

PhoneGap3.4 android 环境安装教程

学习要点:

1. 安装前准备工作
2. 安装 node.js
3. 安装 phonegap3.4 运行 helloworld
4. 安装插件
5. 配置启动项等

主讲教师: (树根)

合作网站: www.phonegap100.com (PhoneGap 中文网)
www.apiying.com (微信营销后台管理系统)

phonegap2.9 安装视频教程地址

<http://bbs.phonegap100.com/thread-174-1-1.html>

一、安装前准备工作




1、下载 java JDK 1.6 以上版本

<http://www.oracle.com/technetwork/java/javase/downloads/jdk7-downloads-1880260.html>

2、下载 Android Development Tools

已经包括了 Android sdk、eclipse, 解压即可使用。

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

 adt-bundle-windows-x86_64-20131030.zip	2014/3/13 17:33	360压缩 ZIP 文件	491,930 KB
 jdk-7u17-windows-x64.exe	2013/4/21 17:35	应用程序	92,592 KB
 node-v0.10.26-x64.msi	2014/3/5 16:40	Windows Install...	5,752 KB

2、设置系统变量

Java 环境变量配置完成以后（也就是 java javac 运行成功以后）还需要额外配置的东西如下

系统变量 Path 后面添加：

- D:\Program Files\ADT\sdk\tools;
- D:\Program Files\ADT\sdk\platform-tools;
- D:\Program Files\ADT\sdk\eclipse\plugins\org.apache.ant_1.8.3.v201301120609\bin

4、更新自己的 android sdk 到 4.4 以上版本（不然会出错）

5、注意 ANT 目录不能是中文 不然编译不成功

6、分别输入 java javac ant 看是否配置环境成功

运行 java 成功

```
C:\Users\<redacted>>java
用法: java [-options] class [args...]
      <执行类>
或 java [-options] -jar jarfile [args...]
      <执行 jar 文件>
其中选项包括:
-d32          使用 32 位数据模型 <如果可用>
-d64          使用 64 位数据模型 <如果可用>
-server      选择 "server" UM
-hotspot     是 "server" UM 的同义词 [已过时]
              默认 UM 是 server.

-cp <目录和 zip/jar 文件的类搜索路径>
-classpath <目录和 zip/jar 文件的类搜索路径>
              用 ; 分隔的目录, JAR 档案
              和 ZIP 档案列表, 用于搜索类文件。
-D<name>=<value>
              设置系统属性
-verbose[:class[:gc[:jni]]
              启用详细输出
```

运行 javac 成功

```
C:\Users\...>javac
用法: javac <options> <source files>
其中, 可能的选项包括:
-g 生成所有调试信息
-g:none 不生成任何调试信息
-g:<lines,vars,source> 只生成某些调试信息
-nowarn 不生成任何警告
-verbose 输出有关编译器正在执行的操作的消息
-deprecation 输出使用已过时的 API 的源位置
-classpath <路径> 指定查找用户类文件和注释处理程序的位置
-cp <路径> 指定查找用户类文件和注释处理程序的位置
-sourcepath <路径> 指定查找输入源文件的位置
-bootclasspath <路径> 覆盖引导类文件的位置
-extdirs <目录> 覆盖所安装扩展的位置
-endorseddirs <目录> 覆盖签名的标准路径的位置
-proc:<none,only> 控制是否执行注释处理和/或编译。
-processor <class1>[,<class2>,<class3>... ] 要运行的注释处理程序的名
-processorpath <路径> 指定查找注释处理程序的位置
-d <目录> 指定放置生成的类文件的位置
-s <目录> 指定放置生成的源文件的位置
-implicit:<none,class> 指定是否为隐式引用文件生成类文件
-encoding <编码> 指定源文件使用的字符编码
-source <发行版> 提供与指定发行版的源兼容性
-target <发行版> 生成特定 VM 版本的类文件
-version 版本信息
-help 输出标准选项的提要
-A关键字 [=值] 传递给注释处理程序的选项
-X 输出非标准选项的提要
-X<标记> 直接将 <标记> 传递给运行时系统
```

ant 成功

```
C:\Users\...>ant
Buildfile: build.xml does not exist!
Build failed
```

二、安装 node.js

下载地址:
<http://nodejs.org/>

下载完成安装 node.js

三、安装 phonegap3.4

打开命令窗口 运行

npm install -g cordova

```
C:\Users\50...>npm install -g cordova
npm http GET https://registry.npmjs.org/cordova
npm http 200 https://registry.npmjs.org/cordova
npm http GET https://registry.npmjs.org/plugman/0.20.2
npm http GET https://registry.npmjs.org/elementtree/0.1.5
npm http GET https://registry.npmjs.org/plist-with-patches
npm http GET https://registry.npmjs.org/mime
npm http GET https://registry.npmjs.org/optimist/0.6.0
npm http GET https://registry.npmjs.org/underscore/1.4.4
npm http GET https://registry.npmjs.org/request
npm http GET https://registry.npmjs.org/q
npm http GET https://registry.npmjs.org/shelljs
npm http GET https://registry.npmjs.org/npmconf
npm http GET https://registry.npmjs.org/tar
npm http GET https://registry.npmjs.org/xcode/0.6.6
npm http GET https://registry.npmjs.org/semver
npm http 304 https://registry.npmjs.org/elementtree/0.1.5
npm http 304 https://registry.npmjs.org/plugman/0.20.2
npm http 304 https://registry.npmjs.org/plist-with-patches
npm http 304 https://registry.npmjs.org/optimist/0.6.0
npm http 304 https://registry.npmjs.org/underscore/1.4.4
npm http 200 https://registry.npmjs.org/mime
npm http 304 https://registry.npmjs.org/tar
npm http 304 https://registry.npmjs.org/npmconf
npm http 200 https://registry.npmjs.org/shelljs
npm http 304 https://registry.npmjs.org/semver
npm http 200 https://registry.npmjs.org/request
npm http 304 https://registry.npmjs.org/xcode/0.6.6
npm http 200 https://registry.npmjs.org/q
```

```
cordova create hello com.example.hello "HelloWorld"
cd hello
cordova platform add android
cordova build
```

