Android 应用程序开发环境搭建

作者: 深圳百问网科技有限公司 钟尚明

一、写在前面

韦东山 Linux 视频第 3 期:项目视频，里面的第 2 个项目实现了这样一个功能：

摄像头==>ARM 开发板<==>WIFI<---------->手机

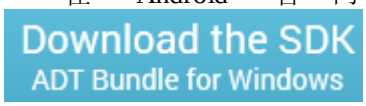
在这个项目里，可以在手机上安装一个 Android 应用程序，用它来观看开发板拍摄到的视频、图片。我们开发了这个 Android 应用程序，它的源码完全开放。

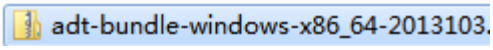
第 3 期视频里不会详细讲解 Android 程序，我们只是简单的介绍环境搭建，所提供的源码里有详细的注释。

开发安卓应用程序，必须具备 JAVA 基础知识，可以看 mars 老师的视频
也可以参考 Android 官网文档，“也许”第 4 期视频开始讲 Android。

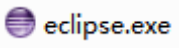
更多信息请关注: www.100ask.net, 100ask.taobao.com

二、下载 SDK

在 Android 官网 http://developer.android.com/intl/zh-cn/sdk/index.html, 单击 

选择相应的系统版本下载，这是 Android 的 sdk 包,里面有 ICE:eclipes, sdk,和 ADT 插件， 我下载的是 

三、打开 eclipes

解压 adt_bundle 后双击 eclipes 目录下的 

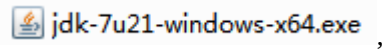
，结果不能打开，会提示未装 jdk。

我们需要安装 JRE

1. Windows XP：

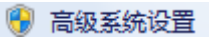
到 http://www.veryhuo.com/down/html/36178.html 下载安装就可，如果是 XP 系统就不用配置环境变量了

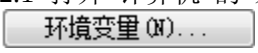
2. Windows 7：

用我们提供的 jdk: 

,当然也可到 eclipes 的官网下载 jdk，方法很简单，就不赘述。

安装好 jre 后还要配置 jre 环境变量，配置环境变量的资料网上 N 多，为方便我这里也总结下：

2.1 打开”计算机”的”系统属性”，点击左边导航项的 

，找到”高级”选项，找到 

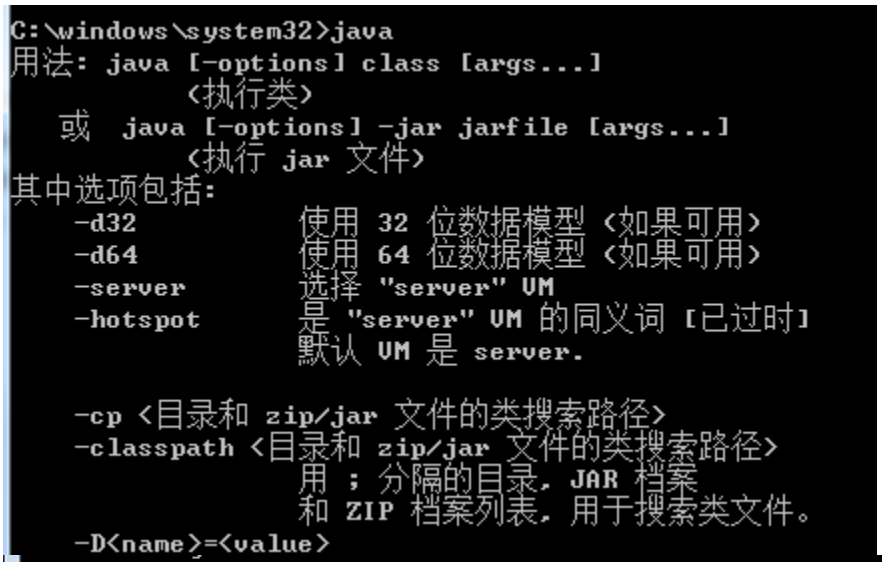
，单击它。

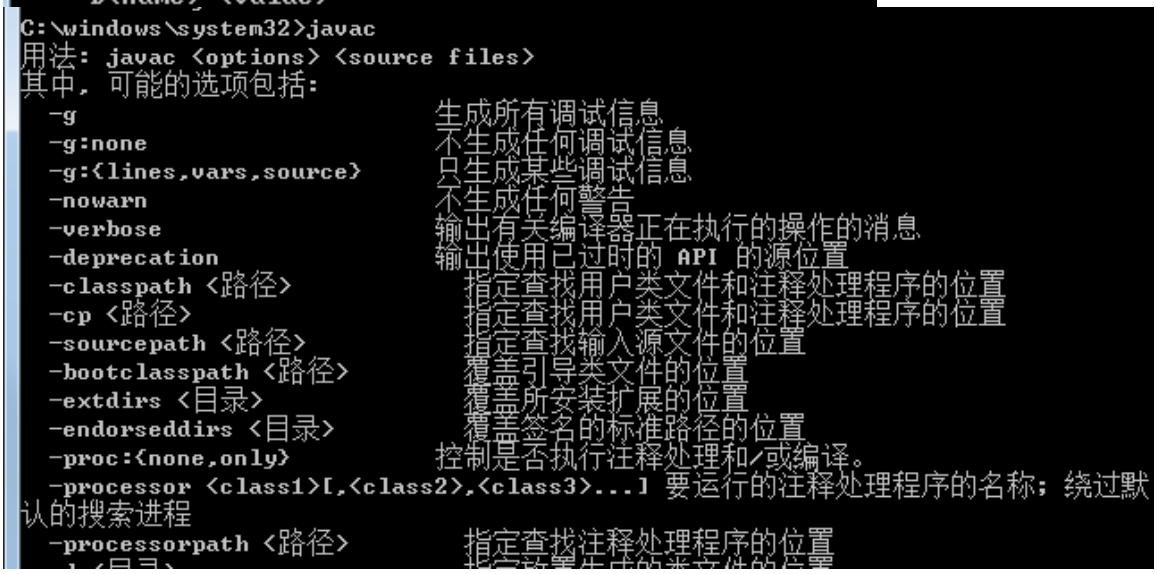
2.2 新建 JAVA_HOME 内容为 C:\Program Files\Java\jdk1.7.0_21 //jre 的安装路径，我用默认安装路径

