

FEDERAL INSTITUTE OF SCIENCE AND TECHNOLOGY (FISAT)TM

HORMIS NAGAR, MOOKKANNOOR, ANGAMALY-683577



FOCUS ON EXCELLENCE

20MCA243 MOBILE APPLICATION DEVELOPMENT LAB LABORATORY RECORD

Name: MINNU JINSON

Branch: MASTER OF COMPUTER APPLICATIONS

Semester: 3 Batch: B Roll No: 18

MARCH 2022

FEDERAL INSTITUTE OF SCIENCE AND TECHNOLOGY (FISAT)TM

HORMIS NAGAR, MOOKKANNOOR, ANGAMALY-683577



FOCUS ON EXCELLENCE

CERTIFICATE

*This is to certify that this is a Bonafide record of the Practical work done by **MINNU JINSON** in the **20MCA243 MOBILE APPLICATION DEVELOPMENT LAB** Laboratory towards the partial fulfilment for the award of the Master Of Computer Applications during the academic year 2021-2022.*

Signature of Staff in Charge

Name:

Signature of H O D

Name:

Date of University practical examination

Signature of
Internal Examiner

Signature of
External Examiner

CONTENT

SI No	Date :	Name of Experiment:	Page No:	Signature of Staff –In – Charge:
1	18/11/2021	Create a simple calculator	1	
2	25/11/2021	Concatenate the two string(The resulted string color is green).	6	
3	02/12/2021	Factorial of given number	10	
4	09/12/2021	Draw different shape and fill with different color	14	
5	16/12/2021	Draw smiley	16	
6	06/01/2022	Intents	22	
7	20/01/2022	Storing data into internal phone memory	27	

8	03/02/2022	Demonstrate GrideView	34	
9	03/02/2022	Demonstrate ImageView and GrideView	37	
10	10/02/2022	Demonstration of Toggle button	43	
11	10/02/2022	Demonstration of Option menu	46	
12	17/02/2022	Spinner widget	50	
13	24/02/2022	Database application using SQLite	55	

PROGRAM 1:

Create a Simple Calculator for demonstrating the basic arithmetic operations (+ , - , * , /)

PROCEDURE:

step 1: Start

step 2: Create the xml file .Drag and drop the 2 edittext and 4 button for the arithmetic calculation such as addition, subtraction, division, multiplication. then drag and drop the textview field to view the calculated result.

Step 3: Create the java code file to perform the calculation its initialize the edit Test, button and textview then create the object of each one.

Step 4: Read the two number and it pass to the switch case do the neccesaryoperation.

Step 5: Display the result on the textview field. Step

6: Stop.

MainActivity.java:

```
package com.example.calculator;
```

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

```
public class MainActivity extends AppCompatActivity implements View.OnClickListener{
```

```
    EditText etNum1;  
    EditText etNum2;
```

```
    Button btnAdd;  
    Button btnSub;  
    Button btnMult;  
    Button btnDiv;
```

```
    TextView tvResult;
```

```
    String oper = "";
```

```
    /** Called when the activity is first created. */  
    @Override  
    public void onCreate(Bundle savedInstanceState) {  
        super.onCreate(savedInstanceState);  
        setContentView(R.layout.activity_main);
```

```
// find the elements
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);

tvResult = (TextView) findViewById(R.id.tvResult);

// set a listener
btnAdd.setOnClickListener(this);
btnSub.setOnClickListener(this);
btnMult.setOnClickListener(this);
btnDiv.setOnClickListener(this);

}

@Override
public void onClick(View v) {
    // TODO Auto-generated method stub
    float num1 = 0;
    float num2 = 0;
    float result = 0;

    // check if the fields are empty
    if (TextUtils.isEmpty(etNum1.getText().toString())
        || TextUtils.isEmpty(etNum2.getText().toString())) {
        return;
    }
    // read EditText and fill variables with numbers
    num1 = Float.parseFloat(etNum1.getText().toString());
    num2 = Float.parseFloat(etNum2.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.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;
    default:
        break;
}

// form the output line
tvResult.setText(num1 + " " + oper + " " + num2 + " = " + result);
}
}

```

Activity_main.xml:

```

<?xml version="1.0" encoding="utf-8"?>
<LinearLayout
    xmlns:android="http://schemas.android.com/apk/res/android"
    android:orientation="vertical" android:layout_width="fill_parent"
    android:layout_height="fill_parent"
    android:weightSum="1">
    <LinearLayout
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:id="@+id/linearLayout1"
        android:layout_marginLeft="10pt"
        android:layout_marginRight="10pt"
        android:layout_marginTop="3pt">
        <EditText
            android:layout_weight="1"
            android:layout_height="wrap_content"
            android:layout_marginRight="5pt"
            android:id="@+id/etNum1"
            android:layout_width="match_parent"
            android:inputType="numberDecimal">
        </EditText>
        <EditText
            android:layout_height="wrap_content"
            android:layout_weight="1"
            android:layout_marginLeft="5pt"
            android:id="@+id/etNum2"
            android:layout_width="match_parent"
            android:inputType="numberDecimal">
        </EditText>
    </LinearLayout>
    <LinearLayout
        android:layout_width="match_parent"

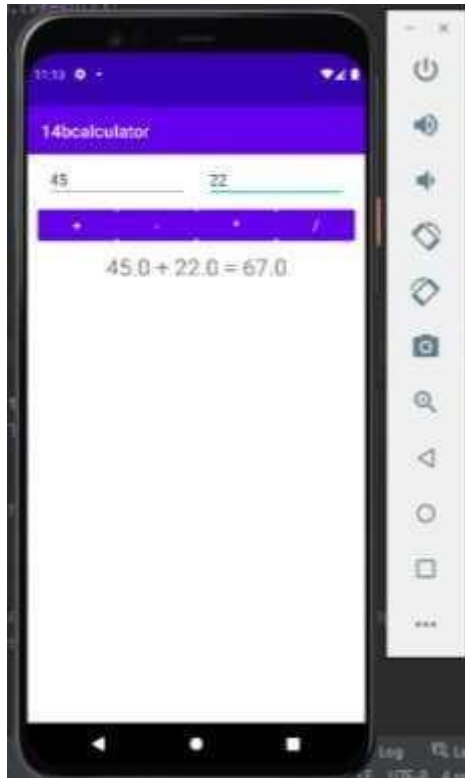
```

```
        android:layout_height="wrap_content"
        android:id="@+id/linearLayout2"
        android:layout_marginTop="3pt"
        android:layout_marginLeft="5pt"
        android:layout_marginRight="5pt">
        <Button
            android:layout_height="wrap_content"
            android:layout_width="match_parent"
            android:layout_weight="1"
            android:text="+" android:textSize="8pt"
            android:id="@+id/btnAdd">
        </Button>
        <Button
            android:layout_height="wrap_content"
            android:layout_width="match_parent"
            android:layout_weight="1" android:text="-"
            android:textSize="8pt"
            android:id="@+id/btnSub">
        </Button>
        <Button
            android:layout_height="wrap_content"
            android:layout_width="match_parent"
            android:layout_weight="1"
            android:text="*" android:textSize="8pt"
            android:id="@+id/btnMult">
        </Button>
        <Button
            android:layout_height="wrap_content"
            android:layout_width="match_parent"
            android:layout_weight="1"
            android:text="/" android:textSize="8pt"
            android:id="@+id/btnDiv">

        </Button>
    </LinearLayout>
    <TextView
        android:layout_height="wrap_content"
        android:layout_width="match_parent"
        android:layout_marginLeft="5pt"
        android:layout_marginRight="5pt"
        android:textSize="12pt"
        android:layout_marginTop="3pt"
        android:id="@+id/tvResult"
        android:gravity="center_horizontal"
        android:layout_weight="0.07">
```



```
</TextView>  
</LinearLayout>
```

OUTPUT:

PROGRAM 2:

Create an application to concatenate two given Strings.
(Consider changing the color of the result string to GREEN*)

Procedure:

Step 1: Start.

Step 2: Create a XML file. Drag and drop 2 EditText to enter 2 strings and 1 Button to concatenate the strings. Then drag and drop a TextView to view the concatenated string.

Step 3: Create a JAVA file to perform concatenation. First initialize the 2 EditText, Button and TextView then create object for each one.

Step 4: Read 2 strings and perform concatenation ('+') operation with those strings.

Step 5: Display the concatenated string on the TextView field. Step 6:

Stop.

MainActivity.java:

```
package com.example.a14bstrconcat;
import android.os.Bundle;
import android.text.TextUtils;
import android.view.View; import
android.widget.Button; import
android.widget.EditText; import
android.widget.TextView;

import androidx.appcompat.app.AppCompatActivity;

public class MainActivity extends AppCompatActivity implements View.OnClickListener{

    EditText etNum1;
    EditText etNum2;

    Button btnconcat;

    TextView tvResult;

    String oper = "";

    /** Called when the activity is first created. */
```

@Override

```
public void onCreate(Bundle savedInstanceState) {
    super.onCreate(savedInstanceState);
    setContentView(R.layout.activity_main);

    // find the elements
    etNum11= (EditText) findViewById(R.id.etNum11);
    etNum22= (EditText) findViewById(R.id.etNum22);

    btnconcat = (Button) findViewById(R.id.btnconcat);tvResult =
    (TextView) findViewById(R.id.tvResult);

    // set a listener
    btnconcat.setOnClickListener(this);
}
```

@Override

```
public void onClick(View v) {
    // TODO Auto-generated method stub
    String S1 = "";
    String S2 = "";

    // check if the fields are empty
    if (TextUtils.isEmpty(etNum11.getText().toString())
        || TextUtils.isEmpty(etNum22.getText().toString())) {
        return;
    }
    // read EditText and fill variables with numbersS1 =
    etNum11.getText().toString();
    S2 = etNum22.getText().toString();

    // form the output line
    tvResult.setText(S1 + " " + S2);
}
}
```

Activity main.xml:

```
<?xml version="1.0" encoding="utf-8"?>
<LinearLayout
    xmlns:android="http://schemas.android.com/apk/res/android"
    android:orientation="vertical" android:layout_width="fill_parent"
    android:layout_height="fill_parent"
    android:weightSum="1">
```

```
<LinearLayout
    android:id="@+id/linearLayout11"
    android:layout_width="wrap_content"
    android:layout_height="159dp"
    android:layout_marginLeft="10pt"
    android:layout_marginTop="3pt"
    android:layout_marginRight="10pt"

    android:orientation="horizontal">

    <EditText android:id="@+id/etNum11"

        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:ems="10"
        android:inputType="textPersonName"
        android:text="Name" />

    <EditText android:id="@+id/etNum22"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:ems="10"
        android:inputType="textPersonName"
        android:text="Name" />

</LinearLayout>

<LinearLayout
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:id="@+id/linearLayout2"
    android:layout_marginTop="3pt"
    android:layout_marginLeft="5pt"
    android:layout_marginRight="5pt">
    <Button
        android:layout_height="wrap_content"
        android:layout_width="match_parent"
        android:layout_weight="1"
        android:text="+" android:textSize="8pt"
        android:id="@+id/btnconcat">
    </Button>
</LinearLayout>
<TextView
    android:layout_height="wrap_content"
    android:layout_width="match_parent"
    android:layout_marginLeft="5pt"
    android:layout_marginRight="5pt"
```

```
        android:textSize="12pt"  
        android:layout_marginTop="3pt"  
        android:id="@+id/tvResult"  
        android:gravity="center_horizontal"  
  
        android:layout_weight="0.07">  
</TextView>  
</LinearLayout>
```

OUTPUT:

PROGRAM 3:

Create an android application to find the factorial of a given number.

