

## Compiling for Android with Eclipse

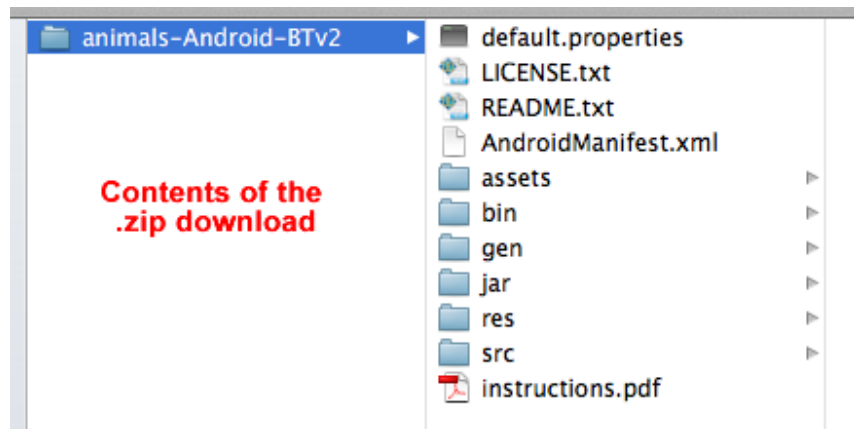
This step-by-step document assumes you have Eclipse installed on your computer along with the latest version of the Android SDK. Any version of Eclipse should work, we use "Galileo" in our workstations. It is also assumed that you successfully compiled the simple Hello World test application. If you have not compiled the simple Hello World application it is very likely that you'll run into trouble compiling your downloaded source-code. Eclipse and the Android SDK can be tricky to setup sometimes and the Hello World application is an easy way to test your setup.

1

### Unzip the application's Project Files

Unzip the archive you downloaded to a safe place on your computer such as "My Documents." Explore this location using Mac's Finder application or Windows Explorer. The directory you downloaded contains 6 sub-directories and several files.

Keep this unzipped archive somewhere permanent on your computer.



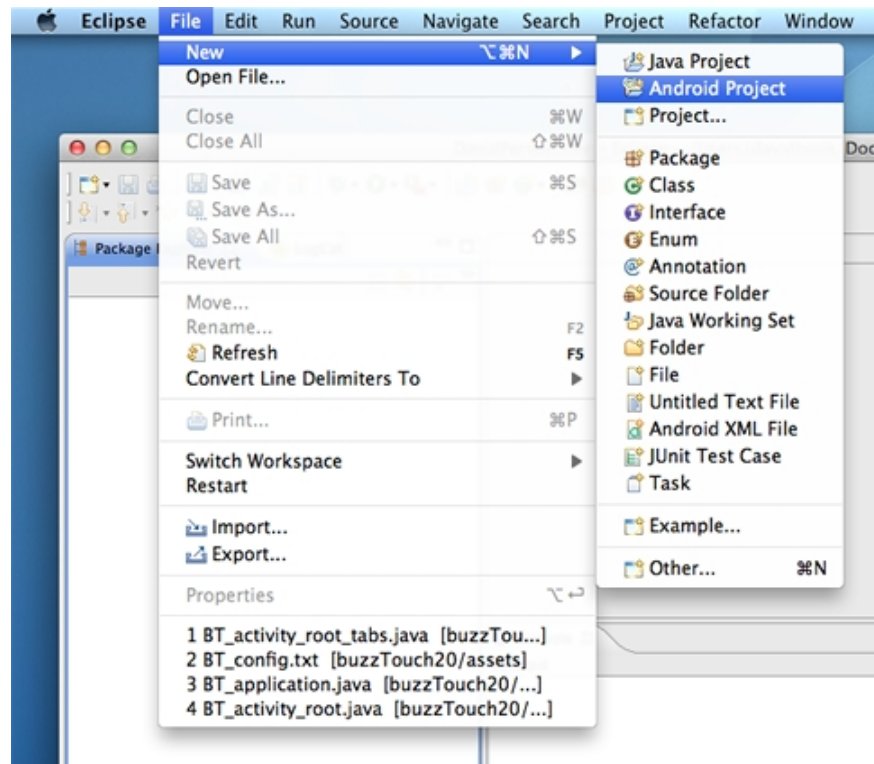
2

### Create a new Android application in Eclipse

Launch the Eclipse development environment. When Eclipse loads you could see any number of views or panels. Eclipse refers to these views as "perspectives" and customizing which panels are open when Eclipse loads is common. The default views will be fine for this step-by-step.

