

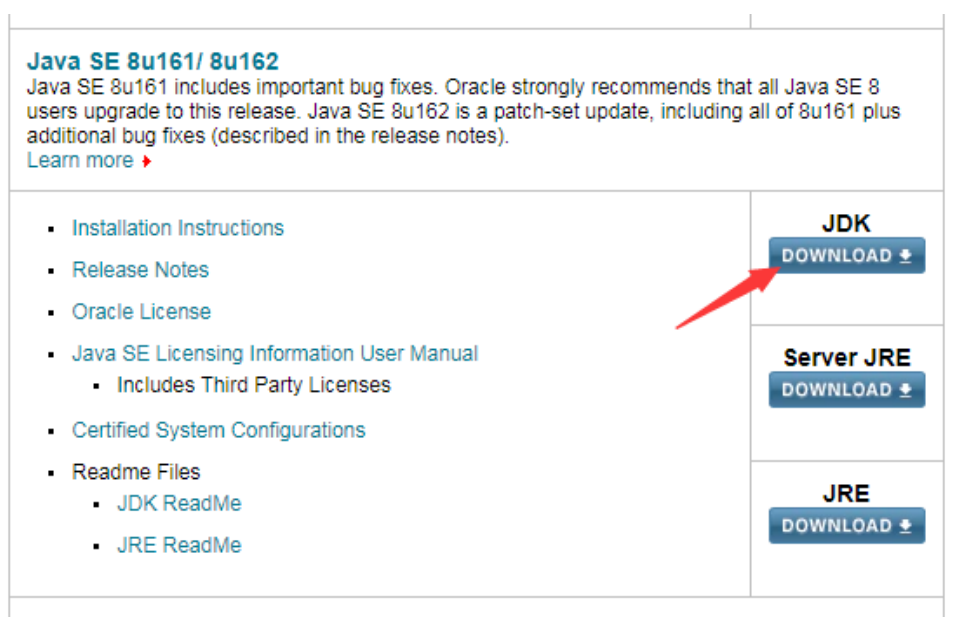
Android 开发环境搭建

第一步：安装 JDK

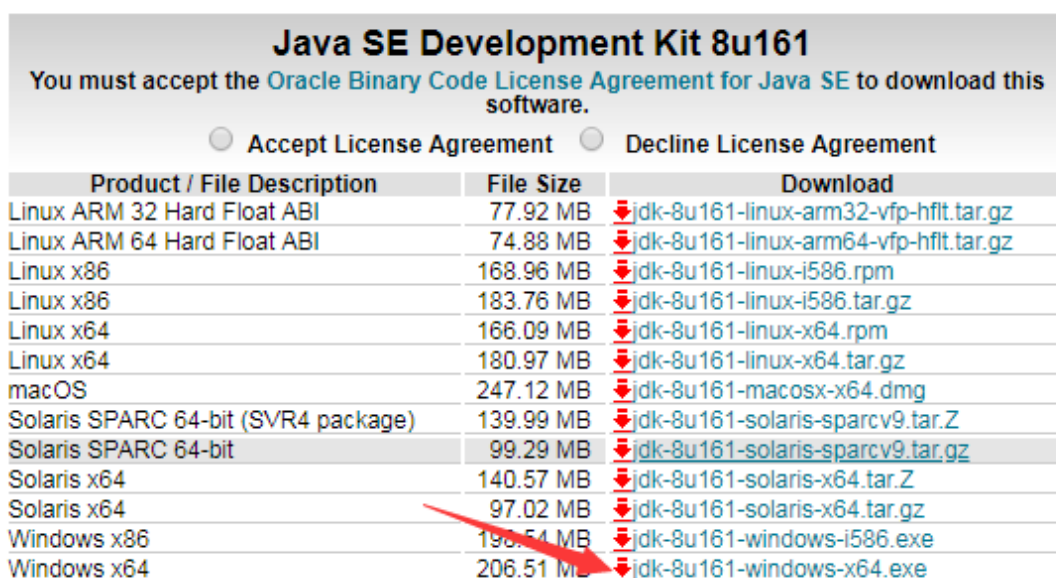
官网：

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

点击图中下载按钮



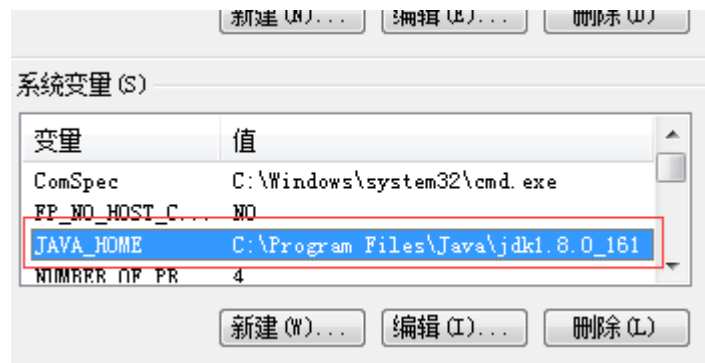
选择图中版本下载



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：



- 2) 在系统变量中找到 `Path`，将 `;%JAVA_HOME%\bin;` 添加到变量值后面
- 3) 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：

```
C:\Users\hl>javac -version
javac 1.8.0_161

C:\Users\hl>
```

