

## Android API Demos 2.3 学习笔记

作者	snowdream
时间	2011 年 08 月 16 日

***谨以此书献给所有和我一样热爱Android的Coder!***

## 前言

由于 Android 从诞生到现在并不是很久，与之有关的学习资料也不是很多。因此对于学习 Android 的人来说，Android SDK 附带的 API Demos 无疑是最好的学习资料。

本书作者试图通过自身学习实践，不断总结，记录笔记，来熟悉和掌握 Android 平台开发相关基础知识，并为后来者学习 Android 提供参考。

作 者

2011 年 8 月

# 目录

第 1 章	导言.....	4
1.1	搭建 Android 开发环境.....	4
1.1.1	搭建 JDK 开发环境.....	4
1.1.2	下载并安装 Eclipse.....	5
1.1.3	下载 Android SDK 以及搭建 Android 开发环境.....	6
1.1.4	创建 Android 虚拟设备 AVD.....	11
1.2	创建第一个 Android 项目 ( Hello World! ) .....	13

# 第 1 章 导言

## 1.1 搭建 Android 开发环境

本书主要介绍在 Ubuntu 11.04 + JDK 7 环境下，如何搭建 Android 开发环境。如果您需要在 Windows 下搭建 Android 开发环境，请参考网络相关内容。

### 1.1.1 搭建 JDK 开发环境

第一步：下载 JDK 7 压缩包

```
wget -c http://download.oracle.com/otn-pub/java/jdk/7/jdk-7-linux-i586.tar.gz
```

(注：如果下载不下来，建议使用迅雷下载，然后拷贝到 Linux 系统上。)

第二步：解压安装

```
sudo tar zxvf ./jdk-7-linux-i586.tar.gz -C /usr/lib/jvm
cd /usr/lib/jvm
sudo mv jdk1.7.0/ java-7-sun
```

第三步：修改环境变量

```
vim ~/.bashrc
```

在该文件末尾添加以下内容：

```
export JAVA_HOME=/usr/lib/jvm/java-7-sun
export JRE_HOME=${JAVA_HOME}/jre
export CLASSPATH=.:${JAVA_HOME}/lib:${JRE_HOME}/lib
export PATH=${JAVA_HOME}/bin:$PATH
```

保存退出，输入以下命令使之立即生效。

```
source ~/.bashrc
```

第四步：配置默认 JDK 版本

由于 Ubuntu 中可能会有默认的 JDK，如 OpenJDK。为了使默认使用的是我们安装的 JDK 7，还要进行以下操作。

执行代码：

```
sudo update-alternatives --install /usr/bin/java java /usr/lib/jvm/java-7-sun/bin/java 300
sudo update-alternatives --install /usr/bin/javac javac /usr/lib/jvm/java-7-sun/bin/javac 300
```

执行代码

```
sudo update-alternatives --config java
```

系统会列出各种 JDK 版本，如下所示：

```
snowdream@snowdream:~$ sudo update-alternatives --config java
```

有 3 个候选项可用于替换 java (提供 /usr/bin/java)。

选择	路径	优先级	状态
-----			
* 0	/usr/lib/jvm/java-6-openjdk/jre/bin/java	1061	自动模式
1	/usr/lib/jvm/java-6-openjdk/jre/bin/java	1061	手动模式
2	/usr/lib/jvm/java-6-sun/jre/bin/java	63	手动模式
3	/usr/lib/jvm/java-7-sun/bin/java	300	手动模式

要维持当前值[\*]请按回车键，或者键入选择的编号：3

update-alternatives: 使用 /usr/lib/jvm/java-7-sun/bin/java 来提供 /usr/bin/java (java)，于手动模式中。

#### 第四步：测试

在终端中输入 `java -version`，测试 JDK 环境是否安装成功。

```
snowdream@snowdream:~$ java -version
java version "1.7.0"
Java(TM) SE Runtime Environment (build 1.7.0-b147)
Java HotSpot(TM) Server VM (build 21.0-b17, mixed mode)
```

### 1.1.2 下载并安装 Eclipse

第一步：下载并安装 Eclipse （官方网站下载：<http://www.eclipse.org/downloads/>）

根据实际情况，推荐安装以下版本：

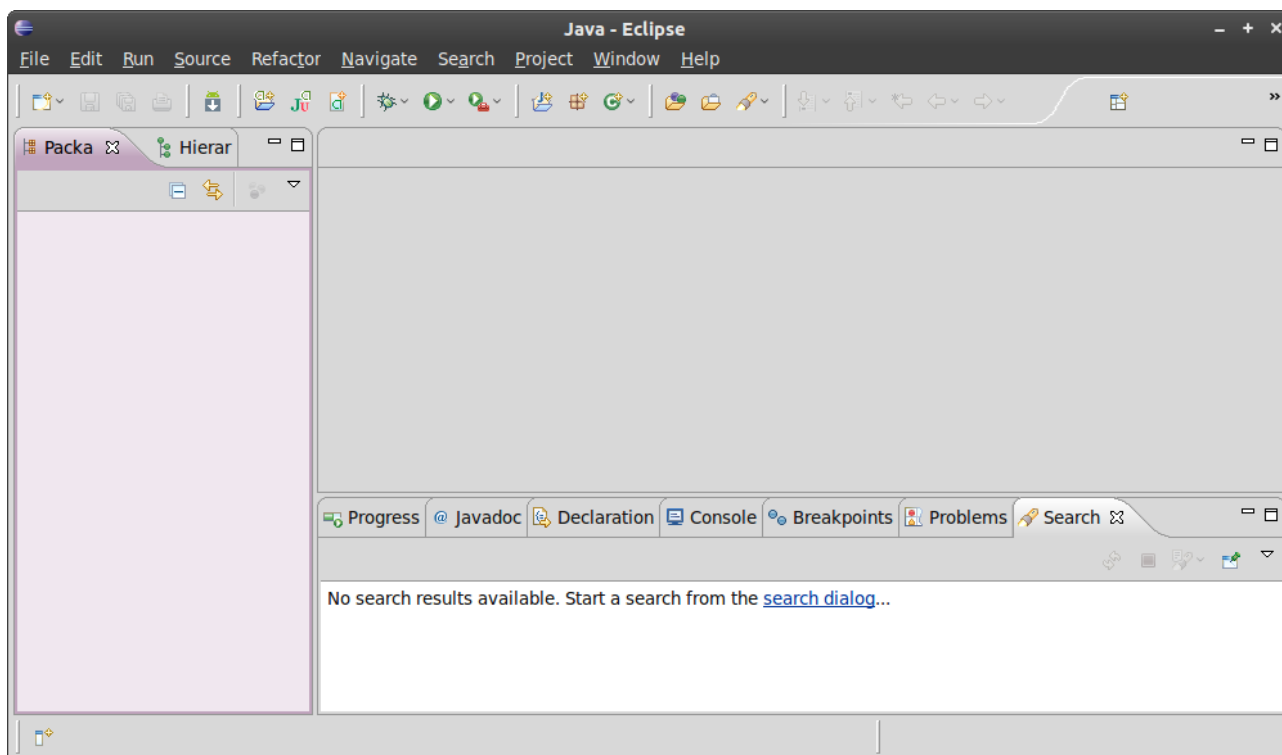
**Eclipse IDE for Java and Report Developers, 250 MB**

