**Mobile App Development Lab Manual**

**1.Login Application**

**Aim**

To create a android application that creates login page.

**Algorithm**

**Step 1 :** Open Android Studio and then click on **File -> New -> New project.**

**Step 2:** Then type the Application name as **“loginapplication″** and click **Next.**

**Step 3:** Then select the **Empty Activity**and click **Next.**

**Step 4:** Finally click F**inish.**

**Step 5:** Click on **app -> res -> layout -> activity\_main.xml.**

**Step 6:** Now click on **code** Then delete the code which is there and type the new code that designs the front end of your application.

**Step 7:** Click on **app -> java -> com.example.loginapplication -> MainActivity.**

**Step 8:** Then delete the code which is there and type the new code that validates the user input.

**Step 9:** Now run the application to see the output.

**Source Code**

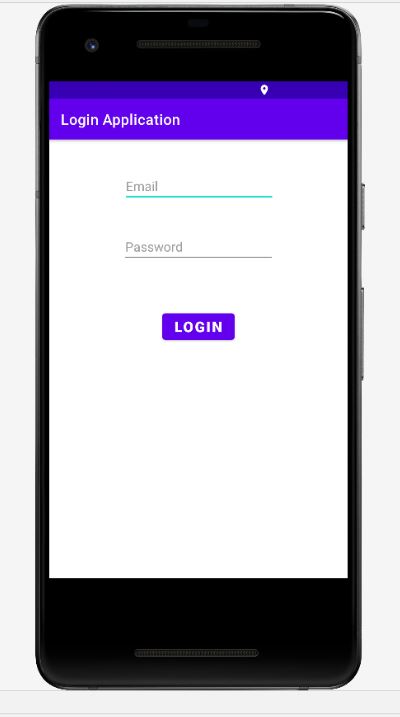
**activity\_main.xml**

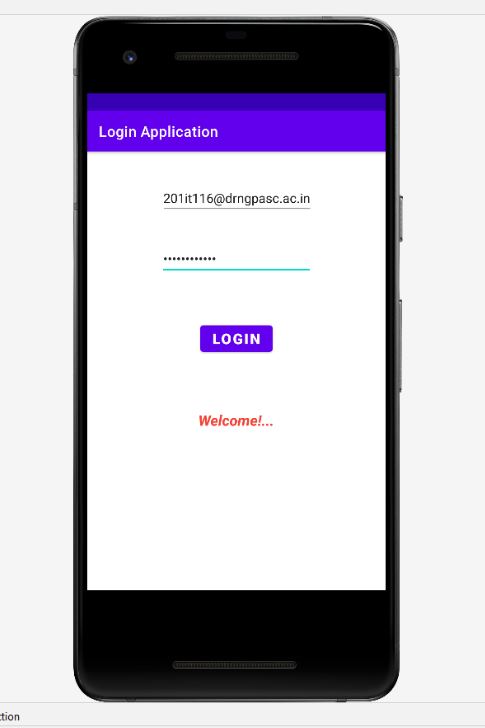
*<?*xml version="1.0" encoding="utf-8"*?>*<androidx.constraintlayout.widget.ConstraintLayout xmlns:android="http://schemas.android.com/apk/res/android"  
 xmlns:app="http://schemas.android.com/apk/res-auto"  
 xmlns:tools="http://schemas.android.com/tools"  
 android:layout\_width="match\_parent"  
 android:layout\_height="match\_parent"  
 tools:context=".MainActivity">  
  
 <EditText  
 android:id="@+id/EmailText"  
 android:layout\_width="wrap\_content"  
 android:layout\_height="wrap\_content"  
 android:layout\_marginStart="101dp"  
 android:layout\_marginTop="106dp"  
 android:layout\_marginEnd="100dp"  
 android:layout\_marginBottom="580dp"  
 android:ems="10"  
 android:hint="Email"  
 android:inputType="textEmailAddress"  
 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" />  
  
 <EditText  
 android:id="@+id/PasswordText"  
 android:layout\_width="wrap\_content"  
 android:layout\_height="wrap\_content"  
 android:layout\_marginStart="101dp"  
 android:layout\_marginTop="32dp"  
 android:layout\_marginEnd="100dp"  
 android:layout\_marginBottom="433dp"  
 android:ems="10"  
 android:hint="Password"  
 android:inputType="textPassword"  
 app:layout\_constraintBottom\_toBottomOf="parent"  
 app:layout\_constraintEnd\_toEndOf="parent"  
 app:layout\_constraintHorizontal\_bias="0.0"  
 app:layout\_constraintStart\_toStartOf="parent"  
 app:layout\_constraintTop\_toBottomOf="@+id/EmailText"  
 app:layout\_constraintVertical\_bias="1.0" />  
  
 <Button  
 android:id="@+id/LoginButton"  
 android:layout\_width="wrap\_content"  
 android:layout\_height="wrap\_content"  
 android:layout\_marginStart="158dp"  
 android:layout\_marginTop="63dp"  
 android:layout\_marginEnd="159dp"  
 android:layout\_marginBottom="322dp"  
 android:text="Login"  
 android:textSize="20sp"  
 android:textStyle="bold"  
 app:layout\_constraintBottom\_toBottomOf="parent"  
 app:layout\_constraintEnd\_toEndOf="parent"  
 app:layout\_constraintStart\_toStartOf="parent"  
 app:layout\_constraintTop\_toBottomOf="@+id/PasswordText"  
 app:layout\_constraintVertical\_bias="1.0" />  
  
 <TextView  
 android:id="@+id/NotificationText"  
 android:layout\_width="wrap\_content"  
 android:layout\_height="wrap\_content"  
 android:layout\_marginStart="176dp"  
 android:layout\_marginTop="78dp"  
 android:layout\_marginEnd="177dp"  
 android:layout\_marginBottom="224dp"  
 android:textColor="#F44336"  
 android:textSize="20sp"  
 android:textStyle="bold|italic"  
 app:layout\_constraintBottom\_toBottomOf="parent"  
 app:layout\_constraintEnd\_toEndOf="parent"  
 app:layout\_constraintStart\_toStartOf="parent"  
 app:layout\_constraintTop\_toBottomOf="@+id/LoginButton" />  
</androidx.constraintlayout.widget.ConstraintLayout>

**MainActivity.java**

package com.example.loginapplication;  
import androidx.appcompat.app.AppCompatActivity;  
  
import android.content.Intent;  
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 {  
 private TextView result;  
 private EditText email;  
 private EditText password;  
 private Button button;  
 @Override  
 protected void onCreate(Bundle savedInstanceState) {  
 super.onCreate(savedInstanceState);  
 setContentView(R.layout.*activity\_main*);  
 email=(EditText) findViewById(R.id.*EmailText*);  
 password=(EditText) findViewById(R.id.*PasswordText*);  
 button =(Button) findViewById(R.id.*LoginButton*);  
 result=(TextView) findViewById(R.id.*NotificationText*);  
 button.setOnClickListener(new View.OnClickListener() {  
 @Override  
 public void onClick(View v) {  
Validate(email.getText().toString(),password.getText().toString());  
 }  
 });  
 }  
 public void Validate(String useremail,String userpswrd)  
 {  
 if((useremail.equals("201it116@drngpasc.ac.in") ) && (userpswrd.equals("Student@1234")) )  
 {  
 result.setText("Welcome!...");  
 }  
 else  
 {  
 result.setText("Invalid Credentials!...");  
 }  
 }  
}

**Output**





**2.Menu Application**

**Aim**

To create a android application that dispalys menu.