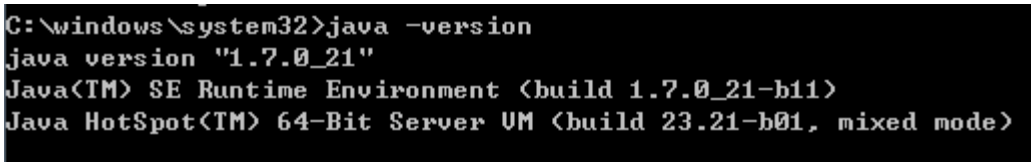
2.3 新建 CLASSPATH
内容是.;%JAVA_HOME%\lib;%JAVA_HOME%\lib\dt.jar;%JAVA_HOME%\lib\tools.jar，最前面的逗号不能少

2.4 编辑 Path //如果没有，则新建 ，内容为
%JAVA_HOME%\bin;%JAVA_HOME%\jre\bin;%SystemRoot%\system32;%SystemRoot%;%SystemRoot%\system32\wbem

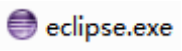
最后进行验证，在 cmd 依次执行 java，javac，java –version，出现下图则表示配置环境变量成功。





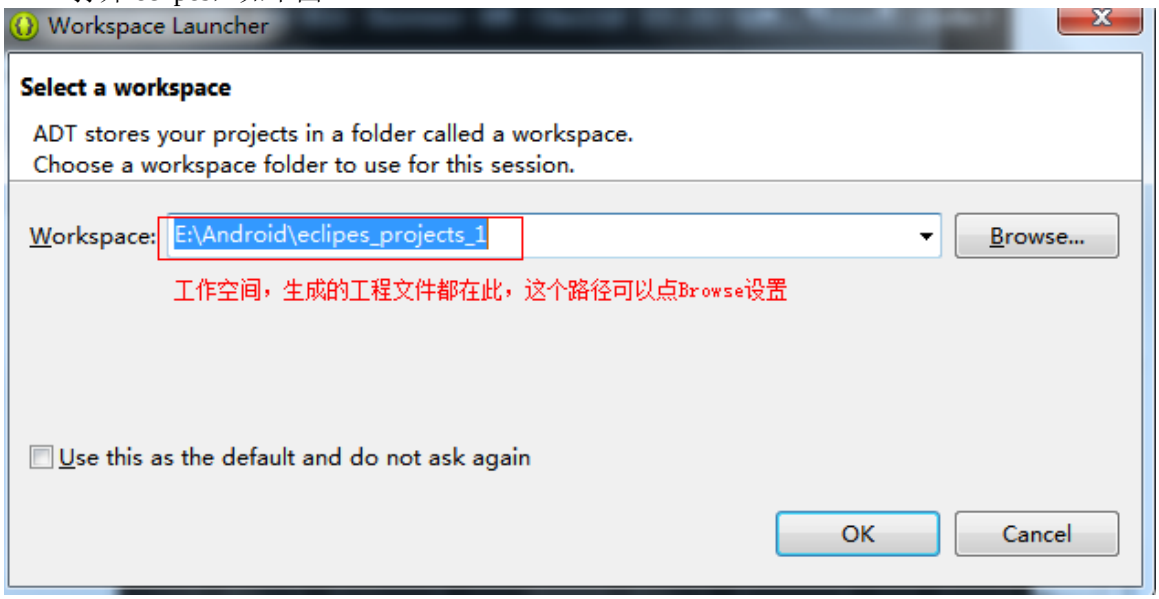


注意：对于 win7 系统，配置好环境变量后，必须重启电脑才生效

安装、配置好 jdk 后，双击 

Say hello to Android

