# [monkeyrunner之环境搭建及实例（三）](http://www.cnblogs.com/lynn-li/p/5885001.html)

    Monkeyrunner工具提供了一个API，使用此API写出的程序可以在Android代码之外控制Android设备和模拟器。

# 一、Monkeyrunner简介

## 1.MOnkeyrunner相对Monkey区别

1）Monkeyrunner工具在工作站上通过API定义的特定命令和事件控制设备或模拟器（可控）

2）精确控制事件之间的事件

3）可以进行：点触屏、拖拽、长按、键盘事件

4）可以智能截图对比和判断

5）回溯出详细具体的BUG路径

## 2.Monkeyrunner优缺点

1） 能完全模拟人工所有操作

2） 有详细的API文档参考

3） 可以写出智能图像对比脚本

4） 支持java和Python两种语言脚本

5） 脚本移植性差

## 3.Monkeyrunner脚本编写

1） 终端USB调成开发者模式

2）电脑安装手机驱动

# 二、Monkeyrunner环境搭建

    Monkeyrunner的环境搭建，需要安装以下工具：jdk、android sdk、python编译器。

## 1.jdk的安装与配置

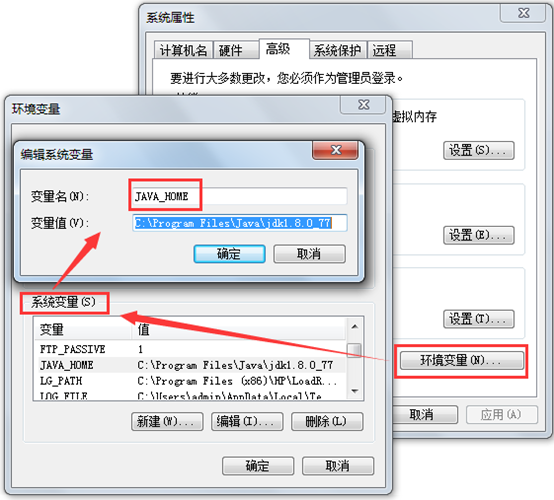
### 1）jdk下载地址

<http://www.oracle.com/technetwork/java/javase/downloads/index.html>

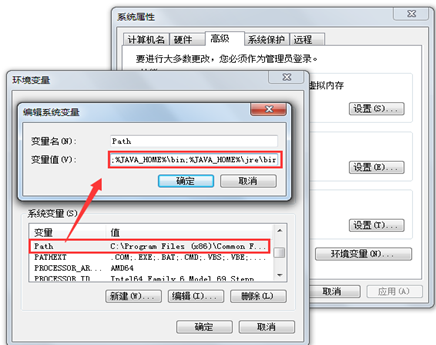
    下载完成后，默认安装即可。

### 2）jdk环境配置

    jdk安装成功后，计算机→属性→高级系统设置→高级→环境变量，在系统变量中，新建JAVA\_HOME变量，变量值填写jdk的安装目录。

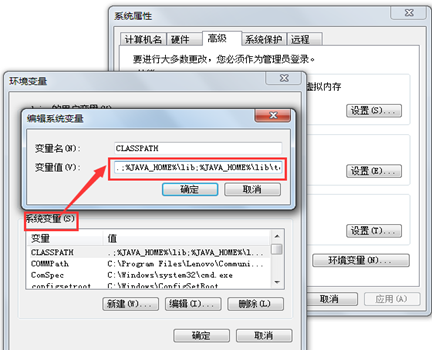


    在系统变量中，编辑Path变量，在变量值最后输入%JAVA\_HOME%\bin;%JAVA\_HOME%\jre\bin;（注意原来Path的变量值末尾有没有;号，如果没有，先输入；号再输入上面的代码）



在系统变量中，新建CLASSPATH变量，变量值填写为：

  .;%JAVA\_HOME%\lib;%JAVA\_HOME%\lib\tools.jar（注意最前面有一点）



    到此，系统变量配置完毕。

### 3）jdk环境检查

    检验jdk环境是否配置成功，则运行cmd，在cmd窗口中，输入 java -version （java 和 -version 之间有空格）。若如图所示，显示版本信息，则说明安装和配置成功。



