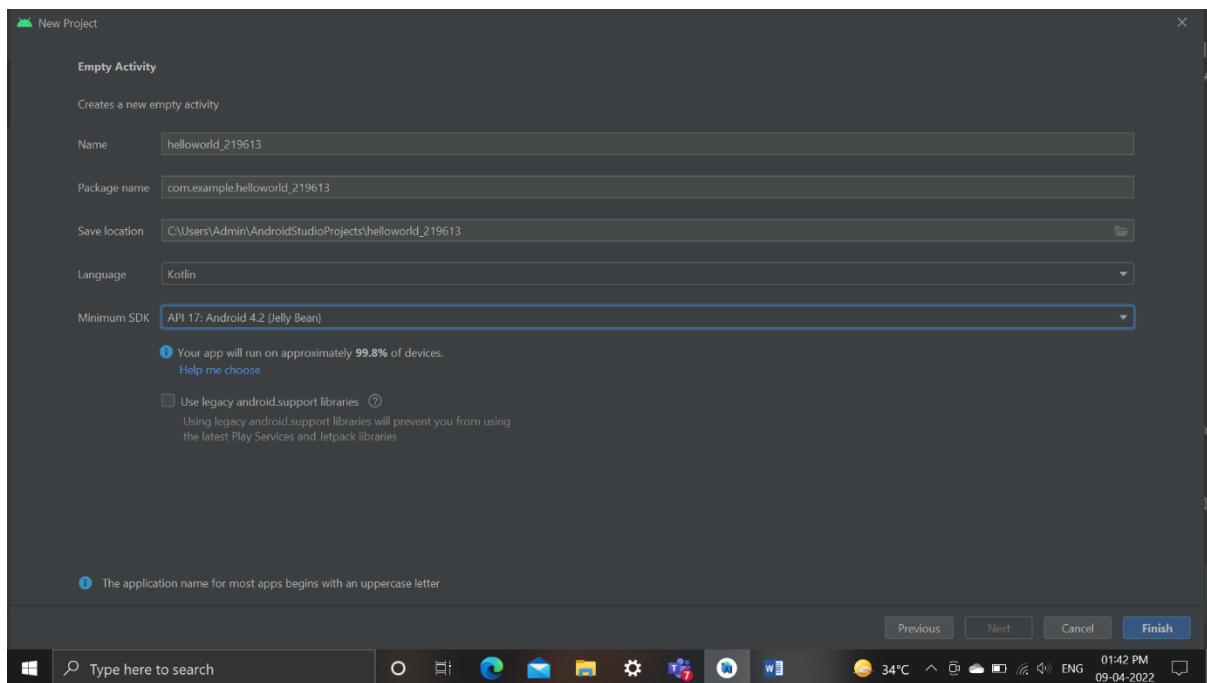
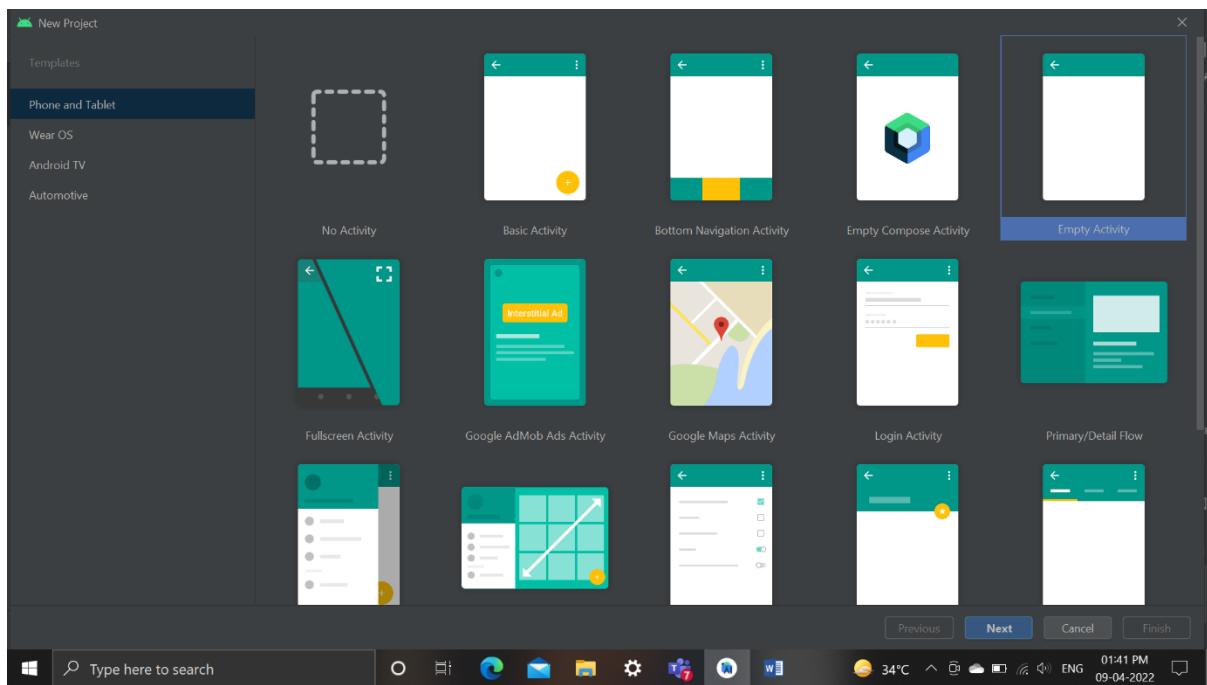


ANDROID MOBILE PROGRAMMING MANUAL

PRACTICAL 1

Creating a project:

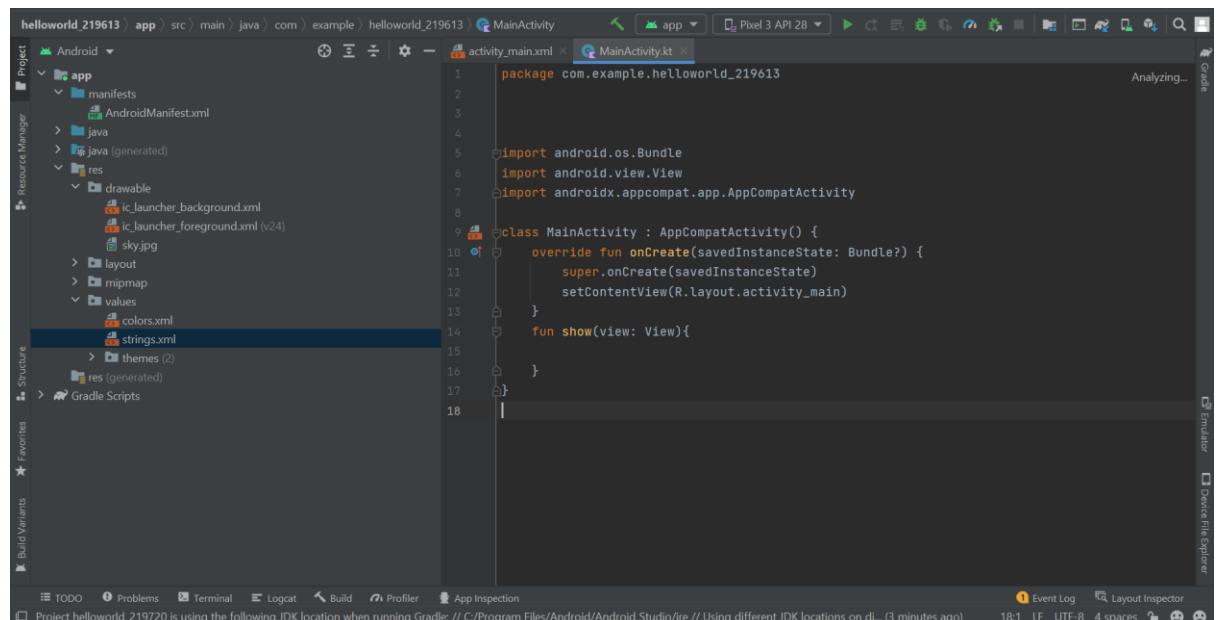


Activity Main.kt

```
package com.example.helloworld

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)
    }
}
```



activity Main.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"
    android:background="@drawable/sky"
    tools:context=".MainActivity">

    <TextView
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="@string/disp"
        app:layout_constraintBottom_toBottomOf="parent"
        app:layout_constraintLeft_toLeftOf="parent"
        app:layout_constraintRight_toRightOf="parent"
        app:layout_constraintTop_toTopOf="parent" />
</androidx.constraintlayout.widget.ConstraintLayout>
```

The screenshot shows the Android Studio interface. On the left is the Project Navigational Bar with sections for Project, Resource Manager, Structure, Favorites, and Build Variants. The main area displays the file structure under 'app' and the XML code for 'activity_main.xml'. The code defines a RelativeLayout containing an ImageView and a Button. The ImageView has a background image named 'sky.jpg'. The Button has a click listener with the text 'Click on me'. The bottom status bar shows a message about JDK location and some developer tools.

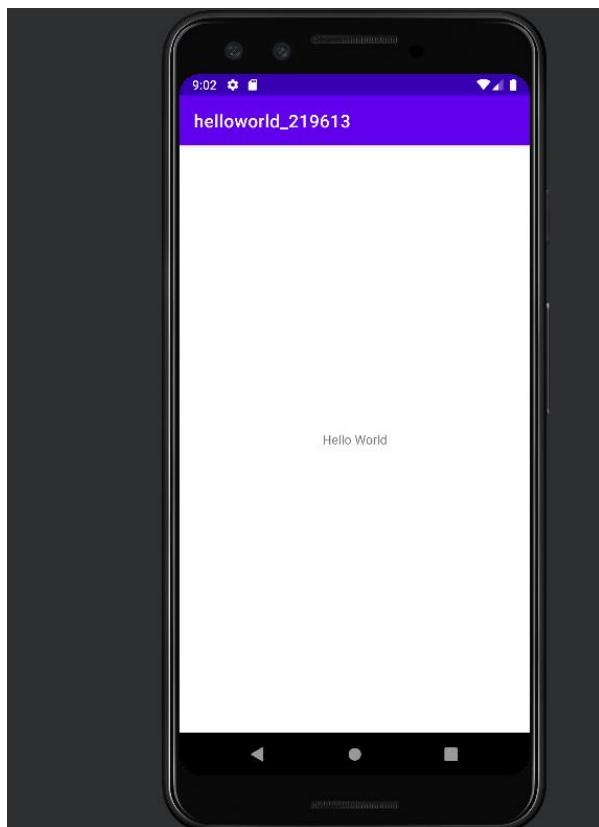
```
<?xml version="1.0" encoding="utf-8"?>
<RelativeLayout
    xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:tools="http://schemas.android.com/tools"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    tools:context=".MainActivity"
    android:background="@drawable/sky">

    <ImageView
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:src="@drawable/sky"
        android:id="@+id/img"
        android:layout_centerHorizontal="true"
        android:layout_alignParentTop="true"
        android:layout_marginTop="117dp"/>

    <Button
        android:id="@+id	btn"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="Click on me"
        android:layout_below="@+id/img"/>

```

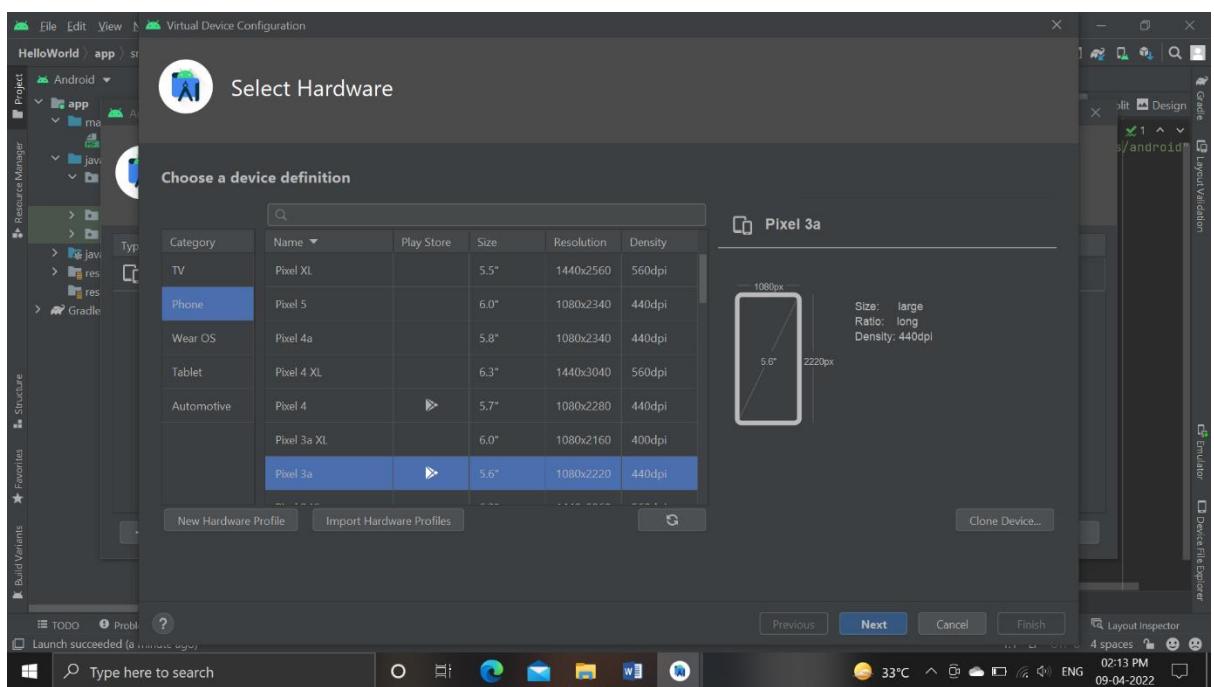
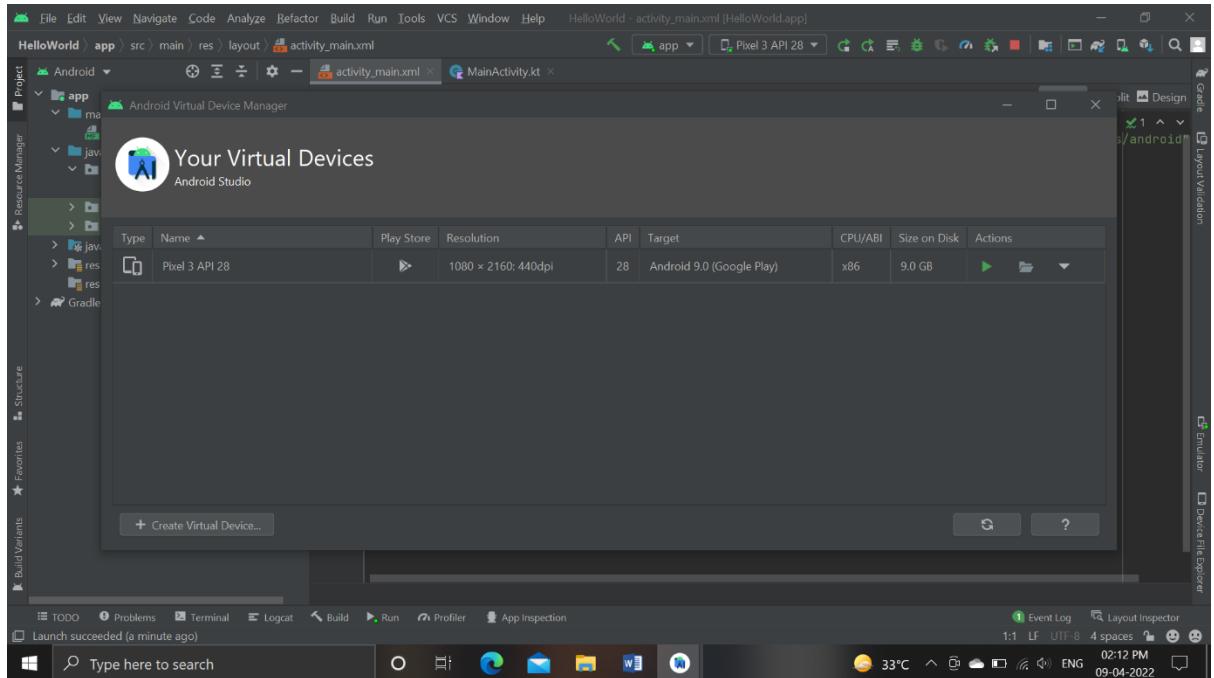
Application in AVD:

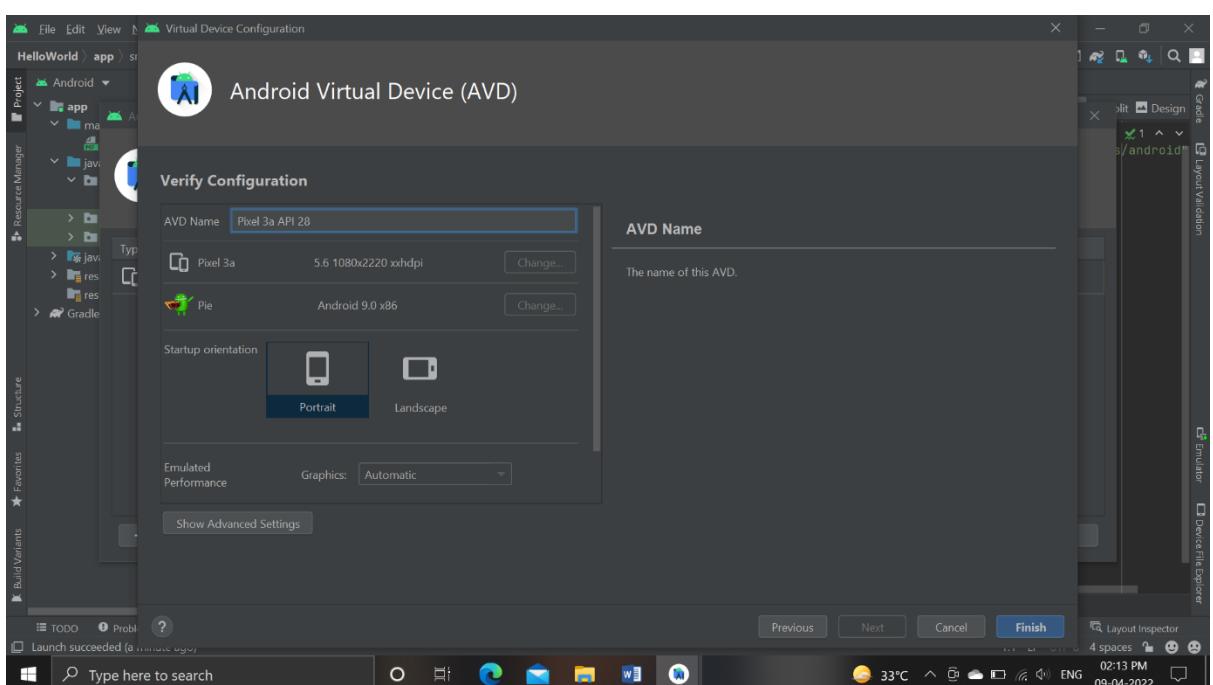
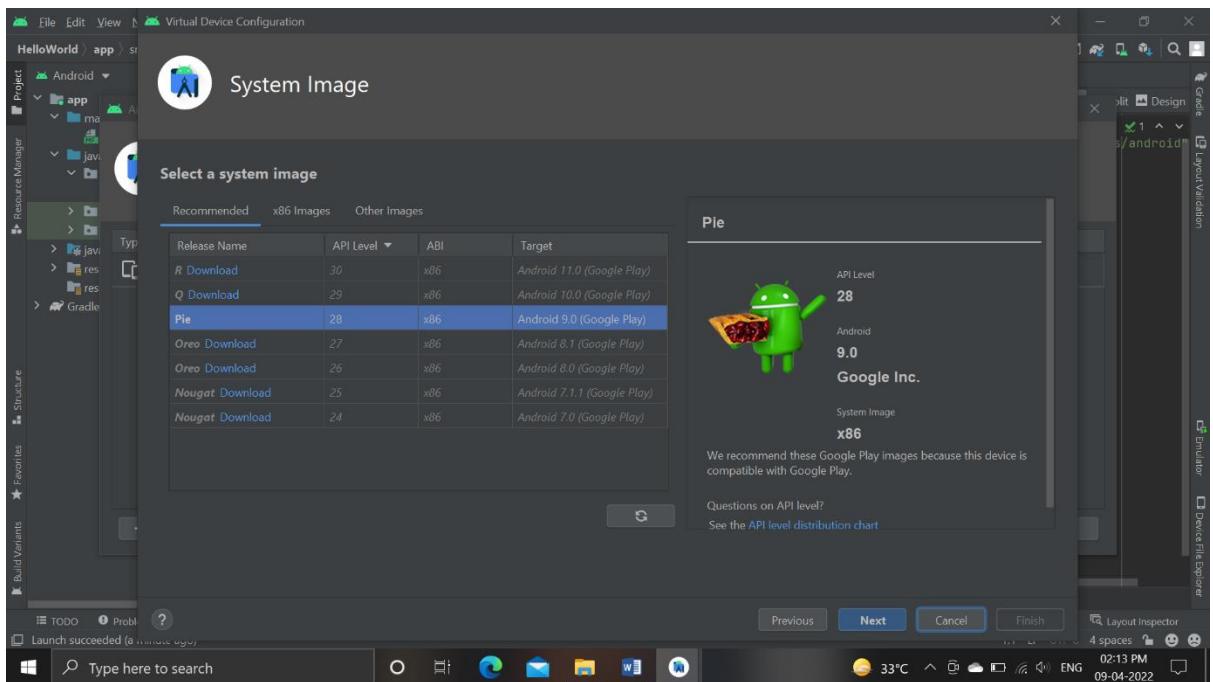


Create and manage virtual devices:

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

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





PRACTICAL 2

Example

Demonstration the use of various resource files

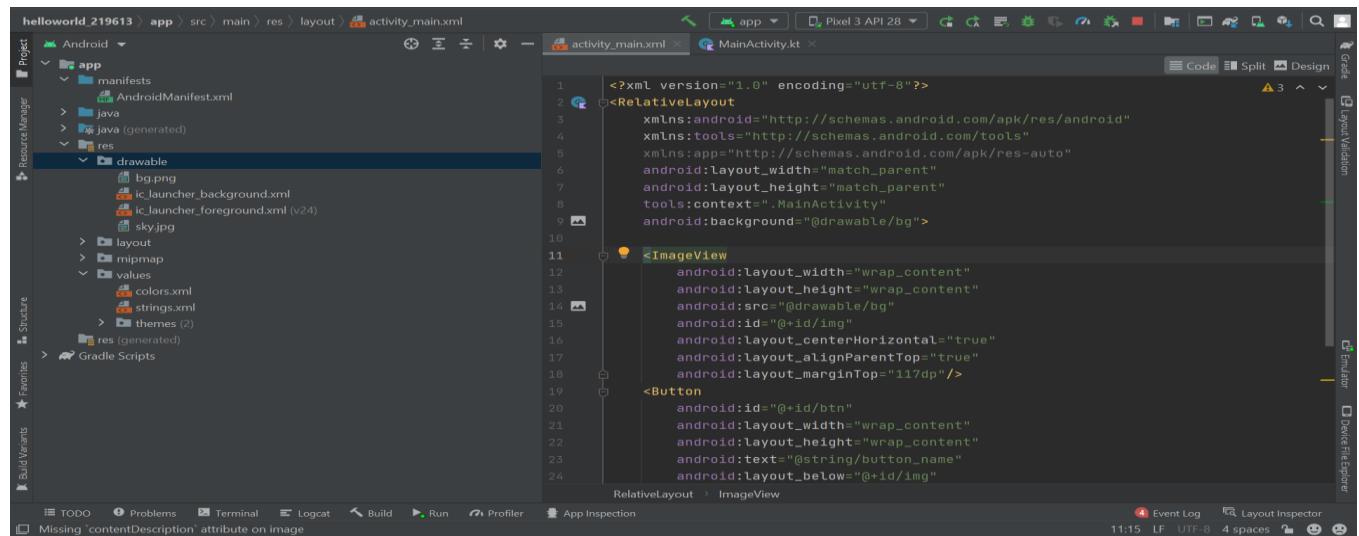
activity_main.xml

```
<?xml version="1.0" encoding="utf-8"?>
<RelativeLayout
    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/sky">

    <ImageView
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:src="@drawable/sky"
        android:id="@+id/img"
        android:layout_centerHorizontal="true"
        android:layout_alignParentTop="true"
        android:layout_marginTop="117dp"/>

    <Button
        android:id="@+id	btn"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="@string/button_name"
        android:layout_below="@+id/img"
        android:layout_centerHorizontal="true"
        android:background="@color(btn_color"
        android:layout_marginTop="123dp"
        android:onClick="show"/>

</RelativeLayout>
```



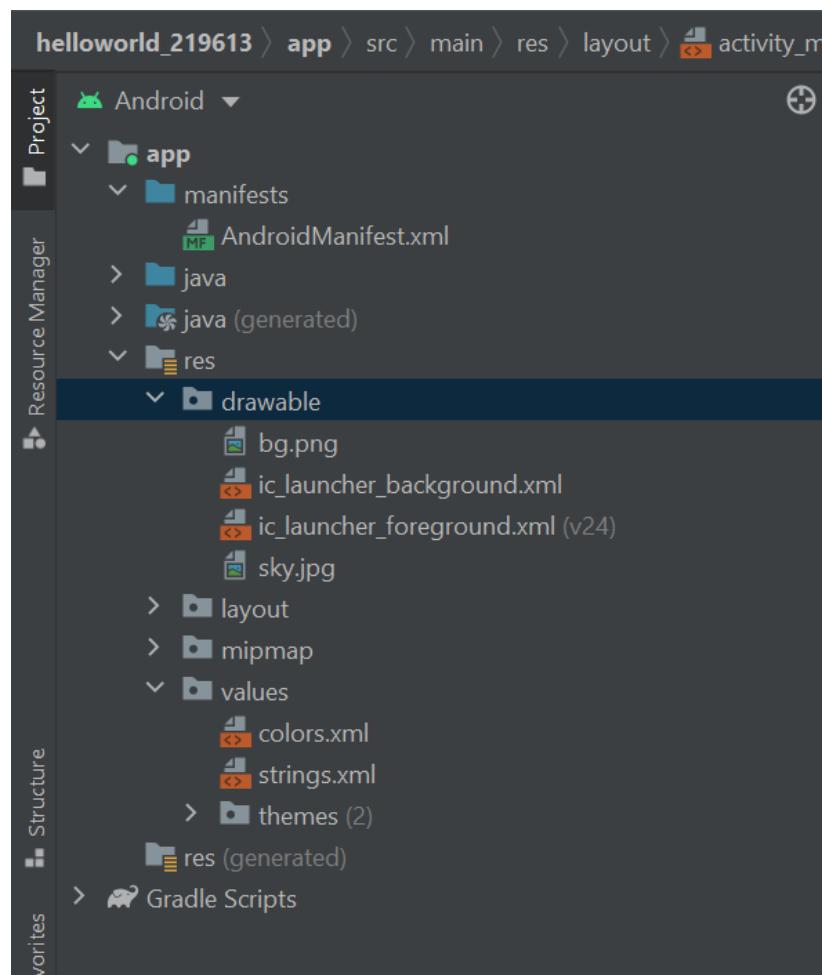
colors.xml

```
<?xml version="1.0" encoding="utf-8"?>
<resources>
    <color name="purple_200">#FFBB86FC</color>
    <color name="purple_500">#FF6200EE</color>
    <color name="purple_700">#FF3700B3</color>
    <color name="teal_200">#FF03DAC5</color>
    <color name="teal_700">#FF018786</color>
    <color name="black">#FF000000</color>
    <color name="white">#FFFFFF</color>
    <color name="btn_color">#FFC49662</color>
</resources>
```

string.xml

```
<resources>
    <string name="app_name">helloworld</string>
    <string name="button_name">Click on me</string>
</resources>
```

In Drawable add the image which you want to display



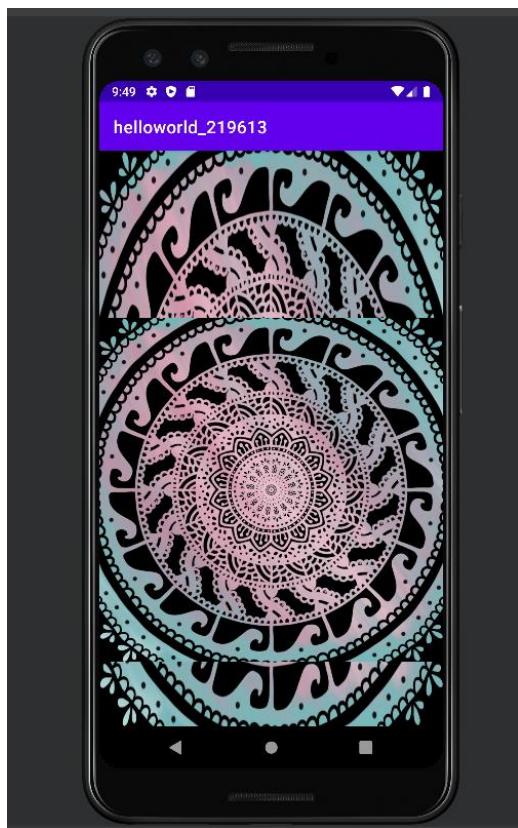
Mainactivity.kt