**Algorithm**

**Step 1 :** Open Android Studio and then click on **File -> New -> New project.**

**Step 2:** Then type the Application name as **“menuapp″** and click **Next.**

**Step 3:** Then select the **Empty Activity**and click **Next.**

**Step 4:** Finally click F**inish.**

**Step 5:** Click on **app -> res -> layout -> activity\_main.xml.**

**Step 6:** Now click on **code** Then delete the code which is there and type the new code that designs the front end of your application.

**Step 7:** Click on **app -> res. Then right click on res and Click on New-> Android Resource File.**

**Step 8:** Type “menu” in filename and select resource type as menu and then click Ok.

**Step 9:** Then write the code in menu.xml file that creates menu.

**Step 7:** Click on **app -> java -> com.example.menuapp -> MainActivity.**

**Step 8:** Then delete the code which is there and type the new code

**Step 9:** Now run the application to see the output.

**Source 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"/>

**menu.xml**

*<?*xml version="1.0" encoding="utf-8"*?>*<menu xmlns:android="http://schemas.android.com/apk/res/android"  
 xmlns:app="http://schemas.android.com/apk/res-auto">  
 <item android:id="@+id/more"  
 android:title=""  
 android:icon="@drawable/ic\_baseline\_read\_more\_24"  
 app:showAsAction="always">  
 <menu>  
 <item android:id="@+id/settings"  
 android:title="Settings"  
 app:showAsAction="ifRoom">  
 <menu>  
 <item android:id="@+id/Account"  
 android:title="Account"  
 app:showAsAction="ifRoom"></item>  
 <item android:id="@+id/Help"  
 android:title="Help"  
 app:showAsAction="ifRoom"></item>  
 </menu>  
 </item>  
 <item android:id="@+id/Download"  
 android:title="Download"  
 app:showAsAction="ifRoom"></item>  
 <item android:id="@+id/Logout"  
 android:title="Logout"  
 app:showAsAction="ifRoom"></item>  
 </menu>  
 </item>  
</menu>

**MainActivity.java**

package com.example.menuapp;  
import androidx.annotation.NonNull;  
import androidx.appcompat.app.AppCompatActivity;  
import android.os.Bundle;  
import android.view.Menu;  
import android.view.MenuInflater;  
import android.view.MenuItem;  
import android.widget.Toast;  
public class MainActivity extends AppCompatActivity {

@Override  
protected void onCreate(Bundle savedInstanceState) {  
super.onCreate(savedInstanceState);  
setContentView(R.layout.*activity\_main*);  
}  
@Override  
public boolean onOptionsItemSelected(@NonNull MenuItem item) {  
switch(item.getItemId())  
{  
case R.id.*Download*:  
 Toast.*makeText*(MainActivity.this,"Downloading!...",Toast.*LENGTH\_LONG*).shoW();

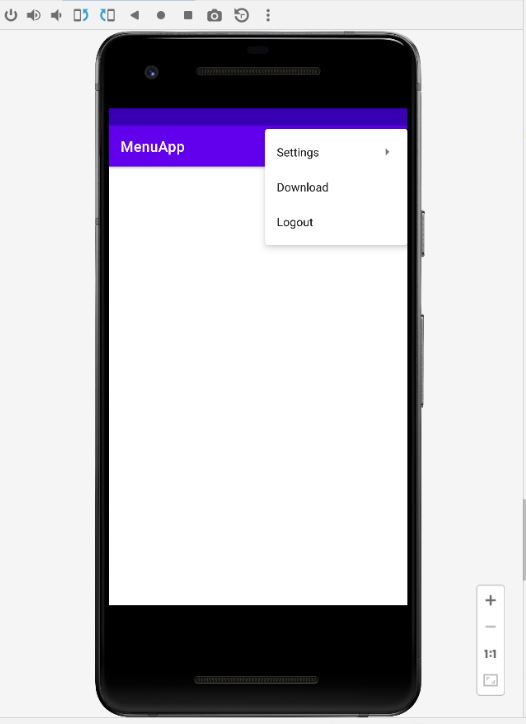
return true;

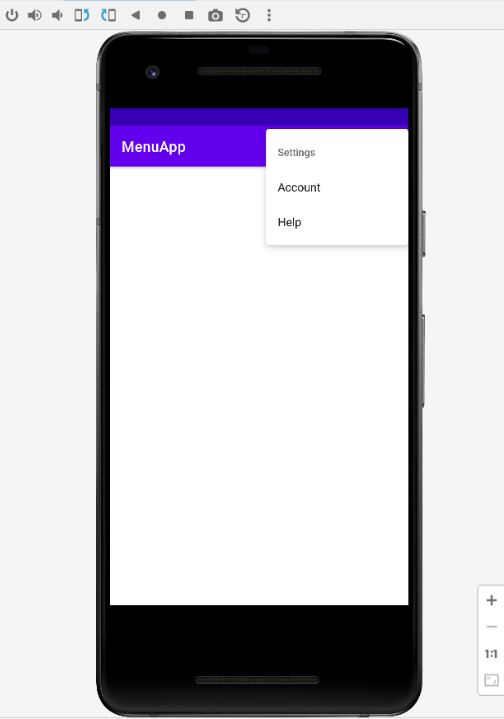
case R.id.*Logout*:

Toast.*makeText*(MainActivity.this,"LogingOut",Toast.*LENGTH\_LONG*).show();  
return true;

case R.id.*Account*:  
 Toast.*makeText*(MainActivity.this,"Dedirecting to Account Page!...",Toast.*LENGTH\_LONG*).show();  
 return true;  
 case R.id.*Help*:  
 Toast.*makeText*(MainActivity.this,"Dedirecting to Help Page!...",Toast.*LENGTH\_LONG*).show();  
 return true;  
 default:  
 return super.onOptionsItemSelected(item);  
 }  
 }  
 @Override  
 public boolean onCreatePanelMenu(int featureId, @NonNull Menu menu) {  
 MenuInflater inflater=getMenuInflater();  
 inflater.inflate(R.menu.*menu*,menu);  
 return true;  
 }  
}

**Output**

****

****

**3.Font and Colors**

**Aim**

To create a android application that uses font and colors.