（ Ubuntu11.04 32 位系统请直接通过以下命令下载并安装 Eclipse ）

```
wget -c
http://mirror.bjtu.edu.cn/eclipse/technology/epp/downloads/release/indigo/R/eclipse-
reporting-indigo-linux-gtk.tar.gz
tar zxvf eclipse-reporting-indigo-linux-gtk.tar.gz
```

## 第二步：测试

进入 Eclipse 安装目录，双击 Eclipse 可执行程序，如果依次出现以下画面，则 Eclipse 安装成功。



### 1.1.3 下载 Android SDK 以及搭建 Android 开发环境

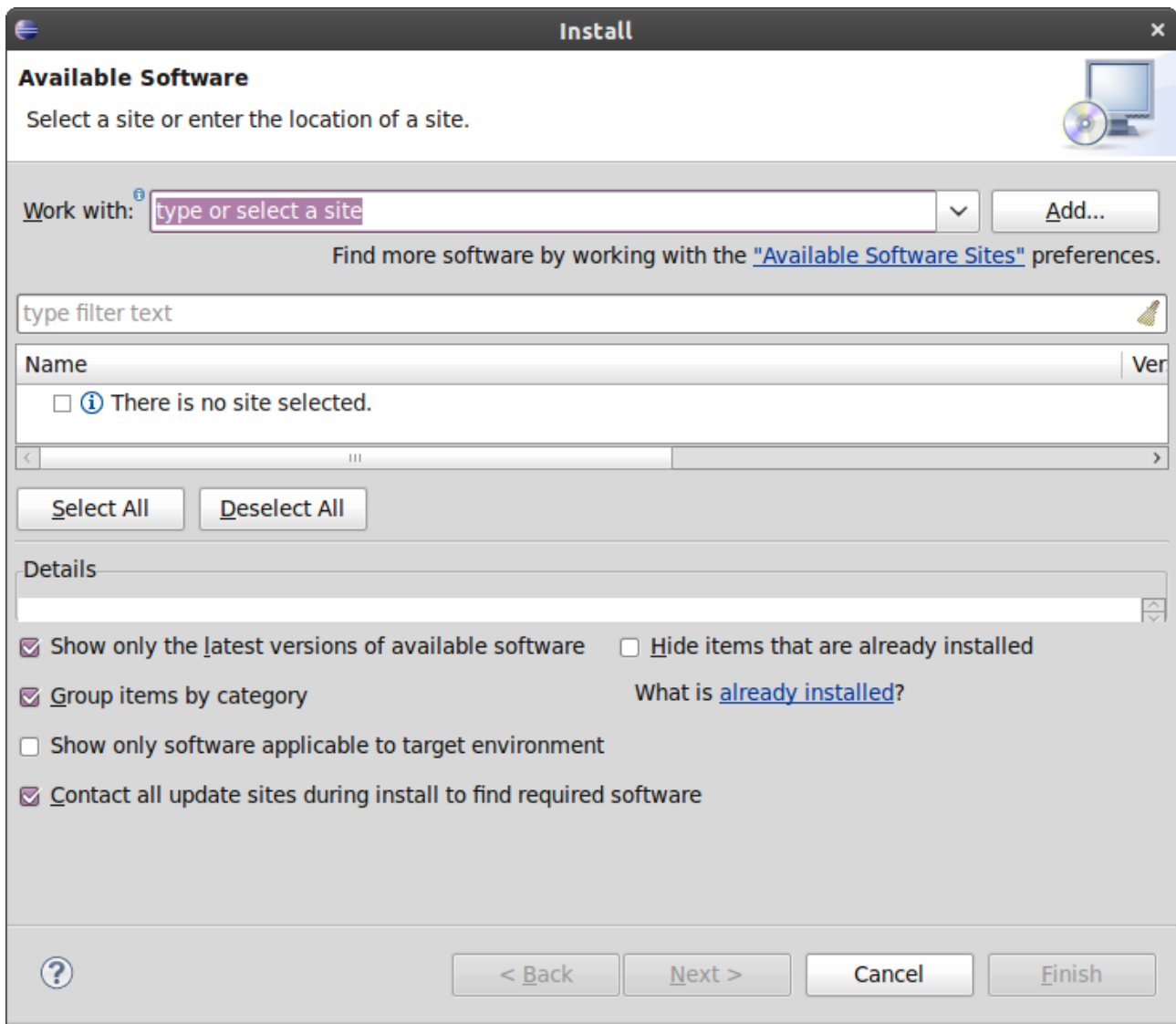
第一步：下载并安装 Android SDK

```
wget -c http://dl.google.com/android/android-sdk\_r12-linux\_x86.tgz  
tar zxvf android-sdk_r12-linux_x86.tgz
```

第二步：在线安装 Eclipse 插件 ADT

启动 Eclipse，然后依次选择菜单：Help > Install New Software....