打开 eclipes，如下图

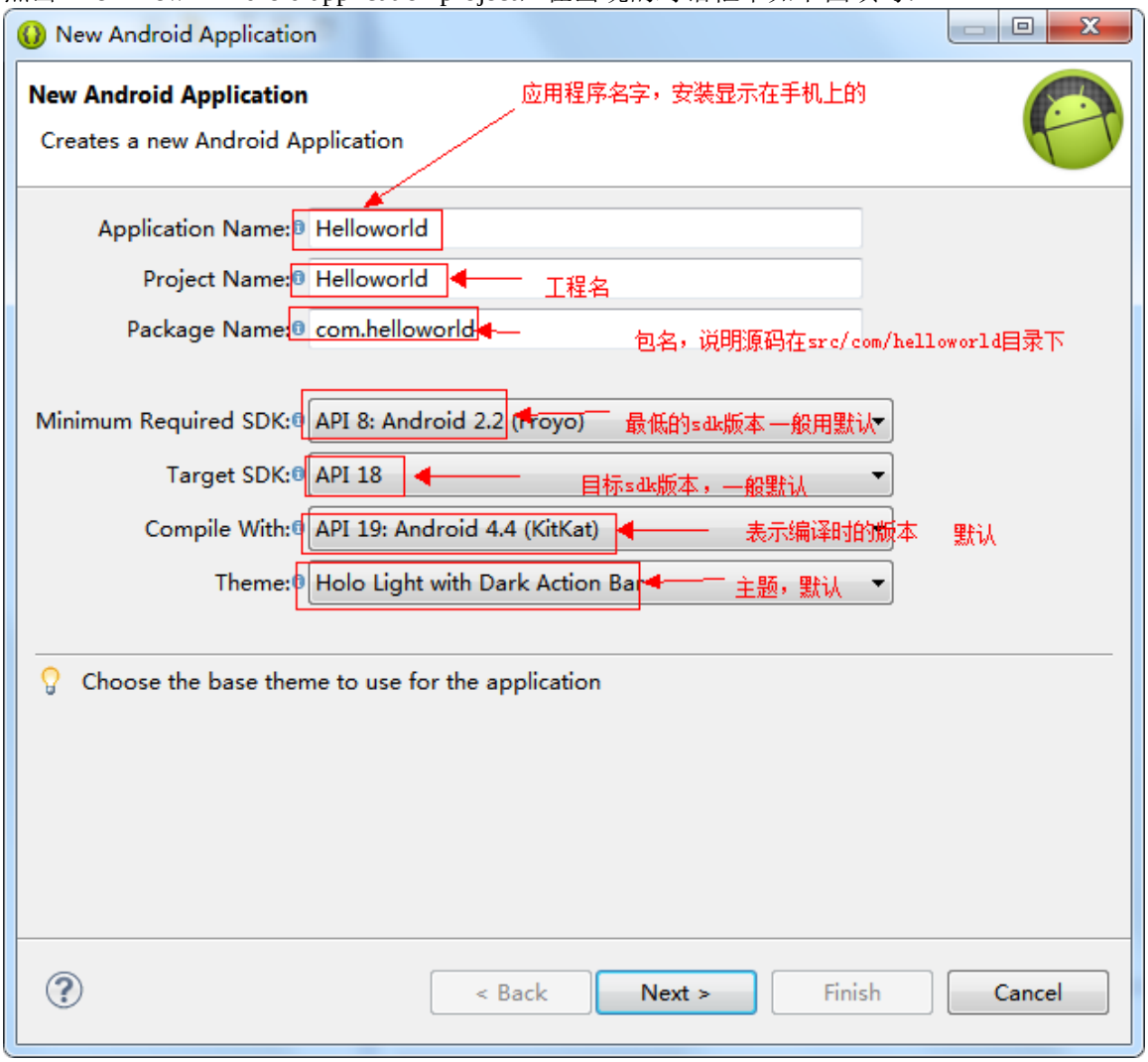


点 OK，工具安装完毕。

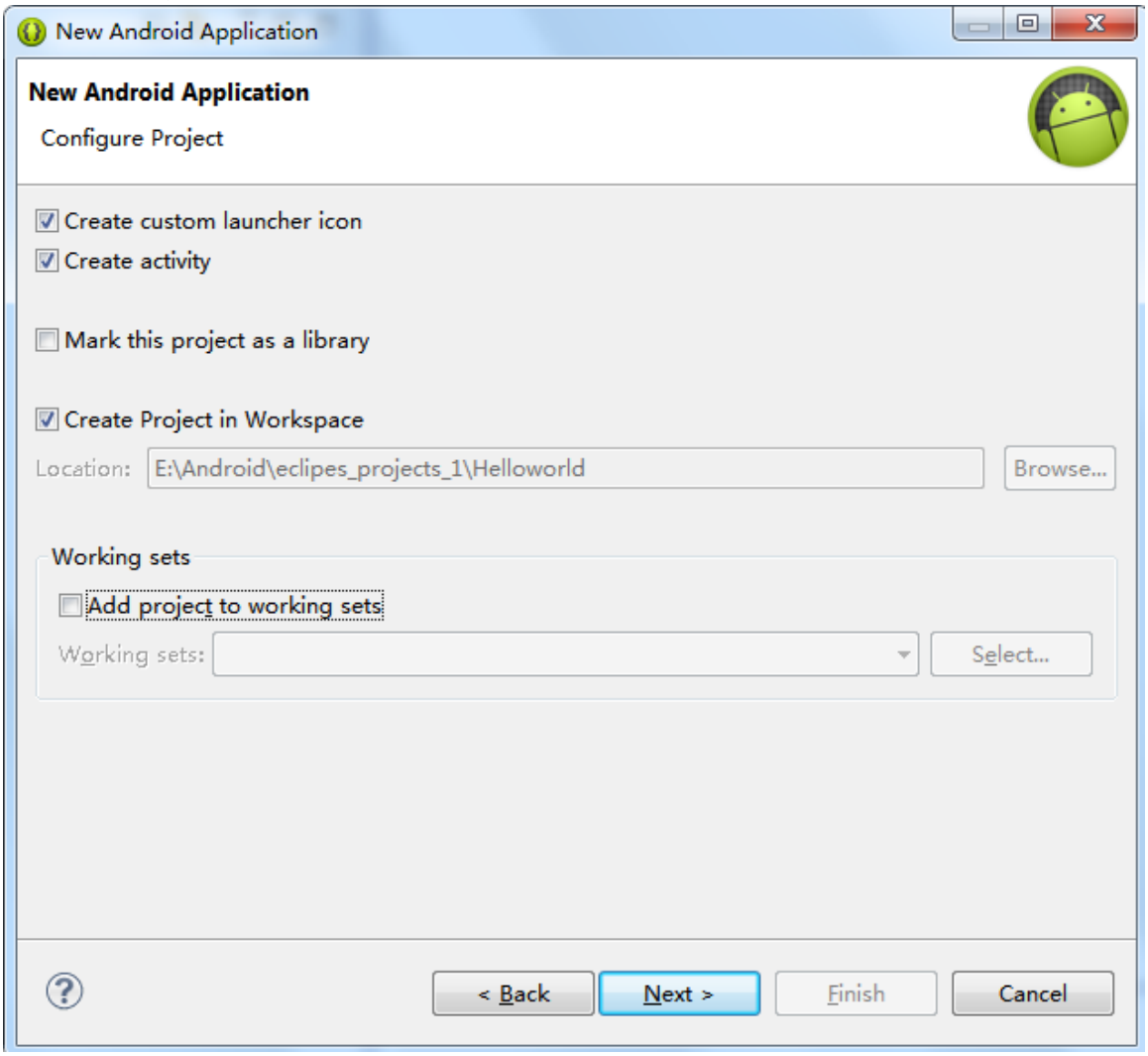
四、新建一个简单工程：helloworld

1. 新建工程：

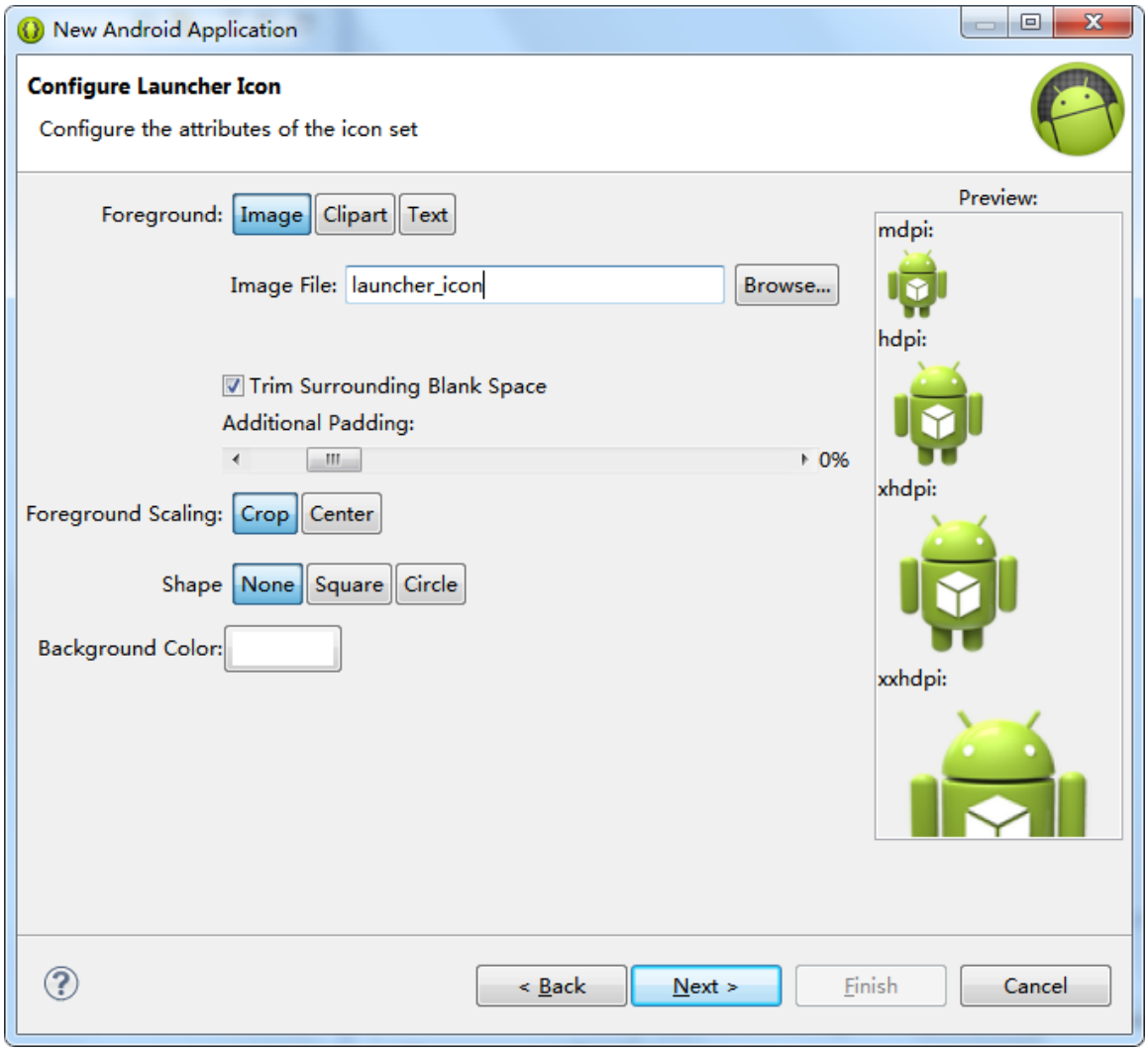
1.1 点击 File->New->Android application project，在出现的对话框中如下图填写：



