

# 联网图像搜索

## 快速入门



腾讯云

## 【版权声明】

©2013-2026 腾讯云版权所有

本文档（含所有文字、数据、图片等内容）完整的著作权归腾讯云计算（北京）有限责任公司单独所有，未经腾讯云事先明确书面许可，任何主体不得以任何形式复制、修改、使用、抄袭、传播本文档全部或部分内容。前述行为构成对腾讯云著作权的侵犯，腾讯云将依法采取措施追究法律责任。

## 【商标声明】



及其它腾讯云服务相关的商标均为腾讯云计算（北京）有限责任公司及其关联公司所有。本文档涉及的第三方主体的商标，依法由权利人所有。未经腾讯云及有关权利人书面许可，任何主体不得以任何方式对前述商标进行使用、复制、修改、传播、抄录等行为，否则将构成对腾讯云及有关权利人商标权的侵犯，腾讯云将依法采取措施追究法律责任。

## 【服务声明】

本文档意在向您介绍腾讯云全部或部分产品、服务的当时的相关概况，部分产品、服务的内容可能不时有所调整。您所购买的腾讯云产品、服务的种类、服务标准等应由您与腾讯云之间的商业合同约定，除非双方另有约定，否则，腾讯云对本文档内容不做任何明示或默示的承诺或保证。

## 【联系我们】

我们致力于为您提供个性化的售前购买咨询服务，及相应的技术售后服务，任何问题请联系 4009100100 或 95716。

# 快速入门

最近更新时间：2026-01-05 14:47:51

本文为您介绍初次使用联网图像搜索需要实施的准备工作及入门操作。

操作步骤	说明
<b>准备工作：注册与登录</b>	注册腾讯云账号，完成实名认证并登录
<b>步骤一：开通服务</b>	登录联网图像搜索控制台并开通服务
<b>步骤二：获取云 API 密钥</b>	获取云 API 密钥的 SecretId 和 SecretKey
<b>步骤三：接入 API</b>	根据 API 文档接入联网图像搜索 API