## 2.android sdk安装与配置

android sdk就是指Android专属的软件开发工具包。android sdk中我们最常用的就是tools和platform-tools文件夹中的工具。

### 1）sdk下载地址

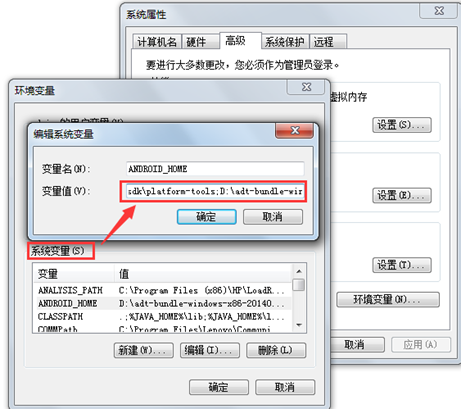
下载地址1：http://developer.android.com/sdk/index.html

下载地址2：http://rj.baidu.com/soft/detail/23485.html?ald

Sdk下载完成后，解压缩到自己的目录，不需要安装。

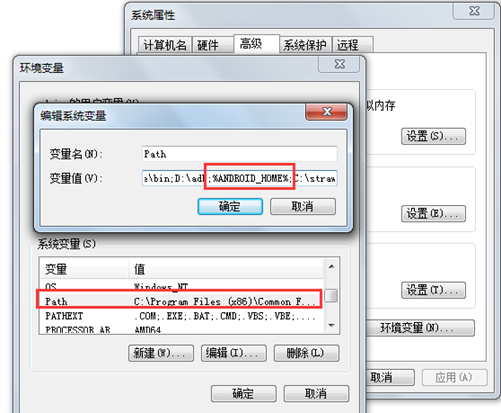
### 2）sdk环境配置

    sdk安装成功后，计算机→属性→高级系统设置→高级→环境变量，在系统变量中，新建ANDROID\_HOME变量，变量值填写sdk中tools和platform-tools的安装目录。



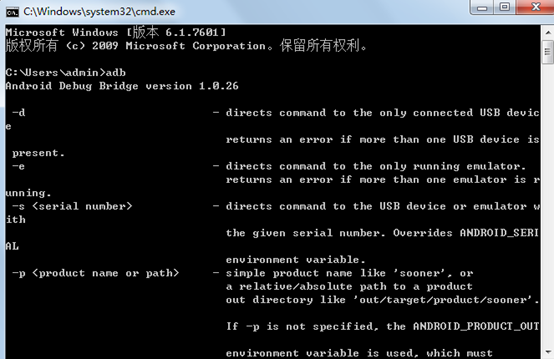
    在系统变量中，编辑Path变量，在变量值最后输入%ANDROID\_HOME%;

（注意原来Path的变量值末尾有没有;号，如果没有，先输入；号再输入上面的代码）



### 3）sdk环境检查

    检验sdk环境是否配置成功，则运行cmd，在cmd窗口中，输入adb。若如图所示，则说明安装和配置成功。



## 3.Python编辑器安装与配置

    python用于支持Monkeyrunner运行，使用python脚本编写用例会大大简化Monkeyrunner用例的编写，且会帮助扩展monkeyrunner的自动化功能。

### 1）Python下载地址

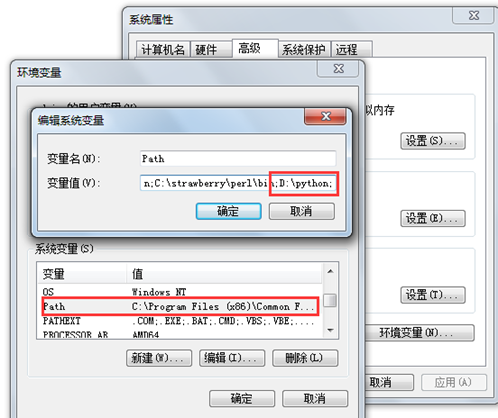
下载地址：[http://www.python.org/download/](http://www.python.org/download/" \t "http://www.cnblogs.com/lynn-li/p/_blank)

下载后，按照提示信息，下一步安装即可。

### 2）Python环境配置

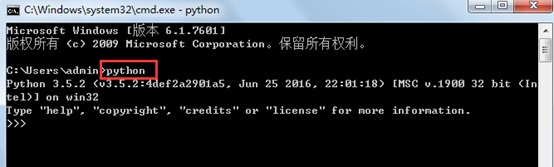
    Python安装成功后，计算机→属性→高级系统设置→高级→环境变量，在系统变量中，编辑Path变量，在变量值最后输入Python的安装路径；

