APPLICATION NOTE

How to build your first Android Application in Windows

3/30/2012
Created by:
Micah Zastrow

Abstract

This application note is designed to teach the reader how to setup the Android Development Environment by using Eclipse, and how to create the readers first Android Application. It first shows how to install and setup all of the software that is needed. It then shows how to create an Android project and how to enter sample code to create an Android application. Lastly, it shows how to either use a virtual device, or connect an android device to the computer and run the application on either of these devices.

Keywords

Eclipse: Eclipse is a platform that has been designed from the ground up for building integrated web and application development tooling.

IDE: is a software application that provides comprehensive facilities to computer programmers for software development.

Software Development Kit (SDK): is typically a set of software development tools that allows for the creation of applications for a certain software package

Android Development Tools (ADT): A plugin for eclipse that extends the Eclipse IDE by providing more tools to develop Android Applications.

Java SE Development Kit (JDK): A popular Java SDK that is used to program Android applications.

Android Virtual Device (AVD): An Android emulator that allows you to simulate how the application will run on an actual Android device.

Introduction

Android applications, like most mobile phone applications, are developed in a host-target development environment. In other words, you develop your application on a host computer, and download it to a target mobile phone for testing and ultimate use. To write your own Android mobile phone applications, you'll first need to collect the required tools and set up an appropriate development environment on your PC. The Android SDK supports several different development environments. For this application note, we will focus on using Eclipse, because it is the best integrated with the SDK.

Objective

This application note provides a practical introduction to developing applications for. It explores the framework for constructing an application, and the tools for developing, testing, and publishing software for the platform.

System Requirements

The sections below describe the system and software requirements for developing Android applications using the Android SDK.
http://developer.android.com/sdk/requirements.html

Supported Operating Systems
- Windows XP (32-bit), Vista (32- or 64-bit), or Windows 7 (32- or 64-bit)

Supported Development Environments
Eclipse IDE
- Eclipse 3.6.2 (Helios) or greater
- Eclipse JDT plugin (included in most Eclipse IDE packages)
- If you need to install or update Eclipse, you can download it from http://www.eclipse.org/downloads/.
- Several types of Eclipse packages are available for each platform. For developing Android applications, we recommend that you install one of these packages:
  - Eclipse IDE for Java Developers
  - Eclipse Classic
  - Eclipse IDE for Java EE Developers
  - JDK 6 (JRE alone is not sufficient)
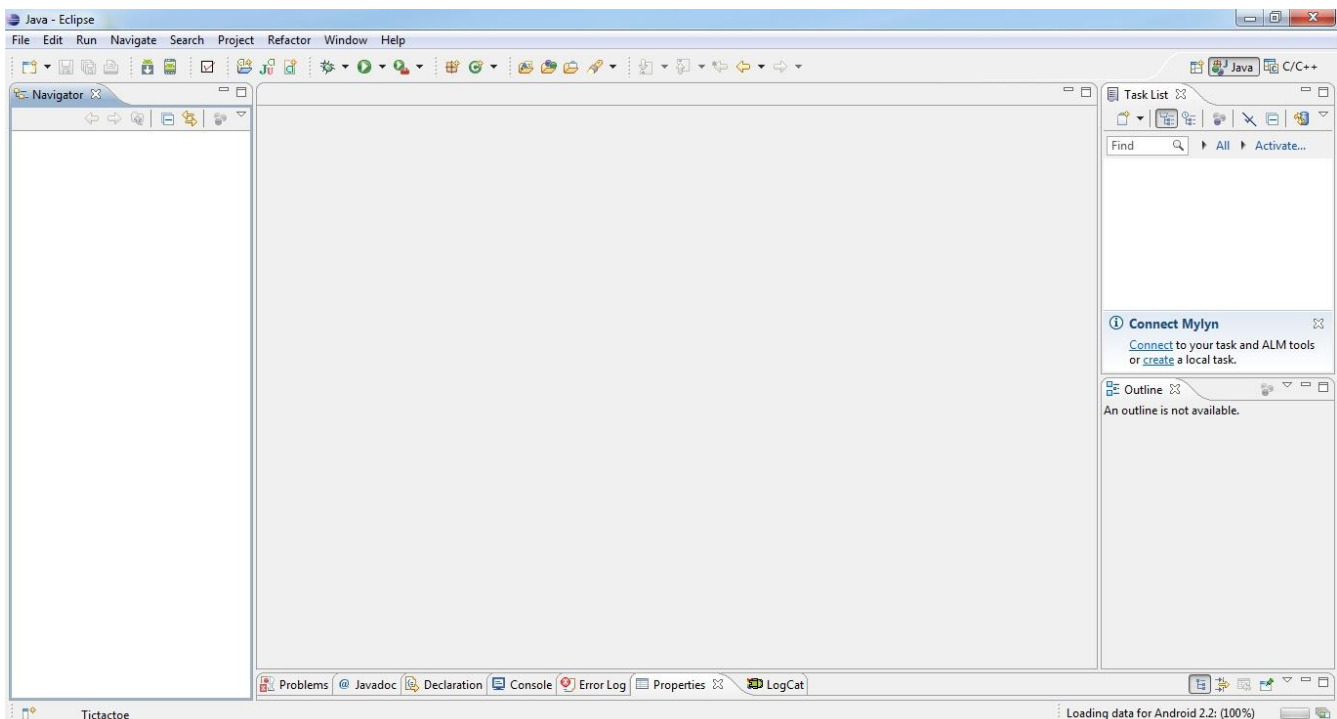 - Android Development Tools plugin (recommended)

