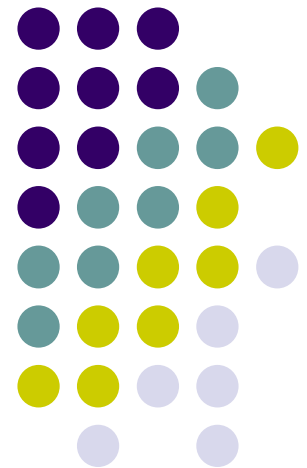


CS 4518 Mobile and Ubiquitous Computing

HW 0 Slides: Introduction to Android

Emmanuel Agu





Tutorials 1-5 Overview



Setting up your Project

- **Tutorials 1 & 2 (Optional):** Installing Java, Android Studio on your own machine
- **Tutorial 3:** Setting up your project
 - How to set up a new Android Project, add new Activity (App screen)
- **Tutorial 4:** Running a Simple App
 - How to select, run app on a virtual device (AVD)
- **Tutorial 5:** Tour of Android Studio Interface
 - Intro to Android Studio menus, toolbars and Drag-and-drop widget palette



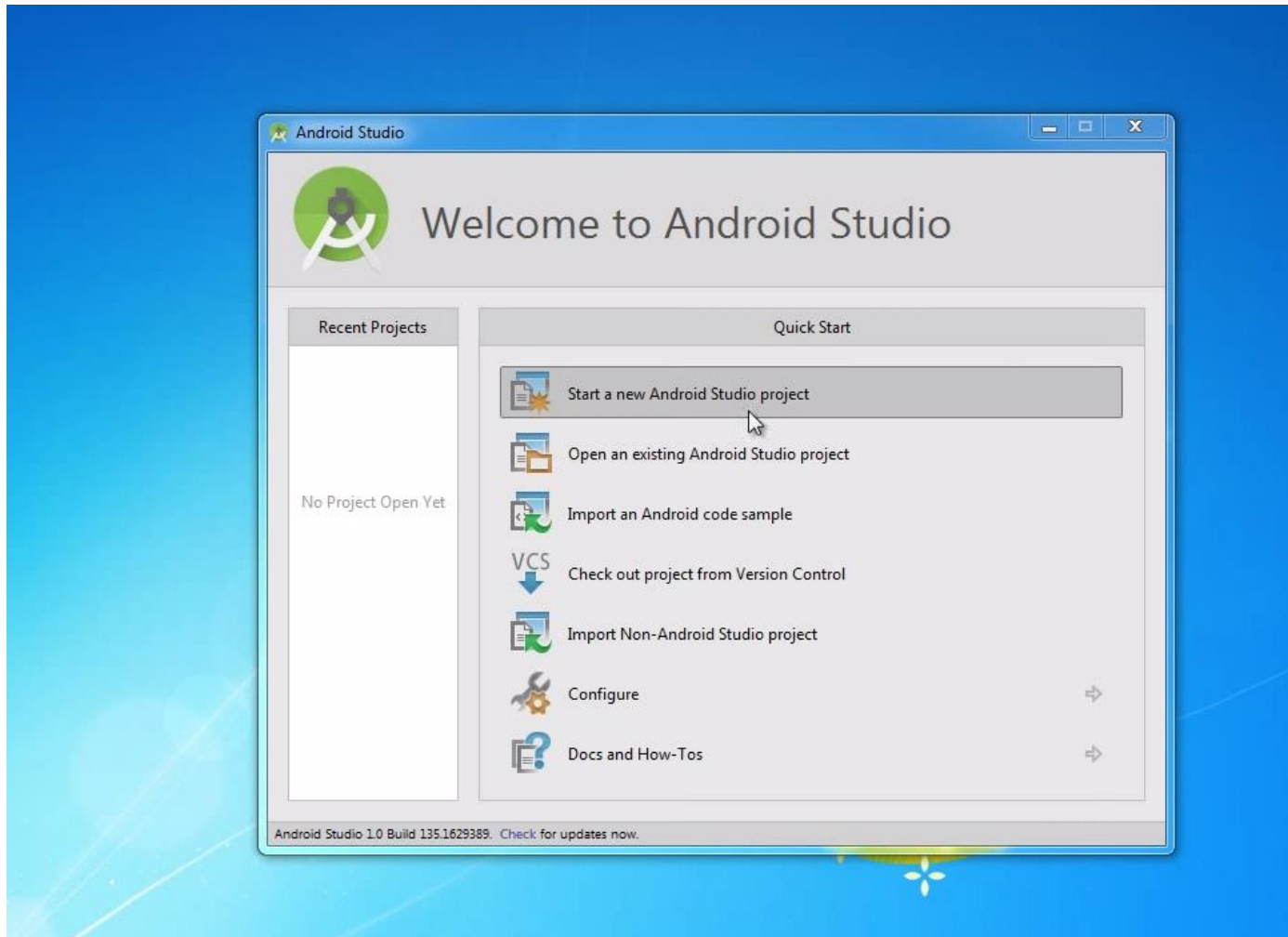
