**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

1. Read/ write the Local data.
2. Create / Read / Write data with database (SQLite).
3. Create an application to send SMS and receive SMS
4. Create an application to send an e-mail.

12 Display Map based on the Current/given location.

1. 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”
2. 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.

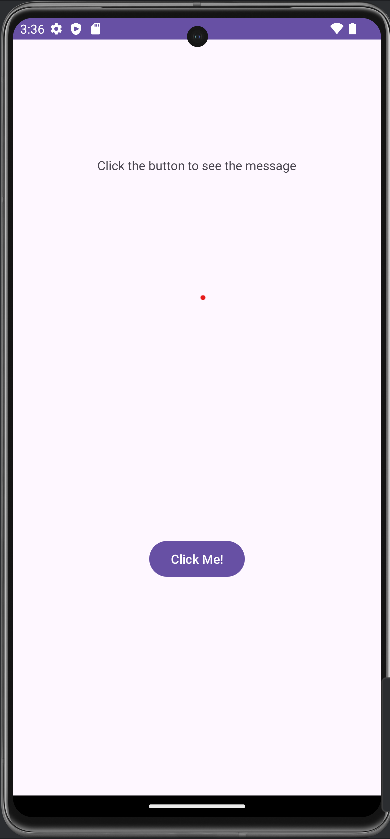
**Design:**

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 :**

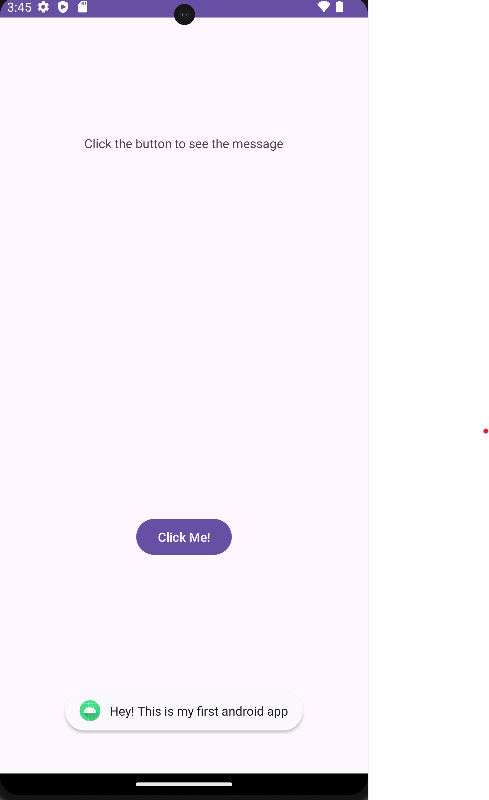
<androidx.constraintlayout.widget.ConstraintLayout xmlns:android="http://schemas.android.com/apk/res/android"  
 xmlns:app="http://schemas.android.com/apk/res-auto"  
 xmlns:tools="http://schemas.android.com/tools"  
 android:layout\_width="match\_parent"  
 android:layout\_height="match\_parent"  
 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:**

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



**Program 2**

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

Steps to start new project:

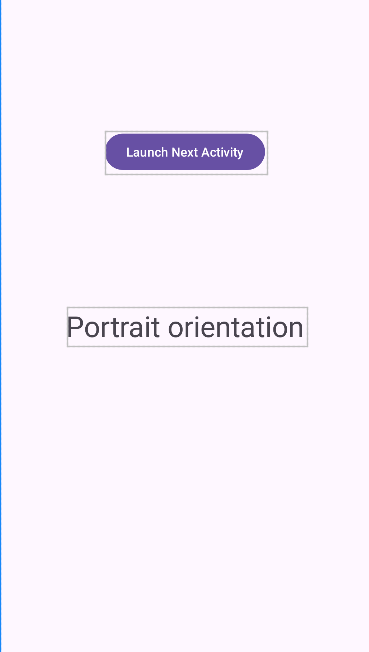
1. Click Start →Android Studio,
2. Welcome to Android Studio dialog box will appear.
3. Click New Project, the New Project Dialog box appears.
4. Choose Empty Views Activity then click Next.
5. Specify the Name of your project, Select the Language as Java, and Select Minimum SDK as “API 29(“Q”; Android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



