

# 《《接口自动化测试框架》》

## 使用说明书

北京乐步教育科技有限公司

Beijing Lebu Education Technology Co., Ltd

## 修订记录

日期	作者	版本	备注
2022-07-06	郭志国	1.0	

## 目 录

<b>1</b>	<b>概述.....</b>	<b>5</b>
1.1	编写目的.....	5
<b>2</b>	<b>目录介绍.....</b>	<b>5</b>
2.1	目录结构.....	5
2.2	CASE 目录.....	5
2.3	CASEYML.....	5
2.4	COMMON .....	6
2.5	CONFIG.....	6
2.6	REPORT.....	6
2.7	REQUEST_DATA .....	6
2.8	RESPONSE_DATA.....	6
2.9	RUNLOG.....	6
2.10	LISTENER.PY.....	6
2.11	PYTEST.INI .....	6
2.12	REQUIREMENT.TXT.....	6
2.13	.GITIGNORE .....	7
<b>3</b>	<b>注意事项.....</b>	<b>7</b>
3.1	YML 文件命名规则.....	7
3.2	YML 文件中 KEY 及变量规则 .....	7
3.2.1	<i>feature</i> .....	7
3.2.2	<i>story</i> .....	7
3.2.3	<i>module_flag</i> .....	7
3.2.4	<i>description</i> .....	7
3.2.5	<i>params_type</i> .....	8
3.2.6	<i>response</i> .....	8
3.2.7	<i>depends_on</i> .....	8
3.2.7.3	<i>case</i> .....	9
3.2.7.4	<i>depend</i> .....	9
3.2.7.5	<i>replace</i> .....	9
3.2.8	<i>assert</i> .....	9
3.3	代码提交注意事项 .....	10
3.4	PYTHON 版本及虚拟环境 .....	10

