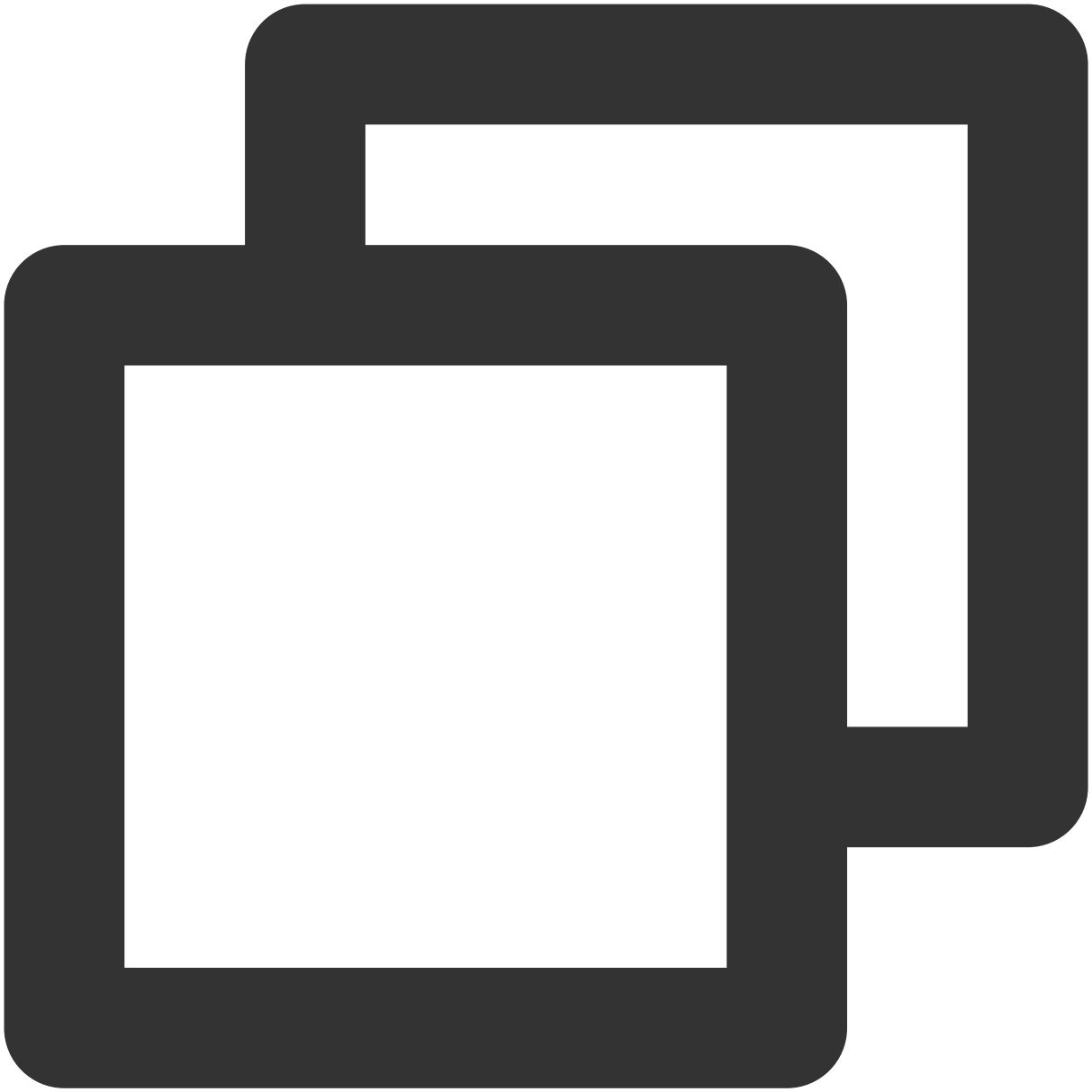
Parameter and Description: function callback, listener for keyboard height change events.

Listener function parameter: Object res.

Attribute	Type	Description
height	number	Keyboard Height

Sample Code

©2013-2022 Tencent Cloud. All rights reserved.



```
wx.onKeyboardHeightChange(res => {
  console.log(res.height)
})
```

hideKeyboard

This API is used via wx.hideKeyboard(Object object).

©2013-2022 Tencent Cloud. All rights reserved.

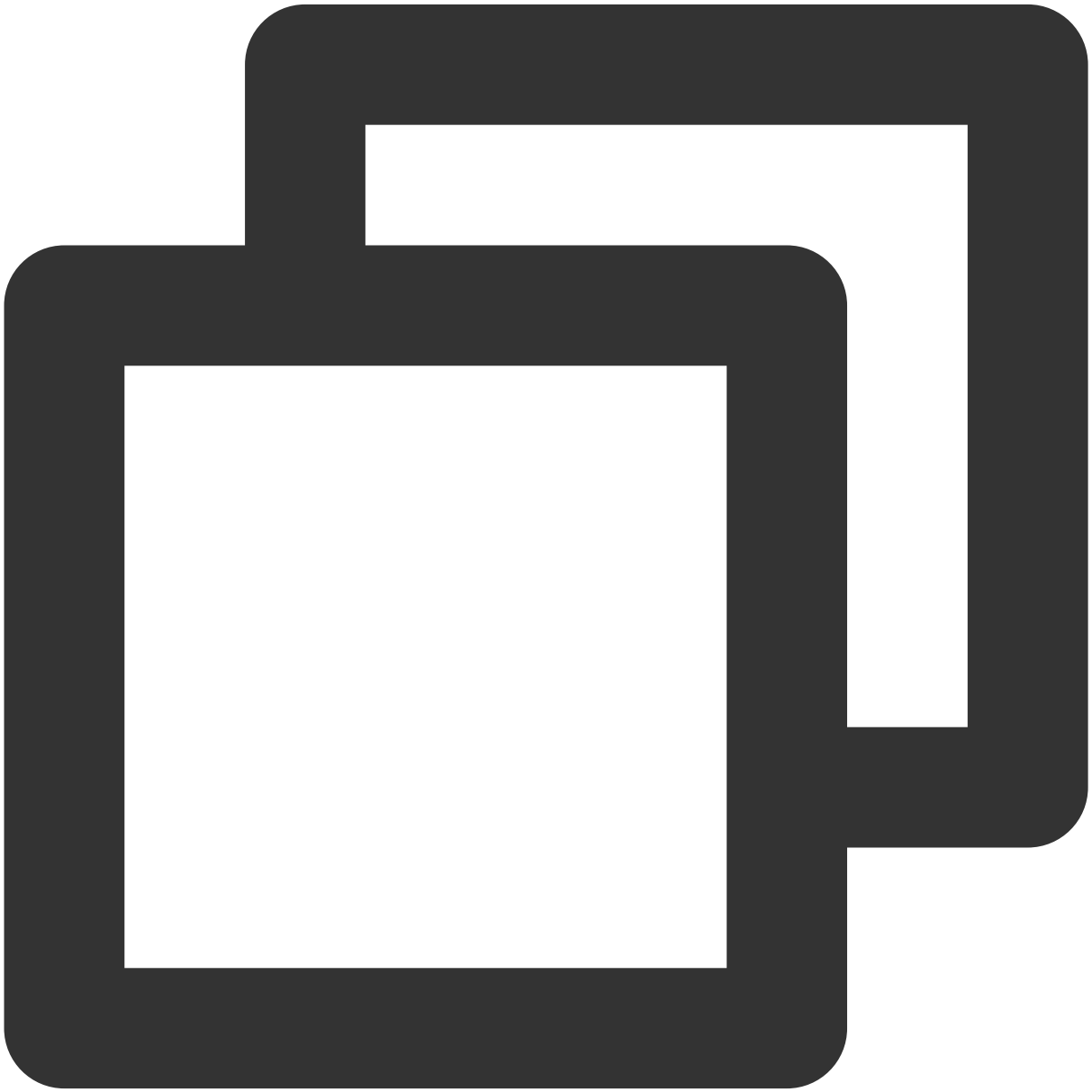
Feature Description: Call this interface manually to retract the keyboard after it has been pulled up by the input, textarea, etc. focus.

Parameter and Description: Object.

Attribute	Type	Default value	Required	Description
success	function	-	No	Callback Function of Successful Interface Call
fail	function	-	No	Callback Function of Failing Interface Call
complete	function	-	No	Callback function executed upon the completion of the interface invocation (both successful and unsuccessful invocations)

Sample Code

©2013-2022 Tencent Cloud. All rights reserved.



```
wx.hideKeyboard({
  complete: res => {
    console.log('hideKeyboard res', res)
  }
})
```

offKeyboardHeightChange

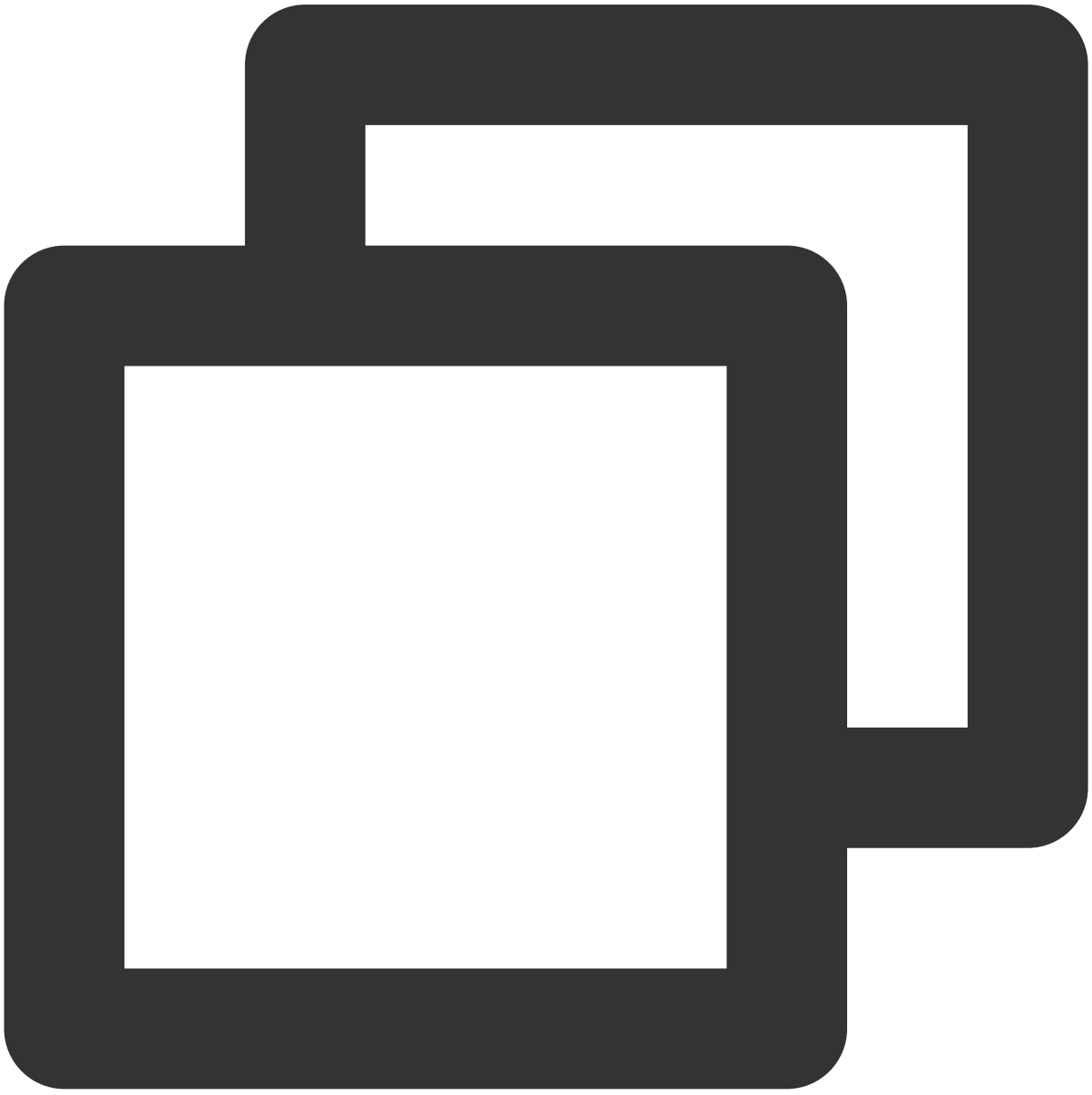
©2013-2022 Tencent Cloud. All rights reserved.

This API is used via wx.offKeyboardHeightChange(function listener).

Feature Description: Removes the listener function for keyboard height change events.

Parameter and Description: function listener, the listener function passed in by onKeyboardHeightChange. If this parameter is not passed in, all listener functions will be removed.

Sample Code



```
const listener = function (res) { console.log(res) }
```

©2013-2022 Tencent Cloud. All rights reserved.

```
wx.onKeyboardHeightChange(listener)
wx.offKeyboardHeightChange(listener) // The same function object as the listener mu
```

getSelectTextRange

This API is used via wx.getSelectedTextRange(Object object).

Feature Description: After input, textarea, etc. are focused, the cursor position in the input box is obtained. **This interface is only effective when called during focus.**

Parameter and Description: Object.

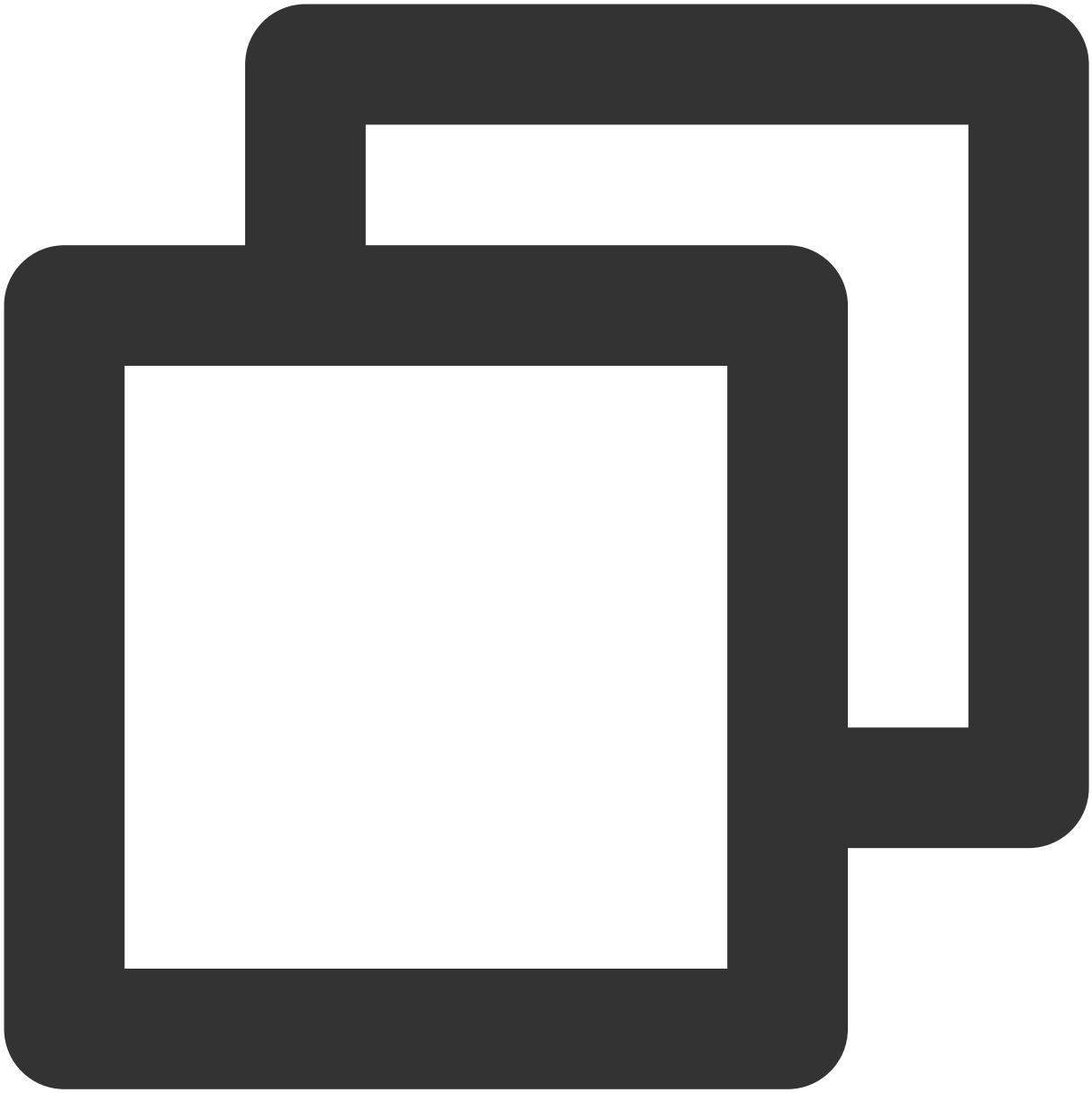
Attribute	Type	Default value	Required	Description
success	function	-	No	Callback Function of Successful Interface Call
fail	function	-	No	Callback Function of Failing Interface Call
complete	function	-	No	Callback function executed upon the completion of the interface invocation (both successful and unsuccessful invocations)

Object.success callback function: Object res.

Attribute	Type	Description
start	number	Initial cursor position in the input field
end	number	Final cursor position in the input field

Sample Code

©2013-2022 Tencent Cloud. All rights reserved.



```
wx.getSelectedTextRange({
  complete: res => {
    console.log('getSelectedTextRange res', res.start, res.end)
  }
})
```

©2013-2022 Tencent Cloud. All rights reserved.

Phone number

Last updated：2024-07-16 17:52:39

makePhonecall

This API is used via wx.makePhoneCall(Object object).

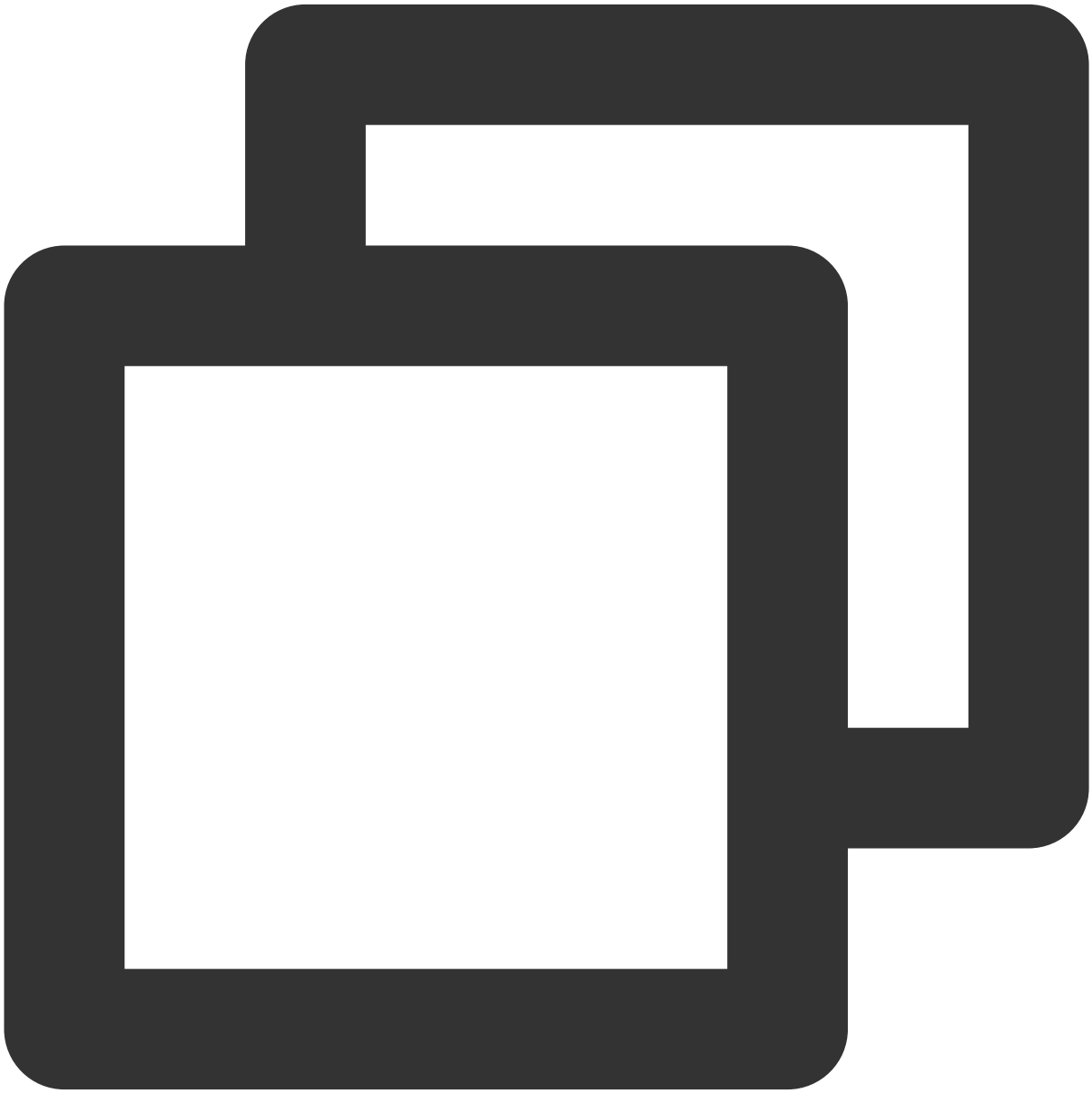
Feature Description: Initiates a phone call.

Parameter: Object.

Attribute	Type	Default value	Required	Description
phoneNumber	string	-	Yes	The phone number that needs to be dialed
success	function	-	No	Callback Function of Successful Interface Call
fail	function	-	No	Callback Function of Failing Interface Call
complete	function	-	No	Callback function executed upon the completion of the interface invocation (both successful and unsuccessful invocations)

Sample Code

©2013-2022 Tencent Cloud. All rights reserved.



```
wx.makePhoneCall({
  phoneNumber: '1340000' // This is merely an example and not an actual phone numbe
})
```

©2013-2022 Tencent Cloud. All rights reserved.

Accelerometer

Last updated：2024-07-16 17:53:01

starAccelerometer

This API is used via wx.startAccelerometer(Object object).

Note:

Depending on the performance of the device, as well as the current CPU and memory usage, there may be some discrepancies between the set `interval` and the actual execution frequency of the wx.onAccelerometerChange callback function.

Feature Description: Starts monitoring acceleration data.

Parameter and Description: Object.

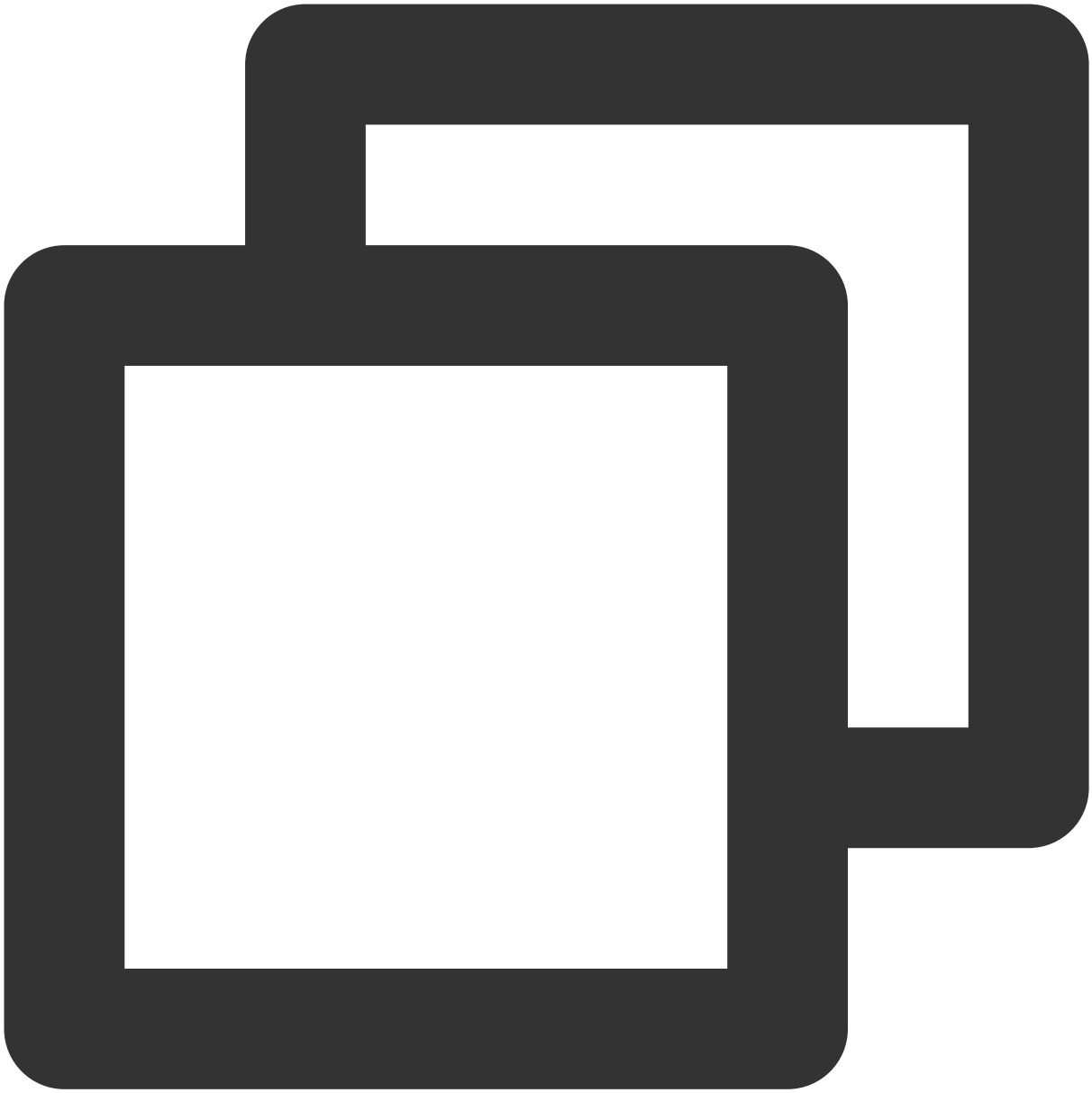
Attribute	Type	Default value	Required	Description
interval	string	normal	No	Execution frequency of the callback function for monitoring acceleration data
success	function	-	No	Callback Function of Successful Interface Call
fail	function	-	No	Callback Function of Failing Interface Call
complete	function	-	No	Callback function executed upon the completion of the interface invocation (both successful and unsuccessful invocations)

Valid values of object.interval

Value	Description
game	Suitable for updating the callback frequency of the game, approximately every 20 ms.
ui	Suitable for updating the callback frequency of the UI, approximately every 60 ms.
normal	Regular callback frequency, approximately every 200 ms.

Sample Code

©2013-2022 Tencent Cloud. All rights reserved.



```
wx.startAccelerometer({
  interval: 'game'
})
```

stopAccelerometer

This API is used via wx.stopAccelerometer(Object object).

©2013-2022 Tencent Cloud. All rights reserved.

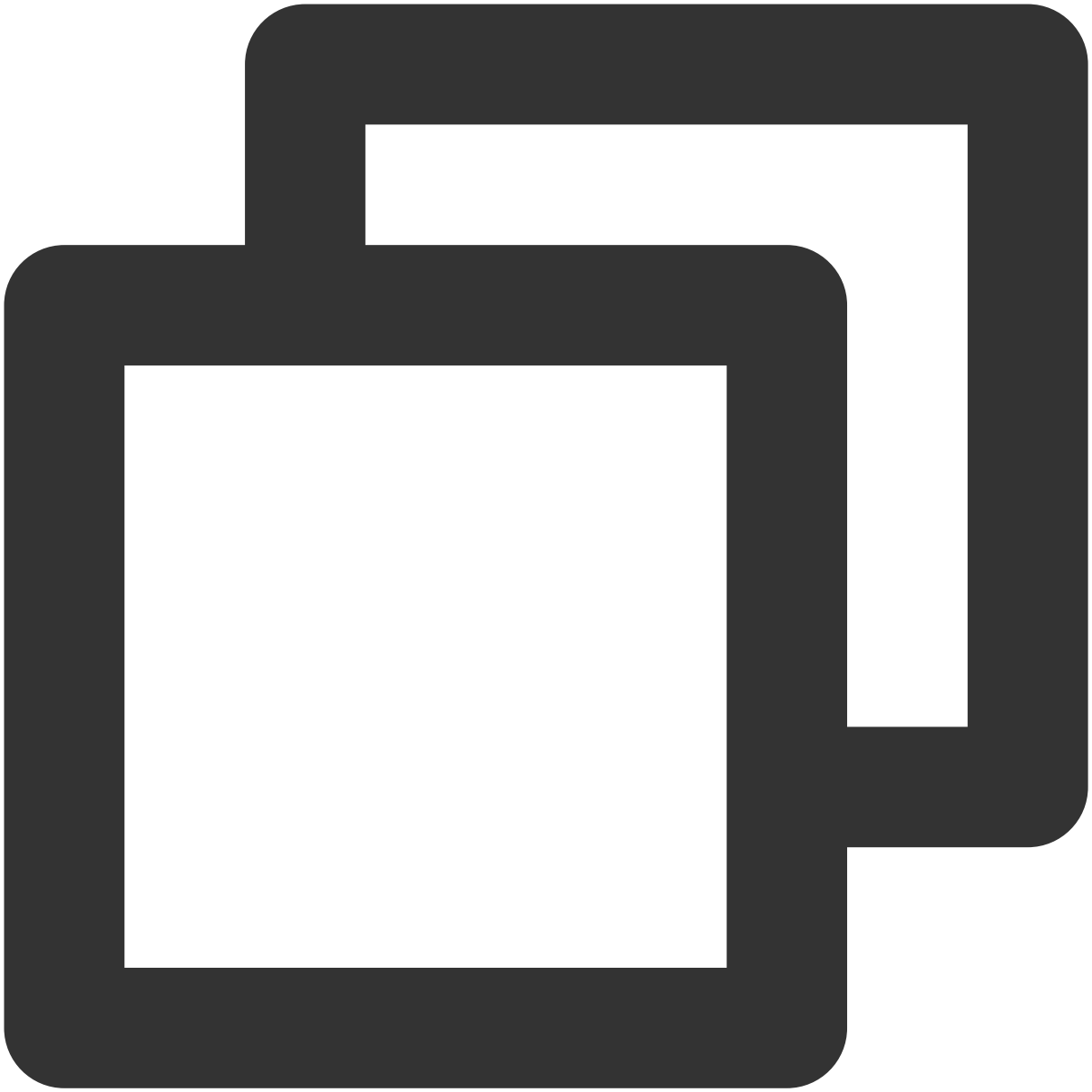
Feature Description: Stops monitoring acceleration data.

Parameter and Description: Object.

Attribute	Type	Default value	Required	Description
success	function	-	No	Callback Function of Successful Interface Call
fail	function	-	No	Callback Function of Failing Interface Call
complete	function	-	No	Callback function executed upon the completion of the interface invocation (both successful and unsuccessful invocations)

Sample Code

©2013-2022 Tencent Cloud. All rights reserved.



```
wx.stopAccelerometer()
```

onAccelerometerChange

This API is used via wx.onAccelerometerChange(function listener).

©2013-2022 Tencent Cloud. All rights reserved.

Feature Description: Monitor acceleration data events. The frequency depends on the interval parameter of wx.startAccelerometer(). The monitoring can be stopped using wx.stopAccelerometer().

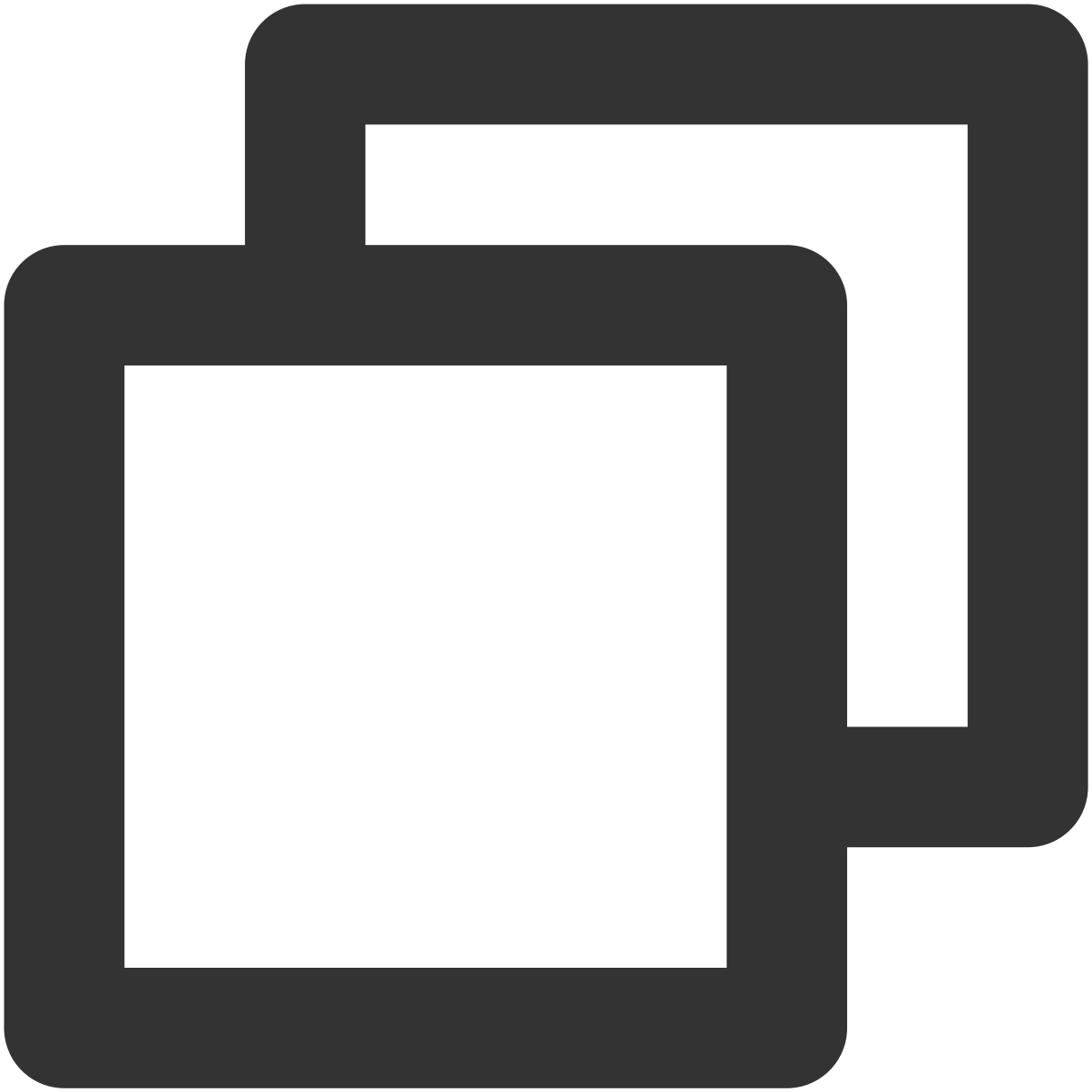
Parameter and Description: Function callback.

The callback function for acceleration data events:

Attribute	Type	Description
x	number	X-axis
y	number	Y-axis
z	number	Z-axis

Sample Code

©2013-2022 Tencent Cloud. All rights reserved.



```
wx.onAccelerometerChange(function (res) {
  console.log(res.x)
  console.log(res.y)
  console.log(res.z)
})
```

offAccelerometerChange

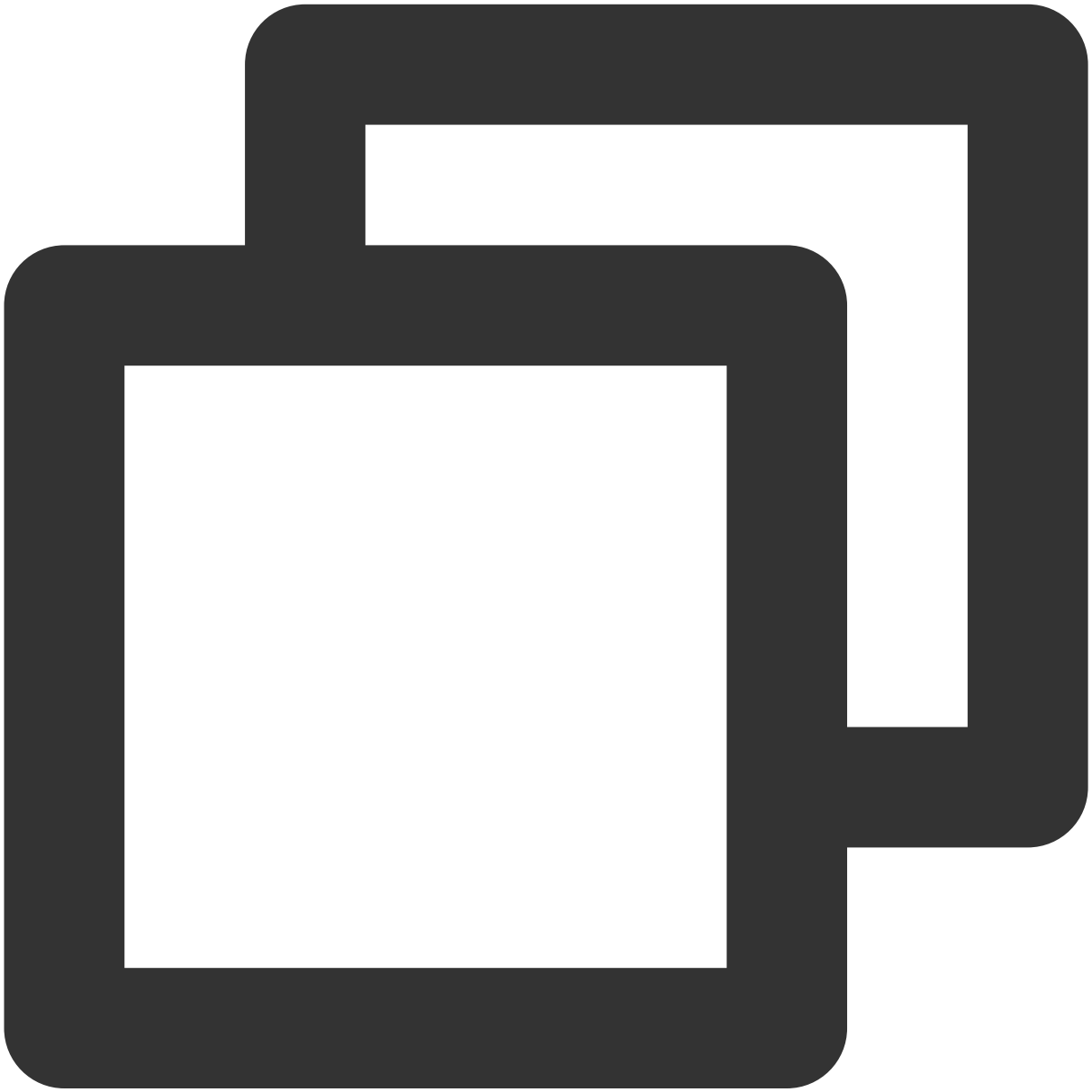
©2013-2022 Tencent Cloud. All rights reserved.

This API is used via wx.offAccelerometerChange(function listener).

Feature Description:

Parameters and Description: function listener, the listening function passed into onAccelerometerChange. If this parameter is not provided, all listening functions will be removed.

Sample Code



```
const listener = function (res) { console.log(res) }
```

©2013-2022 Tencent Cloud. All rights reserved.

```
wx.onAccelerometerChange(listener)
wx.offAccelerometerChange(listener) // The same function object as the listener mus
```

©2013-2022 Tencent Cloud. All rights reserved.

Compass

Last updated：2024-07-16 17:53:21

starCompass

The method of utilizing this API is wx.startCompass(Object object).

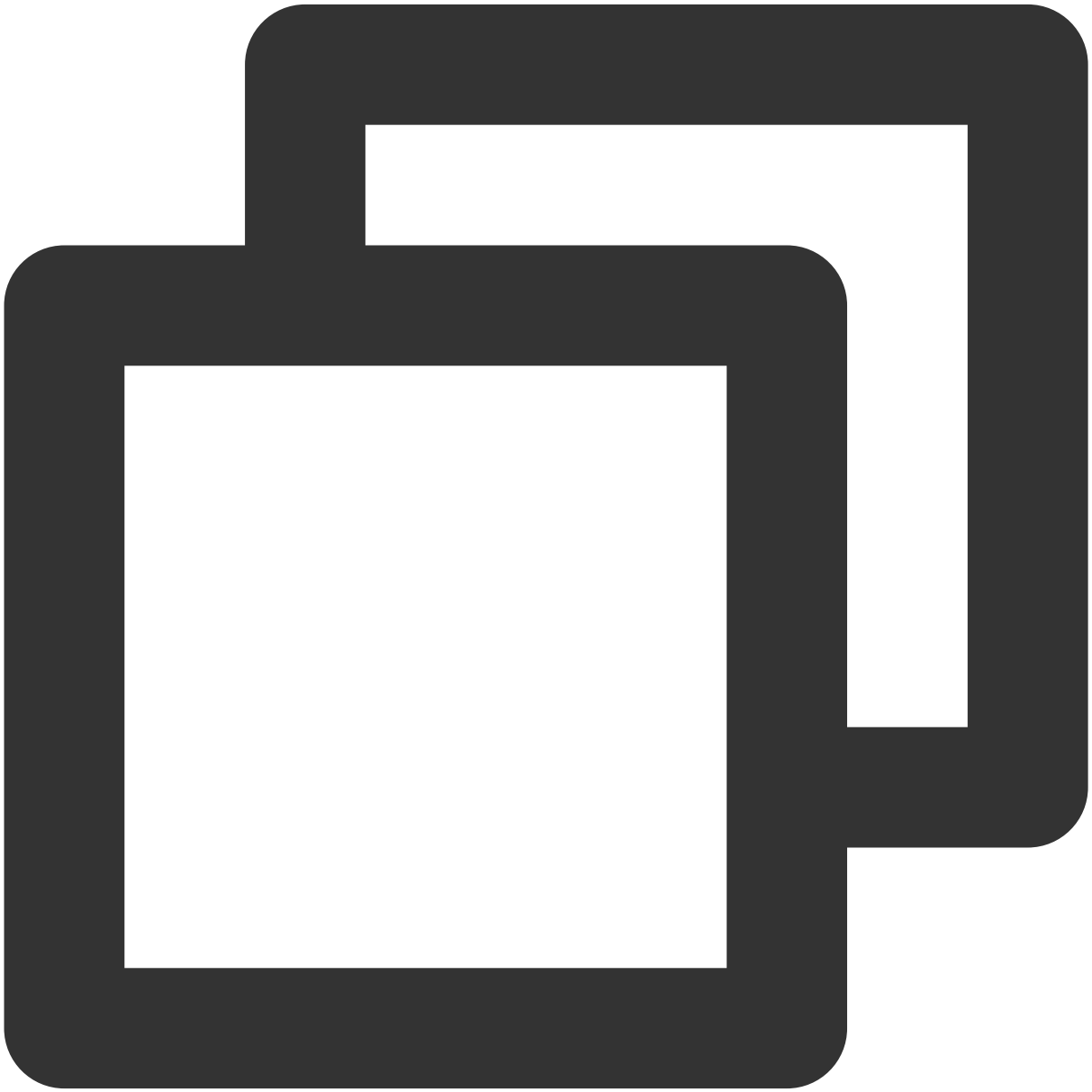
Function Description: Initiate the monitoring of compass data.

Parameters and Description: Object object.

Attributes	Types	Default value	Required	Note
success	function	-	Not required	Successful Interface Call Response
fail	function	-	Not required	Callback function for unsuccessful interface invocation
complete	function	-	Not required	Callback function executed upon the completion of the interface invocation (both successful and unsuccessful invocations)

Sample Code:

©2013-2022 Tencent Cloud. All rights reserved.



```
wx.startCompass()
```

stopCompass

The usage method for this API is wx.stopCompass(Object object)

©2013-2022 Tencent Cloud. All rights reserved.

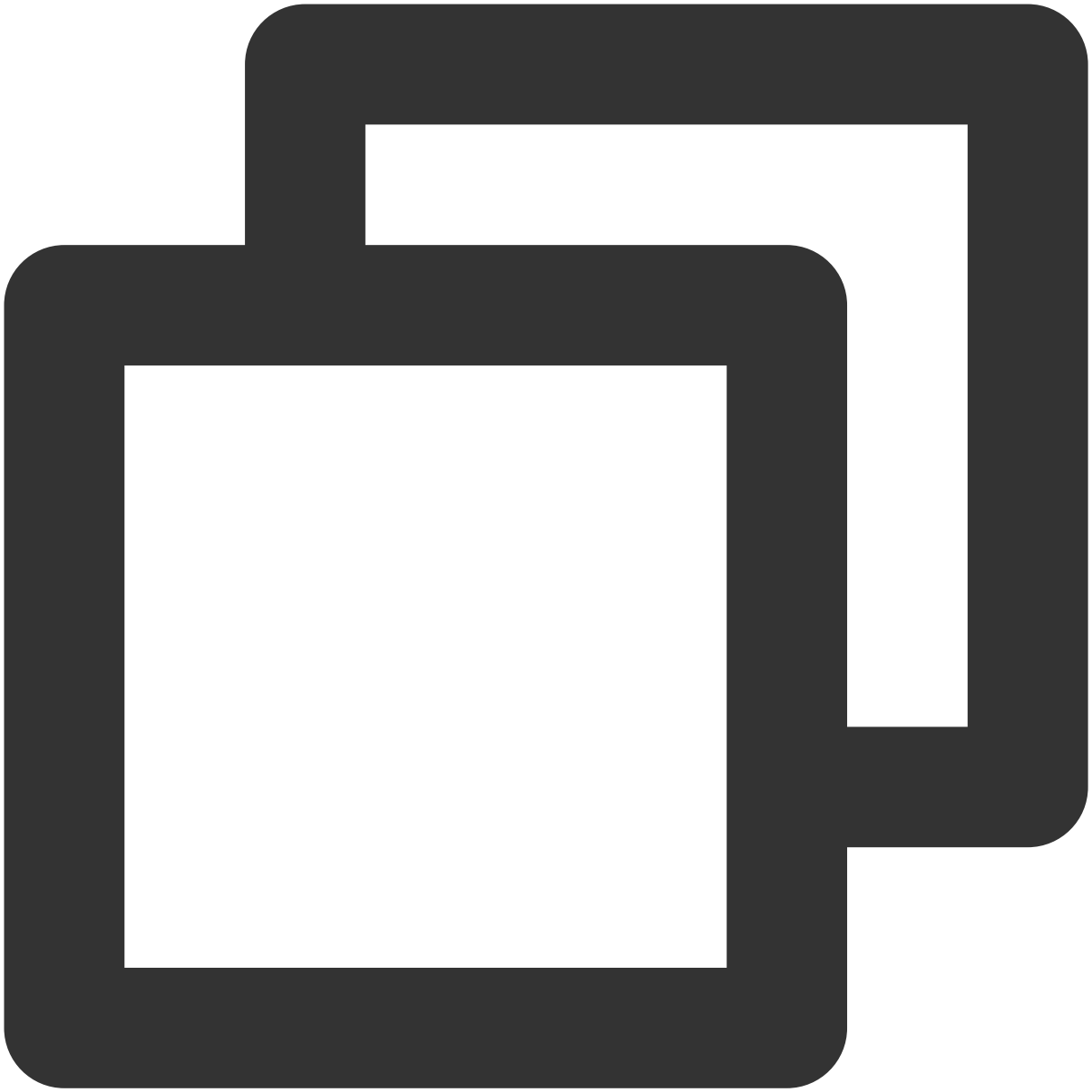
Function Description: Cease the monitoring of compass data.

Parameters and Description: Object object.

Attributes	Types	Default value	Required	Note
success	function	-	Not required	Callback function for successful interface invocation
fail	function	-	Not required	Callback function for unsuccessful interface invocation
complete	function	-	Not required	Callback function executed upon the completion of the interface invocation (both successful and unsuccessful invocations)

Example Code

©2013-2022 Tencent Cloud. All rights reserved.



```
wx.stopCompass()
```

onCompassChange

The usage method for this API is wx.onCompassChange(function callback)

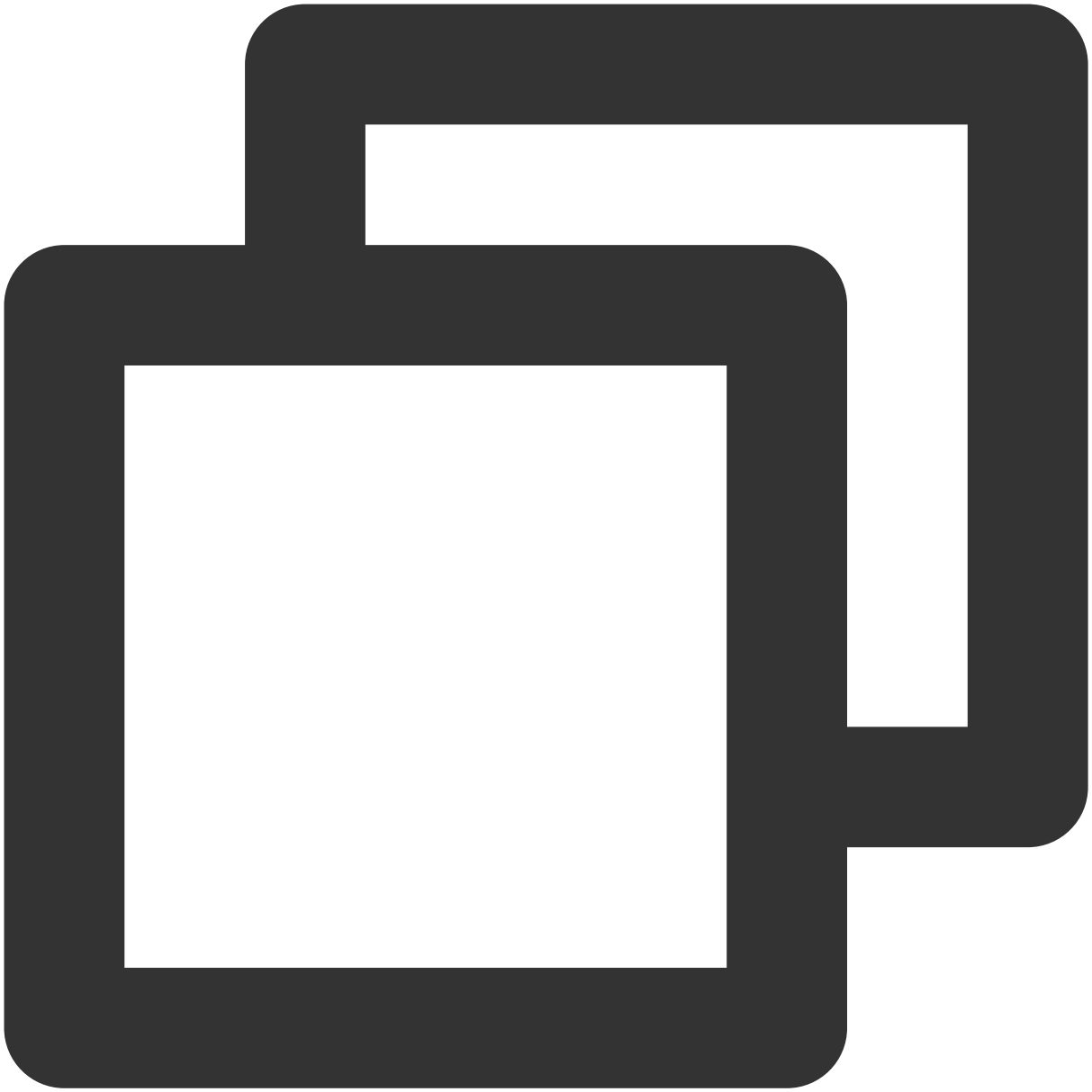
©2013-2022 Tencent Cloud. All rights reserved.

Function Description: Monitor the compass data change event. Frequency: 5 times/second. After the interface is called, it will automatically start monitoring. You can use wx.stopCompass to stop monitoring.
Parameters and Description: function callback, the callback function for the compass data change event. The arguments Object res are as follows:

Attributes	Types	Note
direction	number	Facing direction degrees
accuracy	number/string	Precision

Example Code

©2013-2022 Tencent Cloud. All rights reserved.



```
wx.onCompassChange(function (res) {
    console.log(res.direction)
})
```

Difference of accuracy in iOS/Android: Due to platform differences, the value of accuracy varies between iOS and Android.

iOS: Accuracy is a numerical value representing the deviation relative to the Magnetic North. 0 indicates the device is pointing towards the Magnetic North, 90 indicates East, 180 indicates South, and so forth.

Android: Accuracy is an enumerated value of string type.

©2013-2022 Tencent Cloud. All rights reserved.

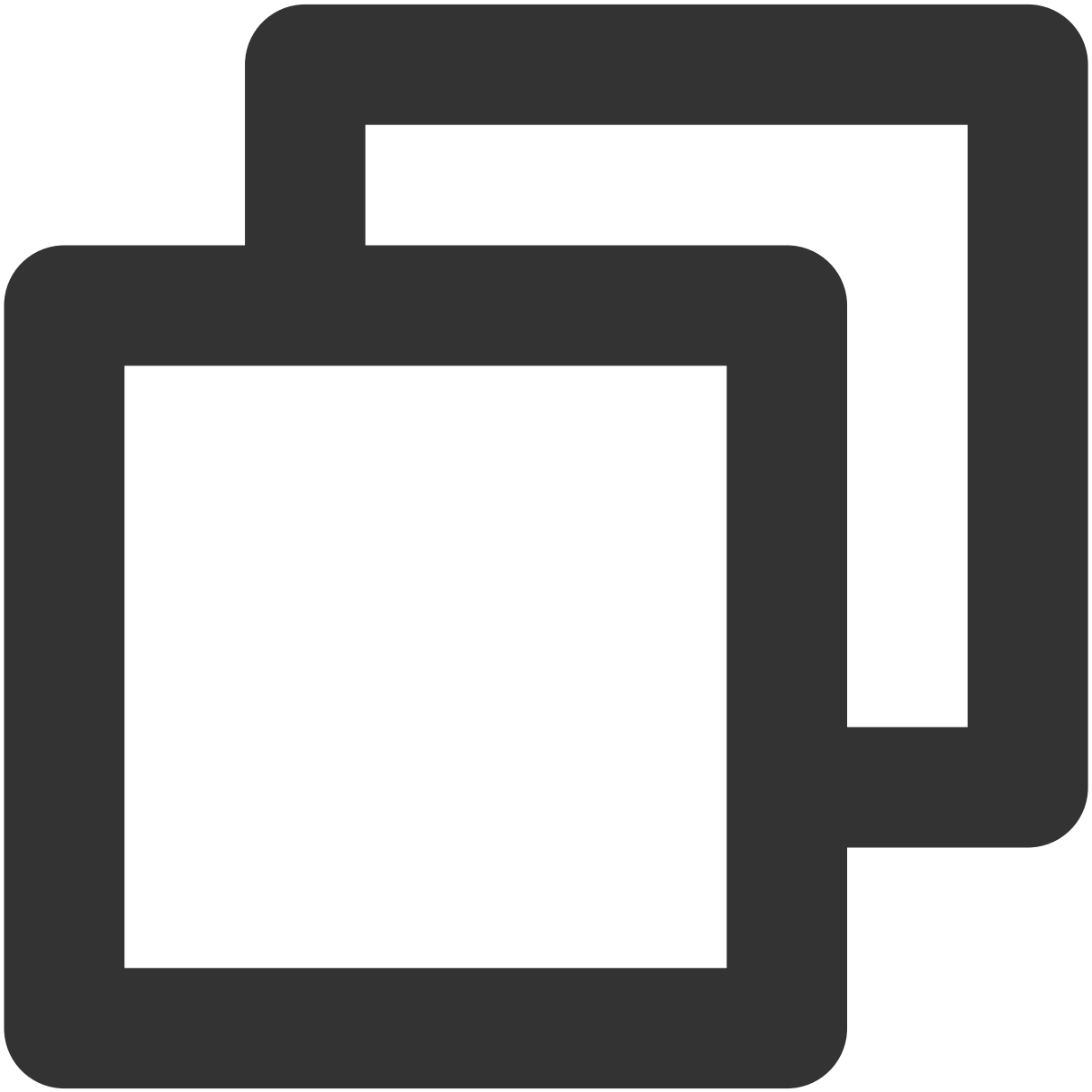
Value	Note
high	High Precision
medium	Moderate Precision
low	Low Precision
no-contact	Unreliable, sensor connection lost.
unreliable	Unreliable, reason unknown.
unknow ${value}	Unknown precision enumeration value, that is, the value representing precision returned by the Android system at this time is not a standard precision enumeration value.

offCompassChange

The usage method for this API is wx.offCompassChange(function listener).

Function Description: Removes the listener function for compass data change events.
Parameters and Description: function listener, the listening function passed into onCompassChange. If this parameter is not passed, all listening functions will be removed.
Sample Code:

©2013-2022 Tencent Cloud. All rights reserved.



```
const listener = function (res) { console.log(res) }

wx.onCompassChange(listener)
wx.offCompassChange(listener) // The same function object as the listener must be p
```

©2013-2022 Tencent Cloud. All rights reserved.

Gyroscope

Last updated：2024-07-16 17:53:42

startGyroscope

This API is used via wx.startGyroscope(Object object).

Feature Description: Starts monitoring gyroscope data.

Parameter and Description: Object.

Attribute	Type	Default value	Required	Description
interval	string	normal	No	Execution frequency of the callback function for monitoring gyroscope data
success	function	-	No	Callback Function of Successful Interface Call
fail	function	-	No	Callback Function of Failing Interface Call
complete	function	-	No	Callback function executed upon the completion of the interface invocation (both successful and unsuccessful invocations)

Valid values of object.interval

Value	Description
game	Suitable for updating the callback frequency of the game, approximately every 20 ms.
ui	Suitable for updating the callback frequency of the UI, approximately every 60 ms.
normal	Regular callback frequency, approximately every 200 ms.

stopGyroscope

This API is used via wx.stopGyroscope(Object object).

©2013-2022 Tencent Cloud. All rights reserved.

Feature Description: Stops monitoring gyroscope data.

Parameter and Description: Object.

Attribute	Type	Default value	Required	Description
success	function	-	No	Callback Function of Successful Interface Call
fail	function	-	No	Callback Function of Failing Interface Call
complete	function	-	No	Callback function executed upon the completion of the interface invocation (both successful and unsuccessful invocations)

onGyroscopeChange

This API is used via wx.onGyroscopeChange(function callback).

Feature Description: Monitors gyroscope data change events. The frequency depends on the interval parameter of wx.startGyroscope(). The monitoring can be stopped using wx.stopGyroscope().

Parameter and Description: function callback, the callback function for gyroscope data change events. The arguments Object res are as follows:

Attribute	Type	Description
x	number	Angular velocity of the x-axis.
y	number	Angular velocity of the y-axis.
z	number	Angular velocity of the z-axis.

offGyroscopeChange

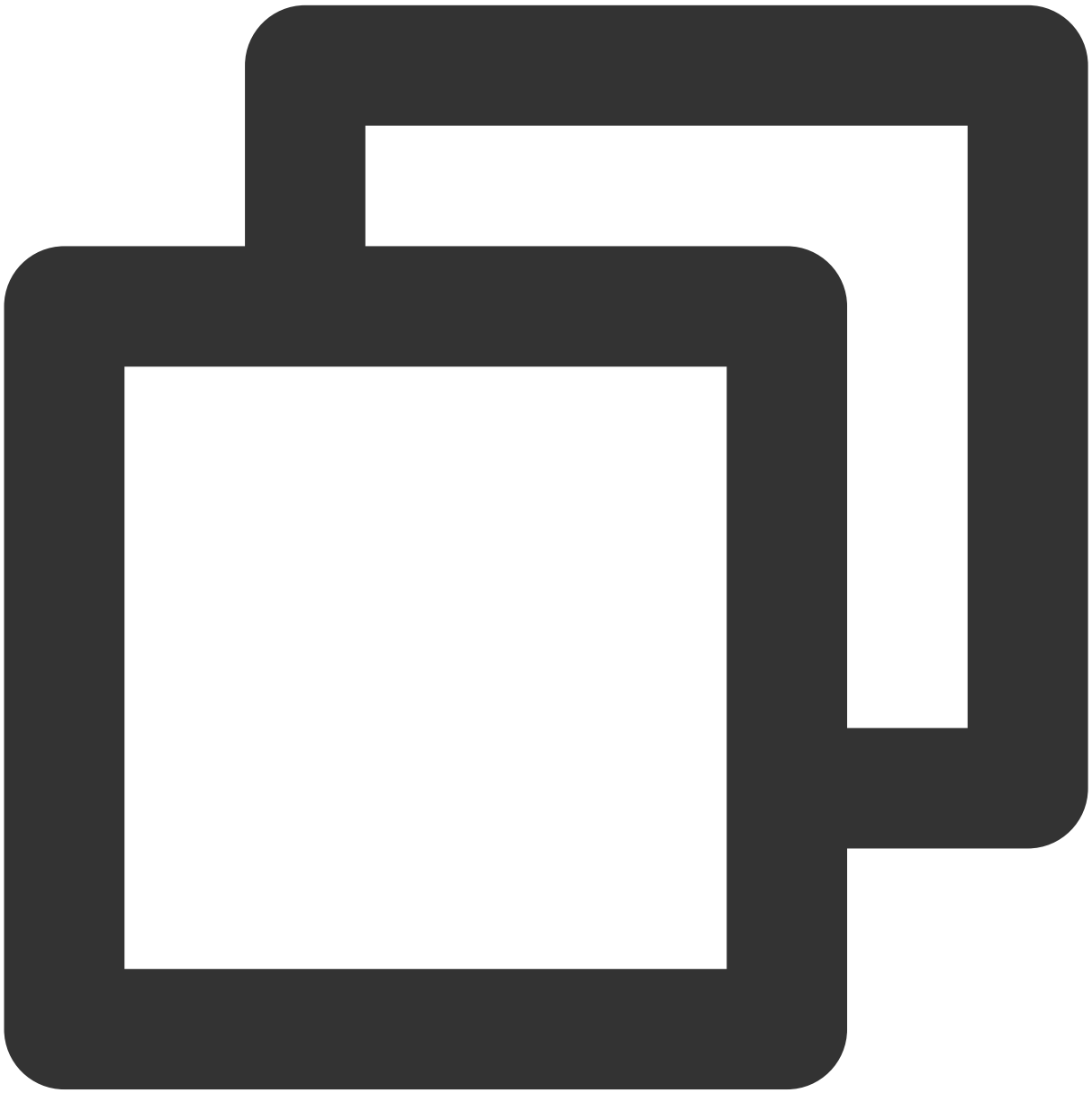
This method is used via wx.offGyroscopeChange(function listener).

Feature Description: Removes the listener function for gyroscope data change events.

Parameter and Description: function listener, the listener function passed in by onGyroscopeChange. If this parameter is not passed in, all listener functions will be removed.

©2013-2022 Tencent Cloud. All rights reserved.

Sample Code



```
const listener = function (res) { console.log(res) }

wx.onGyroscopeChange(listener)
wx.offGyroscopeChange(listener) // The same function object as the listener must be
```

©2013-2022 Tencent Cloud. All rights reserved.

SMS

Last updated：2024-07-16 17:54:01

sendSms

This API is used via wx.sendSms(Object object).

Feature Description: Initiates the mobile interface for sending SMS.

Parameter and Description: Object

Attribute	Type	Default value	Required	Description
phoneNumber	string	-	No	Mobile number pre-filled in the SMS sending panel
content	string	-	No	Pre-filled content in the SMS sending panel
success	function	-	No	Callback Function of Successful Interface Call
fail	function	-	No	Callback Function of Failing Interface Call
complete	function	-	No	Callback function executed upon the completion of the interface invocation (both successful and unsuccessful invocations)

©2013-2022 Tencent Cloud. All rights reserved.

Code scanning

Last updated：2024-07-16 17:54:33

scanCode

This API is used via wx.scanCode(Object object).

Feature Description: Initiates the client's scanning interface for barcode scanning.

Parameter and Description: Object.

Attribute	Type	Valid Values and Descriptions	Default value	Required	Description
onlyFromCamera	boolean	-	false	No	Is it exclusively permissible to scan from the camera, disallowing the selection of images from the album?
scanType	Array.<string>	barCode: Barcode qrCode: QR code datamatrix: Data Matrix code pdf417: PDF417 Barcode	['barCode', 'qrCode']	No	Code scanning type
success	function	-	-	No	Callback Function of Successful Interface Call
fail	function	-	-	No	Callback Function of Failing Interface Call
complete	function	-	-	No	Callback function executed upon the completion of the interface invocation (both successful and unsuccessful invocations)

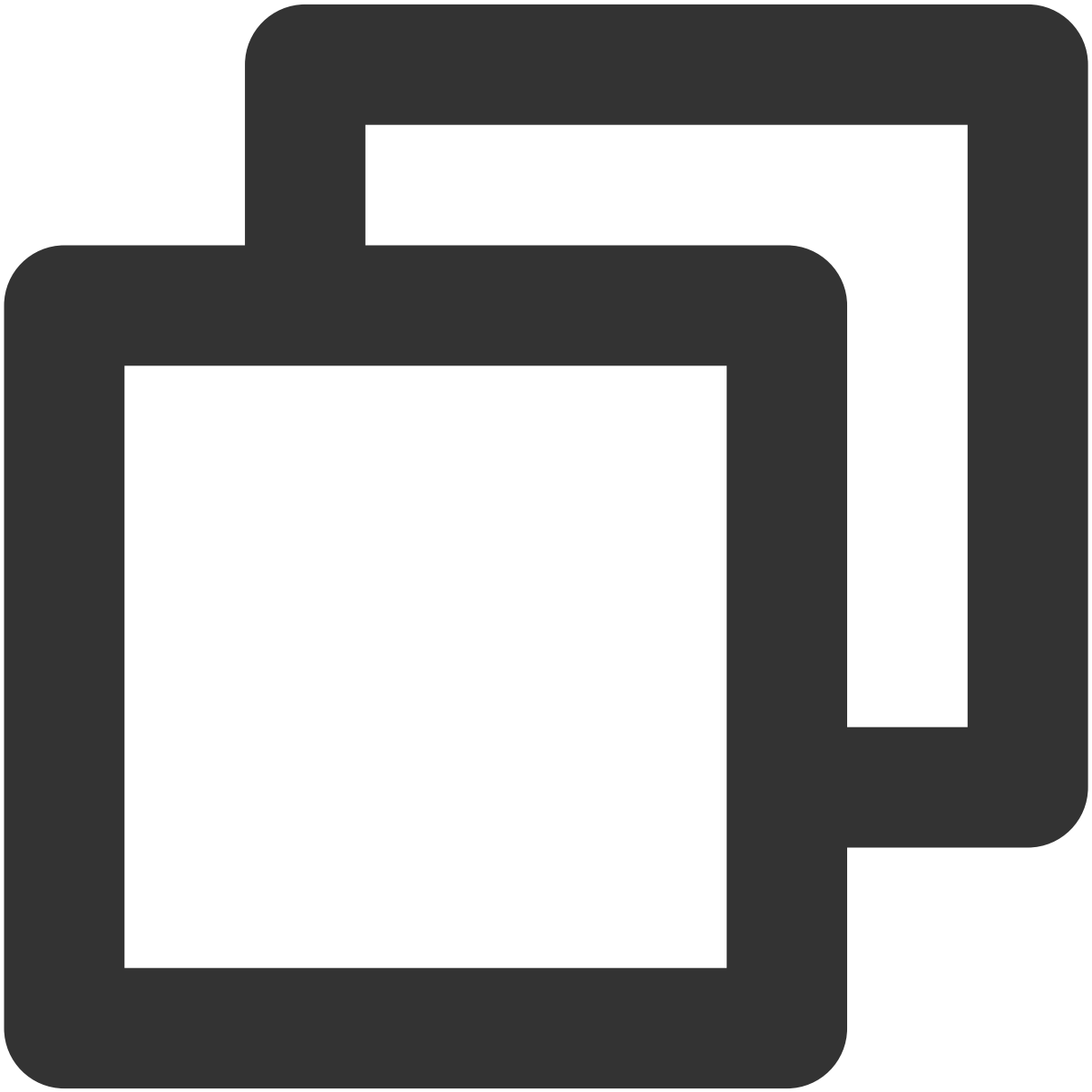
©2013-2022 Tencent Cloud. All rights reserved.

Parameters for object.success callback function: Object res.

Attribute	Type	Valid Values and Descriptions	Description
result	string	-	Content of the scanned code
scanType	string	QR_CODE: QR code AZTEC: Barcode CODABAR: Barcode CODE_39: Barcode CODE_93: Barcode CODE_128: Barcode DATA_MATRIX: QR code EAN_8: Barcode EAN_13: Barcode ITF: Barcode MAXICODE: Barcode PDF_417: QR code RSS_14: Barcode RSS_EXPANDED: Barcode UPC_A: Barcode UPC_E: Barcode UPC_EAN_EXTENSION: Barcode WX_CODE: QR code CODE_25: Barcode	Type of scanned code
charSet	string	-	String set of the scanned code
path	string	-	When the scanned code is the current mini program's QR code, this field will be returned, containing the path carried by the QR code.
rawData	string	-	Raw data, Base64-encoded

Sample Code

©2013-2022 Tencent Cloud. All rights reserved.



```
// Allow scanning from both the camera and album
wx.scanCode({
  success (res) {
    console.log(res)
  }
})

// Allow scanning from the camera only
wx.scanCode({
  onlyFromCamera: true,
  success (res) {
```

©2013-2022 Tencent Cloud. All rights reserved.

```
    console.log(res)
  }
})
```

©2013-2022 Tencent Cloud. All rights reserved.

Encryption

Last updated：2024-07-16 17:54:51

getRandomValues

This API is used via wx.getRandomValues(Object object).

Feature Description: Gets cryptographically secure random numbers.

Parameter and Description: Object.

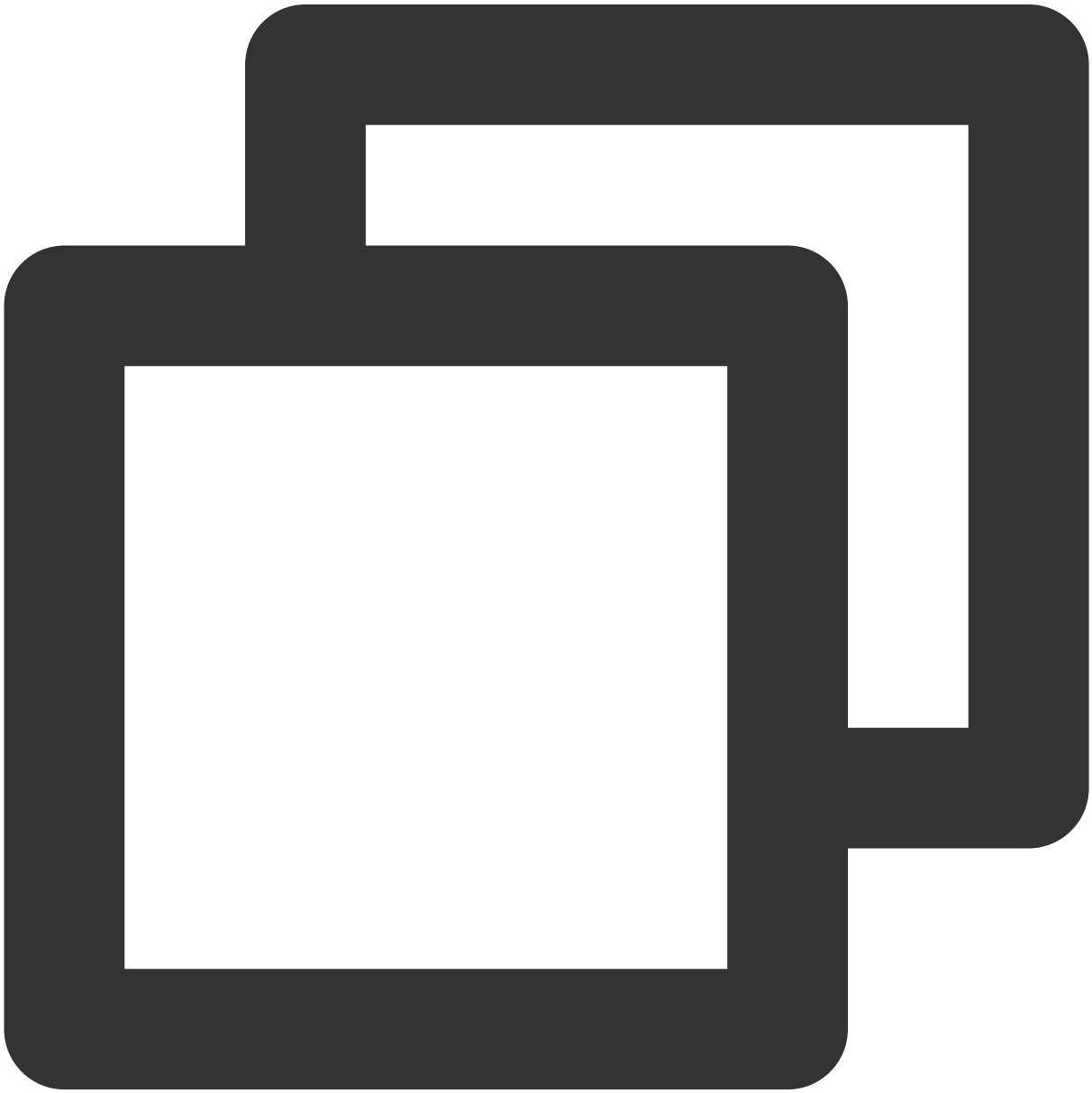
Attribute	Type	Default value	Required	Description
length	number	-	Yes	Integer, the byte size of the generated random number, with a maximum of 1048576.
success	function	-	No	Callback Function of Successful Interface Call
fail	function	-	No	Callback Function of Failing Interface Call
complete	function	-	No	Callback function executed upon the completion of the interface invocation (both successful and unsuccessful invocations)

Parameters for object.success callback function: Object res.

Attribute	Type	Description
randomValues	ArrayBuffer	Random number content, with a length equivalent to the passed-in byte size.

Sample Code

©2013-2022 Tencent Cloud. All rights reserved.



```
wx.getRandomValues({
  length: 6 // Generate a random number of 6-byte length,
  success: res => {
    console.log(wx.arrayBufferToBase64(res.randomValues)) // Print after converting
  }
})
```

©2013-2022 Tencent Cloud. All rights reserved.

NFC—Ndef

Last updated：2024-07-16 17:55:18

.connect

The method is utilized as Ndef.connect().

Function Description: Establishes a connection with the NFC tag.

Parameters and Description: Object object.

Attributes	Types	Default value	Required	Note
success	function	-	Not required	Callback function for successful interface invocation
fail	function	-	Not required	Callback function for unsuccessful interface invocation
complete	function	-	Not required	Callback function executed upon the completion of the interface invocation (both successful and unsuccessful invocations)

.close

The method is employed as Ndef.close().

Function Description: Disconnects the link.

Parameters and Description: Object object.

Attributes	Types	Default value	Required	Note
success	function	-	Not required	Callback function for successful interface invocation
fail	function	-	Not	Callback function for unsuccessful interface

©2013-2022 Tencent Cloud. All rights reserved.

			required	invocation
complete	function	-	Not required	Callback function executed upon the completion of the interface invocation (both successful and unsuccessful invocations)

Error Code

Error Codes	Error Message	Note
13000	The device does not support NFC	-
13001	The system's NFC switch is not activated.	-
13010	An unknown error occurred.	-
13019	user is not authorized	User Authorization Denied
13011	invalid parameter	Invalid parameter
13012	parse NdefMessage failed	Failed to parse the parameter into NdefMessage.
13021	NFC discovery already started	NFC scanning has already commenced.
13018	NFC discovery has not started	Attempted to halt NFC scanning without initiating it.
13022	Tech already connected	The tag has been successfully connected.
13023	Tech has not connected	Attempted to disconnect without a tag being connected.
13013	NFC tag has not been discovered	No NFC tag was detected.
13014	invalid tech	Invalid tag technology.
13015	unavailable tech	Failed to acquire the corresponding technology from the tag.
13024	function not support	The current tag technology does not support this feature.
13017	system internal error	The related read-write operation has failed.
13016	connect fail	Connection failed

©2013-2022 Tencent Cloud. All rights reserved.

.setTimeout

The method is utilized as Ndef.setTimeout(Object object).

Function Description: Setting the timeout duration.

Parameters and Description: Object object

Attributes	Types	Default value	Required	Note
timeout	number	-	Supported	Setting the timeout duration (ms)
success	function	-	Not required	Callback function for successful interface invocation
fail	function	-	Not required	Callback function for unsuccessful interface invocation
complete	function	-	Not required	Callback function executed upon the completion of the interface invocation (both successful and unsuccessful invocations)

Error Code

Error Codes	Error Message	Note
13000	The device does not support NFC	-
13001	The system's NFC switch is not activated.	-
13010	An unknown error occurred.	-
13019	user is not authorized	User Authorization Denied
13011	invalid parameter	Invalid parameter
13012	parse NdefMessage failed	Failed to parse the parameter into NdefMessage.
13021	NFC discovery already started	NFC scanning has already commenced.
13018	NFC discovery has not started	Attempted to halt NFC scanning without initiating it.
13022	Tech already connected	The tag has been successfully connected.

©2013-2022 Tencent Cloud. All rights reserved.

13023	Tech has not connected	Attempted to disconnect without a tag being connected.
13013	NFC tag has not been discovered	No NFC tag was detected.
13014	invalid tech	Invalid tag technology.
13015	unavailable tech	Failed to acquire the corresponding technology from the tag.
13024	function not support	The current tag technology does not support this feature.
13017	system internal error	The related read-write operation has failed.
13016	connect fail	Connection failed

.isConnected

The method is utilized as Ndef.isConnected().

Note:

This interface has been deprecated; developers can maintain the connection status independently.

Function Description: Verifies if a connection has been established.

Parameters and Description: Object object.

Attributes	Types	Default value	Required	Note
success	function	-	Not required	Callback function for successful interface invocation
fail	function	-	Not required	Callback function for unsuccessful interface invocation
complete	function	-	Not required	Callback function executed upon the completion of the interface invocation (both successful and unsuccessful invocations)

Error Code

Error	Error Message	Note

©2013-2022 Tencent Cloud. All rights reserved.

Codes		
13000	The device does not support NFC	-
13001	The system's NFC switch is not activated.	-
13010	An unknown error occurred.	-
13019	user is not authorized	User Authorization Denied
13011	invalid parameter	Invalid parameter
13012	parse NdefMessage failed	Failed to parse the parameter into NdefMessage.
13021	NFC discovery already started	NFC scanning has already commenced.
13018	NFC discovery has not started	Attempted to halt NFC scanning without initiating it.
13022	Tech already connected	The tag has been successfully connected.
13023	Tech has not connected	Attempted to disconnect without a tag being connected.
13013	NFC tag has not been discovered	No NFC tag was detected.
13014	invalid tech	Invalid tag technology.
13015	unavailable tech	Failed to acquire the corresponding technology from the tag.
13024	function not support	The current tag technology does not support this feature.
13017	system internal error	The related read-write operation has failed.
13016	connect fail	Connection failed

.offNdefMessage

The method is utilized as Ndef.onNdefMessage(function callback).

Function Description: Cancels the monitoring of Ndef messages.

Parameters and Description: function callback.

©2013-2022 Tencent Cloud. All rights reserved.

.onNdefMessage

The method is employed as Ndef.offNdefMessage(function callback).

Function Description: Monitors Ndef messages.

Parameters and Description: function callback.

.writeNdefMessage

The method is employed as Ndef.writeNdefMessage(Object object).

Function Description: Overwrites the content of Ndef tags.

Parameters and Description: Object object.

Attributes	Types	Default value	Required	Note
uris	Array	-	Not required	URI array.
texts	Array	-	Not required	Text array.
records	Array	-	Not required	An array of binary objects, requiring specification of id, type, and payload (all of which are of the ArrayBuffer type).
success	function	-	Not required	Callback function for successful interface invocation
fail	function	-	Not required	Callback function for unsuccessful interface invocation
complete	function	-	Not required	Callback function executed upon the completion of the interface invocation (both successful and unsuccessful invocations)

Error Code

Error	Error Message	Note

©2013-2022 Tencent Cloud. All rights reserved.

Codes		
13000	The device does not support NFC	-
13001	The system's NFC switch is not activated.	-
13010	An unknown error occurred.	-
13019	user is not authorized	User Authorization Denied
13011	invalid parameter	Invalid parameter
13012	parse NdefMessage failed	Failed to parse the parameter into NdefMessage.
13021	NFC discovery already started	NFC scanning has already commenced.
13018	NFC discovery has not started	Attempted to halt NFC scanning without initiating it.
13022	Tech already connected	The tag has been successfully connected.
13023	Tech has not connected	Attempted to disconnect without a tag being connected.
13013	NFC tag has not been discovered	No NFC tag was detected.
13014	invalid tech	Invalid tag technology.
13015	unavailable tech	Failed to acquire the corresponding technology from the tag.
13024	function not support	The current tag technology does not support this feature.
13017	system internal error	The related read-write operation has failed.
13016	connect fail	Connection failed

©2013-2022 Tencent Cloud. All rights reserved.

NFC—NfcA

Last updated：2024-07-16 17:55:45

.connect

The method is utilized as NfcA.connect().

Function Description: Establishes a connection with the NFC tag.

Parameters and Description: Object object.

Attributes	Types	Default value	Required	Note
success	function	-	Not required	Callback function for successful interface invocation
fail	function	-	Not required	Callback function for unsuccessful interface invocation
complete	function	-	Not required	Callback function executed at the conclusion of the interface invocation (both successful and unsuccessful invocations)

Error Code

Error Codes	Error Message	Note
13000	The device does not support NFC	-
13001	The system's NFC switch is not activated.	-
13010	An unknown error occurred.	-
13019	user is not authorized	User Authorization Denied
13011	invalid parameter	Invalid parameter
13012	parse NdefMessage failed	Failed to parse the parameter into NdefMessage.
13021	NFC discovery already started	NFC scanning has already commenced.

©2013-2022 Tencent Cloud. All rights reserved.

13018	NFC discovery has not started	Attempted to halt NFC scanning without initiating it.
13022	Tech already connected	The tag has been successfully connected.
13023	Tech has not connected	Attempted to disconnect without a tag being connected.
13013	NFC tag has not been discovered	No NFC tag was detected.
13014	invalid tech	Invalid tag technology.
13015	unavailable tech	Failed to acquire the corresponding technology from the tag.
13024	function not support	The current tag technology does not support this feature.
13017	system internal error	The related read-write operation has failed.
13016	connect fail	Connection failed

.close

The method is utilized as NfcA.close().

Function Description: Disconnect.

Parameters and Description: Object object

Attributes	Types	Default value	Required	Note
success	function	-	Not required	Callback function for successful interface invocation
fail	function	-	Not required	Callback function for unsuccessful interface invocation
complete	function	-	Not required	Callback function executed upon the completion of the interface invocation (both successful and unsuccessful invocations)

Error Code

©2013-2022 Tencent Cloud. All rights reserved.

Error Codes	Error Message	Note
13000	The device does not support NFC	-
13001	The system's NFC switch is not activated.	-
13010	An unknown error occurred.	-
13019	user is not authorized	User Authorization Denied
13011	invalid parameter	Invalid parameter
13012	parse NdefMessage failed	Failed to parse the parameter into NdefMessage.
13021	NFC discovery already started	NFC scanning has already commenced.
13018	NFC discovery has not started	Attempted to halt NFC scanning without initiating it.
13022	Tech already connected	The tag has been successfully connected.
13023	Tech has not connected	Attempted to disconnect without a tag being connected.
13013	NFC tag has not been discovered	No NFC tag was detected.
13014	invalid tech	Invalid tag technology.
13015	unavailable tech	Failed to acquire the corresponding technology from the tag.
13024	function not support	The current tag technology does not support this feature.
13017	system internal error	The related read-write operation has failed.
13016	connect fail	Connection failed

.setTimeout

The method is employed as NfcA.setTimeout(Object object).

Function Description: Setting the timeout duration.

©2013-2022 Tencent Cloud. All rights reserved.

Parameters and Description: Object object.

Attributes	Types	Default value	Required	Note
timeout	number	-	Supported	Setting the timeout duration (ms)
success	function	-	Not required	Callback function for successful interface invocation
fail	function	-	Not required	Callback function for unsuccessful interface invocation
complete	function	-	Not required	Callback function executed upon the completion of the interface invocation (both successful and unsuccessful invocations)

Error Code

Error Codes	Error Message	Note
13000	The device does not support NFC	-
13001	The system's NFC switch is not activated.	-
13010	An unknown error occurred.	-
13019	user is not authorized	User Authorization Denied
13011	invalid parameter	Invalid parameter
13012	parse NdefMessage failed	Failed to parse the parameter into NdefMessage.
13021	NFC discovery already started	NFC scanning has already commenced.
13018	NFC discovery has not started	Attempted to halt NFC scanning without initiating it.
13022	Tech already connected	The tag has been successfully connected.
13023	Tech has not connected	Attempted to disconnect without a tag being connected.
13013	NFC tag has not been discovered	No NFC tag was detected.
13014	invalid tech	Invalid tag technology.
13015	unavailable tech	Failed to acquire the corresponding technology from the tag.

©2013-2022 Tencent Cloud. All rights reserved.

13024	function not support	The current tag technology does not support this feature.
13017	system internal error	The related read-write operation has failed.
13016	connect fail	Connection failed

.isConnected

The method is utilized as NfcA.isConnected().

Note:

This interface has been deprecated; developers can maintain the connection status independently.

Function Description: Verifies if a connection has been established.

Parameters and Description: Object object.

