GRACE COLLEGE OF ENGINEERING

(Approved by AICTE, New Delhi & Affiliated to ANNA University, Chennai)

MULLAKKADU, THOOTHUKUDI - 628 005

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING

BE- Computer Science and Engineering

Anna University Regulation: 2017

CS8662- Mobile Application and Development Laboratory

III Year/VI Semester

Lab Manual

Prepared By,

Ms. S. Porkodi, AP/CSE

Grace College Of Engineering ,

Mullakkadu,Tuticorin-628005

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING

CS8662-

Mobile Application Development Laboratory-

Lab Manual

Ex.no:1	**Develop an application that uses GUI components, Font and Colors**
Date:	

Aim:

To develop an android application that uses GUI Components, Font and colors.

Procedure:

1. Create a New Android Project:
- Click New in the toolbar.
- In the window that appears, open the Android folder, select Android Application Project, and click next.
- Provide the application name and the project name and then finally give the desired package name.
- Choose a launcher icon for your application and then select Blank Activity and then click Next
- Provide the desired Activity name for your project and then click Finish.
2. Create a New AVD (Android Virtual Device):
- click Android Virtual Device Manager from the toolbar.
- In the Android Virtual Device Manager panel, click New.
- Fill in the details for the AVD. Give it a name, a platform target, an SD card size, and a skin (HVGA is default).
- Click Create AVD and Select the new AVD from the Android Virtual Device Manager and click Start.
3. Design the graphical layout with a text view and two command buttons.
4. Run the application.
5. On pressing the change color button, color of the text gets changed.
6. On pressing the change font size button, the size of the font gets altered.
7. Close the Android project.

Program:

MainActivity.java:

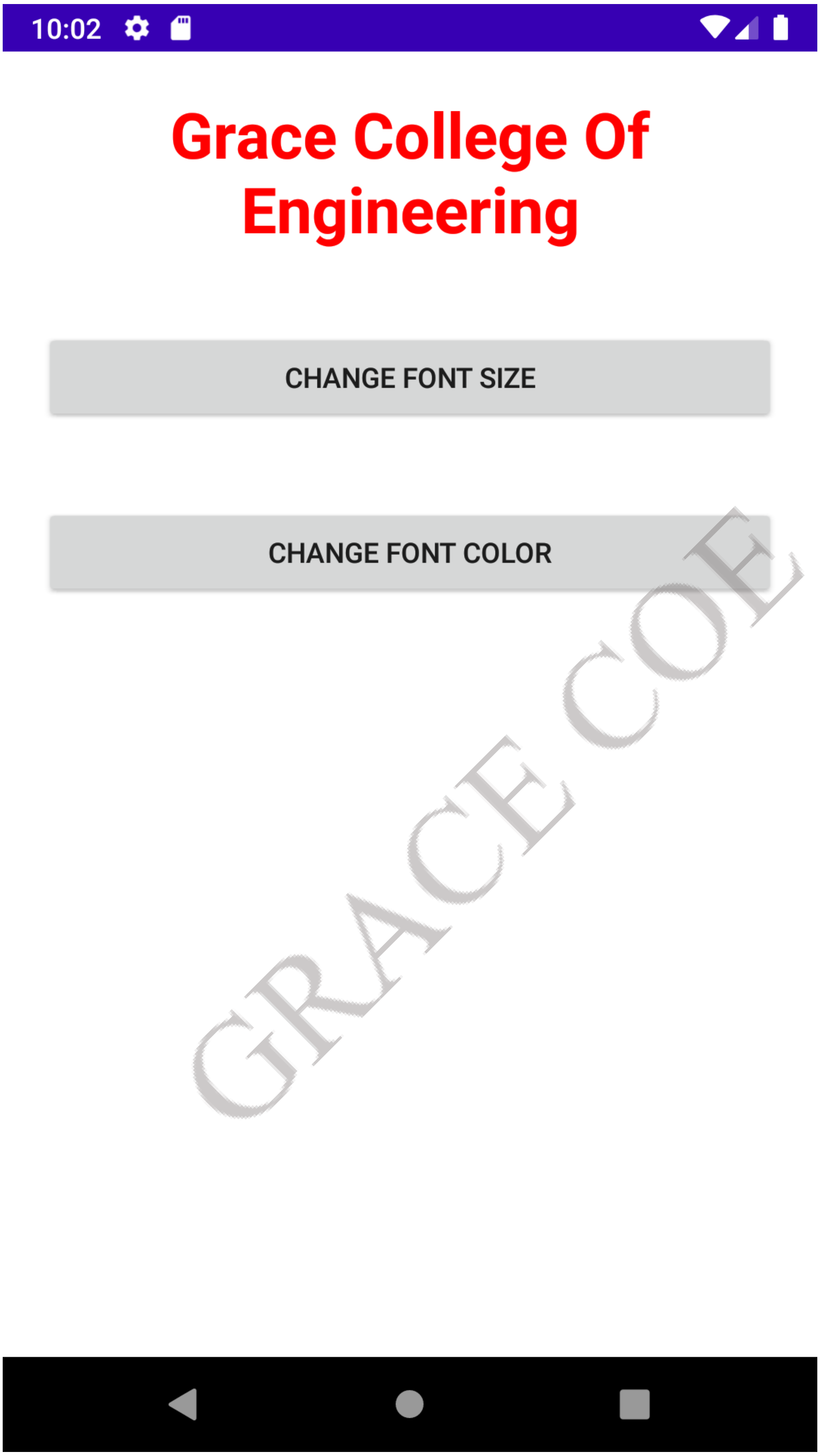
```
import android.app.Activity;
import android.os.Bundle;
import android.graphics.Typeface;
import android.graphics.Color;
import android.view.View;
import android.widget.Button;
import android.widget.TextView;
public class MainActivity extends Activity {
 float font=24;
 int i=1;
 protected void onCreate(Bundle savedInstanceState) {
 super.onCreate(savedInstanceState);
 setContentView(R.layout.activity_main);
 final TextView t1=(TextView) findViewById(R.id.textView);
 Button b1=(Button) findViewById(R.id.button1);
 b1.setOnClickListener(new View.OnClickListener() {
 @Override
 public void onClick(View v) {
 t1.setTextSize(font);
 font+=4;
 if (font==40)
 font=20;
 }
 });
 Button b2=(Button) findViewById(R.id.button2);
 b2.setOnClickListener(new View.OnClickListener() {
 @Override
 public void onClick(View v) {
 switch(i)
 {
 case 1:
 t1.setTextColor(Color.parseColor("#0000FF"));
 break;
 case 2:
 t1.setTextColor(Color.parseColor("#00FF00"));
 break;
 case 3:
 t1.setTextColor(Color.parseColor("#FF0000"));
 break;
 case 4:
 t1.setTextColor(Color.parseColor("#000000"));
 break;
 }
 i++;
 if (i==5)
 i=1;
 } });
 }}
```

