

Mobile Communications: Self-Study (1)

2013.4.19

Jiro Katto

Dept of Computer Science and Engineering,

Submit a report in Word or PDF to CourseN@vi “Mobile Communications” “第 2 回講義 / Self-Study”.

Getting Started: Android SDK

1. Go to <http://developer.android.com/sdk/index.html>.
2. Download Android SDK with ADT Bundle (and Eclipse) to your platform (Windows, Mac or Linux).
3. Follow “Setting Up the ADT Bundle”, “Setting Up and Existing IDE” and “Exploring the SDK”.
4. Go to <http://developer.android.com/training/index.html>.
5. Follow “Getting Started” and “Building Your First App”.
6. Run Eclipse, build an Android project, and build an AVD (Android Virtual Device) emulator.
7. Run your first app on the emulator (or you can try on your actual Android device if you have).
8. Capture the image of your first app by Cntl+Alt+PrintScreen, and attach the image to your report.

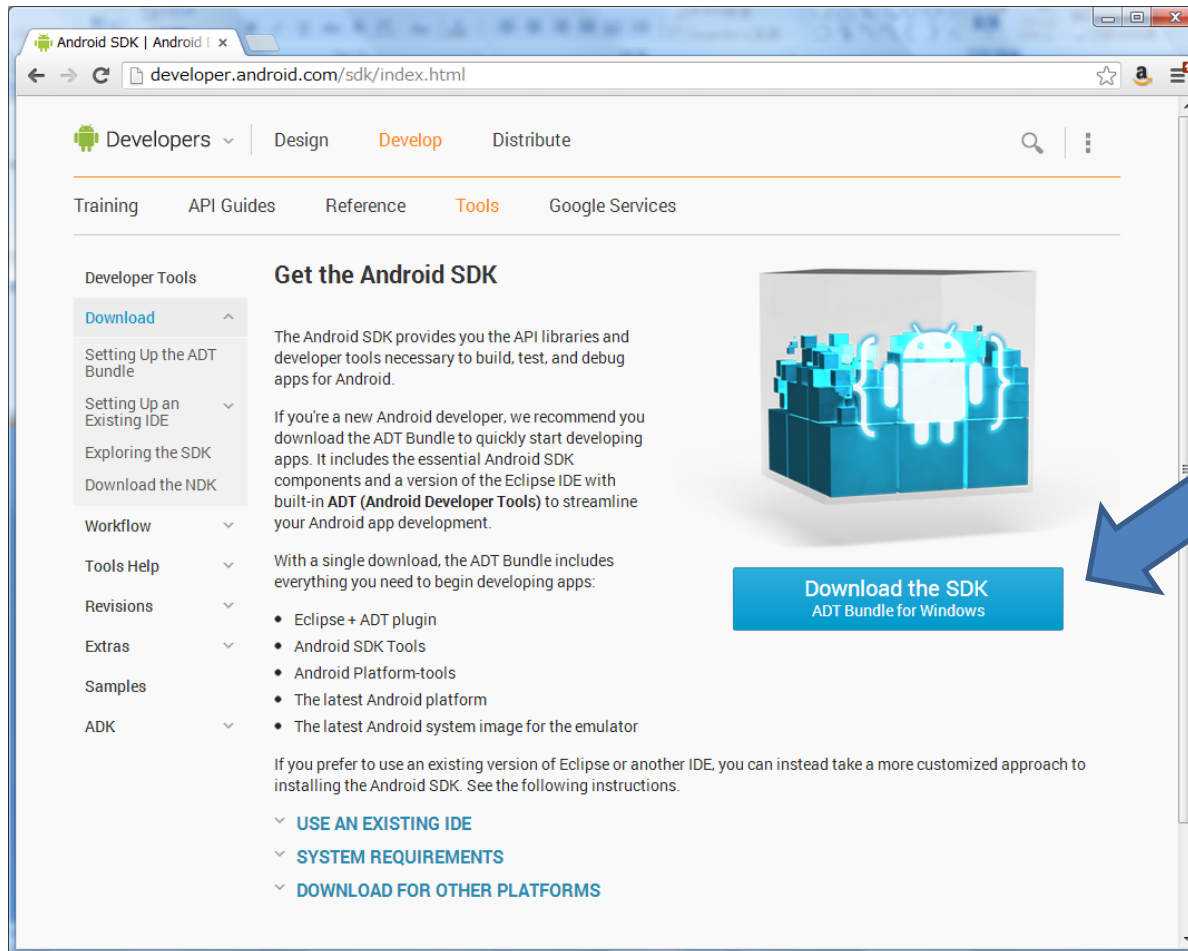
Report structure:

- (1) Write your impression about Android SDK and Android applications.
- (2) Paste the captured image into the document.

Self Study: Apr 19, 2013

Android SDK: Getting Started

Get the Android SDK



The screenshot shows a web browser window with the URL `developer.android.com/sdk/index.html`. The page features a navigation menu with 'Developers', 'Design', 'Develop', and 'Distribute'. Below this, there are links for 'Training', 'API Guides', 'Reference', 'Tools', and 'Google Services'. The main content area is titled 'Get the Android SDK' and includes a sidebar with a 'Download' menu. The main text describes the Android SDK and provides a list of components included in the ADT Bundle. A large blue button labeled 'Download the SDK' is prominently displayed, with a blue arrow pointing to it from the right side of the page.

Android SDK | Android | x

developer.android.com/sdk/index.html

Developers | Design | **Develop** | Distribute

Training | API Guides | Reference | **Tools** | Google Services

Developer Tools

- Download
- Setting Up the ADT Bundle
- Setting Up an Existing IDE
- Exploring the SDK
- Download the NDK

Workflow

Tools Help

Revisions

Extras

Samples

ADK

Get the Android SDK

The Android SDK provides you the API libraries and developer tools necessary to build, test, and debug apps for Android.

If you're a new Android developer, we recommend you download the ADT Bundle to quickly start developing apps. It includes the essential Android SDK components and a version of the Eclipse IDE with built-in **ADT (Android Developer Tools)** to streamline your Android app development.

