Mobile Application Development Lab

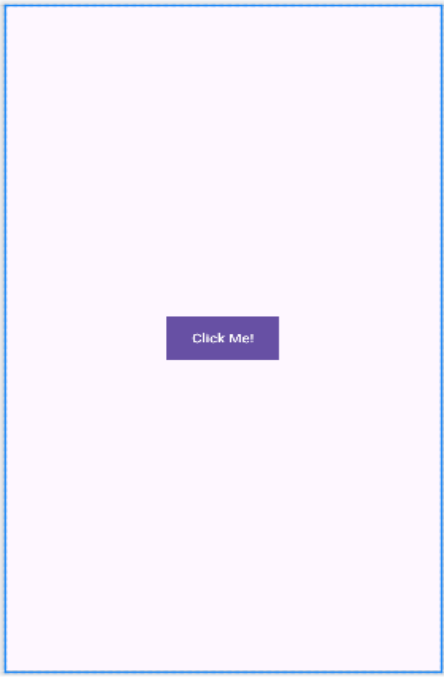
1. Creating “Hello world” Application.
2. Creating an application that displays message based on the screen orientation.
3. Create an application to develop Login window using UI controls.
4. Create an application to implement new activity using explicit intent, implicit intent and content provider.
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

1. Creating “Hello world” Application.
    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 **Minimum SDK** as **API 16 (“Jelly Bean”, Android 4.1)**. Click **Finish** Button.
    4. Create a **Button** resource in **activity_main.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:layout_width="match_parent"
    android:layout_height="match_parent"
    tools:context=".MainActivity">
    <Button
        android:id="@+id/hello"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:background="#535538"
        android:text="Click Me!"
        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.

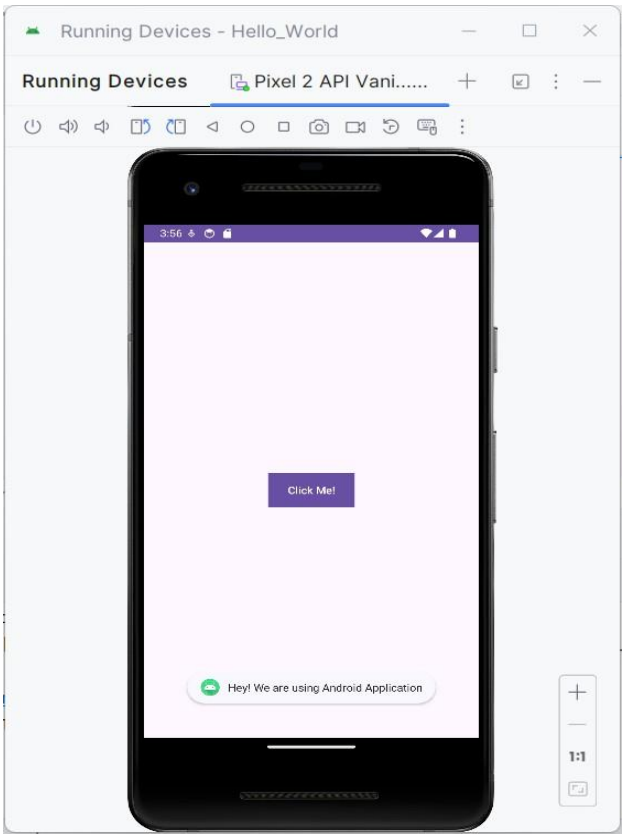


5. Create a **Button** object, create **clickListener**, **onClick** event and update the following code in **MainActivity.java**

```
package com.example.hello_world;
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.hello);
        b.setOnClickListener(new View.OnClickListener() {
            @Override
            public void onClick(View v) {
                Toast.makeText(MainActivity.this, "Hey! We are using Android Application", Toast.LENGTH_SHORT).show();
            }
        });
    }
}
```

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

Output:



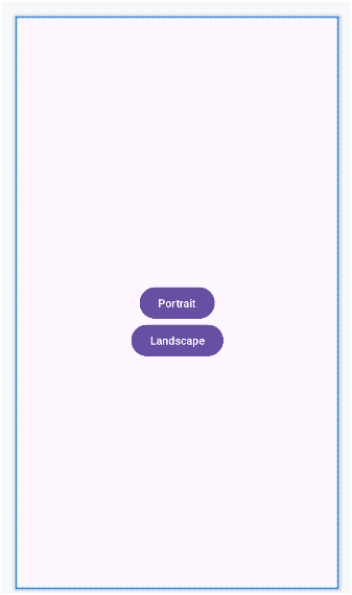
Program 2

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

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 **Minimum SDK** as **API 16 ("Jelly Bean", Android 4.1)**. Click **Finish** Button.
4. Create two **Button** resources in **activity_main.xml** 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">
    <Button
        android:id="@+id/por"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="Portrait"
        android:layout_centerInParent="true"/>
    <Button
        android:id="@+id/lan"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="Landscape"
        android:layout_below="@id/por"
        android:layout_centerInParent="true"/>
</RelativeLayout>
```

The following figure illustrates the design view of the application.

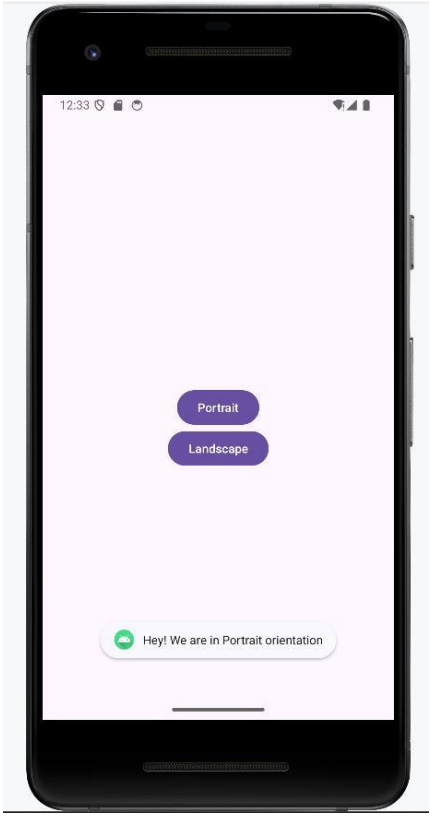


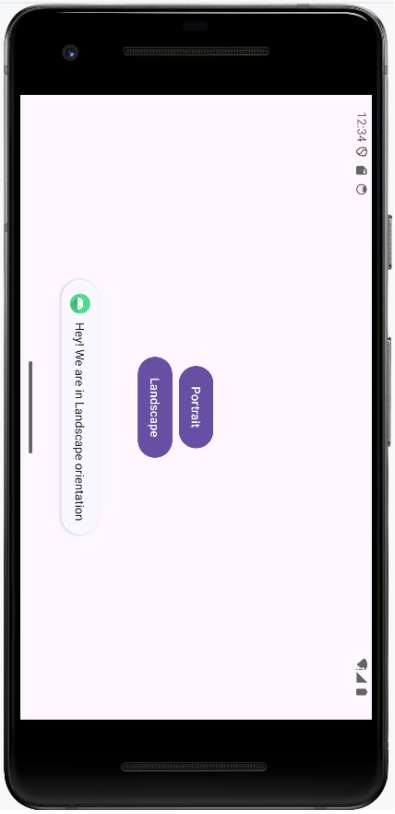
5. Create two **Button** object, create **clickListener**, **onClick** event and update the following code in **MainActivity.java**

```
package com.example.screen;

import android.content.pm.ActivityInfo;
import android.os.Bundle;
import android.view.View;
import android.widget.Button;
import android.widget.Toast;

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);
        Button l,p;
        l=findViewById(R.id.lan);
        p=findViewById(R.id.por);
        l.setOnClickListener(new View.OnClickListener() {
            @Override
            public void onClick(View v) {

setRequestedOrientation(ActivityInfo.SCREEN_ORIENTATION_LANDSCAPE);
                Toast.makeText(MainActivity.this, "Hey! We are in Landscape
orientation", Toast.LENGTH_SHORT).show();
            }
        });
        p.setOnClickListener(new View.OnClickListener() {
            @Override
            public void onClick(View v) {

setRequestedOrientation(ActivityInfo.SCREEN_ORIENTATION_PORTRAIT);
                Toast.makeText(MainActivity.this, "Hey! We are in Portrait orientation",
Toast.LENGTH_SHORT).show();
            }
        });

    }
}
```

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

Output





Program 3

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

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**)
    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">
    <corners android:radius="12dp" />
    <gradient
        android:startColor="#B388FF"
        android:endColor="#397C9A"
```

```
        android:angle="100"/>
    </shape>
```

c. Likewise, 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

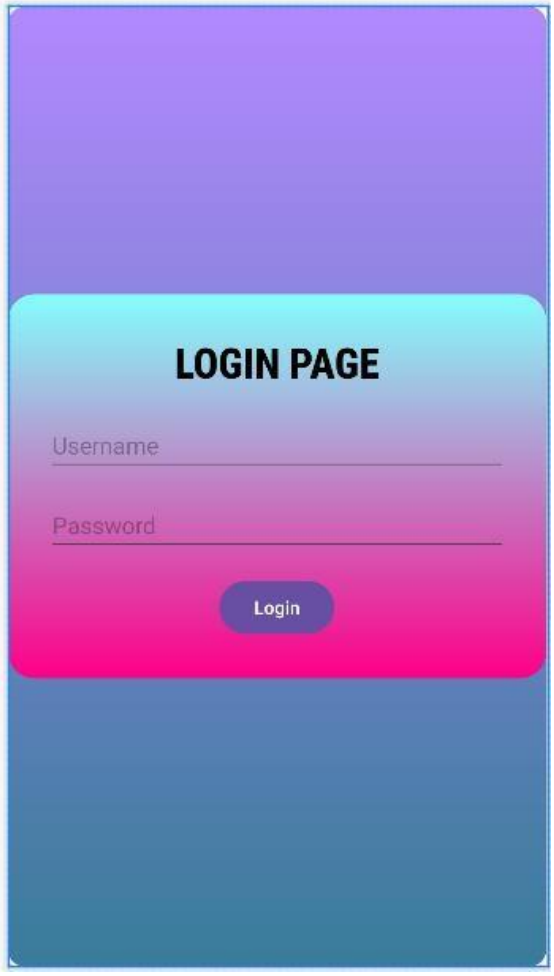
```
<?xml version="1.0" encoding="utf-8"?>
<shape xmlns:android="http://schemas.android.com/apk/res/android">
<gradient
   android:startColor="#84FFFF"
   android:endColor="#f08"
   android:angle="100"/>
   <corners android:radius="20dp"/>

</shape>
```