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="fill_parent"
    android:layout_height="fill_parent"
    android:orientation="vertical">
    <TextView
        android:id="@+id/textView"
        android:layout_width="match_parent"
        android:layout_height="80sp"
        android:layout_margin="20sp"
        android:gravity="center"
        android:text="Grace College Of Engineering"
        android:textColor="#00FF00"
        android:textSize="36sp"
        android:textStyle="bold" />
    <Button
        android:id="@+id/button1"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:layout_margin="20sp"
        android:gravity="center"
        android:text="Change font size" />
    <Button
        android:id="@+id/button2"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:layout_margin="20sp"
        android:gravity="center"
        android:text="Change font color" />
</LinearLayout>
```

Output:



10:02
Grace College Of Engineering
CHANGE FONT SIZE
CHANGE FONT COLOR

Result:

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

Ex.no:2	**Develop an application that uses Layout Managers and Event Listeners**
Date:	

Aim:

To develop an android application that uses Layout Managers and event listeners.

Procedure:

1. Create a New Android Project:

- Click New in the toolbar.
- In the window that appears, open the Android folder, select Android Application Project, and click next.
- Provide the application name and the project name and then finally give the desired package name.
- Choose a launcher icon for your application and then select Blank Activity and then click Next
- Provide the desired Activity name for your project and then click Finish.

2. Create a New AVD (Android Virtual Device):

- click Android Virtual Device Manager from the toolbar.
- In the Android Virtual Device Manager panel, click New.
- Fill in the details for the AVD. Give it a name, a platform target, an SD card size, and a skin (HVGA is default).
- Click Create AVD and Select the new AVD from the Android Virtual Device Manager and click Start.

3. Design the graphical layout with buttons, edit text and text view.
4. Run the application.
5. Provide the required inputs to perform the desired arithmetic operation.
6. Display the result.
7. Close the Android project

Program:

MainActivity.java

```
package com.example.myapplication;
      import android.os.Bundle;
      import android.app.Activity;
      import android.view.View;
      import android.view.View.OnClickListener;
      import android.widget.Button;
      import android.widget.EditText;
      import android.widget.Toast;
      public class MainActivityextends Activity {
EditTexttxtData1,txtData2;
```

```
        float num1,num2,result1,result2;
        protected void onCreate(Bundle savedInstanceState) {
super.onCreate(savedInstanceState);
setContentView(R.layout.activity_main);
Button add = (Button)findViewById(R.id.button1);
add.setOnClickListener(new OnClickListener(){
public void onClick(View v){
try
{
txtData1 = (EditText)findViewById(R.id.editText1);
txtData2 = (EditText)findViewById(R.id.editText2);
num1 = Float.parseFloat(txtData1.getText().toString());
num2 = Float.parseFloat(txtData2.getText().toString());
result1 = num1+num2;
Toast.makeText(getBaseContext(),"ANSWER:"+result1,Toast.LENGTH_SHORT).show();
}
catch(Exception e)
        {
        Toast.makeText(getBaseContext(),e.getMessage(),Toast.LENGTH_SHORT).show();
}
        }
        });
Button sub = (Button)findViewById(R.id.button3);
sub.setOnClickListener(new OnClickListener(){
public void onClick(View v)
        {
try
{
txtData1 = (EditText)findViewById(R.id.editText1);
txtData2 = (EditText)findViewById(R.id.editText2);
num1 = Float.parseFloat(txtData1.getText().toString());
num2 = Float.parseFloat(txtData2.getText().toString());
result2 = num1-num2;

Toast.makeText(getBaseContext(),"ANSWER:"+result2,Toast.LENGTH_SHORT).show();
}
catch(Exception e)
        {
        Toast.makeText(getBaseContext(),e.getMessage(),Toast.LENGTH_SHORT).show();
}
        }
        });
Button clear = (Button)findViewById(R.id.button2);
clear.setOnClickListener(new OnClickListener() {
public void onClick(View v)
```

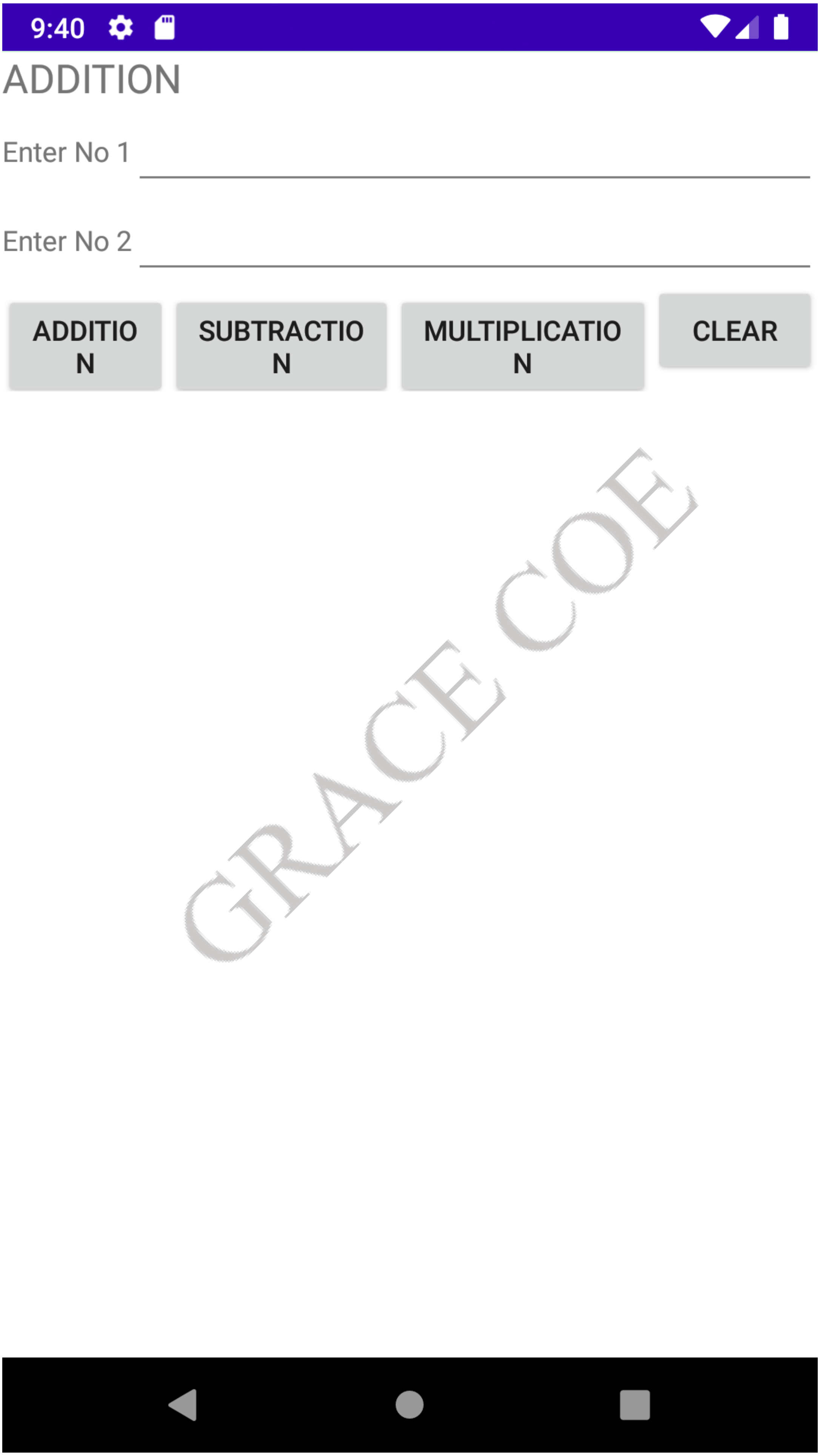
```
        {
  try
  {
  txtData1.setText("");
  txtData2.setText("");
  }
  catch(Exception e)
        {
        Toast.makeText(getBaseContext(),e.getMessage(),Toast.LENGTH_SHORT).show();
  }
        }
        });
  }
        }
```

activity_main.xml:

```
<?xml version="1.0" encoding="utf-8"?>
<RelativeLayout xmlns:android="http://schemas.android.com/apk/res/android"
   android:id="@+id/relativeLayout1"
   android:layout_width="fill_parent"
   android:layout_height="fill_parent">
   <LinearLayout
      android:id="@+id/linearLayout1"
      android:layout_width="wrap_content"
      android:layout_height="wrap_content"
      android:layout_alignParentLeft="true"
      android:layout_alignParentRight="true"
      android:layout_alignParentTop="true">
      <TextView
         android:layout_width="wrap_content"
         android:layout_height="wrap_content"
         android:text="ADDITION"
         android:layout_gravity="center"
         android:textSize="20dp">
      </TextView>
   </LinearLayout>
   <LinearLayout
      android:id="@+id/linearLayout2"
      android:layout_width="wrap_content"
      android:layout_height="wrap_content"
      android:layout_alignParentLeft="true"
      android:layout_alignParentRight="true"
      android:layout_below="@+id/linearLayout1">
      <TextView
         android:layout_width="wrap_content"
         android:layout_height="wrap_content"
      android:text="Enter No 1" />
    <EditText
      android:id="@+id/editText1"
      android:layout_width="wrap_content"
      android:layout_height="wrap_content"
      android:layout_weight="0.20"
      android:inputType="number">
    </EditText>
  </LinearLayout>
  <LinearLayout
    android:id="@+id/linearLayout3"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:layout_alignParentLeft="true"
    android:layout_alignParentRight="true"
    android:layout_below="@+id/linearLayout2">
    <TextView
      android:layout_width="wrap_content"
      android:layout_height="wrap_content"
      android:text="Enter No 2" />
    <EditText
      android:id="@+id/editText2"
      android:layout_width="wrap_content"
      android:layout_height="wrap_content"
      android:layout_weight="0.20"
      android:inputType="number">
    </EditText>
  </LinearLayout>
  <LinearLayout
    android:id="@+id/linearLayout4"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:layout_alignParentLeft="true"
    android:layout_alignParentRight="true"
    android:layout_below="@+id/linearLayout3">
    <Button
      android:id="@+id/button1"
      android:layout_width="wrap_content"
      android:layout_height="wrap_content"
      android:layout_weight="0.50"
      android:text="Addition" />
    <Button
      android:id="@+id/button3"
      android:layout_width="wrap_content"
      android:layout_height="wrap_content"
      android:layout_weight="0.50"
```

```
        android:text="Subtraction" />
    <Button
        android:id="@+id/button4"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_weight="0.50"
        android:text="Multiplication" />
    <Button
        android:id="@+id/button2"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_weight="0.50"
        android:text="Clear" />

  </LinearLayout>
<View
android:id="@+id/linearLayout4"
android:layout_width="fill_parent"
android:layout_height="2px"
android:background="#DDFFDD" />
  </RelativeLayout>
```

Output:



9:40
ADDITION
Enter No 1
Enter No 2
ADDITIO
N
SUBTRACTIO
N
MULTIPLICATIO
N
CLEAR

Result:

Thus, the program for android application that uses layout managers and event listeners was executed successfully.

Ex.no:3	**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 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.
- Then delete the code which is there and type the code as given below.
- Java Coding for the Android Application:
- I)Click on app -> java -> com.example.exno3 -> MainActivity.

Program:

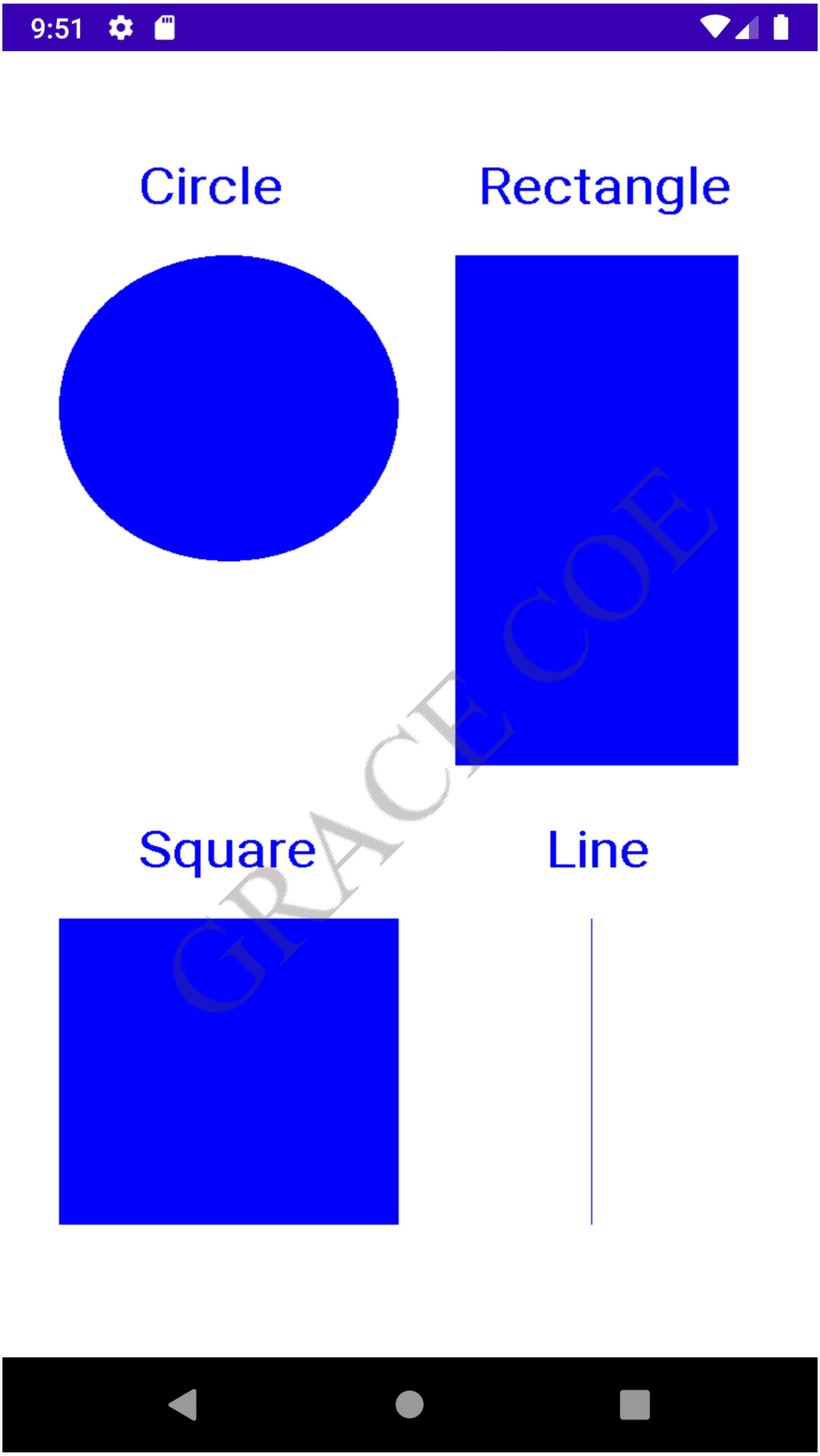
activity_main.xml:

```
<?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.ex3;
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);
 //Creating a Bitmap
 Bitmap bg = Bitmap.createBitmap(720, 1280, Bitmap.Config.ARGB_8888);
 //Setting the Bitmap as background for the ImageView
 ImageView i = (ImageView) findViewById(R.id.imageView);
 i.setBackgroundDrawable(new BitmapDrawable(bg));
 //Creating the Canvas Object
 Canvas canvas = new Canvas(bg);
 //Creating the Paint Object and set its color & TextSize
 Paint paint = new Paint();
 paint.setColor(Color.BLUE);
 paint.setTextSize(50);
 //To draw a Rectangle
 canvas.drawText("Rectangle", 420, 150, paint);
 canvas.drawRect(400, 200, 650, 700, paint);
 //To draw a Circle
 canvas.drawText("Circle", 120, 150, paint);
 canvas.drawCircle(200, 350, 150, paint);
17
 //To draw a Square
 canvas.drawText("Square", 120, 800, paint);
 canvas.drawRect(50, 850, 350, 1150, paint);
 //To draw a  Line
 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:4	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 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:
  i) Click on app -> res -> layout -> activity_main.xml.
- Java Coding for the Android Application:
  i)Click on app -> java -> com.example.exno4 -> MainActivity.

