1. 准备工作
2. python-2.7.12.amd64.msi
3. node-v6.9.5-x64.msi
4. AppiumForWindows\_1.4.16.1.zip
5. jdk-8u121-windows-x64.exe
6. android-studio-bundle-145.3537739-windows.exe（包含Android SDK）
7. JDK安装

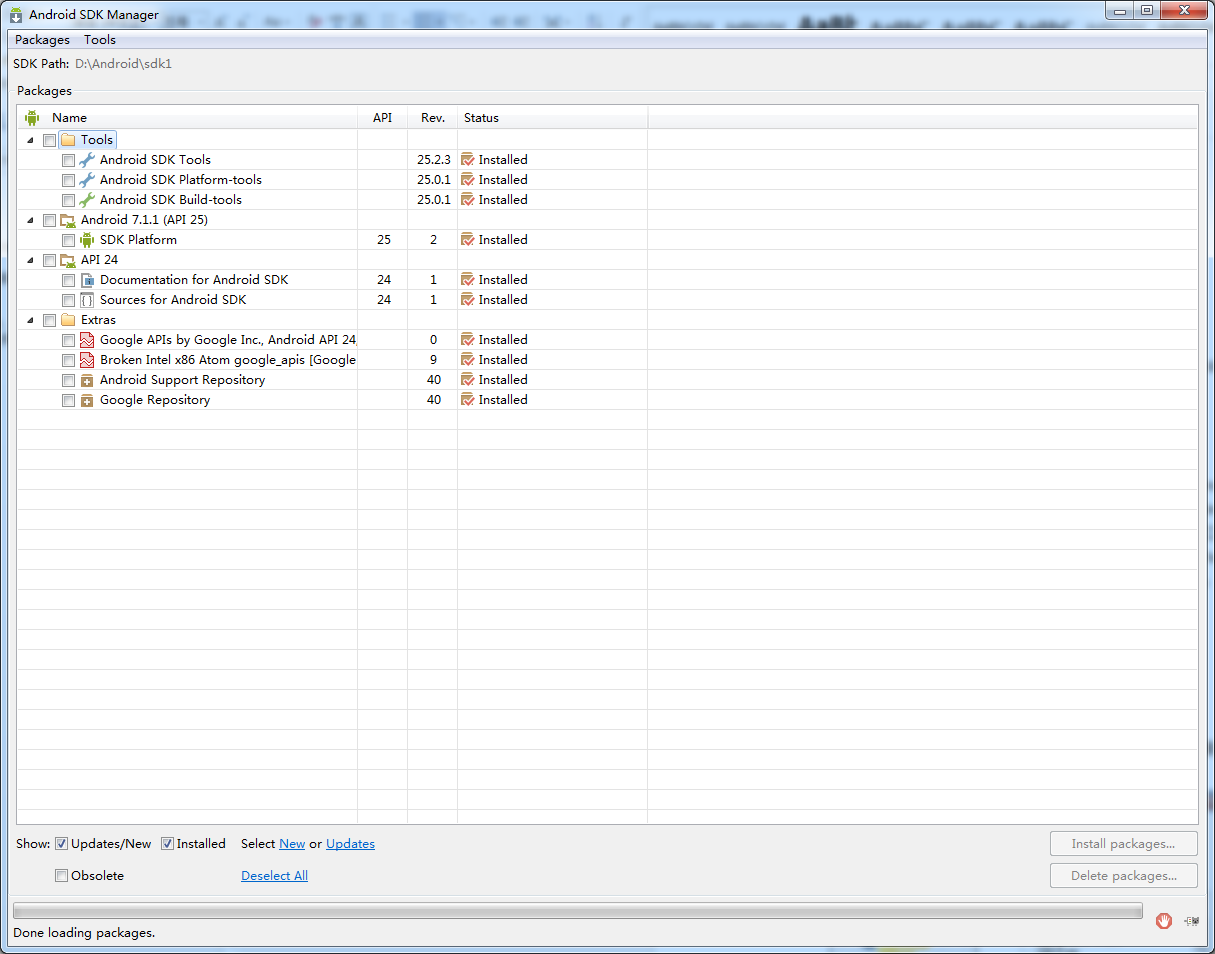
安装完JDK，并配置Java环境变量

1. SDK安装
2. 下载SDK：

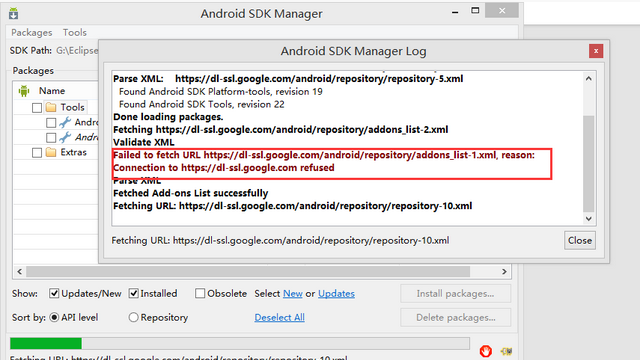
<https://developer.android.com/studio/index.html>

1. 启动SDK Manager.exe

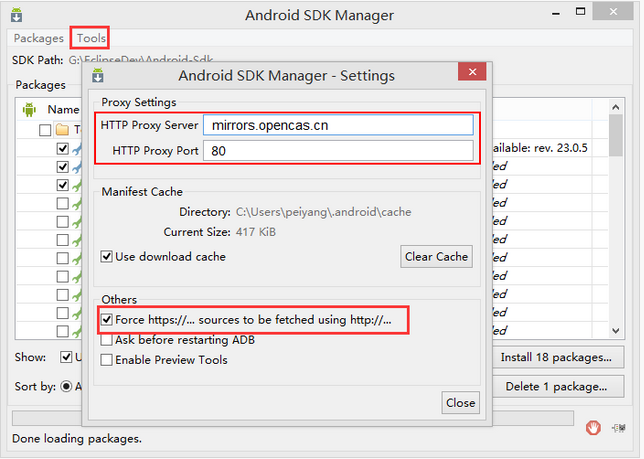
安装完毕后启动Android SDK Manger (SDK Manager.exe，在Android SDK的安装目录下)，安装需要的Android Packages，比如一些工具，需要的Android版本， EABI(模拟器相关的包）等等，工具会自动检测系统当前安了哪些SDK工具，并获取各个版本可安装的SDK，默认情况下只安装了Tools下的Android SDK Tools；



