# [下载HTMLTestRunner 地址](https://www.cnblogs.com/benpao1314/p/9633862.html)

通过pip安装 HTMLTestRunne失败。则需要通过手动下载。

下载地址:http://tungwaiyip.info/software/HTMLTestRunner.html

下载后，把HTMLTestRunner.py 文件复制到Python 安装路径下的lib文件夹中即可。

**在python3中用HTMLTestRunner.py报importError“：No module named 'StringIO'解决办法**

原因是官网的是python2语法写的，看官手动把官网的HTMLTestRunner.py改成python3的语法。  
  
修改内容：

第94行，将import StringIO修改成import io

第539行，将self.outputBuffer = StringIO.StringIO()修改成self.outputBuffer = io.StringIO()

第642行，将if not rmap.has\_key(cls):修改成if not cls in rmap:

第631行，将print >> sys.stderr, '\nTime Elapsed: %s' % (self.stopTime-self.startTime)修改成print(sys.stderr, '\nTime Elapsed: %s' % (self.stopTime-self.startTime))

第766行，将uo = o.decode('latin-1')修改成uo = e

第775行，将ue = e.decode('latin-1')修改成ue = e

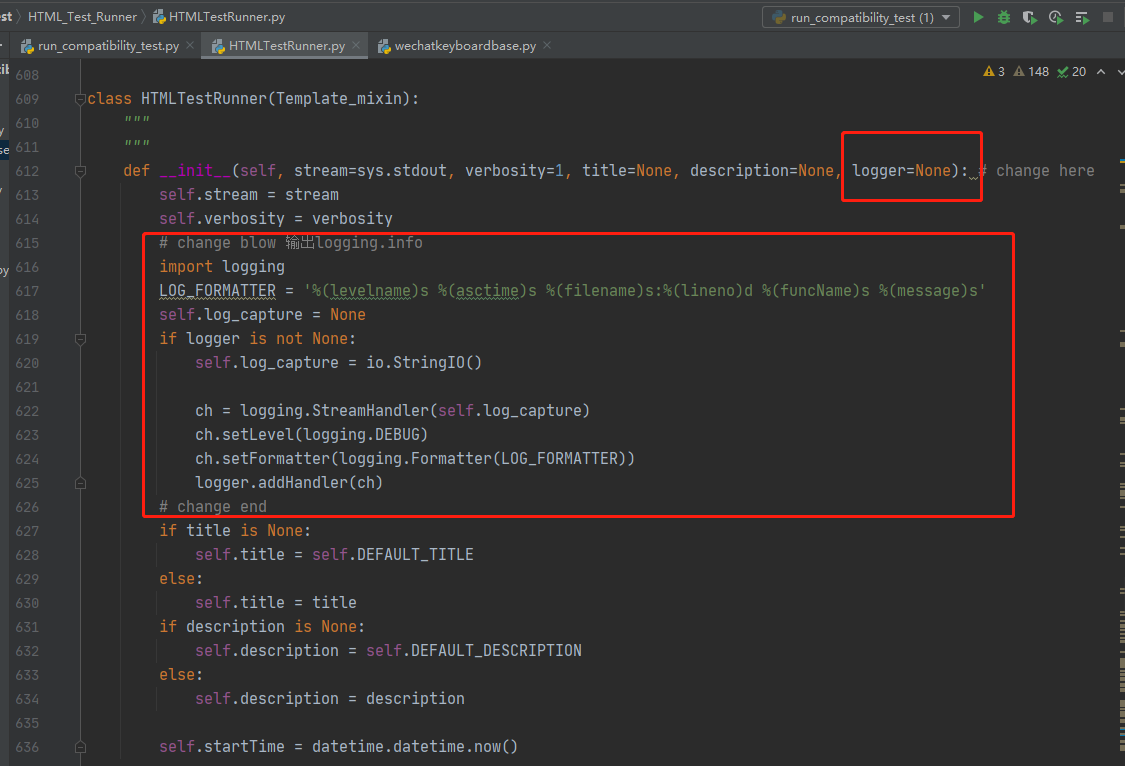
# Python3下的HTMLTestRunner测试报告中文乱码

解决方式：在源文件中添加utf-8解码【ue = e.decode('utf-8')】

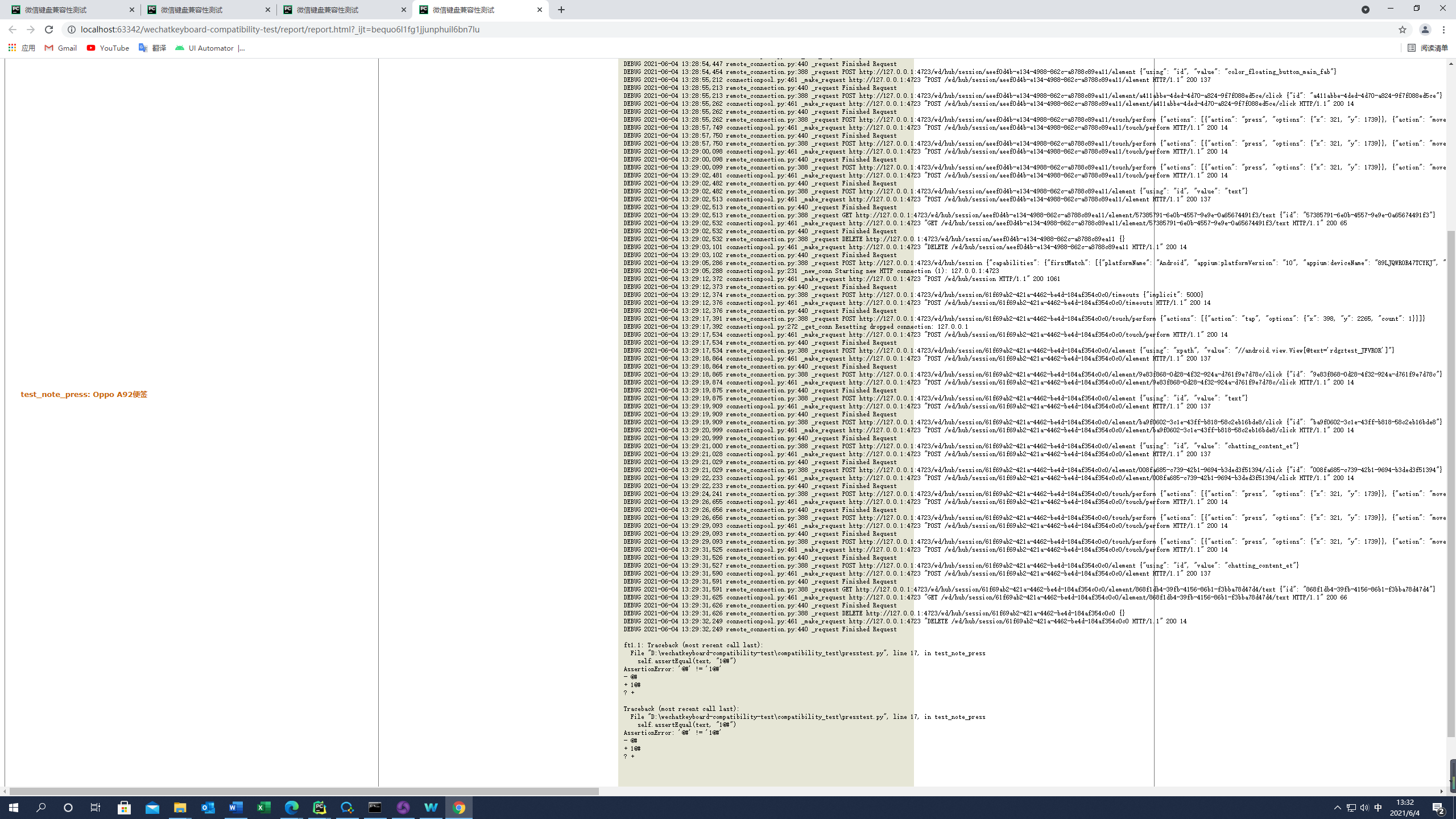
# 问题：使用unittest进行单元测试，用HTMLTestRunner生成html的测试报告，希望报告中能输出logging.info等日志

