

# mm-app接口自动化

参考文档来源：[https://blog.csdn.net/iam\\_emily/article/details/82670540](https://blog.csdn.net/iam_emily/article/details/82670540)

gitlab:<https://gitlab.com/qiujiamin/multilottoapp-api>

## A.基础描述

### 1、进度说明

任务描述	完成度	开始时间	结束时间
基本架构	90%，可用，需优化		2019-12-12
安卓基本接口 ( devserver )	80%，52个基础接口，持续		2019-12-12
登录信息特殊处理	100%，登录/注册成功，更新json文件；logout退出，清空json	2019-12-12	2019-12-12
邮件发送处理	100%，简单版本，仅基本文案及xls执行结果附件	2019-12-13	2019-12-13
使用文档	20%，持续	2019-12-13	
IOS基础用例 ( dev )	20%，配置数据中，未调试	2019-12-16	

### 2、效果

- 用例编写及结果

用例编写										用例执行结果		
ID	case_name	request_name	domain	url_path	run	run_way	header	case_data	data_fiel	nee mo	expect	result
mm-19	进入台头powerball选号页	getchoose	dev_service	/api/lotinfo	yes	post	androidHeader			getch yes	"CODE": "1",	pass
mm-20	获取合买名称	getsyndica	dev_service	/api/order/	yes	post	androidHeader			comrn yes	"CODE": "1",	pass
mm-21	获取合买支付方式	getpaymei	dev_service	/api/order/	yes	post	androidHeader	mm-20 info.syr	syn	getpa yes	"CODE": "1",	pass
mm-22	提交合买Powerball订单	postadddl	dev_service	/api/order/	yes	post	androidHeader	mm-20 info.syr	grou	posta yes	"CODE": "1",	pass
mm-23	powerball完成页推荐彩种	getmaylik	dev_service	/api/lotinfo	yes	post	androidHeader			getch yes	"CODE": "1",	pass
mm-24	获取个人中心	accountce	dev_service	/api/user/	yes	post	androidHeader			comrn yes	"CODE": "1",	pass
mm-25	获取开奖结果列表	getresulti	dev_service	/api/lotinfo	yes	post	androidHeader			comrn yes	"CODE": "1",	pass
mm-26	添加彩种19push订阅	addlottofo	dev_service	/api/lotinfo	yes	post	androidHeader			lottof yes	"CODE": "1",	pass
mm-27	取消彩种19push订阅	dellottofo	dev_service	/api/lotinfo	yes	post	androidHeader			lottof yes	"CODE": "1",	pass
mm-28	获取polandotto开奖结果	getlottes	dev_service	/api/lotinfo	yes	post	androidHeader			getlot yes	"CODE": "1",	pass
mm-29	选号页获取powerball单式喜	favorate_ni	dev_service	/api/lotinfo	yes	post	androidHeader			getch yes	"CODE": "1",	pass
mm-30	获取充值方式	getdeposit	dev_service	/api/depos	yes	post	androidHeader			comrn yes	"CODE": "1",	pass
mm-31	获取充值不同支付方式url	getdeposit	dev_service	/api/depos	yes	post	androidHeader			getde yes	"CODE": "1",	pass
mm-32	获取个人信息	getpersoni	dev_service	/api/user/	yes	post	androidHeader			comrn yes	"CODE": "1",	pass
mm-33	获取email语言列表	getemaila	dev_service	/api/user/	yes	post	androidHeader			comrn yes	"CODE": "1",	pass
mm-34	获取push配置	pushmem	dev_service	/api/user/	yes	post	androidHeader			pushr yes	"CODE": "1",	{ "CODE": "-11100", "MSG": "N
mm-35	获取配置	newestdev	dev_service	/api/user/	yes	post	androidHeader			newes no	"CODE": "1",	pass
mm-36	修改密码	changepai	dev_service	/api/user/	yes	post	androidHeader			chang yes	"CODE": "1",	pass
mm-37	修改个人信息自邮址语音	updatepers	dev_service	/api/user/	yes	post	androidHeader			update yes	"CODE": "1",	pass

