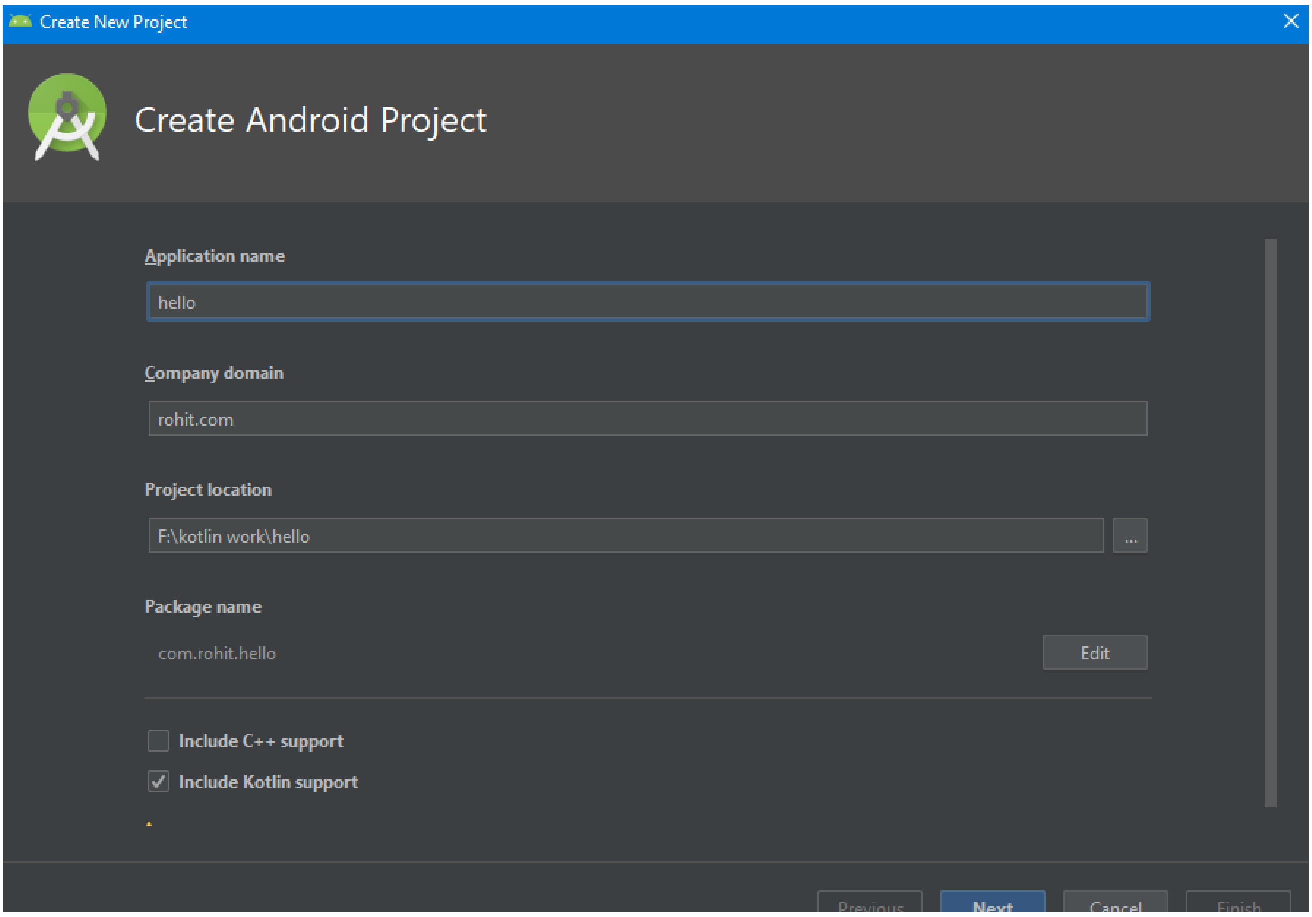
PRACTICAL 1

1. Introduction to Android, Introduction to Android Studio IDE, Application Fundamentals: Creating a Project, Android Components, Activities, Services, Content Providers, Broadcast Receivers, Interface overview, Creating Android Virtual device, USB debugging mode, Android Application Overview. Simple “Hello World” program.

Solution:

Creating a project:



Create New Project

Create Android Project

Application name: hello

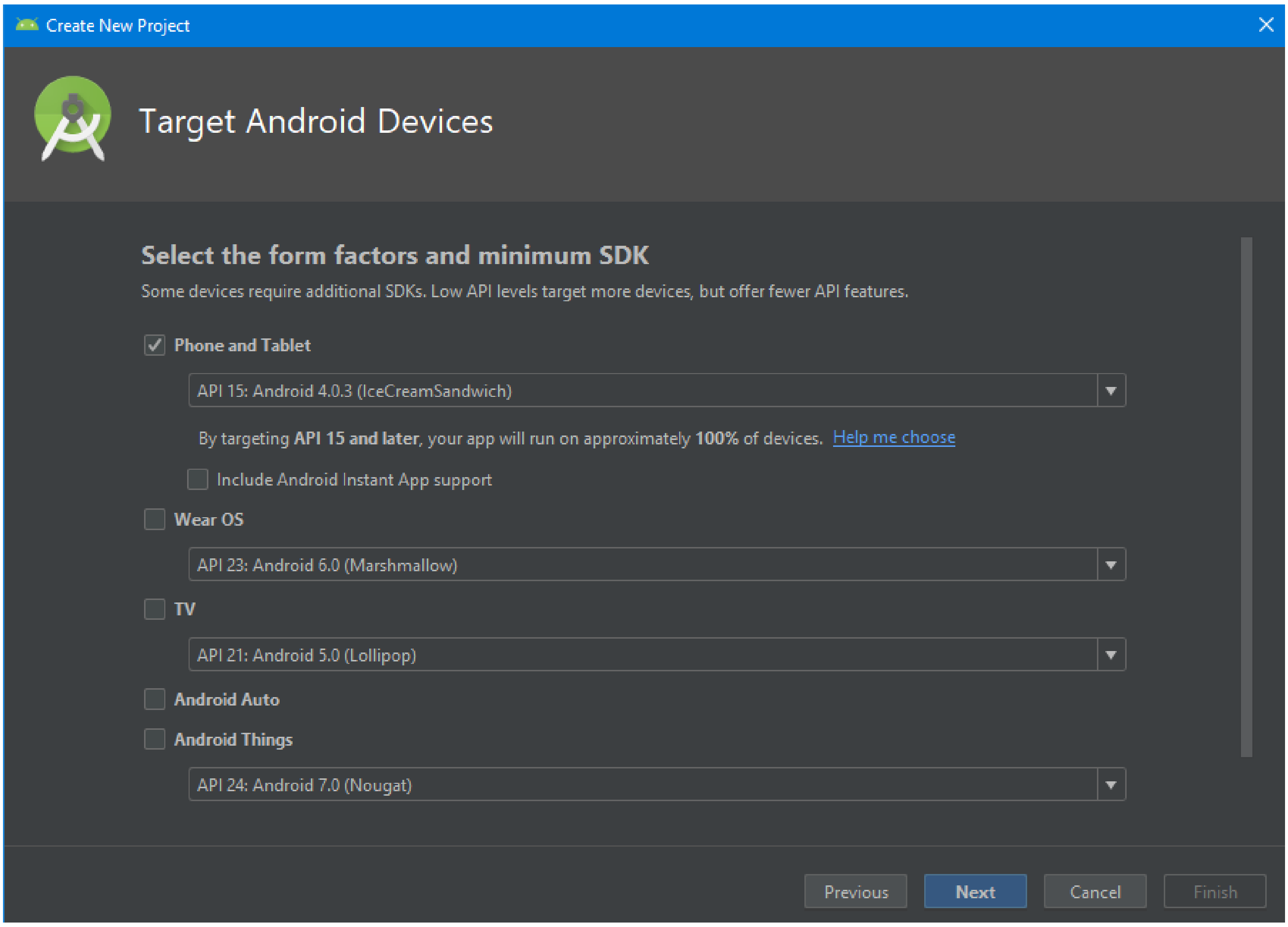
Company domain: rohit.com

Project location: F:\kotlin work\hello

Package name: com.rohit.hello

☐ Include C++ support

☑ Include Kotlin support



Target Android Devices

Select the form factors and minimum SDK

Some devices require additional SDKs. Low API levels target more devices, but offer fewer API features.

☑ Phone and Tablet

API 15: Android 4.0.3 (IceCreamSandwich)

By targeting **API 15 and later**, your app will run on approximately **100%** of devices. Help me choose

□ Include Android Instant App support

□ Wear OS

API 23: Android 6.0 (Marshmallow)

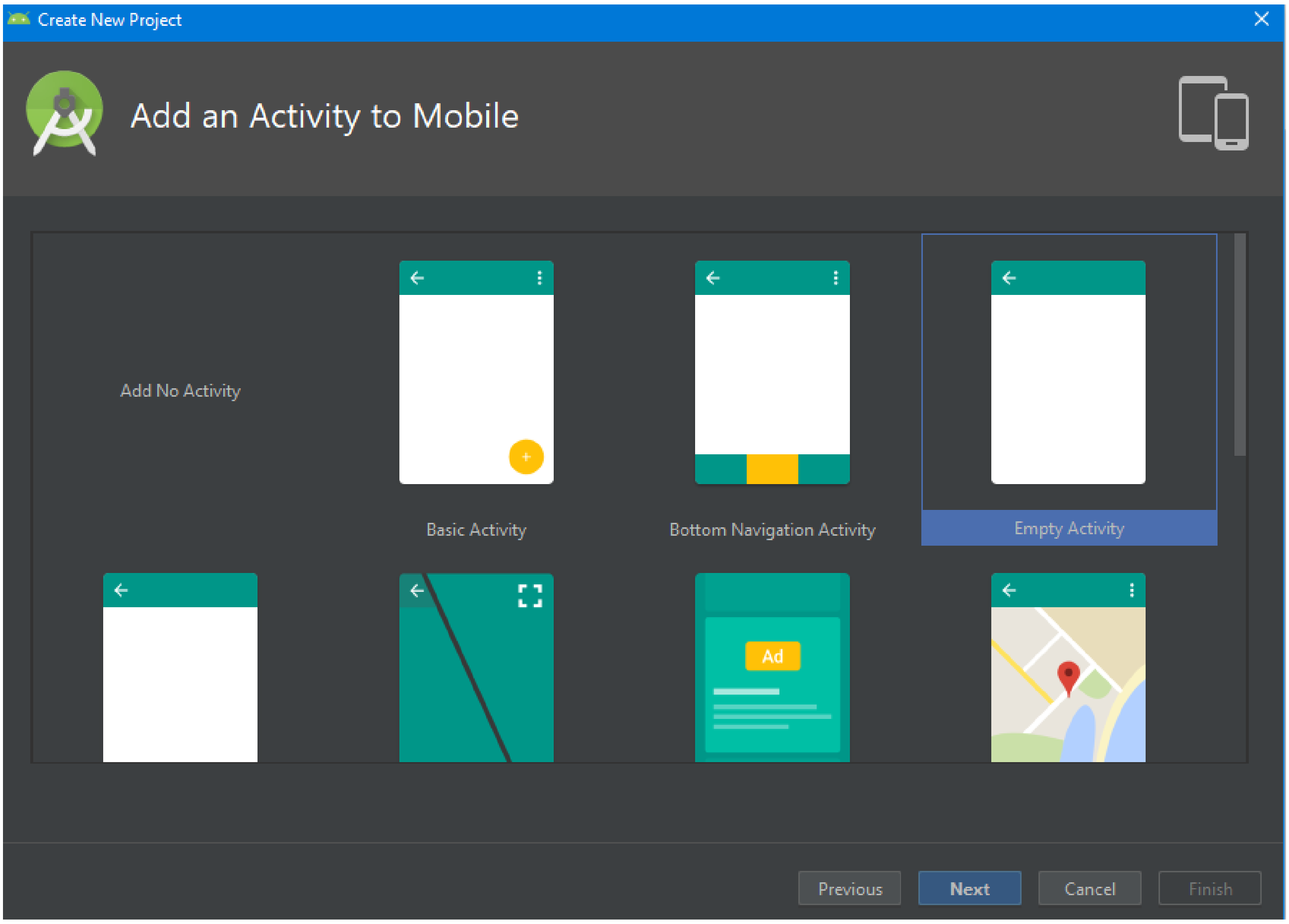
□ TV

API 21: Android 5.0 (Lollipop)

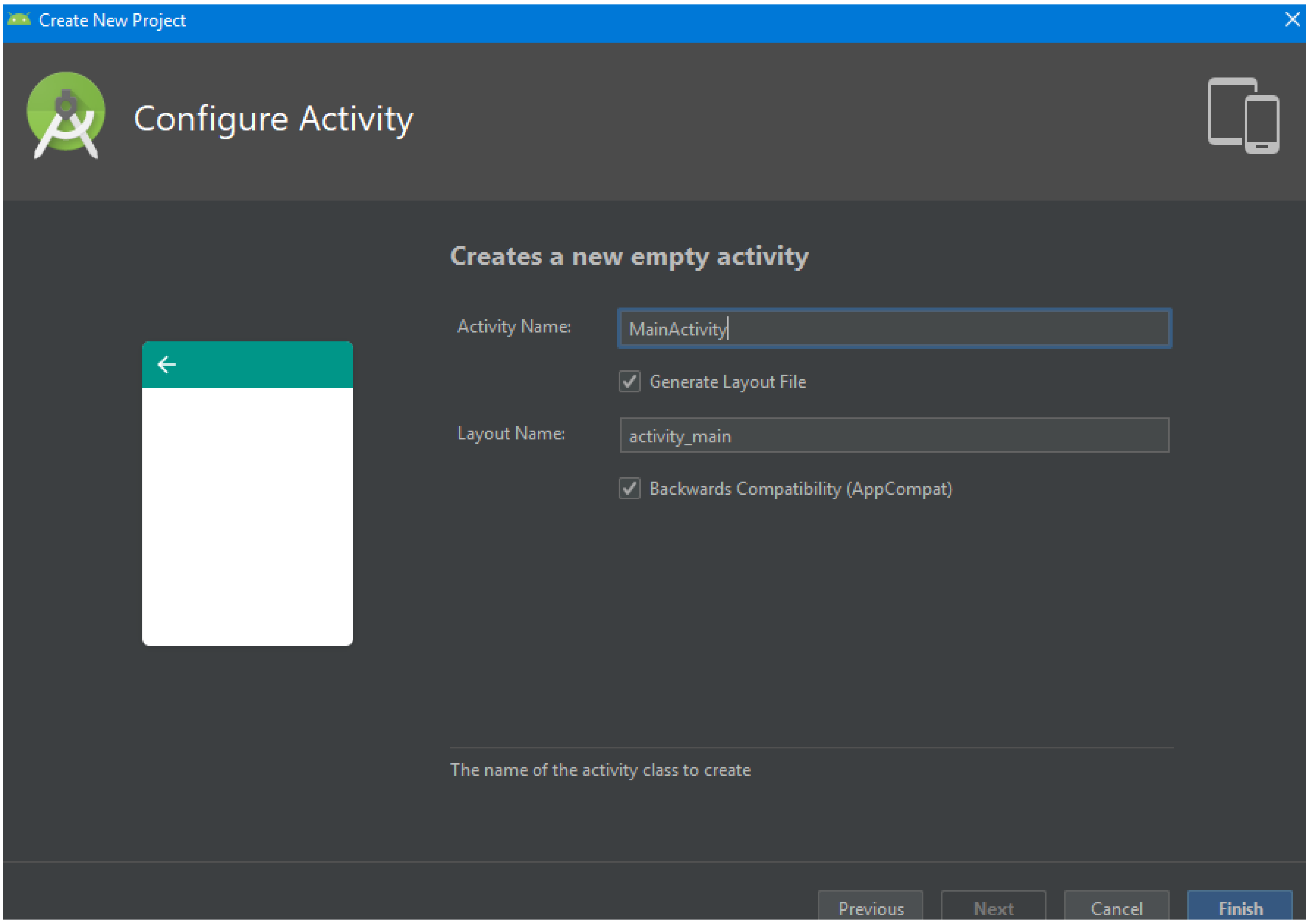
□ Android Auto

□ Android Things

API 24: Android 7.0 (Nougat)



