



DEPARTMENT OF COMPUTER SCIENCE & ENGINEERING

Discover. Learn. Empower.

Experiment No. : 1.1

Student Name: Tanisha Singh	UID: 21BCS9213
Branch: BE CSE	Section: 21BCS_CC_646(B)
Subject Name: Mobile Application Development with Lab	Semester: Sixth
Date of Performance: 18/01/2024	Subject Code: 21CSH-355

Aim: Installation and configuration of Android Studio.

Objective: The objective of installing and configuring Android Studio is to set up a development environment for creating Android applications. The process involves installing the necessary tools and components, configuring the development environment, and preparing the IDE for efficient Android app development.

Input/Apparatus Used:

Input:

Computer: Android Studio is compatible with Windows, macOS, and Linux. Ensure that your computer meets the minimum system requirements for the chosen operating system.

Internet Connection: A reliable internet connection is required to download Android Studio and the necessary SDK components during the installation process.

Apparatus:

Android Studio Installer: Download the Android Studio installer from the official Android Studio website. Choose the appropriate version for your operating system.

Computer Mouse, Keyboard and Monitor/Display: Use a mouse and keyboard to interact with the installation process, configure settings, and navigate through Android Studio. Android Studio requires a monitor or display to visualize the installation process and subsequently to develop and test Android applications.

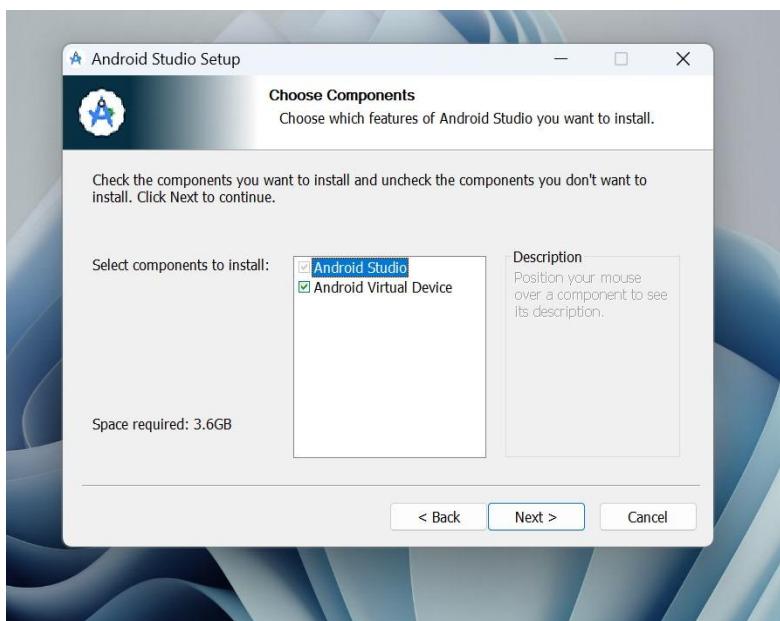
Storage Space: Ensure sufficient free storage space on your computer to accommodate the Android Studio installation and any additional SDK components you may download.

Script and Output:

Step1. Launch AndroidStudio.exe



Step2. Check the components, which are required to create applications

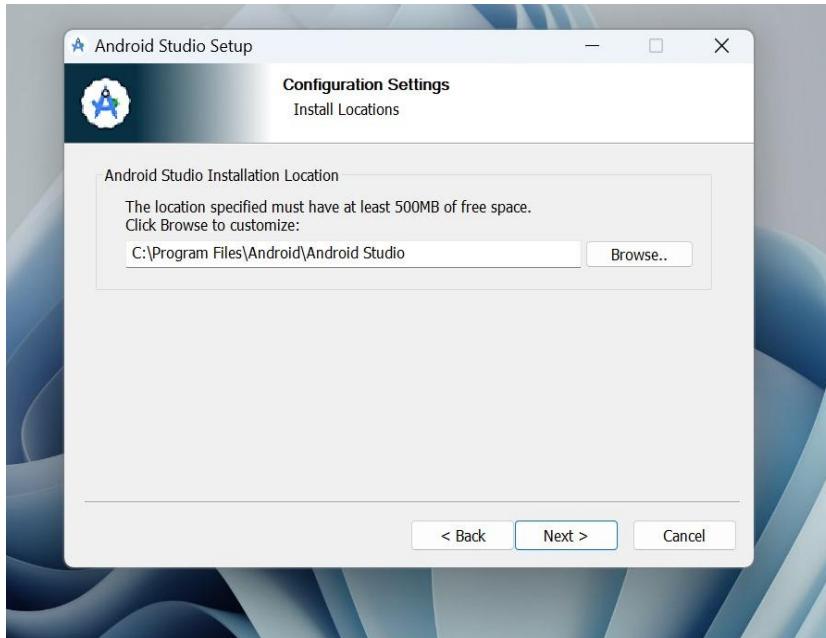




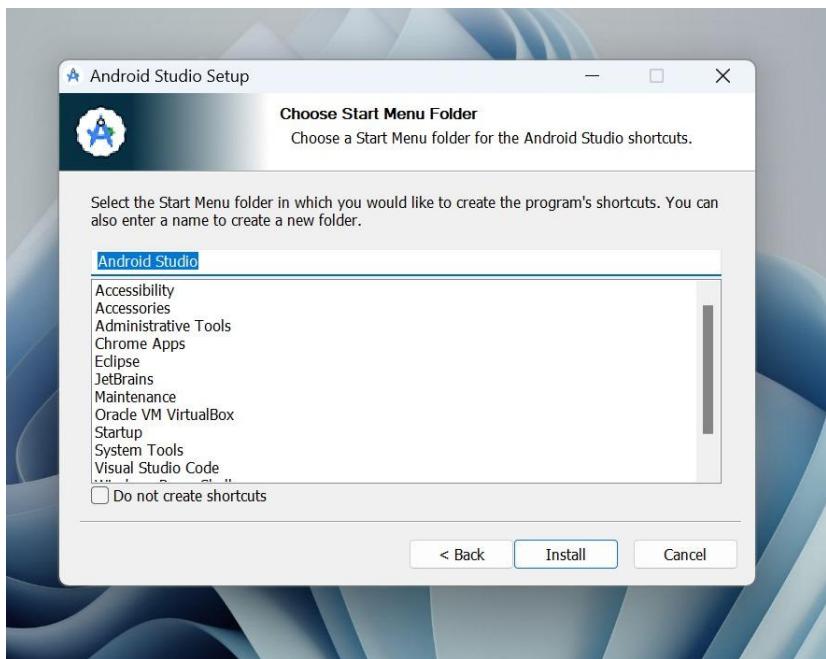
DEPARTMENT OF COMPUTER SCIENCE & ENGINEERING

Discover. Learn. Empower.

Step3. Check the path for storing



Step4. Choose Start Menu folder.

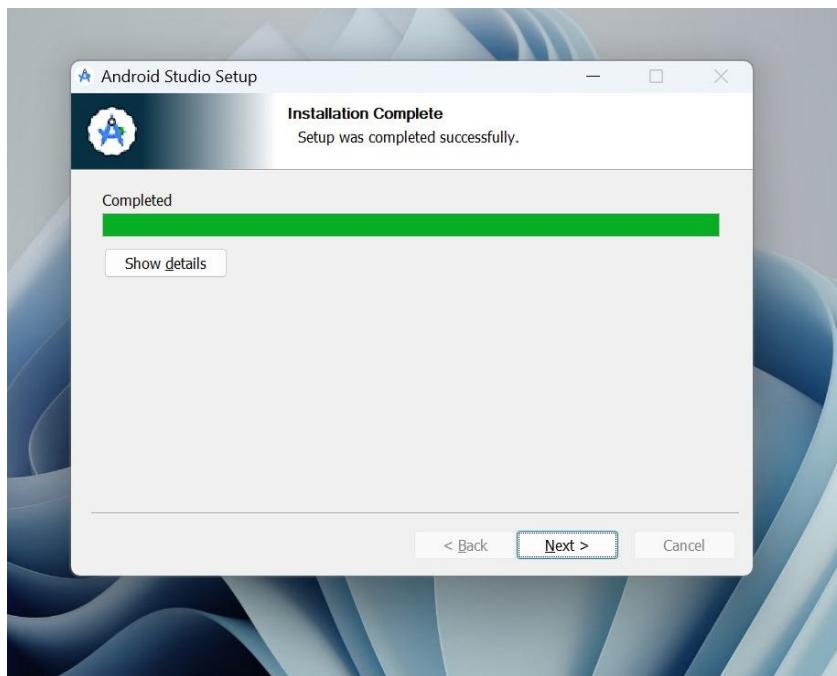




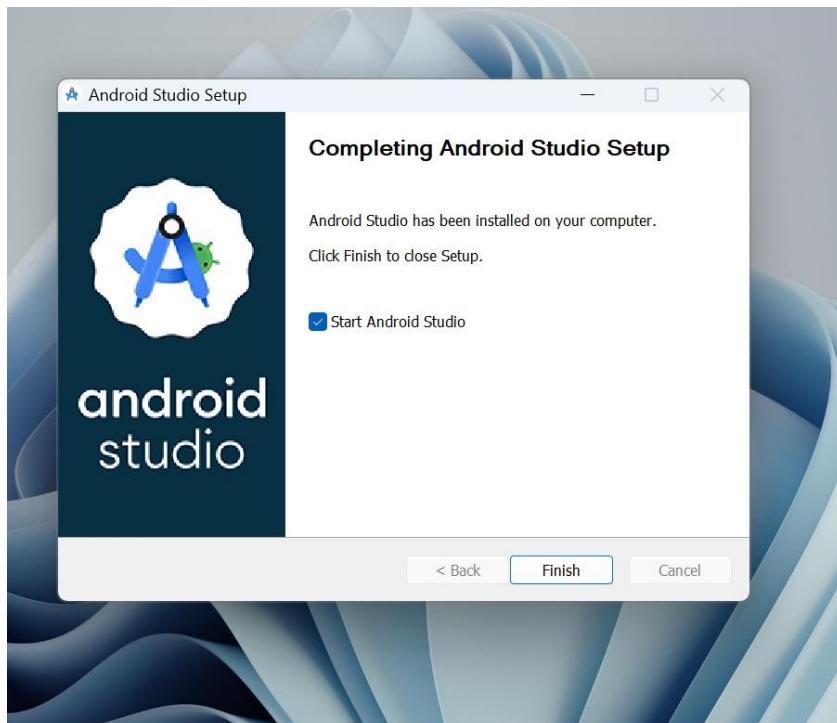
DEPARTMENT OF COMPUTER SCIENCE & ENGINEERING

Discover. Learn. Empower.

Step5. Installation process



Step6. Click Finish to finish the installation.

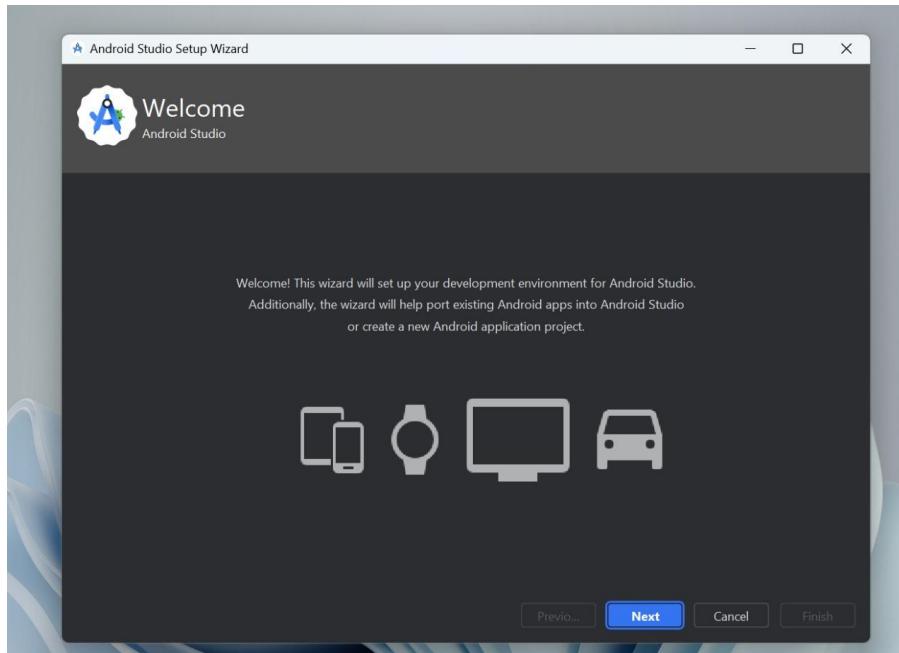




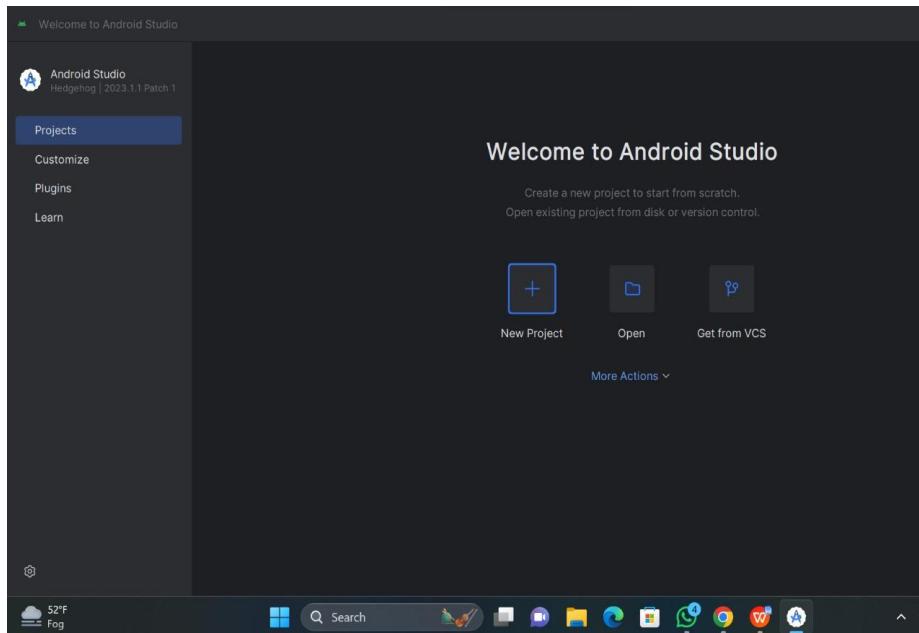
DEPARTMENT OF COMPUTER SCIENCE & ENGINEERING

Discover. Learn. Empower.

Step7. Start the application.



Step8. Start a new project in it.





DEPARTMENT OF COMPUTER SCIENCE & ENGINEERING

Discover. Learn. Empower.

Result:

Successful setup of Android development environment.

Learning outcomes:

- i. Learnt about the use of Android Studio.
- ii. Learnt about the set up required for the development of environment for creating Android applications.
- iii. Learnt how to create new projects using Android Studio.



DEPARTMENT OF COMPUTER SCIENCE & ENGINEERING

Discover. Learn. Empower.

Experiment No. : 1.2

Student Name: Tanisha Singh	UID: 21BCS9213
Branch: BE CSE	Section: 21BCS_CC_646(B)
Subject Name: Mobile Application Development with Lab	Semester: Sixth
Date of Performance: 18/01/2024	Subject Code: 21CSH-355

Aim: Create an application that takes the name from a "Text Box" and shows a "Hello" message along with the name entered in the "Text Box" when the user clicks the "OK" button.

Objective: Setting up Android Studio and configuring a basic Android project. Compiling and running a simple Android application on an emulator or a physical device.

Input/Apparatus Used:

Input:

Computer: Android Studio is compatible with Windows, macOS, and Linux. Ensure that your computer meets the minimum system requirements for the chosen operating system.

Internet Connection: A reliable internet connection is required to download Android Studio and the necessary SDK components during the installation process.

Apparatus:

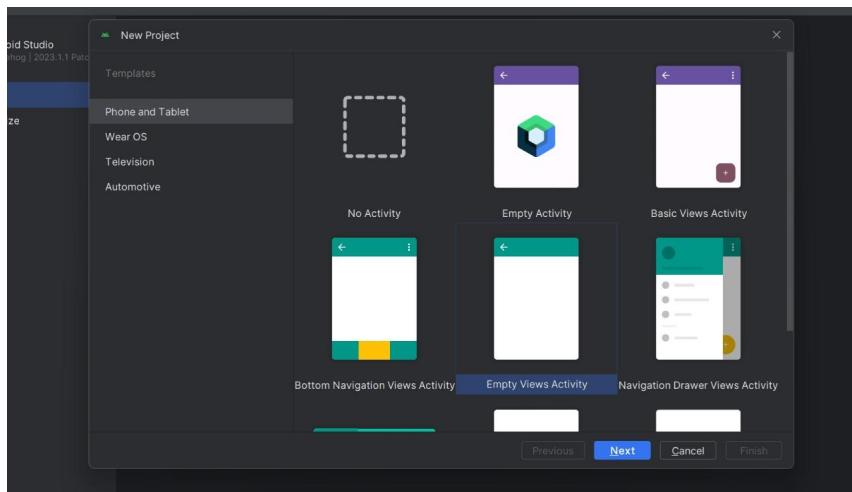
Android Studio Installer: Download the Android Studio installer from the official Android Studio website. Choose the appropriate version for your operating system.

Computer Mouse, Keyboard and Monitor/Display: Use a mouse and keyboard to interact with the installation process, configure settings, and navigate through Android Studio. Android Studio requires a monitor or display to visualize the installation process and subsequently to develop and test Android applications.

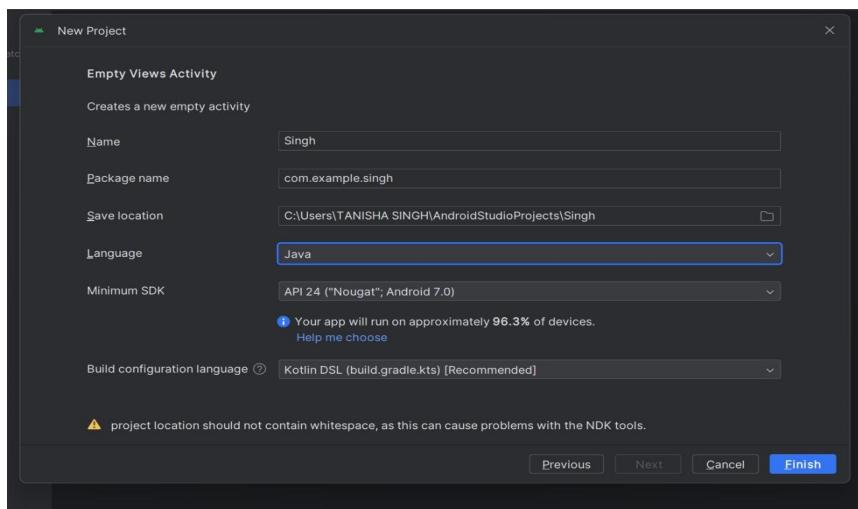
Storage Space: Ensure sufficient free storage space on your computer to accommodate the Android Studio installation and any additional SDK components you may download.

Script and Output:

Step1. Click on the **New Project** in the Android studio and Select **Empty Views Activity** option.



Step2. Modify the name and dependencies of the application with language preference as **Java**.

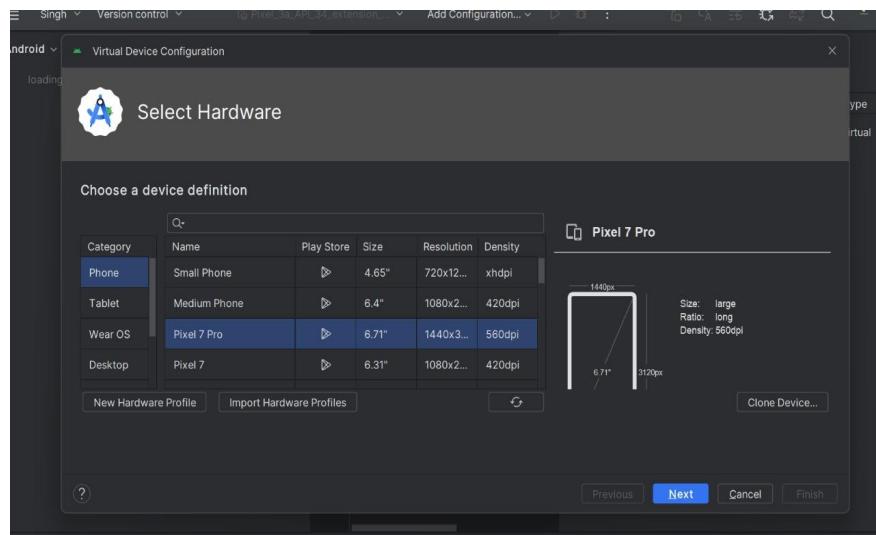




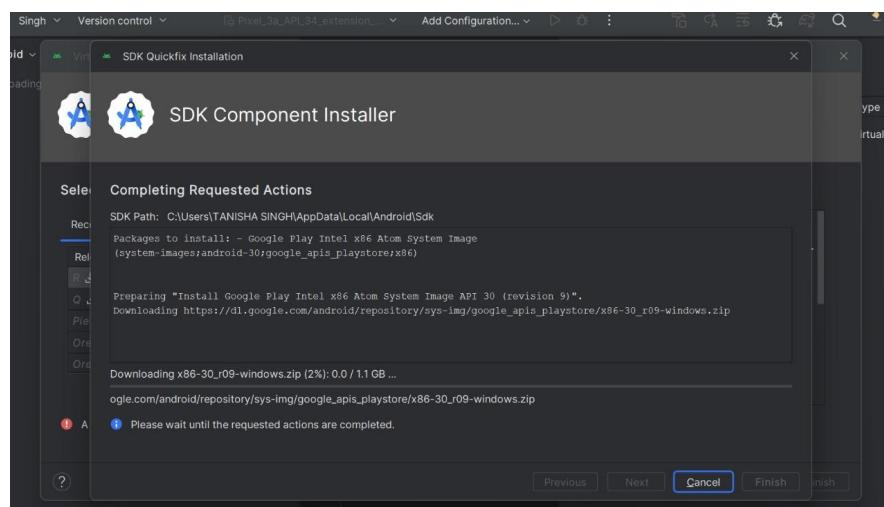
DEPARTMENT OF COMPUTER SCIENCE & ENGINEERING

Discover. Learn. Empower.

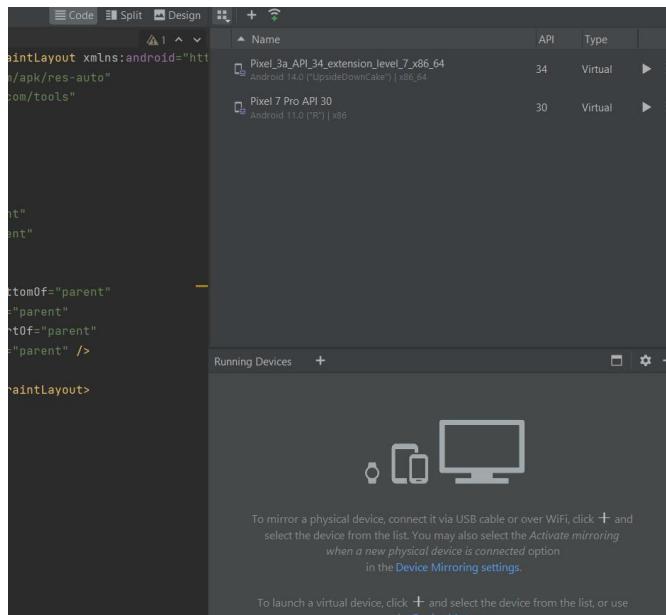
Step3. Go to **Device Manager** and select the required **hardware** components.



Step4. completing **SDK Component Installer**



Step5. Select the device that you created.



