**­­­455-555 Week 1 Notes**

**Topics**

Android Studio IDE intro (Installation, Configurations & Updating)

-The AVD Manager, SDK Manager

-Android Emulator

Application Layouts

-Layout elements & attributes

App Creation Example in Java

Final Project Ideas?

Mobile Business Use Cases

<https://www.apple.com/business/success-stories/>

<https://www.apple.com/business/success-stories/construction/rogers-obrien/>

Api’s

<https://www.youtube.com/watch?v=mDAoLO4G4CQ> (TT 3:04)

Android LifeCycle

<https://stackoverflow.com/questions/8515936/android-activity-life-cycle-what-are-all-these-methods-for>

[ **Updating your SDK!** ]

A good idea is to check *periodically* for updates to your Android Studio, in particular your SDK.

Ex. From your menu go to **Tools > SDK Manager**.If you see under the Status column that an update is available for the SDK version your using (ex. Android 10 Q) then click on the version use wish updated here, then click OK.

**­­­­­­­­­**A screenshot of a cell phone

Description automatically generated

Click OK again to confirm the version to start the update!

A screenshot of a cell phone

Description automatically generated

[ **Android Studio IDE Basics- navigations, etc.** ]

Navigation area (Android) **/ Project Structure**

A screenshot of a cell phone

Description automatically generated

Other Navigation Views

A screenshot of a cell phone

Description automatically generated

Code editor

A screenshot of a cell phone screen with text

Description automatically generated

XML Layout (Code) View

A screenshot of a cell phone

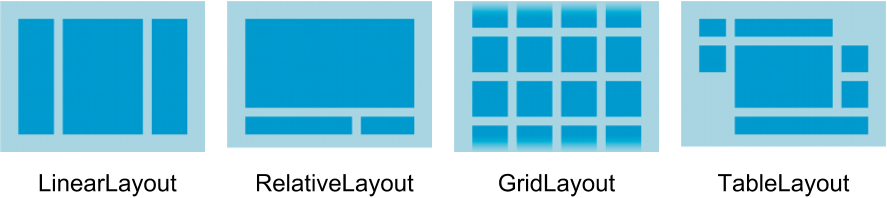
Description automatically generated

XML Layout (Design) View

A screenshot of a cell phone

Description automatically generated

**XML** **Various Layouts**

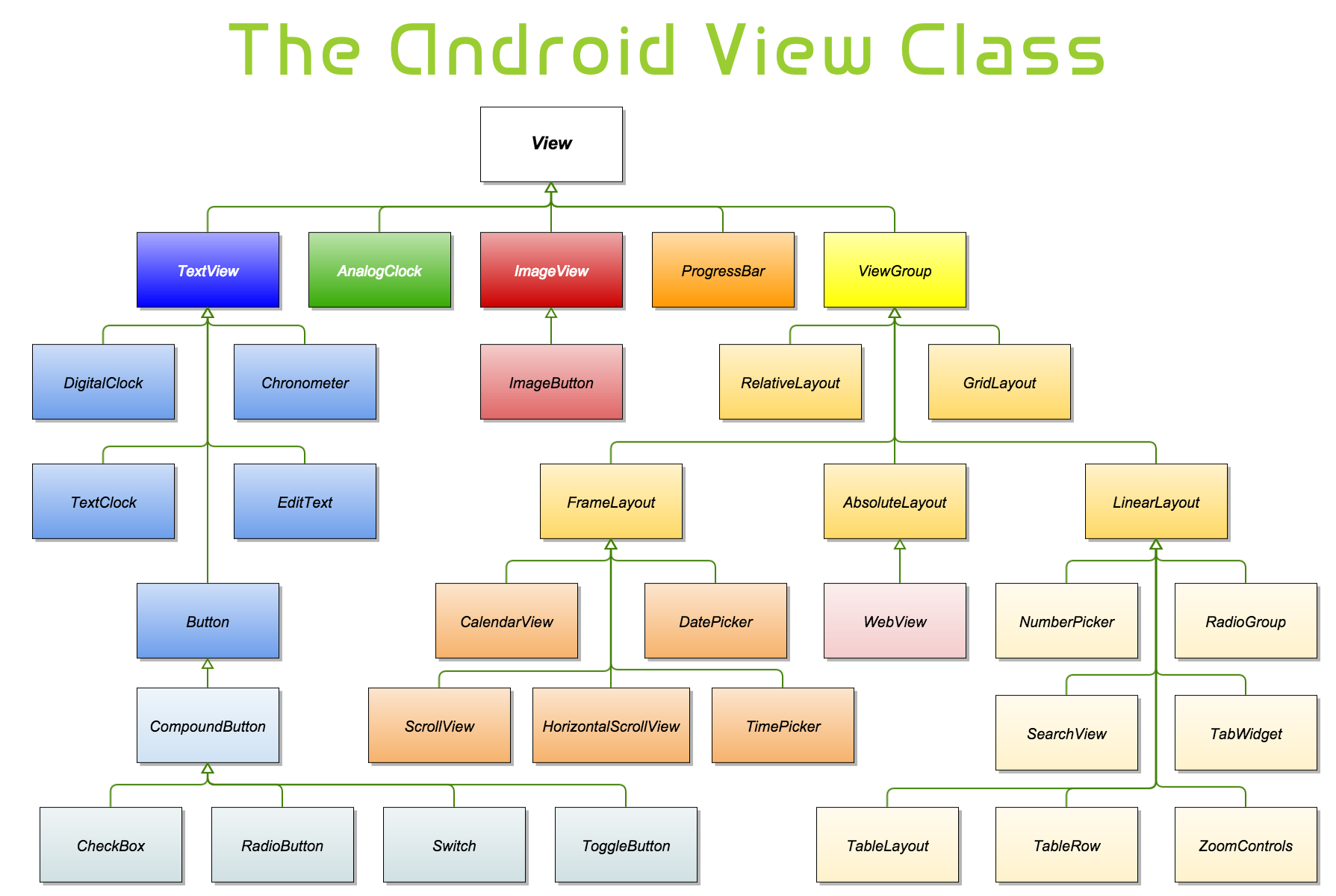


ConstraintLayout

**A screenshot of a computer

Description automatically generated**

**What goes into a layout? Views, Views, Views…**



App Example follows -> Generate Toast “pop-up” message on some click event

