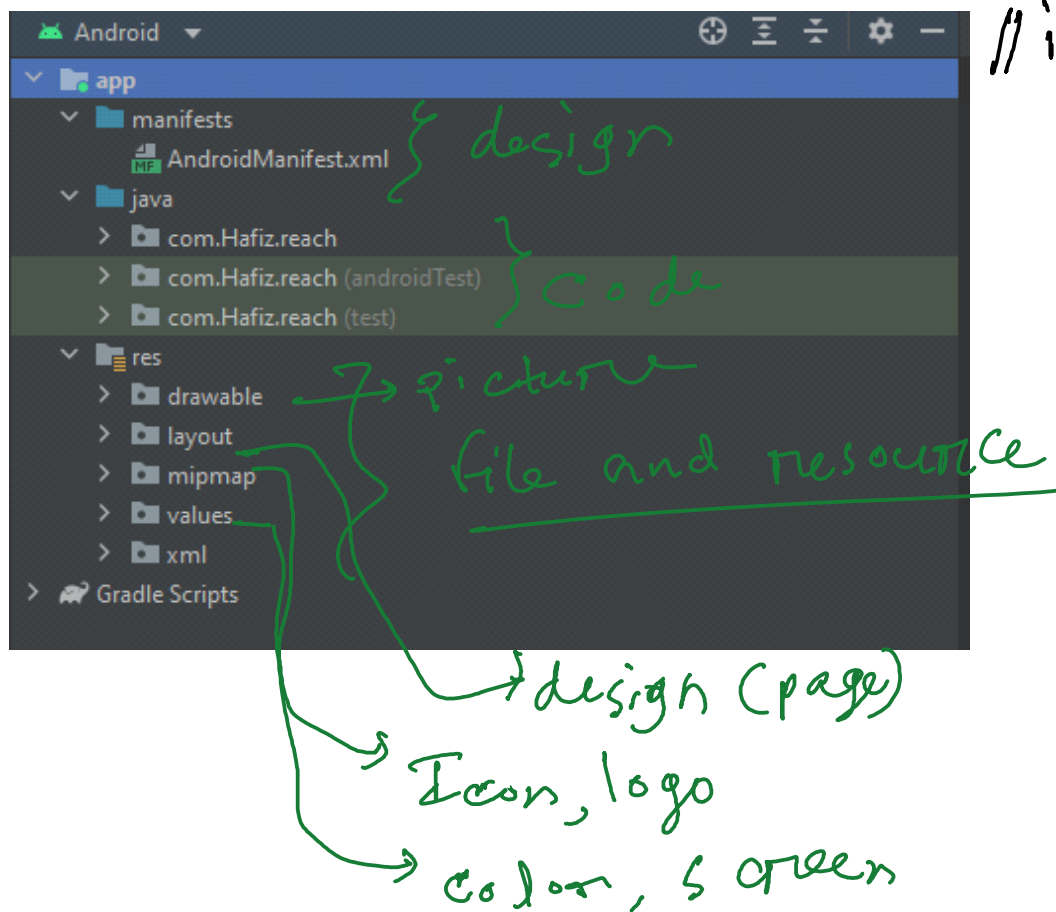


Layouts and file structure

Sunday, October 2, 2022 10:06 AM



Initial Problems-

Cannot run even "Hello world program!!"

Error -

```
java.lang.NullPointerException (no error message)
```

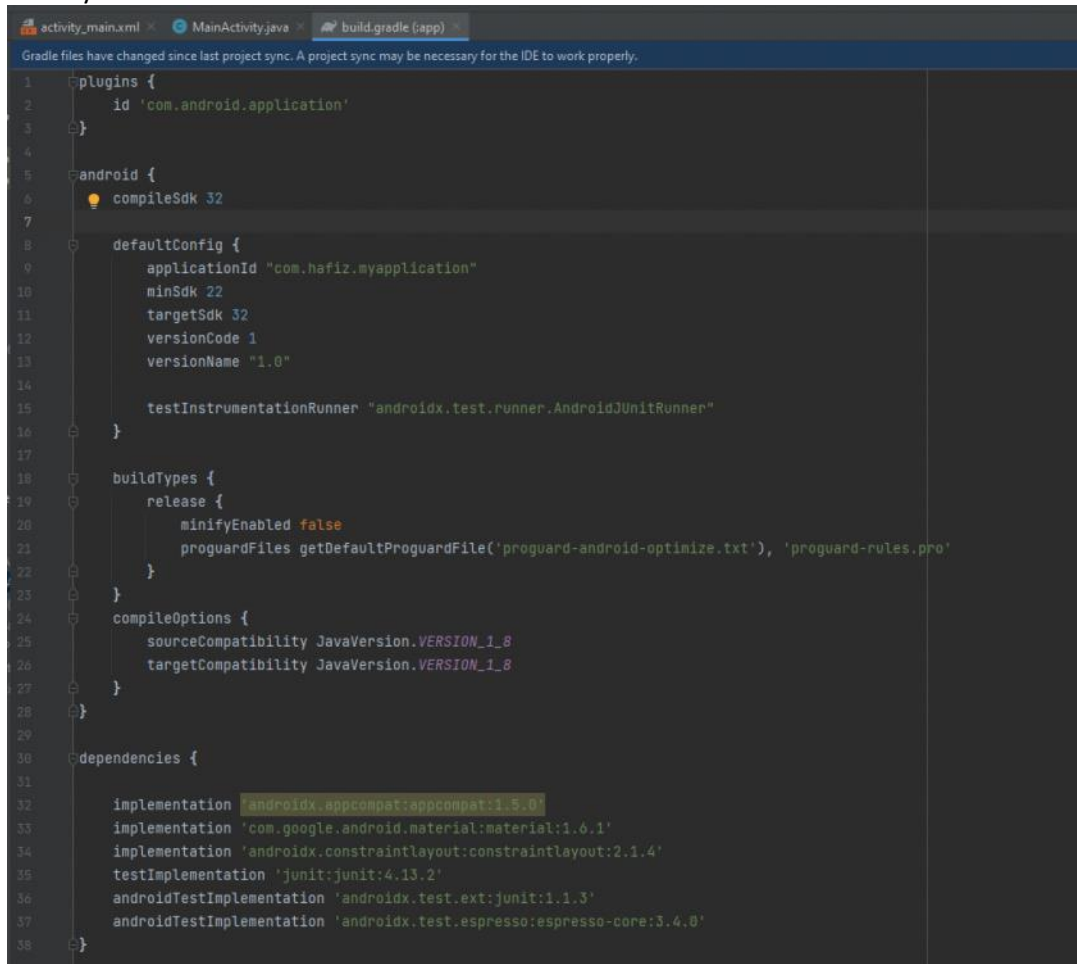
- Try: Run with --stacktrace option to get the stack trace. Run with --info or --debug option to get more log output. Run with --scan to get full insights.
- Get more help at <https://help.gradle.org>

BUILD FAILED in 7s Running Gradle task 'assembleDebug'... 8.8s Exception: Gradle task assembleDebug failed with exit code 1

Solution:

We need to update the '.gradle' file(module)

Initially the code looks like this:

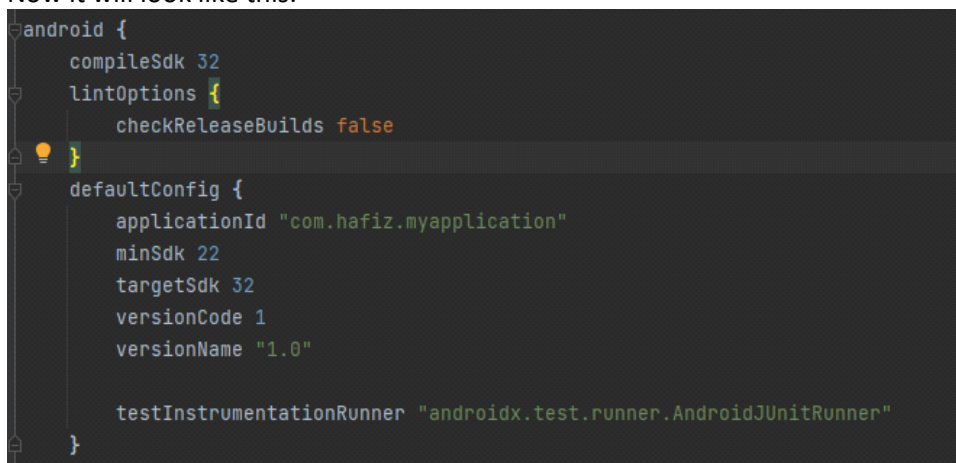


```
1 plugins {
2     id 'com.android.application'
3 }
4
5 android {
6     compileSdk 32
7
8     defaultConfig {
9         applicationId "com.hafiz.myapplication"
10        minSdk 22
11        targetSdk 32
12        versionCode 1
13        versionName "1.0"
14
15        testInstrumentationRunner "androidx.test.runner.AndroidJUnitRunner"
16    }
17
18    buildTypes {
19        release {
20            minifyEnabled false
21            proguardFiles getDefaultProguardFile('proguard-android-optimize.txt'), 'proguard-rules.pro'
22        }
23    }
24
25    compileOptions {
26        sourceCompatibility JavaVersion.VERSION_1_8
27        targetCompatibility JavaVersion.VERSION_1_8
28    }
29
30    dependencies {
31
32        implementation 'androidx.appcompat:appcompat:1.5.0'
33        implementation 'com.google.android.material:material:1.6.1'
34        implementation 'androidx.constraintlayout:constraintlayout:2.1.4'
35        testImplementation 'junit:junit:4.13.2'
36        androidTestImplementation 'androidx.test.ext:junit:1.1.3'
37        androidTestImplementation 'androidx.test.espresso:espresso-core:3.4.0'
38    }
39 }
```

We need to add this line on android class :

```
lintOptions {
    checkReleaseBuilds false
}
```

