

Android Programming

Developing End User Application

Riya Jacob K

Dept of BCA

2020-2021

Developing an End-User Application with the Android SDK

- Android emulator
- Android foundational components
- UI programming
- Services
- Media
- Telephony
- Animation etc.

Android Emulator

- The Android SDK ships with an Eclipse plug-in called Android Development Tools (ADT).
- This Integrated Development Environment (IDE) tool is used for developing, debugging, and testing our Java applications.
- Support an emulator that can use to run, debug, and test our applications.
- The Android emulator accomplishes its work through an open source “processor emulator” technology called QEMU, developed by Fabrice Bellard (<http://wiki.qemu.org/Index.html>).

- With the Android emulator, the processor is based on Advanced RISC Machine (ARM).
- ARM is a 32-bit microprocessor architecture based on Reduced Instruction Set Computing (RISC), in which design simplicity and speed is achieved through a reduced number of instructions in an instruction set.
- The emulator runs the Android version of Linux on this simulated processor.
- ARM is widely used in handhelds and other embedded electronics where lower power consumption is important.
- Much of the mobile market uses processors based on this architecture

The Android UI

- Android uses a UI framework that resembles other desktop-based, full-featured UI frameworks, it's more modern and more asynchronous in nature.
- The Android UI is essentially a fourth-generation UI framework
 - the traditional C-based Microsoft Windows API the first generation
 - and the C++-based Microsoft Foundation Classes (MFC) the second generation.
 - The Java-based Swing UI framework would be the third generation,
- The Android UI, JavaFX, Microsoft Silverlight, and Mozilla XML User Interface Language(XUL) fall under this new type of fourth-generation UI framework
- The UI is declarative and independently themed.

- Programming in the Android UI involves declaring the interface in XML files. Then load these XML view definitions as windows in our UI application.
- Even menus in our Android application are loaded from XML files.
- Screens or windows in Android are often referred to as **activities**, which comprise multiple views that a user needs in order to accomplish a logical unit of action.
- **Views** are Android's basic UI building blocks, and further combine them to form composite views called **view groups**.
- Views internally use the familiar concepts of canvases, painting, and user interaction.
- An activity hosting these composite views, which include views and view groups, is the logical replaceable UI component in Android.
- Android 3.0 introduced a new UI concept called **fragments** to allow developers to chunk views and functionality for display on tablets.

- Tablets provide enough screen space for multipane activities, and fragments provide the abstraction for the panes.
- One of the Android framework's key concepts is the life cycle management of activity windows.
- Protocols are put in place so that Android can manage state as users hide, restore, stop, and close activity windows.

The Android Foundational Components

- A component in Android is a piece of code that has a well defined life cycle.
- An *intent* is an intra- and interprocess mechanism to invoke components in Android. Intent is used to invoke components. It is mainly used to:
 - Start the service
 - Launch an activity
 - Display a web page
 - Display a list of contacts
 - Broadcast a message
 - Dial a phone call
- **For example**, you may write the following code to view the webpage.

```
Intent intent=new Intent(Intent.ACTION_VIEW);
intent.setData(Uri.parse("http://www.javatpoint.com"));
startActivity(intent);
```
- An *activity* representing a window in an Android application is a component.
- A *service* that runs in an Android process and serves other clients is a component.
- A *broadcast receiver* that wakes up in response to an event is another example of a component in Android.
- A *content provider* supplies data from one application to others on request.

Additional Components

- *Fragments* - Fragments are like parts of activity. An activity can display one or more fragments on the screen at the same time.
- *Views* - A view is the UI element such as button, label, text field etc. Anything that you see is a view.
- *Layouts* - View hierarchies that control screen format and appearance of the views.
- *Resources* - External elements such as strings ,constants and draw able pictures.
- *AndroidManifest.xml* - It contains information about activities, content providers, permissions etc. It is like the web.xml file in Java EE.

Android Java Packages

Important packages that are included in the Android SDK:

android.app: Implements the Application model for Android. Primary classes include Application, representing the start and stop semantics, as well as a number of activity-related classes, fragments, controls, dialogs, alerts, and notifications.

android.app.admin: Provides the ability to control the device by folks such as enterprise administrators.

android.accounts: Provides classes to manage accounts such as Google, Facebook, and so on. The primary classes are AccountManager and Account.

android.animation: Hosts all the new property animation classes.

android.app.backup: Provides hooks for applications to back up and restore their data when folks switch their devices.

Setting Up Your Development Environment

- To build applications for Android, you need the Java SE Development Kit (JDK), the Android SDK, and a development environment.
- The Android SDK requires JDK 5 or JDK 6 and Eclipse 3.5 or higher
- The Android SDK is made up of two main parts: the tools and the packages.
- Tools are executables and supporting Environment files to help you develop applications.
- The packages are the files specific to a particular version of Android (called a platform) or a particular add-on to a platform.
- The platforms include Android 1.5 through 4.0.
- The add-ons include the Google Maps API, the Market License Validator, and even vendor-supplied ones such as Samsung's Galaxy Tab add-on. After you install the SDK, you then use one of the tools to download and set up the platforms and add-ons.

Setting Up Your Environment

- To build Android applications, you need to establish a development environment.
 - downloading JDK 6
 - the Eclipse IDE/Android studio,
 - the Android SDK (tools and packages), and
 - ADT

Java JDK

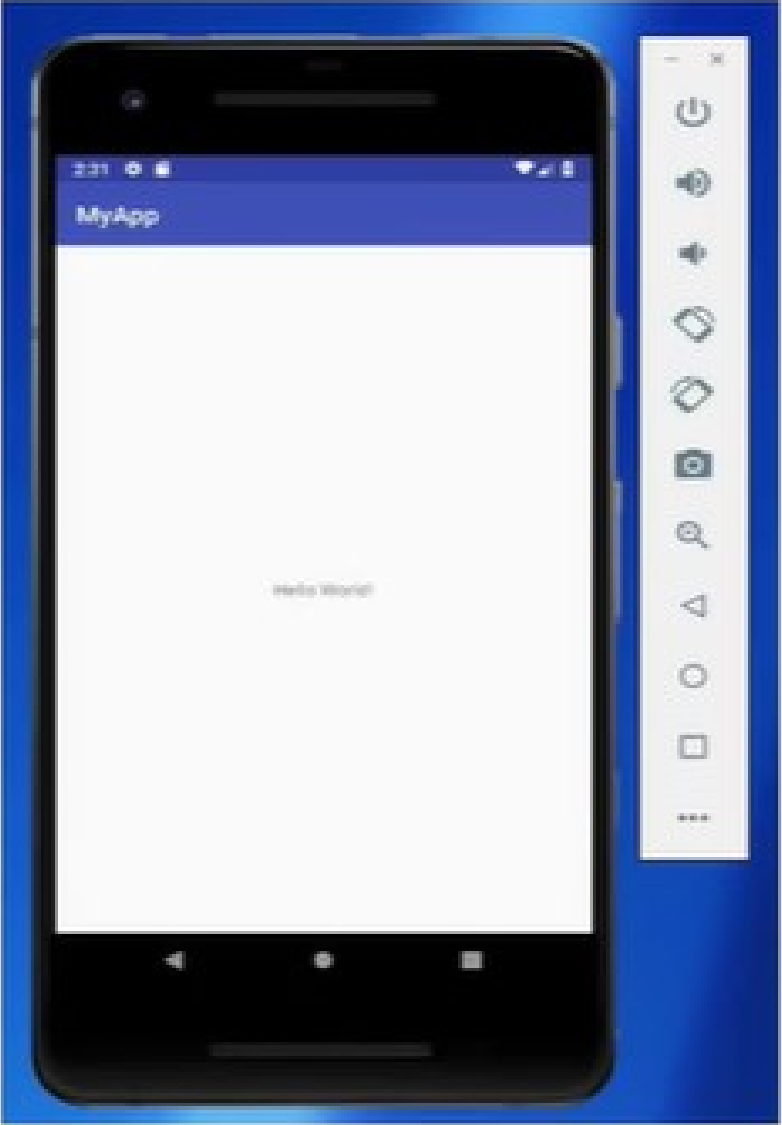
- The Java Development Kit (JDK) is one of three core technology packages used in Java programming, along with the JVM (Java Virtual Machine) and the JRE (Java Runtime Environment).
- [The JVM](#) is the Java platform component that executes programs.
- [The JRE](#) is the on-disk part of Java that creates the JVM.
- The JDK allows developers to create Java programs that can be executed and run by the JVM and JRE.

Android SDK

- The Android SDK (software development kit) is a set of development tools used to develop applications for Android platform.
- The Android SDK includes the following:
 - ✓ Required libraries
 - ✓ Debugger
 - ✓ An emulator
 - ✓ Relevant documentation for the Android application program interfaces (APIs)
 - ✓ Sample source code
 - ✓ Tutorials for the Android OS