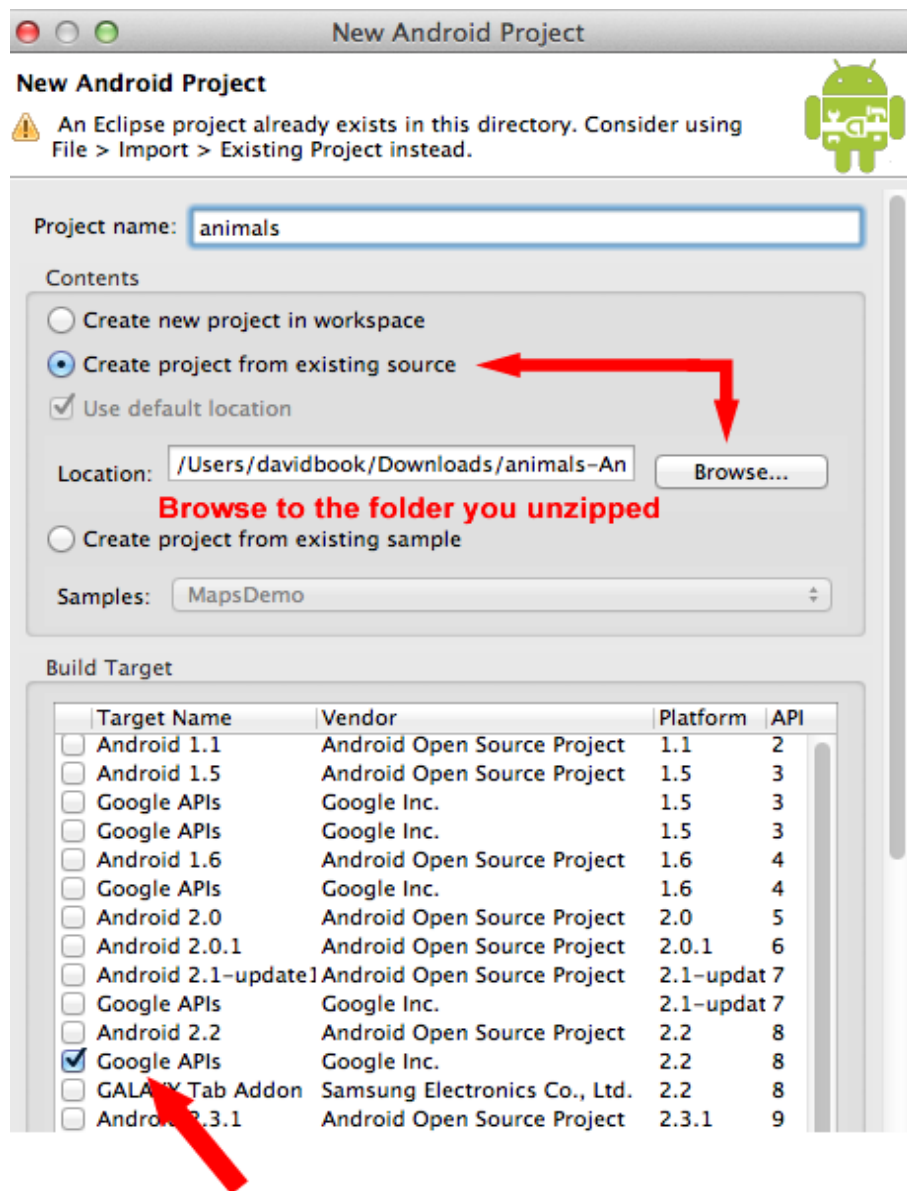
When Eclipse is finished launching, use the top-most toolbar and select File > New > Android Project.

Note: If you did not install the Android SDK in Eclipse you will not see the New > Android Project option. This means you will need to finish installing the Android SDK components before continuing.



When you choose New > Android Project a dialogue (pop-up-window) will open. This window needs to know three things. See the image on the next page.

