ACHARYA INSTITUTE OF GRADUATE STUDIES

(NAAC Re-Accredited ‘A+’ & Affiliated to Bengaluru City University)

Soladevanahalli, Bengaluru-560107

DEPARTMENT OF COMPUTER APPLICATION
ACADEMIC YEAR 2023-24 (EVEN)

LAB MANUAL

Subject: Mobile Application Development
Department: BCA
Semester: VI Semester
Subject Coordinators: Rajeshwari Shetty, Smitha Shivaswamy

Mobile Application Development Lab

1 Creating “Hello world” Application.

2 Creating an application that displays message based on the screen orientation.

3 Create an application to develop Login window using UI controls.

4 Create an application to implement new activity using explicit intent, implicit intent and content provider.

5 Create an application that displays custom designed Opening Screen.

6 Create an UI with all views.

7 Create menu in Application

8 Read/ write the Local data.

9 Create / Read / Write data with database (SQLite).

10 Create an application to send SMS and receive SMS

11 Create an application to send an e-mail.

12 Display Map based on the Current/given location.

13 Create a sample application with login module(check user name and password) On successful login change Textview “Login Successful”. On login fail alert using Toast “login fail”

14 Learn to deploy Android applications.

Program 1:

Creating “Hello world” Application.

Steps to start new project:

1. Click Start → Android Studio,
2. Welcome to Android Studio dialog box will appear.
3. Click New Project, the New Project Dialog box appears.
4. Choose Empty Views Activity then click Next.
5. Specify the Name of your project, Select the Language as Java, and Select Minimum SDK as “API 29(“Q”; Android 10.0)” or any latest version.
6. Click Finish Button.

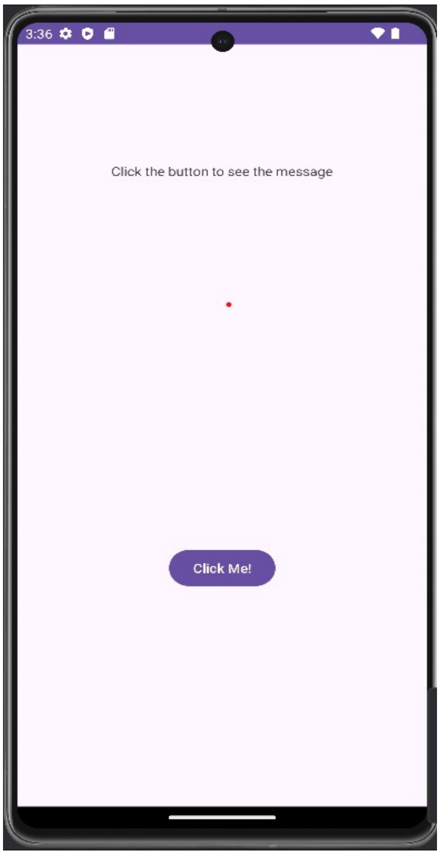
Design:

Design the layout in activity_main.xml file

1. Add a text view and button to the screen by dragging it from the palette to the screen layout.

2. Align the design elements by either in attributes or in the screen.

The figure below illustrates the design of the application



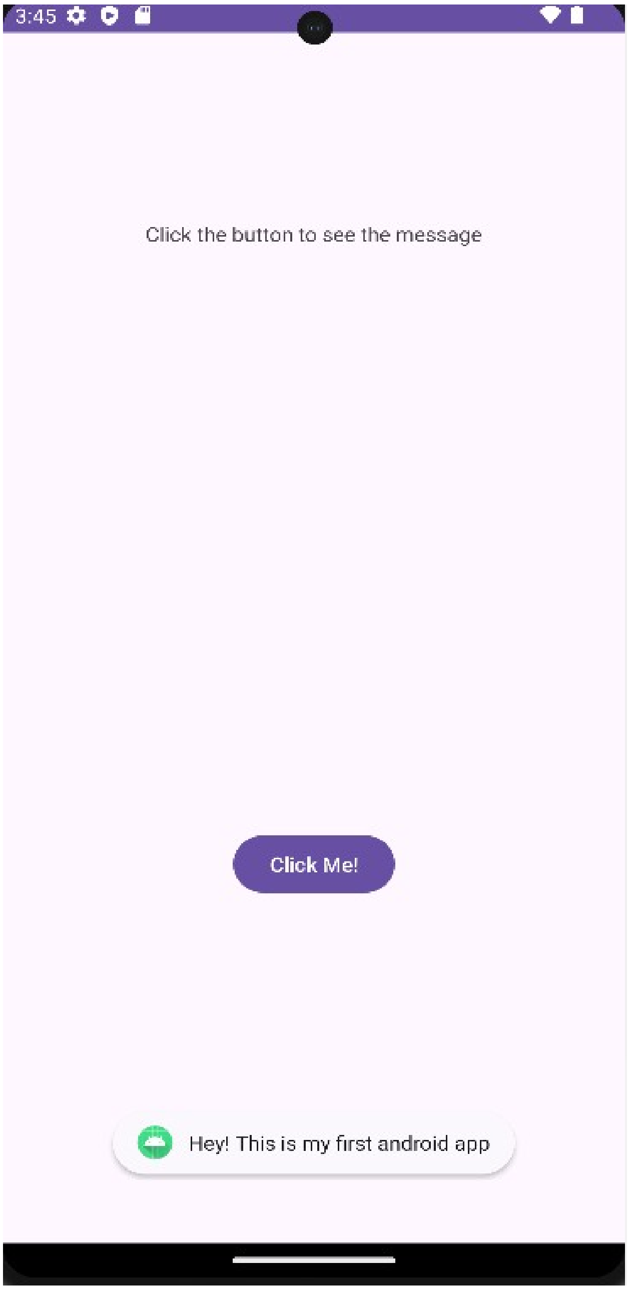
Activity_main.xml :

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

    <TextView
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="Click the button to see the          message"
        app:layout_constraintBottom_toTopOf="@+id/button"
        app:layout_constraintEnd_toEndOf="parent"
        app:layout_constraintStart_toStartOf="parent"
        app:layout_constraintTop_toTopOf="parent"
        app:layout_constraintVertical_bias="0.244" />

    <Button
        android:id="@+id/button"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_marginBottom="240dp"
        android:text="Click Me!"
        app:layout_constraintBottom_toBottomOf="parent"
        app:layout_constraintEnd_toEndOf="parent"
        app:layout_constraintStart_toStartOf="parent" />

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

MainActivity.java:

```java
package com.example1.lab_program1;

import androidx.appcompat.app.AppCompatActivity;

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

public class MainActivity extends AppCompatActivity {

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

               Button b;
        b = findViewById(R.id.button);
        b.setOnClickListener(new View.OnClickListener(){
            @Override
            public void onClick(View v) {
                Toast.makeText(MainActivity.this,"Hey! This is my first android app",Toast.LENGTH_SHORT).show();
            }
        });
    }
}
```

Output: Click Run app or shift+f10 to see the output



3:45

Click the button to see the message

Hey! This is my first android app

Program 2

2. Creating an application that displays message based on the screen orientation.

Steps to start new project:

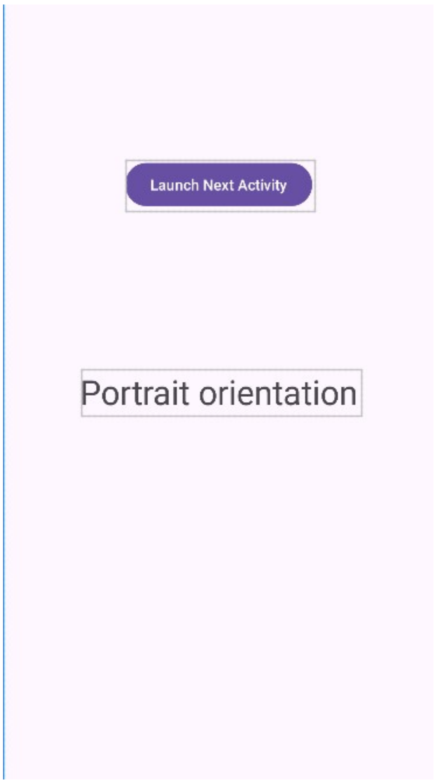
1. Click Start → Android Studio,
2. Welcome to Android Studio dialog box will appear.
3. Click New Project, the New Project Dialog box appears.
4. Choose Empty Views Activity then click Next.
5. Specify the Name of your project, Select the Language as Java, and Select Minimum SDK as "API 29("Q"; Android 10.0)" or any latest version.
6. Click Finish Button.

Design:

Activity_main.xml file :

1. Select the button from palette and drop it on the layout, set the text as Launch Next activity.
2. Similarly select a text view and drop it on the layout below the button, set the text as Portrait orientation.

Pictorial representation of the design



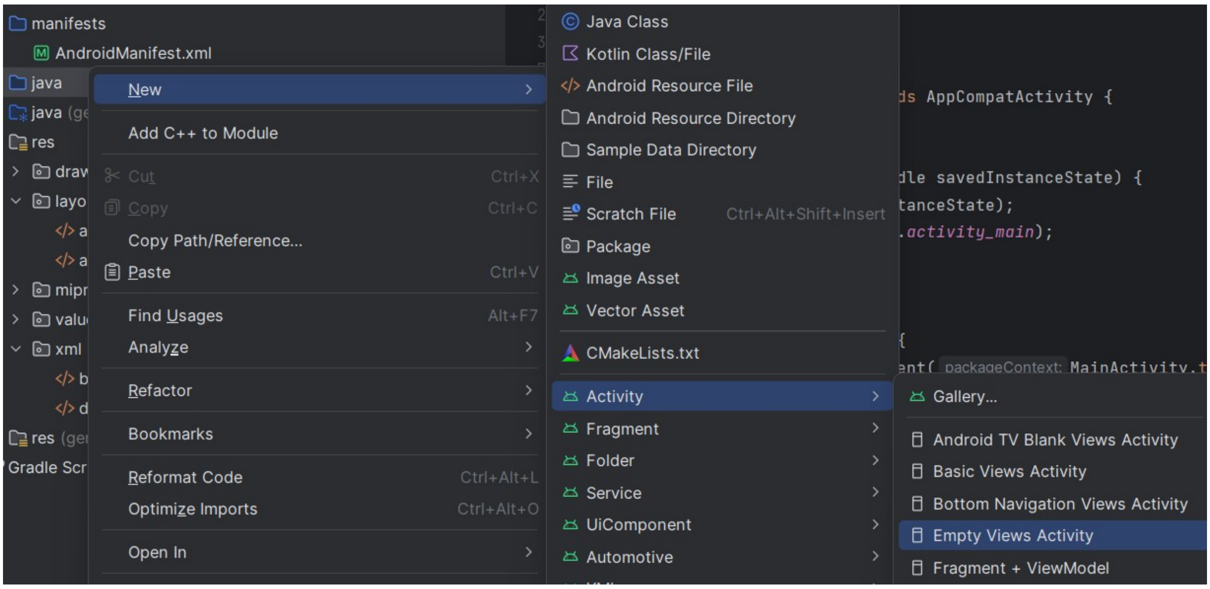
Create another layout :

Follow the steps below to create 2nd layout of this program

1. Right Click java file on left hand side sidebar

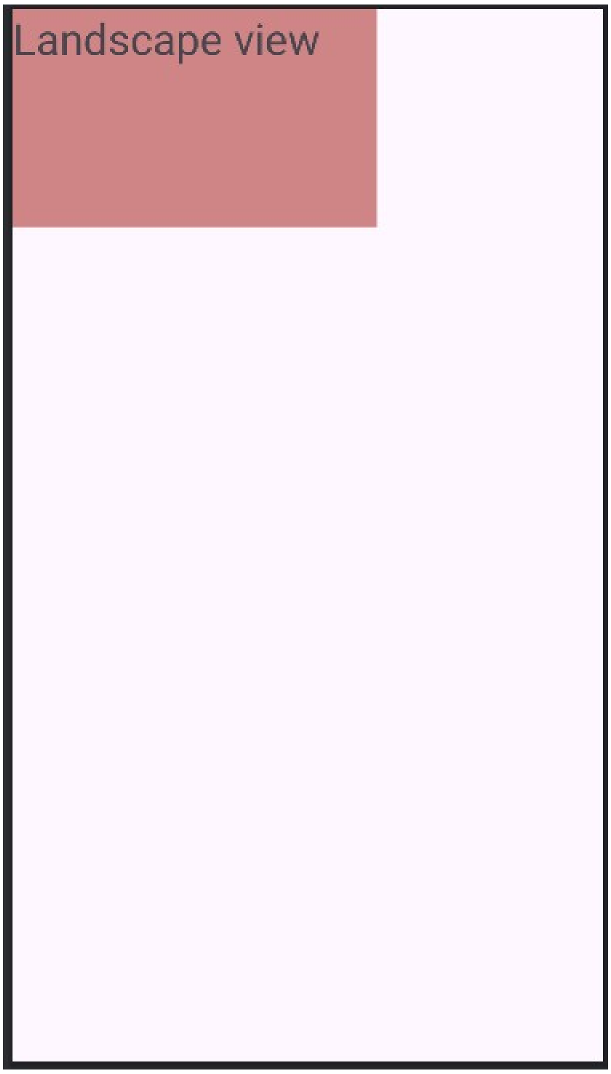
2. New > Activity > Empty views Activity

3. A New Android Activity pop-up window opens, type the activity name as ‘MainActivity2’ and click the finish button



- Select a text view from palette and drop it on screen
- set the text as “Landscape view”

Pictorial representation of the design



Landscape view

Activity_main.xml:

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

    <TextView
        android:id="@+id/textView"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_marginStart="32dp"
        android:layout_marginTop="32dp"
        android:layout_marginEnd="32dp"
        android:layout_marginBottom="32dp"
        android:text="Portrait orientation"
        android:textSize="32sp"
        app:layout_constraintBottom_toBottomOf="parent"
        app:layout_constraintEnd_toEndOf="parent"
        app:layout_constraintStart_toStartOf="parent"
        app:layout_constraintTop_toTopOf="parent"
       tools:ignore="HardcodedText"/>

    <Button
        android:id="@+id/button"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:onClick="onClick"
        android:text="Launch Next Activity"
        app:layout_constraintBottom_toTopOf="@+id/textView"
        app:layout_constraintEnd_toEndOf="parent"
        app:layout_constraintStart_toStartOf="parent"
        app:layout_constraintTop_toTopOf="parent"
       tools:ignore="HardcodedText"/>

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

Activity_main2.xml

```
<?xml version="1.0" encoding="utf-8"?>
<androidx.constraintlayout.widget.ConstraintLayout
xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:app="http://schemas.android.com/apk/res-auto"
    xmlns:tools="http://schemas.android.com/tools"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:background="#CF8585"
    tools:context=".MainActivity2">

    <TextView
        android:id="@+id/textView2"
        android:layout_width="223dp"
        android:layout_height="120dp"
        android:layout_marginEnd="32dp"
        android:layout_marginBottom="32dp"
        android:text="Landscape view"
        android:textSize="30sp"
        app:layout_constraintBottom_toBottomOf="parent"
        app:layout_constraintEnd_toEndOf="parent"
        app:layout_constraintHorizontal_bias="1.0"
        app:layout_constraintStart_toStartOf="parent"
        app:layout_constraintTop_toTopOf="parent"
        app:layout_constraintVertical_bias="1.0"
        tools:ignore="HardcodedText" />
</androidx.constraintlayout.widget.ConstraintLayout>
```

AndroidManifest.xml

```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">

    <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.Labprogram2_2ndscreen"
        tools:targetApi="31">
        <activity
            android:name=".MainActivity2"
            android:exported="false"
            android:screenOrientation="landscape"
            />

        <activity
            android:name=".MainActivity"
            android:exported="true"
            android:screenOrientation="portrait">
            <intent-filter>
                <action android:name="android.intent.action.MAIN" />

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

</manifest>
```

MainActivity.java:

Import content. Intent, view. View from android library

And create the onclick method

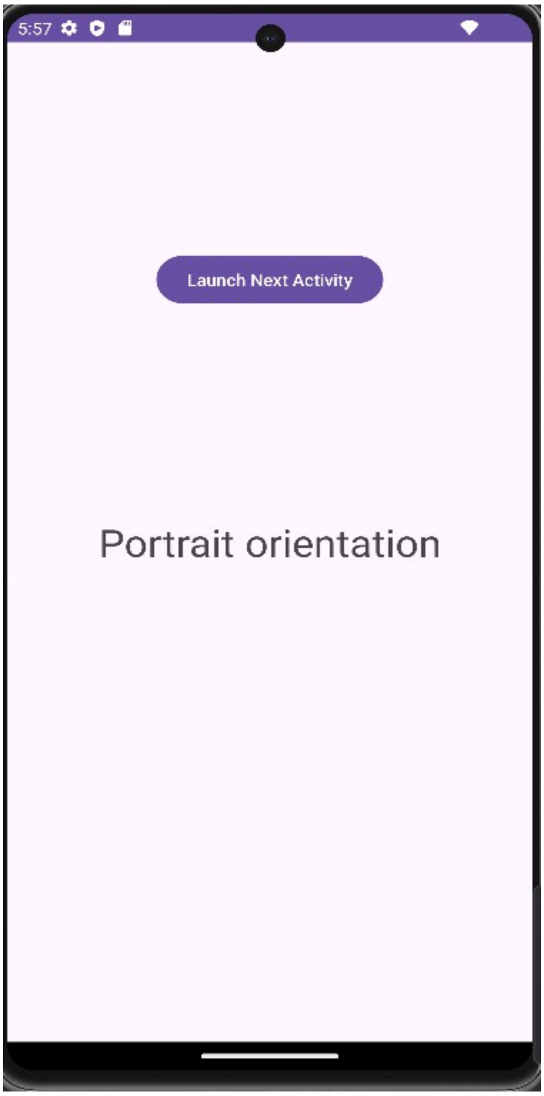
```
package com.example1.labprogram2_2ndscreen;

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

public class MainActivity extends AppCompatActivity {

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

    public void onClick(View v){
        Intent intent = new Intent(MainActivity.this, MainActivity2.class);
        startActivity(intent);
    }
}
```

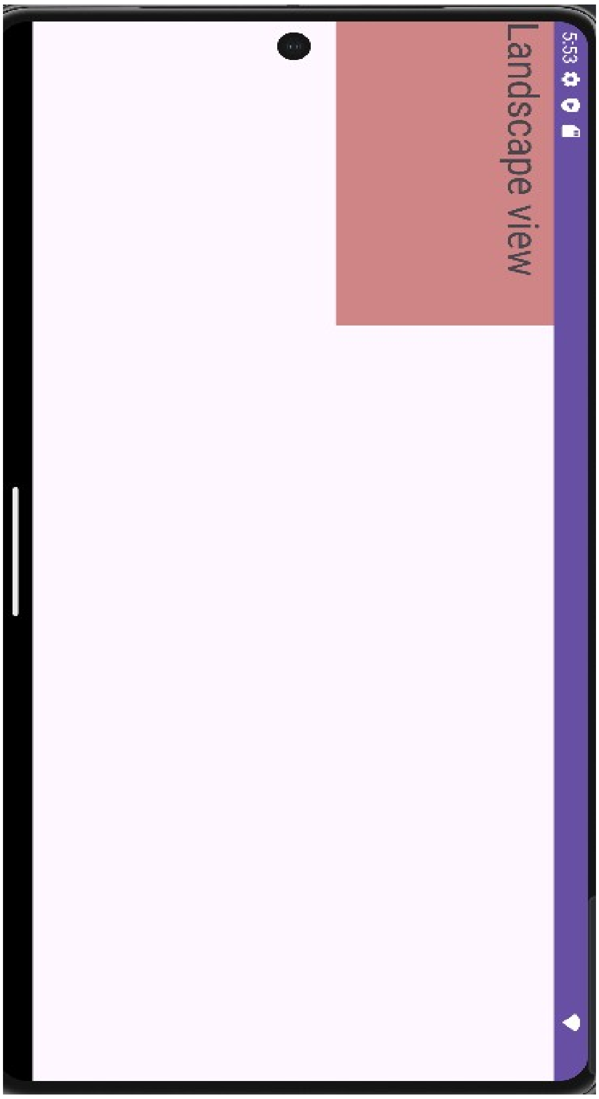
MainActivity2.java:

```
package com.example1.labprogram2_2ndscreen;

import androidx.appcompat.app.AppCompatActivity;

import android.os.Bundle;

public class MainActivity2 extends AppCompatActivity {

    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_main2);
    }
}
```

Output:

Click Run or shift+f10 to run the app on emulator


5:57
Launch Next Activity
Portrait orientation



Landscape view

3. Creating an application to develop Login window using UI controls

Step 1: create new project

1. Click New Project, the New Project Dialog box appears.

2. Choose Empty Views Activity then click Next.

3. Specify the Name of your project, Select the Language as Java, and Select the Minimum SDK as API 16 (“Jelly Bean”, Android 4.1).

Click Finish Button.