**前提：**HTMLTestRunner版本是从网上下载的，版本0.8.2，使用的是python3

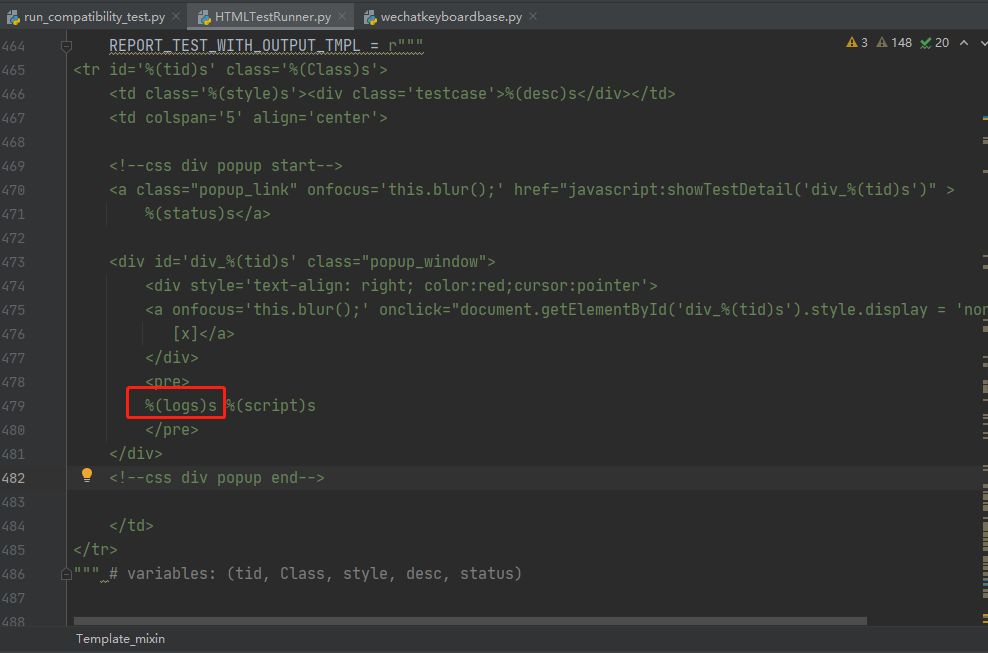
**解决方案：**改HTMLTestRunner.py文件，改动地方如下：

1.添加新logger参数，传递给HTMLTestRunner类的构造函数和附加一个新的处理程序记录器（也就是logging.Handler）：

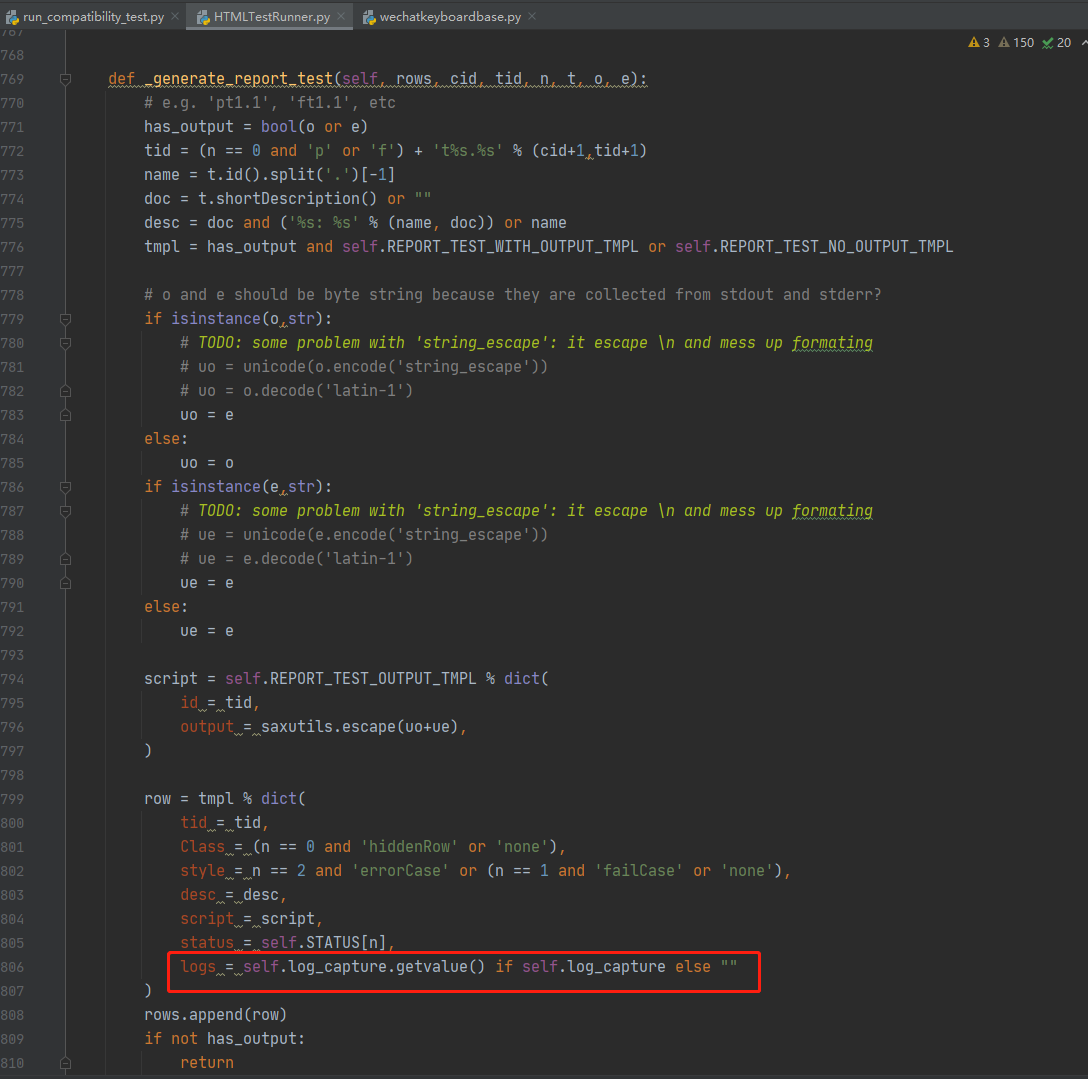
2.我希望在测试用例失败的时候打印日志，类似这样



编辑REPORT\_TEST\_WITH\_OUTPUT\_TMPL模板，添加占位符日志消息%(logs)s



3.改变\_generate\_report\_test方法，填写logs内容



至此，HTMLTestRunner改完

4.在调用HTMLTestRunner生成报告时，传入logger日志器就可以了



**讨论：**这里参考了https://stackoverrun.com/cn/q/13119187，文章估计是用python2，而且打印位置不是我想要的，所以调整了一下。

本文只能在fail或者error的时候才能打印logging，想要在pass的时候也打印的话，还需要再研究下

难点：Appium+Python+Unittest 每次执行完用例后，就会重新启动应用。有没有什么方式来避免这种重新启动？

使用setUpClass和tearDownClass

难点：Appium实际测试中处理界面元素的异常，心得分享

Appium是一款极其强大的移动端自动化测试工具，它不仅支持多种语言，并且可以使用同一套API在不同操作系统平台上进行测试脚本的编写，完美的兼容了android和ios；但是要真正掌握并熟练Appium并不是一个简单的事情，在实际工作中往往会遇到学习过程中无法想象的情况，这次就分享一下我的工作心得

最近一直忙于整理公司接口测试框架，Appium被我疏远了好一阵子，前两天领导说以后可能要开展app端的自动化测试，于是今天花了一个下午梳理了一遍，最痛苦的莫过于前期规范的设计，几乎都是从0开始，好了不扯闲话，下面主要说一下遇到的几个问题：