Step6. - Go to **activity_main.xml** File and add text field saying “HELLO WORLD” and other “text”.

```

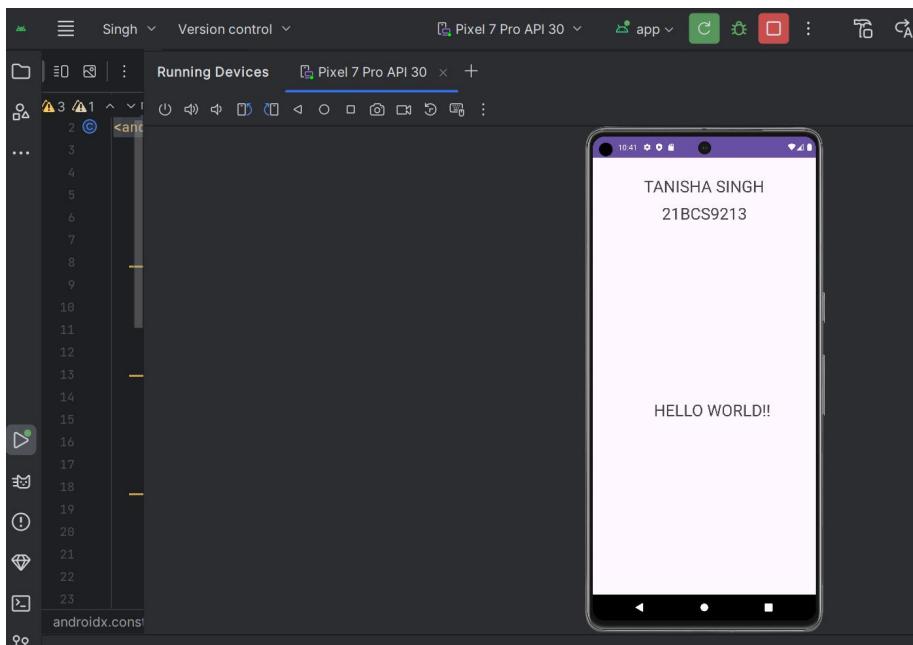
<?xml version="1.0" encoding="utf-8"?>
<androidx.constraintlayout.widget.ConstraintLayout xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:app="http://schemas.android.com/apk/res-auto"
    xmlns:tools="http://schemas.android.com/tools"
    android:layout_width="match_parent"
    android:layout_height="match_parent"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

    <TextView
        android:id="@+id/textView3"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_marginTop="32dp"
        android:text="TANISHA SINGH"
        android:textSize="30sp"
        app:layout_constraintEnd_toEndOf="parent"
        app:layout_constraintHorizontal_bias="0.498"
        app:layout_constraintStart_toStartOf="parent"
        app:layout_constraintTop_toTopOf="parent" />

<androidx.constraintlayout.widget.ConstraintLayout

```

Step7. Run the code in Virtual Device based on compatibility of APIs, the “HELLO WORLD” and other “text”will be displayed on screen.



Result:

Successful setup of Android development environment.

Learning outcomes:

- i. Learnt how to create new projects using Android Studio.
- ii. Learnt how to compile and run a simple Android application on an emulator.
- iii. Learnt about the set up required for the development of environment for creating Android applications.
- iv. Learnt about how to code in android studio and print texts.



DEPARTMENT OF COMPUTER SCIENCE & ENGINEERING

Discover. Learn. Empower.



DEPARTMENT OF COMPUTER SCIENCE & ENGINEERING

Discover. Learn. Empower.

Experiment No. : 1.3

Student Name: Tanisha Singh	UID: 21BCS9213
Branch: BE CSE	Section: 21BCS_CC_646(B)
Subject Name: Mobile Application Development with Lab	Semester: Sixth
Date of Performance: 1/02/2024	Subject Code: 21CSH-355

Aim: Create an Android-based application using widgets. It can be embedded in other applications (such as the home screen) and receive periodic updates.

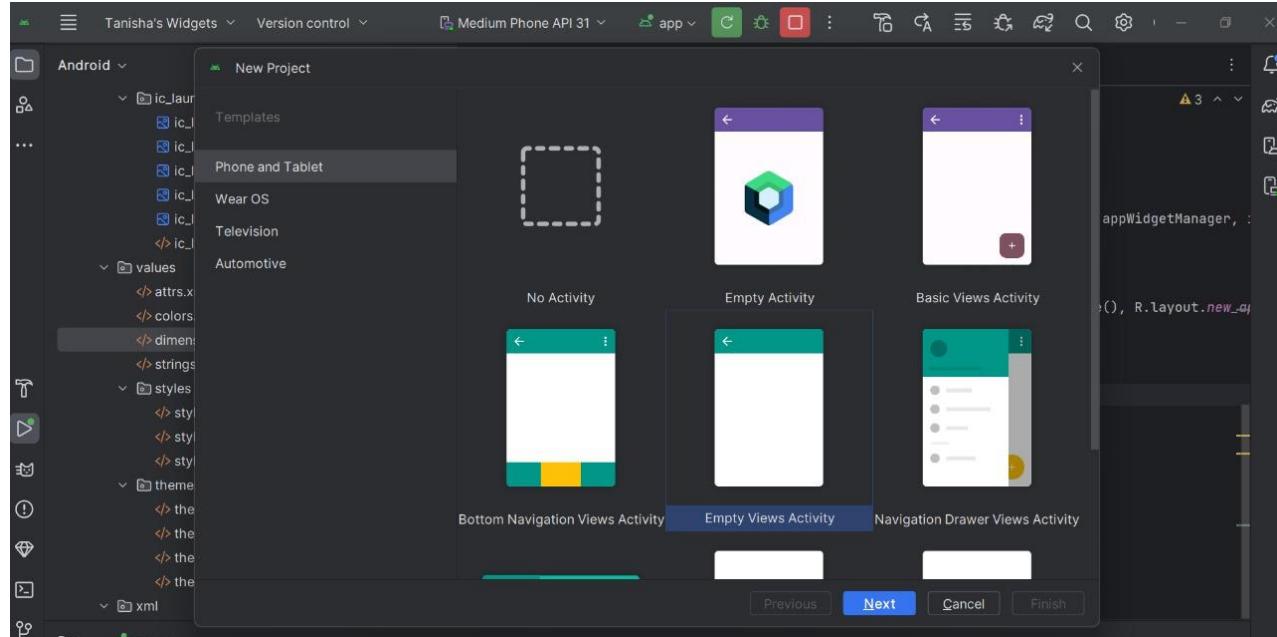
Objective: The objective is to develop an Android application featuring customizable widgets that can be embedded in diverse applications, including the home screen. The application should seamlessly receive periodic updates, ensuring real-time information and enhanced user interaction.

Input/Apparatus Used:

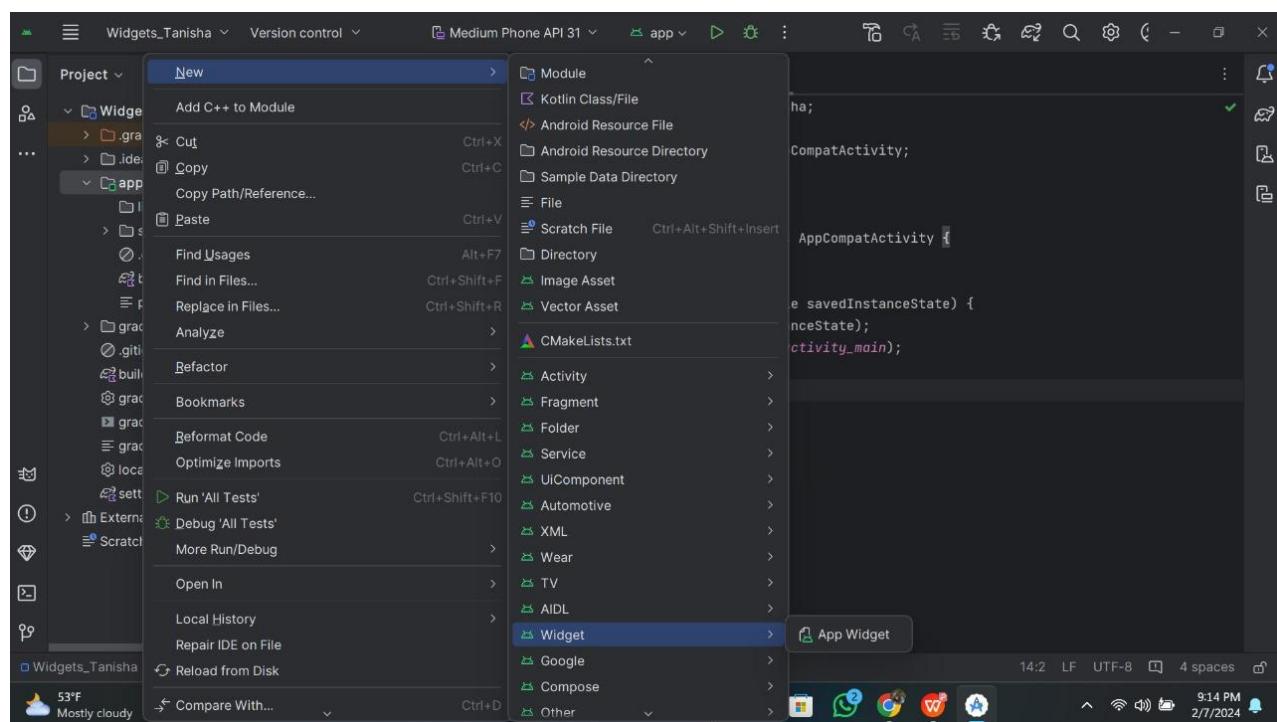
- Integrated Development Environment (IDE):** Android Studio: Official IDE for Android development.
- Programming Language:** Kotlin or Java: Preferred languages for Android development.
- Widget Components:** App Widgets API: Android's API for creating widgets.
- User Interface (UI) Design ML:** Defines widget layout and appearance.
- Graphics and Design:-Graphics Software:** (e.g., Photoshop, Sketch) for visual assets.
- Data Handling: JSON or XML Parsing:** Handles data from servers.
- Networking:** HTTP Libraries (e.g., Retrofit, Volley): Fetches data from servers.
- Background Services:** Android Services: Manages periodic updates in the background.
- Testing Tools:** Android Emulator: Virtual testing on various devices.
- Physical Devices:** Real device testing for compatibility.
- Documentation Tools documentation Software:** Creates user and developer documentation.

Script and Output:

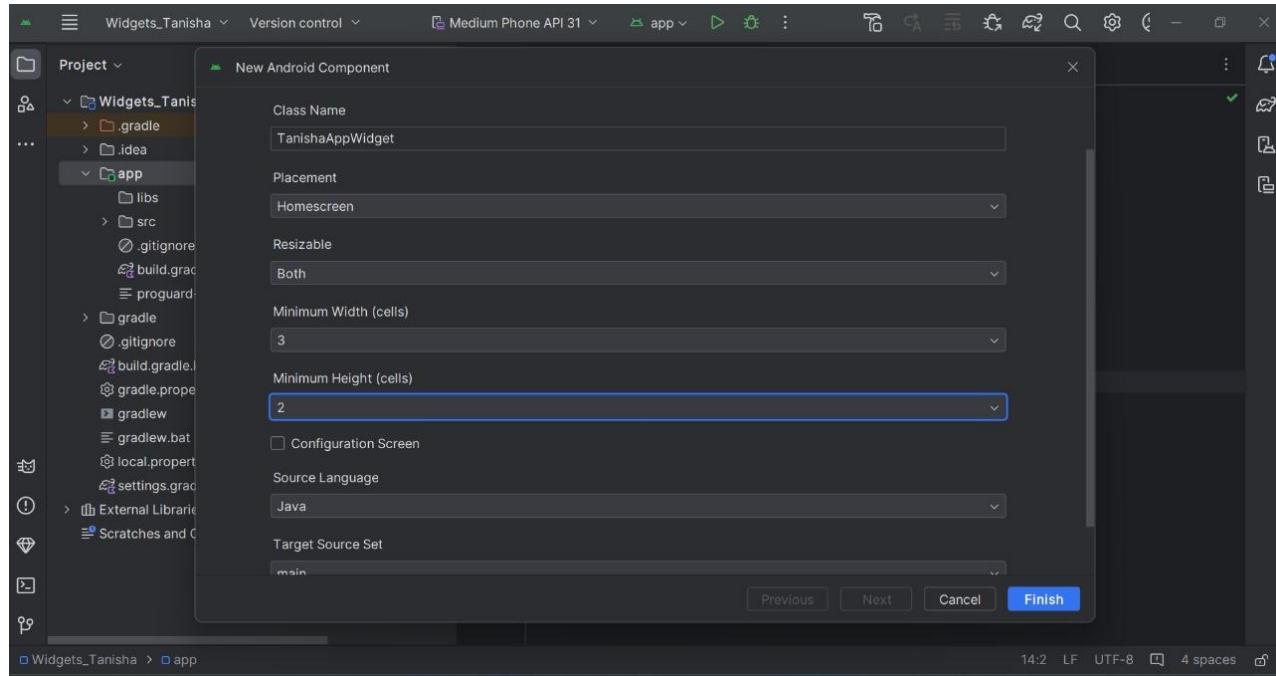
Step1. Click on the **New Project** in the Android studio and Select **Empty Views Activity** option.



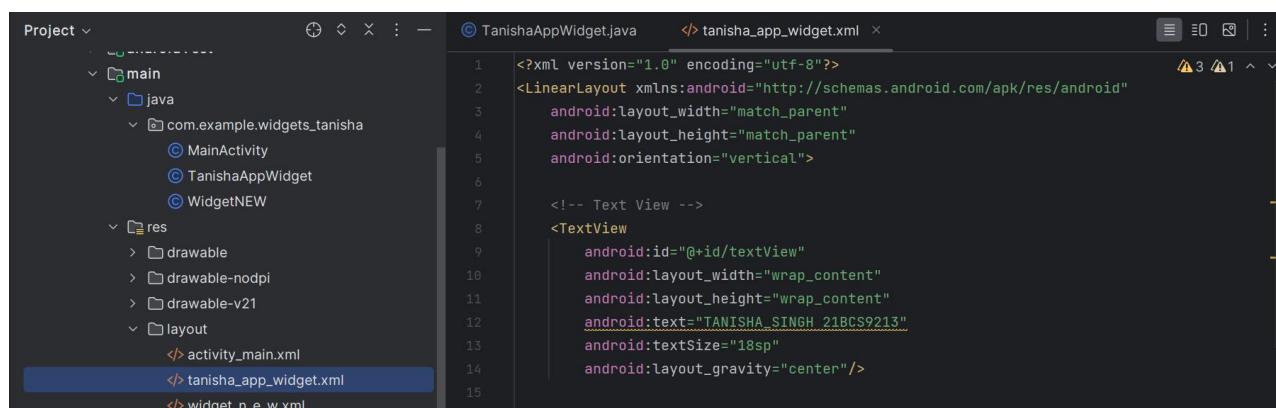
Step2. Add the **App Widget** to the Project from **New** option in **app**.

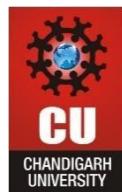


Step3. Select all the configurations you want.



Step4. Open main Java and XML file by double clicking it.





DEPARTMENT OF COMPUTER SCIENCE & ENGINEERING

Discover. Learn. Empower.

Step5. Write **Java code** as given below.

```
④ TanishaAppWidget.java × </> tanisha_app_widget.xml

1 package com.example.widget;
2
3 import com.example.widgets_tanisha.R;
4 import android.appwidget.AppWidgetManager; import android.appwidget.AppWidgetProvider; import android.content.Context;
5 import android.widget.RemoteViews;
6
7 public class TanishaAppWidget extends AppWidgetProvider {
8
9     @Override
10    public void onUpdate(Context context, AppWidgetManager appWidgetManager, int[] appWidgetIds) {
11        // Perform your widget update operations here
12
13        for (int appWidgetId : appWidgetIds) { updateAppWidget(context, appWidgetManager,
14            appWidgetId);
15        }
16    }
17    @
18    private void updateAppWidget(Context context, AppWidgetManager appWidgetManager, int appWidgetId) {
19        // Construct the RemoteViews object
20        RemoteViews views = new RemoteViews(context.getPackageName(), R.layout.tanisha_app_widget);
21        // You can add additional actions or update views here
22
23        // Update the widget
24        appWidgetManager.updateAppWidget(appWidgetId, views);
25    }
26}
```

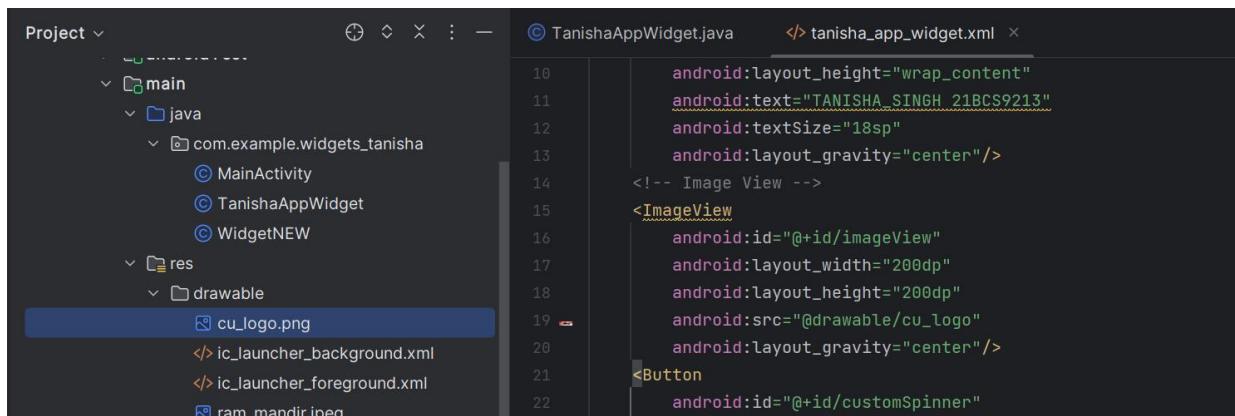
Step6. - Go to **tanisha_app_widget.xml** and write the code as given below.



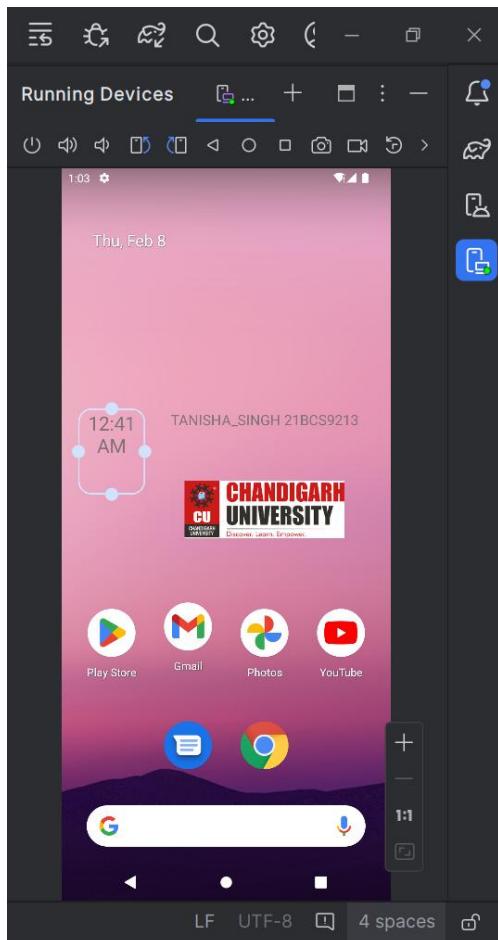
DEPARTMENT OF COMPUTER SCIENCE & ENGINEERING

Discover. Learn. Empower.

Step7. Go to drawable under res and drop the png file for getting image.



Output:





DEPARTMENT OF COMPUTER SCIENCE & ENGINEERING

Discover. Learn. Empower.

Result:

Successful development of Android based application using Widgets which can be embedded on home screen and receive periodic updates.

Learning outcomes:

- i. Understanding the basics of widget development in Android.
- ii. Setting up development environment for Android app development.
- iii. Learnt how to create and update widgets, handle user interactions, and design them to be visually appealing and functional.
- iv. Designing a user-friendly interface for the widget that provides relevant information and easy interaction.



DEPARTMENT OF COMPUTER SCIENCE & ENGINEERING

Discover. Learn. Empower.

Experiment No. : 2.1

Student Name: Tanisha Singh	UID: 21BCS9213
Branch: BE CSE	Section: 21BCS_CC_646(B)
Subject Name: Mobile Application Development with Lab	Semester: Sixth
Date of Performance: 8/02/2024	Subject Code: 21CSH-355

Aim: Create an Android app that uses Intent with button to create a page and passes values from one activity to another.

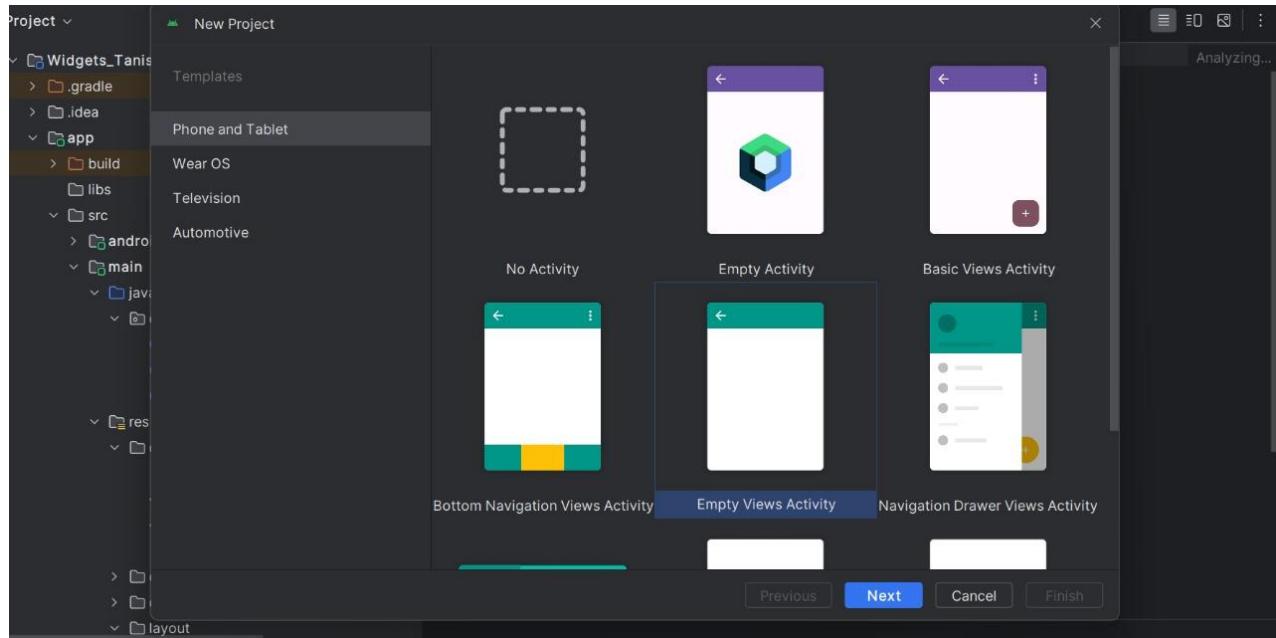
Objective: The objective of an Android app that uses Intent with a button to create a page and passes values from one activity to another could be to demonstrate and implement a simple data communication flow between different activities within an Android application. This type of app is commonly used to understand and showcase the concept of passing data between different screens or pages in Android.

