**EXNO .1 DEVELOP AN APPLICATION THAT USES GUI COMPONENTS ,FONT**

**DATE: AND COLORS**

**AIM:**

To develop an application that uses GUI components , fonts and colors.

**PROCEDURE:**

* Open Genymotion shell and click **Start.**
* Open Android Studio and then click on **File->New->New project**.
* Then type the Application name as **“GUI″** and click **Next.**
* Then select the **Minimum SDK**  and click **Next**.
* Then select the **Empty Activity**and click **Next.**
* Click F**inish**.
* In **content\_main.xml** type the code for components which are to be included in the application.
* In **MainActivity.java** get the user name and password from the user .
* If the user name matches **student** and password matches **cse** ,then login is authenticated.
* The font of the text is set and the background is filled with desired color.
* Now, click **run** option.
* Finally, Android output is present in the Android emulator.

**SOURCE CODE:**

**MAINACTIVITY. JAVA:**

package com.example.dell.gui;  
  
import android.content.DialogInterface;  
import android.content.Intent;  
import android.graphics.Typeface;  
import android.graphics.drawable.ColorDrawable;  
import android.graphics.drawable.Drawable;  
import android.os.Bundle;  
import android.support.design.widget.FloatingActionButton;  
import android.support.design.widget.Snackbar;  
import android.support.v7.app.AppCompatActivity;  
import android.support.v7.widget.Toolbar;  
import android.view.View;  
import android.view.Menu;  
import android.view.MenuItem;  
import android.widget.Button;  
import android.widget.EditText;  
import android.widget.TextView;  
  
import com.example.dell.Gui;  
  
public class MainActivity extends AppCompatActivity {  
 EditText username,password;  
 Button login;  
 TextView heading;  
 @Override  
 protected void onCreate(Bundle savedInstanceState) {  
 super.onCreate(savedInstanceState);  
 setContentView(R.layout.*activity\_main*);  
 Toolbar toolbar = (Toolbar) findViewById(R.id.*toolbar*);  
 setSupportActionBar(toolbar);  
 Drawable d=new ColorDrawable(0xff2196f3);  
 this.getSupportActionBar().setBackgroundDrawable(d);  
 username=(EditText)findViewById(R.id.*user*);  
 password=(EditText)findViewById(R.id.*password*);  
 login=(Button)findViewById(R.id.*login*);  
 heading=(TextView)findViewById(R.id.*heading*);  
 *//Typeface customFont=Typeface.createFromAsset(getAssets(),"fonts/Open sans.ttf");  
 //heading.setTypeface(customFont);* login.setOnClickListener(new View.OnClickListener() {  
 @Override  
 public void onClick(View v) {  
 if(username.getText().toString().equals("student")&&password.getText().toString().equals("cse"))  
 {  
 Intent i=new Intent(getApplicationContext(),Gui.class);  
 startActivity(i);  
 finish();  
 }  
 }  
 });  
 }  
  
 @Override  
 public boolean onCreateOptionsMenu(Menu menu) {  
 *// Inflate the menu; this adds items to the action bar if it is present.* getMenuInflater().inflate(R.menu.*menu\_main*, menu);  
 return true;  
 }  
  
 @Override  
 public boolean onOptionsItemSelected(MenuItem item) {  
 *// Handle action bar item clicks here. The action bar will  
 // automatically handle clicks on the Home/Up button, so long  
 // as you specify a parent activity in AndroidManifest.xml.* int id = item.getItemId();  
  
 *//noinspection SimplifiableIfStatement* if (id == R.id.*action\_settings*) {  
 return true;  
 }  
  
 return super.onOptionsItemSelected(item);  
 }  
}

**GUI.JAVA**

package com.example.dell;  
  
import android.os.Bundle;  
import android.support.design.widget.FloatingActionButton;  
import android.support.design.widget.Snackbar;  
import android.support.v7.app.AppCompatActivity;  
import android.support.v7.widget.Toolbar;  
import android.view.View;  
  
import com.example.dell.gui.R;  
  
public class Gui extends AppCompatActivity {  
  
 @Override  
 protected void onCreate(Bundle savedInstanceState) {  
 super.onCreate(savedInstanceState);  
 setContentView(R.layout.*activity\_gui*);  
 Toolbar toolbar = (Toolbar) findViewById(R.id.*toolbar*);  
 setSupportActionBar(toolbar);  
  
 FloatingActionButton fab = (FloatingActionButton) findViewById(R.id.*fab*);  
 fab.setOnClickListener(new View.OnClickListener() {  
 @Override  
 public void onClick(View view) {  
 Snackbar.*make*(view, "Replace with your own action", Snackbar.*LENGTH\_LONG*)  
 .setAction("Action", null).show();  
 }  
 });  
 }  
  
}

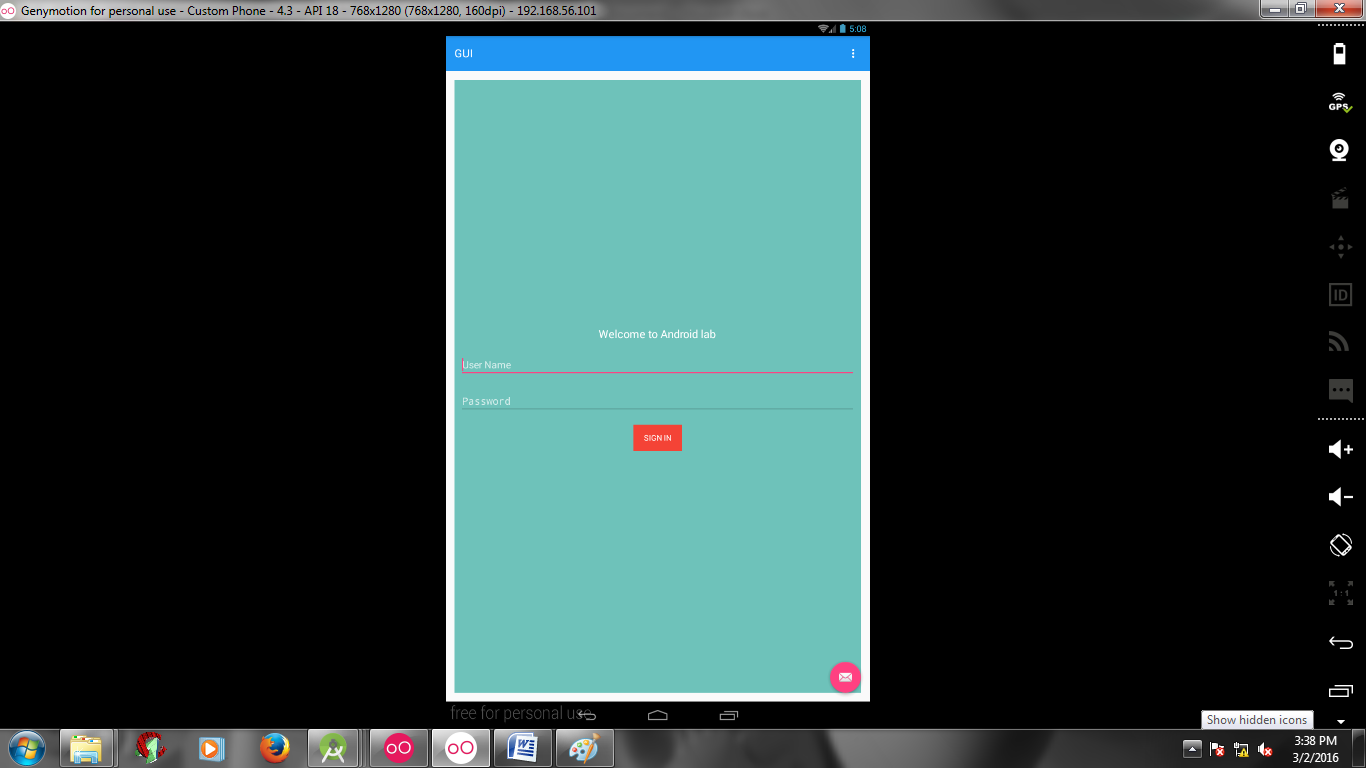
**ACTIVITY\_MAIN.XML**

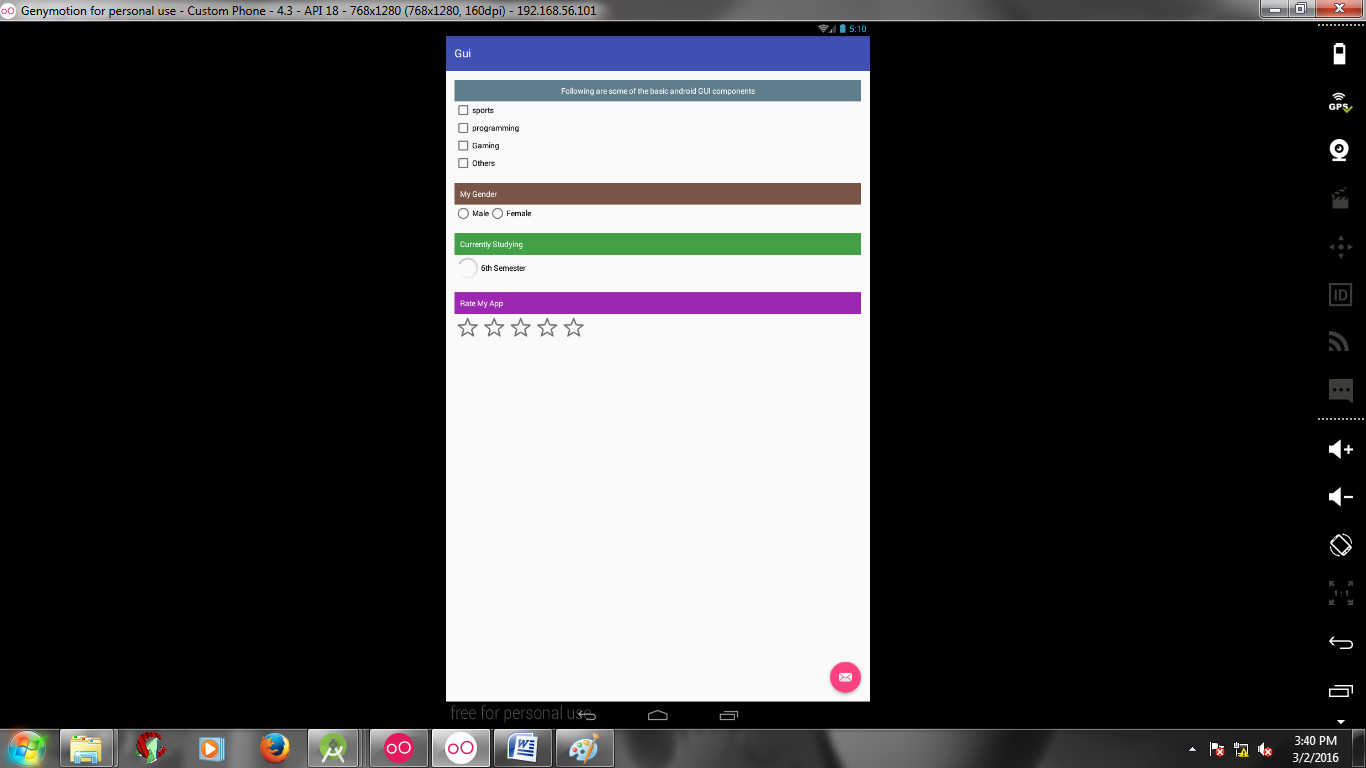
*<?*xml version="1.0" encoding="utf-8"*?>*<android.support.design.widget.CoordinatorLayout xmlns:android="http://schemas.android.com/apk/res/android"  
 xmlns:app="http://schemas.android.com/apk/res-auto"  
 xmlns:tools="http://schemas.android.com/tools"  
 android:layout\_width="match\_parent"  
 android:layout\_height="match\_parent"  
 android:fitsSystemWindows="true"  
 tools:context="com.example.dell.gui.MainActivity">  
  
 <android.support.design.widget.AppBarLayout  
 android:layout\_width="match\_parent"  
 android:layout\_height="wrap\_content"  
 android:theme="@style/AppTheme.AppBarOverlay">  
  
 <android.support.v7.widget.Toolbar  
 android:id="@+id/toolbar"  
 android:layout\_width="match\_parent"  
 android:layout\_height="?attr/actionBarSize"  
 android:background="?attr/colorPrimary"  
 app:popupTheme="@style/AppTheme.PopupOverlay" />  
  
 </android.support.design.widget.AppBarLayout>  
  
 <include layout="@layout/content\_main" />  
  
 <android.support.design.widget.FloatingActionButton  
 android:id="@+id/fab"  
 android:layout\_width="wrap\_content"  
 android:layout\_height="wrap\_content"  
 android:layout\_gravity="bottom|end"  
 android:layout\_margin="@dimen/fab\_margin"  
 android:src="@android:drawable/ic\_dialog\_email" />  
  
</android.support.design.widget.CoordinatorLayout>

**ACTIVITY\_GUI.XML:**

*<?*xml version="1.0" encoding="utf-8"*?>*<android.support.design.widget.CoordinatorLayout xmlns:android="http://schemas.android.com/apk/res/android"  
 xmlns:app="http://schemas.android.com/apk/res-auto"  
 xmlns:tools="http://schemas.android.com/tools"  
 android:layout\_width="match\_parent"  
 android:layout\_height="match\_parent"  
 android:fitsSystemWindows="true"  
 tools:context="com.example.dell.Gui">  
  
 <android.support.design.widget.AppBarLayout  
 android:layout\_width="match\_parent"  
 android:layout\_height="wrap\_content"  
 android:theme="@style/AppTheme.AppBarOverlay">  
  
 <android.support.v7.widget.Toolbar  
 android:id="@+id/toolbar"  
 android:layout\_width="match\_parent"  
 android:layout\_height="?attr/actionBarSize"  
 android:background="?attr/colorPrimary"  
 app:popupTheme="@style/AppTheme.PopupOverlay" />  
  
 </android.support.design.widget.AppBarLayout>  
  
 <include layout="@layout/content\_gui" />  
  
 <android.support.design.widget.FloatingActionButton  
 android:id="@+id/fab"  
 android:layout\_width="wrap\_content"  
 android:layout\_height="wrap\_content"  
 android:layout\_gravity="bottom|end"  
 android:layout\_margin="@dimen/fab\_margin"  
 android:src="@android:drawable/ic\_dialog\_email" />  
  
</android.support.design.widget.CoordinatorLayout>

**OUTPUT:**

******



**RESULT:**

The above project on development of an application that uses gui components ,font and colors was verified and executed successfully.

**EXNO .2 DEVELOP AN APPLICATION THAT USES LAYOUT MANAGERS AND DATE: EVENT LISTENERS**

**AIM:**

To develop an application that uses Layout Managers and Event Listeners.

**PROCEDURE:**

* Open Genymotion shell and click **Start.**
* Open Android Studio and then click on **File->New->New project**.
* Then type the Application name as **“GUI″** and click **Next.**
* Then select the **Minimum SDK**  and click **Next**.
* Then select the **Empty Activity**and click **Next.**
* Click F**inish**.
* In **content\_main.xml** type the code for components which are to be included in the application.
* The **MainActivity.java** is created with two buttons and click event listeners and handlers are defined for it.
* Now, click **run** option.
* Finally, Android output is present in the Android emulator.