**Algorithm**

**Step 1 :** Open Android Studio and then click on **File -> New -> New project.**

**Step 2:** Then type the Application name as **“font\_color″** and click **Next.**

**Step 3:** Then select the **Empty Activity**and click **Next.**

**Step 4:** Finally click F**inish.**

**Step 5:** Click on **app -> res -> layout -> activity\_main.xml.**

**Step 6:** Now click on **code** Then delete the code which is there and type the new code that designs the front end of your application.

**Step 7:** Click on **app -> java -> com.example.font\_color -> MainActivity.**

**Step 8:** Then delete the code which is there and type the new code that uses font and color.

**Step 9:** Now run the application to see the output.

**Source Code**

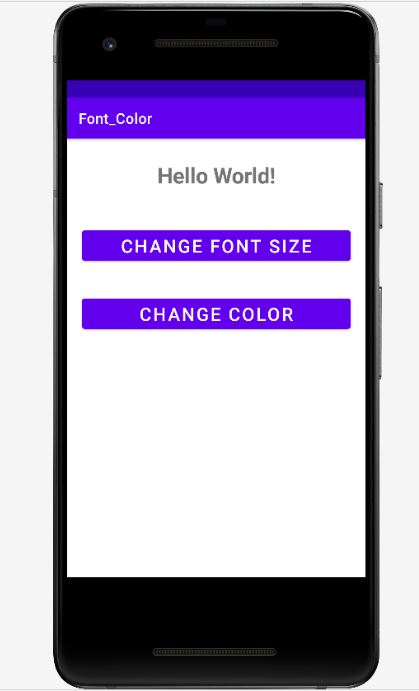
**activity\_main.xml**

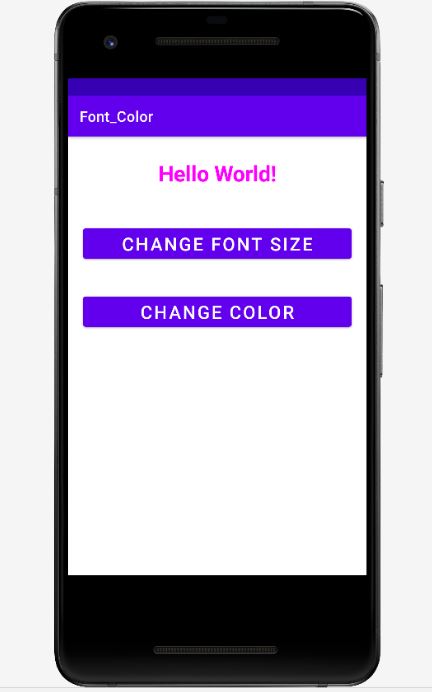
*<?*xml version="1.0" encoding="utf-8"*?>*<LinearLayout xmlns:android="http://schemas.android.com/apk/res/android"  
 xmlns:app="http://schemas.android.com/apk/res-auto"  
 xmlns:tools="http://schemas.android.com/tools"  
 android:layout\_width="match\_parent"  
 android:layout\_height="match\_parent"  
 tools:context=".MainActivity"  
 android:orientation="vertical">  
  
 <TextView  
 android:id="@+id/textView"  
 android:layout\_width="match\_parent"  
 android:layout\_height="wrap\_content"  
 android:layout\_margin="30dp"  
 android:gravity="center"  
 android:text="Hello World!"  
 android:textSize="25sp"  
 android:textStyle="bold" />  
  
  
 <Button  
 android:id="@+id/fontbutton"  
 android:layout\_width="match\_parent"  
 android:layout\_height="wrap\_content"  
 android:layout\_margin="20dp"  
 android:gravity="center"  
 android:text="Change font size"  
 android:textSize="25sp" />  
  
 <Button  
 android:id="@+id/colorbutton"  
 android:layout\_width="match\_parent"  
 android:layout\_height="wrap\_content"  
 android:layout\_margin="20dp"  
 android:gravity="center"  
 android:text="Change color"  
 android:textSize="25sp" />  
  
</LinearLayout>

**MainActivity.java**

package com.example.font\_color;  
import androidx.appcompat.app.AppCompatActivity;  
  
import android.graphics.Color;  
import android.os.Bundle;  
import android.view.View;  
import android.widget.Button;  
import android.widget.TextView;  
public class MainActivity extends AppCompatActivity {  
 int color=1;  
 int fontsize=30;  
 @Override  
 protected void onCreate(Bundle savedInstanceState) {  
 super.onCreate(savedInstanceState);  
 setContentView(R.layout.*activity\_main*);  
 TextView t=(TextView)findViewById(R.id.*textView*);  
 Button changefontsize=(Button) findViewById(R.id.*fontbutton*);  
 changefontsize.setOnClickListener(new View.OnClickListener() {  
 @Override  
 public void onClick(View v) {  
 t.setTextSize(fontsize);  
 fontsize+=5;  
 if(fontsize==50)  
 {  
 fontsize=30;  
 }  
 }  
 });  
 Button changecolor=(Button) findViewById(R.id.*colorbutton*);  
 changecolor.setOnClickListener(new View.OnClickListener() {  
 @Override  
 public void onClick(View v) {  
 switch(color)  
 {  
 case 1:  
 t.setTextColor(Color.*RED*);  
 break;  
 case 2:  
 t.setTextColor(Color.*BLUE*);  
 break;  
 case 3:  
 t.setTextColor(Color.*GREEN*);  
 break;  
 case 4:  
 t.setTextColor(Color.*YELLOW*);  
 break;  
 case 5:  
 t.setTextColor(Color.*CYAN*);  
 break;  
 case 6:  
 t.setTextColor(Color.*MAGENTA*);  
 break;  
 }  
 color++;  
 if(color==7)  
 {  
 color=1;  
 }  
 }  
 });  
 }  
}

**Output**





**5.Layout Managers and Event Listeners**

**Aim**

To develop a simple android application that uses Layout managers and Eventlisteners.