获取可安装的SDK版本时，国内有墙，有时候会出现获取失败的情况，如下图：



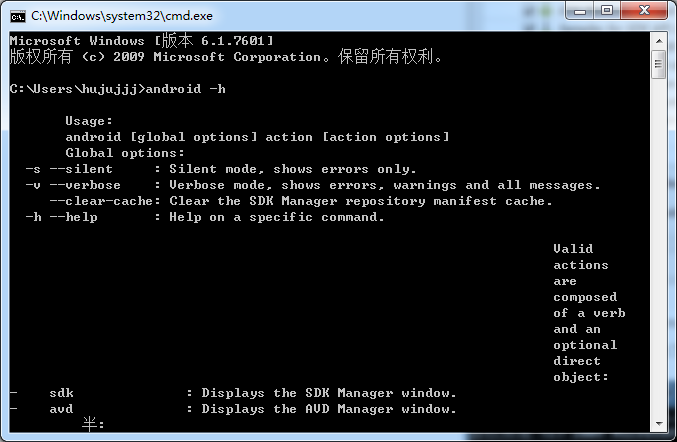
可以通过将HTTP Proxy Server和HTTP Proxy Port，连接服务器使用的代理地址和端口分别修改为“mirrors.opencas.cn”“80” 解决，若依然不能解决，可以从该网站直接下载：<http://www.androiddevtools.cn/>



 修改代理地址和端口后，选择Packages->>Reload，重新获取，等待Android SDK Tools下方提示Done loading packages，即加载完成。选择需要安装的SDK，Appium官方推荐API 17及以上的版本都支持，可以任意安装一个API 17及以上的版本，选择右侧的Install xx packages；

1. 验证是否安装成功：

Cmd→android –h ，出现如下画面，则表示已成功



1. 配置SDK环境变量

添加环境变量ANDROID\_HOME, 值是Android SDK的安装路径； 并把Android SDK下的platform-tools路径和tools路径添加到环境变量Path

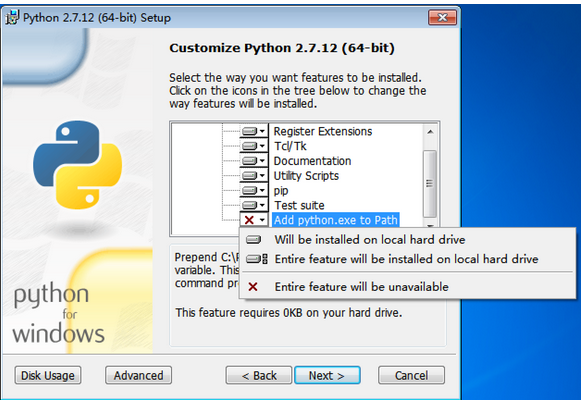
ANDROID\_HOME=D:\Android\sdk1

%ANDROID\_HOME%\platform-tools;

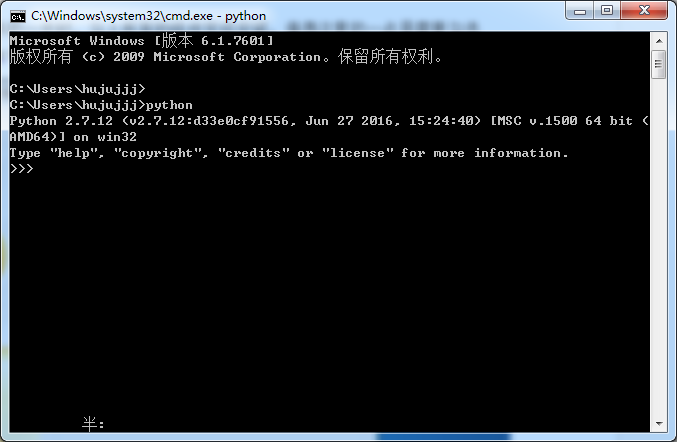
%ANDROID\_HOME%\tools;

1. Cmd运行adb或adb shell 检查安装是否成功
2. Python安装

1. 在选择安装组件的一步时，勾上所有的组件完成安装。值得注意的一点是需要勾选上Add python.exe to Path



2.安装完成后，cmd→Python，出现以下画面，



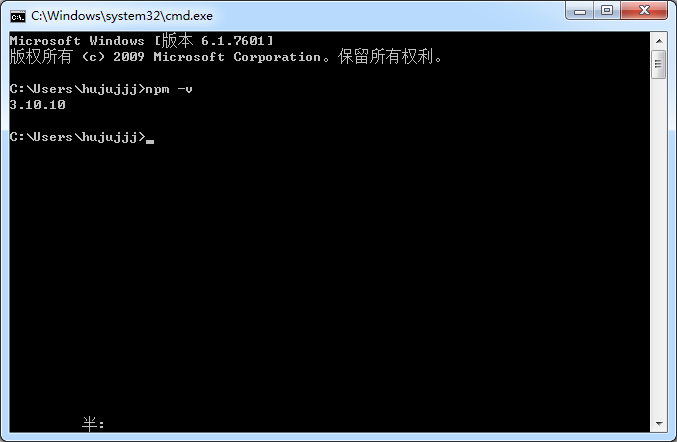
当命令行出现的是： "‘python’不是内部或外部命令，也不是可运行的程序或批处理文件。"，则说明在安装时忘记勾选上Add python.exe to Path。

1. Appium安装
2. Node.js安装

node.js 是javascript和Google V8结合的服务器项目，appium就是node的其中一个开源项目，appiun server端是用node实现，遵循了REST架构，所以appium可以用node的包管理工具npm来进行安装。