Now it will look like this:



```
1 android {
2     compileSdk 32
3     lintOptions {
4         checkReleaseBuilds false
5     }
6
7     defaultConfig {
8         applicationId "com.hafiz.myapplication"
9         minSdk 22
10        targetSdk 32
11        versionCode 1
12        versionName "1.0"
13
14        testInstrumentationRunner "androidx.test.runner.AndroidJUnitRunner"
15    }
16
17    buildTypes {
18        release {
19            minifyEnabled false
20            proguardFiles getDefaultProguardFile('proguard-android-optimize.txt'), 'proguard-rules.pro'
21        }
22    }
23
24    compileOptions {
25        sourceCompatibility JavaVersion.VERSION_1_8
26        targetCompatibility JavaVersion.VERSION_1_8
27    }
28
29    dependencies {
30
31        implementation 'androidx.appcompat:appcompat:1.5.0'
32        implementation 'com.google.android.material:material:1.6.1'
33        implementation 'androidx.constraintlayout:constraintlayout:2.1.4'
34        testImplementation 'junit:junit:4.13.2'
35        androidTestImplementation 'androidx.test.ext:junit:1.1.3'
36        androidTestImplementation 'androidx.test.espresso:espresso-core:3.4.0'
37    }
38 }
```

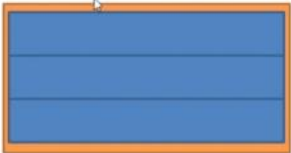
And hopefully solve the problem

Basics

Monday, October 3, 2022 5:50 PM

Container

LinearLayout
(Vertical)



LinearLayout
(Horizontal)



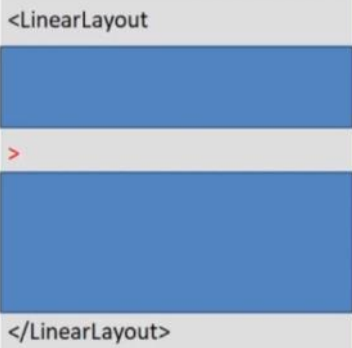
Element

Button



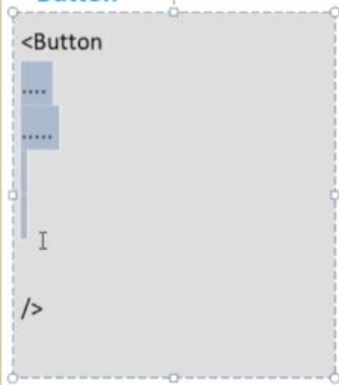
Container

LinearLayout



Element

Button



```

<?xml version="1.0" encoding="utf-8"?>
<androidx.constraintlayout.widget.ConstraintLayout xmlns:android="
    xmlns:app="http://schemas.android.com/apk/res-auto"
    xmlns:tools="http://schemas.android.com/tools"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    tools:context=".MainActivity"
    >

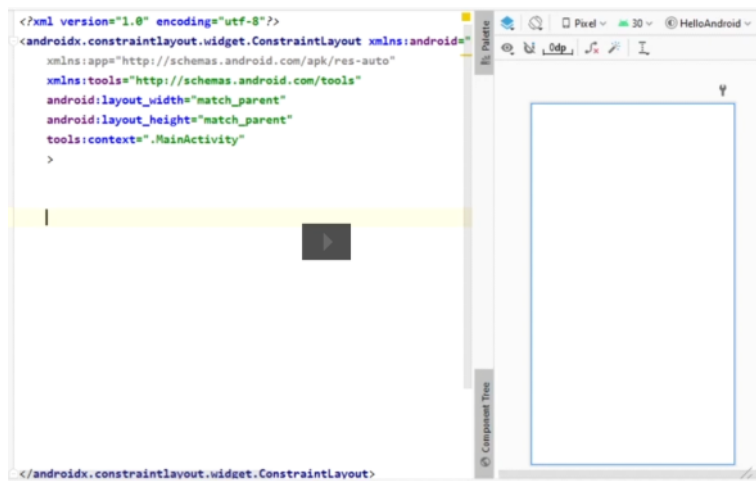
    <TextView
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="Hello! This is my first app"
        app:layout_constraintBottom_toBottomOf="parent"
        app:layout_constraintLeft_toLeftOf="parent"
        app:layout_constraintRight_toRightOf="parent"
        app:layout_constraintTop_toTopOf="parent" />

</androidx.constraintlayout.widget.ConstraintLayout>

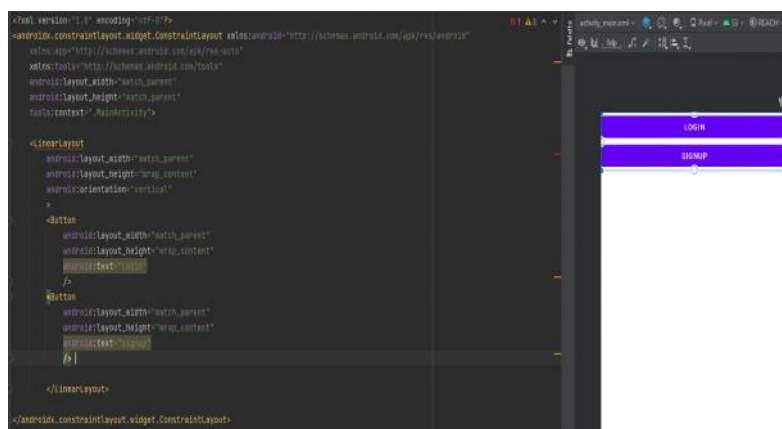
```

// Container

//Element



// ConstraintLayout for Initial size of the app
// we can also use LinearLayout instead

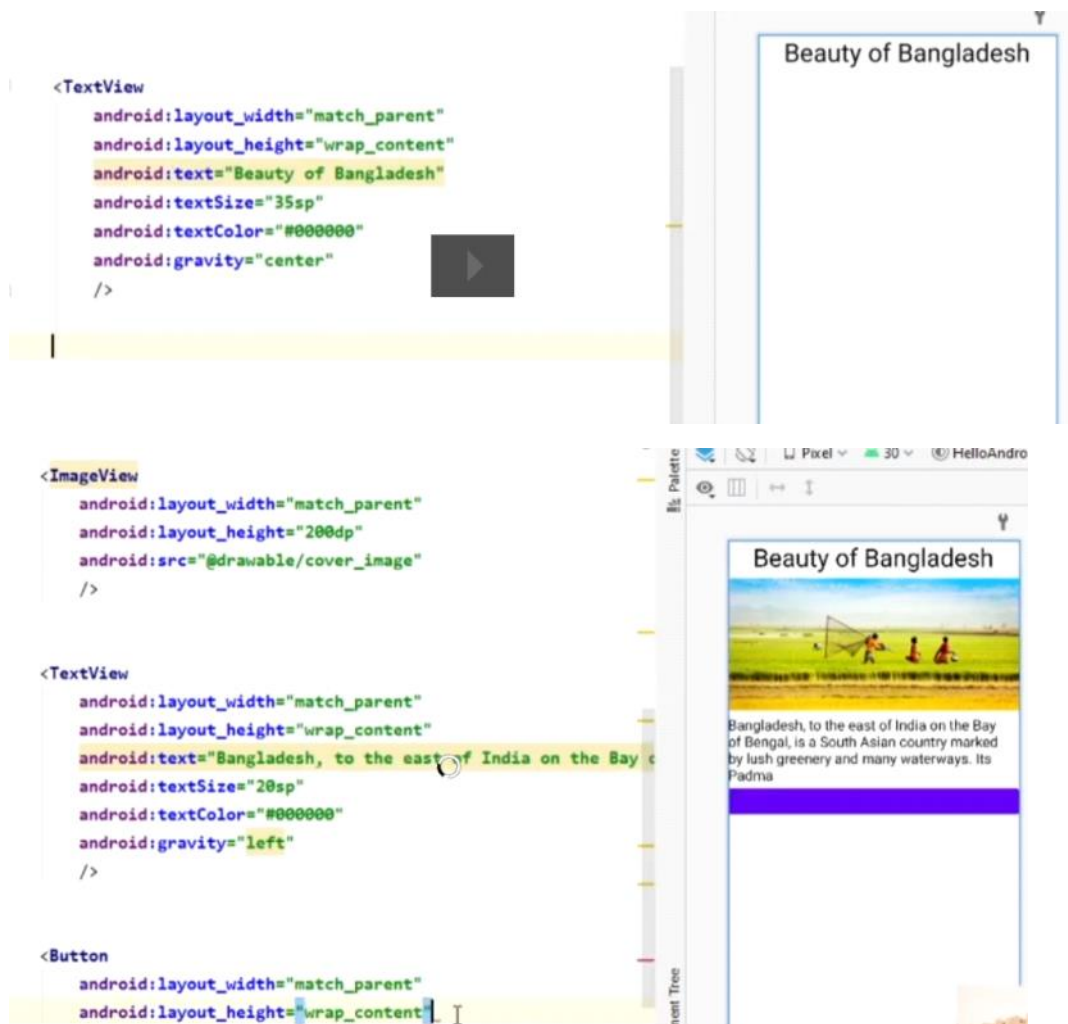


// Horizontal - elements will be added Side by side;
// vertical - " " up to down

// Match Parent - Full screen (According to initial height and width)
// Wrap content - The actual space needed for an element
// More elements- TextView , imageView

Textview and Imageview

Monday, October 3, 2022 7:32 PM



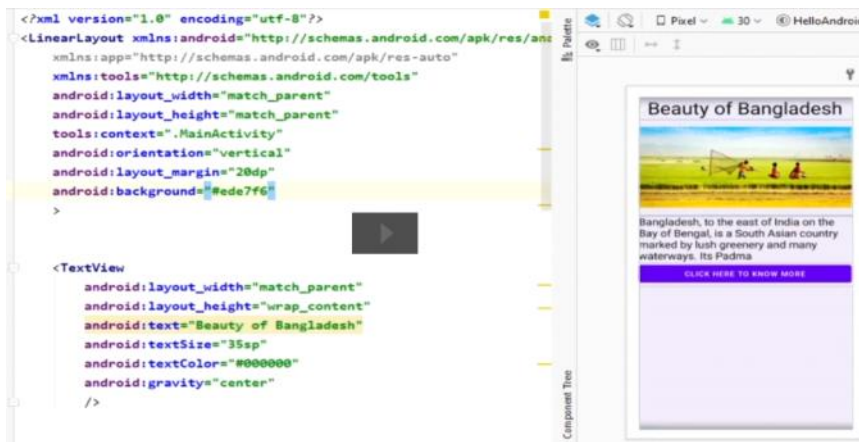
// ScaleType : centerCrop --> fill with the image

More properties:

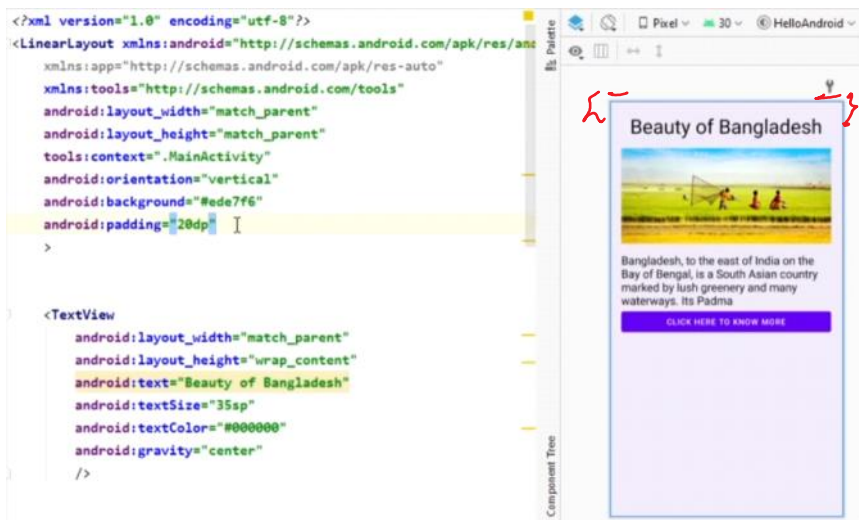
1.Margin

- > can be used in every element
- > variations
 - > marginTop
 - > marginRight/left etc

2.Background



3.Padding



WeightSum and its use

Monday, October 3, 2022 8:10 PM

Suppose we want to ADD three button like this



How can we achieve?

Step 1:

WeightSum : the some of total weights of the buttons

```
<LinearLayout
    android:layout_width="match_parent"
    android:layout_height="100dp"
    android:orientation="horizontal"
    android:background="#FFFFFF"
    android:weightSum="90" // 3 X 30
/>
```

Step 2:

```
<Button
    android:id="@+id/button3"
    android:layout_width="fill_parent"
    android:layout_height="match_parent"
    android:text="Button"
    android:layout_weight="30" // Weight = 30 (each)
/>
```

// fill_parent instead of match_parent

// Weight = 30 (each)

$$3 \times 30 = 90$$

ScrollView

Tuesday, October 4, 2022 1:17 PM

- > Scroll View is a container
- > It Enables us to scroll the elements under it
- > But it cannot work without vertical linear layout

```
<ScrollView
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    >

    <LinearLayout
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:orientation="vertical"
        >

        </LinearLayout>
```

elements

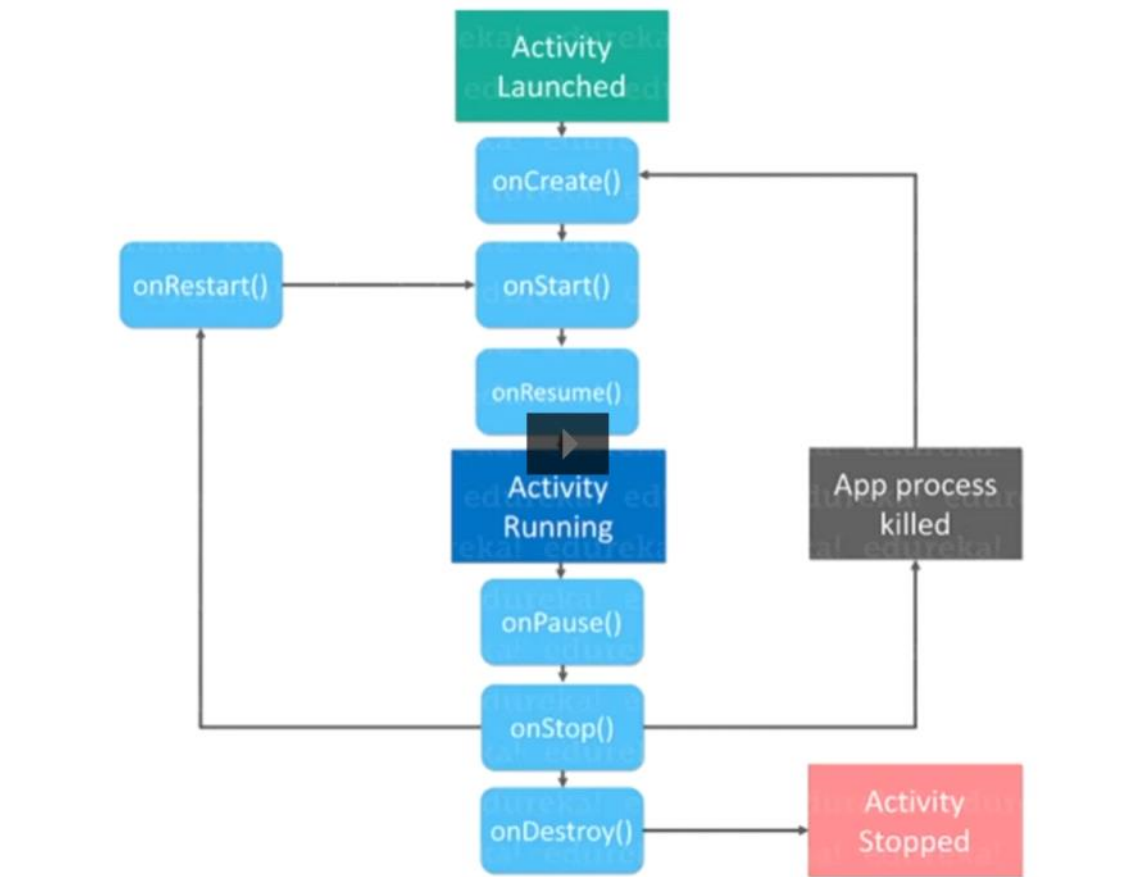
// Here unlimited elements

```
</ScrollView>
```


Life Cycle

Wednesday, October 5, 2022

12:44 AM



OnCreate() - Start The app(initially)

OnPause() - Minimize

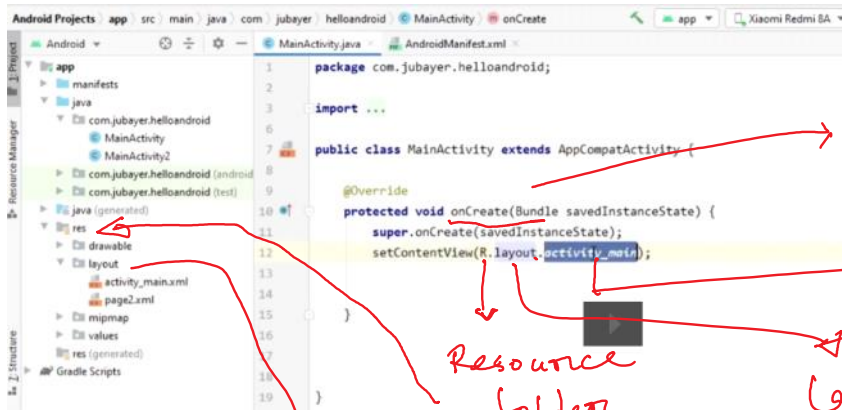
OnResume() - After Minimize

OnRestart() - Restart

OnDestroy()- Closing the app

Main Activity

Wednesday, October 5, 2022 3:00 PM



oncreate()

activity-main.xml

Resource folder

Layout folder

What will run initially? and why?

- MainActivity.java and it will call activity-main.xml

How to connect xml with java?

Wednesday, October 5, 2022 3:45 PM

-> Every Element Should be given an id
Like this:

```
<Button  
    android:id="@+id/myButton"  
    android:layout_width="match_parent"  
    android:layout_height="wrap_content"  
    android:layout_margin="20dp"  
    android:text="Click Here"  
    android:textSize="35sp"  
>
```



Then Connect:

```
public class MainActivity extends AppCompatActivity {  
  
    @Override  
    protected void onCreate(Bundle savedInstanceState) {  
        super.onCreate(savedInstanceState);  
        setContentView(R.layout.activity_main);  
  
        Button clikButton;  
        TextView tvDisplay;  
  
        clikButton = findViewById(R.id.myButton);  
  
    }  
}
```

OnClickListener

Wednesday, October 5, 2022 3:53 PM

```
кликButton.setOnClickListener(new View.OnClickListener() {  
    @Override  
    public void onClick(View v) {  
  
        |  
  
        //desired code here  
  
    }  
});
```

```
MainActivity.java activity_main.xml  
10 public class MainActivity extends AppCompatActivity {  
11  
12     @Override  
13     protected void onCreate(Bundle savedInstanceState) {  
14         super.onCreate(savedInstanceState);  
15         setContentView(R.layout.activity_main);  
16  
17         Button clickButton;  
18         TextView tvDisplay;  
19  
20         clickButton = findViewById(R.id.clickButton);  
21         tvDisplay = findViewById(R.id.tvDisplay);  
22  
23         clickButton.setOnClickListener(new View.OnClickListener() {  
24             @Override  
25             public void onClick(View v) {  
26  
27                 tvDisplay.setText("Hello. Thank you! Its working");  
28             }  
29         });  
30  
31  
32  
33  
34
```

//if we click on
Button, it will show
the text.

