

IoT Explorer

Application Side Development

Guide



Copyright Notice

©2013–2025 Tencent Cloud. All rights reserved.

The complete copyright of this document, including all text, data, images, and other content, is solely and exclusively owned by Tencent Cloud Computing (Beijing) Co., Ltd. ("Tencent Cloud"); Without prior explicit written permission from Tencent Cloud, no entity shall reproduce, modify, use, plagiarize, or disseminate the entire or partial content of this document in any form. Such actions constitute an infringement of Tencent Cloud's copyright, and Tencent Cloud will take legal measures to pursue liability under the applicable laws.

Trademark Notice



This trademark and its related service trademarks are owned by Tencent Cloud Computing (Beijing) Co., Ltd. and its affiliated companies ("Tencent Cloud"). The trademarks of third parties mentioned in this document are the property of their respective owners under the applicable laws. Without the written permission of Tencent Cloud and the relevant trademark rights owners, no entity shall use, reproduce, modify, disseminate, or copy the trademarks as mentioned above in any way. Any such actions will constitute an infringement of Tencent Cloud's and the relevant owners' trademark rights, and Tencent Cloud will take legal measures to pursue liability under the applicable laws.

Service Notice

This document provides an overview of the as-is details of Tencent Cloud's products and services in their entirety or part. The descriptions of certain products and services may be subject to adjustments from time to time.

The commercial contract concluded by you and Tencent Cloud will provide the specific types of Tencent Cloud products and services you purchase and the service standards. Unless otherwise agreed upon by both parties, Tencent Cloud does not make any explicit or implied commitments or warranties regarding the content of this document.

Contact Us

We are committed to providing personalized pre-sales consultation and technical after-sale support. Don't hesitate to contact us at 4009100100 or 95716 for any inquiries or concerns.

Contents

Application Side Development Guide

Tencent Lianlian Custom H5 Development

Overview

Quick Start

Development Guide

Development and Debugging

Principle of H5 Panel

Calling H5 SDK Capabilities

Custom Bluetooth Protocol H5 Panel Development

Ultrafast Mode

H5 Custom Development SDK

Basic Parameters

Device management

User Management

Interface Components

Calling Mini Program Capacity

Application Side API

Event Monitoring

Bluetooth Module

ASR Speech Recognition

TRTC Audio and Video

Cloud Storage Service

Music Service

Underlying SDK Capabilities

Developing Chinese Domestic Brand Mini Programs and Apps

Application Side API

Application Side API Introduction

User Management

WeChat ID Registration and Login

User Logging Out

Distribution Network Management

Generating Wi-Fi Device Configuration Token

Query Distribution Network Token Status

User Binding Wi-Fi Device

Device management

- Query Weixin Authorization Invoice
- Obtaining Product Information
- User Deletes Device
- Getting User Bound Device List
- Get Current Device Status
- Modifying Device Name
- Obtain Device Details
- Retrieve Device Scale-Out Information
- Equipment Replacement Room
- Add Sub-Device to User'S Bound List
- Get the Sub-Device List of the Designated Gateway Device
- Get the Sub-Device List of the Specified Gateway Bound to the Family
- Bluetooth Data Reporting
- Bluetooth Device Sends Reply Messages for Device Behavior
- Bluetooth Device Reporting Event
- Network Configuration Binding Device
- Bind Sub-Device to Gateway
- Unbind Sub-Device Gateway

Device Control

- User Control Device
- Sync Calling Device Behavior
- Asynchronous Invocation of Device Behavior

Device Sharing

- Sending Device Sharing From the App
- Retrieve the Device Sharing Token
- Obtain Device Sharing Token Information
- Devices Shared by Bound Users
- Querying the User-Shared Device List
- Deleting User-Shared Devices
- Querying Device User List
- Deleting the User of Device Sharing

Family Management

- Creating a Family
- Deleting a Family
- Modifying Family
- Get Family List
- Getting Family Details
- Creating a Room

Modifying Room

Deleting a Room

Getting Room List

Invite Family Members

Member Joining Family

Deleting a Family Member

Member Quit Family

Get the Family Member List

Invite Family Members on the App

Device Scheduling

Creating Scheduled Tasks

Modifying a Scheduled Task

Deleting Scheduled Tasks

Modifying Scheduled Task Status

Get Scheduled Task List

Firmware Upgrade

Reporting Device Firmware Version

Query Device Upgradable Firmware Version

Query the Download Address of Device Firmware

Confirm Firmware Upgrade Task

Reporting Firmware Upgrade Task Status

Querying Device Firmware Upgrade Status

View Firmware Information

Message Management

Obtain Message List

Deleting Messages

Manual Intelligence

Create Manual Intelligent Linkage

Get Manual Intelligent Linkage List

Deleting Manual Intelligent Linkage

Modifying Manual Intelligent Linkage

Performing Manual Intelligent Linkage

Automatic Intelligence

Create Automatic Intelligent Linkage

Get Automatic Intelligent Linkage List

Getting Automatic Intelligent Linkage Details

Deleting Automatic Intelligent Linkage

Modifying Automatic Intelligent Linkage

Modify Automatic Intelligent Linkage Status

Data Query

Retrieve the Device Object Model Data

Retrieve Device Object Model Historical Data

Getting Historical Events of Devices

Long Connection Communication

Registering Listening

Heartbeat

Device Status Push

Cloud Storage Service

Pull Cloud Storage Event List

Pulling Cloud Storage Event Thumbnails

Get the Date with Cloud Storage

Get Cloud Storage Timeline on a Certain Day

Getting Video Anti-Hotlinking Playback URL

Pulling Image Streaming Data

Music Service

Song in a Song List

Retrieve the Song Playback Link

Daily Recommendation

Batch Querying Song Information

Recommended Playlist

New Song Release

User Logout

User Information

Application Side Development Guide

Tencent Lianlian Custom H5 Development Overview

Last updated: 2025-04-28 10:51:16

Tencent Lianlian H5 custom development refers to device manufacturers and solution providers managing and controlling intelligent products through the "Tencent Lianlian" mini program. Just perform interactive development of the manufacturer's personalized device control panel according to the Tencent Lianlian H5 custom development specification, and all capabilities provided by "Tencent Lianlian" can be used directly, thereby reducing the development cost of users developing a complete independent application.

WeChat mini programs reduce the threshold for consumers to use intelligent products by enabling use without downloading and instant use after scanning. Tencent Cloud IoT provides the official "Tencent Lianlian" mini program, allowing consumers to interconnect devices of different brands, categories, and communication methods, meeting the needs of manufacturers for personalization and sharing the Tencent Lianlian ecosystem.

Usage Reference for Development

- [Quick Start](#): Deploy and experience the H5 custom panel Demo following the guidance process in the document.
- [Development debugging call](#)
- [H5 Custom Development SDK](#)

Quick Start

Last updated: 2025-04-28 10:51:39

Tencent Cloud IoT Explorer provides developers with an H5 custom panel demo. Developers can deploy and experience the H5 custom panel demo following the guidance process in this document. For more information about the demo, please see [H5 Panel Demo](#).

Prerequisites

Sign up for a Tencent Cloud account and complete Identity Verification.

Procedure 1: Create a Product and Devices

This step guides you to create a smart light product and device through the quick start feature of IoT Explorer, and connect the simulated device to the platform for data reporting. If you need to create products and devices by yourself, please refer to [Product Development](#).

1. Log in to Tencent Cloud and enter the [IoT Explorer console](#), select the **public instance** or the **standard enterprise instance** you have purchased, and enter the **product development**.
2. Refer to [Beginner Instruction](#) to create products and devices, and successfully integrate the devices into IoT Explorer and report data.

Procedure 2: Configure Custom H5 Panel

1. Download [JS files](#) and [CSS file](#) of the H5 panel.
2. Log in to Tencent Cloud and enter the [IoT Explorer console](#), select the project created in [Step 1](#), and enter the project details page.
3. Select the product created in [Step 1](#) from the product list and enter the product detail page.
4. Select **Interactive Development** > **Panel Configuration**, click **Configure** to enter the panel configuration page.

✓ 物模型定义 > ✓ 设备开发 > **3 交互开发** > 4 设备调试 > 5 批量投产

如果您的产品需要接入腾讯连连官方小程序，请开启“接入腾讯连连官方小程序”，接入腾讯连连平台会在批量投产时对交互开发面板配置、产品功能进行审核认证。

接入腾讯连连官方小程序 ☒

您选择使用平台的官方小程序控制产品，您可在下方配置产品在腾讯连连小程序中的展示效果。您可通过 [腾讯连连小程序二维码](#) 前往小程序

产品展示配置

您可以自定义产品在腾讯连连首页-设备列表中展示的产品图标和默认名称

快捷入口配置

您可以自定义产品在设备列表-快捷功能区域显示的快捷操作

面板配置

您可以自定义产品控制面板的风格、布局、按钮样式等配置

配网引导

您可以自定义产品的配网图文，使产品配网流程引导更明了、有效、符合产品的实际情况

扫一扫产品介绍

您可以自定义用户在使用微信扫一扫添加设备时的产品介绍页

智能联动配置

您可以自定义用户在添加智能时，该产品可作为条件或任务的功能属性

配置

配置

配置

配置

配置

配置

上一步

下一步

5. In the dashboard type, select "H5 Custom Dashboard".



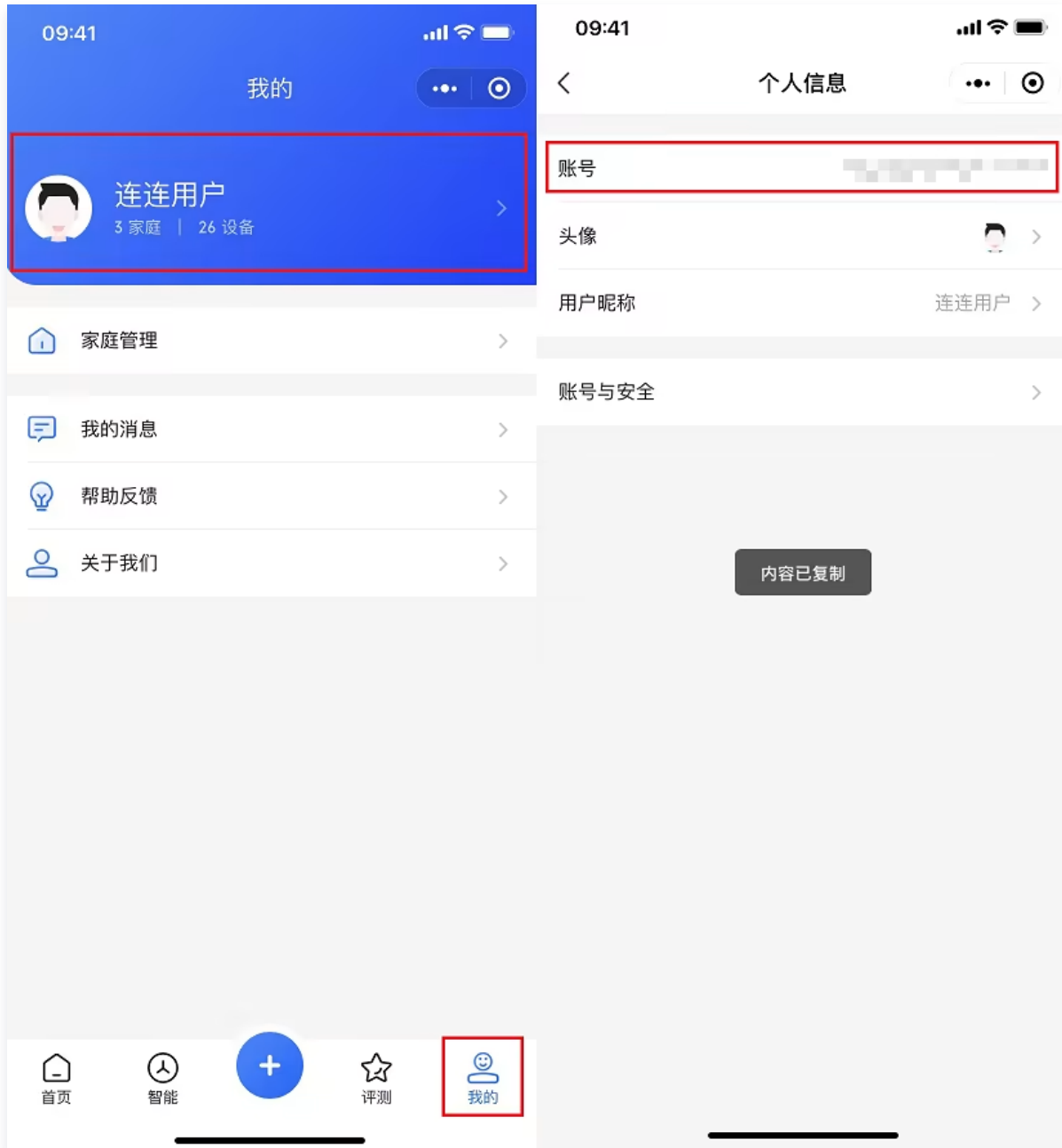
6. In the "Upload JS" form, click **Upload** and select the JS file of the H5 dashboard. After upload completion, the page displays the file name, size, and MD5 info of the JS file.
7. In the "Upload CSS" form, click **Upload** and select the CSS file of the H5 dashboard. After upload completion, the page displays the file name, size, and MD5 info of the CSS file.



Procedure 3: Add Lianlian Users to the H5 Panel Access Allowlist

During the development phase, you need to set the access allowlist to use Tencent Lianlian to debug the H5 custom panel. The operation steps to set the access allowlist are as follows:

1. Open the Tencent Lianlian Mini Program.
2. Click **Mine**, then click your personal information at the top of the page to enter the personal information page.
3. Long-press **Account** to copy the Tencent Lianlian user ID. After copy success, the page prompts that the content has been copied.



4. Log in to the [IoT Explorer console](#), and select the corresponding project and product.
5. Click **Interactive Development > Configure Mini Program > Panel Configuration > Configuration** in the product to enter the panel configuration page.

物模型定义

设备开发

3 交互开发

4 设备调试

5 批量投产

如果您的产品需要接入腾讯连连官方小程序，请开启“接入腾讯连连官方小程序”，接入腾讯连连平台会在批量投产时对交互开发面板配置、产品功能进行审核认证。

接入腾讯连连官方小程序 ☒

您选择使用平台的官方小程序控制产品，您可在下方配置产品在腾讯连连小程序中的展示效果。您可以通过 [腾讯连连小程序二维码](#) 前往小程序

产品展示配置

您可以自定义产品在腾讯连连首页-设备列表中展示的产品图标和默认名称

配置

快捷入口配置

您可以自定义产品在设备列表-快捷功能区域显示的快捷操作

配置

面板配置

您可以自定义产品控制面板的风格、布局、按钮样式等配置

配置

配网引导

您可以自定义产品的配网图文，使产品配网流程引导更明了、有效、符合产品的实际情况

配置

扫一扫产品介绍

您可以自定义用户在使用微信扫一扫添加设备时的产品介绍页

配置

智能联动配置

您可以自定义用户在添加智能时，该产品可作为条件或任务的功能属性

配置

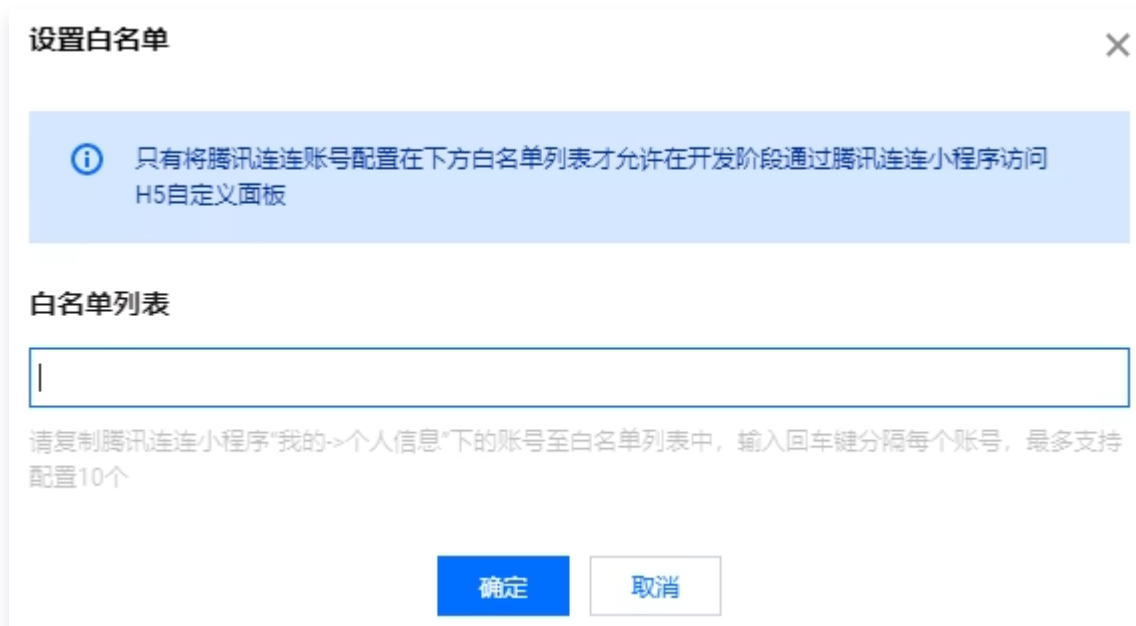
上一步

下一步

6. Select the "Dashboard Type" as "Custom H5 Dashboard", click **Set up allowlist**, and enter the allowlist setting page.



7. In the allowlist, input the Tencent Lianlian user ID copied from the above steps.



8. Click **Confirm**. The page prompt shows that the saving succeeded.

9. After closing the pop-up of "Set up allowlist", click **Save** on the configuration page.

Procedure 4: Bind Device in Mini Program

Virtual Device

1. On the product details page, select **Device Debugging** > **Virtual Device Debugging** to enter the virtual device debugging page. The page will display the QR code for virtual device debugging.



2. Use WeChat Scan to scan this QR code. WeChat will navigate to Tencent Lianlian Mini Program. After authorization completed, Tencent Lianlian Mini Program will automatically bind the device. The page prompt shows that the device has been added successfully.

Real Device

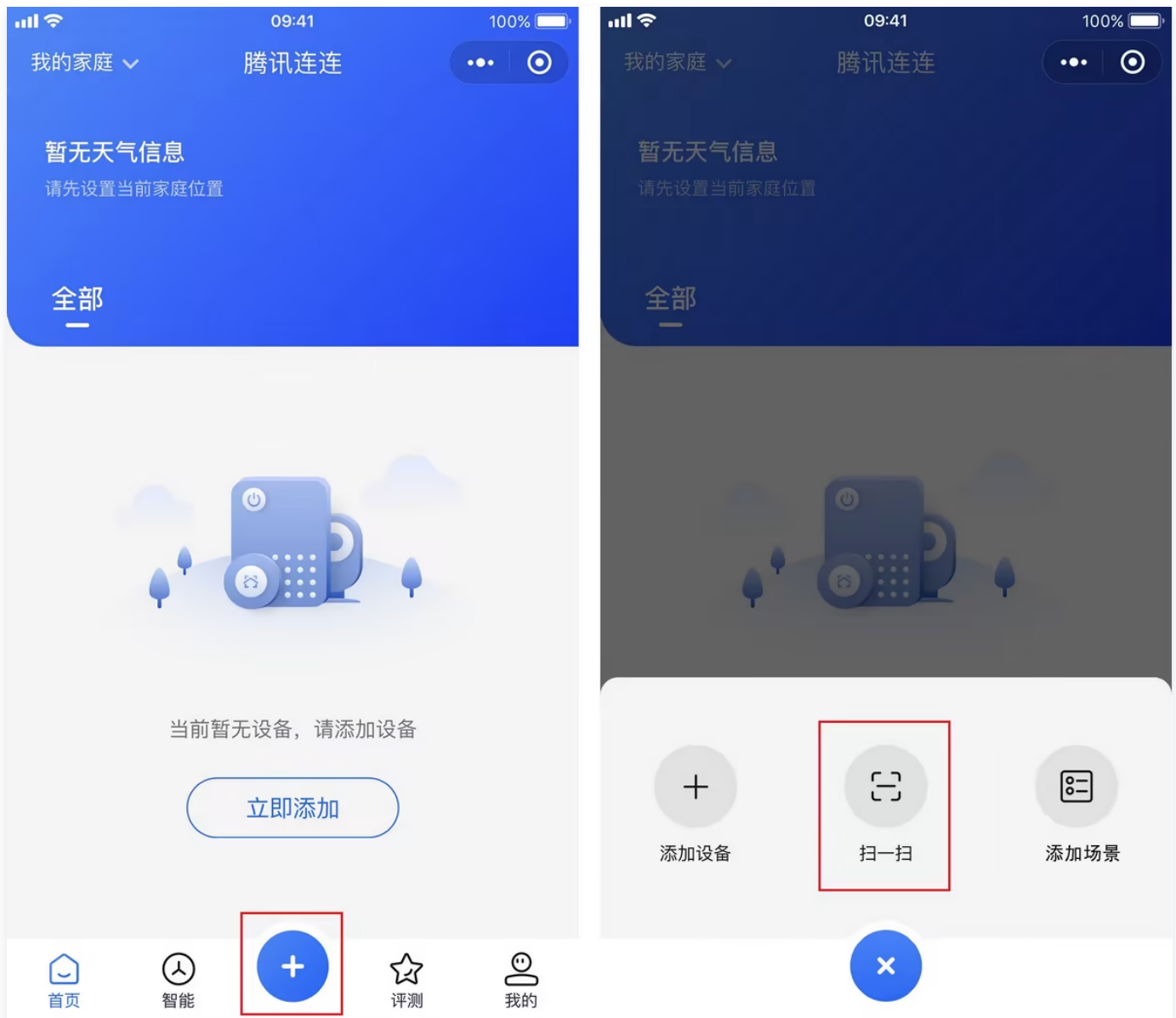
1. On the product details page, click **Device Debugging** to enter the device debugging page. The page displays the device list.
2. In the device list, select the device created in [Step 1](#) and click **QR Code**. The page displays the device binding QR code.



3. Use WeChat Scan to scan the mini program code below and open Tencent Lianlian Mini Program.



4. 1. Select "+" > **scan**, and scan the QR code shown on the console page. After the scan is successful, Tencent Lianlian Mini Program will automatically bind the device. The page prompt shows that the device has been added successfully.



Procedure 4: Experience Custom H5 Panel

In the Tencent Lianlian Mini Program, select the device bound in [Step 3](#) and enter the device control page. The page displays a custom H5 panel, where you can perform device control

and other operations.



Development Guide

Development and Debugging

Last updated: 2025-04-28 10:52:07

H5 Panel Scaffolding

We can use the following command to initialize an H5 panel project:

```
npx --package caz@1.0.0 caz tencentyun/iotexplorer-h5-panel-template my-h5-panel
```

If you have used tools like `vue-cli` to initialize a project, then this command is similar to it. This command will pull [iotexplorer-h5-panel-template](#) as a template, and then perform some feature selections, such as version number, choice of framework, etc. As shown below, finally, it will initialize an H5 panel project.

```
✓ Project name ... my-h5-panel
✓ Project version ... 0.1.0
✓ Project description ... Awesome h5-panel apps.
✓ 请选一个框架 > vue 3
✓ Choose the features you need >
✓ Install dependencies ... no
Created a new panel project in my-h5-panel by the tencentyun/iotexplorer-h5-panel-template template.

Getting Started:
$ cd my-h5-panel
$ npm install # or yarn
$ npm run dev -- --productId=your_product_id --deviceName=your_devicename # deviceName 参数可选
```

After the project creation is completed, developers can refer to `README.md` in the project directory. After installing dependencies, run the project through the following command:

```
npm run dev -- --productId=your_productId
```

Among them, `your_productId` is the `productId` of the corresponding product of the H5 panel.

H5 Panel Demo

IoT Explorer provides an H5 panel Demo for developers' reference. The H5 panel Demo is open to developers in an open-source way, making it easy for developers to refer to when developing H5 panels.

The H5 panel Demo project is written using the React framework. For details, see [iotexplorer-h5-panel-demo](#).

We also provide a scaffolding tool that supports the creation of H5 panel demos with React or Vue frameworks. For details, see [iotexplorer-H5-panel-template](#).

Starting Debugging

During the development and debugging of the H5 panel, execute the following commands.

1. Install dependencies (execute once after downloading the Demo).

```
npm install
```

2. Real-time compilation H5 panel (needs to keep running during the development process).

```
npm run dev
```

When this command is executed, the JS file of the H5 panel can be accessed via `http://127.0.0.1:9000/index.js`, and the CSS file can be accessed via `http://127.0.0.1:9000/index.css`. After the H5 panel source code file is changed, the JS file and CSS file will be automatically recompiled. Refresh the H5 panel page to load the updated JS file and CSS file.

Compilation and Building

After the H5 panel development is completed, execute the following commands to build the JS file and CSS file of the H5 panel.

```
npm run release
```

After the build is complete, the JS file and CSS file will be output to `dist/index.js` and `dist/index.css` respectively. Developers need to upload these files to the interactive development configuration item of the product in the [IoT Explorer Console](#). For operations, see [Upload H5 Panel](#).

Debugging the H5 Panel

Use the [official account webpage debug](#) feature of a browser or WeChat Developer Tools to access the specified URL, accessible in H5 panel development mode, to debug the H5 panel. In development mode, the H5 panel framework will load the JS file and CSS file from the following two URLs respectively. You need to configure a proxy so that the browser or WeChat Developer Tools can load the JS file and CSS file of the H5 panel through these two URLs, thereby debugging the H5 panel.

- JS file URL: `https://developing.script/developing.js`

- **CSS file URL:** `https://developing.style/developing.css`

Note:

Development mode requires your Tencent Cloud login status. You must use a Tencent Cloud account with permission to operate on the product to log in and enter the H5 panel development mode of the corresponding product.

Local Debugging

Local debugging will register a Tencent Lianlian user associated with your Tencent Cloud account as a Debug User, and create a family and a device for debugging, so you no need to pay attention to the logic of creating a family, device binding, etc., just focus on the development of the H5 panel.

Access the following URL to perform local debugging of the H5 panel. Replace ABCDEFG with the actual product ID.

```
https://iot.cloud.tencent.com/h5panel/developing?productId=ABCDEFG
```

The following describes the detailed directions by taking the local debugging of the H5 panel Demo with WeChat Developer Tools as an example.

Step 1: Download a Demo Project

1. Pull the source code of the Demo project.

```
git clone https://github.com/tencentyun/iotexplorer-h5-panel-demo.git
```

2. Install dependencies.

```
cd iotexplorer-h5-panel-demo
npm install
```

Step 2: Install and Configure whistle

If you are not familiar with whistle, you can also try an easier [proxy-free mode](#) to skip step 2.

1. Install whistle, execute the following command:

```
npm install -g whistle
```


2. Start up whistle, execute the following command:

```
w2 start
```

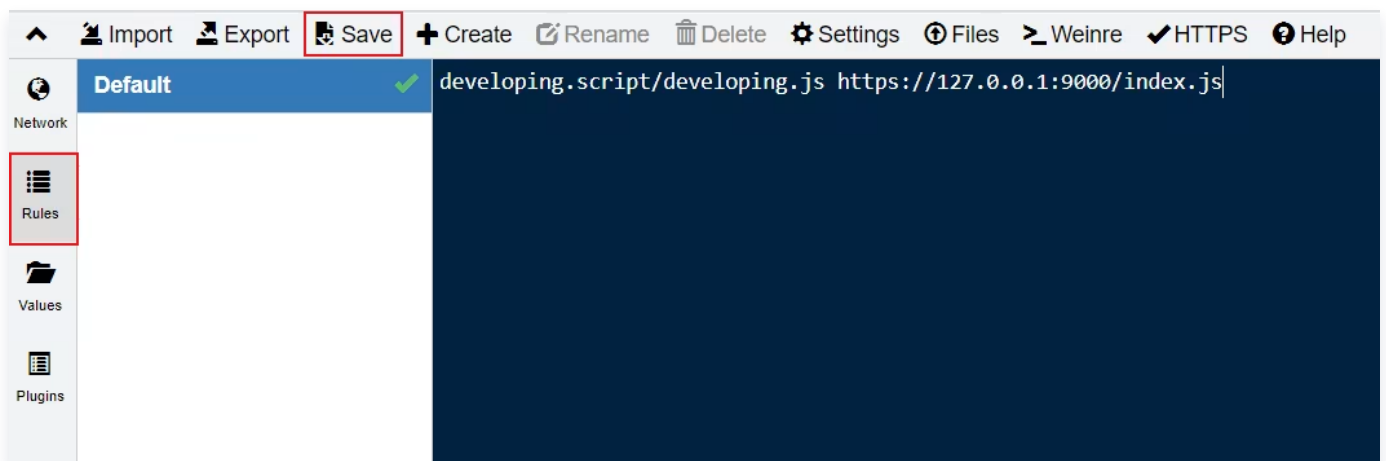
After whistle starts successfully, the command line has the following output:

```
[i] whistle@2.5.7 started
[i] 1. use your device to visit the following URL list, gets the IP
of the URL you can access:
    http://127.0.0.1:8899/
... (output omitted)
```

3. Open `http://127.0.0.1:8899/` through a browser and enter the user interface of whistle.
4. Select **Rules** > **Default** and enter the rule configuration page. Fill in the following proxy rules in the input box.

```
developing.script/developing.js http://127.0.0.1:9000/index.js
developing.style/developing.css http://127.0.0.1:9000/index.css
```

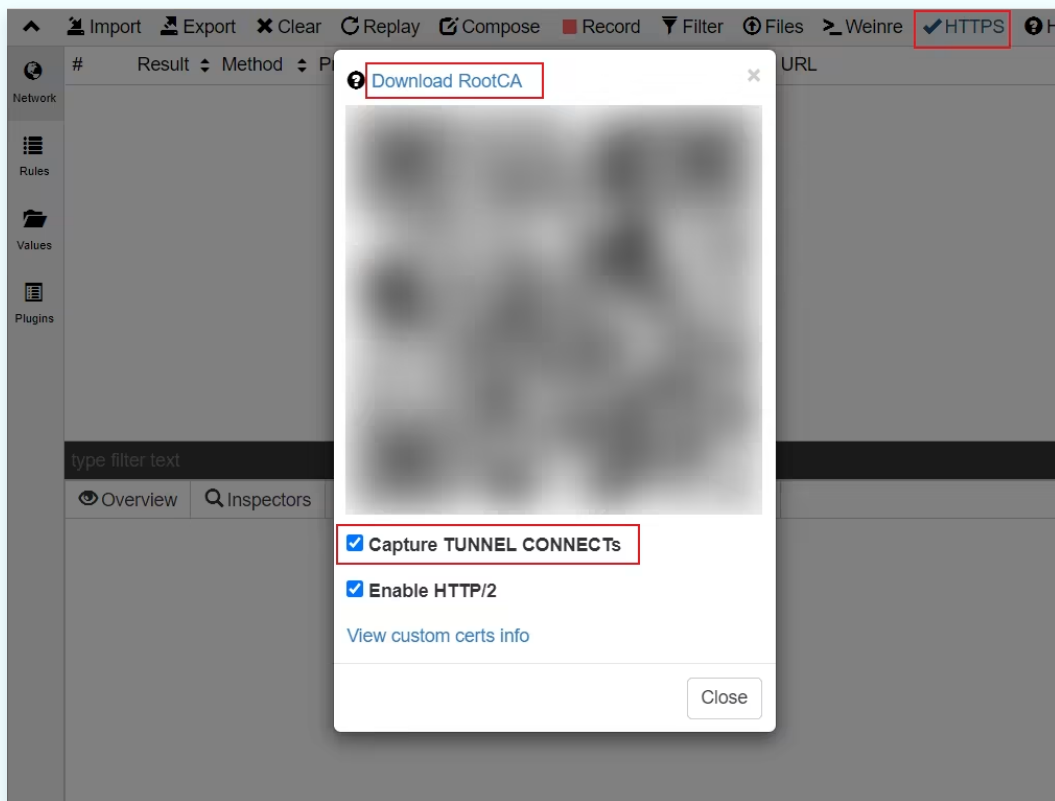
Click **Save** to save the changed content.



5. 1. Click HTTPS in the menu bar to enter the HTTPS configuration interface. Check "**Capture TUNNEL CONNECTs**".
6. 1. Click **Download RootCA** to download the root certificate locally.

Note:

No need to pay attention to the QR code information in this step. Click **Download RootCA** and check "**Capture TUNNEL CONNECTs**".



7. Install the root certificate. For specific steps, please see [Installing Root Certificate](#).

Step 3: Start Up Real-Time Compilation of the Demo Project

In the Demo project directory, execute the following command:

```
npm run dev
```

After successful compilation, the command line has the following output:

```
i [wds]: Project is running at https://localhost:9000/  
... (intermediate output omitted)  
i [wdm]: Compiled successfully.
```

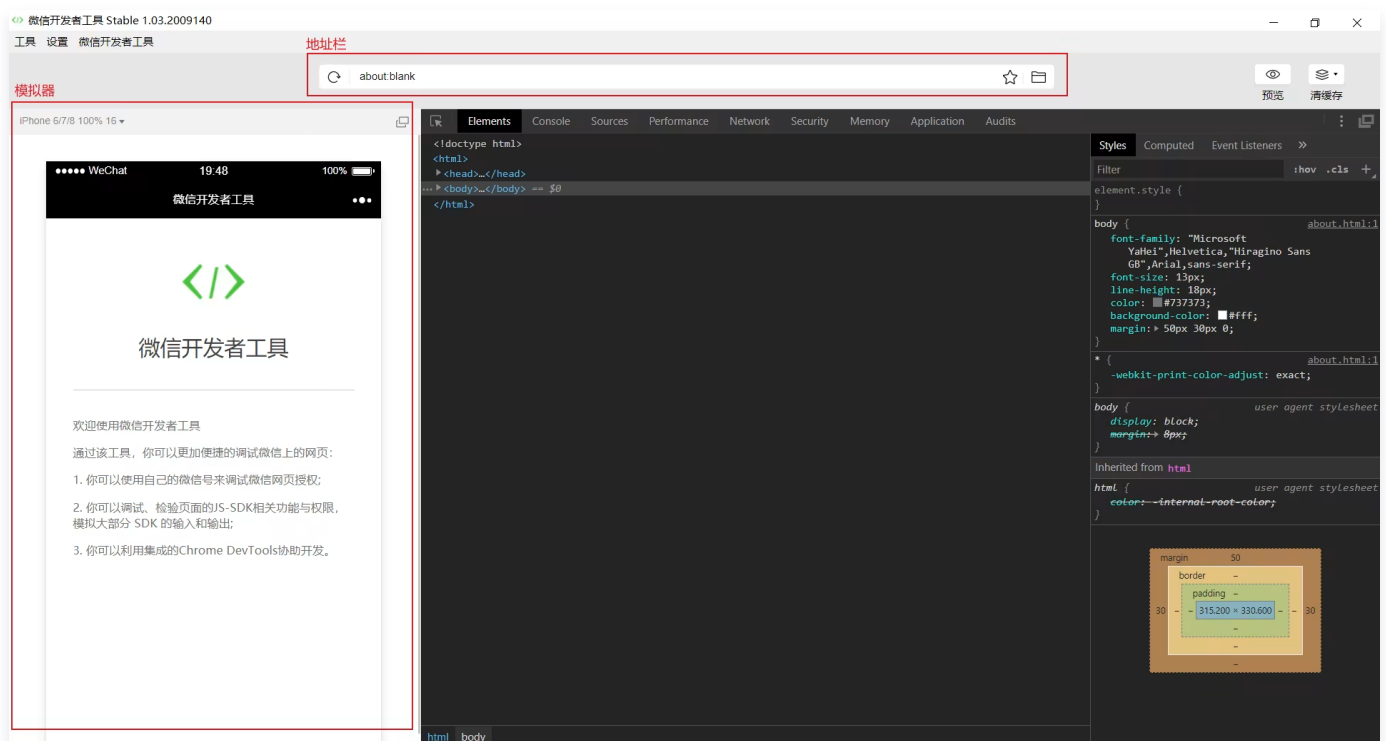
Step 4: Configure WeChat Developer Tool

1. Open WeChat Developer Tools and select **Settings > Proxy > Proxy Settings** from the menu bar.
2. Select **manually set proxy**, fill in the proxy address with `127.0.0.1`, and fill in the port with `8899`.



Step 5: Open the H5 Panel in WeChat Developer Tool

1. Open WeChat Developer Tools, click the official account webpage on the new project page, and enter the Official Account Debugging Page.



2. Verify whether the H5 panel JS files can be accessed.

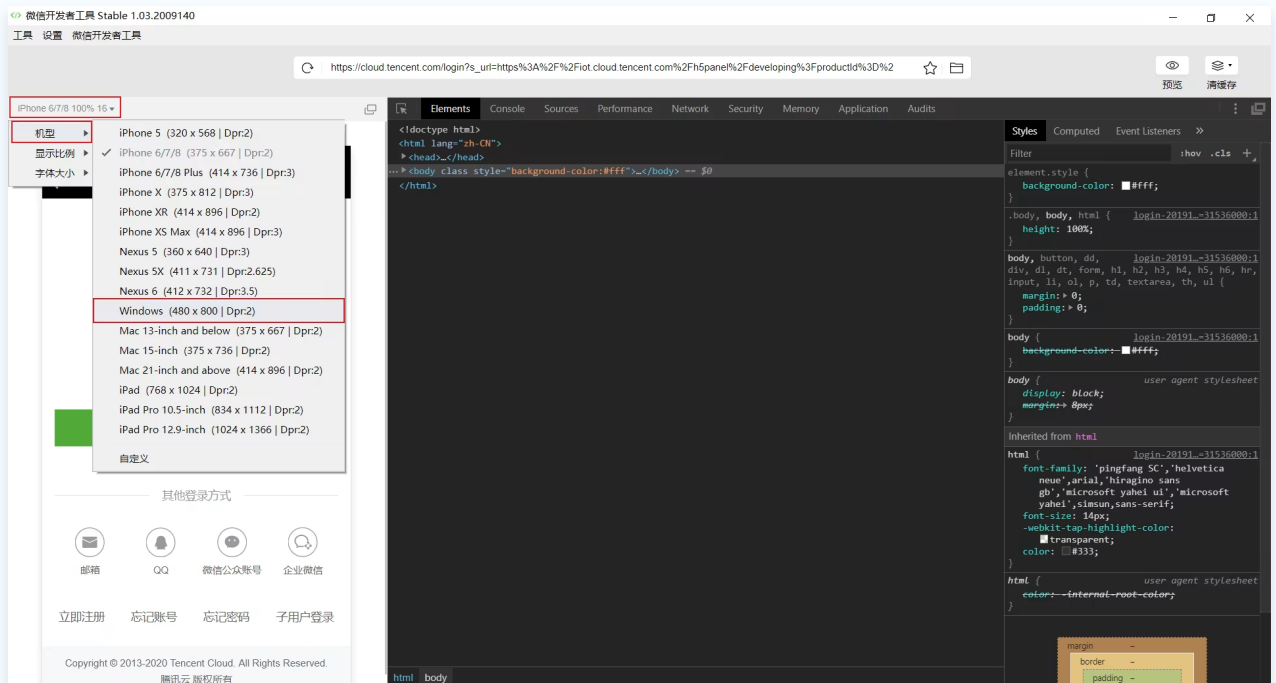
Enter `https://developing.script/developing.js` in the address bar and press Enter. If the JS file code can be loaded and displayed in the emulator, it indicates that the H5 panel JS files can be accessed.

3. Enter the [local debugging](#) URL in the address bar and press Enter. The emulator will jump to the login page of the Tencent Cloud console.

4. Select Model > "Windows (480×800 | Dpr:2)" from the emulator menu.

Note:

If the emulator simulates a mobile environment, WeChat or QQ login cannot be carried out normally. Here, change to the Windows model first. After logging in, it can be changed back to the original model setting.



5. Log in to the Tencent Cloud console on the emulator page. Once logged in successfully, the emulator will navigate to the H5 panel, where H5 panel debugging can be performed.

Agentless Mode

The way to enable the proxy-free mode is also simple. Just add a parameter after the link in the development panel: `&sw=true`. Suppose we want to develop a device with productId ABCDEFG and deviceName dev1. Then, the proxy-free mode can be enabled via the following link.

```
- https://iot.cloud.tencent.com/h5panel/developing?
productId=ABCDEFG&deviceName=dev1
+ https://iot.cloud.tencent.com/h5panel/developing?
productId=ABCDEFG&deviceName=dev1&sw=true
```

By adding `sw=true`, a service-worker will be enabled on the page. Through the interception and forwarding of requests by the service-worker, requests for `developing.script` and `deve`

`developing.style` will be proxied to the localhost development server. This approach is also effective for subsequent real device debugging and testing on actual devices.

Overview of Agentless Mode Principle

As previously mentioned, in development mode, the H5 panel framework will load JS files and CSS files from the following two URLs respectively:

- JS file URL: `https://developing.script/developing.js`
- – CSS file URL: `https://developing.style/developing.css`

The files packaged by the development server are accessed via localhost. Therefore, we need to perform a proxy between the two. The above-mentioned whistle plays such a role. If we don't use whistle, we can also use the browser's built-in service-worker. By listening to the `fetch` event and then intercepting and modifying requests, we can complete such work. This is the proxy-free mode. That is to say, the so-called proxy-free mode is not actually agentless. It merely avoids installing tools like whistle and achieves request proxying through the browser's own capabilities (service-worker).

Real Device Debugging

Through actual device debugging, you can perform H5 panel debugging on the specified device. Replace ABCDEFG with the actual product ID and abcdefg with the actual device name. If you need to debug the virtual device provided by IoT Explorer, replace abcdefg with `virtualDev`.

```
https://iot.cloud.tencent.com/h5panel/developing?
productId=ABCDEFG&deviceName=abcdefg
```

Note:

When debugging the specified device, this device will be automatically bound to the family of your Debug User, and the original binding relationship of this device will be removed at the same time.

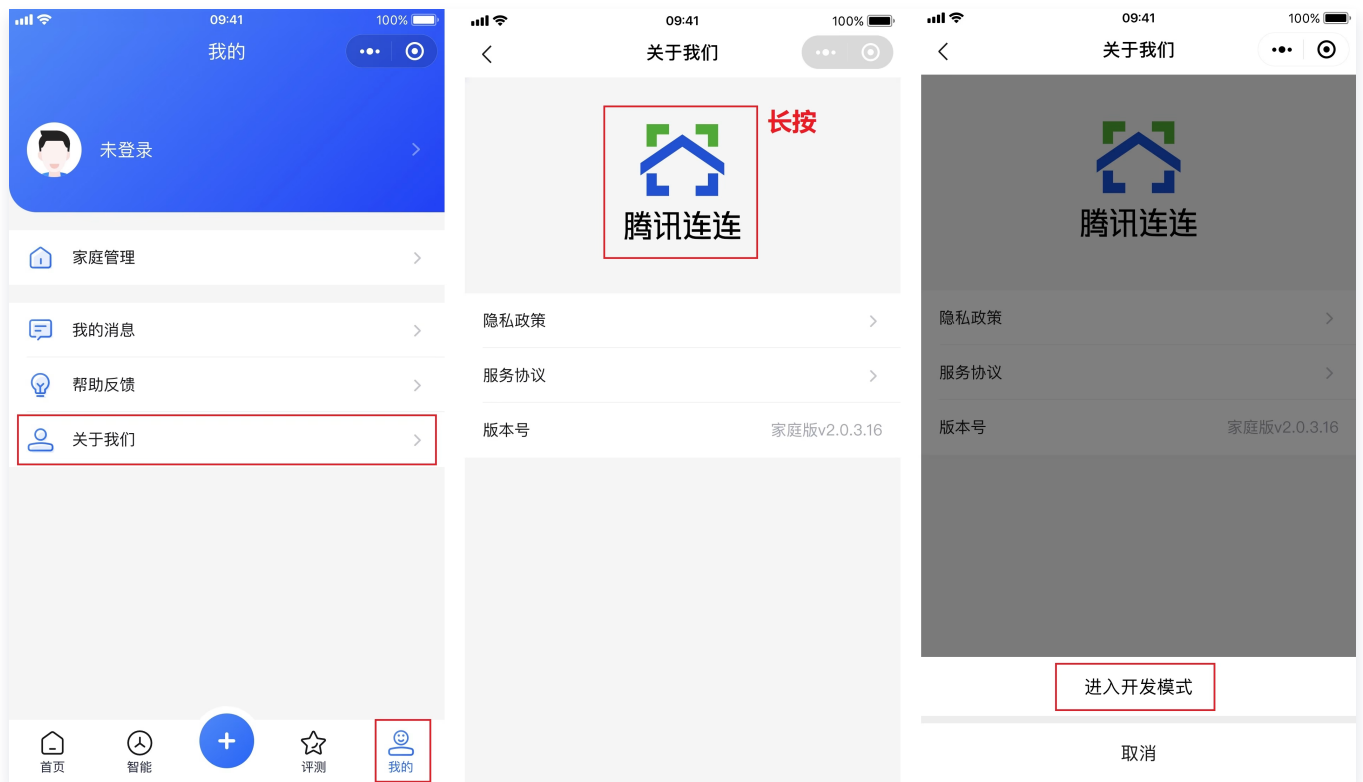
Released products do not support real device debugging, because there are risks of unbinding relationships of devices in use by real users.

Real Device Debugging

For released products, real device debugging is not open, and it is very difficult to debug the H5 panel in the mini program. To address this issue, the platform provides a real device debugging solution. Real device debugging allows debugging of the H5 panel while maintaining the device's existing binding relationship. Before starting, you need to use the

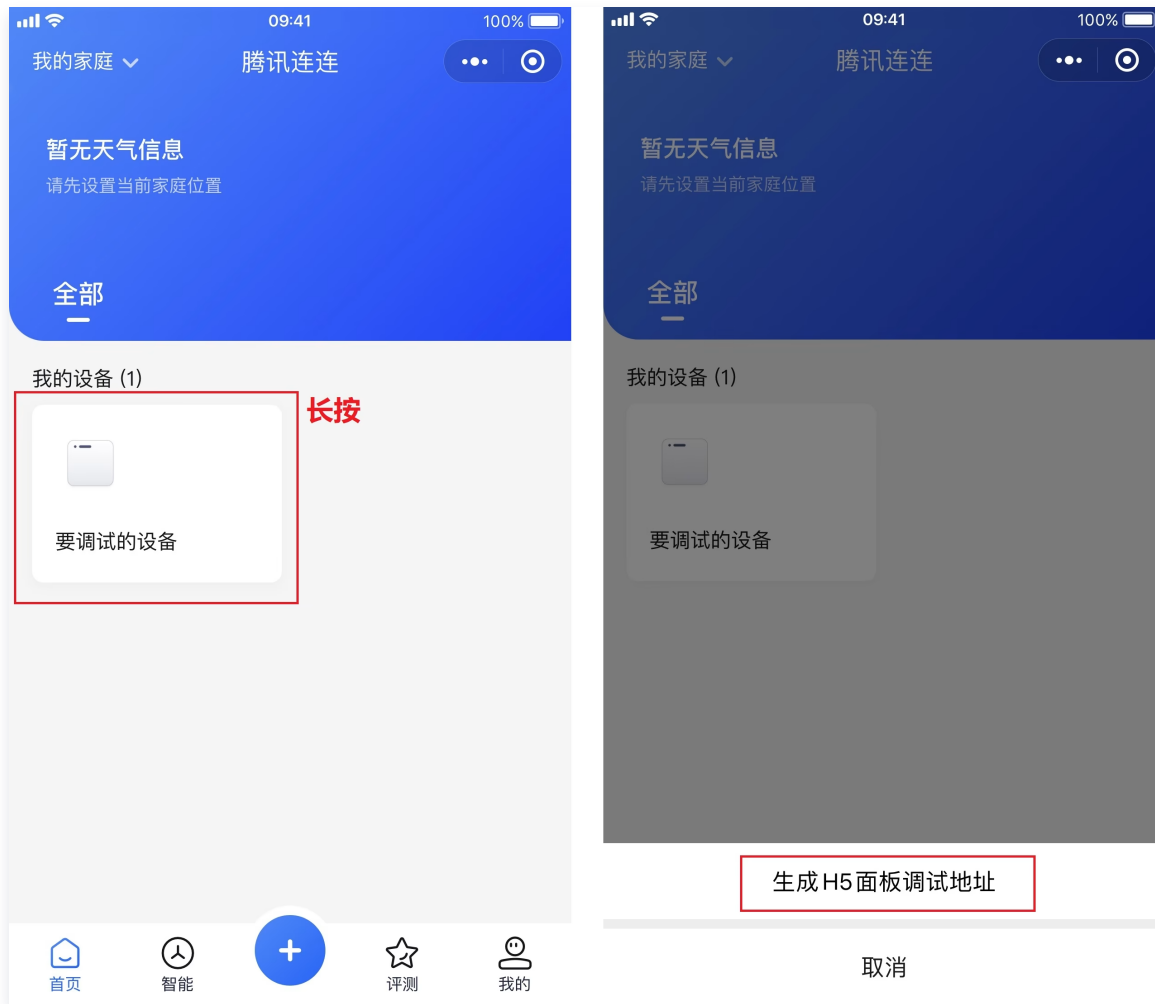
Tencent Lianlian Mini Program to bind the device to be debugged. Specific steps are as follows.

1. Open Tencent Lianlian Mini Program, select **Mine > About Us** to enter the About Us page.
2. Long-press the Tencent Lianlian Logo and select **Enter Development Mode** in the pop-up menu. Page prompt indicates successful entry into development mode.



3. Return to the homepage of Tencent Lianlian Mini Program. In the device list, long-press the device that needs to be debugged and select "**Generate H5 Panel Debugging Address**"

in the pop-up menu. Then, this debugging address will be copied to the clipboard.

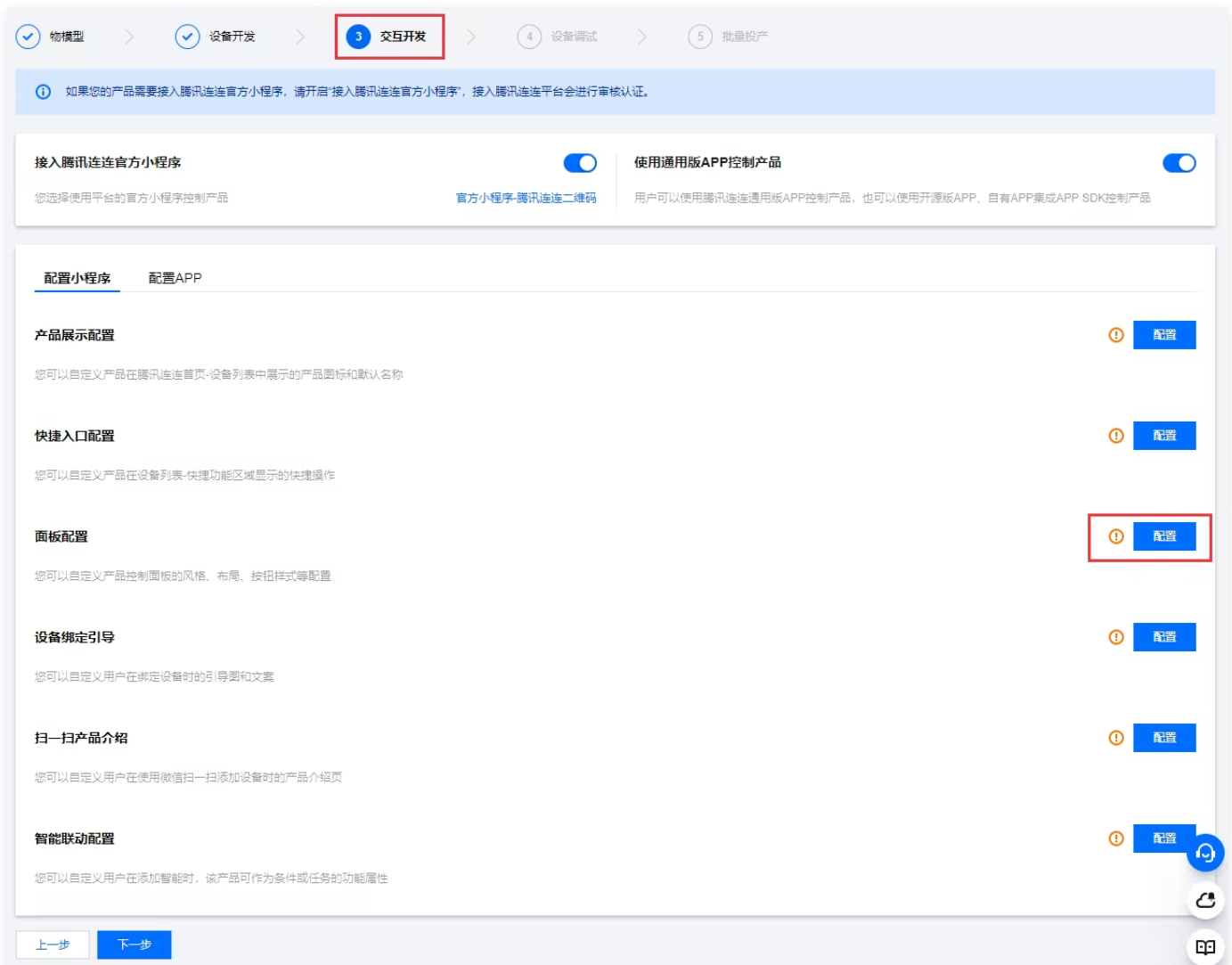


4. Use browser to access this debugging address to perform H5 panel real device debugging.

Uploading the H5 Panel

After the H5 panel development is completed, upload the JS file and CSS file of the H5 panel in the console. Specific operations are as follows:

1. Log in to Tencent Cloud and enter the [IoT Explorer Console](#), select a project and products, and enter the product detail page.
2. Select **Interactive Development > Panel Configuration**, click **Configure** to enter the panel configuration page.



