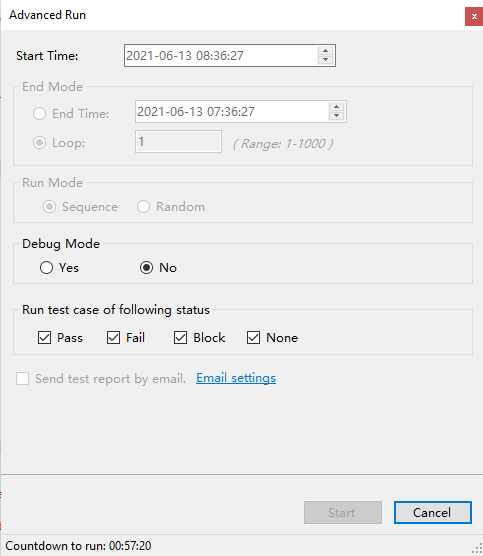
# AutoTest Studio入门系列4：任务调度

AutoTest Studio有一个强大的任务调度引擎，可以让用户灵活的运行测试任务，如，定时运行任务，定量运行任务，运行指定状态的测试用例。

为例便于演示，我们将第二节中的测试用例复制5份，同时将caseID也对应修改为：TEST-1, TEST-2, TEST-3, TEST-4, TEST-5。

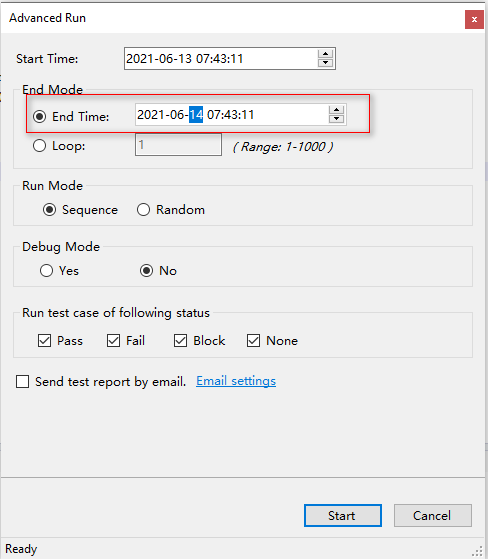
**定时运行**

当设定的Start Time大于当前时间时，AutoTest Stduio会进入倒计时，直到到达指定时间时，才会运行，如果start time小于当前时间时，则会立即运行任务。



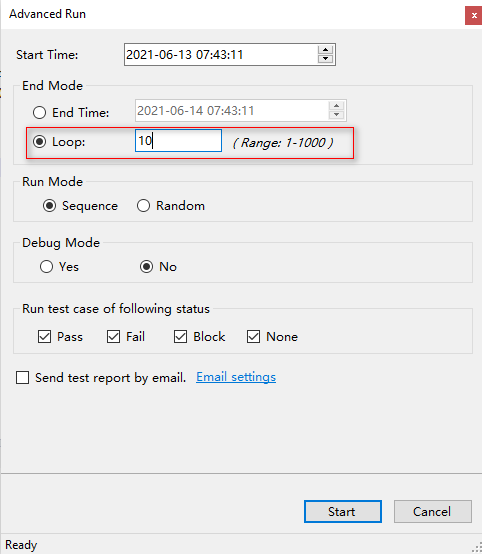
**定时结束**

AutoTest Studio允许设定结束时间，当到达结束时间时，程序立即终止任务运行，这里你不用担心AutoTest Studio会暴力的结束任务，事实上，结束时间到达后，AutoTest Studio会运行完当前测试用例后在终止任务。