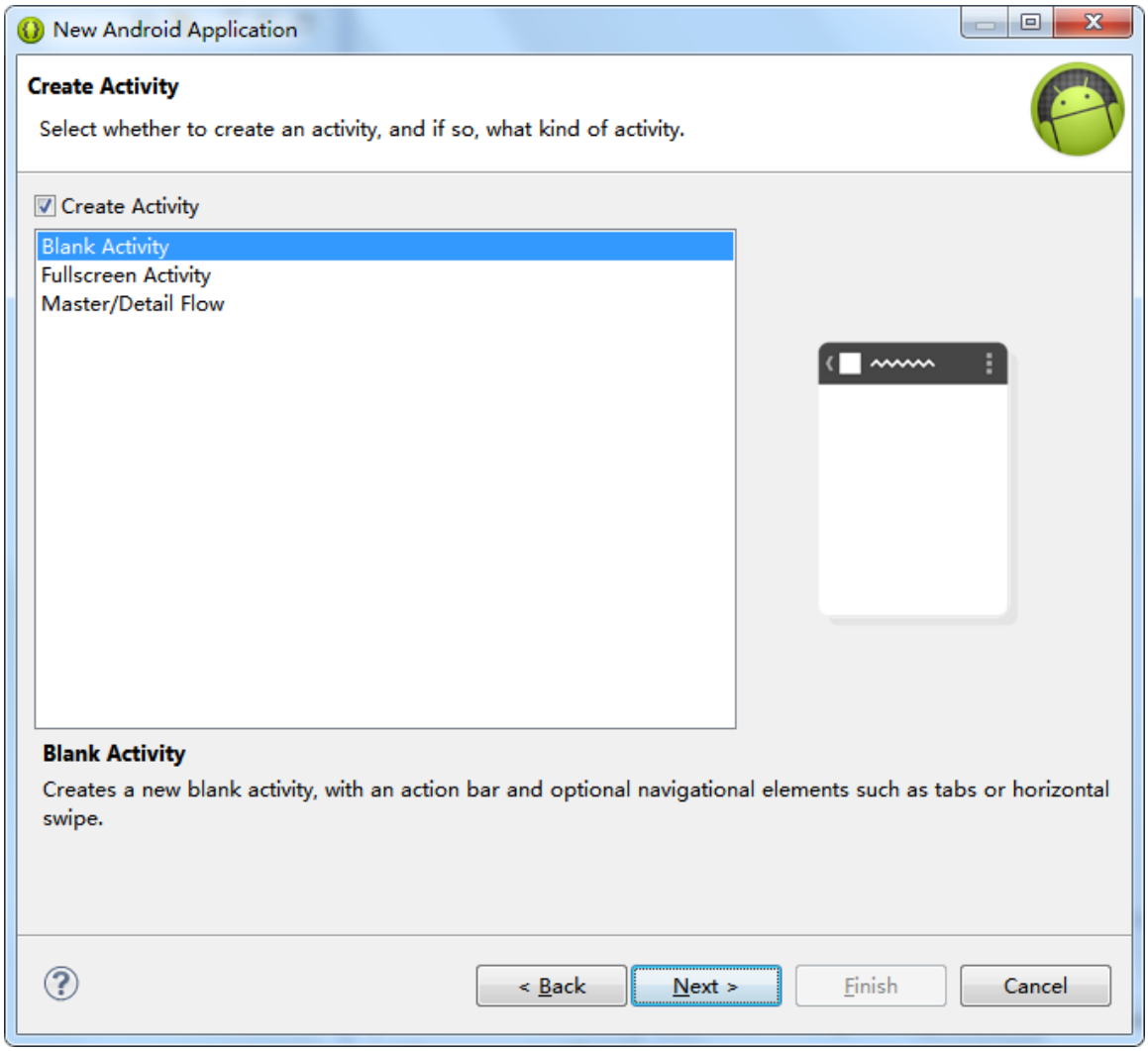
1.2 点“next”出现下图：



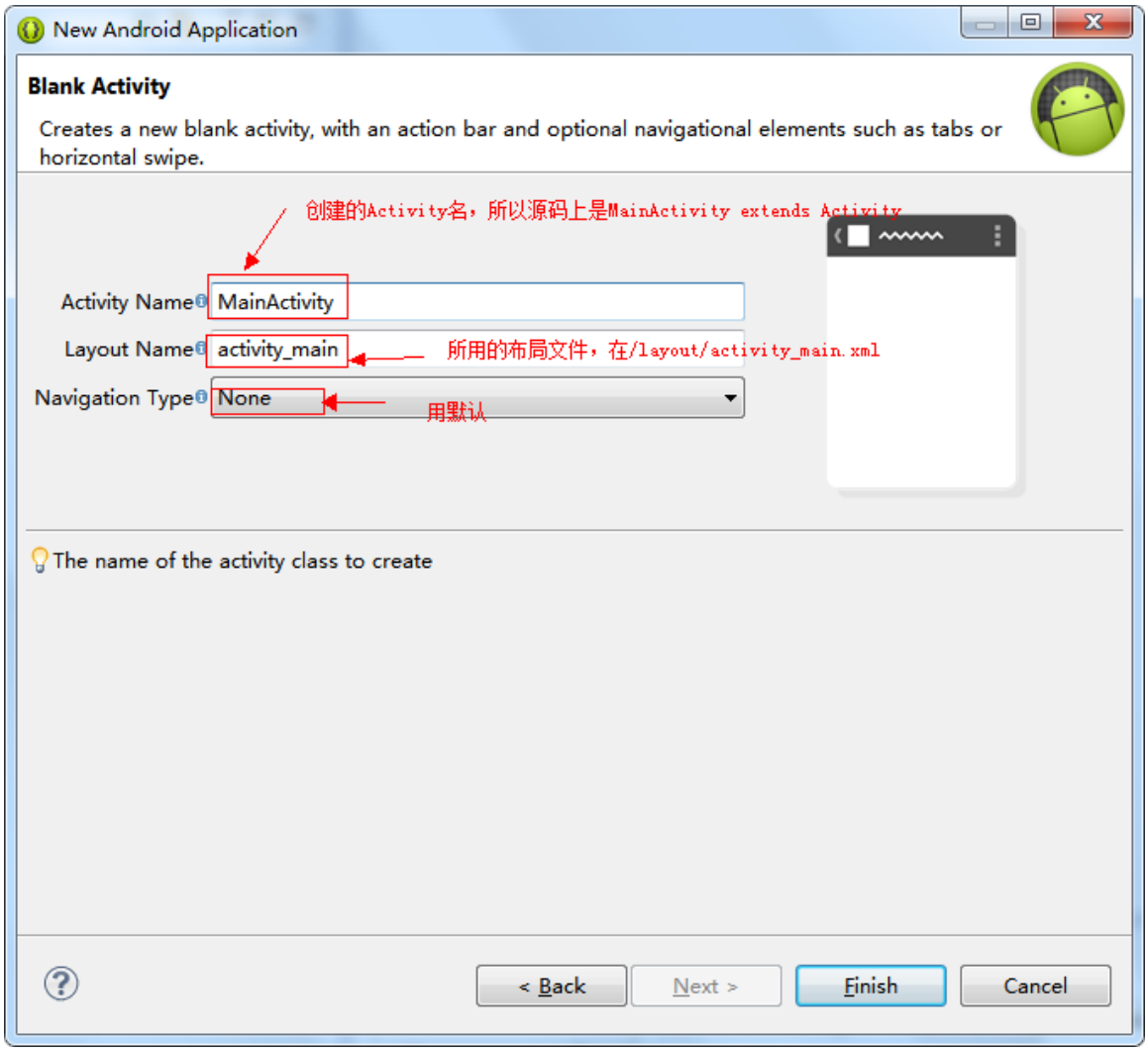
直接点“next”出现下图：



继续点“next”出现：


New Android Application
Create Activity
Select whether to create an activity, and if so, what kind of activity.
Create Activity
Blank Activity
Fullscreen Activity
Master/Detail Flow
Blank Activity
Creates a new blank activity, with an action bar and optional navigational elements such as tabs or horizontal swipe.
< Back
Next >
Finish
Cancel