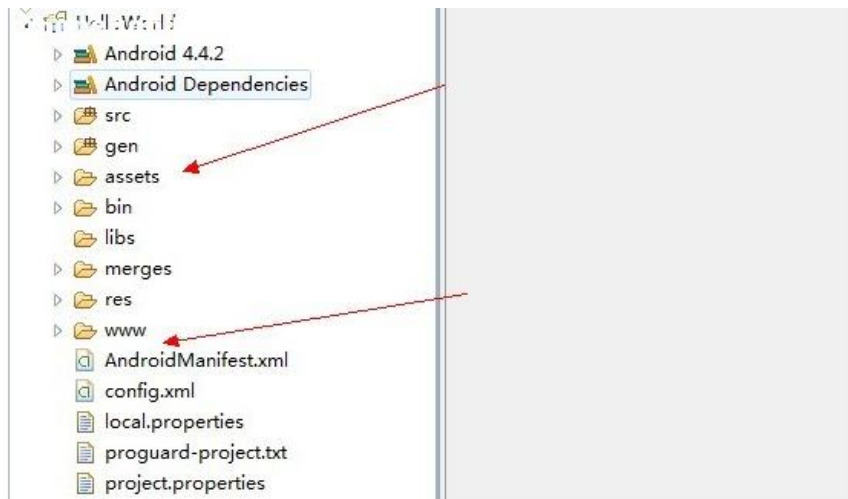
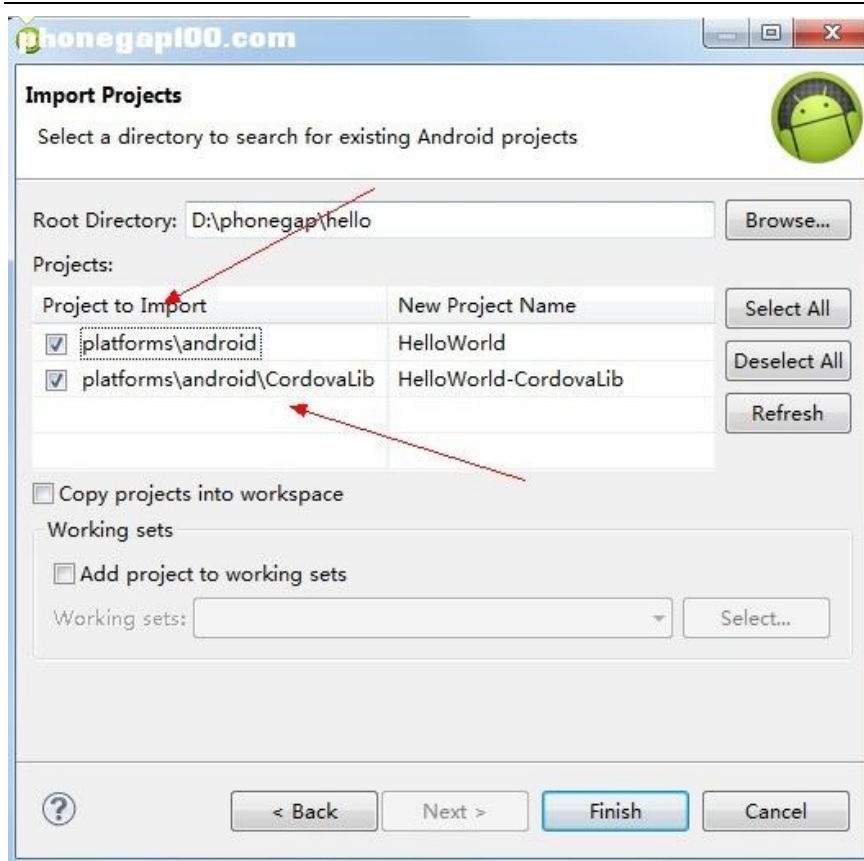
cordova run android 运行模拟器

使用 **eclipse** 工具打开并编辑该项目

File>New>Project>Android>Android Project from Existing Code

选择则刚才建立项目所在路径，即可使用 ADT 打开进行编辑。





四、安装 删除 插件

<code>cordova plugin add org.apache.cordova.device</code>	安装对应的 device 插件
<code>cordova plugin list</code>	列出已经安装的插件
<code>cordova plugin remove org.apache.cordova.device</code>	删除 device 插件

五、配置启动项

```
<?xml version='1.0' encoding='utf-8'?>
<widget id="com.example.hello" version="0.0.1" xmlns="http://
  <name>HelloWorld</name>
  <description>
    A sample Apache Cordova application that responds to
  </description>
  <author email="dev@cordova.apache.org" href="http://c
    Apache Cordova Team
  </author>
  <content src="apidemo/phonegap_event001.html" />
  <access origin="*" />
</widget>
```

感谢收看本次教程
本教程由 [phonegap 中文网\(phonegap100.com\)](http://phonegap100.com)
apiying.com 提供
我是主讲老师：树根
我的邮箱：phonegap100@qq.com