**SOURCE CODE:**

**MAIN\_ACTIVITY:**

package com.example.dell.event;  
import android.app.ProgressDialog;  
import android.os.Bundle;  
import android.support.design.widget.FloatingActionButton;  
import android.support.design.widget.Snackbar;  
import android.support.v7.app.AppCompatActivity;  
import android.support.v7.widget.Toolbar;  
import android.view.View;  
import android.view.Menu;  
import android.view.MenuItem;  
import android.widget.Button;  
import android.widget.TextView;  
  
public class MainActivity extends AppCompatActivity {  
private ProgressDialog progress;  
 Button b1,b2;  
 @Override  
 protected void onCreate(Bundle savedInstanceState) {  
 super.onCreate(savedInstanceState);  
 setContentView(R.layout.*activity\_main*);  
 Toolbar toolbar = (Toolbar) findViewById(R.id.*toolbar*);  
 setSupportActionBar(toolbar);  
 progress =new ProgressDialog(this);  
 b1=(Button)findViewById(R.id.*button*);  
 b2=(Button)findViewById(R.id.*button2*);  
 b1.setOnClickListener(new View.OnClickListener() {  
 @Override  
 public void onClick(View v) {  
 TextView txtView=(TextView)findViewById(R.id.*textview*);  
 txtView.setTextSize(25);  
 }  
 });  
 b2.setOnClickListener(new View.OnClickListener() {  
 @Override  
 public void onClick(View v) {  
 TextView txtView=(TextView)findViewById(R.id.*textview*);  
 txtView.setTextSize(55);  
 }  
 });  
 }  
 @Override  
 public boolean onCreateOptionsMenu(Menu menu) {  
 *// Inflate the menu; this adds items to the action bar if it is present.* getMenuInflater().inflate(R.menu.*menu\_main*, menu);  
 return true;  
 }  
 @Override  
 public boolean onOptionsItemSelected(MenuItem item) {  
 *// Handle action bar item clicks here. The action bar will  
 // automatically handle clicks on the Home/Up button, so long  
 // as you specify a parent activity in AndroidManifest.xml.* int id = item.getItemId();  
  
 *//noinspection SimplifiableIfStatement* if (id == R.id.*action\_settings*) {  
 return true;  
 }  
 return super.onOptionsItemSelected(item);  
} }

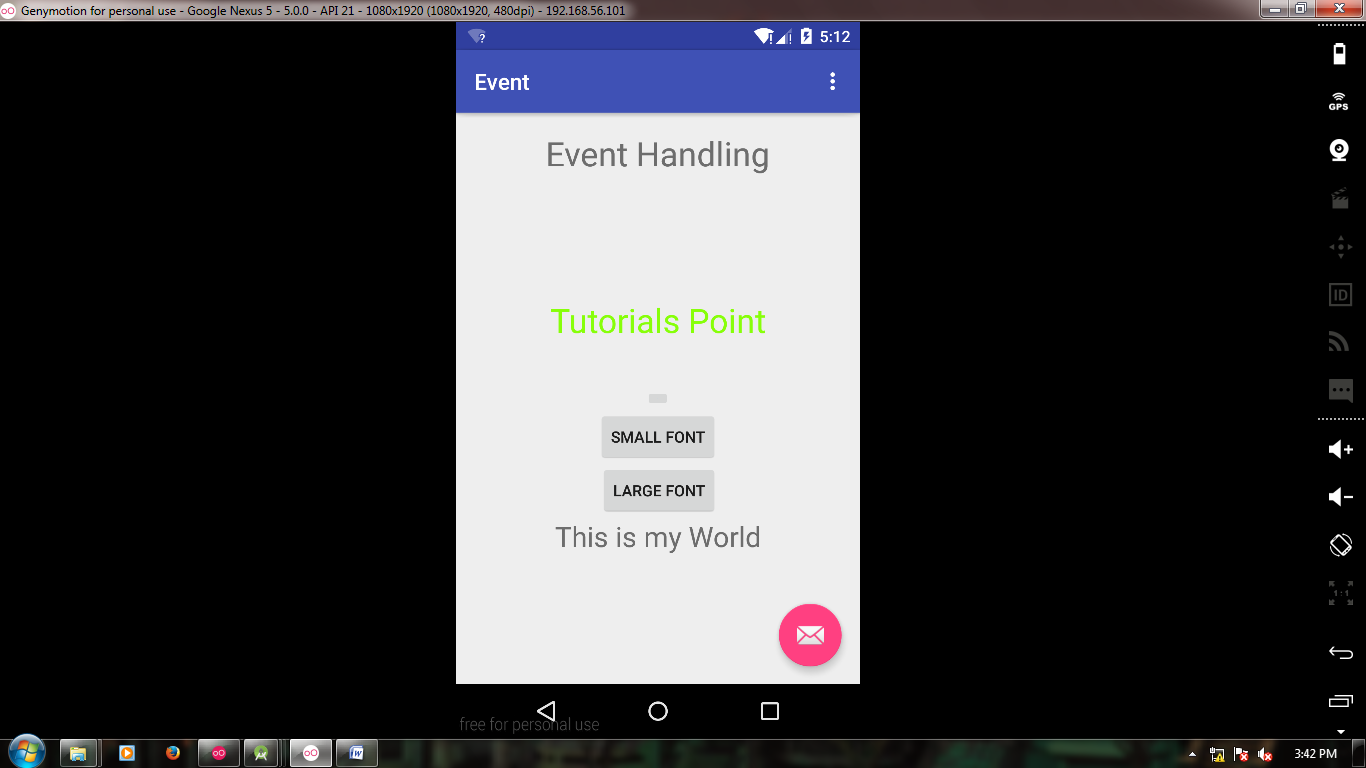
**ACTIVITY\_MAIN:**

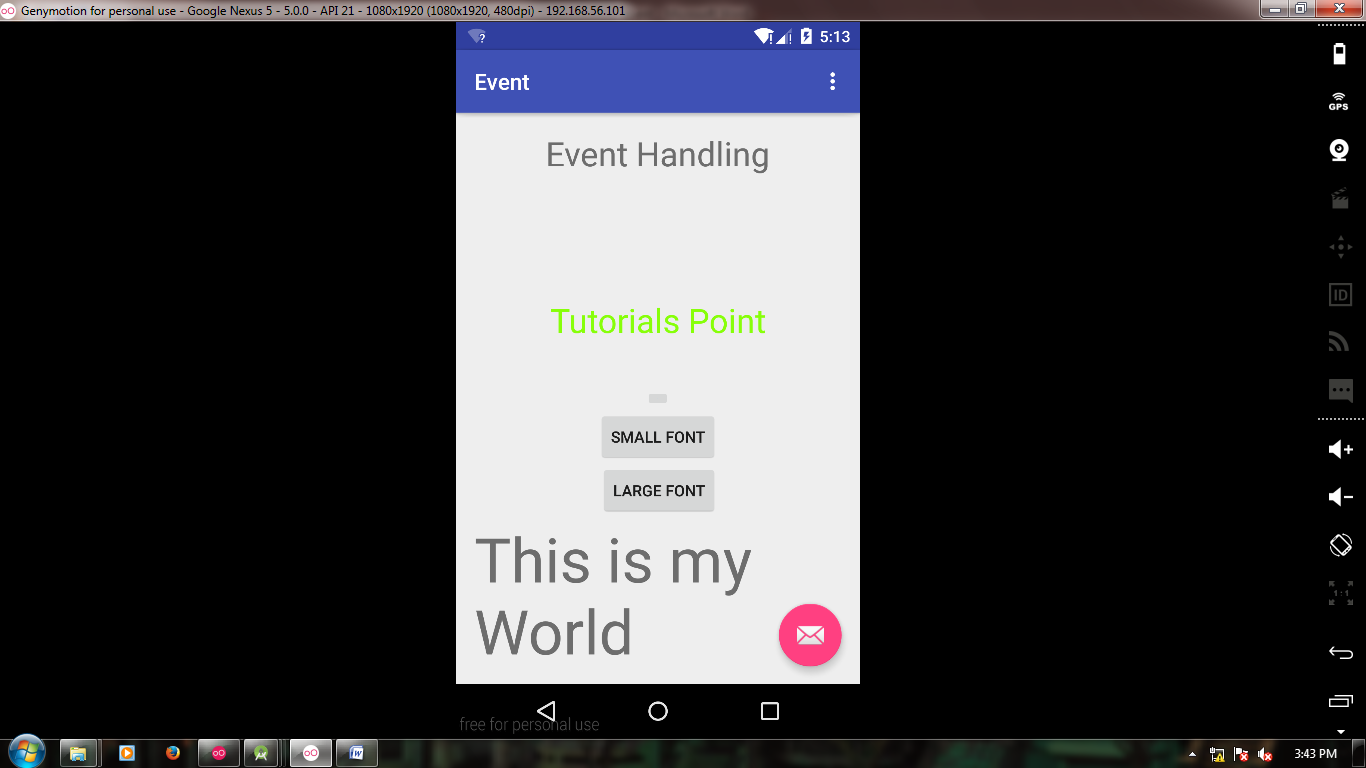
*<?*xml version="1.0" encoding="utf-8"*?>*<android.support.design.widget.CoordinatorLayout xmlns:android="http://schemas.android.com/apk/res/android"  
 xmlns:app="http://schemas.android.com/apk/res-auto"  
 xmlns:tools="http://schemas.android.com/tools"  
 android:layout\_width="match\_parent"  
 android:layout\_height="match\_parent"  
 android:fitsSystemWindows="true"  
 tools:context="com.example.dell.event.MainActivity">  
 <android.support.design.widget.AppBarLayout  
 android:layout\_width="match\_parent"  
 android:layout\_height="wrap\_content"  
 android:theme="@style/AppTheme.AppBarOverlay">  
 <android.support.v7.widget.Toolbar  
 android:id="@+id/toolbar"  
 android:layout\_width="match\_parent"  
 android:layout\_height="?attr/actionBarSize"  
 android:background="?attr/colorPrimary"  
 app:popupTheme="@style/AppTheme.PopupOverlay" />  
  
 </android.support.design.widget.AppBarLayout>  
 <include layout="@layout/content\_main" />  
 <android.support.design.widget.FloatingActionButton  
 android:id="@+id/fab"  
 android:layout\_width="wrap\_content"  
 android:layout\_height="wrap\_content"  
 android:layout\_gravity="bottom|end"  
 android:layout\_margin="@dimen/fab\_margin"  
 android:src="@android:drawable/ic\_dialog\_email" />  
</android.support.design.widget.CoordinatorLayout>

**CONTENT\_MAIN:**

*<?*xml version="1.0" encoding="utf-8"*?>*<RelativeLayout xmlns:android="http://schemas.android.com/apk/res/android"  
 xmlns:app="http://schemas.android.com/apk/res-auto"  
 xmlns:tools="http://schemas.android.com/tools"  
 android:layout\_width="match\_parent"  
 android:layout\_height="match\_parent"  
 android:paddingBottom="@dimen/activity\_vertical\_margin"  
 android:paddingLeft="@dimen/activity\_horizontal\_margin"  
 android:paddingRight="@dimen/activity\_horizontal\_margin"  
 android:paddingTop="@dimen/activity\_vertical\_margin"  
 app:layout\_behavior="@string/appbar\_scrolling\_view\_behavior"  
 tools:context="com.example.dell.event.MainActivity"  
 tools:showIn="@layout/activity\_main">  
 <TextView  
 android:layout\_width="wrap\_content"  
 android:layout\_height="wrap\_content"  
 android:id="@+id/textview1"  
 android:text="Event Handling"  
 android:layout\_alignParentTop="true"  
 android:layout\_centerHorizontal="true"  
 android:textSize="30dp"  
 />  
 <TextView  
 android:layout\_width="wrap\_content"  
 android:layout\_height="wrap\_content"  
 android:id="@+id/textview2"  
 android:text="Tutorials Point"  
 android:textColor="#ff87ff09"  
 android:textSize="30dp"  
 android:layout\_above="@+id/imageButton"  
 android:layout\_centerHorizontal="true"  
 android:layout\_marginBottom="40dp"  
 />  
 <ImageButton  
 android:layout\_width="wrap\_content"  
 android:layout\_height="wrap\_content"  
 android:id="@+id/imageButton"  
 android:src="#feabcdea"  
 android:layout\_centerVertical="true"  
 android:layout\_centerHorizontal="true"  
 />  
 <Button  
 android:layout\_width="wrap\_content"  
 android:layout\_height="wrap\_content"  
 android:text="Small font"  
 android:id="@+id/button"  
 android:layout\_below="@+id/imageButton"  
 android:layout\_centerHorizontal="true"  
 />  
 <Button  
 android:layout\_width="wrap\_content"  
 android:layout\_height="wrap\_content"  
 android:text="Large Font"  
 android:id="@+id/button2"  
 android:layout\_below="@+id/button"  
 android:layout\_alignRight="@+id/button"  
 android:layout\_alignEnd="@+id/button"  
 />  
 <TextView  
 android:layout\_width="wrap\_content"  
 android:layout\_height="wrap\_content"  
 android:text="This is my World"  
 android:id="@+id/textview"  
 android:layout\_below="@+id/button2"  
 android:layout\_centerHorizontal="true"  
 android:textSize="25dp"  
 />  
</RelativeLayout>

**OUTPUT:**

****



**RESULT:**

­­­­­­­­­­­­­­­­­­­­­­­­­­ The above project on development of an application that uses layout managers and event listeners was verified and executed successfully.

**EXNO .3 DEVELOP A NATIVE CALCULATOR APPLICATION**

**DATE:**

**AIM:**

To develop a Native calculator application.

**PROCEDURE:**

* Open Genymotion shell and click **Start.**
* Open Android Studio and then click on **File->New->New project**.
* Then type the Application name as **“GUI″** and click **Next.**
* Then select the **Minimum SDK**  and click **Next**.
* Then select the **Empty Activity**and click **Next.**
* Click F**inish**.
* In **content\_main.xml** type the code for components which are to be included in the application.
* In the MainActivity.java ,Two text fields are provided to get the two operands from the user .
* Five buttons are created each one for addition, subtraction, multiplication ,division and modulo operator and click event listeners and handlers are defined for them.
* Using switch case, the functions of the operators are defined.
* click **run** option.
* Finally, Android output is present in the Android emulator.

**SOURCE CODE:**

**MAIN\_ACTIVITY.JAVA:**

package com.example.dell.calc;  
  
import android.content.DialogInterface;  
import android.os.Bundle;  
import android.support.design.widget.FloatingActionButton;  
import android.support.design.widget.Snackbar;  
import android.support.v7.app.AppCompatActivity;  
import android.support.v7.widget.Toolbar;  
import android.text.TextUtils;  
import android.view.View;  
import android.view.Menu;  
import android.view.MenuItem;  
import android.widget.Button;  
import android.widget.EditText;  
import android.widget.TextView;  
  
public class MainActivity extends AppCompatActivity implements View.OnClickListener {  
EditText etNum1,etNum2;  
 Button btnAdd,btnSub,btnMult,btnDiv,btnMod;  
 TextView tvResult;  
 String oper="";  
 @Override  
 protected void onCreate(Bundle savedInstanceState) {  
 super.onCreate(savedInstanceState);  
 setContentView(R.layout.activity\_main);  
 Toolbar toolbar = (Toolbar) findViewById(R.id.toolbar);  
 setSupportActionBar(toolbar);  
  
 FloatingActionButton fab = (FloatingActionButton) findViewById(R.id.fab);  
 fab.setOnClickListener(new View.OnClickListener() {  
 @Override  
 public void onClick(View view) {  
 Snackbar.make(view, "Replace with your own action", Snackbar.LENGTH\_LONG)  
 .setAction("Action", null).show();  
 }  
 });  
 etNum1=(EditText)findViewById(R.id.etNum1);  
 etNum2=(EditText)findViewById(R.id.etNum2);  
 btnAdd=(Button)findViewById(R.id.btnAdd);  
 btnSub=(Button)findViewById(R.id.btnSub);  
 btnMult=(Button)findViewById(R.id.btnMult);  
 btnDiv=(Button)findViewById(R.id.btnDiv);  
 btnMod=(Button)findViewById(R.id.btnMod);  
 tvResult=(TextView)findViewById(R.id.tvResult);  
 *//set listener* btnAdd.setOnClickListener(this);  
 btnSub.setOnClickListener(this);  
 btnMult.setOnClickListener(this);  
 btnDiv.setOnClickListener(this);  
 btnMod.setOnClickListener(this);  
 }  
 @Override  
 public boolean onCreateOptionsMenu(Menu menu) {  
 *// Inflate the menu; this adds items to the action bar if it is present.* getMenuInflater().inflate(R.menu.menu\_main, menu);  
 return true;  
 }  
  
 @Override  
 public boolean onOptionsItemSelected(MenuItem item) {  
 *// Handle action bar item clicks here. The action bar will  
 // automatically handle clicks on the Home/Up button, so long  
 // as you specify a parent activity in AndroidManifest.xml.* int id = item.getItemId();  
  
 *//noinspection SimplifiableIfStatement* if (id == R.id.action\_settings) {  
 return true;  
 }  
  
 return super.onOptionsItemSelected(item);  
 }  
  
 @Override  
 public void onClick(View v) {  
 float num1=0,num2=0,result=0;  
 if(TextUtils.*isEmpty*(etNum1.getText().toString())||TextUtils.*isEmpty*(etNum2.getText().toString()))  
 {  
 return;  
 }  
 num1=Float.*parseFloat*(etNum1.getText().toString());  
 num2=Float.*parseFloat*(etNum2.getText().toString());  
 switch(v.getId())  
 {  
 case R.id.btnAdd:  
 oper="+";  
 result=num1+num2;  
 break;  
 case R.id.btnSub:  
 oper="-";  
 result=num1-num2;  
 break;  
 case R.id.btnMult:  
 oper="\*";  
 result=num1\*num2;  
 break;  
 case R.id.btnDiv:  
 oper="/";  
 result=num1/num2;  
 break;  
 case R.id.btnMod:  
 oper="%";  
 result=num1%num2;  
 break;  
 default:  
 break;  
 }  
 tvResult.setText(num1+""+oper+""+num2+"="+result);  
 }  
}