Appium使用Node.js作为服务器引擎

安装完后，命令行：npm –v ，看到版本号，表示安装成功



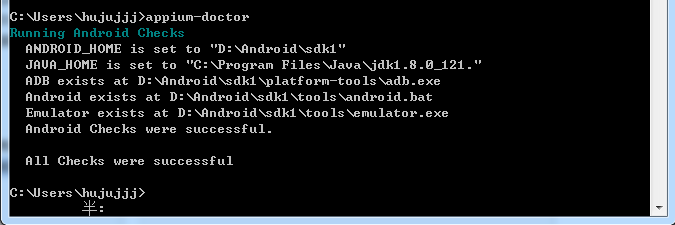
1. Appium安装

双击appium-installer.exe，一直Next默认安装即可，安装完成后，启动Appium，显示如下appium的服务端已安装完成



1. 验证appium安装是否成功

在命令行输入：appium-doctor



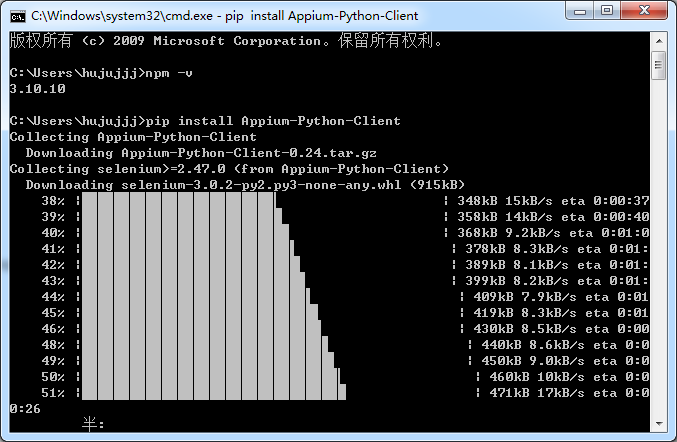
问题一：输入appium-doctor时，提示appium-doctor不是内部命令，意味着appium对应的环境变量未设置。

变量名：APPIUM\_HOME，变量值：D:\nodejs\node\_modules\appium  
变量名：PATH，变量值：%APPIUM\_HOME%\node\_modules\.bin\;

修改变量名后，可能变量名未启用，建议重启电脑。

1. Appium-Python-Client安装

接下来需要安装Appium客户端，由于是基于Python环境，可以可以直接用pip安装Appium客户端，在命令行输入：pip install Appium-Python-Client



等待提示Success。

1. 配置Appium环境变量

由于使用的是第三方安装，所以还需要手动配置Appium的环境变量

设置方法：选中计算机点击右键-->属性-->高级系统设置，选中高级-->系统变量

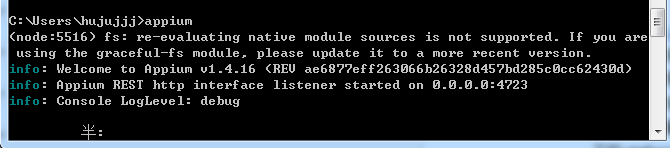
新建系统变量，变量名为"APPIUM\_HOME"，变量值为"D:\Program Files (x86)\Appium";

再编辑PATH，添加变量值";%APPIUM\_HOME%\node\_modules\.bin;"