Attributes	Types	Default value	Required	Note
success	function	-	Not required	Callback function for successful interface invocation
fail	function	-	Not required	Callback function for unsuccessful interface invocation
complete	function	-	Not required	Callback function executed upon the completion of the interface invocation (both successful and unsuccessful invocations)

Error Code

Error Codes	Error Message	Note
13000	The device does not support NFC	-
13001	The system's NFC switch is not activated.	-
13010	An unknown error occurred.	-
13019	user is not authorized	User Authorization Denied

©2013-2022 Tencent Cloud. All rights reserved.

13011	invalid parameter	Invalid parameter
13012	parse NdefMessage failed	Failed to parse the parameter into NdefMessage.
13021	NFC discovery already started	NFC scanning has already commenced.
13018	NFC discovery has not started	Attempted to halt NFC scanning without initiating it.
13022	Tech already connected	The tag has been successfully connected.
13023	Tech has not connected	Attempted to disconnect without a tag being connected.
13013	NFC tag has not been discovered	No NFC tag was detected.
13014	invalid tech	Invalid tag technology.
13015	unavailable tech	Failed to acquire the corresponding technology from the tag.
13024	function not support	The current tag technology does not support this feature.
13017	system internal error	The related read-write operation has failed.
13016	connect fail	Connection failed

.getMaxTransceiveLength

The method is utilized as NfcA.getMaxTransceiveLength().

Function Description: Retrieves the maximum transmission length.

Parameters and Description: Object object.

Attributes	Types	Default value	Required	Note
success	function	-	Not required	Callback function for successful interface invocation
fail	function	-	Not required	Callback function for unsuccessful interface invocation
complete	function	-	Not required	Callback function executed upon the completion of the interface invocation (both successful and

©2013-2022 Tencent Cloud. All rights reserved.

				unsuccessful invocations)

object.success Callback Function Parameters: Object res.

Attributes	Types	Note
length	number	Maximum Transmission Length

Error Codes

Error Codes	Error Message	Note
13000	The device does not support NFC	-
13001	The system's NFC switch is not activated.	-
13010	An unknown error occurred.	-
13019	user is not authorized	User Authorization Denied
13011	invalid parameter	Invalid parameter
13012	parse NdefMessage failed	Failed to parse the parameter into NdefMessage.
13021	NFC discovery already started	NFC scanning has already commenced.
13018	NFC discovery has not started	Attempted to halt NFC scanning without initiating it.
13022	Tech already connected	The tag has been successfully connected.
13023	Tech has not connected	Attempted to disconnect without a tag being connected.
13013	NFC tag has not been discovered	No NFC tag was detected.
13014	invalid tech	Invalid tag technology.
13015	unavailable tech	Failed to acquire the corresponding technology from the tag.
13024	function not support	The current tag technology does not support this feature.
13017	system internal error	The related read-write operation has failed.

©2013-2022 Tencent Cloud. All rights reserved.

13016	connect fail	Connection failed

.transceive

The method is employed as NfcA.transceive(Object object).

Function Description: Transmits data.

Parameters and Description: Object object.

Attributes	Types	Default value	Required	Note
data	ArrayBuffer	-	Supported	Binary data required for transmission
success	function	-	Not required	Callback function for successful interface invocation
fail	function	-	Not required	Callback function for unsuccessful interface invocation
complete	function	-	Not required	Callback function executed upon the completion of the interface invocation (both successful and unsuccessful invocations)

object.success Callback Function Parameters: Object res.

Attributes	Types	Note
data	ArrayBuffer	-

Error Code

Error Codes	Error Message	Note
13000	The device does not support NFC	-
13001	The system's NFC switch is not activated.	-
13010	An unknown error occurred.	-
13019	user is not authorized	User Authorization Denied

©2013-2022 Tencent Cloud. All rights reserved.

13011	invalid parameter	Invalid parameter
13012	parse NdefMessage failed	Failed to parse the parameter into NdefMessage.
13021	NFC discovery already started	NFC scanning has already commenced.
13018	NFC discovery has not started	Attempted to halt NFC scanning without initiating it.
13022	Tech already connected	The tag has been successfully connected.
13023	Tech has not connected	Attempted to disconnect without a tag being connected.
13013	NFC tag has not been discovered	No NFC tag was detected.
13014	invalid tech	Invalid tag technology.
13015	unavailable tech	Failed to acquire the corresponding technology from the tag.
13024	function not support	The current tag technology does not support this feature.
13017	system internal error	The related read-write operation has failed.
13016	connect fail	Connection failed

.getAtqa

The method is utilized as NfcA.getAtqa().

Function Description: Retrieves ATQA information.

Parameters and Description: Object object.

Attributes	Types	Default value	Required	Note
success	function	-	Not required	Callback function for successful interface invocation
fail	function	-	Not required	Callback function for unsuccessful interface invocation

©2013-2022 Tencent Cloud. All rights reserved.

complete	function	-	Not required	Callback function executed upon the completion of the interface invocation (both successful and unsuccessful invocations)

object.success Callback Function Parameters: Object res.

Attributes	Types	Note
atqa	ArrayBuffer	Returns ATQA/SENS_RES data.

Error Code

Error Codes	Error Message	Note
13000	The device does not support NFC	-
13001	The system's NFC switch is not activated.	-
13010	An unknown error occurred.	-
13019	user is not authorized	User Authorization Denied
13011	invalid parameter	Invalid parameter
13012	parse NdefMessage failed	Failed to parse the parameter into NdefMessage.
13021	NFC discovery already started	NFC scanning has already commenced.
13018	NFC discovery has not started	Attempted to halt NFC scanning without initiating it.
13022	Tech already connected	The tag has been successfully connected.
13023	Tech has not connected	Attempted to disconnect without a tag being connected.
13013	NFC tag has not been discovered	No NFC tag was detected.
13014	invalid tech	Invalid tag technology.
13015	unavailable tech	Failed to acquire the corresponding technology from the tag.
13024	function not support	The current tag technology does not support this feature.

©2013-2022 Tencent Cloud. All rights reserved.

13017	system internal error	The related read-write operation has failed.
13016	connect fail	Connection failed

.getSak

The method is employed as NfcA.getSak().

Function Description: Acquires SAK information.

Parameters and Description: Object object.

Attributes	Types	Default value	Required	Note
success	function	-	Not required	Callback function for successful interface invocation
fail	function	-	Not required	Callback function for unsuccessful interface invocation
complete	function	-	Not required	Callback function executed upon the completion of the interface invocation (both successful and unsuccessful invocations)

object.success Callback Function Parameters: Object res.

Attributes	Types	Note
sak	number	Returns SAK/SEL_RES data.

Error Code

Error Codes	Error Message	Note
13000	The device does not support NFC	-
13001	The system's NFC switch is not activated.	-
13010	An unknown error occurred.	-
13019	user is not authorized	User Authorization Denied

©2013-2022 Tencent Cloud. All rights reserved.

13011	invalid parameter	Invalid parameter
13012	parse NdefMessage failed	Failed to parse the parameter into NdefMessage.
13021	NFC discovery already started	NFC scanning has already commenced.
13018	NFC discovery has not started	Attempted to halt NFC scanning without initiating it.
13022	Tech already connected	The tag has been successfully connected.
13023	Tech has not connected	Attempted to disconnect without a tag being connected.
13013	NFC tag has not been discovered	No NFC tag was detected.
13014	invalid tech	Invalid tag technology.
13015	unavailable tech	Failed to acquire the corresponding technology from the tag.
13024	function not support	The current tag technology does not support this feature.
13017	system internal error	The related read-write operation has failed.
13016	connect fail	Connection failed

©2013-2022 Tencent Cloud. All rights reserved.

NFC—NfcB

Last updated：2024-07-16 17:56:19

.connect

The method is utilized via NfcB.connect().

Function Description: Establishes a connection with the NFC tag.

Parameters and Description: Object - object.

Attributes	Types	Default value	Required	Note
success	function	-	Not required	Callback function for successful interface invocation
fail	function	-	Not required	Callback function for unsuccessful interface invocation
complete	function	-	Not required	Callback function executed upon the completion of the interface invocation (both successful and unsuccessful invocations)

Error Code

Error Codes	Error Message	Note
13000	The device does not support NFC	-
13001	The system's NFC switch is not activated.	-
13010	An unknown error occurred.	-
13019	user is not authorized	User Authorization Denied
13011	invalid parameter	Invalid parameter
13012	parse NdefMessage failed	Failed to parse the parameter into NdefMessage.
13021	NFC discovery already started	NFC scanning has already commenced.

©2013-2022 Tencent Cloud. All rights reserved.

13018	NFC discovery has not started	Attempted to halt NFC scanning without initiating it.
13022	Tech already connected	The tag has been successfully connected.
13023	Tech has not connected	Attempted to disconnect without a tag being connected.
13013	NFC tag has not been discovered	No NFC tag was detected.
13014	invalid tech	Invalid tag technology.
13015	unavailable tech	Failed to acquire the corresponding technology from the tag.
13024	function not support	The current tag technology does not support this feature.
13017	system internal error	The related read-write operation has failed.
13016	connect fail	Connection failed

.close

The method is utilized as NfcB.close().

Function Description: Disconnects the link.

Parameters and Description: Object object

Attributes	Types	Default value	Required	Note
success	function	-	Not required	Callback function for successful interface invocation
fail	function	-	Not required	Callback function for unsuccessful interface invocation
complete	function	-	Not required	Callback function executed upon the completion of the interface invocation (both successful and unsuccessful invocations)

Error Code

©2013-2022 Tencent Cloud. All rights reserved.

Error Codes	Error Message	Note
13000	The device does not support NFC	-
13001	The system's NFC switch is not activated.	-
13010	An unknown error occurred.	-
13019	user is not authorized	User Authorization Denied
13011	invalid parameter	Invalid parameter
13012	parse NdefMessage failed	Failed to parse the parameter into NdefMessage.
13021	NFC discovery already started	NFC scanning has already commenced.
13018	NFC discovery has not started	Attempted to halt NFC scanning without initiating it.
13022	Tech already connected	The tag has been successfully connected.
13023	Tech has not connected	Attempted to disconnect without a tag being connected.
13013	NFC tag has not been discovered	No NFC tag was detected.
13014	invalid tech	Invalid tag technology.
13015	unavailable tech	Failed to acquire the corresponding technology from the tag.
13024	function not support	The current tag technology does not support this feature.
13017	system internal error	The related read-write operation has failed.
13016	connect fail	Connection failed

.setTimeout

The method is employed as NfcB.setTimeout(Object object).

Function Description: Setting the timeout duration.

©2013-2022 Tencent Cloud. All rights reserved.

Parameters and Description: Object object

Attributes	Types	Default value	Required	Note
timeout	number	-	Supported	Setting the timeout duration (ms)
success	function	-	Not required	Callback function for successful interface invocation
fail	function	-	Not required	Callback function for unsuccessful interface invocation
complete	function	-	Not required	Callback function executed upon the completion of the interface invocation (both successful and unsuccessful invocations)

Error Code

Error Codes	Error Message	Note
13000	The device does not support NFC	-
13001	The system's NFC switch is not activated.	-
13010	An unknown error occurred.	-
13019	user is not authorized	User Authorization Denied
13011	invalid parameter	Invalid parameter
13012	parse NdefMessage failed	Failed to parse the parameter into NdefMessage.
13021	NFC discovery already started	NFC scanning has already commenced.
13018	NFC discovery has not started	Attempted to halt NFC scanning without initiating it.
13022	Tech already connected	The tag has been successfully connected.
13023	Tech has not connected	Attempted to disconnect without a tag being connected.
13013	NFC tag has not been discovered	No NFC tag was detected.
13014	invalid tech	Invalid tag technology.
13015	unavailable tech	Failed to acquire the corresponding technology

©2013-2022 Tencent Cloud. All rights reserved.

		from the tag.
13024	function not support	The current tag technology does not support this feature.
13017	system internal error	The related read-write operation has failed.
13016	connect fail	Connection failed

.isConnected

The method is utilized as NfcB.isConnected().

Function Description: Verifies if a connection has been established.

Parameters and Description: Object - object.

Attributes	Types	Default value	Required	Note
success	function	-	Not required	Callback function for successful interface invocation
fail	function	-	Not required	Callback function for unsuccessful interface invocation
complete	function	-	Not required	Callback function executed upon the completion of the interface invocation (both successful and unsuccessful invocations)

Error Code

Error Codes	Error Message	Note
13000	The device does not support NFC	-
13001	The system's NFC switch is not activated.	-
13010	An unknown error occurred.	-
13019	user is not authorized	User Authorization Denied
13011	invalid parameter	Invalid parameter

©2013-2022 Tencent Cloud. All rights reserved.

13012	parse NdefMessage failed	Failed to parse the parameter into NdefMessage.
13021	NFC discovery already started	NFC scanning has already commenced.
13018	NFC discovery has not started	Attempted to halt NFC scanning without initiating it.
13022	Tech already connected	The tag has been successfully connected.
13023	Tech has not connected	Attempted to disconnect without a tag being connected.
13013	NFC tag has not been discovered	No NFC tag was detected.
13014	invalid tech	Invalid tag technology.
13015	unavailable tech	Failed to acquire the corresponding technology from the tag.
13024	function not support	The current tag technology does not support this feature.
13017	system internal error	The related read-write operation has failed.
13016	connect fail	Connection failed

.getMaxTransceiveLength

The method is utilized as NfcB.getMaxTransceiveLength().

Function Description: Retrieves the maximum transmission length.

Parameters and Description: Object - object.

Attributes	Types	Default value	Required	Note
success	function	-	Not required	Callback function for successful interface invocation
fail	function	-	Not required	Callback function for unsuccessful interface invocation
complete	function	-	Not required	Callback function executed upon the completion of the interface invocation (both successful and unsuccessful

©2013-2022 Tencent Cloud. All rights reserved.

				invocations)

object.success Callback Function Parameters: Object res.

Attributes	Types	Note
length	number	Maximum Transmission Length

Error Code

Error Codes	Error Message	Note
13000	The device does not support NFC	-
13001	The system's NFC switch is not activated.	-
13010	An unknown error occurred.	-
13019	user is not authorized	User Authorization Denied
13011	invalid parameter	Invalid parameter
13012	parse NdefMessage failed	Failed to parse the parameter into NdefMessage.
13021	NFC discovery already started	NFC scanning has already commenced.
13018	NFC discovery has not started	Attempted to halt NFC scanning without initiating it.
13022	Tech already connected	The tag has been successfully connected.
13023	Tech has not connected	Attempted to disconnect without a tag being connected.
13013	NFC tag has not been discovered	No NFC tag was detected.
13014	invalid tech	Invalid tag technology.
13015	unavailable tech	Failed to acquire the corresponding technology from the tag.
13024	function not support	The current tag technology does not support this feature.
13017	system internal error	The related read-write operation has failed.
13016	connect fail	Connection failed

©2013-2022 Tencent Cloud. All rights reserved.

.transceive

The method is employed as NfcB.transceive(Object object).

Function Description: Transmits data.

Parameters and Description: Object - object.

Attributes	Types	Default value	Required	Note
data	ArrayBuffer	-	Supported	Binary data required for transmission
success	function	-	Not required	Callback function for successful interface invocation
fail	function	-	Not required	Callback function for unsuccessful interface invocation
complete	function	-	Not required	Callback function executed upon the completion of the interface invocation (both successful and unsuccessful invocations)

object.success Callback Function Parameters: Object res.

Attributes	Types	Note
data	ArrayBuffer	-

Error Code

Error Codes	Error Message	Note
13000	The device does not support NFC	-
13001	The system's NFC switch is not activated.	-
13010	An unknown error occurred.	-
13019	user is not authorized	User Authorization Denied
13011	invalid parameter	Invalid parameter

©2013-2022 Tencent Cloud. All rights reserved.

13012	parse NdefMessage failed	Failed to parse the parameter into NdefMessage.
13021	NFC discovery already started	NFC scanning has already commenced.
13018	NFC discovery has not started	Attempted to halt NFC scanning without initiating it.
13022	Tech already connected	The tag has been successfully connected.
13023	Tech has not connected	Attempted to disconnect without a tag being connected.
13013	NFC tag has not been discovered	No NFC tag was detected.
13014	invalid tech	Invalid tag technology.
13015	unavailable tech	Failed to acquire the corresponding technology from the tag.
13024	function not support	The current tag technology does not support this feature.
13017	system internal error	The related read-write operation has failed.
13016	connect fail	Connection failed

©2013-2022 Tencent Cloud. All rights reserved.

NFC—NfcF

Last updated：2024-07-16 17:56:58

.connect

The method is utilized via NfcF.connect().

Function Description: Establishes a connection with the NFC tag.

Parameters and Description: Object - object.

Attributes	Types	Default value	Required	Note
success	function	-	Not required	Callback function for successful interface invocation
fail	function	-	Not required	Callback function for unsuccessful interface invocation
complete	function	-	Not required	Callback function executed upon the completion of the interface invocation (both successful and unsuccessful invocations)

Error Code

Error Codes	Error Message	Note
13000	The device does not support NFC	-
13001	The system's NFC switch is not activated.	-
13010	An unknown error occurred.	-
13019	user is not authorized	User Authorization Denied
13011	invalid parameter	Invalid parameter
13012	parse NdefMessage failed	Failed to parse the parameter into NdefMessage.
13021	NFC discovery already started	NFC scanning has already commenced.

©2013-2022 Tencent Cloud. All rights reserved.

13018	NFC discovery has not started	Attempted to halt NFC scanning without initiating it.
13022	Tech already connected	The tag has been successfully connected.
13023	Tech has not connected	Attempted to disconnect without a tag being connected.
13013	NFC tag has not been discovered	No NFC tag was detected.
13014	invalid tech	Invalid tag technology.
13015	unavailable tech	Failed to acquire the corresponding technology from the tag.
13024	function not support	The current tag technology does not support this feature.
13017	system internal error	The related read-write operation has failed.
13016	connect fail	Connection failed

.close

The method is utilized as NfcF.close().

Function Description: Disconnects the link.

Parameters and Description: Object - object.

Attributes	Types	Default value	Required	Note
success	function	-	Not required	Callback function for successful interface invocation
fail	function	-	Not required	Callback function for unsuccessful interface invocation
complete	function	-	Not required	Callback function executed upon the completion of the interface invocation (both successful and unsuccessful invocations)

©2013-2022 Tencent Cloud. All rights reserved.

Error Code

Error Codes	Error Message	Note
13000	The device does not support NFC	-
13001	The system's NFC switch is not activated.	-
13010	An unknown error occurred.	-
13019	user is not authorized	User Authorization Denied
13011	invalid parameter	Invalid parameter
13012	parse NdefMessage failed	Failed to parse the parameter into NdefMessage.
13021	NFC discovery already started	NFC scanning has already commenced.
13018	NFC discovery has not started	Attempted to halt NFC scanning without initiating it.
13022	Tech already connected	The tag has been successfully connected.
13023	Tech has not connected	Attempted to disconnect without a tag being connected.
13013	NFC tag has not been discovered	No NFC tag was detected.
13014	invalid tech	Invalid tag technology.
13015	unavailable tech	Failed to acquire the corresponding technology from the tag.
13024	function not support	The current tag technology does not support this feature.
13017	system internal error	The related read-write operation has failed.
13016	connect fail	Connection failed

.setTimeout

The method is employed as NfcF.setTimeout(Object object).

©2013-2022 Tencent Cloud. All rights reserved.

Function Description: Setting the timeout duration.

Parameters and Description: Object - object.

Attributes	Types	Default value	Required	Note
timeout	number	-	Supported	Setting the timeout duration (ms).
success	function	-	Not required	Callback function for successful interface invocation
fail	function	-	Not required	Callback function for unsuccessful interface invocation
complete	function	-	Not required	Callback function executed upon the completion of the interface invocation (both successful and unsuccessful invocations)

Error Code

Error Codes	Error Message	Note
13000	The device does not support NFC	-
13001	The system's NFC switch is not activated.	-
13010	An unknown error occurred.	-
13019	user is not authorized	User Authorization Denied
13011	invalid parameter	Invalid parameter
13012	parse NdefMessage failed	Failed to parse the parameter into NdefMessage.
13021	NFC discovery already started	NFC scanning has already commenced.
13018	NFC discovery has not started	Attempted to halt NFC scanning without initiating it.
13022	Tech already connected	The tag has been successfully connected.
13023	Tech has not connected	Attempted to disconnect without a tag being connected.
13013	NFC tag has not been discovered	No NFC tag was detected.

©2013-2022 Tencent Cloud. All rights reserved.

13014	invalid tech	Invalid tag technology.
13015	unavailable tech	Failed to acquire the corresponding technology from the tag.
13024	function not support	The current tag technology does not support this feature.
13017	system internal error	The related read-write operation has failed.
13016	connect fail	Connection failed

.isConnected

The method is utilized as NfcF.isConnected().

Function Description: Verifies if a connection has been established.

Parameters and Description: Object - object.

Attributes	Types	Default value	Required	Note
success	function	-	Not required	Callback function for successful interface invocation
fail	function	-	Not required	Callback function for unsuccessful interface invocation
complete	function	-	Not required	Callback function executed upon the completion of the interface invocation (both successful and unsuccessful invocations)

Error Code

Error Codes	Error Message	Note
13000	The device does not support NFC	-
13001	The system's NFC switch is not activated.	-
13010	An unknown error occurred.	-

©2013-2022 Tencent Cloud. All rights reserved.

13019	user is not authorized	User Authorization Denied
13011	invalid parameter	Invalid parameter
13012	parse NdefMessage failed	Failed to parse the parameter into NdefMessage.
13021	NFC discovery already started	NFC scanning has already commenced.
13018	NFC discovery has not started	Attempted to halt NFC scanning without initiating it.
13022	Tech already connected	The tag has been successfully connected.
13023	Tech has not connected	Attempted to disconnect without a tag being connected.
13013	NFC tag has not been discovered	No NFC tag was detected.
13014	invalid tech	Invalid tag technology.
13015	unavailable tech	Failed to acquire the corresponding technology from the tag.
13024	function not support	The current tag technology does not support this feature.
13017	system internal error	The related read-write operation has failed.
13016	connect fail	Connection failed

.getMaxTransceiveLength

The method is employed as NfcF.getMaxTransceiveLength().

Function Description: Retrieves the maximum transmission length.

Parameters and Description: Object - object.

Attributes	Types	Default value	Required	Note
success	function	-	Not required	Callback function for successful interface invocation
fail	function	-	Not required	Callback function for unsuccessful interface invocation

©2013-2022 Tencent Cloud. All rights reserved.

complete	function	-	Not required	Callback function executed upon the completion of the interface invocation (both successful and unsuccessful invocations)

object.success Callback Function Parameters: Object res.

Attributes	Types	Note
length	number	Maximum Transmission Length

Error Code

Error Codes	Error Message	Note
13000	The device does not support NFC	-
13001	The system's NFC switch is not activated.	-
13010	An unknown error occurred.	-
13019	user is not authorized	User Authorization Denied
13011	invalid parameter	Invalid parameter
13012	parse NdefMessage failed	Failed to parse the parameter into NdefMessage.
13021	NFC discovery already started	NFC scanning has already commenced.
13018	NFC discovery has not started	Attempted to halt NFC scanning without initiating it.
13022	Tech already connected	The tag has been successfully connected.
13023	Tech has not connected	Attempted to disconnect without a tag being connected.
13013	NFC tag has not been discovered	No NFC tag was detected.
13014	invalid tech	Invalid tag technology.
13015	unavailable tech	Failed to acquire the corresponding technology from the tag.
13024	function not support	The current tag technology does not support this feature.

©2013-2022 Tencent Cloud. All rights reserved.

13017	system internal error	The related read-write operation has failed.
13016	connect fail	Connection failed

.transceive

The method is utilized as NfcF.transceive(Object object).

Function Description: Transmits data.

Parameters and Description: Object - object.

Attributes	Types	Default value	Required	Note
data	ArrayBuffer	-	Supported	Binary data required for transmission
success	function	-	Not required	Callback function for successful interface invocation
fail	function	-	Not required	Callback function for unsuccessful interface invocation
complete	function	-	Not required	Callback function executed upon the completion of the interface invocation (both successful and unsuccessful invocations)

object.success Callback Function Parameters: Object res.

Attributes	Types	Note
data	ArrayBuffer	-

Error Code

Error Codes	Error Message	Note
13000	The device does not support NFC	-
13001	The system's NFC switch is not activated.	-
13010	An unknown error occurred.	-

©2013-2022 Tencent Cloud. All rights reserved.

13019	user is not authorized	User Authorization Denied
13011	invalid parameter	Invalid parameter
13012	parse NdefMessage failed	Failed to parse the parameter into NdefMessage.
13021	NFC discovery already started	NFC scanning has already commenced.
13018	NFC discovery has not started	Attempted to halt NFC scanning without initiating it.
13022	Tech already connected	The tag has been successfully connected.
13023	Tech has not connected	Attempted to disconnect without a tag being connected.
13013	NFC tag has not been discovered	No NFC tag was detected.
13014	invalid tech	Invalid tag technology.
13015	unavailable tech	Failed to acquire the corresponding technology from the tag.
13024	function not support	The current tag technology does not support this feature.
13017	system internal error	The related read-write operation has failed.
13016	connect fail	Connection failed

©2013-2022 Tencent Cloud. All rights reserved.

NFC—NfcV

Last updated：2024-07-16 17:57:45

.connect

The method is utilized via NfcV.connect().

Function Description: Establishes a connection with the NFC tag.

Parameters and Description: Object object.

Attributes	Types	Default value	Required	Note
success	function	-	Not required	Callback function for successful interface invocation
fail	function	-	Not required	Callback function for unsuccessful interface invocation
complete	function	-	Not required	Callback function executed upon the completion of the interface invocation (both successful and unsuccessful invocations)

Error Code

Error Codes	Error Message	Note
13000	The device does not support NFC	-
13001	The system's NFC switch is not activated.	-
13010	An unknown error occurred.	-
13019	user is not authorized	User Authorization Denied
13011	invalid parameter	Invalid parameter
13012	parse NdefMessage failed	Failed to parse the parameter into NdefMessage.
13021	NFC discovery already started	NFC scanning has already commenced.

©2013-2022 Tencent Cloud. All rights reserved.

13018	NFC discovery has not started	Attempted to halt NFC scanning without initiating it.
13022	Tech already connected	The tag has been successfully connected.
13023	Tech has not connected	Attempted to disconnect without a tag being connected.
13013	NFC tag has not been discovered	No NFC tag was detected.
13014	invalid tech	Invalid tag technology.
13015	unavailable tech	Failed to acquire the corresponding technology from the tag.
13024	function not support	The current tag technology does not support this feature.
13017	system internal error	The related read-write operation has failed.
13016	connect fail	Connection failed

.close

The method is utilized as NfcV.close().

Function Description: Disconnects the link.

Parameters and Description: Object object.

Attributes	Types	Default value	Required	Note
success	function	-	Not required	Callback function for successful interface invocation
fail	function	-	Not required	Callback function for unsuccessful interface invocation
complete	function	-	Not required	Callback function executed upon the completion of the interface invocation (both successful and unsuccessful invocations)

©2013-2022 Tencent Cloud. All rights reserved.

Error Code

Error Codes	Error Message	Note
13000	The device does not support NFC	-
13001	The system's NFC switch is not activated.	-
13010	An unknown error occurred.	-
13019	user is not authorized	User Authorization Denied
13011	invalid parameter	Invalid parameter
13012	parse NdefMessage failed	Failed to parse the parameter into NdefMessage.
13021	NFC discovery already started	NFC scanning has already commenced.
13018	NFC discovery has not started	Attempted to halt NFC scanning without initiating it.
13022	Tech already connected	The tag has been successfully connected.
13023	Tech has not connected	Attempted to disconnect without a tag being connected.
13013	NFC tag has not been discovered	No NFC tag was detected.
13014	invalid tech	Invalid tag technology.
13015	unavailable tech	Failed to acquire the corresponding technology from the tag.
13024	function not support	The current tag technology does not support this feature.
13017	system internal error	The related read-write operation has failed.
13016	connect fail	Connection failed

.setTimeout

The method is employed as NfcV.setTimeout(Object object).

©2013-2022 Tencent Cloud. All rights reserved.

Function Description: Setting the timeout duration.

Parameters and Description: Object object

Attributes	Types	Default value	Required	Note
timeout	number	-	Supported	Setting the timeout duration (ms)
success	function	-	Not required	Callback function for successful interface invocation
fail	function	-	Not required	Callback function for unsuccessful interface invocation
complete	function	-	Not required	Callback function executed upon the completion of the interface invocation (both successful and unsuccessful invocations)

Error Code

Error Codes	Error Message	Note
13000	The device does not support NFC	-
13001	The system's NFC switch is not activated.	-
13010	An unknown error occurred.	-
13019	user is not authorized	User Authorization Denied
13011	invalid parameter	Invalid parameter
13012	parse NdefMessage failed	Failed to parse the parameter into NdefMessage.
13021	NFC discovery already started	NFC scanning has already commenced.
13018	NFC discovery has not started	Attempted to halt NFC scanning without initiating it.
13022	Tech already connected	The tag has been successfully connected.
13023	Tech has not connected	Attempted to disconnect without a tag being connected.
13013	NFC tag has not been discovered	No NFC tag was detected.

©2013-2022 Tencent Cloud. All rights reserved.

13014	invalid tech	Invalid tag technology.
13015	unavailable tech	Failed to acquire the corresponding technology from the tag.
13024	function not support	The current tag technology does not support this feature.
13017	system internal error	The related read-write operation has failed.
13016	connect fail	Connection failed

.isConnected

The method is utilized as NfcV.isConnected()

Note:

This interface has been deprecated; developers can maintain the connection status independently.

Function Description: Verifies if a connection has been established.

Parameters and Description: Object object.

Attributes	Types	Default value	Required	Note
success	function	-	Not required	Callback function for successful interface invocation
fail	function	-	Not required	Callback function for unsuccessful interface invocation
complete	function	-	Not required	Callback function executed upon the completion of the interface invocation (both successful and unsuccessful invocations)

Error Code

Error Codes	Error Message	Note
13000	The device does not support NFC	-
13001	The system's NFC switch is not activated.	-
13010	An unknown error occurred.	-

©2013-2022 Tencent Cloud. All rights reserved.

13019	user is not authorized	User Authorization Denied
13011	invalid parameter	Invalid parameter
13012	parse NdefMessage failed	Failed to parse the parameter into NdefMessage.
13021	NFC discovery already started	NFC scanning has already commenced.
13018	NFC discovery has not started	Attempted to halt NFC scanning without initiating it.
13022	Tech already connected	The tag has been successfully connected.
13023	Tech has not connected	Attempted to disconnect without a tag being connected.
13013	NFC tag has not been discovered	No NFC tag was detected.
13014	invalid tech	Invalid tag technology.
13015	unavailable tech	Failed to acquire the corresponding technology from the tag.
13024	function not support	The current tag technology does not support this feature.
13017	system internal error	The related read-write operation has failed.
13016	connect fail	Connection failed

.getMaxTransceiveLength

The method is employed as NfcV.getMaxTransceiveLength()

Function Description: Retrieves the maximum transmission length.

Parameters and Description: Object object.

Attributes	Types	Default value	Required	Note
success	function	-	Not required	Callback function for successful interface invocation
fail	function	-	Not required	Callback function for unsuccessful interface invocation

©2013-2022 Tencent Cloud. All rights reserved.

complete	function	-	Not required	Callback function executed upon the completion of the interface invocation (both successful and unsuccessful invocations)

object.success Callback Function Parameters: Object res.

Attributes	Types	Note
length	number	Maximum Transmission Length

Error Code

Error Codes	Error Message	Note
13000	The device does not support NFC	-
13001	The system's NFC switch is not activated.	-
13010	An unknown error occurred.	-
13019	user is not authorized	User Authorization Denied
13011	invalid parameter	Invalid parameter
13012	parse NdefMessage failed	Failed to parse the parameter into NdefMessage.
13021	NFC discovery already started	NFC scanning has already commenced.
13018	NFC discovery has not started	Attempted to halt NFC scanning without initiating it.
13022	Tech already connected	The tag has been successfully connected.
13023	Tech has not connected	Attempted to disconnect without a tag being connected.
13013	NFC tag has not been discovered	No NFC tag was detected.
13014	invalid tech	Invalid tag technology.
13015	unavailable tech	Failed to acquire the corresponding technology from the tag.
13024	function not support	The current tag technology does not support this feature.
13017	system internal error	The related read-write operation has failed.

©2013-2022 Tencent Cloud. All rights reserved.

13016	connect fail	Connection failed

.transceive

The method is utilized as NfcV.transceive(Object object)

Function Description: Transmits data.

Parameters and Description: Object object.

Attributes	Types	Default value	Required	Note
data	ArrayBuffer	-	Supported	Binary data required for transmission
success	function	-	Not required	Callback function for successful interface invocation
fail	function	-	Not required	Callback function for unsuccessful interface invocation
complete	function	-	Not required	Callback function executed upon the completion of the interface invocation (both successful and unsuccessful invocations)

object.success Callback Function Parameters: Object res.

Attributes	Types	Note
data	ArrayBuffer	-

Error Code

Error Codes	Error Message	Note
13000	The device does not support NFC	-
13001	The system's NFC switch is not activated.	-
13010	An unknown error occurred.	-

©2013-2022 Tencent Cloud. All rights reserved.

13019	user is not authorized	User Authorization Denied
13011	invalid parameter	Invalid parameter
13012	parse NdefMessage failed	Failed to parse the parameter into NdefMessage.
13021	NFC discovery already started	NFC scanning has already commenced.
13018	NFC discovery has not started	Attempted to halt NFC scanning without initiating it.
13022	Tech already connected	The tag has been successfully connected.
13023	Tech has not connected	Attempted to disconnect without a tag being connected.
13013	NFC tag has not been discovered	No NFC tag was detected.
13014	invalid tech	Invalid tag technology.
13015	unavailable tech	Failed to acquire the corresponding technology from the tag.
13024	function not support	The current tag technology does not support this feature.
13017	system internal error	The related read-write operation has failed.
13016	connect fail	Connection failed

©2013-2022 Tencent Cloud. All rights reserved.

WiFi

Last updated：2024-07-16 17:58:12

onWifiConnected

The method of utilizing this API is wx.onWifiConnected(function listener).

Note:

Compatibility:

Android: Supported.

iOS: Supported.

Function Description: Monitors the event of connecting to a Wi-Fi network.

Parameters and Description: function listener, the listener function for the event of connecting to Wi-Fi, with the following arguments Object res:

Attributes	Types	Note
wifi	WifiInfo	Wi-Fi Information

offWifiConnected

The method of utilizing this API is wx.onWifiConnected(function listener).

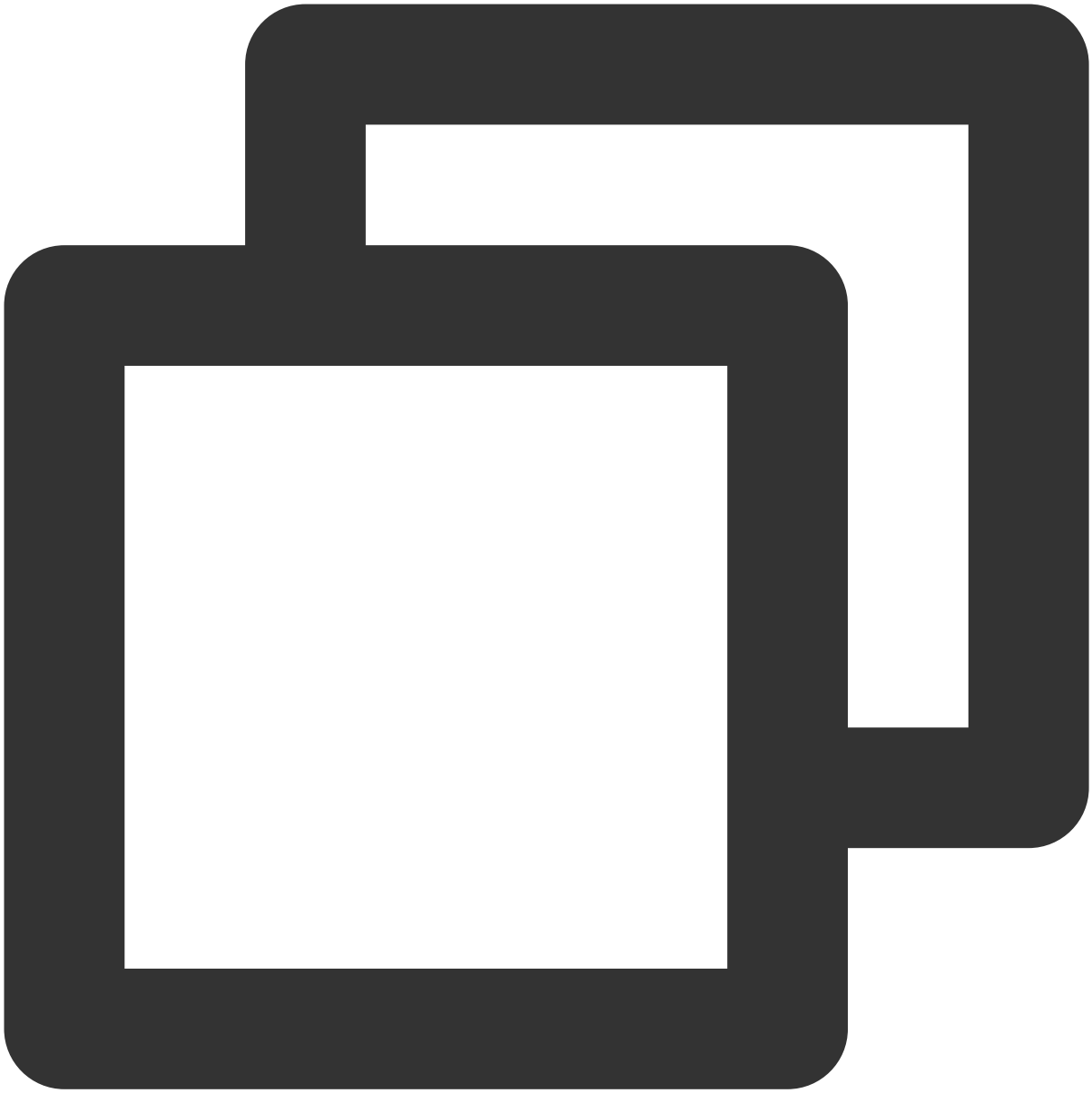
Function Description: Monitors the event of connecting to a Wi-Fi network.

Parameters and Description: function listener, onWifiConnected The listener function passed in. Not passing this parameter removes all listener functions.

Attributes	Types	Note
wifi	WifiInfo	Wi-Fi Information

Sample code

©2013-2022 Tencent Cloud. All rights reserved.



```
const listener = function (res) { console.log(res) }

wx.onWifiConnected(listener)
wx.offWifiConnected(listener) // Pass in the same function object as the listener.
```

stopWifi

©2013-2022 Tencent Cloud. All rights reserved.

The usage method for this API is wx.stopWifi(Object object)

Function Description: Deactivates the Wi-Fi module.

Parameters and Description: Object object.

Attributes	Types	Default value	Required	Note
success	function	-	Not required	Callback function for successful interface invocation
fail	function	-	Not required	Callback function for unsuccessful interface invocation
complete	function	-	Not required	Callback function executed upon the completion of the interface invocation (both successful and unsuccessful invocations)

Error Code

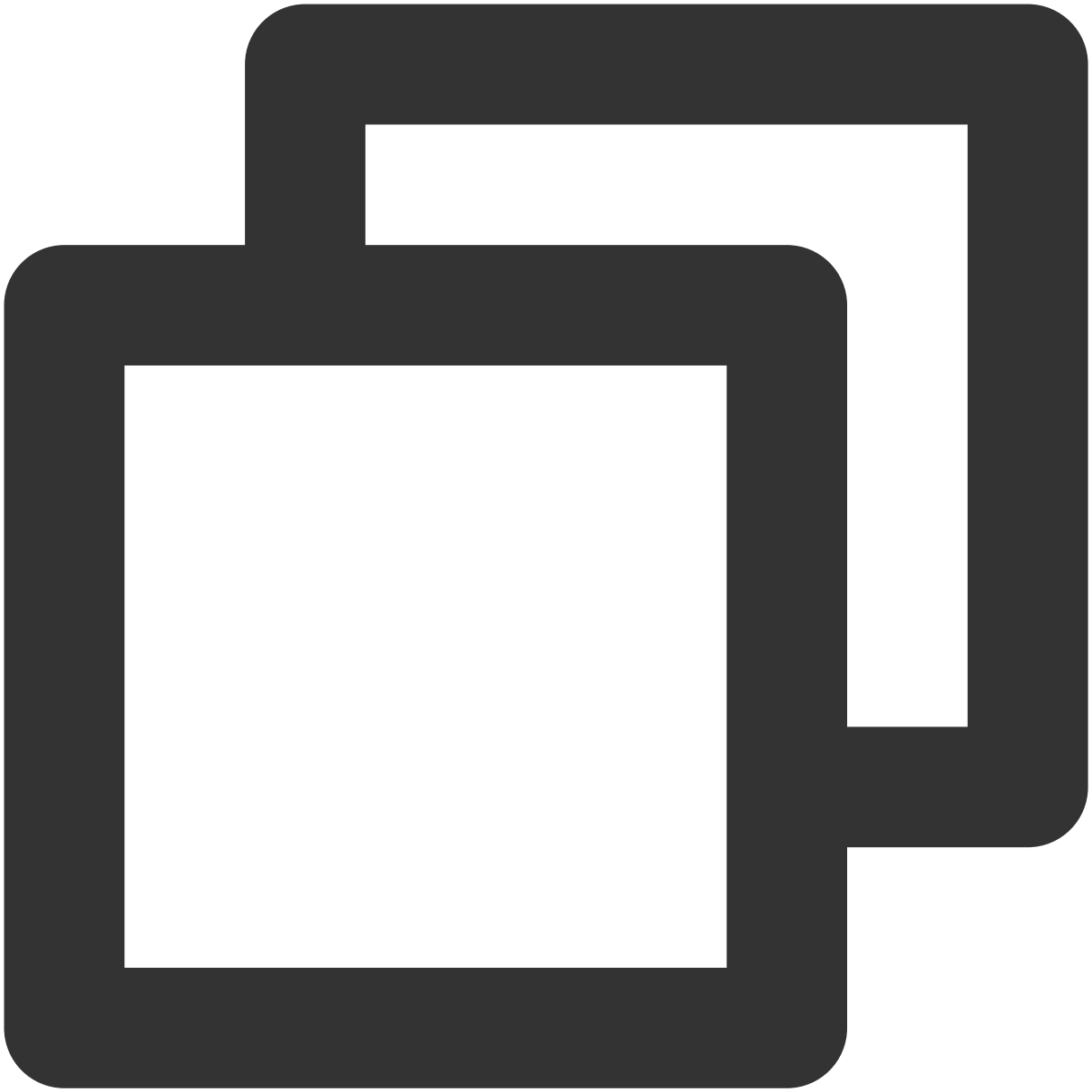
Error Codes	Error Message	Note
0	ok	Normal
12000	not init	The startWifi interface was not invoked beforehand.
12001	system not support	The current system does not support the relevant capabilities.
12002	password error Wi-Fi	The password is incorrect.
12003	connection timeout	Connection timeout, supported only on Android.
12004	duplicate request	Repeated connection to Wi-Fi.
12005	wifi not turned on	Exclusive to Android, the Wi-Fi switch has not been turned on.
12006	gps not turned on	Exclusive to Android, the GPS location switch has not been turned on.
12007	user denied	The user has denied authorization to connect to Wi-Fi.
12008	invalid SSID	Invalid SSID.
12009	system config err	The system operator configuration has denied the

©2013-2022 Tencent Cloud. All rights reserved.

		connection to Wi-Fi.
12010	system internal error	System miscellaneous error, it is necessary to print the specific error reason in errmsg.
12011	weapp in background	The application is unable to configure Wi-Fi while running in the background.
12013	wifi config may be expired	The system's saved Wi-Fi configuration has expired. It is recommended to forget the Wi-Fi and retry. This is only supported on Android.
12014	invalid WEP / WPA password	Specific to iOS, the WEP/WPA password is invalid.

Example Code

©2013-2022 Tencent Cloud. All rights reserved.



```
wx.stopWifi({
  success (res) {
    console.log(res.errMsg)
  }
})
```

startWifi

©2013-2022 Tencent Cloud. All rights reserved.

The usage method for this API is wx.startWifi(Object object).

Function Description: Initializes the Wi-Fi module.

Parameters and Description: Object object.

Attributes	Types	Default value	Required	Note
success	function	-	Not required	Callback function for successful interface invocation
fail	function	-	Not required	Callback function for unsuccessful interface invocation
complete	function	-	Not required	Callback function executed upon the completion of the interface invocation (both successful and unsuccessful invocations)

Error Code

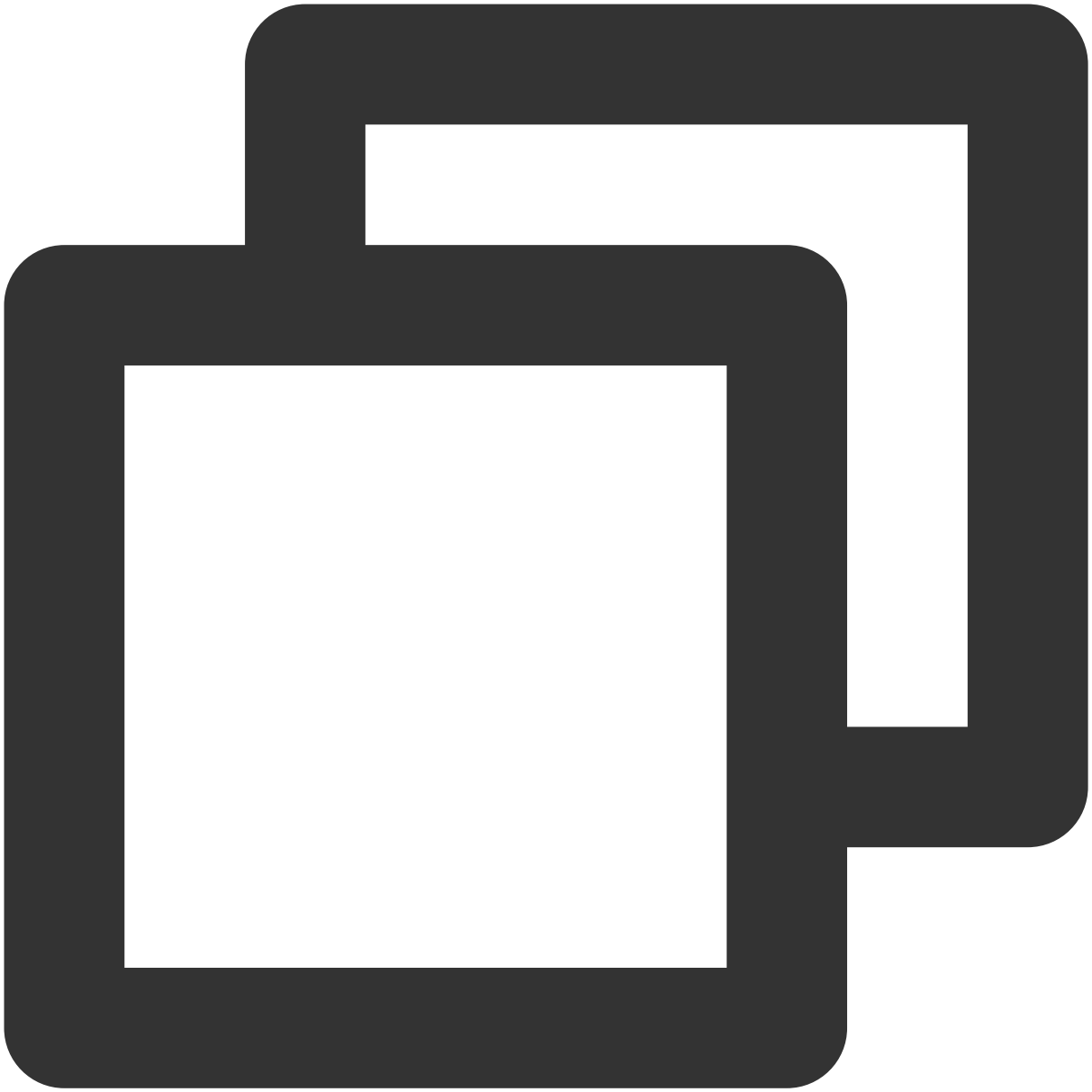
Error Codes	Error Message	Note
0	ok	Normal
12000	not init	The startWifi interface was not invoked beforehand.
12001	system not support	The current system does not support the relevant capabilities.
12002	password error Wi-Fi	The password is incorrect.
12003	connection timeout	Connection timeout, supported only on Android.
12004	duplicate request	Repeated connection to Wi-Fi.
12005	wifi not turned on	Exclusive to Android, the Wi-Fi switch has not been turned on.
12006	gps not turned on	Exclusive to Android, the GPS location switch has not been turned on.
12007	user denied	The user has denied authorization to connect to Wi-Fi.
12008	invalid SSID	Invalid SSID.
12009	system config err	The system operator configuration has denied the connection to Wi-Fi.

©2013-2022 Tencent Cloud. All rights reserved.

12010	system internal error	System miscellaneous error, it is necessary to print the specific error reason in errmsg.
12011	weapp in background	The application is unable to configure Wi-Fi while running in the background.
12013	wifi config may be expired	The system's saved Wi-Fi configuration has expired. It is recommended to forget the Wi-Fi and retry. This is only supported on Android.
12014	invalid WEP / WPA password	Specific to iOS, the WEP/WPA password is invalid.

Example Code

©2013-2022 Tencent Cloud. All rights reserved.



```
wx.startWifi({
  success (res) {
    console.log(res.errMsg)
  }
})
```

onGetWifiList

©2013-2022 Tencent Cloud. All rights reserved.

The usage method for this API is wx.onGetWifiList(function listener).

Function Description: Monitors the event of receiving Wi-Fi list data.
Parameters and Description: Object res parameter, function listener, the monitoring function for the event of receiving Wi-Fi list data.

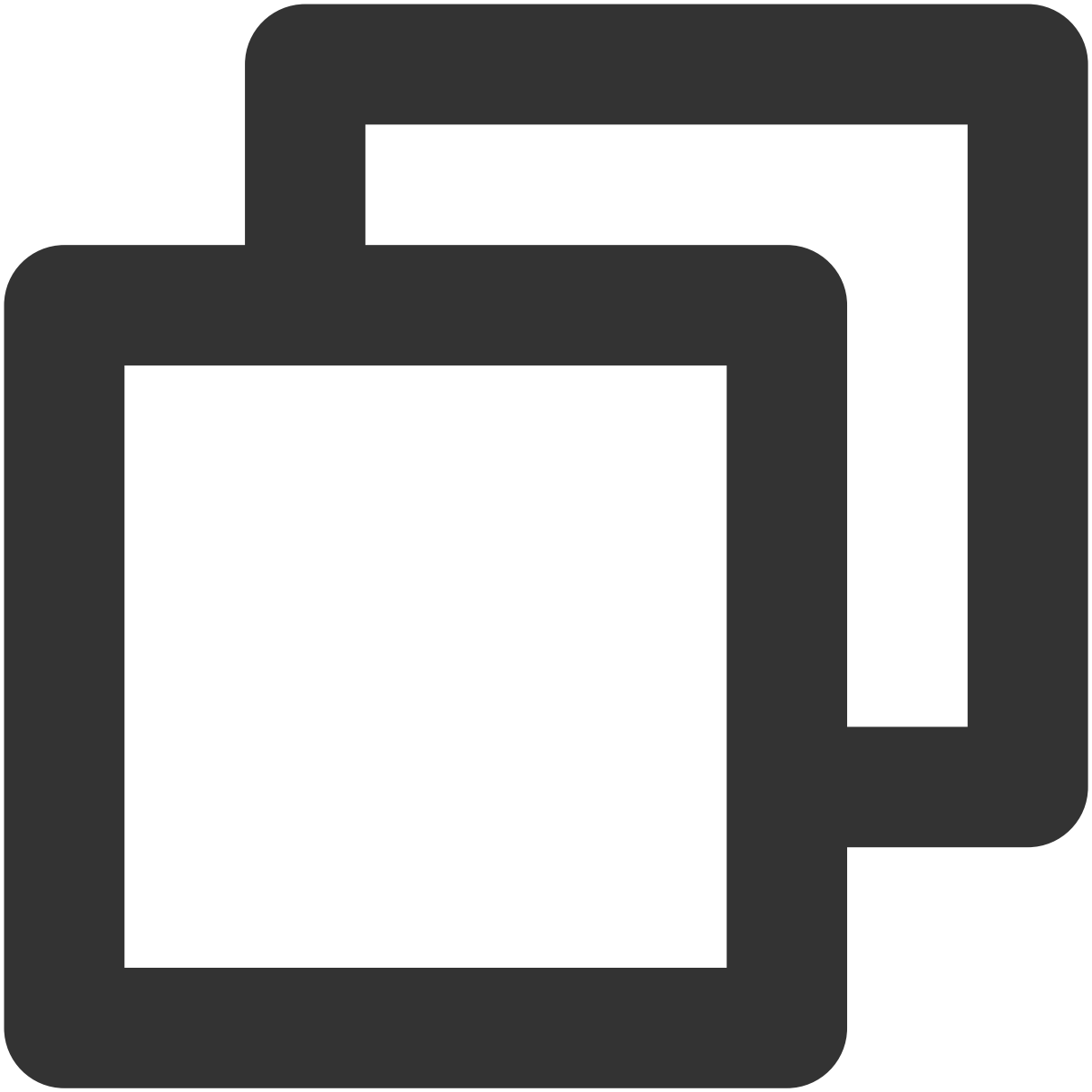
Attributes	Types	Note
wifiList	Array.<WifiInfo>	Wi-Fi List Data

offGetWifiList

The usage method for this API is wx.offGetWifiList(function listener).

Function Description: Removes the monitoring function for the event of receiving Wi-Fi list data.
Parameters and Description: Function listener, the monitoring function passed into onGetWifiList. If this parameter is not provided, all monitoring functions will be removed.
Sample code

©2013-2022 Tencent Cloud. All rights reserved.



```
const listener = function (res) { console.log(res) }

wx.onWifiConnected(listener)
wx.offWifiConnected(listener) // Pass in the same function object as the listener.
```

getConnectedWifi

©2013-2022 Tencent Cloud. All rights reserved.

The usage method for this API is wx.getConnectedWifi(Object object).

Function Description: Retrieves information about the currently connected Wi-Fi.

Parameters and Description: Object object

Attributes	Types	Default value	Required	Note
partialInfo	boolean	false	Not required	Whether partial Wi-Fi information needs to be returned.
success	function	-	Not required	Callback function for successful interface invocation
fail	function	-	Not required	Callback function for unsuccessful interface invocation
complete	function	-	Not required	Callback function executed upon the completion of the interface invocation (both successful and unsuccessful invocations)

object.success Callback Function Parameters: Object res

Attributes	Types	Note
wifi	WifiInfo	Wi-Fi Information

Error Code

Error Codes	Error Message	Note
0	ok	Normal
12000	not init	The startWifi interface was not invoked beforehand.
12001	system not support	The current system does not support the relevant capabilities.
12002	password error Wi-Fi	The password is incorrect.
12003	connection timeout	Connection timeout, supported only on Android.
12004	duplicate request	Repeated connection to Wi-Fi.
12005	wifi not turned on	Exclusive to Android, the Wi-Fi switch has not been turned on.

©2013-2022 Tencent Cloud. All rights reserved.

12006	gps not turned on	Exclusive to Android, the GPS location switch has not been turned on.
12007	user denied	The user has denied authorization to connect to Wi-Fi.
12008	invalid SSID	Invalid SSID.
12009	system config err	The system operator configuration has denied the connection to Wi-Fi.
12010	system internal error	System miscellaneous error, it is necessary to print the specific error reason in errmsg.
12011	weapp in background	The application is unable to configure Wi-Fi while running in the background.
12013	wifi config may be expired	The system's saved Wi-Fi configuration has expired. It is recommended to forget the Wi-Fi and retry. This is only supported on Android.
12014	invalid WEP / WPA password	Specific to iOS, the WEP/WPA password is invalid.

WifiInfo

Function Description: Wi-Fi Information.

Note:

For Android wx.connectWifi / wx.getConnectedWifi, if partialInfo:true is set, or if the wx.onWifiConnectedWithPartialInfo event is called, a WifiInfo object containing only the SSID property will be returned.

For iOS wx.getConnectedWifi, if partialInfo:true is set, a WifiInfo object containing only the SSID and BSSID properties will be returned. However, the user must enable the host client's location permissions for the results to be correctly returned.

In certain circumstances, even though the Wi-Fi may have successfully connected, an error may occur due to the inability to obtain a complete WifiInfo object. The specific error messages are `errCode:12010, errMsg: can't gain current wifi` or `no wifi is connected`. If developers do not require a complete WifiInfo object, they can resolve the error issue by adopting the aforementioned strategy.

Attributes

string SSID: The SSID of the Wi-Fi.

string BSSID: The BSSID of the Wi-Fi.

©2013-2022 Tencent Cloud. All rights reserved.

boolean secure: The security status of the Wi-Fi.

number signalStrength: The signal strength of the Wi-Fi. For Android, the value ranges from 0 to 100, while for iOS, the value ranges from 0 to 1. The larger the value, the stronger the signal strength.

number frequency: The frequency band of the Wi-Fi, measured in MHz.

©2013-2022 Tencent Cloud. All rights reserved.

Calendar

Last updated：2024-07-16 17:58:37

addPhoneCalendar

This API is used via wx.addPhoneCalendar(Object object)

Feature Description: Adds events into the system calendar.

Parameter and Description: Object.

Attribute	Type	Default value	Required	Description
title	string	-	Yes	Calendar Event Title
startTime	number	-	Yes	Unix timestamp of the start time
allDay	boolean	-	No	Whether it is an all-day event, which defaults to "false".
description	string	-	No	Event Description
location	string	-	No	Event Location
endTime	string	-	No	Unix timestamp of the end time
alarm	boolean	-	No	Whether to send a reminder, which defaults to "true".
alarmOffset	number	-	No	Reminder lead time, in seconds. The default value is 0, indicating a reminder at the start of the event.
success	function	-	No	Callback Function of Successful Interface Call
fail	function	-	No	Callback Function of Failing Interface Call
complete	function	-	No	Callback function executed upon the completion of the interface invocation (both successful and unsuccessful invocations)

addPhoneRepeatCalendar

©2013-2022 Tencent Cloud. All rights reserved.

This API is used via wx.addPhoneRepeatCalendar(Object object)

Feature Description: Adds recurring events into the system calendar.

Parameter and Description: Object.

Attribute	Type	Default value	Required	Description
title	string	-	Yes	Calendar Event Title
startTime	number	-	Yes	Unix timestamp of the start time (number of seconds elapsed since January 1, 1970)
allDay	boolean	-	No	Whether it is an all-day event, which defaults to "false".
description	string	-	No	Event Description
location	string	-	No	Event Location
endTime	string	-	No	Unix timestamp of the end time
alarm	boolean	-	No	Whether to send a reminder, which defaults to "true".
alarmOffset	number	-	No	Reminder lead time, in seconds. The default value is 0, indicating a reminder at the start of the event.
repeatInterval	string	-	No	Repetition cycle, defaulting to 'month' for monthly repetition
repeatEndTime	number	-	No	Unix timestamp for the end of the repetition cycle. If left blank, it implies indefinite repetition.
success	function	-	No	Callback Function of Successful Interface Call
fail	function	-	No	Callback Function of Failing Interface Call
complete	function	-	No	Callback function executed upon the completion of the interface invocation (both successful and unsuccessful invocations)

repeatInterval

Valid Values	Description
day	Recurring day by day

©2013-2022 Tencent Cloud. All rights reserved.

week	Recurring week by week
month	Recurring month by month. Under this mode, the date should not exceed the 28th day.
year	Recurring year by year

©2013-2022 Tencent Cloud. All rights reserved.

Clipboard

Last updated：2024-07-16 17:58:59

setClipboardData

This API is used via wx.setClipboardData(Object object).

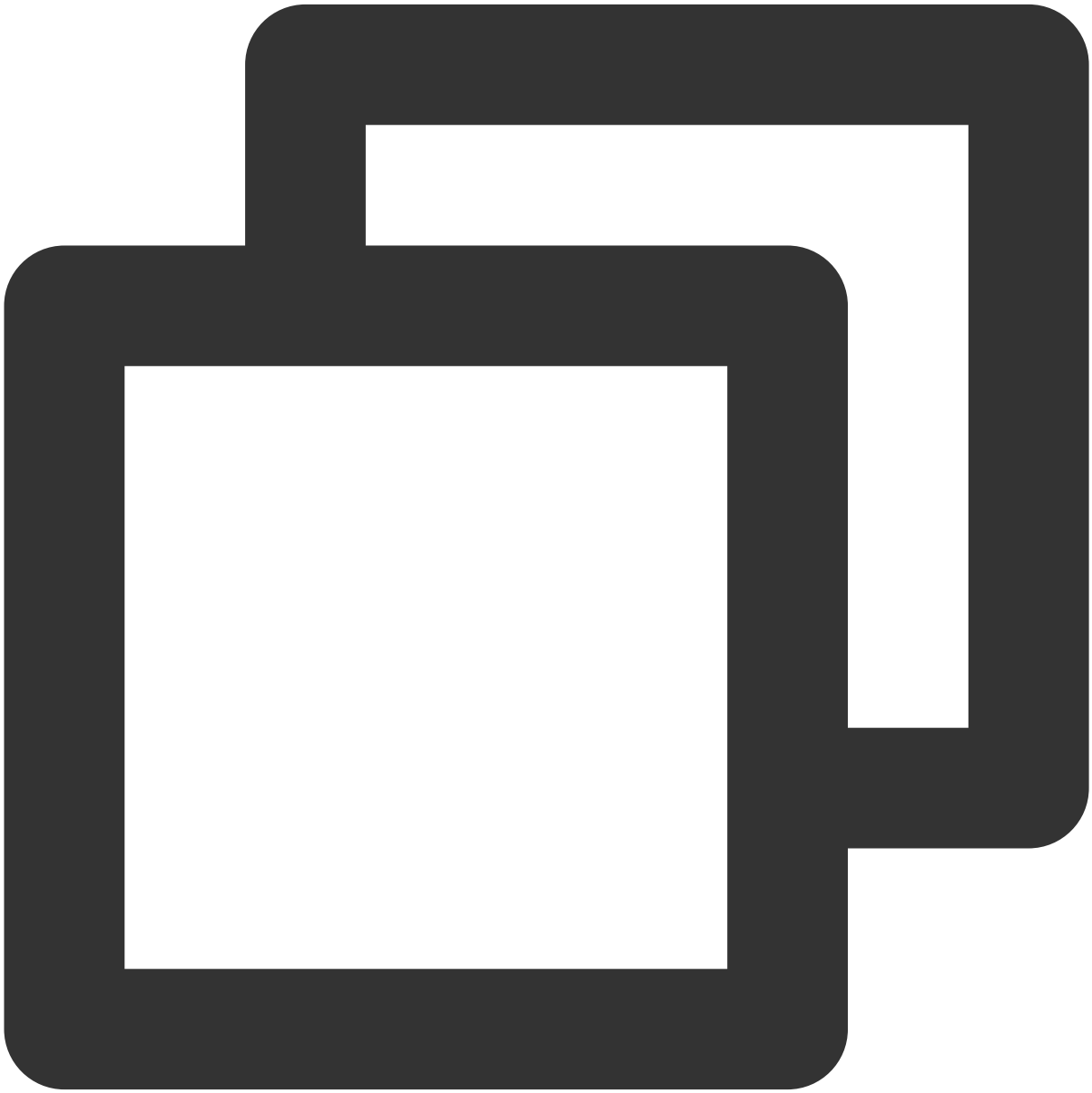
Feature Description: Sets the content of the system clipboard. For individual users, it should not be invoked more than once per second.

Parameter and Description: Object.

Attribute	Type	Default value	Required	Description
data	string	-	Yes	Clipboard content
success	function	-	No	Callback Function of Successful Interface Call
fail	function	-	No	Callback Function of Failing Interface Call
complete	function	-	No	Callback function executed upon the completion of the interface invocation (both successful and unsuccessful invocations)

Sample Code

©2013-2022 Tencent Cloud. All rights reserved.



```
wx.setClipboardData({
  data: 'data',
  success(res) {
    wx.getClipboardData({
      success(res) {
        console.log(res.data) // data
      }
    })
  }
})
```

©2013-2022 Tencent Cloud. All rights reserved.

getClipboardData

This API is used via wx.getClipboardData(Object object).

Feature Description: Gets the content of the system clipboard. For individual users, the invocation does not exceed `1` time per second.

Parameter and Description: Object.

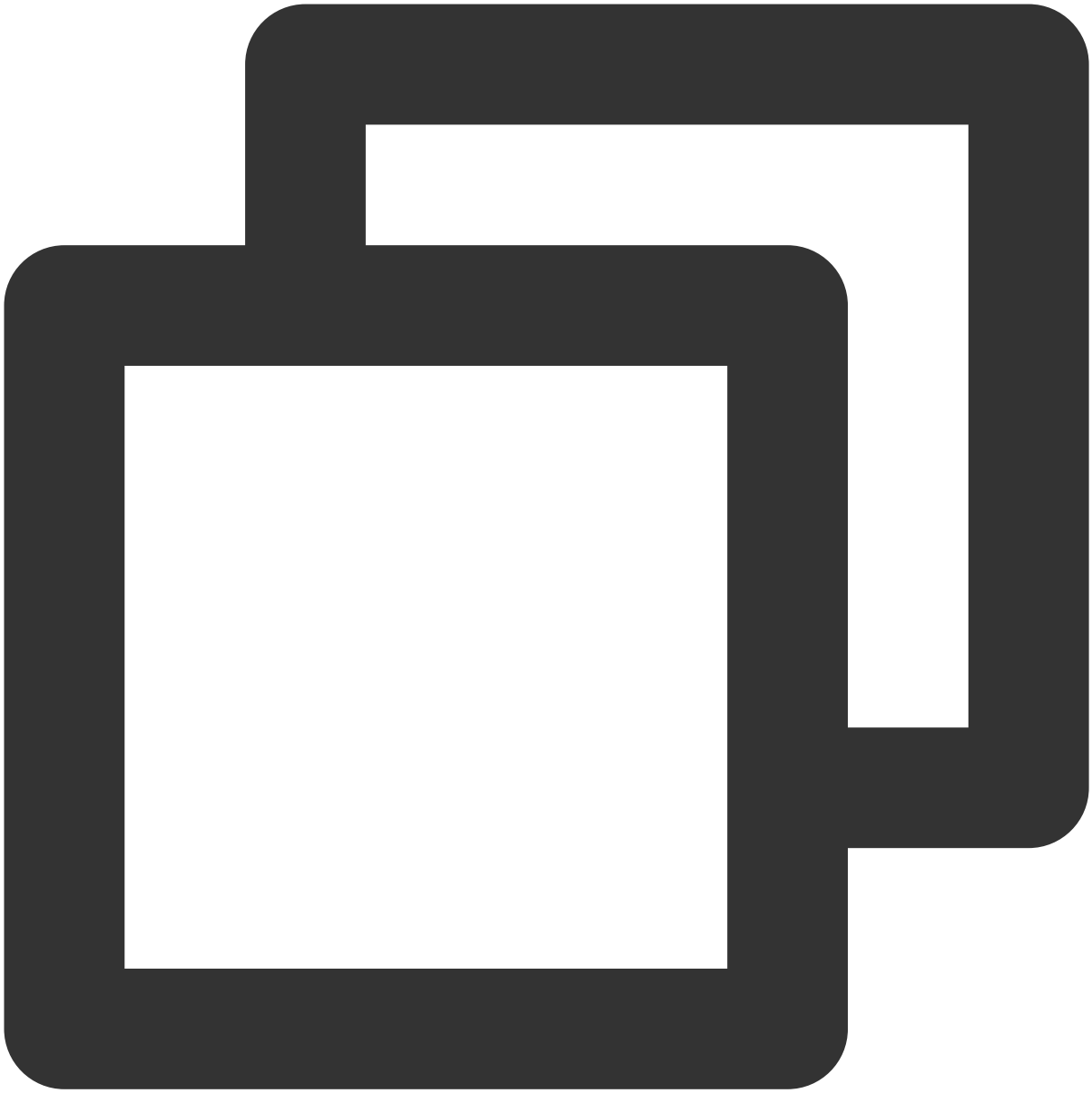
Attribute	Type	Default value	Required	Description
success	function	-	No	Callback Function of Successful Interface Call
fail	function	-	No	Callback Function of Failing Interface Call
complete	function	-	No	Callback function executed upon the completion of the interface invocation (both successful and unsuccessful invocations)

object.success callback function: Object.

Attribute	Type	Description
data	string	Clipboard content

Sample Code

©2013-2022 Tencent Cloud. All rights reserved.



```
wx.getClipboardData({
  success(res) {
    console.log(res.data)
  }
})
```

©2013-2022 Tencent Cloud. All rights reserved.

NFC—getNFCAdapter

Last updated：2024-07-16 17:59:25

The application of this API is through the NFCAdapter wx.getNFCAdapter().

Note:

Compatibility:

Android Version: Supported.

iOS Version: Not Supported.

Function Description: Acquire NFC Instance.

Parameters and Description: Returns NFCAdapter NFC Instance.

Error Code

Error Codes	Error Message	Note
13000	The device does not support NFC	-
13001	The system's NFC switch is not activated.	-
13010	An unknown error occurred.	-
13019	user is not authorized	User has not granted permission.
13011	invalid parameter	Invalid parameter
13012	parse NdefMessage failed	Failed to parse the parameter into NdefMessage.
13021	NFC discovery already started	NFC scanning has already commenced.
13018	NFC discovery has not started	Attempted to halt NFC scanning without initiating it.
13022	Tech already connected	The tag has been successfully connected.
13023	Tech has not connected	Attempted to disconnect without a tag being connected.
13013	NFC tag has not been discovered	No NFC tag was detected.
13014	invalid tech	Invalid tag technology.
13015	unavailable tech	Failed to acquire the corresponding

©2013-2022 Tencent Cloud. All rights reserved.

		technology from the tag.
13024	function not support	The current tag technology does not support this feature.
13017	system internal error	The related read-write operation has failed.
13016	connect fail	Connection failed

©2013-2022 Tencent Cloud. All rights reserved.

NFC—NFCAdapter

Last updated：2024-07-16 17:59:45

NFCAdapter

Parameters and Description: Object tech, enumeration of tag types.

Attributes	Types	Note
ndef	String	Corresponds to the Ndef instance, the instance supports reading and writing of NDEF data on NFC tags in NDEF format.
nfcA	String	Corresponds to the NfcA instance, the instance supports reading and writing in accordance with the NFC-A (ISO 14443-3A) standard.
nfcB	String	Corresponds to the NfcB instance, the instance supports reading and writing in accordance with the NFC-B (ISO 14443-3B) standard.
isoDep	String	Corresponds to the IsoDep instance, the instance supports reading and writing in accordance with the ISO-DEP (ISO 14443-4) standard.
nfcF	String	Corresponds to the NfcF instance, the instance supports reading and writing in accordance with the NFC-F (JIS 6319-4) standard.
nfcV	String	Corresponds to the NfcV instance, the instance supports reading and writing in accordance with the NFC-V (ISO 15693) standard.
mifareClassic	String	Corresponds to the MifareClassic instance, the instance supports reading and writing of MIFARE Classic tags.
mifareUltralight	String	Corresponds to the MifareUltralight instance, the instance supports reading and writing of MIFARE Ultralight tags.

.startDiscovery

The method is utilized as NFCAdapter.startDiscovery().

Parameters and Description: Object object.

Attributes	Types	Default value	Required	Note
success	Function	-	Not	Callback function for successful interface

©2013-2022 Tencent Cloud. All rights reserved.

			required	invocation
fail	Function	-	Not required	Callback function for successful interface invocation
complete	Function	-	Not required	Callback function executed upon the completion of the interface invocation (both successful and unsuccessful invocations)

Error Code

Error Codes	Error Message	Note
13000	The device does not support NFC	-
13001	The system's NFC switch is not activated.	-
13010	An unknown error occurred.	-
13019	user is not authorized	User Authorization Denied
13011	invalid parameter	Invalid parameter
13012	parse NdefMessage failed	Failed to parse the parameter into NdefMessage.
13021	NFC discovery already started	NFC scanning has already commenced.
13018	NFC discovery has not started	Attempted to halt NFC scanning without initiating it.
13022	Tech already connected	The tag has been successfully connected.
13023	Tech has not connected	Attempted to disconnect without a tag being connected.
13013	NFC tag has not been discovered	No NFC tag was detected.
13014	invalid tech	Invalid tag technology.
13015	unavailable tech	Failed to acquire the corresponding technology from the tag.
13024	function not support	The current tag technology does not support this feature.

©2013-2022 Tencent Cloud. All rights reserved.

13017	system internal error	The related read-write operation has failed.
13016	connect fail	Connection failed

.stopDiscovery

The method is utilized as NFCAdapter.stopDiscovery().

Parameters and Description: Object object.

Attributes	Types	Default value	Required	Note
success	function	-	Not required	Callback function for successful interface invocation
fail	function	-	Not required	Callback function for unsuccessful interface invocation
complete	function	-	Not required	Callback function executed upon the completion of the interface invocation (both successful and unsuccessful invocations)

Error Code

Error Codes	Error Message	Note
13000	The device does not support NFC	-
13001	The system's NFC switch is not activated.	-
13010	An unknown error occurred.	-
13019	user is not authorized	User Authorization Denied
13011	invalid parameter	Invalid parameter
13012	parse NdefMessage failed	Failed to parse the parameter into NdefMessage.
13021	NFC discovery already started	NFC scanning has already commenced.
13018	NFC discovery has not started	Attempted to halt NFC scanning without initiating it.
13022	Tech already connected	The tag has been successfully connected.

©2013-2022 Tencent Cloud. All rights reserved.

13023	Tech has not connected	Attempted to disconnect without a tag being connected.
13013	NFC tag has not been discovered	No NFC tag was detected.
13014	invalid tech	Invalid tag technology.
13015	unavailable tech	Failed to acquire the corresponding technology from the tag.
13024	function not support	The current tag technology does not support this feature.
13017	system internal error	The related read-write operation has failed.
13016	connect fail	Connection failed

.getNdef

The method is employed as Ndef NFCAdapter.getNdef().

Function Description: Acquire the Ndef instance, which supports reading and writing of NDEF data on NFC tags in NDEF format.
Return Value: Ndef.

.getNfcA

The method is implemented as NfcA NFCAdapter.getNfcA().

Function Description: Obtain the NfcA instance, which supports reading and writing in accordance with the NFC-A (ISO 14443-3A) standard.
Return Value: NfcA.

.getNfcB

The method is utilized as NfcB NFCAdapter.getNfcB().

©2013-2022 Tencent Cloud. All rights reserved.

Function Description: Acquire the NfcB instance, which facilitates reading and writing in compliance with the NFC-B (ISO 14443-3B) standard.
Return Value: NfcB.

.getNfcF

The method is employed as NfcF NFCAdapter.getNfcF().

Function Description: Procure the NfcF instance, which supports reading and writing in accordance with the NFC-F (JIS 6319-4) standard.
Return Value: NfcF.

.getNfcV

The method is utilized as NfcV NFCAdapter.getNfcV().

Function Description: Acquire the NfcV instance, which supports reading and writing in compliance with the NFC-V (ISO 15693) standard.
Return Value: NfcV.

.getIsoDep

The method is employed as IsoDep NFCAdapter.getIsoDep().

Function Description: Obtain the IsoDep instance, which supports reading and writing in accordance with the ISO-DEP (ISO 14443-4) standard.
Return Value: IsoDep.

.getMifareClassic

The method is utilized as MifareClassic NFCAdapter.getMifareClassic().

Function Description: Acquire the MifareClassic instance, which supports reading and writing of MIFARE Classic tags.

©2013-2022 Tencent Cloud. All rights reserved.

Return Value: MifareUltralight.

.getMifareUltralight

The method is employed as MifareUltralight NFCAdapter.getMifareUltralight().

Function Description: Obtain the MifareUltralight instance, which supports the reading and writing of MIFARE Ultralight tags.
Parameter Description: MifareUltralight.

.onDiscovered

The method is utilized as NFCAdapter.onDiscovered(function listener).

Function Description: Monitor NFC Tag.
Parameter Description:function listener, listener function, with the following arguments Object res:

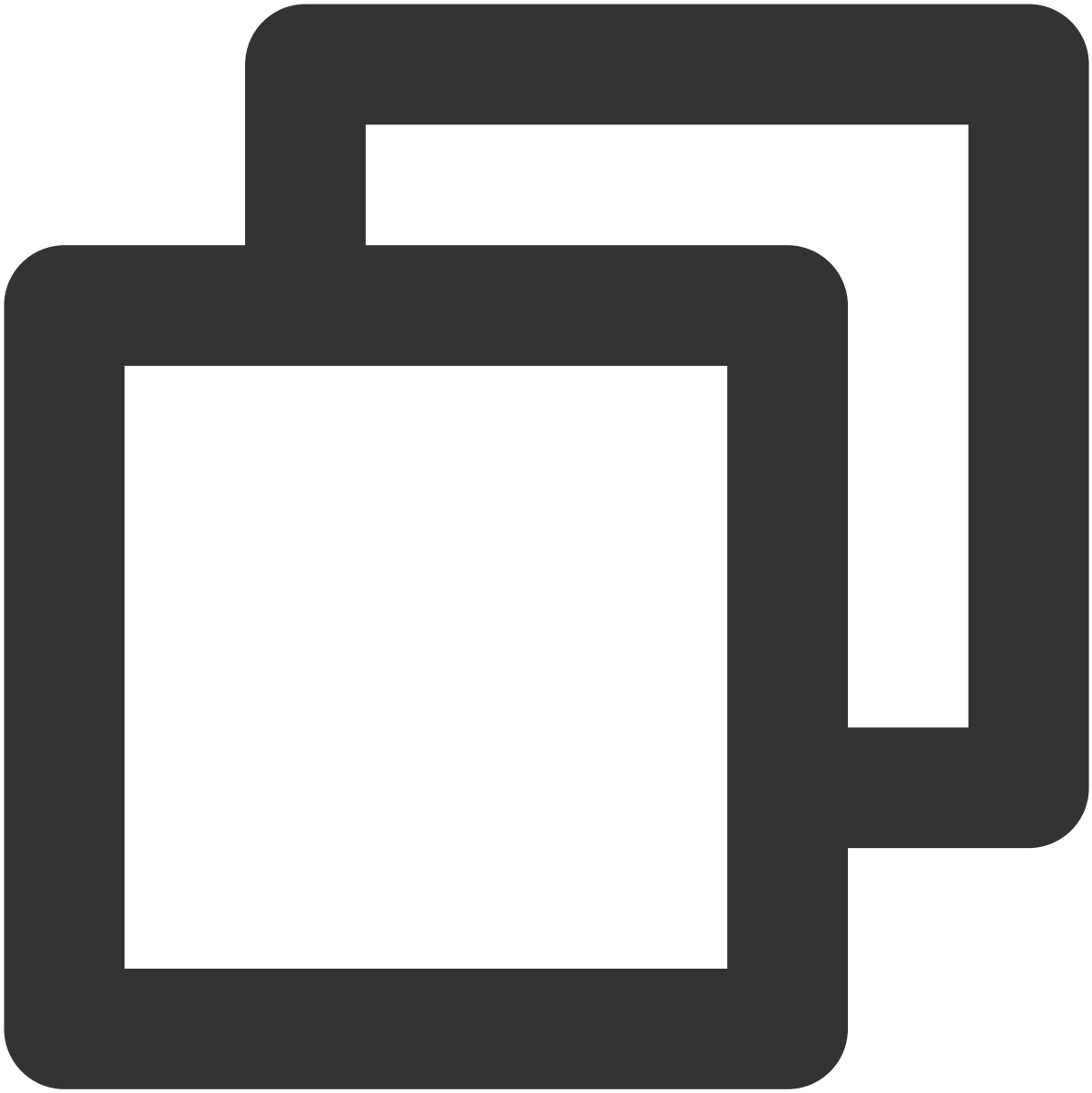
Attributes	Types	Note
id	ArrayBuffer	-
techs	Array	Tech array, used to match which specific standards (such as NfcA instances) can be used to handle NFC cards.
messages	Array	NdefMessage array, the message format is {id: ArrayBuffer, type: ArrayBuffer, payload: ArrayBuffer}.

.offDiscovered

The method is utilized as NFCAdapter.offDiscovered(function listener).

Function Description: Remove the listener function of the NFC Tag.
Parameters and Description: function listener, the listening function passed in by onDiscovered. If this parameter is not passed, all listening functions will be removed.
Example Code

©2013-2022 Tencent Cloud. All rights reserved.



```
const listener = function (res) { console.log(res) }

NFCAdapter.onDiscovered(listener)
NFCAdapter.offDiscovered(listener) // The same function object as the listener must
```

©2013-2022 Tencent Cloud. All rights reserved.

NFC—IsoDep

Last updated：2024-07-16 18:00:18

.connect

The method is utilized as IsoDep.connect().

Function Description: Establishes a connection with the NFC tag.

Parameters and Description: Object object.

Attributes	Types	Default value	Required	Note
success	function	-	Not required	Callback function for successful interface invocation
fail	function	-	Not required	Callback for interface invocation failure.
complete	function	-	Not required	Callback function executed upon the completion of the interface invocation (both successful and unsuccessful invocations)

Error Code

Error Codes	Error Message	Note
13000	The device does not support NFC.	-
13001	The system's NFC switch is not activated.	-
13010	An unknown error occurred.	-
13019	user is not authorized	User Authorization Denied
13011	invalid parameter	Invalid parameter
13012	parse NdefMessage failed	Failed to parse the parameter into NdefMessage.
13021	NFC discovery already started	NFC scanning has already commenced.

©2013-2022 Tencent Cloud. All rights reserved.

13018	NFC discovery has not started	Attempted to halt NFC scanning without initiating it.
13022	Tech already connected	The tag has been successfully connected.
13023	Tech has not connected	Attempted to disconnect without a tag being connected.
13013	NFC tag has not been discovered	No NFC tag was detected.
13014	invalid tech	Invalid tag technology.
13015	unavailable tech	Failed to acquire the corresponding technology from the tag.
13024	function not support	The current tag technology does not support this feature.
13017	system internal error	The related read-write operation has failed.
13016	connect fail	Connection failed

.close

The method is employed as IsoDep.close().

Function Description: Disconnects the link.

Parameters and Description: Object object.

Attributes	Types	Default value	Required	Note
success	function	-	Not required	Callback function for successful interface invocation
fail	function	-	Not required	Callback function for unsuccessful interface invocation
complete	function	-	Not required	Callback function executed upon the completion of the interface invocation (both successful and unsuccessful invocations)

Error Code

©2013-2022 Tencent Cloud. All rights reserved.

Error Codes	Error Message	Note
13000	The device does not support NFC.	-
13001	The system's NFC switch is not activated.	-
13010	An unknown error occurred.	
13019	user is not authorized	User Authorization Denied
13011	invalid parameter	Invalid parameter
13012	parse NdefMessage failed	Failed to parse the parameter into NdefMessage.
13021	NFC discovery already started	NFC scanning has already commenced.
13018	NFC discovery has not started	Attempted to halt NFC scanning without initiating it.
13022	Tech already connected	The tag has been successfully connected.
13023	Tech has not connected	Attempted to disconnect without a tag being connected.
13013	NFC tag has not been discovered	No NFC tag was detected.
13014	invalid tech	Invalid tag technology.
13015	unavailable tech	Failed to acquire the corresponding technology from the tag.
13024	function not support	The current tag technology does not support this feature.
13017	system internal error	The related read-write operation has failed.
13016	connect fail	Connection failed

.setTimeout

> The method is implemented as IsoDep.setTimeout(Object object).

Function Description: Setting the timeout duration.

©2013-2022 Tencent Cloud. All rights reserved.

Parameters and Description: Object object.

Attributes	Types	Default value	Required	Note
timeout	number	-	Supported	Setting the timeout duration (ms).
success	function	-	Not required	Callback function for successful interface invocation
fail	function	-	Not required	Callback function for unsuccessful interface invocation
complete	function		Not required	Callback function executed upon the completion of the interface invocation (both successful and unsuccessful invocations)

Error Code

Error Codes	Error Message	Note
13000	The device does not support NFC.	-
13001	The system's NFC switch is not activated.	-
13010	An unknown error occurred.	-
13019	user is not authorized	User Authorization Denied
13011	invalid parameter	Invalid parameter
13012	parse NdefMessage failed	Failed to parse the parameter into NdefMessage.
13021	NFC discovery already started	NFC scanning has already commenced.
13018	NFC discovery has not started	Attempted to halt NFC scanning without initiating it.
13022	Tech already connected	The tag has been successfully connected.
13023	Tech has not connected	Attempted to disconnect without a tag being connected.
13013	NFC tag has not been discovered	No NFC tag was detected.
13014	invalid tech	Invalid tag technology.
13015	unavailable tech	Failed to acquire the corresponding technology from the tag.

©2013-2022 Tencent Cloud. All rights reserved.

13024	function not support	The current tag technology does not support this feature.
13017	system internal error	The related read-write operation has failed.
13016	connect fail	Connection failed

.isConnected

The method is utilized as IsoDep.isConnected().

Function Description: Verifies if a connection has been established.

Parameters and Description: Object object

Attributes	Types	Default value	Required	Note
success	function	-	Not required	Callback function for successful interface invocation
fail	function	-	Not required	Callback function for unsuccessful interface invocation
complete	function	-	Not required	Callback function executed upon the completion of the interface invocation (both successful and unsuccessful invocations)

Error Code

Error Codes	Error Message	Note
13000	The device does not support NFC.	-
13001	The system's NFC switch is not activated.	-
13010	An unknown error occurred.	-
13019	user is not authorized	User Authorization Denied
13011	invalid parameter	Invalid parameter
13012	parse NdefMessage failed	Failed to parse the parameter into NdefMessage.