With a single download, the ADT Bundle includes everything you need to begin developing apps:

- Eclipse + ADT plugin
- Android SDK Tools
- Android Platform-tools
- The latest Android platform
- The latest Android system image for the emulator

If you prefer to use an existing version of Eclipse or another IDE, you can instead take a more customized approach to installing the Android SDK. See the following instructions.

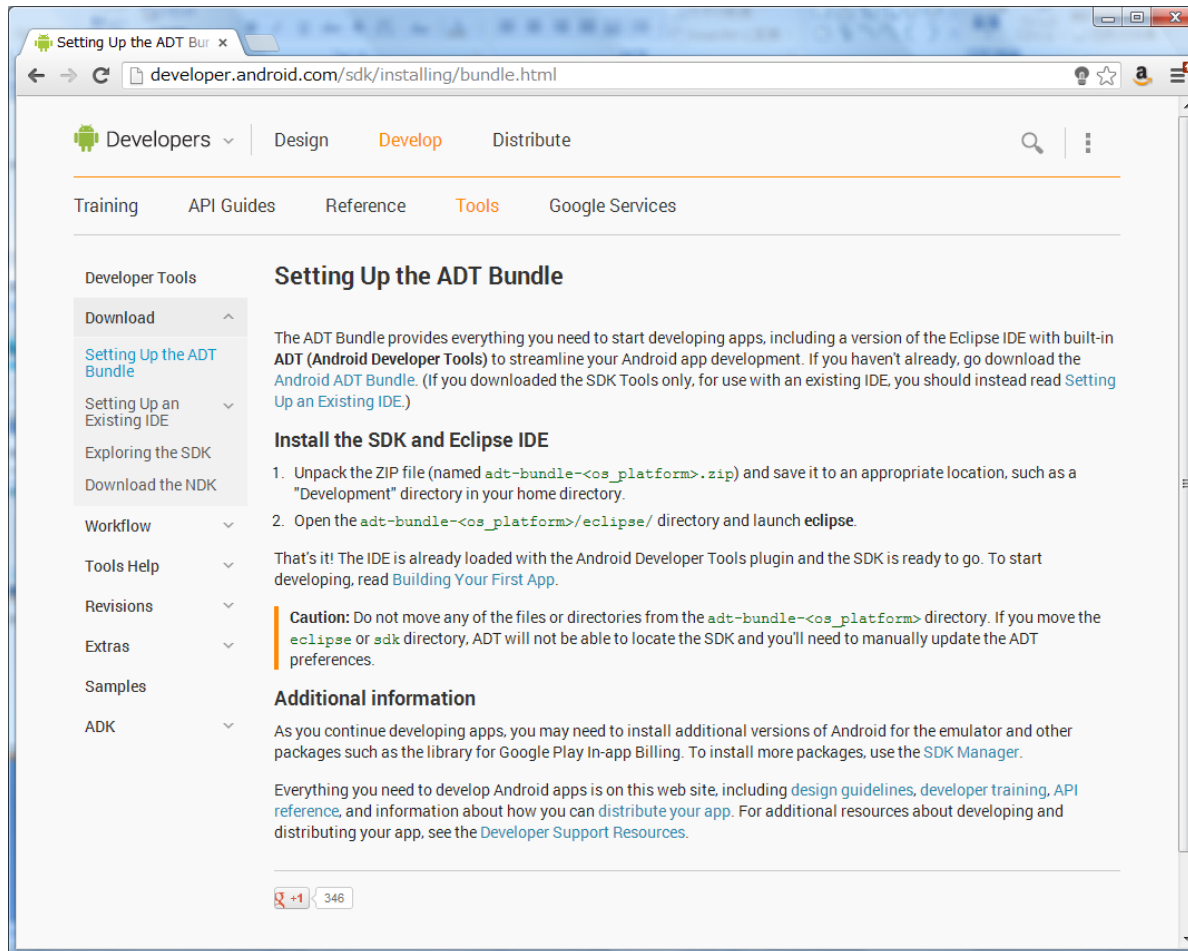
- ▼ [USE AN EXISTING IDE](#)
- ▼ [SYSTEM REQUIREMENTS](#)
- ▼ [DOWNLOAD FOR OTHER PLATFORMS](#)

Download the SDK
ADT Bundle for Windows

download

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

Setting Up the ADT Bundle



The screenshot shows a web browser window with the URL `developer.android.com/sdk/installing/bundle.html`. The page title is "Setting Up the ADT Bundle". The navigation bar includes "Developers", "Design", "Develop", and "Distribute". Below the navigation bar, there are links for "Training", "API Guides", "Reference", "Tools", and "Google Services". The main content area is titled "Setting Up the ADT Bundle" and contains the following text:

The ADT Bundle provides everything you need to start developing apps, including a version of the Eclipse IDE with built-in **ADT (Android Developer Tools)** to streamline your Android app development. If you haven't already, go download the [Android ADT Bundle](#). (If you downloaded the SDK Tools only, for use with an existing IDE, you should instead read [Setting Up an Existing IDE](#).)

Install the SDK and Eclipse IDE

1. Unpack the ZIP file (named `adt-bundle-<os_platform>.zip`) and save it to an appropriate location, such as a "Development" directory in your home directory.
2. Open the `adt-bundle-<os_platform>/eclipse/` directory and launch **eclipse**.

That's it! The IDE is already loaded with the Android Developer Tools plugin and the SDK is ready to go. To start developing, read [Building Your First App](#).

Caution: Do not move any of the files or directories from the `adt-bundle-<os_platform>` directory. If you move the `eclipse` or `sdk` directory, ADT will not be able to locate the SDK and you'll need to manually update the ADT preferences.