Procedure:

Step 1: Start

Step 2: Create a XML file. Drag and drop a EditText to enter the number, a Button to findfactorial and a TextView to display the result.

Step 3: Create a JAVA file to find factorial. First initialize the the EditText, Button andTextView then create object for each one.

Step 4: Read the number and perform necessary operations to find factorial.

Step 5: Display the result on the TextView field.

Step 6: Stop.

MainActivity.java

```
package com.example.factorial;
import androidx.appcompat.app.AppCompatActivity;import android.view.View;import
android.widget.Button;
import android.widget.EditText;import android.widget.TextView;import
android.os.Bundle;

public class MainActivity extends AppCompatActivity implementsView.OnClickListener {EditText

    etNum1; Button btnAdd; TextView tvResult;String oper = "";
    /**
     * Called when the activity is first created.
     */
    @Override
    public void onCreate(Bundle savedInstanceState) { super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_main);

        // find the elements

        etNum1 = (EditText) findViewById(R.id.etNum1);

        btnAdd = (Button) findViewById(R.id.btnAdd); tvResult = (TextView)
        findViewById(R.id.tvResult);
```

```

// set a listener

    btnAdd.setOnClickListener(this);

}

@Override
public void onClick(View v) {
// TODO Auto-generated method stub

    float num1 = 0; float fact = 1; float result = 0;

// check if the fields are empty

    num1 = Float.parseFloat(etNum1.getText().toString());

// read EditText and fill variables with numbers

// 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.btnAdd:
        oper = "+";
        for (int i = 1; i <= num1; i++) {fact = fact * i;
        }

        result = fact; break;
    default:
        break;
    }

// form the output line

    tvResult.setText("Factorial of" + " " + num1 + " = " + result);
}
}

```

Activity main.xml

```

<?xml version="1.0" encoding="utf-8"?>
<LinearLayout xmlns:android="http://schemas.android.com/apk/res/android"
    android:layout_width="fill_parent" android:layout_height="fill_parent"
    android:orientation="vertical" android:weightSum="1">

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

```

```
        android:layout_height="wrap_content" android:layout_marginLeft="10pt"
        android:layout_marginTop="3pt" android:layout_marginRight="10pt">

        <EditText
            android:id="@+id/etNum1" android:layout_width="wrap_content"
            android:layout_height="wrap_content" android:layout_marginRight="5pt"
            android:layout_weight="1" android:inputType="numberDecimal"></EditText>
    </LinearLayout>

    <LinearLayout
        android:id="@+id/linearLayout2" android:layout_width="match_parent"
        android:layout_height="wrap_content" android:layout_marginLeft="5pt" android:layout_marginTop="3pt"
        android:layout_marginRight="5pt">
        <Button
            android:id="@+id/btnAdd" android:layout_width="match_parent" android:layout_height="wrap_content"
            android:layout_weight="1" android:text="RESULT" android:textSize="8pt"></Button>
    </LinearLayout>
    <TextView
        android:id="@+id/tvResult" android:layout_width="match_parent" android:layout_height="wrap_content"
        android:layout_marginLeft="5pt" android:layout_marginTop="3pt" android:layout_marginRight="5pt"
        android:layout_weight="0.07" android:gravity="center_horizontal" android:textSize="12pt"></TextView>
</LinearLayout>
```


OUTPUT :

PROGRAM 4:

Develop a canvas to draw different shapes and to fill the shapes with different colors.

Procedure:

Step 1: Start.

Step 2: Create 2 JAVA files. CustomView.java for create the shape and set colour it using paint and MainActivity.java for display the shape using setContentView.

Step 3: Enter the required measures for the shape and create it then set colour for the shape.

Step 4: Display the shape using setContentView in MainActivity.java file. Step 5:

Stop

MainActivity.java

```
package com.example.shape;
```

```
import androidx.appcompat.app.AppCompatActivity; import android.os.Bundle; public
class MainActivity extends AppCompatActivity { @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(new com.example.shape.CustomView(this));
    }
}
```

CustomView.java

```
package com.example.shape;
```

```
import android.content.Context; import android.graphics.Canvas; import
android.graphics.Color; import android.graphics.Paint; import android.graphics.Rect; import
android.view.View;
public class CustomView extends View { private Rect rectangle;
    private Paint paint, p1;

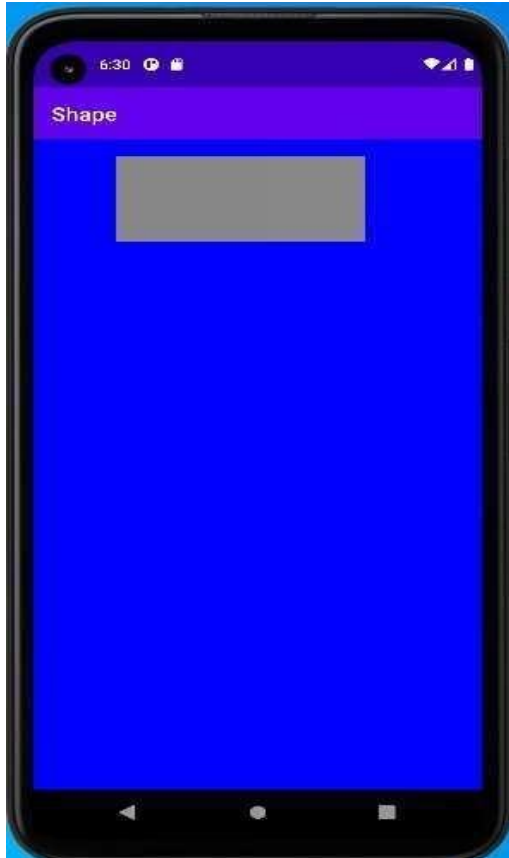
    public CustomView(Context context) { super(context); int x
        = 200; int y = 50;
        int width = 800; int height = 300;
```

```
// create a rectangle that we'll draw later
rectangle = new Rect(x, y, width, height);

// create the Paint and set its color paint = new Paint();
paint.setColor(Color.GRAY);

p1 = new Paint(); p1.setColor(Color.RED);
}

@Override
protected void onDraw(Canvas canvas) {canvas.drawColor(Color.BLUE);
    canvas.drawRect(rectangle, paint);
}
}
```

OUTPUT :

PROGRAM 5:

Create an application to show happy face smiley and sad face smiley to demonstrate buttonclick Events.

Procedure:

Step 1: Start

Step 2: Create two activity with buttons.

Step 3: Create a main activity JAVA file which direct the page to another page on the click of the button from the activity main.

Step 4: Create another activity java smily with to navigate with the onclick listener to mainactivity page.

Step 5: Create face view class two draw the smily with dimension for happy face with canva drawColor,draw circle,Oval,drawArch.

Step 6: Create another face view to draw the smily with dimensions for sad face with canva drawColor,draw circle,Oval,drawArch.

Step 7: Stop.

MainActivity.java

package com.example.a5happyface;

import androidx.appcompat.app.AppCompatActivity;import android.content.Intent; import android.os.Bundle; import android.view.View; import android.widget.Button;

public class MainActivity extends AppCompatActivity {Button button;

@Override

```
protected void onCreate(Bundle savedInstanceState) {  
    super.onCreate(savedInstanceState); setContentView(R.layout.activity_main);button  
        = (Button) findViewById(R.id.button); button.setOnClickListener(new  
        View.OnClickListener() {  
            @Override  
            public void onClick(View v) {openNewActivity();  
            }  
        });  
}
```

```
public void openNewActivity() {  
    Intent intent = new Intent(this, MainActivity2.class);startActivity(intent);  
}
```

```
}
```

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="wrap_content"
    android:layout_height="wrap_content" android:orientation="vertical"
    tools:context=".MainActivity">
    <com.example.a5happyface.FaceView android:layout_width="wrap_content"
        android:layout_height="wrap_content" />
    <Button
        android:id="@+id/button"    android:layout_width="match_parent"

        android:layout_height="wrap_content"    android:text="--->  Sad
Face" />
</RelativeLayout>

```

MainActivity2.java

```
package com.example.a5happyface;
```

```

import android.content.Intent;import android.os.Bundle; import android.view.View; import
android.widget.Button;
import androidx.appcompat.app.AppCompatActivity; public class MainActivity2 extends
AppCompatActivity {
    Button button1;@Override
    protected void onCreate(Bundle savedInstanceState)
    { super.onCreate(savedInstanceState); setContentView(R.layout.activity_main2);
        button1
            = (Button) findViewById(R.id.button1); button1.setOnClickListener(new
            View.OnClickListener() {
                @Override
                public void onClick(View v) {openNewActivity();
                }
            });
    }
    public void openNewActivity(){
        Intent intent1 = new Intent(this,MainActivity.class);startActivity(intent1);
    }
}

```

Activity_main2.xml

```

<?xml version="1.0" encoding="utf-8"?>
<RelativeLayout xmlns:android="http://schemas.android.com/apk/res/android"
    android:layout_width="match_parent"
    android:layout_height="match_parent">
    <com.example.a5happyface.FaceView2 android:layout_width="wrap_content"
        android:layout_height="wrap_content" />
    <Button
        android:id="@+id/button1" android:layout_width="match_parent"
        android:layout_height="wrap_content" android:text="---> Happy Face" />
</RelativeLayout>

```

FaceView.java

```
package com.example.a5happyface;
```

```
import android.content.Context; import android.graphics.Canvas; import android.graphics.Color;
import android.graphics.Paint; import android.graphics.RectF; import android.util.AttributeSet; import
android.view.View;
```

```
public class FaceView extends View {
    private static final String COLOR_HEX = "WHITE"; private final Paint mPaint; private float xPosition;
    private float yPosition; private float radius; private float strokeWidth
    = 20; private float defaultScale = 0.90f; private float
    eyeRadius = 60; private float
    eyeYPosition; private float leftEyeXPosition; private float rightEyeXPosition; public
    FaceView(Context context, AttributeSet attrs) { super(context, attrs); mPaint =
        new Paint(); mPaint.setAntiAlias(true);
    }

```

@Override

```
protected void onDraw(Canvas canvas) { super.onDraw(canvas);
    mPaint.setColor(Color.parseColor(COLOR_HEX));
    mPaint.setStrokeWidth(strokeWidth); mPaint.setStyle(Paint.Style.STROKE);
    canvas.drawPaint(mPaint); canvas.drawColor(Color.BLACK);
    // drawing outer circle

```

```
    // lets setup x cord, y cord, radius

```

```
    // x, y position should point to center. // radius should be half the width

```

```
    / height

```

```

        xPosition = getMeasuredWidth() / 2; yPosition = getMeasuredHeight() / 2; radius =
        xPosition < yPosition ? xPosition : yPosition; radius *= defaultScale;
        canvas.drawCircle(xPosition, yPosition, radius, mPaint);
        // Drawing Eyes.

```

```
    // lets find eye y position

```

```

        eyeYPosition = (float) (yPosition / 1.2);
        // lets find eye x position

        leftEyeXPosition = xPosition < yPosition ? xPosition / 2 : (float)(xPosition / 1.3);

        // lets find right eye x position

        rightEyeXPosition = xPosition < yPosition ? xPosition + xPosition /
2 :
        xPosition + xPosition / 4;
        // left eye

        canvas.drawCircle(leftEyeXPosition, eyeYPosition, eyeRadius,
mPaint);
        // right eye

        canvas.drawCircle(rightEyeXPosition, eyeYPosition, eyeRadius, mPaint);
        // lets draw mouth.