**Algorithm**

**Step 1 :** Open Android Studio and then click on **File -> New -> New project.**

**Step 2:** Then type the Application name as **“layoutmanager″** and click **Next.**

**Step 3:** Then select the **Empty Activity**and click **Next.**

**Step 4:** Finally click F**inish.**

**Step 5:** Click on **app -> res -> layout -> activity\_main.xml.**

**Step 6:** Now click on **code** Then delete the code which is there and type the new code

**Step 7:** Click on **app -> java -> com.example.layoutmanager -> MainActivity.**

**Step 8:** Then delete the code which is there and type the new MainActivtiy code

**Step 9:** Click on **app -> res -> layout -> activity\_main2.xml.**

**Step 10:** Now click on **Text** then delete the code which is there and type the new code that designs the front end of your application

**Step 11:** Click on **app -> java -> com.example.layoutmanager -> MainActivity2.**

**Step 12:** Then delete the code which is there and type the new code that uses layout managers and event listeners

**Step 9:** Now run the application to see the output.

**Source Code**

**activity\_main.xml**

*<?*xml version="1.0" encoding="utf-8"*?>*<RelativeLayout xmlns:android="http://schemas.android.com/apk/res/android"  
 xmlns:tools="http://schemas.android.com/tools"  
 android:layout\_width="match\_parent"  
 android:layout\_height="match\_parent"  
 tools:context=".MainActivity">  
  
 <LinearLayout  
 android:layout\_width="match\_parent"  
 android:layout\_height="100dp">  
  
 <TextView  
 android:id="@+id/textView"  
 android:layout\_width="match\_parent"  
 android:layout\_height="wrap\_content"  
 android:layout\_margin="30dp"  
 android:gravity="center"  
 android:text="Student Login Form"  
 android:textSize="25sp" />  
 </LinearLayout>  
  
 <GridLayout  
 android:id="@+id/gridLayout"  
 android:layout\_width="match\_parent"  
 android:layout\_height="match\_parent"  
 android:layout\_marginTop="100dp"  
 android:layout\_marginBottom="200dp"  
 android:columnCount="2"  
 android:rowCount="3">  
  
 <TextView  
 android:id="@+id/textView1"  
 android:layout\_width="wrap\_content"  
 android:layout\_height="wrap\_content"  
 android:layout\_row="0"  
 android:layout\_column="0"  
 android:layout\_margin="10dp"  
 android:gravity="center"  
 android:text="Name"  
 android:textSize="20sp" />  
  
 <EditText  
 android:id="@+id/editText"  
 android:layout\_width="wrap\_content"  
 android:layout\_height="wrap\_content"  
 android:layout\_row="0"  
 android:layout\_column="1"  
 android:layout\_margin="10dp"  
 android:ems="10" />  
  
 <TextView  
 android:id="@+id/textView2"  
 android:layout\_width="wrap\_content"  
 android:layout\_height="wrap\_content"  
 android:layout\_row="1"  
 android:layout\_column="0"  
 android:layout\_margin="10dp"  
 android:gravity="center"  
 android:text="Reg.No"  
 android:textSize="20sp" />  
  
 <EditText  
 android:id="@+id/editText2"  
 android:layout\_width="wrap\_content"  
 android:layout\_height="wrap\_content"  
 android:layout\_row="1"  
 android:layout\_column="1"  
 android:layout\_margin="10dp"  
 android:ems="10"  
 android:inputType="text" />  
  
 <TextView  
 android:id="@+id/textView3"  
 android:layout\_width="wrap\_content"  
 android:layout\_height="wrap\_content"  
 android:layout\_row="2"  
 android:layout\_column="0"  
 android:layout\_margin="10dp"  
 android:gravity="center"  
 android:text="Dept"  
 android:textSize="20sp" />  
  
 <Spinner  
 android:id="@+id/spinner"  
 android:layout\_width="wrap\_content"  
 android:layout\_height="wrap\_content"  
 android:layout\_row="2"  
 android:layout\_column="1"  
 android:layout\_margin="10dp"  
 android:spinnerMode="dropdown" />  
  
 </GridLayout>  
  
 <Button  
 android:id="@+id/button"  
 android:layout\_width="wrap\_content"  
 android:layout\_height="wrap\_content"  
 android:layout\_alignParentBottom="true"  
 android:layout\_centerInParent="true"  
 android:layout\_marginBottom="150dp"  
 android:text="Submit"/>  
  
</RelativeLayout>

**MainActivity.java**

package com.example.layoutmanager;  
  
import android.content.Intent;  
import android.os.Bundle;  
import android.view.View;  
import android.widget.ArrayAdapter;  
import android.widget.Button;  
import android.widget.EditText;  
import android.widget.Spinner;  
  
import androidx.appcompat.app.AppCompatActivity;  
  
public class MainActivity extends AppCompatActivity {  
  
 EditText e1,e2;  
 Button bt;  
 Spinner s;  
 String [] dept\_array={"IT","CT","CS","CSDA","CA"};  
  
 String name,reg,dept;  
  
 @Override  
 protected void onCreate(Bundle savedInstanceState) {  
 super.onCreate(savedInstanceState);  
 setContentView(R.layout.*activity\_main*);  
 e1= (EditText) findViewById(R.id.*editText*);  
 e2= (EditText) findViewById(R.id.*editText2*);  
  
 bt= (Button) findViewById(R.id.*button*);  
  
 s= (Spinner) findViewById(R.id.*spinner*);  
 ArrayAdapter adapter= new ArrayAdapter(MainActivity.this,android.R.layout.*simple\_spinner\_item*,dept\_array);  
 s.setAdapter(adapter);  
  
 bt.setOnClickListener(new View.OnClickListener() {  
 @Override  
 public void onClick(View v) {  
  
 name=e1.getText().toString();  
 reg=e2.getText().toString();  
 dept=s.getSelectedItem().toString();  
  
 Intent i = new Intent(MainActivity.this,MainActivity2.class);  
  
 i.putExtra("name\_key", name);  
 i.putExtra("reg\_key",reg);  
 i.putExtra("dept\_key", dept);  
  
 startActivity(i);  
  