5. Create **two EditText box** and **a Button** resource in **activity_main.xml** 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:background="@drawable/bg_outer">

   <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"
      >
      <TextView
         android:layout_width="wrap_content"
         android:layout_height="wrap_content"
         android:text="LOGIN PAGE"
         android:textSize="32sp"
         android:textStyle="bold"
         android:fontFamily="sans-serif-condensed-medium"
         android:textColor="@color/black"
         android:paddingBottom="20dp"
         />

   <EditText
      android:id="@+id/editTextUsername"
      android:layout_width="match_parent"
```

```
        android:layout_height="wrap_content"
        android:hint="Username"
        android:layout_marginBottom="16dp"/>

    <EditText
        android:id="@+id/editTextPassword"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:hint="Password"
        android:layout_below="@id/editTextUsername"
        android:layout_marginBottom="16dp"
        android:inputType="textPassword"/>

    <Button
        android:id="@+id/buttonLogin"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="Login"
        android:layout_below="@id/editTextPassword"/>
    </LinearLayout>
</RelativeLayout>
```

The following figure illustrates the design view of the application.

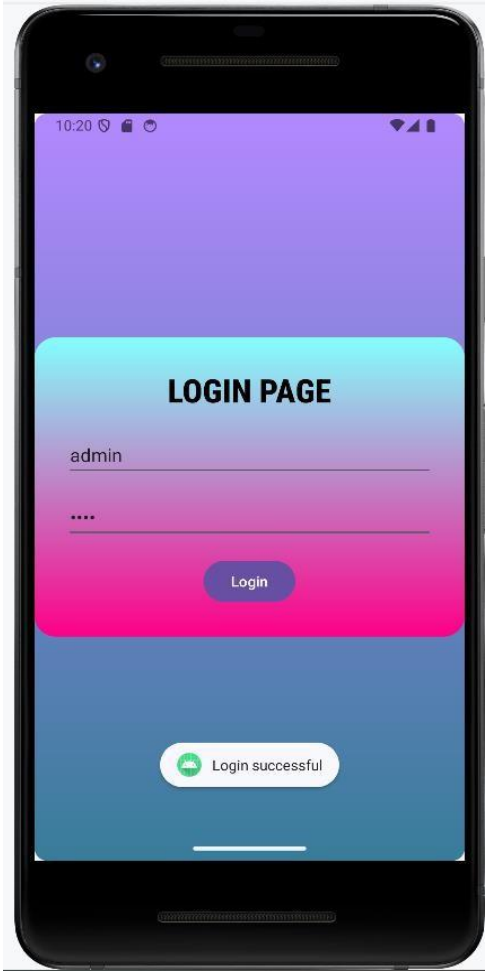


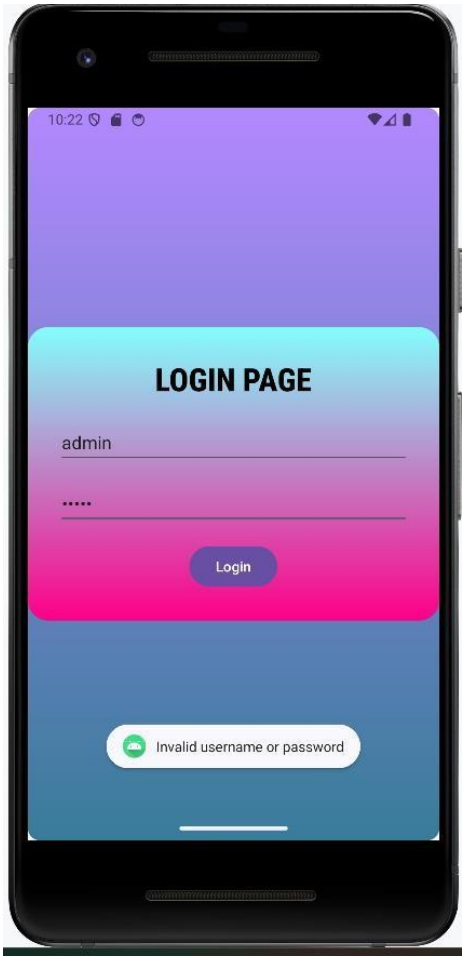
6. Create two **EditText** and a **Button** object, create **clickListener**, **onClick** event for button object and update the following code in **MainActivity.java**

```
package com.example.controls;

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

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 EditText editTextUsername,editTextPassword;
    private Button buttonLogin;
    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        EdgeToEdge.enable(this);
        setContentView(R.layout.activity_main);

        editTextUsername = findViewById(R.id.editTextUsername);
        editTextPassword = findViewById(R.id.editTextPassword);
        buttonLogin = findViewById(R.id.buttonLogin);
        buttonLogin.setOnClickListener(new View.OnClickListener() {
            @Override
            public void onClick(View v) {
                String username = editTextUsername.getText().toString().trim();
                String password = editTextPassword.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();
                }
            }
        });

    }

}
```

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

Output





Program 4

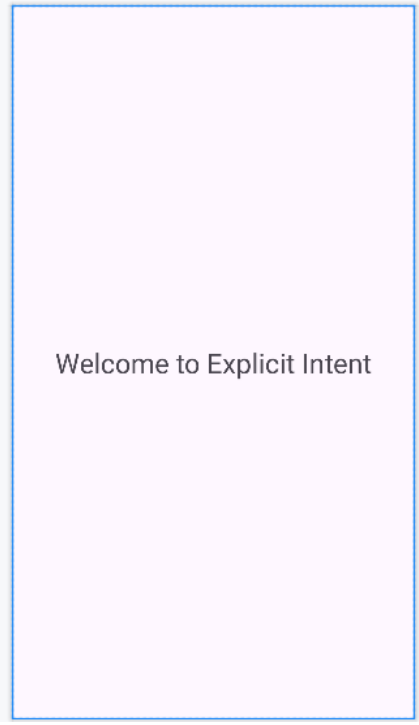
4. Create an application to implement new activity using explicit intent, implicit intent and content provider.

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** as **API 16 ("Jelly Bean", Android 4.1)**. 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**).



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

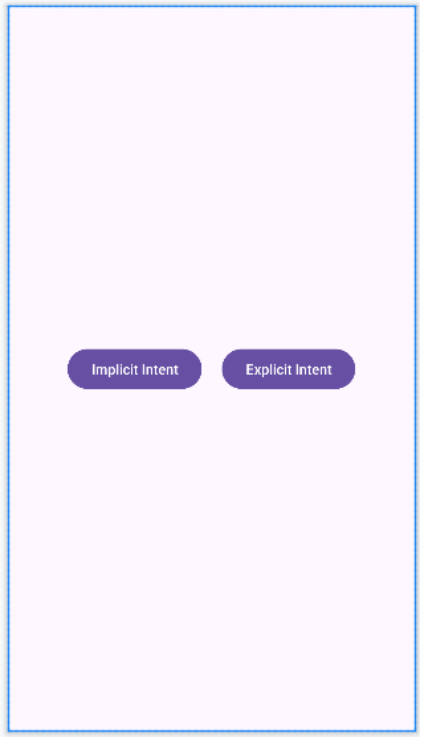
```
package com.example.intentexample;

import android.content.Intent;
import android.net.Uri;
import android.os.Bundle;
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);
    }
}
```

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>
```

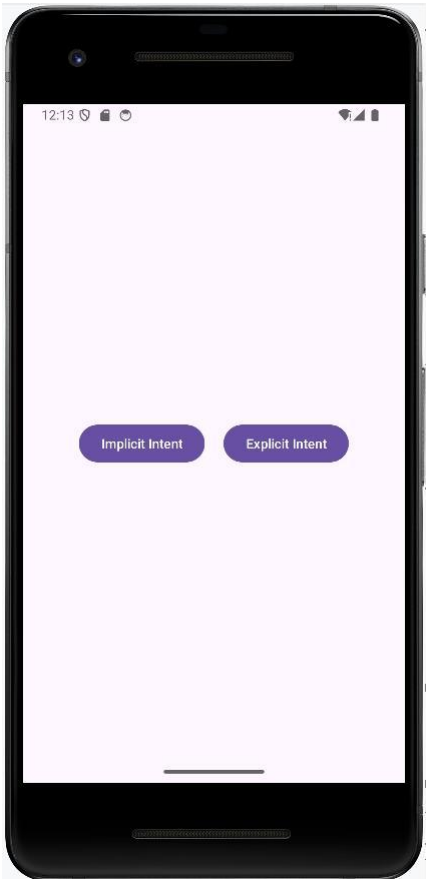
The following figure illustrates the design view of the application(**activity_main.xml**).



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

Output

For Implicit Intent



For Explicit Intent

Program 5

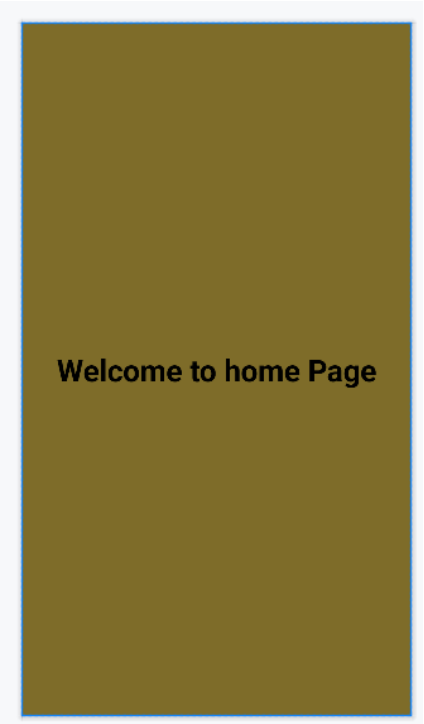
5. Create an application that displays custom designed Opening Screen.