- 邮件通知：仅基本效果，附件为用例及result结果



jiamin <jiamin721@163.com>

接口自动化测试报告

收件人 onepiece721@163.com; qiujm(丘佳敏)



testResult2019-12-13.xls  
18 KB

Dear all,

此次一共运行接口个数 54.0 个，通过个数为 51.0 个，失败个数为 3.0，通过率为 94.44%，失败率为 5.56%。详情请看 excel 文件。  
谢谢

### 3、基础使用方式

- 1、编辑用例：

- o 1-1 编辑用例文件：参考 dataconfig/all.xls

ps.文件字段说明，参考data/data\_config.py文件

该文件会持续更新接口基础用例

- o 1-2 编辑请求数据主文件：参考dataconfig/all.json

接口的实际请求数据。接口参数一致时，可使用同一个key.若有参数变化的不同接口，为了接口参数易读性，尽量写一次完整数据。最好不要在modifydata.json中加。

ps.名称不要修改，如修改请将代码中写死的地方也修改。该文件会持续更新



- o 1-3（编辑请求数据副文件）：参考dataconfig/modifyrequestdata.json

作为补充，修改data列数据（实际请求参数）。以后同一个接口多个用例时，可编辑该列。基础用例不需要填写。

ps.名称不要修改，如修改请将代码中写死的地方也修改。该文件会持续更新。

ader	case_dep	data_dep	field_dep	data	need_log	modifydata	expect	result
droidHeader				common	no	common	"CODE":	pass
droidHeader				common	no		"CODE":	pass
droidHeader				asterdevices	no		"CODE":	pass

修改all.json中数据

修改主请求参数中部分数据，如当前将language值改为 de ,其他参数值不变，进行新请求  
适用于同一个接口的参数组合测试，方便阅读比较不同的地方。以覆盖异常流程

```

1 {
2   "common": {"language": "de", "usercheck": ""},
3   "signup": {
4     "currency": "EUR",
5     "address": "507 Xian Dao",
6     "email": "ixde01@gmail.com",
7     "firstname": "ddfdce",
8     "country": "DE",
9     "lastname": "ere"
10  }
11 }

```

## • 2、修改执行文件：

修改要执行的文件当前调用：当前程序中调用OperationExcel时，大部分都没有传文件名（如果要改，请改一下对应的调用）。目前默认读取all.xls，如果要执行其他文件，就把默认值文件修改是最简单的。如果不想修改，直接把你执行的文件改为all.xls吧。位置放在dataconfig路径下。

ps.后续需要优化路径读取

utilconf/operation\_excel.py

```

1 import xlrd
2 from xlutils.copy import copy
3
4 class OperationExcel:
5     def __init__(self, file_name=None, sheet_id=None):
6         # 传进来的时候就用
7         # self.dataconfig = self.get_data(file_name, sheet_id)
8         if file_name:
9             self.file_name = file_name
10            self.sheet_id = sheet_id
11            # self.dataconfig = self.get_data(file_name, sheet_id)
12        else:
13            self.file_name = '../dataconfig/all.xls'
14            # self.file_name = '../dataconfig/buylotto.xls'
15            self.sheet_id = 0
16            self.data = self.get_data()
17        # 金山Excel数据

```

要执行的excel文件

## • 3、（修改发送邮件信息）：

utilconf/send\_email.py中，请配置自己的邮箱信息。

当前只有简版，后续再优化。

```
E:\pypro\multilottoApp-API\utilconf\send_email.py - Notepad++ [Administrator]
File Edit Search View Encoding Language Settings Tools Macro Run Plugins Window ?
年度学习任务及目标计划.txt new 1.txt slider.txt test.txt logininfo.json all.json modifyrequestdata.json
1 import smtplib
2 from email.mime.text import MIMEText
3 from email.mime.multipart import MIMEMultipart
4 import smtplib
5 import time
6 class SendEmail:
7     global send_user
8     global email_host
9     global password
10    global toaddrs
11    global xlsFile
12    # 邮箱服务, 本人用163
13    email_host = "smtp.163.com"
14    # 发送邮箱
15    send_user = "jiamin721@163.com"
16    # 邮箱密码
17    password = "XXX"
18    # 接收人list
19    toaddrs = ['XXX@163.com', 'qiujm@500wan.com']
20    # 文件
21    xlsFile = '../dataconfig/all.xls'
22
```