Visibility And Toast

Wednesday, October 5, 2022 4:25 PM

-> Every Element and container have a property - Visibility

-> It has 3 value : Visible, invisible, Gone

Visible :

Invisible : exist on screen but not shown

Gone : Out of the screen

```
<ImageView  
    android:id="@+id/image"  
    android:layout_width="match_parent"  
    android:layout_height="300dp"  
    android:src="@drawable/cover_image"  
    android:scaleType="centerCrop"  
    android:visibility="visible"  
/>
```

In Java Code

```

com / jubayer / helloandroid / MainActivity / onCreate
MainActivity.java activity_main.xml
20 protected void onCreate(Bundle savedInstanceState) {
21     super.onCreate(savedInstanceState);
22     setContentView(R.layout.activity_main);
23
24     image = findViewById(R.id.image);
25     bGone = findViewById(R.id.bGone);
26     bShow = findViewById(R.id.bShow);
27     mainLayout = findViewById(R.id.mainLayout);
28
29
30     bShow.setOnClickListener(new View.OnClickListener() {
31         @Override
32         public void onClick(View v) {
33
34             image.setVisibility(View.VISIBLE);
35         }
36     });
37
38     bGone.setOnClickListener(new View.OnClickListener() {
39         @Override
40         public void onClick(View v) {
41
42             image.setVisibility(View.GONE);
43             mainLayout.setVisibility(View.GONE);

```

Toast:

Toast is Basically a PopUp message from the system (Like 'Invalid Input' , 'Try again')

```

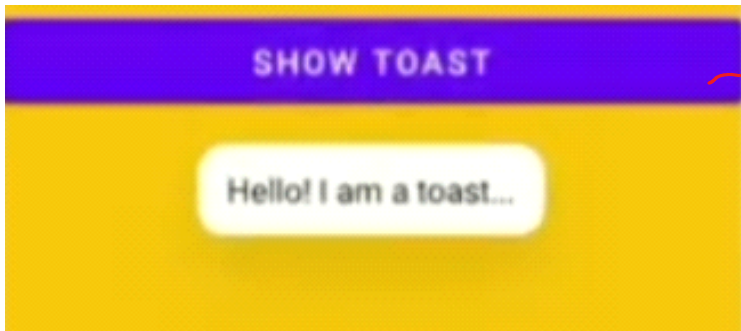
bToast.setOnClickListener(new View.OnClickListener() {
    @Override
    public void onClick(View v) {
        // code here
        Toast.makeText(context: MainActivity.this, text: "Toast is showing here...", Toast.LENGTH_LONG).show();
    }
});

```

Handwritten annotations:

- Button to be clicked (pointing to `bToast`)
- Message (pointing to `"Toast is showing here..."`)
- duration (pointing to `Toast.LENGTH_LONG`)
- ↓ class (pointing to `context: MainActivity.this`)
- ↓ current class (pointing to `MainActivity.this`)

SHOW TOAST → Button



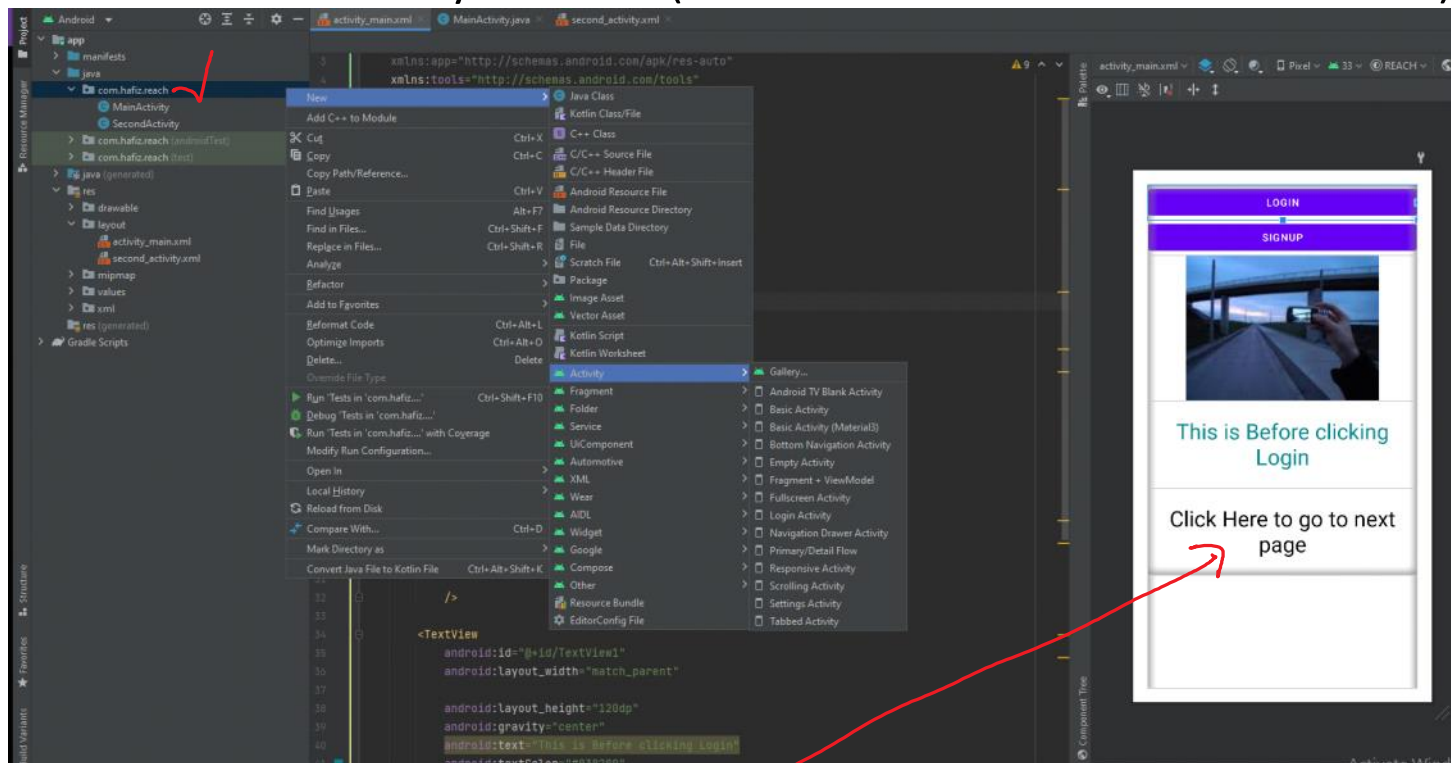
→ Button

How to Connect One Page with another?

Wednesday, October 5, 2022 4:36 PM

First Select an element which will be used to switch the page.

-> Create New Activity Java File (also xml file will be created with this)



-> Select an element which will be used for switching
say this TextView

-> Then In The current JavaFile (MainActivity.java)

"Intent" Class is used For Switching


```
TextView tb2;  
tb2 = findViewById(R.id.TextView3);  
tb2.setOnClickListener(new View.OnClickListener() {  
    @Override  
    public void onClick(View view) {  
        Intent myintent = new Intent( packageContext: MainActivity.this , SecondActivity.class);  
        startActivity(myintent);  
    }  
});
```

Convert website into app

Thursday, October 6, 2022 3:37 AM

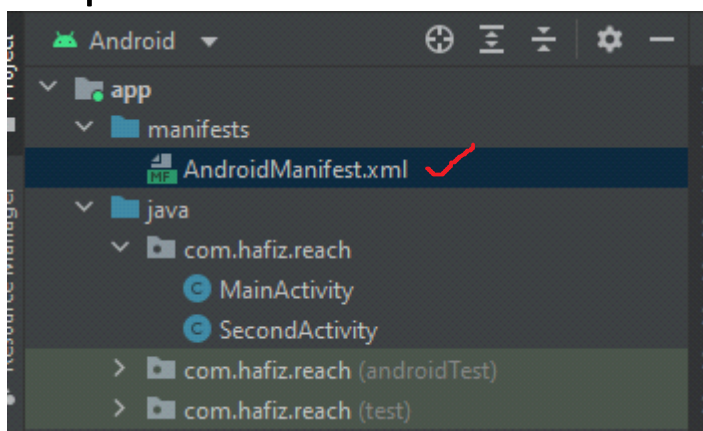
Step -1 : WebView;

```
<!--  
<WebView  
    android:id="@+id/Web"  
    android:layout_width="match_parent"  
    android:layout_height="match_parent"  
-->
```

Step 2:

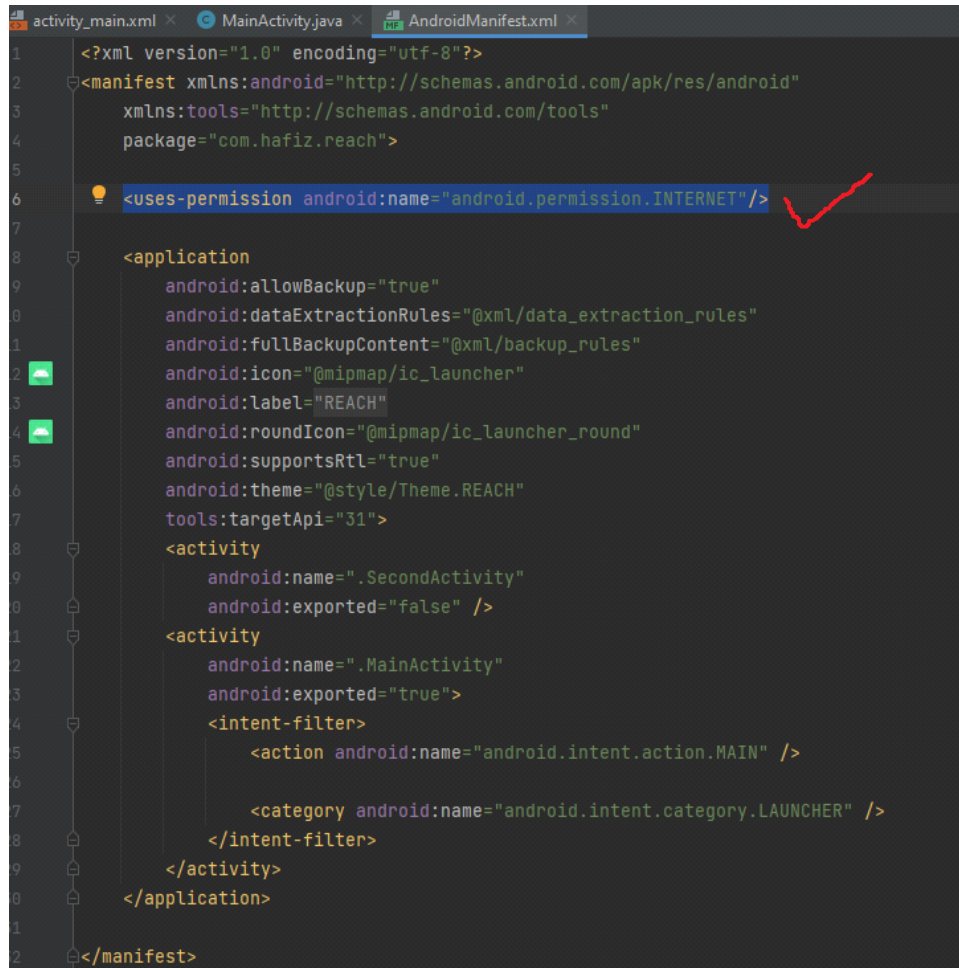
```
WebView wb = findViewById(R.id.Web);  
wb.getSettings().getJavaScriptEnabled();  
wb.loadUrl("https://bongoacademy.com/");
```