3. Select H5 Custom Dashboard from the dashboard type.



4. In the **Upload JS** form, click **Upload** and select the JS files of the H5 panel. After upload completion, the page displays the file names, sizes, and MD5 info of the JS files.

上传文件

保留原文件名 ☐

上传JS (2MB以内) 操作

文件: d.js 文件大小: 20.83KB md5: 68e33 上传 删除

添加JS

上传CSS (2MB以内) 操作

文件: 上传 删除

添加CSS

5. If necessary, upload a CSS file. In the "**Upload CSS**" form, click **Upload** and select the CSS file of the H5 panel. After upload completion, the page displays the file name, size, and MD5 info of the CSS file.
6. Click **Save**. The page prompt indicates that the saving is successful. The uploaded JS files and CSS files are saved to take effect.

Note:

If the panel contains small images, they can be inlined into the code through build tools. If the image is relatively large, it is recommended to upload it to COS and then refer to it through a link.

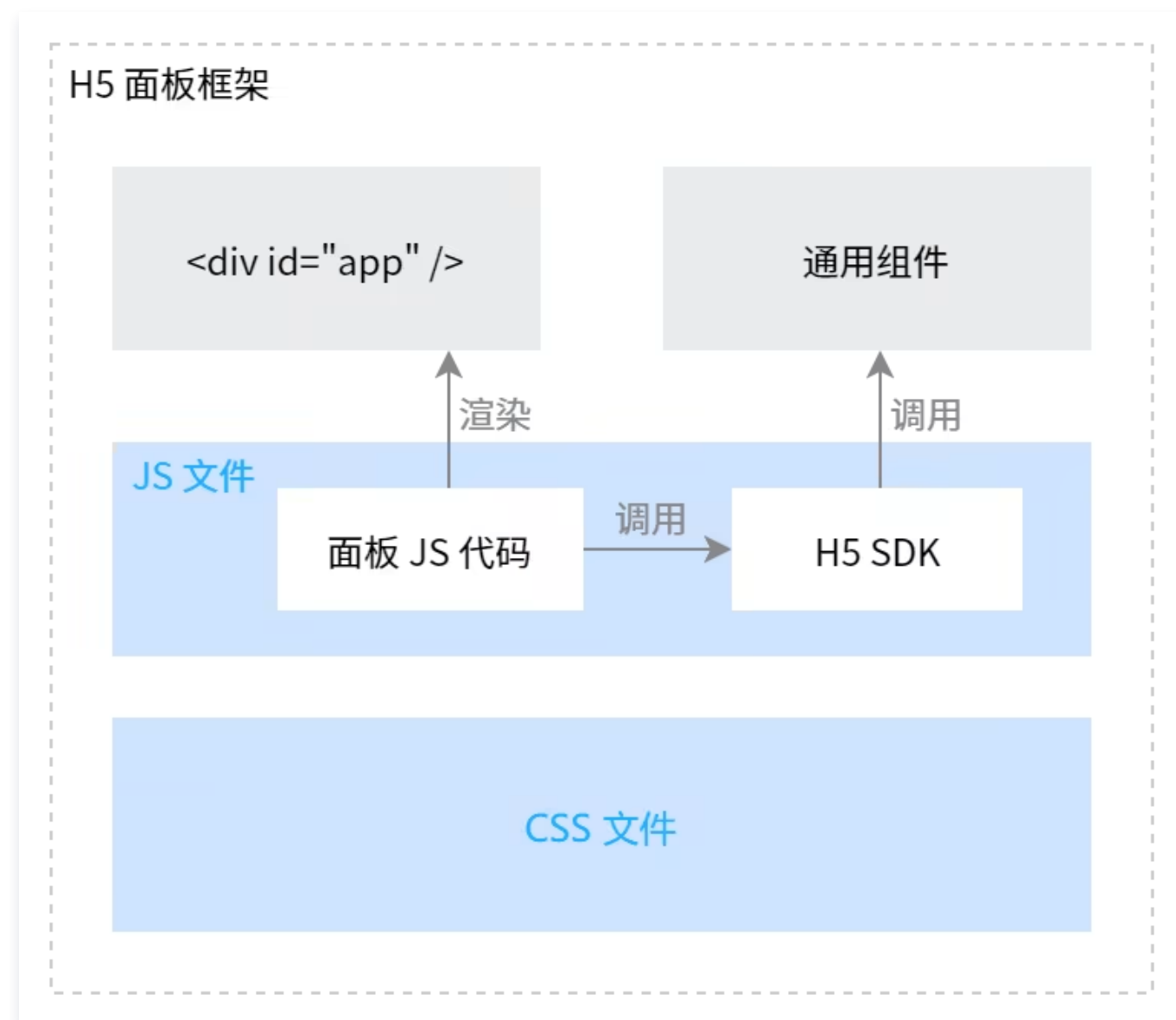
H5 Panel Access Allowlist

During the development phase, you need to set an access allowlist to use Tencent Lianlian to debug the H5 custom panel. For more information on how to set the access allowlist, see [Quick Start](#).

Principle of H5 Panel

Last updated: 2025-04-28 10:52:24

The H5 panel framework consists of the following:



- The `<div id="app"></div>` node, the H5 panel needs to be rendered under this DOM node.
- The general components of the H5 panel (such as the device details page, modal prompt dialog box, top notification, etc.) can be called through the H5 SDK.
- JS files and CSS files are uploaded by developers.

Developers need to implement the logic of rendering the H5 panel to the `<div id="app"></div>` node in the JS file. They can choose to use any technical solution for front-end rendering. Finally, they only need to output a JS file and a CSS file (the CSS file is optional), which can be loaded by the H5 panel.

Parameters required by the H5 panel, standard components of the H5 panel, and capabilities provided by the platform can be obtained through the H5 SDK.

Calling H5 SDK Capabilities

Last updated: 2025-04-28 10:52:44

Support for Importing the SDK for HTML5

Directly access through Windows.

```
window.h5PanelSdk;
```

Or configure externals in webpack.

```
// webpack.config.js
module.exports = {
  externals: {
    'qcloud-iotexplorer-h5-panel-sdk': 'h5PanelSdk',
  },
};
```

Retrieve Basic Parameters

H5 SDK provides basic parameters such as product information, device data, user information, and family information for use by the H5 panel and can be obtained through `sdk.attribute name`. For specific parameter list, see [Basic Parameters](#).

```
console.log(sdk.deviceId);
```

Call the Application-Side API

For information about the application-side API, see [Application-side API Introduction](#). The calling procedure of the application-side API by the H5 SDK has been encapsulated. When sending a request, common parameters `AccessToken` and `RequestId` will be automatically included. The following example code takes calling the application-side API of [Obtain Current Device Status](#) as an example.

```
const action = 'AppGetDeviceStatuses'; // Action name for requesting
application side API
const params = { // Parameters for requesting application side API
  DeviceIds: [sdk.deviceId]
```

```
};  
sdk.requestTokenApi(action, params).then(data => {  
  // Request successful  
  console.log(data);  
}).catch(err => {  
  //Request failed  
  console.error(err);  
});
```

Control Device

Calling the [Control Device Attributes](#) API of the H5 SDK can initiate control operations on the specified device.

```
// Attribute data of the device to be controlled  
const deviceData = {  
  light_switch: 0  
};  
  
// Control the current device of the H5 panel  
sdk.controlDeviceData(deviceData).then(data => {  
  // Request successful  
  console.log(data);  
}).catch(err => {  
  //Request failed  
  console.error(err);  
});  
  
// Control the specified device  
const deviceId = 'The ID of the device to control';  
sdk.controlDeviceData(deviceData, deviceId).then(data => {  
  // Request successful  
  console.log(data);  
}).catch(err => {  
  //Request failed  
  console.error(err);  
});
```

Firmware Upgrade

The SDK encapsulates the API for checking firmware updates, as well as a firmware upgrade prompt with a default style. Developers can check if there is an upgrade for the device's firmware currently and perform firmware upgrade operations by calling the interface provided by the H5 SDK.

Check Firmware Upgrade

Call the [Check if the Device Has an Upgradeable Firmware](#) API of the H5 SDK to check if the specified device has an upgradeable firmware. Developers can directly use the default-style firmware upgrade prompt provided by the H5 SDK, or they can use the returned value of this API to implement a custom firmware upgrade prompt. Details are as follows.

- **Method one: Use the firmware upgrade prompt with the default style provided by the H5 SDK**

Perform the following command to call `sdk.checkFirmwareUpgrade`.

```
sdk.checkFirmwareUpgrade();
```

If the current device has an upgradable firmware and the device is online, a default-style firmware upgrade prompt will pop up.



- **Method two: Use a custom firmware upgrade prompt**

When calling `sdk.checkFirmwareUpgrade`, if the `silent` parameter value is set to `true`, the above default style firmware upgrade prompt will not pop up. Developers can use the firmware upgrade information returned by the API to display a custom firmware upgrade prompt. Among them, `CurrentVersion` is the current firmware version of the device, and `DstVersion` is the upgradable firmware version.

```
sdk.checkFirmwareUpgrade({
  silent: true // If the silent parameter is true, the default
  firmware upgrade prompt of the H5 SDK will not pop up.
}).then(({ CurrentVersion, DstVersion }) => {
```

```
if (DstVersion && CurrentVersion !== DstVersion) {  
    // Developers need to implement the display of custom firmware  
    upgrade prompts here.  
}  
});
```

Performing Firmware Upgrade

Call the [Perform Firmware Upgrade](#) interface of the H5 SDK to perform firmware upgrade on the specified device. After calling this API, you will be redirected to the firmware upgrade page of the mini program to perform the firmware upgrade operation.

```
// Perform firmware upgrade on the current device  
sdk.goFirmwareUpgradePage();  
  
// Perform firmware upgrade on the specified device  
sdk.goFirmwareUpgradePage({  
    deviceId: 'The ID of the device to perform firmware upgrade'  
});
```

Long Connection Communication

The H5 SDK automatically connects to the WebSocket server by default during initialization and subscribes to the report data and status information of the current device.

```
// Listen for device-reported data push  
sdk.on('wsReport', ({ deviceId, deviceData }) => {  
    console.log('websocket device report', deviceId, deviceData);  
});  
  
// Listen for device online status change push  
sdk.on('wsStatusChange', ({ deviceId, deviceStatus }) => {  
    console.log('websocket device status change', deviceId, deviceStatus);  
});  
  
// Listen for device control push  
sdk.on('wsControl', ({ deviceId, deviceData }) => {  
    console.log('websocket device control', deviceId, deviceData);  
});
```

Device Details

The H5 SDK provides developers with two kinds of device details pages, which developers can choose to use as needed.

- **Standard device details:** Navigate to the device details page in the standard panel of the mini program.
- **H5 Custom Device Details:** Show the device details page within the H5 panel, and add custom menus and buttons on the basis of the standard device details.

Calling Standard Device Details

After executing the following code, you will be navigated to the device details page in the standard panel of the mini program.

```
sdk.goDeviceDetailPage();
```

Calling H5 Custom Device Details

After executing the following code, the device details page will be shown in the H5 panel. For the description of API parameters for H5 custom device details, see [Show H5 Custom Device Detail View](#).

```
sdk.showDeviceDetail({
  labelWidth: 120, // Label width of device detail
  margin-top: 8, // Padding-top of device detail
  shareParams: { // Parameters for custom share when sharing devices
    a: 'a',
    b: 'b',
  },
  extendItems: [
    {
      labelIcon:
'https://main.qcloudimg.com/raw/be1d876d55ec2479d384e17c94****e69.svg',
      label: 'custom menu'
      content: 'Custom menu content (optional)'
      onClick: () => console.log('Click custom menu')
    },
  ],
});
```



```
extendButtons: [
  {
    text: 'Custom button'
    type: 'warning',
    onClick: () => console.log('click Custom button')
  },
  {
    text: 'Custom button 2'
    type: 'primary',
    onClick: () => console.log('click Custom button 2')
  },
  {
    text: 'Get custom share parameters'
    onClick: async () => {
      const shareParams = await sdk.getShareParams();
      console.log('Custom share parameters: ', shareParams);
    }
  },
  {
    text: 'Hide device detail',
    type: '',
    onClick: () => sdk.hideDeviceDetail(),
  },
],
});
```

Sub-Device Using Gateway Panel

The platform supports setting the panel configuration of the gateway to take effect on all subproducts. Once enabled, the subproducts under the gateway will use the panel configuration of the gateway. The configuration steps are as follows:

1. Log in to Tencent Cloud and enter the [IoT Explorer console](#), select a project and a gateway product, and enter the product detail page of the gateway product.
2. Select **Interactive Development > Panel Configuration**, click **Configure**, and enter the panel configuration page.

物模型

设备开发

3 交互开发

4 设备调试

5 批量投产

如果您的产品需要接入腾讯连连官方小程序，请开启“接入腾讯连连官方小程序”，接入腾讯连连平台会进行审核认证。

接入腾讯连连官方小程序

使用通用版APP控制产品

您选择使用平台的官方小程序控制产品

官方小程序-腾讯连连二维码

用户可以使用腾讯连连通用版APP控制产品，也可以使用开源版APP、自有APP集成APP SDK控制产品

配置小程序

配置APP

产品展示配置

您可以自定义产品在腾讯连连首页-设备列表中展示的产品图标和默认名称

配置

快捷入口配置

您可以自定义产品在设备列表-快捷功能区域显示的快捷操作

配置

面板配置

您可以自定义产品控制面板的风格、布局、按钮样式等配置

配置

配网引导

您可以自定义产品的配网图文，使产品配网流程引导更明了、有效、符合产品的实际情况

配置

扫一扫产品介绍

您可以自定义用户在使用微信扫一扫添加设备时的产品介绍页

配置

智能联动配置

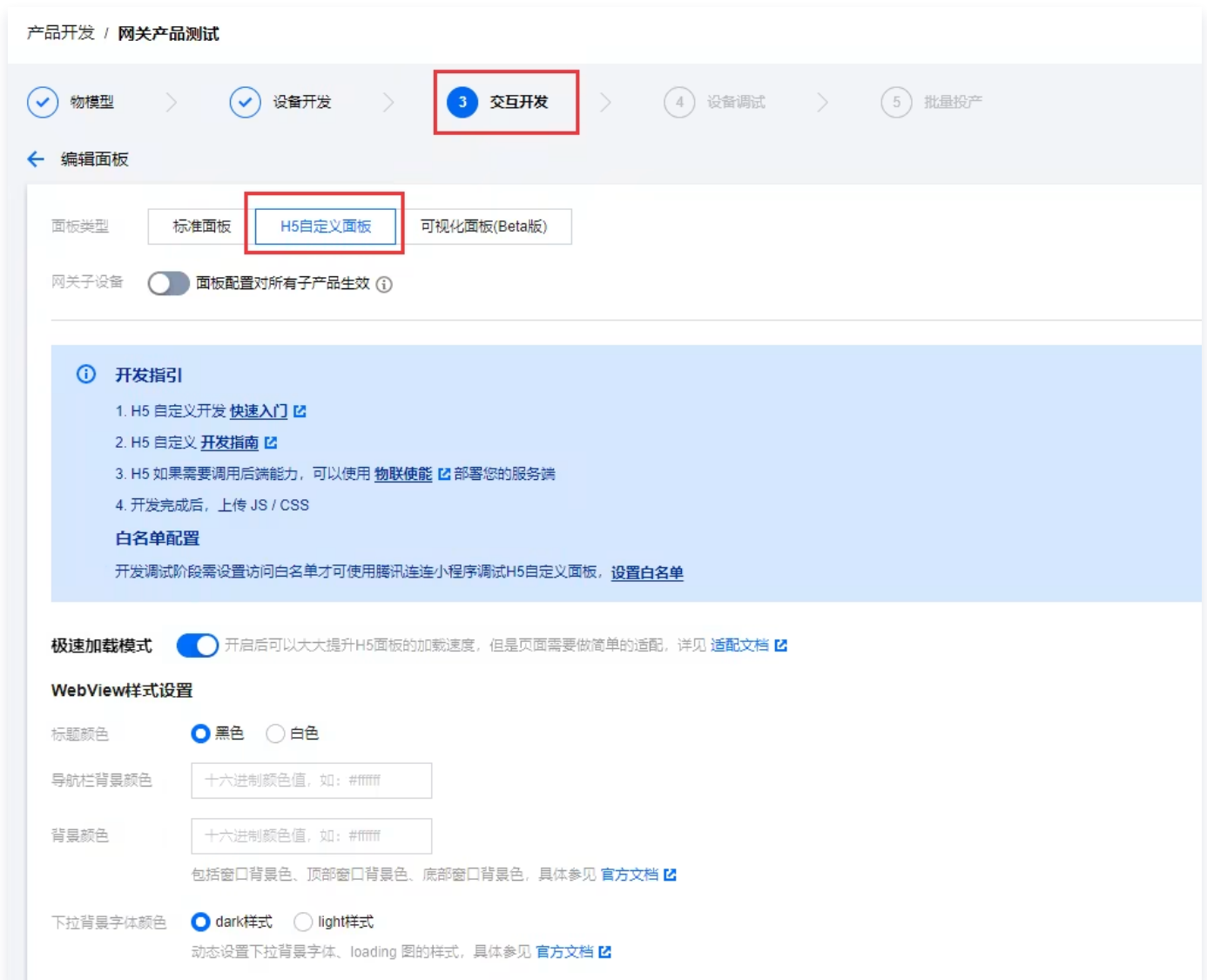
您可以自定义用户在添加智能时，该产品可作为条件或任务的功能属性

配置

上一步

下一步

3. Select "H5 Custom Dashboard" from the dashboard type.



4. Turn on the switch for **Dashboard Configuration to Take Effect on ALL Sub-products** in the gateway subdevice.



Listen for Page Display and Hide Events

The H5 SDK provides page display and hide event callbacks. When the H5 page or mini program switches to the foreground or background, corresponding events will be triggered. Developers can listen to these events and execute their own business logic when corresponding events are triggered.

```
// Listen for the H5 page to move to the foreground
sdk.on('pageShow', () => {
  console.log('on pageShow');
});

// Listen for the H5 page to switch to the background
sdk.on('pageHide', () => {
  console.log('on pageHide');
});

// Listen for the mini program to move to the foreground
sdk.on('appShow', () => {
  console.log('on appShow');
});

// Listen for the mini program to switch to the background
```

```
sdk.on('appHide', () => {
  console.log('on appHide');
});

// When the H5 page switches to the foreground, refresh the device name
// of the page title
sdk.on('pageShow', async () => {
  const deviceInfo = await sdk.getDeviceInfo();

  // Device display name
  const deviceDisplayName = deviceInfo.AliasName ||
sdk.productInfo.Name;
  // Update page title
  window.document.title = deviceDisplayName;
});
```

Bluetooth Device Communication

H5 panels cannot directly call the Bluetooth API of mini programs. The H5 SDK has encapsulated the Bluetooth protocol communication mechanism between the H5 panel and the mini program. To use Bluetooth communication capabilities in the H5 panel, you must use the Bluetooth module provided by the H5 SDK. For H5 panel development related to Bluetooth devices, please see [Bluetooth Device Documentation](#).

Custom Bluetooth Protocol H5 Panel Development

Last updated: 2025-04-28 10:53:03

Feature Overview

For products with a communication method of "BLE" and a device development method of "developing based on Custom Bluetooth Protocol", the device and device panel only support customized h5, which needs to be implemented by yourself.

Page Introduction

Two pages need to be implemented: device search page and device panel page. Currently, these two pages share the h5 development-free panel JS files. The manufacturer needs to manually handle the pages rendered in the JS according to the route. The **routes** of the two pages are as follows:

Debug Mode:

- Search page: `/h5panel/developing/live/bluetooth-search`
- Panel page: `/h5panel/developing/live`

Production Environment:

- Search page: `/h5panel/bluetooth-search`
- Panel page: `/h5panel`

For rendering pages, you can refer to the route implementation in `app.jsx` in [iotexplorer-h5-panel-demo](#).

Device Search Page

Main Features

Responsible for processing Bluetooth search, filtering out the manufacturer's own devices, and binding the devices to the user's family.

Since the protocols and broadcast contents of different Bluetooth devices vary, manufacturers need to implement their own [Device Adapter \(DeviceAdapter\)](#) and [Add Device Adapter](#). For the device search page, you can refer to the [BluetoothDemo/SearchPage](#) page in the demo.

Page Entrance

For details, see [Custom Bluetooth Protocol Device – Search Device H5 Page Entrance](#).

Development Process

1. Enter Tencent Lianlian development mode (refer to [H5 Panel Debugging – Real Device Debugging](#)).
2. Open the add device page, long-press the search Bluetooth device area, and select "Generate H5 Bluetooth Search Page Debugging Address" in the pop-up.
3. Send the link to the computer via Mobile WeChat or other means, and access the debugging link through the browser on the computer. The link format is as follows. Developers need to fill in the product ID to be developed in the link.

```
https://iot.cloud.tencent.com/h5panel/developing/live/bluetooth-search?h5PanelDevId=xxx&productId=
```

4. Like H5 panel debugging, the Search page will by default load the `//developing.script/developing.js` and `//developing.style/developing.css` files. You need to configure proxy forwarding to forward these two addresses to your locally developed JS and CSS.

Device Panel Page

Main Features

Like the H5 panel of other devices, developers can customize the device panel page and control the device through h5sdk.

Page Entrance

After successful binding to the family on the search page, you can click the corresponding device on the Tencent Lianlian homepage to enter the device panel page.

Development Process

1. First, you need to complete the development of the Search page. Only after the device is successfully bound to the family can you perform the development of the H5 panel page.
2. Just follow [H5 panel development](#).

Search for Device H5 Page Entrance

The H5 page entrance for Bluetooth search is divided into two situations: "product unpublished" and "product released". Details are as follows.

Product Unreleased

Method 1

1. Log in to [console](#), select the corresponding project, and click **Product Development** to enter the product list page.
2. Select a product with the status of "under development" to enter the product detail page, click **Interactive Development** > "Scan the product introduction" on the right of

Configuration to enter the configuration page.

3. Upload information related to "product entity diagram", "product remark", etc. Click **Save**. Then, you can scan the QR code provided in the bottom-right corner of the page for experience.

Method 2

1. As described in [real device debugging](#), enter Tencent Lianlian development mode.
2. Click "+" > **Add Device**, long-press the "Scan Nearby Bluetooth Devices" module at the top of the page, and input the product ID to go to the H5 Bluetooth Search Page in the pop-up dialog.

Product Released

Method 1

1. Log in to [console](#), select the corresponding project, and click **Product Development** to enter the product list page.
2. Select a product with the status "Released" to enter the product detail page, click **Batch Production**, and you can scan the QR code provided at the bottom of the page for experience.

**Note:**

If the page opened by scanning the code is not the custom Bluetooth search page, but the page for adding a device, please check whether the product configuration has been completed.

Method 2

You can [submit an application](#) and go through the authentication process of Tencent Lianlian. After becoming a recommended product; the product will be shown in **Add Device > Product**

Category List for you to choose from.

电工照明

安防报警

家用电器

网络设备

厨房电器

运动健康

影音办公

推荐


遥控大师空调伴
侣


华来小方智能彩
灯带STP01


窗帘


五路排插


灯


插座


开关面板


插排


五路灯


排插

Ultrafast Mode

Last updated: 2025-04-28 10:53:20

Feature Overview

When H5 is loading, it needs to pull all the data required by the panel from the server, which causes slow loading speed. To solve this problem, we introduced the ultrafast mode. The principle is that the server does not pull data during loading. Instead, after H5 is opened, the Mini Program Client pushes the local data to H5 through our unique link, which can greatly accelerate the loading time. However, since the original method of synchronously preparing all data at the time of direct output has been changed to asynchronously loading data after page rendering, the panels that depend on the directly output data during rendering need certain adaptations to function properly.

Adaptation Method

You only need to wait for a `sdkReady` Promise before rendering the page. For specific implementation, you can refer to the following examples of React implementation:

```
function Page() {
  const [sdkReady, setSdkReady] = useState(false);
  useEffect(() => {
    h5PanelSdk.sdkReady().then(() => setSdkReady(true));
  }, []);

  return !sdkReady ? <div> loading...</div> : (
    <div>
      {Render your page normally}
    </div>
  );
}
```

Enable Descriptions

Since the ultrafast mode takes effect at the product level, for existing products, please ensure a full release of the compatible version before enabling the ultrafast mode. Otherwise, it may affect the normal use of legacy users.

H5 Custom Development SDK

Basic Parameters

Last updated: 2025-04-28 10:53:48

Basic Parameters

H5 SDK provides basic parameters such as product information, device data, user information, and family information for use by the H5 panel, which can be obtained via `sdk.tributeName`.

Attribute Name	Type	Description
deviceId	string	Device ID consists of <code>{productId}/{deviceName}</code> .
productId	string	Product ID.
deviceName	string	Device name
deviceInfo	DeviceList ShareDevices	Equipment information has the same data structure as DeviceList ; if it is sharing devices, the data structure is the same as ShareDevices .
roomList	RoomList	The current room list of the family.
roomName	string	Room name of the current device.
dataTemplate	string	The Thing Model of the product where the device resides matches the Thing Model definition shown in Data Template > View JSON .
deviceStatus	number	Device online status. • Online: 1. • Offline: 0.
deviceDisplayName	string	Device display name, which will be displayed in sequence as: AliasName > productInfo; name > deviceName.
isShareDevice	boolean	Whether to share the device.

familyId	string	Family ID where the device resides. If it is a shared device, this value does not exist.
roomId	string	Room ID where the device is located. If it is a shared device, this value does not exist.
familyInfo	FamilyList	Family details where the device is located. If it is a shared device, this value does not exist.
isFamilyOwner	boolean	Whether the user is the owner of the current family.
userInfo	object	User Information. • Avatar:string; // Profile photo URL. • CountryCode: string; country code. • Email: string; email address. • NickName: string. • PhoneNumber: string; phone number. • UserID: string; // User ID.

Device management

Last updated: 2025-04-28 10:54:05

Obtain Product Information

- API definition

```
sdk.getProductInfo({ productId?: string }) => Promise<{  
  ProductId: string,  
  Name: string,  
  Description: string,  
  DataTemplate: string,  
  NetType: string,  
  CategoryId: number,  
  ProductType: number,  
  UpdateTime: number,  

```

- Parameter Description

Parameter Name	Parameter Description	Type	Required
productId	Optional. If not provided, use the product ID of the current device.	string	No

- Return value

Return a Promise. Output parameters are as follows:

Parameter Name	Parameter Description	Type
ProductId	Product ID.	string
Name	Product name.	string
Description	Product description.	string
DataTemplate	Product Data Template.	string

NetType	Modes of communication (subdevice uses Access Gateway Protocol).	string
CategoryId	Product Classification ID.	number
ProductType	Product type (0: general product; 5: gateway product).	number
UpdateTime	Last updated Unix timestamp (in seconds).	number

Get Device Information

- API definition

```
sdk.getDeviceInfo({ deviceId?: string }) => Promise<{
  ProductId: string,
  DeviceName: string,
  DeviceId: string,
  IconUrl: string,
  AliasName: string,
  UserId: string,
  RoomId: string,
  CreateTime: number,
  UpdateTime: number
} | {
  ProductId: string,
  DeviceName: string,
  DeviceId: string,
  IconUrl: string,
  AliasName: string,
  CreateTime: string
}>
```

- Parameter Description

Parameter Name	Parameter Description	Type	Required
deviceId	Optional. If not provided, use the device ID of the current device.	string	No

- Return Value

Return a Promise. For output parameters, please see [DeviceList](#) (non-shared device) and [ShareDevices](#) (shared device).

Get Sub-Device List Bound to Gateway

This API can only be used for gateway devices.

- API definition

```
sdk.getSubDeviceList() => Promise<{
  subDeviceList: Device[],
  syncFailList: Device[],
}>
```

- Return Value

Return a Promise. `subDeviceList` is the sub-device list bound to a family under this gateway. `syncFailList` is the sub-device list not bound to a family under this gateway. For the data structure of Device, see [deviceList](#).

Control Device Attributes

- API definition

```
sdk.controlDeviceData(data, deviceId?: string) => Promise
```

- Parameter Description

Parameter Name	Parameter Description	Type	Required
data	Device property data to be controlled.	object	Yes
deviceId	Optional. If not provided, use the device ID of the current device.	string	No

- Return Value

Return a Promise. For output parameters, please see [user control device](#).

Calling Device Behavior

- API definition


```
sdk.callDeviceAction(actionPayload: ActionPayload, actionId: string,  
deviceId?: string) => Promise
```

- **Parameter Description**

Parameter Name	Parameter Description	Type	Required
actionPayload	Behavior invocation input parameters defined in the Thing Model.	object	Yes
actionId	The identifier of this behavior defined in the Thing Model.	string	Yes
deviceId	Optional. If not provided, use the device ID of the current device.	string	No

- **Return Value**

Return a Promise. For output parameters, please see [synchronous invocation of device behavior](#).

Retrieve Device Object Model Data

- **API definition**

```
sdk.getDeviceData({ deviceId?: string }) => Promise<object>
```

- **Parameter Description**

Parameter Name	Parameter Description	Type	Required
deviceId	Optional. If not provided, use the device ID of the current device.	string	No

- **Return Value**

Return a Promise. The output parameter is the device's model data.

Retrieve Device Object Model Historical Data

- **API definition**

```
sdk.getDeviceDataHistory({
```

```
    FileName: string,
    MaxTime: number,
    MinTime: number,
    Context?: string,
    Limit: number
  }) => Promise<{
    RequestId: string,
    Context: string,
    FileName: string,
    Listover: boolean,
    Results: DataHistoryItem[]
  }>
```

• Parameter Description

Parameter Name	Parameter Description	Type	Required
FieldName	Inquired attribute name.	string	Yes
MaxTime	End time, timestamp in milliseconds.	number	Yes
MinTime	Start time, timestamp in milliseconds.	number	Yes
Context	Paging cursor, optional for the initial query.	string	No
Limit	Single page data volume.	number	Yes

• Return Value

Return a Promise. For the output parameter, please see [retrieving historical data of the device object model](#).

Get Current Device Status

• API definition

```
sdk.getDeviceStatus({ deviceId?: string }) => Promise<0 | 1>
```

• Parameter Description

Parameter Name	Parameter Description	Type	Required
----------------	-----------------------	------	----------

deviceId	Optional. If not provided, use the device ID of the current device.	string	No
----------	---	--------	----

- **Return Value**

Return a Promise. The output result values are as follows:

- 0: The device is offline.
- 1: Device online.

Deleting a Device

- **API definition**

```
sdk.deleteDevice({ deviceId?: string }) => Promise<void>
```

- **Parameter Description**

Parameter Name	Parameter Description	Type	Required
deviceId	Optional. If not provided, use the device ID of the current device.	string	No

- **Return Value**

Return a Promise.

Get Custom Share Parameters

If the device is a shared device and the sharer has set custom share parameters, the sharee can get custom share parameters through this API after accepting the sharing.

- **API definition**

```
sdk.getShareParams({ deviceId?: string }) => Promise<any>
```

- **Parameter Description**

Parameter Name	Parameter Description	Type	Required
deviceId	Optional. If not provided, use the device ID of the current device.	string	No

- **{return value}**

Return a Promise. The output parameter is the custom share parameter.

Check If the Device Has Upgradable Firmware

Check if there is an upgrade for the firmware of the specified device. If there is an upgrade and the device is online, pop up a firmware upgrade prompt.

- **API definition**

```
sdk.checkFirmwareUpgrade({
  deviceId?: string,
  silent?: boolean
}) => Promise<{
  CurrentVersion: string,
  DstVersion: string,
}>
```

- **Parameter Description**

Parameter Name	Parameter Description	Type	Required
deviceId	Optional. If not provided, use the device ID of the current device.	string	No
silent	Selectable, is false by default. • true: Only check for firmware upgrades without popping up a firmware upgrade prompt. • false: Check for firmware upgrades. If an upgradable firmware exists and the device is online, pop up a firmware upgrade prompt.	boolean	No

- **Return Value**

Return a Promise. For output parameters, please see [Query Whether Device Firmware Is Upgraded](#).

Retrieve the Latest Firmware Information

Get the firmware information pushed by the console.

- **API definition**

```
sdk.getDeviceLatestOTAInfo({ deviceId?: string }) => Promise
```

- **Parameter Description**

Parameter Name	Parameter Description	Type	Required
deviceId	Optional. If not provided, use the device ID of the current device.	string	No

- **Return Value**

Return a Promise. The output parameters are as follows:

Parameter Name	Parameter Description	Type
TargetVersion	Target firmware version.	string
FirmwareURL	Firmware Download Address.	string

Download Firmware

Download the firmware pushed by the console.

- **API definition**

```
sdk.downloadDeviceOTA({ deviceId?: string }) => Promise
```

- **Parameter Description**

Parameter Name	Parameter Description	Type	Required
deviceId	Optional. If not provided, use the device ID of the current device.	string	No

- **Return Value**

Return a Promise. The output parameters are as follows:

Parameter Name	Parameter Description	Type
----------------	-----------------------	------

otaData	Firmware in ArrayBuffer format.	ArrayBuffer
deviceId	Device ID.	string

Firmware Version Reporting

Submit the firmware version of the device side.

- **API definition**

```
sdk.reportFirmwareVersion({ deviceId?: string, version: string }) =>  
Promise
```

- **{parameter description}**

Parameter Name	Parameter Description	Type	Required
deviceId	Optional. If not provided, use the device ID of the current device.	string	No
version	Firmware version of the current device.	string	Yes

Firmware Upgrade Status Report

Report the firmware upgrade status of the device side.

- **API definition**

```
sdk.reportOTAStatus({ deviceId?: string, state: string, percent:  
string, version: string, resultCode: number, resultMsg?: string }) =>  
Promise
```

- **{parameter description}**

Parameter Name	Parameter Description	Type	Required
deviceId	Optional. If not provided, use the device ID of the current device.	string	No
state	• downloading: Indicates that the firmware is being downloaded. • burning: Indicates that it is being burned. • done: indicates completion.	string	Yes

	• failure		
percent	Firmware upgrade progress 0 – 100.	number	Yes
version	Firmware version of the current device.	string	Yes
resultCode	Error code. • -1: Download Timeout. • -2: File does not exist. • -3: Signature expired. • -4: MD5 mismatch. • -5: Firmware update failed.	number	Yes
resultMsg	Error message, required if unsuccessful, error cause.	string	No

Firmware Upgrade Information Query

Query the firmware upgrade information of the device

- **API definition**

```
sdk.checkFirmwareUpdate({ ProductId: string, DeviceName: string }) =>
Promise
```

- **{parameter description}**

Parameter Name	Parameter Description	Type	Required
ProductId	Product id	string	Yes
DeviceName	Device name	string	Yes

- **Return value**

Return a Promise. Output parameters are as follows:

Parameter Name	Parameter Description	Type
CurrentVersion	Current version of the device	string
DstVersion	Released version	string

Performing Firmware Upgrade

Navigate to the firmware upgrade page of the mini program and perform firmware upgrade.

- API definition

```
sdk.goFirmwareUpgradePage({ deviceId?: string }) => Promise
```

- {parameter description}

Parameter Name	Parameter Description	Type	Required
deviceId	Optional. If not provided, use the device ID of the current device.	string	No

- Return Value

Return a Promise.

User Management

Last updated: 2025-04-28 10:54:23

Retrieve User Information

- API definition

```
sdk.getUserInfo() => Promise<{
  Avatar: string,
  CountryCode: string,
  Email: string,
  NickName: string,
  PhoneNumber: string,
  UserID: string
}>
```

- Return Value

Return a Promise. The output parameters are as follows:

Parameter Name	Parameter Description	Type
Avatar	Profile photo URL.	string
CountryCode	Country code.	string
Email	Email address.	string
NickName	Nickname.	string
PhoneNumber	Phone number.	string
UserID	User ID.	string

Interface Components

Last updated: 2025-04-28 10:54:42

tips Component

tips component, style and style match Tencent Lianlian Mini Program.

Basic Usage

- API definition

```
sdk.tips.show(message, options) => Promise
```

- {parameter description}

Parameter Name	Type	Required	Parameter Description
message	string	Yes	Prompt text.
options.type	'info' 'danger' 'loading' 'success'	No	tips type.
options.waitForHide	boolean	No	If it is true, the show method returns a Promise, and the resolve will only be triggered after closing.
options.duration	number	No	The time when the notification shows, in milliseconds, default 1500.
options.delayDuration	number	No	The default value is 0, in milliseconds. The prompt will show after this delay.
options.canClickClose	boolean	No	Set to true by default. Whether it is possible to close prompt by clicking mask.

options.canBeReplace	boolean	No	Default to false. When it is false, if the previous notification is not closed, subsequent calls to tips.show will be ignored.
----------------------	---------	----	--

Close Prompt

Interface Definition

```
sdk.tips.hide() => Promise
```

Display a Loading Notification

The encapsulated `tips.show` method is equivalent to:

```
sdk.tips.show(message, {
  type: 'loading',
  canBeReplace: true,
  duration: 0,
  delayDuration: 200,
  canClickClose: false,
  ...options,
});
```

- API definition

```
sdk.tips.showLoading(message, options) => Promise
```

- {parameter description}

Please see [parameter description of displaying tips](#).

Close Loading Notification

[Show loading tips](#). You must proactively call this API, otherwise the tips will be kept.

Interface Definition

```
sdk.tips.hideLoading() => Promise
```

Display a Success Notification

The encapsulated `tips.show` method is equivalent to:

```
sdk.tips.show(message, { type: 'success', ...options });
```

- **API definition**

```
sdk.tips.showSuccess(message, options) => Promise
```

- **Parameter Description**

Please see [parameter description of displaying tips](#).

Display General Notifications

The encapsulated `tips.show` method is equivalent to:

```
sdk.tips.show(message, { type: 'info', ...options });
```

- **API definition**

```
sdk.tips.showInfo(message, options) => Promise
```

- **Parameter Description**

Please see [parameter description of displaying tips](#).

Display Error Notifications

First perform standardized processing on error display information, then show error tips.
Equivalent to:

```
sdk.tips.show(getErrorMsg(error), { type: 'error', ...options });
```

- **API definition**

```
sdk.tips.showError(error, options) => Promise
```

- **{parameter description}**

Parameter Name	Type	Required	Parameter Description
----------------	------	----------	-----------------------

error	- Error { code, msg } - string	Yes	Error object or error information.
options	object	No	Please see parameter description of displaying tips .

Modal Dialog Box

Basic Usage

Show a popup. Its parameters, functionality and style are basically the same as the native showModal of the mini program.

- API definition

```
sdk.tips.showModal({
  title?: string,
  content?: string,
  showCancel?: boolean,
  cancelText?: string,
  cancelColor?: string,
  confirmText?: string,
  confirmColor?: string,
}) => Promise<boolean>
```

- Parameter Description

Parameter Name	Type	Required	Parameter Description
title	string	No	Pop-up title
content	string	No	Pop-up content.
showCancel	boolean	No	Whether to display the cancel button. Set to true by default.
cancelText	string	No	Cancel button text, defaults to "Cancel".
cancelColor	string	No	Cancel button color, default is "#6C7078".

confirmText	string	No	Confirmation button text, defaults to "Confirm".
confirmColor	string	No	Confirmation button color, default is "#0066FF".

- **Return Value**

Return a `Promise<boolean>`. Output values are as follows:

- true: User clicks Confirm
- false: User clicks Cancel

Confirm Modal Dialog Box

Encapsulation based on `sdk.tips.showModal`, for use in confirming operations with users twice.

- **API definition**

```
sdk.tips.confirm(title, content, options) => Promise<boolean>
```

- **{parameter description}**

Parameter Name	Type	Required	Parameter Description
title	string	No	Pop-up title
content	string	No	Pop-up content.
options	object	No	Please see Show Modal Dialog Box .

- **Sample code**

```
const isConfirm = await sdk.tips.confirm('Confirm deletion of the device?');
if (isConfirm) {
  // User confirmation, execute subsequent process
}
```

Prompt Modal Dialog Box

Encapsulation based on `sdk.tips.showModal`, used for producing message prompts when operating for users.

- **API definition**

```
sdk.tips.alert(content, options) => Promise<boolean>
```

- **{parameter description}**

Parameter Name	Type	Required	Parameter Description
content	string	No	Pop-up content.
options	object	No	Please see Show Modal Dialog Box .

- **Example code**

Produce a message prompt for the user.

```
await sdk.tips.alert('The feature is temporarily unavailable. Try again later.');
```

Device Offline Prompt Component

Prompt component for offline devices, with the same style as the Tencent Lianlian Mini Program.

Basic Usage

- **API definition**

```
// Calling method 1
sdk.offlineTip.show() => void
// Calling method 2
sdk.showOfflineTip() => void
```

Close Prompt

- **API definition**

```
// Calling method 1
sdk.offlineTip.hide() => void
// Calling method 2
```

```
sdk.hideOfflineTip() => void
```

H5 Custom Device Detail View

Display Device Detail View

Currently show a full-screen device detail view in H5, support adding custom menu items and buttons.

- **API definition**

```
sdk.showDeviceDetail({
  deviceInfo?: object,
  labelWidth?: number,
  marginTop?: number,
  shareParams?: object,
  extendItems?: ExtendItemConfig[],
  extendButtons?: ExtendButtonConfig[],
  containerClassName?: string
}) => void
```

- **Parameter Description**

Parameter Name	Type	Required	Parameter Description
extendItems[].label	string	Yes	Title of custom menu item.
extendButtons[].text	string	Yes	Custom button text.
extendButtons[].onClick	function	Yes	Callback triggered after clicking the custom button.
deviceInfo	object	No	Display details of equipment information. If not provided, use the current device information.
labelWidth	number	No	The label width of device detail, default 110, unit px.
marginTop	number	No	Padding-top of device detail: The default value is 10, and the

			unit is px.
shareParams	- object - string	No	Custom share parameters.
extendItems	ExtendItemConfig[]	No	Custom menu configuration.
extendItems[].labelIcon	string	No	The icon address displayed in front of the label.
extendItems[].content	string	No	The content of a custom menu item.
extendItems[].className	string	No	Style class name of custom menu item.
extendItems[].onClick	function	No	Callback triggered after clicking a custom menu item.
extendButtons	ExtendButtonConfig[]	No	Custom button configuration.
extendButtons[].className	string	No	Style class name of the custom button.
extendButtons[].type	'danger' 'primary' 'warning'	No	Style of the custom button.
containerClassName	string	No	Style class name of the container.

Close Device Detail View

Interface Definition

```
sdk.hideDeviceDetail() => void
```

Calling Mini Program Capacity

Last updated: 2025-04-28 11:02:13

Standard Device Details Page for Mini Program Redirection

- API definition

```
sdk.goDeviceDetailPage({
  reload?: boolean,
  deviceId?: string,
  isShareDevice?: boolean,
  shareParams?: object | string,
}) => Promise
```

- Parameter Description

Parameter Name	Parameter Description	Type	Required
reload	- If true, the data of the device will be re-pulled once after entering the details page. - If false, the cached device data will be used after entering the details page.	boolean	No
deviceId	Optional. If not provided, use the device ID of the current device.	string	No
isShareDevice	Optional. Whether the device is a shared device. If not provided, use the current <code>sdk.isShareDevice</code> .	boolean	No
shareParams	Optional. Custom share parameters for the device.	- object - string	No

- Return Value

Return a Promise.

Gateway for Mini Program Redirection – Add Sub-Device Page

- API definition

```
sdk.goGatewayAddSubDevicePage (gatewayDeviceId: string);
```

- **parameter description**

Parameter Name	Parameter Description	Type	Required
gatewayDeviceId	Gateway device ID.	string	Yes

- **Return Value**

Return a Promise.

Feedback Page for Mini Program Redirection

- **API definition**

```
sdk.goFeedBackPage() => Promise
```

- **return value**

Return a Promise.

Device Information Page for Mini Program Redirection

- **API definition**

```
sdk.goDeviceInfoPage({ deviceId?: string }) => Promise
```

- **Parameter Description**

Parameter Name	Parameter Description	Type	Required
deviceId	Optional. If not provided, use the device ID of the current device.	string	No

- **return value**

Return a Promise.

Mini Program Redirection – Modify Device Name Page

- **API definition**

```
sdk.goEditDeviceNamePage({ deviceId?: string, name?: string }) =>
Promise
```

- **Parameter Description**

Parameter Name	Parameter Description	Type	Required
deviceId	Optional. If not provided, use the device ID of the current device.	string	No
name	Optional. If not provided, use the aliasName of the current device.	string	No

- **Return Value**

Return a Promise.

Dashboard Page for Other Devices Redirection

- **API definition**

```
sdk.goDevicePanelPage(deviceId: string, customParams?: {
passThroughParams: Record<string, string> }) => Promise
```

- **parameter description**

Parameter Name	Parameter Description	Type	Required
deviceId	The device ID to redirect to the panel.	string	Yes
passThroughParams	It will be parsed into a query string and appended to the address of the panel to be redirected to.	Record <string, string>	No

- **return value**

Return a Promise.

Mini Program Redirection – Room Settings Page

- **API definition**

```
sdk.goRoomSettingPage({ deviceId?: string }) => Promise
```

- **Parameter Description**

Parameter Name	Parameter Description	Type	Required
deviceId	Optional. If not provided, use the device ID of the current device.	string	No

- **return value**

Return a Promise.

Jump to the Device Sharing Page of the Mini Program

- **API definition**

```
sdk.goShareDevicePage({ deviceId?: string }) => Promise
```

- **Parameter Description**

Parameter Name	Parameter Description	Type	Required
deviceId	Optional. If not provided, use the device ID of the current device.	string	No

- **return value**

Return a Promise.

Refresh Data in the Mini Program

Require the mini program to perform a data refresh once after the current H5 panel is closed.

- **API definition**

```
sdk.reloadAfterUnmount() => Promise
```

- **return value**

Return a Promise.

Navigate to Another Mini Program

Navigate to another mini program. Applicable to scenarios such as in-panel purchase of added services. For input parameters and descriptions, see [wx.navigateToMiniprogram](#).

Note:

Only after adding the appid of the target mini program to the allowlist can you call this API. If necessary, please contact customer service to enable it.

- **API definition**

```
sdk.navigateToMiniprogram({ appid, extraData, envVersion }) => Promise
```

- **return value**

Return a Promise.

Return to the Previous Level Page of the Mini Program

Can be used to proactively close the H5 panel.

- **API definition**

```
sdk.navBack() => Promise
```

- **return value**

Return a Promise.

Set the Sharing Content of the Current Page

Set the share content of the current page and push sharing information to the mini program through [wx.miniProgram.postMessage](#). For details, refer to [Mini Program Page Sharing Documentation](#).

- **API definition**

```
sdk.setShareConfig({ title: string, imgUrl: string? }) => Promise
```

- **Parameter Description**

Parameter Name	Parameter Description	Type	Required
title	Title for sharing.	string	Yes

imgUrl	The address for sharing an image will, by default, be a screenshot of the current page.	string	No
--------	---	--------	----

- **return value**

Return a Promise.

Redirect to Cloud Scheduled

- **API definition**

```
sdk.goTimingProjectPage({ deviceId, isShareDevice, featureId } = {})
```

- **parameter description**

Parameter Name	Parameter Description	Type	Required
deviceId	Optional. If not provided, use the device ID of the current device.	string	No
isShareDevice	Optional. Whether the device is a shared device. If not provided, use the current <code>sdk.isShareDevice</code> .	boolean	No
featureId	Thing Model identifier. After the input parameter is passed in, only the scheduled tasks corresponding to the identifier will be filtered out.	string	No

- **return value**

No return value.

Notify the Mini Program to Trigger Vibration

- **API definition**

```
sdk.triggerVibrateShort(type)
```

- **Parameter Description**

Parameter Name	Parameter Description	Type	Required
----------------	-----------------------	------	----------

type	Vibration intensity type, valid values: heavy, medium, light.	string	Yes
------	---	--------	-----

• **Return Value**

No return value.

Dashboard Page for video Equipment Redirection

• **API definition**

```
sdk.goVideoPanelPage({ deviceId, passThroughParams, redirect })
```

• **Parameter Description**

Parameter Name	Parameter Description	Type	Required
deviceId	The device ID to redirect to the panel. If not provided, the current device ID will be used.	string	No
passThroughParams	It will be parsed into a query string and appended to the address of the panel to be redirected to.	Record	No
redirect	Redirect via `redirectTo` if true, via `navigateTo` if false.	boolean	No

• **return value**

No return value.

Application Side API

Last updated: 2025-04-28 11:02:29

During the process of developing the H5 panel, we need to obtain product information, device information, or the device online status, and control the device. These capabilities are provided through the application-side API. The complete list of APIs can be referred to [Application-side API](#). The H5 custom development SDK encapsulates the process of calling the application-side API, thereby facilitating developers to make calls.

Calling Application API

The H5 SDK encapsulates the calling procedure of the application-side API. When sending a request, it will automatically carry the common parameters `AccessToken` and `RequestId`.

- **API definition**

```
sdk.requestTokenApi(action, data, options) => Promise
```

- **{parameter description}**

Parameter Name	Type	Required	Parameter Description
action	string	Yes	Specific application-side API Action Names, such as: <code>AppGetDeviceStatuses</code> .
data	object	No	Interface call parameters.
options	object	No	see wx.request .

- **return value**

- Request succeeded (code=0)

Return a resolved Promise with the value of the `Response` part of data in the application side API response.

- request failure

Return a rejected Promise with a value having a data structure of: `{ code, msg, ... detail }`.

Event Monitoring

Last updated: 2025-04-28 11:02:44

Listening for Events

- API definition

```
sdk.on(type: string, listener: (...args) => void) => void
```

- Parameter Description

Parameter Name	Type	Parameter Description
type	string	The event to be listened to.
listener	(...args) => void	Callback function when the event is triggered.

Cancel Monitoring Event

- API definition

```
sdk.off(type: string, listener: (...args) => void) => void
```

- Parameter Description

Parameter Name	Type	Parameter Description
type	string	The event whose listening is to be canceled.
listener	(...args) => void	Callback function for the event whose listening is to be canceled. If not provided, all callback functions for this event will be cleared.

WebSocket Event

wsClose Event

`close` event of WebSocket.

Parameter Name	Type	Parameter Description
code	number	Close Code sent by the server.
reason	string	The causes for the server to close a connection.

wsError Event

Error event of WebSocket.

wsControl Event

Triggered when WebSocket receives a `control` instruction.

Parameter Name	Type	Parameter Description
deviceId	string	device ID
deviceData	object	Device data

wsReport Event

Triggered when WebSocket receives the `report` instruction.

Parameter Name	Type	Parameter Description
deviceId	string	device ID
deviceData	object	Device data

wsStatusChange Event

Triggered when WebSocket receives a `wsStatusChange` instruction.

Parameter Name	Type	Parameter Description
deviceId	string	device ID
deviceStatus	number	Device online status, 0 means offline, 1 means online.

wsEventReport Event

Triggered when WebSocket receives a `wsEventReport` instruction.

Parameter Name	Type	Parameter Description
deviceId	string	Device ID.
Payload	object	Details of the device reply.

wsActionPush Event

Triggered when WebSocket receives a `wsActionPush` instruction.

Parameter Name	Type	Parameter Description
deviceId	string	device ID
Payload	object	Details specified by the issued Action.

wsActionReport Event

Triggered when WebSocket receives a `wsActionReport` instruction.

Parameter Name	Type	Parameter Description
deviceId	string	device ID
Payload	object	Action response details of the device reply.

Switching between Foreground and Background Event

appShow Event

Triggered after the mini program `App.onShow` is executed.

appHide Event

Triggered after the mini program `App.onHide` is executed.

pageShow Event

Triggered after the mini program `Page.onShow` is executed.

pageHide Event

Triggered after `Page.onHide` of the mini program is executed.

Bluetooth Module

Last updated: 2025-04-28 11:03:02

Interpretation of Terms

Noun	Meaning
serviceId	Service id, Bluetooth service uuid. When searching for devices, we mainly use <code>serviceId</code> to filter the devices we need.
deviceId	The device identifier searched by the mini program API. When connecting to a device, we mainly use <code>deviceId</code> to identify the device to be connected.
explorerDeviceId	The device ID defined on the IoT Explorer side is used as the device identifier when querying device data and submitting device data.
Bluetooth adapter	Global Singleton. Methods such as Bluetooth search and connection are declared on the instance.
Device adapter	The module actually used to connect to devices and communicate with them. Each device connection corresponds to a device adapter instance. The device adapter will be instantiated after connecting to the device and terminated after the device is disconnected. The adapter construct functions for different types of devices are distinguished according to different <code>serviceId</code> .

Bluetooth Adapter

Adding a Device Adapter

Add a device adapter. When using the Bluetooth module, you need to create a device adapter according to the specific device situation and call this API to add its construct function to the Bluetooth adapter. The H5 SDK adds a device adapter that supports the LLSync Bluetooth Protocol by default.