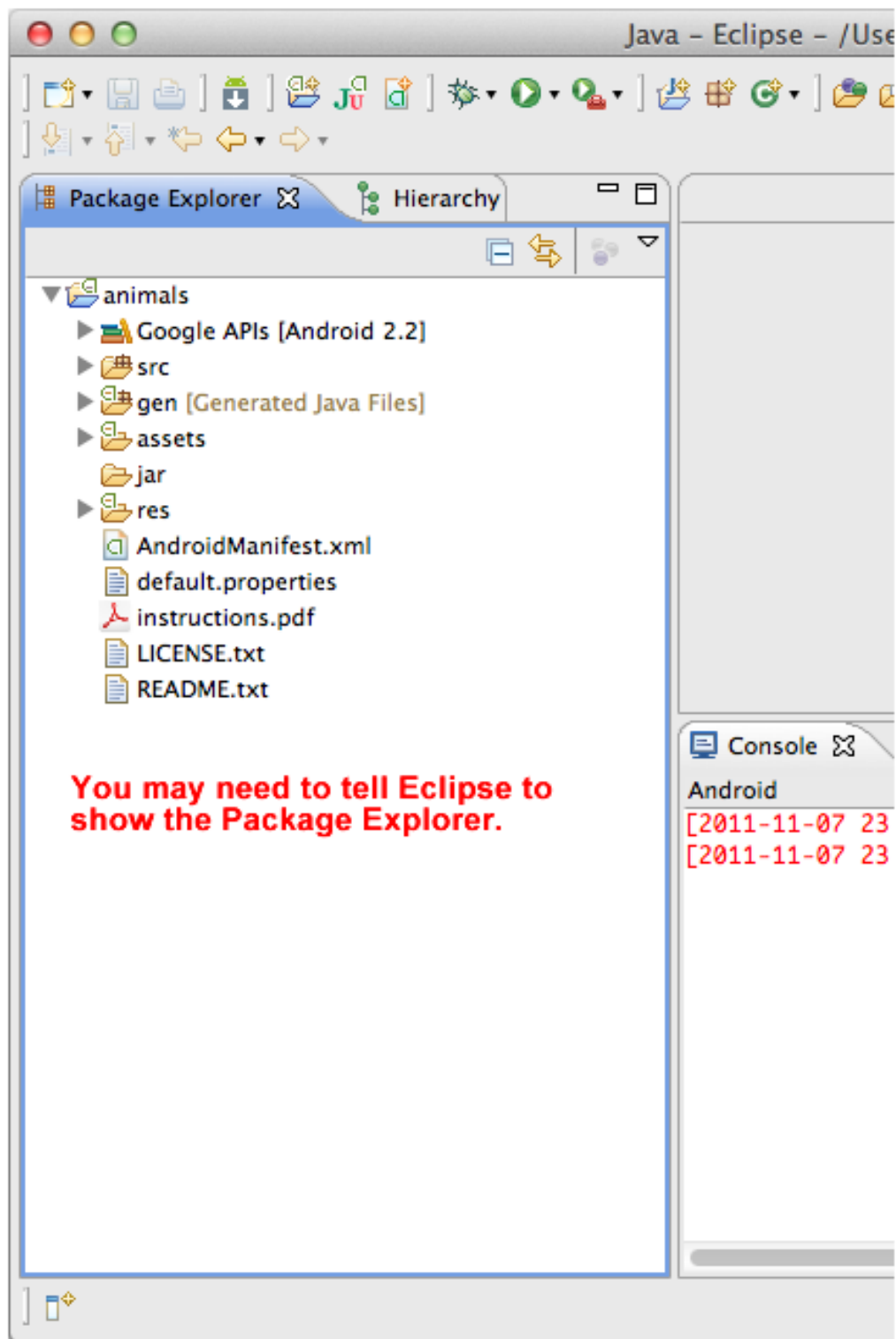
- 1) The name of your new project. Enter a project name. Do not use any special characters or spaces in the project name you enter.
- 2) The location of the source-code. Use the "Create Project from Existing Source" option. After choosing this, use the Browse button to navigate to the folder you downloaded and unzipped in step 1. Select the name of your folder in Mac's Finder app or Windows Explorer. Do not select any files inside the folder, select the folder itself.
- 3) The Build Target. You should see a long list of build targets available in the New Android Project window. Google API's 2.2 should be pre-selected. **This is sometimes NOT THE CASE. If another platform is selected, change it to Google API's 2.2. This is different than Android 2.2.** If you do not see the Google API's 2.2 choice you have not installed all the necessary components in Eclipse and you'll need to do that before continuing.



**Google API's 2.2 NOT Android 2.2. There is a difference**

The image above does not show the bottom part of the New Android Project window. The bottom part of this window is pre-filled with some values you don't need to change. When you're done with 1, 2, 3 on this window click Finish to dismiss the window. At this point you'll see the name of your new project in the Package Explorer. If you do not see the Package Explorer it's because Eclipse is not showing it. In this case you'll need to use Eclipse's menu options to display the Package Explorer.

Your project will look like the graphic on the next page.