Tutorial 3



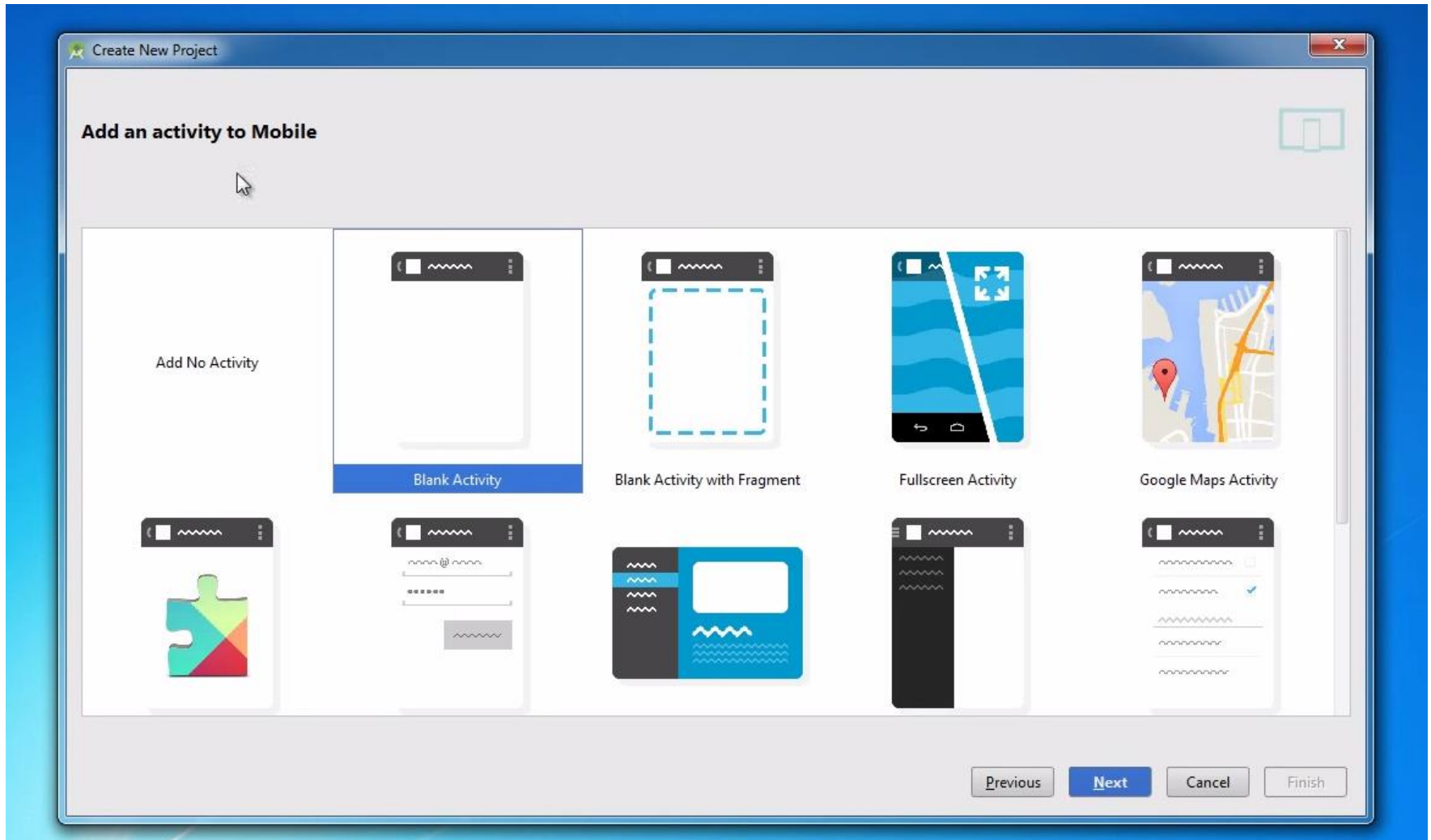
Setting up your Project

- **Tutorial 3: Android App Development for Beginners - 3**
 - Setting up your project by Bucky Roberts (thenewboston)
 - <https://www.youtube.com/watch?v=r4olez0sfvY>
- Main steps to set up Android Project
 - Start a new Android Project
 - Configure new Android Project (select app name, domain name, etc)
 - Set platform and minimum SDK
 - Add an Activity

Start a new Android Project



Add an Activity (Blank Activity is Simplest)





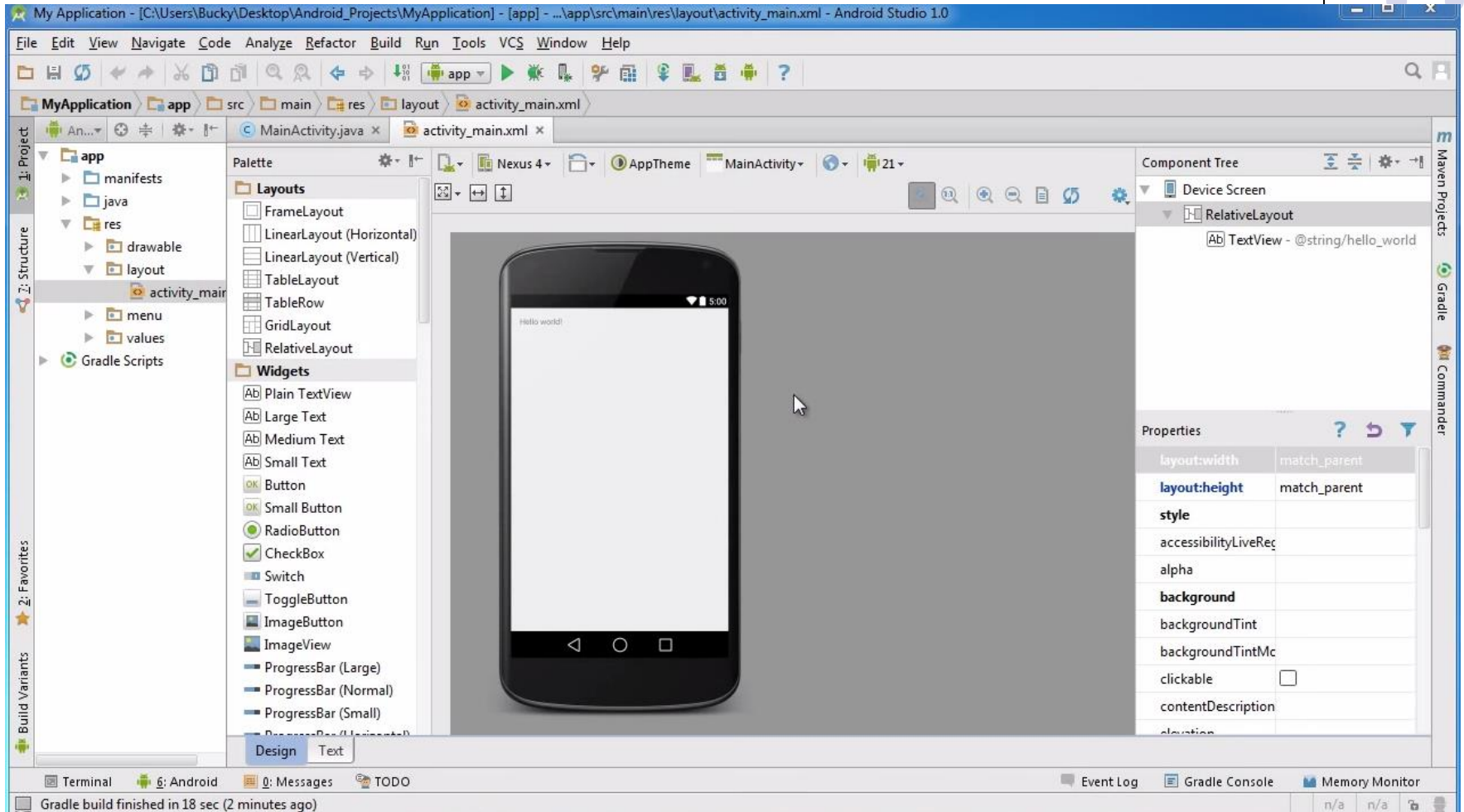
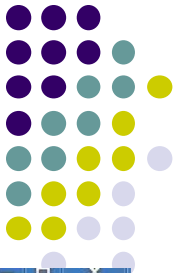
Tutorial 4