Additional information

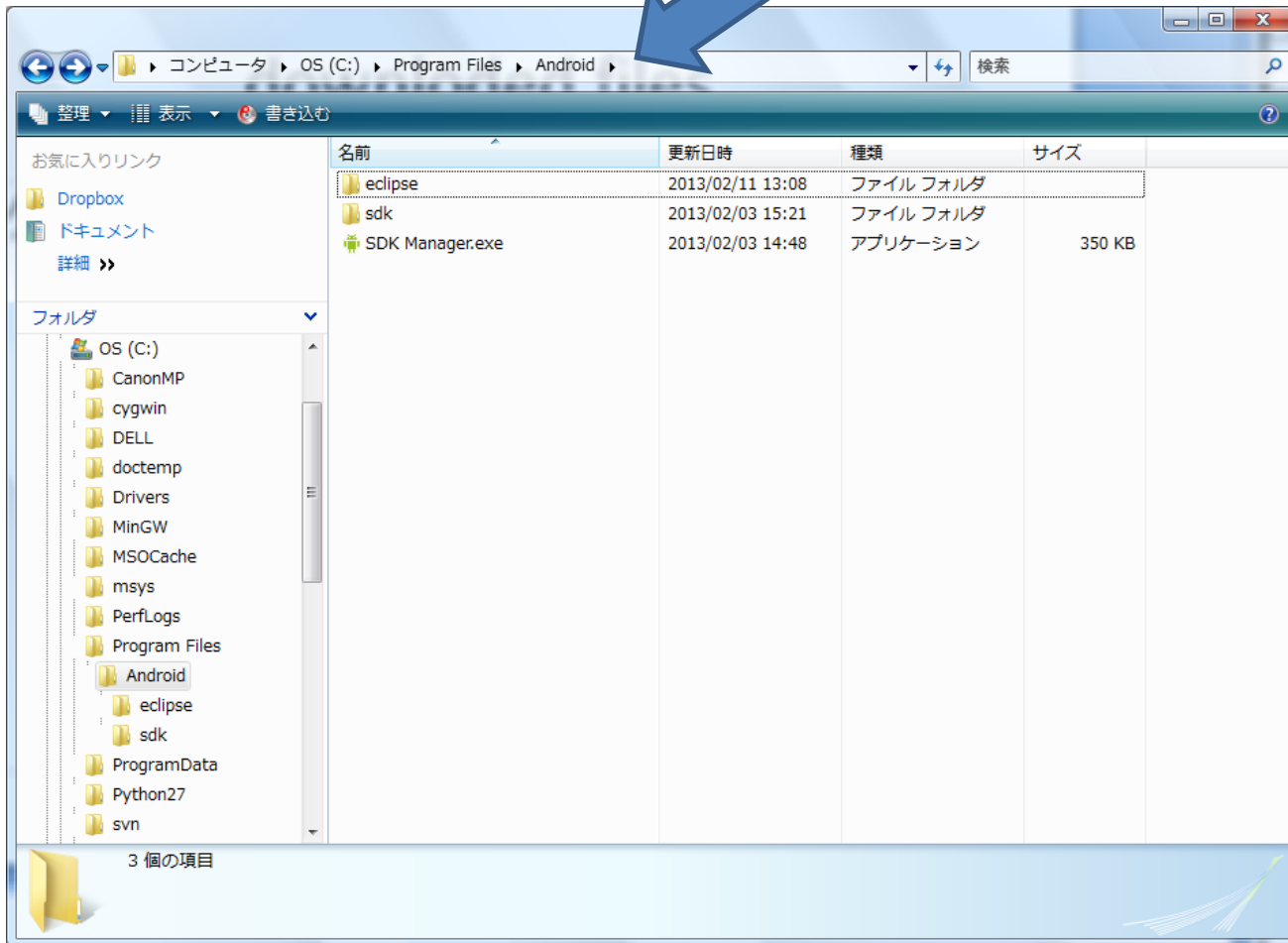
As you continue developing apps, you may need to install additional versions of Android for the emulator and other packages such as the library for Google Play In-app Billing. To install more packages, use the [SDK Manager](#).

Everything you need to develop Android apps is on this web site, including [design guidelines](#), [developer training](#), [API reference](#), and information about how you can [distribute your app](#). For additional resources about developing and distributing your app, see the [Developer Support Resources](#).

At the bottom of the page, there is a social sharing icon and a count of 346 shares.

