

今日内容：接口自动化框架封装思想建立之httprunner框架（下）

@auth：码尚学院_百里老师

@Email：2971330037@qq.com

一、httprunner如何实现数据驱动

3.X开始，使用`parameters`定义数据源。应用于测试用例层。

第一种：直接在脚本里面指定参数列表，最简单。适合于参数比较少的情況。

```
1 config:
2   name: 测试用例
3   parameters:
4     appid-grant_type-secret-assert_str:
5       - ["wx74a8627810cfa308", "client_credential", "e40a02f9d79a8097df497e6aaf93ab80", "access_token"]
6       - ["", "client_credential", "e40a02f9d79a8097df497e6aaf93ab80", "errcode"]
7       - ["wx74a8627810cfa308", "", "e40a02f9d79a8097df497e6aaf93ab80", "errmsg"]
8   teststeps:
9     - name: 获得token鉴权码API
10    api: api/get_token.yml
```

第二种：使用CSV文件，适合于参数比较大的情况。

```
1 config:
2   name: 测试用例
3   parameters:
4     appid-grant_type-secret-assert_str: ${P(data/get_token_data.csv)}
5   teststeps:
6     - name: 获得token鉴权码API
7     api: api/get_token.yml
```

```
1 appid,grant_type,secret,assert_str
2 "wx74a8627810cfa308","client_credential","e40a02f9d79a8097df497e6aaf93ab80","access_token"
3 "", "client_credential", "e40a02f9d79a8097df497e6aaf93ab80", "errcode"
4 "wx74a8627810cfa308", "", "e40a02f9d79a8097df497e6aaf93ab80", "errmsg"
```

注意：

- 1.csv文件中第一行必须放参数名称，并且参数名称必须和测试用例里面的名称一致。
- 2.csv文件中第二行放数据，每一组数据占一行。
- 3.parameters指定的参数顺序可以不一致，个数也可以不一致。

第三种方式：使用函数生成数据，适用于数据变化大的情况。

```
1 #生成数据
2 def get_token_data():
3     return [
4         {"appid": "wx74a8627810cfa308", "grant_type": "client_credential", "secret": "e40a02f9d79a8097df497e6aaf93ab80", "assert_str": "access_token"},
5         {"appid": "", "grant_type": "client_credential", "secret": "e40a02f9d79a8097df497e6aaf93ab80", "assert_str": "errcode"},
6         {"appid": "wx74a8627810cfa308", "grant_type": "", "secret": "e40a02f9d79a8097df497e6aaf93ab80", "assert_str": "errmsg"}
7     ]
```

```
1 config:
2     name: 测试用例
3     parameters:
4         appid-grant_type-secret-assert_str: ${get_token_data()}
5     teststeps:
6         - name: 获得token鉴权码API
7         api: api/get_token.yml
```

二、httprunner文件上传

前提条件：需要安装如下两个包

pip install requests_toolbelt filetype

pip install "httprunner[upload]"

三、如何生成allure报告

第一步

- 1.官网下载allure文件：<https://github.com/allure-framework/allure2/releases>
- 2.下载之后解压到非中文的目录

3.把bin路径配置到系统变量path中：E:\allure-2.13.7\bin （注意分号不要是中文的）

第二步：

安装allure报告：pip install allure-pytest

验证：allure --version

注意：可能需要重启pycharm。

第三步：

1.在reports目录下生成temps目录，并且在temps目录下生产临时的json格式的临时报告。

```
1 os.system("hrun testsuites/test_suites.yml --alluredir=reports/temps --clean-alluredir")
```

加上--clean-alluredir表示：每执行一次把原来的清除。

2.根据临时json报告生成allure报告

```
1 os.system("allure generate reports/temps -o reports/allures --clean")
```

加上--clean表示：每执行一次把原来的清除。

四、接口自动化框架文件对应关系

一个接口对应一个yaml文件，一个yaml文件对应一个csv文件。

优点：相对来说比较成熟，唯一的，不需要写脚本。

缺点：规则太难了，小心翼翼，报错机制不完善。有些功能不齐全。