

识图精灵





1. 探索新知

1.1

准备工作

想要实现今天的效果，将用到【百度智能云】平台通用物体和场景识别高级版的接口



1. 注册百度智能云账号



给大家介绍准备工作的详细过程

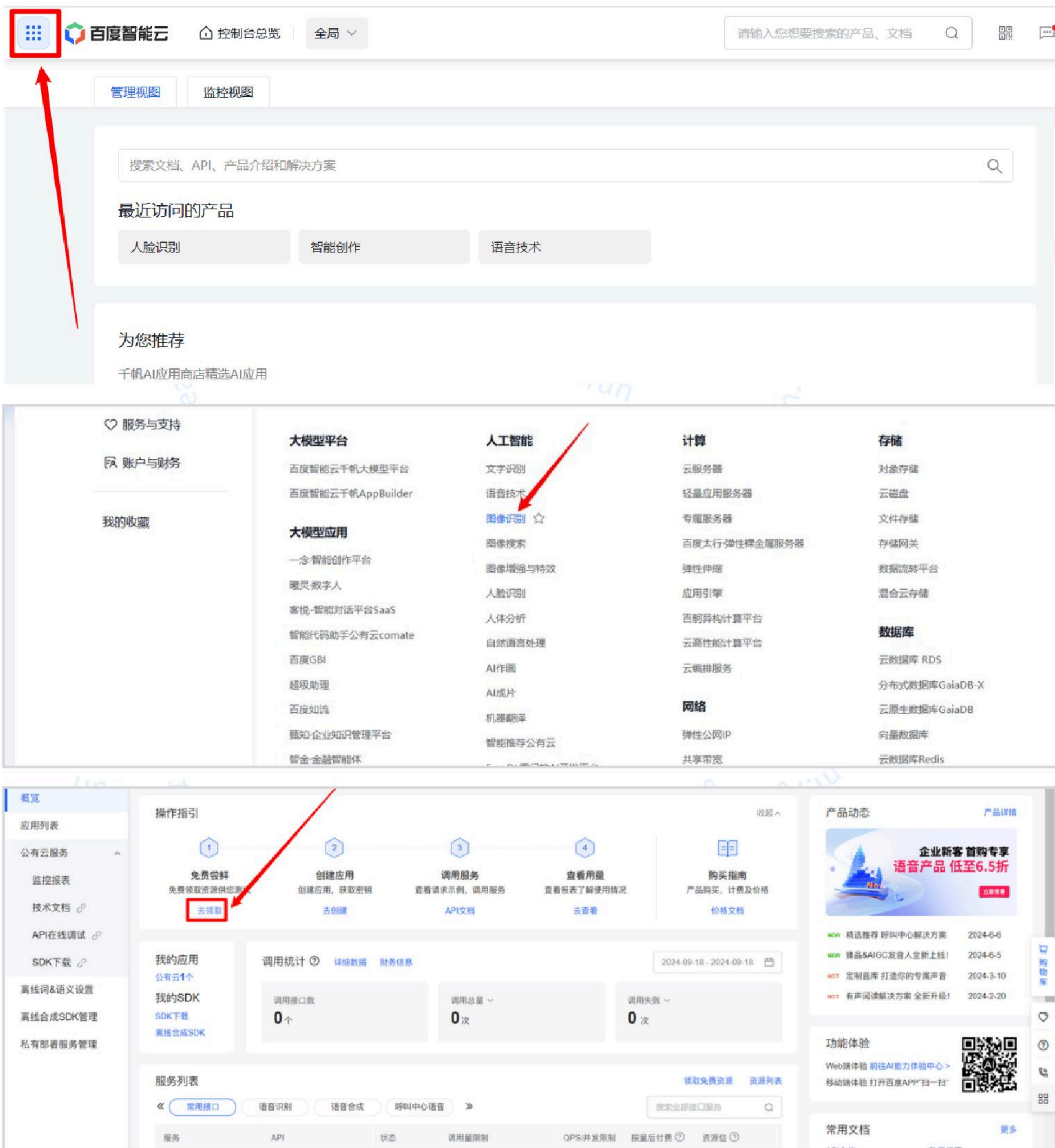


2. 打开产品—【图像识别】



1.1

准备工作



3. 领取免费接口

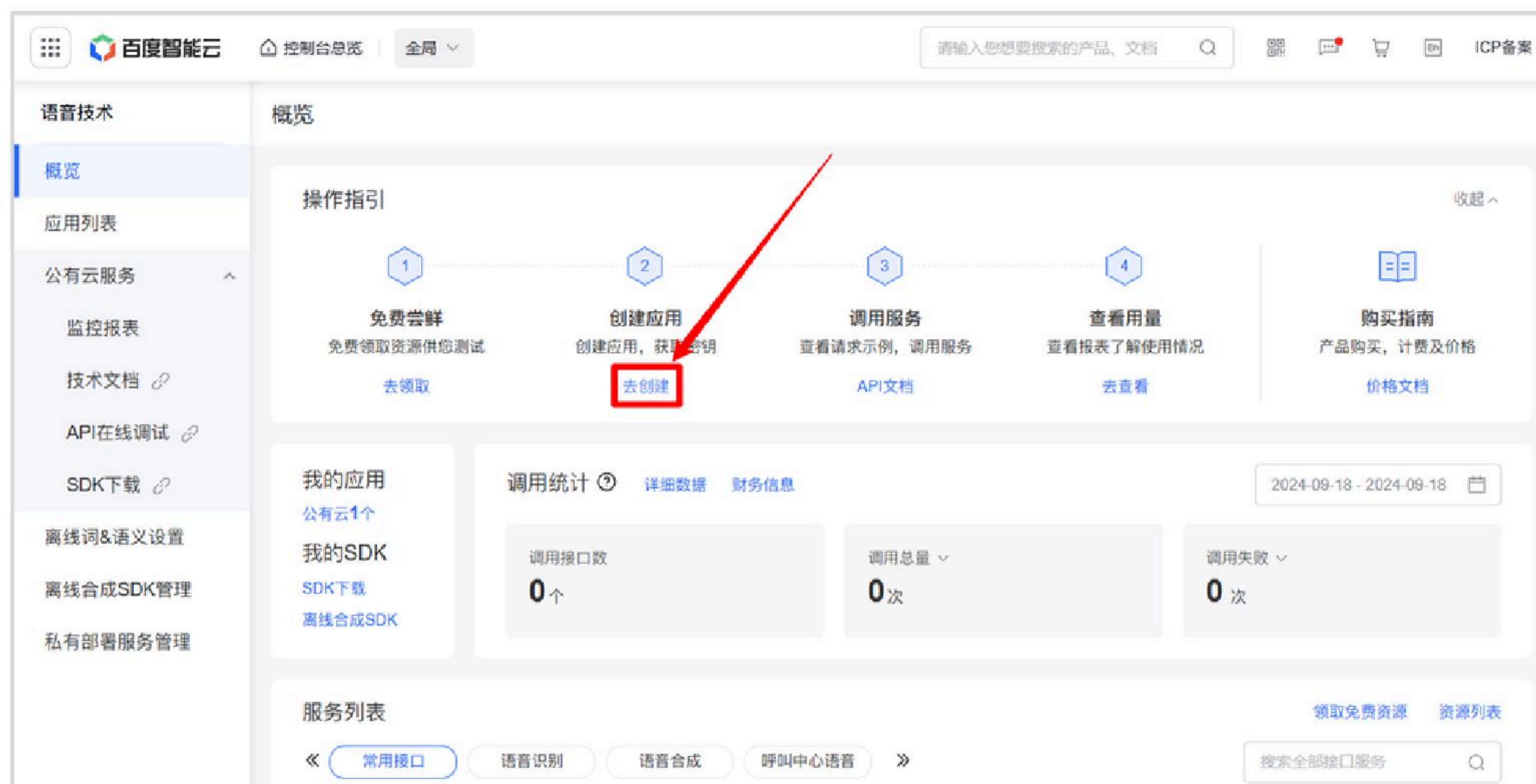


我已经领取过接口，所以接口会在【已领接口】中。大家在操作时，至少选择这个接口，也可以直接领取【全部】接口，点击左下方的【0元领取】，即可领取试用（试用时长180天）。

1.1

准备工作

4. 创建应用



注意:
在创建应用之前，
请确认自己已成功
领取免费测试资源

根据业务需求，填写如下应用信息：

The screenshot shows the application creation form. The '应用名称' (Application Name) field contains '图像识别' (Image Recognition). The '接口选择' (Interface Selection) section has '通用物体和场景识别高级版' (General Object and Scene Recognition Advanced Edition) checked. The '应用归属' (Application Belonging) section has '个人' (Personal) selected. The '应用描述' (Application Description) field contains '图像识别'. There are '立即创建' (Create Immediately) and '取消' (Cancel) buttons at the bottom.