Running a Simple App

- Tutorial 4: Android App Development for Beginners - 4 – Running a Simple App [10:48 mins] by Bucky Roberts
 - <https://www.youtube.com/watch?v=qKRWC3Q8wRw>
- Main steps
 - Run Android Studio
 - Fix any remaining issues
 - Run AVD, select virtual device
 - Run App on selected virtual device

Open Android Studio

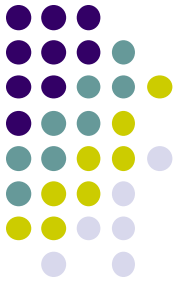


Run AVD Manager

A screenshot of the AVD Manager window in Android Studio. The window title is "AVD Manager" and it features a green header with the Android Studio logo and the text "Your Virtual Devices". Below the header is a table listing two virtual devices: "Buckys Phone" and "Nexus 5 API 21 x86". The table has columns for Type, Name, Resolution, API, Target, CPU/ABI, Size on Disk, and Actions. At the bottom of the window, there is a "Create Virtual Device..." button and "OK" and "Cancel" buttons.

| Type | Name | Resolution | API | Target | CPU/ABI | Size on Disk | Actions |
|------|--------------------|---------------------|-----|---------------|---------|--------------|---------|
| | Buckys Phone | 1080 x 1920: xxhdpi | 21 | Android 5.0.1 | arm | 1 GB | |
| | Nexus 5 API 21 x86 | 1080 x 1920: xxhdpi | 21 | Google APIs | x86 | 1 GB | |

How to Run the App?



Click here to run the app

The screenshot displays the Android Studio interface. On the left, the Project view shows the file structure for 'OMGAndroid', with 'activity_my.xml' selected in the 'layout' folder. The central editor shows the XML code for 'activity_my.xml':

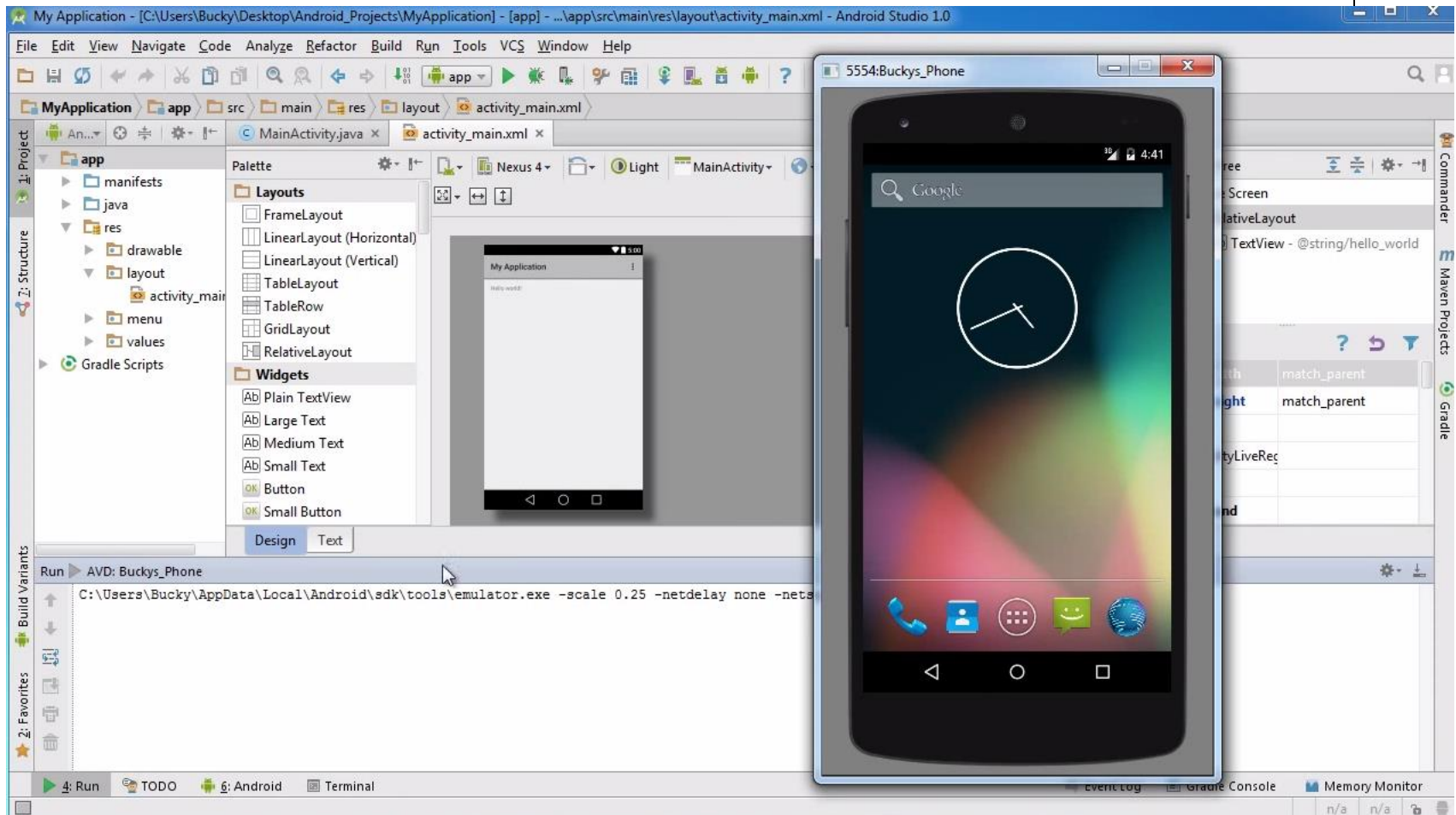
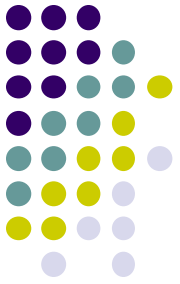
```
<RelativeLayout xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:tools="http://schemas.android.com/tools"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    android:paddingRight="64dp"
    android:paddingLeft="64dp"
    android:paddingTop="16dp"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

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

</RelativeLayout>
```

On the right, the Preview window shows a Nexus 4 device with the app running. The app's title bar is 'OMG Android' and the main content area displays 'Hello world!'. A red arrow points from the text 'Click here to run the app' to the play button icon in the top toolbar of the IDE.