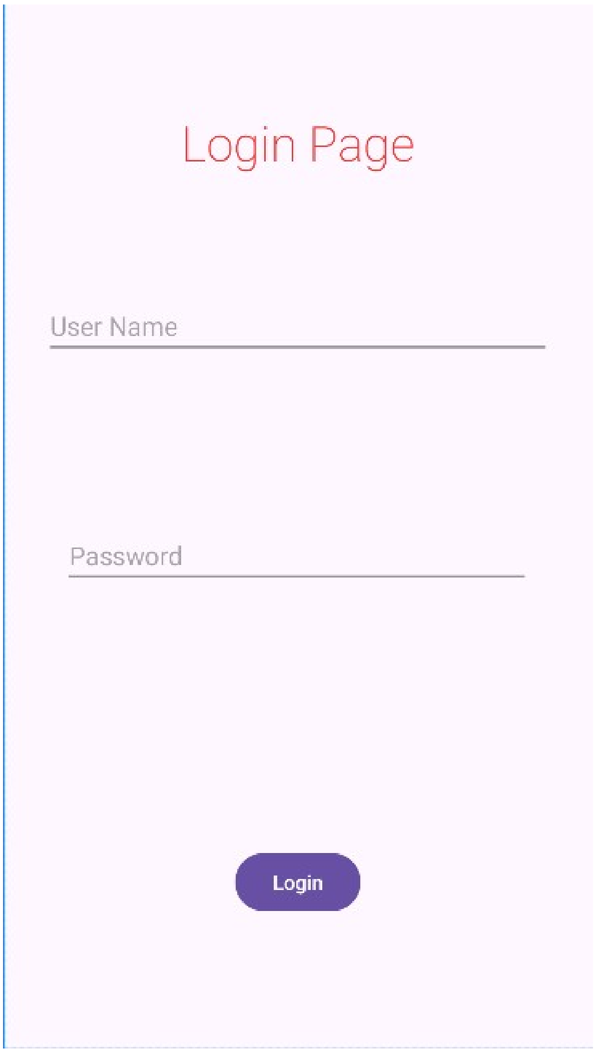
Step 2: Design the Activity_main.xml screen

Create one Textview, two EditText box and a Button resource in activity_main.xml and update the following code

Activity_main.xml

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

   <!-- <LinearLayout
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:gravity="center"
        android:layout_centerInParent="true"
        android:orientation="vertical"
        android:background="@drawable/br_outer"
        android:padding="30dp"
       />-->


    <com.google.android.material.textfield.TextInputEditText
        android:id="@+id/textInputEditText"
        android:layout_width="353dp"
        android:layout_height="45dp"
        android:hint="User Name"
        app:layout_constraintBottom_toTopOf="@+id/textInputEditText2"
        app:layout_constraintEnd_toEndOf="parent"
        app:layout_constraintStart_toStartOf="parent"
        app:layout_constraintTop_toTopOf="parent"
        app:layout_constraintVertical_bias="0.644" />

    <Button
        android:id="@+id/button"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_marginBottom="92dp"
        android:text="Login"
        app:layout_constraintBottom_toBottomOf="parent"
        app:layout_constraintEnd_toEndOf="parent"
        app:layout_constraintStart_toStartOf="parent" />

    <com.google.android.material.textfield.TextInputEditText
        android:id="@+id/textInputEditText2"
        android:layout_width="326dp"
        android:layout_height="45dp"
        android:layout_marginBottom="180dp"
        android:hint="Password"
        app:layout_constraintBottom_toTopOf="@+id/button"
        app:layout_constraintEnd_toEndOf="parent"
        app:layout_constraintHorizontal_bias="0.494"
        app:layout_constraintStart_toStartOf="parent" />

    <TextView
        android:id="@+id/textView"
        android:layout_width="201dp"
        android:layout_height="49dp"
        android:fontFamily="sans-serif-black"
        android:lineHeight="2dp"
        android:text="Login Page"
        android:textAlignment="center"
        android:textAppearance="@style/TextAppearance.AppCompat.Body1"
        android:textColor="#EF3535"
        android:textDirection="inherit"
        android:textFontWeight="@android:integer/config_shortAnimTime"
        android:textSize="34sp"
        android:textStyle="bold"
        app:layout_constraintBottom_toTopOf="@+id/textInputEditText"
        app:layout_constraintEnd_toEndOf="parent"
        app:layout_constraintStart_toStartOf="parent"
        app:layout_constraintTop_toTopOf="parent" />

   <!--<LinearLayout
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:gravity="center"
        android:layout_centerInParent="true"
        android:orientation="vertical"
        android:background="@drawable/bg_inner"
        android:padding="30dp"
       />
        -->

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

The following figure illustrates the design view of the application.



Step 3: Code

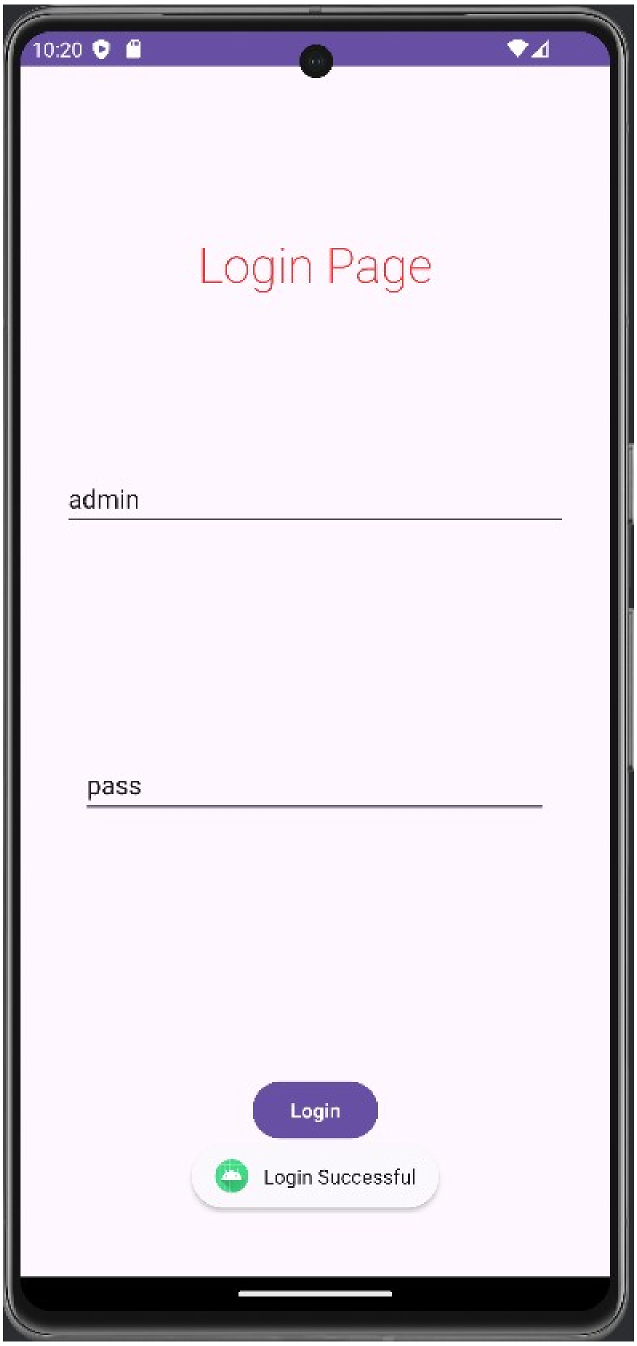
MainActivity.java code

Create two EditText and a Button object, create onclickListener, onClick event for button object and update the following code in MainActivity.java

```
package com.example1.lab_program3;

import androidx.appcompat.app.AppCompatActivity;

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

public class MainActivity extends AppCompatActivity {

    private EditText textInputEditText,textInputEditText2;
    private Button button;


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

        textInputEditText = findViewById(R.id.textInputEditText);
        textInputEditText2 = findViewById(R.id.textInputEditText2);
        button = findViewById(R.id.button);
        button.setOnClickListener(new View.OnClickListener() {
            @Override
            public void onClick(View v) {
                String username = textInputEditText.getText().toString().trim();
                String password = textInputEditText2.getText().toString().trim();

                if (username.equals("admin") && password.equals("pass")) {
                    Toast.makeText(MainActivity.this, "Login Successful", Toast.LENGTH_SHORT).show();

                }
                else
                {
                    Toast.makeText(MainActivity.this, "Invalid username or password",
Toast.LENGTH_SHORT).show();

                }
            }
        });

    }

}
```

Step 4: Output

Output 1:



Login Page

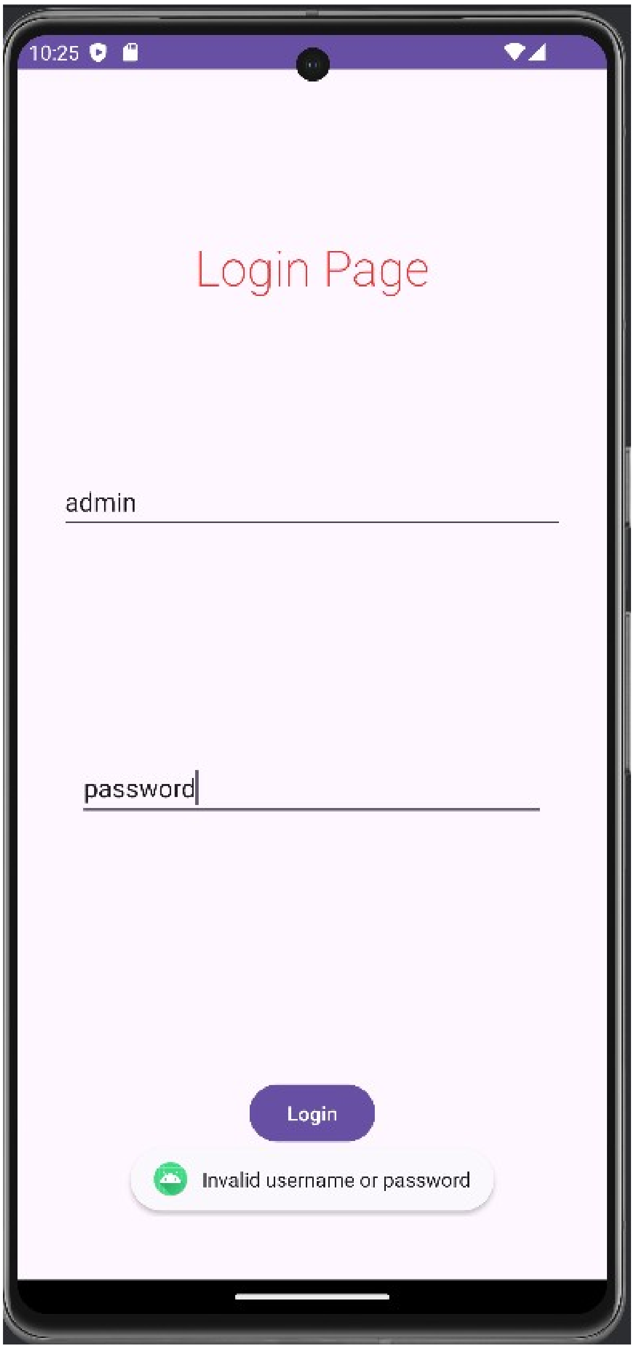
admin

pass

Login

Login Successful

Output 2:



admin

password

Invalid username or password

Program 4

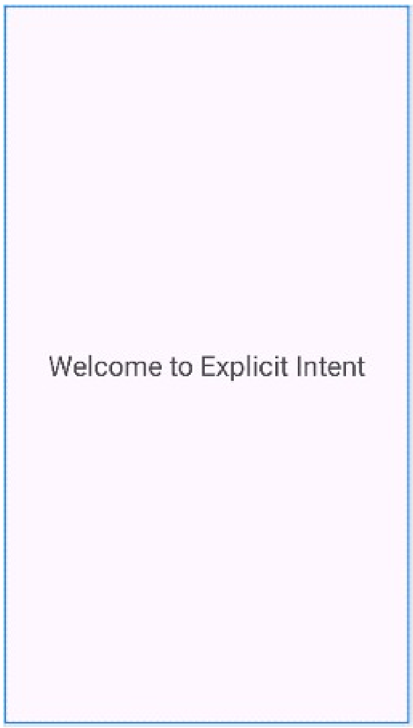
Create an application to implement new activity using explicit intent, implicit intent

and content provider.

1. Click New Project, the New Project Dialog box appears.
2. Choose Empty Views Activity then click Next.
3. Specify the Name of your project, Select the Language as Java, and Select the minimum SDK (Choose Default). Click Finish Button.
4. To create another activity for Explicit Intent, Click File→New→Activity→ Empty Views Activity. A New Android Activity dialog box appears, Specify the Name of the activity as NewActivity then click Finish.
5. Create one TextView resource in activity_new.xml and update the following code

```
<?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:id="@+id/main"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    tools:context=".NewActivity">

    <TextView
        android:id="@+id/textView"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="Welcome to Explicit Intent"
        android:textSize="28sp"
        app:layout_constraintBottom_toBottomOf="parent"
        app:layout_constraintEnd_toEndOf="parent"
        app:layout_constraintStart_toStartOf="parent"
        app:layout_constraintTop_toTopOf="parent" />
</androidx.constraintlayout.widget.ConstraintLayout>
```

The following figure illustrates the design view of the application(activity_new.xml).

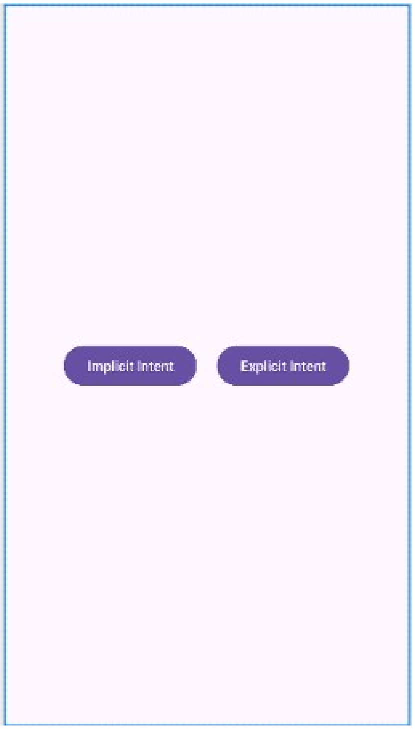


Welcome to Explicit Intent

6. Add Linear Layout in the design view
7. Add two Button resource in activity_main.xml and update the following code.

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


    <Button
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="Implicit Intent"
        android:onClick="onImplicitButtonClicked"
        />

    <Button
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="Explicit Intent"
        android:onClick="onExplicitButtonClicked"
         />

    </LinearLayout>
```

The following figure illustrates the design view of the application(activity_main.xml).

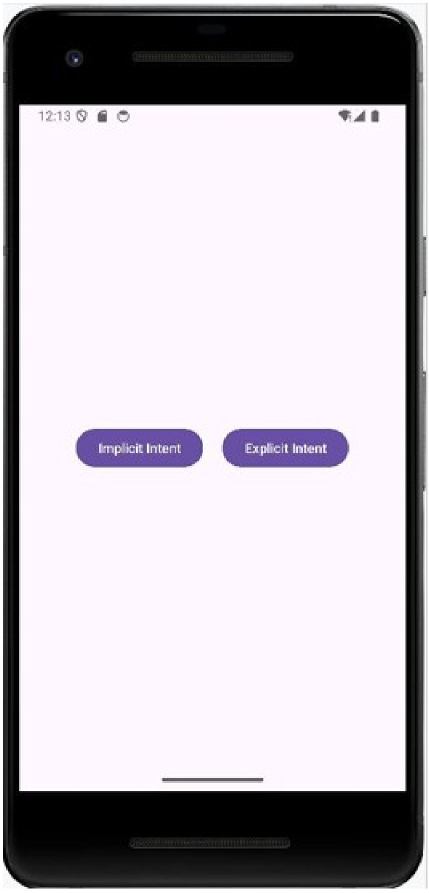


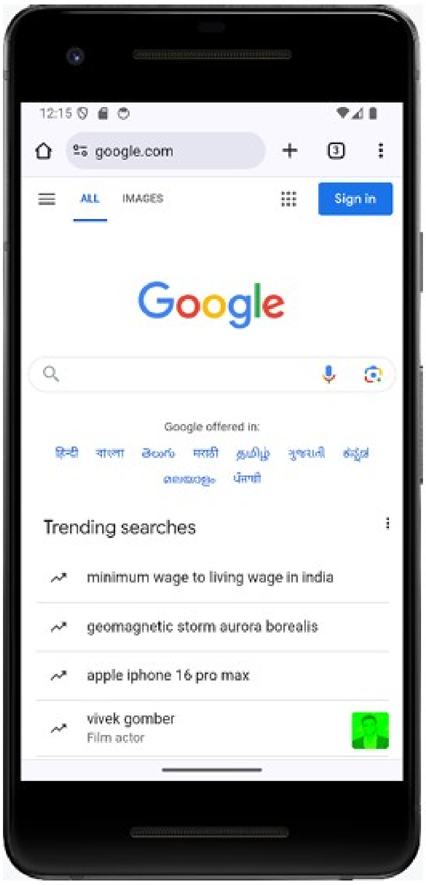
8. Add two events named as onImplicitButtonClicked, onExplicitButtonClicked and update the following code in MainActivity.java

```
package com.example.app_int_content_provider;

import android.os.Bundle;
import android.content.Intent;
import android.net.Uri;
import android.view.View;

import androidx.activity.EdgeToEdge;
import androidx.appcompat.app.AppCompatActivity;
import androidx.core.graphics.Insets;
import androidx.core.view.ViewCompat;
import androidx.core.view.WindowInsetsCompat;

public class MainActivity extends AppCompatActivity {

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

    }
    public void onImplicitButtonClicked(View view)
    {
        Uri url=Uri.parse("https://www.google.com");
        Intent i=new Intent(Intent.ACTION_VIEW, url);
        startActivity(i);
    }
    public void onExplicitButtonClicked(View view ) {
        Intent i = new Intent(MainActivity.this, NewActivity.class);
        startActivity(i);
    }
}
```

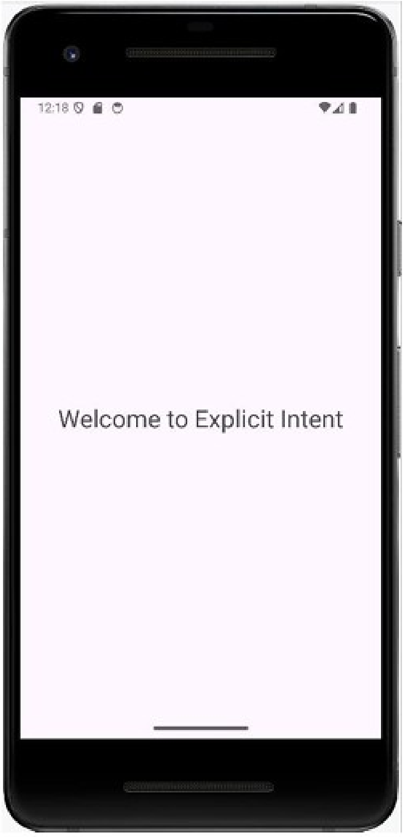
Output

For Implicit Intent





For Explicit Intent



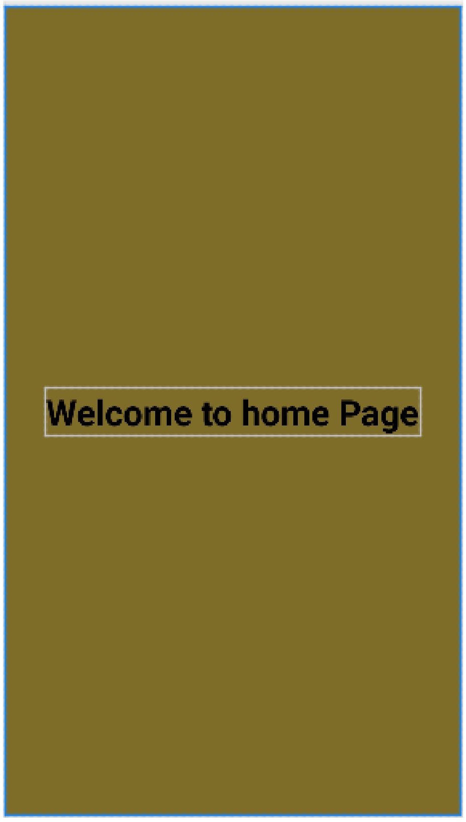
Welcome to Explicit Intent

Program 5: Create an application that displays custom designed Opening Screen

1) Click New Project, the New Project Dialog box appears.
2) Choose Empty Views Activity then click Next.
3) Specify the Name of your project, Select the Language as Java, and Select the minimum SDK as (Leave it to Default) Click Finish Button.
4) To create another activity for Home Page, Right Click App→New→Activity→ Empty Views Activity. A New Android Activity dialog box appears, Specify the Name of the activity as mainScreen then click Finish.
5) Create one TextView resource in activity_mainScreen.xml
6) Delete the constraintLayout and add Relative Layout in the code and update the following code

```xml
<?xml version="1.0" encoding="utf-8"?>
<RelativeLayout
xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:app="http://schemas.android.com/apk/res-auto"
    xmlns:tools="http://schemas.android.com/tools"
    android:id="@+id/main"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    tools:context=".mainScreen"
    android:gravity="center"
    android:background="#7E6C29">

    <TextView
        android:id="@+id/textView"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="Welcome to home Page"
        android:textStyle="bold"
        android:textSize="32sp"
        android:textColor="@color/black" />