Program:

MainActivity.java:

```
package com.example.ex4;
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 MainActivityextends Activity implements OnClickListener
{
EditTextRollno,Name,Marks;
Button Insert,Delete,Update,View,ViewAll;
SQLiteDatabasedb;
/** Called when the activity is first created. */
@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);
// Creating database and table
db=openOrCreateDatabase("StudentDB", Context.MODE_PRIVATE, null);
db.execSQL("CREATE TABLE IF NOT EXISTS student(rollnoVARCHAR,nameVARCHAR,marks
VARCHAR);");
}
public void onClick(View view)
   {
// Inserting a record to the Student table
if(view==Insert)
      {
// Checking for empty fields
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();
}
// Deleting a record from the Student table
if(view==Delete)
      {
// Checking for empty roll number
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();
}
// Updating a record in the Student table
if(view==Update)
     {
// Checking for empty roll number
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();
}
// Display a record from the Student table
if(view==View)
     {
// Checking for empty roll number
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();
}
      }
// Displaying all the records
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();
}
}
```

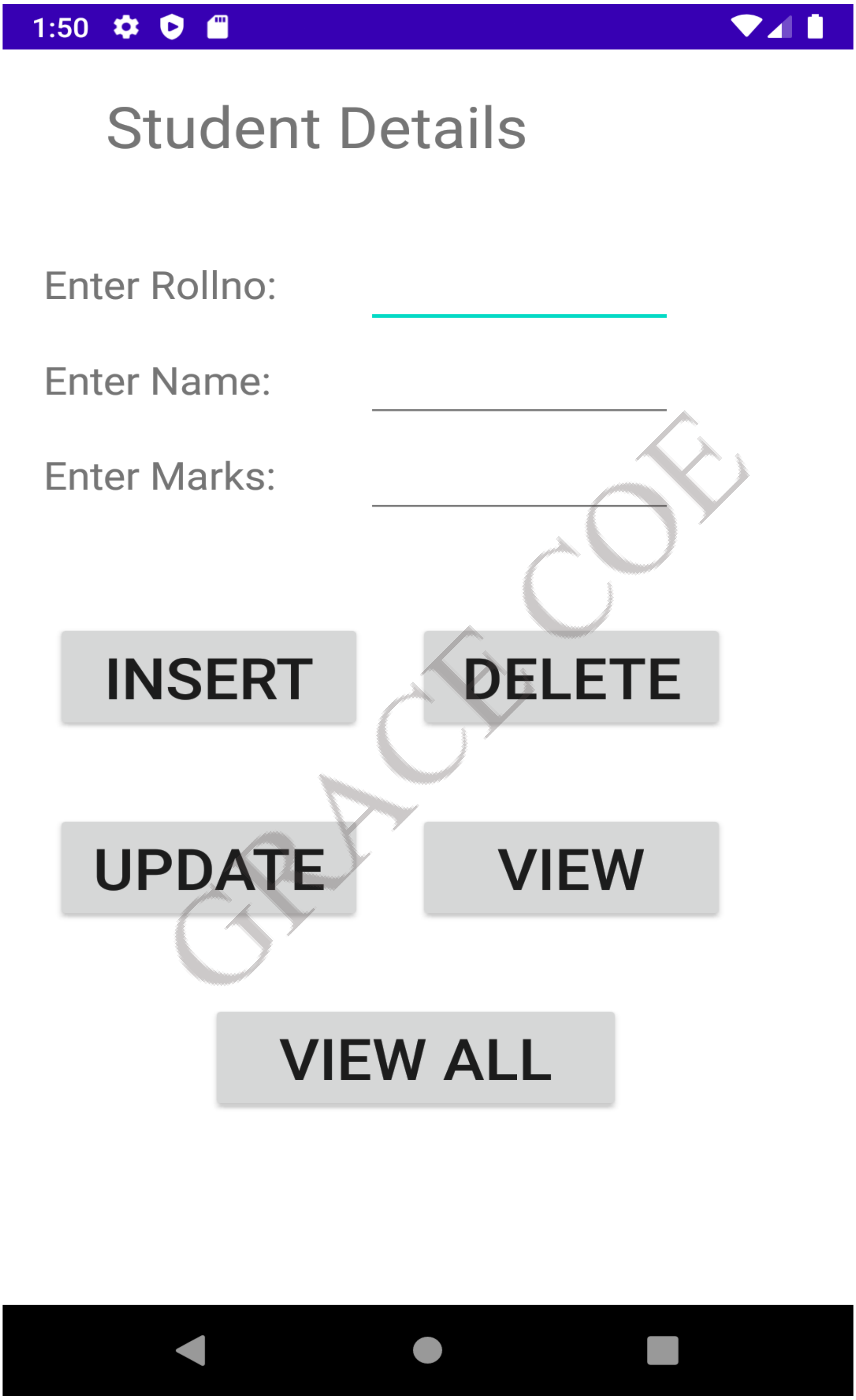
activity_main.xml:

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

Output:



Student Details

Enter Rollno:

Enter Name:

Enter Marks:

Result:

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

Ex.no:5	Develop an application that makes use of Notification Manager
Date:	

AIM:
To develop a Android Application that that makes use of Notification Manager.

ALGORITHM:

1. Create a New Android Project:
• Click New in the toolbar.
• In the window that appears, open the Android folder, select Android Application Project, and click next.
• Provide the application name and the project name and then finally give the desired package name.
• Choose a launcher icon for your application and then select Blank Activity and then click Next.
• Provide the desired Activity name for your project and then click Finish.
2. Create a New AVD (Android Virtual Device):
• click Android Virtual Device Manager from the toolbar.
• In the Android Virtual Device Manager panel, click New.
• Fill in the details for the AVD. Give it a name, a platform target, an SD card size, and a skin (HVGA is default).
• Click Create AVD and Select the new AVD from the Android Virtual Device Manager and click Start.
3. Design the graphical layout using buttons, text and ImageView.
4. Creating Second Activity for the Android Application:
5. Click on File -> New -> Activity -> Empty Activity.
6. Type the Activity Name as Second Activity and click Finish button.
7. Run the application.
8.Add this code in the AndroidManifest.xml

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

9. Display the output by clicking the Notify button.
10. Close the Android project.

PROGRAM:

MainActivity.java:

```
package com.example.myapplication;
import android.app.Notification;
import android.app.NotificationChannel;
import android.app.NotificationManager;
import android.app.PendingIntent;
import android.content.Intent;
import android.os.Bundle;
import androidx.appcompat.app.AppCompatActivity;
import android.view.View;
import android.widget.Button;
import android.widget.EditText;

import androidx.appcompat.app.AppCompatActivity;
// import statement for NotificationCompat
import androidx.core.app.NotificationCompat;

public class MainActivity extends AppCompatActivity {
  Button notify;
  EditText e;

  @Override
  protected void onCreate(Bundle savedInstanceState) {
    super.onCreate(savedInstanceState);
    setContentView(R.layout.activity_main);
    notify = (Button) findViewById(R.id.button);
    e = (EditText) findViewById(R.id.editText);
    notify.setOnClickListener(new View.OnClickListener() {
      @Override
      public void onClick(View v) {
        Intent intent = new Intent(MainActivity.this, Main2Activity.class);
        PendingIntent pending = PendingIntent.getActivity(MainActivity.this, 0, intent, 0);

        // Create the notification builder
        NotificationCompat.Builder builder = new
NotificationCompat.Builder(MainActivity.this, "channel_id")
            .setSmallIcon(R.mipmap.ic_launcher)
            .setContentTitle("New Message")
            .setContentText(e.getText().toString())
            .setContentIntent(pending)
            .setAutoCancel(true);

        // Get the NotificationManager service
        NotificationManager notificationManager = (NotificationManager)
getSystemService(NOTIFICATION_SERVICE);

        // Create a channel for the notification (for Android O and above)
        if (android.os.Build.VERSION.SDK_INT >= android.os.Build.VERSION_CODES.O) {
          NotificationChannel channel = new NotificationChannel("channel_id", "Channel
Name", NotificationManager.IMPORTANCE_HIGH);
```

```
                notificationManager.createNotificationChannel(channel);
            }

            // Show the notification
            notificationManager.notify(0, builder.build());
          }
       });
    }
}
```

Main2Activity.java:

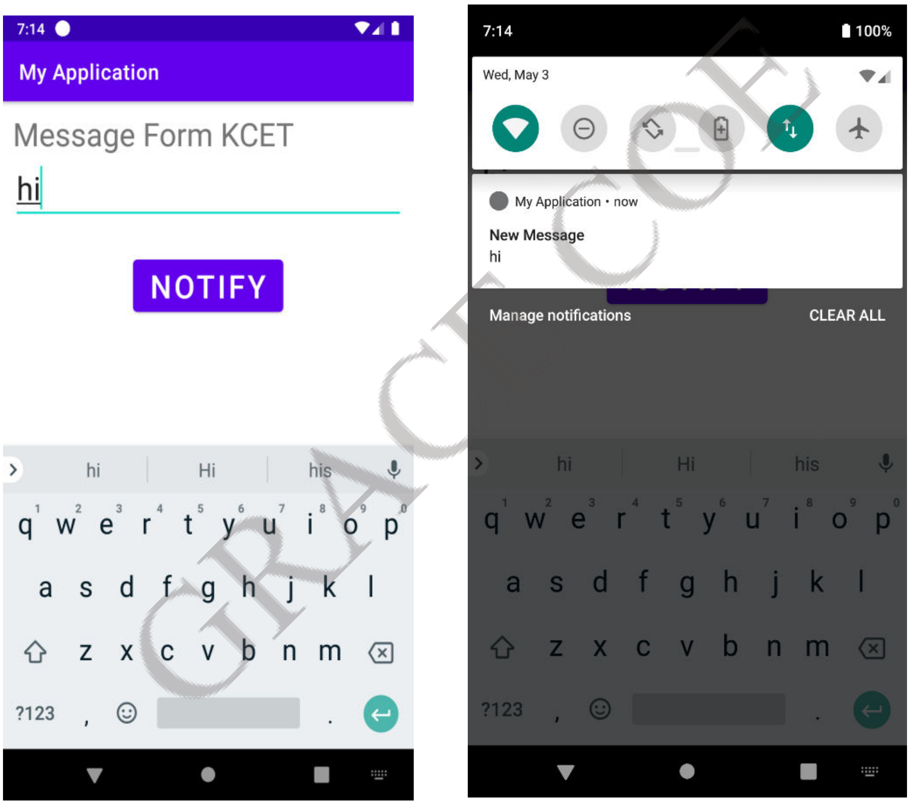
```
package com.example.myapplication;
import androidx.appcompat.app.AppCompatActivity;
import android.os.Bundle;
public class Main2Activity extends AppCompatActivity {
   @Override
   protected void onCreate(Bundle savedInstanceState) {
      super.onCreate(savedInstanceState);
      setContentView(R.layout.activity_main2);

   }
}
```

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:layout_margin="10dp"
   android:orientation="vertical">
<TextView
android:layout_width="wrap_content"
android:layout_height="wrap_content"
   android:text="Message Form KCET"
android:textSize="30sp" />
<EditText
android:id="@+id/editText"
android:layout_width="match_parent"
android:layout_height="wrap_content"
android:textSize="30sp" />
<Button
android:id="@+id/button"
```

```
android:layout_width="wrap_content"
android:layout_height="wrap_content"
android:layout_margin="30dp"
android:layout_gravity="center"
android:text="Notify"
android:textSize="30sp"/>
   </LinearLayout>
```

Output:



Message Form KCET

hi

NOTIFY

My Application • now

New Message

hi

Result:
Thus Android Application that that makes use of Notification Manager is developed and executed successfully.

Ex.no:6	Implement an application that implements Multithreading
Date:	

AIM:

To develop an android application that implements multithreading.

ALGORITHM:

1. Create a New Android Project:
- Click New in the toolbar.
- In the window that appears, open the Android folder, select Android Application Project, and click next.
- Provide the application name and the project name and then finally give the desired package name.
- Choose a launcher icon for your application and then select Blank Activity and then click Next
- Provide the desired Activity name for your project and then click Finish.

2. Create a New AVD (Android Virtual Device):
- click Android Virtual Device Manager from the toolbar.
- In the Android Virtual Device Manager panel, click New.
- Fill in the details for the AVD. Give it a name, a platform target, an SD card size, and a skin (HVGA is default).
- Click Create AVD and Select the new AVD from the Android Virtual Device Manager and click Start.

3. Design the graphical layout.
4. Run the application.
5. The requested data is retrieved from the database named myFriendsDb.
6. Close the Android project

PROGRAM:

MainActivity.java:

```
package com.example.myapplication;
import android.os.Bundle;
import androidx.appcompat.app.AppCompatActivity;
import android.view.View;
import android.widget.Button;
import android.widget.ImageView;
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)
```

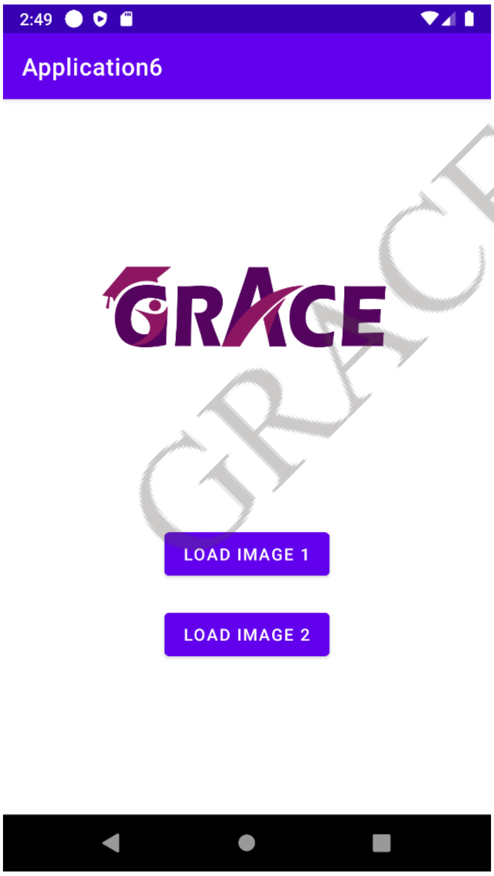
```
        {
            new Thread(new Runnable()

            {
                @Override
                public void run()

                {
                    img.post(new Runnable()

                    {
                        @Override
                        public void run()

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

                        }
                    });

                }
            }).start();

        }
    });