```
package com.example.helloworld

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

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

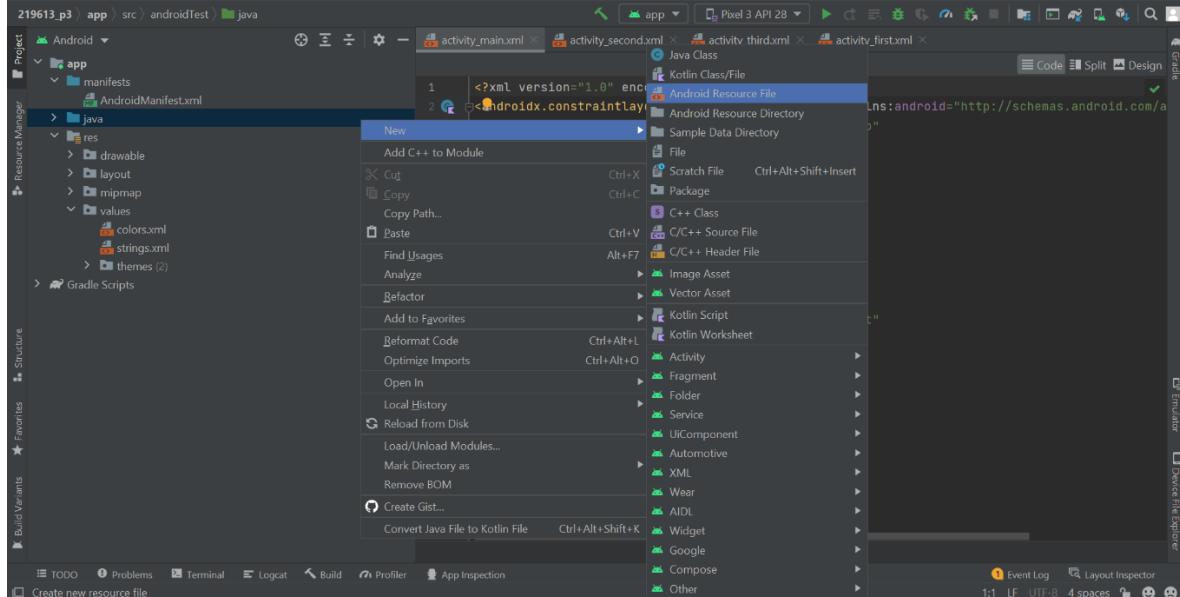
Output:



PRACTICAL 3

Right click on Java > New > Android Resource File > activity_first.xml

Similarly create activity_second.xml and activity_third.xml



Activity_first.xml:

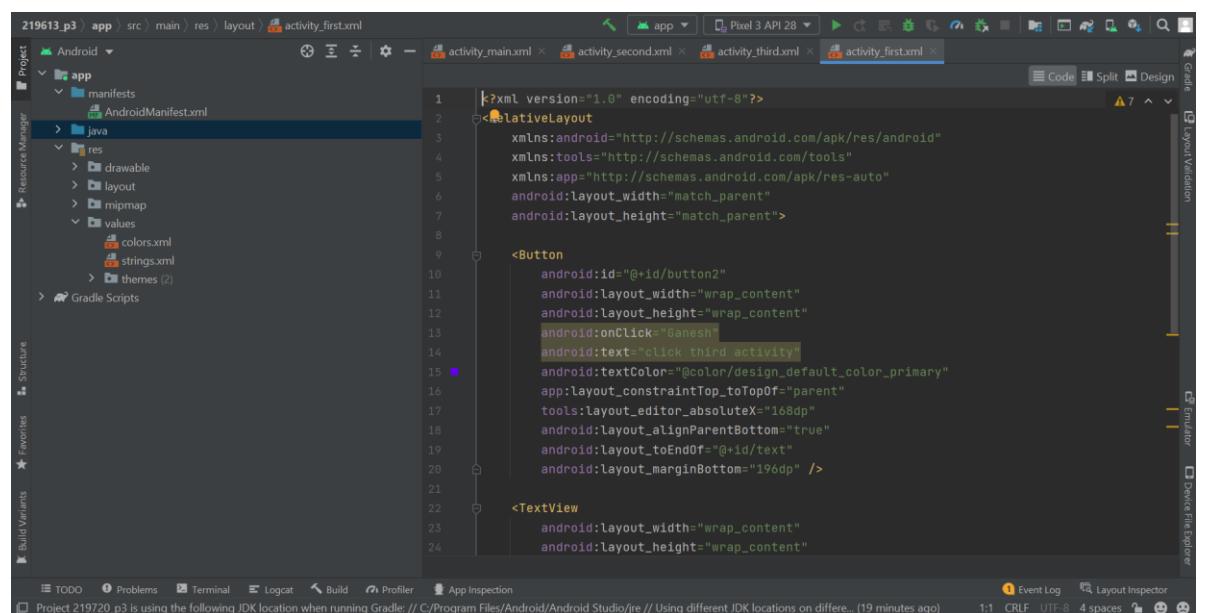
```
<?xml version="1.0" encoding="utf-8"?>
<RelativeLayout
    xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:tools="http://schemas.android.com/tools"
    xmlns:app="http://schemas.android.com/apk/res-auto"
    android:layout_width="match_parent"
    android:layout_height="match_parent">

    <Button
        android:id="@+id/button2"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:onClick="Ganesh"
        android:text="click third activity"
        android:textColor="@color/design_default_color_primary"
        app:layout_constraintTop_toTopOf="parent"
        tools:layout_editor_absoluteX="168dp"
        android:layout_alignParentBottom="true"
        android:layout_toEndOf="@+id/text"
        android:layout_marginBottom="196dp" />

    <TextView
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="This s my first app!"
        android:id="@+id/text"
        tools:layout_editor_absoluteY="8dp"
        tools:layout_editor_absoluteX="8dp" />
    <Button
        android:layout_width="wrap_content"
```

```
        android:layout_height="wrap_content"
        android:id="@+id/button"
        android:text="click second activity"
        android:textColor="@color/design_default_color_primary"
        android:onClick="Ganesh"
        tools:layout_editor_absoluteX="168dp"
        app:layout_constraintTop_toTopOf="parent"
        android:layout_above="@+id/button2"
        android:layout_alignStart="@+id/button2"
        android:layout_marginBottom="40dp" />

</RelativeLayout>
```



Activity_second.xml:

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

    <TextView
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_margin="20pt"
        android:text="second activity is working...."
        android:textAllCaps="true"
        android:textColor="@color/design_default_color_primary_dark"/>

</LinearLayout>
```

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

    <TextView
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_margin="20pt"
        android:text="Second activity is working....."
        android:textAllCaps="true"
        android:textColor="@color/design_default_color_primary_dark"/>

</LinearLayout>
```

Activity_third.xml:

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

    <TextView
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_margin="20pt"
        android:text="Third activity is working ....."
        android:textAllCaps="true"
        android:textColor="@color/design_default_color_primary"/>

</LinearLayout>
```

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

    <TextView
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_margin="20pt"
        android:text="Third activity is working ....."
        android:textAllCaps="true"
        android:textColor="@color/design_default_color_primary"/>

</LinearLayout>
```

PRACTICAL 4

Programs related to different Layouts

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

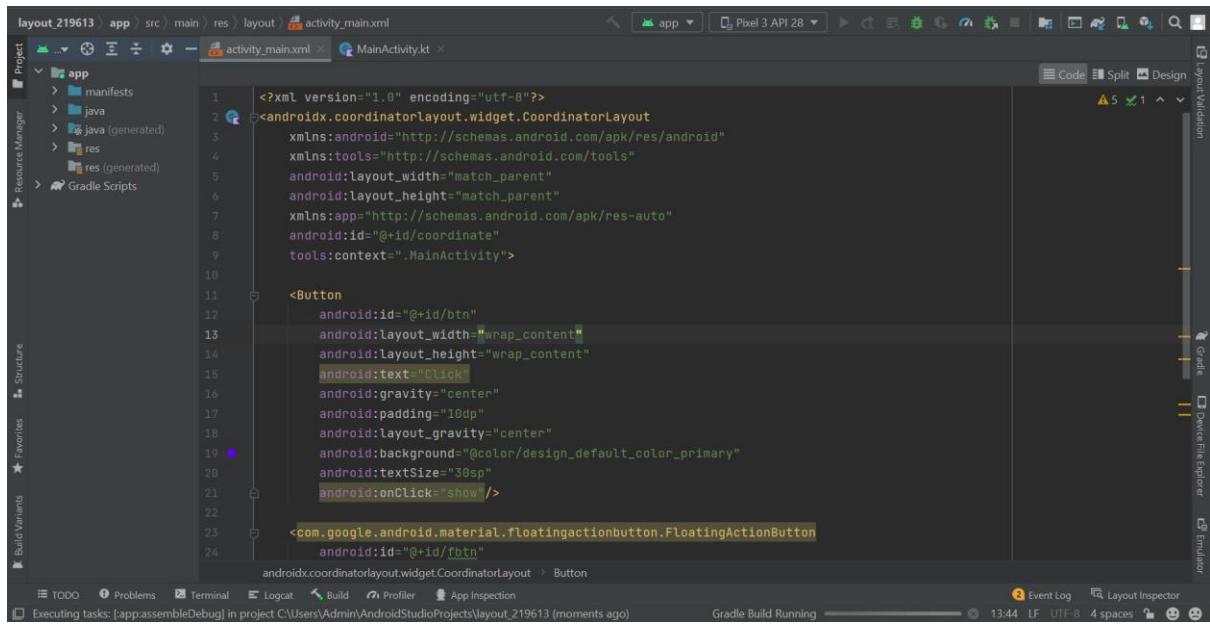
Coordinate Layout

activity_main.xml

```
<?xml version="1.0" encoding="utf-8"?>
<androidx.coordinatorlayout.widget.CoordinatorLayout
    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"
    xmlns:app="http://schemas.android.com/apk/res-auto"
    android:id="@+id/coordinate"
    tools:context=".MainActivity">

    <Button
        android:id="@+id	btn"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="Click"
        android:gravity="center"
        android:padding="10dp"
        android:layout_gravity="center"
        android:background="@color/design_default_color_primary"
        android:textSize="30sp"
        android:onClick="show"/>

    <com.google.android.material.floatingactionbutton.FloatingActionButton
        android:id="@+id/fbtn"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_gravity="bottom|right"
        android:layout_marginRight="20dp"
        android:layout_marginBottom="30dp"
        android:src="@drawable/plus_icon"
        app:fabCustomSize="75dp"
        app:maxImageSize="60dp"
        android:scaleType="center"/>
</androidx.coordinatorlayout.widget.CoordinatorLayout>
```



MainActivity.kt

```
package com.example.layout

import androidx.appcompat.app.AppCompatActivity
import android.os.Bundle
import android.view.View
import androidx.coordinatorlayout.widget.CoordinatorLayout
import com.google.android.material.snackbar.Snackbar

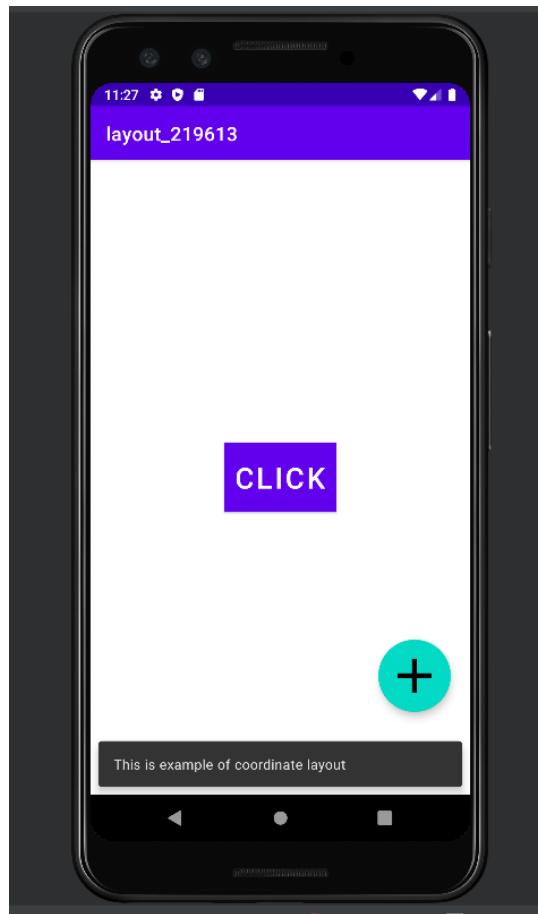
class MainActivity : AppCompatActivity() {
    override fun onCreate(savedInstanceState: Bundle?) {
        super.onCreate(savedInstanceState)
        setContentView(R.layout.activity_main)
    }
    fun show(view: View){
        var snackbar = Snackbar.make(view,"This is example of coordinate layout", Snackbar.LENGTH_SHORT)
        snackbar.show()
    }
}
```

The screenshot shows the Android Studio interface. On the left is the Project structure, displaying the app module with files like AndroidManifest.xml, MainActivity.kt, and various Java and XML files. The main area shows the code for MainActivity.kt:

```
1 package com.example.layout_219613
2 import androidx.appcompat.app.AppCompatActivity
3 import android.os.Bundle
4 import android.view.View
5 import androidx.coordinatorlayout.widget.CoordinatorLayout
6 import com.example.layout_219613.R
7 import com.google.android.material.snackbar.Snackbar
8
9 class MainActivity : AppCompatActivity() {
10     override fun onCreate(savedInstanceState: Bundle?) {
11         super.onCreate(savedInstanceState)
12         setContentView(R.layout.activity_main)
13     }
14     fun show(view: View){
15         var snackbar = Snackbar.make(view, "This is example of coordinate layout", Snackbar.LENGTH_SHORT)
16         snackbar.show()
17     }
18 }
19
```

The right side of the screen shows the emulator running a Pixel 3 API 28 device. The app's title bar says "layout_219613". Inside the app, there is a purple button with the text "CLICK" and a green circular button with a white plus sign. A black Snackbar at the bottom displays the message "This is example of coordinate layout". The bottom of the screen shows the Android navigation bar.

Output:



Linear Layout

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

    <TextView
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:text="Vertical Orientation"
        android:textAlignment="center" />

    <Button
        android:id="@+id/button8"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:text="Button" />

    <Button
        android:id="@+id/button11"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:text="Button" />

    <TextView
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:text="Horizontal Orientation"
        android:textAlignment="center"/>

    <LinearLayout
        android:layout_width="match_parent"
        android:layout_height="match_parent"
        android:orientation="horizontal"
        tools:context=".MainActivity">

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

        <Button
            android:id="@+id/button4"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:layout_weight="1"
            android:text="Button" />

        <Button
            android:id="@+id/button3"
```

```
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:layout_weight="1"
    android:text="Button" />

<Button
    android:id="@+id/button7"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:layout_weight="1"
    android:text="Button" />
</LinearLayout>

<LinearLayout
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    android:orientation="vertical"
    tools:context=".MainActivity">
    <Button
        android:id="@+id/button5"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_weight="1"
        android:text="Button" />

    <Button
        android:id="@+id/button6"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_weight="1"
        android:text="Button" />
</LinearLayout>

<Button
    android:id="@+id/button9"
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:text="Button" />
</LinearLayout>
```

```
<?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"
    xmlns:app="http://schemas.android.com/apk/res-auto"
    android:orientation="vertical"
    tools:context=".MainActivity">

    <TextView
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:text="Vertical Orientation"
        android:textAlignment="center" />

    <Button
        android:id="@+id/button8"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:text="Button" />
</LinearLayout>
```

Output:

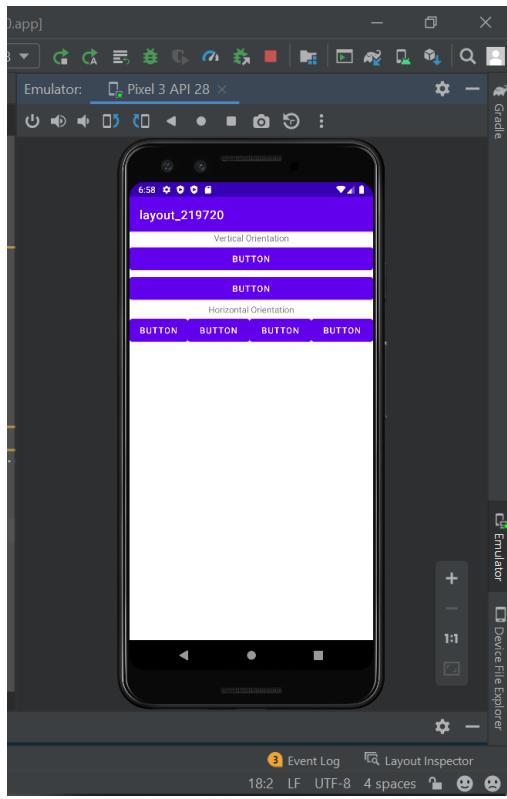


Table Layout

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"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    android:orientation="vertical"
    tools:context=".MainActivity">

    <EditText
        android:id="@+id/edt_num1"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:background="@color/white"
        android:hint="Enter the first number " />

    <EditText
        android:id="@+id/edt_num2"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:background="@color/white"
        android:hint="Enter the second number " />

    <Button
        android:id="@+id	btn_add"
        android:layout_width="70dp"
        android:layout_height="wrap_content"
        android:text="+"
        android:textSize="25dp" />

    <Button
        android:id="@+id	btn_minus"
```

```

        android:layout_width="70dp"
        android:layout_height="wrap_content"
        android:text="-"
        android:textSize="25dp" />

<Button
    android:id="@+id	btn_mul"
    android:layout_width="70dp"
    android:layout_height="wrap_content"
    android:text="x"
    android:textSize="25dp" />

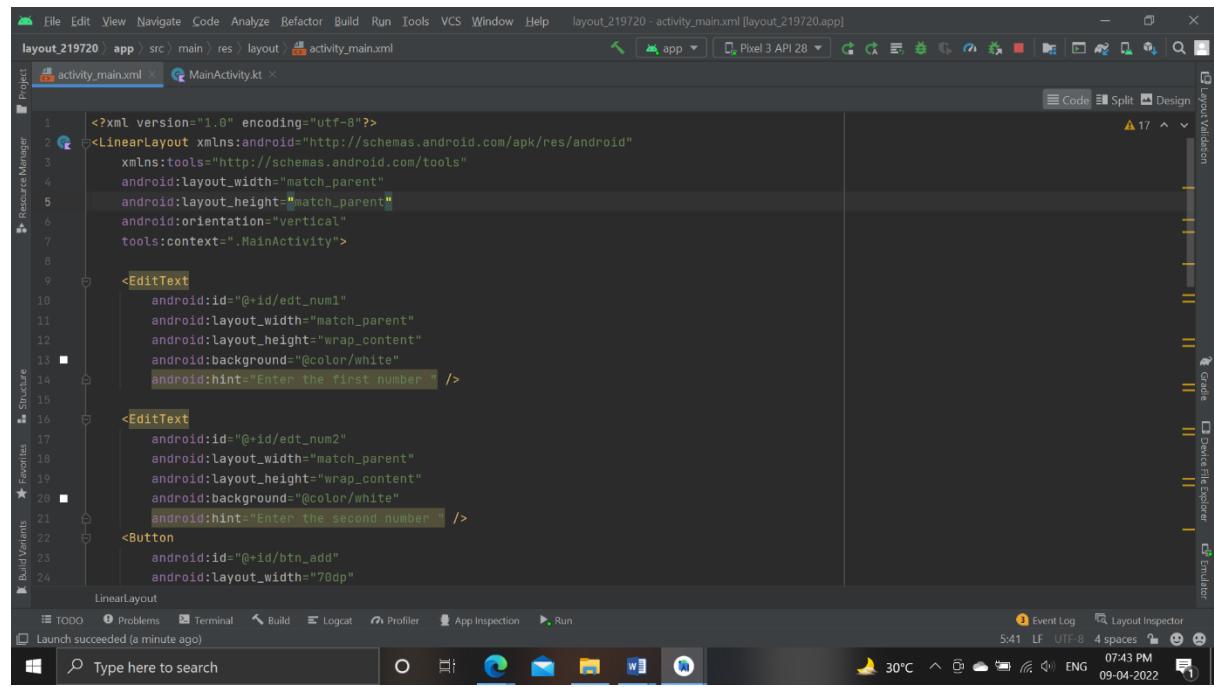
<Button
    android:id="@+id	btn_div"
    android:layout_width="70dp"
    android:layout_height="wrap_content"
    android:text="/"
    android:textSize="25dp" />

<Button
    android:id="@+id	btn_clr"
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:text="Clear"
    android:textSize="25dp" />

<TextView
    android:id="@+id/text_result"
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:background="@color/white"
    android:textSize="25dp"
    android:textStyle="bold" />

</LinearLayout>

```



mainactivity.kt

```
package com.example.layout
import androidx.appcompat.app.AppCompatActivity
import android.os.Bundle
import android.widget.Button
import android.widget.EditText
import android.widget.TextView

class MainActivity : AppCompatActivity() {
    override fun onCreate(savedInstanceState: Bundle?) {
        super.onCreate(savedInstanceState)
        setContentView(R.layout.activity_main)
        val edt_num1 = findViewById<EditText>(R.id.edt_num1)
        val edt_num2 = findViewById<EditText>(R.id.edt_num2)
        val btn_add = findViewById<Button>(R.id.btn_add)
        val btn_minus = findViewById<Button>(R.id.btn_minus)
        val btn_mul = findViewById<Button>(R.id.btn_mul)
        val btn_div = findViewById<Button>(R.id.btn_div)
        val btn_clr = findViewById<Button>(R.id.btn_clr)
        val text_result = findViewById<TextView>(R.id.text_result)

        btn_add.setOnClickListener {
            val num1 = edt_num1.text.toString().toInt()
            val num2 = edt_num2.text.toString().toInt()
            val addition = num1 + num2
            text_result.setText(addition.toString())
        }

        btn_minus.setOnClickListener {
            val num1 = edt_num1.text.toString().toInt()
            val num2 = edt_num2.text.toString().toInt()
            val subtraction = num1 - num2
            text_result.setText(subtraction.toString())
        }

        btn_mul.setOnClickListener {
            val num1 = edt_num1.text.toString().toInt()
            val num2 = edt_num2.text.toString().toInt()
            val multiply = num1 * num2
            text_result.setText(multiply.toString())
        }

        btn_div.setOnClickListener {
            val num1 = edt_num1.text.toString().toInt()
            val num2 = edt_num2.text.toString().toInt()
            val division = num1 / num2
            text_result.setText(division.toString())
        }

        btn_clr.setOnClickListener {
            edt_num1.setText("")
            edt_num2.setText("")
            text_result.setText("")
        }
    }
}
```

```
}
```

```
File Edit View Navigate Code Analyze Refactor Build Run Tools VCS Window Help layout_219720 - MainActivity.kt [layout_219720.app]
```

```
layout_219720 > app > src > main > java > com > example > layout_219720 > MainActivity
```

```
activity_main.xml > MainActivity.kt
```

```
Project Resource Manager Structure Favorites Build Variants
```

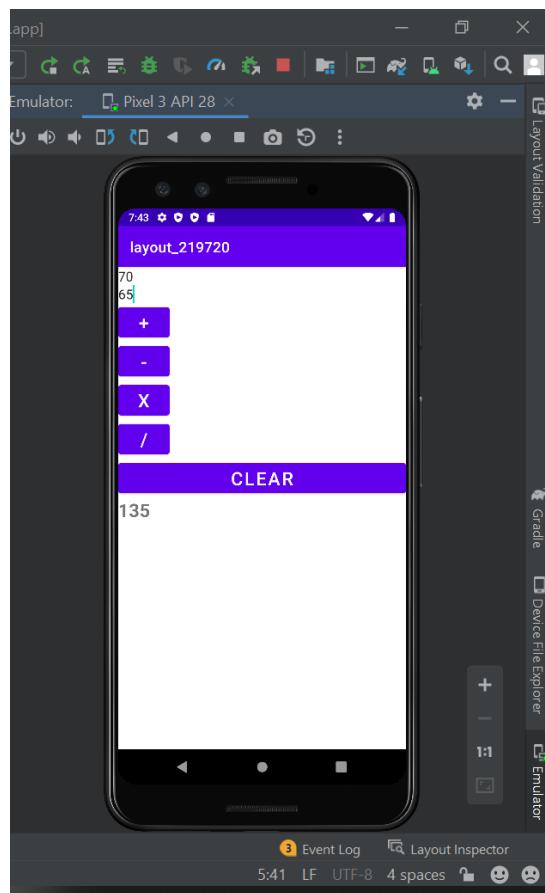
```
1 package com.example.layout_219720
2 import androidx.appcompat.app.AppCompatActivity
3 import android.os.Bundle
4 import android.widget.Button
5 import android.widget.EditText
6 import android.widget.TextView
7
8 class MainActivity : AppCompatActivity() {
9     override fun onCreate(savedInstanceState: Bundle?) {
10         super.onCreate(savedInstanceState)
11         setContentView(R.layout.activity_main)
12         val edt_num1 = findViewById<EditText>(R.id.edt_num1)
13         val edt_num2 = findViewById<EditText>(R.id.edt_num2)
14         val btn_add = findViewById<Button>(R.id.btn_add)
15         val btn_minus = findViewById<Button>(R.id.btn_minus)
16         val btn_mlt = findViewById<Button>(R.id.btn_mlt)
17         val btn_div = findViewById<Button>(R.id.btn_div)
18         val btn_clr = findViewById<Button>(R.id.btn_clr)
19         val text_result = findViewById<TextView>(R.id.text_result)
20
21         btn_add.setOnClickListener { it: View! }
22             val num1 = edt_num1.text.toString().toInt()
23             val num2 = edt_num2.text.toString().toInt()
24             val addition = num1 + num2
25             text_result.setText(addition.toString())
26     }
27 }
```

```
TODO Problems Terminal Build Logcat Profiler App Inspection Run Event Log Layout Inspector
59:2 LF UTF-8 4 spaces 30°C 7:44 PM 07:44 PM 09-04-2022
Launch succeeded (a minute ago)
```

```
Type here to search
```

```
Event Log Layout Inspector
5:41 LF UTF-8 4 spaces 30°C 7:44 PM 07:44 PM 09-04-2022
```

Output:



Grid Layout

activity_main.xml

```
<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"
    android:orientation="vertical"
    android:weightSum="10"
    android:background="@drawable/ic_launcher_foreground"
    tools:context=".MainActivity">
    <GridLayout
        android:id="@+id/grid"
        android:rowCount="3"
        android:columnCount="2"
        android:alignmentMode="alignMargins"
        android:columnOrderPreserved="false"
        android:layout_weight="8"
        android:layout_width="match_parent"
        android:layout_height="0dp"
        android:layout_marginTop="65dp"
        android:padding="14dp">
        <androidx.cardview.widget.CardView
            android:layout_width="0dp"
            android:layout_height="0dp"
            android:layout_columnWeight="1"
            android:layout_marginBottom="16dp"
            android:layout_rowWeight="1"
            android:layout_marginLeft="16dp"
```

```
        android:layout_marginRight="16dp"
        app:cardElevation="8dp"
        app:cardCornerRadius="8dp"
        android:id="@+id/fb">
    <LinearLayout
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_gravity="center_horizontal|center_vertical"
        android:layout_margin="16dp"
        android:orientation="vertical">
        <TextView
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:textColor="#0000"
            android:textSize="18sp"
            android:textStyle="bold"
            android:textAlignment="center"
            android:text="Facebook"/>
    </LinearLayout>
</androidx.cardview.widget.CardView>
<androidx.cardview.widget.CardView
    android:layout_width="0dp"
    android:layout_height="0dp"
    android:layout_columnWeight="1"
    android:layout_marginBottom="16dp"
    android:layout_rowWeight="1"
    android:layout_marginLeft="16dp"
    android:layout_marginRight="16dp"
    app:cardElevation="8dp"
    app:cardCornerRadius="8dp"
    android:id="@+id/google">
    <LinearLayout
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_gravity="center_horizontal|center_vertical"
        android:layout_margin="16dp"
        android:orientation="vertical">
        <TextView
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:text="Google"
            android:textColor="#0000"
            android:textSize="18sp"
            android:textStyle="bold"
            android:textAlignment="center"/>
    </LinearLayout>
</androidx.cardview.widget.CardView>
<androidx.cardview.widget.CardView
    android:layout_width="0dp"
    android:layout_height="0dp"
    android:layout_columnWeight="1"
    android:layout_marginBottom="16dp"
    android:layout_rowWeight="1"
    android:layout_marginLeft="16dp"
    android:layout_marginRight="16dp"
    app:cardElevation="8dp"
    app:cardCornerRadius="8dp"
    android:id="@+id/twitter">
    <LinearLayout
        android:layout_width="wrap_content"
```

```
        android:layout_height="wrap_content"
        android:layout_gravity="center_horizontal|center_vertical"
        android:layout_margin="16dp"
        android:orientation="vertical">
    <TextView
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="Twitter"
        android:textColor="#0000"
        android:textSize="18sp"
        android:textStyle="bold"
        android:textAlignment="center"/>
    </LinearLayout>
</androidx.cardview.widget.CardView>
<androidx.cardview.widget.CardView
    android:layout_width="0dp"
    android:layout_height="0dp"
    android:layout_columnWeight="1"
    android:layout_marginBottom="16dp"
    android:layout_rowWeight="1"
    android:layout_marginLeft="16dp"
    android:layout_marginRight="16dp"
    app:cardElevation="8dp"
    app:cardCornerRadius="8dp"
    android:id="@+id/linkd">
    <LinearLayout
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_gravity="center_horizontal|center_vertical"
        android:layout_margin="16dp"
        android:orientation="vertical">
        <TextView
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:text="Linkedin"
            android:textColor="#0000"
            android:textSize="18sp"
            android:textStyle="bold"
            android:textAlignment="center"/>
        </LinearLayout>
</androidx.cardview.widget.CardView>
<androidx.cardview.widget.CardView
    android:layout_width="0dp"
    android:layout_height="0dp"
    android:layout_columnWeight="1"
    android:layout_marginBottom="16dp"
    android:layout_rowWeight="1"
    android:layout_marginLeft="16dp"
    android:layout_marginRight="16dp"
    app:cardElevation="8dp"
    app:cardCornerRadius="8dp"
    android:id="@+id/insta">
    <LinearLayout
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_gravity="center_horizontal|center_vertical"
        android:layout_margin="16dp"
        android:orientation="vertical">
        <TextView
            android:layout_width="wrap_content"
```

```

        android:layout_height="wrap_content"
        android:text="Instagram"
        android:textColor="#0000"
        android:textSize="18sp"
        android:textStyle="bold"
        android:textAlignment="center"/>
    
```

```
</LinearLayout>
```

```
</androidx.cardview.widget.CardView>
```

```
<androidx.cardview.widget.CardView
```

```
    android:layout_width="0dp"
```

```
    android:layout_height="0dp"
```

```
    android:layout_columnWeight="1"
```

```
    android:layout_marginBottom="16dp"
```

```
    android:layout_rowWeight="1"
```

```
    android:layout_marginLeft="16dp"
```

```
    android:layout_marginRight="16dp"
```

```
    app:cardElevation="8dp"
```

```
    app:cardCornerRadius="8dp"
```

```
    android:id="@+id/youtube">
```

```
<LinearLayout
```

```
    android:layout_width="wrap_content"
```

```
    android:layout_height="wrap_content"
```

```
    android:layout_gravity="center_horizontal|center_vertical"
```

```
    android:layout_margin="16dp"
```

```
    android:orientation="vertical">
```

```
<TextView
```

```
    android:layout_width="wrap_content"
```

```
    android:layout_height="wrap_content"
```

```
    android:text="YouTube"
```

```
    android:textColor="#0000"
```

```
    android:textSize="18sp"
```

```
    android:textStyle="bold"
```