Input/Apparatus Used:

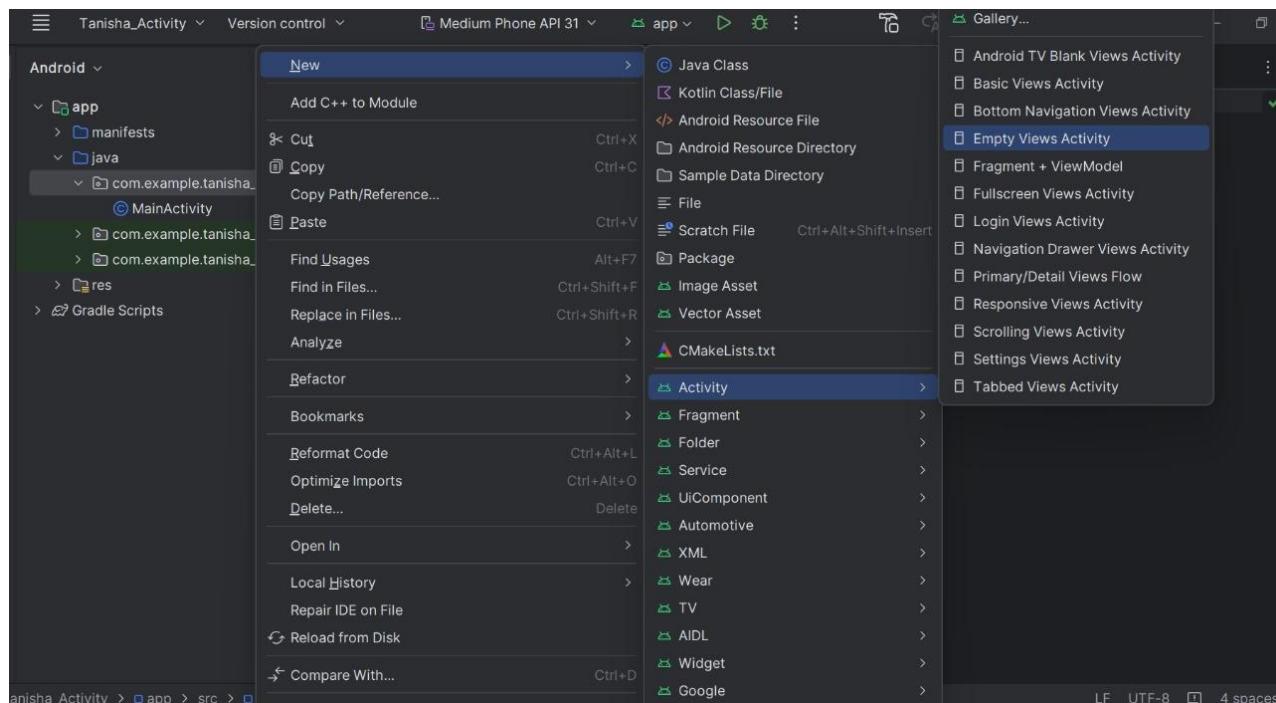
- 1. Integrated Development Environment (IDE) Android Studio:** The official IDE for Android development. Download and install Android Studio from the official website: [Android Studio](https://developer.android.com/studio).
- 2. Android SDK:** The Android Software Development Kit (SDK) is essential for developing Android applications. Android Studio usually comes bundled with the SDK, but you may need to update it through the SDK Manager within Android Studio.
- 3. Java Development Kit (JDK):** Android apps are primarily written in Java or Kotlin. Make sure you have the Java Development Kit installed. Android Studio supports JDK. You can download it from the Oracle website: [Java SE Downloads](https://www.oracle.com/java/technologies/javase/javase-jdk-downloads.html).
- 4. Android Virtual Device (AVD) or Physical Android Device:** You need a device to test your Android application. You can use an emulator (AVD) that comes with Android Studio or a physical Android device connected to your computer.

Script and Output:

Step1. Click on the New Project in the Android studio and Select **Empty Views Activity** option.



Step2. Create the New Activity (Empty views Activity) from New option in app.

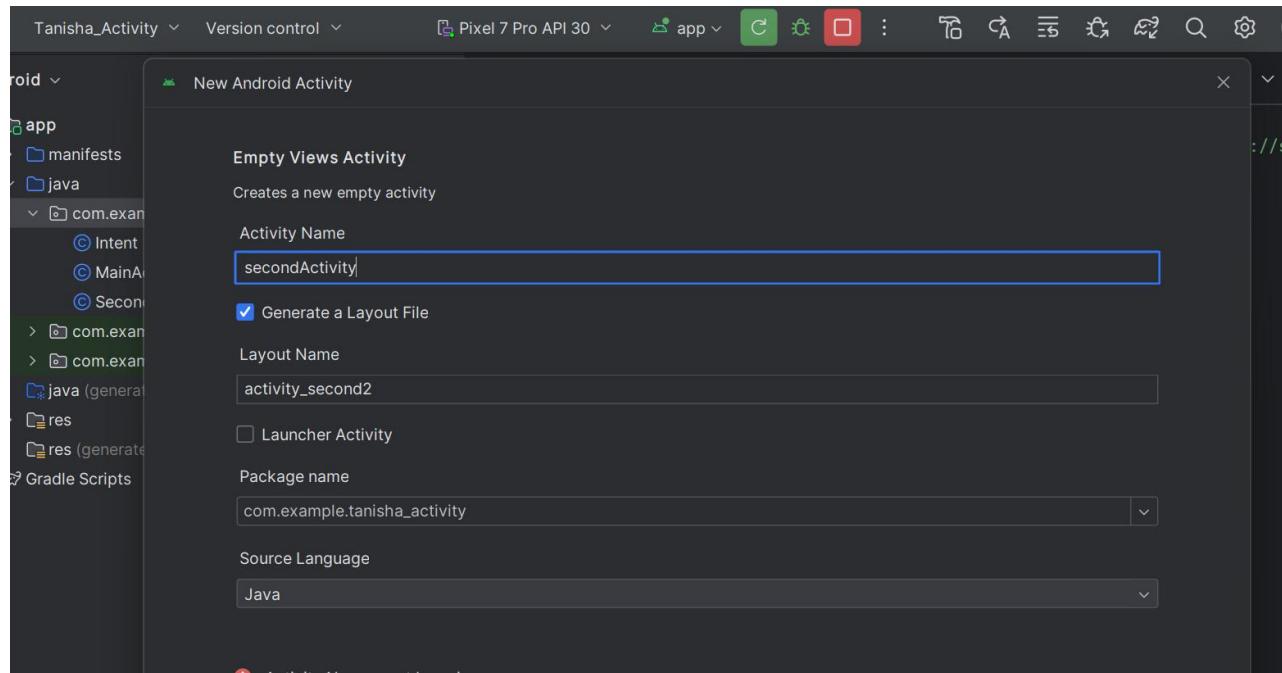




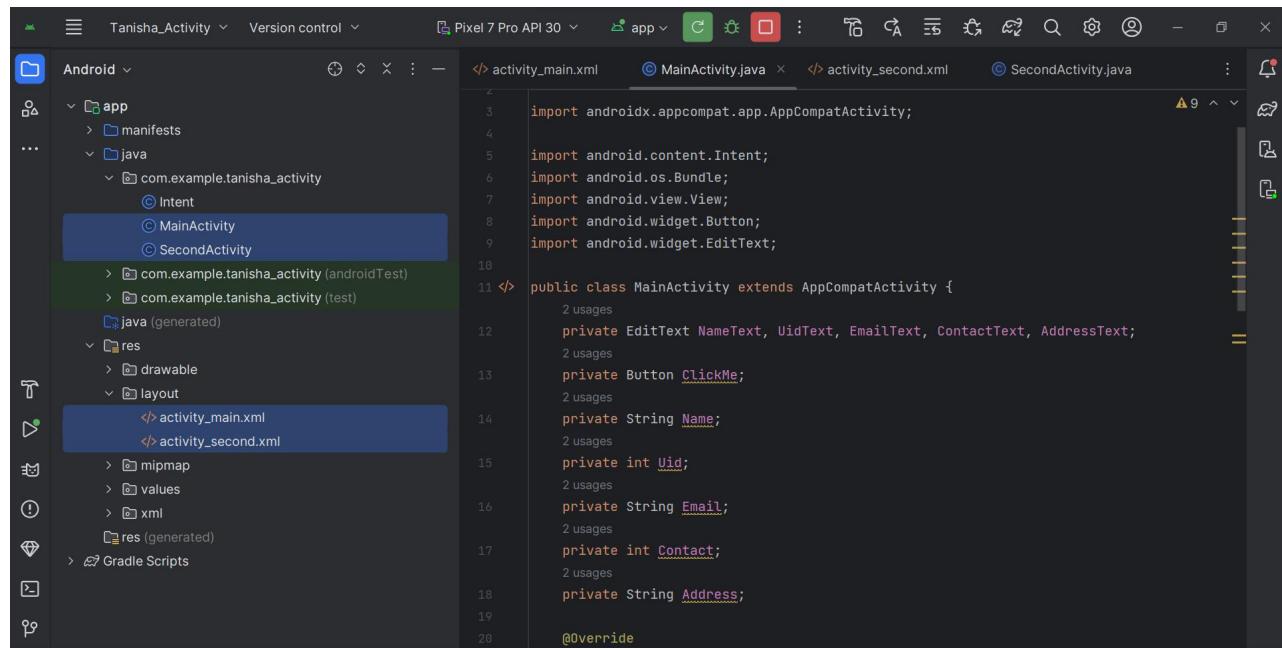
DEPARTMENT OF COMPUTER SCIENCE & ENGINEERING

Discover. Learn. Empower.

Step3. Name the Activity as you want.



Step4. Open main Java and XML file by double clicking it.





DEPARTMENT OF COMPUTER SCIENCE & ENGINEERING

Discover. Learn. Empower.

Step5. Write Java code as given below under **MainActivity.java**.

```
</> activity_main.xml      </> MainActivity.java </> activity_second.xml      </> SecondActivity.java

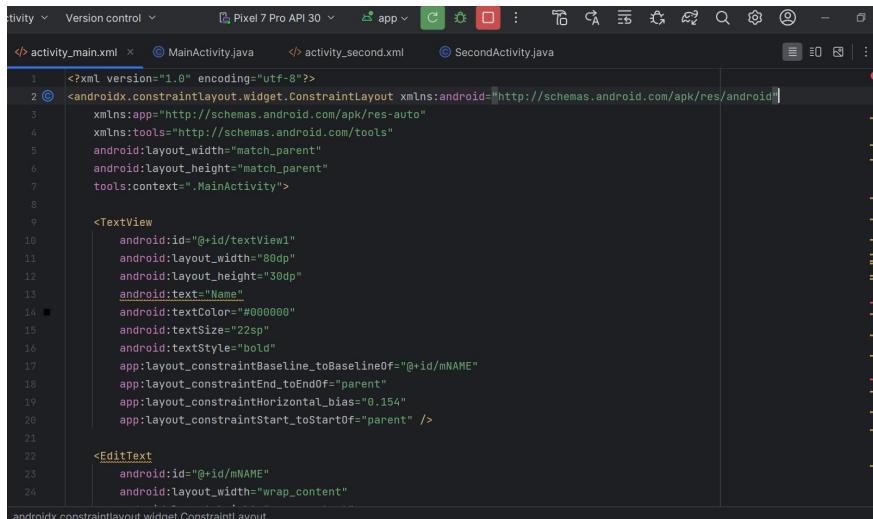
1 package com.example.tanisha_activity;
2
3 import androidx.appcompat.app.AppCompatActivity;
4
5 import android.content.Intent;
6 import android.os.Bundle;
7 import android.view.View;
8 import android.widget.Button;
9 import android.widget.EditText;
10
11 </> public class MainActivity extends AppCompatActivity {
12
13     private EditText NameText, UidText, EmailText, ContactText, AddressText;
14
15     private Button ClickMe;
16
17     private String Name;
18
19     private int Uid;
20
21     private String Email;
22
23     private int Contact;
24
25     private String Address;
26
27
28     @Override
29     protected void onCreate(Bundle savedInstanceState) {
30
31         super.onCreate(savedInstanceState);
32         setContentView(R.layout.activity_main);
33         NameText = findViewById(R.id.mNAME);
34         UidText = findViewById(R.id.mUID);
35         EmailText = findViewById(R.id.mEMAIL);
36         ContactText = findViewById(R.id.mCONTACT);
37         AddressText = findViewById(R.id.mADDRESS);
38         ClickMe = findViewById(R.id.button);
39         ClickMe.setOnClickListener(new View.OnClickListener() {
40
41             @Override
42             public void onClick(View v) {
43                 sendData();
44             }
45         });
46     }
47
48     public void sendData()
49     {
50
51         Name = NameText.getText().toString().trim();
52         Uid= Integer.parseInt(UidText.getText().toString().trim());
53         Email = EmailText.getText().toString().trim();
54         Contact=Integer.parseInt(ContactText.getText().toString().trim());
55         Address = AddressText.getText().toString().trim();
56         Intent i = new Intent( packageContext: MainActivity.this,SecondActivity.class);
57
58         i.putExtra(SecondActivity.NAME, Name);
59         i.putExtra(SecondActivity.UID, Uid);
60         i.putExtra(SecondActivity.EMAIL, Email);
61         i.putExtra(SecondActivity.CONTACT, Contact);
62         i.putExtra(SecondActivity.ADDRESS, Address);
63
64         startActivity(i);
65     }
66
67 }
```



DEPARTMENT OF COMPUTER SCIENCE & ENGINEERING

Discover. Learn. Empower.

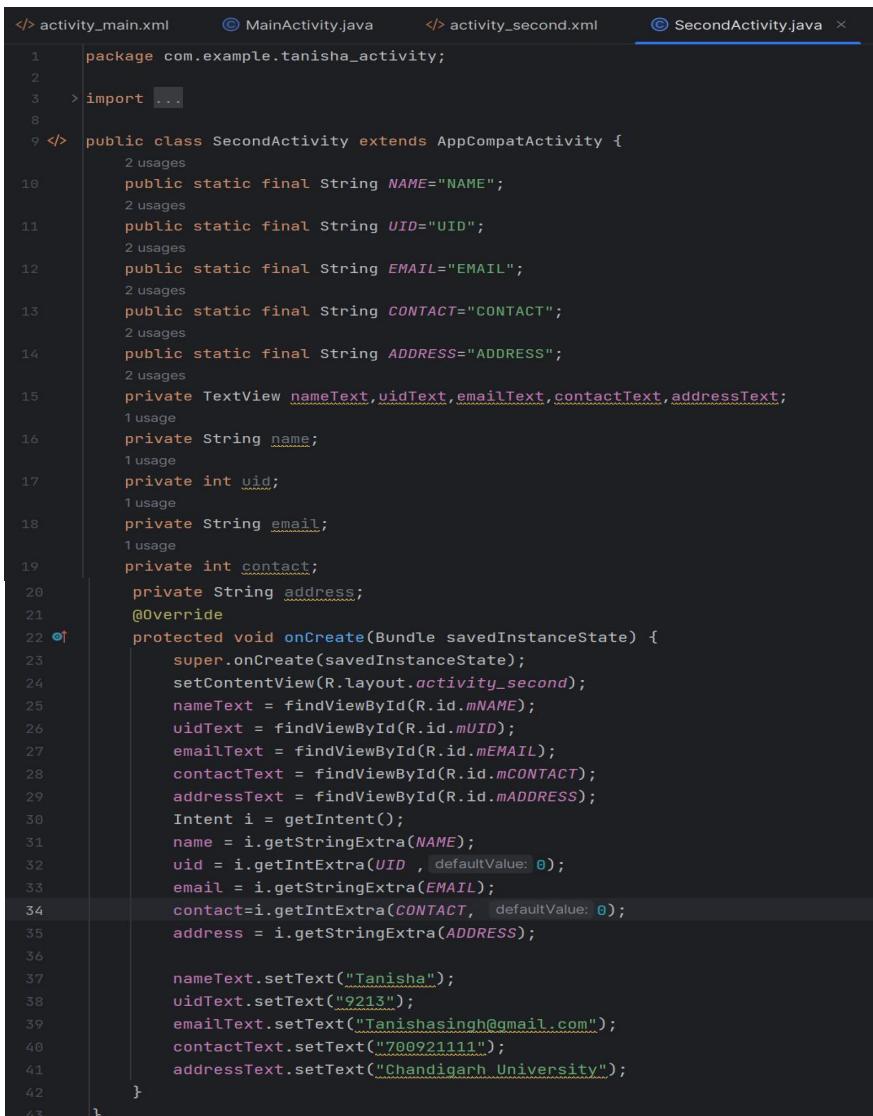
Step6. - Go to **activity_main.xml** and write the code as given below.



```
<?xml version="1.0" encoding="utf-8"?>
<androidx.constraintlayout.widget.ConstraintLayout xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:app="http://schemas.android.com/apk/res-auto"
    xmlns:tools="http://schemas.android.com/tools"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    tools:context=".MainActivity">
    <TextView
        android:id="@+id/textView1"
        android:layout_width="80dp"
        android:layout_height="30dp"
        android:text="Name"
        android:textColor="#000000"
        android:textSize="22sp"
        android:textStyle="bold"
        app:layout_constraintBaseline_toBaselineOf="@+id/mNAME"
        app:layout_constraintEnd_toEndOf="parent"
        app:layout_constraintHorizontal_bias="0.154"
        app:layout_constraintStart_toStartOf="parent" />
    <EditText
        android:id="@+id/mNAME"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"/>

```

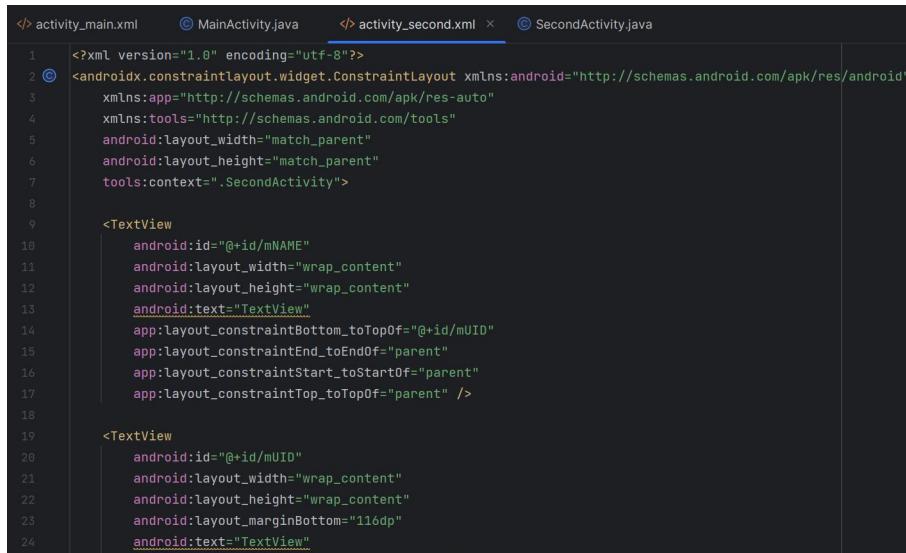
Step7. Go to **SecondActivity.java** and write the code as given below



```
package com.example.tanisha_activity;
import ...
public class SecondActivity extends AppCompatActivity {
    public static final String NAME="NAME";
    public static final String UID="UID";
    public static final String EMAIL="EMAIL";
    public static final String CONTACT="CONTACT";
    public static final String ADDRESS="ADDRESS";
    private TextView nameText,uidText,emailText,contactText,addressText;
    private String name;
    private int uid;
    private String email;
    private int contact;
    private String address;
    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_second);
        nameText = findViewById(R.id.mNAME);
        uidText = findViewById(R.id.mUID);
        emailText = findViewById(R.id.mEMAIL);
        contactText = findViewById(R.id.mCONTACT);
        addressText = findViewById(R.id.mADDRESS);
        Intent i = getIntent();
        name = i.getStringExtra(NAME);
        uid = i.getIntExtra(UID , defaultValue: 0);
        email = i.getStringExtra(EMAIL);
        contact=i.getIntExtra(CONTACT, defaultValue: 0);
        address = i.getStringExtra(ADDRESS);

        nameText.setText("Tanisha");
        uidText.setText("9213");
        emailText.setText("Tanishasingh@gmail.com");
        contactText.setText("700921111");
        addressText.setText("Chandigarh University");
    }
}
```

Step 8. Go to **activity_second.xml** and write the code as given.



```

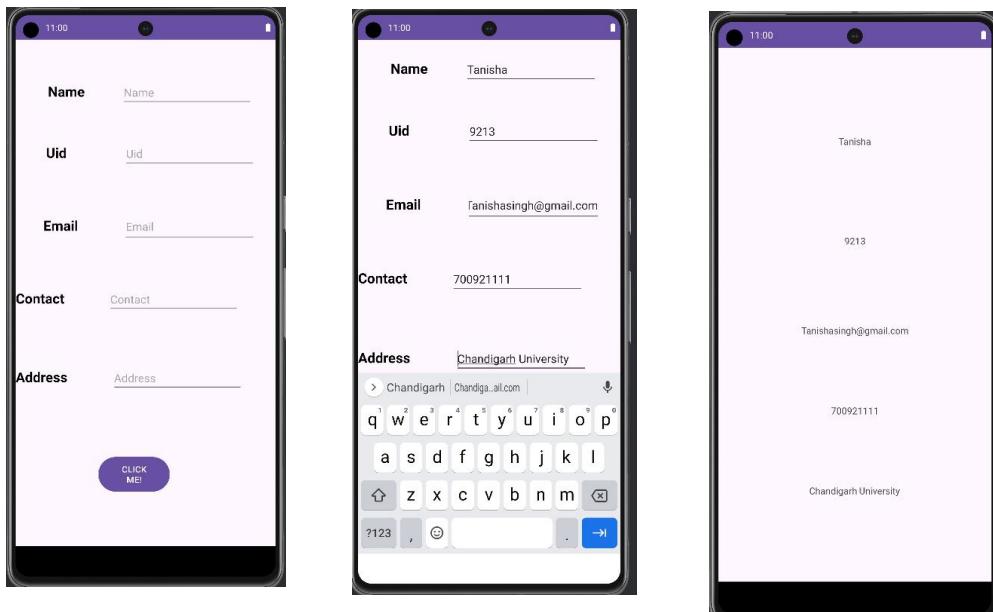
<?xml version="1.0" encoding="utf-8"?>
<androidx.constraintlayout.widget.ConstraintLayout xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:app="http://schemas.android.com/apk/res-auto"
    xmlns:tools="http://schemas.android.com/tools"
    android:layout_width="match_parent"
    android:layout_height="match_parent"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

    <TextView
        android:id="@+id/mNAME"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="TextView"
        app:layout_constraintBottom_toTopOf="@+id/mUID"
        app:layout_constraintEnd_toEndOf="parent"
        app:layout_constraintStart_toStartOf="parent"
        app:layout_constraintTop_toTopOf="parent" />

    <TextView
        android:id="@+id/mUID"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_marginBottom="116dp"
        android:text="TextView"
        tools:layout_editor_absoluteY="116dp" />

```

Output:



Result:

Successful creation of android app that uses intent with button to create a page and pass on values from one activity to another .

Learning outcomes:

- i. Understanding the basics of Activities in Android.
- ii. Setting up development environment for Android app development.
- iii. Learnt how to use intent sharing.
- iv. Designing a user-friendly interface for passing values of file to another using a button.



DEPARTMENT OF COMPUTER SCIENCE & ENGINEERING

Discover. Learn. Empower.

Experiment No. : 2.2

Student Name: Tanisha Singh	UID: 21BCS9213
Branch: BE CSE	Section: 21BCS_CC_646(B)
Subject Name: Mobile Application Development with Lab	Semester: Sixth
Date of Performance: 22/02/2024	Subject Code: 21CSH-355

Aim: Create an Android App using various controls such TexEdit, CheckBox, RadioButton, RadioGroup, etc.