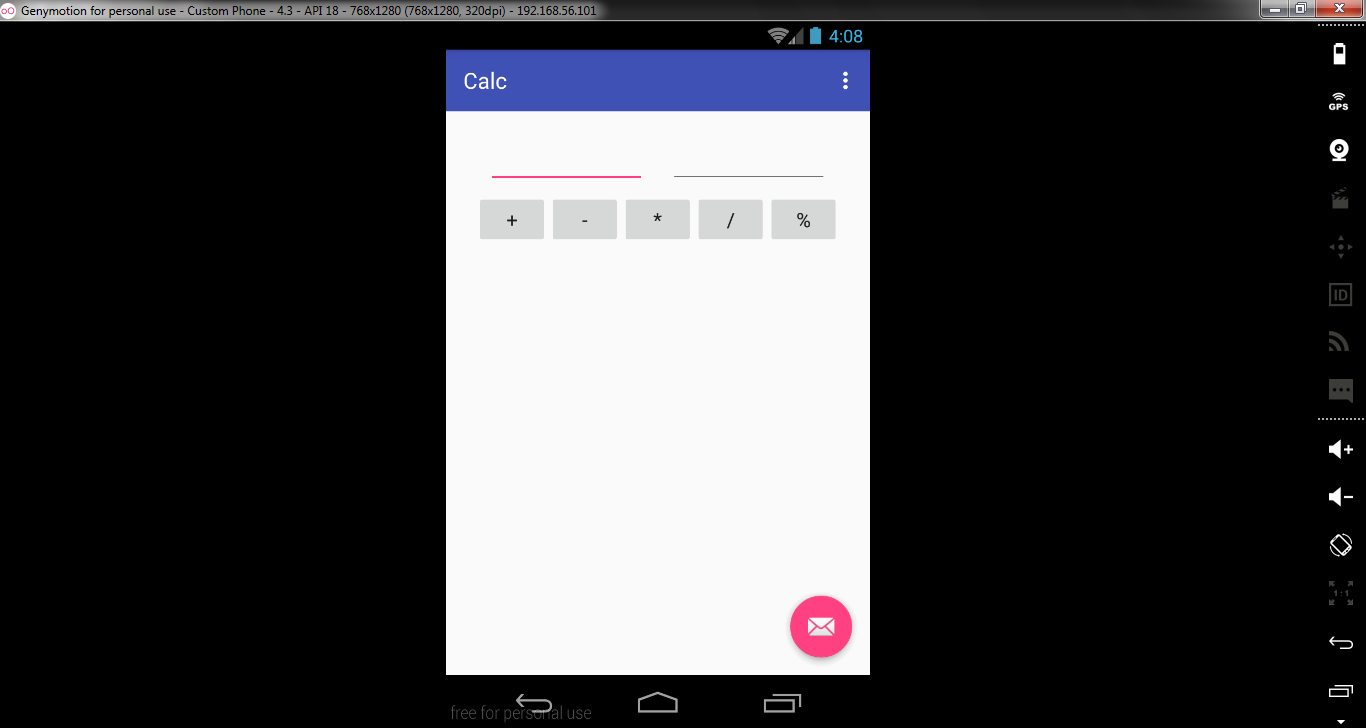
**ACTIVITY\_MAIN.XML:**

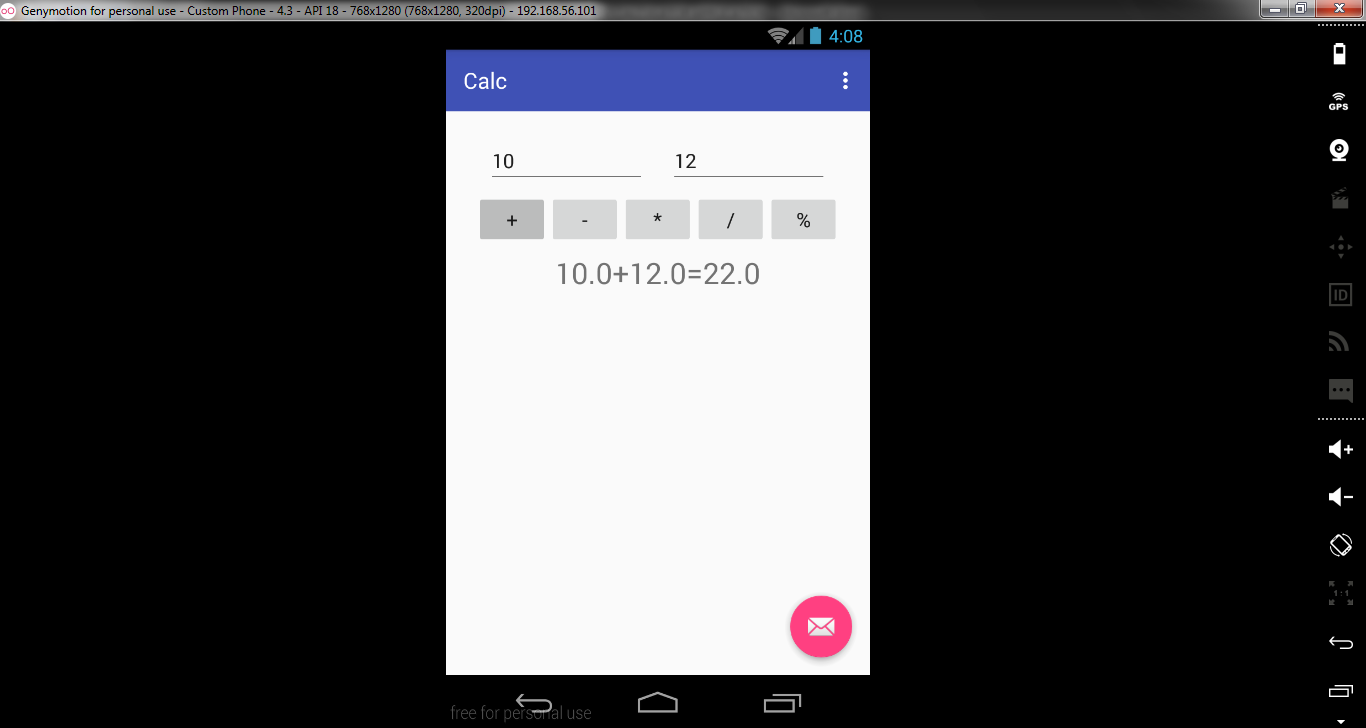
*<?*xml version="1.0" encoding="utf-8"*?>*<android.support.design.widget.CoordinatorLayout xmlns:android="http://schemas.android.com/apk/res/android"  
 xmlns:app="http://schemas.android.com/apk/res-auto"  
 xmlns:tools="http://schemas.android.com/tools"  
 android:layout\_width="match\_parent"  
 android:layout\_height="match\_parent"  
 android:fitsSystemWindows="true"  
 tools:context="com.example.dell.calc.MainActivity">  
  
 <android.support.design.widget.AppBarLayout  
 android:layout\_width="match\_parent"  
 android:layout\_height="wrap\_content"  
 android:theme="@style/AppTheme.AppBarOverlay">  
  
 <android.support.v7.widget.Toolbar  
 android:id="@+id/toolbar"  
 android:layout\_width="match\_parent"  
 android:layout\_height="?attr/actionBarSize"  
 android:background="?attr/colorPrimary"  
 app:popupTheme="@style/AppTheme.PopupOverlay" />  
  
 </android.support.design.widget.AppBarLayout>  
 <include layout="@layout/content\_main" />  
 <android.support.design.widget.FloatingActionButton  
 android:id="@+id/fab"  
 android:layout\_width="wrap\_content"  
 android:layout\_height="wrap\_content"  
 android:layout\_gravity="bottom|end"  
 android:layout\_margin="@dimen/fab\_margin"  
 android:src="@android:drawable/ic\_dialog\_email" />  
</android.support.design.widget.CoordinatorLayout>

**CONTENT\_MAIN.XML:**

*<?*xml version="1.0" encoding="utf-8"*?>*<LinearLayout xmlns:android="http://schemas.android.com/apk/res/android"  
 xmlns:app="http://schemas.android.com/apk/res-auto"  
 xmlns:tools="http://schemas.android.com/tools"  
 android:layout\_width="match\_parent"  
 android:layout\_height="match\_parent"  
 android:orientation="vertical"  
 android:paddingBottom="@dimen/activity\_vertical\_margin"  
 android:paddingLeft="@dimen/activity\_horizontal\_margin"  
 android:paddingRight="@dimen/activity\_horizontal\_margin"  
 android:paddingTop="@dimen/activity\_vertical\_margin"  
 app:layout\_behavior="@string/appbar\_scrolling\_view\_behavior"  
 tools:context="com.example.dell.calc.MainActivity"  
 tools:showIn="@layout/activity\_main">  
 <LinearLayout  
 android:layout\_width="match\_parent"  
 android:layout\_height="wrap\_content"  
 android:id="@+id/linearLayout1"  
 android:orientation="horizontal"  
 android:layout\_marginLeft="10pt"  
 android:layout\_marginRight="10pt"  
 android:layout\_marginTop="3pt">  
 <EditText  
 android:layout\_width="match\_parent"  
 android:layout\_height="wrap\_content"  
 android:id="@+id/etNum1"  
 android:layout\_marginRight="5pt"  
 android:layout\_weight="1"  
 android:inputType="numberDecimal"  
 />  
 <EditText  
 android:layout\_width="match\_parent"  
 android:layout\_height="wrap\_content"  
 android:id="@+id/etNum2"  
 android:layout\_marginLeft="5pt"  
 android:layout\_weight="1"  
 android:inputType="numberDecimal"  
 />  
 </LinearLayout>  
 <LinearLayout  
 android:layout\_width="match\_parent"  
 android:layout\_height="wrap\_content"  
 android:orientation="horizontal"  
 android:id="@+id/linearLayout2"  
 android:layout\_marginLeft="5pt"  
 android:layout\_marginRight="5pt"  
 android:layout\_marginTop="3pt"  
 >  
 <Button  
 android:layout\_width="match\_parent"  
 android:layout\_height="wrap\_content"  
 android:id="@+id/btnAdd"  
 android:layout\_weight="1"  
 android:text="+"  
 android:textSize="8pt"  
 />  
 <Button  
 android:layout\_width="match\_parent"  
 android:layout\_height="wrap\_content"  
 android:id="@+id/btnSub"  
 android:layout\_weight="1"  
 android:text="-"  
 android:textSize="8pt"  
 />  
 <Button  
 android:layout\_width="match\_parent"  
 android:layout\_height="wrap\_content"  
 android:id="@+id/btnMult"  
 android:layout\_weight="1"  
 android:text="\*"  
 android:textSize="8pt"  
 />  
 <Button  
 android:layout\_width="match\_parent"  
 android:layout\_height="wrap\_content"  
 android:id="@+id/btnDiv"  
 android:layout\_weight="1"  
 android:text="/"  
 android:textSize="8pt"  
 />  
 <Button  
 android:layout\_width="match\_parent"  
 android:layout\_height="wrap\_content"  
 android:id="@+id/btnMod"  
 android:layout\_weight="1"  
 android:text="%"  
 android:textSize="8pt"  
 />  
 </LinearLayout>  
 <TextView  
 android:layout\_width="match\_parent"  
 android:layout\_height="wrap\_content"  
 android:id="@+id/tvResult"  
 android:layout\_marginLeft="5pt"  
 android:layout\_marginRight="5pt"  
 android:layout\_marginTop="3pt"  
 android:gravity="center\_horizontal"  
 android:textSize="12pt"  
 />  
</LinearLayout>

**OUTPUT:**





**RESULT:**

The above project on development of a native calculator application was verified and executed successfully.

**EXNO .4 WRITE AN APPLICATION THAT DRAWS BASIC GRAPHICAL**

**DATE: PRIMITIVES ON THE** **SCREEN**

**AIM:**

To develop an application that draws basic graphical primitives on the screen.

**PROCEDURE:**

* Open Genymotion shell and click **Start.**
* Open Android Studio and then click on **File->New->New project**.
* Then type the Application name as **“GUI″** and click **Next.**
* Then select the **Minimum SDK**  and click **Next**.
* Then select the **Empty Activity**and click **Next.**
* Click **Finish.**
* In **content\_main.xml** type the code for components which are to be included in the application.
* In  **MainActivity.java** ,Two text fields are provided to get the two operands from the user .
* Five buttons are created each one for addition, subtraction, multiplication ,division and modulo operator and click event listeners and handlers are defined for them.
* Using switch case, the functions of the operators are defined.
* Click **run** option.
* Finally, Android output is present in the Android emulator.

**SOURCE CODE:**

**MAIN ACTIVITY:**

package com.example.dell.graphicprimitives;  
  
import android.os.Bundle;  
import android.support.design.widget.FloatingActionButton;  
import android.support.design.widget.Snackbar;  
import android.support.v7.app.AppCompatActivity;  
import android.support.v7.widget.Toolbar;  
import android.view.View;  
import android.view.Menu;  
import android.view.MenuItem;  
import android.widget.Button;  
import android.widget.LinearLayout;  
  
public class MainActivity extends AppCompatActivity {  
Button line,circle;  
 LinearLayout ll;  
 @Override  
 protected void onCreate(Bundle savedInstanceState) {  
 super.onCreate(savedInstanceState);  
 setContentView(R.layout.*activity\_main*);  
 Toolbar toolbar = (Toolbar) findViewById(R.id.*toolbar*);  
 setSupportActionBar(toolbar);  
  
 ll = (LinearLayout)findViewById(R.id.*ll*);  
 line = (Button)findViewById(R.id.*line*);  
 circle = (Button)findViewById(R.id.*circle*);  
 line.setOnClickListener(new View.OnClickListener() {  
 @Override  
 public void onClick(View v) {  
 ll.removeAllViews();  
 ll.addView(new DrawPrimitive(getApplicationContext(), "line"));  
 }  
 });  
 circle.setOnClickListener(new View.OnClickListener() {  
 @Override  
 public void onClick(View v) {  
 ll.removeAllViews();  
 ll.addView(new DrawPrimitive(getApplicationContext(), "circle"));  
 }  
 });  
  
 }  
  
 @Override  
 public boolean onCreateOptionsMenu(Menu menu) {  
 *// Inflate the menu; this adds items to the action bar if it is present.* getMenuInflater().inflate(R.menu.*menu\_main*, menu);  
 return true;  
 }  
  
 @Override  
 public boolean onOptionsItemSelected(MenuItem item) {  
 *// Handle action bar item clicks here. The action bar will  
 // automatically handle clicks on the Home/Up button, so long  
 // as you specify a parent activity in AndroidManifest.xml.* int id = item.getItemId();  
 *//noinspection SimplifiableIfStatement* if (id == R.id.*action\_settings*) {  
 return true;  
 }  
  