Run App on Virtual Device (Phone)





Tutorial 5



Tour of Android Studio Interface

- Tutorial 5: Tour of Android Studio Interface [6:01 mins]
 - <https://www.youtube.com/watch?v=-pdTqBq2TFQ>
- Quick overview of main sections of Android Studio
 - Windows menu bar
 - Android tool bar
 - Project window
 - Editor Window
 - Palette for Drag-and-Drop Design of Android buttons
- More detailed coverage of specific UI aspects later

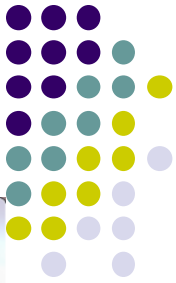
Typical Windows Menu Bar (File, edit, etc)



The screenshot displays the Android Studio 1.0.1 interface. A red box highlights the menu bar at the top, which includes the following items: File, Edit, View, Navigate, Code, Analyze, Refactor, Build, Run, Tools, VCS, Window, and Help. A red arrow points from the text 'Typical Windows Menu Bar (File, edit, etc)' to this menu bar. The main workspace shows a project named 'Hello world' with a 'main' package containing 'activity_main.xml'. The central editor displays a mobile emulator with the text 'Hello world'. The right sidebar contains the Component Tree and Properties panels. The bottom status bar shows 'Run' and 'AVD: Nexus_5_API21_x86'.

| Property | Value |
|----------------------|--------------------------|
| layout:width | match_parent |
| layout:height | match_parent |
| style | |
| accessibilityLiveReg | |
| alpha | |
| background | |
| backgroundTint | |
| backgroundTintMoc | |
| clickable | <input type="checkbox"/> |
| contentDescription | |
| elevation | |
| focusable | <input type="checkbox"/> |
| focusableInTouchM | <input type="checkbox"/> |

Tool Bar: Shortcuts to Frequently used Android-specific Functions (E.g. One-click access to SDK manager)



The screenshot displays the Android Studio 1.0.1 IDE. A red box highlights the toolbar, which contains icons for file operations (New, Open, Save, Print), navigation (Back, Forward), and development (Run, Stop, Refresh, Undo, Redo, Find, Replace, Run on Device, Run on Real Device, Run on Real Device with Camera, Run on Real Device with Location, Run on Real Device with Sensors, Run on Real Device with Network, Run on Real Device with Bluetooth, Run on Real Device with NFC, Run on Real Device with USB, Run on Real Device with ADB, Run on Real Device with Logcat, Run on Real Device with Studio, Run on Real Device with Help).

The main workspace shows a 'Hello world' app running on a Nexus 5 virtual device. The interface includes a Project view on the left, a Palette in the center, a Component Tree on the right, and a Properties panel at the bottom right. The Run/Log panel at the bottom shows the output of the app's execution.

| Property | Value |
|----------------------|--------------------------|
| layout:width | match_parent |
| layout:height | match_parent |
| style | |
| accessibilityLiveReg | |
| alpha | |
| background | |
| backgroundTint | |
| backgroundTintMoc | |
| clickable | <input type="checkbox"/> |
| contentDescription | |
| elevation | |
| focusable | <input type="checkbox"/> |
| focusableInTouchM | <input type="checkbox"/> |

```
Run ▶ AVD: Nexus_5_API21_x86
Inode size: 256
Journal blocks: 1024
Label:
Blocks: 16896
Block groups: 1
Reserved block group size: 7
Created filesystem with 11/4224 inodes and 1302/16896 blocks
```

Path to Current File in IDE Window (Clickable)

The screenshot shows the Android Studio IDE interface. A red box highlights the breadcrumb path in the top toolbar: `Helloworld > app > src > main > res > layout > activity_main.xml`. A red arrow points from the text above to the `activity_main.xml` part of the path. The main editor displays a preview of the app with the text "Hello world!". The left sidebar shows the Project view with the file structure. The right sidebar shows the Component Tree and Properties panels.

File Edit View Navigate Code Analyze Refactor Build Run Tools VCS Window Help

Helloworld > app > src > main > res > layout > activity_main.xml

Project

- app
 - manifests
 - java
 - res
 - drawable
 - layout
 - activity_main.xml
 - menu
 - values
 - Gradle Scripts

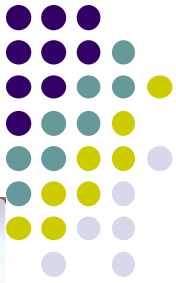
Build Variants

Run ▶ AVD: Nexus_5_API21_x86

Inode size: 256
Journal blocks: 1024
Label:
Blocks: 16896
Block groups: 1
Reserved block group size: 7
Created filesystem with 11/4224 inodes and 1302/16896 blocks

Event Log | Gradle Console | Memory Monitor

21:1 n/a n/a



Editor Window (Allows editing of current file we are working on)



The screenshot displays the Android Studio IDE interface. The main window is titled "Hello world - [C:\temp\android_studio_projects\Helloworld] - [app] - ...\app\src\main\res\layout\activity_main.xml - Android Studio 1.0.1". The interface is divided into several panels:

- Project Structure:** Shows the project hierarchy: `app` (manifests, java, res, layout, menu, values) and `Gradle Scripts`.
- Palette:** Contains "Layouts" (FrameLayout, LinearLayout, etc.) and "Widgets" (TextView, Button, etc.).
- Editor Window:** The central area, highlighted with a red box, shows the design view of the app. It displays a mobile screen with the text "Hello world" and "Hello world!". A red arrow points to the "activity_main.xml" tab above this window.
- Component Tree:** Shows the hierarchy of UI components: `RelativeLayout` containing `TextView` with the text `@string/hello_world`.
- Properties:** A table showing the properties of the selected `TextView` widget.
- Run Console:** Shows the output of the AVD emulator: "AVD: Nexus_5_API_21_x86" and "Created filesystem with 11/4224 inodes and 1302/16896 blocks".