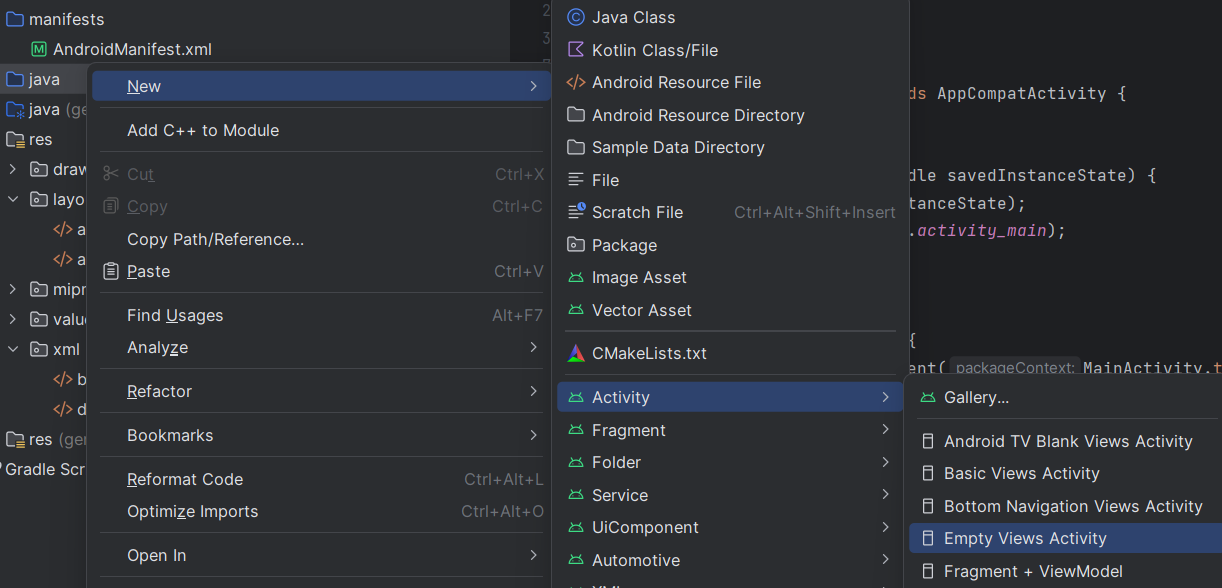
**Create another layout :**

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

1. Right Click java file on left hand side sidebar

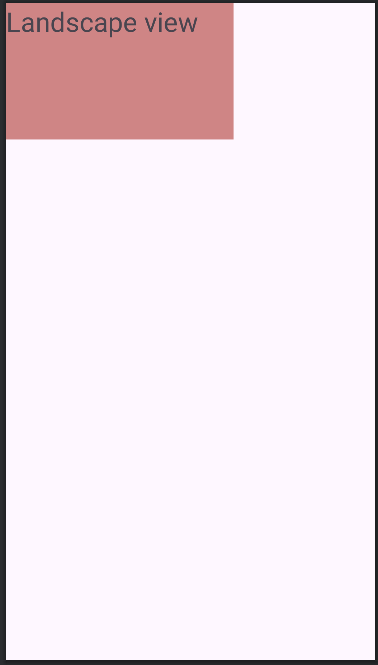
2. New > Activity > Empty views Activity

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**



**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 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

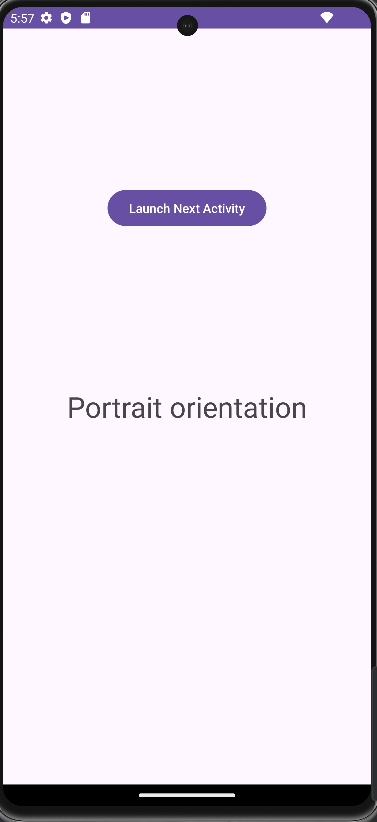
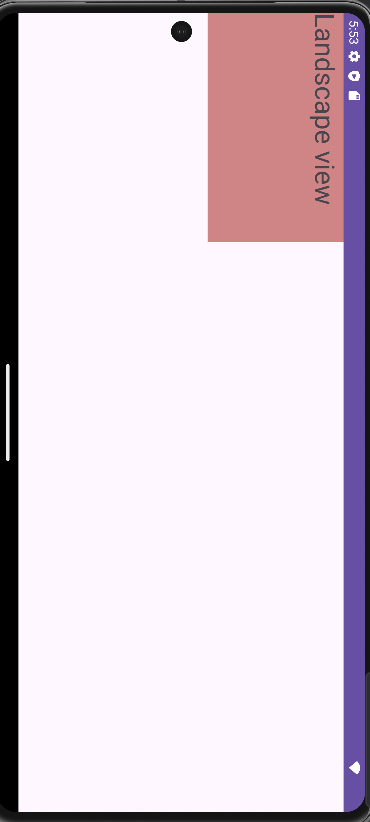
package com.example1.labprogram2\_2ndscreen;  
  
import androidx.appcompat.app.AppCompatActivity;  
***import android.content.Intent;  
import android.os.Bundle;  
import android.view.View;***  
public class MainActivity extends AppCompatActivity {  
  
 @Override  
 protected void onCreate(Bundle savedInstanceState) {  
 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

1. **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.

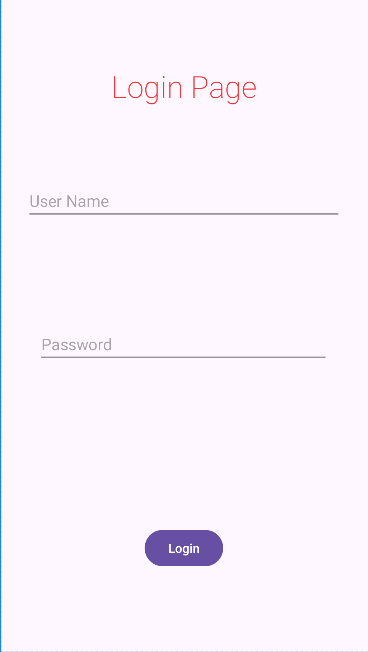
**Step 2: Design the Activity\_main.xml screen**

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**

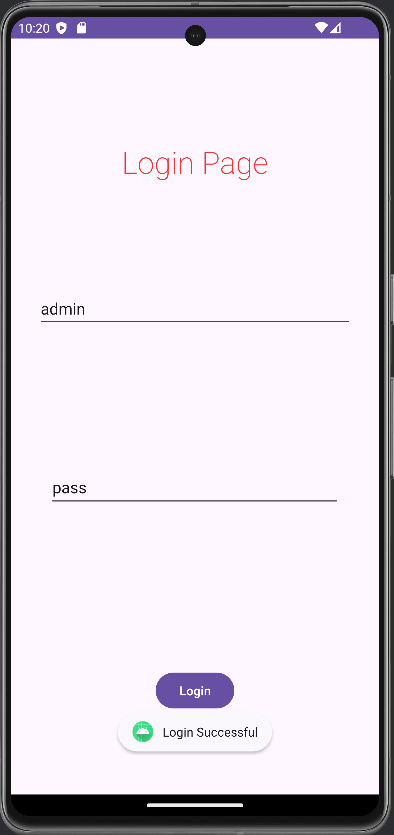
**MainActivity.java code**

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

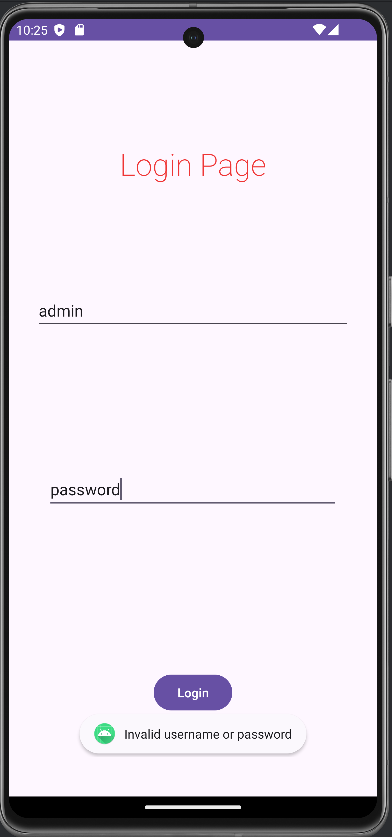
package com.example1.lab\_program3;  
  
import androidx.appcompat.app.AppCompatActivity;  
  
import android.os.Bundle;  
**import android.view.View;  
import android.widget.Button;  
import android.widget.Toast;  
import android.widget.EditText;**  
public class MainActivity extends AppCompatActivity {  
  