Hardware requirements

The Android SDK requires disk storage for all of the components that you choose to install. The table below provides a rough idea of the disk-space requirements to expect, based on the components that you plan to use.

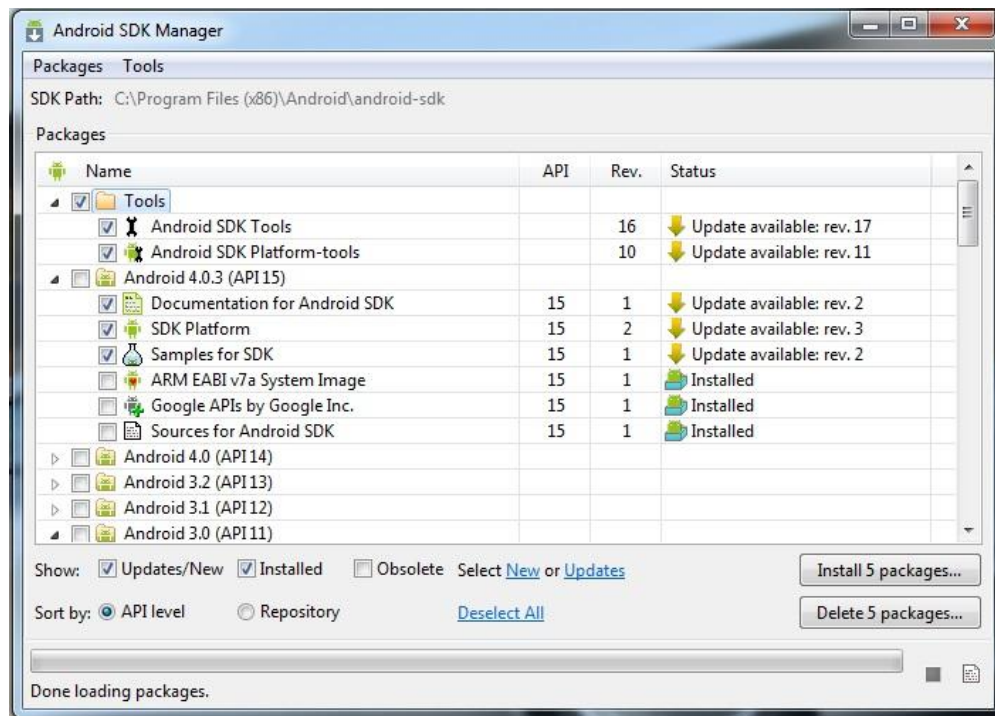| Component type	Approximate size	Comments
SDK Tools	35 MB	Required.
SDK Platform-tools	6 MB	Required.
Android platform (each)	150 MB	At least one platform is required.
SDK Add-on (each)	100 MB	Optional.
USB Driver for Windows	10 MB	Optional. For Windows only.
Samples (per platform)	10M	Optional.
Offline documentation	250 MB	Optional.

Installation

Eclipse: Eclipse is the preferred development environment for Android. You can download eclipse at http://www.eclipse.org/downloads/. It is recommended that you install either classic or the Java Version of eclipse.