 return super.onOptionsItemSelected(item);  
 }  
}

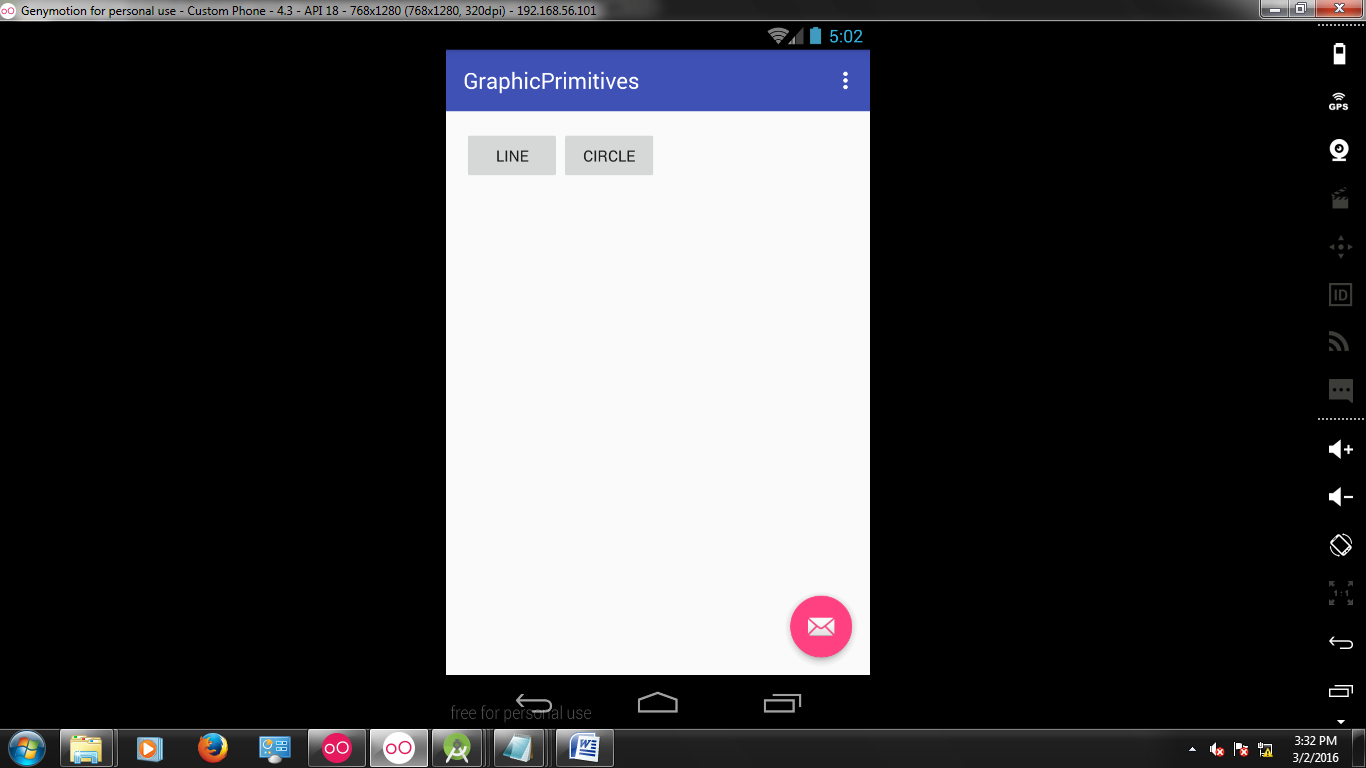
**ACTIVITY\_MAIN.XML:**

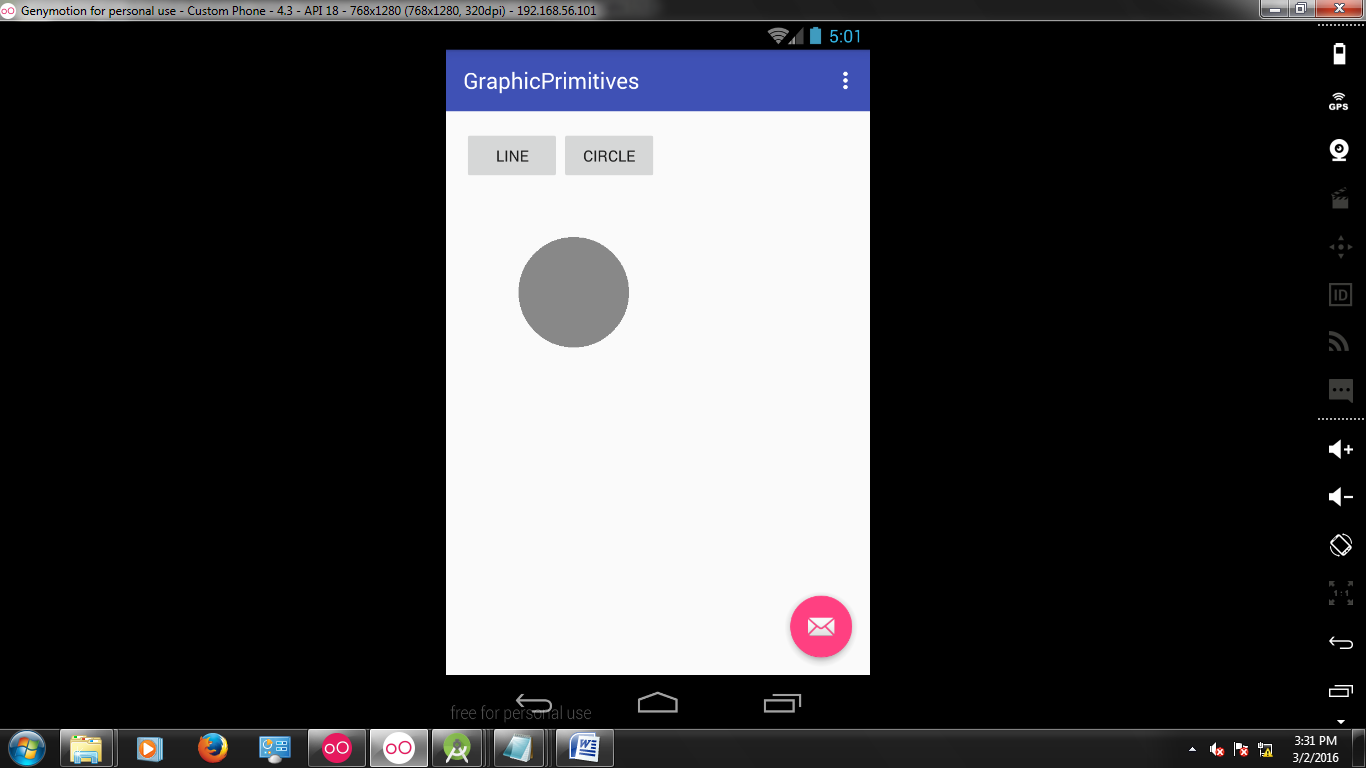
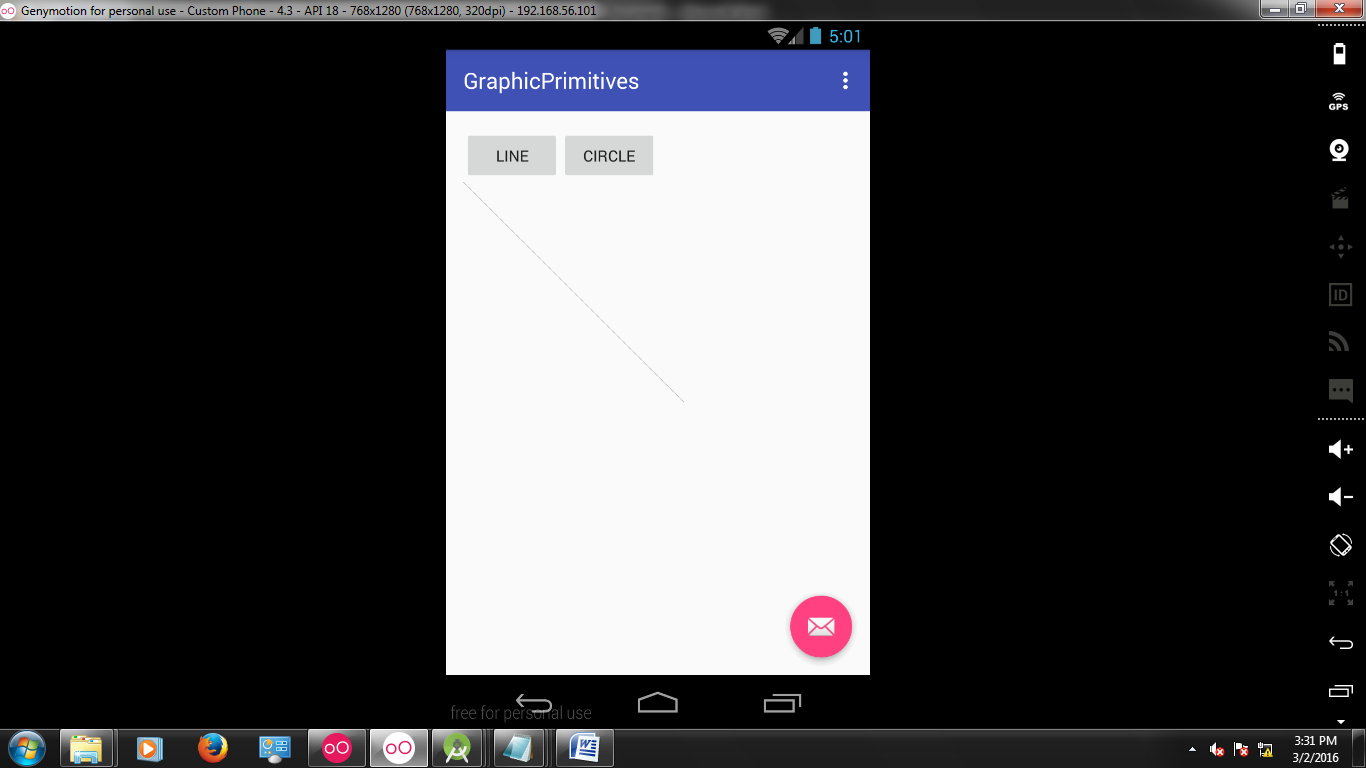
*<?*xml version="1.0" encoding="utf-8"*?>*<android.support.design.widget.CoordinatorLayout xmlns:android="http://schemas.android.com/apk/res/android"  
 xmlns:app="http://schemas.android.com/apk/res-auto"  
 xmlns:tools="http://schemas.android.com/tools"  
 android:layout\_width="match\_parent"  
 android:layout\_height="match\_parent"  
 android:fitsSystemWindows="true"  
 tools:context="com.example.dell.graphicprimitives.MainActivity">  
  
 <android.support.design.widget.AppBarLayout  
 android:layout\_width="match\_parent"  
 android:layout\_height="wrap\_content"  
 android:theme="@style/AppTheme.AppBarOverlay">  
  
 <android.support.v7.widget.Toolbar  
 android:id="@+id/toolbar"  
 android:layout\_width="match\_parent"  
 android:layout\_height="?attr/actionBarSize"  
 android:background="?attr/colorPrimary"  
 app:popupTheme="@style/AppTheme.PopupOverlay" />  
  
 </android.support.design.widget.AppBarLayout>  
  
 <include layout="@layout/content\_main" />  
  
 <android.support.design.widget.FloatingActionButton  
 android:id="@+id/fab"  
 android:layout\_width="wrap\_content"  
 android:layout\_height="wrap\_content"  
 android:layout\_gravity="bottom|end"  
 android:layout\_margin="@dimen/fab\_margin"  
 android:src="@android:drawable/ic\_dialog\_email" />  
  
</android.support.design.widget.CoordinatorLayout>

**CONTENT\_MAIN.XML**

*<?*xml version="1.0" encoding="utf-8"*?>*<RelativeLayout xmlns:android="http://schemas.android.com/apk/res/android"  
 xmlns:app="http://schemas.android.com/apk/res-auto"  
 xmlns:tools="http://schemas.android.com/tools"  
 android:layout\_width="match\_parent"  
 android:layout\_height="match\_parent"  
 android:paddingBottom="@dimen/activity\_vertical\_margin"  
 android:paddingLeft="@dimen/activity\_horizontal\_margin"  
 android:paddingRight="@dimen/activity\_horizontal\_margin"  
 android:paddingTop="@dimen/activity\_vertical\_margin"  
 app:layout\_behavior="@string/appbar\_scrolling\_view\_behavior"  
 tools:context="com.example.dell.graphicprimitives.MainActivity"  
 tools:showIn="@layout/activity\_main">  
<Button  
 android:layout\_width="wrap\_content"  
 android:layout\_height="wrap\_content"  
 android:text="line"  
 android:id="@+id/line"/>  
 <Button  
 android:layout\_width="wrap\_content"  
 android:layout\_height="wrap\_content"  
 android:text="circle"  
 android:layout\_toRightOf="@+id/line"  
 android:id="@+id/circle"/>  
 <LinearLayout  
 android:layout\_width="match\_parent"  
 android:layout\_height="match\_parent"  
 android:id="@+id/ll"  
 android:layout\_below="@+id/line"  
 android:orientation="vertical"></LinearLayout>  
</RelativeLayout>

**OUTPUT:**





**RESULT:**

The above project on writing an application that draws basic graphical primitives on the screen was verified and executed successfully.

**EXNO .5 DEVELOP AN APPLICATION THAT MAKES USE OF DATABASE**

**DATE:**

**AIM:**

To develop an application that makes use of database

**PROCEDURE:**

* Open Genymotion shell and click **Start.**
* Open Android Studio and then click on **File->New->New project**.
* Then type the Application name as **“RSS″** and click **Next.**
* Then select the **Minimum SDK**  and click **Next**.
* Then select the **Empty Activity**and click **Next.**
* Click F**inish**.
* In **content\_main.xml** type the code for components which are to be included in the application.
* In main activity ,get the contacts from user using **getAllcontacts()** function.
* The database handler gets the contacts from the user and updates it into database and performs different operations using the following functions-**addContact(),getContact(),updateContact(),deleteContact().**
* Finally, Android output is present in the Android emulator.

**SOURCE CODE:**

**MANIFEST:**

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

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

package="com.example.nirmal.db">

<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>

</application>

</manifest>

**MainActivity.java:**

package com.example.nirmal.db;

import android.support.v7.app.AppCompatActivity;

import android.os.Bundle;

import android.view.View;

import android.widget.Button;

import android.widget.EditText;

import android.widget.TextView;

import android.widget.Toast;

public class MainActivity extends AppCompatActivity {

Button insertButton,selectButton;

EditText name,id;

TextView output;

DBHandler handler;

@Override

protected void onCreate(Bundle savedInstanceState) {

super.onCreate(savedInstanceState);

setContentView(R.layout.activity\_main);

insertButton = (Button)findViewById(R.id.insertData);

selectButton = (Button)findViewById(R.id.getData);

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

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

output = (TextView)findViewById(R.id.Output);

handler = new DBHandler(getApplicationContext());

insertButton.setOnClickListener(new View.OnClickListener() {

@Override

public void onClick(View view) {

String sName = name.getText().toString().trim();

String sId = id.getText().toString().trim();

long result = handler.insertDate(sName,sId);

Toast.makeText(getApplicationContext(),result+" ",Toast.LENGTH\_SHORT).show();

}

});

selectButton.setOnClickListener(new View.OnClickListener() {

@Override

public void onClick(View view) {

String s = handler.getData();

output.setText(s);

}

});

}

}

**DBHandler.java:**

package com.example.nirmal.db;

import android.content.ContentValues;

import android.content.Context;

import android.database.Cursor;

import android.database.sqlite.SQLiteDatabase;

import android.database.sqlite.SQLiteOpenHelper;

public class DBHandler extends SQLiteOpenHelper{

private static final int DATABASE\_VERSION = 1;

private static final String DATABASE\_NAME = "SairamApp";

private static final String TABLE\_NAME = "InMatesDetail";

private static final String KEY\_INMATE\_NAME = "Name";

private static final String KEY\_INMATE\_ID = "ID";

public DBHandler(Context context){

super(context,DATABASE\_NAME,null,DATABASE\_VERSION);

}

@Override

public void onCreate(SQLiteDatabase sqLiteDatabase) {

String createQuery = "CREATE TABLE "+ TABLE\_NAME + "("+ KEY\_INMATE\_NAME + " TEXT," +KEY\_INMATE\_ID+ " TEXT)";

sqLiteDatabase.execSQL(createQuery);

}

@Override

public void onUpgrade(SQLiteDatabase sqLiteDatabase, int i, int i1) {

sqLiteDatabase.execSQL("DROP TABLE IF EXISTS"+TABLE\_NAME);

onCreate(sqLiteDatabase);

}

public long insertDate(String name,String ID){

ContentValues contentValues = new ContentValues();

contentValues.put(KEY\_INMATE\_NAME,name);

contentValues.put(KEY\_INMATE\_ID,ID);

SQLiteDatabase db = this.getWritableDatabase();

long output = db.insert(TABLE\_NAME,null,contentValues);

db.close();

return output;

}

public String getData(){

String selectQuery = "select \* from "+TABLE\_NAME;

SQLiteDatabase sqLiteDatabase = this.getReadableDatabase();

Cursor cr = sqLiteDatabase.rawQuery(selectQuery,null);

int count = cr.getCount();

StringBuilder sb = new StringBuilder();

int i = 0;

cr.moveToFirst();

while(i<count){

sb.append("Name : "+cr.getString(0)+" ; ID : "+cr.getString(1)+"\n");

cr.moveToNext();

i++;

}

cr.close();

return sb.toString();

} }

**ACTIVITY\_MAIN.XML:**

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

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

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

android:id="@+id/activity\_main"

android:layout\_width="match\_parent"

android:layout\_height="match\_parent"

android:paddingBottom="@dimen/activity\_vertical\_margin"

android:paddingLeft="@dimen/activity\_horizontal\_margin"

android:paddingRight="@dimen/activity\_horizontal\_margin"

android:paddingTop="@dimen/activity\_vertical\_margin"

tools:context="com.example.nirmal.db.MainActivity"

android:orientation="vertical">

<EditText

android:layout\_width="wrap\_content"

android:layout\_height="wrap\_content"

android:layout\_gravity="center"

android:id="@+id/name"

android:hint="Enter Name"/>

<EditText

android:layout\_width="wrap\_content"

android:layout\_height="wrap\_content"

android:layout\_gravity="center"

android:id="@+id/id"

android:inputType="number"

android:hint="Enter id"/>

<Button

android:layout\_width="wrap\_content"

android:layout\_height="wrap\_content"

android:layout\_gravity="center"

android:text="Insert Data"

android:id="@+id/insertData"/>

<Button

android:layout\_width="wrap\_content"

android:layout\_height="wrap\_content"

android:layout\_gravity="center"

android:text="GET Data"

android:id="@+id/getData"/>

<TextView

android:layout\_width="wrap\_content"

android:layout\_height="wrap\_content"

