# 05 场景测试用例开发

# 回顾

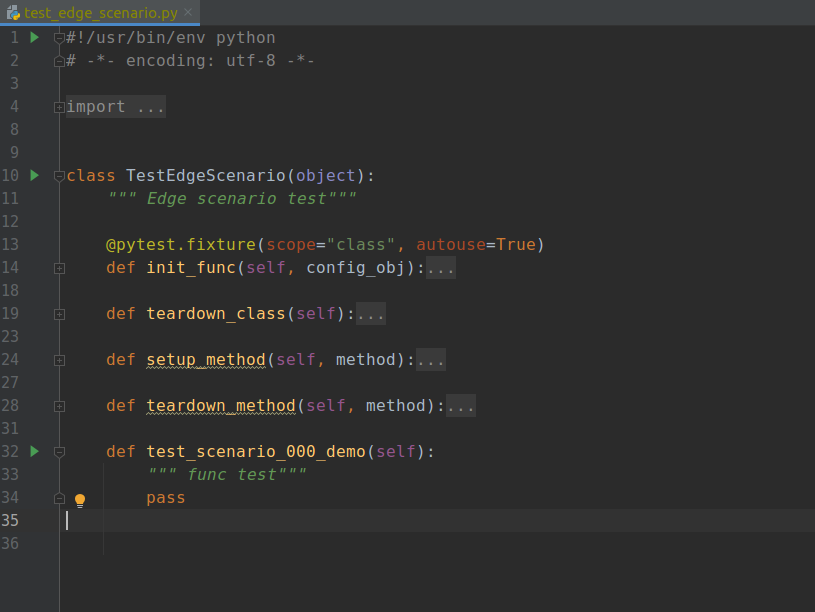
根据前文的步骤,  我们已经完成了以下几个步骤:

* 生成了大部分代码, 拷贝到指定的目录下
* 配置了conf下的toml配置文件, 填入了一些数据参数
* 完善项目级conftest的代码, 使接口对象可以正常调用
* 能够进行单接口测试用例的开发
* 能够进行参数边界值测试用例的开发

接下来, 就可以正式开始场景测试用例的开发啦

# 脚本开发

我门依然先看下场景脚本中生成了哪些代码, 如下图:



是的, 如您所见, 场景测试只是生成了一个代码架子, 内部不包含任何涉及的接口

但我能想到, 当聪明的您, 读到这里的时候, 应该对apitest的接口测试框架有了一定了解, 肯定是可以理解上图只包含一个空架子的原因了.

回想一下, 我们通过swagger去生成一些必要的接口对象的代码, 单接口测试用例代码, 这些都是接口文档中的基础设定的内容.

而场景呢, 是需要不同接口组合起来用, 才会形成的. 换个说法, 就是用户的真实使用习惯, 这些具体的业务逻辑是无法用代码生成的,需要我们根据实际情况去码代码.

但是也不用紧张, 因为当熟悉了apitest框架后, 就会发现, 写场景测试用例的代码, 实际就是接口组装的过程.

下面直接贴一个子设备crud的例子, 如下图:



这个例子中, 验证部分写的不是很充分, 只是为了作为一个例子参考, 该流程是实现了子设备的crd过程

* 创建一个子设备
* 查询该子设备是否已存在(由于是异步接口, 所以不能通过创建接口的resp中内容判断是否创建成功)
* 删除该子设备
* 查询该子设备是否删除成功(同上, 异步接口不能通过删除接口的resp判断是否删除成功)

例子很简单, 只是抛砖引玉, 具体测试中的业务逻辑, 还是需要根据实际情况去开发

至此, 场景测试的介绍就补充完啦, 如果想了解更多的例子,可以参考代码