- API definition

```
sdk.bluetoothAdapter.addAdapter(deviceAdapter:  
DeviceAdapterConstructor) => void
```

- parameter description

Parameter Name	Parameter Description	Type	Required
deviceAdapter	The construct function of the device adapter to be added.	DeviceAdapterConstructor	Yes

- Example code

```
class DemoDeviceAdapter extends DeviceAdapter {
  static serviceId = '0000FFF0-0000-1000-8000-00805F9B34CC';
  static deviceFilter(deviceInfo) {
    if (deviceInfo.advertisServiceUUIDs) {
      const matchedServiceId =
deviceInfo.advertisServiceUUIDs.find(id => id ===
DemoDeviceAdapter.serviceId);
      if (matchedServiceId && deviceInfo.advertisData) {
        try {
          const macArr = deviceInfo.advertisData.slice(2);
          const mac = macArr.join(':');
          return {
            ...deviceInfo,
            deviceName: mac,
            serviceId: matchedServiceId,
          };
        } catch (err) {
          console.error('parse mac error', err);
        }
      }
    }
  }
  handleBLEMessage(hex) {
    return {
      type: 'unknown',
      data: hex,
    };
  }
}

sdk.bluetoothAdapter.addAdapter(DemoDeviceAdapter);
```

Initializing the Bluetooth Module

Initialize the Bluetooth module, including initializing the Bluetooth module, establishing Bluetooth communication between mini programs, and registering a global callback, etc. This interface can be called repeatedly and can be called before using the Bluetooth module each time.

- **API definition**

```
sdk.bluetoothAdapter.init() => Promise<void>
```

- **Return value**

Return a cached Promise that resolves after successful initialization. If the initialization is incomplete or has already been successful, calling it multiple times will return the same Promise. If the initialization fails, the cached Promise will be released after rejection, and reinitializing will occur upon subsequent calls.

- **Example code**

```
sdk.bluetoothAdapter.init().then(() => {  
  // Call Bluetooth module capabilities  
});
```

Start Searching for Bluetooth Devices

Start searching for bluetooth devices. (This will call [wx.startBluetoothDevicesDiscovery](#), which consumes system resources. Make sure to call [Stop searching for bluetooth devices](#) when searching is no longer needed, such as after exiting the page.)

- **API definition**

```
sdk.bluetoothAdapter.startSearch({  
  serviceId?: string,  
  serviceIds?: string[],  
  ignoreDeviceIds?: string[],  
  onSearch: (DeviceInfo[]) => void,  
  onError: (Error) => void,  
  timeout: number  
}) => Promise<void>
```

- **parameter description**

Parameter Name	Parameter Description	Type	Required
serviceld	Specify the serviceld to be searched. If not provided, the servicelds of all currently supported DeviceAdapters will be used for matching.	string	No
servicelds	Parameter description is the same as serviceld.	string[]	No
ignoreDevices	Selectable id list of devices that need to be filtered out (for example, the devices that have just been added). These devices will not occur in the search result.	string[]	No
onSearch	Call after the search result is updated. Return the searched device list.	(DeviceInfo[]) => void	Yes
onError	Call when an error occurs during the search process. Device search will be terminated after being triggered.	(Error) => void	Yes
timeout	Optional; default 20000 milliseconds. A timeout error will be triggered if no device is found after exceeding the specified duration.	number	Yes

- **Return value**

Return a Promise.

Stop Searching for Bluetooth Devices

Interface Definition

```
sdk.bluetoothAdapter.stopSearch() => void
```

Searching for a Single Bluetooth Device

Start searching for bluetooth devices and stop searching after the first eligible device is found.

- **API definition**

```
sdk.bluetoothAdapter.searchDevice({
  deviceName: string,
```

```
serviceId?: string,  
serviceIds?: string[],  
ignoreDeviceIds?: string[]  
}) => Promise<DeviceInfo>
```

• Parameter Description

Parameter Name	Parameter Description	Type	Required
deviceName	Specify the deviceName of the device to search.	string	Yes
serviceId	Specify the serviceId to be searched. If not provided, the serviceIds of all currently supported DeviceAdapters will be used for matching.	string	No
serviceIds	Parameter description is the same as serviceId.	string[]	No
ignoreDeviceIds	Selectable id list of deviceIds that need to be filtered out (for example, the devices that have just been added). These devices will not occur in the search result.	string[]	No

• Return value

Return a Promise that resolves after finding the first eligible device.

Connecting to a Bluetooth Device

Connect to the specified bluetooth device.

• API definition

```
blueToothAdapter.connectDevice(deviceInfo: DeviceInfo, options?: {  
  autoNotify?: boolean }) => Promise<DeviceAdapter>
```

• Parameter Description

Parameter Name	Parameter Description	Type	Required
deviceInfo	Import Start searching for bluetooth devices or Search for a single Bluetooth device API	DeviceInfo	Yes

	search results for DeviceInfo.		
options.autoNotify	Optional, the default is true. When set to true, after connecting to the device, it will automatically pull the service list and the characteristic value list under the primary service, and will automatically subscribe to the notify of the first notifyId or indicateId characteristic value. If the device contains multiple services or multiple notify characteristic values, please pass false and manually obtain and subscribe to the characteristic values through methods such as getBLEDeviceServices, getBLEDeviceCharacteristics, and notifyBLECharacteristicValueChange.	boolean	No

- Return value

Return a Promise that returns the device adapter once the connection is successful.

Get Device Adapter Instance

Search for the corresponding device adapter instance according to `deviceId`.

- API definition

```
sdk.bluetoothAdapter.getDeviceAdapter({ explorerDeviceId: string }) =>
DeviceAdapter
```

- Parameter Description

Parameter Name	Parameter Description	Type	Required
explorerDeviceId	The explorerDeviceId of the device adapter instance can be obtained through <code>sdk.deviceId</code> .	string	Yes

- Return value

Return the device adapter instance that matches the incoming `explorerDeviceId`. If not found, return `undefined`.

Reporting Device Information

Bluetooth devices cannot report the mac address and other information of the device directly through mqtt. Therefore, it is necessary for the H5 side to report. This corresponds to the device information in the device details.



Note:

The manufacturer name and product model inside the picture are filled in on the console during mass production of the device. The mac address, firmware version, etc. are reported by the H5 client.

• API definition

```
sdk.blueToothAdapter.reportDeviceInfo({ productId: string, deviceName: string, deviceInfo: any }) => Promise;
```

• parameter description

Parameter Name	Parameter Description	Type
productId	product ID.	string
deviceName	device name.	string
deviceInfo	Equipment information.	any, for details, see the following example

```
deviceInfo: {
```

```
"module_hardwareinfo": "Specific hardware model N10 of the module"
"module_softwareinfo": "module software version"
"fw_ver": "mcu firmware version"
"imei": "IMEI number of the device, optional to report"
"mac": "MAC address of the device, optional to report"
"device_label": {
  "append_info": "custom additional information of the product by
the equipment vendor"
}
```

Listening to Bluetooth Adapter Events

Listen to Bluetooth adapter events.

- **API definition**

```
sdk.bluetoothAdapter.on(type: string, listener: (...args) => void) =>
void
```

- **parameter description**

Parameter Name	Parameter Description	Type	Required
type	Events to be listened to.	string	Yes
listener	Callback function for event triggering.	(...args) => void	Yes

Cancel Listening to Bluetooth Adapter Events

Cancel listening to Bluetooth adapter events.

- **API definition**

```
sdk.bluetoothAdapter.off(type: string, listener: (...args) => void) =>
void
```

- **parameter description**

Parameter Name	Parameter Description	Type	Required
----------------	-----------------------	------	----------

type	Events to cancel listening to.	string	Yes
listener	The callback function of the event to cancel listening to. If not provided, all callback functions of the event will be cleared.	(...args) => void	No

Bluetooth Adapter Events

adapterStateChange event: Triggered when the adapter state changes.

Parameter Name	Parameter Description	Type
available	Whether the Bluetooth adapter is available	boolean
discovering	If the Bluetooth adapter is in the search state	boolean

Device Adapter

Custom Device Adapter

Custom Device Adapter Class needs to inherit `DeviceAdapter` and supplement the following implementation.

- **serviceld**: This attribute needs to be set for the custom device adapter class. It represents the Primary Service ID of the device.
- **deviceFilter**: The custom device adapter class needs to implement this static method. When searching for bluetooth devices, the information of each searched device will be passed into this method. If it is judged to be a device of this product, the unique device identifier `deviceName` and `serviceld` need to be returned outside the input parameter `deviceInfo`; otherwise, return null.

```
DeviceAdapter.deviceFilter: (deviceInfo: DeviceInfo) => {  
  deviceName: string,  
  serviceId: string,  
  ...deviceInfo  
}
```

- **handleBLEMessage**: This method needs to be implemented by the Custom Device Adapter Class for processing protocol parsing after receiving the `onBLECharacteristicValueChange` callback.

```
DeviceAdapter.handleBLEMessage: (hexString, { serviceId,
characteristicId }) => {
  reportData?: any,
  ...any
}
```

If the return value contains `reportData`, this part of data will be reported to the cloud (Note: it needs to match the Thing Model defined in the product). Other fields will be passed through in the payload of the `message` event.

Device Adapter Properties

Attribute Name	Attribute Description	Type
explorerDeviceld	Read-only, the explorerDeviceld of the device.	string
isConnected	Read-only, whether the device is currently connected.	boolean
deviceld	Read-only, the deviceld of the device.	string
serviceId	Read-only, the Primary Service ID of the device, consistent with the static property DeviceAdapter.serviceId in the construct function.	string
originName	Read-only, the original name of the device, i.e., the name field when the Mini Program API is searched.	string

Disconnect Device Connection

API definition

```
deviceAdapter.disconnectDevice() => void
```

Get the Device Service List

• API definition

```
deviceAdapter.getBLEDeviceServices() => Promise<ServiceList>
```

- **Return value**

Return a Promise. For the `ServiceList` data structure, please see [wx.getBLEDeviceServices](#).

Get Eigenvalue List of Specified Service for Device

Obtain the eigenvalue list in the specified services of the device, and store the eigenvalues in the `deviceAdapter` instance.

- **API definition**

```
deviceAdapter.getBLEDeviceCharacteristics({ serviceId: string }) =>
Promise<CharacteristicsList>
```

- **Parameter Description**

Parameter Name	Parameter Description	Type	Required
serviceId	Selectable. Specify the Service ID for which to obtain the eigenvalue list. The default is the Primary Service ID.	string	No

- **Return value**

Return a Promise. For the `CharacteristicsList` data structure, please see [wx.getBLEDeviceCharacteristics](#).

- **Description**

Store the obtained eigenvalues in the `deviceAdapter` instance according to the following data structure.

```
deviceAdapter.characteristicsMap[serviceId] = {
  notifyIds: string[],
  indicateIds: string[],
  writeIds: string[],
  readIds: string[]
}
```

Negotiate Bluetooth Maximum Transmission Unit Settings

This interface is only supported on the Android system. It is not supported on iOS due to system limitations.

- **API definition**


```
deviceAdapter.setBLEMTU({ mtu: number }) => Promise
```

- **Parameter Description**

Parameter Name	Parameter Description	Type	Required
mtu	Maximum transmission unit. Setting range is within the interval (22,512), unit: bytes.	number	Yes

- **Return value**

Return a Promise.

Read Binary Data of Specified Eigenvalue

- **API definition**

```
deviceAdapter.readBLECharacteristicValue({  
  serviceId?: string,  
  characteristicId: string  
}) => Promise<void>
```

- **Parameter Description**

Parameter Name	Parameter Description	Type	Required
serviceId	Selectable. The default is the Primary Service ID.	string	No
characteristicId	The eigenvalue ID to be read will, by default, be the first read eigenvalue under the primary service.	string	Yes

- **Return Value**

Return a Promise. The information obtained from the API needs to be retrieved in the callback registered in the `onBLECharacteristicValueChange` method. Refer to [wx.readBLECharacteristicValue](#) for details.

Get Bluetooth Device Signal Strength

- **API definition**

```
deviceAdapter.getBLEDeviceRSSI() => Promise
```

- **Return value**

Return a Promise. For the specific data structure, please see [wx.getBLEDeviceRSSI](#).

Enable the notify Feature When the Eigenvalue Changes

- **API definition**

```
deviceAdapter.notifyBLECharacteristicValueChange({  
  characteristicId?: string,  
  serviceId?: string,  
  state?: boolean  
}) => Promise<void>
```

- **parameter description**

Parameter Name	Parameter Description	Type	Required
serviceId	The required subscribed service ID will, by default, be the Primary Service ID.	string	No
characteristicId	The eigenvalue ID required for subscription. By default, the first notify or indicate eigenvalue under the primary service will be taken.	string	No
state	Whether to enable notify. The default value is true.	boolean	No

- **Return Value**

Return a Promise.

Write Binary Data to Specified Eigenvalue

- **API definition**

```
deviceAdapter.write(hexString: string, options?: {  
  writeId?: string,  
  serviceId?: string  
})
```

```
}) => Promise
```

• Parameter Description

Parameter Name	Parameter Description	Type	Required
hexString	A hexadecimal string required for the bluetooth device.	string	Yes
options.writeId	Optional. The eigenvalue ID to be written. By default, the first writeld under the primary service will be taken.	string	No
options.serviceId	Optional. The service ID to be written. By default, it will be the Primary Service ID.	string	No

Listen for Events

You can use `deviceAdapter.on` to [listen to device adapter events](#).

• API definition

```
deviceAdapter.on(type: string, listener: (...args) => void) => void
```

• parameter description

Parameter Name	Parameter Description	Type	Required
type	Events to be listened to.	string	Yes
listener	Callback function for event triggering.	(...args) => void	Yes

Cancel Event Listening

Cancel listening to device adapter events.

• API definition

```
deviceAdapter.off(type: string, listener: (...args) => void) => void
```

• Parameter Description

Parameter Name	Parameter Description	Type	Required
type	Events to cancel listening to.	string	Yes
listener	The callback function of the event to cancel listening to. If not provided, all callback functions of the event will be cleared.	(...args) => void	No

Device Adapter Events

- **connect event:** Triggered after device connection.
- **disconnect event:** Triggered after device disconnection.
- **message event:** Triggered when `onBLECharacteristicValueChange` callback is received and processed by `handleBLEMessage`.

Parameter Name	Parameter Description	Type
timestamp	The timestamp when the device message was received, in milliseconds.	number
dataReported	Whether the messages received from the device have been reported to the cloud.	boolean
(others)	Other parameters returned by the <code>handleBLEMessage</code> function will be passed through to the message event.	any

- **BLEConnectionStateChange event:** Triggered when `onBleConnectionStateChange` is triggered. If `connected` is true, the `connect` event will be triggered next; otherwise, the `disconnect` event will be triggered.

Parameter Name	Parameter Description	Type
connected	Whether the device is connected.	boolean

Standard Bluetooth Protocol

The binding process of LLSync standard Bluetooth protocol devices is carried out in the mini program. Developers only need to focus on device connection, control operations, and event monitoring in the customized H5 panel. The customized H5 panel has added the device adapter (`StandardDeviceAdapter`) for the LLSync standard Bluetooth protocol by default and

supports communication with devices through the LLSync standard Bluetooth protocol. The H5 panel Demo provides a [standard Bluetooth protocol demo](#) for developers' reference.

Search and Connect a Device

The process of establishing a connection with standard Bluetooth protocol devices consists of three steps: searching for devices, connecting to devices, and connection authentication.

Sample code is as follows:

```
import { StandardDeviceAdapter } from 'qcloud-iotexplorer-h5-panel-sdk';

Initialize the Bluetooth adapter
await sdk.bluetoothAdapter.init();

// Search for devices
const deviceInfo = await sdk.bluetoothAdapter.searchDevice({
  deviceName: sdk.deviceName,
  serviceId: StandardDeviceAdapter.serviceId,
  productId: sdk.productId,
  disableCache: true,
});

if (!deviceInfo) {
  throw new Error('Device not found');
}

// Connect to a device
const deviceAdapter = await sdk.bluetoothAdapter.connectDevice({
  ...device,
  productId: sdk.productId,
});

// Connect authentication
if (!deviceAdapter.authorized) {
  await deviceAdapter.authenticateConnection({
    deviceName: sdk.deviceName,
  });
}
```

Device Communication

After establishing a connection with the device through the above process, the H5 panel can control the device and receive the attributes and event reports of the device.

- Control device

For API definition, please see [Control Device Attributes](#) . Sample code is as follows:

```
sdk.controlDeviceData({
  // Device property to be controlled
  power_switch: 1
});
```

- Monitor device events

For API definition, please see [event monitoring](#) . Sample code:

```
// Monitoring device attribute reporting
sdk.on('wsReport', ({ deviceId, deviceData }) => {
  console.log('device', deviceId, 'report_property', deviceData);
});

// Listen to device event reporting
sdk.on('wsEventReport', ({ deviceId, Payload }) => {
  console.log('device', deviceId, 'report_event', Payload);
});
```

Device Adapter Properties

Attribute Name	Attribute Description	Type
explorerDeviceld	Read-only, the Deviceld of the device in IoT Explorer.	string
isConnected	Read-only, whether the device is currently connected.	boolean
deviceld	Read-only, the deviceld provided by the Mini Program Bluetooth API.	string
serviceld	Read-only, the Primary Service ID of the device, consistent with the static property DeviceAdapter.serviceld in the construct function.	string
originName	Read-only, the original name of the device, i.e., the name	string

	field when the Mini Program API is searched.	
bleVersion	Read-only, LLSync protocol version number.	number
mtu	Read-only, the maximum transmission unit (MTU) size required by device settings.	number
otaVersion	Read-only, the firmware version number of the device.	string
extendInfo.macStr	Read-only, the mac address of the device.	string
authorized	Read-only, whether authentication with the device is completed.	boolean

BLE+WIFI Dual-Mode Communication

For the Wi-Fi + BLE Combo module, provides device-side SDK and H5 SDK. Supports communication between the mini program and the device via the Standard Bluetooth Protocol in the event of Wi-Fi offline status, providing users with a better experience when Wi-Fi is disconnected. For the device-side SDK, refer to [Development Guide](#).

For custom H5 panels, developers do not need to pay attention to the network configuration process. Developers need to focus on in the panel:

1. Listen to the device online status. Check whether the wifi connection is normal.
2. Enable Bluetooth to communicate when the device is offline. Separate the following descriptions.

Listen to Device Online Status

We can sense the online and offline status of the device by listening to the `wsStatusChange` event in [WebSocket events](#).

```
sdk.on('wsStatusChange', ({deviceId, deviceStatus}) => {  
  if (deviceStatus === 0) {  
    // The device is offline. Start to enable the bluetooth  
    connection to communicate, see step 2  
  }  
})
```

Enable Bluetooth Communication

We provide a [Dual-channel Communication demo](#) for reference.

1. Add `BleComboDualModeDeviceAdapter4H5`

BleComboDualModeDeviceAdapter4H5 is a device adapter instance. If you don't know about device adapters yet, please read [Device Adapter](#).

```
import { BleComboDualModeDeviceAdapter4H5 } from 'qcloud-iotexplorer-appdev-plugin-wificonf-blecombo/lib/protocols/BleComboDualMode';

const sdk = window.h5PanelSdk;
BleComboDualModeDeviceAdapter4H5.injectOptions({ appDevSdk:
sdk.appDevSdk });

sdk.blueToothAdapter.addAdapter(BleComboDualModeDeviceAdapter4H5,
true);
```

2. Search for devices

For the parameters of `searchDevice`, see the parameter description section for searching for a single Bluetooth device in [Bluetooth adapter](#).

```
await blueToothAdapter.init();
console.log('Start search for device', sdk.deviceName);
const deviceInfo = await blueToothAdapter.searchDevice({
  deviceName: sdk.deviceName,
  serviceId: BleComboDualModeDeviceAdapter4H5.serviceId,
  productId: sdk.productId,
  disableCache: true,
});
```

3. Connect devices

Using the `deviceInfo` obtained in the previous step, we can call `connectDevice` to connect the device to get the `deviceAdapter`. For the input parameter definition, see [Connect to the Bluetooth device](#) section.

```
const deviceAdapter = await blueToothAdapter.connectDevice({
  ...device,
  productId: sdk.productId,
});
console.log('deviceAdapter:', deviceAdapter);
// After being authorized, control data can be sent to the device.
if (!deviceAdapter.authorized) {
```



```
await deviceAdapter.authenticateConnection({
  deviceName: sdk.deviceName,
});
}
```

Once the Bluetooth connection between the mini program and the device is successful, the panel can control the device. The sdk will send control data to the device via Bluetooth. For example, enable/disable control:

```
sdk.controlDeviceData({ power_switch: 1 });
```

ASR Speech Recognition

Last updated: 2025-04-28 11:03:21

Speech Recognition

Currently, two scenarios are supported, namely "recording file" and "single-sentence recognition". The input parameters will vary in these two scenarios.

Since the identification process is asynchronous, the API "voiceRecognition" will not immediately return the recognition result. It can be understood that this API creates an "async task". After this successfully created "async task" is executed, the result will be pushed to all terminals subscribing to the deviceId via websocket. The following context gives you a detailed introduction to ASR speech recognition.

- **API definition**

```
sdk.voiceRecognition({...}) => Promise<{...}>
```

- **Parameter Description**

- Scenario support, size limit, and relevant fields for the "recording file" audio type. For details, see [recording file recognition](#).
- About the audio types, size limits, and relevant fields supported in the "single-sentence recognition" scenario. For details, see [single-sentence recognition](#).

- **common parameter**

Parameter Name	Type	Required	Parameter Description
DeviceId	string	No	By default, use the device ID of the current device.
AudioType	string	Yes	Recognition scenario. • The parameter value of "recording file" is "file". • The parameter value of "single-sentence recognition" is "sentence".
Data	• Blob • File	Yes	Audio file.
ResourceName	string	No	Filename. If the Data type is File, take its "name" as the default value.

EngineType	string	No	Engine model type, default value is "16k_zh".
FilterDirty	number	No	Whether to filter profanity.
FilterModal	number	No	Whether to filter modal particles.
FilterPunc	number	No	Whether to filter punctuation.
ConvertNumMode	number	No	Whether to perform intelligent conversion of Arabic numerals.

- **additional parameter of recording file**

Parameter Name	Type	Required	Parameter Description
ChannelNum	number	No	Number of sound channels, defaults to 1.
SpeakerDiarization	number	No	Whether to enable speaker separation.
SpeakerNumber	number	No	Number of speaker separation users.

- **return value**

Parameter Name	Type	Parameter Description
ResourceToken	string	Unique identifier of an audio file under a certain device.

Listening to Recognition Results

For the "recording file" scenario, if the audio file is too large, the backend may perform shard recognition on the audio file. After each shard recognition is completed, a websocket message will be pushed, but the order of the pushed messages is not guaranteed (for example, the result of shard 2 may arrive first).

For the "asrResponse" event, it is actually a secondary encapsulation based on the "wsControl" event; of course, you can also obtain the recognition result by listening to the "wsControl" event.

- **API definition**

```
sdk.on('asrResponse', ({ deviceId, data }) => void)
```

• Return Value Description

Parameter Name	Type	Parameter Description
deviceId	string	Device ID.
data	object	Recognition result data.
data.resource_token	string	Unique identifier of an audio file under a certain device.
data.result_code	number	Status code: 0 means success.
data.total_num	number	Total number of shards.
data.seq	number	Current shard number.
data.res_text	string	The current fragmented recognition result, for the "recording file" scenario, will include segment timestamps in the recognition result.

Get the File Download Link for the Speech

Note:

Only in the "recording file" scenario.

• API definition

```
sdk.getAsrDownloadUrl({...}) => Promise<{...}>
```

• Parameter Description

Parameter Name	Type	Required	Parameter Description
DeviceId	string	Yes	Device ID.
ResourceT	string	Yes	ResourceToken returned by calling

oken		voiceRecognition.
------	--	-------------------

• Return value description

Parameter Name	Type	Parameter Description
ResourceURL	string	cos access link.

TRTC Audio and Video

Last updated: 2025-04-28 11:03:35

API Description

Tencent Cloud IoT Explorer tencent real-time communication (TRTC) service. Call this API to navigate from h5 to Tencent Lianlian Mini Program TRTC page.

API Definition

```
 sdk.TRTCManager.goTRTCPage()
```

Parameter Description

Parameter Name	Type	Required	Parameter Description
callType	string	Yes	Call type: - video - audio

Cloud Storage Service

Last updated: 2025-04-28 11:03:52

Cloud Storage Service

Add video Event Cloud Storage capacity on the Explorer platform. Associate Event Cloud Storage with event correlation in the Thing Model. It is convenient for users to query corresponding cloud-stored pictures when performing log queries.

API request method

Call the API interface through the application-side API `sdk.requestTokenApi(action, data, options)` => Promise.

For instructions, see [Application-side API](#).

Pull Cloud Storage Event List

- API definition

```
sdk.requestTokenApi("IotVideoDescribeCloudStorageEvents", {  
  ProductId: string,  
  DeviceName: string,  
  StartTime?: number,  
  EndTime?: number,  
  Context?: string,  
  Size?: number,  
  EventId?: string  
}) => Promise;
```

- Parameter Description

See [Pull Cloud Storage Event List](#).

Pull Cloud Storage Event Thumbnail

- API definition

```
sdk.requestTokenApi("IotVideoDescribeCloudStorageThumbnail", {  
  ProductId: string,  
  DeviceName: string,  
  Thumbnail: string  
}) => Promise;
```

- **{parameter description}**

See [Pull Cloud Storage Event Thumbnail](#).

Get Dates with Cloud Storage

- **API definition**

```
sdk.requestTokenApi("IotVideoDescribeCloudStorageDate", {  
  ProductId: string,  
  DeviceName: string  
}) => Promise;
```

- **Parameter Description**

See [Retrieve the date with cloud storage](#).

Get Cloud Storage Timeline for a Certain Day

- **API definition**

```
sdk.requestTokenApi("IotVideoDescribeCloudStorageTime", {  
  ProductId: string,  
  DeviceName: string,  
  Date: string,  
  StartTime?: number,  
  EndTime?: number  
}) => Promise;
```

- **{parameter description}**

See [Retrieve the Cloud Storage Timeline for a Certain Day](#).

Get Video Anti-Hotlinking Playback URL

- **API definition**

```
sdk.requestTokenApi("IotVideoGenerateSignedVideoURL", {  
  ProductId: string,  
  VideoURL: string,  
  ExpireTime: number  
}) => Promise;
```


- **Parameter Description**

Concatenate VideoURL, use the return of the previous API, add start time and end time: `https://zylcb.iotvideo.tencentcs.com/timeshift/live/xxxx/timeshift.m3u8?starttime_epoch={StartTime}&endtime_epoch={EndTime}`

for example: `https://zylcb.iotvideo.tencentcs.com/timeshift/live/xxxx/timeshift.m3u8?starttime_epoch=1631849895&endtime_epoch=1631849907` For other parameters, please see [Get playback URL for video anti-hotlinking](#).

Pull Image Streaming Data

- **API definition**

```
sdk.requestTokenApi("IotVideoDescribeCloudStorageStreamData", {
  DeviceName: string,
  ProductId: string,
  StartTime: string
}) => Promise;
```

- **Parameter Description**

Parameter Name	Description	Type	Required
DeviceName	Device name.	String	Yes
ProductId	Product ID.	String	Yes
StartTime	Start time.	String	Yes

- **Return Value**

Parameter Name	Description	Type
RequestId	ID of the sent request.	String
VideoStream	Image stream video address.	String
AudioStream	Image stream audio address.	String

Music Service

Last updated: 2025-04-28 11:04:15

Tencent Lianlian custom H5 SDK provides QQ Music API capabilities, including retrieval of basic information, playback links, singer information, playlist information, personalized recommended songs, recently played songs, and latest songs of QQ Music songs. It can be used to support on-demand playback of music content within the H5 panel.

API Calling Method

```
window.h5PanelSdk.qqMusic.<api name>
```

For example:

```
window.h5PanelSdk.qqMusic.goAuthPage()
```

Note:

Except for "login authorization" and "login-free API", other APIs can be called only after login authorization is completed. You can call [redirect to QQ Music login authorization page](#) to complete login authorization.

If you call a login-required API in a not logged in state, an error will be reported `InvalidParameterValue.OAuthClientNotExist`.

Login Authorization

After logon authorization is completed, the H5 SDK and device can use the listening data and membership benefits of the logged-in QQ Music account.

Querying QQ Music Authorization Status

Check whether the current user has completed QQ Music login authorization.

```
getAuthStatus(): Promise<boolean>
```

Input Parameter

None

Output Parameter

```
boolean
```

Redirect to QQ Music Login Authorization Page

After calling this API, it will navigate to Tencent Lianlian Mini Program to complete QQ Music login authorization.

```
goAuthPage(): void
```

Input Parameter

None

Output Parameter

None

Song

Batch Obtain Song Details Info

Retrieve the details of one or more songs by song ID or song mid. When both `SongIds` and `SongMids` are provided, priority using `SongMids`.

```
describeSongInfoBatch(params: object): Promise<QQMusicSongInfo[]>
```

Input Parameter

Parameter Name	Parameter Type	Parameter Description
params.SongIds	number[]	song id array
params.SongMids	string[]	song mid array

Output Parameter

```
Promise<QQMusicSongInfo[]>
```

Get Song Lyrics

Retrieve the lyrics of a song by song ID or song mid. When both `SongId` and `SongMid` are provided, priority using `SongMid`.

```
describeLyric(params: object): Promise<DescribeLyricResp>
```

Input Parameter

Parameter Name	Parameter Type	Parameter Description
params.SongId	number	song id
params.SongMid	string	song mid

Output Parameter

```
Promise<DescribeLyricResp>
```

Get Recent Operational New Songs

```
describeNewTrack(params: object): Promise<QQMusicNewTrackSongItem>
```

Input Parameter

Parameter Name	Parameter Type	Parameter Description
params.Tag	number	Category (12: Mainland; 9: Korea; 13: Hong Kong/Macao/Taiwan; 3: Europe and America; 8: Japan; 1: Latest)

Output Parameter

```
Promise<QQMusicNewTrackSongItem>
```

Device Control

Construct IoT Device Playback List

Convert QQ Music song information to the playback list data format for IoT devices.

```
pickSongInfoForSync(list, options): QQMusicSyncSongEntry[]
```

Input Parameter

Parameter Name	Parameter Type	Parameter Description
list.SongId	number	song ID
list.SongName	string	song title
list.SingerName	string	singer name
list.SongPlayUrlDolby	string	Dolby quality playback link
list.SongPlayUrlXq	string	Hi-Res quality playback link
list.SongPlayUrlSq	string	Ultra-quality playback link
list.SongPlayUrlHq	string	High-quality playback link
list.SongPlayUrlStandard	string	Standard quality playback link
list.SongPlayUrl	string	Smooth quality playback link
list.Try30SUrl	string	Audition playback link
list.TryBegin	number	Audition start time (ms)
list.TryEnd	number	Audition end time (ms)
list.UnplayableCode	number	Unplayable status code
list.UnplayableMsg	string	Unplayable prompt message
list.SongPlayTime	number	Song duration (s)
options.quality	QQMusicQualityType	Preferred sound quality
options.necessarySongId	number	Optional parameter. If the song corresponding to the id cannot be played (UnplayableCode!= 0), an

exception will be thrown.

Output Parameter

```
QQMusicSyncSongEntry[]
```

Deliver Playback List to IoT Device

```
syncSongList(params: object): Promise<void>
```

Input Parameter

Parameter Name	Parameter Type	Parameter Description
params.Id	string	Playback list ID
params.Name	string	Playback list title
params.Type	QQMusicPlayListType	Playback list type
params.Quality	QQMusicQualityType	Currently selected sound quality
params.Total	number	Total number of songs in the current playlist
params.Page	number	Current playlist page number
params.PageSize	number	Page length
params.SongList	QQMusicSyncSongEntry[]	Playback list (can call pickSongInfoForSync to generate)
params.PlaySongId	number	song ID of the song to start playing

Output Parameter

```
Promise<void>
```

Song List

Retrieve Personal Song List Directory

```
describeSelfSongList(): Promise<QQMusicSelfSongList[]>
```

Input Parameter

None

Output Parameter

```
Promise<QQMusicSelfSongList[]>
```

Retrieve the Song List in the Song List

```
describeSongList(params: object): Promise<DescribeSongListResp>
```

Input Parameter

Parameter Name	Parameter Type	Parameter Description
params.DissId	number	song list ID
params.Page	number	page number (starting from 0)
params.PageSize	number	Page length (maximum 30)

Output Parameter

```
Promise<DescribeSongListResp>
```

Retrieve, Favorite, Unfavorite Song Lists

Perform collection and cancellation operations on the song list.

```
operateSquareSongList(params: object):  
Promise<OperateSquareSongListResp>
```

Input Parameter

Parameter	Parameter	Parameter Description
-----------	-----------	-----------------------

Name	Type	
params.Dissl d	number	song list id of the operated song list
params.Op	number	Type of operation performed (1: Favorite, 2: Unfavorite, 3: Get favorite song lists)

Output Parameter

```
Promise<OperateSquareSongListResp>
```

MV

Get Latest or Hottest MVs

```
describeMVList(params: object): Promise<QQMusicMVItem[]>
```

Input Parameter

Parameter Name	Parameter Type	Parameter Description
params.MV Type	number	MV type (0: Get the latest MVs; 1: Get the hottest MVs)
params.MV Tag	number	MV Tag (0: Recommend; 12: Mainland; 9: Korea; 13: Hong Kong and Taiwan; 3: Europe and America; 8: Japan)

Output Parameter

```
Promise<QQMusicMVItem[]>
```

Batch Obtain MV Details

Retrieve the details of one or more MVs by MV id or MV vid. Support two kinds of id parameters, `MVIds` and `MVVids`, for query; preferentially use `MVVids` when both are provided.

```
describeMV(params: object): Promise<QQMusicMVInfo[]>
```


Input Parameter

Parameter Name	Parameter Type	Parameter Description
params.MVIds	number[]	song id array
params.MVVids	string[]	song mid array

Output Parameter

```
Promise<QQMusicMVInfo[]>
```

Search

Search for Songs, Albums, MVs

Search for songs, albums, MVs, or radio stations based on input keywords.

```
searchMusic(params: object): Promise<SearchMusicResp>
```

Input Parameter

Parameter Name	Parameter Type	Parameter Description
params.KeyWord	string	search keywords
params.SearchType	number	Search type (0: single, 8: album, 12: MV, 15: radio station)
params.Page	number	Page number (starting from 1, maximum 4 pages)
params.Num	number	Page length (maximum 50)

Output Parameter

```
Promise<SearchMusicResp>
```

Singer

Search Singer List

Search for singers based on input keywords.

```
searchSingerList(params: object): Promise<QQMusicSearchSingerItem>
```

Input Parameter

Parameter Name	Parameter Type	Parameter Description
params.Keyword	string	Search keywords
params.Page	number	Page number (starting from 1, maximum 2 pages)
params.PageSize	number	Page length (maximum 20)

Output Parameter

```
Promise<QQMusicSearchSingerItem>
```

Get a Popular Singer List

Pull the popular singer id list under the corresponding category according to the category index (supporting region, gender, genre).

```
describeSingerList(params: object): Promise<QQMusicPopularSingerItem[]>
```

Input Parameter

Parameter Name	Parameter Type	Parameter Description
params.Area	number	Singer's located region index (-100: All 200: Mainland 2: Hong Kong and Taiwan 3: Korea 4: Japan 5: Europe and America)
params.Type	number	Singer gender index (-100: All 0: Male 1: Female 2: Composite)
params.Genre	number	Singer's genre index (-100: All 1: Popular 2: Rock 3: Folk 4: Electronic 5: Jazz 6: Hip-hop 8: R&B 9: Light music 10:

Folk song 14: Classical 19: Chinese style 20: Blues 25: Country)

Output Parameter

```
Promise<QQMusicPopularSingerItem[]>
```

Retrieve Singer Song Information

Search for song information under a singer by singer id.

```
describeSinger(params: object): Promise<QQMusicSingerInfo>
```

Input Parameter

Parameter Name	Parameter Type	Parameter Description
params.SingerId	number	singer id
params.Page	number	page number (starting from 0)
params.PageSize	number	Page length (maximum 50)
params.Order	number	Song sorting method (0: By Time, 1: By Popularity)
params.Wiki	number	Whether singer wiki info is required (0: Not required, 1: Required)

Output Parameter

```
Promise<QQMusicSingerInfo>
```

Recent Playback

Get Recent Playback List

Obtain the user's recent playback list. The maximum support is the most recent 100 records. (Under the All tab, it does not include the detail of the song's historical playback. Need to pass through the Song tab to view details.)

```
describeRecentPlay(params: object): Promise<DescribeRecentPlayResp>
```

Input Parameter

Parameter Name	Parameter Type	Parameter Description
params.Type	number	tab type (1: All, 2: Song, 3: Album, 4: Song list)
params.UpdateTime	number	Latest update time is sent by the api (see response data). The client inputs it for incremental retrieval of data to reduce the amount of data returned. Initially, 0 can be input.

Output Parameter

```
Promise<DescribeRecentPlayResp>
```

Reporting Recent Playback

Submit the user's recent playback records.

```
reportRecentPlay(params: object): Promise<void>
```

Input Parameter

Parameter Name	Parameter Type	Parameter Description
params.ResourceId	number	resource id
params.Type	number	resource type (2: song, 3: album, 4: song list)

Output Parameter

```
Promise<void>
```

Recommendations

Retrieve Home Page Recommendation

Get content according to the recommended content type of the input.

```
describeHomepageSongList(params: object):  
Promise<DescribeHomepageSongListResp>
```

Input Parameter

Parameter Name	Parameter Type	Parameter Description
params.SN	string	machine code / serial number / unique ID
params.Type	string	Content type to be obtained (200: single; 500: song list; multiple selections allowed, for example: 200,500)

Output Parameter

```
Promise<DescribeHomepageSongListResp>
```

Get Personalized Recommended Songs

Retrieve songs recommended by the personalized algorithm, that is, the personalized radio station.

```
describeIndividualRadio(): Promise<QQMusicIndividualRadioSongItem[]>
```

Input Parameter

None

Output Parameter

```
Promise<QQMusicIndividualRadioSongItem[]>
```

Get 30 Recommended Songs Daily

```
describeDailySongs(): Promise<QQMusicDailySongItem[]>
```

Input Parameter

None

Output Parameter

```
Promise<QQMusicDailySongItem[]>
```

Ranked List

Retrieve the Ranked List

Retrieve the song list of QQ Music rankings.

```
describeTopList(): Promise<QQMusicTopListGroupItem[]>
```

Input Parameter

None

Output Parameter

```
Promise<QQMusicTopListGroupItem[]>
```

Retrieve the Detailed Information of the Rankings List

Retrieve the songs of QQ Music rankings.

```
describeTopListInfo(params: object): Promise<DescribeTopListInfoResp>
```

Input Parameter

Parameter Name	Parameter Type	Parameter Description
params.TopId	number	Ranking list id
params.Page	number	page number (starting from 0)
params.PageSize	number	Page length (maximum 50)

Output Parameter

```
Promise<DescribeTopListInfoResp>
```

No-Login Zone

Unlogged-In User Obtaining Playback List

You can retrieve the playable song list when the user is not logged in. This API does not issue playback links.

```
describeFreeSongList(): Promise<QQMusicFreeSongGroupItem>
```

Input Parameter

None

Output Parameter

```
Promise<QQMusicFreeSongGroupItem>
```

Unlogged-In User Obtaining Playback Link

You can obtain the playback links of playable songs when the user is not logged in.

```
describeFreeSongInfo(params: object): Promise<QQMusicFreeSongInfo>
```

Input Parameter

Parameter Name	Parameter Type	Parameter Description
params.SongId	number	song id
params.SongToken	string	song token (callable via describeFreeSongList to obtain)

Output Parameter

```
Promise<QQMusicFreeSongInfo>
```

Data Structure

QQMusicSongInfo

```
interface QQMusicSongInfo {  
  SongId: number;
```

```
SongMid: string;
SongName: string;
SongTitle: string;
IsOnly: number;
Vip: number;
Language: string;
Genre: string;
PublicTime: string;
SongPlayTime: number;
IsDigitalAlbum: number;
Copyright: number;
Hot: number;
Playable: number;
SongH5Url: string;
MvId: number;
MvVid: string;
KSongId: number;
KSongMid: string;
SingerId: number;
SingerMid: string;
SingerName: string;
SingerTitle: string;
SingerPic: string;
SingerPic150X150: string;
SingerPic300X300: string;
SingerPic500X500: string;
OtherSingerList: {
  SingerId: number;
  SingerMid: string;
  SingerName: string;
  SingerTitle: string;
  SingerPic: string;
}[];
AlbumId: number;
AlbumMid: string;
AlbumName: string;
AlbumTitle: string;
AlbumPic: string;
AlbumPic150X150: string;
AlbumPic300X300: string;
```



```
AlbumPic500X500: string;
SongPlayUrl: string;
SongPlayUrlStandard: string;
SongPlayUrlHq: string;
SongPlayUrlSq: string;
SongSize: number;
SongSizeStandard: number;
SongSizeHq: number;
SongSizeSq: number;
TryPlayable: number;
TryBegin: number;
TryEnd: number;
Try30SUrl: string;
TryFileSize: number;
UnplayableCode: number;
UnplayableMsg: string;
UserOwnRule: number;
SongVersion: number;
WeightPlayCnt: number;
FNote: number;
EditAllow: number;
SongPlayUrlDolby: string;
SongSizeDolby: number;
Action: {
  Switch: number;
  Switch2: number;
  PlayAccess: {
    FQ: number;
    STANDARD: number;
    HQ: number;
    SQ: number;
    XQ: number;
    DOLBY: number;
    FLYRA: number;
    ATMOS: number;
  };
  DownloadAccess: {
    FQ: number;
    STANDARD: number;
    HQ: number;
```

```
SQ: number;
XQ: number;
DOLBY: number;
FLYRA: number;
ATMOS: number;
};
CacheAccess: {
  FQ: number;
  STANDARD: number;
  HQ: number;
  SQ: number;
  XQ: number;
  DOLBY: number;
  FLYRA: number;
  ATMOS: number;
};
};
SongEkeyStandard: string;
SongEkeyHq: string;
SongEkeySq: string;
SongEkeyDolby: string;
Extra: Record<string, unknown>;
SongEkeyXq: string;
SongPlayUrlXq: string;
SongSizeXq: number;
SongTypeDolby: string;
Bpm: number;
LimitFree: number;
ShouldPay: number;
PayPrice: number;
VolumeGain: number;
VolumePeak: number;
VolumeLra: number;
PayStatus: number;
SongPlayUrlPq: string;
SongEkeyPq: string;
SongSizePq: number;
SongTypePq: string;
}
```

DescribeLyricResp

```
interface DescribeLyricResp {  
    SongId: number;  
    SongLyric: string;  
    SongTranslateEncLyric: string;  
    SongRomeEncLyric: string;  
    SongQrcEncLyric: string;  
}
```

QQMusicNewTrackSongItem

```
interface QQMusicNewTrackSongItem {  
    SongId: number;  
    SongMid: string;  
    SongName: string;  
    SongTitle: string;  
    IsOnly: number;  
    Vip: number;  
    Language: string;  
    Genre: string;  
    PublicTime: string;  
    SongPlayTime: number;  
    IsDigitalAlbum: number;  
    Copyright: number;  
    Hot: number;  
    Playable: number;  
    SongH5Url: string;  
    MvId: number;  
    MvVid: string;  
    KSongId: number;  
    KSongMid: string;  
    SingerId: number;  
    SingerMid: string;  
    SingerName: string;  
    SingerTitle: string;  
    SingerPic: string;  
    SingerPic150X150: string;  
    SingerPic300X300: string;
```

```
SingerPic500X500: string;
OtherSingerList: {
  SingerId: number;
  SingerMid: string;
  SingerName: string;
  SingerTitle: string;
  SingerPic: string;
}[];
AlbumId: number;
AlbumMid: string;
AlbumName: string;
AlbumTitle: string;
AlbumPic: string;
AlbumPic150X150: string;
AlbumPic300X300: string;
AlbumPic500X500: string;
SongPlayUrl: string;
SongPlayUrlStandard: string;
SongPlayUrlHq: string;
SongPlayUrlSq: string;
SongSize: number;
SongSizeStandard: number;
SongSizeHq: number;
SongSizeSq: number;
TryPlayable: number;
TryBegin: number;
TryEnd: number;
Try30SUrl: string;
TryFileSize: number;
UnplayableCode: number;
UnplayableMsg: string;
UserOwnRule: number;
SongVersion: number;
WeightPlayCnt: number;
FNote: number;
EditAllow: number;
SongPlayUrlDolby: string;
SongSizeDolby: number;
Action: {
  Switch: number;
```

```
Switch2: number;
PlayAccess: {
  FQ: number;
  STANDARD: number;
  HQ: number;
  SQ: number;
  XQ: number;
  DOLBY: number;
  FLYRA: number;
  ATMOS: number;
};
DownloadAccess: {
  FQ: number;
  STANDARD: number;
  HQ: number;
  SQ: number;
  XQ: number;
  DOLBY: number;
  FLYRA: number;
  ATMOS: number;
};
CacheAccess: {
  FQ: number;
  STANDARD: number;
  HQ: number;
  SQ: number;
  XQ: number;
  DOLBY: number;
  FLYRA: number;
  ATMOS: number;
};
};
SongEkeyStandard: string;
SongEkeyHq: string;
SongEkeySq: string;
SongEkeyDolby: string;
Extra: Record<string, unknown>;
SongEkeyXq: string;
SongPlayUrlXq: string;
SongSizeXq: number;
```

```
SongTypeDolby: string;
Bpm: number;
LimitFree: number;
ShouldPay: number;
PayPrice: number;
VolumeGain: number;
VolumePeak: number;
VolumeLra: number;
PayStatus: number;
SongPlayUrlPq: string;
SongEkeyPq: string;
SongSizePq: number;
SongTypePq: string;
}
```

QQMusicSyncSongEntry

```
interface QQMusicSyncSongEntry {
  UserOwnRule: number;
  SongId: number;
  SongPlayTime: number;
  SongURL: string;
  SongName: string;
  SingerName: string;
}
```

QQMusicSelfSongList

```
interface QQMusicSelfSongList {
  /** Update time of the song list */
  UpdateTime: number;
  /** Creation time of the song list */
  CreateTime: number;
  song list ID
  DissId: number;
  /** Song list name */
  DissName: string;
  /** Song list cover */
  DissPic: string;
}
```

```
Number of songs in the song list
SongNum: number;
/** Number of song list listens */
ListenNum: number;
}
```

DescribeSongListResp

```
interface DescribeSongListResp {
    DissId: number;
    Hot: number;
    DissTitle: string;
    PicUrl: string;
    TotalNum: number;
    OwnerFlag: number;
    SongList: QQMusicSongInDiss[];
}
```

QQMusicSongInDiss

```
interface QQMusicSongInDiss {
    SongId: number;
    SongName: string;
    SongTitle: string;
    SongMid: string;
    AlbumId: number;
    AlbumMid: string;
    AlbumName: string;
    SongPlayTime: number;
    SingerId: number;
    SingerMid: string;
    SingerName: string;
    QQMusicFlag: number;
    UserOwnRule: number;
    OpiPlayFlag: number;
    SongType: number;
}
```

OperateSquareSongListResp

```
interface OperateSquareSongListResp {  
    Data: QQMusicSelfSongList[];  
}
```

QQMusicMVItem

```
interface QQMusicMVItem {  
    Id: number;  
    Vid: string;  
    Title: string;  
    Subtitle: string;  
    PicURL: string;  
    Duration: number;  
    PlayCnt: number;  
    CommentCnt: number;  
}
```

QQMusicMVInfo

```
interface QQMusicMVInfo {  
    /** Blu-ray streaming media size */  
    MvBlueRaySize: number;  
    /** Blu-ray streaming media url */  
    MvBlueRayURL: string;  
    /** High-definition streaming media size */  
    MvHQSize: number;  
    The url for high-definition streaming media  
    MvHQUrl: string;  
    /** Low-quality streaming media size */  
    MvLQSize: number;  
    /** Low-quality streaming media url */  
    MvLQUrl: string;  
    /** Ultra-high-definition streaming media size */  
    MvSQSize: number;  
    /** Ultra-high-definition streaming media url */  
    MvSQUrl: string;
```



```

/** File id of the Mv */
MVFileid: string;
/** id of the Mv */
MvId: number;
/** Playback duration */
MvPlayTime: number;
/** Title of Mv */
MvTitle: string;
/** vid of Mv */
MvVid: string;
/** image url of Mv */
PicUrl: string;
/** Playback permission */
Playable: number;
/** Release time */
PublicTime: string;
Singer id
SingerId: number;
/** Singer mid */
SingerMid: string;
Singer name
SingerName: string;
Collection of singers of MVs
Singers: {
    Singer id
    Id: number;
    /** Singer mid */
    Mid: string;
    /** Singer name */
    Name: string;
}[];
/** Unplayable permission block code */
UnplayableCode: number;
}

```

SearchMusicResp

```

interface SearchMusicResp {
    Current number of returns

```

```
CurNum: number;
/** Current page */
CurPage: number;
Total number of results for this search term
TotalNum: number;
/** Search term */
Keyword: string;
/** Single list */
List: QQMusicSearchMusicItem[];
/** */
DirectInfo: QQMusicDirectInfo;
}
```

QQMusicSearchSingerItem

```
interface QQMusicSearchSingerItem {
  SingerId: number;
  SingerMid: string;
  SingerName: string;
  SingerPic: string;
  AlbumNum: string;
  SongNum: string;
}
```

QQMusicPopularSingerItem

```
interface QQMusicPopularSingerItem {
  Country: string;
  SingerId: number;
  SingerMid: string;
  SingerName: string;
  SingerTranslatorName: string;
}
```

QQMusicSingerInfo

```
interface QQMusicSingerInfo {
  Area: string;
```

```
SingerId: number;
SingerMid: string;
SingerName: string;
SingerPic: string;
SingerTranslatorName: string;
SongSum: number;
SongList: {
  UserOwnRule: number;
  AlbumId: number;
  AlbumMid: string;
  AlbumName: string;
  AlbumPic: string;
  Genre: string;
  IsOnly: number;
  KSongId: number;
  KSongMid: string;
  Language: string;
  Playable: number;
  PublicTime: string;
  SingerId: number;
  SingerMid: string;
  SingerName: string;
  SingerPic: string;
  SizeTry: number;
  SongH5Url: string;
  SongId: number;
  SongMid: string;
  SongName: string;
  SongPlayTime: number;
  SongPlayUrl: string;
  SongPlayUrlHq: string;
  SongPlayUrlSq: string;
  SongPlayUrlStandard: string;
  SongSize: number;
  SongSizeHq: number;
  SongSizeSq: number;
  SongSizeStandard: number;
  TryBegin: number;
  TryEnd: number;
  IsDigitalAlbum: number;
```

```
    }[];  
  }  
}
```

DescribeRecentPlayResp

```
interface DescribeRecentPlayResp {  
  Data: {  
    All: Array<{  
      Type: number;  
      TypeName: string;  
      LastTime: number;  
      Detail: {  
        Comm: {  
          Title: string;  
          Pic: string;  
          Count: number;  
        };  
        Album: {  
          Title: string;  
          Mid: string;  
          Singer: string;  
          Count: number;  
          Pic: string;  
          Pic150X150: string;  
          Pic300X300: string;  
          Pic500X500: string;  
          Id: number;  
          LastTime: number;  
        };  
        Playlist: {  
          Title: string;  
          Id: number;  
          Creator: string;  
          Count: number;  
          Pic: string;  
          LastTime: number;  
        };  
      };  
    };  
  };  
}>;
```

```
Song: Array<{
  Title: string;
  Mid: string;
  Singer: string[];
  AlbumTitle: string;
  IsOnly: number;
  Size320Mp3: number;
  SizeFlac: number;
  ListenCount: number;
  SubTitle: string;
  Id: number;
  Vip: number;
  LastTime: number;
}>;
Album: Array<{
  Title: string;
  Mid: string;
  Singer: string;
  Count: number;
  Pic: string;
  Pic150X150: string;
  Pic300X300: string;
  Pic500X500: string;
  Id: number;
  LastTime: number;
}>;
Playlist: Array<{
  Title: string;
  Id: number;
  Creator: string;
  Count: number;
  Pic: string;
  LastTime: number;
}>;
};
UpdateTime: number;
}
```

DescribeHomepageSongListResp

```
interface DescribeHomepageSongListResp {
  ShelfArr: {
    Title: string;
    CardArr: {
      Type: number;
      Id: string;
      Title: string;
      Subtitle: string;
      Cover: string;
      Cnt: number;
      SubId: string;
      FavCnt: number;
    }[];
  }[];
  RcItemArr: {
    DirId: number;
    DirName: string;
    DissId: number;
    AlbumPicUrl: string;
  }[];
}
```

QQMusicIndividualRadioSongItem

```
interface QQMusicIndividualRadioSongItem {
  SongId: number;
  SongMid: string;
  SongName: string;
  SongTitle: string;
  Isonly: number;
  Vip: number;
  Language: string;
  Genre: string;
  PublicTime: string;
  SongPlayTime: number;
  IsdigitalAlbum: number;
  Copyright: number;
  Hot: number;
  Playable: number;
```

```
SongH5Url: string;
MvId: number;
MvVid: string;
KSongId: number;
KSongMid: string;
SingerId: number;
SingerMid: string;
SingerName: string;
SingerTitle: string;
SingerPic: string;
SingerPic150X150: string;
SingerPic300X300: string;
SingerPic500X500: string;
OtherSingerList: {
  SingerId: number;
  SingerMid: string;
  SingerName: string;
  SingerTitle: string;
  SingerPic: string;
}[];
AlbumId: number;
AlbumMid: string;
AlbumName: string;
AlbumTitle: string;
AlbumPic: string;
AlbumPic150X150: string;
AlbumPic300X300: string;
AlbumPic500X500: string;
SongPlayUrl: string;
SongPlayUrlStandard: string;
SongPlayUrlHq: string;
SongPlayUrlSq: string;
SongSize: number;
SongSizeStandard: number;
SongSizeHq: number;
SongSizeSq: number;
TryPlayable: number;
TryBegin: number;
TryEnd: number;
Try30SUrl: string;
```

```
TryFileSize: number;
UnplayableCode: number;
UnplayableMsg: string;
UserOwnRule: number;
SongVersion: number;
WeightPlayCnt: number;
Fnote: number;
EditAllow: number;
SongPlayUrlDolby: string;
SongSizeDolby: number;
Action: {
  Switch: number;
  Switch2: number;
};
SongEkeyStandard: string;
SongEkeyHq: string;
SongEkeySq: string;
SongEkeyDolby: string;
Extra: Record<string, unknown>;
}
```