 **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:



Output 2:



Program 4

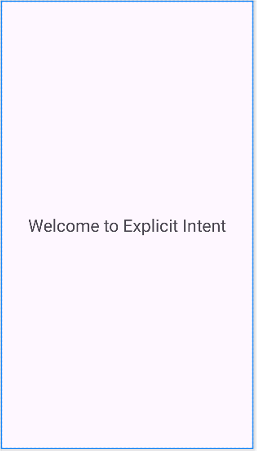
Create an application to implement new activity using explicit intent, implicit intent

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>

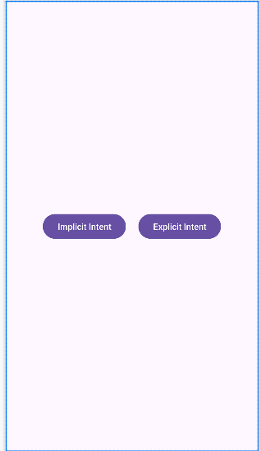
The following figure illustrates the design view of the application(activity\_new.xml).



1. Add Linear Layout in the design view
2. 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).

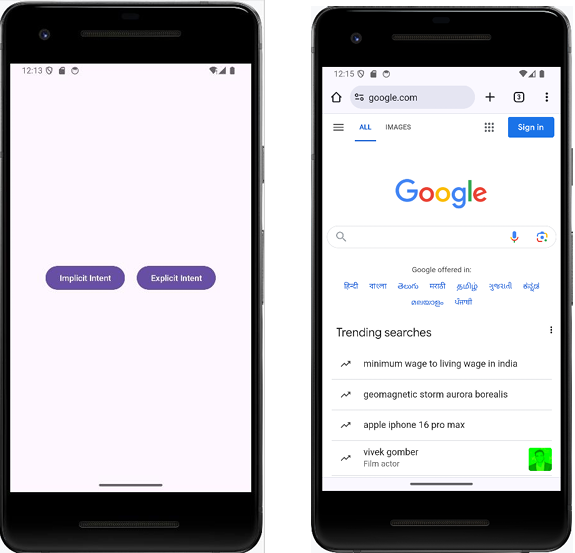


1. Add two events named as onImplicitButtonClicked, onExplicitButtonClicked and update the following code in MainActivity.java

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**

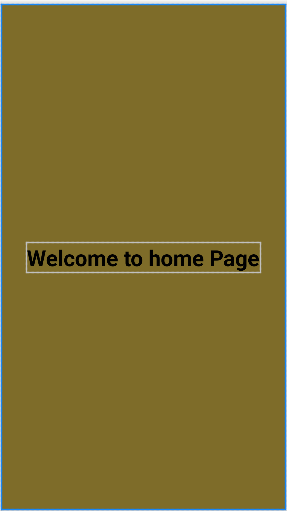
****

**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 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**>

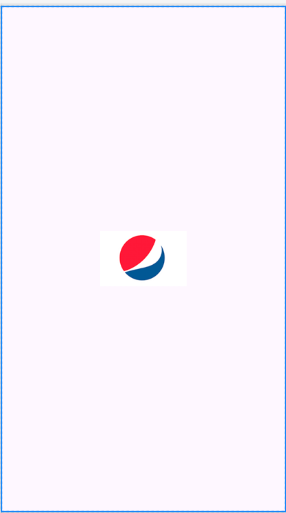
1. The following figure illustrates the design view of the application(activity\_mainScreen.xml).



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

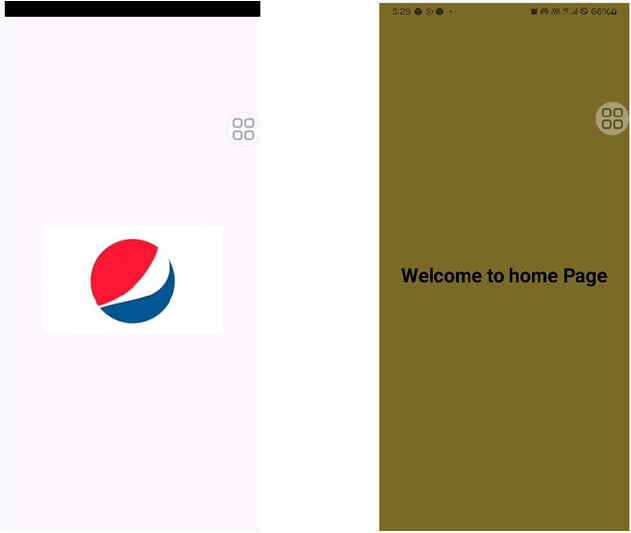
*<?***xml version="1.0" encoding="utf-8"***?>*<**RelativeLayout xmlns:android="http://schemas.android.com/apk/res/android"  
 xmlns:app="http://schemas.android.com/apk/res-auto"  
 xmlns:tools="http://schemas.android.com/tools"  
 android:id="@+id/main"  
 android:layout\_width="match\_parent"  
 android:layout\_height="match\_parent"  
 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**>