©2013-2022 Tencent Cloud. All rights reserved.

13021	NFC discovery already started	NFC scanning has already commenced.
13018	NFC discovery has not started	Attempted to halt NFC scanning without initiating it.
13022	Tech already connected	The tag has been successfully connected.
13023	Tech has not connected	Attempted to disconnect without a tag being connected.
13013	NFC tag has not been discovered	No NFC tag was detected.
13014	invalid tech	Invalid tag technology.
13015	unavailable tech	Failed to acquire the corresponding technology from the tag.
13024	function not support	The current tag technology does not support this feature.
13017	system internal error	The related read-write operation has failed.
13016	connect fail	Connection failed

.getMaxTransceiveLength

The method is executed as IsoDep.getMaxTransceiveLength().

Function Description: Retrieves the maximum transmission length.

Parameters and Description: Object object

Attributes	Types	Default value	Required	Note
success	function	-	Not required	Callback function for successful interface invocation
fail	function	-	Not required	Callback function for unsuccessful interface invocation
complete	function	-	Not required	Callback function executed upon the completion of the interface invocation (both successful and unsuccessful invocations)

object.success Callback Function Parameters: Object res

Attributes	Types	Note

©2013-2022 Tencent Cloud. All rights reserved.

length	number	Maximum Transmission Length

Error Code

Error Codes	Error Message	Note
13000	The device does not support NFC.	-
13001	The system's NFC switch is not activated.	-
13010	An unknown error occurred.	-
13019	user is not authorized	User Authorization Denied
13011	invalid parameter	Invalid parameter
13012	parse NdefMessage failed	Failed to parse the parameter into NdefMessage.
13021	NFC discovery already started	NFC scanning has already commenced.
13018	NFC discovery has not started	Attempted to halt NFC scanning without initiating it.
13022	Tech already connected	The tag has been successfully connected.
13023	Tech has not connected	Attempted to disconnect without a tag being connected.
13013	NFC tag has not been discovered	No NFC tag was detected.
13014	invalid tech	Invalid tag technology.
13015	unavailable tech	Failed to acquire the corresponding technology from the tag.
13024	function not support	The current tag technology does not support this feature.
13017	system internal error	The related read-write operation has failed.
13016	connect fail	Connection failed

.transceive

©2013-2022 Tencent Cloud. All rights reserved.

The method is executed as IIsoDep.transceive(Object object).

Function Description: Transmits data.

Parameters and Description: Object object

Attributes	Types	Default value	Required	Note
data	ArrayBuffer	-	Supported	Binary data required for transmission
success	function	-	Not required	Callback function for successful interface invocation
fail	function	-	Not required	Callback function for unsuccessful interface invocation
complete	function	-	Not required	Callback function executed upon the completion of the interface invocation (both successful and unsuccessful invocations)

object.success Callback Function Parameters: Object res

Attributes	Types	Note
data	ArrayBuffer	-

Error Code

Error Codes	Error Message	Note
13000	The device does not support NFC.	-
13001	The system's NFC switch is not activated.	-
13010	An unknown error occurred.	-
13019	user is not authorized	User Authorization Denied
13011	invalid parameter	Invalid parameter
13012	parse NdefMessage failed	Failed to parse the parameter into NdefMessage.
13021	NFC discovery already started	NFC scanning has already commenced.

©2013-2022 Tencent Cloud. All rights reserved.

13018	NFC discovery has not started	Attempted to halt NFC scanning without initiating it.
13022	Tech already connected	The tag has been successfully connected.
13023	Tech has not connected	Attempted to disconnect without a tag being connected.
13013	NFC tag has not been discovered	No NFC tag was detected.
13014	invalid tech	Invalid tag technology.
13015	unavailable tech	Failed to acquire the corresponding technology from the tag.
13024	function not support	The current tag technology does not support this feature.
13017	system internal error	The related read-write operation has failed.
13016	connect fail	Connection failed

.getHistoricalBytes

The method is invoked as IsoDep.getHistoricalBytes().

Function Description: Acquiring reset information.

Parameters and Description: Object object

Attributes	Types	Default value	Required	Note
success	function	-	Not required	Callback function for successful interface invocation
fail	function	-	Not required	Callback function for unsuccessful interface invocation
complete	function	-	Not required	Callback function executed upon the completion of the interface invocation (both successful and unsuccessful invocations)

object.success Callback Function Parameters: Object res

©2013-2022 Tencent Cloud. All rights reserved.

Attributes	Types	Note
histBytes	ArrayBuffer	Returns historical binary data.

Error Code

Error Codes	Error Message	Note
13000	The device does not support NFC.	-
13001	The system's NFC switch is not activated.	-
13010	An unknown error occurred.	-
13019	user is not authorized	User Authorization Denied
13011	invalid parameter	Invalid parameter
13012	parse NdefMessage failed	Failed to parse the parameter into NdefMessage.
13021	NFC discovery already started	NFC scanning has already commenced.
13018	NFC discovery has not started	Attempted to halt NFC scanning without initiating it.
13022	Tech already connected	The tag has been successfully connected.
13023	Tech has not connected	Attempted to disconnect without a tag being connected.
13013	NFC tag has not been discovered	No NFC tag was detected.
13014	invalid tech	Invalid tag technology.
13015	unavailable tech	Failed to acquire the corresponding technology from the tag.
13024	function not support	The current tag technology does not support this feature.
13017	system internal error	The related read-write operation has failed.
13016	connect fail	Connection failed

©2013-2022 Tencent Cloud. All rights reserved.

NFC—MifareClassic

Last updated：2024-07-16 18:00:50

.connect

The method is utilized as MifareClassic.connect().

Function Description: Establishes a connection with the NFC tag.

Parameters and Description: Object object.

Attributes	Types	Default value	Required	Note
success	function	-	Not required	Callback function for successful interface invocation
fail	function	-	Not required	Callback function for unsuccessful interface invocation
complete	function	-	Not required	Callback function executed upon the completion of the interface invocation (both successful and unsuccessful invocations)

Error Code

Error Codes	Error Message	Note
13000	The device does not support NFC	-
13001	The system's NFC switch is not activated.	-
13010	An unknown error occurred.	-
13019	user is not authorized	User Authorization Denied
13011	invalid parameter	Invalid parameter
13012	parse NdefMessage failed	Failed to parse the parameter into NdefMessage.

©2013-2022 Tencent Cloud. All rights reserved.

13021	NFC discovery already started	NFC scanning has already commenced.
13018	NFC discovery has not started	Attempted to halt NFC scanning without initiating it.
13022	Tech already connected	The tag has been successfully connected.
13023	Tech has not connected	Attempted to disconnect without a tag being connected.
13013	NFC tag has not been discovered	No NFC tag was detected.
13014	invalid tech	Invalid tag technology.
13015	unavailable tech	Failed to acquire the corresponding technology from the tag.
13024	function not support	The current tag technology does not support this feature.
13017	system internal error	The related read-write operation has failed.
13016	connect fail	Connection failed

.close

The method is employed as MifareClassic.close().

Function Description: Disconnects the link.

Parameters and Description: Object object.

Attributes	Types	Default value	Required	Note
success	function	-	Not required	Callback function for successful interface invocation
fail	function	-	Not required	Callback function for unsuccessful interface invocation
complete	function	-	Not required	Callback function executed upon the completion of the interface invocation (both successful and unsuccessful invocations)

Error Code

Error	Error Message	Note

©2013-2022 Tencent Cloud. All rights reserved.

Codes		
13000	The device does not support NFC	-
13001	The system's NFC switch is not activated.	-
13010	An unknown error occurred.	-
13019	user is not authorized	User Authorization Denied
13011	invalid parameter	Invalid parameter
13012	parse NdefMessage failed	Failed to parse the parameter into NdefMessage.
13021	NFC discovery already started	NFC scanning has already commenced.
13018	NFC discovery has not started	Attempted to halt NFC scanning without initiating it.
13022	Tech already connected	The tag has been successfully connected.
13023	Tech has not connected	Attempted to disconnect without a tag being connected.
13013	NFC tag has not been discovered	No NFC tag was detected.
13014	invalid tech	Invalid tag technology.
13015	unavailable tech	Failed to acquire the corresponding technology from the tag.
13024	function not support	The current tag technology does not support this feature.
13017	system internal error	The related read-write operation has failed.
13016	connect fail	Connection failed

.setTimeout

The method is implemented as MifareClassic.setTimeout(Object object).

Function Description: Setting the timeout duration.

Parameters and Description: Object object.

©2013-2022 Tencent Cloud. All rights reserved.

Attributes	Types	Default value	Required	Note
timeout	number	-	Supported	Setting the timeout duration (ms).
success	function	-	Not required	Callback function for successful interface invocation
fail	function	-	Not required	Callback function for unsuccessful interface invocation
complete	function	-	Not required	Callback function executed upon the completion of the interface invocation (both successful and unsuccessful invocations)

Error Code

Error Codes	Error Message	Note
13000	The device does not support NFC	-
13001	The system's NFC switch is not activated.	-
13010	An unknown error occurred.	-
13019	user is not authorized	User Authorization Denied
13011	invalid parameter	Invalid parameter
13012	parse NdefMessage failed	Failed to parse the parameter into NdefMessage.
13021	NFC discovery already started	NFC scanning has already commenced.
13018	NFC discovery has not started	Attempted to halt NFC scanning without initiating it.
13022	Tech already connected	The tag has been successfully connected.
13023	Tech has not connected	Attempted to disconnect without a tag being connected.
13013	NFC tag has not been discovered	No NFC tag was detected.
13014	invalid tech	Invalid tag technology.
13015	unavailable tech	Failed to acquire the corresponding technology from the tag.

©2013-2022 Tencent Cloud. All rights reserved.

13024	function not support	The current tag technology does not support this feature.
13017	system internal error	The related read-write operation has failed.
13016	connect fail	Connection failed

.isConnected

The method is invoked as MifareClassic.isConnected().

Note:

This interface has been deprecated; developers can maintain the connection status independently.

Function Description: Verifies if a connection has been established.

Parameters and Description: Object object.

Attributes	Types	Default value	Required	Note
success	function	-	Not required	Callback function for successful interface invocation
fail	function	-	Not required	Callback function for unsuccessful interface invocation
complete	function	-	Not required	Callback function executed upon the completion of the interface invocation (both successful and unsuccessful invocations)

Error Code

Error Codes	Error Message	Note
13000	The device does not support NFC	-
13001	The system's NFC switch is not activated.	-
13010	An unknown error occurred.	-
13019	user is not authorized	User Authorization Denied

©2013-2022 Tencent Cloud. All rights reserved.

13011	invalid parameter	Invalid parameter
13012	parse NdefMessage failed	Failed to parse the parameter into NdefMessage.
13021	NFC discovery already started	NFC scanning has already commenced.
13018	NFC discovery has not started	Attempted to halt NFC scanning without initiating it.
13022	Tech already connected	The tag has been successfully connected.
13023	Tech has not connected	Attempted to disconnect without a tag being connected.
13013	NFC tag has not been discovered	No NFC tag was detected.
13014	invalid tech	Invalid tag technology.
13015	unavailable tech	Failed to acquire the corresponding technology from the tag.
13024	function not support	The current tag technology does not support this feature.
13017	system internal error	The related read-write operation has failed.
13016	connect fail	Connection failed

.getMaxTransceiveLength

The method is executed as MifareClassic.getMaxTransceiveLength().

Function Description: Acquiring the Maximum Transmission Length.

Parameters and Description: Object object

Attributes	Types	Default value	Required	Note
success	function	-	Not required	Callback function for successful interface invocation
fail	function	-	Not required	Callback function for unsuccessful interface invocation
complete	function	-	Not required	Callback function executed upon the completion of the interface invocation (both successful and unsuccessful invocations)

©2013-2022 Tencent Cloud. All rights reserved.

object.success Callback Function Parameters: Object res

Attributes	Types	Note
length	number	Maximum Transmission Length

Error Code

Error Codes	Error Message	Note
13000	The device does not support NFC	-
13001	The system's NFC switch is not activated.	-
13010	An unknown error occurred.	-
13019	user is not authorized	User Authorization Denied
13011	invalid parameter	Invalid parameter
13012	parse NdefMessage failed	Failed to parse the parameter into NdefMessage.
13021	NFC discovery already started	NFC scanning has already commenced.
13018	NFC discovery has not started	Attempted to halt NFC scanning without initiating it.
13022	Tech already connected	The tag has been successfully connected.
13023	Tech has not connected	Attempted to disconnect without a tag being connected.
13013	NFC tag has not been discovered	No NFC tag was detected.
13014	invalid tech	Invalid tag technology.
13015	unavailable tech	Failed to acquire the corresponding technology from the tag.
13024	function not support	The current tag technology does not support this feature.
13017	system internal error	The related read-write operation has failed.
13016	connect fail	Connection failed

.transceive

©2013-2022 Tencent Cloud. All rights reserved.

The method is utilized as MifareClassic.transceive(Object object).

Function Description: Transmitting data.

Parameters and Description: Object object.

Attributes	Types	Default value	Required	Note
data	ArrayBuffer	-	Supported	Binary data required for transmission
success	function	-	Not required	Callback function for successful interface invocation
fail	function	-	Not required	Callback function for unsuccessful interface invocation
complete	function	-	Not required	Callback function executed upon the completion of the interface invocation (both successful and unsuccessful invocations)

object.success Callback Function Parameters: Object res

Attributes	Types	Note
data	ArrayBuffer	-

Error Code

Error Codes	Error Message	Note
13000	The device does not support NFC	-
13001	The system's NFC switch is not activated.	-
13010	An unknown error occurred.	-
13019	user is not authorized	User Authorization Denied
13011	invalid parameter	Invalid parameter
13012	parse NdefMessage failed	Failed to parse the parameter into NdefMessage.
13021	NFC discovery already started	NFC scanning has already commenced.

©2013-2022 Tencent Cloud. All rights reserved.

13018	NFC discovery has not started	Attempted to halt NFC scanning without initiating it.
13022	Tech already connected	The tag has been successfully connected.
13023	Tech has not connected	Attempted to disconnect without a tag being connected.
13013	NFC tag has not been discovered	No NFC tag was detected.
13014	invalid tech	Invalid tag technology.
13015	unavailable tech	Failed to acquire the corresponding technology from the tag.
13024	function not support	The current tag technology does not support this feature.
13017	system internal error	The related read-write operation has failed.
13016	connect fail	Connection failed

©2013-2022 Tencent Cloud. All rights reserved.

NFC—MifareUltralight

Last updated：2024-07-16 18:01:22

.connect

The method is utilized as MifareUltralight.connect().

Function Description: Establishes a connection with the NFC tag.

Parameters and Description: Object object.

Attributes	Types	Default value	Required	Note
success	function	-	Not required	Callback function for successful interface invocation
fail	function	-	Not required	Callback function for failed interface invocation
complete	function	-	Not required	Callback function executed upon the completion of the interface invocation (both successful and unsuccessful invocations)

Error Code

Error Codes	Error Message	Note
13000	The device does not support NFC	-
13001	The system's NFC switch is not activated.	-
13010	An unknown error occurred.	-
13019	user is not authorized	User Authorization Denied
13011	invalid parameter	Invalid parameter
13012	parse NdefMessage failed	Failed to parse the parameter into NdefMessage.
13021	NFC discovery already started	NFC scanning has already commenced.
13018	NFC discovery has not started	Attempted to halt NFC scanning without initiating it.

©2013-2022 Tencent Cloud. All rights reserved.

13022	Tech already connected	The tag has been successfully connected.
13023	Tech has not connected	Attempted to disconnect without a tag being connected.
13013	NFC tag has not been discovered	No NFC tag was detected.
13014	invalid tech	Invalid tag technology.
13015	unavailable tech	Failed to acquire the corresponding technology from the tag.
13024	function not support	The current tag technology does not support this feature.
13017	system internal error	The related read-write operation has failed.
13016	connect fail	Connection failed

.close

The method is utilized as MifareUltralight.close().

Function Description: Disconnects the link.

Parameters and Description: Object object.

Attributes	Types	Default value	Required	Note
success	function	-	Not required	Callback function for successful interface invocation
fail	function	-	Not required	Callback function for unsuccessful interface invocation
complete	function	-	Not required	Callback function executed upon the completion of the interface invocation (both successful and unsuccessful invocations)

Error Code

Error Codes	Error Message	Note
13000	The device does not support NFC	-

©2013-2022 Tencent Cloud. All rights reserved.

13001	The system's NFC switch is not activated.	-
13010	An unknown error occurred.	-
13019	user is not authorized	User Authorization Denied
13011	invalid parameter	Invalid parameter
13012	parse NdefMessage failed	Failed to parse the parameter into NdefMessage.
13021	NFC discovery already started	NFC scanning has already commenced.
13018	NFC discovery has not started	Attempted to halt NFC scanning without initiating it.
13022	Tech already connected	The tag has been successfully connected.
13023	Tech has not connected	Attempted to disconnect without a tag being connected.
13013	NFC tag has not been discovered	No NFC tag was detected.
13014	invalid tech	Invalid tag technology.
13015	unavailable tech	Failed to acquire the corresponding technology from the tag.
13024	function not support	The current tag technology does not support this feature.
13017	system internal error	The related read-write operation has failed.
13016	connect fail	Connection failed

.setTimeout

The method is employed as MifareUltralight.setTimeout(Object object).

Function Description: Setting the timeout duration.

Parameters and Description: Object object.

Attributes	Types	Default value	Required	Note

©2013-2022 Tencent Cloud. All rights reserved.

timeout	number	-	Supported	Setting the timeout duration (ms)
success	function	-	Not required	Callback function for successful interface invocation
fail	function	-	Not required	Callback function for unsuccessful interface invocation
complete	function	-	Not required	Callback function executed upon the completion of the interface invocation (both successful and unsuccessful invocations)

Error Code

Error Codes	Error Message	Note
13000	The device does not support NFC	-
13001	The system's NFC switch is not activated.	-
13010	An unknown error occurred.	-
13019	user is not authorized	User Authorization Denied
13011	invalid parameter	Invalid parameter
13012	parse NdefMessage failed	Failed to parse the parameter into NdefMessage.
13021	NFC discovery already started	NFC scanning has already commenced.
13018	NFC discovery has not started	Attempted to halt NFC scanning without initiating it.
13022	Tech already connected	The tag has been successfully connected.
13023	Tech has not connected	Attempted to disconnect without a tag being connected.
13013	NFC tag has not been discovered	No NFC tag was detected.
13014	invalid tech	Invalid tag technology.
13015	unavailable tech	Failed to acquire the corresponding technology from the tag.
13024	function not support	The current tag technology does not support this feature.
13017	system internal error	The related read-write operation has failed.
13016	connect fail	Connection failed

©2013-2022 Tencent Cloud. All rights reserved.

.isConnected

The method is utilized as MifareUltralight.isConnected().

Note:

This interface has been deprecated; developers can maintain the connection status independently.

Function Description: Verifies if a connection has been established.

Parameters and Description: Object object.

Attributes	Types	Default value	Required	Note
success	function	-	Not required	Callback function for successful interface invocation
fail	function	-	Not required	Callback function for unsuccessful interface invocation
complete	function	-	Not required	Callback function executed upon the completion of the interface invocation (both successful and unsuccessful invocations)

Error Code

Error Codes	Error Message	Note
13000	The device does not support NFC	-
13001	The system's NFC switch is not activated.	-
13010	An unknown error occurred.	-
13019	user is not authorized	User Authorization Denied
13011	invalid parameter	Invalid parameter
13012	parse NdefMessage failed	Failed to parse the parameter into NdefMessage.
13021	NFC discovery already started	NFC scanning has already commenced.

©2013-2022 Tencent Cloud. All rights reserved.

13018	NFC discovery has not started	Attempted to halt NFC scanning without initiating it.
13022	Tech already connected	The tag has been successfully connected.
13023	Tech has not connected	Attempted to disconnect without a tag being connected.
13013	NFC tag has not been discovered	No NFC tag was detected.
13014	invalid tech	Invalid tag technology.
13015	unavailable tech	Failed to acquire the corresponding technology from the tag.
13024	function not support	The current tag technology does not support this feature.
13017	system internal error	The related read-write operation has failed.
13016	connect fail	Connection failed

.getMaxTransceiveLength

The method is utilized as MifareUltralight.getMaxTransceiveLength().

Function Description: Retrieves the maximum transmission length.

Parameters and Description: Object object.

Attributes	Types	Default value	Required	Note
success	function	-	Not required	Callback function for successful interface invocation
fail	function	-	Not required	Callback function for unsuccessful interface invocation
complete	function	-	Not required	Callback function executed upon the completion of the interface invocation (both successful and unsuccessful invocations)

object.success Callback Function Parameters: Object res

Attributes	Types	Note
length	number	Maximum Transmission Length

©2013-2022 Tencent Cloud. All rights reserved.

Error Code

Error Codes	Error Message	Note
13000	The device does not support NFC	-
13001	The system's NFC switch is not activated.	-
13010	An unknown error occurred.	-
13019	user is not authorized	User Authorization Denied
13011	invalid parameter	Invalid parameter
13012	parse NdefMessage failed	Failed to parse the parameter into NdefMessage.
13021	NFC discovery already started	NFC scanning has already commenced.
13018	NFC discovery has not started	Attempted to halt NFC scanning without initiating it.
13022	Tech already connected	The tag has been successfully connected.
13023	Tech has not connected	Attempted to disconnect without a tag being connected.
13013	NFC tag has not been discovered	No NFC tag was detected.
13014	invalid tech	Invalid tag technology.
13015	unavailable tech	Failed to acquire the corresponding technology from the tag.
13024	function not support	The current tag technology does not support this feature.
13017	system internal error	The related read-write operation has failed.
13016	connect fail	Connection failed

.transceive

The method is employed as MifareUltralight.transceive(Object object).

©2013-2022 Tencent Cloud. All rights reserved.

Function Description: Transmits data.

Parameters and Description: Object object.

Attributes	Types	Default value	Required	Note
data	ArrayBuffer	-	Supported	Binary data required for transmission
success	function	-	Not required	Callback function for successful interface invocation
fail	function	-	Not required	Callback function for unsuccessful interface invocation
complete	function	-	Not required	Callback function executed upon the completion of the interface invocation (both successful and unsuccessful invocations)

object.success Callback Function Parameters: Object res.

Attributes	Types	Note
data	ArrayBuffer	-

Error Code

Error Codes	Error Message	Note
13000	The device does not support NFC	-
13001	The system's NFC switch is not activated.	-
13010	An unknown error occurred.	-
13019	user is not authorized	User Authorization Denied
13011	invalid parameter	Invalid parameter
13012	parse NdefMessage failed	Failed to parse the parameter into NdefMessage.
13021	NFC discovery already started	NFC scanning has already commenced.
13018	NFC discovery has not started	Attempted to halt NFC scanning without initiating it.
13022	Tech already connected	The tag has been successfully connected.

©2013-2022 Tencent Cloud. All rights reserved.

13023	Tech has not connected	Attempted to disconnect without a tag being connected.
13013	NFC tag has not been discovered	No NFC tag was detected.
13014	invalid tech	Invalid tag technology.
13015	unavailable tech	Failed to acquire the corresponding technology from the tag.
13024	function not support	The current tag technology does not support this feature.
13017	system internal error	The related read-write operation has failed.
13016	connect fail	Connection failed

©2013-2022 Tencent Cloud. All rights reserved.

Bluetooth - Low Energy Central Device

Last updated：2024-07-16 18:01:59

writeBLECharacteristicValue

This API is used via wx.writeBLECharacteristicValue(Object object).

Note:

Simultaneous multiple calls may potentially result in write failures.

While the mini program does not impose restrictions on the size of the data packet written, the system and Bluetooth device will limit the data size of a single Bluetooth 4.0 transmission. Since exceeding the maximum byte count may result in write errors, it is recommended not to exceed 20 bytes per write operation.

If data is written in a single instance that is excessively long, iOS may not provide any callbacks, including error callbacks.

On the Android platform, invoking this interface immediately after successfully calling wx.notifyBLECharacteristicValueChange may result in a system error coded 10008 on certain models.

Feature Description: Writes binary data into the characteristic value of a low-energy Bluetooth device. **Successful invocation requires the device's characteristic to support write operations**.

Parameter and Description: Object.

Attribute	Type	Required	Description
deviceId	string	Yes	Bluetooth Device ID
serviceId	string	Yes	UUID corresponding to the Bluetooth characteristic service
characteristicId	string	Yes	Bluetooth characteristic UUID
value	ArrayBuffer	Yes	The binary data of the characteristic value from the Bluetooth device.
writeType	string	No	Write mode setting for Bluetooth feature value, there are two modes, iOS priority write and Android priority writeNoResponse. legal value is: write: Mandatory write response, reporting error when unsupported. 'writeNoResponse': Mandatory write without response, reporting error when unsupported.

©2013-2022 Tencent Cloud. All rights reserved.

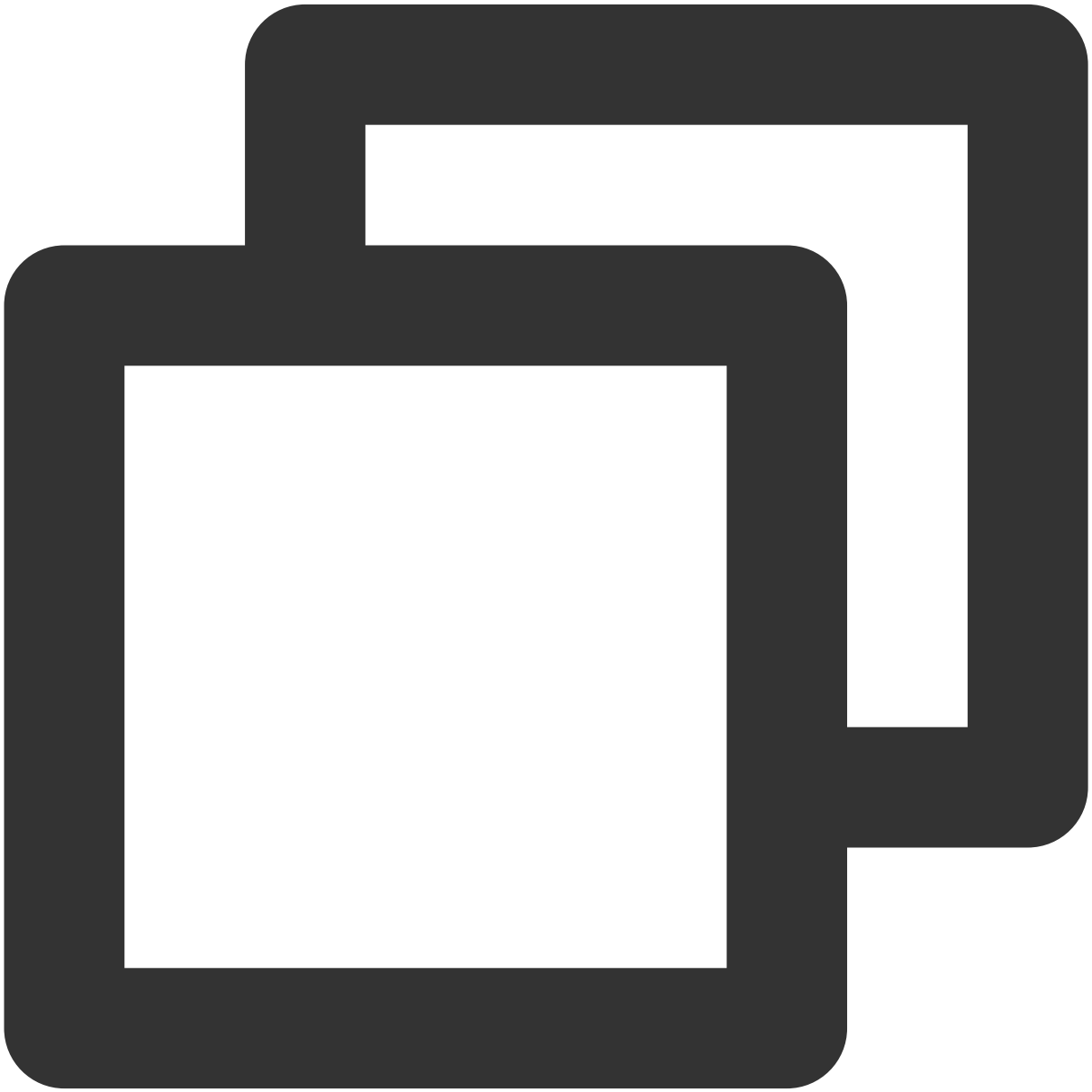
success	function	No	Callback Function of Successful Interface Call
fail	function	No	Callback Function of Successful Interface Call
complete	function	No	Callback function executed upon the completion of the interface invocation (both successful and unsuccessful invocations)

Error Code

Error code	Error Message	Description
0	ok	Normal
-1	already connect	Connected
10000	not init	Bluetooth Adapter Not Initialized
10001	not available	The current Bluetooth adapter is unavailable.
10002	no device	Specified Device Not Found
10003	connection fail	Connection failed.
10004	no service	Specified Service Not Found
10005	no characteristic	Specified characteristic not found
10006	no connection	The current connection has been disconnected.
10007	property not support	The current characteristic does not support this operation.
10008	system error	Exceptions Reported by All Other Systems
10009	system not support	Exclusive to Android system, and BLE is not supported on versions earlier than 4.3.
10012	operate time out	Connection timeout
10013	invalid_data	The connection deviceId is either null or incorrectly formatted.

Sample Code

©2013-2022 Tencent Cloud. All rights reserved.



```
// Transmit a hexadecimal data of 0x00 to the Bluetooth device.
let buffer = new ArrayBuffer(1)
let dataView = new DataView(buffer)
dataView.setUint8(0, 0)

wx.writeBLECharacteristicValue({
  // The deviceId here must be obtained through the getBluetoothDevices or onBlueto
  deviceId,
  // The ServiceId here must be obtained from the wx.getBLEDeviceServices interface
  serviceId,
  // The CharacteristicId Here must be obtained from the getBLEDeviceCharacteristic
```

©2013-2022 Tencent Cloud. All rights reserved.

```
  characteristicId,
  // The value here is of ArrayBuffer type.
  value: buffer,
  success (res) {
    console.log('writeBLECharacteristicValue success', res.errMsg)
  }
})
```

readBLECharacteristicValue

This API is used via wx.readBLECharacteristicValue(Object object).

Note:

Simultaneous multiple calls may potentially result in read failures.

The Information read from the interface must be obtained in the callback registered by the wx.onBLECharacteristicValueChange method.

Feature Description: Reads the binary data of the low-energy Bluetooth device characteristics. **Successful invocation is only possible if the device's characteristic supports read**.

Parameter and Description: Object.

Attribute	Type	Default value	Required	Description
deviceId	string	-	Yes	Bluetooth Device ID
serviceId	string	-	Yes	UUID corresponding to the Bluetooth characteristic service
characteristicId	string	-	Yes	Bluetooth characteristic UUID
success	function	-	No	Callback Function of Successful Interface Call
fail	function	-	No	Callback Function of Failing Interface Call
complete	function	-	No	Callback function executed upon the completion of the interface invocation (both successful and unsuccessful invocations)

Error Code

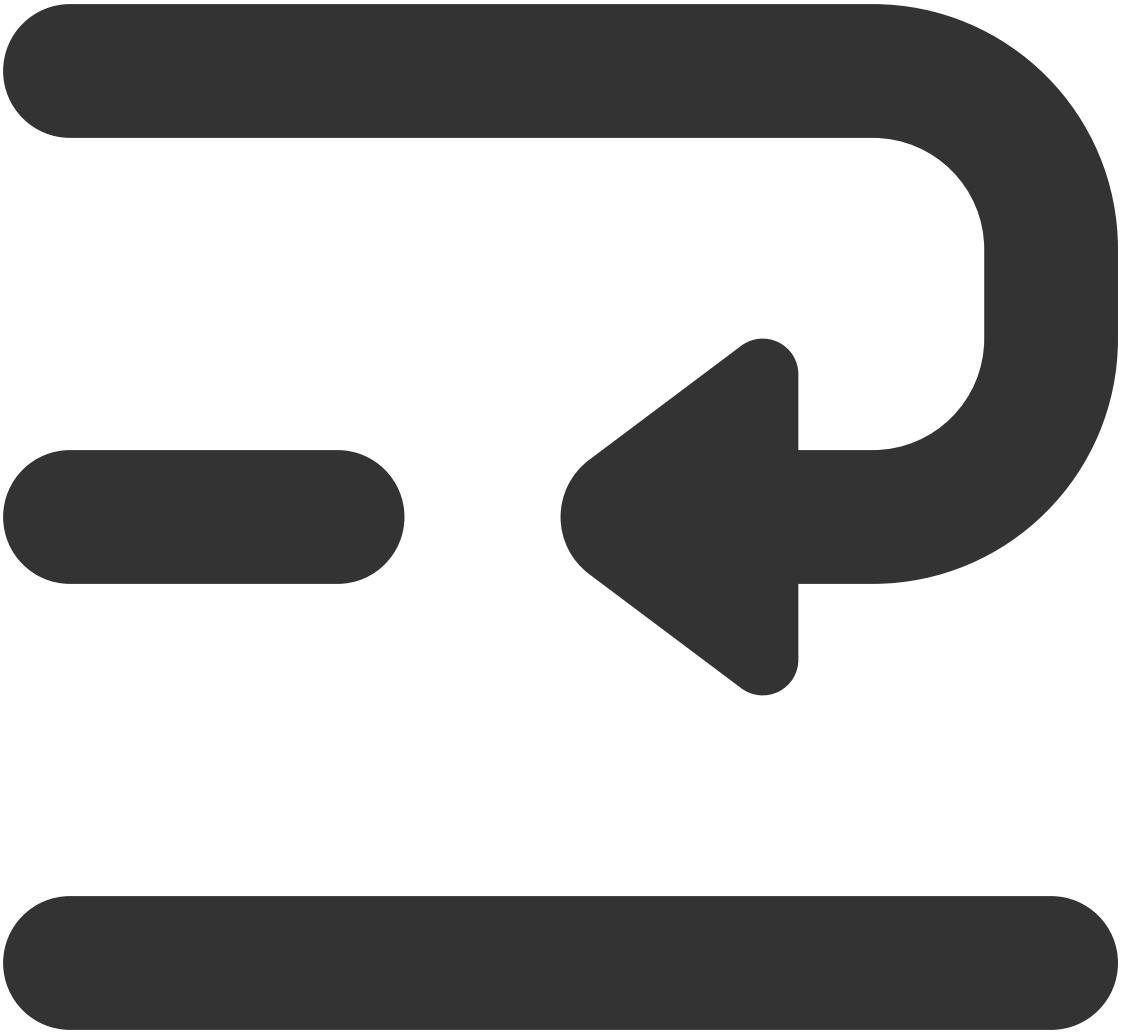
Error	Error Message	Description

©2013-2022 Tencent Cloud. All rights reserved.

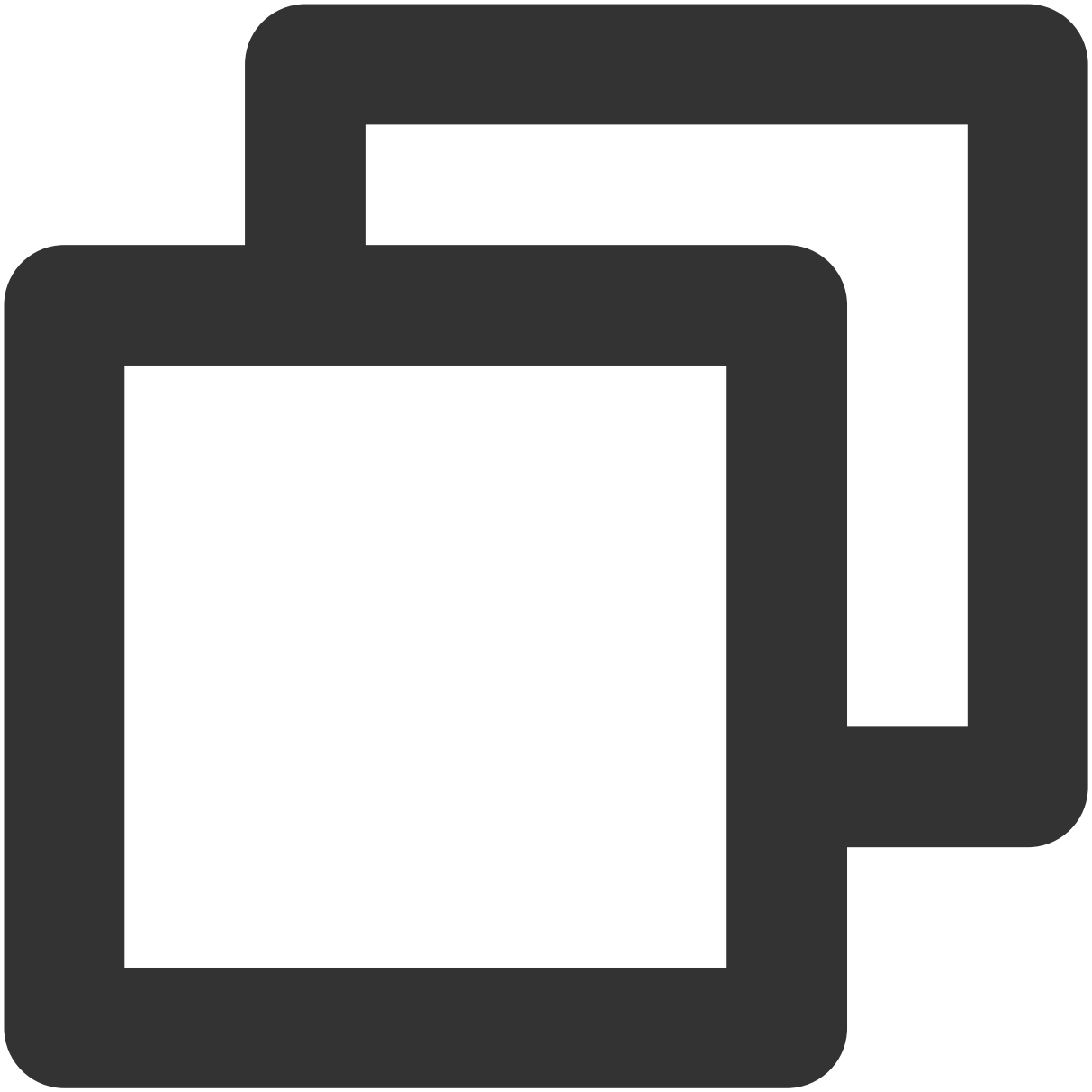
code		
0	ok	Normal
-1	already connect	Connected
10000	not init	Bluetooth Adapter Not Initialized
10001	not available	The current Bluetooth adapter is unavailable.
10002	no device	Specified Device Not Found
10003	connection fail	Connection failed.
10004	no service	Specified Service Not Found
10005	no characteristic	Specified characteristic not found
10006	no connection	The current connection has been disconnected.
10007	property not support	The current characteristic does not support this operation.
10008	system error	Exceptions Reported by All Other Systems
10009	system not support	Exclusive to Android system, and BLE is not supported on versions earlier than 4.3.
10012	operate time out	Connection timeout
10013	invalid_data	The connection deviceId is either null or incorrectly formatted.

Sample Code

©2013-2022 Tencent Cloud. All rights reserved.



©2013-2022 Tencent Cloud. All rights reserved.



```
// Retrieval is only possible in this callback.
wx.onBLECharacteristicValueChange(function(characteristic) {
  console.log('characteristic value comed:', characteristic)
})

wx.readBLECharacteristicValue({
  // The deviceId here must have already established a link with the corresponding
  deviceId,
  // The ServiceId here must be obtained from the wx.getBLEDeviceServices interface
  serviceId,
  // The CharacteristicId Here Must be Retrieved from the getBLEDeviceCharacteristi
```

©2013-2022 Tencent Cloud. All rights reserved.

```
  characteristicId,
  success (res) {
    console.log('readBLECharacteristicValue:', res.errCode)
  }
})
```

setBLEMTU

This API is used via wx.setBLEMTU(Object object).

Feature Description: Negotiates the setting of the maximum transmission unit (MTU) for low-energy Bluetooth devices. It needs to be called after wx.createBLEConnection has been successfully invoked. This is only effective on Android systems version 5.1 and later versions, and is not supported on iOS due to system limitations.
Parameter and Description: Object.

Attribute	Type	Default value	Required	Description
deviceId	string	-	Yes	Bluetooth Device ID
mtu	number	-	Yes	Maximum Transmission Unit. The setting range is within the interval (22,512), measured in bytes.
success	function	-	No	Callback Function of Successful Interface Call
fail	function	-	No	Callback Function of Failing Interface Call
complete	function	-	No	Callback function executed upon the completion of the interface invocation (both successful and unsuccessful invocations)

Parameters for object.success callback function: Object res.

Attribute	Type	Description
mtu	number	The final negotiated MTU value, which is consistent with the parameter passed in. It is supported from Android client 8.0.9 onwards.

getBLEMTU

©2013-2022 Tencent Cloud. All rights reserved.

This API is used via wx.getBLEMTU(Object object).

Note:

Within the mini program, the MTU is ATT_MTU, which includes the length of the Op-Code and Attribute Handle. The actual data transmission length is ATT_MTU - 3.

In the iOS system, the MTU is a fixed value; in the Android system, the MTU changes after successful system negotiation. It is recommended to use wx.onBLEMTUChange for monitoring.

Feature Description: Gets the maximum transmission unit of low-energy Bluetooth devices. It needs to be called after wx.createBLEConnection has been successfully invoked.

Parameter and Description: Object.

Attribute	Type	Default value	Required	Description
deviceId	string	-	Yes	Bluetooth Device ID
writeType	string	write	No	Write mode (specific to iOS). Valid values are: write: Write with response. writeNoResponse: Write without response.
success	function	-	No	Callback Function of Successful Interface Call
fail	function	-	No	Callback Function of Successful Interface Call
complete	function	-	No	Callback function executed upon the completion of the interface invocation (both successful and unsuccessful invocations)

Parameters for object.success callback function: Object res.

Attribute	Type	Description
mtu	number	Maximum Transmission Unit

Error Code

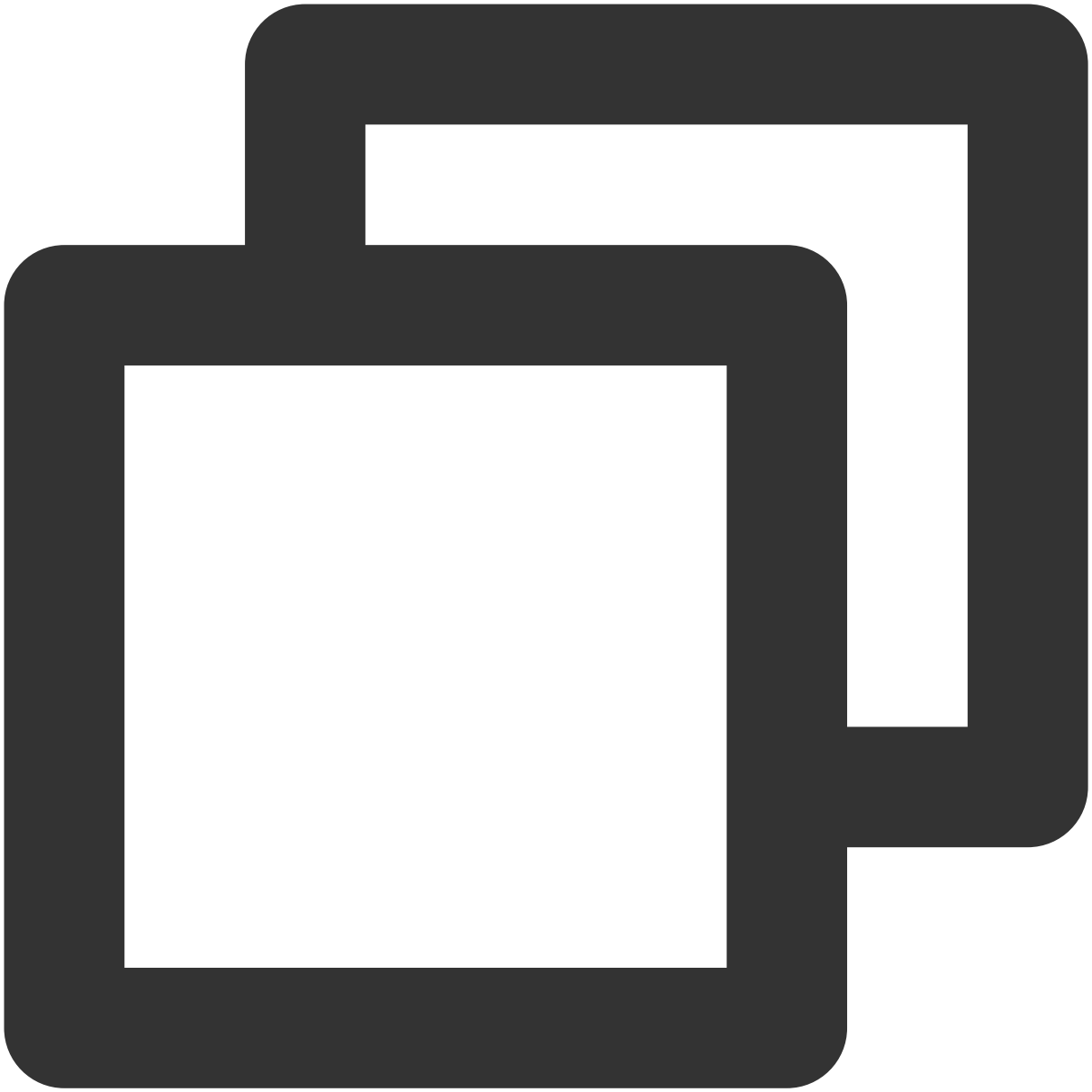
Error code	Error Message	Description
0	ok	Normal
-1	already connect	Connected
10000	not init	Bluetooth Adapter Not Initialized

©2013-2022 Tencent Cloud. All rights reserved.

10001	not available	The current Bluetooth adapter is unavailable.
10002	no device	Specified Device Not Found
10003	connection fail	Connection failed.
10004	no service	Specified Service Not Found
10005	no characteristic	Specified characteristic not found
10006	no connection	The current connection has been disconnected.
10007	property not support	The current characteristic does not support this operation.
10008	system error	Exceptions Reported by All Other Systems
10009	system not support	Exclusive to Android system, and BLE is not supported on versions earlier than 4.3.
10012	operate time out	Connection timeout
10013	invalid_data	The connection deviceId is either null or incorrectly formatted.

Sample Code

©2013-2022 Tencent Cloud. All rights reserved.



```
wx.getBLEMTU({
  deviceId: '',
  writeType: 'write',
  success (res) {
    console.log(res)
  }
})
```

©2013-2022 Tencent Cloud. All rights reserved.

onBLEMTUChange

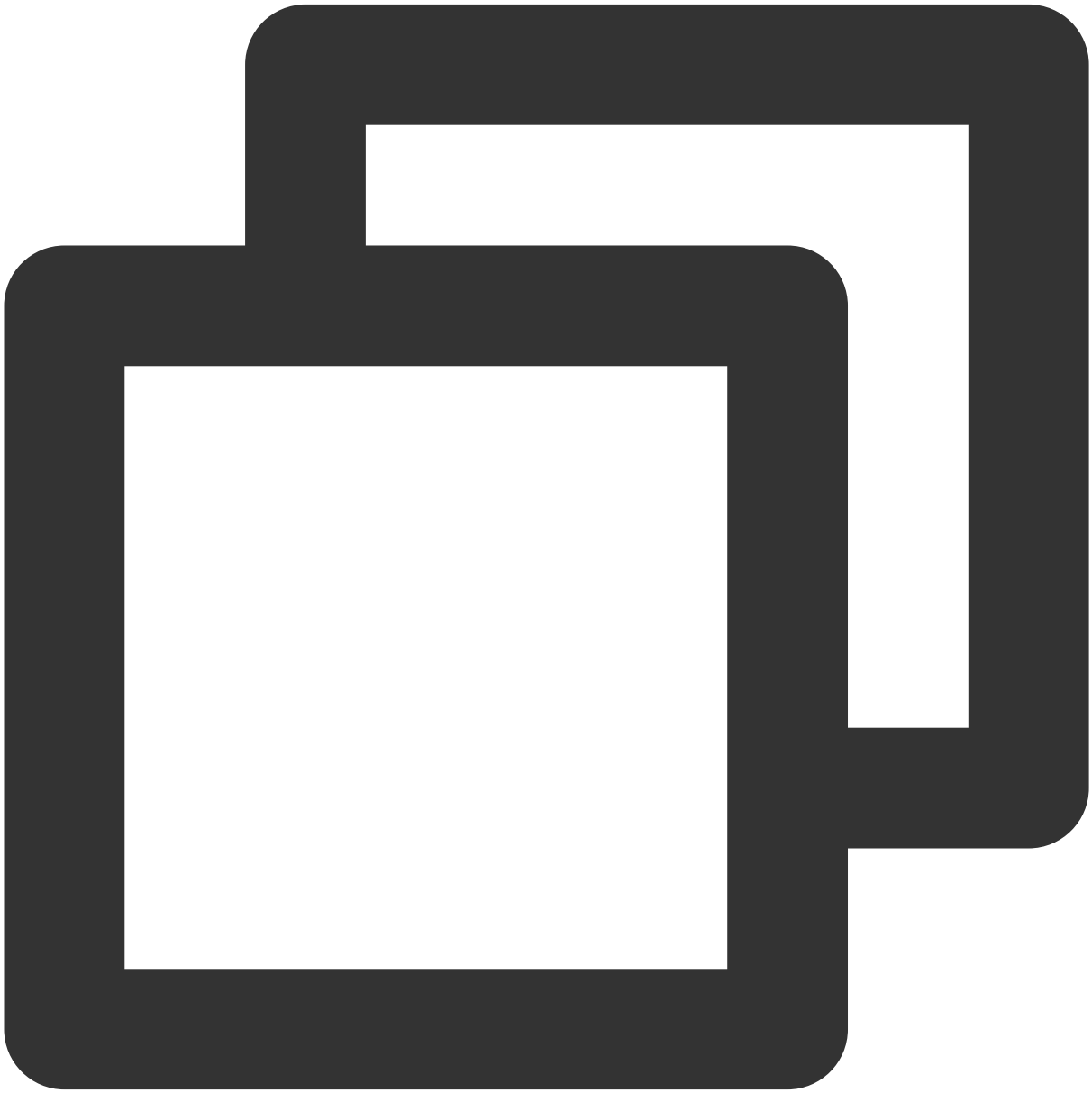
This API is used via wx.onBLEMTUChange(function listener).

Feature Description: Monitors events of MTU changes in low-energy Bluetooth devices (only triggered on Android).
Parameter and Description: Object res parameter, function listener, the listener function for events of MTU changes in low-energy Bluetooth devices.

Attribute	Type	Description
deviceId	string	Bluetooth Device ID
mtu	number	Maximum Transmission Unit

Sample Code

©2013-2022 Tencent Cloud. All rights reserved.



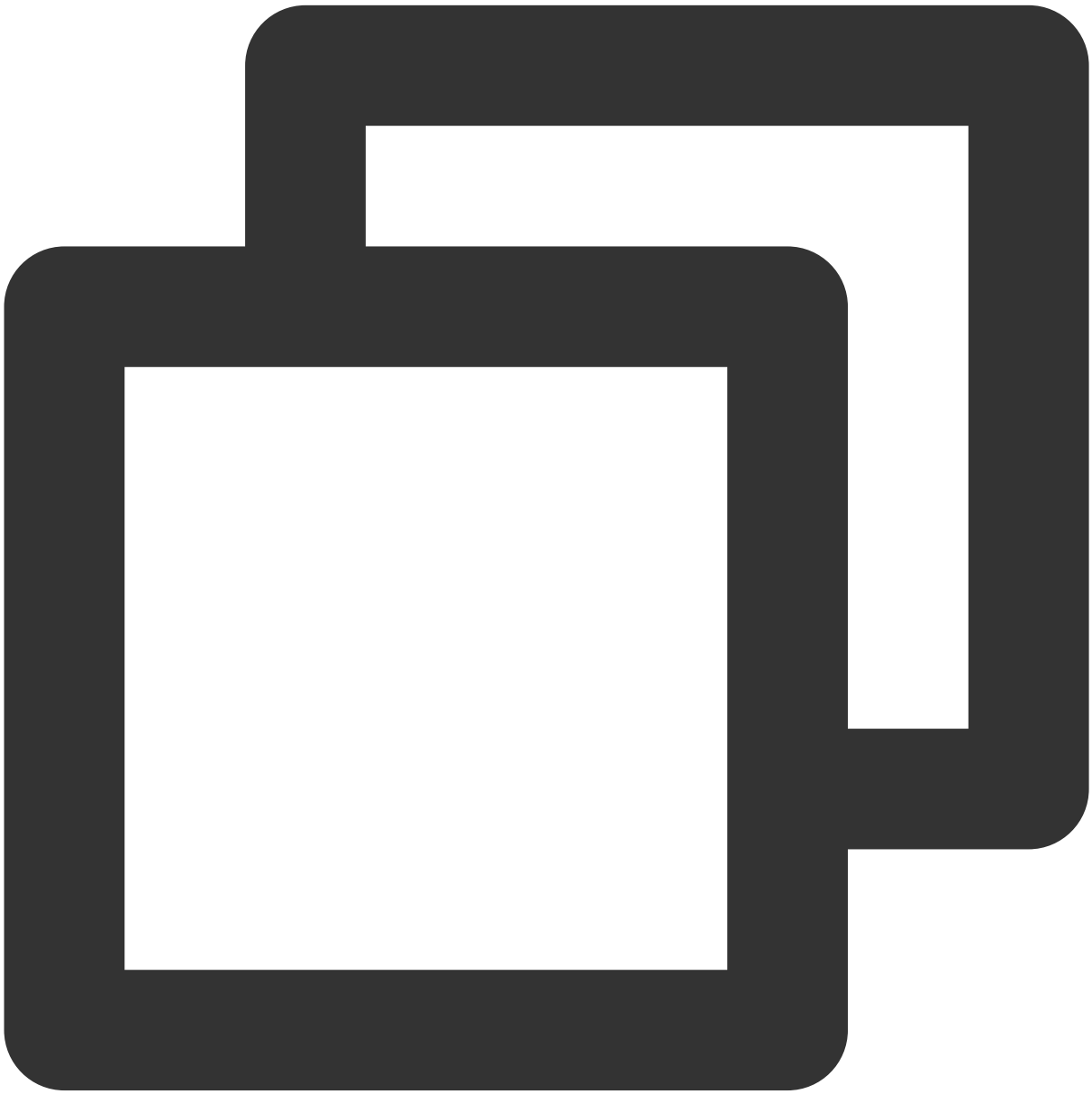
```
wx.onBLEMTUChange(function (res) {
  console.log('bluetooth mtu is', res.mtu)
})
```

offBLEMTUChange

This method is used via wx.offBLEMTUChange(function listener).

©2013-2022 Tencent Cloud. All rights reserved.

Feature Description: Removes the listener function for events of MTU changes in low-energy Bluetooth devices.
Parameter and Description: function listener, the listener function passed in by onBLEMTUChange. If this parameter is not passed in, all listener functions will be removed.
Sample Code



```
const listener = function (res) { console.log(res) }

wx.onBLEMTUChange(listener)
wx.offBLEMTUChange(listener) // The same function object as the listener must be pa
```

©2013-2022 Tencent Cloud. All rights reserved.

onBLEConnectionStateChange

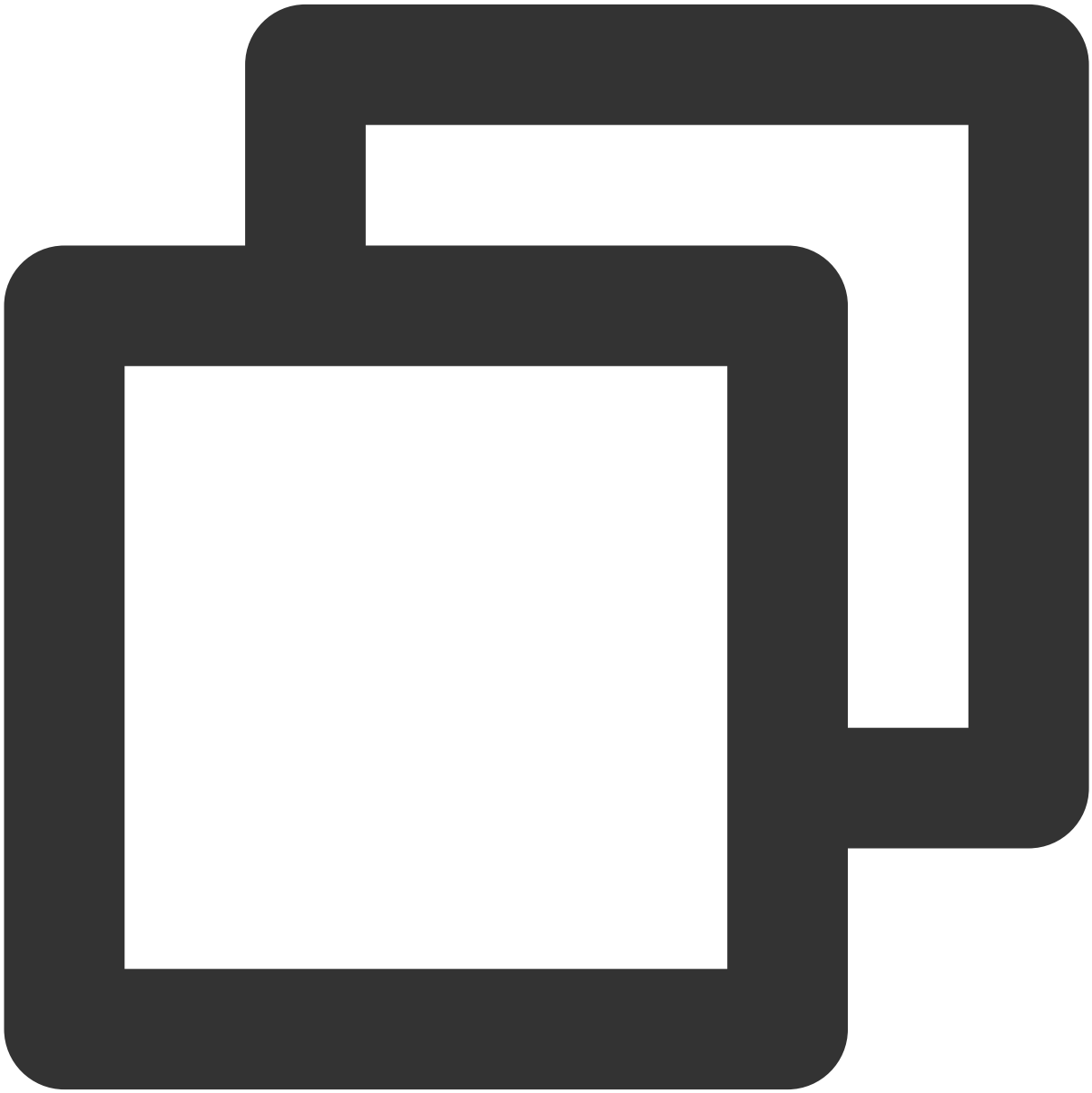
This API is used via wx.onBLEConnectionStateChange(function listener).

Feature Description: Monitors events of changes to low-energy Bluetooth connection status. This includes active connections or disconnections by developers, device loss, exceptional disconnections, and so on.
Parameters and Description: function listener, the Bluetooth low power connection state change event listener function, the parameter Object res is as follows:

Attribute	Type	Description
deviceId	string	Bluetooth Device ID
connected	boolean	Whether it is in connected status.

Sample Code

©2013-2022 Tencent Cloud. All rights reserved.



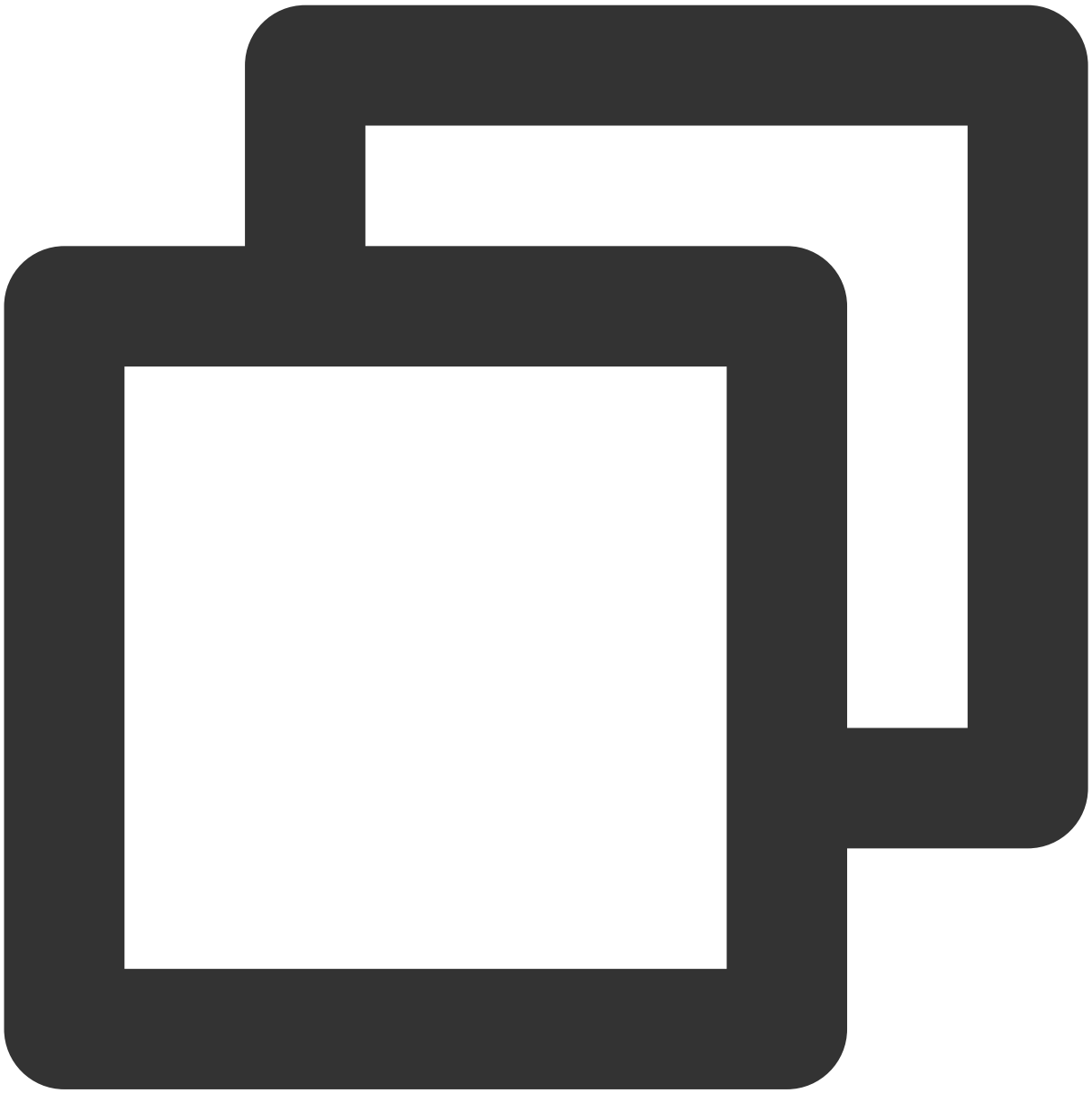
```
wx.onBLEConnectionStateChange(function(res) {
  // This method callback can be used to handle exceptional situations such as unex
  console.log(device ${res.deviceId} state has changed, connected: ${res.connected}
})
```

offBLEConnectionStateChange

©2013-2022 Tencent Cloud. All rights reserved.

This API is used via wx.offBLEConnectionStateChange(function listener).

Feature Description: Removes the listener function for events of changes to low-energy Bluetooth connection status.
Parameter and Description: function listener, the listener function passed in by onBLEConnectionStateChange. If this parameter is not passed in, all listener functions will be removed.
Sample Code



```
const listener = function (res) { console.log(res) }
```

©2013-2022 Tencent Cloud. All rights reserved.

```
wx.onBLEConnectionStateChange(listener)
wx.offBLEConnectionStateChange(listener) // The same function object as the listene
```

onBLECharacteristicValueChange

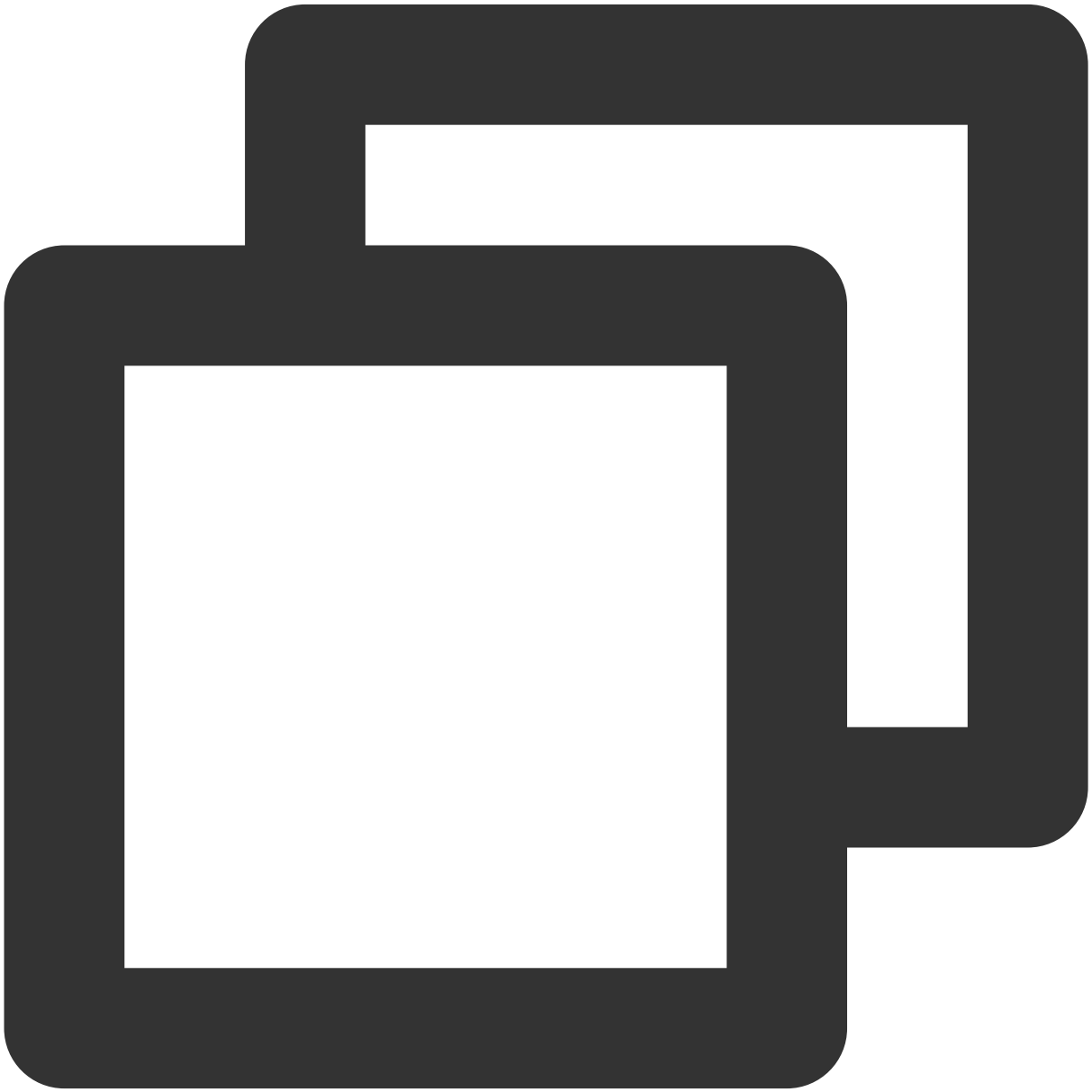
This API is used via wx.onBLECharacteristicValueChange(function listener).

Feature Description: Monitors events of characteristic value changes for low-energy Bluetooth devices. You must first call the wx.notifyBLECharacteristicValueChange interface to receive notifications pushed by the device.
Parameter and Description: function listener, the listener function for the feature value change event of the Bluetooth low power device, with the following parameters Object res:

Attribute	Type	Description
deviceId	string	Bluetooth Device ID
serviceId	string	UUID corresponding to the Bluetooth characteristic service
characteristicId	string	Bluetooth characteristic UUID
value	ArrayBuffer	The latest characteristic value

Sample Code

©2013-2022 Tencent Cloud. All rights reserved.



```
// Example of converting ArrayBuffer to hexadecimal string
function ab2hex(buffer) {
  let hexArr = Array.prototype.map.call(
    new Uint8Array(buffer),
    function(bit) {
      return ('00' + bit.toString(16)).slice(-2)
    }
  )
  return hexArr.join('');
}
wx.onBLECharacteristicValueChange(function(res) {
```

©2013-2022 Tencent Cloud. All rights reserved.

```
  console.log(characteristic ${res.characteristicId} has changed, now is ${res.valu
  console.log(ab2hex(res.value))
})
```

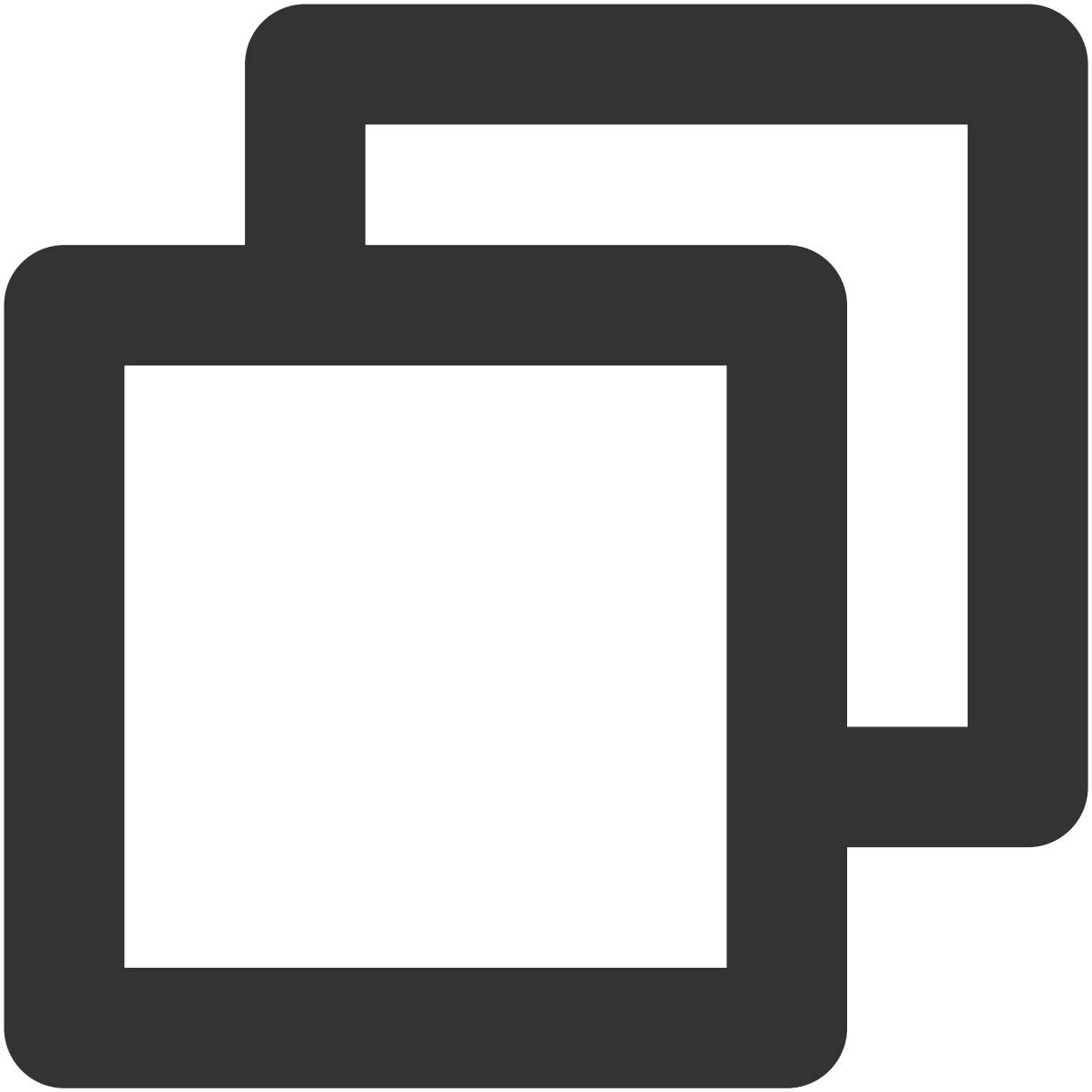
offBLECharacteristicValueChange

This API is used via wx.offBLECharacteristicValueChange().

Feature Description: Removes all listener functions for events of characteristic value changes for low-energy Bluetooth devices.

Sample Code

©2013-2022 Tencent Cloud. All rights reserved.



```
wx.offBLECharacteristicValueChange()
```

notifyBLECharacteristicValueChange

This API is used via wx.notifyBLECharacteristicValueChange(Object object).

©2013-2022 Tencent Cloud. All rights reserved.

Note:

After the subscription operation is successful, the device needs to actively update the characteristic value to trigger the wx.onBLECharacteristicValueChange callback.

On the Android platform, invoking the wx.writeBLECharacteristicValue interface immediately after successfully calling this interface may result in a system error coded 10008 on certain models.

The device's characteristic must support "notify" or "indicate" for the call to be successful. Additionally, wx.notifyBLECharacteristicValueChange must be enabled first to monitor the device's characteristicValueChange event.

Feature Description: Enables the "notify" feature when the characteristic value of low-energy Bluetooth device changes, subscribing to the characteristics.

Parameter and Description: Object.

Attribute	Type	Default value	Required	Description
deviceId	string	-	Yes	Bluetooth Device ID
serviceId	string	-	Yes	UUID corresponding to the Bluetooth characteristic service
characteristicId	string	-	Yes	Bluetooth characteristic UUID
state	boolean	-	Yes	Whether to enable "notify"
type	string	indication	No	Set the characteristic subscription type, whose valid values include "notification" and "indication".
success	function	-	No	Callback Function of Successful Interface Call
fail	function	-	No	Callback Function of Failing Interface Call
complete	function	-	No	Callback function executed upon the completion of the interface invocation (both successful and unsuccessful invocations)

Error Code

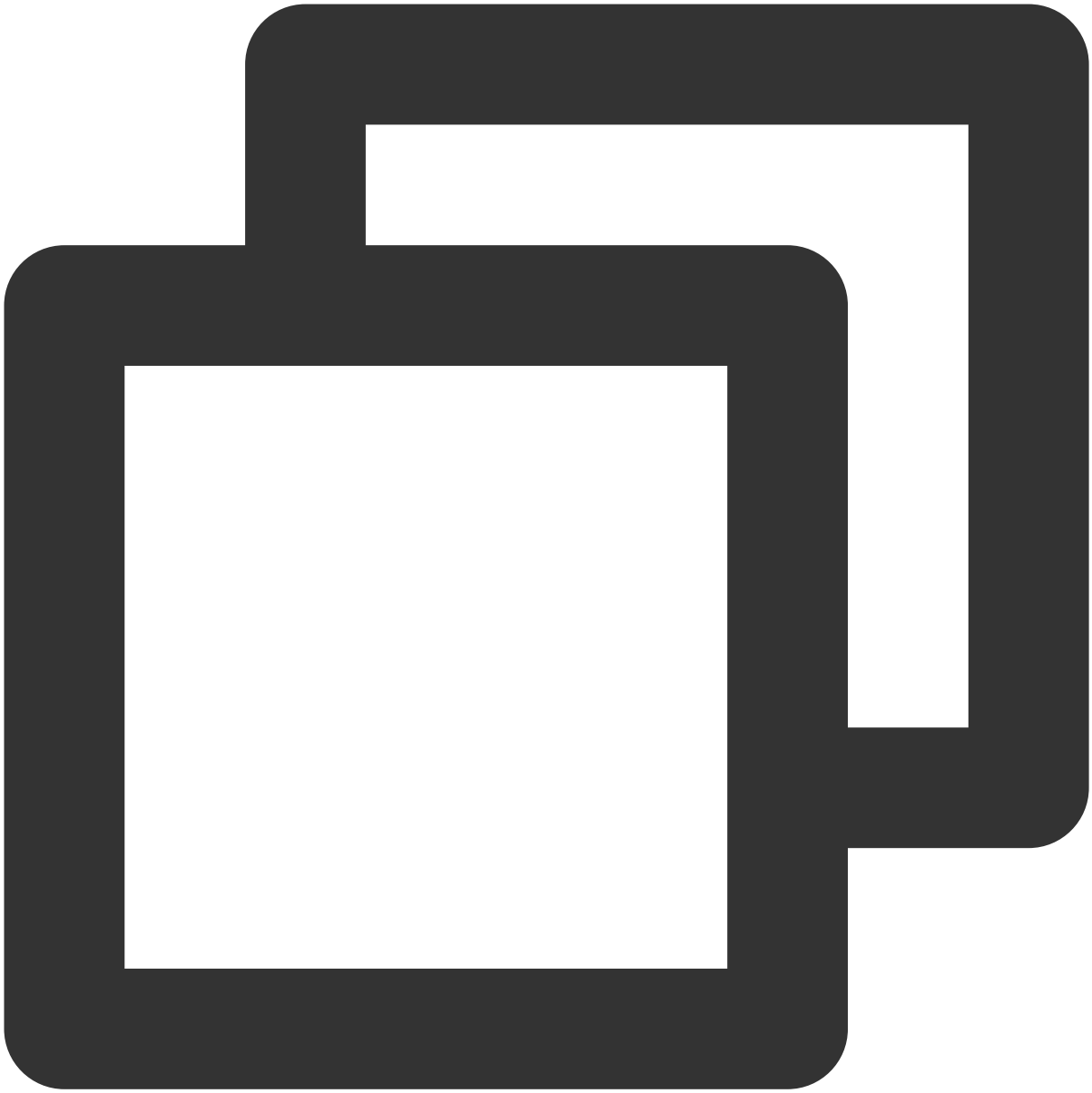
Error code	Error Message	Description
0	ok	Normal
-1	already connect	Connected
10000	not init	Bluetooth Adapter Not Initialized
10001	not available	The current Bluetooth adapter is unavailable.

©2013-2022 Tencent Cloud. All rights reserved.

10002	no device	Specified Device Not Found
10003	connection fail	Connection failed.
10004	no service	Specified Service Not Found
10005	no characteristic	Specified characteristic not found
10006	no connection	The current connection has been disconnected.
10007	property not support	The current characteristic does not support this operation.
10008	system error	Exceptions Reported by All Other Systems
10009	system not support	Exclusive to Android system, and BLE is not supported on versions earlier than 4.3.
10012	operate time out	Connection timeout
10013	invalid_data	The connection deviceId is either null or incorrectly formatted.

Sample Code

©2013-2022 Tencent Cloud. All rights reserved.



```
// Transmit a hexadecimal data of 0x00 to the Bluetooth device.
let buffer = new ArrayBuffer(1)
let dataView = new DataView(buffer)
dataView.setUint8(0, 0)

wx.writeBLECharacteristicValue({
  // The deviceId here must be obtained through the getBluetoothDevices or onBlueto
  deviceId,
  // The ServiceId here must be obtained from the wx.getBLEDeviceServices interface
  serviceId,
  // The CharacteristicId Here Must be Retrieved from the getBLEDeviceCharacteristi
```

©2013-2022 Tencent Cloud. All rights reserved.

```
  characteristicId,
  // The Value Here is of ArrayBuffer Type
  value: buffer,
  success (res) {
    console.log('writeBLECharacteristicValue success', res.errMsg)
  }
})
```

getBLEDeviceServices

This API is used via wx.getBLEDeviceServices(Object object).

Feature Description: Gets all services from the low-energy Bluetooth device.

Parameter and Description: Object.

Attribute	Type	Default value	Required	Description
deviceId	string	-	Yes	Bluetooth device id. A connection must have already been established through wx.createBLEConnection.
success	function	-	No	Callback Function of Successful Interface Call
fail	function	-	No	Callback Function of Failing Interface Call
complete	function	-	No	Callback function executed upon the completion of the interface invocation (both successful and unsuccessful invocations)

Parameters for object.success callback function: Object res.

Attribute	Type	Description
services	Array.<Object>	Device service list

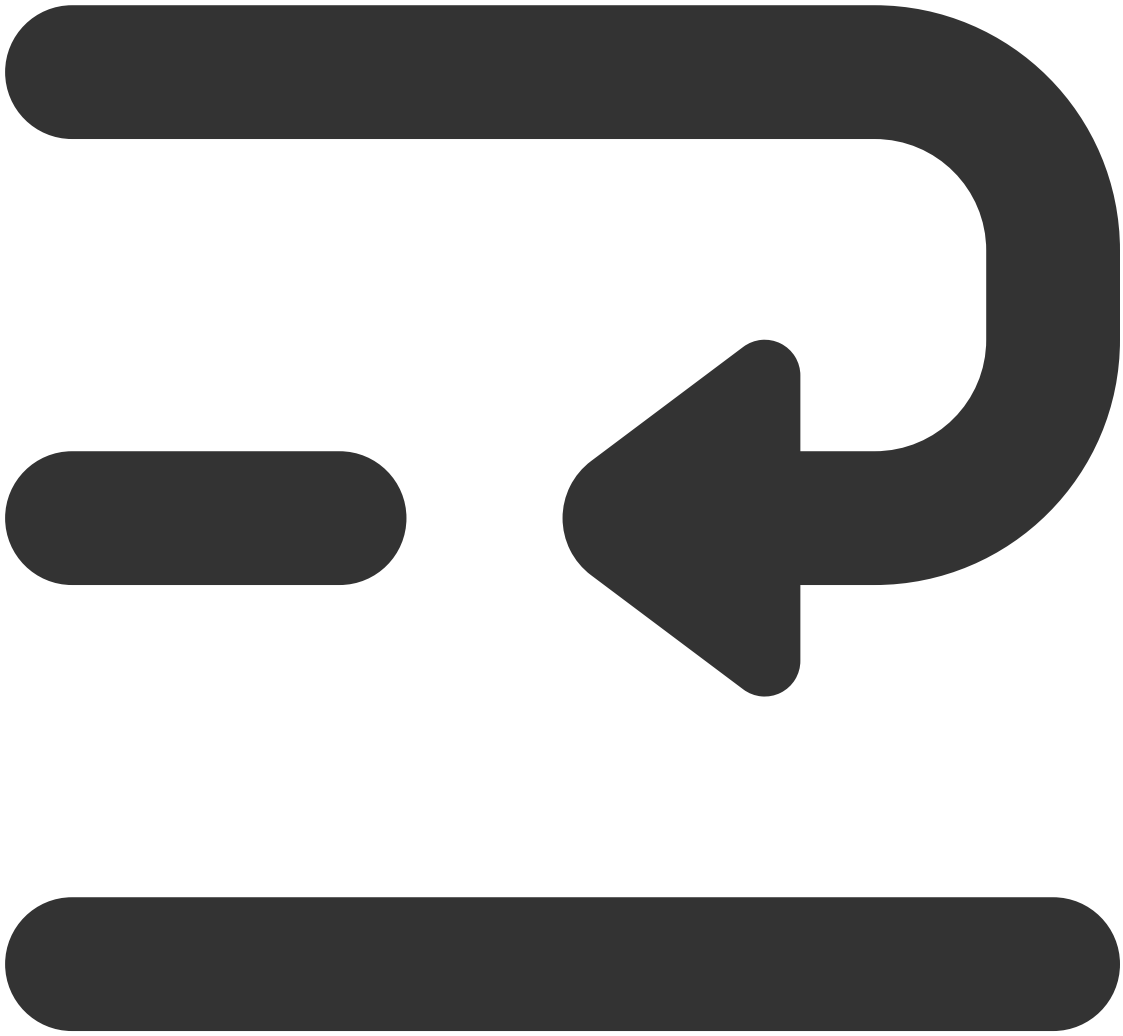
Structure attribute of service

Attribute	Type	Description
uuid	string	Bluetooth device service UUID
isPrimary	boolean	Whether the service is a primary

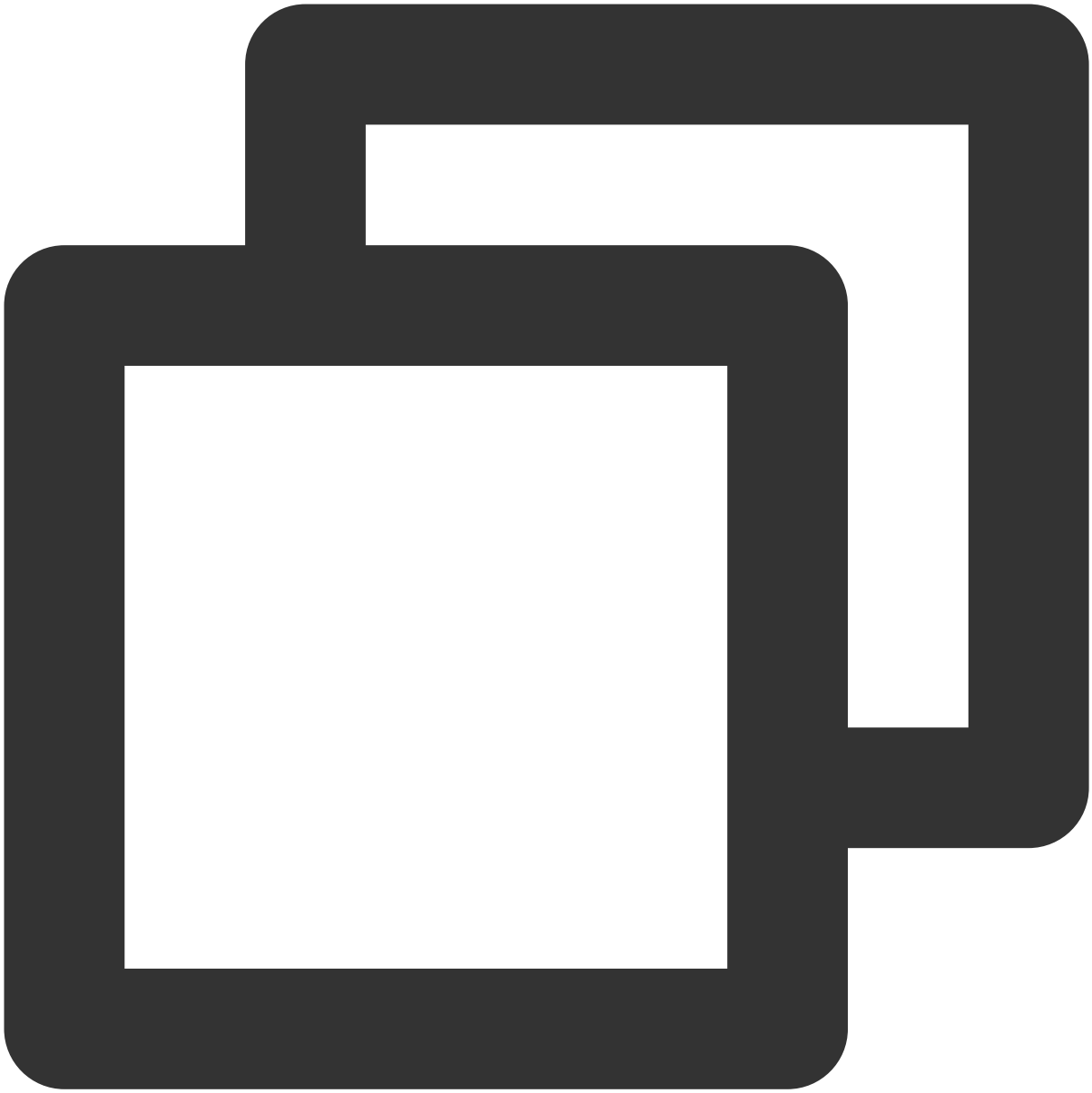
©2013-2022 Tencent Cloud. All rights reserved.

		service.