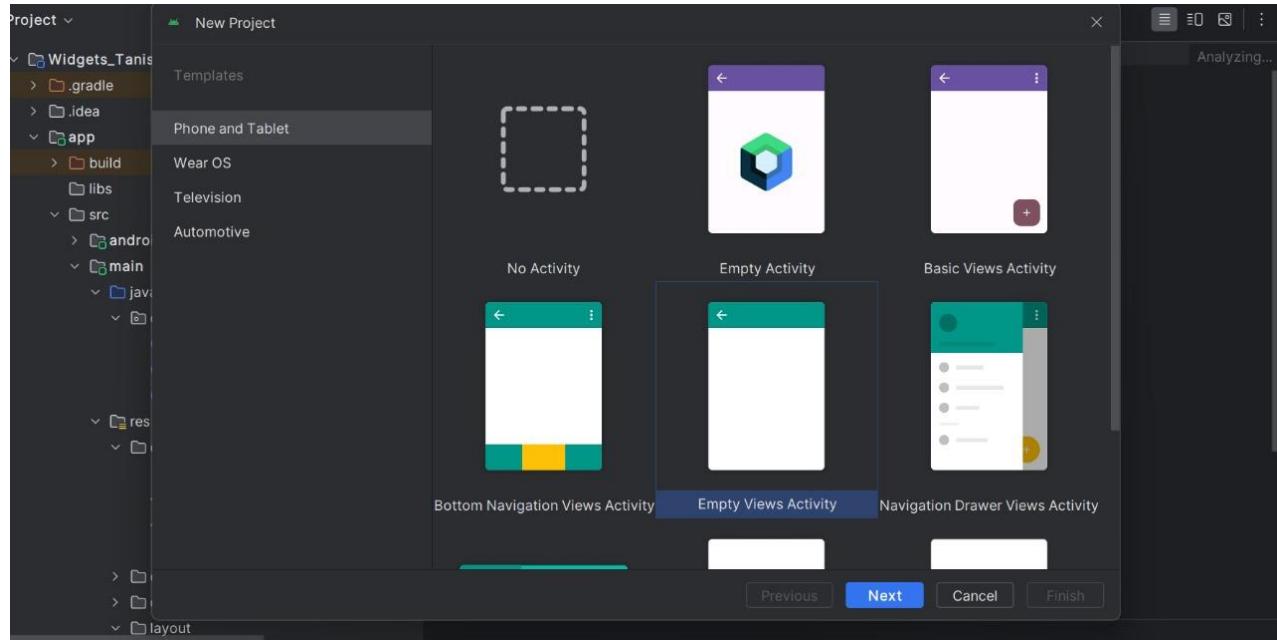
Objective: The objective of an Android app using various controls such as EditText, CheckBox, RadioButton, and RadioGroup could be to create a user interface that involves user input, selection, and interaction with different types of controls. This type of app aims to showcase the usage and functionalities of these UI elements to enhance the user experience.

Input/Apparatus Used:

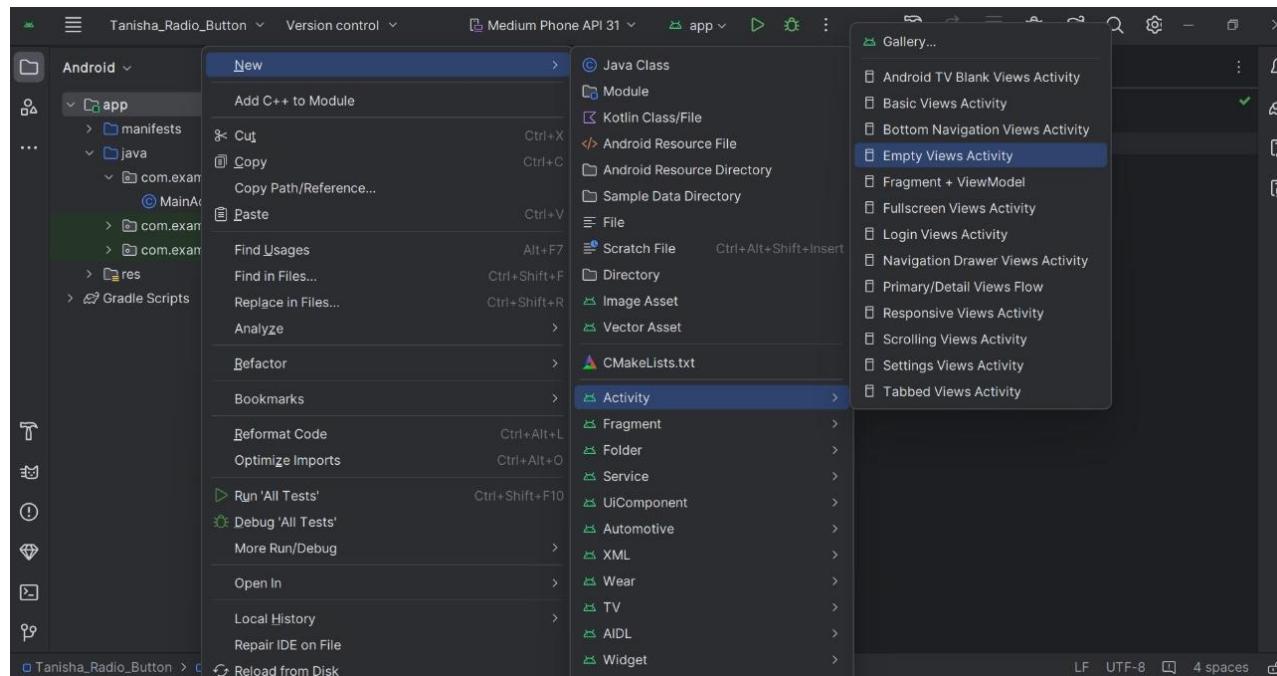
- 1. Integrated Development Environment (IDE) Android Studio:** The official IDE for Android development. Download and install Android Studio from the official website: Android Studio.
- 2. Android SDK:** The Android Software Development Kit (SDK) is essential for developing Android applications. Android Studio usually comes bundled with the SDK, but you may need to update it through the SDK Manager within Android Studio.
- 3. Java Development Kit (JDK):** Android apps are primarily written in Java or Kotlin. Make sure you have the Java Development Kit installed. Android Studio supports JDK. You can download it from the Oracle website: Java SE Downloads.
- 4. Android Virtual Device (AVD) or Physical Android Device:** You need a device to test your Android application. You can use an emulator (AVD) that comes with Android Studio or a physical Android device connected to your computer.

Script and Output:

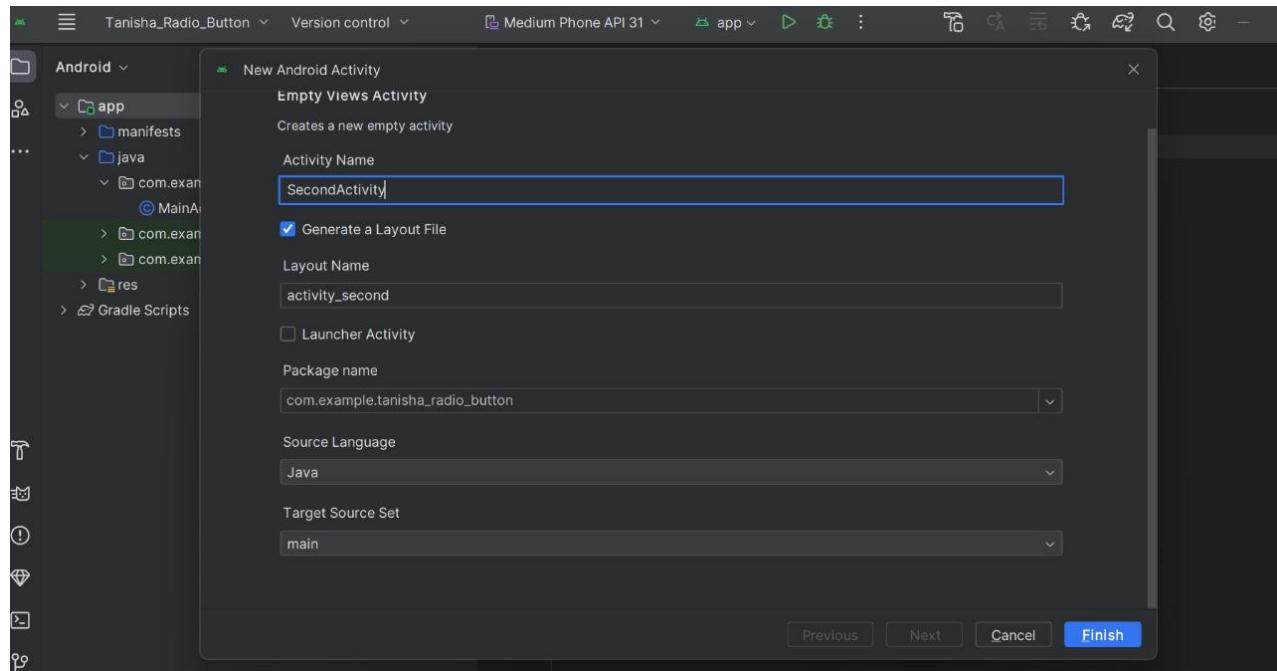
Step1. Click on the **New Project** in the Android studio and Select **Empty Views Activity** option.



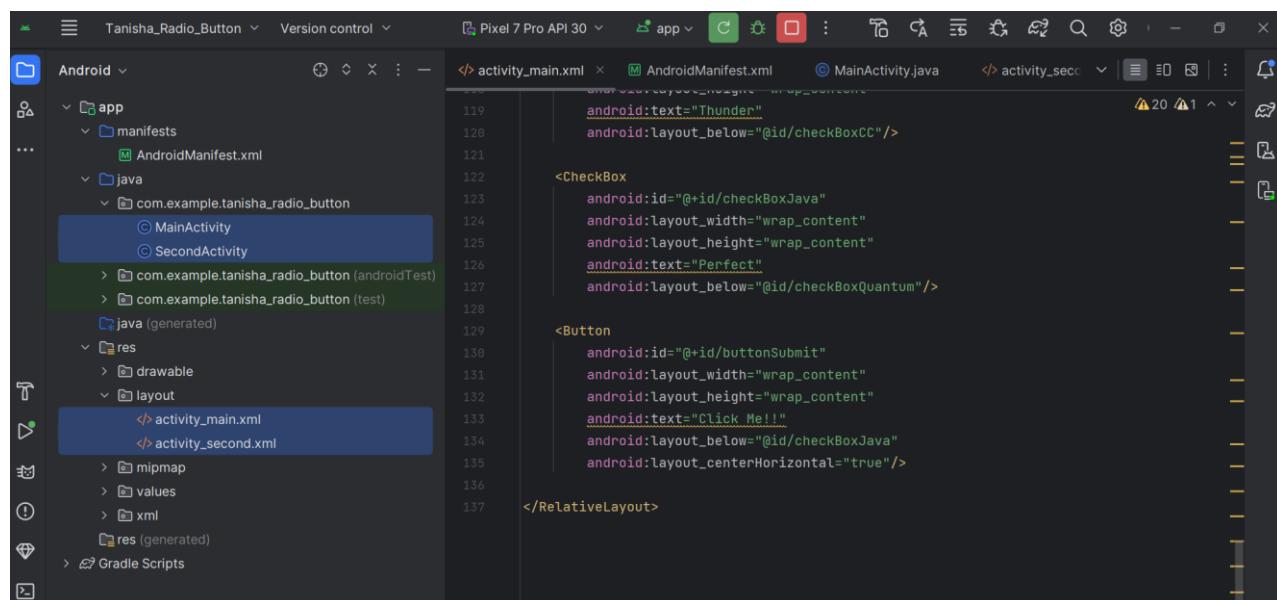
Step2. Create the **New Activity (Empty views Activity)** from New option in app.



Step3. Name the Activity as you want.



Step4. Open main Java and XML file by double clicking it.

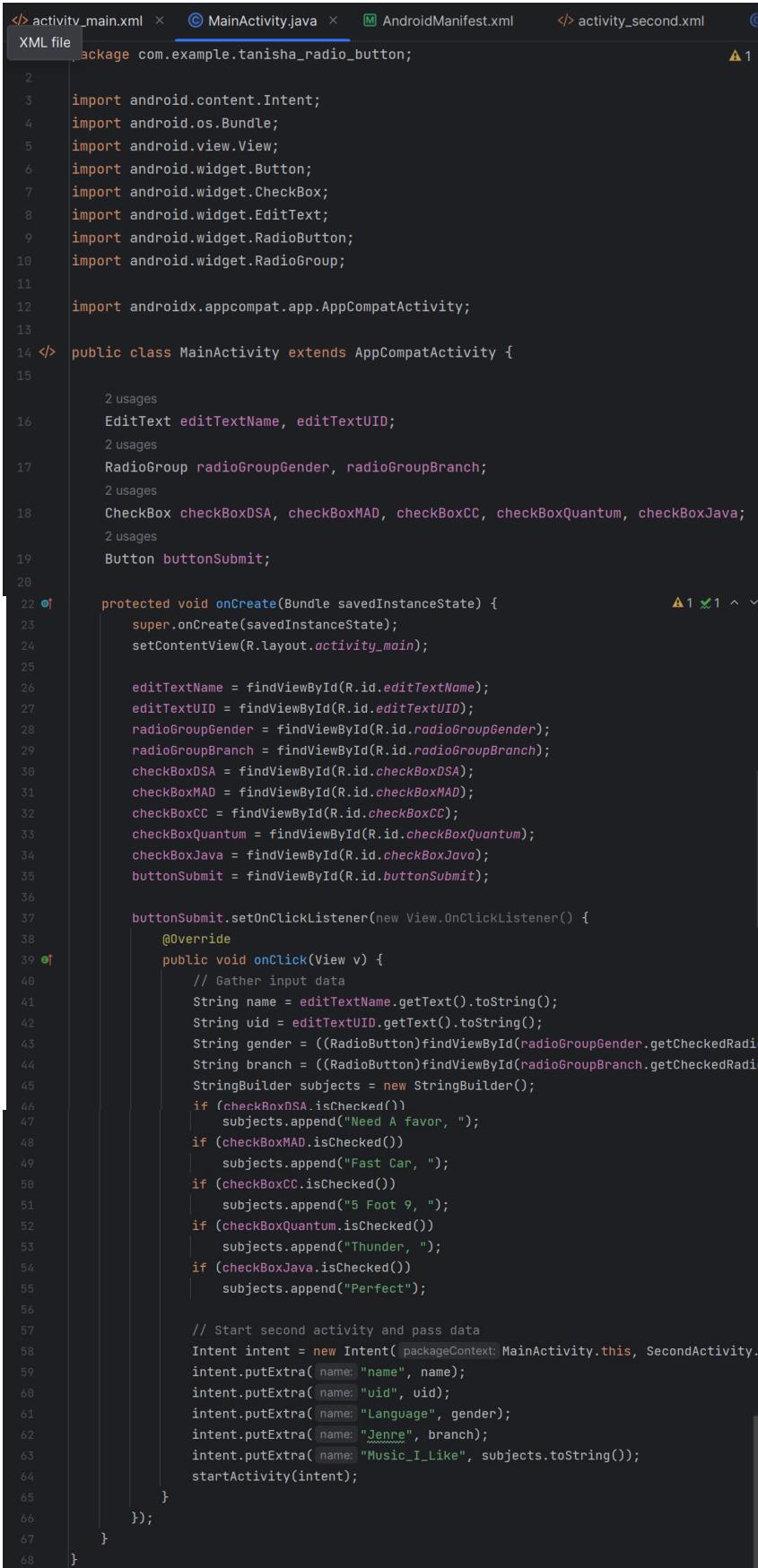




DEPARTMENT OF COMPUTER SCIENCE & ENGINEERING

Discover. Learn. Empower.

Step5. Write Java code as given below under MainActivity.java.



```
</> activity_main.xml <>MainActivity.java </> AndroidManifest.xml </> activity_second.xml </>
XML file package com.example.tanisha_radio_button;

2
3 import android.content.Intent;
4 import android.os.Bundle;
5 import android.view.View;
6 import android.widget.Button;
7 import android.widget.CheckBox;
8 import android.widget.EditText;
9 import android.widget.RadioButton;
10 import android.widget.RadioGroup;
11
12 import androidx.appcompat.app.AppCompatActivity;
13
14 </> public class MainActivity extends AppCompatActivity {
15
16     2 usages
17     EditText editTextName, editTextUID;
18     2 usages
19     RadioGroup radioGroupGender, radioGroupBranch;
20     2 usages
21     CheckBox checkBoxDSA, checkBoxMAD, checkBoxCC, checkBoxQuantum, checkBoxJava;
22     2 usages
23     Button buttonSubmit;
24
25     protected void onCreate(Bundle savedInstanceState) {
26         super.onCreate(savedInstanceState);
27         setContentView(R.layout.activity_main);
28
29         editTextName = findViewById(R.id.editTextName);
30         editTextUID = findViewById(R.id.editTextUID);
31         radioGroupGender = findViewById(R.id.radioGroupGender);
32         radioGroupBranch = findViewById(R.id.radioGroupBranch);
33         checkBoxDSA = findViewById(R.id.checkBoxDSA);
34         checkBoxMAD = findViewById(R.id.checkBoxMAD);
35         checkBoxCC = findViewById(R.id.checkBoxCC);
36         checkBoxQuantum = findViewById(R.id.checkBoxQuantum);
37         checkBoxJava = findViewById(R.id.checkBoxJava);
38         buttonSubmit = findViewById(R.id.buttonSubmit);
39
40         buttonSubmit.setOnClickListener(new View.OnClickListener() {
41             @Override
42             public void onClick(View v) {
43                 // Gather input data
44                 String name = editTextName.getText().toString();
45                 String uid = editTextUID.getText().toString();
46                 String gender = ((RadioButton)findViewById(radioGroupGender.getCheckedRadioButtonId())).getText().toString();
47                 String branch = ((RadioButton)findViewById(radioGroupBranch.getCheckedRadioButtonId())).getText().toString();
48                 StringBuilder subjects = new StringBuilder();
49                 if (checkBoxDSA.isChecked())
50                     subjects.append("Need A favor, ");
51                 if (checkBoxMAD.isChecked())
52                     subjects.append("Fast Car, ");
53                 if (checkBoxCC.isChecked())
54                     subjects.append("5 Foot 9, ");
55                 if (checkBoxQuantum.isChecked())
56                     subjects.append("Thunder, ");
57                 if (checkBoxJava.isChecked())
58                     subjects.append("Perfect");
59
60                 // Start second activity and pass data
61                 Intent intent = new Intent(getApplicationContext(), SecondActivity.class);
62                 intent.putExtra("name", name);
63                 intent.putExtra("uid", uid);
64                 intent.putExtra("Language", gender);
65                 intent.putExtra("Jenne", branch);
66                 intent.putExtra("Music_I_Like", subjects.toString());
67                 startActivity(intent);
68             }
69         });
70     }
71 }
```



DEPARTMENT OF COMPUTER SCIENCE & ENGINEERING

Discover. Learn. Empower.

Step6. - Go to **activity_main.xml** and write the code as given below.

```
1  <?xml version="1.0" encoding="utf-8"?>
2  <RelativeLayout xmlns:android="http://schemas.android.com/apk/res/android"
3      xmlns:tools="http://schemas.android.com/tools"
4      android:layout_width="match_parent"
5      android:layout_height="match_parent"
6      tools:context=".MainActivity">
7
8      <EditText
9          android:id="@+id/editTextName"
10         android:layout_width="match_parent"
11         android:layout_height="wrap_content"
12         android:hint="Name"/>
13
14      <EditText
15          android:id="@+id/editTextUID"
16          android:layout_width="match_parent"
17          android:layout_height="wrap_content"
18          android:layout_below="@+id/editTextName"
19          android:hint="UID (alphanumeric)"
20          android:minHeight="48dp"/>
21
22      <TextView
23          android:id="@+id/textViewGender"
24          android:layout_width="wrap_content"
25          android:layout_height="wrap_content"
26          android:layout_alignParentBottom="true"
27          android:layout_centerHorizontal="true"
28          android:gravity="center"
29          android:text="Gender: <input type="text" value=""/>
```

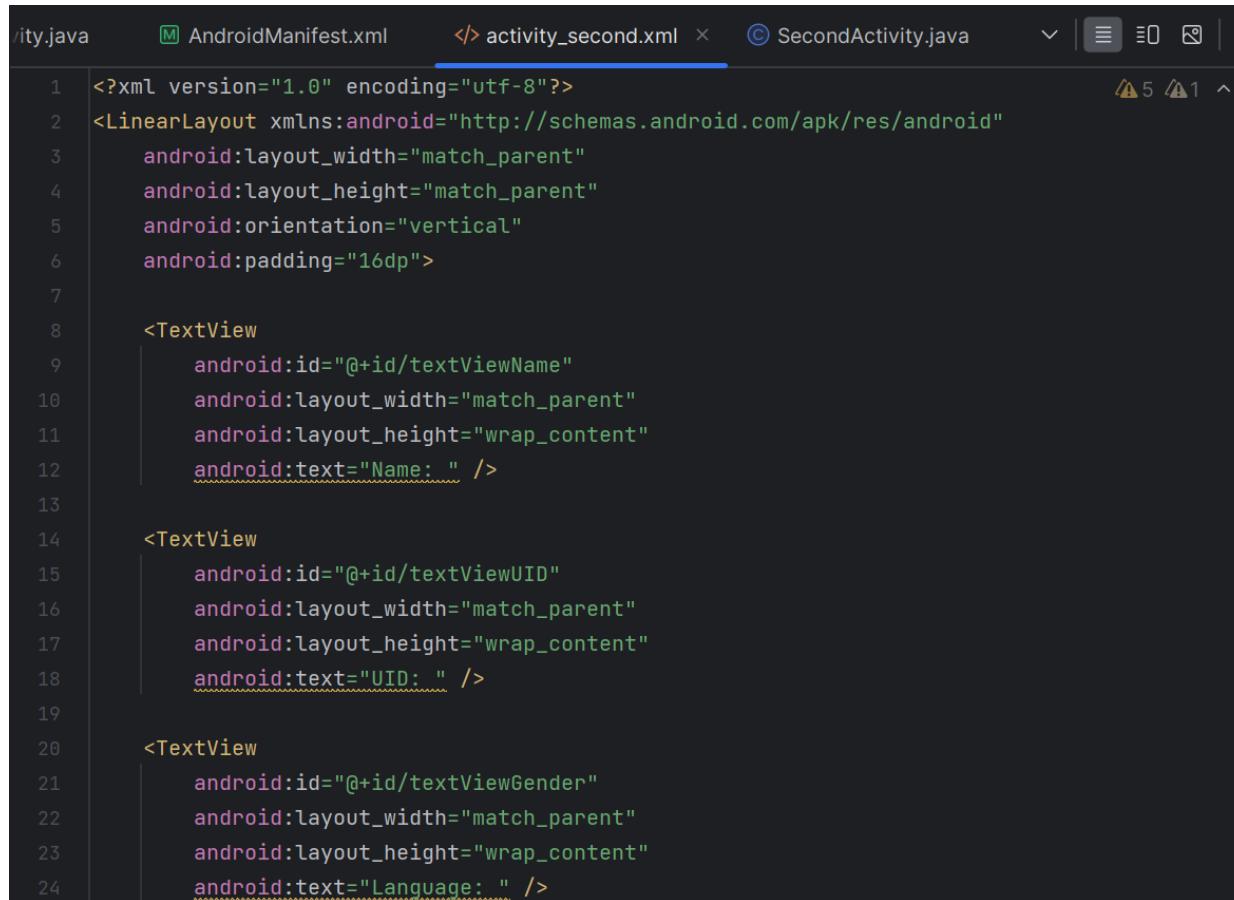
Step7. Go to **SecondActivity.java** and write the code as given below



DEPARTMENT OF COMPUTER SCIENCE & ENGINEERING

Discover. Learn. Empower.

Step 8. Go to **activity_second.xml** and write the code as given.