3.4.1	<i>python 版本</i> .....	10
3.4.2	<i>虚拟环境</i> .....	10
3.5	删除数据 .....	10

# 1 概述

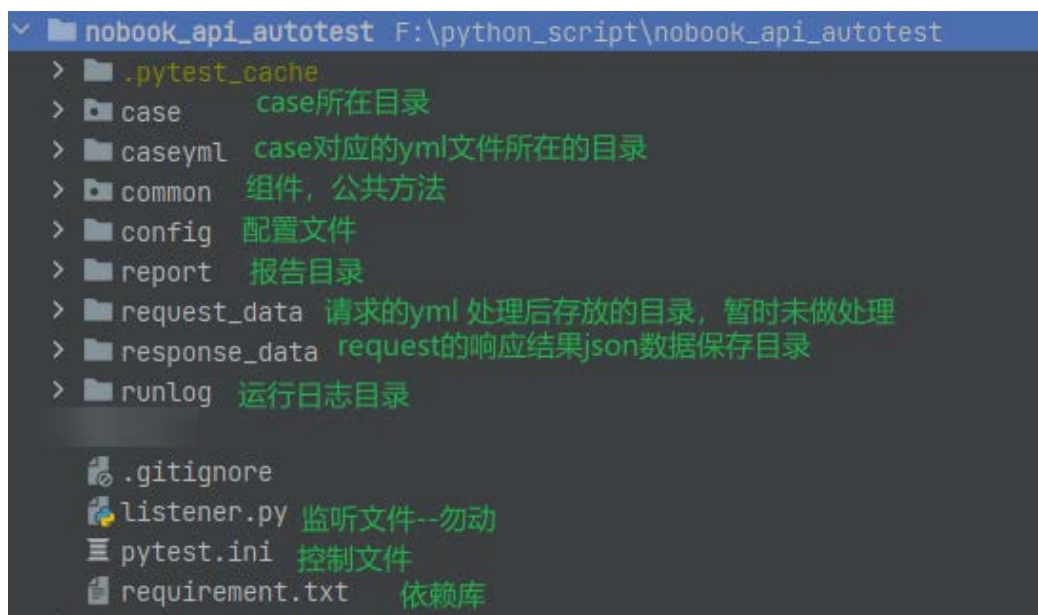
## 1.1 编写目的

为方便组内成员快速上手接口自动化测试框架，同时也为了统一使用规则，有效减少使用问题，提升工作效率。

# 2 目录介绍

## 2.1 目录结构

目录结构如下图所示：



## 2.2 case 目录

所有的 case 的存在的目录

## 2.3 caseyaml

所有的 case 存在的 py 文件对应的 yaml 文件所在的目录

## 2.4 common

所有的组件及公共方法所在的目录

## 2.5 config

config.yml 配置文件所在的目录

## 2.6 report

报告所在的目录

## 2.7 request\_data

暂定：case 对应的 yml 文件中的数据，经过处理后的数据，然后转存的 yml 文件所在的目录。目前框架中未启用该目录

## 2.8 response\_data

那些需要保存 request 响应结果的 json 数据的 json 文件所在的目录

## 2.9 runlog

运行日志所在的目录

## 2.10 listener.py

Allure 库的 listener.py 文件，经过了修改

## 2.11 pytest.ini

配置文件

## 2.12 requirement.txt

虚拟环境所依赖的第三方库信息

## 2.13 .gitignore

该文件里配置了提交代码的一些忽略项

# 3 注意事项

## 3.1 yml 文件命名规则

1. yml 文件的名称要与对应的 case 所在的 py 文件名保持一致。如：testdemo.py 对应的 yml 文件就应该为 testdemo.yml
2. 所以的 yml 文件名的后缀为.yml，不建议使用 .yaml
3. caseyaml 目录下的 CaseTemplate.yml 为模板 yml，请勿变动。

## 3.2 yml 文件中 key 及变量规则

主要介绍 yml 中的一些需要注意点及容易犯错的地方。

### 3.2.1 feature

该 yml 文件中的 case 所属于哪个功能版块，统筹该 yml 下的所有 case。

### 3.2.2 story

小功能模块，如登录

### 3.2.3 module\_flag

功能模块标记，这个是预留的扩展，是大版块，简单的说如：v6 文控，v5 之类的。可以简单直接的理解为，只要域名一致，那么 module\_flag 的值就是一样的。

### 3.2.4 description

该 case 的目的，用于生成报告图表中的 case 的 title。

### 3.2.5 params\_type

指的是 request 请求的参数是 params 还是 data 或者 json。其值与 request\_param 下的 params 的 key 一致。如 params\_type 的值为 data 时，则 request\_param 下的 params 就要变更为 data

### 3.2.6 response

Request 的响应结果是否需要怎样处理，以及怎样处理。通常用于数据依赖和 token 依赖。若是不需要处理响应结果，其值直接为空即可。

#### 3.2.6.1 filepath

其值为两种。第一种值为 online，此时表示使用在线的方式，来处理数据依赖；第二种为 xxx.json，指的是将 request 的响应结果的 json 数据保存到 response\_data 目录下的 xxx.json 文件中。如 0015.json 表示的是将 response json 数据保存到 response\_data 目录下 0015.json 文件中。如果写的是 response\_data/0015.json，则表示的是将结果保存到 response\_data 目录下的子目录 response\_data 下的 0015.json 文件中。

#### 3.2.6.2 resp\_keys

表示的是要将响应结果的哪些 key 的值提取出来。需要注意的是 resp\_keys 的值为 list，且均为 jsonpath 表达式。如：[ \$.auth\_token ]

#### 3.2.6.3 keys

将提取的结果保存到 json 文件中的哪个 key 下。其值为 list。[ 0015\_token ]

### 3.2.7 depends\_on

数据依赖，指的是当前 case 在发送请求前，需要将请求参数中哪些字段的值进行替换。若是没有数据依赖，直接为空即可。

#### 3.2.7.1 online

指的是数据的方式，是读取的在线的数据变量，还是读取的 json 或 yml 文件。当 online 的值为 true 时，表示读取的在线 online\_data 对象的属性值；当 online 的值为 false 时，表示读取的是 json 或 yml 文件。

online 的默认为 true，即使为空也表示 online 的值为 true。当 online 的值为 true 时，无需填写 type 和 case\_id 的值。



### 3.2.7.2 type

当 `online` 为 `false` 时，需要填写该值。其值为固定的两个值，`reponse` 和 `request`。目前框架中未涉及到 `request`，所以目前只需要写 `response` 即可。

