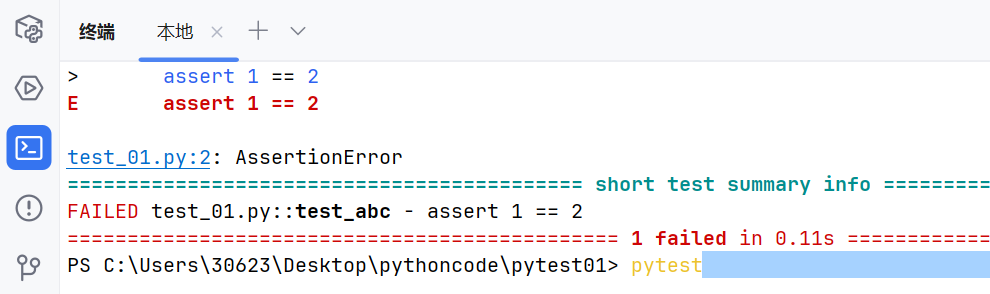
**Pytest框架**

安装好pytest后第一步——使用（创建用例、执行用例）



如何执行用例？

在pycharm的本地命令框中输入pytest，按下回车，即可运行pytest





本地框中输入pytest-h，即可获得pytest官方的使用说明和参数

后续学习可以按参数、mark、fixture、hook（行业天花板，难度高，薪资20k上）这个顺序慢慢学习

**requests（向接口发送请求）**

若能将【向接口发送请求】与【pytset】结合，即可做接口自动化测试

若能将【控制浏览器】与【pytest】结合，即可做浏览器自动化测试

若能将【控制手机APP】与【pytest】结合，即可做APP自动化测试

.py文件第一句代码必须为import requests，为导入requests工具

request的使用格式



在request中常见的几种小难题

1. 表单参数

主要应用在Web项目，需要注意参数只能是字符串；请求头中包含form



（此项目报错，怀疑跟网址有关，api是什么）

技巧：在requests中，如果data参数是一个字典，则自动将其识别为表单，并自动添加请求头，不需要自己再写headers

1. JSON参数（支持多种数据类型，表单只支持字符串）

主要用在各类项目中，需要注意请求头中包含JSON



技巧：在requests中，如果传递json参数，则自动将其识别为json参数类型，并自动添加请求头，不需要自己再写headers

1. 文件上传（用户上传图片等）常用常见

主要用在各类项目中，其中上传方式有两种：一是body直传（用得较少）二是表单。注意：请求头需要说明使用哪种方式，是body直传还是表单



技巧：在requests中，如果传递files参数，则自动将其识别为表单文件上传，并自动添加请求头，不需要自己再写headers

1. 接口关联：一个框架中至少要有两个接口，这样才是接口关联

需要掌握编程语言基础：创建变量、传递变量、读取变量和打印变量

如何从上一个响应中提取变量：一是re；二是jsonpath（常用于json接口）；三是xpath

接口关联流程：先请求第一个接口，得到响应；再从响应中提取数据（token），创建变量（用于下一个接口）；最后在第二个接口中，使用变量（内含token）

1. 参数化测试

与pytest框架中的mark有关，应先学习pytest

学会参数化测试后，可以再学数据驱动测试，数据驱动测试=参数化测试+数据文件（主要有csv,json,yaml,excel,mysql）

**日志、报告、插件等框架封装**

由于pytest是只专注于用例执行的框架，如果需要增加更多的功能，则需借助第三方插件

IPO的作用

使输入简单：用例不再是代码，而是数据；使输出简单：不再是文本，而是精美的网页；使处理灵活：可以记录日志、或者发送通知

**接口自动化**

接口测试的本质：遵循接口协议，完成数据的收发，进行断言

接口自动化测试的本质：自动化数据收发（requests），自动化数据断言（responses-validator）

**补充：**

[在用例前写一行@pytest.mark.skip](mailto:在用例前写一行@pytest.mark.skip)即可跳过下面的这个用例，在使用前需要导入pytest，即需要写一行import pytest



**总结：**

pytest是测试的基础框架，是首要学习的。

用例的建立需要以函数形式存在，即测试框架中的函数被称为用例（有自己独特的名称，可能在前端和后端也有不一样的名称（我猜的））。

对requests的使用需要放到自己创造的函数（即用例）中，且只能在函数中使用，一是方便自己使用，二是方便寻找（函数名称不同）。

针对不同的细节、框架（说人话就是面对各种情况），建议进行不同的项目练习，这样才代表自己了解，之后才能说明自己掌握了