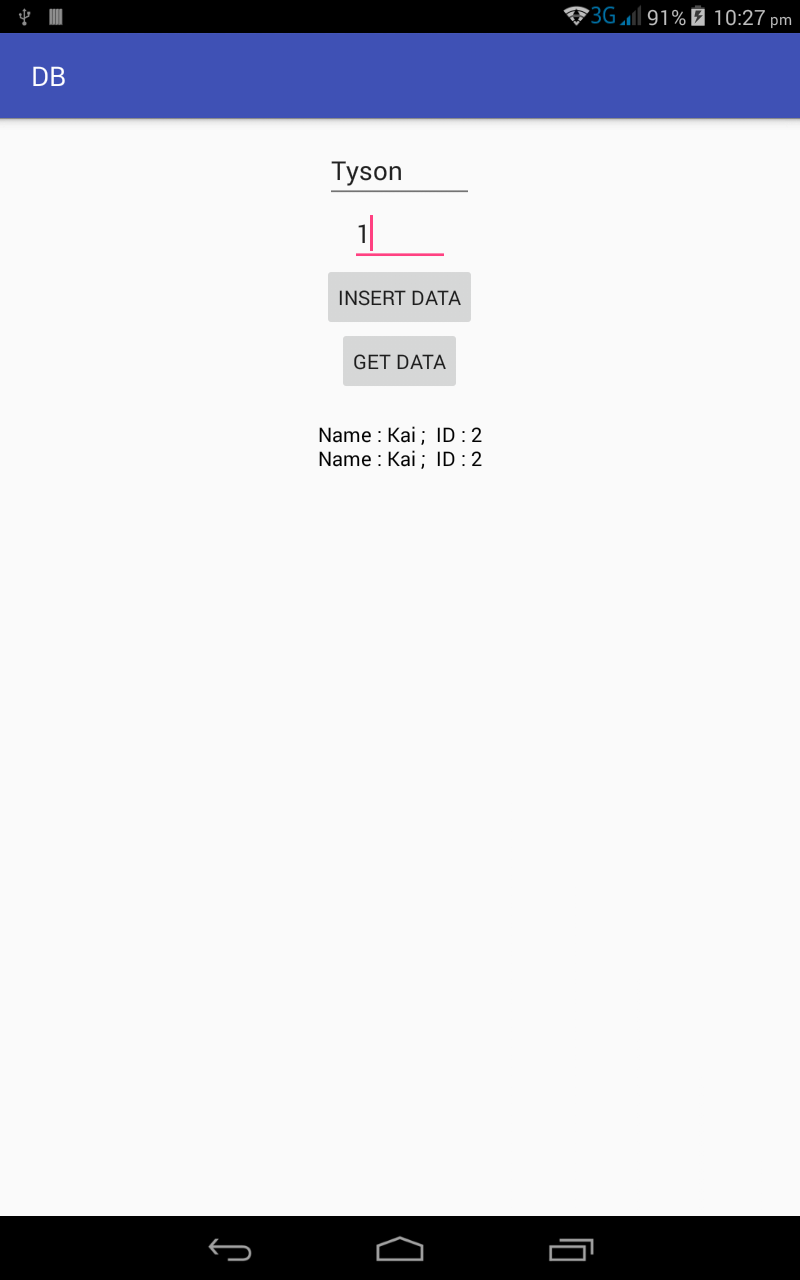
android:textColor="#000000"

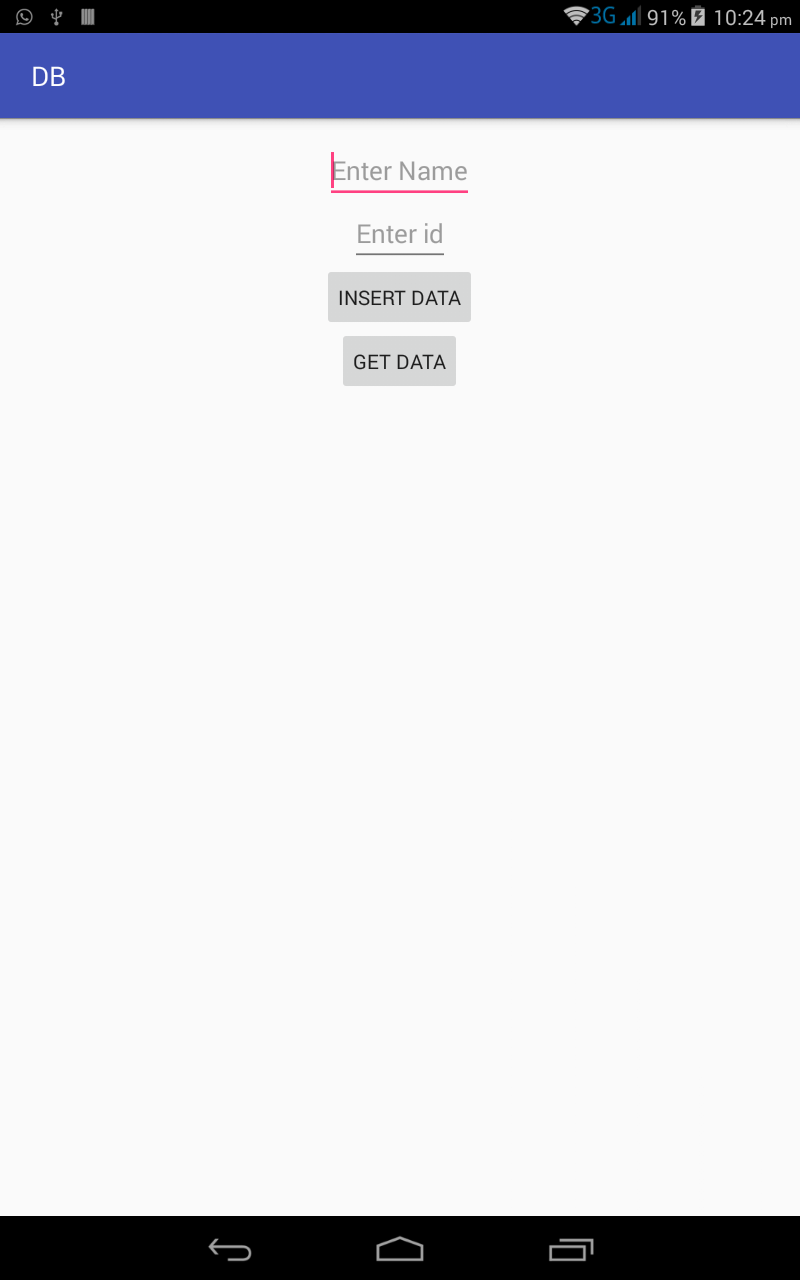
android:layout\_marginTop="20dp"

android:layout\_gravity="center"

android:id="@+id/Output"/>

</LinearLayout>





**RESULT**

The above project on development of an application that makes use of database was verified and executed successfully.

**EXNO .6 DEVELOP AN APPLICATION THAT MAKES USE OF RSS FEED**

**DATE:**

**AIM:**

To develop an application that makes use of RSS Feed

**PROCEDURE:**

* Open Genymotion shell and click **Start.**
* Open Android Studio and then click on **File->New->New project**.
* Then type the Application name as **“RSS″** and click **Next.**
* Then select the **Minimum SDK**  and click **Next**.
* Then select the **Empty Activity**and click **Next.**
* Click F**inish**.
* In **content\_main.xml** type the code for components which are to be included in the application.
* In **MainActivity.java** open  **SimpleRSSReaderActivity** and make it extend **ListActivity** to display the headlines in the ListView by binding to an array which will hold our data, using the list view adapter.
* Add 2 instance variables: “headlines” and “links” of type **List**.
* In the **onCreate()** method pass an input stream to set the Input.
* override the **onListItemClick()** method, get the position of article in the ListView, retrieve the coresponding link, and pass the url of that article to ACTION\_VIEW intent which takes care further of displaying the web page.
* Finally, Android output is present in the Android emulator.

**SOURCE CODE:**

**MAIN\_ACTIVITY.JAVA:**

package com.example.dell.rss;  
  
import android.app.Activity;  
import android.content.Intent;  
import android.os.Bundle;  
  
import android.view.Menu;  
import android.view.MenuItem;  
import android.view.MotionEvent;  
import android.view.View;  
  
import android.widget.Button;  
import android.widget.EditText;  
import android.widget.TextView;  
  
import java.util.Set;  
  
public class MainActivity extends Activity {  
 EditText title,link,description;  
 Button b1,b2;  
 private String finalUrl="http://tutorialspoint.com/android/sampleXML.xml";  
 private HandleXML obj;  
  
 @Override  
 protected void onCreate(Bundle savedInstanceState) {  
 super.onCreate(savedInstanceState);  
 setContentView(R.layout.*activity\_main*);  
  
 title = (EditText) findViewById(R.id.*editText*);  
 link = (EditText) findViewById(R.id.*editText2*);  
 description = (EditText) findViewById(R.id.*editText3*);  
  
 b1=(Button)findViewById(R.id.*button*);  
 b2=(Button)findViewById(R.id.*button2*);  
 b1.setOnClickListener(new View.OnClickListener() {  
 @Override  
 public void onClick(View v) {  
 obj = new HandleXML(finalUrl);  
 obj.fetchXML();  
  
 while(obj.parsingComplete);  
 title.setText(obj.getTitle());  
 link.setText(obj.getLink());  
 description.setText(obj.getDescription());  
 }  
 });  
  
 b2.setOnClickListener(new View.OnClickListener() {  
 @Override  
 public void onClick(View v) {  
 Intent in=new Intent(MainActivity.this,second.class);  
 startActivity(in);  
 }  
 });  
 }  
  
 @Override  
 public boolean onCreateOptionsMenu(Menu menu) {  
 *// Inflate the menu; this adds items to the action bar if it is present.* getMenuInflater().inflate(R.menu.*menu\_main*, menu);  
 return true;  
 }  
  
 @Override  
 public boolean onOptionsItemSelected(MenuItem item) {  
 *// Handle action bar item clicks here. The action bar will  
 // automatically handle clicks on the Home/Up button, so long  
 // as you specify a parent activity in AndroidManifest.xml.* int id = item.getItemId();  
  
 *//noinspection SimplifiableIfStatement* if (id == R.id.*action\_settings*) {  
 return true;  
 }  
 return super.onOptionsItemSelected(item);  
 }  
}

**HANDLEXML.JAVA:**

package com.example.dell.rss;  
import java.io.InputStream;  
import java.net.HttpURLConnection;  
import java.net.URL;  
  
import org.xmlpull.v1.XmlPullParser;  
import org.xmlpull.v1.XmlPullParserFactory;  
  
import android.util.Log;  
  
public class HandleXML {  
 private String title = "title";  
 private String link = "link";  
 private String description = "description";  
 private String urlString = null;  
 private XmlPullParserFactory xmlFactoryObject;  
 public volatile boolean parsingComplete = true;  
  
 public HandleXML(String url){  
 this.urlString = url;  
 }  
  
 public String getTitle(){  
 return title;  
 }  
  
 public String getLink(){  
 return link;  
 }  
  
 public String getDescription(){  
 return description;  
 }  
  
 public void parseXMLAndStoreIt(XmlPullParser myParser) {  
 int event;  
 String text=null;  
  
 try {  
 event = myParser.getEventType();  
  
 while (event != XmlPullParser.*END\_DOCUMENT*) {  
 String name=myParser.getName();  
  
 switch (event){  
 case XmlPullParser.*START\_TAG*:  
 break;  
  
 case XmlPullParser.*TEXT*:  
 text = myParser.getText();  
 break;  
  
 case XmlPullParser.*END\_TAG*:  
  
 if(name.equals("title")){  
 title = text;  
 }  
  
 else if(name.equals("link")){  
 link = text;  
 }  
  
 else if(name.equals("description")){  
 description = text;  
 }  
  
 else{  
 }  
  
 break;  
 }  
  
 event = myParser.next();  
 }  
  
 parsingComplete = false;  
 }  
  
 catch (Exception e) {  
 e.printStackTrace();  
 }  
 }  
  
 public void fetchXML(){  
 Thread thread = new Thread(new Runnable(){  
 @Override  
 public void run() {  
  
 try {  
 URL url = new URL(urlString);  
 HttpURLConnection conn = (HttpURLConnection) url.openConnection();  
  
 conn.setReadTimeout(10000 */\* milliseconds \*/*);  
 conn.setConnectTimeout(15000 */\* milliseconds \*/*);  
 conn.setRequestMethod("GET");  
 conn.setDoInput(true);  
  
 *// Starts the query* conn.connect();  
 InputStream stream = conn.getInputStream();  
  
 xmlFactoryObject = XmlPullParserFactory.*newInstance*();  
 XmlPullParser myparser = xmlFactoryObject.newPullParser();  
  
 myparser.setFeature(XmlPullParser.*FEATURE\_PROCESS\_NAMESPACES*, false);  
 myparser.setInput(stream, null);  
  
 parseXMLAndStoreIt(myparser);  
 stream.close();  
 }  
  
 catch (Exception e) {  
 }  
 }  
 });  
 thread.start();  
 }  
}

**SECOND.JAVA:**

package com.example.dell.rss;  
  
import android.app.Activity;  
import android.os.Bundle;  
import android.webkit.WebView;  
  
public class second extends Activity {  
 @Override  
 protected void onCreate(Bundle savedInstanceState) {  
 super.onCreate(savedInstanceState);  
 setContentView(R.layout.*activity\_second*);  
 WebView w1=(WebView)findViewById(R.id.*webView*);  
 w1.loadUrl("http://tutorialspoint.com/android/sampleXML.xml");  
 }  
}

**ACTIVITY\_MAIN.XML:**

*<?*xml version="1.0" encoding="utf-8"*?>*<android.support.design.widget.CoordinatorLayout xmlns:android="http://schemas.android.com/apk/res/android"  
 xmlns:app="http://schemas.android.com/apk/res-auto"  
 xmlns:tools="http://schemas.android.com/tools"  
 android:layout\_width="match\_parent"  
 android:layout\_height="match\_parent"  
 android:fitsSystemWindows="true"  
 tools:context="com.example.dell.rss.MainActivity">  
  
 <android.support.design.widget.AppBarLayout  
 android:layout\_width="match\_parent"  
 android:layout\_height="wrap\_content"  
 android:theme="@style/AppTheme.AppBarOverlay">  
  
 <android.support.v7.widget.Toolbar  
 android:id="@+id/toolbar"  
 android:layout\_width="match\_parent"  
 android:layout\_height="?attr/actionBarSize"  
 android:background="?attr/colorPrimary"  
 app:popupTheme="@style/AppTheme.PopupOverlay" />  
  
 </android.support.design.widget.AppBarLayout>  
  
 <include layout="@layout/content\_main" />  
  
 <android.support.design.widget.FloatingActionButton  
 android:id="@+id/fab"  
 android:layout\_width="wrap\_content"  
 android:layout\_height="wrap\_content"  
 android:layout\_gravity="bottom|end"  
 android:layout\_margin="@dimen/fab\_margin"  
 android:src="@android:drawable/ic\_dialog\_email" />  
  
</android.support.design.widget.CoordinatorLayout>

**ACTIVITY\_SECOND.XML:**

*<?*xml version="1.0" encoding="utf-8"*?>*<android.support.design.widget.CoordinatorLayout xmlns:android="http://schemas.android.com/apk/res/android"  
 xmlns:app="http://schemas.android.com/apk/res-auto"  
 xmlns:tools="http://schemas.android.com/tools"  
 android:layout\_width="match\_parent"  
 android:layout\_height="match\_parent"  
 android:fitsSystemWindows="true"  
 tools:context="com.example.dell.rss.second">  
  
 <android.support.design.widget.AppBarLayout  
 android:layout\_width="match\_parent"  
 android:layout\_height="wrap\_content"  
 android:theme="@style/AppTheme.AppBarOverlay">  
  
 <android.support.v7.widget.Toolbar  
 android:id="@+id/toolbar"  
 android:layout\_width="match\_parent"  
 android:layout\_height="?attr/actionBarSize"  
 android:background="?attr/colorPrimary"  
 app:popupTheme="@style/AppTheme.PopupOverlay" />  
  
 </android.support.design.widget.AppBarLayout>  
  
 <include layout="@layout/content\_second" />  
  
 <android.support.design.widget.FloatingActionButton  
 android:id="@+id/fab"  
 android:layout\_width="wrap\_content"  
 android:layout\_height="wrap\_content"  
 android:layout\_gravity="bottom|end"  
 android:layout\_margin="@dimen/fab\_margin"  
 android:src="@android:drawable/ic\_dialog\_email" />  
  