</RelativeLayout>
```

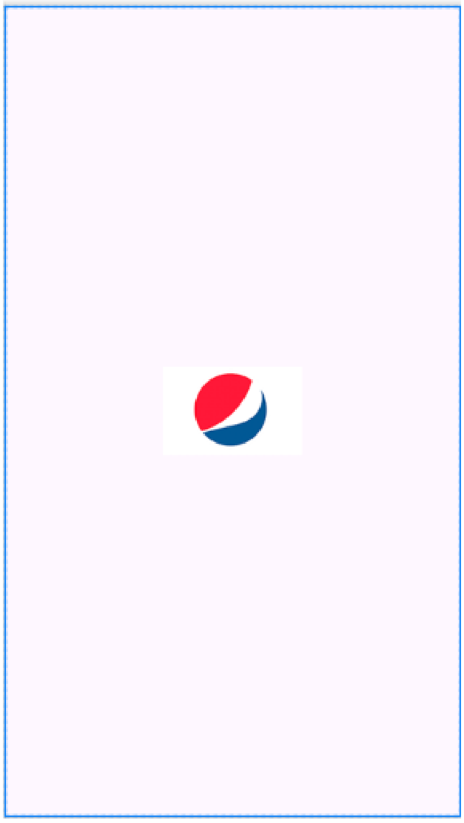
7) The following figure illustrates the design view of the application(activity_mainScreen.xml).


Welcome to home Page

8) To add an ImageView resource: Copy an image and paste it into drawable folder (Right-click Drawable→ Paste the image[img.jpg]).
9) Set an image as src in activity_main.xml
10) Delete the constraint layout and add Relative layout and update the following code.

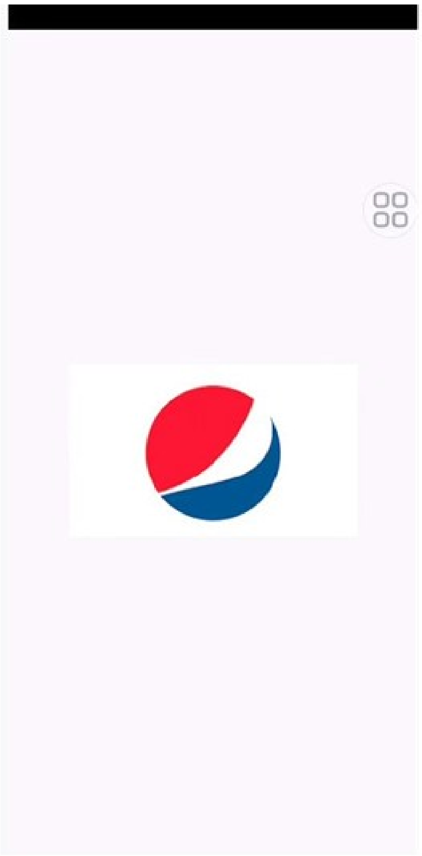
```
<?xml version="1.0" encoding="utf-8"?>
<RelativeLayout xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:app="http://schemas.android.com/apk/res-auto"
    xmlns:tools="http://schemas.android.com/tools"
    android:id="@+id/main"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    tools:context=".MainActivity"
    android:gravity="center">

    <ImageView
        android:id="@+id/imageView"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        app:srcCompat="@drawable/img" />

</RelativeLayout>
```

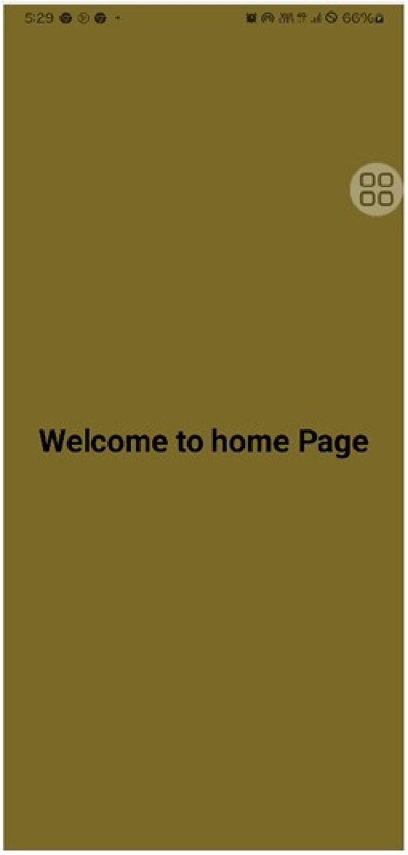
11) The following figure illustrates the design view of the application(activity_main.xml).



12) Update the following code in Mainactivity.java

```
package com.example.customscreen1;

import android.os.Bundle;
import android.content.Intent;
import android.os.Handler;
import android.view.WindowManager;

import androidx.activity.EdgeToEdge;
import androidx.appcompat.app.AppCompatActivity;
import androidx.core.graphics.Insets;
import androidx.core.view.ViewCompat;
import androidx.core.view.WindowInsetsCompat;

public class MainActivity extends AppCompatActivity {
    private static final int SPLASH_SCREEN_TIME_OUT = 2000;
    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        EdgeToEdge.enable(this);
        setContentView(R.layout.activity_main);

getWindow().setFlags(WindowManager.LayoutParams.FLAG_FULLSCREEN,
                WindowManager.LayoutParams.FLAG_FULLSCREEN);
        new Handler().postDelayed(new Runnable() {
            @Override
            public void run() {
                Intent i = new Intent(MainActivity.this,
mainScreen.class);
                startActivity(i);
                finish();
            }
        }, SPLASH_SCREEN_TIME_OUT);
    }
}
```





Welcome to home Page

Program 6

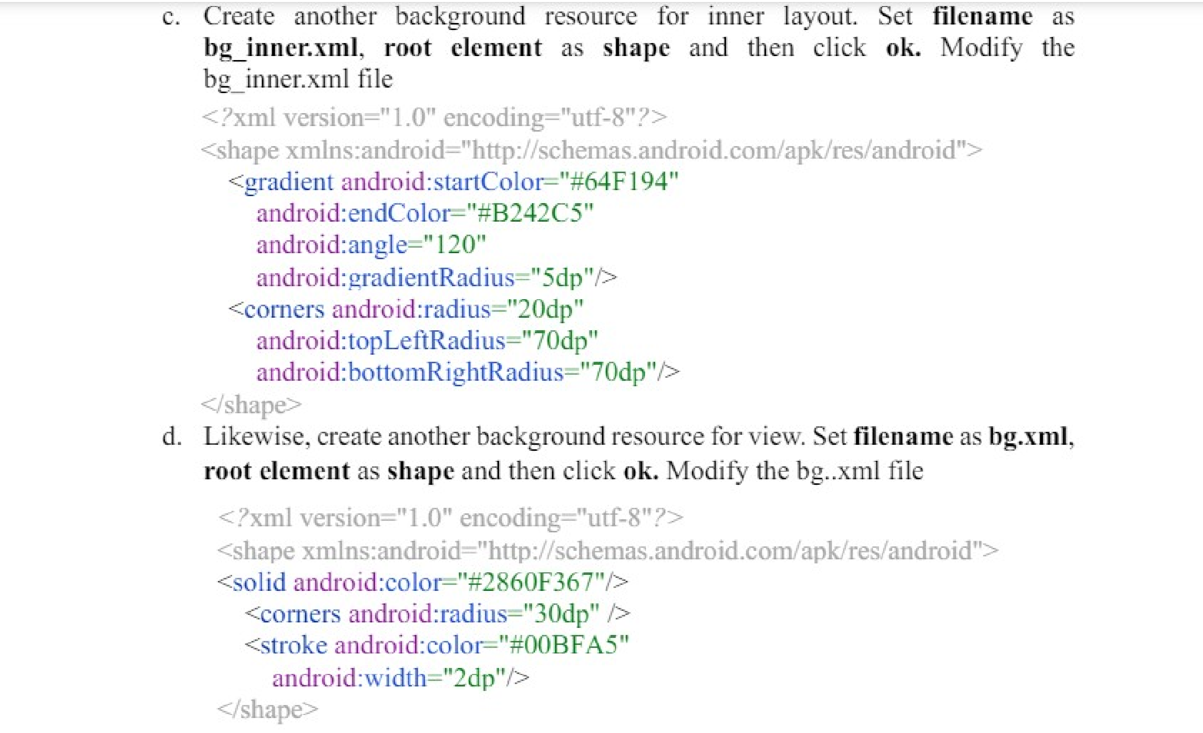
6. Create an UI with all views.

   1. Click **New Project**, the **New Project Dialog box** appears.
   2. Choose **Empty Views Activity** then click **Next**.
   3. Specify the **Name** of your project, Select the **Language** as **Java**, and Select the **Minimum SDK** as **API 16 ("Jelly Bean", Android 4.1)**. Click **Finish** Button.
   4. Create **background** resources(**bg_outer.xml**, **bg_inner.xml, bg.xml**)

      a. To create resource file click **app→res→drawable.** Right click **drawable→New→ Drawable Resource File.** The **New Resource File** dialog box appears.

      b. Set **filename** as **bg_outer.xml**, **root element** as **shape** and then click **ok.** Modify the bg_outer.xml file

```
<?xml version="1.0" encoding="utf-8"?>
<shape xmlns:android="http://schemas.android.com/apk/res/android">
<gradient android:startColor="#64EFAE"
  android:endColor="#84FFFF"
  android:angle="120"
  android:gradientRadius="5dp"/>
  <corners android:radius="20dp"/>
</shape>
```



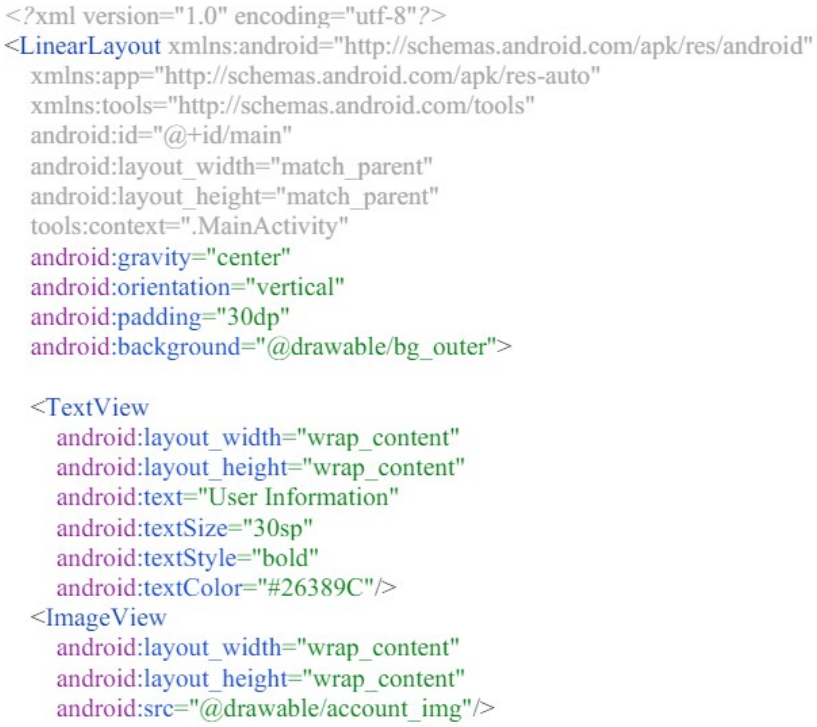
c. Create another background resource for inner layout. Set **filename** as **bg_inner.xml**, **root element** as **shape** and then click **ok.** Modify the bg_inner.xml file

```
<?xml version="1.0" encoding="utf-8"?>
<shape xmlns:android="http://schemas.android.com/apk/res/android">
  <gradient android:startColor="#64F194"
    android:endColor="#B242C5"
    android:angle="120"
    android:gradientRadius="5dp"/>
  <corners android:radius="20dp"
    android:topLeftRadius="70dp"
    android:bottomRightRadius="70dp"/>
</shape>
```

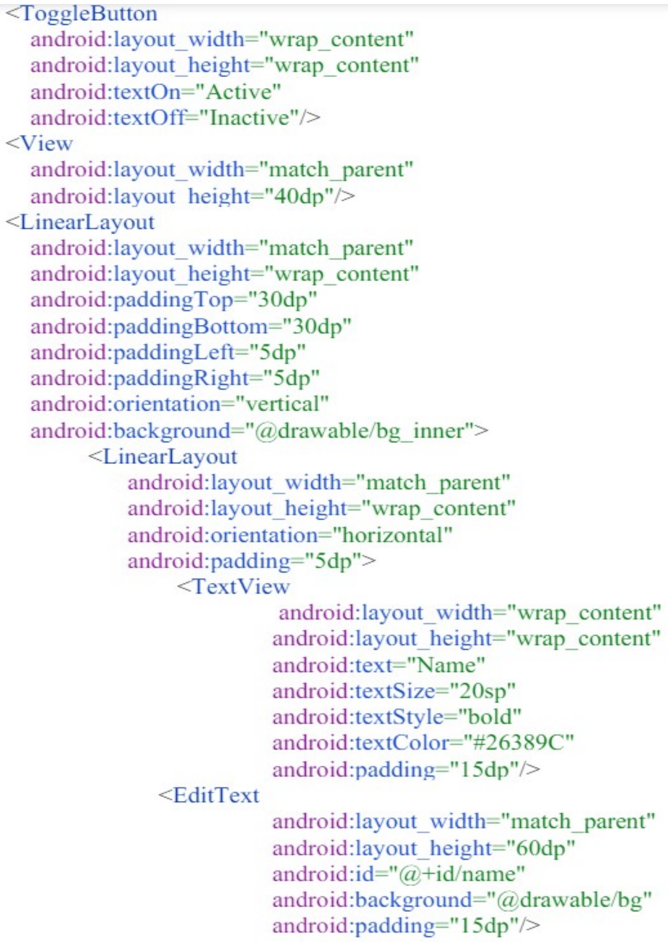
d. Likewise, create another background resource for view. Set **filename** as **bg.xml**, **root element** as **shape** and then click **ok.** Modify the bg..xml file

```
<?xml version="1.0" encoding="utf-8"?>
<shape xmlns:android="http://schemas.android.com/apk/res/android">
<solid android:color="#2860F367"/>
  <corners android:radius="30dp" />
  <stroke android:color="#00BFA5"
    android:width="2dp"/>
</shape>
```

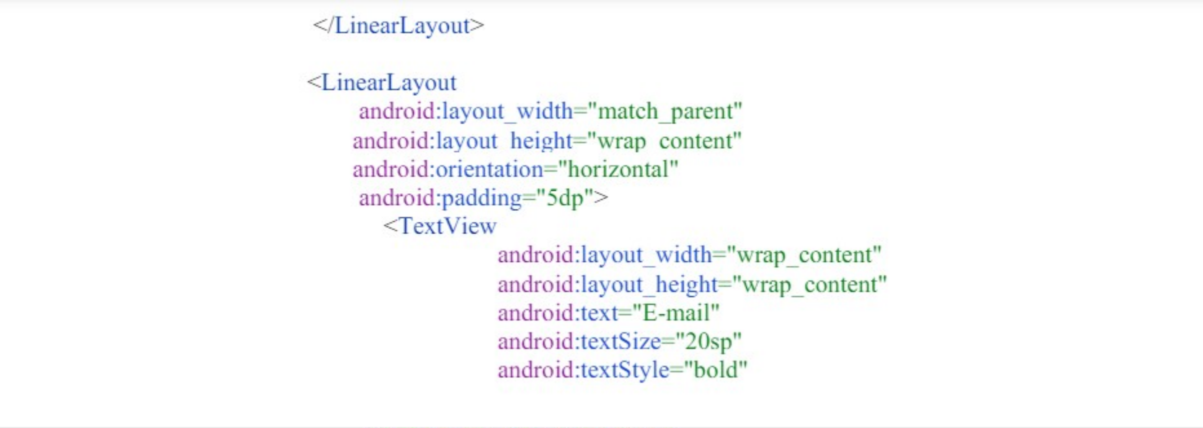
5. Create a **TextView, EditText, ToggleButton, ImageView, RadioGroup, RadioButton, spinner** and **a Button** resource in **activity_main.xml** and update the following code.



```xml
<?xml version="1.0" encoding="utf-8"?>
<LinearLayout xmlns:android="http://schemas.android.com/apk/res/android"
  xmlns:app="http://schemas.android.com/apk/res-auto"
  xmlns:tools="http://schemas.android.com/tools"
  android:id="@+id/main"
  android:layout_width="match_parent"
  android:layout_height="match_parent"
  tools:context=".MainActivity"
  android:gravity="center"
  android:orientation="vertical"
  android:padding="30dp"
  android:background="@drawable/bg_outer">

  <TextView
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:text="User Information"
    android:textSize="30sp"
    android:textStyle="bold"
    android:textColor="#26389C"/>
  <ImageView
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:src="@drawable/account_img"/>
```

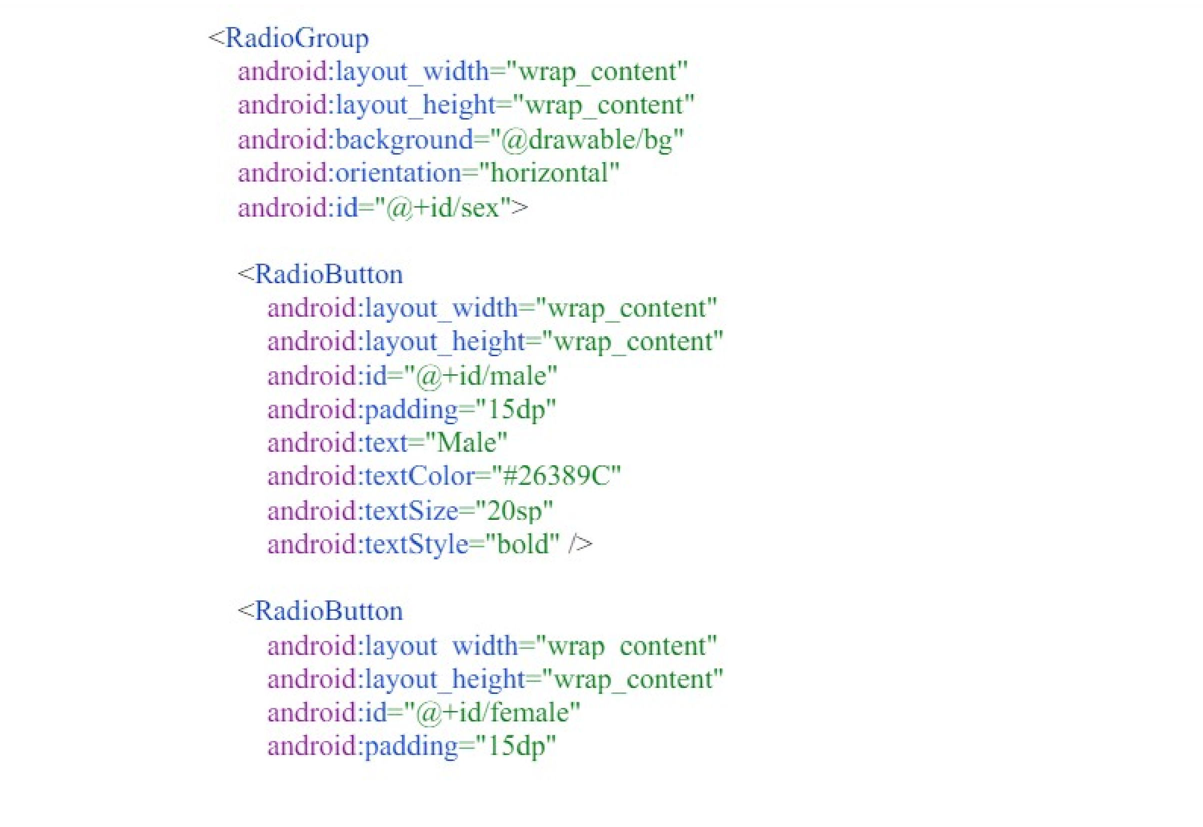


```xml
  <ToggleButton
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:textOn="Active"
    android:textOff="Inactive"/>
  <View
    android:layout_width="match_parent"
    android:layout_height="40dp"/>
  <LinearLayout
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:paddingTop="30dp"
    android:paddingBottom="30dp"
    android:paddingLeft="5dp"
    android:paddingRight="5dp"
    android:orientation="vertical"
    android:background="@drawable/bg_inner">
        <LinearLayout
            android:layout_width="match_parent"
            android:layout_height="wrap_content"
            android:orientation="horizontal"
            android:padding="5dp">
              <TextView
                        android:layout_width="wrap_content"
                       android:layout_height="wrap_content"
                       android:text="Name"
                       android:textSize="20sp"
                       android:textStyle="bold"
                       android:textColor="#26389C"
                       android:padding="15dp"/>
            <EditText
                       android:layout_width="match_parent"
                       android:layout_height="60dp"
                       android:id="@+id/name"
                       android:background="@drawable/bg"
                       android:padding="15dp"/>
```

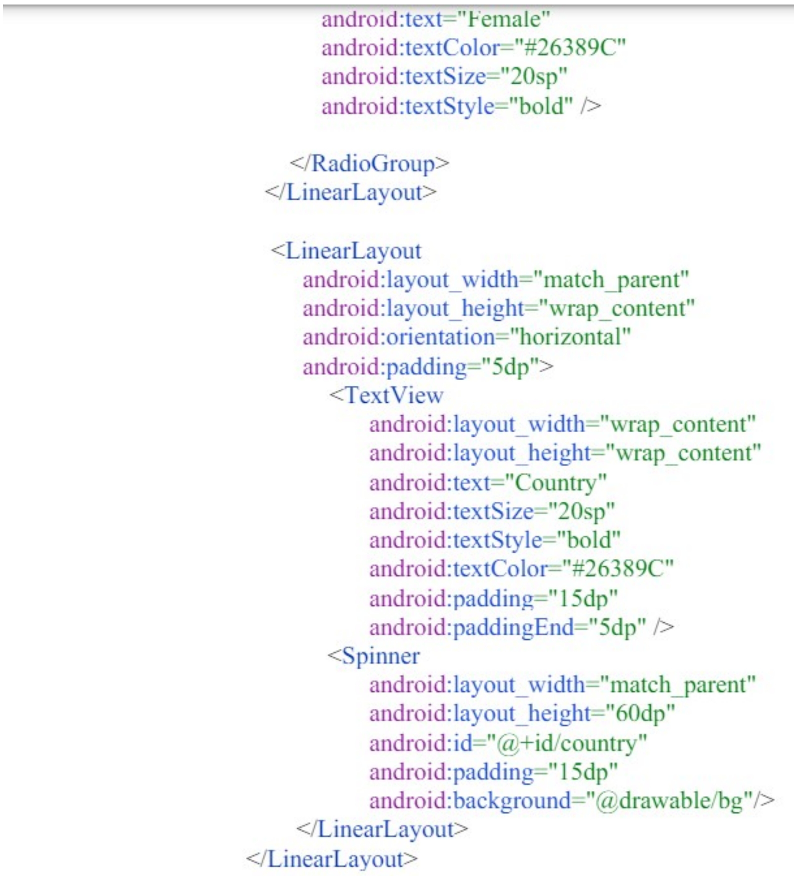