JDK: Download the latest JDK by going here:
http://www.oracle.com/technetwork/java/javase/downloads/index.html

Android SDK: Go to http://developer.android.com/sdk/index.html and click on the Windows installer.

Eclipse ADT Plugin:
1. Start Eclipse, then select **Help** > **Install New Software...**.
2. Click **Add**, in the top-right corner.
3. In the Add Repository dialog that appears, enter "ADT Plugin" for the *Name* and the following URL for the *Location*: https://dl-ssl.google.com/android/eclipse
4. Click **OK**
5. In the Available Software dialog, select the checkbox next to Developer Tools and click **Next**.
6. In the next window, you'll see a list of the tools to be downloaded. Click **Next**.

7. Read and accept the license agreements, then click **Finish**.
8. When the installation completes, restart Eclipse.
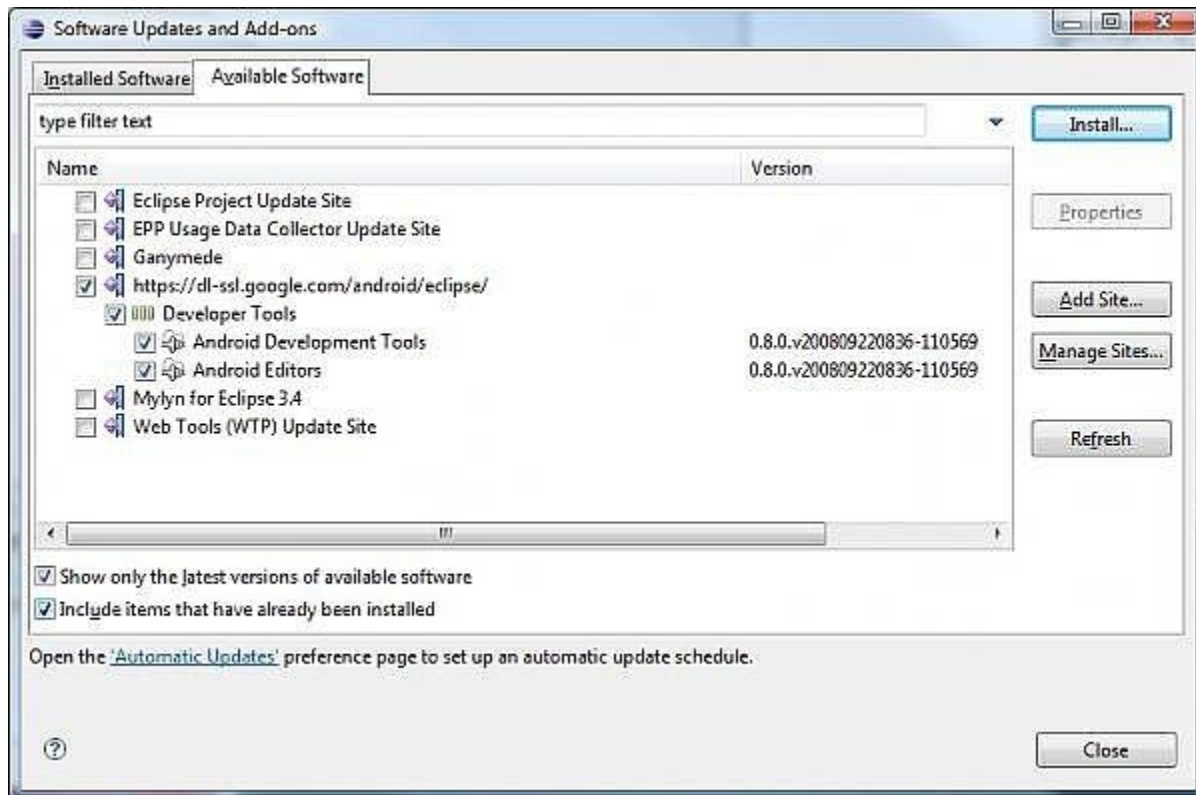
Configuring the ADT Plugin

After you've successfully downloaded the ADT as described above, the next step is to modify your ADT preferences in Eclipse to point to the Android SDK directory:

1. Select **Window** > **Preferences...** to open the Preferences panel
2. Select **Android** from the left panel.
3. For the *SDK Location* in the main panel, click **Browse...** and locate your downloaded SDK directory.
4. Click **Apply**, then **OK**.
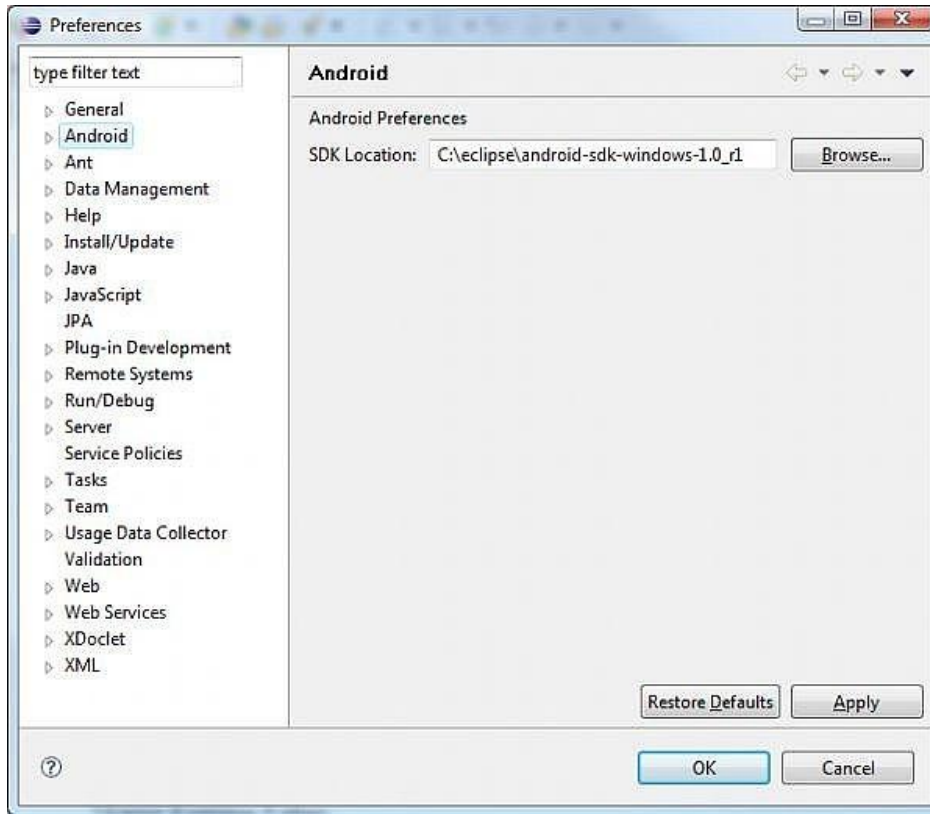
Adding SDK Components.

1. Click on Window->Android SDK Manager.
2. This will allow you to choose the Android platform versions, add-ons tools and other components. Choose the version of Android that you would like you application to work on.
3. Click Install Selected and wait for the components to down. When the download is finished, verify and accept the new components.

Creating an Android Application

1. In Eclipse go to File->New->Project…
2. Select an Android Project from the Android Folder and press Next.
3. Fill in the details of your Android application.
    a. Project Name: The project name and folder that Eclipse will store the

project files

b. Build Target: The version of the Android SDK that will be used when you build your program. Select a platform that is equal to or lower than the target chosen for the AVD.

c. Application Name: This is the name of the application.

d. Package Name: The namespace that all of the source code will reside under.

e. Create Activity: The name for that class stub that is generated by the plugin.