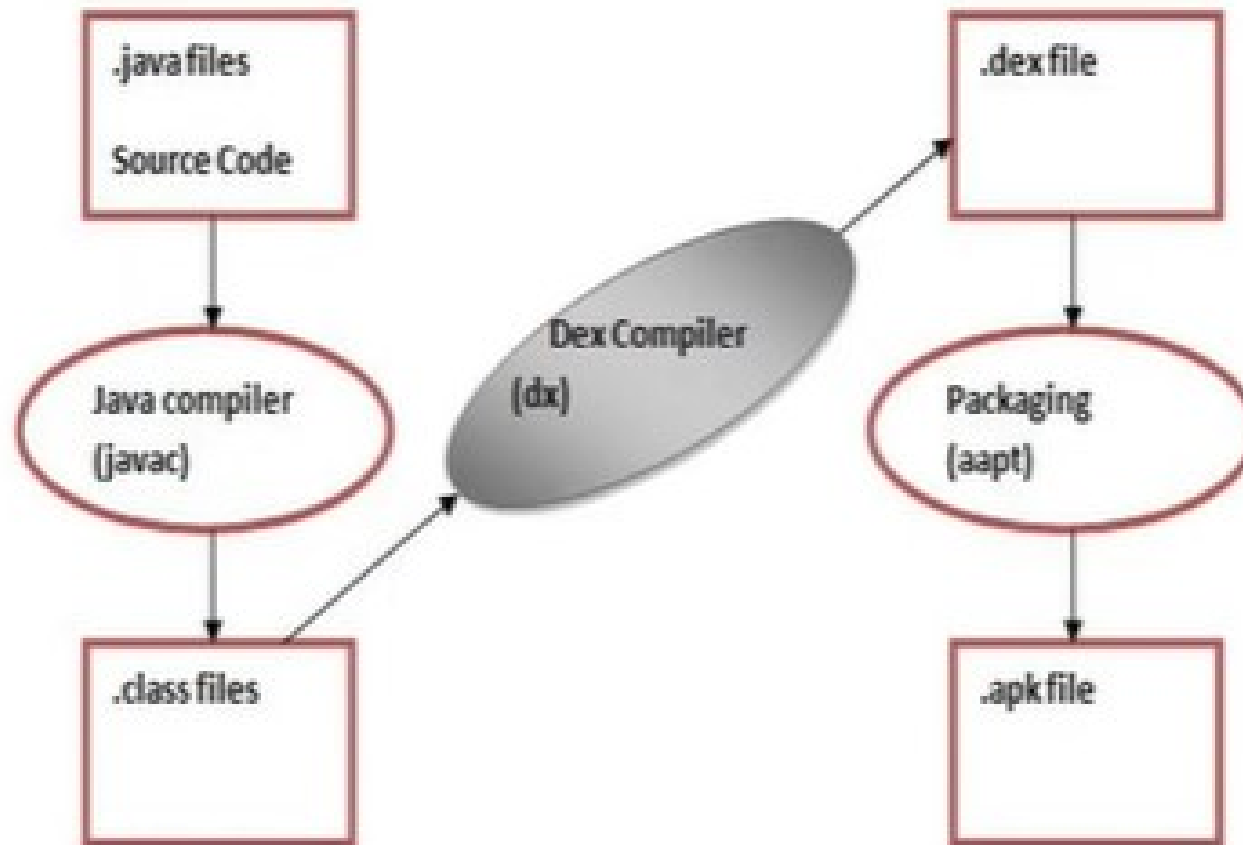
Android Virtual Devices (AVD) / Emulator

- An **Android Virtual Device (AVD)** is an **emulator configuration** that allows developers to test the application by simulating the real device capabilities.
- AVD manager enables an easy way of creating and managing the AVD with its graphical interface.
- We can create as many AVDs as we need, based on the types of device we want to test for.
- An AVD contains a **hardware profile, system image, storage area, and other properties.**



Dalvik Virtual Machine

- The **Dalvik Virtual Machine (DVM)** is an android virtual machine optimized for mobile devices.
- Dalvik is a name of a town in Iceland. The Dalvik VM was written by Dan Bornstein.
- The Dex compiler converts the class files into the .dex file that run on the Dalvik VM. Multiple class files are converted into one dex file.



- The **javac tool** compiles the java source file into the class file.
- The **dex tool** takes all the class files of your application and generates a single .dex file.
- The **Android Assets Packaging Tool (aapt)** handles the packaging process.

APK File

- APK stands for Android Package Kit (also Android Application Package) and is the file format that **Android uses to distribute and install apps.**
- An APK file contains all of a program's code (such as .dex files), resources, assets, certificates and manifest file

Android Studio

- The official IDE for **Android** platform.
 - Contains text editor, debugging tools and tools for running your applications
 - It has a strong editor tool for developing creative UI and emulators for different versions to test without having actual **Android** devices.
 - Android Studio was first announced at a Google I/O conference in 2013 and was released to the general public in 2014 after various beta versions

System Requirements

- Microsoft Windows 7/8/10 (64-bit) or Linux
- 4 GB RAM minimum, 8 GB RAM recommended
- Java