QQMusicDailySongItem

```
interface QQMusicDailySongItem {
  SongId: number;
  SongMid: string;
  SongName: string;
  SongTitle: string;
  Isonly: number;
  Vip: number;
  Language: string;
  Genre: string;
  PublicTime: string;
  SongPlayTime: number;
  IsdigitalAlbum: number;
  Copyright: number;
  Hot: number;
  Playable: number;
  SongH5Url: string;
}
```



```
MvId: number;
MvVid: string;
KSongId: number;
KSongMid: string;
SingerId: number;
SingerMid: string;
SingerName: string;
SingerTitle: string;
SingerPic: string;
SingerPic150X150: string;
SingerPic300X300: string;
SingerPic500X500: string;
OtherSingerList: {
  SingerId: number;
  SingerMid: string;
  SingerName: string;
  SingerTitle: string;
  SingerPic: string;
}[];
AlbumId: number;
AlbumMid: string;
AlbumName: string;
AlbumTitle: string;
AlbumPic: string;
AlbumPic150X150: string;
AlbumPic300X300: string;
AlbumPic500X500: string;
SongPlayUrl: string;
SongPlayUrlStandard: string;
SongPlayUrlHq: string;
SongPlayUrlSq: string;
SongSize: number;
SongSizeStandard: number;
SongSizeHq: number;
SongSizeSq: number;
TryPlayable: number;
TryBegin: number;
TryEnd: number;
Try30SUrl: string;
TryFileSize: number;
```

```
UnplayableCode: number;
UnplayableMsg: string;
UserOwnRule: number;
SongVersion: number;
WeightPlayCnt: number;
Fnote: number;
EditAllow: number;
SongPlayUrlDolby: string;
SongSizeDolby: number;
Action: {
  Switch: number;
  Switch2: number;
  PlayAccess: {
    FQ: number;
    STANDARD: number;
    HQ: number;
    SQ: number;
    XQ: number;
    DOLBY: number;
    FLYRA: number;
    ATMOS: number;
  };
  DownloadAccess: {
    FQ: number;
    STANDARD: number;
    HQ: number;
    SQ: number;
    XQ: number;
    DOLBY: number;
    FLYRA: number;
    ATMOS: number;
  };
  CacheAccess: {
    FQ: number;
    STANDARD: number;
    HQ: number;
    SQ: number;
    XQ: number;
    DOLBY: number;
    FLYRA: number;
```

```
    ATMOS: number;
};
};
SongEkeyStandard: string;
SongEkeyHq: string;
SongEkeySq: string;
SongEkeyDolby: string;
Extra: {
    Abt: string;
    Cmd: string;
    Tf: string;
    Trace: string;
    UniqueTraceId: string;
};
SongEkeyXq: string;
SongPlayUrlXq: string;
SongSizeXq: number;
SongTypeDolby: string;
Bpm: number;
LimitFree: number;
ShouldPay: number;
PayPrice: number;
VolumeGain: number;
VolumePeak: number;
VolumeLra: number;
PayStatus: number;
SongPlayUrlPq: string;
SongEkeyPq: string;
SongSizePq: number;
SongTypePq: string;
}
```

QQMusicTopListGroupItem

```
interface QQMusicTopListGroupItem {
    GroupId: number;
    GroupName: string;
    GroupTopList: {
        ListenNum: number;
```

```
ShowTime: string;
SongList: {
  Rank: number;
  SingerId: number;
  SingerMid: string;
  SingerName: string;
  SongId: number;
  SongMid: string;
  SongName: string;
}[];
TopBannerPic: string;
TopHeaderPic: string;
TopId: number;
TopName: string;
TopType: number;
TotalNum: number;
}[];
GroupType: number;
}
```

DescribeTopListInfoResp

```
interface DescribeTopListInfoResp {
  ListenNum: number;
  SongList: {
    UserOwnRule: number;
    AlbumId: number;
    AlbumMid: string;
    AlbumName: string;
    AlbumPic: string;
    Genre: string;
    Hot: number;
    IsOnly: number;
    KSongId: number;
    KSongMid: string;
    Language: string;
    Playable: number;
    PublicTime: string;
    SingerId: number;
  }[];
}
```

```
SingerMid: string;
SingerName: string;
SingerPic: string;
SongH5Url: string;
SongId: number;
SongMid: string;
SongName: string;
SongPlayTime: number;
SongPlayUrl: string;
SongPlayUrlHq: string;
SongPlayUrlSq: string;
SongPlayUrlStandard: string;
SongSize: number;
SongSizeHq: number;
SongSizeSq: number;
SongSizeStandard: number;
TopRankIncrease: number;
TryBegin: number;
TryEnd: number;
IsDigitalAlbum: number;
}[];
TopBannerPic: string;
TopDesc: string;
TopHeaderPic: string;
TopId: number;
TopName: string;
TopType: number;
}
```

QQMusicFreeSongGroupItem

```
interface QQMusicFreeSongGroupItem {
  SongList: {
    SongId: number;
    SongMid: string;
    SongTitle: string;
    Vip: number;
    SingerId: number;
    SingerMid: string;
  }
}
```

```
SingerName: string;
TryPlayable: number;
SongPlayTime: number;
Playable: number;
UnplayableCode: number;
AlbumId: number;
AlbumMid: string;
AlbumTitle: string;
AlbumPic: string;
AlbumPic300X300: string;
AlbumPic500X500: string;
SongToken: string;
}[];
Number: number;
Name: string;
Pic: string;
BannerPic: string;
}
```

QQMusicFreeSongInfo

```
interface QQMusicFreeSongInfo {
  SongId: number;
  SongMid: string;
  SongName: string;
  SongTitle: string;
  IsOnly: number;
  Vip: number;
  Language: string;
  Genre: string;
  PublicTime: string;
  SongPlayTime: number;
  IsDigitalAlbum: number;
  Copyright: number;
  Hot: number;
  Playable: number;
  SongH5Url: string;
  MvId: number;
  MvVid: string;
}
```

```
KSongId: number;
KSongMid: string;
SingerId: number;
SingerMid: string;
SingerName: string;
SingerTitle: string;
SingerPic: string;
SingerPic150X150: string;
SingerPic300X300: string;
SingerPic500X500: string;
OtherSingerList: {
  SingerId: number;
  SingerMid: string;
  SingerName: string;
  SingerTitle: string;
  SingerPic: string;
}[];
AlbumId: number;
AlbumMid: string;
AlbumName: string;
AlbumTitle: string;
AlbumPic: string;
AlbumPic150X150: string;
AlbumPic300X300: string;
AlbumPic500X500: string;
SongPlayUrl: string;
SongPlayUrlStandard: string;
SongPlayUrlHq: string;
SongPlayUrlSq: string;
SongSize: number;
SongSizeStandard: number;
SongSizeHq: number;
SongSizeSq: number;
TryPlayable: number;
TryBegin: number;
TryEnd: number;
Try30SUrl: string;
TryFileSize: number;
UnplayableCode: number;
UnplayableMsg: string;
```

```
UserOwnRule: number;
SongVersion: number;
WeightPlayCnt: number;
FNote: number;
EditAllow: number;
SongPlayUrlDolby: string;
SongSizeDolby: number;
Action: {
  Switch: number;
  Switch2: number;
  PlayAccess: {
    FQ: number;
    STANDARD: number;
    HQ: number;
    SQ: number;
    XQ: number;
    DOLBY: number;
    FLYRA: number;
    ATMOS: number;
  };
  DownloadAccess: {
    FQ: number;
    STANDARD: number;
    HQ: number;
    SQ: number;
    XQ: number;
    DOLBY: number;
    FLYRA: number;
    ATMOS: number;
  };
  CacheAccess: {
    FQ: number;
    STANDARD: number;
    HQ: number;
    SQ: number;
    XQ: number;
    DOLBY: number;
    FLYRA: number;
    ATMOS: number;
  };
};
```



```
};  
SongEkeyStandard: string;  
SongEkeyHq: string;  
SongEkeySq: string;  
SongEkeyDolby: string;  
Extra: Record<string, unknown>;  
SongEkeyXq: string;  
SongPlayUrlXq: string;  
SongSizeXq: number;  
SongTypeDolby: string;  
Bpm: number;  
}
```

QQMusicQualityType

```
const enum QQMusicQualityType {  
    Smooth  
    FQ = 1,  
    /** Standard */  
    STANDARD = 2,  
    /** High-quality */  
    HQ = 4,  
    /** Ultra-quality */  
    SQ = 8,  
    /** Hi-Res */  
    XQ = 16,  
    /** Dolby */  
    DOLBY = 32,  
    /** Master tape */  
    FLYRA = 64,  
    /** Premium Panorama Sound */  
    ATMOS = 128,  
}
```

QQMusicPlayListType

```
const enum QQMusicPlayListType {  
    /** Song list */  
    SongList = 5,  
}
```

```
/** Song */  
Song = 6,  
/** Radio station */  
Radio = 7,  
/** Ranked list */  
Rank = 8,  
/** Album */  
Album = 9,  
Singer  
Singer = 10,  
Recommended  
Recommend = 12,  
/** No-login zone */  
NoLoginArea = 14,  
}
```

Underlying SDK Capabilities

Last updated: 2025-04-28 11:04:32

Application Development SDK

H5 SDK relies on Application Development Mini Program SDK at the underlying level. The following code can be used to obtain an instance of the Application Development SDK. For more invocation capabilities, please see the [Application Development Mini Program SDK](#) document.

API definition

```
sdk.appDevSdk
```

WeChat JS SDK

The WeChat JSSDK instance can be obtained through the following code. For specific usage, please refer to the [Mini Program web-view document](#). Before use, you need to call **Initialize JS SDK**.

API definition

```
sdk.wx
```

Initializing the JS SDK

- API definition

```
sdk.wxSdkReady() => Promise
```

- return value

Return a cached Promise that resolves after successful initialization. If the initialization is ongoing or has already been successful, the same Promise will be returned after multiple calls. If the initialization fails, the cached Promise will be released after rejection, and subsequent calls will reinitialize.

- Sample code

```
sdk.wxSdkReady().then(() => wx.miniProgram.navigateBack());
```

Developing Chinese Domestic Brand Mini Programs and Apps

Last updated: 2025-04-28 11:04:46

Overview

This document guides device vendors to develop Chinese domestic brand mini programs and apps.

Access Guide

One-Stop Development

Based on the **application-side API of Tencent Cloud IoT Explorer**, develop mini programs and apps of Chinese domestic brands. You don't need to develop your own business backend.

1. Refer to the [application development](#) guide, create an application on the IoT Explorer, and obtain the App Key and App Secret.
2. Upstream data from devices. Mini programs and apps can refer to [persistent connection communication](#) for subscription receipt, or obtain through the [data query](#) API.
3. Mini programs and apps can send control commands to devices through the [device control](#) API.

Note:

If you need to develop audio/video call features, please first communicate with the business team and purchase video activation codes. At that time, there will be a dedicated group to support your integration and usage.

Cloud Integration Development

Based on **TencentCloud API** of Tencent Cloud IoT Explorer, to develop mini programs and apps of Chinese domestic brands, you need to develop your own business backend.

1. Mini program/App obtains device upstream data
 - 1.1 The reported data from the device is forwarded to the customer business backend through the [rule engine](#).
 - 1.2 The customer business backend receives the reported data from the device, which can be saved for querying by the mini program/App.
 - 1.3 The mini program/App sends a request to the customer business backend through the interface provided by the customer business backend to obtain the latest data or historical data of the device.

2. Mini program/App controls device

- 2.1 The mini program/App sends the control command of the device to the customer business backend through the interface provided by the customer business backend.
- 2.2 The customer business backend sends control commands to the device through the TencentCloud API for [remotely controlling devices](#).

Note:

If you need to develop audio/video call features, please first communicate with the business team and purchase video activation codes. At that time, there will be a dedicated group to support your integration and usage.

Application Side API

Application Side API Introduction

Last updated: 2025-04-28 15:12:20

Overview

The application-side API is a cloud service provided by the development platform to meet the smart home scenario. It is developed for users to develop their own brand's mini programs or apps. Users do not need to implement basic capabilities such as user management, device management, device scheduling, and family management. They can quickly complete the development of the mobile application end by calling the application-side API.

Calling Method

1. The API URL used before login is:

Note:

- Chinese mainland platform:
`https://iot.cloud.tencent.com/api/exploreropen/appapi`
- Overseas platform:
`https://oversea.iotcloud.tencentiotcloud.com/api/exploreropen/appapi`

Among them, the common parameters are: Action, RequestId, AppKey, Signature, Timestamp, Nonce.

- Action used to identify the request method name.
- RequestId used to identify a unique request. It is recommended to use uuid as the parameter value. It is advisable to provide this parameter value when locating issues.
- AppKey is the key of the application.
- Signature is the signature of the current request. For specific calculation, see the example below in this document.
- Timestamp is the UNIX second-level timestamp of the current request.
- Nonce is a random positive integer, used together with the timestamp to prevent API replay attacks.

2. The API URL used after log-in is:

Note:

- Chinese mainland platform:
<https://iot.cloud.tencent.com/api/exploreropen/tokenapi>
- Overseas platform:
<https://oversea.iotcloud.tencentiotcloud.com/api/exploreropen/tokenapi>

Among them, the common parameters are: Action, RequestId, AccessToken.

- Action used to identify the request method name.
- RequestId used to identify a unique request. It is recommended to use uuid as the parameter value. It is advisable to use this parameter value when locating issues.
- AccessToken used to identify a logged-in user.

Note:

Before calling the application side API interface, you need to call the login-related API (application side API/user management) first. After successful log-in, obtain `AccessToken`.

`AccessToken` is used to identify a user. After a user logs in successfully, carry `AccessToken` to call APIs of other applications.

URL suffix for calling before login is `.../appapi` URL suffix for calling after login is `.../tokenapi`

Signature Algorithm

Retrieve the Application AppKey and AppSecret

If the user does not use Tencent's official "Tencent Lianlian" mini program, the user can also develop their own brand mini program through the platform's open capabilities. When creating an application, the platform will generate a security credential corresponding to the mini program for the user. The security credential includes AppKey and AppSecret. AppKey is used to identify the API caller identity, and AppSecret is the key used to encrypt signature strings and for server-side verification of signature strings.

Note:

Users should strictly keep their AppSecrets to avoid leakage.

The specific retrieval steps are as follows:

- 1.1 Log in to the [IoT Explorer console](#) and enter the development center.
- 1.2 Select **Application Development > Mini Program Development** in the left menu, create a mini program. For specific creation steps, see [Application Development](#).

1.3 Upon success in creating a mini program, you can obtain the system-generated AppKey and AppSecret.

Generate Signature String

After obtaining the security credentials AppKey and AppSecret, you can generate a signature string. Below is a detailed process of generating a signature string.

Assuming the user's AppKey and AppSecret are:

- AppKey: ahPxdK****TGrejd
- AppSecret: NcbHqk****TCGbKnQH

Note:

This article is only an example. Please continue subsequent operations according to your actual AppKey and AppSecret .

Take the request to register an account via mobile phone number AppCreateCellphoneUser as an example. When a user makes this API call, its request parameters **may be** as follows:

Parameter Name	Type	Description	Reference Value
RequestId	String	Common parameters, unique request ID, can be generated manually, uuid is recommended for use. Locating issues, it is required to provide the RequestId for the request.	8b8d499bbba1ac28b6da21b4
Action	String	Common parameters, API method name of the call.	AppCreateCellphoneUser
AppKey	String	Common parameters, application AppKey, used to identify the corresponding app.	ahPxdK****TGrejd
Signature	String	Common parameters, signature of the request.	Szxai9Qs7****OXahbFbseZ+uE=
Timestamp	Int64	Common parameters, current UNIX timestamp (in seconds).	1546315200
Nonce	Int	Common parameters, a random positive integer, used together with the timestamp for API anti-replay.	71087795

CountryCode	String	Country Code.	86
PhoneNumber	String	Phone Number.	13900000000
Password	String	Password.	password
VerificationCode	String	SMS Captcha.	123456

Note:

- The common request parameters in the request parameter include: RequestId, Action, AppKey, Timestamp, Nonce, Signature.
- AppCreateCellphoneUser API-specific parameters: CountryCode, PhoneNumber, Password, VerificationCode.

The parameter Signature (signature string) is generated by the above parameters together. Specific steps are as follows:

1. Sort the args parameter.

Sort all request parameters in lexicographical ascending order of the parameter name. Lexicographical ascending order is intuitively like sorting words in a dictionary. Sort according to the ascending order in the alphabet or number table. That is, first consider the first “letter” . In the same case, consider the second “letter” , and so on. You can use relevant sorting functions in programming languages to achieve this feature. For example, the ksort function in PHP. The sorting results of the above example parameters are as follows:

```
{  
    Action=AppCreateCellphoneUser,  
    AppKey=ahPxdK****TGrejd,  
    CountryCode=86,  
    Nonce=71087795,  
    Password=My!P@ssword,  
    PhoneNumber=13900000000,  
    RequestId=8b8d499bbba1ac28b6da21b4,  
    Timestamp=1546315200,  
    VerificationCode=123456  
}
```

When using other programming languages for development, you can sort the parameters in the above example. Results consistent with the obtained ones are acceptable.

2. Concatenate the request string.

Format the sorted request parameters from the previous step into the form of "parameter name" = "parameter value". For example, for the action parameter, whose parameter name is "Action" and parameter value is "AppCreateCellphoneUser", it will be formatted as Action=AppCreateCellphoneUser after formatting.

- The parameter value is the original value, not the URL-encoded value.
- If the "key" in the input parameter contains an underscore, it needs to be converted to ".". No additional operation is required for the "value".

Concatenate each formatted parameter with "&". The final request string generated is:

```
Action=AppCreateCellphoneUser&AppKey=ahPxdK***TGrejd&CountryCode=86&Nonce=71087795&Password=My!P@ssword&PhoneNumber=13900000000&RequestId=8b8d499bbba1ac28b6da21b4&Timestamp=1546315200&VerificationCode=123456
```

3. Generate the signature string.

Use the HMAC-SHA1 algorithm to sign the signature source string obtained in the previous step. Then encode the generated signature string using Base64 to obtain the final signature string.

The specific code is as follows, taking PHP as an example:

```
$secretKey = 'NcbHqk***TCGbKnQH';
$srcStr =
'Action=AppCreateCellphoneUser&AppKey=ahPxdK***TGrejd&CountryCode=86&Nonce=71087795&Password=My!P@ssword&PhoneNumber=13900000000&RequestId=8b8d499bbba1ac28b6da21b4&Timestamp=1546315200&VerificationCode=123456'
;
$signStr = base64_encode(hash_hmac('sha1', $srcStr, $secretKey, true));
echo $signStr
```

The final obtained signature string is:

```
Szxai9Qs7O3lBoOXahbFbseZ+uE=
```

When using other programming languages for development, you can use the original text in the above example for signature verification. The obtained signature string should be consistent with that in the example.

Common Issues

For parameters that are strings and empty, they should be omitted from the signature string. It is not necessary to include them in the signature.

DevTools

Use the application-side API [DevTools](#) to quickly debug the application-side API.

User Management

WeChat ID Registration and Login

Last updated: 2025-04-28 15:12:53

1. API Description

API request domain name: `iot.cloud.tencent.com/api/exploreropen/appapi`.

This interface (AppGetTokenByWeiXin) is used for WeChat user registration and login to obtain the user access Token of the development platform. When called for the first time, it automatically registers a corresponding account for the WeChat ID.

2. Input Parameters

Name	Type	Required	Description
Action	String	Yes	Common parameter, this interface takes values: AppGetTokenByWeiXin.
RequestId	String	Yes	Common parameter, unique request ID, can be generated manually, uuid is recommended for use. Locating issues, need to provide the RequestId for the request.
AppKey	String	Yes	Common parameter, application AppKey, used to identify the corresponding mini program or app.
Signature	String	Yes	Common parameter, signature request, which needs to be generated by the user on one's own and is used for verifying the legitimacy of the request.
Timestamp	Int64	Yes	Common parameter, request's UNIX timestamp (in seconds).
Nonce	Int	Yes	Common parameter, a random positive integer, combined with Timestamp to prevent replay attacks.
WxOpenID	String	Yes	OpenID or UnionID of a WeChat user.
NickName	String	Yes	Nickname.

me			
Avatar	String	Yes	Avatar.

3. Output Parameters

Name	Type	Description
RequestId	String	Common parameters, unique request ID, consistent with the input parameters.
ExpireAt	Int64	Deadline, UNIX second-level timestamp.
Token	String	AccessToken returned by the development platform. Use this Token to make api requests after logging in.

4. Samples

Input example

```
POST https://iot.cloud.tencent.com/api/exploreropen/appapi HTTP/1.1
content-type: application/json
{
  "WxOpenID": "0123456789",
  "NickName": "tests",
  "Avatar": "http://example.com/path/to/avatar.jpg",
  "Signature": "IY0yHX7zWCZKZ97xDonQoy16W4U=",
  "Timestamp": 1552621825,
  "Nonce": 2,
  "Action": "AppGetTokenByWeiXin",
  "AppKey": "ahPxdK*****TGrejd",
  "RequestId": "rest-client"
}
```

Output example: successful

```
{
  "Response": {
    "Data": {
      "ExpireAt": 1556076201,
      "Token": "d2*****8514"
    },

```

```
    "RequestId": "rest-client"
  }
}
```

Output example: failure

```
{
  "Response": {
    "Error": {
      "Code": "InvalidParameterValue.ErrorUserNotExists",
      "Message": "account does not exist"
    },
    "RequestId": "rest-client"
  }
}
```

5. Error Code

Error Code	Description
InternalServerError	Internal error.
ErrorRequiredParamNotFound	Required parameter is missing.
InvalidAction	Illegal Action.
InvalidParameterValue	Parameters are abnormal.
InvalidParameterValue.InvalidJSON	The request format is not JSON.
InvalidParameterValue.NickNameLengthInvalid	Nickname length is illegal.

User Logging Out

Last updated: 2025-12-19 09:14:42

1. API Description

api request domain name: `iot.cloud.tencent.com/api/exploreropen/tokenapi`.

This API (AppLogoutUser) is used for user logout.

2. Input Parameters

Name	Type	Required	Description
Action	String	Yes	Common parameter, this interface takes values: AppLogoutUser.
AccessToken	String	Yes	Common parameter, AccessToken is used for user authentication of a logged-in user.
RequestId	String	Yes	Common parameter, unique request ID, can be generated manually, uuid is recommended for use. Locating issues, need to provide the RequestId for the request.

3. Output Parameters

Name	Type	Description
RequestId	String	Common parameter, unique request ID, same as the input parameter. Locating issues, need to provide the RequestId for the request.
Data	String	Response result.

4. Samples

Input example

```
POST https://iot.cloud.tencent.com/api/exploreropen/tokenapi HTTP/1.1
content-type: application/json
{
  "AccessToken": "bf*****d098",
  "Action": "AppLogoutUser",
```

```
"RequestId": "rest-client"
}
```

Output sample: successful

```
{
  "Response": {
    "Data": "OK",
    "RequestId": "f92406b3-5a9a-4fe8-bc43-45e3d794bb68"
  }
}
```

Output example: failure

```
{
  "Response": {
    "Error": {
      "Code": "InvalidParameterValue.InvalidAccessToken",
      "Message": "Invalid Token"
    },
    "RequestId": ""
  }
}
```

5. Error Code

Error Code	Description
InternalServerError	An internal error occurred.
ErrorRequiredParamNotFound	Required parameter is missing.
InvalidAction	Illegal Action.
InvalidParameterValue	Invalid parameter.
InvalidParameterValue.InvalidJSON	The request format is not JSON.
InvalidParameterValue.InvalidAccessToken	Invalid Token.

Distribution Network Management

Generating Wi-Fi Device Configuration Token

Last updated: 2025-04-30 15:17:48

1. API Description

API request domain name: `iot.cloud.tencent.com/api/exploreropen/tokenapi`

This interface (AppCreateDeviceBindToken) is used to generate a random Token for the Wi-Fi configuration task.

2. Input Parameter

Name	Type	Required	Description
Action	String	Yes	Common parameter, this interface takes values: AppCreateDeviceBindToken.
RequestId	String	Yes	Common parameter, unique request ID, can be generated manually, uuid is recommended. Locating issues, need to provide the RequestId for the request.
AccessToken	String	Yes	Common parameter, AccessToken is used for user authentication of a logged-in user.

3. Output Parameters

Name	Type	Description
RequestId	String	RequestId.
Token	String	Generated distribution network Token.

4. Examples

Example 1 Generate a Wi-Fi Device Networking Token

Input example

```
POST https://iot.cloud.tencent.com/api/exploreropen/tokenapi HTTP/1.1
content-type: application/json

{
  "Action": "AppCreateDeviceBindToken",
  "AccessToken": "bf*****d098",
  "RequestId": "f92406b3-5a9a-4fe8-bc43-45e3d794bb68"
}
```

Output sample: successful

```
{
  "Response": {
    "Data": {
      "Token": "d7*****1b3b",
      "RequestId": "f92406b3-5a9a-4fe8-bc43-45e3d794bb68"
    }
  }
}
```

Output sample: fail

```
{
  "Response": {
    "Error": {
      "Code": "InvalidParameterValue.InvalidAccessToken",
      "Message": "Invalid Token"
    },
    "RequestId": "f92406b3-5a9a-4fe8-bc43-45e3d794bb68"
  }
}
```

5. Error Code

Error Code	Description
InvalidParameter	Invalid parameter.
InvalidParameterValue	Invalid parameter value.

InvalidParameterValue.TokenInfoSaveError	Token saving failed.
--	----------------------

Query Distribution Network Token Status

Last updated: 2025-04-30 15:18:03

1. API Description

api request domain name: `iot.cloud.tencent.com/api/exploreropen/tokenapi`

This interface (AppGetDeviceBindTokenState) is used for querying the current status of the network configuration Token.

2. Input Parameter

Name	Type	Required	Description
Action	String	Yes	Common parameter, this interface takes values: AppGetDeviceBindTokenState.
RequestId	String	Yes	Common parameter, unique request ID, can be generated manually, uuid is recommended for use. Locating issues, need to provide the RequestId for the request.
AccessToken	String	Yes	Common parameter, the Access Token obtained upon successful login through WeChat ID, mobile phone, or email account.
Token	String	Yes	The network configuration Token generated by the AppCreateDeviceBindToken API.

3. Output Parameter

Name	Type	Description
RequestId	String	RequestId.
State	Uint	Token status. - Initial production. 2: Usable status.

4. Example

Sample 1 Query the Status of the Distribution Network Token

Input Example

```
POST https://iot.cloud.tencent.com/api/exploreropen/tokenapi HTTP/1.1
content-type: application/json
{
  "Action": "AppCreateDeviceBindToken",
  "AccessToken": "bf*****d098",
  "RequestId": "f92406b3-5a9a-4fe8-bc43-45e3d794bb68",
  "Token": "d7a1*****a861b3b"
}
```

Output example: success

```
{
  "Response": {
    "Data": {
      "State": 1,
      "RequestId": "f92406b3-5a9a-4fe8-bc43-45e3d794bb68"
    }
  }
}
```

Output example: failure

```
{
  "Response": {
    "Error": {
      "Code": "InvalidParameterValue.TokenNotExist",
      "Message": "Token does not exist"
    },
    "RequestId": "f92406b3-5a9a-4fe8-bc43-45e3d794bb68"
  }
}
```

5. Error Code

Error Code	Description
------------	-------------

InvalidParameter	Invalid parameter.
InvalidParameterValue	Invalid parameter value.
InvalidParameterValue.TokenNotExist	Token does not exist.
InvalidParameterValue.TokenIsExpire	Token has expired.
InvalidParameterValue.ReadTokenInfoError	Token read error.

User Binding Wi-Fi Device

Last updated: 2025-04-30 15:18:21

1. API Description

api request domain name: `iot.cloud.tencent.com/api/exploreropen/tokenapi`

This interface (AppTokenBindDeviceFamily) is used for user binding of Wi-Fi devices in mini programs or apps.

2. Input Parameter

Name	Type	Required	Description
Action	String	Yes	Common parameter, this interface takes values: AppTokenBindDeviceFamily.
RequestId	String	Yes	Common parameter, unique request ID, can be generated manually, recommended to use uuid. Locating issues, need to provide the RequestId for the request.
AccessToken	String	Yes	Common parameter, the Access Token obtained after users through WeChat ID, mobile phone or email account login successfully.
Token	String	Yes	Distribution network Token.
ProductId	String	Yes	Product ID.
DeviceName	String	Yes	Device name.
FamilyId	String	Yes	Family ID.
RoomId	String	Yes	Room ID.

3. Output Parameter

Name	Type	Description
RequestId	String	Common parameter, unique request ID, can be generated manually, recommended to use uuid.

Locating issues, need to provide the RequestId for the request.

4. Example

User Binding Wi-Fi Device

Input Example

```
POST https://iot.cloud.tencent.com/api/exploreropen/tokenapi HTTP/1.1
content-type: application/json
{
  "Action": "AppTokenBindDeviceFamily",
  "AccessToken": "tesetoken",
  "Token": "bf*****d098",
  "ProductId": "*****",
  "DeviceName": "*****",
  "FamilyId": "f*****cc1",
  "RoomId": "0"
  "RequestId": "f92406b3-5a9a-4fe8-bc43-45e3d794bb68"
}
```

Output example: successful

```
{
  "Response": {
    "Data": {
      "RequestId": "f92406b3-5a9a-4fe8-bc43-45e3d794bb68"
    }
  }
}
```

Output example failure

```
{
  "Response": {
    "Error": {
      "Code": "InvalidParameterValue.InvalidAccessToken",
      "Message": "Invalid Token"
    },

```



```
"RequestId": "f92406b3-5a9a-4fe8-bc43-45e3d794bb68"
}
```

5. Error Code

Error Code	Description
InvalidParameter	Invalid parameter.
InvalidParameterValue	Invalid parameter value.
InvalidParameterValue.TokenNotExist	The Token does not exist.
InvalidParameterValue.TokenIsExpired	Token has expired.
InvalidParameterValue.ReadTokenInfoError	Token read error.
InvalidParameterValue.TokenNotBind	The Token is not bound, that is, the return status queried by the previous step API AppGetDeviceBindTokenState is in the initial status, not in the usable status.
InvalidParameterValue.FamilyDeviceCountReadError	Error reading device quantity.
UnauthorizedOperation	No operation permissions.
UnauthorizedOperation.APPNoPermissionToStudioProduct	The App has no permission to perform operations on this product.
UnauthorizedOperation.NoPermissionToFamily	No permission to operate this family.
ResourceNotFound	Resource does not exist.
ResourceNotFound.StudioFamilyNotExist	The family is not created or has been deleted.
ResourceNotFound.StudioProductNotExist	The product is not created or has been deleted.
ResourceNotFound.DeviceNotExist	The device is not created or has been deleted.

LimitExceeded	Quantity Limit.
LimitExceeded.FamilyDeviceExceedLimit	The device quantity exceeds the limit.
InternalError	Internal error.
InternalError.InternalServerError	Internal error.
InternalError.InternalServerErrorDB	Internal DB error.
UnsupportedOperation	Unsupported operation.
UnsupportedOperation.CannotBindSameFamily	The device has been bound. Do not repeat binding.

Device management

Query Weixin Authorization Invoice

Last updated: 2025-04-30 16:26:32

1. API Description

api request domain name: `iot.cloud.tencent.com/api/exploreropen/tokenapi`

This interface (APPGetWechatDeviceTicket) is used to obtain WeChat Device Ticket.

2. Input Parameter

Name	Type	Required	Description
Action	String	Yes	Common parameter, this interface takes values: APPGetWechatDeviceTicket.
RequestId	String	Yes	Common parameter, unique request ID, can be generated manually, recommended to use uuid. When locating issues, it is required to provide the RequestId for the request.
AccessToken	String	Yes	Common parameter, AccessToken is used for user authentication of a logged-in user.
ProductId	String	Yes	Product ID.
Devicename	String	Yes	Device name.
IsThirdApp	Int	Yes	Whether it is a third-party mini program: 1 = Yes.
ModelId	String	No	Template ID
MiniProgramAppId	String	No	Mini program APPID

3. Output Parameter

Name	Type	Description
------	------	-------------

RequestId	String	RequestId.
Data	WXDeviceInfo	Generated WeChat Device Ticket info.

4. Example

Example 1 Obtaining WeChat Device Ticket

<input example>

```
POST https://iot.cloud.tencent.com/api/exploreropen/tokenapi HTTP/1.1
content-type: application/json
{
  "Action": "APPGetWechatDeviceTicket",
  "AccessToken": "bf*****d098",
  "RequestId": "f92406b3-5a9a-4fe8-bc43-45e3d794bb68",
  "ProductId": "PRODUCT_ID",
  "Devicename": "DEVICE_NAME",
  "IsThirdApp": 1
}
```

Output sample: successful

```
{
  "Response": {
    "Data": {
      "DeviceId": "xxx",
      "WXIoTDeviceInfo": {
        "SN": "xxx",
        "SNTicket": "xxx",
        "ModelId": "xxx"
      },
      "RequestId": "f92406b3-5a9a-4fe8-bc43-45e3d794bb68"
    }
  }
}
```

Output example: fail

```
{
  "Response": {
    "Error": {
      "Code": "InvalidParameterValue.InvalidAccessToken",
      "Message": "Invalid Token"
    },
    "RequestId": "f92406b3-5a9a-4fe8-bc43-45e3d794bb68"
  }
}
```

5. Error Code

Error Code	Description
InvalidParameter	Invalid parameter.
InvalidParameterValue	Invalid parameter value.
InvalidParameterValue.TokenInfoSaveError	Token saving failed.

Obtaining Product Information

Last updated: 2025-04-30 15:28:00

1. API Description

api request domain name: `iot.cloud.tencent.com/api/exploreropen/tokenapi`.

This interface (AppGetProducts) obtains product information.

2. Input Parameter

Name	Type	Required	Description
AccessToken	String	Yes	common parameter, AccessToken is used for user authentication of a logged-in user.
RequestId	String	Yes	Common parameter, unique request ID, can be generated manually. It is recommended to use uuid. When locating issues, you need to provide the RequestId for the request.
Action	String	Yes	Common parameter, this interface takes values: AppGetProducts.
ProductIds	Array of String	Yes	Product ID array.
DeviceIds	Array of String	No	Device ID array.

3. Output Parameter

Name	Type	Description
Products	Array	Return product information.
RequestId	String	Common parameters, unique request ID, same as input parameters.

4. Example

Input Example

```
POST https://iot.cloud.tencent.com/api/exploreropen/tokenapi HTTP/1.1
```

```
content-type: application/json

{
  "AccessToken": "8b4a70dd16105f*****18edd4e78a3bb8ec",
  "RequestId": "27HPvi1ylv",
  "Action": "AppGetProducts",
  "ProductIds": ["FVYYYEL4ON"],
  "DeviceIds": ["QDA1PZLBNB/aaa", "QDA1PZLBNB/bbb"]
}
```

Output example: succeed

```
{
  "Response": {
    "Products": [
      {
        "ProductId": "FVYYYEL4ON",
        "Name": "c1",
        "Description": "yes",
        "State": "",
        "DataTemplate": "{ \"version\": \"1.0\", \"profile\": { \"ProductId\": \"FVYYYEL4ON\", \"CategoryId\": \"1\", \"properties\": [{ \"id\": \"mac\", \"name\": \"mac address\", \"desc\": \"wifi router mac address\", \"required\": true, \"mode\": \"r\", \"define\": { \"type\": \"string\", \"min\": \"0\", \"max\": \"30\" } }, { \"id\": \"signal\", \"name\": \"wifi signal strength\", \"desc\": \"wifi signal strength value\", \"mode\": \"r\", \"define\": { \"type\": \"int\", \"unit\": \"dbm\", \"step\": \"1\", \"min\": \"-180\", \"max\": \"180\", \"start\": \"1\" } }, { \"id\": \"ssid\", \"name\": \"SSID\", \"desc\": \"SSID of the wifi hotspot connected by the device\", \"mode\": \"r\", \"define\": { \"type\": \"string\", \"min\": \"0\", \"max\": \"64\" } }, { \"id\": \"severip\", \"name\": \"IP address integrated by the wifi router\", \"desc\": \"IP address integrated by the wifi router\", \"mode\": \"r\", \"define\": { \"type\": \"string\", \"min\": \"0\", \"max\": \"64\" } }, { \"id\": \"nmacs\", \"name\": \"other hotspot MAC address\", \"desc\": \"other hotspot MAC addresses that can be received\", \"mode\": \"r\", \"define\": { \"type\": \"string\", \"min\": \"0\", \"max\": \"64\" } } ] }, \"events\": [], \"actions\": [] }",
        "AppTemplate": "",
        "NetType": "wifi",
      }
    ]
  }
}
```

```

        "CategoryId":1,
        "ProductType":0,
        "UpdateTime":1600159203
    },
    {
        "ProductId":"QDA1PZLBNNB",
        "Name":"light5 do not delete"
        "Description":"","
        "State":"","
        "DataTemplate":{"version":"1.0","profile":
{"ProductId":"QDA1PZLBNNB","CategoryId":"141"},"properties":
[{"id":"power_switch","name":"light switch","desc":"control light
on/off","required":true,"mode":"rw","define":{"type":"bool","mapping":
{"0":"off","1":"on"}}},{id":"color","name":"light color","desc":"light
color","mode":"rw","define":{"type":"enum","mapping":
{"0":"Red","1":"Green","2":"Blue"}}},{id":"brightness","name":"light
brightness","desc":"light brightness","mode":"rw","define":
{"type":"int","unit":"%","step":"1","min":"0","max":"100","start":"1"}},
{"id":"name","name":"light position name","desc":"light position name,
such as study room, living room","mode":"rw","required":false,"define":
{"type":"string","min":"0","max":"64"}},
{"id":"Temperature","name":"temperature","desc":"","mode":"rw","define":
{"type":"float","min":"0","max":"100","start":"0","step":"1","unit":""},
"required":false},{id":"temp_max","name":"max
temperature","desc":"","mode":"rw","define":
{"type":"int","min":"0","max":"100","start":"0","step":"1","unit":"","r
equired":false},
{"id":"LBS_BS","name":"LBS_BS","desc":"","mode":"rw","define":
{"type":"string","min":"0","max":"2048"},"required":false},
{"id":"FOO","name":"FOO","desc":"","mode":"rw","define":
{"type":"string","min":"0","max":"2048"},"required":false},
{"id":"GPS_Ext","name":"GPS_Ext","desc":"","mode":"rw","define":
{"type":"string","min":"0","max":"2048"},"required":false}], "events":
[{"id":"status_report","name":"DeviceStatus","desc":"Report the device
status","type":"info","required":false,"params":
[{"id":"status","name":"running_state","desc":"Report current device
running state","define":{"type":"bool","mapping":
{"0":"normal","1":"fault"}}},
{"id":"message","name":"Message","desc":"Some extra message","define":
{"type":"string","min":"0","max":"64"}}]]},

```



```

{"id":"low_voltage","name":"LowVoltage","desc":"Alert for device voltage
is low","type":"alert","required":false,"params":
[{"id":"voltage","name":"Voltage","desc":"Current voltage","define":
{"type":"float","unit":"V","step":"1","min":"0.0","max":"24.0","start":
"1"}]}],{"id":"hardware_fault","name":"Hardware_fault","desc":"Report
hardware fault","type":"fault","required":false,"params":
[{"id":"name","name":"Name","desc":"Name like: memory,tf card,
sensors...","define":{"type":"string","min":"0","max":"64"}},
{"id":"error_code","name":"Error_Code","desc":"Error code for
fault","define":
{"type":"int","unit":"","step":"1","min":"0","max":"2000","start":"1"}]}
]],"actions":[]
    "AppTemplate":"","
    "NetType":"else",
    "CategoryId":141,
    "ProductType":0,
    "UpdateTime":1600265188
  },
  {
    "ProductId":"QDA1PZLBNNB",
    "Name":"light5 do not delete"
    "Description":"","
    "State":"","
    "DataTemplate":{"version":"1.0","profile":
{"ProductId":"QDA1PZLBNNB","CategoryId":"141"},"properties":
[{"id":"power_switch","name":"light switch","desc":"control light on and
off","required":true,"mode":"rw","define":{"type":"bool","mapping":
{"0":"off","1":"on"}}},{ "id":"color","name":"color","desc":"light
color","mode":"rw","define":{"type":"enum","mapping":
{"0":"Red","1":"Green","2":"Blue"}}},
{"id":"brightness","name":"brightness","desc":"light
brightness","mode":"rw","define":
{"type":"int","unit":"%","step":"1","min":"0","max":"100","start":"1"}},
{"id":"name","name":"light position name","desc":"light position name,
such as study room, living room,
etc.","mode":"rw","required":false,"define":
{"type":"string","min":"0","max":"64"}},
{"id":"Temperature","name":"temperature","desc":"","mode":"rw","define":
{"type":"float","min":"0","max":"100","start":"0","step":"1","unit":""},
"required":false},{ "id":"temp_max","name":"max

```

```

temperature", "desc": "", "mode": "rw", "define":
{
  "type": "int", "min": "0", "max": "100", "start": "0", "step": "1", "unit": "", "required": false,
  "id": "LBS_BS", "name": "LBS_BS", "desc": "", "mode": "rw", "define":
  {
    "type": "string", "min": "0", "max": "2048", "required": false,
    "id": "FOO", "name": "FOO", "desc": "", "mode": "rw", "define":
    {
      "type": "string", "min": "0", "max": "2048", "required": false,
      "id": "GPS_Ext", "name": "GPS_Ext", "desc": "", "mode": "rw", "define":
      {
        "type": "string", "min": "0", "max": "2048", "required": false
      }
    }, "events":
    [
      {
        "id": "status_report", "name": "DeviceStatus", "desc": "Report the device status", "type": "info", "required": false, "params":
        [
          {
            "id": "status", "name": "running_state", "desc": "Report current device running state", "define": {
              "type": "bool", "mapping": {
                "0": "normal", "1": "fault"
              }
            },
            "id": "message", "name": "Message", "desc": "Some extra message", "define": {
              "type": "string", "min": "0", "max": "64"
            }
          }
        ],
        "id": "low_voltage", "name": "LowVoltage", "desc": "Alert for device voltage is low", "type": "alert", "required": false, "params":
        [
          {
            "id": "voltage", "name": "Voltage", "desc": "Current voltage", "define": {
              "type": "float", "unit": "V", "step": "1", "min": "0.0", "max": "24.0", "start": "1"
            }
          },
          {
            "id": "hardware_fault", "name": "Hardware_fault", "desc": "Report hardware fault", "type": "fault", "required": false, "params":
            [
              {
                "id": "name", "name": "Name", "desc": "Name like: memory, tf card, sensors...", "define": {
                  "type": "string", "min": "0", "max": "64"
                },
                "id": "error_code", "name": "Error_Code", "desc": "Error code for fault", "define": {
                  "type": "int", "unit": "", "step": "1", "min": "0", "max": "2000", "start": "1"
                }
              }
            ]
          }
        ], "actions": []
      }
    ], "AppTemplate": "",
    "NetType": "else",
    "CategoryId": 141,
    "ProductType": 0,
    "UpdateTime": 1600265188
  }
],
"RequestId": "27HPvi1ylv"
}
}

```

5. Error Code

Error Code	Description
InternalError	Internal error.
InvalidParameterValue	Invalid parameter value.
InvalidParameterValue.InvalidAccessToken	The Token is invalid.

User Deletes Device

Last updated: 2025-04-30 15:30:34

1. API Description

api request domain name: `iot.cloud.tencent.com/api/exploreropen/tokenapi`.

This interface (AppDeleteDeviceInFamily) provides users with the feature to delete devices. After deletion, users need to reconfigure the network to proceed to bind.

2. Input Parameter

Name	Type	Required	Description
AccessToken	String	Yes	Common parameter, AccessToken is used for user authentication of a logged-in user.
RequestId	String	Yes	Common parameter, unique request ID, can be generated manually, uuid is recommended for use. Locating issues, need to provide the RequestId for the request.
Action	String	Yes	Common parameter, this interface takes values: AppDeleteDeviceInFamily.
FamilyId	String	Yes	Family ID, returned after successful creation of a family.
RoomId	String	Yes	room ID.
ProductId	String	Yes	product ID.
DeviceName	String	Yes	Device name.

3. Output Parameters

Name	Type	Description
RequestId	String	Common parameter, unique request ID, can be generated manually, uuid is recommended for use. Locating issues, need to provide the RequestId for the request.

4. Example

Input example

```
POST https://iot.cloud.tencent.com/api/exploreropen/tokenapi HTTP/1.1
content-type: application/json
{
  "RequestId": "42a96dbf-bdca-48f0-8f44-e36146d44e17",
  "Action": "AppDeleteDeviceInFamily",
  "AccessToken": "bf*****d098",
  "FamilyId": "1",
  "ProductId": "R32ONVL0EU",
  "DeviceName": "df2eSJyY"
}
```

Output example: successful

```
{
  "Response": {
    "RequestId": "42a96dbf-bdca-48f0-8f44-e36146d44e17"
  }
}
```

5. Error Code

Error Code	Description
InternalServerError	Internal error.
InvalidParameterValue	Invalid parameter value.
InvalidParameterValue.InvalidAccessToken	The Token is invalid.

Getting User Bound Device List

Last updated: 2025-04-30 15:30:50

1. API Description

api request domain name: `iot.cloud.tencent.com/api/exploreropen/tokenapi`.

This interface (AppGetFamilyDeviceList) is used to obtain the bound device list of the user.

2. Input Parameters

Name	Type	Required	Description
AccessToken	String	Yes	Common parameter. AccessToken is used for user authentication of a logged-in user.
RequestId	String	Yes	Common parameter. Unique request ID can be generated manually. It is recommended to use uuid. Locating issues, you need to provide the RequestId for the request.
Action	String	Yes	Common parameter. This interface takes values: AppGetFamilyDeviceList.
FamilyId	String	Yes	Family ID.
RoomId	String	Yes	room ID.
Offset	Int	No	Message offset.
Limit	Int	No	Maximum return message count, with a maximum value of 50.

3. Output Parameter

Name	Type	Description
RequestId	String	Common parameter. Unique request ID can be generated manually. It is recommended to use uuid. Locating issues, you need to provide the RequestId for the request.
Total	Int	Total number of devices.

DeviceList	Array	Device list, see DeviceList .
------------	-------	---

4. Example

input example

```
POST https://iot.cloud.tencent.com/api/exploreropen/tokenapi HTTP/1.1
content-type: application/json
{
  "RequestId": "req_1",
  "Action": "AppGetFamilyDeviceList",
  "FamilyId": "e0e23a2b33a24652b6*****a1cf",
  "RoomId": "5", //If you want to get all devices in the family, set
this field to "", "0" stands for the default room.
  "Offset": 0,
  "Limit": 2,
  "AccessToken": "bf*****d098"
}
```

Output Instance

```
{
  "Response": {
    "DeviceList": [{
      "ProductId": "R32*****EU",
      "DeviceName": "df2eSJyY",
      "DeviceId": "R32****0EU/df****yY",
      "AliasName": "df2eSJyY",
      "UserID": "1",
      "RoomId": "r_4b27c753ef774d458e*****86ad",
      "IconUrl": "",
      "CreateTime": 1574664969,
      "UpdateTime": 1574668779
    }],
    "RequestId": "req_1",
    "Total": 1
  }
}
```

5. Error Code

Error Code	Description
InternalError	Internal Error
InvalidParameterValue	Parameter value error
InvalidParameterValue.InvalidAccessToken	Token invalid

Get Current Device Status

Last updated: 2025-04-30 15:31:04

1. API Description

api request domain name: `iot.cloud.tencent.com/api/exploreropen/tokenapi`.

This interface (AppGetDeviceStatuses) is used to query device status.

2. Input Parameter

Name	Type	Required	Description
AccessToken	String	Yes	Common parameter. AccessToken is used for user authentication of a logged-in user.
RequestId	String	Yes	Common parameter. Unique request ID can be generated manually. It is recommended to use uuid. Locating issues, you need to provide the RequestId for the request.
Action	String	Yes	Common parameter. This interface takes values: AppGetDeviceStatuses.
DeviceIds	Array of String	Yes	Device ID array.

3. Output Parameters

Name	Type	Description
RequestId	String	Common parameter. Unique request ID can be generated manually. It is recommended to use uuid. Locating issues, you need to provide the RequestId for the request.
Devices	Array of Devices	Return the device.

4. Examples

Input example

```
POST https://iot.cloud.tencent.com/api/exploreropen/tokenapi HTTP/1.1
```

```
content-type: application/json

{
  "RequestId": "892497d8-a733-480b-91d5-5559c6b49551",
  "Action": "AppGetDeviceStatuses",
  "DeviceIds": ["HY4DHFM5P6/81386276"],
  "AccessToken": "6bf4003d7f*****400f3186f"
}
```

Output example: success

```
{
  "Response": {
    "DeviceStatuses": [{
      "DeviceId": "22F9Y6II70/light1",
      "ProductId": "22F9Y6II70",
      "DeviceName": "light1",
      "Online": 0 //0 Online; 1: Offline
    }],
    "RequestId": "892497d8-a733-480b-91d5-5559c6b49551"
  }
}
```

5. Error Code

Error Code	Description
InternalError	Internal error.
InvalidParameterValue	Invalid parameter value.
InvalidParameterValue.InvalidAccessToken	The Token is invalid.

Modifying Device Name

Last updated: 2025-04-30 15:31:20

1. API Description

api request domain name: `iot.cloud.tencent.com/api/exploreropen/tokenapi`.

This interface (AppUpdateDeviceInFamily) is used to modify the device name.

2. Input Parameters

Name	Type	Required	Description
AccessToken	String	Yes	Common parameter. AccessToken is used for user authentication of a logged-in user.
RequestId	String	Yes	Common parameter. Unique request ID can be generated manually. It is recommended to use uuid. Locating issues, you need to provide the RequestId for the request.
Action	String	Yes	Common parameter. This interface takes values: AppUpdateDeviceInFamily.
ProductId	String	Yes	product ID.
DeviceName	String	Yes	Device name.
AliasName	String	Yes	Device alias.

3. Output Parameters

Name	Type	Description
RequestId	String	Common parameter. Unique request ID can be generated manually. It is recommended to use uuid. Locating issues, you need to provide the RequestId for the request.
DeviceId	String	Device ID.

ProductId	String	product ID.
DeviceName	String	Device name.
AliasName	String	Device alias.
IconUrl	String	Icon URL.
FamilyId	String	Family ID.
RoomId	String	Room ID.
CreateTime	Int64	Creation time.
UpdateTime	Int64	Update time.

4. Example

Input example

```
POST https://iot.cloud.tencent.com/api/exploreropen/tokenapi HTTP/1.1
content-type: application/json
{
  "RequestId": "641150a3-4238-430d-888b-c01b11298351",
  "Action": "AppUpdateDeviceInFamily",
  "AccessToken": "xxxv2",
  "ProductId": "VZS14OZSN5",
  "DeviceName": "~virtualDev",
  "AliasName": "12345"
}
```

Output sample: successful

```
{
  "Response": {
    "Data": {
      "DeviceId": "R32ONVL0EU/df2eSJyY",
      "ProductId": "R32ONVL0EU",
      "DeviceName": "df2eSJyY",

```

```
      "AliasName": "12345",
      "IconUrl": "",
      "FamilyId": "",
      "RoomId": 0,
      "CreateTime": 1574859101,
      "UpdateTime": 1574859144
    },
    "RequestId": "641150a3-4238-430d-888b-c01b11298351"
  }
}
```

5. Error Code

Error Code	Description
InternalServerError	Internal error.
InvalidParameterValue	Invalid parameter value.
InvalidParameterValue.InvalidAccessToken	The Token is invalid.

Obtain Device Details

Last updated: 2025-04-30 15:31:35

1. API Description

api request domain name: `iot.cloud.tencent.com/api/exploreropen/tokenapi`.

This interface (AppGetDeviceInFamily) is used to query device details.

2. Input Parameter

Name	Type	Required	Description
AccessToken	String	Yes	Common parameter. AccessToken is used for user authentication of a logged-in user.
RequestId	String	Yes	Common parameter. Unique request ID can be generated manually. It is recommended to use uuid. When locating issues, you need to provide the RequestId for the request.
Action	String	Yes	Common parameter. This interface takes values: AppGetDeviceInFamily.
ProductId	String	Yes	product ID.
FamilyId	String	Yes	Family ID.
DeviceName	String	Yes	Device name.

3. Output Parameter

Name	Type	Description
RequestId	String	Common parameter. Unique request ID can be generated manually. It is recommended to use uuid. When locating issues, you need to provide the RequestId for the request.
DeviceId	String	Device ID is a combination of product ID and device name.

ProductId	String	product ID.
DeviceName	String	Device name.
AliasName	String	Device alias.
IconUrl	String	Icon URL.
FamilyId	String	Family ID.
RoomId	String	Room ID.
CreateTime	Int64	Creation time, UNIX second-level timestamp.
UpdateTime	Int64	Last update time, UNIX second-level timestamp.