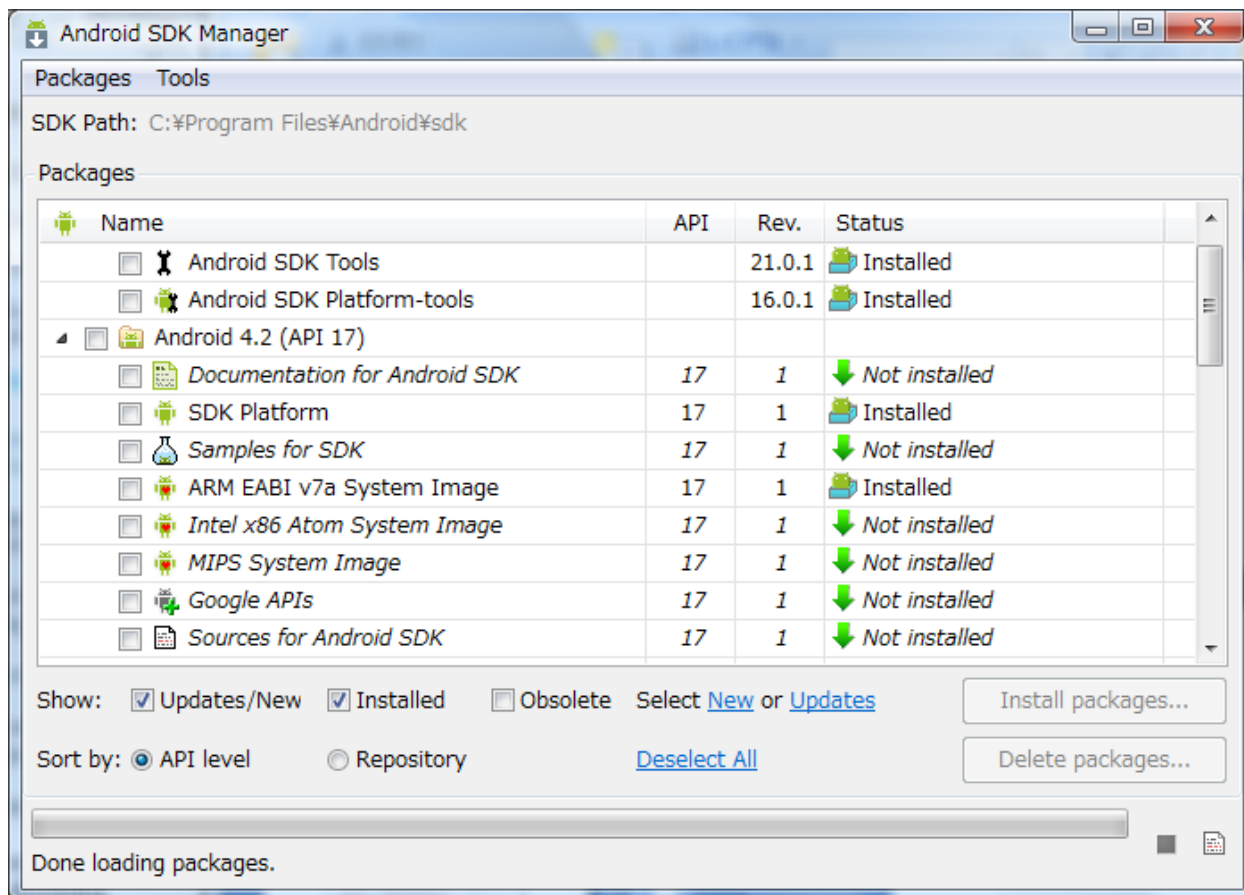
downloaded files

anywhere you like

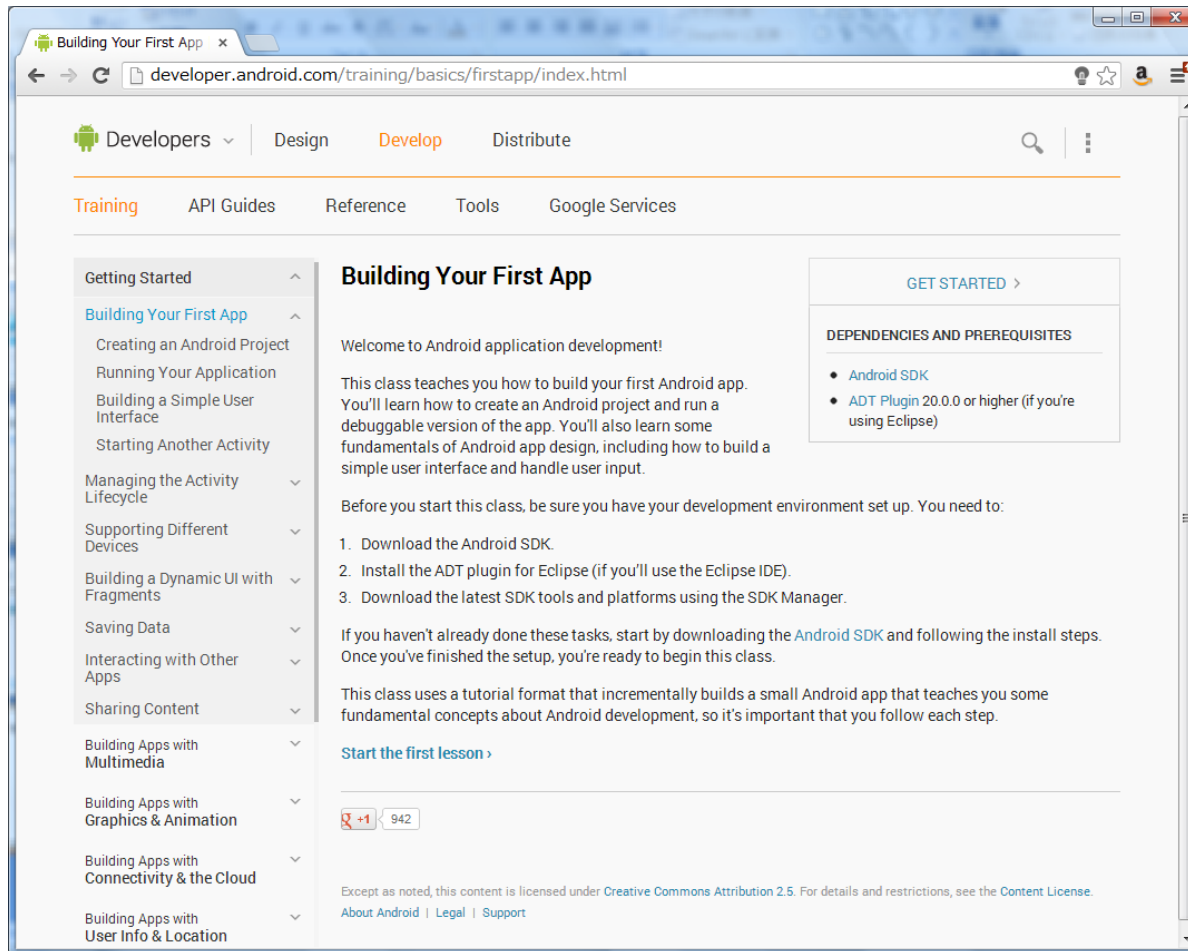


SDK Manager

- Install and update



Building Your First App



The screenshot shows a web browser window displaying the Android Developer website. The page title is "Building Your First App" and the URL is "developer.android.com/training/basics/firstapp/index.html". The navigation menu includes "Developers", "Design", "Develop", and "Distribute". The "Training" section is active, with sub-links for "API Guides", "Reference", "Tools", and "Google Services". The main content area is titled "Building Your First App" and includes a "GET STARTED >" button. The page content includes a welcome message, a list of prerequisites (Android SDK and ADT Plugin), and a list of steps to get started. A sidebar on the left contains a table of contents for the tutorial series.

Building Your First App

Welcome to Android application development!

This class teaches you how to build your first Android app. You'll learn how to create an Android project and run a debuggable version of the app. You'll also learn some fundamentals of Android app design, including how to build a simple user interface and handle user input.

Before you start this class, be sure you have your development environment set up. You need to:

1. Download the Android SDK.
2. Install the ADT plugin for Eclipse (if you'll use the Eclipse IDE).
3. Download the latest SDK tools and platforms using the SDK Manager.

If you haven't already done these tasks, start by downloading the [Android SDK](#) and following the install steps. Once you've finished the setup, you're ready to begin this class.