</android.support.design.widget.CoordinatorLayout>

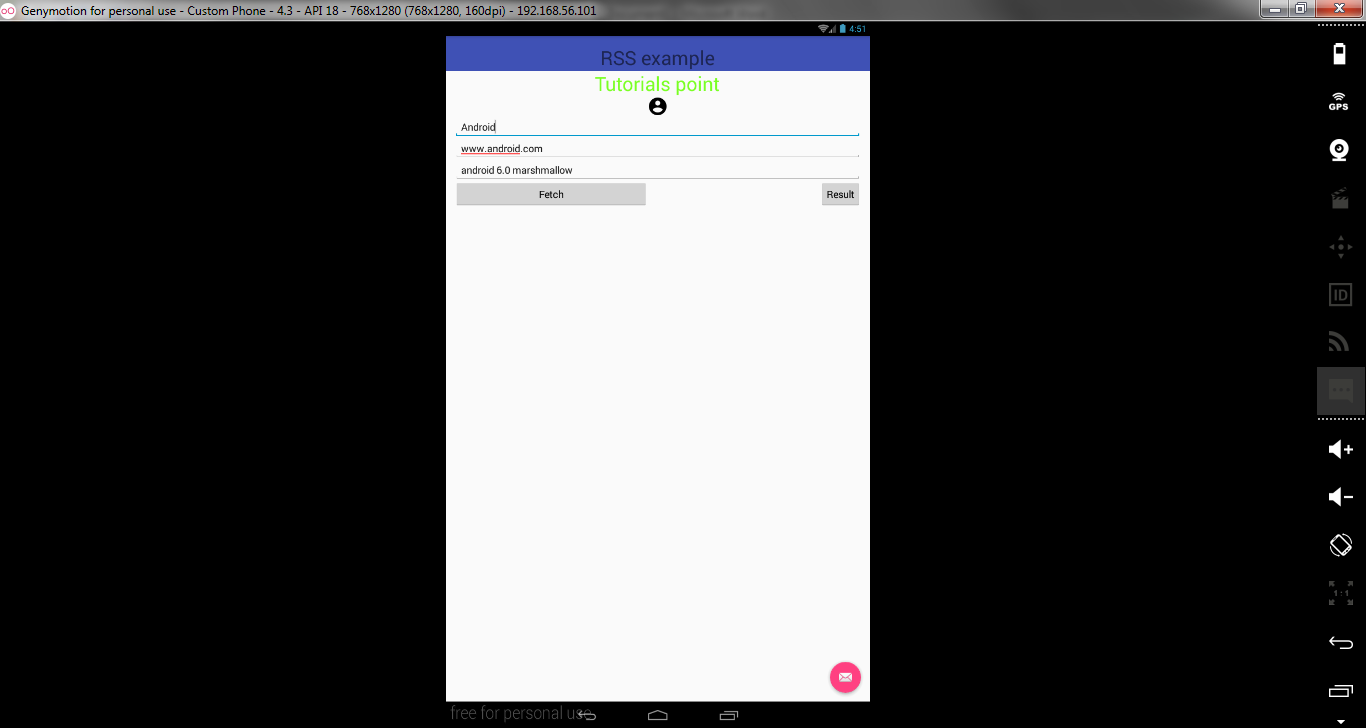
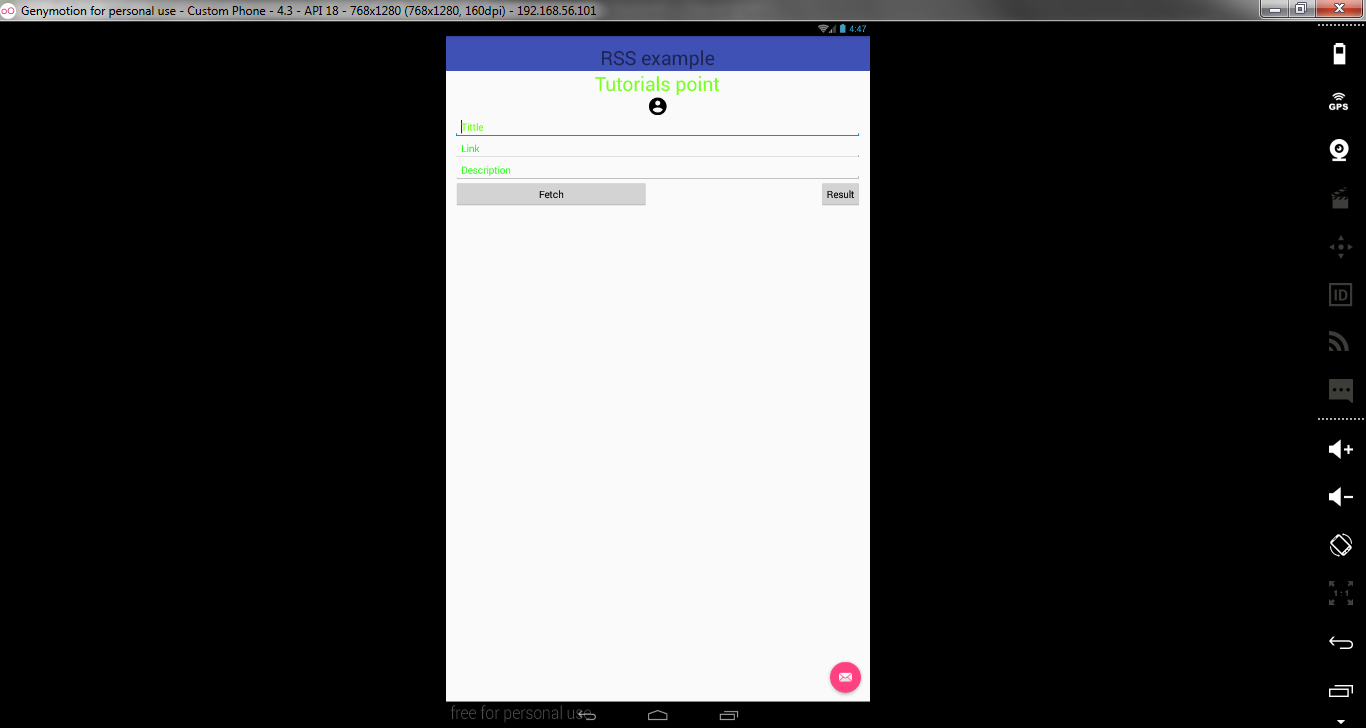
**CONTENT\_MAIN.XML:**

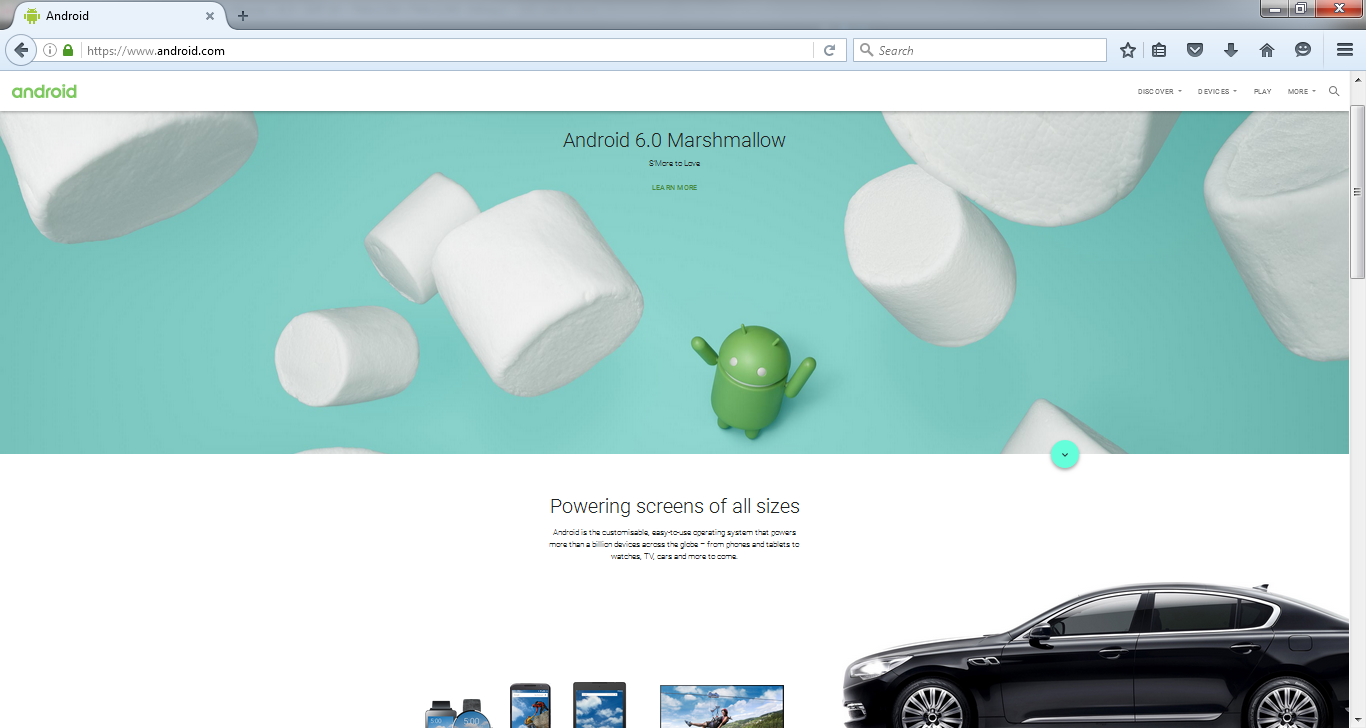
*<?*xml version="1.0" encoding="utf-8"*?>*<RelativeLayout xmlns:android="http://schemas.android.com/apk/res/android"  
 xmlns:tools="http://schemas.android.com/tools" android:layout\_width="match\_parent"  
 android:layout\_height="match\_parent" android:paddingLeft="@dimen/activity\_horizontal\_margin"  
 android:paddingRight="@dimen/activity\_horizontal\_margin"  
 android:paddingTop="@dimen/activity\_vertical\_margin"  
 android:paddingBottom="@dimen/activity\_vertical\_margin"  
 tools:context=".MainActivity"  
 android:transitionGroup="true">  
  
 <TextView android:text="RSS example" android:layout\_width="wrap\_content"  
 android:layout\_height="wrap\_content"  
 android:id="@+id/textview"  
 android:textSize="35dp"  
 android:layout\_alignParentTop="true"  
 android:layout\_centerHorizontal="true" />  
  
 <TextView  
 android:layout\_width="wrap\_content"  
 android:layout\_height="wrap\_content"  
 android:text="Tutorials point"  
 android:id="@+id/textView"  
 android:layout\_below="@+id/textview"  
 android:layout\_centerHorizontal="true"  
 android:textColor="#ff7aff24"  
 android:textSize="35dp" />  
  
 <ImageView  
 android:layout\_width="wrap\_content"  
 android:layout\_height="wrap\_content"  
 android:id="@+id/imageView"  
 android:src="@drawable/ic\_account\_circle\_black\_24dp"  
 android:layout\_below="@+id/textView"  
 android:layout\_centerHorizontal="true"  
 android:theme="@style/Base.TextAppearance.AppCompat" />  
  
 <EditText  
 android:layout\_width="wrap\_content"  
 android:layout\_height="wrap\_content"  
 android:id="@+id/editText"  
 android:layout\_below="@+id/imageView"  
 android:hint="Tittle"  
 android:textColorHint="#ff69ff0e"  
 android:layout\_alignParentRight="true"  
 android:layout\_alignParentEnd="true"  
 android:layout\_alignParentLeft="true"  
 android:layout\_alignParentStart="true" />  
  
 <EditText  
 android:layout\_width="wrap\_content"  
 android:layout\_height="wrap\_content"  
 android:id="@+id/editText2"  
 android:layout\_below="@+id/editText"  
 android:layout\_alignLeft="@+id/editText"  
 android:layout\_alignStart="@+id/editText"  
 android:textColorHint="#ff21ff11"  
 android:hint="Link"  
 android:layout\_alignRight="@+id/editText"  
 android:layout\_alignEnd="@+id/editText" />  
  
 <EditText  
 android:layout\_width="wrap\_content"  
 android:layout\_height="wrap\_content"  
 android:id="@+id/editText3"  
 android:layout\_below="@+id/editText2"  
 android:layout\_alignLeft="@+id/editText2"  
 android:layout\_alignStart="@+id/editText2"  
 android:hint="Description"  
 android:textColorHint="#ff33ff20"  
 android:layout\_alignRight="@+id/editText2"  
 android:layout\_alignEnd="@+id/editText2" />  
  
 <Button  
 android:layout\_width="wrap\_content"  
 android:layout\_height="wrap\_content"  
 android:text="Fetch"  
 android:id="@+id/button"  
 android:layout\_below="@+id/editText3"  
 android:layout\_alignParentLeft="true"  
 android:layout\_alignParentStart="true"  
 android:layout\_toLeftOf="@+id/imageView"  
 android:layout\_toStartOf="@+id/imageView" />  
  
 <Button  
 android:layout\_width="wrap\_content"  
 android:layout\_height="wrap\_content"  
 android:text="Result"  
 android:id="@+id/button2"  
 android:layout\_alignTop="@+id/button"  
 android:layout\_alignRight="@+id/editText3"  
 android:layout\_alignEnd="@+id/editText3" />  
  
</RelativeLayout>

**CONTENT\_SECOND.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">  
  
 <WebView  
 android:layout\_width="match\_parent"  
 android:layout\_height="match\_parent"  
 android:id="@+id/webView"  
 android:layout\_gravity="center\_horizontal" />  
</LinearLayout>

**OUTPUT:**





**RESULT:**

The above project on development of an application that makes use of rss feed was verified and executed successfully.

**EXNO .7 IMPLEMENT AN APPLICATION THAT IMPLEMENTS MULTITHREADING**

**DATE:**

**AIM:**

To develop an application that implements multithreading.

**PROCEDURE:**

* Open Genymotion shell and click **Start.**
* Open Android Studio and then click on **File->New->New project**.
* Then type the Application name as **“GUI″** and click **Next.**
* Then select the **Minimum SDK**  and click **Next**.
* Then select the **Empty Activity**and click **Next.**
* Click F**inish**.
* In **content\_main.xml** type the code for components which are to be included in the application.
* In **MainActivity.java** the timer sets the value for the processes selected by user.
* The threads are invoked based on the timer values.
* Now, click **run** option.
* Finally, Android output is present in the Android emulator.

**SOURCE CODE:**

**MAIN\_ACTIVITY.JAVA:**

package com.example.nirmal.threading;

import android.os.Handler;

import android.support.v7.app.AppCompatActivity;

import android.os.Bundle;

import android.view.View;

import android.widget.Button;

import android.widget.Toast;

public class MainActivity extends AppCompatActivity {

Runnable runnable ;

Handler mHandler;

Button myButton;

@Override

protected void onCreate(Bundle savedInstanceState) {

super.onCreate(savedInstanceState);

setContentView(R.layout.activity\_main);

myButton = (Button)findViewById(R.id.buttontouch);

mHandler = new Handler();

runnable = new Runnable() {

@Override

public void run() {

try{

Toast.makeText(MainActivity.this, "Thread 1", Toast.LENGTH\_SHORT).show();

}finally {

mHandler.postDelayed(runnable,2500);

}

}

};

myButton.setOnClickListener(new View.OnClickListener() {

@Override

public void onClick(View v) {

runnable.run();

}

});

}

}

**Activity\_Main.XML:**

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

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

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

android:layout\_width="match\_parent"

android:layout\_height="match\_parent"

android:paddingBottom="@dimen/activity\_vertical\_margin"

android:paddingLeft="@dimen/activity\_horizontal\_margin"

android:paddingRight="@dimen/activity\_horizontal\_margin"

android:paddingTop="@dimen/activity\_vertical\_margin"

tools:context="com.example.nirmal.threading.MainActivity">

<Button

android:layout\_width="wrap\_content"

android:layout\_height="wrap\_content"

android:id="@+id/buttontouch"

android:text="Start"/>

</RelativeLayout>

**MANIFEST:**

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

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

package="com.example.nirmal.threading">

<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>

</application>

</manifest>



**RESULT:**

The above project on implementation of an application that implements multithreading was verified and executed successfully.

**EXNO .8 DEVELOP A NATIVE APPLICATION THAT USES GPS LOCATION DATE: INFORMATION**

**AIM:**

To develop an application that uses GPS location information.

**PROCEDURE:**

* Open Genymotion shell and click **Start.**
* Open Android Studio and then click on **File->New->New project**.
* Then type the Application name as **“GUI″** and click **Next.**
* Then select the **Minimum SDK**  and click **Next**.
* Then select the **Empty Activity**and click **Next.**
* Click F**inish**.
* In **content\_main.xml** type the code for components which are to be included in the application.
* In **MainActivity.java** ,gps arrives at a feasible solution to aid the travelers from getting lost .
* It stores and directs the users to waypoints using latitude and longitude co-ordinates.
* Now, click **run** option.
* Finally, Android output is present in the Android emulator.