Step 3:



Add this line for Internet permission

```
<uses-permission android:name="android.permission.INTERNET"/>
```

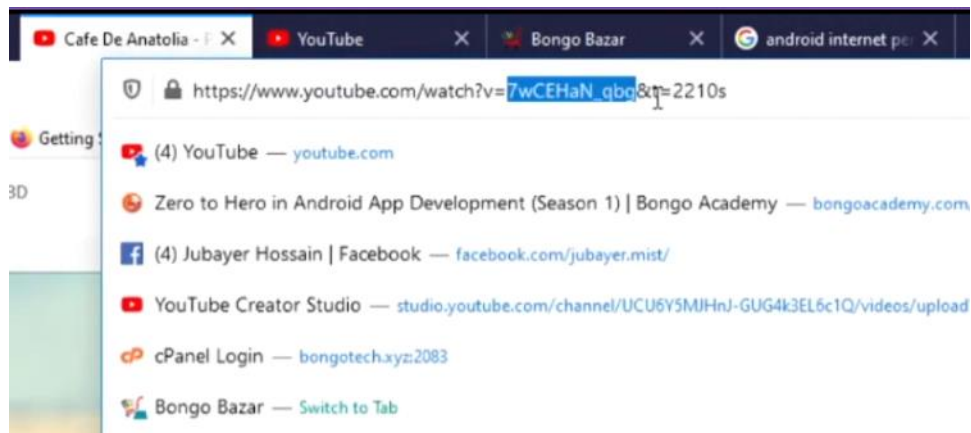


All Set now!!

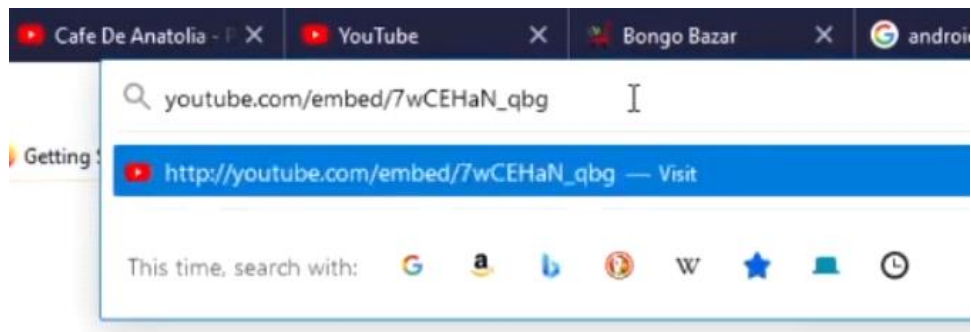
For Playing Video in App !

```
mWebView.getSettings().setJavaScriptEnabled(true);
mWebView.loadUrl("https://www.youtube.com/embed/7wCEHaN_qbg");
```

Change the link In this way:



Then:

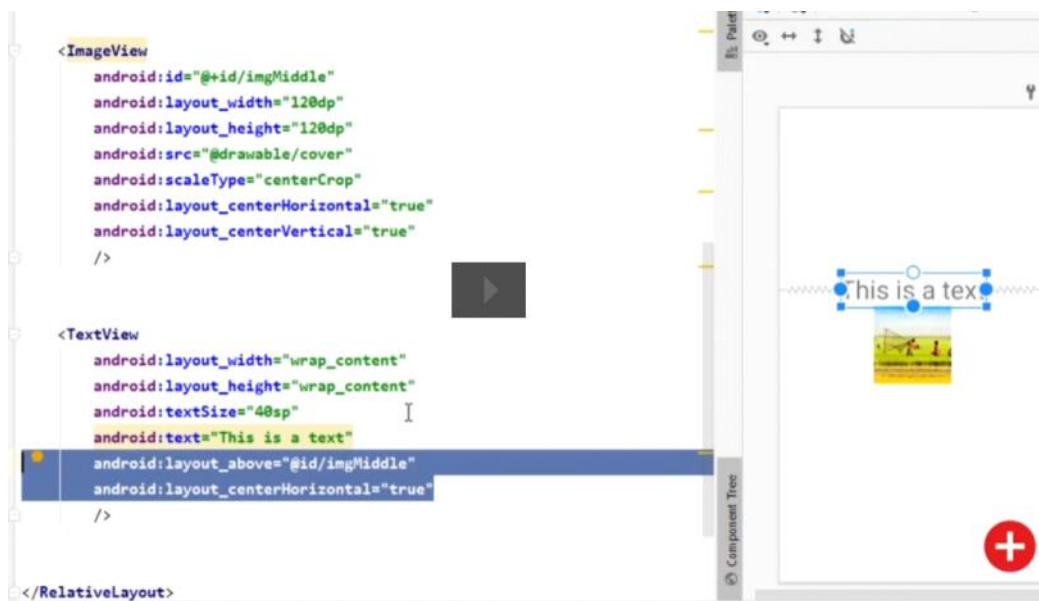
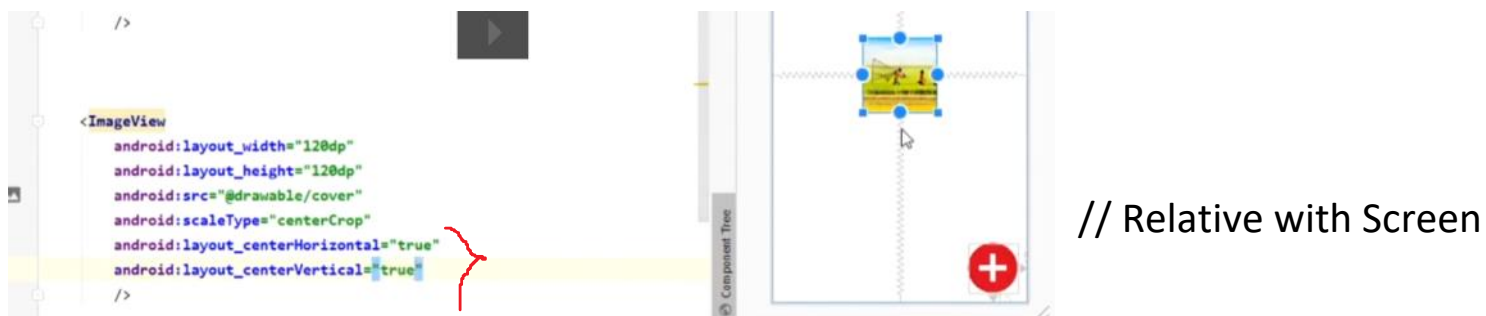
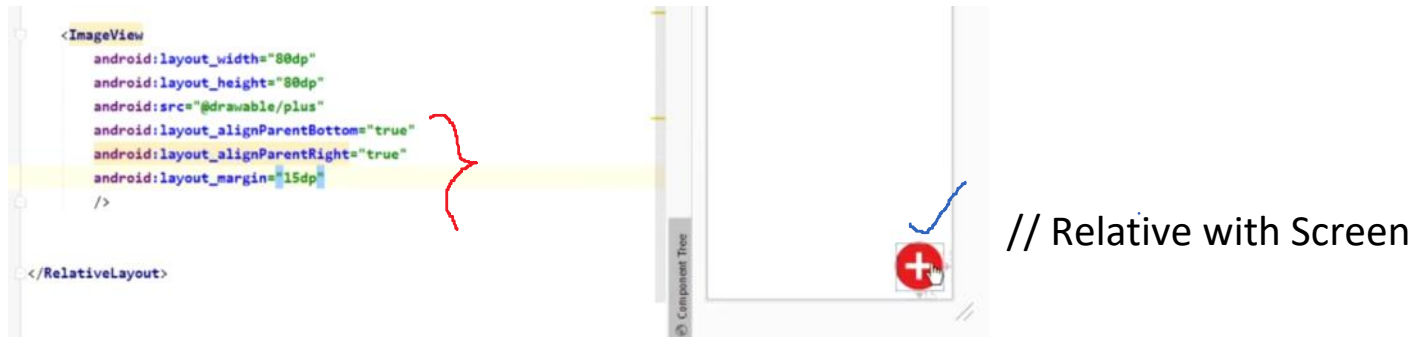