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 **API 16 ("Jelly Bean", Android 4.1)**. 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** 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=".mainscreen"
    android:gravity="center"
```

```
    android:background="#7E6C29">
    <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>
```

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



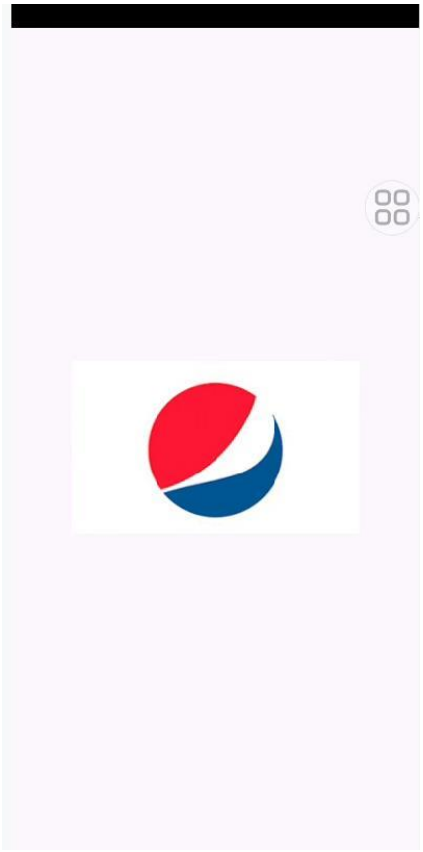
6. **To add an ImageView resource:** Copy an image and paste it into **drawable folder (**Right-click **Drawable→ Paste** the image**[img1.jpg]**).
7. Set an **image** as **src** in **activity_main.xml** and update the following code.

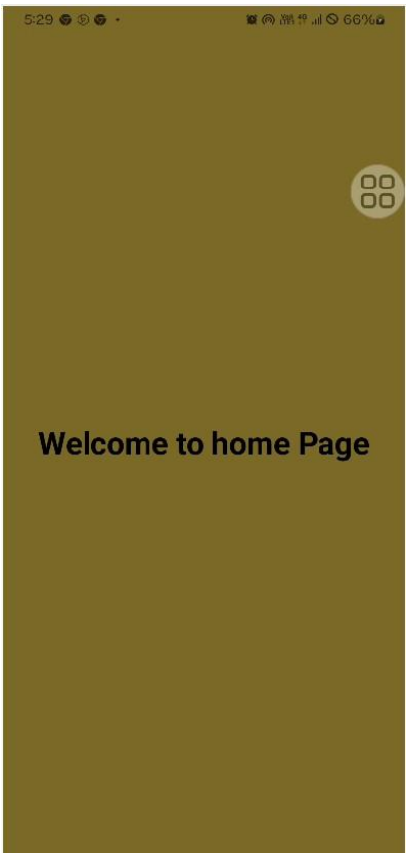
```
<?xml version="1.0" encoding="utf-8"?>
<RelativeLayout xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:app="http://schemas.android.com/apk/res-auto"
    xmlns:tools="http://schemas.android.com/tools"
    android:id="@+id/main"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    tools:context=".MainActivity"
    android:gravity="center">
    <ImageView
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:src="@drawable/img_1"/>

</RelativeLayout>
```

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

8. Update the following code in **MainActivity.java**

```
package com.example.pgm3;

import android.content.Intent;
import android.os.Bundle;
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);
    }
}
```





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.

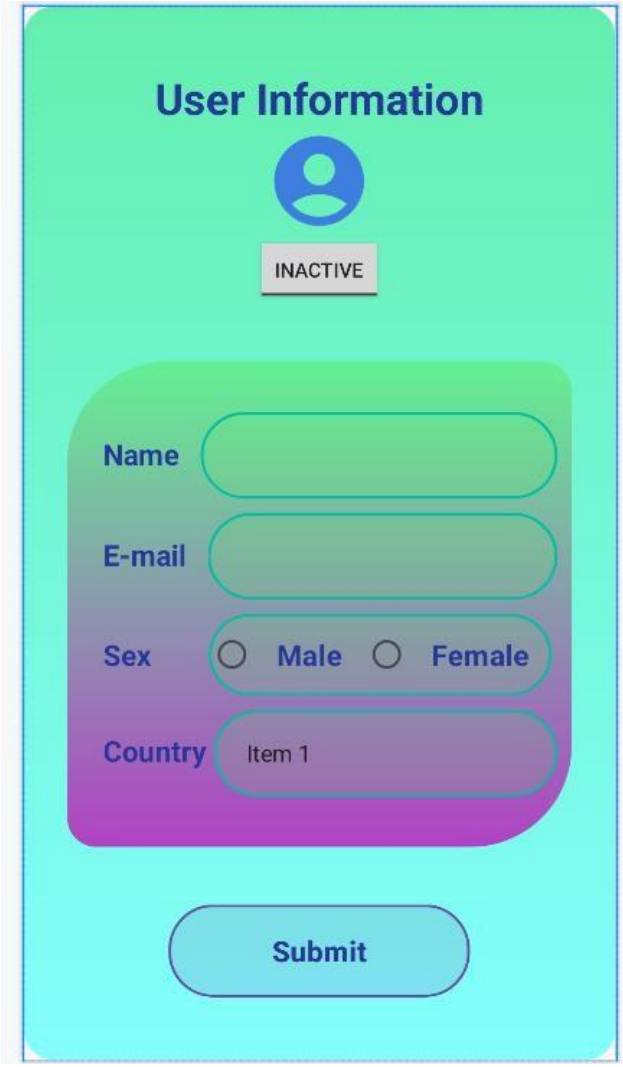
```
<?xml version="1.0" encoding="utf-8"?>
<LinearLayout xmlns:android="http://schemas.android.com/apk/res/android"
   xmlns:app="http://schemas.android.com/apk/res-auto"
   xmlns:tools="http://schemas.android.com/tools"
   android:id="@+id/main"
   android:layout_width="match_parent"
   android:layout_height="match_parent"
   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"/><ImageVie
      w android:layout_width="wrap_content"
      android:layout_height="wrap_content"
      android:src="@drawable/account_img"/>
```

```
<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.



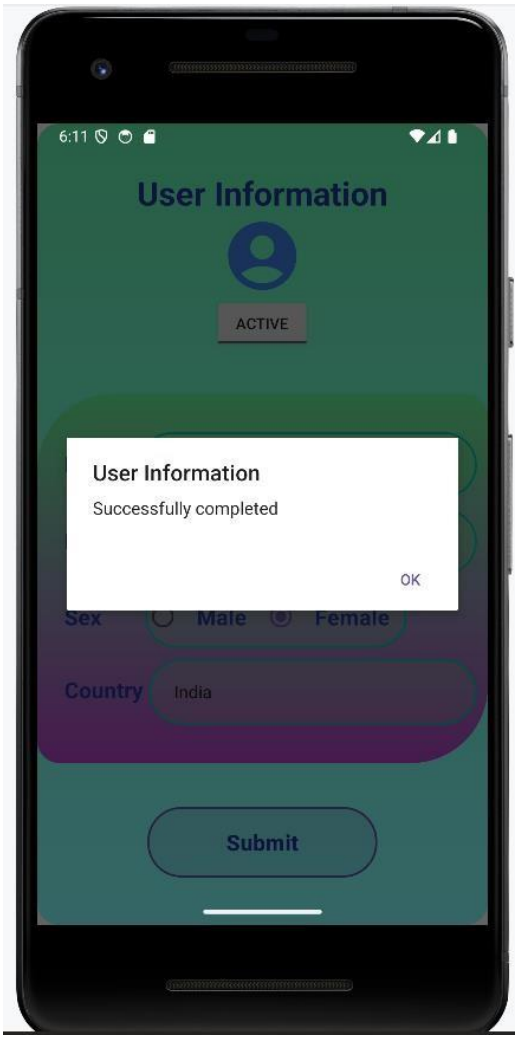
6. Create two **EditText** and a **Button** object, create **clickListener**, **onClick** event for button object and update the following code in **MainActivity.java**

```
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



Program 7

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 **Minimum SDK** as **API 16 ("Jelly Bean", Android 4.1)**. 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 **HomeScreen** then click **Finish**.
    5. **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**, Root element as **Menu** and update the following content.

```
<?xml version="1.0" encoding="utf-8"?>
<menu 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>
```

The menu design is as follows:

6. Update the following code in **MainActivity.java**

```
package com.example.menuexample;

import android.content.Intent;
import android.os.Bundle;
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.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);
    }

    @Override
    public boolean onCreatePanelMenu(int featureId, @NonNull Menu menu) {
        MenuInflater inflater=getMenuInflater();
        inflater.inflate(R.menu.menus,menu);
        return true;

    }

    @Override
    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);
    }
}
```

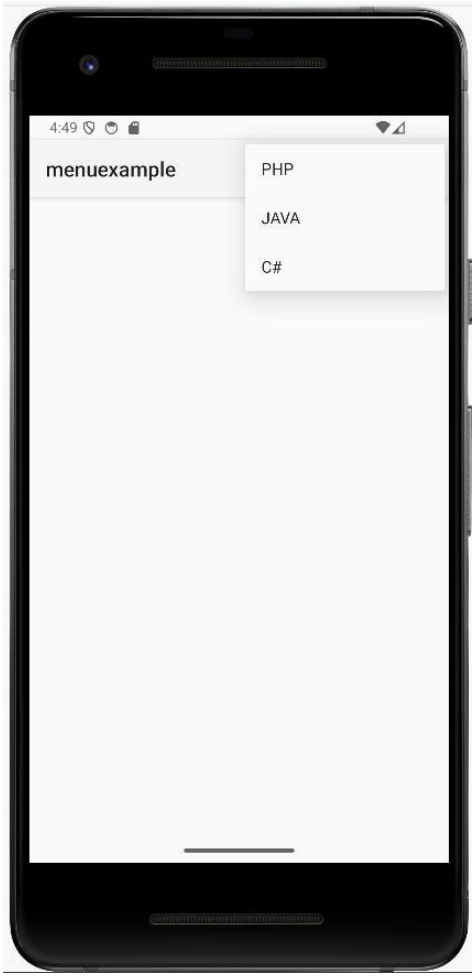
7. Set the **Uses-Permission** in **AndroidManifest.xml**

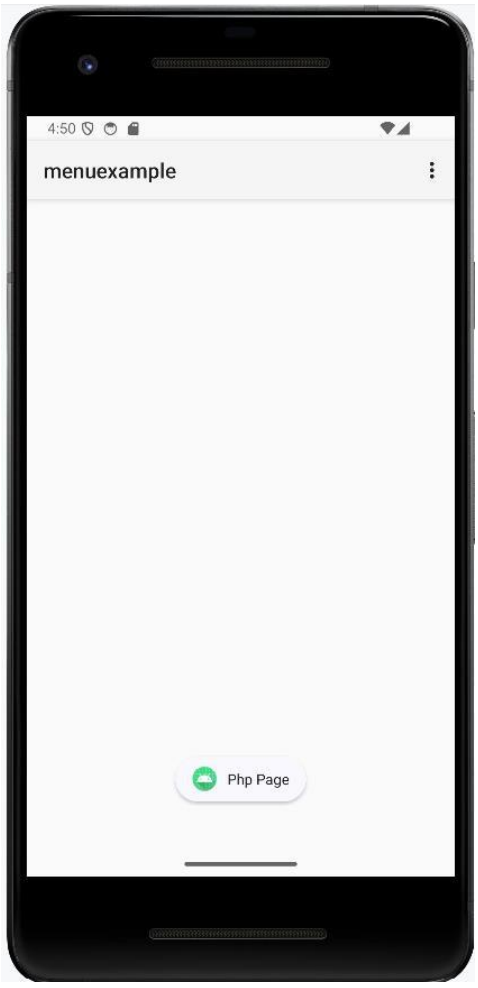
```
<?xml version="1.0" encoding="utf-8"?>
<manifest xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:tools="http://schemas.android.com/tools">