 }  
 });  
 }  
}

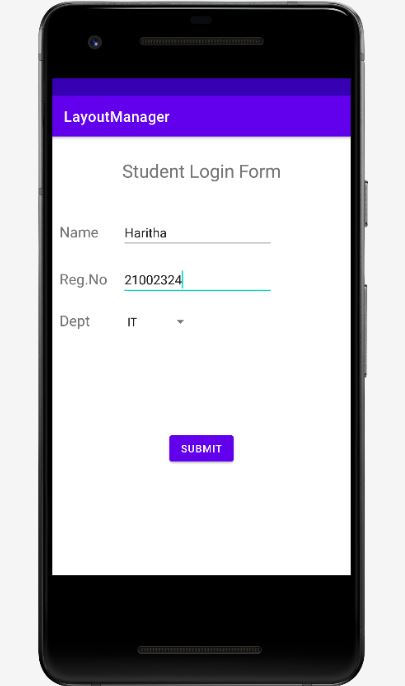
**activity\_main2.xml**

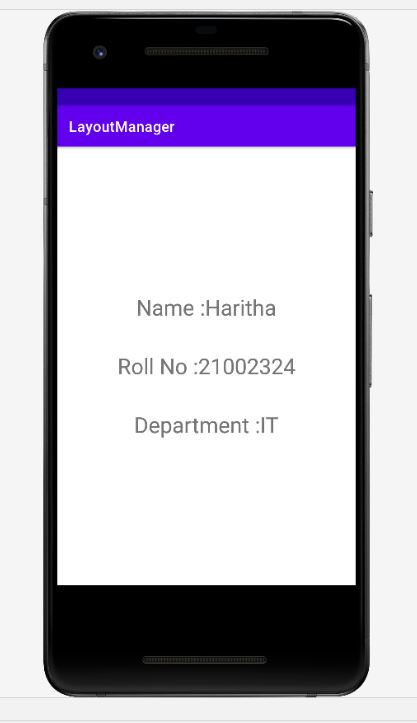
*<?*xml version="1.0" encoding="utf-8"*?>*<LinearLayout xmlns:android="http://schemas.android.com/apk/res/android"  
 xmlns:tools="http://schemas.android.com/tools"  
 android:layout\_width="match\_parent"  
 android:layout\_height="match\_parent"  
 tools:context="com.example.layoutmanager.MainActivity2"  
 android:orientation="vertical"  
 android:gravity="center">  
  
 <TextView  
 android:id="@+id/textView1"  
 android:layout\_width="wrap\_content"  
 android:layout\_height="wrap\_content"  
 android:layout\_margin="20dp"  
 android:text="New Text"  
 android:textSize="30sp"/>  
  
 <TextView  
 android:id="@+id/textView2"  
 android:layout\_width="wrap\_content"  
 android:layout\_height="wrap\_content"  
 android:layout\_margin="20dp"  
 android:text="New Text"  
 android:textSize="30sp"/>  
  
 <TextView  
 android:id="@+id/textView3"  
 android:layout\_width="wrap\_content"  
 android:layout\_height="wrap\_content"  
 android:layout\_margin="20dp"  
 android:text="New Text"  
 android:textSize="30sp"/>  
  
</LinearLayout>

**MainActivity2.java**

package com.example.layoutmanager;  
import androidx.appcompat.app.AppCompatActivity;  
  
import android.content.Intent;  
import android.os.Bundle;  
import android.widget.TextView;  
public class MainActivity2 extends AppCompatActivity {  
 TextView t1,t2,t3;  
 String name,reg,dept;  
 @Override  
 protected void onCreate(Bundle savedInstanceState) {  
 super.onCreate(savedInstanceState);  
 setContentView(R.layout.*activity\_main2*);  
  
 t1= (TextView) findViewById(R.id.*textView1*);  
 t2= (TextView) findViewById(R.id.*textView2*);  
 t3= (TextView) findViewById(R.id.*textView3*);  
  
 *//Getting the Intent* Intent i = getIntent();  
  
 *//Getting the Values from First Activity using the Intent received* name=i.getStringExtra("name\_key");  
 reg=i.getStringExtra("reg\_key");  
 dept=i.getStringExtra("dept\_key");  
  
 *//Setting the Values to Intent* t1.setText("Name :"+name);  
 t2.setText("Roll No :"+reg);  
 t3.setText("Department :"+dept);  
  
 }  
}

**Output**





**6. GPS Location**

**Aim**

To develop a simple android application that displays the current location.

**Algorithm**

**Step 1 :** Open Android Studio and then click on **File -> New -> New project.**

**Step 2:** Then type the Application name as **“GPSapplication”** and click **Next.**

**Step 3:** Then select the **Empty Activity**and click **Next.**

**Step 4:** Finally click F**inish.**

**Step 5:** Click on **app -> res -> layout -> activity\_main.xml.**

**Step 6:** Now click on **code** Then delete the code which is there and type the new code

**Step 7:** Click on **app -> java -> com.example.gpsapplication -> MainActivity.**

**Step 8:** Then delete the code which is there and type the new MainActivtiy code

**Step 9:** Click on **app -> manifest ->AndroidManifest.xml.**

**Step 10:** Then delete the code which is there and type the new code. Now run the application to see the output.

**Source Code**

**activity\_main.xml**

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

<LinearLayout xmlns:android="http://schemas.android.com/apk/res/android"

xmlns:tools="http://schemas.android.com/tools"

android:layout\_width="match\_parent"

android:layout\_height="match\_parent"

android:orientation="vertical"

android:gravity="center"

tools:context=".MainActivity">

<TextView

android:id="@+id/showLocation"

android:layout\_width="wrap\_content"

android:layout\_height="wrap\_content"

android:hint="Location"

android:textSize="24sp" />

<Button

android:id="@+id/btnGetLocation"

android:layout\_width="match\_parent"

android:layout\_height="wrap\_content"

android:text="Get Location" />

</LinearLayout>

**MainActivity.java**

import android.Manifest;

import android.app.AlertDialog;

import android.content.Context;

import android.content.DialogInterface;

import android.content.Intent;

import android.content.pm.PackageManager;

import android.location.Location;

import android.location.LocationManager;

import android.provider.Settings;

import android.support.v4.app.ActivityCompat;

import android.support.v7.app.AppCompatActivity;

import android.os.Bundle;

import android.view.View;

import android.widget.Button;

import android.widget.TextView;

import android.widget.Toast;

public class MainActivity extends AppCompatActivity {

private static final int REQUEST\_LOCATION = 1;

Button btnGetLocation;

TextView showLocation;

LocationManager locationManager;

String latitude, longitude;

@Override

protected void onCreate(Bundle savedInstanceState) {

super.onCreate(savedInstanceState);

setContentView(R.layout.activity\_main);

ActivityCompat.requestPermissions( this,

new String[] {Manifest.permission.ACCESS\_FINE\_LOCATION}, REQUEST\_LOCATION);

showLocation = findViewById(R.id.showLocation);

btnGetLocation = findViewById(R.id.btnGetLocation);

btnGetLocation.setOnClickListener(new View.OnClickListener() {

@Override

public void onClick(View v) {

location nManager = (LocationManager) getSystemService(Context.LOCATION\_SERVICE);

if (!locationManager.isProviderEnabled(LocationManager.GPS\_PROVIDER)) {

OnGPS();

} else {

getLocation();

}

}

});

