

实训五 使用云服务

一、实训目的

- 1、了解微信小程序云开发云函数；
- 2、了解国内知名公有云的云服务；
- 2、掌握微信小程序云开发云函数的使用；
- 3、掌握国内知名公有云的云服务的使用。

二、实训学时

4 学时

三、实训类型

综合型

四、实训需求

1、硬件

每组配备计算机 6 台，不低于双核 CPU、8G 内存、500GB 硬盘，开启硬件虚拟化支持。

2、软件

操作系统：Windows 10；

应用软件：微信开发者工具、Node.js。

3、网络

计算机使用固定 IP 地址接入局域网，并支持对互联网的访问。

4、素材

无。

五、实训任务

- 1、完成云函数的开发与调用；
- 2、完成腾讯云-人工智能-自然语言处理 NLP-关键词提取的对接开发；
- 3、完成腾讯云-人工智能-图像分析-图像标签的对接开发；
- 4、完成腾讯云-人工智能-语音识别 ASR-实时语音识别的对接开发。

六、实训考核

- 1、提交《互联网应用开发实训-实训五-年级-专业-学号-姓名》成果截图。

七、实训内容及步骤

1、安装 Node.js

(1) 获取 Node.js 的安装程序，可通过本课程网站 (<http://wzxsx.g.hactcm.edu.cn>) 下载获得，或者访问官网 <https://nodejs.org> 获取。

(2) 双击 Node.js 安装程序，按照软件安装提示，依次点击【Next】，完成安装。

2、创建云函数

(1) 创建项目

打开微信开发者工具，创建云开发项目。

(2) 配置云开发本地目录

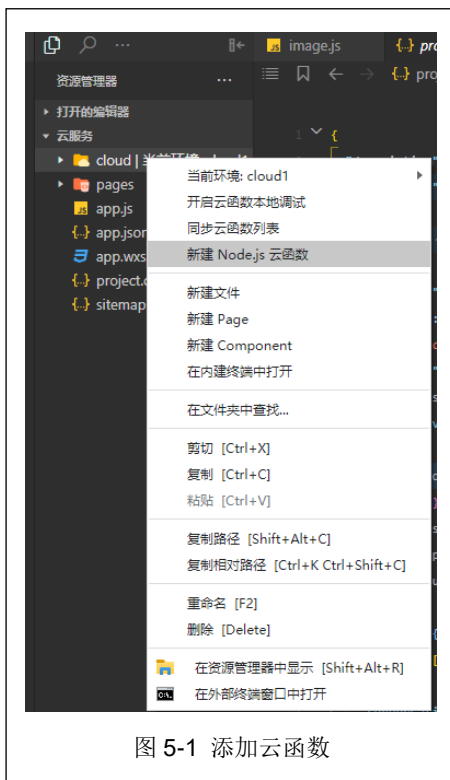
在创建的小程序项目中，创建“cloud”文件夹。

在小程序根目录下找到并打开 `project.config.json` 文件，在内容的根节点下新增“`cloudfunctionRoot`”字段，指定本地已存在的目录作为云开发的本地根目录。配置参数如下。

```
"cloudfunctionRoot": "cloud/",
```

(3) 创建云函数

完成指定之后，云开发根目录的图标会变成“云开发图标”，云函数根目录下的第一级目录（云函数目录）是与云函数名字相同的。在云函数根目录上右键，在右键菜单中，选择“新建 Node.js 云函数”，将该云函数命名为 `add`，如图 5-1 所示。



双击打开云函数的“`index.js`”文件，可以看到云函数的入口函数处传入参数有两个，一个是 `event` 对象，一个是 `context` 对象。

`event` 指的是触发云函数的事件，当小程序端调用云函数时，`event` 就是小程序端调用云函数时传入的参数。

`context` 对象包含了此处调用的调用信息和运行状态，可以用来了解服务运行的情况。

在模板中也默认 `require` 了 `wx-server-sdk`，这是一个帮助我们在云函数中操作数据库、

存储以及调用其他云函数的微信提供的库。

返回参数值有 event、小程序用户的 openid、小程序的 appid 以及小程序用户的 unionid。修改一下返回值，让其进行一次简单的计算，修改代码如下。

```
exports.main = async (event, context) => {  
  // ...  
  return {  
    sum: event.a + event.b //将传递的参数相加，并返回相加后结果  
  }  
}
```