Add an Activity to Mobile



Creates a new empty activity

Activity Name: MainActivity

☑ Generate Layout File

Layout Name: activity_main

☑ Backwards Compatibility (AppCompat)

The name of the activity class to create

Activity_Main.Kt

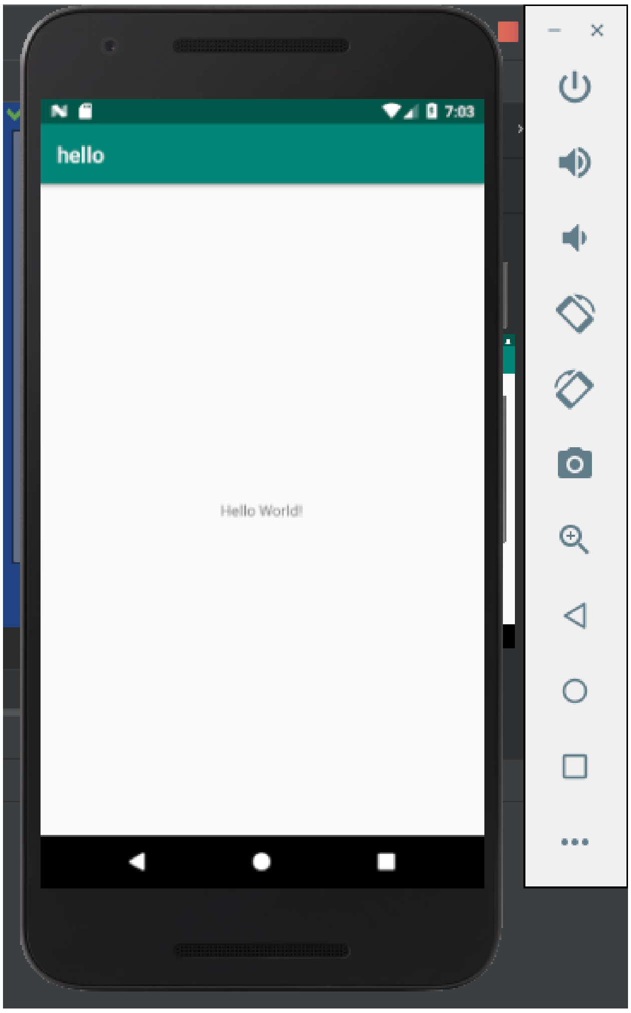
```kotlin
package com.rohit.hello

import android.support.v7.app.AppCompatActivity
import android.os.Bundle

class MainActivity : AppCompatActivity() {

    override fun onCreate(savedInstanceState: Bundle?) {
        super.onCreate(savedInstanceState)
        setContentView(R.layout.activity_main)
    }
}
```

Activity_Main.xml

```xml
<?xml version="1.0" encoding="utf-8"?>
<android.support.constraint.ConstraintLayout
    xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:tools="http://schemas.android.com/tools"
    xmlns:app="http://schemas.android.com/apk/res-auto"
    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 World!"
        app:layout_constraintBottom_toBottomOf="parent"
        app:layout_constraintLeft_toLeftOf="parent"
        app:layout_constraintRight_toRightOf="parent"
        app:layout_constraintTop_toTopOf="parent"/>
```

```
</android.support.constraint.ConstraintLayout>
```

Apk in avd:



BroadcastActivity:

How to receiving Broadcast

Apps can receive and android BroadcastReceiver in two ways: through manifest-declared receivers and context-registered receivers. In this example, we are approaching manifest-declared Receiver. Learn step by step to the kotlin broadcast receiver example works.

Step 1. Create an android app, For creating an Android app with kotlin read this tutorial.

Step 2. Creating Broadcast Receiver

Create and extend Subclass and BroadcastReceiver implement. onReceive(Context, Intent) where onReceive method each message is received as an Intent object parameter.

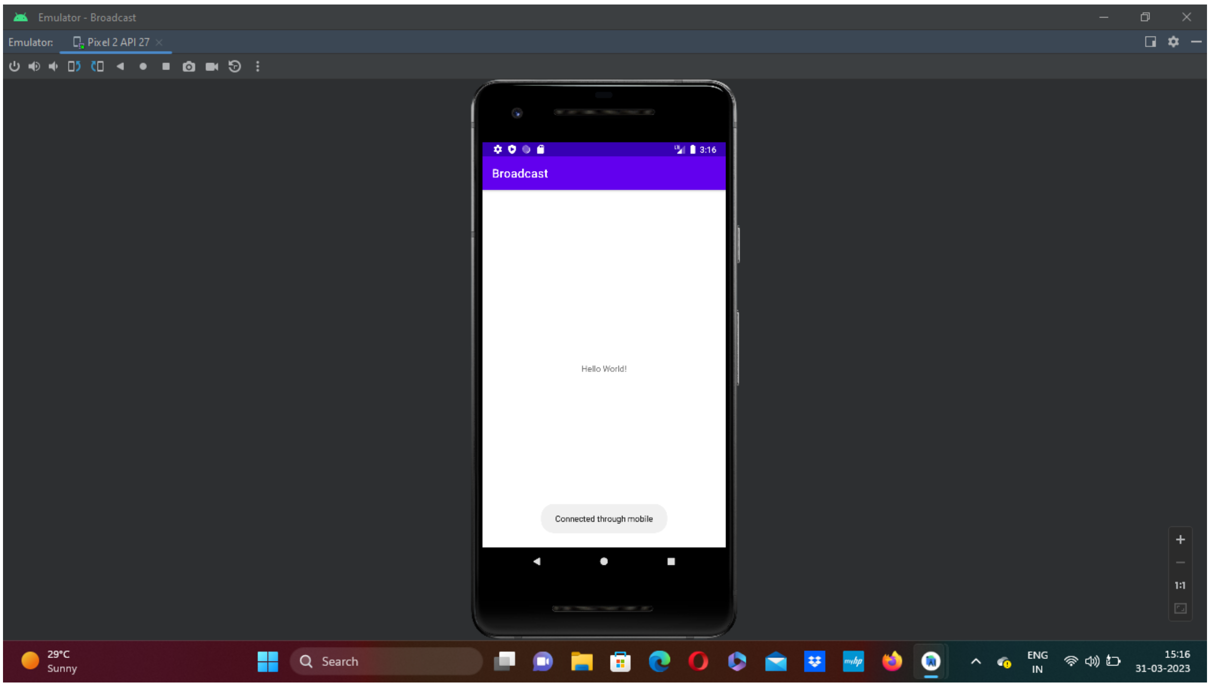
MainActivity.kt

```
@file:Suppress("DEPRECATION")

package com.example.broadcast
import android.content.Context
import android.net.ConnectivityManager
import android.net.ConnectivityManager.EXTRA_NO_CONNECTIVITY
import androidx.appcompat.app.AppCompatActivity
import android.os.Bundle
import android.widget.Toast

class MainActivity : AppCompatActivity() {
  override fun onCreate(savedInstanceState: Bundle?) {
    super.onCreate(savedInstanceState)
    setContentView(R.layout.activity_main)
    val c = applicationContext.getSystemService(Context.CONNECTIVITY_SERVICE) as
ConnectivityManager
    val networkInfo=c.activeNetworkInfo
    if(networkInfo!=null && networkInfo.isConnected)
    {
      if(networkInfo.type==ConnectivityManager.TYPE_MOBILE)
      {
        Toast.makeText(applicationContext,"Connected through
mobile",Toast.LENGTH_LONG).show()
      }
      if(networkInfo.type==ConnectivityManager.TYPE_WIFI)
      {
        Toast.makeText(applicationContext, "connected through wifi", Toast.LENGTH_LONG).show()
      }

      else
        Toast.makeText(applicationContext, "you are offline", Toast.LENGTH_LONG).show()
    }
  }
}
```

AndroidManifest.xml

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

<uses-permission android:name="android.permission.ACCESS_NETWORK_STATE"></uses-
permission>
  <application
    android:allowBackup="true"
    android:dataExtractionRules="@xml/data_extraction_rules"
    android:fullBackupContent="@xml/backup_rules"
    android:icon="@mipmap/ic_launcher"
    android:label="@string/app_name"
    android:supportsRtl="true"
    android:theme="@style/Theme.Broadcast"
    tools:targetApi="31">
    <activity
      android:name=".MainActivity"
      android:exported="true">
      <intent-filter>
        <action android:name="android.intent.action.MAIN" />

        <category android:name="android.intent.category.LAUNCHER" />
```

```
            </intent-filter>
        </activity>
    </application>

</manifest>
```

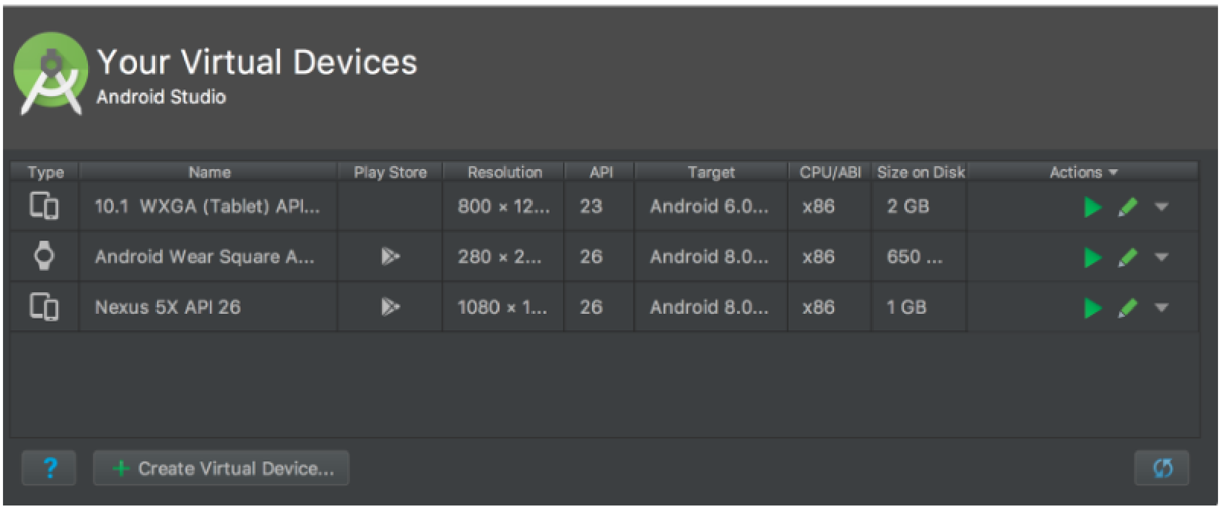


Connected through mobile

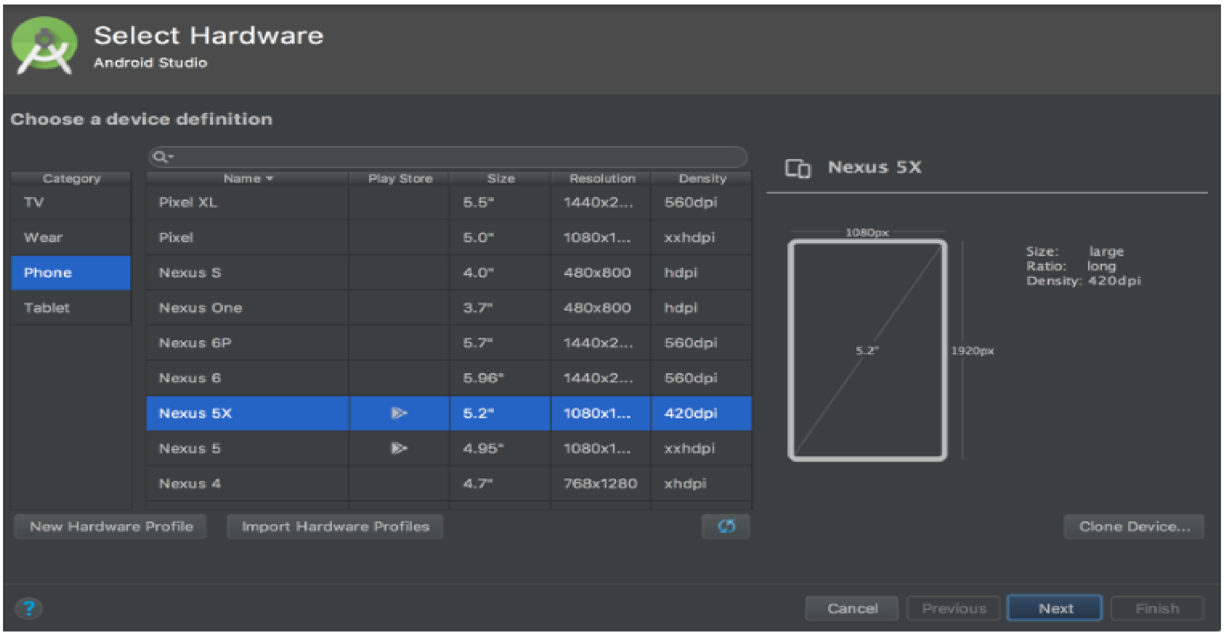
Create and manage virtual devices:

To open the AVD Manager, do one of the following:

- Select Tools > AVD Manager.
- Click AVD Manager AVD Manager icon in the toolbar.



Type	Name	Play Store	Resolution	API	Target	CPU/ABI	Size on Disk	Actions ▾
	10.1 WXGA (Tablet) API...		800 × 12...	23	Android 6.0...	x86	2 GB	▶ ✎ ▾
	Android Wear Square A...	▶	280 × 2...	26	Android 8.0...	x86	650 ...	▶ ✎ ▾
	Nexus 5X API 26	▶	1080 × 1...	26	Android 8.0...	x86	1 GB	▶ ✎ ▾



Select Hardware

Android Studio

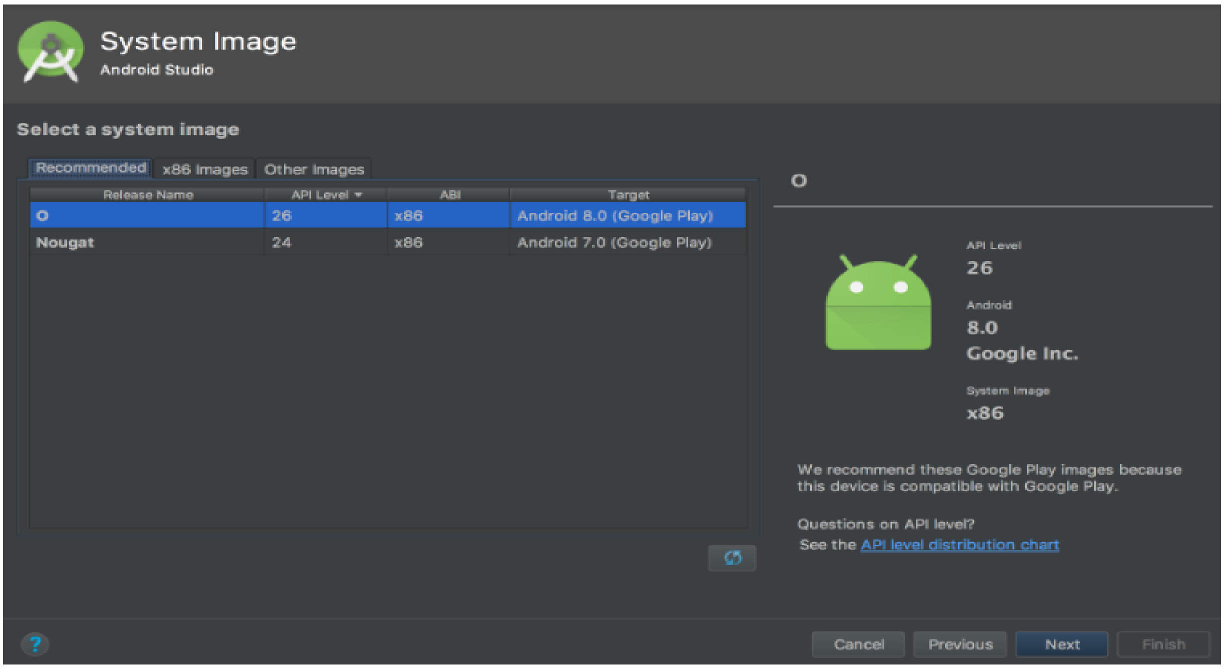
Choose a device definition

Category
TV
Wear
Phone
Tablet

Name ▾	Play Store	Size	Resolution	Density
Pixel XL		5.5"	1440x2...	560dpi
Pixel		5.0"	1080x1...	xxhdpi
Nexus S		4.0"	480x800	hdpi
Nexus One		3.7"	480x800	hdpi
Nexus 6P		5.7"	1440x2...	560dpi
Nexus 6		5.96"	1440x2...	560dpi
Nexus 5X	▶	5.2"	1080x1...	420dpi
Nexus 5	▶	4.95"	1080x1...	xxhdpi
Nexus 4		4.7"	768x1280	xhdpi

Nexus 5X

Size: large
Ratio: long
Density: 420dpi



System Image

Android Studio

Select a system image

Release Name	API Level ▾	ABI	Target
O	26	x86	Android 8.0 (Google Play)
Nougat	24	x86	Android 7.0 (Google Play)

O

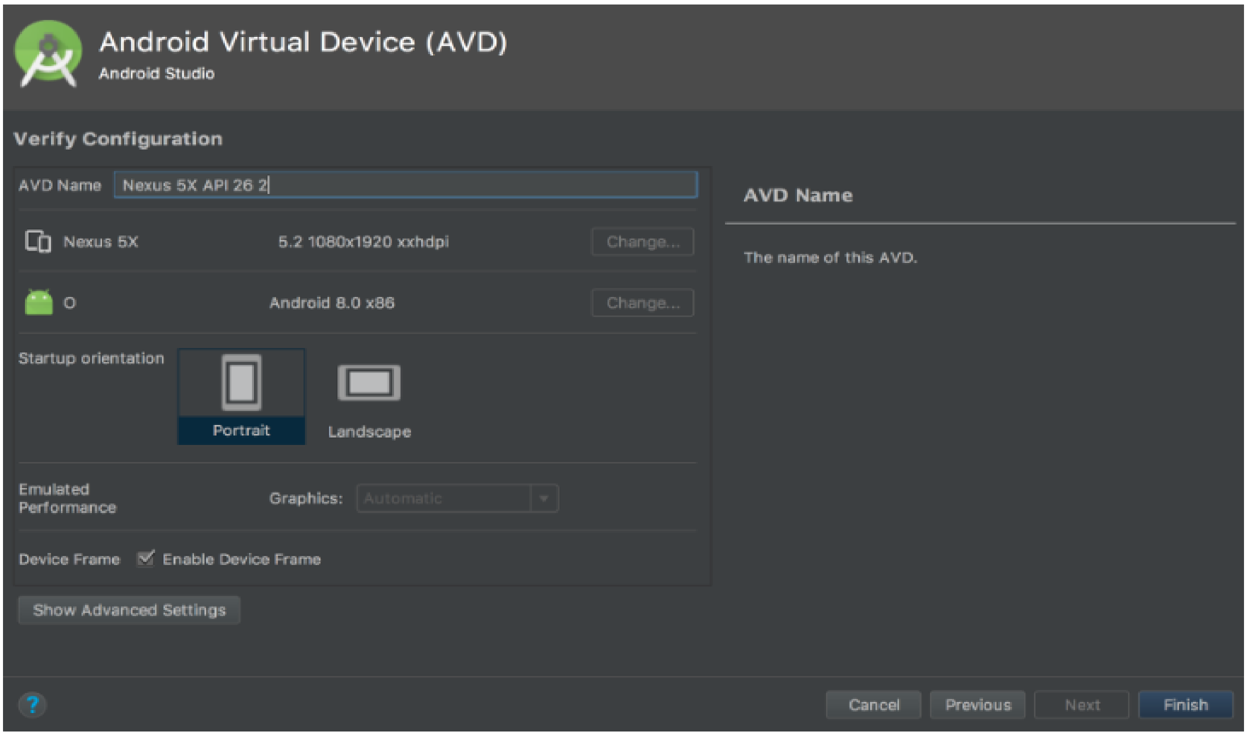
API Level
26

Android
8.0
Google Inc.

System Image
x86

We recommend these Google Play images because this device is compatible with Google Play.

Questions on API level?
See the API level distribution chart



Android Virtual Device (AVD)

Android Studio

Verify Configuration

AVD Name: Nexus 5X API 26 2

Nexus 5X 5.2 1080x1920 xxhdpi

O Android 8.0 x86

Startup orientation: Portrait

Emulated Performance Graphics: Automatic

Device Frame ☑ Enable Device Frame

AVD Name

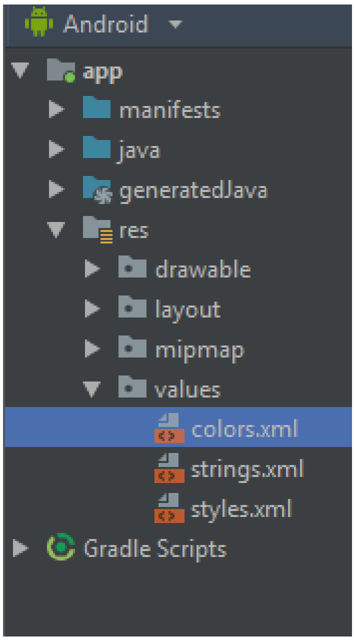
The name of this AVD.

PRACTICAL 2

Programming Resources

Android Resources: (Color, Theme, String, Drawable, Dimension, Image).

Color:



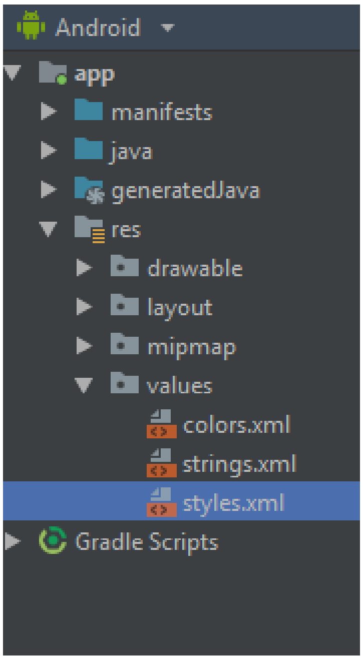
Color.xml

