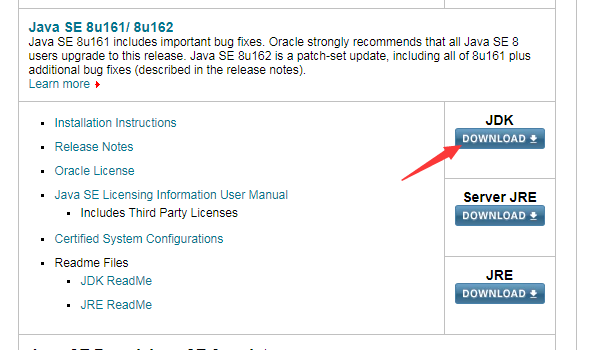
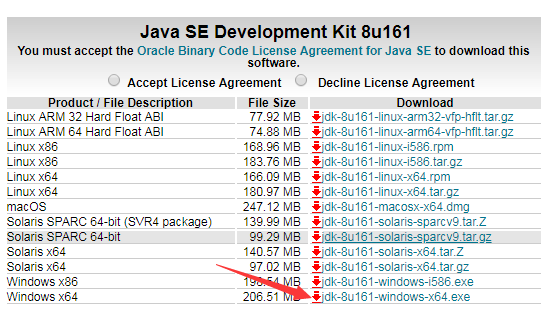
Android开发环境搭建

第一步：安装JDK

官网：http://www.oracle.com/technetwork/Java/javase/downloads/index.html

点击图中下载按钮

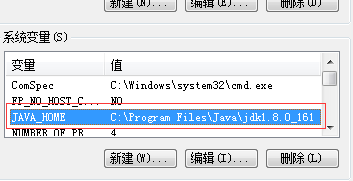


选择图中版本下载

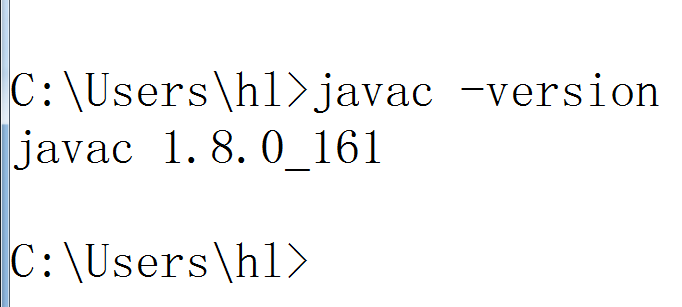
自定义路径安装，不要保存在中文路径下；

配置环境变量：计算机——属性——高级系统设置——环境变量

1. 新建一个系统环境变量，变量名为JAVA\_HOME，变量值为JDK的安装路径，如下图：



1. 在系统变量中找到Path，将;%JAVA\_HOME%\bin;添加到变量值后面
2. 检查是否配置成功。打开命令行窗口，输入javac –version，如成功则如下图所示：