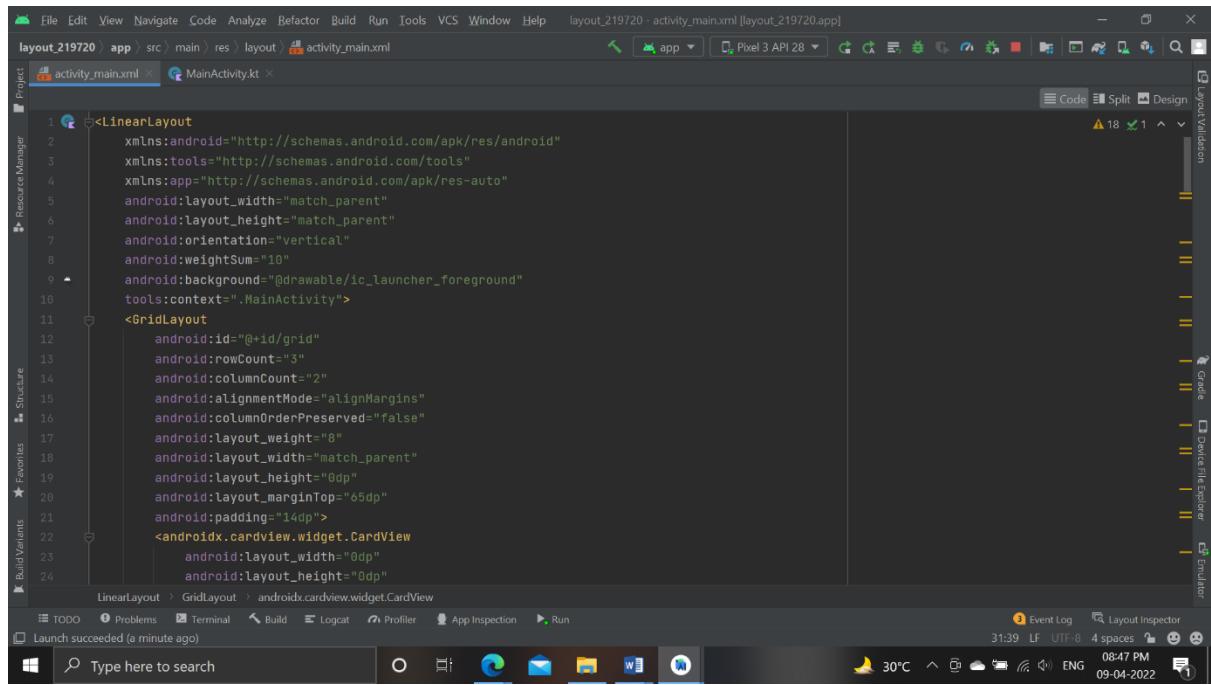
```
    android:textAlignment="center"/>
```

```
</LinearLayout>
```

```
</androidx.cardview.widget.CardView>
```

```
</GridLayout>
```

```
</LinearLayout>
```

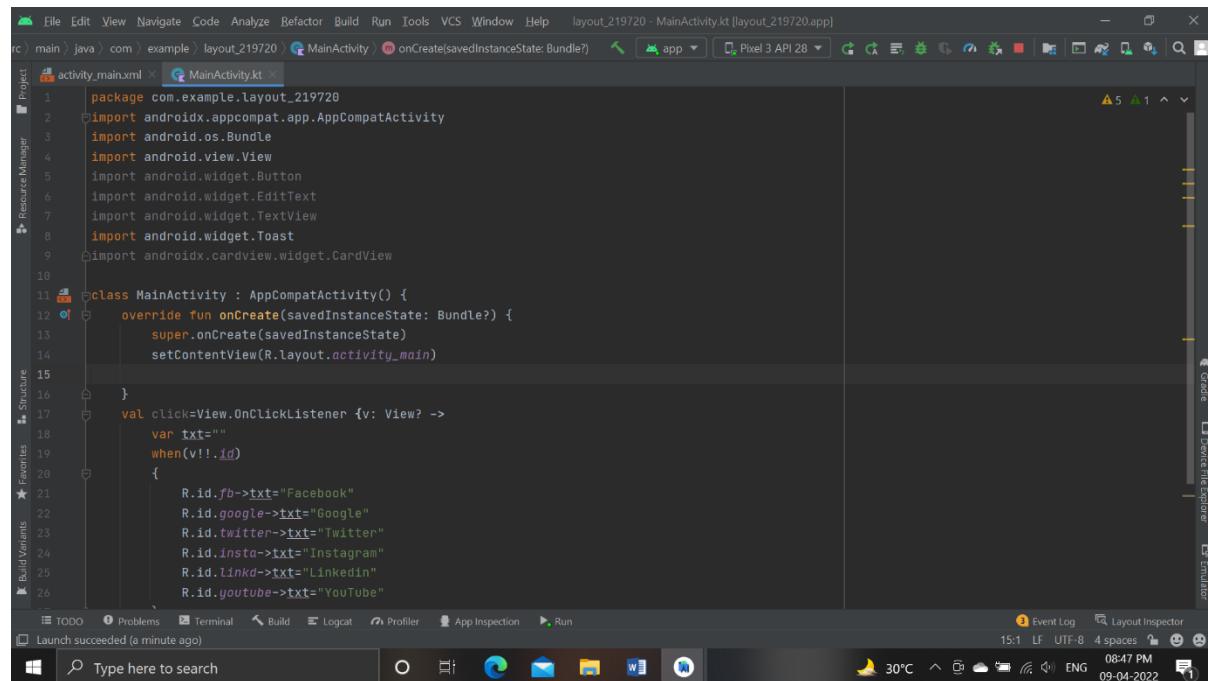


mainactivity.kt

```
package com.example.layout
import androidx.appcompat.app.AppCompatActivity
import android.os.Bundle
import android.view.View
import android.widget.Button
import android.widget.EditText
import android.widget.TextView
import android.widget.Toast
import androidx.cardview.widget.CardView

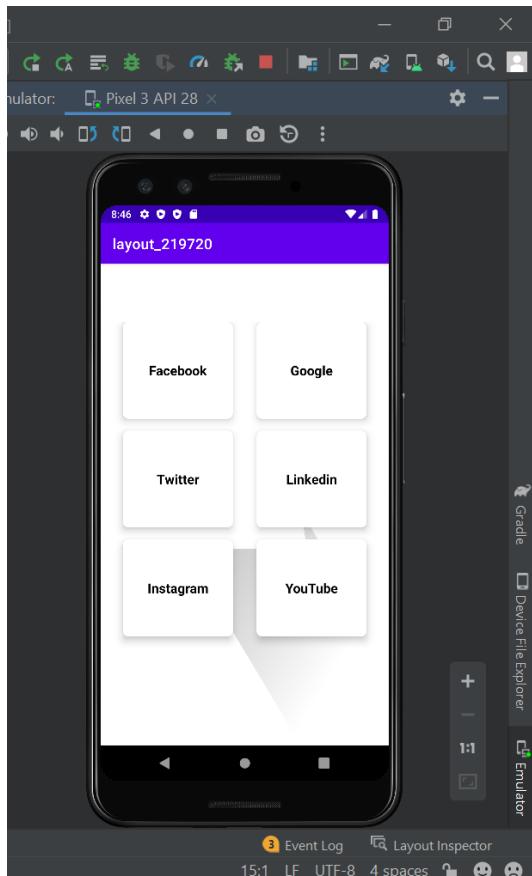
class MainActivity : AppCompatActivity() {
    override fun onCreate(savedInstanceState: Bundle?) {
        super.onCreate(savedInstanceState)
        setContentView(R.layout.activity_main)

        val click=View.OnClickListener {v: View? ->
            var txt=""
            when(v!!.id)
            {
                R.id.fb->txt="Facebook"
                R.id.google->txt="Google"
                R.id.twitter->txt="Twitter"
                R.id.insta->txt="Instagram"
                R.id.linkd->txt="Linkedin"
                R.id.youtube->txt="YouTube"
            }
            Toast.makeText(this,"You clicked "+txt, Toast.LENGTH_SHORT).show()
        }
    }
}
```



Output:

STUD--Talks: Follow us on for more videos and updates



PRACTICAL 5

Programming UI elements

Design App With UI:

MainActivity.kt:

```
package com.example.a219720_p5

import android.content.Intent
import android.os.Bundle
import androidx.appcompat.app.AppCompatActivity
import com.example.a219720_p5.R.id.login
import com.example.a219720_p5.R.id.newaccount

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

        // start your next activity
        startActivity(intent)
    }
}
```

```

newaccount

    // start your next activity
    startActivity(intent)

}

class RegisterActivity {

}

class LoginActivity {
}

```

The screenshot shows the Android Studio interface with the project navigation bar at the top. Below it is the Project tool window showing the app module structure. The main editor window displays the MainActivity.kt file. The code includes a newaccount function and imports for Intent, Bundle, AppCompatActivity, and R.id.newaccount. The bottom of the screen shows the taskbar with various icons and system status.

Activity_main.xml

```

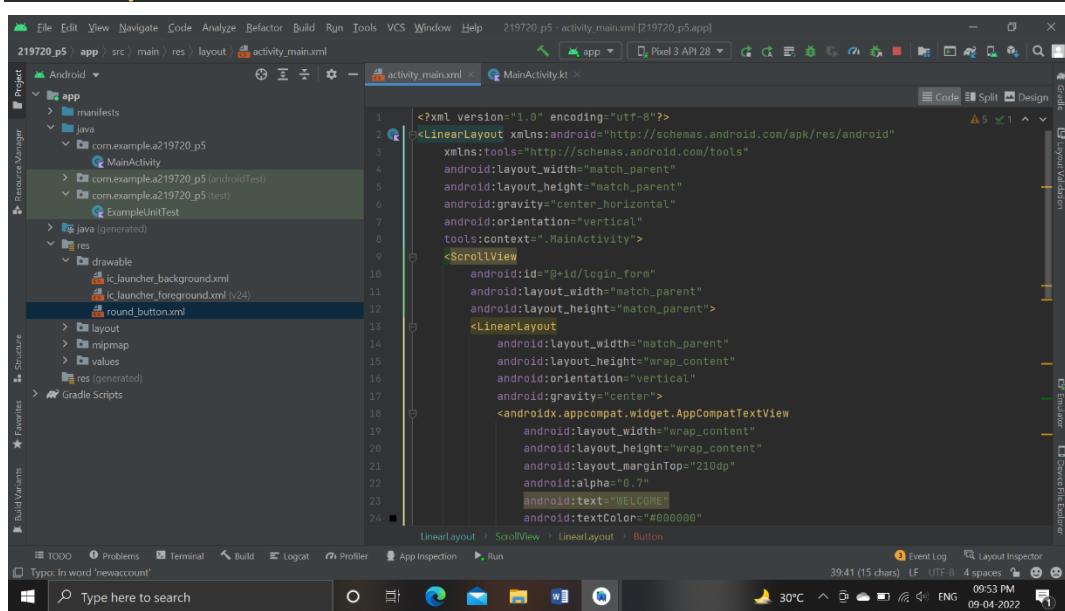
<?xml version="1.0" encoding="utf-8"?>
<LinearLayout xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:tools="http://schemas.android.com/tools"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    android:gravity="center_horizontal"
    android:orientation="vertical"
    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">
            <androidx.appcompat.widget.AppCompatTextView

```

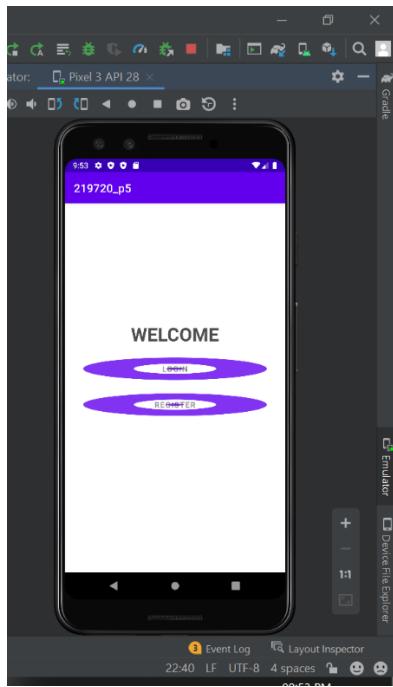
```

        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_marginTop="210dp"
        android:alpha="0.7"
        android:text="WELCOME"
        android:textColor="#000000"
        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:background="@drawable/round_button"
    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:background="@drawable/round_button"
    android:alpha="0.8"
    android:textStyle="bold" />
</LinearLayout>
</ScrollView>
</LinearLayout>

```



Output:



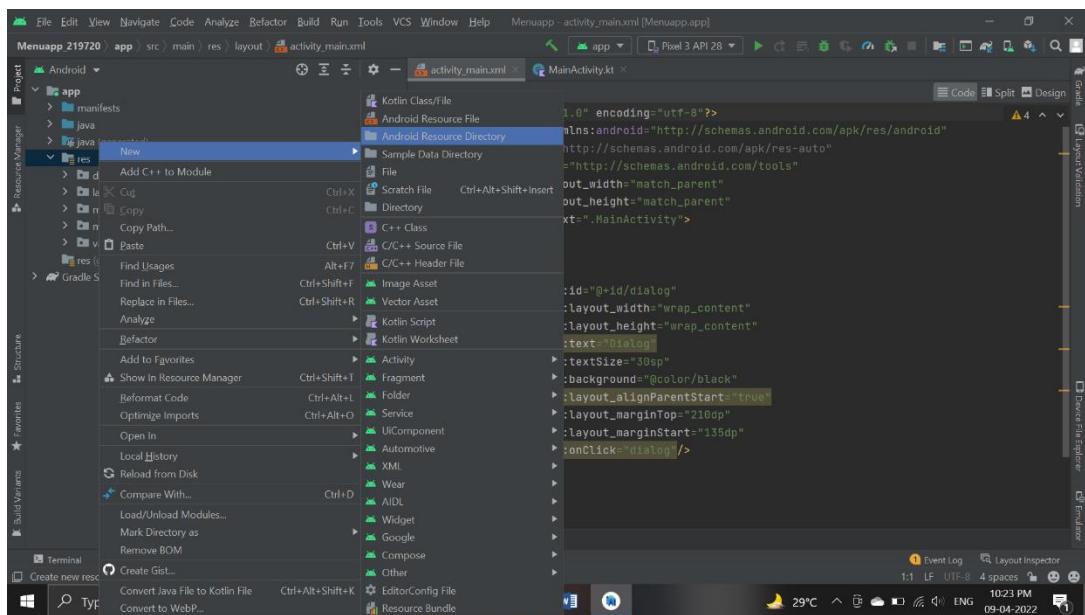
PRACTICAL 6

Programming menus, dialog, dialog fragments

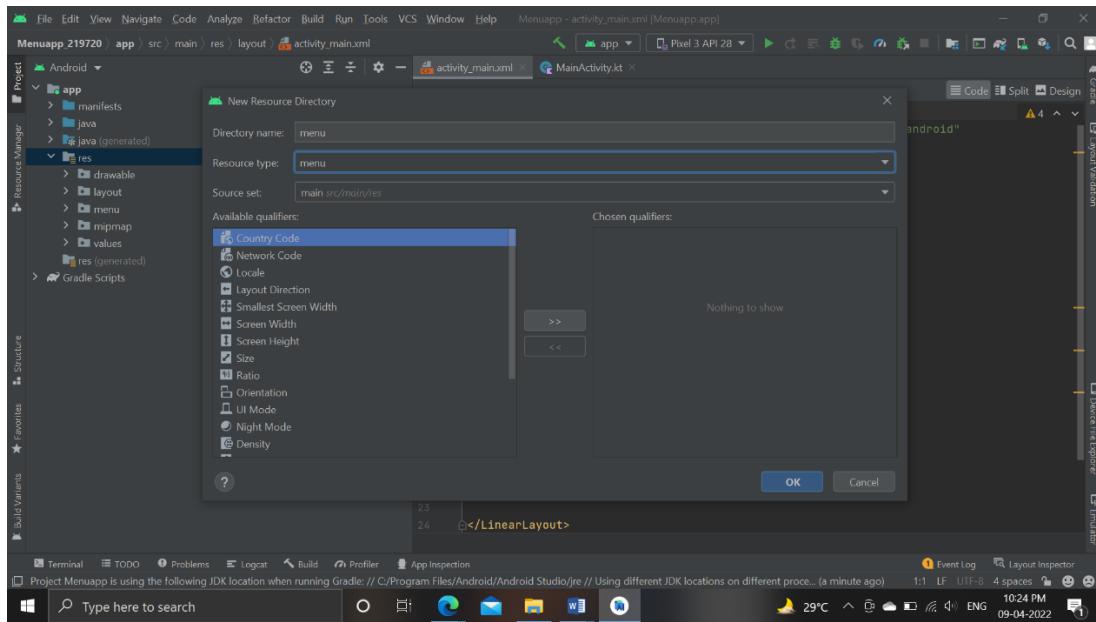
Menu

Example :

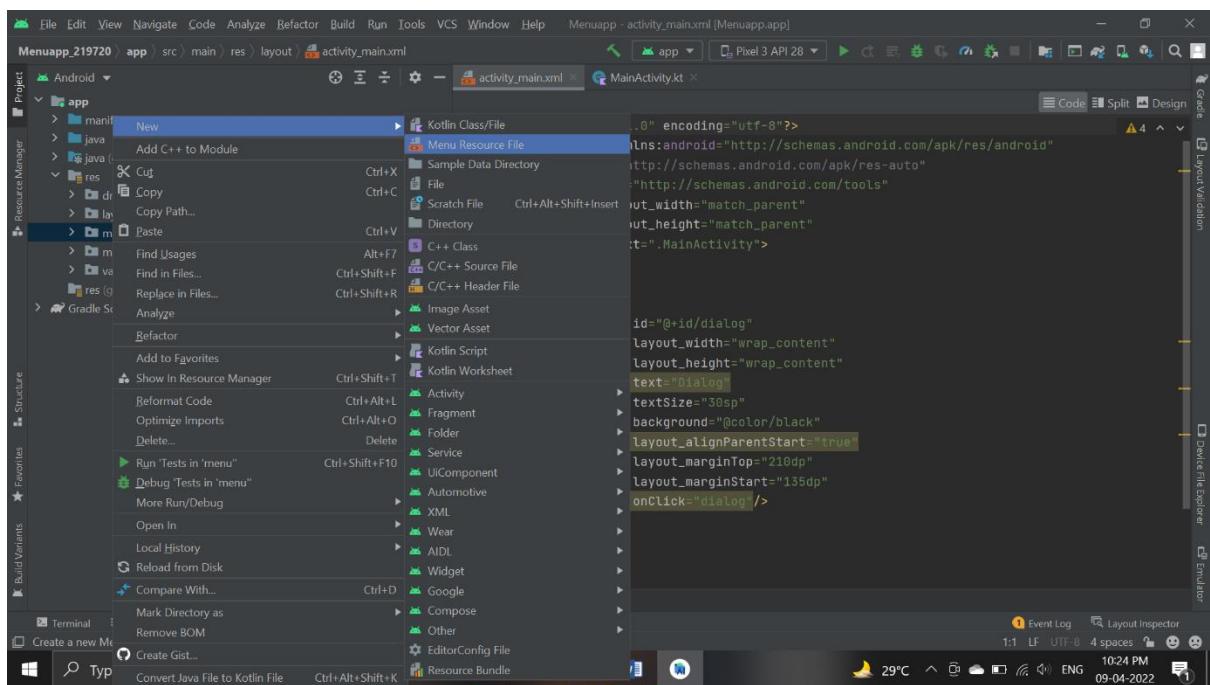
To define the menu, create an XML file inside your project's res/menu/ directory Right click on res directory then click new > Android Resource Directory



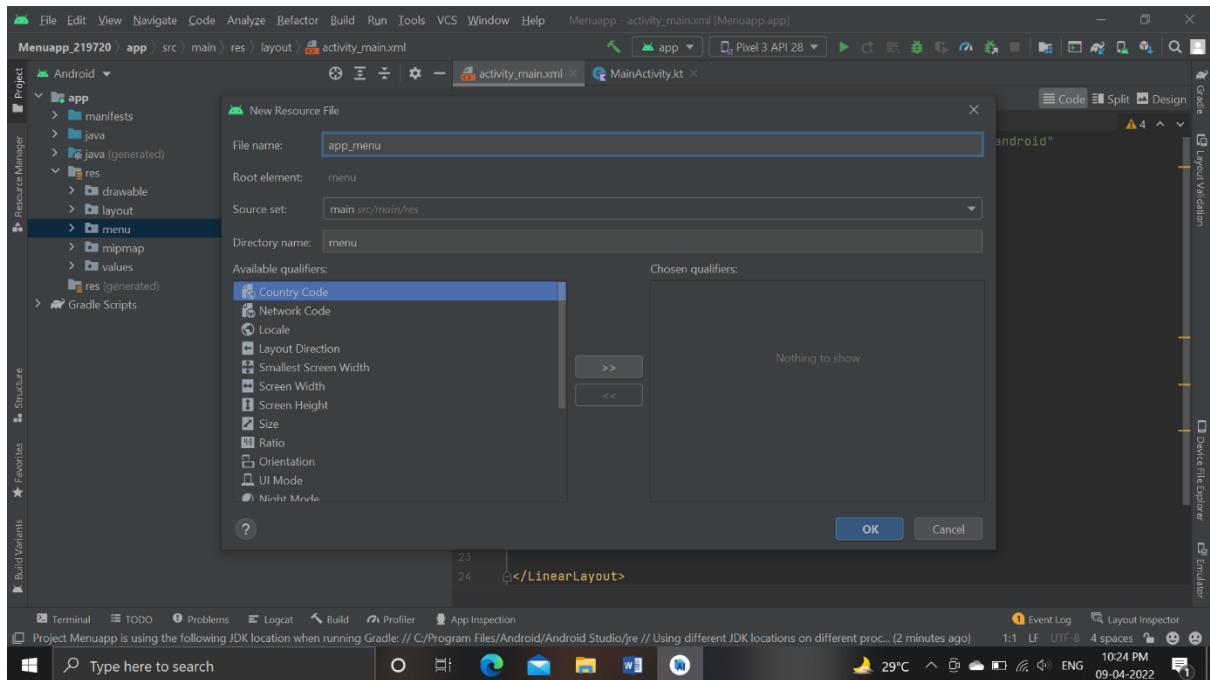
Change the Resource type from value to menu



Then Right click on menu directory then new>Menu resource file



Give the file name

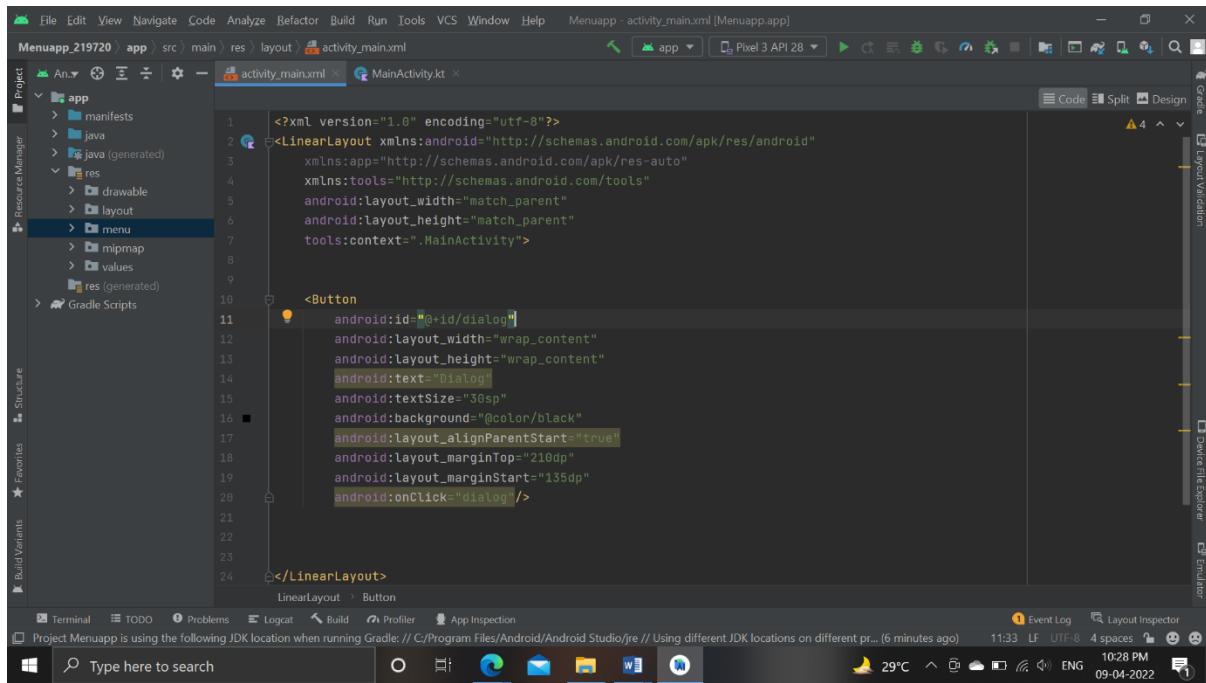


Activity_main.xml

```
<?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=".MainActivity">

    <Button
        android:id="@+id/dialog"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="Dialog"
        android:textSize="30sp"
        android:background="@color/black"
        android:layout_alignParentStart="true"
        android:layout_marginTop="210dp"
        android:layout_marginStart="135dp"
        android:onClick="dialog"/>

</LinearLayout>
```



MainActivity.kt:

```

package com.example.menuapp

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

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

    override fun onBackPressed() {
        val builder= AlertDialog.Builder(this)
        builder.setTitle("Are you sure")
        builder.setMessage("Are you sure you want to close the app?")
        builder.setPositiveButton("Yes", {dialogInterface, i -> finish()})
        builder.setNegativeButton("No", {dialogInterface, i -> })
        builder.show()
    }

    fun dialog(view: View){
        var alert= AlertDialog.Builder(this@MainActivity)
        alert.setTitle("Dialog Box")
        alert.setMessage("Please click ok or cancel")
        alert.setPositiveButton("OK"){
            dialog, which ->
            Toast.makeText(applicationContext,"YOU CLICKED OK", Toast.LENGTH_SHORT).show()
        }
    }
}

```

```

        alert.setNegativeButton("CANCEL"){
            dialog, which ->
            Toast.makeText(applicationContext,"YOU CLICKED OK", Toast.LENGTH_SHORT).show()
        }
        alert.show()
    }

override fun onCreateOptionsMenu(menu: Menu?): Boolean {
    var inflater=menuInflater
    inflater.inflate(R.menu.app_menu,menu)
    return true
}

override fun onPrepareOptionsMenu(menu: Menu?): Boolean {
    //to disable the 3rd menu item (contact)
    menu?.getItem(2)?.setEnabled(false)

    //to remove the 4th menu item (about)
    menu?.getItem(3)?.isVisible=false
    return super.onPrepareOptionsMenu(menu)
}

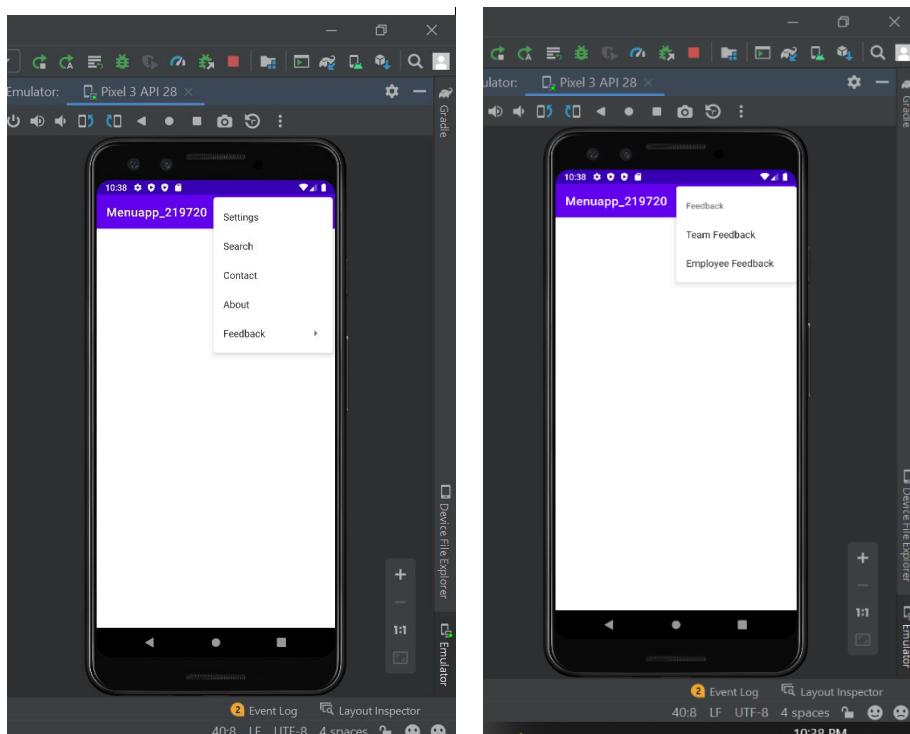
override fun onOptionsItemSelected(item: MenuItem): Boolean {
    when (item!!.itemId) {
        R.id.settings ->{
            Toast.makeText(this,"You clicked on settings", Toast.LENGTH_SHORT).show()
        }
        R.id.search ->{
            Toast.makeText(this,"You clicked on search", Toast.LENGTH_SHORT).show()
        }
        R.id.contact ->{
            Toast.makeText(this,"You clicked on contact", Toast.LENGTH_SHORT).show()
        }
        R.id.about ->{
            Toast.makeText(this,"You clicked on about", Toast.LENGTH_SHORT).show()
        }
        R.id.feedback ->{
            Toast.makeText(this,"You clicked on feedback", Toast.LENGTH_SHORT).show()
        }
        R.id.team ->{
            Toast.makeText(this,"You clicked on team feedback", Toast.LENGTH_SHORT).show()
        }
        R.id.employee ->{
            Toast.makeText(this,"You clicked on employee feedback", Toast.LENGTH_SHORT).show()
        }
    }
    return super.onOptionsItemSelected(item)
}