4. Example

Input example

```
POST https://iot.cloud.tencent.com/api/exploreropen/tokenapi HTTP/1.1
content-type: application/json
{
  "RequestId": "082b70e-3d05-4179-9c94-6c70082e4a7",
  "Action": "AppGetDeviceInFamily",
  "ProductId": "R32ONVL0EU",
  "FamilyId": "xxx",
  "DeviceName": "df2eSJyY",
  "AccessToken": "xxxv2"
}
```

Output sample: successful

```
{
  "Response": {
    "Data": {
      "DeviceId": "R32ONVL0EU/df2eSJyY",
      "ProductId": "R32ONVL0EU",
      "DeviceName": "df2eSJyY",

```

```
"AliasName": "12345",
"IconUrl": "",
"FamilyId": "",
"RoomId": "",
"DeviceType": 0, // 0 Ordinary 1 Gateway device 2 Subdevice
"CreateTime": 1574931773,
"UpdateTime": 1574931945
},
"RequestId": "082b70e-3d05-4179-9c94-6c70082e4a7"
}
```

5. Error Code

Error Code	Description
InternalServerError	Internal error.
InvalidParameterValue	Invalid parameter value.
InvalidParameterValue.InvalidAccessToken	The Token is invalid.
UnauthorizedOperation	No permission to operate.
UnauthorizedOperation.APPNoPermissionToStudioProduct	The App has no permission to operate this product.
ResourceNotFound	The resource does not exist.
ResourceNotFound.StudioMemberNotExist	The family member is not created or has been deleted.

Retrieve Device Scale-Out Information

Last updated: 2025-04-30 15:31:51

1. API Description

api request domain name: `iot.cloud.tencent.com/api/exploreropen/tokenapi`.

This interface (AppGetDeviceExtInfo) is used to obtain the extended information of the device.

2. Input Parameter

Name	Type	Required	Description
AccessToken	String	Yes	common parameter, AccessToken is used for user authentication of a logged-in user.
RequestId	String	Yes	common parameter, unique request ID, can be generated manually, recommended for use uuld. Locating issues, need to provide the RequestId for the request.
Action	String	Yes	common parameter, this interface takes values: AppGetDeviceExtInfo.
DeviceId	String	Yes	Device ID.

3. Output Parameter

Name	Type	Description
RequestId	String	common parameter, unique request ID, can be generated manually, recommended for use uuld. Locating issues, need to provide the RequestId for the request.
ManufacturerName	String	Manufacturer Name.
ProductModel	String	Product model.
Mac	String	MAC address in Basic Information Reported by Device .

FwVer	String	Firmware version in Basic Information Reported by Device .
IP	String	The IP address of the device.

4. Example

Example 1

Input Example

```
POST https://iot.cloud.tencent.com/api/exploreropen/tokenapi HTTP/1.1
content-type: application/json
{
  "RequestId": "584406e9-bf59-46a3-8d45-39f1891bed7b",
  "Action": "AppGetDeviceExtInfo",
  "DeviceId": "PR*****ID/de***01",
  "AccessToken": "871d2d42*****692afae0"
}
```

Output example: successful

```
{
  "RequestId": "584406e9-bf59-46a3-8d45-39f1891bed7b",
  "ManufacturerName": "Manufacturer Name"
  "ProductModel": "product model"
  "Mac": "AABBCCDDEEFF",
  "FwVer": "1.2.0",
  "IP": ""
}
```

5. Error Code

Error Code	Description
InternalError	An internal error occurred.
InvalidParameterValue	Invalid parameter value.
InvalidParameterValue.InvalidAccessToken	The Token is invalid.

Equipment Replacement Room

Last updated: 2025-04-30 15:32:06

1. API Description

api request domain name: `iot.cloud.tencent.com/api/exploreropen/tokenapi`.

This interface (AppModifyFamilyDeviceRoom) is for Equipment Replacement Room.

2. Input Parameters

Name	Type	Required	Description
AccessToken	String	Yes	Common parameter. AccessToken is used for user authentication of a logged-in user.
RequestId	String	Yes	Common parameter. Unique request ID can be generated manually. It is recommended to use uuid. When locating issues, you need to provide the RequestId for the request.
Action	String	Yes	Common parameter. This interface takes values: AppModifyFamilyDeviceRoom.
FamilyId	String	Yes	Family ID.
RoomId	String	Yes	Room ID.
ProductId	String	Yes	Product ID.
DeviceName	String	Yes	Device name.

3. Output Parameter

Name	Type	Description
RequestId	String	Common parameter. Unique request ID can be generated manually. It is recommended to use uuid. When locating issues, you need to provide the RequestId for the request.

4. Example

Example 1

Input Example

```
POST https://iot.cloud.tencent.com/api/exploreropen/tokenapi HTTP/1.1
content-type: application/json

{
  "RequestId": "8c012671-3934-4dc2-b07c-82c5ba3766f2",
  "Action": "AppModifyFamilyDeviceRoom",
  "AccessToken": "8b4a70dd16105f*****18edd4e78a3bb8ec",
  "FamilyId": "1",
  "RoomId": "r_4b27c753ef774d458e*****86ad",
  "ProductId": "R32****0EU",
  "DeviceName": "df2eSJyY"
}
```

Output example: succeed

```
{
  "Response": {
    "RequestId": "8c012671-3934-4dc2-b07c-82*****766f2"
  }
}
```

5. Error Code

Error Code	Description
InternalError	Internal error.
InvalidParameterValue	Invalid parameter value.
InvalidParameterValue.InvalidAccessToken	The Token is invalid.

Add Sub-Device to User'S Bound List

Last updated: 2025-04-30 15:32:21

1. API Description

API request domain name: `iot.cloud.tencent.com/api/exploreropen/tokenapi`.

This interface (AppBindSubDeviceInFamily) is used to bind a sub-device that has been bound to a gateway device to a family.

2. Input Parameter

Name	Type	Required	Description
AccessToken	String	Yes	Common parameter. AccessToken is used for user authentication of a logged-in user.
RequestId	String	Yes	Common parameter. Unique request ID can be generated manually. It is recommended to use uuid. When locating issues, provide the RequestId for the request.
Action	String	Yes	Common parameter. This interface takes values: AppBindSubDeviceInFamily.
GatewayProductId	String	Yes	Gateway product ID.
GatewayDeviceName	String	Yes	Gateway device name.
ProductId	String	Yes	Sub-product ID.
DeviceName	String	Yes	Subdevice name.

3. Output Parameters

Name	Type	Description
RequestId	String	Common parameter. Unique request ID, same as the input parameter. When locating issues, provide the RequestId for the request.

AppDeviceInfo o	Object of AppDeviceInfo	Returned result
--------------------	--	-----------------

4. Example

Example 1

Input Example

```
POST https://iot.cloud.tencent.com/api/exploreropen/tokenapi HTTP/1.1
content-type: application/json
{
  "RequestId": "keyi20191119-012345",
  "Action": "AppBindSubDeviceInFamily",
  "AccessToken": "8b4a70dd16105f*****18edd4e78a3bb8ec",
  "ProductId": "LAEG4YJE1A",
  "DeviceName": "subdev2",
  "GatewayProductId": "NJ27OVLZT4",
  "GatewayDeviceName": "gwdev"
}
```

Output example: successful

```
{
  "Response": {
    "Data": {
      "AppDeviceInfo": {
        "DeviceId": "R32ONVL0EU/df2eSJyY",
        "ProductId": "R32ONVL0EU",
        "DeviceName": "df2eSJyY",
        "AliasName": "",
        "IconUrl": "",
        "FamilyId": "1",
        "RoomId": "0",
        "CreateTime": 1574152773,
        "UpdateTime": 1574152773
      }
    },
    "RequestId": "keyi20191119-012345"
  }
}
```

```
}  
  
}
```

5. Error Code

Error Code	Description
InternalError	Internal error.
InvalidParameterValue	Invalid parameter value.
InvalidParameterValue.InvalidAccessToken	The Token is invalid.

Get the Sub-Device List of the Designated Gateway Device

Last updated: 2025-04-30 15:32:35

1. API Description

api request domain name: `iot.cloud.tencent.com/api/exploreropen/tokenapi`.

this interface (AppGetGatewayBindDeviceList) is used to obtain the sub-device list of the designated gateway device.

2. Input Parameters

Name	Type	Required	Description
AccessToken	String	Yes	Common parameter. AccessToken is used for user authentication of a logged-in user.
RequestId	String	Yes	Common parameter. Unique request ID can be generated manually. It is recommended to use uuid. When locating issues, provide the RequestId for the request.
Action	String	Yes	Common parameter. This interface takes values: AppGetGatewayBindDeviceList.
ProductId	String	Yes	Subdevice's ProductId.
GatewayProductId	String	Yes	ProductId of the gateway device.
GatewayDeviceName	String	Yes	DeviceName of the gateway device.
Offset	Int	Yes	Message offset.
Limit	Int	Yes	Maximum return message count, with a maximum value of 50.

3. Output Parameters

Name	Type	Description
------	------	-------------

RequestId	String	Common parameter. Unique request ID can be generated manually. It is recommended to use uuid. When locating issues, provide the RequestId for the request.
Total	String	Total number.
DeviceList	Array of DeviceList	Device list.

4. Example

Input example

```
POST https://iot.cloud.tencent.com/api/exploreropen/tokenapi HTTP/1.1
content-type: application/json
{
  "RequestId": "req_1",
  "Action": "AppGetGatewayBindDeviceList",
  "ProductId": "LAEG4YJE1A" // Subdevice's productId
  "GatewayProductId": "NJ27OVLZT4", //gateway device's productId
  "GatewayDeviceName": gwdev // deviceName of the gateway device
  "Offset": 0,
  "Limit": 2,
  "AccessToken": "8b4a70dd16105f*****18edd4e78a3bb8ec"
}
```

Output example

```
{
  "Response": {
    "DeviceList": [{
      "ProductId": "LAEG4YJE1A",
      "DeviceName": "subdev2"
      "DeviceId": "LAEG4YJE1A/subdev2", //subdevice's deviceId
      "AliasName": "",
      "IconUrl": "",
      "BindStatus": 0 //not bound to family
    }, {
      "ProductId": "LAEG4YJE1A",
      "DeviceName": "subdev1",

```

```
        "DeviceId": "LAEG4YJE1A/subdev1",
        "AliasName": "",
        "IconUrl": "",
        "BindStatus": 1 //already bound to family
    }],
    "RequestId": "req_1",
    "Total": 2
}
}
```

5. Error Code

Error Code	Description
InternalServerError	Internal error.
InternalServerError.InternalRPCError	Call timeout.
InternalServerError.InternalRPCError	The call returned failure.
InvalidParameterValue	Invalid parameter value.
InvalidParameterValue.InvalidAccessToken	Token invalid.
UnsupportedOperation	Unsupported operation.
UnsupportedOperation.NotSupportVirtualDevice	Virtual devices are not supported.
UnsupportedOperation.DeviceNotGateway	Not a gateway device.
ResourceNotFound	The resource does not exist.
ResourceNotFound.DeviceNotExist	The device is not created or has been deleted.
ResourceNotFound.StudioMemberNotExist	Family members are not created or have been deleted.

Get the Sub-Device List of the Specified Gateway Bound to the Family

Last updated: 2025-04-30 15:32:52

1. API Description

api request domain name: `iot.cloud.tencent.com/api/exploreropen/tokenapi`.

This interface (AppGetFamilySubDeviceList) is used to obtain the sub-device list of gateways under a family.

2. Input Parameters

Name	Type	Required	Description
AccessToken	String	Yes	Common parameter. AccessToken is used for user authentication of a logged-in user.
RequestId	String	Yes	Common parameter. Unique request ID can be generated manually. It is recommended to use uuid. Locating issues, you need to provide the RequestId for the request.
Action	String	Yes	Common parameter. This interface takes values: AppGetFamilySubDeviceList.
GatewayProductId	String	Yes	Gateway device product ID.
GatewayDeviceName	String	Yes	Gateway device Name.
Offset	Int	Yes	Message offset.
Limit	Int	Yes	Maximum return message count, with a maximum value of 50.

3. Output Parameter

Name	Type	Description
------	------	-------------

RequestId	String	Common parameter. Unique request ID can be generated manually. It is recommended to use uuid. Locating issues, you need to provide the RequestId for the request.
Total	String	Total number.
DeviceList	Array of DeviceList	Device List.

4. Samples

Input example

```
POST https://iot.cloud.tencent.com/api/exploreropen/tokenapi HTTP/1.1
content-type: application/json
{
  "RequestId": "req_1",
  "Action": "AppGetFamilySubDeviceList",
  "GatewayProductId": "NJ27OVLZT4",
  "GatewayDeviceName": "gwdev",
  "Offset": 0,
  "Limit": 2,
  "AccessToken": "8b4a70dd16105f*****18edd4e78a3bb8ec"
}
```

Output example

```
{
  "Response": {
    "DeviceList": [{
      "FamilyId": "f_9b309d84c962****0a11b4c3d9588fcc1",
      "ProductId": "LAEG4YJE1A",
      "DeviceName": "subdev1",
      "DeviceId": "LAEG4YJE1A/subdev1",
      "AliasName": "",
      "RoomId": "0",
      "IconUrl": "",
      "DeviceType": 2,
      "CreateTime": 1583492992,
      "UpdateTime": 1583492992
    }
  ]
}
```

```
    },
    "RequestId": "req_1",
    "Total": 1
  }
}
```

5. Error Code

Error Code	Description
InternalServerError	Internal error.
InternalServerError.InternalRPCError	Call timeout.
InternalServerError.InternalRPCError	The call returned failure.
InvalidParameterValue	Invalid parameter value.
InvalidParameterValue.InvalidAccessToken	The Token is invalid.
InvalidParameterValue.NoPermission	The user has no permission to this device.
UnsupportedOperation	Unsupported operation.
UnsupportedOperation.ERR_APP_INVALID_IOTAPPID	The AppId is invalid. lotAppID.
ResourceNotFound	The resource does not exist.
ResourceNotFound.DeviceNotExist	The device is not created or has been deleted.

Bluetooth Data Reporting

Last updated: 2025-04-30 15:33:06

1. API Description

api request domain name: `iot.cloud.tencent.com/api/exploreropen/tokenapi`.

This interface (AppReportDataAsDevice) is used for data reporting by mini programs or apps.

2. Input Parameters

Name	Type	Required	Description
AccessToken	String	Yes	Common parameter. AccessToken is used for user authentication of a logged-in user.
RequestId	String	Yes	Common parameter. Unique request ID can be generated manually. It is recommended to use uuid. Locating issues requires providing the RequestId for the request.
Action	String	Yes	Common parameter. This interface takes values: AppReportDataAsDevice.
DeviceId	String	Yes	Device ID.
Data	String	Yes	Modified model data.
TimeStamp	Int64	Yes	Unix millisecond-level timestamp.

3. Output Parameter

Name	Type	Description
RequestId	String	Common parameter. Unique request ID can be generated manually. It is recommended to use uuid. Locating issues requires providing the RequestId for the request.
Data	String	Returned data information.

4. Example

Input Example

```
POST https://iot.cloud.tencent.com/api/exploreropen/tokenapi HTTP/1.1
content-type: application/json
{
  "RequestId": "123e4567-e89b-12d3-a456-426614174000",
  "Action": "AppReportDataAsDevice",
  "DeviceId": "22*****O/light1",
  "Data": "{\"light_switch\":0}",
  "TimeStamp": 1581422682000,
  "AccessToken": "8b4a70dd16*****2826868b18edd4e78a3bb8ec"
}
```

Output example: succeed

```
{
  "Response": {
    "Data": "",
    "RequestId": "123e4567-e89b-12d3-a456-426614174000"
  }
}
```

5. Error Code

Error Code	Description
InternalServerError	Internal error.
InvalidParameterValue	Invalid parameter value.
InvalidParameterValue.BindDeviceNotConnected	The device has not connected to the cloud recently.
InvalidParameterValue.InvalidAccessToken	The Token is invalid.

Bluetooth Device Sends Reply Messages for Device Behavior

Last updated: 2025-04-30 15:33:19

1. API Description

api request domain name: `iot.cloud.tencent.com/api/exploreropen/tokenapi`.

This interface (AppPublishMsgAsDevice) provides the message reply feature for mini programs or apps to simulate device behavior.

2. Input Parameter

Name	Type	Required	Description
AccessToken	String	Yes	Common parameter. AccessToken is used for user authentication of a logged-in user.
RequestId	String	Yes	Common parameter. Unique request ID can be generated manually. It is recommended to use uuid. Locating issues, you need to provide the RequestId for the request.
Action	String	Yes	Common parameter. This interface takes values: AppPublishMsgAsDevice.
ProductId	String	Yes	product ID.
DeviceName	String	Yes	Device name.
Topic	String	Yes	Reporting Topic.
Payload	String	Yes	Message content.

3. Output Parameter

Name	Type	Description
RequestId	String	Common parameter. Unique request ID can be generated manually. It is recommended to use uuid. Locating issues, you need to provide the RequestId for the request.

4. Example

Input Example

```
POST https://iot.cloud.tencent.com/api/exploreropen/tokenapi HTTP/1.1
content-type: application/json
{
  "RequestId": "123e4567-e89b-12d3-a456-426614174000",
  "Action": "AppPublishMsgAsDevice",
  "ProductId": "22*****O",
  "DeviceName": "light1",
  "Topic": "$thing/up/action/22*****O/light1"
  "Payload": "e1wibGlnaHRfc3dpdGNoXCI6MH0="
  "AccessToken": "8b4a70dd16*****2826868b18edd4e78a3bb8ec"
}
```

Output sample: successful

```
{
  "Response": {
    "RequestId": "123e4567-e89b-12d3-a456-426614174000"
  }
}
```

5. Error Code

Error Code	Description
InternalError	Internal error.
InvalidParameterValue	Parameter value error.
InvalidParameterValue.BindDeviceNotConnected	The device has not connected to the cloud recently.
InvalidParameterValue.InvalidAccessToken	The Token is invalid.

Bluetooth Device Reporting Event

Last updated: 2025-04-30 15:33:33

1. API Description

api request domain name: `iot.cloud.tencent.com/api/exploreropen/tokenapi`.

This interface (AppReportDeviceEvent) is used for bluetooth devices to report events through App or mini program.

2. Input Parameter

Name	Type	Required	Description
AccessToken	String	Yes	Common parameter. AccessToken is used for user authentication of a logged-in user.
RequestId	String	Yes	Common parameter. Unique request ID can be generated manually. It is recommended to use uuid. When locating issues, provide the RequestId for the request.
Action	String	Yes	Common parameter. This interface takes values: AppReportDeviceEvent.
DeviceId	String	Yes	Device ID, composed of ProductId/DeviceName.
EventId	String	Yes	Event ID.
Params	String	Yes	Event parameter in JSON format.
Method	String	Yes	Reporting method, can be ReportEventAsDevice or reported (default).
EventTimeStamp	Int64	Yes	Unix millisecond-level timestamp.

3. Output Parameters

Name	Type	Description
RequestId	String	Common parameter. Unique request ID can be generated manually. It is recommended to use uuid.

When locating issues, provide the RequestId for the request.

4. Examples

Input example

```
POST https://iot.cloud.tencent.com/api/exploreropen/tokenapi HTTP/1.1
content-type: application/json
{
  "RequestId": "123e4567-e89b-12d3-a456-426614174000",
  "Action": "AppReportDeviceEvent",
  "DeviceId": "QD****BNB/dev01",
  "EventId": "low_voltage",
  "Params": "{\"voltage\":3.1}",
  "Method": "ReportEventAsDevice",
  "AccessToken": "ningtoken*****"
}
```

Output sample: successful

```
{
  "Response": {
    "Data": "",
    "RequestId": "123e4567-e89b-12d3-a456-426614174000"
  }
}
```

5. Error Code

Error Code	Description
InternalServerError	Internal error.
InvalidParameterValue	Invalid parameter value.
InvalidParameterValue.BindDeviceNotConnected	The device has not connected to the cloud recently.
InvalidParameterValue.InvalidAccessToken	The Token is invalid.

Network Configuration Binding Device

Last updated: 2025-04-30 15:33:53

1. API Description

api request domain name: `iot.cloud.tencent.com/api/exploreropen/tokenapi`.

This interface (AppSigBindDeviceInFamily) is used for mini programs or apps to perform binding operations for Wi-Fi device networking.

2. Input Parameter

Name	Type	Required	Description
AccessToken	String	Yes	Common parameter. AccessToken is used for user authentication of a logged-in user.
RequestId	String	Yes	Common parameter. Unique request ID can be generated manually. It is recommended to use uuid. When locating issues, you need to provide the RequestId for the request.
Action	String	Yes	Common parameter. This interface takes values: AppSigBindDeviceInFamily.
FamilyId	String	Yes	Family ID.
ProductId	String	Yes	Product ID.
DeviceName	String	Yes	Device name.
RoomId	String	No	Room ID.
DeviceTimestamp	Int64	Yes	Device timestamp, Unix second-level timestamp.
ConnId	String	No	Random string, recommend 5 bytes length.
Signature	String	Yes	Dynamic signature, a signature transmitted to the mini program and App by the device according to the Distribution Network Protocol. For calculation example, see below.

BindType	String	Yes	Binding type. - wifi_sign: WIFI binding (default). - bluetooth_sign: Bluetooth binding. - other_sign: Other.
SignMethod	String	No	Signature algorithm. - hmacsha1: HMACSHA1 encryption algorithm (default). - hmacsha256: HMACSHA256 encryption algorithm.

3. Output Parameters

Name	Type	Description
RequestId	String	Common parameter. Unique request ID can be generated manually. It is recommended to use uuid. When locating issues, you need to provide the RequestId for the request.
AppDeviceInfo	Object of AppDeviceInfo	Equipment information.

4. Dynamic Signature Calculation

Procedure for calculating signature

1. Confirm whether it is WIFI configuration, Bluetooth configuration, or other configurations.
The concatenation of plaintext strings that need encryption varies slightly.
2. Obtain the psk of the device, which can be viewed in the console or obtained via API.
3. Decode the psk through base64 and use it as the key of the hash to encode the plaintext.
4. Just concatenate the result of the hash into a hexadecimal string.

The steps for calculating the signature for WIFI binding are the same as those for Bluetooth or other bindings. The only difference is the concatenation of the plaintext for encryption. For details, see the example code.

WIFI Network Configuration Binding Signature

Plaintext concatenation format: "DeviceName=%s&DeviceTimestamp=%d&ProductId=%s&ConnId=%s"

```
Java
```

```
import javax.crypto.Mac;
import javax.crypto.spec.SecretKeySpec;
import java.util.Base64;

public class AppBind {
    private static final String HMAC_SHA1_ALGORITHM = "HmacSHA1";
    private static final String HMAC_SHA256_ALGORITHM = "HmacSHA256";
    public static final String DEVICE_PSK = "4B***==";
    public static final String PRODUCT_ID = "productId";
    public static final String DEVICE_NAME = "d1";
    public static final String CONN_ID = "12345";
    public static final int TIMESTAMP = 1694141664;
    public static final String ContentFormat =
"DeviceName=%s&DeviceTimestamp=%d&ProductId=%s&ConnId=%s";

    public static void main(String[] args) throws Exception {
        String content = String.format(ContentFormat, DEVICE_NAME,
TIMESTAMP, PRODUCT_ID, CONN_ID);
        // sha1
        System.out.println(calculateHMACSHA1(content,
Base64.getDecoder().decode(DEVICE_PSK)));
        // sha256
        System.out.println(calculateHMACSHA256(content,
Base64.getDecoder().decode(DEVICE_PSK)));
    }

    // Calculate sha1 signature
    public static String calculateHMACSHA1(String data, byte[] keyBytes)
throws Exception {
        SecretKeySpec signingKey = new SecretKeySpec(keyBytes,
HMAC_SHA1_ALGORITHM);
        Mac mac = Mac.getInstance(HMAC_SHA1_ALGORITHM);
        mac.init(signingKey);
        byte[] dataBytes = data.getBytes();
        byte[] signatureBytes = mac.doFinal(dataBytes);

        return bytesToHex(signatureBytes);
    }
}
```



```
// Calculate sha256 signature
public static String calculateHMACSHA256(String data, byte[]
keyBytes) throws Exception {
    SecretKeySpec signingKey = new SecretKeySpec(keyBytes,
HMAC_SHA1_ALGORITHM);
    Mac mac = Mac.getInstance(HMAC_SHA256_ALGORITHM);
    mac.init(signingKey);
    byte[] dataBytes = data.getBytes();
    byte[] signatureBytes = mac.doFinal(dataBytes);

    return bytesToHex(signatureBytes);
}

private static String bytesToHex(byte[] bytes) {
    StringBuilder builder = new StringBuilder();
    for (byte b : bytes) {
        builder.append(String.format("%02x", b));
    }

    return builder.toString();
}
}
```

Go

```
import (
    "crypto/hmac"
    "crypto/sha1"
    "encoding/base64"
    "encoding/hex"
    "fmt"
    "hash"
)

var (
    devicePsk    = "Re****Q=="
    contentFmt   = "DeviceName=%s&DeviceTimestamp=%d&ProductId=%s&ConnId=%s"
)
```

```

func main() {
    fmt.Println(getSign("deviceName", "productId", "connId",
1693898532))
}

// Get sha1 signature
func getSha1Sign() string {
    psk, _ := base64.StdEncoding.DecodeString(devicePsk)
    var mac hash.Hash
    mac = hmac.New(sha1.New, psk)
    mac.Write([]byte(contentFmt))
    result := mac.Sum(nil)

    return hex.EncodeToString(result)
}

// Get sha256 signature
func getSha256Sign() string {
    psk, _ := base64.StdEncoding.DecodeString(devicePsk)
    var mac hash.Hash
    mac = hmac.New(sha256.New, psk)
    mac.Write([]byte(contentFmt))
    result := mac.Sum(nil)

    return hex.EncodeToString(result)
}

```

C

```

#include <stdio.h>
#include <string.h>
#include <openssl/hmac.h>

const char *devicePsk = "Re5***==";
const char *contentFmt =
"DeviceName=%s&DeviceTimestamp=%ld&ProductId=%s&ConnId=%s";

char *getSign(char *deviceName, char *productId, char *connId, long
deviceTimestamp);

```

```
int main() {
    printf("%s\n", getSign("d1", "BRS**F", "9102344", 1693898532));
    return 0;
}

char *getSign(char *deviceName, char *productId, char *connId, long
deviceTimestamp) {
    unsigned char psk[16];
    int psk_len = EVP_DecodeBlock(psk, devicePsk, strlen(devicePsk));
    printf("%s\n", psk);

    char content[256];
    sprintf(content, contentFmt, deviceName, deviceTimestamp, productId,
connId);
    printf("%s\n", content);

    unsigned char result[20];
    unsigned int result_len;
    HMAC(EVP_sha1(), psk, psk_len, (unsigned char *) content,
strlen(content), result, &result_len);

    char *hexResult = malloc(result_len * 2 + 1);
    for (int i = 0; i < result_len; i++) {
        sprintf(hexResult + i * 2, "%02X", result[i]);
    }
    hexResult[result_len * 2] = '\0';
    return hexResult;
}
```

Bluetooth and Other Network Configuration Binding Signatures

Plaintext concatenation format: "%s%s;%s;%d". The fields are `ProductId`, `DeviceName`, `ConnId`, `DeviceTimestamp` in sequence.

Java

```
import javax.crypto.Mac;
import javax.crypto.spec.SecretKeySpec;
```

```
import java.util.Base64;

public class AppBind {
    private static final String HMAC_SHA1_ALGORITHM = "HmacSHA1";
    private static final String HMAC_SHA256_ALGORITHM = "HmacSHA256";
    public static final String DEVICE_PSK = "4B***==";
    public static final String PRODUCT_ID = "productId";
    public static final String DEVICE_NAME = "d1";
    public static final String CONN_ID = "12345";
    public static final int TIMESTAMP = 1694141664;
    public static final String ContentFormat = "%s%s;%s;%d"; //
    ProductId;DeviceName;ConnId;DeviceTimestamp

    public static void main(String[] args) throws Exception {
        String content = String.format(ContentFormat, PRODUCT_ID,
        DEVICE_NAME, CONN_ID, TIMESTAMP);
        // sha1
        System.out.println(calculateHMACSHA1(content,
        Base64.getDecoder().decode(DEVICE_PSK)));
        // sha256
        System.out.println(calculateHMACSHA256(content,
        Base64.getDecoder().decode(DEVICE_PSK)));
    }

    // Calculate sha1 signature
    public static String calculateHMACSHA1(String data, byte[] keyBytes)
    throws Exception {
        SecretKeySpec signingKey = new SecretKeySpec(keyBytes,
        HMAC_SHA1_ALGORITHM);
        Mac mac = Mac.getInstance(HMAC_SHA1_ALGORITHM);
        mac.init(signingKey);
        byte[] dataBytes = data.getBytes();
        byte[] signatureBytes = mac.doFinal(dataBytes);

        return bytesToHex(signatureBytes);
    }

    // Calculate sha256 signature
    public static String calculateHMACSHA256(String data, byte[]
    keyBytes) throws Exception {
```

```

        SecretKeySpec signingKey = new SecretKeySpec(keyBytes,
HMAC_SHA1_ALGORITHM);

        Mac mac = Mac.getInstance(HMAC_SHA256_ALGORITHM);
        mac.init(signingKey);

        byte[] dataBytes = data.getBytes();
        byte[] signatureBytes = mac.doFinal(dataBytes);

        return bytesToHex(signatureBytes);
    }

    private static String bytesToHex(byte[] bytes) {
        StringBuilder builder = new StringBuilder();
        for (byte b : bytes) {
            builder.append(String.format("%02x", b));
        }

        return builder.toString();
    }
}

```

Go

```

import (
    "crypto/hmac"
    "crypto/sha1"
    "encoding/base64"
    "encoding/hex"
    "fmt"
    "hash"
)

var (
    devicePsk          = "Re****=="
    ProductId          = "productId"
    DeviceName         = "d1"
    ConnId             = "12345"
    DeviceTimestamp int64 = 1694141664
    contentFmt         = fmt.Sprintf("%s%s;%s;%d", ProductId,
DeviceName, ConnId, DeviceTimestamp) //

```

```

ProductId;DeviceName;ConnId;DeviceTimestamp)

func main() {
    fmt.Println(getSign())
}

func getSign() string {
    psk, _ := base64.StdEncoding.DecodeString(devicePsk)
    var mac hash.Hash
    mac = hmac.New(sha1.New, psk)
    mac.Write([]byte(contentFmt))
    result := mac.Sum(nil)

    return hex.EncodeToString(result)
}

```

C

```

#include <stdio.h>
#include <string.h>
#include <openssl/hmac.h>

const char *devicePsk = "Re5***==";
const char *contentFmt = "%s%s;%s;%d"; //
ProductId;DeviceName;ConnId;DeviceTimestamp

char *getSign(char *deviceName, char *productId, char *connId, long
deviceTimestamp);

int main() {
    printf("%s\n", getSign("d1", "BRS**F", "9102344", 1693898532));
    return 0;
}

char *getSign(char *deviceName, char *productId, char *connId, long
deviceTimestamp) {
    unsigned char psk[16];
    int psk_len = EVP_DecodeBlock(psk, devicePsk, strlen(devicePsk));
    printf("%s\n", psk);

    char content[256];

```

```

    sprintf(content, contentFmt, productId, deviceName, connId,
deviceTimestamp);
    printf("%s\n", content);

    unsigned char result[20];
    unsigned int result_len;
    HMAC(EVP_sha1(), psk, psk_len, (unsigned char *) content,
strlen(content), result, &result_len);

    char *hexResult = malloc(result_len * 2 + 1);
    for (int i = 0; i < result_len; i++) {
        sprintf(hexResult + i * 2, "%02X", result[i]);
    }
    hexResult[result_len * 2] = '\0';

    return hexResult;

```

5. Example

Default WIFI Network Configuration Binding request input example 1:

```

POST https://iot.cloud.tencent.com/api/exploreropen/tokenapi HTTP/1.1
content-type: application/json
{
    "DeviceTimestamp": 1694141664,
    "Action": "AppSigBindDeviceInFamily",
    "ConnId": "1938",
    "RequestId": "dd7b-1014bc4-rey76",
    "AccessToken": "51dbxxx4d4fdb",
    "FamilyId": "f_8xxxxfb1a",
    "Signature": "7CE3518***69EA8C",
    "ProductId": "productId",
    "RoomId": ""
    "BindType": "wifi_sign",
    "DeviceName": "d1"
}

```

Default WIFI Network Configuration Binding request input example 2:

```

POST https://iot.cloud.tencent.com/api/exploreropen/tokenapi HTTP/1.1

```

```
content-type: application/json

{
  "DeviceTimestamp": 1694141664,
  "Action": "AppSigBindDeviceInFamily",
  "ConnId": "1938",
  "RequestId": "dd7b-1014bc4-rey76",
  "AccessToken": "51dbxxx4d4fdb",
  "FamilyId": "f_8xxxxfb1a",
  "Signature": "7CE3518***69EA8C",
  "ProductId": "productId",
  "RoomId": ""
  "BindType": "wifi_sign",
  "SignMethod": "hmacsha256",
  "DeviceName": "d1"
}
```

Default Bluetooth Network Configuration Binding Request Input Example 3:

```
POST https://iot.cloud.tencent.com/api/exploreropen/tokenapi HTTP/1.1
content-type: application/json

{
  "DeviceTimestamp": 1694141664,
  "Action": "AppSigBindDeviceInFamily",
  "ConnId": "1938",
  "RequestId": "dd7b-1014bc4-rey76",
  "AccessToken": "51dbxxx4d4fdb",
  "FamilyId": "f_8xxxxfb1a",
  "Signature": "7CE3518***69EA8C",
  "ProductId": "productId",
  "RoomId": ""
  "BindType": "bluetooth_sign",
  "DeviceName": "d1"
}
```

Default Bluetooth Network Configuration Binding request input example 4:

```
POST https://iot.cloud.tencent.com/api/exploreropen/tokenapi HTTP/1.1
content-type: application/json

{
```



```
"DeviceTimestamp": 1694141664,
"Action": "AppSigBindDeviceInFamily",
"ConnId": "1938",
"RequestId": "dd7b-1014bc4-rey76",
"AccessToken": "51dbxxx4d4fdb",
"FamilyId": "f_8xxxxfb1a",
"Signature": "7CE3518***69EA8C",
"ProductId": "productId",
"RoomId": ""
"BindType": "bluetooth_sign",
"SignMethod": "hmacsha256",
"DeviceName": "d1"
}
```

Output sample: Success.

```
{
  "Response": {
    "RequestId": "req_1",
    "data": {
      "AppDeviceInfo": {
        "DeviceId": "45454/d2",
        "ProductId": "45454",
        "DeviceName": "d2",
        "AliasName": "",
        "CreateTime": 1552910676,
        "UpdateTime": 1552910676,
        "FamilyId": "1",
        "RoomId": "0",
        "IconUrl": ""
      }
    }
  }
}
```

5. Error Code

Error Code	Description
------------	-------------

InternalError	Internal error.
InvalidParameterValue	Invalid parameter value.
InvalidParameterValue.BindDeviceNotConnected	The device has not connected to the cloud recently.
InvalidParameterValue.InvalidAccessToken	The Token is invalid.

Bind Sub-Device to Gateway

Last updated: 2025-04-30 15:34:21

1. API Description

api request domain name: api request domain name replacement

this interface (AppSigGatewayBindDevice) binds a sub-device to a gateway device

2. Input Parameter

Name	Type	Required	Description
AccessToken	String	Yes	Common parameter. AccessToken is used for user authentication of a logged-in user.
RequestId	String	Yes	Common parameter. Unique request ID can be generated manually. It is recommended to use uuid. Locating issues, you need to provide the RequestId for the request.
Action	String	Yes	Common parameter. This interface takes values: AppSigGatewayBindDevice.
ProductId,	String	Yes	Product ID.
DeviceName	String	Yes	Device name.
GatewayProductId	String	Yes	Gateway product ID.
GatewayDeviceName	String	Yes	Gateway device Name.
DeviceTimestamp	Int64	Yes	Device timestamp, Unix second-level timestamp.
Nonce	String	Yes	Random string, recommend 5 bytes length.
Signature	String	Yes	Dynamic Signature.

3. Output Parameters

Name	Type	Description
------	------	-------------

RequestId	String	Common parameter. Unique request ID can be generated manually. It is recommended to use uuid. Locating issues, you need to provide the RequestId for the request.
-----------	--------	---

4. Samples

Signature generation procedure:

1. Sort the (deviceName, nonce, productId, timestamp) parameters in ascending lexicographical order.
2. Concatenate the above parameters into strings in the form of parameter name = parameter value & parameter name = parameter value.
3. Use the HMAC-sha1 algorithm to calculate the string obtained in the previous step. The key is ProductSecret.
4. Encode the generated result using Base64, and you can obtain the final signature string to put into signature.

Example 1

Input example

```
POST https://iot.cloud.tencent.com/api/exploreropen/tokenapi HTTP/1.1
content-type: application/json
{
  "RequestId": "zzY1pZb3sz1N",
  "Action": "AppSigGatewayBindDevice",
  "AccessToken": "8b4a70dd16105f*****18edd4e78a3bb8ec",
  "GatewayProductId": "NJ27OVLZT4",
  "GatewayDeviceName": "gwdev",
  "ProductId": "LAEG4YJE1A",
  "DeviceName": "subdev3",
  "DeviceTimestamp": 1583810805,
  "Nonce": "123456",
  "Signature": "YZGf0E***WGUEJGn+SxUf23HYA="
}
```

Output example: Success

```
{
  "Response": {
```

```
"RequestId": "zzY1pZb3sz1N"
}
```

5. Error Code

Error Code	Description
InternalServerError	Internal error.
InternalServerError.InternalServerError	An error occurs.
InvalidParameterValue	Invalid parameter value.
InvalidParameterValue.BindDeviceNotConnected	The device has not connected to the cloud recently.
InvalidParameterValue.InvalidAccessToken	The Token is invalid.
InvalidParameterValue.InvalidAppParameter	The App's request has invalid parameters.
InvalidParameterValue.BindDeviceSigMismatch	Dynamic signature error when binding device.
UnsupportedOperation	Unsupported operation.
UnsupportedOperation.NotBindVirtualDevice	Unable to bind virtual device.
UnsupportedOperation.DeviceType	Device type error.
ResourceNotFound	The resource does not exist.
ResourceNotFound.StudioProductNotExist	The product is not created or has been deleted.
ResourceNotFound.DeviceNotExist	The device is not created or has been deleted.
ResourceNotFound.AppNotExists	The application is not created or has been deleted.
UnauthorizedOperation	No permission to operate.
UnauthorizedOperation.APPNoPermissionToStudioProduct	The App has no permission to operate this product.

UnauthorizedOperation.NoPermissionToFamily	No permission to operate this family.
LimitExceeded	Exceed the limit.
LimitExceeded.FamilyDeviceExceedLimit	The device quantity exceeds the limit.

Unbind Sub-Device Gateway

Last updated: 2025-04-30 15:34:35

1. API Description

api request domain name: api request domain name replacement

this interface (AppGatewayUnbindDevice) unbinds a sub-device from the gateway

2. Input Parameters

Name	Type	Required	Description
AccessToken	String	Yes	Common parameter. AccessToken is used for user authentication of a logged-in user.
RequestId	String	Yes	Common parameter. Unique request ID can be generated manually. It is recommended to use uuid. Locating issues, you need to provide the RequestId for the request.
Action	String	Yes	Common parameter. This interface takes values: AppGatewayUnbindDevice.
ProductId,	String	Yes	Product ID.
DeviceName	String	Yes	Device name.
GatewayProductId	String	Yes	Gateway product ID.
GatewayDeviceName	String	Yes	Gateway device Name.

3. Output Parameter

Name	Type	Description
RequestId	String	Common parameter. Unique request ID can be generated manually. It is recommended to use uuid. Locating issues, you need to provide the RequestId for the request.

4. Samples

Example 1

Input Example

```
POST https://iot.cloud.tencent.com/api/exploreropen/tokenapi HTTP/1.1
content-type: application/json
{
  "RequestId": "zzY1pZb3sz1N",
  "Action": "AppGatewayUnbindDevice",
  "AccessToken": "8b4a70dd16105f*****18edd4e78a3bb8ec",
  "GatewayProductId": "NJ****T4",
  "GatewayDeviceName": "gwdev",
  "ProductId": "LA*****1A",
  "DeviceName": "dev"
}
```

Output example: successful

```
{
  "Response": {
    "RequestId": "zzY1pZb3sz1N"
  }
}
```

5. Error Code

Error Code	Description
InternalServerError	Internal error.
InternalServerError.InternalServerError	An error occurs.
InvalidParameterValue	Invalid parameter value.
InvalidParameterValue.BindDeviceNotConnected	The device has not connected to the cloud recently.
InvalidParameterValue.InvalidAccessToken	The Token is invalid.
InvalidParameterValue.InvalidAppParameter	Wrong parameters in the App request.
InvalidParameterValue.BindDeviceSigMismatch	Error in dynamic signature for binding

tch	device.
UnsupportedOperation	Unsupported operation.
UnsupportedOperation.NotBindVirtualDevice	Unable to bind virtual device.
UnsupportedOperation.DeviceType	Error in device type.
ResourceNotFound	The resource does not exist.
ResourceNotFound.StudioProductNotExist	The product is not created or has been deleted.
ResourceNotFound.DeviceNotExist	The device is not created or has been deleted.
ResourceNotFound.AppNotExists	The application is not created or has been deleted.
UnauthorizedOperation	No permission to operate.
UnauthorizedOperation.APPNoPermissionToStudioProduct	The App has no permission to operate this product.
UnauthorizedOperation.NoPermissionToFamily	No permission to operate this family.
LimitExceeded	Exceed the limit.
LimitExceeded.FamilyDeviceExceedLimit	Device quantity exceeds the limit.

Device Control

User Control Device

Last updated: 2025-04-30 15:35:01

1. API Description

api request domain name: `iot.cloud.tencent.com/api/exploreropen/tokenapi`.

This interface (AppControlDeviceData) is used for users to initiate control operations on bound devices.

2. Input Parameter

Name	Type	Required	Description
AccessToken	String	Yes	Common parameter. AccessToken is used for user authentication of a logged-in user.
RequestId	String	Yes	Common parameter. Unique request ID can be generated manually. It is recommended to use uuid. When locating issues, provide the RequestId for the request.
Action	String	Yes	Common parameter. This interface takes values: AppControlDeviceData.
ProductId	String	Yes	product ID.
DeviceName	String	Yes	Device name.
Data	String	Yes	Control device message.

3. Output Parameter

Name	Type	Description
RequestId	String	Common parameter. Unique request ID can be generated manually. It is recommended to use uuid. When locating issues, provide the RequestId for the request.

Data	String	Return data.
------	--------	--------------

4. Example

Input example

```
POST https://iot.cloud.tencent.com/api/exploreropen/tokenapi HTTP/1.1
content-type: application/json
{
  "RequestId": "req_1",
  "AppId": 1111,
  "Action": "AppControlDeviceData",
  "ProductId": "22F9Y6II7O",
  "DeviceName": "light1",
  "Data": "{\"light_switch\":0}",
  "AccessToken": "8b*****b8ec"
}
```

Output sample: successful

```
{
  "Response": {
    "Data": "",
    "RequestId": "req_1"
  }
}
```

5. Error Code

Error Code	Description
InternalServerError	Internal error.
InvalidParameterValue	Invalid parameter value.
InvalidParameterValue.InvalidAccessToken	The Token is invalid.

Sync Calling Device Behavior

Last updated: 2025-04-30 15:35:15

1. API Description

api request domain name: `iot.cloud.tencent.com/api/exploreropen/tokenapi`.

This interface (AppCallDeviceActionSync) is used for users to initiate synchronous behavior invocation to successfully bound devices.

2. Input Parameter

Name	Type	Required	Description
AccessToken	String	Yes	Common parameter. AccessToken is used for user authentication of a logged-in user.
RequestId	String	Yes	Common parameter. Unique request ID can be generated manually. It is recommended to use uuid. When locating issues, provide the RequestId for the request.
Action	String	Yes	AppCallDeviceActionSync
ProductId	String	Yes	Product ID.
DeviceName	String	Yes	Device name.
ActionId	String	Yes	The identifier of the behavior function, ActionId, is defined in the behavior of the product Thing Model.
InputParams	String	No	Input parameter.

3. Output Parameter

Name	Type	Description
RequestId	String	Common parameter. Unique request ID can be generated manually. It is recommended to use uuid.

		When locating issues, provide the RequestId for the request.
ClientToken	String	Call ID.
Status	String	Return status.
OutputParams	String	Output Parameter. This field may return null, indicating that no valid value was obtained.

4. Samples

Example 1

Input Example

```
POST https://iot.cloud.tencent.com/api/exploreropen/tokenapi HTTP/1.1
content-type: application/json
{
  "AccessToken": "accesstoken*****acc",
  "RequestId": "026fe48b-0c31-4a7c-aaf6-83ba3e9s4bcf",
  "ProductId": "AB*****2345",
  "DeviceName": "DEV",
  "ActionId": "actionid",
  "InputParams": "{\"color\":6}",
  "Action": "AppCallDeviceActionSync"
}
```

Output Example: Success

```
{
  "Response": {
    "ClientToken": "1ee12*****7df703e0d",
    "OutputParams": "",
    "RequestId": "026fe48b-0c31-4a7c-aaf6-83ba3e9s4bcf",
    "Status": "succ"
  }
}
```



5. Error Code

Error Code	Description
InternalServerError	Internal error.
InvalidParameterValue	Invalid parameter value.
InvalidParameterValue.InvalidAccessToke n	The Token is invalid.

Asynchronous Invocation of Device Behavior

Last updated: 2025-04-30 15:35:29

1. API Description

API request domain name: `iot.cloud.tencent.com/api/exploreropen/tokenapi`.

This interface (AppCallDeviceActionAsync) is used to initiate an asynchronous behavior invocation request to the device bound by the user. The returned results of the device behavior called asynchronously can be accessed in real time through persistent connection communication.

2. Input Parameter

Name	Type	Required	Description
AccessToken	String	Yes	Common parameter. AccessToken is used for user authentication of a logged-in user.
RequestId	String	Yes	Common parameter. Unique request ID can be generated manually. It is recommended to use uuid. When locating issues, you need to provide the RequestId for the request.
Action	String	Yes	AppCallDeviceActionAsync
ProductId	String	Yes	product ID.
DeviceName	String	Yes	Device name.
ActionId	String	Yes	Identifier of behavior function.
InputParams	String	No	Input parameter.

3. Output Parameter

Name	Type	Description
------	------	-------------

RequestId	String	Common parameter. Unique request ID can be generated manually. It is recommended to use uuid. When locating issues, you need to provide the RequestId for the request.
ClientToken	String	Call ID.
Status	String	Return status.

4. Example

Example 1

Input example

```
POST https://iot.cloud.tencent.com/api/exploreropen/tokenapi HTTP/1.1
content-type: application/json
{
  "AccessToken": "accesstoken*****acc",
  "RequestId": "026fe48b-0c31-4a7c-aaf6-83ba3e9s4bcf",
  "ProductId": "ABCDE12345",
  "DeviceName": "DEV",
  "ActionId": "actionid",
  "InputParams": "{\"color\":6}",
  "Action": "AppCallDeviceActionAsync"
}
```

Output example: succeed

```
{
  "Response": {
    "ClientToken": "1ee12*****7df703e0d",
    "RequestId": "026fe48b-0c31-4a7c-aaf6-83ba3e9s4bcf",
    "Status": "Sent"
  }
}
```

5. Error Code

Error Code	Description
InternalError	Internal error.
InvalidParameterValue	Invalid parameter value.
InvalidParameterValue.InvalidAccessToken	The Token is invalid.

Device Sharing

Sending Device Sharing From the App

Last updated: 2025-04-30 15:36:02

1. API Description

api request domain name: `iot.cloud.tencent.com/api/exploreropen/tokenapi`.

This interface (AppSendShareDeviceInvite) is used for sending device sharing invitations.

2. Input Parameter

Name	Type	Required	Description
AccessToken	String	Yes	Common parameter. AccessToken is used for user authentication of a logged-in user.
RequestId	String	Yes	Common parameter. Unique request ID can be generated manually. It is recommended to use uuid. When locating issues, you need to provide the RequestId for the request.
Action	String	Yes	Common parameter. This interface takes values: AppSendShareDeviceInvite.
ProductId	String	Yes	product ID.
DeviceName	String	Yes	Device name.
ToUserId	String	Yes	ID of the shared user.

3. Output Parameters

Name	Type	Description
RequestId	String	Common parameter. Unique request ID, same as the input parameter. When locating issues, you need to provide the RequestId for the request.

Status	String	Request result.
--------	--------	-----------------

4. Example

Input Example

```
POST https://iot.cloud.tencent.com/api/exploreropen/tokenapi HTTP/1.1
content-type: application/json
{
  "Action": "AppSendShareDeviceInvite",
  "AccessToken": "8b4a70dd16105f*****18edd4e78a3bb8ec",
  "RequestId": "1555507435215",
  "ProductId": "22F9Y6II7O",
  "DeviceName": "light1",
  "ToUserID": "123456"
}
```

Output Example: Succeed

```
{
  "Response": {
    "RequestId": "1555507435215",
    "Status": "OK"
  }
}
```

Output Example: Fail

```
{
  "Response": {
    "Error": {
      "Code": "InvalidParameterValue.InvalidAccessToken",
      "Message": "Invalid Token"
    },
    "RequestId": "1555507435215"
  }
}
```

5. Error Code

Error Code	Description
InternalError	Internal error.
InvalidParameterValue	Invalid parameter value.
InvalidParameterValue.InvalidAccessToken	The Token is invalid.
InvalidParameterValue.NoPermission	User has no permission for the device.

Retrieve the Device Sharing Token

Last updated: 2025-04-30 15:36:17

1. API Description

api request domain name: `iot.cloud.tencent.com/api/exploreropen/tokenapi`.

This interface (AppCreateShareDeviceToken) is used to obtain device sharing token.

2. Input Parameter

Name	Type	Required	Description
AccessToken	String	Yes	common parameter. AccessToken is used for user authentication of a logged-in user.
RequestId	String	Yes	Common parameter. Unique request ID can be generated manually. It is recommended to use uuid. Locating issues, you need to provide the RequestId for the request.
Action	String	Yes	Common parameter. This interface takes values: AppCreateShareDeviceToken.
FamilyId	String	Yes	Family ID, returned after successful creation of a family
ProductId	String	Yes	Product ID.
DeviceName	String	Yes	Device name.
TokenContext	String	No	The context of sharing.

3. Output Parameter

Name	Type	Description
RequestId	String	Common parameter. Unique request ID, same as the input parameter. Locating issues, you need to provide the RequestId for the request.
ShareDeviceToken	String	Device sharing Token.

4. Example

Input Example

```
POST https://iot.cloud.tencent.com/api/exploreropen/tokenapi HTTP/1.1
content-type: application/json
{
  "Action": "AppCreateShareDeviceToken",
  "AccessToken": "8b4a70dd16105f*****18edd4e78a3bb8ec",
  "RequestId": "155550****15",
  "ProductId": "22F9****70",
  "DeviceName": "light1"
}
```

Output Example: successful

```
{
  "Response": {
    "RequestId": "1555507****15",
    "ShareDeviceToken": "49d2b71dc5814****7c68a4c756d3ce9"
  }
}
```

Output Example: failure

```
{
  "Response": {
    "Error": {
      "Code": "InvalidParameterValue.InvalidAccessToken",
      "Message": "Invalid Token"
    },
    "RequestId": "1555507****15"
  }
}
```

5. Error Code

Error Code	Description
------------	-------------

InternalError	Internal error.
InvalidParameterValue	Invalid parameter value.
InvalidParameterValue.InvalidAccessToken	The Token is invalid.
InvalidParameterValue.NoPermission	User has no permission for the device.

Obtain Device Sharing Token Information

Last updated: 2025-04-30 15:36:36

1. API Description

api request domain name: `iot.cloud.tencent.com/api/exploreropen/tokenapi`.

This interface (AppDescribeShareDeviceToken) is used to obtain token information for device sharing.

2. Input Parameters

Name	Type	Required	Description
AccessToken	String	Yes	Common parameter. AccessToken is used for user authentication of a logged-in user.
RequestId	String	Yes	Common parameter. Unique request ID can be generated manually. It is recommended to use uuid. When locating issues, you need to provide the RequestId for the request.
Action	String	Yes	Common parameter. This interface takes values: AppDescribeShareDeviceToken.
ShareDeviceToken	String	Yes	Device sharing Token.

3. Output Parameters

Name	Type	Description
RequestId	String	Common parameter. Unique request ID, same as the input parameter. When locating issues, you need to provide the RequestId for the request.
ShareDeviceTokenInfo	Object of ShareDeviceTokenInfo	Response result.
AliasName	String	Device alias

IconUrl	String	Icon URL
---------	--------	----------

4. Example

Input example

```
POST https://iot.cloud.tencent.com/api/exploreropen/tokenapi HTTP/1.1
content-type: application/json
{
  "Action": "AppDescribeShareDeviceToken",
  "AccessToken": "8b4a70dd16105f*****18edd4e78a3bb8ec",
  "RequestId": "1555507****15",
  "ShareDeviceToken": "49d2b71dc5814354****8a4c756d3ce9"
}
```

Output example: succeed

```
// UserId, UserNick are empty before being accepted by the sharee.
// After the sharee accepts the sharing, UserId saves the information of
the sharee.
// The frontend can determine the binding status by whether the UserId
is empty.
{
  "Response": {
    "RequestId": "1555507****15",
    "ShareDeviceTokenInfo": {
      "FromUserId": "1",
      "FromUserNick": "tests",
      "UserId": "2",
      "UserNick": "test2",
      "ProductId": "22F9Y6II7O",
      "DeviceName": "light1",
      "AliasName": "my_light",
      "IconUrl": "",
      "BindTime": 0,
      "ExpireTime": 1574757262,
      "CreateTime": 1574152462
    }
  }
}
```

```
}
```

Output example: failure

```
{
  "Response": {
    "Error": {
      "Code": "InvalidParameterValue.InvalidShareDeviceToken",
      "Message": "The device sharing Token is invalid."
    },
    "RequestId": "1555507****15"
  }
}
```

5. Error Code

Error Code	Description
InternalServerError	Internal error.
InvalidParameterValue	Invalid parameter value.
InvalidParameterValue.InvalidShareDeviceToken	The Device sharing Token is invalid.