继续点“next”出现如下界面，设置如下：

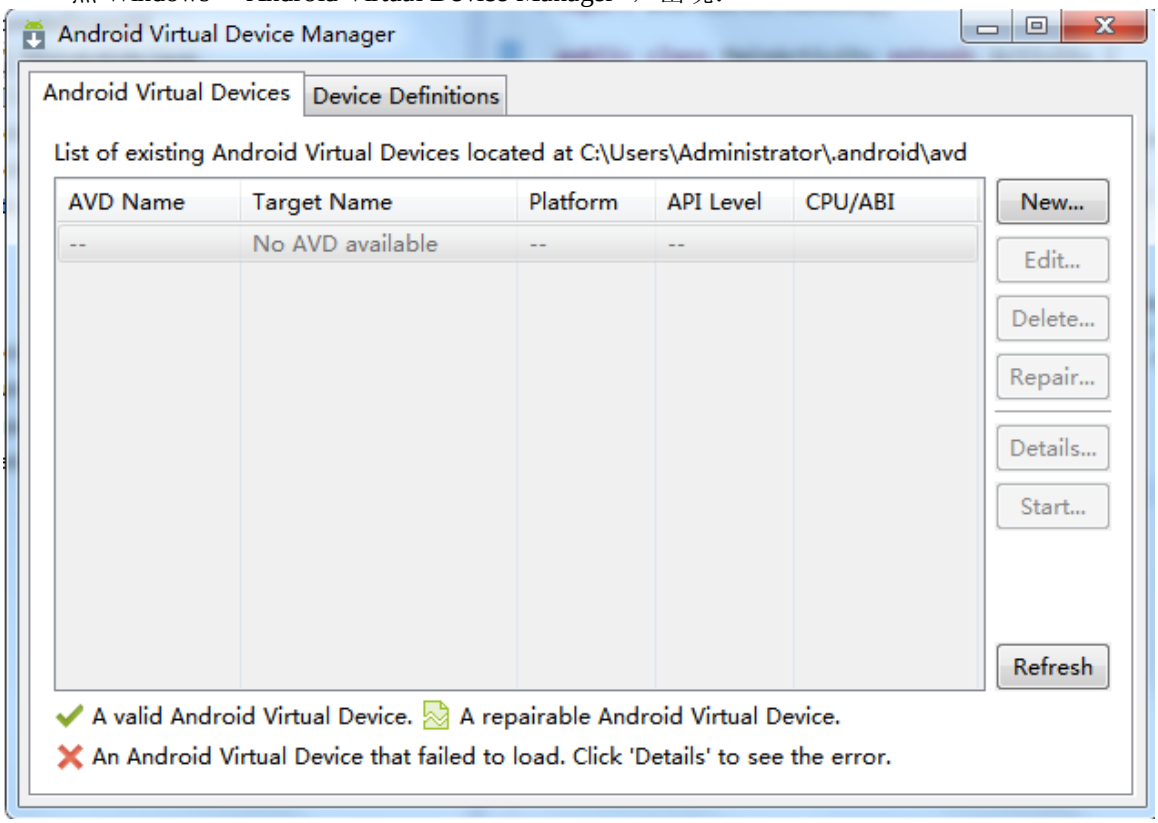


最后点击 Finish，一个简单的工程建立完毕。

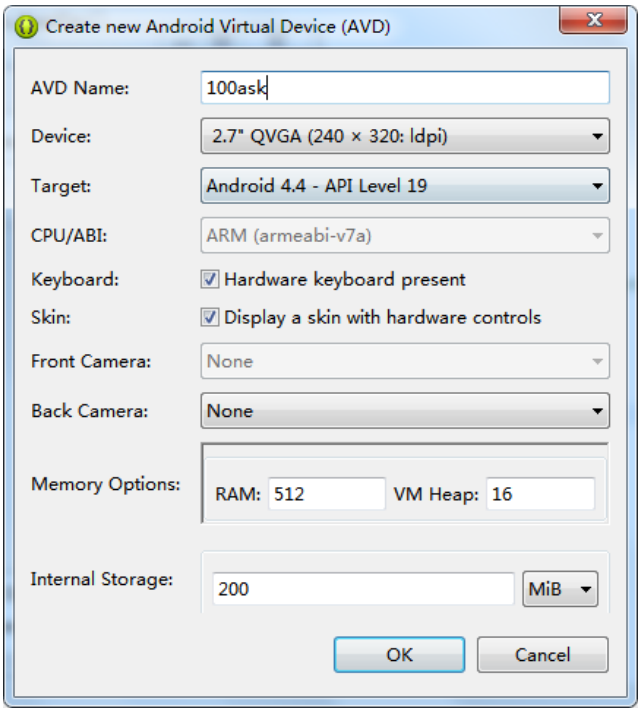
2. 编译工程

2.1 新建模拟器

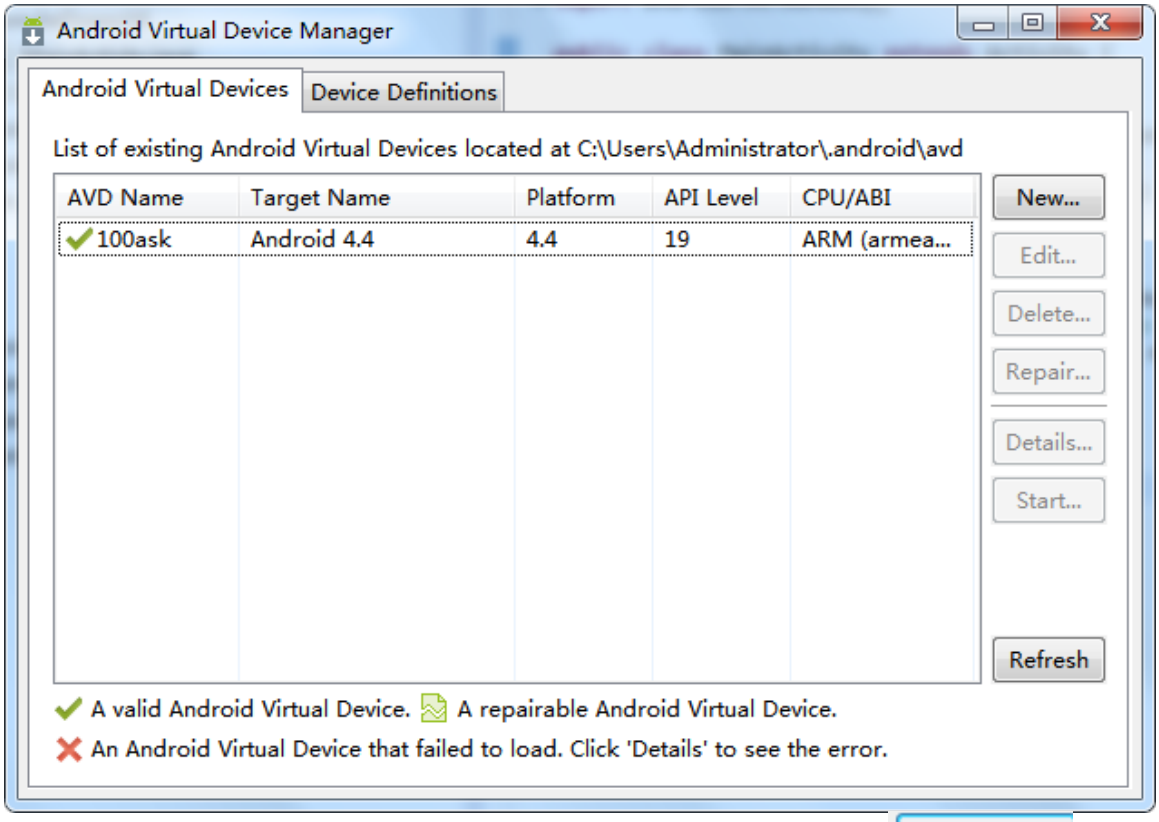
点 Windows-->Android Virtual Device Manager ， 出现：