```

```
219720 app src main java com example menuapp_219720 MainActivity onBackPressed() Pixel 3 API 28

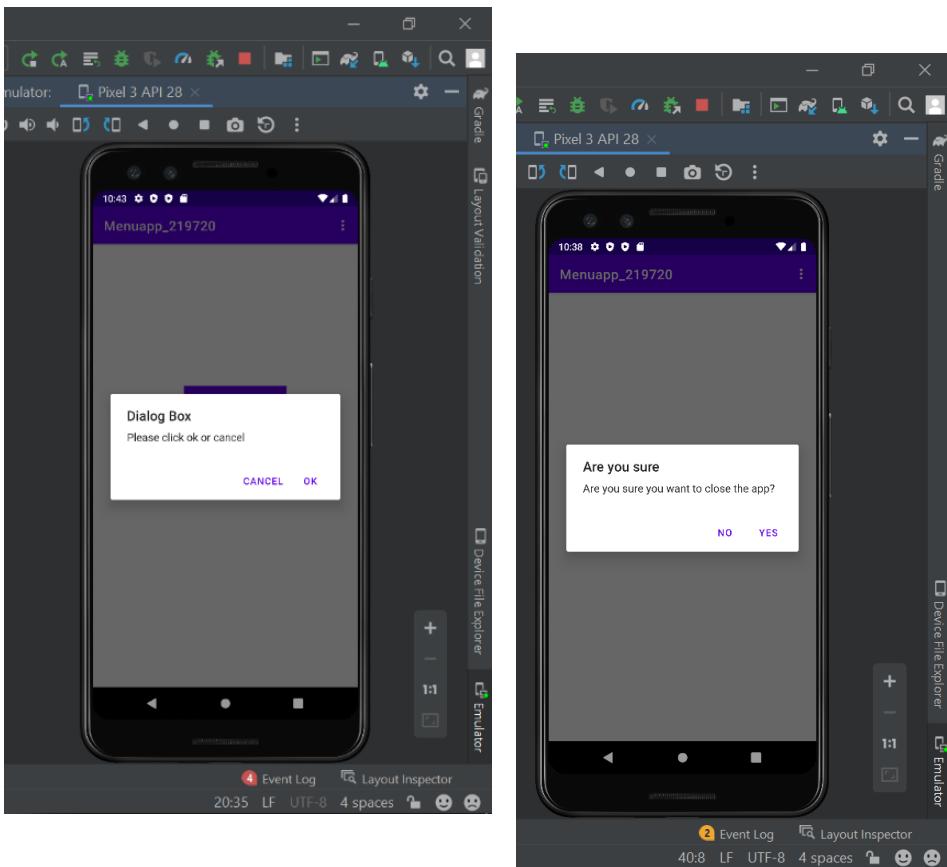
1 package com.example.menuapp_219720
2
3 import ...
4
5 class MainActivity : AppCompatActivity() {
6     override fun onCreate(savedInstanceState: Bundle?) {
7         super.onCreate(savedInstanceState)
8         setContentView(R.layout.activity_main)
9     }
10
11     override fun onBackPressed() {
12         val builder= AlertDialog.Builder( context: this)
13         builder.setTitle("Are you sure")
14         builder.setMessage("Are you sure you want to close the app?")
15         builder.setPositiveButton( text: "Yes", {dialogInterface, i -> finish()})
16         builder.setNegativeButton( text: "No", {dialogInterface, i -> })
17         builder.show()
18     }
19
20     fun dialog(view: View){
21         var alert= AlertDialog.Builder( context: this@MainActivity)
22         alert.setTitle("Dialog Box")
23         alert.setMessage("Please click OK or cancel")
24         alert.setPositiveButton( text: "OK"){ dialog, which ->
25             dialog.dismiss()
26         }
27     }
28
29
30
31
32 }
```

Output:



Dialog

Alertdialog on Back Button Pressed event



Practical No: 7

Programs on Intents,Event Listeners and Adapters

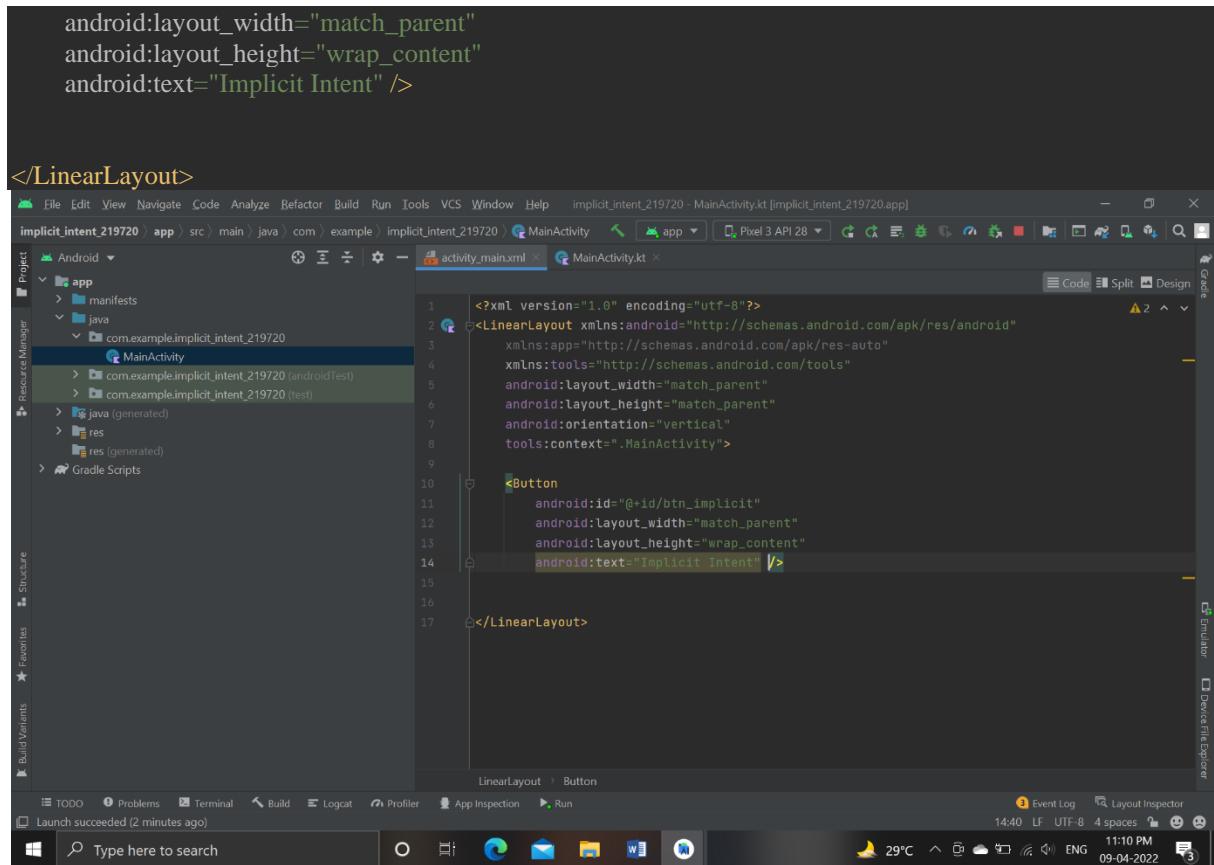
An intent is an abstract description of an operation to be performed. It is a way of ‘Message passing between two or more than two components of android’.

Example : Demonstration of explicit and implicit intents

activity_main.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="vertical"
    tools:context=".MainActivity">

    <Button
        android:id="@+id/btn_implicit"
```



Main_activity.kt

```

package com.example.implicit_intent_219720

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 btn = findViewById<Button>(R.id.btn_implicit)

        btn.setOnClickListener {
            val intent = Intent(Intent.ACTION_SEND)
            intent.putExtra(Intent.EXTRA_TEXT,"Hello, TyIT")
            intent.type="text/plain"
            startActivity(intent)
        }
    }
}

```

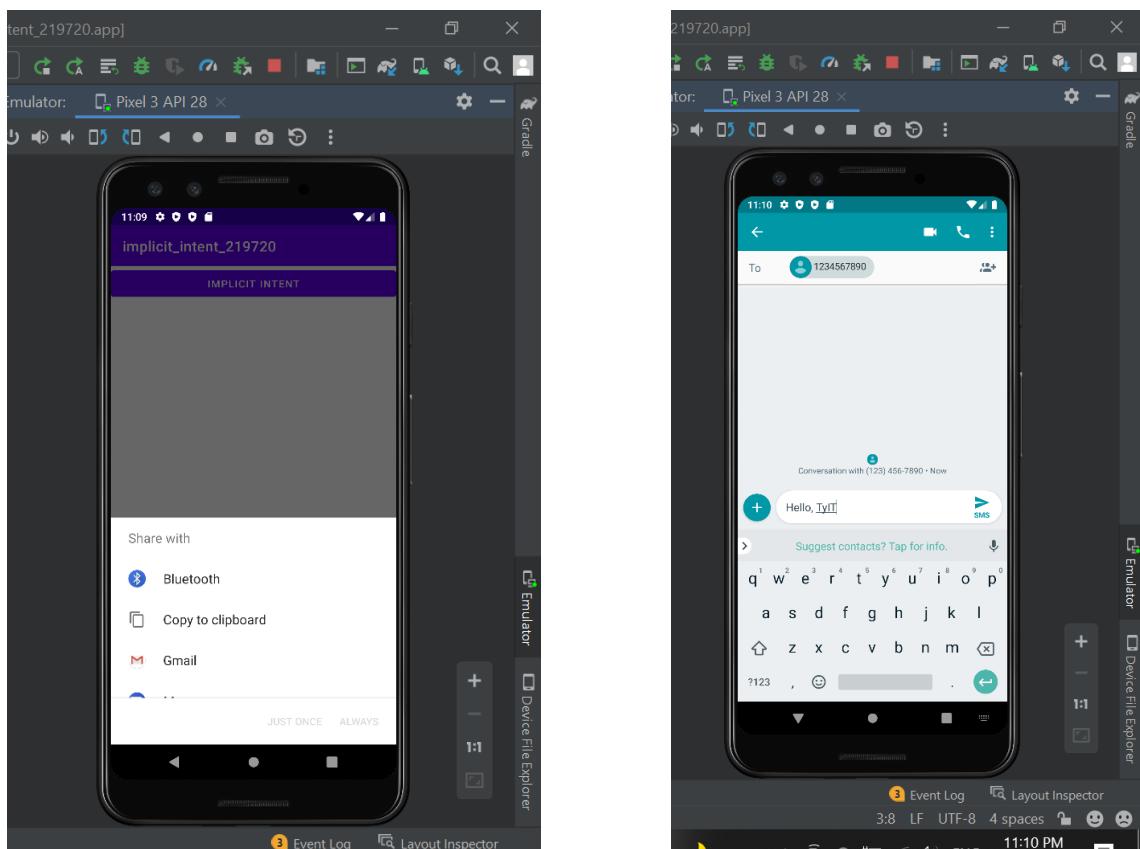
```
package com.example.implicit_intent_219720

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 btn = findViewById<Button>(R.id.btn_implicit)

        btn.setOnClickListener { view ->
            val intent = Intent(Intent.ACTION_SEND)
            intent.putExtra(Intent.EXTRA_TEXT, "Hello, TyIT")
            intent.type = "text/plain"
            startActivity(intent)
        }
    }
}
```

Output:



Practical No:8

Programs on Services,Notification and Broadcast Receivers

Services

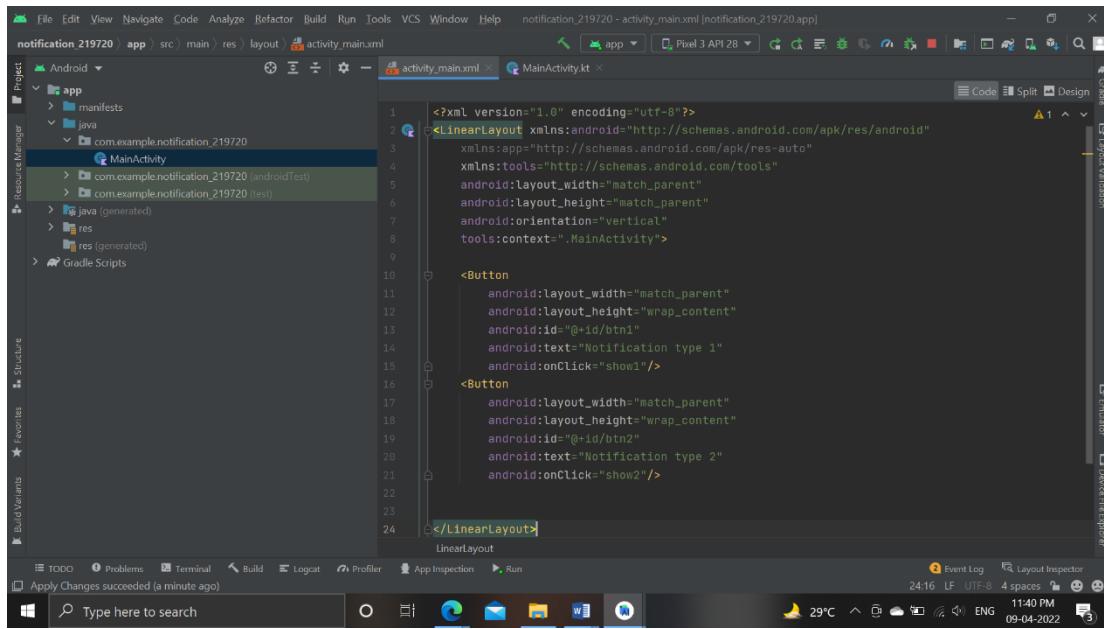
A Service is an application component that can perform long-running operations in the background, and it doesn't provide a user interface.

activity_main.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="vertical"
    tools:context=".MainActivity">

    <Button
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:id="@+id/btn1"
        android:text="Notification type 1"
        android:onClick="show1"/>
    <Button
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:id="@+id/btn2"
        android:text="Notification type 2"
        android:onClick="show2"/>

</LinearLayout>
```



Main_activity.kt

```
package com.example.notification_219720

import android.app.NotificationChannel
import android.app.NotificationManager
import android.app.PendingIntent
import android.content.Context
import android.content.Intent
import android.os.Build
import android.os.Bundle
import android.view.View
import androidx.appcompat.app.AppCompatActivity
import androidx.core.app.NotificationCompat
import androidx.core.app.NotificationManagerCompat

class MainActivity : AppCompatActivity() {
    val CHANNEL_ID = "123"
    val NOTIFICATION_ID_1 = 111
    val NOTIFICATION_ID_2 = 222
    override fun onCreate(savedInstanceState: Bundle?) {
        super.onCreate(savedInstanceState)
        setContentView(R.layout.activity_main)
        createchannel()
    }
    fun createchannel() {
        if (Build.VERSION.SDK_INT >= Build.VERSION_CODES.O){
            val channel =
                NotificationChannel(CHANNEL_ID,"notify",NotificationManager.IMPORTANCE_DEFAULT)
            notificationManager = NotificationManager(
                getSystemService(Context.NOTIFICATION_SERVICE) as NotificationManager
            )
            notificationManager.createNotificationChannel(channel)
        }
    }
    fun show1(view: View){
        var notify =
            NotificationCompat.Builder(this,CHANNEL_ID).setSmallIcon(androidx.core.R.drawable.notification_icon_bac
        kground).setContentTitle("This is the First Notification").setContentText("This is non Clickable")
    }
}
```

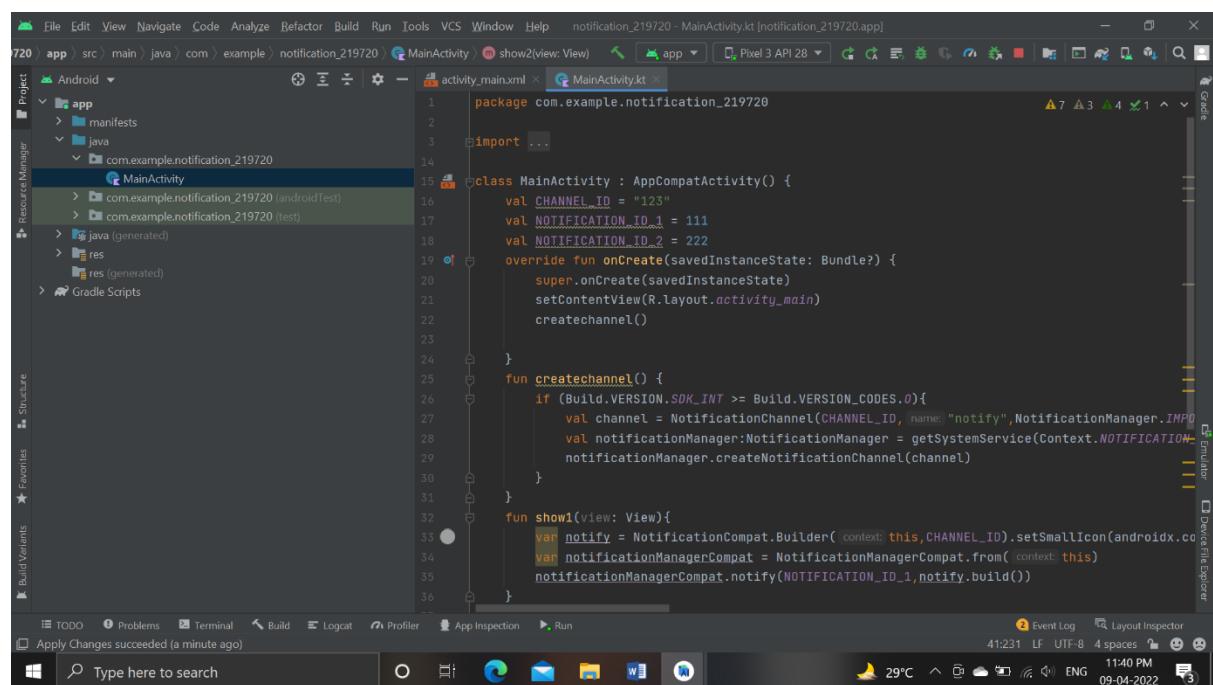
```

Notification").setPriority(NotificationCompat.PRIORITY_DEFAULT)
    var notificationManagerCompat = NotificationManagerCompat.from(this)
    notificationManagerCompat.notify(NOTIFICATION_ID_1,notify.build())
}

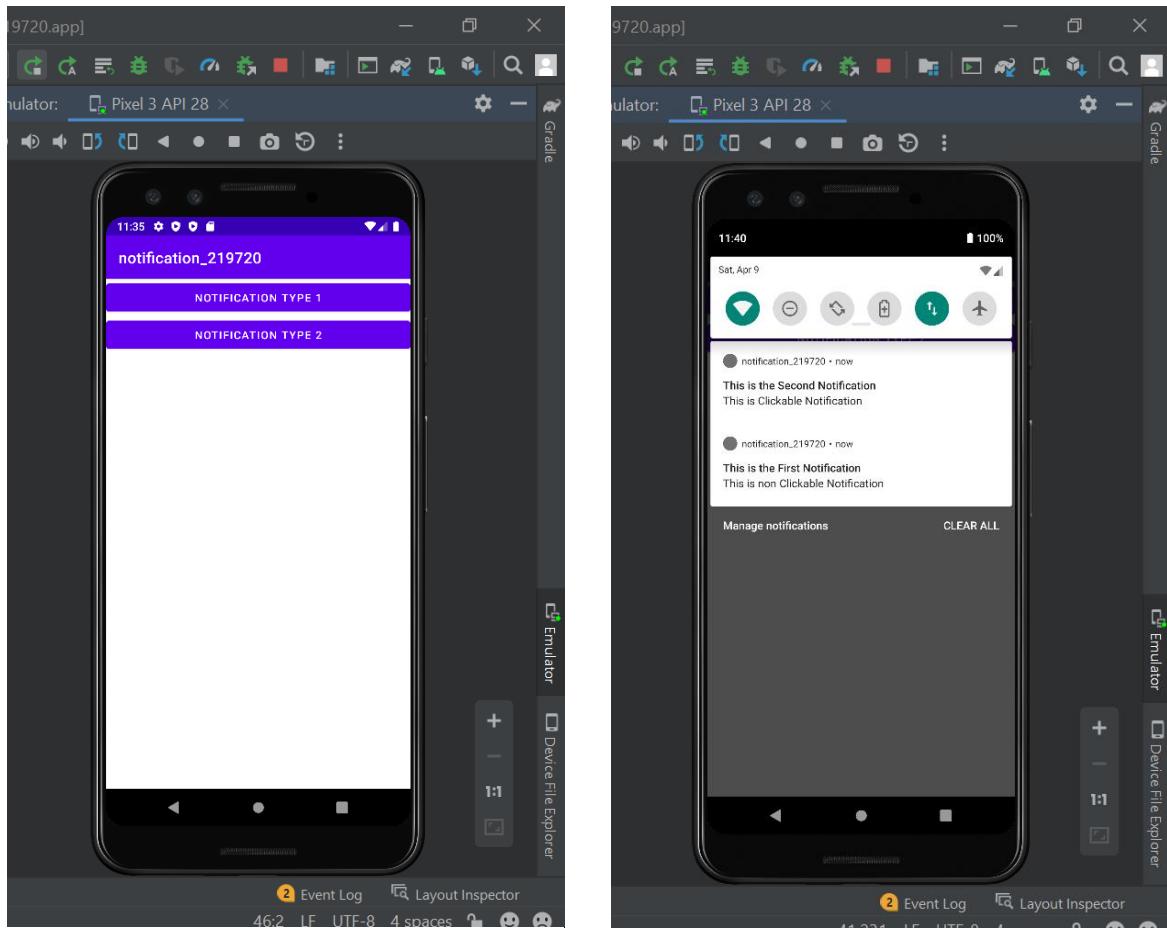
fun show2(view: View){
    val intent = Intent(this,MainActivity::class.java)
    val pendingIntent = PendingIntent.getActivity(this,0,intent,0)
    var notify =
NotificationCompat.Builder(this,CHANNEL_ID).setSmallIcon(androidx.core.R.drawable.notification_icon_bac
kground).setContentTitle("This is the Second Notification").setContentText("This is Clickable
Notification").setPriority(NotificationCompat.PRIORITY_DEFAULT).setContentIntent(pendingIntent).setAuto
Cancel(true)
    var notificationManagerCompat = NotificationManagerCompat.from(this)
    notificationManagerCompat.notify(NOTIFICATION_ID_2,notify.build())

}
}

```



Output:



Add the service in androidmanifest.xml file in the application tag

```
<service android:name=".Myservice" android:enabled="true"/>
```

MainActivity.kt

The screenshot shows the Android Studio interface with the project navigation bar at the top. Below it is the main workspace with the following details:

- Project Tree:** Shows the project structure under "notification_219720 > app". It includes "Android", "app", "manifests", "AndroidManifest.xml", "java", "MainActivity.kt", and "res".
- Main Activity Code (MainActivity.kt):**

```
1 package com.example.notification_219720
2
3 import android.content.Intent
4 import androidx.appcompat.app.AppCompatActivity
5 import android.os.Bundle
6 import android.widget.Button
7
8 class MainActivity : AppCompatActivity() {
9     override fun onCreate(savedInstanceState: Bundle?) {
10         super.onCreate(savedInstanceState)
11         setContentView(R.layout.activity_main)
12         val start = findViewById<Button>(R.id.btn_start)
13         val stop = findViewById<Button>(R.id.btn_stop)
14         start.setOnClickListener { it: View!
15             startService(Intent(packageContext: this, NewService::class.java))
16         }
17         stop.setOnClickListener { it: View!
18             stopService(Intent(packageContext: this, NewService::class.java))
19         }
20     }
21 }
22
23 }
```

- Bottom Bar:** Includes tabs for "TODO", "Problems", "Terminal", "Build", "Logcat", "Profiler", "App Inspection", and "Run". It also shows the status "Launch succeeded (a minute ago)".
- System Tray:** Shows the date (10-04-2022), time (12:12 AM), and weather (29°C).

activity_main.xml

The screenshot shows the Android Studio interface with the project navigation bar at the top. Below it is the main workspace with the following details:

- Project Tree:** Shows the project structure under "notification_219720 > app". It includes "Android", "app", "manifests", "AndroidManifest.xml", "java", "MainActivity.kt", and "res".
- Layout File (activity_main.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="vertical"
    tools:context=".MainActivity">

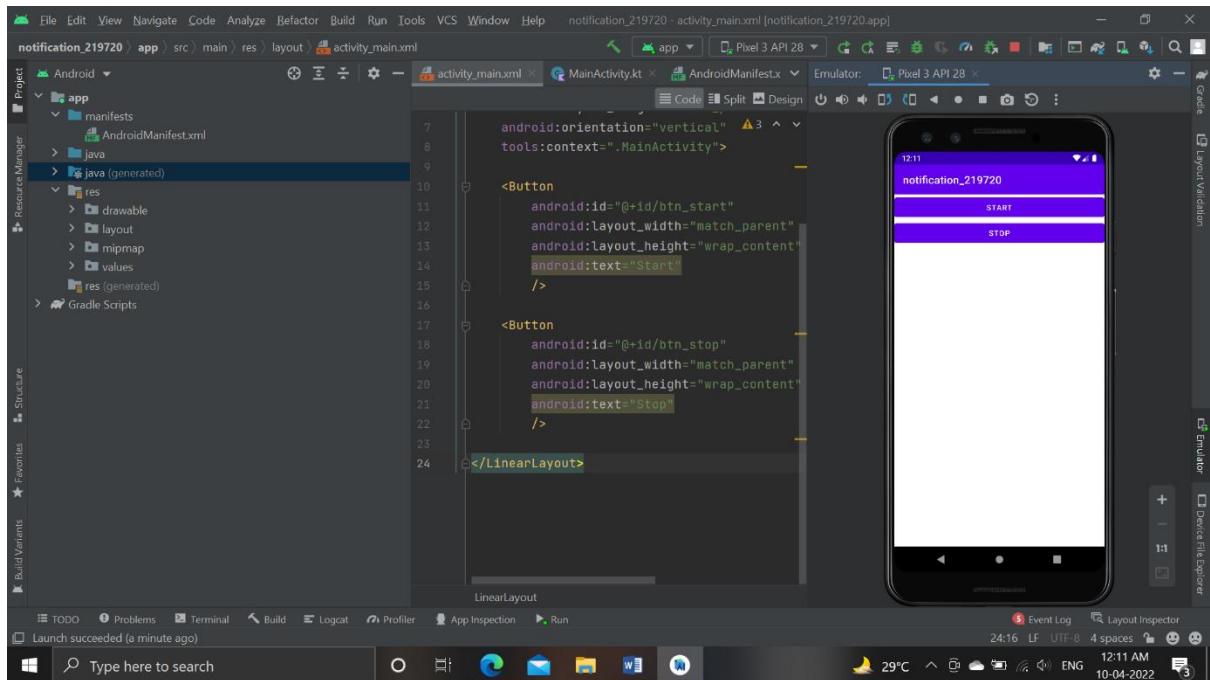
    <Button
        android:id="@+id	btn_start"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:text="Start"
    />

    <Button
        android:id="@+id	btn_stop"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:text="Stop"
    />

</LinearLayout>
```

- Bottom Bar:** Includes tabs for "TODO", "Problems", "Terminal", "Build", "Logcat", "Profiler", "App Inspection", and "Run". It also shows the status "Launch succeeded (a minute ago)".
- System Tray:** Shows the date (10-04-2022), time (12:11 AM), and weather (29°C).

Output:



PRACTICAL 9

Database Programming with SQLite

STUD--Talks: Follow us on [YouTube](#) [Instagram](#) [Facebook](#) [Telegram](#) for more videos and updates

Step1: Layout Design

Activity_main.xml:

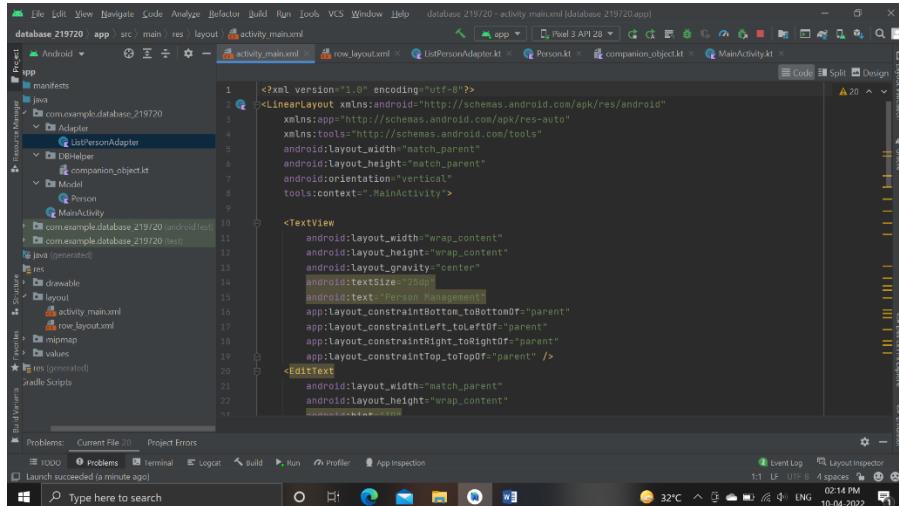
```
<?xml version="1.0" encoding="utf-8"?>
<LinearLayout xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:app="http://schemas.android.com/apk/res-auto"
    xmlns:tools="http://schemas.android.com/tools"
    android:layout_width="match_parent"
    android:layout_height="match_parent"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

    <TextView
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_gravity="center"
        android:textSize="25dp"
        android:text="Person Management"
        app:layout_constraintBottom_toBottomOf="parent"
        app:layout_constraintLeft_toLeftOf="parent"
        app:layout_constraintRight_toRightOf="parent"
        app:layout_constraintTop_toTopOf="parent" />
    <EditText
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:hint="ID"
        android:inputType="number"
        android:id="@+id/edt_id"/>
    <EditText
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:hint="Name"
        android:inputType="text"
        android:id="@+id/edt_name"/>
    <EditText
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:hint="Email"
        android:inputType="textEmailAddress"
        android:id="@+id/edt_email"/>
    <LinearLayout
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:orientation="horizontal"
        android:layout_gravity="center">
        <Button
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:text="ADD"
            android:layout_weight="1"
            android:layout_marginLeft="20dp"
            android:onClick="add"
            android:id="@+id/btn_add"/>
        <Button
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:layout_marginLeft="20dp"
            android:text="UPDATE"
            android:onClick="update"
```