| Property | Value |
|----------------------|--------------------------|
| layout:width | match_parent |
| layout:height | match_parent |
| style | |
| accessibilityLiveReg | |
| alpha | |
| background | |
| backgroundTint | |
| backgroundTintMoc | |
| clickable | <input type="checkbox"/> |
| contentDescription | |
| elevation | |
| focusable | <input type="checkbox"/> |
| focusableInTouchM | <input type="checkbox"/> |

Clicking on Editor Window Tabs switches between project files



The screenshot shows the Android Studio IDE interface. The top menu bar includes File, Edit, View, Navigate, Code, Analyze, Refactor, Build, Run, Tools, VCS, Window, and Help. The toolbar contains various icons for file operations and development. The breadcrumb navigation shows the path: Helloworld > app > src > main > java > com > example > emmanuel > helloworld > MainActivity. The editor window has two tabs: MainActivity.java (active) and activity_main.xml. A red arrow points to the MainActivity.java tab. The editor displays the following Java code:

```
package com.example.emmanuel.helloworld;

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

public class MainActivity extends AppCompatActivity {

    @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 AndroidManifest.xml.
        int id = item.getItemId();

        //noinspection SimplifiableIfStatement
        if (id == R.id.action_settings) {
            return true;
        }

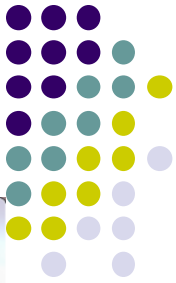
        return super.onOptionsItemSelected(item);
    }
}
```

The bottom of the IDE shows the Run tab with the following output:

```
Run ▶ AVD: Nexus_5_API_21_x86
Inode size: 256
Journal blocks: 1024
Label:
Blocks: 16896
Block groups: 1
Reserved block group size: 7
Created filesystem with 11/4224 inodes and 1302/16896 blocks
```

The status bar at the bottom indicates the current encoding is 1:1 CRLF + UTF-8 + [icon].

Project Window (Shows project files, packages, etc)



The screenshot displays the Android Studio IDE with the following components:

- Project Window (Left):** A tree view showing the project structure for 'Hello world'. The 'app' folder is expanded, showing subfolders for 'manifests', 'java', 'res' (containing 'drawable', 'layout', 'menu', 'values'), and 'Gradle Scripts'. A red box highlights this window, and a red arrow points to the 'res/layout' folder.
- Palette (Middle-Left):** A list of UI components categorized into 'Layouts' (FrameLayout, LinearLayout, etc.) and 'Widgets' (TextView, Button, etc.).
- Design View (Center):** A visual representation of the app's main screen, showing a white background with the text 'Hello world!' and an Android logo.
- Component Tree (Middle-Right):** A hierarchical view of the UI components, showing a 'RelativeLayout' containing a 'TextView' with the text '@string/hello_world'.
- Properties Panel (Bottom-Right):** A table of properties for the selected 'TextView' widget.

| Property | Value |
|----------------------|--------------------------|
| layout:width | match_parent |
| layout:height | match_parent |
| style | |
| accessibilityLiveReg | |
| alpha | |
| background | |
| backgroundTint | |
| backgroundTintMoc | |
| clickable | <input type="checkbox"/> |
| contentDescription | |
| elevation | |
| focusable | <input type="checkbox"/> |
| focusableInTouchM | <input type="checkbox"/> |

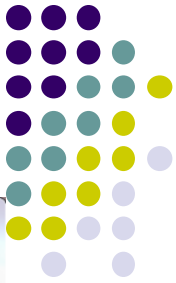
At the bottom, the 'Run' console shows the output of the AVD creation process, including details about the filesystem and storage allocation.

Palette of Drag-and-Drop Elements for Designing Interface (Layout, widgets, etc)



The screenshot displays the Android Studio interface for an Android application. The central focus is the **Palette** window, which is highlighted with a red box and a red arrow pointing from the title above. The palette is divided into two sections: **Layouts** and **Widgets**. The **Layouts** section includes options like FrameLayout, LinearLayout (Horizontal and Vertical), TableLayout, GridLayout, and RelativeLayout. The **Widgets** section includes Plain TextView, Large Text, Medium Text, Small Text, Button, Small Button, RadioButton, CheckBox, Switch, ToggleButton, ImageButton, ImageView, and various ProgressBar sizes. To the right of the palette is a preview window showing a mobile device screen with the text "Hello world!". Further right are the **Component Tree** and **Properties** panels. The Component Tree shows a RelativeLayout containing a TextView with the text "@string/hello_world". The Properties panel shows various attributes for the selected TextView, such as layout:width, layout:height, style, accessibilityLiveReg, alpha, background, backgroundTint, backgroundTintMoc, clickable, contentDescription, elevation, focusable, and focusableInTouchM. The bottom of the screen shows the **Run** console with output for an AVD (Nexus_5_API_21_x86), including details like inode size, journal blocks, label, blocks, block groups, reserved block group size, and filesystem creation information.

Parameters of Drag-and-Drop Elements for Designing Interface (e.g. colors, dimensions of widgets, etc)



The screenshot displays the Android Studio interface for an Android application. The central design view shows a mobile device screen with the text "Hello world!". The Properties panel on the right is highlighted with a red box, showing the following parameters for the selected TextView widget:

| Property | Value |
|----------------------|--------------------------|
| layout:width | match_parent |
| layout:height | match_parent |
| style | |
| accessibilityLiveReg | |
| alpha | |
| background | |
| backgroundTint | |
| backgroundTintMoc | |
| clickable | <input type="checkbox"/> |
| contentDescription | |
| elevation | |
| focusable | <input type="checkbox"/> |
| focusableInTouchM | <input type="checkbox"/> |

The Properties panel also includes a search bar and icons for help, undo, and redo. The interface also shows the Project view on the left, the Palette of widgets, and the Component Tree on the right.



