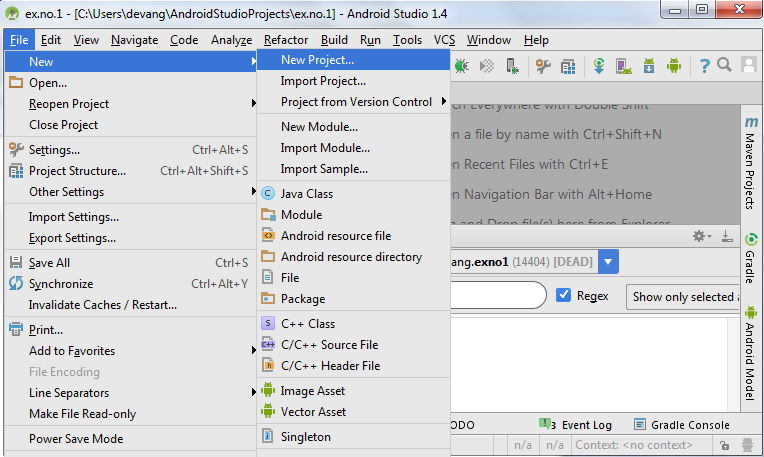
Android Application that uses GUI components, Font and Colors

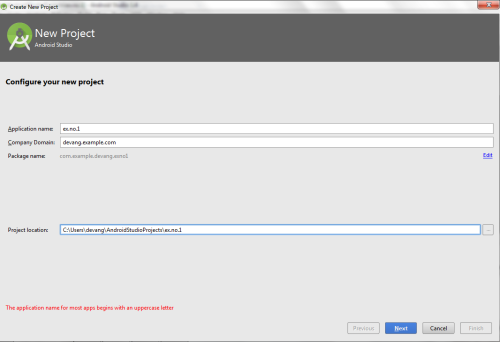
Procedure:

Creating a New project:

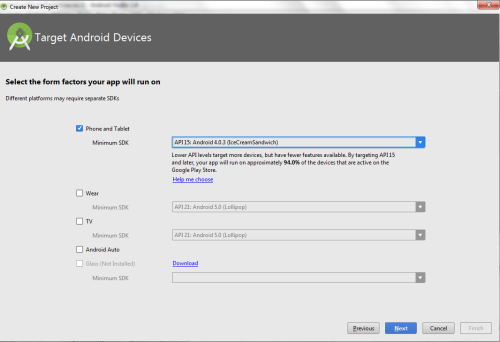
* Open Android Stdio and then click on **File -> New -> New project.**

[](https://codingconnect.net/wp-content/uploads/2016/02/new-project.png)

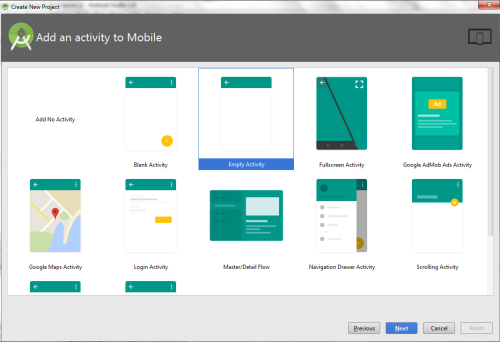
* Then type the Application name as “**ex.no.1″** and click **Next.**

[](https://codingconnect.net/wp-content/uploads/2016/02/application-name-e1456070044915.png)

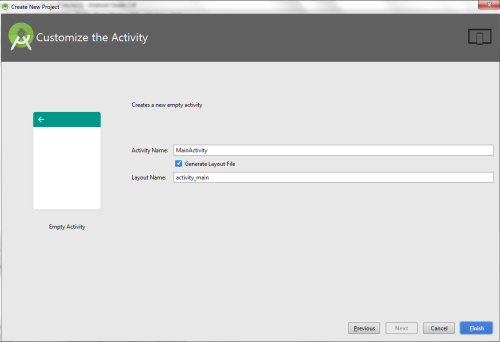
* Then select the **Minimum SDK** as shown below and click **Next**.

[](https://codingconnect.net/wp-content/uploads/2016/02/minimum-sdk-e1456069978476.png)

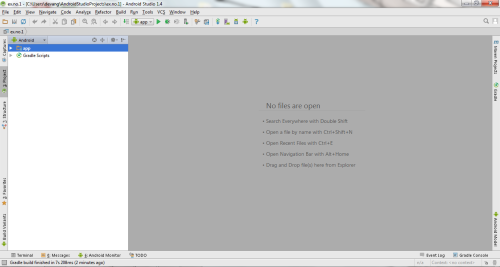
* Then select the **Empty Activity**and click **Next.**

[](https://codingconnect.net/wp-content/uploads/2016/02/empty-activity-e1456120797105.png)

* Finally click F**inish**.

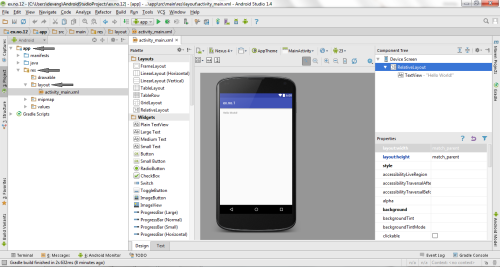
[](https://codingconnect.net/wp-content/uploads/2016/02/finish-e1456121463611.png)

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

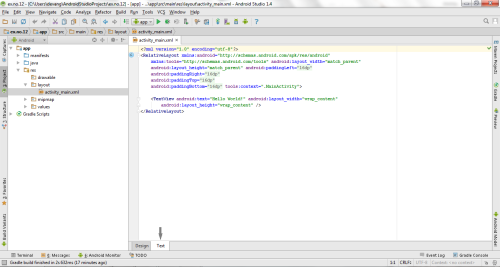
[](https://codingconnect.net/wp-content/uploads/2016/02/new-e1456122290334.png)

Designing layout for the Android Application:

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

[](https://codingconnect.net/wp-content/uploads/2016/02/activity_main-e1456123429693.png)

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

[](https://codingconnect.net/wp-content/uploads/2016/02/text-e1456124006902.png)

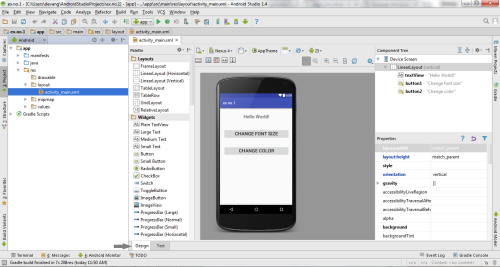
* Then delete the code which is there and type the code as given below.

**Code for Activity\_main.xml:**

[?](https://www.codingconnect.net/android-application-gui-components-font-and-colors/)

|  |  |
| --- | --- |
| 1  2  3  4  5  6  7  8  9  10  11  12  13  14  15  16  17  18  19  20  21  22  23  24  25  26  27  28  29  30  31  32  33 | <?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> |

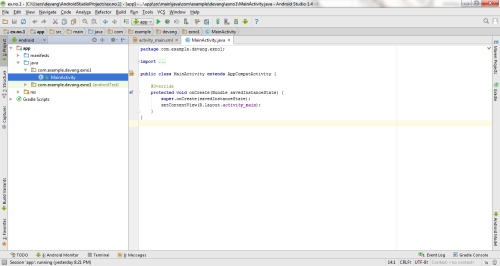
* Now click on Design and your application will look as given below.

[](https://codingconnect.net/wp-content/uploads/2016/02/design-e1456126325166.png)

* So now the designing part is completed.

Java Coding for the Android Application:

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

[](https://codingconnect.net/wp-content/uploads/2016/02/MainActivity-e1456597575355.png)

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

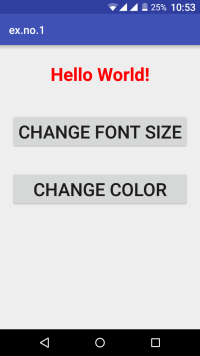
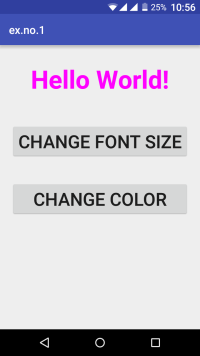
**Code for MainActivity.java:**

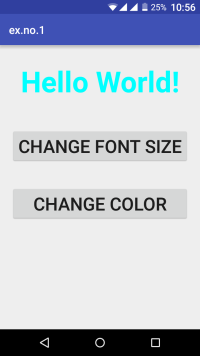
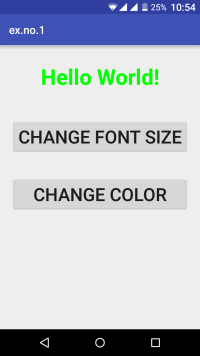
[?](https://www.codingconnect.net/android-application-gui-components-font-and-colors/)

|  |  |
| --- | --- |
| 1  2  3  4  5  6  7  8  9  10  11  12  13  14  15  16  17  18  19  20  21  22  23  24  25  26  27  28  29  30  31  32  33  34  35  36  37  38  39  40  41  42  43  44  45  46  47  48  49  50  51  52  53  54  55  56  57  58  59  60 | 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:

[](https://codingconnect.net/wp-content/uploads/2016/02/ex1-s1-e1456595860566.png)  [](https://codingconnect.net/wp-content/uploads/2016/02/ex1-s2-e1456595960660.png)  [](https://codingconnect.net/wp-content/uploads/2016/02/ex1-s7-e1456596189388.png)

[](https://codingconnect.net/wp-content/uploads/2016/02/ex1-s5-e1456596159472.png)  [](https://codingconnect.net/wp-content/uploads/2016/02/ex1-s4-e1456596139830.png)  [](https://codingconnect.net/wp-content/uploads/2016/02/ex1-s3-e1456596067470.png)

Result:

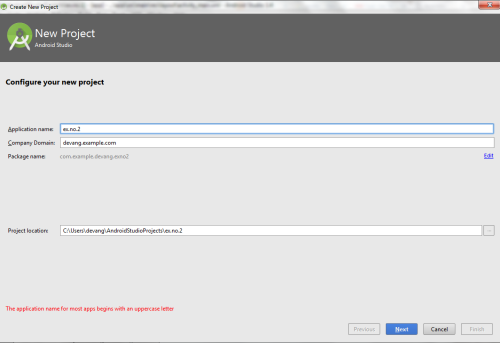
 Thus a Simple Android Application that uses GUI components, Font and Colors is developed and executed successfully.