Devices Shared by Bound Users

Last updated: 2025-04-30 15:36:50

1. API Description

api request domain name: `iot.cloud.tencent.com/api/exploreropen/tokenapi`.

This interface (AppBindUserShareDevice) is used for binding devices shared by users.

2. Input Parameter

Name	Type	Required	Description
AccessToken	String	Yes	Common parameter. AccessToken is used for user authentication of a logged-in user.
RequestId	String	Yes	Common parameter. Unique request ID can be generated manually. It is recommended to use uuid. Locating issues, you need to provide the RequestId for the request.
Action	String	Yes	Common parameter. This interface takes values: AppBindUserShareDevice.
ShareDeviceToken	String	Yes	Token for device sharing.
ProductId	String	Yes	Product ID.
DeviceName	String	Yes	Device name.

3. Output Parameter

Name	Type	Description
RequestId	String	Common parameter. Unique request ID, same as the input parameter. Locating issues, you need to provide the RequestId for the request.
Status	String	Response result.

4. Example

Input example

```
POST https://iot.cloud.tencent.com/api/exploreropen/tokenapi HTTP/1.1
content-type: application/json
{
  "Action": "AppBindUserShareDevice",
  "AccessToken": "8b4a70dd16105f*****18edd4e78a3bb8ec",
  "RequestId": "1555507****15",
  "ProductId": "22F9Y6II7O",
  "DeviceName": "light1",
  "ShareDeviceToken": "0820728466424d8f****422ec06bd398"
}
```

Output example: succeed

```
{
  "Response": {
    "RequestId": "1555507****15",
    "Status": "OK"
  }
}
```

Output example: fail

```
{
  "Response": {
    "Error": {
      "Code": "InvalidParameterValue.DeviceBinded",
      "Message": "The device has been bound to the current user"
    },
    "RequestId": "1555507****15"
  }
}
```

5. Error Code

Error Code	Description
InternalError	Internal error.

InvalidParameterValue	Invalid parameter value.
InvalidParameterValue.DeviecBinded	The device has already been bound to the current user.
InvalidParameterValue.UserBindExceedLimit	User binding sharing device quantity has reached the limit.
InvalidParameterValue.DeviceShareExceededLimit	The maximum number of sharing users for the device has been reached.
InvalidParameterValue.CannotBindOwnDevice	Cannot bind self-sharing devices.
InvalidParameterValue.ShareDeviceTokenUsed	Device sharing Token has been used.
InvalidParameterValue.InvalidShareDeviceToken	Device sharing Token is invalid.

Querying the User-Shared Device List

Last updated: 2025-04-30 15:37:04

1. API Description

api request domain name: `iot.cloud.tencent.com/api/exploreropen/tokenapi`.

This interface (AppListUserShareDevices) is used for querying the device list shared by the user.

2. Input Parameter

Name	Type	Required	Description
AccessToken	String	Yes	Common parameter. AccessToken is used for user authentication of a logged-in user.
RequestId	String	Yes	Common parameter. Unique request ID can be generated manually. It is recommended to use uuid. Locating issues, you need to provide the RequestId for the request.
Action	String	Yes	Common parameter. This interface takes values: AppListUserShareDevices.
Offset	Int	Yes	Pagination offset, starting from 0, maximum not exceeding 500.
Limit	Int	Yes	Pull count per time. Default is 10. Max. 100.

3. Output Parameters

Name	Type	Description
RequestId	String	Common parameter. Unique request ID, same as the input parameter. Locating issues, you need to provide the RequestId for the request.
Total	Int	Total number of returned messages.
ShareDevices	Array of ShareDevices	Respond to device information.

4. Examples

Input example

```
POST https://iot.cloud.tencent.com/api/exploreropen/tokenapi HTTP/1.1
content-type: application/json
{
  "Action": "AppListUserShareDevices",
  "AccessToken": "8b4a70dd16105f*****18edd4e78a3bb8ec",
  "RequestId": "1555507****15",
  "Offset": 0,
  "Limit": 10
}
```

Output example: successful

```
{
  "Response": {
    "RequestId": "1555507****15",
    "ShareDevices": [
      {
        "ProductId": "WTXXX0MF2",
        "DeviceName": "device_XXXX001",
        "DeviceId": "WTXXX0MF2/device_XXXX001",
        "AliasName": "test_device",
        "IconUrl": "https://yyy-xxxx-xxx.cos.ap-
guangzhou.myqcloud.com/113xxxx7a0b273bxxxxc0b9040feeb/1xxxxxxxxx0.png",
        "CreateTime": "2019-12-17T19:39:21+08:00"
      }
    ],
    "Total": 1
  }
}
```

Output example: fail

```
{
  "Response": {
    "Error": {
```

```
    "Code": "InvalidParameterValue.InvalidAccessToken",
    "Message": "Invalid Token"
  },
  "RequestId": "1555507****15"
}
```

5. Error Code

Error Code	Description
InternalServerError	An internal error occurred.
InvalidParameterValue	Invalid parameter value.
InvalidParameterValue.InvalidAccess Token	The Token is invalid.

Deleting User-Shared Devices

Last updated: 2025-04-30 15:37:17

1. API Description

api request domain name: `iot.cloud.tencent.com/api/exploreropen/tokenapi`.

This interface (AppRemoveUserShareDevice) is used to delete the device shared by the user.

2. Input Parameter

Name	Type	Required	Description
AccessToken	String	Yes	Common parameter. AccessToken is used for user authentication of a logged-in user.
RequestId	String	Yes	Common parameter. Unique request ID can be generated manually. It is recommended to use uuid. When locating issues, provide the RequestId for the request.
Action	String	Yes	Common parameter. This interface takes values: AppRemoveUserShareDevice.
ProductId	String	Yes	Product ID.
DeviceName	String	Yes	Device name.

3. Output Parameter

Name	Type	Description
RequestId	String	Common parameter. Unique request ID, same as the input parameter. When locating issues, provide the RequestId for the request.
Status	String	Response result.

4. Example

Input example

```
POST https://iot.cloud.tencent.com/api/exploreropen/tokenapi HTTP/1.1
content-type: application/json
{
  "Action": "AppRemoveUserShareDevice",
  "AccessToken": "8b4a70dd16105f*****18edd4e78a3bb8ec",
  "RequestId": "1555507****15",
  "ProductId": "22F9Y6II7O",
  "DeviceName": "light1"
}
```

Output sample: successful

```
{
  "Response": {
    "RequestId": "1555507****15",
    "Status": "OK"
  }
}
```

Output example: Failure

```
{
  "Response": {
    "Error": {
      "Code": "InvalidParameterValue.AppDeviceNotExist",
      "Message": "No bound devices were found."
    },
    "RequestId": "1555507****15"
  }
}
```

5. Error Code

Error Code	Description
InternalServerError	Internal error.
InvalidParameterValue	Invalid parameter value.

InvalidParameterValue.AppDeviceNot Exist	No bound devices were found.
---	------------------------------

Querying Device User List

Last updated: 2025-04-30 15:37:32

1. API Description

api request domain name: `iot.cloud.tencent.com/api/exploreropen/tokenapi`.

This interface (AppListShareDeviceUsers) is used for querying device user list.

2. Input Parameters

Name	Type	Required	Description
AccessToken	String	Yes	Common parameter. AccessToken is used for user authentication of a logged-in user.
RequestId	String	Yes	Common parameter. Unique request ID can be generated manually. It is recommended to use uuid. Locating issues, you need to provide the RequestId for the request.
Action	String	Yes	Common parameter. This interface takes values: AppListShareDeviceUsers.
ProductId	String	Yes	Product ID.
DeviceName	String	Yes	Device name.
Offset	Int	Yes	Pagination offset, starting from 0, maximum not exceeding 500.
Limit	Int	Yes	Pull count per time. Default is 10. Maximum not exceeding 100.

3. Output Parameters

Name	Type	Description
RequestId	String	Common parameter. Unique request ID, same as the input parameter. Locating issues, you need to provide the RequestId for the request.
Total	Int	Total count of returned data.

Users	Array of ShareUserInfo	List of users.
-------	--	----------------

4. Example

Input Example

```
POST https://iot.cloud.tencent.com/api/exploreropen/tokenapi HTTP/1.1
content-type: application/json
{
  "Action": "AppListShareDeviceUsers",
  "AccessToken": "user2",
  "RequestId": "1555507****15",
  "ProductId": "22F9Y6II7O",
  "DeviceName": "light1",
  "Offset": 0,
  "Limit": 10
}
```

Output Example: Succeed

```
{
  "Response": {
    "RequestId": "1555507****15",
    "Users": [
      {
        "UserId": "1",
        "CountryCode": "86",
        "PhoneNumber": "139****5678",
        "NickName": "tests",
        "Avatar": "",
        "BindTime": 1574153536
      }
    ],
    "Total": 1
  }
}
```

Output Example: Fail

```
{
  "Response": {
    "Error": {
      "Code": "InvalidParameterValue.InvalidAccessToken",
      "Message": "Invalid Token"
    },
    "RequestId": "1555507****15"
  }
}
```

5. Error Code

Error Code	Description
InternalServerError	Internal error.
InvalidParameterValue	Invalid parameter value.
InvalidParameterValue.InvalidAccessTo ken	The Token is invalid.

Deleting the User of Device Sharing

Last updated: 2025-04-30 15:37:47

1. API Description

api request domain name: `iot.cloud.tencent.com/api/exploreropen/tokenapi`.

This interface (AppRemoveShareDeviceUser) is used to delete the user of device sharing.

2. Input Parameter

Name	Type	Required	Description
AccessToken	String	Yes	Common parameter. AccessToken is used for user authentication of a logged-in user.
RequestId	String	Yes	Common parameter. Unique request ID can be generated manually. It is recommended to use uuid. Locating issues, you need to provide the RequestId for the request.
Action	String	Yes	Common parameter. This interface takes values: AppRemoveShareDeviceUser.
RemoveUserId	String	Yes	User ID to be deleted.
ProductId	String	Yes	Product ID.
DeviceName	String	Yes	Device name.

3. Output Parameter

Name	Type	Description
RequestId	String	Common parameter. Unique request ID, same as the input parameter. Locating issues, you need to provide the RequestId for the request.
Status	String	Response result.

4. Example

Input Example

```
POST https://iot.cloud.tencent.com/api/exploreropen/tokenapi HTTP/1.1
content-type: application/json
{
  "Action": "AppRemoveShareDeviceUser",
  "AccessToken": "user2",
  "RequestId": "15555*****15",
  "RemoveUserId": "1234555",
  "ProductId": "22F9Y6II7O",
  "DeviceName": "light1"
}
```

Output sample: successful

```
{
  "Response": {
    "RequestId": "155550*****15",
    "Status": "OK"
  }
}
```

Output example: failure

```
{
  "Response": {
    "Error": {
      "Code": "InvalidParameterValue.AppDeviceNotExist",
      "Message": "No devices were found bound"
    },
    "RequestId": "15555*****15"
  }
}
```

5. Error Code

Error Code	Description
InternalServerError	Internal error.

InvalidParameterValue	Invalid parameter value.
InvalidParameterValue.AppDeviceNot Exist	No bound devices were found.
InvalidParameterValue.NoPermission	The user has no permission to access the device.

Family Management

Creating a Family

Last updated: 2025-04-30 15:38:12

1. API Description

api request domain name: `iot.cloud.tencent.com/api/exploreropen/tokenapi`.

This interface (AppCreateFamily) is used to create a family.

2. Input Parameters

Name	Type	Required	Description
AccessToken	String	Yes	Common parameter. AccessToken is used for user authentication of a logged-in user.
RequestId	String	Yes	Common parameter. Unique request ID can be generated manually. It is recommended to use uuid. Locating issues, you need to provide the RequestId for the request.
Action	String	Yes	Common parameter. This interface takes values: AppCreateFamily.
Name	String	Yes	Family name.
Address	String	Yes	Residence address.

3. Output Parameter

Name	Type	Description
RequestId	String	Common parameter. Unique request ID is the same as the input parameter. Locating issues, you need to provide the RequestId for the request.
FamilyId	String	The Family ID successfully created.

4. Samples

Input Example

```
POST https://iot.cloud.tencent.com/api/exploreropen/tokenapi HTTP/1.1
content-type: application/json
{
  "RequestId": "req_1",
  "Action": "AppCreateFamily",
  "Name": "family_name",
  "Address": "family_address",
  "AccessToken": "c1*****847e"
}
```

Output sample: successful

```
{
  "Response": {
    "Data": {
      "FamilyId": "e0e23a2b33a24652b6*****a1cf"
    },
    "RequestId": "req_1"
  }
}
```

5. Error Code

Error Code	Description
InternalServerError	Internal error.
InvalidParameterValue	Invalid parameter value.
InvalidParameterValue.InvalidAccessToken	The Token is invalid.

Deleting a Family

Last updated: 2025-04-30 15:38:28

1. API Description

api request domain name: `iot.cloud.tencent.com/api/exploreropen/tokenapi`.

This interface (AppDeleteFamily) is used to delete a family.

2. Input Parameters

Name	Type	Required	Description
AccessToken	String	Yes	Common parameter. AccessToken is used for user authentication of a logged-in user.
RequestId	String	Yes	Common parameter. Unique request ID can be generated manually. It is recommended to use uuid. When locating issues, provide the RequestId for the request.
Action	String	Yes	Common parameter. This interface takes values: AppDeleteFamily.
Name	String	Yes	The family name to be deleted.
FamilyId	String	Yes	The Family ID to be deleted.

3. Output Parameter

Name	Type	Description
RequestId	String	Common parameter. Unique request ID, same as the input parameter. When locating issues, provide the RequestId for the request.

4. Samples

Input Example

```
POST https://iot.cloud.tencent.com/api/exploreropen/tokenapi HTTP/1.1
content-type: application/json
{
```

```
"RequestId": "req_1",
"Action": "AppDeleteFamily",
"Name": "XXX's family"
"FamilyId": "e0e23a2b33a24652b6*****a1cf",
"AccessToken": "c1*****847e"
}
```

Output sample: successful

```
{
  "Response": {
    "RequestId": "req_1"
  }
}
```

5. Error Code

Error Code	Description
InternalServerError	Internal error.
InvalidParameterValue	Invalid parameter value.
InvalidParameterValue.InvalidAccessT oken	The Token is invalid.

Modifying Family

Last updated: 2025-04-30 15:38:47

1. API Description

api request domain name: `iot.cloud.tencent.com/api/exploreropen/tokenapi`.

This interface (AppModifyFamily) is used to modify a family.

2. Input Parameter

Name	Type	Required	Description
AccessToken	String	Yes	Common parameter. AccessToken is used for user authentication of a logged-in user.
RequestId	String	Yes	Common parameter. Unique request ID can be generated manually. It is recommended to use uuid. When locating issues, you need to provide the RequestId for the request.
Action	String	Yes	Common parameter. This interface takes values: AppModifyFamily.
Name	String	Yes	The modified family name.
FamilyId	String	Yes	The Family ID that needs to be modified.
Address	String	Yes	The modified residence address.

3. Output Parameter

Name	Type	Description
RequestId	String	Common parameter. Unique request ID, same as the input parameter. When locating issues, you need to provide the RequestId for the request.

4. Samples

Input example

```
POST https://iot.cloud.tencent.com/api/exploreropen/tokenapi HTTP/1.1
content-type: application/json
```

```
{
  "RequestId": "req_1",
  "Action": "AppModifyFamily",
  "Name": "family_name",
  "Address": "family_address",
  "FamilyId": "e0e23a2b33a24652b6*****a1cf",
  "AccessToken": "c1*****847e"
}
```

Output sample: successful

```
{
  "Response": {
    "RequestId": "req_1"
  }
}
```

5. Error Code

Error Code	Description
InternalServerError	An internal error occurred.
InvalidParameterValue	Invalid parameter value.
InvalidParameterValue.InvalidAccessToken	The Token is invalid.

Get Family List

Last updated: 2025-04-30 15:39:05

1. API Description

API request domain name: `iot.cloud.tencent.com/api/exploreropen/tokenapi`.

This interface (AppGetFamilyList) is used to obtain the family list.

2. Input Parameters

Name	Type	Required	Description
AccessToken	String	Yes	Common parameter. AccessToken is used for user authentication of a logged-in user.
RequestId	String	Yes	Common parameter. Unique request ID can be generated manually. It is recommended to use uuid. Locating issues, provide the RequestId for the request.
Action	String	Yes	Common parameter. This interface takes values: AppGetFamilyList.
Offset	Int	No	The offset of the data to be queried.
Limit	Int	No	The return limit is a maximum of 50 items.

3. Output Parameter

Name	Type	Description
RequestId	String	Common parameters, unique request ID, consistent with the input parameters.
Total	Int	Total number of returned households.
FamilyList	Array	The array of list information for the returned families. Please see FamilyList .

4. Example

Input example


```
POST https://iot.cloud.tencent.com/api/exploreropen/tokenapi HTTP/1.1
content-type: application/json

{
  "RequestId": "req_1",
  "Action": "AppGetFamilyList",
  "Offset":0,
  "Limit":2,
  "AccessToken": "c1*****847e"
}
```

Output sample: successful

```
{
  "Response": {
    "RequestId": "req_1",
    "FamilyList": [{
      "FamilyId": "a1c6939b39d345b897*****a12c",
      "Name": "family_name",
      "Role": 0, // 1:Administrator 0:Ordinary member
      "CreateTime": 1570786578,
      "UpdateTime": 1570790807
    }],
    "Total":1
  }
}
```

5. Error Code

Error Code	Description
InternalServerError	Internal error.
InvalidParameterValue	Invalid parameter value.
InvalidParameterValue.InvalidAccessToken	The Token is invalid.

Getting Family Details

Last updated: 2025-04-30 15:39:51

1. API Description

api request domain name: `iot.cloud.tencent.com/api/exploreropen/tokenapi`.

This interface (AppDescribeFamily) is used to obtain family details.

2. Input Parameters

Name	Type	Required	Description
AccessToken	String	Yes	Common parameter. AccessToken is used for user authentication of a logged-in user.
RequestId	String	Yes	Common parameter. Unique request ID can be generated manually. It is recommended to use uuid. Locating issues requires providing the RequestId for the request.
Action	String	Yes	Common parameter. This interface takes values: AppDescribeFamily.
FamilyId	String	Yes	The Family ID for which details are to be queried.

3. Output Parameters

Name	Type	Description
RequestId	String	Common parameter. Unique request ID is the same as the input parameter. Locating issues requires providing the RequestId for the request.
FamilyId	String	Family ID.
FamilyName	String	Family name.
Address	String	Residence address.
CreateTime	Int64	Creation time, UNIX timestamp (in seconds).

UpdateTime	Int64	Last update time, UNIX timestamp (in seconds).
------------	-------	--

4. Samples

Input Example

```
POST https://iot.cloud.tencent.com/api/exploreropen/tokenapi HTTP/1.1
content-type: application/json
{
  "RequestId": "req_1",
  "Action": "AppDescribeFamily",
  "FamilyId": "a1c6939b39d345b897*****a12c",
  "AccessToken": "c1*****847e"
}
```

Output sample: successful

```
{
  "Response": {
    "RequestId": "req_1",
    "Data": {
      "FamilyId": "a1c6939b39d345b897*****a12c",
      "FamilyName": "family_name",
      "Address": "family_address",
      "CreateTime": 1570786578,
      "UpdateTime": 1570790807
    }
  }
}
```

5. Error Code

Error Code	Description
InternalError	Internal error.
InvalidParameterValue	Invalid parameter value.
InvalidParameterValue.InvalidAccessTo	The Token is invalid.

ken	
-----	--

Creating a Room

Last updated: 2025-04-30 15:40:05

1. API Description

API request domain name: `iot.cloud.tencent.com/api/exploreropen/tokenapi`.

This interface (AppCreateRoom) is used to create a room.

2. Input Parameters

Name	Type	Required	Description
AccessToken	String	Yes	Common parameter. AccessToken is used for user authentication of a logged-in user.
RequestId	String	Yes	Common parameter. Unique request ID can be generated manually. It is recommended to use uuid. When locating issues, provide the RequestId for the request.
Action	String	Yes	Common parameter. This interface takes values: AppCreateRoom.
FamilyId	String	Yes	Family ID that the created room belongs to.
Name	String	Yes	Name of the created room.

3. Output Parameters

Name	Type	Description
RequestId	String	Common parameter. Unique request ID is the same as the input parameter. When locating issues, provide the RequestId for the request.
RoomId	String	Room ID successfully created.

4. Samples

Input Example

```
POST https://iot.cloud.tencent.com/api/exploreropen/tokenapi HTTP/1.1
content-type: application/json
```

```
{
  "RequestId": "req_1",
  "Action": "AppCreateRoom",
  "Name": "name",
  "FamilyId": "family_address",
  "AccessToken": "c1*****847e"
}
```

Output Example: Success

```
{
  "Response": {
    "Data": {
      "RoomId": 123456
    },
    "RequestId": "req_1"
  }
}
```

5. Error Code

Error Code	Description
InternalServerError	Internal error.
InvalidParameterValue	Invalid parameter value.
InvalidParameterValue.InvalidAccessToken	The Token is invalid.

Modifying Room

Last updated: 2025-04-30 15:40:23

1. API Description

api request domain name: `iot.cloud.tencent.com/api/exploreropen/tokenapi`.

This interface (AppModifyRoom) is used to modify a room.

2. Input Parameters

Name	Type	Required	Description
AccessToken	String	Yes	Common parameter. AccessToken is used for user authentication of a logged-in user.
RequestId	String	Yes	Common parameter. Unique request ID can be generated manually. It is recommended to use uuid. Locating issues, you need to provide the RequestId for the request.
Action	String	Yes	Common parameter. This interface takes values: AppModifyRoom.
FamilyId	String	Yes	Family ID that the room to be modified belongs to.
Name	String	No	Room name.
RoomId	String	Yes	Room ID that needs to be modified.

3. Output Parameter

Name	Type	Description
RequestId	String	Common parameter. Unique request ID, consistent with the input parameter. Locating issues, you need to provide the RequestId for the request.

4. Samples

Input Example

```
POST https://iot.cloud.tencent.com/api/exploreropen/tokenapi HTTP/1.1
content-type: application/json

{
  "RequestId": "req_1",
  "Action": "AppModifyRoom",
  "FamilyId": "fa070f59a2fe4****43815310c794fdf",
  "RoomId": "5",
  "Name": "TestName",
  "AccessToken":
"8b4a70dd16105f*****18edd4e78a3bb8ec"
}
```

Output sample: successful

```
{
  "Response": {
    "RequestId": "req_1"
  }
}
```

5. Error Code

Error Code	Description
InternalError	Internal error.
InvalidParameterValue	Invalid parameter value.
InvalidParameterValue.InvalidAccessToken	The Token is invalid.

Deleting a Room

Last updated: 2025-04-30 15:40:39

1. API Description

api request domain name: `iot.cloud.tencent.com/api/exploreropen/tokenapi`.

This interface (AppDeleteRoom) is used to delete a room.

2. Input Parameters

Name	Type	Required	Description
AccessToken	String	Yes	Common parameter. AccessToken is used for user authentication of a logged-in user.
RequestId	String	Yes	Common parameter. Unique request ID can be generated manually. It is recommended to use uuid. Locating issues requires providing the RequestId for the request.
Action	String	Yes	Common parameter. This interface takes values: AppDeleteRoom.
FamilyId	String	Yes	Family ID that the room to be deleted belongs to.
RoomId	String	Yes	Room ID to be deleted.

3. Output Parameter

Name	Type	Description
RequestId	String	Common parameter. Unique request ID, same as the input parameter. Locating issues requires providing the RequestId for the request.

4. Samples

Input example

```
POST https://iot.cloud.tencent.com/api/exploreropen/tokenapi HTTP/1.1
content-type: application/json
{
```

```
"RequestId": "req_1",
"Action": "AppDeleteRoom",
"FamilyId": "e0e23a2b33a24652b6*****a1cf",
"RoomId": "123456",
"AccessToken": "c1*****847e"
}
```

Output example: successful

```
{
  "Response": {
    "RequestId": "req_1"
  }
}
```

5. Error Code

Error Code	Description
InternalServerError	Internal error.
InvalidParameterValue	Invalid parameter value.
InvalidParameterValue.InvalidAccessToken	The Token is invalid.

Getting Room List

Last updated: 2025-04-30 15:40:53

1. API Description

api request domain name: `iot.cloud.tencent.com/api/exploreropen/tokenapi`.

This interface (AppGetRoomList) is used to obtain room list.

2. Input Parameters

Name	Type	Required	Description
AccessToken	String	Yes	Common parameter. AccessToken is used for user authentication of a logged-in user.
RequestId	String	Yes	Common parameter. Unique request ID can be generated manually. It is recommended to use uuid. Locating issues, you need to provide the RequestId for the request.
Action	String	Yes	Common parameter. This interface takes values: AppGetRoomList.
Offset	Int	No	The offset of the queried data.
Limit	Int	No	The total limit you want to query. Maximum returns 50.
FamilyId	String	No	Family ID needed to get.

3. Output Parameters

Name	Type	Description
RequestId	String	Common parameter. Unique request ID, same as the input parameter. Locating issues, you need to provide the RequestId for the request.
Total	Int	Total number of returned devices
RoomList	Array	The information array of the returned family list. Please see Roomlist .

4. Example

Input Example

```
POST https://iot.cloud.tencent.com/api/exploreropen/tokenapi HTTP/1.1
content-type: application/json
{
  "RequestId": "req_1",
  "Action": "AppGetRoomList",
  "FamilyId": "fa070f59a2fe4f1da4*****4fdf",
  "Offset": 0,
  "Limit": 2,
  "AccessToken": "xxx"
}
```

Output example: successful

```
{
  "Response": {
    "RequestId": "req_1",
    "RoomList": [{
      "RoomId": "123456",
      "RoomName": "name",
      "DeviceNum": 2,
      "CreateTime": 1570786578,
      "UpdateTime": 1570790807
    }],
    "Total": 1
  }
}
```

5. Error Code

Error Code	Description
InternalServerError	Internal error.
InvalidParameterValue	Invalid parameter value.
InvalidParameterValue.InvalidAccessT	Token invalid

oken	
------	--

Invite Family Members

Last updated: 2025-04-30 15:41:07

1. API Description

api request domain name: `iot.cloud.tencent.com/api/exploreropen/tokenapi`.

This interface (ApplInviteMember) is used to send a request to a WeChat friend to inviting others to join family.

2. Input Parameters

Name	Type	Required	Description
AccessToken	String	Yes	Common parameter. AccessToken is used for user authentication of a logged-in user.
RequestId	String	Yes	Common parameter. Unique request ID can be generated manually. It is recommended to use uuid. When locating issues, you need to provide the RequestId for the request.
Action	String	Yes	Common parameter. This interface takes values: ApplInviteMember.
FamilyId	String	Yes	Family ID.

3. Output Parameters

Name	Type	Description
RequestId	String	Common parameter. Unique request ID, same as the input parameter. When locating issues, you need to provide the RequestId for the request.
ShareToken	String	Share the Token. The invitee can use this Token to make an API call to the Member Join Family API to join the family.

4. Example

Input Example

```
POST https://iot.cloud.tencent.com/api/exploreropen/tokenapi HTTP/1.1
content-type: application/json
{
  "RequestId": "req_1",
  "Action": "AppInviteMember",
  "FamilyId": "e0e23a2b33a24652b6*****a1cf",
  "AccessToken": "c1*****847e"
}
```

Output sample: successful

```
{
  "Response": {
    "ShareToken": "e0*****a1cf",
    "RequestId": "req_1"
  }
}
```

5. Error Code

Error Code	Description
InternalError	Internal error.
InvalidParameterValue	Invalid parameter value.
InvalidParameterValue.InvalidAccessToken	The Token is invalid.

Member Joining Family

Last updated: 2025-04-30 15:41:21

1. API Description

api request domain name: `iot.cloud.tencent.com/api/exploreropen/tokenapi`.

This interface (AppJoinFamily) is used for member join family.

2. Input Parameter

Name	Type	Required	Description
AccessToken	String	Yes	Common parameter. AccessToken is used for user authentication of a logged-in user.
RequestId	String	Yes	Common parameter. Unique request ID can be generated manually. It is recommended to use uuid. When locating issues, you need to provide the RequestId for the request.
Action	String	Yes	Common parameter. This interface takes values: AppJoinFamily.
ShareToken	String	Yes	Share Token, generated by the family administrator calling Invite Family Member . The invitee can join the appropriate family with this Share Token.

3. Output Parameters

Name	Type	Description
RequestId	String	Common parameter. Unique request ID, same as the input parameter. When locating issues, you need to provide the RequestId for the request.

4. Example

Input Example

```
POST https://iot.cloud.tencent.com/api/exploreropen/tokenapi HTTP/1.1
content-type: application/json
```



```
{
  "RequestId": "550e8400-e29b-41d4-a716-446655440000",
  "Action": "AppJoinFamily",
  "ShareToken": "e0*****a1cf",
  "AccessToken": "c1*****847e"
}
```

Output Example: successful

```
{
  "Response": {
    "RequestId": "550e8400-e29b-41d4-a716-446655440000"
  }
}
```

5. Error Code

Error Code	Description
InternalServerError	Internal error.
InvalidParameterValue	Invalid parameter value.
InvalidParameterValue.InvalidAccessToken	The Token is invalid.

Deleting a Family Member

Last updated: 2025-04-30 15:41:34

1. API Description

api request domain name: `iot.cloud.tencent.com/api/exploreropen/tokenapi`.

This interface (AppDeleteFamilyMember) is used to delete family members by administrators.

2. Input Parameters

Name	Type	Required	Description
AccessToken	String	Yes	Common parameter. AccessToken is used for user authentication of a logged-in user.
RequestId	String	Yes	Common parameter. Unique request ID can be generated manually. It is recommended to use uuid. When locating issues, you need to provide the RequestId for the request.
Action	String	Yes	Common parameter. This interface takes values: AppDeleteFamilyMember.
FamilyId	String	No	The Family ID from which to remove the member.
MemberId	String	Yes	The member ID to be removed.

3. Output Parameter

Name	Type	Description
RequestId	String	Common parameter. Unique request ID is the same as the input parameter. When locating issues, you need to provide the RequestId for the request.

4. Samples

Input Example

```
POST https://iot.cloud.tencent.com/api/exploreropen/tokenapi HTTP/1.1
content-type: application/json
```

```
{
  "RequestId": "550e8400-e29b-41d4-a716-446655440000",
  "Action": "AppDeleteFamilyMember",
  "FamilyId": "f_abcd****abcd",
  "MemberId": "12345678",
  "AccessToken": "c1*****847e"
}
```

Output sample: successful

```
{
  "Response": {
    "RequestId": "550e8400-e29b-41d4-a716-446655440000"
  }
}
```

5. Error Code

Error Code	Description
InternalError	Internal error.
InvalidParameterValue	Invalid parameter value.
InvalidParameterValue.InvalidAccessToken	The Token is invalid.

Member Quit Family

Last updated: 2025-04-30 15:41:50

1. API Description

API request domain name: `iot.cloud.tencent.com/api/exploreropen/tokenapi`.

This interface (AppExitFamily) is used for members to proactively exit a family.

2. Input Parameter

Name	Type	Required	Description
AccessToken	String	Yes	Common parameter. AccessToken is used for user authentication of a logged-in user.
RequestId	String	Yes	Common parameter. Unique request ID can be generated manually. It is recommended to use uuid. Locating issues, you need to provide the RequestId for the request.
Action	String	Yes	Common parameter. This interface takes values: AppExitFamily.
FamilyId	String	Yes	Family ID to exit.

3. Output Parameter

Name	Type	Description
RequestId	String	Common parameter. Unique request ID, same as the input parameter. Locating issues, you need to provide the RequestId for the request.

4. Example

Input Example

```
POST https://iot.cloud.tencent.com/api/exploreropen/tokenapi HTTP/1.1
content-type: application/json
{
  "RequestId": "550e8400-e29b-41d4-a716-44665*****00",
  "Action": "AppExitFamily",
```

```
"FamilyId": "e0e23a2b33a24652b6*****a1cf",  
"AccessToken": "c1*****847e"  
}
```

Output example: success

```
{  
  "Response": {  
    "RequestId": "550e8400-e29b-41d4-a716-44665*****00"  
  }  
}
```

5. Error Code

Error Code	Description
InternalServerError	Internal error.
InvalidParameterValue	Invalid parameter value.
InvalidParameterValue.InvalidAccessToken	The Token is invalid.

Get the Family Member List

Last updated: 2025-04-30 15:42:06

1. API Description

api request domain name: `iot.cloud.tencent.com/api/exploreropen/tokenapi`.

This interface (AppGetFamilyMemberList) is used to obtain the family member list.

2. Input Parameter

Name	Type	Required	Description
AccessToken	String	Yes	Common parameter. AccessToken is used for user authentication of a logged-in user.
RequestId	String	Yes	Common parameter. Unique request ID can be generated manually. It is recommended to use uuid. Locating issues, you need to provide the RequestId for the request.
Action	String	Yes	Common parameter. this interface takes values: AppGetFamilyMemberList.
Offset	Int	No	The offset of the data you want to query.
Limit	Int	No	The total limit of data you want to query. Maximum returns 50 items.
FamilyId	String	Yes	Family ID needed to get.

3. Output Parameters

Name	Type	Description
RequestId	String	Common parameter. Unique request ID, same as the input parameter. Locating issues, you need to provide the RequestId for the request.
Total	Int	Total number of family members returned.
MemberList	Array	Array of family members returned. Please see MemberList .

4. Samples

Input Example

```
POST https://iot.cloud.tencent.com/api/exploreropen/tokenapi HTTP/1.1
content-type: application/json
{
  "FamilyId": "f_b3ac7e6b1cd24392b41e*****5a",
  "Offset": 0,
  "Limit": 100,
  "Action": "AppGetFamilyMemberList",
  "RequestId": "hojkE0Hx88",
  "AccessToken": "3c*****40c5"
}
```

Output example: successful

```
{
  "MemberList": [{
    "UserID": "7267",
    "NickName": "deny",
    "Avatar":
      "https://wx.qlogo.cn/mmopen/vi_32/V2kIBBE35phMrFKu5sra16bzpUuPtokyWTbMRP
      t7bRGBjx4fEqQrZxoDYyxNVbtkXMosuxQeIqiafIZJ8ADr8cw/132",
    "Role": 1
  }],
  "Total": 1,
  "RequestId": "hojkE0Hx88"
}
```

5. Error Code

Error Code	Description
InternalError	Internal error.
InvalidParameterValue	Invalid parameter value.
InvalidParameterValue.InvalidAccessToken	The Token is invalid.

Invite Family Members on the App

Last updated: 2025-04-30 15:42:24

1. API Description

api request domain name: `iot.cloud.tencent.com/api/exploreropen/tokenapi`.

This interface (AppSendShareFamilyInvite) is used to send share invitations to family.

2. Input Parameter

Name	Type	Required	Description
AccessToken	String	Yes	Common parameter. AccessToken is used for user authentication of a logged-in user.
RequestId	String	Yes	Common parameter. Unique request ID can be generated manually. It is recommended to use uuid. Locating issues, you need to provide the RequestId for the request.
Action	String	Yes	Common parameter. This interface takes values: AppSendShareFamilyInvite
FamilyId	String	Yes	Family ID
ToUserId	String	Yes	ID of the shared user

3. Output Parameter

Name	Type	Description
RequestId	String	Common parameter. Unique request ID, same as the input parameter. Locating issues, you need to provide the RequestId for the request.

4. Samples

Example 1

Input Example

```
POST https://iot.cloud.tencent.com/api/exploreropen/tokenapi HTTP/1.1
```

```
content-type: application/json

{
  "Action": "AppSendShareFamilyInvite",
  "AccessToken": "8b4a70dd16105f*****18edd4e78a3bb8ec",
  "RequestId": "1555*****5215",
  "FamilyId": "e0e23a2b33a24652b6*****a1cf",
  "ToUserID": "123456"
}
```

Output example: successful

```
{
  "Response": {
    "RequestId": "1555*****5215"
  }
}
```

Output example: Failure

```
{
  "Response": {
    "Error": {
      "Code": "InvalidParameterValue.InvalidAccessToken",
      "Message": "Invalid Token"
    },
    "RequestId": "1555*****5215"
  }
}
```

5. Error Code

Error Code	Description
InternalServerError	Internal error.
InvalidParameterValue	Invalid parameter value.
InvalidParameterValue.InvalidAccessToken	The Token is invalid.
InvalidParameterValue.NoPermission	The user has no permission for this

family.

Device Scheduling

Creating Scheduled Tasks

Last updated: 2025-04-30 15:42:49

1. API Description

api request domain name: `iot.cloud.tencent.com/api/exploreropen/tokenapi`.

this interface (AppCreateTimer) is used for users to create scheduled tasks for devices.

2. Input Parameter

Name	Type	Required	Description
AccessToken	String	Yes	Common parameter. AccessToken is used for user authentication of a logged-in user.
RequestId	String	Yes	Common parameter. Unique request ID can be generated manually. It is recommended to use uuid. Locating issues, you need to provide the RequestId for the request.
Action	String	Yes	Common parameter. This interface takes values AppCreateTimer.
ProductId	String	Yes	Associated product ID of the created scheduled task.
DeviceName	String	Yes	Device name controlled by the created scheduled task.
TimerName	String	Yes	Name of the timer.
Days	String	Yes	Timer activation time, each digit, 0: Off, 1: On, from left to right sequentially represent: Sunday, Monday, Tuesday, Wednesday, Thursday, Friday, Saturday.
TimePoint	String	Yes	Timer start time.
Repeat	int	Yes	Whether to loop. 0: Not required.

- Requirement

Data	String	Yes	Control message sent when the timer starts.
------	--------	-----	---

3. Output Parameter

Name	Type	Description
RequestId	String	Common parameter. Unique request ID, same as the input parameter. Locating issues, you need to provide the RequestId for the request.
TimerId	String	Timer ID.
TimerName	String	Timer name.

4. Samples

Input Example

```
POST https://iot.cloud.tencent.com/api/exploreropen/tokenapi HTTP/1.1
content-type: application/json
{
  "RequestId": "req_1",
  "Action": "AppCreateTimer",
  "ProductId": "US4*****IK",
  "DeviceName": "411_3",
  "AccessToken": "c1*****847e",
  "TimerName": "timer_test",
  "Days": "1000000", // Each 0: Off, 1: On, from left to right in
sequence represent: Sunday Monday Tuesday Wednesday Thursday Friday
Saturday
  "TimePoint": "06:30",
  "Repeat": 1, //Whether required to loop 0: Not required 1: Required
  "Data": "{\"brightness\": 25}"
}
```

Output sample: successful

```
{
  "Response": {
```

```
    "Data": {
      "TimerId": "e0e23a2b33a24652b6*****a1cf",
      "TimerName": "timer_test"
    },
    "RequestId": "req_1"
  }
}
```

5. Error Code

Error Code	Description
InternalServerError	Internal error.
InvalidParameterValue	Invalid parameter value.
InvalidParameterValue.InvalidAccessToken	The Token is invalid.

Modifying a Scheduled Task

Last updated: 2025-04-30 15:43:04

1. API Description

api request domain name: `iot.cloud.tencent.com/api/exploreropen/tokenapi`.

This interface (AppModifyTimer) is used to modify the scheduled task of a device.

2. Input Parameters

Name	Type	Required	Description
AccessToken	String	Yes	Common parameter. AccessToken is used for user authentication of a logged-in user.
RequestId	String	Yes	Common parameter. Unique request ID can be generated manually. It is recommended to use uuid. When locating issues, provide the RequestId for the request.
Action	String	Yes	Common parameter. This interface takes values AppModifyTimer.
ProductId	String	Yes	Modify the product ID corresponding to the timer.
DeviceName	String	Yes	Device name corresponding to the timer.
TimerName	String	Yes	Timer name.
TimerId	String	Yes	Timer ID.
TimePoint	String	Yes	Timer start time.
Days	String	Yes	Timer activation time, each digit, 0: Off, 1: On, from left to right in sequence represent: Sunday, Monday, Tuesday, Wednesday, Thursday, Friday, Saturday.
Repeat	Int	Yes	Whether to loop. 0: Not required.

- Requirement

Data	String	Yes	Control message sent when the timer starts.
------	--------	-----	---

3. Output Parameters

Name	Type	Description
RequestId	String	Common parameter. Unique request ID, same as the input parameter. When locating issues, provide the RequestId for the request.

4. Samples

Input Example

```
POST https://iot.cloud.tencent.com/api/exploreropen/tokenapi HTTP/1.1
content-type: application/json
{
  "RequestId": "req_1",
  "Action": "AppModifyTimer",
  "ProductId": "US*****DIK",
  "DeviceName": "411_3",
  "AccessToken": "c1*****847e",
  "TimerId": "a1c6939b39d345b897*****a12c",
  "TimerName": "timer_test_modify",
  "Days": "1100000",
  "TimePoint": "9:30",
  "Repeat": 0,
  "Data": "{\"brightness\": 28}"
}
```

Output sample: successful

```
{
  "Response": {
    "RequestId": "req_1"
  }
}
```


5. Error Code

Error Code	Description
InternalError	Internal error.
InvalidParameterValue	Invalid parameter value.
InvalidParameterValue.InvalidAccessToken	The Token is invalid.

Deleting Scheduled Tasks

Last updated: 2025-04-30 15:43:19

1. API Description

api request domain name: `iot.cloud.tencent.com/api/exploreropen/tokenapi`.

This interface (AppDeleteTimer) is used to delete device scheduling.

2. Input Parameter

Name	Type	Required	Description
AccessToken	String	Yes	Common parameter. AccessToken is used for user authentication of a logged-in user.
RequestId	String	Yes	Common parameter. Unique request ID can be generated manually. It is recommended to use uuid. When locating issues, you need to provide the RequestId for the request.
Action	String	Yes	Common parameter. This interface takes values AppDeleteTimer.
ProductId	String	Yes	Modify the product ID corresponding to the timer.
DeviceName	String	Yes	Device name corresponding to the timer.
TimerId	String	Yes	Timer ID.

3. Output Parameter

Name	Type	Description
RequestId	String	Common parameter. Unique request ID is the same as the input parameter. When locating issues, you need to provide the RequestId for the request.

4. Example

Input Example

```
POST https://iot.cloud.tencent.com/api/exploreropen/tokenapi HTTP/1.1
content-type: application/json
{
  "RequestId": "req_1",
  "Action": "AppDeleteTimer",
  "ProductId": "US4*****IK",
  "DeviceName": "411_3",
  "TimerId": "e0e23a2b33a24652b6*****a1cf",
  "AccessToken": "c1*****847e"
}
```

Output sample: successful

```
{
  "Response": {
    "RequestId": "req_1"
  }
}
```

5. Error Code

Error Code	Description
InternalServerError	Internal error.
InvalidParameterValue	Invalid parameter value.
InvalidParameterValue.InvalidAccessToken	The Token is invalid.

Modifying Scheduled Task Status

Last updated: 2025-04-30 15:43:33

1. API Description

api request domain name: `iot.cloud.tencent.com/api/exploreropen/tokenapi`.

This interface (AppModifyTimerStatus) is used to modify the start-stop status of a scheduled task.

2. Input Parameters

Name	Type	Required	Description
AccessToken	String	Yes	Common parameter. AccessToken is used for user authentication of a logged-in user.
RequestId	String	Yes	Common parameter. Unique request ID can be generated manually. It is recommended to use uuid. When locating issues, you need to provide the RequestId for the request.
Action	String	Yes	Common parameter. This interface takes values AppModifyTimerStatus.
ProductId	String	Yes	The product ID of the scheduled task list needs to be modified.
DeviceName	String	Yes	The timer status or device name needs modification.
TimerId	String	Yes	Timer ID.
Status	Int	No	The timer status that needs to be modified. • 0: Off. • 1: Enable.

3. Output Parameters

Name	Type	Description
RequestId	String	Common parameter. Unique request ID, consistent with the input parameter. When locating issues, you need to provide the RequestId for the request.

4. Example

Input Example

```
POST https://iot.cloud.tencent.com/api/exploreropen/tokenapi HTTP/1.1
content-type: application/json
{
  "RequestId": "req_1",
  "Action": "AppModifyTimerStatus",
  "ProductId": "US4*****IK",
  "DeviceName": "411_3",
  "TimerId": "a1c6939b39d345b897*****a12c",
  "Status": 0,
  "AccessToken": "c1*****847e"
}
```

Output example: successful

```
{
  "Response": {
    "RequestId": "req_1"
  }
}
```

5. Error Code

Error Code	Description
InternalError	Internal error.
InvalidParameterValue	Invalid parameter value.
InvalidParameterValue.InvalidAccessToken	The Token is invalid.

Get Scheduled Task List

Last updated: 2025-04-30 15:43:49

1. API Description

api request domain name: `iot.cloud.tencent.com/api/exploreropen/tokenapi`.

This interface (AppGetTimerList) is used to obtain the timer list.

2. Input Parameter

Name	Type	Required	Description
AccessToken	String	Yes	Common parameter. AccessToken is used for user authentication of a logged-in user.
RequestId	String	Yes	Common parameter. Unique request ID can be generated manually. It is recommended to use uuid. When locating issues, provide the RequestId for the request.
Action	String	Yes	Common parameter. This interface takes values AppGetTimerList.
ProductId	String	Yes	Product ID needed to get the scheduled task list.
DeviceName	String	Yes	Device name needed to get the scheduled task list.
Offset	Int	No	The list offset obtained.
Limit	Int	No	The total obtained message count, with a maximum of 50.

3. Output Parameter

Name	Type	Description
RequestId	String	Common parameter. Unique request ID, same as the input parameter. When locating issues, provide the RequestId for the request.
TimerList	Array	The obtained product list. Please see TimerList .

Total	Int	Total number of items owned.
-------	-----	------------------------------

4. Samples

Input Example

```
POST https://iot.cloud.tencent.com/api/exploreropen/tokenapi HTTP/1.1
content-type: application/json
{
  "RequestId": "req_1",
  "Action": "AppGetTimerList",
  "ProductId": "US4*****IK",
  "DeviceName": "411_3",
  "Offset": 0,
  "Limit": 2,
  "AccessToken": "c1*****847e"
}
```

Output sample: successful

```
{
  "Response": {
    "RequestId": "req_1",
    "Total": 2,
    "TimerList": [{
      "TimerId": "a1c6939b39d345b897*****a12c",
      "TimerName": "timer_test_modify",
      "ProductId": "US4*****IK",
      "DeviceName": "411_3",
      "Days": "1100000",
      "TimePoint": "09:30",
      "Repeat": 0,
      "Data": "{\"brightness\": 28}",
      "Status": 0,
      "CreateTime": 1570786578,
      "UpdateTime": 1570790807
    }, {
      "TimerId": "f888dd20c70744939a*****1c3a",
      "TimerName": "timer_test",

```

```
      "ProductId": "US4*****IK",
      "DeviceName": "411_3",
      "Days": "1000000",
      "TimePoint": "07:00",
      "Repeat": 1,
      "Data": "{\\"brightness\\": 25}",
      "Status": 1,
      "CreateTime": 1570786483,
      "UpdateTime": 1570786483
    }
  ]
}
```

5. Error Code

Error Code	Description
InternalServerError	Internal error.
InvalidParameterValue	Invalid parameter value.
InvalidParameterValue.InvalidAccessToken	The Token is invalid.

Firmware Upgrade

Reporting Device Firmware Version

Last updated: 2025-04-30 15:44:15

1. API Description

api request domain name: `iot.cloud.tencent.com/api/exploreropen/tokenapi`.

This interface (AppReportFirmwareVersion) is used for submitting device firmware version.

2. Input Parameters

Name	Type	Required	Description
Access Token	String	Yes	Common parameter. AccessToken is used for user authentication of a logged-in user.
RequestId	String	Yes	Common parameter. Unique request ID can be generated manually. It is recommended to use uuid. Locating issues, you need to provide the RequestId for the request.
Action	String	Yes	Common parameter. This interface takes values: AppReportFirmwareVersion.
ProductId	String	Yes	Product ID.
Device Name	String	Yes	Device name.
Version	String	Yes	Firmware version reported by the device.

3. Output Parameters

Name	Type	Description
RequestId	String	Common parameter. Unique request ID, same as the input parameter. Locating issues, you need to provide the RequestId for the request.

4. Samples

Example 1

Input example

```
POST https://iot.cloud.tencent.com/api/exploreropen/tokenapi HTTP/1.1
content-type: application/json
{
  "RequestId": "584406e9-bf59-46a3-8d45-39f1891bed7b",
  "Action": "AppReportFirmwareVersion",
  "Version": "1.0",
  "ProductId": "PR*****ID",
  "DeviceName": "de***01",
  "AccessToken": "871d2d42*****692afae0"
}
```

Output example: successful

```
{
  "Response": {
    "RequestId": "584406e9-bf59-46a3-8d45-39f1891bed7b"
  }
}
```

5. Error Code

Error Code	Description
InternalServerError	Internal error.
InvalidParameterValue	Invalid parameter value.
InvalidParameterValue.InvalidAccessToken	The Token is invalid.
InvalidParameterValue.NoPermission	The user has no permission for the device.

Query Device Upgradable Firmware Version

Last updated: 2025-04-30 15:44:31

1. API Description

api request domain name: `iot.cloud.tencent.com/api/exploreropen/tokenapi`.

This interface (AppCheckFirmwareUpdate) is used for querying the firmware version that can be upgraded on the device.

2. Input Parameters

Name	Type	Required	Description
AccessToken	String	Yes	Common parameter. AccessToken is used for user authentication of a logged-in user.
RequestId	String	Yes	Common parameter. Unique request ID can be generated manually. It is recommended to use uuid. Locating issues, you need to provide the RequestId for the request.
Action	String	Yes	Common parameter. This interface takes values: AppCheckFirmwareUpdate.
ProductId	String	Yes	Product ID.
DeviceName	String	Yes	Device name.

3. Output Parameters

Name	Type	Description
RequestId	String	Common parameter. Unique request ID is the same as the input parameter. Locating issues, you need to provide the RequestId for the request.
CurrentVersion	String	Current firmware version of the device.
DstVersion	String	Firmware can be upgraded.

4. Example

Sample 1

Input example

```
POST https://iot.cloud.tencent.com/api/exploreropen/tokenapi HTTP/1.1
content-type: application/json
{
  "RequestId": "584406e9-bf59-46a3-8d45-39f1891bed7b",
  "Action": "AppCheckFirmwareUpdate",
  "ProductId": "PR*****ID",
  "DeviceName": "de***01",
  "AccessToken": "871d2d42*****692afae0"
}
```

Output example: succeed

```
{
  "Response": {
    "DstVersion": "1.1",
    "CurrentVersion": "1.0",
    "RequestId": "584406e9-bf59-46a3-8d45-39f1891bed7b"
  }
}
```

5. Error Code

Error Code	Description
InternalServerError	Internal error.
InvalidParameterValue	Invalid parameter value.
InvalidParameterValue.InvalidAccessToken	The Token is invalid.
InvalidParameterValue.NoPermission	User has no permission for the device.
FailedOperation	Operation failed.

FailedOperation.DeviceInfoOutdated	Device firmware version error (the firmware version reported by the device does not match the current firmware upgrade task).
------------------------------------	---

Query the Download Address of Device Firmware

Last updated: 2025-04-30 15:44:46

1. API Description

api request domain name: `iot.cloud.tencent.com/api/exploreropen/tokenapi`.

This interface (AppGetDeviceOTAInfo) is used for querying the firmware download address of a device.

2. Input Parameter

Name	Type	Required	Description
Access Token	String	Yes	Common parameter. AccessToken is used for user authentication of a logged-in user.
RequestId	String	Yes	Common parameter. Unique request ID can be generated manually. It is recommended to use uuid. Locating issues, you need to provide the RequestId for the request.
Action	String	Yes	Common parameter. This interface takes values: AppGetDeviceOTAInfo.
ProductId	String	Yes	product ID.
Device Name	String	Yes	device name.

3. Output Parameter

Name	Type	Description
RequestId	String	Common parameter. Unique request ID, same as the input parameter. Locating issues, you need to provide the RequestId for the request.
FirmwareURL	String	Firmware Download Address.

TargetVersion	String	Target Firmware Version.
UploadVersion	String	Firmware version reported by the device.

4. Example

Example 1

Input Example

```
POST https://iot.cloud.tencent.com/api/exploreropen/tokenapi HTTP/1.1
content-type: application/json
{
  "AccessToken": "871d2d42*****692afae0",
  "Action": "AppGetDeviceOTAInfo",
  "RequestId": "584406e9-bf59-46a3-8d45-39f1891bed7b",
  "ProductId": "PR*****ID",
  "DeviceName": "de***01",
}
```

Output sample: successful

```
{
  "Response": {
    "FirmwareURL": "https://examplebucket-12500000000.cos.ap-
guangzhou.myqcloud.com/path/to/firmware?sign=***",
    "RequestId": "584406e9-bf59-46a3-8d45-39f1891bed7b",
    "TargetVersion": "1.1",
    "UploadVersion": "1.0"
  }
}
```

5. Error Code

Error Code	Description
InternalError	Internal error.
InvalidParameterValue	Invalid parameter value.