1. The following figure illustrates the design view of the application(activity\_main.xml).

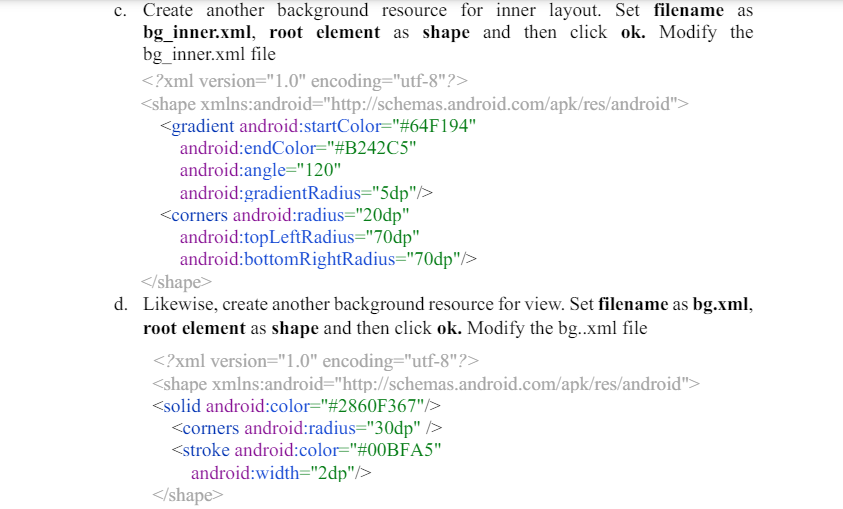


1. Update the following code in Mainactivity.java

**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*);** }  
}













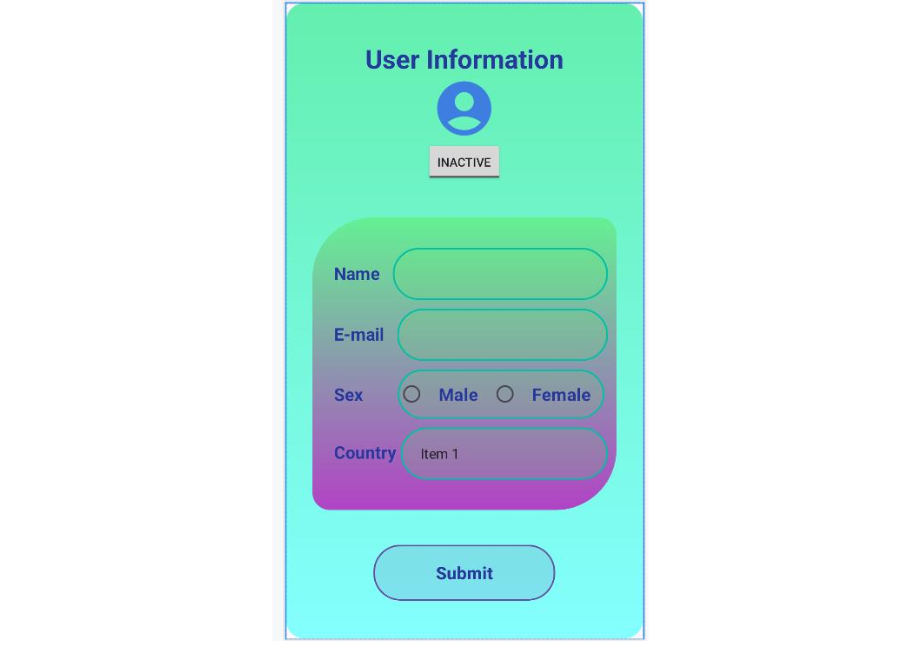


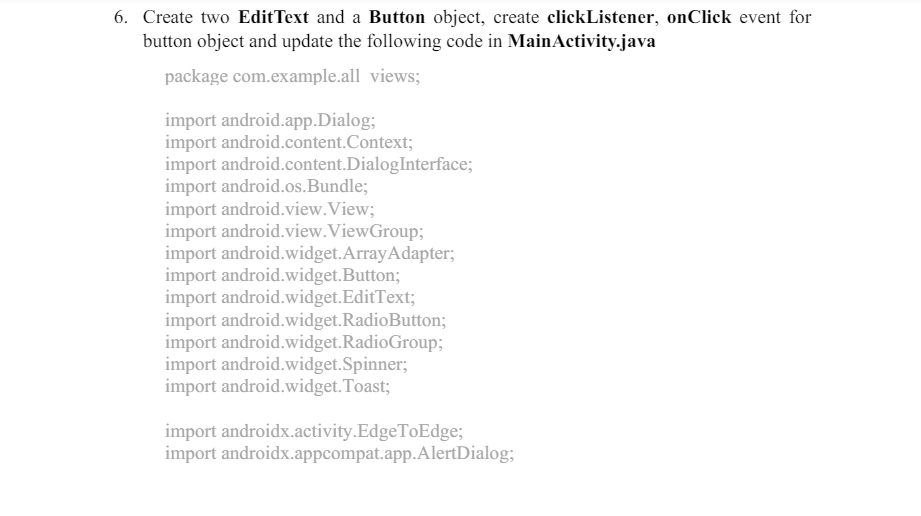