**Android Application for Layout Managers and Event Listeners**

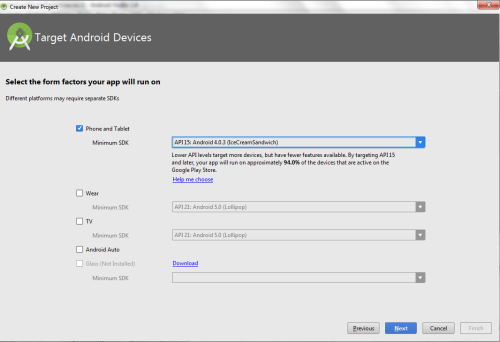
Procedure:

Creating a New project:

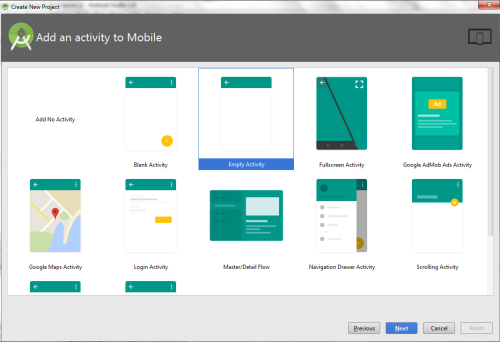
* Open Android Stdio and then click on **File -> New -> New project.**
* Then type the Application name as “**ex.no.2″** and click **Next.**

[](https://codingconnect.net/wp-content/uploads/2016/02/application-name-2-e1456664675131.png)

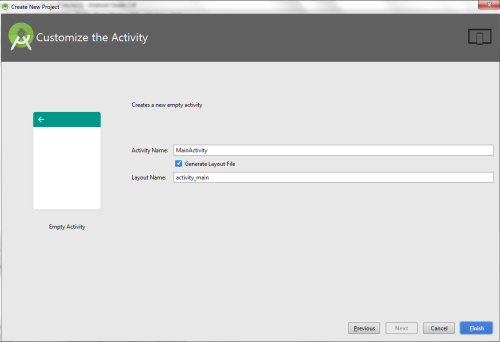
* Then select the **Minimum SDK** as shown below and click **Next**.

[](https://codingconnect.net/wp-content/uploads/2016/02/minimum-sdk-e1456069978476.png)

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

[](https://codingconnect.net/wp-content/uploads/2016/02/empty-activity-e1456120797105.png)

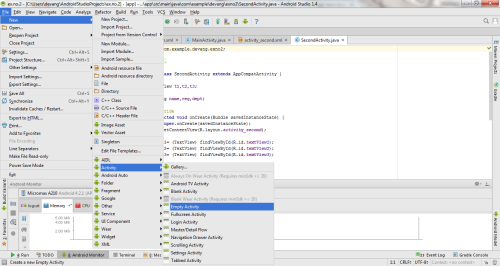
* Finally click F**inish**.

[](https://codingconnect.net/wp-content/uploads/2016/02/finish-e1456121463611.png)

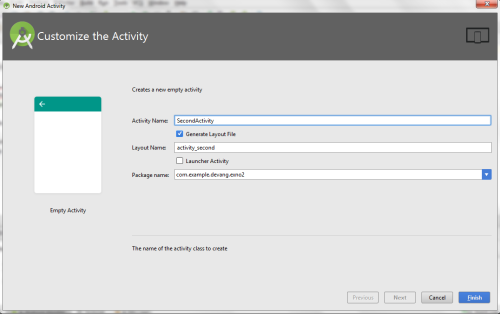
* It will take some time to build and load the project.
* After completion it will look as given below.

Creating Second Activity for the Android Application:

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

[](https://codingconnect.net/wp-content/uploads/2016/02/New-activity-e1456766876210.png)

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

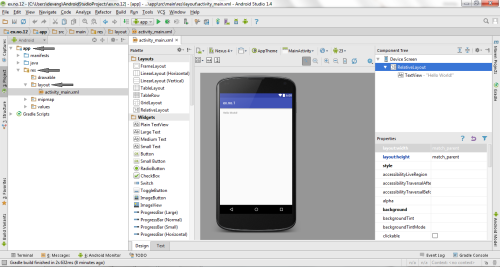
[](https://codingconnect.net/wp-content/uploads/2016/02/SecondActivity-e1456766896789.png)

* Thus Second Activity For the application is created.

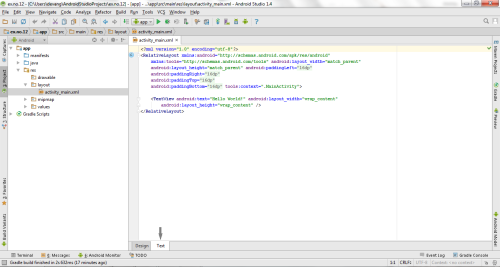
Designing layout for the Android Application:

Designing Layout for Main Activity:

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

[](https://codingconnect.net/wp-content/uploads/2016/02/activity_main-e1456123429693.png)

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

[](https://codingconnect.net/wp-content/uploads/2016/02/text-e1456124006902.png)

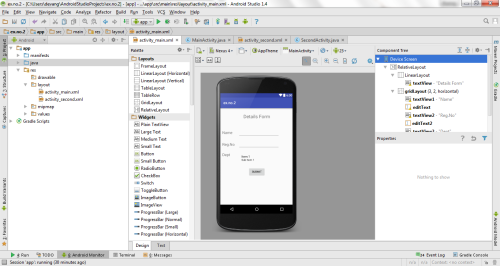
* Then delete the code which is there and type the code as given below.

**Code for Activity\_main.xml:**

[?](https://www.codingconnect.net/android-application-for-layout-managers-event-listners/)

|  |
| --- |
| <?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> |

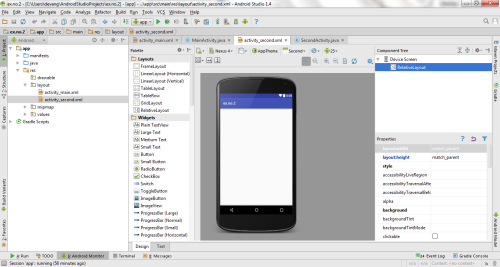
* Now click on Design and your activity will look as given below.

[](https://codingconnect.net/wp-content/uploads/2016/02/design-2a-e1456766487747.png)

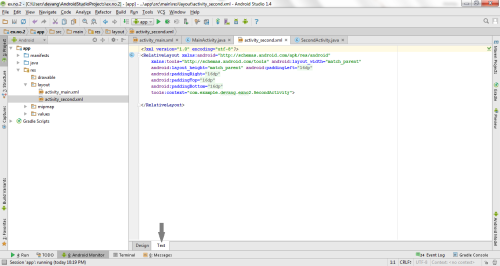
* So now the designing part of Main Activity is completed.

Designing Layout for Second Activity:

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

[](https://codingconnect.net/wp-content/uploads/2016/02/activity_second-e1456768173270.png)

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

[](https://codingconnect.net/wp-content/uploads/2016/02/text-2-e1456768346482.png)

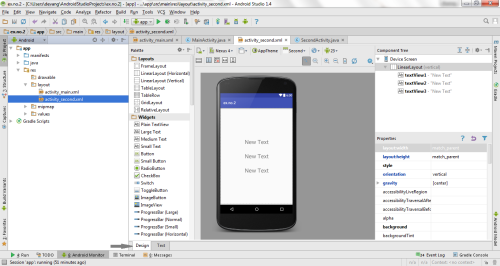
* Then delete the code which is there and type the code as given below.

**Code for Activity\_second.xml:**

[?](https://www.codingconnect.net/android-application-for-layout-managers-event-listners/)

|  |
| --- |
| <?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.

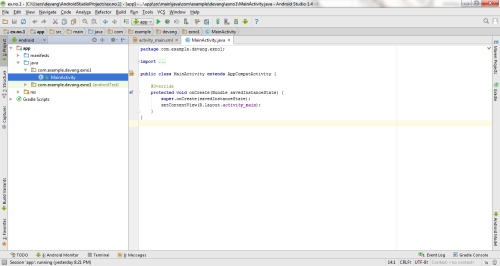
[](https://codingconnect.net/wp-content/uploads/2016/02/design-2b-e1456768479400.png)

* 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.**

[](https://codingconnect.net/wp-content/uploads/2016/02/MainActivity-e1456597575355.png)

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

**Code for MainActivity.java:**

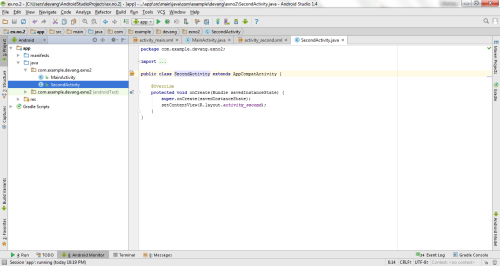
[?](https://www.codingconnect.net/android-application-for-layout-managers-event-listners/)

|  |
| --- |
| 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.**

[](https://codingconnect.net/wp-content/uploads/2016/02/SecondActivity-java-e1456769026596.png)

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

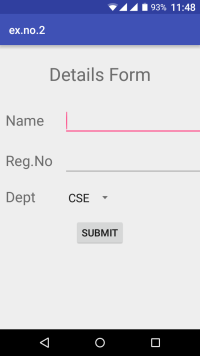
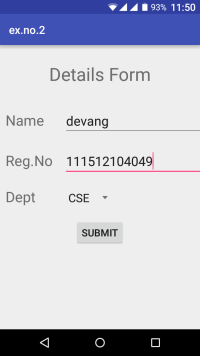
**Code for SecondActivity.java:**

[?](https://www.codingconnect.net/android-application-for-layout-managers-event-listners/)

|  |
| --- |
| 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:

[](https://codingconnect.net/wp-content/uploads/2016/02/Screenshot_2016-02-29-23-48-58-e1456770217201.png)  [](https://codingconnect.net/wp-content/uploads/2016/02/Screenshot_2016-02-29-23-50-31-e1456770206237.png)  [](https://codingconnect.net/wp-content/uploads/2016/02/Screenshot_2016-02-29-23-50-41-e1456770195455.png)

Result:

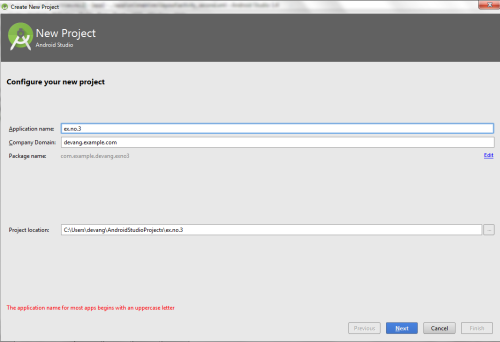
 Thus a Simple Android Application that uses Layout Managers and Event Listeners is developed and executed successfully.