        RectF oval = new RectF(leftEyeXPosition, yPosition + yPosition /
12, rightEyeXPosition, (float) (yPosition + yPosition / 2.5));
        // left top rightbottom

        canvas.drawArc(oval, 10, 150, false, mPaint); // happy
face.

```

FaceView2.java

```
package com.example.a5happyface;
```

```

import android.content.Context; import android.graphics.Canvas; import android.graphics.Color;
import android.graphics.Paint; import android.graphics.RectF; import android.util.AttributeSet; import
android.view.View;
public class FaceView2 extends View {
    private static final String COLOR_HEX = "WHITE"; private final Paint mPaint; private float xPosition;
    private float yPosition; private float radius; private float strokeWidth
= 20; private float defaultScale = 0.90f; private float
    eyeRadius = 60; private float
        eyeYPosition; private float leftEyeXPosition; private float rightEyeXPosition; public
    FaceView2(Context context, AttributeSet attrs)
    { super(context, attrs); mPaint = new Paint(); mPaint.setAntiAlias(true);
    }
    @Override
    protected void onDraw(Canvas canvas) { super.onDraw(canvas);
        mPaint.setColor(Color.parseColor(COLOR_HEX));
        mPaint.setStrokeWidth(strokeWidth); mPaint.setStyle(Paint.Style.STROKE);
        canvas.drawPaint(mPaint); canvas.drawColor(Color.BLACK);
    }
}

```

```
// drawing outer circle

// lets setup x cord, y cord, radius

// x, y position should point to center.

// radius should be half the width / height xPositon = getMeasuredWidth() / 2; yPositon =
getMeasuredHeight() / 2;

radius = xPositon < yPositon ? xPositon : yPositon ;radius *= defaultScale;
canvas.drawCircle(xPositon, yPositon, radius, mPaint);
// Drawing Eyes.

// lets find eye y position

eyeYPositon = (float) (yPositon / 1.2);
// lets find eye x position

leftEyeXPositon = xPositon < yPositon ? xPositon / 2 : (float)(xPositon / 1.3);
// lets find right eye x position

rightEyeXPositon = xPositon < yPositon ? xPositon + xPositio

/ 2 :

xPositon + xPositon / 4;
// left eye

canvas.drawCircle(leftEyeXPositon, eyeYPositon, eyeRadius,

mPaint);
// right eye

canvas.drawCircle(rightEyeXPositon, eyeYPositon, eyeRadius, mPaint);
// lets draw mouth.

RectF oval = new RectF(leftEyeXPositon, yPositon + yPositon /

rightEyeXPositon, (float) (yPositon + yPositon / 2)); //left top right bottom

canvas.drawArc(oval, 200, 140, false, mPaint); // sad face.

}

}
```


OUTPUT:

PROGRAM 6:

Create an application to demonstrate the use of Intents to communicate between different activity.

Procedures:**Implicit intent**

Step1:create Xml file and Java file.

Step2:Open activity_main.xml file and add editText to input

text and button to open web page in a constraint layout.Also add

IDs for each component.

Step3:Open MainActivity.java file and instantiate the button

created in the xml file using findViewById() method.This method binds

the created object to the UI components with the help of assigned ID.

Step4:To display toast message,first add listener on button and this button will open

webpage.

Step5:Create string type variable to store the value of EditText.Value is accepted and

converted to string.

Step7:Create an intent object MainActivity.java class to of the webpage.

Step8:The start activity() method starts to call a webpage for opening specified by intent.

Explicit intent

Step1:create xml file and java file.

Step2:Open activity_main.xml and add a button for moving to second activity and a TextView for

viewing some text.Also add IDs for each components.

Step3:Open MainActivity.java file and instantiate the button, textview created in the xml file using

findViewById.This

method binds the created object to the UI components with the assigned id.

Step4: To create explicit intent, first add the listener on button and using this button you will

move to other activity. Now create an intent and start the targeted activity.

Step5: Now we have to create a second activity as a destination activity.

Step6: open second xml file. Add button and textview to moving back to home activity and to write

some text on activity. Assign id to button and textview.

Step7: open second activity java file. first add the listener on button and using this button move to

home activity. create an intent and start the targeted activity.

MainActivity.java

```
package com.example.a6intents;
import androidx.appcompat.app.AppCompatActivity; import android.content.Intent; import
android.net.Uri; import android.os.Bundle; import android.view.View; import
android.widget.Button;
```

```
public class MainActivity extends AppCompatActivity {Button button;
@Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState); setContentView(R.layout.activity_main);
        button=findViewById(R.id.button);
        //button.setOnClickListener(this);
    }
    public void show(View view){
        Intent intent = new Intent(Intent.ACTION_VIEW);
        intent.setData(Uri.parse("https://www.fisat.ac.in")); startActivity(intent);
    }

    public void callSecondActivity(View view){
        Intent i=new Intent(getApplicationContext(),MainActivity2.class);startActivity(i);
    }
}
```

Activity_main.xml

```
<?xml version="1.0" encoding="utf-8"?>
<androidx.constraintlayout.widget.ConstraintLayout

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

```

xmlns:tools="http://schemas.android.com/tools"
xmlns:app="http://schemas.android.com/apk/res-auto"
android:layout_width="match_parent" android:layout_height="match_parent"
tools:context=".MainActivity"> TextView android:layout_width="wrap_content"
android:layout_height="wrap_content" android:layout_marginEnd="8dp"
android:layout_marginStart="8dp" android:layout_marginTop="8dp"
android:text="First Activity"

```

```

app:layout_constraintBottom_toBottomOf="parent" app:layout_constraintEnd_toEndOf="parent"
app:layout_constraintHorizontal_bias="0.454" app:layout_constraintLeft_toLeftOf="parent"
app:layout_constraintRight_toRightOf="parent" app:layout_constraintStart_toStartOf="parent"
app:layout_constraintTop_toTopOf="parent" app:layout_constraintVertical_bias="0.06" />

```

```
<Button
```

```

    android:id="@+id/button" android:layout_width="wrap_content"
    android:layout_height="wrap_content" android:layout_marginEnd="8dp"
    android:layout_marginStart="8dp" android:layout_marginTop="392dp"
    android:onClick="callSecondActivity" android:text="Call second activity"
    app:layout_constraintEnd_toEndOf="parent" app:layout_constraintStart_toStartOf="parent"
    app:layout_constraintTop_toTopOf="parent" />

```

```

<Button    android:id="@+id/button3" android:layout_width="wrap_content"
    android:layout_height="wrap_content" android:onClick="show" android:text="implicit
    intent" tools:layout_editor_absoluteX="135dp" tools:layout_editor_absoluteY="204dp" />
</androidx.constraintlayout.widget.ConstraintLayout>

```

MainActivity2.java

```
package com.example.a6intents;
```

```

import androidx.appcompat.app.AppCompatActivity; import android.content.Intent; import
    android.os.Bundle; import android.view.View; import android.widget.Button;
public class MainActivity2 extends AppCompatActivity {Button button;
    @Override
    protected void onCreate(Bundle savedInstanceState)
    { super.onCreate(savedInstanceState); setContentView(R.layout.activity_main2);
        Bundle
            extras = getIntent().getExtras(); button=findViewById(R.id.button);
    }
    public void callFirstActivity(View view){
        Intent i=new Intent(getApplicationContext(),MainActivity.class);startActivity(i);
    }
}

```

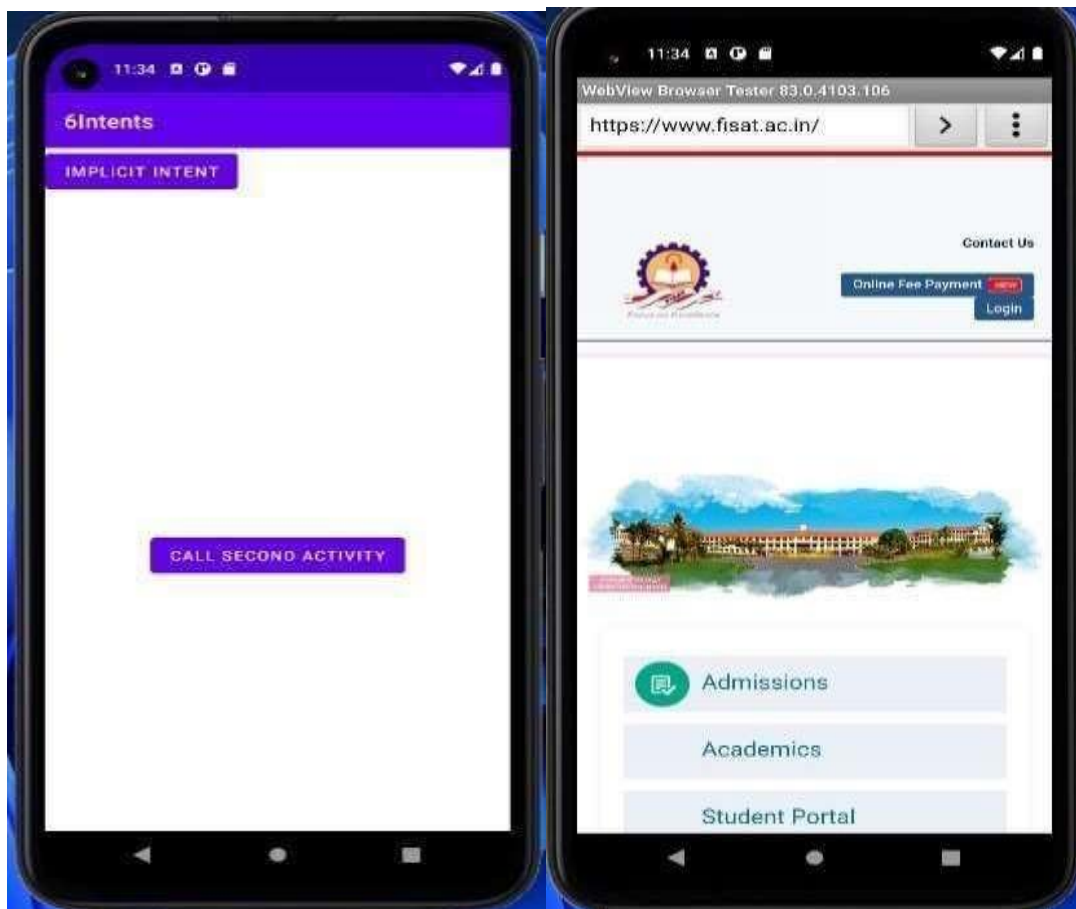
Activity_main2.xml

```
package com.example.a6intents;
```

```
import androidx.appcompat.app.AppCompatActivity;
```

```
import android.content.Intent;import android.os.Bundle; import android.view.View;import  
android.widget.Button;
```

```
public class MainActivity2 extends AppCompatActivity {Button button;  
    @Override  
    protected void onCreate(Bundle savedInstanceState)  
    { super.onCreate(savedInstanceState); setContentView(R.layout.activity_main2);  
      Bundle  
        extras = getIntent().getExtras();button=findViewById(R.id.button);  
    }  
    public void callFirstActivity(View view){  
        Intent i=new Intent(getApplicationContext(),MainActivity.class);startActivity(i);  
    }  
}
```