**指定运行的次数**

AutoTest Studio除了提供指定时间结束任务，也提供指定运行次数，如下图，AutoTest Studio会将任务中的所有用例运行10次后终止任务。

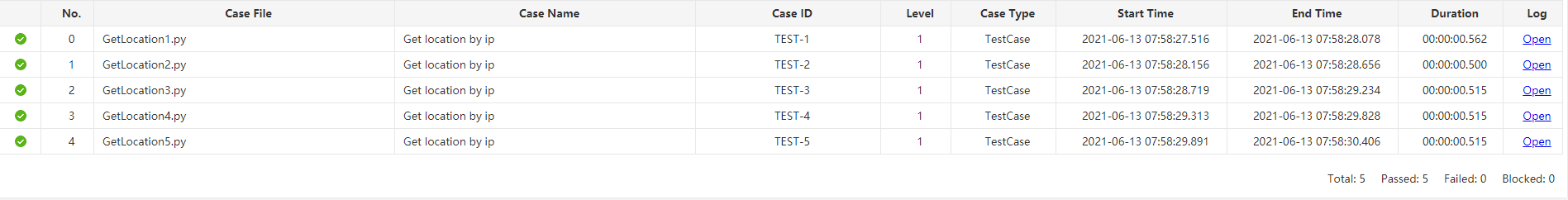


**运行模式**

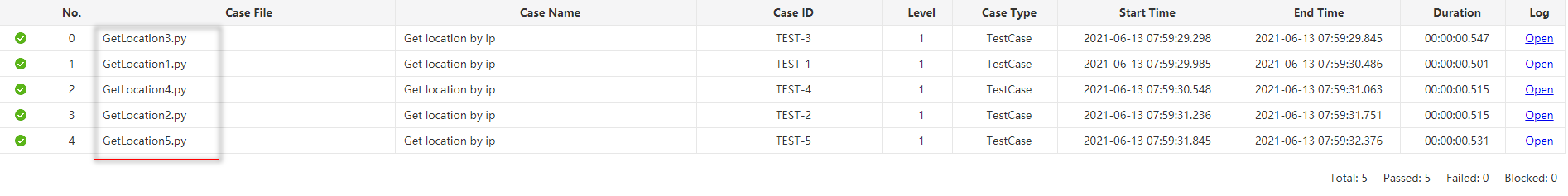
AutoTest Studio默认情况下会按顺序运行所有用例，但也提供了随机方式运行用例。

**Sequence**，顺序运行用例；**Random**，随机方式运行用例。

顺序运行的结果：



随机运行的结果

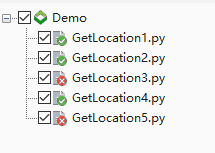


**调式运行**

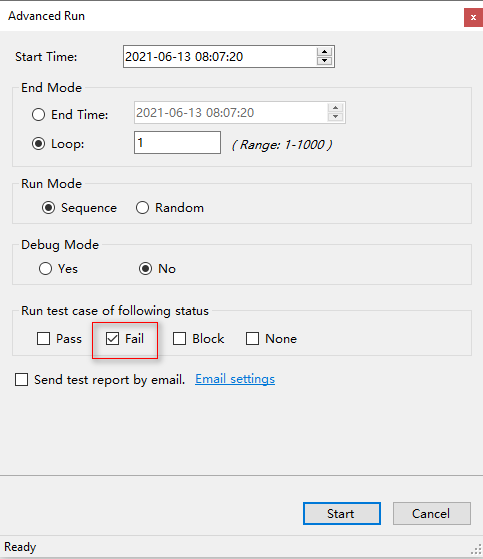
当Debug Mode 设置为“Yes”时，AutoTest Studio可以以调式方式方式运行测试用例，这个功能在我们定位一些“奇怪”问题时非常有用，我们可以让用例运行到特殊条件时进入断点。

**运行指定状态的用例**

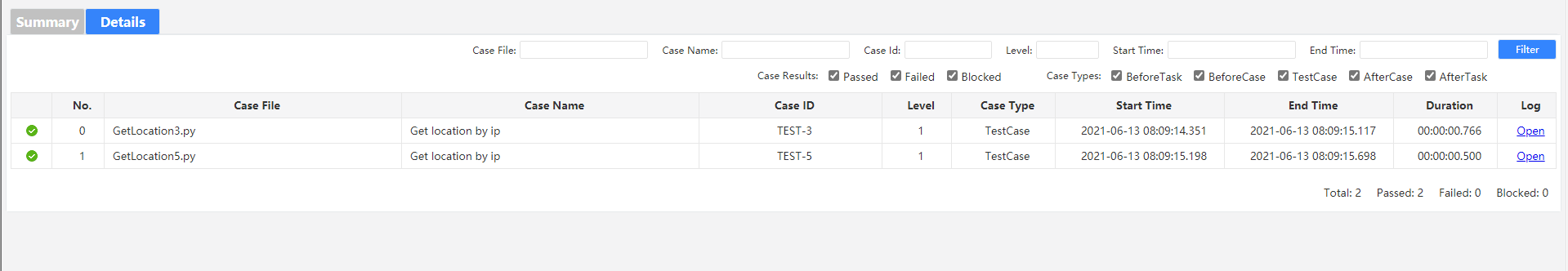
AutoTest Studio可以运行指定状态的用例，比如运行上一轮fail的用例。我们假定上一轮运行的结果如下图所示。