**Simple Android Application for Native Calculator**

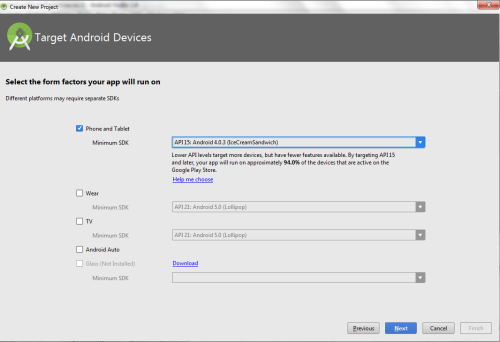
Procedure:

Creating a New project:

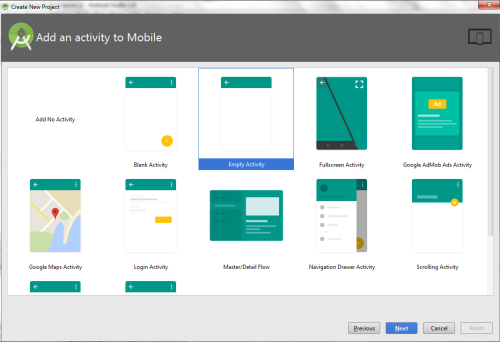
* Open Android Stdio and then click on **File -> New -> New project.**
* Then type the Application name as “**ex.no.3″** and click **Next.**

[](https://codingconnect.net/wp-content/uploads/2016/03/application-name-3-e1456851583870.png)

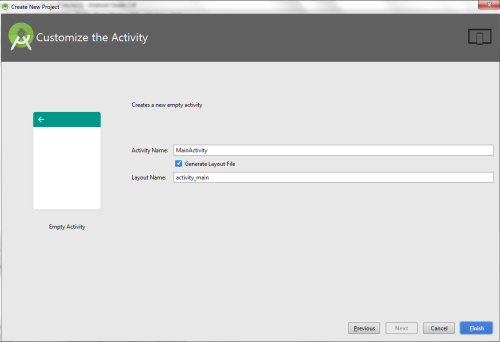
* Then select the **Minimum SDK** as shown below and click **Next**.

[](https://codingconnect.net/wp-content/uploads/2016/02/minimum-sdk-e1456069978476.png)

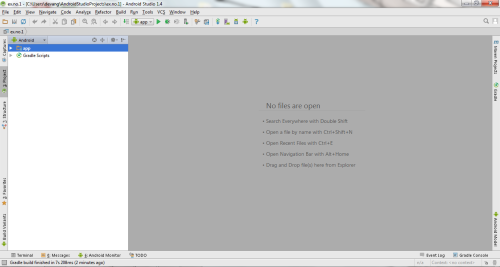
* Then select the **Empty Activity**and click **Next.**

[](https://codingconnect.net/wp-content/uploads/2016/02/empty-activity-e1456120797105.png)

* Finally click F**inish**.

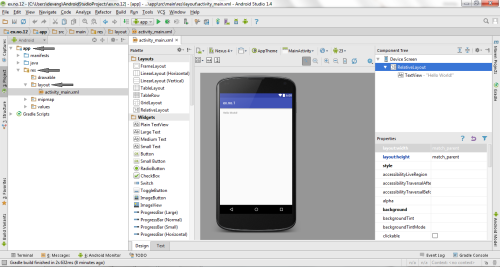
[](https://codingconnect.net/wp-content/uploads/2016/02/finish-e1456121463611.png)

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

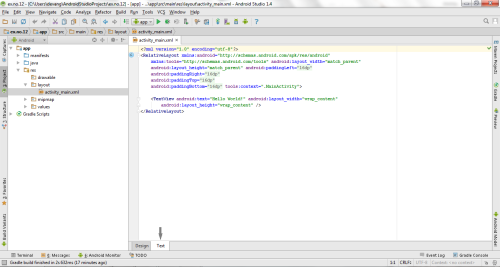
[](https://codingconnect.net/wp-content/uploads/2016/02/new-e1456122290334.png)

Designing layout for the Android Application:

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

[](https://codingconnect.net/wp-content/uploads/2016/02/activity_main-e1456123429693.png)

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

[](https://codingconnect.net/wp-content/uploads/2016/02/text-e1456124006902.png)

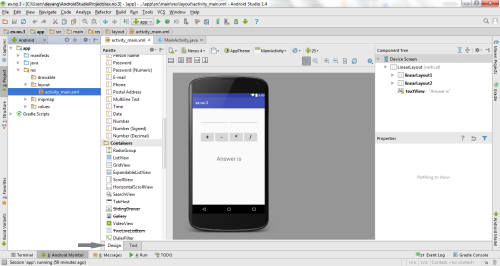
* Then delete the code which is there and type the code as given below.

**Code for Activity\_main.xml:**

[?](https://www.codingconnect.net/android-application-for-native-calculator/)

|  |
| --- |
| <?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> |

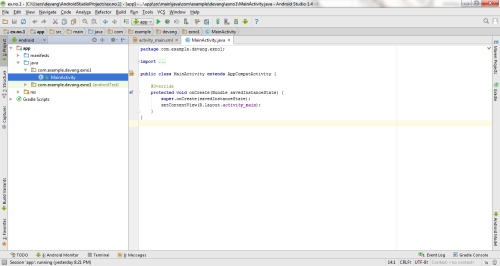
* Now click on Design and your application will look as given below.

[](https://codingconnect.net/wp-content/uploads/2016/03/design-3-e1456851967119.png)

* So now the designing part is completed.

Java Coding for the Android Application:

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

[](https://codingconnect.net/wp-content/uploads/2016/02/MainActivity-e1456597575355.png)

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

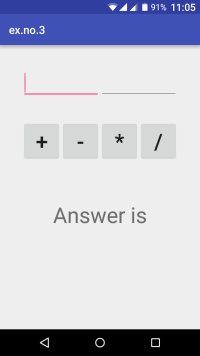
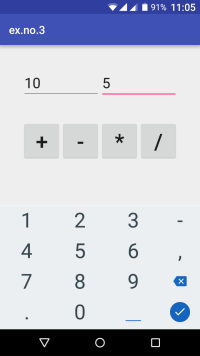
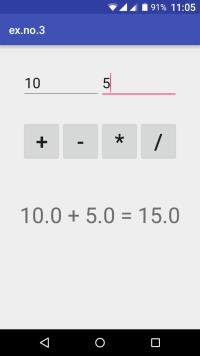
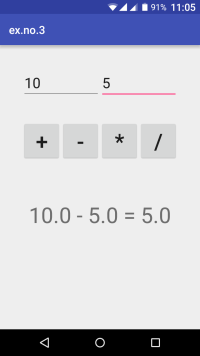
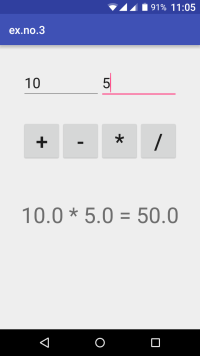
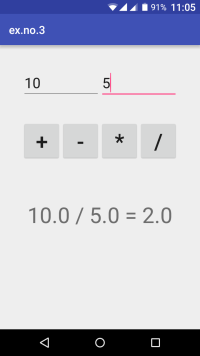
**Code for MainActivity.java:**

[?](https://www.codingconnect.net/android-application-for-native-calculator/)

|  |
| --- |
| 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.

Output:

[](https://codingconnect.net/wp-content/uploads/2016/03/Screenshot_2016-03-01-23-05-04-e1456854119583.png)  [](https://codingconnect.net/wp-content/uploads/2016/03/Screenshot_2016-03-01-23-05-18-e1456854175471.png)  [](https://codingconnect.net/wp-content/uploads/2016/03/Screenshot_2016-03-01-23-05-36-e1456854164331.png)  
   
[](https://codingconnect.net/wp-content/uploads/2016/03/Screenshot_2016-03-01-23-05-42-e1456854153461.png)  [](https://codingconnect.net/wp-content/uploads/2016/03/Screenshot_2016-03-01-23-05-47-e1456854141934.png)  [](https://codingconnect.net/wp-content/uploads/2016/03/Screenshot_2016-03-01-23-05-52-e1456854131402.png)

Result:

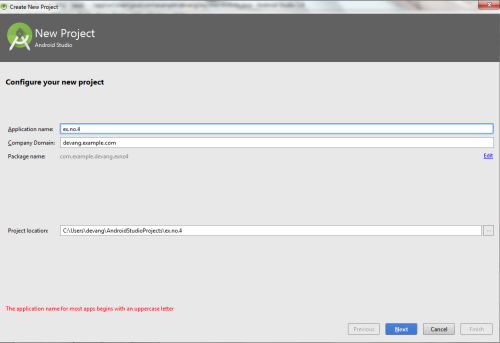
 Thus a Simple Android Application for Native Calculator is developed and executed successfully.