```
<?xml version="1.0" encoding="utf-8"?>
<resources>
    <color name="colorPrimary">#008577</color>
    <color name="colorPrimaryDark">#00574B</color>
    <color name="colorAccent">#D81B60</color>
</resources>
```

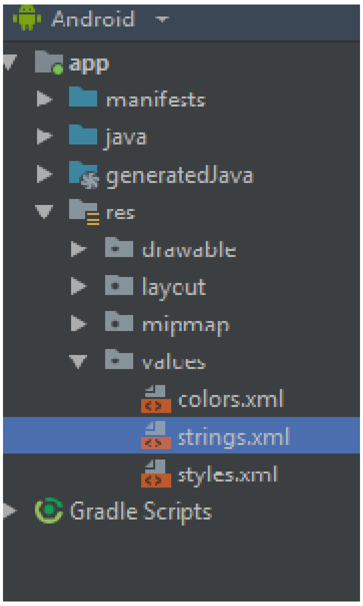
Theme:

Style.xml

```
<resources>

    <!-- Base application theme. -->
    <style name="AppTheme" parent="Theme.AppCompat.Light.DarkActionBar">
        <!-- Customize your theme here. -->
        <item name="colorPrimary">@color/colorPrimary</item>
        <item name="colorPrimaryDark">@color/colorPrimaryDark</item>
        <item name="colorAccent">@color/colorAccent</item>
    </style>

</resources>
```



String:

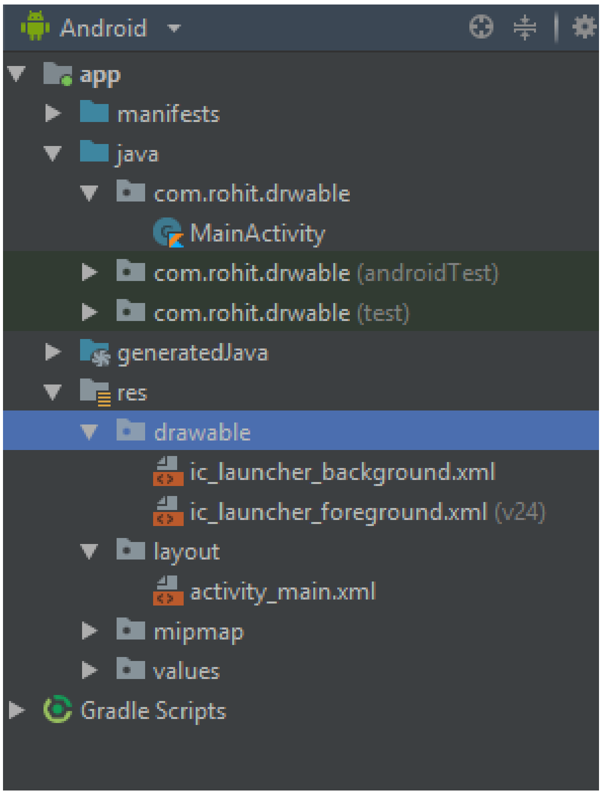


String.xml:

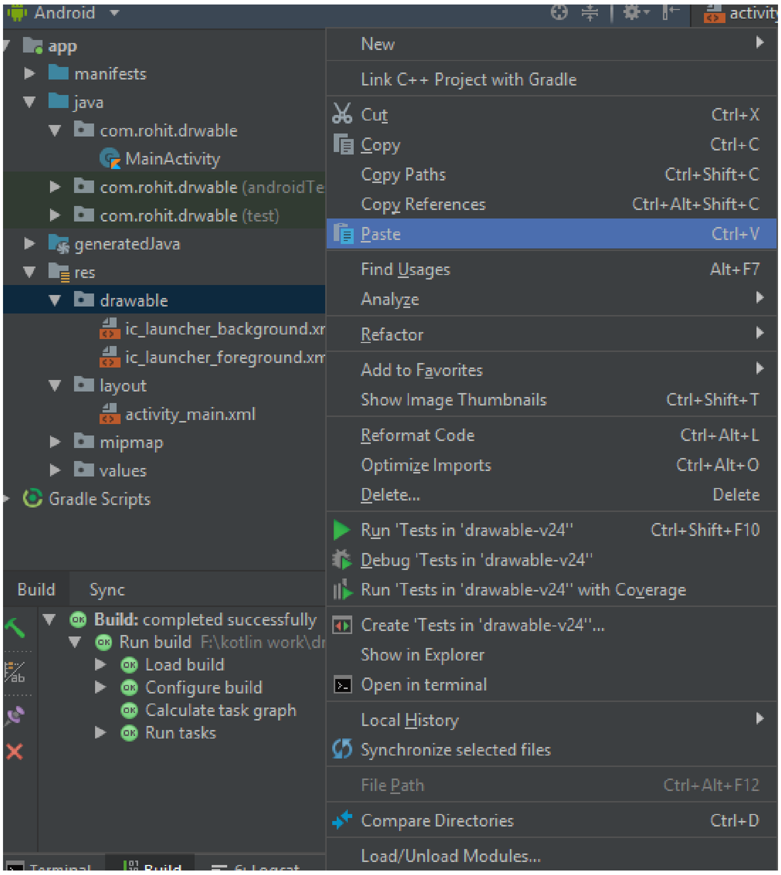
```
<resources>
  <string name="app_name">hello</string>
  <string name="numbers">
    <item>1</item>
    <item>2</item>
    <item>3</item>
    </item>
  </string>
</resources>
```

Drawable:

1. Right click on drawable folder



2. Copy the image if you want to create image drawable
3. Paste that image file inside the drawable folder



Note: to create drawable resource, right click on drawable folder and select drawable resource file.

Dimension, Image:

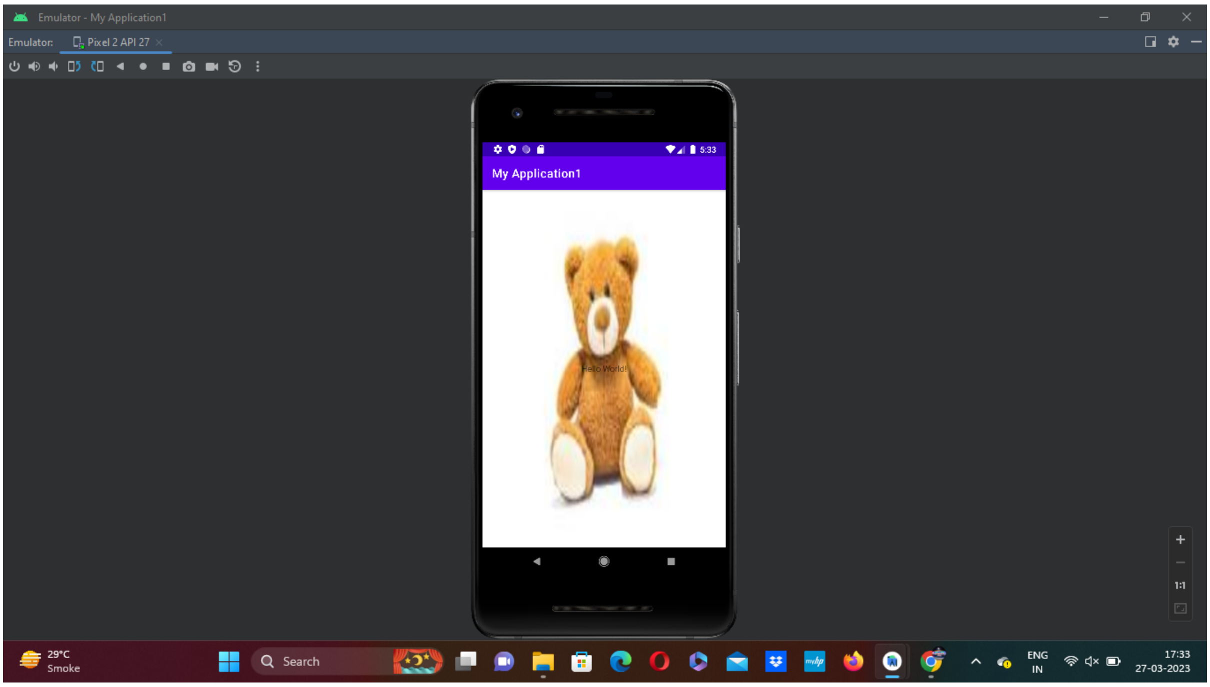
Main_Activity.kt:

```
package com.rohit.drwable

import android.support.v7.app.AppCompatActivity
import android.os.Bundle

class MainActivity : AppCompatActivity() {

   override fun onCreate(savedInstanceState: Bundle?) {
      super.onCreate(savedInstanceState)
      setContentView(R.layout.activity_main)
   }
}
```

activity_main.xml:

```
<?xml version="1.0" encoding="utf-8"?>
<LinearLayout
      xmlns:android="http://schemas.android.com/apk/res/android"
      xmlns:tools="http://schemas.android.com/tools"
      xmlns:app="http://schemas.android.com/apk/res-auto"
      android:layout_width="match_parent"
      android:layout_height="match_parent"
      tools:context=".MainActivity"
      android:background="@drawable/one">