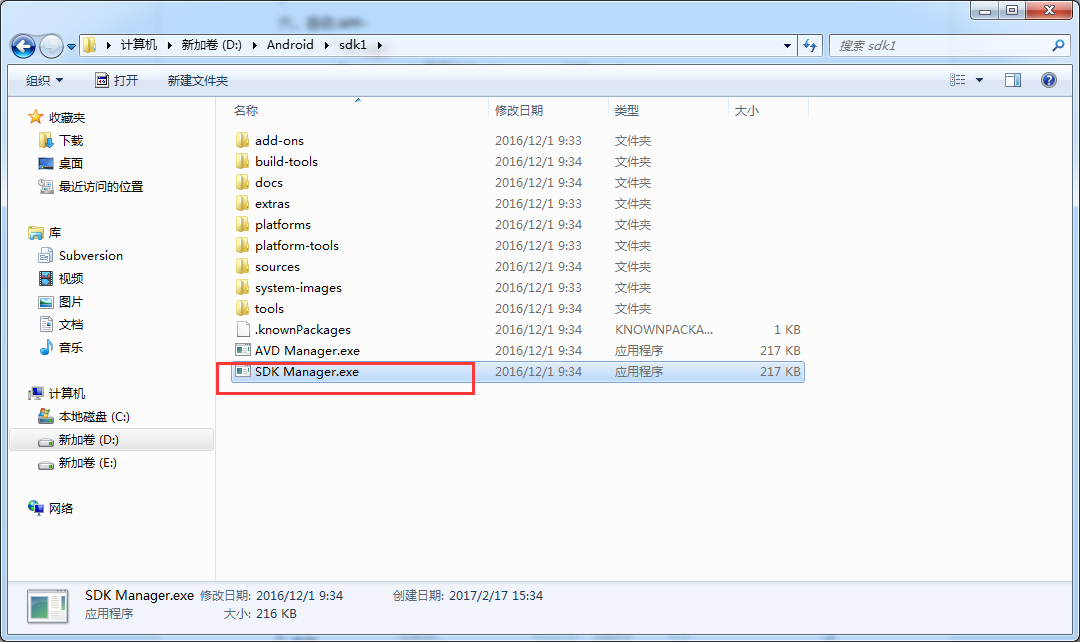
环境变量配置完成后，

1. 验证是否成功

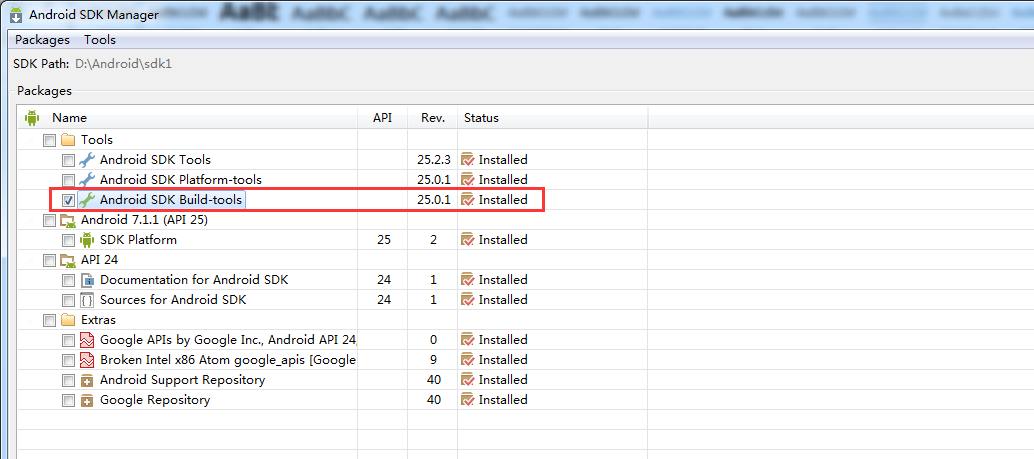
Cmd中，运行命令：appium。默认端口：4723，运行状态如下：



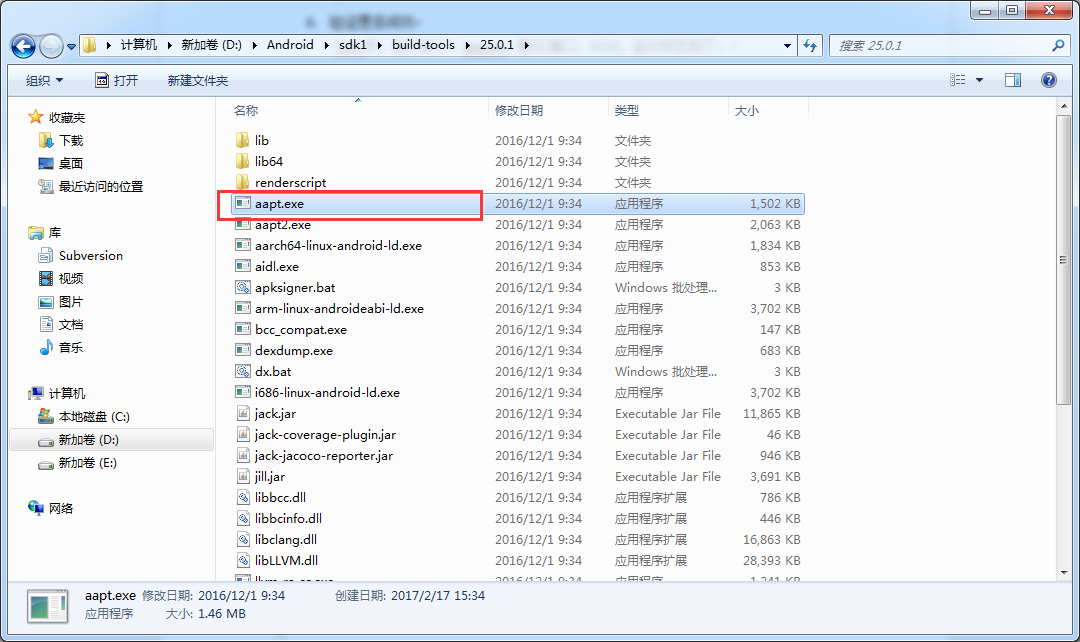
1. 启动APP
2. 下载aapt
3. 在android-sdk里面双击SDK-manager,下载buidl-tools



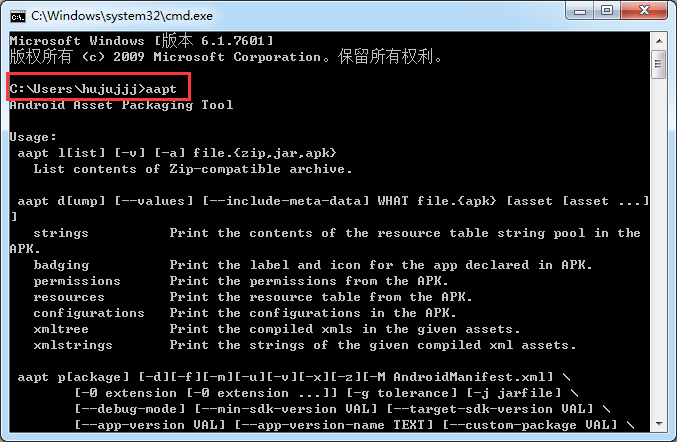
1. 勾选build\_tools，这里可以随便选一个版本



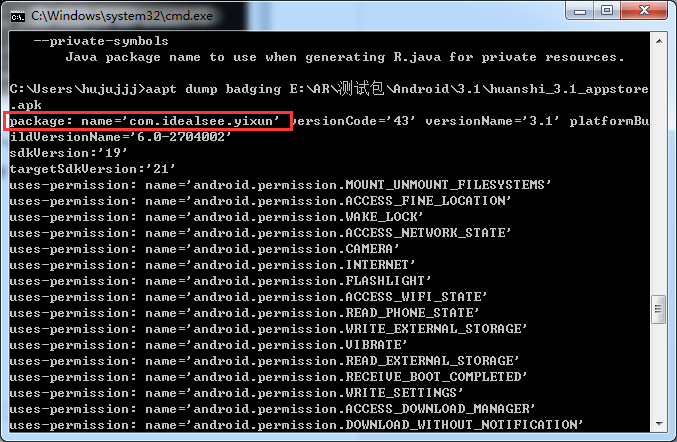
1. 下载完成后，在D:\Android\sdk1\build-tools\25.0.1目录下找到aapt.exe，将这个路径设置环境变量，添加到path下



1. Cmd中输入aapt，出现如下界面，说明环境OK

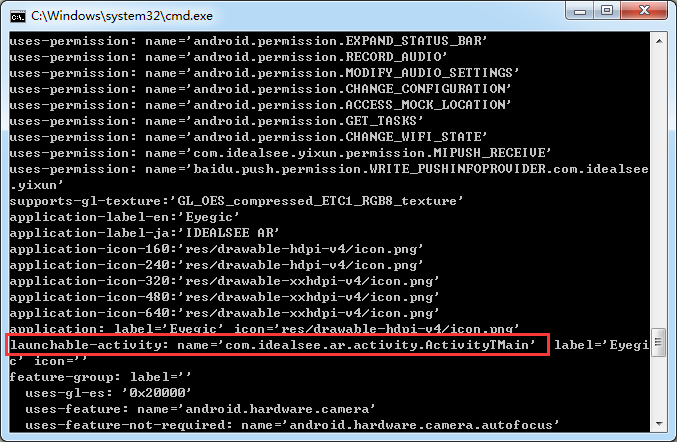


1. 获取apk包名
2. 将准备测试的apk放到某个目录，如“E:\AR\测试包\Android\3.1”
3. Cmd中输入指令：aapt dump badging E:\AR\测试包\Android\3.1\xxx.apk，如huanshi\_3.1\_appstore.apk
4. 以幻视APP为例，如下图，可以得到幻视apk的包名：com.idealsee.yixun