```

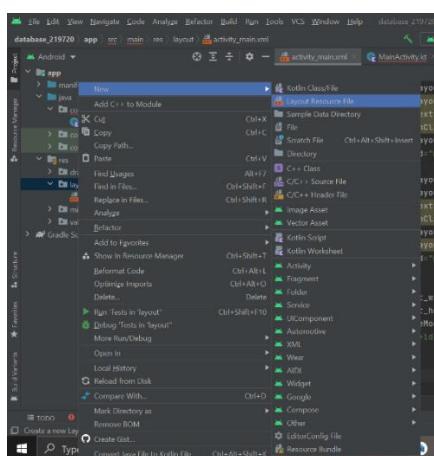
        android:layout_weight="1"
        android:id="@+id	btn_update"/>
    <Button
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="DELETE"
        android:onClick="delete"
        android:layout_weight="1"
        android:layout_marginLeft="20dp"
        android:id="@+id	btn_delete"/>
</LinearLayout>
<ListView
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:choiceMode="singleChoice"
    android:id="@+id/list_persons">
</ListView>
</LinearLayout>

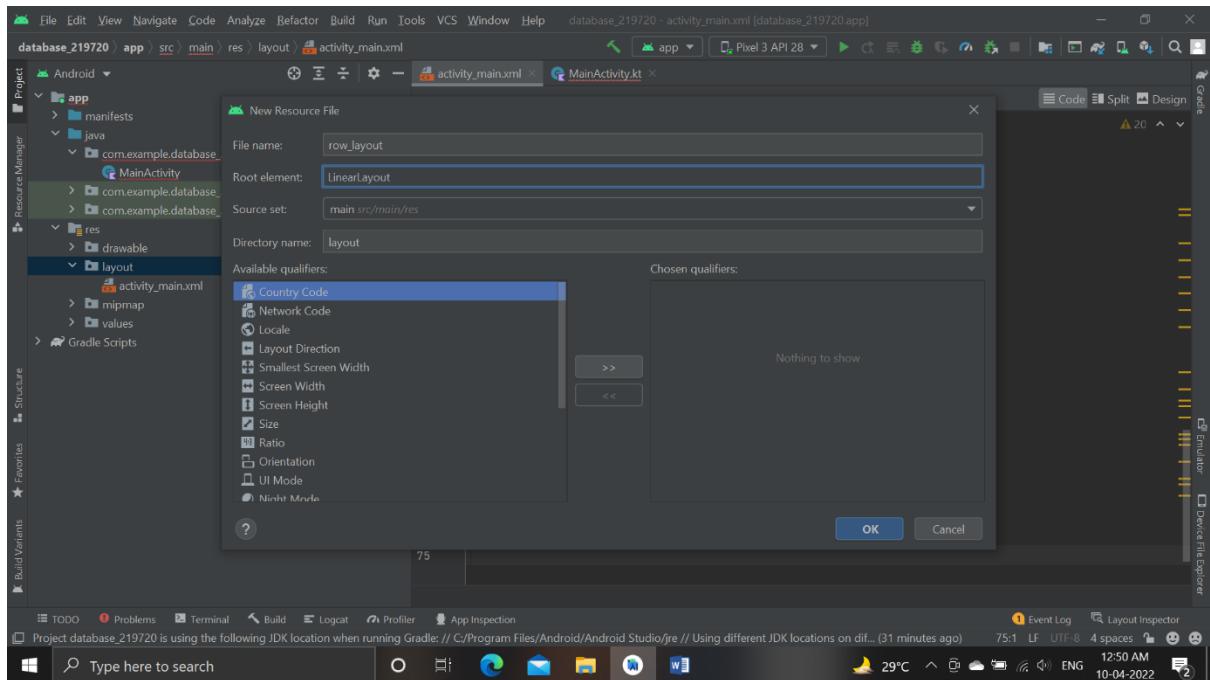
```



Step2: Create row layout to display information

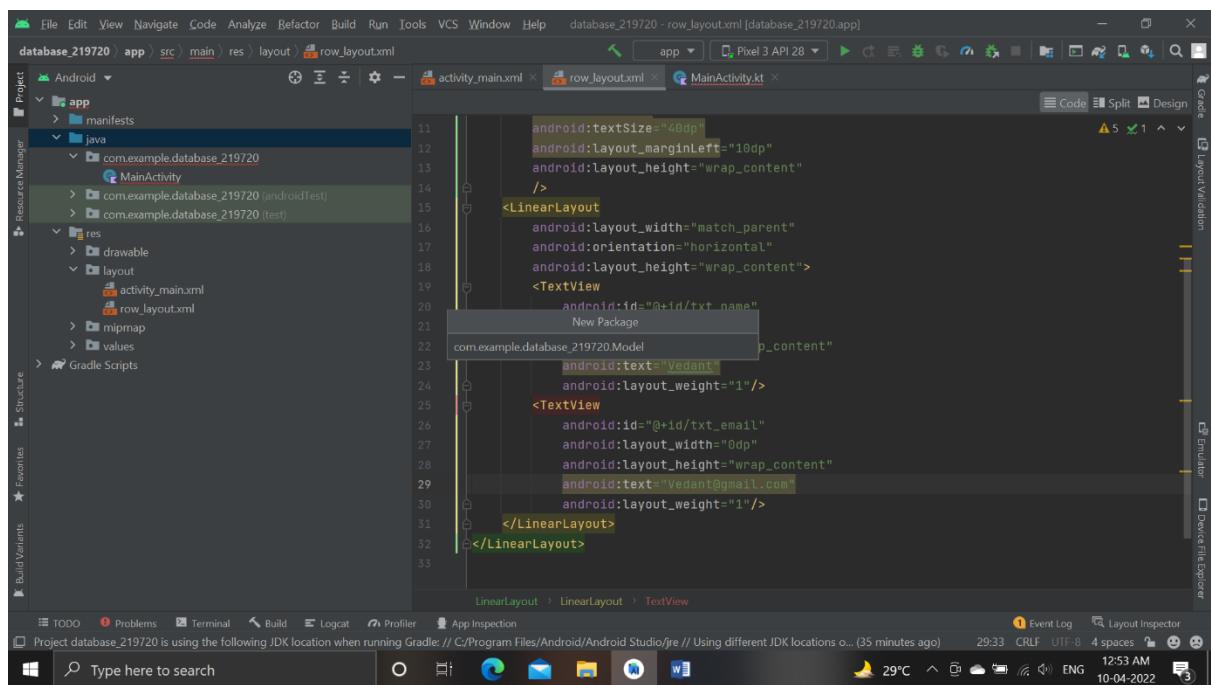
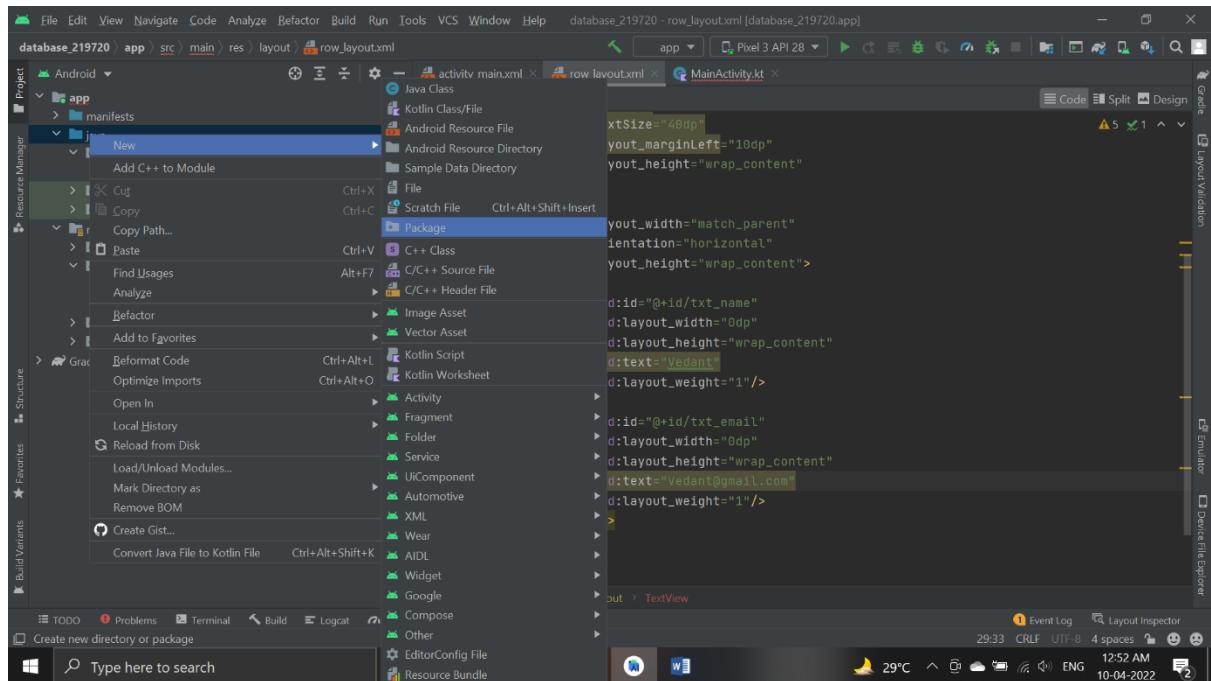
Right click on layout > New > Layout Resource File

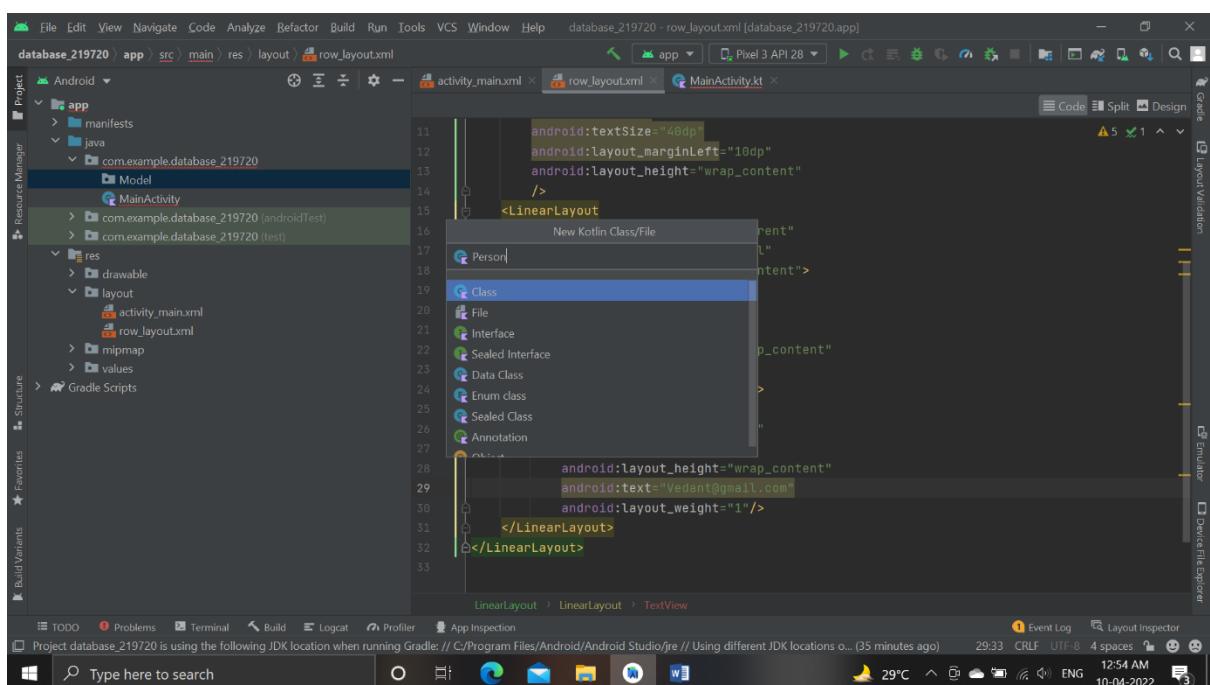
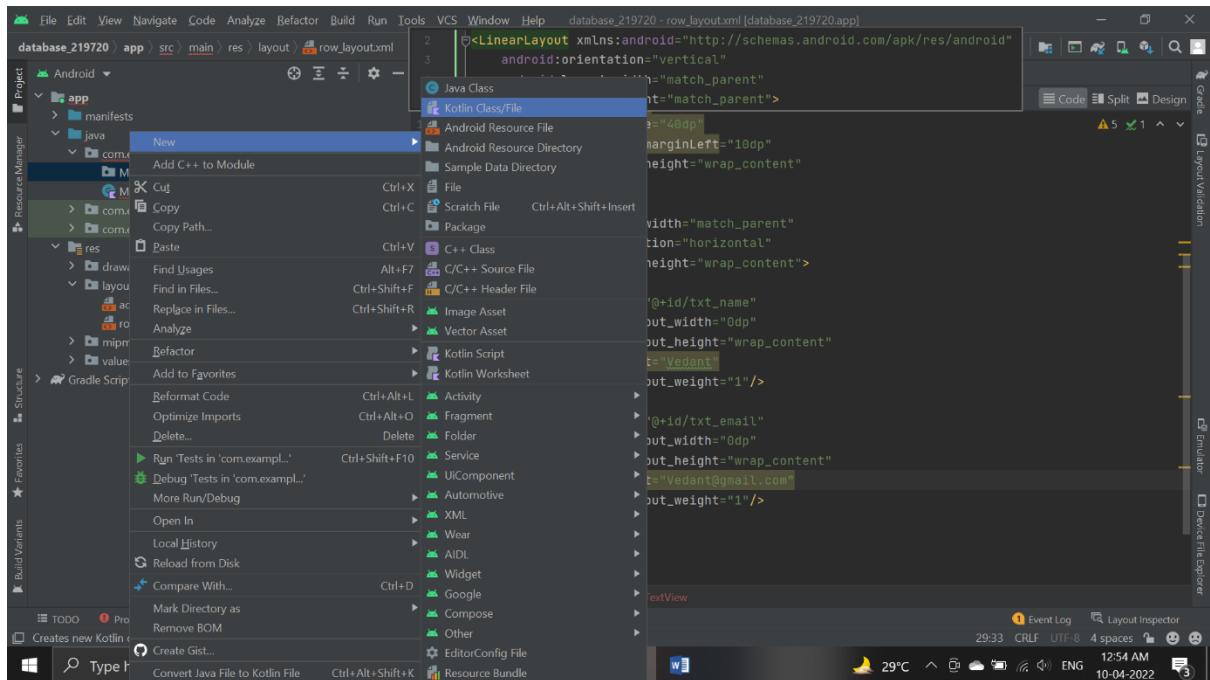




```
<?xml version="1.0" encoding="utf-8"?>
<LinearLayout xmlns:android="http://schemas.android.com/apk/res/android"
    android:orientation="vertical"
    android:layout_width="match_parent"
    android:layout_height="match_parent">
    <TextView
        android:id="@+id/txt_row_id"
        android:layout_width="match_parent"
        android:textStyle="bold"
        android:text="0"
        android:textSize="40dp"
        android:layout_marginLeft="10dp"
        android:layout_height="wrap_content"
    />
    <LinearLayout
        android:layout_width="match_parent"
        android:orientation="horizontal"
        android:layout_height="wrap_content">
        <TextView
            android:id="@+id/txt_name"
            android:layout_width="0dp"
            android:layout_height="wrap_content"
            android:text="Vedant"
            android:layout_weight="1"/>
        <TextView
            android:id="@+id/txt_email"
            android:layout_width="0dp"
            android:layout_height="wrap_content"
            android:text="Vedant@gmail.com"
            android:layout_weight="1"/>
    </LinearLayout>
</LinearLayout>
```

Step 3: Create model class for person. Right click on Java > New > Package

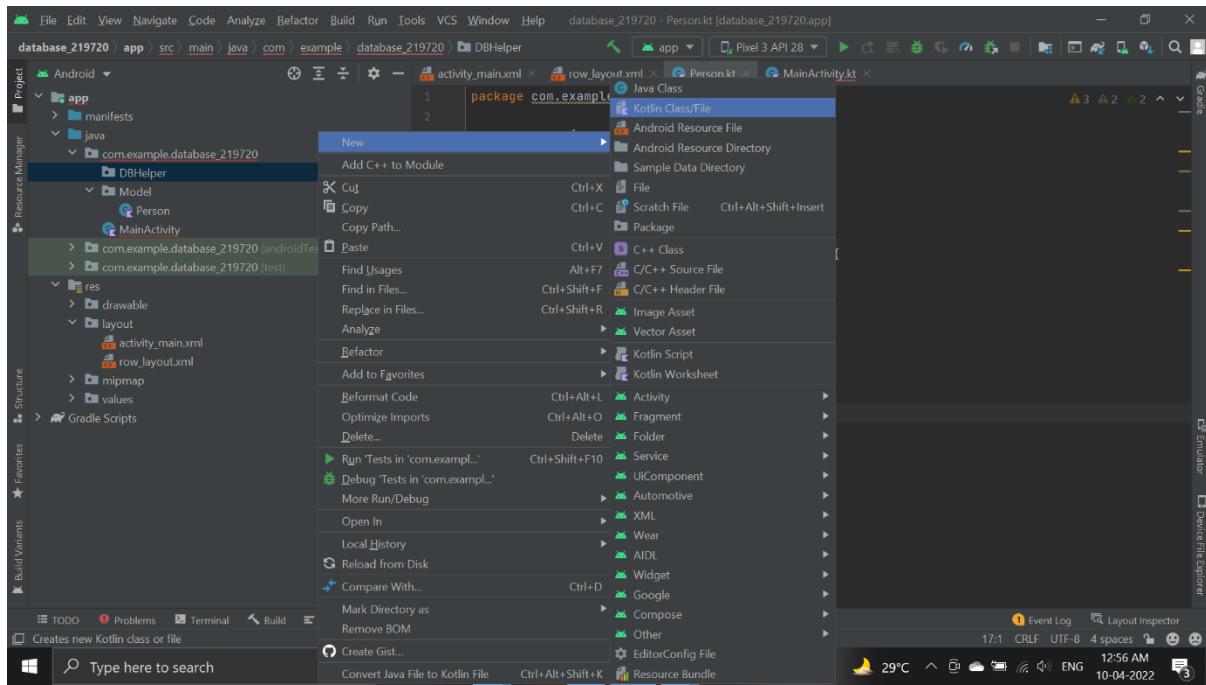




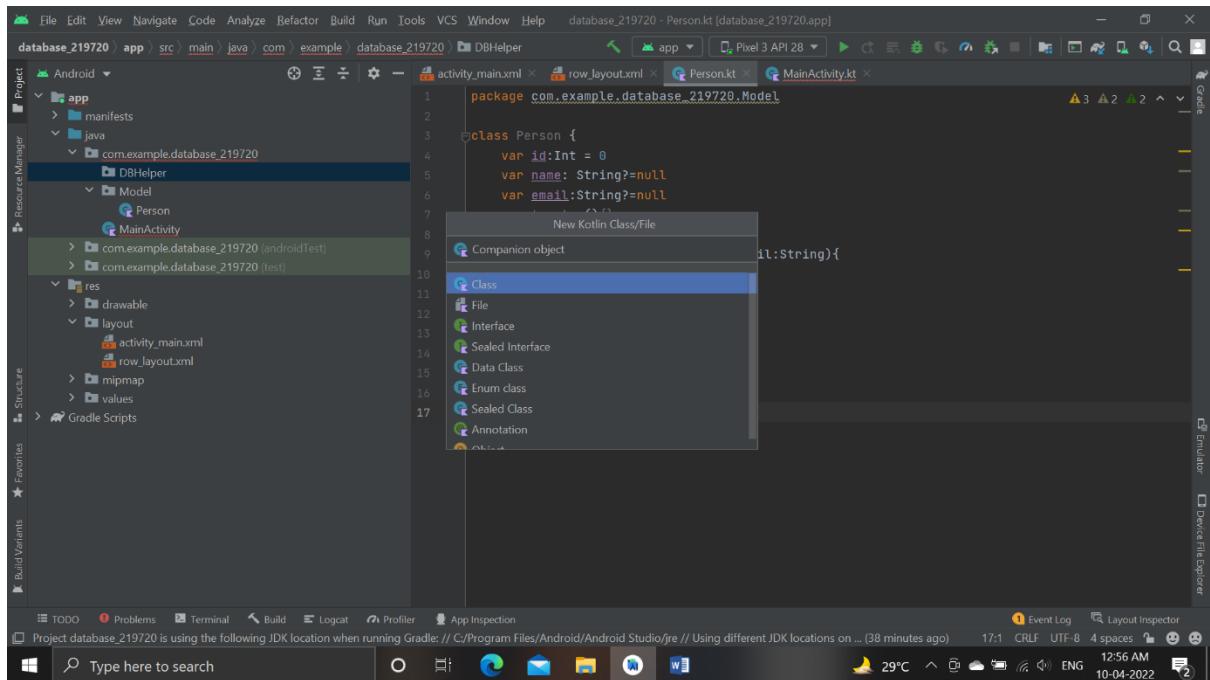
```

1 package com.example.database_219720.Model
2
3 class Person {
4     var id:Int = 0
5     var name: String?=null
6     var email:String?=null
7     constructor(){}
8
9     constructor(id:Int, name:String, email:String){
10         this.id = id
11         this.name = name
12         this.email = email
13     }
14
15 }
16
17 }
```

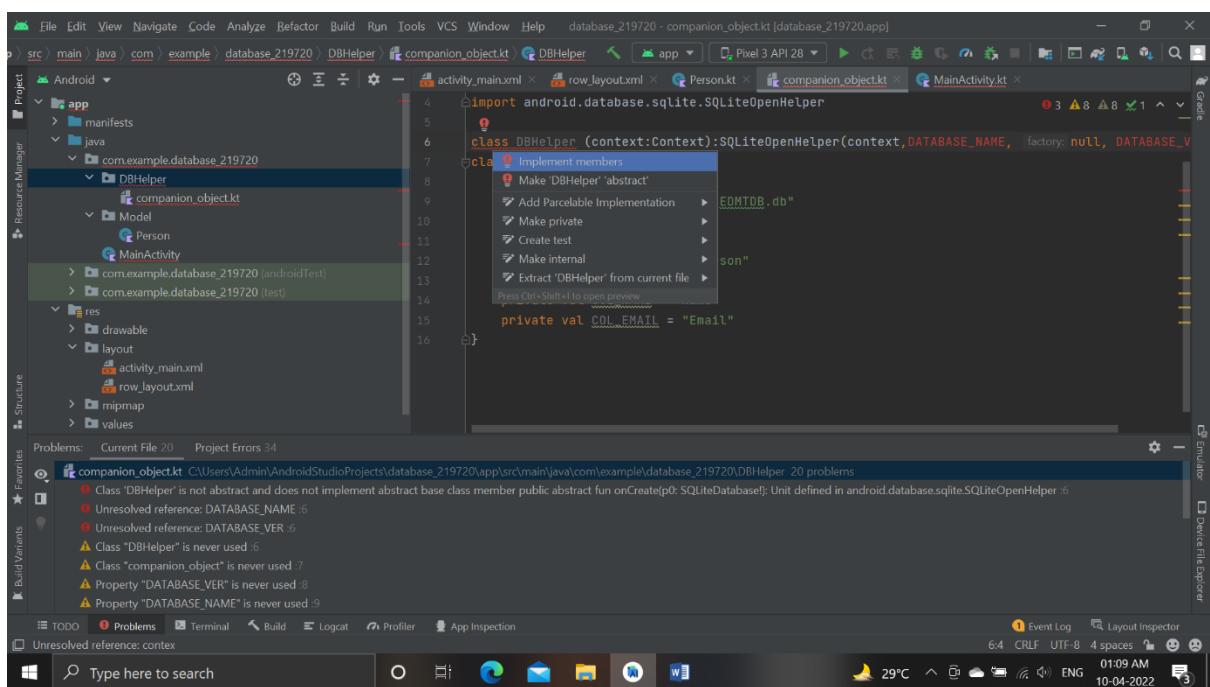
Step 4: Create new Package DBHelper and create new class file DBHelper

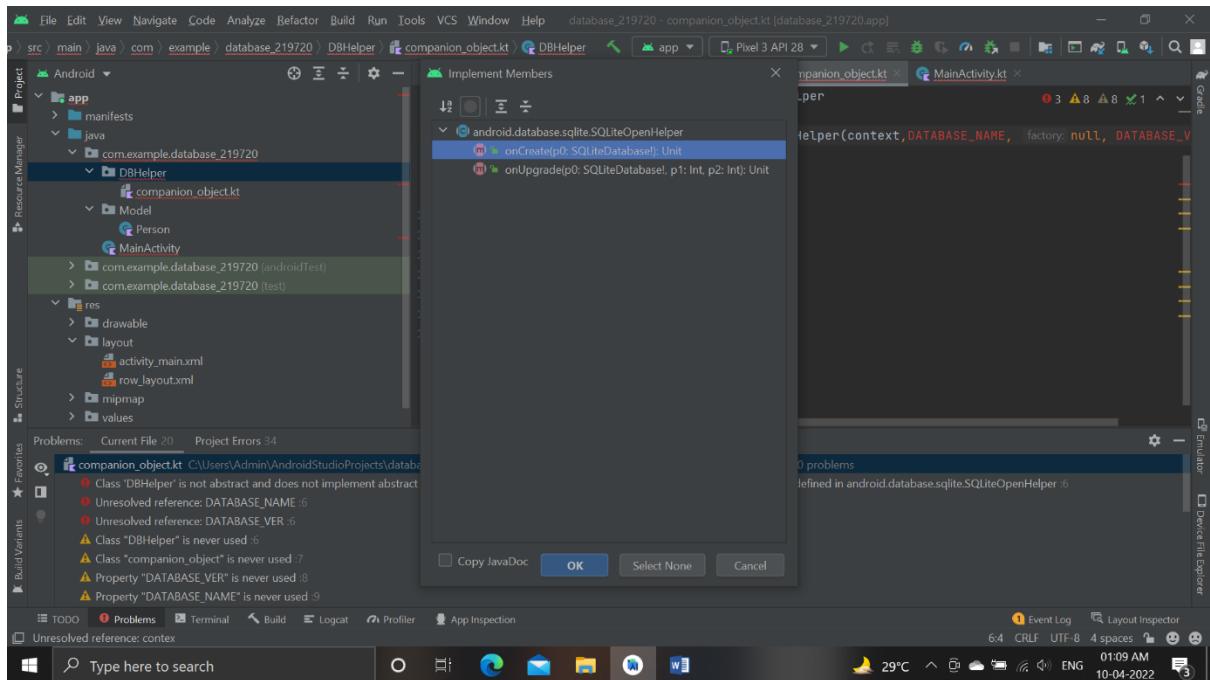


Step 5: Create companion object



Step 6: implement the default methods





```

package com.example.database_219720.DBHelper

import android.annotation.SuppressLint
import android.content.ContentValues
import android.content.Context
import android.database.sqlite.SQLiteDatabase
import android.database.sqlite.SQLiteOpenHelper
import com.example.database_219720.Model.Person

class DBHelper(context:Context):SQLiteOpenHelper(context,DATABASE_NAME,null,DATABASE_VER) {
    override fun onCreate(db: SQLiteDatabase?) {
        val CREATE_TABLE_QUERY = ("CREATE TABLE $TABLE_NAME ($COL_ID INTEGER PRIMARY KEY, $COL_NAME TEXT, $COL_EMAIL TEXT)")
        db.execSQL(CREATE_TABLE_QUERY);
    }

    override fun onUpgrade(db: SQLiteDatabase?, oldVersion: Int, newVersion: Int) {
        db.execSQL("DROP TABLE IF EXISTS $TABLE_NAME")
        onCreate(db!!)
    }

    // CRUD
    val allPerson:List<Person>
        @SuppressLint("Range")
        get() {
            val lstPersons = ArrayList<Person>()
            val selectQuery = "SELECT * FROM $TABLE_NAME"
            val db = this.writableDatabase
            val cursor = db.rawQuery(selectQuery,null)
            if (cursor.moveToFirst())
            {
                do {
                    val person = Person()
                    person.id = cursor.getInt(cursor.getColumnIndex(COL_ID))
                    ...
                } while (cursor.moveToNext())
            }
            cursor.close()
            return lstPersons
        }
}

```

```

        person.name = cursor.getString(cursor.getColumnIndex(COL_NAME))
        person.email = cursor.getString(cursor.getColumnIndex(COL_EMAIL))
        lstPersons.add(person)
    }while (cursor.moveToNext())
}
db.close()
return lstPersons
}

fun addPerson(person: Person){
    val db = this.writableDatabase
    val values = ContentValues()
    values.put(COL_ID, person.id)
    values.put(COL_NAME, person.name)
    values.put(COL_EMAIL, person.email)
    db.insert(TABLE_NAME, null, values)
    db.close()
}

fun updatePerson(person: Person): Int {
    val db = this.writableDatabase
    val values = ContentValues()
    values.put(COL_ID, person.id)
    values.put(COL_NAME, person.name)
    values.put(COL_EMAIL, person.email)
    return db.update(TABLE_NAME, values, "$COL_ID=?", arrayOf(person.id.toString()))
}

fun deletePerson(person: Person) {
    val db = this.writableDatabase

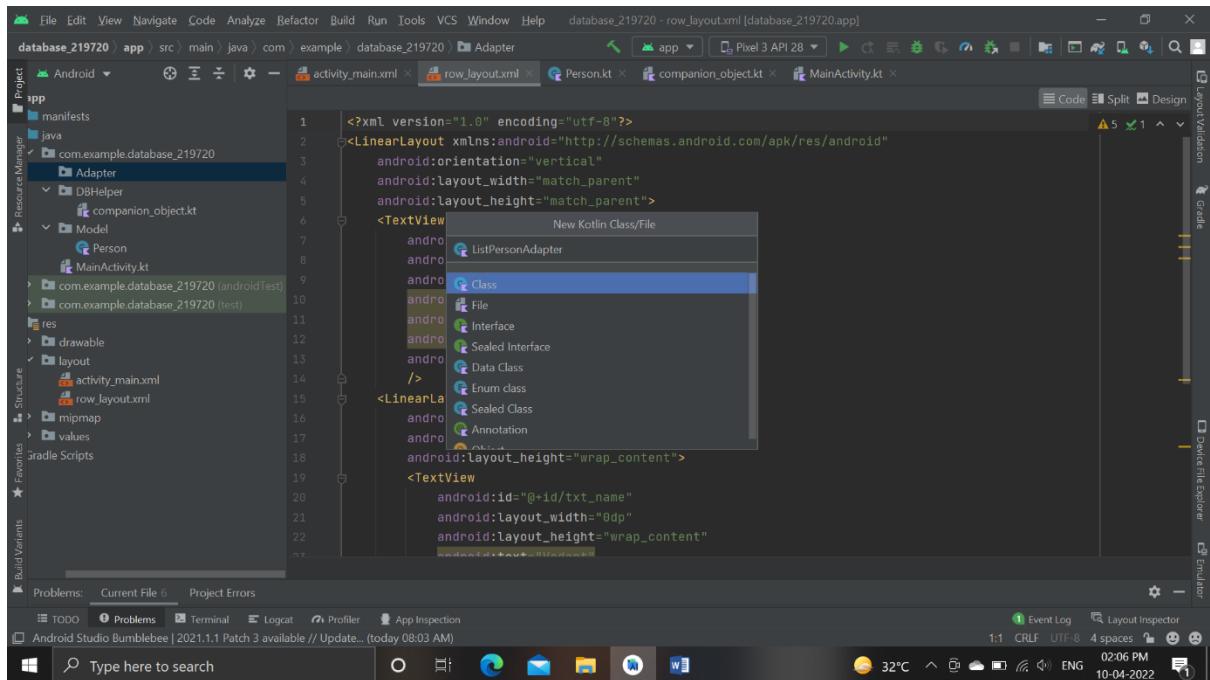
    db.delete(TABLE_NAME, "$COL_ID=?", arrayOf(person.id.toString()))
    db.close()
}

companion object{
    private val DATABASE_VER = 1
    private val DATABASE_NAME = "EDMTDB.db"