**Android Application to draw Basic Graphical Primitives**

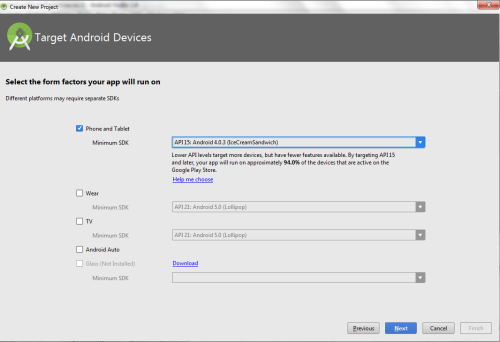
Procedure:

Creating a New project:

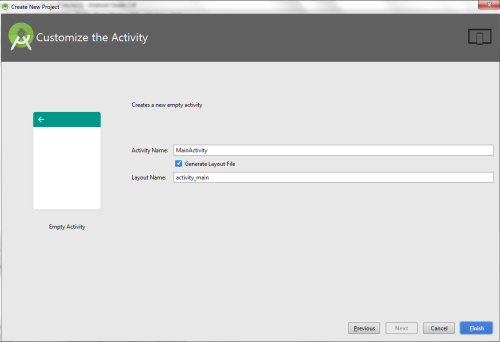
* Open Android Studio and then click on **File -> New -> New project.**
* Then type the Application name as “**ex.no.4″** and click **Next.**

[](https://codingconnect.net/wp-content/uploads/2016/03/application-name-4-e1457453961332.png)

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

[](https://codingconnect.net/wp-content/uploads/2016/02/minimum-sdk-e1456069978476.png)

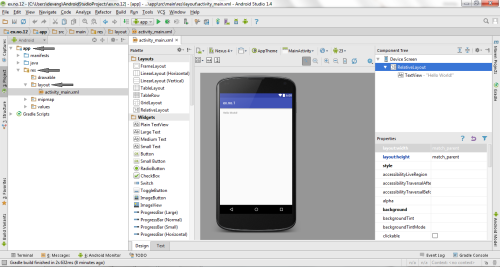
* Then select the **Empty Activity**and click **Next.**
* Finally click F**inish**.

[](https://codingconnect.net/wp-content/uploads/2016/02/finish-e1456121463611.png)

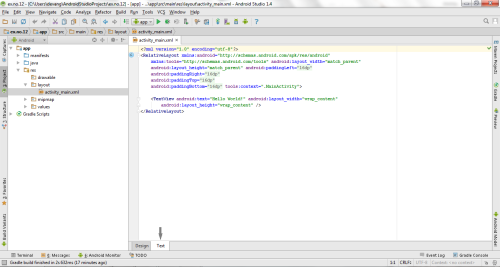
* It will take some time to build and load the project.
* After completion it will look as given below.

Designing layout for the Android Application:

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

[](https://codingconnect.net/wp-content/uploads/2016/02/activity_main-e1456123429693.png)

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

[](https://codingconnect.net/wp-content/uploads/2016/02/text-e1456124006902.png)

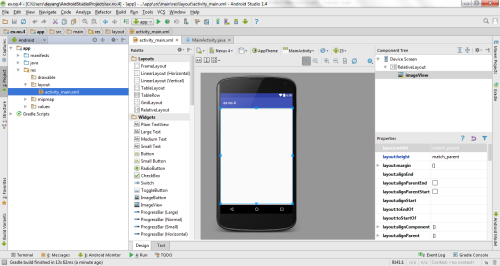
* Then delete the code which is there and type the code as given below.

**Code for Activity\_main.xml:**

[?](https://www.codingconnect.net/android-application-basic-graphical-primitives/)

|  |
| --- |
| <?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> |

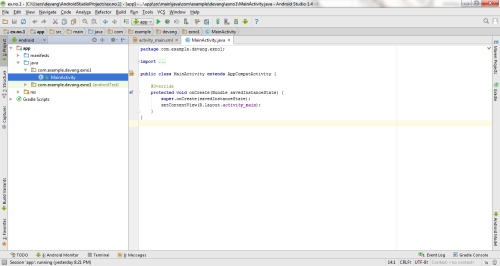
* Now click on **Design** and your application will look as given below.

[](https://codingconnect.net/wp-content/uploads/2016/03/design-4-e1457454769185.png)

* So now the designing part is completed.

Java Coding for the Android Application:

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

[](https://codingconnect.net/wp-content/uploads/2016/02/MainActivity-e1456597575355.png)

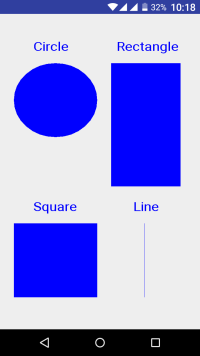
* Then delete the code which is there and type the code as given below.

**Code for MainActivity.java:**

|  |  |
| --- | --- |
| 1  2  3  4  5  6  7  8  9  10  11  12  13  14  15  16  17  18  19  20  21  22  23  24  25  26  27  28  29  30  31  32  33  34  35  36  37  38  39  40  41  42  43  44  45  46  47  48  49  50  51 | 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:

[](https://codingconnect.net/wp-content/uploads/2016/03/Screenshot_2016-03-08-22-18-18-e1457455792843.png)

Result:

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

**Simple Android Application that makes use of Database**

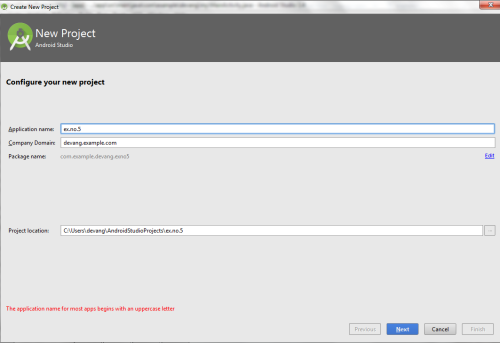
Aim:

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

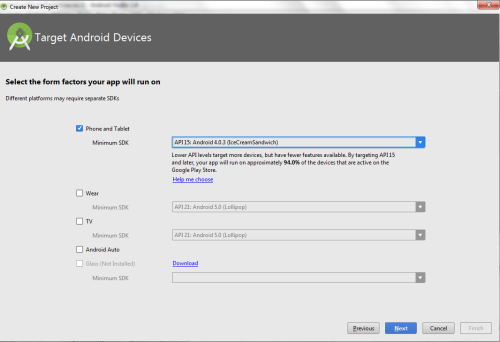
Procedure:

Creating a New project:

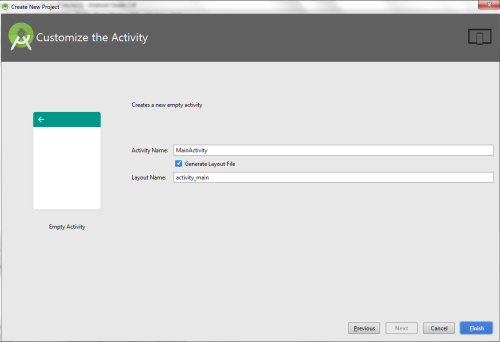
* Open Android Studio and then click on **File -> New -> New project.**
* Then type the Application name as “**ex.no.5″** and click **Next.**

[](https://codingconnect.net/wp-content/uploads/2016/03/application-name-5-e1457454785115.png)

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

[](https://codingconnect.net/wp-content/uploads/2016/02/minimum-sdk-e1456069978476.png)

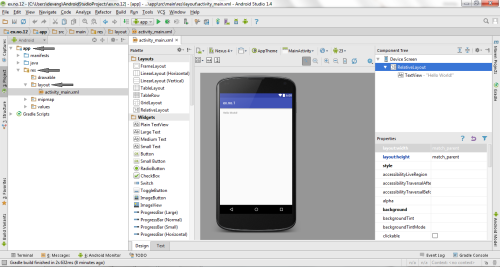
* Then select the **Empty Activity**and click **Next.**
* Finally click F**inish**.

[](https://codingconnect.net/wp-content/uploads/2016/02/finish-e1456121463611.png)

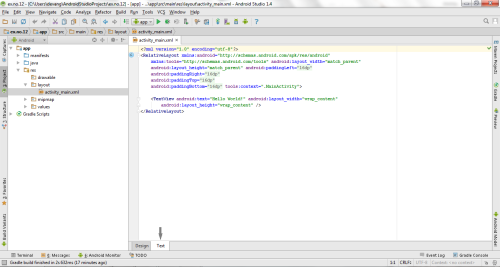
* It will take some time to build and load the project.
* After completion it will look as given below.

Designing layout for the Android Application:

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

[](https://codingconnect.net/wp-content/uploads/2016/02/activity_main-e1456123429693.png)

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

[](https://codingconnect.net/wp-content/uploads/2016/02/text-e1456124006902.png)

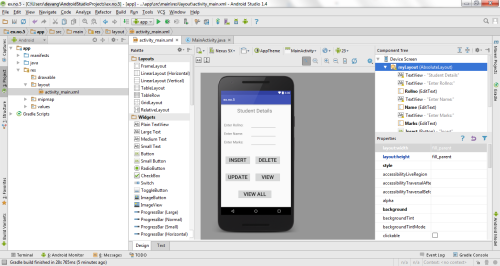
* Then delete the code which is there and type the code as given below.