  }
}
```

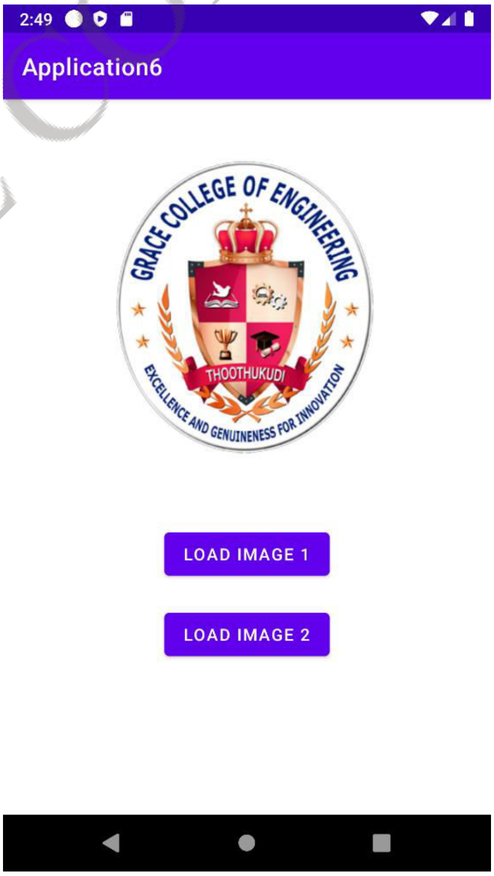
activity_main.xml:

```
<?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>
```

Output:




2:49
Application6
GRACE COLLEGE OF ENGINEERING
THOOTHUKUDI
EXCELLENCE AND GENUINENESS FOR INNOVATION
LOAD IMAGE 1
LOAD IMAGE 2

RESULT:

Thus, the program for android application that makes use of multithreading was executed successfully.

Ex.no:7	**Develop a native application that uses GPS location information**

AIM:

To develop an android application that uses GPS location information.

ALGORITHM:

1. Create a New Android Project:
- Click New in the toolbar.
- In the window that appears, open the Android folder, select Android Application Project, and click next.
- Provide the application name and the project name and then finally give the desired package name.
- Choose a launcher icon for your application and then select Blank Activity and then click Next
- Provide the desired Activity name for your project and then click Finish.

2. Create a New AVD (Android Virtual Device):
- click Android Virtual Device Manager from the toolbar.
- In the Android Virtual Device Manager panel, click New.
- Fill in the details for the AVD. Give it a name, a platform target, an SD card size, and a skin (HVGA is default).
- Click Create AVD and Select the new AVD from the Android Virtual Device Manager and click Start.

3. Design the graphical layout.
4. Run the application.
5. The requested data is retrieved from the database named myFriendsDb.
6. Close the Android project.

PROGRAM CODE:

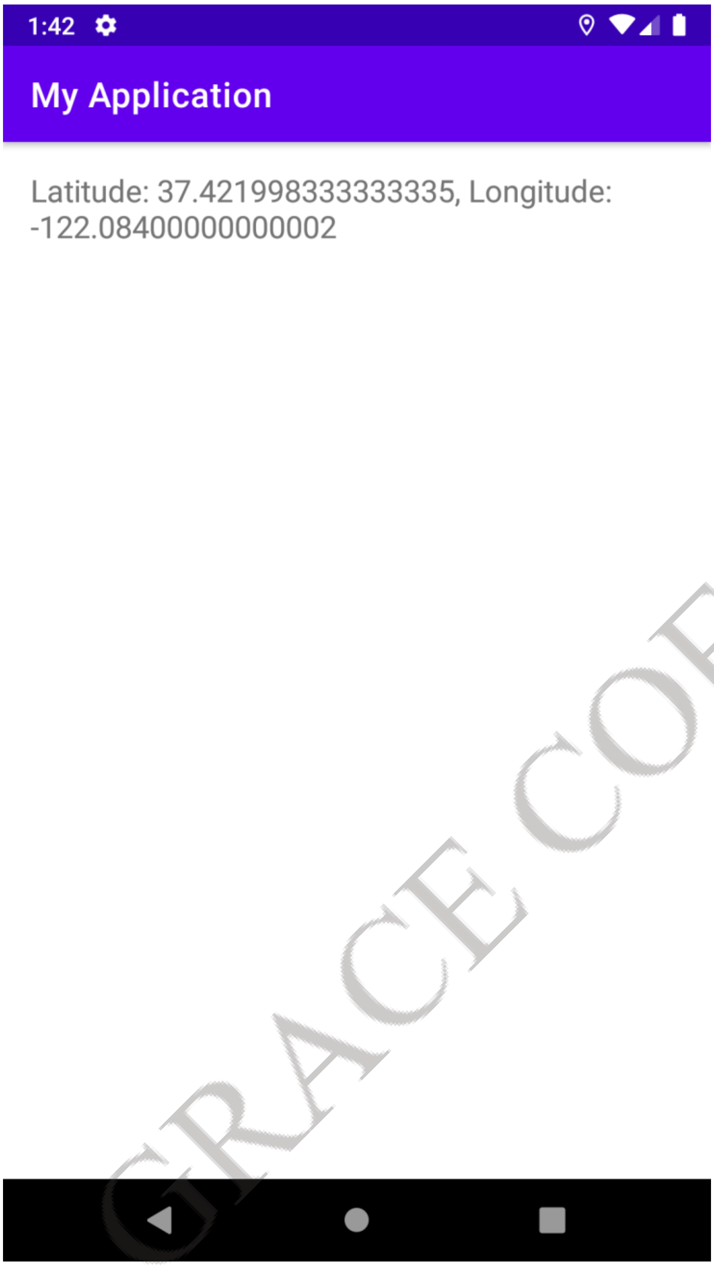
MainActivity.java:

```
package com.example.myapplication;

import android.Manifest;
import android.annotation.SuppressLint;
import android.content.pm.PackageManager;
import android.location.Location;
import android.location.LocationListener;
import android.location.LocationManager;
import android.os.Bundle;
import android.widget.TextView;

import androidx.appcompat.app.AppCompatActivity;
import androidx.core.app.ActivityCompat;
import androidx.core.content.ContextCompat;

public class MainActivity extends AppCompatActivity implements LocationListener {

    private LocationManager locationManager;
    private TextView locationTextView;

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

        // Get a reference to the TextView that will display the location
        locationTextView = findViewById(R.id.locationTextView);

        // Check if we have permission to access location data
        if (ContextCompat.checkSelfPermission(this,
Manifest.permission.ACCESS_FINE_LOCATION) !=
PackageManager.PERMISSION_GRANTED) {
            ActivityCompat.requestPermissions(this, new
String[]{Manifest.permission.ACCESS_FINE_LOCATION}, 1);
        }

        // Get a reference to the location manager
        locationManager = (LocationManager) getSystemService(LOCATION_SERVICE);
    }

    @Override
    protected void onResume() {
        super.onResume();

        // Request location updates from the location manager
        if (ContextCompat.checkSelfPermission(this,
Manifest.permission.ACCESS_FINE_LOCATION) ==
PackageManager.PERMISSION_GRANTED) {
            locationManager.requestLocationUpdates(LocationManager.GPS_PROVIDER, 0, 0,
this);
        }
    }

    @Override
    protected void onPause() {
        super.onPause();
        // Stop receiving location updates when the activity is paused
        locationManager.removeUpdates(this);
    }

    @Override
    public void onLocationChanged(Location location) {
        // Update the TextView with the new location
        locationTextView.setText("Latitude: " + location.getLatitude() + ", Longitude: " +
location.getLongitude());
    }

    @Override
    public void onStatusChanged(String provider, int status, Bundle extras) {}

    @Override
    public void onProviderEnabled(String provider) {}

    @Override
    public void onProviderDisabled(String provider) {}
}
```

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:padding="16dp"
    tools:context=".MainActivity">

    <TextView
        android:id="@+id/locationTextView"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:textSize="18sp" />

</LinearLayout>
```

OUTPUT:



Latitude: 37.421998333333335, Longitude: -122.08400000000002

RESULT:

Thus, the program for android application that makes use of GPS information was executed successfully.