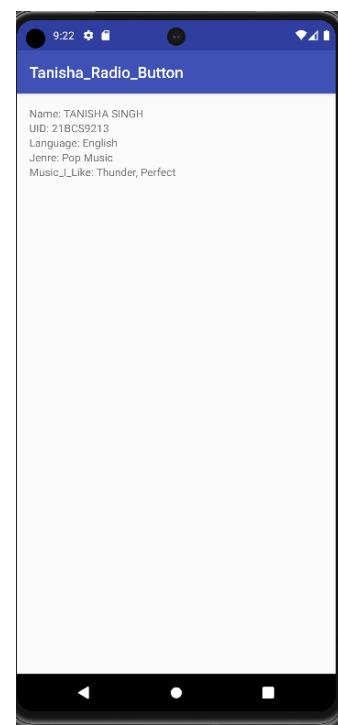
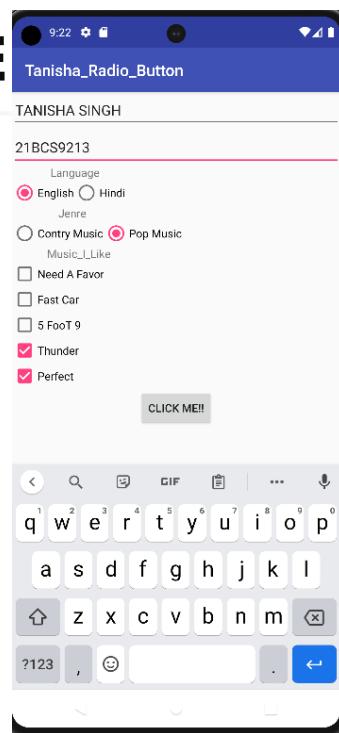
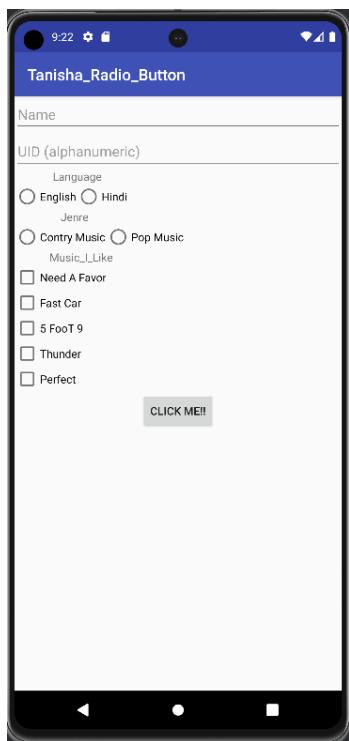
```
1 <?xml version="1.0" encoding="utf-8"?>
2 <LinearLayout xmlns:android="http://schemas.android.com/apk/res/android"
3     android:layout_width="match_parent"
4     android:layout_height="match_parent"
5     android:orientation="vertical"
6     android:padding="16dp">
7
8     <TextView
9         android:id="@+id/textViewName"
10        android:layout_width="match_parent"
11        android:layout_height="wrap_content"
12        android:text="Name: " />
13
14     <TextView
15         android:id="@+id/textViewUID"
16         android:layout_width="match_parent"
17         android:layout_height="wrap_content"
18         android:text="UID: " />
19
20     <TextView
21         android:id="@+id/textViewGender"
22         android:layout_width="match_parent"
23         android:layout_height="wrap_content"
24         android:text="Language: " />
```

Output:



DEPARTMENT OF COMPUTER SCIENCE

Discover. Learn. Empower.



Result:

Successful creation of android app using Button and CheckBox.

Learning outcomes:

- i. Understanding the basics of Activities in Android.
- ii. Setting up development environment for Android app development.



DEPARTMENT OF COMPUTER SCIENCE & ENGINEERING

iii. Learnt how to use Buttons and CheckBox.
iv. Designing a user-friendly interface for passing values of file to another

Discover. Learn. Implement.



DEPARTMENT OF COMPUTER SCIENCE & ENGINEERING

Discover. Learn. Empower.

Experiment No. : 2.3

Student Name: Tanisha Singh	UID: 21BCS9213
Branch: BE CSE	Section: 21BCS_CC_646(B)
Subject Name: Mobile Application Development with Lab	Semester: Sixth
Date of Performance: 29/02/2024	Subject Code: 21CSH-355

Aim: To design an android application Send SMS using Intent.

Objective: The objective of an Android-based application that uses Intent to send SMS can be to create a convenient and user-friendly tool for sending text messages. This type of app aims to leverage the Android platform's capabilities to provide a seamless and efficient way for users to compose and send SMS messages.

Input/Apparatus Used:

- 1. Integrated Development Environment (IDE) Android Studio:** The official IDE for Android development. Download and install Android Studio from the official website: Android Studio.
- 2. Android SDK:** The Android Software Development Kit (SDK) is essential for developing Android applications. Android Studio usually comes bundled with the SDK, but you may need to update it through the SDK Manager within Android Studio.
- 3. Java Development Kit (JDK):** Android apps are primarily written in Java or Kotlin. Make sure you have the Java Development Kit installed. Android Studio supports JDK. You can download it from the Oracle website: Java SE Downloads.
- 4. Android Virtual Device (AVD) or Physical Android Device:** You need a device to test your Android application. You can use an emulator (AVD) that comes with Android Studio or a physical Android device connected to your computer.

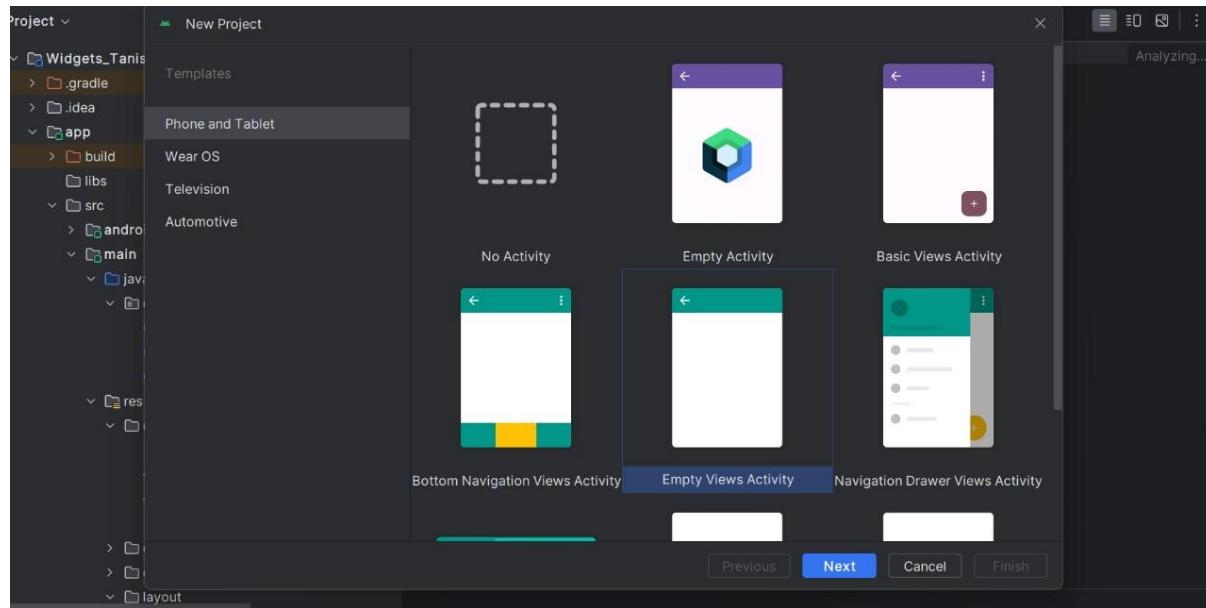


DEPARTMENT OF COMPUTER SCIENCE & ENGINEERING

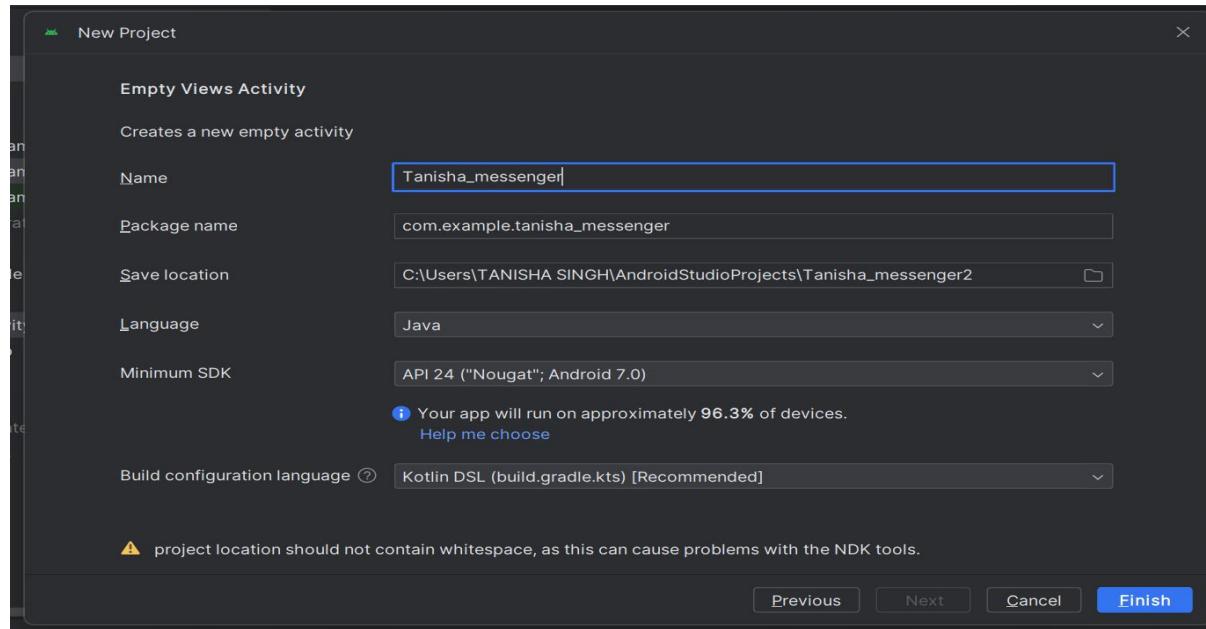
Discover. Learn. Empower.

Script and Output:

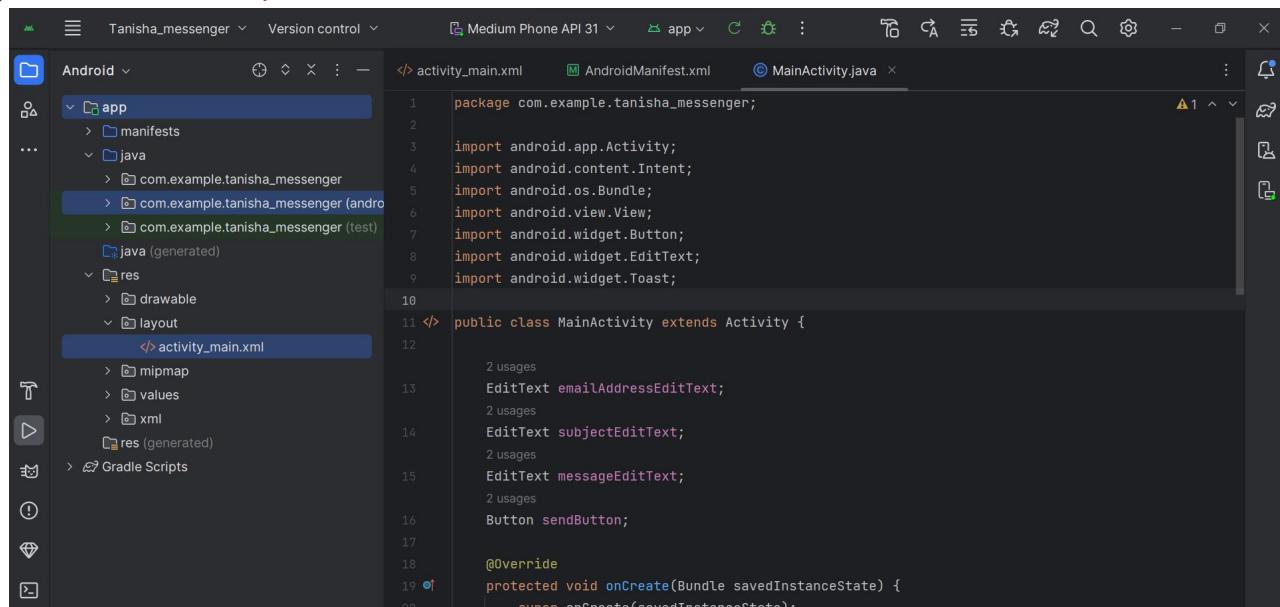
Step1. Click on the New Project in the Android studio and Select **Empty Views Activity** option.



Step2. Name the project as **Tanisha_messenger**.



Step3. Open main **Java** and **XML** file by double clicking it.



The screenshot shows the Android Studio interface. The left sidebar displays the project structure for 'Tanisha_messenger' under 'Android'. The 'app' folder is expanded, showing 'manifests', 'java', 'res', and 'layout'. The 'layout' folder contains 'activity_main.xml'. The right panel shows the code for 'MainActivity.java'. The code defines a class 'MainActivity' that extends 'Activity'. It imports various Android components like Activity, Intent, Bundle, View, Button, EditText, and Toast. The class has an empty onCreate method.

```

1 package com.example.tanisha_messenger;
2
3 import android.app.Activity;
4 import android.content.Intent;
5 import android.os.Bundle;
6 import android.view.View;
7 import android.widget.Button;
8 import android.widget.EditText;
9 import android.widget.Toast;
10
11 public class MainActivity extends Activity {
12
13     EditText emailAddressEditText;
14     EditText subjectEditText;
15     EditText messageEditText;
16     Button sendButton;
17
18     @Override
19     protected void onCreate(Bundle savedInstanceState) {
20         super.onCreate(savedInstanceState);
21     }
22 }

```

Step4. – Go to **activity_main.xml** and write the code as given below:

```

<RelativeLayout xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:tools="http://schemas.android.com/tools"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    android:padding="16dp"
    tools:context=".MainActivity">

    <EditText
        android:id="@+id/emailAddressEditText"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:hint="Email Address"
        android:inputType="textEmailAddress" />

    <EditText
        android:id="@+id/subjectEditText"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:layout_below="@+id/emailAddressEditText"
        android:layout_marginTop="16dp"
        android:hint="Subject" />

    <EditText
        android:id="@+id/messageEditText"

```



DEPARTMENT OF COMPUTER SCIENCE & ENGINEERING

Discover. Learn. Empower.

```
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:layout_below="@+id/subjectEditText"
    android:layout_marginTop="16dp"
    android:hint="Message"
    android:inputType="textMultiLine" />
```

```
<Button
    android:id="@+id/sendButton"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:layout_below="@+id/messageEditText"
    android:layout_marginTop="16dp"
    android:text="Send" />
```

```
</RelativeLayout>
```

Step4. Write **Java code** as given below under **MainActivity.java**.

```
package com.example.tanisha_messenger;
```

```
import android.app.Activity;
import android.content.Intent;
import android.os.Bundle;
import android.view.View;
import android.widget.Button;
import android.widget.EditText;
import android.widget.Toast;

public class MainActivity extends Activity {

    EditText emailAddressEditText;
    EditText subjectEditText;
    EditText messageEditText;
    Button sendButton;

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

        emailAddressEditText = findViewById(R.id.emailAddressEditText);
        subjectEditText = findViewById(R.id.subjectEditText);
        messageEditText = findViewById(R.id.messageEditText);
        sendButton = findViewById(R.id.sendButton);
    }

    public void sendMessage(View view) {
        String email = emailAddressEditText.getText().toString();
        String subject = subjectEditText.getText().toString();
        String message = messageEditText.getText().toString();

        if (email.isEmpty() || subject.isEmpty() || message.isEmpty()) {
            Toast.makeText(this, "All fields are required", Toast.LENGTH_SHORT).show();
        } else {
            Intent intent = new Intent(Intent.ACTION_SENDTO);
            intent.setData(Uri.parse("mailto:"));
            intent.putExtra(Intent.EXTRA_EMAIL, new String[]{email});
            intent.putExtra(Intent.EXTRA_SUBJECT, subject);
            intent.putExtra(Intent.EXTRA_TEXT, message);

            if (intent.resolveActivity(getPackageManager()) != null) {
                startActivity(intent);
            } else {
                Toast.makeText(this, "No email client found", Toast.LENGTH_SHORT).show();
            }
        }
    }
}
```



DEPARTMENT OF COMPUTER SCIENCE & ENGINEERING

Discover. Learn. Empower.

```
messageEditText = findViewById(R.id.messageEditText);
sendButton = findViewById(R.id.sendButton);

sendButton.setOnClickListener(new View.OnClickListener() {
    @Override
    public void onClick(View v) {
        String emailAddress = emailAddressEditText.getText().toString();
        String subject = subjectEditText.getText().toString();
        String message = messageEditText.getText().toString();

        if (!emailAddress.isEmpty() && !subject.isEmpty()
        && !message.isEmpty()) {
            sendEmail(emailAddress, subject, message);
        } else {
            Toast.makeText(MainActivity.this, "Please enter email address,
subject, and message", Toast.LENGTH_SHORT).show();
        }
    }
});

private void sendEmail(String emailAddress, String subject, String message) {
    Intent intent = new Intent(Intent.ACTION_SEND);
    intent.setType("message/rfc822");
    intent.putExtra(Intent.EXTRA_EMAIL, new String[]{emailAddress});
    intent.putExtra(Intent.EXTRA_SUBJECT, subject);
    intent.putExtra(Intent.EXTRA_TEXT, message);
    try {
        startActivity(Intent.createChooser(intent, "Send email using"));
    } catch (android.content.ActivityNotFoundException ex) {
        Toast.makeText(MainActivity.this, "No email clients installed.",
Toast.LENGTH_SHORT).show();
    }
}
```

Step5. Give the permission in the **AndroidManifest.xml**.

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



DEPARTMENT OF COMPUTER SCIENCE & ENGINEERING

Discover. Learn. Empower.

```
    android:name="android.hardware.telephony"
    android:required="false" />
<uses-permission android:name="android.permission.INTERNET" />

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

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

</manifest>
```

Output:

1. In the first section:

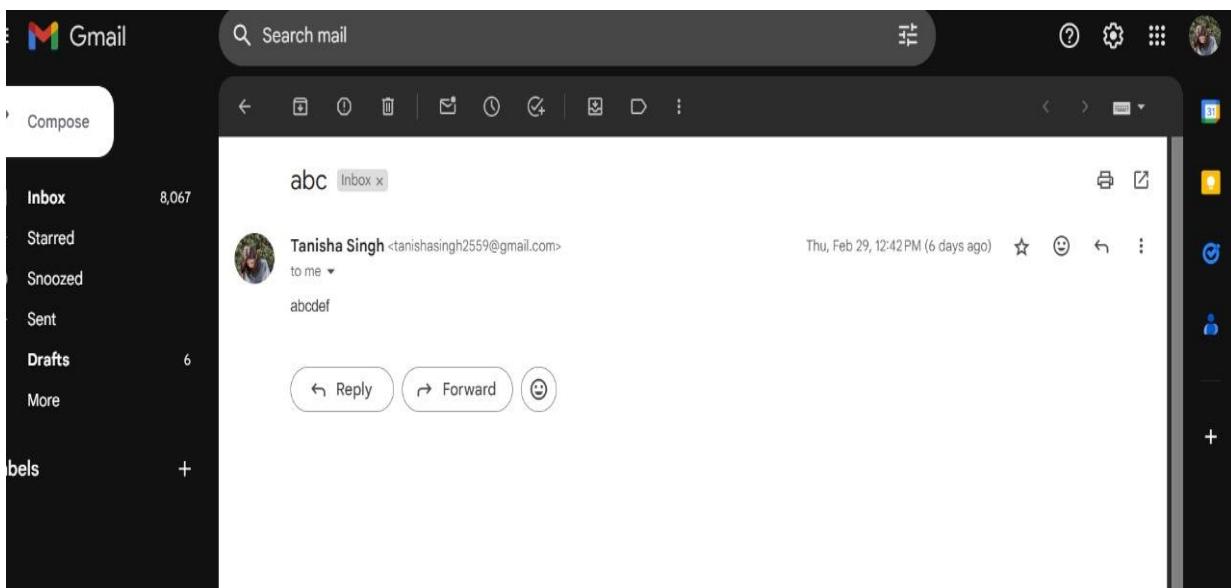
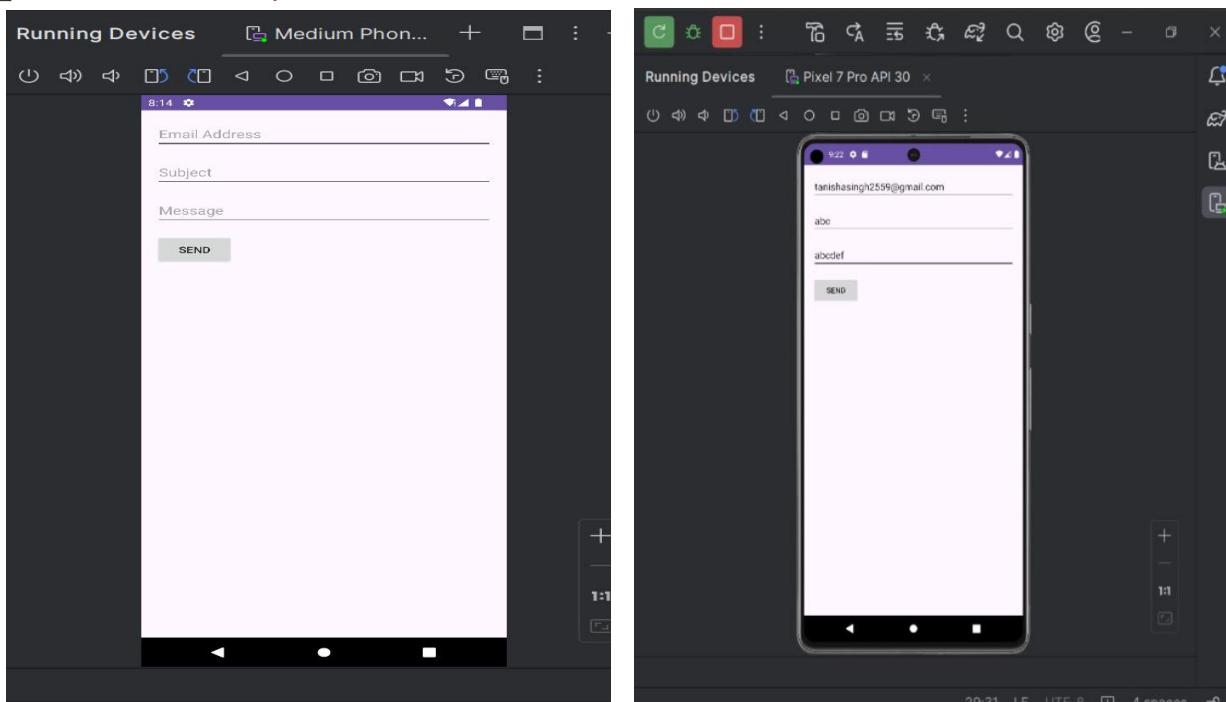
- i. Write the email address.
- ii. Write the subject.
- iii. Write the message.

2. In the second section:

Email is received on the address entered.

DEPARTMENT OF COMPUTER SCIENCE & ENGINEERING

Discover. Learn. Empower.



Result:

Successful creation of android app to send SMS using Intent.

Learning outcomes:

- i. Understanding the basics of Activities in Android.
- ii. Learnt how to create a convenient and user-friendly tool for sending text messages.
- iii. Learnt how to send messages using Intent.



DEPARTMENT OF COMPUTER SCIENCE & ENGINEERING

Discover. Learn. Empower.

