|  |
| --- |
| **Ex. No: 01 Develop an application that uses GUI components, Font and Colors**  **Date:** |

**Aim:**

To develop a Simple Android Application that uses GUI components, Font and Colors.

**Procedure:**

**Creating a New project:**

∙ Open Android Studio and then click on **File -> New -> New project**.

∙ Then type the Application name as **“exno1″**and click Next.

∙ Then **select the Minimum SDK** as shown below and click Next.

∙ Then select the **Empty Activity** and click Next.

∙ Finally click **Finish**.

∙ It will take some time to build and load the project.

∙ After completion it will look as given below.

**Designing layout for the Android Application:**

∙ Click on **app -> res -> layout -> activity\_main.xml**.

∙ Now click on Text as shown below.

∙ Then delete the code which is there and type the code as given below.

**Code for Activity\_main.xml:**

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

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

android:layout\_width="match\_parent"

android:layout\_height="match\_parent">

<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/button1"

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/button2"

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>

∙ Now click on Design and your application will look as given below.

∙ So now the designing part is completed.

**Java Coding for the Android Application:**

∙ Click on **app -> java -> com.example.exno1 -> MainActivity**.

∙ Then delete the code which is there and type the code as given below. **Code for MainActivity.java:**

package com.example.exno1;

import android.graphics.Color;

//import android.support.v7.app.AppCompatActivity;

import android.os.Bundle;

import android.view.View;

import android.widget.Button;

import android.widget.TextView;

import androidx.appcompat.app.AppCompatActivity;

public class MainActivity extends AppCompatActivity

{

int ch=1;

float font=30;

@Override

protected void onCreate(Bundle savedInstanceState) {

super.onCreate(savedInstanceState);

setContentView(R.layout.activity\_main);

final TextView t= (TextView) findViewById(R.id.textView); Button b1= (Button) findViewById(R.id.button1); b1.setOnClickListener(new View.OnClickListener() { @Override

public void onClick(View v) {

t.setTextSize(font);

font = font + 5;

if (font == 50)

font = 30;

}

});

Button b2= (Button) findViewById(R.id.button2); b2.setOnClickListener(new View.OnClickListener() { @Override

public void onClick(View v) {

switch (ch) {

case 1:

t.setTextColor(Color.RED);

break;

case 2:

t.setTextColor(Color.GREEN);

break;

case 3:

t.setTextColor(Color.BLUE);

break;

case 4:

t.setTextColor(Color.CYAN);

break;

case 5:

t.setTextColor(Color.YELLOW);

break;

case 6:

t.setTextColor(Color.MAGENTA);

break;

}

ch++;

if (ch == 7)

ch = 1;

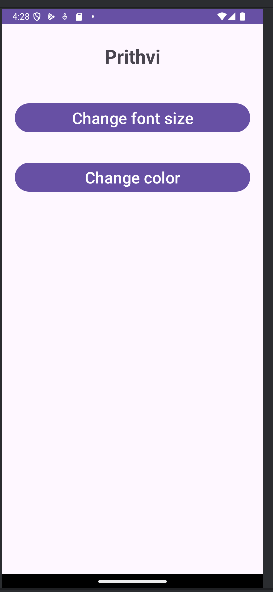
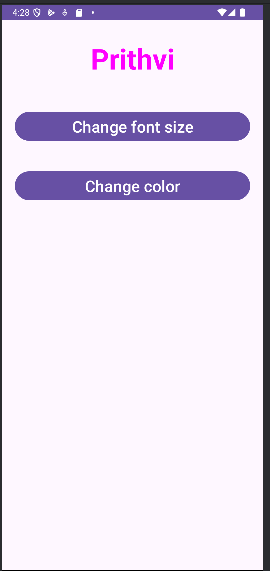
}

});

}

}

**Output:**

****

**Result:**

Thus a Simple Android Application that uses GUI components, Font and Colours is developed and executed successfully.

|  |
| --- |
| **Ex. No. 02 Develop an application that uses Layout Managers and Event Listeners**  **Date:** |

**Aim:**

To develop a Simple Android Application that uses Layout Managers and Event Listeners. **Procedure:**

**Creating a New project:**

∙ Open Android Studio and then click on **File -> New -> New project**.

∙ Then type the Application name as **“exno2″** and click Next.

∙ Then **select the Minimum SDK** as shown below and click Next.

∙ Then **select the Empty Activity** and click Next.

∙ Finally click **Finish**.

∙ It will take some time to build and load the project.

∙ After completion it will look as given below.

**Creating Second Activity for the Android Application:**

∙ Click on File -> New -> Activity -> Empty Activity.

∙ Type the Activity Name as **SecondActivity** and click Finish button. ∙ Thus Second Activity For the application is created.

**Designing Layout for Main Activity:**

∙ Click on **app -> res -> layout -> activity\_main.xml**.

∙ Now click on Text as shown below.

∙ Then delete the code which is there and type the code as given below. **Code for 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:text="Details Form"

android:textSize="25sp"

android:gravity="center"/>

</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\_margin="10dp" android:layout\_row="0"

android:layout\_column="0"

android:text="Name"

android:textSize="20sp"

android:gravity="center"/>

<EditText

android:id="@+id/editText"

android:layout\_width="wrap\_content" android:layout\_height="wrap\_content" android:layout\_margin="10dp"

android:layout\_row="0"

android:layout\_column="1"

android:ems="10"/>

<TextView

android:id="@+id/textView2" android:layout\_width="wrap\_content" android:layout\_height="wrap\_content" android:layout\_margin="10dp" android:layout\_row="1"

android:layout\_column="0"

android:text="Reg.No"

android:textSize="20sp"

android:gravity="center"/>

<EditText

android:id="@+id/editText2" android:layout\_width="wrap\_content" android:layout\_height="wrap\_content" android:layout\_margin="10dp" android:layout\_row="1"

android:layout\_column="1"

android:inputType="number" android:ems="10"/>

<TextView

android:id="@+id/textView3" android:layout\_width="wrap\_content" android:layout\_height="wrap\_content" android:layout\_margin="10dp" android:layout\_row="2"

android:layout\_column="0"

android:text="Dept"

android:textSize="20sp"

android:gravity="center"/>

<Spinner

android:id="@+id/spinner"

android:layout\_width="wrap\_content" android:layout\_height="wrap\_content" android:layout\_margin="10dp" android:layout\_row="2"

android:layout\_column="1"

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>

**Designing Layout for Second Activity:**

∙ Click on **app -> res -> layout -> activity\_second.xml**.