}

private void OnGPS() {

final AlertDialog.Builder builder = new AlertDialog.Builder(this);

builder.setMessage("Enable GPS").setCancelable(false).setPositiveButton("Yes", new DialogInterface.OnClickListener() {

@Override

public void onClick(DialogInterface dialog, int which) {

startActivity(new Intent(Settings.ACTION\_LOCATION\_SOURCE\_SETTINGS));

}

}).setNegativeButton("No", new DialogInterface.OnClickListener() {

@Override

public void onClick(DialogInterface dialog, int which) {

dialog.cancel();

}

});

final AlertDialog alertDialog = builder.create();

alertDialog.show();

}

private void getLocation() {

if (ActivityCompat.checkSelfPermission(

MainActivity.this,Manifest.permission.ACCESS\_FINE\_LOCATION) != PackageManager.PERMISSION\_GRANTED && ActivityCompat.checkSelfPermission(

MainActivity.this, Manifest.permission.ACCESS\_COARSE\_LOCATION) != PackageManager.PERMISSION\_GRANTED) {

ActivityCompat.requestPermissions(this, new String[]{Manifest.permission.ACCESS\_FINE\_LOCATION}, REQUEST\_LOCATION);

} else {

Location locationGPS = locationManager.getLastKnownLocation(LocationManager.GPS\_PROVIDER);

if (locationGPS != null) {

double lat = locationGPS.getLatitude();

double longi = locationGPS.getLongitude();

latitude = String.valueOf(lat);

longitude = String.valueOf(longi);

showLocation.setText("Your Location: " + "\n" + "Latitude: " + latitude + "\n" + "Longitude: " + longitude);

} else {

Toast.makeText(this, "Unable to find location.", Toast.LENGTH\_SHORT).show();

}

}

}

}

**AndroidManifest.xml**

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

<manifest xmlns:android="http://schemas.android.com/apk/res/android"

package="app.com.sample">

<uses-permission android:name="android.permission.ACCESS\_FINE\_LOCATION"/>

<uses-permission android:name="android.permission.ACCESS\_COARSE\_LOCATION"/>

<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">

<activity android:name=".MainActivity">

<intent-filter>

<action android:name="android.intent.action.MAIN" />

<category android:name="android.intent.category.LAUNCHER" />

</intent-filter>

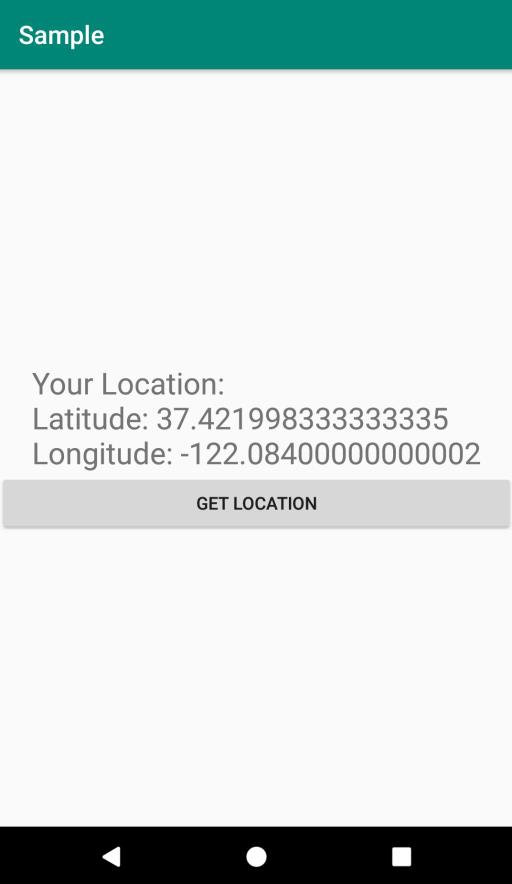
</activity>

</application>

</manifest>

**Output**





**7.Alert Message**

**Aim**

To develop a simple android application that displays alert message.

**Algorithm**

**Step 1 :** Open Android Studio and then click on **File -> New -> New project.**

**Step 2:** Then type the Application name as **“alertapplication″** and click **Next.**

**Step 3:** Then select the **Empty Activity**and click **Next.**

**Step 4:** Finally click F**inish.**

**Step 5:** Click on **app -> res -> layout -> activity\_main.xml.**

**Step 6:** Now click on **code** Then delete the code which is there and type the new code that designs the front end of your application.

**Step 7:** Click on **app -> java -> com.example.alertapplication -> MainActivity.**

**Step 8:** Then delete the code which is there and type the new code that displays alert message.

**Step 9:** Now run the application to see the output.

**Source Code**

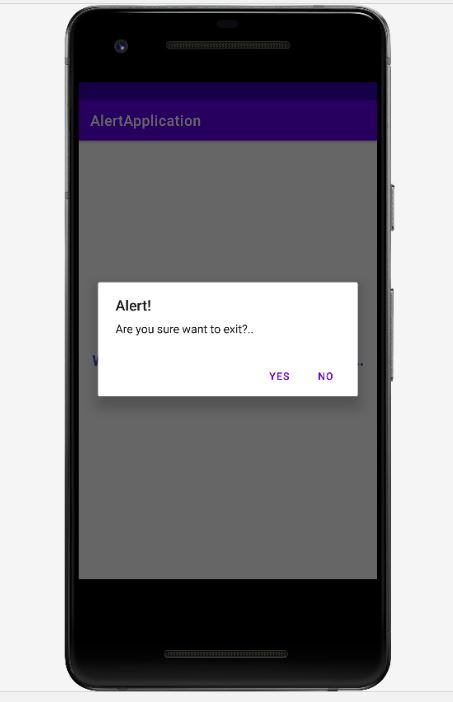
**activity\_main.xml**

*<?*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">  
  
 <TextView  
 android:layout\_width="wrap\_content"  
 android:layout\_height="wrap\_content"  
 android:text="Welcome to Alert Meassge Application!..."  
 android:textColor="#3F51B5"  
 android:textSize="20sp"  
 android:textStyle="bold"  
 app:layout\_constraintBottom\_toBottomOf="parent"  
 app:layout\_constraintLeft\_toLeftOf="parent"  
 app:layout\_constraintRight\_toRightOf="parent"  
 app:layout\_constraintTop\_toTopOf="parent" />  
  
</androidx.constraintlayout.widget.ConstraintLayout>

**MainActivity.java**

package com.example.alertapplication;  
import androidx.appcompat.app.AppCompatActivity;  
import android.app.AlertDialog;  
import android.app.Dialog;  
import android.content.DialogInterface;  
import android.os.Bundle;  
  
public class MainActivity extends AppCompatActivity {  
  
 @Override  
 protected void onCreate(Bundle savedInstanceState) {  
 super.onCreate(savedInstanceState);  
 setContentView(R.layout.*activity\_main*);  
 }  
  
 @Override  
 public void onBackPressed() {  
 AlertDialog.Builder builder=new AlertDialog.Builder(MainActivity.this);  
 builder.setTitle("Alert!");  
 builder.setMessage("Are you sure want to exit?..");  
 builder.setCancelable(false);  
 builder.setNegativeButton("Yes", (DialogInterface.OnClickListener)(dialog,which)->{  
 finish();  
 });  
 builder.setPositiveButton("No",(DialogInterface.OnClickListener)(dialog,which)->{  
 dialog.cancel();  
 });  
 AlertDialog alert=builder.create();  
 alert.show();  
 }  
}

**Output**



**8.Alarm Clock**

**Aim**

To develop a simple android application that create an alarm clock.