OUTPUT:



PROGRAM 7:

Create an android application to demonstrate storing data into internal phone memory.

Procedures:

Step1:create Xml file and Java file.

Step2:Open activity_main.xml file and add editText to input text and button to open webpage in a constraint layout.

Step3:Open MainActivity.java file and instantiate the button and editText created in the xmlfile using findViewById() method.This method binds the created object to the UI components with the help of assigned ID.

Step4:To display the information null file should be created using FILEOUTPUTSTREAM.

Step5:Create string type variable to store the value of EditText.Value is accepted and converted to string.

Step7:Create an intent object MainActivity.java class to open the webpage.

Step8:The start activity() method starts to call a webpage for opening specified by intent.

INTENT

Step1:create java file.

Step2:Open activity_main.xml and by using findViewById get the values passed from the first MainActivity.java file.

Step3:To create intent,first add the listener on button and using this button you will move to other activity.Now create an intent and start the targeted activity.

Step4:Using fileInputStream the intended file will display the content passed by the MainActivity.java file.

MainActivity.java

```
package com.example.a7storingdata;
```

```
import androidx.appcompat.app.AppCompatActivity;import android.os.Bundle;
import android.content.Context;import android.content.Intent; import android.view.View;import
android.widget.EditText;import android.widget.Toast; import java.io.File;
import java.io.FileOutputStream;import java.io.IOException;
```

```

public class MainActivity extends AppCompatActivity {EditText editname,editpass;
    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState); setContentView(R.layout.activity_main);
        editname = (EditText) findViewById(R.id.editName);editpass= (EditText)
        findViewById(R.id.editPass);
    }

    public void save(View view) //SAVE
    {
        File file= null;
        String name = editname.getText().toString(); String password =editpass.getText().toString();

        FileOutputStream fileOutputStream = null;try { name =
            name + " "; file = getFilesDir(); fileOutputStream =
            openFileOutput("Code.txt",
Context.MODE_PRIVATE); //MODE PRIVATE

            fileOutputStream.write(name.getBytes()); fileOutputStream.write(password.getBytes());
            Toast.makeText(this, "Saved
            \n" + "Path --" + file +
            "\tCode.txt", Toast.LENGTH_SHORT).show();editname.setText(""); editpass.setText("");return;
        } catch (Exception ex) { ex.printStackTrace();
        } finally {
            try { fileOutputStream.close();

        } catch (IOException e) {e.printStackTrace();
        }
    }

    public void next( View view) //NEXT
    {
        Toast.makeText(this,"NEXT", Toast.LENGTH_SHORT).show();Intent intent= new
        Intent(this, MainActivity2.class); startActivity(intent);

    }
}

```

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:id="@+id/activity_main"
    android:layout_width="match_parent" android:layout_height="match_parent"
    tools:context="com.example.a7storingdata.MainActivity">

```


<TextView

```
    android:text="@string/name" android:layout_width="wrap_content"
    android:layout_height="wrap_content" android:layout_alignParentTop="true"
    android:layout_alignParentLeft="true" android:layout_alignParentStart="true"
    android:layout_marginLeft="51dp" android:layout_marginStart="51dp"
    android:layout_marginTop="59dp" android:id="@+id/txtname"
    android:textStyle="bold|italic" android:textSize="18sp" />
```

<TextView

```
    android:text="@string/password" android:layout_width="wrap_content"
    android:layout_height="wrap_content" android:layout_below="@+id/txtname"
    android:layout_alignLeft="@+id/txtname" android:layout_alignStart="@+id/txtname"
    android:layout_marginTop="56dp" android:id="@+id/txtpass"
    android:textStyle="bold|italic" android:textSize="18sp" />
```

<EditText

```
    android:id="@+id/editName" android:layout_width="wrap_content"
    android:layout_height="wrap_content" android:layout_alignParentTop="true"
    android:layout_marginStart="21dp" android:layout_marginLeft="21dp"
    android:layout_marginTop="48dp" android:layout_toEndOf="@+id/txtpass"
```

```
    android:layout_toRightOf="@+id/txtpass" android:ems="8"
    android:inputType="textPersonName" />
```

<EditText

```
    android:id="@+id/editPass" android:layout_width="wrap_content"
    android:layout_height="wrap_content" android:layout_below="@+id/editName"
    android:layout_alignStart="@+id/editName" android:layout_alignLeft="@+id/editName"
    android:layout_marginTop="35dp" android:ems="10" android:inputType="textPassword" />
```

<Button

```
    android:text="@string/save" android:layout_width="wrap_content"
    android:layout_height="wrap_content" android:layout_below="@+id/editPass"
    android:layout_alignLeft="@+id/txtpass" android:layout_alignStart="@+id/txtpass"
    android:layout_marginTop="86dp" android:id="@+id/button" android:onClick="save"/>
    // onClick "save"
```

```

<Button
    android:text="@string/next" android:layout_width="wrap_content"
    android:layout_height="wrap_content" android:layout_alignTop="@+id/button"
    android:layout_alignRight="@+id/editName" android:layout_alignEnd="@+id/editName"
    android:layout_marginRight="25dp" android:layout_marginEnd="25dp"
    android:id="@+id/button2" android:onClick="next"/> // onClick "next"
</RelativeLayout>

```

MainActivity2.java

```
package com.example.a7storingdata;
```

```

import androidx.appcompat.app.AppCompatActivity; import android.os.Bundle; import
android.content.Intent; import android.util.Log; import android.view.View; import
android.widget.TextView; import android.widget.Toast; import java.io.FileInputStream;
public class MainActivity2 extends AppCompatActivity {TextView getname, getpass;
    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);

        setContentView(R.layout.activity_main2); getname =
        (TextView)findViewById(R.id.getname); getpass =
        (TextView)findViewById(R.id.getpass);
    }
    public void load(View view)
    {
        try {
            FileInputStream fileInputStream = openFileInput("Code.txt"); int read = -1;
            StringBuffer buffer = new StringBuffer(); while((read
            =fileInputStream.read())!= -1){ buffer.append((char)read);
            }
            Log.d("Code", buffer.toString());
            String name = buffer.substring(0,buffer.indexOf(" ")); String pass =
            buffer.substring(buffer.indexOf(" ")+1); getname.setText(name);
            getpass.setText(pass);
        } catch (Exception e) { e.printStackTrace();
        }
        Toast.makeText(this,"Loaded", Toast.LENGTH_SHORT).show();
    }
}

```

```
public void back( View view)
{
    Toast.makeText(this,"Back", Toast.LENGTH_SHORT).show();Intent intent= new Intent(this,
    MainActivity.class); startActivity(intent);
}
}
```

Activity main2.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:id="@+id/activity_main2"
    android:layout_width="match_parent" android:layout_height="match_parent"
    tools:context="com.example.a7storingdata.MainActivity2">

    <TextView
        android:text="@string/getname" android:layout_width="wrap_content"
        android:layout_height="wrap_content" android:layout_alignParentTop="true"
        android:layout_alignRight="@+id/button3" android:layout_alignEnd="@+id/button3"
        android:layout_marginRight="11dp" android:layout_marginEnd="11dp"
        android:layout_marginTop="76dp" android:id="@+id/textView3"
        android:textSize="18sp" android:textStyle="bold|italic" />

    <TextView
        android:text="@string/getpassword"
        android:layout_width="wrap_content" android:layout_height="wrap_content"
        android:layout_below="@+id/textView3" android:layout_alignRight="@+id/textView3"
        android:layout_alignEnd="@+id/textView3" android:layout_marginTop="33dp"
        android:id="@+id/textView4" android:textStyle="bold|italic" android:textSize="18sp"
        />

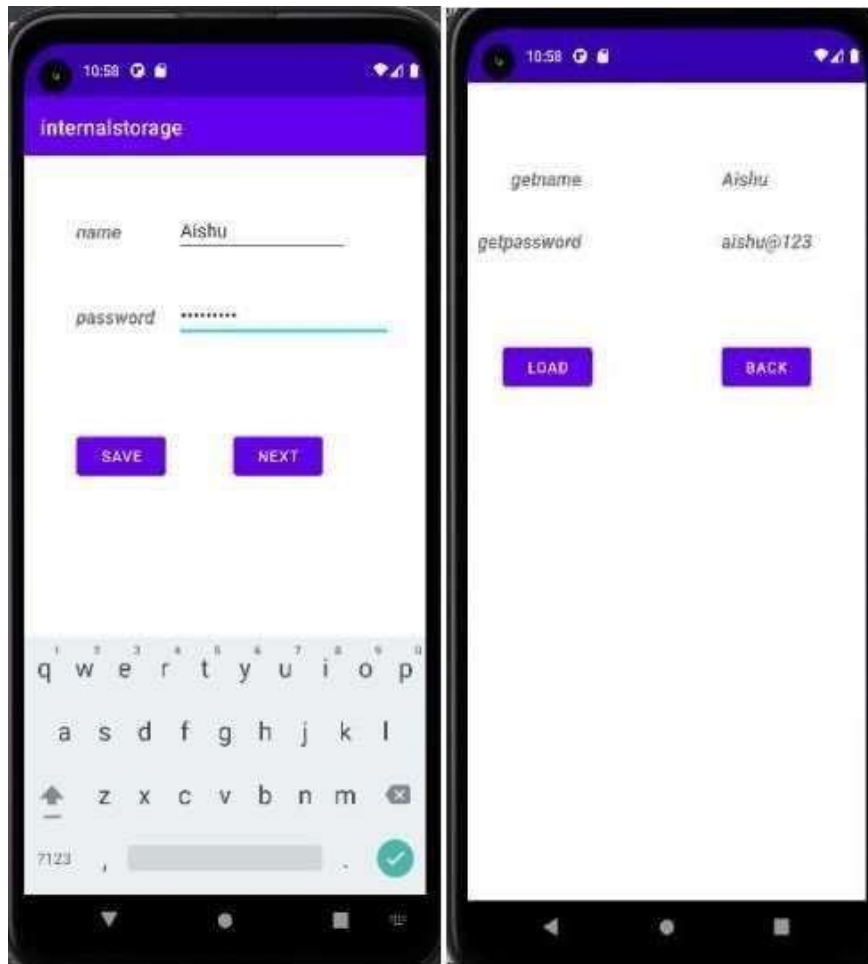
    <TextView
        android:layout_width="wrap_content" android:layout_height="wrap_content"
        android:layout_above="@+id/textView4" android:layout_alignLeft="@+id/button4"
        android:layout_alignStart="@+id/button4" android:id="@+id/getname"
        android:textStyle="bold|italic" android:textSize="18sp" />
```

```
<TextView
    android:layout_width="wrap_content" android:layout_height="wrap_content"
    android:layout_alignBottom="@+id/textView4" android:layout_alignLeft="@+id/getname"
    android:layout_alignStart="@+id/getname" android:id="@+id/getpass"
    android:textStyle="bold|italic" android:textSize="18sp" />
```

```
<Button
    android:text="@string/load" android:layout_width="wrap_content"
    android:layout_height="wrap_content" android:id="@+id/button3"
    android:layout_marginLeft="35dp" android:layout_marginStart="35dp"
    android:onClick="load" android:layout_below="@+id/textView4"
    android:layout_alignParentLeft="true" android:layout_alignParentStart="true"
    android:layout_marginTop="80dp" />
```

```
<Button
    android:text="@string/back" android:layout_width="wrap_content"
    android:layout_height="wrap_content" android:layout_marginRight="54dp"
    android:layout_marginEnd="54dp" android:id="@+id/button4" android:onClick="back"
    android:layout_alignBaseline="@+id/button3" android:layout_alignBottom="@+id/button3"
    android:layout_alignParentRight="true" android:layout_alignParentEnd="true" />
```

```
</RelativeLayout>
```