第一次打开时按键 New 是灰色的，不能点击，到处点击使得出现按键 New，然后点 New, 设置如下：



接着点 OK 出现：



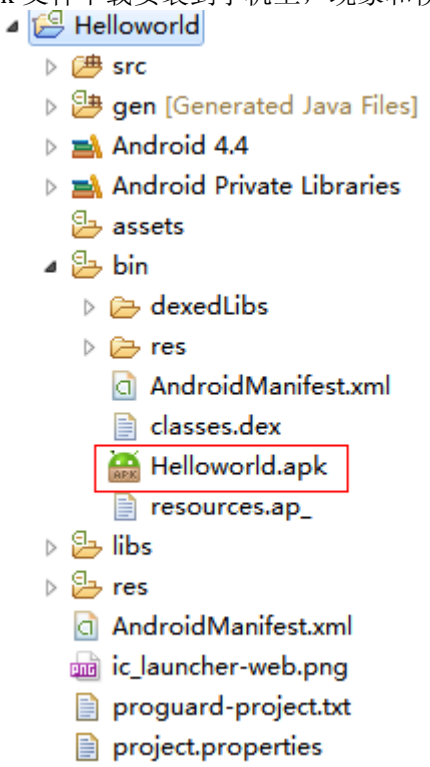
选中 AVD Name 的“100ask”，然后点击“start”启动模拟器．再点 Launch 。模拟器就启动好了。下面就可以运行前面新建的工程 helloworld。