修改一下发送者邮箱信息：  
文件为要发送的附件，即用例excel文件。如果有修改名字，在这里也修改一下。

#### 4、执行主程序：

直接执行 main/run\_test.py,即可跑dataconfig/XXX.xls用例。

ps.当前仅读一个文件，以后再优化。

```
dataconfig
├── main
│   ├── __init__.py
│   └── run_test.py
├── report
├── utilconf
│   ├── __init__.py
│   └── python-interfacetest-ml.md
├── multilottoApp-API-1
└── multilottointerface-master
109
110
111
112
113
114
115
116
except IndexError as e:
    print("该次请求出错了")
    pass
# 输入密码后再调用该行
# self.send_email.send_main(pass_count, fail_count)
print(len(pass_count))
print(len(fail_count))
if __name__ == '__main__':
```

需要发邮件的话，取消注释

#### 4、问题记录

- 有时候vpn不好执行不成功

----当前仅在dev运行，服务接口均不需要vpn

ProxyError('Cannot connect to proxy.',

NewConnectionError('<urllib3.connection.VerifiedHTTPSConnection object at 0x0000000004B7D1D0>:  
Failed to establish a new connection: [WinError 10061] 由于目标计算机积极拒绝，无法连接。'))

File "D:\python371\lib\site-packages\requests\adapters.py", line 510, in send raise ProxyError(e,  
request=request) requests.exceptions.ProxyError: HTTPSConnectionPool(host='h5app-dev.multilotto.net', port=443): Max retries exceeded with url: /api/others/get\_config (Caused by  
ProxyError('Cannot connect to proxy.', RemoteDisconnected('Remote end closed connection without  
response')))

- pushmem参数未知

--需要问开发该接口作用

## 5、待优化点

- 1、日志
- 2、路径配置
- 3、执行多个文件
- 4、邮件报告
- 5、头信息优化：当前暂未处理H5app请求

## B.代码说明：（待更新）

### 1、接口方法的实现和封装：runmethod

requests库可以很好的帮助我们实现HTTP请求，[API参考文档](#)，这里我创建了runmethod.py，里面包含RunMethod类：

```
import requests
import json
class RunMethod:
    def post_main(self,url,data,header=None):
        res = None
        if header !=None:
            res = requests.post(url =url,data=data,headers=header)
        else:
            res = requests.post(url =url,data=data)
        return res.json()
    def get_main(self,url,data=None,header=None):
        res = None
        if header !=None:
            res = requests.get(url =url,data=data,headers=header,verify=False)
        else:
            res = requests.get(url =url,data=data,verify=False)
        return res.json()
    def run_main(self,method,url,data=None,header=None):
        res = None
        if method == 'post':
            res = self.post_main(url,data,header)
        else:
            res = self.get_main(url,data,header)
        return json.dumps(res,ensure_ascii=False,sort_keys=True,indent=2)
```

这里需要注意就是python默认参数和可选参数要放在必选参数后面.对于相应数据使用json格式进行返回。参数verify=false表示忽略对 SSL 证书的验证。