∙ Now click on Text as shown below.

∙ Then delete the code which is there and type the code as given below. **Code for Activity\_second.xml:**

<?xmlversion="1.0"encoding="utf-8"?>

<LinearLayoutxmlns: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.devang.exno2.SecondActivity" 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>

∙ Now click on Design and your activity will look as given below. ∙ So now the designing part of Second Activity is also completed. **Java Coding for the Android Application:**

∙ Java Coidng for Main Activity:

∙ Click on **app -> java -> com.example.exno2 -> MainActivity**. ∙ Then delete the code which is there and type the code as given below. **Code for MainActivity.java:**

package com.example.exno2;

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={"CSE","ECE","IT","Mech","Civil"};

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,SecondActivity.class

i.putExtra("name\_key", name);

i.putExtra("reg\_key",reg);

i.putExtra("dept\_key", dept);

startActivity(i);

}

});

}

}

**Java Coding for Second Activity:**

∙ Click on **app -> java -> com.example.exno2 -> SecondActivity**.

∙ Then delete the code which is there and type the code as given below.

**Code for SecondActivity.java:**

package com.example.exno2;

import android.content.Intent;

//import android.support.v7.app.AppCompatActivity; import android.os.Bundle;

import android.widget.TextView;

import androidx.appcompat.app.AppCompatActivity;

public class SecondActivity extends AppCompatActivity { TextView t1,t2,t3;

String name,reg,dept;

@Override

protected void onCreate(Bundle savedInstanceState) { super.onCreate(savedInstanceState);

setContentView(R.layout.activity\_second);

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);

t2.setText(reg);

t3.setText(dept);

}

}

**Output:**

**A screenshot of a phone

Description automatically generatedA screenshot of a phone

Description automatically generated**

**Result: Result: SsS**

Thus a Simple Android Application that uses Layout Managers and Event Listeners is developed and executed successfully.

Thus a Simple Android Application that uses Layout Managers and Event Listeners is developed and executed successfully.

**Rt: Result: SSA**

**Result:**

Thus a Simple Android Application that uses Layout Managers and Event Listeners is developed and executed successfully.

|  |
| --- |
| **Ex No. 03 Write an application that draws Basic Graphical Primitives on the screen**  **Date:** |

**Aim:**

To develop a Simple Android Application that draws basic Graphical Primitives on the screen. **Procedure:**

**Creating a New project:**

∙ Open Android Studio and then click on **File -> New -> New project**.

∙ Then type the Application name as **“exno3″** and click Next.

∙ Then **select the Minimum SDK** as shown below and click Next.

∙ Then **select the Empty Act**ivity and click Next.

∙ Finally click **Finish**.

∙ It will take some time to build and load the project.

∙ After completion it will look as given below.

**Designing layout for the Android Application:**

∙ Click on **app -> res -> layout -> activity\_main.xml**.

∙ Now click on Text as shown below.

∙ Then delete the code which is there and type the code as given below.

**Code for 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>

∙ Now click on Design and your application will look as given below.

∙ So now the designing part is completed.

**Java Coding for the Android Application:**

∙ Click on **app -> java -> com.example.exno3 -> MainActivity**.

∙ Then delete the code which is there and type the code as given below.

**Code for MainActivity.java:**

package com.example.exno3;

import android.app.Activity;

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 Activity

{

@Override

public 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("Circle", 120, 150, paint);

canvas.drawCircle(200, 350, 150, paint);

canvas.drawText("Square", 120, 800, paint);

canvas.drawRect(50, 850, 350, 1150, paint);

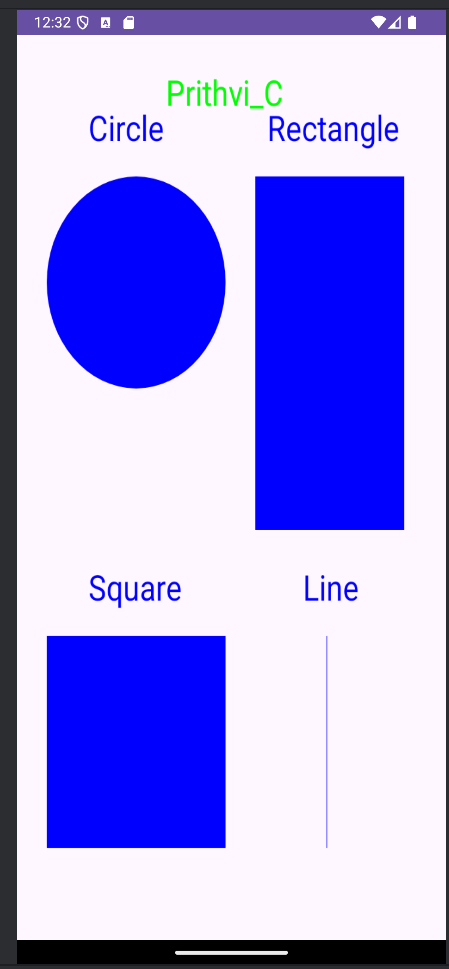
canvas.drawText("Line", 480, 800, paint);

canvas.drawLine(520, 850, 520, 1150, paint);

}

}

**Output:**

****

**Result:**

Thus a Simple Android Application that draws basic Graphical Primitives on the screen is developed and executed successfully.

|  |
| --- |
| **Ex. No. 04 Develop an application that makes use of database**  **Date:** |

**Aim:**

To develop a Simple Android Application that makes use of Database.

**Procedure:**

**Creating a New project:**

∙ Open Android Studio and then click on **File -> New -> New project**.

∙ Then type the Application name as **“exno4″** and click Next.

∙ Then **select the Minimum S**DK as shown below and click Next.

∙ Then **select the Empty Activity** and click Next.

∙ Finally click **Finish**.