Sample Code



©2013-2022 Tencent Cloud. All rights reserved.



```
wx.getBLEDeviceServices({
    // The deviceId here must have already established a link with the correspondin
  deviceId,
  success (res) {
    console.log('device services:', res.services)
  }
})
```

©2013-2022 Tencent Cloud. All rights reserved.

getBLEDeviceRSSI

This API is used via wx.getBLEDeviceRSSI(Object object).

Feature Description: Gets the signal strength (Received Signal Strength Indication, RSSI) of the low-energy Bluetooth device.

Parameter and Description: Object.

Attribute	Type	Default value	Required	Description
deviceId	string	-	Yes	Bluetooth Device ID
success	function	-	No	Callback Function of Successful Interface Call
fail	function	-	No	Callback Function of Failing Interface Call
complete	function	-	No	Callback function executed upon the completion of the interface invocation (both successful and unsuccessful invocations)

Parameters for object.success callback function: Object res.

Attribute	Type	Description
RSSI	Number	Signal strength, measured in dBm

getBLEDeviceCharacteristics

This API is used via wx.getBLEDeviceCharacteristics(Object object).

Feature Description: Gets all characteristics within a service from a low-energy Bluetooth device.

Parameter and Description: Object.

Attribute	Type	Default value	Required	Description
deviceId	string	-	Yes	Bluetooth device id. A connection must have already been established through wx.createBLEConnection.

©2013-2022 Tencent Cloud. All rights reserved.

serviceId	string	-	Yes	Bluetooth service UUID. It is necessary to first invoke wx.getBLEDeviceServices to get it.
success	function	-	No	Callback Function of Successful Interface Call
fail	function	-	No	Callback Function of Failing Interface Call
complete	function	-	No	Callback function executed upon the completion of the interface invocation (both successful and unsuccessful invocations)

Parameters for object.success callback function: Object res.

Attribute	Type	Description
characteristics	Array.<Object>	Device characteristic value list

Structure attributes of characteristics

Structure attributes	Type	Description
uuid	string	Bluetooth device characteristic UUID
properties	Object	Operation types supported by this characteristic

Structure attributes of properties

Attribute	Type	Description
read	boolean	Whether this characteristic supports the read operation.
write	boolean	Whether this characteristic supports the write operation.
notify	boolean	Whether this characteristic supports the notify operation.
indicate	boolean	Whether this characteristic supports the indicate operation.
writeNoResponse	boolean	Whether this characteristic supports the writeNoResponse operation.
writeDefault	boolean	Whether this characteristic supports the writeDefault operation.

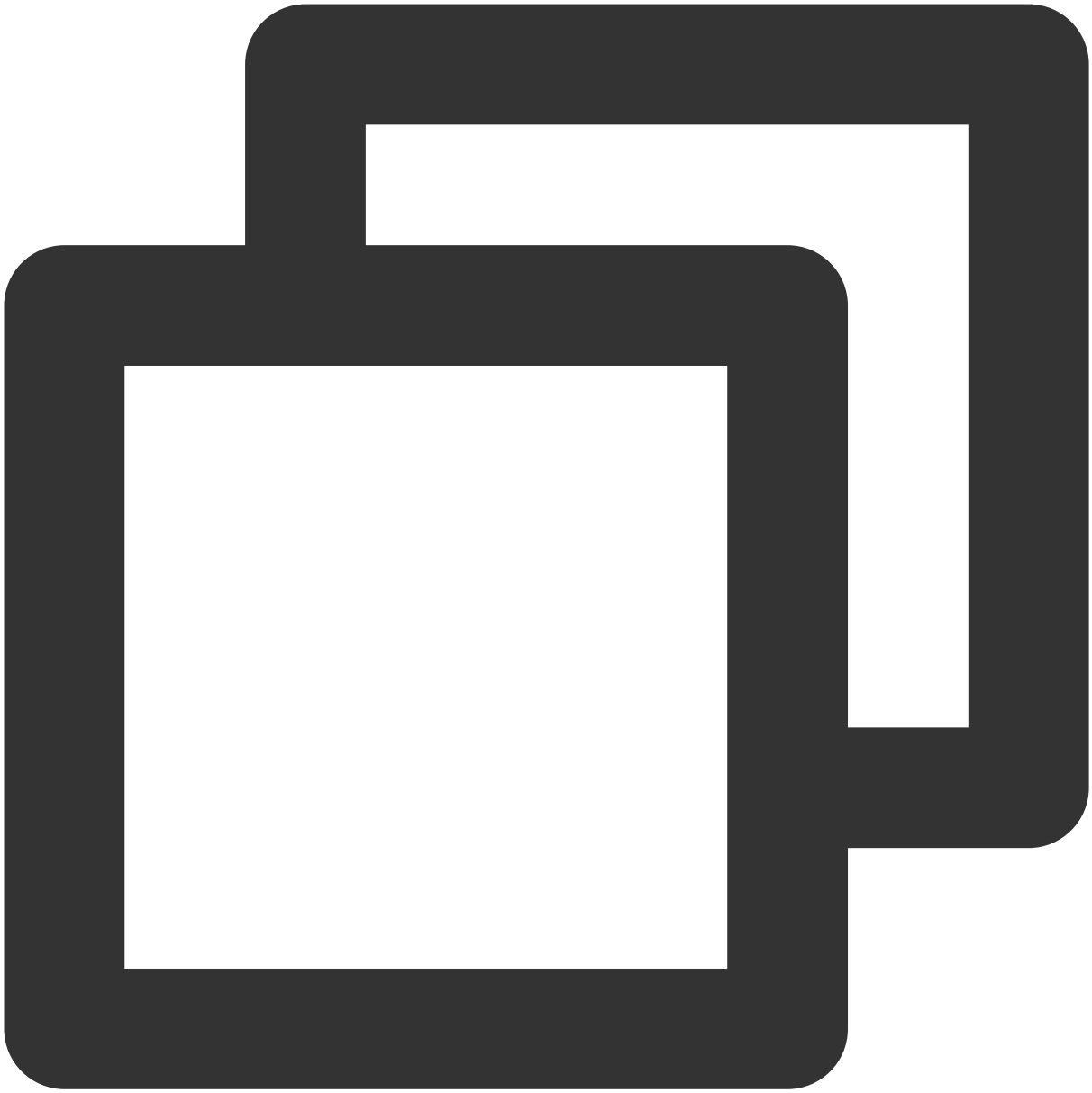
©2013-2022 Tencent Cloud. All rights reserved.

Error Code

Error code	Error Message	Description
0	ok	Normal
-1	already connect	Connected
10000	not init	Bluetooth Adapter Not Initialized
10001	not available	The current Bluetooth adapter is unavailable.
10002	no device	Specified Device Not Found
10003	connection fail	Connection failed.
10004	no service	Specified Service Not Found
10005	no characteristic	Specified characteristic not found
10006	no connection	The current connection has been disconnected.
10007	property not support	The current characteristic does not support this operation.
10008	system error	Exceptions Reported by All Other Systems
10009	system not support	Exclusive to Android system, and BLE is not supported on versions earlier than 4.3.
10012	operate time out	Connection timeout
10013	invalid_data	The connection deviceId is either null or incorrectly formatted.

Sample Code

©2013-2022 Tencent Cloud. All rights reserved.



```
wx.getBLEDeviceCharacteristics({
  // The deviceId here needs to have already established a link with the correspond
  deviceId,
  // The ServiceId here must be obtained from the wx.getBLEDeviceServices interface
  serviceId,
  success (res) {
    console.log('device getBLEDeviceCharacteristics:', res.characteristics)
  }
})
```

©2013-2022 Tencent Cloud. All rights reserved.

createBLEConnection

This API is used via wx.createBLEConnection(Object object).

Note:

Please ensure to pair the calls to wx.createBLEConnection and wx.closeBLEConnection interfaces as much as possible. On Android, if wx.createBLEConnection is repeatedly called to create a connection, it may cause the system to hold multiple connection instances of the same device, resulting in the inability to truly disconnect from the device when calling closeBLEConnection.

The Bluetooth connection may disconnect at any time. It is recommended to monitor wx.onBLEConnectionStateChange callback events and perform reconnection operations as needed when the Bluetooth device is disconnected.

If data read and write operation interfaces are called on unconnected devices or devices that have disconnected, an error coded 10006 will be returned. It is recommended to perform reconnection operations.

Feature Description: Connects to a low-energy Bluetooth device. If the mini program has previously searched for a certain Bluetooth device and successfully established a connection, you can directly pass in the deviceId obtained from the previous search to attempt to connect to the device, eliminating the need for another search operation.

Parameter and Description: Object.

Attribute	Type	Default value	Required	Description
deviceId	string	-	Yes	Bluetooth Device ID
timeout	number	-	No	Timeout period, in milliseconds (ms). If left blank, it indicates that there will be no timeout.
success	function	-	No	Callback Function of Successful Interface Call
fail	function	-	No	Callback Function of Failing Interface Call
complete	function	-	No	Callback function executed upon the completion of the interface invocation (both successful and unsuccessful invocations)

Error Code

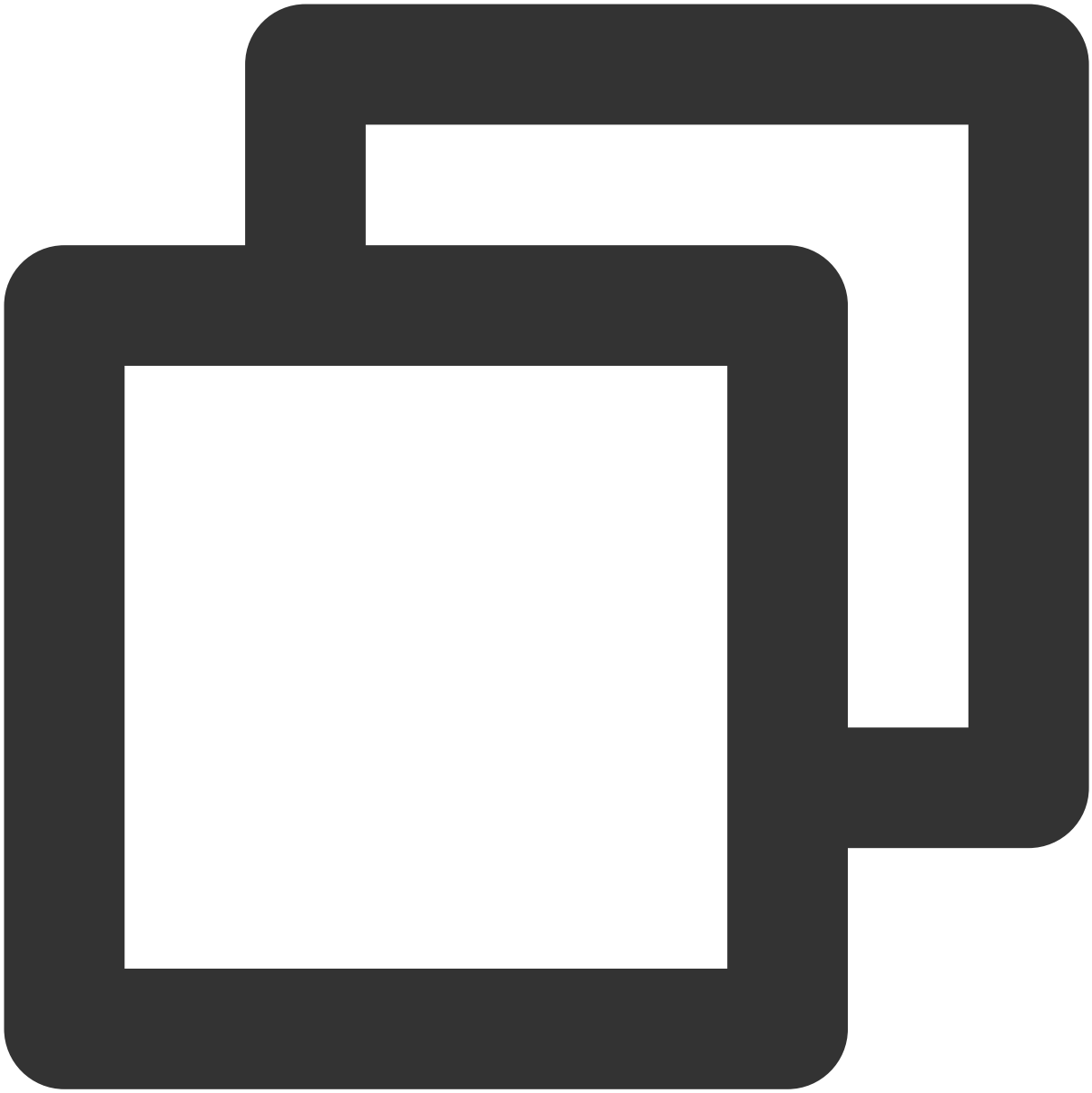
Error code	Error Message	Description
0	ok	Normal

©2013-2022 Tencent Cloud. All rights reserved.

-1	already connect	Connected
10000	not init	Bluetooth Adapter Not Initialized
10001	not available	The current Bluetooth adapter is unavailable.
10002	no device	Specified Device Not Found
10003	connection fail	Connection failed.
10004	no service	Specified Service Not Found
10005	no characteristic	Specified characteristic not found
10006	no connection	The current connection has been disconnected.
10007	property not support	The current characteristic does not support this operation.
10008	system error	Exceptions Reported by All Other Systems
10009	system not support	Exclusive to Android system, and BLE is not supported on versions earlier than 4.3.
10012	operate time out	Connection timeout
10013	invalid_data	The connection deviceId is either null or incorrectly formatted.

Sample Code

©2013-2022 Tencent Cloud. All rights reserved.



```
wx.createBLEConnection({
  deviceId,
  success (res) {
    console.log(res)
  }
})
```

closeBLEConnection

©2013-2022 Tencent Cloud. All rights reserved.

This API is used wx.closeBLEConnection(Object object).

Feature Description: Disconnects from the low-energy Bluetooth device.

Parameter and Description: Object.

Attribute	Type	Default value	Required	Description
deviceId	string	-	Yes	Bluetooth Device ID
success	function	-	No	Callback Function of Successful Interface Call
fail	function	-	No	Callback Function of Failing Interface Call
complete	function	-	No	Callback function executed upon the completion of the interface invocation (both successful and unsuccessful invocations)

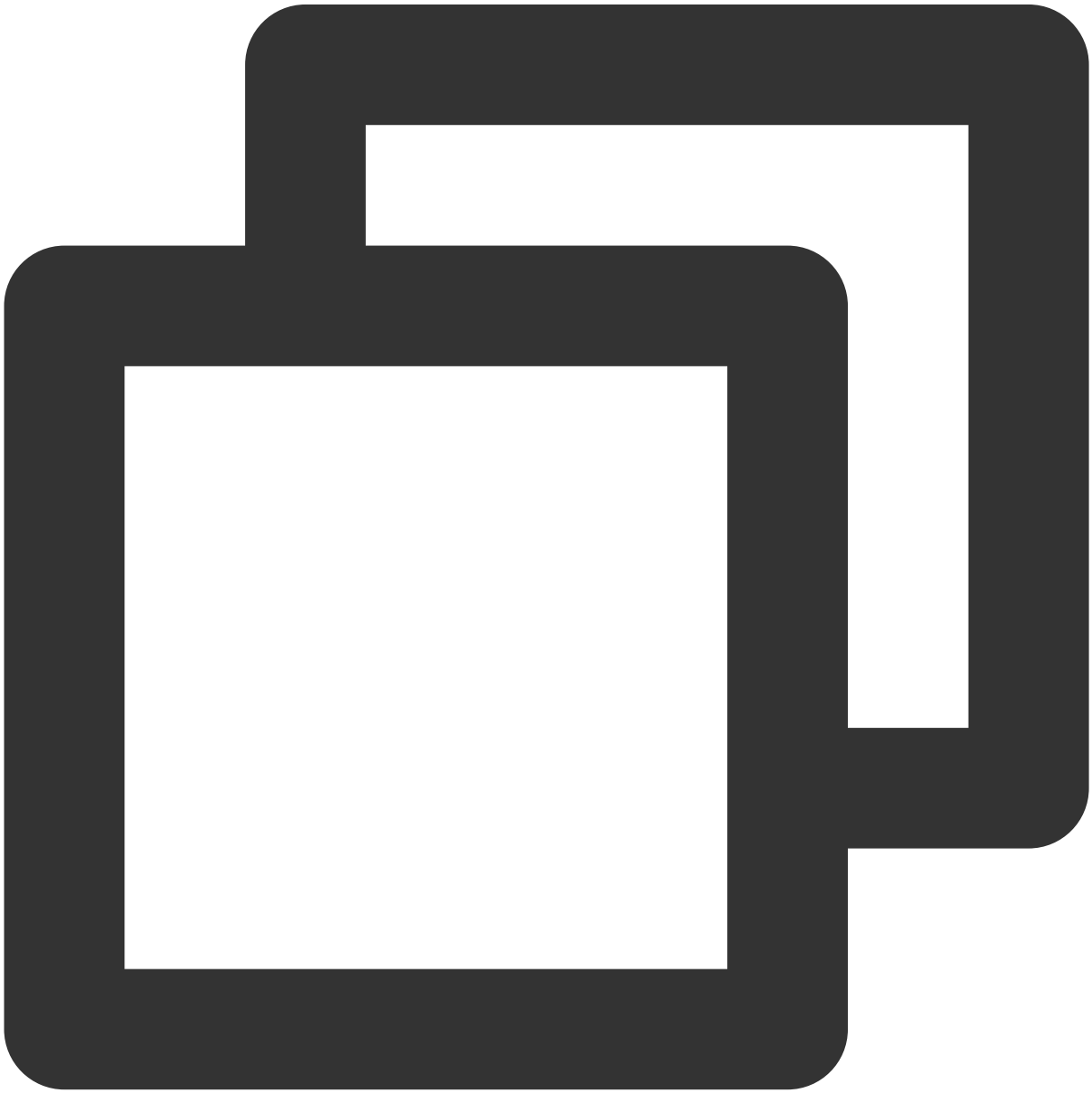
Error Code

Error code	Error Message	Description
0	ok	Normal
-1	already connect	Connected
10000	not init	Bluetooth Adapter Not Initialized
10001	not available	The current Bluetooth adapter is unavailable.
10002	no device	Specified Device Not Found
10003	connection fail	Connection failed.
10004	no service	Specified Service Not Found
10005	no characteristic	Specified characteristic not found
10006	no connection	The current connection has been disconnected.
10007	property not support	The current characteristic does not support this operation.
10008	system error	Exceptions Reported by All Other Systems
10009	system not support	Exclusive to Android system, and BLE is not supported on versions earlier than 4.3.

©2013-2022 Tencent Cloud. All rights reserved.

10012	operate time out	Connection timeout
10013	invalid_data	The connection deviceId is either null or incorrectly formatted.

Sample Code



```
wx.closeBLEConnection({
  deviceId,
  success (res) {
    console.log(res)
  }
```

©2013-2022 Tencent Cloud. All rights reserved.

```
})
```

©2013-2022 Tencent Cloud. All rights reserved.

Bluetooth - Low Energy Peripheral Device

Last updated：2024-07-16 18:02:33

onBLEPeripheralConnectionStateChanged

This API is used via wx.onBLEPeripheralConnectionStateChanged(function listener).

Feature Description: Monitors the current peripheral device for connection or disconnection events.
Parameter and Description: function listener, the listener for when the peripheral is connected or disconnected, with the following arguments Object res:

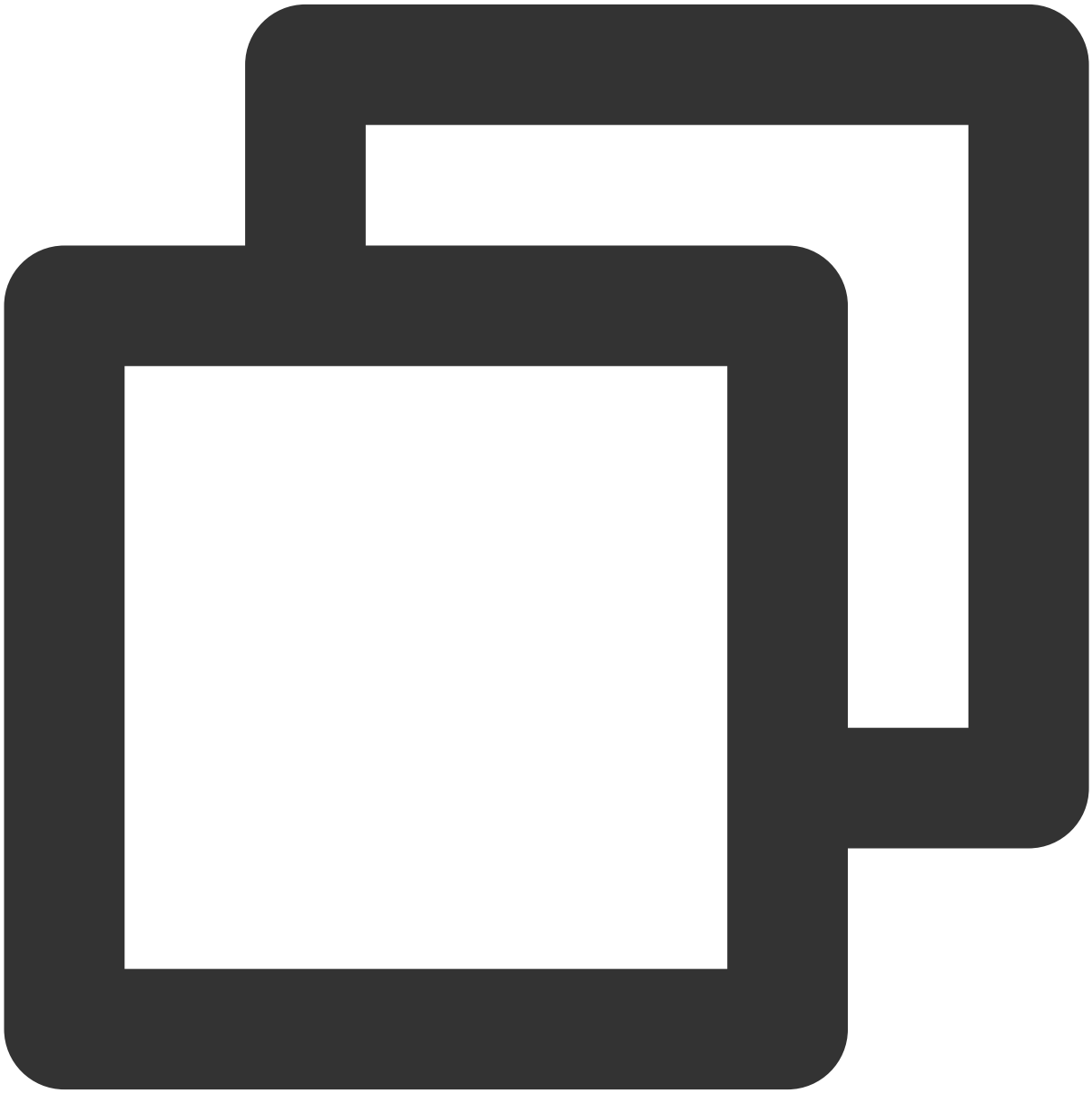
Attribute	Type	Description
deviceId	String	Device ID with changing connection status
serverId	String	Server's UUID
connected	Boolean	Current Connection Status

offBLEPeripheralConnectionStateChanged

This API is used via wx.offBLEPeripheralConnectionStateChanged(function listener).

Feature Description: Removes the listener function for connection or disconnection events of the current peripheral device.
Parameter and Description: function listener, the listener function passed in by onBLEPeripheralConnectionStateChanged. If this parameter is not passed in, all listener functions will be removed.
Sample Code

©2013-2022 Tencent Cloud. All rights reserved.



```
const listener = function (res) { console.log(res) }

wx.onBLEPeripheralConnectionStateChanged(listener)
wx.offBLEPeripheralConnectionStateChanged(listener) // The same function object as
```

createBLEPeripheralServer

©2013-2022 Tencent Cloud. All rights reserved.

This API is used via wx.createBLEPeripheralServer(Object object).

Feature Description: Establishes a local server as a Bluetooth Low Energy peripheral device and allows for the creation of multiple servers.

Parameter and Description: Object.

Attribute	Type	Default value	Required	Description
success	function	-	No	Callback Function of Successful Interface Call
fail	function	-	No	Callback Function of Failing Interface Call
complete	function	-	No	Callback function executed upon the completion of the interface invocation (both successful and unsuccessful invocations)

Parameters for object.success callback function: Object res.

Attribute	Type	Description
server	BLEPeripheralServer	Server of a peripheral device

BLEPeripheralServer

Note:

Server of a peripheral device.

.addService

This method is used via BLEPeripheralServer.addService(Object object).

Feature Description: Adds service.

Parameter and Description: Object.

Attribute	Type	Required	Description
service	Object	Yes	Object describing the service
success	function	No	Callback Function of Successful Interface Call

©2013-2022 Tencent Cloud. All rights reserved.

fail	function	No	Callback Function of Failing Interface Call
complete	function	No	Callback function executed upon the completion of the interface invocation (both successful and unsuccessful invocations)

Structure attribute of service

Structure attributes	Type	Required	Description
uuid	string	Yes	Bluetooth service UUID
characteristics	Array.<Object>	Yes	List of Characteristics

Structure attributes of characteristics

Structure attributes	Type	Required	Description
uuid	string	Yes	UUID of characteristic
properties	Object	No	Operations Supported by the Characteristic
permission	Object	No	Permission of characteristic
value	Array.<Object>	No	Binary data corresponding to the characteristic
descriptors	Array.<Object>	No	File descriptor data

Structure attributes of properties

Structure attributes	Type	Default value	Required	Description
write	boolean	false	No	Write
writeNoResponse	boolean	false	No	Write Without Response
read	boolean	false	No	Read
notify	boolean	false	No	Subscribe
indicate	boolean	false	No	Return packet

Structure attributes of permission

Structure attributes	Type	Default	Required	Description

©2013-2022 Tencent Cloud. All rights reserved.

		value		
readable	boolean	false	No	Readable
writeable	boolean	false	No	Writable
readEncryptionRequired	boolean	false	No	Encrypted Read Request
writeEncryptionRequired	boolean	false	No	Encrypted Write Request

Structure attributes of descriptors

Structure attributes	Type	Required	Description
uuid	string	Yes	UUID of Descriptors
permission	Object	No	Permissions of descriptors
value	ArrayBuffer	No	File descriptor data

Structure attributes of permission

Structure attributes	Type	Default value	Required	Description
write	boolean	false	No	Write
read	boolean	false	No	Read

.removeService

This method is used via BLEPeripheralServer.removeService(Object object).

Feature Description: Removes service.

Parameter and Description: Object.

Attribute	Type	Default value	Required	Description
serviceId	String	-	Yes	UUID of service
success	function	-	No	Callback Function of Successful Interface Call
fail	function	-	No	Callback Function of Failing Interface Call

©2013-2022 Tencent Cloud. All rights reserved.

complete	function	-	No	Callback function executed upon the completion of the interface invocation (both successful and unsuccessful invocations)

.startAdvertising

This method is used via BLEPeripheralServer.startAdvertising(Object Object).

Feature Description: Initiates broadcasting of locally created peripheral devices.

Parameter and Description: Object.

Attribute	Type	Default value	Required	Description
advertiseRequest	object	-	Yes	//Custom parameters of broadcasting
powerLevel	string	medium	No	Broadcasting power, whose valid values are: Low: Low power Medium: Moderate power High: High power
success	function	-	No	Callback Function of Successful Interface Call
fail	function	-	No	Callback Function of Failing Interface Call
complete	function	-	No	Callback function executed upon the completion of the interface invocation (both successful and unsuccessful invocations)

Structure attributes of advertiseRequest

Structure attributes	Type	Default value	Required	Description
connectable	boolean	true	No	Current device's connectivity status
deviceName	string	-	No	The deviceName field in the broadcasting, which is empty by default.
serviceUuids	Array.<string>	-	No	List of service UUIDs to be broadcast. Follow the notes when using 16/32 bit UUIDs.

©2013-2022 Tencent Cloud. All rights reserved.

manufactureData	Array.<Object>	-	No	Broadcasting manufacturer information. Customization is supported on Android only and not possible on iOS due to system limitations.
beacon	object	-	No	Parameters broadcast in the form of a beacon device.

Structure attributes of manufactureData

Structure attributes	Type	Required	Description
manufacturerId	String	Yes	Manufacturer ID, a hexadecimal beginning with 0x.
manufacturerSpecificData	ArrayBuffer	No	Manufacturer information

Structure attributes of "beacon"

Structure attributes	Type	Required	Description
uuid	number	Yes	UUID broadcast by a beacon device.
major	number	Yes	Primary ID of beacon device
minor	number	Yes	Secondary ID of beacon device
measurePower	number	Yes	Reference value for determining the RSSI size when the distance to the device is 1 meter.

.stopAdvertising

This method is used via BLEPeripheralServer.stopAdvertising().

Feature Description: Ceases broadcasting.

Parameter and Description: Object.

Attribute	Type	Default value	Required	Description
success	function	-	No	Callback Function of Successful Interface Call
fail	function	-	No	Callback Function of Failing Interface Call

©2013-2022 Tencent Cloud. All rights reserved.

complete	function	-	No	Callback function executed upon the completion of the interface invocation (both successful and unsuccessful invocations)

.writeCharacteristicValue

This method is used via BLEPeripheralServer.writeCharacteristicValue(Object Object).

Feature Description: Writes binary data values into the specified characteristic and notify the connected host. As the replica node's characteristic value has changed, this interface will handle whether to take the return packet or the subscription route.

Parameter and Description: Object.

Attribute	Type	Default value	Required	Description
serviceId	string	-	Yes	UUID corresponding to the Bluetooth characteristic service
characteristicId	string	-	Yes	Bluetooth characteristic UUID
value	ArrayBuffer	-	Yes	Binary data corresponding to the characteristic
needNotify	boolean	-	Yes	Whether it is necessary to notify the host that the value has been updated.
callbackId	number	-	No	Optional, used when handling return packets.
success	function	-	No	Callback Function of Successful Interface Call
fail	function	-	No	Callback Function of Failing Interface Call
complete	function	-	No	Callback function executed upon the completion of the interface invocation (both successful and unsuccessful invocations)

.onCharacteristicWriteRequest

This method is used via BLEPeripheralServer.onCharacteristicWriteRequest(function listener).

©2013-2022 Tencent Cloud. All rights reserved.

Feature Description: Monitors events where the connected device requests to write the characteristic value of the current peripheral device. Upon receiving this message, it is imperative to immediately call writeCharacteristicValue to write back the data. Otherwise, the host will not receive a response.
Parameter and Description: function listener, listener for a connected device requesting to write a feature value event for the current peripheral device, with the arguments Object res as follows.

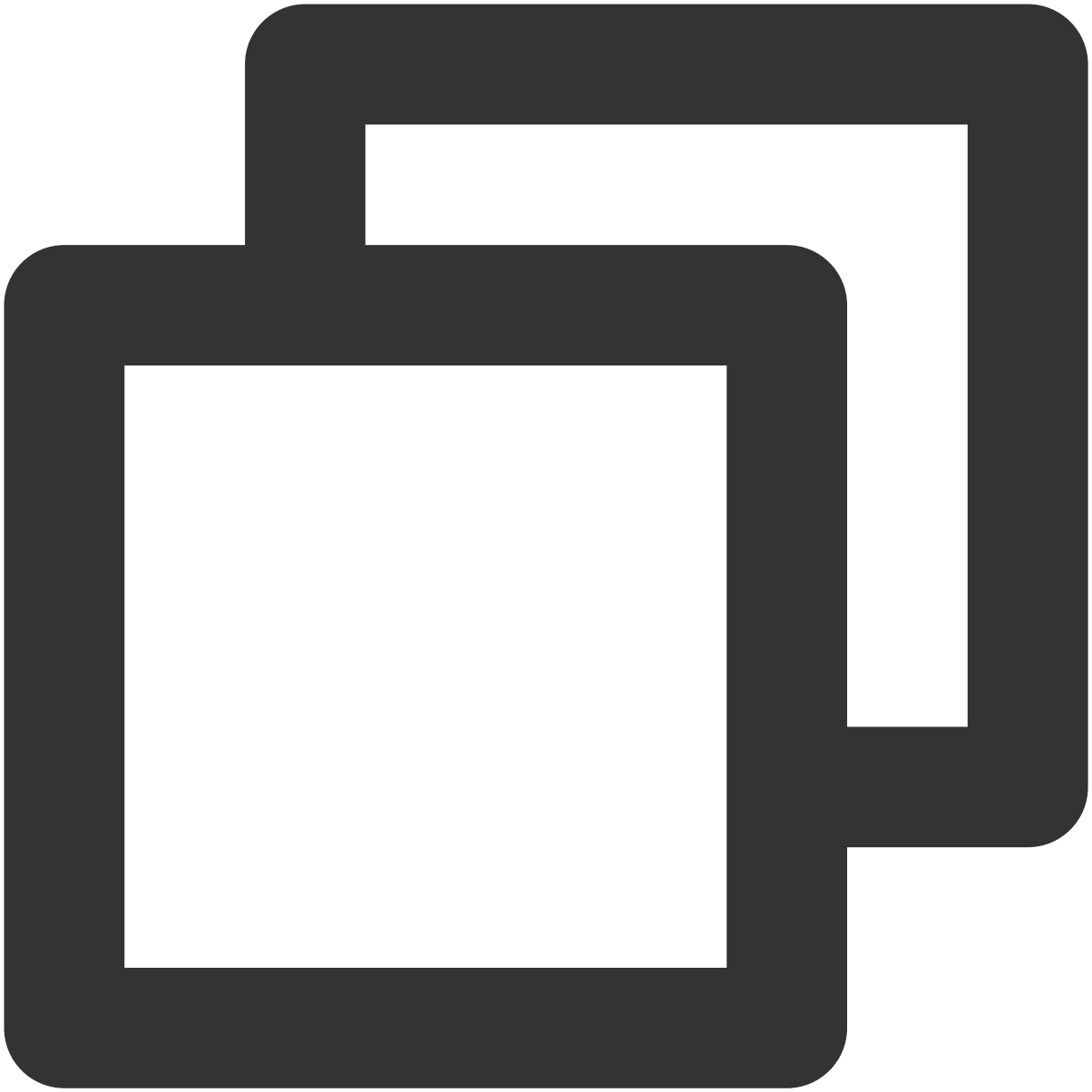
Attribute	Type	Description
serviceId	String	UUID corresponding to the Bluetooth characteristic service
characteristicId	String	Bluetooth characteristic UUID
callbackId	Number	Unique identifier, used when invoking writeCharacteristicValue.
value	ArrayBuffer	Binary data value requested for characteristic write.

.offCharacteristicWriteRequest

This method is used via BLEPeripheralServer.offCharacteristicWriteRequest(function listener).

Feature Description: Removes the listener function for events associated with characteristic value changes of the currently connected peripheral device.
Parameter and Description: function listener, the listener function passed in by onCharacteristicWriteRequest. If this parameter is not passed in, all listener functions will be removed.
Sample Code

©2013-2022 Tencent Cloud. All rights reserved.



```
const listener = function (res) { console.log(res) }

BLEPeripheralServer.onCharacteristicWriteRequest(listener)
BLEPeripheralServer.offCharacteristicWriteRequest(listener) // You need to pass the
```

.onCharacteristicReadRequest

This method is used via BLEPeripheralServer.onCharacteristicReadRequest(function listener).

©2013-2022 Tencent Cloud. All rights reserved.

Feature Description: Monitors events where the connected device requests to read the characteristic value of the current peripheral device. Upon receiving this message, it is imperative to immediately call writeCharacteristicValue to write back the data. Otherwise, the host will not receive a response.
Parameter and Description: function listener, listener for the event that a connected device requests to read the feature value of the current peripheral device, with the arguments Object res as follows.

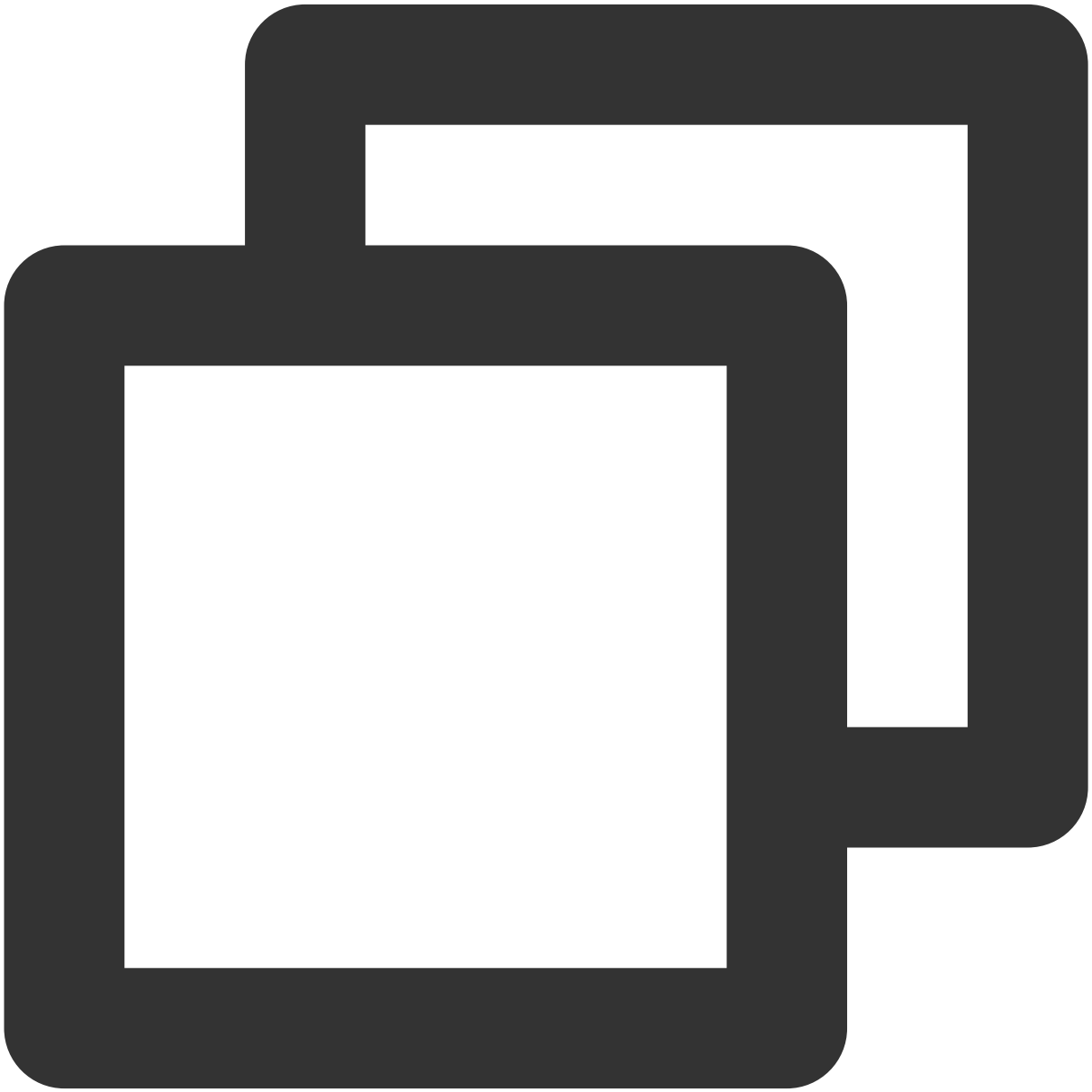
Attribute	Type	Description
serviceId	String	UUID corresponding to the Bluetooth characteristic service
characteristicId	String	Bluetooth characteristic UUID
callbackId	Number	Unique identifier, used when invoking writeCharacteristicValue.

.offCharacteristicReadRequest

This method is used via BLEPeripheralServer.offCharacteristicReadRequest(function listener).

Feature Description: Removes the listener function for events where the connected device requests to read the characteristic value of the current peripheral device.
Parameter and Description: function listener, the listener function passed in by onCharacteristicReadRequest. If this parameter is not passed in, all listener functions will be removed.
Sample Code

©2013-2022 Tencent Cloud. All rights reserved.



```
const listener = function (res) { console.log(res) }

BLEPeripheralServer.onCharacteristicReadRequest(listener)
BLEPeripheralServer.offCharacteristicReadRequest(listener) // The same function obj
```

.onCharacteristicSubscribed

This method is used via BLEPeripheralServer.onCharacteristicSubscribed(function listener).

©2013-2022 Tencent Cloud. All rights reserved.

Feature Description: Monitors characteristic subscription events, supported exclusively on iOS.
Parameter and Description: function listener, a listener function for feature subscription events, with the following arguments Object res:

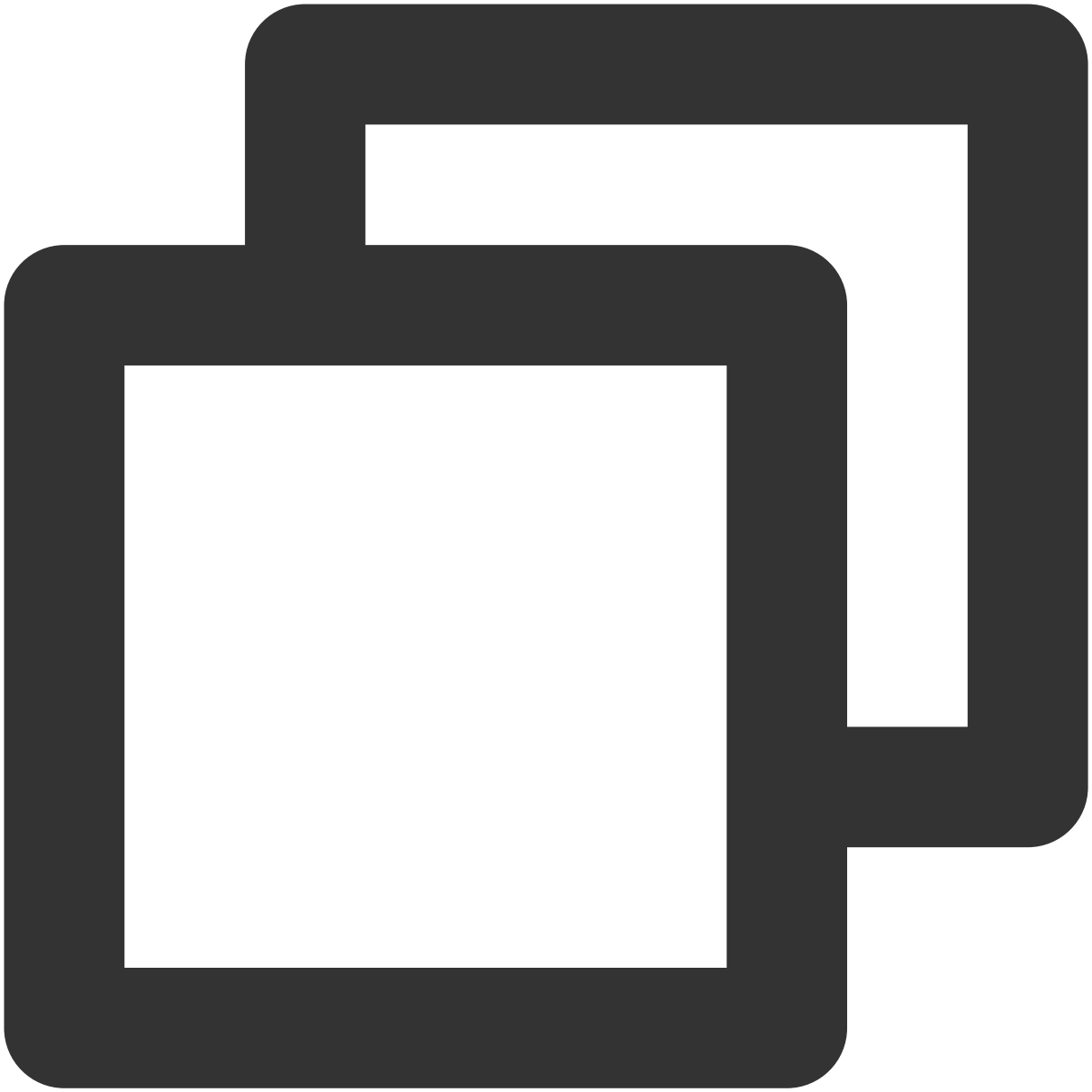
Attribute	Type	Description
serviceId	String	UUID corresponding to the Bluetooth characteristic service
characteristicId	String	Bluetooth characteristic UUID

.offCharacteristicSubscribed

This method is used via BLEPeripheralServer.offCharacteristicSubscribed(function listener).

Feature Description: Removes the listener function for characteristic subscription events.
Parameter and Description: function listener, the listener function passed in by onCharacteristicSubscribed. If this parameter is not passed in, all listener functions will be removed.
Sample Code

©2013-2022 Tencent Cloud. All rights reserved.



```
const listener = function (res) { console.log(res) }

BLEPeripheralServer.onCharacteristicSubscribed(listener)
BLEPeripheralServer.offCharacteristicSubscribed(listener) // The same function obje
```

.onCharacteristicUnsubscribed

This method is used via BLEPeripheralServer.onCharacteristicUnsubscribed(function listener).

©2013-2022 Tencent Cloud. All rights reserved.

Feature Description: Monitors characteristic unsubscription events, supported exclusively on iOS.

Parameter and Description: function listener, a listener function that cancels a feature subscription event, with the following arguments Object res:

Attribute	Type	Description
serviceId	String	UUID corresponding to the Bluetooth characteristic service
characteristicId	String	Bluetooth characteristic UUID

.offCharacteristicUnsubscribed

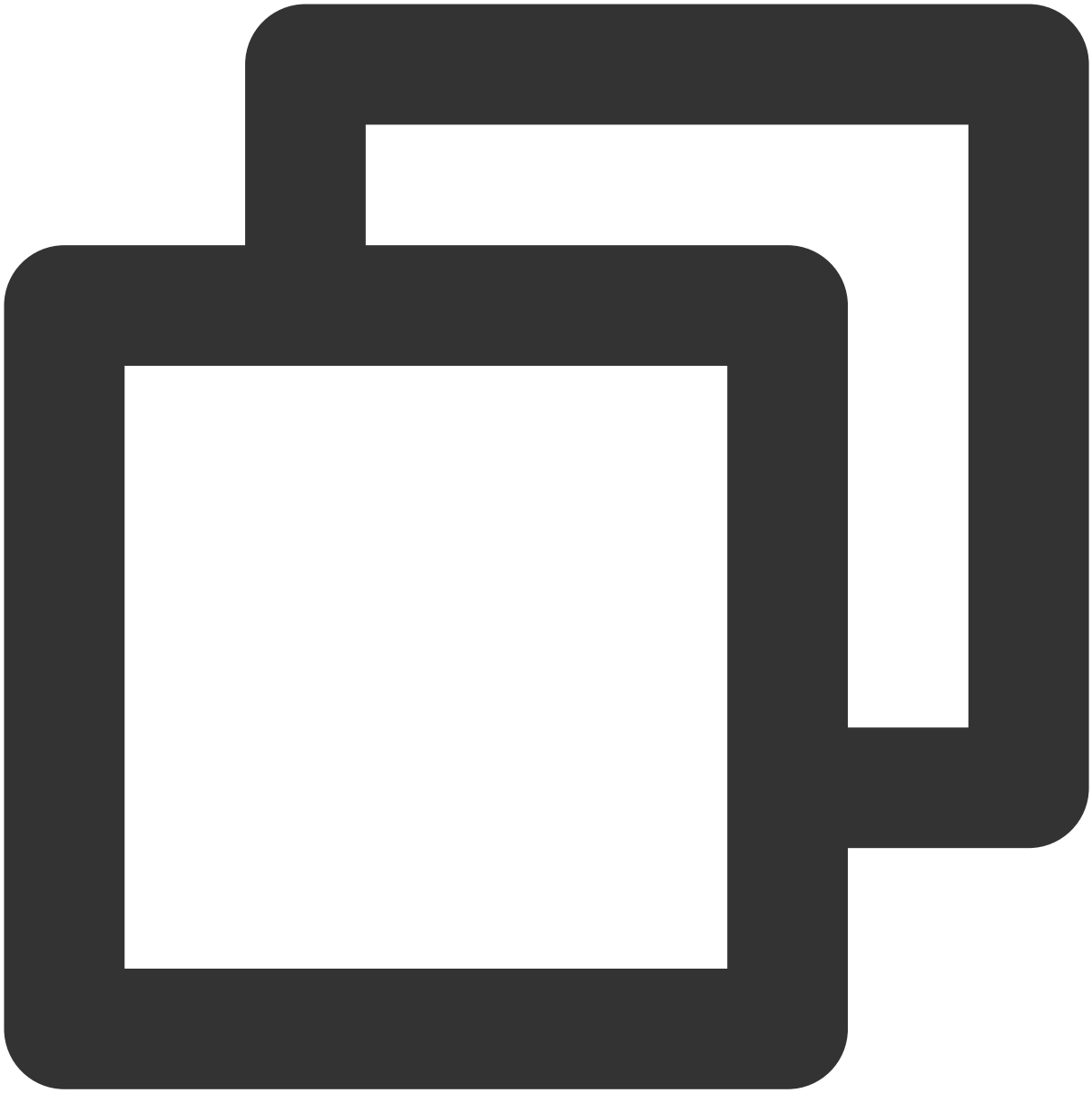
This method is used via BLEPeripheralServer.offCharacteristicUnsubscribed(function listener).

Feature Description: Removes the listener function for characteristic unsubscription events.

Parameter and Description: function listener, the listener function passed in by onCharacteristicUnsubscribed. If this parameter is not passed in, all listener functions will be removed.

Sample Code

©2013-2022 Tencent Cloud. All rights reserved.



```
const listener = function (res) { console.log(res) }

BLEPeripheralServer.onCharacteristicUnsubscribed(listener)
BLEPeripheralServer.offCharacteristicUnsubscribed(listener) // The same function ob
```

©2013-2022 Tencent Cloud. All rights reserved.

Bluetooth - Beacon

Last updated：2024-07-16 18:02:59

stopBeaconDiscovery

The method of utilizing this API is wx.stopBeaconDiscovery(Object object).

Function Description: Cease the search for nearby Beacon devices.

Parameters and Description: Object object.

Attributes	Types	Default value	Required	Note
success	function	-	Not required	Callback function for successful interface invocation
fail	function	-	Not required	Callback function for unsuccessful interface invocation
complete	function	-	Not required	Callback function executed upon the completion of the interface invocation (both successful and unsuccessful invocations)

Error Code

Error Codes	Error Message	Note
0	ok	Normal
11000	unsupport	System or device not supported
11001	bluetooth service unavailable	Bluetooth service is unavailable
11002	location service unavailable	Location services are unavailable
11003	already start	Search has commenced
11004	not startBeaconDiscovery	Search has not yet commenced
11005	system error	System error
11006	invalid data	Parameters are incorrect

©2013-2022 Tencent Cloud. All rights reserved.

startBeaconDiscovery

The method for using this API is wx.startBeaconDiscovery(Object object)

Function Description: Initiates the search for nearby Beacon devices.

Parameters and Description: Object object.

Attributes	Types	Default value	Required	Note
uuids	Array.<string>	-	Supported	List of UUIDs broadcasted by Beacon devices
ignoreBluetoothAvailable	boolean	false	Not required	Verifies the Bluetooth switch, effective only on iOS. From iOS 11 onwards, even if Bluetooth is turned off in the control panel, the Beacon service can still be used.
success	function	-	Not required	Callback function for successful interface invocation
fail	function	-	Not required	Callback function for unsuccessful interface invocation
complete	function	-	Not required	Callback function executed upon the completion of the interface invocation (both successful and unsuccessful invocations)

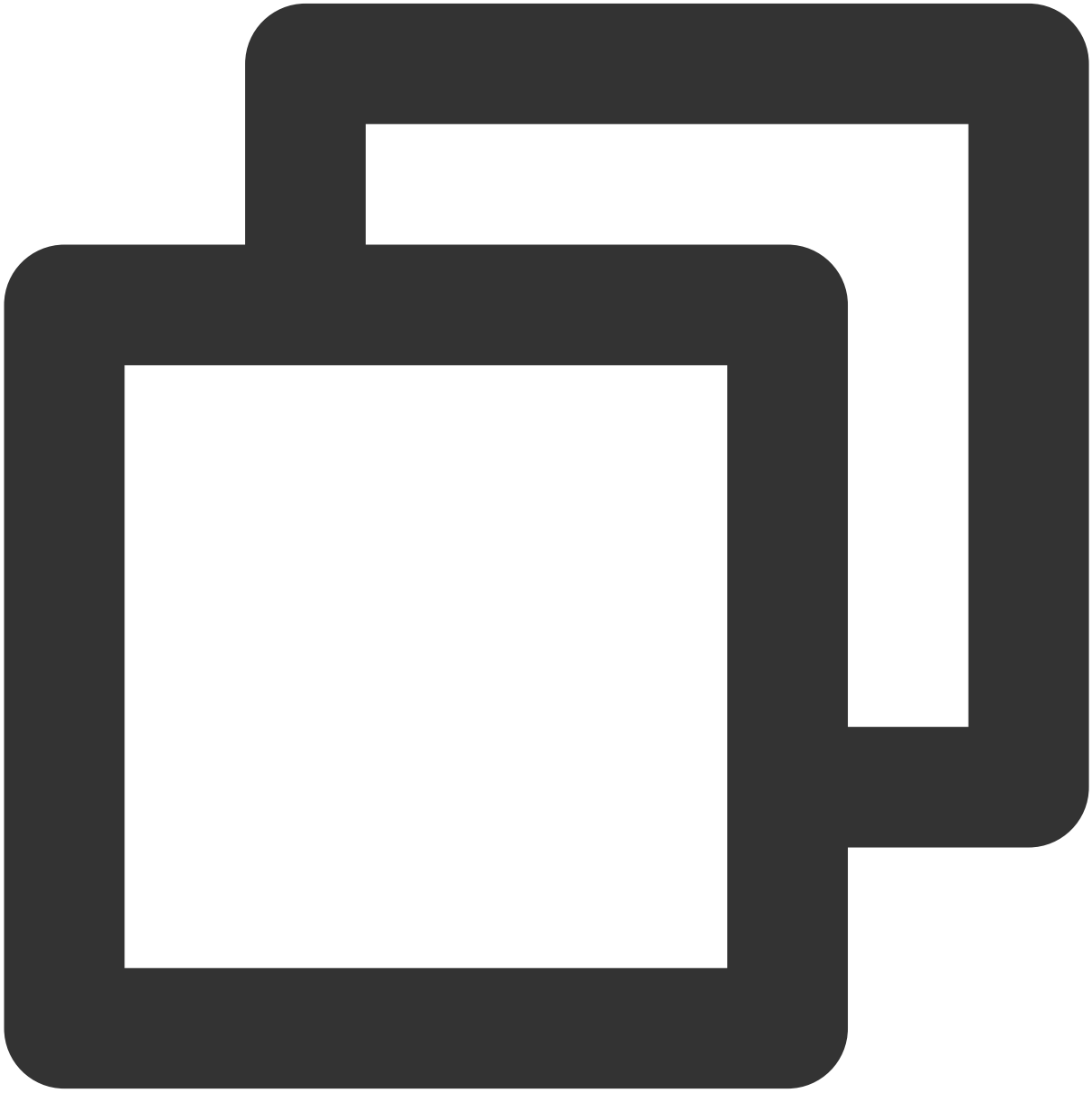
Error Code

Error Codes	Error Message	Note
0	ok	Normal
11000	unsupport	System or device not supported
11001	bluetooth service unavailable	Bluetooth service is unavailable
11002	location service unavailable	Location services are unavailable
11003	already start	Search has commenced
11004	not startBeaconDiscovery	Search has not yet commenced

©2013-2022 Tencent Cloud. All rights reserved.

11005	system error	System error
11006	invalid data	Parameters are incorrect

Example Code



```
wx.startBeaconDiscovery({
  success(res) { }
})
```

©2013-2022 Tencent Cloud. All rights reserved.

onBeaconUpdate

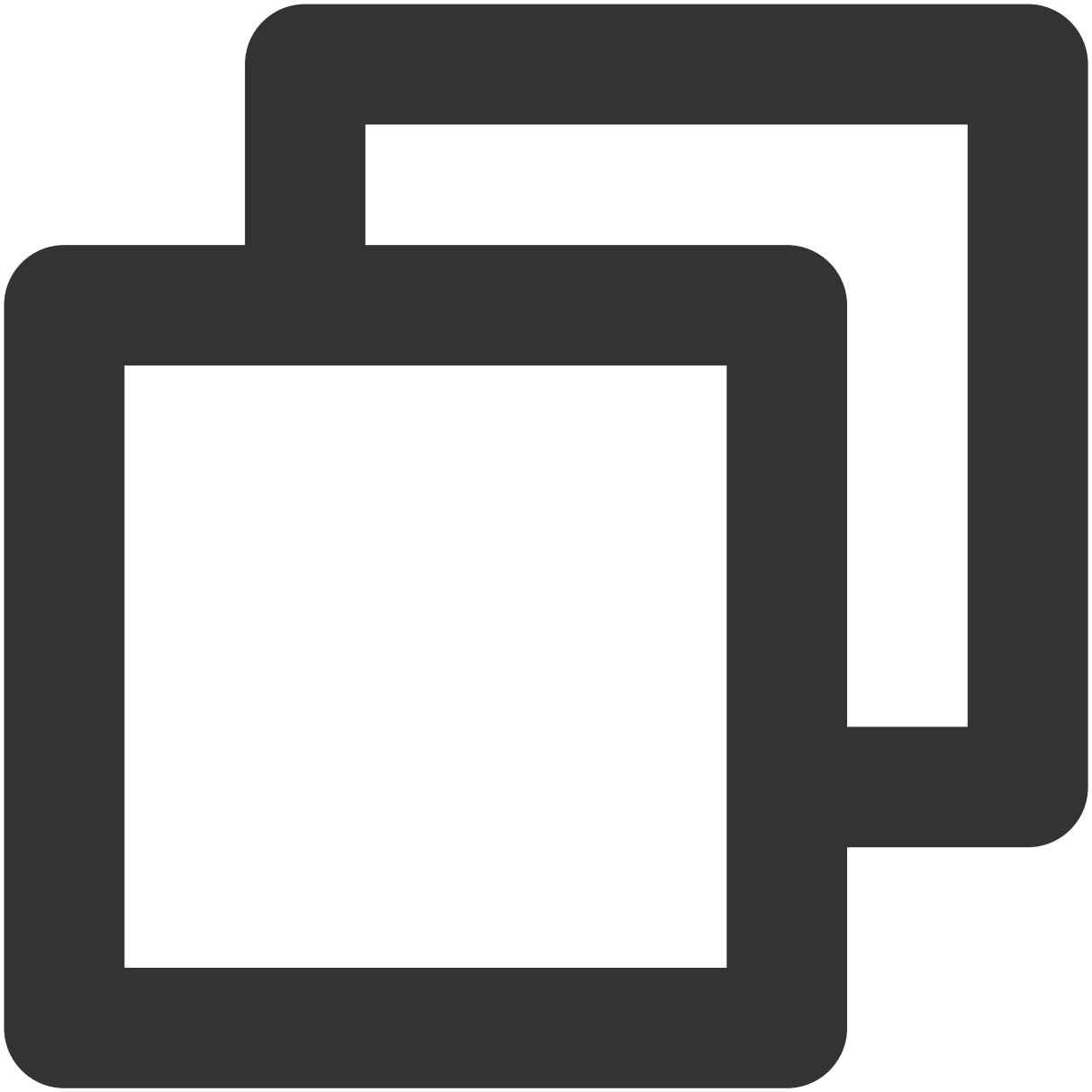
The method for using this API is wx.onBeaconUpdate(function listener)

Function Description: Monitors the Beacon device update event, only one listener can be registered.
Parameters and Description: function listener, the listener function for the Beacon device update event, with the following arguments Object res:

Attributes	Types	Note
beacons	Array.<BeaconInfo>	The list of all Beacon devices currently detected.

Example Code

©2013-2022 Tencent Cloud. All rights reserved.



```
wx.onBeaconUpdate(res => {
   console.log(res.beacons)
})
```

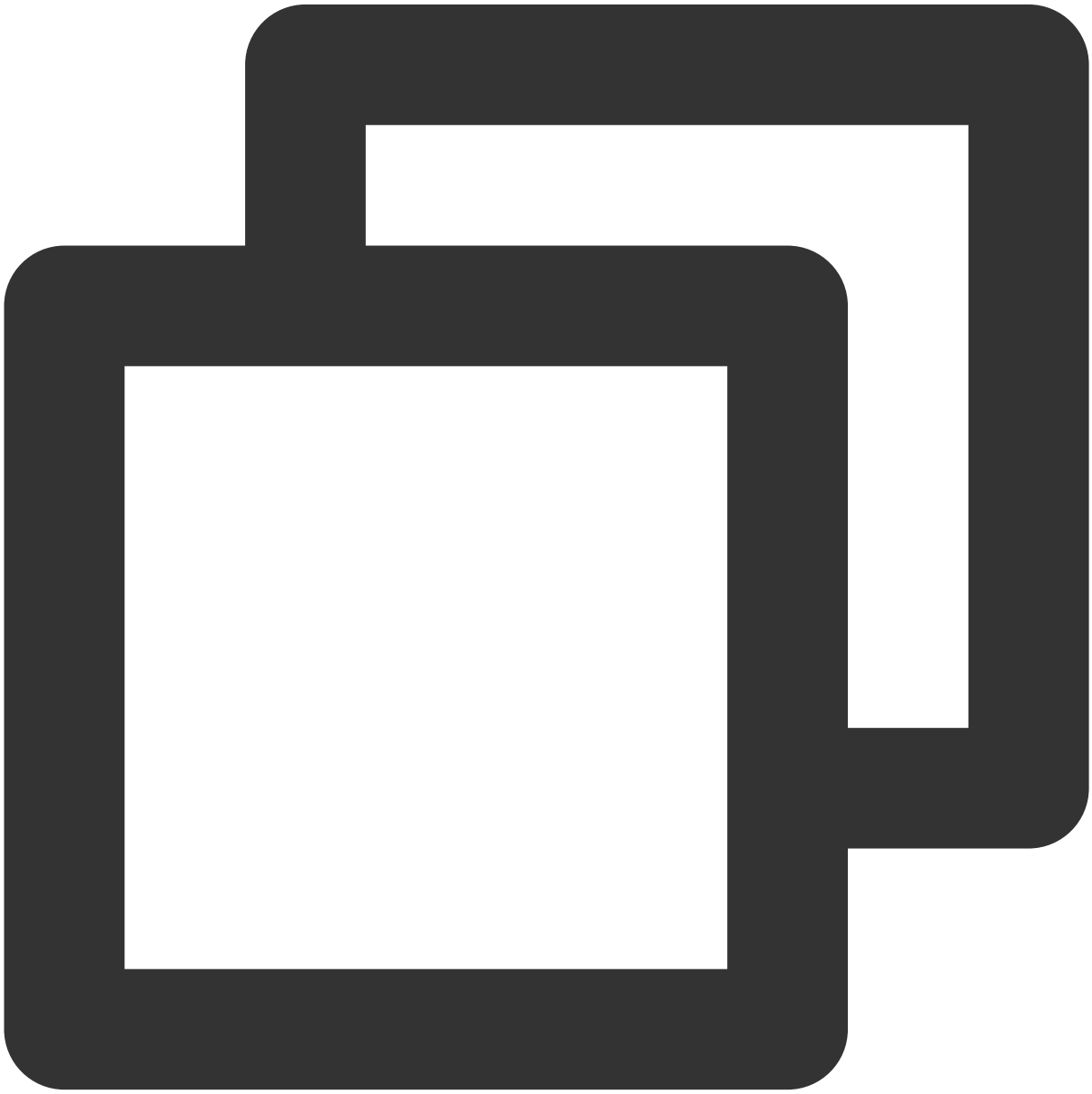
offBeaconUpdate

The method for using this API is wx.offBeaconUpdate()

©2013-2022 Tencent Cloud. All rights reserved.

Function Description: Removes all listener functions for the Beacon device update event.

Sample Code



```
wx.offBeaconUpdate()
```

onBeaconServiceChange

©2013-2022 Tencent Cloud. All rights reserved.

The method for using this API is wx.onBeaconServiceChange(function listener)

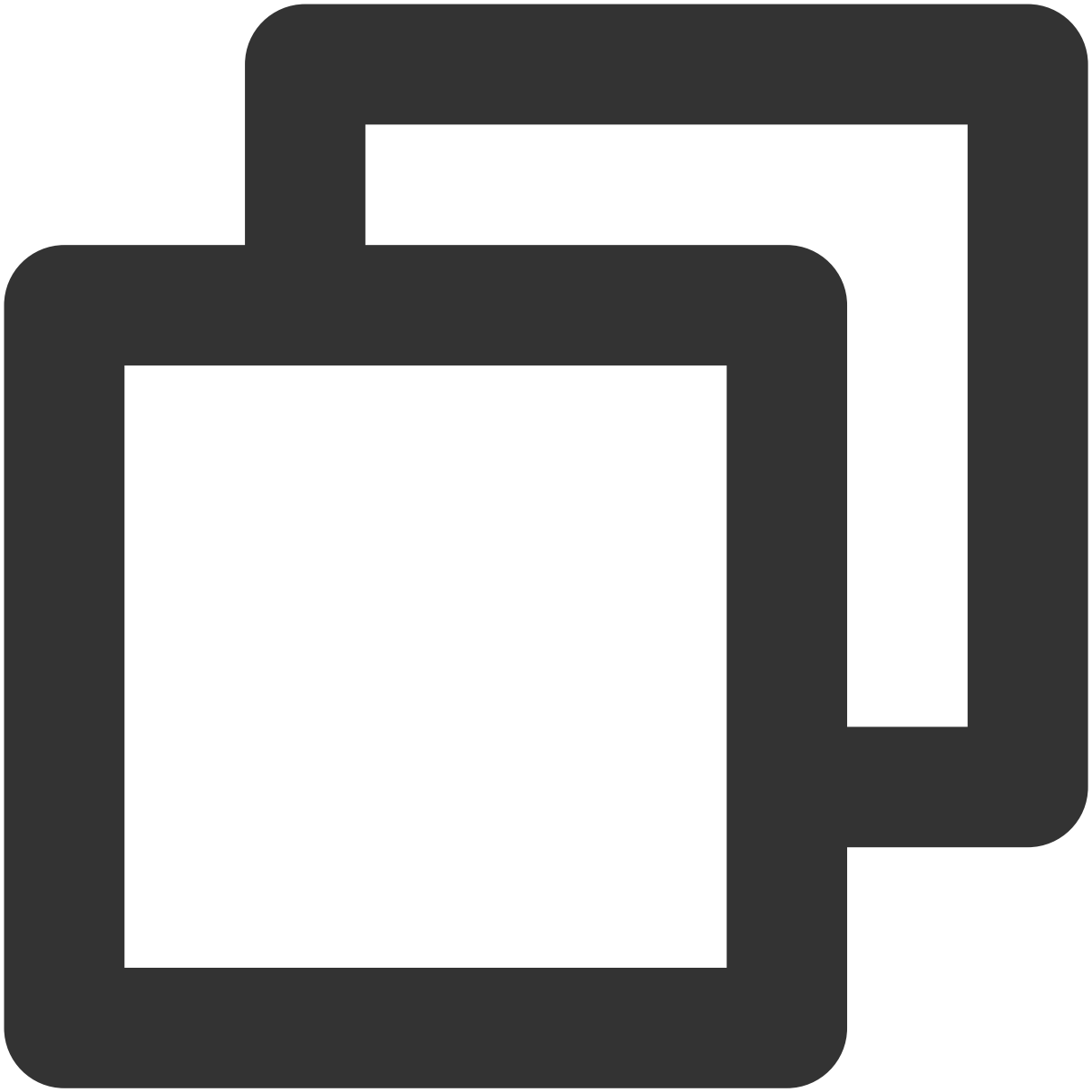
Function Description: Monitors the Beacon service status change event, only one listener can be registered.

Parameters and Description: function listener, listener for Beacon service state change events, with the following arguments Object res:

Attributes	Types	Note
available	boolean	Is the service currently available?
discovering	boolean	Is it currently in search mode?

Example Code

©2013-2022 Tencent Cloud. All rights reserved.



```
wx.onBeaconServiceChange(res => {
   console.log(res.available, res.discovering)
})
```

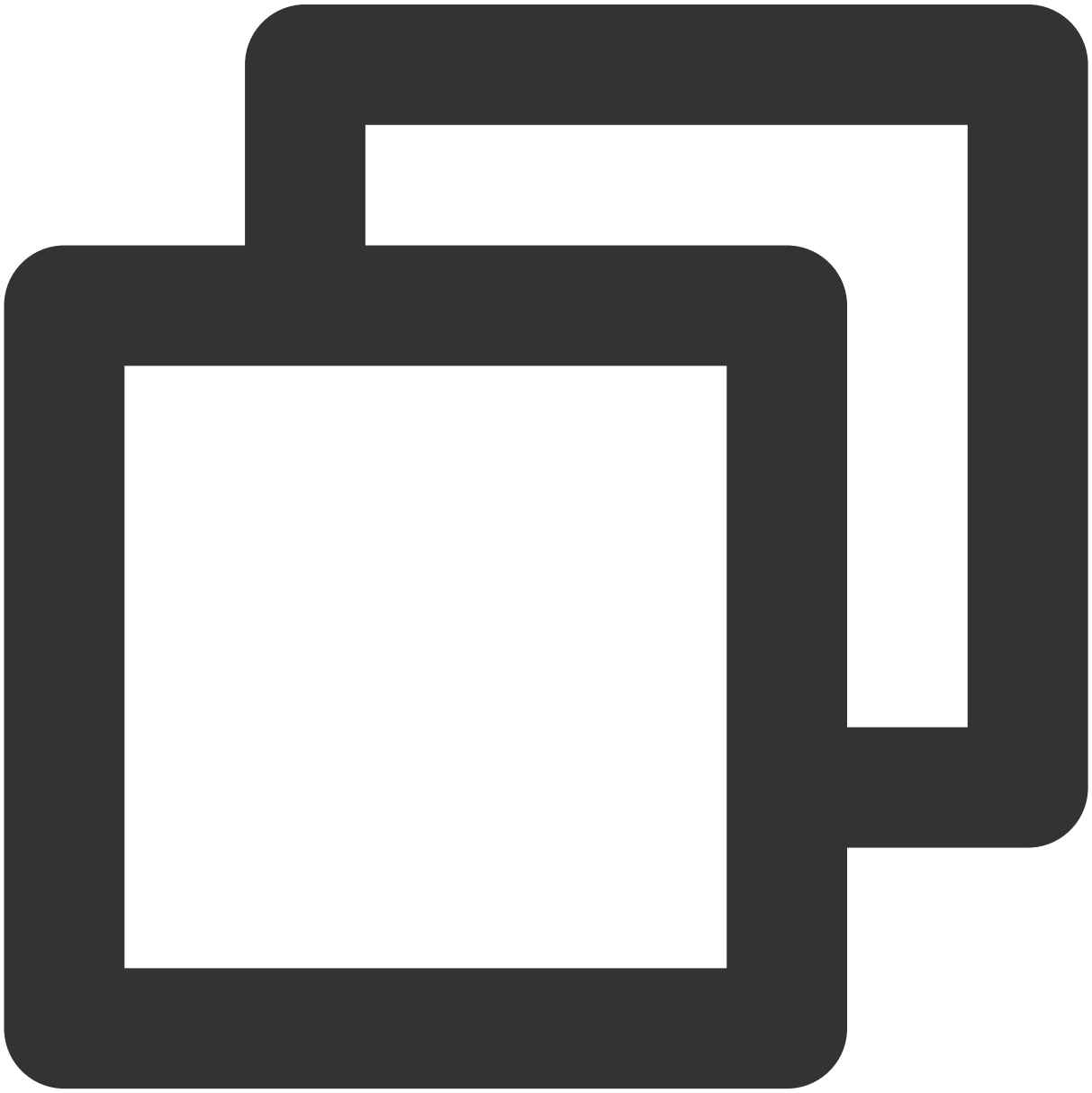
offBeaconServiceChange

The method for using this API is wx.offBeaconServiceChange()

©2013-2022 Tencent Cloud. All rights reserved.

Function Description: Removes all listener functions for the Beacon service status change event.

Sample Code



```
wx.offBeaconServiceChange()
```

getBeacons

©2013-2022 Tencent Cloud. All rights reserved.

The method for using this API is wx.getBeacons(Object object)

Function Description: Retrieves all detected Beacon devices.

Parameters and Description: Object object.

Attributes	Types	Default value	Required	Note
success	function	-	Not required	Callback function for successful interface invocation
fail	function	-	Not required	Callback function for unsuccessful interface invocation
complete	function	-	Not required	Callback function executed upon the completion of the interface invocation (both successful and unsuccessful invocations)

object.success Callback Function Parameters: Object res.

Attributes	Types	Note
beacons	Array.<BeaconInfo>	Beacon Device List

Error Code

Error Codes	Error Message	Note
0	ok	Normal
11000	unsupport	System or device not supported
11001	bluetooth service unavailable	Bluetooth service is unavailable
11002	location service unavailable	Location services are unavailable
11003	already start	Search has commenced
11004	not startBeaconDiscovery	Search has not yet commenced
11005	system error	System error
11006	invalid data	Parameters are incorrect

©2013-2022 Tencent Cloud. All rights reserved.

BeaconInfo

Function Description: Beacon Device.

Properties and Descriptions

Attribute	Types	Note
uuid	string	UUID broadcasted by the beacon device.
major	number	Primary ID of the beacon device.
minor	number	Secondary ID of the beacon device.
proximity	number	Enumeration value representing the device's distance (iOS only)
accuracy	number	Distance to the beacon device, measured in meters. On iOS, when proximity is 0, accuracy is -1.
rssi	number	Indicates the strength of the device's signal, measured in dBm.

Valid values for proximity

Value	Note
0	The signal is too weak to calculate the distance, or the device is not an iOS device.
1	Extremely close
2	Relatively close
3	Distant

©2013-2022 Tencent Cloud. All rights reserved.

Bluetooth - General

Last updated：2024-07-16 18:03:27

makeBluetoothPairstopBluetoothDevicesDiscovery

This API is used via wx.stopBluetoothDevicesDiscovery(Object object).

Feature Description: Stops searching for nearby Bluetooth peripheral devices. It is advisable to invoke this interface to cease the Bluetooth search when the required Bluetooth device has been located and there is no need for further search.

Parameter and Description: Object.

Attribute	Type	Default value	Required	Description
success	function	-	No	Callback Function of Successful Interface Call
fail	function	-	No	Callback Function of Failing Interface Call
complete	function	-	No	Callback function executed upon the completion of the interface invocation (both successful and unsuccessful invocations)

Error Code

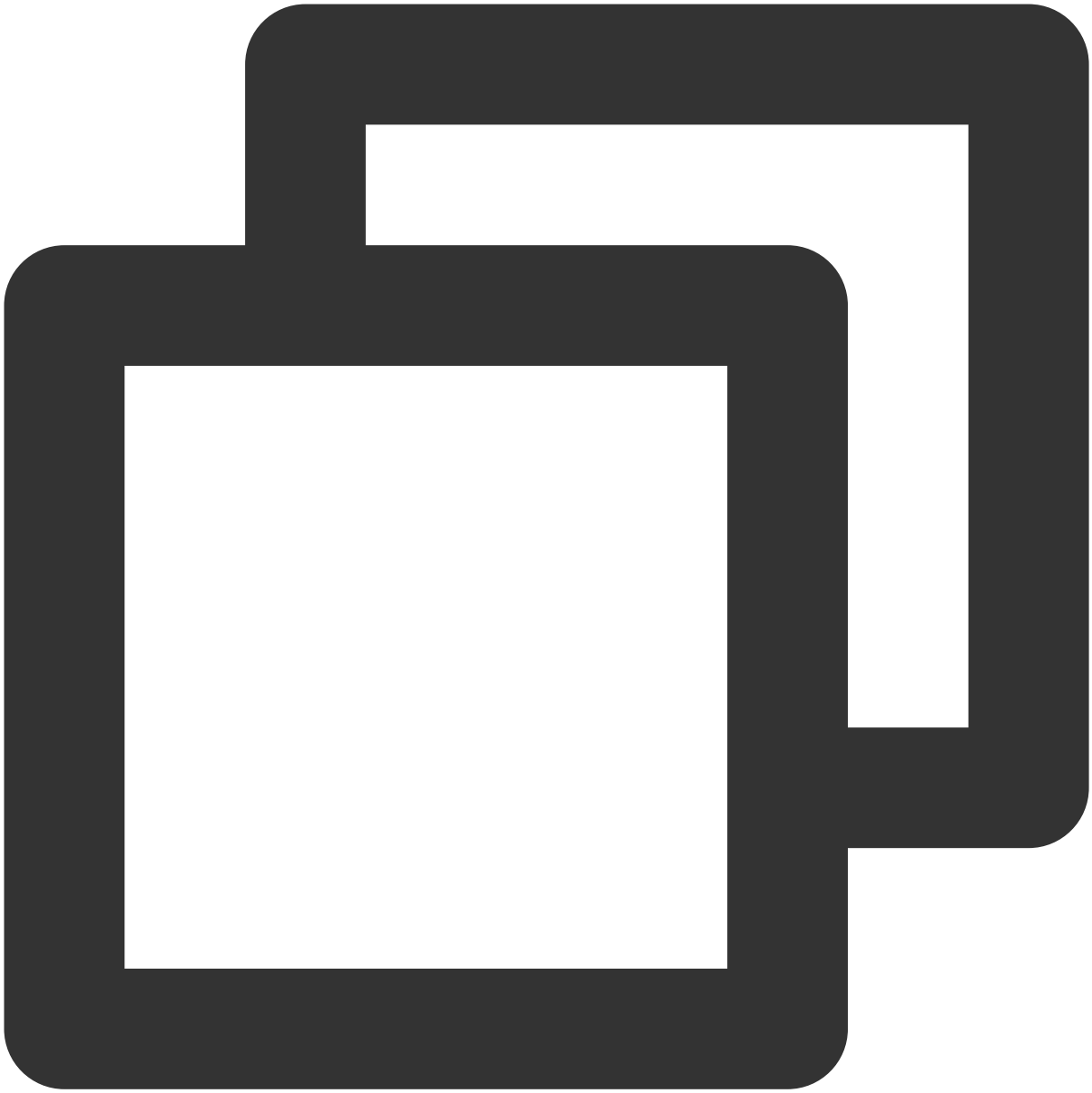
Error code	Error Message	Description
0	ok	Normal
-1	already connect	Connected
10000	not init	Bluetooth Adapter Not Initialized
10001	not available	The current Bluetooth adapter is unavailable.
10002	no device	Specified Device Not Found
10003	connection fail	Connection failed.
10004	no service	Specified Service Not Found

©2013-2022 Tencent Cloud. All rights reserved.

10005	no characteristic	Specified characteristic not found
10006	no connection	The current connection has been disconnected.
10007	property not support	The current characteristic does not support this operation.
10008	system error	Exceptions Reported by All Other Systems
10009	system not support	Exclusive to Android system, and BLE is not supported on versions earlier than 4.3.
10012	operate time out	Connection timeout
10013	invalid_data	The connection deviceId is either null or incorrectly formatted.

Sample Code

©2013-2022 Tencent Cloud. All rights reserved.



```
wx.stopBluetoothDevicesDiscovery({
  success (res) {
    console.log(res)
  }
})
```

startBluetoothDevicesDiscovery

©2013-2022 Tencent Cloud. All rights reserved.

This API is used via wx.startBluetoothDevicesDiscovery(Object object).

Note:

Given that Bluetooth functionality can indirectly facilitate location tracking, Android 6.0 and later versions cannot perform device searches without location permissions or if the location switch is not activated. Under these circumstances, for versions prior to Android 8.0.16, the interface call will succeed but the device cannot be scanned; for Android 8.0.16 and later versions, an error prompt will be returned.

Feature Description: Starts the search for nearby Bluetooth peripheral devices. This operation is resource-intensive, hence it is advisable to promptly call wx.stopBluetoothDevicesDiscovery to stop the search once the required device has been found.

Parameter and Description: Object.

Attribute	Type	Default value	Required	Description
services	Array.<string>	-	No	The list of UUIDs for the primary services of the Bluetooth devices to be searched (supporting 16/32/128 bit UUIDs). Some Bluetooth devices will broadcast their primary service's UUID. If this parameter is set, only Bluetooth devices broadcasting a primary service with the corresponding UUID will be searched. It is recommended to use this parameter to filter out other surrounding Bluetooth devices that do not need to be processed.
allowDuplicatesKey	boolean	false	No	If or not allow to report the same device more than once. If repeat reporting is allowed, the wx.onBluetoothDeviceFound method will report the same device multiple times, but with different RSSI values.
interval	number	0	No	Interval for reporting devices, in ms. 0 means that new devices are found and reported immediately, other values are reported according to the incoming interval.
powerLevel	string	medium	No	The scanning mode. The higher the mode, the faster the scan and the greater the power consumption. Only Android host clients in V 7.0.12 and later versions support this, with valid values being: Low: Low power

©2013-2022 Tencent Cloud. All rights reserved.