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

</LinearLayout>
```

Output:

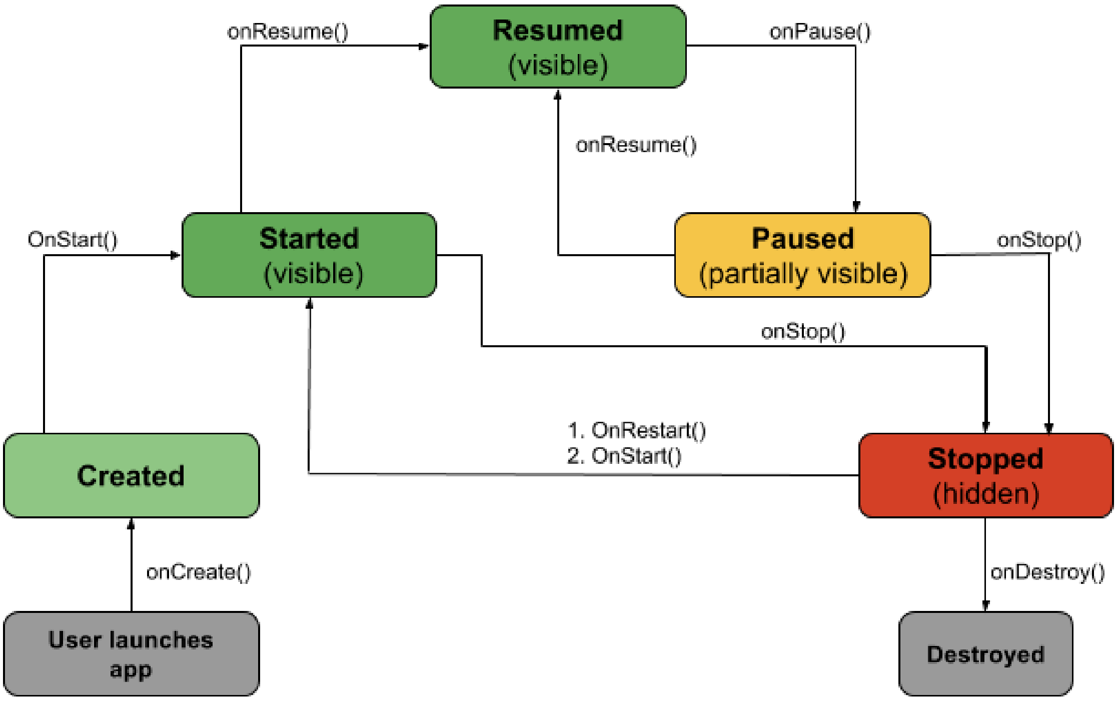


PRACTICAL 3

Programming Activities and fragments

Activity Life Cycle, Activity methods, Multiple Activities, Life Cycle of fragments and multiple fragments.

Activity Lifecycle:



- **onCreate():** Called by the OS when the activity is first created. This is where you initialize any UI elements or data objects. You also have the savedInstanceState of the activity that contains its previously saved state, and you can use it to recreate that state.\

```
fun onCreate(savedInstanceState: Bundle?) {
  super.onCreate(savedInstanceState)
  setContentView(R.layout.activity_task_description)
```

- **onStart():** Just before presenting the user with an activity, this method is called. It's always followed by onResume(). In here, you generally should start UI animations, audio based content or anything else that requires the activity's contents to be on screen.

-

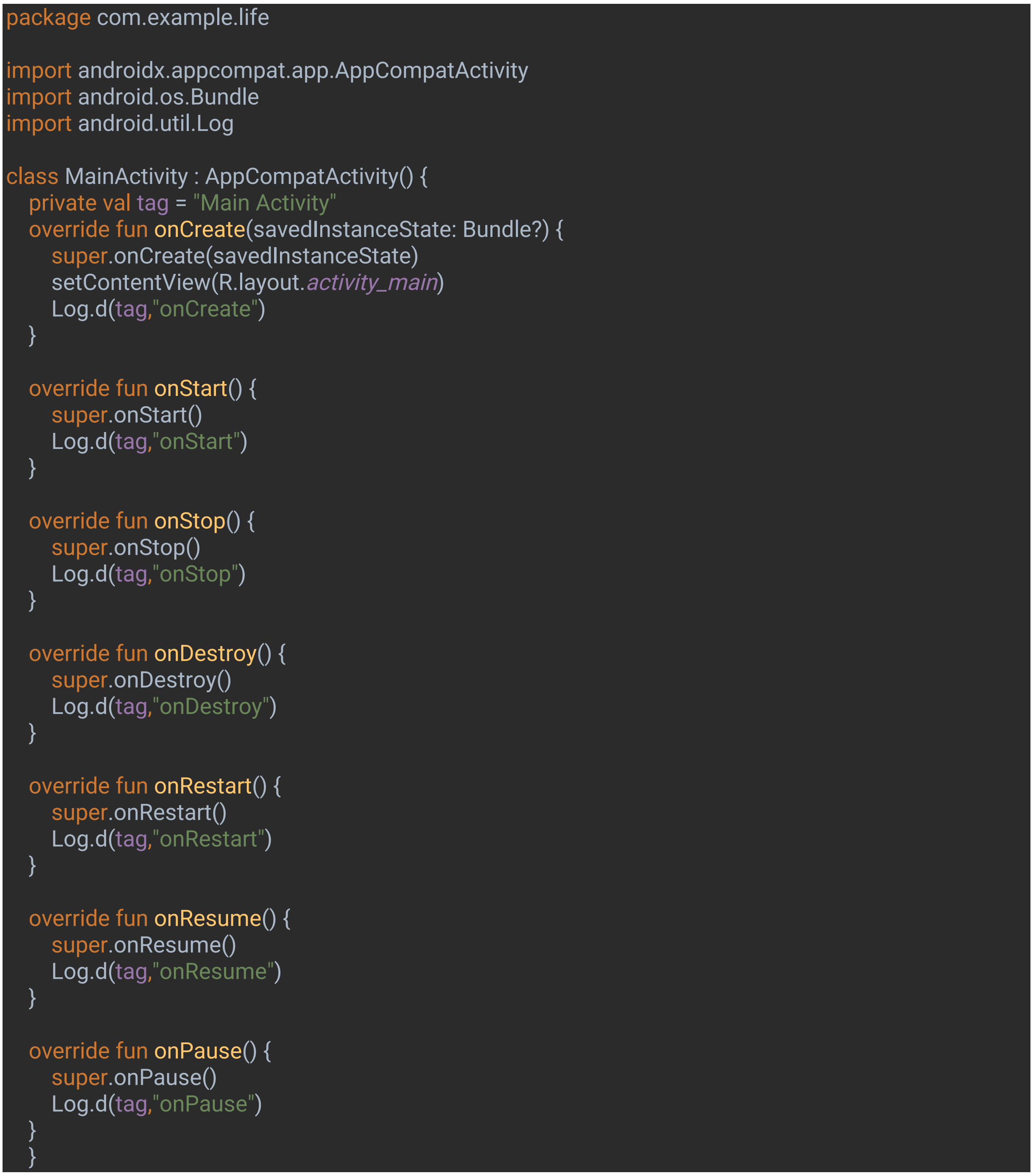
- **onResume():** As an activity enters the foreground, this method is called. Here you have a good place to restart animations, update UI elements, restart camera previews, resume audio/video playback or initialize any components that you release during onPause().

- **onPause():** This method is called before sliding into the background. Here you should stop any visuals or audio associated with the activity such as UI animations, music playback or the camera. This method is followed by onResume() if the activity returns to the foreground or by onStop() if it becomes hidden.

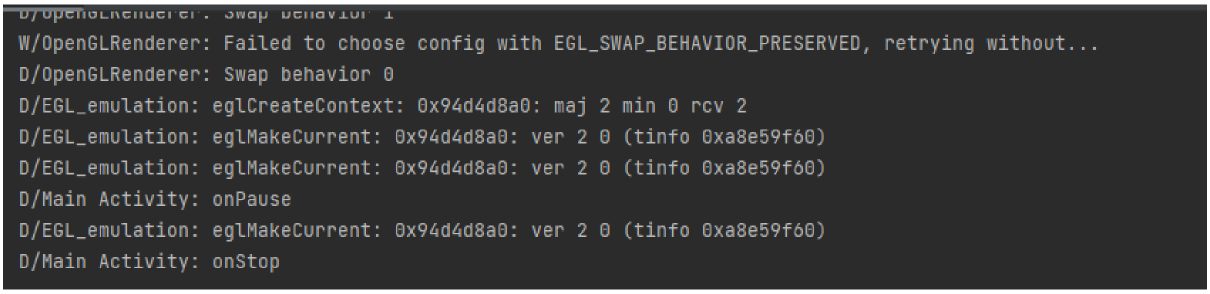
- **onStop():** This method is called right after onPause(), when the activity is no longer visible to the user, and it's a good place to save data that you want to commit to the disk. It's followed by either onRestart(), if this activity is coming back to the foreground, or onDestroy() if it's being released from memory.

- **onRestart():** Called after stopping an activity, but just before starting it again. It's always followed by onStart().

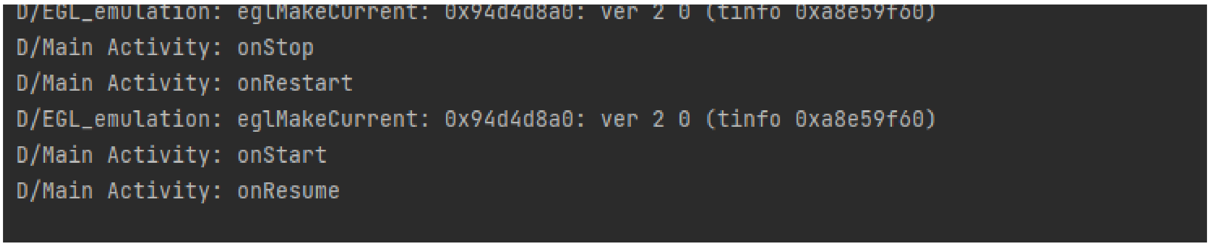
- **onDestroy():** This is the final callback you'll receive from the OS before the activity is destroyed. You can trigger an activity's desctruction by calling finish(), or it can be triggered by the system when the system needs to recoup memory. If your activity includes any background threads or other long-running resources, destruction could lead to a memory leak if they're not released, so you need to remember to stop these processes here as well.



```
package com.example.life

import androidx.appcompat.app.AppCompatActivity
import android.os.Bundle
import android.util.Log

class MainActivity : AppCompatActivity() {
    private val tag = "Main Activity"
    override fun onCreate(savedInstanceState: Bundle?) {
        super.onCreate(savedInstanceState)
        setContentView(R.layout.activity_main)
        Log.d(tag,"onCreate")
    }

    override fun onStart() {
        super.onStart()
        Log.d(tag,"onStart")
    }

    override fun onStop() {
        super.onStop()
        Log.d(tag,"onStop")
    }

    override fun onDestroy() {
        super.onDestroy()
        Log.d(tag,"onDestroy")
    }

    override fun onRestart() {
        super.onRestart()
        Log.d(tag,"onRestart")
    }

    override fun onResume() {
        super.onResume()
        Log.d(tag,"onResume")
    }

    override fun onPause() {
        super.onPause()
        Log.d(tag,"onPause")
    }
    }
```



```
D/OpenGLRenderer: Swap behavior 1
W/OpenGLRenderer: Failed to choose config with EGL_SWAP_BEHAVIOR_PRESERVED, retrying without...
D/OpenGLRenderer: Swap behavior 0
D/EGL_emulation: eglCreateContext: 0x94d4d8a0: maj 2 min 0 rcv 2
D/EGL_emulation: eglMakeCurrent: 0x94d4d8a0: ver 2 0 (tinfo 0xa8e59f60)
D/EGL_emulation: eglMakeCurrent: 0x94d4d8a0: ver 2 0 (tinfo 0xa8e59f60)
D/Main Activity: onPause
D/EGL_emulation: eglMakeCurrent: 0x94d4d8a0: ver 2 0 (tinfo 0xa8e59f60)
D/Main Activity: onStop
```



```
D/EGL_emulation: eglMakeCurrent: 0x94d4d8a0: ver 2 0 (tinfo 0xa8e59f60)
D/Main Activity: onStop
D/Main Activity: onRestart
D/EGL_emulation: eglMakeCurrent: 0x94d4d8a0: ver 2 0 (tinfo 0xa8e59f60)
D/Main Activity: onStart
D/Main Activity: onResume
```

PRACTICAL 4

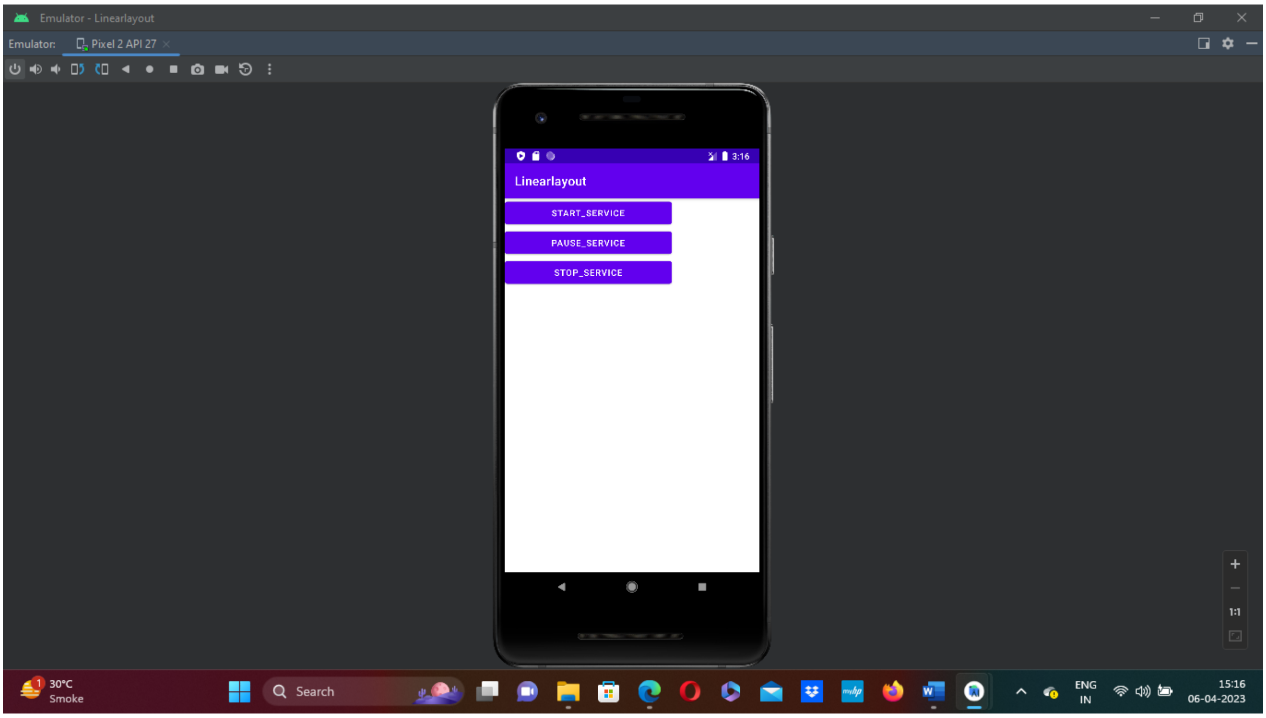
Programs related to different Layouts

Coordinate, Linear, Relative, Table, Absolute, Frame, List View, Grid View.

1. linear layout:

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

<Button android:id="@+id/btnStartService"
android:layout_width="270dp"
android:layout_height="wrap_content"
android:text="start_service"/>

<Button android:id="@+id/btnPauseService"
android:layout_width="270dp"
android:layout_height="wrap_content"
android:text="pause_service"/>
```

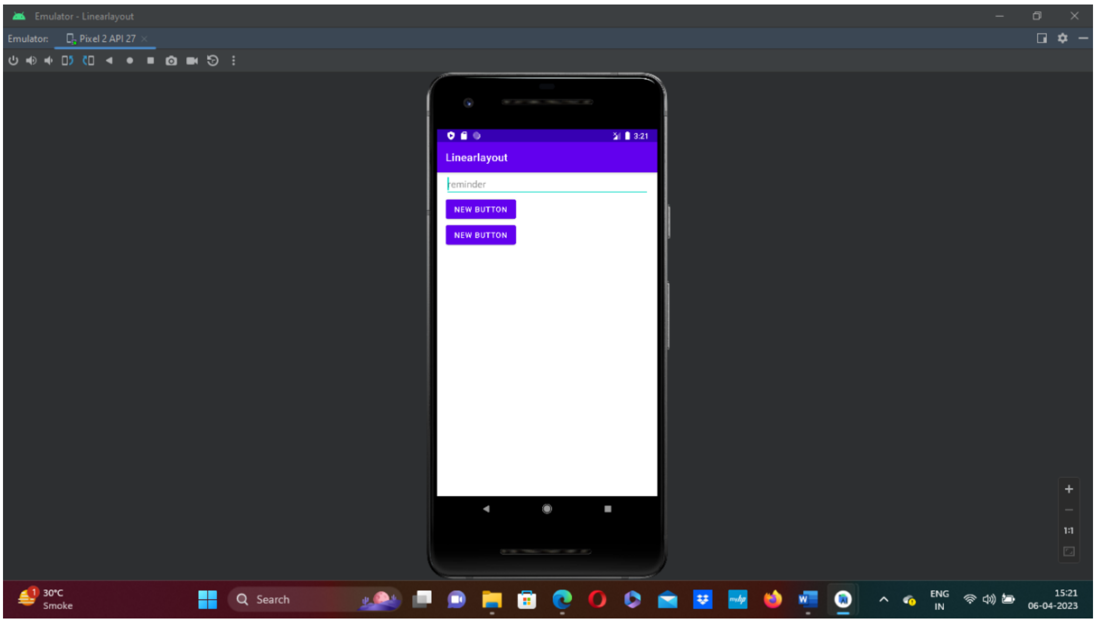
```
<Button android:id="@+id/btnStopService"
android:layout_width="270dp"
android:layout_height="wrap_content"
android:text="stop_service"/>

</LinearLayout>
```