Experiment No. : 3.1

Student Name: Tanisha Singh	UID: 21BCS9213
Branch: BE CSE	Section: 21BCS_CC_646(B)
Subject Name: Mobile Application Development with Lab	Semester: Sixth
Date of Performance: 14/03/2024	Subject Code: 21CSH-355

Aim: Create an Android application using Fragments.

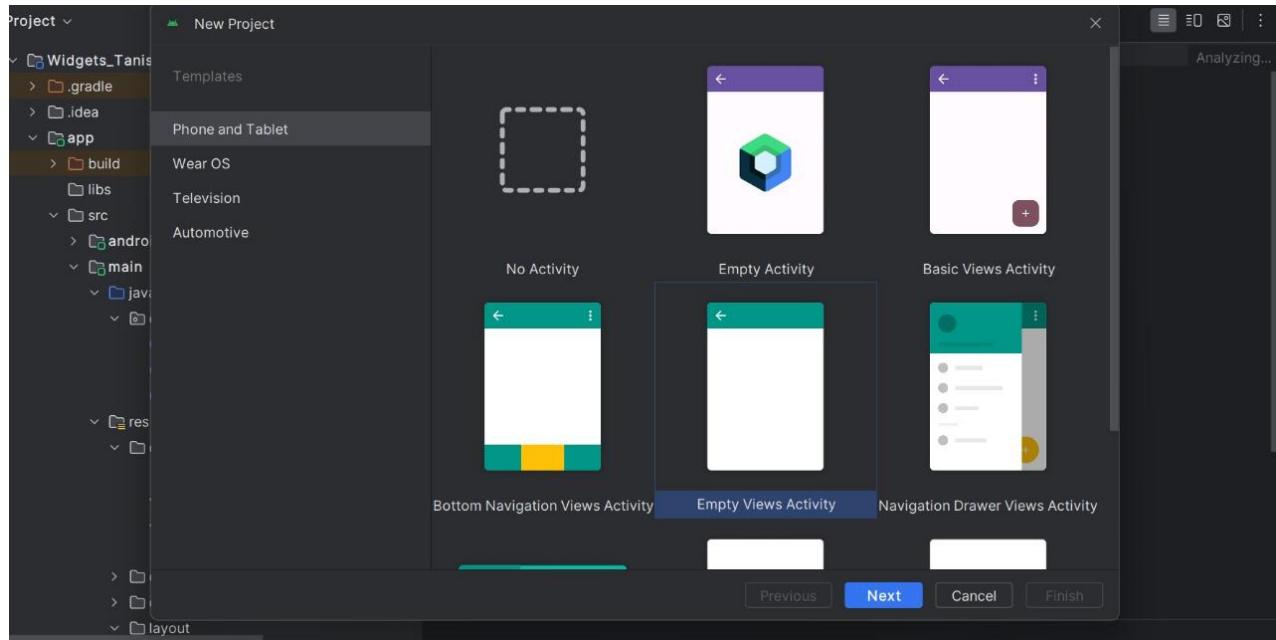
Objective: The objective of an Android application using Fragments can be to enhance the user interface and improve the overall user experience by leveraging the benefits of fragment-based design. Fragments allow developers to create more modular, flexible, and scalable UI components.

Input/Apparatus Used:

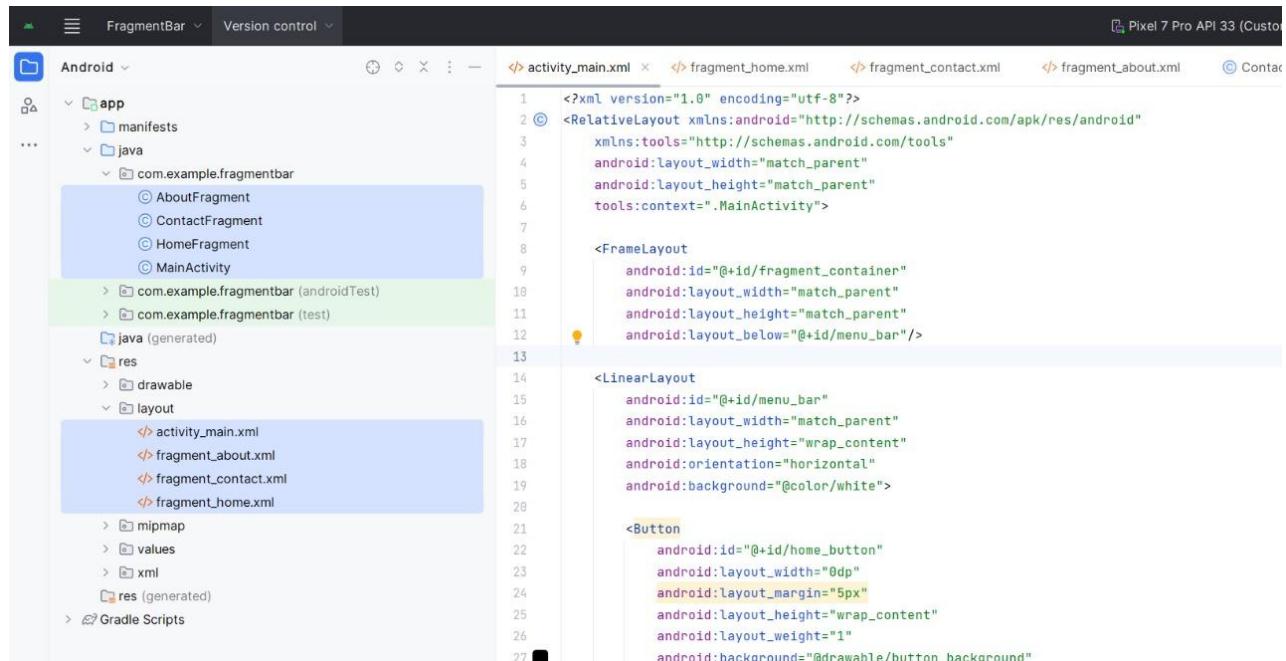
- 1. Integrated Development Environment (IDE) Android Studio:** The official IDE for Android development. Download and install Android Studio from the official website: [Android Studio](https://developer.android.com/studio).
- 2. Android SDK:** The Android Software Development Kit (SDK) is essential for developing Android applications. Android Studio usually comes bundled with the SDK, but you may need to update it through the SDK Manager within Android Studio.
- 3. Java Development Kit (JDK):** Android apps are primarily written in Java or Kotlin. Make sure you have the Java Development Kit installed. Android Studio supports JDK. You can download it from the Oracle website: [Java SE Downloads](https://www.oracle.com/java/technologies/javase-jdk-downloads.html).
- 4. Android Virtual Device (AVD) or Physical Android Device:** You need a device to test your Android application. You can use an emulator (AVD) that comes with Android Studio or a physical Android device connected to your computer.

Script and Output:

Step1. Click on the New Project in the Android studio and Select **Empty Views Activity** option.



Step2. Open main Java and XML file by double clicking it.





DEPARTMENT OF COMPUTER SCIENCE & ENGINEERING

Discover. Learn. Empower.

Step3. Write Java code as given below under **MainActivity.java**.

```
① AboutFragment.java ② ContactFragment.java ③ HomeFragment.java ④ MainActivity.java ×
1 package com.example.fragmentbar;
2
3 import androidx.appcompat.app.AppCompatActivity;
4 import android.os.Bundle;
5 import android.view.View;
6 import android.widget.Button;
7 import androidx.fragment.app.Fragment;
8 import androidx.fragment.app.FragmentManager;
9 import androidx.fragment.app.FragmentTransaction;
10 import com.example.fragmentbar.R;
11
12 </> public class MainActivity extends AppCompatActivity implements View.OnClickListener {
13
14     2 usages
15     Button homeButton, aboutButton, contactButton;
16
17     @Override
18     protected void onCreate(Bundle savedInstanceState) {
19         super.onCreate(savedInstanceState);
20         setContentView(R.layout.activity_main);
21
22         homeButton = findViewById(R.id.home_button);
23         aboutButton = findViewById(R.id.about_button);
24         contactButton = findViewById(R.id.contact_button);
25
26         homeButton.setOnClickListener(this);
27         aboutButton.setOnClickListener(this);
28         contactButton.setOnClickListener(this);
29
30         // Display home fragment initially
31         displayFragment(new HomeFragment());
32     }
33
34     @Override
35     public void onClick(View view) {
36         Fragment fragment = null;
37         switch (view.getId()) {
38             case R.id.home_button:
39                 fragment = new HomeFragment();
40                 break;
41             case R.id.about_button:
42                 fragment = new AboutFragment();
43                 break;
44             case R.id.contact_button:
45                 fragment = new ContactFragment();
46                 break;
47         }
48         displayFragment(fragment);
49     }
}
```



DEPARTMENT OF COMPUTER SCIENCE & ENGINEERING

Discover. Learn. Empower.

Step4. – Go to `activity_main.xml` and write the code as given below.

```
</> activity_main.xml  </> fragment_home.xml  </> fragment_contact.xml  </> fragment_about.xml  </>

1  <?xml version="1.0" encoding="utf-8"?>
2  <RelativeLayout xmlns:android="http://schemas.android.com/apk/res/android"
3      xmlns:tools="http://schemas.android.com/tools"
4      android:layout_width="match_parent"
5      android:layout_height="match_parent"
6      tools:context=".MainActivity">
7
8      <FrameLayout
9          android:id="@+id/fragment_container"
10         android:layout_width="match_parent"
11         android:layout_height="match_parent"
12         android:layout_below="@+id/menu_bar"/>
13
14      <LinearLayout
15          android:id="@+id/menu_bar"
16          android:layout_width="match_parent"
17          android:layout_height="wrap_content"
18          android:orientation="horizontal"
19          android:background="@color/white">
20
21          <Button
22              android:id="@+id/home_button"
23              android:layout_width="0dp"
24              android:layout_margin="5px"
25              android:layout_height="wrap_content"
26              android:layout_weight="1"
27              android:background="@drawable/button_background"
28              android:text="Home"
29              android:textColor="@color/white"
30              android:textAllCaps="false"/>
31
32          <Button
33              android:id="@+id/about_button"
34              android:layout_width="0dp"
35              android:layout_margin="5px"
36              android:layout_height="wrap_content"
37              android:layout_weight="1"
38              android:background="@drawable/button_background"
39              android:text="Hobbies"
40              android:textColor="@color/white"
41              android:textAllCaps="false"/>
42
43          <Button
44              android:id="@+id/contact_button"
45              android:layout_width="0dp"
46              android:layout_margin="5px"
47              android:layout_height="wrap_content"
48              android:layout_weight="1"
49              android:background="@drawable/button_background"
50              android:text="Contact"
51              android:textColor="@color/white"
52              android:textAllCaps="false"/>
```



DEPARTMENT OF COMPUTER SCIENCE & ENGINEERING

Discover. Learn. Empower.

Step5. Now Create 3 Fragments:

i. Home:



The screenshot shows a code editor with two tabs: HomeFragment.java and activity_main.xml. The HomeFragment.java tab is active, displaying the following Java code:

```
1 package com.example.fragmentbar;
2
3 import android.os.Bundle;
4 import androidx.fragment.app.Fragment;
5 import android.view.LayoutInflater;
6 import android.view.View;
7 import android.view.ViewGroup;
8
9 public class HomeFragment extends Fragment {
10
11     public HomeFragment() {
12         // Required empty public constructor
13     }
14
15     @Override
16     public View onCreateView(LayoutInflater inflater, ViewGroup container,
17         Bundle savedInstanceState) {
18         // Inflate the layout for this fragment
19         return inflater.inflate(R.layout.fragment_home, container, false);
20     }
21 }
22
```

The activity_main.xml tab is visible below, showing the XML layout for the fragment:

```
1 <?xml version="1.0" encoding="utf-8"?>
2 <RelativeLayout xmlns:android="http://schemas.android.com/apk/res/android"
3     android:layout_width="match_parent"
4     android:layout_height="match_parent"
5     android:background="@color/beige">
6
7     <TextView
8         android:id="@+id/name_text"
9         android:layout_width="wrap_content"
10        android:layout_height="wrap_content"
11        android:text="Tanisha Singh"
12        android:textSize="24sp"
13        android:textStyle="bold"
14        android:layout_centerInParent="true"/>
15
16     <TextView
17         android:layout_width="wrap_content"
18         android:layout_height="wrap_content"
19         android:text="21BCS9213"
20         android:textSize="16sp"
21         android:textStyle="italic"
22         android:layout_below="@+id/name_text"
23         android:layout_centerHorizontal="true"/>
24
25 </RelativeLayout>
26
```



DEPARTMENT OF COMPUTER SCIENCE & ENGINEERING

Discover. Learn. Empower.

ii. Hobbies:

```
activity_main.xml fragment_home.xml fragment_contact.xml fragment_about.xml AboutFragment.java
1 package com.example.fragmentbar;
2
3 import android.os.Bundle;
4 import androidx.fragment.app.Fragment;
5 import android.view.LayoutInflater;
6 import android.view.View;
7 import android.view.ViewGroup;
8
9 public class AboutFragment extends Fragment {
10
11     public AboutFragment() {
12         // Required empty public constructor
13     }
14
15     @Override
16     public View onCreateView(LayoutInflater inflater, ViewGroup container,
17                             Bundle savedInstanceState) {
18         // Inflate the layout for this fragment
19         return inflater.inflate(R.layout.fragment_about, container, false);
20     }
21 }
22
```

```
activity_main.xml fragment_home.xml fragment_contact.xml fragment_about.xml
1 <?xml version="1.0" encoding="utf-8"?>
2 <ScrollView xmlns:android="http://schemas.android.com/apk/res/android"
3     android:layout_width="match_parent"
4     android:layout_height="match_parent"
5     android:background="@color/beige">
6
7     <LinearLayout
8         android:layout_width="match_parent"
9         android:layout_height="wrap_content"
10        android:orientation="vertical"
11        android:padding="16dp">
12
13         <TextView
14             android:layout_width="wrap_content"
15             android:layout_height="wrap_content"
16             android:text="Hobbies"
17             android:textSize="40sp"
18             android:textStyle="bold"/>
19
20         <TextView
21             android:layout_width="wrap_content"
22             android:layout_height="wrap_content"
23             android:text="ART
24             JOURNALING
25             LISTENING TO MUSIC"
26             android:textSize="20sp"/>
27
28     </LinearLayout>
29 </ScrollView>
30 |
```



iii. Contact:

```
<?xml version="1.0" encoding="utf-8"?>
<LinearLayout xmlns:android="http://schemas.android.com/apk/res/android"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    android:orientation="vertical"
    android:padding="16dp"
    android:background="@color/beige">

    <!-- Email Address -->
    <TextView
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:text="Email Address"
        android:textSize="15sp"/>

    <TextView
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:text="Tanishasingh2559@gmail.com"
        android:textSize="20sp"/>

    <!-- Subject -->
    <EditText
        android:id="@+id/subject_edit_text"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:hint="Subject"
        android:textColor="@color/black"
        android:inputType="text"
        android:minHeight="48dp" />

    <!-- Message -->
    <EditText
        android:id="@+id/message_edit_text"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:inputType="textMultiLine"
        android:minHeight="100dp" />

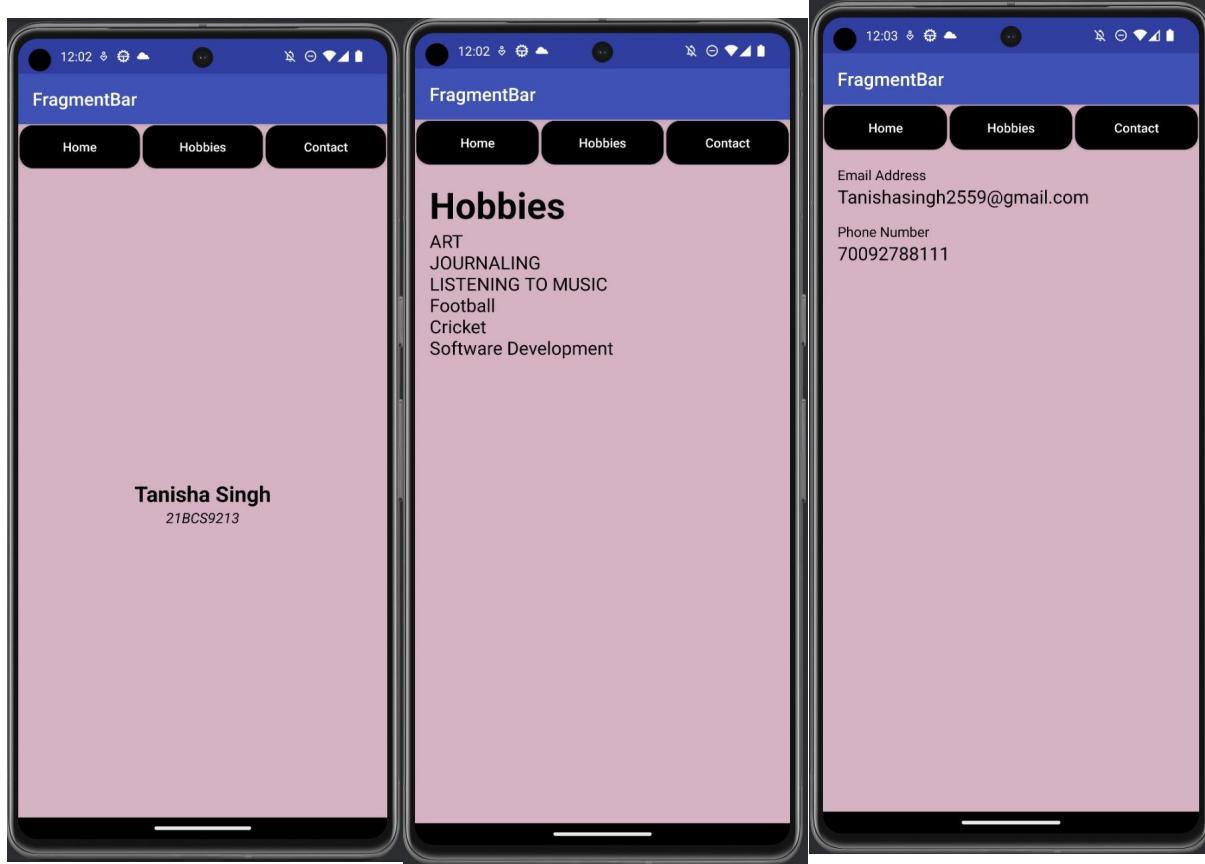
```



DEPARTMENT OF COMPUTER SCIENCE & ENGINEERING

Discover. Learn. Empower.

Output:



Result:

Successful creation of android app using Fragments.

Learning outcomes:

- Successful use of Android development environment.
- Project Workspace.
- Configuration button Completion.
- How to add Fragments.



DEPARTMENT OF COMPUTER SCIENCE & ENGINEERING

Discover. Learn. Empower.

Experiment No. : 3.2

Student Name: Tanisha Singh	UID: 21BCS9213
Branch: BE CSE	Section: 21BCS_CC_646(B)
Subject Name: Mobile Application Development with Lab	Semester: Sixth
Date of Performance: 21/03/2024	Subject Code: 21CSH-355

Aim: Implement building blocks for Android Application using different layouts such as linear, relative and absolute .

Objective: The objective of implementing building blocks for an Android application using different layouts such as linear, relative, and absolute is to create a diverse and visually appealing user interface that accommodates various design requirements. Different layout types offer flexibility in organizing UI components, and understanding their usage is crucial for effective Android app development.

Input/Apparatus Used:

1. Integrated Development Environment (IDE) Android Studio: The official IDE for Android development. Download and install Android Studio from the official website: [Android Studio](#).

2. Android SDK: The Android Software Development Kit (SDK) is essential for developing Android applications. Android Studio usually comes bundled with the SDK, but you may need to update it through the SDK Manager within Android Studio.

3. Java Development Kit (JDK): Android apps are primarily written in Java or Kotlin. Make sure you have the Java Development Kit installed. Android Studio supports JDK. You can download it from the Oracle website: [Java SE Downloads](#).

4. Android Virtual Device (AVD) or Physical Android Device: You need a device to test your Android application. You can use an emulator (AVD) that comes with Android Studio or a physical Android device connected to your computer.

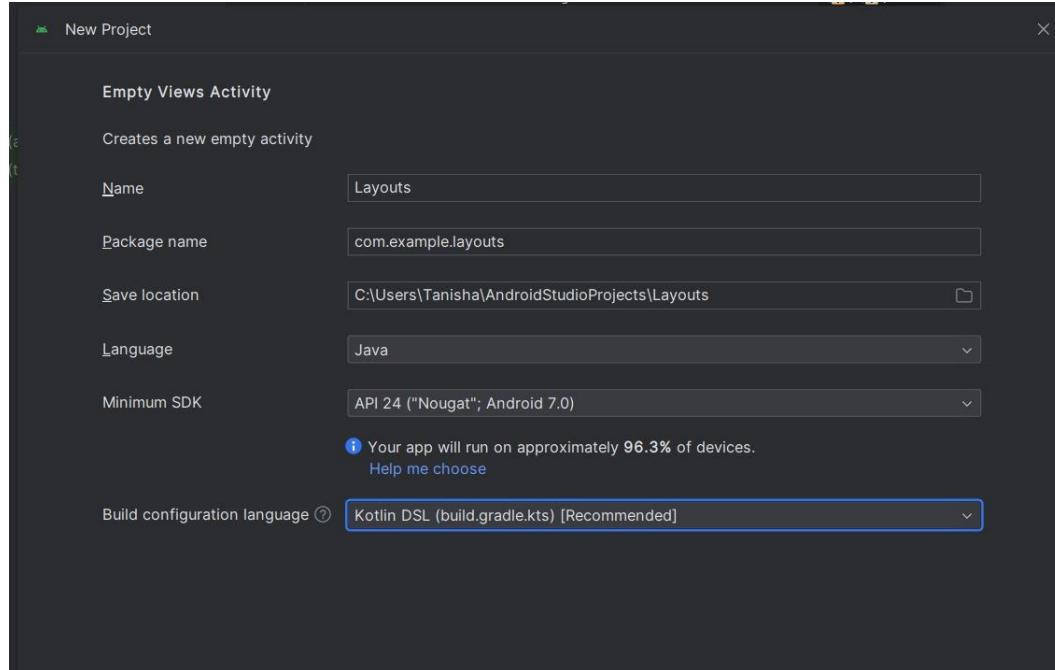


DEPARTMENT OF COMPUTER SCIENCE & ENGINEERING

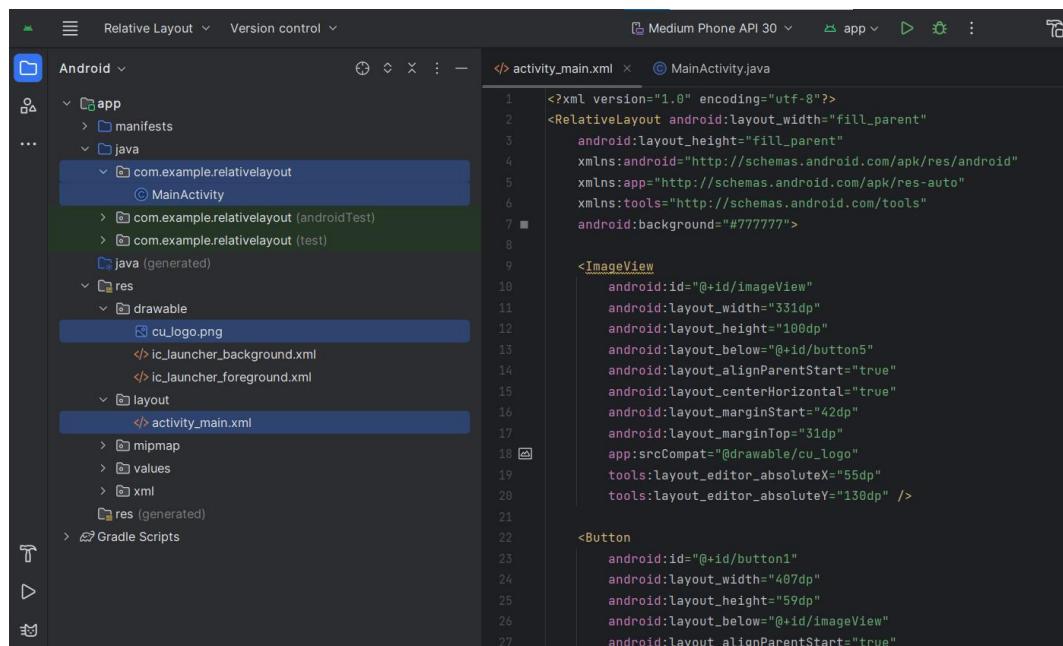
Discover. Learn. Empower.