4. The values that are used in this example are:

a. Project Name: HelloAndroid

b. Build Target: 2.1

c. Application Name: Hello, Android

d. Package Name: com.example.helloandroid

e. Create Activity: HelloAndroid

5. Click on Finish.
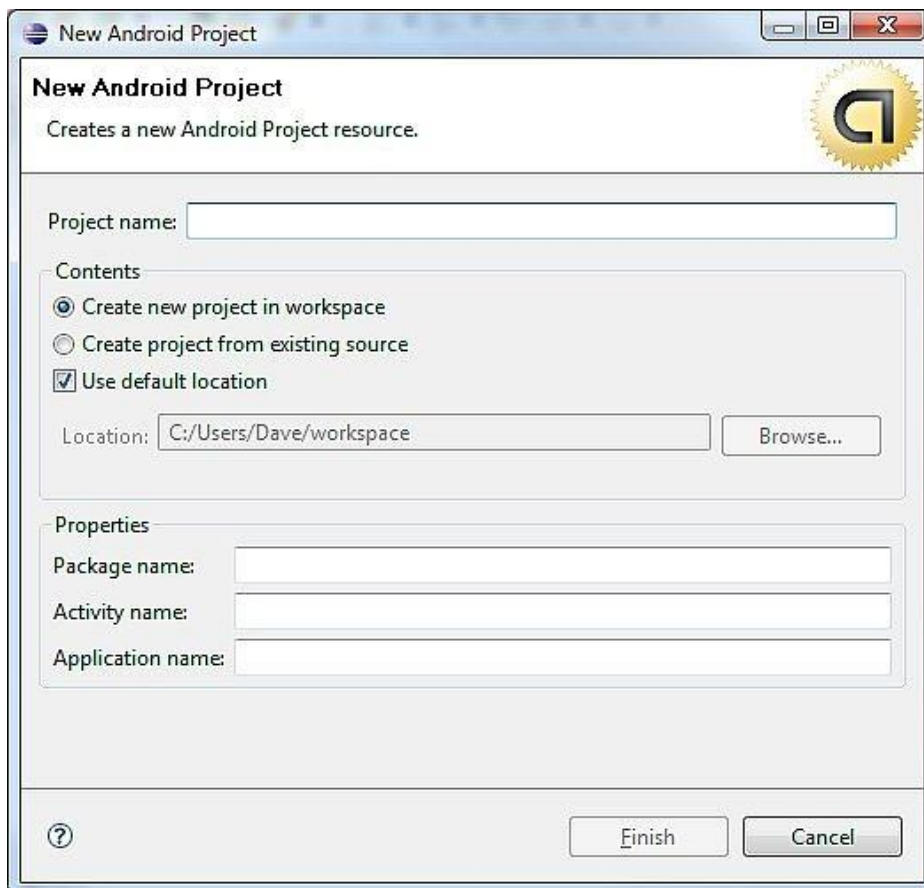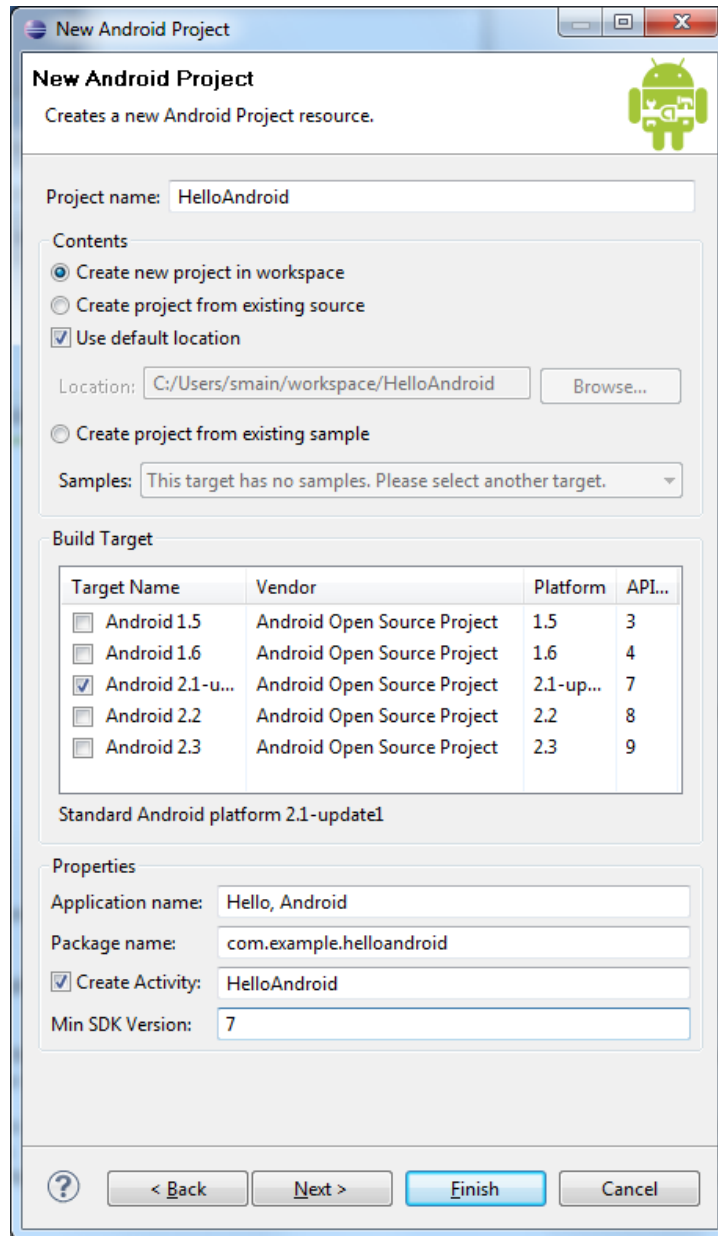
Writing the Code

Your Android project is now ready. It should be visible in the Package Explorer on the left. Open the HelloAndroid.java file, located inside *HelloAndroid > src > com.example.helloandroid*). It should look like this:

```
package com.example.helloandroid;

import android.app.Activity;
import android.os.Bundle;

public class HelloAndroid extends Activity {
  /** Called when the activity is first created. */
  @Override
  public void onCreate(Bundle savedInstanceState) {
    super.onCreate(savedInstanceState);
    setContentView(R.layout.main);
  }
}
```