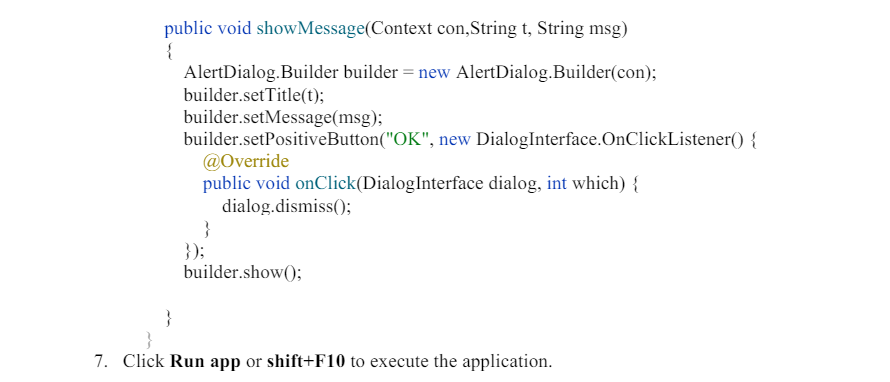
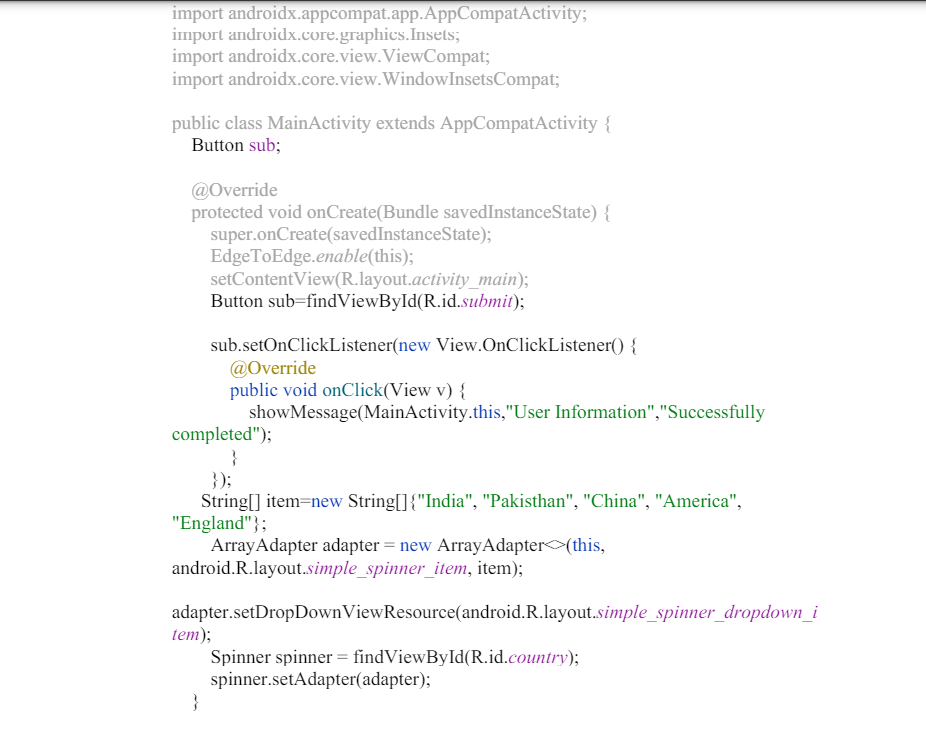




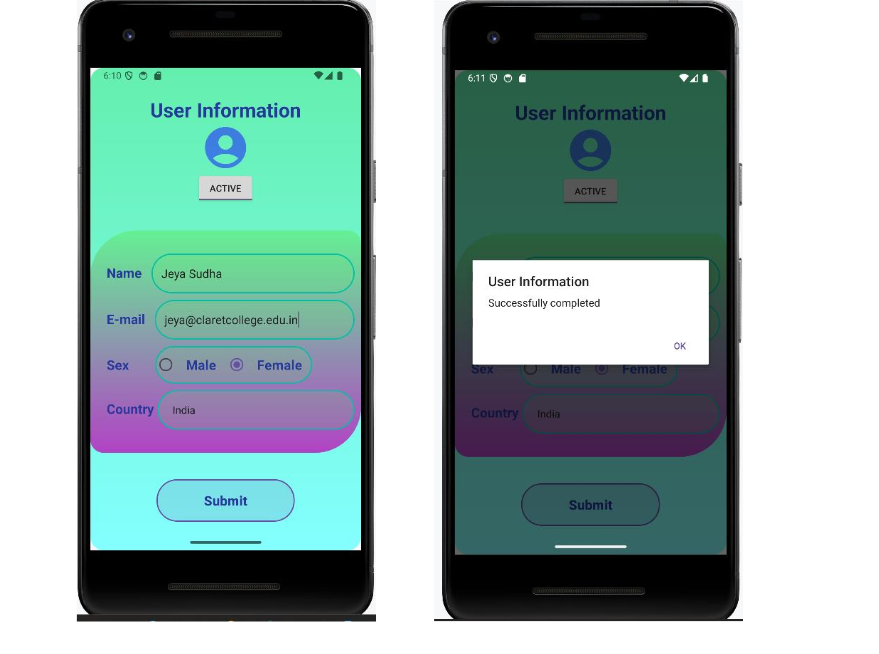
The following figure illustrates the design view of the application







**Output:**



**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**:

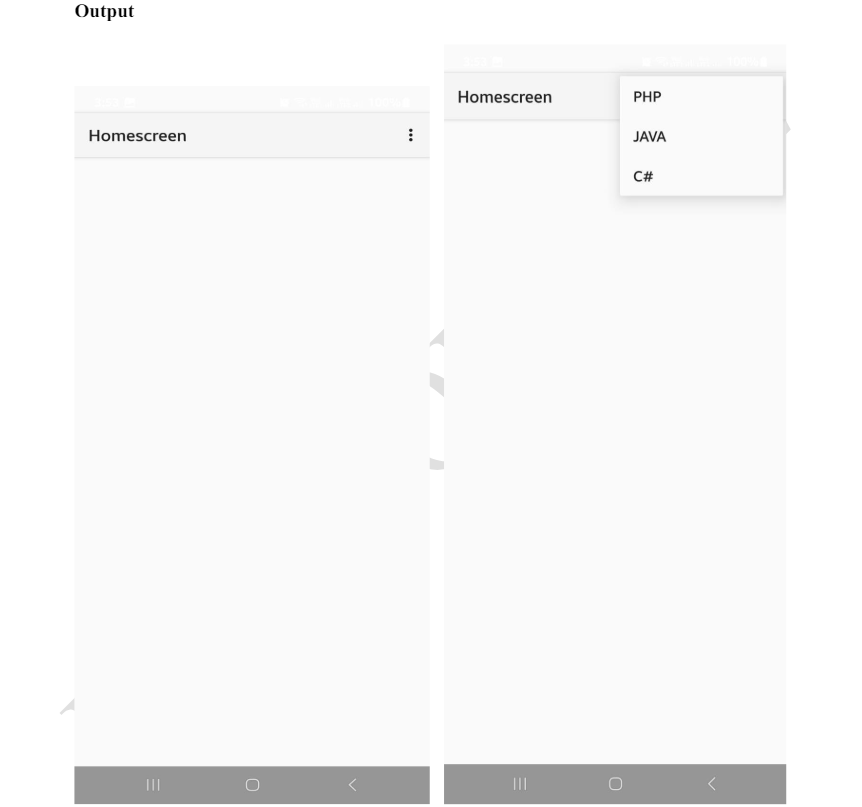
Set the Uses Permission in AndroidManifest.xml