Script and Output:

Step1. Click on the **New Project** in the Android studio and Select **Empty Views Activity** option.



Step2. Open main Java and XML file by double clicking it.





DEPARTMENT OF COMPUTER SCIENCE & ENGINEERING

Discover. Learn. Empower.

(i) .Relative Layout:

Activity_main.xml-

```
<?xml version="1.0" encoding="utf-8"?>
<RelativeLayout android:layout_width="fill_parent"
    android:layout_height="fill_parent"
    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:background="#777777">

    <ImageView
        android:id="@+id/imageView"
        android:layout_width="331dp"
        android:layout_height="100dp"
        android:layout_below="@+id/button5"
        android:layout_alignParentStart="true"
        android:layout_centerHorizontal="true"
        android:layout_marginStart="42dp"
        android:layout_marginTop="31dp"
        app:srcCompat="@drawable/cu_logo"
        tools:layout_editor_absoluteX="55dp"
        tools:layout_editor_absoluteY="130dp" />

    <Button
        android:id="@+id/button1"
        android:layout_width="407dp"
        android:layout_height="59dp"
        android:layout_below="@+id/imageView"
        android:layout_alignParentStart="true"
        android:layout_alignParentLeft="true"
        android:layout_alignParentEnd="true"
        android:layout_centerHorizontal="true"
        android:layout_marginStart="4dp"
        android:layout_marginLeft="4dp"
        android:layout_marginTop="74dp"
        android:layout_marginEnd="0dp"
        android:text="Tanisha Singh" />

    <Button
        android:id="@+id/button2"
```



DEPARTMENT OF COMPUTER SCIENCE & ENGINEERING

Discover. Learn. Empower.

```
    android:layout_width="180dp"
    android:layout_height="wrap_content"
    android:layout_below="@+id/button1"
    android:layout_alignParentEnd="true"
    android:layout_alignParentRight="true"
    android:layout_marginBottom="142dp"
    android:text="21BCS9213" />
```

```
<Button
    android:id="@+id/button3"
    android:layout_width="214dp"
    android:layout_height="wrap_content"
    android:layout_below="@+id/button1"
    android:layout_alignParentStart="true"
    android:layout_alignParentLeft="true"
    android:layout_marginEnd="39dp"
    android:layout_toStartOf="@+id/button2"
    android:text="21BCS_CC_646(B)" />
```

```
<Button
    android:id="@+id/button5"
    android:layout_width="match_parent"
    android:layout_height="82dp"
    android:layout_alignParentStart="true"
    android:layout_alignParentTop="true"
    android:layout_marginStart="9dp"
    android:layout_marginTop="108dp"
    android:text="MAD Experiment-8th" />
</RelativeLayout>
```

MainActivity.java-

```
package com.example.relativelayout;
import androidx.appcompat.app.AppCompatActivity;
import android.os.Bundle;
public class MainActivity extends AppCompatActivity {
    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_main); }}
```



DEPARTMENT OF COMPUTER SCIENCE & ENGINEERING

Discover. Learn. Empower.

Output:





DEPARTMENT OF COMPUTER SCIENCE & ENGINEERING

Discover. Learn. Empower.

(ii). Absolute Layout:

Activity_main.xml:

```
<?xml version="1.0" encoding="utf-8"?>
<AbsoluteLayout
    xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:tools="http://schemas.android.com/tools"
    android:layout_width="fill_parent"
    android:layout_height="fill_parent"
    tools:context=".MainActivity">

    <TextView
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="AbsoluteLayout Example"
        android:textSize="24sp"
        android:textStyle="bold"
        android:layout_x="16dp"
        android:layout_y="16dp"
        />

    <!-- Setting up TextViews -->
    <TextView android:id="@+id/heading"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_x="100px"
        android:layout_y="300px"
        android:text="Tanisha Singh" />

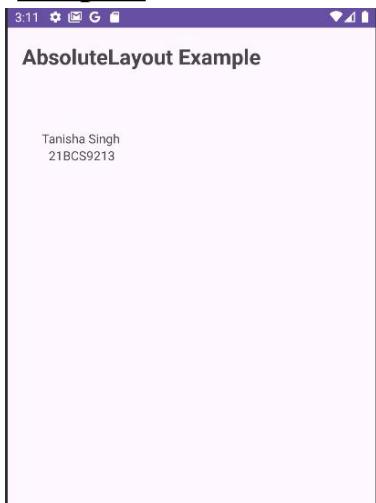
    <TextView
        android:id="@+id/subHeading"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_x="120px"
        android:layout_y="350px"
        android:text="21BCS9213" />
</AbsoluteLayout>
```

MainActivity.java-

```
package com.example.absolutelayout;
import android.os.Bundle;

import android.widget.TextView;
import androidx.appcompat.app.AppCompatActivity; public class MainActivity
extends AppCompatActivity {
    TextView heading, subHeading; @Override
    protected void onCreate(Bundle savedInstanceState)
    { super.onCreate(savedInstanceState); setContentView(R.layout.activity_main);
    // Referencing the TextViews
    heading = findViewById(R.id.heading); subHeading =
    findViewById(R.id.subHeading);
    // Setting text dynamically heading.setText("Chandigarh University");
    subHeading.setText("Computer Science Portal");
    }
}
```

Output:



Result:

Successful creation of android app using Fragments.

Learning outcomes:

- Successful USE of your Android development environment.
- Project Workspace.
- Configuration button Completion.
- How to add new layouts.



DEPARTMENT OF COMPUTER SCIENCE & ENGINEERING

Discover. Learn. Empower.

Experiment No. : 3.4

Student Name: Tanisha Singh	UID: 21BCS9213
Branch: BE CSE	Section: 21BCS_CC_646(B)
Subject Name: Mobile Application Development with Lab	Semester: Sixth
Date of Performance: 04/03/2024	Subject Code: 21CSH-355

Aim: Create an Android application for user registration that stores the user details in a database table.

Objective: The objective of an Android application for user registration that stores user details in a database table is to create a secure, efficient, and user-friendly registration system. This type of app is commonly developed for services that require user accounts, such as social media platforms, e-commerce applications, or any service where personalized user data needs to be stored.

Input/Apparatus Used:

- 1. Integrated Development Environment (IDE) Android Studio:** The official IDE for Android development. Download and install Android Studio from the official website: [Android Studio](https://developer.android.com/studio).
- 2. Android SDK:** The Android Software Development Kit (SDK) is essential for developing Android applications. Android Studio usually comes bundled with the SDK, but you may need to update it through the SDK Manager within Android Studio.
- 3. Java Development Kit (JDK):** Android apps are primarily written in Java or Kotlin. Make sure you have the Java Development Kit installed. Android Studio supports JDK. You can download it from the Oracle website: [Java SE Downloads](https://www.oracle.com/java/technologies/javase/javase-jdk-downloads.html).
- 4. Android Virtual Device (AVD) or Physical Android Device:** You need a device to test your Android application. You can use an emulator (AVD) that comes with Android Studio or a physical Android device connected to your computer.

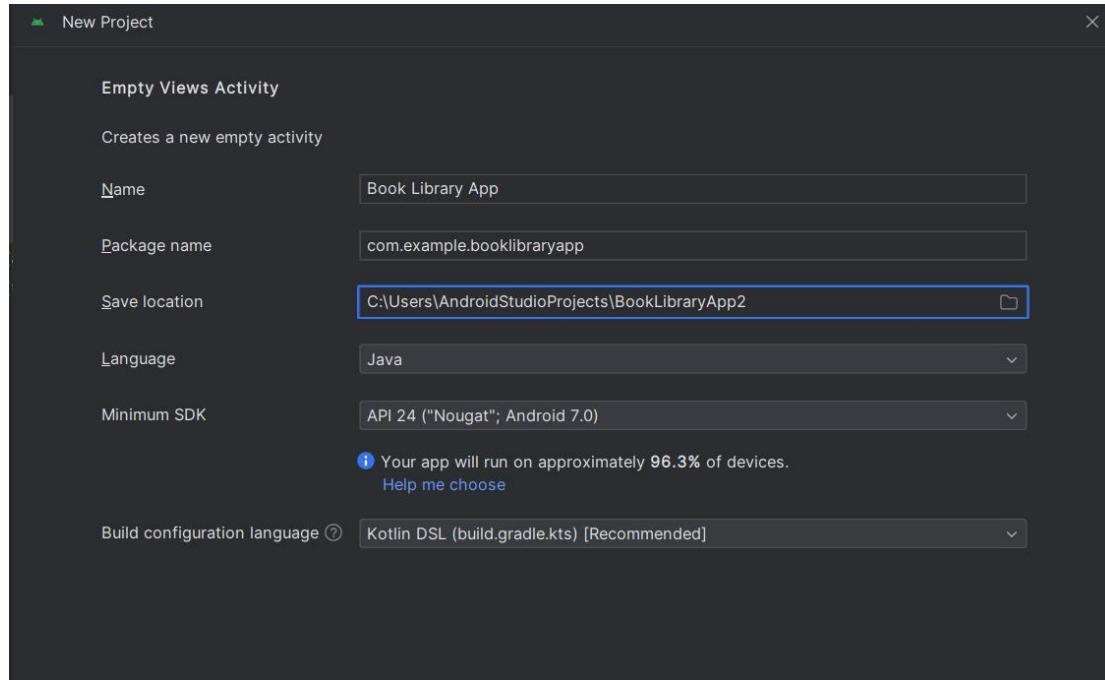


DEPARTMENT OF COMPUTER SCIENCE & ENGINEERING

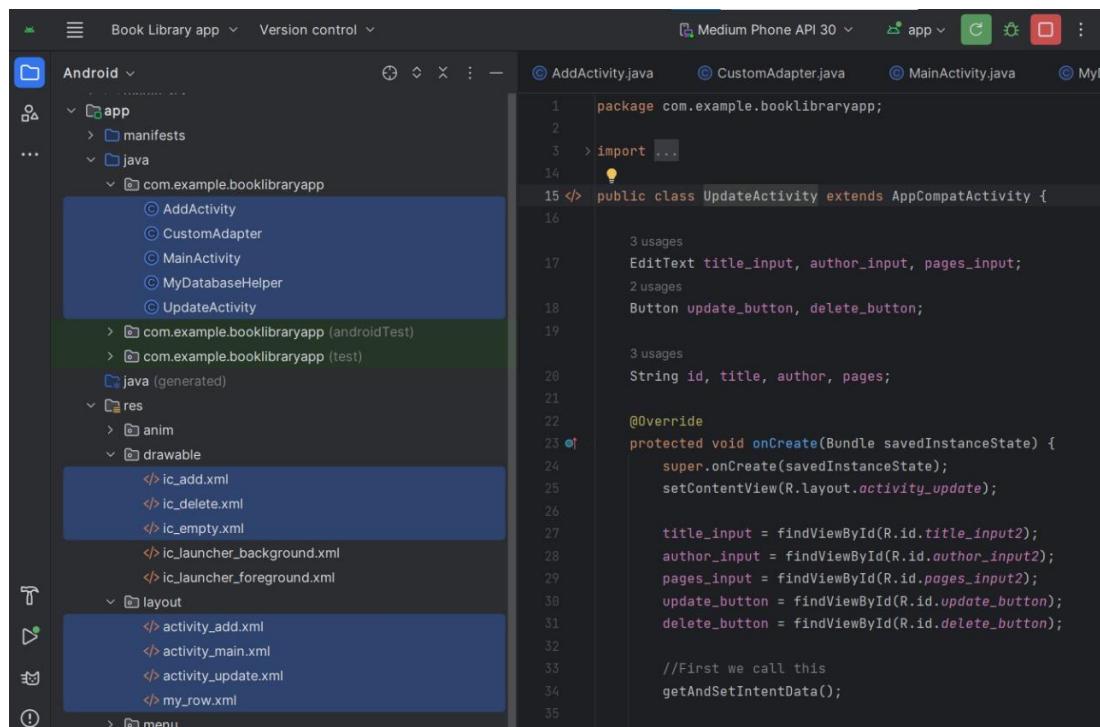
Discover. Learn. Empower.

Script and Output:

Step1. Click on the **New Project** in the Android studio and Select **Empty Views Activity** option.



Step2. Open main **Java** and **XML** file by double clicking it.





DEPARTMENT OF COMPUTER SCIENCE & ENGINEERING

Discover. Learn. Empower.

● **MainActivity.java-**

```
package com.example.booklibraryapp;

import androidx.annotation.Nullable;
import androidx.appcompat.app.AlertDialog;
import androidx.appcompat.app.AppCompatActivity;
import androidx.recyclerview.widget.LinearLayoutManager;
import androidx.recyclerview.widget.RecyclerView;

import android.content.DialogInterface;
import android.content.Intent;
import android.database.Cursor;
import android.database.sqlite.SQLiteDatabase;
import android.os.Bundle;
import android.view.ContextMenu;
import android.view.Menu;
import android.view.MenuInflater;
import android.view.MenuItem;
import android.view.View;
import android.widget.ImageView;
import android.widget.TextView;
import android.widget.Toast;

import com.google.android.material.floatingactionbutton.FloatingActionButton;

import java.util.ArrayList;

public class MainActivity extends AppCompatActivity {

    RecyclerView recyclerView;
    FloatingActionButton add_button;
    ImageView empty_imageview;
    TextView no_data;

    MyDatabaseHelper myDB;
    ArrayList<String> book_id, book_title, book_author, book_pages;
    CustomAdapter customAdapter;

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



DEPARTMENT OF COMPUTER SCIENCE & ENGINEERING

Discover. Learn. Empower.

```
super.onCreate(savedInstanceState);
setContentView(R.layout.activity_main);

recyclerView = findViewById(R.id.recyclerView);
add_button = findViewById(R.id.add_button);
empty_imageview = findViewById(R.id.empty_imageview);
no_data = findViewById(R.id.no_data);
add_button.setOnClickListener(new View.OnClickListener() {
    @Override
    public void onClick(View view) {
        Intent intent = new Intent(MainActivity.this, AddActivity.class);
        startActivity(intent);
    }
});

myDB = new MyDatabaseHelper(MainActivity.this);
book_id = new ArrayList<>();
book_title = new ArrayList<>();
book_author = new ArrayList<>();
book_pages = new ArrayList<>();

storeDataInArrays();

customAdapter = new CustomAdapter(MainActivity.this, this, book_id,
book_title, book_author,
book_pages);
recyclerView.setAdapter(customAdapter);
recyclerView.setLayoutManager(new
LinearLayoutManager(MainActivity.this));

}

@Override
protected void onActivityResult(int requestCode, int resultCode, @Nullable Intent
data) {
    super.onActivityResult(requestCode, resultCode, data);
    if(requestCode == 1){
        recreate();
    }
}

void storeDataInArrays(){
```



DEPARTMENT OF COMPUTER SCIENCE & ENGINEERING

Discover. Learn. Empower.

```
Cursor cursor = myDB.readAllData();
if(cursor.getCount() == 0){
    empty_imageview.setVisibility(View.VISIBLE);
    no_data.setVisibility(View.VISIBLE);
} else{
    while (cursor.moveToNext()){
        book_id.add(cursor.getString(0));
        book_title.add(cursor.getString(1));
        book_author.add(cursor.getString(2));
        book_pages.add(cursor.getString(3));
    }
    empty_imageview.setVisibility(View.GONE);
    no_data.setVisibility(View.GONE);
}

@Override
public boolean onCreateOptionsMenu(Menu menu) {
    MenuInflater inflater = getMenuInflater();
    inflater.inflate(R.menu.my_menu, menu);
    return super.onCreateOptionsMenu(menu);
}

@Override
public boolean onOptionsItemSelected(MenuItem item) {
    if(item.getItemId() == R.id.delete_all){
        confirmDialog();
    }
    return super.onOptionsItemSelected(item);
}

void confirmDialog(){
    AlertDialog.Builder builder = new AlertDialog.Builder(this);
    builder.setTitle("Delete All?");
    builder.setMessage("Are you sure you want to delete all Data?");
    builder.setPositiveButton("Yes", new DialogInterface.OnClickListener() {
        @Override
        public void onClick(DialogInterface dialogInterface, int i) {
            MyDatabaseHelper myDB = new MyDatabaseHelper(MainActivity.this);
            myDB.deleteAllData();
            //Refresh Activity
            Intent intent = new Intent(MainActivity.this, MainActivity.class);
        }
    });
}
```



DEPARTMENT OF COMPUTER SCIENCE & ENGINEERING

Discover. Learn. Empower.

```
        startActivity(intent);
        finish();
    }
});
builder.setNegativeButton("No", new DialogInterface.OnClickListener() {
    @Override
    public void onClick(DialogInterface dialogInterface, int i) {

    }
});
builder.create().show();
}
}
```

● CustomAdaptor.java-

```
package com.example.booklibraryapp;

import android.app.Activity;
import android.content.Context;
import android.content.Intent;
import android.os.Build;
import android.view.LayoutInflater;
import android.view.View;
import android.view.ViewGroup;
import android.view.animation.Animation;
import android.view.animation.AnimationUtils;
import android.widget.LinearLayout;
import android.widget.TextView;

import androidx.annotation.NonNull;
import androidx.annotation.RequiresApi;
import androidx.cardview.widget.CardView;
import androidx.recyclerview.widget.RecyclerView;

import java.util.ArrayList;

public class CustomAdapter extends RecyclerView.Adapter<CustomAdapter.MyViewHolder> {
```

```
private Context context;
private Activity activity;
private ArrayList book_id, book_title, book_author, book_pages;
```

```
CustomAdapter(Activity activity, Context context, ArrayList book_id, ArrayList
book_title, ArrayList book_author,
ArrayList book_pages){
    this.activity = activity;
    this.context = context;
    this.book_id = book_id;
    this.book_title = book_title;
    this.book_author = book_author;
    this.book_pages = book_pages;
}
```

```
@NonNull
@Override
public MyViewHolder onCreateViewHolder(@NonNull ViewGroup parent, int
viewType) {
    LayoutInflator inflater = LayoutInflator.from(context);
    View view = inflater.inflate(R.layout.my_row, parent, false);
    return new MyViewHolder(view);
}
```

```
@RequiresApi(api = Build.VERSION_CODES.M)
@Override
public void onBindViewHolder(@NonNull final MyViewHolder holder, final
int position) {
    holder.book_id_txt.setText(String.valueOf(book_id.get(position)));
    holder.book_title_txt.setText(String.valueOf(book_title.get(position)));
    holder.book_author_txt.setText(String.valueOf(book_author.get(position)));
    holder.book_pages_txt.setText(String.valueOf(book_pages.get(position)));
    //Recyclerview onClickListener
    holder.mainLayout.setOnClickListener(new View.OnClickListener() {
        @Override
        public void onClick(View view) {
            Intent intent = new Intent(context, UpdateActivity.class);
            intent.putExtra("id", String.valueOf(book_id.get(position)));
            intent.putExtra("title", String.valueOf(book_title.get(position)));
            intent.putExtra("author", String.valueOf(book_author.get(position)));
            intent.putExtra("pages", String.valueOf(book_pages.get(position)));
        }
    });
}
```



DEPARTMENT OF COMPUTER SCIENCE & ENGINEERING

Discover. Learn. Empower.

```
        activity.startActivityForResult(intent, 1);
    }
});

@Override
public int getItemCount() {
    return book_id.size();
}

class MyViewHolder extends RecyclerView.ViewHolder {

    TextView book_id_txt, book_title_txt, book_author_txt, book_pages_txt;
    LinearLayout mainLayout;

    MyViewHolder(@NonNull View itemView) {
        super(itemView);
        book_id_txt = itemView.findViewById(R.id.book_id_txt);
        book_title_txt = itemView.findViewById(R.id.book_title_txt);
        book_author_txt = itemView.findViewById(R.id.book_author_txt);
        book_pages_txt = itemView.findViewById(R.id.book_pages_txt);
        mainLayout = itemView.findViewById(R.id.mainLayout);
        //Animate Recyclerview
        Animation translate_anim = AnimationUtils.loadAnimation(context,
R.anim.translate_anim);
        mainLayout.setAnimation(translate_anim);
    }

}

}
```



DEPARTMENT OF COMPUTER SCIENCE & ENGINEERING

Discover. Learn. Empower.

● MyDataBaseHelper.java-

```
package com.example.booklibraryapp;

import android.content.ContentValues;
import android.content.Context;
import android.database.Cursor;
import android.database.sqlite.SQLiteDatabase;
import android.database.sqlite.SQLiteOpenHelper;
import android.util.Log;
import android.widget.Toast;
import androidx.annotation.Nullable;

class MyDatabaseHelper extends SQLiteOpenHelper {

    private Context context;
    private static final String DATABASE_NAME = "BookLibrary.db";
    private static final int DATABASE_VERSION = 1;

    private static final String TABLE_NAME = "my_library";
    private static final String COLUMN_ID = "_id";
    private static final String COLUMN_TITLE = "book_title";
    private static final String COLUMN_AUTHOR = "book_author";
    private static final String COLUMN_PAGES = "book_pages";

    MyDatabaseHelper(@Nullable Context context) {
        super(context, DATABASE_NAME, null, DATABASE_VERSION);
        this.context = context;
    }

    @Override
    public void onCreate(SQLiteDatabase db) {
        String query = "CREATE TABLE " + TABLE_NAME +
                " (" + COLUMN_ID + " INTEGER PRIMARY KEY
        AUTOINCREMENT, " +
                COLUMN_TITLE + " TEXT, " +
                COLUMN_AUTHOR + " TEXT, " +
                COLUMN_PAGES + " INTEGER);";
        db.execSQL(query);
    }

    @Override
```



DEPARTMENT OF COMPUTER SCIENCE & ENGINEERING

Discover. Learn. Empower.

```
public void onUpgrade(SQLiteDatabase db, int i, int i1) {  
    db.execSQL("DROP TABLE IF EXISTS " + TABLE_NAME);  
    onCreate(db);  
}  
  
void addBook(String title, String author, int pages){  
    SQLiteDatabase db = this.getWritableDatabase();  
    ContentValues cv = new ContentValues();  
  
    cv.put(COLUMN_TITLE, title);  
    cv.put(COLUMN_AUTHOR, author);  
    cv.put(COLUMN_PAGES, pages);  
    long result = db.insert(TABLE_NAME,null, cv);  
    if(result == -1){  
        Toast.makeText(context, "Failed", Toast.LENGTH_SHORT).show();  
    }else {  
        Toast.makeText(context, "Added Successfully!",  
        Toast.LENGTH_SHORT).show();  
    }  
}  
  
Cursor readAllData(){  
    String query = "SELECT * FROM " + TABLE_NAME;  
    SQLiteDatabase db = this.getReadableDatabase();  
  
    Cursor cursor = null;  
    if(db != null){  
        cursor = db.rawQuery(query, null);  
    }  
    return cursor;  
}  
  
void updateData(String row_id, String title, String author, String pages){  
    SQLiteDatabase db = this.getWritableDatabase();  
    ContentValues cv = new ContentValues();  
    cv.put(COLUMN_TITLE, title);  
    cv.put(COLUMN_AUTHOR, author);  
    cv.put(COLUMN_PAGES, pages);  
  
    long result = db.update(TABLE_NAME, cv, "_id=?", new String[]{row_id});  
    if(result == -1){
```



DEPARTMENT OF COMPUTER SCIENCE & ENGINEERING

Discover. Learn. Empower.