## 2 组织测试与生成报告

没有采用默认的报告，都用excel方式组织，后面再优化。

## 3、测试数据处理

这一部分主要包括设计测试数据，数据提取和参数化，以及解决数据依赖。

### 1、excel配置

N8															
黄色组合成域名, domain当前配置dve/线上区分															
绿色组合请求参数, data为基础, 读取all.json modifydata作为data辅助, 有值则读取modifydata.json, 加入/修 改到data, 仅区别不同参数组合															
ID	case_name	request_name	domain	url_path	run	run_way	header	case_de	data_depend	field_depend	data	need_login	modifydata	expect	result
mm-01	获取config	get_config	dev_service	/api/others/get_config	yes	post	yes				common	no	common	"CODE": "1"	
mm-02	获取版本是否更新	updatecontrol	dev_service	/api/others/updatecontrol	yes	post	yes				common	no		"CODE": "1"	
mm-03	注册设备	registerdevices	dev_service	/api/others/registerdevices	yes	post	yes				registerdevices	no		"CODE": "1"	
mm-04	获取首页弹框事件	getupcomingevents	dev_service	/api/others/getupcomingevents	yes	post	yes				getupcomingevents	no		"CODE": "1"	
mm-05	未登录打开首页	showhome	dev_service	/api/lotinfo/showhome	yes	post	yes				showhome_unlogin	no		"CODE": "1"	
mm-06	获取全部彩种类型	getgametypelist	dev_service	/api/lotinfo/getgametypelist	yes	post	yes				common	no		"CODE": "1"	
mm-07	登录德国账号	login	dev_service	/api/user/login	yes	post	yes				login	no		"CODE": "1"	
mm-08	登录首页	showhome_log	dev_service	/api/lotinfo/showhome	yes	post	yes				showhome_login	yes		"CODE": "1"	
mm-09	获取lotto列表	getlottories	dev_service	/api/lotinfo/getlottories	yes	post	yes				common	yes		"CODE": "1"	
mm-10	获取casino列表	getcasino	dev_service	/api/lotinfo/getcasino	yes	post	yes				common	yes		"CODE": "1"	
mm-11	获取casino搜索	casino_search	dev_service	/api/lotinfo/casino_search	yes	post	yes				casino_search	yes		"CODE": "1"	
mm-12	获取开奖结果列表	getresultlist	dev_service	/api/lotinfo/getresultlist	yes	post	yes				common	yes		"CODE": "1"	
mm-13	进入powerball单式选号页	getchoosedraw	dev_service	/api/lotinfo/getchoosedraws	yes	post	yes				getchoosedraws	yes		"CODE": "1"	
mm-14	获取支付方式	getpaymentme	dev_service	/api/order/getpaymentmethods	yes	post	yes				getpaymentmethods	yes		"CODE": "1"	
mm-15	提交单式Powerball订单	postadddotask	dev_service	/api/order/postadddotask	yes	post	yes				postadddotask	yes		"CODE": "1"	
mm-16	进入复式Powerball选号页	getchoosedraw	dev_service	/api/lotinfo/getchoosedraws	yes	post	yes				getchoosedraws_system	yes		"CODE": "1"	
mm-17	获取复式支付方式	getpaymentme	dev_service	/api/order/getpaymentmethods	yes	post	yes				getpaymentmethods_system	yes		"CODE": "1"	
mm-18	提交复式Powerball订单	postadddotask	dev_service	/api/order/postadddotask	yes	post	yes				postadddotask_system	yes		"CODE": "1"	
mm-19	进入合买powerball选号页	getchoosedraw	dev_service	/api/lotinfo/getchoosedraws	yes	post	yes				getchoosedraws_syndicate	yes		"CODE": "1"	
mm-20	获取合买名称	getsyndicatename	dev_service	/api/order/getsyndicatename	yes	post	yes				common	yes		"CODE": "1"	
mm-21	获取合买支付方式	getpaymentme	dev_service	/api/order/getpaymentmethods	yes	post	yes	mm-20	info_syndicatename	syndicatename#groupname#userid	getpaymentmethods_syndicate	yes		"CODE": "1"	
mm-22	提交合买Powerball订单	postadddotask	dev_service	/api/order/postadddotask	yes	post	yes	mm-20	info_syndicatename	groupname#userid	postadddotask_syndicate	yes		"CODE": "1"	
mm-23	powerball完成员推荐彩种	getmaylikelott	dev_service	/api/lotinfo/getmaylikelottories	yes	post	yes				getchoosedraws	yes		"CODE": "1"	
mm-24	获取个人中心	accountcenter	dev_service	/api/user/accountcenter	yes	post	yes				common	yes		"CODE": "1"	
mm-25	获取工作经历列表	getworklist	dev_service	/api/lotinfo/getworklist	yes	post	yes				common	yes		"CODE": "1"	