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>





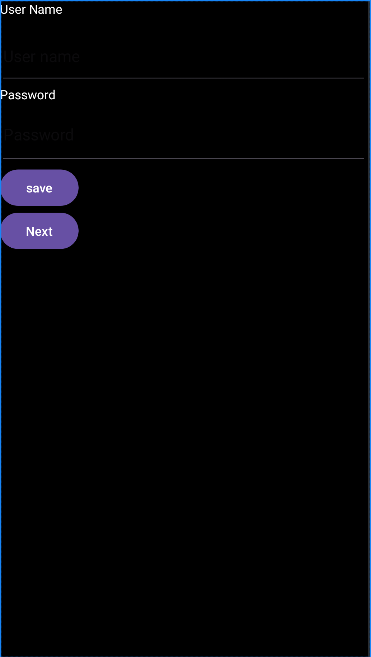
**Program 8**

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

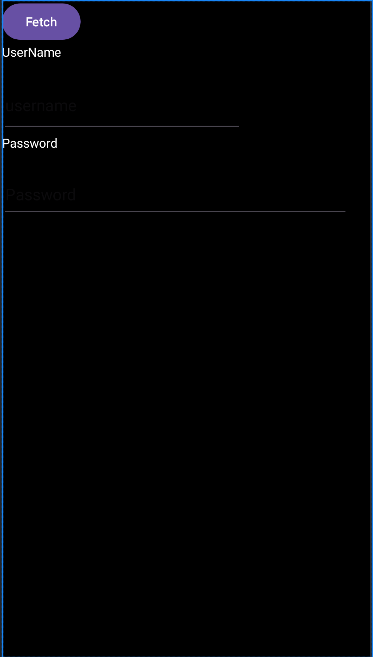
Give suitable name to the file and click finish

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>

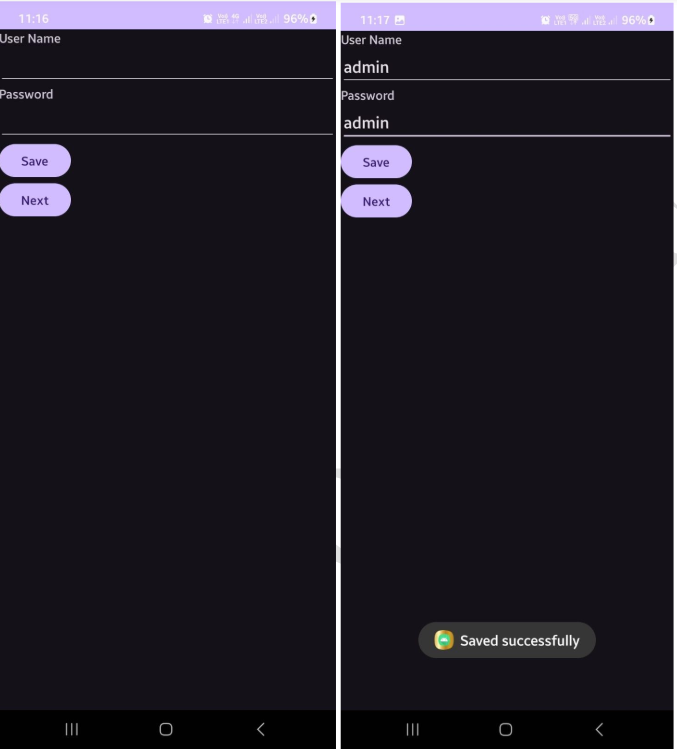
**Design:**

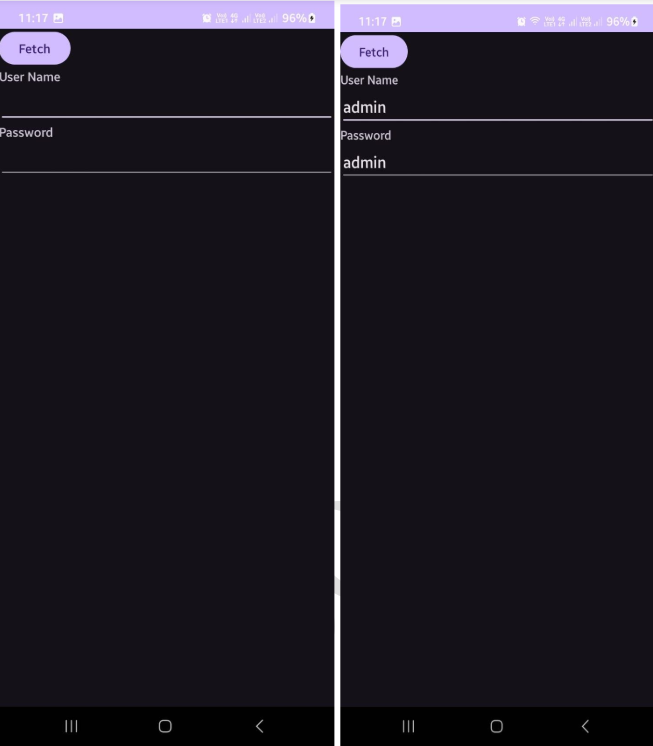


**MainActivity2 .java**

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:**





**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"  
 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  
 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,"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);  
 }  
 });  
 }  
 }