注：也可以将apk放在桌面上，输入指令后，拖入cmd框汇总。

1. 获取launcherActivity
2. 接上一步，将cmd拖到中间，找到LauncherActivity，可以看到LauncherActivity值为“com.idealsee.ar.activity.ActivityTMain”



1. 写脚本

**from** appium **import** webdriver  
desired\_caps = {  
 'platformName': 'Android',  
 # 手机设备名称，通过adb device查看；  
 'deviceName': '5f1f945c',  
 # Android系统版本号  
 'platformVersion': '6.0.1',  
 # APK包名  
 'appPackage': 'com.idealsee.yixun',  
 # APK的launcherActivity  
 'appActivity': 'com.idealsee.ar.activity.ActivityTMain'  
 }  
driver = webdriver.Remote('http://127.0.0.1:4723/wd/hub', desired\_caps)

1. 允许appium
2. 启动appium，右上角点击三角按钮，变成正方形，就是启动状态



1. 确认手机连接上电脑

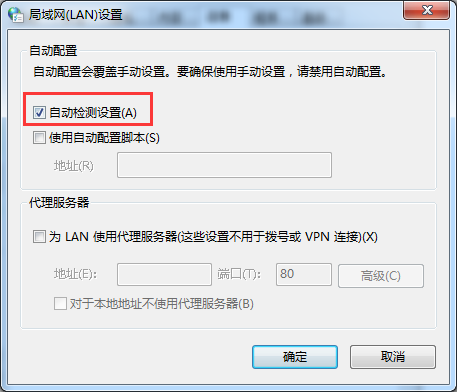


1. 在Pycharm上运行脚本
2. 问题解答

问题一：

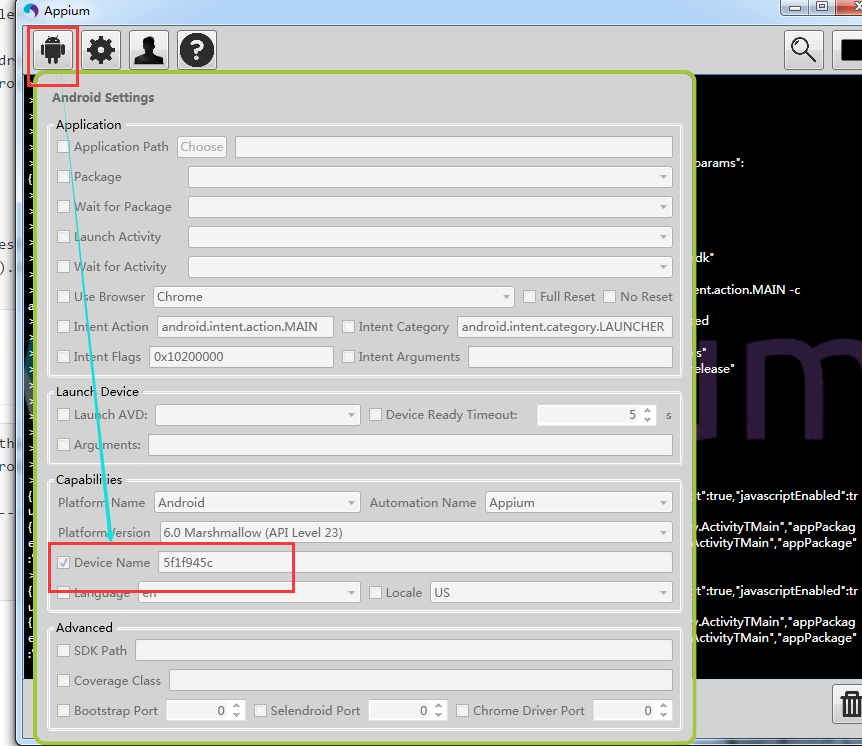
运行报错：urllib2.URLError: <urlopen error [Errno 10061] >

解决方法：打开IE浏览器，依次选择 工具——Internet选项——连接——局域网设置，取消代理服务器复选框，选中自定检测设置，重启IE即可。



问题二：

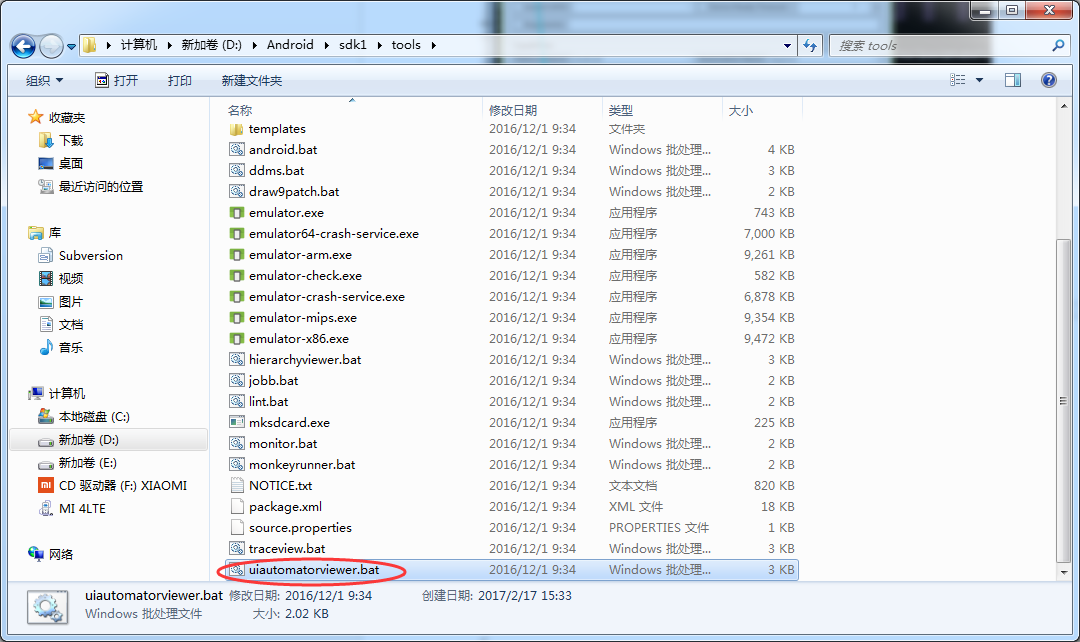
在使用前，appium中Android Settings 窗口勾选Device Name，并填入设备名称，不然运行脚本时会报错，APP启动不起来。