OUTPUT:

PROGRAM 8:

Create an android application to demonstrate GridView

Step1:create Xml file and Java file.

Step2:Open activity_main.xml file and add GridView Layout.

Step3:Open MainActivity.java file and instantiate the gridview created in the xml file usingfindViewById() method. Then create setAdapter for the gridview.

IMAGEADAPTER

Step1:Create a new Imageadapter.java file. The class ImageAdapter will extend the BaseAdapter.

Step7:The BaseAdapter set Gridview for the images.

Step8:Using R.drawable will assign the imageView.

MainActivity.java

```
package com.example.prgm8;
```

```
import androidx.appcompat.app.AppCompatActivity;import android.os.Bundle;
import android.app.Activity; import android.view.Menu; import android.widget.GridView;
```

```
public class MainActivity extends Activity {@Override protected
    void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState); setContentView(R.layout.activity_main);

        GridView gridview = (GridView) findViewById(R.id.gridview);
        gridview.setAdapter(new ImageAdapter(this));
    }
}
```

Activity main.xml

```
<?xml version="1.0" encoding="utf-8"?>
<GridView xmlns:android="http://schemas.android.com/apk/res/android"android:id="@+id/gridview"
    android:layout_width="fill_parent" android:layout_height="fill_parent"
    android:columnWidth="120dp" android:numColumns="3"
    android:verticalSpacing="10dp" android:horizontalSpacing="10dp"
    android:stretchMode="columnWidth" android:gravity="center"
/>
```

ImageAdapter.java

```
package com.example.prgm8;
```

```
import android.content.Context;import android.view.View; import
android.view.ViewGroup;
import android.widget.BaseAdapter;import android.widget.GridView; import
android.widget.ImageView;
```

```
class ImageAdapter extends BaseAdapter {private Context mContext;public
ImageAdapter(Context c) {mContext = c;
```

```
}
```

```
public int getCount() { return picIds.length;
}
```

```
public Object getItem(int position) {return null;
}
public long getItemId(int position) {return 0;
}
```

```
// create a new ImageView for each item referenced by the Adapter
```

```
public View getView(int position, View convertView, ViewGroup parent) {ImageView
imageView;
```

```
if (convertView == null) {
    imageView = new ImageView(mContext);imageView.setLayoutParams(new
    GridView.LayoutParams(85,85));
```

```
    imageView.setScaleType(ImageView.ScaleType.CENTER_CROP);
```

```
    imageView.setPadding(8,8,8,8);
```

```
}
```

```
else
```

```
{
```

```
    imageView = (ImageView) convertView;
```

```
}
```

```
imageView.setImageResource(picIds[position]); return imageView;
```

```
}
```

```
// Keep all Images in arraypublic Integer[] picIds = {R.drawable.sample2,
R.drawable.sample3, R.drawable.sample4, R.drawable.sample5,
R.drawable.sample6, R.drawable.sample7, R.drawable.sample0,
R.drawable.sample1, R.drawable.sample2, R.drawable.sample3,
R.drawable.sample4, R.drawable.sample5, R.drawable.sample6,
R.drawable.sample7, R.drawable.sample0, R.drawable.sample1,
R.drawable.sample2, R.drawable.sample3, R.drawable.sample4, R.drawable.sample5
```

R.drawable.*sample7*,R.drawable.*sample0*,R.drawable.*sample1*

```
};  
}
```

OUTPUT :



PROGRAM 9:

Demonstrate ImageView and GridView

Procedure:**GridView**

Step 1: Creating a New Project

Step 2: Add google repository in the build.gradle file of the application project.

Step 3: Modify the activity_main.xml file

Step 4: Create an XML layout file for each item of GridView

Step 5: Create a Modal Class for storing Data

Step 6: Create an Adapter Class

Step 7: Modify the MainActivity.java file

Image View

Step 1: Create a New Project

Step 2: Working with the activity_main.xml file

Step 3: Working with the MainActivity file

MainActivity.java

```
package com.example.prgm9;
```

```
import androidx.appcompat.app.AppCompatActivity;
import android.app.Activity;
import android.content.Intent;
import android.os.Bundle;
import android.view.View;
import android.widget.AdapterView;
import android.widget.AdapterView.OnItemClickListener;
import android.widget.GridView;
```

```
public class MainActivity extends Activity {
    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_main);
```

```
        GridView gridView = (GridView) findViewById(R.id.gridview);
        gridView.setAdapter(new ImageAdapter(this));
```

```
        gridView.setOnItemClickListener(new AdapterView.OnItemClickListener() {
            public void onItemClick(AdapterView<?> parent, View v, int position, long id){
```

```
                // Send intent to SingleViewActivity
```

```
                Intent i = new Intent(getApplicationContext(), SingleViewActivity.class);
```

```
                // Pass image index
```

```
                i.putExtra("id", position);
                startActivity(i);
```

```
            }
```

```
        });
```

```
    }
```

```
}
```

Activity main.xml

```
<?xml version="1.0" encoding="utf-8"?>
<GridView xmlns:android="http://schemas.android.com/apk/res/android" android:id="@+id/gridview"
    android:layout_width="fill_parent" android:layout_height="fill_parent"
    android:columnWidth="120dp" android:numColumns="3"
    android:verticalSpacing="10dp" android:horizontalSpacing="10dp"
    android:stretchMode="columnWidth"

    android:gravity="center"
/>
```

ImageAdapter.java

```
package com.example.prgm9;
```

```
import android.content.Context; import android.view.View; import
android.view.ViewGroup;
import android.widget.BaseAdapter; import android.widget.GridView; import
android.widget.ImageView;
```

```
class ImageAdapter extends BaseAdapter {private Context mContext;
```

```
    // Constructor
```

```
    public ImageAdapter(Context c) {mContext = c;
    }
```

```
    public int getCount() { return picIds.length;
    }
```

```
    public Object getItem(int position) {return null;
    }
```

```
    public long getItemId(int position) {

        return 0;
    }
```

```
    // create a new ImageView for each item referenced by the Adapter
```

```
public View getView(int position, View convertView, ViewGroup parent) {ImageView
    imageView;

    if (convertView == null) {
        imageView = new ImageView(mContext);imageView.setLayoutParams(new
            GridView.LayoutParams(85, 85));

        imageView.setScaleType(ImageView.ScaleType.CENTER_CROP); imageView.setPadding(8, 8, 8,
            8);
    } else {
        imageView = (ImageView) convertView;
    }

    imageView.setImageResource(picIds[position]); return imageView;

}

// Keep all Images in arraypublic Integer[] picIds = {R.drawable.sample2,

    R.drawable.sample3,
    R.drawable.sample4,
    R.drawable.sample5,
    R.drawable.sample6,
    R.drawable.sample7,
    R.drawable.sample0,
    R.drawable.sample1,
    R.drawable.sample2,
    R.drawable.sample3,
    R.drawable.sample4,
    R.drawable.sample5,
    R.drawable.sample6,
    R.drawable.sample7,
    R.drawable.sample0,
    R.drawable.sample1,
    R.drawable.sample2,
    R.drawable.sample3,
    R.drawable.sample4,
    R.drawable.sample5,
    R.drawable.sample6,
    R.drawable.sample7,
    R.drawable.sample0, R.drawable.sample1

};
}
```