```
</LinearLayout>

<LinearLayout
      android:layout_width="match_parent"
     android:layout  height="wrap  content"
     android:orientation="horizontal"
      android:padding="5dp">
        <TextView
                    android:layout_width="wrap_content"
                    android:layout_height="wrap_content"
                    android:text="E-mail"
                    android:textSize="20sp"
                    android:textStyle="bold"
```

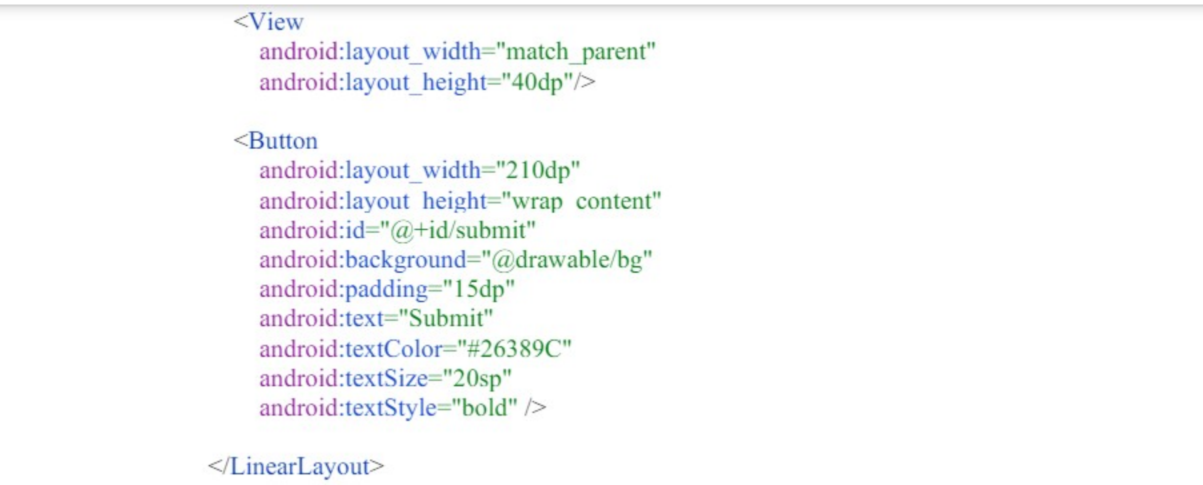
```
         android:textColor="#26389C"
         android:padding="15dp"/>

    <EditText
         android:id="@+id/email"
         android:layout_width="match_parent"
         android:layout  height="60dp"
         android:ems="10"
         android:inputType="textEmailAddress"
         android:background="@drawable/bg"
         android:padding="15dp"/>
</LinearLayout>
<LinearLayout
   android:layout_width="match_parent"
   android:layout  height="wrap  content"
   android:orientation="horizontal"
   android:padding="5dp">
   <TextView
      android:layout_width="wrap_content"
      android:layout_height="wrap_content"
      android:text="Sex"
      android:textSize="20sp"
      android:textStyle="bold"
      android:textColor="#26389C"
      android:padding="15dp"
      android:paddingEnd="40dp"
      />
```



```
<RadioGroup
  android:layout_width="wrap_content"
  android:layout_height="wrap_content"
  android:background="@drawable/bg"
  android:orientation="horizontal"
  android:id="@+id/sex">

  <RadioButton
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:id="@+id/male"
    android:padding="15dp"
    android:text="Male"
    android:textColor="#26389C"
    android:textSize="20sp"
    android:textStyle="bold" />

  <RadioButton
    android:layout width="wrap content"
    android:layout_height="wrap_content"
    android:id="@+id/female"
    android:padding="15dp"
```

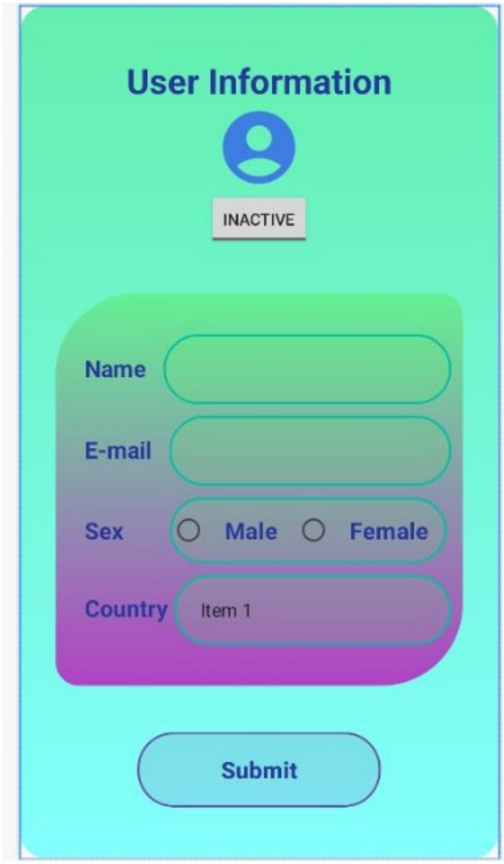


```
        android:text="Female"
        android:textColor="#26389C"
        android:textSize="20sp"
        android:textStyle="bold" />

    </RadioGroup>
  </LinearLayout>

  <LinearLayout
     android:layout_width="match_parent"
     android:layout_height="wrap_content"
     android:orientation="horizontal"
     android:padding="5dp">
       <TextView
           android:layout_width="wrap_content"
           android:layout_height="wrap_content"
           android:text="Country"
           android:textSize="20sp"
           android:textStyle="bold"
           android:textColor="#26389C"
           android:padding="15dp"
           android:paddingEnd="5dp" />
       <Spinner
           android:layout_width="match_parent"
           android:layout_height="60dp"
           android:id="@+id/country"
           android:padding="15dp"
           android:background="@drawable/bg"/>
    </LinearLayout>
</LinearLayout>
```



```
    <View
        android:layout_width="match_parent"
        android:layout_height="40dp"/>

    <Button
        android:layout_width="210dp"
        android:layout_height="wrap_content"
        android:id="@+id/submit"
        android:background="@drawable/bg"
        android:padding="15dp"
        android:text="Submit"
        android:textColor="#26389C"
        android:textSize="20sp"
        android:textStyle="bold" />

</LinearLayout>
```

The following figure illustrates the design view of the application

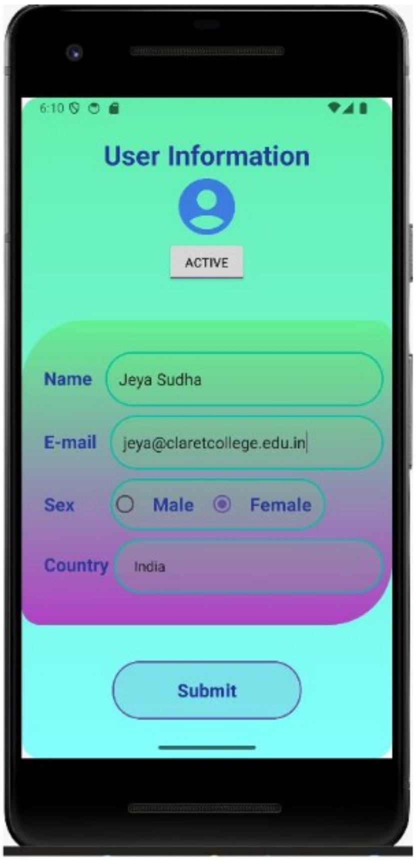


6. Create two **EditText** and a **Button** object, create **clickListener**, **onClick** event for button object and update the following code in **MainActivity.java**

```
package com.example.all  views;

import android.app.Dialog;
import android.content.Context;
import android.content.DialogInterface;
import android.os.Bundle;
import android.view.View;
import android.view.ViewGroup;
import android.widget.ArrayAdapter;
import android.widget.Button;
import android.widget.EditText;
import android.widget.RadioButton;
import android.widget.RadioGroup;
import android.widget.Spinner;
import android.widget.Toast;

import androidx.activity.EdgeToEdge;
import androidx.appcompat.app.AlertDialog;
import androidx.appcompat.app.AppCompatActivity;
import androidx.core.graphics.Insets;
import androidx.core.view.ViewCompat;
import androidx.core.view.WindowInsetsCompat;

public class MainActivity extends AppCompatActivity {
  Button sub;

  @Override
  protected void onCreate(Bundle savedInstanceState) {
    super.onCreate(savedInstanceState);
    EdgeToEdge.enable(this);
    setContentView(R.layout.activity_main);
    Button sub=findViewById(R.id.submit);

    sub.setOnClickListener(new View.OnClickListener() {
      @Override
      public void onClick(View v) {
        showMessage(MainActivity.this,"User Information","Successfully
completed");
      }
    });
   String[] item=new String[]{"India", "Pakisthan", "China", "America",
"England"};
    ArrayAdapter adapter = new ArrayAdapter<>(this,
android.R.layout.simple_spinner_item, item);

adapter.setDropDownViewResource(android.R.layout.simple_spinner_dropdown_i
tem);
    Spinner spinner = findViewById(R.id.country);
    spinner.setAdapter(adapter);
  }


public void showMessage(Context con,String t, String msg)
{
  AlertDialog.Builder builder = new AlertDialog.Builder(con);
  builder.setTitle(t);
  builder.setMessage(msg);
  builder.setPositiveButton("OK", new DialogInterface.OnClickListener() {
    @Override
    public void onClick(DialogInterface dialog, int which) {
      dialog.dismiss();
    }
  });
  builder.show();

}
}
```

7. Click **Run app** or **shift+F10** to execute the application.

Output:

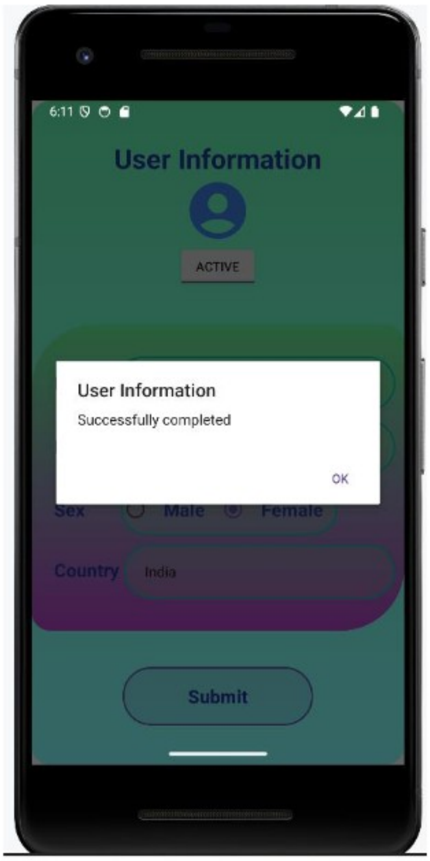


Name: Jeya Sudha

E-mail: jeya@claretcollege.edu.in

Sex: ○ Male ● Female

Country: India



User Information

Successfully completed

Program 7 :

Create menu in Application

1. Click New Project, the New Project Dialog box appears.

2. Choose Empty Views Activity then click Next.

3. Specify the Name of your project, Select the Language as Java, and Select the SDK

as API 24(“Nougat”,Android 7.0).Click Finish Button

4. To create a Menu Resource File: Right-click on the res directory in your Android project, navigate to New Android Resource File, and name the file menus.xml, set Resource Type as Menu and update the following content.

```
<?xml version="1.0" encoding="utf-8"?>
<menu xmlns:app="http://schemas.android.com/apk/res-auto"
    xmlns:android="http://schemas.android.com/apk/res/android">
<item android:id="@+id/php"
    android:title="PHP"/>
    <item
        android:id="@+id/java"
        android:title="Java" />
    <item
        android:id="@+id/csharp"
        android:title="C#" />
</menu>
```

Main Activity.Java

```
package com.example1.labprogram7;

import androidx.appcompat.app.AppCompatActivity;

import android.os.Bundle;
import android.content.Intent;
import android.view.Menu;
import android.view.MenuInflater;
import android.view.MenuItem;
import android.widget.Toast;
import androidx.activity.EdgeToEdge;
import androidx.annotation.NonNull;

import androidx.core.graphics.Insets;
import androidx.core.view.ViewCompat;
import androidx.core.view.WindowInsetsCompat;
```

```
public class MainActivity extends AppCompatActivity {

    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        EdgeToEdge.enable(this);
        setContentView(R.layout.activity_main);
    }
    public boolean onCreatePanelMenu(int featureId, @NonNull Menu menu){
        MenuInflater inflater=getMenuInflater();
        inflater.inflate(R.menu.menus, menu);
        return true;
    }

    public boolean onOptionsItemSelected(@NonNull MenuItem item){
        if(item.getItemId()==R.id.php){
            Toast.makeText(this, "php Page", Toast.LENGTH_SHORT).show();
        }
        if(item.getItemId()==R.id.java){
            Toast.makeText(this, "Java Page", Toast.LENGTH_SHORT).show();
        }
        if(item.getItemId()==R.id.csharp) {
            Toast.makeText(this, "C# Page", Toast.LENGTH_SHORT).show();
        }

        return super.onOptionsItemSelected(item);


    }
}
```

Manifest file:

Set the Uses Permission in AndroidManifest.xml

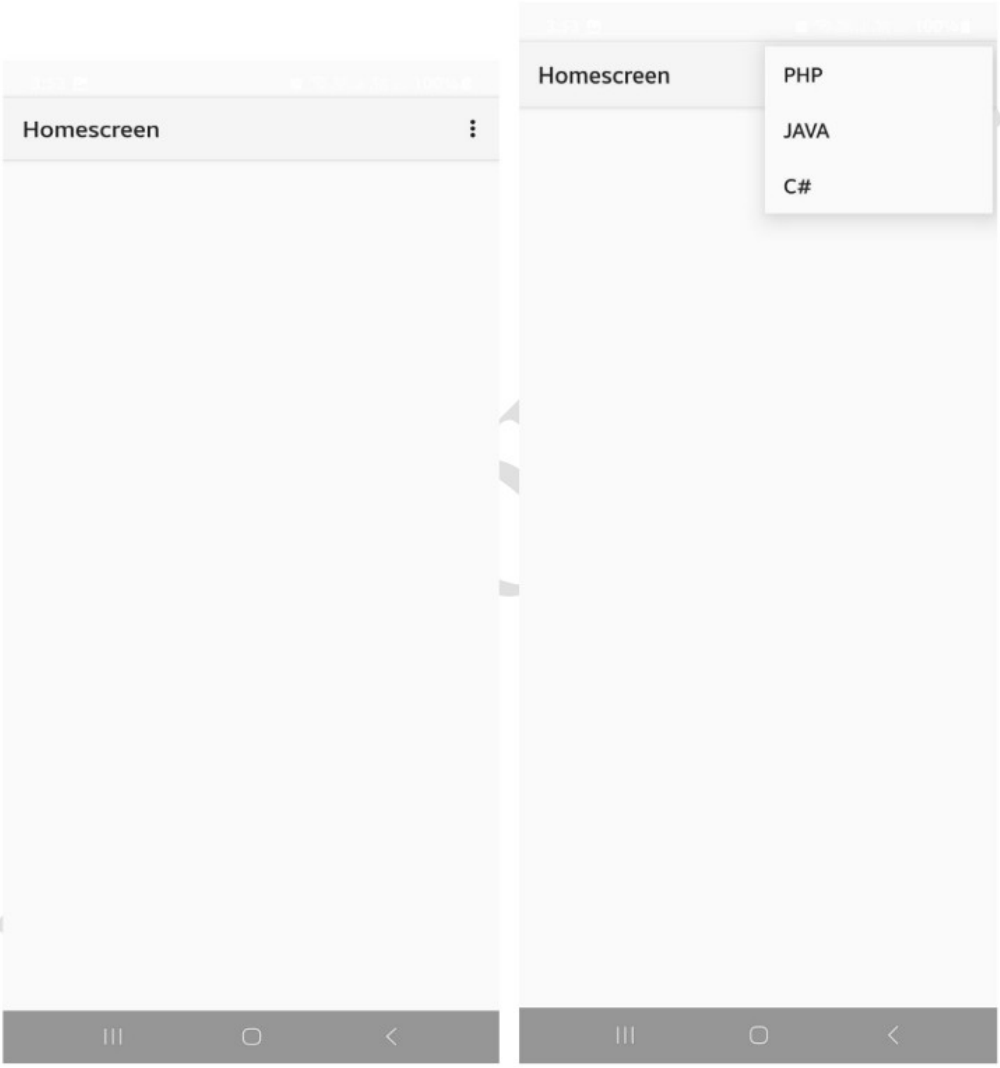
Set the theme as **`"@style/Theme.AppCompat.Light”`**

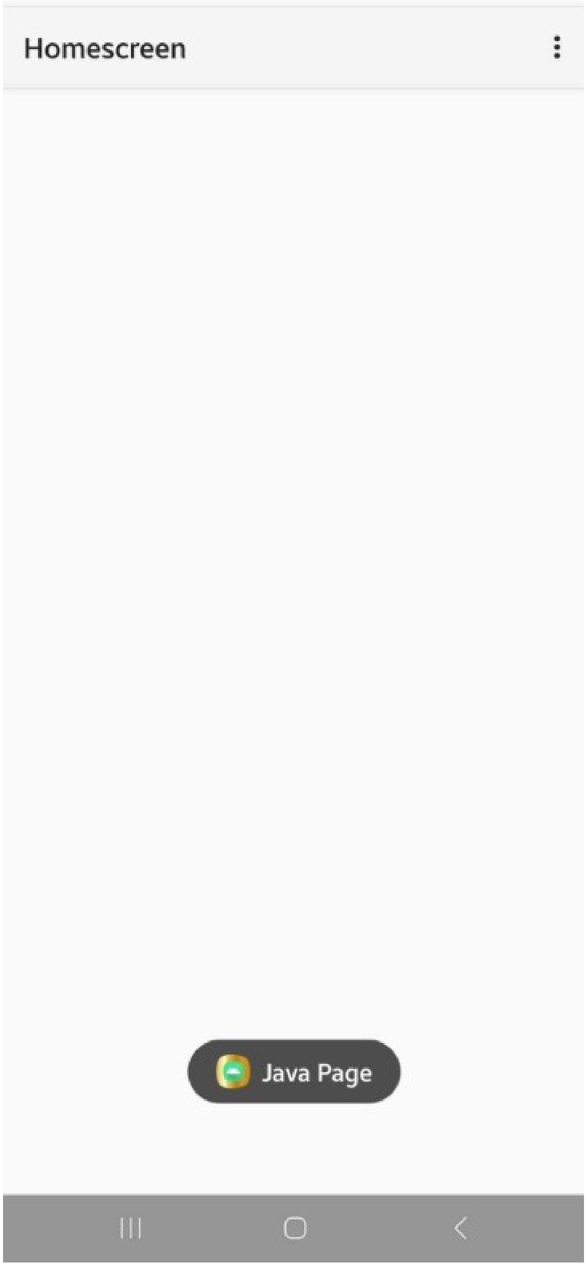
```
<?xml version="1.0" encoding="utf-8"?>
<manifest xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:tools="http://schemas.android.com/tools">
    <uses-permission android:name="android.permission.INTERNET"/>
    <application
        android:allowBackup="true"
        android:dataExtractionRules="@xml/data_extraction_rules"
        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.AppCompat.Light"
        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>
```

Activity Main.xml

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

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

Output




Homescreen
Java Page

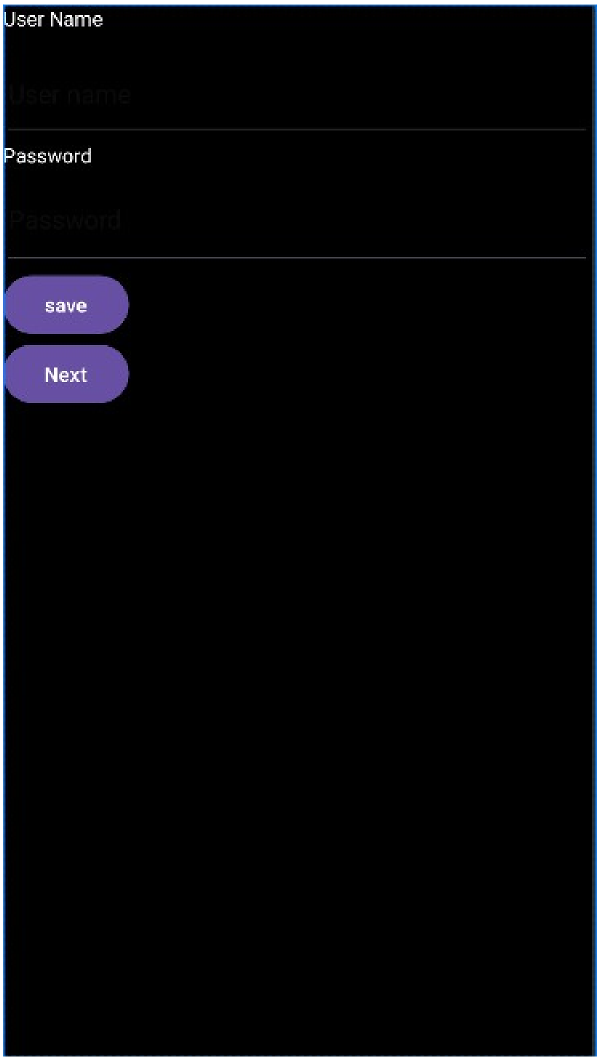
Program 8

Read/ write the Local data

Activity.xml file:

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

    <TextView
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="User Name"
         />

    <EditText
        android:id="@+id/username"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:hint="User name" />

    <TextView

        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:text="Password" />

    <EditText
        android:id="@+id/password"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:hint="Password" />

    <Button
        android:id="@+id/btnsave"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="save"/>
    <Button
        android:id="@+id/btnnext"
        android:layout_height="wrap_content"
        android:layout_width="wrap_content"
        android:text="Next"/>




</LinearLayout>
```