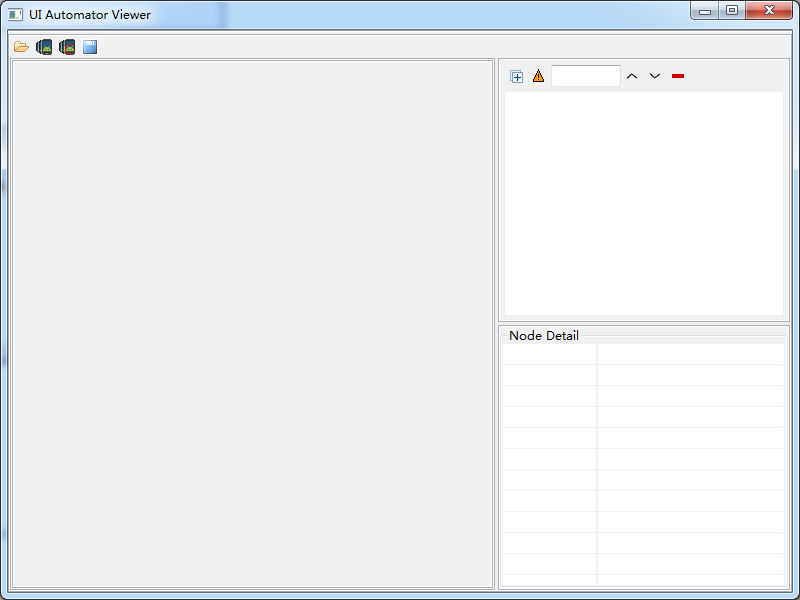
1. 元素定位uiautomatorviewer

Uiautomatorviewer是Android-sdk自带的一个元素定位工具，使用uiautomatorviewer，可以检查一个app的布局和组件以及相关属性

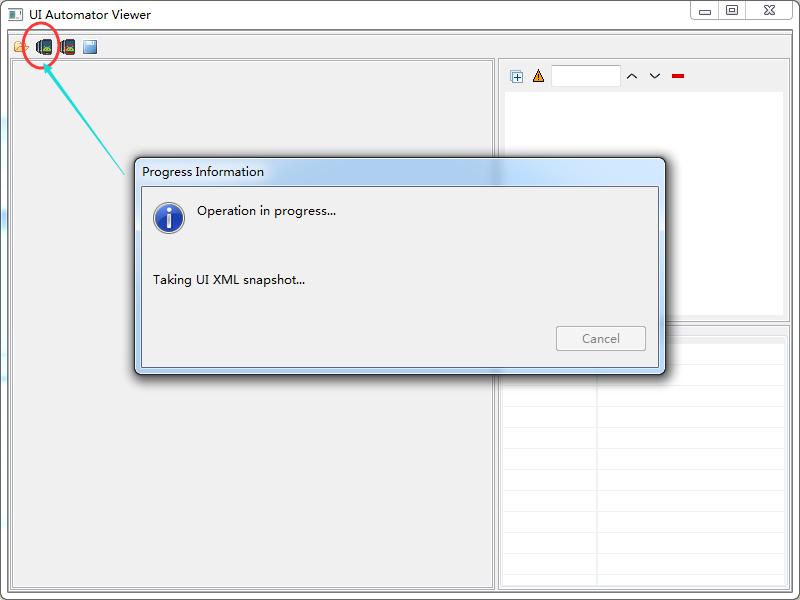
1. 启动uiautomatorviewer.bat
2. 在Android sdk目录下的tools文件夹中找到uiautomatorviewer.bat



1. 双击启动，或者在cmd中通过指令启动：先cd到tools目录，然后 输入 uiautomatorviewer.bat 回车即可。启动后界面如下

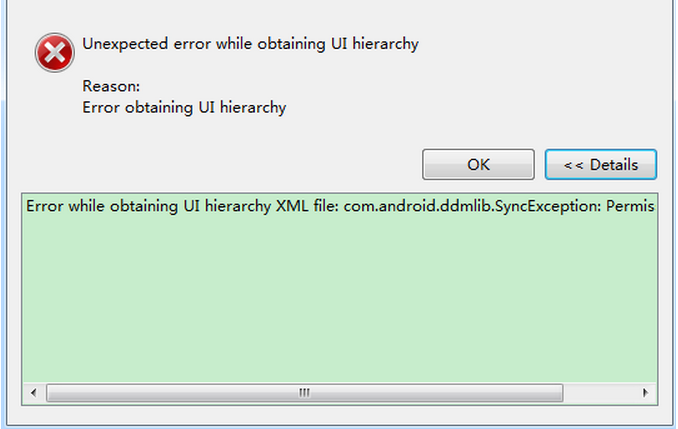


1. 连接手机
2. Cmd中输入adb devices，确认手机已连上
3. 打开幻视app，使得屏幕处于高亮
4. 点击左上角Android机器人按钮Devices Screenshot按钮刷新页面

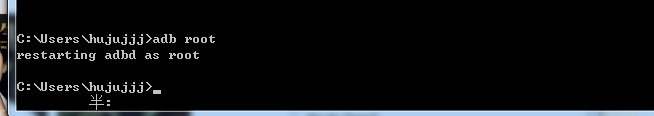


问题：

报错：Error while obtaining UI hierarchy XML file: com.android.ddmlib.SyncException: Permission denied Error while obtaining UI hierarchy XML file: com.android.ddmlib.SyncException: Permission denied