## 准备工作

注册并通过个人实名认证或企业认证后，登录 [腾讯云](#)。如果没有账号，请参考 [注册腾讯云](#)。

## 操作步骤

### 步骤一：开通服务

进入 [联网图像搜索控制台](#) 概览页面，单击开通服务，按照 [购买方式](#) 文档指引按需完成服务开通和购买。

### 步骤二：获取云 API 密钥

1. 前往腾讯云官网控制台 [API 密钥管理](#) 界面获取密钥。

## 2. 单击新建密钥，弹窗获取 SecretId 和 SecretKey，可单击下载 CSV 文件保存至本地。

### ⚠ 注意：

为降低密钥泄漏的风险，自2023年11月30日起，新建的密钥只在创建时提供 SecretKey，后续不可再进行查询，请保存好 SecretKey。

## 步骤三：接入 API

请根据接入指南，参考 [API 文档](#) 示例调用对应的接口，即可开始使用。

## 通过 API 3.0 Explorer 在线调用

**适用对象：**开发初学者，有代码编写基础人员。

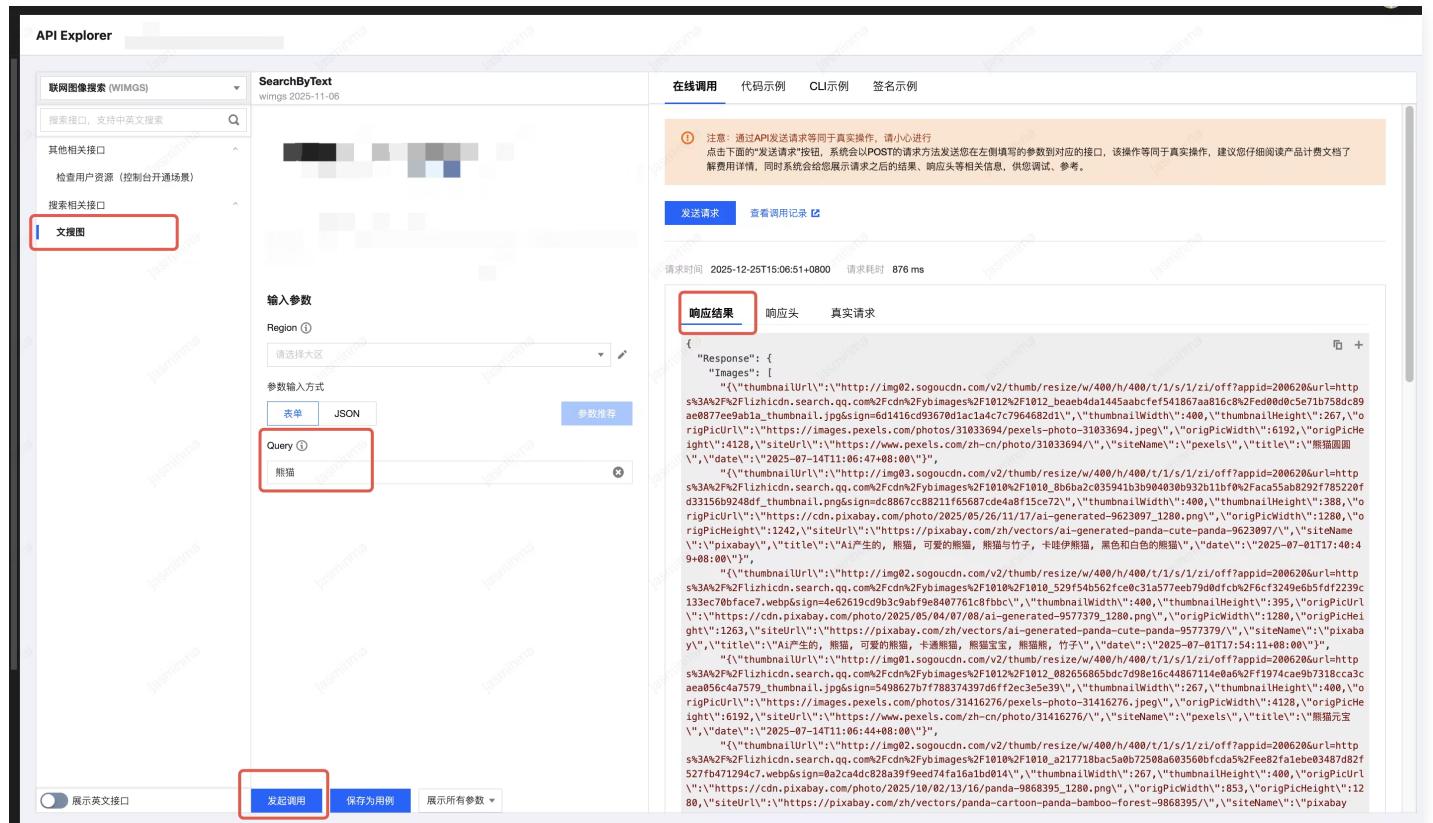
**说明:** 此方式能够实现在线调用、签名验证、SDK 代码生成和快速检索接口等能力。

登录 [腾讯云 API Explorer](#)，搜索 [联网图像搜索 \(WIMGS\)](#)，选择需要调用的 API 进入调试页面。填写输入参数后，单击发起调用，在响应结果中查看请求结果。

输入参数在 API 3.0 Explorer 界面的“文档说明”选项卡中可以查看对应接口输入参数的具体含义。

**说明:**

平台将对登录用户提供临时 Access Key，以便进行调试。



## 通过编写代码，调用 API 进行开发

**适用对象：**开发工程师，熟悉代码编写人员。

**说明:** 腾讯云已编写好的开发工具集 (SDK)，支持通过调用联网图像搜索 API 开发功能。目前 SDK 已支持多种语言，包括 Python、Java、PHP、Go、Node.js、.Net 等，可在每个服务的文档中下载对应的 SDK。

进入 [腾讯云 API Explorer](#) 页面，选择需要调用的接口，在右侧选项卡中选择“代码示例”，选择接入方式和开发语言，并填写参数，即可得到示例代码。

API Explorer

联网图像搜索 (WIMGS) SearchByText wings 2025-11-06

在线调用 代码示例 CLU示例 签名示例

接入方式 SDK SDK Common Client HTTP Request

开发语言 Golang Python Java C++ Node.js PHP .Net

// 该接口只在官网展示, 因此github上的SDK不包含该接口, 若需要使用该示例代码  
// 请从云API部署平台(包含不展示接口)或工蜂(包含公内部展示接口)  
// https://git.woa.com/groups/tencentcloud-internal中下载一份SDK加入到项目中  
package com.tencent;  
import com.tencentcloudapi.common.AbstractModel;  
  
import com.tencentcloudapi.common.Credential;  
import com.tencentcloudapi.common.profile.ClientProfile;  
import com.tencentcloudapi.common.profile.HttpProfile;  
import com.tencentcloudapi.common.exception.TencentCloudSDKException;  
import com.tencentcloudapi.wimg.v20251106.WimgClient;  
import com.tencentcloudapi.wimg.v20251106.models.\*;  
  