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

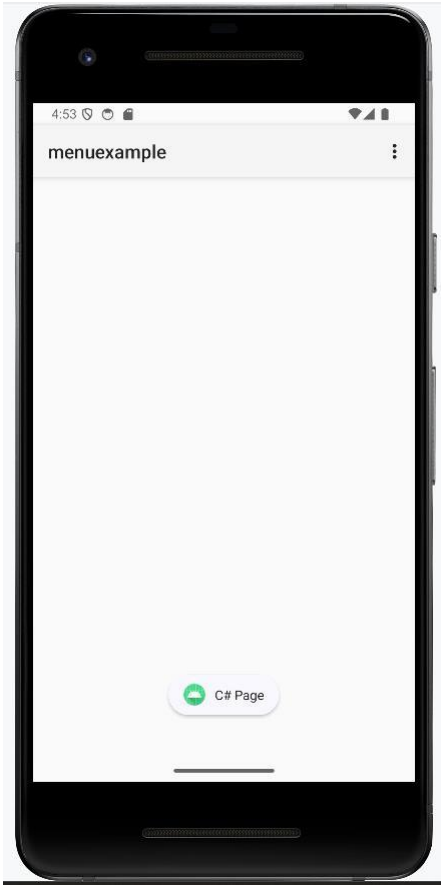
```
            <category android:name="android.intent.category.LAUNCHER" />
        </intent-filter>
    </activity>
</application>

</manifest>
```

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

Output







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 8

Read/ write the Local data

activity_main.xml

<?xml version="1.0" encoding="utf8"?>

```
<LinearLayout xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:app="http://schemas.android.com/apk/resauto"
    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="match_parent"
        android:layout_height="wrap_content"
        android:text="User Name"></TextView>
    <EditText
        android:id="@+id/etUserName"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        ></EditText>
    <TextView
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:text="Password"></TextView>
    <EditText
        android:id="@+id/etPassword"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        ></EditText>
    <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_width="wrap_content"
        android:layout_height="wrap_content"
```

```
        android:text="Next" />
</LinearLayout>
```

MainActivity.java

```
package com.bca.localdata;
import androidx.appcompat.app.AppCompatActivity;
import android.content.Context;
import android.content.Intent;
import android.content.SharedPreferences;
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 {
    Button btnsave,btnnext;
    EditText etUserName,etPassword;
    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_main);
        btnsave=(Button) findViewById(R.id.btnsave);
        btnnext = (Button) findViewById(R.id.btnnext);
        etUserName = (EditText)findViewById(R.id.etUserName);
        etPassword = (EditText)findViewById(R.id.etPassword);
        btnsave.setOnClickListener(new View.OnClickListener() { @Override
        public void onClick(View view) {
// Writing data to SharedPreferences
   SharedPreferences sharedPreferences = getSharedPreferences("MyPrefs",
Context.MODE_PRIVATE);
            SharedPreferences.Editor editor = sharedPreferences.edit();
            editor.putString("username",
                    etUserName.getText().toString());
            editor.putString("password",
```

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

    });

    btnnext.setOnClickListener(new View.OnClickListener() { @Override
    public void onClick(View view) {

      Intent intent = new
            Intent(getApplicationContext(),MainActivity2.class);

       startActivity(intent);
    }

    });
  }

}
```

activity_main2.xml

```
<?xml version="1.0" encoding="utf8"?>
<LinearLayout xmlns:android="http://schemas.android.com/apk/res/android"
  xmlns:app="http://schemas.android.com/apk/resauto"
  xmlns:tools="http://schemas.android.com/tools"
  android:layout_width="match_parent"
  android:layout_height="match_parent"
  tools:context=".MainActivity2"
  android:orientation="vertical">
```

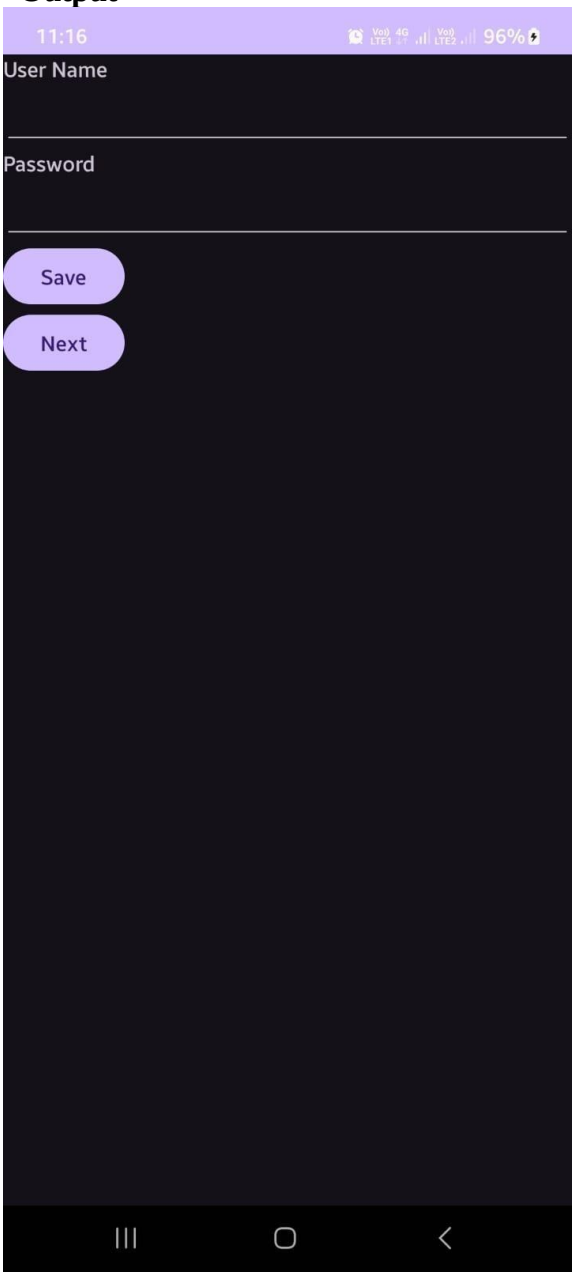
```
    <Button android:id="@+id/btnFetch"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="Fetch" />
    <TextView
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:text="User Name"></TextView>
    <EditText
        android:id="@+id/etUserName"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        ></EditText>
    <TextView
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:text="Password"></TextView>
    <EditText
        android:id="@+id/etPassword"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        ></EditText>
</LinearLayout>
```

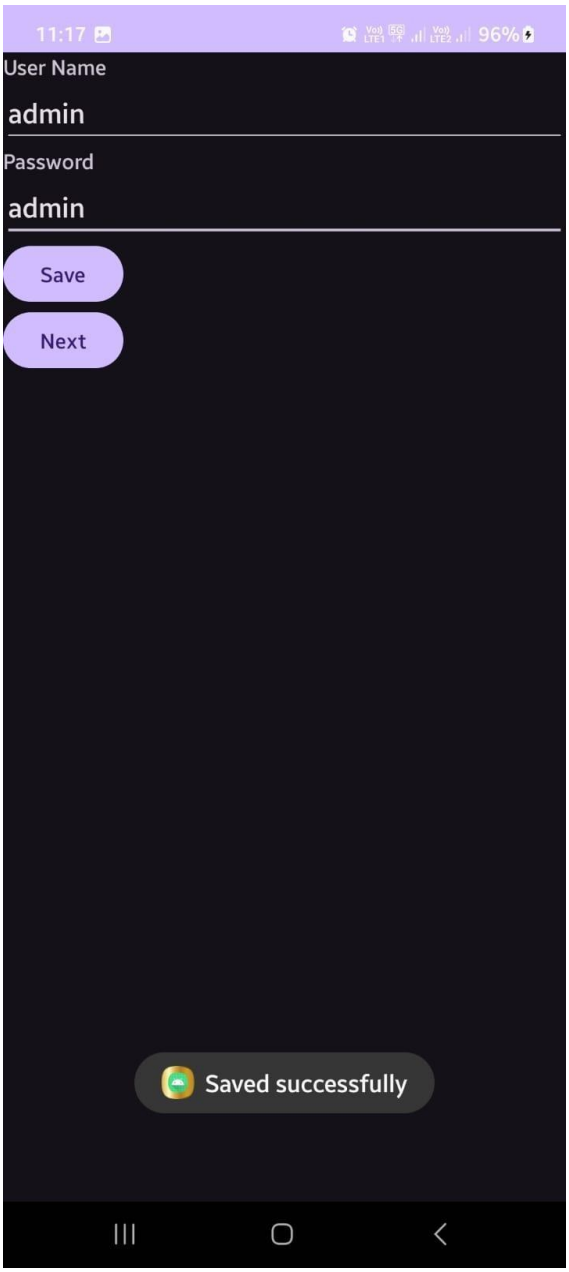
MainActivity2.java

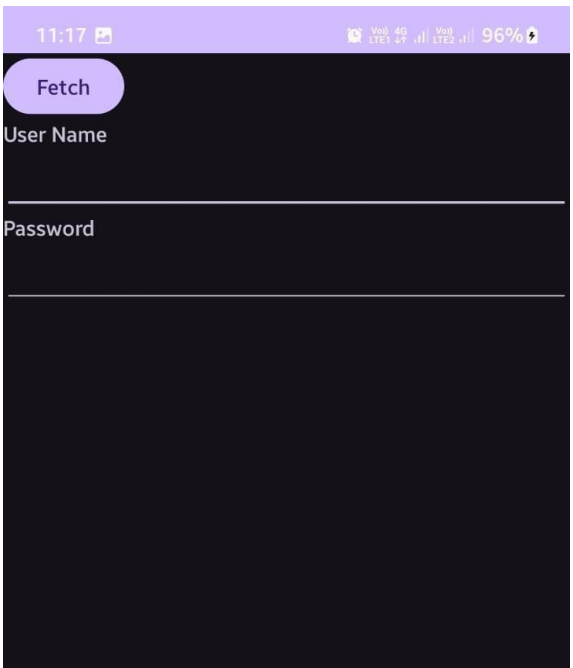
```
package com.bca.localdata;
import androidx.appcompat.app.AppCompatActivity;
import android.content.Context;
import android.content.SharedPreferences;
import android.os.Bundle;
import android.view.View;
```

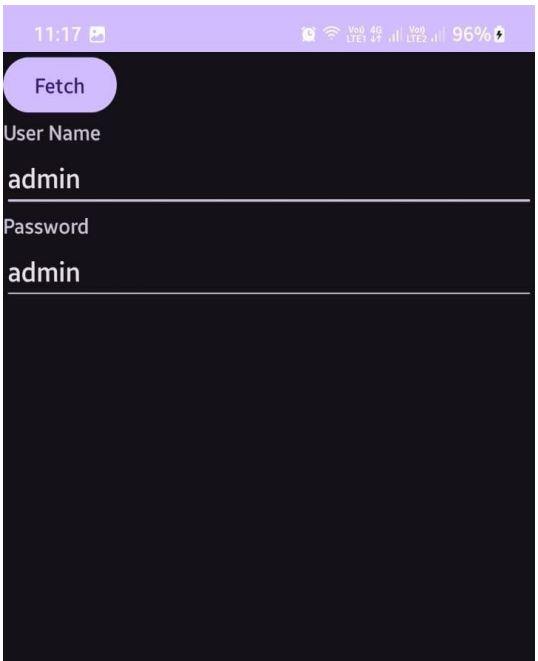
```
import android.widget.Button;
import android.widget.EditText;
public class MainActivity2 extends AppCompatActivity {