-Elements (aka Controls or Objects): **Textview, button**

-Events:

*For buttons*: 1. setOnClickListener (example shown next) or

2. “home brew style” Method creation

A screenshot of a cell phone

Description automatically generated

Run Time snapshot

**Sample XML Layout file: File=>activity\_main.xml**

*<?***xml version="1.0" encoding="utf-8"***?>*<**LinearLayout xmlns:android="http://schemas.android.com/apk/res/android"  
 android:layout\_width="match\_parent"  
 android:layout\_height="match\_parent"**>  
  
 <**TextView  
 android:id="@+id/textView"  
 android:layout\_width="25px"  
 android:layout\_height="wrap\_content"  
 android:layout\_weight="1"  
 android:text="My Text my life"** />  
  
 <**Button  
 android:id="@+id/button"  
 android:layout\_width="wrap\_content"  
 android:layout\_height="wrap\_content"  
 android:layout\_weight="1"  
 android:text="Show Message"** />  
</**LinearLayout**>

**Sample Application code: Code => MainActivity.java**

Note- Package & Import Statements!

**package** com.example.jamespapademas.message;  
**import** android.os.Bundle;  
**import** androidx.appcompat.app.AppCompatActivity;

**import** android.view.View;  
**import** android.widget.Button;  
**import** android.widget.Toast;

Note Java Inheritance!

**public class** MainActivity **extends** AppCompatActivity {

static int *ctr* = 0;

@Override  
 **protected void** onCreate(Bundle savedInstanceState) {

**super**.**onCreate**(savedInstanceState);  
 setContentView(R.layout.***activity\_main***);  
  
 **Button button** = findViewById(R.id.***button***);  
   
 **button**.setOnClickListener(**new** View.OnClickListener() {  
  
 @Override  
 **public void** onClick(View arg0) {  
  
 Toast.*makeText*(getApplicationContext(),  
 **"Button num clicked " +** ctr,

Toast.***LENGTH\_LONG***).show();

ctr++;  
 }  
 });  
  
 }

Place your cursor at the end of the line needing an import and press **ALT + ENTER** to import needed package(s)

**A screenshot of a cell phone

Description automatically generated**

Note for the Mac press **option + enter**.