Tutorial 8



Basic Overview of an App

- Tutorial 8: Basic Overview of an App [11:36 mins]
 - <https://www.youtube.com/watch?v=9l1fWAIHPg>

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

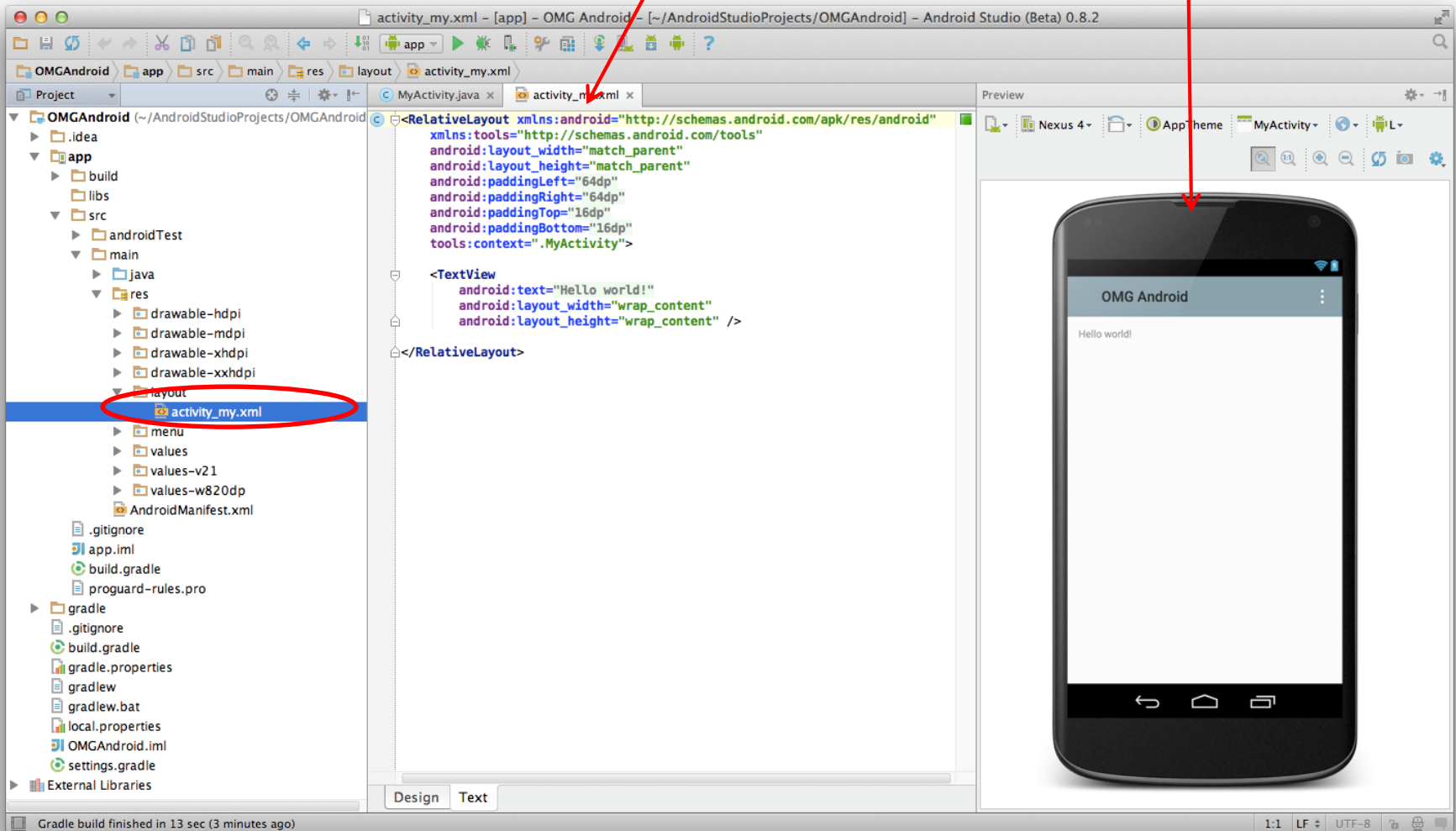
Editing Android

Activity_my.xml
(can edit directly)

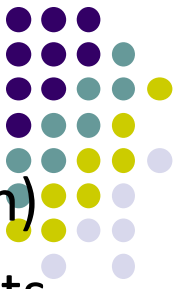
App running on
Emulator (can edit
Text, drag and drop)