**SOURCE CODE**

**MANIFEST:**

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

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

package="com.example.nirmal.gpslocation">

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

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

<application

android:allowBackup="true"

android:icon="@mipmap/ic\_launcher"

android:label="@string/app\_name"

android:supportsRtl="true"

android:theme="@style/AppTheme">

<activity android:name=".MainActivity">

<intent-filter>

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

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

</intent-filter>

</activity>

</application>

</manifest>

**Activity\_Main.XML:**

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

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

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

android:layout\_width="match\_parent"

android:layout\_height="match\_parent"

android:paddingBottom="@dimen/activity\_vertical\_margin"

android:paddingLeft="@dimen/activity\_horizontal\_margin"

android:paddingRight="@dimen/activity\_horizontal\_margin"

android:paddingTop="@dimen/activity\_vertical\_margin"

tools:context="com.example.nirmal.gpslocation"

android:orientation="vertical">

<Button

android:layout\_width="wrap\_content"

android:layout\_height="wrap\_content"

android:id="@+id/clickbutton"

android:text="Get Location"

android:gravity="center"

android:layout\_gravity="center"/>

<TextView

android:layout\_width="wrap\_content"

android:layout\_height="wrap\_content"

android:text="Hello World!"

android:id="@+id/showtext"

android:layout\_alignParentTop="true"

android:layout\_alignParentLeft="true"

android:layout\_alignParentStart="true"

android:layout\_gravity="center"

android:layout\_marginTop="20dp"

android:textColor="#000000"/>

</LinearLayout>

**MainActivity.java:**

package com.example.nirmal.gpslocation;

import android.location.Location;

import android.location.LocationListener;

import android.location.LocationManager;

import android.support.v7.app.AppCompatActivity;

import android.os.Bundle;

import android.view.View;

import android.widget.Button;

import android.widget.TextView;

import android.widget.Toast;

import static android.content.Context.LOCATION\_SERVICE;

public class MainActivity extends AppCompatActivity implements LocationListener{

LocationManager loc;

Button button;

TextView textView;

@Override

protected void onCreate(Bundle savedInstanceState) {

super.onCreate(savedInstanceState);

setContentView(R.layout.activity\_main);

loc=(LocationManager)getSystemService(LOCATION\_SERVICE);

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

textView = (TextView)findViewById(R.id.showtext);

button.setOnClickListener(new View.OnClickListener() {

@Override

public void onClick(View view) {

if(loc == null);

else

try

{

Location location=loc.getLastKnownLocation(LocationManager.PASSIVE\_PROVIDER);

boolean flag = loc.isProviderEnabled(LocationManager.PASSIVE\_PROVIDER);

// Toast.makeText(MainActivity.this, flag+"", Toast.LENGTH\_SHORT).show();

if(location!=null)

onLocationChanged(location);

else {

textView.setText("Location is null");

}

}catch(SecurityException e) {

textView.setText(e.toString());

}

}

});

}

@Override

public void onProviderEnabled(String provider) {

Toast.makeText(this, "Enabled new provider " + provider,

Toast.LENGTH\_SHORT).show();

}

@Override

public void onProviderDisabled(String provider) {

Toast.makeText(this, "Disabled new provider " + provider,

Toast.LENGTH\_SHORT).show();

}

@Override

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

Toast.makeText(this, "Status changed " + provider,Toast.LENGTH\_SHORT).show();

}

@Override

public void onLocationChanged(Location location){

double longitude=location.getLongitude();

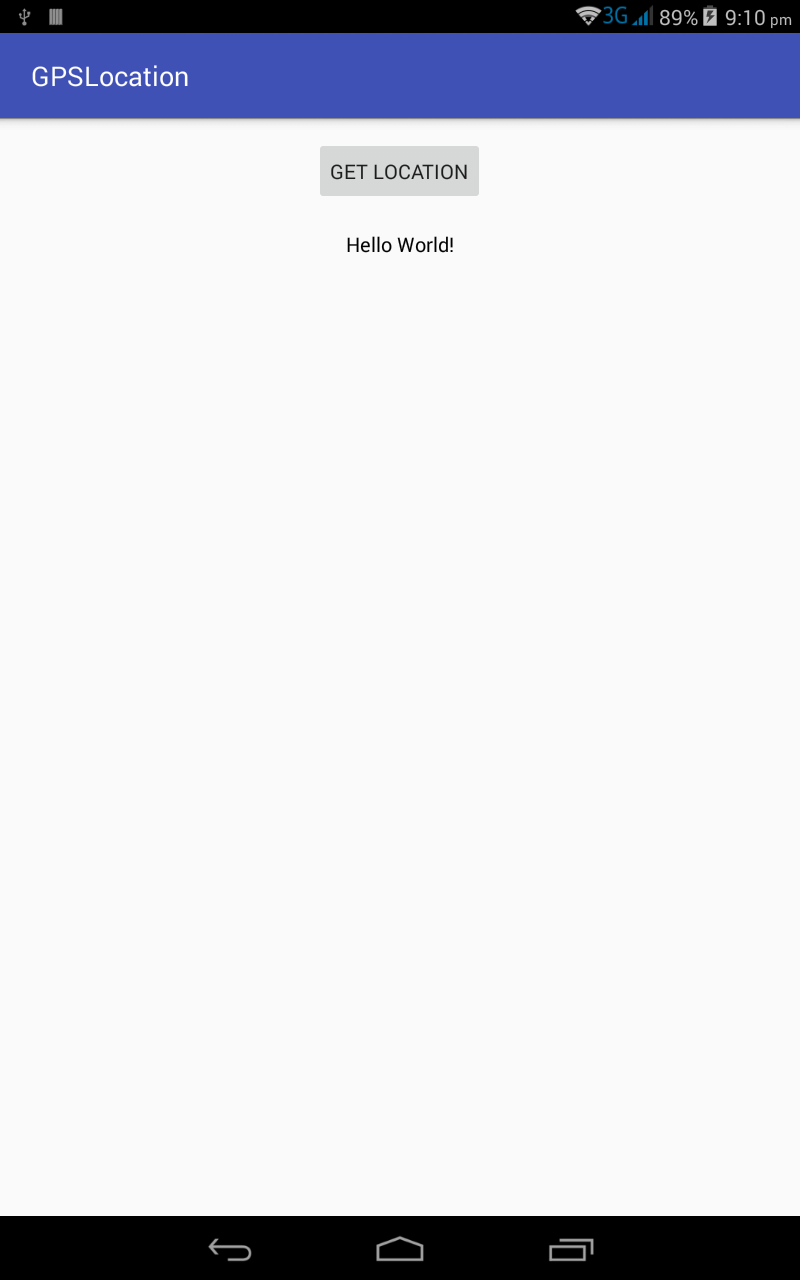
double latitude=location.getLatitude();

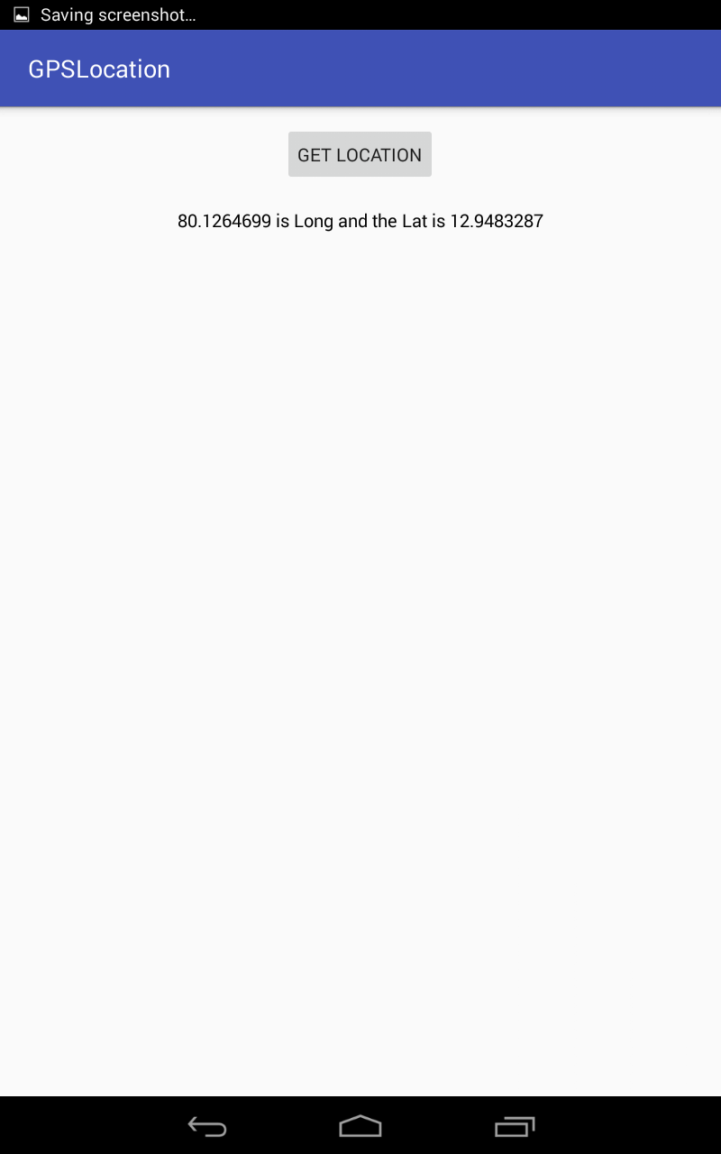
textView.setText(longitude + " is Long and the Lat is "+ latitude);

}

}

**OUTPUT:**





**RESULT:**

The above project on develop a native application that uses gps location date: information was verified and executed successfully.

**EXNO .9 IMPLEMENT AN APPLICATION THAT WRITES DATA TO SD CARD**

**DATE:**

**AIM:**

To implement an application that writes data to sd card.

**PROCEDURE:**

* Open Genymotion shell and click **Start.**
* Open Android Studio and then click on **File->New->New project**.
* Then type the Application name as **“GUI″** and click **Next.**
* Then select the **Minimum SDK**  and click **Next**.
* Then select the **Empty Activity**and click **Next.**
* Click F**inish**.
* In **content\_main.xml** type the code for components which are to be included in the application.
* In **MainActivity.java** ,the fetch file function is used to get the data from the user.
* now the data received from the user is copied into the SDCARD successfully using save file function.
* Now, click **run** option.
* Finally, Android output is present in the Android emulator.

**SOURCE CODE:**

**MANIFEST:**

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

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

package="com.example.nirmal.sdcard">

<uses-permission android:name="android.permission.WRITE\_EXTERNAL\_STORAGE"/>

<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>

</application>

</manifest>

**ACTIVITY\_MAIN.XML**

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

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

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

android:layout\_width="match\_parent"

android:layout\_height="match\_parent"

android:paddingBottom="@dimen/activity\_vertical\_margin"

android:paddingLeft="@dimen/activity\_horizontal\_margin"

android:paddingRight="@dimen/activity\_horizontal\_margin"

android:paddingTop="@dimen/activity\_vertical\_margin"

android:orientation="vertical"

tools:context="com.example.nirmal.sdcard.MainActivity">

<EditText

android:layout\_width="278dp"

android:layout\_height="wrap\_content"

android:id="@+id/editText" />

<Button

android:layout\_width="wrap\_content"

android:layout\_height="wrap\_content"

android:text="New Button"

android:id="@+id/button" />

</LinearLayout>

**MainActivity.java:**

package com.example.nirmal.sdcard;

import android.support.v7.app.AppCompatActivity;

import android.os.Bundle;

import android.view.View;

import android.widget.Button;

import android.widget.EditText;

import android.widget.Toast;

import java.io.File;

import java.io.FileOutputStream;

import java.io.OutputStreamWriter;

public class MainActivity extends AppCompatActivity {

EditText text;

Button myButton;

@Override

protected void onCreate(Bundle savedInstanceState) {

super.onCreate(savedInstanceState);

setContentView(R.layout.activity\_main);

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

text = (EditText)findViewById(R.id.editText);

myButton.setOnClickListener(new View.OnClickListener() {

@Override

public void onClick(View v) {

writeToSDCard();

}

});

}

private void writeToSDCard(){

try {

File myFile = new File("/sdcard/mysdfile.txt");

myFile.createNewFile();

FileOutputStream fOut = new FileOutputStream(myFile);

OutputStreamWriter myOutWriter = new OutputStreamWriter(fOut);

myOutWriter.append(text.getText());

myOutWriter.close();

fOut.close();

Toast.makeText(getBaseContext(),"Done writing SD 'mysdfile.txt'",Toast.LENGTH\_SHORT).show();

} catch (Exception e) {

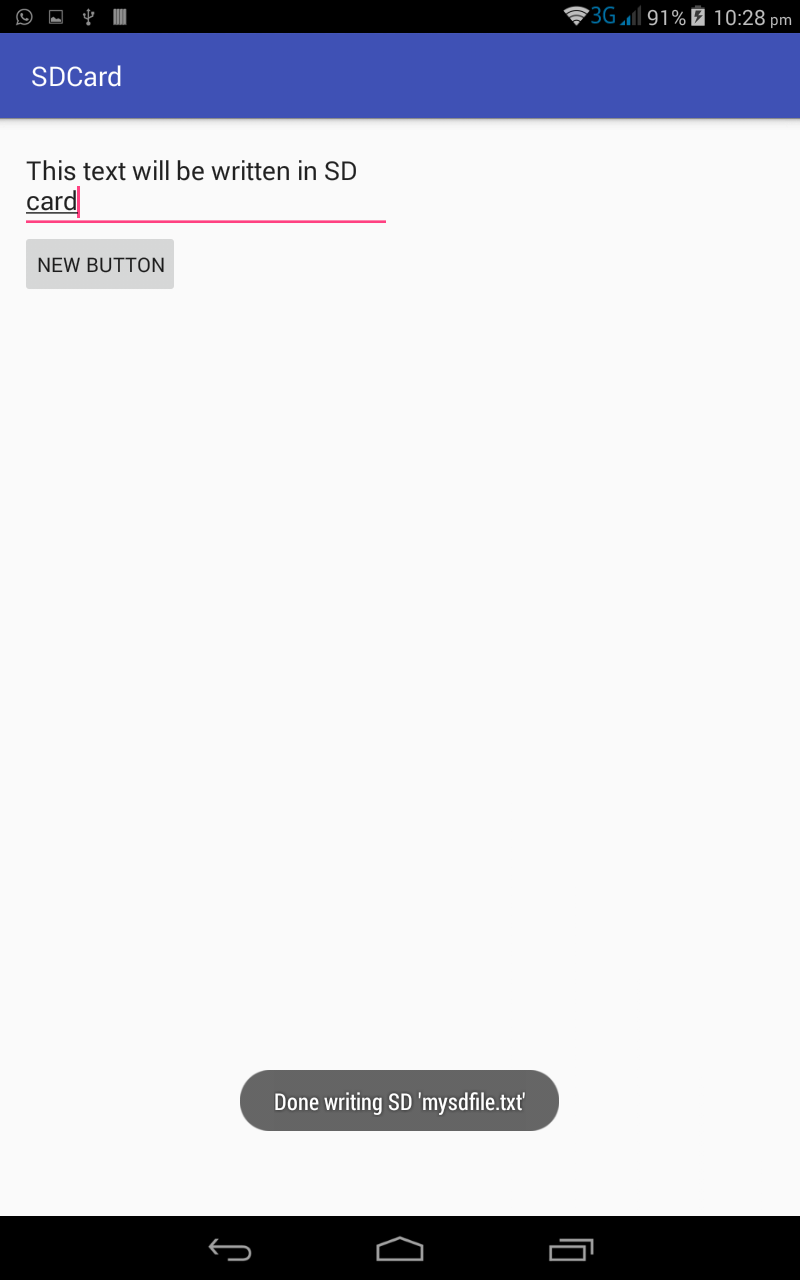
Toast.makeText(getBaseContext(), e.getMessage(), Toast.LENGTH\_SHORT).show();

}

}

}

**OUTPUT:**



**RESULT:**

The above project on implementation of an application that writes data to sd card

was verified and executed successfully.

**EXNO .10 IMPLEMENTATION OF AN APPLICATION THAT CREATES AN ALERT DATE: UPON CREATING A MESSAGE**

**AIM:**

Toimplementation of an application that creates an alert upon creating a message

**PROCEDURE:**

* Open Genymotion shell and click **Start.**
* Open Android Studio and then click on **File->New->New project**.
* Then type the Application name as **“RSS″** and click **Next.**
* Then select the **Minimum SDK**  and click **Next**.
* Then select the **Empty Activity** and click **Next.**
* Click F**inish**.
* In **content\_main.xml** type the code for components which are to be included in the application.
* In **MainActivity.java** open  **SimpleRSSReaderActivity** and make it extend **ListActivity** to display the headlines in the ListView by binding to an array which will hold our data, using the list view adapter.
* Add 2 instance variables: “headlines” and “links” of type **List**.
* In the **onCreate()** method pass an input stream to set the Input.
* override the **onListItemClick()** method, get the position of article in the ListView, retrieve the coresponding link, and pass the url of that article to ACTION\_VIEW intent which takes care further of displaying the web page.
* Finally, Android output is present in the Android emulator.