在云函数中引入第三方依赖可以帮助我们更快的开发。云函数的运行环境是 Node.js，因此可以使用 npm 安装第三方依赖，其使用方法为在云函数目录上右键，在选项中选择“在外部终端窗口中打开”，即可在打开的终端上进行 npm 操作。

云函数调整完毕之后，需将其部署到云端，具体操作方法为：在云函数目录上右键，在选项中选择“上传并部署”，如图 5-2 所示。

(4) 调用云函数

在微信小程序中，可以使用下列代码调用云函数。

```
//初始化  
wx.cloud.init();  
//调用云函数  
wx.cloud.callFunction({  
  // 云函数名称  
  name: 'add',  
  // 传给云函数的参数  
  data: {  
    a: 1,  
    b:10  
  },  
  //调用成功触发函数  
  success: function (res) {  
    //在小程序中回显结果  
    that.setData({  
      result:res.result.sum  
    })  
  },  
  //调用失败触发函数  
  fail: console.error  
})
```

程序执行完毕，将出现计算后的结果“11”。

3、使用腾讯云自然语言处理 NLP-关键词提取功能

(1) 访问腾讯云

在浏览器中访问腾讯云，地址为：<https://cloud.tencent.com>，鼠标悬停“产品”->“人工智能”->“自然语言处理”，然后点击“自然语言处理”，如图 5-3 所示。出现产品介绍页面，点击【免费体验】进入控制台（如未登录，需使用手机微信扫码登录）。