**Code for Activity\_main.xml:**

[?](https://www.codingconnect.net/android-application-makes-use-database/)

|  |  |
| --- | --- |
| 1  2  3  4  5  6  7  8  9  10  11  12  13  14  15  16  17  18  19  20  21  22  23  24  25  26  27  28  29  30  31  32  33  34  35  36  37  38  39  40  41  42  43  44  45  46  47  48  49  50  51  52  53  54  55  56  57  58  59  60  61  62  63  64  65  66  67  68  69  70  71  72  73  74  75  76  77  78  79  80  81  82  83  84  85  86  87  88  89  90  91  92  93  94  95  96  97  98  99  100  101  102  103  104  105  106  107  108  109 | <?xml version="1.0" encoding="utf-8"?>  <AbsoluteLayout xmlns:android="http://schemas.android.com/apk/res/android"      android:layout\_width="match\_parent"      android:layout\_height="match\_parent">      <TextView          android:layout\_width="wrap\_content"          android:layout\_height="wrap\_content"          android:layout\_x="50dp"          android:layout\_y="20dp"          android:text="Student Details"          android:textSize="30sp" />        <TextView          android:layout\_width="wrap\_content"          android:layout\_height="wrap\_content"          android:layout\_x="20dp"          android:layout\_y="110dp"          android:text="Enter Rollno:"          android:textSize="20sp" />        <EditText          android:id="@+id/Rollno"          android:layout\_width="150dp"          android:layout\_height="wrap\_content"          android:layout\_x="175dp"          android:layout\_y="100dp"          android:inputType="number"          android:textSize="20sp" />        <TextView          android:layout\_width="wrap\_content"          android:layout\_height="wrap\_content"          android:layout\_x="20dp"          android:layout\_y="160dp"          android:text="Enter Name:"          android:textSize="20sp" />        <EditText          android:id="@+id/Name"          android:layout\_width="150dp"          android:layout\_height="wrap\_content"          android:layout\_x="175dp"          android:layout\_y="150dp"          android:inputType="text"          android:textSize="20sp" />        <TextView          android:layout\_width="wrap\_content"          android:layout\_height="wrap\_content"          android:layout\_x="20dp"          android:layout\_y="210dp"          android:text="Enter Marks:"          android:textSize="20sp" />        <EditText          android:id="@+id/Marks"          android:layout\_width="150dp"          android:layout\_height="wrap\_content"          android:layout\_x="175dp"          android:layout\_y="200dp"          android:inputType="number"          android:textSize="20sp" />        <Button          android:id="@+id/Insert"          android:layout\_width="150dp"          android:layout\_height="wrap\_content"          android:layout\_x="25dp"          android:layout\_y="300dp"          android:text="Insert"          android:textSize="30dp" />        <Button          android:id="@+id/Delete"          android:layout\_width="150dp"          android:layout\_height="wrap\_content"          android:layout\_x="200dp"          android:layout\_y="300dp"          android:text="Delete"          android:textSize="30dp" />        <Button          android:id="@+id/Update"          android:layout\_width="150dp"          android:layout\_height="wrap\_content"          android:layout\_x="25dp"          android:layout\_y="400dp"          android:text="Update"          android:textSize="30dp" />        <Button          android:id="@+id/View"          android:layout\_width="150dp"          android:layout\_height="wrap\_content"          android:layout\_x="200dp"          android:layout\_y="400dp"          android:text="View"          android:textSize="30dp" />        <Button          android:id="@+id/ViewAll"          android:layout\_width="200dp"          android:layout\_height="wrap\_content"          android:layout\_x="100dp"          android:layout\_y="500dp"          android:text="View All"          android:textSize="30dp" />    </AbsoluteLayout> |

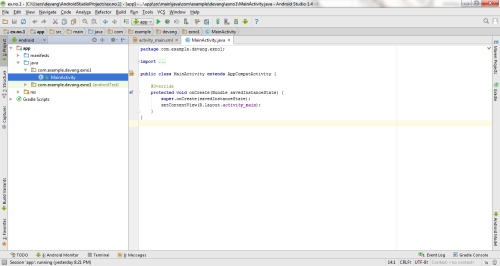
* Now click on **Design** and your application will look as given below.

[](https://codingconnect.net/wp-content/uploads/2016/03/design-5-e1457458832682.png)

* So now the designing part is completed.

Java Coding for the Android Application:

* Click on **app -> java -> com.example.exno5 -> MainActivity.**

[](https://codingconnect.net/wp-content/uploads/2016/02/MainActivity-e1456597575355.png)

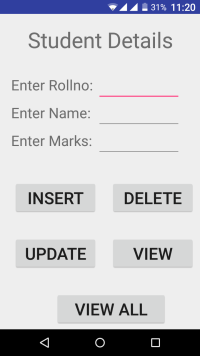
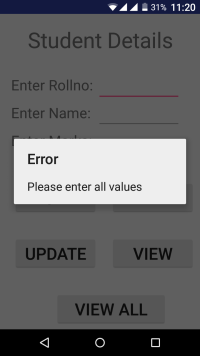
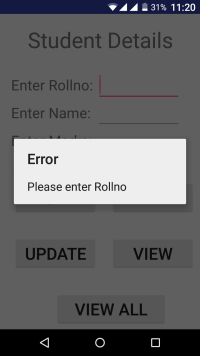
* Then delete the code which is there and type the code as given below.

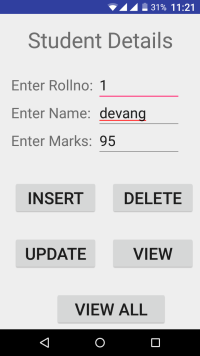
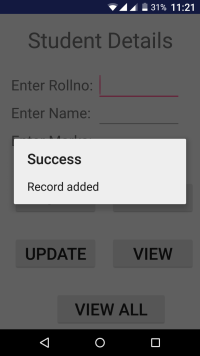
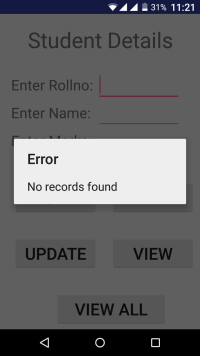
**Code for MainActivity.java:**

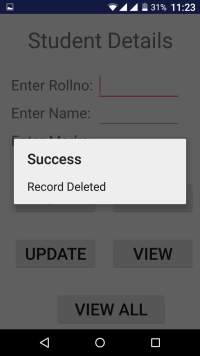
|  |
| --- |
| package com.example.exno5;    import android.app.Activity;  import android.app.AlertDialog.Builder;  import android.content.Context;  import android.database.Cursor;  import android.database.sqlite.SQLiteDatabase;  import android.os.Bundle;  import android.view.View;  import android.view.View.OnClickListener;  import android.widget.Button;  import android.widget.EditText;    public class MainActivity extends Activity implements OnClickListener  {      EditText Rollno,Name,Marks;      Button Insert,Delete,Update,View,ViewAll;      SQLiteDatabase db;      /\*\* 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(rollno VARCHAR,name VARCHAR,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();      }  } |

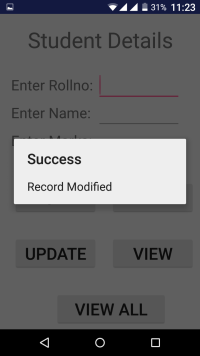
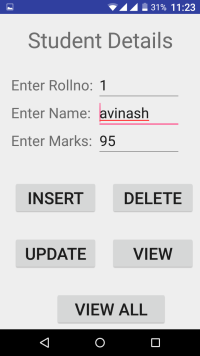
* So now the Coding part is also completed.
* Now run the application to see the output.

Output:

[](https://codingconnect.net/wp-content/uploads/2016/03/Screenshot_2016-03-08-23-20-15-e1457459858916.png)  [](https://codingconnect.net/wp-content/uploads/2016/03/Screenshot_2016-03-08-23-20-35-e1457460034257.png)  [](https://codingconnect.net/wp-content/uploads/2016/03/Screenshot_2016-03-08-23-20-47-e1457460023839.png)

[](https://codingconnect.net/wp-content/uploads/2016/03/Screenshot_2016-03-08-23-21-55-e1457459941204.png)  [](https://codingconnect.net/wp-content/uploads/2016/03/Screenshot_2016-03-08-23-21-33-e1457459970749.png)  [](https://codingconnect.net/wp-content/uploads/2016/03/Screenshot_2016-03-08-23-21-03-e1457460010510.png)

[](https://codingconnect.net/wp-content/uploads/2016/03/Screenshot_2016-03-08-23-22-56-e1457460189888.png)  [](https://codingconnect.net/wp-content/uploads/2016/03/Screenshot_2016-03-08-23-23-51-e1457460229382.png)  [](https://codingconnect.net/wp-content/uploads/2016/03/Screenshot_2016-03-08-23-23-56-e1457460274877.png)

[](https://codingconnect.net/wp-content/uploads/2016/03/Screenshot_2016-03-08-23-23-26-e1457460299942.png)  [](https://codingconnect.net/wp-content/uploads/2016/03/Screenshot_2016-03-08-23-23-44-e1457460323870.png)

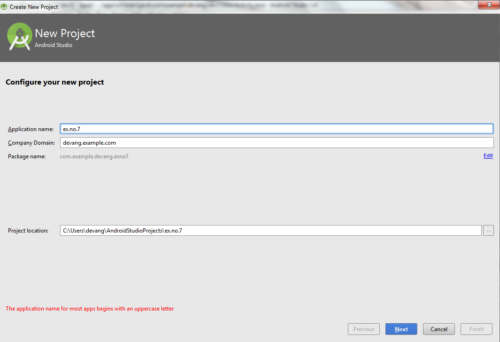
Result:

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

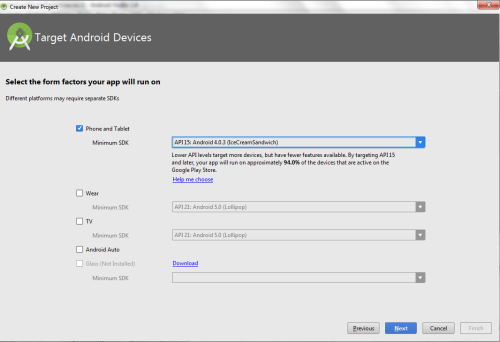
**Android Application that implements Multi threading**

Creating a New project:

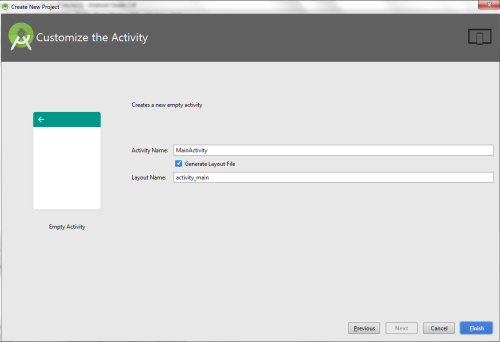
* Open Android Studio and then click on **File -> New -> New project.**
* Then type the Application name as “**ex.no.7″** and click **Next.**

[](https://codingconnect.net/wp-content/uploads/2016/04/application-name-7-e1461512099973.png)

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

[](https://codingconnect.net/wp-content/uploads/2016/02/minimum-sdk-e1456069978476.png)

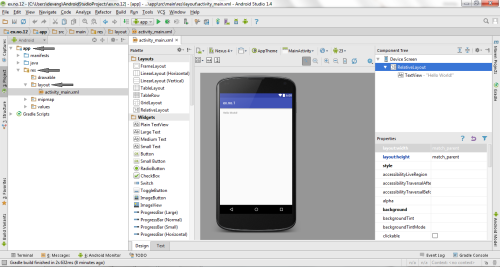
* Then select the **Empty Activity**and click **Next.**
* Finally click F**inish**.