（注意原来Path的变量值末尾有没有;号，如果没有，先输入；号再输入上面的代码）



### 3）Python环境检查

    检验Python环境是否配置成功，则运行cmd，在cmd窗口中，输入python。若如图所示，显示版本信息，则说明安装和配置成功。



## 4.Monkeyrunner环境检查

    若以上步骤均完成，且各环境变量也配置正确，至此，Monkeyrunner环境已经搭建完成。检验Monkeyrunner环境是否搭建成功，则同样运行cmd，在cmd窗口中，输入monkeyrunner。如下图所示，则说明Monkeyrunner环境搭建成功。



    下面就可以用Monkeyrunner连接模拟器来进行自动化的测试了。

# 三、Monkeyrunner使用方法

    Moneyrunner在使用前，必须先打开模拟器或连接上手机设备。下面是Monkeyrunner的实例操作。

## 1.模拟器启动

    我们这里选择命令打开模拟器。运行cmd，在cmd窗口，输入命令：emulator -avd AVD\_test，其中AVD\_test是模拟器的名称，填写自己创建的模拟器名称。



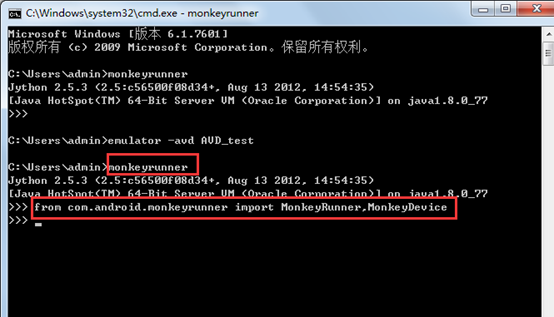
    模拟器启动成功后，我们仍在cmd环境中操作。现在进入Monkeyrunner的shell命令交互模式。

输入命令：monkeyrunner

进入shell命令交互模式后，首要一件事就是导入monkeyrunner所要使用的模块。直接在shell命令下输入命令：

from com.android.monkeyrunner import MonkeyRunner,MonkeyDevice

再回车，这步完成我们就可以利用monkeyrunner进行测试工作了。

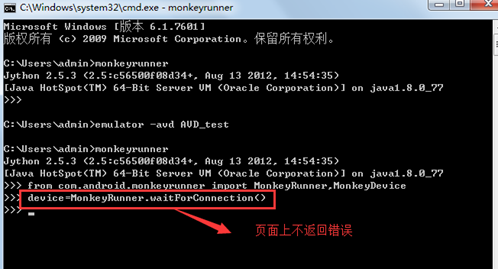


## 2.模拟器连接

    下面我们就要Monkeyrunner连接上模拟器，进行一系列操作了。输入命令：

device=MonkeyRunner.waitForConnection()

其中，device=MonkeyRunner.waitForConnection(6,’emulator-5554′)  
参数1：超时时间，单位秒，浮点数，默认是无限期地等待。  
参数2：指定的设备名称device\_id，默认为当前设备（手机优先，其次为模拟器）



    输入命令后，页面上没有错误信息返回，即成功连接设备。

## 3.app安装并启动

### 1）app安装

    模拟器启动成功后，我们安装自己想要的apk，这里我们选择qq音乐安装。

输入命令：device.installPackage('F:\\QQyinle\_439.apk')，其中，参数是APK的相对路径。

    安装成功返回true，此时查看模拟器我们可以在IDLE界面上看到安装的APK的图标了。



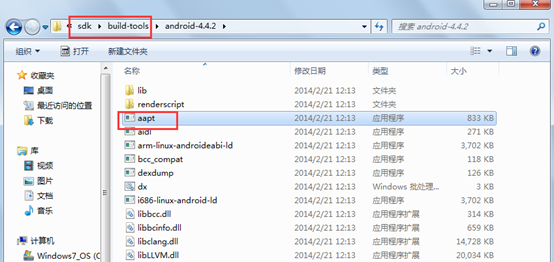
### 2）app启动

    app安装成功后，现在启动该app，命令为：

device.startActivity(component="package名/.activity")