    // Table
    private val TABLE_NAME = "Person"
    private val COL_ID = "Id"
    private val COL_NAME = "Name"
    private val COL_EMAIL = "Email"
}
}

```

Step 5: Create simple Adapter for ListView



```

package com.example.database_219720.Adapter

import android.app.Activity
import android.content.Context
import android.util.Log
import android.view.LayoutInflater
import android.view.View
import android.view.ViewGroup
import android.widget.BaseAdapter
import android.widget.EditText
import android.widget.TextView
import com.example.database_219720.Model.Person
import com.example.database_219720.R

class ListPersonAdapter(internal var activity: Activity,
    internal var lstPerson: List<Person>,
    internal var edt_id: EditText,
    internal var edt_name: EditText,
    internal var edt_email: EditText): BaseAdapter() {
    internal var inflater: LayoutInflater
    init {
        inflater = activity.getSystemService(Context.LAYOUT_INFLATER_SERVICE) as LayoutInflater
    }
    override fun getView(position: Int, convertView: View?, parent: ViewGroup?): View {
        val rowView: View
        rowView = inflater.inflate(R.layout.row_layout, null)
        val rowView1 = rowView.findViewById<TextView>(R.id.txt_row_id)
        val rowView2 = rowView.findViewById<TextView>(R.id.txt_name)
        val rowView3 = rowView.findViewById<TextView>(R.id.txt_email)
        rowView1.text = lstPerson[position].id.toString()
        rowView2.text = lstPerson[position].name.toString()
        rowView3.text = lstPerson[position].email.toString()
        return rowView
    }
    override fun getItem(position: Int): Any {
        return lstPerson[position]
    }
}

```

```

override fun getItemId(position: Int): Long {
    return lstPerson[position].id.toLong()
}
override fun getCount(): Int {
    return lstPerson.size
}
}

```

mainactivity.kt

```

package com.example.database_219720

import androidx.appcompat.app.AppCompatActivity
import android.os.Bundle
import android.view.View
import android.widget.EditText
import android.widget.ListView
import android.widget.Toast
import com.example.database_219720.Adapter.ListPersonAdapter
import com.example.database_219720.DBHelper.DBHelper
import com.example.database_219720.Model.Person

class MainActivity : AppCompatActivity() {

    lateinit var db: DBHelper
    lateinit var listPerson: ArrayList<Person>
    lateinit var id: EditText
    lateinit var name: EditText
    lateinit var email: EditText
    lateinit var person: ListView
    override fun onCreate(savedInstanceState: Bundle?) {
        super.onCreate(savedInstanceState)
        setContentView(R.layout.activity_main)
        id = findViewById(R.id.edt_id)
        name = findViewById(R.id.edt_name)
        email = findViewById(R.id.edt_email)
        person = findViewById(R.id.list_persons)
        db = DBHelper(this)
        ListViewData()
    }

    fun ListViewData() {
        listPerson = db.allPerson as ArrayList<Person>
        val adapter = ListPersonAdapter(this, listPerson, id, name, email)
        person.adapter = adapter
    }

    fun add(view: View) {
        if (!id.text.isEmpty() && !name.text.isEmpty() && !email.text.isEmpty()) {
            val person = Person()
            person.id = id.text.toString().trim().toInt()
            person.name = name.text.toString().trim()
            person.email = email.text.toString().trim().toString()
            db.addPerson(person)
            id.setText("")
            name.setText("")
            email.setText("")
        }
    }
}

```

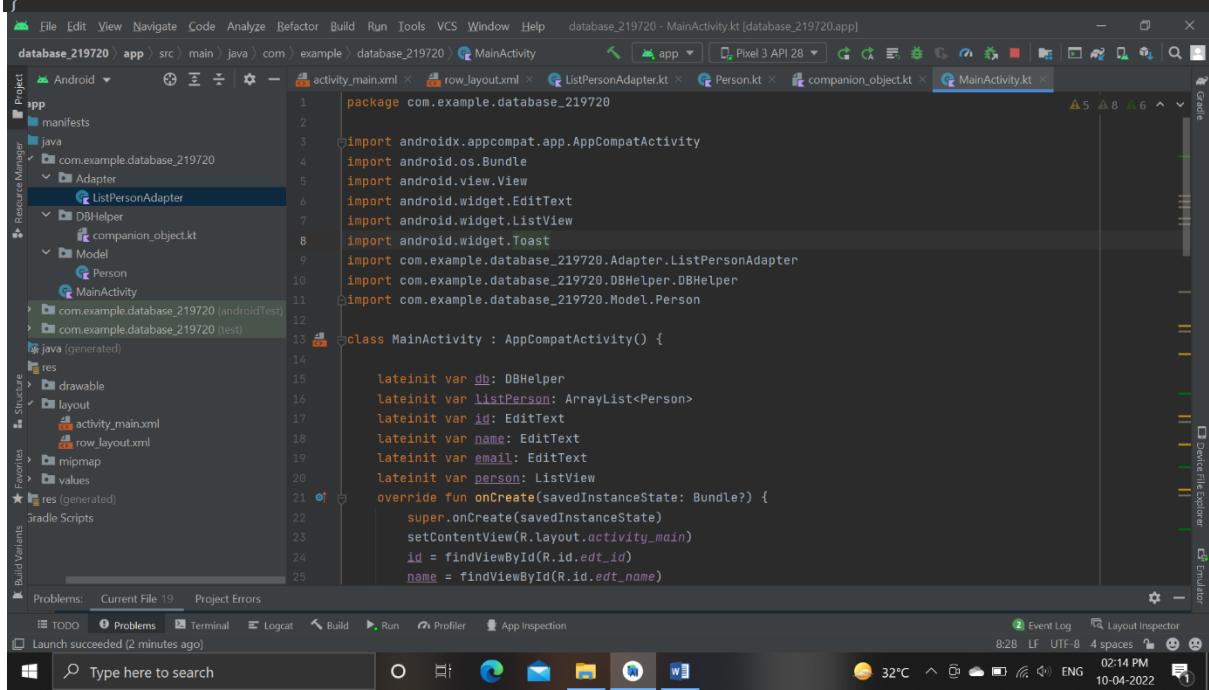
```

        ListViewData()
    } else {
        Toast.makeText(this, "Fill all Fields", Toast.LENGTH_SHORT).show()
    }
}

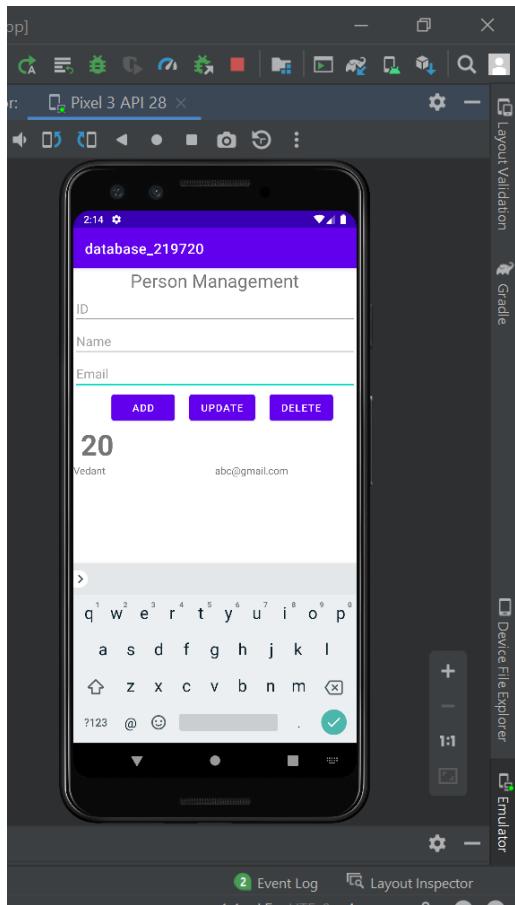
fun update(view: View) {
    if (!id.text.isEmpty() && !name.text.isEmpty() && !email.text.isEmpty()) {
        val person = Person()
        person.id = id.text.toString().trim().toInt()
        person.name = name.text.toString().trim()
        person.email = email.text.toString().trim().toString()
        db.updatePerson(person)
        ListViewData()
    } else {
        Toast.makeText(this, "Fill all Fields", Toast.LENGTH_SHORT).show()
    }
}

fun delete(view: View) {
    if (!id.text.isEmpty()) {
        val person = Person()
        person.id = id.text.toString().trim().toInt()
        db.deletePerson(person)
        ListViewData()
    } else {
        Toast.makeText(this, "TYPE ROLL_NO", Toast.LENGTH_SHORT).show()
    }
}
}

```



Output:



Practical No.10

Programming Media API and Telephone API

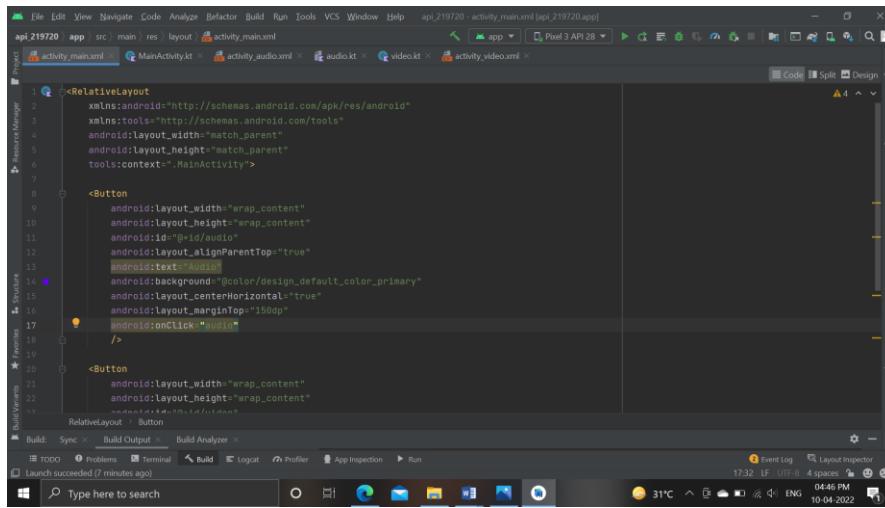
Media API

Activity main.xml

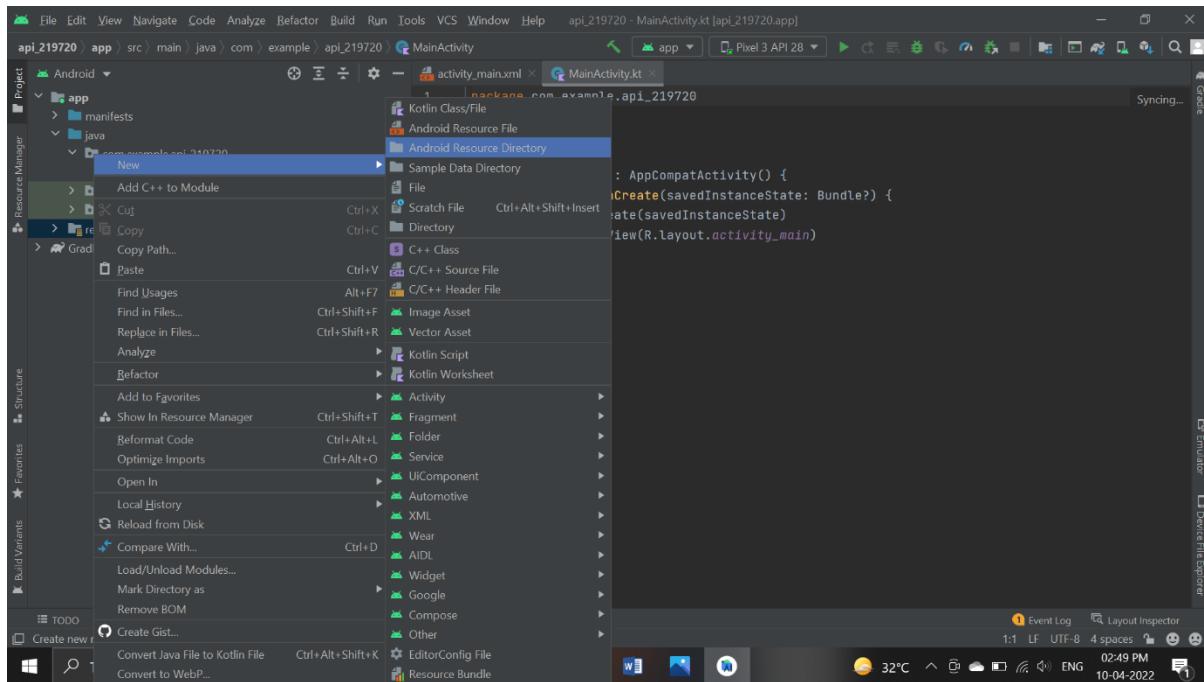
```
<RelativeLayout
    xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:tools="http://schemas.android.com/tools"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    tools:context=".MainActivity">

    <Button
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:id="@+id/audio"
        android:layout_alignParentTop="true"
        android:text="Audio"
        android:background="@color/design_default_color_primary"
        android:layout_centerHorizontal="true"
        android:layout_marginTop="150dp"
        android:onClick="audio"
    />

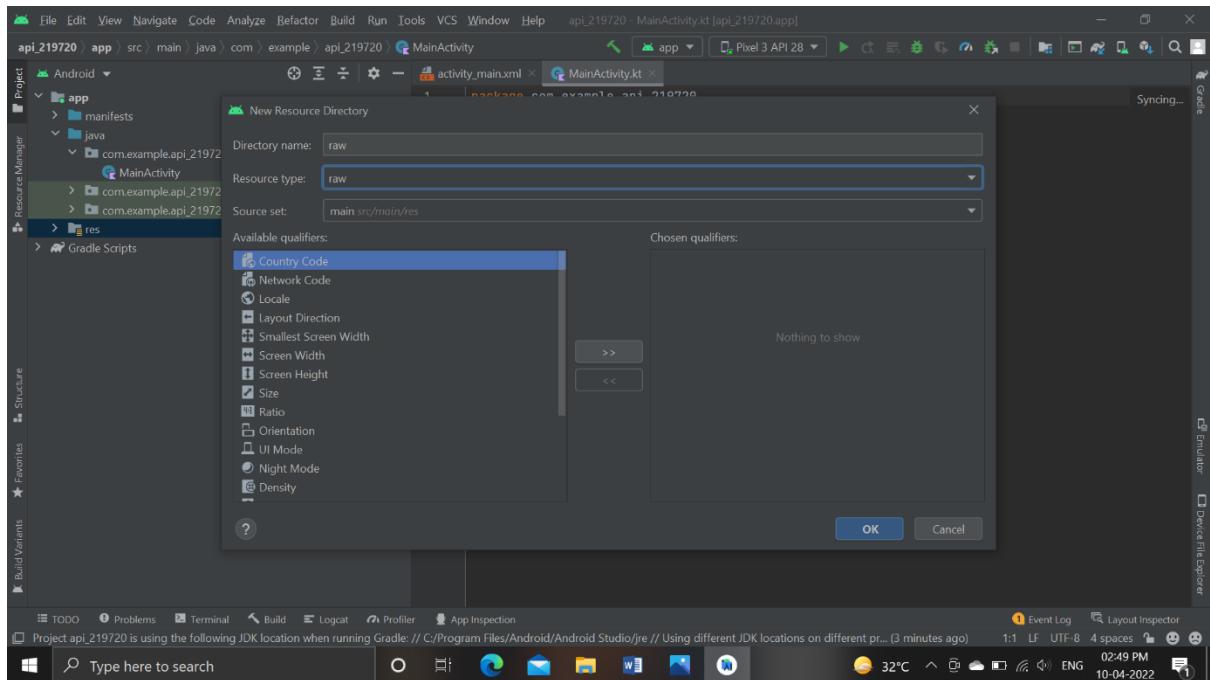
    <Button
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:id="@+id/video"
        android:text="video"
        android:background="@color/design_default_color_primary"
        android:layout_centerHorizontal="true"
        android:layout_below="@+id/audio"
        android:layout_marginTop="60dp"
        android:onClick="video"
    />
</RelativeLayout>
```



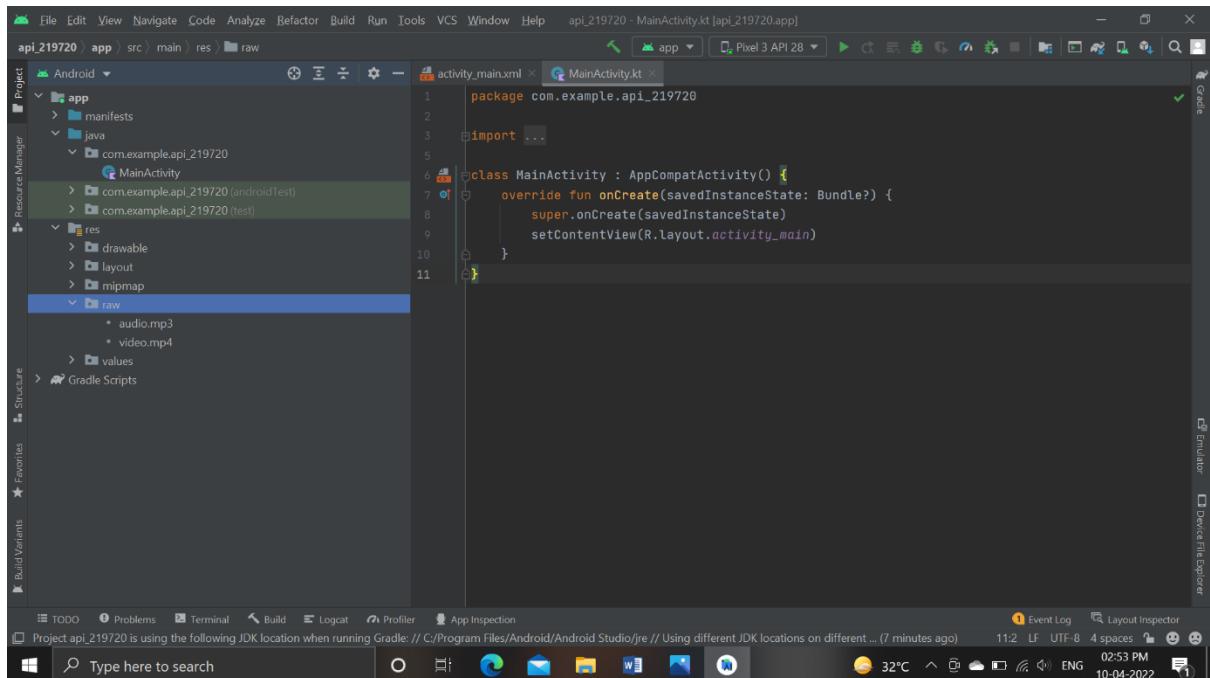
Create android resource directory under res folder to store audio or video file



Change the resource type to raw



Paste the audio (.mp3) / video(.mp4) file in the raw folder



Create the activities for audio and video files and rename them

activity_audio.xml

STUD--Talks: Follow us on for more videos and updates

```

<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"
    tools:context=".Audio">
    <SeekBar
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:id="@+id/audiobar"
        android:layout_alignParentTop="true"
        android:layout_marginTop="130dp"/>
    <Button
        android:id="@+id/play"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="Play"
        android:background="@color/design_default_color_primary"
        android:layout_below="@id/audiobar"
        android:layout_alignParentLeft="true"
        android:layout_marginLeft="40dp"
        android:layout_marginTop="150dp"
        android:layout_marginRight="70dp"
        android:onClick="play"
        />
    <Button
        android:id="@+id/stop"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="Stop"
        android:background="@color/design_default_color_primary"
        android:layout_below="@id/audiobar"
        android:layout_alignParentRight="true"
        android:layout_toRightOf="@+id/play"
        android:layout_marginRight="40dp"
        android:layout_marginLeft="70dp"
        android:layout_marginTop="150dp"
        android:onClick="stop"
        />
    <Button
        android:id="@+id/back"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="BACK"
        android:background="@color/design_default_color_primary"
        android:layout_below="@+id/play"
        android:layout_marginTop="50dp"
        android:layout_centerHorizontal="true"
        android:onClick="back"
        />
</RelativeLayout>

```

Audio.kt

```

package com.example.api_219720

import android.content.Intent
import android.media.AudioAttributes
import android.media.MediaPlayer

```

```

import android.os.Build
import android.os.Bundle
import android.os.Handler
import android.view.View
import android.widget.SeekBar
import androidx.annotation.RequiresApi
import androidx.appcompat.app.AppCompatActivity
import com.example.api_219720.R.id.audioobar

class Audio : AppCompatActivity() {
    val audiobarr = findViewById<SeekBar>(audiobar)
    lateinit var mediaPlayer: MediaPlayer
    lateinit var handler: Handler
    lateinit var runnable: Runnable
    @RequiresApi(Build.VERSION_CODES.LOLLIPOP)
    override fun onCreate(savedInstanceState: Bundle?) {
        super.onCreate(savedInstanceState)
        setContentView(R.layout.activity_audio)
        mediaPlayer = MediaPlayer.create(applicationContext, R.raw.audio)
        var attributes = AudioAttributes.Builder()
            .setContentType(AudioAttributes.CONTENT_TYPE_MUSIC)
            .build()
        mediaPlayer.setAudioAttributes(attributes)
        handler = Handler()
        mediaPlayer.setOnPreparedListener(object : MediaPlayer.OnPreparedListener {
            override fun onPrepared(mp: MediaPlayer?) {
                audiobarr.max = mediaPlayer.duration
                changeprogress()
                mediaPlayer.start()
            }
        })
        audiobarr.setOnSeekBarChangeListener(object : SeekBar.OnSeekBarChangeListener {
            override fun onProgressChanged(seekBar: SeekBar?, progress: Int, input: Boolean) {
                // Toast.makeText(applicationContext, progress.toString(), Toast.LENGTH_SHORT).show()
                if (input) {
                    mediaPlayer.seekTo(progress)
                }
            }
            override fun onStartTrackingTouch(seekBar: SeekBar?) {
            }
            override fun onStopTrackingTouch(seekBar: SeekBar?) {
            }
        })
    }
    fun changeprogress() {
        audiobarr.progress = mediaPlayer.currentPosition
        if (mediaPlayer.isPlaying) {
            runnable = Runnable {
                changeprogress()
            }
            handler.postDelayed(runnable, 1000)
        }
    }
    fun play(view: View) {
        audiobarr.progress = mediaPlayer.currentPosition
        mediaPlayer.start()
    }
    fun stop(view: View) {
        mediaPlayer.pause()
    }
}

```

```

fun back(view:View){
    startActivity(Intent(this,MainActivity::class.java))
}

override fun onResume() {
    super.onResume()
    mediaPlayer.start()
    audiobarr.progress = mediaPlayer.currentPosition
}
override fun onPause() {
    super.onPause()
    mediaPlayer.pause()
}
override fun onDestroy() {
    super.onDestroy()
    mediaPlayer.release()
    handler.removeCallbacks(runnable)
}
}

```

activity_video.xml

```

<RelativeLayout
    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=".video">

    <VideoView
        android:layout_width="match_parent"
        android:layout_height="match_parent"
        android:id="@+id/videoView"/>

</RelativeLayout>

```

video.kt

```

package com.example.api_219720

import android.net.Uri
import android.os.Bundle
import android.widget.MediaController
import android.widget.SeekBar
import androidx.appcompat.app.AppCompatActivity
import com.example.api_219720.R.id.videoView
import android.widget.VideoView

class video : AppCompatActivity() {
    val videoVieww = findViewById<VideoView>(videoView)
    override fun onCreate(savedInstanceState: Bundle?) {

        super.onCreate(savedInstanceState)
        setContentView(R.layout.activity_video)
        var mediaController= MediaController(this)
        var path="android.resource://com.example.api_219720/${R.raw.video}"
        var uri= Uri.parse(path)
    }
}

```

```

        videoVieww.setVideoURI(uri)
        mediaController.setAnchorView(videoVieww)
        videoVieww.setMediaController(mediaController)
        videoVieww.start()
    }
}

```

MainActivity.kt

```

package com.example.api_219720

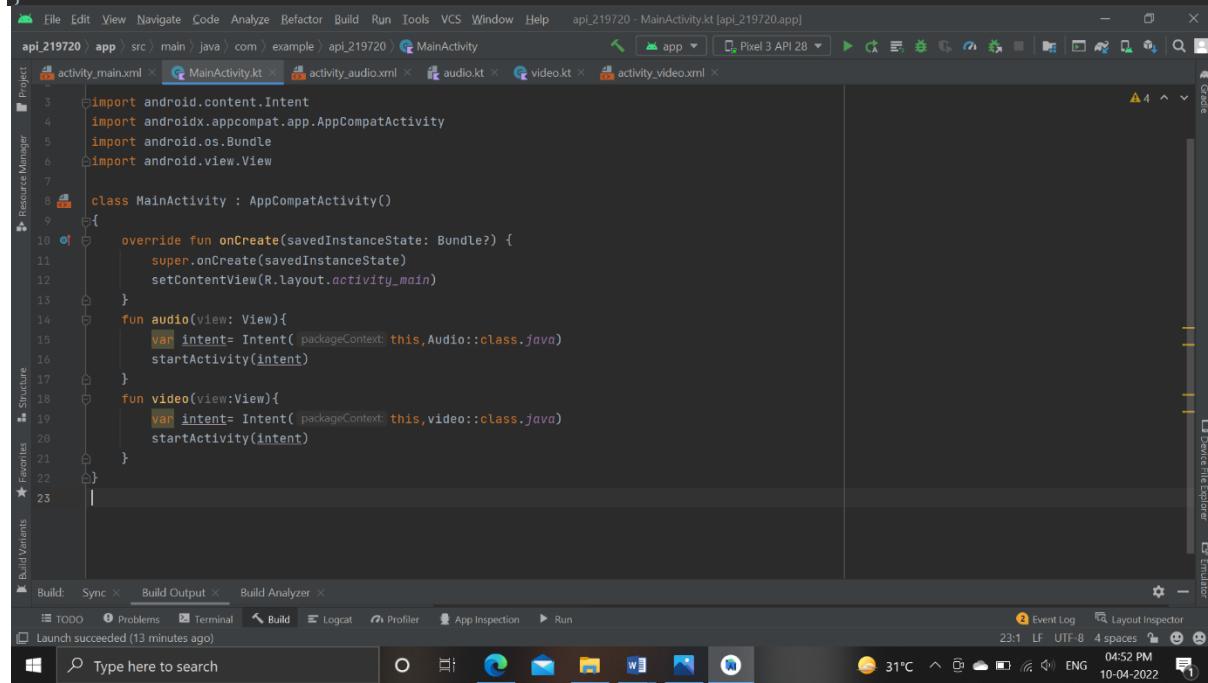
import android.content.Intent
import androidx.appcompat.app.AppCompatActivity
import android.os.Bundle
import android.view.View

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

    fun audio(view: View){
        var intent= Intent(this,Audio::class.java)
        startActivity(intent)
    }