**Algorithm**

**Step 1 :** Open Android Studio and then click on **File -> New -> New project.**

**Step 2:** Then type the Application name as **“alarmapplication″** and click **Next.**

**Step 3:** Then select the **Empty Activity**and click **Next.**

**Step 4:** Finally click F**inish.**

**Step 5:** Click on **app -> res -> layout -> activity\_main.xml.**

**Step 6:** Now click on **code** Then delete the code which is there and type the new code that designs the alarm clock.

**Step 7:** Click on **app -> java -> com.example.alarmapplication -> MainActivity.**

**Step 8:** Then delete the code which is there and type the new code that runs the alarm clock.

**Step 9:** Now run the application to see the output.

**Source Code**

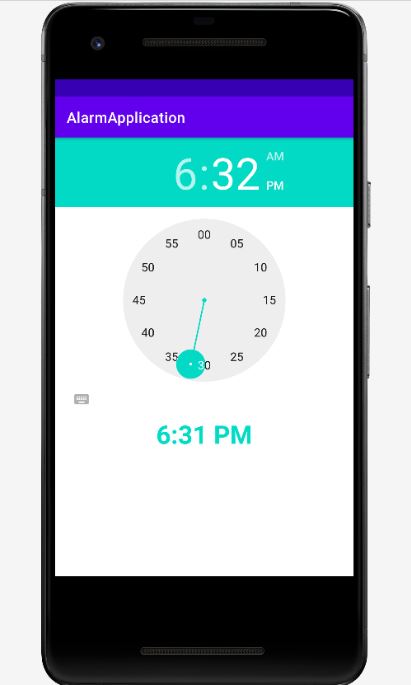
**activity\_main.xml**

*<?*xml version="1.0" encoding="utf-8"*?>*<LinearLayout xmlns:android="http://schemas.android.com/apk/res/android"  
 xmlns:app="http://schemas.android.com/apk/res-auto"  
 xmlns:tools="http://schemas.android.com/tools"  
 android:layout\_width="match\_parent"  
 android:layout\_height="match\_parent"  
 android:orientation="vertical"  
 tools:context=".MainActivity">  
 <TimePicker  
 android:id="@+id/TimePicker"  
 android:layout\_width="wrap\_content"  
 android:layout\_height="wrap\_content"></TimePicker>  
  
 <TextClock  
 android:id="@+id/TextClock"  
 android:layout\_width="match\_parent"  
 android:layout\_height="100sp"  
 android:textAlignment="center"  
 android:textColor="#01DCC7"  
 android:textSize="35sp"  
 android:textStyle="bold"></TextClock>  
</LinearLayout>

**MainActivity.java**

package com.example.alarmapplication;  
import androidx.appcompat.app.AppCompatActivity;  
  
import android.media.Ringtone;  
import android.media.RingtoneManager;  
import android.os.Bundle;  
import android.widget.TextClock;  
import android.widget.TimePicker;  
  
import java.util.Timer;  
import java.util.TimerTask;  
public class MainActivity extends AppCompatActivity {  
 TimePicker alarmtime;  
 TextClock currentTime;  
 @Override  
 protected void onCreate(Bundle savedInstanceState) {  
 super.onCreate(savedInstanceState);  
 setContentView(R.layout.*activity\_main*);  
 alarmtime=findViewById(R.id.*TimePicker*);  
 currentTime=findViewById(R.id.*TextClock*);  
 final Ringtone r= RingtoneManager.*getRingtone*(getApplicationContext(),RingtoneManager.*getDefaultUri*(RingtoneManager.*TYPE\_RINGTONE*));  
 Timer t=new Timer();  
 t.scheduleAtFixedRate(new TimerTask() {  
 @Override  
 public void run() {  
 if(currentTime.getText().toString().equals(Alarmtime())) {  
 r.play();  
 }  
 else {  
 r.stop();  
 }  
 }  
 },0,1000);  
 }  
 public String Alarmtime()  
 {  
 Integer alarmhour=alarmtime.getCurrentHour();  
 Integer alarmminute=alarmtime.getCurrentMinute();  
 String alarmmin="0";  
 String stringalarmtime;  
 if(alarmminute<10)  
 {  
 alarmmin=alarmmin.concat(alarmminute.toString());  
 }  
 else  
 {  
 alarmmin=alarmminute.toString();  
 }  
 if(alarmhour>12)  
 {  
 alarmhour-=12;  
 stringalarmtime=alarmhour.toString().concat(":").concat(alarmmin).concat(" PM");  
 }  
 else  
 {  
 stringalarmtime=alarmhour.toString().concat(":").concat(alarmmin).concat(" AM");  
 }  
 return stringalarmtime;  
 }  
}

**Output**



**9.Graphical Primitives**

**Aim**

To create a android application that uses graphical primitives.

**Algorithm**

**Step 1 :** Open Android Studio and then click on **File -> New -> New project.**

**Step 2:** Then type the Application name as **“graphical\_primitives″** and click **Next.**

**Step 3:** Then select the **Empty Activity**and click **Next.**

**Step 4:** Finally click F**inish.**

**Step 5:** Click on **app -> res -> layout -> activity\_main.xml.**

**Step 6:** Now click on **code** Then delete the code which is there and type the new code that designs the front end of your application.