【应用名称】 必填项，所创建的应用名称，支持中英文、数字、下划线及中横线

【接口选择】 勾选该应用需要用到的接口

【应用归属】 必选项，选择【个人】即可

【应用描述】 必填项，对此应用的业务场景进行简单描述

5. 获得应用的AK/SK



如果密码被别人知道了，
记得及时【管理】-
【更新Secret Key】

1.2

图像识别

• 安装需要的第三方库



```
代码区 安装第三方库.py
1 # 点击运行【安装第三方库.py】，即可安装本课用到的第三方库
2
3 import pip
4 pip.main(['install','baidu-aip'])
5 pip.main(['install','chardet'])
```

1. 【baidu-aip】 百度人工智能开放平台提供的Python SDK（软件开发工具包），用于连接并调用百度AI开放平台的各种API接口
2. 【chardet】 用于检测文本文件或字符串的字符编码的Python库

1. 引入所需要的库

```
from aip import AipImageClassify
```

AipImageClassify



这行代码告诉电脑：
“请帮我把 AipImageClassify
这个工具拿过来，我要用。”



2. 定义常量

```
APP_ID = '你的 AppID '  
API_KEY = '你的 API Key '  
SECRET_KEY = '你的 Secret Key '
```

这里，我们告诉电脑一些重要的信息（称为常量），让我们可以使用百度提供的图像识别服务。这些信息是从百度智能云平台获取的。

（这三个信息是应用开发中的重要凭证，请谨慎保管）

3. 准备【图像识别器】

```
client = AipImageClassify (APP_ID, API_KEY, SECRET_KEY)
```

1.2

图像识别

这行代码我们准备好一个叫 `client` 的工具，这个工具就像一台机器，它通过前面设好的信息 (`APP_ID`, `API_KEY`, `SECRET_KEY`) 来工作，我们可以用它来与百度的图像识别服务进行沟通。

• 为什么要起名为 `client` ？

`client` 是一个常见的命名方式，表示某种服务接口的客户端对象。在这儿我们就称呼 `client` 为【图像识别器】吧。



准备好了【图像识别器】，接下来我们开始编程，传入需要识别的图片，并返回识别结果

4. 主函数

```
# 主函数
图片数据 = 读图片("花.png")
识别结果 = 识别图片(图片数据)
print(f"API返回结果: {识别结果}\n")
打印结果(识别结果)
print('识别完毕')
```

```
# 主函数
data = read_img("花.png")
result = find_img(data)
print(f"API返回结果: {result}\n")
print_result(result)
print('识别完毕')
```

1.2

图像识别

1. `data = read_img("花.png")`: 我们定义 `read_img` 函数来读取图片数据。
2. `result = find_img(data)`: 我们定义 `find_img` 函数来识别图片内容，并把结果保存到 `result` 中。
3. `print(f"API返回结果: {result}\n")`: 打印出从百度返回的原始识别结果。
4. `print_result(result)`: 我们定义 `print_result` 函数来打印具体的识别内容和相似度。
5. `print('识别完毕')`: 最后打印一个确认信息，告诉我们识别过程结束了。

```
# 读取图片文件并返回二进制数据
def read_img(file):
    pass
# 调用【图像识别器】并返回结果
def find_img(img):
    pass
# 打印识别结果
def print_result(res):
    pass
```

为避免程序报错，主函数要用到的函数，先定义为空函数，后面再挨个编写。

5. 读取图片文件并返回二进制数据

```
# 读取图片文件并返回二进制数据
def read_img(file):
    with open(file, "rb") as f:
        return f.read()
```