To run your application merely press on the Run icon  at the top of your IDE (assuming you have set up a particular AVD – ‘Android Virtual Device’)

A screenshot of a cell phone

Description automatically generated

**AndroidManifest.xml (One file needed per app)**

*<?***xml version="1.0" encoding="utf-8"***?>*<**manifest xmlns:android="http://schemas.android.com/apk/res/android"  
 xmlns:tools="http://schemas.android.com/tools"  
 package="com.example.jamespapademas.message"**>  
  
 <**application  
 android:allowBackup="true"  
 android:icon="@mipmap/ic\_launcher"  
 android:label="@string/app\_name"  
 android:roundIcon="@mipmap/ic\_launcher\_round"  
 android:supportsRtl="true"  
 android:theme="@style/AppTheme"  
 tools:ignore="AllowBackup,GoogleAppIndexingWarning"**>

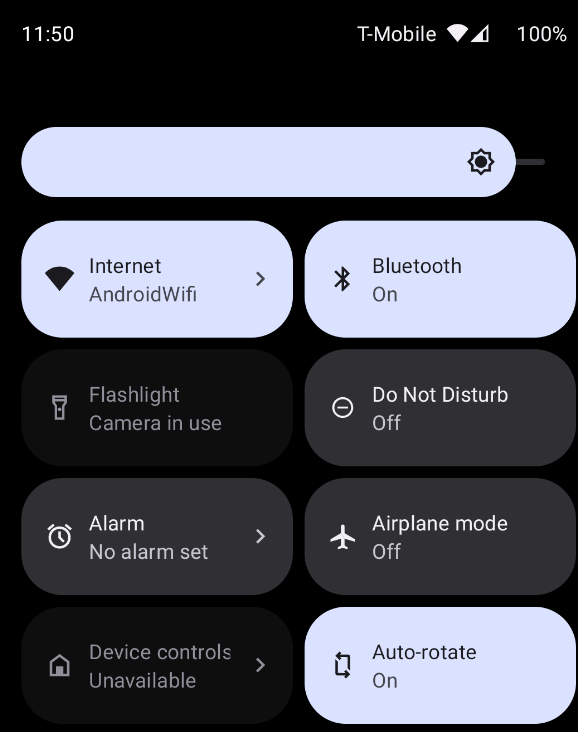
<**activity android:name=".MainActivity"**>  
 <**intent-filter**>  
 <**action android:name="android.intent.action.MAIN"** />  
  
 <**category**

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

</**application**>  
  
</**manifest**>

**Landscape vs. Portrait modalities support**

Click on the Auto-rotate icon (pull down from the top of your Emulator screen) to enable various rotation modes.

****

Click on one the following icons to ‘toggle’ views from your Emulator bar strip.

Portrait view

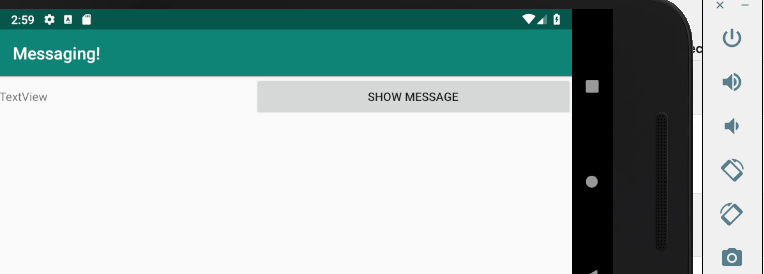
A screenshot of a cell phone

Description automatically generated

Rotate left

Rotate right

Landscape view



Snapshots anybody? Click camera icon for a png of your screen activity dropped to your desktop!



**Next week Chapter Readings:**

**Chap. 2** - More on getting started

**Chap. 3** - APPLICATIONS, ACTIVITIES, AND FRAGMENTS

**Categories to overview:**

App project designs

AVD “tweaks”

Activity Life Cycles (see ppt in PowerPoints folder)

Lab. 1 coming soon! - [Temperature App](https://play.google.com/store/apps/details?id=asav.roomtemprature)

*Next week discussion and practice exercise*

Create a payroll application.

Allow for three views (2 EditText views and 1 TextView)

Each EditView (shown as PlainText in your Palette) should either represent **hours** as an id and one as **rate** as an id. Your TextView should represent **result** as an id.

Sample setup

Layout zoom

Design Surface toggle modes

A screenshot of a cell phone

Description automatically generated

Various text views dragged in order from pallet in design view mode

Run time snapshot follows.

A screenshot of a cell phone

Description automatically generated

Desired UI action at run time

When the button is pressed, an action occurs to calculate the payroll (hours\*rate) and allow the result to appear in the TextView field.