MOBILE APPLICATION DEVELOPMENT

LABORATORY EXERCISES (1 - 6)

Ex No: 1. Android Application that uses GUI components, Font and Colors

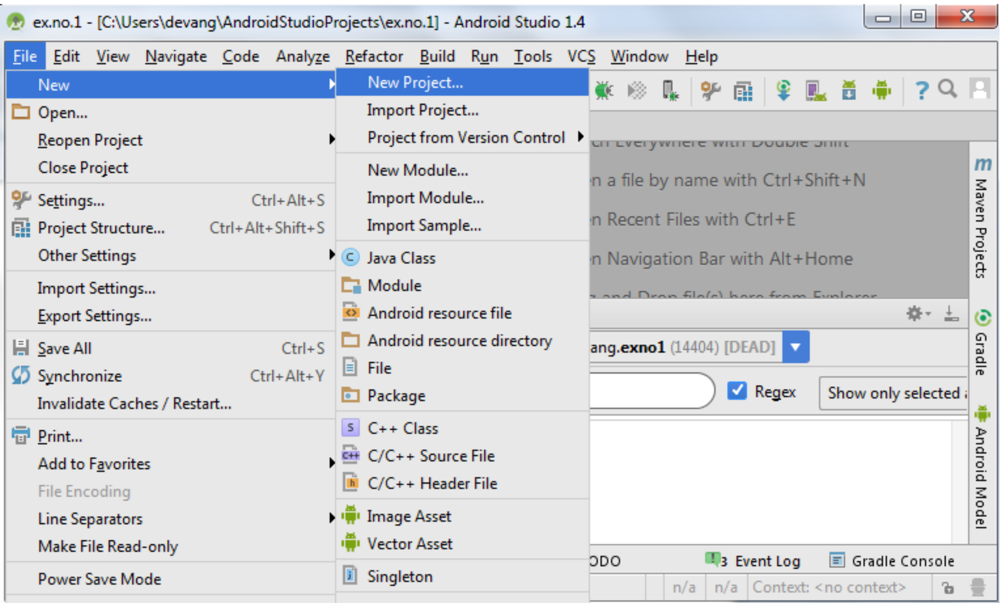
Aim:

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

Procedure:

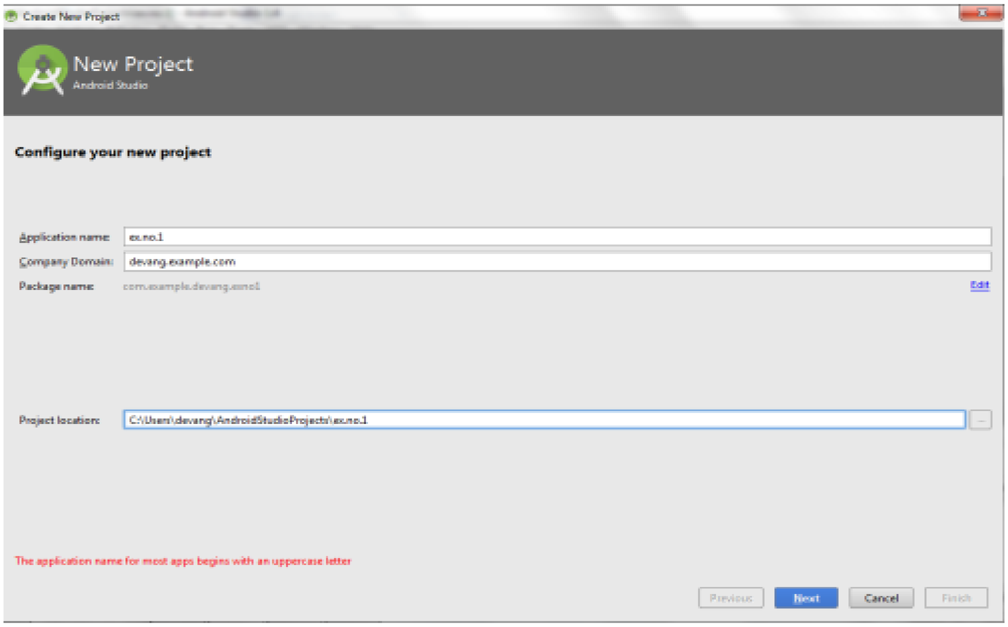
Creating a New project:

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



ex.no.1 - [C:\Users\devang\AndroidStudioProjects\ex.no.1] - Android Studio 1.4

- Then type the Application name as "**ex.no.1"** and click **Next.**



Create New Project

New Project

Android Studio

Configure your new project

Application name: ex.no.1

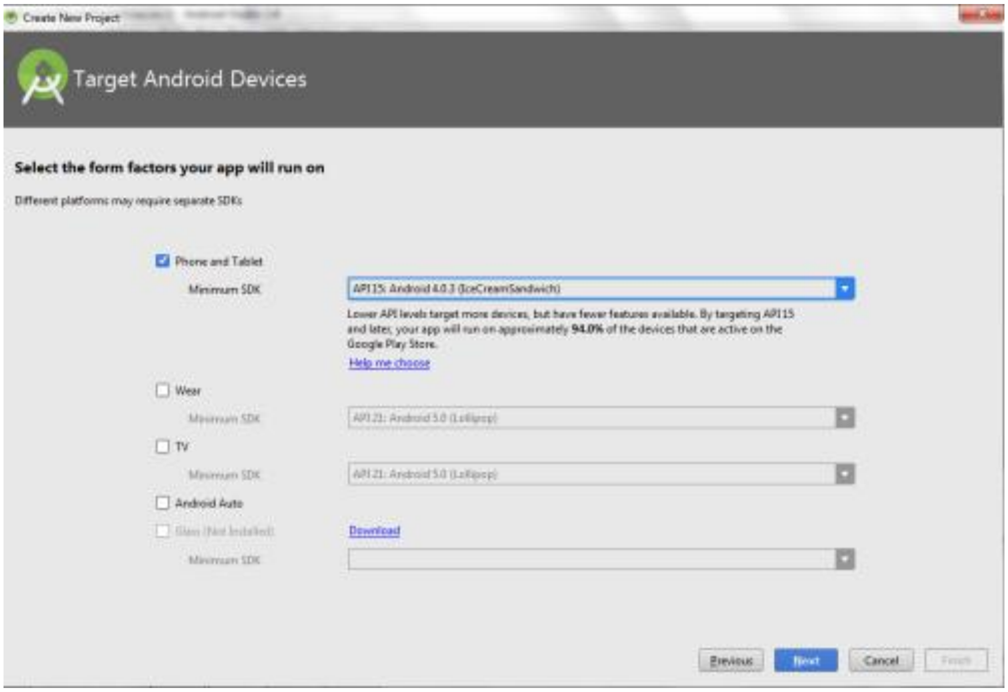
Company Domain: devang.example.com

Package name: com.example.devang.exno1

Project location: C:\Users\devang\AndroidStudioProjects\ex.no.1

The application name for most apps begins with an uppercase letter

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



Create New Project

Target Android Devices

Select the form factors your app will run on

Different platforms may require separate SDKs

☑ Phone and Tablet

Minimum SDK: API15: Android 4.0.3 (IceCreamSandwich)

Lower API levels target more devices, but have fewer features available. By targeting API15 and later, your app will run on approximately **94.0%** of the devices that are active on the Google Play Store.

Help me choose

☐ Wear

Minimum SDK: API 21: Android 5.0 (Lollipop)

☐ TV

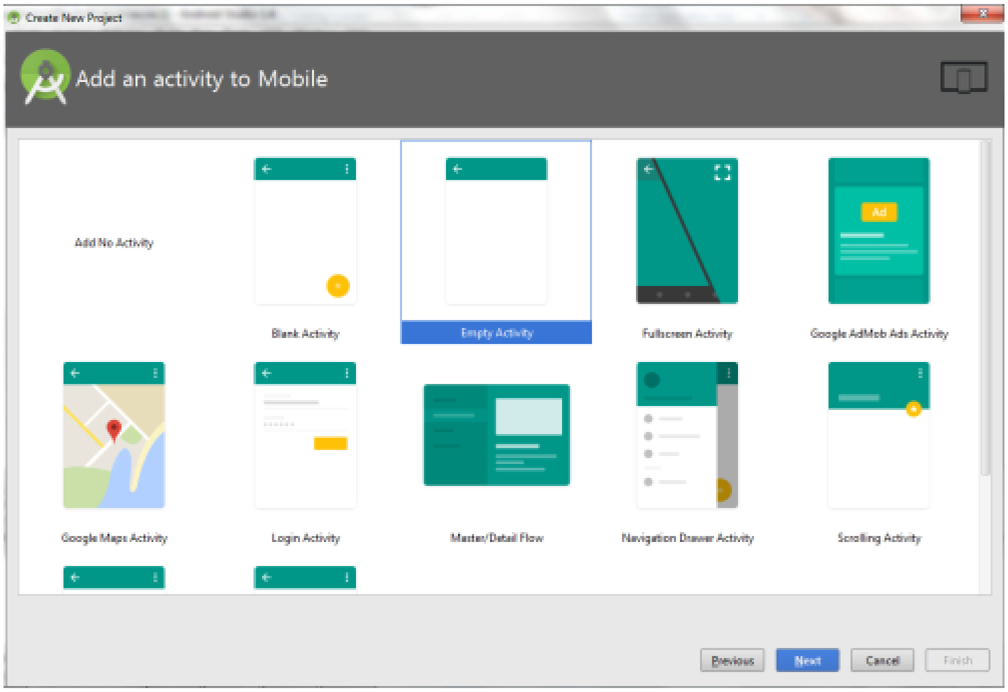
Minimum SDK: API 21: Android 5.0 (Lollipop)

☐ Android Auto

☐ Glass (Not Installed) Download

Minimum SDK

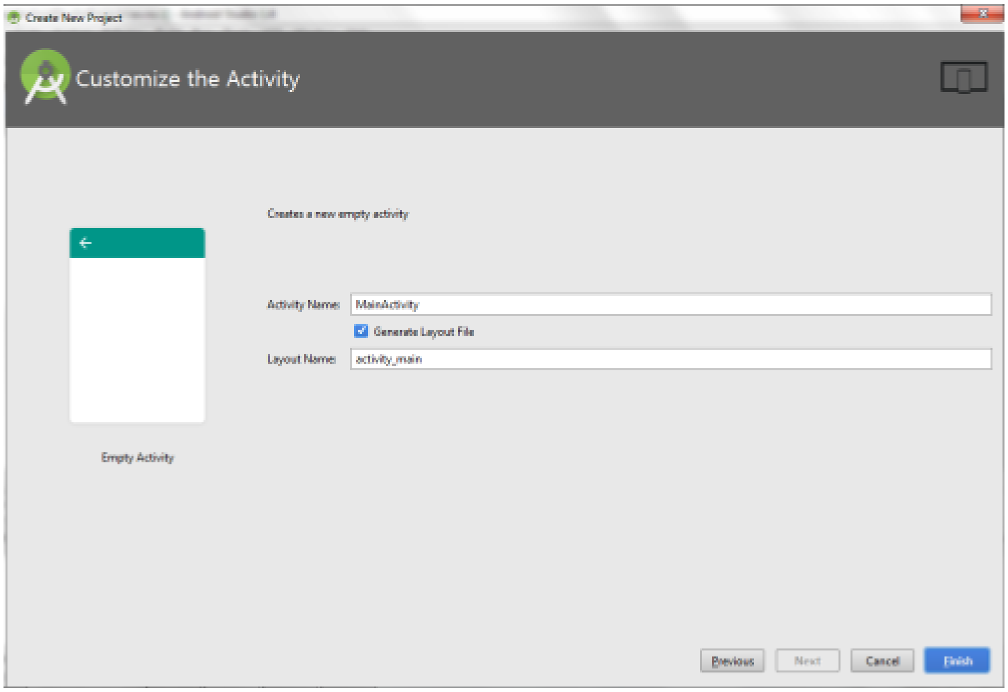
- Then select the **Empty Activity** and click **Next.**



Create New Project

Add an activity to Mobile

- Finally click F**inish**.



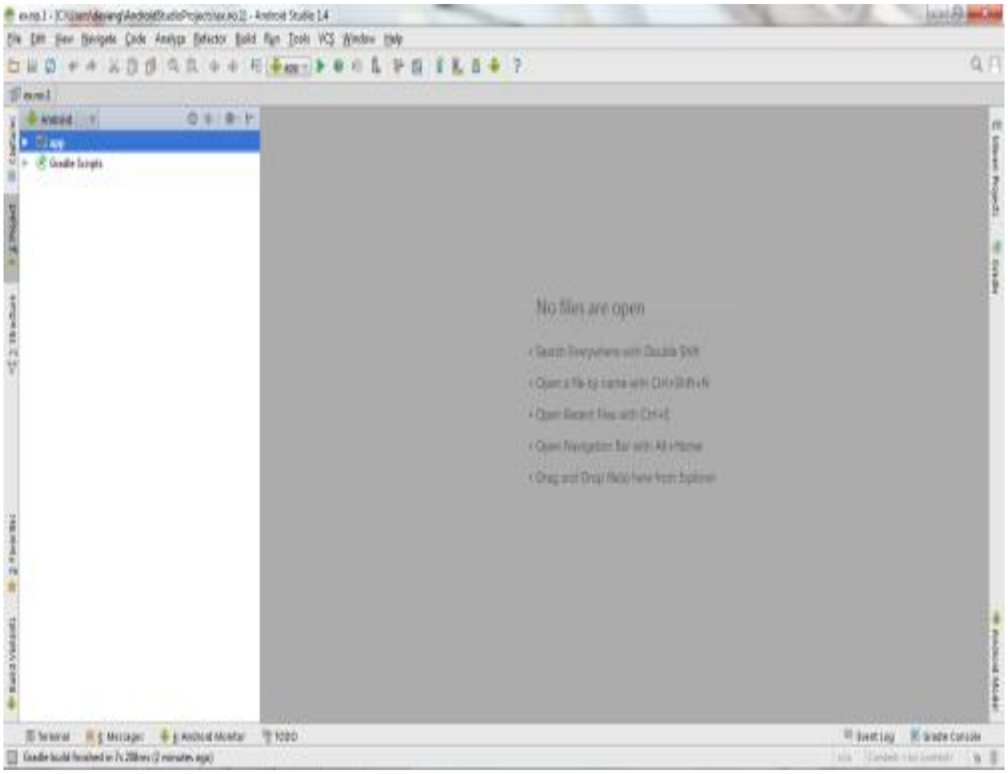
Creates a new empty activity

Activity Name: MainActivity

☑ Generate Layout File

Layout Name: activity_main

- It will take some time to build and load the project.
- After completion it will look as given below.

