Making your first Android app

*** workshop ***

BY: *Jigyasa Grover*

foss asia

With over one billion devices activated, Android is an exciting space to make apps to help you communicate, organize, educate, entertain or anything else you're passionate about.

Clearly there's a demand for Android app development, and it's turning the platform with the lovable green mascot into more and more of a strong first choice rather than just a secondary option.

So if you've been intent on, thinking about, or simply playing with the idea of learning Android… Make Your First Android App right here !

The session aims at introducing budding developers with the basic concepts and terminology in Android Development.

It shall begin from scratch and discuss how to setup the environment and build a very own personal Android App with a Splash Screen for starters.

JIGYASA GROVER

- Student Developer, Google Summer of Code at FOSSASIA
- Mentor, Google Code In at FOSSASIA
- Platform Developer, Program Organizer & Python (PyGame) Mentor, Learn IT, Girl ! Edition 2
- Pharo 4.0, Smalltalk IDE Development Contributor and Content Writer
- Lead, Women Tech Makers New Delhi
- Director, Women Who Code Delhi
- Junior Student(3rd Year), Bachelor of Technology, Computer Science and Engineering, Delhi Technological University (formerly known as Delhi College of Engineering)

Email: grover.jigyasa1@gmail.com, jigyasa@womenwhocode.com
Twitter: @jigyasa_grover

What is Android ?

- Android is a mobile operating system (OS) ; Currently developed by Google & Based on the Linux kernel
- Designed primarily for touchscreen mobile devices such as smartphones and tablets.
- Android's user interface is mainly based on direct manipulation; Uses touch gestures such as swiping, tapping and pinching, to manipulate on-screen objects
- In addition to touchscreen devices, Google has further developed Android TV for televisions, Android Auto for cars, and Android Wear for wrist watches
- Variants of Android are also used on notebooks, game consoles, digital cameras, and other electronics.
- As of 2015, Android has the largest installed base of all operating systems.

REASONS TO LEARN ANDROID APP DEVELOPMENT

- Open source - Make what you want; Easy accessibility; No Licensing Fees
- Flexibility- No developmental restrictions; Many developmental possibilities.
- Penetration on Different devices & Growing user base - So many smartphones, tablets, glasses, and watches out there, coming from so many manufacturers;
- Multi-fold increase in Android mobile apps market; High demand for Android developers
- Cross-Platform Compatibility - Android platform offers developers chance for developmental research in conjunction with other platforms.
- Support for Android development - Plethora of support at their helm, from Google support system as well as online training providers, to rely on to learn more and influence their market-viability.

Getting Started

- In layman's terms:
  Programming - "*what you want your app to do*" - Java file(s)
  Designing layouts – "*how you want your app to look*" - XML file(s)

- A build tool is used to compile all the project files and package them together into a ***.apk*** file that can run on Android devices

- All files (Java, XML and others) of the app are managed by an *Integrated Development Environment (IDE)*. The IDE is the program used to edit the code files and to manage projects.

- The standard IDE for Android used to be Eclipse, but is now replaced by Google's own *Android Studio*.

- If you zoom in (metaphorically), one shall find more in-depth processes going on behind the scenes during all of the above steps. For example, advanced users will want to investigate the role of the *Dalvik Virtual Machine*

Steps To Be Covered

- Download and install Android Studio.
- Set up testing on devices and emulators.
- Create a simple “Hello World!” Android app
- Improvise it further to make a personalized android app with a splash screen and your bio

Installing Android Studio

- Check if the Java Development Kit (JDK) installed
  Type *java –version* in the terminal
  If installed already:
  java version "1.7.0_51"
  Java(TM) SE Runtime Environment (build 1.7.0_51-b13)
  Java HotSpot(TM) 64-Bit Server VM (build 24.51-b03, mixed mode)
  Else:
  Command not found
  Download and install JDK from Oracle.

- When ready, download Android Studio from developer.android.com

- Once downloaded, install Android Studio similar to any other program following the instructions provided.

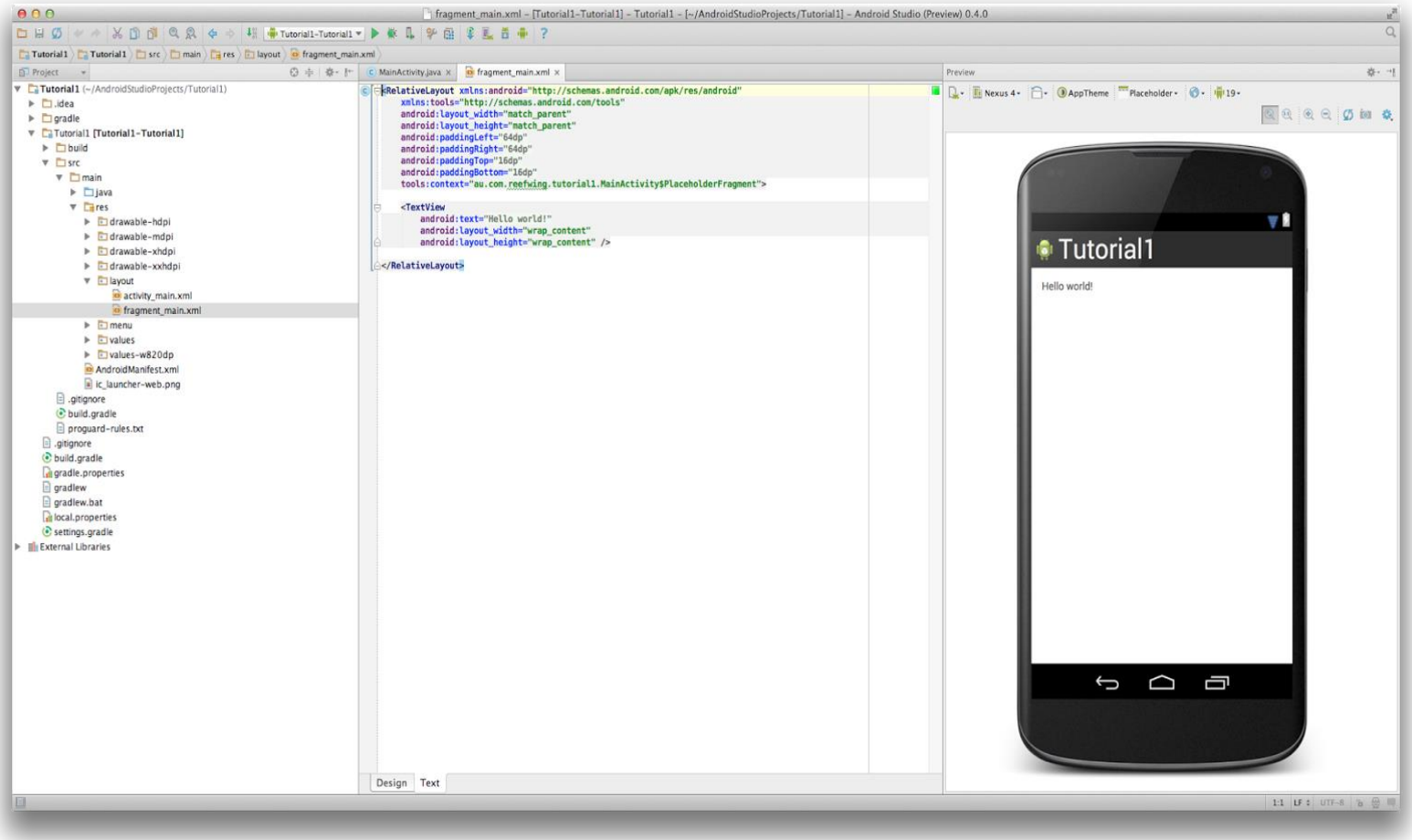
- Launch Android Studio and choose desirable settings.

"Hello World"

- Create a new project 'File >> New Project'
- Sample settings:

Minimum required SDK sets the minimum version of Android needed to run your app.

Target SDK and Compile With aren't crucial here. They should each just be the highest value possible at the time. As new SDK versions come out, this number will rise.



Quick Details

- **Activity** is usually thought of as a "*screen*" in the app. It could be full-screen or it could be a partial-screen pop-up. It is very flexible. If you want your user interacting with something in your app, that's at some level happening via an Activity.

- **Layout** is the design of the android activity/fragment (sub-activity) defined in Android XML file

- **Gradle** is a new build tool. Basically, it takes Java code and XML layouts, and uses the latest Android build tools to create the app package file, known as an APK file. It can have customized configurations to have development or production versions of the app, or add dependencies for third-party libraries.

- **Maven** is another project build tool, and it can also refer to the Maven Central repository of java libraries. It is absurdly easy to use Gradle and Maven Central in concert with Android Studio to incorporate all sorts of functionality from the Android development community.

Android Emulator

- Android Studio comes free with the ability to set up a software-based Android device on your computer and run apps on it, browse websites, debug and everything. This capability is known as the **Android Emulator.**

- One can set up multiple emulators and can set the screen size and platform version for each new emulator to whatever desired. This is great, as it can avoid having to use multiple physical devices for testing.

- Unfortunately, the emulator is painfully slow.
  Note: The truth is that the emulator is prohibitively slow and should generally be avoided unless you have no other option. Over the course of an Android development career, one could literally waste hours of cumulative time waiting for the emulator to launch and load the app

- This opens the Android Virtual Device (AVD) Manager. This is the place where you can create, edit, and delete the various emulators you might want to work with.

Android Project Structure

- Java: The Professional
- Resources: The Artist
- AndroidManifest.xml: The Boss
- Intent: The Job itself

Java: The Professional

- It's the job of the Java code to get things done.
- The code is resides in *"src\main\java directory"* under your main project folder.

Resources: The Artist

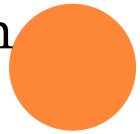
- It's not enough to just get the job done. It needs to be done in style. It is where one can define icons and images, well-designed layouts and animations.
- Initially, the *"src\main\res"* (Resources) folder contains:
- Drawable folders that hold images.
- Layout folder with XML that represents the screen designs.
- Menu folder with XML of the items that will appear on the Action Bar.
- Values folder with XML containing dimensions, strings, and styles.

AndroidManifest.xml: The Boss

- This XML file informs the system of the app's hardware and software requirements and contains the app's name, icon, and version.
- The manifest also filters the Intents coming in.

Intent: The Job itself

- Showing a scree, Navigating to a website etc. All jobs are the form of an Intent in Android
- Makes it easy for different apps on the device to talk to each other.
- Allows to send and receive what are essentially requests for jobs to be done.
- A job could get picked up by the app's own boss (the manifest) or another app.
- For an immediate example, the "Hello World" app already has an Activity called *MainActivity*.
  The manifest has it labelled an intent filter that causes the *MainActivity* to launch when the user selects the app icon from their home screen.
  One could potentially move that filter to another Activity and then that activity would launch instead of MainActivity.
  Basically, the app does whatever the boss says.



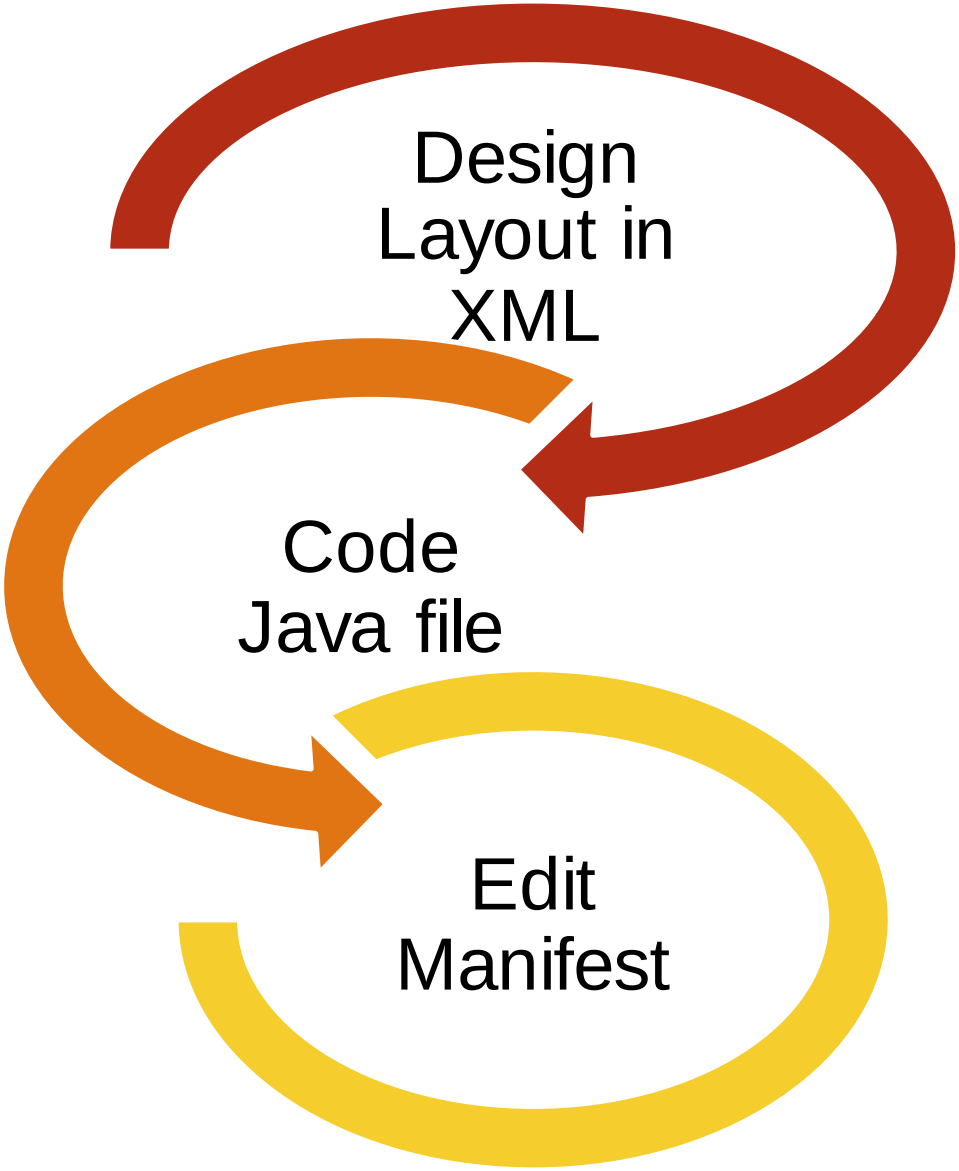
TASK

Develop a Simple Android Application named *'[your_name]'* with splash screen.

App has the following features:

- Splash Screen displaying your name
- Home Screen with background as your picture and text introducing yourself

Sample code: https://github.com/jig08/conferences-seminars-talks/tree/master/fossasia_2016



Rudimentary Steps to be followed...


Github:
jig08
Thank
You
Twitter:
@jigyasa
_grover