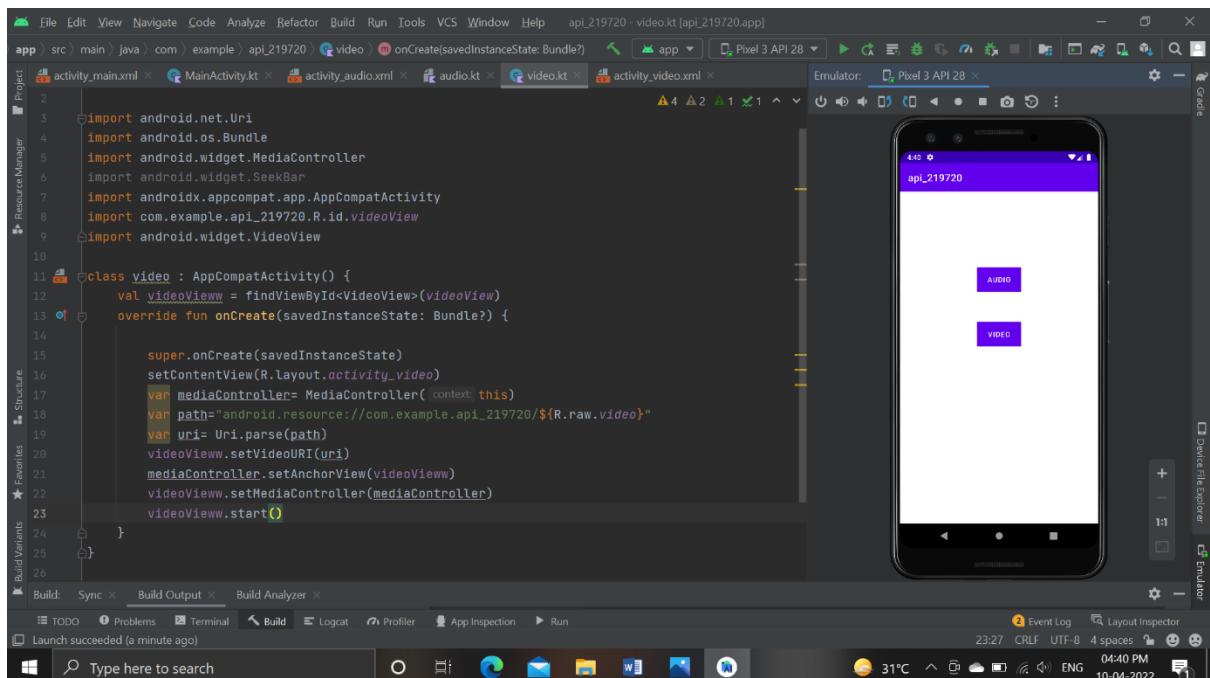
    fun video(view:View){
        var intent= Intent(this,video::class.java)
        startActivity(intent)
    }
}

```



Output:

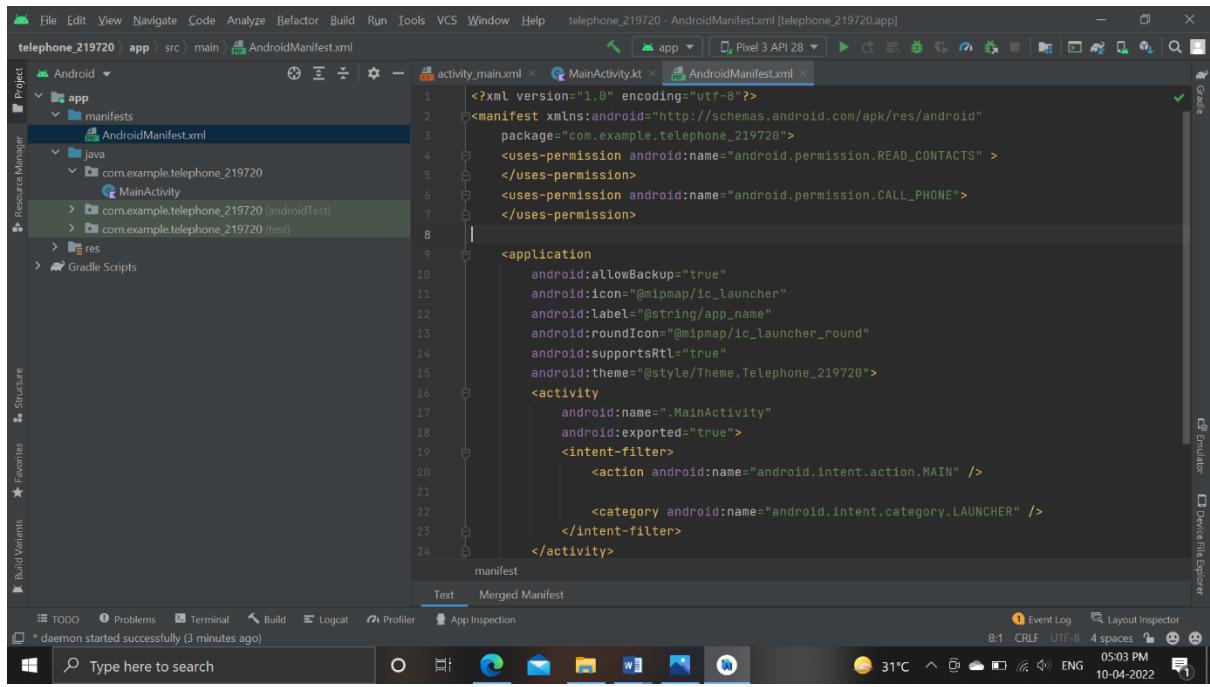
STUD--Talks: Follow us on for more videos and updates



Telephone API

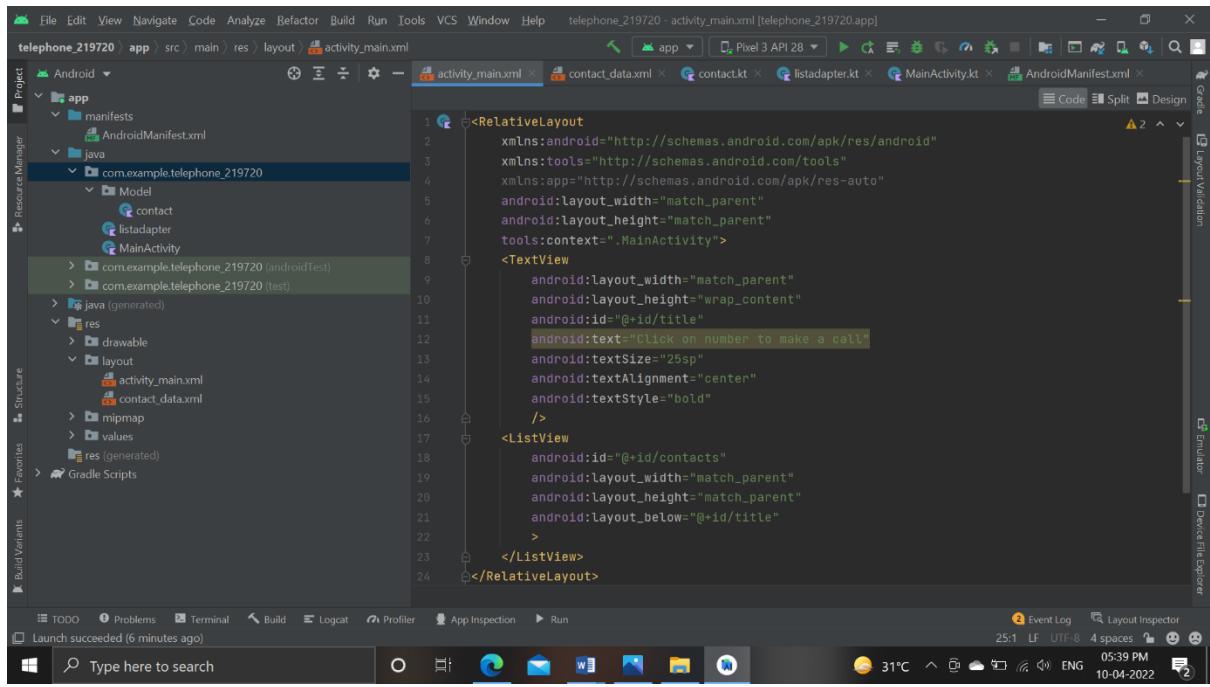
Add the following permissions in androidmanifest.xml file in the manifest tag

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



Activity_main.xml

```
<RelativeLayout
    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="match_parent"
        android:layout_height="wrap_content"
        android:id="@+id/title"
        android:text="Click on number to make a call"
        android:textSize="25sp"
        android:textAlignment="center"
        android:textStyle="bold"
        />
    <ListView
        android:id="@+id/contacts"
        android:layout_width="match_parent"
        android:layout_height="match_parent"
        android:layout_below="@+id/title"
        />
</ListView>
</RelativeLayout>
```

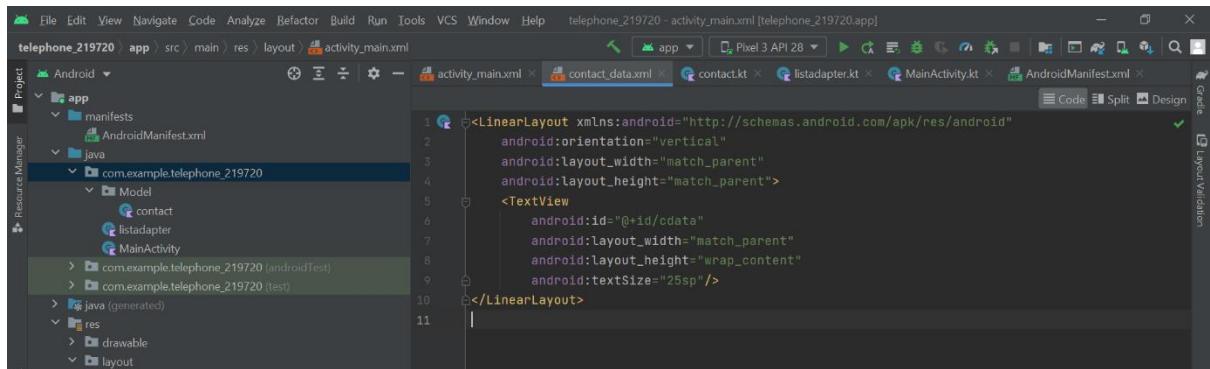


Create a new xml resource file for inflating in listadapter and rename it (contact_data.xml)

```

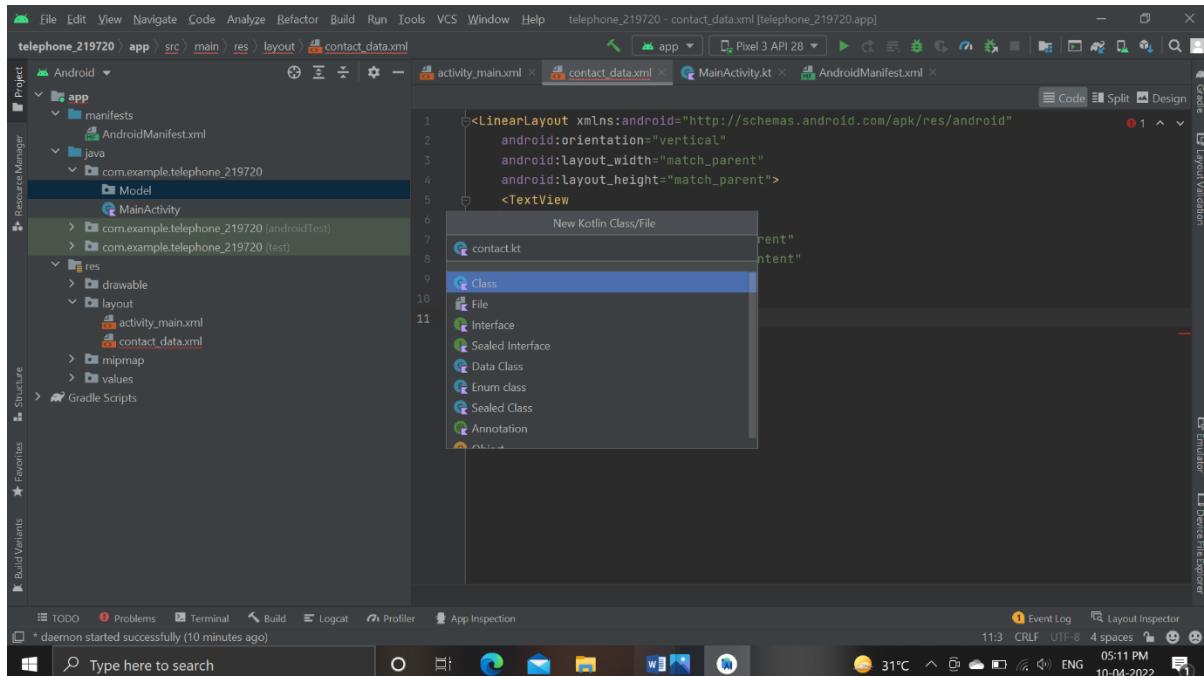
<LinearLayout xmlns:android="http://schemas.android.com/apk/res/android"
    android:orientation="vertical"
    android:layout_width="match_parent"
    android:layout_height="match_parent">
    <TextView
        android:id="@+id/cdata"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:textSize="25sp"/>
</LinearLayout>

```



Create a model class from storing the contact details like name, phone number etc

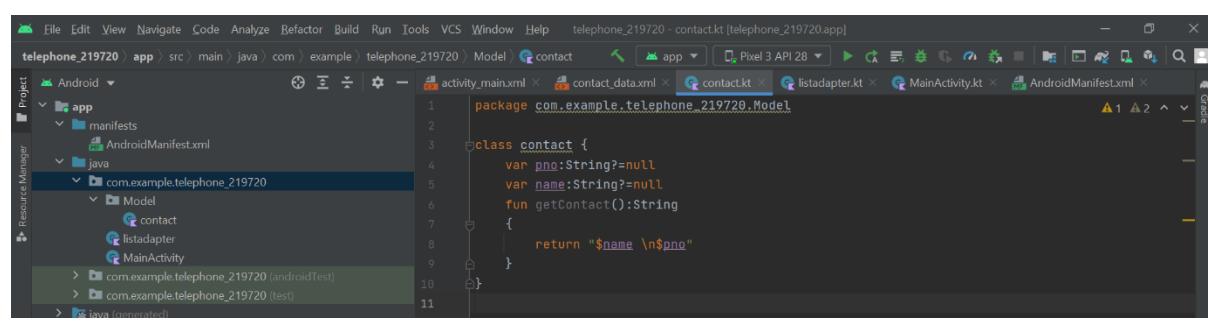
contact.kt



The screenshot shows the Android Studio interface with the project 'telephone_219720' open. In the Project tool window, under the 'app' module's 'java' directory, a 'Model' package is selected. A context menu is open over the 'Model' package, with the 'New Kotlin Class/File' option highlighted. A dropdown menu shows 'contact.kt' as the selected class name. The code editor below contains the initial skeleton of the 'contact' class:

```
package com.example.telephone_219613.Model

class contact {
```



The screenshot shows the completed 'contact' class definition in the code editor. The class now includes variable declarations for 'pno' and 'name', and a 'getContact()' function that returns a string concatenating 'name' and 'pno' separated by a newline character.

```
package com.example.telephone_219720.Model

class contact {
    var pno:String?=null
    var name:String?=null
    fun getContact():String
    {
        return "$name \n$pno"
    }
}
```

Create a listadapter class for storing the contact details in list

listadapter.kt

```

package com.example.telephone_219720.Model

class contact {
    var pno:String?=null
    var name:String?=null
    fun getContact():String
}

New Kotlin Class/File
listadapter
Class
File
Interface
Sealed Interface
Data Class
Enum class
Sealed Class
Annotation
Object

```

```

package com.example.telephone_219613

import android.app.Activity
import android.app.AlertDialog
import android.content.DialogInterface
import android.content.Intent
import android.net.Uri
import android.view.LayoutInflater
import android.view.View
import android.view.ViewGroup
import android.widget.BaseAdapter
import android.widget.Toast
import com.example.telephone_219613.Model.contact

class listadapter(var activity: Activity, var contactlist:ArrayList<contact>): BaseAdapter() {
    override fun getView(position: Int, convertView: View?, parent: ViewGroup?): View {
        var inflater= LayoutInflater.from(activity)
        var view=inflater.inflate(R.layout.contact_data,null)

        view.setOnClickListener {
            var alertDialog= AlertDialog.Builder(activity)
            alertDialog.setTitle("Make a call")
            alertDialog.setMessage("to:${contactlist[position].name}\n${contactlist[position].pno}")
            alertDialog.setPositiveButton("Yes", DialogInterface.OnClickListener {
                dialog, which ->
                try {
                    var intent = Intent(Intent.ACTION_CALL)
                    intent.data = Uri.parse("tel:${contactlist[position].pno}")
                    activity.startActivity(intent)
                }
                catch(e:Exception){
                    Toast.makeText(activity,"Please allow the permission" +
                        "",Toast.LENGTH_SHORT).show()
                }
            })
            alertDialog.setNegativeButton("No",DialogInterface.OnClickListener { dialog, which ->
                dialog.cancel()
            })
        }
    }
}

```

```

        })
        alertDialog.create()
        alertDialog.show()
    }
    return view
}

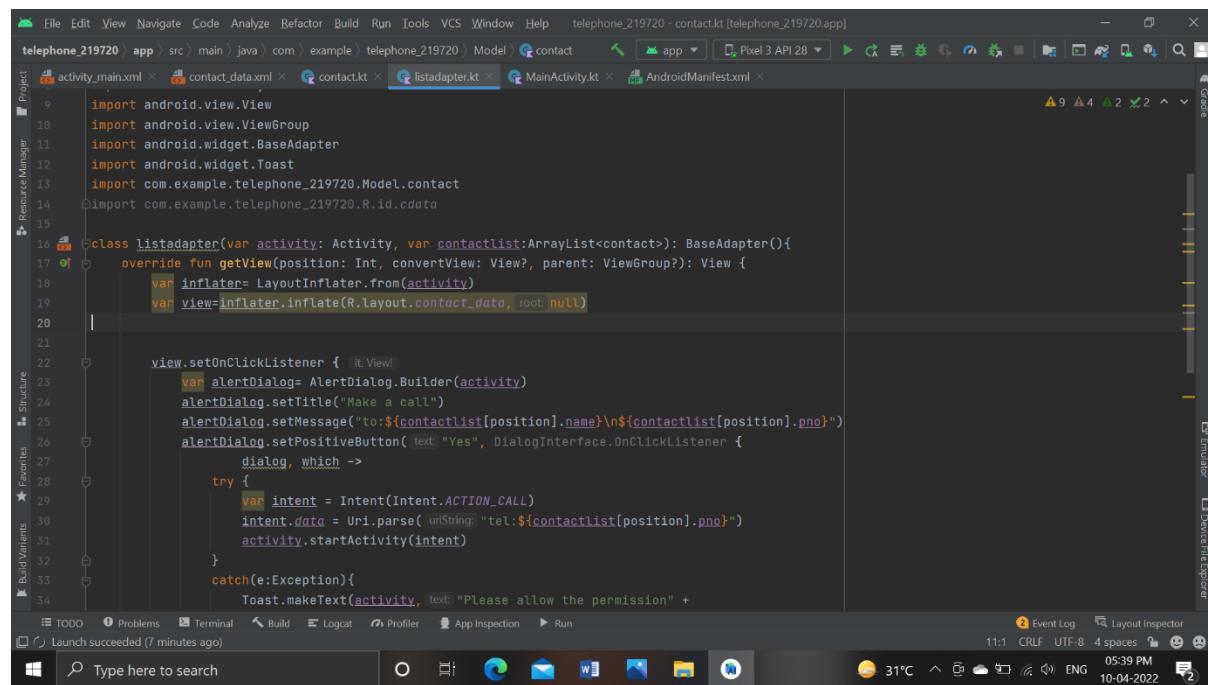
override fun getItem(position: Int): Any {
    return contactlist[position]
}

override fun getItemId(position: Int): Long {
    return position.toLong()
}

override fun getCount(): Int {
    return contactlist.size
}

}

```



MainActivity.kt

```

package com.example.telephone_219613

import android.Manifest
import android.annotation.SuppressLint
import android.content.pm.PackageManager
import android.database.Cursor
import androidx.appcompat.app.AppCompatActivity
import android.os.Bundle
import android.provider.ContactsContract
import android.widget.Toast
import androidx.core.app.ActivityCompat
import androidx.core.content.ContextCompat
import com.example.telephone_219613.Model.contact

```

```

class MainActivity : AppCompatActivity() {
    lateinit var adapter:listadapter
    lateinit var contactList:ArrayList<contact>
    var cursor: Cursor?=null
    override fun onCreate(savedInstanceState: Bundle?) {
        super.onCreate(savedInstanceState)
        setContentView(R.layout.activity_main)
        var permissionCode1= ContextCompat.checkSelfPermission(this,
Manifest.permission.READ_CONTACTS)
        var permissionCode2= ContextCompat.checkSelfPermission(this, Manifest.permission.CALL_PHONE)
        if(permissionCode1== PackageManager.PERMISSION_GRANTED &&
permissionCode2==PackageManager.PERMISSION_GRANTED)
        {
            showContacts()
        }
        else{
            ActivityCompat.requestPermissions(this,
arrayOf(Manifest.permission.READ_CONTACTS,Manifest.permission.CALL_PHONE),100)
        }
        adapter= listadapter(this,contactList)
        contacts.adapter=adapter
    }

    override fun onRequestPermissionsResult(requestCode: Int, permissions: Array<out String>, grantResults: IntArray) {
        super.onRequestPermissionsResult(requestCode, permissions, grantResults)
        if(requestCode==100){
            showContacts()
        }
        else{
            Toast.makeText(this,"Please Provide the Contact permission",Toast.LENGTH_SHORT).show()
        }
    }
    @SuppressLint("Range")
    private fun showContacts() {
        cursor=contentResolver.query(
            ContactsContract.CommonDataKinds.Phone.CONTENT_URI,
            null,null,null,ContactsContract.Contacts.SORT_KEY_PRIMARY )
        contactList=ArrayList<contact>()
        var number: String?
        var lastnumber:String?=null
        while(cursor!!.moveToNext())
        {

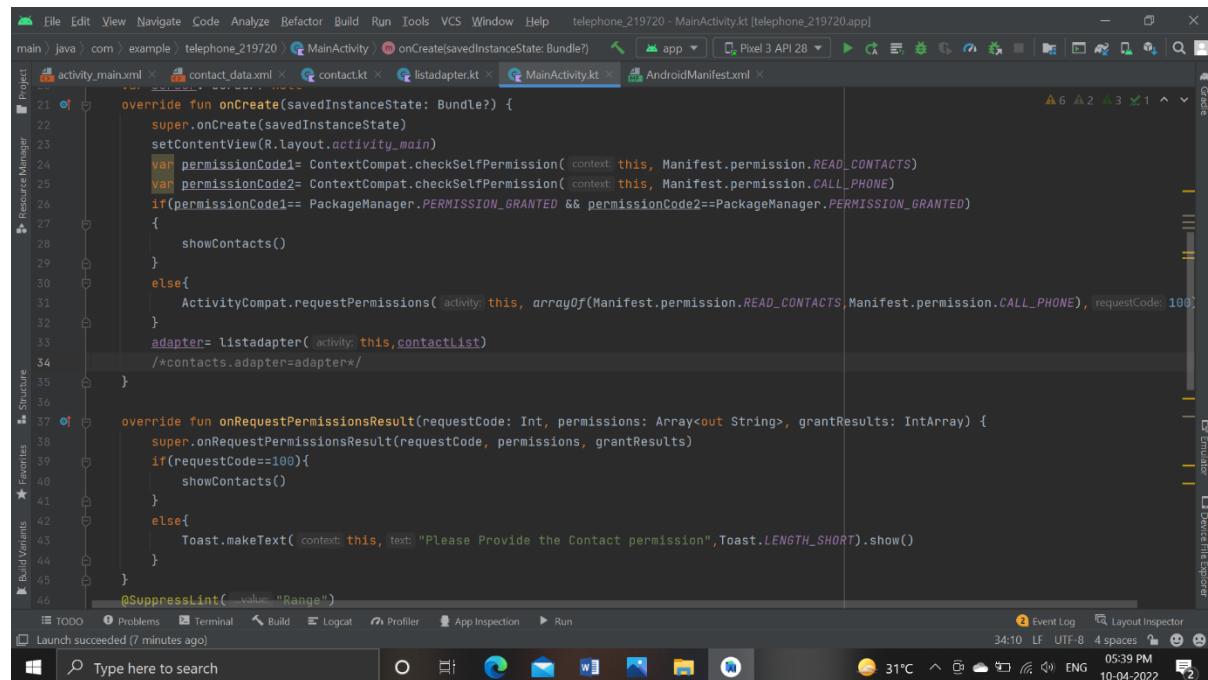
number=(cursor!!.getString(cursor.getColumnIndex(ContactsContract.CommonDataKinds.Phone.NUMBER)))
        )
        if(number!=null ) {
            number=number.replace("\\s".toRegex(),"")
            if (!number!!.equals(lastnumber)) {
                lastnumber = number

                var contact = contact()
                contact.name =
cursor!!.getString(cursor.getColumnIndex(ContactsContract.CommonDataKinds.Phone.DISPLAY_NAME))

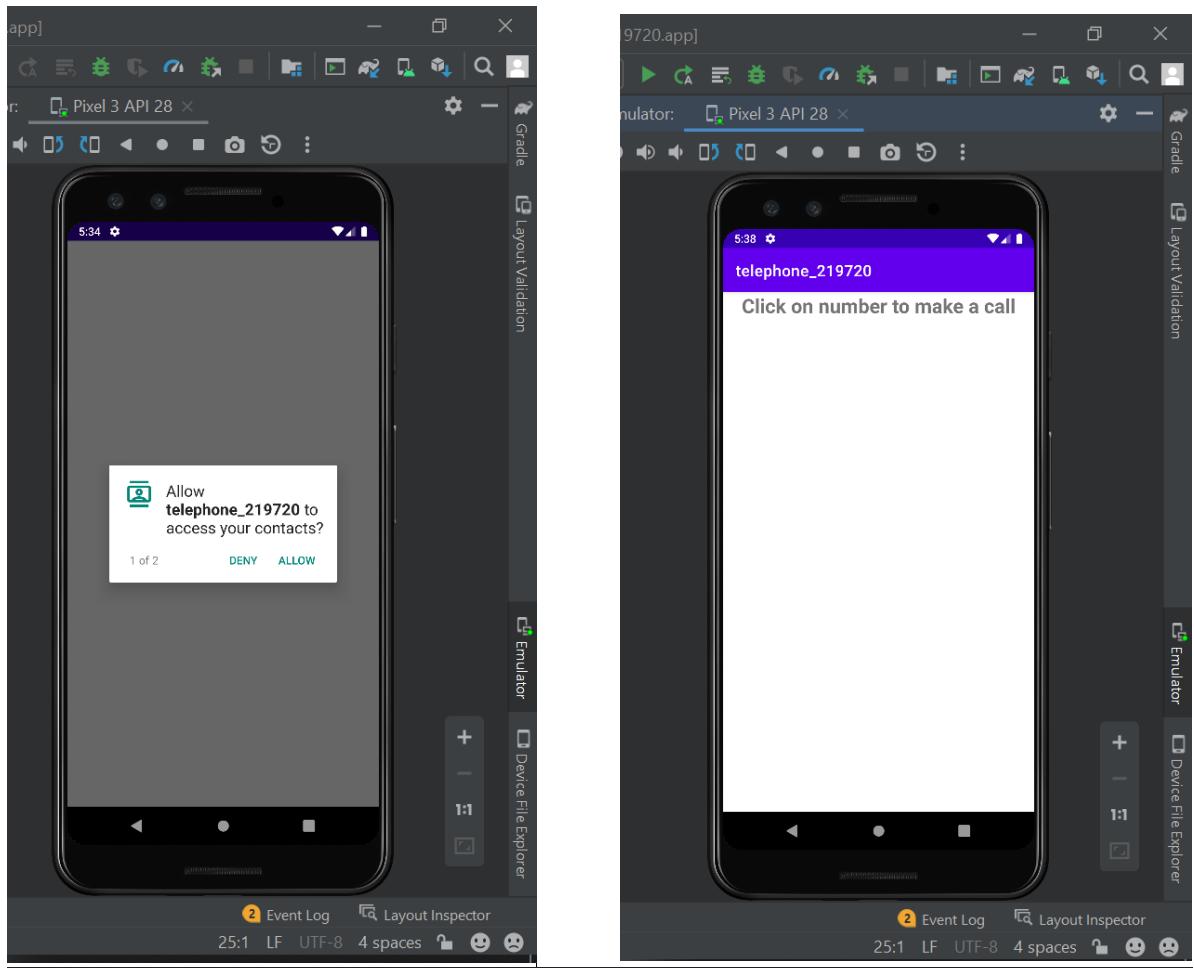
                contact.pno =
cursor!!.getString(cursor.getColumnIndex(ContactsContract.CommonDataKinds.Phone.NUMBER))

```