**Step 7:** Click on **app -> java -> com.example.graphical\_primitives -> MainActivity.**

**Step 8:** Then delete the code which is there and type the new code that uses graphical primitives.

**Step 9:** Now run the application to see the output.

**Source Code**

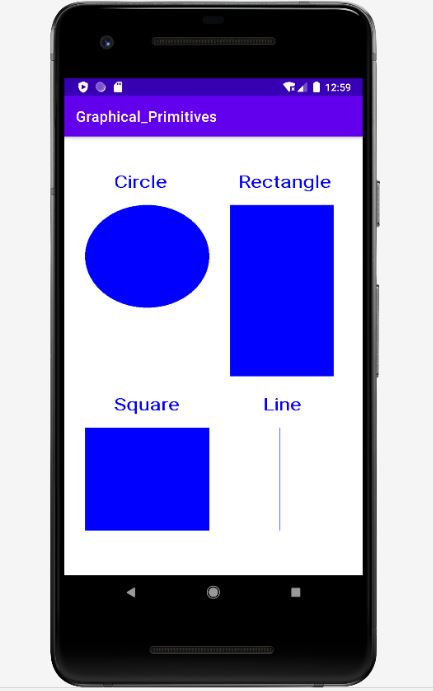
**activity\_main.xml**

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

**MainActivtiy.java**

package com.example.graphical\_primitives;  
import androidx.appcompat.app.AppCompatActivity;  
import android.graphics.Bitmap;  
import android.graphics.Canvas;  
import android.graphics.Color;  
import android.graphics.Paint;  
import android.graphics.drawable.BitmapDrawable;  
import android.os.Bundle;  
import android.widget.ImageView;  
  
public class MainActivity extends AppCompatActivity {  
  
 @Override  
 protected void onCreate(Bundle savedInstanceState) {  
 super.onCreate(savedInstanceState);  
 setContentView(R.layout.*activity\_main*);  
 Bitmap bg=Bitmap.*createBitmap*(720,1280,Bitmap.Config.*ARGB\_8888*);  
 ImageView i=(ImageView) findViewById(R.id.*ImageView*);  
 i.setBackgroundDrawable(new BitmapDrawable(bg));  
 Canvas canvas=new Canvas(bg);  
 Paint paint=new Paint();  
 paint.setColor(Color.*BLUE*);  
 paint.setTextSize(50);  
 canvas.drawText("Rectangle",420,150,paint);  
 canvas.drawRect(400,200,650,700,paint);  
 canvas.drawText("Square",120,800,paint);  
 canvas.drawRect(50,850,350,1150,paint);  
 canvas.drawText("Circle",120,150,paint);  
 canvas.drawCircle(200,350,150,paint);  
 canvas.drawText("Line",480,800,paint);  
 canvas.drawLine(520,850,520,1150,paint);  
 }  
}

**Output**



**10.Navigation Page**

**Aim**

To develop a simple android application that uses Navigation drawer.

**Algorithm**

**Step 1 :** Open Android Studio and then click on **File -> New -> New project.**

**Step 2:** Then type the Application name as **“navigationdrawer″** and click **Next.**

**Step 3:** Then select the **Empty Activity**and click **Next.**

**Step 4:** Finally click F**inish.**

**Step 5:** Click on **app -> res -> layout -> activity\_main.xml.**

**Step 6:** Now click on **code** Then delete the code which is there and type the new code that designs the front end of your application.

**Step 7:** Click on **app -> res. Then right click on res and Click on New-> Android Resource File.**

**Step 8:** Type “menu” in filename and select resource type as menu and then click Ok.

**Step 9:** Then write the code in menu.xml file that creates menu.

**Step 10:** Click on **app -> res -> values -> strings.xml.** Then set names for navigation open and navigation close.

**Step 7:** Click on **app -> java -> com.example.navigationdrawer -> MainActivity.**

**Step 8:** Then delete the code which is there and type the new code

**Step 9:** Now run the application to see the output.

**Source Code**

**activity\_main.xml**

*<?*xml version="1.0" encoding="utf-8"*?>*<androidx.drawerlayout.widget.DrawerLayout  
 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/my\_drawer\_layout"  
 android:layout\_width="match\_parent"  
 android:layout\_height="match\_parent"  
 tools:context=".MainActivity"  
 tools:ignore="HardcodedText">  
 <LinearLayout  
 android:layout\_width="match\_parent"  
 android:layout\_height="match\_parent">  
 <TextView  
 android:layout\_width="match\_parent"  
 android:layout\_height="wrap\_content"  
 android:layout\_marginTop="128dp"  
 android:gravity="center"  
 android:text="Welcome to Navigation Drawer"  
 android:textColor="#2196F3"  
 android:textSize="25sp"  
 android:textStyle="bold" />  
 </LinearLayout>  
 <com.google.android.material.navigation.NavigationView  
 android:layout\_width="wrap\_content"  
 android:layout\_height="match\_parent"  
 android:layout\_gravity="start"  
 app:menu="@menu/menu" />  
</androidx.drawerlayout.widget.DrawerLayout>

**string.xml**

<resources>  
 <string name="app\_name">Navigation Drawer</string>  
 <string name="nav\_open">Open</string>  
 <string name="nav\_close">Close</string>  
</resources>

**menu.xml**

*<?*xml version="1.0" encoding="utf-8"*?>*<menu xmlns:android="http://schemas.android.com/apk/res/android">  
<item android:id="@+id/nav\_Account"  
 android:title="Account"></item>  
 <item android:id="@+id/nav\_settings"  
 android:title="Settings"></item>  
 <item android:id="@+id/nav\_Logout"  
 android:title="Log out"></item>  
</menu>

**MainActivity.java**

package com.example.navigationdrawer;  
import androidx.annotation.NonNull;  
import androidx.appcompat.app.ActionBarDrawerToggle;  
import androidx.appcompat.app.AppCompatActivity;  
import androidx.drawerlayout.widget.DrawerLayout;  
import android.os.Bundle;  
import android.view.MenuItem;  
import android.widget.Toast;  
public class MainActivity extends AppCompatActivity {  
 private DrawerLayout drawerlayout;  
 private ActionBarDrawerToggle actionbardrawertoggle;  
  
 @Override  
 protected void onCreate(Bundle savedInstanceState) {  
 super.onCreate(savedInstanceState);  
 setContentView(R.layout.*activity\_main*);  
 drawerlayout = (DrawerLayout) findViewById(R.id.*my\_drawer\_layout*);  
 actionbardrawertoggle = new ActionBarDrawerToggle(MainActivity.this, drawerlayout, R.string.*nav\_open*, R.string.*nav\_close*);  
 drawerlayout.addDrawerListener(actionbardrawertoggle);  
 actionbardrawertoggle.syncState();  
 getSupportActionBar().setDisplayHomeAsUpEnabled(true);  
 }  
  
 @Override  
 public boolean onOptionsItemSelected(@NonNull MenuItem item) {  
 if(actionbardrawertoggle.onOptionsItemSelected(item)) {  
 return true;  
 }  
 else {  
 return super.onOptionsItemSelected(item);  
 }  
 }  
}

**Output**