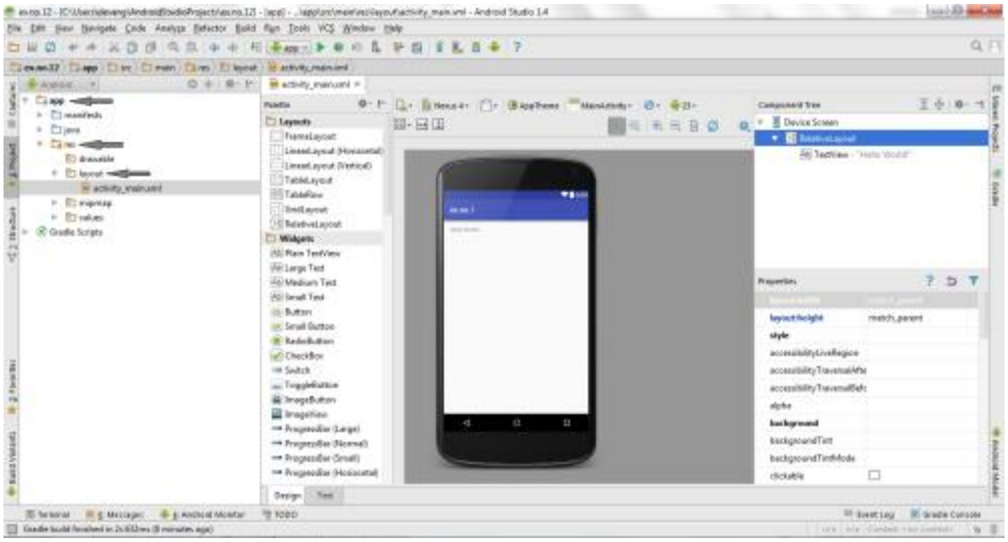


No files are open

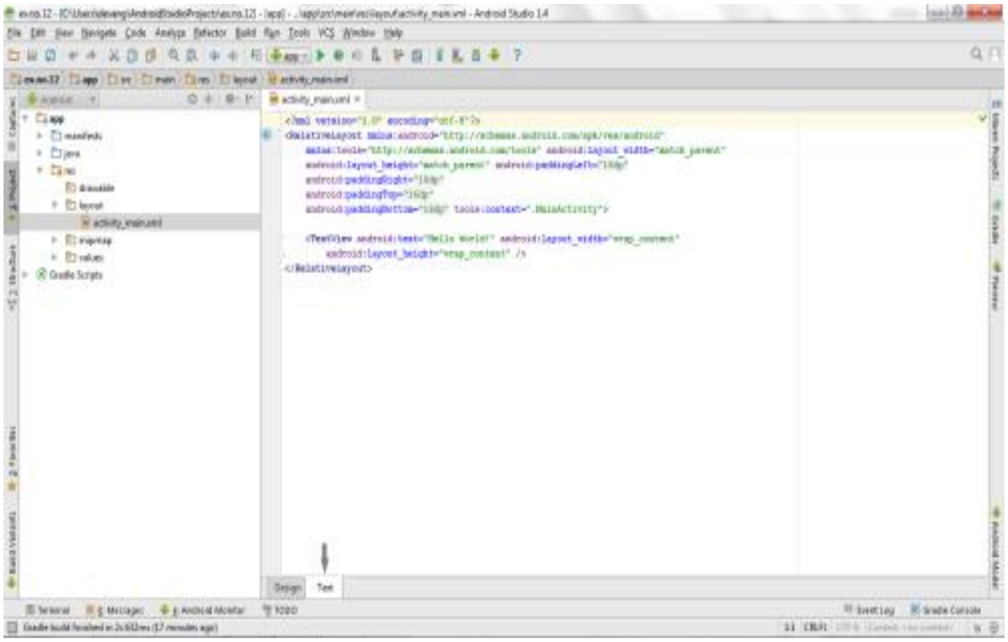
- Search Everywhere with Double Shift
- Open a file by name with Ctrl+Shift+N
- Open Recent Files with Ctrl+E
- Open Navigation Bar with Alt+Home
- Drag and Drop file(s) here from Explorer

Designing layout for the Android Application:

- Click on **app -> res -> layout -> activity_main.xml.**



- Now click on **Text** as shown below.



```
<?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:paddingLeft="16dp"
    android:paddingRight="16dp"
    android:paddingTop="16dp"
    android:paddingBottom="16dp" tools:context=".MainActivity">

    <TextView android:text="Hello World!" android:layout_width="wrap_content"
        android:layout_height="wrap_content" />
</RelativeLayout>
```

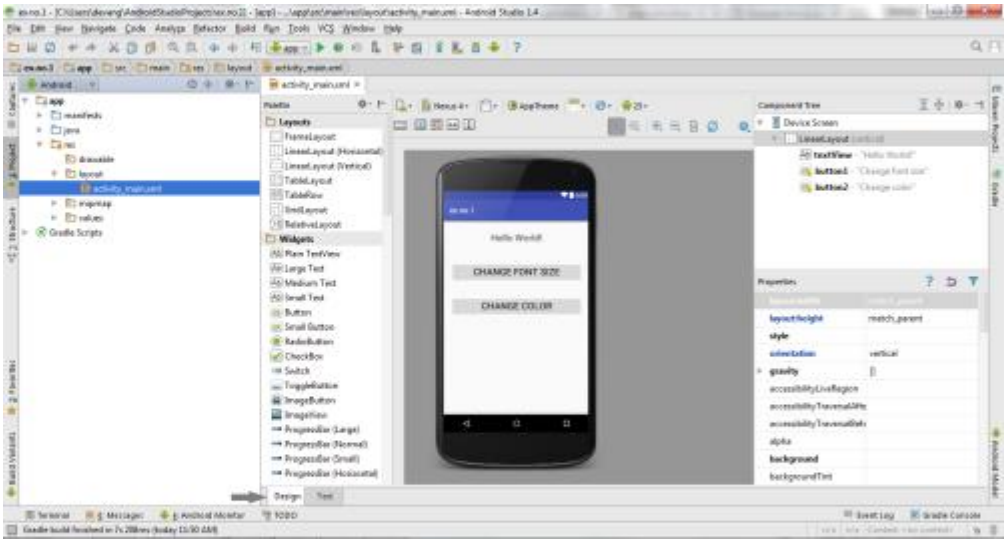
- Then delete the code which is there and type the code as given below.

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>
```

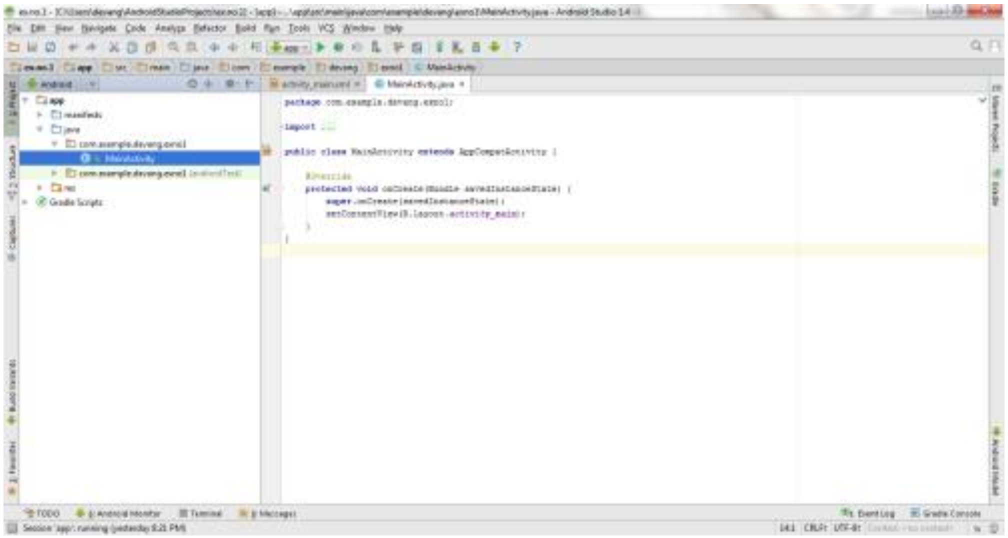
Now click on Design and your application will look as given below.



So now the designing part is completed.

Java Coding for the Android Application:

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



```
package com.example.devang.exno1;

import ...

public class MainActivity extends AppCompatActivity {

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

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

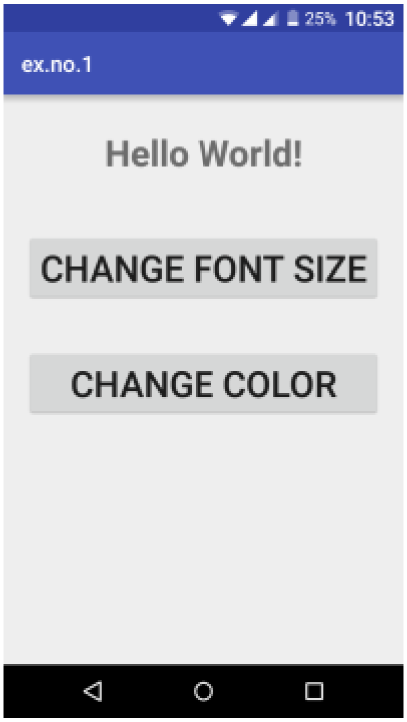
Code for MainActivity.java:

```
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;

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;
            }
        });
    }
}
```

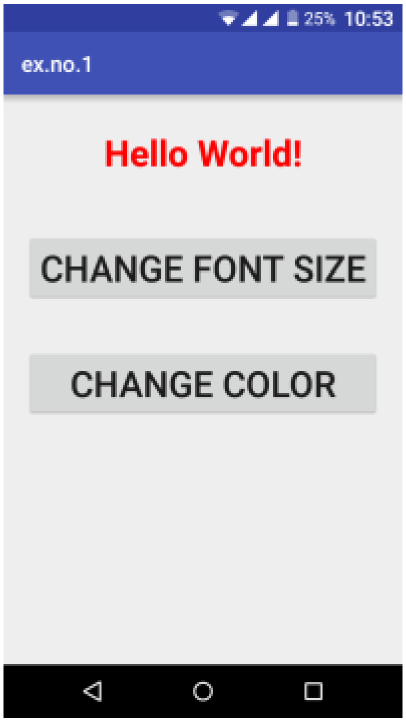
- So now the Coding part is also completed.
- Now run the application to see the output.

Output:



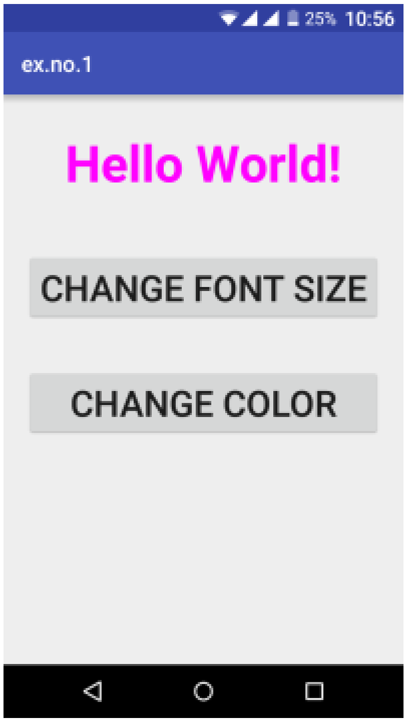
ex.no.1

Hello World!



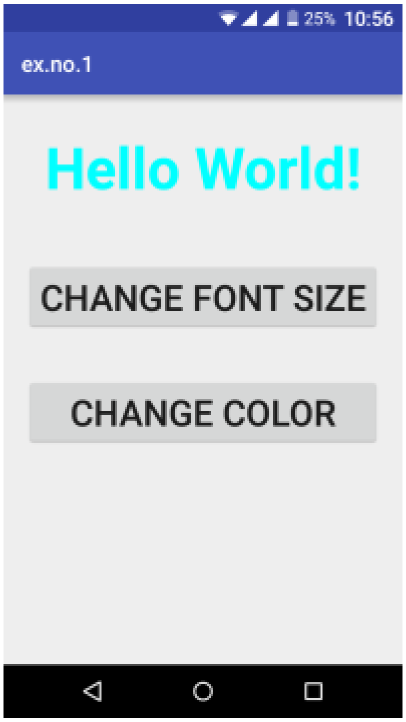
ex.no.1

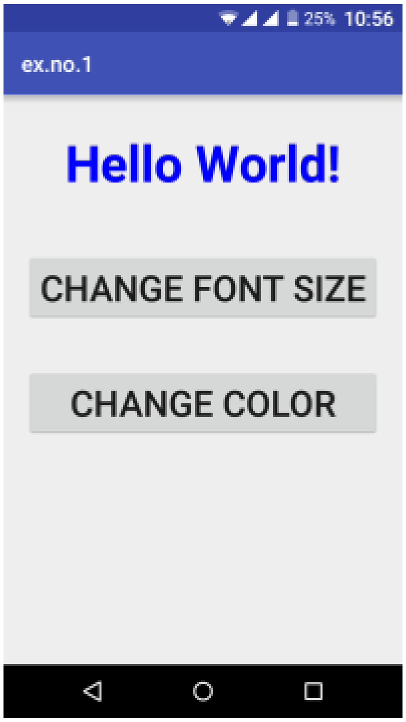
Hello World!

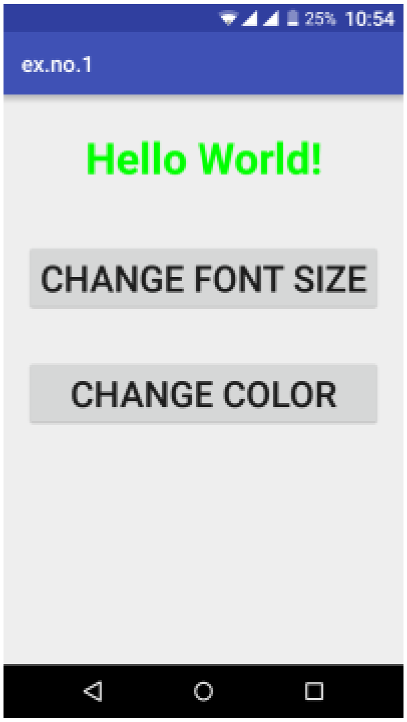


ex.no.1

Hello World!