基本描述如上, 每一列对应值含义在data-data\_config.py文件

```

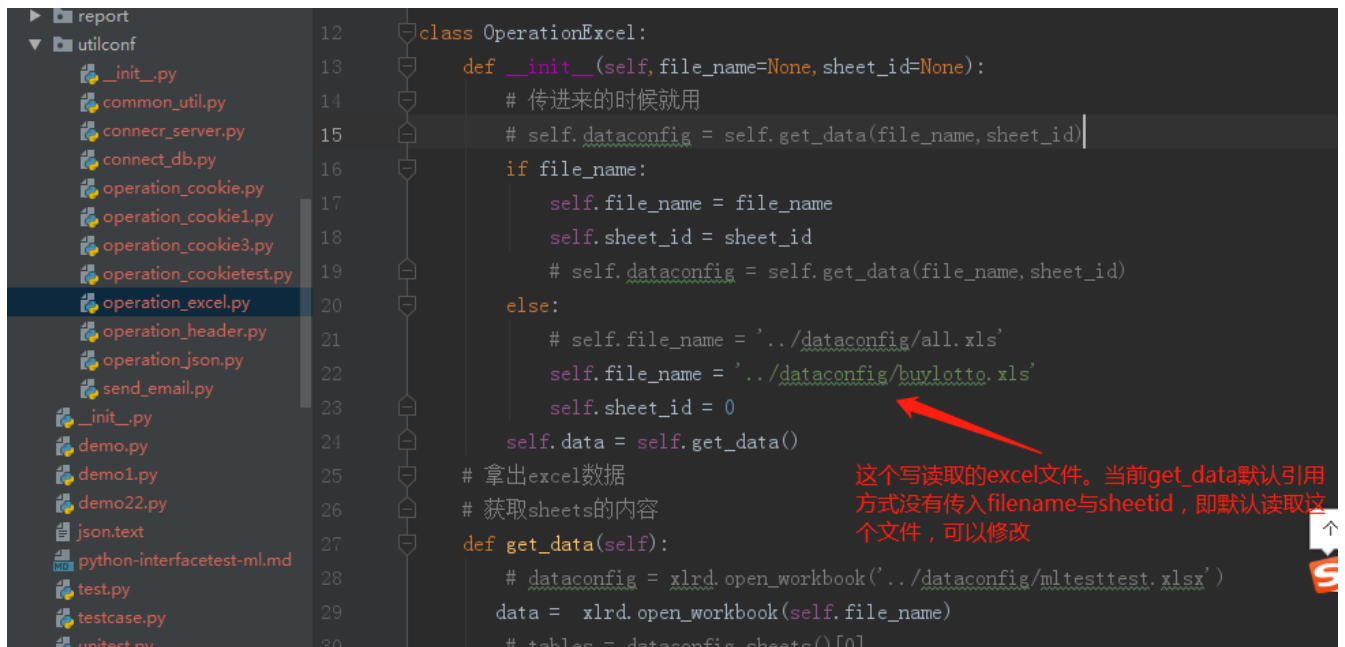
common
data
  __init__.py
  data_config.py
  depend_data.py
  get_data.py
dataconfig
  android
  bak
  __init__.py
  all.json
  all.xls
  buylotto.xls
  logininfo.json
  modifyrequestdata.json
main
rites
pypro
Bookmarks

# 数据依赖 依赖的返回数据。填写提取正则。当前与一个, 依赖多
data_depend = '9'
# 字段依赖
field_depend = '10'
# 请求数据, 写个关键字, 细节到json中写, 为解决excel加密, 且
data = '11'
# 是否需要登录信息, 登录一次, 后续请求参数直接添加登录参数
need_login = '12'
# 请求数据修改, 同data, 仅仅为了好看, 为同一个接口修改data
modifydata = '13'
# 预期结果, 一般写错误码
expect = '14'
# 实际运行结果, 初始不需要填写。与预期匹配正确, 写入pass;与
result = '15'
# 获取caseid
def get_id():
    return global_var.Id
  