2.2 右击工程名 helloworld --->Run As ---> 1. Android application ， 出现

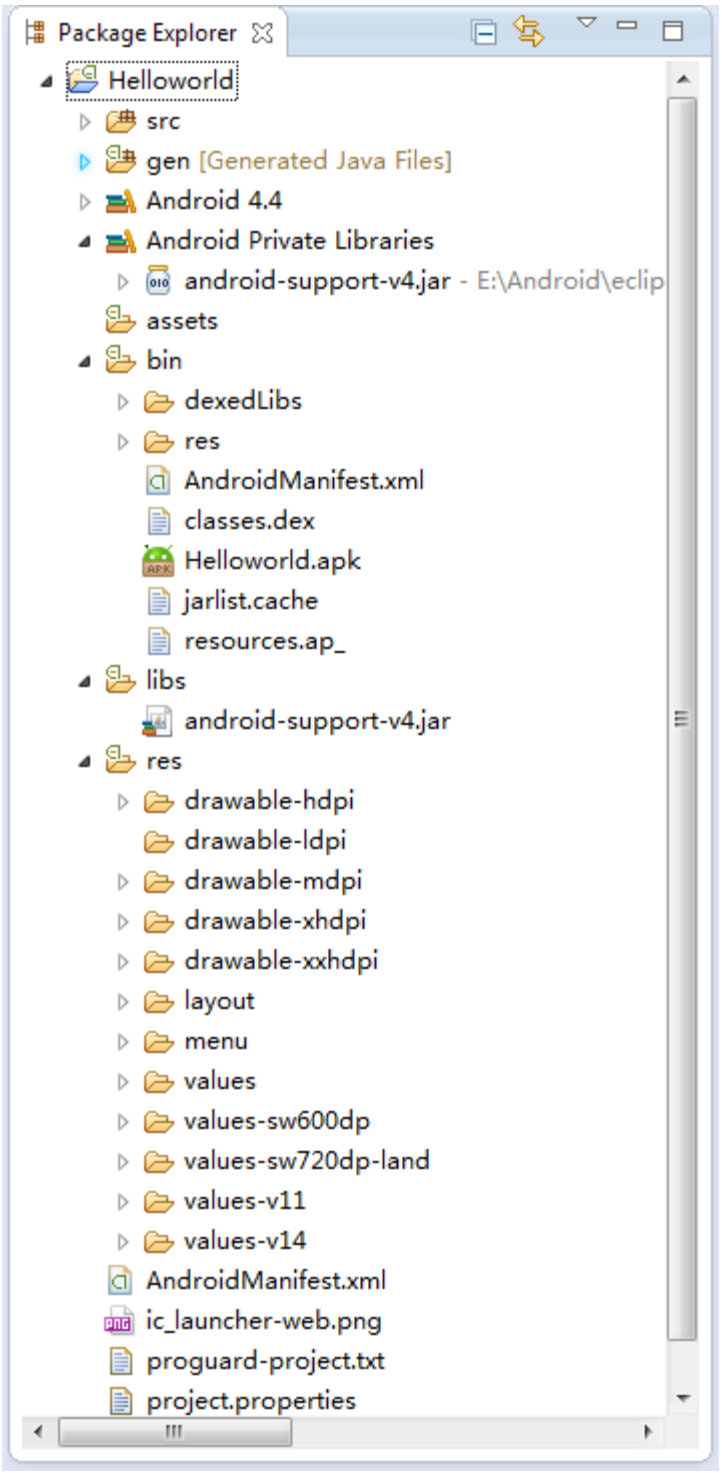


激动人心的 Helloworld.apk 就出来了！

也可以将下图中的 apk 文件下载安装到手机上，现象和模拟器一样！



五、简析 **helloworld** 工程的目录和作用



上图就是整个工程的目录

src：我们写的 JAVA 源代码，在这个目录下还可以新建子目录

Gen：eclipes 插件自动生成的，不能修改，里面的 R.java 如下：

```java
/* AUTO-GENERATED FILE.  DO NOT MODIFY.

package com.helloworld;

public final class R {
    public static final class attr {
    }
    public static final class dimen {
        /**  Default screen margins, per the Android Design guidelines.

         Customize dimensions originally defined in res/values/dimens.xml (such as
         screen margins) for sw720dp devices (e.g. 10" tablets) in landscape here.

         */
        public static final int activity_horizontal_margin=0x7f040000;
        public static final int activity_vertical_margin=0x7f040001;
    }
    public static final class drawable {
        public static final int ic_launcher=0x7f020000;
    }
    public static final class id {
        public static final int action_settings=0x7f080000;
    }
    public static final class layout {
        public static final int activity_main=0x7f030000;
    }
    public static final class menu {
        public static final int main=0x7f070000;
    }
```

工程的每一个资源都会在这个文件生成相应的 ID，src 目录的代码就能根据这个 ID 通过 API 使用相应的资源

Android 4.4

Android Private Libraries: 都是 Android 需要的库，也无需修改

assets: 资源文件，一般是 html 文件

bin 目录下有 dexedLibs 和 res

dexedLibs : android 支持的 java 库
res ：编译前什么也没有，编译后，有 AndroidManifest,classes.dex，Helloworld.apk，jarlist.cache，resource.ap_

AndroidManifest ： 整个的应用程序的管理文件，不可缺少，这个是来自工程的根目录的 AndroidManifest，一个应用程序只有一个 AndroidManifest

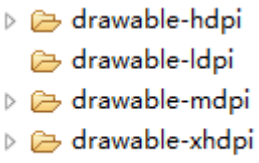
classes.dex:jre 虚拟机文件，编译后，工程所有类的集合都在这个文件

Helloworld.apk ：用过 Android 手机的都知道，这就是可以安装到手机上的 apk

jarlist.cache：所依赖 java 库的的缓存文件，不需要编辑，编译后自动生成

resource.ap_：应用程序使用的所有资源的 16 进制文件，编译后自动生成

libs : android 的 java 库



根目录下的 res 目录 ： 包含 , 这 5 个目录是放的是不同分辨率下的.png 图片文件

layout 目录: 一般是 xxx.xml，应用程序的布局文件，表示着应用程序控件的布局

menu 目录: xml 文件，表示着按下手机的 Menu 按钮后，执行相应操作的布局文件

values : 本 SDK 的 API 下的资源文件，如字符串资源，尺寸大小，程序风格

values-sw600dp

values-sw720dp-land : 根据 values 目录在 sw720dp devices 自动生成的 xml 文件，无需编辑

values-v11

values-v14 : 其他 API 下的 values 文件，自动生成，也无需编辑

AndroidManifest.xml:

真正的应用程序布局，当增加了 Activity，或者需要指定哪个 java 文件先执行时，也在这设置，还可以在这设置应用程序的权限，非常重要

ic_launcher-web.png : 应用安装后的图标

proguard-project.txt : 一些规则，无需编辑

project.properties: 表示目标 sdk 的 API 版本，由插件自动生成