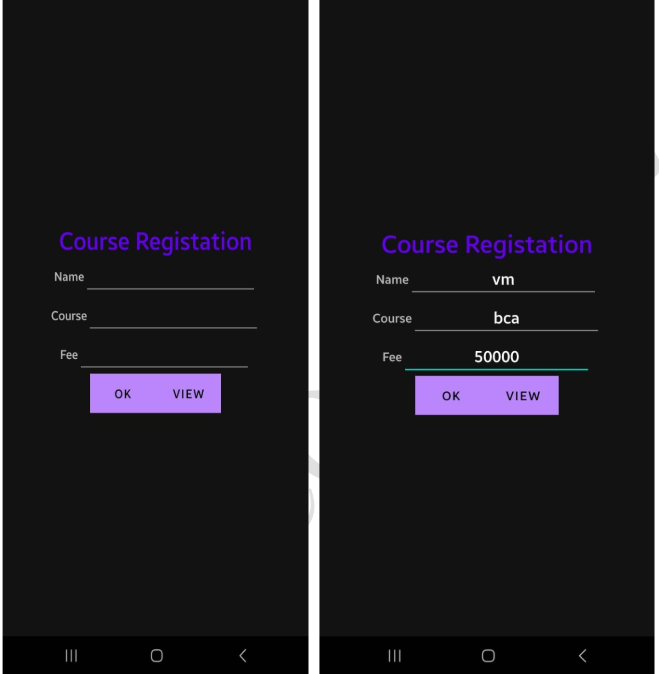
**activity\_edit.xml**

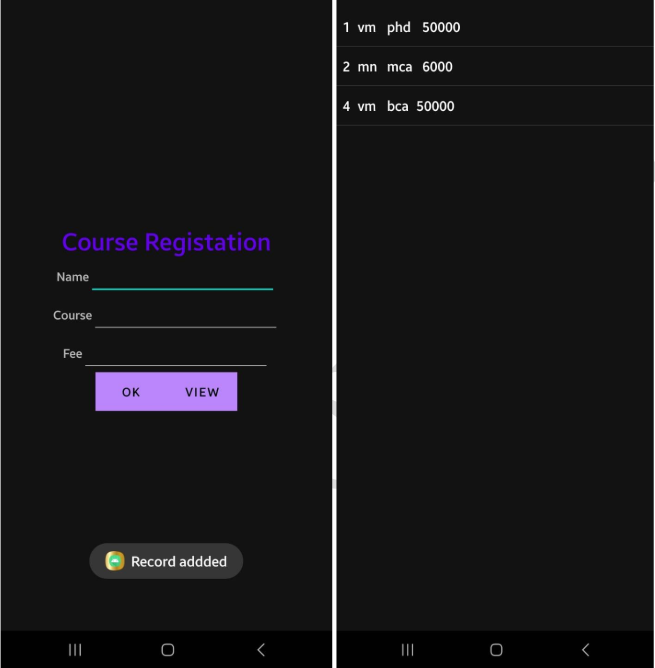
<?xml version="1.0" encoding="utf-8"?>  
<LinearLayout xmlns:android="http://schemas.android.com/apk/res/android"  
 xmlns:app="http://schemas.android.com/apk/res-auto"  
 xmlns:tools="http://schemas.android.com/tools"  
 android:layout\_width="match\_parent"  
 android:layout\_height="match\_parent"  
 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:**

****

****

**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.

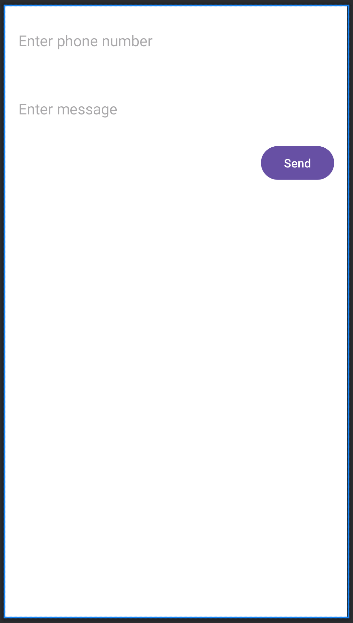
3. Specify the Name of your project, Select the Language as Java, and Select the SDK as API 24(“Nougat”,Android 7.0). Click Finish Button.

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**

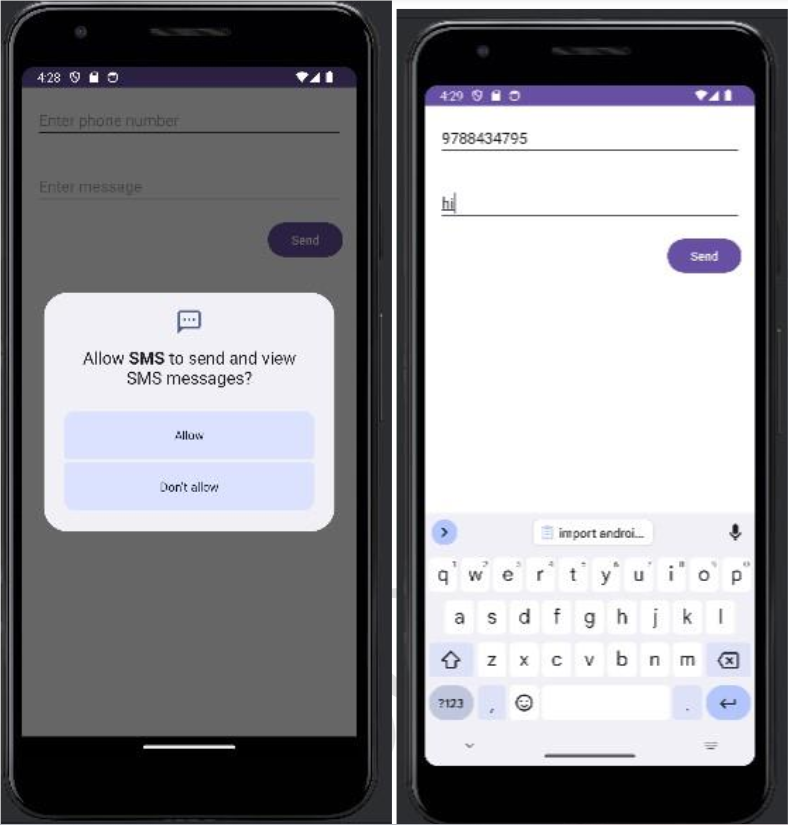
package com.example1.lab10;  
  
import androidx.appcompat.app.AppCompatActivity;  
  
import android.os.Bundle;  
import androidx.appcompat.app.AppCompatActivity;  
import androidx.core.app.ActivityCompat;  
import androidx.core.content.ContextCompat;  
import android.content.BroadcastReceiver;  
import 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 ” />**  
 <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:**

****

**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.

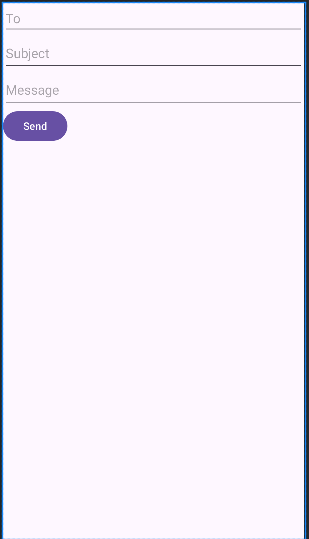
3. Specify the Name of your project, Select the Language as Java, and Select the SDK as API 24(“Nougat”,Android 7.0).Click Finish Button.