```

## 2、操作表格

操作Excel的模块, 主要包含了对Excel表格的操作, 获取表单、行、列、单元格内容等。

其中写数据用于将运行结果写入Excel文件 (result:通过写pass,不通过写返回值), 先用copy复制整个文件, 通过get\_sheet()获取的sheet有write()方法。



```
12 class OperationExcel:
13     def __init__(self, file_name=None, sheet_id=None):
14         # 传进来的时候就用
15         # self.dataconfig = self.get_data(file_name, sheet_id)
16         if file_name:
17             self.file_name = file_name
18             self.sheet_id = sheet_id
19             # self.dataconfig = self.get_data(file_name, sheet_id)
20         else:
21             # self.file_name = '../dataconfig/all.xls'
22             self.file_name = '../dataconfig/buylotto.xls'
23             self.sheet_id = 0
24             self.data = self.get_data()
25         # 拿出excel数据
26         # 获取sheets的内容
27     def get_data(self):
28         # dataconfig = xlrd.open_workbook('../dataconfig/mltesttest.xlsx')
29         data = xlrd.open_workbook(self.file_name)
30         # tables = dataconfig.sheets()[0]
```

这个写读取的excel文件。当前get\_data默认引用方式没有传入filename与sheetid，即默认读取这个文件，可以修改

### 3、json读取处理

为解决数据展示统一和excel加密不好修改数据情况，请求参数原始值（data）及修改参数（modifydata项）或其他特殊处理(如登录信息保存)，提供根据关键字进行json文件读，获取，写等操作的方法operation\_json.py

读写操作使用的是 `json.load()` , `json.dump()` 传入的是文件句柄。

数据以字典格式处理，common即为关键字，如图。



```
depend_data.py 2 {
get_data.py 3
dataconfig 4
├─ android 5
├─ bak 6
├─ _init_.py 7
├─ all.json 8
├─ all.xls 9
├─ buylotto.xls 10
├─ logininfo.json 11
├─ modifyrequestdata.json 12
└─ main
    └─ _init_.py
        {
            "common": {
                "useragent": "MI 6 ",
                "pushproject": "curacao",
                "platform": "4000",
                "osversion": "5.1.1",
                "remote_addr": "172.17.100.15",
                "version": "3.0.0",
                "resolution": "900x1600",
                "language": "EN",
                "pushid": "e4f05aae-e0f0-47bb-92b3-a483cb53199f",
```