Design:



Activity_Main.java

```
package com.example1.lab8;

import androidx.appcompat.app.AppCompatActivity;

import android.os.Bundle;
import android.content.Context;
import android.content.Intent;
import android.content.SharedPreferences;
import android.view.View;
import android.widget.Button;
import android.widget.EditText;
import android.widget.Toast;

public class MainActivity extends AppCompatActivity {
    Button buttonsave, buttonnext;
    EditText username, password;

    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_main);
        buttonsave = (Button) findViewById(R.id.btnsave);
        buttonnext = (Button) findViewById(R.id.btnnext);
        username = (EditText) findViewById(R.id.username);
        password = (EditText) findViewById(R.id.password);
        buttonsave.setOnClickListener(new View.OnClickListener() {
            @Override
            public void onClick(View view) {
                //Writing Data To Share Preferences
                SharedPreferences sharedPreferences =
getSharedPreferences("MyPrefs", Context.MODE_PRIVATE);
                SharedPreferences.Editor editor = sharedPreferences.edit();
                editor.putString("password",
                        password.getText().toString());
editor.putString("username",
                        username.getText().toString());

                editor.apply();
                Toast.makeText(getApplicationContext(), "Saved
successfully", Toast.LENGTH_LONG).show();
            }
        });

        buttonnext.setOnClickListener(new View.OnClickListener() {
            @Override
            public void onClick(View view) {
                Intent intent = new
                        Intent(getApplicationContext(),
MainActivity2.class);
                startActivity(intent);
            }


        });
    }
}
```

Create new activity

Right Click app →new →activity →Empty views activity

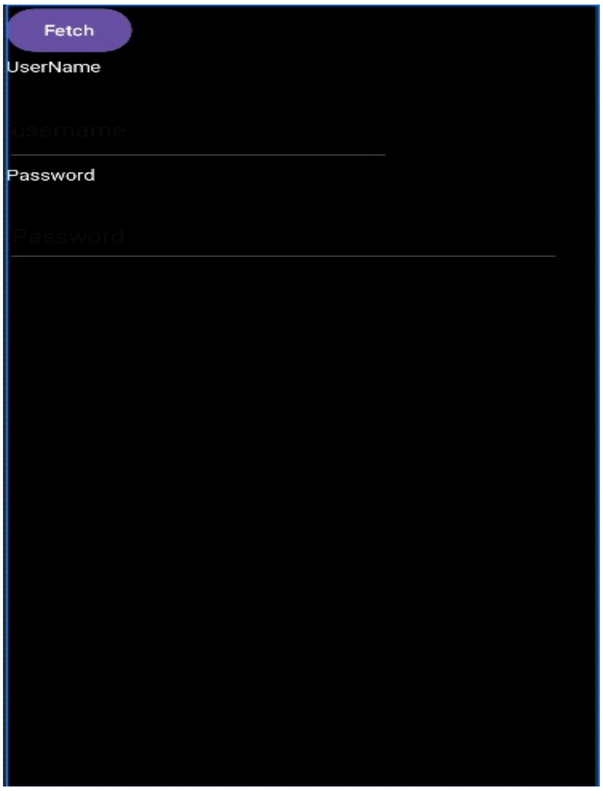
Give suitable name to the file and click finish

Make the layout as linear layout, add a button, 2 Textview and 2 EditText

activity_main2.xml

```
<?xml version="1.0" encoding="utf-8"?>
<LinearLayout xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:app="http://schemas.android.com/apk/res-auto"
    xmlns:tools="http://schemas.android.com/tools"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    tools:context=".MainActivity2"
    android:orientation="vertical">

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

    <TextView

        android:layout_width="263dp"
        android:layout_height="38dp"
        android:text="UserName" />

    <EditText
        android:id="@+id/UserName"
        android:layout_width="269dp"
        android:layout_height="63dp"
        android:hint="username" />

    <TextView
        android:layout_width="268dp"
        android:layout_height="40dp"
        android:text="Password" />

    <EditText
        android:id="@+id/Password"
        android:layout_width="388dp"
        android:layout_height="55dp"
        android:hint="Password" />




</LinearLayout>
```

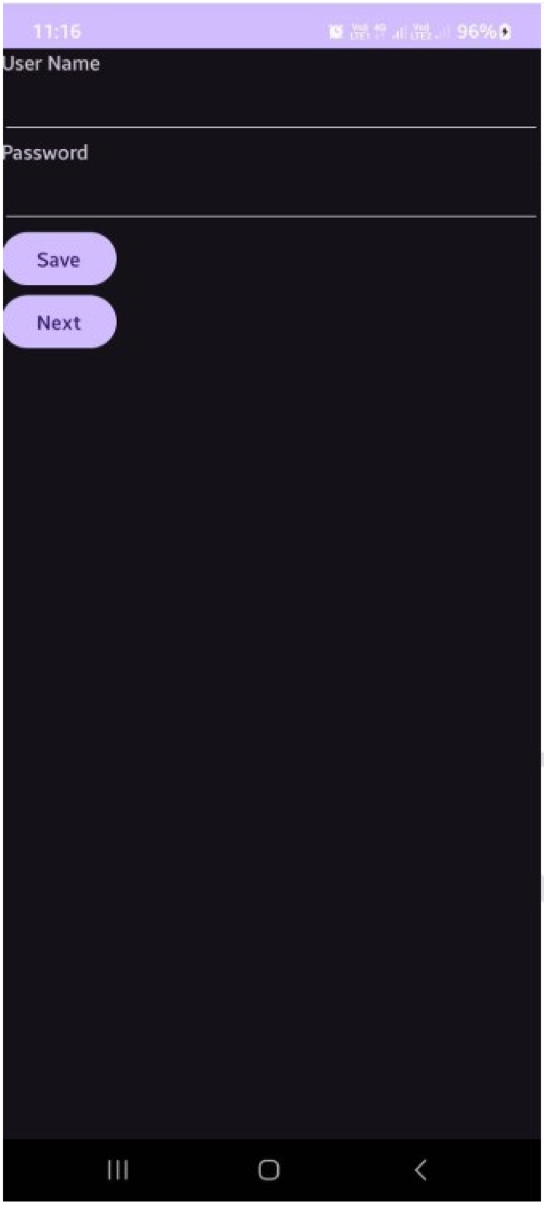
Design:

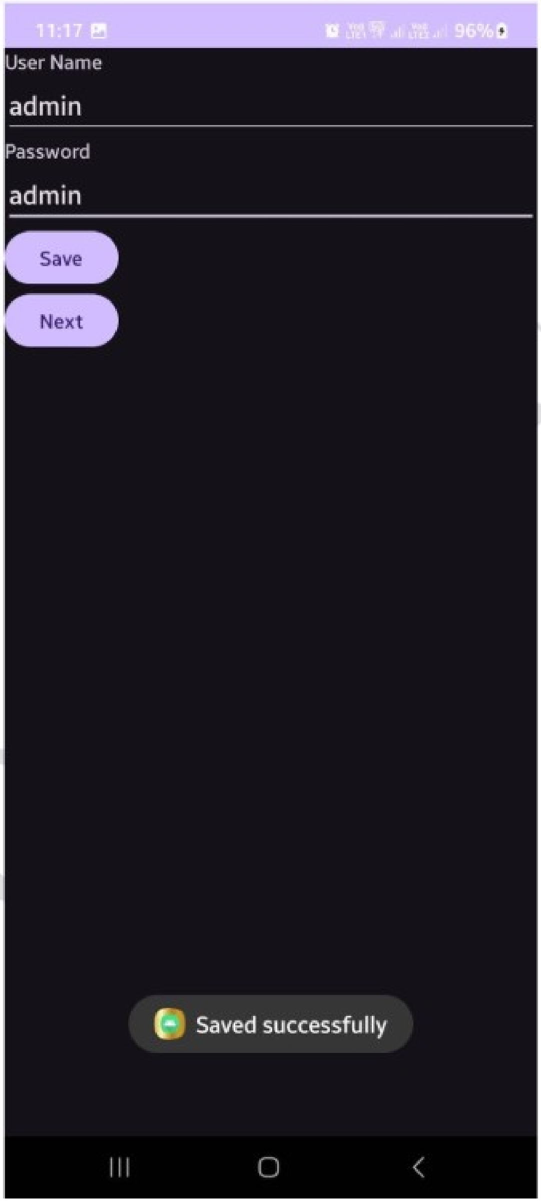


MainActivity2 .java

```
package com.example1.lab8;

import androidx.appcompat.app.AppCompatActivity;

import android.os.Bundle;
import android.content.Context;
import android.content.SharedPreferences;
import android.view.View;
import android.widget.Button;
import android.widget.EditText;


public class MainActivity2 extends AppCompatActivity {

    Button btnFetch;
    EditText UserName, Password;

    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_main2);
        btnFetch = (Button) findViewById(R.id.btnFetch);
        UserName = (EditText)findViewById(R.id.UserName);
        Password = (EditText)findViewById(R.id.Password);
        btnFetch.setOnClickListener(new View.OnClickListener() {
            @Override
            public void onClick(View view) {
// Reading data from SharedPreferences
                SharedPreferences sharedPreferences =
getSharedPreferences("MyPrefs",
                        Context.MODE_PRIVATE);
                String username = sharedPreferences.getString("username",
"");
                String password = sharedPreferences.getString("password",
"");
                UserName.setText(username);
                Password.setText(password);
            }
        });

        }
}
```

Output:





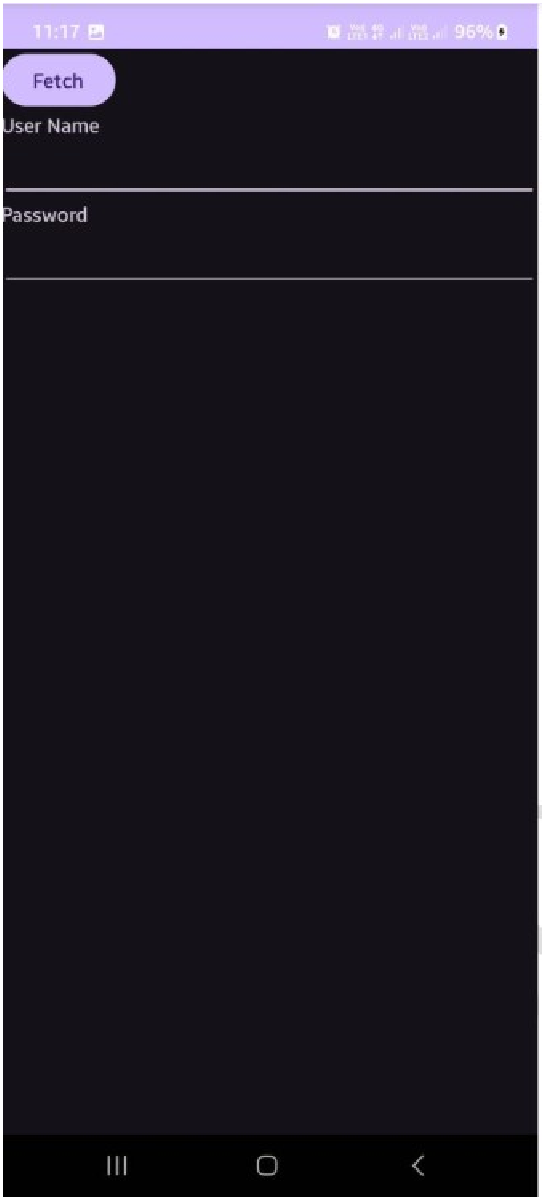
User Name

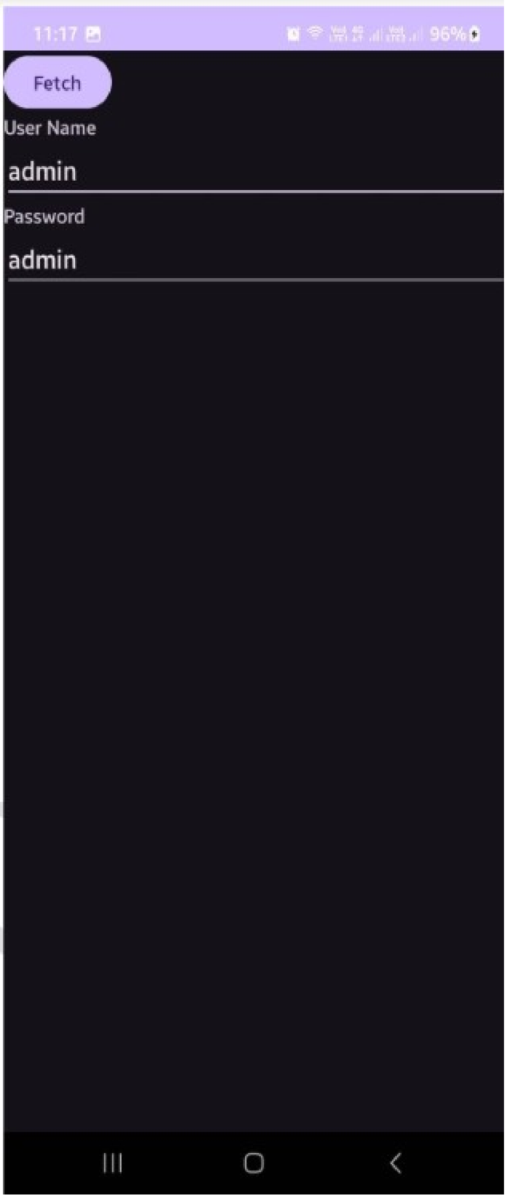
admin

Password

admin

Saved successfully





User Name
admin

Password
admin

Program 9

Create / Read / Write data with database (SQL Lite)

Steps: 1. Click Start - Android Studio, a Welcome to Android Studio dialog box will appear. Click New Project, the New Project Dialog box appears.

2. Choose Empty Views Activity then click Next.

3. Specify the Name of your project, Select the Language as Java, and Select the SDK as API 24(“Nougat”,Android 7.0). Click Finish Button.

4. Update the following code in activity_main.xml, activity_view.xml, MainActivity.java and ViewActivity.java

5. Create a class file right click app- new- java class name it as student and update the following code in student.java

6. To Create another activity right click on app – new activity-Empty views Activity.

7. Update the following code in activity_edit.xml and EditActivity.java.

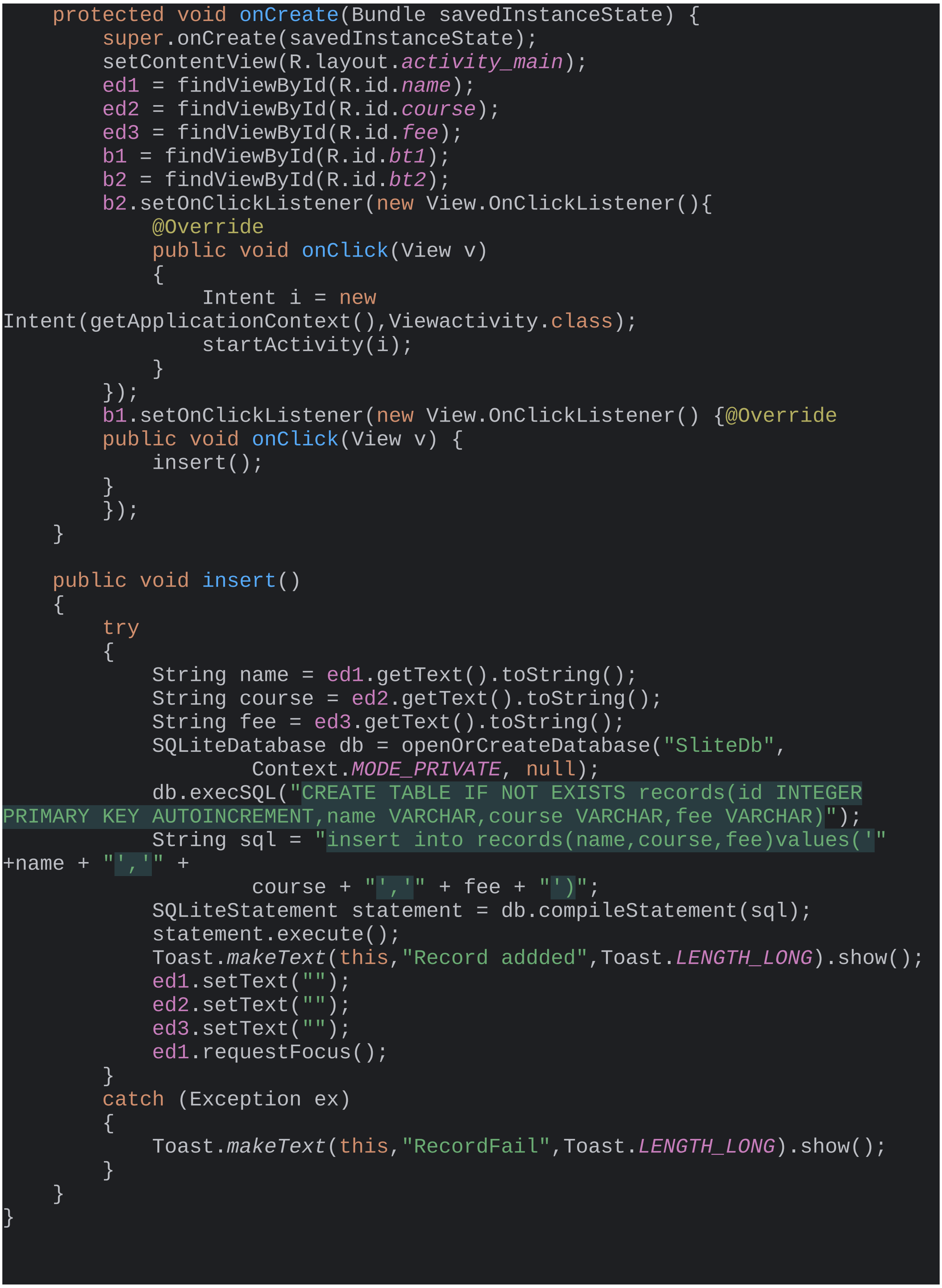
8. Click Run App or Shift+F10 to execute the application.

Activity_main.xml

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

        <TextView
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:text="Course Registation"
            android:textColor="@color/black"
            android:textSize="30dp" />
    </LinearLayout>
    <Space
        android:layout_width="40dp"
        android:layout_height="40dp"/>

    <LinearLayout
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:gravity="center"
        android:orientation="horizontal">

        <TextView
            android:layout_width="76dp"
            android:layout_height="31dp"
            android:text="Name" />

        <EditText
            android:id="@+id/name"
            android:layout_width="305dp"
            android:layout_height="65dp"
            android:layout_weight="1"
            android:ems="10"
            android:hint="name"
            android:textAlignment="center" />
    </LinearLayout>
    <LinearLayout
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:orientation="horizontal" android:gravity="center">

        <TextView
            android:layout_width="79dp"
            android:layout_height="wrap_content"
            android:text="Course" />

        <EditText
            android:id="@+id/course"
            android:layout_width="329dp"
            android:layout_height="67dp"
            android:layout_weight="1"
            android:ems="10"
            android:hint="course"
            android:textAlignment="center" />
    </LinearLayout>

    <LinearLayout
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:orientation="horizontal" android:gravity="center">

        <TextView
            android:layout_width="78dp"
            android:layout_height="32dp"
            android:text="Fee" />

        <EditText
            android:id="@+id/fee"
            android:layout_width="360dp"
            android:layout_height="65dp"
            android:layout_weight="1"
            android:ems="10"
            android:hint="fee"
            android:textAlignment="center" />
    </LinearLayout>
    <Space
        android:layout_width="30dp"
        android:layout_height="30dp"/>
    <LinearLayout
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:orientation="horizontal" android:gravity="center">
<Space
    android:layout_width="70dp"
    android:layout_height="30dp"/>
        <Button
            android:id="@+id/bt1"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:layout_weight="1"
            android:background="@color/cardview_dark_background"
            android:text="Ok" />

        <Space
            android:layout_width="30dp"
            android:layout_height="30dp"/>

        <Button
            android:id="@+id/bt2"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:layout_weight="1"
            android:background="@color/cardview_dark_background"
            android:text="View" />