This class uses a tutorial format that incrementally builds a small Android app that teaches you some fundamental concepts about Android development, so it's important that you follow each step.

[Start the first lesson >](#)

GET STARTED >

DEPENDENCIES AND PREREQUISITES

- [Android SDK](#)
- [ADT Plugin 20.0.0](#) or higher (if you're using Eclipse)

+1 942

Except as noted, this content is licensed under [Creative Commons Attribution 2.5](#). For details and restrictions, see the [Content License](#).

[About Android](#) | [Legal](#) | [Support](#)

Getting Started

Building Your First App

Creating an Android Project

Running Your Application

Building a Simple User Interface

Starting Another Activity

Managing the Activity Lifecycle

Supporting Different Devices

Building a Dynamic UI with Fragments

Saving Data

Interacting with Other Apps

Sharing Content

Building Apps with Multimedia

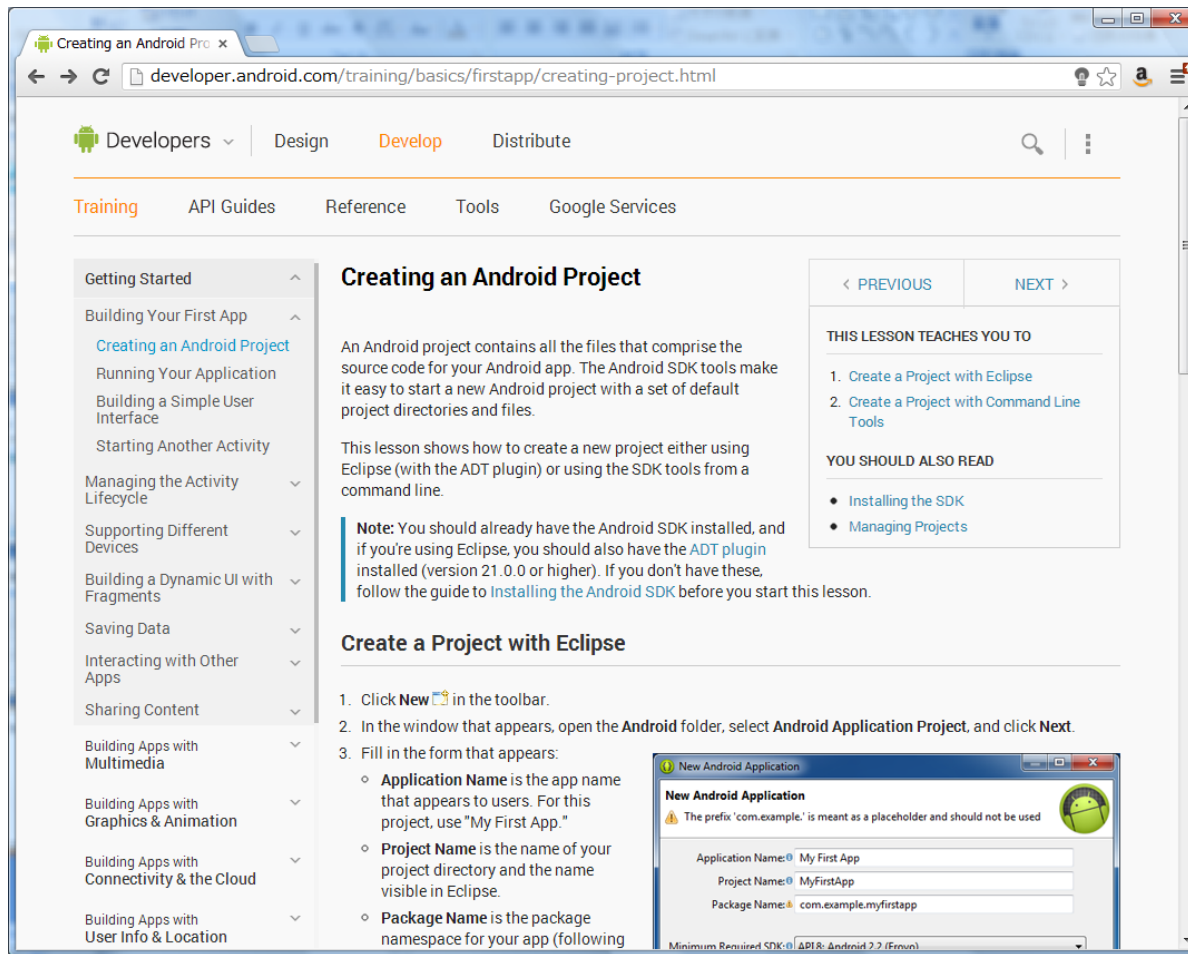
Building Apps with Graphics & Animation

Building Apps with Connectivity & the Cloud

Building Apps with User Info & Location

<http://developer.android.com/training/basics/firstapp/index.html>

Creating an Android Project



Creating an Android Project

An Android project contains all the files that comprise the source code for your Android app. The Android SDK tools make it easy to start a new Android project with a set of default project directories and files.

This lesson shows how to create a new project either using Eclipse (with the ADT plugin) or using the SDK tools from a command line.

Note: You should already have the Android SDK installed, and if you're using Eclipse, you should also have the [ADT plugin](#) installed (version 21.0.0 or higher). If you don't have these, follow the guide to [Installing the Android SDK](#) before you start this lesson.

Create a Project with Eclipse

1. Click **New** in the toolbar.
2. In the window that appears, open the **Android** folder, select **Android Application Project**, and click **Next**.
3. Fill in the form that appears:
 - **Application Name** is the app name that appears to users. For this project, use "My First App."
 - **Project Name** is the name of your project directory and the name visible in Eclipse.
 - **Package Name** is the package namespace for your app (following

THIS LESSON TEACHES YOU TO

1. [Create a Project with Eclipse](#)
2. [Create a Project with Command Line Tools](#)

YOU SHOULD ALSO READ

- [Installing the SDK](#)
- [Managing Projects](#)