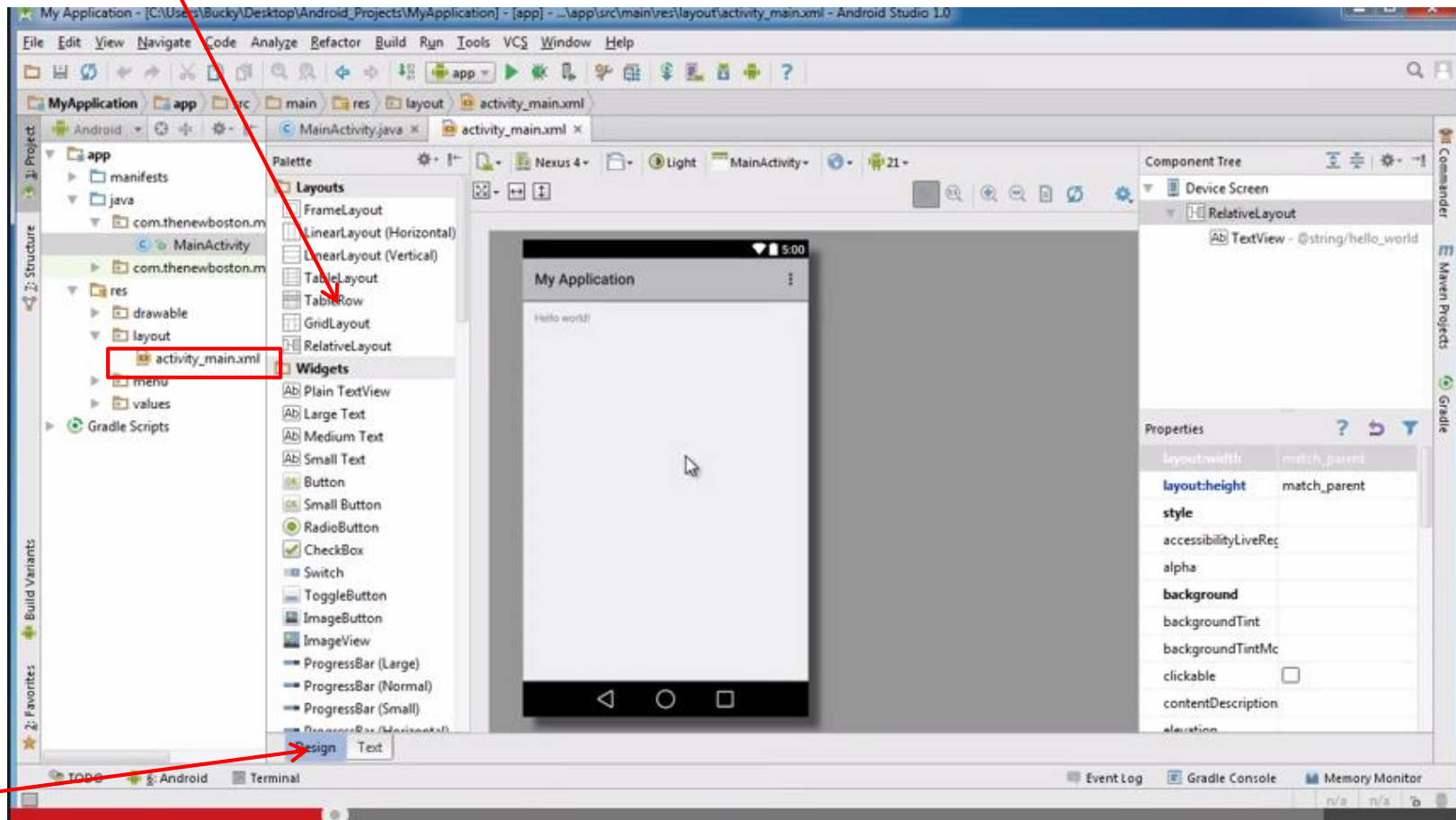
- Activity_my.xml is XML file specifying screen layout, widgets
- Can edit XML directly or drag and drop



Activity_main.xml



- **Widgets:** elements that can be dragged onto activity (screen)
- **Design View:** Design app screen using Drag-and-drop widgets

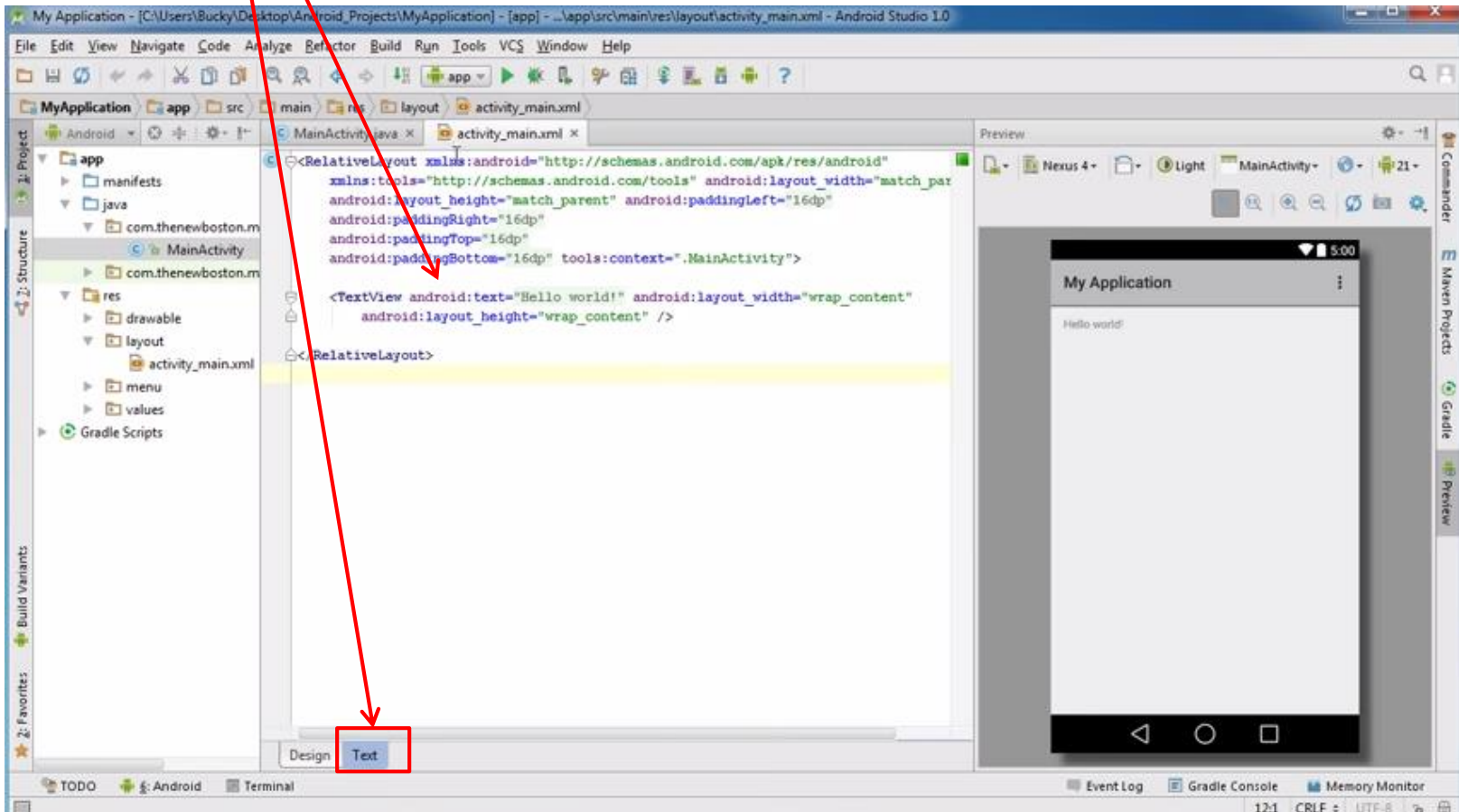


Design
view

Activity_main.xml: Text View



- **Text view:** Design screen by editing XML file directly
- **Note:** dragging and dropping widgets auto-generates corresponding XML





MainActivity.java

- Java code, defines actions, handles interaction/put taken (intelligence)
 - E.g. What app will do when button/screen clicked

```
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 AndroidManifest.xml.
        int id = item.getItemId();

        //noinspection SimplifiableIfStatement
    }
}
```

AndroidManifest.xml

- App's starting point (a bit like main() in C)

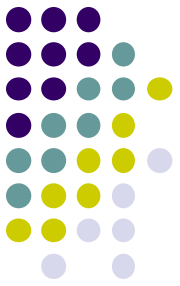
A screenshot of the Android Studio IDE. The main editor window displays the content of the AndroidManifest.xml file. The file is highlighted in yellow. A red box highlights the <activity> element, and another red box highlights the <category android:name="android.intent.category.LAUNCHER" /> line within it. The Project view on the left shows the file structure with 'AndroidManifest.xml' selected. The status bar at the bottom indicates '4:1 CRLF UTF-8'.