图 5-3 访问腾讯云

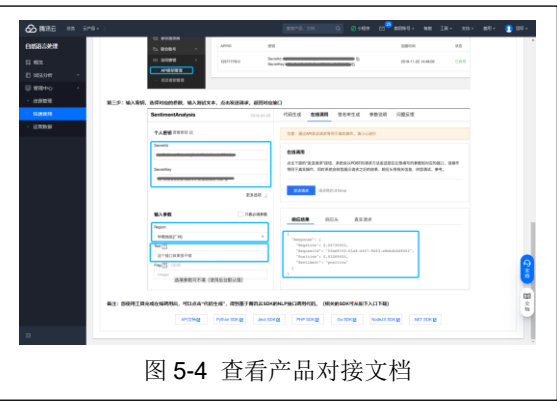


图 5-4 查看产品对接文档

(2) 查看产品文档

在“快速使用”页面，滑动至最底部，点击【API 文档】即可查看“自然语言处理”的 API 接口文档，如图 5-4 所示。

(3) 查看 SDK 使用方法

微信小程序云函数是基于 Node.js 部署的，因此其应使用 NodeJS SDK。

在“快速使用”页面，滑动至最底部，点击【NodeJS SDK】即可查看 SDK 的使用方法。

(4) 获取访问密钥

在“快速使用”页面，点击页面顶部的“云产品”一直向下滚动，找到“管理与审计”项，然后点击“访问密钥”，如图 5-5 所示。

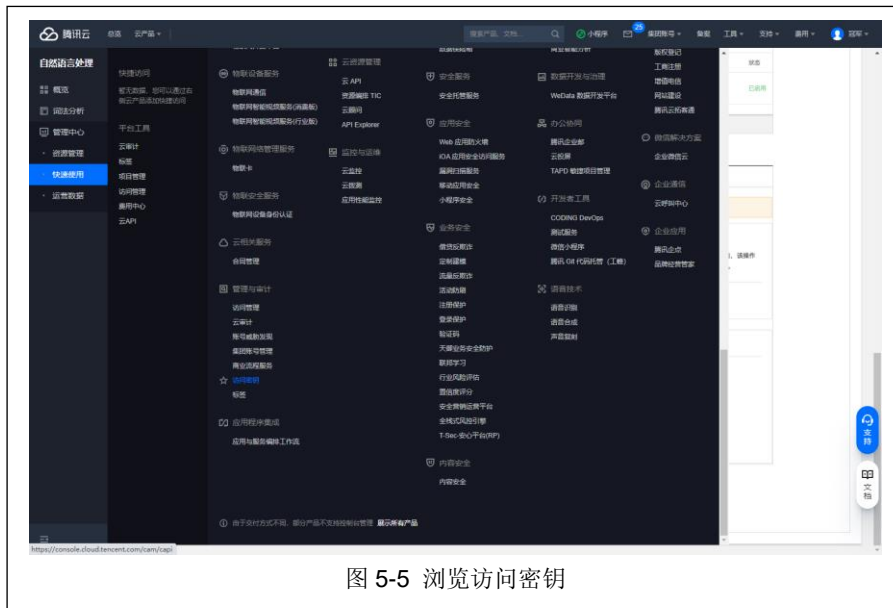


图 5-5 浏览访问密钥

如未创建密钥，在此页面点击【新建密钥】，即可自动生成访问密钥，保存好 APPID、SecretId、SecretKey，在之后的腾讯云 API 开发中都需用到此数据。

(5) 创建云函数

打开创建的微信小程序云开发项目，按照之前的步骤创建云函数，创建完成之后，找到云函数目录下的“package.json”文件，在“dependencies”节点下增加“request”节点（添加“request”依赖包），代码如下。

```
"dependencies": {
  "request": "^2.85.0",
```

```
"wx-server-sdk": "~2.5.3"
}
```

在创建的云函数上右键，选择“在外部终端窗口中打开”，在打开的终端上通过 npm 安装腾讯云 API SDK，命令如下。

```
npm install tencentcloud-sdk-nodejs --save
```

安装完毕之后，再次输入以下命令，进行 npm 包的自动安装。

```
npm install
```

安装完成之后，打开云函数下的“index.js”文件，进行代码编码，此时在“API 文档”页面找到“篇章分析相关接口”->“关键词提取”，即可看到接口描述信息，如图 5-6 所示。



图 5-6 查看 API 介绍

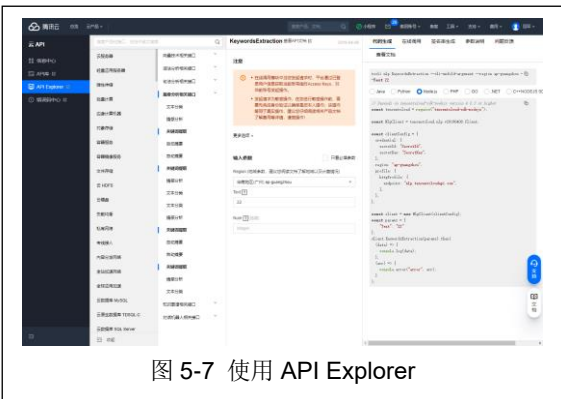


图 5-7 使用 API Explorer

在接口描述部分，点击【</>点击调试】，可跳转至 API Explorer 页面，API Explorer 提供了在线调用、签名验证、SDK 代码生成和快速检索接口等能力，如图 5-7 所示。

点击“代码生成”->“Node.js”，然后选择输入参数，将 API Explorer 中的代码复制到“index.js”文件中。根据代码提示修改对应的参数，并修改返回程序，使云函数能够接收参数并进行处理，返回部分代码如下所示。

```
return new Promise((resolve, reject) => { // 通过 Promise 容器来接收
  异步 API 的回调，然后通过当前脚本返回给客户端
  const params = {
    "Text": event.text
  };
  client.KeywordsExtraction(params).then(
    (data) => {
      resolve(data)
    },
    (err) => {
      resolve(err)
      console.error("error", err)
    }
  )
});
```

云函数至此全部开发完毕，将其上传并部署即可。

(6) 小程序调用云函数

在小程序中新建页面，添加一个文本输入框、一个触发按钮、一个结果展示组件，进行“关键词提取”开发，展示样式如图 5-8 所示。



小程序调用云函数，并传递参数的代码如下所示。

```

var that = this;
//获取输入值
const params = e.detail.value;
//初始化
wx.cloud.init();
wx.cloud.callFunction({
  // 云函数名称
  name: 'keyword',
  // 传给云函数的参数
  data: {
    text: params.words
  },
  success: function (res) {
    //设置结果值
    that.setData({
      result: res.result.Keywords
    })
  },
  fail: console.error
})