InvalidParameterValue.InvalidAccessToken	The Token is invalid.
InvalidParameterValue.NoPermission	The user has no permission for the device.
ResourceNotFound	The resource does not exist.
ResourceNotFound.DeviceFirmwareTaskNotExist	Failed to query firmware upgrade tasks.

Confirm Firmware Upgrade Task

Last updated: 2025-04-30 15:45:09

1. API Description

api request domain name: `iot.cloud.tencent.com/api/exploreropen/tokenapi`.

This interface (AppPublishFirmwareUpdateMessage) is used for the cloud to initiate a firmware upgrade request to the device after the user confirms the upgrade.

2. Input Parameter

Name	Type	Required	Description
AccessToken	String	Yes	Common parameter. AccessToken is used for user authentication of a logged-in user.
RequestId	String	Yes	Common parameter. Unique request ID can be generated manually. It is recommended to use uuid. When locating issues, provide the RequestId for the request.
Action	String	Yes	Common parameter. This interface takes values: AppPublishFirmwareUpdateMessage.
ProductId	String	Yes	Product ID.
DeviceName	String	Yes	Device name.

3. Output Parameters

Name	Type	Description
RequestId	String	Common parameter. Unique request ID is the same as the input parameter. When locating issues, provide the RequestId for the request.

4. Example

Example 1

Input Example

```
POST https://iot.cloud.tencent.com/api/exploreropen/tokenapi HTTP/1.1
content-type: application/json
{
  "RequestId": "584406e9-bf59-46a3-8d45-39f1891bed7b",
  "Action": "AppPublishFirmwareUpdateMessage",
  "ProductId": "PR*****ID",
  "DeviceName": "de***01",
  "AccessToken": "871d2d42*****692afae0"
}
```

Output Example: Succeed

```
{
  "Response": {
    "RequestId": "584406e9-bf59-46a3-8d45-39f1891bed7b"
  }
}
```

5. Error Code

Error Code	Description
InternalError	Internal error.
InvalidParameterValue	Invalid parameter value.
InvalidParameterValue.InvalidAccessToken	The Token is invalid.
InvalidParameterValue.NoPermission	The user has no permission to access the device.
ResourceNotFound	The resource does not exist.
ResourceNotFound.DeviceHasNoFirmware	The device has no firmware version.
FailedOperation	Operation failed.
FailedOperation.DeviceOffline	The device is offline.

Reporting Firmware Upgrade Task Status

Last updated: 2025-04-30 15:45:41

1. API Description

api request domain name: `iot.cloud.tencent.com/api/exploreropen/tokenapi`.

This interface (AppReportOTAStatus) is used to report device firmware upgrade status and progress.

2. Input Parameters

Name	Type	Required	Description
Access Token	String	Yes	Common parameter. AccessToken is used for user authentication of a logged-in user.
RequestId	String	Yes	Common parameter. Unique request ID can be generated manually. It is recommended to use uuid. Locating issues, you need to provide the RequestId for the request.
Action	String	Yes	Common parameter. This interface takes values: AppReportOTAStatus.
ProductId	String	Yes	Product ID.
Device Name	String	Yes	Device name.
Version	String	Yes	Target upgrade version of the firmware.
State	String	Yes	Firmware Upgrade Status: • downloading • FAIL: Upgrade failed. • Upgrade successful. • burning • updating

Result Code	Int	Yes	Error Code: • 0: Succeed. • -1: Download Timeout. • -2: File does not exist. • -3: Signature expired. • -4: MD5 mismatch. • -5: Firmware update failed.
Result Msg	String	No	Error additional information, required when State value is fail.
Persent	Int	No	Upgrade progress percentage, in the range of 0 ~ 100.

3. Output Parameters

Name	Type	Description
RequestId	String	Common parameter. Unique request ID, same as the input parameter. Locating issues, you need to provide the RequestId for the request.

4. Example

Sample 1

Input example

```
POST https://iot.cloud.tencent.com/api/exploreropen/tokenapi HTTP/1.1
content-type: application/json
{
  "AccessToken": "871d2d42*****692afae0",
  "Action": "AppReportOTAStatus",
  "RequestId": "584406e9-bf59-46a3-8d45-39f1891bed7b",
  "ProductId": "PR*****ID",
  "DeviceName": "de***01",
  "State": "done",
  "ResultCode": 0,
  "ResultMsg": "",
  "Version": "1.1",
  "Persent": 100
}
```

Output example: succeed

```
{
  "Response": {
    "RequestId": "584406e9-bf59-46a3-8d45-39f1891bed7b"
  }
}
```

Example 2

Input example

```
POST https://iot.cloud.tencent.com/api/exploreropen/tokenapi HTTP/1.1
content-type: application/json
{
  "AccessToken": "871d2d42*****692afae0",
  "Action": "AppReportOTAStatus",
  "RequestId": "584406e9-bf59-46a3-8d45-39f1891bed7b",
  "ProductId": "PR*****ID",
  "DeviceName": "de***01",
  "State": "downloading",
  "ResultCode": 0,
  "ResultMsg": "",
  "Version": "2.0",
  "Persent": 20
}
```

Output example: successful

```
{
  "Response": {
    "RequestId": "584406e9-bf59-46a3-8d45-39f1891bed7b"
  }
}
```

5. Error Code

Error Code	Description
------------	-------------

InternalError	Internal error.
InvalidParameterValue	Invalid parameter value.
InvalidParameterValue.InvalidAccessToken	The Token is invalid.
InvalidParameterValue.NoPermission	User has no permission for the device.

Querying Device Firmware Upgrade Status

Last updated: 2025-04-30 15:45:55

1. API Description

api request domain name: `iot.cloud.tencent.com/api/exploreropen/tokenapi`.

This interface (AppDescribeFirmwareUpdateStatus) is used for querying the device firmware upgrade status and progress.

2. Input Parameter

Name	Type	Required	Description
AccessToken	String	Yes	Common parameter. AccessToken is used for user authentication of a logged-in user.
RequestId	String	Yes	Common parameter. Unique request ID can be generated manually. It is recommended to use uuid. When locating issues, you need to provide the RequestId for the request.
Action	String	Yes	Common parameter. This interface takes values: AppDescribeFirmwareUpdateStatus.
ProductId	String	Yes	product ID.
DeviceName	String	Yes	Device name.

3. Output Parameter

Name	Type	Description
RequestId	String	Common parameter. Unique request ID is the same as the input parameter. When locating issues, you need to provide the RequestId for the request.
OriginVersion	String	Source version of the upgrade task.

DstVersion	String	Target version of the upgrade task.
ErrMsg	String	Error information.
Percent	Int64	Upgrade progress percentage.
Status	Int64	Upgrade Status: • Device is offline. • 1: Pending processing. • 2: Message delivery successful. • 3: Downloading. • 4: Burning. • 5: Failed. • 6: Update completed. • 7: Is being processed. • 8: Wait for user confirmation. • 20: Download completed.

4. Example

Input Example

```
POST https://iot.cloud.tencent.com/api/exploreropen/tokenapi HTTP/1.1
content-type: application/json
{
  "RequestId": "f92406b3-5a9a-4fe8-bc43-45e3d794bb68",
  "Action": "AppDescribeFirmwareUpdateStatus",
  "ProductId": "22*****0",
  "DeviceName": "li****1",
  "AccessToken": "8b4d16105f*****18edd4e78a3bb8ec"
}
```

Output Example: Succeed

```
{
  "Response": {
    "DstVersion": "2.0",
    "ErrMsg": "",
    "OriVersion": "1.0",
```



```
    "Percent": 0,
    "RequestId": "f92406b3-5a9a-4fe8-bc43-45e3d794bb68"
  }
}
```

5. Error Code

Error Code	Description
InternalServerError	Internal error.
InvalidParameterValue	Invalid parameter value.
InvalidParameterValue.InvalidAccessToken	The Token is invalid.

View Firmware Information

Last updated: 2025-04-30 15:46:09

1. API Description

api request domain name: `iot.cloud.tencent.com/api/exploreropen/tokenapi`.

This interface (AppDescribeFirmware) is used for querying device firmware information.

2. Input Parameters

Name	Type	Required	Description
AccessToken	String	Yes	Common parameter. AccessToken is used for user authentication of a logged-in user.
RequestId	String	Yes	Common parameter. Unique request ID can be generated manually. It is recommended to use uuid. When locating issues, you need to provide the RequestId for the request.
Action	String	Yes	Common parameter. This interface takes values: AppDescribeFirmware.
ProductID	String	Yes	product ID
FirmwareVersion	String	Yes	Firmware Version Number

3. Output Parameter

Name	Type	Description
RequestId	String	Common parameter. Unique request ID, same as the input parameter. When locating issues, you need to provide the RequestId for the request.
ProductId	String	product ID
Name	String	Firmware name. Note: This field may return null, indicating that no valid value was obtained.
Description	String	Firmware description. Note: This field may return null, indicating that no valid

		value was obtained.
Md5sum	String	Firmware Md5 value. Note: This field may return null, indicating that no valid value was obtained.
Createtime	Integer	Unix second-level timestamp for firmware upload. Note: This field may return null, indicating that no valid value was obtained.
ProductName	String	Product name.
FwType	String	Firmware upgrade module. Note: This field may return null, indicating that no valid value was obtained.
Version	String	Firmware Version Number

4. Samples

Input example

```
POST https://iot.cloud.tencent.com/api/exploreropen/tokenapi HTTP/1.1
content-type: application/json

{
  "RequestId": "f92406b3-5a9a-4fe8-bc43-45e3d794bb68",
  "Action": "AppDescribeFirmware",
  "ProductId": "22*****0",
  "FirmwareVersion": "t.2",
  "AccessToken": "8b4d16105f*****18edd4e78a3bb8ec"
}
```

Output example: successful

```
{
  "Response": {
    "Createtime": 1599626765,
    "Description": "ttttt",
    "Md5sum": "1a5c386576074d22a604b795e8917e1a",
    "Name": "test",
    "ProductId": "I6KTCZ170U",
    "ProductName": "8bvsn6go_Test OTA Upgrade"
  }
}
```

```
"RequestId": "01365e8c-f025-40b9-97a4-fa583d6c569e",
"Version": "t.2",
"FwType": "mcu"
}
}
```

5. Error Code

Error Code	Description
InternalServerError.InternalRPCError	Internal RPC error.
ResourceNotFound.FirmwareNotExist	Firmware does not exist.
ResourceNotFound.StudioProductNotExist	The product does not exist.

Message Management

Obtain Message List

Last updated: 2025-04-30 15:46:34

1. API Description

api request domain name: `iot.cloud.tencent.com/api/exploreropen/tokenapi`.

This interface (AppGetMessages) is used to query message list, which includes device alarm messages and operation notification messages.

2. Input Parameter

Name	Type	Required	Description
AccessToken	String	Yes	Common parameter. AccessToken is used for user authentication of a logged-in user.
RequestId	String	Yes	Common parameter. Unique request ID can be generated manually. It is recommended to use uuid. RequestId is required for locating a problem.
Action	String	Yes	Common parameter. This interface takes values: AppGetMessages.
Category	Int64	Yes	Primary type. 1: Device. 2: family. 3: notification.
MsgID	String	No	Message ID, optional for the first time.
MsgTimestamp	Int64	No	Timestamp of the message, optional for the first time or can be 0.
Limit	Int64	Yes	The maximum return limit, no more than 100.

3. Output Parameters

Name	Type	Description
------	------	-------------

RequestId	String	Common parameter. Unique request ID, same as the input parameter. Provide the RequestId for the request when locating issues.
Listover	String	For pagination.

4. Samples

Input Example

```
POST https://iot.cloud.tencent.com/api/exploreropen/tokenapi HTTP/1.1
content-type: application/json
{
  "Action": "AppGetMessages",
  "RequestId": "f92406b3-5a9a-4fe8-bc43-45e3d794bb68",
  "AccessToken": "8b*****b8ec",
  "Category": 2,
  "MsgID": "",
  "MsgTimestamp": 0,
  "Limit": 5
}
```

Output Example: Success

```
{
  "Response": {
    "Data": {
      "Listover": true,
      "Msgs": [{
        "UserID": "1",
        "FromUserID": "",
        "MsgID": "65167410520395776",
        "Category": 1,
        "MsgType": 100,
        "MsgTitle": "Device Warning",
        "MsgContent": "Some message Wed Nov 27 2019 11:52:01
GMT+0000.",
        "MsgTimestamp": 1574855521420,
        "ProductId": "",
        "DeviceName": ""
      }
    ]
  }
}
```

```
    "DeviceAlias": "",
    "FamilyId": "",
    "FamilyName": "",
    "RelateUserID": "",
    "CreateAt": "2019-11-27T11:52:01.42Z"
  }, {
    "UserID": "1",
    "FromUserID": "1",
    "MsgID": "65167395785805824",
    "Category": 2,
    "MsgType": 200,
    "MsgTitle": "Member addition result notification"
    "MsgContent": "\"jordan\" adds \"dylan\" as a member of
\"Home in Shenzhen\". Members can view and control smart devices at
home. Wed Nov 27 2019 11:51:57 GMT+0000"
    "MsgTimestamp": 1574855517907,
    "ProductId": "",
    "DeviceName": "",
    "DeviceAlias": "",
    "FamilyId": "1",
    "FamilyName": "test",
    "RelateUserID": "2",
    "CreateAt": "2019-11-27T11:51:57.907Z"
  }, {
    "UserID": "1",
    "FromUserID": "1",
    "MsgID": "65167386239569920",
    "Category": 2,
    "MsgType": 201,
    "MsgTitle": "Member deletion result notification"
    "MsgContent": "\"jordan\" removed \"dylan\" from \"the home
in shenzhen\". Once deleted, he/she cannot view or control the smart
devices in the home. Wed Nov 27 2019 11:51:54 GMT+0000"
    "MsgTimestamp": 1574855515631,
    "ProductId": "",
    "DeviceName": "",
    "DeviceAlias": "",
    "FamilyId": "1",
    "FamilyName": "test",
    "RelateUserID": "2",
```

```
        "CreateAt": "2019-11-27T11:51:55.631Z"
      }, {
        "UserID": "1",
        "FromUserID": "1",
        "MsgID": "65167339519217664",
        "Category": 3,
        "MsgType": 300,
        "MsgTitle": "Device sharing result notification"
        "MsgContent": "\"dylan\" has successfully received the
device \"smart light\" shared by \"jordan\". Now it can view and control the
device. Wed Nov 27 2019 11:51:44 GMT+0000"
        "MsgTimestamp": 1574855504492,
        "ProductId": "22F9Y6II7O",
        "DeviceName": "light1",
        "DeviceAlias": "light1",
        "FamilyId": "",
        "FamilyName": "",
        "RelateUserID": "2",
        "CreateAt": "2019-11-27T11:51:44.492Z"
      }]
    },
    "RequestId": "f92406b3-5a9a-4fe8-bc43-45e3d794bb68"
  }
}
```

5. Error Code

Error Code	Description
InternalError	Internal error.
InvalidParameterValue	Invalid parameter value.
InvalidParameterValue.InvalidAccessToken	The Token is invalid.

Deleting Messages

Last updated: 2025-04-30 15:46:49

1. API Description

api request domain name: `iot.cloud.tencent.com/api/exploreropen/tokenapi`.

This interface (AppDeleteMessage) is used to delete messages.

2. Input Parameter

Name	Type	Required	Description
AccessToken	String	Yes	Common parameter. AccessToken is used for user authentication of a logged-in user.
RequestId	String	Yes	Common parameter. Unique request ID can be generated manually. It is recommended to use uuid. When locating issues, you need to provide the RequestId for the request.
Action	String	Yes	Common parameter. This interface takes values: AppDeleteMessage.
MsgID	String	Yes	Message ID.

3. Output Parameter

Name	Type	Description
RequestId	String	Common parameter. Unique request ID, same as the input parameter. When locating issues, you need to provide the RequestId for the request.
Data	String	Return data.

4. Example

Input Example

```
POST https://iot.cloud.tencent.com/api/exploreropen/tokenapi HTTP/1.1
content-type: application/json
{
```

```
"Action": "AppDeleteMessage",
"RequestId": "f92406b3-5a9a-4fe8-bc43-45e3d794bb68",
"AccessToken": "8b*****b8ec",
"MsgID": 1633328834809856
}
```

Output example: successful

```
{
  "Response": {
    "Data": "OK",
    "RequestId": "f92406b3-5a9a-4fe8-bc43-45e3d794bb68"
  }
}
```

Output example: failure

```
{
  "Response": {
    "Error": {
      "Code": "InvalidParameterValue.InvalidAccessToken",
      "Message": "Invalid Token"
    },
    "RequestId": ""
  }
}
```

5. Error Code

Error Code	Description
InternalServerError	Internal error.
InvalidParameterValue	Invalid parameter value.
InvalidParameterValue.InvalidAccessToken	The Token is invalid.
InvalidParameterValue.MsgIDInvalid	Illegal message ID.

Manual Intelligence

Create Manual Intelligent Linkage

Last updated: 2025-04-30 15:47:14

1. API Description

api request domain name: `iot.cloud.tencent.com/api/exploreropen/appapi`.

This interface (AppCreateScene) is used to create manual intelligent linkage.

2. Input Parameter

Name	Type	Required	Description
AccessToken	String	Yes	Common parameter, the Access Token obtained upon successful login by users through WeChat ID, mobile phone or email account.
RequestId	String	Yes	Common parameter, unique request ID, can be generated manually, recommended to use uuid. Locating issues, need to provide the RequestId for the request.
Action	String	Yes	Common parameter, this interface takes values: AppCreateScene.
FamilyId	String	Yes	Family ID that the manual intelligent linkage belongs to.
SceneName	String	Yes	Name of the manual intelligent linkage.
SceneIcon	String	Yes	Background image address of the manual intelligent linkage.
Actions	Array of Action	Yes	Manual intelligent linkage action list.

3. Output Parameter

Name	Type	Description
------	------	-------------

Request Id	String	Common parameters, unique request ID, same as the input parameters. Provide the RequestId for the request when locating issues.
Sceneld	String	Manual intelligent linkage ID.

4. Example

Input Example

```
POST https://iot.cloud.tencent.com/api/exploreropen/tokenapi HTTP/1.1
content-type: application/json
{
  "Actions": [
    {
      "ActionType": 0,
      "ProductId": "R320***0EU",
      "DeviceName": "df***yY",
      "Data": "{\"brightness\": 25}"
    }
  ],
  "AccessToken": "x***v2",
  "RequestId": "re***_1",
  "Action": "AppCreateScene",
  "FamilyId": "f_9b30*****c3d9588fcc1",
  "SceneName": "name",
  "SceneIcon": "icon"
}
```

Output Sample: succeed

```
{
  "Response": {
    "SceneId": "a_cd61ddbe****47d21d6aeffea",
    "RequestId": "req_1"
  }
}
```

5. Error Code

Error Code	Description
InternalError	Internal error.
InternalError.InternalServerErrorDB	DB error.
InvalidParameterValue	Invalid parameter value.
InvalidParameterValue.InvalidAccessToken	The Token is invalid.
InvalidParameterValue.NoPermission	User has no permission for the device.

Get Manual Intelligent Linkage List

Last updated: 2025-04-30 15:47:30

1. API Description

api request domain name: `iot.cloud.tencent.com/api/exploreropen/appapi`.

This interface (AppGetSceneList) is used to obtain the Manual Intelligent Linkage List.

2. Input Parameter

Name	Type	Required	Description
AccessToken	String	Yes	Common parameter, the Access Token obtained upon successful login by users through WeChat ID, mobile phone or email account.
RequestId	String	Yes	Common parameter, unique request ID, can be generated manually, uuid is recommended for use. Locating issues, need to provide the RequestId for the request.
Action	String	Yes	Common parameter, this interface takes values: AppGetSceneList.
FamilyId	String	Yes	Family ID.
Offset	Int	Yes	Message offset.
Limit	Int	Yes	Maximum number of return messages, with a maximum value of 50.

3. Output Parameter

Name	Type	Description
RequestId	String	Common parameter, unique request ID, same as the input parameter. Locating issues, need to provide the RequestId for the request.
SceneList	Array of Scene	Manual Intelligent Linkage List.
Total	Int	Total number of manual intelligent linkages.

4. Samples

Input Example

```
POST https://iot.cloud.tencent.com/api/exploreropen/tokenapi HTTP/1.1
content-type: application/json
{
  "AccessToken": "***v2",
  "RequestId": "req_1",
  "Action": "AppGetSceneList",
  "FamilyId": "f_9b309*****4c3d9588fcc1",
  "Offset": 0,
  "Limit": 10
}
```

Output example: successful

```
{
  "Response": {
    "RequestId": "req_1",
    "SceneList": [
      {
        "SceneId": "s_f7feb440d*****28f3a598d00c",
        "FamilyId": "f_9b309d8*****a11b4c3d9588fcc1",
        "SceneName": "f***na",
        "SceneIcon": "",
        "Actions": [          // Each field in the Actions array; it does
not appear if empty
          {
            "ActionType": 0,
            "ProductId": "R32****0EU",
            "DeviceName": "df***yY",
            "Data": "{\"brightness\": 25}",
            "AliasName": "1***45"      // Description: AliasName is a
device alias.
          },
          {
            "ActionType": 1,
            "Data": "100"
          }
        ]
      }
    ]
  }
}
```

```
    }
  ],
  "UserId": "1",
  "CreateTime": 1578557536,
  "UpdateTime": 1578560598,
  "Flag": 0, // Flag: 0 for normal scenarios, 1: device anomalies
in scenarios, not in the home
  "Status": 0 // Running state of the scenario: 0 for not running,
1 for running
}
],
"Total": 1
}
}
// IconUrl is the url of the specific device.
```

5. Error Code

Error Code	Description
InternalServerError	Internal error.
InternalServerError.InternalServerErrorDB	DB error.
InvalidParameterValue	Invalid parameter value.
InvalidParameterValue.InvalidAccessToken	The Token is invalid.
InvalidParameterValue.NoPermission	User has no permission for the device.
UnsupportedOperation.SceneRunInProgress	Scenario in execution.

Deleting Manual Intelligent Linkage

Last updated: 2025-04-30 15:47:43

1. API Description

api request domain name: `iot.cloud.tencent.com/api/exploreropen/appapi`.

This interface (AppDeleteScene) is used to delete manual intelligent linkage.

2. Input Parameter

Name	Type	Required	Description
AccessToken	String	Yes	Common parameter, the Access Token obtained after users through WeChat ID, mobile phone or email account login succeed.
RequestId	String	Yes	Common parameter, unique request ID, can be generated manually. It is recommended to use uuid. When locating issues, you need to provide the RequestId for the request.
Action	String	Yes	Common parameter, this interface takes values: AppDeleteScene.
SceneId	String	Yes	Manual intelligent linkage ID.

3. Output Parameter

Name	Type	Description
RequestId	String	Common parameter, unique request ID, same as the input parameter. When locating issues, you need to provide the RequestId for the request.

4. Example

Input example

```
POST https://iot.cloud.tencent.com/api/exploreropen/tokenapi HTTP/1.1
content-type: application/json
{
  "AccessToken": "***v2",
```

```
"RequestId": "req_1",  
"SceneId": "s_527cb52*****b453cb6a46f653",  
"Action": "AppDeleteScene"  
}
```

Output sample: successful

```
{"Response":{"RequestId":"req_1"}}
```

5. Error Code

Error Code	Description
InternalServerError	Internal Error
InternalServerError.InternalServerErrorExceptionDB	DB error.
InvalidParameterValue	Invalid parameter value.
InvalidParameterValue.InvalidAccessToken	The Token is invalid.
InvalidParameterValue.NoPermission	User has no permission for the device.
UnsupportedOperation.SceneRunInProgress	Scenario in execution.

Modifying Manual Intelligent Linkage

Last updated: 2025-04-30 15:47:59

1. API Description

api request domain name: `iot.cloud.tencent.com/api/exploreropen/appapi`.

This interface (AppModifyScene) is used to modify manual intelligent linkage.

2. Input Parameters

Name	Type	Required	Description
AccessToken	String	Yes	Common parameters, the Access Token obtained by users after successfully logging in via WeChat ID, mobile phone or email account.
RequestId	String	Yes	Common parameters, unique request ID, can be generated manually, uuid is recommended for use. Locating issues, need to provide the RequestId for the request.
Action	String	Yes	Common parameter, this interface takes values: AppModifyScene.
SceneId	String	Yes	Manual intelligent linkage ID.
SceneName	String	Yes	Name of the manual intelligent linkage.
SceneIcon	String	Yes	Background image address of the manual intelligent linkage.
Actions	Array of Action	Yes	Manual intelligent linkage action list.

3. Output Parameter

Name	Type	Description
RequestId	String	Common parameters, unique request ID, same as the input parameters. Locating issues, need to provide the RequestId for the request.

4. Example

Input Example

```
POST https://iot.cloud.tencent.com/api/exploreropen/tokenapi HTTP/1.1
content-type: application/json
{
  "Actions": [
    {
      "ActionType": 0,
      "ProductId": "R32****0EU",
      "DeviceName": "df****yY",
      "Data": "{\"brightness\": 25}"
    }
  ],
  "AccessToken": "***v2",
  "RequestId": "req_1",
  "Action": "AppModifyScene",
  "SceneId": "s_527cb5*****3cb6a46f653",
  "SceneName": "name",
  "SceneIcon": "***icon"
}
```

Output example: successful

```
{"Response":{"RequestId":"req_1"}}
```

5. Error Code

Error Code	Description
InternalServerError	Internal error.
InternalServerError.InternalServerErrorExceptionDB	DB error.
InvalidParameterValue	Invalid parameter value.
InvalidParameterValue.InvalidAccessToken	The Token is invalid.
InvalidParameterValue.NoPermission	The user has no permission for the device.

UnsupportedOperation.SceneRunInProgress	Scenario in execution.
---	------------------------

Performing Manual Intelligent Linkage

Last updated: 2025-04-30 15:48:13

1. API Description

api request domain name: `iot.cloud.tencent.com/api/exploreropen/appapi`.

This interface (AppRunScene) is for execution of manual intelligent linkage.

2. Input Parameters

Name	Type	Required	Description
AccessToken	String	Yes	Common parameter, the Access Token obtained upon successful login by users through WeChat ID, mobile phone or email account.
RequestId	String	Yes	Common parameter, unique request ID, can be generated manually, recommended for use as uuid. Locating issues, need to provide the RequestId for the request.
Action	String	Yes	Common parameter, this interface takes values: AppRunScene.
SceneId	String	Yes	Manual intelligent linkage ID.

3. Output Parameter

Name	Type	Description
RequestId	String	Common parameter, unique request ID, same as the input parameter. Locating issues, need to provide the RequestId for the request.

4. Example

Input Example

```
POST https://iot.cloud.tencent.com/api/exploreropen/tokenapi HTTP/1.1
content-type: application/json
```

```
{
  "AccessToken": "***v2",
  "RequestId": "req_1",
  "SceneId": "s_527cb5*****53cb6a46f653",
  "Action": "AppRunScene"
}
```

Output example: successful

```
{"Response":{"RequestId":"req_1"}}
```

5. Error Code

Error Code	Description
InternalServerError	Internal error.
InternalServerError.InternalServerErrorExceptionDB	DB error.
InvalidParameterValue	Invalid parameter value.
InvalidParameterValue.InvalidAccessToken	The Token is invalid.
InvalidParameterValue.NoPermission	User has no permission for the device.
UnsupportedOperation.SceneRunInProgress	Scenario in execution.

Automatic Intelligence

Create Automatic Intelligent Linkage

Last updated: 2025-04-30 15:48:37

1. API Description

api request domain name: `iot.cloud.tencent.com/api/exploreropen/appapi`.

This interface (AppCreateAutomation) is used to create automatic intelligent linkage.

2. Input Parameter

Name	Type	Required	Description
AccessToken	String	Yes	Common parameters, Access Token obtained upon successful login by users through WeChat ID, mobile phone or email account.
RequestId	String	Yes	Common parameters, unique request ID, can be generated manually. It is recommended to use uuid. When locating issues, provide the RequestId for the request.
Action	String	Yes	Common parameters, this interface takes values: AppCreateAutomation.
FamilyId	String	Yes	Family ID that the automatic intelligent linkage belongs to.
Icon	String	Yes	Background image address of the automatic intelligent linkage.
Name	String	Yes	Name of the automatic intelligent linkage.
Status	int	Yes	Automatic intelligent linkage switch state. • 0: Disable. • 1: Enable.
MatchType	int	Yes	Condition match type. • 0: All conditions are met. • 1: One condition is met.

Conditions	Array of Condition	Yes	Automatic intelligent linkage trigger conditions.
Actions	Array of Action	Yes	Action list of automatic intelligent linkage.
EffectiveBeginTime	String	Yes	Start of effective time.
EffectiveEndTime	String	Yes	Expiration time.
EffectiveDays	String	Yes	The effective date is a 7-digit number composed of 0s and 1s. 0 means not execute, and 1 means execute. The first digit represents Sunday, and the following digits sequentially represent Monday to Saturday.

3. Output Parameter

Name	Type	Description
RequestId	String	Common parameters, unique request ID, same as the input parameters. When locating issues, provide the RequestId for the request.
AutomationId	String	Automatic intelligent linkage ID.

4. Example

Input Example

```
POST https://iot.cloud.tencent.com/api/exploreropen/tokenapi HTTP/1.1
content-type: application/json
{
  "RequestId": "req_1",
  "Action": "AppCreateAutomation",
  "FamilyId": "f_9****c1",
  "Icon": "https://www.icon.com/",
  "Name": "autoName",
  "Status": 0, //Integration switch: 0: Disabled; 1: Enabled.
```

```

    "MatchType": 1, //Condition match type: 0: All conditions AND; 1:
All conditions OR
    "EffectiveBeginTime": "00:00", // Two new configuration parameters
to indicate start and end times
    "EffectiveEndTime": "23:59",
    "EffectiveDays": "1111111", // A 7-digit number composed of 0s and
1s, where 0 means not execute and 1 means execute. The first digit
represents Sunday, followed by Monday to Saturday in sequence.
    "Conditions": [
        {
            "CondId": "abc", // Condition ID, ensure uniqueness within a
linkage just
            "CondType": 0, //Condition type: 0 - Device attribute value
condition; 1 - Scheduling condition
            "Property": {
                "ProductId": "N****4",
                "DeviceName": "g***ev",
                "AliasName": "devName",
                "IconUrl": "productIcon",
                "PropertyId": "switch", //Attribute ID of the device
                "Op": "equal", //conditional operator eq: equal ne:
not equal gt: more than lt: less than ge: larger than le: less than
or equal to
                "Value": 1 //Comparative Value
            }
        },
        {
            "CondId": "efg",
            "CondType": 1,
            "Timer": {
                "Days": "1000000", // A 7-digit number composed of 0s
and 1s. 0 means not execute, and 1 means execute. The first digit
represents Sunday, and the following digits represent Monday to Saturday
sequentially.
                "TimePoint": "09:30" // Trigger time, 24-hour format,
such as "14:00"
            }
        }
    ],
    "Actions": [

```

```
{
    "ActionType": 0, // 0: Device action, specific parameter set
to Data 0: Device action 1: Delay 2: Scenario 3: Notification
    "ProductId": "R32***0EU",
    "DeviceName": "df***yY",
    "Data": "{\"brightness\": 25}", // ActionType is 0: Device
action parameter, 1: Delay in seconds, 2: Execution scene Id, 3:
Notification content
    "AliasName": "12345",
    "IconUrl": "url"
},
{
    "ActionType": 1, // 1: Delay action, specific delay time is
Data, value in seconds
    "Data": "100"
}
],
"AccessToken": "***v2"
}
```

Output sample: successful

```
{
  "Response": {
    "AutomationId": "a_cd61dd*****d21d6aeffea",
    "RequestId": "req_1"
  }
}
```

5. Error Code

Error Code	Description
InternalServerError	Internal error.
InternalServerError.InternalServerErrorDB	DB error.
InvalidParameterValue	Invalid parameter value.

InvalidParameterValue.InvalidAccessToken	The Token is invalid.
InvalidParameterValue.NoPermission	The user has no permission to the device.
LimitExceeded	The quantity exceeds the limit.

Get Automatic Intelligent Linkage List

Last updated: 2025-04-30 15:48:52

1. API Description

api request domain name: `iot.cloud.tencent.com/api/exploreropen/appapi`.

This interface (AppGetAutomationList) is used to obtain the Automatic Intelligent Linkage List.

2. Input Parameter

Name	Type	Required	Description
AccessToken	String	Yes	Common parameter, the Access Token obtained by users upon successful login through WeChat ID, mobile phone or email account.
RequestId	String	Yes	Common parameter, unique request ID, can be generated manually. It is recommended to use uuid. When locating issues, provide the RequestId for the request.
Action	String	Yes	Common parameter, this interface takes values: AppGetAutomationList.
FamilyId	String	Yes	Family ID.

3. Output Parameter

Name	Type	Description
RequestId	String	Common parameter, unique request ID, same as the input parameter. When locating issues, provide the RequestId for the request.
List	Array of AutomationListitem	Automatic Intelligent Linkage List.

4. Example

Input example

```
POST https://iot.cloud.tencent.com/api/exploreropen/tokenapi HTTP/1.1
content-type: application/json
{
  "RequestId": "req_1",
  "Action": "AppGetAutomationList",
  "FamilyId": "f_9****c1",
  "AccessToken": "8b4a70dd16105f*****18edd4e78a3bb8ec"
}
```

Output example: successful

```
{
  "Response": {
    "List": [
      {
        "AutomationId": "a_cd61*****d21d6aeffea",
        "Icon": "https://www.****.com/",
        "Name": "autoName",
        "Status": 0
      }
    ],
    "RequestId": "req_1"
  }
}
```

5. Error Code

Error Code	Description
InternalServerError	Internal error.
InternalServerError.InternalServerErrorDB	DB error.
InvalidParameterValue	Invalid parameter value.
InvalidParameterValue.InvalidAccessToken	The Token is invalid.
InvalidParameterValue.NoPermission	User has no permission for the device.

Getting Automatic Intelligent Linkage Details

Last updated: 2025-04-30 15:49:07

1. API Description

api request domain name: `iot.cloud.tencent.com/api/exploreropen/appapi`.

This interface (AppDescribeAutomation) is used to obtain automatic intelligent linkage details.

2. Input Parameter

Name	Type	Required	Description
AccessToken	String	Yes	common parameter, access Token obtained upon successful login by users through WeChat ID, mobile phone or email account.
RequestId	String	Yes	common parameter, unique request ID, can be generated manually, recommended to use uuid. When locating issues, you need to provide the RequestId for the request.
Action	String	Yes	common parameter, this interface takes values: AppDescribeAutomation.
AutomationId	String	Yes	Automatic intelligent linkage ID.

3. Output Parameters

Name	Type	Description
RequestId	String	common parameter, unique request ID, same as the input parameter, when locating issues, you need to provide the RequestId for the request
Data	Object of Automation	Automatic intelligent linkage entity

4. Example

Input Example

```
POST https://iot.cloud.tencent.com/api/exploreropen/tokenapi HTTP/1.1
content-type: application/json
{
  "RequestId": "req_1",
  "Action": "AppDescribeAutomation",
  "AutomationId": "a_cd61dd*****7d21d6aeffea",
  "AccessToken": "8b4a70dd16105f*****18edd4e78a3bb8ec"
}
```

Output sample: successful

```
{
  "Response": {
    "Data": {
      "AutomationId": "a_cd61dd*****d6aeffea",
      "Icon": "https://www.****.com/",
      "Name": "autoName",
      "Status": 0,
      "MatchType": 1,
      "EffectiveBeginTime": "00:00", // Two new configuration
parameters to indicate start and end times
      "EffectiveEndTime": "23:59",
      "EffectiveDays": "1111111", // A 7-digit number composed of 0s and
1s, where 0 indicates not to execute, 1 indicates to execute, the first
digit represents Sunday, and the following digits represent Monday to
Saturday sequentially
      "Conditions": [
        {
          "CondId": "abc",
          "CondType": 0,
          "Property": {
            "ProductId": "NJ****LZT4",
            "DeviceName": "gw***v",
            "AliasName": "d****me",
            "IconUrl": "productIcon",
            "PropertyId": "switch",
            "Op": "eq",
```

```
        "Value": 1
      }
    },
    {
      "CondId": "efg",
      "CondType": 1,
      "Timer": {
        "Days": "1000000",
        "TimePoint": "09:30"
      }
    }
  ],
  "Actions": [
    {
      "ActionType": 0,
      "ProductId": "R3****0EU",
      "DeviceName": "df****yY",
      "Data": "{\"brightness\": 25}",
      "AliasName": "12345",
      "IconUrl": "url"
    },
    {
      "ActionType": 1,
      "Data": "100"
    }
  ]
},
"RequestId": "req_1"
}
```

5. Error Code

Error Code	Description
InternalServerError	Internal error.
InternalServerError.InternalServerErrorExceptionDB	DB error.
InvalidParameterValue	Invalid parameter value.

InvalidParameterValue.InvalidAccessToken	The Token is invalid.
InvalidParameterValue.NoPermission	User has no permission for the device.

Deleting Automatic Intelligent Linkage

Last updated: 2025-12-19 09:16:28

1. API Description

api request domain name: `iot.cloud.tencent.com/api/exploreropen/appapi`.

This API (AppDeleteAutomation) is used to delete automatic intelligent linkage.

2. Input Parameters

Name	Type	Required	Description
AccessToken	String	Yes	Common parameter, the Access Token obtained upon successful login by users through WeChat ID, mobile phone or email account.
RequestId	String	Yes	Common parameter, unique request ID, can be generated manually, uuid is recommended for use. Locating issues, need to provide the RequestId for the request.
Action	String	Yes	Common parameter, this interface takes values: AppDeleteAutomation.
AutomationId	String	Yes	Automatic intelligent linkage ID.

3. Output Parameters

Name	Type	Description
RequestId	String	Common parameter, unique request ID, same as the input parameter. Locating issues, need to provide the RequestId for the request.

4. Example

Input example

```
POST https://iot.cloud.tencent.com/api/exploreropen/tokenapi HTTP/1.1
```

```
content-type: application/json
{
  "RequestId": "req_1",
  "Action": "AppDeleteAutomation",
  "AutomationId": "a_cd61d*****1d6aeffea",
  "AccessToken": "8b4a70dd16105f*****18edd4e78a3bb8ec"
}
```

Output sample: successful

```
{"Response":{"RequestId":"req_1"}}
```

5. Error Code

Error Code	Description
InternalError	Internal error.
InternalError.InternalServerExceptionDB	DB error.
InvalidParameterValue	Invalid parameter value.
InvalidParameterValue.InvalidAccessToken	The Token is invalid.
InvalidParameterValue.NoPermission	The user has no permission to access the device.

Modifying Automatic Intelligent Linkage

Last updated: 2025-04-30 15:50:15

1. API Description

api request domain name: `iot.cloud.tencent.com/api/exploreropen/appapi`.

This interface (AppModifyAutomation) is used to modify automatic intelligent linkage.

2. Input Parameters

Name	Type	Required	Description
AccessToken	String	Yes	Common parameter, the Access Token obtained upon successful login by users through WeChat ID, mobile phone or email account.
RequestId	String	Yes	Common parameter, unique request ID, can be generated manually, recommended for use as uuid. Locating issues, need to provide the RequestId for the request.
Action	String	Yes	Common parameter, this interface takes values: AppModifyAutomation.
AutomationId	String	Yes	Automatic intelligent linkage ID.
Icon	String	Yes	Create the background image address for automatic intelligent linkage.
Name	String	Yes	Name of automatic intelligent linkage.
Status	int	Yes	Automatic intelligent linkage switch state. • 0: Disable. • 1: Enable.
MatchType	int	Yes	Condition match type. • 0: All conditions are met. • 1: One condition is met.

Conditions	Array of Condition	Yes	Automatic intelligent linkage trigger condition.
Actions	Array of Action	Yes	Automatic intelligent linkage action list.
EffectiveBeginTime	String	Yes	Start of effective time.
EffectiveEndTime	String	Yes	Expiration time.
EffectiveDays	String	Yes	A 7-digit number composed of 0s and 1s. 0 means not execute, and 1 means execute. The first digit represents Sunday, and the following digits represent Monday to Saturday sequentially.

3. Output Parameters

Name	Type	Description
RequestId	String	Common parameter, unique request ID, same as the input parameter. Locating issues, need to provide the RequestId for the request.

4. Example

Input example

```
POST https://iot.cloud.tencent.com/api/exploreropen/tokenapi HTTP/1.1
content-type: application/json
{
  "RequestId": "req_1",
  "Action": "AppCreateAutomation",
  "AutomationId": "a_cd61ddbe*****47d21d6aeffea",
  "Icon": "https://www.****.com/",
  "Name": "autoName",
  "Status": 0, //Integration switch: 0: Disabled; 1: Enabled
  "MatchType": 1, //Condition match type: 0: All conditions AND; 1:
All conditions OR
  "EffectiveBeginTime": "00:00", // Two new configuration parameters,
indicate start and end times
```

```

    "EffectiveEndTime": "23:59",
    "EffectiveDays": "1111111", // A 7-digit number composed of 0s and
1s, where 0 means not execute and 1 means execute. The first digit
represents Sunday, followed by Monday to Saturday in sequence.
    "Conditions": [
        {
            "CondId": "abc", // Condition ID, ensure it is unique within
a linkage
            "CondType": 0, //Condition type: 0 - Device attribute value
condition; 1 - Scheduling condition
            "Property": {
                "ProductId": "N****4",
                "DeviceName": "gw**v",
                "AliasName": "devName",
                "IconUrl": "productIcon",
                "PropertyId": "switch", //Attribute ID of the device
                "Op": "eq", // Conditional operator. eq: equal. ne: not
equal. gt: more than. lt: less than. ge: larger than or equal to. le:
less than or equal to.
                "Value": 1 //Comparative Value
            }
        },
        {
            "CondId": "efg",
            "CondType": 1,
            "Timer": {
                "Days": "1000000", // A 7-digit number composed of 0s
and 1s, where 0 means not execute and 1 means execute. The first digit
represents Sunday, followed by Monday to Saturday in sequence.
                "TimePoint": "09:30" // Trigger time, 24-hour format,
for example, "14:00"
            }
        }
    ],
    "Actions": [
        {
            "ActionType": 0, // 0: Device action, specific parameter
setting is Data    0: Device action    1: Delay    2: Scenario    3:
Notification
            "ProductId": "R3****0EU",

```



```
    "DeviceName": "df****yY",
    "Data": "{\"brightness\": 25}", // ActionType is 0: Device
action parameter; 1: Delay in seconds; 2: Execution scene Id; 3:
Notification content
    "AliasName": "12345",
    "IconUrl": "url"
  },
  {
    "ActionType": 1, // 1: Delay action, specific delay time is
Data, value in seconds
    "Data": "100"
  }
],
  "AccessToken": "***v2"
}
```

Output example: succeed

```
{"Response":{"RequestId":"req_1"}}
```

5. Error Code

Error Code	Description
InternalServerError	Internal error.
InternalServerError.InternalServerErrorDB	DB error.
InvalidParameterValue	Invalid parameter value.
InvalidParameterValue.InvalidAccessToken	The Token is invalid.
InvalidParameterValue.NoPermission	The user has no permission for the device.
LimitExceeded	The quantity exceeds the limit.

Modify Automatic Intelligent Linkage Status

Last updated: 2025-04-30 16:16:14

1. API Description

api request domain name: `iot.cloud.tencent.com/api/exploreropen/appapi`.

This interface (AppModifyAutomationStatus) is used to modify the automatic intelligent linkage status.

2. Input Parameter

Name	Type	Required	Description
AccessToken	String	Yes	Common parameters, Access Token obtained upon successful login by users through WeChat ID, mobile phone or email account.
RequestId	String	Yes	Common parameters. Unique request ID can be generated manually. It is recommended to use uuid. When locating issues, provide the RequestId for the request.
Action	String	Yes	Common parameters. This interface takes values: AppModifyAutomationStatus.
AutomationId	String	Yes	Automatic intelligent linkage ID.
Status	int	Yes	Automatic intelligent linkage switch state. • 0: Disable. • 1: Enable.

3. Output Parameter

Name	Type	Description
RequestId	String	Common parameters. Unique request ID, same as the input parameter. When locating issues, provide the RequestId for the request.

4. Example

Input example

```
POST https://iot.cloud.tencent.com/api/exploreropen/tokenapi HTTP/1.1
content-type: application/json
{
  "RequestId": "req_1",
  "Action": "AppGetAutomationList",
  "AutomationId": "a_cd61dd*****1d6aeffea",
  "Status": 1, //Integration switch: 0: Disabled; 1: Enabled
  "AccessToken": "8b4a70dd16105f*****18edd4e78a3bb8ec"
}
```

Output sample: successful

```
{"Response":{"RequestId":"req_1"}}
```

5. Error Code

Error Code	Description
InternalServerError	Internal error.
InternalServerError.InternalServerErrorDB	DB error.
InvalidParameterValue	Invalid parameter value.
InvalidParameterValue.InvalidAccessToken	The Token is invalid.
InvalidParameterValue.NoPermission	User has no permission for the device.

Data Query

Retrieve the Device Object Model Data

Last updated: 2025-04-30 16:17:09

1. API Description

api request domain name: `iot.cloud.tencent.com/api/exploreropen/tokenapi`.

This interface (AppGetDeviceData) is used to obtain Device Object Model data.

2. Input Parameter

Name	Type	Required	Description
AccessToken	String	Yes	Common parameter. AccessToken is used for user authentication of a logged-in user.
RequestId	String	Yes	Common parameter. Unique request ID can be generated manually. It is recommended to use uuid. When locating issues, you need to provide the RequestId for the request.
Action	String	Yes	Common parameter. This interface takes values: AppGetDeviceData.
ProductId	String	Yes	Product ID.
DeviceName	String	Yes	Device name.

3. Output Parameters

Name	Type	Description
RequestId	String	Common parameter. Unique request ID, same as the input parameter. When locating issues, you need to provide the RequestId for the request.
Data	String	Return data.

4. Samples

Input Example

```
POST https://iot.cloud.tencent.com/api/exploreropen/tokenapi HTTP/1.1
content-type: application/json
{
  "RequestId": "req_1",
  "Action": "AppGetDeviceData",
  "ProductId": "22F9Y****O",
  "DeviceName": "light1",
  "AccessToken": "8b4a70dd16105f*****18edd4e78a3bb8ec"
}
```

Output sample: successful

```
{
  "Response": {
    "Data": "{\\"brightness\\":
{\\\"Value\\":8,\\\"LastUpdate\\":1552904784552},\\\"color\\":
{\\\"Value\\":1,\\\"LastUpdate\\":1552904784552},\\\"light_switch\\":
{\\\"Value\\":0,\\\"LastUpdate\\":1552904784552},\\\"name\\":
{\\\"Value\\\":\\"light1\\",\\\"LastUpdate\\":1552904784552}}",
    "RequestId": "req_1"
  }
}
```

5. Error Code

Error Code	Description
InternalServerError	Internal error.
InvalidParameterValue	Invalid parameter value.
InvalidParameterValue.InvalidAccessToken	The Token is invalid.

6. Error Information

Name	Type	Description
Code	String	Error code.
Message	String	Error message.

Retrieve Device Object Model Historical Data

Last updated: 2025-04-30 16:17:24

1. API Description

api request domain name: `iot.cloud.tencent.com/api/exploreropen/tokenapi`.

This interface (AppGetDeviceDataHistory) is used to obtain Device Object Model historical data.

2. Input Parameter

Name	Type	Required	Description
AccessToken	String	Yes	Common parameter. AccessToken is used for user authentication of a logged-in user.
RequestId	String	Yes	Common parameter. Unique request ID can be generated manually. It is recommended to use uuid. When locating issues, you need to provide the RequestId for the request.
Action	String	Yes	Common parameter. This interface takes values: AppGetDeviceDataHistory.
DeviceId	String	Yes	Device ID, product ID/device name.
FieldName	String	Yes	Inquiry attribute name.
Context	String	No	The paging cursor can be omitted for the initial query.
MinTime	Int	Yes	Start time, millisecond timestamp.
MaxTime	Int	Yes	End time, millisecond timestamp.
Limit	Int	Yes	Single page data volume.

3. Output Parameters

Name	Type	Description
RequestId	String	Common parameter. Unique request ID, same as the input parameter. When locating issues, you need to provide the RequestId for the request.
Context	String	Pagination cursor
FieldName	String	Inquiry attribute name.
Listover	Bool	Whether all the data has been queried. • True indicates that you can continue reading if not fully read. • False indicates that it has been completely read.
Results	Array of DataHistoryItem	Return data.

4. Samples

Example 1

Input Example

```
POST https://iot.cloud.tencent.com/api/exploreropen/tokenapi HTTP/1.1
content-type: application/json
{
  "RequestId": "f92406b3-5a9a-4fe8-bc43-45e3d794bb68",
  "Action": "AppGetDeviceDataHistory",
  "ProductId": "22F9Y6II7O",
  "DeviceName": "light1",
  "MinTime": 1560949017000,
  "MaxTime": 1561035417000,
  "Limit": 20,
  "FieldName": "light_switch",
  "AccessToken": "8b4a70dd16105f*****18edd4e78a3bb8ec"
}
```

Output example: successful


```
{
  "Response": {
    "Context": "abc",
    "FieldName": "brightness",
    "Listover": false,
    "RequestId": "f92406b3-5a9a-4fe8-bc43-45e3d794bb68",
    "Results": [
      {
        "Time": "1560949025494",
        "Value": "77"
      }
    ]
  }
}
```

5. Error Code

Error Code	Description
InternalError	Internal error.
InvalidParameterValue	Invalid parameter value.
InvalidParameterValue.InvalidAccessToken	The Token is invalid.

Getting Historical Events of Devices

Last updated: 2025-04-30 16:17:39

1. API Description

api request domain name: `iot.cloud.tencent.com/api/exploreropen/tokenapi`.

This interface (AppListEventHistory) is used to obtain the historical events of a device.

2. Input Parameters

Name	Type	Required	Description
AccessToken	String	Yes	Common parameter, the Access Token obtained upon successful login by users through WeChat ID, mobile phone or email account.
RequestId	String	Yes	Common parameters, unique request ID, can be generated manually, recommended for use as uuid. Locating issues, need to provide the RequestId for the request.
Action	String	Yes	AppListEventHistory.
ProductId	String	Yes	product ID.
DeviceName	String	Yes	Device name.
Type	String	No	Event type to search: alert indicates an Alarm, fault indicates a failure, info indicates information. Leaving it blank means querying all the above event types.
StartTime	Int64	No	Start time (Unix timestamp, in seconds), 0 means current system time.
EndTime	Int64	No	End time (Unix timestamp, in seconds), 0 means current system time.
Context	String	No	Search context, used as query cursor.
Size	Int64	No	The maximum number of historical data items obtained at a time.

EventId	String	No	Event identifier can be used to specify a specified query for a specific event. If not specified, query all events.
---------	--------	----	---

3. Output Parameter

Name	Type	Description
RequestId	String	Common parameters, unique request ID, consistent with the input parameters.
Context	String	Search context, used as query cursor.
Total	Int64	Number of search results.
Listover	Bool	Whether the search result has ended.
EventHistory	Array	Collect the result set.

4. Example

Example 1

Input Example

```
POST https://iot.cloud.tencent.com/api/exploreropen/tokenapi HTTP/1.1
content-type: application/json
{
  "AccessToken": "accesstoken*****acc",
  "RequestId": "026fe48b-0c31-4a7c-aaf6-83ba3e9s4bcf",
  "ProductID": "ABCDE12345",
  "DeviceName": "DEV",
  "StartTime": 0,
  "EndTime": 0,
  "Type": "info",
  "Size": 5,
  "Context": "",
  "Action": "AppListEventHistory"
}
```

Output example: successful

```
{
  "Response": {
    "ClientToken": "1ee12*****7df703e0d",
    "OutputParams": "",
    "RequestId": "026fe48b-0c31-4a7c-aaf6-83ba3e9s4bcf",
    "Context":
"DnF1ZXJ5VGhlbkZld*****SUQ0UWVPZU1MLXI1QnBkbGc=",
    "EventHistory": [{
      "TimeStamp": 1553484903,
      "ProductId": "product1",
      "DeviceName": "light3",
      "EventId": "status_report",
      "Type": "info",
      "Data": "{\\"status\\": 0,\\"message\\": \\"test
message:11:35:03.288277\\"}"
    }],
    "Total": 1
  }
}
```

5. Error Code

Error Code	Description
InternalError	Internal error.
InvalidParameterValue	Parameter value error
InvalidParameterValue.InvalidAccessToken	The Token is invalid.

Long Connection Communication

Registering Listening

Last updated: 2025-04-30 16:21:36

1. API Description

api request domain name: `wss://iot.cloud.tencent.com/iotstudio_v2_weapp_1`

This interface (DeviceRegistrationMonitor) is used to subscribe to the WS server. After successful subscription, device online/offline status and attribute data can be obtained through real-time push by listening to `ws.onmessage`.

2. Input Parameter

Name	Type	Required	Description
action	String	Yes	Common parameter, this interface takes values: ActivePush.
reqId	String	Yes	Common parameter, indicating the request sequence, unique request ID, which can be generated manually.
params	Object of ActionPushParam	Yes	Passed parameters when registering a device.

3. Output Parameters

Name	Type	Description
reqId	String	Common parameter, indicating the request sequence, consistent with the input parameter.
error	String	Status code, being empty indicates correct.
error_message	String	Detailed error description. Being empty indicates correctness.
data	Object of ActionPushData	Server response packet after listening.