2. Relative:

```
<RelativeLayout xmlns:android="http://schemas.android.com/apk/res/android"
android:layout_width="fill_parent"
android:layout_height="fill_parent"
android:paddingLeft="16dp"
android:paddingRight="16dp" >

<EditText
android:id="@+id/name"
android:layout_width="fill_parent"
android:layout_height="wrap_content"
android:hint="@string/reminder" />

<LinearLayout
android:orientation="vertical"
android:layout_width="fill_parent"
android:layout_height="fill_parent"
android:layout_alignParentStart="true"
android:layout_below="@+id/name">

<Button
android:layout_width="wrap_content"
android:layout_height="wrap_content"
android:text="New Button"
android:id="@+id/button" />

<Button
android:layout_width="wrap_content"
android:layout_height="wrap_content"
android:text="New Button"
android:id="@+id/button2" />
```

```
</LinearLayout>

</RelativeLayout>
```



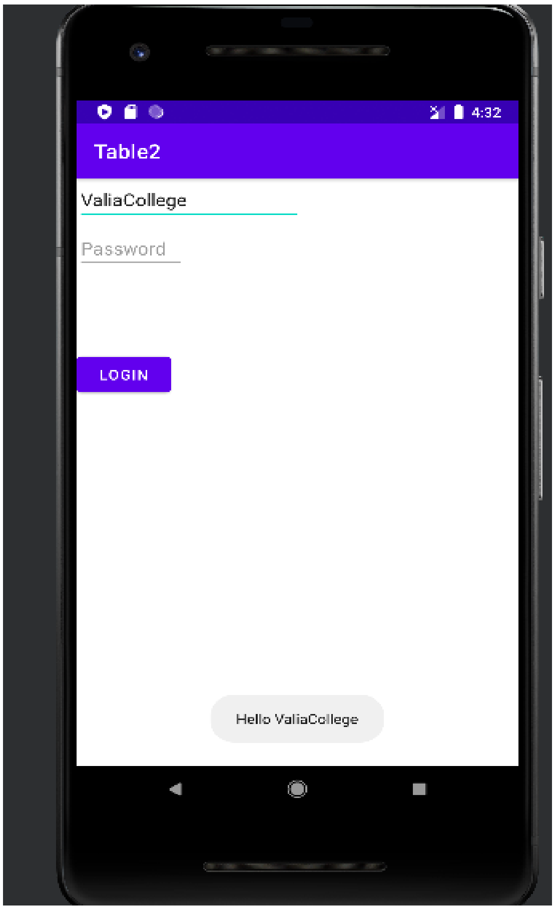
3. Table:

Activity_main.xml

Table View

```
<?xml version="1.0" encoding="utf-8"?>
<androidx.constraintlayout.widget.ConstraintLayout
xmlns:android="http://schemas.android.com/apk/res/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">

    <TableRow
        android:layout_width="match_parent"
        android:layout_height="match_parent"
        >

        <EditText
            android:id="@+id/my_text"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:ems="10"
            android:inputType="textPersonName"
            android:hint="UserName" />

    </TableRow>

    <TableRow
        android:layout_width="177dp"
        android:layout_height="609dp"
        tools:layout_editor_absoluteX="309dp"
        tools:layout_editor_absoluteY="170dp"
        tools:ignore="MissingConstraints">

        <EditText
            android:id="@+id/mypass_text"
            android:layout_width="100dp"
            android:layout_height="wrap_content"
            android:layout_marginTop="50dp"
            android:ems="10"
            android:hint="Password" />
    </TableRow>

    <TableRow
        android:layout_width="match_parent"
        android:layout_height="match_parent" >

        <Button
            android:id="@+id/Login"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:layout_marginBottom="100dp"
            android:layout_gravity="center"
            android:text="Login" />
    </TableRow>
</androidx.constraintlayout.widget.ConstraintLayout>
```

```
import androidx.appcompat.app.AppCompatActivity
import android.os.Bundle
import android.widget.Button
import android.widget.EditText
import android.widget.Toast


class MainActivity : AppCompatActivity() {
    override fun onCreate(savedInstanceState: Bundle?) {
        super.onCreate(savedInstanceState)
        setContentView(R.layout.activity_main)
        var text=findViewById<EditText>(R.id.my_text)
        var btn=findViewById<Button>(R.id.Login)
        btn.setOnClickListener()
        {
            var strMessage:String
            strMessage=text.getText().toString()
            Toast.makeText(this,"Hello $strMessage",Toast.LENGTH_LONG).show()
        }
    }
}
```

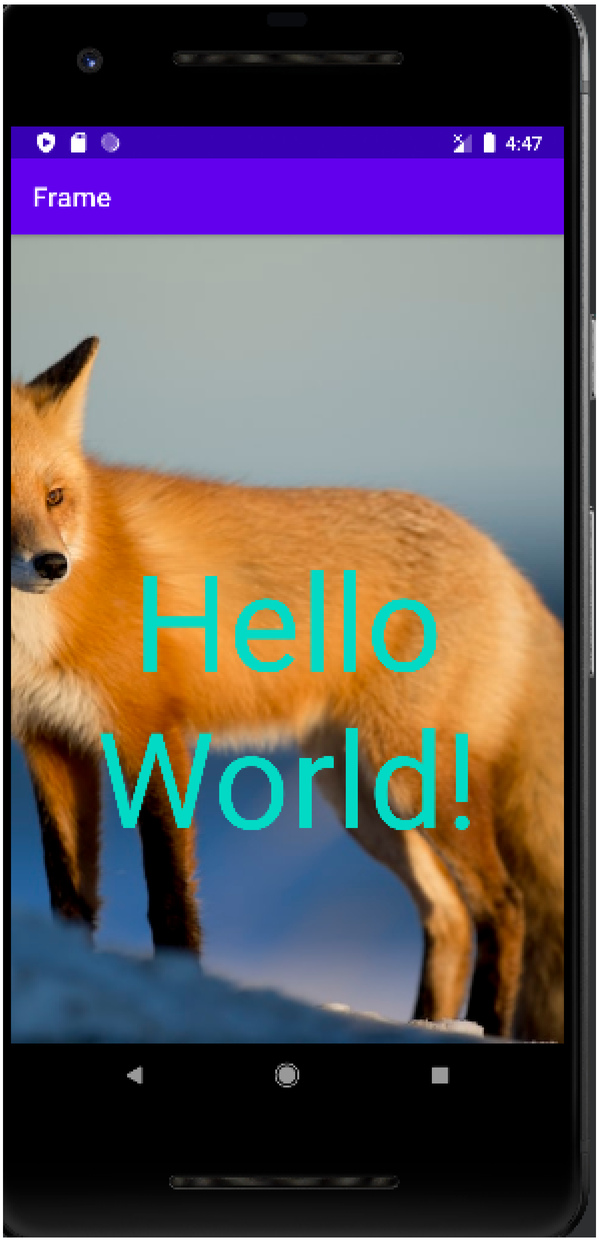


ValiaCollege

Hello ValiaCollege

FrameViewLayout

```
<?xml version="1.0" encoding="utf-8"?>
<FrameLayout
   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"
   tools:context=".MainActivity">

   <ImageView android:layout_width="match_parent"
      android:layout_height="match_parent"
      android:src="@drawable/fox"
      android:scaleType="centerCrop"/>
   <TextView
      android:textSize="100dp"
      android:layout_width="wrap_content"
      android:layout_height="wrap_content"
      android:text="Hello World!"
      android:gravity="center"
      android:textColor="@color/teal_200"
      android:layout_marginTop="220dp"
      />

</FrameLayout>
```

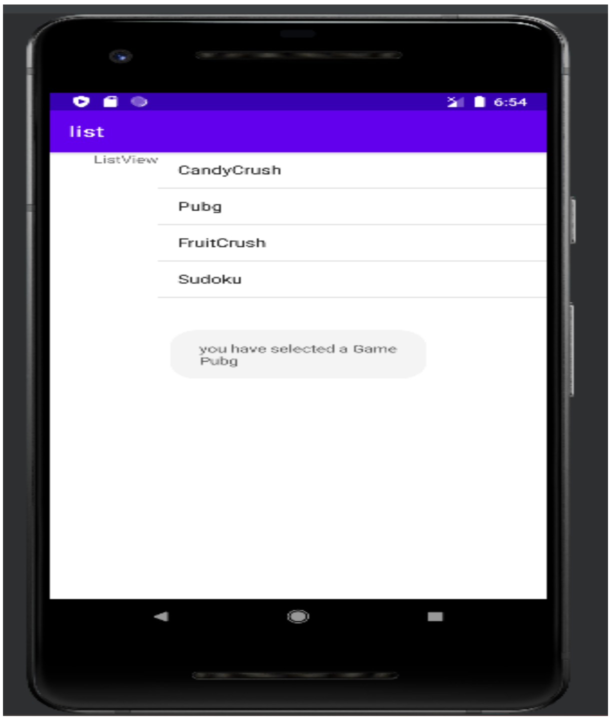


List View.kt

```
package com.example.list
import androidx.appcompat.app.AppCompatActivity
import android.os.Bundle
import android.view.Gravity
import android.widget.ArrayAdapter
import android.widget.ListView
import android.widget.Toast
class MainActivity : AppCompatActivity() {

  override fun onCreate(savedInstanceState: Bundle?) {
    super.onCreate(savedInstanceState)
    setContentView(R.layout.activity_main)
    val sub= arrayOf<String>("CandyCrush","Pubg","FruitCrush","Sudoku")
    val lview=findViewById<ListView>(R.id.my_list)
    lview.adapter=ArrayAdapter<String>(this,android.R.layout.simple_list_item_1,sub)
    lview.setOnItemClickListener { parent, view, position, id ->
      val toast =Toast.makeText(this,"you have selected a
Game\n"+sub[position],Toast.LENGTH_LONG)
      toast.show()
      toast.setGravity(Gravity.CENTER,0,0)
```

```
            toast.show()

        }

                }
    }
```

```
<?xml version="1.0" encoding="utf-8"?>
<LinearLayout xmlns:android="http://schemas.android.com/apk/res/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="ListView"
        android:layout_marginLeft="36dp"
        app:layout_constraintBottom_toBottomOf="parent"
        app:layout_constraintEnd_toEndOf="parent"
        app:layout_constraintStart_toStartOf="parent"
        app:layout_constraintTop_toTopOf="parent" />

    <ListView
        android:layout_width="match_parent"
        android:layout_height="match_parent"
        android:id="@+id/my_list"/>

</LinearLayout>
```



ListView

CandyCrush

Pubg

FruitCrush

Sudoku

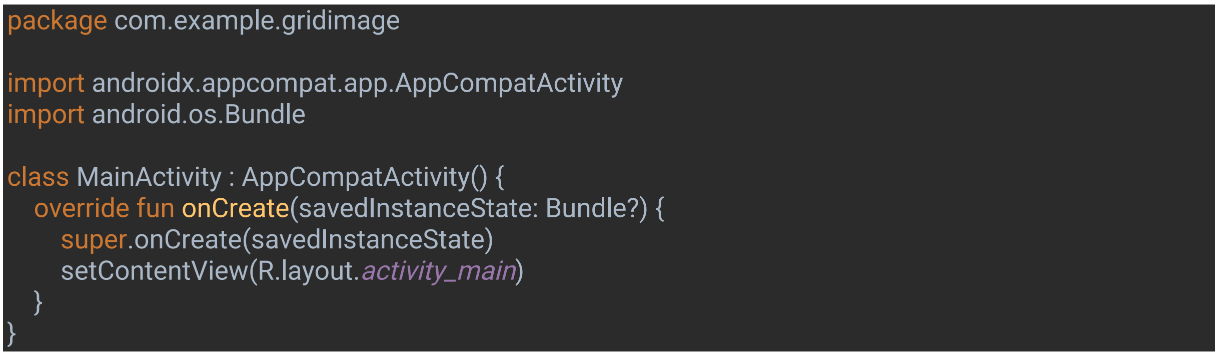
you have selected a Game
Pubg

1. Grid layout:

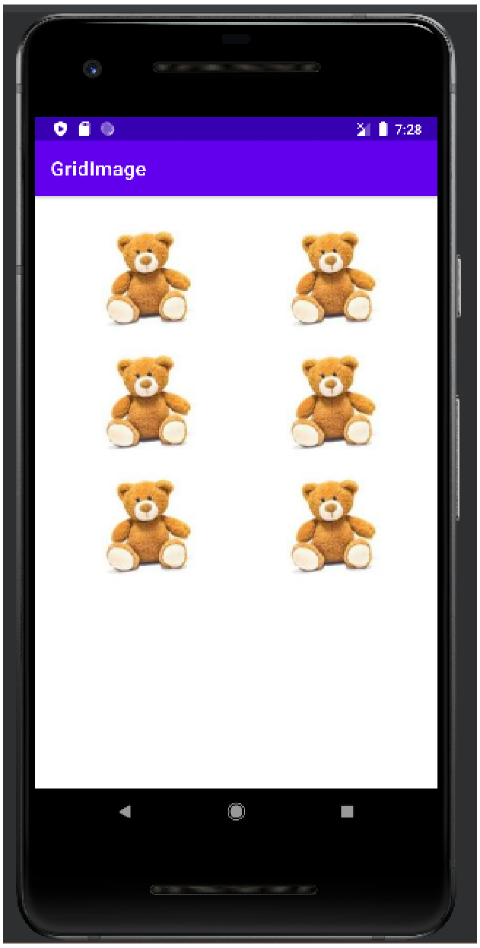
```
2.          <?xml version="1.0"
encoding="utf-8"?>
<GridLayout
        xmlns:android="http://schemas.android.com/apk/res/android"
            xmlns:app="http://schemas.android.com/apk/res-auto"
            xmlns:tools="http://schemas.android.com/tools"
            android:layout_width="match_parent"
            android:layout_height="match_parent"
            android:columnCount="3"
            android:rowCount="3"
            android:layout_margin="20dp"
            tools:context=".MainActivity">
```

```
  <ImageView
    android:id="@+id/imageView"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    app:srcCompat="@drawable/images"
    android:layout_column="0"
    android:layout_row="0"/>
  <ImageView
    android:id="@+id/imageView1"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    app:srcCompat="@drawable/images"
    android:layout_column="1"
    android:layout_row="0"/>
  <ImageView
    android:id="@+id/imageView2"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    app:srcCompat="@drawable/images"
    android:layout_column="2"
    android:layout_row="0"/>
  <ImageView
    android:id="@+id/imageView3"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    app:srcCompat="@drawable/images"
    android:layout_column="0"
    android:layout_row="1"/>
  <ImageView
    android:id="@+id/imageView4"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    app:srcCompat="@drawable/images"
    android:layout_column="1"
    android:layout_row="1"/>
  <ImageView
    android:id="@+id/imageView5"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    app:srcCompat="@drawable/images"
    android:layout_column="2"
    android:layout_row="1"/>
  <ImageView
    android:id="@+id/imageView6"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    app:srcCompat="@drawable/images"
    android:layout_column="0"
    android:layout_row="2"/>
  <ImageView
    android:id="@+id/imageView7"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    app:srcCompat="@drawable/images"
    android:layout_column="1"
    android:layout_row="2"/>
  <ImageView
    android:id="@+id/imageView8"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    app:srcCompat="@drawable/images"
    android:layout_column="2"
    android:layout_row="2"/>

</GridLayout>
```

mainActvity.kt:



```
package com.example.gridimage

import androidx.appcompat.app.AppCompatActivity
import android.os.Bundle

class MainActivity : AppCompatActivity() {
   override fun onCreate(savedInstanceState: Bundle?) {
      super.onCreate(savedInstanceState)
      setContentView(R.layout.activity_main)
   }
}
```



PRACTICAL 5

Program on Appbar

To create appbar first we have to create a menu for that go to File-

>New->Android resources file->menu and select menu type as menu and click ok

Now menu.xml will get created under menu folder in res

We are going to add a search menu for that we must add an image in the drawable folder. Copy the image and paste it in the drawable.