Take a look at the revised code below and then make the same changes to your HelloAndroid class. The bold items are lines that have been added.

```
package com.example.helloandroid;

import android.app.Activity;
import android.os.Bundle;
import android.widget.TextView;

public class HelloAndroid extends Activity {
  /** Called when the activity is first created. */
  @Override
  public void onCreate(Bundle savedInstanceState) {
    super.onCreate(savedInstanceState);
    TextView tv = new TextView(this);
    tv.setText("Hello, Android");
    setContentView(tv);
  }
}
```

Run the Application

Adding an AVD

1. In Eclipse go to Window->AVD Manager->New.
2. Type in the Name of the AVD and choose a Target. The target is the version of the Android SDK that you would like to run on the emulator.
3. Click Create AVD.
4. Select **Run > Run**.
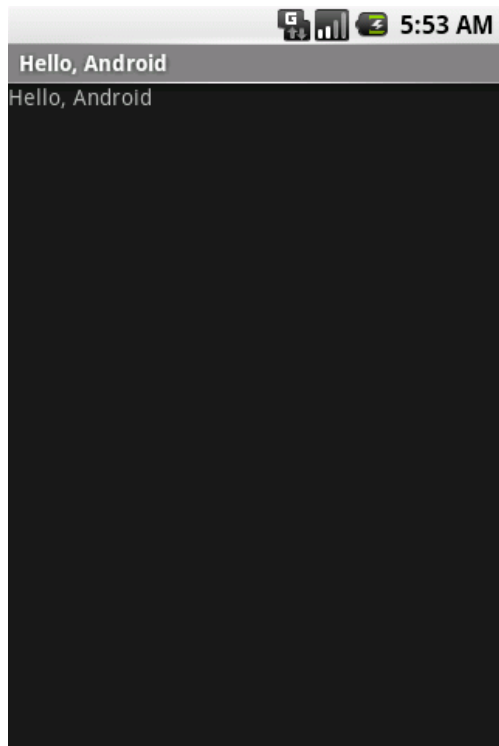5. Select "Android Application".