Android UI Tour

Home Screen

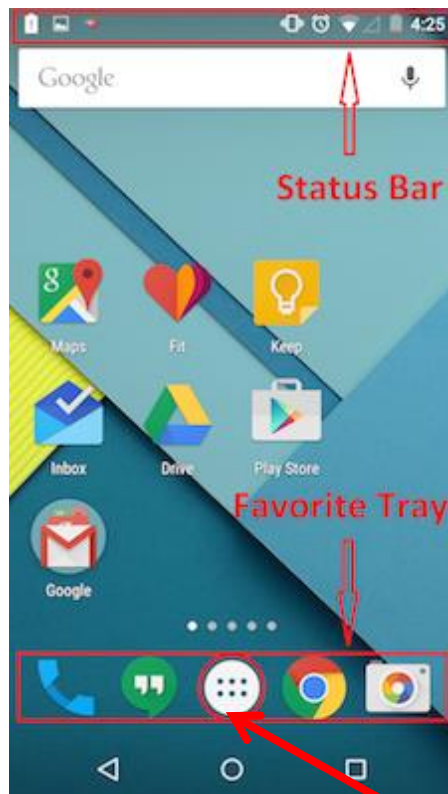
- First screen, includes **favorites** tray (e.g phone, mail, messaging, web, etc)





All Apps Screen

- Accessed by touching **all apps button** in favorites tray
- Can swipe through multiple app screens, customizable



Android 5.0

all apps button

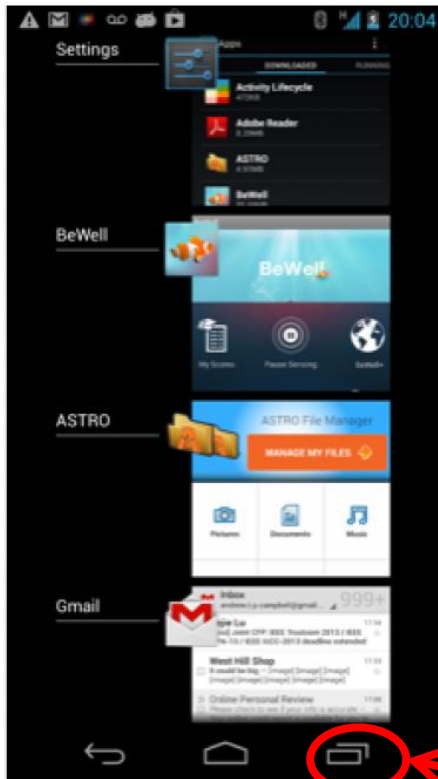


Android 5.0

Recent Apps Screen



- Accessed by touching **recent apps button**
- Shows recently used apps, touch app to switch to it



recent apps button



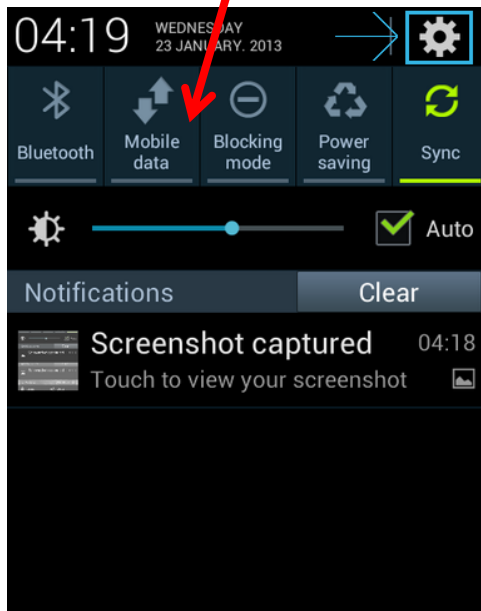
Android 5.0

Status Bar and Notification Screen

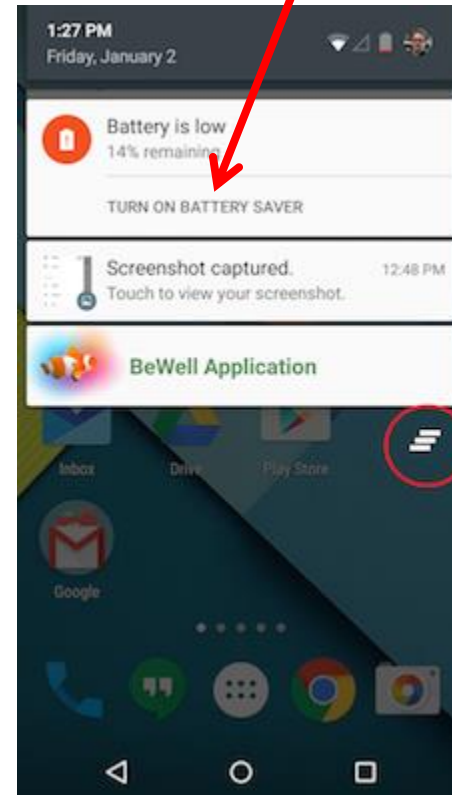


- **Status:** time, battery, cell signal strength, bluetooth enabled, etc
- **Notification:** wifi, mail, bewell, voicemail, usb active, music, etc

Status bar



Notification Screen





References

- Android App Development for Beginners videos by Bucky Roberts (thenewboston)
- Ask A Dev, Android Wear: What Developers Need to Know, <https://www.youtube.com/watch?v=zTS2NZpLyQg>
- Ask A Dev, Mobile Minute: What to (Android) Wear, https://www.youtube.com/watch?v=n5Yjzn3b_aQ
- 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