<u>Result:</u>

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

Ex No: 2 Android Application for Layout Managers and Event Listeners

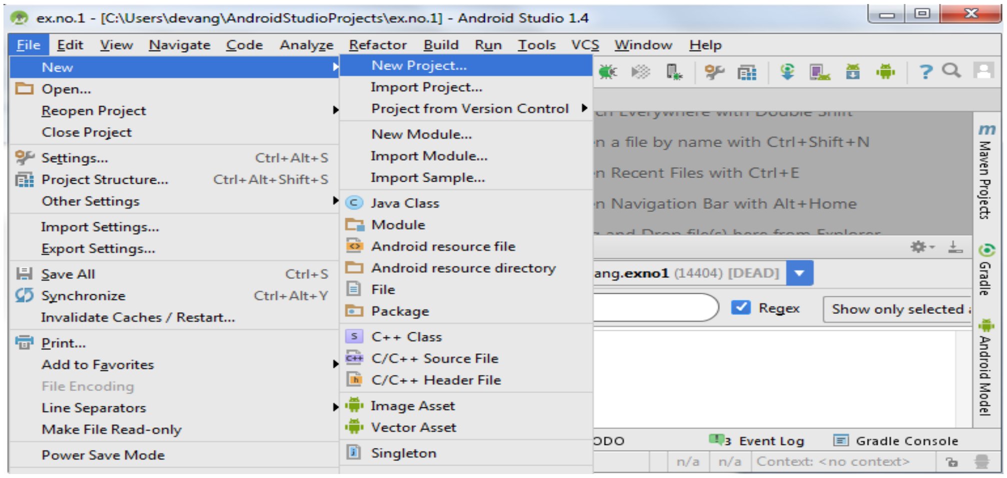
Aim:

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

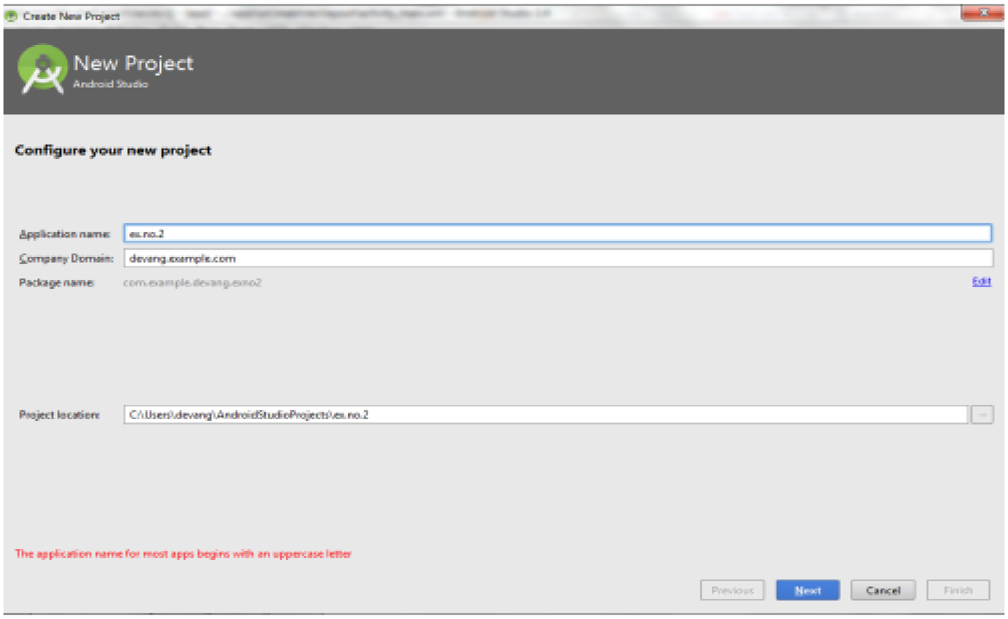
Procedure:

Creating a New project:

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



- Then type the Application name as “**ex.no.2”** and click **Next.**



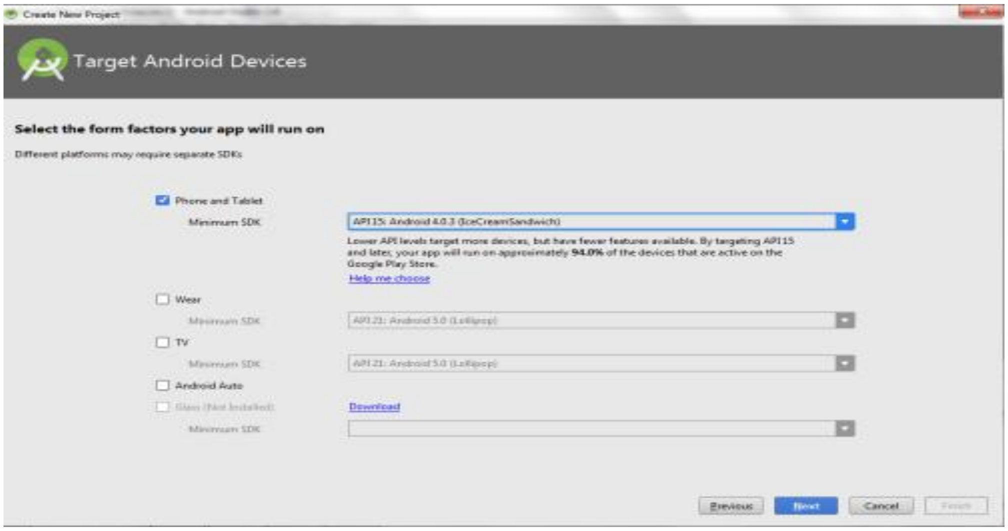
Configure your new project

Application name: ex.no.2
Company Domain: devang.example.com
Package name: com.example.devang.exno2 Edit

Project location: C:\Users\devang\AndroidStudioProjects\ex.no.2

The application name for most apps begins with an uppercase letter

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



Select the form factors your app will run on

Different platforms may require separate SDKs

☑ Phone and Tablet

Minimum SDK: API15: Android 4.0.3 (IceCreamSandwich)

Lower API levels target more devices, but have fewer features available. By targeting API15 and later, your app will run on approximately **94.0%** of the devices that are active on the Google Play Store.

Help me choose

☐ Wear

Minimum SDK: API 21: Android 5.0 (Lollipop)

☐ TV

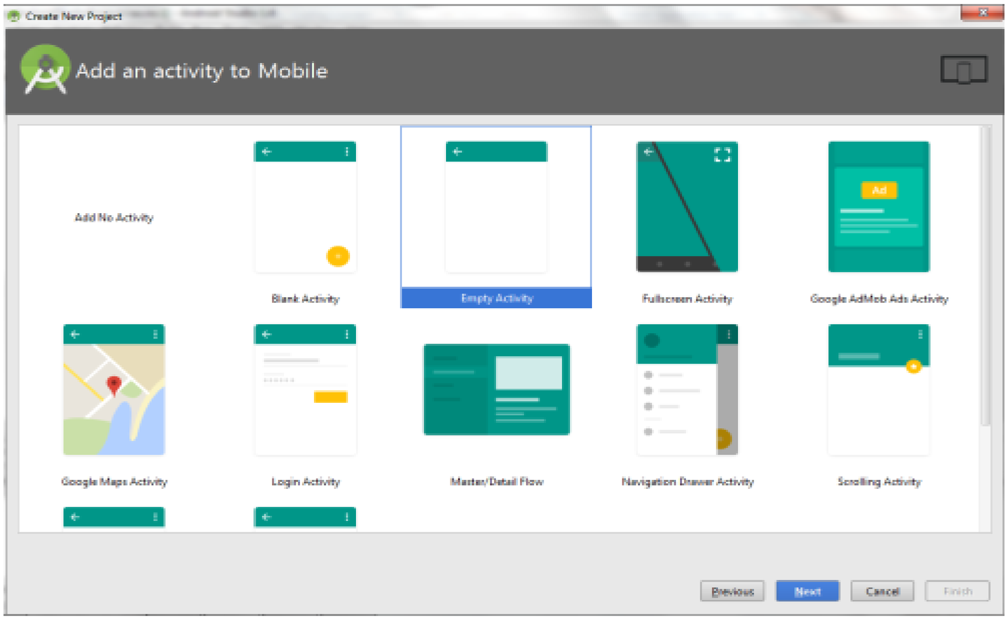
Minimum SDK: API 21: Android 5.0 (Lollipop)

☐ Android Auto

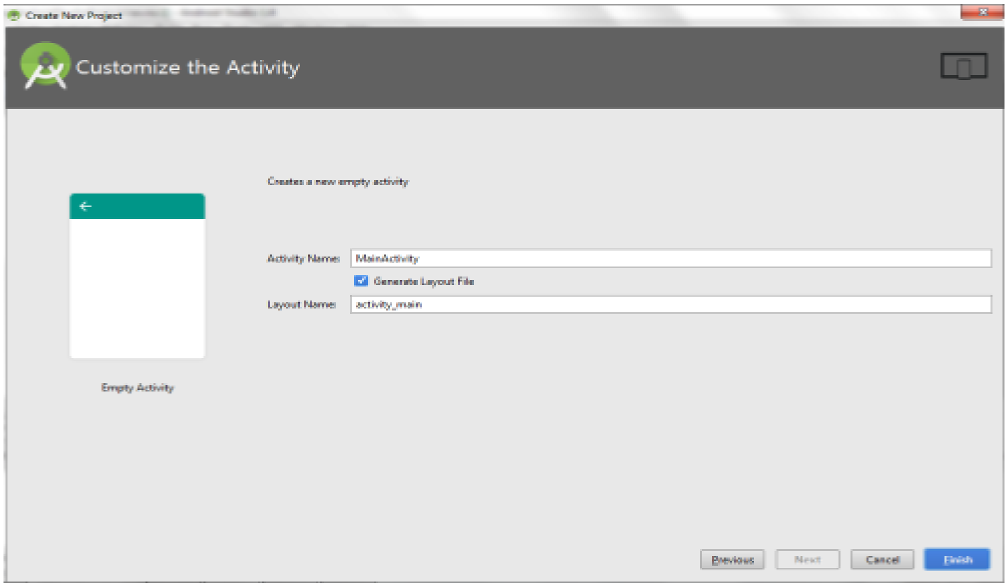
☐ Glass (Not Installed) Download

Minimum SDK

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



- Finally click F**inish**.



Creates a new empty activity

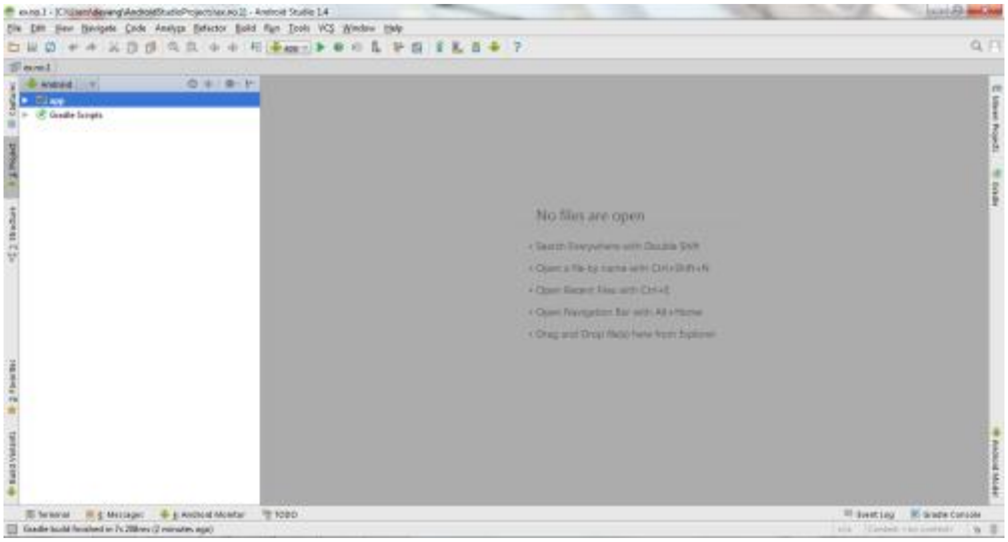
Activity Name: MainActivity

☑ Generate Layout File

Layout Name: activity_main

Empty Activity

- It will take some time to build and load the project.
- After completion it will look as given below.



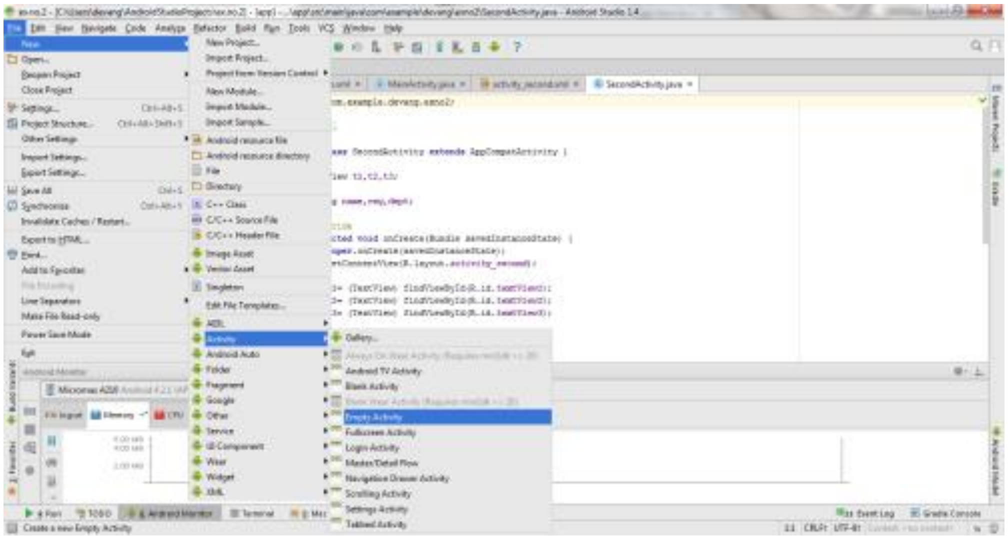
No files are open

- Search Everywhere with Double Shift
- Open a file by name with Ctrl+Shift+N
- Open Recent Files with Ctrl+E
- Open Navigation Bar with Alt+Home
- Drag and Drop file(s) here from Explorer

Gradle build finished in 7s 208ms (2 minutes ago)

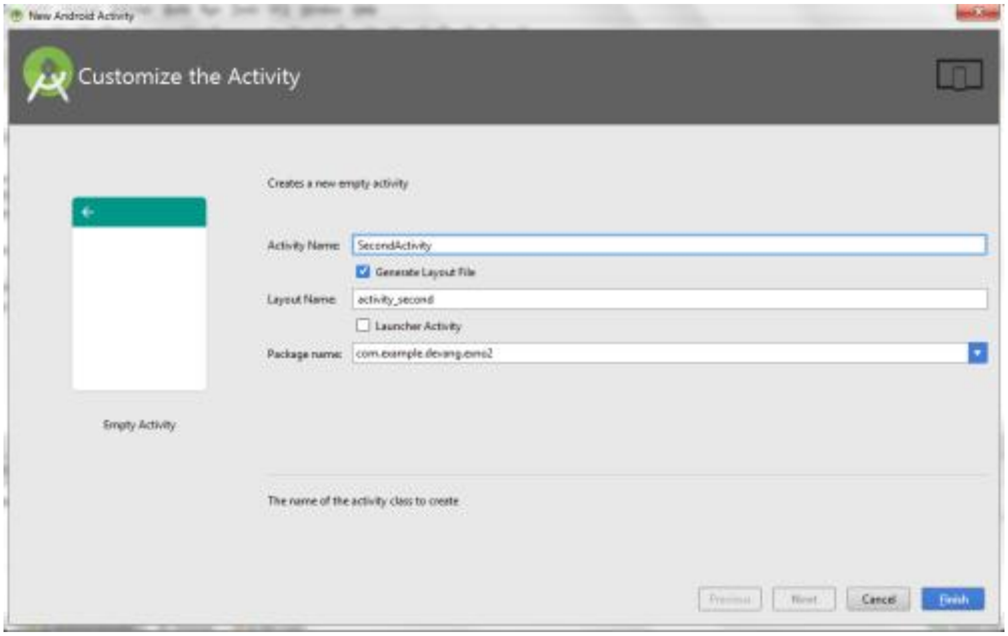
Creating Second Activity for the Android Application:

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



```
om.example.devang.exno2;

ass SecondActivity extends AppCompatActivity {

iew t1,t2,t3;

g name,reg,dept;

tted void onCreate(Bundle savedInstanceState) {
uper.onCreate(savedInstanceState);
etContentView(R.layout.activity_second);

1= (TextView) findViewById(R.id.textView1);
2= (TextView) findViewById(R.id.textView2);
3= (TextView) findViewById(R.id.textView3);
```

- Type the Activity Name as **SecondActivity** and click **Finish** button.
-



Creates a new empty activity

Activity Name: SecondActivity

☑ Generate Layout File

Layout Name: activity_second

☐ Launcher Activity

Package name: com.example.devang.exno2

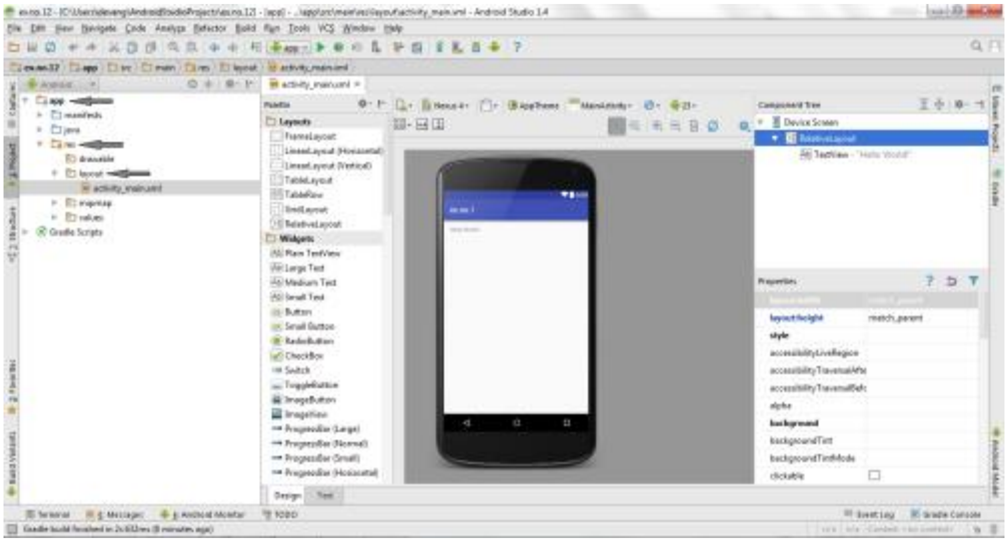
The name of the activity class to create

- Thus Second Activity For the application is created.

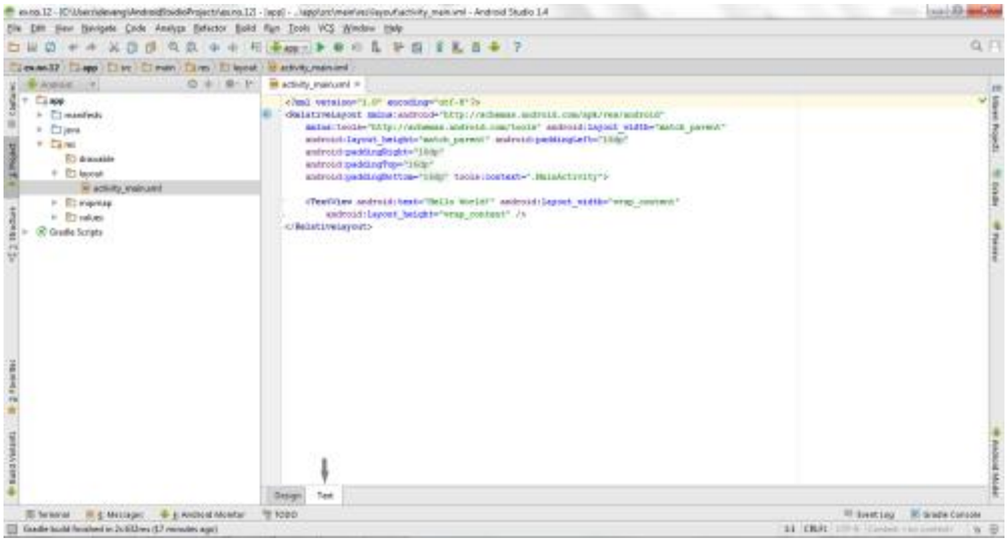
Designing layout for the Android Application:

Designing Layout for Main Activity:

- Click on **app -> res -> layout -> activity_main.xml.**



- Now click on **Text** as shown below.



```
<?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:paddingLeft="16dp"
    android:paddingRight="16dp"
    android:paddingTop="16dp"
    android:paddingBottom="16dp" tools:context=".MainActivity">

    <TextView android:text="Hello World!" android:layout_width="wrap_content"
        android:layout_height="wrap_content" />
</RelativeLayout>
```

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

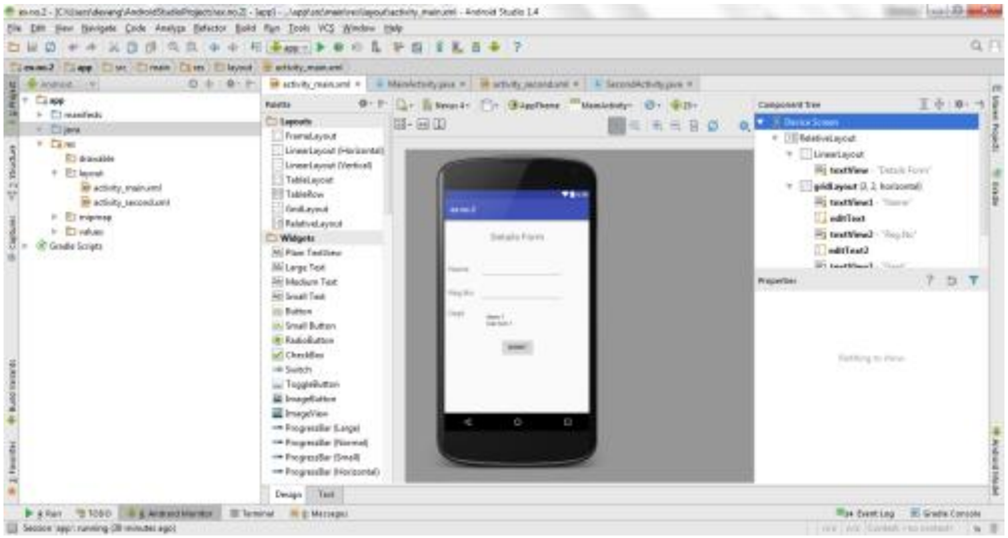
Code for Activity_main.xml:

```
<?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>
```

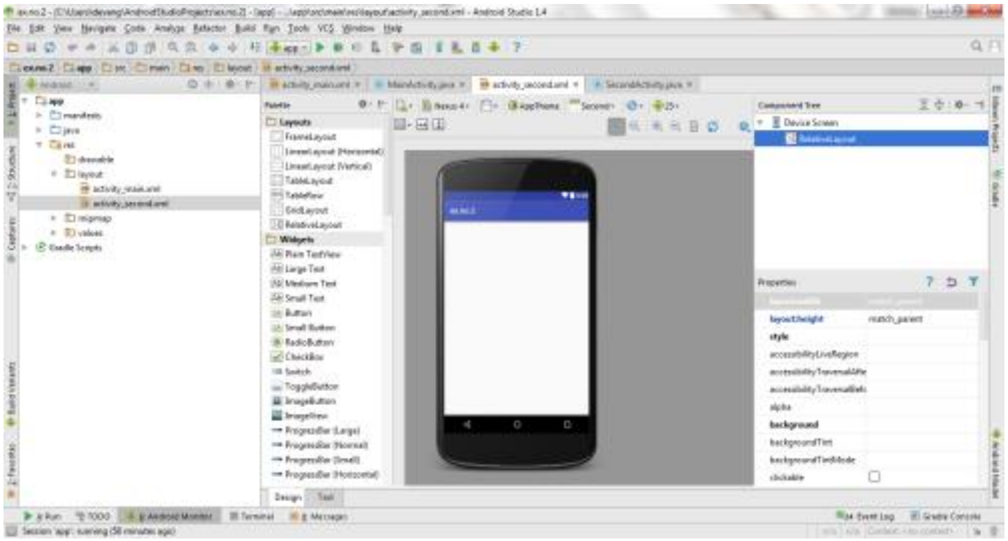
- Now click on Design and your activity will look as given below.
-



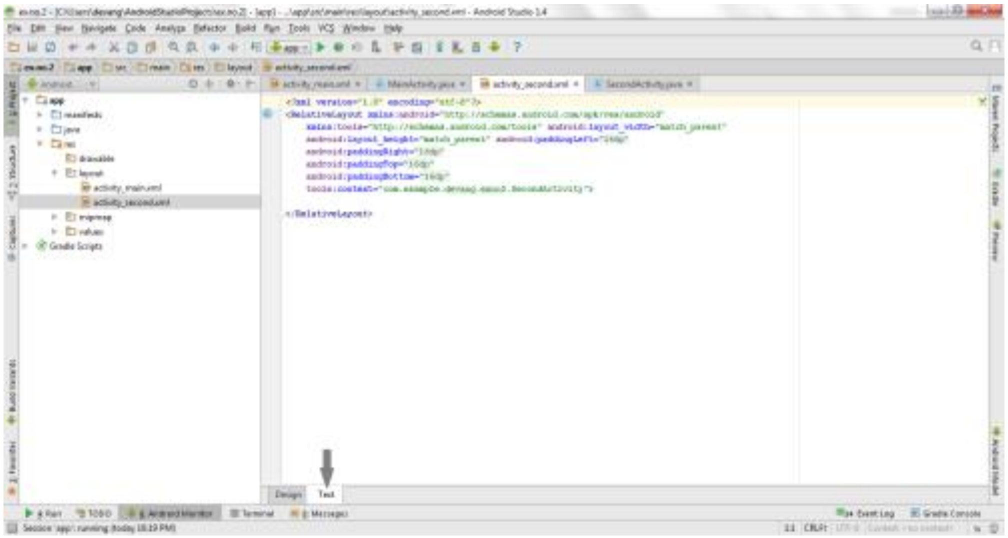
- So now the designing part of Main Activity is completed.

Designing Layout for Second Activity:

- Click on **app -> res -> layout -> activity_second.xml.**



- Now click on **Text** as shown below.



```
<?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:paddingLeft="16dp"
    android:paddingRight="16dp"
    android:paddingTop="16dp"
    android:paddingBottom="16dp"
    tools:context="com.example.devang.exno2.SecondActivity">

</RelativeLayout>
```

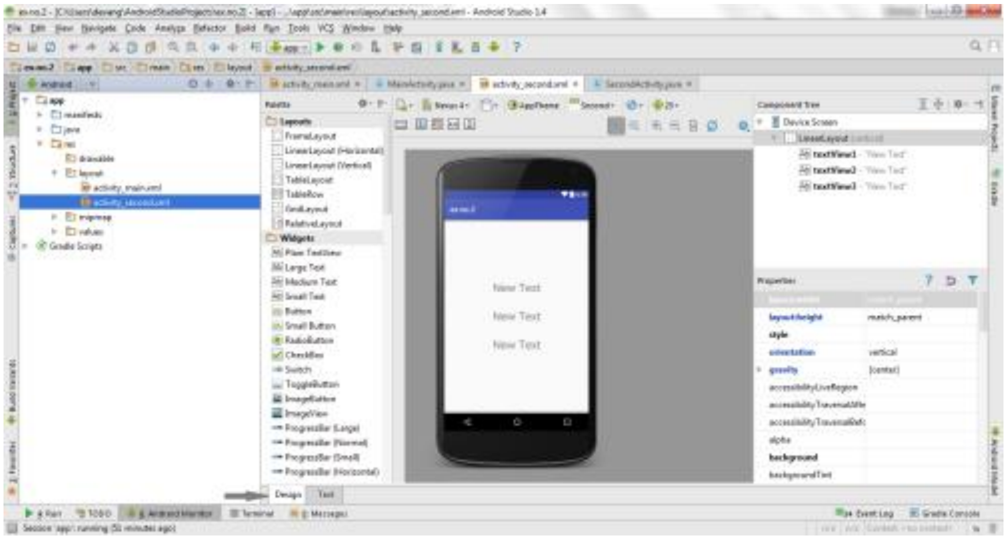
- Then delete the code which is there and type the code as given below.

Code for Activity_second.xml:

```
<?xml version="1.0" encoding="utf-8"?>
<LinearLayout xmlns:android="http://schemas.android.com/apk/res/android"
  xmlns:tools="http://schemas.android.com/tools"
  android:layout_width="match_parent"
  android:layout_height="match_parent"
  tools:context="com.example.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.

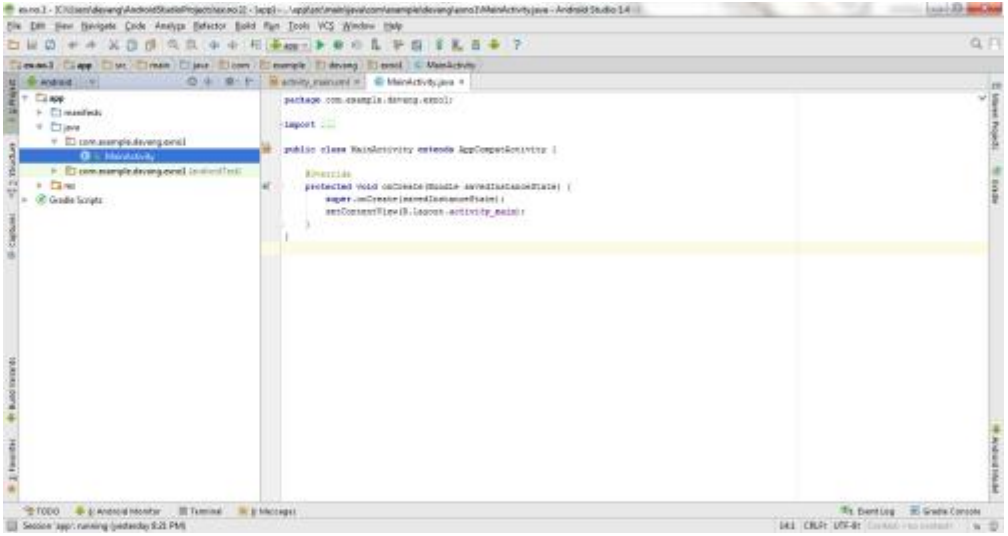


- So now the designing part of Second Activity is also completed.

Java Coding for the Android Application:

Java Coidng for Main Activity:

- Click on **app -> java -> com.example.exno2 -> MainActivity.**



```
package com.example.devang.exno2;

import ...