众所周知appium是依赖于selenium的移动端自动化测试框架，所以它里面的方法基本上和selenium是一样的，不同的是它模拟的是app端的驱动而selenium是web端浏览器的驱动；通过脚本来实现模拟事件操作，所以对手机界面元素的定位对于appium测试来说是最重要的一个环节；appium提供的主要定位方法有xpah，id，name，classname以及对应控件中content-desc属性的accessibility\_id方法；这些方法都有两个，个返回一个webelement对象（就是你要模拟操作的界面元素），另外一个返回一个包含多个webelement对象的列表（适用于没有唯一值属性的元素，通过遍历获得要操作的元素）；android系统可以通过谷歌原生的uiautomatorviewer（配置好androidsdk环境后即可使用）进行定位，如图所示

ios系统使用mac版appium自带的工具定位即可。

那么我们实际工作中会遇到什么棘手的问题呢？模拟的事件操作都是发生在元素定位后，那如果这个元素不存在呢？这是个很常见的场景，需求的更改或者开发自己的变动都会导致这种情况出现，

如果我们写好的脚本无法实现持续通用的话自动化测意义就不大了，所以我们要在appium的源码之上封装一层自己的代码来实现这些异常处理，原理很简单--就是在每次获取界面元素时判断一下是否存在

如果存在返回一个可操作的对象，如果不存在则进行异常处理；我们运气不错，强大的appium给我们提供了解决这个问题的异常判断NoSuchElementException.

还有一个问题就是time，我们每次去定位元素时都是在界面加载完成后才开始的，如果你不想每次刷新，跳转或重新打开界面后都来一句time.sleep（）来设置等待时间（这样也确实影响脚本美观）的话；

那就一起在封装层实现吧，appium这时又体现出它的强大之处了，它提供了一个全局的隐式等待时间implicitly\_wait(time)，设置后appium驱动对象的生命周期内去定位元素都会使用此处的时间来作为最大等待时长

双击？

# BUG：

# 小米手机和oppo手机appium启动时报错:writing to settings requires:android.permission.WRITE\_SECURE\_SETTINGS

报错信息：

java.lang.SecurityException: Permission denial: writing to settings requires:android.permission.WRITE\_SECURE\_SETTINGS

解决办法：

小米：在开发者选项里，把“USB调试（安全设置）"打开即可。  允许USB调试修改权限或模拟点击

oppo：在开发者选项里，把"禁止权限监控"打开即可。

# Unable to find an active device or emulator with OS 7. The following are available: 5ENDU19821001622

报错原因：

remote Appium时两个参数（安卓版本号与安卓设备名）的顺序反了，导致版本号与手机没有对应上

解决办法：

打开微信进行输入时第一个元素未选中或者未执行：因为微信加载需要时间，而程序执行非常快，需要使用sleep()方法设定一个等待时间

# 忘记返回所需参数导致测试用例未通过

# 440： Login Timeout 登录超时

# NoSuchElementException: Message: An element could not be located on the page using the given search parameters.

# Socked hang up

[python+appium 解决socket hand up](https://blog.csdn.net/cjxiong12345/article/details/115438012)

# NoSuchElementException: Message: An element could not be located on the page using the given search parameters.

原因：1. 元素定位信息错误；2.appium server还未响应，页面中该元素还未存在

# TypeError: auto\_input() missing 1 required positional argument: 'self'

错因：def setUpClass(cls) -> None:  
 cls.auto = cls.auto\_input()

在我们定义函数和类的时候，在第一个参数的位置默认应该添加self，例如add（self，str，str1），然后我们在调用的时候应该首先实例化这个类，如Add().add('aaa')，而非直接调用Add.add('aaa')。

解决方法：def setUpClass(cls) -> None:  
 cls.auto = cls().auto\_input()

## appium 报错：Original error:Could not proxy command to remote server. Original error:socket hang up

原因：我在运行期间，使用appium元素查看器

记录一下今天遇到的一个问题，上周还跑的好好的脚本，今天运行时appium一值报错 soket hang up  导致根本无法获取系统的toast，这样我就没办法断言用例是否通过了。