∙ It will take some time to build and load the project.

∙ After completion it will look as given below.

**Designing layout for the Android Application:**

∙ Click on **app -> res -> layout -> activity\_main.xml**.

∙ Now click on Text as shown below.

∙ Then delete the code which is there and type the code as given below.

**Code for Activity\_main.xml:**

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

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

android:layout\_width="match\_parent"

android:layout\_height="match\_parent">

<TextView

android:layout\_width="wrap\_content"

android:layout\_height="wrap\_content"

android:layout\_x="50dp"

android:layout\_y="20dp"

android:text="Student Details"

android:textSize="30sp" />

<TextView

android:layout\_width="wrap\_content"

android:layout\_height="wrap\_content"

android:layout\_x="20dp"

android:layout\_y="110dp"

android:text="Enter Rollno:"

android:textSize="20sp" />

<EditText

android:id="@+id/Rollno"

android:layout\_width="150dp" android:layout\_height="wrap\_content" android:layout\_x="175dp"

android:layout\_y="100dp"

android:inputType="number" android:textSize="20sp" />

<TextView

android:layout\_width="wrap\_content" android:layout\_height="wrap\_content" android:layout\_x="20dp"

android:layout\_y="160dp"

android:text="Enter Name:"

android:textSize="20sp" />

<EditText

android:id="@+id/Name"

android:layout\_width="150dp" android:layout\_height="wrap\_content" android:layout\_x="175dp"

android:layout\_y="150dp"

android:inputType="text"

android:textSize="20sp" />

<TextView

android:layout\_width="wrap\_content" android:layout\_height="wrap\_content" android:layout\_x="20dp"

android:layout\_y="210dp"

android:text="Enter Marks:"

android:textSize="20sp" />

<EditText

android:id="@+id/Marks"

android:layout\_width="150dp" android:layout\_height="wrap\_content"

android:layout\_x="175dp"

android:layout\_y="200dp"

android:inputType="number" android:textSize="20sp" />

<Button

android:id="@+id/Insert"

android:layout\_width="150dp" android:layout\_height="wrap\_content" android:layout\_x="25dp"

android:layout\_y="300dp"

android:text="Insert"

android:textSize="30dp" />

<Button

android:id="@+id/Delete"

android:layout\_width="150dp" android:layout\_height="wrap\_content" android:layout\_x="200dp"

android:layout\_y="300dp"

android:text="Delete"

android:textSize="30dp" />

<Button

android:id="@+id/Update"

android:layout\_width="150dp" android:layout\_height="wrap\_content" android:layout\_x="25dp"

android:layout\_y="400dp"

android:text="Update"

android:textSize="30dp" />

<Button

android:id="@+id/View"

android:layout\_width="150dp" android:layout\_height="wrap\_content" android:layout\_x="200dp"

android:layout\_y="400dp"

android:text="View"

android:textSize="30dp" />

<Button

android:id="@+id/ViewAll"

android:layout\_width="200dp"

android:layout\_height="wrap\_content"

android:layout\_x="100dp"

android:layout\_y="500dp"

android:text="View All"

android:textSize="30dp" />

</AbsoluteLayout>

**Java Coding for the Android Application:**

∙ Click on **app -> java -> com.example.exno4 -> MainActivity**. ∙ Then delete the code which is there and type the code as given below. **Code for MainActivity.java:**

packagecom.example.exno4;

import android.app.Activity;

import android.app.AlertDialog.Builder;

import android.content.Context;

import android.database.Cursor;

import android.database.sqlite.SQLiteDatabase;

import android.os.Bundle;

import android.view.View;

import android.view.View.OnClickListener;

import android.widget.Button;

import android.widget.EditText;

public class MainActivity extends Activity implements OnClickListener {

EditText Rollno,Name,Marks;

Button Insert,Delete,Update,View,ViewAll;

SQLiteDatabase db;

@Override

public void onCreate(Bundle savedInstanceState)

{

super.onCreate(savedInstanceState);

setContentView(R.layout.activity\_main); Rollno=(EditText)findViewById(R.id.Rollno);

Name=(EditText)findViewById(R.id.Name);

Marks=(EditText)findViewById(R.id.Marks);

Insert=(Button)findViewById(R.id.Insert);

Delete=(Button)findViewById(R.id.Delete);

Update=(Button)findViewById(R.id.Update);

View=(Button)findViewById(R.id.View);

ViewAll=(Button)findViewById(R.id.ViewAll);

Insert.setOnClickListener(this);

Delete.setOnClickListener(this);

Update.setOnClickListener(this);

View.setOnClickListener(this);

ViewAll.setOnClickListener(this);

db=openOrCreateDatabase("StudentDB", Context.MODE\_PRIVATE, null); db.execSQL("CREATE TABLE IF NOT EXISTS student(rollno VARCHAR,name VARCHAR,marks VARCHAR);");

}

public void onClick(View view)

{

if(view==Insert)

{

if(Rollno.getText().toString().trim().length()==0||

Name.getText().toString().trim().length()==0||

Marks.getText().toString().trim().length()==0)

{

showMessage("Error", "Please enter all values");

return;

}

db.execSQL("INSERT INTO student VALUES('"+Rollno.getText()+"','"+Name.getText()+ "','"+Marks.getText()+"');");

showMessage("Success", "Record added");

clearText();

}

if(view==Delete)

{

if(Rollno.getText().toString().trim().length()==0)

{

showMessage("Error", "Please enter Rollno");

return;

}

Cursor c=db.rawQuery("SELECT \* FROM student WHERE rollno='"+Rollno.getText()+"'", null); if(c.moveToFirst())

{

db.execSQL("DELETE FROM student WHERE rollno='"+Rollno.getText()+"'"); showMessage("Success", "Record Deleted");

}

else

{

showMessage("Error", "Invalid Rollno");

}

clearText();

}

if(view==Update)

{

if(Rollno.getText().toString().trim().length()==0)

{

showMessage("Error", "Please enter Rollno");

return;