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>

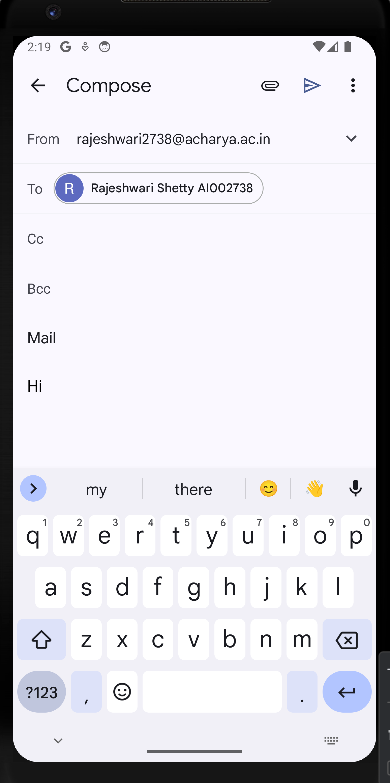
**Design:**

****

**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:**

****

**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.

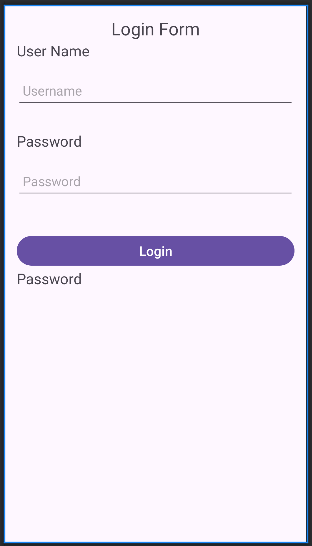
4. Update the following code in activity\_main.xml and MainActivity.java

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>

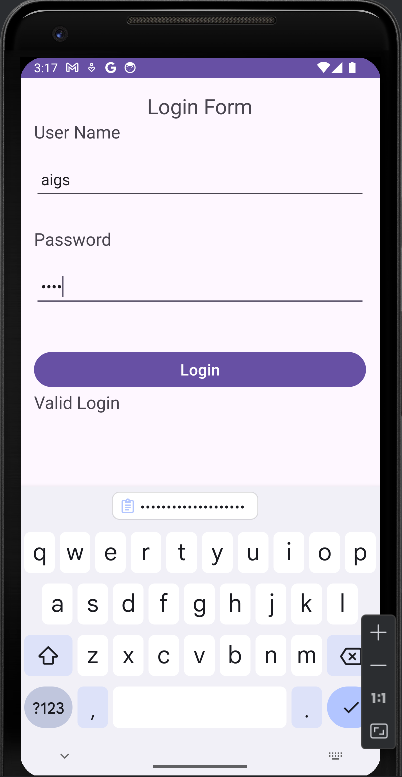
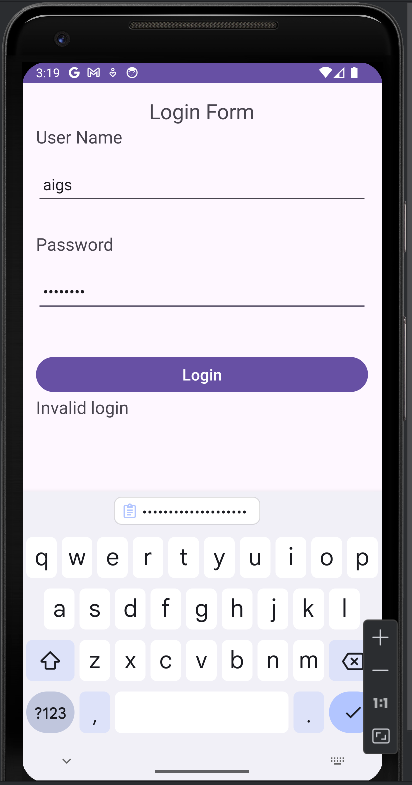
**Design:**

****

**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:**

** **

**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;  
  
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.

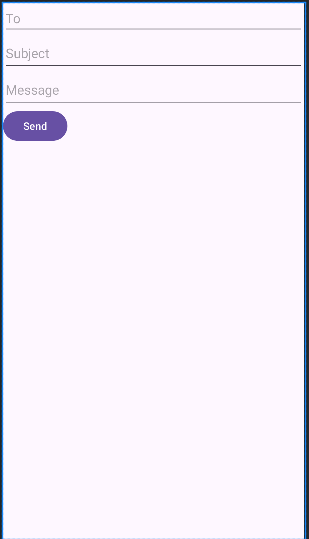
3. Specify the Name of your project, Select the Language as Java, and Select the SDK as API 24(“Nougat”,Android 7.0).Click Finish Button.

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>

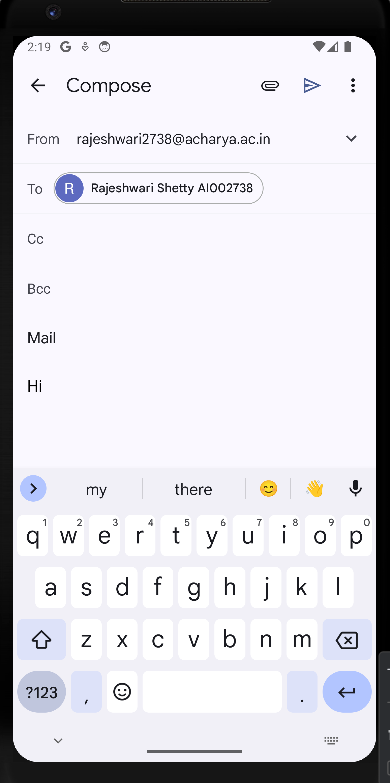
**Design:**

****

**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:**

****

**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;  
  
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.