    Button btnFetch;
    EditText etUserName,etPassword;

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

       btnFetch = (Button) findViewById(R.id.btnFetch);
       etUserName = (EditText)findViewById(R.id.etUserName);
       etPassword = (EditText)findViewById(R.id.etPassword);
       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", "");
           etUserName.setText(username);
etPassword.setText(password);
   }
        });
     }
}
```

Output


11:16
96%
User Name
Password
Save
Next


11:17
96%
User Name
admin
Password
admin
Save
Next
Saved successfully


11:17
96%
Fetch
User Name
Password


11:17
96%
Fetch
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 version="1.0" encoding="utf-8"?>
<LinearLayout xmlns:android="http://schemas.android.com/apk/res/android"
xmlns:app="http://schemas.android.com/apk/res-auto"
xmlns:tools="http://schemas.android.com/tools"
android:layout_width="match_parent"
android:layout_height="match_parent"
android:orientation="vertical"android:gravity="center"
tools:context=".MainActivity">

<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/colorAccent"
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="Name"
/>

<EditText
android:layout_width="wrap_content"
android:layout_height="wrap_content"
android:layout_weight="1" android:ems="10"
android:id="@+id/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:layout_width="wrap_content"
android:layout_height="wrap_content"
android:layout_weight="1" android:ems="10"
android:id="@+id/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:layout_width="wrap_content"
android:layout_height="wrap_content"
android:layout_weight="1" android:ems="10"
android:id="@+id/fee"
android:textAlignment="center"
/>

</LinearLayout>

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

<Button
android:layout_width="wrap_content"
android:layout_height="wrap_content"
android:layout_weight="1" android:id="@+id/bt1"
android:text="Ok"
android:background="@color/colorPrimary"
/>

<Button
android:layout_width="wrap_content"
android:layout_height="wrap_content"
```

android:layout_weight="1" android:id="@+id/bt2"
android:text="View"
android:background="@color/colorAccent"
/>

</LinearLayout>
</LinearLayout>

MainActivity.java

```
package com.bca.sqlite;
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 ndroid.widget.EditText;
import android.widget.Toast;

public class MainActivity extends AppCompatActivity {

EditText ed1,ed2,ed3;

Button b1,b2;

@Override

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 addded",Toast.LENGTH_LONG).show();

ed1.setText("");

ed2.setText("");

ed3.setText("");

ed1.requestFocus();

}

catch (Exception ex)

{

Toast.makeText(this,"Record Fail",Toast.LENGTH_LONG).show();

}

}

}
```

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

Student.class

```
package com.bca.sqlite;

public class Student {

String id;
```

String **name**;

String **course**;

String **fee**;

String **titles**;

}

activity_view.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.bca.sqlite;

import androidx.appcompat.app.AppCompatActivity;

import ndroid.content.Context;
import android.content.Intent;

import ndroid.database.Cursor;

import android.database.sqlite.SQLiteDatabase;

import android.os.Bundle;

import android.view.View;

```
import  ndroid.widget.AdapterView;
import ndroid.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_view);

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 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/colorAccent"
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:layout_width="wrap_content"
android:layout_height="wrap_content"
android:layout_weight="1" android:ems="10"
android:id="@+id/id"
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="Name"
/>
<EditText
android:layout_width="wrap_content"
android:layout_height="wrap_content"
android:layout_weight="1" android:ems="10"
android:id="@+id/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:layout_width="wrap_content"
android:layout_height="wrap_content"
android:layout_weight="1" android:ems="10"
android:id="@+id/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:layout_width="wrap_content"
android:layout_height="wrap_content"
android:layout_weight="1"
android:ems="10"
android:id="@+id/fee"
android:textAlignment="center"
```

```
/>

</LinearLayout>
<LinearLayout
android:layout_width="wrap_content"
android:layout_height="wrap_content"
android:orientation="horizontal" android:gravity="center">
<Button
android:layout_width="wrap_content"
android:layout_height="wrap_content"
android:layout_weight="1" android:id="@+id/bt1"
android:text="Edit"
android:background="@color/colorPrimary"
/>

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

<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/colorPrimaryDark"
/>

</LinearLayout>
</LinearLayout>
```

EditActivity.java

```
package com.bca.sqlite;
```

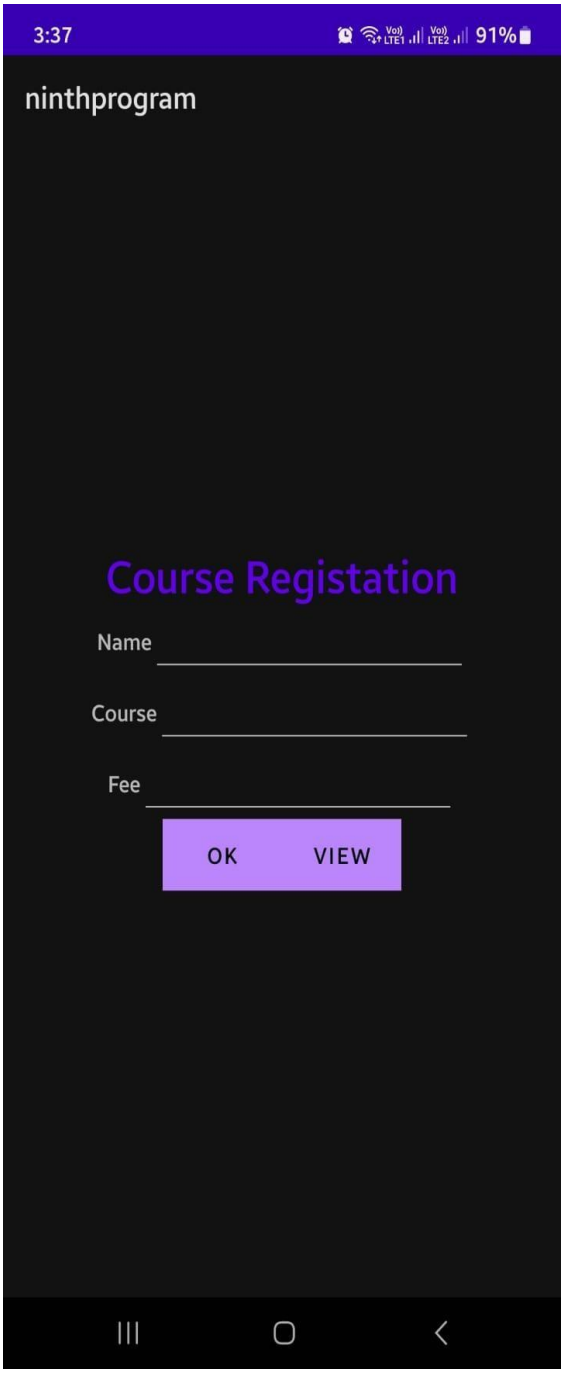
```
import androidx.appcompat.app.AppCompatActivity;

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 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);
```

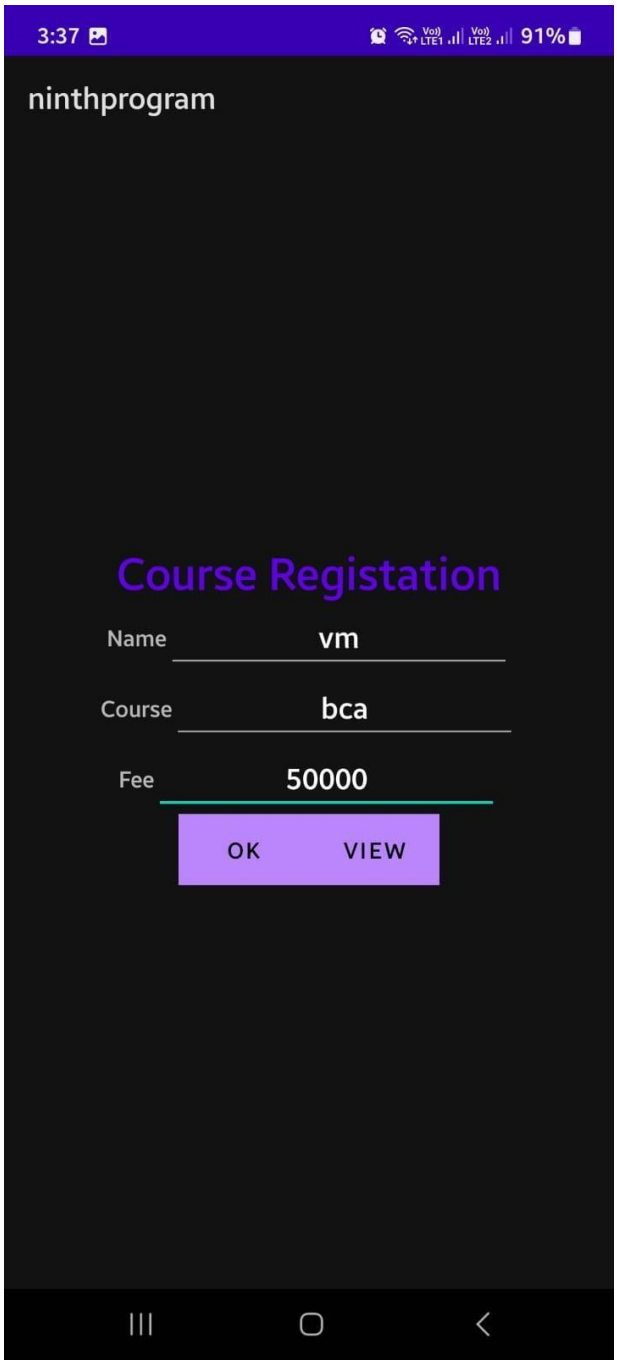
```
Intent i = getIntent();

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();

}
```