审阅 视图 安全 开发工具 特色应用 查找命令、搜索模板

合并居中 自动换行

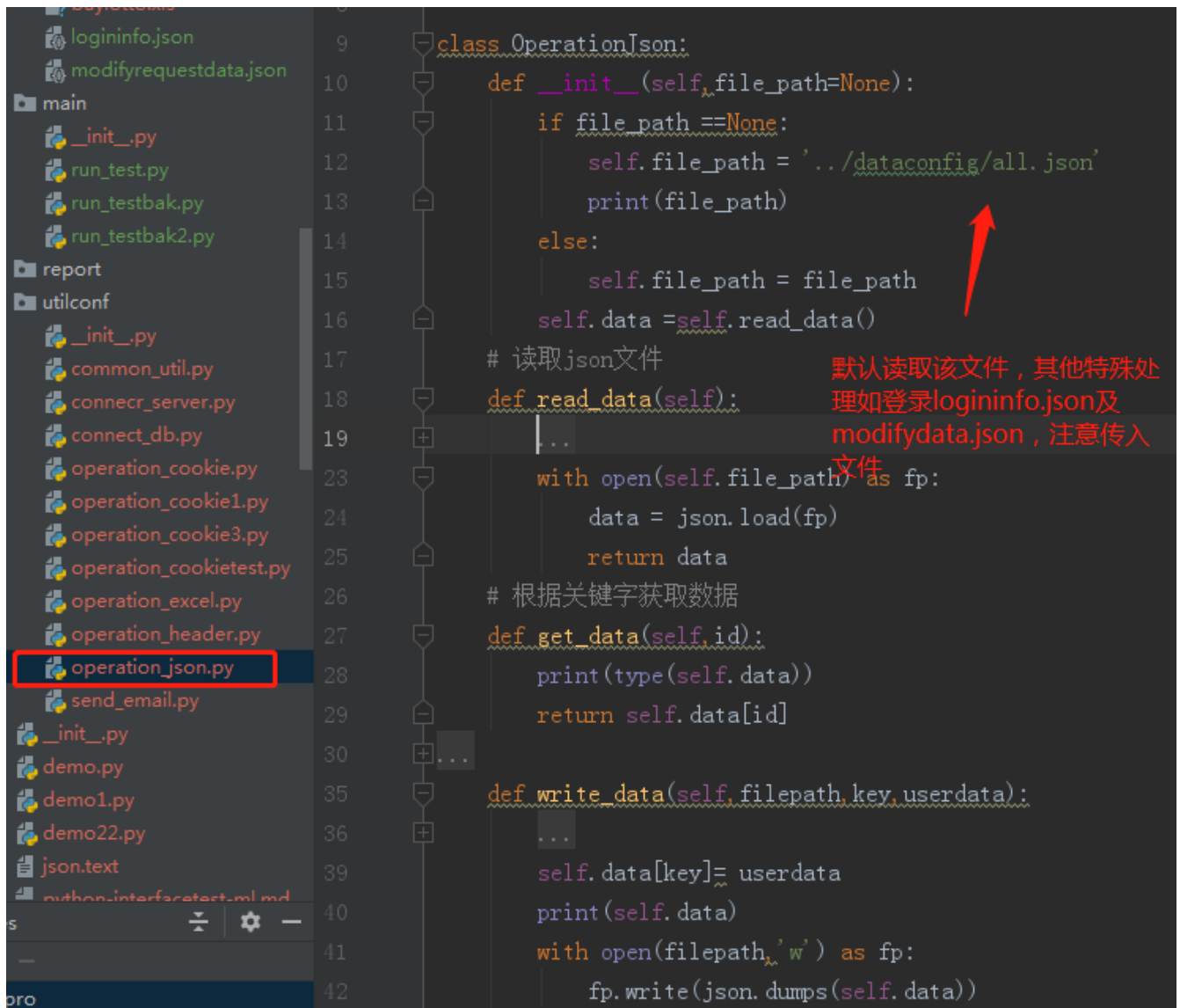
常规

条件格式 表格样式

文档助手 求和 筛选 排序 格式 行和列 工作表

	F	G	H	I	J	K	L	M	N	
	run	run_way	header	case_de	data_depend	field_depend	data	need_login	modifydata	e
	yes	post	yes				common	no	common	"i
ol	yes	post	yes				common	no		"i
es	yes	post	yes				registerdevices	no		"i
events	yes	post	yes				getupcomingevents	no		"i
	yes	post	yes				showhome_unlogin	no		"i
sid	yes	post	yes				common	no		"i





```
9 class OperationIson:
10     def __init__(self, file_path=None):
11         if file_path == None:
12             self.file_path = '../dataconfig/all.json'
13             print(file_path)
14         else:
15             self.file_path = file_path
16             self.data = self.read_data()
17         # 读取json文件
18     def read_data(self):
19         ..
20         with open(self.file_path) as fp:
21             data = json.load(fp)
22             return data
23         # 根据关键字获取数据
24     def get_data(self, id):
25         print(type(self.data))
26         return self.data[id]
27     ..
28     def write_data(self, filepath, key, userdata):
29         ..
30         self.data[key] = userdata
31         print(self.data)
32         with open(filepath, 'w') as fp:
33             fp.write(json.dumps(self.data))
```