解决办法：在cmd命令窗口中输入 adb root即可



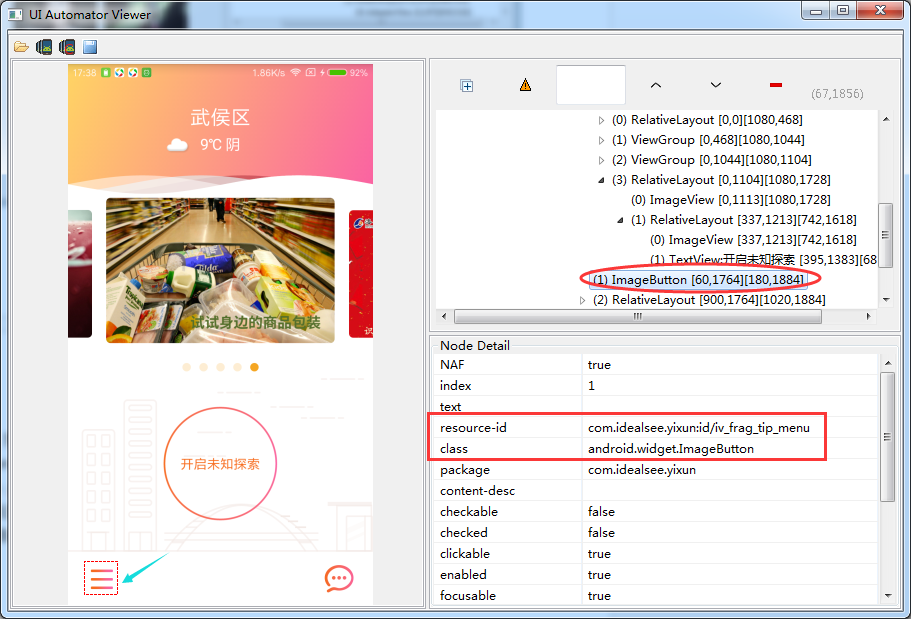
1. 定位元素

移动鼠标到需要定位的元素上，如幻视首页左下角的设置按钮，得到

ImageButton：[60,1764][180,1884]

Resource-id：com.idealsee.yixun:id/iv\_frag\_tip\_menu

Class：android.widget.ImageButton

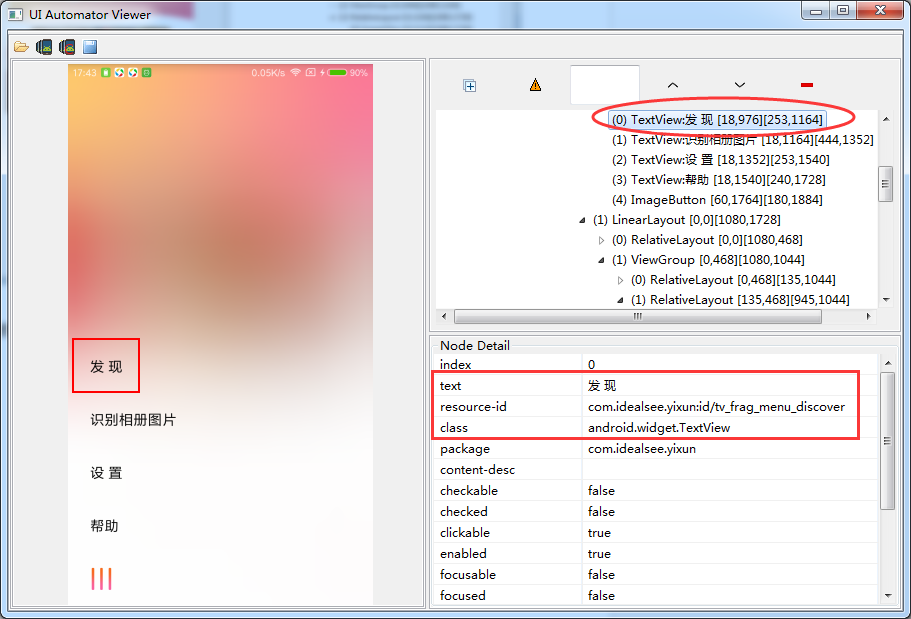


在设置菜单栏的“发现”

Text：发现

Resource-id：com.idealsee.yixun:id/tv\_frag\_menu\_discover

Class：android.widget.TextView

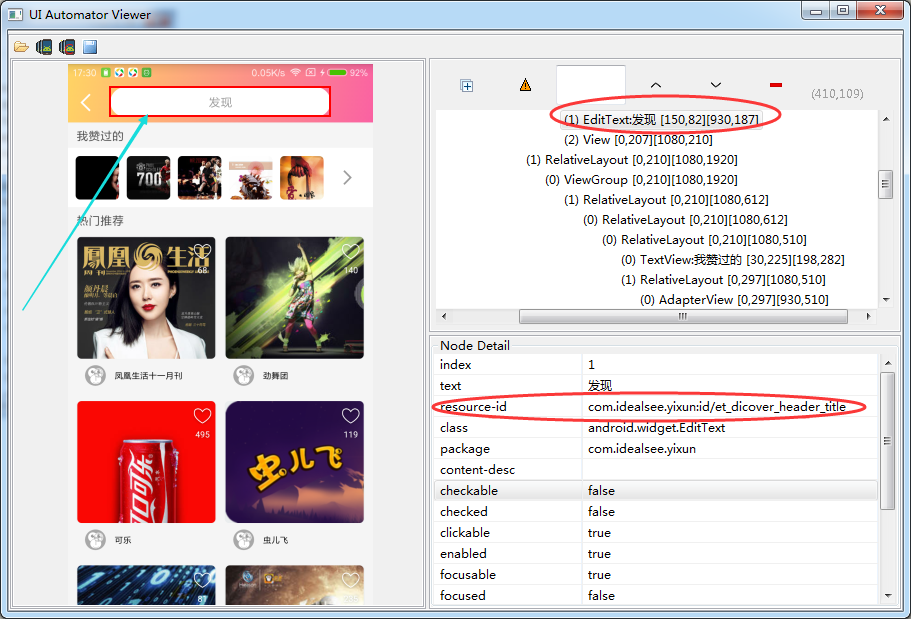


发现页的搜索框

Text：发现

Resource-id：com.idealsee.yixun:id/et\_dicover\_header\_title

Class：android.widget.EditText



1. 点击元素代码

前面启动app后，休眠5s，等待页面加载完成

time.sleep(5)

driver.find\_element\_by\_id(“com.idealsee.yixun:id/iv\_frag\_tip\_menu”).click()

time.sleep(5)