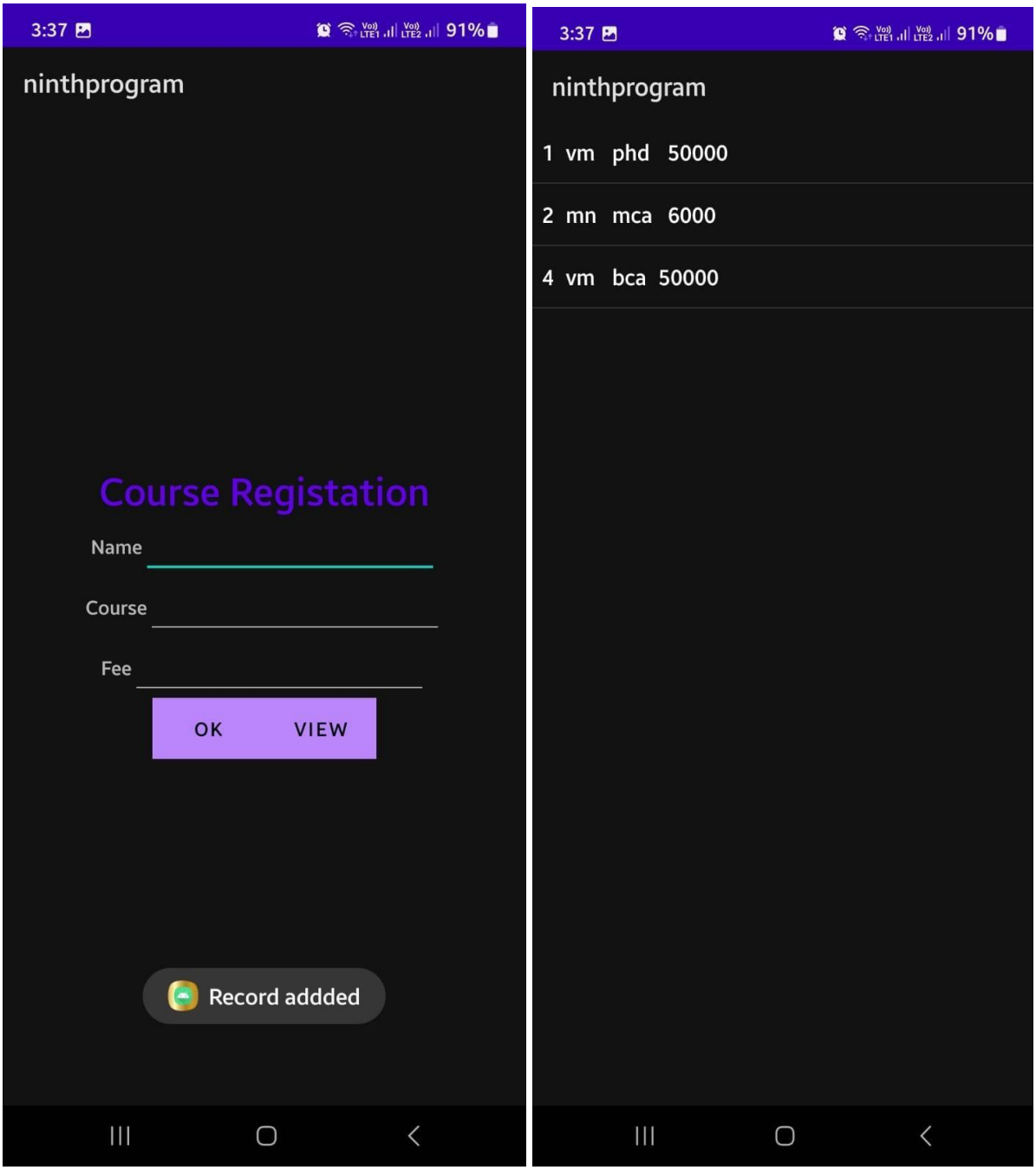
```
catch (Exception ex)
{
Toast.makeText(EditActivity.this,"Record  Fail",Toast.LENGTH_LONG).show();
}
}
});
b3.setOnClickListener(new View.OnClickListener() {@Override
public void onClick(View v) {
Intent i = new Intent(getApplicationContext(),ViewActivity.class);
startActivity(i);
}
});
b1.setOnClickListener(new 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


3:37
91%
ninthprogram
Course Registation
Name
Course
Fee
OK
VIEW


3:37
91%
ninthprogram
Course Registation
Name
vm
Course
bca
Fee
50000
OK
VIEW



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.

activity_main.xml

```
<?xml version="1.0" encoding="utf-8"?>
<RelativeLayout xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:tools="http://schemas.android.com/tools"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    android:background="@color/white" tools:context=".MainActivity">
    <EditText
        android:id="@+id/editTextPhoneNumber"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:hint="Enter phone number"
        android:layout_margin="16dp"/>
    <EditText
        android:id="@+id/editTextMessage"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:hint="Enter message"
        android:layout_below="@id/editTextPhoneNumber"
        android:layout_margin="16dp"/>
    <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="wrap_content"
        android:layout_below="@id/buttonSend"
        android:layout_marginStart="16dp"
        android:layout_marginTop="16dp"
        android:layout_marginEnd="16dp"
        android:layout_marginBottom="16dp"
        android:textColor="@color/black" />
</RelativeLayout>
```

MainActivity.java

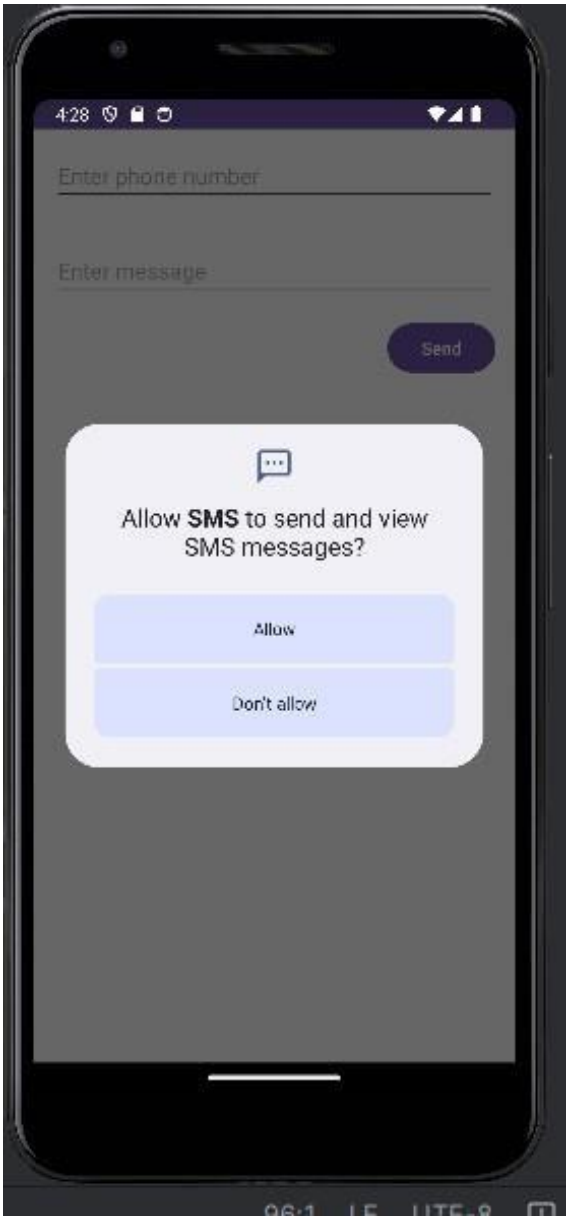
```
package com.bca.sms;
import androidx.appcompat.app.AppCompatActivity;
import androidx.core.app.ActivityCompat;
import androidx.core.content.ContextCompat;
import android.content.BroadcastReceiver;
import android.content.Context;
import android.content.Intent;
import android.content.IntentFilter;
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;
import androidx.appcompat.app.AppCompatActivity;