Relative Layout

Thursday, October 6, 2022 3:37 AM

1.Relative with screen

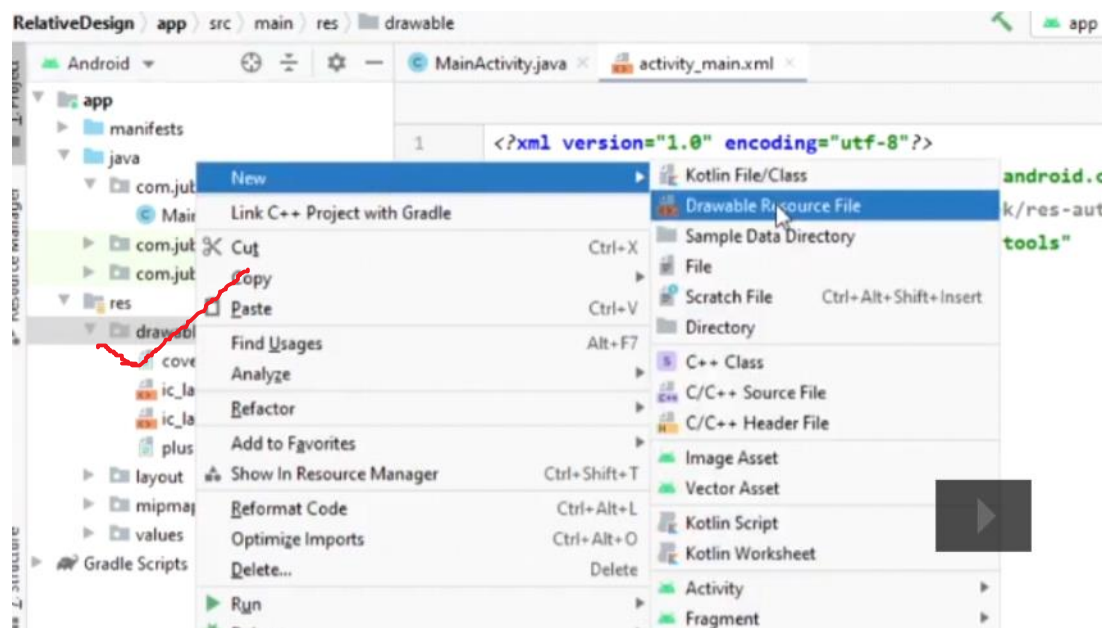
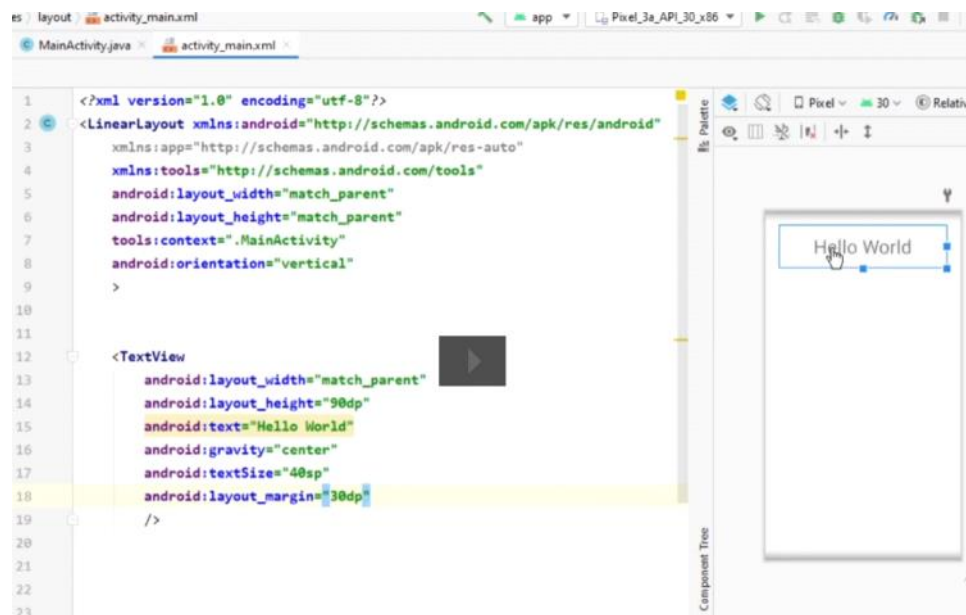
2.Relative with another layout

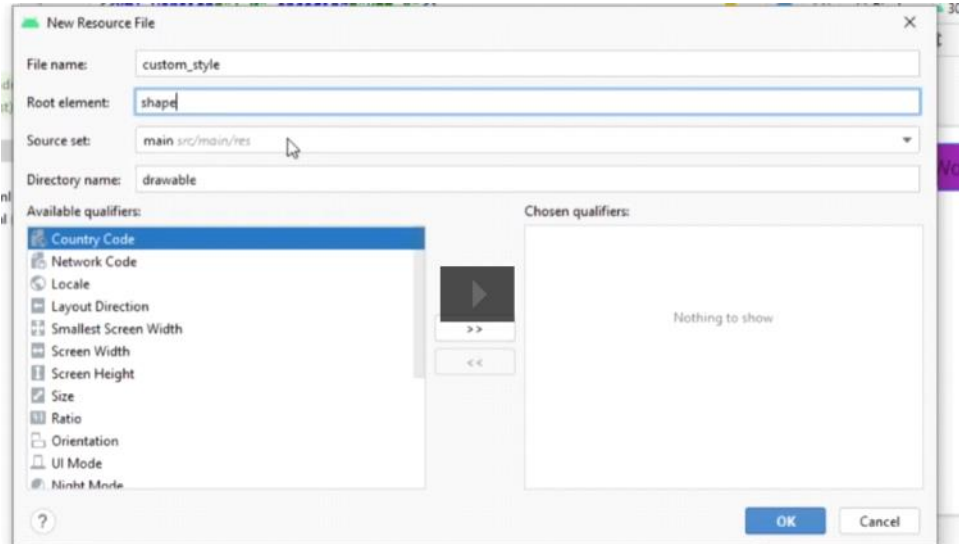
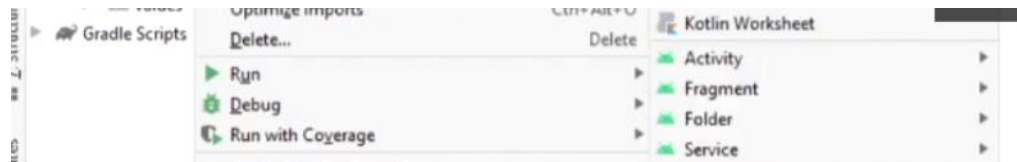


Custom Drawable

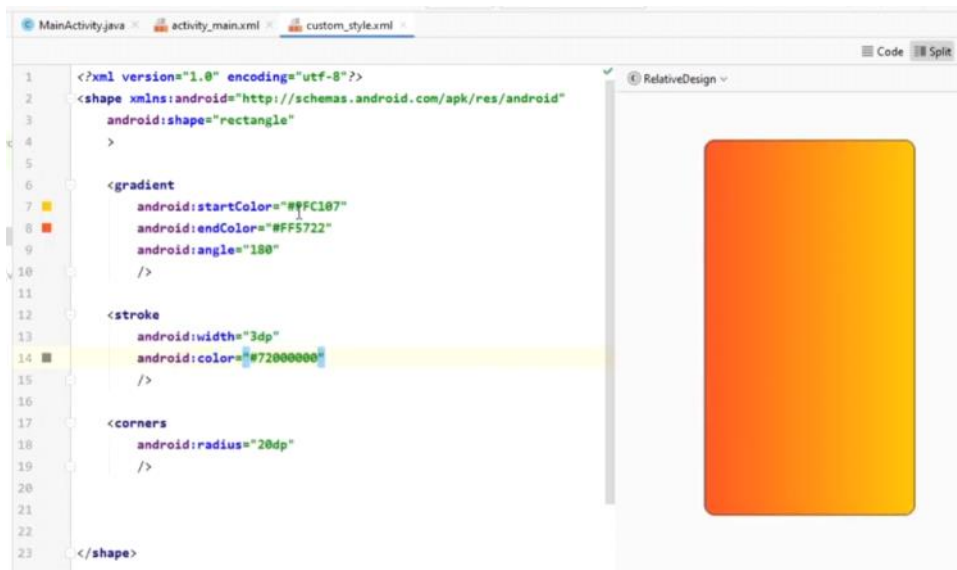
Sunday, October 9, 2022 2:11 PM

- Any Layout ,Element , Container Can be Customized
- Now we Can Use by Default BG
- Or We can Create a separate custom .xml file
For custom bacground and shape





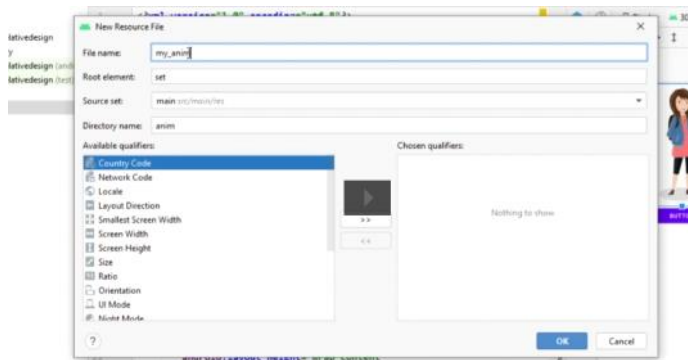
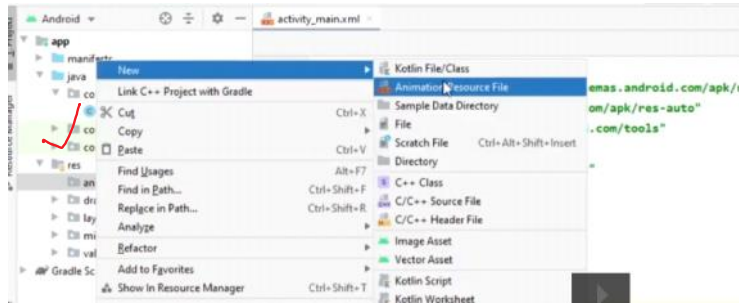
```
<TextView
    android:layout_width="match_parent"
    android:layout_height="90dp"
    android:text="Hello World"
    android:gravity="center"
    android:textSize="40sp"
    android:layout_margin="30dp"
    android:background="@drawable/custom_style"
/>
```



Animations

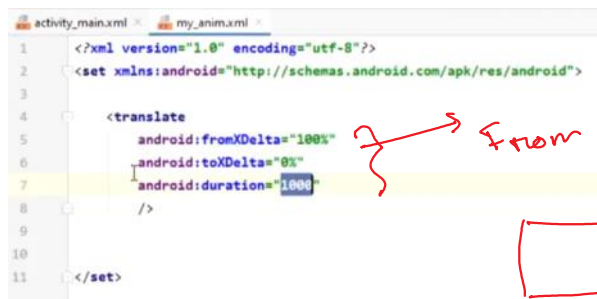
Same as the Drawable

-we can create a different Folder (directory) for this



NOW:

Create animation file



From which position? (x axis)
100% -> Screen
0% -> image
100% -> image

To Animate:



Animation class

... a class object
... anim.xml

```

15     ImageView imageView;
16     Animation myAnim;
17
18     @Override
19     protected void onCreate(Bundle savedInstanceState) {
20         super.onCreate(savedInstanceState);
21         setContentView(R.layout.activity_main);
22
23         button = findViewById(R.id.button);
24         imageView = findViewById(R.id.imageView);
25         myAnim = AnimationUtils.loadAnimation(context: MainActivity.this, R.anim.my_anim);
26
27
28         button.setOnClickListener(new View.OnClickListener() {
29             @Override
30             public void onClick(View v) {
31                 // animation code here
32             }
33         });
34     }
35 }

```

Animation class

current java class obje-

Animation file (my_anim.xml)

helper

like are +

```

button.setOnClickListener(new View.OnClickListener() {
    @Override
    public void onClick(View v) {
        // animation code here

        imageView.startAnimation(myAnim);
    }
});