}

Cursor c=db.rawQuery("SELECT \* FROM student WHERE rollno='"+Rollno.getText()+"'", null); if(c.moveToFirst()) {

db.execSQL("UPDATE student SET name='" + Name.getText() + "',marks='" + Marks.getText() + "' WHERE rollno='"+Rollno.getText()+"'");

showMessage("Success", "Record Modified");

}

else {

showMessage("Error", "Invalid Rollno");

}

clearText();

}

if(view==View)

{

if(Rollno.getText().toString().trim().length()==0)

{

showMessage("Error", "Please enter Rollno");

return;

}

Cursor c=db.rawQuery("SELECT \* FROM student WHERE rollno='"+Rollno.getText()+"'", null); if(c.moveToFirst())

{

Name.setText(c.getString(1));

Marks.setText(c.getString(2));

}

else

{

showMessage("Error", "Invalid Rollno");

clearText();

}

}

if(view==ViewAll)

{

Cursor c=db.rawQuery("SELECT \* FROM student", null);

if(c.getCount()==0)

{

showMessage("Error", "No records found");

return;

}

StringBuffer buffer=new StringBuffer();

while(c.moveToNext())

{

buffer.append("Rollno: "+c.getString(0)+"\n");

buffer.append("Name: "+c.getString(1)+"\n");

buffer.append("Marks: "+c.getString(2)+"\n\n");

}

showMessage("Student Details", buffer.toString());

}

}

public void showMessage(String title,String message)

{

Builder builder=new Builder(this);

builder.setCancelable(true);

builder.setTitle(title);

builder.setMessage(message);

builder.show();

}

public void clearText()

{

Rollno.setText("");

Name.setText("");

Marks.setText("");

Rollno.requestFocus();

}

}

**Output:**

**A screenshot of a phone

Description automatically generated A screenshot of a computer

Description automatically generated**

**A screenshot of a computer

Description automatically generatedA screenshot of a phone

Description automatically generated**

**Result:**

Thus a Simple Android Application that makes use of Database is developed and executed successfully.

|  |
| --- |
| **Ex. No. 06 Implement an application that uses Multi-threading**  **Date:** |

**Aim:**

To develop an Android Application that implements Multi threading.

**Procedure:**

**Creating a New project:**

∙ Open Android Studio and then click on **File -> New -> New project**.

∙ Then type the Application name as **“exno6″**and click Next.

∙ Then **select the Minimum SDK** as shown below and click Next.

∙ Then **select the Empty Activity** and click Next.

∙ Finally click **Finish**.

∙ It will take some time to build and load the project.

∙ After completion it will look as given below.

**Designing layout for the Android Application:**

∙ Click on **app -> res -> layout -> activity\_main.xml.**

∙ Now click on Text as shown below.

∙ Then delete the code which is there and type the code as given below.

**Code for Activity\_main.xml:**

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

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

android:layout\_width="match\_parent"

android:layout\_height="match\_parent"

android:orientation="vertical" >

<ImageView

android:id="@+id/imageView"

android:layout\_width="250dp"

android:layout\_height="250dp"

android:layout\_margin="50dp"

android:layout\_gravity="center" />

<Button

android:id="@+id/button"

android:layout\_width="wrap\_content"

android:layout\_height="wrap\_content"

android:layout\_margin="10dp"

android:layout\_gravity="center"

android:text="Load Image 1" />

<Button

android:id="@+id/button2"

android:layout\_width="wrap\_content"

android:layout\_height="wrap\_content"

android:layout\_margin="10dp"

android:layout\_gravity="center"

android:text="Load image 2" />

</LinearLayout>

∙ Now click on Design and your application will look as given below.

∙ So now the designing part is completed.

**Java Coding for the Android Application:**

∙ Click on **app -> java -> com.example.exno6 -> MainActivity**. ∙ Then delete the code which is there and type the code as given below.

**Code for MainActivity.java:**

packagecom.example.exno6;

import android.os.Bundle;

//import android.support.v7.app.AppCompatActivity;

import android.view.View;

import android.widget.Button;

import android.widget.ImageView;

import androidx.appcompat.app.AppCompatActivity;

public class MainActivity extends AppCompatActivity

{

ImageView img;

Button bt1,bt2;

@Override

protected void onCreate(Bundle savedInstanceState)

{

super.onCreate(savedInstanceState);

setContentView(R.layout.activity\_main);

bt1 = (Button)findViewById(R.id.button);

bt2= (Button) findViewById(R.id.button2);

img = (ImageView)findViewById(R.id.imageView);

bt1.setOnClickListener(new View.OnClickListener() {

@Override

public void onClick(View v)

{

new Thread(new Runnable()

{

@Override

public void run()

{

img.post(new Runnable()

{

@Override

public void run()

{

img.setImageResource(R.drawable.india1); }

});

}

}).start();

}

});

bt2.setOnClickListener(new View.OnClickListener() {

@Override

public void onClick(View v)

{

new Thread(new Runnable()

{

@Override

public void run()

{

img.post(new Runnable()

{

@Override

public void run()

{

img.setImageResource(R.drawable.india2); }

});

}

}).start();

}

});

}

}

Output:

A screenshot of a phone

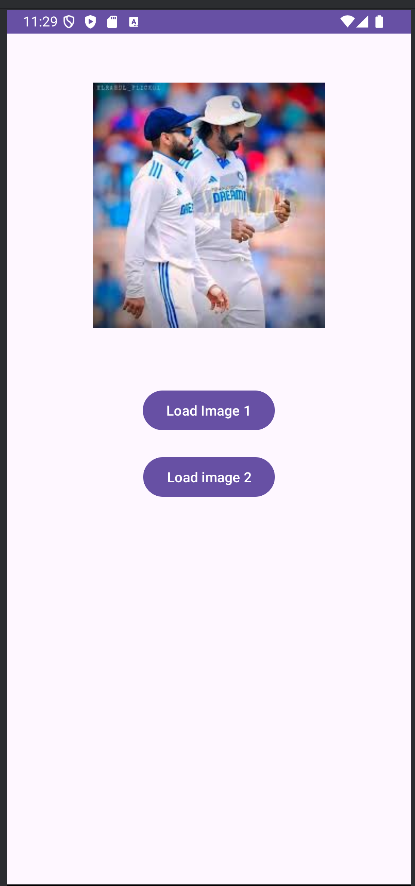
Description automatically generated

Load Image 1

A screenshot of a phone

Description automatically generated

Load Image 2 :



**Result:**

Thus Android Application that implements Multi threading is developed and executed successfully.