New Android Application

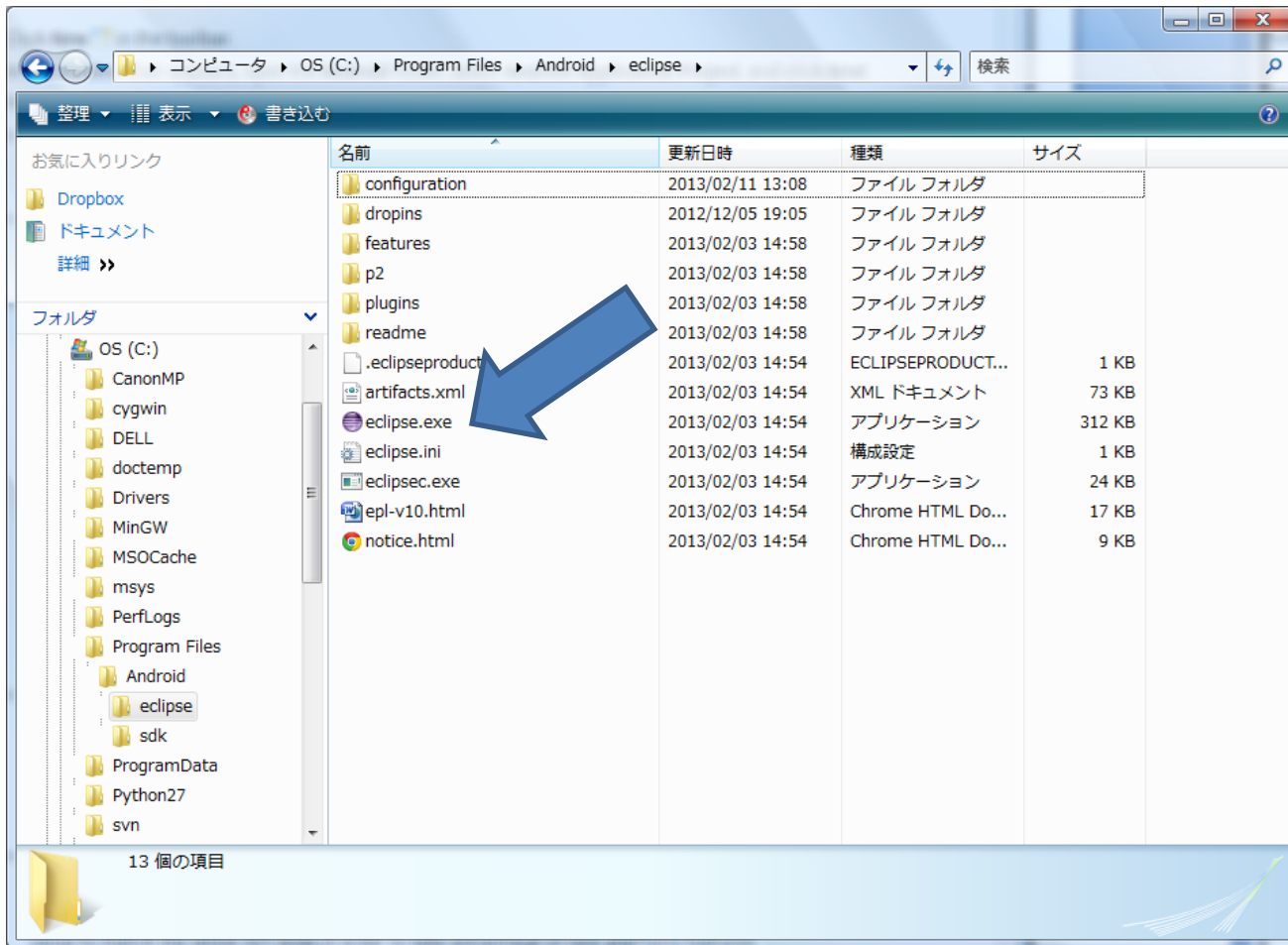
The prefix 'com.example.' is meant as a placeholder and should not be used

Application Name: My First App
Project Name: MyFirstApp
Package Name: com.example.myfirstapp

Minimum Required SDK: API 8: Android 2.2 (Froyo)

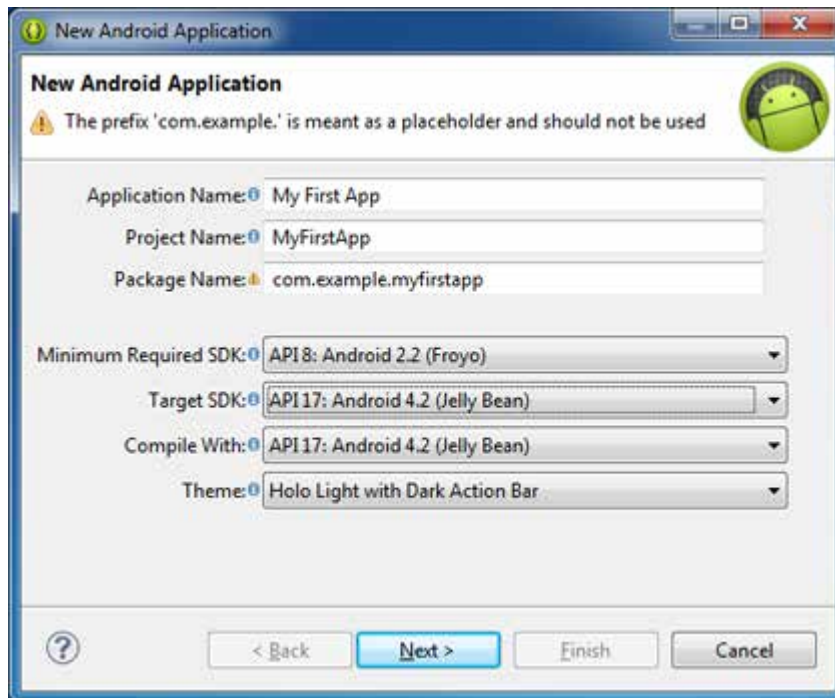
<http://developer.android.com/training/basics/firstapp/creating-project.html>