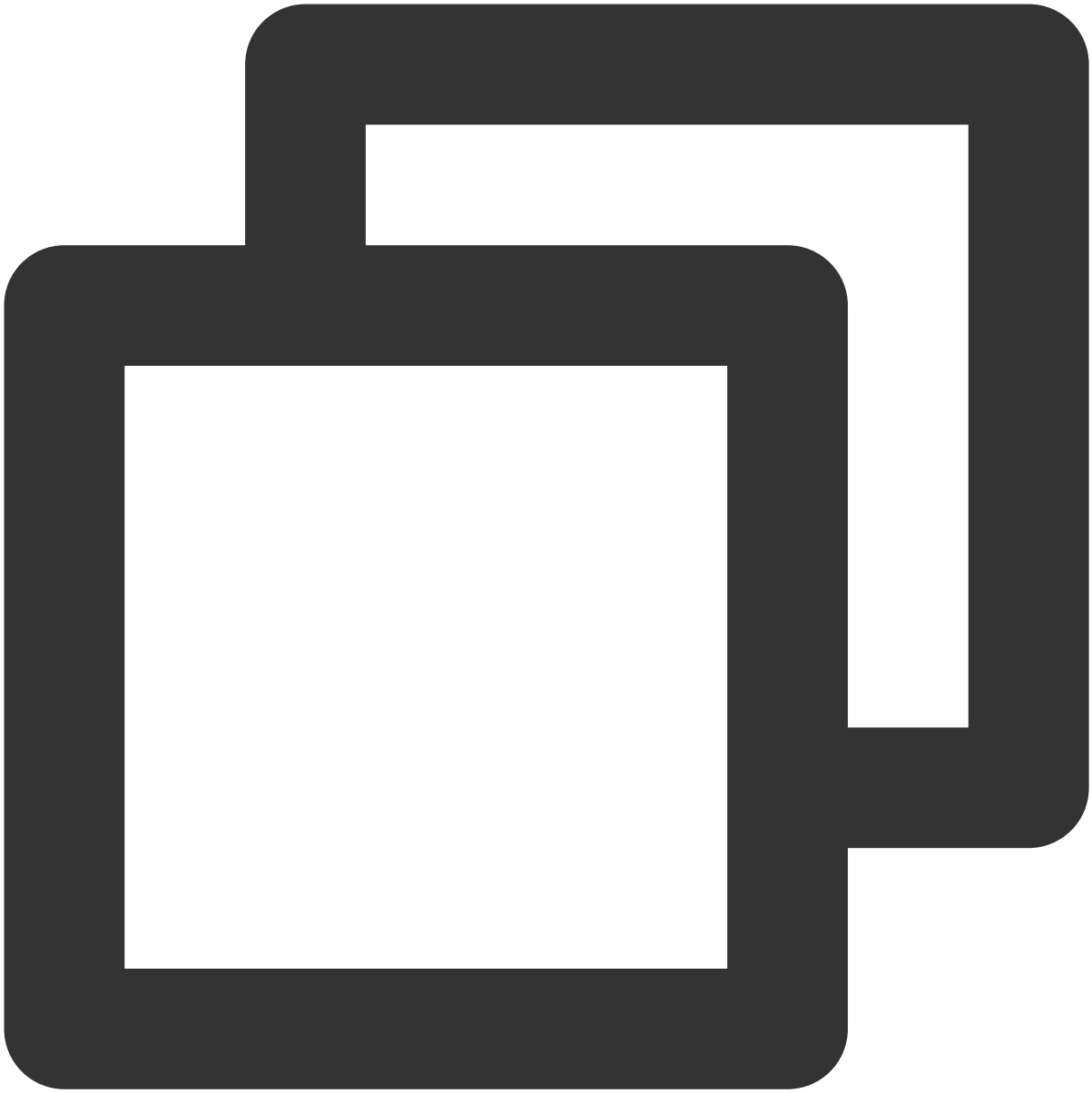
				Medium: Moderate power High: High power
success	function	-	No	Callback Function of Successful Interface Call
fail	function	-	No	Callback Function of Failing Interface Call
complete	function	-	No	Callback function executed upon the completion of the interface invocation (both successful and unsuccessful invocations)

Error Code

Error code	Error Message	Description
0	ok	Normal
-1	already connect	Connected
10000	not init	Bluetooth Adapter Not Initialized
10001	not available	The current Bluetooth adapter is unavailable.
10002	no device	Specified Device Not Found
10003	connection fail	Connection failed.
10004	no service	Specified Service Not Found
10005	no characteristic	Specified characteristic not found
10006	no connection	The current connection has been disconnected.
10007	property not support	The current characteristic does not support this operation.
10008	system error	Exceptions Reported by All Other Systems
10009	system not support	Exclusive to Android system, and BLE is not supported on versions earlier than 4.3.
10012	operate time out	Connection timeout
10013	invalid_data	The connection deviceId is either null or incorrectly formatted.

Sample Code

©2013-2022 Tencent Cloud. All rights reserved.



```
// Taking the Bluetooth smart light of the host client's hardware platform as an ex
wx.startBluetoothDevicesDiscovery({
  services: ['FEE7'],
  success (res) {
    console.log(res)
  }
})
```

©2013-2022 Tencent Cloud. All rights reserved.

openBluetoothAdapter

This API is used via wx.openBluetoothAdapter(Object object).

Note:

Other Bluetooth-related APIs must be used after calling wx.openBluetoothAdapter. Otherwise, the API will return an error prompt (errCode=10000).

In situations where the user's Bluetooth switch is not turned on or the mobile phone does not support Bluetooth functionality, calling wx.openBluetoothAdapter will return an error code (errCode=10001), indicating that the mobile phone's Bluetooth functionality is unavailable.

Feature Description: Initializes the Bluetooth module. On iOS, when turning on the host/replica (peripheral) mode, it needs to be called once respectively, and the corresponding `mode` should be specified.

Parameter and Description: Object.

Attribute	Type	Valid Values and Descriptions	Default value	Required	Description
mode	string	central: Host mode peripheral: Replica (peripheral device) mode	central	No	Bluetooth mode, which can function as a host/replica device, only required for iOS.
success	function	-	-	No	Callback Function of Successful Interface Call
fail	function	-	-	No	Callback Function of Failing Interface Call
complete	function	-	-	No	Callback function executed upon the completion of the interface invocation (both successful and unsuccessful invocations)

Error Return Value:

Error code	Error Message	Description
0	ok	Normal
-1	already connect	Connected
10000	not init	Bluetooth Adapter Not Initialized

©2013-2022 Tencent Cloud. All rights reserved.

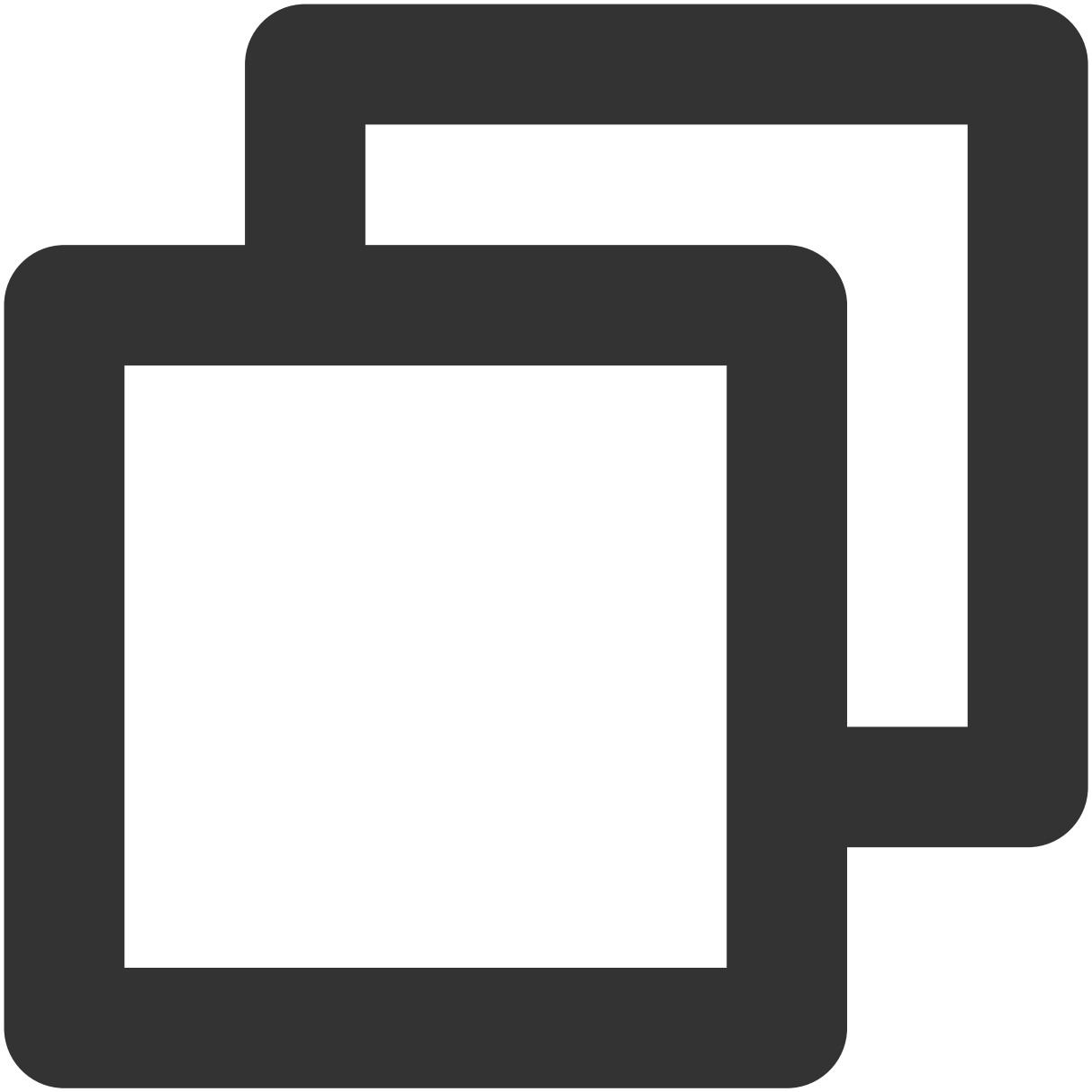
10001	not available	The current Bluetooth adapter is unavailable.
10002	no device	Specified Device Not Found
10003	connection fail	Connection failed.
10004	no service	Specified Service Not Found
10005	no characteristic	Specified characteristic not found
10006	no connection	The current connection has been disconnected.
10007	property not support	The current characteristic does not support this operation.
10008	system error	Exceptions Reported by All Other Systems
10009	system not support	Exclusive to Android system, and BLE is not supported on versions earlier than 4.3.
10012	operate time out	Connection timeout
10013	invalid_data	The connection deviceId is either null or incorrectly formatted.

state parameter returned by the object.fail callback function (iOS only)

Status code	Description
0	Unknown
1	Resetting
2	Not supported
3	Not authorized
4	Not enabled

Sample Code

©2013-2022 Tencent Cloud. All rights reserved.



```
wx.openBluetoothAdapter({
  success (res) {
    console.log(res)
  }
})
```

getConnectedBluetoothDevices

©2013-2022 Tencent Cloud. All rights reserved.

This API is used via wx.getConnectedBluetoothDevices(Object object).

Feature Description: Gets connected Bluetooth devices based on the primary service UUID.

Parameter and Description: Object.

Attribute	Type	Default value	Required	Description
services	Array.	-	Yes	List of primary service UUIDs for the Bluetooth device (supporting 16/32/128 bit UUIDs)
success	function	-	No	Callback Function of Successful Interface Call
fail	function	-	No	Callback Function of Failing Interface Call
complete	function	-	No	Callback function executed upon the completion of the interface invocation (both successful and unsuccessful invocations)

Object.success callback function: Object res.

Attribute	Type	Description
devices	Array.	List of located devices

Structure attributes	Type	Description
name	string	Bluetooth device name, which may not be available for some devices.
deviceId	string	Identifier used to distinguish devices

Error Return Value

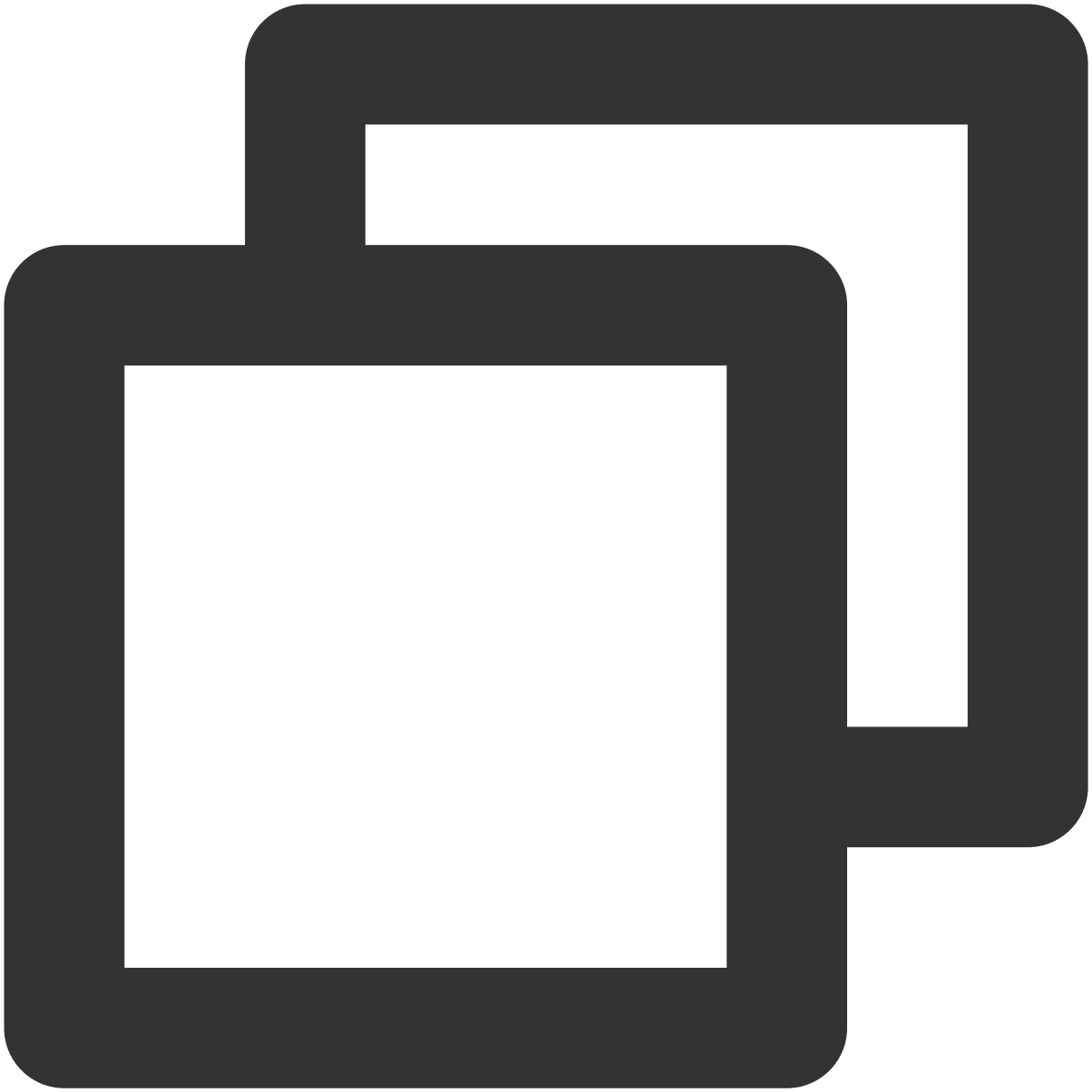
Error code	Error Message	Description
0	ok	Normal
-1	already connect	Connected
10000	not init	Bluetooth Adapter Not Initialized
10001	not available	The current Bluetooth adapter is unavailable.

©2013-2022 Tencent Cloud. All rights reserved.

10002	no device	Specified Device Not Found
10003	connection fail	Connection failed.
10004	no service	Specified Service Not Found
10005	no characteristic	Specified characteristic not found
10006	no connection	The current connection has been disconnected.
10007	property not support	The current characteristic does not support this operation.
10008	system error	Exceptions Reported by All Other Systems
10009	system not support	Exclusive to Android system, and BLE is not supported on versions earlier than 4.3.
10012	operate time out	Connection timeout
10013	invalid_data	The connection deviceId is either null or incorrectly formatted.

Sample Code

©2013-2022 Tencent Cloud. All rights reserved.



```
wx.getConnectedBluetoothDevices({
  services: ['FEE7'],
  success (res) {
    console.log(res)
  }
})
```

getBluetoothDevices

©2013-2022 Tencent Cloud. All rights reserved.

This API is used via wx.getBluetoothDevices(Object object).

Note：

The device list obtained by this interface is all the searched Bluetooth devices during the effective period of the Bluetooth module, if you do not call wx.closeBluetoothAdapter to release the resources in time after the end of the Bluetooth module use process, there will be a call to this interface will return to the previous Bluetooth device searched in the Bluetooth use process, and the device may no longer be around the user and can not be connected.

Feature Description: Gets all Bluetooth devices detected during the effective period of the Bluetooth module, including devices already connected to this device.

Parameter and Description: Object.

Attribute	Type	Default value	Required	Description
success	function	-	No	Callback Function of Successful Interface Call
fail	function	-	No	Callback Function of Failing Interface Call
complete	function	-	No	Callback function executed upon the completion of the interface invocation (both successful and unsuccessful invocations)

Object.success callback function: Object res.

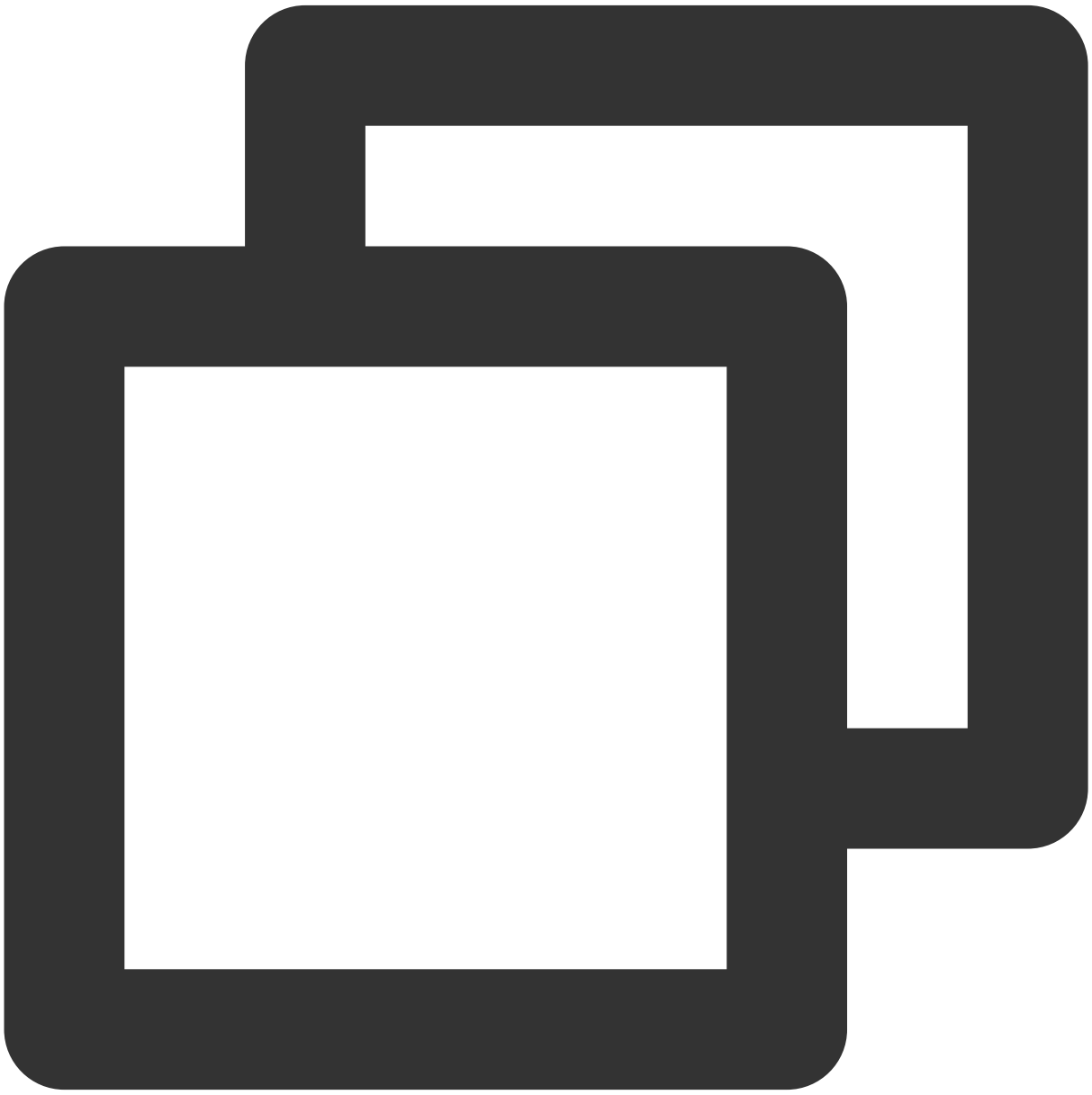
Attribute	Type	Description
devices	Array.<Object>	List of devices connected corresponding to the UUID

Structure attributes	Type	Description
name	string	Bluetooth device name, which may not be available for some devices.
deviceId	string	Bluetooth Device ID
RSSI	number	Signal strength of the current Bluetooth device, measured in dBm.
advertisData	ArrayBuffer	ManufacturerData segment in the broadcast data segment of the current Bluetooth device
advertisServiceUUIDs	Array.<string>	ServiceUUIDs data segment in the broadcast data segment of the current Bluetooth device

©2013-2022 Tencent Cloud. All rights reserved.

localName	string	LocalName data segment in the broadcast data segment of the current Bluetooth device
serviceData	Object	ServiceData segment in the broadcast data segment of the current Bluetooth device
connectable	boolean	Connectivity status of the current Bluetooth device (not supported on Android 8.0 and earlier Android versions)

Sample Code



©2013-2022 Tencent Cloud. All rights reserved.

```
// ArrayBuffer to 16 progress string example
function ab2hex(buffer) {
  var hexArr = Array.prototype.map.call(
    new Uint8Array(buffer),
    function(bit) {
      return ('00' + bit.toString(16)).slice(-2)
    }
  )
  return hexArr.join('');
}
wx.getBluetoothDevices({
  success: function (res) {
    console.log(res)
    if (res.devices[0]) {
      console.log(ab2hex(res.devices[0].advertisData))
    }
  }
})
```

getBluetoothAdapterState

This API is used via wx.getBluetoothAdapterState(Object object).

Feature Description: Gets the status of the local Bluetooth adapter.

Parameter and Description: Object.

Attribute	Type	Default value	Required	Description
success	function	-	No	Callback Function of Successful Interface Call
fail	function	-	No	Callback Function of Failing Interface Call
complete	function	-	No	Callback function executed upon the completion of the interface invocation (both successful and unsuccessful invocations)

Object.success callback function: Object res.

Attribute	Type	Description

©2013-2022 Tencent Cloud. All rights reserved.

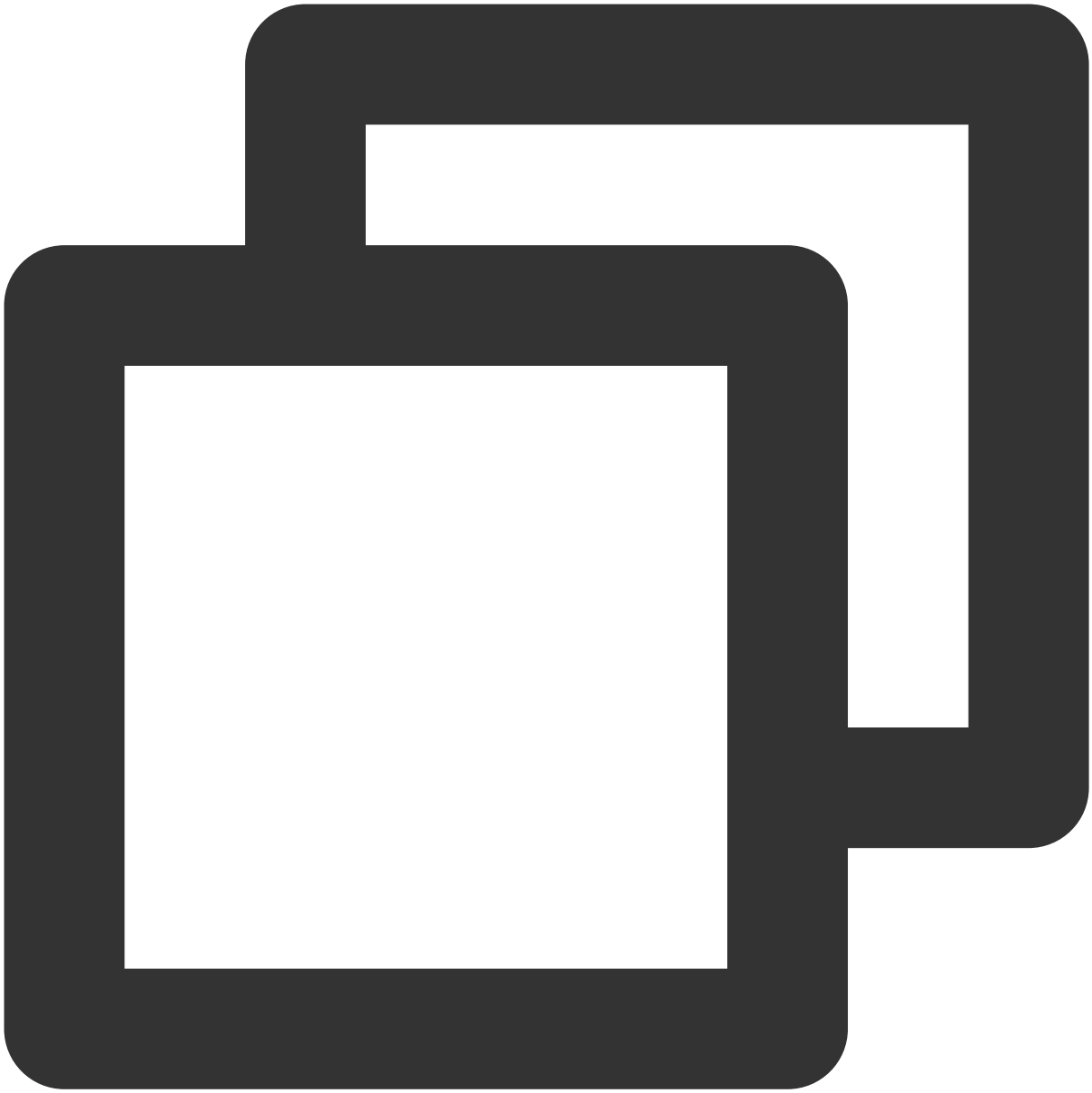
discovering	boolean	Is search for device in progress?
available	boolean	Whether the Bluetooth adapter is available.

Error Return Value

Error code	Error Message	Description
0	ok	Normal
-1	already connect	Connected
10000	not init	Bluetooth Adapter Not Initialized
10001	not available	The current Bluetooth adapter is unavailable.
10002	no device	Specified Device Not Found
10003	connection fail	Connection failed.
10004	no service	Specified Service Not Found
10005	no characteristic	Specified characteristic not found
10006	no connection	The current connection has been disconnected.
10007	property not support	The current characteristic does not support this operation.
10008	system error	Exceptions Reported by All Other Systems
10009	system not support	Exclusive to Android system, and BLE is not supported on versions earlier than 4.3.
10012	operate time out	Connection timeout
10013	invalid_data	The connection deviceId is either null or incorrectly formatted.

Sample Code

©2013-2022 Tencent Cloud. All rights reserved.



```
wx.getBluetoothAdapterState({
  success (res) {
    console.log(res)
  }
})
```

closeBluetoothAdapter

©2013-2022 Tencent Cloud. All rights reserved.

This API is used via wx.closeBluetoothAdapter(Object object).

Feature Description: Disables the Bluetooth module. Invoking this method will disconnect all established connections and free up system resources. It is recommended to pair this with wx.openBluetoothAdapter after the Bluetooth process.

Parameter and Description: Object.

Attribute	Type	Default value	Required	Description
success	function	-	No	Callback Function of Successful Interface Call
fail	function	-	No	Callback Function of Failing Interface Call
complete	function	-	No	Callback function executed upon the completion of the interface invocation (both successful and unsuccessful invocations)

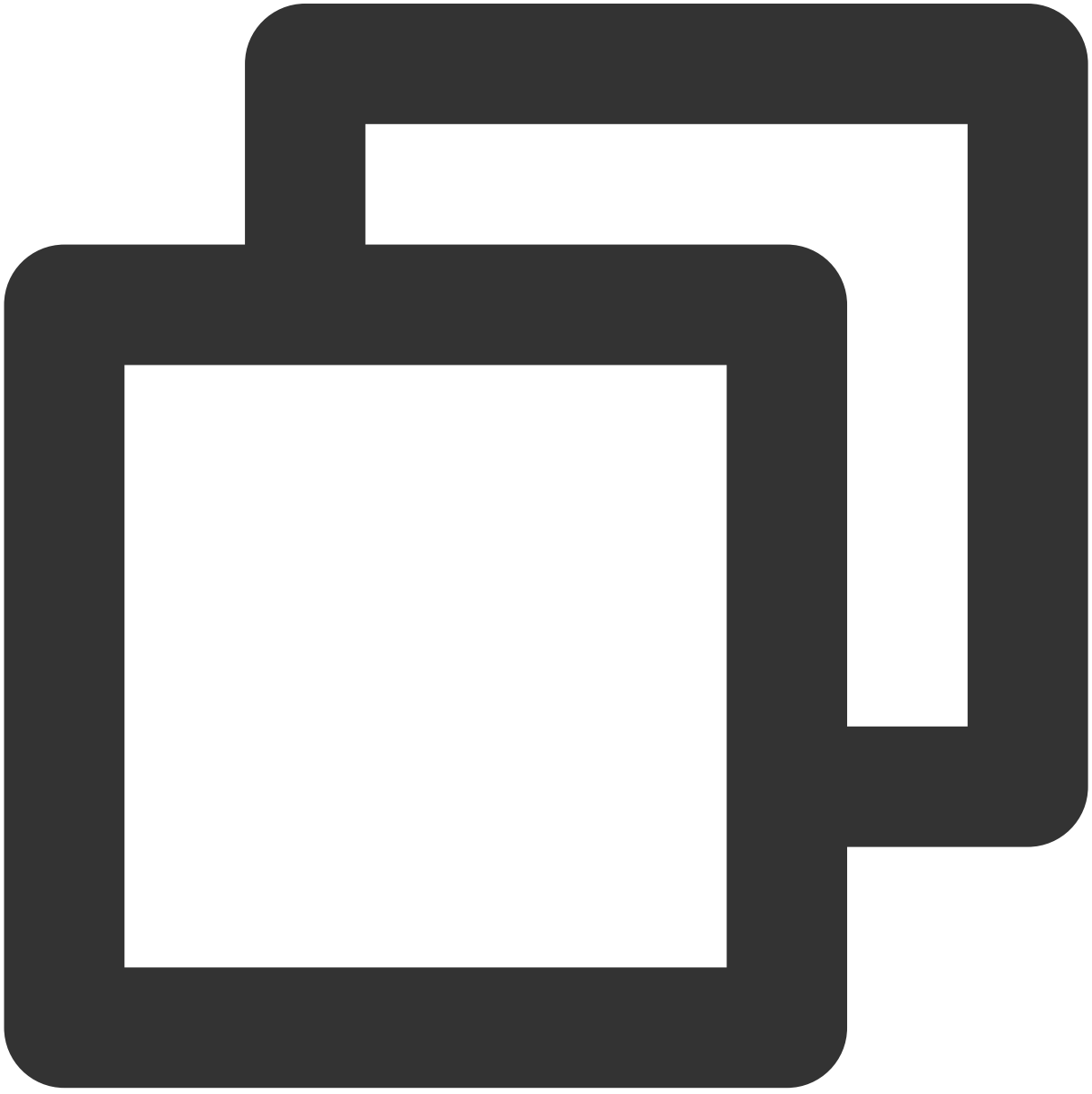
Error Return Value

Error code	Error Message	Description
0	ok	Normal
-1	already connect	Connected
10000	not init	Bluetooth Adapter Not Initialized
10001	not available	The current Bluetooth adapter is unavailable.
10002	no device	Specified Device Not Found
10003	connection fail	Connection failed.
10004	no service	Specified Service Not Found
10005	no characteristic	Specified characteristic not found
10006	no connection	The current connection has been disconnected.
10007	property not support	The current characteristic does not support this operation.
10008	system error	Exceptions Reported by All Other Systems
10009	system not support	Exclusive to Android system, and BLE is not supported on versions earlier than 4.3.

©2013-2022 Tencent Cloud. All rights reserved.

10012	operate time out	Connection timeout
10013	invalid_data	The connection deviceId is either null or incorrectly formatted.

Sample Code



```
wx.closeBluetoothAdapter({
  success (res) {
    console.log(res)
  }
```

©2013-2022 Tencent Cloud. All rights reserved.

```
})
```

onBluetoothDeviceFound

This API is used via wx.onBluetoothDeviceFound(function listener).

Note:

If a device is called back in wx.onBluetoothDeviceFound, this device will be added to the array obtained by the wx.getBluetoothDevices interface.

When a Bluetooth device is discovered, the system generally returns the device name from the LocalName field in the broadcast packet as the name field. However, if a connection is established with the Bluetooth device, the system's name field will change to the GattName obtained from the Bluetooth device. If there is a need to dynamically change and display the device name, it is recommended to use the localName field.

Please note that some Android models require location permissions to search for devices. Ensure that location permissions are enabled.

Feature Description: Monitors events of finding new devices.

Parameter and Description: function listener, the listener function for events that search for new devices, with the arguments object res as follows:

Attribute	Type	Description
devices	Array.<Object>	List of newly located devices

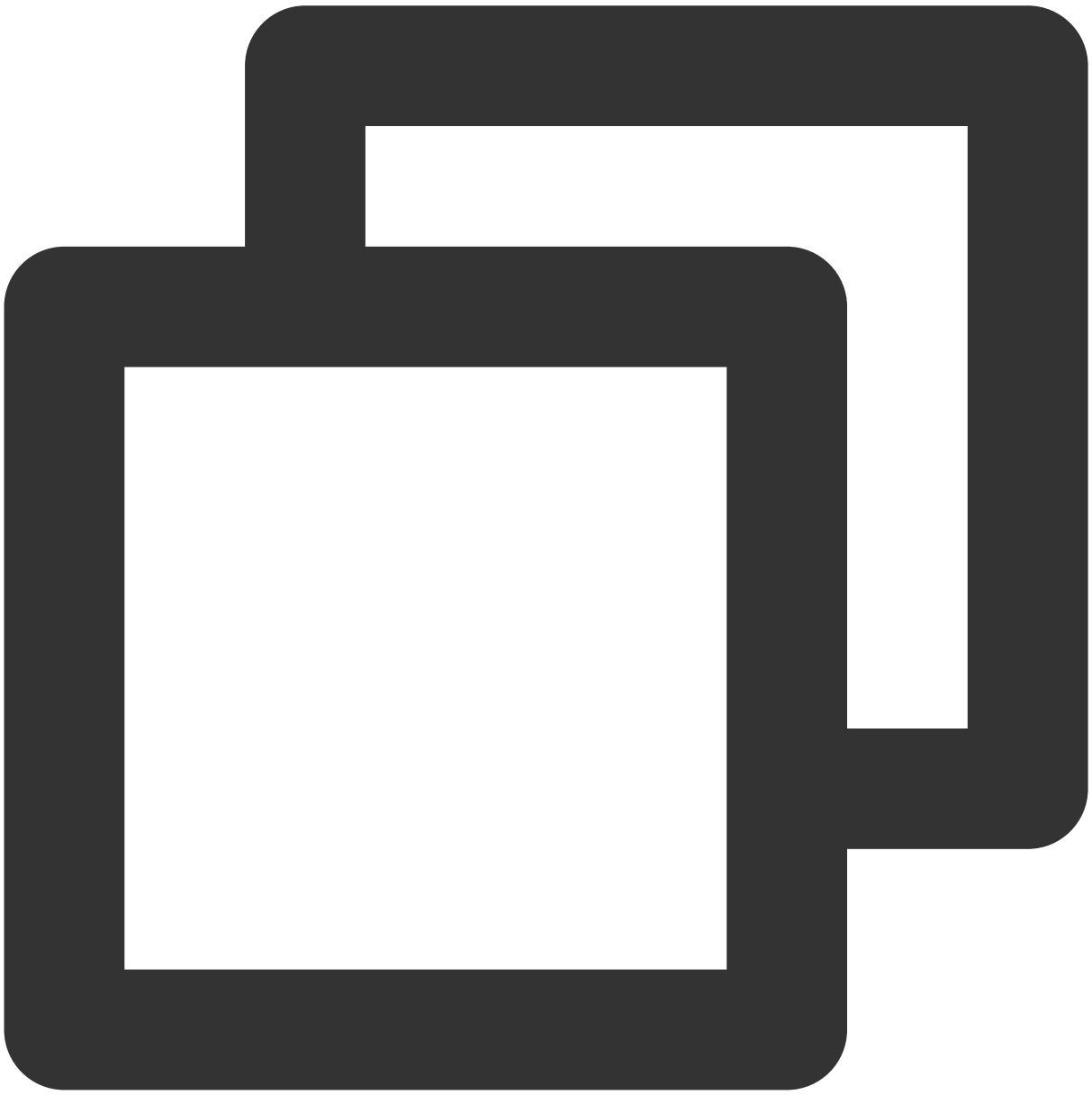
Structure attribute of devices

Structure attributes	Type	Description
name	string	Bluetooth device name, which may not be available for some devices.
deviceId	string	Bluetooth Device ID
RSSI	number	Signal strength of the current Bluetooth device, measured in dBm.
advertisData	ArrayBuffer	ManufacturerData segment in the broadcast data segment of the current Bluetooth device
advertisServiceUUIDs	Array. <string>	ServiceUUIDs data segment in the broadcast data segment of the current Bluetooth device

©2013-2022 Tencent Cloud. All rights reserved.

localName	string	LocalName data segment in the broadcast data segment of the current Bluetooth device
serviceData	Object	ServiceData segment in the broadcast data segment of the current Bluetooth device
connectable	boolean	Connectivity status of the current Bluetooth device (not supported on Android 8.0 and earlier Android versions)

Sample Code



©2013-2022 Tencent Cloud. All rights reserved.

```
// Example of converting ArrayBuffer to hexadecimal string
function ab2hex(buffer) {
  var hexArr = Array.prototype.map.call(
    new Uint8Array(buffer),
    function(bit) {
      return ('00' + bit.toString(16)).slice(-2)
    }
  )
  return hexArr.join('');
}
wx.onBluetoothDeviceFound(function(res) {
  var devices = res.devices;
  console.log('new device list has founded')
  console.dir(devices)
  console.log(ab2hex(devices[0].advertisData))
})
```

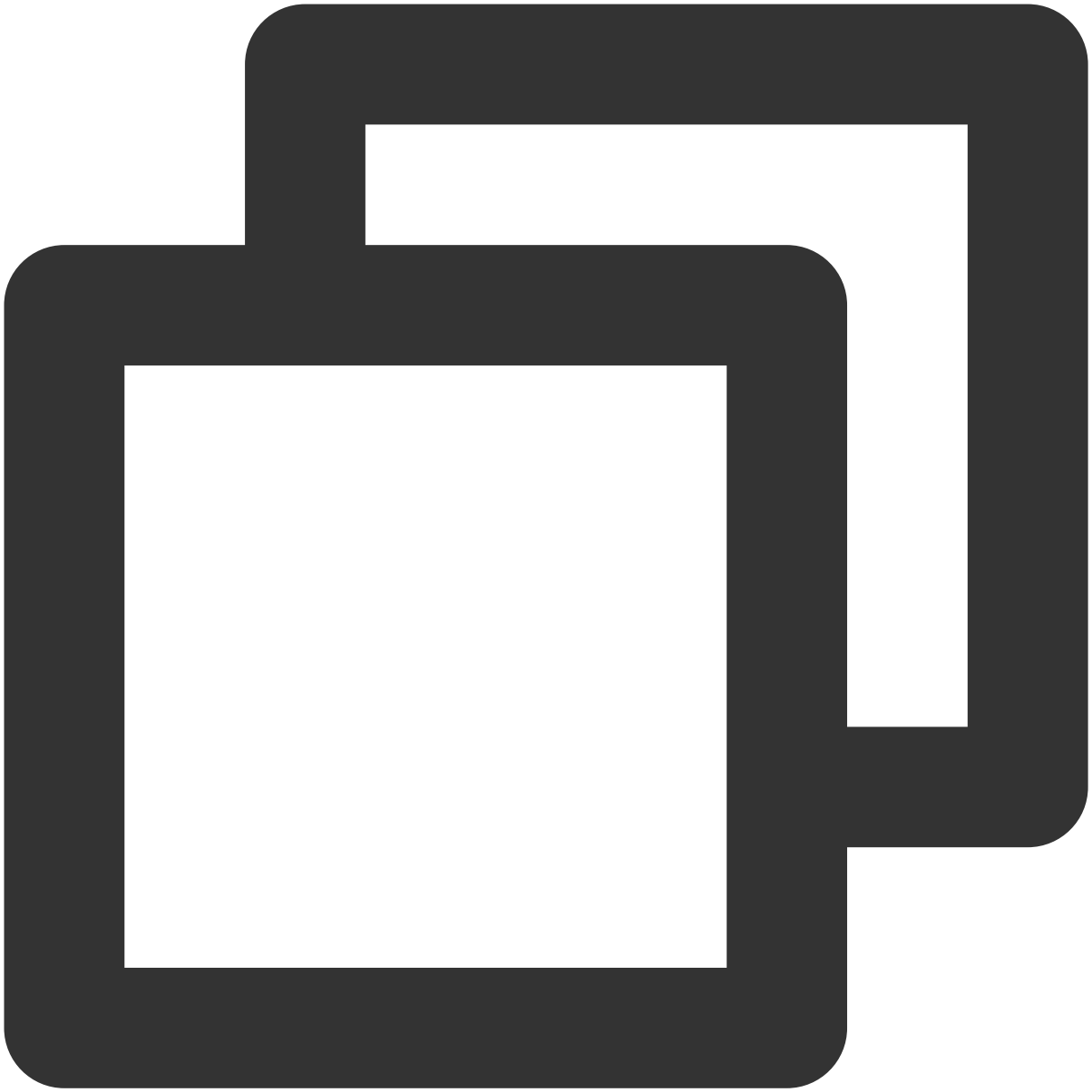
offBluetoothDeviceFound

This API is used via wx.offBluetoothDeviceFound()

Feature Description: Removes all listener functions for events of finding new devices.

Sample Code

©2013-2022 Tencent Cloud. All rights reserved.



```
wx.offBluetoothDeviceFound()
```

onBluetoothAdapterStateChange

This API is used via wx.onBluetoothAdapterStateChange(function listener).

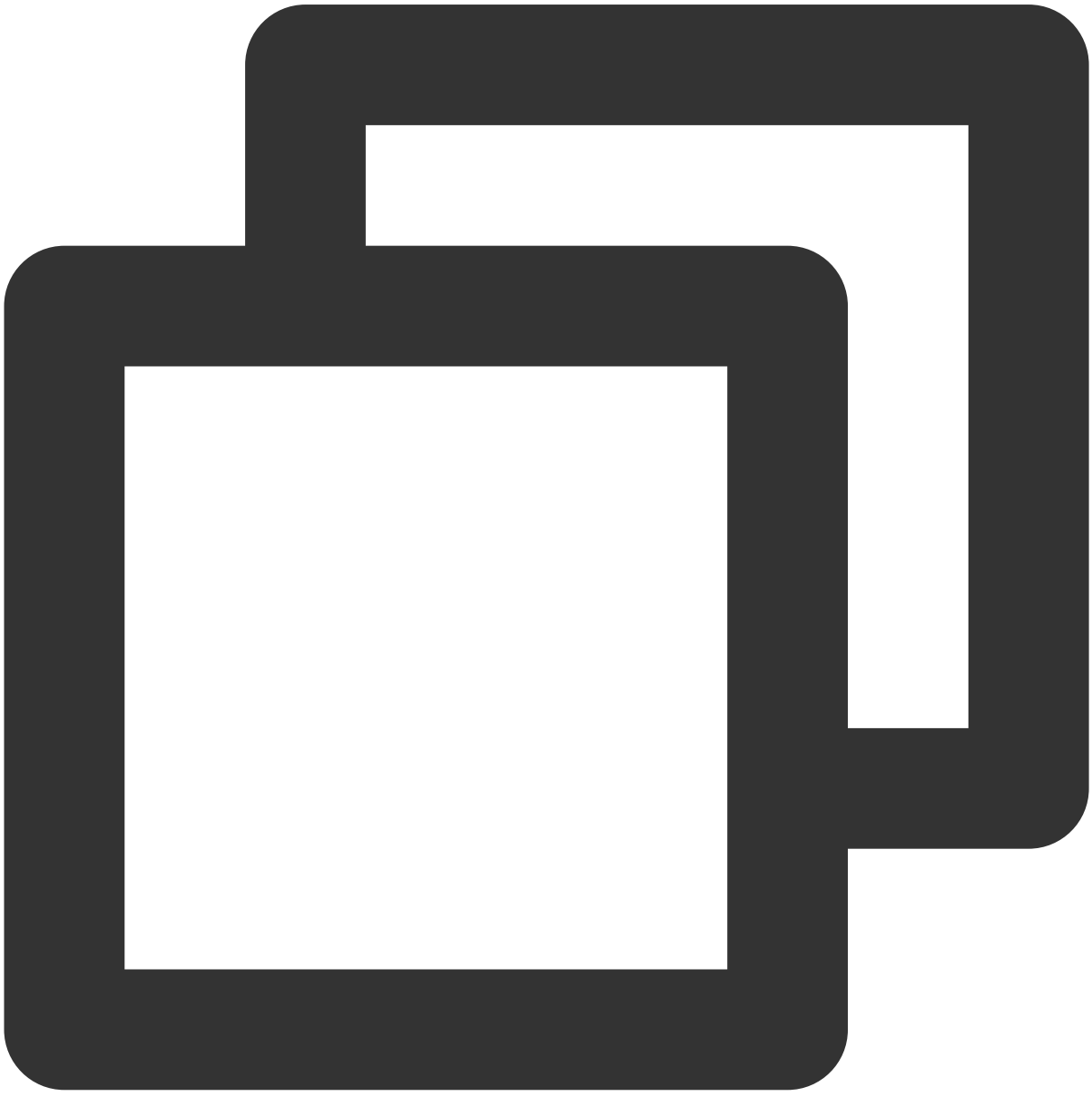
©2013-2022 Tencent Cloud. All rights reserved.

Feature Description: Monitors events of Bluetooth adapter status changes.

Parameters and Description: Object res parameter, function listener, the listener function for events of changes to Bluetooth adapter status.

Attribute	Type	Description
available	boolean	Whether the Bluetooth adapter is available.
discovering	boolean	Whether the Bluetooth adapter is in search mode.

Sample Code



©2013-2022 Tencent Cloud. All rights reserved.

```
wx.onBluetoothAdapterStateChange(function (res) {
  console.log('adapterState changed, now is', res)
})
```

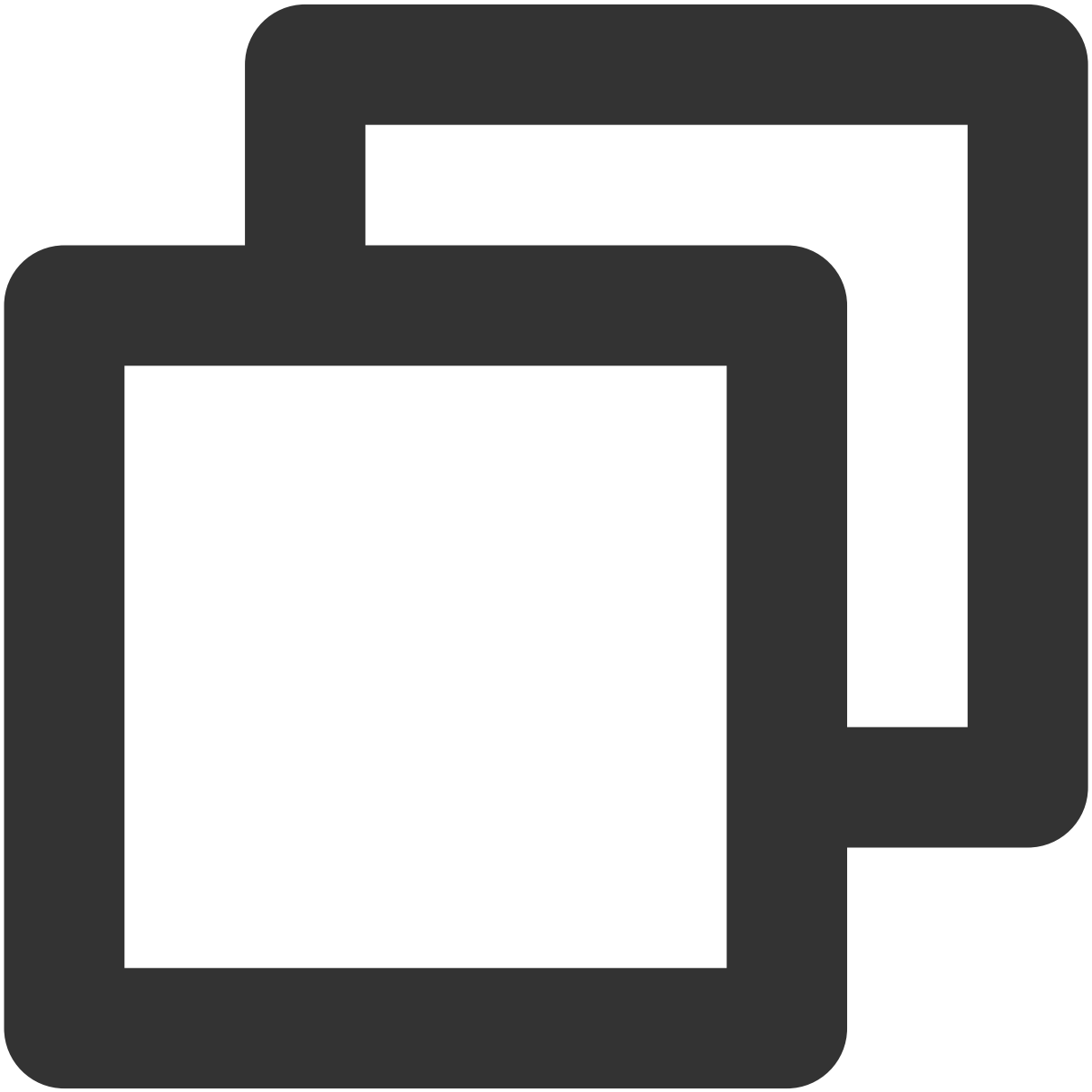
offBluetoothAdapterStateChange

This API is used via wx.offBluetoothAdapterStateChange().

Feature Description: Removes all listener functions for events of changes to Bluetooth adapter status.

Sample Code

©2013-2022 Tencent Cloud. All rights reserved.



```
wx.offBluetoothAdapterStateChange()
```

makeBluetoothPair

This API is used via wx.makeBluetoothPair(Object object).

©2013-2022 Tencent Cloud. All rights reserved.

Feature Description: Bluetooth pairing interface, only supported by Android. Normally (when you need to specify a pin code or password) the system will take over the pairing process and call wx.createBLEConnection directly. This interface should only be used if the developer doesn't want the user to enter the pin code manually and the real phone verification confirms that it works.

Parameter and Description: Object

Attribute	Type	Default value	Required	Description
deviceId	string	-	Yes	Bluetooth Device ID
pin	string	-	Yes	Pin code, in Base64 format.
timeout	number	20000	No	Timeout duration, measured in milliseconds (ms).
success	function	-	No	Callback Function of Successful Interface Call
fail	function	-	No	Callback Function of Failing Interface Call
complete	function	-	No	Callback function executed upon the completion of the interface invocation (both successful and unsuccessful invocations)

isBluetoothDevicePaired

This API is used via wx.isBluetoothDevicePaired(Object object).

Feature Description: This function queries whether a Bluetooth device is paired, and is only supported on Android.

Parameter and Description: Object.

Attribute	Type	Default value	Required	Description
deviceId	string	-	Yes	Bluetooth Device ID
success	function	-	No	Callback Function of Successful Interface Call
fail	function	-	No	Callback Function of Failing Interface Call
complete	function	-	No	Callback function executed upon the completion of the interface invocation (both successful and unsuccessful invocations)

©2013-2022 Tencent Cloud. All rights reserved.

©2013-2022 Tencent Cloud. All rights reserved.

Location

Last updated：2024-07-12 18:27:40

getLocation

This method is used via wx.getLocation(Object object).

Note:

If you want to use this interface, you need to declare it in app.json, otherwise you won't be able to use it properly.

User authorisation scope.userLocation is required before calling.

Feature Description: Get the current geolocation, speed. This interface cannot be called when the user leaves the mini program. High frequency calls can lead to power consumption, if necessary use the continuous location interface wx.onLocationChange.

Parameter and Description: Object.

Attribute	Type	Default value	Required	Description
type	string	wgs84	No	WGS84 returns GPS coordinates, while GCJ02 provides coordinates that can be used with wx.openLocation.
altitude	string	false	No	Passing in "true" will return altitude information. As acquiring altitude requires high precision, it will slow down the response of the interface.
success	function	-	No	Callback Function of Successful Interface Call
fail	function	-	No	Callback Function of Failing Interface Call
complete	function	-	No	Callback function executed upon the completion of the interface invocation (both successful and unsuccessful invocations)

Parameters for object.success callback function: Object res.

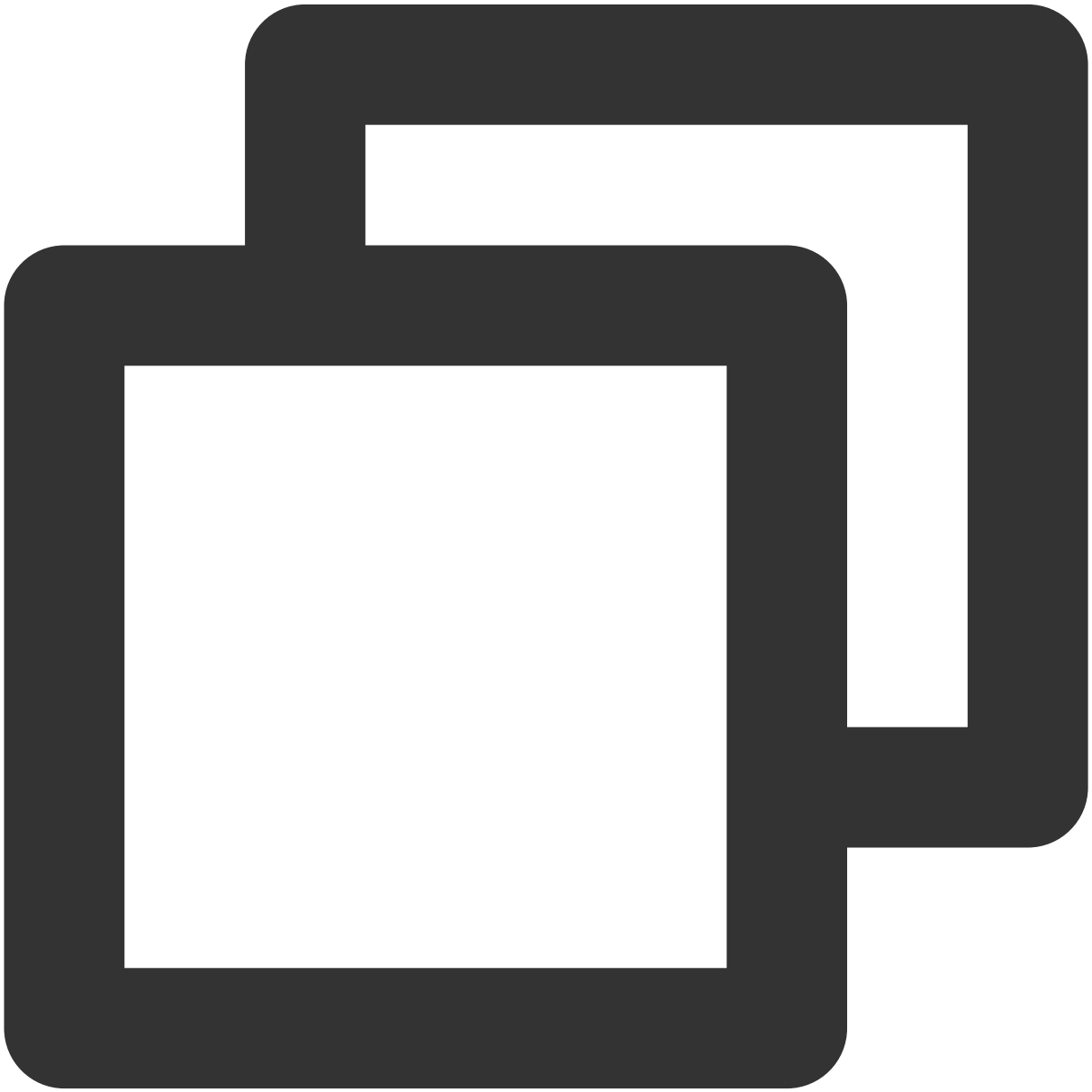
Attribute	Type	Description
latitude	number	Latitude, ranging from -90 to 90, with negative values indicating southern latitude.

©2013-2022 Tencent Cloud. All rights reserved.

longitude	number	Longitude, ranging from -180 to 180, with negative values indicating western longitude.
speed	number	Velocity, measured in meters per second (m/s)
accuracy	number	Precision of the location
altitude	number	Height, measured in (m)
verticalAccuracy	number	Vertical accuracy, measured in meters (m). (Cannot be obtained on Android, returning 0)
horizontalAccuracy	number	Horizontal accuracy, measured in meters (m)

Sample Code

©2013-2022 Tencent Cloud. All rights reserved.



```
wx.getLocation({
  type: 'gcj02',
  success (res) {
    const latitude = res.latitude
    const longitude = res.longitude
    const speed = res.speed
    const accuracy = res.accuracy
  }
})
```

Note:

©2013-2022 Tencent Cloud. All rights reserved.

The location simulation in the tool uses IP positioning, which may introduce some degree of error. Currently, the tool only supports gcj02 coordinates.
When using third-party services for reverse address resolution, make sure the default coordinate system of the third-party service is confirmed for accurate coordinate conversion.

choosePoi

This API is used via wx.choosePoi(Object object).

Note：
If you want to use this interface, you need to declare scope.userLocation in app.json, otherwise you won't be able to use it properly.
Feature Description: Opens the POI list to select a location, which supports both vague positioning (accurate to the city) and precise positioning in a mixed selection.
Parameter and Description: Object.

Attribute	Type	Default value	Required	Description
success	function	-	No	Callback Function of Successful Interface Call
fail	function	-	No	Callback Function of Failing Interface Call
complete	function	-	No	Callback function executed upon the completion of the interface invocation (both successful and unsuccessful invocations)

Parameters for object.success callback function: Object res.

Attribute	Type	Description
type	number	When selecting a city, the value is 1; when selecting an exact location, the value is 2.
city	number	City Name
name	string	Location Name
address	string	Exact Address
latitude	number	Latitude, a floating-point number, ranging from -90 to 90, where a negative number indicates southern latitude, using the gcj02 National Administration of Surveying, Mapping and Geoinformation coordinate system (to be deprecated).

©2013-2022 Tencent Cloud. All rights reserved.

longitude	number	Longitude, a floating-point number, ranging from -180 to 180, where a negative number indicates western longitude, using the gcj02 National Administration of Surveying, Mapping and Geoinformation coordinate system (to be deprecated).

chooseLocation

Note：
If you want to use this interface, you need to declare scope.userLocation in app.json, otherwise you won't be able to use it properly.

This API is used via wx.chooseLocation(Object object).

Feature Description: Opens the map to select a location.
Parameter and Description: Object.

Attribute	Type	Default value	Required	Description
type	string	tencent	No	Tencent Maps (tencent), Google Maps (google), supported by IDE only.
latitude	number	-	No	Target location's latitude.
longitude	number	-	No	Target location's longitude.
success	function	-	No	Callback Function of Successful Interface Call
fail	function	-	No	Callback Function of Failing Interface Call
complete	function	-	No	Callback function executed upon the completion of the interface invocation (both successful and unsuccessful invocations)

Parameters for object.success callback function: Object res.

Attribute	Type	Description
name	string	Location Name
address	string	Exact Address
latitude	number	Latitude, a floating-point number, ranging from -90 to 90, where a negative number

©2013-2022 Tencent Cloud. All rights reserved.

		indicates southern latitude, using the gcj02 National Administration of Surveying, Mapping and Geoinformation coordinate system.
longitude	number	Longitude, a floating-point number, ranging from -180 to 180, where a negative number indicates western longitude, using the gcj02 National Administration of Surveying, Mapping and Geoinformation coordinate system.

stopLocationUpdate

This API is used via wx.stopLocationUpdate(Object object).

Feature Description: Stops monitoring real-time location changes, halting message reception in both the frontend and backend.

Parameter and Description: Object.

Attribute	Type	Default value	Required	Description
success	function	-	No	Callback Function of Successful Interface Call
fail	function	-	No	Callback Function of Failing Interface Call
complete	function	-	No	Callback function executed upon the completion of the interface invocation (both successful and unsuccessful invocations)

startLocationUpdateBackground

This API is used via wx.startLocationUpdateBackground(Object object).

Feature Description: Enables the mini program to receive location messages when entering the frontend and backend, which requires guiding the user to give "authorization". Once authorized, the mini program can receive location message updates during operation or when in the backend.

Parameter and Description: Object.

Attribute	Type	Default value	Required	Description

©2013-2022 Tencent Cloud. All rights reserved.

type	string	gcj02	No	WGS84 returns GPS coordinates, while GCJ02 provides coordinates that can be used with wx.openLocation.
success	function	-	No	Callback Function of Successful Interface Call
fail	function	-	No	Callback Function of Failing Interface Call
complete	function	-	No	Callback function executed upon the completion of the interface invocation (both successful and unsuccessful invocations)

startLocationUpdate

This API is used via wx.startLocationUpdate(Object object).

Feature Description: Enables the mini program to receive location messages upon entering the frontend.

Parameter and Description: Object.

Attribute	Type	Default value	Required	Description
type	string	Coordinates, gcj02	No	WGS84 returns GPS coordinates, while GCJ02 provides coordinates that can be used with wx.openLocation.
success	function	-	No	Callback Function of Successful Interface Call
fail	function	-	No	Callback Function of Failing Interface Call
complete	function	-	No	Callback function executed upon the completion of the interface invocation (both successful and unsuccessful invocations)

openLocation

This API is used via wx.openLocation(Object object).

Feature Description: Uses the built-in map to view the location.

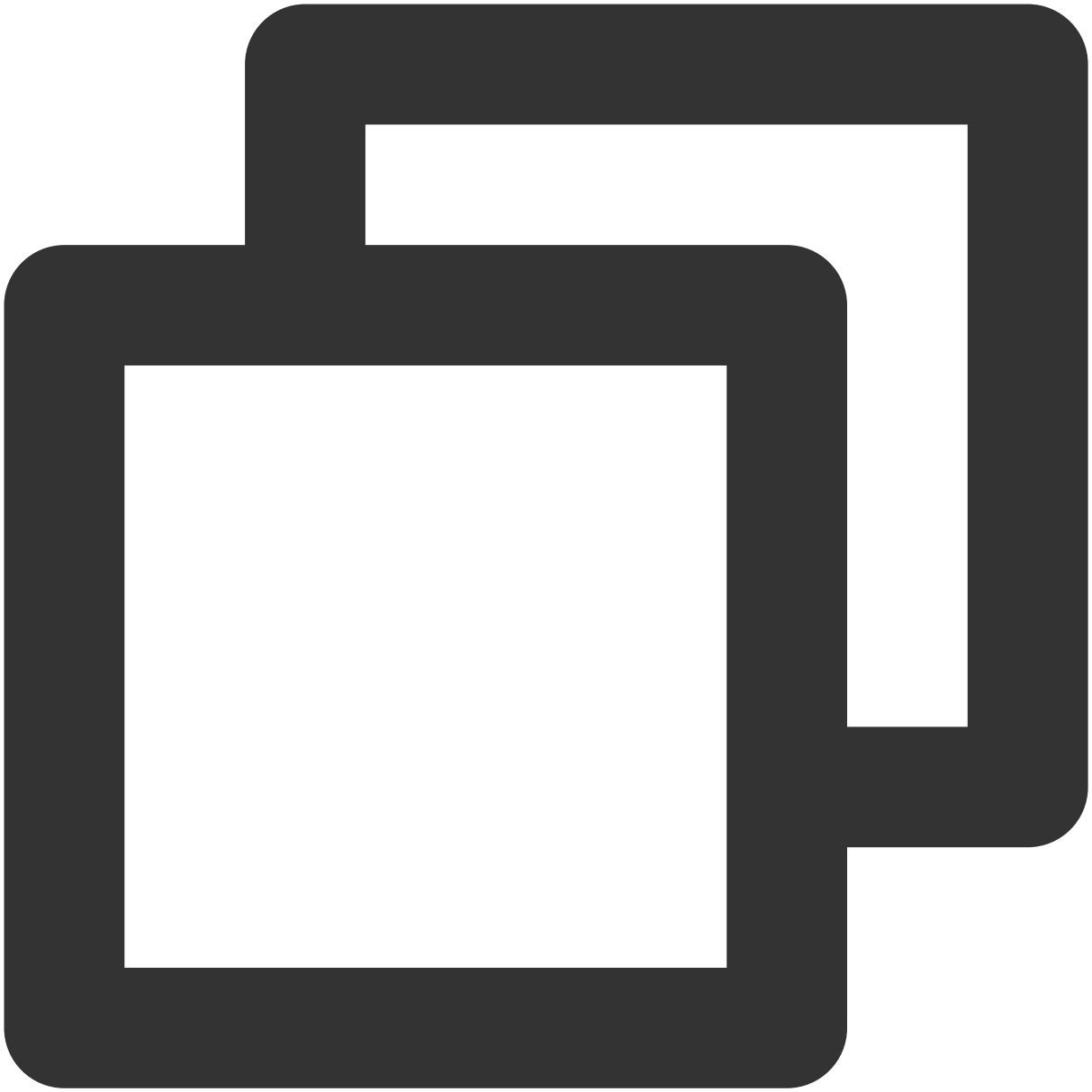
©2013-2022 Tencent Cloud. All rights reserved.

Parameter and Description: Object.

Attribute	Type	Default value	Required	Description
type	string	tencent	No	Tencent Maps (tencent), Google Maps (google), supported by IDE only.
latitude	number	-	Yes	Latitude, ranging from -90 to 90, where a negative number indicates southern latitude, using the gcj02 National Administration of Surveying, Mapping and Geoinformation coordinate system.
longitude	number	-	Yes	Longitude, ranging from -180 to 180, where a negative number indicates western longitude, using the gcj02 National Administration of Surveying, Mapping and Geoinformation coordinate system.
scale	number	18	No	Zoom scale, ranging from 5 to 18.
name	string	-	No	Location Name
address	string	-	No	Detailed description of the address.
success	function	-	No	Callback Function of Successful Interface Call
fail	function	-	No	Callback Function of Failing Interface Call
complete	function	-	No	Callback function executed upon the completion of the interface invocation (both successful and unsuccessful invocations)

Sample Code

©2013-2022 Tencent Cloud. All rights reserved.



```
wx.getLocation({
 type: 'gcj02', // Returns longitude and latitude that can be used for wx.openLocat
 success (res) {
   const latitude = res.latitude
   const longitude = res.longitude
   wx.openLocation({
     latitude,
     longitude,
     scale: 18
   })
 }
```

©2013-2022 Tencent Cloud. All rights reserved.

```
})
```

onLocationChangeError

This API is used via wx.onLocationChangeError(function listener).

Feature Description: Cancels monitoring, which is triggered when the continuous location interface returns a failure.

Parameter and Description: Callback function of "function callback".

Attribute	Type	Description
errCode	number	errCode

onLocationChange

This API is used via wx.onLocationChange(function listener).

Feature Description: Monitors real-time geographical location change events, which needs to be used in conjunction with wx.startLocationUpdateBackground and wx.startLocationUpdate.

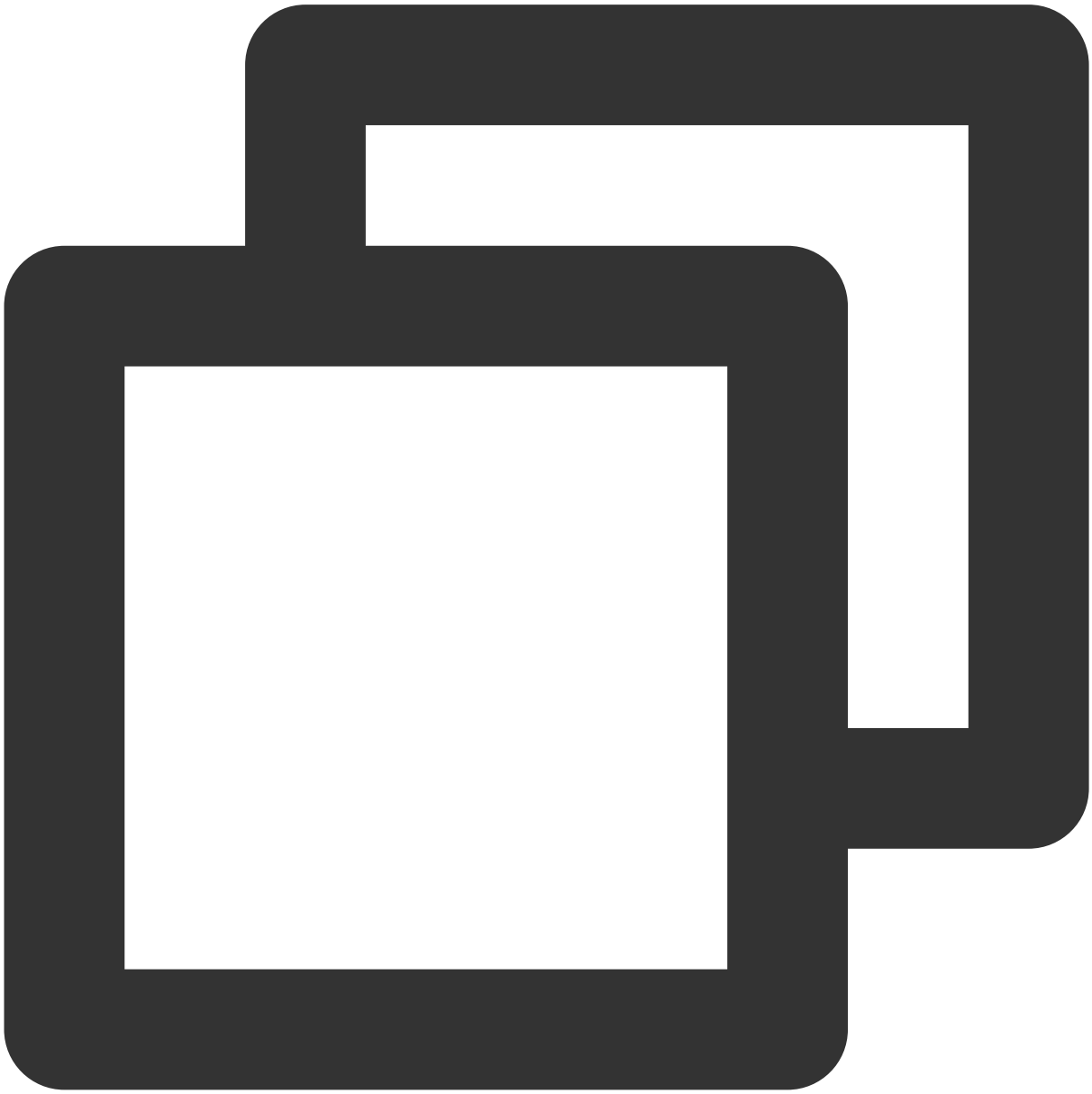
Parameter and Description: function callback, real-time geolocation change event listener function, with the following arguments Object res:

Attribute	Type	Description
latitude	number	Latitude, ranging from -90 to 90, where a negative number indicates southern latitude, using the gcj02 National Administration of Surveying, Mapping and Geoinformation coordinate system.
longitude	number	Longitude, ranging from -180 to 180, where a negative number indicates western longitude, using the gcj02 National Administration of Surveying, Mapping and Geoinformation coordinate system.
speed	number	Velocity, measured in meters per second (m/s)
accuracy	number	Precision of the location
altitude	number	Height, measured in (m)

©2013-2022 Tencent Cloud. All rights reserved.

verticalAccuracy	number	Vertical accuracy, measured in meters (m). (Cannot be obtained on Android, returning 0)
horizontalAccuracy	number	Horizontal accuracy, measured in meters (m)

Sample Code



```
const _locationChangeFn = function(res) {
 console.log('location change', res)
}
wx.onLocationChange(_locationChangeFn)
```

©2013-2022 Tencent Cloud. All rights reserved.

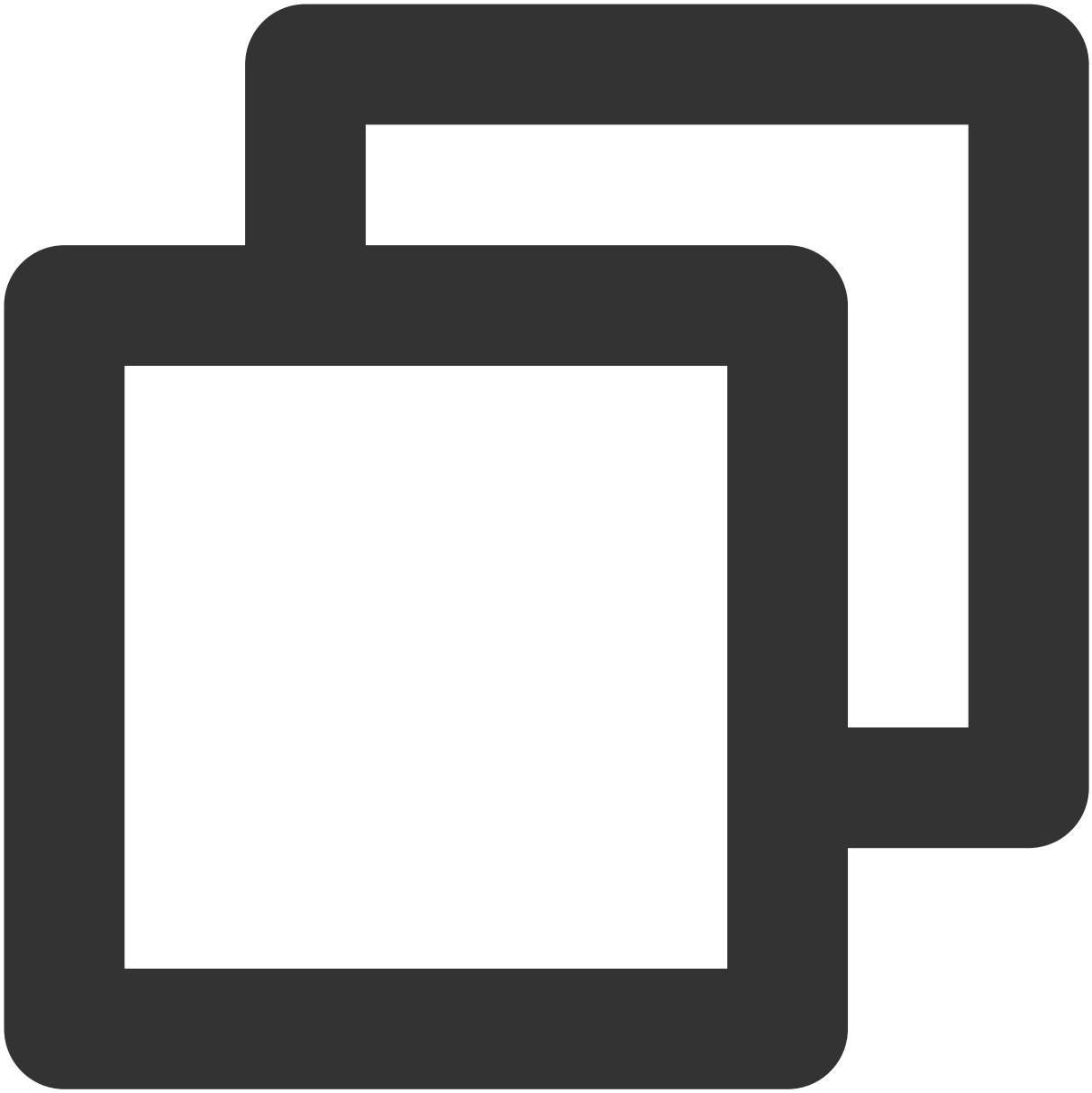
```
wx.offLocationChange(_locationChangeFn)
```

offLocationChangeError

This API is used via wx.offLocationChangeError(function listener).

Feature Description: Cancels monitoring, which is triggered when the continuous location interface returns a failure.
Parameter and Description: function listener, onLocationChangeError The listener function passed in. Not passing this parameter removes all listener functions.
Sample Code

©2013-2022 Tencent Cloud. All rights reserved.



```
const listener = function (res) { console.log(res) }

wx.onLocationChangeError(listener)
wx.offLocationChangeError(listener) // You need to pass the same function object as
```

offLocationChange

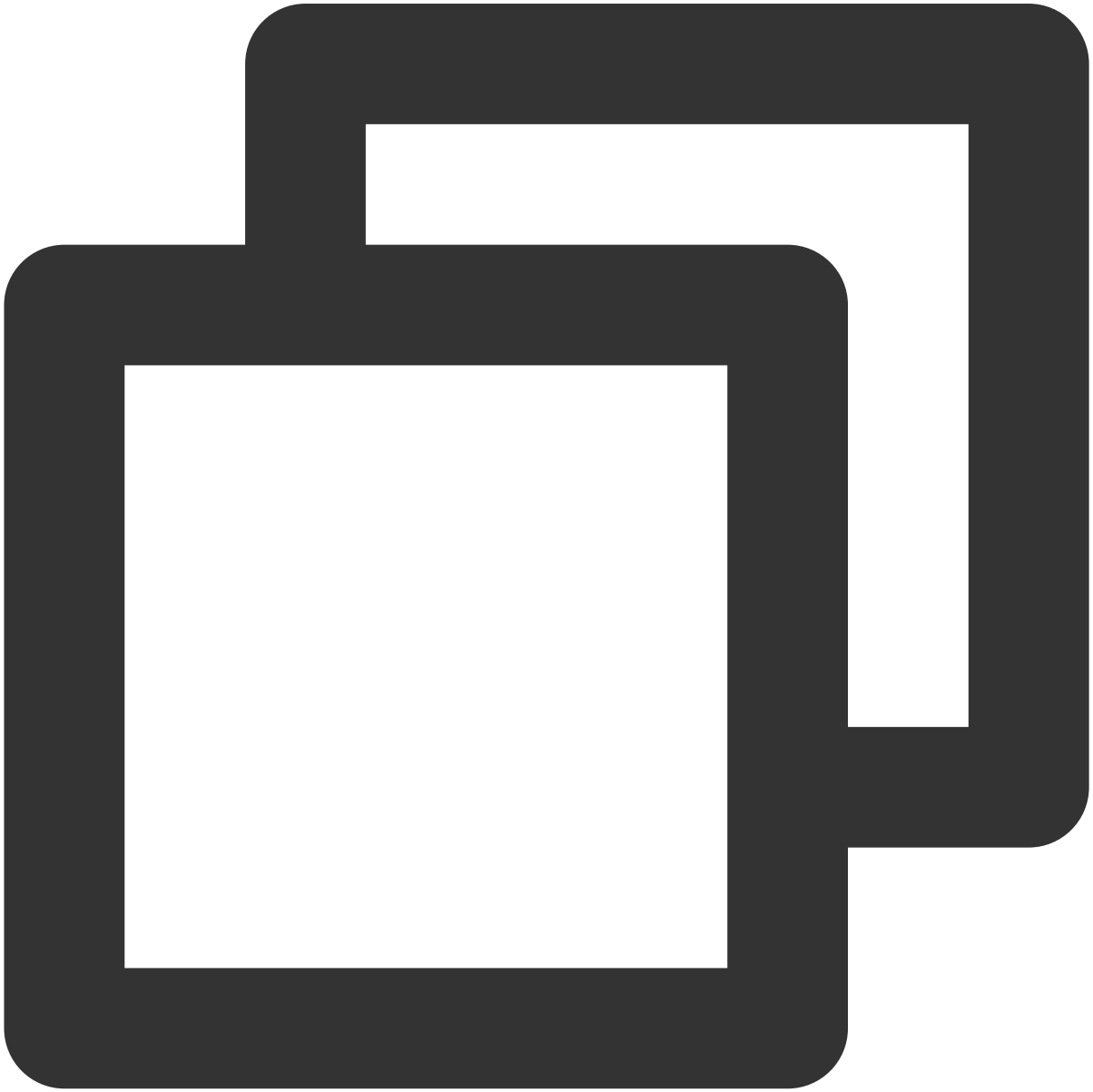
©2013-2022 Tencent Cloud. All rights reserved.

This API is used via wx.offLocationChange(function listener).

Feature Description: Cancels the monitoring of real-time geographical location change events.

Parameter and Description: function listener, the listener passed in by onLocationChange. Not passing this parameter removes all listener functions.

Sample Code



```
const listener = function (res) { console.log(res) }
```

©2013-2022 Tencent Cloud. All rights reserved.

```
wx.onLocationChangeError(listener)
wx.offLocationChangeError(listener) // You need to pass the same function object as
```

getFuzzyLocation

This API is used via wx.getFuzzyLocation(Object object).

Feature Description: Gets the current fuzzy geographical location.

Parameter and Description: Object.

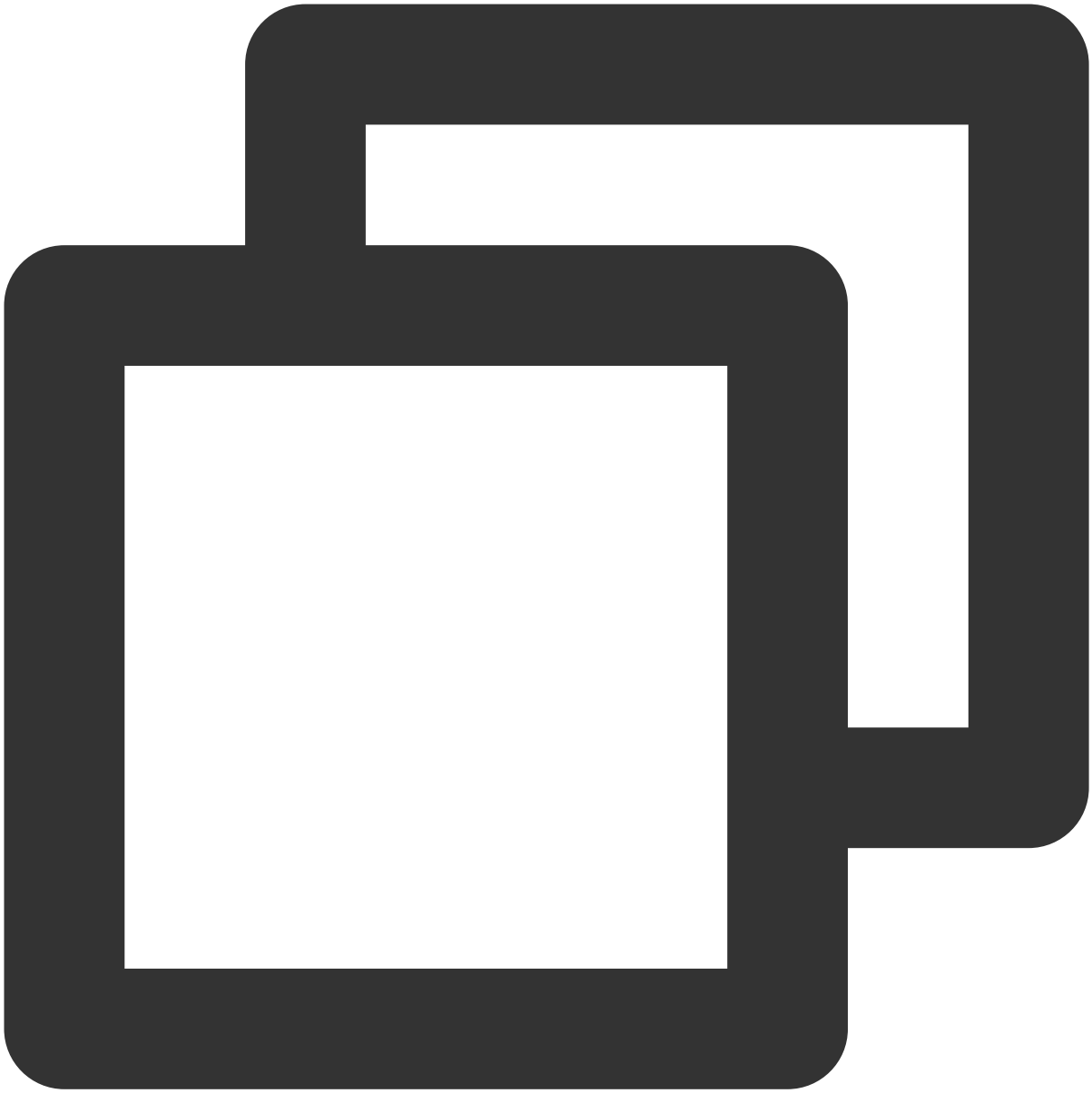
Attribute	Type	Default value	Required	Description
type	string	wgs84	No	WGS84 returns GPS coordinates, while GCJ02 provides coordinates that can be used with wx.openLocation.
success	function	-	No	Callback Function of Successful Interface Call
fail	function	-	No	Callback Function of Failing Interface Call
complete	function	-	No	Callback function executed upon the completion of the interface invocation (both successful and unsuccessful invocations)

Parameters for object.success callback function: Object res.

Attribute	Type	Description
latitude	number	Latitude, ranging from -90 to 90, with negative values indicating southern latitude.
longitude	number	Longitude, ranging from -180 to 180, with negative values indicating western longitude.

Sample Code

©2013-2022 Tencent Cloud. All rights reserved.



```
wx.getFuzzyLocation({
 type: 'wgs84',
 success (res) {
   const latitude = res.latitude
   const longitude = res.longitude
 }
})
```

©2013-2022 Tencent Cloud. All rights reserved.

Open Interfaces

Last updated：2024-07-12 18:26:53

Login

login

This method is used via wx.login(Object object).