Menu.xml

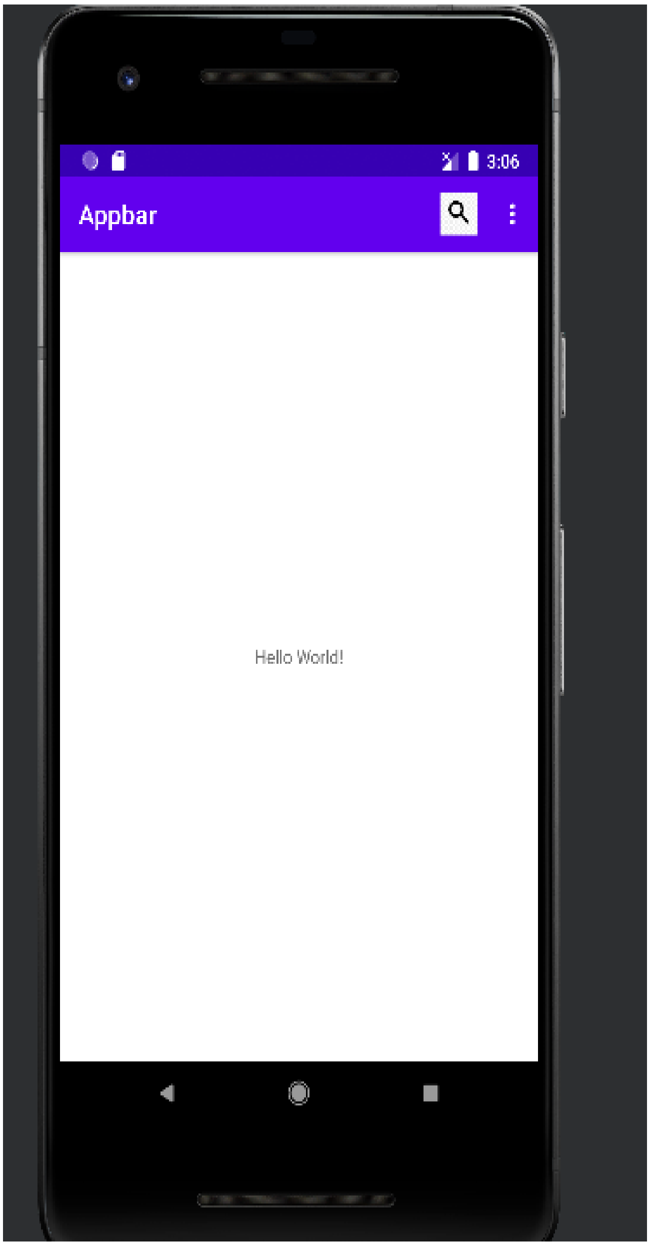
```
<?xml version="1.0" encoding="utf-8"?>
<menu xmlns:android="http://schemas.android.com/apk/res/android"
  xmlns:app="http://schemas.android.com/apk/res-auto">
<item
  android:id="@+id/search"
  android:icon="@drawable/search"
  android:title="Search"
  app:showAsAction="always">

</item>
  <item
    android:id="@+id/aboutus"
    android:title="AboutcUs"></item>
  <item
    android:id="@+id/contactus"
    android:title="Contact US"></item>
</menu>
```

Mainactivity.kt

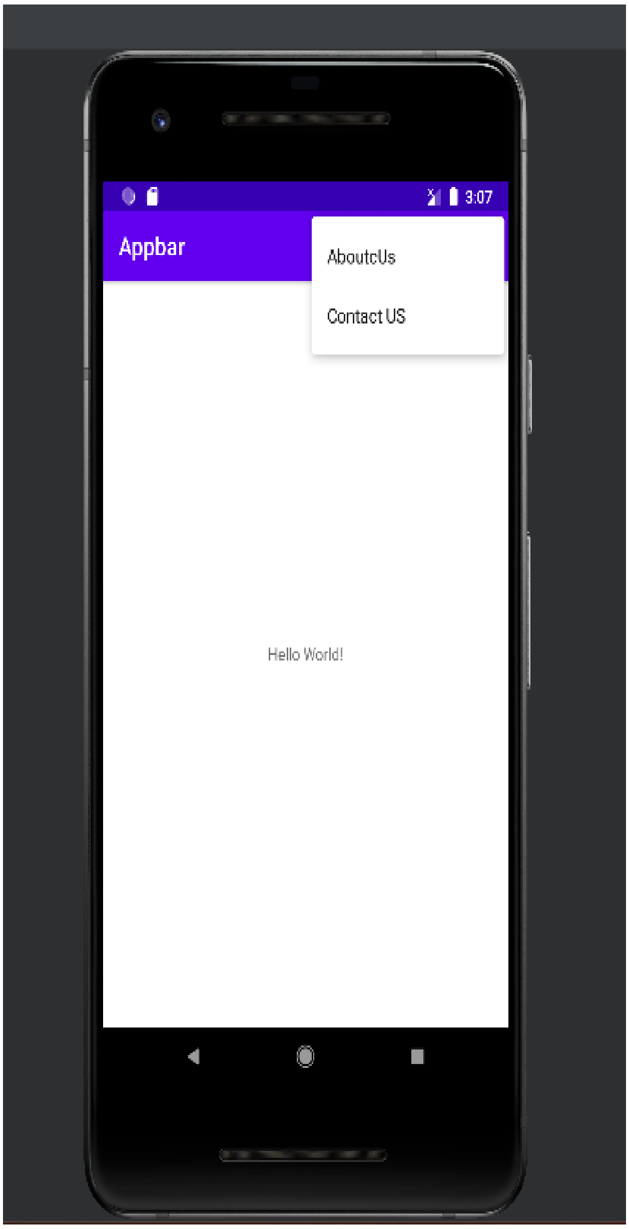
```
package com.example.appbar

import androidx.appcompat.app.AppCompatActivity
import android.os.Bundle
import android.view.Menu
import android.view.MenuItem
import android.widget.Toast

class MainActivity : AppCompatActivity() {
  override fun onCreate(savedInstanceState: Bundle?) {
    super.onCreate(savedInstanceState)
    setContentView(R.layout.activity_main)
  }

  override fun onCreateOptionsMenu(menu: Menu?): Boolean {
    menuInflater.inflate(R.menu.menu,menu)
    return super.onCreateOptionsMenu(menu)
  }

  override fun onOptionsItemSelected(item: MenuItem): Boolean {
    var id =item.itemId
    if(id==R.id.search)
    {
      Toast.makeText(this,"Search",Toast.LENGTH_LONG).show()
    }
    if(id==R.id.aboutus)
    {
      Toast.makeText(this,"About Us",Toast.LENGTH_LONG).show()
    }
    if(id==R.id.contactus)
    {
      Toast.makeText(this,"Contact Us",Toast.LENGTH_LONG).show()
    }
    return super.onOptionsItemSelected(item)
  }
}
```

Mainactivity.xml

```xml
<?xml version="1.0" encoding="utf-8"?>
<androidx.constraintlayout.widget.ConstraintLayout
xmlns:android="http://schemas.android.com/apk/res/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 World!"
        app:layout_constraintBottom_toBottomOf="parent"
        app:layout_constraintEnd_toEndOf="parent"
        app:layout_constraintStart_toStartOf="parent"
        app:layout_constraintTop_toTopOf="parent" />

</androidx.constraintlayout.widget.ConstraintLayout>
```



3:06
Appbar
Hello World!



3:07
Appbar
AboutcUs
Contact US
Hello World!

PRACTICAL 6

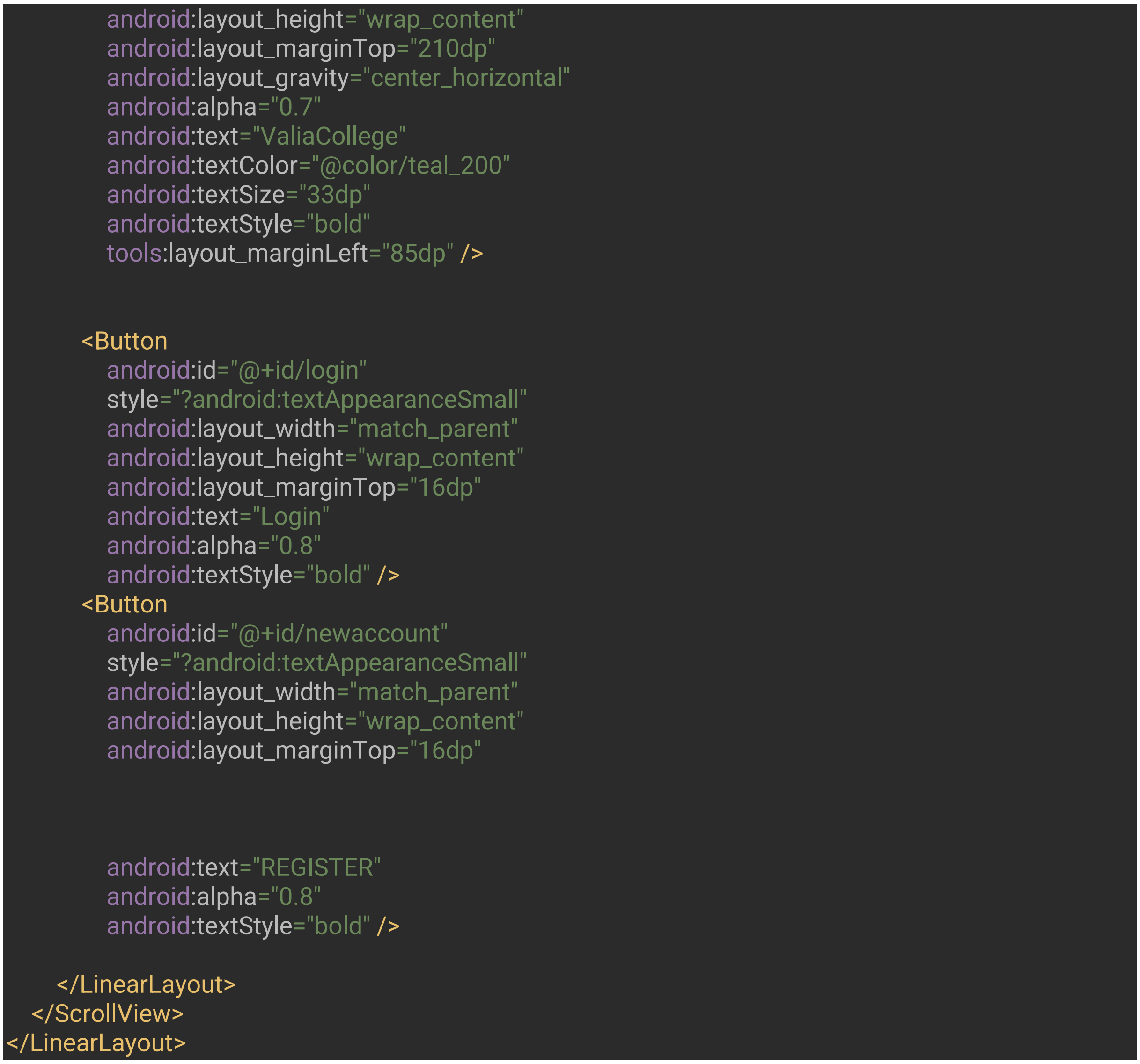
Programming UI elements

MainActivity.kt

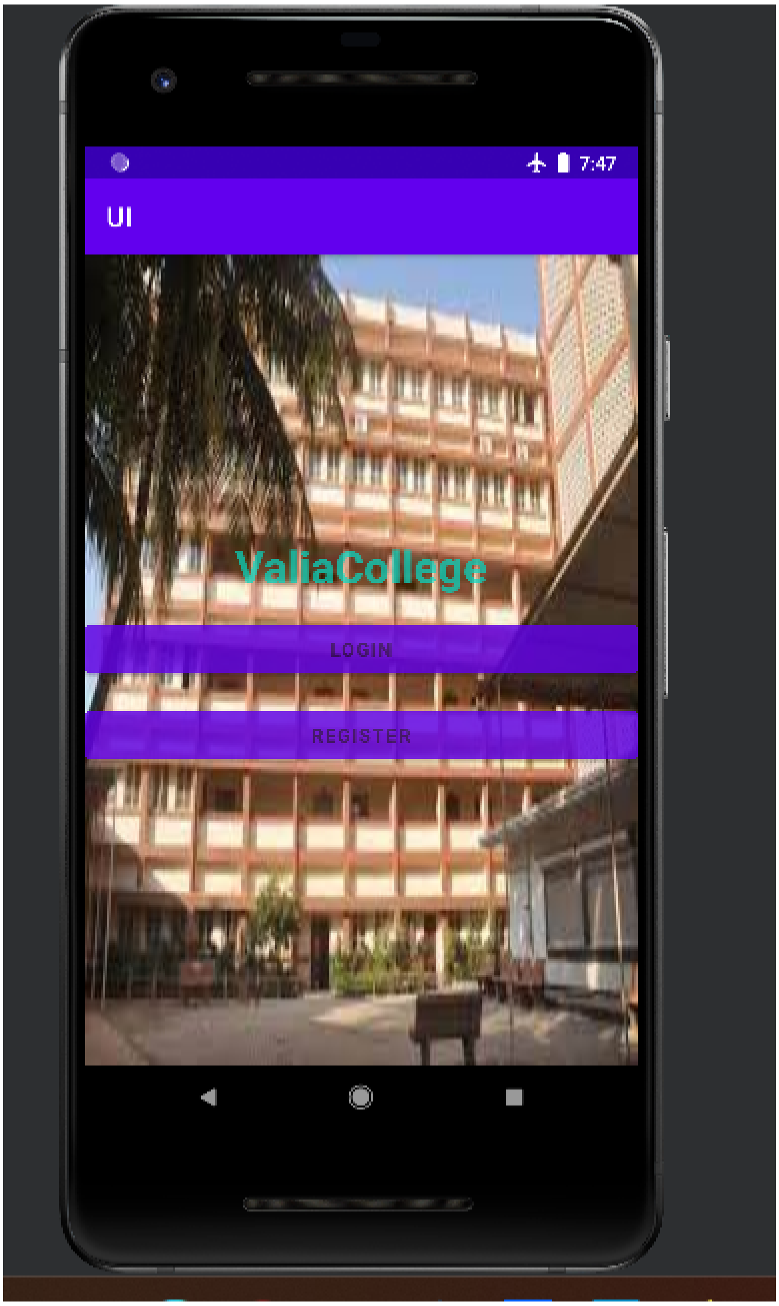
```
package com.example.ui

import android.content.Intent
import androidx.appcompat.app.AppCompatActivity
import android.os.Bundle
import android.widget.Button

class MainActivity : AppCompatActivity() {
   override fun onCreate(savedInstanceState: Bundle?) {
      super.onCreate(savedInstanceState)
      setContentView(R.layout.activity_main)
      val login=findViewById<Button>(R.id.login)
      login.setOnClickListener {
         val intent = Intent(this, LoginActivity::class.java)
         // start your next activity
         startActivity(intent)
      }
     val newaccount=findViewById<Button>(R.id.newaccount)
      newaccount.setOnClickListener {
         val intent = Intent(this, MainActivity::class.java)
         // start your next activity
         startActivity(intent)
      }

   }
}
```

MainActivity.xml

```
<?xml version="1.0" encoding="utf-8"?>
<LinearLayout 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:gravity="center_horizontal"
   android:orientation="vertical"
   android:background="@drawable/valia"
   tools:context=".MainActivity">

   <ScrollView
      android:id="@+id/login_form"
      android:layout_width="match_parent"
      android:layout_height="match_parent">
      <LinearLayout
         android:layout_width="match_parent"
         android:layout_height="wrap_content"
         android:orientation="vertical"
         android:gravity="center">


         <TextView
            android:layout_width="wrap_content"
```



```
        android:layout_height="wrap_content"
        android:layout_marginTop="210dp"
        android:layout_gravity="center_horizontal"
        android:alpha="0.7"
        android:text="ValiaCollege"
        android:textColor="@color/teal_200"
        android:textSize="33dp"
        android:textStyle="bold"
        tools:layout_marginLeft="85dp" />


      <Button
        android:id="@+id/login"
        style="?android:textAppearanceSmall"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:layout_marginTop="16dp"
        android:text="Login"
        android:alpha="0.8"
        android:textStyle="bold" />
      <Button
        android:id="@+id/newaccount"
        style="?android:textAppearanceSmall"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:layout_marginTop="16dp"



        android:text="REGISTER"
        android:alpha="0.8"
        android:textStyle="bold" />

    </LinearLayout>
  </ScrollView>
</LinearLayout>
```