eclipse



Create a Project with Eclipse

- File -> New -> Project -> Android Application Project



Fill Application Name, then

Next

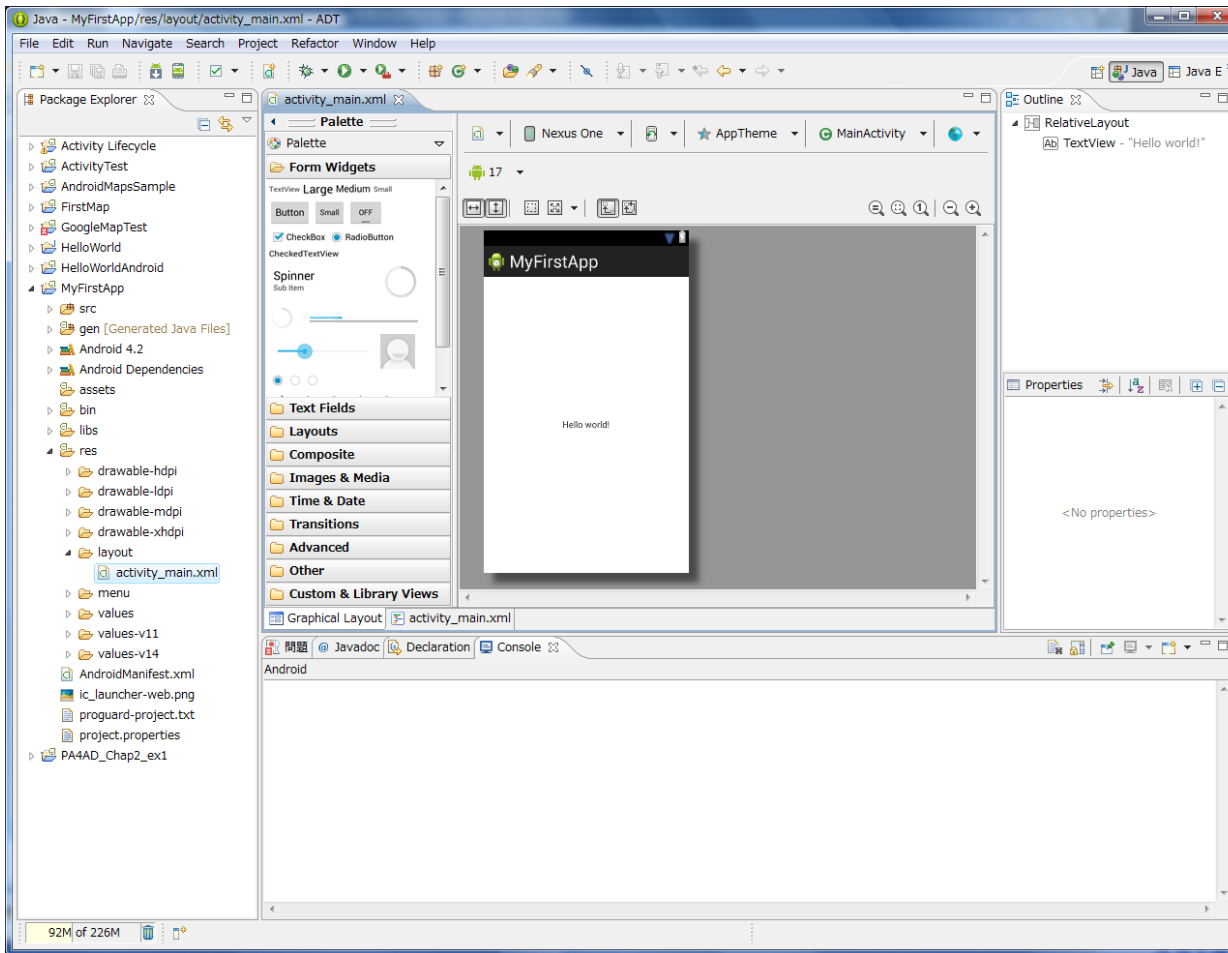
Next

Next

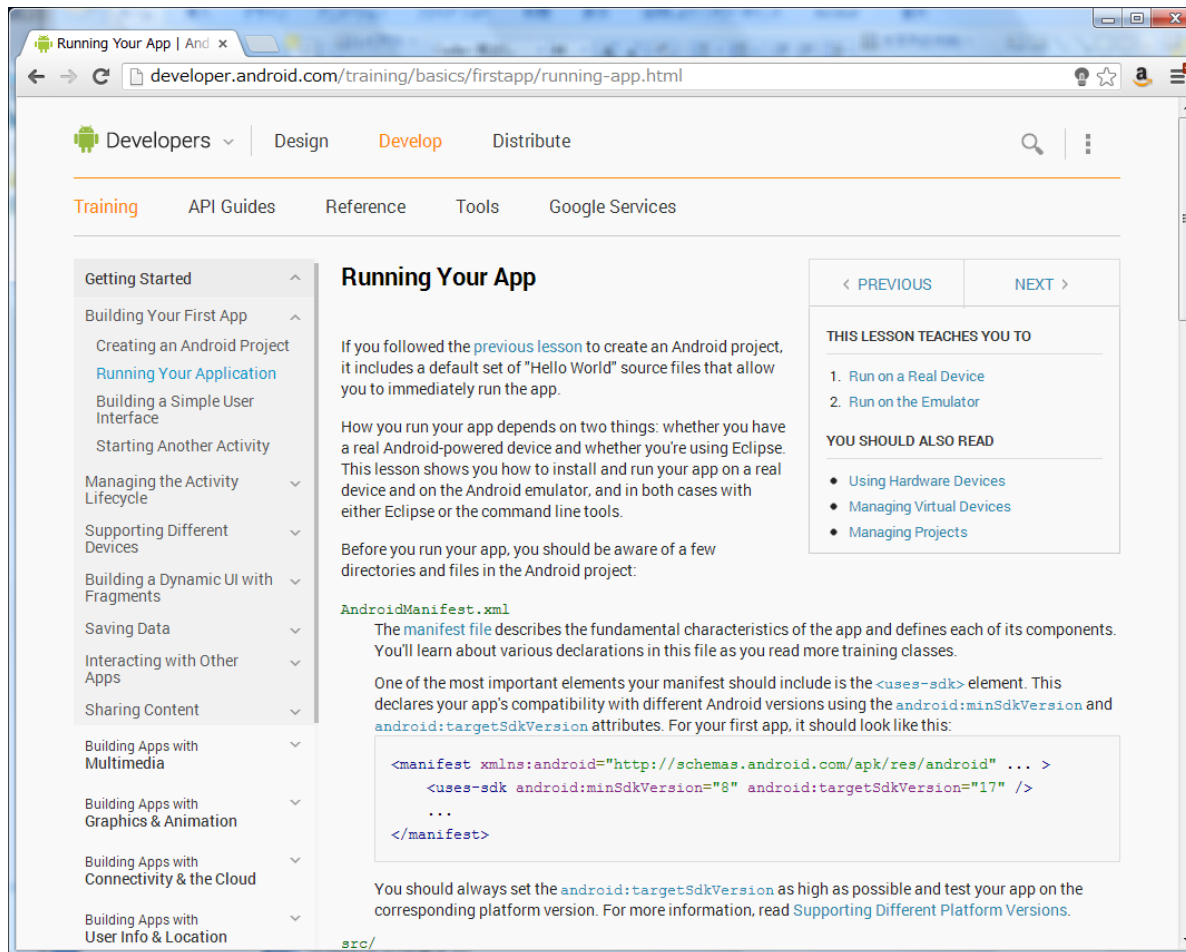
...