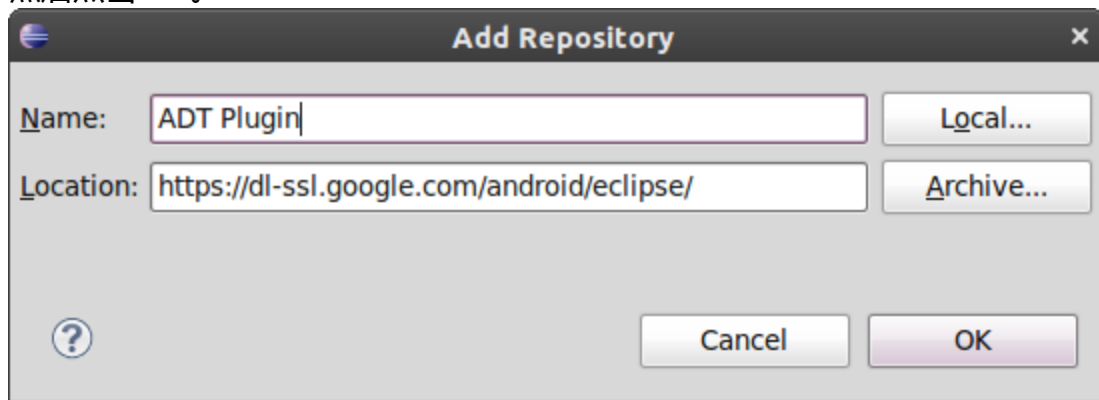
在窗口右上角点击 Add 按钮



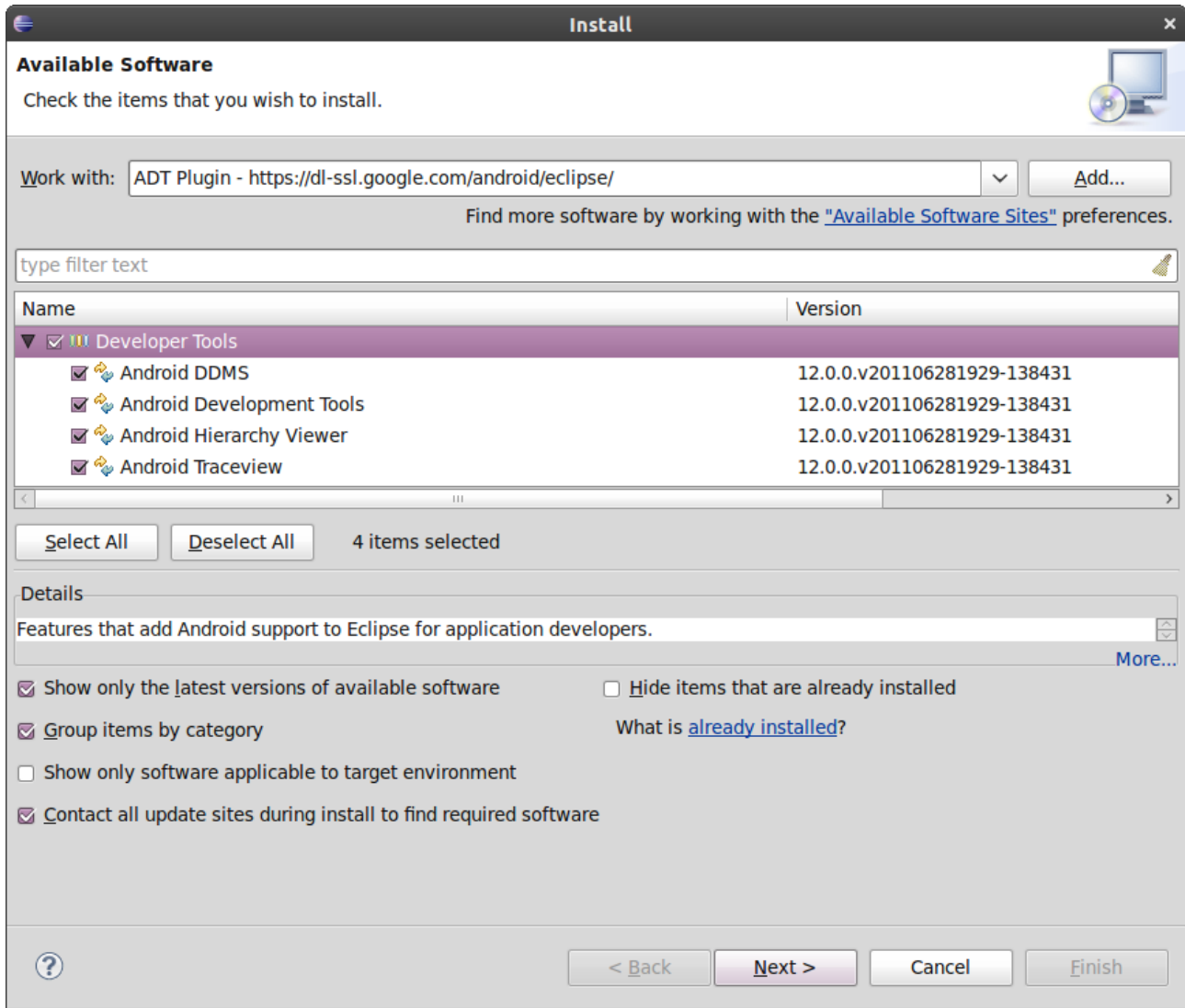
在添加源地址窗口中，在 Name 字段后面填写"ADT Plugin"，而在下面的 Location 字段后面填写以下地址：

<https://dl-ssl.google.com/android/eclipse/>

然后点击 OK。



在可选软件窗口，点击 Select All 全部安装，然后点击 Next。



在下一个窗口，你会看到一系列即将被下载的工具，点击 Next。

阅读并且接受软件协议，然后点击 Finish。

第三步：配置 Eclipse 插件 ADT

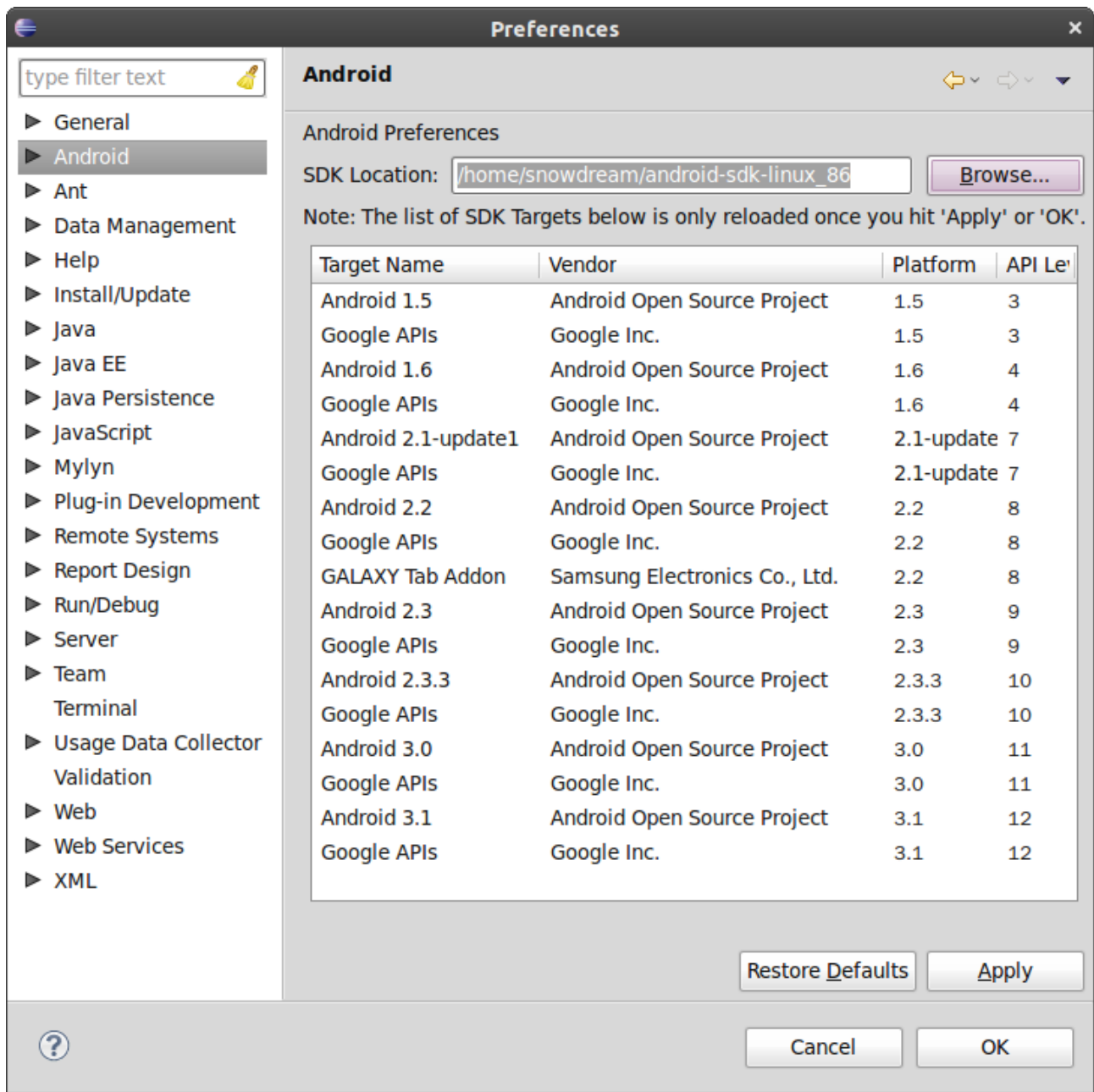
启动 Eclipse，然后依次选择菜单：Window > Preferences...