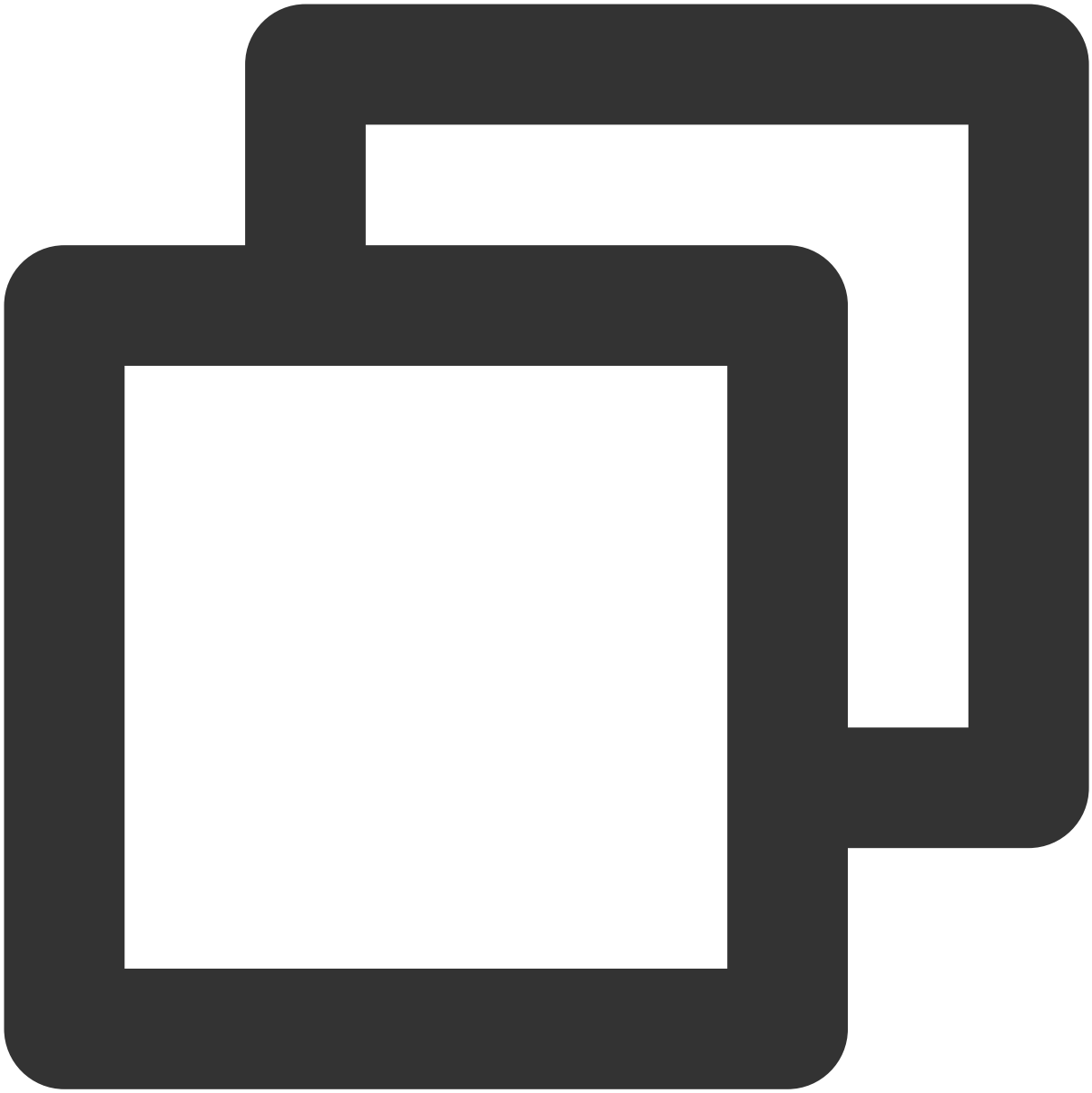
Feature Description: The IDE does not currently support this API. It requires coordination with the host client, and the returned content must be provided by the host client.

Parameter and Description: Object.

Attribute	Type	Default value	Required	Description
success	function	-	No	Callback Function of Successful Interface Call
fail	function	-	No	Callback Function of Failing Interface Call
complete	function	-	No	Callback function executed upon the completion of the interface invocation (both successful and unsuccessful invocations)

Sample Code

©2013-2022 Tencent Cloud. All rights reserved.



```
wx.login({
  success(res) {
    console.log(res ,"---------------info, host app return");
  }
})
```

checkSession

This API is used via wx.checkSession(Object object).

©2013-2022 Tencent Cloud. All rights reserved.

Feature Description:

This checks whether the login status has expired. The user login status obtained through the wx.login interface has a certain validity period. The longer the user does not use the mini program, the more likely the user's login status is to become invalid. Conversely, if the user continues to use the mini program, the user's login status remains valid. The specific validity logic is maintained by the mini program provider and is transparent to developers. Developers only need to call the wx.checkSession interface to check whether the current user's login status is valid.

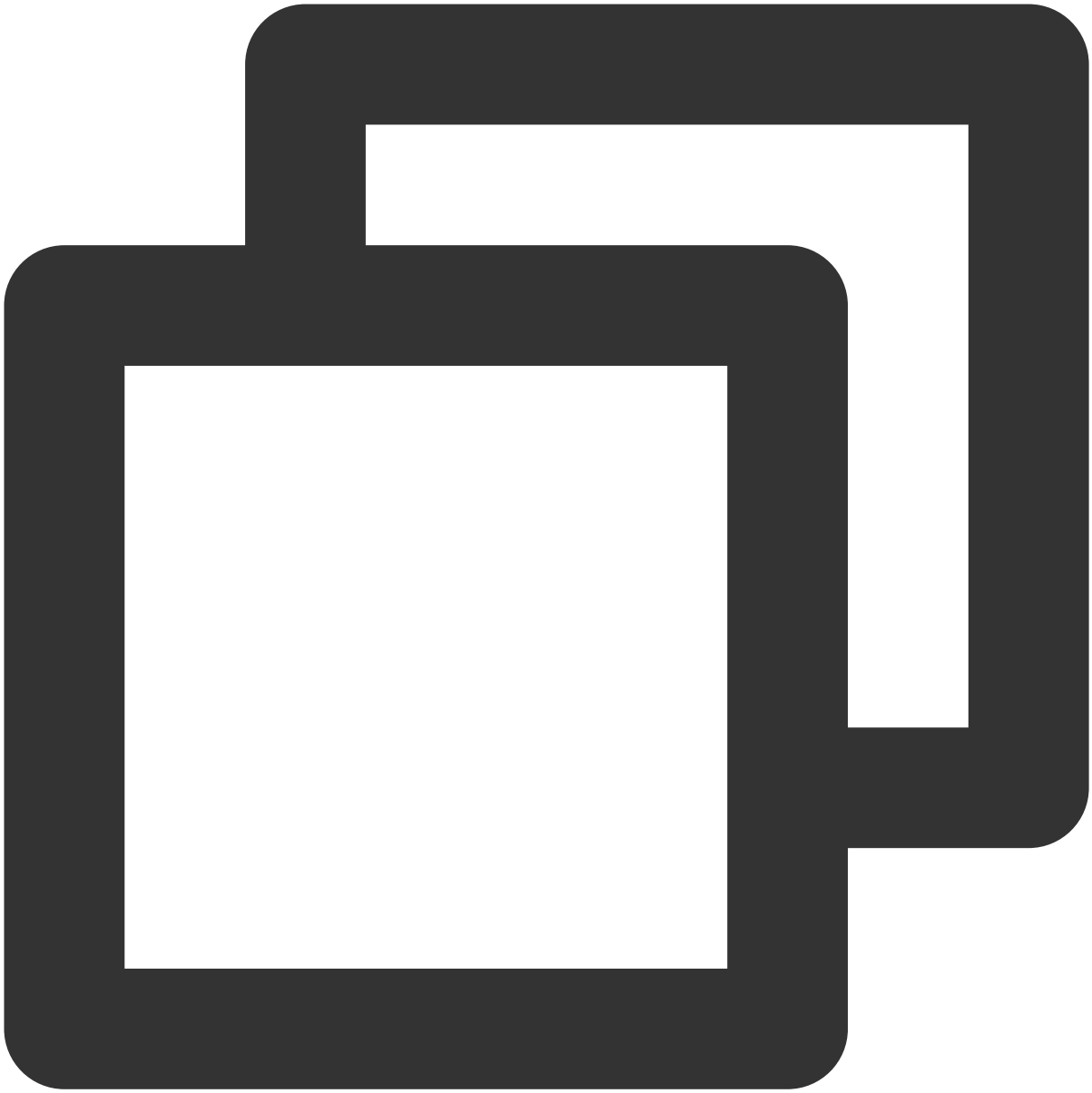
Upon expiration of the login status, developers can call wx.login again to obtain a new user login status. A successful call indicates that the current session_key has not expired, while a failed call indicates the opposite.

Parameter and Description: Object.

Attribute	Type	Default value	Required	Description
success	function	-	No	Callback Function of Successful Interface Call
fail	function	-	No	Callback Function of Failing Interface Call
complete	function	-	No	Callback function executed upon the completion of the interface invocation (both successful and unsuccessful invocations)

Sample Code

©2013-2022 Tencent Cloud. All rights reserved.



```
wx.checkSession({
  success() {
    // The session_key has not expired and remains valid throughout its lifecycle.
  },
  fail () {
    // The session_key has become invalid and the login process needs to be execute
    wx.login() // Log in again.
  }
})
```

©2013-2022 Tencent Cloud. All rights reserved.

Account information

getAccountInfoSync

This API is used via Object wx.getAccountInfoSync().

Feature Description: Gets the current account information. The online mini program version number can only be obtained in the official version of the mini program, and cannot be accessed in the development or trial version.

Return Value: Object, account information.

Attribute	Type	Description
miniProgram	Object	Mini program account information.
plugin	Object	Plugin account information (only included when called within a plugin)

miniProgram Structure Attributes

Structure attributes	Type	Description
appId	string	Mini program appId
envVersion	string	Mini program version, whose valid values are: develop: Development Version trial: Trial Version release: Official Version
version	string	Online Mini Program Version Number

Plugin Structure Attributes

Structure attributes	Type	Description
appId	string	Plugin AppId
version	string	Plugin Version No.

User information

©2013-2022 Tencent Cloud. All rights reserved.

getUserProfile

This API is used via wx.getUserProfile(Object object).

Feature Description: Acquires user information. This can only be invoked after a click event is generated on the page (for instance, within the bindtap callback on a button). Each request will prompt an authorization window, and upon user agreement, userInfo will be returned. This interface is intended to replace wx.getUserInfo. For more details, see Descriptions of User Information Interface Adjustment.

Return Value: Object.

Attribute	Type	Default value	Required	Description
lang	string	en	No	The language for displaying user information, whose valid values are: en: English zh_CN: Simplified Chinese zh_TW: Traditional Chinese
desc	string	-	Yes	Declare the purpose of obtaining user personal information, not exceeding 30 characters.
success	function	-	No	Callback Function of Successful Interface Call
fail	function	-	No	Callback Function of Failing Interface Call
complete	function	-	No	Callback function executed upon the completion of the interface invocation (both successful and unsuccessful invocations)

getUserInfo

This API is used via wx.getUserInfo(Object object).

Feature Description: Gets user information, which requires user authorization for scope.userInfo during its use.

Parameter and Description: Object.

Attribute	Type	Default value	Required	Description
withCredentials	boolean		No	Whether to carry login status information. When

©2013-2022 Tencent Cloud. All rights reserved.

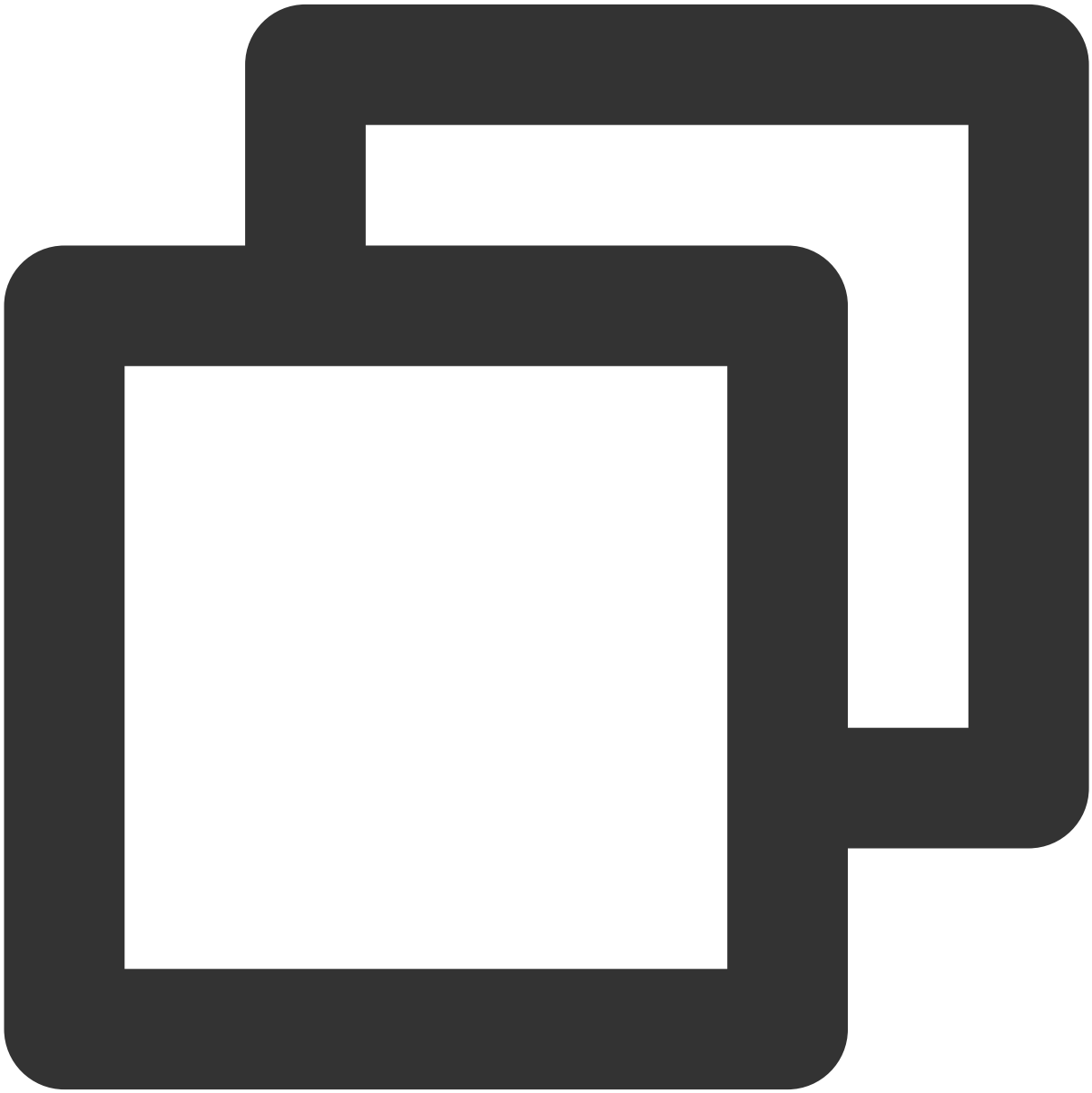
				withCredentials is true, it requires that wx.login has been called previously and the login status has not yet expired, at which point the returned data will include sensitive information such as encryptedData, iv. When withCredentials is false, there is no requirement for a login status, and the returned data does not include sensitive information such as encryptedData, iv.
lang	string	en	No	The language for displaying user information, whose valid values are: en: English zh_CN: Simplified Chinese zh_TW: Traditional Chinese
success	function	-	No	Callback Function of Successful Interface Call
fail	function	-	No	Callback Function of Failing Interface Call
complete	function	-	No	Callback function executed upon the completion of the interface invocation (both successful and unsuccessful invocations)

Parameters for object.success callback function: Object res

Attribute	Type	Description
userInfo	UserInfo	User information object, which does not include sensitive information such as openid.
rawData	string	The raw data string, devoid of sensitive information, is used for signature computation.
signature	string	Use sha1(rawData + sessionkey) to obtain a string, which is used for user information verification.
encryptedData	string	Encrypted data of complete user information, inclusive of sensitive data.
iv	string	Initial vector of the encryption algorithm.
cloudID	string	The cloud ID corresponding to sensitive data, returned only by mini programs that have activated cloud development. Open data can be directly obtained through cloud invocation. For more details, see direct acquisition of open data through cloud invocation.

Sample Code

©2013-2022 Tencent Cloud. All rights reserved.



```
// Must be invoked under the condition that the user has already granted authorizat
wx.getUserInfo({
  success: function(res) {
    var userInfo = res.userInfo
    var nickName = userInfo.nickName
    var avatarUrl = userInfo.avatarUrl
    var gender = userInfo.gender // Gender 0: Unknown; 1: Male; 2: Female
    var province = userInfo.province
    var city = userInfo.city
    var country = userInfo.country
  }
```

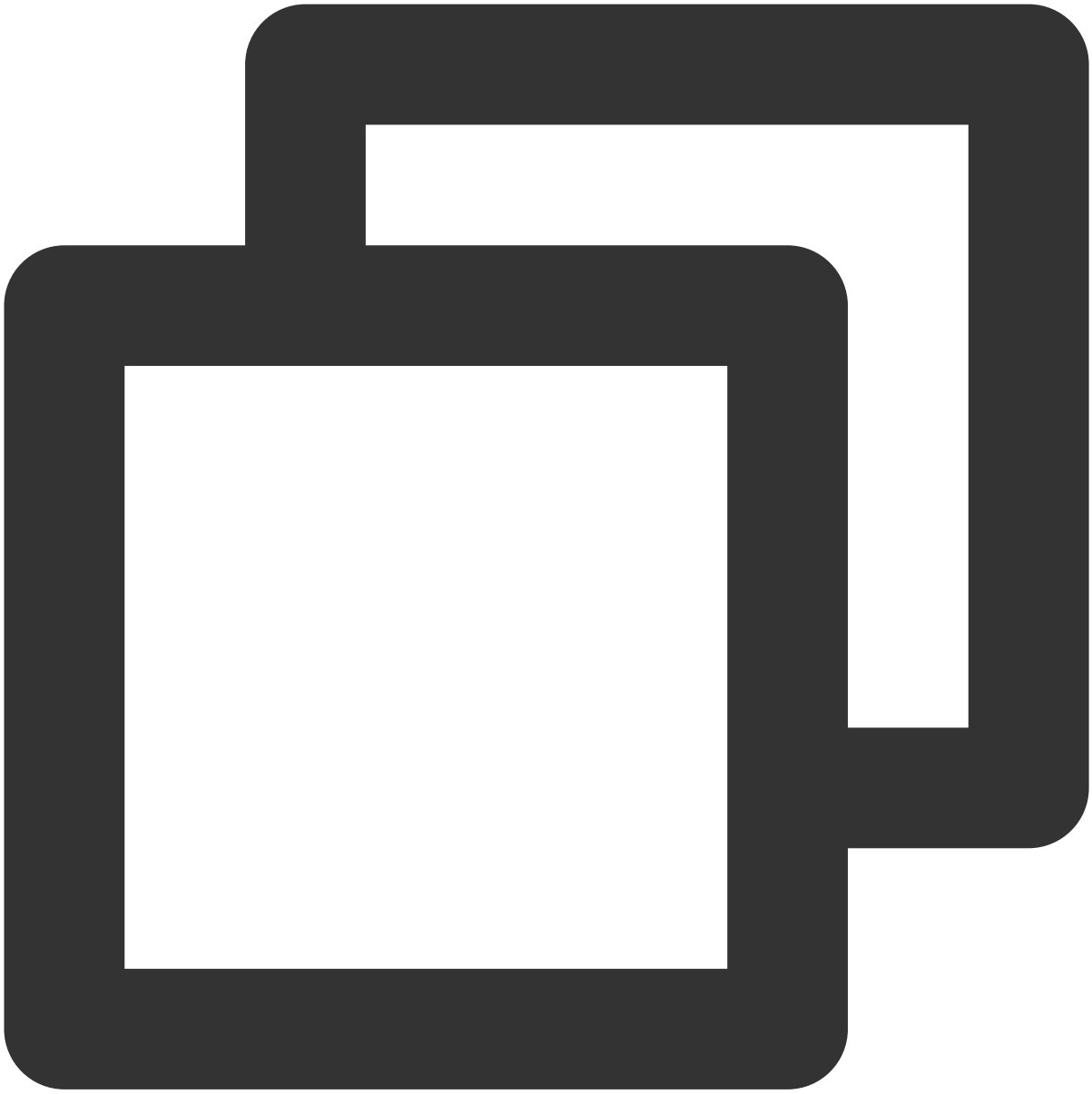
©2013-2022 Tencent Cloud. All rights reserved.

```
})
```

There are two methods to obtain sensitive data:

1. Use the decryption algorithm for encrypted data.

2. Use cloud invocation to directly obtain open data that is structured as follows in JSON format:

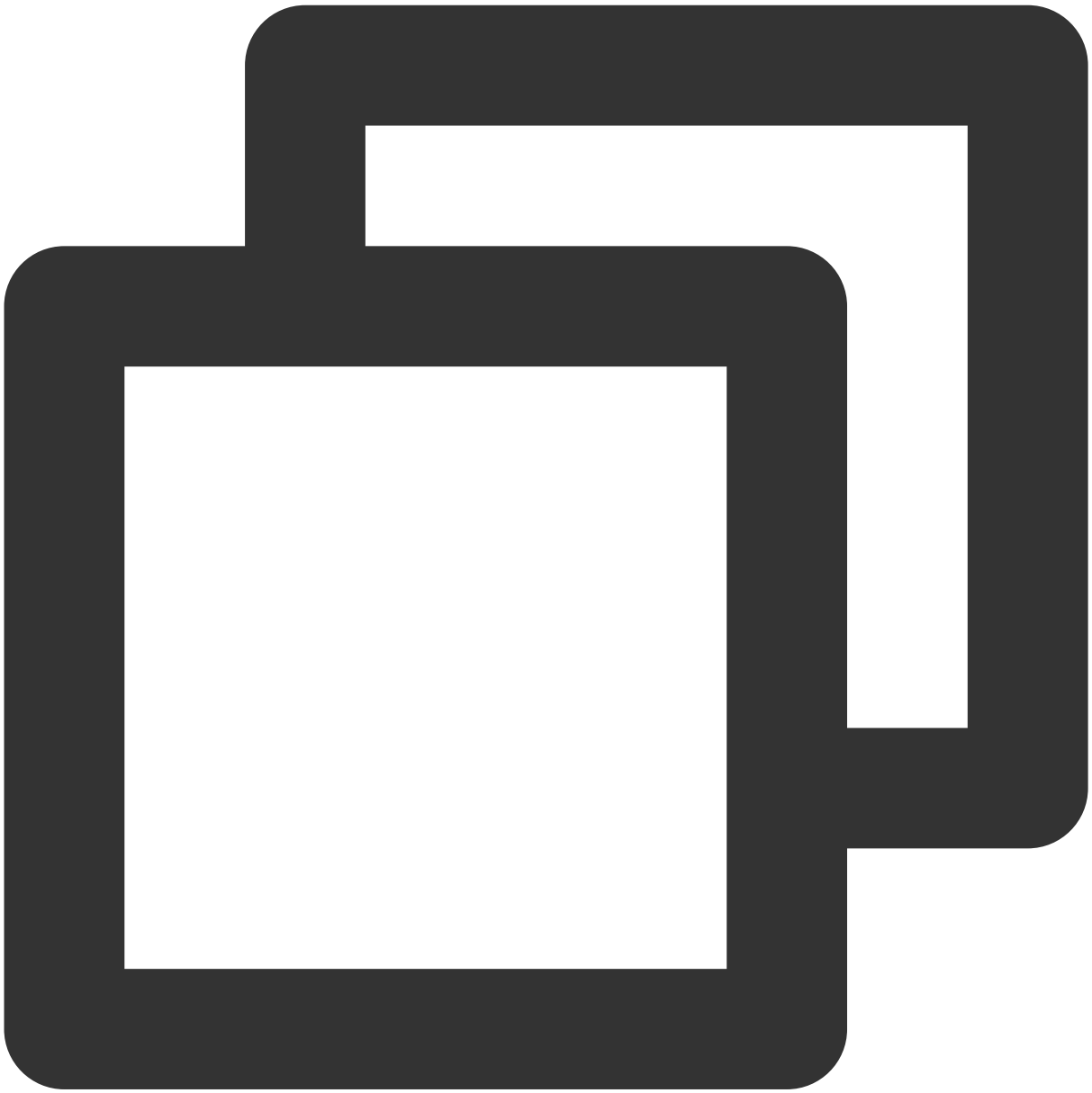


```
{
  "openId": "OPENID",
  "nickName": "NICKNAME",
  "gender": GENDER,
  "city": "CITY",
```

©2013-2022 Tencent Cloud. All rights reserved.

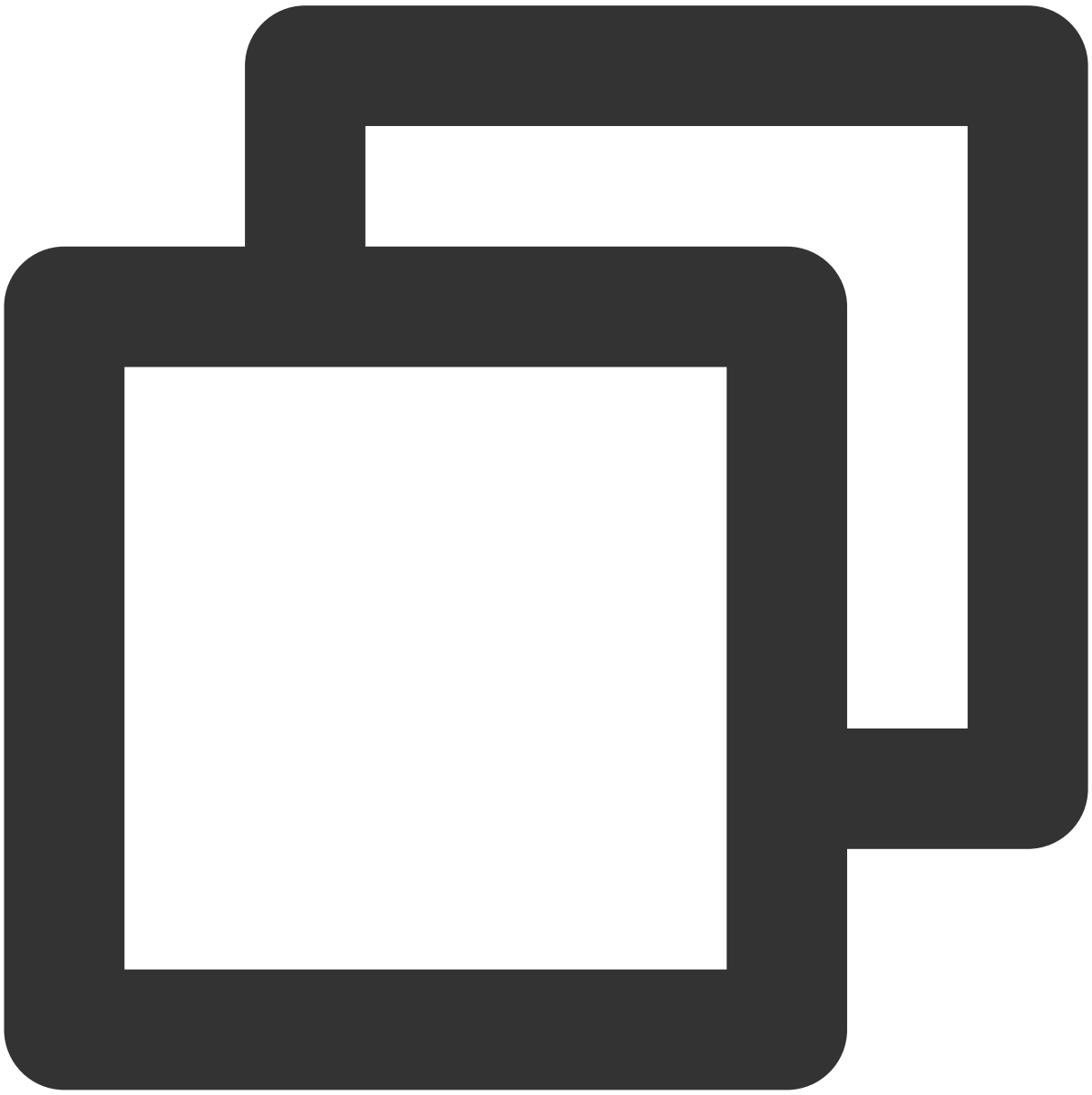
```
    "province": "PROVINCE",
    "country": "COUNTRY",
    "avatarUrl": "AVATARURL",
    "unionId": "UNIONID",
    "watermark": {
      "appid":"APPID",
      "timestamp":TIMESTAMP
    }
  }
```

Example code for the mini program user's information component



©2013-2022 Tencent Cloud. All rights reserved.

```
<!-- If only displaying user avatar and nickname, the <open-data /> component can b
<open-data type="userAvatarUrl"></open-data>
<open-data type="userNickName"></open-data>
<!-- The button must be used for login authorization -->
<button wx:if="{{canIUse}}" open-type="getUserInfo" bindgetuserinfo="bindGetUserInf
<view wx:else>Please upgrade the host client version.</view>
```



```
Page({
  data: {
```

©2013-2022 Tencent Cloud. All rights reserved.

```
    canIUse: wx.canIUse('button.open-type.getUserInfo')
  },
  onLoad: function() {
    // Check if authorization is granted.
    wx.getSetting({
      success(res) {
        if (res.authSetting['scope.userInfo']) {
          // Authorization has been granted and getUserInfo can be directly invoked
          wx.getUserInfo({
            success: function(res) {
              console.log(res.userInfo)
            }
          })
        }
      }
    })
  },
  bindGetUserInfo (e) {
    console.log(e.detail.userInfo)
  }
})
```

userInfo

Feature Description: User information.

Parameter and Description:

Attribute	Type	Description
nickName	string	User Nickname
avatarUrl	string	The URL of the user's avatar image. The last digit in the URL represents the size of the square avatar (options include 0, 46, 64, 96, 132; 0 represents a 640x640 square avatar, 46 represents a 46x46 square avatar, and the remaining values follow this pattern. The default selection is 132). This field is empty if the user does not have an avatar. If the user changes their avatar, the original avatar URL will become invalid.
gender	number	User gender, no longer returned. Valid values are: 0: Unknown 1: Male 2: Female
country	string	The country where the user is located. No longer returned.
province	string	The province where the user is located. No longer returned.

©2013-2022 Tencent Cloud. All rights reserved.

city	string	The city where the user is located. No longer returned.
language	string	The language used to display country, province, city. "zh_CN" is forcibly returned. Valid values are: en: English zh_CN: Simplified Chinese zh_TW: Traditional Chinese

Settings

AuthSetting

Feature Description: User authorization settings information.

Parameter and Description:

Attribute	Description
boolean scope.userLocation	Whether to authorize geographical location, corresponding to the interface wx.getLocation Whether to authorize geographical location, corresponding to the interface wx.chooseLocation
boolean scope.writePhotosAlbum	Whether to authorize saving to the album wx.saveImageToPhotosAlbum
boolean scope.camera	Whether to authorize the camera, corresponding to the <camera /> component.
boolean scope.addFriend	Allow to be added as a friend, proactively call the wx.authorize interface for authorization.

getSetting

This API is used via wx.getSetting(Object object).

Feature Description: Gets the current settings of the user.

Parameter and Description: Object.

Attribute	Type	Default value	Required	Description
withSubscriptions	Boolean	false	No	Whether to simultaneously obtain the subscription status of the user's subscription messages, with the

©2013-2022 Tencent Cloud. All rights reserved.

				default option being "no".
success	function	-	No	Callback Function of Successful Interface Call
fail	function	-	No	Callback Function of Failing Interface Call
complete	function	-	No	Callback function executed upon the completion of the interface invocation (both successful and unsuccessful invocations)

Note:

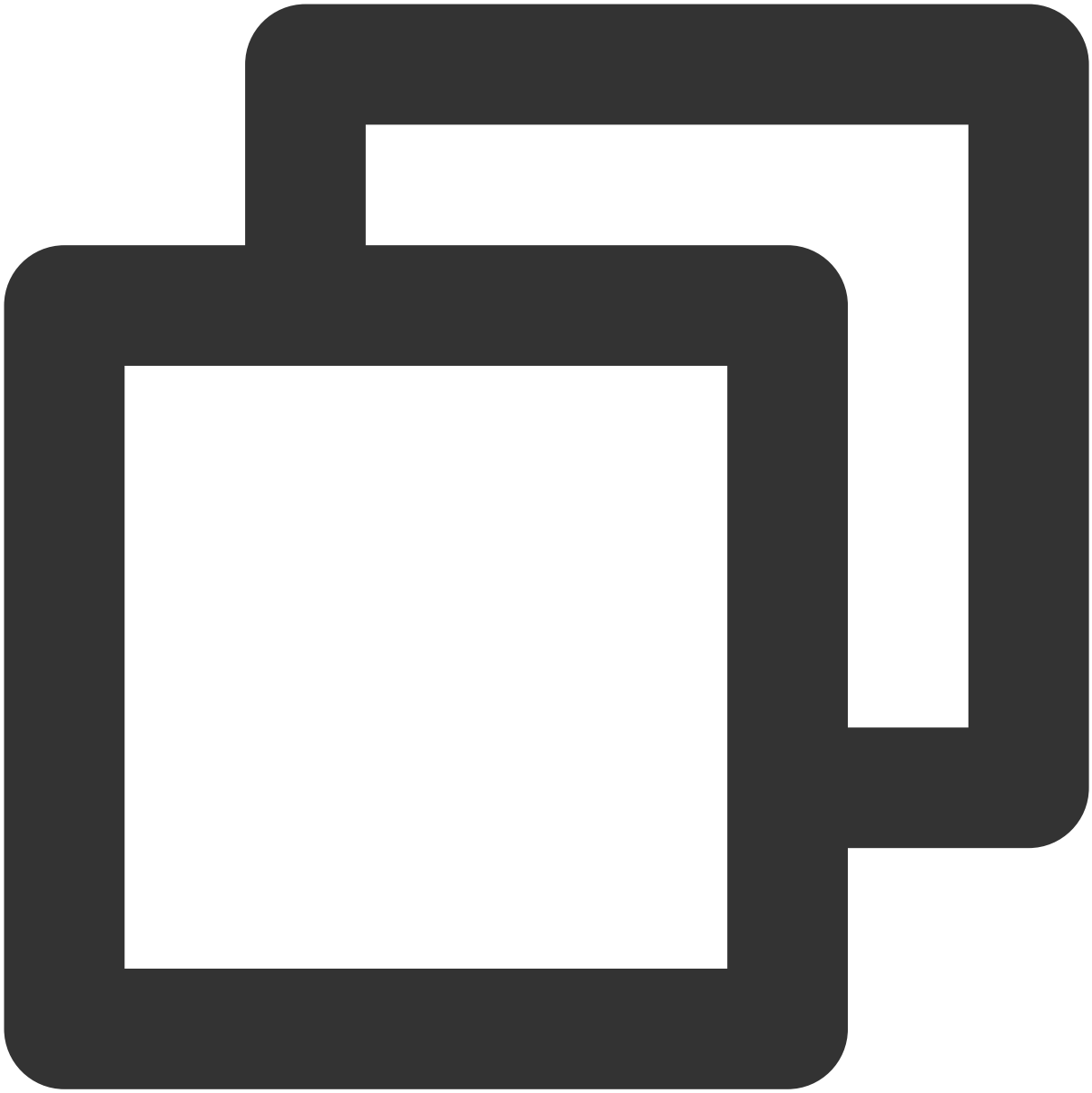
The "withSubscriptions" only returns the subscription messages that the user has checked in the subscription panel with the option "Always keep the above selection. Do not ask again".

Parameters for object.success callback function: Object res.

Attribute	Type	Description
authSetting	AuthSetting	User Authorization Results

Sample Code

©2013-2022 Tencent Cloud. All rights reserved.



```
wx.getSetting({
  success(res) {
    console.log(res.authSetting)
    // res.authSetting = {
    //   "scope.userInfo": true,
    //   "scope.userLocation": true
    // }
  }
})
```

©2013-2022 Tencent Cloud. All rights reserved.

openSetting

This API is used via wx.openSetting(Object object).

Feature Description: This triggers the client's mini program settings interface, returning the user's operation results. After the user clicks, they can redirect to open the settings page, where they can manage authorization information.

Parameter and Description: Object.

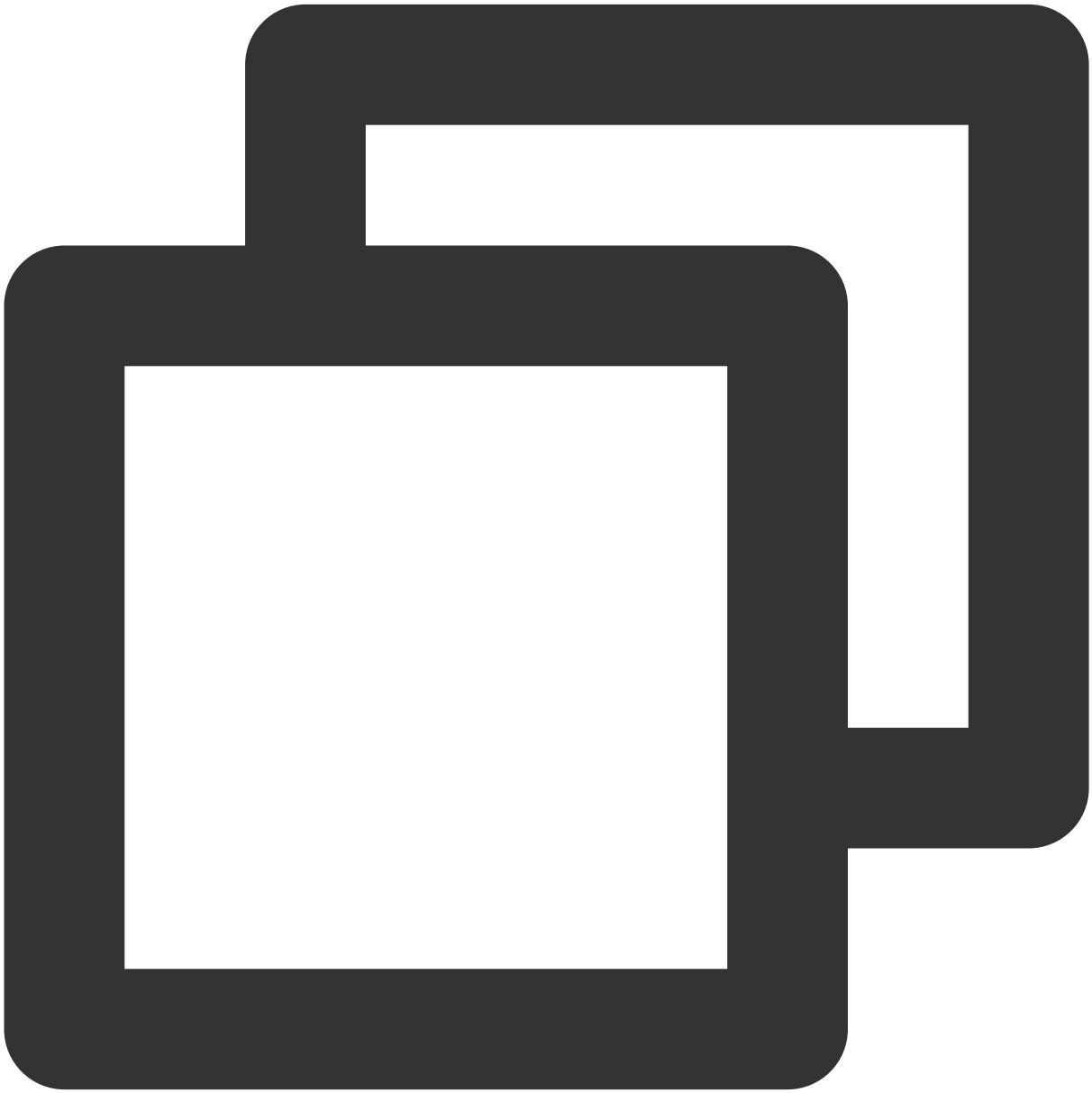
Attribute	Type	Default value	Required	Description
success	function	-	No	Callback Function of Successful Interface Call
fail	function	-	No	Callback Function of Failing Interface Call
complete	function	-	No	Callback function executed upon the completion of the interface invocation (both successful and unsuccessful invocations)

Parameters for object.success callback function: Object res.

Attribute	Type	Description
authSetting	AuthSetting	User Authorization Results

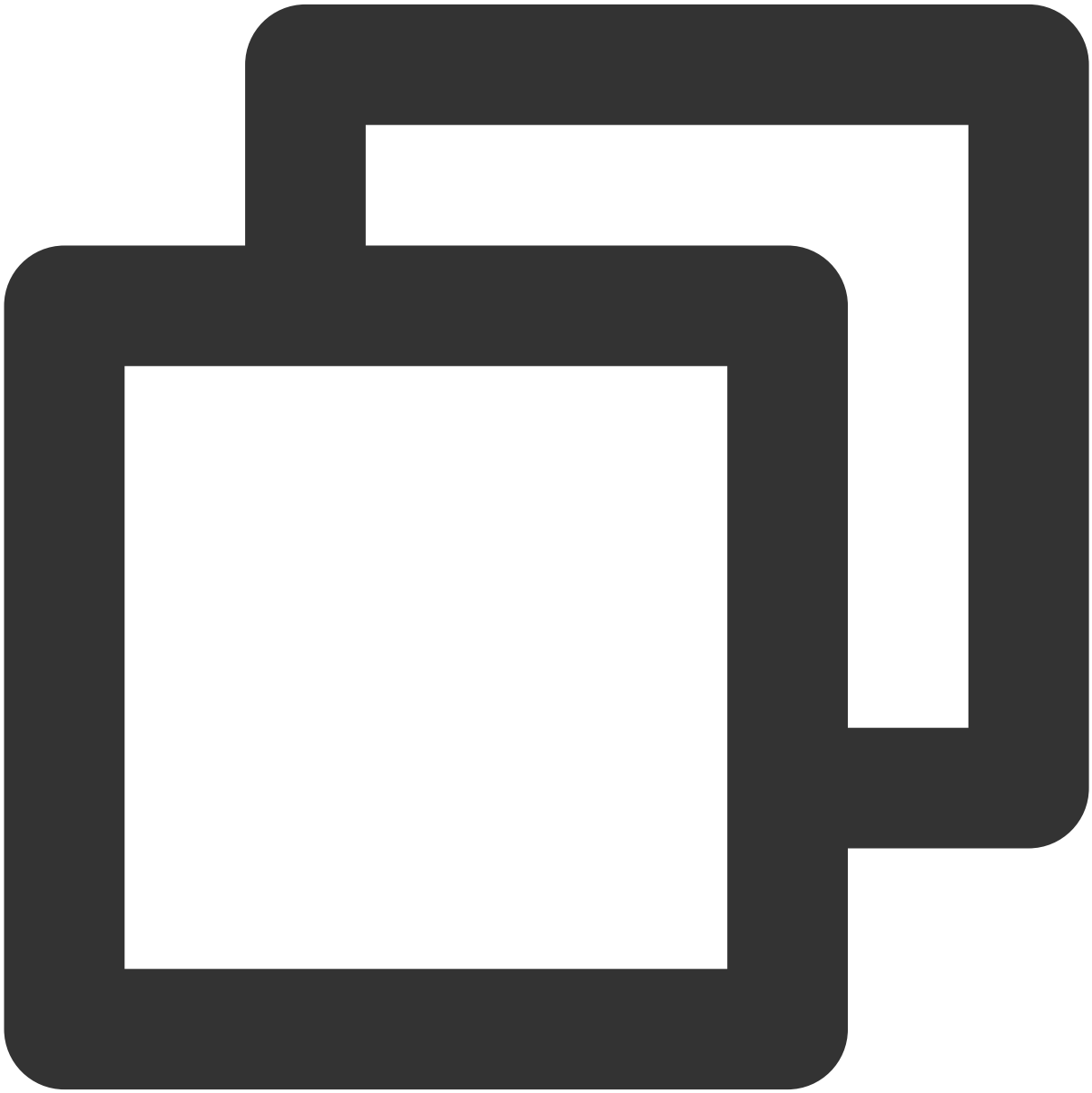
Sample Code

©2013-2022 Tencent Cloud. All rights reserved.



```
wx.openSetting({
  success(res) {
    console.log(res.authSetting)
    // res.authSetting = {
    //   "scope.userInfo": true,
    //   "scope.userLocation": true
    // }
  }
})
```

©2013-2022 Tencent Cloud. All rights reserved.



```
wx.openSetting({
  success(res) {
    console.log(res.authSetting)
    // res.authSetting = {
    //   "scope.userInfo": true,
    //   "scope.userLocation": true
    // }
  }
})
```

©2013-2022 Tencent Cloud. All rights reserved.

Biometric Authentication

checkIsSoterEnrolledInDevice

This API is used via wx.checkIsSoterEnrolledInDevice(Object object).

Feature Description: Verifies whether biometric information has been entered into the device.

Parameter and Description: Object.

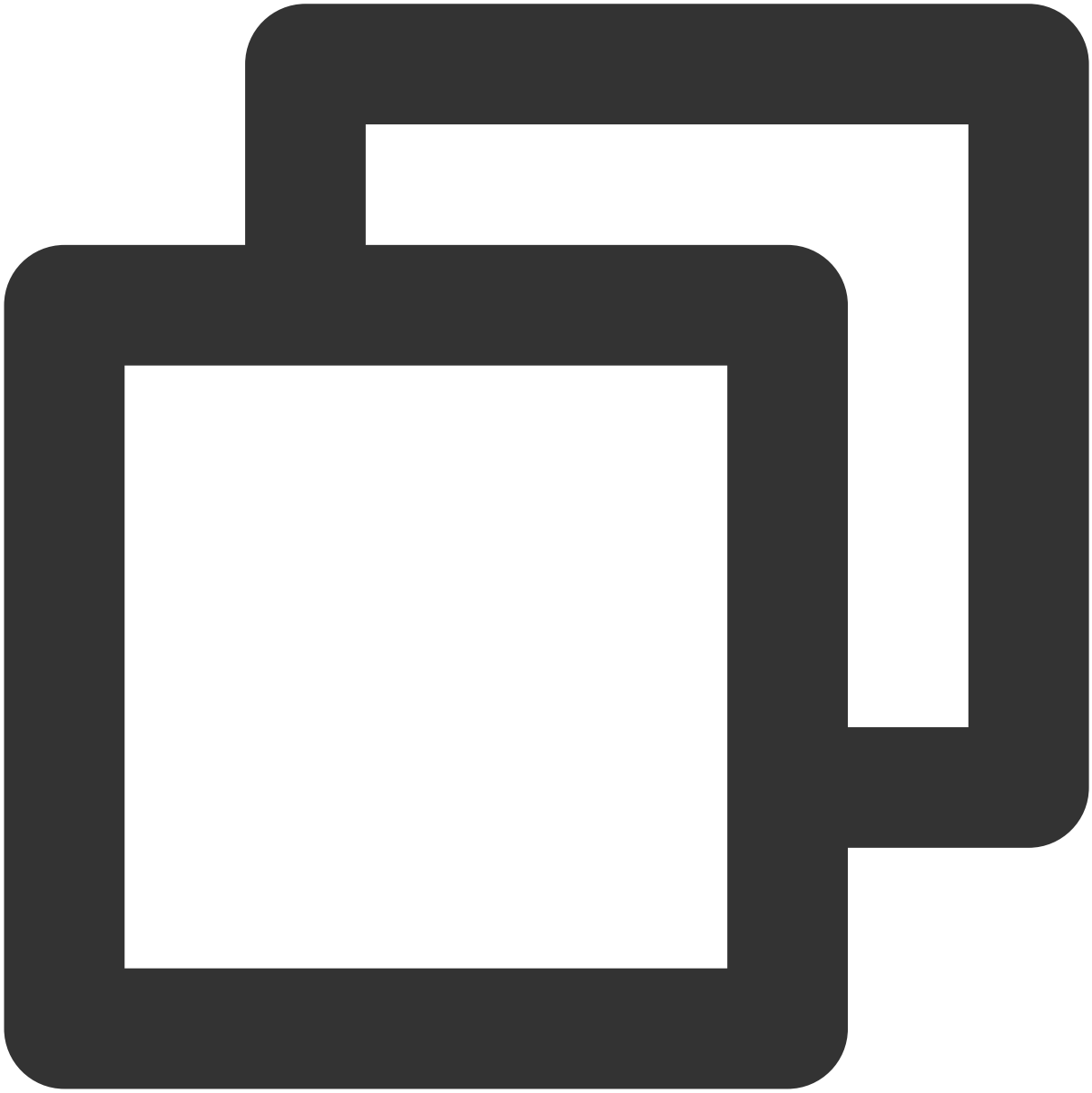
Attribute	Type	Default value	Required	Description
success	function	-	No	Callback Function of Successful Interface Call
fail	function	-	No	Callback Function of Failing Interface Call
complete	function	-	No	Callback function executed upon the completion of the interface invocation (both successful and unsuccessful invocations)

Parameters for object.success callback function: Object res.

Attribute	Type	Description
isEnrolled	boolean	Whether information has been entered.
errMsg	string	Error Message

Sample Code

©2013-2022 Tencent Cloud. All rights reserved.



```
wx.checkIsSoterEnrolledInDevice({
  success(res) {
    console.log(res.isEnrolled)
  }
})
```

checkIsSupportSoterAuthentication

This method is used via wx.checkIsSupportSoterAuthentication(Object object).

©2013-2022 Tencent Cloud. All rights reserved.

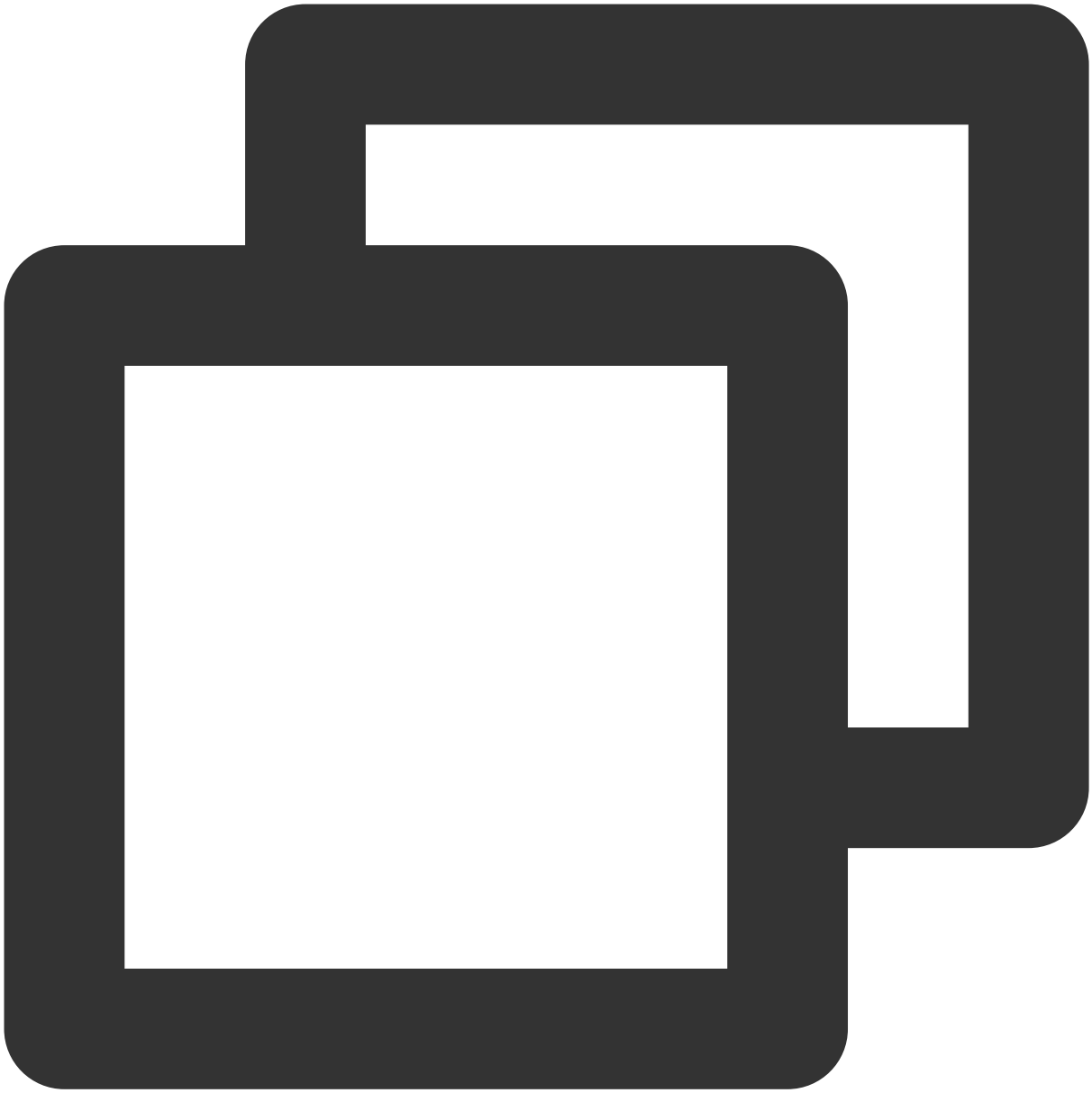
Feature Description: Verifies whether the device supports biometric authentication. If yes, it triggers a success callback; if not, it triggers a fail callback.

Parameter and Description: Object.

Attribute	Type	Default value	Required	Description
success	function	-	No	Callback Function of Successful Interface Call
fail	function	-	No	Callback Function of Failing Interface Call
complete	function	-	No	Callback function executed upon the completion of the interface invocation (both successful and unsuccessful invocations)

Sample Code

©2013-2022 Tencent Cloud. All rights reserved.



```
wx.checkIsSupportSoterAuthentication({
  success() {
    // Supports biometric authentication.
   }
})
```

startSoterAuthentication

This method is used via wx.startSoterAuthentication(Object object).

©2013-2022 Tencent Cloud. All rights reserved.

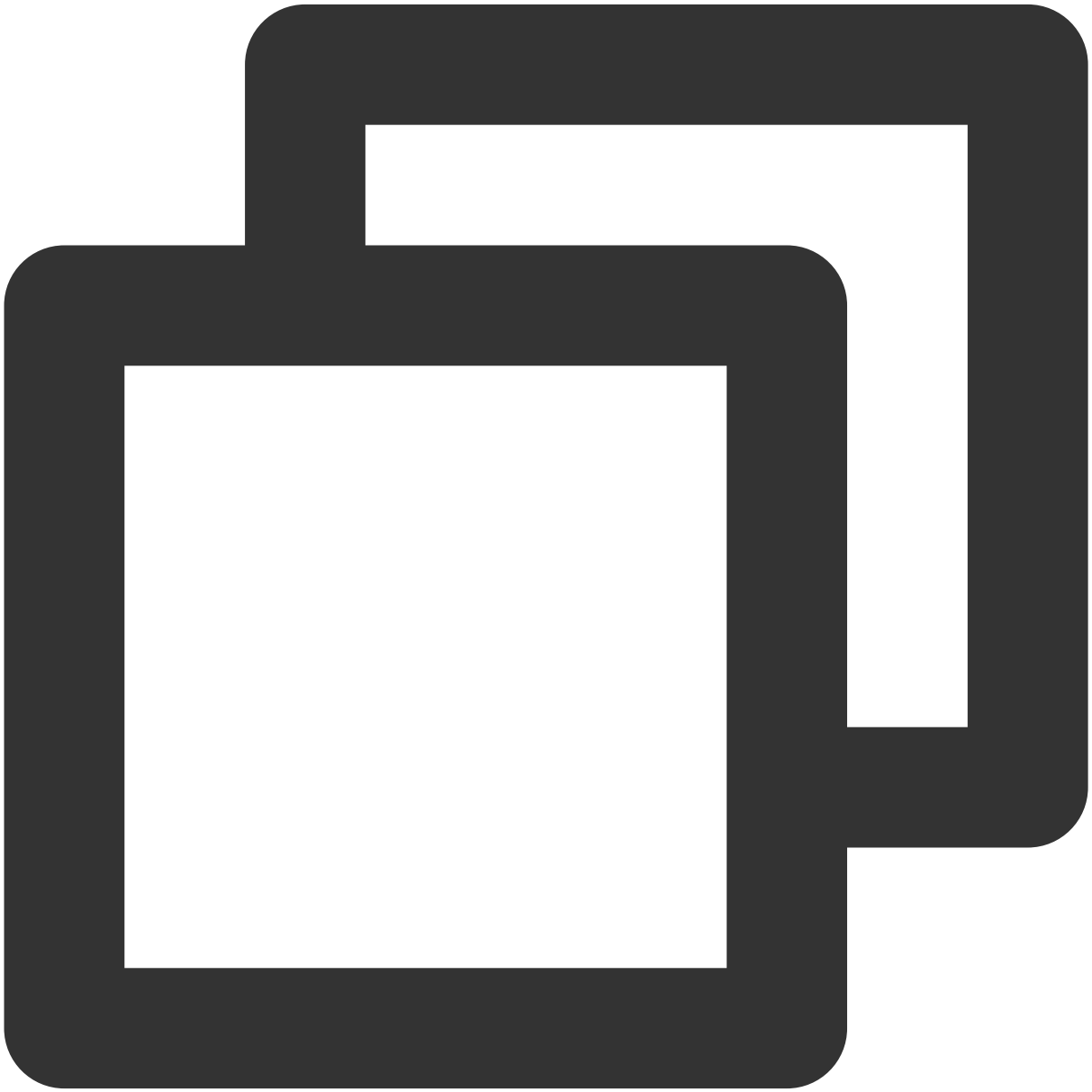
Feature Description: Initiates biometric authentication.

Parameter and Description: Object.

Attribute	Type	Default value	Required	Description
authContent	string	''	No	This is the verification description, meaning the dialog box prompt content will be displayed on the interface during the identification process.
success	function	-	No	Callback Function of Successful Interface Call
fail	function	-	No	Callback Function of Failing Interface Call
complete	function	-	No	Callback function executed upon the completion of the interface invocation (both successful and unsuccessful invocations)

Sample Code

©2013-2022 Tencent Cloud. All rights reserved.



```
wx.startSoterAuthentication({
   authContent: 'Unlock via Biometric Authentication',
   success() {
     // Authentication successful
   }
})
```

Authorization

©2013-2022 Tencent Cloud. All rights reserved.

Authorization

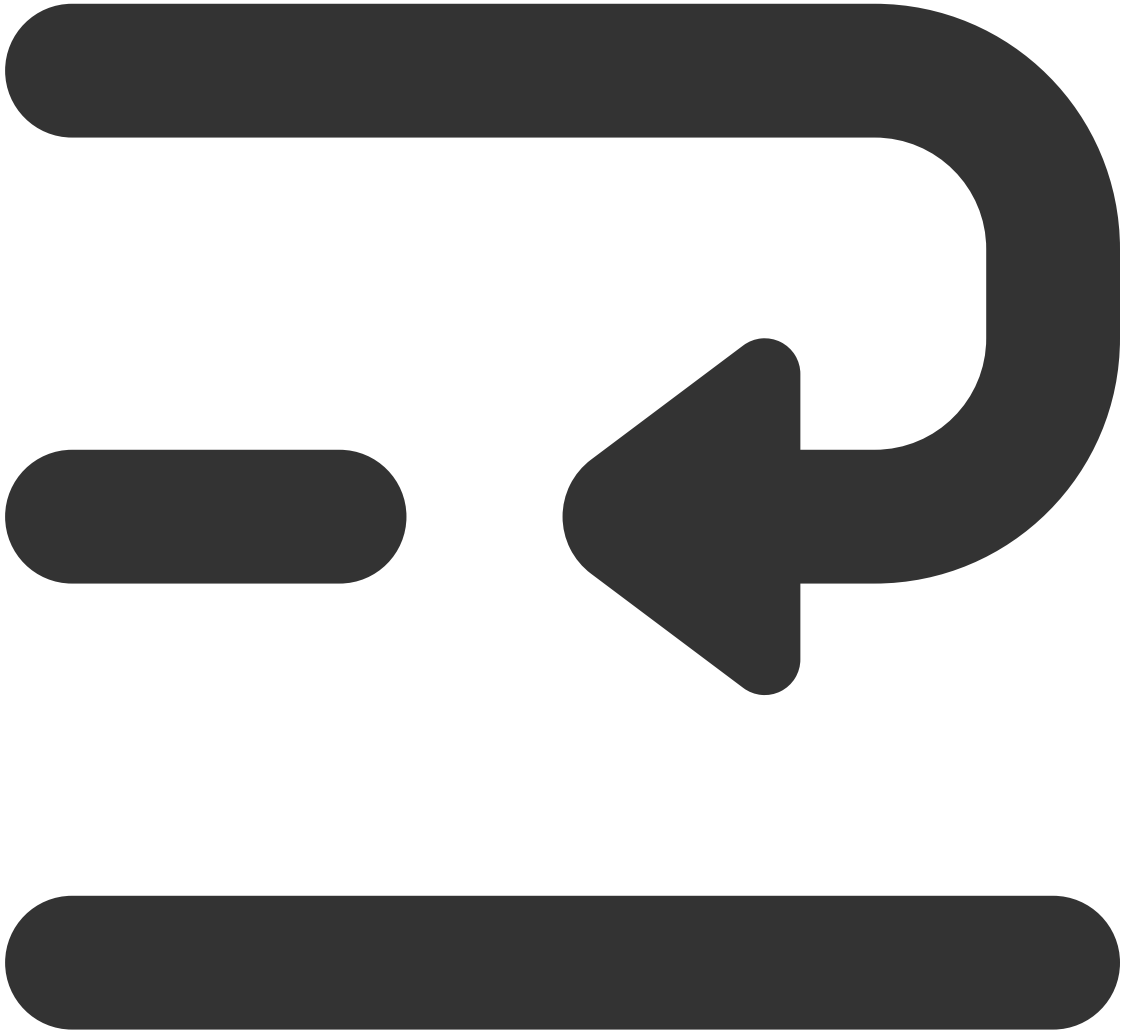
This API is used via wx.authorize(Object object).

Feature Description: Initiates an authorization request to the user in advance. Upon invocation, a pop-up window will immediately appear, asking the user whether they agree to authorize the mini program to use a certain function or to access certain user data. However, it will not actually call the corresponding interface. If the user has previously agreed to the authorization, the pop-up window will not appear and it will directly return success.
Parameter and Description: Object.

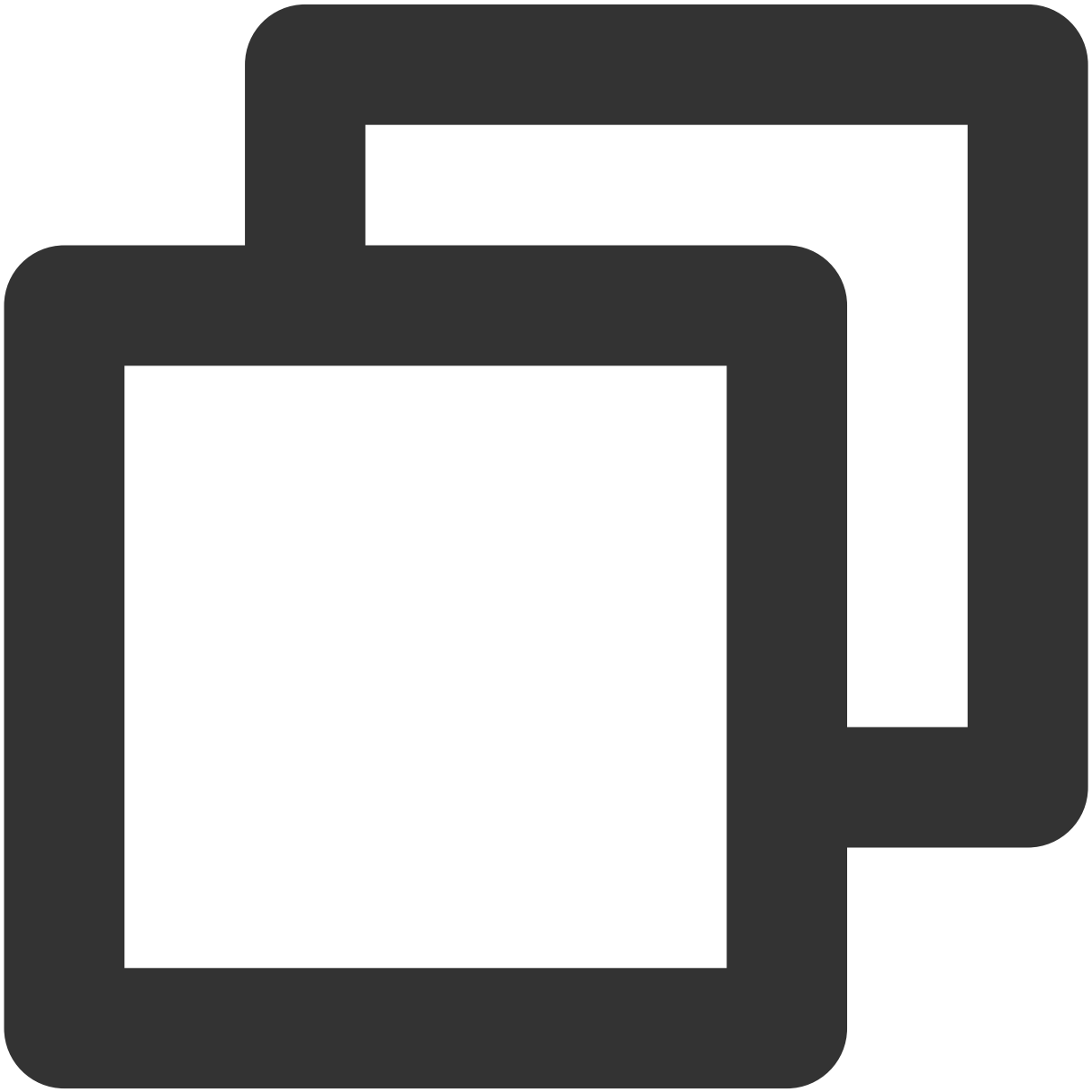
Attribute	Type	Default value	Required	Description
scope	string	-	Yes	Scope of permissions required
success	function	-	No	Callback Function of Successful Interface Call
fail	function	-	No	Callback Function of Failing Interface Call
complete	function	-	No	Callback function executed upon the completion of the interface invocation (both successful and unsuccessful invocations)

Sample Code

©2013-2022 Tencent Cloud. All rights reserved.



©2013-2022 Tencent Cloud. All rights reserved.



```
// You can first query whether the user has authorized the "scope.record" scope usi
wx.getSetting({
  success(res) {
    if (!res.authSetting['scope.record']) {
      wx.authorize({
        scope: 'scope.record',
        success() {

          // The user has already agreed to the mini program's use of the recording
          wx.startRecord()
        }
```

©2013-2022 Tencent Cloud. All rights reserved.

```
          })
        }
    }
})
```

©2013-2022 Tencent Cloud. All rights reserved.

Custom APIs

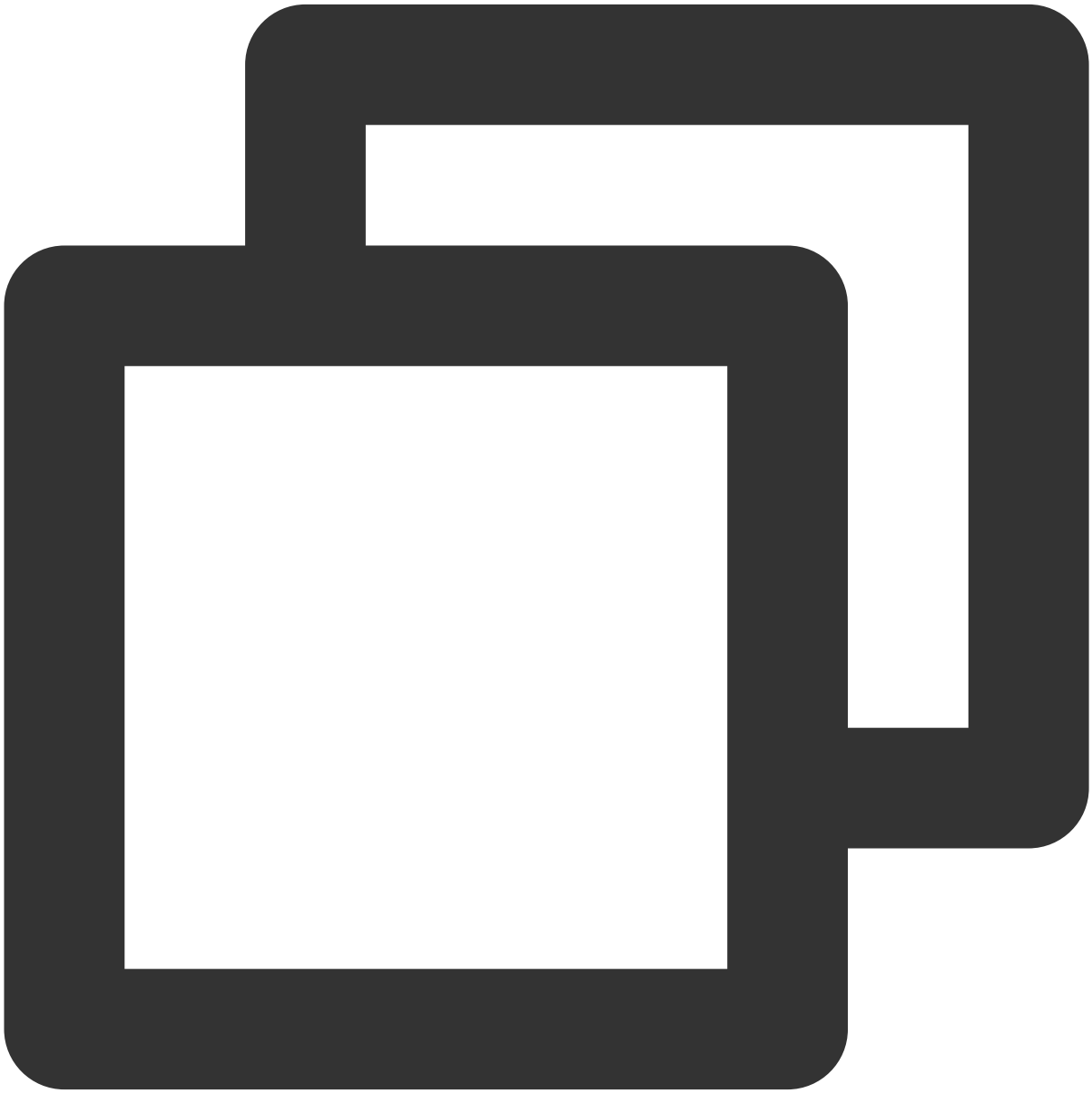
Last updated：2024-07-12 18:02:09

invokeNativePlugin

The TCMPP Mini Program SDK offers interfaces that enable the implementation of certain Open Platform APIs, which can be directly transmitted to the host client, such as login and user information retrieval. Some host clients can be expanded to maintain UI and functionality consistency, such as scanning and sharing. Additionally, it is possible to extend custom APIs for direct transmission to the host client.

Usage method at mini program end

©2013-2022 Tencent Cloud. All rights reserved.



```
var opts = {
  api_name: '', // Name of the API
  success: function(res) {},
  fail: function(res) {},
  complete: function(res) {},
  data: { // Input Parameters
     name : 'kka',
     age : 22,
     data: {...}
  }
}
```

©2013-2022 Tencent Cloud. All rights reserved.

```
wx.invokeNativePlugin(opts); // Invocation
```

Methods

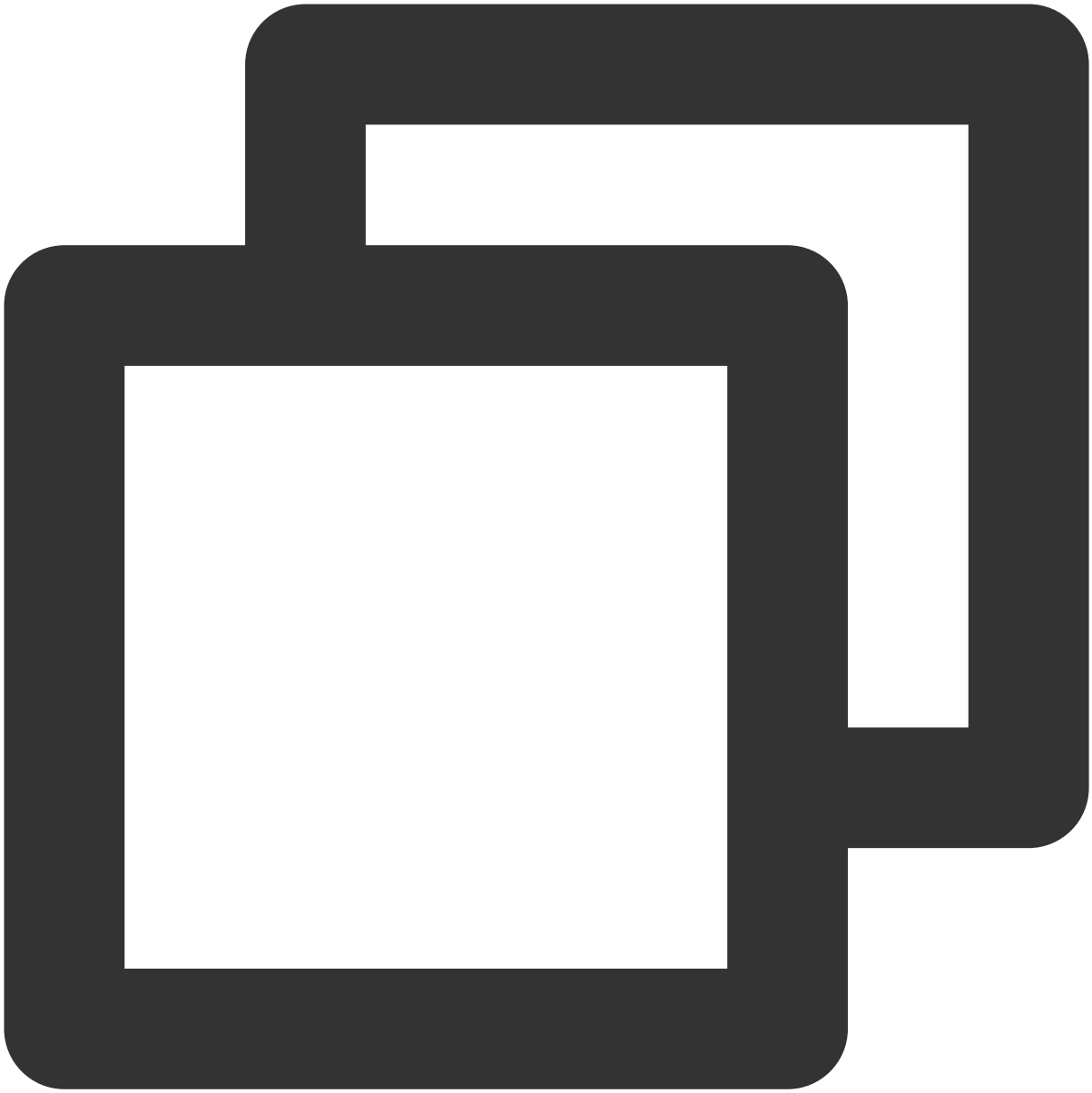
Method	Description
View onCreateLoadingView()	Create a loading view, permitting the host client to render mini program/mini game brand and custom display styles. If returned as null, the default loading view will be exhibited.
View onCreateCapsuleView()	Establish a capsule button, allowing the host to customize the capsule button style. If returned as null, the default capsule button will be displayed.
boolean onShowMenu()	Triggered when the "..." more button is clicked. If onCreateCapsuleView returns null and onShowMenu returns false, the default menu will be displayed.
void onExit()	Activated when the "◯" exit button is clicked. If onCreateCapsuleView returns null and onExit returns false, the default operation will be executed.
boolean onAuthorize(JSONObject params, Value Callback callback)	Implementation of the wx.authorize interface. If it returns false, the notification of wx api call failure will be directly sent.
boolean onOpenSetting(JSONObject params, ValueCallback callback)	Implementation of the wx.openSetting interface. If it returns false, the notification of wx api call failure will be directly sent.
boolean onGetSetting(JSONObject params, ValueCallback callback)	Implementation of the wx.getSetting interface. If it returns false, the notification of wx api call failure will be directly sent.
boolean onLogin(JSONObject params, ValueCallback callback)	Implementation of the wx.login interface. If it returns false, the notification of wx api call failure will be directly sent.
boolean onRefreshSession(JSONObject params, ValueCallback callback)	Implementation of the wx.checkSession interface. If it returns false, the notification of wx api call failure will be directly sent.
boolean onRequestPayment(JSONObject params, ValueCallback callback)	Implementation of the wx.requestPayment interface. If it returns false, the notification of wx api call failure will be directly sent.
boolean onGetUserInfo(JSONObject params, ValueCallback callback)	Implementation of the wx.getUserInfo interface. If it returns false, the notification of wx api call failure will be directly sent. For more details, see onGetUserInfo.
boolean onShareAppMessage(JSONObject params, ValueCallback callback)	Implementation of the wx.shareAppMessage interface. If it returns false, the notification of wx api call failure will be directly sent.

©2013-2022 Tencent Cloud. All rights reserved.

boolean onNavigateToMiniProgram(JSONObject params, ValueCallback callback)	Implementation of the wx.navigateToMiniProgram interface. If it returns false, the notification of wx api call failure will be directly sent.
boolean onScanCode(JSONObject params, ValueCallback callback)	Implementation of the wx.scanCode interface. If it returns false, the notification of wx api call failure will be directly sent.
boolean onOpenDocument(JSONObject params, ValueCallback callback)	Implementation of the wx.openDocument interface. If it returns false, the notification of wx api call failure will be directly sent.
boolean onOpenLocation(JSONObject params, ValueCallback callback)	Implementation of the wx.openLocation interface. If it returns false, the notification of wx api call failure will be directly sent.
boolean onChooseLocation(JSONObject params, ValueCallback callback)	Implementation of the wx.chooseLocation interface. If it returns false, the notification of wx api call failure will be directly sent.
boolean onPreviewImage(JSONObject params, ValueCallback callback)	Implementation of the wx.rviewImage interface. If it returns false, the notification of wx api call failure will be directly sent.
boolean onChooseImage(JSONObject params, ValueCallback callback)	Implementation of the wx.chooseImage interface. If it returns false, the notification of wx api call failure will be directly sent.
boolean onChooseVideo(JSONObject params, ValueCallback callback)	Implementation of the wx.chooseVideo interface. If it returns false, the notification of wx api call failure will be directly sent.
boolean onShowToast(JSONObject params, ValueCallback callback)	Implementation of the wx.showToast and wx.showLoading interfaces. If it returns false, the default style will be displayed.
boolean onHideToast(JSONObject params, ValueCallback callback)	Implementation of the wx.hideToast and wx.hideLoading interfaces. If it returns false, the default style will be displayed.
boolean onShowModal(JSONObject params, ValueCallback callback)	Implementation of the wx.showModal interface. If it returns false, the default style will be displayed.
boolean onInvokeWebAPI(String event, JSONObject params, ValueCallback callback)	Implementation of the custom wx api. If it returns false, the notification of wx api call failure will be directly sent. For more details, see onInvokeWebAPI.
boolean onReportEvent(String event, Map params)	For more details, see onReportEvent.

boolean onGetUserInfo(JSONObject params, ValueCallback callback)

©2013-2022 Tencent Cloud. All rights reserved.



```
-(void)onGetUserInfoWithParams:(NSDictionary *)params inApp:(NSString *)appId callb
    //TODO
        NSDictionary* userInfo = @{
                                   @"nickname" : @"morven",
                                   @"headimgurl" : @"https://wx.qlogo.cn/mmopen/vi_
                                   @"sex" : @(1),
                                   @"province" : @"Guangdong",
                                   @"city" : @"Shenzhen",
                                   @"country" : @"China"
                                   };
        NSMutableDictionary* result = [NSMutableDictionary dictionaryWithCapacity:1
```

©2013-2022 Tencent Cloud. All rights reserved.

```
        NSMutableDictionary* userInfoDic = [NSMutableDictionary dictionaryWithCapac
        NSMutableDictionary* resultDataDic = [NSMutableDictionary dictionaryWithCap
        [userInfoDic setValue:userInfo[@"nickname"] forKey:@"nickName"];
        [userInfoDic setValue:userInfo[@"headimgurl"] forKey:@"avatarUrl"];
        [userInfoDic setValue:userInfo[@"sex"] forKey:@"gender"];
        [userInfoDic setValue:userInfo[@"province"] forKey:@"province"];
        [userInfoDic setValue:userInfo[@"city"] forKey:@"city"];
        [userInfoDic setValue:userInfo[@"country"] forKey:@"country"];

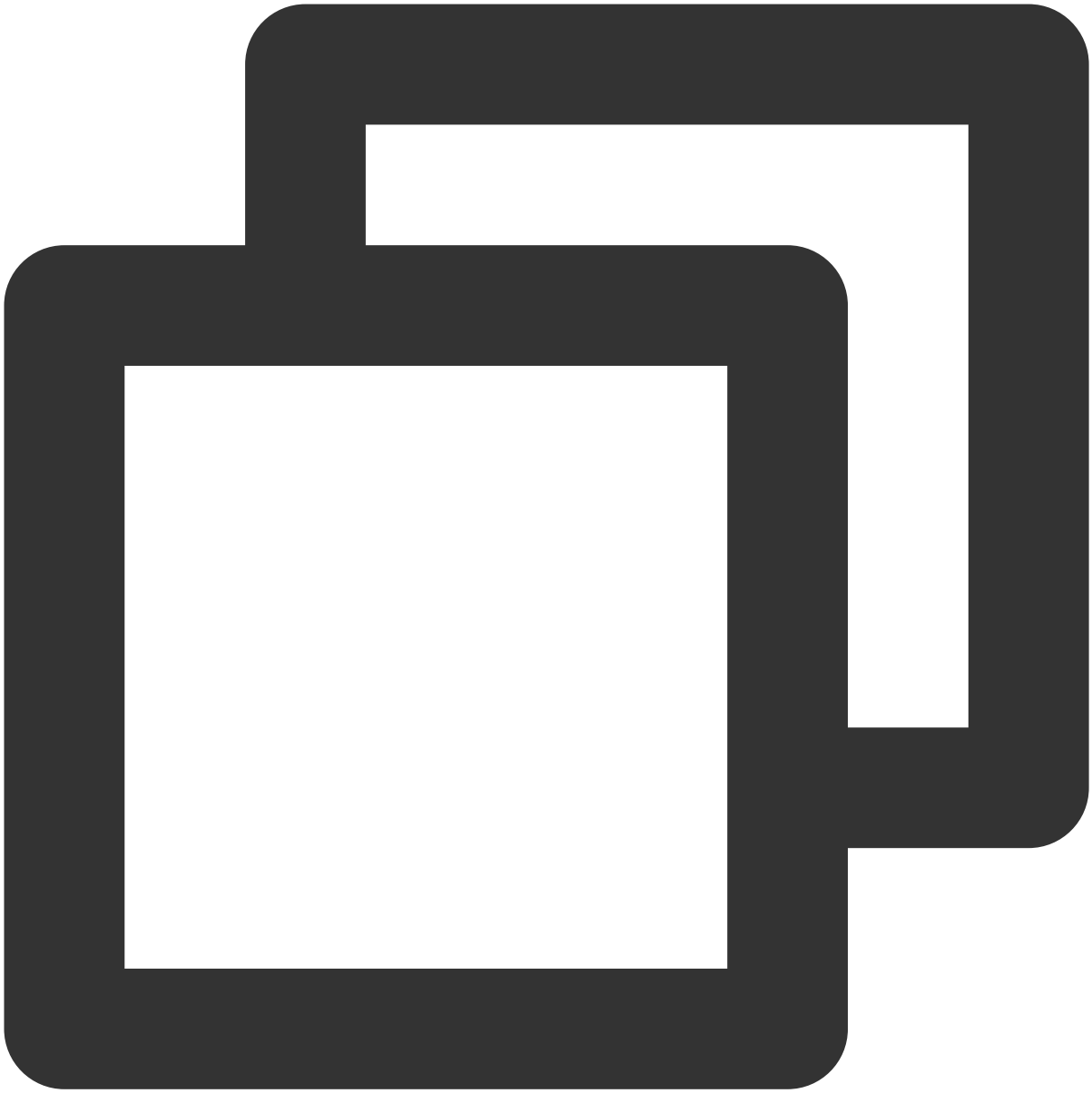
        NSData *data=[NSJSONSerialization dataWithJSONObject:userInfoDic options:NS

        [resultDataDic setValue:[[NSString alloc]initWithData:data encoding:NSUTF8S
        [result setValue:resultDataDic forKey:@"data"];
        [result setValue:@"getUserInfo:ok" forKey:@"errMsg"];

    if (handler) {
        //success
        handler(result, nil);
    }
}
```

boolean onInvokeWebAPI(String event, JSONObject params, ValueCallback callback)

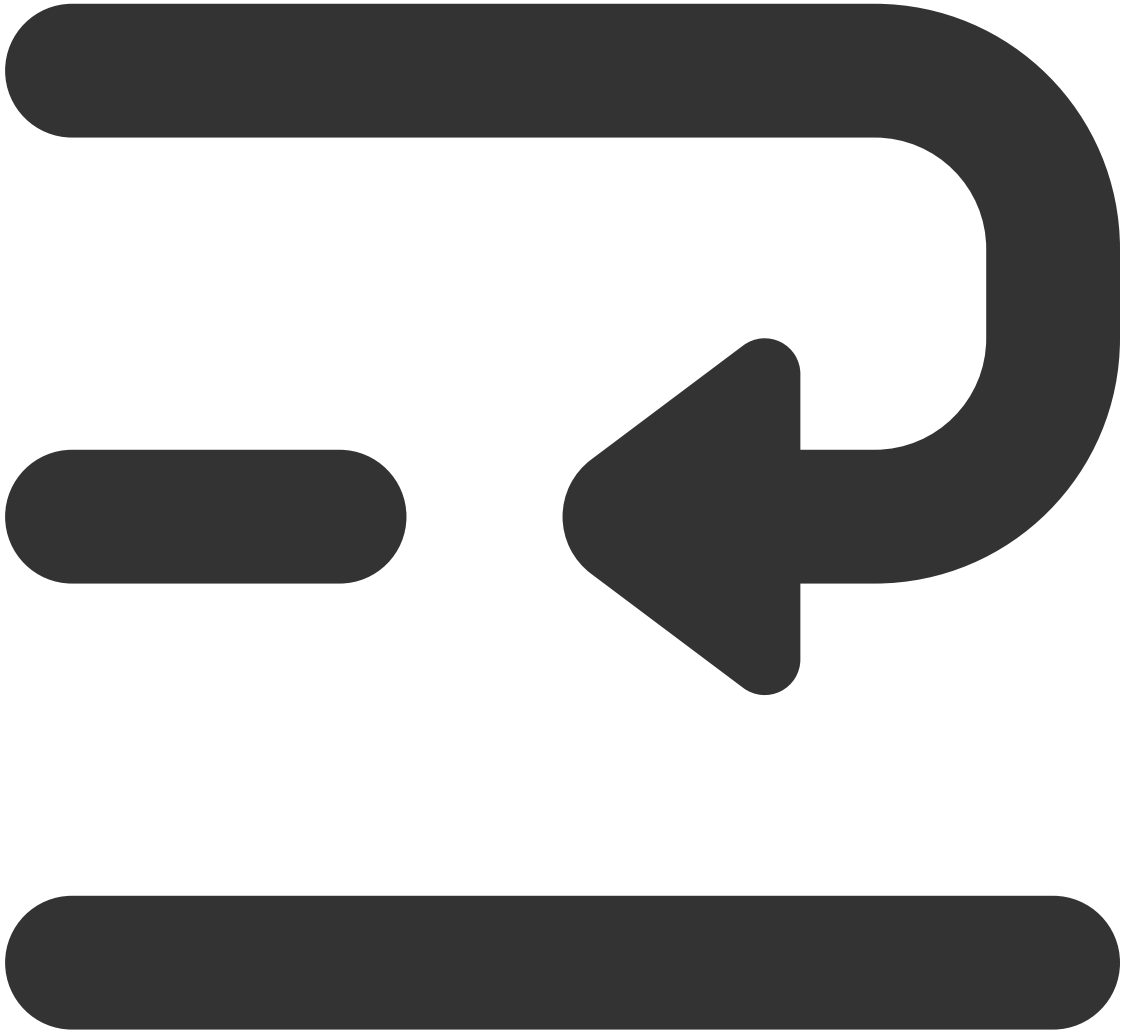
©2013-2022 Tencent Cloud. All rights reserved.