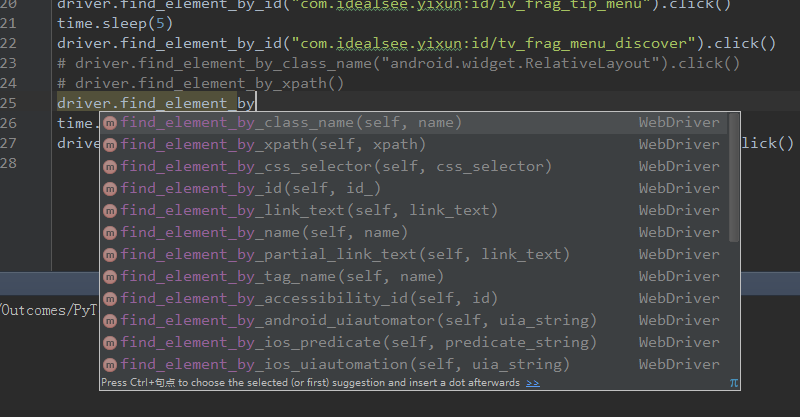
driver.find\_element\_by\_id(“com.idealsee.yixun:id/tv\_frag\_menu\_discover”).click()

time.sleep(5)

driver.find\_element\_by\_id(“com.idealsee.yixun:id/et\_dicover\_header\_title”).click()

1. 关于元素定位的说明

appium的webdriver提供了11种元素定位方法，在selenium的基础上扩展了三个，可以在pycharm里面输入driver.find\_element\_by然后会自动匹配出来



uiautomatorviewer是android sdk自带的，下appium自带个元素定位工具：Inspector

但是使用appium自带的定位工具容易遇到以下问题

A：在使用过程中经常报错：连接服务器失败

B:每次启动都会在手机上重新安装一遍应用

C:不能鼠标指定某个元素，没有uiautomatorviewer方便

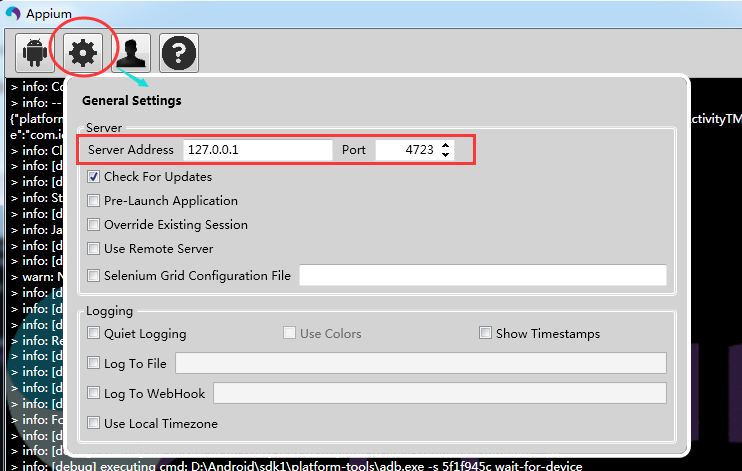
基于以上原因，这里就不写它了(\*^\_\_^\*) 嘻嘻……

1. Remote远程控制

如果自动化工具运行和编写脚本在不同的机器上进行，即远程控制自动化运行，则可以通过remote来达到目的

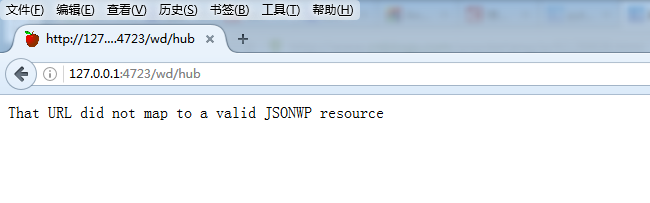
1. 设置IP

打开appium→General Settings界面，，这里的127.0.0.1是默认服务端地址，如果在本机上运行则无需修改



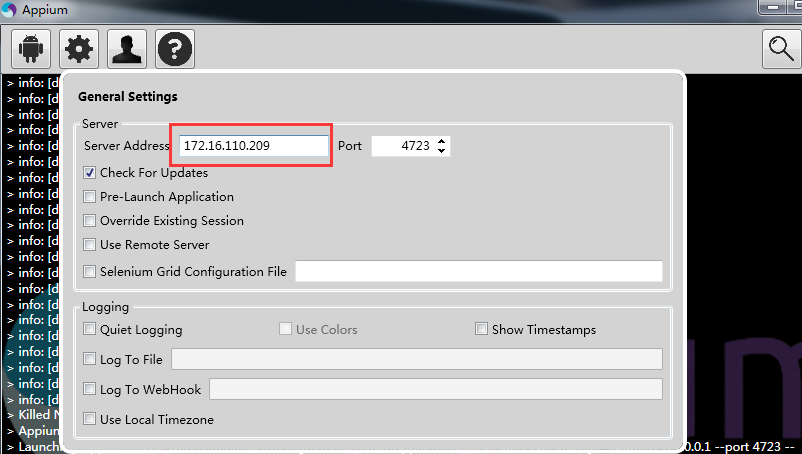
1. 访问地址

启动appium后，在浏览器输入：<http://127.0.0.1:4723/wd/hub>，出现如下信息，则表示服务启动成功，可以appium看出是一个服务端



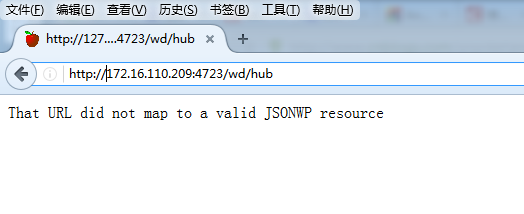
1. 配置测试机

将测试机上的appium服务地址改成本机IP，如：172.16.110.209



1. 远程操作

启动测试机上过的appium，然后在本机上的浏览器输入：<http://172.16.110.209:4723/wd/hub>，确认远程链接成功



1. 在脚本里代码中修改地址

driver = webdriver.Remote('http://172.16.110.209:4723/wd/hub', desired\_caps)

1. 打印HTML 测试报告
2. 加载HTMLTestRunner.py

打印HTML测试报告，需要引入HTMLTestRunner，HTMLTestRunner是Python标准库中单元测试模块的扩展，它生成易于使用的html测试报告

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

下载HTMLTestRunner.py文件后，把HTMLTestRunner文件放到C:\Python27\Lib的目录下即可。 运行cmd，输入python，导入import HTMLTestRunner，无任何的错误提示信息，就表示成功

1. 定义自动化测试报告

代码：

**import** HTMLTestRunner

filename = "./eyegiclog.html" # 定一个报告存放路径，支持相对路径  
fp = file(filename, 'wb')  
runner = HTMLTestRunner.HTMLTestRunner(stream=fp, title='Eyegic\_AutomatedTestReport', description='Eyegic\_AutomatedTestReport')

运行后报告