在左边的面板上选择 Android 选项，如下所示：

点击 Browse... 并且定位到你的 Android SDK 目录,例如 /home/snowdream/android-sdk-linux\_86

先点击 Apply, 然后点击 OK。

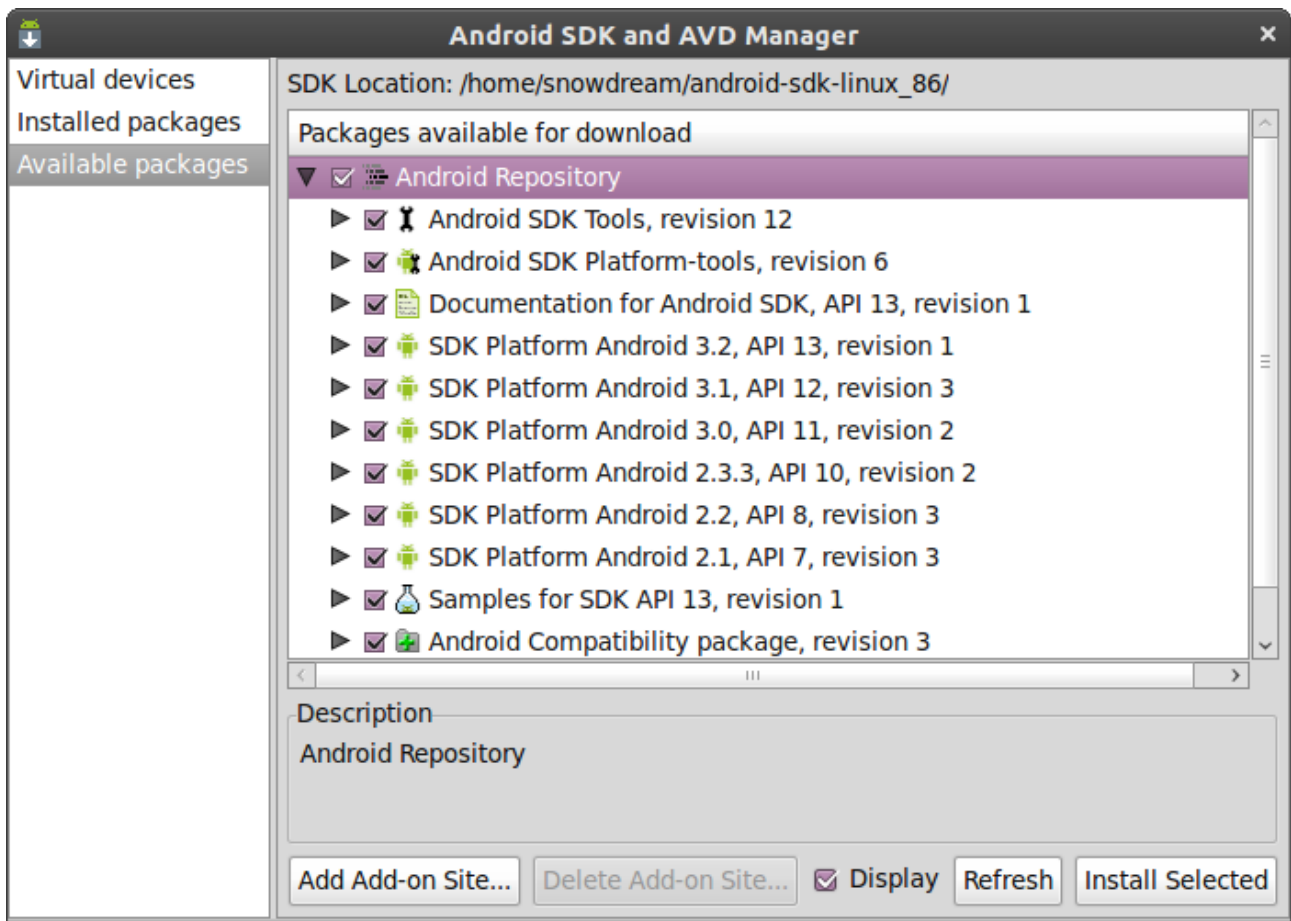




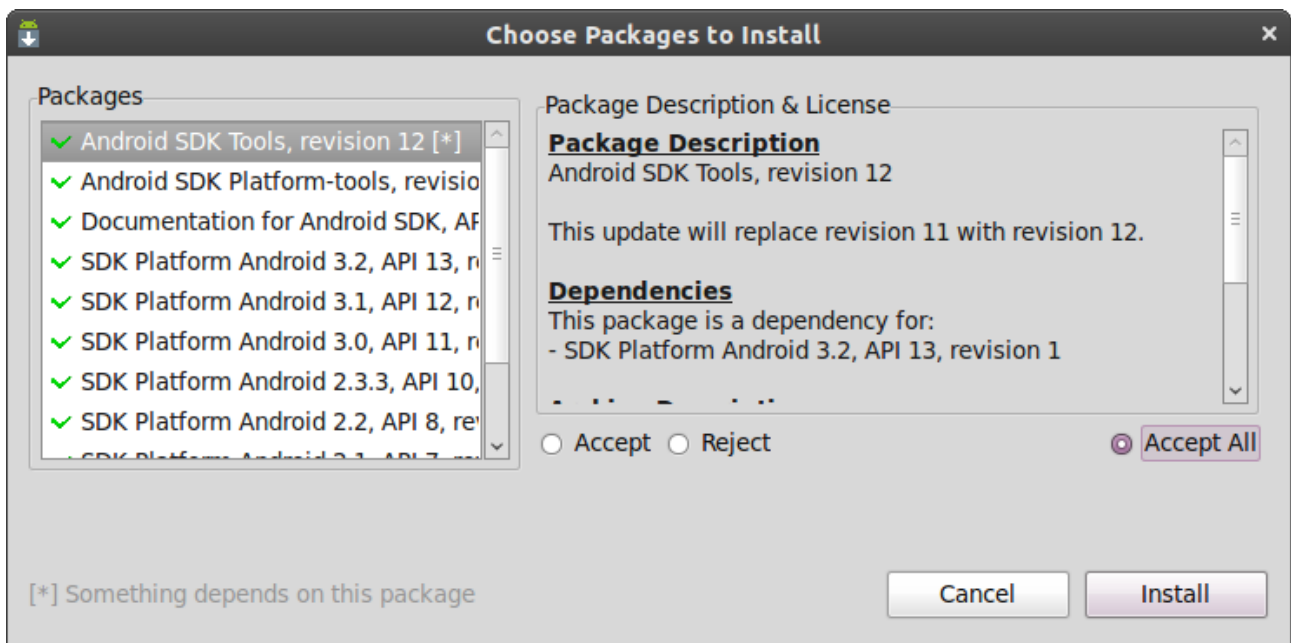
#### 第四步：添加 Android SDK 组件

启动 Eclipse，然后依次选择菜单：Window > Android SDK and AVD Manager

在左侧面板上选择 Available Packages，这将会在右侧显示 SDK 源中所有可以进行下载安装的组件。



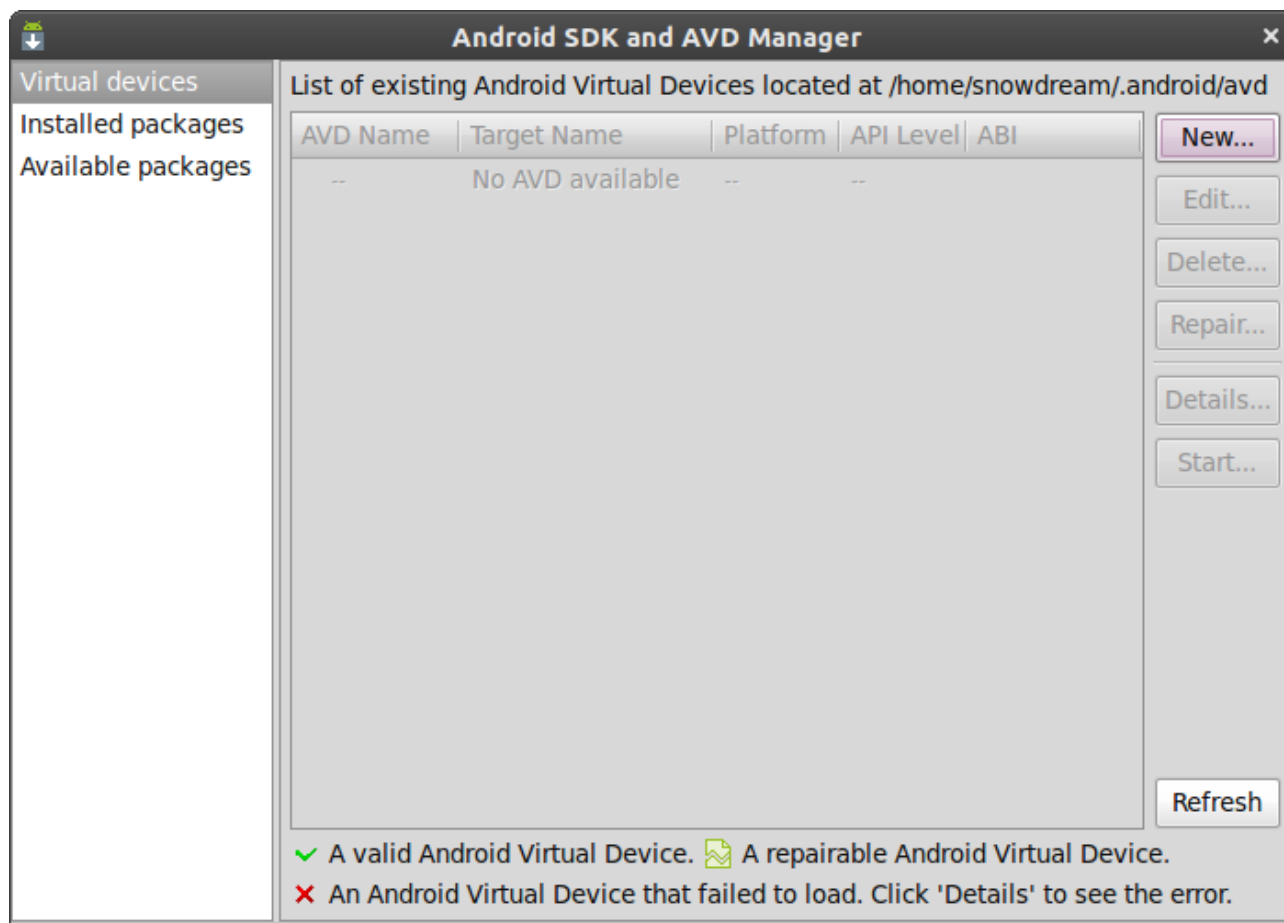
根据需求，选择你所需要安装的组件，然后点击 **Install Selected**。在接下来弹出的阅读协议窗口中，选择 **Accept All**，然后点击 **Install**。这些组件将会安装到您的 Android SDK 安装目录。



**注：**这一步可能需要花费数小时，具体时间和您的网络环境密切相关，请耐心等待。安装完成后，根据提示，需要重新启动 Eclipse 才能应用更新。

### 1.1.4 创建 Android 虚拟设备 AVD

启动 Eclipse , 然后依次选择菜单：Window > Android SDK and AVD Manager



在左侧面板上选择 Virtual Devices , 然后在右上角点击 New... 新建 AVD 设备 , 如下所示：

**Create new Android Virtual Device (AVD)**

Name:

Target:

ABI:

SD Card:

☒ Size:

☐ File:

Snapshot: ☒ Enabled

Skin:

☒ Built-in:

☐ Resolution:  x

Hardware:

Property	Value
Abstracted LCD density	240
Max VM application heap size	24

☐ Override the existing AVD with the same name

注明：

Name：填写 AVD 名称，例如 android2.3

Target：根据常用的 SDK 版本进行选择，例如，Android 2.3-API Level 9

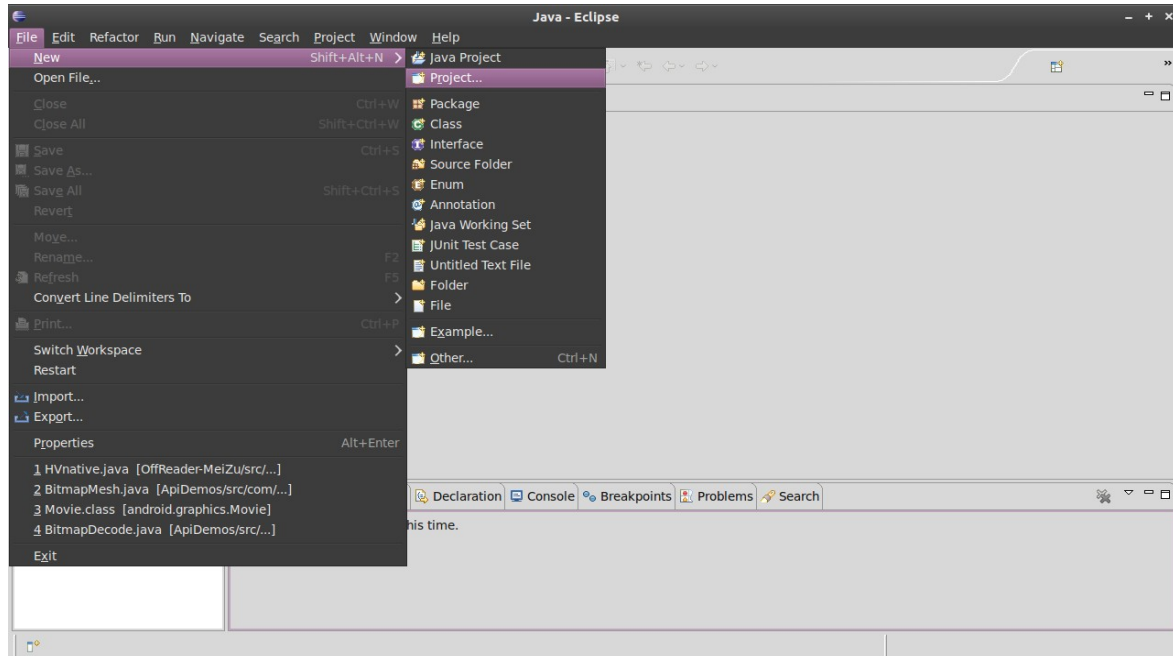
Size：虚拟 sd 卡容量大小，根据实际需求设置，例如 200MiB

Built-in：选择 AVD 的皮肤，这里保持默认选项

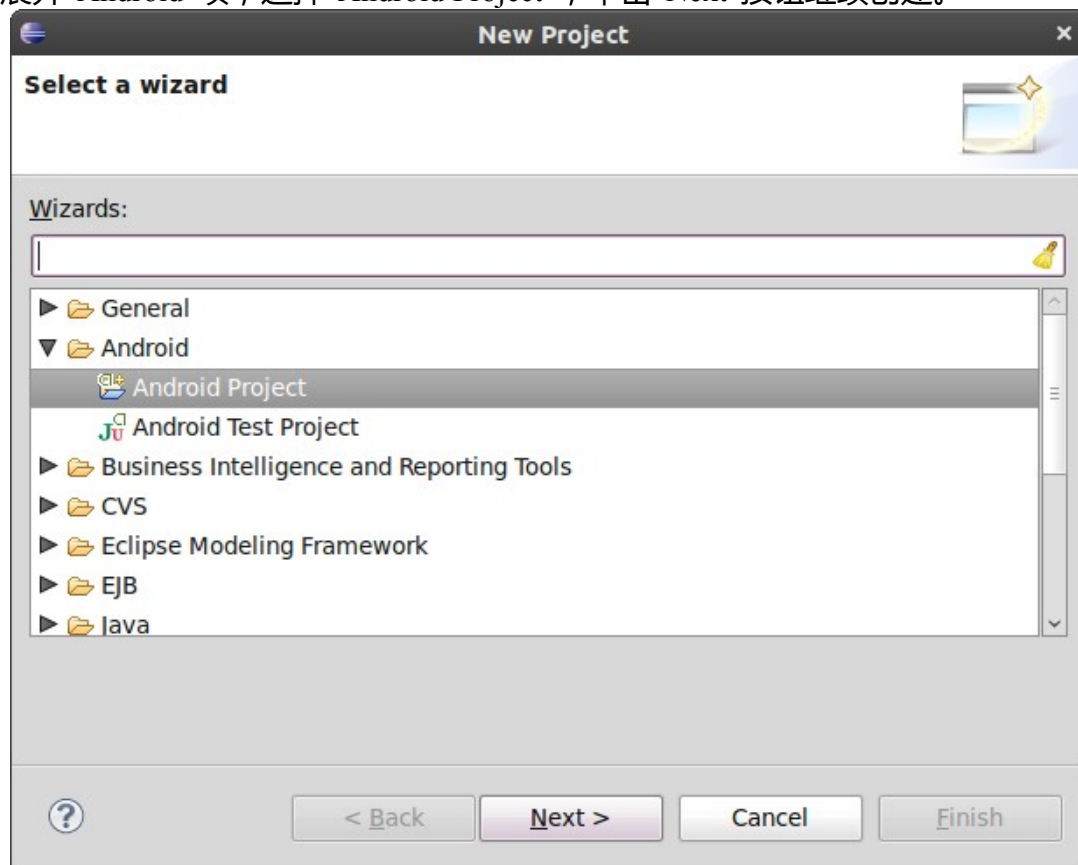
## 1.2 创建第一个 Android 项目 ( Hello World! )