Using an Android Device (Optional)
If you want to use your own Android Device, then you can follow these steps to setup eclipse to use your device.
1. Select **Run > Run Configurations.**
2. Select your project
3. Click on the Target tab
4. Make sure "Manual" is checked under Deployment Target Selection Mode
5. Click run
6. When the Android Device Chooser opens select your android device and click OK.
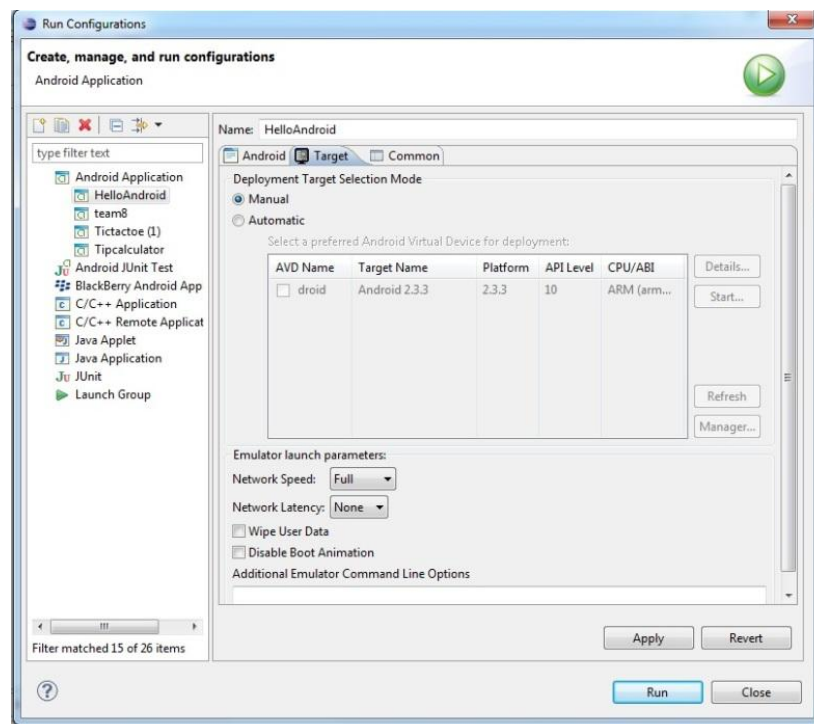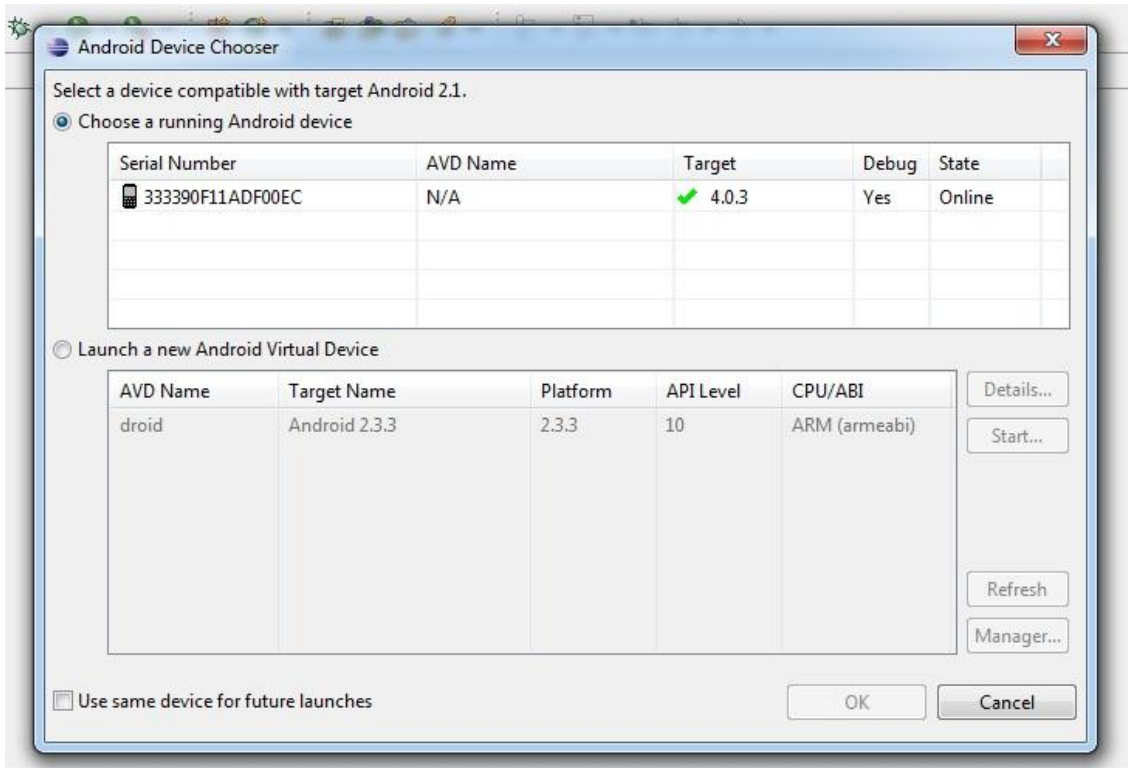
You should now see something like this:

Conclusion

Congratulations you have successfully created an Android application. While this application note taught you how to create a basic application you can use the techniques you learned to expand on your application and turn it into something more powerful.

Resources

http://www.developer.android.com
http://www.eclipse.org/downloads/
http://www.oracle.com/technetwork/java/javase/downloads/index.html