这个报错我基本折腾了一天时间，真的是坑的很！

首先我以为是什么代理的问题，查了一圈，没问题；然后又以为是appium版本与设备哪个地方不兼容问题，重装最新appium v1.13，问题未解决；再然后问群友，群友也是不知道；再然后查博客，有人说是appium-uiautomator2-server/apks下面什么apk的问题，卸载重装，问题依旧；继续搜博客，testerhome中有个类似报错的文章的评论提醒了我：

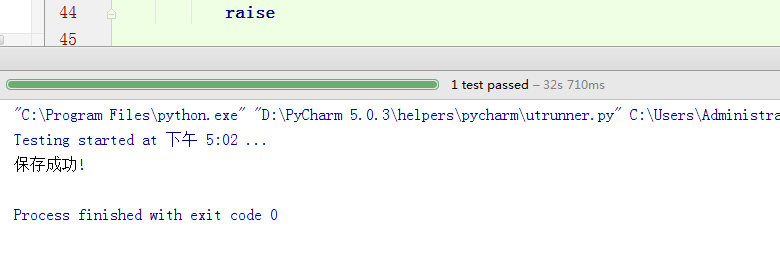
博客链接：<https://testerhome.com/topics/11403?locale=en>

评论中最后一条这样说的：



此时只能司马当成活马医了，果断删除appium seting，卸载重装软件安装包，清楚所有缓存，所有数据等等；然后run

这是惊喜出现了，特么的惊喜出现了， toast成功定位并打印，用例顺利通过！尼玛！从未如此的喜欢绿色，哈哈哈！



到此结束！给遇到相同问题的小伙伴搭个桥，免得再次掉坑！

使用appium自动化测试时，报错：Original error: Could not proxy command to remote server. Original error: Error: socket hang up

去掉uiautomator2后，可执行成功 【我使用了该方法】

desired\_caps['automationName'] = 'uiautomator2'

经分析后，是因为uiautomator2版本不对

解决办法：

在终端执行以下命令：

adb  uninstall io.appium.uiautomator2.server

adb  uninstall io.appium.uiautomator2.server.test

然后重新安装Appium Settings即可解决以上问题

# [appium+python+真机，不能打开APP，报错提示：Cannot start the 'XXX' application](https://www.cnblogs.com/wx170119/p/13131395.html)

appium+python3+真机做自动化测试，连接手机成功后执行自动化代码，执行打开APP的代码时报错，提示信息如下：

Cannot start the 'XXX' application. Visit https://github.com/appium/appium/blob/master/docs/en/writing-running-appium/android/activity-startup.md for troubleshooting. Original error: The permission to start 'XXX' application.

百度了一下，很多小伙伴出现该问题的原因是包名和界面名写错，于是排查了一阵，自己的包名和界面名并没有写错，但是总是打不开，于是打开其它APP，可以正常打开，通过 adb logcat 抓取，能正常抓取到 app 包名和 app 启动后的 activity，如果保证抓取正确，再看有没有报权限不足的异常，如果有那就是权限不足了，需要开发给打开的界面给一个export权限，但是export权限不安全，不想麻烦开发的小伙伴，于是重新获取APP启动界面，于是成功解决该问题

解决：包名跟activity通过adb shell dumpsys window | findstr mCurrentFocus命令获取，以前是adb shell dumpsys activity recents | find “intent={“ 获取

# screenshot\_as\_file(self,filename)将截图保存到指定文件路径：

driver.get\_screenshot\_as\_file("./Screenshots\emoji.png")

时找不到截图

应该使用绝对路径driver.get\_screenshot\_as\_file("D:\wechatkeyboard-compatibility-test\Screenshots\emoji.png")



# urllib3.exceptions.ProtocolError: ('Connection aborted.', RemoteDisconnected('Remote end closed connection without response'))

# 一、运行main函数，未生成测试报告，报错：'\_TestResult' object has no attribute 'outputBuffer'

# 

解决方式：

1、在HTMLTestReportCN.py中

    Ctrl+F搜索\_TestResult方法，在\_\_init\_\_方法中添加：

self.outputBuffer = io.StringIO()  
self.test\_start\_time = round(time.time(), 2)