SingleViewActivity.java

```
package com.example.prgm9;
import androidx.appcompat.app.AppCompatActivity;import android.app.Activity;import
android.content.Intent;import android.os.Bundle;
import android.widget.ImageView;
public class SingleViewActivity extends AppCompatActivity {@Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_single_view);
        // Get intent data

        Intent i = getIntent();

        // Selected image id

        int position = i.getExtras().getInt("id"); ImageAdapter imageAdapter = new ImageAdapter(this);

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

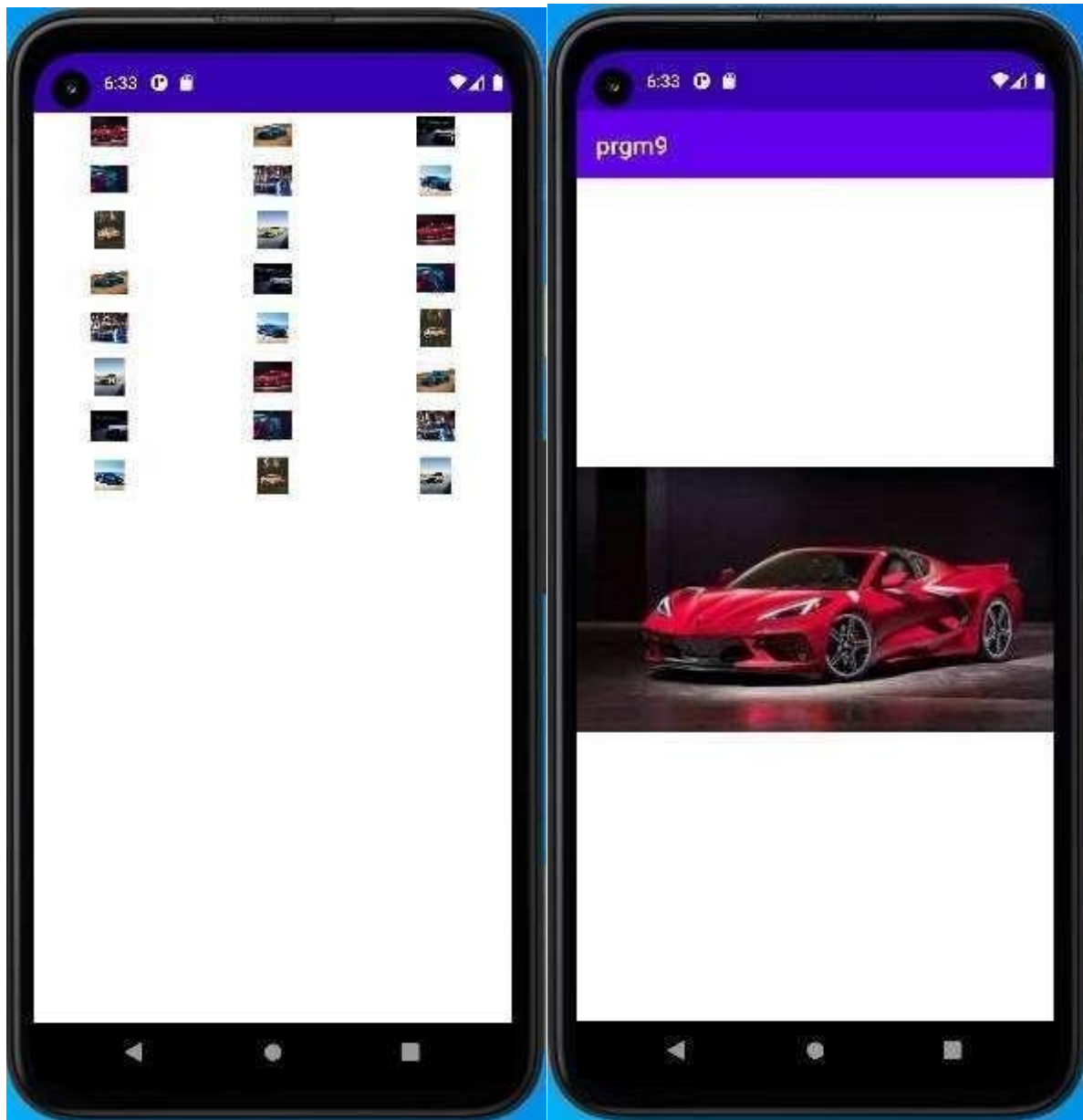
        imageView.setImageResource(imageAdapter.picIds[position]);
    }
}
```

Activity_single_view.xml

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

    <ImageView android:id="@+id/SingleView" android:layout_width="fill_parent"
        android:layout_height="fill_parent"/>

</LinearLayout>
```

OUTPUT:



PROGRAM 10:**Demonstration of Toggle Button****Procedure:**

Step 1: START

Step 2: Create Xml file and Java file.

Step 2: Open activity_main.xml file and one Image View to display image and one button to change images in a frame layout.

Step 3: Download three images and name it piq1.jpg, buttonback.jpg, and pic2.jpg etc and paste it in /src/drawable/ folder.

Step 4: Open MainActivity.java file and import the libraries that are needed.

Step 5: Instantiate the button and Image View created in the xml file using findViewById() method. This method binds the created object to the UI components with the help of assigned ID.

Step 6: By clicking the button with buttonback.jpg, it changes the images between piq1.jpg and pic2.jpg.

Step 7: STOP

MAINACTIVITY.JAVA

```
package com.example.togglebutton;
```

```
import android.app.Activity; import android.view.View; import android.widget.Button; import
android.widget.ImageView; import android.os.Bundle;
```

```
public class MainActivity extends Activity { String s = "Next";
    @Override
```

```
    protected void onCreate(Bundle
        savedInstanceState) {
```

```
    // TODO Auto-generated method stub super.onCreate(savedInstanceState);
    setContentView(R.layout.activity_main); Button next= (Button)
```

```
        findViewById(R.id.next); next.setText(s); next.setOnClickListener(new View.OnClickListener() {
            @Override
```

```
                public void onClick(View v) { if (s.equals("Next")) {
    // TODO Auto-generated method stub
```

```
        ImageView img = (ImageView) findViewById(R.id.imageview);
        img.setImageResource(R.drawable.pic1); Button next= (Button)
            findViewById(R.id.next); s = "Prev";
        next.setText(s);
    } else {
        ImageView img = (ImageView) findViewById(R.id.imageview);

        img.setImageResource(R.drawable.pic2); Button next= (Button)
            findViewById(R.id.next); s = "Next";
        next.setText(s);
```

```
        }  
    });  
    }  
}
```

ACTIVITY MAIN.XML

```
<?xml version="1.0" encoding="utf-8"?>  
<FrameLayout xmlns:android="http://schemas.android.com/apk/res/android"  
    android:layout_width="fill_parent"  
    android:layout_height="fill_parent">  
    <ImageView android:id="@+id/imageview" android:layout_width="fill_parent"  
        android:layout_height="fill_parent" android:scaleType="fitCenter"  
        android:src="@drawable/pic1" />  
    <Button android:id="@+id/next" android:layout_width="wrap_content"  
        android:layout_height="30dp"  
        android:layout_marginBottom="15dp" android:layout_marginRight="10dp"  
        android:layout_gravity="bottom|right" android:paddingTop="2dp" android:paddingBottom="2dp"  
        android:background="@drawable/buttonback" android:textColor="#000000" android:text="Next"  
    />  
</FrameLayout>
```


OUTPUT

PROGRAM 11:

Demonstration of options menu

Procedure:

Step 1: Start

Step 2: Create xml and java file Step 3:

Create optionsmenu.xml file

Step 4: Open optionsmenu.xml file, and add one or more items to your optionsmenu depending on the needs.

Step 5: Open main_activity.java file and import necessary libraries

Step 6: Inflate the menu resources using onCreateOptionsMenu() method.

Step 7: Detect user interaction by add the onOptionsItemSelected method outline after the onCreateOptionsMenu() method.

Step 8: Respond to Menu Item Selection by using switch statement to your method.

Step 9: Stop

MAINACTIVITY.JAVA

```
import androidx.appcompat.app.AppCompatActivity;
import android.os.Bundle;
import android.view.Menu; import
android.view.MenuItem; import
android.widget.TextView; import
android.widget.Toast;
import androidx.appcompat.app.AppCompatActivity;
public class MainActivity extends AppCompatActivity {
// TextView tvMsg;
@Override
protected void onCreate(Bundle savedInstanceState) {

super.onCreate(savedInstanceState);
setContentView(R.layout.activity_main);
// tvMsg= (TextView) findViewById(R.id.textView);
}
// Overriding onCreateOptionsMenu() to make Option menu
@Override
public boolean onCreateOptionsMenu(Menu menu) {
//Inflating menu by overriding inflate() method of
MenuInflater class.
//Inflating here means parsing layout XML to views.
getMenuInflater().inflate(R.menu.menucontext, menu);return
true;
}
//Overriding onOptionsItemSelected to perform event on menu
```

```
items
@Override
public boolean onOptionsItemSelected(MenuItem menuItem) {

    Toast.makeText(this, "The MENU ITEM Selected : " +
        menuItem.getTitle(), Toast.LENGTH_LONG).show(); switch
        (menuItem.getItemId()) {
        case R.id.search:
            //Your code here
            return true;
        case R.id.find:
            //Your code here
            return true;
        case R.id.edit:
            //Your code here
            return true;
        case R.id.relocate:
            //Your code here
            return true;
        case R.id.exit:
            //Your code here
            return true;
        default:
            return super.onOptionsItemSelected(menuItem);
        }
    }
}
```

ACTIVITY MAIN.XML

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

<androidx.constraintlayout.widget.ConstraintLayout
xmlns:android="http://schemas.android.com/apk/res/android"
xmlns:app="http://schemas.android.com/apk/res-auto"
xmlns:tools="http://schemas.android.com/tools"
android:layout_width="match_parent"
android:layout_height="match_parent"
tools:context=".MainActivity">

</androidx.constraintlayout.widget.ConstraintLayout>
```

MENUCONTEXT.XML

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

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

<item

android:id="@+id/search"

android:title="Search" />

<item

android:id="@+id/find"

android:title="Find" />

<item

android:id="@+id/edit"

android:title="Edit" />

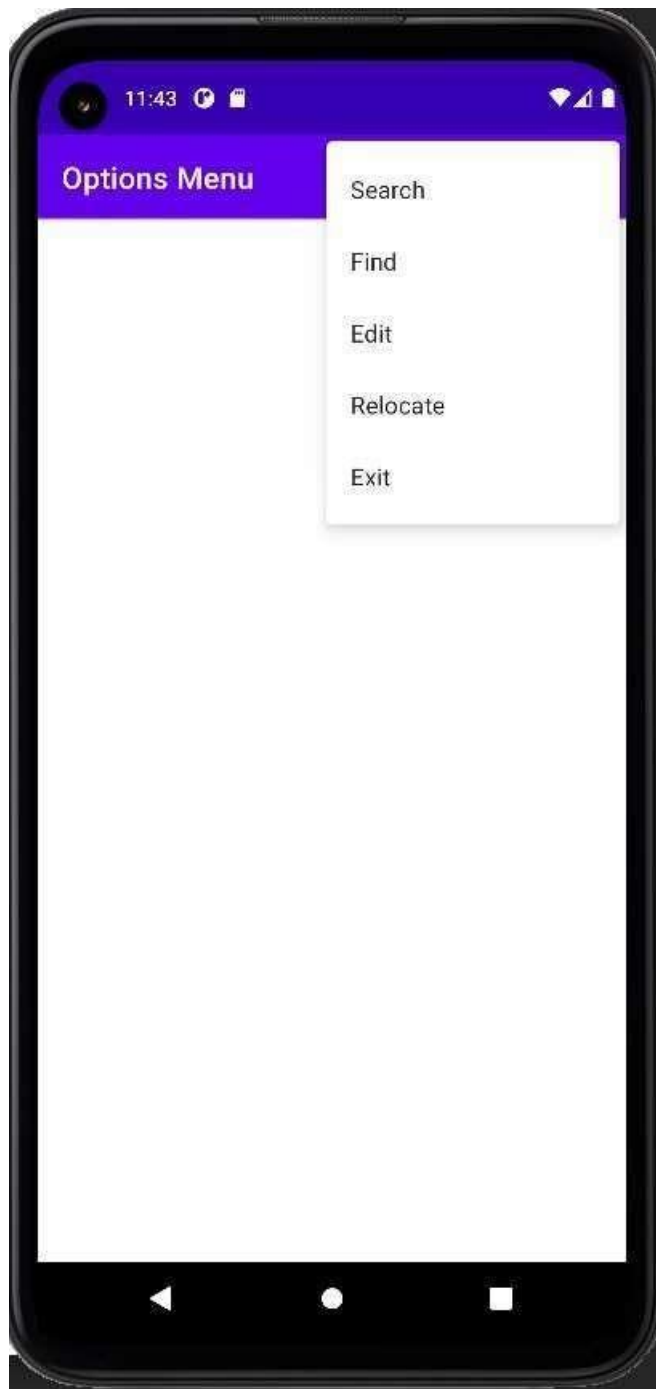
<item android:id="@+id/relocate"