    </LinearLayout>


</LinearLayout>
```

MainActivity.java

```
package com.example1.lab9;

import androidx.appcompat.app.AppCompatActivity;

import android.os.Bundle;
import androidx.appcompat.app.AppCompatActivity;
import android.content.Context;
import android.content.Intent;
import android.database.sqlite.SQLiteDatabase;
import android.database.sqlite.SQLiteStatement;
import android.os.Bundle;
import android.view.View;
import android.widget.Button;
import android.widget.EditText;
import android.widget.Toast;

public class MainActivity extends AppCompatActivity {

    EditText ed1,ed2,ed3;
    Button b1,b2;
    @Override
```



```java
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_main);
        ed1 = findViewById(R.id.name);
        ed2 = findViewById(R.id.course);
        ed3 = findViewById(R.id.fee);
        b1 = findViewById(R.id.bt1);
        b2 = findViewById(R.id.bt2);
        b2.setOnClickListener(new View.OnClickListener(){
            @Override
            public void onClick(View v)
            {
                Intent i = new
Intent(getApplicationContext(),Viewactivity.class);
                startActivity(i);
            }
        });
        b1.setOnClickListener(new View.OnClickListener() {@Override
        public void onClick(View v) {
            insert();
        }
        });
    }

    public void insert()
    {
        try
        {
            String name = ed1.getText().toString();
            String course = ed2.getText().toString();
            String fee = ed3.getText().toString();
            SQLiteDatabase db = openOrCreateDatabase("SliteDb",
                    Context.MODE_PRIVATE, null);
            db.execSQL("CREATE TABLE IF NOT EXISTS records(id INTEGER
PRIMARY KEY AUTOINCREMENT,name VARCHAR,course VARCHAR,fee VARCHAR)");
            String sql = "insert into records(name,course,fee)values('"
+name + "','" +
                    course + "','" + fee + "')";
            SQLiteStatement statement = db.compileStatement(sql);
            statement.execute();
            Toast.makeText(this,"Record adddded",Toast.LENGTH_LONG).show();
            ed1.setText("");
            ed2.setText("");
            ed3.setText("");
            ed1.requestFocus();
        }
        catch (Exception ex)
        {
            Toast.makeText(this,"RecordFail",Toast.LENGTH_LONG).show();
        }
    }
}
```

/* Add Student.class file (Right click on package name */

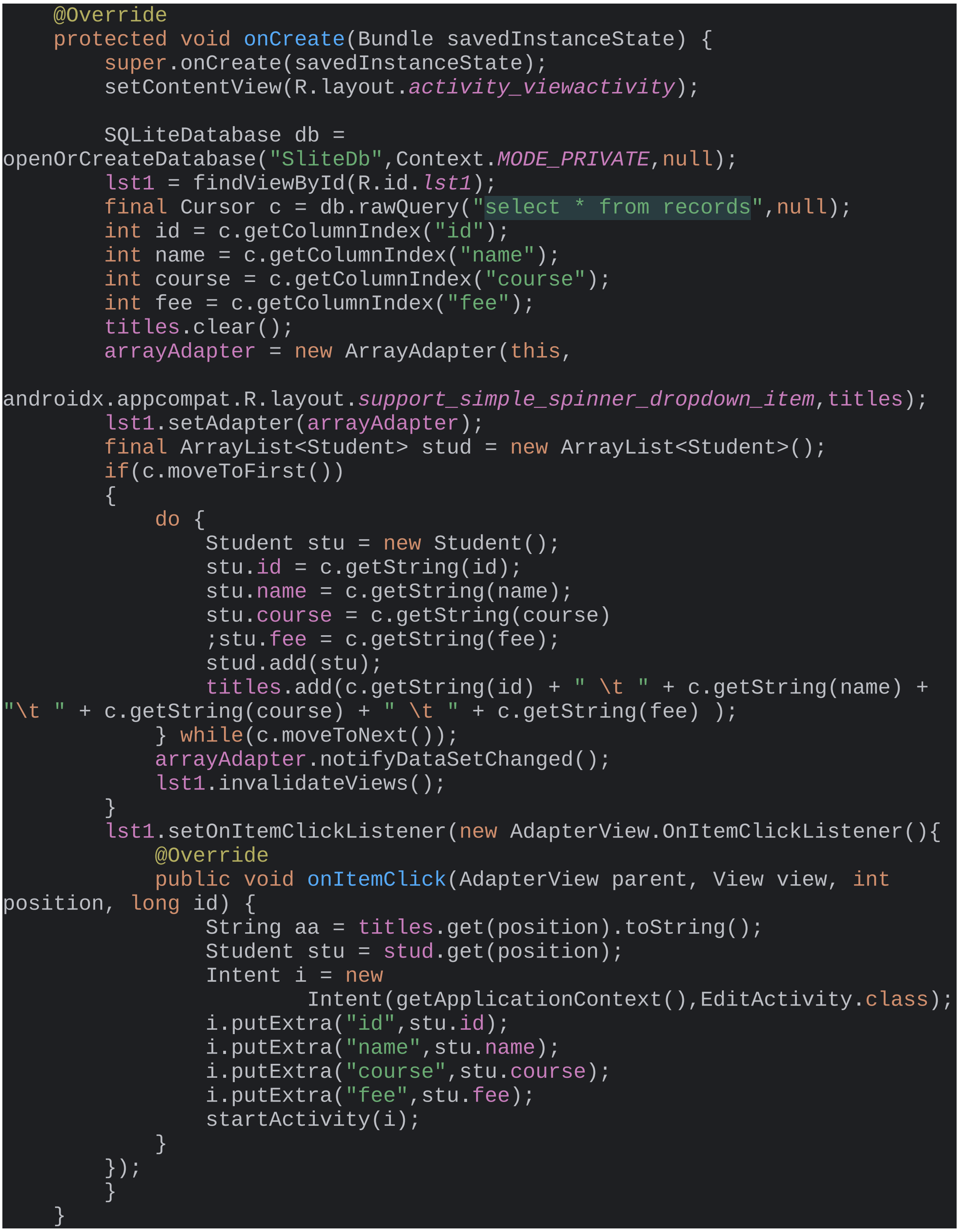
```
package com.example1.lab9;

public class Student {
    String id;
    String name;
    String course;
    String fee;
    String titles;
}
```

activity_viewactivity.xml

```
<?xml version="1.0" encoding="utf-8"?>
<LinearLayout xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:app="http://schemas.android.com/apk/res-auto"
    xmlns:tools="http://schemas.android.com/tools"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    tools:context=".Viewactivity"
    android:orientation="vertical">
    <ListView
        android:layout_width="match_parent"
        android:layout_height="match_parent"
        android:id="@+id/lst1"
        />

</LinearLayout>
```

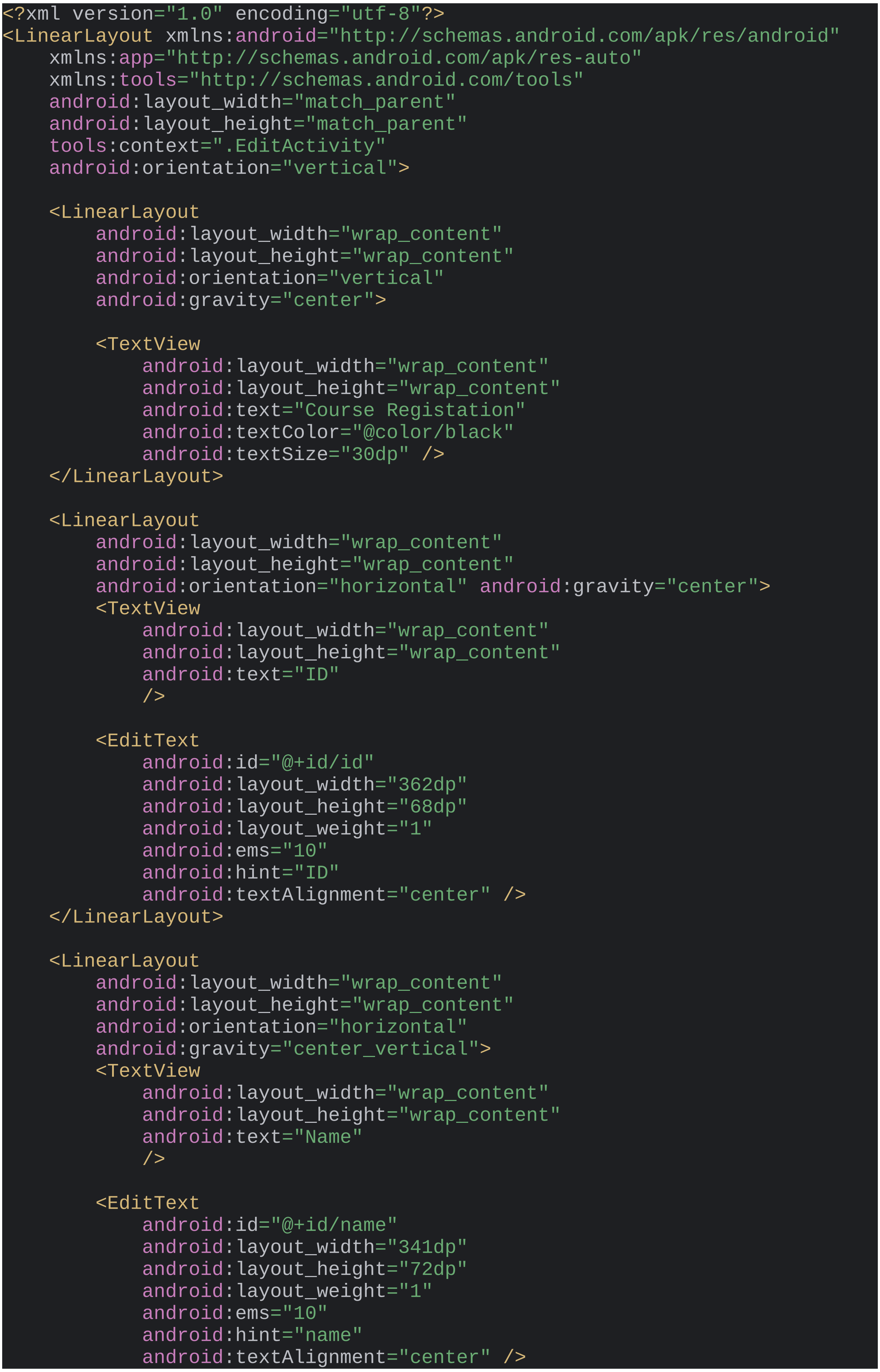
Viewactivity.java

```
package com.example1.lab9;

import androidx.appcompat.app.AppCompatActivity;

import android.os.Bundle;
import androidx.appcompat.app.AppCompatActivity;
import android.content.Context;
import android.content.Intent;
import android.database.Cursor;
import android.database.sqlite.SQLiteDatabase;
import android.os.Bundle;
import android.view.View;
import android.widget.AdapterView;
import android.widget.ArrayAdapter;
import android.widget.ListView;
import java.util.ArrayList;

public class Viewactivity extends AppCompatActivity {

    ListView lst1;
    ArrayList<String> titles = new ArrayList<String>();
    ArrayAdapter arrayAdapter;

```



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

        SQLiteDatabase db =
openOrCreateDatabase("SliteDb",Context.MODE_PRIVATE,null);
        lst1 = findViewById(R.id.lst1);
        final Cursor c = db.rawQuery("select * from records",null);
        int id = c.getColumnIndex("id");
        int name = c.getColumnIndex("name");
        int course = c.getColumnIndex("course");
        int fee = c.getColumnIndex("fee");
        titles.clear();
        arrayAdapter = new ArrayAdapter(this,

androidx.appcompat.R.layout.support_simple_spinner_dropdown_item,titles);
        lst1.setAdapter(arrayAdapter);
        final ArrayList<Student> stud = new ArrayList<Student>();
        if(c.moveToFirst())
        {
            do {
                Student stu = new Student();
                stu.id = c.getString(id);
                stu.name = c.getString(name);
                stu.course = c.getString(course)
                ;stu.fee = c.getString(fee);
                stud.add(stu);
                titles.add(c.getString(id) + " \t " + c.getString(name) +
"\t " + c.getString(course) + " \t " + c.getString(fee) );
            } while(c.moveToNext());
            arrayAdapter.notifyDataSetChanged();
            lst1.invalidateViews();
        }
        lst1.setOnItemClickListener(new AdapterView.OnItemClickListener(){
            @Override
            public void onItemClick(AdapterView parent, View view, int
position, long id) {
                String aa = titles.get(position).toString();
                Student stu = stud.get(position);
                Intent i = new
                        Intent(getApplicationContext(),EditActivity.class);
                i.putExtra("id",stu.id);
                i.putExtra("name",stu.name);
                i.putExtra("course",stu.course);
                i.putExtra("fee",stu.fee);
                startActivity(i);
            }
        });
        }
    }
```

activity_edit.xml

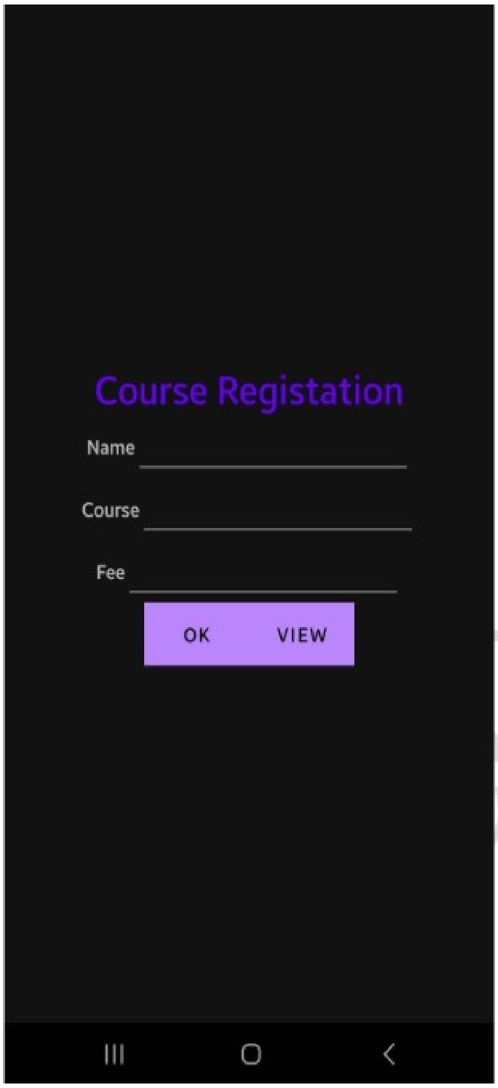


```xml
<?xml version="1.0" encoding="utf-8"?>
<LinearLayout xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:app="http://schemas.android.com/apk/res-auto"
    xmlns:tools="http://schemas.android.com/tools"
    android:layout_width="match_parent"
    android:layout_height="match_parent"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

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

        <TextView
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:text="Course Registation"
            android:textColor="@color/black"
            android:textSize="30dp" />
    </LinearLayout>

    <LinearLayout
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:orientation="horizontal" android:gravity="center">
        <TextView
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:text="ID"
            />

        <EditText
            android:id="@+id/id"
            android:layout_width="362dp"
            android:layout_height="68dp"
            android:layout_weight="1"
            android:ems="10"
            android:hint="ID"
            android:textAlignment="center" />
    </LinearLayout>

    <LinearLayout
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:orientation="horizontal"
        android:gravity="center_vertical">
        <TextView
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:text="Name"
            />

        <EditText
            android:id="@+id/name"
            android:layout_width="341dp"
            android:layout_height="72dp"
            android:layout_weight="1"
            android:ems="10"
            android:hint="name"
            android:textAlignment="center" />
    </LinearLayout>
        <LinearLayout
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:orientation="horizontal" android:gravity="center">
            <TextView
                android:layout_width="wrap_content"
                android:layout_height="wrap_content"
                android:text="Course"/>

            <EditText
                android:id="@+id/course"
                android:layout_width="338dp"
                android:layout_height="81dp"
                android:layout_weight="1"
                android:ems="10"
                android:hint="Course"
                android:textAlignment="center" />
        </LinearLayout>
        <LinearLayout
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:orientation="horizontal"
            android:gravity="center">
            <TextView
                android:layout_width="wrap_content"
                android:layout_height="wrap_content"
                android:text="Fee"
                />

            <EditText
                android:id="@+id/fee"
                android:layout_width="363dp"
                android:layout_height="55dp"
                android:layout_weight="1"
                android:ems="10"
                android:hint="fee"
                android:textAlignment="center" />
        </LinearLayout>
    <Space
        android:layout_width="30dp"
        android:layout_height="30dp">
        <LinearLayout
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:orientation="horizontal" android:gravity="center">

            <Button
                android:id="@+id/bt1"
                android:layout_width="146dp"
                android:layout_height="wrap_content"
                android:layout_weight="1"
                android:background="@color/cardview_dark_background"
                android:text="Edit" />

            <Button
                android:id="@+id/bt2"
                android:layout_width="wrap_content"
                android:layout_height="wrap_content"
                android:layout_weight="1"
                android:background="@color/cardview_dark_background"
                android:text="Delete" />
            <Button
                android:layout_width="wrap_content"
                android:layout_height="wrap_content"
                android:layout_weight="1" android:id="@+id/bt3"
                android:text="Back"
                android:background="@color/cardview_dark_background"
                />
        </LinearLayout>

    </LinearLayout>
```

Activity_edit.java

```
package com.example1.lab9;

import androidx.appcompat.app.AppCompatActivity;

import android.os.Bundle;
import androidx.appcompat.app.AppCompatActivity;
import android.content.Context;
import android.content.Intent;
import android.database.sqlite.SQLiteDatabase;
import android.database.sqlite.SQLiteStatement;
import android.widget.Toast;
import android.view.View;
import android.widget.Button;
import android.widget.EditText;

public class EditActivity extends AppCompatActivity {
    EditText ed1,ed2,ed3,ed4;
    Button b1,b2,b3;
    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_edit);
        ed1 = findViewById(R.id.name);
        ed2 = findViewById(R.id.course);
        ed3 = findViewById(R.id.fee);
        ed4 = findViewById(R.id.id);
        b1 = findViewById(R.id.bt1);
        b2 = findViewById(R.id.bt2);
        b3 = findViewById(R.id.bt3);
        Intent i = getIntent();
        String t1 = i.getStringExtra("id").toString();
        String t2 = i.getStringExtra("name").toString();
        String t3 = i.getStringExtra("course").toString();
        String t4 = i.getStringExtra("fee").toString();
        ed4.setText(t1);
        ed1.setText(t2);
        ed2.setText(t3);
        ed3.setText(t4);
        b2.setOnClickListener(new View.OnClickListener()
        {@Override
        public void onClick(View v) {
            try
            {
                String id = ed4.getText().toString();SQLiteDatabase db =
openOrCreateDatabase("SliteDb",Context.MODE_PRIVATE,null);
                String sql = "delete from records where id = " + id + "";
                SQLiteStatement statement = db.compileStatement(sql);
                statement.execute();

Toast.makeText(EditActivity.this,"RecordDeleted",Toast.LENGTH_LONG).show();
                ed1.setText("");
                ed2.setText("");
                ed3.setText("");
                ed1.requestFocus();
            }
            catch (Exception ex)
            {
                Toast.makeText(EditActivity.this,"Record
Fail",Toast.LENGTH_LONG).show();
            }
        }
        });
        b3.setOnClickListener(new View.OnClickListener() {@Override
        public void onClick(View v) {
            Intent i = new
Intent(getApplicationContext(),Viewactivity.class);
            startActivity(i);
        }
        });
        b1.setOnClickListener(new View.OnClickListener() {@Override
        public void onClick(View v) {
            try {
                String name = ed1.getText().toString();
                String course = ed2.getText().toString();
                String fee = ed3.getText().toString();
                String id = ed4.getText().toString();
                SQLiteDatabase db =
openOrCreateDatabase("SliteDb",Context.MODE_PRIVATE, null);
                String sql = "update records set name = '" + name +
"',course='" +course +
                        "',fee='" + fee + "' where id= " + id + "";
                SQLiteStatement statement = db.compileStatement(sql);
                statement.execute();
                Toast.makeText(EditActivity.this, "Record
Updated",Toast.LENGTH_LONG).show();
                ed1.setText("");
                ed2.setText("");
                ed3.setText("");
                ed1.requestFocus();
            } catch (Exception ex) {
                Toast.makeText(EditActivity.this, "Record Fail",
                        Toast.LENGTH_LONG).show();
            }
        }
        });
    } }
```

output:

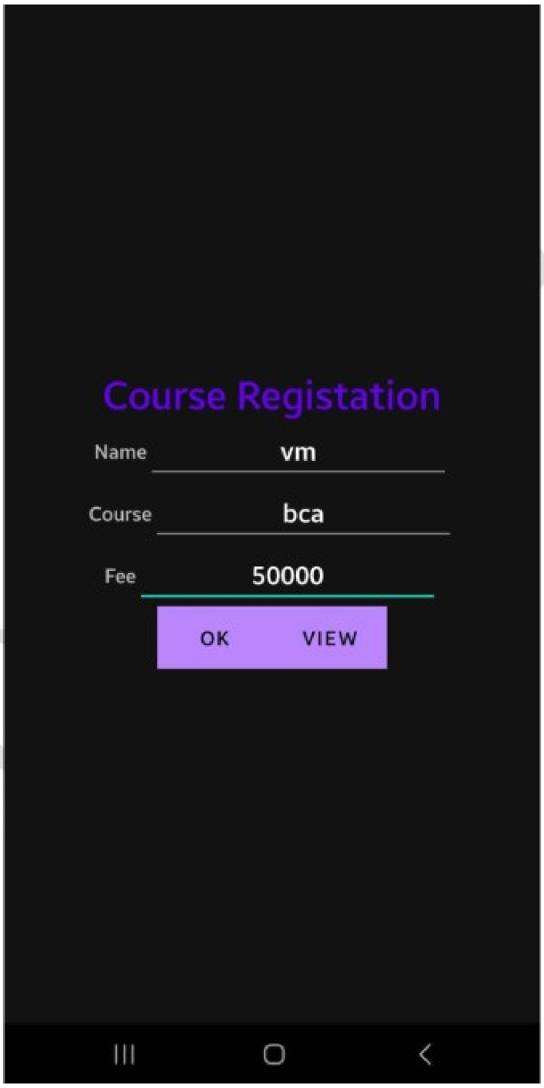


Course Registation

Name

Course

Fee

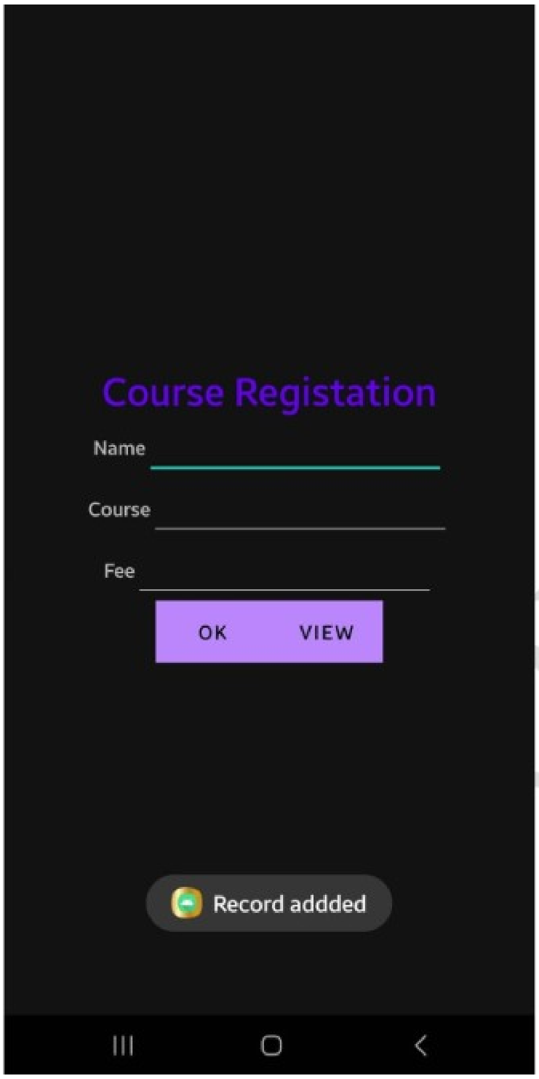


Course Registation

Name vm

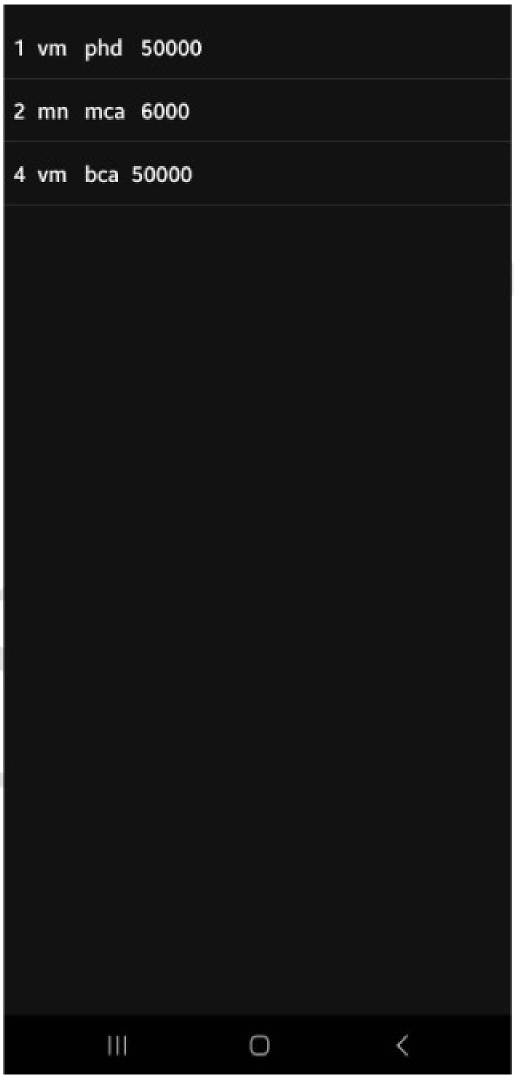
Course bca

Fee 50000



Course Registation

Record adddded



```
1 vm  phd  50000
2 mn  mca  6000
4 vm  bca  50000
```

Lab program 10:

Create an application to send SMS and receive SMS

Steps: 1. Click Start- Android Studio, a Welcome to Android Studio dialog box will appear.