在“Advanced Run”的“Run test case of following status”设定只运行“Fail”状态的用例。

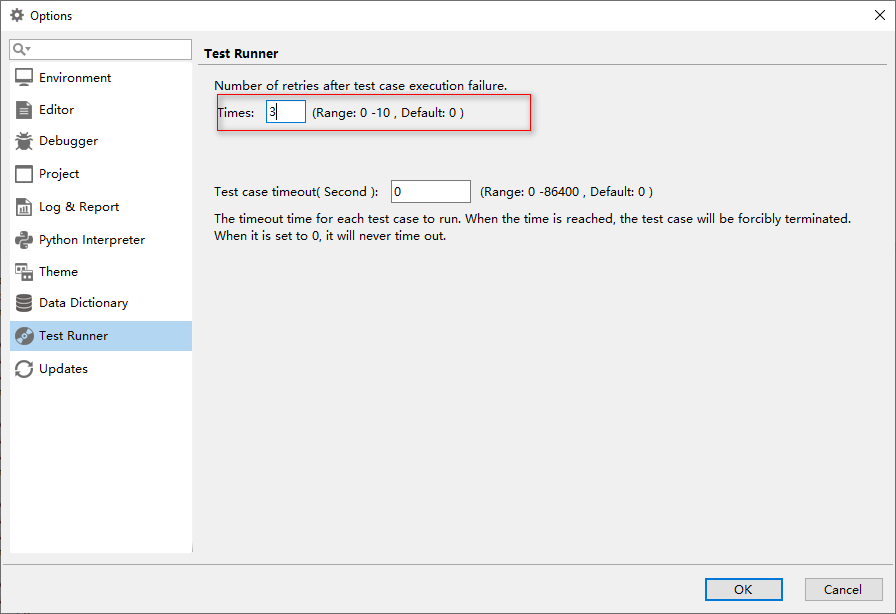


下面是运行的结果，可以看到AutoTest Studio只运行了“GetLocation3.py”，“GetLocation5.py”两个,即上一轮运行失败的用例。



**失败后重新运行**

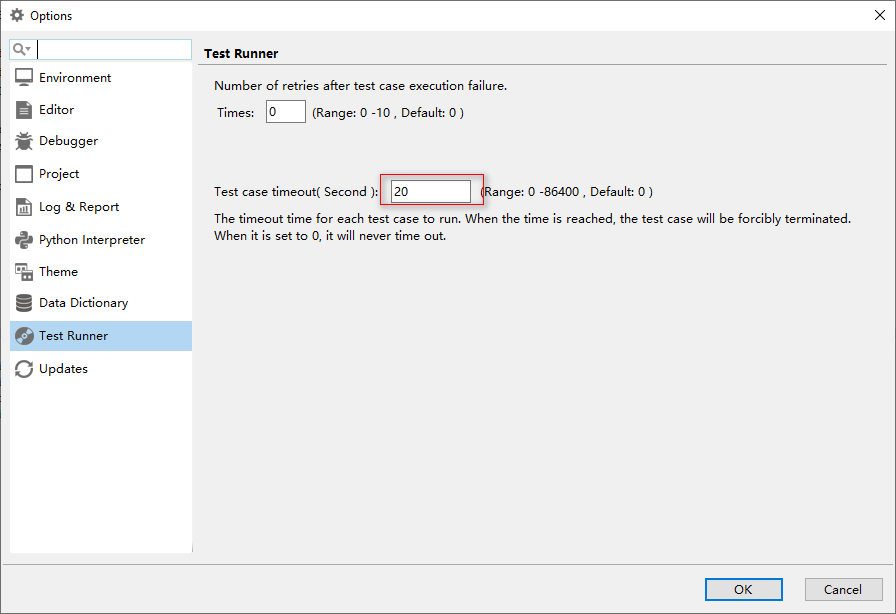
AutoTest Studio提供一个全局配置参数，允许用例运行失败后，重新运行，并能够指定重复运行的次数。默认为0，即不重复。



**用例运行超时设置**

在很多时候，可能因为一些异常原因，导致测试用例运行阻塞，但我们不希望因为这个用例的阻塞导致整个任务被阻塞，因此我们可以设置测试用例超时时间来强制中止阻塞的测试用例，继续运行其他用例，但这里需要注意的是，超时时间不能设置得过小，因为即使用例正常运行时，如果超过这个超时时间，也同样会被终止运行。

假定我们设置用例用例的超时时间为20秒，我们在用例使用time.sleep(25)来阻塞用例。



示例代码：

import time

time.sleep(25)

运行日志结果：