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。

SDK Tools				
版本	平台	下载	大小	SHA-1
3859397	Windows	sdk-tools-windows-3859397.zip	132 MB	7f6037d3a7d6789b4fdc06ee7af041e
	Mac OS X	sdk-tools-darwin-3859397.zip	82 MB	4a81754a760fce88cba74d69c364b0f
	Linux	sdk-tools-linux-3859397.zip	130 MB	444e22ce8ca0f67353bda4b85175ed
24.4.1	Windows	installer_r24.4.1-windows.exe	144 MB	f9b59d72413649d31e6
		android-sdk_r24.4.1-windows.zip	190 MB	66b6a6433053c152b2
	Mac OS X	android-sdk_r24.4.1-macosx.zip	98 MB	85a9cccb0b1f9e6f1f6
	Linux	android-sdk_r24.4.1-linux.tgz	311 MB	725bb360f0f7d04eacc

然后双击“SDK Manager”，启动 SDK Manager

1) Tools 勾选图中三个

Tools			
<input checked="" type="checkbox"/>	Android SDK Tools	25.2.5	Installed
<input checked="" type="checkbox"/>	Android SDK Platform-tools	27.0.1	Installed
<input checked="" type="checkbox"/>	Android SDK Build-tools	27.0.3	Installed

2) 往下滚动，API 勾选下图所示：

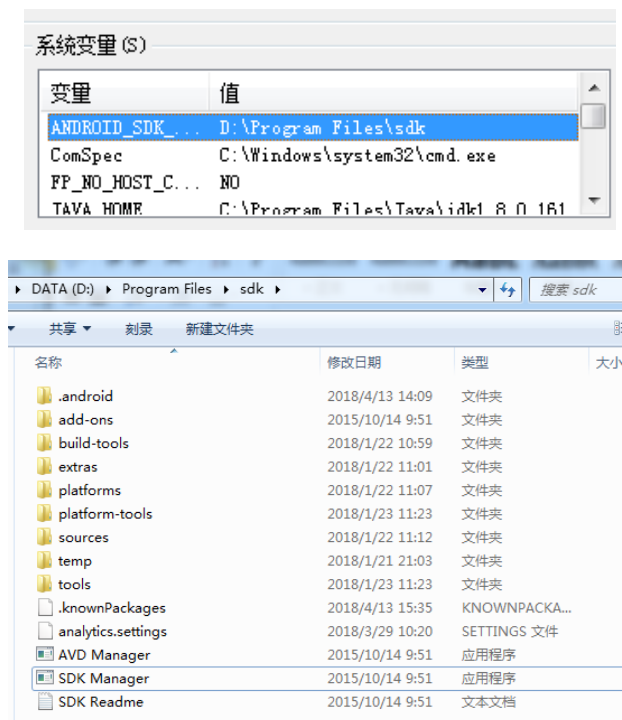
Google APIs Intel x86 Atom System Image			
Android 8.1.0 (API 27)			
<input checked="" type="checkbox"/>	SDK Platform	27	1 Installed
<input type="checkbox"/>	Android TV Intel x86 Atom System Image	27	3 Not installed
<input type="checkbox"/>	Google Play Intel x86 Atom System Image	27	3 Not installed
<input type="checkbox"/>	Google APIs Intel x86 Atom System Image	27	5 Not installed
<input checked="" type="checkbox"/>	Sources for Android SDK	27	1 Installed
Android 8.0.0 (API 26)			
<input checked="" type="checkbox"/>	SDK Platform	26	2 Installed
<input type="checkbox"/>	Android TV Intel x86 Atom System Image	26	8 Not installed

3) 滚动到最后，Extras 全选，然后点击 install

Extras			
<input checked="" type="checkbox"/>	Android Support Repository	47	Installed
<input checked="" type="checkbox"/>	Android Auto Desktop Head Unit emulator	1.1	Installed
<input checked="" type="checkbox"/>	Google Play services	46	Update available: rev. 48
<input checked="" type="checkbox"/>	Instant Apps Development SDK	1.1	Installed
<input checked="" type="checkbox"/>	Google Repository	58	Installed
<input checked="" type="checkbox"/>	Google Play APK Expansion library	1	Installed
<input checked="" type="checkbox"/>	Google Play Licensing Library	1	Installed
<input checked="" type="checkbox"/>	Android Auto API Simulators	1	Installed
<input checked="" type="checkbox"/>	Google USB Driver	11	Installed

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：



- 2) 把%ANDROID_SDK_HOME%\platform-tools;%ANDROID_SDK_HOME%\tools 添加到 Path 系统环境变量中。

- 3) 检查是否配置成功，在命令行输入 adb，成功则出现下图：

```
C:\Users\hl>adb
Android Debug Bridge version 1.0.39
Version 0.0.1-4500957
```

- 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。