```
    contactList.add(contact)
}
}
}
cursor!!.close()
}
}
```



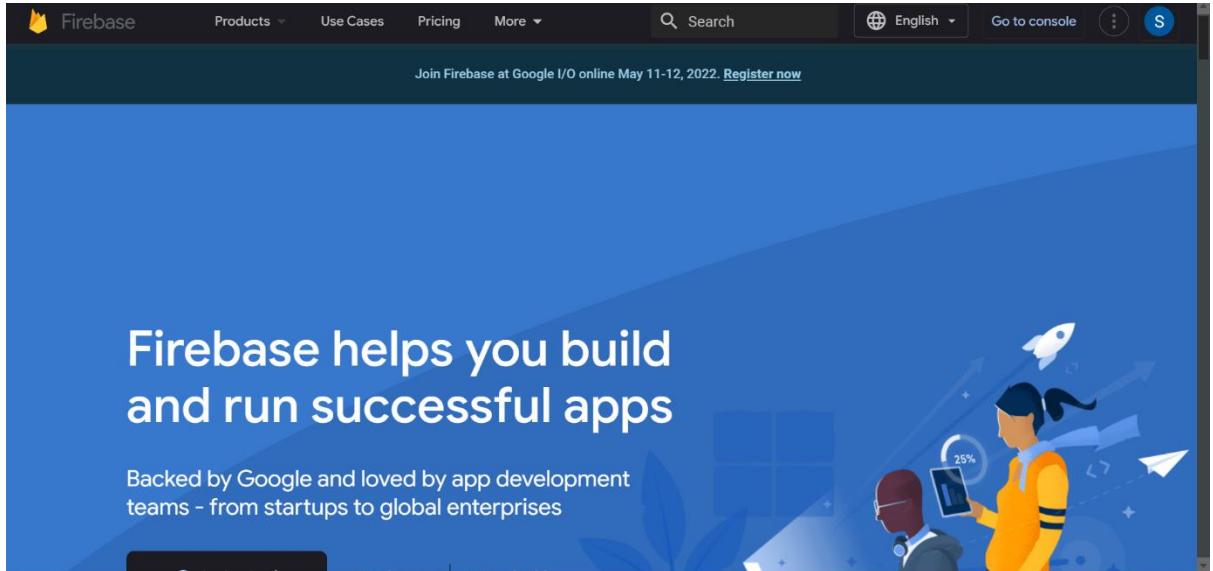
Output:



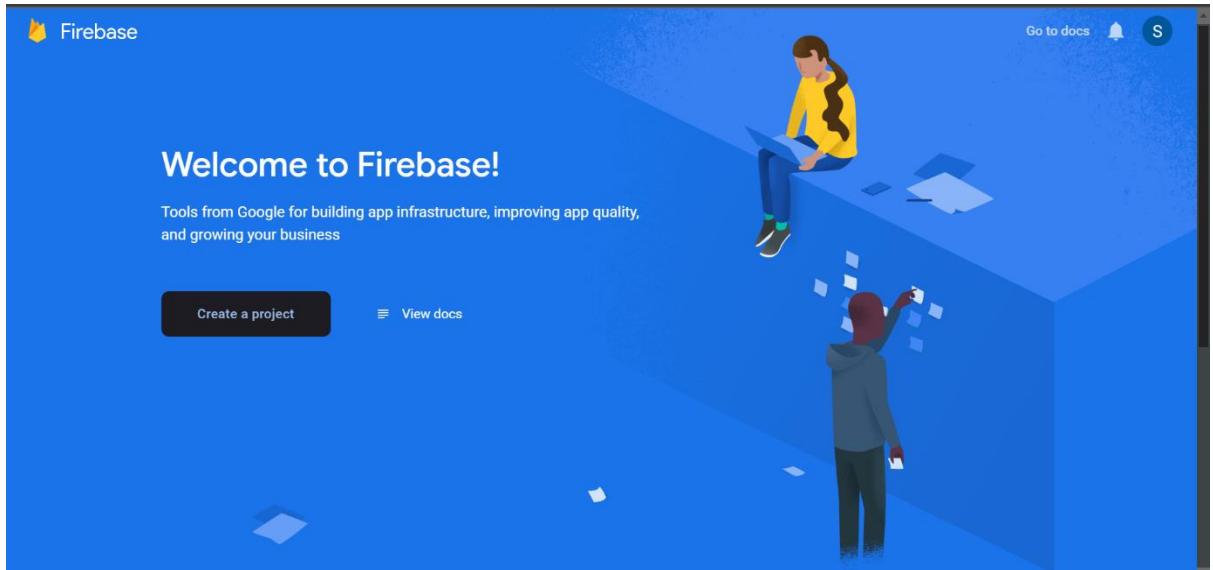
Practical No.11

Firebase Real-time Database

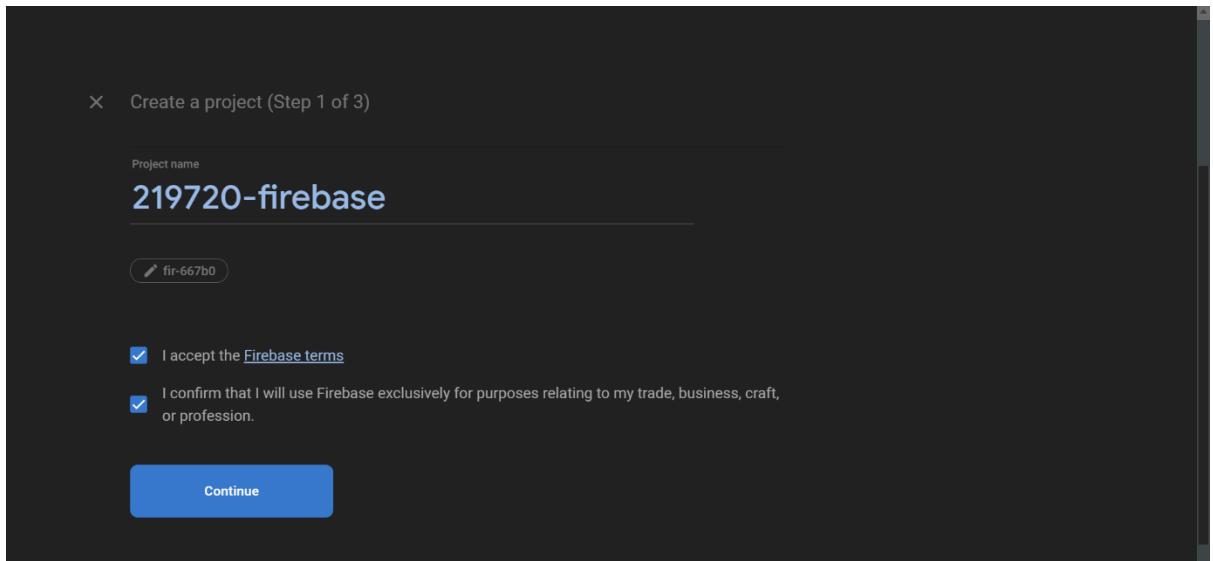
1. Open <https://firebase.google.com/>



2. Go to console



3. Create new Project
4. Give the project name
5. Continue

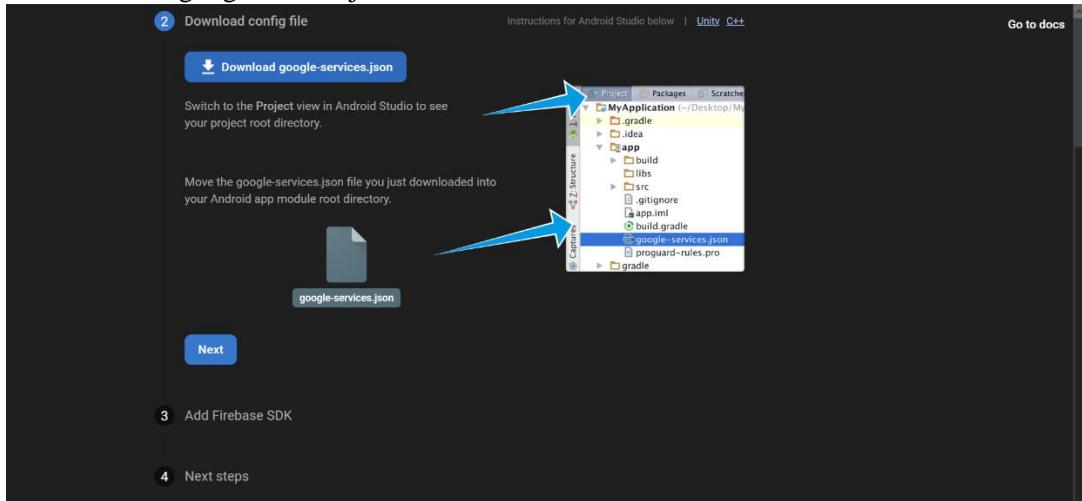


6. Click on the android button and paste the package name of the android project in the package name field and click on register app button

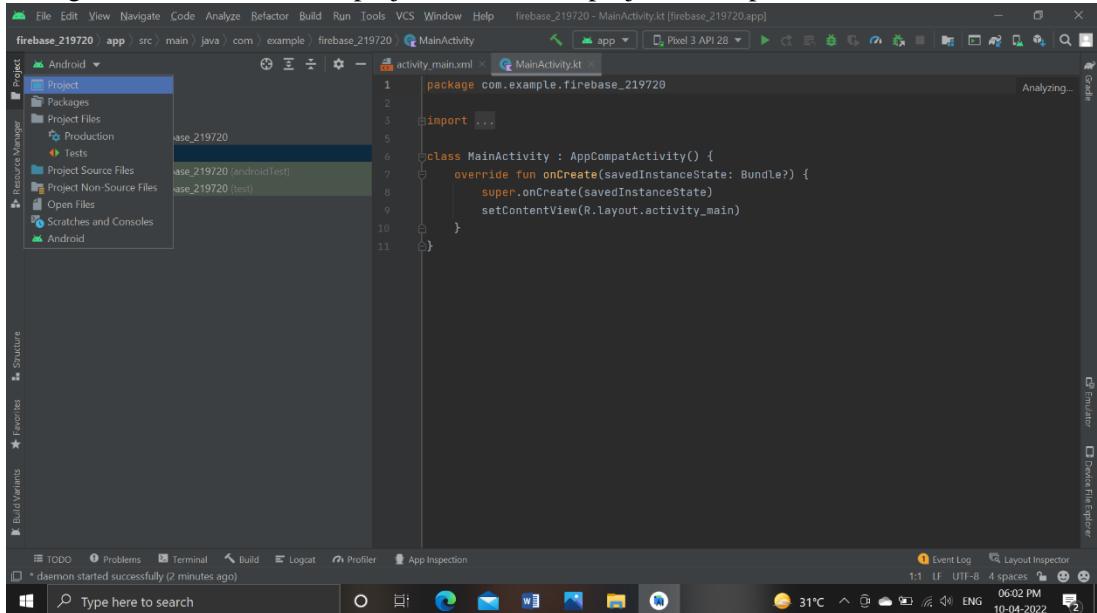
The screenshot shows the Firebase Project Overview page for the project '219720-firebase'. The sidebar on the left includes sections for Build, Release & Monitor, Analytics, Engage, and Extensions. The main content area features a blue background with the project name '219720-firebase' and a call-to-action 'Get started by adding Firebase to your app'. It includes icons for Android, iOS+, and a browser, along with a 'Add an app to get started' button. An illustration of two people interacting with a large mobile device is present.

The screenshot shows the 'Add Firebase to your Android app' registration form. Step 1: Register app. It asks for the 'Android package name' (optional) which is filled with 'com.example.firebaseio_219720'. It also asks for an 'App nickname (optional)' which is filled with 'firebase_219720'. A note at the bottom states: 'Required for Dynamic Links, and Google Sign-In or phone number support in Auth. Edit SHA-1s in Settings.' A 'Register app' button is at the bottom.

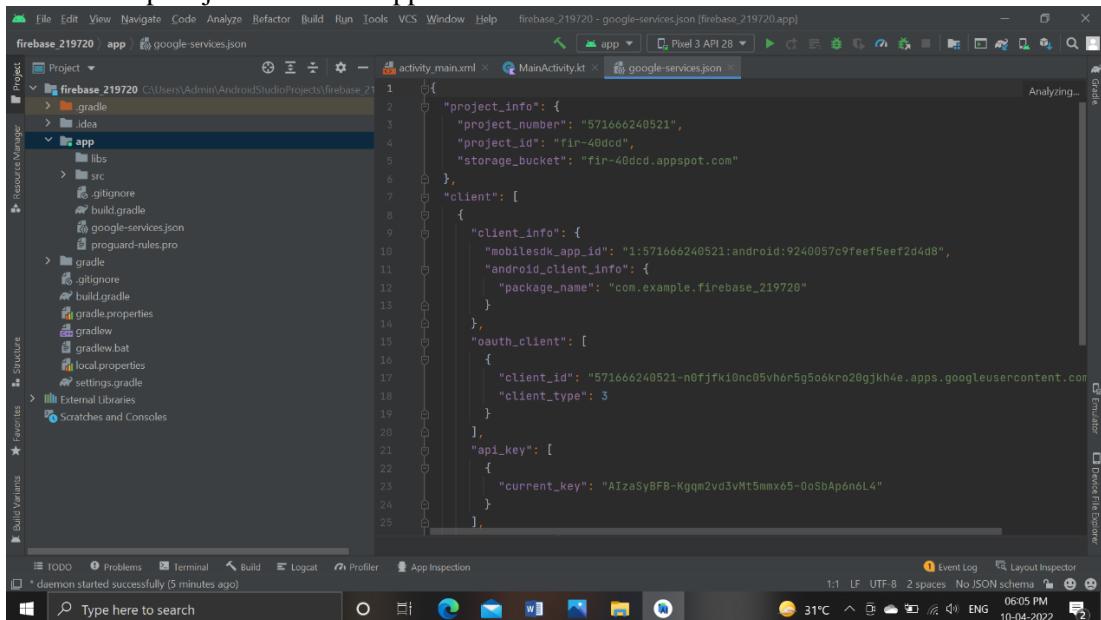
7. Download the google-service.json file



8. Change the Android mode to project mode in the project workspace



9. Paste the copied json file to the app folder



10. Change the project mode to android in the project workspace
11. Add the dependencies in the <>project<> build.gradle file
12. Add the plugin and dependencies in <>module<> build.gradle file and Sync Now

```

firebase_219720 app build.gradle
Gradle files have changed since last project sync. A project sync may be necessary for the IDE to work properly.
Sync Now Ignore these changes
plugins {
    id 'com.android.application'
    id 'kotlin-android'
    id 'com.google.gms.google-services'
}

android {
    compileSdk 32

    defaultConfig {
        applicationId "com.example.firebaseio_219720"
        minSdk 18
        targetSdk 32
        versionCode 1
        versionName "1.0"

        testInstrumentationRunner "androidx.test.runner.AndroidJUnitRunner"
    }

    buildTypes {
        release {
            minifyEnabled false
            proguardFiles getDefaultProguardFile('proguard-android-optimize.txt'), 'proguard-rules.pro'
        }
    }
}

dependencies{
}

```



```

firebase_219720 app build.gradle
Gradle files have changed since last project sync. A project sync may be necessary for the IDE to work properly.
Sync Now Ignore these changes
targetCompatibility JavaVersion.VERSION_1_8
kotlinOptions {
    jvmTarget = '1.8'
}

dependencies {
    implementation 'androidx.core:core-ktx:1.7.0'
    implementation 'androidx.appcompat:appcompat:1.4.1'
    implementation 'com.google.android.material:material:1.5.0'
    implementation 'androidx.constraintlayout:constraintlayout:2.1.3'
    testImplementation 'junit:junit:4.+'
    androidTestImplementation 'androidx.test.ext:junit:1.1.3'
    androidTestImplementation 'androidx.test.espresso:espresso-core:3.4.0'
    implementation platform('com.google.firebase:firebase-bom:29.2.1')
    implementation 'com.google.firebase:firebase-analytics-ktx'
}

```

13. Next button
14. To check the connectivity of firebase and the android project:
15. Tools>Firebase>RealtimeDatabase>save and retrieve data>build the project
16. Connect firebase

```

    plugins {
        id 'com.android.application'
        id 'kotlin-android'
    }

    android {
        compileSdk 32

        defaultConfig {
            applicationId "com.example.firebaseio_219720"
            minSdk 18
            targetSdk 32
            versionCode 1
            versionName "1.0"

            testInstrumentationRunner "androidx.test.runner.AndroidJUnitRunner"
        }

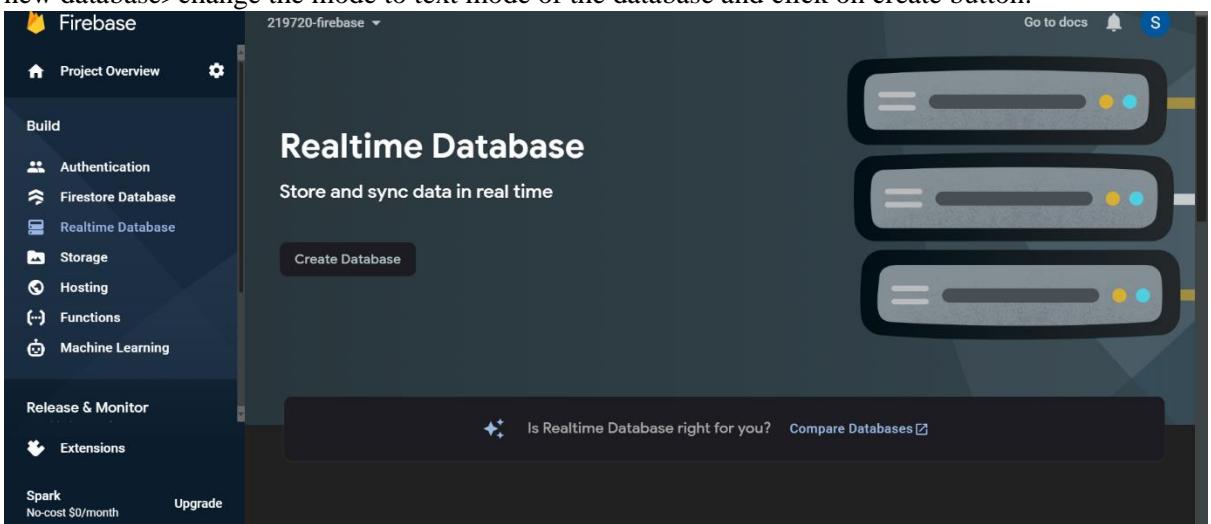
        buildTypes {
            release {
                minifyEnabled false
                proguardFiles getDefaultProguardFile('proguard-android-optimize.txt'), 'proguard-rules.pro'
            }
        }
    }
}

```

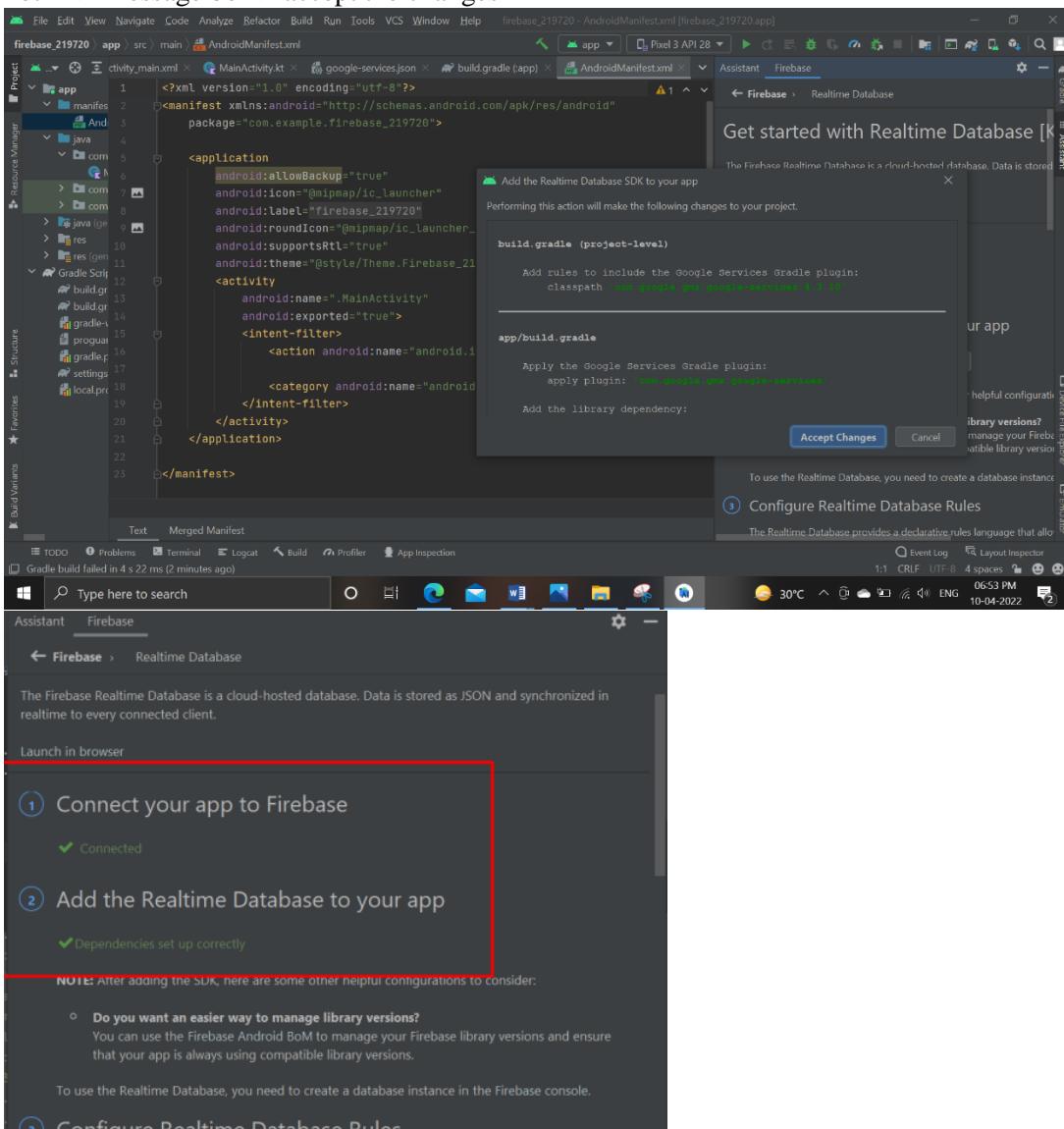
17. Build the apk <<It will show connected message>>

The screenshot shows the Android Studio interface with the build.gradle file open. A modal dialog titled 'Connect to Firebase' is displayed in the foreground. The dialog contains the message 'Your application needs to be built before connecting to Firebase' and two buttons: 'Build' (highlighted with a red arrow) and 'Cancel'. In the background, the Firebase Realtime Database documentation is visible.

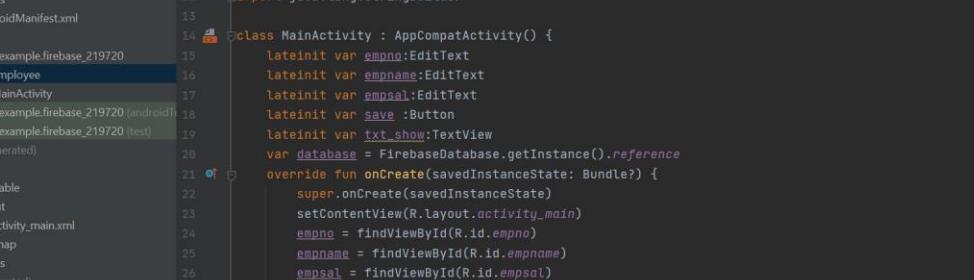
18. Go to web page click on next button and the click on Build > Realtime database > Create the new database>change the mode to text mode of the database and click on create button.



19. Go to android project tools>firebase>realtimedatabase>get started with realtime database with kotlin > Message box > accept the changes



20. Checking the values getting inserted into the database edit the following code in the mainactivity.kt file.



The screenshot shows the Android Studio interface with the following details:

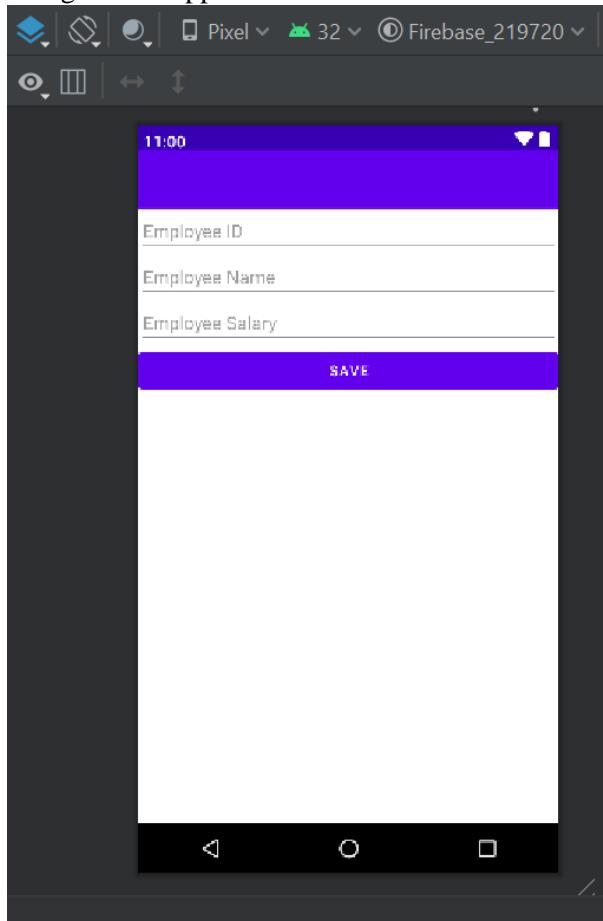
- Project Structure:** The project is named "firebase_219720". It contains an "app" module with "AndroidManifest.xml", "Employee.kt", "MainActivity.kt", and "activity_main.xml".
- Code Editor:** The "MainActivity.kt" file is open, showing Java code for an AppCompatActivity. The code handles button click events to save employee data to a Firebase database.
- Toolbars and Status Bar:** The top bar shows the file path "main.java > com > example > firebase_219720 > MainActivity" and the title "MainActivity.kt [firebase_219720.app]". The bottom status bar shows "30°C" and "10-04-2022".
- Bottom Navigation:** The navigation bar includes icons for TODO, Problems, Terminal, Logcat, Build, Profiler, App Inspection, Event Log, and Layout Inspector.

```
import com.google.firebase.database.ValueEventListener
import java.lang.StringBuilder

class MainActivity : AppCompatActivity() {
    lateinit var empno:EditText
    lateinit var empname:EditText
    lateinit var empsal:EditText
    lateinit var save :Button
    lateinit var txt_show:TextView
    var database = FirebaseDatabase.getInstance().reference
    override fun onCreate(savedInstanceState: Bundle?) {
        super.onCreate(savedInstanceState)
        setContentView(R.layout.activity_main)
        empno = findViewById(R.id.empno)
        empname = findViewById(R.id.empname)
        empsal = findViewById(R.id.empsal)
        save = findViewById(R.id.btn_save)
        txt_show = findViewById(R.id.txt_show)
        save.setOnClickListener { it: View ->
            var empno = empno.text.toString().toInt()
            var empname = empname.text.toString()
            var empsal = empsal.text.toString().toInt()
            database.child(empno.toString()).setValue(Employee(empname,empsal))
        }
    }
    // To get the data from the database
}

class Employee {
    var empno: Int = 0
    var empname: String = ""
    var empsal: Int = 0
}
```

- ## 21. Design of the application



```
<?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"
    android:orientation="vertical">

    <EditText
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:hint="Employee ID"
        android:id="@+id/empno"/>

    <EditText
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:hint="Employee Name"
        android:id="@+id/empname"/>

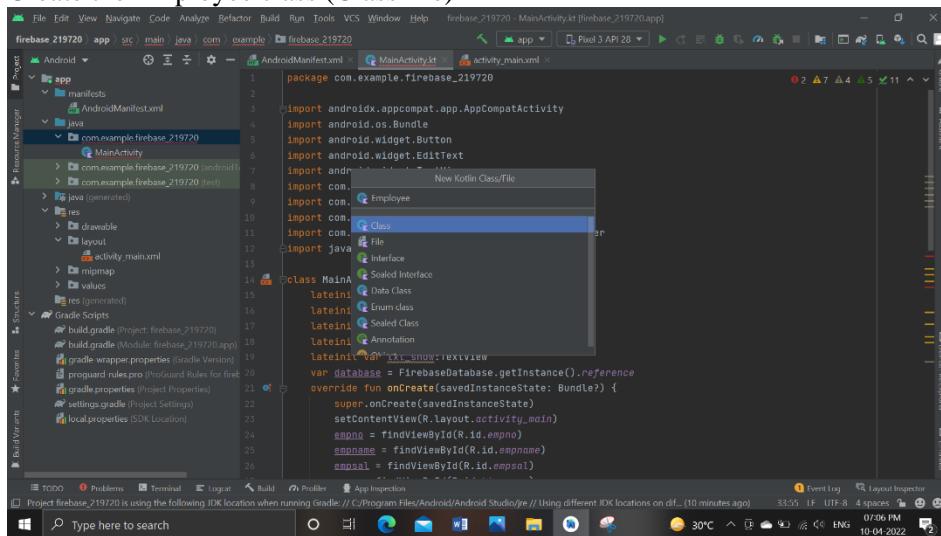
    <EditText
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:hint="Employee Salary"
        android:id="@+id/empsal" />

    <Button
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:text="Save"
        android:id="@+id	btn_save"/>

    <TextView
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:id="@+id/txt_show" />

</LinearLayout>
```

22. Create the Employee class (Class file)



```

package com.example.firebaseio_219720

class Employee {
    var ename = " "
    var esal= 0
    constructor(empname:String,empsal:Int){

        this.ename = empname
        this.esal = empsal
    }
}

```

MainActivity.kt

```

package com.example.firebaseio_219720

import androidx.appcompat.app.AppCompatActivity
import android.os.Bundle
import android.widget.Button
import android.widget.EditText
import android.widget.TextView
import com.google.firebase.database.DataSnapshot
import com.google.firebase.database.DatabaseError
import com.google.firebase.database.FirebaseDatabase
import com.google.firebase.database.ValueEventListener
import java.lang.StringBuilder

class MainActivity : AppCompatActivity() {
    lateinit var empno:EditText
    lateinit var empname:EditText
    lateinit var empsal:EditText
    lateinit var save :Button
    lateinit var txt_show:TextView
    var database = FirebaseDatabase.getInstance().reference
    override fun onCreate(savedInstanceState: Bundle?) {
        super.onCreate(savedInstanceState)
        setContentView(R.layout.activity_main)
        empno = findViewById(R.id.empno)
    }
}

```

```

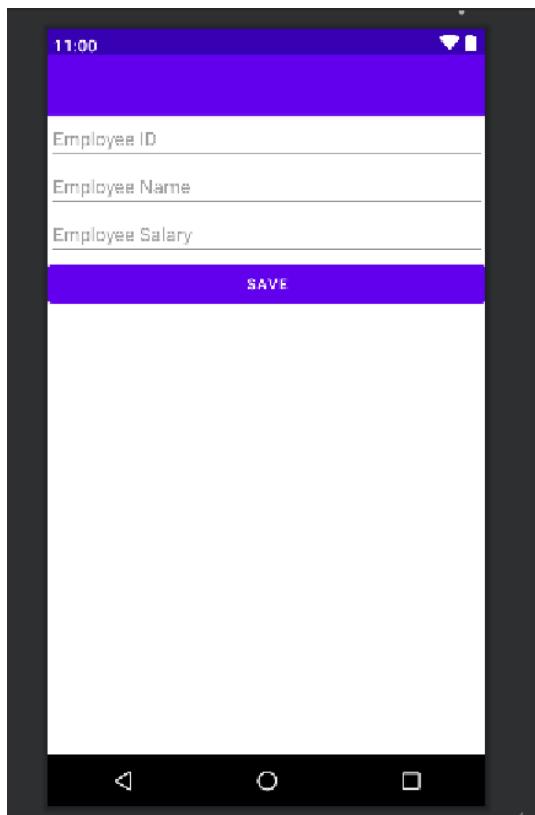
empname = findViewById(R.id.empname)
empsal = findViewById(R.id.empsal)
save = findViewById(R.id.btn_save)
txt_show = findViewById(R.id.txt_show)
save.setOnClickListener {
    var empno = empno.text.toString().toInt()
    var empname = empname.text.toString()
    var empsal = empsal.text.toString().toInt()
    database.child(empno.toString()).setValue(Employee(empname,empsal))
}

// To get the data from the database
var getdata = object :ValueEventListener{
    override fun onDataChange(snapshot: DataSnapshot) {
        var sb = StringBuilder()
        for( i in snapshot.children){
            var ename1 = i.child("ename").getValue()
            var esal = i.child("esal").getValue()
            sb.append("${i.key} $ename1 $esal\n")
        }
        txt_show.setText(sb)
    }

    override fun onCancelled(error: DatabaseError) {

    }
}
database.addValueEventListener(getdata)
database.addListenerForSingleValueEvent(getdata)
}
}

```



Checkout Telegram for Project RAR Files:

https://t.me/STUD_Talks/267

STUD--Talks: Follow us on     for more videos and updates