7:47
UI
ValiaCollege
LOGIN
REGISTER

Practical 7

Create an android application to display Alert Dialog on pressing the Back button

Mainactivity.xml

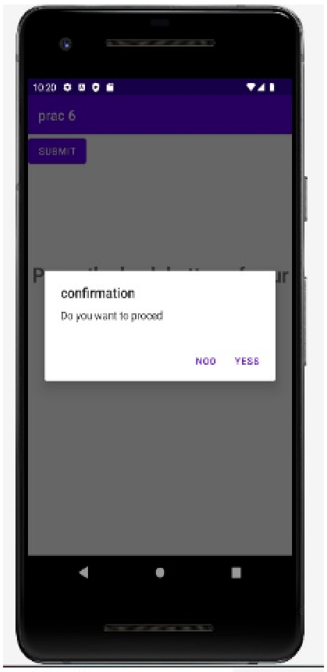
```
<RelativeLayout xmlns:android="http://schemas.android.com/apk/res/android"
   xmlns:app="http://schemas.android.com/apk/res-auto"
   xmlns:tools="http://schemas.android.com/tools"
   android:layout_width="match_parent"
   android:layout_height="match_parent"
   tools:context=".MainActivity">

   <TextView
      android:layout_width="wrap_content"
      android:layout_height="wrap_content"
      android:layout_marginTop="180dp"
      android:gravity="center_horizontal"
      android:text="Press the back button of your phone "
      android:textSize="30dp"
      android:textStyle="bold"/>

   <Button
      android:id="@+id/alert"
      android:layout_width="wrap_content"
      android:layout_height="wrap_content"
      android:text="Submit" />
</RelativeLayout>
```

Main.kt

```
package com.example.prac6

import android.annotation.SuppressLint

import android.content.DialogInterface

import androidx.appcompat.app.AppCompatActivity

import android.os.Bundle

import android.widget.Button

import android.widget.Toast

import androidx.appcompat.app.AlertDialog

class MainActivity : AppCompatActivity() {

   @SuppressLint("MissingInflatedId")

   override fun onCreate(savedInstanceState: Bundle?) {

      super.onCreate(savedInstanceState)

      setContentView(R.layout.activity_main)
```

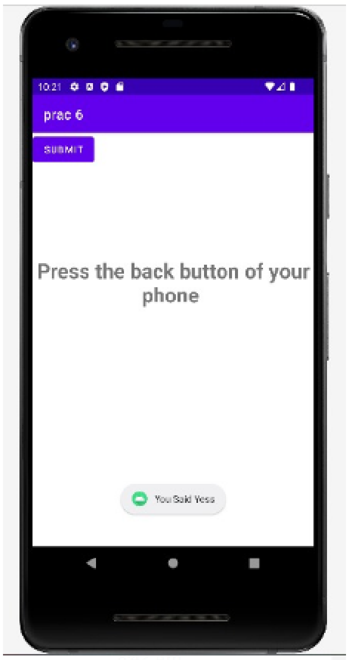
```
    val submit=findViewById<Button>(R.id.alert)
    submit.setOnClickListener {
    val simplealert=AlertDialog.Builder(this).create()
    simplealert.setTitle("confirmation")
    simplealert.setMessage("Do you want to proced")
      simplealert.setButton(AlertDialog.BUTTON_POSITIVE,"Yess")
    {
      dialog:DialogInterface? , which:Int->
      Toast.makeText(this,"You Said Yess",Toast.LENGTH_LONG).show()
    }
      simplealert.setButton(AlertDialog.BUTTON_NEGATIVE,"Noo")
      {
          dialog:DialogInterface? , which:Int->
        Toast.makeText(this,"You Said Noo",Toast.LENGTH_LONG).show()
      }
      simplealert.show()
    }
  }
}
```

Output



confirmation

Do you want to proced



Press the back button of your phone

You Said Yess

PRACTICAL 8

Programs on Intents

Activitymain.xml

```
<?xml version="1.0" encoding="utf-8"?>
<LinearLayout xmlns:android="http://schemas.android.com/apk/res/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"
   android:orientation="horizontal"
   tools:context=".MainActivity">

   <EditText
      android:id="@+id/mytext"
      android:layout_width="wrap_content"
      android:layout_height="wrap_content"
      android:layout_weight="1"
      android:hint="Enter your message here"
      android:layout_marginTop="20dp"
      android:ems="10"
      android:inputType="textPersonName"
      android:text="Name" />

   <Button
      android:id="@+id/Send"
      android:layout_width="wrap_content"
      android:layout_height="wrap_content"
      android:layout_marginTop="20dp"
      android:layout_weight="1"
      android:layout_gravity="center_horizontal"
      android:text="Send" />
```

```
</LinearLayout>
```

Mainactivity.kt

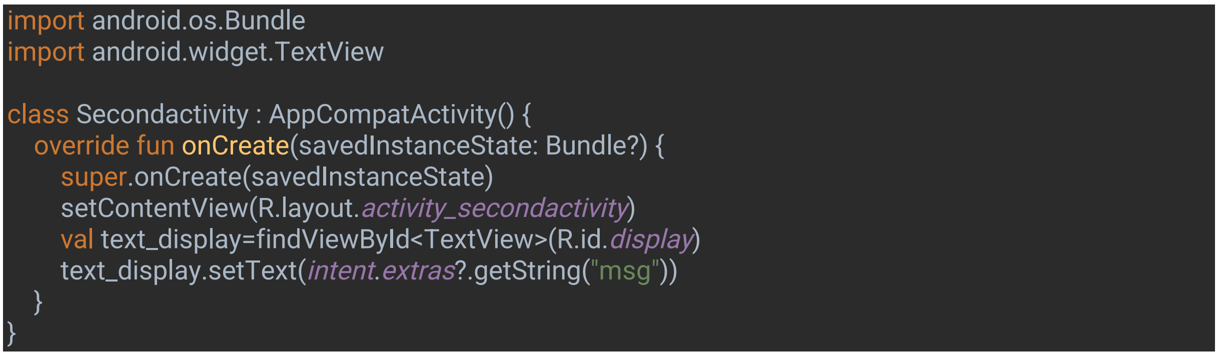
```
package com.example.intent

import android.content.Intent
import androidx.appcompat.app.AppCompatActivity
import android.os.Bundle
import android.view.View
import android.widget.Button
import android.widget.EditText

class MainActivity : AppCompatActivity() {
    override fun onCreate(savedInstanceState: Bundle?) {
        super.onCreate(savedInstanceState)
        setContentView(R.layout.activity_main)
        val message=findViewById<EditText>(R.id.mytext)
        val btn_send=findViewById<Button>(R.id.Send)
        btn_send.setOnClickListener(object : View.OnClickListener
        {
            override fun onClick(v:View?)

            {
                //TODO("Not yet implemented")
                intent= Intent(applicationContext,Secondactivity::class.java)
                intent.putExtra("msg",message.text.toString())
                startActivity(intent)
            }

            })


}
}
```

Secondactivity.xml

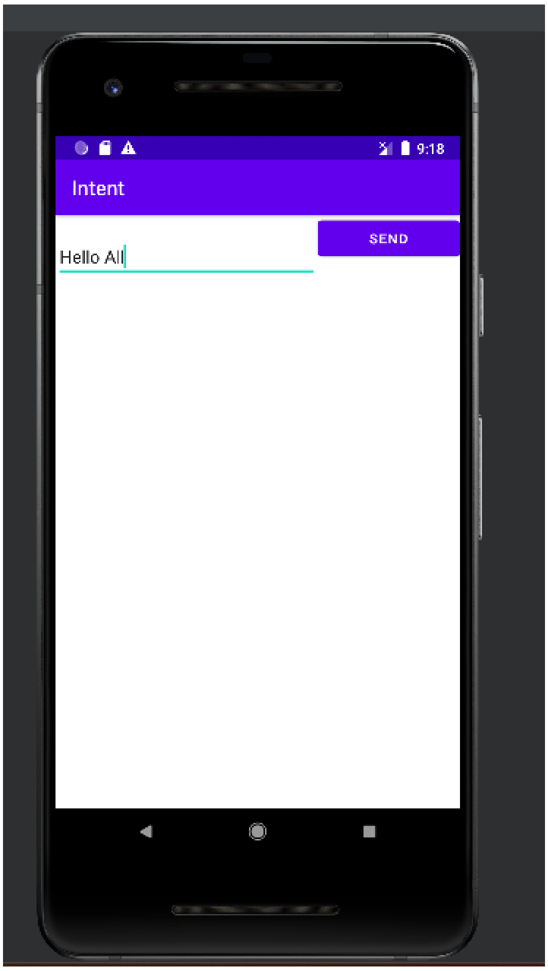
```
<?xml version="1.0" encoding="utf-8"?>
<LinearLayout xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:app="http://schemas.android.com/apk/res-auto"
    xmlns:tools="http://schemas.android.com/tools"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    tools:context=".Secondactivity">
    <TextView
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:id="@+id/display"
        android:layout_marginTop="40dp"/>
</LinearLayout>
```

Secondactivity.kt

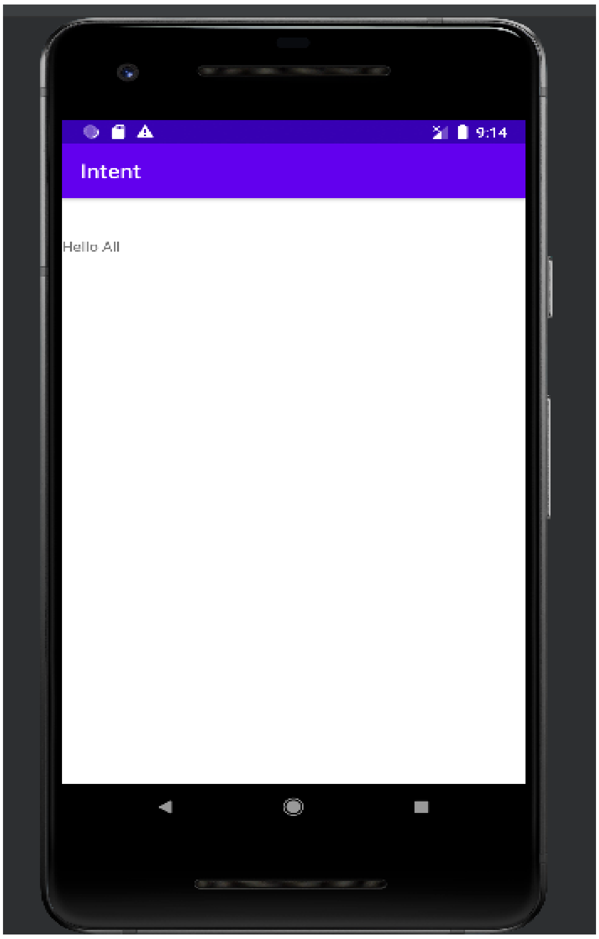
```
package com.example.intent

import androidx.appcompat.app.AppCompatActivity
```



```
import android.os.Bundle
import android.widget.TextView

class Secondactivity : AppCompatActivity() {
  override fun onCreate(savedInstanceState: Bundle?) {
    super.onCreate(savedInstanceState)
    setContentView(R.layout.activity_secondactivity)
    val text_display=findViewById<TextView>(R.id.display)
    text_display.setText(intent.extras?.getString("msg"))
  }
}
```



Hello All



Hello All

Practical 9

Programs on Services, notification, and broadcast receivers

Activity.xml

```
?xml version="1.0" encoding="utf-8"?>
<android.support.constraint.ConstraintLayout
xmlns:android="http://schemas.android.com/apk/res/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"
   android:background="@color/white"
   tools:context="in.eyehunt.androidbroadcasts.MainActivity">


   <ImageView
      android:id="@+id/imageView"
      android:layout_width="40dp"
      android:layout_height="40dp"
      android:layout_margin="8dp"
      android:layout_marginTop="16dp"
      app:layout_constraintStart_toStartOf="parent"
      app:layout_constraintTop_toTopOf="parent"
      app:srcCompat="@mipmap/airplane" />

   <TextView
      android:id="@+id/textView"
      android:layout_width="300dp"
      android:layout_height="36dp"
      android:layout_marginEnd="8dp"
      android:layout_marginStart="8dp"
      android:gravity="center_vertical"
      android:text="Flight Mode"
      android:textColor="@color/white"
      android:textSize="24dp"
      app:layout_constraintEnd_toEndOf="parent"
      app:layout_constraintStart_toEndOf="@+id/imageView"
      app:layout_constraintTop_toTopOf="@+id/imageView" />
</LinearLayout>
```

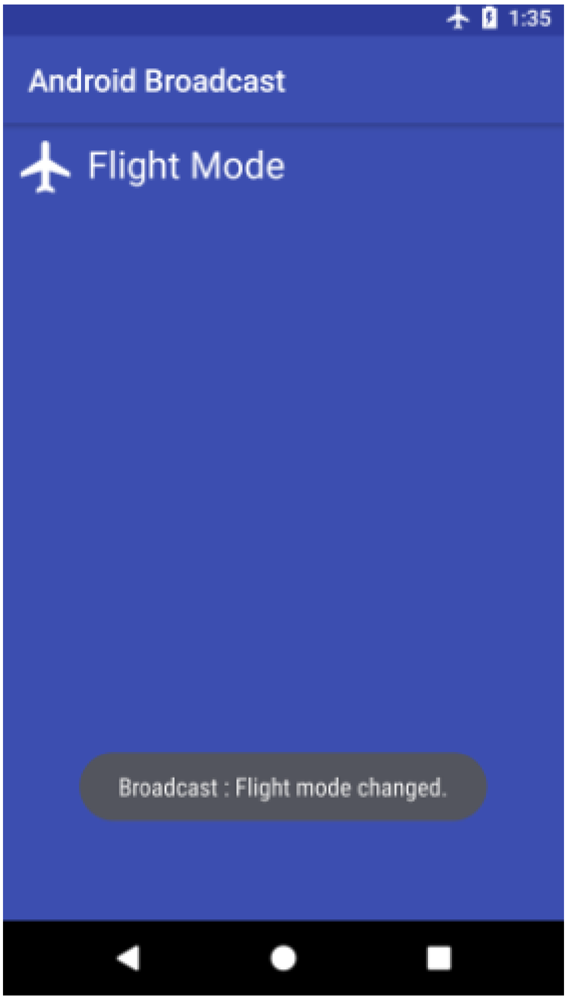
AndroidManifest.xml

```
<?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="in.eyehunt.androidbroadcasts">
   <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/CardView">
     <activity android:name="com.example.broadcastreceiver.MainActivity"
        android:exported="true">
         <intent-filter>
            <action android:name="android.intent.action.MAIN" />

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

      <receiver
         android:name="com.example.broadcastreceiver.MyReceiver"
```

```xml
            android:enabled="true"
            android:exported="true"
            tools:ignore="MissingClass">
            <intent-filter>
                <action android:name="android.intent.action.AIRPLANE_MODE"/>
            </intent-filter>
        </receiver>
    </application>

</manifest>
```

Mainactivity.xml

```xml
<?xml version="1.0" encoding="utf-8"?>
<LinearLayout xmlns:android="http://schemas.android.com/apk/res/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"
    android:background="@color/purple_200"
    tools:context=".MainActivity">

    <ImageView
        android:id="@+id/imageView"
        android:layout_width="40dp"
        android:layout_height="40dp"
        android:layout_margin="8dp"
        android:layout_marginTop="16dp"
        app:layout_constraintStart_toStartOf="parent"
        app:layout_constraintTop_toTopOf="parent"
        app:srcCompat="@mipmap/airplane" />

    <TextView
        android:id="@+id/textView"
        android:layout_width="300dp"
        android:layout_height="36dp"
        android:layout_marginEnd="8dp"
        android:layout_marginStart="8dp"
        android:gravity="center_vertical"
        android:text="Flight Mode"
        android:textColor="@color/white"
        android:textSize="24dp"
        app:layout_constraintEnd_toEndOf="parent"
        app:layout_constraintStart_toEndOf="@+id/imageView"
        app:layout_constraintTop_toTopOf="@+id/imageView" />
</LinearLayout>
```