Ex.no:8	**Implement an application that writes data to the SD card**

AIM:

To develop an android application that writes data to the SD card.

ALGORITHM:

1. Create a New Android Project:

• Click New in the toolbar.

• In the window that appears, open the Android folder, select Android Application Project, and click next.

• Provide the application name and the project name and then finally give the desired package name.

• Choose a launcher icon for your application and then select Blank Activity and then click Next

• Provide the desired Activity name for your project and then click Finish.

2. Create a New AVD (Android Virtual Device):

• click Android Virtual Device Manager from the toolbar.

• In the Android Virtual Device Manager panel, click New.

• Fill in the details for the AVD. Give it a name, a platform target, an SD card size, and a skin (HVGA is default).

• Click Create AVD and Select the new AVD from the Android Virtual Device Manager and click Start.

3. Design the graphical layout using buttons, Textview.

4. Run the application.

5. Writes the data into a file in the SD card.

6. Display the output by clicking the View button.

7. Close the Android project.

PROGRAM CODE:

MainActivity.java:

```
package com.example.myapplication;

import android.annotation.SuppressLint;
import android.app.Activity;
import android.os.Bundle;
import android.os.Environment;
import android.view.View;
import android.view.View.OnClickListener;
import android.widget.Button;
import android.widget.EditText;
import android.widget.TextView;
import android.widget.Toast;

import java.io.BufferedReader;
import java.io.File;
import java.io.FileInputStream;
import java.io.FileOutputStream;
import java.io.IOException;
import java.io.InputStreamReader;

public class MainActivity extends Activity {

    EditText inputText;
    TextView response;
    Button saveButton, readButton;
    private String filename = "SampleFile.txt";
    private String filepath = "MyFileStorage";
    File myExternalFile;
    String myData = "";

    @SuppressLint("MissingInflatedId")
    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_main);
        inputText = findViewById(R.id.myInputText);
        response = findViewById(R.id.response);
        saveButton = findViewById(R.id.savExternalStorage);
        readButton = findViewById(R.id.getExternalStorage);

        // Check if saveButton and readButton are null before setting onClickListener
        if (saveButton != null) {
          saveButton.setOnClickListener(new OnClickListener() {
            @Override
            public void onClick(View v) {
              try {
                // Get external storage directory
                File externalStorageDir = Environment.getExternalStorageDirectory();

                // Create file object for the file in the directory
                File file = new File(externalStorageDir, filename);

                // Write input text to file
                FileOutputStream fos = new FileOutputStream(file);
                fos.write(inputText.getText().toString().getBytes());
                fos.close();

                // Clear input text and set response text
                inputText.setText("");
                response.setText("SampleFile.txt saved to External Storage...");
              } catch (IOException e) {
                e.printStackTrace();
              }
            }
          });
        }

        // Check if readButton is null before setting onClickListener
    if (readButton != null) {
        readButton.setOnClickListener(new OnClickListener() {
            @Override
            public void onClick(View v) {
                try {
                    // Get external storage directory
                    File externalStorageDir = Environment.getExternalStorageDirectory();

                    // Create file object for the file in the directory
                    File file = new File(externalStorageDir, filename);

                    // Read data from file
                    FileInputStream fis = new FileInputStream(file);
                    BufferedReader br = new BufferedReader(new InputStreamReader(fis));
                    String strLine;
                    while ((strLine = br.readLine()) != null) {
                        myData = myData + strLine;
                    }
                    br.close();

                    // Set input text and response text
                    inputText.setText(myData);
                    response.setText("SampleFile.txt data retrieved from External Storage...");
                    myData = ""; // reset myData
                } catch (IOException e) {
                    e.printStackTrace();
                    Toast.makeText(MainActivity.this, "Error: File not found",
Toast.LENGTH_SHORT).show();
                }
            }
        });
    }
```

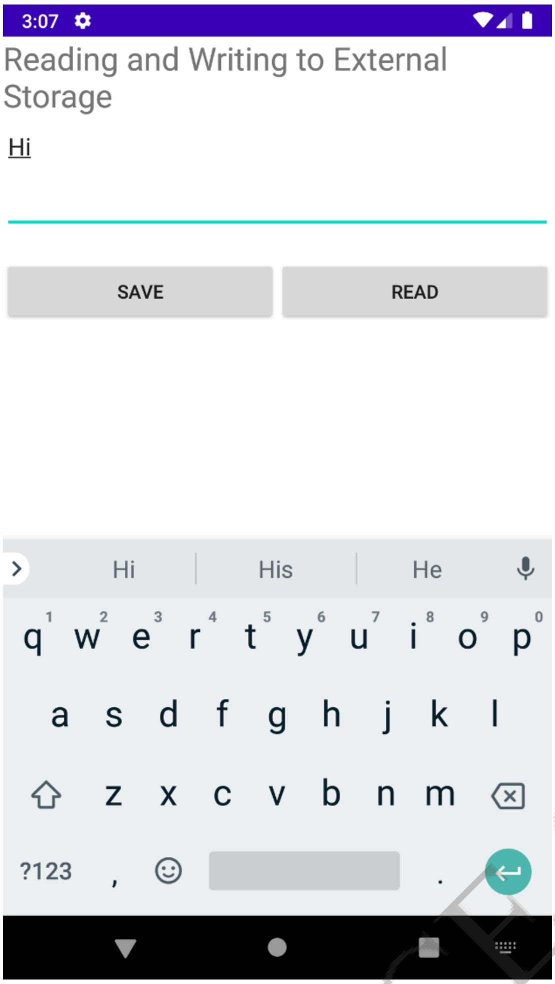
```
        // Check if external storage is available and writable
        if (!isExternalStorageWritable()) {
            saveButton.setEnabled(false);
        }
    }

    /**
     * Returns true if external storage is writable, false otherwise.
     */
    private boolean isExternalStorageWritable() {
        String state = Environment.getExternalStorageState();
        if (Environment.MEDIA_MOUNTED.equals(state)) {
            return true;
        }
        return false;
    }
}
```

activity_main.xml:

```
<?xml version="1.0" encoding="utf-8"?>
<LinearLayout xmlns:android="http://schemas.android.com/apk/res/android"
    android:layout_width="fill_parent" android:layout_height="fill_parent"
    android:orientation="vertical">
<TextView android:layout_width="fill_parent"
    android:layout_height="wrap_content"
    android:text="Reading and Writing to External Storage"
    android:textSize="24sp"/>
<EditText android:id="@+id/myInputText"
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:ems="10" android:lines="5"
    android:minLines="3" android:gravity="top|left"
    android:inputType="textMultiLine">
```

```
<requestFocus/>
   </EditText>
<LinearLayout
android:layout_width="match_parent" android:layout_height="wrap_content"
android:orientation="horizontal"
android:weightSum="1.0"
android:layout_marginTop="20dp">
<Button android:id="@+id/saveExternalStorage"
   android:layout_width="match_parent"
   android:layout_height="wrap_content"
   android:text="SAVE"
   android:layout_weight="0.5"/>
<Button android:id="@+id/getExternalStorage"
   android:layout_width="match_parent"
   android:layout_height="wrap_content"
   android:layout_weight="0.5"
   android:text="READ" />
</LinearLayout>
<TextView android:id="@+id/response"
   android:layout_width="wrap_content"
   android:layout_height="wrap_content" android:padding="5dp"
   android:text=""
   android:textAppearance="?android:attr/textAppearanceMedium" />
   </LinearLayout>
```

Output:



RESULT:

Thus, the program for android application that writes data into the SD Card was executed successfully.

Ex.No:11	**Develop a mobile application to send an email.**

AIM:

To develop an android application that send an email.

ALGORITHM:

1. Create a New Android Project:

• Click New in the toolbar.

• In the window that appears, open the Android folder, select Android Application Project, and click next.

• Provide the application name and the project name and then finally give the desired package name.

• Choose a launcher icon for your application and then select Blank Activity and then click Next

• Provide the desired Activity name for your project and then click Finish.

2. Create a New AVD (Android Virtual Device):

• click Android Virtual Device Manager from the toolbar.

• In the Android Virtual Device Manager panel, click New.

• Fill in the details for the AVD. Give it a name, a platform target, an SD card size, and a skin (HVGA is default).

• Click Create AVD and Select the new AVD from the Android Virtual Device Manager and click Start.

3. Design the graphical layout.

4. Run the application.

5. When the application starts alarm sound will be invoked.

6. Stop alarm button is clicked to stop the alarm.

7. Close the Android project.

PROGRAM CODE:

MainActivity.java:

```
package com.example.myapplication;

import android.content.Intent;
import android.net.Uri;
import android.os.Bundle;
import androidx.appcompat.app.AppCompatActivity;
import android.util.Log;
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 startBtn = (Button) findViewById(R.id.sendbttn);
      startBtn.setOnClickListener(new View.OnClickListener() {
         public void onClick(View view) {
            sendEmail();
         }
      });
   }
   protected void sendEmail() {
      Log.i("Send email", "");
      String[] TO = {
            "muthuramalingam566@gmail.com"
      };
      String[] CC = {
            "ramdurai25@gmail.com"
      };
      Intent emailIntent = new Intent(Intent.ACTION_SEND);
      emailIntent.setData(Uri.parse("mailto:"));
      emailIntent.setType("text/plain");
      emailIntent.putExtra(Intent.EXTRA_EMAIL, TO);
      emailIntent.putExtra(Intent.EXTRA_CC, CC);
      emailIntent.putExtra(Intent.EXTRA_SUBJECT, "Your subject");
```

```
        emailIntent.putExtra(Intent.EXTRA_TEXT, "Email message goes here");
        try {
           startActivity(Intent.createChooser(emailIntent, "Send mail..."));
           finish();
           Log.i("Finished sending email...", "");
        } catch (android.content.ActivityNotFoundException ex) {
           Toast.makeText(MainActivity.this, "There is no email client
installed.",Toast.LENGTH_SHORT).show();
        }
    }
}
```

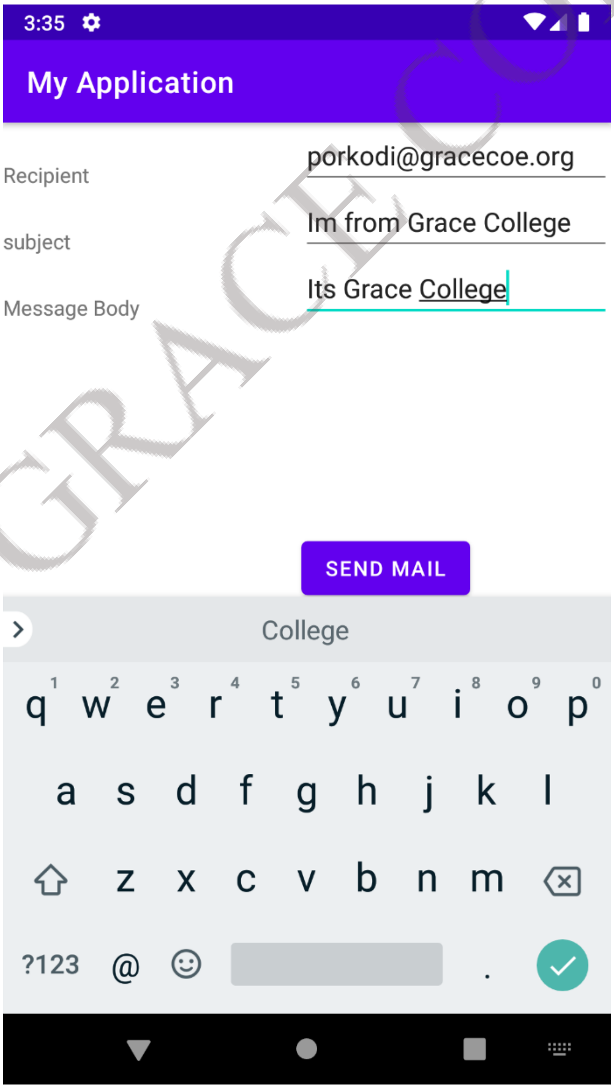
activity_main.xml:

```
<?xml version="1.0" encoding="utf-8"?>
<RelativeLayout xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:tools="http://schemas.android.com/tools"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    android:paddingBottom="@dimen/activity_vertical_margin"
    android:paddingLeft="@dimen/activity_horizontal_margin"
    android:paddingRight="@dimen/activity_horizontal_margin"
    android:paddingTop="@dimen/activity_vertical_margin"
    tools:context="com.example.myapplication.MainActivity">
<EditText android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:inputType="textEmailAddress"
    android:ems="10"
    android:id="@+id/editText"
    android:layout_alignParentTop="true"
    android:layout_alignParentRight="true"
    android:layout_alignParentEnd="true" /> <EditText
android:layout_width="wrap_content"
android:layout_height="wrap_content"
```

```
android:inputType="textEmailAddress"
android:ems="10"
android:id="@+id/editText2"
android:layout_below="@+id/editText"
android:layout_alignRight="@+id/editText"
android:layout_alignEnd="@+id/editText" /> <EditText
android:layout_width="wrap_content"
android:layout_height="wrap_content"
android:inputType="textEmailAddress"
android:ems="10"
android:id="@+id/editText3"
android:layout_below="@+id/editText2"
android:layout_alignRight="@+id/editText2"
android:layout_alignEnd="@+id/editText2" /> <Button
android:layout_width="wrap_content"
android:layout_height="wrap_content"
android:text="SEND MAIL"
android:id="@+id/sendbttn"
android:layout_centerVertical="true"
android:layout_alignLeft="@+id/editText3"
android:layout_alignStart="@+id/editText3" /> <TextView
android:layout_width="wrap_content"
android:layout_height="wrap_content"
android:text="Recipient"
android:id="@+id/textView"
android:layout_alignBottom="@+id/editText"
android:layout_alignParentLeft="true"
android:layout_alignParentStart="true" /> <TextView
android:layout_width="wrap_content"
android:layout_height="wrap_content"
android:text="subject"
android:id="@+id/textView2"
android:layout_alignBottom="@+id/editText2"
```

```
android:layout_alignParentLeft="true"
android:layout_alignParentStart="true" /> <TextView
android:layout_width="wrap_content"
android:layout_height="wrap_content"
android:text="Message Body"
android:id="@+id/textView3"
android:layout_alignBottom="@+id/editText3"
android:layout_alignParentLeft="true"
android:layout_alignParentStart="true" />
   </RelativeLayout
      >
```

Output:



Recipient porkodi@gracecoe.org

subject Im from Grace College

Message Body Its Grace College

RESULT:

Thus, the program for android application to send an email was executed successfully.