# requests实战

```
百度翻译实战
把test翻译成中文
#!/usr/bin/env python
# -*- coding:utf-8 -*-
#====#====#====#====
#Author:
#CreatDate:
#Version:
#====#====#====#====
import hashlib
import requests
import pytest

#加密函数
def MyMd5(mystr):
    mdmystr=hashlib.md5(mystr.encode(encoding='utf-8')).hexdigest()
    return mdmystr

def test_baidu():
    #url
    url='https://fanyi-api.baidu.com/api/trans/vip/translate'
    mdmystr=MyMd5("20210507000816969test123456sGZsjoOY2OThznTznsRT")
    #参数
    data={
        "q":"test",
        "from":"en",
        "to":"zh",
        "appid":"20210507000816969",
        "salt":"123456",
        "sign":mdmystr
    }
    #20210507000816969test123456sGZsjoOY2OThznTznsRT
    #请求
    res=requests.get(url,params=data)
    #把响应转换为json格式,便于后面获取里面的内容
    jr=res.json()
    #获取响应后的翻译结果
    rdata=jr['trans_result'][0]['dst']
    #断言
    assert rdata=="测验"



if __name__=='__main__':
    pytest.main(['01百度翻译实战.py','-s'])


百度翻译升级版(翻译软件核心代码)
```

```
#!/usr/bin/env python
# -*- coding:utf-8 -*-
#====#====#====#====
#Author:
#CreatDate:
#Version:
#====#====#====#====

import hashlib
import requests
import pytest

#获取用户输入的信息
def getUserMsg():
    print("翻译的语种范围为:中文,英文,法语,俄语,日语,韩文,西班牙,葡萄牙,阿拉伯语,德语,意大利语")
    value=input("请输入你要翻译的内容:")
    if value=='q':
        return value
    myfrom=input("请输入你的内容属于哪个语种:")
    myto=input("请输入你要翻译成的目标语言:")

    #判断用户输入的语种属于什么代码
    mydict={"中文":"zh",'英文':'en','法语':'fra',
            '俄语':'ru','日语':'jp','韩文':'kor',
            '西班牙':'spa','葡萄牙':'pt','阿拉伯语':'ara',
            '德语':'de','意大利语':'it'}

    return value,mydict[myfrom],mydict[myto]

#加密函数
def MyMd5(mystr):
    mdmystr=hashlib.md5(mystr.encode(encoding='utf-8')).hexdigest()
    return mdmystr

#测试用例,也是业务函数
def test_maker():
    while True:
        usermsglist=getUserMsg()
        if usermsglist[0]=='q':
            break

        url = 'https://fanyi-api.baidu.com/api/trans/vip/translate'
        mdmystr = MyMd5("20210507000816969"+usermsglist[0]+"123456sGZsjoOY2OThznTznsRT")
        # 参数
        data = {
            "q": usermsglist[0],
            "from": usermsglist[1],
            "to": usermsglist[2],
            "appid": "20210507000816969",
            "salt": "123456",
            "sign": mdmystr
```

```
        }
        # 20210507000816969test123456sGZsjoOY2OThznTznsRT
        # 请求
        res = requests.get(url, params=data)
        # 把响应转换为json格式,便于后面获取里面的内容
        jr = res.json()
        # 获取响应后的翻译结果
        rdata = jr['trans_result'][0]['dst']
        print("翻译的结果为:",rdata)

test_maker()
```

# unittest框架和requests结合

```
测试后台程序4个接口

#!/usr/bin/env python
# -*- coding:utf-8 -*-
#====#====#====#====
#Author:
#CreatDate:
#Version:
#====#====#====#====
import requests
import unittest
import time

class Maker(unittest.TestCase):
    @classmethod
    def setUpClass(cls):
        cls.url='http://127.0.0.1:8808/api'
        globals()['n']=0

    def setUp(self):
        n=globals()['n']
        #每个用例的预期结果
        if n==0:
            self.mydata={"code":"200","msg":"成功","data":None}
        elif n==1:
            self.mydata={"code":"200","msg":"成功","data":"maker"}
        elif n==2:
            self.mydata={"code":"200","msg":"成功","data":"maker"}
        elif n==3:
            self.mydata={"code": "200","msg":"成功","data": {
                "id": 0,
                "money": "金额",
                "name": "测试",
                "number": "等级"
                }
            }
```

```
37
38      #get无参
39      def test_get(self):
40          globals()['n']= 1
41          print("test_get")
42          res=requests.get(self.url+'/block')
43          print(self.mydata)
44          #实际结果
45          rj=res.json()
46          self.assertEqual(rj,self.mydata,'用例未通过')
47
48      #get有参数
49      def test_getmaker(self):
50          globals()['n']= 2
51          print("test_getmaker")
52          data={"name":"maker","passwd":"123456","email":"75242424@qq.com"}
53          res=requests.get(self.url+'/block/register',params=data)
54          print(self.mydata)
55          # 实际结果
56          rj = res.json()
57          self.assertEqual(rj, self.mydata, '用例未通过')
58
59      #post表单
60      def test_post(self):
61          globals()['n'] = 3
62          print("test_post")
63          data={"name":"maker","passwd":"123456"}
64          res=requests.post(self.url+'/block/login',data=data)
65          print(self.mydata)
66          # 实际结果
67          rj = res.json()
68          self.assertEqual(rj, self.mydata, '用例未通过')
69
70      #post的json
71      def test_postjson(self):
72          print("test_postjson")
73          data={"name":"maker"}
74          res = requests.post(self.url + '/block/msg', json=data)
75          print(self.mydata)
76          # 实际结果
77          rj = res.json()
78          self.assertEqual(rj, self.mydata, '用例未通过')
79
80  if __name__=='__main__':
81      unittest.main()
```

# requests参数化-csv文件

```
1  目的:
2  把csv文件中的中文词汇翻译成英文,写到csv对应的地方
3
4  #!/usr/bin/env python
```

```
# -*- coding:utf-8 -*-
#====#====#====#====
#Author:
#CreatDate:
#Version:
#====#====#====#====
import csv
import requests

#读csv
def ReadCsv(filename):
    mylist=[]
    with open(filename, 'r',encoding='utf-8') as f:
        obj = csv.reader(f)
        print(obj)
        for i in obj:
            mylist.append(i)
    return mylist

# ReadCsv("data3.csv")

#写csv
def WriteCsv(mylist,newmylist):
    with open("data3.csv",'w',encoding='utf8',newline='') as f:
        #需要把f转换为csv对象
        obj=csv.writer(f)
        for i in range(2):
            print(mylist[i] + newmylist[i])
            obj.writerow(mylist[i]+newmylist[i])

#
mylist=ReadCsv("data3.csv")
print(len(mylist))
newmylist=[[],["hello"],["baye"]]
WriteCsv(mylist,newmylist)
代码为完成
```