```

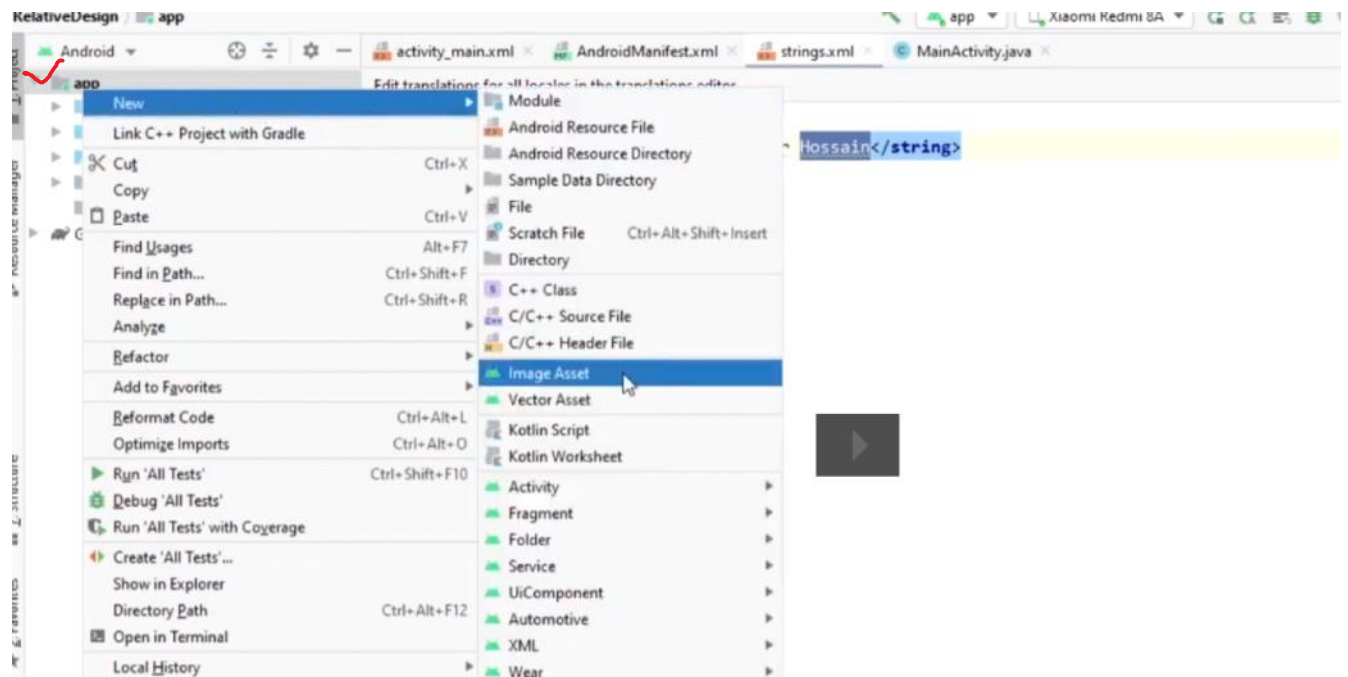
// start

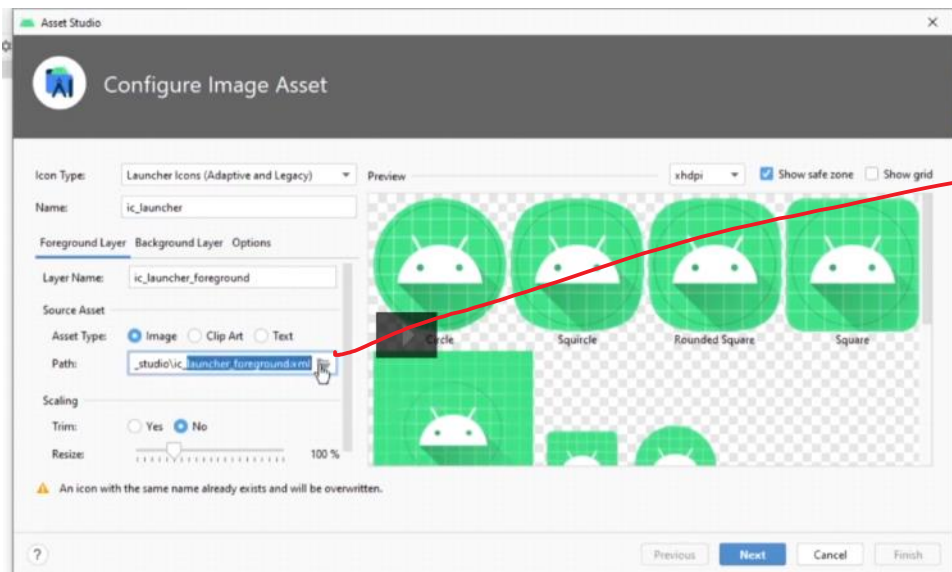
Change name and icon

Monday, October 10, 2022 10:38 AM

-> How to Change name of the app
Values-> strings -> (change the name)

Change Icon:



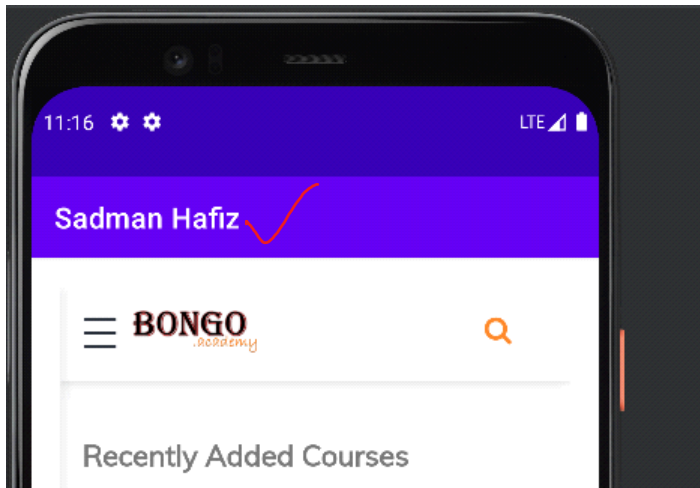


Select
a picture

Work with action Bar

Monday, October 10, 2022 11:15 AM

We can keep , or modify the initial action bar;



For this we need to go :

themes.xml

We can go there in two ways;

1.

-> Go to manifest.xml;

->

```

1 <?xml version="1.0" encoding="utf-8"?>
2 <manifest xmlns:android="http://schemas.android.com/apk/res/android"
3   xmlns:tools="http://schemas.android.com/tools"
4   package="com.hafiz.reach">
5
6   <uses-permission android:name="android.permission.INTERNET"/>
7
8   <application
9       android:allowBackup="true"
10      android:dataExtractionRules="@xml/data_extraction_rules"
11      android:fullBackupContent="@xml/backup_rules"
12      android:icon="@mipmap/ic_launcher"
13      android:label="@string/app_name"
14      android:roundIcon="@mipmap/ic_launcher_round"
15      android:supportRtl="true"
16      android:theme="@style/Theme.REACH"
17      tools:targetApi="31">
18       <activity
19           android:name=".SecondActivity"
20           android:exported="false" />
21       <activity
22           android:name=".MainActivity"
23           android:exported="true">
24           <intent-filter>
25             <action android:name="android.intent.action.MAIN" />
26
27             <category android:name="android.intent.category.LAUNCHER" />
28           </intent-filter>
29         </activity>
30     </application>
31
32 </manifest>

```

→ control + mouse click

->Now

```

1 <resources xmlns:tools="http://schemas.android.com/tools">
2   <!-- Base application theme. -->
3   <style name="Theme.REACH" parent="Theme.MaterialComponents.DayNight.DarkActionBar">
4       <!-- Primary brand color. -->
5       <item name="colorPrimary">@color/purple_500</item>
6       <item name="colorPrimaryVariant">@color/purple_700</item>
7       <item name="colorOnPrimary">@color/white</item>
8       <!-- Secondary brand color. -->
9       <item name="colorSecondary">@color/teal_200</item>
10      <item name="colorSecondaryVariant">@color/teal_700</item>
11      <item name="colorOnSecondary">@color/black</item>
12      <!-- Status bar color. -->
13      <item name="android:statusBarColor" tools:targetApi="31">?attr/colorPrimaryVariant</item>
14      <!-- Customize your theme here. -->
15   </style>
16 </resources>

```

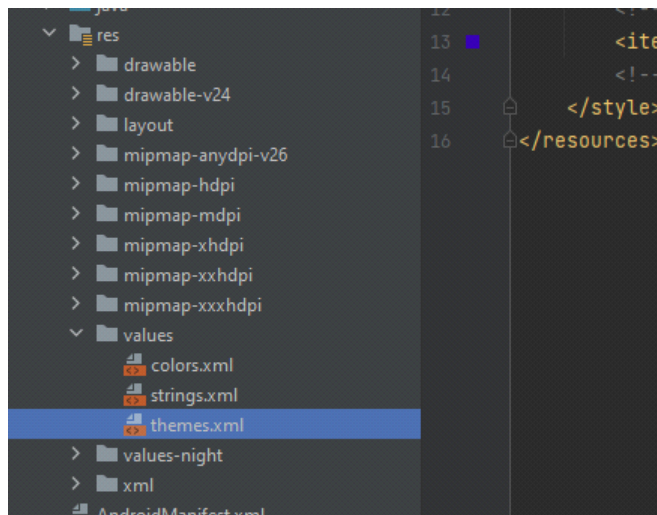
→ Select a theme or no theme

```

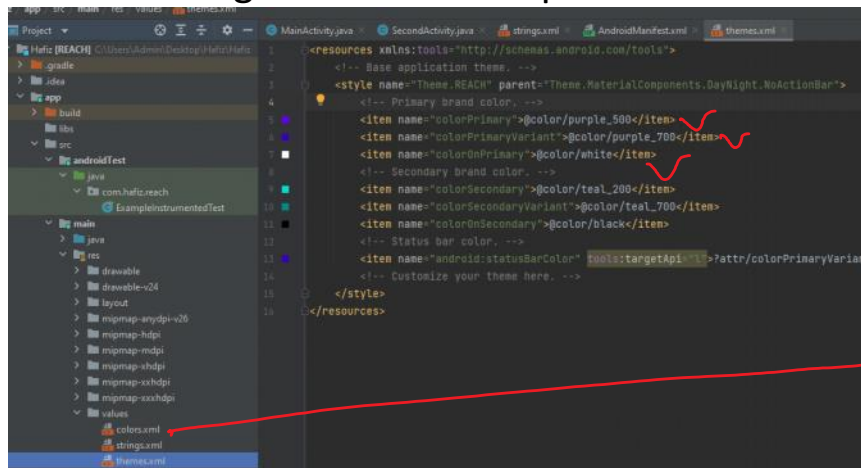
1 <resources xmlns:tools="http://schemas.android.com/tools">
2   <!-- Base application theme. -->
3   <style name="Theme.REACH" parent="Theme.MaterialComponents.DayNight.NoActionBar">
4       <!-- Primary brand color. -->
5       <item name="colorPrimary">@color/purple_500</item>
6       <item name="colorPrimaryVariant">@color/purple_700</item>
7       <item name="colorOnPrimary">@color/white</item>
8   </style>
9 </resources>