第二步：安装SDK

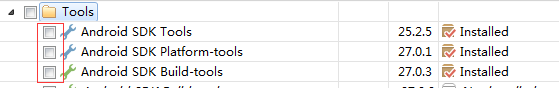
官网：<http://www.androiddevtools.cn/>

往下滚动，找到SDK Tools，下载下图中所示压缩包，无需安装，只需解压到一个目标路径，不要是中文。

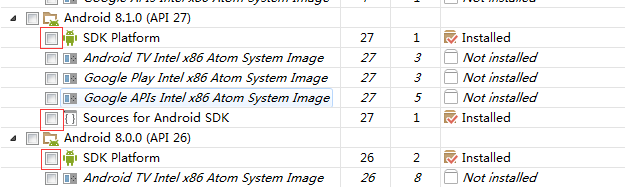


然后双击“SDK Manager”，启动SDK Manager

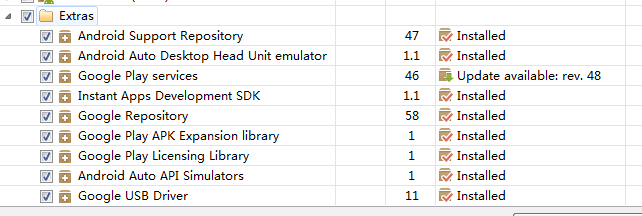
1. Tools勾选图中三个



1. 往下滚动，API勾选下图所示：

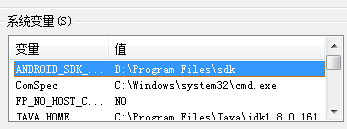


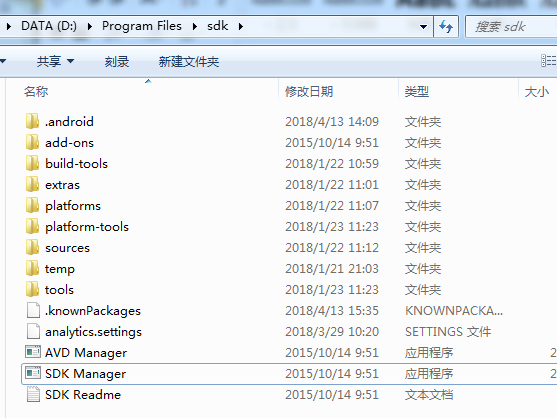
1. 滚动到最后，Extras全选，然后点击 install



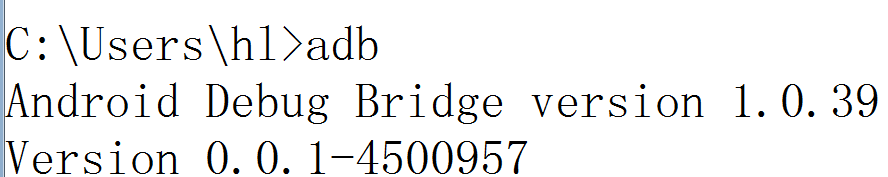
然后配置环境变量：

1. 新建一个系统环境变量，变量名为ANDROID\_SDK\_HOME，变量值为SDK的存放路径，（我将SDK解压到了D盘）如下图：





1. 把%ANDROID\_SDK\_HOME%\platform-tools;%ANDROID\_SDK\_HOME%\tools添加到Path系统环境变量中。
2. 检查是否配置成功，在命令行输入 adb，成功则出现下图：



4）利用ionic打包还需安装gradle，无需安装，只需解压到一个目标路径下，然后配置环境变量，直接将路径添加到Path系统环境变量后面，如：

D:\Program Files\gradle-4.5\bin

第三步：测试

测试最终是否成功的话，进入ionic项目，在命令行执行

ionic cordova platform add android

然后执行

ionic cordova build android

如能在项目中的platforms\android\build\outputs\apk中出现一个android-debug.apk安装包，说明全部安装成功

Ionic打包

1. 命令：ionic cordova build android 生成一个测试版本，不能用于发布
2. 命令：ionic cordova build android --release 生成一个未签名发布版本，编译成功后在项目的platforms/android/build/outputs/apk里面生成默认名字为android-release-unsigned.apk的release版本apk文件
3. 命令：ionic cordova build android --prod --release 代码压缩优化，体积更小
4. 命令：ionic cordova run android -l 用于手机真机测试，代码修改保存，手机自动编译更新

签名

**1、生成keystore文件**（签名时需要用到这个文件）

keytool是JDK自带的加密工具，只需生成一次，然后保存在本地，以后签名直接使用这个文件，不需要重新生成。

**命令**：keytool -genkey -v -keystore my-release-key.keystore -keyalg RSA -keysize 2048 -validity 10000 -alias alias\_name

执行命令后，系统会让我们设置一些密码以及安全问题，根据提示进行设置便好，最后生成默认名为my-release-key.keystore文件，该文件存储在当前命令行执行的目录下。

其中alias\_name,是自己设置的别名，一定要记住，后面会一直用到。

validity 10000表示文件的加密时间为10000天

**2、使用jarsigner签名**

jarsigner是JDK自带的签名工具，如果要将apk发布到android应用市场，就需要对release版本的apk文件使用keystore文件进行签名

**命令**：jarsigner -verbose -sigalg SHA1withRSA -digestalg SHA1 -keystore my-release-key.keystore android-release-unsigned.apk alias\_name

注:android-release-unsigned.apk为你所需要打包的apk的具体路径， alias\_name为上面生成的keystore文件时的别名。