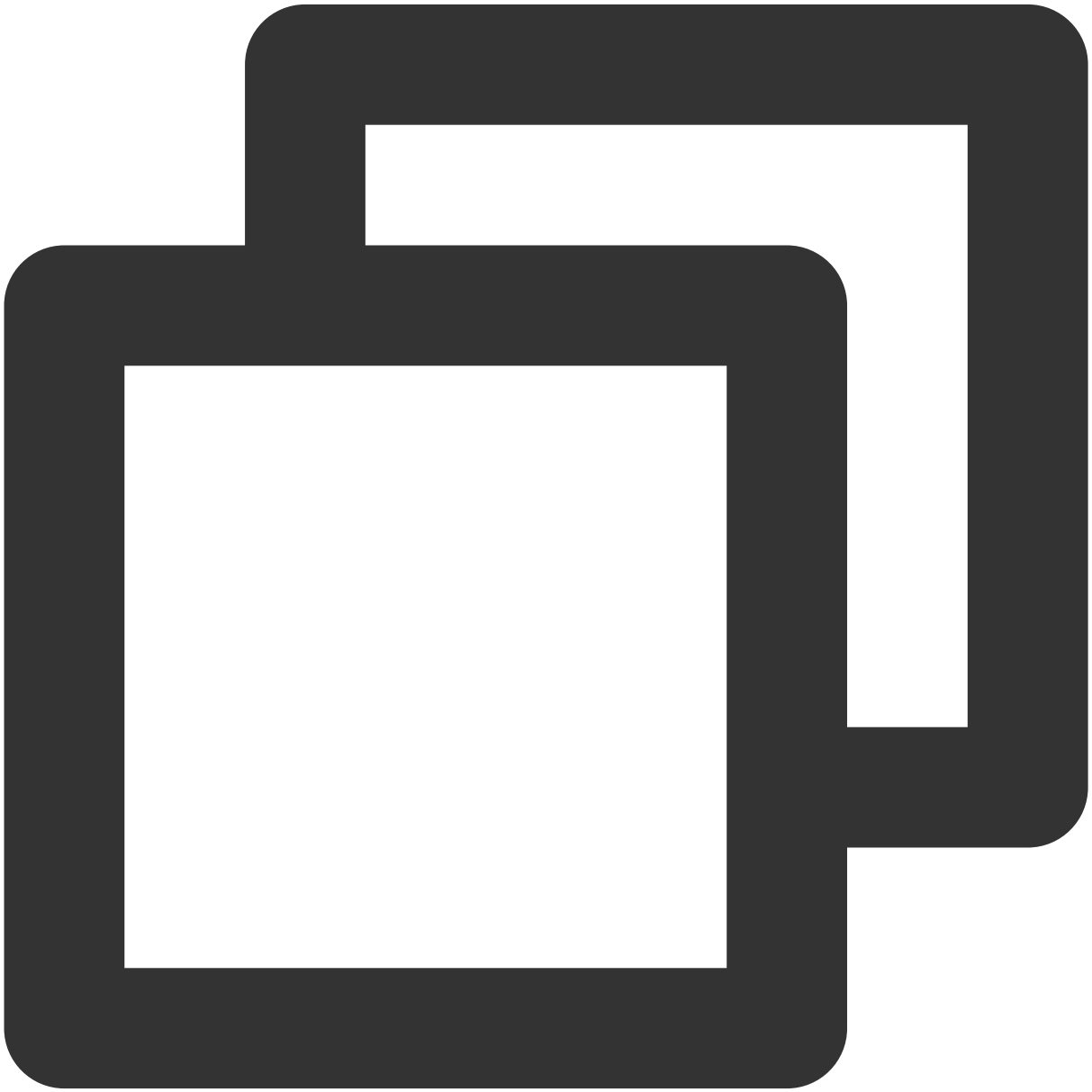
```
- (BOOL)onInvokeWebAPIWithEvent:(NSString*)event params:(NSDictionary*)params callb
    NSLog(@"onInvokeWebAPIWithEvent, event:%@, params:%@", event, params);
    if (handler) {
        handler(@{@"errMsg" : @"onInvokeWebAPIWithEvent"}, nil);
    }
    return YES;
}
```

boolean onReportEvent(String event, Map<String, String> params)

©2013-2022 Tencent Cloud. All rights reserved.



©2013-2022 Tencent Cloud. All rights reserved.



```
Event reporting. The events are as follows:
 1. Launch the mini program, event: MS_EVENT_LAUNCH, keys for params: pagePath, d;
 2. Duration of mini program usage, event: MS_EVENT_USE_TIME, keys for params: useT
 3. Successful launch of the mini program, event: MS_EVENT_LAUNCH_SUCCESS, keys for
 4. Opening a page within the mini program, event: MS_EVENT_OPEN_PAGE, keys for par
 5. Failure to launch the mini program, event: MS_EVENT_LAUNCH_FAIL, keys for param
```

©2013-2022 Tencent Cloud. All rights reserved.

Sensitive API

Last updated：2024-03-05 15:39:26

Warning:

The APIs enumerated on this page are sensitive in nature, pertaining to the acquisition of user information. Developers are urged to thoroughly comprehend the subsequent API descriptions before submitting their applications for approval. In the event of any unlawful or non-compliant activities by the user, the platform reserves the right to revoke permissions, impose account suspensions, and pursue legal liabilities if necessary.

Location

getLocation

English Name: Get the current geographical location and velocity.
Remarks: To acquire the current geographical location and speed. See getLocation for more details.

choosePoi

English Name: Open the POI list to select a location.
Remarks: Open the POI list to select a location, which supports both vague positioning (accurate to the city) and precise positioning in a mixed selection. For more details, see choosePoi.

chooseLocation

English Name: Open the map to select a location.
Remarks: Open the map to select a location. For more details, see chooseLocation.

startLocationUpdateBackground

English name: Enable the mini program to receive location messages upon entering the frontend and backend.
Remarks: Enables the mini program to receive location messages when entering the frontend and backend, which requires guiding the user to give "authorization". For more details, see startLocationUpdateBackground.

startLocationUpdate

English name: Enable the mini program to receive location messages upon entering the frontend.
Remarks: Enable the mini program to receive location messages when entering the frontend. For more details, see startLocationUpdate.

onLocationChange

English name: Monitoring real-time geographical location change events.

©2013-2022 Tencent Cloud. All rights reserved.

Remarks: Monitor real-time geographical location change events. This needs to be used in conjunction with wx.startLocationUpdateBackground and wx.startLocationUpdate. For more details, see onLocationChange.

getFuzzyLocation

English Name: Get the current fuzzy geographical location.

Remarks: Get the fuzzy current geographical location. See getFuzzyLocation for details.

Open Interfaces

startSoterAuthentication

English name: Biometric authentication

Remarks: Initiate SOTER Biometric Authentication (invoke system API). For more details, see startSoterAuthentication.

checkIsSupportSoterAuthentication

English name: Get biometric authentication method.

Remarks: Get the SOTER biometric authentication method supported by the current device (invoke system API). For more details, see checkIsSupportSoterAuthentication.

checkIsSoterEnrolledInDevice

English Name: Invoke Biometric Authentication Interface.

Remarks: Interface used to determine whether biometric information such as fingerprints is registered in the device (invoke system API). For more details, see checkIsSoterEnrolledInDevice.

Contact person

chooseContact

English Name: Get Contact List.

Remarks: Initiate the mobile contact list and select a contact. For more details, see chooseContact.

©2013-2022 Tencent Cloud. All rights reserved.

JS SDK
Web-View

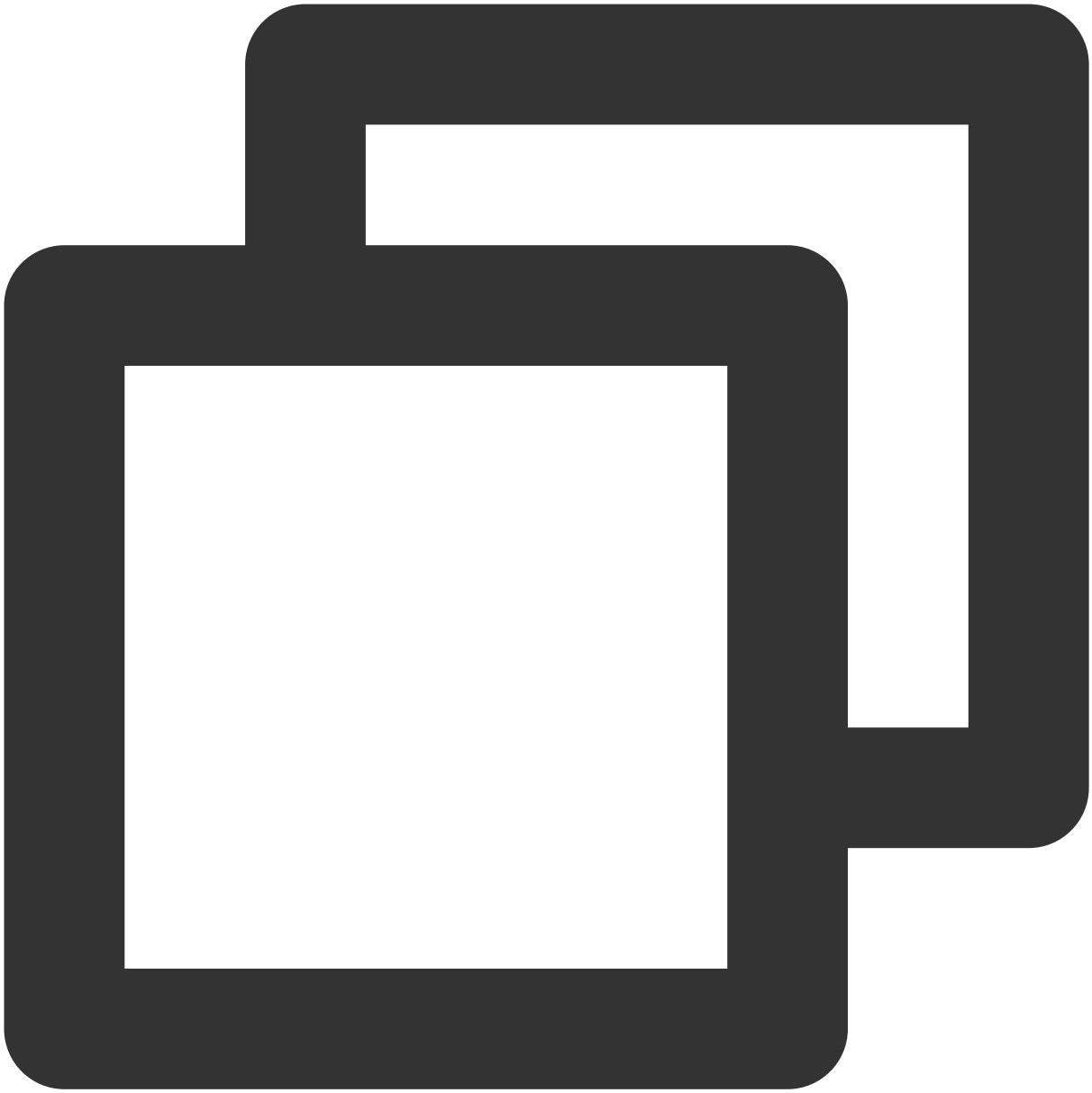
Last updated：2024-07-12 16:51:08

When mini program developers use the WebView component, they may need to call some mini program APIs from the HTML5 page. This is where the JS SDK comes into play.
The JS SDK is a toolkit for web developers. By using the JS SDK, web developers can enable certain native capabilities (such as taking photos, selecting images, recording audio, getting location, etc.) on the HTML5 page within the mini program. This enhances the user experience on the web page.
Since the JS SDK is used within the Webview container of a mini program, it only supports links opened within the mini program. For domain management of the web page, please refer to the domain whitelist configuration in the console.

How to Use

Load the corresponding JS SDK in your web page with the following code:

©2013-2022 Tencent Cloud. All rights reserved.



```
<script src="path/to/jssdk.js"></script>

<script>
    // go to pay Page from click in H5 Page
    document.getElementById("payBtn").click = funciton(e){
        wx.miniProgram.navigateTo({ url : "/pages/pay/pay"});
    }
</script>
```

©2013-2022 Tencent Cloud. All rights reserved.

Download link

jssdk.js(19.5KB)

下载

©2013-2022 Tencent Cloud. All rights reserved.

API description

Last updated：2024-07-12 16:45:48

Relevant APIs

API	Description
wx.miniProgram.navigateTo	The parameters are the same as those of the corresponding mini program API
wx.miniProgram.navigateBack	The parameters are the same as those of the corresponding mini program API
wx.miniProgram.switchTab	The parameters are the same as those of the corresponding mini program API
wx.miniProgram.reLaunch	The parameters are the same as those of the corresponding mini program API
wx.miniProgram.redirectTo	The parameters are the same as those of the corresponding mini program API
wx.miniProgram.postMessage	Sends a message to the mini program, triggering the component's message event in specific scenarios such as returning to a mini program page, terminating or sharing a component and copying links
wx.miniProgram.getEnv	Gets the current environment information

Relevant APIs

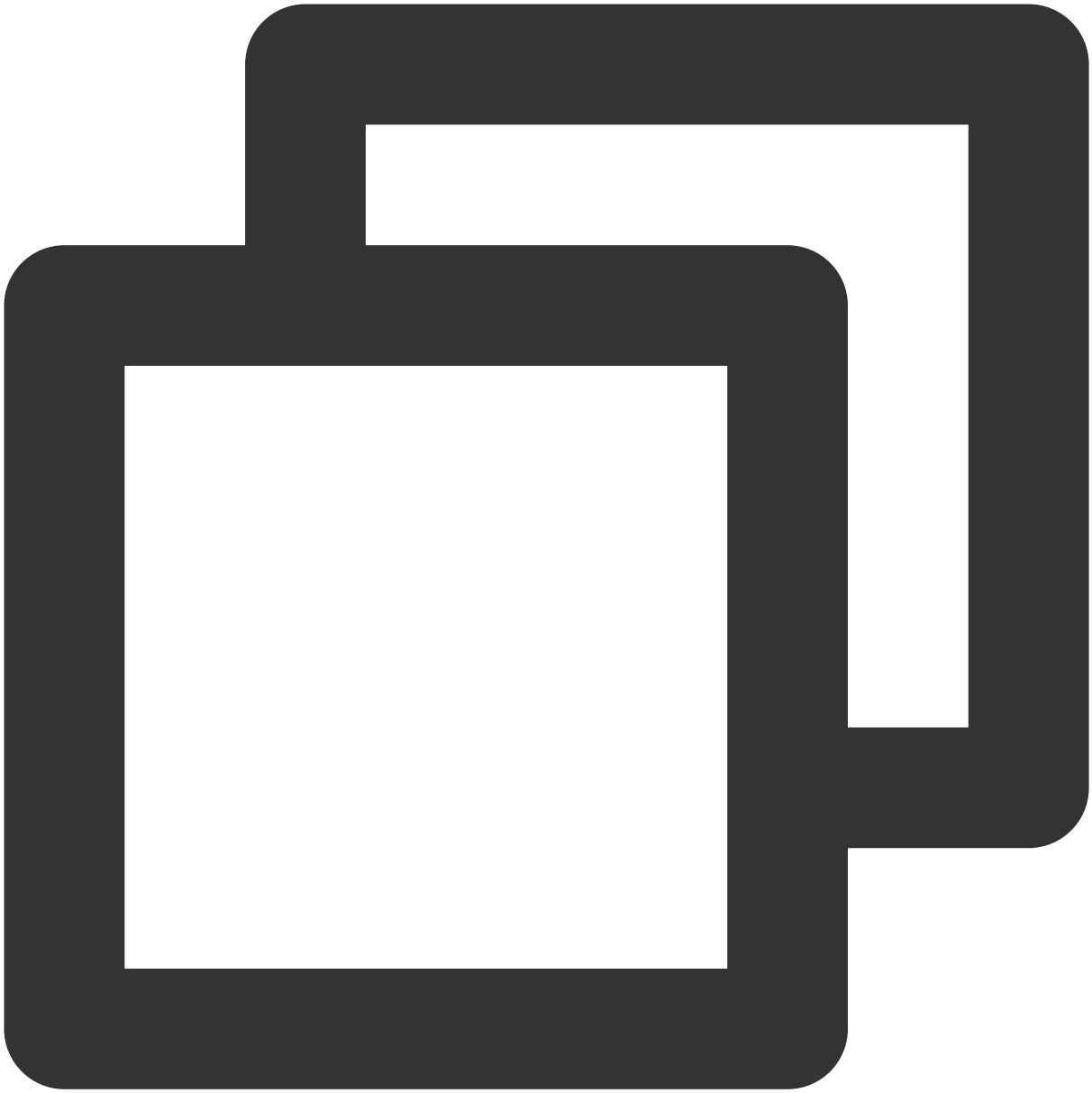
API Module	API description	Specific APIs
Image APIs	Take a photo or upload an image	chooseImage
	Preview an image	previewImage
	Upload an image	uploadImage
	Download an image	downloadImage
	Get local images	getLocalImgData

©2013-2022 Tencent Cloud. All rights reserved.

Audio APIs	Start recording	startRecord
	Stop recording	stopRecord
	Recording listening stops automatically	onVoiceRecordEnd
	Play audio	playVoice
	Pause	pauseVoice
	Stop playback	stopVoice
	Audio playback listening ends	onVoicePlayEnd
Device information	Get network status	getNetworkType
Geographical location	Display location in built-in map	openLocation
	Gets the geographical location	getLocation

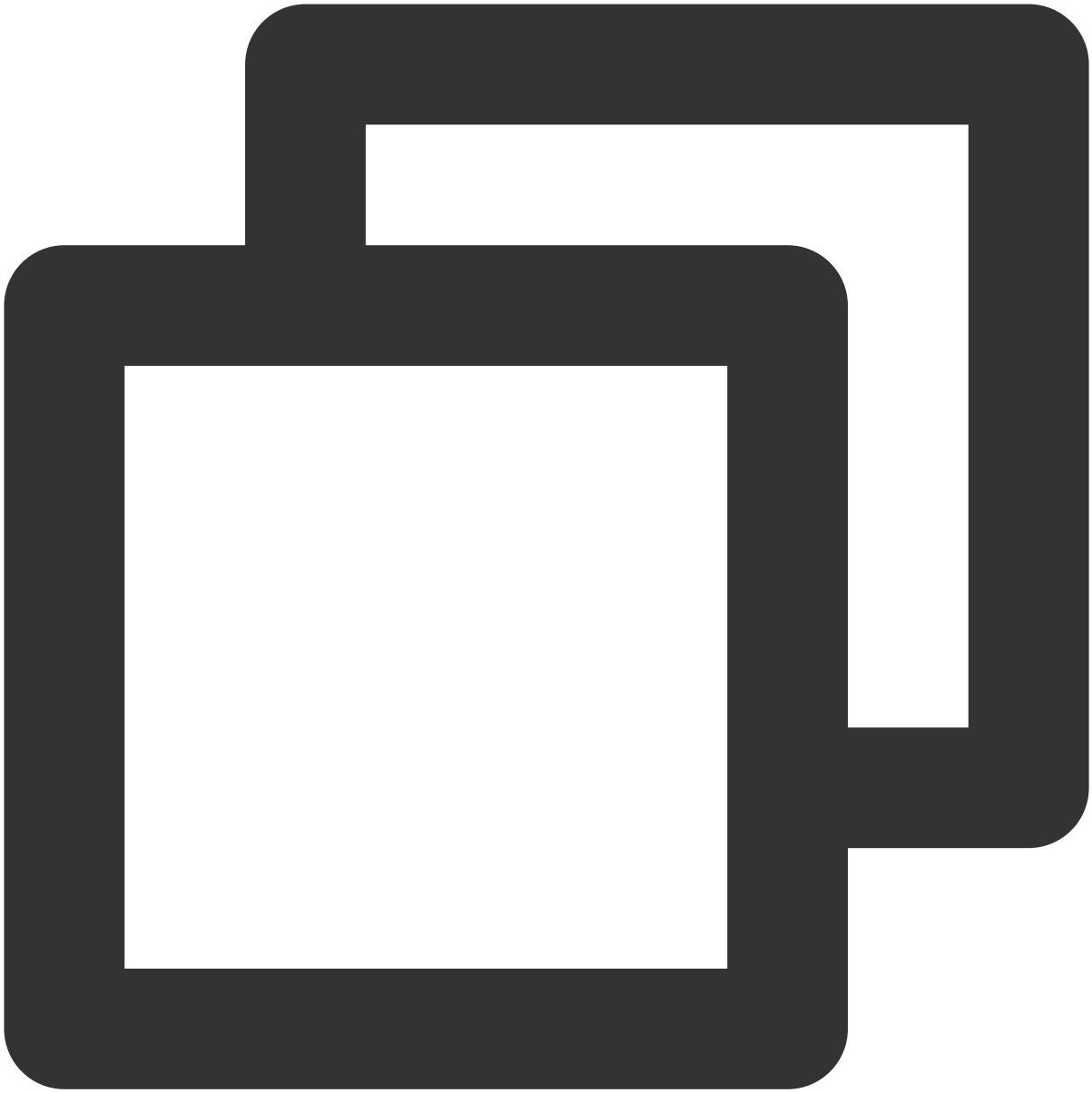
Sample code

©2013-2022 Tencent Cloud. All rights reserved.



```
<!-- html -->
<script type="text/javascript" src="https://raw.githubusercontent.com/TencentCloud/
```

©2013-2022 Tencent Cloud. All rights reserved.



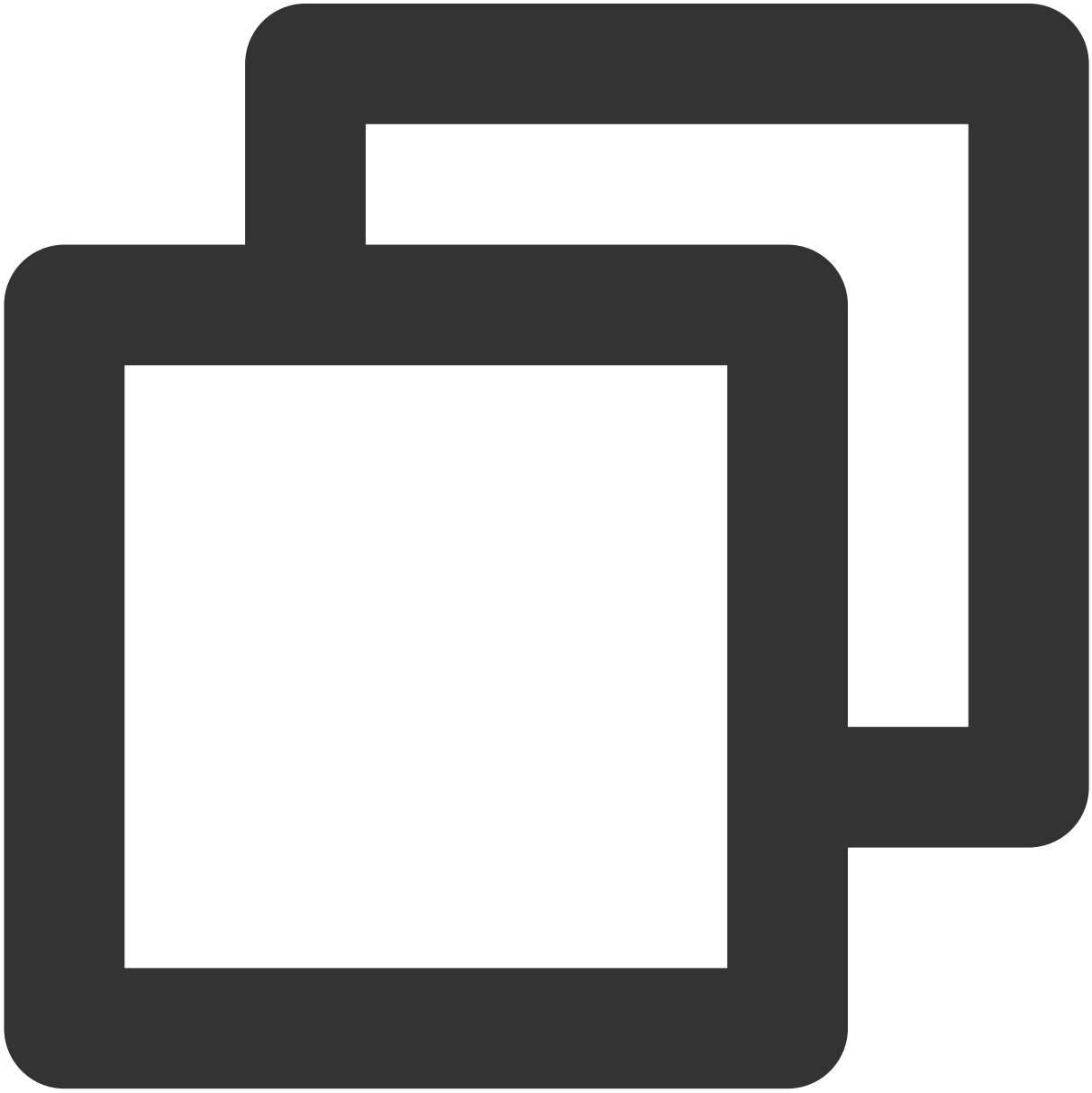
```
wx.miniProgram.navigateTo({url: '/path/to/page'})
wx.miniProgram.postMessage({ data: 'foo' })
wx.miniProgram.postMessage({ data: {foo: 'bar'} })
wx.miniProgram.getEnv(function(res) { console.log(res.miniprogram) })
```

Relevant APIs

©2013-2022 Tencent Cloud. All rights reserved.

When users share, the current WebView URL can be obtained in the onShareAppMessage callback via the webViewUrl parameter.

Sample code



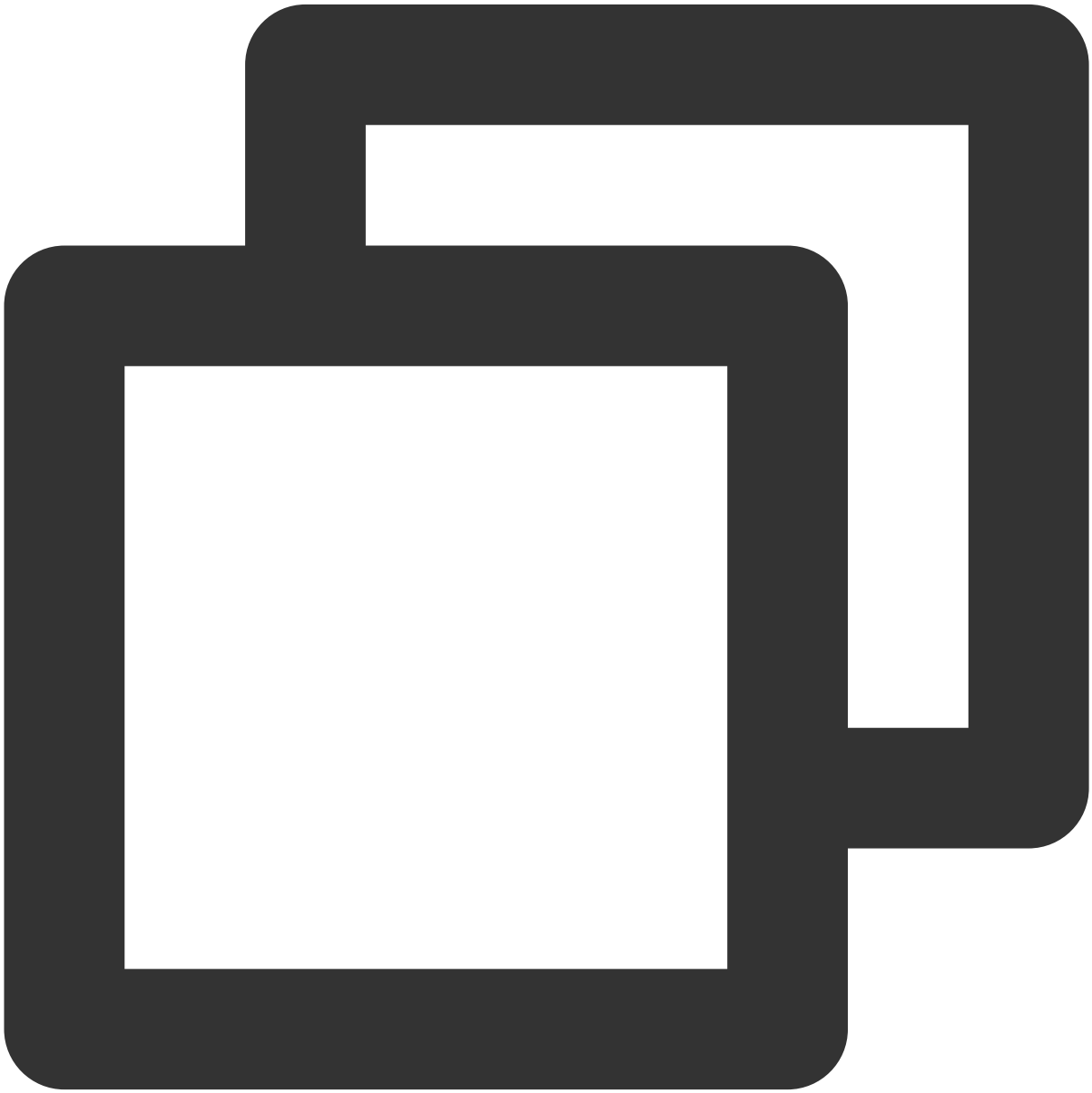
```
Page({
    onShareAppMessage(options) {
        console.log(options.webViewUrl)
    }
})
```

©2013-2022 Tencent Cloud. All rights reserved.

Relevant APIs

In the web page, you can determine if you are in a mini program environment using the window.__wxjs_environment variable. It is recommended to use this in the WeixinJSBridgeReady callback, or you can use the getEnv interface provided by the JS SDK.

Sample code



```
// Inside the WebView page
function ready() {
  console.log(window.__wxjs_environment === 'miniprogram') // true
```

©2013-2022 Tencent Cloud. All rights reserved.

```
}
if (!window.WeixinJSBridge || !WeixinJSBridge.invoke) {
  document.addEventListener('WeixinJSBridgeReady', ready, false)
} else {
  ready()
}

// O
wx.miniProgram.getEnv(function(res) {
  console.log(res.miniprogram) // true
})
```

Tips

The domain of the iframe within the web page also needs to be configured in the domain whitelist;
In the developer tools, you can open the WebView debugging interface in the WebView component preview;
Each page can only have one WebView, which will automatically fill the entire page and cover other components;
Communication between the WebView page and the mini program is only supported through the APIs provided by the JS SDK;
Avoid using Chinese characters in the link, as it may cause a white screen issue on iOS. It is recommended to use encodeURIComponent.

©2013-2022 Tencent Cloud. All rights reserved.

Basic Library
Basic Library Introduction

Last updated：2024-07-12 16:45:16

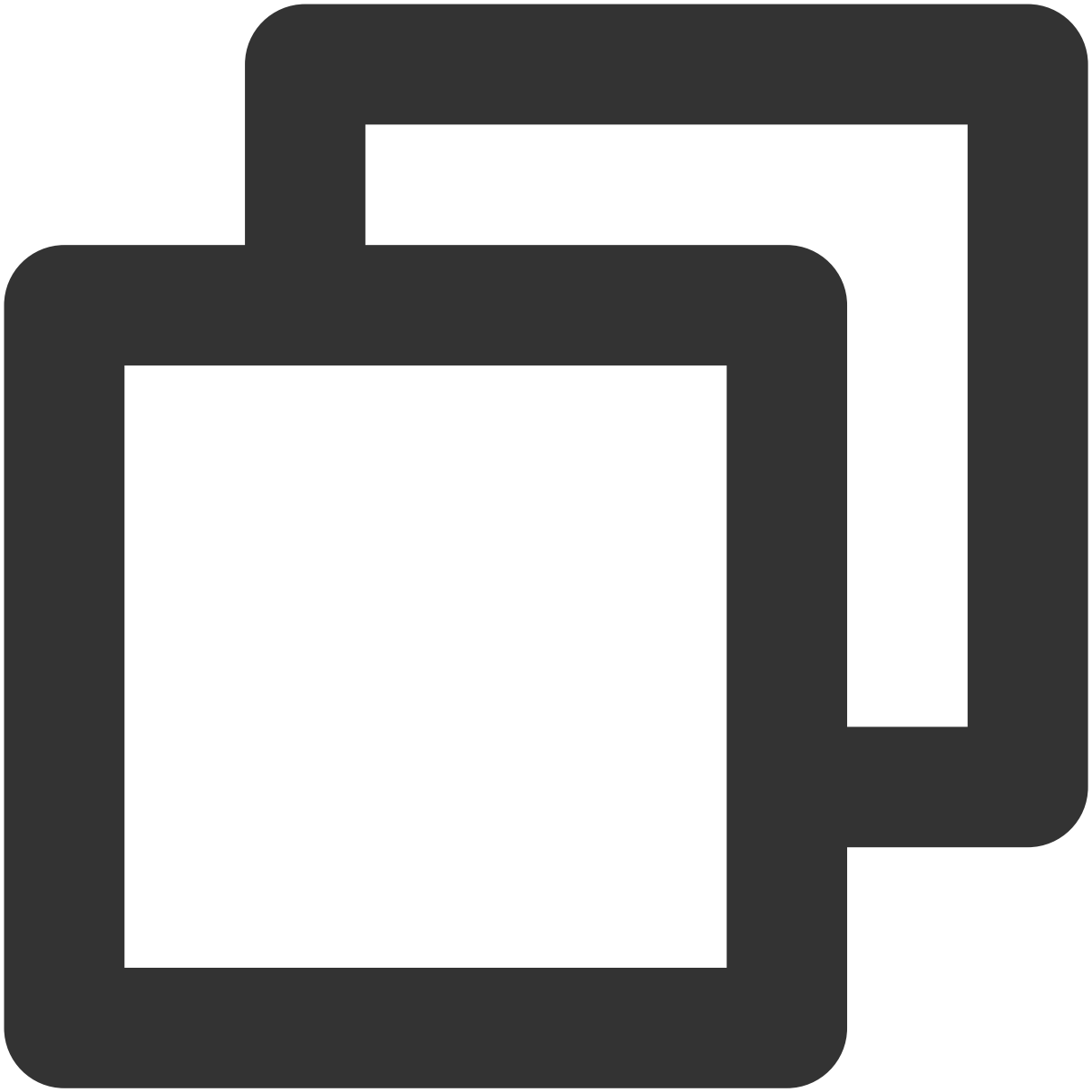
Introduction

The base library acts as a middleware responsible for loading mini programs. It pre-packages many capabilities of the app and the phone (such as Bluetooth, NFC, etc.), allowing developers to use standard components and APIs to develop mini programs.
You can think of the base library as similar to common web development libraries like React or Lodash. However, it is more specialized because the capabilities of mini programs require support from the SDK. Therefore:
The base library, like the host app, has its own version number.
New capabilities added in each version of the base library need to run in specific versions.
Some capabilities of higher versions of the base library are not compatible with lower version SDKs.

Typically, the base library provides two main parts for mini program developers: components and APIs.
WXML files in mini programs, when compiled, can render real DOM nodes in the environment provided by the base library.
APIs like Page and Component used in JS files of mini programs are also exposed globally by the base library.
The wx.xxx APIs used in mini programs actually call native capabilities (like Bluetooth, WiFi, etc.), and the base library acts as a "bridge" between the mini program and the native container.

In short, the base library mainly handles data binding, provides a component system, event system, communication system, and other framework logic. From the dual-process model perspective, it includes rendering issues of the Virtual DOM in the rendering layer and built-in components and APIs in the logic layer. Additionally, the base library provides supplementary capabilities for mini programs, such as custom components and performance compatibility.

©2013-2022 Tencent Cloud. All rights reserved.



```
const {SDKVersion} = wx.getSystemInfoSync(); // fetch jsLib version
```

Base library mechanism

When developing web pages, we often reference third-party JavaScript libraries. Before using the APIs provided by these libraries, we need to include them in our business code.

Similarly, when opening a mini program in an app, we need to load the base library before starting the mini program, and then load the business code. Since the rendering layer and logic layer of mini programs are managed by two

©2013-2022 Tencent Cloud. All rights reserved.

threads, when we talk about the base library, it usually includes the WebView base library (rendering layer) and the AppService base library (logic layer).

Before opening any mini program based on TCMPP, the same base library needs to be injected. Therefore, we do not need to package the base library in the mini program's code package. Instead, we need to integrate the TCMPP mini program SDK in the app and introduce the mini program base library through the SDK. Introducing the base library through the SDK has two benefits:
It can reduce the size of the mini program code package.
It allows adding capabilities or fixing bugs in the base library without modifying the mini program's code.

Mini program startup
When you open a mini program, you'll see a loading page that includes the mini program's icon and name. On this page, the base library and code package are downloaded first. Once the downloads are complete, the base library and code package are loaded separately.

©2013-2022 Tencent Cloud. All rights reserved.

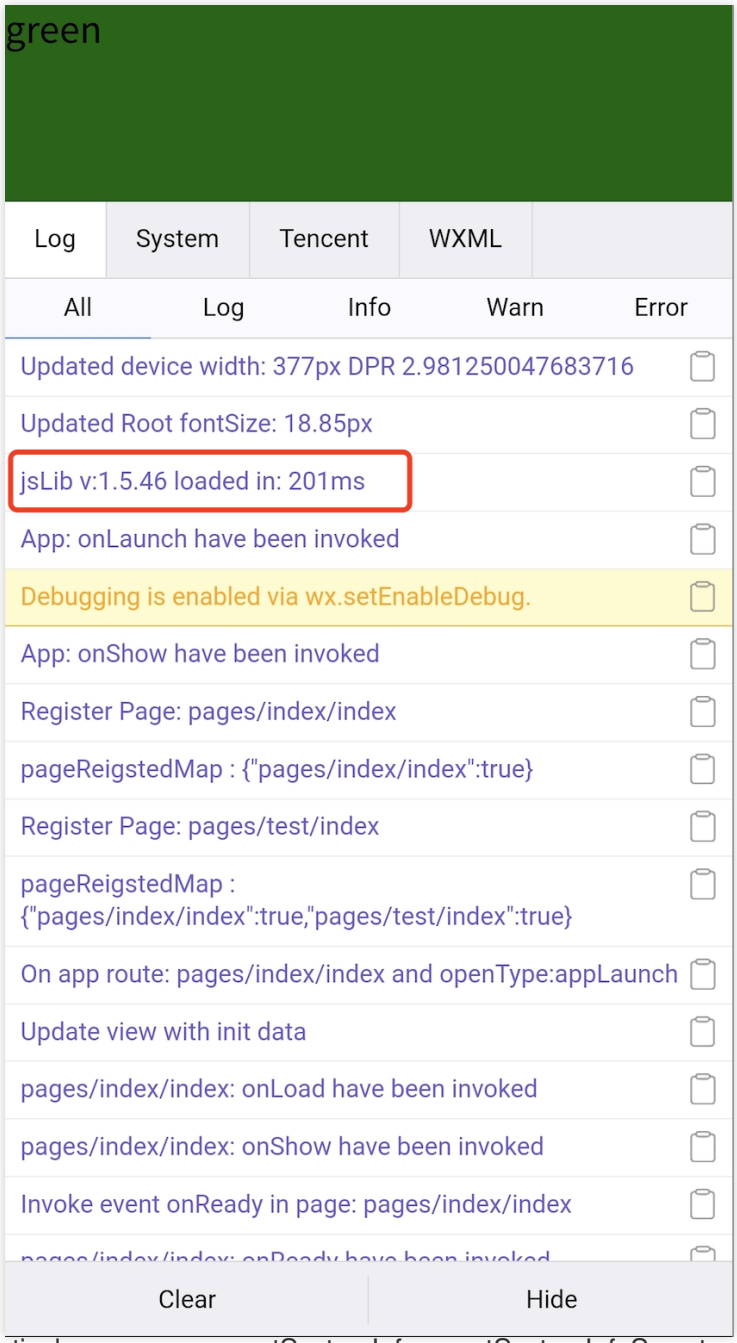
Base library maintenance

Last updated：2024-07-12 16:44:14

Base library FAQs

1. **Who can update the base library version?**
The official TCMPP platform.
2. **How long does it take for the base library update to take effect?**
App：After the base library update is completed, the SDK will check for a new version of the base library the next time you open a mini program. If a new version is detected and the app is running in the foreground, the new version of the base library will be automatically downloaded. The next time you open the mini program, the new version of the base library will be used.
IDE：You can switch to the corresponding version of the base library on the project development panel.
3. **How to check the version of the base library in use?**
If the mini program is in debug mode, you can see the base library version information at the top of the logs.

©2013-2022 Tencent Cloud. All rights reserved.



Alternatively, you can use getSystemInfo or getSystemInfoSync to check the base library version information.

©2013-2022 Tencent Cloud. All rights reserved.

IDE Operation Instructions
Overview

Last updated：2024-07-12 16:42:38

TCMPP IDE is a cross-platform desktop application, designed to assist developers in accomplishing the development, preview, and debugging of Mini Programs. Upon completion of debugging and previewing, developers are required to conduct regression testing on real devices, as certain device APIs (such as Bluetooth, NFC, WIFI, etc.) need such regression testing.

Supported Platforms and Acquisition

TCMPP Mini Program Developer Tools support common mainstream platforms. For the download process, see Getting Started.

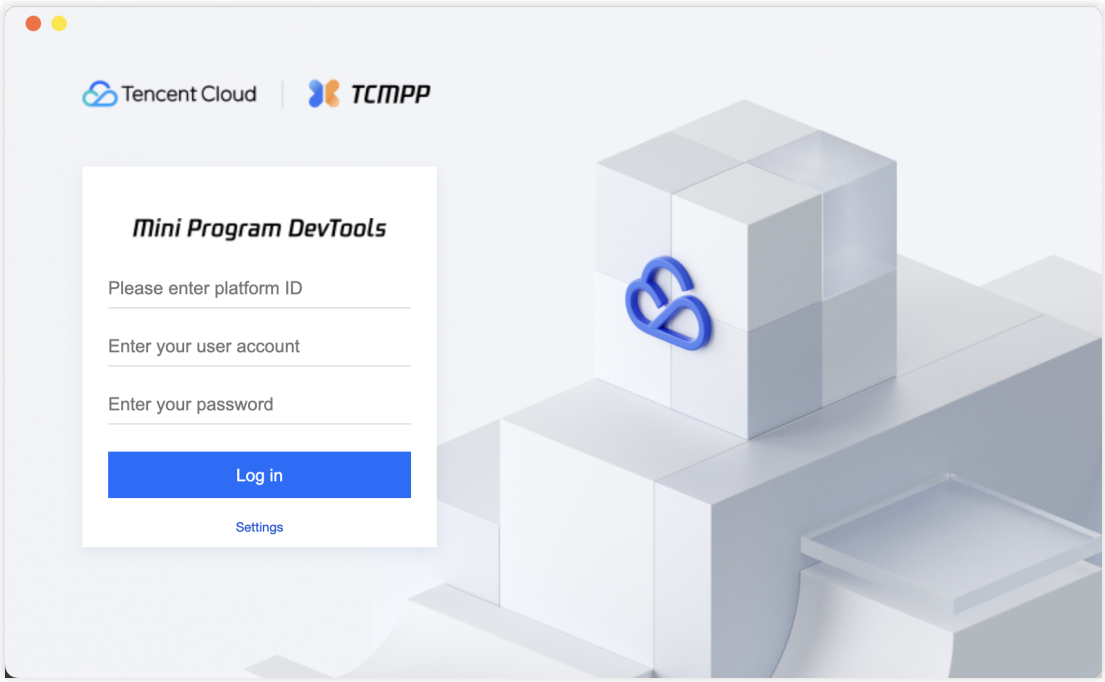
©2013-2022 Tencent Cloud. All rights reserved.

Interfaces

Last updated：2024-07-12 16:43:31

Startup page

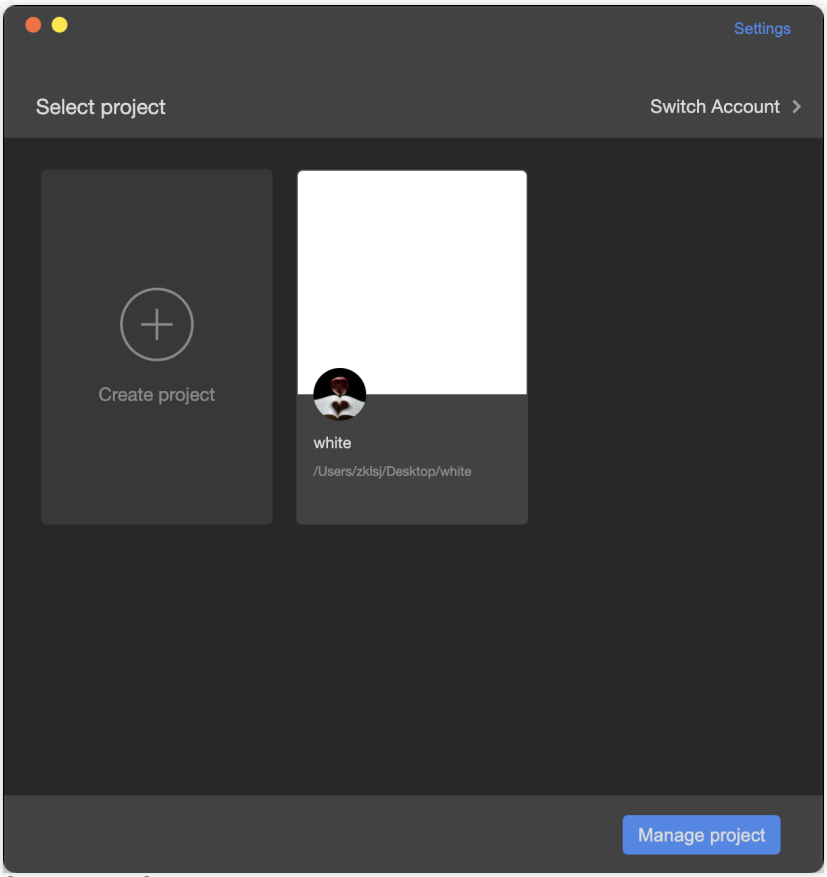
On the login page, users need to enter the platform ID (to distinguish different tenants), account, and password to log in. For branded versions of the IDE, users can log in directly with their account and password.



Project list

After a successful login, you will see a list of existing projects.

©2013-2022 Tencent Cloud. All rights reserved.



Creating a project

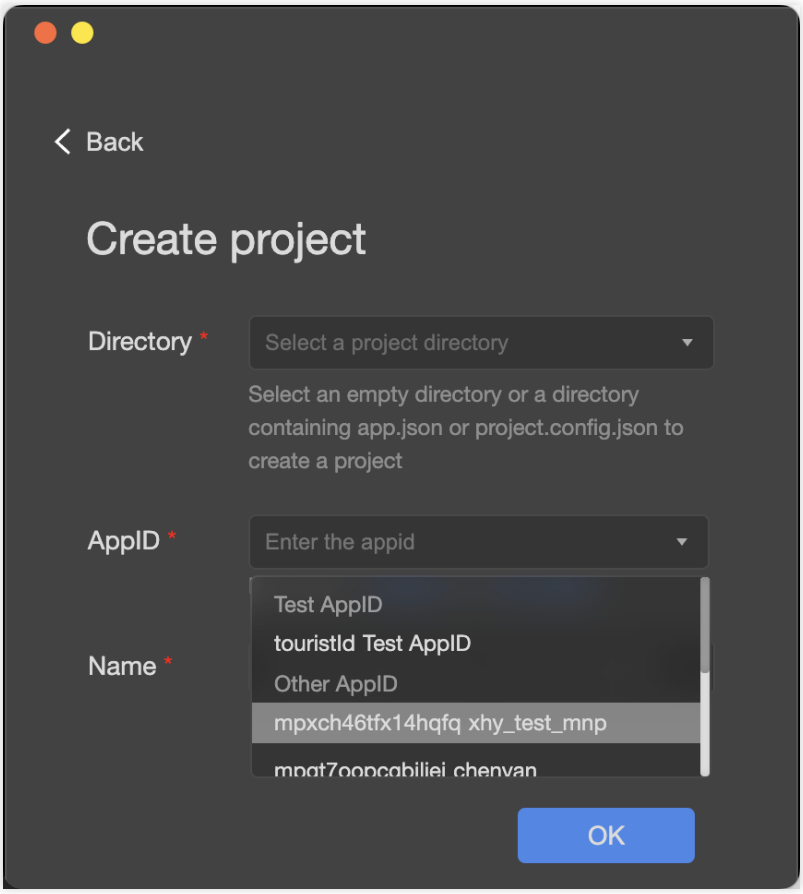
To create a local project, select an empty folder. A simple default project will be generated in this directory. If the folder contains app.json, the project will be opened directly.

When creating a new project, after confirming the path, you also need to fill in the AppId and local name.

The AppId can be manually entered or selected from the current user's existing mini program permissions, or you can use a test mini program AppId.

The name can be anything you choose (for local distinction).

©2013-2022 Tencent Cloud. All rights reserved.

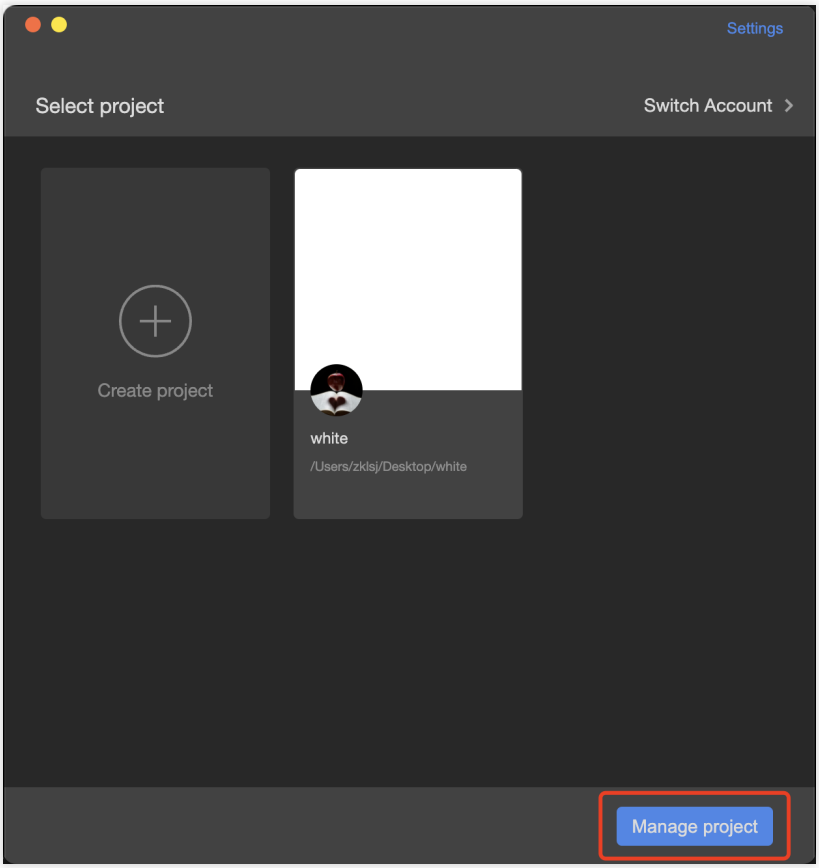


Project management

You can delete or batch delete local projects (this will not affect the actual disk files, only the local project configuration).

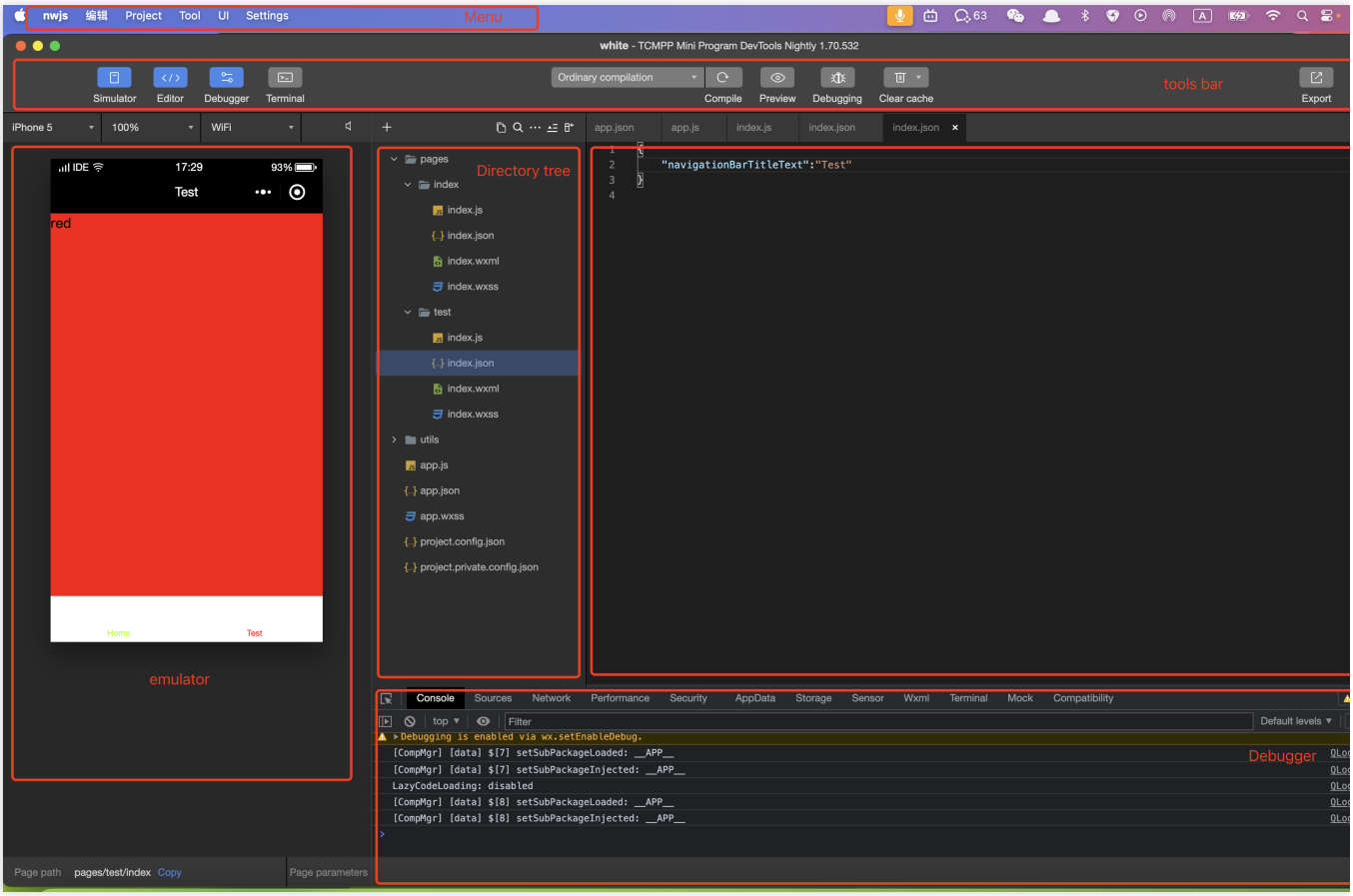
Home page

©2013-2022 Tencent Cloud. All rights reserved.



Home page

©2013-2022 Tencent Cloud. All rights reserved.



Menu bar

TCMPP-Devtools

Switch account: Quickly switch the logged-in user.

About: Information about the developer tools.

Check for updates: Check for version updates.

Debug: Debug the developer tools and editor. If you encounter a suspected bug in the developer tools or editor, you can open the debug tools to check for error logs and provide feedback.

Restart: Restart the IDE.

Exit: Close the developer tools.

Editing

System built-in menu for common text editing operations.

Projects

Create a new project

Open a recent project

View all projects

©2013-2022 Tencent Cloud. All rights reserved.

Close current project

Tools

Compile: Recompile the project.

Refresh: Refresh the project.

Preview: Preview the project and generate a preview QR code.

Upload: Upload the mini program to the backend and open the upload window.

Build npm: Build the current mini program npm package. You can refer to NPM usage for details.

Global search: Entry point for global file search.

UI

Toolbar: Toggle to display and hide.

Simulator: Toggle to display and hide.

Debugger: Toggle to display and hide.

Editor: Toggle to display and hide.

Settings

Language: Switch languages.

Theme: Set theme mode, either dark or light.

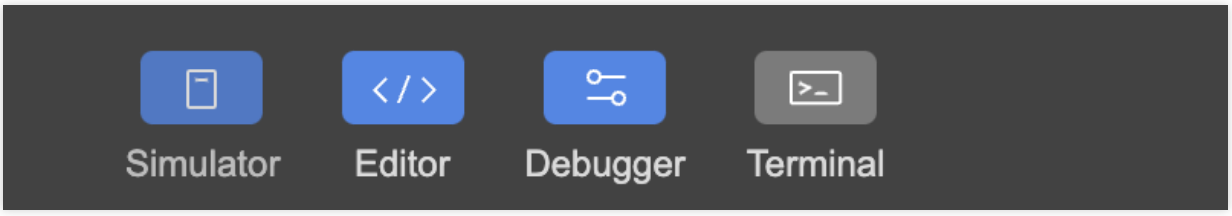
Proxy: Set proxy.

Security settings: Enable server port.

Shortcut settings: Modify shortcuts.

Project settings: View project details panel.

Toolbar



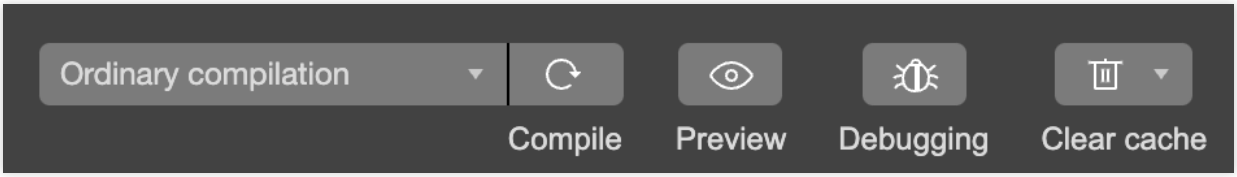
Simulator: Toggle to display and hide the simulator.

Editor: Toggle to display and hide the editor.

Debugger: Toggle to display and hide the debugger.

Terminal: Terminal shortcut.

©2013-2022 Tencent Cloud. All rights reserved.

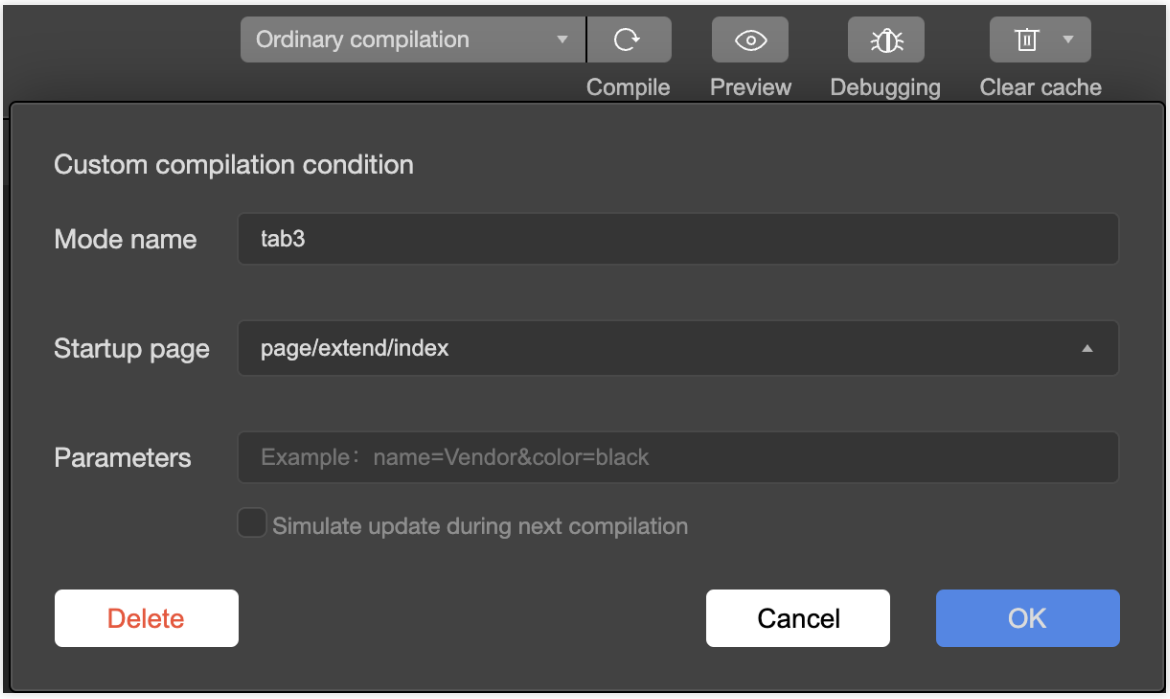


Compile: Recompile the mini program.

Preview: Generate a preview QR code.

Real Device Debugging: Generate a real device debugging code, then scan with a real device to debug.

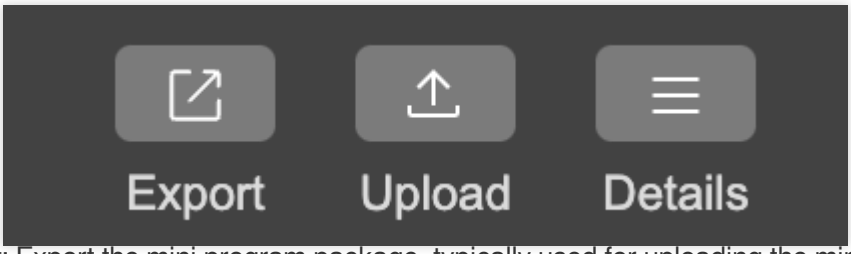
Clear Cache: Conveniently clear file cache, data cache, and backend authorization data on the tool, making it easier for developers to debug.



Add compilation mode: Quickly open a specified page and configure its startup parameters. Once this configuration is set, it will be saved in the project's project.config.json file. It is recommended to commit this file to your Git repository for easy switching of compile modes in the future.

Simulate update during next compilation: Automatically triggers an update request once, which will be automatically canceled the next time.

©2013-2022 Tencent Cloud. All rights reserved.



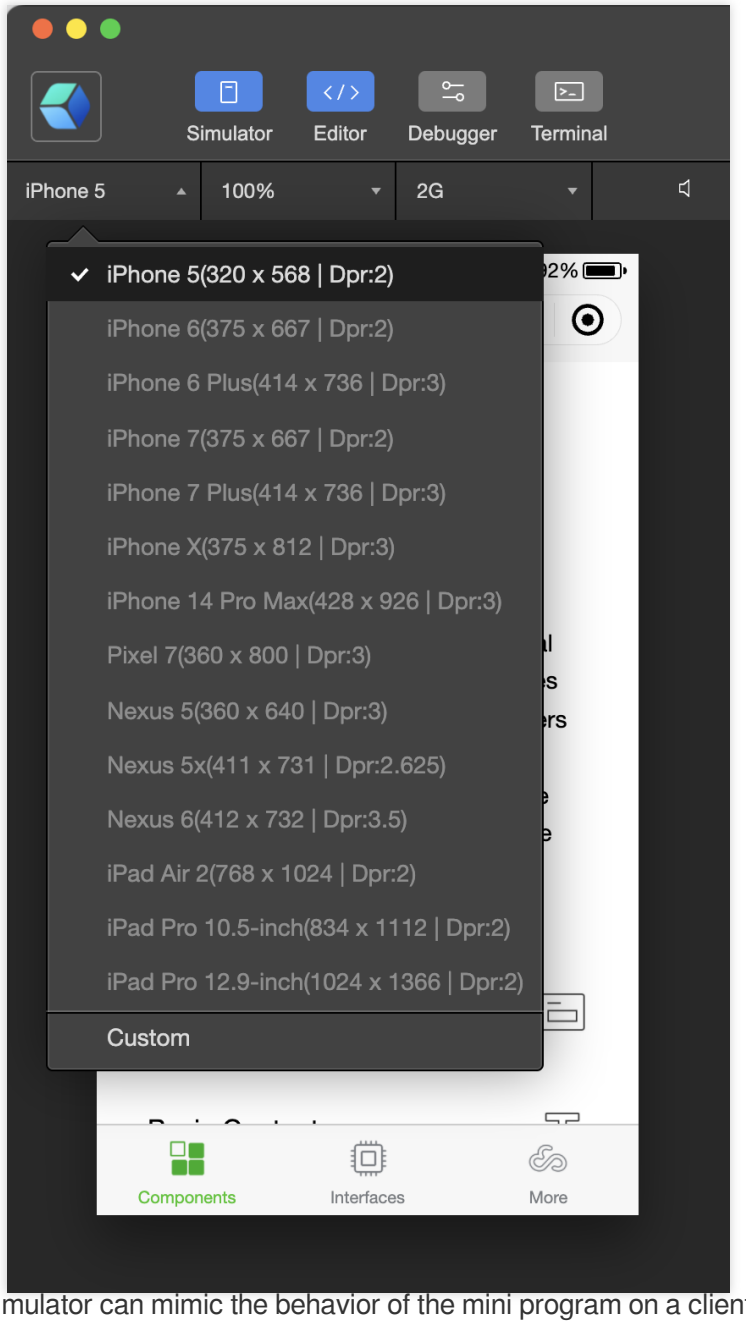
Export: Export the mini program package, typically used for uploading the mini program to the console.

Upload: Directly upload the current mini program to the backend.

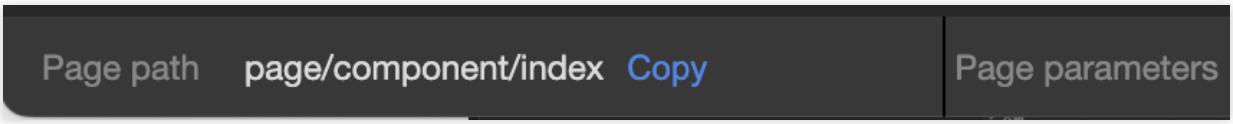
Details: View the mini program details panel.

Simulator

©2013-2022 Tencent Cloud. All rights reserved.



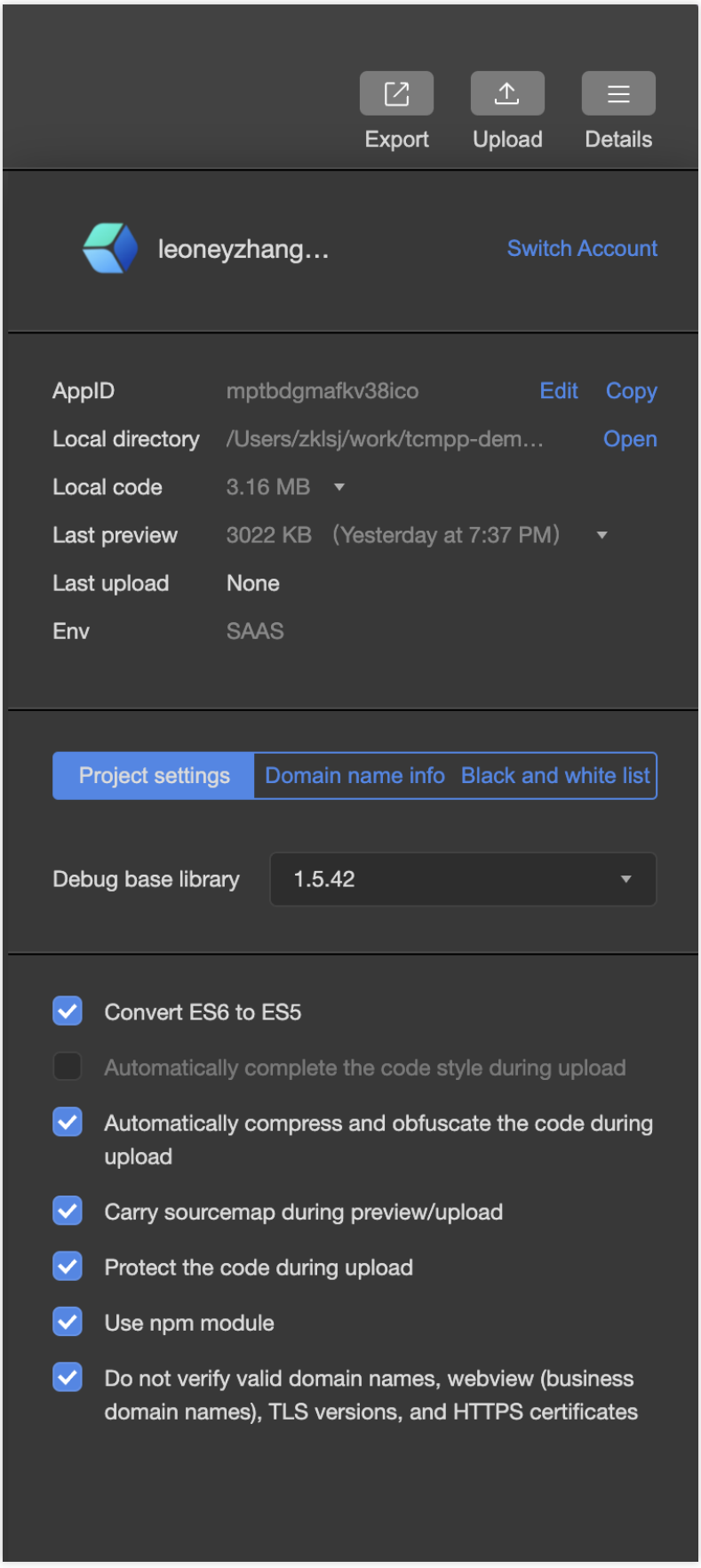
The simulator can mimic the behavior of the mini program on a client. After compiling the mini program's code, it can run directly in the simulator. Developers can choose different devices or add custom devices to debug how the mini program adapts to different screen sizes.



At the bottom status bar of the simulator, you can clearly see the current page path and page parameters of the running mini program.

Project tab

©2013-2022 Tencent Cloud. All rights reserved.



©2013-2022 Tencent Cloud. All rights reserved.

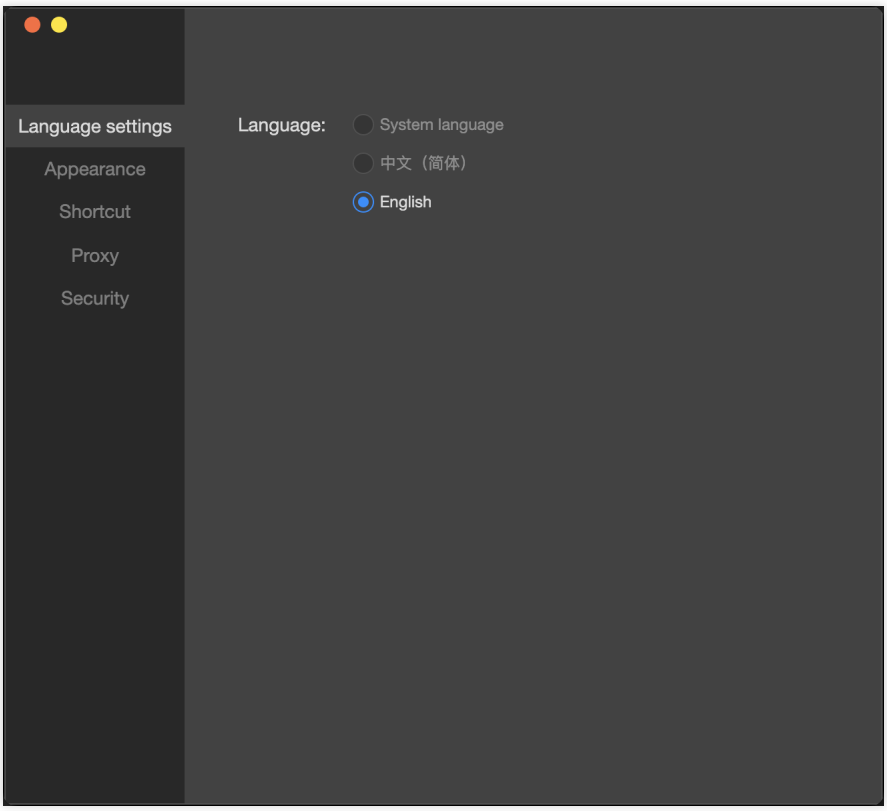
The current panel displays basic information about the mini program, such as AppId, local directory, local code size, last preview, upload information, and current environment information.
Additionally, it shows the base library version, allowing developers to switch between different versions for development and adaptation.
Under domain information, developers can refresh the backend-configured domains. This feature works in conjunction with "Do not verify valid domains, WebView (business domains)." Normally, the request domain and WebView business domain of the mini program need to be configured in advance before going live. For details, see the domain information configuration guide.
The blocklist/allowlist for the same domain information is controlled by the platform. For details see the configuration and usage documentation.

©2013-2022 Tencent Cloud. All rights reserved.

Settings

Last updated：2024-07-12 16:37:01

The IDE settings page allows you to manage and configure various global settings, including theme, language, shortcuts, and security proxies.



Language settings

Set the language for the current editor, supporting Chinese and English, with the default following the system language.

Editor theme

Set the IDE theme, supporting both dark and light mode.

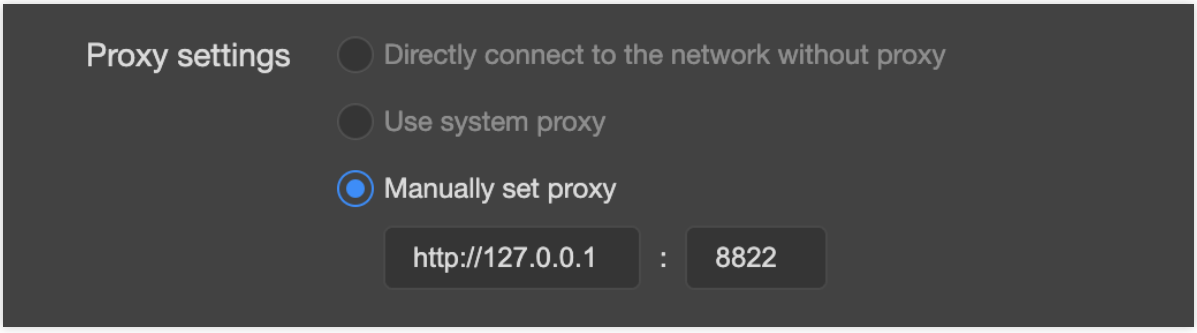
Shortcuts

©2013-2022 Tencent Cloud. All rights reserved.

Include interface toggles and tool shortcuts
for showing and hiding the simulator, debugger, editor, and toolbar.

Proxy

Users can choose to use the system proxy, no proxy (direct connection), or manually set a proxy.
In special network environments (e.g., intranet), you may need to configure a special proxy to access the public network. Manual proxy settings can be configured for such cases.



©2013-2022 Tencent Cloud. All rights reserved.

Code editing

Last updated：2024-07-12 16:36:40

The editing area allows you to write code, add, delete, and rename files for the current project.

File format

Since iOS only supports UTF-8 encoding, the latest version of the developer tool will check the encoding format of code files during upload.

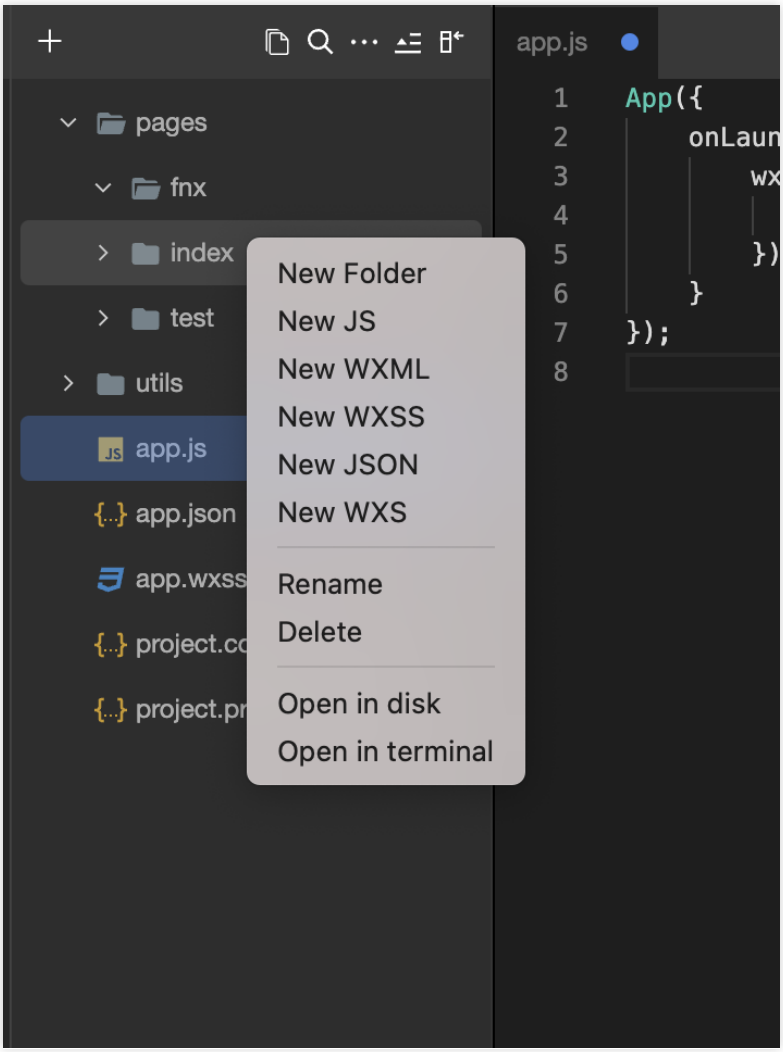
Supported files

The tool currently supports editing five types of files: WXML, WXSS, JS, JSON, WXS, and image file previews.

File operations

Users can right-click on the file tree to quickly create new folders and files.

©2013-2022 Tencent Cloud. All rights reserved.



Real-time preview

When WXML, WXSS, JS, OR JSON files are modified, the mini program will automatically trigger a compilation, and the changes will be previewed in real-time in the simulator.

©2013-2022 Tencent Cloud. All rights reserved.

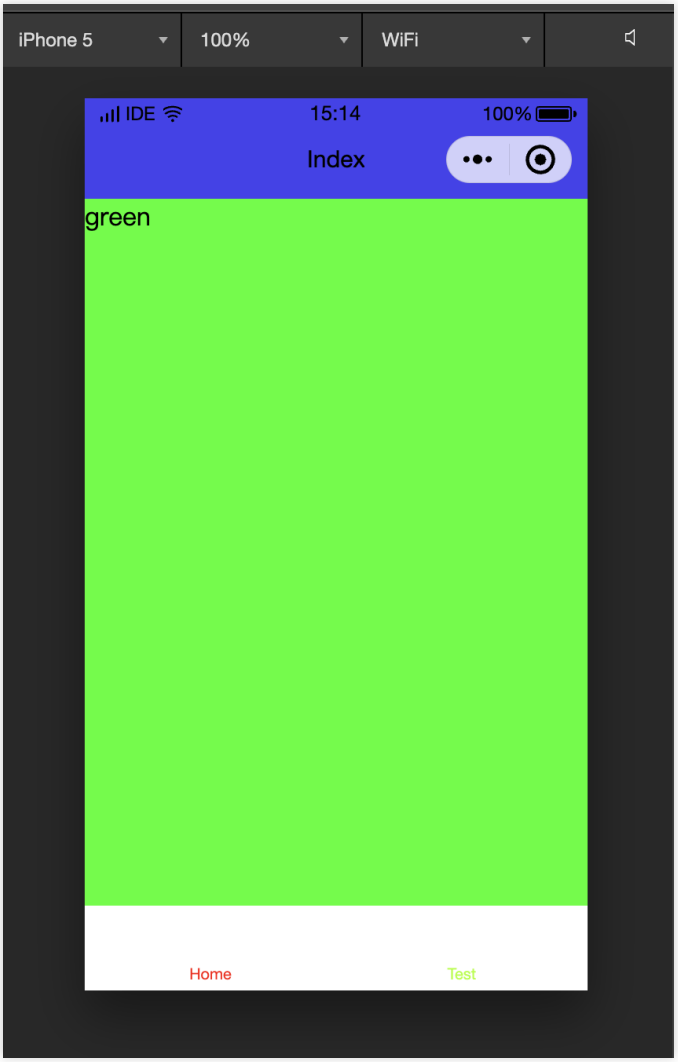
Mini program debugging

Last updated：2024-07-12 16:34:45

The main debugging areas include the simulator, debugging tools, and the mini program operation area.

Simulator

The simulator mimics the real behavior of the mini program on a client, accurately presenting most APIs.



Custom compilation

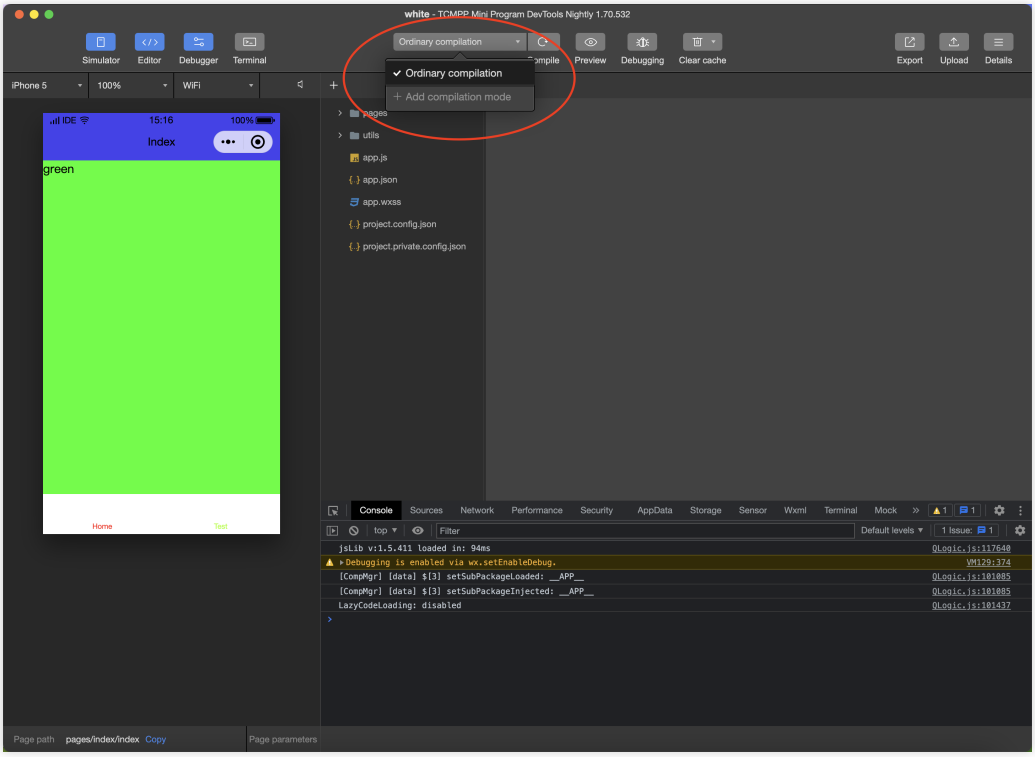
©2013-2022 Tencent Cloud. All rights reserved.

Click the compile button in the toolbar or use the shortcut Ctrl (⌘) + B to compile the current code and automatically refresh the simulator.

Developers can add or select existing custom compile conditions to debug specific pages with different parameters.

Note：

Compile conditions are project-specific and can be saved for each project.



Debugging tool

The debugging tools are divided into nine functional modules: WXML, Console, Sources, Network, Appdata, Storage, Sensor, Mock, and Compatibility.

Wxml panel

This helps developers see the transformed Wxml interface, showing the real page structure and corresponding WXSS properties. You can modify WXSS properties and see changes in real-time in the simulator (changes are not saved to the file). The selector in the top left of the debugging module allows quick location of WXML code for page components.