```

2.WE can directly go :



-> We can Change the color of top of the screen also



Colors
are defined
here

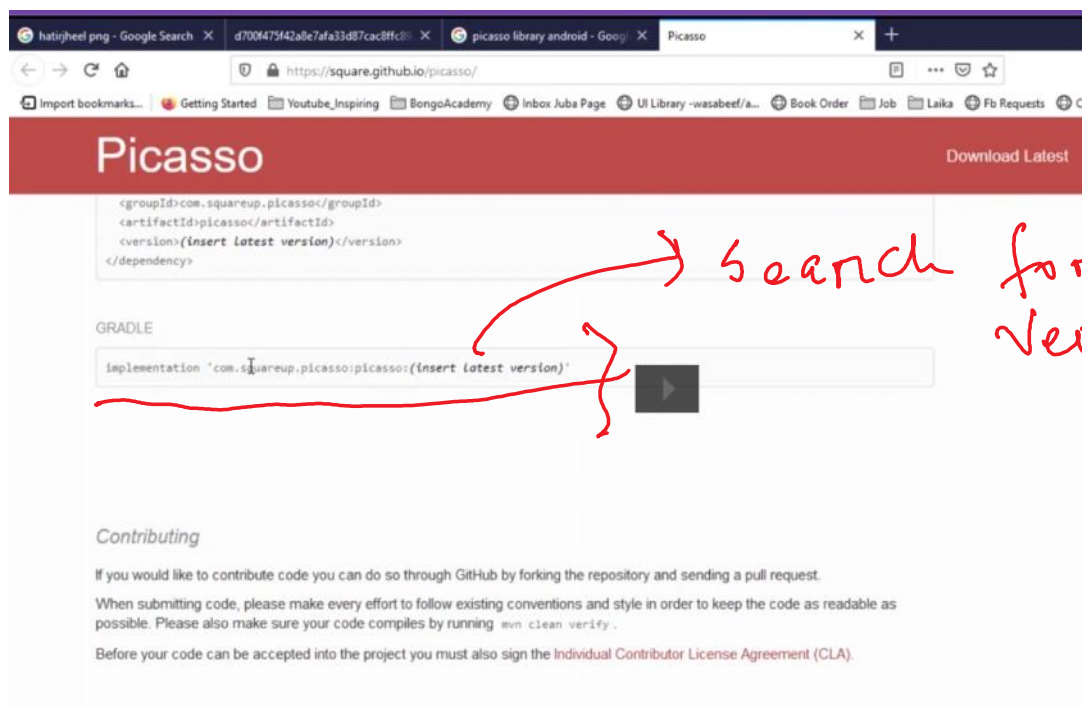
Picasso - For web Image

Monday, October 10, 2022 2:38 PM

When we want to insert an image from remote website , we can use through the Picasso Library

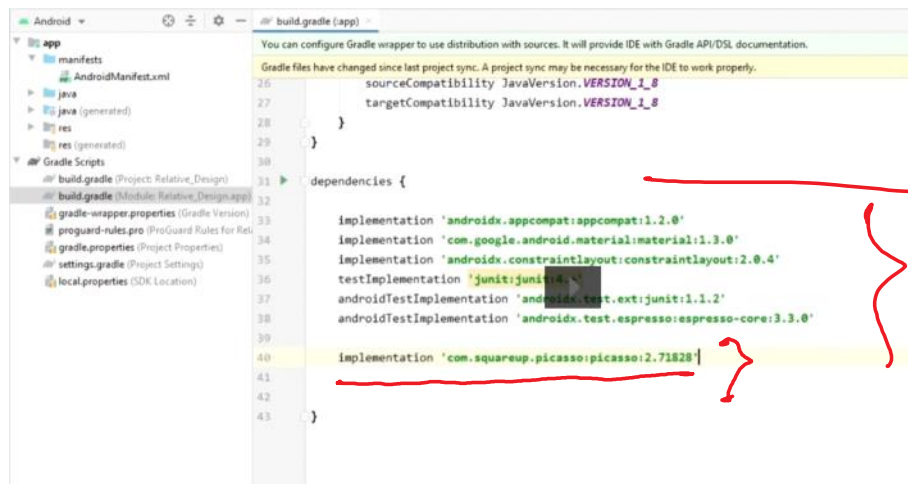
How to install:

Go to : <https://square.github.io/picasso/>



For installing any library in android studio:
We need to add a line of code in grade(app) file

Insert the code with latest version

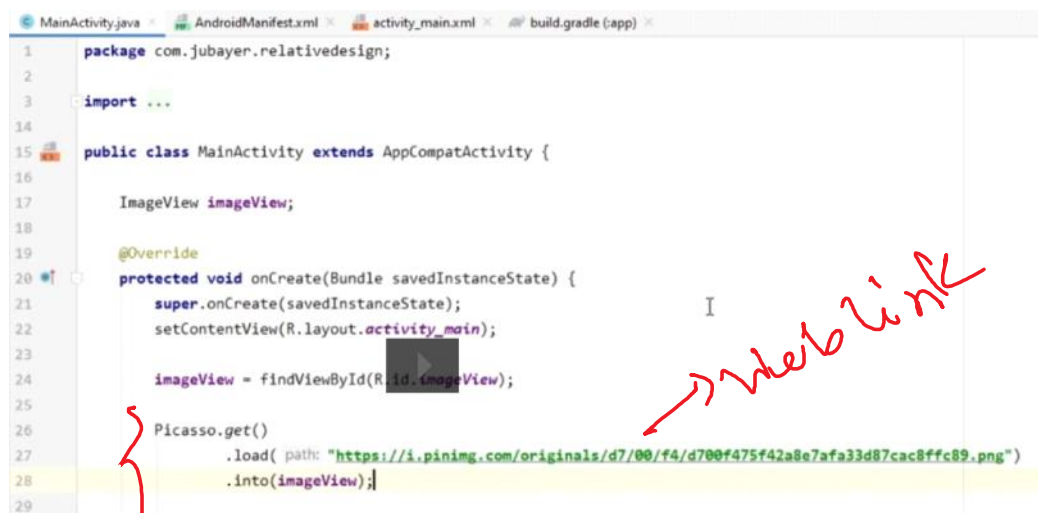


It Will ask for Sync(install automatically)



AS It will use internet , we need to permit internet access (manifest.xml)-explained earlier

NOW:



```
26 Picasso.get()  
27 .load( path: "https://i.pinimg.com/originals/d7/00/f4/d700f475f42a8e7afa33d87cac8ffc89.png")  
28 .into(imageView);  
29  
30  
31  
32  
33 }  
34 }
```

```
Picasso.get().load("http://i.imgur.com/DvpvklR.png").into(imageView);
```

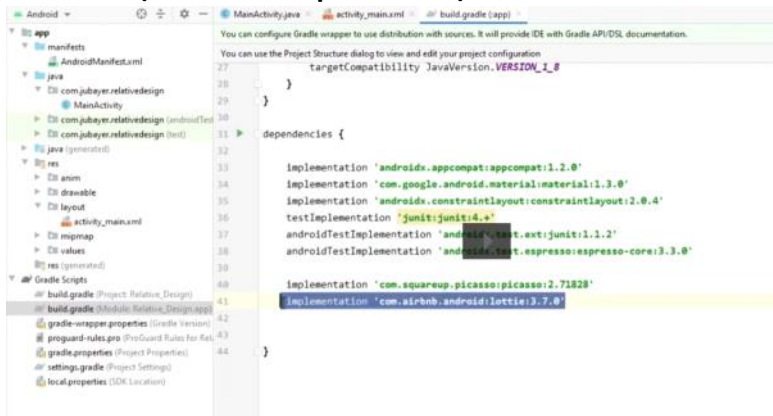
Lottie - For advance built-in animation

Monday, October 10, 2022 2:52 PM

It Provides some good free animation for using in app.

Go to : <https://github.com/airbnb/lottie-android>

Install:(same as picasso)



Click sync

Now, go to <https://lottiefiles.com/blog/working-with-lottie/getting-started-with-lottie-animations-in-android-app/>

(login with account/gmail)

Directly add to the layOut xml file (like as element)

```
<com.airbnb.lottie.LottieAnimationView  
    android:id="@+id/animationView"  
    android:layout_width="match_parent"  
    android:layout_height="wrap_content"  
    app:lottie_url="REPLACE_JSON_URL"
```

→ by default
we can change the
url

```
android:layout_height="wrap_content"  
app:lottie_url="REPLACE_JSON_URL"  
app:lottie_autoPlay="true"  
app:lottie_loop="true"/>
```

codes

We can use lottie animation file in two ways-

1. Direct Web link from <https://lottiefiles.com>
 - > select a animation, a link will be given with that file
 - > Use the link here
 - > (internet required)
2. Download .json file
 - > Create a new directory (like as "raw") in resource folder
 - > paste the downloaded file

Now

Add a new tag :

```
<com.airbnb.lottie.LottieAnimationView  
    android:id="@+id/animationView"  
    android:layout_width="match_parent"  
    android:layout_height="wrap_content"  
    app:lottie_url="REPLACE_JSON_URL"  
    app:lottie_autoPlay="true"  
    app:lottie_loop="true"  
    app:lottie_rawRes="@raw/giftbox"  
/>
```

That will do , Animation will work

-> We can also use animation with button click

In java file

Add :

```
LottieAnimationView lottieView;
```

```
lottieView = findViewById(R.id.lottieView);
```

As usual

Now

On buttonclick listener:

```
lottieView.setAnimation(R.raw.man);  
lottieView.playAnimation();  
lottieView.loop(true);
```

→ jason file