接口程序说明

版本：1.0

## 引言

* 1. 设计目标

用户通过填写excel表，自动生成接口测试用例，完成测试之后，返回html形式的测试报告

* 1. 适用范围

接口设计返回值的固定形式为

{ “message” : ” ” , “code” : “ ”, “data” : “”}

* 1. 相关术语

## 分析设计

* 1. 设计思想

用户按需求填写EXCEL表后，触发jenkins执行测试程序，测试程序按EXCEL生成测试用例的脚本，Jenkins调用nosetests框架执行测试脚本，生成测试报告发送给用户

* 1. 程序目录说明



* 1. 用户EXCEL表填写规范

该excel表是用户填写的接口数据，通过该excel表生成相应的测试脚本，每个sheet页面是一个py测试脚本，每个脚本里面的测试函数就是该sheet页面每一行数据。按照惯例，通常我们将一个业务相关的接口数据放在同一个sheet页面。Sheet页的名字必须是英文字母，不能使用中文

2.3.1 选项填写说明

1.执行步骤

该选项的值是自然序列（1,2,3,4.....），通过该序列决定每个case的先后执行顺序，序列为1的该接口比序列为2的接口先执行，依次类推。若该选项不填（即为空）,则表示该接口不执行

2.用例名称:

该接口的说明，用于在测试报告的直观显示

3.访问地址:

接口的URL地址

4.调用方法:

http的请求方式，分为POST, GET, DELETE, PUT等

5.数据类型:

服务端允许接受的数据类型，目前只实现application/json的数据格式

6.数据内容:

需要发送的数据，若数据类型是application/json，则该项填写json格式的数据，需要注意的是当请求方式为GET时，并且带有条件查询，则需要将查询条件也以json的格式填写在此项，注意此项可以发送带参数的变量，变量的定义第9个选项说明，建议将使用到的变量用红色字体标识，注意变量未使用双引号包围，例如：

{

"name": "接口普通会员8231",

"gender": 2,

"dept": kechuang3,

"role": kechuang8,

"data\_access": [

kechuang3

]

}

7.返回结果:

该接口的返回值

8.获取授权字段

登录接口返回的access-token字段，用于接口授权

9.变量对应关系表

接口返回的结果如果在后面的接口中需要用到该项数据，则可以在此项结果中定义变量，变量的定义必须以kechuang开头，建议将变量用红色标识出来，注意变量使用双引号包围，例如：

{

"code": 1000,

"message": "查询成功",

"data": {

"total": 5,

"role\_list": [

{

"id": "kechuang4",

"role\_name": "工会主席",

"role\_description": "工会主席描述",

"role\_slug": "dept\_master",

"deleted": false

},

{

"id": "kechuang8",

"role\_name": "普通会员",

"role\_description": "无",

"role\_slug": "common\_member",

"deleted": false

}

]

}

}

10. 是否比较返回值

默认为空，除非用户确定每次调用的该接口，返回结果的值都是一模一样

* 1. 测试流程分析

Jekins检查项目工程中的源码是否有变更，有变更就触发该项目的运行，/public/generate.py会根据excel表生成对应的测试脚本, 假设为/testcase/test\_1.py , test\_2.py。nosetests会执行这些文件中的测试函数。

并且生成对应的测试报告/report/report.html

## 程序函数说明

* 1. generate\_case.py

|  |  |  |  |
| --- | --- | --- | --- |
| 函数名 | 功能说明 | 参数 | 返回值 |
| Generate\_func | 在测试脚本里面生成测试函数 | My\_dict： 字典结构，为excel表中的每一行  Index：执行步骤  File\_name：测试脚本 | 无 |
| convert\_values | 将字符型数字转化成数字 | Value：待转字符 | 数字 |
| generate\_test\_case | 将excel表中数据根据执行步骤拍讯 | 无 | 无 |

* 1. explainExcel.py

|  |  |  |  |
| --- | --- | --- | --- |
| 函数名 | 功能说明 | 参数 | 返回值 |
| Read\_excel | 读取excel的表单和名字 | File\_path： excel表的文件路径 | （workbook, sheetname） |
| Read\_sheet | 读取每个sheet页的每一行 | Workbook: 名称  Sheet\_name: sheet名字 | List：包含所有的sheet每一行 |

* 1. template.py

|  |  |  |  |
| --- | --- | --- | --- |
| 函数名 | 功能说明 | 参数 | 返回值 |
| Check\_dict | 遍历字典结构，找出变量的位置 | Mydict:待解析的字典结构  Parent\_list: 上级目录 | 无 |
| Encode\_password | 加密 | Password:待转换的密码明文 | 密文 |
| Makesure | 装饰器函数，做测试断言 | Func:需要加断言的函数名 | 无 |
| Call\_API | 接口测试公共模板,调用requests方法发送请求 | Test\_dict:sheet每一行的测试数据  Cert:调用https的证书 | Bool |
| generate | 将变量所在的层次结构转换成实际位置 | List\_all:变量所在的层次结构 | string |

* 1. 其他py

|  |  |
| --- | --- |
| 文件名 | 功能说明 |
| Mylog.py | 日志记录函数 |
| Mail.py | 发送邮件服务 |

## 其他约定

目前只实现http, https的传输协议，其他协议不支持

1.只有协议是https双向认证的，才将客户端的证书和密匙放在/data目录

2.客户端的证书和密匙的文件名请按下列方式命名

客户端证书: newfile.crt.pem 客户端密匙: newfile.key.pem

3.其他协议的，请不要在此目录放证书

4.假如客户端证书是path.p12格式的，利用openssl就可以分离出证书和密匙

生成的client端的证书在windows上为p12格式，假如客户端证书为path.p12

openssl pkcs12 -in path.p12 -out newfile.crt.pem -clcerts -nokeys 客户端证书

openssl pkcs12 -in path.p12 -out newfile.key.pem -nocerts -nodes 客户端密匙