这里定义了一个函数 `read_img`。函数就像是一个小机器，它可以帮我们完成一些特定的任务。这个小机器的任务是读取一张图片并把它转成一种叫做“二进制”的数据形式，计算机可以更好地理解这种数据。

6. 调用【图像识别器】并返回结果

```
# 调用【图像识别器】并返回结果
def find_img(img):
    # 返回的信息是字典类型，结果保存在键 'result' 的值中
    return client.advancedGeneral(img)
```

这里定义了另一个函数 `find_img`。这个小机器的任务是使用 `client` 对象中的 `advancedGeneral` 方法来识别图片内容，并返回识别结果。这个识别结果是一个字典，结果保存在键“result”的值中。

7. 打印识别结果

```
# 打印识别结果
def print_result(res):
    if 'result' in res:
        for i in res['result']:
            if i['score'] >= 0.6:
                print(f"结果为 {i['keyword']} ，相似度为 {i['score']*100:.2f}% ")
    else:
        # 失败响应提示信息
        print(f"识别失败，错误码：{res.get('error_code')}，错误信息：{res.get('error_msg')}")
```

- 这里定义的函数 `print_result` 会检查百度返回的识别结果。
- 如果识别成功，它会打印出相似度高于60%的识别的内容（keyword）和相似度（score）。
- 如果识别失败，它会打印出错误信息。
- 相似度是一种度量，告诉我们识别出来的内容有多准确。

完整代码

```
# 调用百度智能模块中图像识别类方法
from aip import AipImageClassify
# 填写个人的授权使用信息
APP_ID = '你的 AppID '
API_KEY = '你的 API Key '
SECRET_KEY = '你的 Secret Key '
# 准备【图像识别器】
client = AipImageClassify(APP_ID, API_KEY, SECRET_KEY)

# 读取图片文件并返回二进制数据
def read_img(file):
    with open(file, "rb") as f:
        return f.read()
# 调用【图像识别器】并返回结果
def find_img(img):
    return client.advancedGeneral(img)
# 打印识别结果
def print_result(res):
    if 'result' in res:
        for i in res['result']:
            if i['score'] >= 0.6:
                print(f"结果为 {i['keyword']} , 相似度为{i['score']*100:.2f}% ")
    else:
        # 失败响应提示信息
        print(f"识别失败, 错误码: {res.get('error_code')},
              错误信息: {res.get('error_msg')}")

# 主函数
data = read_img("花.png")
result = find_img(data)
print(f"API返回结果: {result}\n")
print_result(result)
print('识别完毕')
```



2. 强化练习

1. 如何初始化 `AipImageClassify` 对象? ()

- A. `client = AipImageClassify(APP_ID, API_KEY)`
- B. `client = AipImageClassify(APP_ID)`
- C. `client = AipImageClassify(APP_ID, API_KEY, SECRET_KEY)`
- D. `AipImageClassify(APP_ID, API_KEY, SECRET_KEY)`

2. 哪一行代码用于读取图片文件并返回二进制数据? ()

- A. `open(file, "rb" as f)`
- B. `open(file, "rb") as f`
- C. `with open(file, "rb") as f:`
- D. `with file(file, "rb") as f:`

3. 如何判断图像识别结果是否成功? ()

- A. 检查返回结果是否为空
- B. 检查返回结果中是否有"result"键
- C. 检查返回结果是否有数字
- D. 检查返回结果的长度



2. 强化练习

4. AipImageClassify 是什么? ()

- A. 一个图像处理库
- B. 一种编程语言
- C. 百度智能模块中图像识别类
- D. 一种文件格式

5. 执行下面Python代码后, 文件ip.txt中的内容用记事本打开时显示为? ()

```
with open("ip.txt", "w") as f:  
    lst = ["202.206.224.21\n", "192.168.224.1\n"]  
    f.writelines(lst)
```

- A. 202.206.224.21192.168.224.1
- B. 202.206.224.21
192.168.224.1
- C. 202.206.224.21\n192.168.224.1
- D. 202.206.224.21\n192.168.224.1\n

3.术语箱

client (计算机) 客户端

4.课后挑战

行走的百科全书

任务描述:

从【图库】中插入以下图片，尝试用本课做的【识图精灵】识别图片中的物体

