

一、@pytest.mark.parametrize()基本用法

@pytest.mark.parametrize(args_name,args_value)

args_name：参数名

args_value：参数值(列表，元祖，字典列表，字典元祖)，**有多个值用例就会执行多少次**

第一种方式：

```
1 import pytest
2
3 class TestApi:
4
5     @pytest.mark.parametrize('args',['百里','星瑶','依然'])
6     def test_01_baili(self,args):
7         print(args)
8
9 if __name__ == '__main__':
10     pytest.main()
```

第二种方式：跟unittest的ddt里面的@unpack解包的一样

```
1 import pytest
2
3 class TestApi:
4
5     @pytest.mark.parametrize('name,age',[['百里','38'],['微微','18']])
6     def test_01_baili(self,name,age):
7         print(name,age)
8
9 if __name__ == '__main__':
10     pytest.main()
```

二、YAML文件详解---实现接口自动化

1.用于全局的配置文件 ini/yaml

2.用于写测试用例（接口测试用例）

yaml简介：

yaml是一种数据格式，支持注释，换行，多行字符串，裸字符串（整形，字符串）。

语法规则：

- 1.区分大小写
- 2.使用缩进表示层级，不能使用tab键缩进，只能用空格（和python一样）
- 3.缩进没有数量的，只要前面是对其的就行。
- 4.注释是#

数据组成

- 1.Map对象，键值对 键:(空格)值

多行的写法：

msxy:

 name:百里

 age:18

一行的写法：

msxy: {name: 百里,age: 18}

- 2.数组（list），用一组横线开头，如：

多行的写法：

-

msxy:

- name: 百里

- age: 18

-

msjy:

- name: xingyao

- age: 20

一行的写法：

-

msxy: [{name: 百里},{age: 18}]

三、yaml接口自动化实战

这个数据驱动太粗糙。不够

- 1.断言的封装。

2.allure报告的定制。

3.关键字驱动和数据驱动结合实现接口自动化测试。

4.python的反射

正常：先初始化对象，在调方法。

反射：通过对象得到类对象，然后通过类对象调用方法

5.jenkins的持续集成和allure报告集成，并且根据自动化的报告的错误率发送电子邮件。

系统：

循序暂进。

四、别让假努力毁掉你

1.熬夜早起搞学习，自我感动很积极。

2.碎片时间也要提升自己，然后在黄金时间摸鱼玩手机。

3.疯狂收集视频和笔记，过后让它在云盘吃灰

4.目标计划猛如虎，实操上手二十五

1.忙碌陷阱

忙工作，给公司

忙面试，给工作

忙赚钱，给消费

忙前途，职业规划。

3个半月。职业生涯:20年。自动化领域去工作。

10-15K 一个月的涨薪。

,

2.认知陷阱

自动化无用。

更真实，更脚踏实地，更系统去学。

很多人自学：功能，自动化，1年没办法跨入。

- 1.项目实战
- 2.简历不会包装
- 3.没有面试葵花宝典。

32 走技术还是走管理？

12K，接口会一点，不敢说精通，自动化会一点，性能会一点。

测试能干多久？

功能测试：35到头。

自动化测试/性能/安全 45

管理，讲师。

框架不会做什么自动化。

外包，项目，3-6个月。做什么自动化。

自研，产品，1-20年，必须。

中国电信。3年多。

一个星期一次迭代，紧急发版。多少次？回归？业务越来越大。点点

德邦，沃尔玛，达飞金融（快分期）

新的高度。停在原地。

接触技术，快车道，职业生涯进入快车道。

功能测试，（竞争） 35中年危机 14K。

自动化/性能 10% （ 竞争 ） 15K起步

学习，越快越好

现在不学，以后100%会学，除非你不做测试了。

自动化/性能。

10年买房，