public class MainActivity extends AppCompatActivity {

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

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

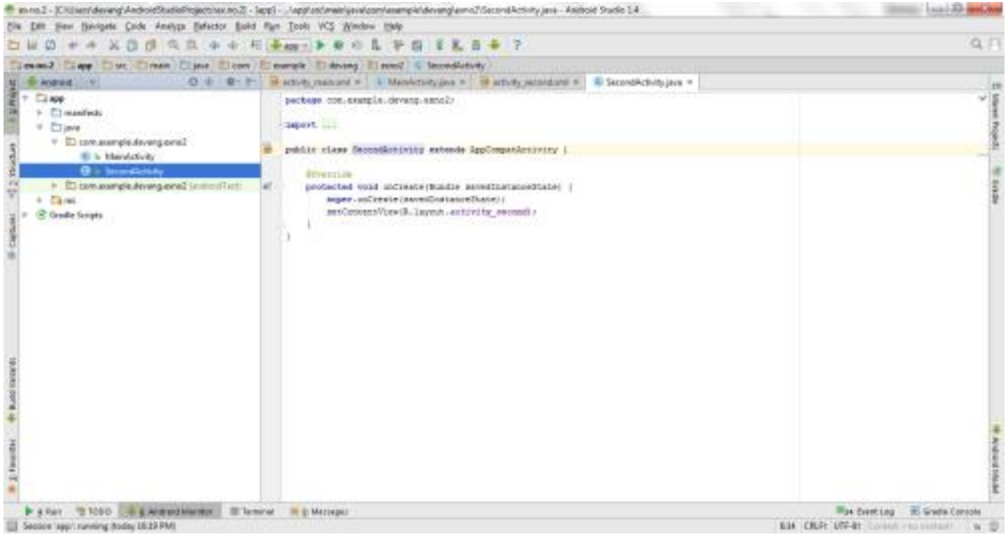
Code for MainActivity.java:

```
package com.example.exno2;

import android.content.Intent;
import android.support.v7.app.AppCompatActivity;
import android.os.Bundle;
import android.view.View;
import android.widget.ArrayAdapter;
import android.widget.Button;
import android.widget.EditText;
import android.widget.Spinner;

public class MainActivity extends AppCompatActivity {

    //Defining the Views
    EditText e1,e2;
    Button bt;
    Spinner s;

    //Data for populating in Spinner
    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);

        //Referring the Views
        e1= (EditText) findViewById(R.id.editText);
        e2= (EditText) findViewById(R.id.editText2);

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

        s= (Spinner) findViewById(R.id.spinner);

        //Creating Adapter for Spinner for adapting the data from array to Spinner
        ArrayAdapter adapter= new
ArrayAdapter(MainActivity.this,android.R.layout.simple_spinner_item,dept_array);
        s.setAdapter(adapter);

        //Creating Listener for Button
        bt.setOnClickListener(new View.OnClickListener() {
            @Override
            public void onClick(View v) {
                //Getting the Values from Views(Edittext & Spinner)
                name=e1.getText().toString();
                reg=e2.getText().toString();
                dept=s.getSelectedItem().toString();

                //Intent For Navigating to Second Activity
                Intent i = new Intent(MainActivity.this,SecondActivity.class);

                //For Passing the Values to Second Activity
                i.putExtra("name_key", name);
                i.putExtra("reg_key",reg);
                i.putExtra("dept_key", dept);

                startActivity(i);

            }
        });
    }
}
```

- So now the Coding part of Main Activity is completed.

Java Coding for Second Activity:

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



```
package com.example.devang.exno2;

import ...

public class SecondActivity extends AppCompatActivity {

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

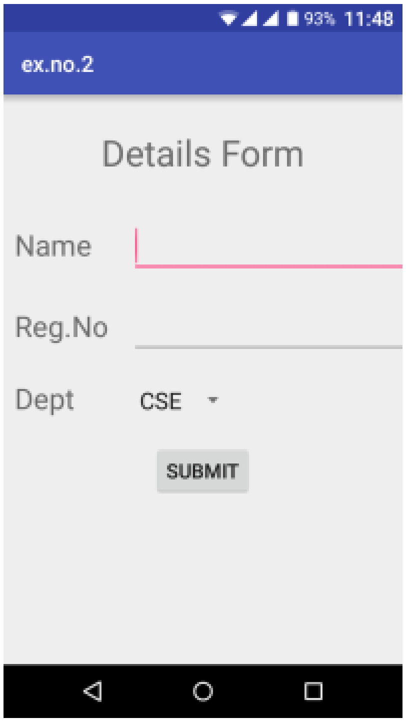
- Then delete the code which is there and type the code as given below.

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;

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);

    }
}
```

- So now the Coding part of Second Activity is also completed.
- Now run the application to see the output.

Output:

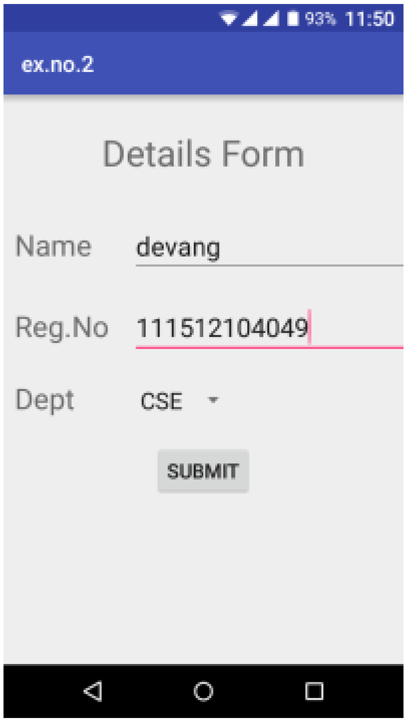


Details Form

Name

Reg.No

Dept CSE

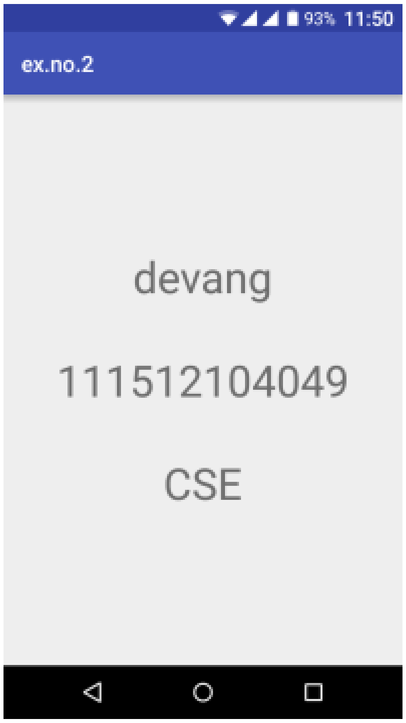


Details Form

Name devang

Reg.No 111512104049

Dept CSE



devang

111512104049

CSE

Result:

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

Ex No: 3 Simple Android Application for Native Calculator

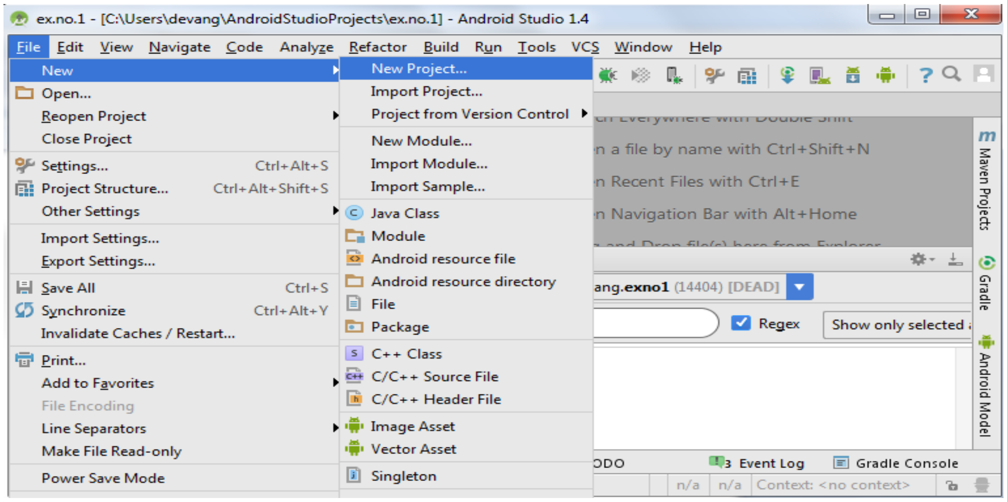
Aim:

To develop a Simple Android Application for Native Calculator.

Procedure:

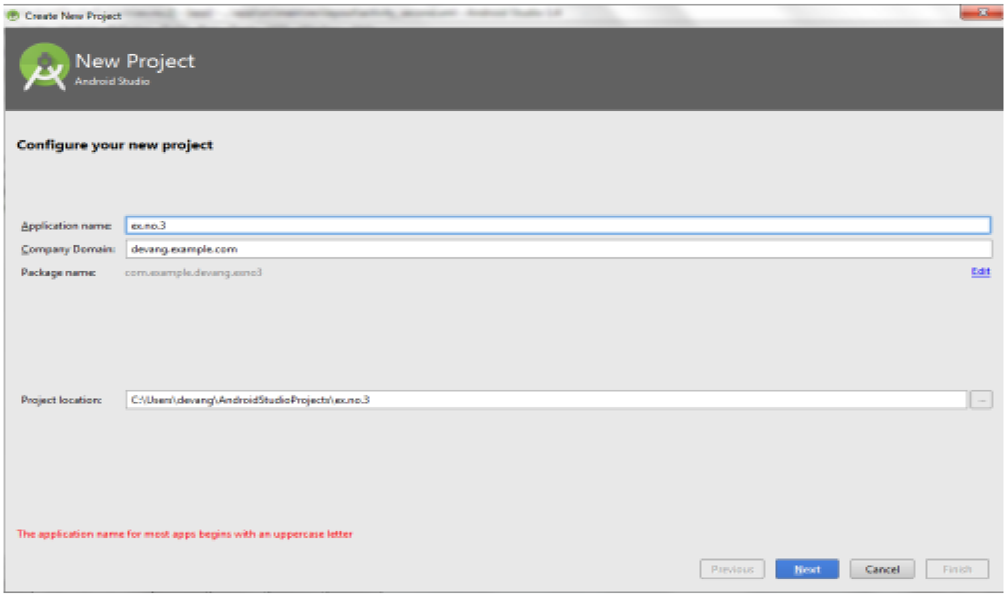
Creating a New project:

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



ex.no.1 - [C:\Users\devang\AndroidStudioProjects\ex.no.1] - Android Studio 1.4

- Then type the Application name as "**ex.no.3"** and click **Next.**



Create New Project

New Project
Android Studio

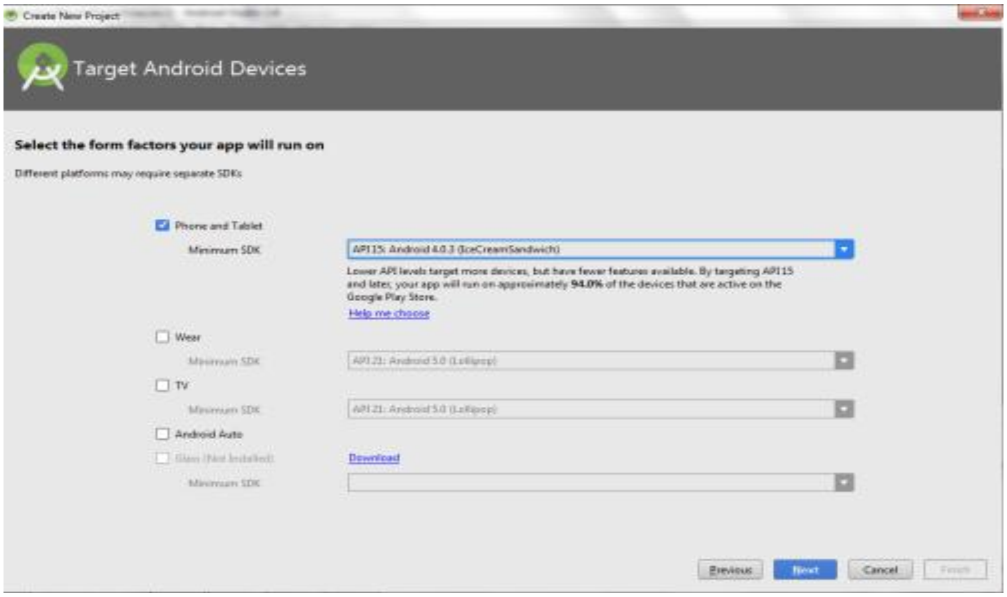
Configure your new project

Application name: ex.no.3
Company Domain: devang.example.com
Package name: com.example.devang.exno3

Project location: C:\Users\devang\AndroidStudioProjects\ex.no.3

The application name for most apps begins with an uppercase letter

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



Target Android Devices

Select the form factors your app will run on

Different platforms may require separate SDKs

☑ Phone and Tablet

Minimum SDK: API15: Android 4.0.3 (IceCreamSandwich)

Lower API levels target more devices, but have fewer features available. By targeting API15 and later, your app will run on approximately **94.0%** of the devices that are active on the Google Play Store.