`reponse` 表示的是数据依赖为某个接口请求的响应结果

`request` 表示的是数据依赖为某个接口请求的请求参数值。

### 3.2.7.3 case\_id

当 `online` 的值为 `false` 时，需要填写该值。当 `type` 的值为 `response` 时，需要填写 json 文件在 `response_data` 目录下相对于 `response_data` 目录的 相对路径(含文件名)；当 `type` 的值为 `request` 时，需要填写 yml 文件，规则为 case 对应的 yml 文件::case 对应的 yml 文件中的 key，如：`testdemo.py::login`

### 3.2.7.4 depend\_key

表示想要替换的值在源数据中存于哪个 key 下。其值为 `list`。

### 3.2.7.5 replace\_key

表示想要替换当前 case 请求参数的哪个参数的值。其值为 `list`

## 3.2.8 assert

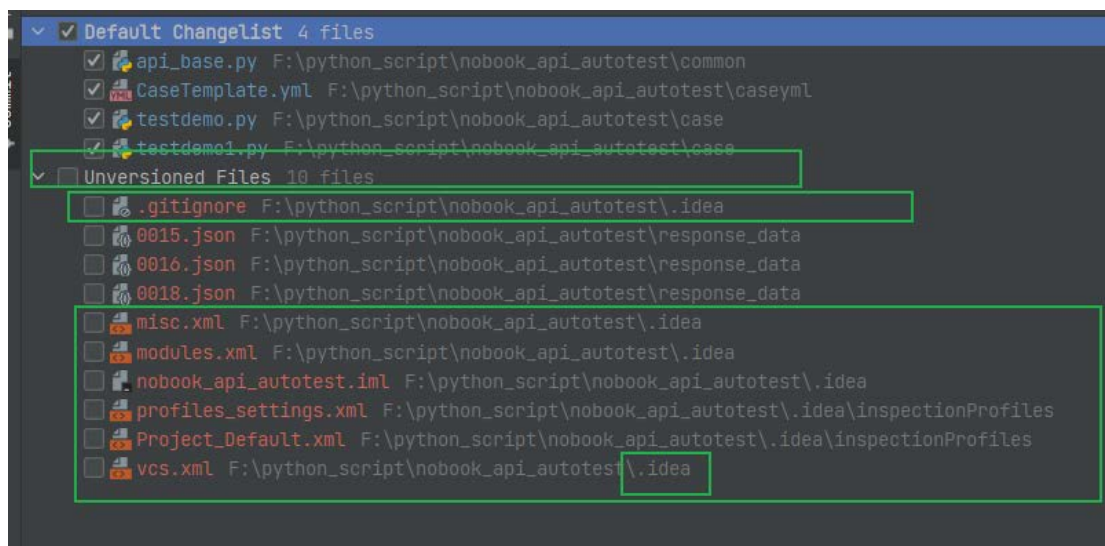
断言。

- 当一个 case 有多重断言时，其该 case 的断言为 `list`，如下图所示。

```
- [pytest.assume(200==req.status_code),  
  "pytest.assume('奥林匹克化学学校' in (jsonpath(req_json,'$.schoolname')))" ]
```

- 断言只支持 `pytest.assume()`，不支持其他的断言
- `jsonpath` 表达式的开头必须为 `jsonpath`，不能为 `jsonpath.jsonpath`
- 需要想要获取状态码，固定写法: `req.status_code`
- 如果想要断言响应 json 结果的某个字段，变量名必须为 `req_json`，如：想要断言成功登录接口的响应结果的学校名称，则提取学校的 `jsonpath` 表达式为：  
`jsonpath(req_json,'$.schoolname')`，而完事的断言表达式为: `pytest.assume('实验小学' in jsonpath(req_json,'$.schoolname'))`

### 3.3 代码提交注意事项



如上图所示，在进行提交代码时，不得提交根目录下的.idea 文件夹下的任何内容

### 3.4 python 版本及虚拟环境

#### 3.4.1 python 版本

python 的版本区间[3.8,3.10)，即必须为 3.8.x 和 3.9.x 这两个系列的版本。

#### 3.4.2 虚拟环境

##### 3.4.2.1 名称

虚拟环境的名称必须为 python\_api\_venv

##### 3.4.2.2 包

虚拟环境中的第三方依赖包，必须安装 requirement.txt 中的内容。安装方法如下：

```
pip install -r requirement.txt
```

不得安装 requirement.txt 中以外的包。

### 3.5 删除数据

凡是有新增数据动作的 case，最后都要对数据进行删除，以防垃圾数据堆积。