Click New Project, the New Project Dialog box appears.

2. Choose Empty Views Activity then click Next.

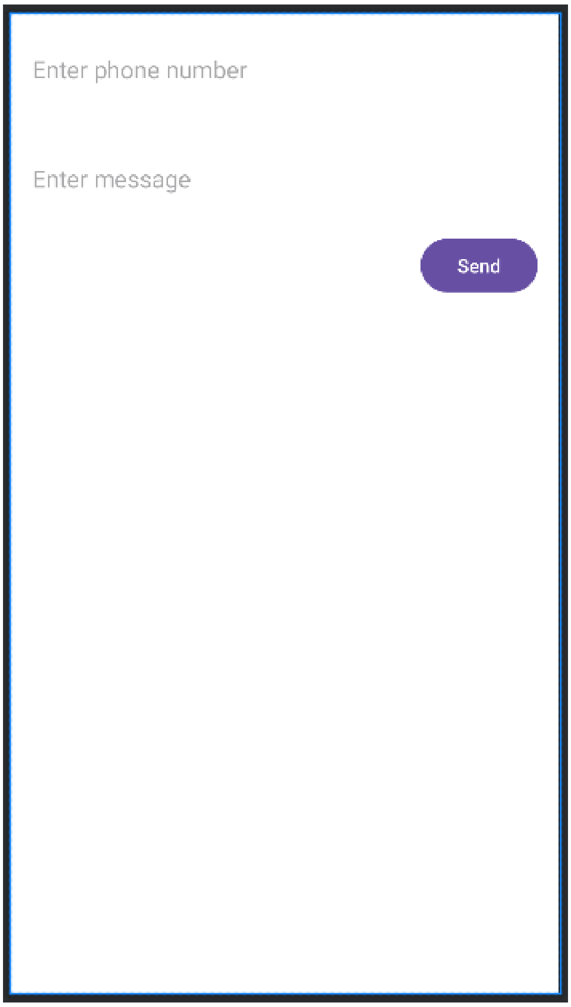
3. Specify the Name of your project, Select the Language as Java, and Select the SDK as API 24(“Nougat”,Android 7.0). Click Finish Button.

4. Update the following code in activity_main.xml and MainActivity.java 5. Click Run app or shift+F10 to execute the application.

Main Activity.xml

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

    <EditText
        android:id="@+id/editTextPhoneNumber"
        android:layout_width="match_parent"
        android:layout_height="50dp"
        android:layout_marginStart="16dp"
        android:layout_marginTop="16dp"
        android:layout_marginEnd="16dp"
        android:layout_marginBottom="16dp"
        android:background="@color/white"
        android:hint="Enter phone number" />

    <EditText
        android:id="@+id/editTextMessage"
        android:layout_width="match_parent"
        android:layout_height="50dp"
        android:hint="Enter message"
        android:layout_below="@id/editTextPhoneNumber"
        android:layout_margin="16dp"
        android:background="@color/white"/>

    <Button
        android:id="@+id/buttonSend"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="Send"
        android:layout_below="@id/editTextMessage"
        android:layout_alignParentEnd="true"
        android:layout_marginEnd="16dp"
        android:onClick="sendMessage"
        tools:ignore="UsingOnClickInXml" />

    <TextView
        android:id="@+id/textViewReceivedMessages"
        android:layout_width="match_parent"
        android:layout_height="50dp"
        android:layout_below="@id/buttonSend"
        android:layout_marginStart="16dp"
        android:layout_marginTop="16dp"
        android:layout_marginEnd="16dp"
        android:layout_marginBottom="16dp"
        android:textColor="@color/white"
        android:background = "@color/white"/>


</RelativeLayout>
```

Design:



MainActivity.java

```
package com.example1.lab10;

import androidx.appcompat.app.AppCompatActivity;

import android.os.Bundle;
import androidx.appcompat.app.AppCompatActivity;
import androidx.core.app.ActivityCompat;
import androidx.core.content.ContextCompat;
import android.content.BroadcastReceiver;
import android.content.Context;
import android.content.Intent;
import android.content.IntentFilter;
import android.content.pm.PackageManager;
import android.os.Bundle;
import android.telephony.SmsManager;
import android.telephony.SmsMessage;
import android.view.View;
import android.widget.EditText;
import android.widget.TextView;
import android.widget.Toast;
import android.Manifest;

public class MainActivity extends AppCompatActivity {
    private static final int SMS_PERMISSION_CODE = 101;
    private EditText editTextPhoneNumber;
    private EditText editTextMessage;
    private TextView textViewReceivedMessage;
    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_main);

        editTextPhoneNumber = findViewById(R.id.editTextPhoneNumber);
        editTextMessage = findViewById(R.id.editTextMessage);
        textViewReceivedMessage =
findViewById(R.id.textViewReceivedMessages);

        //Request SMS permissions if not granted

        if (!checkSMSPermission())
        {
            requestSMSPermission();
        }

        /// Register SMS receiver
        IntentFilter intentFilter = new IntentFilter();
        intentFilter.addAction("android.provider.Telephony.SMS_RECEIVED");
        registerReceiver(smsReceiver, intentFilter);
    }
    protected void onDestroy()
    {
        super.onDestroy();
        unregisterReceiver(smsReceiver);
    }
    // Button click listener for sending SMS

    public void sendMessage(View view) { String phoneNumber =
            editTextPhoneNumber.getText().toString().trim();
        String message = editTextMessage.getText().toString();
        if (phoneNumber.isEmpty())
        {
            Toast.makeText(this, "Please enter a valid phone number",
                    Toast.LENGTH_SHORT).show();
            return;
        }

        try {
            SmsManager smsManager = SmsManager.getDefault();
            smsManager.sendTextMessage(phoneNumber, null, message, null,
null);
            Toast.makeText(this, "Message sent",
Toast.LENGTH_SHORT).show();
```

```
        }
        catch (IllegalArgumentException e)
        {
            Toast.makeText(this, "Invalid phone number format",
                    Toast.LENGTH_SHORT).show();
        } catch (Exception e) {
            Toast.makeText(this, "Failed to send message",
Toast.LENGTH_SHORT).show();
            e.printStackTrace();
        }
}


    // Check if SMS permission is granted
    private boolean checkSMSPermission() {
        return ContextCompat.checkSelfPermission(this,
Manifest.permission.SEND_SMS) ==
                PackageManager.PERMISSION_GRANTED;
    }
    // Request SMS permission
    private void requestSMSPermission()
{ ActivityCompat.requestPermissions(this, new
            String[]{Manifest.permission.SEND_SMS}, SMS_PERMISSION_CODE);
    }
    // SMS receiver
    private final BroadcastReceiver smsReceiver = new BroadcastReceiver()
    { @Override
    public void onReceive(Context context, Intent intent) { Bundle bundle =
intent.getExtras();
        if (bundle != null)


        {
            Object[] pdus = (Object[]) bundle.get("pdus");
            if (pdus != null) {
                for (Object pdu : pdus)
                {
                    SmsMessage smsMessage =
SmsMessage.createFromPdu((byte[]) pdu);
                    String senderPhoneNumber =
smsMessage.getDisplayOriginatingAddress();
                    String messageBody = smsMessage.getMessageBody();
                    textViewReceivedMessage.append("From: " +
senderPhoneNumber + "\n");
                    textViewReceivedMessage.append("Message: " +
                            messageBody + "\n\n");
                }
            }
        }
    }
    };
}
```

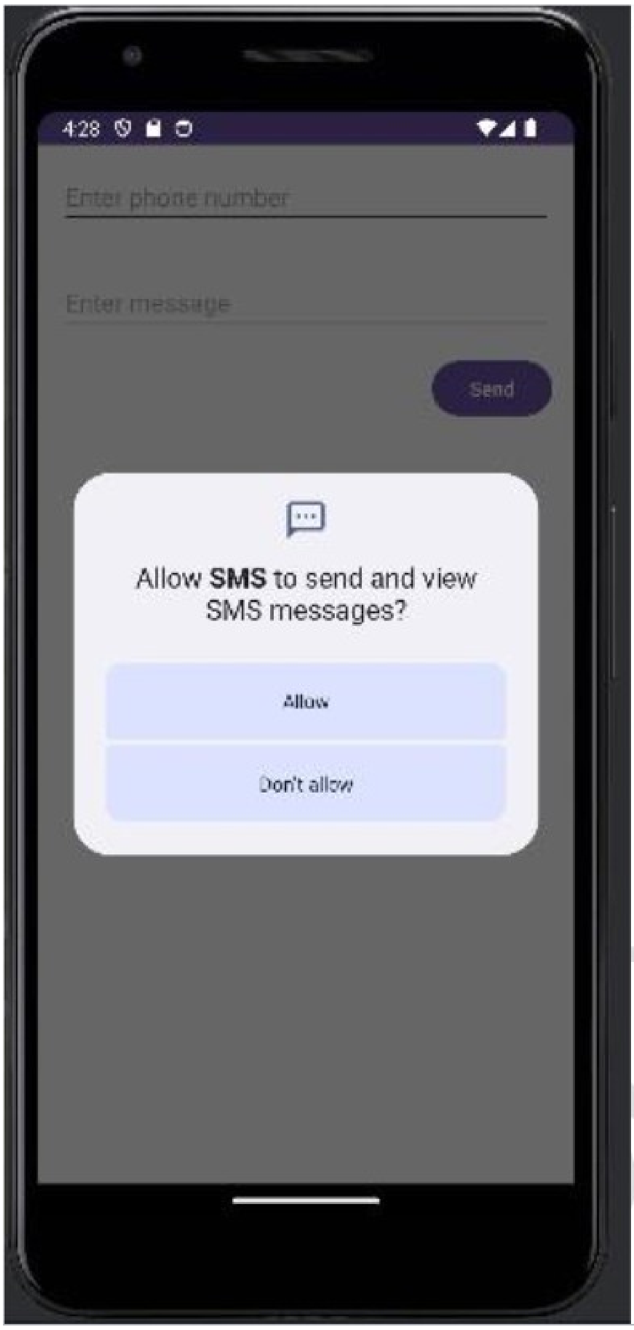
Manifest.xml:

Add SMS permission in Manifest file

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

```xml
    <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.Lab10"
        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>
```

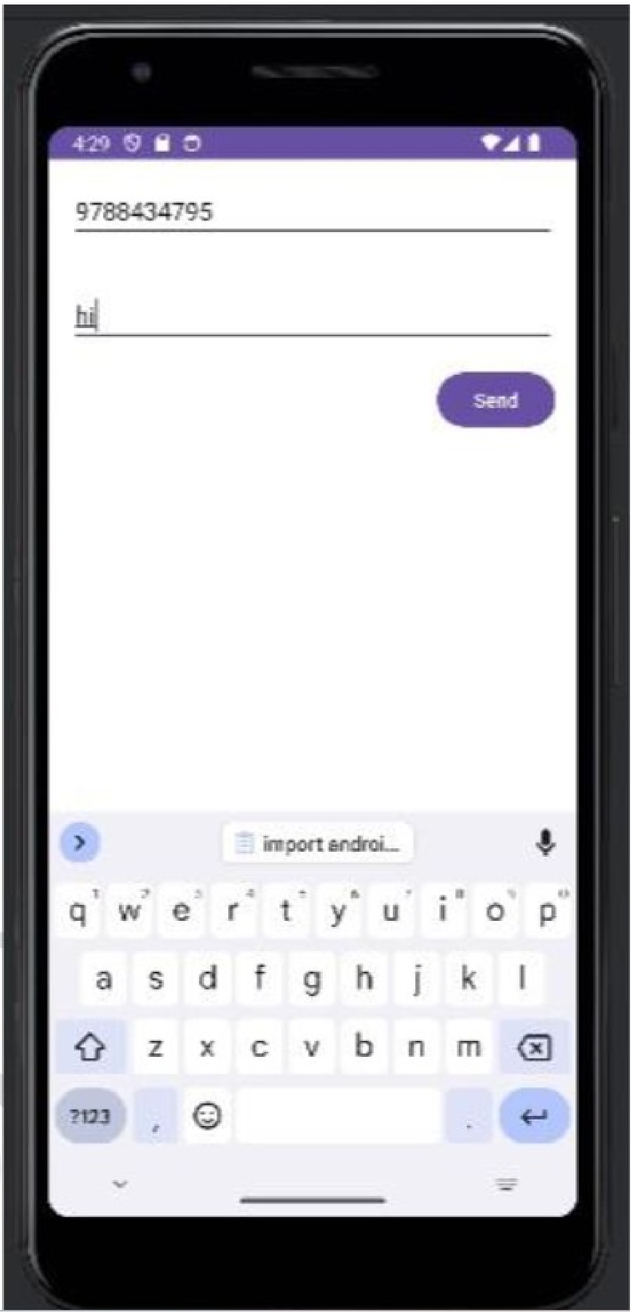
Output:



Enter phone number

Enter message

Allow **SMS** to send and view SMS messages?



9788434795

hi

Program 11

Create an application to send an Email.

Steps: 1. Click Start- Android Studio, a Welcome to Android Studio dialog box will appear. Click New Project, the New Project Dialog box appears.

2. Choose Empty Views Activity then click Next.

3. Specify the Name of your project, Select the Language as Java, and Select the SDK as API 24(“Nougat”,Android 7.0).Click Finish Button.

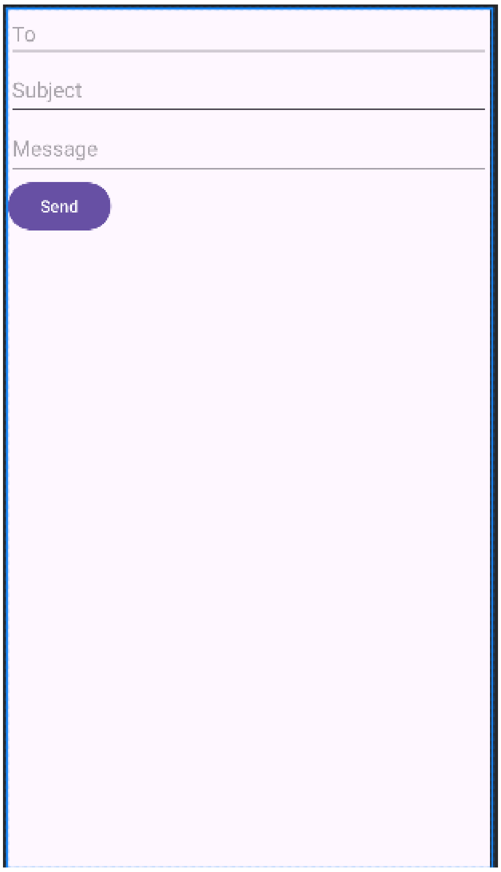
4. Update the following code in activity_main.xml and MainActivity.java

Activity_main.xml:

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

    <EditText
        android:id="@+id/editTextTo"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:hint="To"/>

    <EditText
        android:id="@+id/editTextSubject"
        android:layout_width="match_parent"
        android:layout_height="50dp"
        android:layout_below="@id/editTextTo"
        android:hint="Subject"/>

    <EditText
        android:id="@+id/editTextMessage"
        android:layout_width="match_parent"
        android:layout_height="50dp"
        android:layout_below="@id/editTextSubject"
        android:hint="Message"/>

    <Button
        android:id="@+id/buttonSend"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_below="@id/editTextMessage"
        android:text="Send"/>
```

```
</RelativeLayout>
```

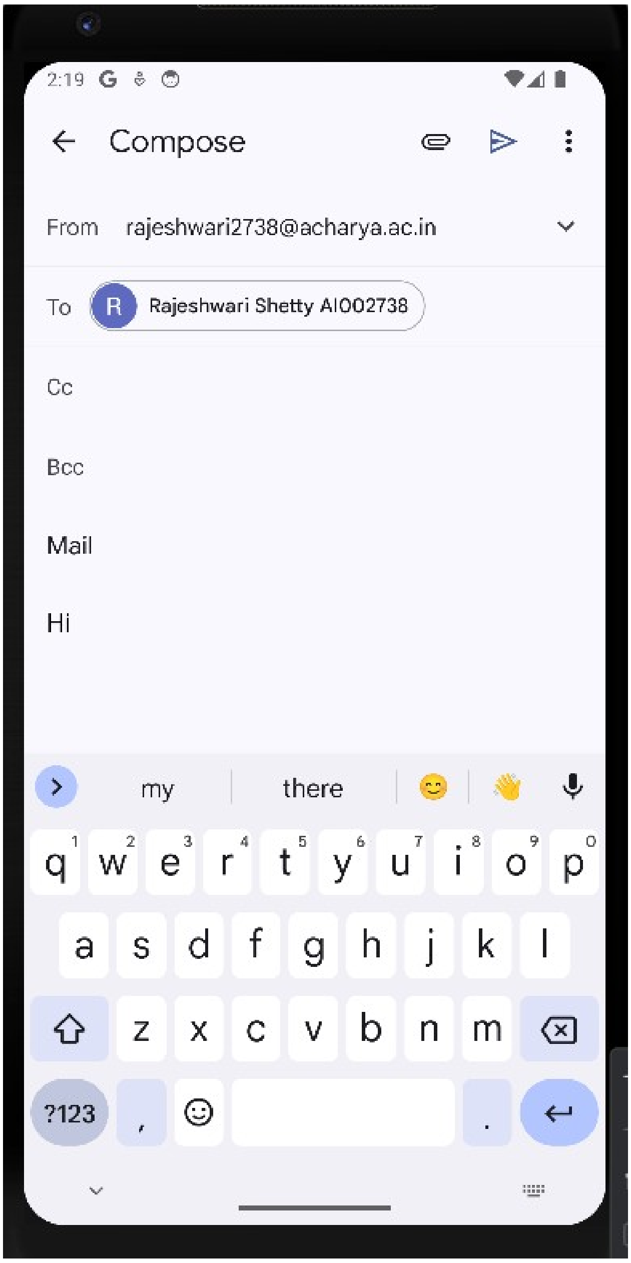
Design:



MainActivity.java:

```
package com.example1.lab11;

import androidx.appcompat.app.AppCompatActivity;

import android.os.Bundle;
import android.annotation.SuppressLint;
import android.content.Intent;
import android.view.View;
import android.widget.Button;
import android.widget.EditText;

public class MainActivity extends AppCompatActivity {
    EditText editTextTo, editTextSubject, editTextMessage;
    Button buttonSend;
    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_main);
        editTextTo = findViewById(R.id.editTextTo);
        editTextSubject = findViewById(R.id.editTextSubject);
        editTextMessage = findViewById(R.id.editTextMessage);
        buttonSend = findViewById(R.id.buttonSend);
        buttonSend.setOnClickListener(new View.OnClickListener(){

            @Override
            public void onClick(View v) {
                sendEmail();
            }
        });

    }

    @SuppressLint("QueryPermissionsNeeded")
    private void sendEmail() {
        String to = editTextTo.getText().toString().trim();
        String subject = editTextSubject.getText().toString().trim();
        String message = editTextMessage.getText().toString().trim();
        Intent intent = new Intent(Intent.ACTION_SEND);
        intent.setType("text/plain");
        intent.putExtra(Intent.EXTRA_EMAIL, new String[]{to});
        intent.putExtra(Intent.EXTRA_SUBJECT, subject);
        intent.putExtra(Intent.EXTRA_TEXT, message);
        if (intent.resolveActivity(getPackageManager()) != null)
        {
            startActivity(Intent.createChooser(intent, "Choose an email
client"));
        }
    }
}
```

Output:



From rajeshwari2738@acharya.ac.in

To Rajeshwari Shetty AI002738

Cc

Bcc

Mail

Hi

Lab Program 12:

Display Map based on the Current/given location.