[](https://codingconnect.net/wp-content/uploads/2016/02/finish-e1456121463611.png)

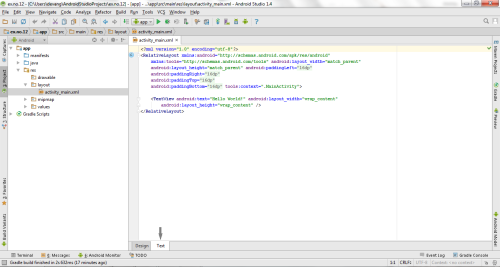
* It will take some time to build and load the project.
* After completion it will look as given below.

Designing layout for the Android Application:

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

[](https://codingconnect.net/wp-content/uploads/2016/02/activity_main-e1456123429693.png)

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

[](https://codingconnect.net/wp-content/uploads/2016/02/text-e1456124006902.png)

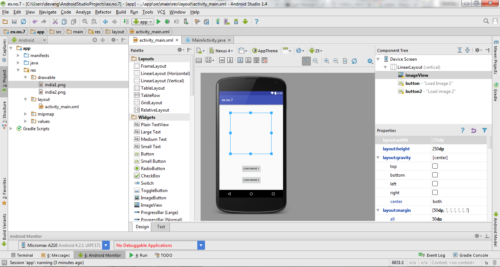
* Then delete the code which is there and type the code as given below.

**Code for Activity\_main.xml:**

[?](https://www.codingconnect.net/android-application-multi-threading/)

|  |  |
| --- | --- |
| 1  2  3  4  5  6  7  8  9  10  11  12  13  14  15  16  17  18  19  20  21  22  23  24  25  26  27  28  29  30 | <?xml version="1.0" encoding="utf-8"?>  <LinearLayout xmlns:android="http://schemas.android.com/apk/res/android"      android:layout\_width="match\_parent"      android:layout\_height="match\_parent"      android:orientation="vertical" >        <ImageView          android:id="@+id/imageView"          android:layout\_width="250dp"          android:layout\_height="250dp"          android:layout\_margin="50dp"          android:layout\_gravity="center" />        <Button          android:id="@+id/button"          android:layout\_width="wrap\_content"          android:layout\_height="wrap\_content"          android:layout\_margin="10dp"          android:layout\_gravity="center"          android:text="Load Image 1" />        <Button          android:id="@+id/button2"          android:layout\_width="wrap\_content"          android:layout\_height="wrap\_content"          android:layout\_margin="10dp"          android:layout\_gravity="center"          android:text="Load image 2" />    </LinearLayout> |

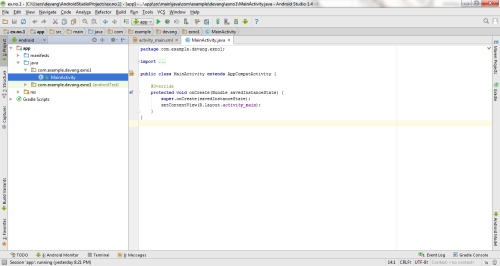
* Now click on **Design** and your application will look as given below.

[](https://codingconnect.net/wp-content/uploads/2016/04/design-7-e1461572193954.png)

* So now the designing part is completed.

Java Coding for the Android Application:

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

[](https://codingconnect.net/wp-content/uploads/2016/02/MainActivity-e1456597575355.png)

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

**Code for MainActivity.java:**

[?](https://www.codingconnect.net/android-application-multi-threading/)

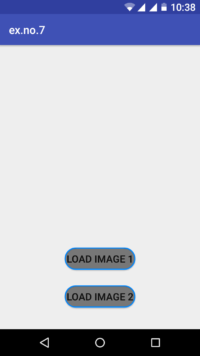
|  |  |
| --- | --- |
| 1  2  3  4  5  6  7  8  9  10  11  12  13  14  15  16  17  18  19  20  21  22  23  24  25  26  27  28  29  30  31  32  33  34  35  36  37  38  39  40  41  42  43  44  45  46  47  48  49  50  51  52  53  54  55  56  57  58  59  60  61  62  63  64  65  66  67  68 | package com.example.exno7;    import android.os.Bundle;  import android.support.v7.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();              }          });      }  } |

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

**Note:** **Before Running the Application, Copy the Images given below and Paste it in*“app -> res -> drawable”*by pressing “right click mouse button on *drawable*” and selecting the “*Paste*” option.**

**TO DOWNLOAD THE IMAGES :**[**CLICK HERE**](https://codingconnect.net/wp-content/uploads/2017/02/drawable.zip)

Output:

[](https://codingconnect.net/wp-content/uploads/2017/02/Screenshot_2017-02-19-22-38-59.png)  [](https://codingconnect.net/wp-content/uploads/2017/02/Screenshot_2017-02-19-22-44-10.png)  [](https://codingconnect.net/wp-content/uploads/2017/02/Screenshot_2017-02-19-22-44-28.png)

Result:

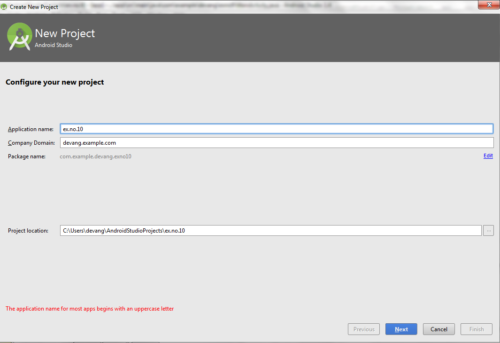
 Thus Android Application that implements Multi threading is developed and executed successfully.

**Android Application that creates an alert upon receiving a message**

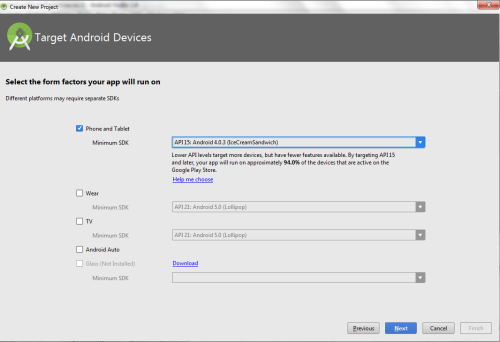
Procedure:

Creating a New project:

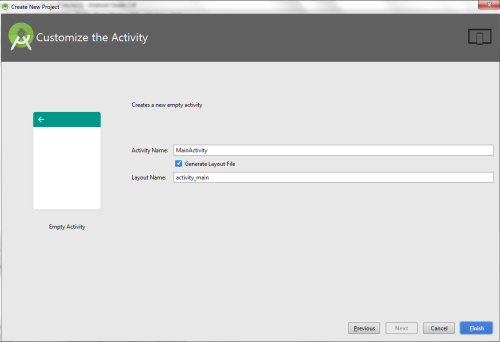
* Open Android Studio and then click on **File -> New -> New project.**
* Then type the Application name as “**ex.no.10″** and click **Next.**

[](https://codingconnect.net/wp-content/uploads/2016/04/application-name-10-e1461085361545.png)

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

[](https://codingconnect.net/wp-content/uploads/2016/02/minimum-sdk-e1456069978476.png)

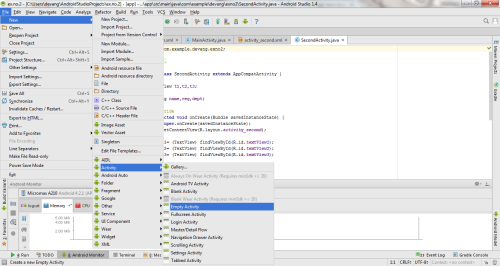
* Then select the **Empty Activity**and click **Next.**
* Finally click F**inish**.

[](https://codingconnect.net/wp-content/uploads/2016/02/finish-e1456121463611.png)

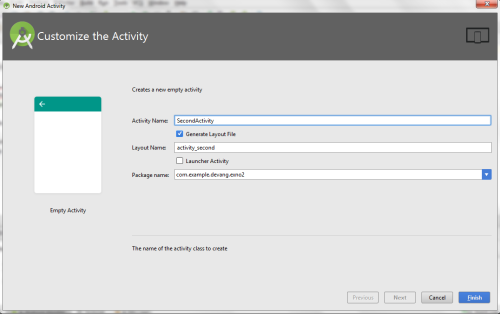
* It will take some time to build and load the project.
* After completion it will look as given below.

Creating Second Activity for the Android Application:

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

[](https://codingconnect.net/wp-content/uploads/2016/02/New-activity-e1456766876210.png)

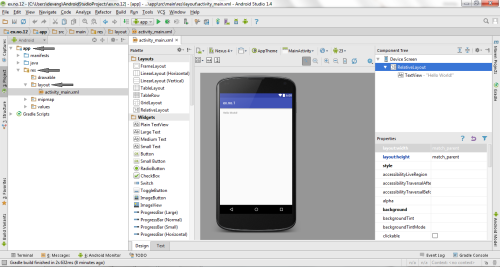
* Type the Activity Name as **SecondActivity**and click **Finish** button.

[](https://codingconnect.net/wp-content/uploads/2016/02/SecondActivity-e1456766896789.png)

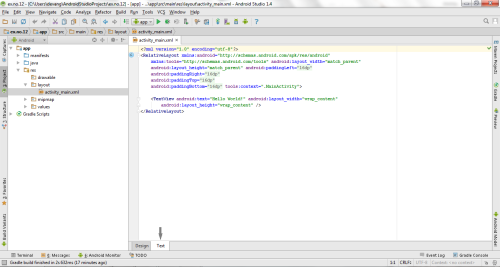
* Thus Second Activity For the application is created.