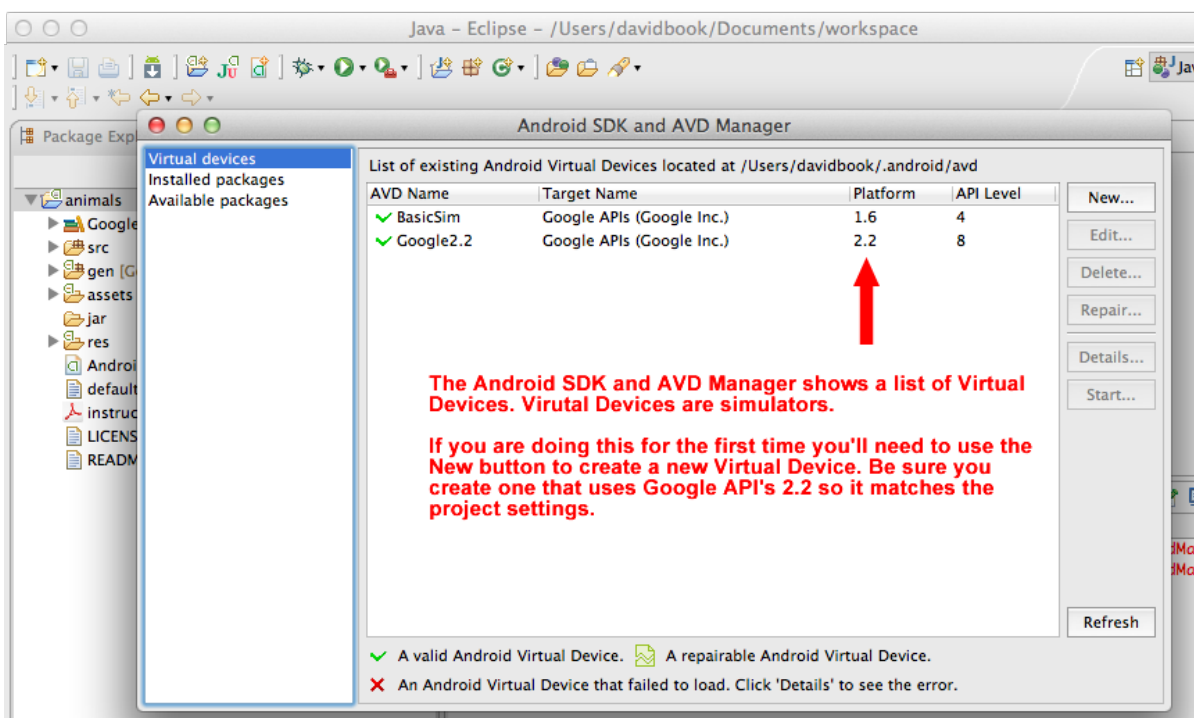


# 3

## Setting up Virtual Devices (simulators)

Before the project can be compiled and ran you'll need to determine what device to run it on. You can run it on an actual device or an Android Virtual Device (simulator). An actual device works best but virtual devices work well too. There are some limitations on how Android Virtual Devices work (such as not playing videos) but generally they are OK for testing.

The image below shows the Android SDK and AVD Manager window. This is the window you use to setup virtual devices and add or remove Android SDK components, packages, and software. Android developers spend lots of time in this window.



If you're running on an actual device you will not need to setup any virtual devices in the Android SDK and AVD Manager but make sure the actual device is connected to the computer with a USB cable. And, also, make sure the device is setup to "allow debugging" in it's settings panel.

# 4

## Run As Android Project

With your device connected, or your virtual device running, select the projects name in the Package Explorer (single click). Then, choose Run As > Android Project from the main menu.

The Run As > Android Project menu option is tough to find

sometimes. You may have a shortcut button in your toolbar, or you may have to use Eclipse's menu options. Either way you need to make sure a device is connected (or a virtual device is running) and that the project name is selected in the Package Explorer before you click Run as Android Project.

## **Errors:**

It's possible that Eclipse will report lots and lots of errors when you try to Run as an Android project. There are countless reasons for this but it almost always boils down to one of two things...

a) Your machine is not setup to compile Android Projects. You may think it is but hundreds of errors is a guarantee that it's not. Make sure you have the appropriate Android libraries installed in Eclipse. Use the Android SDK and Virtual Device Manager to find and install the necessary libraries.

b) The project you setup is not set to use the correct Google API's 2.2 libraries. This is very common, Android libraries are different, this project expects you to use Google API's version 2.2.

## **Images and Other Assets:**

Some plugins provide all the necessary images and files (audio, video, documents) required to make the plugin work, others do not. In some cases, you'll need to provide your own. If you need to provide images or other files necessary for a plugin, you need to add them to the appropriate project folder AFTER downloading your source-code from the online control panel.

If you need to add files to your project, here's where they go:

### **/assets/BT\_Docs**

Put .HTML, .PDF, .DOC, and other document based assets in this folder.

### **/assets/BT\_Video**

Put video files in this folder.

### **/assets/BT\_Audio**

Put audio files in this folder.

Video and Audio format is Android is a confusing point for lots of people. In most cases, Video is H.264 compressed and Audio is .mp3. There are lots of other supported audio / video formats. If you're unsure what format to use, consult the plugins readme.txt file or contact the plugin author.

Android Media Formats Reference:

<http://developer.android.com/guide/appendix/media-formats.html>