Steps: Click Start- Android Studio, a Welcome to Android Studio dialog box will appear.

Click New Project, the New Project Dialog box appears.

Choose Empty Views Activity then click Next. Specify the Name of your project, Select the Language as Java, and Select the SDK as API 24(“Nougat”,Android 7.0).

Click Finish Button.

Activity_main.xml

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


        <fragment
            android:id="@+id/map"
            android:name="com.google.android.gms.maps.SupportMapFragment"
            android:layout_width="match_parent"
            android:layout_height="match_parent"
            android:layout_alignParentTop="true"
            android:layout_alignParentBottom="true"
            android:layout_alignParentStart="true"
            android:layout_alignParentEnd="true" />



</RelativeLayout>
```

MainActivity.java

```
package com.example1.lab12;

import androidx.appcompat.app.AppCompatActivity;

import android.os.Bundle;
import android.os.Bundle;
import android.widget.Toast;
import androidx.annotation.NonNull;
import androidx.appcompat.app.AppCompatActivity;
import com.google.android.gms.maps.CameraUpdateFactory;
import com.google.android.gms.maps.GoogleMap;
import com.google.android.gms.maps.OnMapReadyCallback;
import com.google.android.gms.maps.SupportMapFragment;
import com.google.android.gms.maps.model.LatLng;
import com.google.android.gms.maps.model.MarkerOptions;


public class MainActivity extends AppCompatActivity implements
OnMapReadyCallback {
    private GoogleMap mMap;
    private double latitude = 0.0;
    private double longitude = 0.0;
    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_main);
        // Obtain the SupportMapFragment and get notified when the map is
ready to be used.
        SupportMapFragment mapFragment = (SupportMapFragment)
                getSupportFragmentManager()
                        .findFragmentById(R.id.map);
        if (mapFragment != null) {
            mapFragment.getMapAsync(this);
        } else {
            Toast.makeText(this, "Map Fragment Not Found",
Toast.LENGTH_SHORT).show();
        }
    }

    @Override
    public void onMapReady(@NonNull GoogleMap googleMap) {
        mMap = googleMap;
        // Add a marker at current or given location and move the camera
        LatLng location = new LatLng(latitude, longitude);
        mMap.addMarker(new
MarkerOptions().position(location).title("Marker"));
        mMap.moveCamera(CameraUpdateFactory.newLatLngZoom(location, 15));
    }
}
```

Manifest.xml

```
<?xml version="1.0" encoding="utf-8"?>
<manifest xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:tools="http://schemas.android.com/tools">
    <usespermission android:name="android.permission.ACCESS_FINE_LOCATION"
/>
    <usespermission
android:name="android.permission.ACCESS_COARSE_LOCATION" />
    <usespermission 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.Lab12"
        tools:targetApi="31">



        <! Google Maps API Key >
        <metadata
            android:name="com.google.android.geo.API_KEY"
            android:value="YOUR_API_KEY_HERE" />

        <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>
```

Lab Program 13:

Create a sample application with login module (check user name and password) On successful login change Textview “Login Successful”. On login fail alert using Toast “login fail”

Steps: 1. Click Start- Android Studio, a Welcome to Android Studio dialog box will appear. Click New Project, the New Project Dialog box appears.

2. Choose Empty Views Activity then click Next.

3. Specify the Name of your project, Select the Language as Java, and Select the SDK as API 24(“Nougat”,Android 7.0).Click Finish Button.

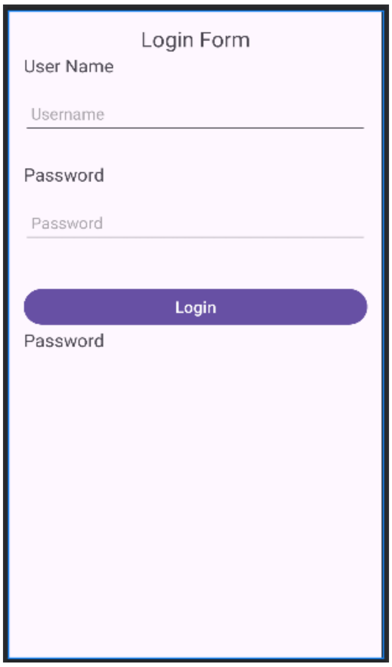
4. Update the following code in activity_main.xml and MainActivity.java

5. Click Run app or shift+F10 to execute the application.

Activity_main.xml

```
<?xml version="1.0" encoding="utf-8"?>
<LinearLayout xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:app="http://schemas.android.com/apk/res-auto"
    xmlns:tools="http://schemas.android.com/tools"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    tools:context=".MainActivity"
    android:orientation="vertical"
    android:padding="16dp">

    <TextView android:id="@+id/tvTitle"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:textSize="24sp"
        android:text="Login Form"
        android:layout_gravity="center"/>
    <TextView android:id="@+id/tvUserName"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:textSize="20sp"
        android:text="User Name" />
    <EditText
        android:id="@+id/etUsername"
        android:layout_width="match_parent"
        android:layout_height="50dp"
        android:hint="Username"
        android:inputType="text"
        android:padding="8dp"
        android:layout_marginTop="16dp"
        android:layout_marginBottom="30dp"/>
    <TextView android:id="@+id/tvPassword"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:textSize="20sp"
        android:text="Password" />
    <EditText
        android:id="@+id/etPassword"
        android:layout_width="match_parent"
        android:layout_height="50dp"
        android:hint="Password"
        android:inputType="textPassword"
        android:padding="8dp"
        android:layout_marginTop="16dp"
        android:layout_marginBottom="30dp"/>

    <Button
        android:id="@+id/btnLogin"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:layout_marginTop="16dp"
        android:text="Login"
        android:textSize="18sp" />

    <TextView android:id="@+id/tvMessage"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:textSize="20sp"
        android:text="Password" />

</LinearLayout>
```

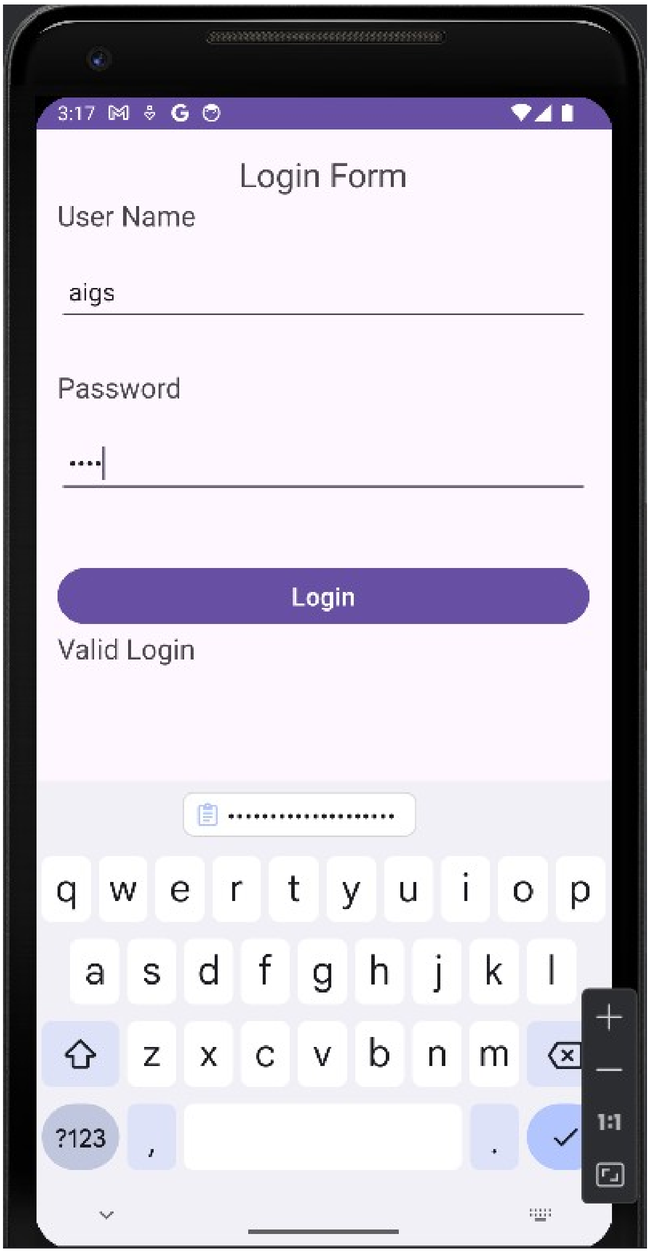
Design:



MainActivity.java

```
package com.example1.lab13;

import androidx.appcompat.app.AppCompatActivity;

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

public class MainActivity extends AppCompatActivity {
    EditText etUsername,etPassword; Button btnLogin;
    TextView tvMessage;
    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_main);
        btnLogin = (Button) findViewById(R.id.btnLogin);
        etUsername = (EditText) findViewById(R.id.etUsername);
        etPassword = (EditText) findViewById(R.id.etPassword);
        tvMessage = (TextView) findViewById(R.id.tvMessage);
        btnLogin.setOnClickListener(new View.OnClickListener()
        { @Override
        public void onClick(View view)
        {
            if(etUsername.getText().toString().isEmpty())
            {
                etUsername.setError("Enter User name");
            } else if (etPassword.getText().toString().isEmpty()) {
                etPassword.setError("Enter Password");
            }
            else if(etUsername.getText().toString().equals("aigs") &&
                    etPassword.getText().toString().equals("aigs"))
            {
                tvMessage.setText("Valid Login");
            }
            else
            {
                tvMessage.setText("Invalid login");
            }
        }
        });
    }
```

Output:



Login Form

User Name

aigs

Password

••••

Valid Login

Lab Program 14:

Learn to deploy Android applications

Steps to Deploy an Android Application

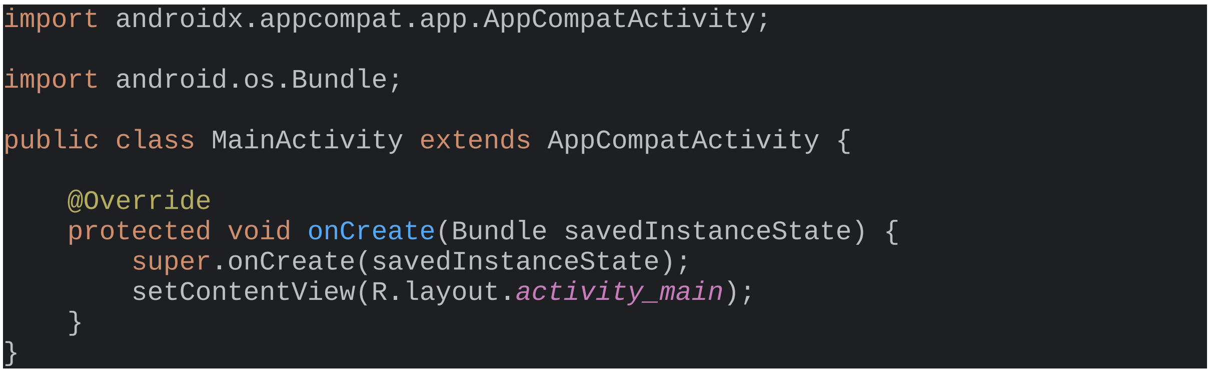
1. Prepare App (use Program 1 Hello world for this program)
   - Optimize performance and test thoroughly.
     - Ensure compatibility with various devices.

Activity_Main.xml

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

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

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

MainActivity.java

```
package com.example1.lab14;
```



```java
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);
    }
}
```

2.Generate Signed APK (Android Package Kit):

- In Android Studio, navigate to Build > Generate Signed Bundle/APK.

- Follow the prompts to create a new keystore or use an existing one. A keystore is a binary file that contains a set of private keys.

- Configure the build type (release) and signing configuration.

- Generate the signed APK file.

3.Test Your Signed APK:

- Before distributing your app, test the signed APK to ensure that the signing process didn't introduce any issues.

- Install the APK on various devices and perform thorough testing.

- Release on Google Play Console:

- Sign in to the Google Play Console (https://play.google.com/apps/publish).

- Create a new app entry if this is your first release or select an existing app.
- Complete all the required information for the app listing, including the title, description, screenshots, and categorization.
- Upload your signed APK file.
- Set pricing and distribution options.
- Optimize your store listing for search and conversion.
- Once everything is set, click the "Publish" button to release your app to the Google Play Store.

5. Other Distribution Channels (Optional):

- Besides Google Play, you can distribute your app through other channels such as Amazon Appstore, Samsung Galaxy Store, or thirdparty app marketplaces.
- Each distribution channel may have its own requirements and submission process, so be sure to follow their guidelines.

6. Monitor and Update:

- Keep an eye on user feedback and app performance metrics through the Google Play Console.
- Regularly update your app to fix bugs, add new features, and improve user experience based on feedback

Program 11

Create an application to send an Email.

Steps: 1. Click Start- Android Studio, a Welcome to Android Studio dialog box will appear. Click New Project, the New Project Dialog box appears.

2. Choose Empty Views Activity then click Next.

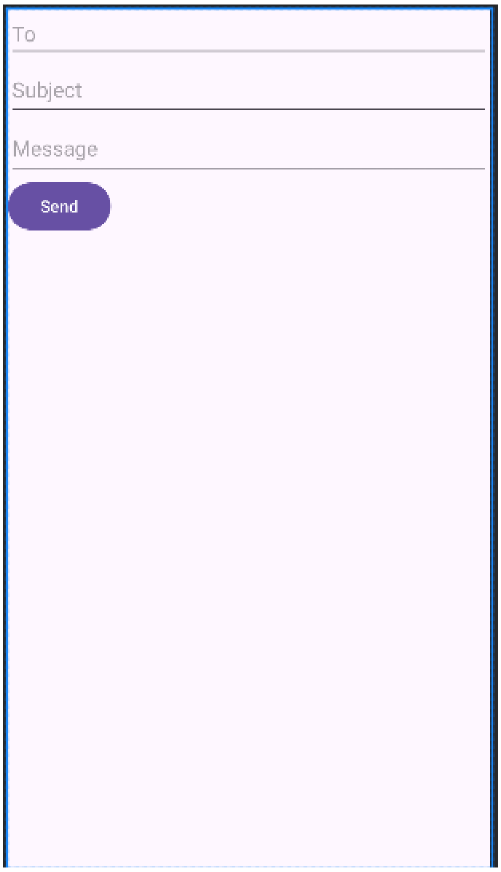
3. Specify the Name of your project, Select the Language as Java, and Select the SDK as API 24(“Nougat”,Android 7.0).Click Finish Button.

4. Update the following code in activity_main.xml and MainActivity.java

Activity_main.xml:

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

    <EditText
        android:id="@+id/editTextTo"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:hint="To"/>

    <EditText
        android:id="@+id/editTextSubject"
        android:layout_width="match_parent"
        android:layout_height="50dp"
        android:layout_below="@id/editTextTo"
        android:hint="Subject"/>

    <EditText
        android:id="@+id/editTextMessage"
        android:layout_width="match_parent"
        android:layout_height="50dp"
        android:layout_below="@id/editTextSubject"
        android:hint="Message"/>

    <Button
        android:id="@+id/buttonSend"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_below="@id/editTextMessage"
        android:text="Send"/>



</RelativeLayout>
```

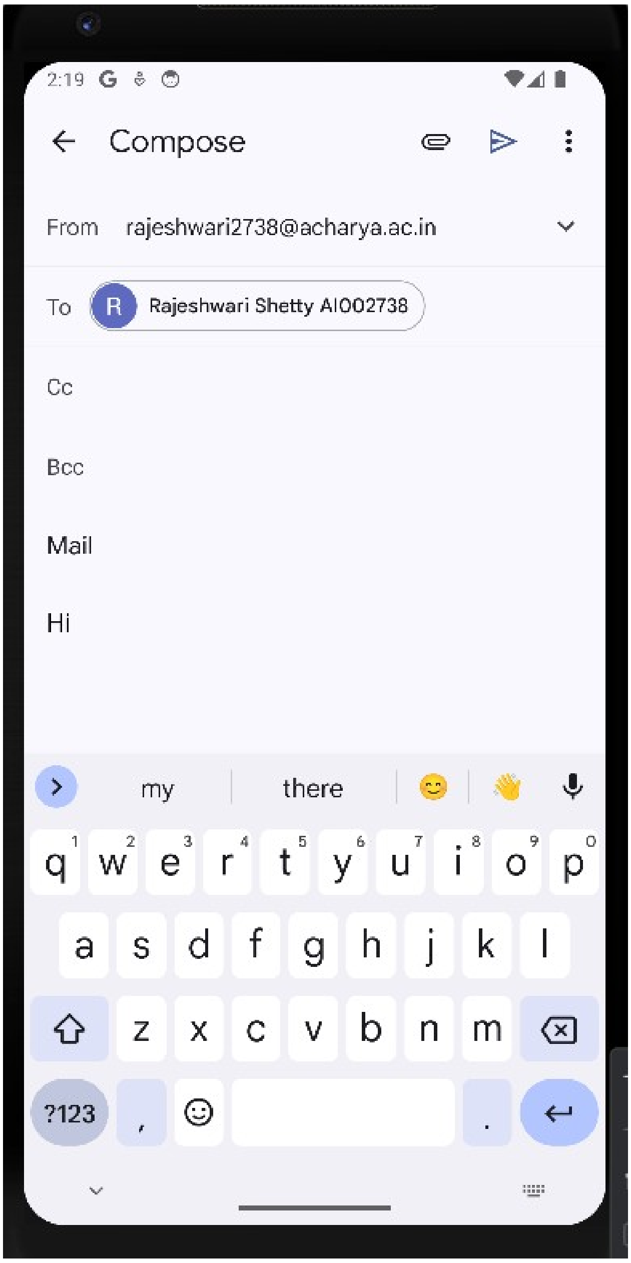
Design:



MainActivity.java:

```
package com.example1.lab11;

import androidx.appcompat.app.AppCompatActivity;

import android.os.Bundle;
import android.annotation.SuppressLint;
import android.content.Intent;
import android.view.View;
import android.widget.Button;
import android.widget.EditText;

public class MainActivity extends AppCompatActivity {
    EditText editTextTo, editTextSubject, editTextMessage;
    Button buttonSend;
    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_main);
        editTextTo = findViewById(R.id.editTextTo);
        editTextSubject = findViewById(R.id.editTextSubject);
        editTextMessage = findViewById(R.id.editTextMessage);
        buttonSend = findViewById(R.id.buttonSend);
        buttonSend.setOnClickListener(new View.OnClickListener(){

            @Override
            public void onClick(View v) {
                sendEmail();
            }
        });

    }

    @SuppressLint("QueryPermissionsNeeded")
    private void sendEmail() {
        String to = editTextTo.getText().toString().trim();
        String subject = editTextSubject.getText().toString().trim();
        String message = editTextMessage.getText().toString().trim();
        Intent intent = new Intent(Intent.ACTION_SEND);
        intent.setType("text/plain");
        intent.putExtra(Intent.EXTRA_EMAIL, new String[]{to});
        intent.putExtra(Intent.EXTRA_SUBJECT, subject);
        intent.putExtra(Intent.EXTRA_TEXT, message);
        if (intent.resolveActivity(getPackageManager()) != null)
        {
            startActivity(Intent.createChooser(intent, "Choose an email
client"));
        }
    }
}
```

Output:



2:19

Compose

From rajeshwari2738@acharya.ac.in

To Rajeshwari Shetty AI002738

Cc

Bcc

Mail

Hi

Lab Program 14:

Learn to deploy Android applications

Steps to Deploy an Android Application

2. Prepare App (use Program 1 Hello world for this program)
   - Optimize performance and test thoroughly.
     - Ensure compatibility with various devices.

Activity_Main.xml

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

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

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

MainActivity.java

```
package com.example1.lab14;

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);
    }
}
```

2.Generate Signed APK (Android Package Kit):

- In Android Studio, navigate to Build > Generate Signed Bundle/APK.
- Follow the prompts to create a new keystore or use an existing one. A keystore is a binary file that contains a set of private keys.
- Configure the build type (release) and signing configuration.
- Generate the signed APK file.

3.Test Your Signed APK:

- Before distributing your app, test the signed APK to ensure that the signing process didn't introduce any issues.
- Install the APK on various devices and perform thorough testing.
- Release on Google Play Console:
- Sign in to the Google Play Console (https://play.google.com/apps/publish).
- Create a new app entry if this is your first release or select an existing app.
- Complete all the required information for the app listing, including the title, description, screenshots, and categorization.
- Upload your signed APK file.
- Set pricing and distribution options.
- Optimize your store listing for search and conversion.

- Once everything is set, click the "Publish" button to release your app to the Google Play Store.

5. Other Distribution Channels (Optional):

- Besides Google Play, you can distribute your app through other channels such as Amazon Appstore, Samsung Galaxy Store, or thirdparty app marketplaces.
- Each distribution channel may have its own requirements and submission process, so be sure to follow their guidelines.

6. Monitor and Update:

- Keep an eye on user feedback and app performance metrics through the Google Play Console.
- Regularly update your app to fix bugs, add new features, and improve user experience based on feedback.