第一步：根据新建项目向导创建项目

启动 Eclipse, 选择"File"--"New"--"Project",打开新建项目向导。

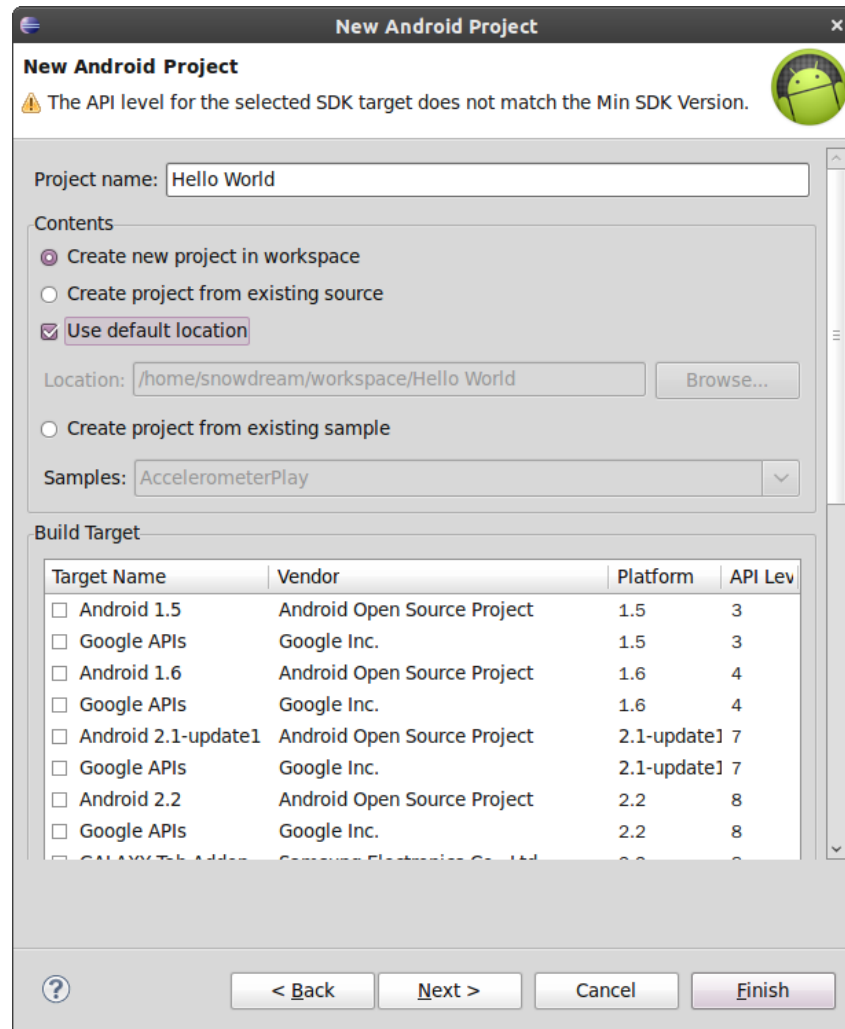


展开"Android"项，选择"Android Project"，单击"Next"按钮继续创建。



在"Project name:"字段后填写项目名称"Hello World"。

注：默认在 Eclipse 工作目录下以项目名称创建一个新文件夹作为该项目的主文件夹，如果您需要自定义项目主文件夹，需要先点击掉"Use default location"选项，然后在下面的"location"字段后面填写自定义路径。