Help me choose

□ Wear

Minimum SDK: API 21: Android 5.0 (Lollipop)

□ TV

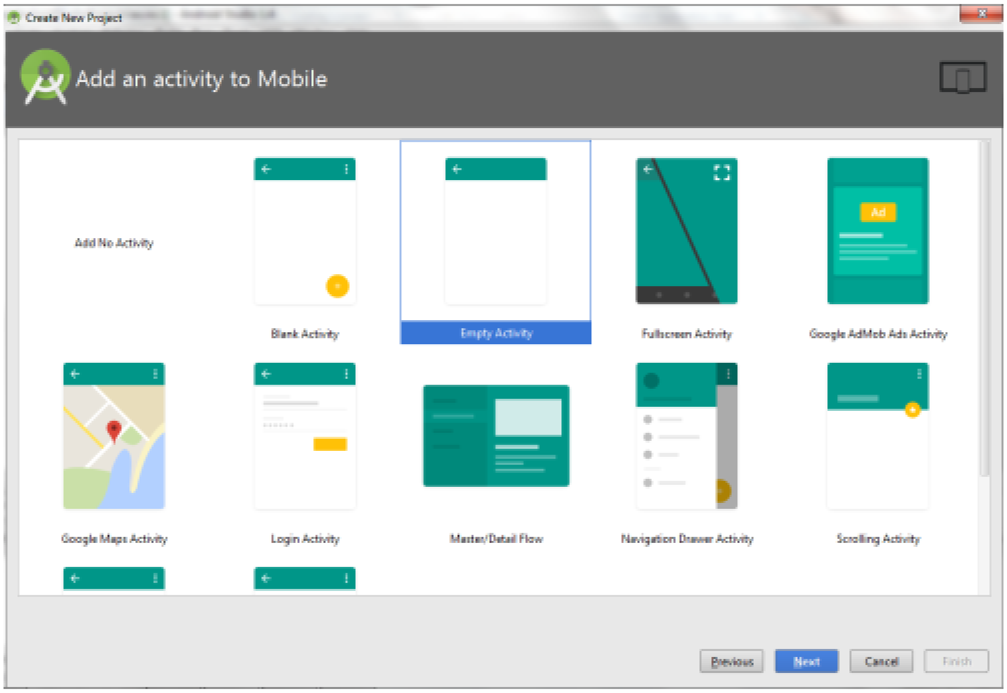
Minimum SDK: API 21: Android 5.0 (Lollipop)

□ Android Auto

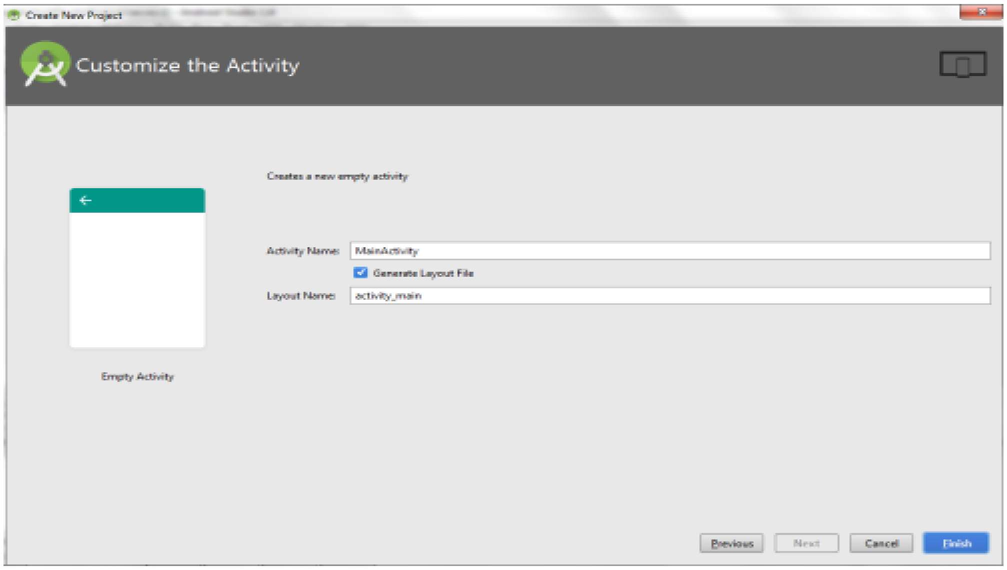
□ Glass (Not Installed) Download

Minimum SDK

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



- Finally click **Finish**.



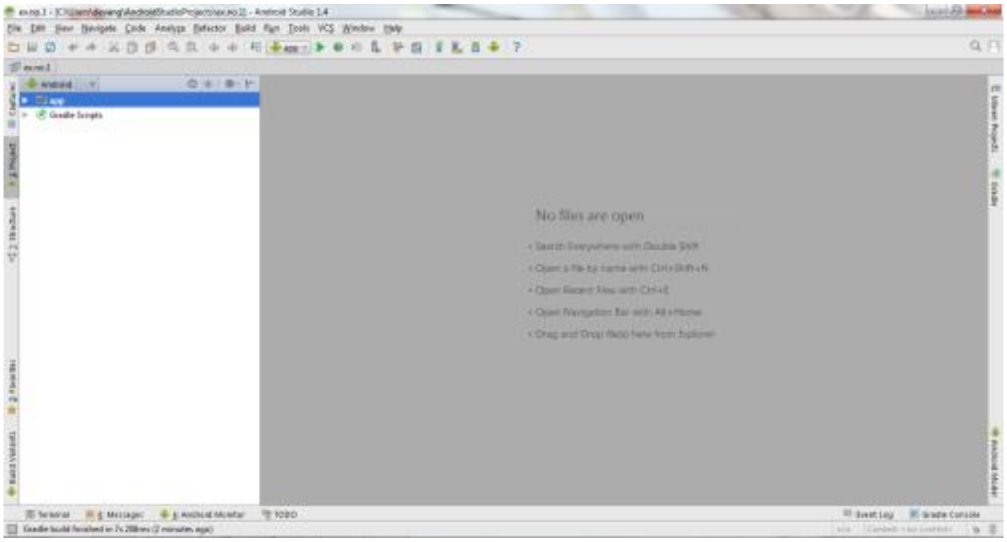
Creates a new empty activity

Activity Name: MainActivity

☑ Generate Layout File

Layout Name: activity_main

- It will take some time to build and load the project.
- After completion it will look as given below.



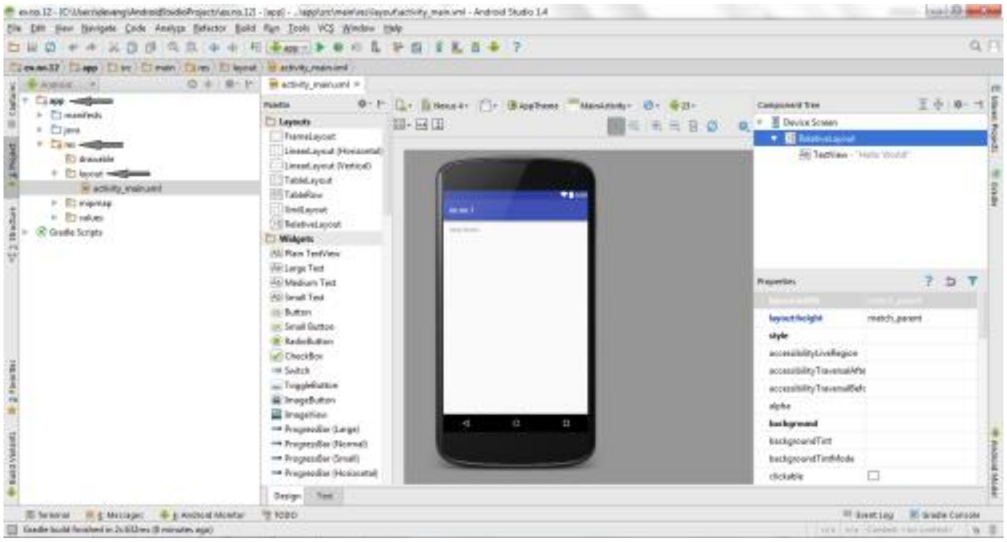
No files are open

- Search Everywhere with Double Shift
- Open a file by name with Ctrl+Shift+N
- Open Recent Files with Ctrl+E
- Open Navigation Bar with Alt+Home
- Drag and Drop file(s) here from Explorer

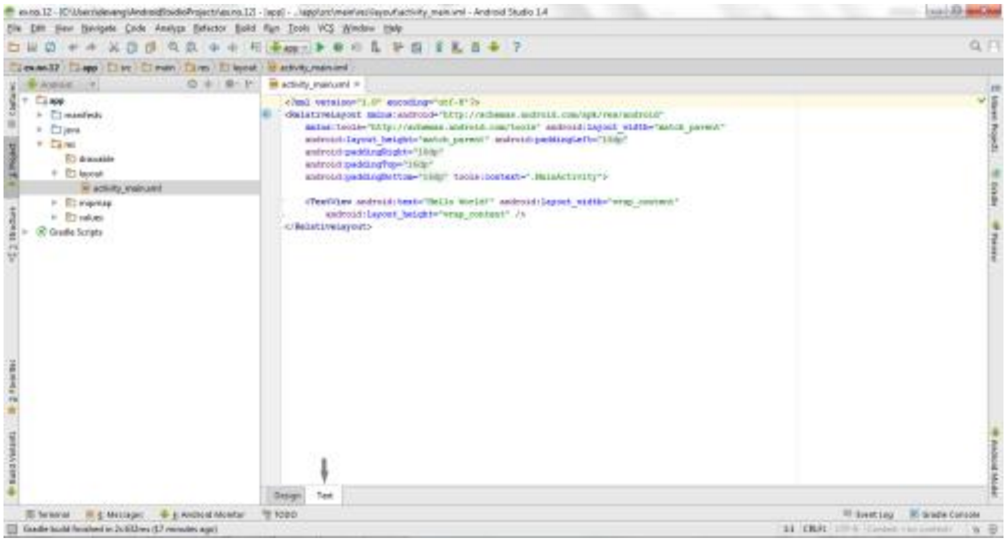
Gradle build finished in 7s 208ms (2 minutes ago)

Designing layout for the Android Application:

- Click on **app -> res -> layout -> activity_main.xml.**



- Now click on **Text** as shown below.



```
<?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:paddingLeft="16dp"
    android:paddingRight="16dp"
    android:paddingTop="16dp"
    android:paddingBottom="16dp" tools:context=".MainActivity">

    <TextView android:text="Hello World!" android:layout_width="wrap_content"
        android:layout_height="wrap_content" />
</RelativeLayout>
```

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

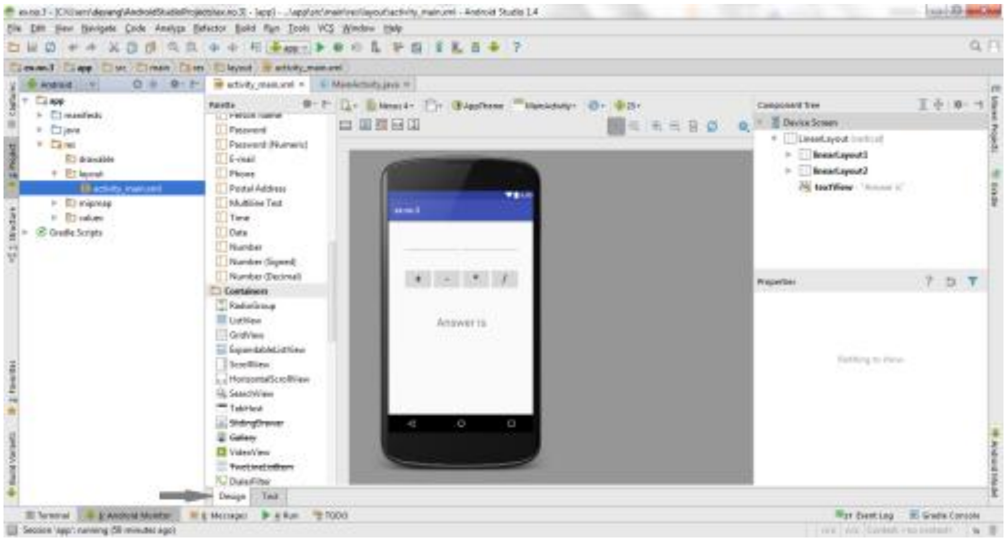
Code for Activity_main.xml:

```
<?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"
   android:layout_margin="20dp">

   <LinearLayout
      android:id="@+id/linearLayout1"
      android:layout_width="match_parent"
      android:layout_height="wrap_content"
```

```
    android:layout_margin="20dp">

    <EditText
        android:id="@+id/editText1"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:layout_weight="1"
        android:inputType="numberDecimal"
        android:textSize="20sp" />

    <EditText
        android:id="@+id/editText2"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:layout_weight="1"
        android:inputType="numberDecimal"
        android:textSize="20sp" />

</LinearLayout>

<LinearLayout
    android:id="@+id/linearLayout2"
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:layout_margin="20dp">

    <Button
        android:id="@+id/Add"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:layout_weight="1"
        android:text="+"
        android:textSize="30sp"/>

    <Button
        android:id="@+id/Sub"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:layout_weight="1"
        android:text="-"
        android:textSize="30sp"/>

    <Button
        android:id="@+id/Mul"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:layout_weight="1"
        android:text="*"
```

```
            android:textSize="30sp"/>

        <Button
            android:id="@+id/Div"
            android:layout_width="match_parent"
            android:layout_height="wrap_content"
            android:layout_weight="1"
            android:text="/"
            android:textSize="30sp"/>

    </LinearLayout>

    <TextView
        android:id="@+id/textView"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:layout_marginTop="50dp"
        android:text="Answer is"
        android:textSize="30sp"
        android:gravity="center"/>

</LinearLayout>
```

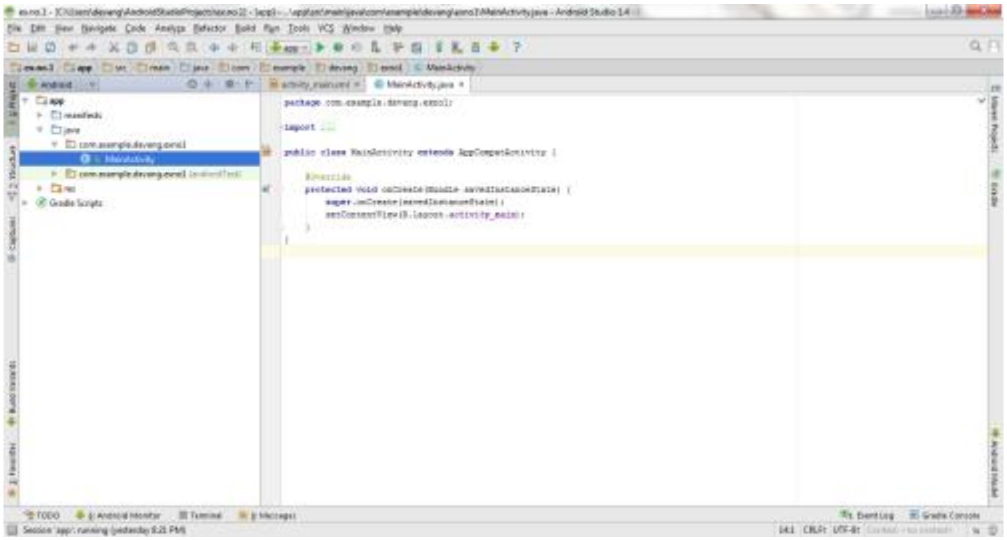
- Now click on Design and your application will look as given below.



- So now the designing part is completed.