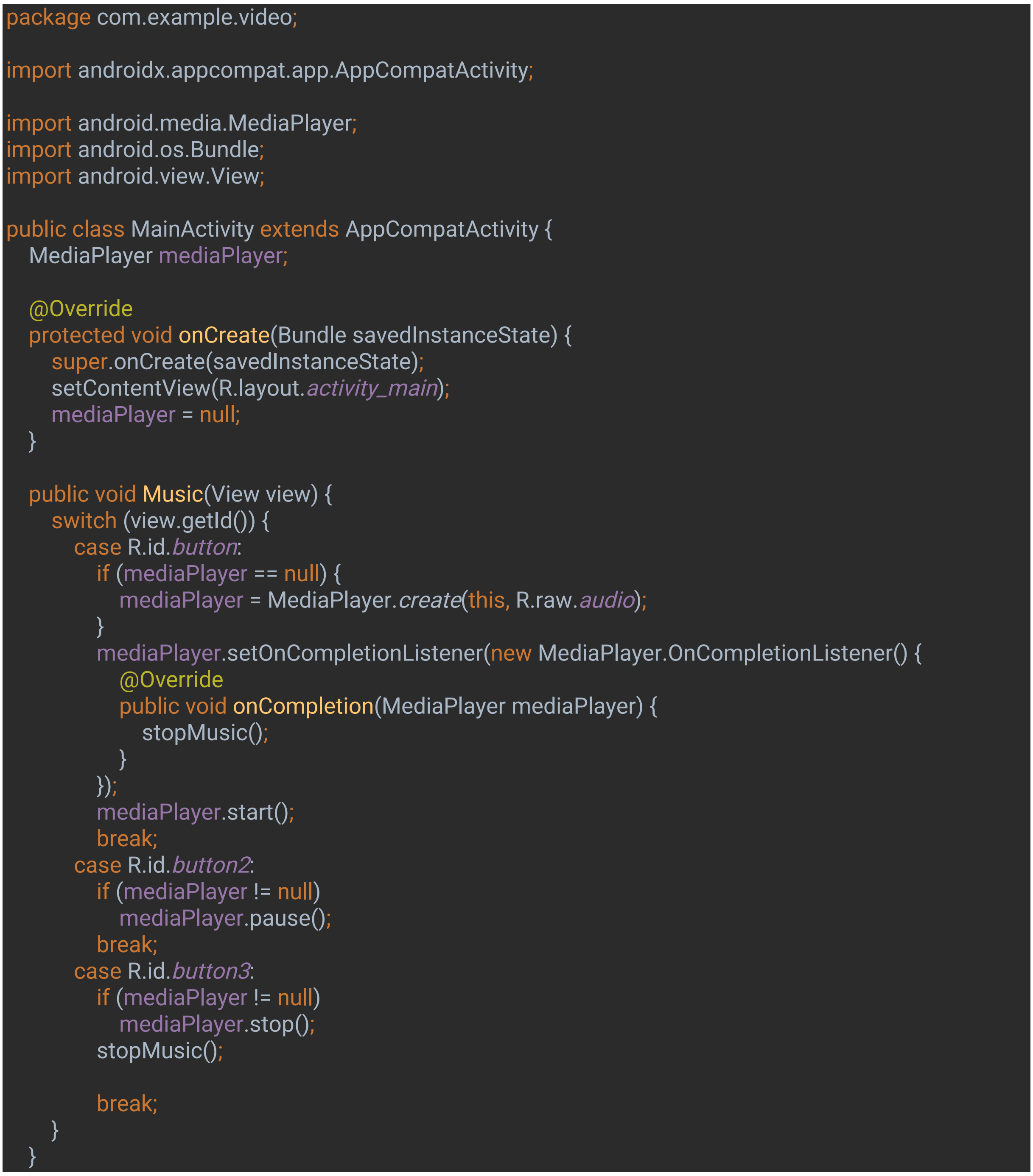


Broadcast : Flight mode changed.

Practical 10

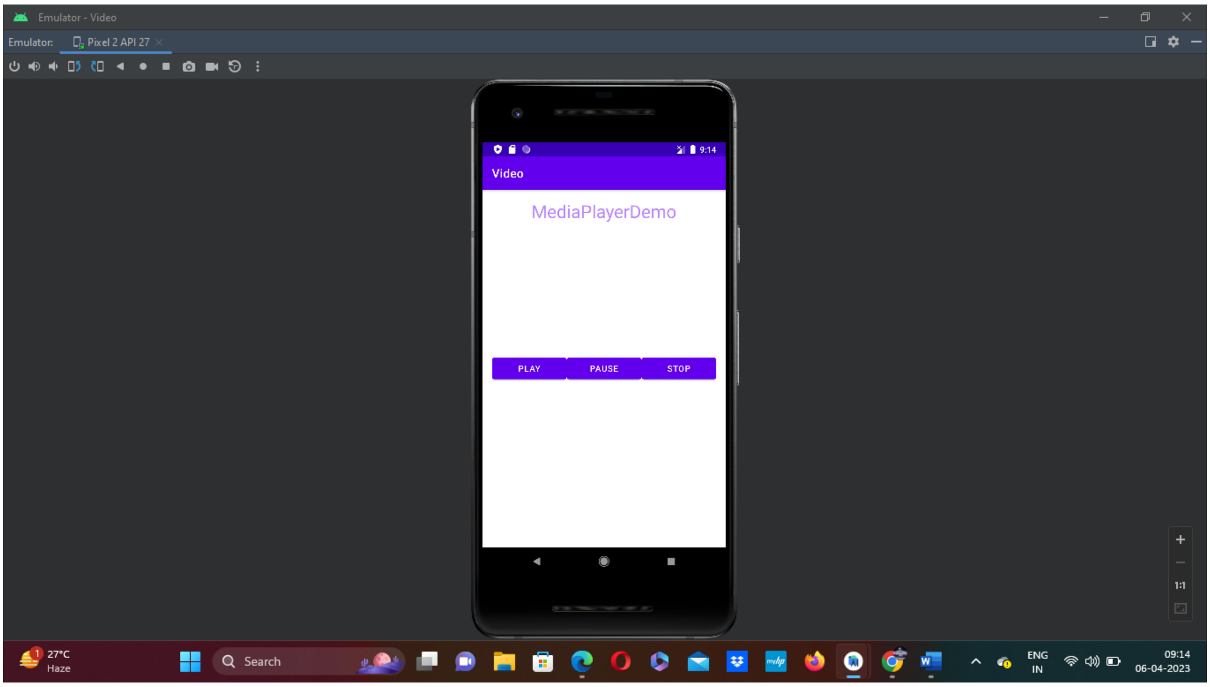
Create the media API in android to play an audio file with buttons play,start,stop(code is in java)

To add music in android studio Add a new directory by right click on res and then click New->Directory add the directory with the name raw and then paste the music file in the directory.



```java
package com.example.video;

import androidx.appcompat.app.AppCompatActivity;

import android.media.MediaPlayer;
import android.os.Bundle;
import android.view.View;

public class MainActivity extends AppCompatActivity {
    MediaPlayer mediaPlayer;

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

    public void Music(View view) {
        switch (view.getId()) {
            case R.id.button:
                if (mediaPlayer == null) {
                    mediaPlayer = MediaPlayer.create(this, R.raw.audio);
                }
                mediaPlayer.setOnCompletionListener(new MediaPlayer.OnCompletionListener() {
                    @Override
                    public void onCompletion(MediaPlayer mediaPlayer) {
                        stopMusic();
                    }
                });
                mediaPlayer.start();
                break;
            case R.id.button2:
                if (mediaPlayer != null)
                    mediaPlayer.pause();
                break;
            case R.id.button3:
                if (mediaPlayer != null)
                    mediaPlayer.stop();
                stopMusic();

                break;
        }
    }
    public void stopMusic() {
        mediaPlayer.release();
        mediaPlayer = null;
    }

    @Override
    public void onStop()
    {
        super.onStop();
    }
}
```

```
<?xml version="1.0" encoding="utf-8"?>
<RelativeLayout xmlns:android="http://schemas.android.com/apk/res/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"
    android:padding="16dp"
    tools:context=".MainActivity">


    <TextView
        android:id="@+id/textView"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_centerHorizontal="true"
        android:textSize="30sp"
        android:textColor="@color/purple_200"
        android:text="MediaPlayerDemo" />

    <LinearLayout
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:orientation="horizontal"
        android:layout_centerVertical="true">

        <Button
            android:id="@+id/button"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:layout_weight="1"
            android:text="Play"
            android:onClick="Music"/>

        <Button
            android:id="@+id/button2"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:layout_weight="1"
            android:text="Pause"
            android:onClick="Music"/>

        <Button
            android:id="@+id/button3"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:layout_weight="1"
            android:text="Stop"
            android:onClick="Music"/>
    </LinearLayout>
</RelativeLayout>
```



Practical 11

Create the Telephone API in android

Activity.xml

```
<?xml version="1.0" encoding="utf-8"?>
<LinearLayout xmlns:android="http://schemas.android.com/apk/res/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"
   android:orientation="horizontal"
   tools:context=".MainActivity">


   <Button
      android:id="@+id/placecall"
      android:layout_width="match_parent"
      android:layout_height="wrap_content"
      android:layout_marginTop="200dp"
      android:text="Place Call" />
</LinearLayout>
```

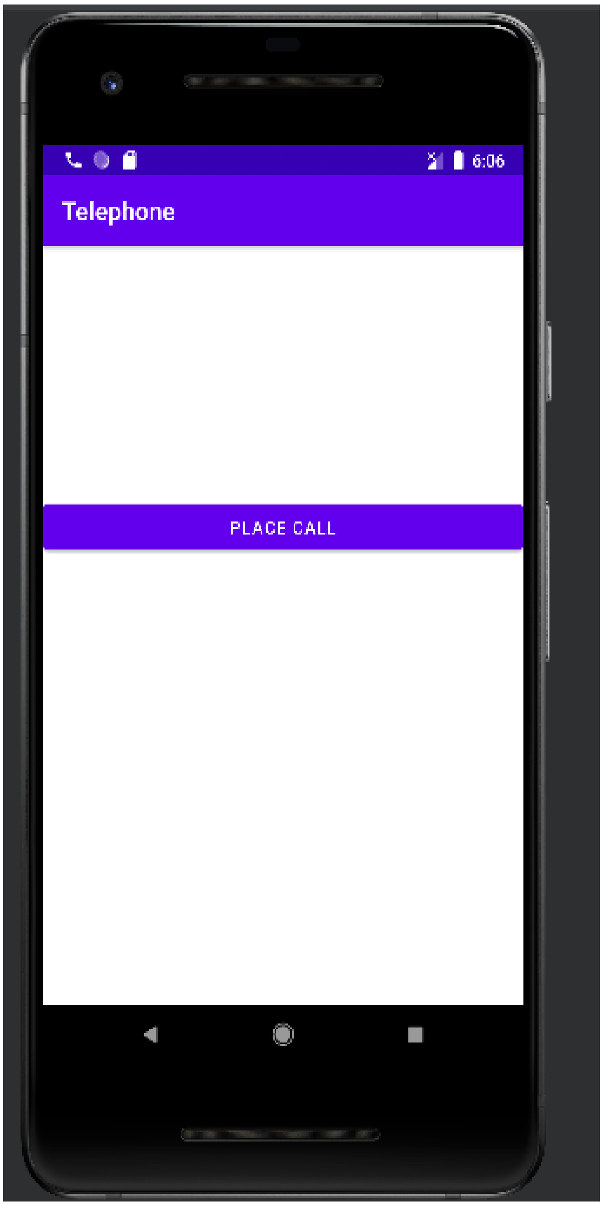
Mainactivity.kt

```
package com.example.telephone

import android.content.Intent
import android.content.pm.PackageManager
import android.net.Uri
import androidx.appcompat.app.AppCompatActivity
import android.os.Bundle
import android.widget.Button
import androidx.core.app.ActivityCompat

class MainActivity : AppCompatActivity() {
    val mobno: String = "1234567890"
    val REQUEST_PHONE_CALL = 1
    override fun onCreate(savedInstanceState: Bundle?) {
        super.onCreate(savedInstanceState)
        setContentView(R.layout.activity_main)
        val call = findViewById<Button>(R.id.placecall)
        call.setOnClickListener {
            if (ActivityCompat.checkSelfPermission(
                    this,
                    android.Manifest.permission.CALL_PHONE
                ) != PackageManager.PERMISSION_GRANTED
            ) {
                ActivityCompat.requestPermissions(
                    this,
                    arrayOf(android.Manifest.permission.CALL_PHONE),
                    REQUEST_PHONE_CALL
                )
            } else {
                makecall()
            }
        }
    }
         private fun makecall() {
                val intent = Intent(Intent.ACTION_CALL, Uri.fromParts("tel", mobno, null))
                startActivity(intent)
            }

            override fun onRequestPermissionsResult(requestCode: Int, permissions: Array<out String>,
grantResults: IntArray) {
                super.onRequestPermissionsResult(requestCode, permissions, grantResults)
                super.onRequestPermissionsResult(requestCode, permissions, grantResults)
                val REQUEST_PHONE_CALL = null
                if (requestCode == REQUEST_PHONE_CALL) {
                    makecall()
                }
            }
        }
```

Manifest.xml

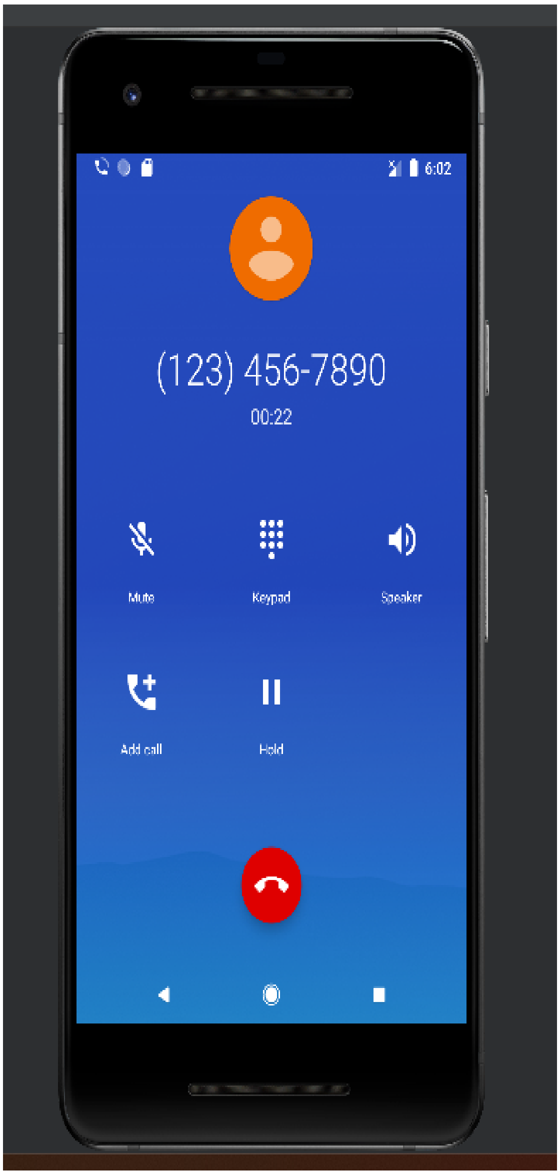
```
<?xml version="1.0" encoding="utf-8"?>
<manifest xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:tools="http://schemas.android.com/tools">
<uses-permission android:name="android.permission.CALL_PHONE"></uses-permission>
    <application
        android:allowBackup="true"
        android:dataExtractionRules="@xml/data_extraction_rules"
        android:fullBackupContent="@xml/backup_rules"
        android:icon="@mipmap/ic_launcher"
        android:label="@string/app_name"
        android:supportsRtl="true"
        android:theme="@style/Theme.Telephone"
        tools:targetApi="31">
        <activity
            android:name=".MainActivity"
            android:exported="true">
            <intent-filter>
                <action android:name="android.intent.action.MAIN" />

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

```
</manifest>
```



Telephone

PLACE CALL



(123) 456-7890

00:22