public class MainActivity extends AppCompatActivity
{
```

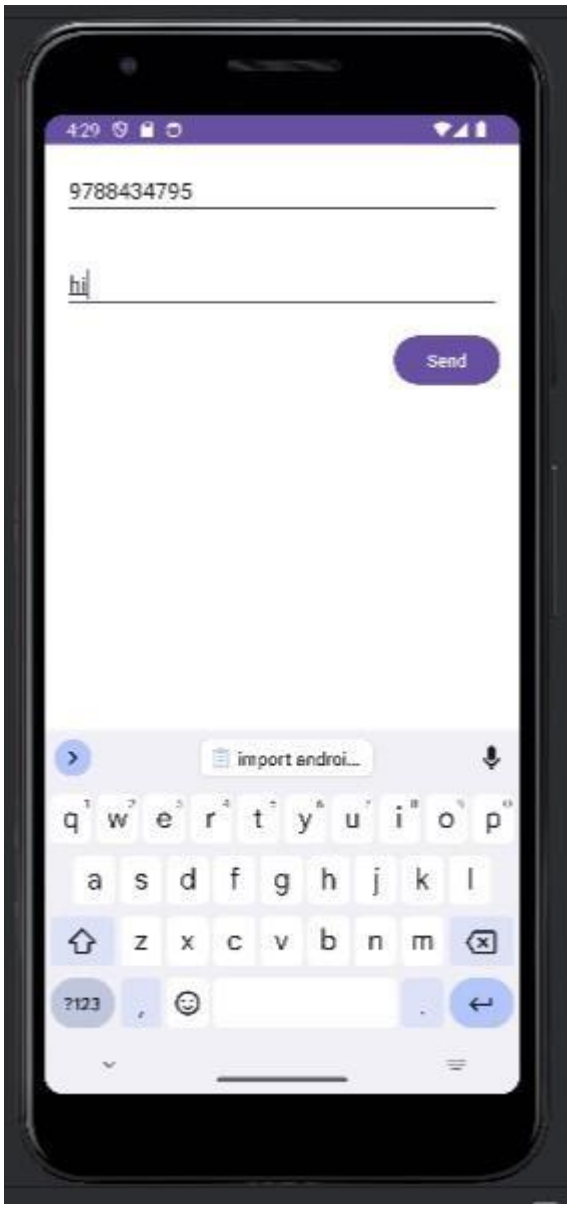
```
    private static final int SMS_PERMISSION_CODE = 101;
    private EditText editTextPhoneNumber;
    private EditText editTextMessage;
    private TextView textViewReceivedMessages;
    @Override
    protected void onCreate(Bundle savedInstanceState)
    {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_main);
        editTextPhoneNumber = findViewById(R.id.editTextPhoneNumber);
        editTextMessage = findViewById(R.id.editTextMessage);
        textViewReceivedMessages = 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);
    }
    @Override
    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();
               textViewReceivedMessages.append("From: " + senderPhoneNumber + "\n");
               textViewReceivedMessages.append("Message: " +
                     messageBody + "\n\n");
           }
        }
     }
  }
  };
}
```

Output


4:28
Enter phone number
Enter message
Send
Allow SMS to send and view SMS messages?
Allow
Don't allow


4:29
9788434795
hi
Send
import androi...

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**

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

activity_main.xml

```
<?xml version="1.0" encoding="utf8"?>
<RelativeLayout xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:tools="http://schemas.android.com/tools"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    tools:context=".MainActivity">

    <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="wrap_content"
      android:layout_below="@id/editTextTo"
      android:hint="Subject"/>
    <EditText
        android:id="@+id/editTextMessage"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        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"
```

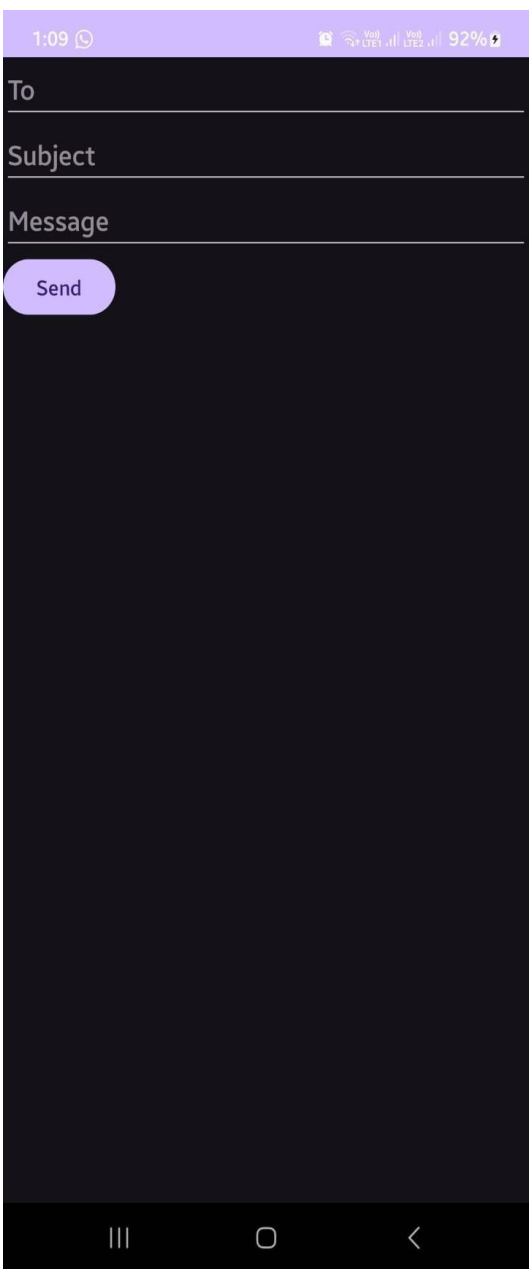
```
    android:text="Send"/>
</RelativeLayout>
```

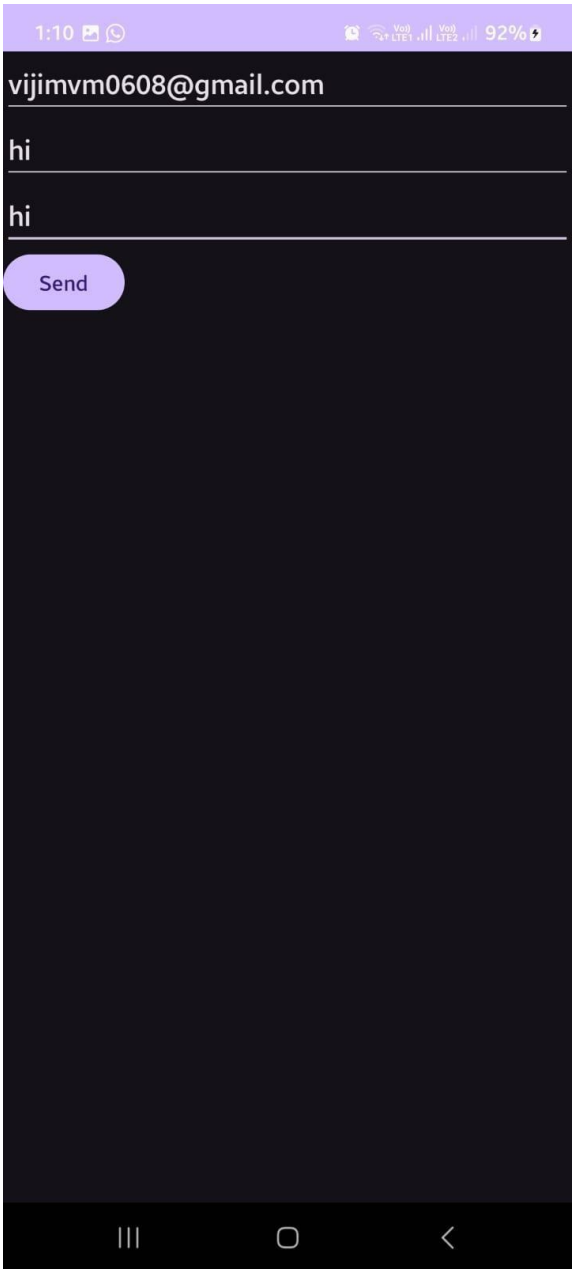
Mainactivity.java

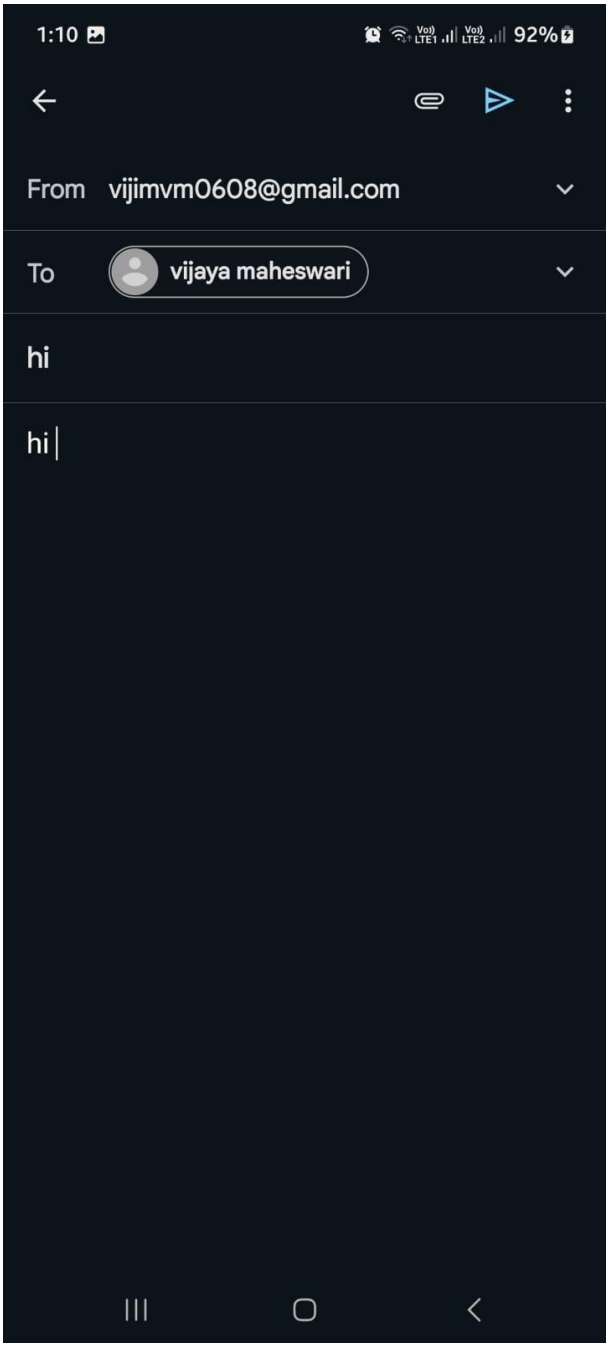
```
package com.bca.email;

import androidx.appcompat.app.AppCompatActivity;
import android.annotation.SuppressLint;
import android.content.Intent;
import android.os.Bundle;
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


1:09
92%
To
Subject
Message
Send


1:10
92%
vijimvm0608@gmail.com
hi
hi
Send



Program 12

Display Map based on the Current/given location.