Java Coding for the Android Application:

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



```
package com.example.devang.exno3;

import ...

public class MainActivity extends AppCompatActivity {

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

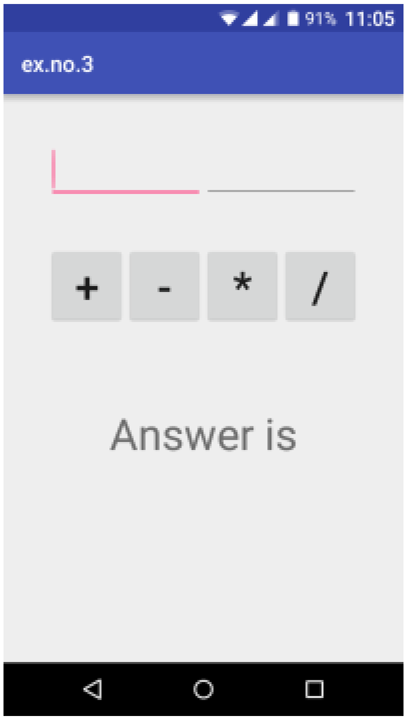
- Then delete the code which is there and type the code as given below.

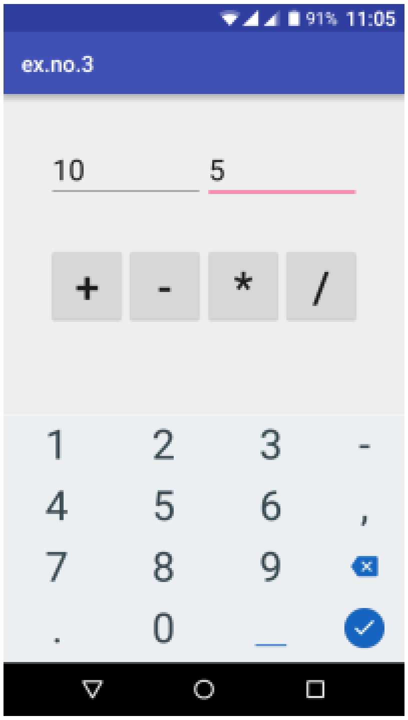
Code for MainActivity.java:

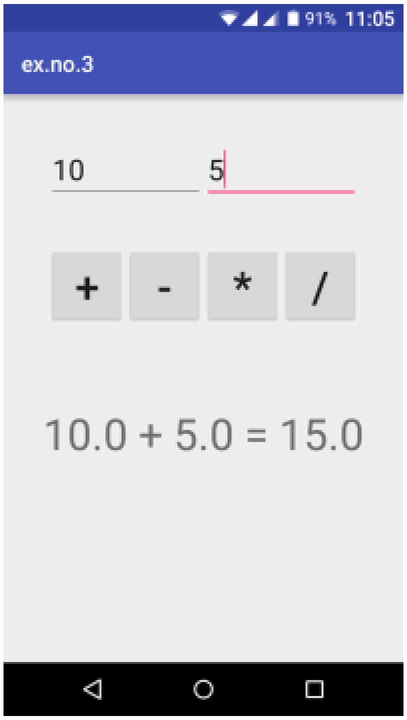
```
package com.example.devang.exno3;

import android.os.Bundle;
import android.support.v7.app.AppCompatActivity;
import android.text.TextUtils;
import android.view.View;
import android.view.View.OnClickListener;
import android.widget.Button;
import android.widget.EditText;
import android.widget.TextView;

public class MainActivity extends AppCompatActivity implements OnClickListener
{
  //Defining the Views
  EditText Num1;
  EditText Num2;
  Button Add;
  Button Sub;
  Button Mul;
  Button Div;
  TextView Result;

  @Override
  public void onCreate(Bundle savedInstanceState)
  {
    super.onCreate(savedInstanceState);
    setContentView(R.layout.activity_main);
    //Referring the Views
    Num1 = (EditText) findViewById(R.id.editText1);
    Num2 = (EditText) findViewById(R.id.editText2);
    Add = (Button) findViewById(R.id.Add);
    Sub = (Button) findViewById(R.id.Sub);
    Mul = (Button) findViewById(R.id.Mul);
    Div = (Button) findViewById(R.id.Div);
    Result = (TextView) findViewById(R.id.textView);

    // set a listener
    Add.setOnClickListener(this);
    Sub.setOnClickListener(this);
    Mul.setOnClickListener(this);
    Div.setOnClickListener(this);
  }

  @Override
  public void onClick (View v)
  {

    float num1 = 0;
    float num2 = 0;
    float result = 0;
    String oper = "";

    // check if the fields are empty
    if (TextUtils.isEmpty(Num1.getText().toString()) ||
TextUtils.isEmpty(Num2.getText().toString()))
        return;

    // read EditText and fill variables with numbers
    num1 = Float.parseFloat(Num1.getText().toString());
    num2 = Float.parseFloat(Num2.getText().toString());

    // defines the button that has been clicked and performs the corresponding operation
    // write operation into oper, we will use it later for output
    switch (v.getId())
    {
      case R.id.Add:
        oper = "+";
        result = num1 + num2;
        break;
      case R.id.Sub:
        oper = "-";
        result = num1 - num2;
        break;
      case R.id.Mul:
        oper = "*";
        result = num1 * num2;
        break;
      case R.id.Div:
        oper = "/";
        result = num1 / num2;
        break;
      default:
        break;
    }
    // form the output line
    Result.setText(num1 + " " + oper + " " + num2 + " = " + result);
  }
}
```

- So now the Coding part is also completed.
- Now run the application to see the output.

<u>Output:</u>







```
10        5

10.0 + 5.0 = 15.0
```

Result:

Thus a Simple Android Application for Native Calculator is developed and executed successfully.

Ex No: 4 Android Application to draw Basic Graphical Primitives

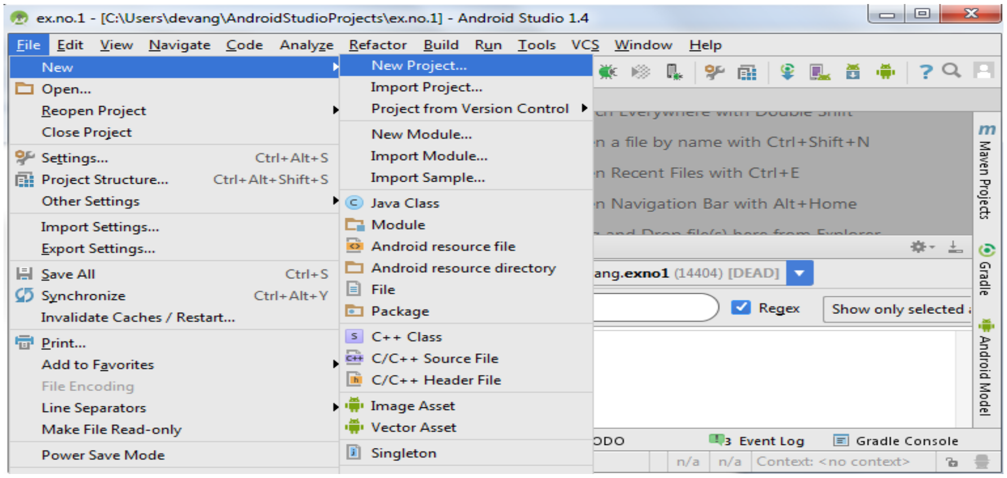
Aim:

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

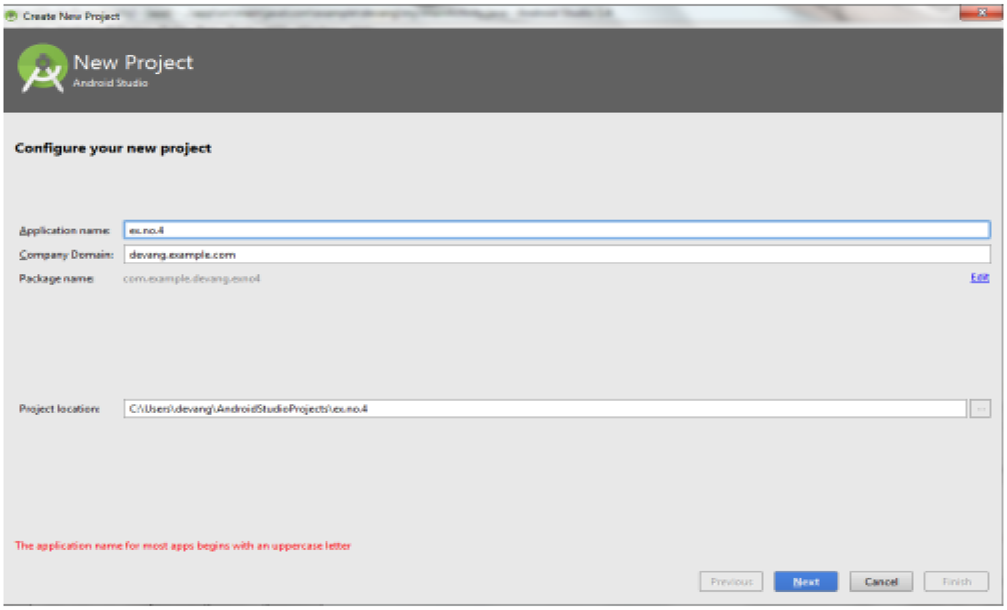
Procedure:

Creating a New project:

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



- Then type the Application name as “**ex.no.4”** and click **Next.**



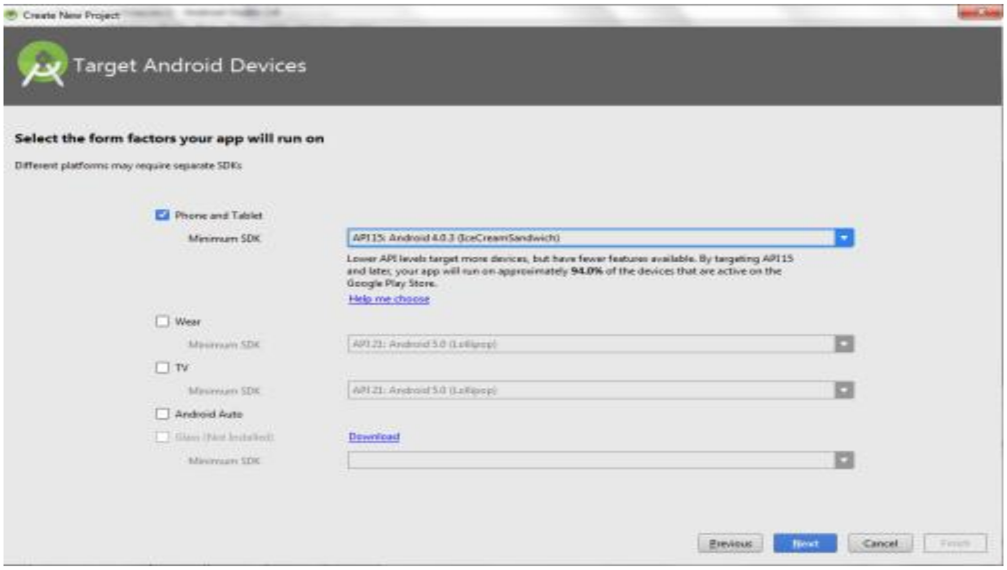
Configure your new project

Application name: ex.no.4
Company Domain: devang.example.com
Package name: com.example.devang.exno4

Project location: C:\Users\devang\AndroidStudioProjects\ex.no.4

The application name for most apps begins with an uppercase letter

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



Create New Project

Target Android Devices

Select the form factors your app will run on

Different platforms may require separate SDKs

☑ Phone and Tablet

Minimum SDK: API 15: Android 4.0.3 (IceCreamSandwich)

Lower API levels target more devices, but have fewer features available. By targeting API 15 and later, your app will run on approximately **94.0%** of the devices that are active on the Google Play Store.

Help me choose

☐ Wear

Minimum SDK: API 21: Android 5.0 (Lollipop)

☐ TV

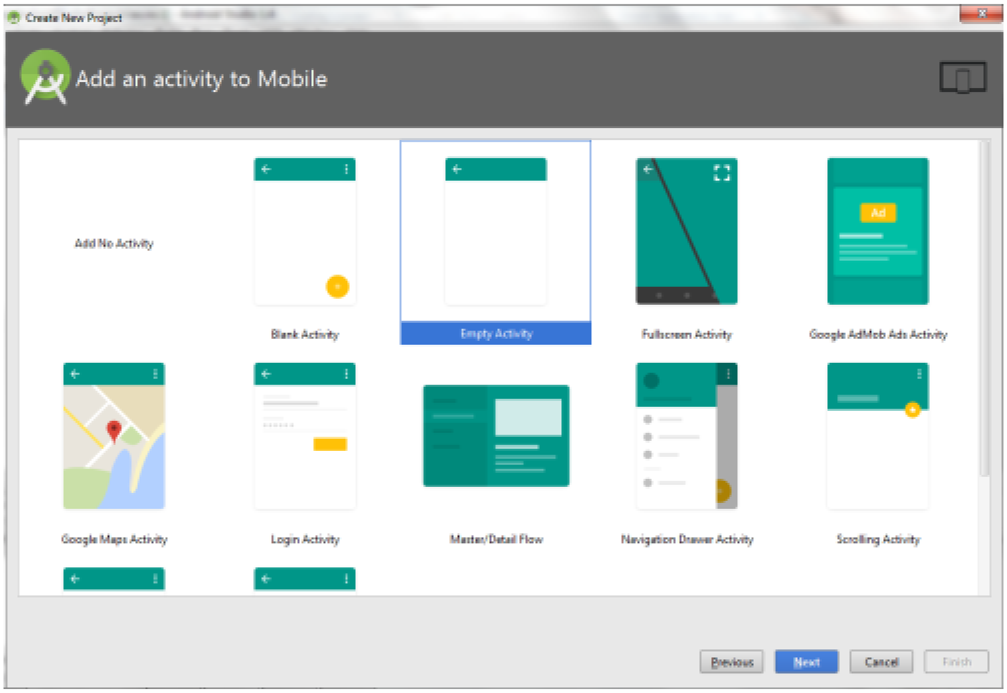
Minimum SDK: API 21: Android 5.0 (Lollipop)

☐ Android Auto

☐ Glass (Not Installed) Download

Minimum SDK

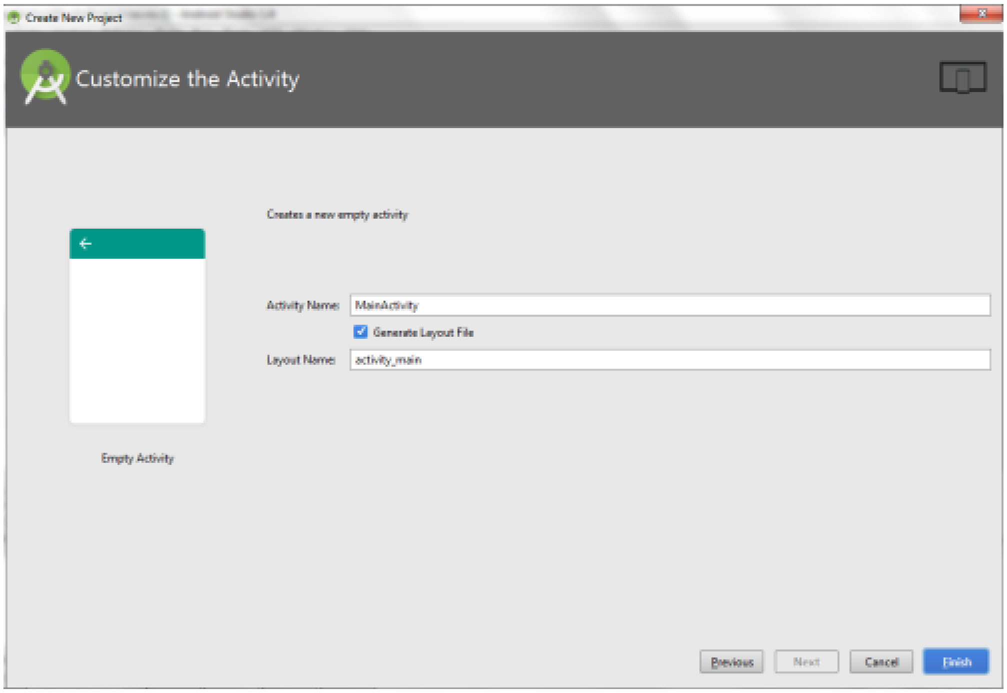
- Then select the **Empty Activity** and click **Next.**



Create New Project

Add an activity to Mobile

- Finally click F**inish.**



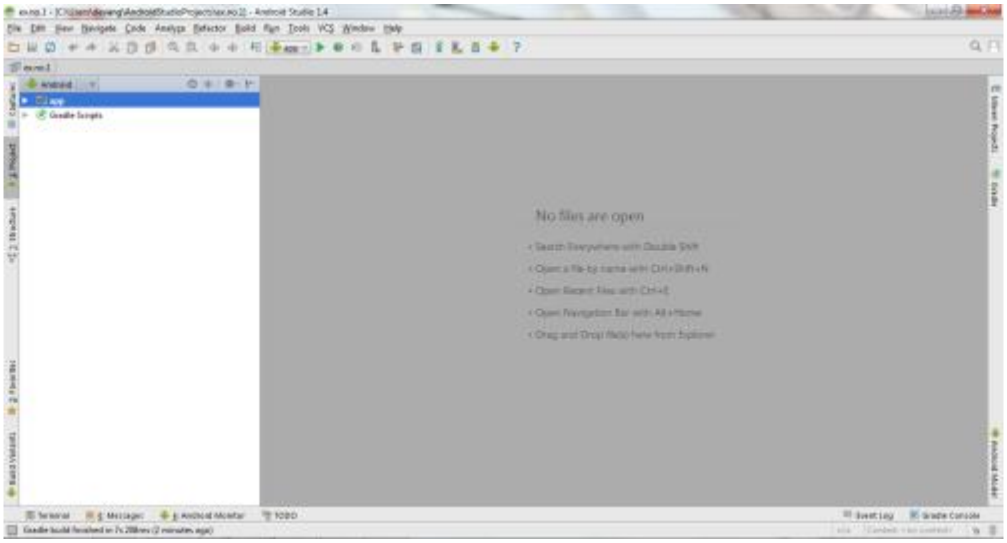
Creates a new empty activity

Activity Name: MainActivity

☑ Generate Layout File

Layout Name: activity_main

- It will take some time to build and load the project.
- After completion it will look as given below.



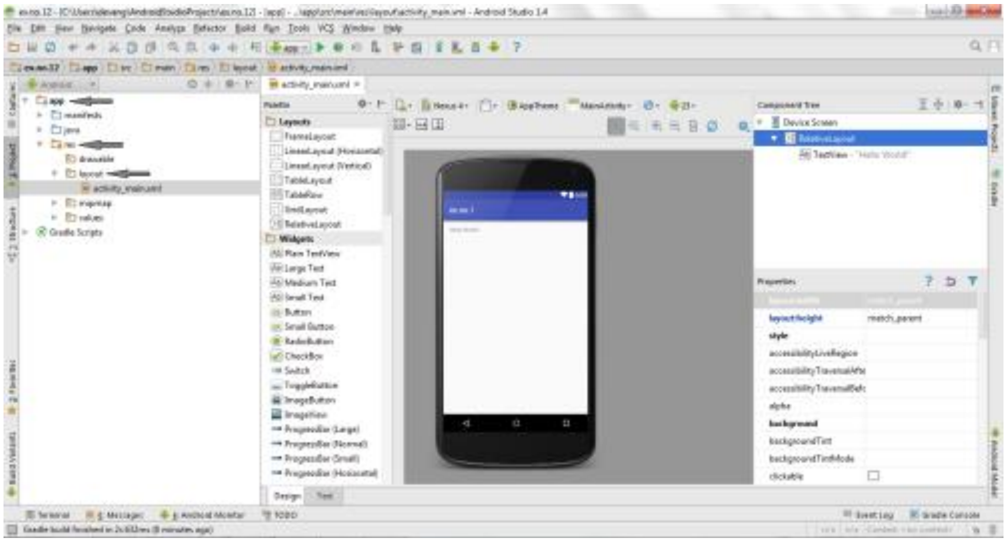
No files are open

- Search Everywhere with Double Shift
- Open a file by name with Ctrl+Shift+N
- Open Recent Files with Ctrl+E
- Open Navigation Bar with Alt+Home
- Drag and Drop file(s) here from Explorer

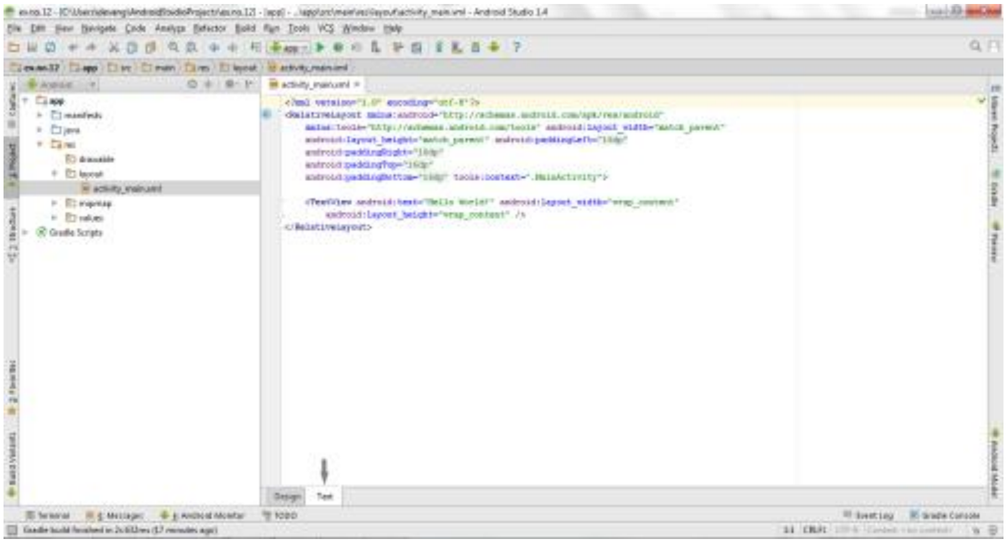
Gradle build finished in 7s 298ms (2 minutes ago)

Designing layout for the Android Application:

- Click on **app -> res -> layout -> activity_main.xml.**



- Now click on **Text** as shown below.



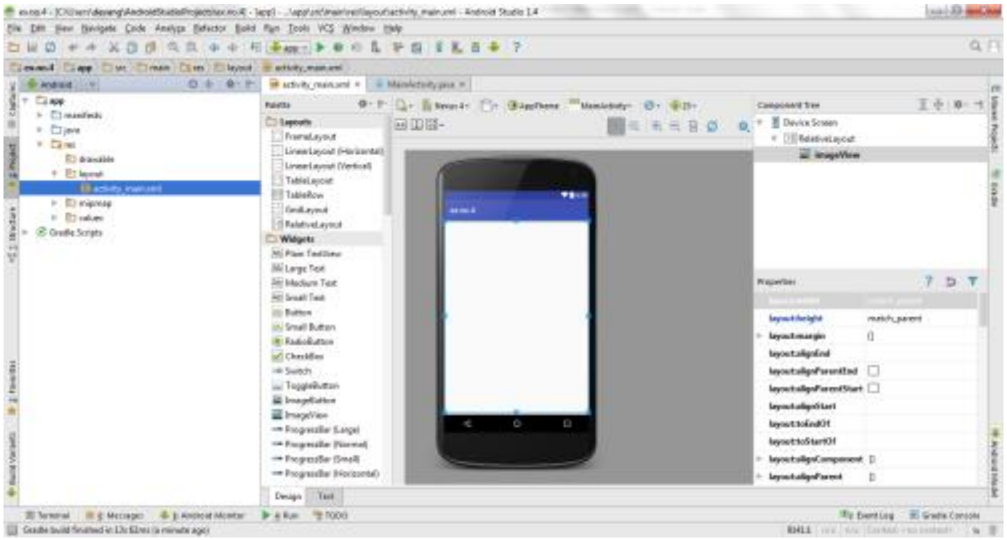
```
<?xml version="1.0" encoding="utf-8"?>
<RelativeLayout xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:tools="http://schemas.android.com/tools" android:layout_width="match_parent"
    android:layout_height="match_parent" android:paddingLeft="16dp"
    android:paddingRight="16dp"
    android:paddingTop="16dp"
    android:paddingBottom="16dp" tools:context=".MainActivity">