把右边的滚动条往下拉，在"Build Target"下面选择您编译需要使用的 SDK 版本，这里我们选择版本"Android 2.3"。其他字段填写说明如下：

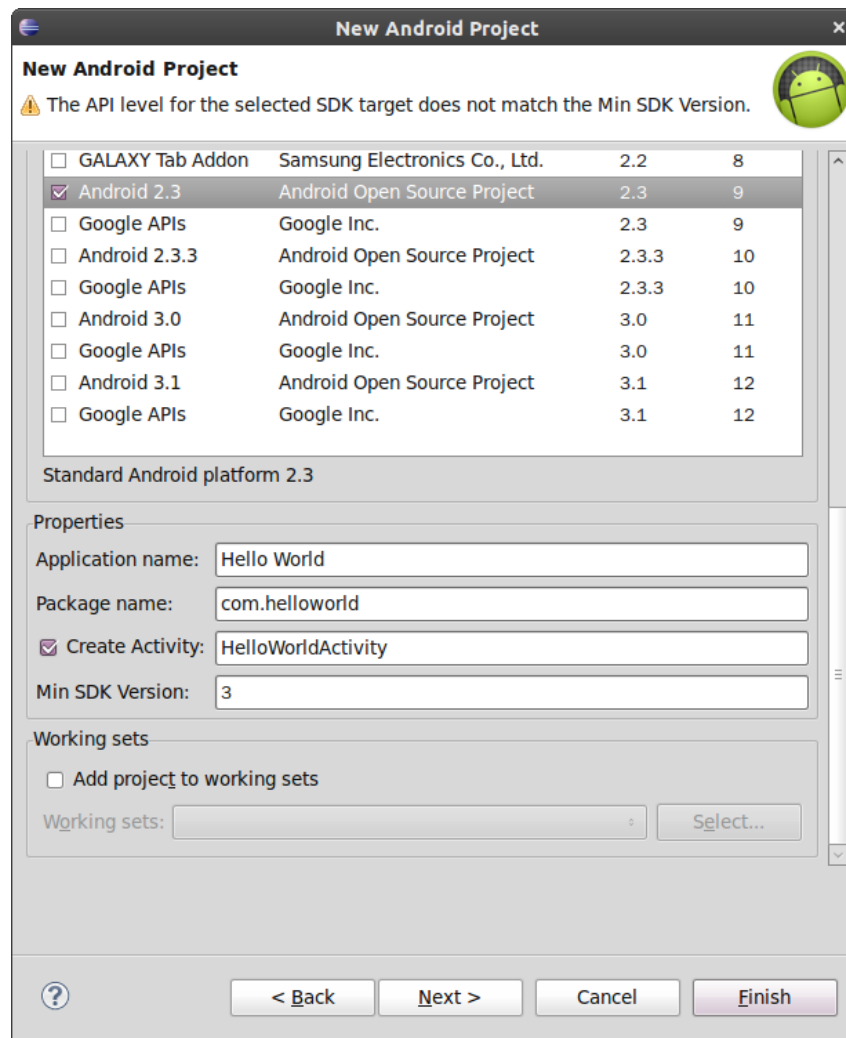
Application name : Hello World //程序名称

Package name: com.helloworld //软件包名称

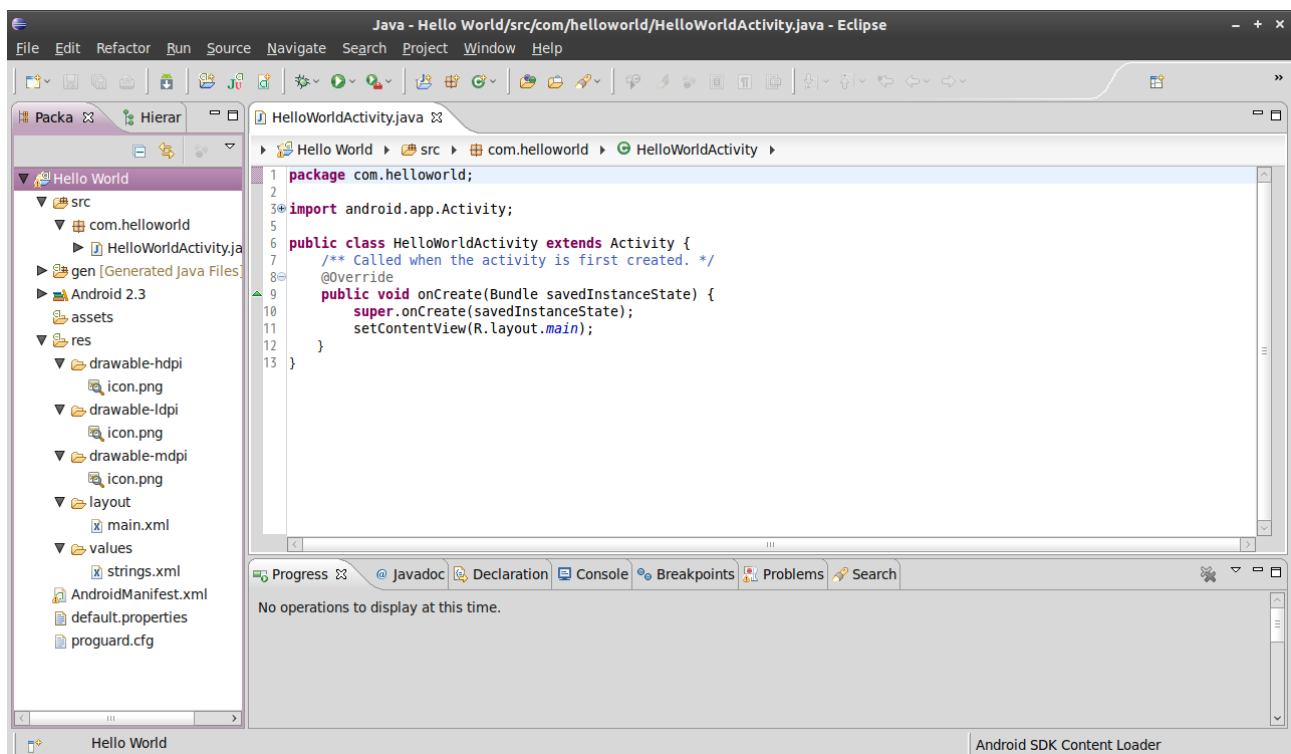
Create Activity: HelloWorldActivity //Android 项目主 Activity 名称

Min SDK Version: 3 //向下兼容的最低 Android 版本，对应“ Build Target“下面的” API Level“

如下图所示：

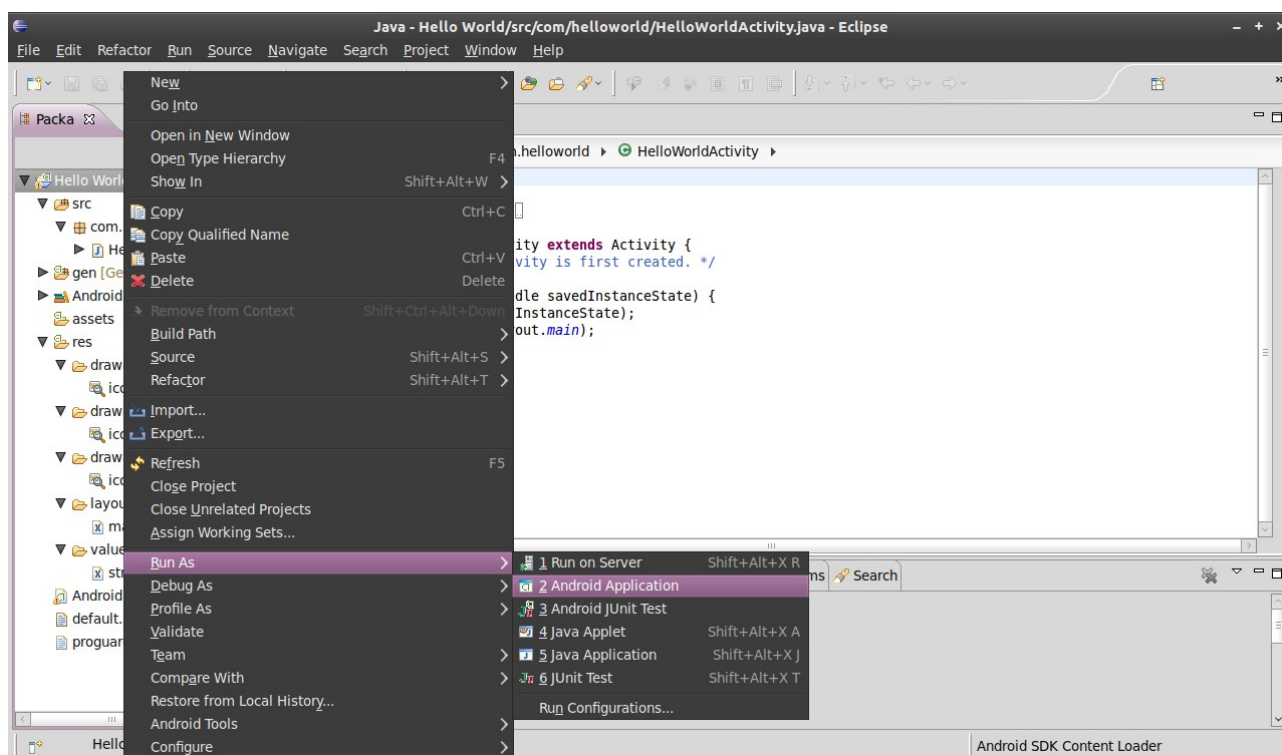


最后单击"Finish"按钮，项目创建完成。



## 第二步：运行 Android 项目程序

在 Eclipse 左侧"Package Explorer"窗口，右键点击刚刚创建好的"Hello World"项目文件夹，在功能菜单上选择"Run As--Android Application"功能。如下图所示：



如果已经创建 AVD 虚拟设备，则会自动启动模拟器。否则，请参考[1.1.4 创建 Android 虚拟设备 AVD](#)章节先创建一个 AVD 虚拟设备。如果您拥有 Android 手机，也可以不用创建该设备，直接使用手机运行调试 Android 程序。

### 运行效果：