默认读取该文件，其他特殊处理如登录logininfo.json及modifydata.json，注意传入文件

#### 4、获取测试数据并做对应处理

在定义好Excel和json操作及data配置模块后，我们将其应用于我们的测试表单，定义一个获取数据模块：

该模块将Excel操作类实例化后用于操作测试表单，分别获得测试运行所需的各种条件。

```
15         col = int(data_config.get_id())
16         caseid= self.opera_excel.get_cell_value(row,col)
17         return caseid
18         # 获取casename, 对如登录等需要做特殊处理
19         def get_case_name(self,row):
20             col = int(data_config.get_case_name())
21             casename= self.opera_excel.get_cell_value(row,col)
22             return casename
23         # 获取域名
24         def get_domain(self,row):
25             col = int(data_config.get_domain())
26             domain_flag = self.opera_excel.get_cell_value(row,col)
27             if domain_flag == 'dev_service':
28                 domain = 'https://h5app-dev.multilotto.net'
29             elif domain_flag == 'dev_h5app':
30                 domain = 'https://h5app-dev.multilotto.com'
31             elif domain_flag == 'online_service':
32                 domain = 'https://service.multilotto.net'
33             elif domain_flag == 'online_h5app':
34                 domain = 'https://h5app.multilotto.net'
35             else:
36                 print('域名错误或域名为空, 将填充dev_service, 请检查')
```

域名可配置这几个, 后续可写到配置, 当前写死

```
def is_header(self,row):
    col = int(data_config.get_header())
    header = self.opera_excel.get_cell_value(row,col)
    if header == 'yes':
        # return 'header'
        return data_config.get_header_value()
    else:
        return None
    # return header
```

header的值当前也写死, 后续应该会增加读写

```
return urlpath
def request_url(self,row):
    domain = self.get_domain(row)
    urlpath = self.get_urlpath(row)
    request_url = domain+urlpath
    return request_url
# 获取是否执行
```

请求url由两列组合

# 获取请求数据

```
def get_request_data(self, row):
    col = int(data_config.get_data())
    data = self.opera_excel.get_cell_value(row, col)
    if data == '':
        return None
    return data

def get_modifyrequest_data(self, row):
    col = int(data_config.get_modifydata())
    modifydata = self.opera_excel.get_cell_value(row, col)
    if modifydata == '':
        return None
    return modifydata
```

请求参数与modify组合

# 通过获取关键字 拿到data数据

```
def get_data_for_json(self, row):
    # 不在构造函数，而在这里实例化，因为有的data不需要json
    opera_json = OperationJson()
    ...
    request_data = opera_json.get_data(self.get_request_data(row))
    return request_data

def get_modifydata_for_json(self, row):...
# 处理data, 有modifydata就修改data, 没有不处理
def opra_request_data(self, row):
    requestdata = self.get_request_data(row)
    modifydata = self.get_modifyrequest_data(row)
    if requestdata != None:
        requestdata = self.get_data_for_json(row)
    if modifydata != None:
        modifydata = self.get_modifydata_for_json(row)
```

# 判断是否有case依赖

```
def is_depend(self, row):  
    col = int(data_config.get_case_depend())  
    depend_case_id = self.opera_excel.get_cell_value(row, col)  
    if depend_case_id == "":  
        return None  
    else:  
        return depend_case_id
```

# 获取数据依赖字段

```
def get_depend_field(self, row):  
    # 等会要处理出列表  
    col = int(data_config.get_field_depend())  
    data = self.opera_excel.get_cell_value(row, col)  
    if data == "":  
        return None  
    else:  
        if "#" in data:  
            datalist = data.split("#")  
            return datalist  
        else:  
            return data
```

依赖数据分隔

响应数据的处理在主函数  
中，类似以#分隔

## 5、数据依赖