**MAD experiment : 07**

**Types of Sensors**

**DATE : 26.02.2025**

**Aim :**

To list the available sensors that are available in the android studio.

**Activity\_main.XML :**

<?xml version="1.0" encoding="utf-8"?>  
<LinearLayout xmlns:android="http://schemas.android.com/apk/res/android"  
android:layout\_width="match\_parent"  
android:layout\_height="match\_parent"  
android:orientation="vertical"  
android:padding="16dp">  
 <TextView  
 android:id="@+id/infoTextView"  
 android:layout\_width="wrap\_content"  
 android:layout\_height="wrap\_content"  
 android:textSize="18sp"  
 android:text="Prithvi C 22BCS093"  
 android:textStyle="bold"  
 android:textColor="#4CAF50"  
 android:layout\_marginBottom="16dp"  
 android:background="#E0F7FA"/>  
<TextView  
 android:id="@+id/sensorTextView"  
 android:layout\_width="wrap\_content"  
 android:layout\_height="wrap\_content"  
 android:text="Available Sensors:"  
 android:background="#FFEBEE"  
 android:textColor="#D32F2F"/>  
</LinearLayout>

**Main\_activity.java :**

package com.example.exno7;  
import android.content.Context;  
import android.hardware.Sensor;  
import android.hardware.SensorManager;  
import android.os.Bundle;  
import android.widget.TextView;  
import androidx.appcompat.app.AppCompatActivity;  
import java.util.List;  
  
public class MainActivity extends AppCompatActivity {  
  
 @Override  
 protected void onCreate(Bundle savedInstanceState) {  
 super.onCreate(savedInstanceState);  
 setContentView(R.layout.*activity\_main*);  
  
 TextView sensorTextView = findViewById(R.id.*sensorTextView*);  
 SensorManager sensorManager = (SensorManager) getSystemService(Context.*SENSOR\_SERVICE*);  
 List<Sensor> sensorList = sensorManager.getSensorList(Sensor.*TYPE\_ALL*);  
  
 StringBuilder sensorInfo = new StringBuilder("Available Sensors:\n\n");  
 for (Sensor sensor : sensorList) {  
 sensorInfo.append(sensor.getName()).append("\n");  
 }  
  
 sensorTextView.setText(sensorInfo.toString());  
 }  
}

**output :**

A screenshot of a phone

AI-generated content may be incorrect.

22BCS093

Prithvi C

**MAD experiment : 08**

**Accelerometer Sensor**

**AIM**:

To create the Accelerometer sensor using the Android studio.

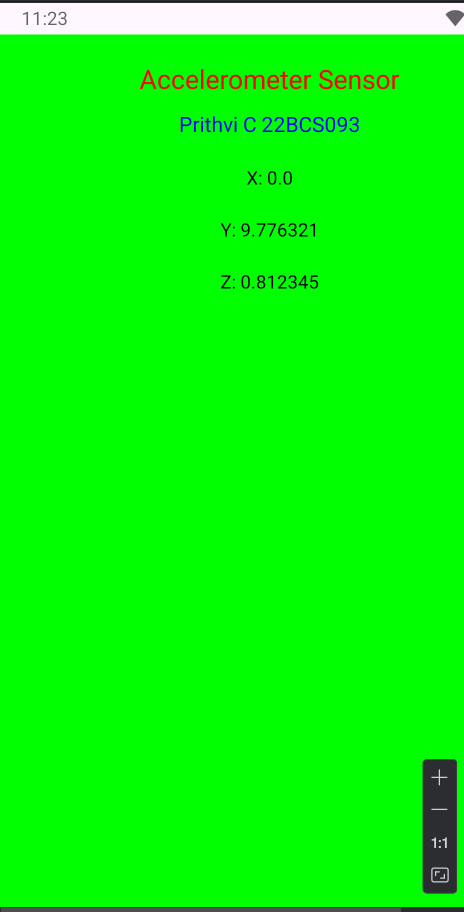
**Activity\_main.XML :**

<?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"  
 android:background="#00FF00">  
  
 <TextView  
 android:id="@+id/textViewXx"  
 android:layout\_width="wrap\_content"  
 android:layout\_height="wrap\_content"  
 android:text="Accelerometer Sensor"  
 android:textColor="#FF0000"  
 android:textSize="20sp"  
 android:layout\_marginTop="20dp"  
 android:layout\_centerHorizontal="true"/>  
  
 <TextView  
 android:id="@+id/textViewXxx"  
 android:layout\_width="wrap\_content"  
 android:layout\_height="wrap\_content"  
 android:text="Prithvi C 22BCS093"  
 android:background="#00FF00"  
 android:textColor="#0000FF"  
 android:textSize="16sp"  
 android:layout\_below="@id/textViewXx"  
 android:layout\_marginTop="10dp"  
 android:layout\_centerHorizontal="true"/>  
  
 <TextView  
 android:id="@+id/textViewX"  
 android:layout\_width="wrap\_content"  
 android:layout\_height="wrap\_content"  
 android:text="X: 0.0"  
 android:textColor="#000000"  
 android:layout\_below="@id/textViewXxx"  
 android:layout\_marginTop="20dp"  
 android:layout\_centerHorizontal="true"/>  
  
 <TextView  
 android:id="@+id/textViewY"  
 android:layout\_width="wrap\_content"  
 android:layout\_height="wrap\_content"  
 android:text="Y: 0.0"  
 android:textColor="#000000"  
 android:layout\_below="@id/textViewX"  
 android:layout\_marginTop="20dp"  
 android:layout\_centerHorizontal="true"/>  
  
 <TextView  
 android:id="@+id/textViewZ"  
 android:layout\_width="wrap\_content"  
 android:layout\_height="wrap\_content"  
 android:text="Z: 0.0"  
 android:textColor="#000000"  
 android:layout\_below="@id/textViewY"  
 android:layout\_marginTop="20dp"  
 android:layout\_centerHorizontal="true"/>  
  
</RelativeLayout>

**Main\_activity.java:**

package com.example.exno8;  
  
import android.hardware.Sensor;  
import android.hardware.SensorEvent;  
import android.hardware.SensorEventListener;  
import android.hardware.SensorManager;  
import android.os.Bundle;  
import android.widget.TextView;  
import androidx.appcompat.app.AppCompatActivity;  
  
public class MainActivity extends AppCompatActivity implements SensorEventListener {  
  
 private SensorManager sensorManager;  
 private Sensor accelerometer;  
 private TextView textViewX, textViewY, textViewZ;  
  
 @Override  
 protected void onCreate(Bundle savedInstanceState) {  
 super.onCreate(savedInstanceState);  
 setContentView(R.layout.*activity\_main*);  
  
 textViewX = findViewById(R.id.*textViewX*);  
 textViewY = findViewById(R.id.*textViewY*);  
 textViewZ = findViewById(R.id.*textViewZ*);  
  
 // Initialize the SensorManager and accelerometer sensor  
 sensorManager = (SensorManager) getSystemService(*SENSOR\_SERVICE*);  
 if (sensorManager != null) {  
 accelerometer = sensorManager.getDefaultSensor(Sensor.*TYPE\_ACCELEROMETER*);  
 }  
 }  
  
 @Override  
 protected void onResume() {  
 super.onResume();  
 // Register the sensor listener when the activity resumes  
 if (accelerometer != null) {  
 sensorManager.registerListener(this, accelerometer, SensorManager.*SENSOR\_DELAY\_NORMAL*);  
 }  
 }  
  
 @Override  
 protected void onPause() {  
 super.onPause();  
 // Unregister the sensor listener when the activity pauses  
 sensorManager.unregisterListener(this);  
 }  
  
 @Override  
 public void onSensorChanged(SensorEvent event) {  
 // Update the UI with the new accelerometer data  
 if (event.sensor.getType() == Sensor.*TYPE\_ACCELEROMETER*) {  
 float x = event.values[0];  
 float y = event.values[1];  
 float z = event.values[2];  
  
 textViewX.setText("X: " + x);  
 textViewY.setText("Y: " + y);  
 textViewZ.setText("Z: " + z);  
 }  
 }  
  
 @Override  
 public void onAccuracyChanged(Sensor sensor, int accuracy) {  
 // Handle accuracy changes if needed  
 }  
}

**output :**



A screen shot of a green screen

AI-generated content may be incorrect.

**Result :**

The given task was executed successfully and the output is verified.

22BCS093

Prithvi C

**PROXIMITY SENSOR**

**AIM :**

**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"

android:background="#00FF00">

<!-- Title: Proximity Sensor -->

<TextView

android:id="@+id/textViewTitle"

android:layout\_width="wrap\_content"

android:layout\_height="wrap\_content"

android:text="Proximity Sensor"

android:textColor="#FF0000"

android:textSize="20sp"

android:layout\_marginTop="20dp"

android:layout\_centerHorizontal="true"/>

<!-- Subtitle: Prithvi C 22BCS093 -->

<TextView

android:id="@+id/textViewSubtitle"

android:layout\_width="wrap\_content"

android:layout\_height="wrap\_content"

android:text="Prithvi C 22BCS093"

android:background="#00FF00"

android:textColor="#000000"

android:textSize="16sp"

android:layout\_below="@id/textViewTitle"

android:layout\_marginTop="10dp"

android:layout\_centerHorizontal="true"/>

<!-- Proximity Value -->

<TextView

android:id="@+id/textViewProximity"

android:layout\_width="wrap\_content"

android:layout\_height="wrap\_content"

android:text="Proximity: 0.0"

android:textColor="#000000"

android:layout\_below="@id/textViewSubtitle"

android:layout\_marginTop="20dp"

android:layout\_centerHorizontal="true"/>

</RelativeLayout>

**Main\_activity.java :**

package com.example.exno9;

import android.hardware.Sensor;

import android.hardware.SensorEvent;

import android.hardware.SensorEventListener;

import android.hardware.SensorManager;

import android.os.Bundle;

import android.widget.TextView;

import androidx.appcompat.app.AppCompatActivity;

public class MainActivity extends AppCompatActivity implements SensorEventListener {

private SensorManager sensorManager;

private Sensor proximitySensor;

private TextView textViewProximity;

@Override

protected void onCreate(Bundle savedInstanceState) {

super.onCreate(savedInstanceState);

setContentView(R.layout.activity\_main);

// Initialize the TextView for proximity value

textViewProximity = findViewById(R.id.textViewProximity);

// Initialize the SensorManager and proximity sensor

sensorManager = (SensorManager) getSystemService(SENSOR\_SERVICE);

if (sensorManager != null) {

proximitySensor = sensorManager.getDefaultSensor(Sensor.TYPE\_PROXIMITY);

}

// Check if the device has a proximity sensor

if (proximitySensor == null) {

textViewProximity.setText("Proximity Sensor not available");

}

}

@Override

protected void onResume() {

super.onResume();

// Register the sensor listener when the activity resumes

if (proximitySensor != null) {

sensorManager.registerListener(this, proximitySensor, SensorManager.SENSOR\_DELAY\_NORMAL);

}

}

@Override

protected void onPause() {

super.onPause();

// Unregister the sensor listener when the activity pauses

sensorManager.unregisterListener(this);

}

@Override

public void onSensorChanged(SensorEvent event) {

// Update the proximity value in the TextView

if (event.sensor.getType() == Sensor.TYPE\_PROXIMITY) {

float proximityValue = event.values[0];

textViewProximity.setText("Proximity: " + proximityValue);

}

}

@Override

public void onAccuracyChanged(Sensor sensor, int accuracy) {

// Handle accuracy changes if needed

}

}

**OUTPUT :**

A green screen with black border

AI-generated content may be incorrect.

**Result :**

Thus a Simple Android Application for Proximity Sensor developed and executed successfully.

|  |
| --- |
| **Ex. No: 10 Develop an application that enables and disables wifi**  **Date: 12.3.25** |

Name: Prithvi C

Roll.No: 22BCS093

**Aim:**

To enables and disables wifi using android studio.

**Procedure:**

**Creating a New project:**

∙ Open Android Studio and then click on **File -> New -> New project**.

∙ Then type the Application name as **“exno1″**and click Next.

∙ Then **select the Minimum SDK** as shown below and click Next.

∙ Then select the **Empty Activity** and click Next.

∙ Finally click **Finish**.

∙ It will take some time to build and load the project.

∙ After completion it will look as given below.

**Designing layout for the Android Application:**

∙ Click on **app -> res -> layout -> activity\_main.xml**.

∙ Now click on Text as shown below.

∙ Then delete the code which is there and type the code as given below.

**Code for 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">  
  
 <Button  
 android:id="@+id/button1"  
 android:layout\_width="wrap\_content"  
 android:layout\_height="wrap\_content"  
 android:layout\_alignParentStart="true"  
 android:layout\_alignParentEnd="true"  
 android:layout\_alignParentBottom="true"  
 android:layout\_marginStart="146dp"  
 android:layout\_marginLeft="76dp"  
 android:layout\_marginEnd="134dp"  
 android:layout\_marginBottom="457dp"  
 android:text="Enable Wifi" />  
 <TextView  
 android:layout\_width="wrap\_content"  
 android:layout\_height="wrap\_content"  
 android:text="Prithvi C"

android:textColor=”@color/green”  
 android:bgColor="@color/black"

android:textSize="20dp"/>  
  
  
 <Button  
 android:id="@+id/button2"  
 android:layout\_width="wrap\_content"  
 android:layout\_height="wrap\_content"  
 android:layout\_alignParentEnd="true"  
 android:layout\_alignParentBottom="true"  
 android:layout\_marginEnd="131dp"  
 android:layout\_marginBottom="241dp"  
 android:text="Disable Wifi" />  
  
  
</RelativeLayout>

∙ Now click on Design and your application will look as given below.

∙ So now the designing part is completed.

**Java Coding for the Android Application:**

∙ Click on **app -> java -> com.example.exno1 -> MainActivity**.

∙ Then delete the code which is there and type the code as given below. **Code for MainActivity.java:**

package com.example.wifiendi;  
  
import androidx.appcompat.app.AppCompatActivity;  
  
import android.content.Context;  
import android.content.Intent;  
import android.net.wifi.WifiManager;  
import android.os.Build;  
import android.os.Bundle;  
import android.view.View;  
import android.widget.Button;  
  
public class MainActivity extends AppCompatActivity {  
  
 Button enableButton, disButton;  
  
 @Override  
 protected void onCreate(Bundle savedInstanceState) {  
 super.onCreate(savedInstanceState);  
 setContentView(R.layout.activity\_main);  
  
 enableButton = findViewById(R.id.button1);  
 disButton = findViewById(R.id.button2);  
  
 enableButton.setOnClickListener(new View.OnClickListener() {  
 @Override  
 public void onClick(View view) {  
 // Check if the device is running Android 10 (API 29) or higher  
 if (Build.VERSION.SDK\_INT >= Build.VERSION\_CODES.Q) {  
 // Direct the user to the Wi-Fi settings page instead of enabling Wi-Fi programmatically  
 Intent intent = new Intent(android.provider.Settings.ACTION\_WIFI\_SETTINGS);  
 startActivity(intent);  
 } else {  
 // For older versions, Wi-Fi can still be enabled programmatically  
 WifiManager wifiManager = (WifiManager) getApplicationContext().getSystemService(Context.WIFI\_SERVICE);  
 wifiManager.setWifiEnabled(true);  
 }  
 }  
 });  
  
 disButton.setOnClickListener(new View.OnClickListener() {  
 @Override  
 public void onClick(View view) {  
 // Check if the device is running Android 10 (API 29) or higher  
 if (Build.VERSION.SDK\_INT >= Build.VERSION\_CODES.Q) {  
 // Direct the user to the Wi-Fi settings page instead of disabling Wi-Fi programmatically  
 Intent intent = new Intent(android.provider.Settings.ACTION\_WIFI\_SETTINGS);  
 startActivity(intent);  
 } else {  
 // For older versions, Wi-Fi can still be disabled programmatically  
 WifiManager wifiManager = (WifiManager) getApplicationContext().getSystemService(Context.WIFI\_SERVICE);  
 wifiManager.setWifiEnabled(false);  
 }  
 }  
 });  
 }  
}

**Output:**

A screenshot of a phone

AI-generated content may be incorrect.

A close-up of a phone screen

AI-generated content may be incorrect.

**Result:**

Thus a Simple Android Application has done to enables and disables wifi.

|  |
| --- |
| **Ex. No. 11 Develop a native application that uses GPS location information Date:12/03/25** |

**Aim:**

To develop an Android Application that uses GPS location information.

**Procedure:**

**Creating a New project:**

∙ Open Android Studio and then click on **File -> New -> New project**.

∙ Then type the Application name as **“exno7″**and click Next.

∙ Then **select the Minimum SDK** as shown below and click Next.

∙ Then **select the Empty Activity** and click Next.

∙ Finally click **Finish**.

∙ It will take some time to build and load the project.

∙ After completion it will look as given below.

**Designing layout for the Android Application:**

∙ Click on **app -> res -> layout -> activity\_main.xml**.

∙ Now click on Text as shown below.

∙ Then delete the code which is there and type the code as given below.

**Code for Activity\_main.xml:**

**<?xml version="1.0" encoding="utf-8"?>**  
**<LinearLayout xmlns:android="http://schemas.android.com/apk/res/android"**  
 **android:layout\_width="match\_parent"**  
 **android:layout\_height="match\_parent"**  
 **android:orientation="vertical"**  
 **android:padding="20dp"**  
 **android:gravity="center">**  
  
 **<TextView**  
 **android:id="@+id/locationText"**  
 **android:layout\_width="wrap\_content"**  
 **android:layout\_height="wrap\_content"**  
 **android:text="Location will be displayed here"**  
 **android:textSize="18sp"**  
 **android:padding="10dp"/>**  
  
 **<Button**  
 **android:id="@+id/getLocationButton"**  
 **android:layout\_width="wrap\_content"**  
 **android:layout\_height="wrap\_content"**  
 **android:text="Get Location"**  
 **android:padding="10dp"**  
 **android:layout\_marginTop="20dp"/>**  
  
**</LinearLayout>**

**Following will be the content of res/values/themes.xml :**

**<resources xmlns:tools="http://schemas.android.com/tools">**  
 **<style name="Theme.LocationApp"**

**parent="Theme.MaterialComponents.DayNight.DarkActionBar">**  
 **<!-- Customize colors -->**  
 **<item name="colorPrimary">@color/purple\_500</item>**  
 **<item name="colorPrimaryVariant">@color/purple\_700</item>**  
 **<item name="colorOnPrimary">@color/white</item>**  
 **<item name="colorSecondary">@color/teal\_200</item>**  
 **<item name="colorOnSecondary">@color/black</item>**  
 **<item name="android:statusBarColor">@color/purple\_700</item>**  
 **<item name="android:navigationBarColor">@color/black</item>**  
 **</style>**  
**</resources>**

**Adding permissions in Manifest for the Android Application:**

∙ Click on **app -> manifests -> AndroidManifest.xml.**

**Code for AndroidManifest.xml:**

<?xml version="1.0" encoding="utf-8"?>  
<manifest xmlns:android="http://schemas.android.com/apk/res/android">  
  
 <!-- Permissions for GPS -->  
 <uses-permission android:name="android.permission.ACCESS\_FINE\_LOCATION"/>  
 <uses-permission android:name="android.permission.ACCESS\_COARSE\_LOCATION"/>  
  
 <uses-feature android:name="android.hardware.location.gps" />  
  
 <application  
 android:allowBackup="true"  
 android:icon="@mipmap/ic\_launcher"  
 android:label="Location App"  
 android:roundIcon="@mipmap/ic\_launcher\_round"  
 android:supportsRtl="true"  
 android:theme="@style/Theme.LocationApp">  
  
 <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>

**Java Coding for the Android Application:**

∙ Click on **app -> java -> com.example.exno7 -> MainActivity**.

∙ Then delete the code which is there and type the code as given below.

**Code for MainActivity.java:**

package com.example.exno11;  
  
import android.Manifest;  
import android.content.pm.PackageManager;  
import android.location.Location;  
import android.os.Bundle;  
import android.widget.Button;  
import android.widget.TextView;  
import android.widget.Toast;  
import androidx.annotation.NonNull;  
import androidx.appcompat.app.AppCompatActivity;  
import androidx.core.app.ActivityCompat;  
  
import com.google.android.gms.location.FusedLocationProviderClient;  
import com.google.android.gms.location.LocationServices;  
import com.google.android.gms.tasks.OnSuccessListener;  
import com.google.android.gms.tasks.Task;  
  
public class MainActivity extends AppCompatActivity {  
  
 private static final int LOCATION\_PERMISSION\_REQUEST\_CODE = 1;  
 private FusedLocationProviderClient fusedLocationProviderClient;  
 private TextView locationText;  
 private Button getLocationButton;  
  
 @Override  
 protected void onCreate(Bundle savedInstanceState) {  
 super.onCreate(savedInstanceState);  
 setContentView(R.layout.activity\_main);  
  
 locationText = findViewById(R.id.locationText);  
 getLocationButton = findViewById(R.id.getLocationButton);  
  
 // Initialize FusedLocationProviderClient  
 fusedLocationProviderClient = LocationServices.getFusedLocationProviderClient(this);  
  
 getLocationButton.setOnClickListener(v -> getLocation());  
 }  
  
 private void getLocation() {  
 // Check for permission  
 if (ActivityCompat.checkSelfPermission(this, Manifest.permission.ACCESS\_FINE\_LOCATION)  
 != PackageManager.PERMISSION\_GRANTED &&  
 ActivityCompat.checkSelfPermission(this, Manifest.permission.ACCESS\_COARSE\_LOCATION)  
 != PackageManager.PERMISSION\_GRANTED) {  
  
 // Request permission  
 ActivityCompat.requestPermissions(this,  
 new String[]{Manifest.permission.ACCESS\_FINE\_LOCATION, Manifest.permission.ACCESS\_COARSE\_LOCATION},  
 LOCATION\_PERMISSION\_REQUEST\_CODE);  
 return;  
 }  
  
 // Get last known location  
 Task<Location> task = fusedLocationProviderClient.getLastLocation();  
 task.addOnSuccessListener(new OnSuccessListener<Location>() {  
 @Override  
 public void onSuccess(Location location) {  
 if (location != null) {  
 double latitude = location.getLatitude();  
 double longitude = location.getLongitude();  
 locationText.setText("Latitude: " + latitude + "\nLongitude: " + longitude);  
 } else {  
 locationText.setText("Location not available");  
 }  
 }  
 });  
 }  
  
 // Handle permission result  
 @Override  
 public void onRequestPermissionsResult(int requestCode, @NonNull String[] permissions, @NonNull int[] grantResults) {  
 super.onRequestPermissionsResult(requestCode, permissions, grantResults);  
 if (requestCode == LOCATION\_PERMISSION\_REQUEST\_CODE) {  
 if (grantResults.length > 0 && grantResults[0] == PackageManager.PERMISSION\_GRANTED) {  
 getLocation();  
 } else {  
 Toast.makeText(this, "Permission Denied!", Toast.LENGTH\_SHORT).show();  
 }  
 }  
 }  
}

Make changes in colors.xml:  
<resources>  
 <color name="purple\_500">#6200EE</color>  
 <color name="purple\_700">#3700B3</color>  
 <color name="teal\_200">#03DAC5</color>  
 <color name="white">#FFFFFF</color>  
 <color name="black">#000000</color>  
</resources>

**Add line in dependencies in build.gradle:**

**implementation("com.google.android.gms:play-services-location:21.0.1"**

**Output:**

A screenshot of a phone

AI-generated content may be incorrect. A screenshot of a phone

AI-generated content may be incorrect.

**Result:**

The Android Application that implements GPS Location Information is developed and executed successfully.