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, Androidmanifest.xml and MainActivity.java**

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

MainActivity.java

```
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));
    }
}
```

activity_main.xml

```
<?xml version="1.0" encoding="utf8"?>
<RelativeLayout xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:tools="http://schemas.android.com/tools"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    tools:context=".MainActivity">
```

```
    <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>
```

AndroidManifest.xml

```
<?xml version="1.0" encoding="utf8"?>
<manifest xmlns:android="http://schemas.android.com/apk/res/android"
    package="com.yourpackage.yourappname">
    <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:icon="@mipmap/ic_launcher"
        android:label="@string/app_name"
        android:roundIcon="@mipmap/ic_launcher_round"
        android:supportsRtl="true"
        android:theme="@style/AppTheme">
```

```
    <! Google Maps API Key >
    <metadata
      android:name="com.google.android.geo.API_KEY"
      android:value="YOUR_API_KEY_HERE" />
    <activity android:name=".MainActivity">
      <intentfilter>
        <action android:name="android.intent.action.MAIN" />
        <category android:name="android.intent.category.LAUNCHER" />
      </intentfilter>
    </activity>
      </application>
</manifest>
```

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="wrap_content"
        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"
```

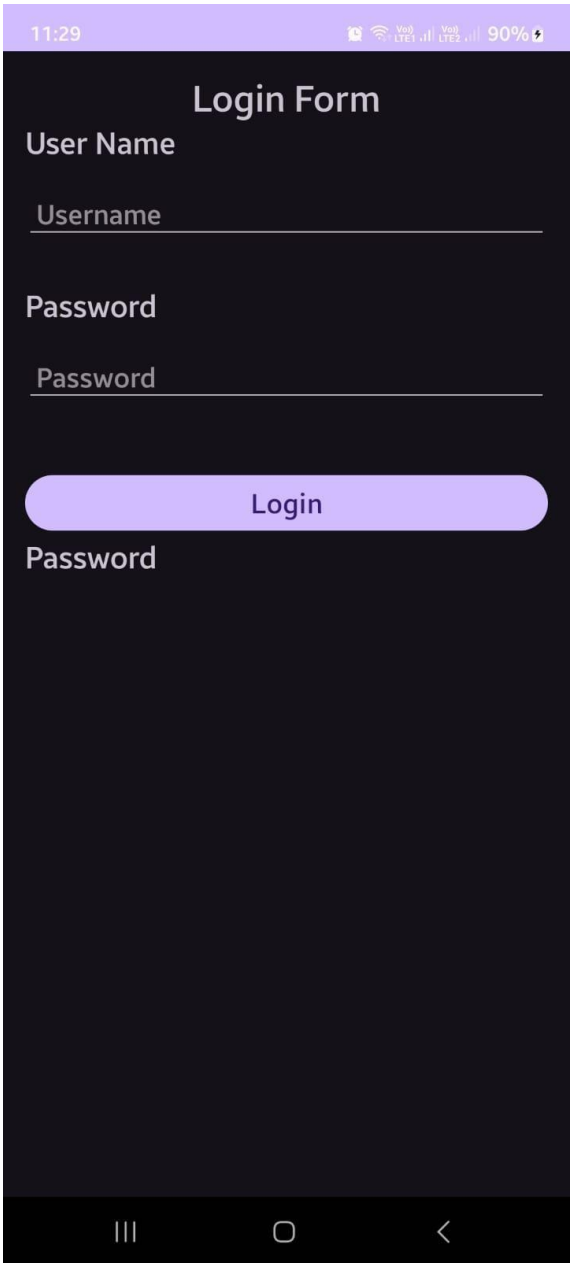
```
    android:text="Password" />
    <EditText
    android:id="@+id/etPassword"
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:hint="Password"
    android:inputType="textPassword"
    android:padding="8dp"
    android:layout_marginTop="16dp"
    android:layout_marginBottom="30dp"/>
    <Button
    android:id="@+id/btnLogin"
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:text="Login"
    android:textSize="18sp"
    android:layout_marginTop="16dp"/>
    <TextView android:id="@+id/tvMessage"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:textSize="20sp"
    android:text="Password" />
  </LinearLayout>
```

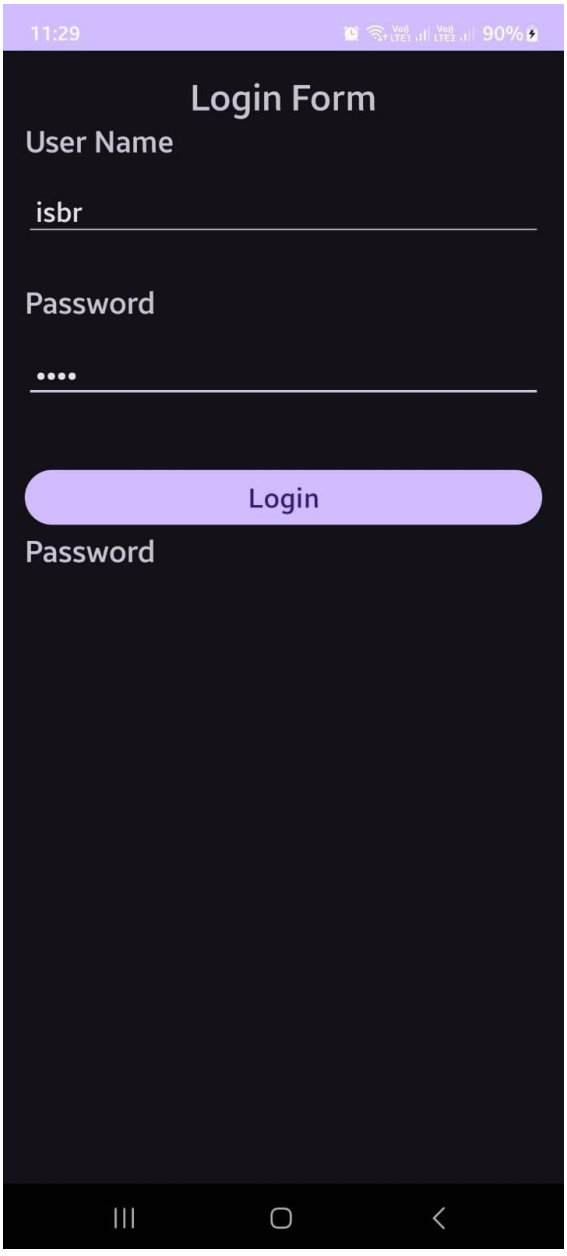
Mainactivity.java

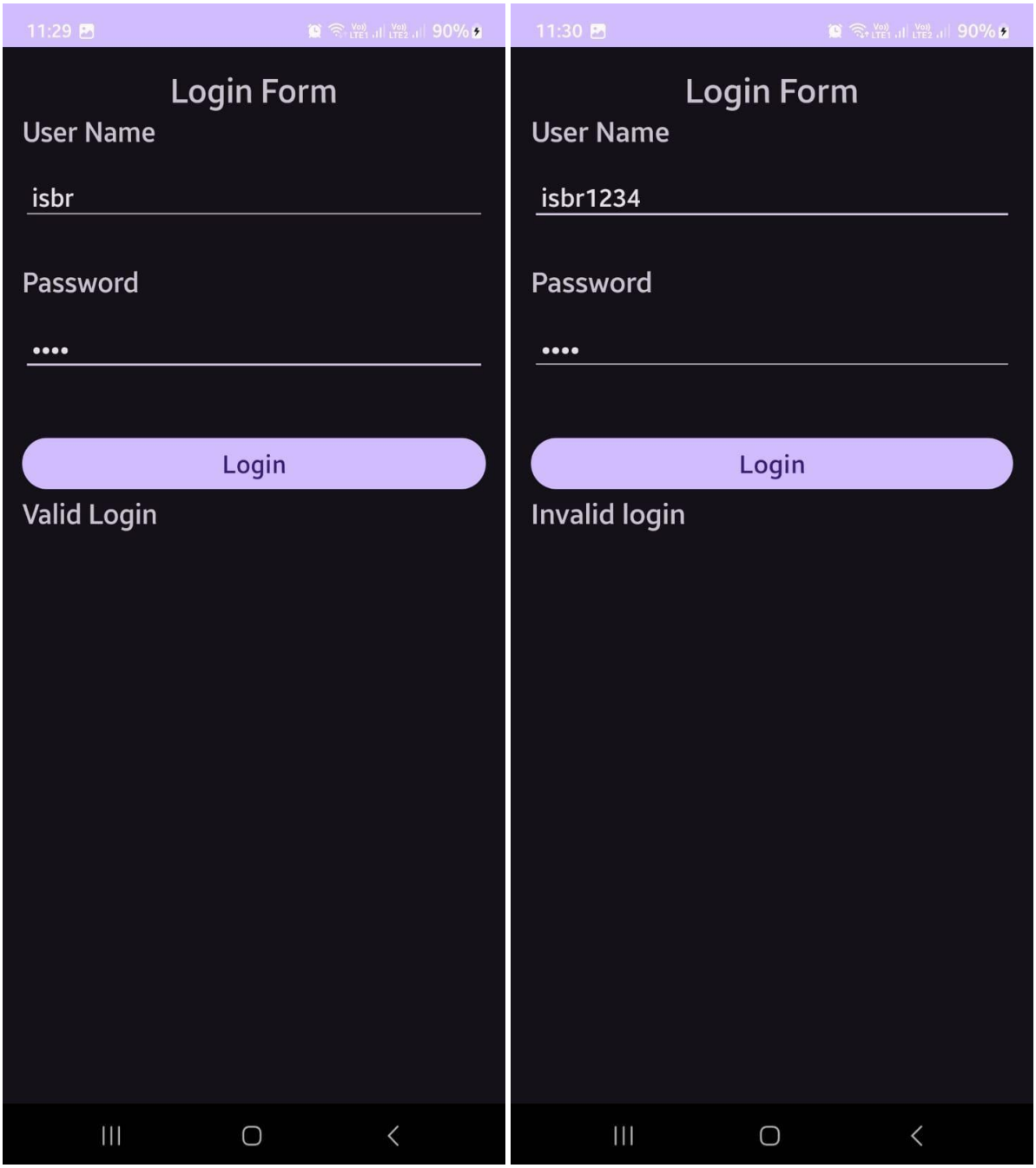
```
package com.bca.loginprgrm;
import androidx.appcompat.app.AppCompatActivity;
import android.os.Bundle;
import android.view.View;
import android.widget.Button;
import android.widget.EditText;
import android.widget.TextView;
public class MainActivity extends AppCompatActivity {
  EditText etUsername,etPassword; 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("isbr") &&
                   etPassword.getText().toString().equals("isbr"))
             {
                tvMessage.setText("Valid Login");
             }
             else
             {
                tvMessage.setText("Invalid login");
             }
          }
          });
       }
  }
```

OUTPUT


11:29
90%
Login Form
User Name
Username
Password
Password
Login
Password


11:29
90%
Login Form
User Name
isbr
Password
Login
Password



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="utf8"?>
<androidx.constraintlayout.widget.ConstraintLayout
    xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:app="http://schemas.android.com/apk/resauto"
    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"
        android:textSize="30sp"/>

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

MainActivity.java

```
package com.bca.helloworld;
import androidx.appcompat.app.AppCompatActivity;
import android.os.Bundle;
public class MainActivity extends AppCompatActivity

{
    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.