**SOURCE CODE:**

**MAIN\_ACTIVITY.JAVA:**  
package com.example.dell.alertmessage;  
  
import android.os.Bundle;  
import android.app.Activity;  
import android.view.Menu;  
  
public class MainActivity extends Activity  
{  
 */\*\*10. Implement an application that creates an alert upon receiving a message.  
 create notification \*\*/  
  
 //Get these permissions in Manifest.xml file  
 /\*\*  
 uses-permission android:name="android.permission.RECEIVE\_SMS"  
 uses-permission android:name="android.permission.READ\_SMS"  
 uses-permission android:name="android.permission.SEND\_SMS"  
 uses-permission android:name="android.permission.READ\_PROFILE"  
 \*\*/* @Override  
 protected void onCreate(Bundle savedInstanceState)  
 {  
 super.onCreate(savedInstanceState);  
 setContentView(R.layout.*activity\_main*);  
  
 }  
  
 @Override  
 public boolean onCreateOptionsMenu(Menu menu) {  
 *// Inflate the menu; this adds items to the action bar if it is present.* getMenuInflater().inflate(R.menu.*menu\_main*, menu);  
 return true;  
 }  
  
}

**SMS RECEIVING BROADCAST.JAVA:**

package com.example.dell.alertmessage;  
import android.annotation.SuppressLint;  
import android.annotation.TargetApi;  
import android.app.Notification;  
import android.app.NotificationManager;  
import android.content.BroadcastReceiver;  
import android.content.Context;  
import android.content.Intent;  
import android.os.Build;  
import android.os.Bundle;  
import android.telephony.SmsManager;  
import android.telephony.SmsMessage;  
import android.widget.Toast;  
  
@SuppressLint("NewApi")  
public class SmsReceivingBroadcast extends BroadcastReceiver  
{  
  
 String phonenum;  
 SmsManager sms;  
  
 @TargetApi(Build.VERSION\_CODES.*HONEYCOMB*)  
 @SuppressLint("NewApi")  
 @Override  
 public void onReceive(Context context, Intent intent)  
 {  
 *// TODO Auto-generated method stub* Bundle b = intent.getExtras();  
 SmsMessage msg[] = null;  
  
 String str = "";  
 *//If bundle is not empty i.e, bundle has a message* if (b != null)  
 {  
 Object[] pdus = (Object[]) b.get("pdus");  
 msg = new SmsMessage[pdus.length];  
 for (int i = 0; i < msg.length; i++)  
 {  
  
 msg[i] = SmsMessage.*createFromPdu*((byte[]) pdus[i]);  
 str = msg[i].getMessageBody().toString();  
 *//Getting the Message Body* str += msg[i].getMessageBody().toString();  
 *//Getting the Phone number* phonenum = msg[i].getOriginatingAddress().toString();  
  
 *//Accessing Notification Manager for handling Notification* NotificationManager notificationManager = (NotificationManager) context.getSystemService(Context.*NOTIFICATION\_SERVICE*);  
 *//Accessing Notification Builder for building notification* Notification.Builder builder = new Notification.Builder(context);  
  
 *//Set the Notification Icon  
 //builder.setSmallIcon(R.drawable);  
  
 //Notification Title* builder.setContentTitle("SMS Notification");  
  
 *//Notification Message* builder.setContentText("U have received a message");  
  
 *//Notofocation Ticker Text* builder.setTicker("SMS");  
  
 *//Notification Auto cancel for dismissing the notification after viewing* builder.setAutoCancel(true);  
  
 Notification notification = builder.getNotification();  
 notificationManager.notify(R.drawable.*ic\_account\_circle\_black\_24dp*,notification);  
  
 *//Showing Toast Message* Toast.*makeText*(context, "message:" + str + "phone:" + phonenum,Toast.*LENGTH\_LONG*).show();  
  
  
 }  
  
 }  
 }  
  
}

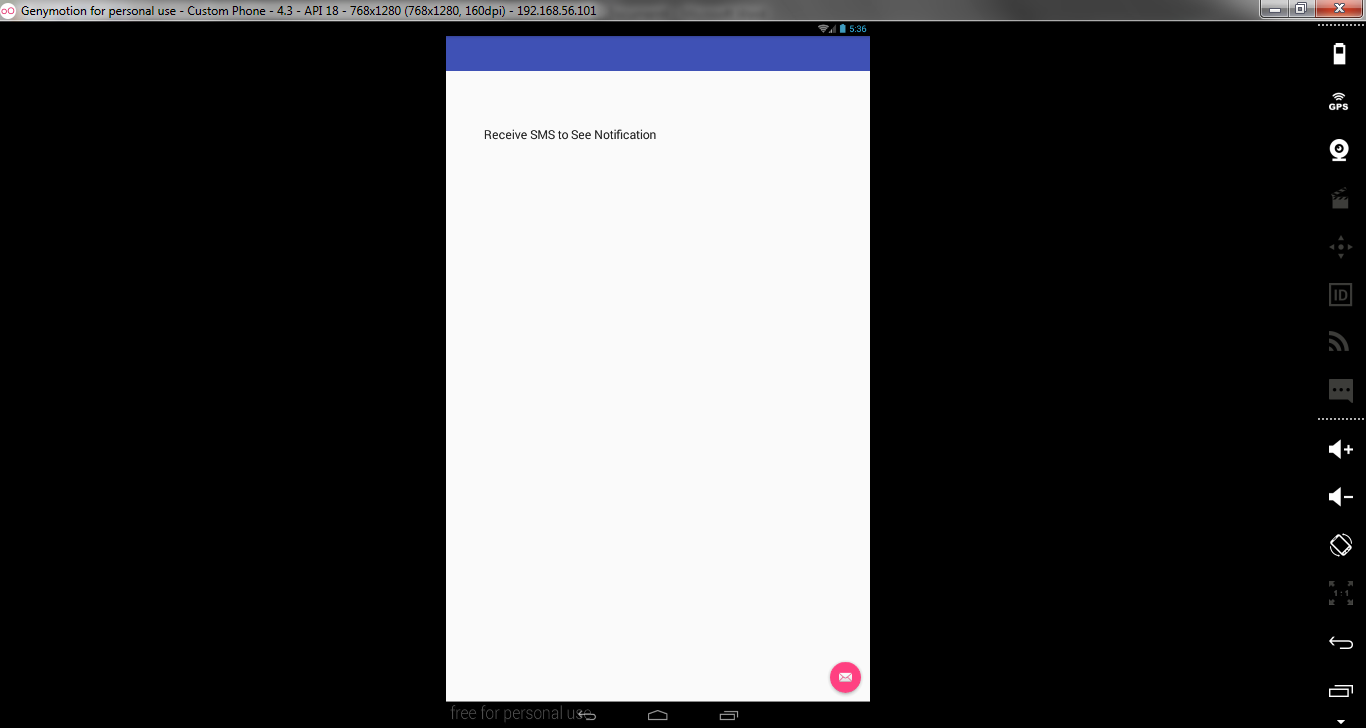
**ACTIVITY\_MAIN.XML:**

*<?*xml version="1.0" encoding="utf-8"*?>*<android.support.design.widget.CoordinatorLayout xmlns:android="http://schemas.android.com/apk/res/android"  
 xmlns:app="http://schemas.android.com/apk/res-auto"  
 xmlns:tools="http://schemas.android.com/tools"  
 android:layout\_width="match\_parent"  
 android:layout\_height="match\_parent"  
 android:fitsSystemWindows="true"  
 tools:context="com.example.dell.alertmessage.MainActivity">  
  
 <android.support.design.widget.AppBarLayout  
 android:layout\_width="match\_parent"  
 android:layout\_height="wrap\_content"  
 android:theme="@style/AppTheme.AppBarOverlay">  
  
 <android.support.v7.widget.Toolbar  
 android:id="@+id/toolbar"  
 android:layout\_width="match\_parent"  
 android:layout\_height="?attr/actionBarSize"  
 android:background="?attr/colorPrimary"  
 app:popupTheme="@style/AppTheme.PopupOverlay" />  
  
 </android.support.design.widget.AppBarLayout>  
  
 <include layout="@layout/content\_main" />  
  
 <android.support.design.widget.FloatingActionButton  
 android:id="@+id/fab"  
 android:layout\_width="wrap\_content"  
 android:layout\_height="wrap\_content"  
 android:layout\_gravity="bottom|end"  
 android:layout\_margin="@dimen/fab\_margin"  
 android:src="@android:drawable/ic\_dialog\_email" />  
  
</android.support.design.widget.CoordinatorLayout>

**CONTENT\_MAIN.XML:**

*<?*xml version="1.0" encoding="utf-8"*?>*<RelativeLayout xmlns:android="http://schemas.android.com/apk/res/android"  
 xmlns:tools="http://schemas.android.com/tools"  
 android:layout\_width="match\_parent"  
 android:layout\_height="match\_parent"  
 android:paddingBottom="@dimen/activity\_vertical\_margin"  
 android:paddingLeft="@dimen/activity\_horizontal\_margin"  
 android:paddingRight="@dimen/activity\_horizontal\_margin"  
 android:paddingTop="@dimen/activity\_vertical\_margin"  
 tools:context=".MainActivity" >  
  
 <TextView  
 android:id="@+id/textView1"  
 android:layout\_width="wrap\_content"  
 android:layout\_height="wrap\_content"  
 android:layout\_alignParentLeft="true"  
 android:layout\_alignParentTop="true"  
 android:layout\_marginLeft="53dp"  
 android:layout\_marginTop="147dp"  
 android:text="Receive SMS to See Notification "  
 android:textAppearance="?android:attr/textAppearanceLarge" />  
  
</RelativeLayout>

**OUTPUT:**



**RESULT:**

The above project on implementation of an application that creates an alert date: upon creating a message was verified and executed successfully.

**EXNO .11 WRITE A MOBILE APPLICATION THAT CREATES ALARM CLOCK**

**DATE:**

**AIM:**

To develop a mobile application that creates alarm clock.

**PROCEDURE:**

* Open Genymotion shell and click **Start.**
* Open Android Studio and then click on **File->New->New project**.
* Then type the Application name as **“GUI″** and click **Next.**
* Then select the **Minimum SDK**  and click **Next**.
* Then select the **Empty Activity**and click **Next.**
* Click F**inish**.
* In **content\_main.xml** type the code for components which are to be included in the application.
* In **MainActivity.java** get the time for which the alarm to be set.
* Now, the set alarm function sets up the wakeup call for the given time based on the time available in the calendar .
* Now, click **run** option.
* Finally, Android output is present in the Android emulator.

**SOURCE CODE:**

**MANIFEST:**

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

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

package="com.example.nirmal.alarm">

<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=".handleBroadcast"/>

<activity android:name=".myBroadcastReceiver"/>

</application>

</manifest>

**Acitivity\_Main.XML:**

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

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

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

android:layout\_width="match\_parent"

android:layout\_height="match\_parent"

android:paddingBottom="@dimen/activity\_vertical\_margin"

android:paddingLeft="@dimen/activity\_horizontal\_margin"

android:paddingRight="@dimen/activity\_horizontal\_margin"

android:paddingTop="@dimen/activity\_vertical\_margin"

tools:context="com.example.admin.alarm.MainActivity">

<Button

android:layout\_width="wrap\_content"

android:layout\_height="wrap\_content"

android:id="@+id/myButton"

android:text="Set the alarm" />

</RelativeLayout>

**MainActivity.java:**

package com.example.nirmal.alarm;

import android.app.AlarmManager;

import android.app.PendingIntent;

import android.content.Intent;

import android.support.v7.app.AppCompatActivity;

import android.os.Bundle;

import android.view.View;

import android.widget.Button;

import android.widget.Toast;

public class MainActivity extends AppCompatActivity {

Button but;

@Override

protected void onCreate(Bundle savedInstanceState) {

super.onCreate(savedInstanceState);

setContentView(R.layout.activity\_main);

but = (Button)findViewById(R.id.myButton);

but.setOnClickListener(new View.OnClickListener() {

@Override

public void onClick(View v) {

setTheAlarm();

}

});

}

public void setTheAlarm(){

Intent i = new Intent(MainActivity.this,handleBroadcast.class);

PendingIntent pendingIntent = PendingIntent.getBroadcast(this.getApplicationContext(), 2, i, 0);

AlarmManager alarmManager = (AlarmManager)getSystemService(ALARM\_SERVICE);

Toast.makeText(getApplicationContext(),"Alarm set you can close the app",Toast.LENGTH\_LONG).show();

alarmManager.set(AlarmManager.RTC\_WAKEUP, System.currentTimeMillis()+ 5000, pendingIntent);

}

}

**handleBroadcast.java:**

package com.example.nirmal.alarm;

import android.content.BroadcastReceiver;

import android.content.Context;

import android.content.Intent;

import android.widget.Toast;

public class handleBroadcast extends BroadcastReceiver {

@Override

public void onReceive(Context context, Intent intent) {

Intent i = new Intent(context,myBroadcastReceiver.class);

i.setFlags(Intent.FLAG\_ACTIVITY\_NEW\_TASK);

context.startActivity(i);

} }

**myBroadcastReceiver.java:**

package com.example.nirmal.alarm;

import android.app.Activity;

import android.content.Context;

import android.content.DialogInterface;

import android.content.Intent;

import android.os.Bundle;

import android.support.annotation.Nullable;

import android.support.v7.app.AlertDialog;

import android.util.AttributeSet;

import android.view.View;

import android.widget.Toast;

import org.json.JSONArray;

import org.json.JSONException;

import org.json.JSONObject;

import java.util.HashMap;

import java.util.Map;

public class myBroadcastReceiver extends Activity {

@Override

protected void onCreate(Bundle savedInstanceState) {

super.onCreate(savedInstanceState);

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

builder.setMessage("This is the alarm you created").setCancelable(false).setPositiveButton("Yes", new DialogInterface.OnClickListener() {

public void onClick(DialogInterface dialog, int id) {

Intent i = new Intent(myBroadcastReceiver.this,MainActivity.class);

startActivity(i);

dialog.cancel();

}

}).

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

public void onClick(DialogInterface dialog, int id) {

dialog.cancel();

}

});

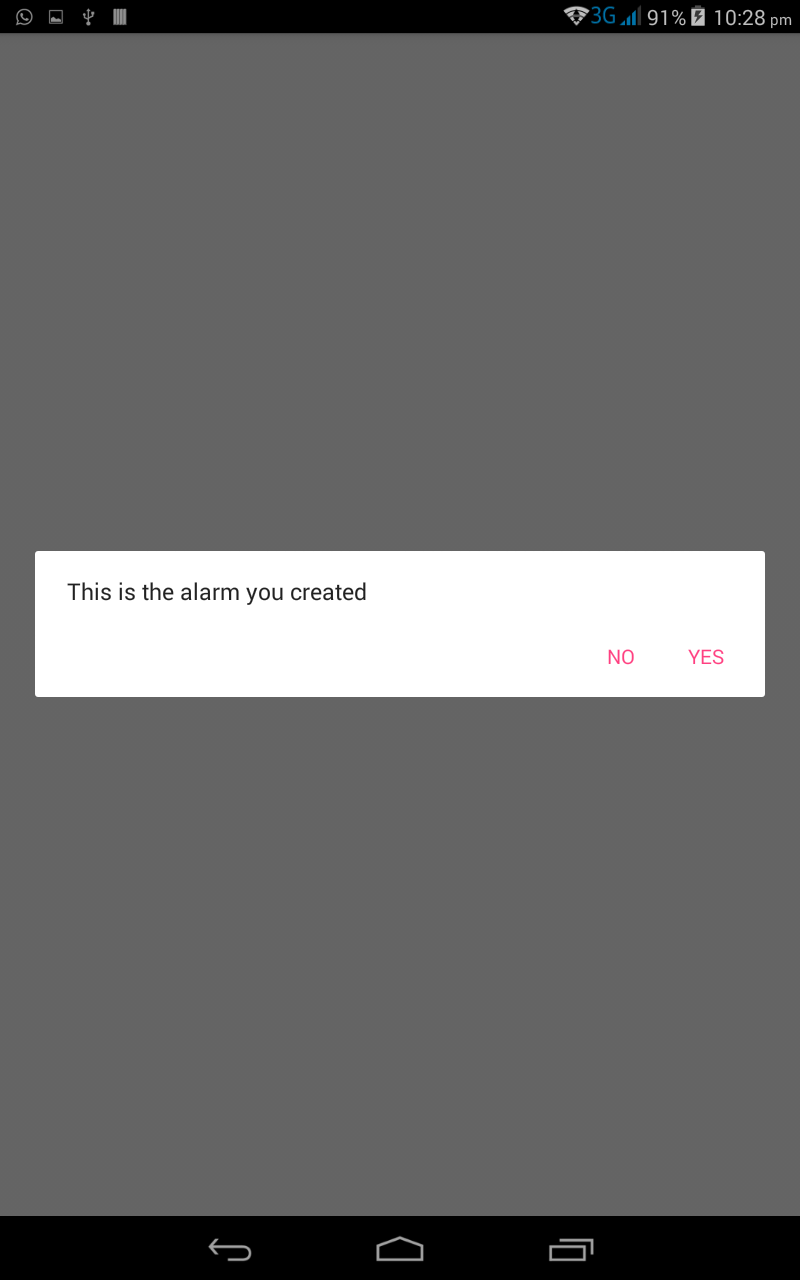
AlertDialog alert = builder.create();

alert.show();

} }

}

**OUTPUT:**



**RESULT:**

The above project on writing a mobile application that creates alarm clock was verified and executed successfully.