    <TextView android:text="Hello World!" android:layout_width="wrap_content"
        android:layout_height="wrap_content" />
</RelativeLayout>
```

- 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>
```

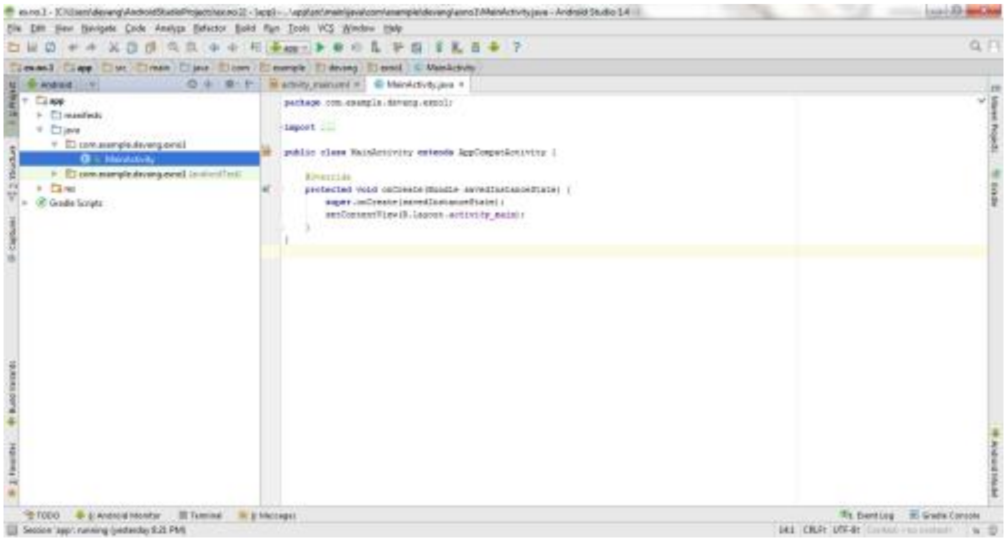
- Now click on **Design** and your application will look as given below.



- So now the designing part is completed.

Java Coding for the Android Application:

- Click on **app -> java -> com.example.exno4 -> MainActivity.**



```
package com.example.devang.exno1;

import ...

public class MainActivity extends AppCompatActivity {

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

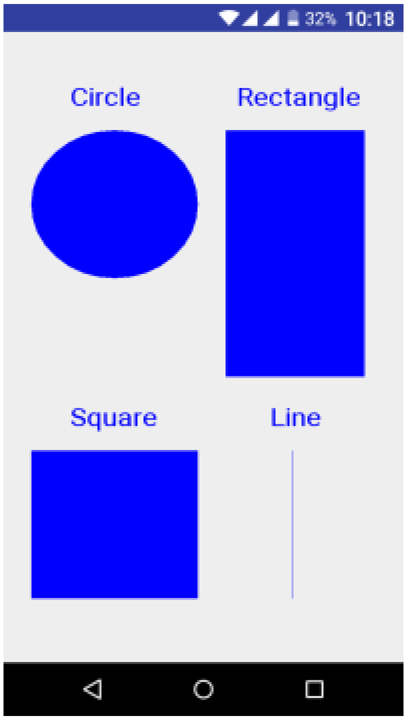
- Then delete the code which is there and type the code as given below.

Code for MainActivity.java:

```
package com.example.exno4;

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);

      //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);
   }
}
```

- So now the Coding part is also completed.
- Now run the application to see the output.

Output:



Result:

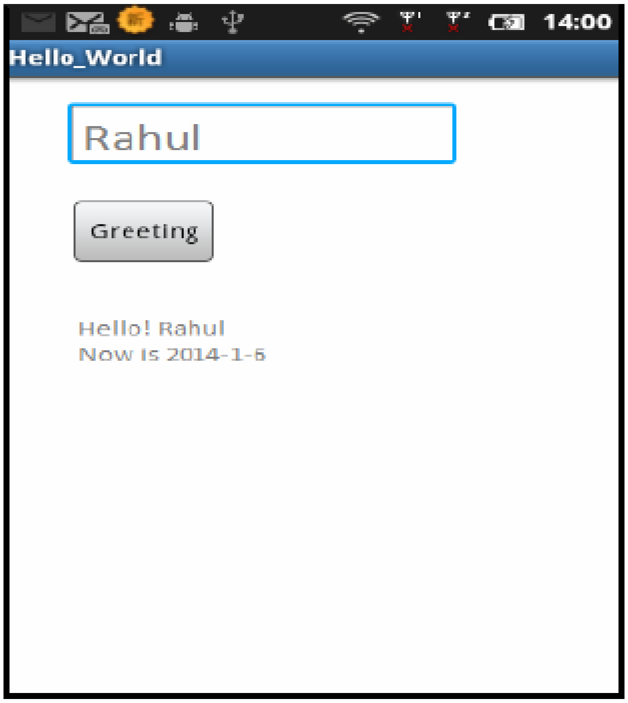
Thus a Simple Android Application that draws basic Graphical Primitives on the screen is developed and executed successfully.

Ex No: 5 Form design for mobile application

Aim: To create a mobile application that contains a form

Procedure: Refer lab manual Ex No. 2 (Page:14)

Output :



Rahul

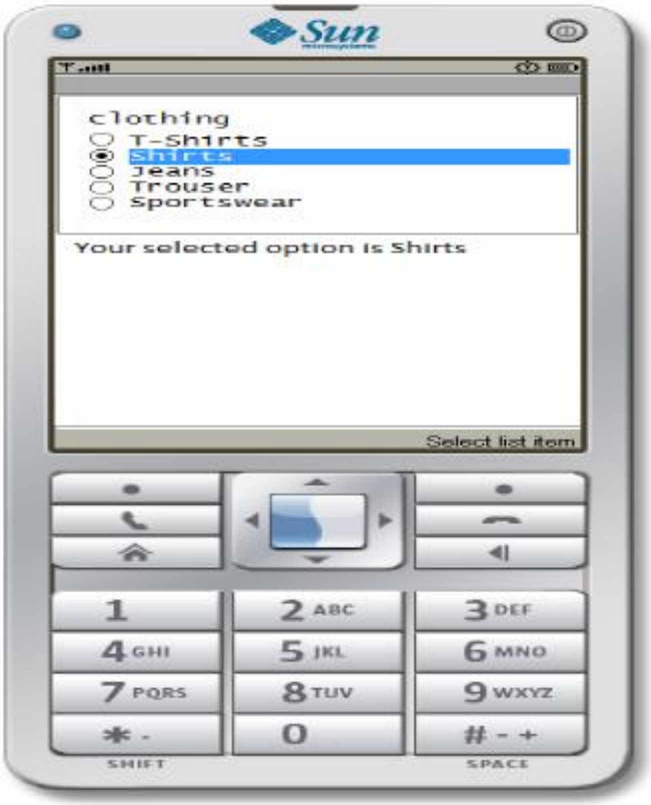
Hello! Rahul
Now is 2014-1-6

Ex No: 6 Application using controls

Aim: To create an application using some controls

Procedure: Refer lab manual Ex No:3 (Page No. 23)

Output:



clothing

○ T-Shirts
● Shirts
○ Jeans
○ Trouser
○ Sportswear

Your selected option is Shirts

Select list item