4. Samples

Example 1

Input example (Convert data to a json string before sending)

```
const ws = new
WebSocket("wss://iot.cloud.tencent.com/iotstudio_v2_weapp_1")

ws.send(JSON.stringify({
  "action": "ActivePush",
  "reqId": "req0001",
  "params":{
    "DeviceIds": ["product ID/DeviceName","product ID/DeviceName"]
    "AccessToken": "f6*****3aad"
  }
}));
```

Output sample: succeed

```
{
  "error": "",
  "error_message": "",
  "data": {
    "deviceIds": ["product ID/DeviceName", "product ID/DeviceName"]
  },
  "reqId": "req0001"
}
```

Output sample: failure

```
{
  "error": "",
  "error_message": "",
  "data": {
    "deviceIds": []
  },
  "reqId": "req0001"
}
```

Heartbeat

Last updated: 2025-04-30 16:21:50

1. API Description

API request domain name: `wss://iot.cloud.tencent.com/iotstudio_v2_weapp_1`

Device keep-alive. Update the device timestamp through this API to maintain the device's listening status and keep the WebSocket connection simultaneous. Multiple calls are allowed, once every 60 seconds.

2. Input Parameters

Name	Type	Required	Description
action	String	Yes	Common parameter, this interface takes values: YunApi.
reqId	String	Yes	Common parameter, indicating the request sequence, unique request ID, can be generated manually.
params	Object	Yes	For the detailed parameters passed in when calling the api, please see YunApiParam .

3. Output Parameters

Name	Type	Description
reqId	String	Common parameter, indicating the request sequence, same as the input parameter.
error	String	status code, being empty indicates correct.
error_message	String	Detailed error description, being empty indicates correct.
data	Object	For the server response of the Heartbeat API, please see HeartBeatData .

4. Samples

Input example (Convert data to json string before sending)

```
const ws = new
WebSocket("wss://iot.cloud.tencent.com/iotstudio_v2_weapp_1")

ws.send(JSON.stringify({
  "action": "YunApi",
  "reqId": "req0001",
  "params": {
    "Action": "AppDeviceTraceHeartBeat",
    "ActionParams": {
      "DeviceIds": ["product ID/device name", "product ID/device
name"]
      "AccessToken": "f6*****3aad",
      "RequestId": "rest-client"
    }
  }
}))
```

Output sample: successful

```
{
  "error": "",
  "error_message": "",
  "data": {
    "Response": {
      "RequestId": "rest-client"
    }
  },
  "reqId": "req0001"
}
```

Output sample: failure (server error)

```
{
  "error": "-1",
  "error_message": "Error connecting with backend, contact the admin"
  "data": {},
  "reqId": "req0001"
}
```


Output sample: failure (logic error)

```
{
  "error": "",
  "error_message": "",
  "data": {
    "Response": {
      "Error": {
        "Code":
"UnauthorizedOperation.APPNoPermissionToStudioProduct",
        "Message": "App has no permission to perform operations
on this product"
      },
      "RequestId": "rest-client"
    }
  },
  "reqId": "req0001"
}
```

Device Status Push

Last updated: 2025-04-30 16:22:04

1. API Description

api request domain name: `wss://iot.cloud.tencent.com/iotstudio_v2_weapp_1`

Device status push is used to obtain the online and offline status of user-bound devices, attributes and events reported by devices, and execution results of device behaviors in real time in mini programs or apps. It is required to successfully call the registered device listening API and obtain real-time push of device status and attribute values through listening to WebSocket's OnMessage. If the obtained `event.data.push` is `true`, it signifies that the message is a proactive push of device status change.

2. Input Parameter

Name	Type	Description
push	Boolean	This message is a proactive push for device status change.
action	String	Fixed as <code>DeviceChange</code> , representing device status change.
params	String	Return device information. See detailed explanation in message type description .

3. Output Parameters

Name	Type	Description
RequestId	String	Request ID.

4. Example

Application Side ws Subscription to Device Status Change

Request Sample

```
ws.onmessage = (event) => {  
  let data;  
  try {  
    data = JSON.parse(event.data);  
  }  
}
```

```
    } catch (e) {
      console.log(`onMessage parse event.data error: ${event.data}`);
      return;
    }

    if (data.push) {
      this.emit('push', data);
    } else if (typeof data.reqId !== 'undefined' &&
this.requestHandlerMap.has(data.reqId)) {
      this.requestHandlerMap.get(data.reqId)(null, data);
    }
  };
};
```

Input Parameter Example:

```
{
  "action": "DeviceChange",
  "params": {
    "Time": "2023-07-13T14:19:48+08:00",
    "Type": "Property",
    "SubType": "Report",
    "Topic": "$thing/up/property/productId/deviceName",
    "Payload": "ey*****iOjQwfX0=",
    "Seq": 1689229188406,
    "DeviceId": "productId/deviceName"
  },
  "push": true
}
```

Output Example: Succeed

```
{
  "Response": {
    "RequestId": "up03"
  }
}
```

5. Parameter Descriptions

Parameter Name	Parameter Type	Description
Time	String	Time, in UTC format, for example: 2023-07-13T14:19:48+08:00.
Type	String	Message type. See Type description .
SubType	String	Message subtype. See SubType description .
Topic	String	Message Topic. Log in to IoT Explorer Console and enter the target product development page. Select Device Development > Topic List to view.
Payload	String	Message body base64 encoded content. See Payload description .
Seq	Int64	Serial number.
DeviceId	String	Device ID, product ID in deviceName format.

Type Description

Type	Description
StatusChange	Device status change message, including device online and offline notification.
Shadow	Shadow messages.
Property	Attribute message.
Template	Thing model message.
Event	Event message.
Action	Behavior message.
UserDefined	Custom.

SubType Description

SubType	Description
Online	Device goes online.
Offline	Device offline.
Report	Device reporting.
Push	Server push to device.

Payload Description

Payload is the string literal of the message body after base64 encoding. An example is as follows:

Message Body

base64 Encoding

Template Message Example

The Thing Model message Topic is empty. Payload is the content of the message after base64 encoding.

```

    "Type": "Template",
    "SubType": "Push",
    "Topic": "",
    "Payload":
"eyJtZXRob2QiOiJjb250cm9sIiwia2xpZW50VG9rZW4iOiIxMjM0NTY3OCIsInBhcmFtcyI6eyJzd2l0Y2giOjB9fQ==",
    "Seq": 1689231611907,
    "DeviceId": "productId/deviceName"
  },
  "push": true
}

```

base64 decoded message body of the Payload

See detailed explanation in [Thing Model](#).

```

{
  "method": "control",
  "clientToken": "12345678",
  "params": {
    "switch": 0
  }
}

```

Event Message Example

```

{
  "Time": "2019-04-11T20:42:36+08:00",
  "Type": "Event",
  "SubType": "Report",
  "Topic": "$thing/up/event/US4CJ11DIK/LIAO",
  "Payload":
"ewogICAgICAgICJtZXRob2QiOiAiZXZlbnRfcG9zdCIsc2VudFRva2VuLTU1OSJ9",
  "Seq": 300,
  "DeviceId": "US4CJ11DIK/LIAO"
}

```

base64-decode the message body of the Payload

```
{
  "method": "event_post",
  "version": "1.0",
  "eventId": "low_voltage",
  "params": {
    "voltage": 2.278081
  },
  "clientToken": "clientToken-559"
}
```

Attribute Message Example

```
{
  "Time": "2019-10-30T10:41:00+08:00",
  "Type": "Property",
  "SubType": "Report",
  "Topic": "$thing/up/property/WTJTZX0MF2/device_property001",
  "Payload":
    "eyJ0aW1lc3RhbXAiOiAwLCAicGFyYW1zIjogeyJicmlnaHRuZXNzIjogNTl9LCAibWV0aG9kIjogInJlcG9ydCIsICJjbGllbnRUb2t1biiI6ICJ0bTIwMTkxMDMwMTA0MDUzXzEifQ==",
  "Seq": 6,
  "DeviceId": "WTJTZX0MF2/device_property001"
}
```

base64 decoded message body of the Payload

```
{
  "method": "report",
  "params": {
    "timestamp": 0,
    "params": {
      "brightness": 59
    }
  },
  "method": "report",
  "clientToken": "tm20191030104053_1"
}
```

```
}
```

Status Change Message Example

Device online message.

```
{
  "Time": "2019-03-28T15:17:33+08:00",
  "Type": "StatusChange",
  "SubType": "Online",
  "Topic": "",
  "Payload": "",
  "Seq": 6
}
```

Device offline.

```
{
  "Time": "2019-03-28T15:17:33+08:00",
  "Type": "StatusChange",
  "SubType": "Offline",
  "Topic": "",
  "Payload": "",
  "Seq": 7
}
```

Action: Sample Behavior

Sample push for Action message

```
{
  "Time": "2021-05-13T21:34:05+08:00",
  "Type": "Action",
  "SubType": "Push",
  "Topic": "",
  "Payload":
    "eyJtZXRob2QiOiJhY3Rpb24iLCJjbGllbnRUb2t1biI6IjEzNjk0NzQ5Mzo6cmVxXzEiLCJhY3Rpb25JZCI6ImxpZ2h0X2JsaW5rIiwidGltZXN0YW1wIjoxNjIwOTIyOTQ1LCJwYXJhbXMhOnsiY29sb3IiOiJEsInRpbWUiOiJYsInRvdGFsX3RpbWUiOiJYwfX0K",
  "Seq": 1620912845,
}
```


base64 decoded message body of the Payload

Action message report example

base64 decoded message body of the Payload

Page 397 of 444

```
"response": {  
  "result": 1  
}
```

Cloud Storage Service

Pull Cloud Storage Event List

Last updated: 2025-04-30 16:22:31

1. API Description

api request domain name: `iot.cloud.tencent.com/api/exploreropen/tokenapi`

This interface (lotVideoDescribeCloudStorageEvents) is used to pull the Cloud Storage Event list.

2. Input Parameters

Name	Type	Required	Description
Action	String	Yes	common parameter, this interface takes values: lotVideoDescribeCloudStorageEvents.
RequestId	String	Yes	common parameter, unique request ID, can be generated manually, uuid is recommended. Locating issues, need to provide the RequestId for the request.
AccessToken	String	Yes	common parameter, AccessToken is used for user authentication of a logged-in user.
ProductId	String	Yes	product ID.
DeviceName	String	Yes	Device name.
StartTime	Integer	No	Start time (Unix timestamp, in seconds), 0 means current system time - 24h
EndTime	Integer	No	End time (Unix timestamp, in seconds), 0 means current time
Context	String	No	Request context, used as query cursor
Size	Integer	No	The maximum number of data items that can be queried is 10 by default. Assuming Size=10 is passed,

			and the actual number of events returned is N, then $5 \leq N \leq 10$.
EventId	String	No	Event identifier can be used to specify a query for a specific event. If not specified, all events will be queried.
UserId	String	No	User ID
ChannelId	Integer	No	Channel ID. Leave it blank for non-NVR devices. Required for NVR devices. None by default.

3. Output Parameters

Name	Type	Description
RequestId	String	RequestId.
Events	Array of CloudStorageEvent	Cloud Storage Event List
Context	String	Request context, used as query cursor
Listover	Boolean	Whether pulling results has ended
Total	Integer	Number of internal results, which is not equivalent to the total number of events
VideoURL	String	Video playback URL

4. Example

Input Example

```
POST https://iot.cloud.tencent.com/api/exploreropen/tokenapi HTTP/1.1
content-type: application/json
{
  "Action": "AppDescribeTemplateBinding",
  "AccessToken": "bf*****d098",
  "RequestId": "f92406b3-5a9a-4fe8-bc43-45e3d794bb68",
  "ProductId": "PRODUCT_ID",
```

```
"DeviceName": "DEVICENAME"
}
```

Output sample: successful

```
{
  "Response": {
    "RequestId": "bd3ad0f6-5b9f-47cd-83db-223356926772",
    "Total": 13,
    "Events": [
      {
        "StartTime": 842084401,
        "EndTime": 842084401,
        "Thumbnail": "",
        "EventId": "id1_data"
      },
      {
        "StartTime": 842084401,
        "EndTime": 842084401,
        "Thumbnail": "",
        "EventId": "id1_data"
      },
      {
        "StartTime": 842084401,
        "EndTime": 842084401,
        "Thumbnail": "",
        "EventId": "id1_data"
      },
      {
        "StartTime": 1614847697,
        "EndTime": 167000000,
        "Thumbnail": "face_dection.jpg",
        "EventId": "id1_data"
      },
      {
        "StartTime": 1614847697,
        "EndTime": 167000000,
        "Thumbnail": "face_dection.jpg",
        "EventId": "id1_data"
      }
    ]
  }
}
```

```

    },
    {
      "StartTime": 1614847697,
      "EndTime": 1670000000,
      "Thumbnail": "face_dection.jpg",
      "EventId": "id1_data"
    },
    {
      "StartTime": 1614847697,
      "EndTime": 1670000000,
      "Thumbnail": "face_dection.jpg",
      "EventId": "id1_data"
    },
    {
      "StartTime": 1614847697,
      "EndTime": 1670000000,
      "Thumbnail": "face_dection.jpg",
      "EventId": "id1_data"
    },
    {
      "StartTime": 1614847697,
      "EndTime": 1670000000,
      "Thumbnail": "face_dection.jpg",
      "EventId": "id1_data"
    }
  ],
  "Context":
    "DnF1ZXJ5VGhlbkZldGNoKAAAAAABxjxlFm1DUHBXME5XVE55eUdHMDUwZTV4WWcAAAAAAca
    q9BZFeGQwTXM5X1RvNnFRMFNhRm1MNTlRAAAAAAHGqvYWRXhkME1zOV9UbzZxUTBTYUZtTDU
    5UQAAAAAABxk0BFnN5Q1hWNUUp0UjcyZkZma2licmF3SmcAAAAAAcaq9RZFeGQwTXM5X1RvNnF
    RMFNhRm1MNTlRAAAAAAHGhvUWTz12c3FBNVdUWnFSWXh4d012MG1zdWAAAAAABxjxmFm1DUHB
    XME5XVE55eUdHMDUwZTV4WWcAAAAAAcaG9xZPOXZzcUE1V1RacVJZeHh3TXYwaXN3AAAAAAH
    GVbkWTVBJSndzX0VReVNoU3JOb2pWUm40UQAAAAAABxk0AFnN5Q1hWNUUp0UjcyZkZma2licmF
    3SmcAAAAAAcaG-

```

```

BZPOXZzcUE1V1RacVJZeHh3TXywaXN3AAAAAAHGqvgWRXhkME1zOV9UbzZxUTBTYUZtTDU5U
QAAAAABxkfDFjdNdGYtaTR5UTBPNmFFcHkyUmlUVHcAAAAAAcY88hZOVnkwbjNfNFRqV2YwS
Wk1LWczc2V3AAAAAAHGhvyWTzl2c3FBNVdUWnFSWXh4d012MGlzdwAAAAABxqr3FkV4ZDBNc
z1fVG82cVEwU2FGbUw1OVEAAAAAAcAG-
RZPOXZzcUE1V1RacVJZeHh3TXywaXN3AAAAAAHGR8IWN010Zi1pNHlRME82YUVweTJSaVRUd
wAAAAABxquSFnJKTjhzbk5mUmZxMVVCcVhpVUNTZFEAAAAAAcY8ZxZpQ1BwVzBOV1ROeXlHR
zA1MGU1eFlnAAAAAAHGehEWTTRRa1JkRDhUek9qem1CSmNFbHhUZwAAAAABxqr5FkV4ZDBNc
z1fVG82cVEwU2FGbUw1OVEAAAAAAcZNBBZzeUNYVjVKdFI3MmZGZmtpYnJhd0pnAAAAAAHGV
boWTVBJSndzX0VReVNoU3JOb2pWUm40UQAAAAABxnoQFk00UWtSZEQ4VHpPanptQkpjRWx4V
GcAAAAAAcarkxZySk44c25OZlJmcTFVQnFYaVVDU2RRAAAAAAHGTQIWc3lDWFY1SnRSNzJmR
mZraWJyYXdkZWAAAAABxpZ4Fno4eVNTTD1LUVUyblVfMFFFcnhramcAAAAAAcY88xZOVnkwb
jNfNFRqV2YwSWk1LWczc2V3AAAAAAHGTQMwc3lDWFY1SnRSNzJmRmZraWJyYXdkZWAAAAABx
kfEFjdNdGYtaTR5UTBPNmFFcHkyUmlUVHcAAAAAAcAr1BZySk44c25OZlJmcTFVQnFYaVVDU
2RRAAAAAAHGQvoWRXhkME1zOV9UbzZxUTBTYUZtTDU5UQAAAAABxpZ6Fno4eVNTTD1LUVUybl
VfMFFFcnhramcAAAAAAcZHxRY3TXRmLWk0eVEwTzZhRxB5MlJpVFR3AAAAAAHGVbsWTVBJS
ndzX0VReVNoU3JOb2pWUm40UQAAAAABxpZ5Fno4eVNTTD1LUVUyblVfMFFFcnhramcAAAAAA
cY8aBZpQ1BwVzBOV1ROeXlHRzA1MGU1eFlnAAAAAAHGQvsWRXhkME1zOV9UbzZxUTBTYUZtT
DU5UQAAAAABxpZ7Fno4eVNTTD1LUVUyblVfMFFFcnhramc=",
    "Listover": false,
    "VideoURL":
"http://test.iotvideo.tencentcs.com/timeshift/live/test.m3u8"
  }
}

```

Output example: failure

```

{
  "Response": {
    "Error": {
      "Code": "InvalidParameterValue.InvalidAccessToken",
      "Message": "Invalid Token"
    },
    "RequestId": "f92406b3-5a9a-4fe8-bc43-45e3d794bb68"
  }
}

```

5. Error Code

Error Code	Description
------------	-------------

InvalidParameter	Invalid parameter.
InvalidParameterValue	Invalid parameter value.
InvalidParameterValue.TokenInfoSaveError	Token saving failed.

Pulling Cloud Storage Event Thumbnails

Last updated: 2025-04-30 16:22:45

1. API Description

api request domain name: `iot.cloud.tencent.com/api/exploreropen/tokenapi`

This interface (lotVideoDescribeCloudStorageThumbnail) is used to pull the thumbnail of cloud storage events.

2. Input Parameters

Name	Type	Required	Description
Action	String	Yes	common parameter, this interface takes values: lotVideoDescribeCloudStorageThumbnail.
RequestId	String	Yes	common parameter, unique request ID, can be generated manually, recommended to use uuid. Locating issues, need to provide the RequestId for the request.
AccessToken	String	Yes	Common parameter, AccessToken is used for user authentication of a logged-in user.
ProductId	String	Yes	Product ID.
DeviceName	String	Yes	Device name.
Thumbnail	String	Yes	Thumbnail file name

3. Output Parameters

Name	Type	Description
RequestId	String	RequestId.

Thumbnail URL	String	Thumbnail access address
---------------	--------	--------------------------

4. Example

Input Example

```
POST https://iot.cloud.tencent.com/api/exploreropen/tokenapi HTTP/1.1
content-type: application/json
{
  "Action": "AppDescribeTemplateBinding",
  "AccessToken": "bf*****d098",
  "RequestId": "f92406b3-5a9a-4fe8-bc43-45e3d794bb68",
  "ProductId": "PRODUCT_ID",
  "DeviceName": "DEVICENAME",
  "Thumbnail":
"/100008401725/AQTV2839QJ/sp01_32820237_7/events/1615200613.jpg"
}
```

Output sample: successful

```
{
  "Response": {
    "RequestId": "c299d2c9-805f-41a5-853d-db32ba1a16b5",
    "ThumbnailURL": "https://video-cv-1258344699.cos.ap-
guangzhou.myqcloud.com/%2F100008401725%2FAQTV2839QJ%2Fsp01_32820237_5%2F
events%2F1614908216.jpg?q-sign-algorithm=sha1&q-
ak=*****&q-sign-
time=1614909378%3B1614912978&q-key-time=1614909378%3B1614912978&q-
header-list=host&q-url-param-list=&q-
signature=59cfa8d69bb77980ba757cb111de62fc6c4880f5"
  }
}
```

Output example: fail

```
{
  "Response": {
    "Error": {
```

```
      "Code": "InvalidParameterValue.InvalidAccessToken",
      "Message": "Invalid Token"
    },
    "RequestId": "f92406b3-5a9a-4fe8-bc43-45e3d794bb68"
  }
}
```

5. Error Code

Error Code	Description
InvalidParameter	Invalid parameter.
InvalidParameterValue	Invalid parameter value.
InvalidParameterValue.TokenInfoSaveError	Token saving failed.

Get the Date with Cloud Storage

Last updated: 2025-04-30 16:22:59

1. API Description

API request domain name: `iot.cloud.tencent.com/api/exploreropen/tokenapi`

This interface (lotVideoDescribeCloudStorageDate) is used to obtain dates with cloud storage.

2. Input Parameters

Name	Type	Required	Description
Action	String	Yes	Common parameter, this interface takes values: lotVideoDescribeCloudStorageDate.
RequestId	String	Yes	Common parameter, unique request ID, can be generated manually, uuid is recommended for use. Locating issues, need to provide the RequestId for the request.
AccessToken	String	Yes	Common parameter, AccessToken is used for user authentication of a logged-in user.
ProductId	String	Yes	product ID.
DeviceName	String	Yes	Device name.
UserId	String	No	User ID

3. Output Parameters

Name	Type	Description
RequestId	String	RequestId.
Data	Array of String	Cloud storage date array, example value: ["2021-01-05","2021-01-06"]

4. Example

Input Example

```
POST https://iot.cloud.tencent.com/api/exploreropen/tokenapi HTTP/1.1
content-type: application/json
{
  "Action": "AppDescribeTemplateBinding",
  "AccessToken": "bf*****d098",
  "RequestId": "f92406b3-5a9a-4fe8-bc43-45e3d794bb68",
  "ProductId": "PRODUCT_ID",
  "DeviceName": "DEVICENAME"
}
```

Output sample: successful

```
{
  "Response": {
    "RequestId": "77d90d07-e48c-4cf0-bec5-afe5b137f80e",
    "Data": [
      "2021-01-14"
    ]
  }
}
```

Output example: failure

```
{
  "Response": {
    "Error": {
      "Code": "InvalidParameterValue.InvalidAccessToken",
      "Message": "Invalid Token"
    },
    "RequestId": "f92406b3-5a9a-4fe8-bc43-45e3d794bb68"
  }
}
```

5. Error Code

Error Code	Description
InvalidParameter	Invalid parameter.
InvalidParameterValue	Invalid parameter value.
InvalidParameterValue.TokenInfoSaveError	Token saving failed.

Get Cloud Storage Timeline on a Certain Day

Last updated: 2025-04-30 16:23:14

1. API Description

API request domain name: `iot.cloud.tencent.com/api/exploreropen/tokenapi`

This interface (`lotVideoDescribeCloudStorageTime`) is used to obtain the Cloud Storage Timeline of a certain day.

2. Input Parameter

Name	Type	Required	Description
Action	String	Yes	Common parameter, this interface takes values: <code>lotVideoDescribeCloudStorageTime</code> .
RequestId	String	Yes	Common parameter, unique request ID, can be generated manually. It is recommended to use uuid. When locating issues, you need to provide the RequestId for the request.
AccessToken	String	Yes	Common parameter. AccessToken is used for user authentication of a logged-in user.
ProductId	String	Yes	Product ID.
DeviceName	String	Yes	Device name.
Date	String	Yes	Cloud Storage Date, for example, "2020-01-05"
StartTime	Integer	No	Start Time, unix Timestamp
EndTime	Integer	No	End Time, unix Timestamp
UserId	String	No	User ID

3. Output Parameters

Name	Type	Description
RequestId	String	RequestId.
Data	CloudStorageTimeData	Returned data

4. Example

Input Example

```
POST https://iot.cloud.tencent.com/api/exploreropen/tokenapi HTTP/1.1
content-type: application/json
{
  "Action": "AppDescribeTemplateBinding",
  "AccessToken": "bf*****d098",
  "RequestId": "f92406b3-5a9a-4fe8-bc43-45e3d794bb68",
  "ProductId": "PRODUCT_ID",
  "DeviceName": "DEVICENAME",
  "Date": "2021-01-14"
}
```

Output sample: successful

```
{
  "Response": {
    "RequestId": "ac74ecca-b866-4ab1-87e0-18b92816fb20",
    "Data": {
      "VideoURL": "47b9d579-9088-4cc2-a16e-2b8cf245319b",
      "TimeList": [
        {
          "StartTime": 1610607600,
          "EndTime": 1610608200
        },
        {
          "StartTime": 1610607600,
          "EndTime": 1610608200
        }
      ]
    }
  }
}
```



```
{
  {
    "StartTime": 1610607600,
    "EndTime": 1610608200
  },
  {
    "StartTime": 1610607600,
    "EndTime": 1610608200
  }
]
}
}
```

Output example: failure

```
{
  "Response": {
    "Error": {
      "Code": "InvalidParameterValue.InvalidAccessToken",
      "Message": "Invalid Token"
    },
    "RequestId": "f92406b3-5a9a-4fe8-bc43-45e3d794bb68"
  }
}
```

5. Error Code

Error Code	Description
InvalidParameter	Invalid parameter.
InvalidParameterValue	Invalid parameter value.
InvalidParameterValue.TokenInfoSaveError	Token saving failed.

Getting Video Anti-Hotlinking Playback URL

Last updated: 2025-12-19 09:20:30

1. API Description

api request domain name: `iot.cloud.tencent.com/api/exploreropen/tokenapi`

This API (lotVideoGenerateSignedVideoURL) is used to getVideo anti-leech playback URL.

2. Input Parameters

Name	Type	Required	Description
Action	String	Yes	Common parameter, this interface takes values: lotVideoGenerateSignedVideoURL.
RequestId	String	Yes	Common parameter, unique request ID, can be generated manually, recommended to use uuid. Locating issues, need to provide the RequestId for the request.
AccessToken	String	Yes	Common parameter, AccessToken is used for user authentication of a logged-in user.
ProductId	String	Yes	Product ID.
DeviceName	String	Yes	Device name.
VideoURL	String	Yes	Original URL address for video playback
ExpireTime	Integer	Yes	Expiration time of the playback link

3. Output Parameter

Name	Type	Description
RequestId	String	RequestId.

SignedVideoURL	String	Anti-hotlinking playback URL for video
----------------	--------	--

4. Samples

Input example

```
POST https://iot.cloud.tencent.com/api/exploreropen/tokenapi HTTP/1.1
content-type: application/json
{
  "Action": "AppDescribeTemplateBinding",
  "AccessToken": "bf*****d098",
  "RequestId": "f92406b3-5a9a-4fe8-bc43-45e3d794bb68",
  "ProductId": "PRODUCT_ID",
  "DeviceName": "DEVICENAME",
  "ExpireTime": 1619331648,
  "VideoURL":
"http://zylcb.iotvideo.tencentcs.com/timeshift/live/1.m3u8"
}
```

Output sample: successful

```
{
  "Response": {
    "SignedVideoURL":
"http://zylcb.iotvideo.tencentcs.com/timeshift/live/1.m3u8?sign=x",
    "RequestId": "xx"
  }
}
```

Output sample: fail

```
{
  "Response": {
    "Error": {
      "Code": "InvalidParameterValue.InvalidAccessToken",
      "Message": "Invalid Token"
    },
    "RequestId": "f92406b3-5a9a-4fe8-bc43-45e3d794bb68"
  }
}
```

```
    }  
}
```

5. Error Code

Error Code	Description
InvalidParameter	Invalid parameter.
InvalidParameterValue	Invalid parameter value.
InvalidParameterValue.TokenInfoSaveError	Token saving failed.

Pulling Image Streaming Data

Last updated: 2025-12-19 09:18:26

1. API Description

api request domain name: `iot.cloud.tencent.com/api/exploreropen/tokenapi`

This interface (lotVideoDescribeCloudStorageStreamData) is used to pull streaming data.

2. Input Parameters

Name	Type	Required	Description
Action	String	Yes	common parameter, this interface takes values: lotVideoDescribeCloudStorageStreamData.
RequestId	String	Yes	common parameter, unique request ID, can be generated manually, uuid is recommended. Locating issues, need to provide the RequestId for the request.
AccessToken	String	Yes	Common parameter, AccessToken is used for user authentication of a logged-in user.
ProductId	String	Yes	Product ID.
DeviceName	String	Yes	Device name.
StartTime	Integer	Yes	Start time

3. Output Parameters

Name	Type	Description
RequestId	String	RequestId.
VideoStream	String	Image stream video address
AudioStr	String	Image stream audio address

eam

4. Example

Input Example

```
POST https://iot.cloud.tencent.com/api/exploreropen/tokenapi HTTP/1.1
content-type: application/json

{
  "Action": "AppDescribeTemplateBinding",
  "AccessToken": "bf*****d098",
  "RequestId": "f92406b3-5a9a-4fe8-bc43-45e3d794bb68",
  "ProductId": "PRODUCT_ID",
  "DeviceName": "DEVICENAME",
  "StartTime": 1619331648
}
```

Output sample: successful

```
{
  "Response": {
    "VideoStream": "http://xxxxxxx",
    "AudioStream": "http://xxxxxxx",
    "RequestId": "xx"
  }
}
```

Output example: failure

```
{
  "Response": {
    "Error": {
      "Code": "InvalidParameterValue.InvalidAccessToken",
      "Message": "Invalid Token"
    },
    "RequestId": "f92406b3-5a9a-4fe8-bc43-45e3d794bb68"
  }
}
```

5. Error Code

Error Code	Description
InvalidParameter	Invalid parameter.
InvalidParameterValue	Invalid parameter value.
InvalidParameterValue.TokenInfoSaveError	Token saving failed.

Music Service

Song in a Song List

Last updated: 2025-04-30 16:24:08

1. API Description

api request domain name: iot.cloud.tencent.com/api/explorer/open/tokenapi This interface (playlist_song) is used to obtain songs in a song list.

2. Input Parameters

Name	Type	Required	Description
AccessToken	String	Yes	Common parameter. AccessToken is used for user authentication of a logged-in user.
RequestId	String	Yes	Common parameter. Unique request ID can be generated manually. It is recommended to use uuid. Locating issues, you need to provide the RequestId for the request.
Action	String	Yes	Common parameter. This interface takes values: AppKugouUserCommand.
KGCommand	String	Yes	Subcommand. This interface takes values: playlist_song.
DeviceId	String	Yes	Device Id consists of product Id and device Name.
KugouParams	Object of KugouParams_playlist_song	Yes	Parameters of the subcommand.

3. Output Parameter

Name	Type	Description
------	------	-------------

Request Id	String	Common parameter. Unique request ID, same as the input parameter. Locating issues, you need to provide the RequestId for the request.
ErrorCode	Integer	The error code returned by the request, 0 means no error.
ErrorMsg	String	The error information returned by the request.
Data	Object of Data_playlist_song	Returned result

4. Examples

Example 1

Input Example

```
POST https://iot.cloud.tencent.com/api/exploreropen/tokenapi HTTP/1.1
content-type: application/json
{
  "RequestId": "test0191119-012345",
  "Action": "AppKugouUserCommand",
  "AccessToken": "8b4a70dd16105f*****18edd4e78a3bb8ec",
  "DeviceId": "LAEG4YJE1A/dev01",
  "KGCommand": "playlist_song",
  "KugouParams": {...}
}
```

Output example: successful

```
{
  "Response": {
    "Data": {...},
    "ErrorCode": 0, "ErrorMsg": "",
    "RequestId": "keyi20191119-012345"
  }
}
```

5. Error Code

Error Code	Description
InternalServerError	Internal error.
InvalidParameterValue	Invalid parameter value.
InvalidParameterValue.InvalidAccessToken	The Token is invalid.
InvalidParameterValue.InternalRPCError	Call timeout.
ResourceNotFound	The resource does not exist.
ResourceNotFound.DeviceNotExist	The device is not created or has been deleted.
ResourceNotFound.StudioProductNotExist	The product is not created or has been deleted.
ResourceNotFound.AppNotExists	The application is not created or has been deleted.
UnsupportedOperation	Unsupported operation.
UnsupportedOperation.DeviceNotGateway	Not a gateway device.
UnsupportedOperation.NotBindVirtualDevice	Unable to bind virtual device.
UnsupportedOperation.DeviceNotBind	The subdevice is not bound to the gateway device.
UnsupportedOperation.CannotBindSameFamily	The device has been bound. Do not perform repeat binding.
LimitExceeded	Exceed the limit.
LimitExceeded.FamilyDeviceExceedLimit	The device quantity exceeds the limit.
UnauthorizedOperation	No permission to operate.
UnauthorizedOperation.APPNoPermissionToStudioProduct	The App has no permission to operate this product.

Retrieve the Song Playback Link

Last updated: 2025-04-30 16:24:22

1. API Description

api request domain name: `iot.cloud.tencent.com/api/exploreropen/tokenapi` This interface (song_url) is used to obtain the song playback link.

2. Input Parameter

Name	Type	Required	Description
AccessToken	String	Yes	Common parameter. AccessToken is used for user authentication of a logged-in user.
RequestId	String	Yes	Common parameter. Unique request ID can be generated manually. It is recommended to use uuid. Locating issues, you need to provide the RequestId for the request.
Action	String	Yes	Common parameter. This interface takes values: AppKugouUserCommand.
KGCommand	String	Yes	Subcommand. This interface takes values: song_url.
DeviceId	String	Yes	Device Id consists of product Id and device Name.
KugouParams	Object of KugouParams_song_url	Yes	Parameters of the subcommand.

3. Output Parameter

Name	Type	Description
RequestId	String	Common parameter. Unique request ID, same as the input parameter. Locating issues, you need to provide the RequestId for the request.

ErrorCode	Integer	The error code returned by the request, 0 means no error.
ErrorMsg	String	The error information returned by the request.
Data	Object of Data_song_url	Returning Result

4. Samples

Example 1

Input Example

```
POST https://iot.cloud.tencent.com/api/exploreropen/tokenapi HTTP/1.1
content-type: application/json
{
  "RequestId": "test0191119-012345",
  "Action": "AppKugouUserCommand",
  "AccessToken": "8b4a70dd16105f*****18edd4e78a3bb8ec",
  "DeviceId": "LAEG4YJE1A/dev01",
  "KGCommand": "song_url",
  "KugouParams": {...}
}
```

Output Sample: successful

```
{
  "Response": {
    "Data": {...},
  },
  "ErrorCode": 0, "ErrorMsg": "",
  "RequestId": "keyi20191119-012345"
}
```

5. Error Code

Error Code	Description
------------	-------------

InternalError	Internal error.
InvalidParameterValue	Invalid parameter value.
InvalidParameterValue.InvalidAccessToken	The Token is invalid.
InvalidParameterValue.InternalRPCError	Call timeout.
ResourceNotFound	The resource does not exist.
ResourceNotFound.DeviceNotExist	The device is not created or has been deleted.
ResourceNotFound.StudioProductNotExist	The product is not created or has been deleted.
ResourceNotFound.AppNotExists	The application is not created or has been deleted.
UnsupportedOperation	Unsupported operation.
UnsupportedOperation.DeviceNotGateway	Not a gateway device.
UnsupportedOperation.NotBindVirtualDevice	Unable to bind virtual device.
UnsupportedOperation.DeviceNotBind	The subdevice is not bound to the gateway device.
UnsupportedOperation.CannotBindSameFamily	The device has been bound. Do not repeat binding.
LimitExceeded	Exceed the limit.
LimitExceeded.FamilyDeviceExceedLimit	The device quantity exceeds the limit.
UnauthorizedOperation	No permission to operate.
UnauthorizedOperation.APPNoPermissionToStudioProduct	The App has no permission to operate this product.

Daily Recommendation

Last updated: 2025-04-30 16:24:37

1. API Description

API request domain name: `iot.cloud.tencent.com/api/exploreropen/tokenapi`

This interface (`awesome_everyday`) is used to obtain the daily recommended playlist.

2. Input Parameters

Name	Type	Required	Description
AccessToken	String	Yes	Common parameters. AccessToken is used for user authentication of a logged-in user.
RequestId	String	Yes	Common parameters. A unique request ID can be generated manually. It is recommended to use uuid. When locating issues, provide the RequestId for the request.
Action	String	Yes	Common parameter. This interface takes values: <code>AppKugouUserCommand</code> .
KGCommand	String	Yes	Subcommand. This interface takes values: <code>awesome_everyday</code> .
DeviceId	String	Yes	Device Id consists of product Id and device Name.
KugouParams	Object of KugouParams_awesome_everyday	Yes	Parameters of the subcommand.

3. Output Parameter

Name	Type	Description
------	------	-------------

RequestId	String	Common parameters. A unique request ID, same as the input parameter. When locating issues, provide the RequestId for the request.
ErrorCode	Integer	The error code returned by the request, 0 means no error.
ErrorMsg	String	The error information returned by the request.
Data	Object of Data_awesome_everyday	Returned result

4. Samples

Example 1

Input Example

```
POST https://iot.cloud.tencent.com/api/exploreropen/tokenapi HTTP/1.1
content-type: application/json
{
  "RequestId": "test0191119-012345",
  "Action": "AppKugouUserCommand",
  "AccessToken": "8b4a70dd16105f*****18edd4e78a3bb8ec",
  "DeviceId": "LAEG4YJE1A/dev01",
  "KGCommand": "awesome_everyday",
  "KugouParams": {...}
}
```

Output example: succeed

```
{
  "Response": {
    "Data": {...},
    "ErrorCode": 0, "ErrorMsg": "",
    "RequestId": "keyi20191119-012345"
  }
}
```

>

5. Error Code

Error Code	Description
InternalServerError	Internal error.
InvalidParameterValue	Invalid parameter value.
InvalidParameterValue.InvalidAccessToken	The Token is invalid.
InvalidParameterValue.InternalRPCError	Call timeout.
ResourceNotFound	The resource does not exist.
ResourceNotFound.DeviceNotExist	The device is not created or has been deleted.
ResourceNotFound.StudioProductNotExist	The product is not created or has been deleted.
ResourceNotFound.AppNotExists	The application is not created or has been deleted.
UnsupportedOperation	Unsupported operation.
UnsupportedOperation.DeviceNotGateway	Not a gateway device.
UnsupportedOperation.NotBindVirtualDevice	Unable to bind virtual device.
UnsupportedOperation.DeviceNotBind	The subdevice is not bound to the gateway device.
UnsupportedOperation.CannotBindSameFamily	The device has been bound. Do not perform repeat binding.
LimitExceeded	Exceed the limit.
LimitExceeded.FamilyDeviceExceedLimit	The device quantity exceeds the limit.
UnauthorizedOperation	No permission to operate.
UnauthorizedOperation.APPNoPermissionToStudioProduct	The App has no permission to operate this product.

Batch Querying Song Information

Last updated: 2025-04-30 16:24:50

1. API Description

api request domain name: `iot.cloud.tencent.com/api/exploreropen/tokenapi` This interface (song_infos) is used for batch querying song information.

2. Input Parameters

Name	Type	Required	Description
AccessToken	String	Yes	Common parameter. AccessToken is used for user authentication of a logged-in user.
RequestId	String	Yes	Common parameter. Unique request ID can be generated manually. It is recommended to use uuid. Locating issues, you need to provide the RequestId for the request.
Action	String	Yes	Common parameter. This interface takes values: AppKugouUserCommand.
KGCommand	String	Yes	Subcommand. This interface takes values: song_infos.
DeviceId	String	Yes	Device Id consists of product Id and device Name.
KugouParams	Object of KugouParams_song_infos	Yes	Parameters of the subcommand.

3. Output Parameters

Name	Type	Description
RequestId	String	Common parameter. Unique request ID, same as the input parameter. Locating issues, you need to provide the RequestId for the request.

ErrorCode	Integer	The error code returned by the request, 0 means no error.
ErrorMsg	String	The error information returned by the request.
Data	Object of Data_song_infos	Returned result

4. Example

Example 1

Input Example

```
POST https://iot.cloud.tencent.com/api/exploreropen/tokenapi HTTP/1.1
content-type: application/json
{
  "RequestId": "test0191119-012345",
  "Action": "AppKugouUserCommand",
  "AccessToken": "8b4a70dd16105f*****18edd4e78a3bb8ec",
  "DeviceId": "LAEG4YJE1A/dev01",
  "KGCommand": "song_infos",
  "KugouParams": {...}
}
```

Output example: succeed

```
{
  "Response": {
    "Data": {...}
  },
  "ErrorCode": 0, "ErrorMsg": "",
  "RequestId": "keyi20191119-012345"
}
```

5. Error Code

Error Code	Description
InternalServerError	An internal error occurred.
InvalidParameterValue	Invalid parameter value.
InvalidParameterValue.InvalidAccessToken	The Token is invalid.
InvalidParameterValue.InternalRPCError	Call timeout.
ResourceNotFound	The resource does not exist.
ResourceNotFound.DeviceNotExist	The device is not created or has been deleted.
ResourceNotFound.StudioProductNotExist	The product is not created or has been deleted.
ResourceNotFound.AppNotExists	The application is not created or has been deleted.
UnsupportedOperation	Unsupported operation.
UnsupportedOperation.DeviceNotGateway	Not a gateway device.
UnsupportedOperation.NotBindVirtualDevice	Unable to bind virtual device.
UnsupportedOperation.DeviceNotBind	The subdevice is not bound to the gateway device.
UnsupportedOperation.CannotBindSameFamily	The device has been bound. Do not repeat the binding.
LimitExceeded	Exceed the limit.
LimitExceeded.FamilyDeviceExceedLimit	The device quantity exceeds the limit.
UnauthorizedOperation	No permission to operate.
UnauthorizedOperation.APPNoPermissionToStudioProduct	The App has no permission to operate this product.

Recommended Playlist

Last updated: 2025-04-30 16:25:10

1. API Description

api request domain name: `iot.cloud.tencent.com/api/exploreropen/tokenapi` This interface (playlist_awesome) is used to obtain a recommended playlist.

2. Input Parameter

Name	Type	Required	Description
AccessToken	String	Yes	Common parameter. AccessToken is used for user authentication of a logged-in user.
RequestId	String	Yes	Common parameter. Unique request ID can be generated manually. It is recommended to use uuid. When locating issues, you need to provide the RequestId for the request.
Action	String	Yes	Common parameter. This interface takes values: AppKugouUserCommand.
KGCommand	String	Yes	Subcommand. This interface takes values: playlist_awesome.
DeviceId	String	Yes	Device Id consists of product Id and device Name.
KugouParams	Object of KugouParams_playlist_awesome	Yes	Parameters of the subcommand.

3. Output Parameters

Name	Type	Description
------	------	-------------

RequestId	String	Common parameter. Unique request ID, same as the input parameter. When locating issues, you need to provide the RequestId for the request.
ErrorCode	Integer	The error code returned by the request, 0 means no error.
ErrorMsg	String	The error information returned by the request.
Data	Object of data_playlist_awesome	Returned result

4. Samples

Example 1

Input Example

```
POST https://iot.cloud.tencent.com/api/exploreropen/tokenapi HTTP/1.1
content-type: application/json
{
  "RequestId": "test0191119-012345",
  "Action": "AppKugouUserCommand",
  "AccessToken": "8b4a70dd16105f*****18edd4e78a3bb8ec",
  "DeviceId": "LAEG4YJE1A/dev01",
  "KGCommand": "playlist_awesome",
  "KugouParams": {...}
}
```

Output example: successful

```
{
  "Response": {
    "Data": {...},
    "ErrorCode": 0, "ErrorMsg": "",
    "RequestId": "keyi20191119-012345"
  }
}
```

>

5. Error Code

Error Code	Description
InternalServerError	Internal error.
InvalidParameterValue	Invalid parameter value.
InvalidParameterValue.InvalidAccessToken	The Token is invalid.
InvalidParameterValue.InternalRPCError	Call timeout.
ResourceNotFound	The resource does not exist.
ResourceNotFound.DeviceNotExist	The device is not created or has been deleted.
ResourceNotFound.StudioProductNotExist	The product is not created or has been deleted.
ResourceNotFound.AppNotExists	The application is not created or has been deleted.
UnsupportedOperation	Unsupported operation.
UnsupportedOperation.DeviceNotGateway	Not a gateway device.
UnsupportedOperation.NotBindVirtualDevice	Unable to bind virtual device.
UnsupportedOperation.DeviceNotBind	The subdevice is not bound to the gateway device.
UnsupportedOperation.CannotBindSameFamily	The device has been bound. Do not repeat binding.
LimitExceeded	Exceed the limit.
LimitExceeded.FamilyDeviceExceedLimit	The device quantity exceeds the limit.
UnauthorizedOperation	No permission to operate.
UnauthorizedOperation.APPNoPermissionToStudioProduct	The App has no permission to operate this product.

New Song Release

Last updated: 2025-04-30 16:25:25

1. API Description

api request domain name: iot.cloud.tencent.com/api/exploreropen/tokenapi

This interface (awesome_newsong) is used to obtain first release new songs.

2. Input Parameters

Name	Type	Required	Description
AccessToken	String	Yes	common parameter, AccessToken is used for user authentication of a logged-in user.
RequestId	String	Yes	Common parameter, unique request ID, can be generated manually. It is recommended to use uuid. When locating issues, you need to provide the RequestId for the request.
Action	String	Yes	Common parameter, this interface takes values: AppKugouUserCommand.
KGCommand	String	Yes	Subcommand, this interface takes values: awesome_newsong.
DeviceId	String	Yes	Device Id consists of product Id and device Name.
KugouParams	Object of KugouParams_awesome_newsong	Yes	Parameters of the subcommand.

3. Output Parameter

Name	Type	Description
------	------	-------------

RequestId	String	Common parameter, unique request ID, same as the input parameter. When locating issues, you need to provide the RequestId for the request.
ErrorCode	Integer	The error code returned by the request, 0 means no error.
ErrorMsg	String	The returned error information of the request.
Data	Object of Data_awesome_newsong	Returned result

4. Example

Example 1

Input Example

```
POST https://iot.cloud.tencent.com/api/exploreropen/tokenapi HTTP/1.1
content-type: application/json
{
  "RequestId": "test0191119-012345",
  "Action": "AppKugouUserCommand",
  "AccessToken": "8b4a70dd16105f*****18edd4e78a3bb8ec",
  "DeviceId": "LAEG4YJE1A/dev01",
  "KGCommand": "awesome_newsong",
  "KugouParams": { ... }
}
```

Output example: successful

```
{
  "Response": {
    "Data": { ... },
    "ErrorCode": 0, "ErrorMsg": "",
    "RequestId": "keyi20191119-012345"
  }
}
```


>

5. Error Code

Error Code	Description
InternalServerError	Internal error.
InvalidParameterValue	Invalid parameter value.
InvalidParameterValue.InvalidAccessToken	The Token is invalid.
InvalidParameterValue.InternalRPCError	Call timeout.
ResourceNotFound	The resource does not exist.
ResourceNotFound.DeviceNotExist	The equipment is not created or has been deleted.
ResourceNotFound.StudioProductNotExist	The product is not created or has been deleted.
ResourceNotFound.AppNotExists	The application is not created or has been deleted.
UnsupportedOperation	Unsupported operation.
UnsupportedOperation.DeviceNotGateway	Not a gateway device.
UnsupportedOperation.NotBindVirtualDevice	Unable to bind virtual device.
UnsupportedOperation.DeviceNotBind	The subdevice is not bound to the gateway device.
UnsupportedOperation.CannotBindSameFamily	The device has been bound. Do not repeat binding.
LimitExceeded	Exceed the limit.
LimitExceeded.FamilyDeviceExceedLimit	The device quantity exceeds the limit.
UnauthorizedOperation	No permission to operate.
UnauthorizedOperation.APPNoPermissionToStudioProduct	The App has no permission to operate this product.

User Logout

Last updated: 2025-04-30 16:25:38

1. API Description

api request domain name: iot.cloud.tencent.com/api/exploreropen/tokenapi This interface (user_logout) is used for user logout.

2. Input Parameter

Name	Type	Required	Description
AccessToken	String	Yes	Common parameter. AccessToken is used for user authentication of a logged-in user.
RequestId	String	Yes	Common parameter. Unique request ID can be generated manually. It is recommended to use uuid. Locating issues, you need to provide the RequestId for the request.
Action	String	Yes	Common parameter. This interface takes values: AppKugouUserCommand.
KGCommand	String	Yes	Subcommand. This interface takes values: user_logout.
DeviceId	String	Yes	Device Id consists of product Id and device Name.
KugouParams	Object of KugouParams_user_logout	Yes	Parameters of the subcommand.

3. Output Parameters

Name	Type	Description
------	------	-------------

RequestId	String	Common parameter. Unique request ID, same as the input parameter. Locating issues, you need to provide the RequestId for the request.
ErrorCode	Integer	The error code returned by the request, 0 means no error.
ErrorMsg	String	The error information returned by the request.
Data	Object of data_user_logout	Returned result

4. Example

Example 1

Input Example

```
POST https://iot.cloud.tencent.com/api/exploreropen/tokenapi HTTP/1.1
content-type: application/json
{
  "RequestId": "test0191119-012345",
  "Action": "AppKugouUserCommand",
  "AccessToken": "8b4a70dd16105f*****18edd4e78a3bb8ec",
  "DeviceId": "LAEG4YJE1A/dev01",
  "KGCommand": "user_logout",
  "KugouParams": {...}
}
```

Output example: successful

```
{
  "Response": {
    "Data": {...},
    "ErrorCode": 0, "ErrorMsg": "",
    "RequestId": "keyi20191119-012345"
  }
}
```

>

5. Error Code

Error Code	Description
InternalServerError	Internal error.
InvalidParameterValue	Invalid parameter value.
InvalidParameterValue.InvalidAccessToken	The Token is invalid.
InvalidParameterValue.InternalRPCError	Call timeout.
ResourceNotFound	The resource does not exist.
ResourceNotFound.DeviceNotExist	The device is not created or has been deleted.
ResourceNotFound.StudioProductNotExist	The product is not created or has been deleted.
ResourceNotFound.AppNotExists	The application is not created or has been deleted.
UnsupportedOperation	Unsupported operation.
UnsupportedOperation.DeviceNotGateway	Not a gateway device.
UnsupportedOperation.NotBindVirtualDevice	Unable to bind virtual device.
UnsupportedOperation.DeviceNotBind	The subdevice is not bound to the gateway device.
UnsupportedOperation.CannotBindSameFamily	The device has been bound. Do not repeat the binding.
LimitExceeded	Exceed the limit.
LimitExceeded.FamilyDeviceExceedLimit	The device quantity exceeds the limit.
UnauthorizedOperation	No permission to operate.
UnauthorizedOperation.APPNoPermission	The App has no permission to operate this

nToStudioProduct	product.
------------------	----------

User Information

Last updated: 2025-04-30 16:25:53

1. API Description

api request domain name: `iot.cloud.tencent.com/api/exploreropen/tokenapi` This interface (user_info) is used to retrieve user information.

2. Input Parameter

Name	Type	Required	Description
AccessToken	String	Yes	Common parameter. AccessToken is used for user authentication of a logged-in user.
RequestId	String	Yes	Common parameter. Unique request ID can be generated manually. It is recommended to use uuid. Locating issues requires providing the RequestId for the request.
Action	String	Yes	Common parameter. This interface takes values: AppKugouUserCommand.
KGCommand	String	Yes	Subcommand. This interface takes values: user_info.
DeviceId	String	Yes	Device Id consists of product Id and device Name.
KugouParams	Object of KugouParams_user_info	Yes	Parameters of the subcommand.

3. Output Parameters

Name	Type	Description
RequestId	String	Common parameter. Unique request ID, same as the input parameter. Locating issues

		requires providing the RequestId for the request.
ErrorCode	Integer	The error code returned by the request, 0 means no error.
ErrorMsg	String	The returned error information of the request.
Data	Object of Data_user_info	Returned result

4. Example

Sample 1

Input Example

```
POST https://iot.cloud.tencent.com/api/exploreropen/tokenapi HTTP/1.1
content-type: application/json
{
  "RequestId": "test0191119-012345",
  "Action": "AppKugouUserCommand",
  "AccessToken": "8b4a70dd16105f*****18edd4e78a3bb8ec",
  "DeviceId": "LAEG4YJE1A/dev01",
  "KGCommand": "user_info",
  "KugouParams": { ... }
}
```

Output example: succeed

```
{
  "Response": {
    "Data": { ... },
    "ErrorCode": 0, "ErrorMsg": "",
    "RequestId": "keyi20191119-012345"
  }
}
```

5. Error Code

Error Code	Description
InternalServerError	Internal error.
InvalidParameterValue	Invalid parameter value.
InvalidParameterValue.InvalidAccessToken	The Token is invalid.
InvalidParameterValue.InternalRPCError	Call timeout.
ResourceNotFound	Resource does not exist.
ResourceNotFound.DeviceNotExist	The device is not created or has been deleted.
ResourceNotFound.StudioProductNotExist	The product is not created or has been deleted.
ResourceNotFound.AppNotExists	The application is not created or has been deleted.
UnsupportedOperation	Unsupported operation.
UnsupportedOperation.DeviceNotGateway	Not a gateway device.
UnsupportedOperation.NotBindVirtualDevice	Unable to bind virtual device.
UnsupportedOperation.DeviceNotBind	The subdevice is not bound to the gateway device.
UnsupportedOperation.CannotBindSameFamily	The device has been bound. Do not repeat binding.
LimitExceeded	Exceed the limit.
LimitExceeded.FamilyDeviceExceedLimit	The device quantity exceeds the limit.
UnauthorizedOperation	No permission to operate.
UnauthorizedOperation.APPNoPermissionToStudioProduct	The App has no permission to operate this product.