android:title="Relocate" />

<item

android:id="@+id/exit"

android:title="Exit" /> </menu>
```

OUTPUT:

Use of Spinner widget in android application demonstration.

Step 1 : Start

Step 3 : Open activity_main.xml file and add a spinner object inside relativelayout and one textview

Step 5 : Open strings.xml file and add string under resource element with fewitems using string-array

Step 7 : Getting the instance of spinner using `findViewById()` and applying `OnItemSelectedListener` on it

Step 8 : We use array adapter to fill the data in spinner, also we use toast todisplay when the item in spinner is selected.

Step 9 : Performing action OnItemSelected and OnNothingSelected

```
package com.example.a12spinnerwidget;import android.os.Bundle;import
android.view.View;
import android.widget.AdapterView;import android.widget.Spinner; import android.widget.Toast;
```

```
//these are the global variables
```

Spinner `classSpinner, divSpinner;`

```
// string variable to store selected values
```

```
String selectedClass, selectedDiv;
```

@Override

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

```
classSpinner = (Spinner) findViewById(R.id.classSpinner); divSpinner = (Spinner)
findViewById(R.id.divSpinner);
```

```
// Class Spinner implementing onItemSelectedListener
```

```
classSpinner.setOnItemSelectedListener(new AdapterView.OnItemSelectedListener()
```

{

@Override

```
public void onItemSelected(AdapterView<?> parent, View view, int position,
```

```
long id) {

    String selectedClass = parent.getItemAtPosition(position).toString();switch
(selectedClass) {case "Class 1":
    // assigning div item list defined in XMLto the div

    Spinner

        divSpinner.setAdapter(new
            ArrayAdapter<String>(MainActivity.this,

android.R.layout.simple_spinner_dropdown_item,

getResources().getStringArray(R.array.items_div_class_1)));

        break;

        case "Class 2": divSpinner.setAdapter(new
            ArrayAdapter<String>(MainActivity.this,

android.R.layout.simple_spinner_dropdown_item,

getResources().getStringArray(R.array.items_div_class_2)));

        break;

        case "Class 3": divSpinner.setAdapter(new
            ArrayAdapter<String>(MainActivity.this,

android.R.layout.simple_spinner_dropdown_item,
getResources().getStringArray(R.array.items_div_class_3)));

        Toast.makeText(MainActivity.this, "\n Class: \t " +selectedClass,
        Toast.LENGTH_LONG).show();

        break;

        case "Class 4": divSpinner.setAdapter(new
            ArrayAdapter<String>(MainActivity.this,

android.R.layout.simple_spinner_dropdown_item,
getResources().getStringArray(R.array.items_div_class_4)));

        Toast.makeText(MainActivity.this, "\n Class: \t " +selectedClass,
        Toast.LENGTH_LONG).show();

        break;
    }
    //set divSpinner Visibility to Visible

    divSpinner.setVisibility(View.VISIBLE);
}
```

```

        @Override
        public void onNothingSelected(AdapterView<?> parent) {
            // can leave this empty
        }
    };
    // Div Spinner implementing onItemSelectedListener

    divSpinner.setOnItemSelectedListener(new AdapterView.OnItemSelectedListener() { @Override
        public void onItemSelected(AdapterView<?> parent, Viewview, int position, long
            id) {
                selectedDiv =
                    parent.getItemAtPosition(position).toString();
                // create a Toast to show the values on screen

                Toast.makeText(MainActivity.this,

                    "\n Div: \t" + selectedDiv, Toast.LENGTH_LONG).show();

            }

        @Override
        public void onNothingSelected(AdapterView<?> parent) {
            // can leave this empty
        }

    });
}
}

```

activity_main.xml

```

<?xml version="1.0" encoding="utf-8"?>
<RelativeLayout xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:tools="http://schemas.android.com/tools" android:layout_width="match_parent"
    android:layout_height="match_parent"
    tools:context="com.example.a12spinnerwidget.MainActivity">

    <TextView
        android:id="@+id/tvDemo" android:layout_width="match_parent"
        android:layout_height="wrap_content" android:layout_alignParentStart="true"
        android:layout_alignParentTop="true" android:gravity="center" android:text="SPINNER
        DEMO" android:layout_alignParentLeft="true" />

```



```
<Spinner
    android:id="@+id/classSpinner" android:layout_width="match_parent"
    android:layout_height="wrap_content" android:layout_below="@+id/tvDemo"
    android:layout_marginTop="25dp" android:entries="@array/items_class"/>

<Spinner
    android:id="@+id/divSpinner"    android:visibility="gone"

    android:layout_width="match_parent" android:layout_height="wrap_content"
    android:layout_below="@id/classSpinner" android:layout_toLeftOf="@id/classSpinner"
    android:layout_marginTop="10dp"
    />
</RelativeLayout>
```

strings.xml

```
<resources>
<string name="app_name">SpinnerDemo</string>

<string-array name="items_class">
    <item>Class 1</item>
    <item>Class 2</item>
    <item>Class 3</item>
    <item>Class 4</item>
</string-array>

<string-array name="items_div_class_1">
    <item>Div 1-A</item>
    <item>Div 1-B</item>

    <item>Div 1-C</item>
    <item>Div 1-D</item>
</string-array>

<string-array name="items_div_class_2">
    <item>Div 2-A</item>
    <item>Div 2-B</item>

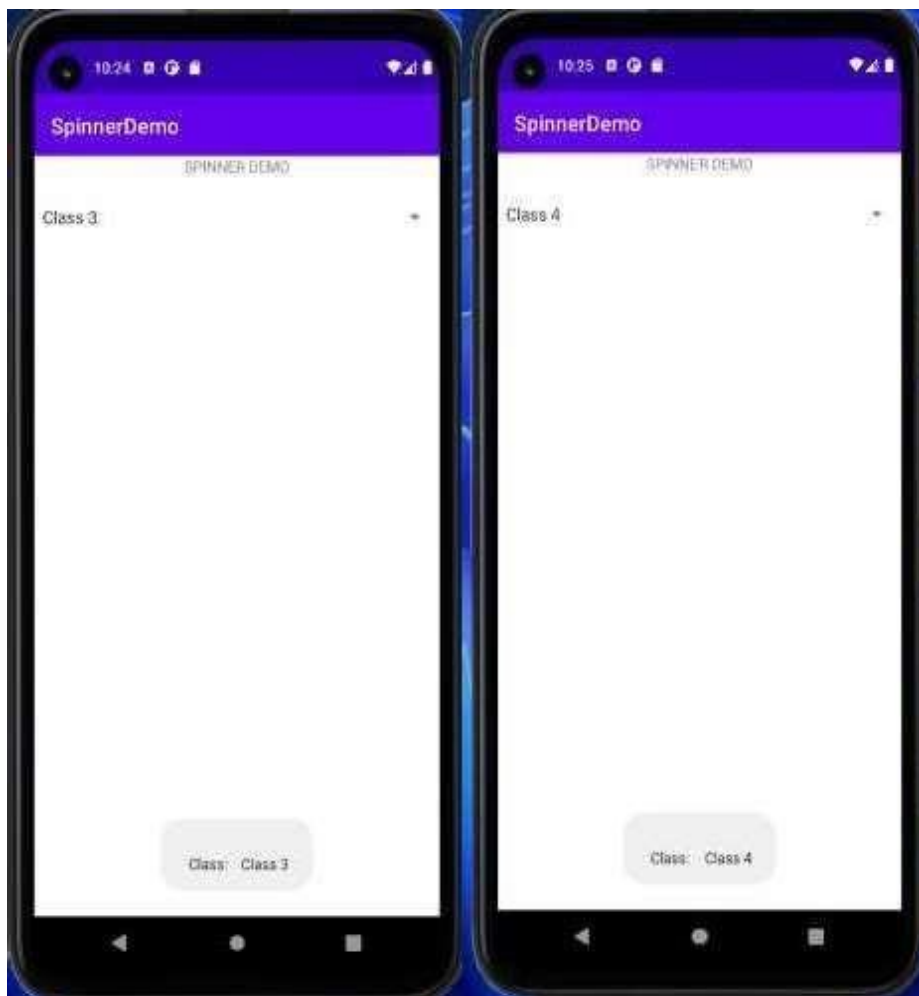
    <item>Div 2-C</item>
    <item>Div 2-D</item>
</string-array>

<string-array name="items_div_class_3">
    <item>Div 3-A</item>
    <item>Div 3-B</item>
    <item>Div 3-C</item>
```

```
<item>Div 3-D</item>
</string-array>

<string-array name="items_div_class_4">
  <item>Div 4-A</item>
  <item>Div 4-B</item>
  <item>Div 4-C</item>
  <item>Div 4-D</item>
</string-array>