Finish

MyFisrtApp



Running Your App



The screenshot shows a web browser window displaying the Android developer website. The page title is "Running Your App" and the URL is <http://developer.android.com/training/basics/firstapp/running-app.html>. The page features a navigation menu with "Developers", "Design", "Develop", and "Distribute" options. Below the menu, there are tabs for "Training", "API Guides", "Reference", "Tools", and "Google Services". The main content area is titled "Running Your App" and includes a sidebar with a list of training topics. The main text explains that the lesson covers running an app on a real device or emulator, depending on the tools used. It also discusses the importance of the `AndroidManifest.xml` file and the `uses-sdk` element, providing an example of the XML code. The page concludes with a recommendation to set the `targetSdkVersion` as high as possible and to test the app on the corresponding platform version.

Running Your App

If you followed the [previous lesson](#) to create an Android project, it includes a default set of "Hello World" source files that allow you to immediately run the app.

How you run your app depends on two things: whether you have a real Android-powered device and whether you're using Eclipse. This lesson shows you how to install and run your app on a real device and on the Android emulator, and in both cases with either Eclipse or the command line tools.

Before you run your app, you should be aware of a few directories and files in the Android project:

AndroidManifest.xml

The **manifest file** describes the fundamental characteristics of the app and defines each of its components. You'll learn about various declarations in this file as you read more training classes.

One of the most important elements your manifest should include is the `<uses-sdk>` element. This declares your app's compatibility with different Android versions using the `android:minSdkVersion` and `android:targetSdkVersion` attributes. For your first app, it should look like this:

```
<manifest xmlns:android="http://schemas.android.com/apk/res/android" ... >
  <uses-sdk android:minSdkVersion="8" android:targetSdkVersion="17" />
  ...
</manifest>
```

You should always set the `android:targetSdkVersion` as high as possible and test your app on the corresponding platform version. For more information, read [Supporting Different Platform Versions](#).

<http://developer.android.com/training/basics/firstapp/running-app.html>

Files and Directories

- `AndroidManifest.xml`
 - declarations of the application
- `src/`
 - main source files of the application
- `res/`
 - resources of the application

Run on a Real Device

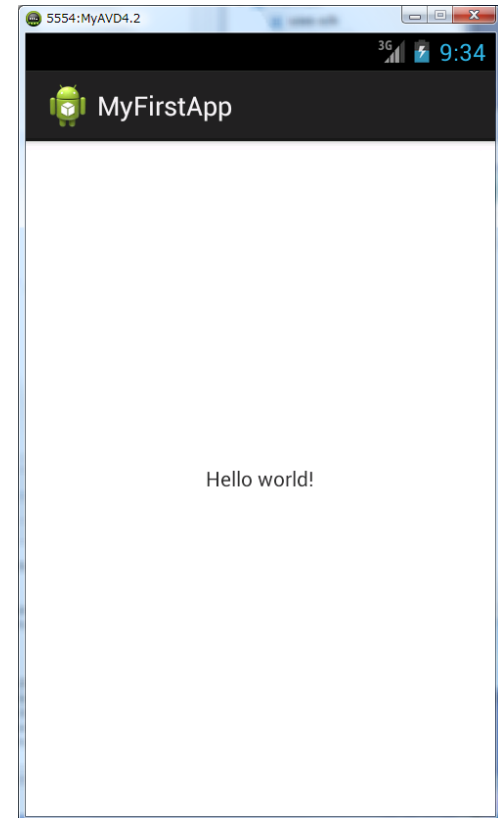
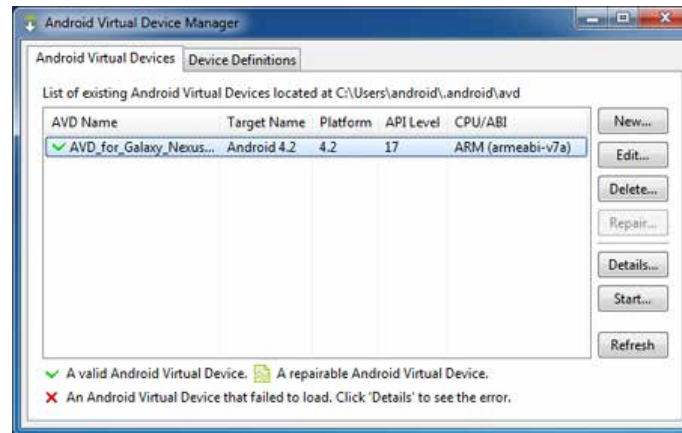
- Enable “USB Debugging” mode of your Android device
- Attach your Android device to your PC
- Run -> Run
- Select actual Android device

If you get error messages like “Can't bind to local 86xx for debugger”, edit `c:\windows\system32\drivers\etc\hosts` by administrator mode.

<http://developer.android.com/training/basics/firstapp/running-app.html>

Run on the Emulator

- Create Android Virtual Device (AVD)
 - Android Virtual Device Manager
 - Set parameters
 - Create AVD
- Run -> Run
- Select AVD



If your AVD doesn't work, try Google about parameter setting.