```

展示效果如图 5-9 所示。

4、使用腾讯云图像分析-图像标签功能

(1) 访问腾讯云

在浏览器中访问腾讯云，鼠标悬停“产品”->“人工智能”->“图像识别”，然后点击“图像分析”。出现产品介绍页面，点击【Demo 体验】进入体验页面。

(2) 查看接口文档

在体验页面点击【立即使用】，在控制台中点击【接口文档】，然后点击【新窗口查看文

档】，在此页面点击【</>点击调试】，可跳转至 API Explorer 页面，在此页面中设置参数后即可复制代码。

(3) 创建云函数

对比前面的步骤完成云函数的创建，并将 API Explorer 中的代码复制到云函数中，完成代码编写以及相应的参数、环境配置，最后上传部署该云函数。

(4) 小程序调用云函数

在小程序中新建页面，添加一个选择图片按钮、一个选择图片展示框、一个结果展示组件，进行“图像分析-图像标签”的开发，展示样式如图 5-10 所示。

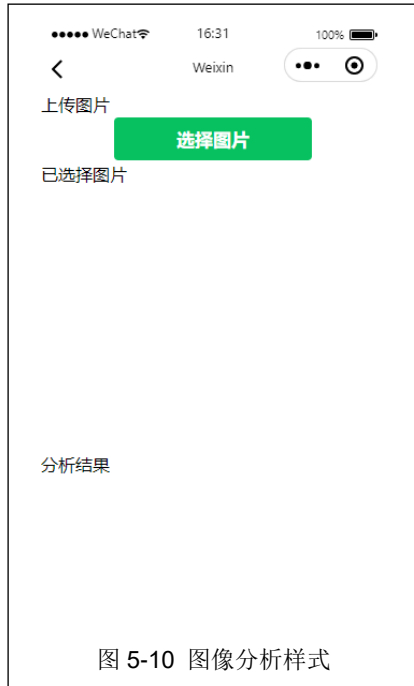


图 5-10 图像分析样式



图 5-11 图像分析结果

小程序调用云函数，并传递选择的图像数据，展示效果如图 5-11 所示。

5、使用腾讯云语音识别 ASR-实时语音识别功能

(1) 访问腾讯云

在浏览器中访问腾讯云，鼠标悬停“产品”->“人工智能”->“语音技术”，然后点击“语音识别”。出现产品介绍页面，点击【立即使用】开通免费试用版。

(2) 查看接入文档

在“概览”页面点击右侧“文档和帮助”中的“实时语音识别接口文档”，之后点击【新窗口查看文档】，然后找到“SDK 文档”->“SDK 概览”，如图 5-12 所示。

(3) 小程序接入

在此可看到接入类型有“小程序”，点击“小程序”对应行后面的“一分钟跑通集成 SDK”，在弹出的新页面中添加“腾讯云智能语音”（需登录微信小程序管理平台之后再添加，或者在微信小程序管理平台的“设置”->“第三方服务”->“插件管理”中，搜索并添加此插件），如图 5-13 所示。

并可在此页面查看“腾讯云智能语音”插件的具体用法，小程序使用“实时语音识别（http）”进行开发，在介绍的最下面有完整的调用示例。



图 5-12 查看实时语音识别 SDK 接入



图 5-13 添加“腾讯云智能语音”插件

(4) 实现效果

最终实现效果为语音实时转换为文字，如图 5-14 所示，调试时微信小程序需要麦克风连接，因此最好使用“真机调试”模式进行调试。

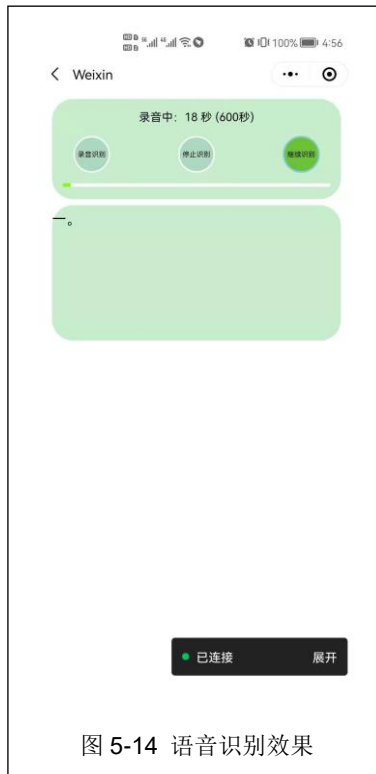


图 5-14 语音识别效果

八、附件

《附件 5-1：使用云服务-开发案例》