</resources>
```

OUTPUT:

PROGRAM 13:

Database application using SQLite.

Procedure:

Step 1: Start

Step 2: Create xml and java files

Step 3: Open activity_main.xml file and add four textview, edittext and addfour buttons to perform add, view, delete and update

Step 4: Open main_activity.java file and import the libraries that are needed
Step 5: Create mydb object for the databasehelper class

Step 6: Instantiate the buttons and edittext created in the xml file using findViewById() method. This method binds the created object to the UIcomponents with the help of assigned ID.

Step 7: Define methods deletedata(), adddata(), updatedata() ,viewall(), which returns delete particular data, insert data, update data, and view all dataoperations respectively

Step 8: OnCreateOptionsMenu() method specify the options menu for the activity. It inflates the menu resource defined in xml into menu provided.

Step 9: By using onOptionsItemSelected() method we can handle action baritems that clicks.

Step 10: Create databasehelper.java file to handle database operations thatare defined using sqliteopenhelper

Step 11: Mention all database informations such as database, table, columns etc.

Step 12: Call methods inorder to handle the database opertions such as creation, upgrading, reading, writing, deleting

Step 13: Stop

ACTIVITY MAIN.XML

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

<RelativeLayout xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:tools="http://schemas.android.com/tools"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    tools:context=".MainActivity">

    <TextView android:layout_width="wrap_content"
        android:layout_height="wrap_content"
```

```
android:textAppearance="?android:attr/textAppearanceLarge" android:text="Name"
android:id="@+id/textView"
android:layout_alignParentTop="true"
android:layout_alignParentLeft="true"
android:layout_alignParentStart="true" />
<TextView android:layout_width="wrap_content"
android:layout_height="wrap_content"
android:textAppearance="?android:attr/textAppearanceLarge"
android:text="Surname"
android:id="@+id/textView2"
android:layout_below="@+id/editText_name"
android:layout_alignParentLeft="true"
android:layout_alignParentStart="true" />
<TextView android:layout_width="wrap_content"
android:layout_height="wrap_content"
android:textAppearance="?android:attr/textAppearanceLarge"

android:text="Marks" android:id="@+id/textView3"
android:layout_below="@+id/editText_surname"
android:layout_alignParentLeft="true"
android:layout_alignParentStart="true" />
<EditText
android:layout_width="match_parent"
android:layout_height="wrap_content"
android:id="@+id/editText_name"
android:layout_alignTop="@+id/textView"
```

```
android:layout_toRightOf="@+id/textView" android:layout_toEndOf="@+id/textView"
/>
<EditText android:layout_width="match_parent"
android:layout_height="wrap_content"
android:id="@+id/editText_surname"
android:layout_alignTop="@+id/textView2"
android:layout_toRightOf="@+id/textView2"
android:layout_toEndOf="@+id/textView2" />
<EditText android:layout_width="match_parent"
android:layout_height="wrap_content"
android:id="@+id/editText_Marks"
android:layout_below="@+id/editText_surname"
android:layout_toRightOf="@+id/textView3"
android:layout_toEndOf="@+id/textView3" />
<Button
android:layout_width="wrap_content"
android:layout_height="wrap_content"

android:text="Add Data"
android:id="@+id/button_add"
android:layout_below="@+id/editText_Marks"
android:layout_alignParentLeft="true"
android:layout_alignParentStart="true"
android:layout_marginTop="76dp" />
<Button
android:layout_width="wrap_content"
android:layout_height="wrap_content"
android:text="View All"
```

```
android:id="@+id/button_viewAll" android:layout_above="@+id/button_update"
android:layout_centerHorizontal="true" />
<Button android:layout_width="wrap_content"
android:layout_height="wrap_content"
android:text="Update"
android:id="@+id/button_update"
android:layout_below="@+id/button_add"
android:layout_alignParentLeft="true"
android:layout_alignParentStart="true" />
<Button android:layout_width="wrap_content"
android:layout_height="wrap_content"
android:text="Delete"
android:id="@+id/button_delete"
android:layout_centerVertical="true"
android:layout_below="@+id/button_viewAll"
android:layout_alignLeft="@+id/button_viewAll"

android:layout_alignStart="@+id/button_viewAll" />
<TextView android:layout_width="wrap_content"
android:layout_height="wrap_content"
android:textAppearance="?android:attr/textAppearanceLarge"
android:text="id"
android:id="@+id/textView_id"
android:layout_below="@+id/editText_Marks"
android:layout_alignParentLeft="true"
android:layout_alignParentStart="true" />
```

```
<EditText android:layout_width="match_parent"
android:layout_height="wrap_content"
android:id="@+id/editText_id"
android:layout_alignTop="@+id/textView_id"
android:layout_toRightOf="@+id/textView3"
android:layout_toEndOf="@+id/textView3" />
</RelativeLayout>
```

MAINACTIVITY.JAVA

```
package com.example.dbtest;

import androidx.appcompat.app.AlertDialog;
import androidx.appcompat.app.AppCompatActivity;
import android.database.Cursor;
import android.os.Bundle; import
android.view.Menu; import
android.view.MenuItem;import
android.view.View; import
android.widget.Button; import
android.widget.EditText;import
android.widget.Toast;

public class MainActivity extends AppCompatActivity {
    DatabaseHelper myDb;
    EditText editName,editSurname,editMarks ,editTextId;
    Button btnAddData;
    Button btnviewAll;
    Button btnDelete; Button
    btnviewUpdate;
```

```
@Override

protected void onCreate(Bundle savedInstanceState) {

    super.onCreate(savedInstanceState);

    setContentView(R.layout.activity_main);

    myDb = new DatabaseHelper(this);

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

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


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

    editTextId = (EditText)findViewById(R.id.editText_id); btnAddData

    = (Button)findViewById(R.id.button_add); btnviewAll =

    (Button)findViewById(R.id.button_viewAll); btnviewUpdate=

    (Button)findViewById(R.id.button_update);btnDelete=

    (Button)findViewById(R.id.button_delete); AddData();

    viewAll();

    UpdateData();

    DeleteData();

}

public void DeleteData() {

    btnDelete.setOnClickListener(new

    View.OnClickListener() { @Override

    public void onClick(View v) {

        Integer deletedRows =

        myDb.deleteData(editTextId.getText().toString());

        if(deletedRows > 0)

            Toast.makeText(MainActivity.this,"Data

            Deleted",Toast.LENGTH_LONG).show();
```



```
else Toast.makeText(MainActivity.this,"Data
not Deleted",Toast.LENGTH_LONG).show();

}

}

);

}

public void UpdateData() {
    btnviewUpdate.setOnClickListener(

new View.OnClickListener() {
    @Override
    public void onClick(View v) {
        boolean isUpdate =
        myDb.updateData(editTextId.getText().toString(),
        editName.getText().toString(),

        editSurname.getText().toString(),editMarks.getText().toString());
        if(isUpdate == true)
        Toast.makeText(MainActivity.this,"Data
        Update",Toast.LENGTH_LONG).show(); else
        Toast.makeText(MainActivity.this,"Data
        not Updated",Toast.LENGTH_LONG).show();
    }
}

);

}

public void AddData() {
    btnAddData.setOnClickListener(
```

```
new View.OnClickListener() {  
    @Override  
    public void onClick(View v) {  
        boolean isInserted =  
        myDb.insertData(editName.getText().toString(),  
        editSurname.getText().toString(),  
        editMarks.getText().toString() );  
  
        if(isInserted == true)  
            Toast.makeText(MainActivity.this,"Data  
            Inserted",Toast.LENGTH_LONG).show();else  
  
            Toast.makeText(MainActivity.this,"Data  
            not Inserted",Toast.LENGTH_LONG).show();  
        }  
    }  
};  
  
public void viewAll() {  
    btnviewAll.setOnClickListener(new  
    View.OnClickListener() { @Override  
    public void onClick(View v) {  
        Cursor res = myDb.getAllData();  
        if(res.getCount() == 0) {  
            // show message showMessage("Error","Nothing  
            found");
```

```
return;
```

```
}
```

```
StringBuffer buffer = new StringBuffer();while
```

```
(res.moveToNext()) { buffer.append("Id :"+
```

```
res.getString(0)+"\n"); buffer.append("Name
```

```
:"+ res.getString(1)+"\n");
```

```
buffer.append("Surname :"+
```

```
res.getString(2)+"\n"); buffer.append("Marks
```

```
:"+ res.getString(3)+"\n\n");
```

```
}
```

```
// Show all data
```

```
showMessage("Data",buffer.toString());
```

```
}
```

```
}
```

```
);
```

```
}
```

```
public void showMessage(String title,String Message){
```

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

```
builder.setCancelable(true);
```

```
builder.setTitle(title);
```

```
builder.setMessage(Message);
```

```
builder.show();
```

```
}
```

```
@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);

}

}
```

DATABASEHELPER.JAVA

```
package com.example.dbtest;
```

```
import android.content.ContentValues;
```

```
import android.content.Context; import
```

```
android.database.Cursor;
```

```
import android.database.sqlite.SQLiteDatabase; import
```

```
android.database.sqlite.SQLiteOpenHelper;
```

```
public class DatabaseHelper extends SQLiteOpenHelper { public
```

```
static final String DATABASE_NAME = "Student.db";public static
```

```
final String TABLE_NAME = "student_table"; public static final
```

```
String COL_1 = "ID";
```

```
public static final String COL_2 = "NAME"; public
```

```
static final String COL_3 = "SURNAME";public static
```

```
final String COL_4 = "MARKS";
```

```
public DatabaseHelper(Context context) {
```

```
super(context, DATABASE_NAME, null, 1);
```

```
}
```

```
@Override
```

```
public void onCreate(SQLiteDatabase db) {
```

```
db.execSQL("create table " + TABLE_NAME + " (ID INTEGER PRIMARY KEY
```

```
AUTOINCREMENT,NAME TEXT,SURNAME TEXT,MARKS INTEGER)");
```

```
}
```

```
@Override
```

```
public void onUpgrade(SQLiteDatabase db, int oldVersion, int
```

```
newVersion) {
```

FEDERAL INSTITUTE OF SCIENCE AND TECHNOLOGY

```
db.execSQL("DROP TABLE IF EXISTS "+TABLE_NAME);
```

```
onCreate(db);
```

```
}
```

```
public boolean insertData(String name,String surname,String marks) {SQLiteDatabase
```

```
db = this.getWritableDatabase();
```

```
ContentValues contentValues = new ContentValues();
```

```
contentValues.put(COL_2,name);
```

```
contentValues.put(COL_3,surname);
```

```
contentValues.put(COL_4,marks);
```

```
long result = db.insert(TABLE_NAME,null ,contentValues);
```

```
if(result == -1)
```

```
return false;
```

```
else
```

```
return true;
```

```
}
```

```
public Cursor getAllData() {
```

```
SQLiteDatabase db = this.getWritableDatabase();
```

```
Cursor res = db.rawQuery("select * from "+TABLE_NAME,null);
```

```
return res;
```

```
}
```

```
public boolean updateData(String id,String name,String surname,Stringmarks) {
```

```
SQLiteDatabase db = this.getWritableDatabase();
```

```
ContentValues contentValues = new ContentValues();
```

```
contentValues.put(COL_1,id);
```

```
contentValues.put(COL_2,name);
```

```
contentValues.put(COL_3,surname);
contentValues.put(COL_4,marks);
db.update(TABLE_NAME, contentValues, "ID = ?",new String[]
{ id });

return true;
}

public Integer deleteData (String id) { SQLiteDatabase
db = this.getWritableDatabase();
return db.delete(TABLE_NAME, "ID = ?",new String[] {id});
}
}
```

OUTPUT: