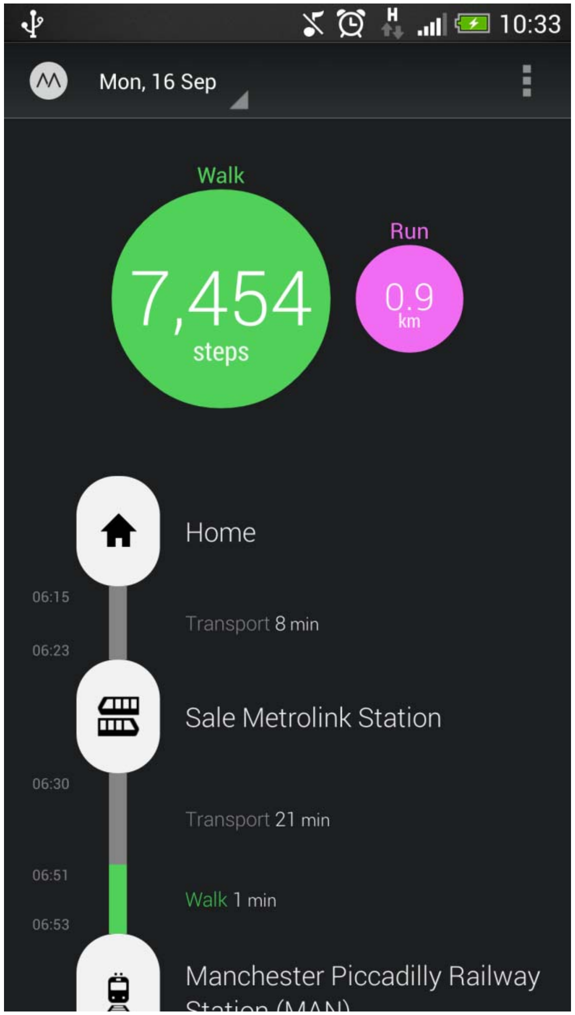
CS 525M Mobile and Ubiquitous Computing

Lecture 3: Intro to Android Programming

Emmanuel Agu

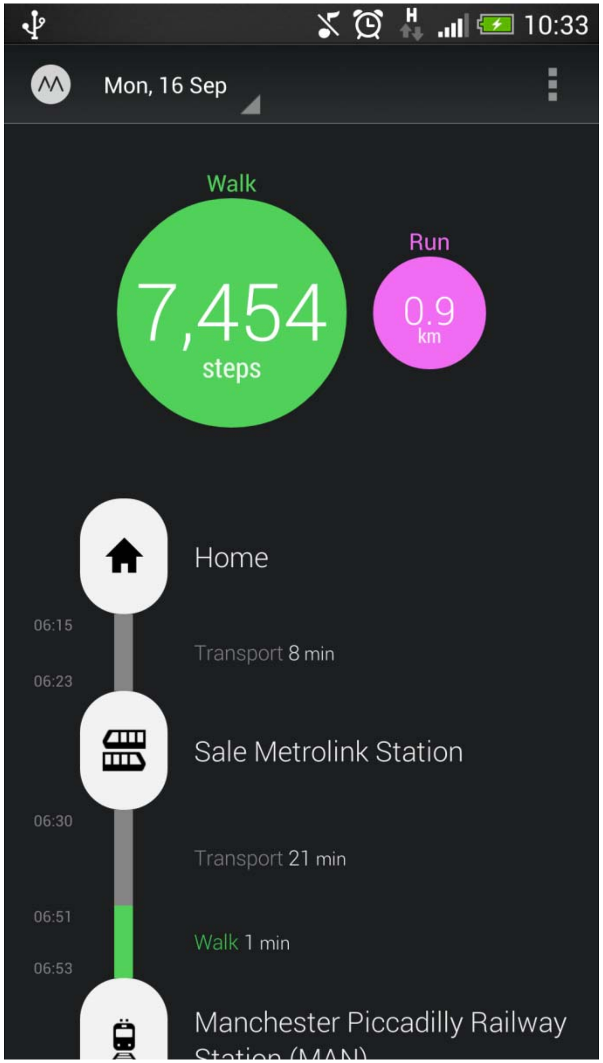
Android App

- Most apps written in Java
- Android SDK tools compile code, data and resource files into **Android PacKage (filename.apk)**.
- Apps download from Google Play, or copied to device as **filename.apk**
- Installation = installing **apk file**
- App elements
  - User Interface
  - Other code designed to run in background (multi-task)



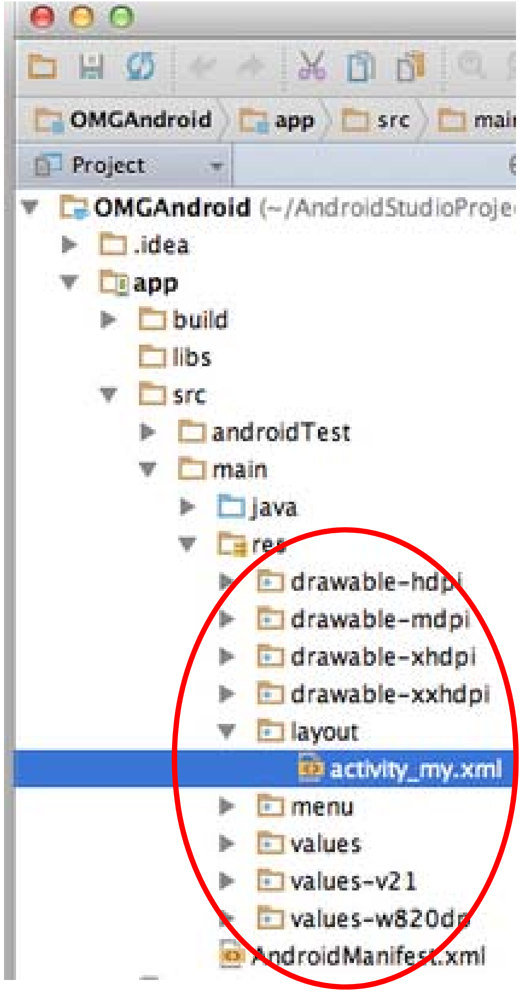
UI Design using XML

- Android separates UI design from the program
- Why? Theoretically, UI can be modified without changing program, Java code
- **Example:** In app shown, shapes, colors can be changed in XML file without changing Java program
- UI designed using graphical (WYSIWYG) tool or Extensible Markup Language (XML)
- XML: Markup language that is both human-readable and machine-readable''



Files in an Android Project

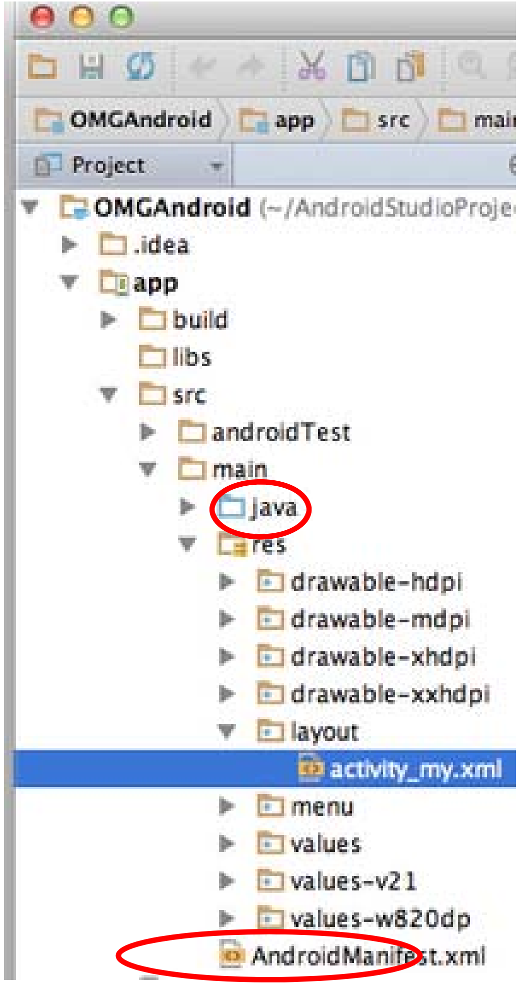
- **res/layout/:** XML files for look or layout of Android screens
- **res/menu/:** XML files for menu specs
- **res/drawable-xyz/:** images (PNG, JPEG, etc) at various resolutions
- **res/raw:** general-purpose files (e.g. audio clips, CSV files
- **res/values/:** strings, dimensions, etc



Files in an Android Project

- **java/:** Java code for programming the “brains” of the app. E.g. What happens on user input, etc

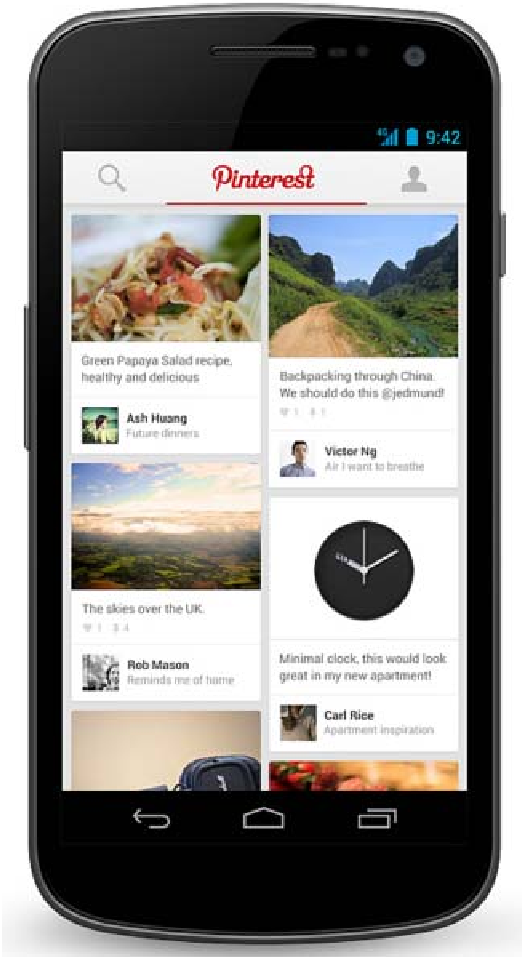
- **Configuration files:** (e.g. AndroidManifext.xml) Contains app name, app screens, etc



```
OMGAndroid 〉 app 〉 src 〉 mai
Project
▼ OMGAndroid (~/AndroidStudioProje
  ▶ .idea
  ▼ app
    ▶ build
      libs
    ▼ src
      ▶ androidTest
      ▼ main
        ▶ java
        ▼ res
          ▶ drawable-hdpi
          ▶ drawable-mdpi
          ▶ drawable-xhdpi
          ▶ drawable-xxhdpi
          ▼ layout
              activity_my.xml
          ▶ menu
          ▶ values
          ▶ values-v21
          ▶ values-w820dp
          AndroidManifest.xml
```

Example: Files in an Android Project

- **res/layout:** The width, height, layout of screen cells are specified in XML file here
- **res/drawable-xyz/:** The images stored in jpg or other format here
- **java/:** App's behavior when user clicks on a selection in java file here
- **AndroidManifext.XML:** Contains app name (Pinterest), list of app screens, etc



Green Papaya Salad recipe, healthy and delicious

Ash Huang
Future dinners

Backpacking through China. We should do this @jedmund!

Victor Ng
Air I want to breathe

The skies over the UK.

Rob Mason
Reminds me of home

Minimal clock, this would look great in my new apartment!

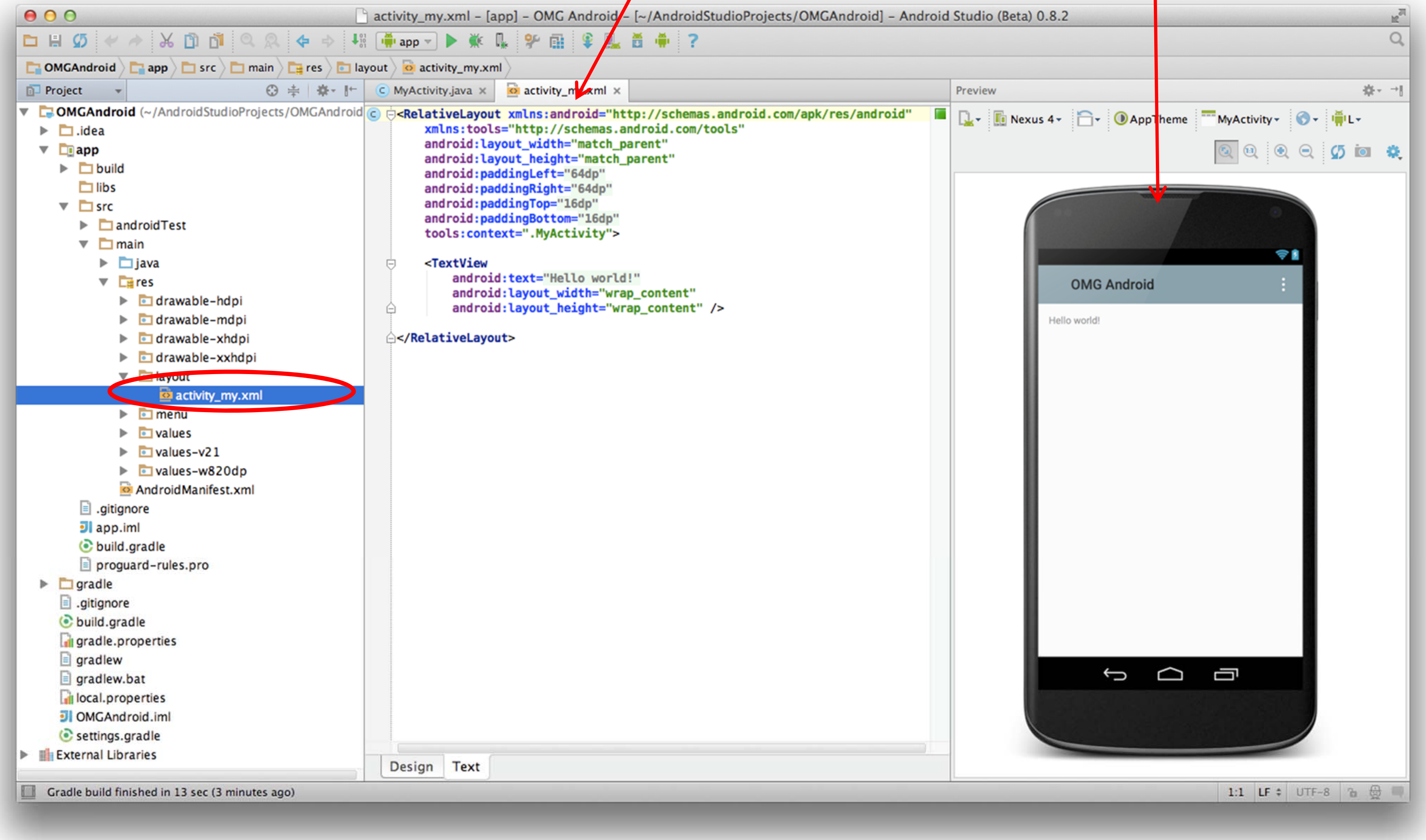
Carl Rice
Apartment inspiration

Editting Android

- Activity_my.xml is XML file specifying screen layout
- Can edit XML directly or drag and drop

Activity_my.xml
(can edit directly)

App running on
Emulator (can edit
Text, drag and drop)

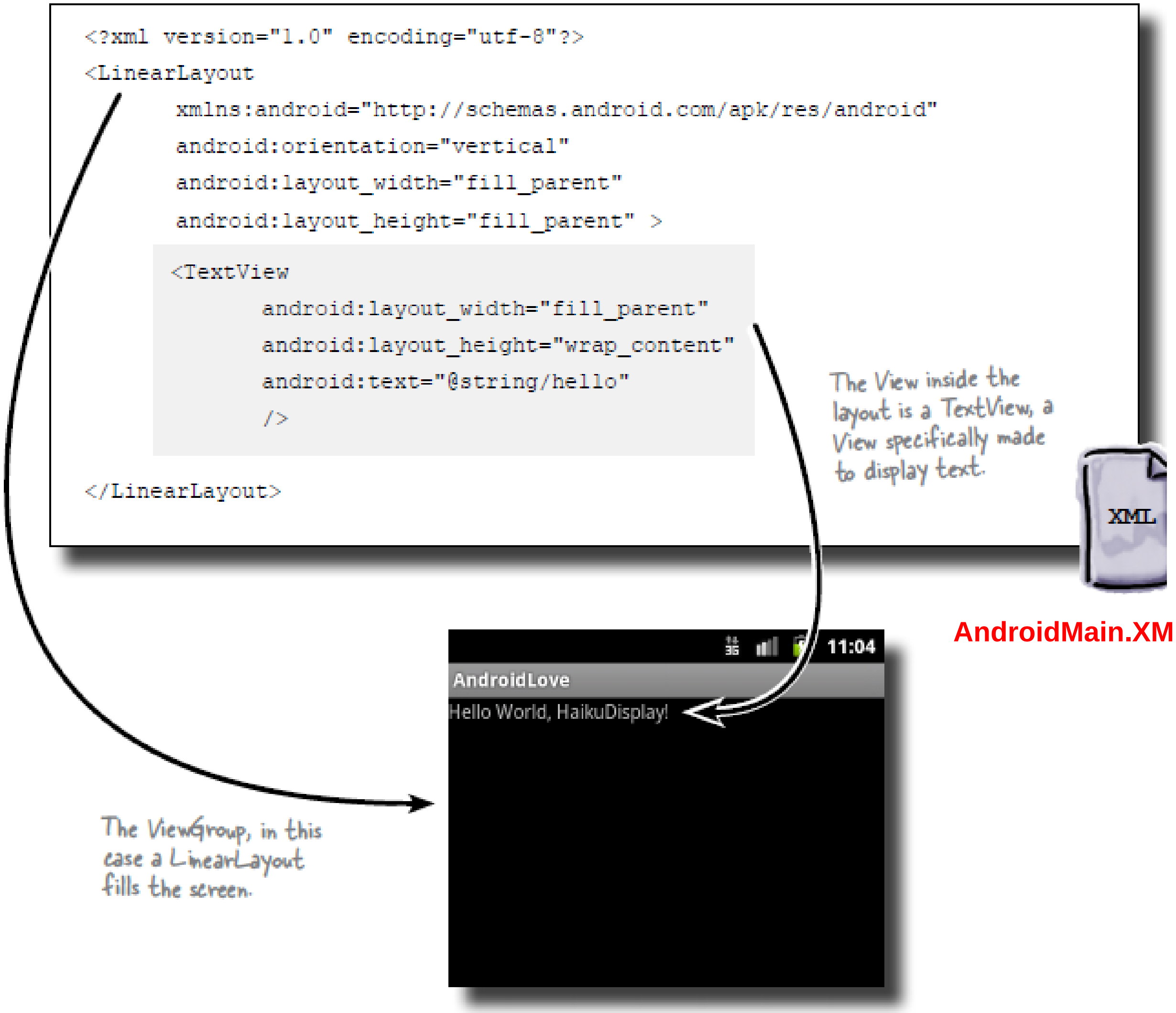


```
<RelativeLayout xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:tools="http://schemas.android.com/tools"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    android:paddingLeft="64dp"
    android:paddingRight="64dp"
    android:paddingTop="16dp"
    android:paddingBottom="16dp"
    tools:context=".MyActivity">

    <TextView
        android:text="Hello world!"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content" />

</RelativeLayout>
```

Gradle build finished in 13 sec (3 minutes ago)

What’s in the XML File?

- Android XML files consist of:
  - UI components called **Views**
  - **ViewGroups** (or layout managers)
- The example XML file shown contains:
  - 1 ViewGroup (LinearLayout) that fills the entire screen
  - 1 View (TextView) that contains text



```
<?xml version="1.0" encoding="utf-8"?>
<LinearLayout
       xmlns:android="http://schemas.android.com/apk/res/android"
       android:orientation="vertical"
       android:layout_width="fill_parent"
       android:layout_height="fill_parent" >
      <TextView
              android:layout_width="fill_parent"
              android:layout_height="wrap_content"
              android:text="@string/hello"
              />

</LinearLayout>
```

The View inside the layout is a TextView, a View specifically made to display text.

The ViewGroup, in this case a LinearLayout fills the screen.

AndroidMain.XML

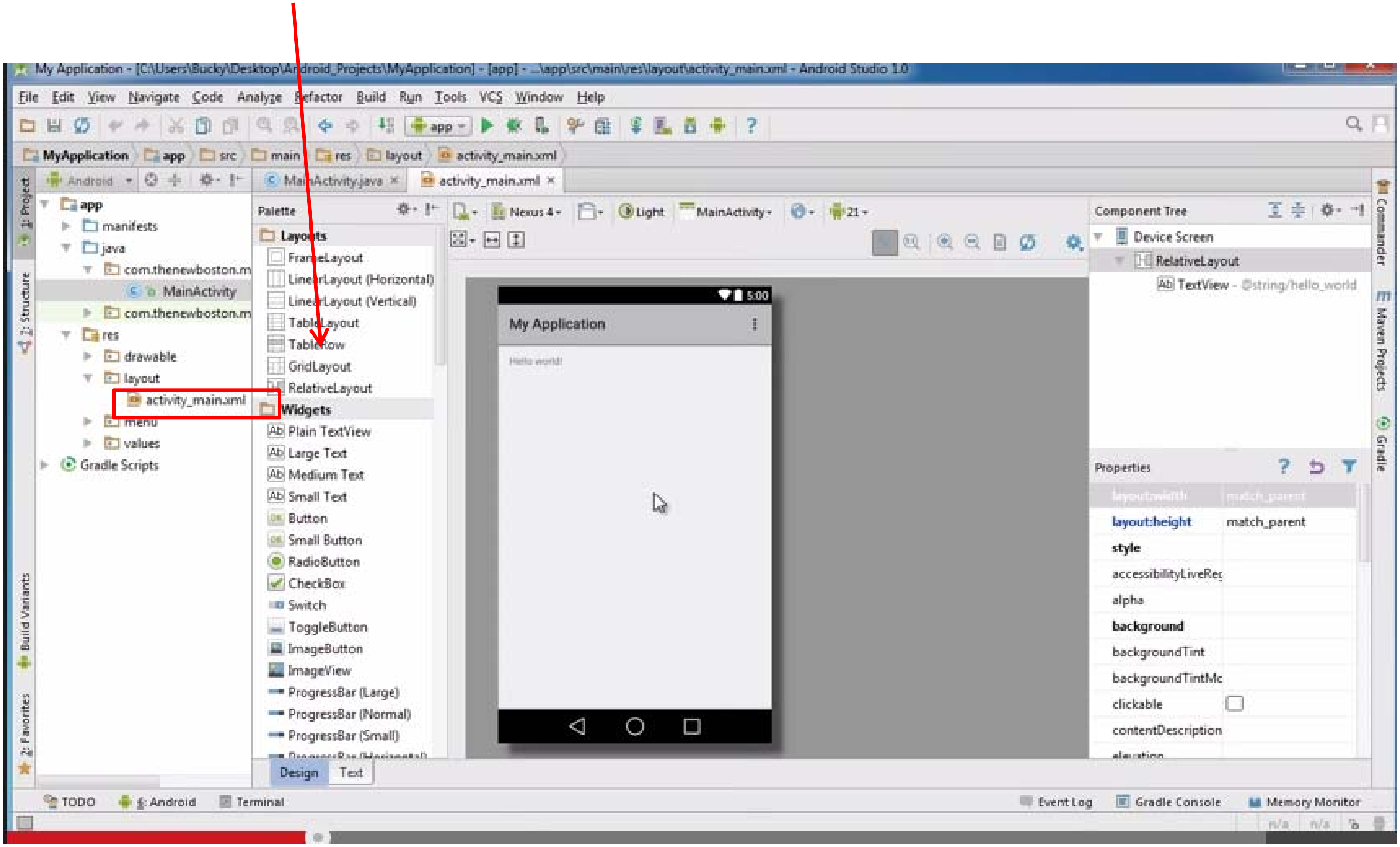
Basic Overview of an App

- Tutorial 8: Basic Overview of an App [11:36 mins]
  - https://www.youtube.com/watch?v=9l1lfWAiHPg

- Main topics
  - Introduces main files of Android App
    - Activity_main.xml
    - MainActivity.java
    - AndroidManifest.xml
  - How to work with these files within Android Studio
  - Editting files using either drag-and-drop interface or XML
  - Flow of basic app

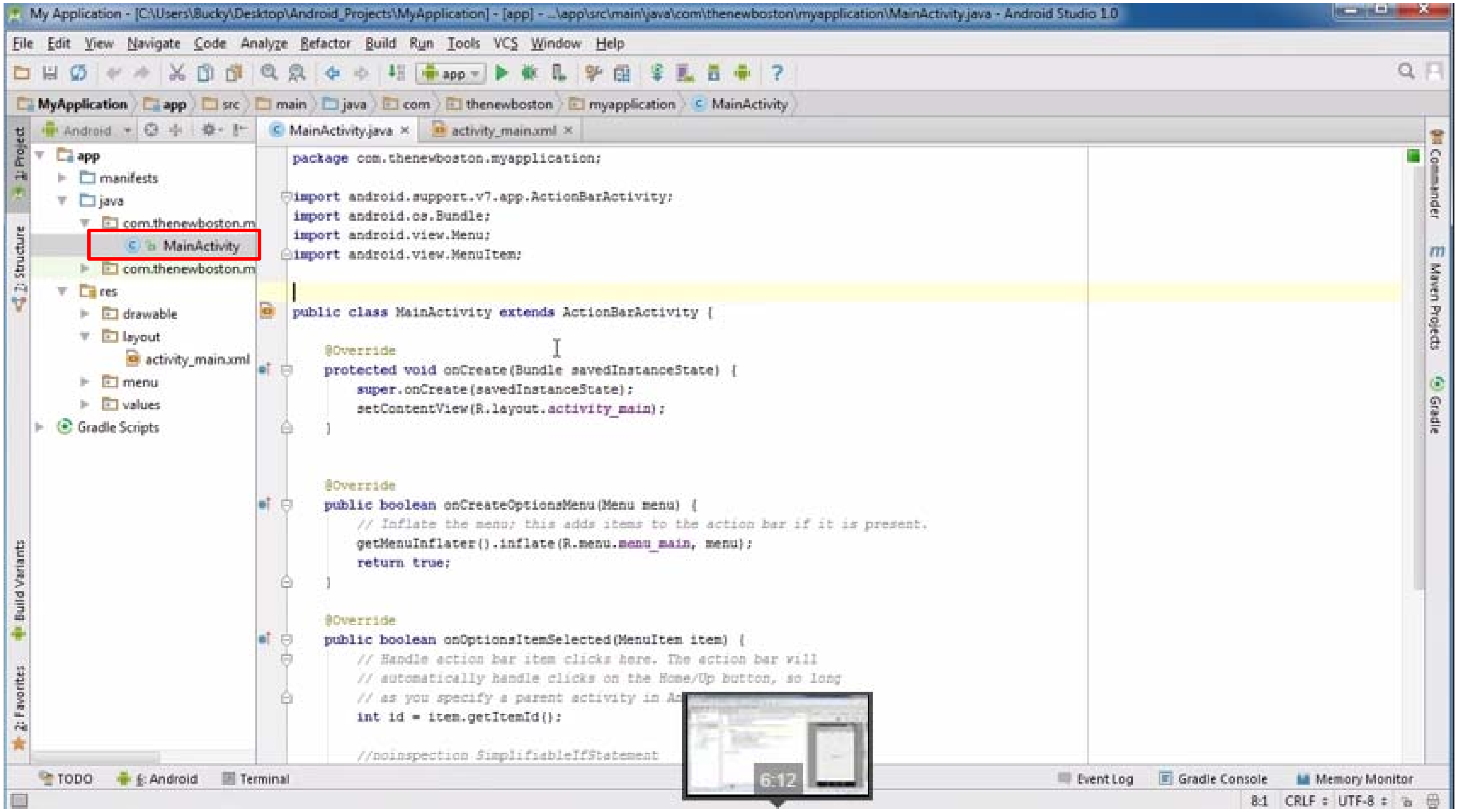
Activity_main.xml

- XML file used to design screen layout, buttons, etc
- **Widgets:** elements that can be dragged onto activity (screen)



MainActivity.java

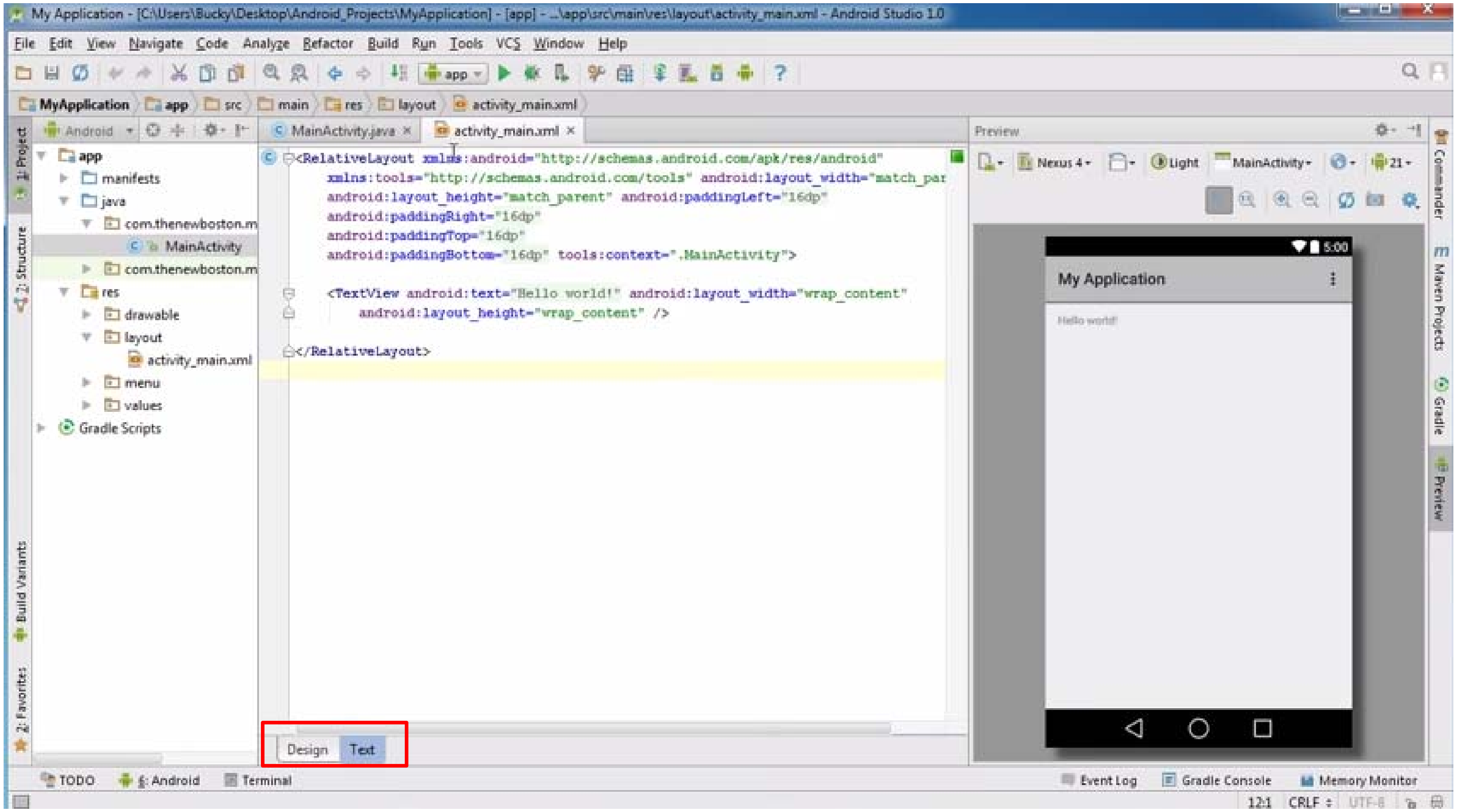
- Used to define actions taken when button clicked (intelligence)


My Application - [C:\Users\Bucky\Desktop\Android_Projects\MyApplication] - [app] - ...\app\src\main\java\com\thenewboston\myapplication\MainActivity.java - Android Studio 1.0
File Edit View Navigate Code Analyze Refactor Build Run Tools VCS Window Help
MyApplication app src main java com thenewboston myapplication MainActivity
Android
MainActivity.java
activity_main.xml
app
manifests
java
com.thenewboston.m
MainActivity
com.thenewboston.m
res
drawable
layout
activity_main.xml
menu
values
Gradle Scripts
1: Project
2: Structure
Build Variants
2: Favorites
Commander
Maven Projects
Gradle
TODO
6: Android
Terminal
Event Log
Gradle Console
Memory Monitor
6:12
8:1 CRLF UTF-8

```
package com.thenewboston.myapplication;

import android.support.v7.app.ActionBarActivity;
import android.os.Bundle;
import android.view.Menu;
import android.view.MenuItem;


public class MainActivity extends ActionBarActivity {

    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_main);
    }


    @Override
    public boolean onCreateOptionsMenu(Menu menu) {
        // Inflate the menu; this adds items to the action bar if it is present.
        getMenuInflater().inflate(R.menu.menu_main, menu);
        return true;
    }

    @Override
    public boolean onOptionsItemSelected(MenuItem item) {
        // Handle action bar item clicks here. The action bar will
        // automatically handle clicks on the Home/Up button, so long
        // as you specify a parent activity in An
        int id = item.getItemId();

        //noinspection SimplifiableIfStatement
```

Activity_main.xml: Text View

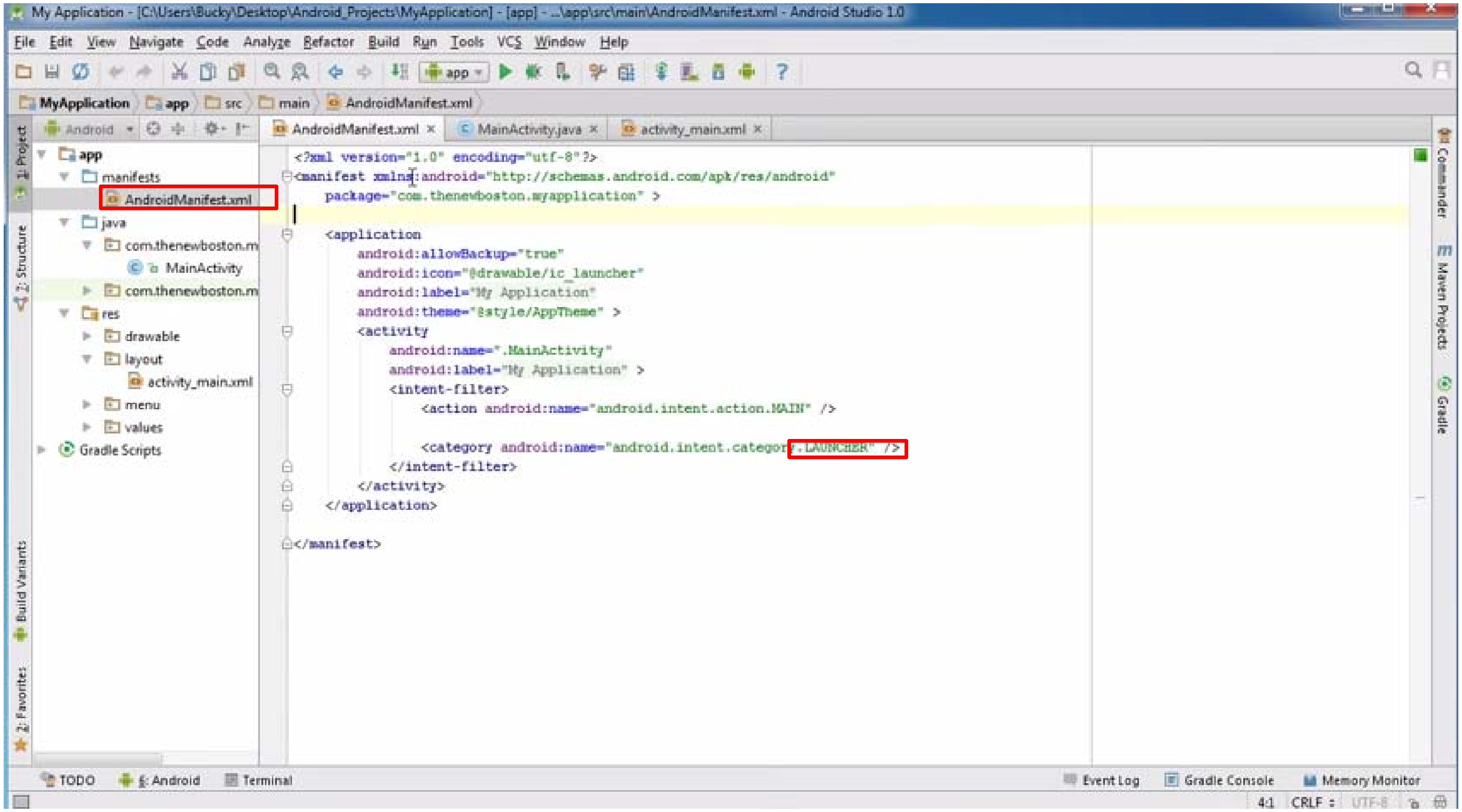
- **Design View:** Drag-and-drop screen (Activity) design
- **Text view:** Directly edit XML file defining screen



```
<RelativeLayout xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:tools="http://schemas.android.com/tools" android:layout_width="match_par
    android:layout_height="match_parent" android:paddingLeft="16dp"
    android:paddingRight="16dp"
    android:paddingTop="16dp"
    android:paddingBottom="16dp" tools:context=".MainActivity">

    <TextView android:text="Hello world!" android:layout_width="wrap_content"
        android:layout_height="wrap_content" />

</RelativeLayout>
```

AndroidManifest.xml

- App’s starting point (a bit like main( ) in C)
- All activities (screens) are listed in AndroidManifest.xml
- Activity with tag “LAUNCHER” is launched first (starting point)


My Application - [C:\Users\Bucky\Desktop\Android_Projects\MyApplication] - [app] - ...\app\src\main\AndroidManifest.xml - Android Studio 1.0
File Edit View Navigate Code Analyze Refactor Build Run Tools VCS Window Help
MyApplication app src main AndroidManifest.xml
Android
AndroidManifest.xml MainActivity.java activity_main.xml
app
manifests
AndroidManifest.xml
java
com.thenewboston.m
MainActivity
com.thenewboston.m
res
drawable
layout
activity_main.xml
menu
values
Gradle Scripts
1: Project
2: Structure
Build Variants
2: Favorites
Commander
Maven Projects
Gradle
TODO
6: Android
Terminal
Event Log
Gradle Console
Memory Monitor
4:1 CRLF UTF-8

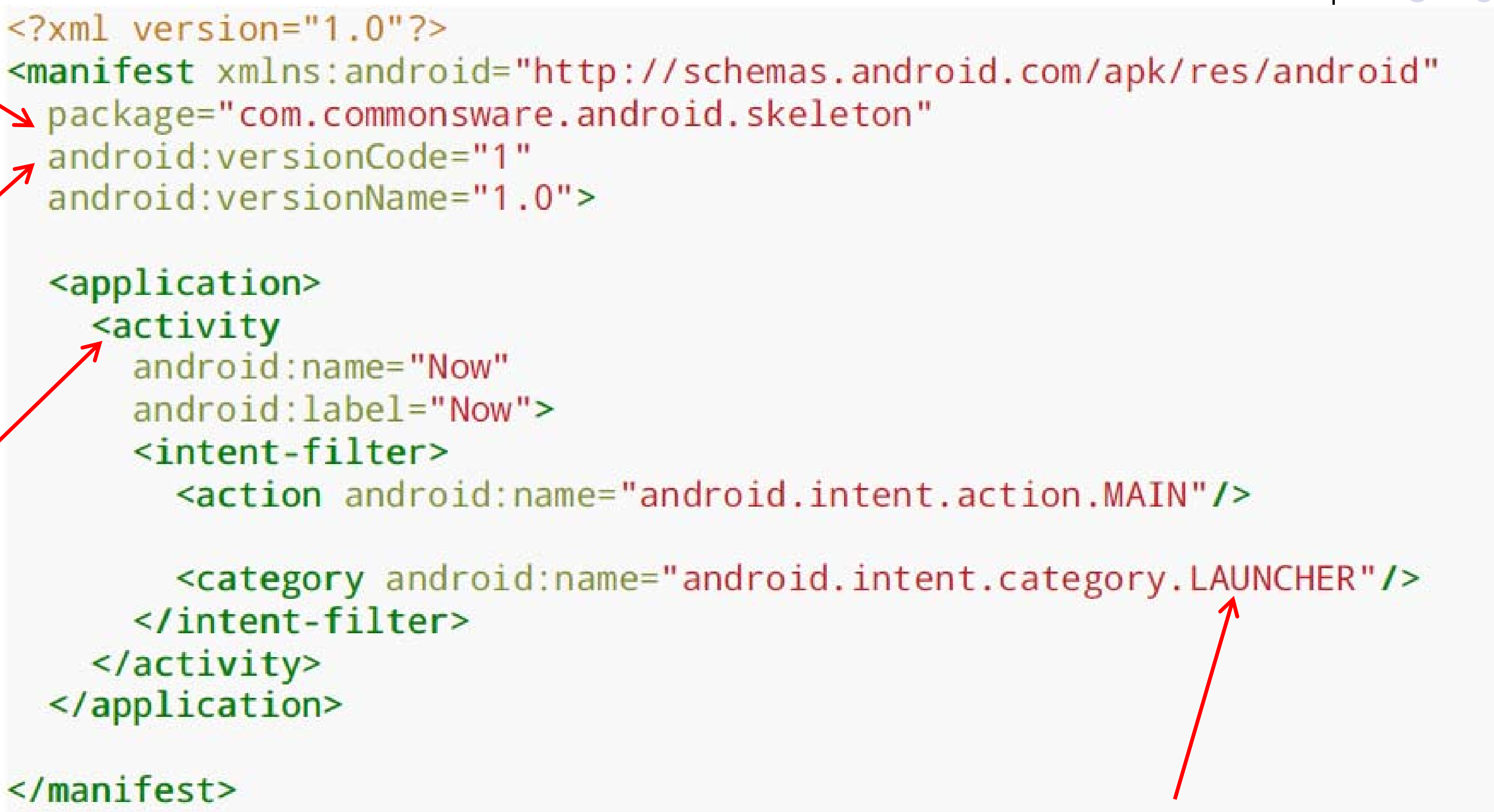
```
<?xml version="1.0" encoding="utf-8"?>
<manifest xmlns:android="http://schemas.android.com/apk/res/android"
    package="com.thenewboston.myapplication" >

    <application
        android:allowBackup="true"
        android:icon="@drawable/ic_launcher"
        android:label="My Application"
        android:theme="@style/AppTheme" >
        <activity
            android:name=".MainActivity"
            android:label="My Application" >
            <intent-filter>
                <action android:name="android.intent.action.MAIN" />

                <category android:name="android.intent.category.LAUNCHER" />
            </intent-filter>
        </activity>
    </application>

</manifest>
```