Console panel

The Console panel has two main features:

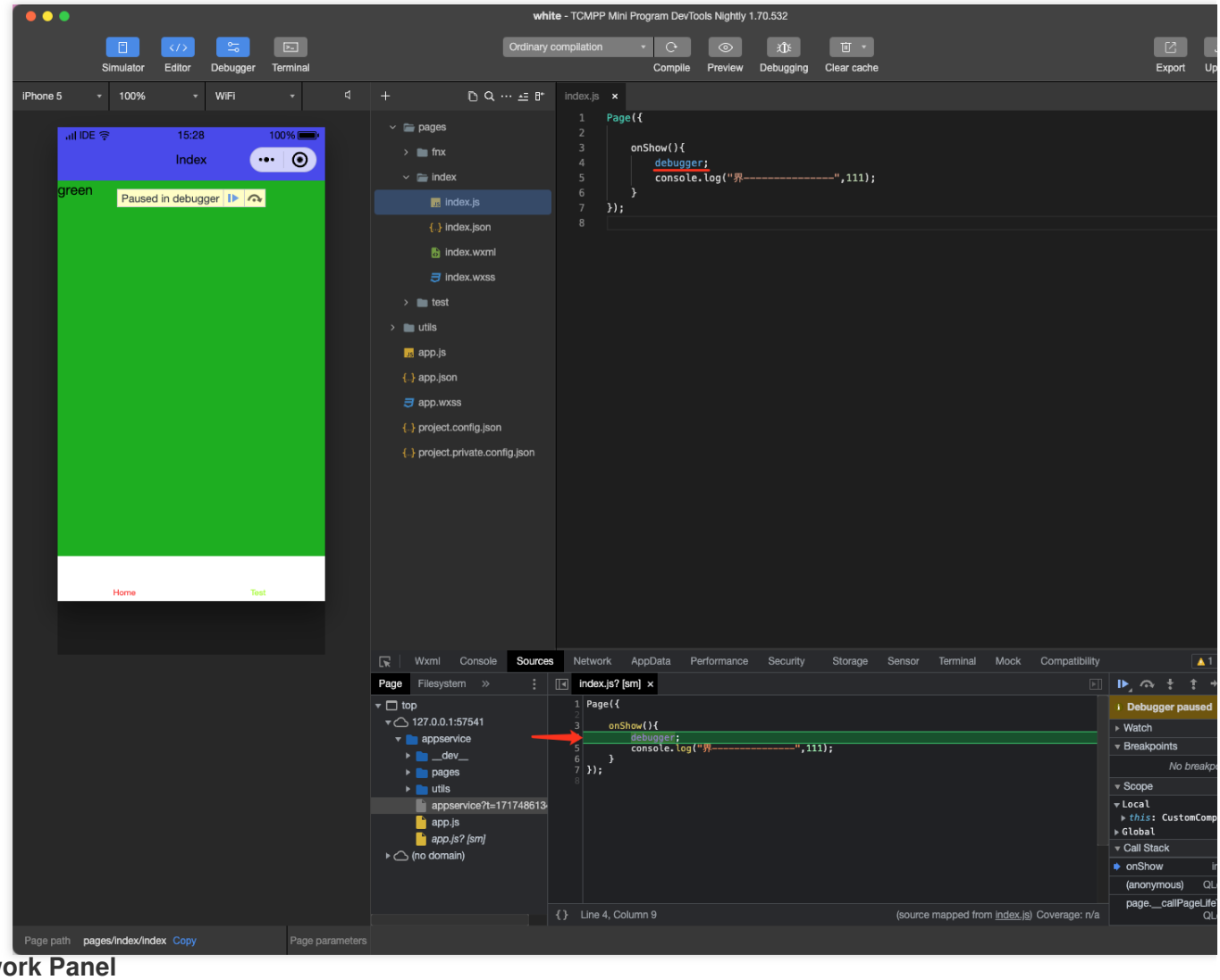
Developers can input and debug code here.

Errors from the mini program are displayed here.

©2013-2022 Tencent Cloud. All rights reserved.

Source panel

This is used to display the script files of the current project. Unlike browser development, the mini program framework compiles script files, so the files shown in the Sources panel are processed scripts. Developer code is wrapped in a define function, and Page code includes a require call at the end.
Developers can add debugger in the code to set breakpoints.



Network Panel

The Network Panel is used to observe and display request and socket request statuses.

Appdata Panel

The AppData panel is used to show the specific data of the mini program's AppData during runtime, reflecting project data in real-time. You can edit data here, and it will be immediately reflected on the interface.

Storage Panel

Storage panel is used to display data stored using wx.setStorage or wx.setStorageSync.
You can delete (using the delete key), add, or modify data directly in the Storage panel.

©2013-2022 Tencent Cloud. All rights reserved.

Sensor Panel

The Sensor panel has two main features:

Developers can simulate geographic location.

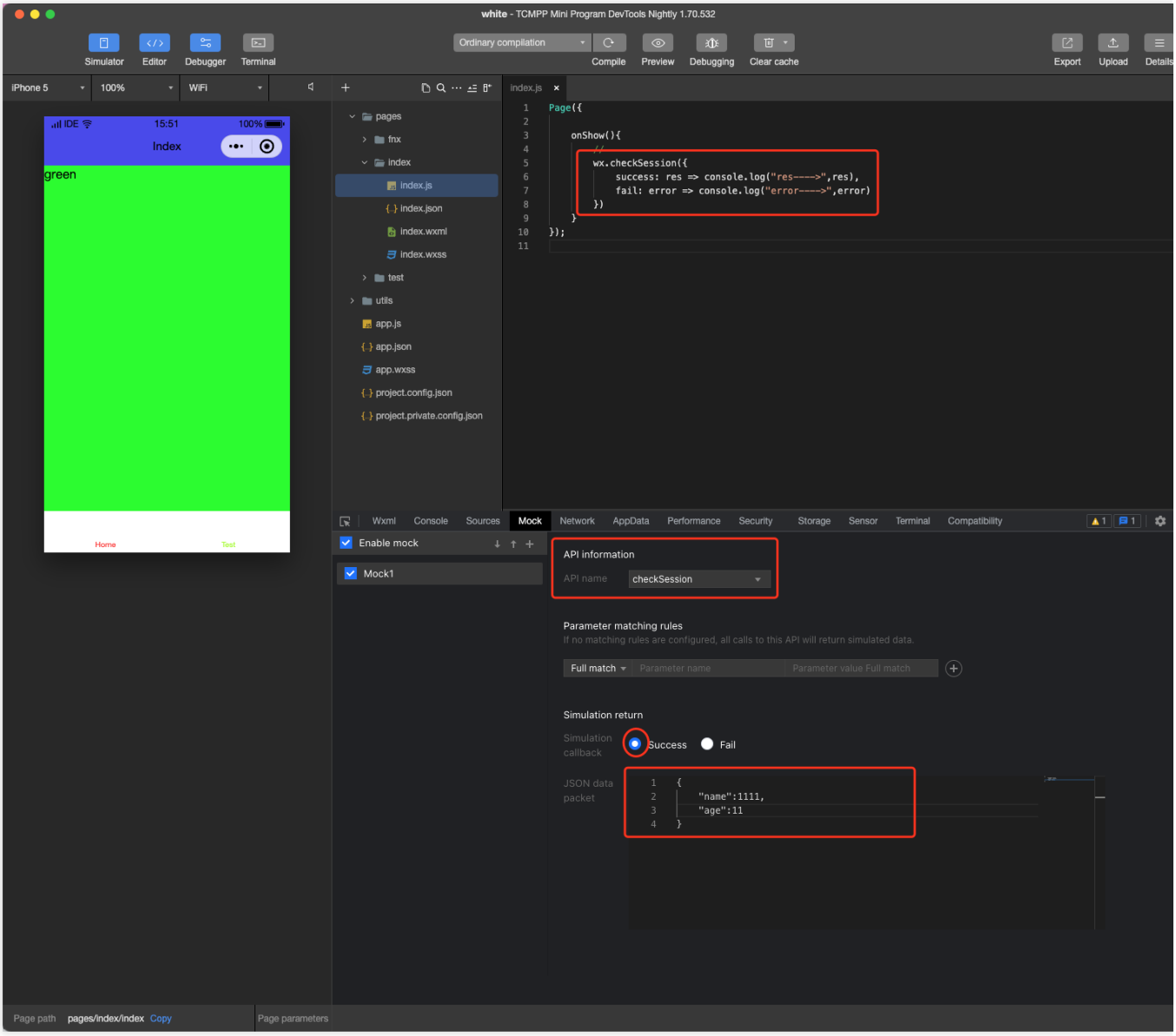
Developers can simulate mobile device behavior for debugging gravity sensor APIs.

Mock Panel

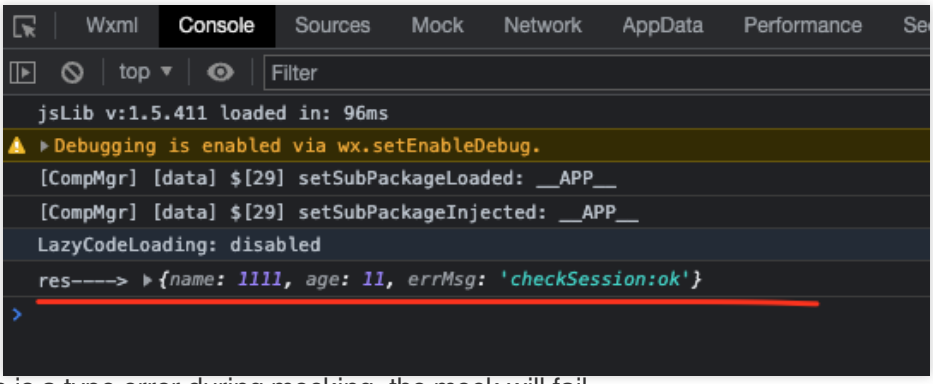
The developer tool provides API Mock capabilities to simulate API call results, useful for developing mini programs.

Developers can mock custom APIs or user information.

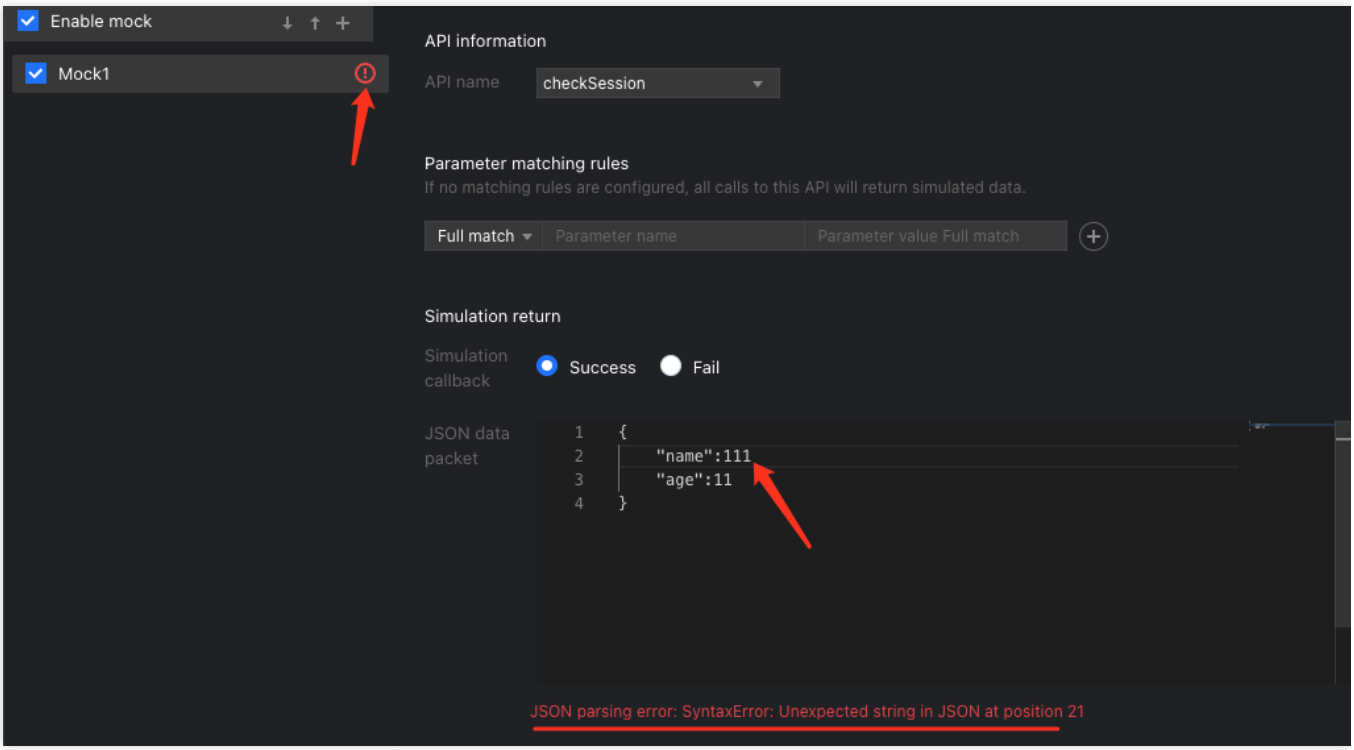
In the following snippet, the checkSession API will be mocked.



©2013-2022 Tencent Cloud. All rights reserved.



If there is a type error during mocking, the mock will fail.



Compatiblity Panel

Mini program migration tool - Compatibility

You can initiate Weixin mini program compatibility checks.

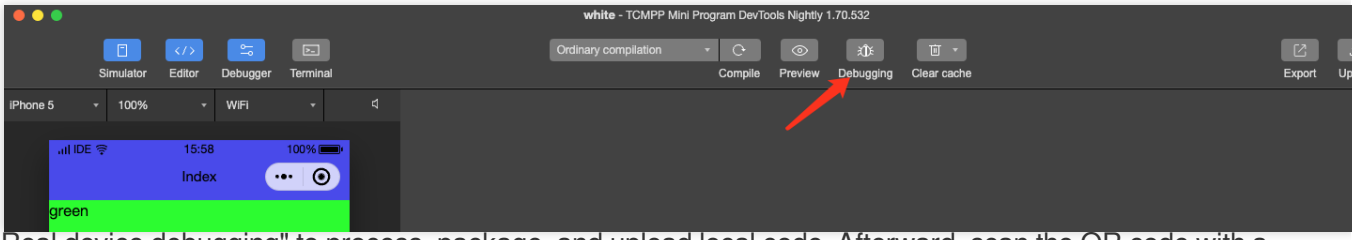
The tool will traverse components, APIs, and JSON files in the mini program, displaying results in the lower window.

You can export the results as an Excel file.

Remote debugging

Real device debugging:

©2013-2022 Tencent Cloud. All rights reserved.



Click "Real device debugging" to process, package, and upload local code. Afterward, scan the QR code with a mobile client to open the debugging window and start real device debugging.

Note：

The versions of Android and iOS SDK must be greater than 1.5.12 for real device debugging.

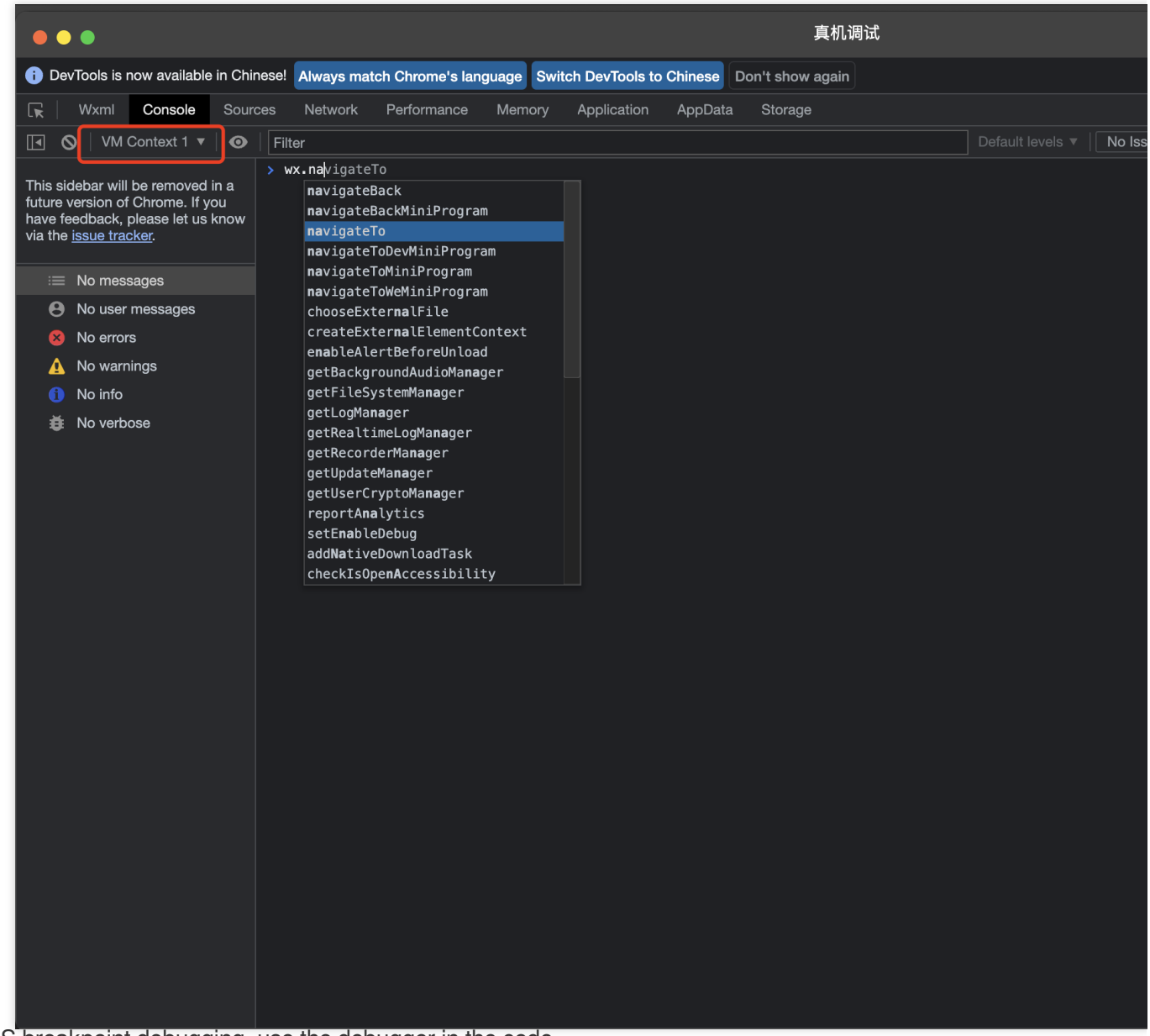
Use an app with the integrated SDK to scan the QR code to start remote debugging.

To end debugging, close the debugging window or click the "End Debugging" button in the lower right corner.

The remote debugging window is divided into two parts: the debugger view on the left and the information view on the right. Developers can debug code directly in the debugger and check Storage status. The information view shows the connection status with the phone and server, and any errors.

In the remote debugger, developers can debug code in the Console panel, view source code and set breakpoints in the Sources panel, and check Storage usage in the Storage panel.

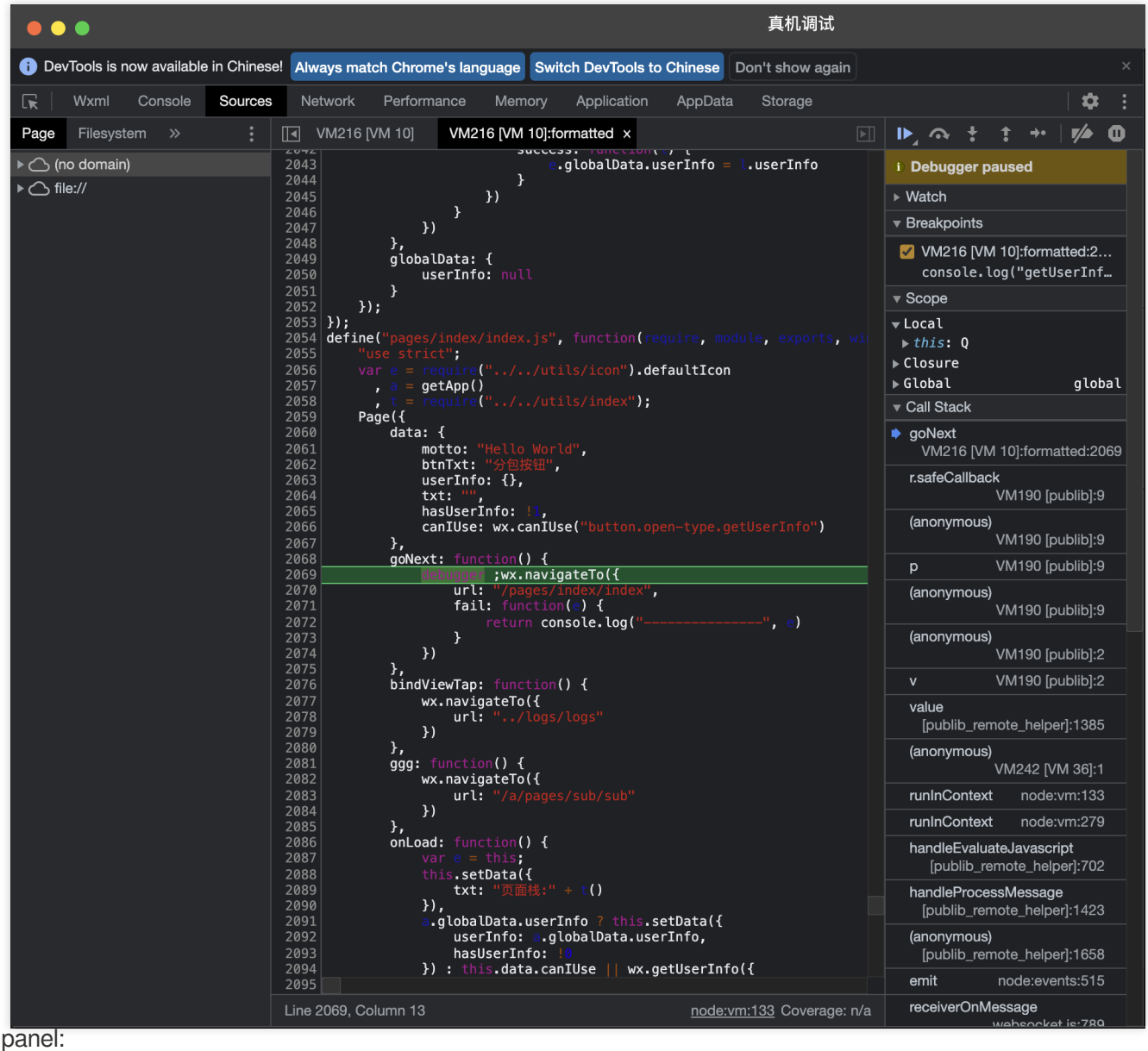
©2013-2022 Tencent Cloud. All rights reserved.



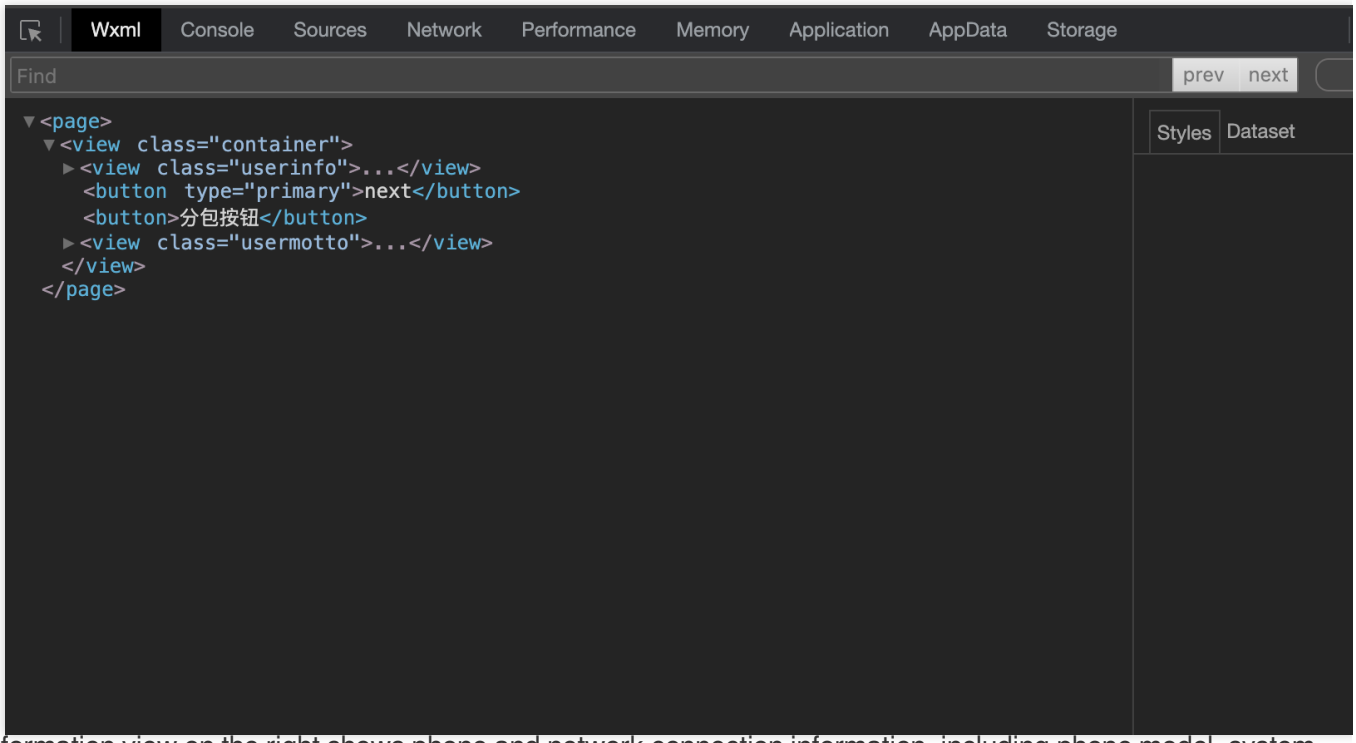
For JS breakpoint debugging, use the debugger in the code.

```
goNext(){
  debugger;
  wx.navigateTo({
    url:'/pages/index/index',
    fail: err => console.log("---------------",err),
  })
},
// event handler function
```

©2013-2022 Tencent Cloud. All rights reserved.


真机调试
DevTools is now available in Chinese!
Always match Chrome's language
Switch DevTools to Chinese
Don't show again
Wxml
Console
Sources
Network
Performance
Memory
Application
AppData
Storage
Page
Filesystem
(no domain)
file://
VM216 [VM 10]
VM216 [VM 10]:formatted
Debugger paused
Watch
Breakpoints
Scope
Local
Closure
Global
Call Stack
goNext
VM216 [VM 10]:formatted:2069
Line 2069, Column 13
node:vm:133 Coverage: n/a

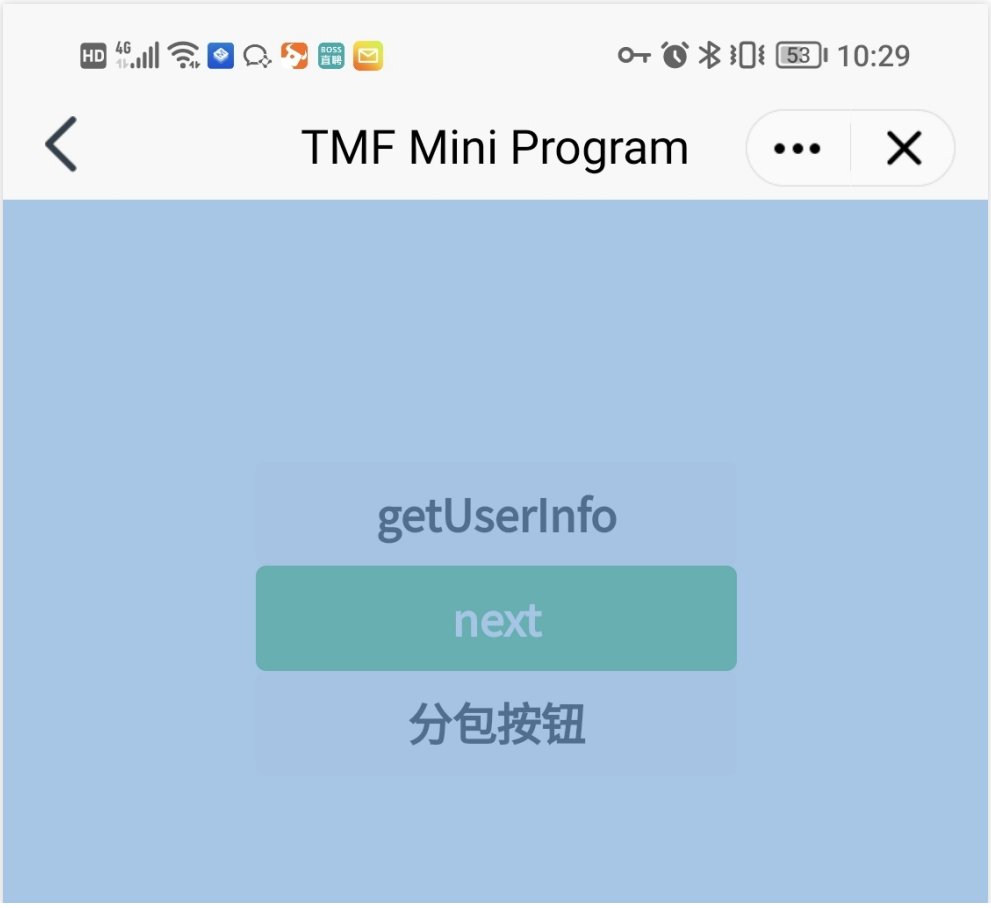
wxml panel:



The information view on the right shows phone and network connection information, including phone model, system, name, host app version, and communication latency. Lower latency means smoother communication with the phone.

Enable node inspection mode to click elements on the real device and view their details.

Display on mobile phones:



©2013-2022 Tencent Cloud. All rights reserved.



©2013-2022 Tencent Cloud. All rights reserved.

Additional information

Last updated：2024-07-12 16:34:06

Mini program code hardening

Feature overview

To enhance platform security, we address security issues in the mini program development process, such as code being easily decompiled, core business logic being deciphered, and algorithms being repackaged. These issues pose risks of the mini program being cracked and core code being stolen. The platform provides a code hardening feature to encrypt the front-end code of mini programs. This makes it difficult for attackers to read the front-end code logic, while developers should also maintain rigorous development standards to ensure code quality and protect the mini program's security. Code hardening is a feature that allows developers to encrypt the front-end code of mini programs to prevent code exposure.

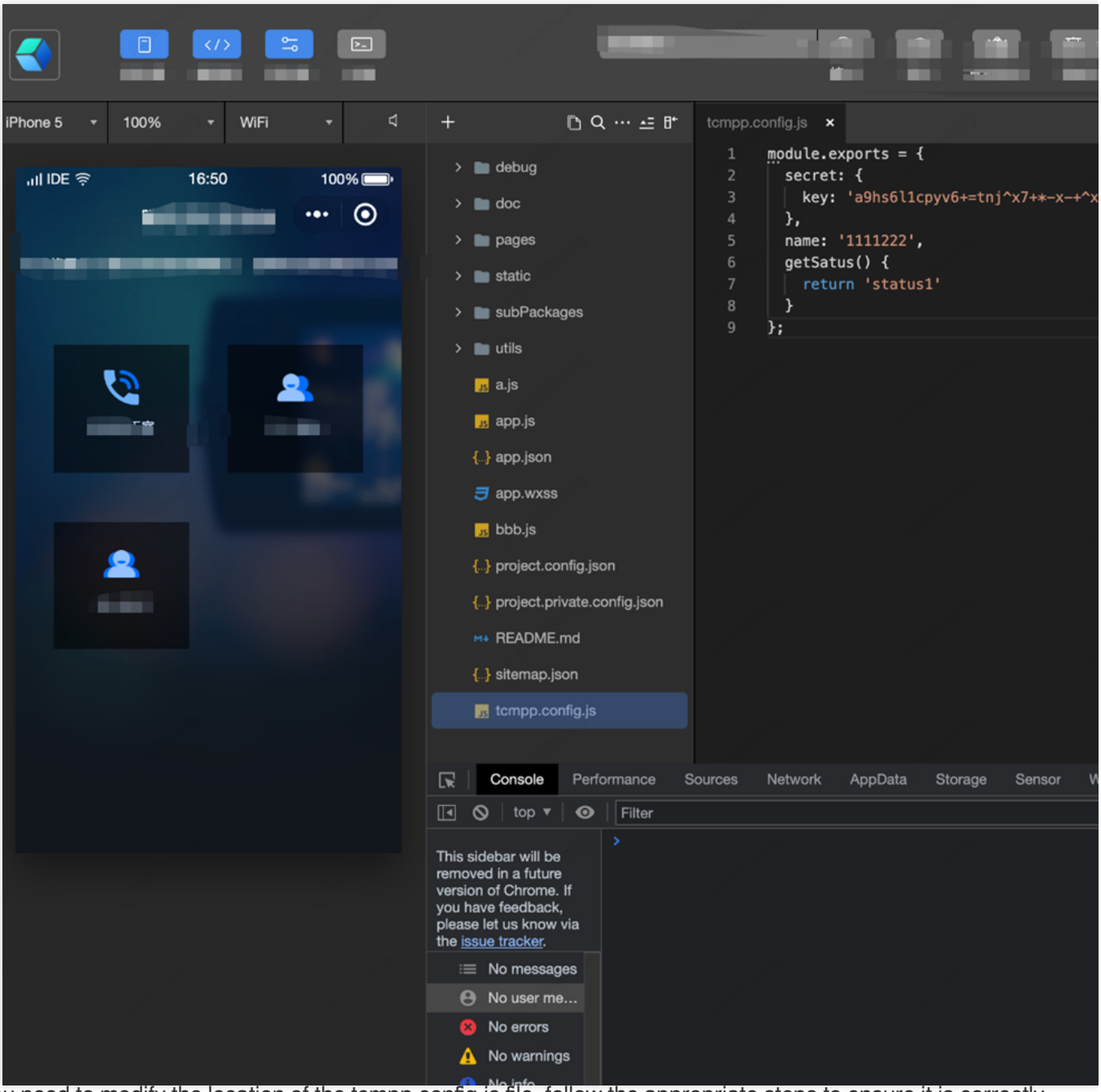
How to use

The code hardening feature allows JavaScript files to be passed to an encryption tool, enabling multiple protection measures such as string encryption, property encryption, call transformation, and code obfuscation. It is recommended to use code obfuscation for sensitive information in the mini program (such as SDK keys), core algorithm logic, version content, and other content that requires security protection.

Directions

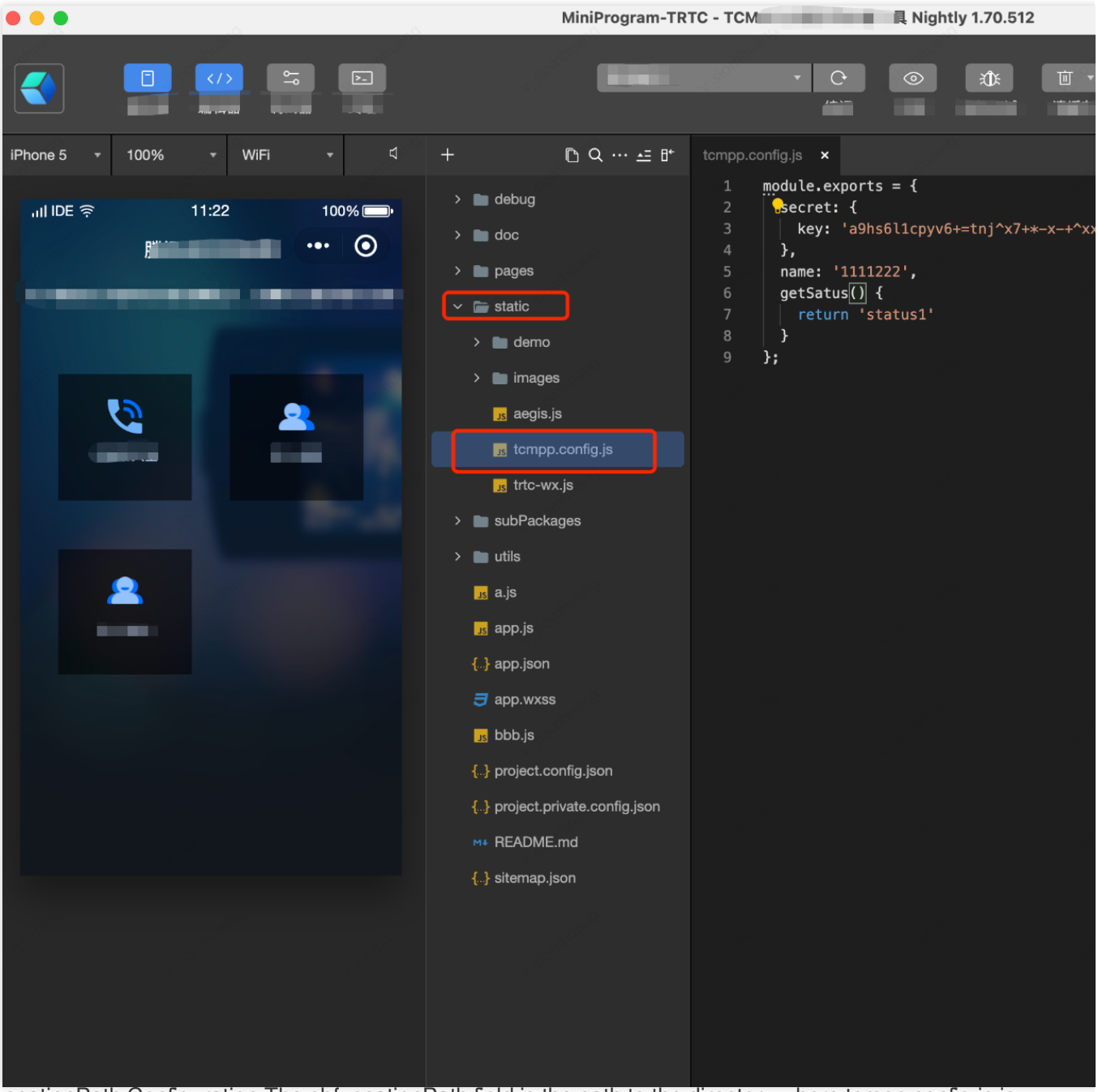
1. Create a tcmpp.config.js file in the same directory as app.json in the mini program project. You can write sensitive information or core algorithms in this JS file and export them for use. During the compilation phase, tcmpp.config.js will be obfuscated and encrypted.

©2013-2022 Tencent Cloud. All rights reserved.



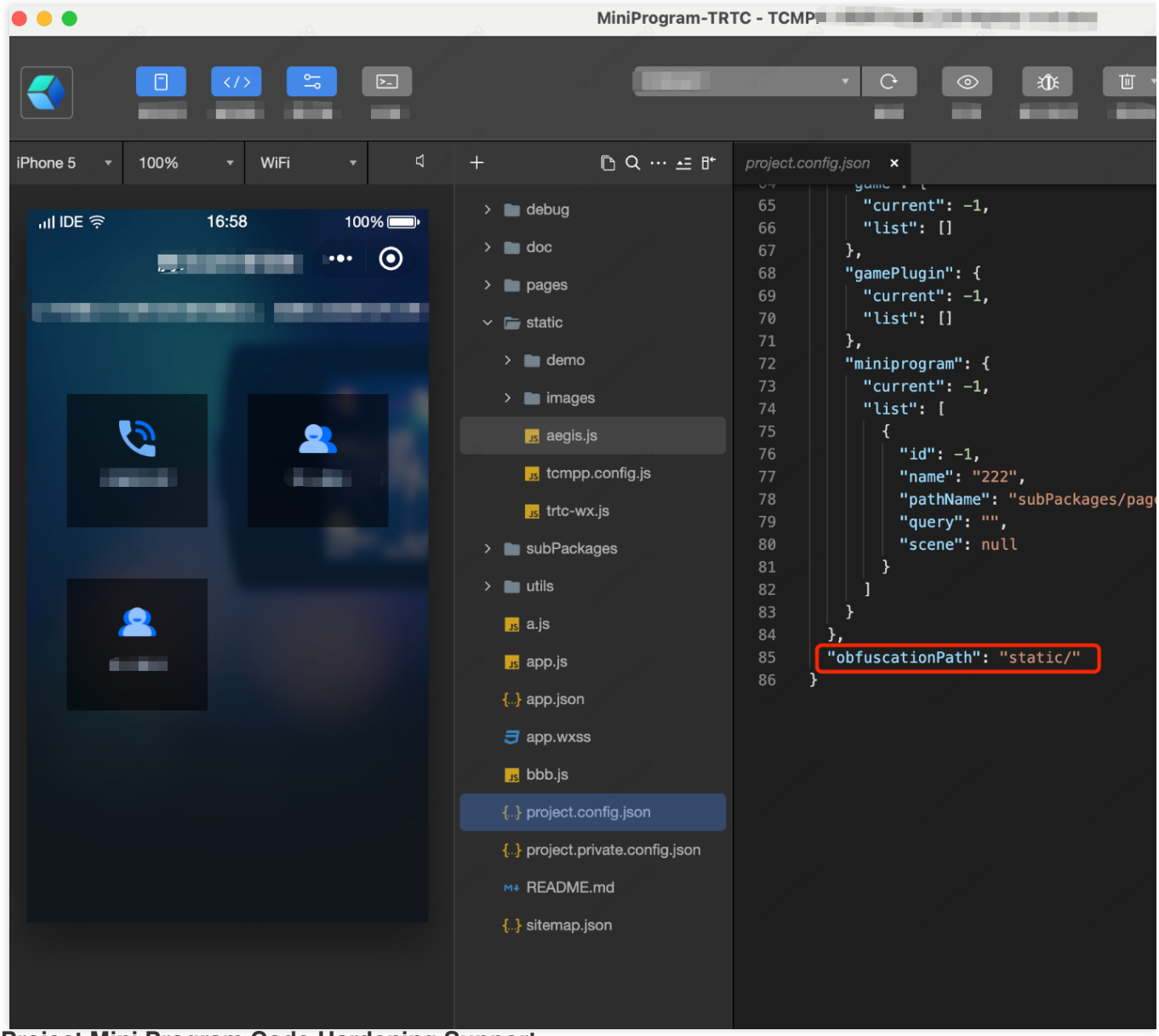
2. If you need to modify the location of the tcmpp.config.js file, follow the appropriate steps to ensure it is correctly referenced and encrypted during the build process.If you want to change the directory where tcmpp.config.js file is located, such as putting this file into a static-like directory, you can configure the path of the directory where tcmpp.config.js is located by using the obfuscationPath field in project.config.json. If you don't configure the obfuscationPath field or set it to an empty string, tcmpp.config.js will be found in the same directory as app.json by default.

©2013-2022 Tencent Cloud. All rights reserved.



3. obfuscationPath Configuration The obfuscationPath field is the path to the directory where tcmpp.config.js is located:

©2013-2022 Tencent Cloud. All rights reserved.



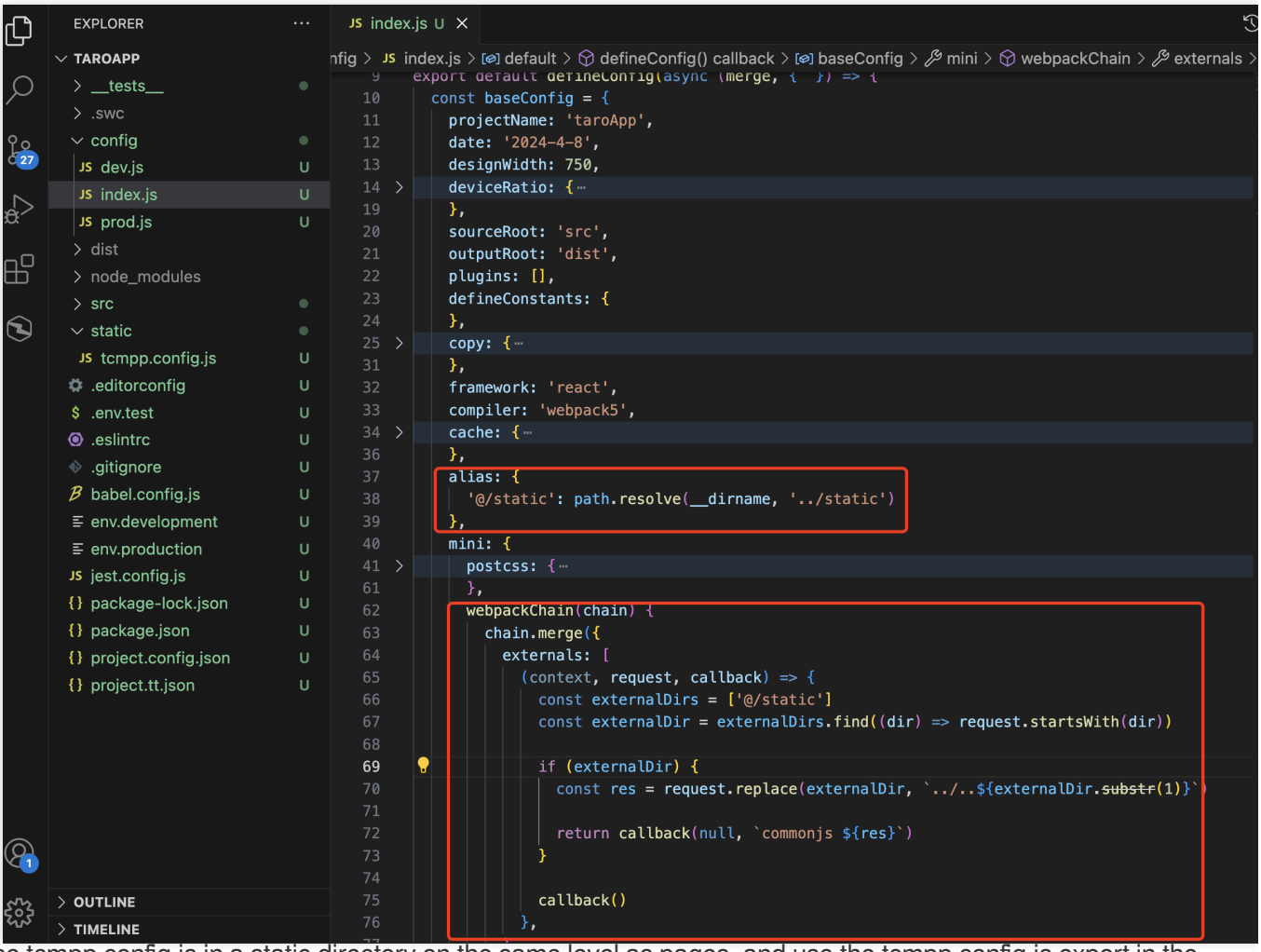
Taro Project Mini Program Code Hardening Support

1. In taro mini program, tcmpp.config.js needs to be packaged as an external native public js module, and tcmpp.config.js needs to be written in the same way as mini program js code, i.e., using module.exports to ensure that tcmpp.config.js is not packaged into taro's product but there is a separate tcmpp.config.js file in the dist, so that it can be obfuscated and encrypted in the compilation stage of tcmpp's IDE. After taro is packaged, tcmpp.config.js will not be packaged into the taro product, but a separate tcmpp.config.js file will exist in the dist, so that it can be obfuscated and encrypted during the compilation phase of tcmpp's IDE. Therefore, you need to configure tcmpp.config.js with the externals configured by the taro project. You can see the Taro project's JS file that references the native project.

2. Example of a hello world project configuration:

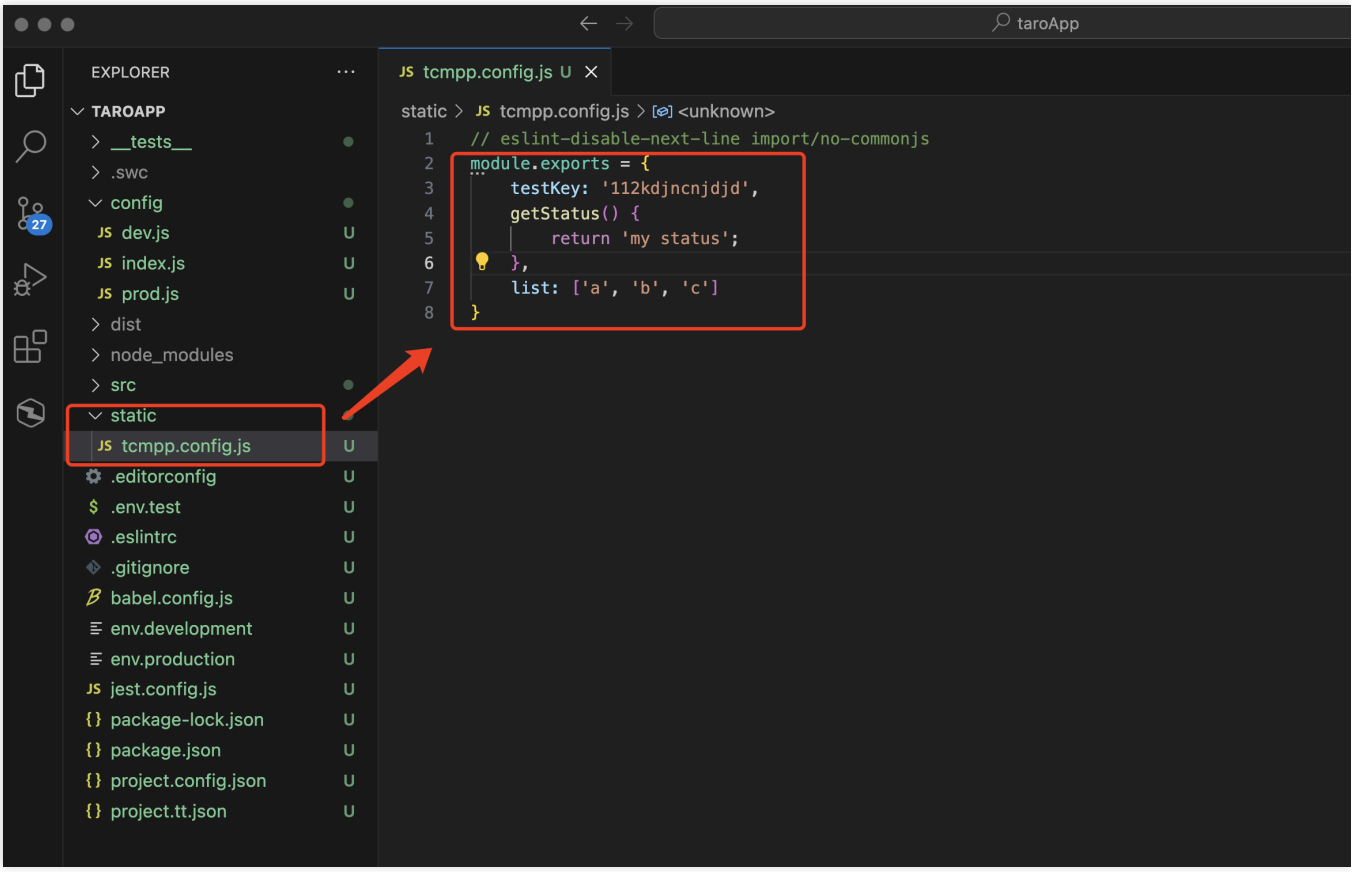
2.1 Configure the taro package, set the static directory to alias, and set the externals configuration:

©2013-2022 Tencent Cloud. All rights reserved.

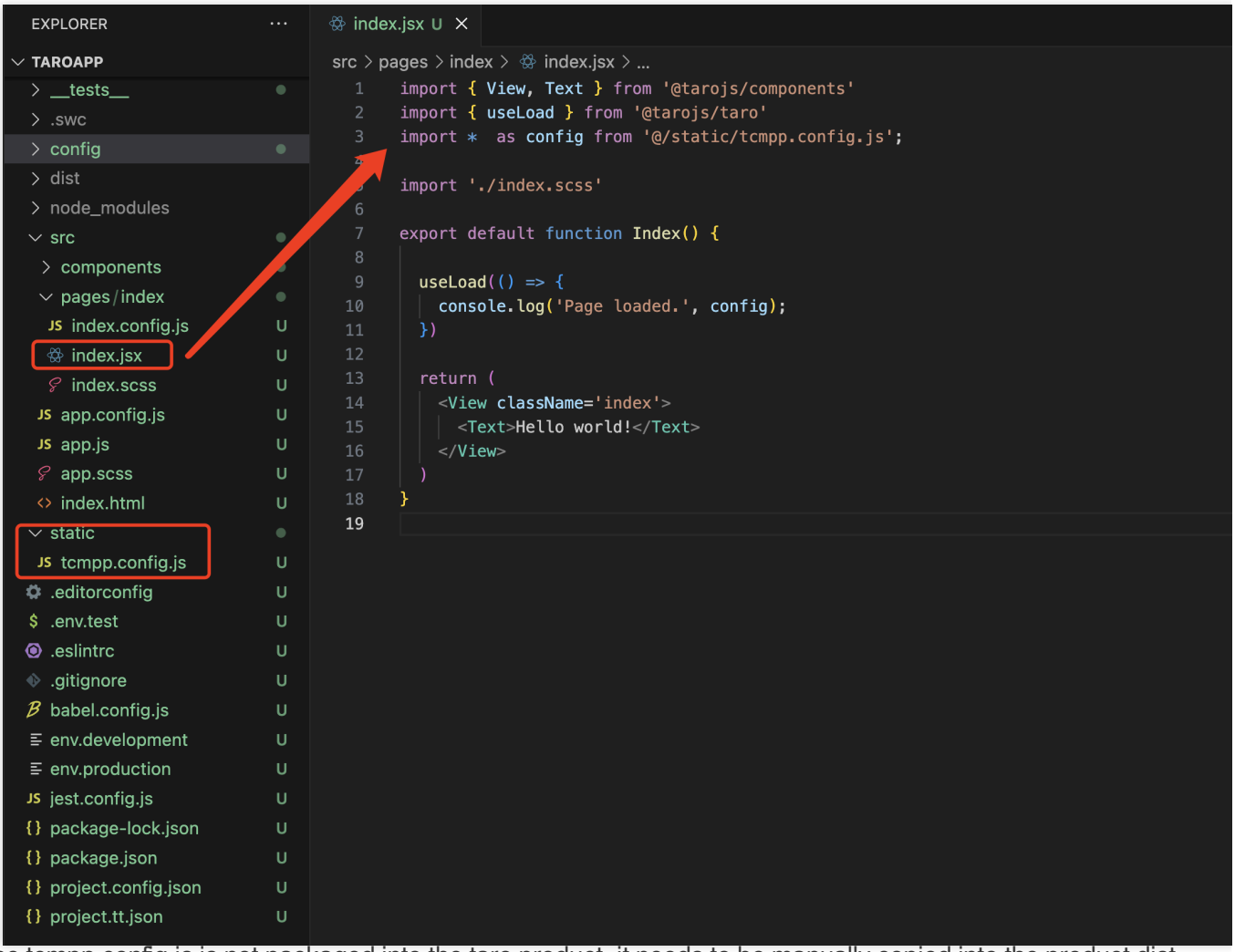


3. Place tcmpp.config.js in a static directory on the same level as pages, and use the tcmpp.config.js export in the project pages/index/index.jsx:

©2013-2022 Tencent Cloud. All rights reserved.

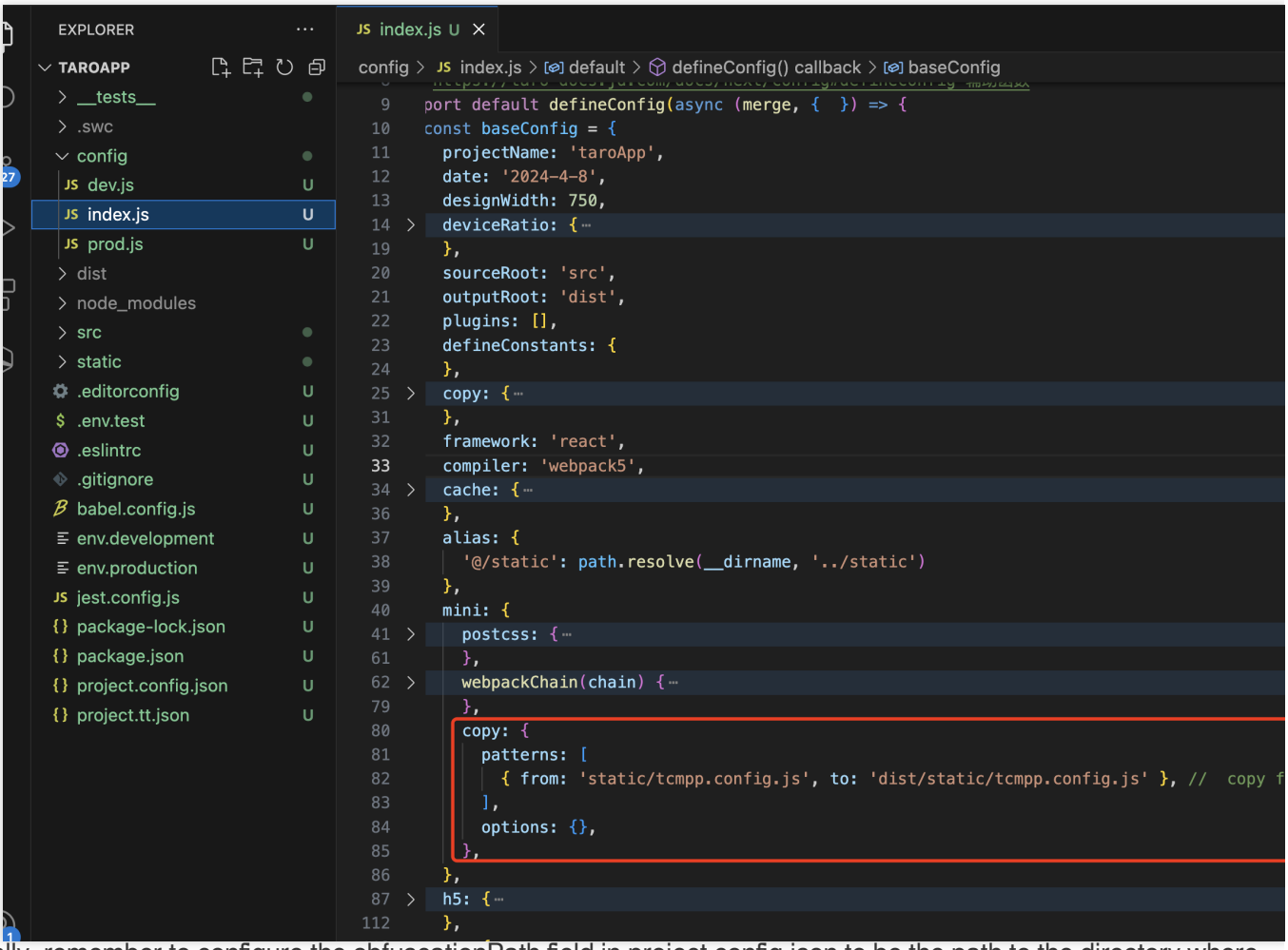


©2013-2022 Tencent Cloud. All rights reserved.



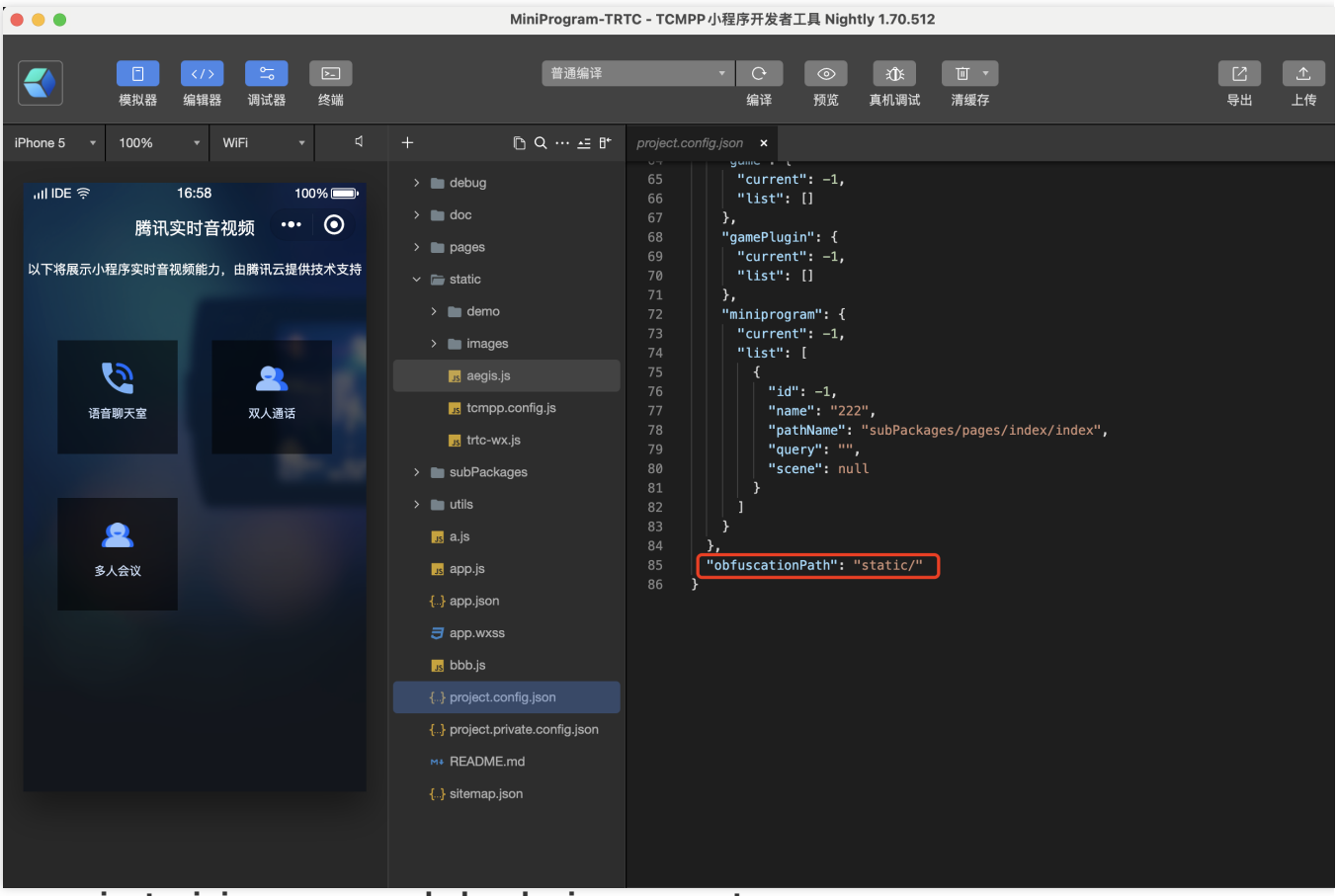
4. Since tcmpp.config.js is not packaged into the taro product, it needs to be manually copied into the product dist directory via webpack configuration:

©2013-2022 Tencent Cloud. All rights reserved.



5. Finally, remember to configure the obfuscationPath field in project.config.json to be the path to the directory where tcmpp.config.js is located so that the IDE can find this file for obfuscation.

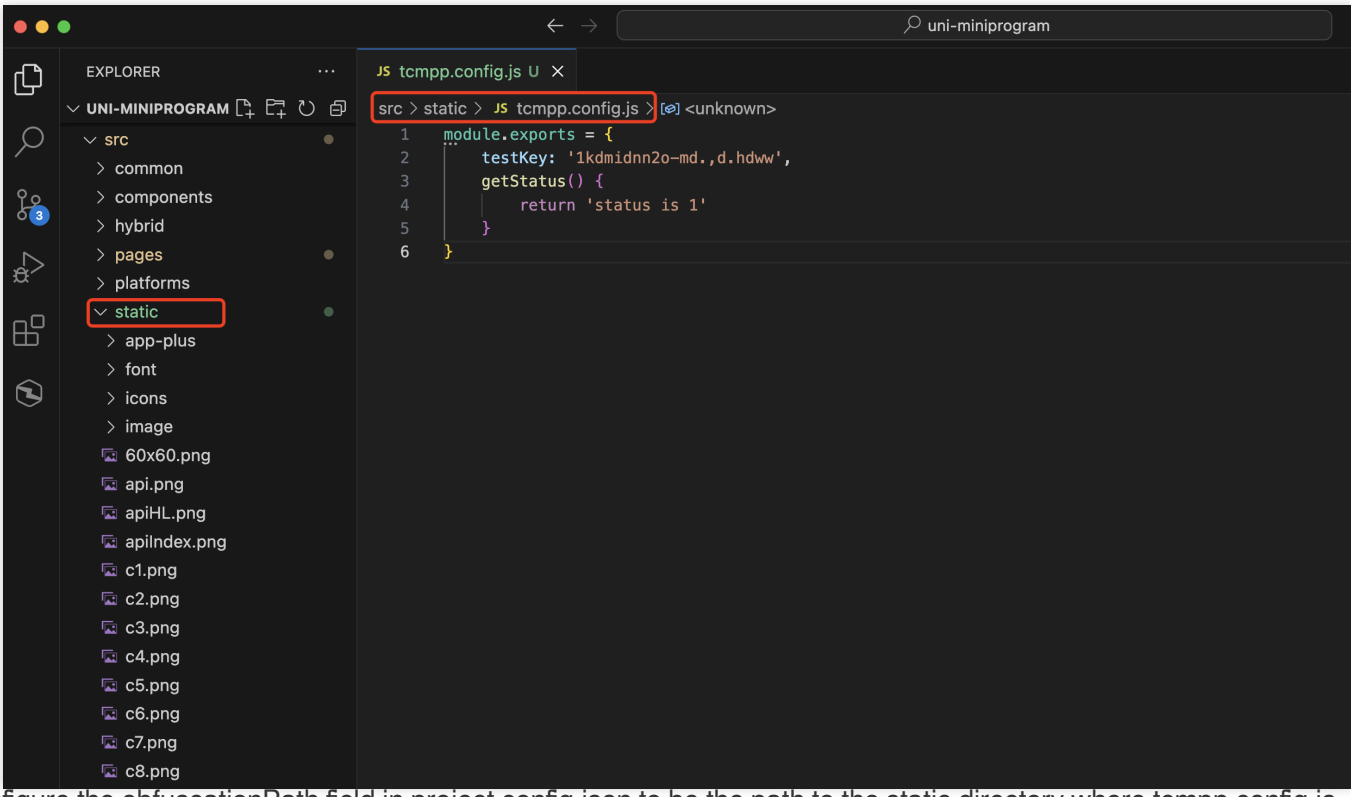
©2013-2022 Tencent Cloud. All rights reserved.



uni-app project mini program code hardening support

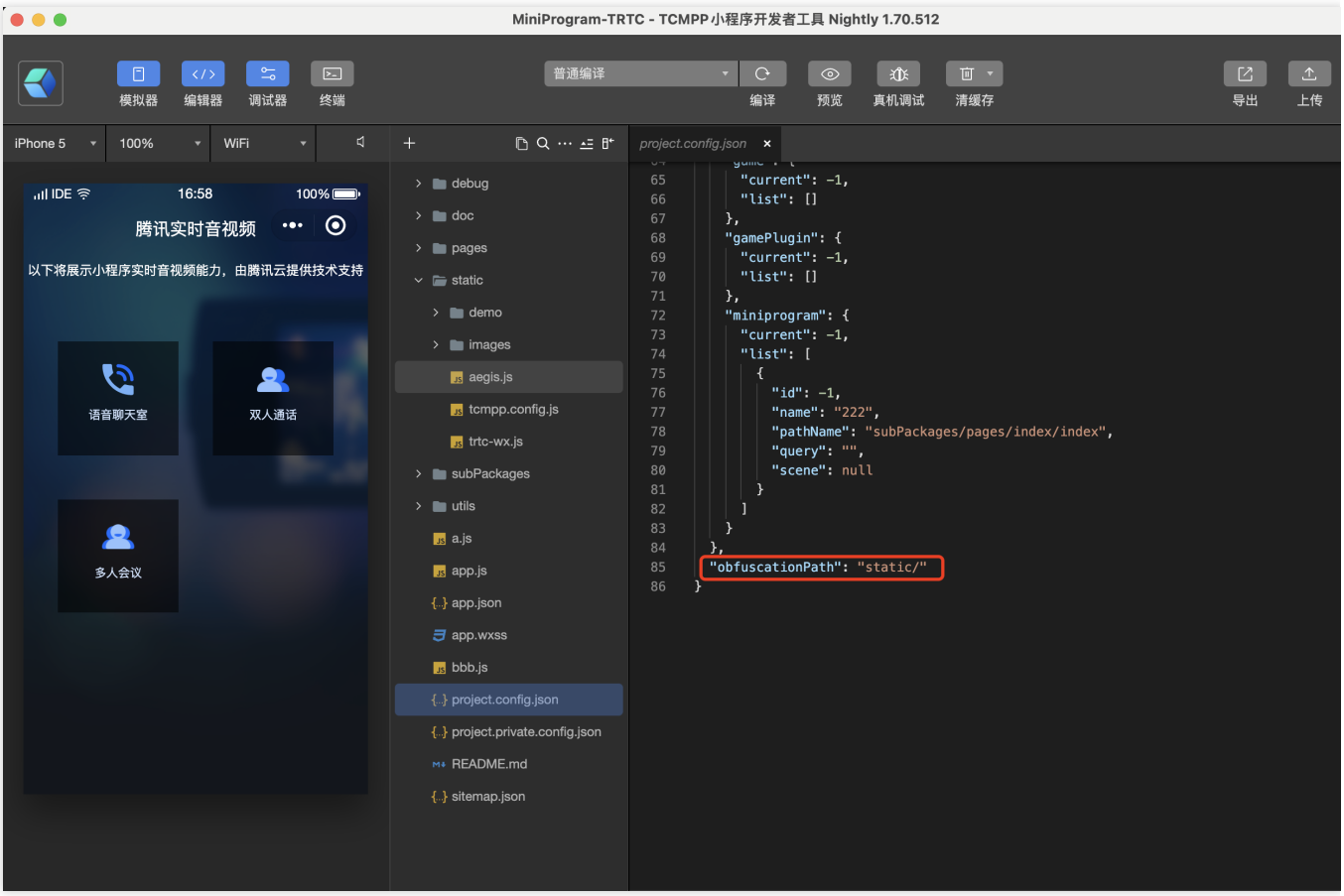
1. uni-app provides a static directory, which will be copied to the final package, see uni-app staticuni-app's static directory. Therefore, you need to put tcmpp.config.js into the static directory as well, and use module.exports to export the contents.

©2013-2022 Tencent Cloud. All rights reserved.



2. Configure the obfuscationPath field in project.config.json to be the path to the static directory where tcmpp.config.js is located so that the IDE can find this file for obfuscation.

©2013-2022 Tencent Cloud. All rights reserved.



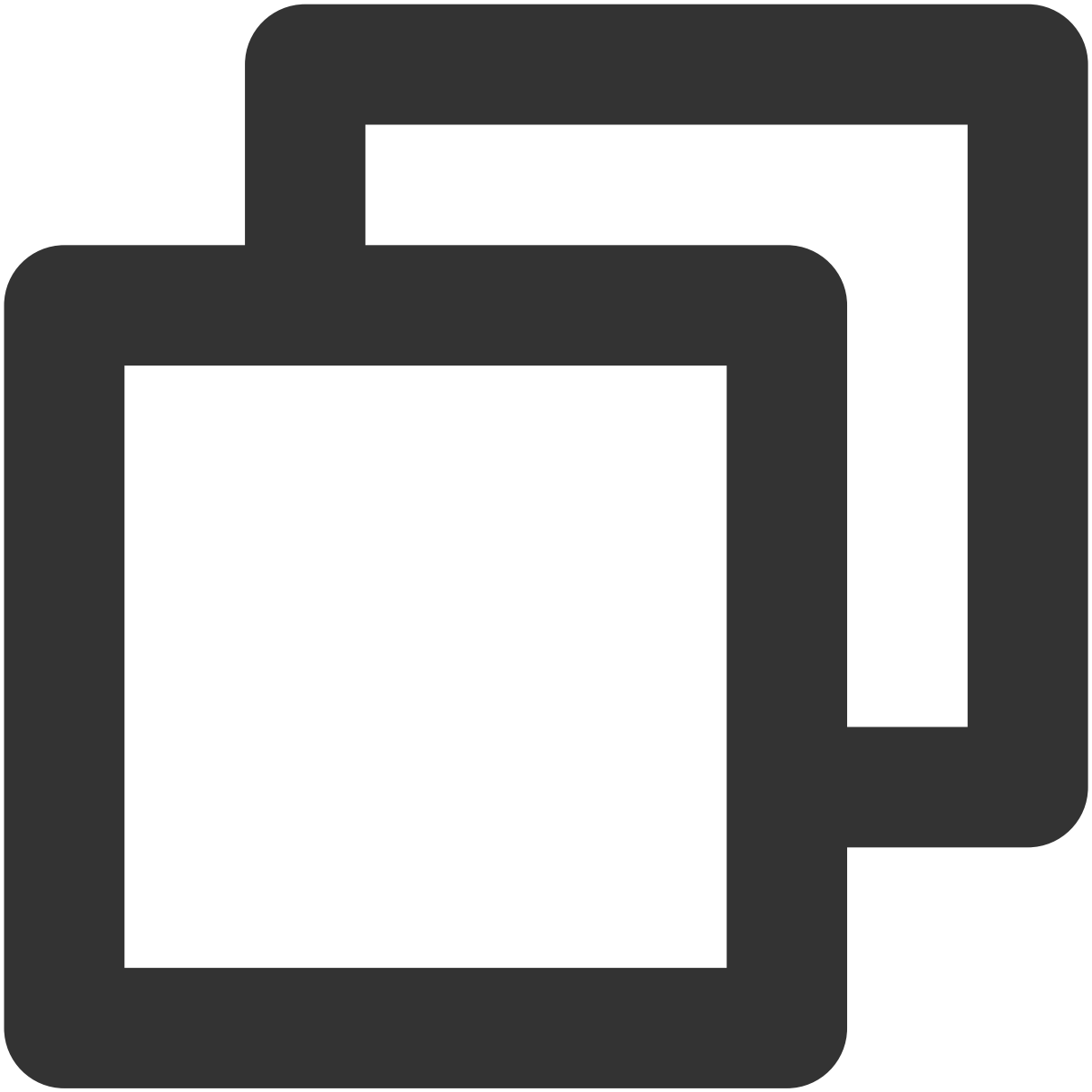
Using Google Maps in the IDE

Function Introduction

If Google Map-related functions are involved in the business development of the mini program, and you want to implement the preview in the IDE during the development phase, you need to configure the Google JavaScript API key in the project, for example, when using the map component or related APIs:

Components

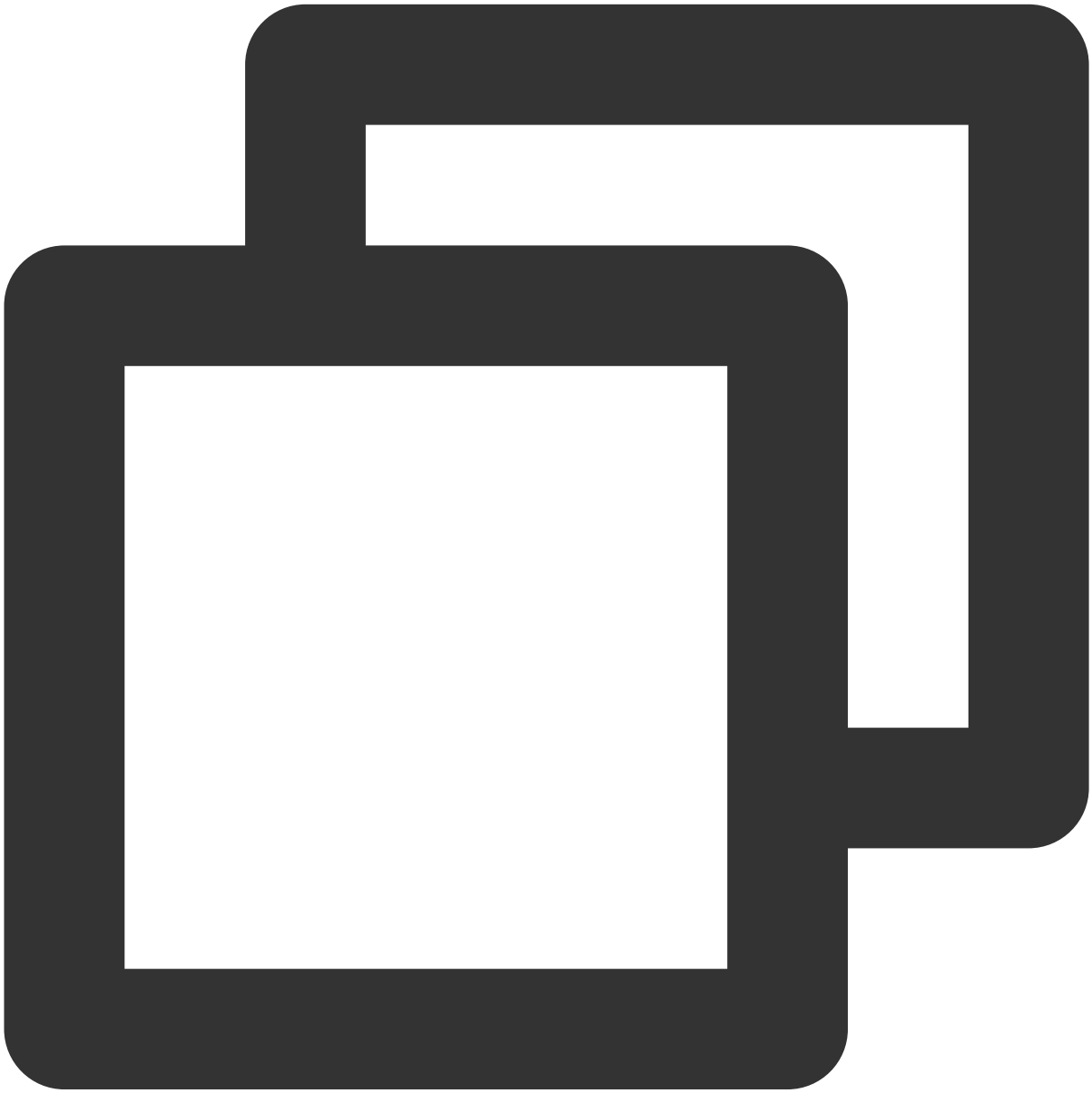
©2013-2022 Tencent Cloud. All rights reserved.



```
<map type='google' ...></map>
```

API

©2013-2022 Tencent Cloud. All rights reserved.



```
wx.openLocation({
  type: 'google',
  ...
})

wx.chooseLocation({
  type: 'google',
  ...
})
```

©2013-2022 Tencent Cloud. All rights reserved.

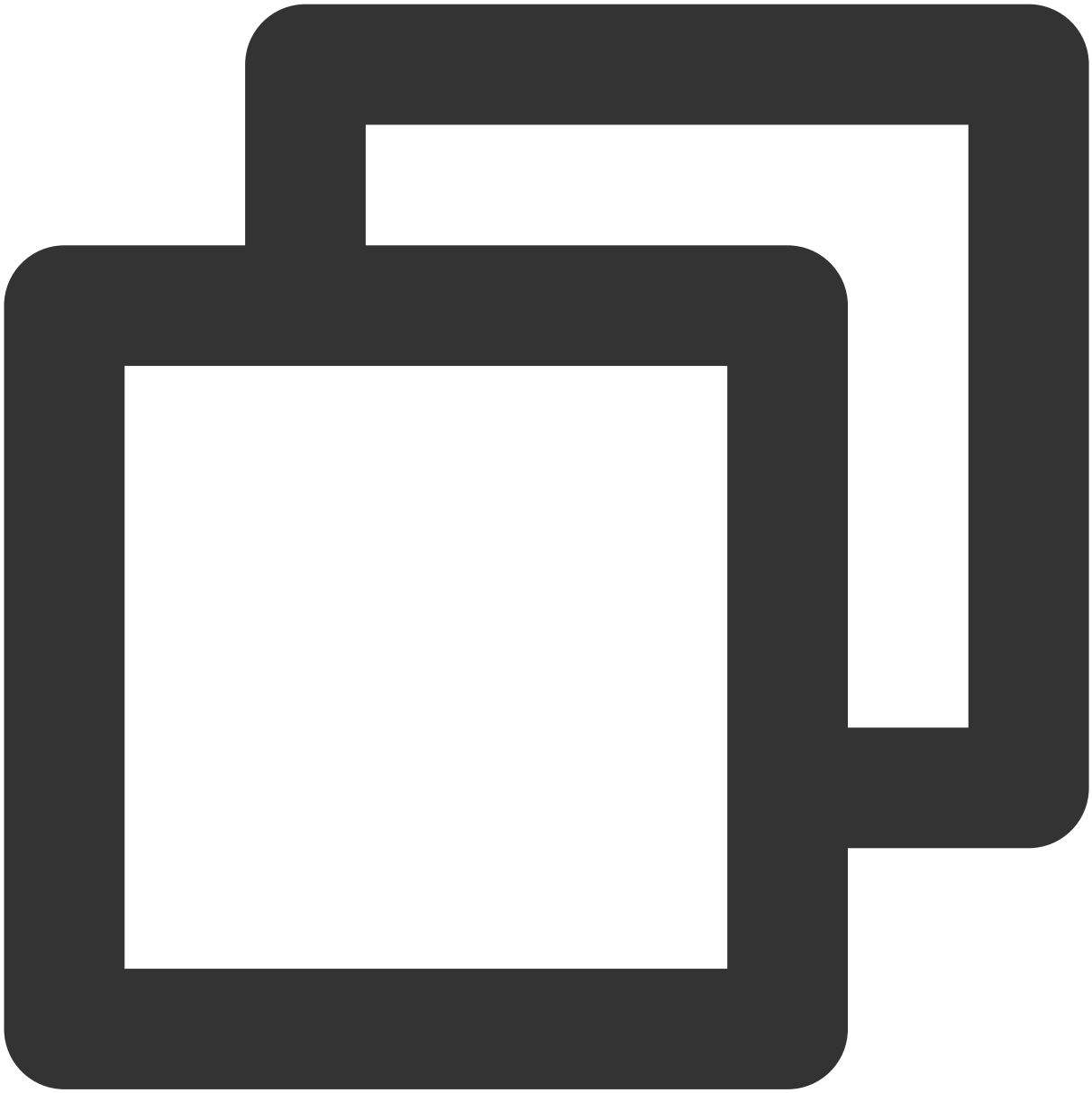
Introduction to Use

Considering that the user's mini-program business key will use the code hardening capability, in order to facilitate the user's unified management of the key, the Google JavaScript API key needs to be configured in the tcmpp.config.js file, and since the key is only used for IDE preview during the development phase, it is configured using annotations, and it will be configured during the final mini-program compilation and packaging.filtered when the app is finally compiled and packaged.

Usage Process

1. The process of creating the tcmpp.config.js file is the same as that of code hardening, please refer to the process of using the code hardening capability of the mini program.
2. Configure the Google JavaScript API key by annotation, the key key in the annotation must be googleMapApiKey and use [] to encapsulate the key, otherwise it will lead to key reading failure, the format is as follows:

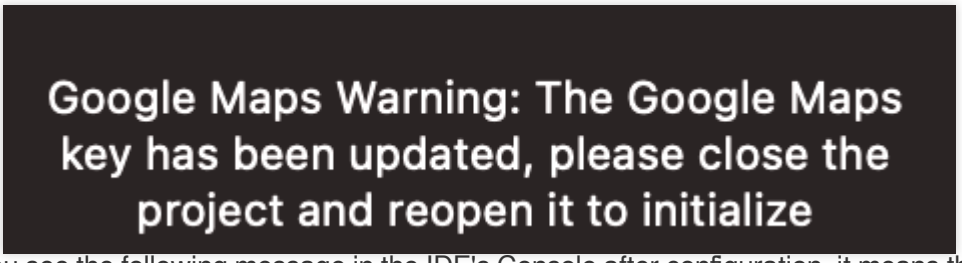
©2013-2022 Tencent Cloud. All rights reserved.



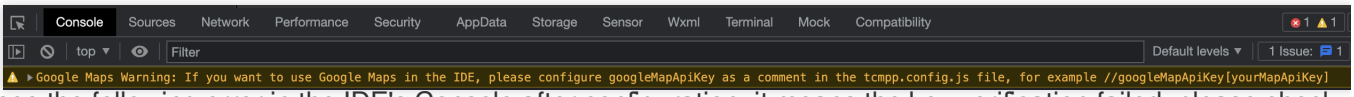
```
//googleMapApiKey[xxxxxxxxxxxxxxxxxxxxxxxxxxxx]
```

3. After editing the googleMapApiKey in tcmpp.config.js file in IDE, since you need to re-inject Google JavaScript API script, please close the project and reopen it, IDE will also give you the corresponding pop-up window: Google Maps Warning: The Google Maps key has been updated, please close the project and reopen it to initialize.

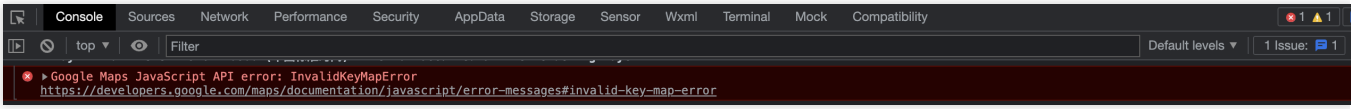
©2013-2022 Tencent Cloud. All rights reserved.



4. If you see the following message in the IDE's Console after configuration, it means the configuration of googleMapApiKey has not taken effect, please check whether the configuration follows the relevant format.



If you see the following error in the IDE's Console after configuration, it means the key verification failed, please check if you have configured the correct Google JavaScript API Key.



©2013-2022 Tencent Cloud. All rights reserved.