Designing layout for the Android Application:

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

[](https://codingconnect.net/wp-content/uploads/2016/02/activity_main-e1456123429693.png)

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

[](https://codingconnect.net/wp-content/uploads/2016/02/text-e1456124006902.png)

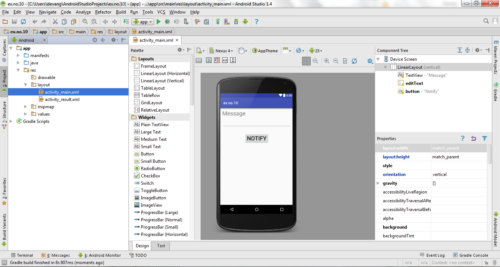
* Then delete the code which is there and type the code as given below.

**Code for Activity\_main.xml:**

[?](https://www.codingconnect.net/android-application-create-alert-message/)

|  |  |
| --- | --- |
| 1  2  3  4  5  6  7  8  9  10  11  12  13  14  15  16  17  18  19  20  21  22  23  24  25  26  27  28  29  30 | <?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"          android:textSize="30sp" />        <EditText          android:id="@+id/editText"          android:layout\_width="match\_parent"          android:layout\_height="wrap\_content"          android:singleLine="true"          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> |

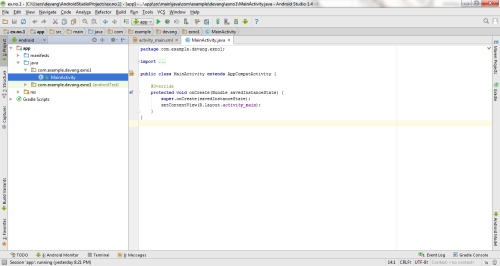
* Now click on **Design** and your application will look as given below.

[](https://codingconnect.net/wp-content/uploads/2016/04/design-10-e1461085567516.png)

* So now the designing part is completed.

Java Coding for the Android Application:

* Click on **app -> java -> com.example.exno10 -> MainActivity.**

[](https://codingconnect.net/wp-content/uploads/2016/02/MainActivity-e1456597575355.png)

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

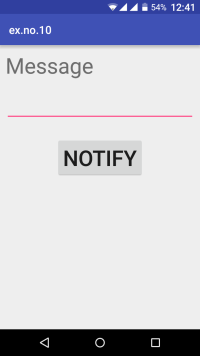
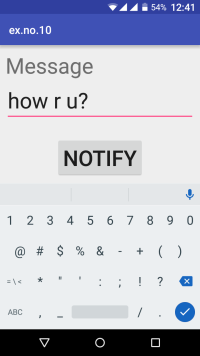
**Code for MainActivity.java:**

[?](https://www.codingconnect.net/android-application-create-alert-message/)

|  |  |
| --- | --- |
| 1  2  3  4  5  6  7  8  9  10  11  12  13  14  15  16  17  18  19  20  21  22  23  24  25  26  27  28  29  30  31  32  33  34  35  36  37  38  39  40 | package com.example.exno10;    import android.app.Notification;  import android.app.NotificationManager;  import android.app.PendingIntent;  import android.content.Intent;  import android.os.Bundle;  import android.support.v7.app.AppCompatActivity;  import android.view.View;  import android.widget.Button;  import android.widget.EditText;    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, SecondActivity.class);                  PendingIntent pending = PendingIntent.getActivity(MainActivity.this, 0, intent, 0);                  Notification noti = new Notification.Builder(MainActivity.this).setContentTitle("New Message")  .setContentText(e.getText().toString()).setSmallIcon(R.mipmap.ic\_launcher).setContentIntent(pending).build();                  NotificationManager manager = (NotificationManager) getSystemService(NOTIFICATION\_SERVICE);                  noti.flags |= Notification.FLAG\_AUTO\_CANCEL;                  manager.notify(0, noti);              }          });      }  } |

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

Output:

[[](https://codingconnect.net/wp-content/uploads/2016/03/Screenshot_2016-03-16-00-41-10-e1458112466211.png)](https://codingconnect.net/wp-content/uploads/2016/03/Screenshot_2016-03-16-00-41-10-e1458112466211.png)[](https://codingconnect.net/wp-content/uploads/2016/03/Screenshot_2016-03-16-00-41-32-e1458112448635.png)

[[](https://codingconnect.net/wp-content/uploads/2016/03/Screenshot_2016-03-16-00-41-44-e1458112387180.png)](https://codingconnect.net/wp-content/uploads/2016/03/Screenshot_2016-03-16-00-41-44-e1458112387180.png)[](https://codingconnect.net/wp-content/uploads/2016/03/Screenshot_2016-03-16-00-41-53-e1458112335383.png)

Result:

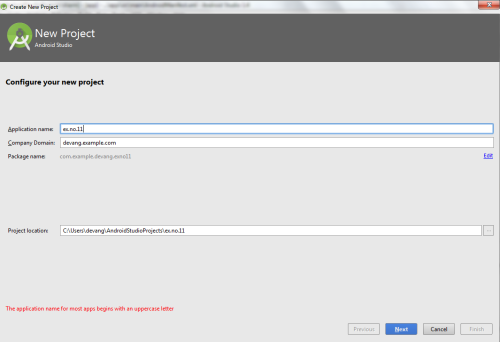
 Thus Android Application that creates an alert upon receiving a message is developed and executed successfully.

**Android Application that creates Alarm Clock**

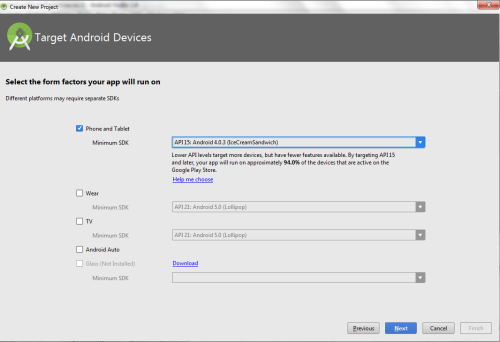
Procedure:

Creating a New project:

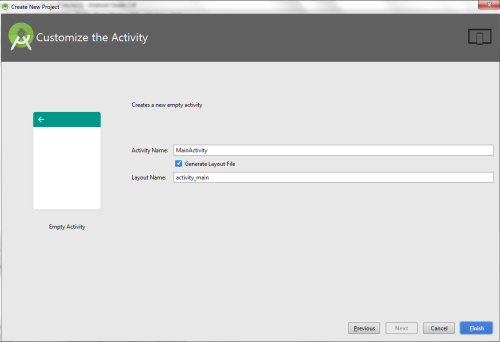
* Open Android Studio and then click on **File -> New -> New project.**
* Then type the Application name as “**ex.no.11″** and click **Next.**

[](https://codingconnect.net/wp-content/uploads/2016/04/application-name-11-e1459702276672.png)

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

[](https://codingconnect.net/wp-content/uploads/2016/02/minimum-sdk-e1456069978476.png)

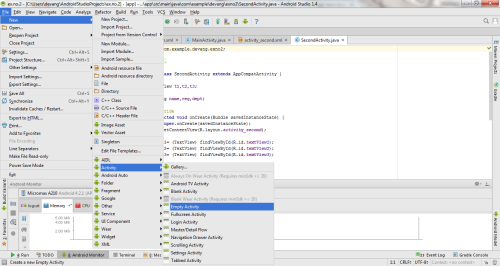
* Then select the **Empty Activity**and click **Next.**
* Finally click F**inish**.

[](https://codingconnect.net/wp-content/uploads/2016/02/finish-e1456121463611.png)

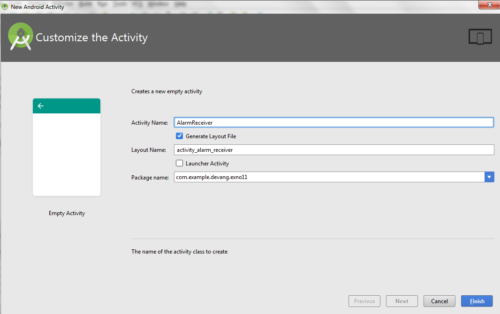
* It will take some time to build and load the project.
* After completion it will look as given below.

Creating Second Activity for the Android Application:

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

[](https://codingconnect.net/wp-content/uploads/2016/02/New-activity-e1456766876210.png)

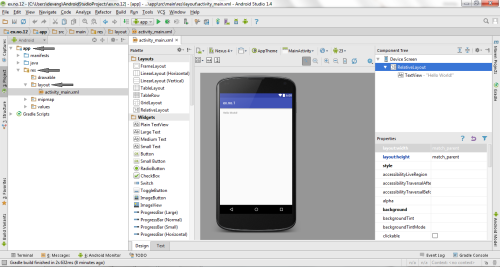
* Type the Activity Name as **AlarmReceiver**and click **Finish** button.

[](https://codingconnect.net/wp-content/uploads/2016/04/AlarmActivity-e1461605395201.png)

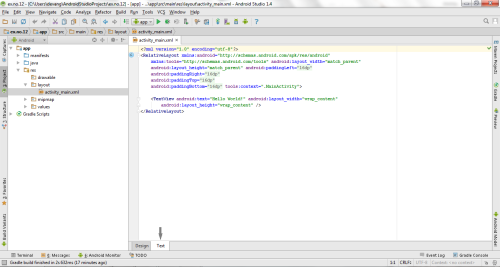
* Thus Second Activity For the application is created.

Designing layout for the Android Application:

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

[](https://codingconnect.net/wp-content/uploads/2016/02/activity_main-e1456123429693.png)

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

[](https://codingconnect.net/wp-content/uploads/2016/02/text-e1456124006902.png)

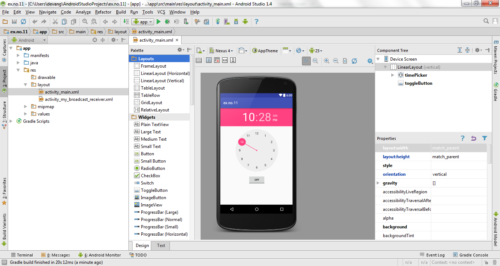
* Then delete the code which is there and type the code as given below.