Inside "Hello World" AndroidManifest.xml

Your package name

Android version

List of activities (screens) in your app

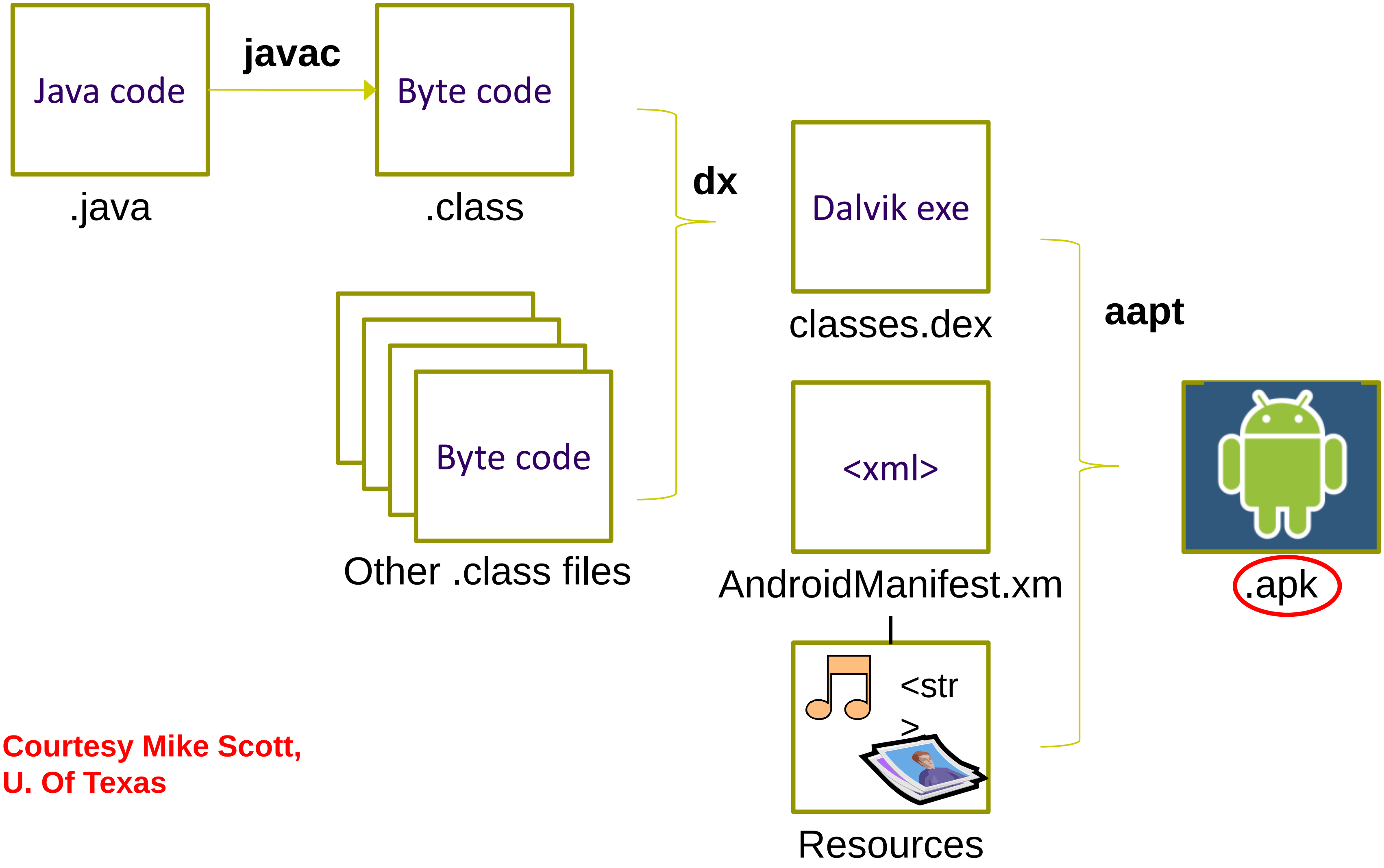


```
<?xml version="1.0"?>
<manifest xmlns:android="http://schemas.android.com/apk/res/android"
  package="com.commonsware.android.skeleton"
  android:versionCode="1"
  android:versionName="1.0">

  <application>
    <activity
      android:name="Now"
      android:label="Now">
      <intent-filter>
        <action android:name="android.intent.action.MAIN"/>

        <category android:name="android.intent.category.LAUNCHER"/>
      </intent-filter>
    </activity>
  </application>

</manifest>
```

One activity (screen) designated LAUNCHER. The app starts running here

Android Compilation Process/Steps

- Dalvik is Android virtual machine
  - Works like Java virtual machine, but optimized for mobile devices



Java code

.java

javac

Byte code

.class

Byte code

Other .class files

dx

Dalvik exe

classes.dex

<xml>

AndroidManifest.xml

<str>

Resources

aapt

.apk

Courtesy Mike Scott,
U. Of Texas

Project 0

- Not to be submitted
- Just step by step guide to:
  - Download course textbook
  - Run tutorials to get started with Android Studio (on emulator)

References

- Android App Development for Beginners videos by Bucky Roberts (thenewboston)
- Busy Coder’s guide to Android version 4.4
- CS 65/165 slides, Dartmouth College, Spring 2014
- CS 371M slides, U of Texas Austin, Spring 2014