```
        Toast.makeText(context, "Failed", Toast.LENGTH_SHORT).show();
    }else {
        Toast.makeText(context, "Updated Successfully!",
Toast.LENGTH_SHORT).show();
    }

}

void deleteOneRow(String row_id){
    SQLiteDataBase db = this.getWritableDatabase();
    long result = db.delete(TABLE_NAME, "_id=?", new String[]{row_id});
    if(result == -1){
        Toast.makeText(context, "Failed to Delete.",
Toast.LENGTH_SHORT).show();
    }else{
        Toast.makeText(context, "Successfully Deleted.",
Toast.LENGTH_SHORT).show();
    }
}

void deleteAllData(){
    SQLiteDataBase db = this.getWritableDatabase();
    db.execSQL("DELETE FROM " + TABLE_NAME);
}

}
```



DEPARTMENT OF COMPUTER SCIENCE & ENGINEERING

Discover. Learn. Empower.

● AddActivity.java-

```
package com.example.booklibraryapp;

import androidx.appcompat.app.AppCompatActivity;

import android.os.Bundle;
import android.view.View;
import android.widget.Button;
import android.widget.EditText;

public class AddActivity extends AppCompatActivity {

    EditText title_input, author_input, pages_input;
    Button add_button;

    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_add);

        title_input = findViewById(R.id.title_input);
        author_input = findViewById(R.id.author_input);
        pages_input = findViewById(R.id.pages_input);
        add_button = findViewById(R.id.add_button);
        add_button.setOnClickListener(new View.OnClickListener() {
            @Override
            public void onClick(View view) {
                MyDatabaseHelper myDB = new MyDatabaseHelper(AddActivity.this);
                myDB.addBook(title_input.getText().toString().trim(),
                            author_input.getText().toString().trim(),
                            Integer.valueOf(pages_input.getText().toString().trim()));
            }
        });
    }
}
```



DEPARTMENT OF COMPUTER SCIENCE & ENGINEERING

Discover. Learn. Empower.

● **UpdateActivity.java-**

```
package com.example.booklibraryapp;

import androidx.appcompat.app.ActionBar;
import androidx.appcompat.app.AlertDialog;
import androidx.appcompat.app.AppCompatActivity;

import android.content.DialogInterface;
import android.os.Bundle;
import android.util.Log;
import android.view.View;
import android.widget.Button;
import android.widget.EditText;
import android.widget.Toast;

public class UpdateActivity extends AppCompatActivity {

    EditText title_input, author_input, pages_input;
    Button update_button, delete_button;

    String id, title, author, pages;

    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_update);

        title_input = findViewById(R.id.title_input2);
        author_input = findViewById(R.id.author_input2);
        pages_input = findViewById(R.id.pages_input2);
        update_button = findViewById(R.id.update_button);
        delete_button = findViewById(R.id.delete_button);

        //First we call this
        getAndSetIntentData();

        //Set actionBar title after getAndSetIntentData method
        ActionBar ab = getSupportActionBar();
        if (ab != null) {
            ab.setTitle(title);
        }
    }
}
```



DEPARTMENT OF COMPUTER SCIENCE & ENGINEERING

Discover. Learn. Empower.

```
}

update_button.setOnClickListener(new View.OnClickListener() {
    @Override
    public void onClick(View view) {
        //And only then we call this
        MyDatabaseHelper myDB = new
MyDatabaseHelper(UpdateActivity.this);
        title = title_input.getText().toString().trim();
        author = author_input.getText().toString().trim();
        pages = pages_input.getText().toString().trim();
        myDB.updateData(id, title, author, pages);
    }
});

delete_button.setOnClickListener(new View.OnClickListener() {
    @Override
    public void onClick(View view) {
        confirmDialog();
    }
});

}

void getAndSetIntentData(){
    if(getIntent().hasExtra("id") && getIntent().hasExtra("title") &&
        getIntent().hasExtra("author") && getIntent().hasExtra("pages")){
        //Getting Data from Intent
        id = getIntent().getStringExtra("id");
        title = getIntent().getStringExtra("title");
        author = getIntent().getStringExtra("author");
        pages = getIntent().getStringExtra("pages");

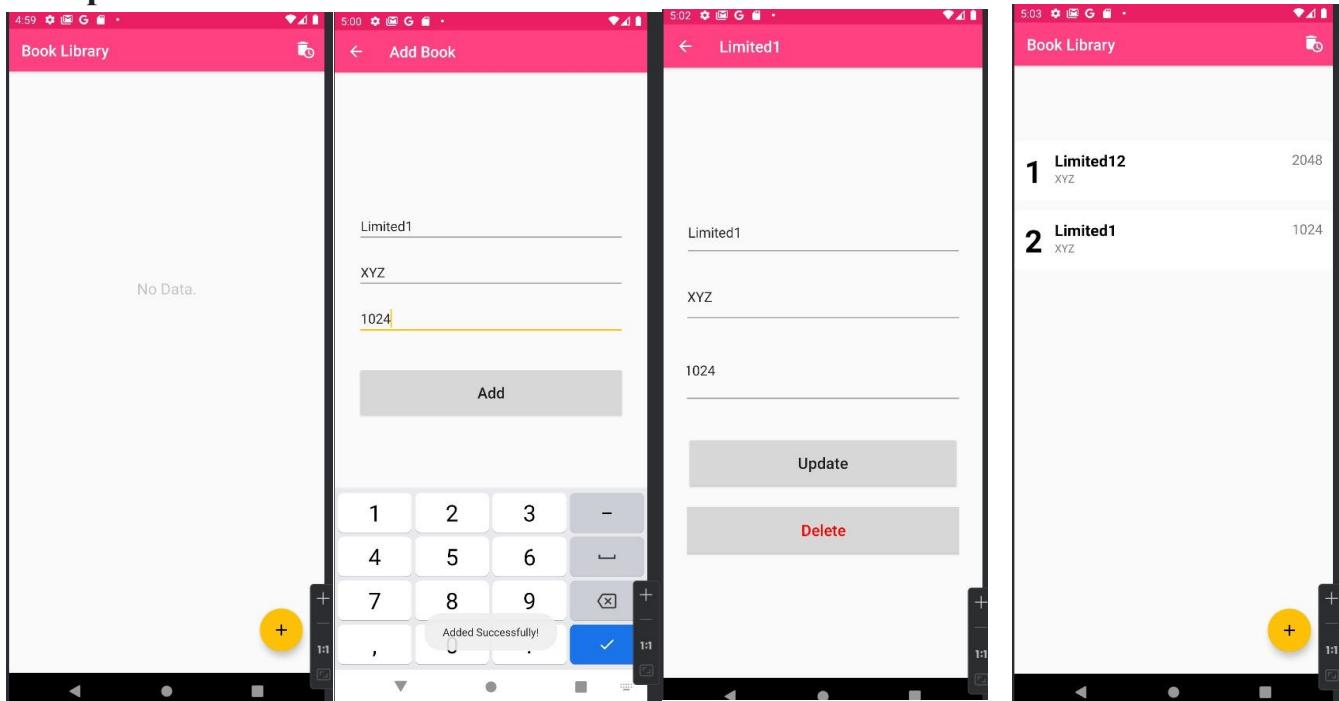
        //Setting Intent Data
        title_input.setText(title);
        author_input.setText(author);
        pages_input.setText(pages);
        Log.d("stev", title+" "+author+" "+pages);
    }else{
        Toast.makeText(this, "No data.", Toast.LENGTH_SHORT).show();
    }
}
```

```

void confirmDialog(){
    AlertDialog.Builder builder = new AlertDialog.Builder(this);
    builder.setTitle("Delete " + title + " ?");
    builder.setMessage("Are you sure you want to delete " + title + " ?");
    builder.setPositiveButton("Yes", new DialogInterface.OnClickListener() {
        @Override
        public void onClick(DialogInterface dialogInterface, int i) {
            MyDatabaseHelper myDB = new
            MyDatabaseHelper(UpdateActivity.this);
            myDB.deleteOneRow(id);
            finish();
        }
    });
    builder.setNegativeButton("No", new DialogInterface.OnClickListener() {
        @Override
        public void onClick(DialogInterface dialogInterface, int i) {
    });
    builder.create().show();
}
}

```

Output:





DEPARTMENT OF COMPUTER SCIENCE & ENGINEERING

Discover. Learn. Empower.

Result:

Successful creation of android app using user registration.

Learning outcomes:

- Successful USE of your Android development environment.
- Project Workspace.
- Configuration button Completion.
- How to add icons,join activities and much more.



DEPARTMENT OF COMPUTER SCIENCE & ENGINEERING

Discover. Learn. Empower.

Experiment No. : 3.3

Student Name: Tanisha Singh	UID: 21BCS9213
Branch: BE CSE	Section: 21BCS_CC_646(B)
Subject Name: Mobile Application Development with Lab	Semester: Sixth
Date of Performance: 29/03/2024	Subject Code: 21CSH-355

Aim: Design the Android application using menus and action bar

Objective: The objective of designing an Android application using menus and the action bar is to create a user-friendly and consistent interface that allows users to navigate, access functionality, and perform actions efficiently.

Input/Apparatus Used:

- 1. Integrated Development Environment (IDE) Android Studio:** The official IDE for Android development. Download and install Android Studio from the official website: [Android Studio](#).
- 2. Android SDK:** The Android Software Development Kit (SDK) is essential for developing Android applications. Android Studio usually comes bundled with the SDK, but you may need to update it through the SDK Manager within Android Studio.
- 3. Java Development Kit (JDK):** Android apps are primarily written in Java or Kotlin. Make sure you have the Java Development Kit installed. Android Studio supports JDK. You can download it from the Oracle website: [Java SE Downloads](#).
- 4. Android Virtual Device (AVD) or Physical Android Device:** You need a device to test your Android application. You can use an emulator (AVD) that comes with Android Studio or a physical Android device connected to your computer.

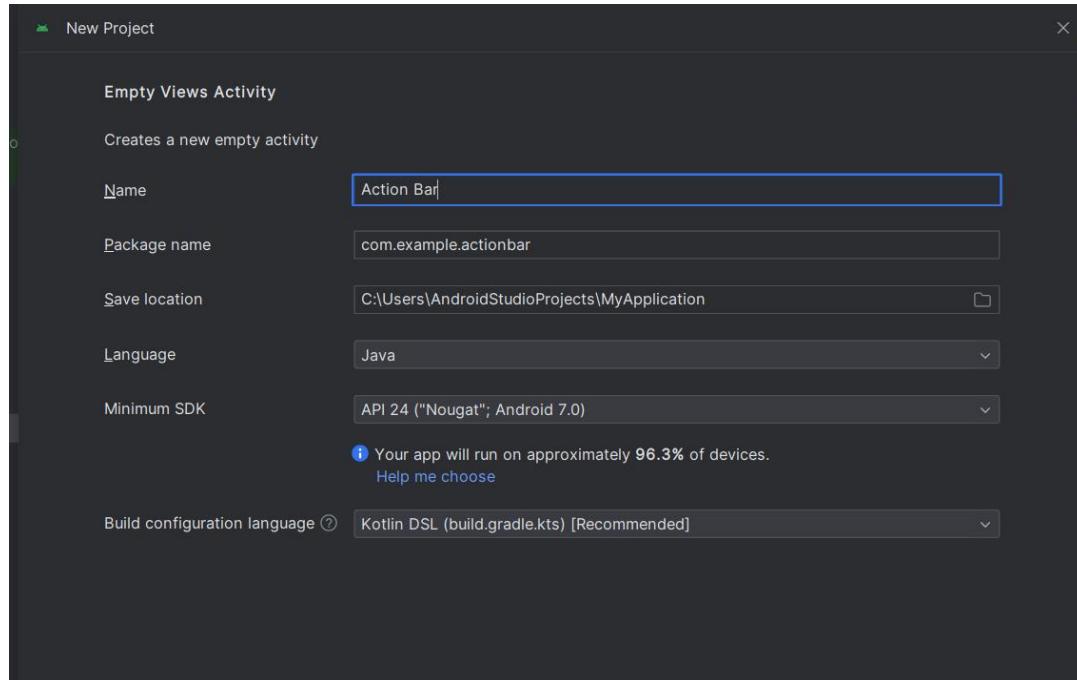


DEPARTMENT OF COMPUTER SCIENCE & ENGINEERING

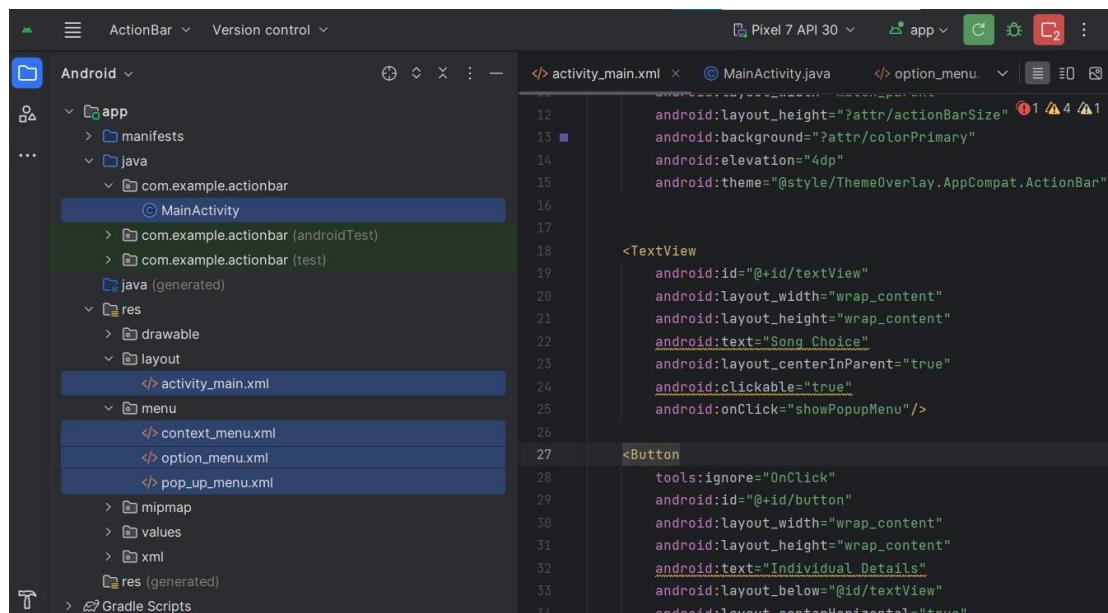
Discover. Learn. Empower.

Script and Output:

Step1. Click on the **New Project** in the Android studio and Select **Empty Views Activity** option.



Step2. Open main **Java** and **XML** file by double clicking it.





DEPARTMENT OF COMPUTER SCIENCE & ENGINEERING

Discover. Learn. Empower.

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

    <androidx.appcompat.widget.Toolbar
        android:id="@+id/toolbar"
        android:layout_width="match_parent"
        android:layout_height="?attr/actionBarSize"
        android:background="?attr/colorPrimary"
        android:elevation="4dp"
        android:theme="@style/ThemeOverlay.AppCompat.ActionBar" />

    <TextView
        android:id="@+id/textView"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="Song Choice"
        android:layout_centerInParent="true"
        android:clickable="true"
        android:onClick="showPopupMenu"/>

    <Button
        tools:ignore="OnClick"
        android:id="@+id/button"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="Individual Details"
        android:layout_below="@+id/textView"
        android:layout_centerHorizontal="true"
        android:layout_marginTop="16dp"
        android:clickable="true"/>
</RelativeLayout>
```



DEPARTMENT OF COMPUTER SCIENCE & ENGINEERING

Discover. Learn. Empower.

● **MainActivity.java-**

```
package com.example.actionbar;

import android.os.Bundle;
import android.util.Log;
import android.view.ContextMenu;
import android.view.Menu;
import android.view.MenuItem;
import android.view.View;
import android.widget.Button;
import android.widget.PopupMenu;
import android.widget.TextView;
import androidx.appcompat.app.AppCompatActivity;
import androidx.appcompat.widget.Toolbar;

public class MainActivity extends AppCompatActivity {
    private static final int CONTEXT_ITEM_1_ID = 1001;
    private static final int CONTEXT_ITEM_2_ID = 1002;
    private static final int CONTEXT_ITEM_3_ID = 1003;

    TextView textView;
    Button button;
    Toolbar toolbar;

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

        textView = findViewById(R.id.textView);
        button = findViewById(R.id.button);
        toolbar = findViewById(R.id.toolbar);
        setSupportActionBar(toolbar);

        // Registering the textView for context menu
        registerForContextMenu(textView);
        // Registering the button for context menu
        registerForContextMenu(button);
    }

    @Override
```



DEPARTMENT OF COMPUTER SCIENCE & ENGINEERING

Discover. Learn. Empower.

```
public boolean onCreateOptionsMenu(Menu menu) {
    getMenuInflater().inflate(R.menu.option_menu, menu);
    return true;
}

@Override
public boolean onOptionsItemSelected(MenuItem item) {
    int id = item.getItemId();

    if (id == R.id.action_settings) {
        // Handle settings action
        Log.d("OptionsMenu", "Settings clicked");
        return true;
    } else if (id == R.id.action_refresh) {
        // Handle refresh action
        Log.d("OptionsMenu", "Refresh clicked");
        return true;
    }

    return super.onOptionsItemSelected(item);
}

@Override
public void onCreateContextMenu(ContextMenu menu, View v,
        ContextMenu.ContextMenuInfo menuInfo) {
    super.onCreateContextMenu(menu, v, menuInfo);
    getMenuInflater().inflate(R.menu.context_menu, menu);

    // Set the IDs for the context menu items programmatically
    if (v.getId() == R.id.textView) {

menu.findItem(R.id.context_item1).setIcon(R.drawable.ic_launcher_background);

menu.findItem(R.id.context_item2).setIcon(R.drawable.ic_launcher_background);

menu.findItem(R.id.context_item3).setIcon(R.drawable.ic_launcher_background);
    }
}

@Override
```



DEPARTMENT OF COMPUTER SCIENCE & ENGINEERING

Discover. Learn. Empower.

```
public boolean onContextItemSelected(MenuItem item) {  
    int id = item.getItemId();  
  
    switch (id) {  
        case CONTEXT_ITEM_1_ID:  
            // Handle context item 1  
            textView.setText("Tanisha Singh");  
            return true;  
        case CONTEXT_ITEM_2_ID:  
            // Handle context item 2  
            textView.setText("21BCS9213");  
            return true;  
        case CONTEXT_ITEM_3_ID:  
            // Handle context item 3  
            textView.setText("Chandigarh University");  
            return true;  
        default:  
            return super.onContextItemSelected(item);  
    }  
}  
  
// Method to show the popup menu  
public void showPopupMenu(View v) {  
    PopupMenu popupMenu = new PopupMenu(this, v);  
    popupMenu.getMenuInflater().inflate(R.menu.pop_up_menu,  
    popupMenu.getMenu());  
    popupMenu.show();  
}  
}
```

● Option_menu.xml-

```
<?xml version="1.0" encoding="utf-8"?>  
<menu xmlns:android="http://schemas.android.com/apk/res/android">  
    <item  
        android:id="@+id/action_settings" android:title="Settings"  
        android:orderInCategory="100" />  
    <item  
        android:id="@+id/action_refresh" android:title="Refresh"  
        android:orderInCategory="101" />  
</menu>
```



DEPARTMENT OF COMPUTER SCIENCE & ENGINEERING

Discover. Learn. Empower.

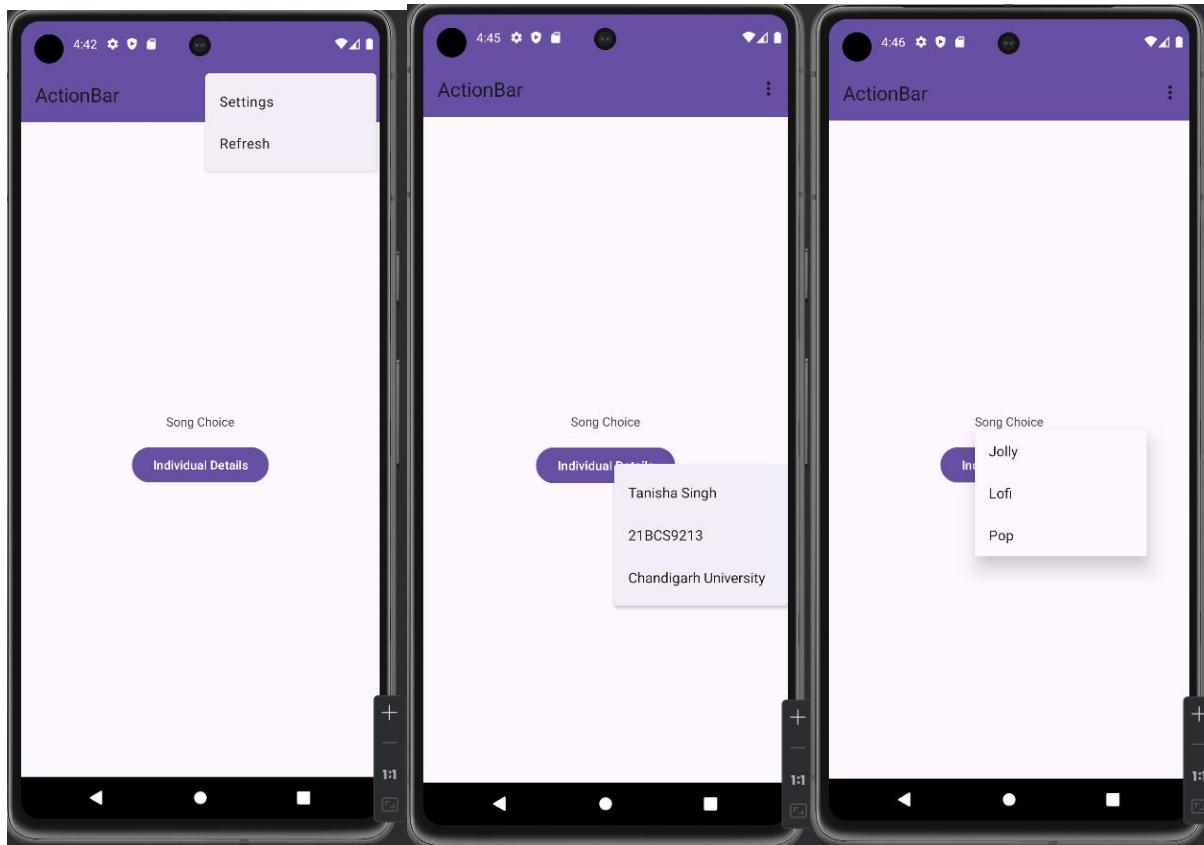
● Context_menu.xml-

```
<?xml version="1.0" encoding="utf-8"?>
<menu xmlns:android="http://schemas.android.com/apk/res/android">
    <item
        android:id="@+id/context_item1" android:title="Tanisha Singh"/>
    <item
        android:id="@+id/context_item2" android:title="21BCS9213" />
    <item
        android:id="@+id/context_item3" android:title="Chandigarh University" />
</menu>
```

● Pop_menu.xml-

```
<?xml version="1.0" encoding="utf-8"?>
<menu xmlns:android="http://schemas.android.com/apk/res/android">
    <item
        android:id="@+id/popup_item1" android:title="Jolly" />
    <item
        android:id="@+id/popup_item2" android:title="Lofi" />
    <item
        android:id="@+id/popup_item3" android:title="Pop" />
</menu>
```

Output:



Result:

Successful creation of android app using Fragments.

Learning outcomes:

- Successful USE of your Android development environment.
- Project Workspace.
- Configuration button Completion.
- How to add Action Bar and using menus.