首先，我们有必要说一下，**如何获取一个app的package名和activity**。这里，我们只描述一种获取方式。

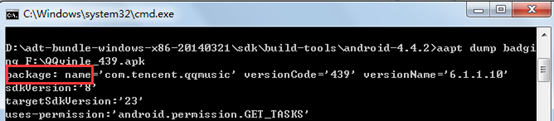
使用aapt，其中aapt是sdk自带的一个工具，在sdk\builds-tools\目录下：

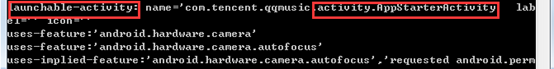


    以存储在F盘的qq音乐为例，运行cmd，命令行中切换到aapt.exe目录，

**方法一：**

执行命令：aapt dump badging F:\QQyinle\_439.apk ，注意，apk路径中一定不能有空格。





由上图可知：package name：com.tencent.qqmusic

activity：.activity.AppStarterActivity

方法一由于日志较多，寻找起来比较费劲，所以我们引出方法二。

**方法二（推荐）：**

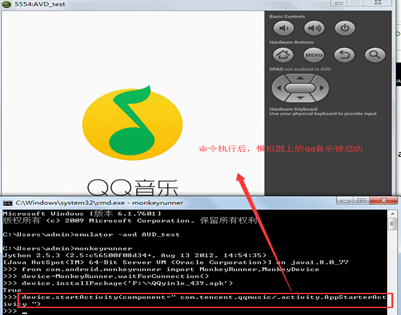
把日志存储在特定的文件中，在文件中通过搜索关键字，得到包名及活动名，这里我把结果输出到F盘的log.txt中：

aapt dump badging F:\QQyinle\_439.apk > F:\log.txt

    到此，已经获取了app的package名和activity。下面，我们真正的启动app。在原有cmd运行窗口，输入命令：

device.startActivity(component=" com.tencent.qqmusic/.activity.AppStarterActivity ")

    命令执行后，模拟器上的app被启动。这表示命令启动app成功。这里的关键是app的package name和activity对应获取正确，否则启动不了特定app。



    此时可以向模拟器发送如按键、滚动、截图、存储等操作了。

# 四、Monkeyrunner运行python脚本

    同样，Monkeyrunner可以直接调用指定python脚本，将命令写到python文件里，命名例如\*\*\*.py，然后我们再从命令行直接通过monkeyrunner运行它即可。比如，我们还是用上面的例子，语法如下：monkeyrunner \*\*\*.py。接下来monkeyrunner会自动调用\*\*\*.py，并执行其中的语句，相当方便。

    我们这里将上述例子，所有命令放在python文件里，并命名test.py，然后存储到本地F盘，即路径为：F:\test.py。

[IMG_274](http://www.cnblogs.com/lynn-li/p/javascript:void(0);)

#coding:utf-8from com.android.monkeyrunner import MonkeyRunner,MonkeyDevice

device=MonkeyRunner.waitForConnection()

device.installPackage('F:\\QQ\_374.apk')

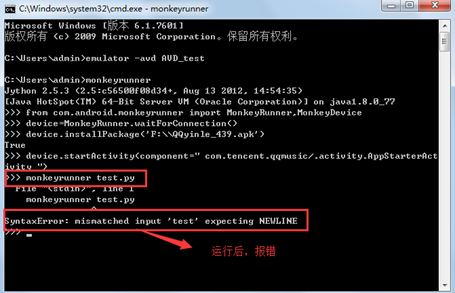
MonkeyRunner.sleep(3.0)

runComponent = "com.tencent.qqmusic/.activity.AppStarterActivity"

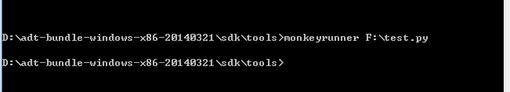
device.startActivity(component=runComponent)

[IMG_275](http://www.cnblogs.com/lynn-li/p/javascript:void(0);)

    在cmd中运行monkeyrunner F:\test.py，这里的python脚本路径为相对路径。结果报错：**SyntaxError：mismatched input 'test' expecting NEWLINE**，如下：



    这是因为python脚本应在dos模式下执行，不要进入monkeyrunner的shell命令交互模式。正确的方式如下，输入命令monkeyrunner F:\test.py：



    运行成功后，则可以在模拟器上看到启动的qq音乐app。