public class Sample  
{  
 public static void main(String [] args) {  
 try{  
 // 密钥信息从环境变量读取, 需要提前在环境变量中设置 TENCENTCLOUD\_SECRET\_ID 和 TENCENTCLOUD\_SECRET\_KEY  
 // 使用环境变量方式可以避免密钥硬编码在代码中, 提高安全性  
 // 生产环境建议使用更安全的密钥管理方案, 如密钥管理系统(KMS), 密钥密钥注入等  
 // 请参考: https://cloud.tencent.com/document/product/1278/85385  
 // 密钥可通过官网控制台 https://console.cloud.tencent.com/cam/capi 进行移除  
 Credential cred = new Credential(System.getenv("TENCENTCLOUD\_SECRET\_ID"), System.getenv("TENCENTCLOUD\_SECRET\_KEY"));  
 // 使用临时密钥示例  
 // Credential cred = new Credential("SecretId", "SecretKey", "Token");  
 // 实例化一个HTTP请求, 可选的, 没有会将请求可以跳过  
 HttpProfile httpProfile = new HttpProfile();  
 // 推荐使用化配置, 相关指引访问如下链接  
 // https://git.woa.com/tencentcloud-internal/tencentcloud-sdk-java/E9%C9%7E6%9E81E6%98%9F  
 httpProfile.setEndpoint("wimg.tencentcloudapi.com");  
 // 实例化一个Client选项, 可选的, 没有特需求可以跳过  
 ClientProfile clientProfile = new ClientProfile();  
 clientProfile.setHttpProfile(httpProfile);  
 // 客户端主要通过ClientProfile来设置的  
 WimgClient client = new WimgClient(cred, "", clientProfile);  
 // 实例化一个请求对象, 每个接口都有对应的request对象  
 SearchByTextRequest req = new SearchByTextRequest();  
 req.setQuery("熊猫");  
 // 返回一个resp是每一个searchByTextResponse的实例, 与请求对象对应  
 SearchByTextResponse resp = client.SearchByText(req);  
 // 输出JSON格式的字符串回包  
 }  
 }  
}

展示英文接口 发起调用 保存为用例 展示所有参数

下面以 Python 代码为例, 说明 API 本地调用的方法。

## 1. 安装 腾讯云 SDK 对应版本代码。

```
pip install tencentcloud-sdk-python
```

## SDK 中心

欢迎使用腾讯云开发者工具套件 (SDK), SDK 提供多种语言版本, 统一了接口调用方式, 规范了错误码以及返回包格式, 方便您快速接入和使用腾讯云服务。

请输入产品名称

热门产品: 即时通信 IM 轻量应用服务器 云服务器 对象存储 实时音视频

全部产品

分类: 全部 Java Python Go PHP Node.js .Net C++ Ruby iOS Android Others

计算

Tencent Cloud SDK for Python API3.0

最新版本: 3.0.1317

最后更新: 2025-02-12 15:51:11

相关链接: [SDK仓库](#) | [帮助文档](#) | [API Explorer](#)

```
pip install tencentcloud-sdk-python
```

## 2. 按如下示例代码调用接口:

仅需修改如下示例代码中: **TENCENTCLOUD\_SECRET\_ID, TENCENTCLOUD\_SECRET\_KEY** 密钥信息, 以及对应的接口入参。

```
# -*- coding: utf-8 -*-
import os
import json
import types
from tencentcloud.common import credential
from tencentcloud.common.profile.client_profile import ClientProfile
from tencentcloud.common.profile.http_profile import HttpProfile
from tencentcloud.common.exception.tencent_cloud_sdk_exception import
TencentCloudSDKException
from tencentcloud.wimgs.v20251106 import wimgs_client, models
try:
```

**# 密钥信息从环境变量读取，需要提前在环境变量中设置 TENCENTCLOUD\_SECRET\_ID 和 TENCENTCLOUD\_SECRET\_KEY**

**# 使用环境变量方式可以避免密钥硬编码在代码中，提高安全性**  
**# 生产环境建议使用更安全的密钥管理方案，如密钥管理系统 (KMS)、容器密钥注入等**  
**# 请参见: <https://cloud.tencent.com/document/product/1278/85305>**  
**# 密钥可前往官网控制台 <https://console.cloud.tencent.com/cam/capi> 进行**

**获取**

```
cred = credential.Credential(os.getenv("TENCENTCLOUD_SECRET_ID"),
os.getenv("TENCENTCLOUD_SECRET_KEY"))
```

**# 使用临时密钥示例**

```
# cred = credential.Credential("SecretId", "SecretKey", "Token")
```

**# 实例化一个http选项，可选的，没有特殊需求可以跳过**

```
httpProfile = HttpProfile()
```

```
httpProfile.endpoint = "wimgs.tencentcloudapi.woa.com"
```

**# 实例化一个client选项，可选的，没有特殊需求可以跳过**

```
clientProfile = ClientProfile()
```

```
clientProfile.httpProfile = httpProfile
```

**# 实例化要请求产品的client对象，clientProfile是可选的**

```
client = wimgs_client.WimgsClient(cred, "", clientProfile)
```

**# 实例化一个请求对象，每个接口都会对应一个request对象**

```
req = models.SearchByTextRequest()
```

```
params = {
```

**"Query": "北京故宫"**

```
}
```

```
req.from_json_string(json.dumps(params))
```

```
# 返回的resp是一个SearchByTextResponse的实例，与请求对象对应
resp = client.SearchByText(req)
# 输出json格式的字符串回包
print(resp.to_json_string())

except TencentCloudSDKException as err:
    print(err)
```