**Code for Activity\_main.xml:**

[?](https://www.codingconnect.net/android-application-creates-alarm-clock/)

|  |  |
| --- | --- |
| 1  2  3  4  5  6  7  8  9  10  11  12  13  14  15  16  17  18  19  20  21  22 | <?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">        <TimePicker          android:id="@+id/timePicker"          android:layout\_width="wrap\_content"          android:layout\_height="wrap\_content"          android:layout\_gravity="center" />        <ToggleButton          android:id="@+id/toggleButton"          android:layout\_width="wrap\_content"          android:layout\_height="wrap\_content"          android:layout\_gravity="center"          android:layout\_margin="20dp"          android:checked="false"          android:onClick="OnToggleClicked" />    </LinearLayout> |

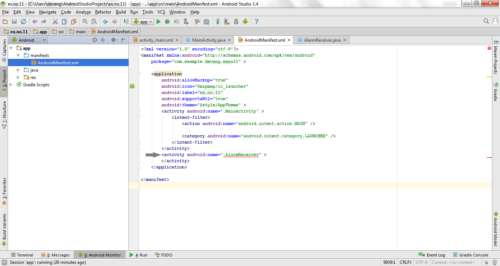
* Now click on **Design** and your application will look as given below.

[](https://codingconnect.net/wp-content/uploads/2016/04/design-11-e1461603622148.png)

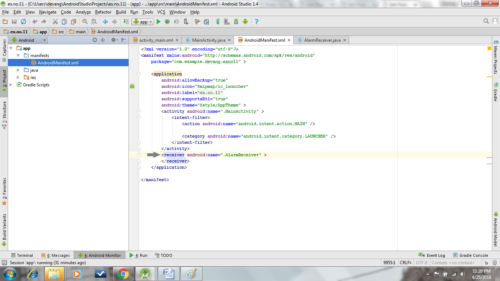
* So now the designing part is completed.

Changes in Manifest for the Android Application:

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

[](https://codingconnect.net/wp-content/uploads/2016/04/manifest-11-a-e1461606529961.png)

* Now change the **activity tag** to **receiver tag** in the AndroidManifest.xml file as shown below

[](https://codingconnect.net/wp-content/uploads/2016/04/manifest-11-e1461606723102.png)

**Code for AndroidManifest.xml:**

[?](https://www.codingconnect.net/android-application-creates-alarm-clock/)

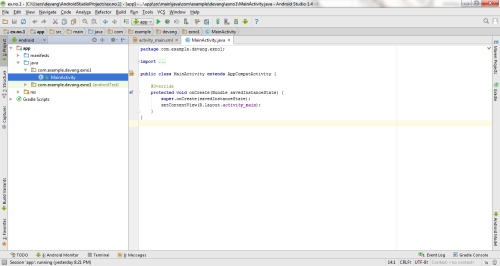
|  |  |
| --- | --- |
| 1  2  3  4  5  6  7  8  9  10  11  12  13  14  15  16  17  18  19  20  21  22 | <?xml version="1.0" encoding="utf-8"?>  <manifest xmlns:android="http://schemas.android.com/apk/res/android"      package="com.example.exno11" >        <application          android:allowBackup="true"          android:icon="@mipmap/ic\_launcher"          android:label="@string/app\_name"          android:supportsRtl="true"          android:theme="@style/AppTheme" >          <activity android:name=".MainActivity" >              <intent-filter>                  <action android:name="android.intent.action.MAIN" />                    <category android:name="android.intent.category.LAUNCHER" />              </intent-filter>          </activity>          <receiver android:name=".AlarmReceiver" >          </receiver>      </application>    </manifest> |

* So now the changes are done in the Manifest.

Java Coding for the Android Application:

Java Coding for Main Activity:

* Click on **app -> java -> com.example.exno11 -> MainActivity.**

[](https://codingconnect.net/wp-content/uploads/2016/02/MainActivity-e1456597575355.png)

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

**Code for MainActivity.java:**

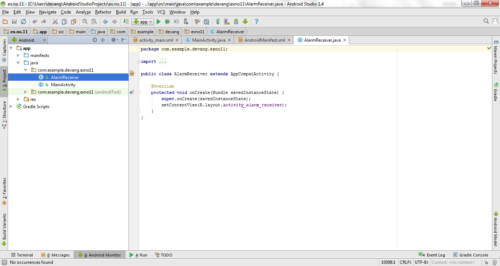
[?](https://www.codingconnect.net/android-application-creates-alarm-clock/)

|  |  |
| --- | --- |
| 1  2  3  4  5  6  7  8  9  10  11  12  13  14  15  16  17  18  19  20  21  22  23  24  25  26  27  28  29  30  31  32  33  34  35  36  37  38  39  40  41  42  43  44  45  46  47  48  49  50  51  52  53  54  55  56  57 | package com.example.exno11;    import android.app.AlarmManager;  import android.app.PendingIntent;  import android.content.Intent;  import android.os.Bundle;  import android.support.v7.app.AppCompatActivity;  import android.view.View;  import android.widget.TimePicker;  import android.widget.Toast;  import android.widget.ToggleButton;    import java.util.Calendar;    public class MainActivity extends AppCompatActivity  {      TimePicker alarmTimePicker;      PendingIntent pendingIntent;      AlarmManager alarmManager;        @Override      protected void onCreate(Bundle savedInstanceState)      {          super.onCreate(savedInstanceState);          setContentView(R.layout.activity\_main);          alarmTimePicker = (TimePicker) findViewById(R.id.timePicker);          alarmManager = (AlarmManager) getSystemService(ALARM\_SERVICE);      }      public void OnToggleClicked(View view)      {          long time;          if (((ToggleButton) view).isChecked())          {              Toast.makeText(MainActivity.this, "ALARM ON", Toast.LENGTH\_SHORT).show();              Calendar calendar = Calendar.getInstance();              calendar.set(Calendar.HOUR\_OF\_DAY, alarmTimePicker.getCurrentHour());              calendar.set(Calendar.MINUTE, alarmTimePicker.getCurrentMinute());              Intent intent = new Intent(this, AlarmReceiver.class);              pendingIntent = PendingIntent.getBroadcast(this, 0, intent, 0);                time=(calendar.getTimeInMillis()-(calendar.getTimeInMillis()%60000));              if(System.currentTimeMillis()>time)              {                  if (calendar.AM\_PM == 0)                      time = time + (1000\*60\*60\*12);                  else                      time = time + (1000\*60\*60\*24);              }              alarmManager.setRepeating(AlarmManager.RTC\_WAKEUP, time, 10000, pendingIntent);          }          else          {              alarmManager.cancel(pendingIntent);              Toast.makeText(MainActivity.this, "ALARM OFF", Toast.LENGTH\_SHORT).show();          }      }  } |

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

Java Coding for Alarm Receiver:

* Click on **app -> java -> com.example.exno11 -> AlarmReceiver.**

[](https://codingconnect.net/wp-content/uploads/2016/04/AlarmActivity-java-e1461605967294.png)

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

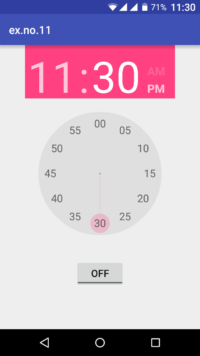
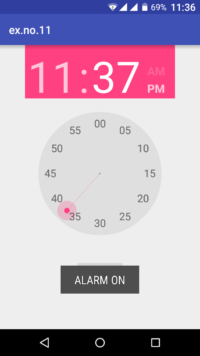
**Code for AlarmReceiver.java:**

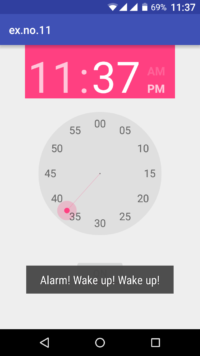
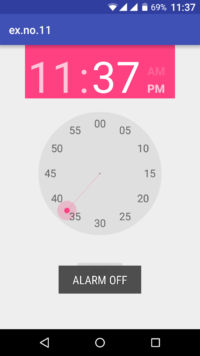
[?](https://www.codingconnect.net/android-application-creates-alarm-clock/)

|  |  |
| --- | --- |
| 1  2  3  4  5  6  7  8  9  10  11  12  13  14  15  16  17  18  19  20  21  22  23  24  25 | package com.example.exno11;    import android.content.BroadcastReceiver;  import android.content.Context;  import android.content.Intent;  import android.media.Ringtone;  import android.media.RingtoneManager;  import android.net.Uri;  import android.widget.Toast;    public class AlarmReceiver extends BroadcastReceiver  {      @Override      public void onReceive(Context context, Intent intent)      {          Toast.makeText(context, "Alarm! Wake up! Wake up!", Toast.LENGTH\_LONG).show();          Uri alarmUri = RingtoneManager.getDefaultUri(RingtoneManager.TYPE\_ALARM);          if (alarmUri == null)          {              alarmUri = RingtoneManager.getDefaultUri(RingtoneManager.TYPE\_NOTIFICATION);          }          Ringtone ringtone = RingtoneManager.getRingtone(context, alarmUri);          ringtone.play();      }  } |

* So now the Coding part of Alarm Receiver is also completed.
* Now run the application to see the output.

Output:

[](https://codingconnect.net/wp-content/uploads/2016/04/Screenshot_2016-04-25-23-30-44-e1461607767683.png)  [](https://codingconnect.net/wp-content/uploads/2016/04/Screenshot_2016-04-25-23-36-43-e1461607830225.png)

[](https://codingconnect.net/wp-content/uploads/2016/04/Screenshot_2016-04-25-23-37-02-e1461607816435.png)  [](https://codingconnect.net/wp-content/uploads/2016/04/Screenshot_2016-04-25-23-37-09-e1461607798432.png)

Result:

 Thus Android Application that creates Alarm Clock is developed and executed successfully.