

D.K.M. COLLEGE FOR WOMEN

(AUTONOMOUS)

VELLORE-1

AFFILIATED TO THIRUVALLUVAR UNIVERSITY



PRACTICAL MANUAL ANDROID PROGRAMMING

**III B.Sc Computer Science
2022-2023 EVEN SEMESTER**

Prepared By

Mrs. R.Divya.,MCA., M.Phil
Assistant Professor,
Department of Computer Science,
D.K.M. College for Women (Autonomous).

Approved By

Mrs. S. Shanthi., M.C.A, M.Phil., SET
Assistant Professor & Head,
Department of Computer Science,
D.K.M. College for Women (Autonomous).

INDEX

<i>S.NO</i>	<i>DATE</i>	<i>PROGRAM NAME</i>	<i>PAGE NO</i>
<i>1</i>		<i>Button, TextView, EditText</i>	
<i>2</i>		<i>RadioButton, RadioGroup</i>	
<i>3</i>		<i>CheckBox</i>	
<i>4</i>		<i>DialogBox</i>	
<i>5</i>		<i>Date Picker</i>	
<i>6</i>		<i>Progress Bar</i>	
<i>7</i>		<i>Spinner</i>	
<i>8</i>		<i>Simple Menu</i>	
<i>9</i>		<i>Menu with Intents</i>	
<i>10</i>		<i>Files</i>	

Ex.No.1**BUTTON,TEXTVIEW,EDITTEXT****AIM:**

To Develop a android Application using Button,Text view,Edit text.

PROCEDURE:

Step 1 : Start the process.

Step 2: Create a new Project by opening the Android Studio and then click on File --> new -->new project.

Step 3: Select the basic activity and choose next option.

Step 4: Change the Application name and press finish button.

Step 5: Next Going to Design layout by selecting content - main.xml

Step 6: Place the appropriate Controls for Developing the Application

Step 7: Change the constraints and required properties for the controls

S.No.	Control	Control values	Properties & Values
1	Text	Text View (ENTER THE FIRST NUMBER)	ID (Text View)
2	Text	Text View (ENTER THE SECOND VALUE)	ID (TextView2)
3	Text	Plaintext ()	ID (Text1)
4	Text	Plaintext ()	ID (Text2)
5	Text	Plaintext ()	ID (tans)
6	Button	Text (ADD)	ID (add btn)
7	Button	Text (SUB)	ID (sub btn)
8	Button	Text (MUL)	ID (mul btn)
9	Button	Text (DIV)	ID (div btn)

Step 8: Enter Java coding for Application by selecting MainActivity.java

Step 9: Stop the process

PROGRAM:

Content 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"
    app:layout_behavior="@string/appbar_scrolling_view_behavior"
    tools:context=".MainActivity"
    tools:showIn="@layout/activity_main">
```

```
<EditText
    android:id="@+id/text1"
    android:layout_width="87dp"
    android:layout_height="50dp"
    android:layout_marginStart="212dp"
    android:layout_marginLeft="212dp"
    android:ems="10"
    android:inputType="textPersonName"
    android:text=""
    app:layout_constraintBottom_toBottomOf="parent"
    app:layout_constraintEnd_toEndOf="parent"
    app:layout_constraintEnd_toStartOf="@+id/editText2"
    app:layout_constraintHorizontal_bias="0.5"
    app:layout_constraintStart_toEndOf="@+id/textView2"
    app:layout_constraintStart_toStartOf="parent"
    app:layout_constraintTop_toTopOf="parent"
    app:layout_constraintVertical_bias="0.154" />
```

```
<Button
    android:id="@+id/addbtn"
    android:layout_width="90dp"
    android:layout_height="40dp"
    android:text="Add"
    app:layout_constraintBottom_toBottomOf="parent"
    app:layout_constraintEnd_toEndOf="parent"
    app:layout_constraintHorizontal_bias="0.051"
    app:layout_constraintStart_toStartOf="parent"
    app:layout_constraintTop_toTopOf="parent"
    app:layout_constraintVertical_bias="0.469" />
```

```
<androidx.constraintlayout.widget.Guideline
android:id="@+id/guideline"
android:layout_width="wrap_content"
android:layout_height="wrap_content"
android:orientation="vertical"
app:layout_constraintGuide_begin="20dp" />
```

```
<androidx.constraintlayout.widget.Guideline
android:id="@+id/guideline2"
android:layout_width="wrap_content"
android:layout_height="wrap_content"
android:orientation="vertical"
app:layout_constraintGuide_begin="20dp" />
```

```
<androidx.constraintlayout.widget.Guideline
android:id="@+id/guideline3"
android:layout_width="wrap_content"
android:layout_height="wrap_content"
android:orientation="vertical"
app:layout_constraintGuide_begin="-17dp" />
<androidx.constraintlayout.widget.Barrier
android:id="@+id/barrier"
android:layout_width="wrap_content"
android:layout_height="wrap_content"
app:barrierDirection="top" />
```

```
<Button
android:id="@+id/subbtn"
android:layout_width="90dp"
android:layout_height="40dp"
android:text="SUB"
app:layout_constraintBottom_toBottomOf="parent"
app:layout_constraintEnd_toEndOf="parent"
app:layout_constraintHorizontal_bias="0.379"
app:layout_constraintStart_toStartOf="parent"
app:layout_constraintTop_toTopOf="parent"
app:layout_constraintVertical_bias="0.464" />
```

```
<Button
android:id="@+id/mulbtn"
android:layout_width="90dp"
android:layout_height="40dp"
android:layout_marginStart="40dp"
android:layout_marginLeft="40dp"
android:layout_marginTop="32dp"
android:layout_marginBottom="32dp"
android:text="MUL"
```

```
app:layout_constraintBottom_toBottomOf="parent"
app:layout_constraintEnd_toEndOf="parent"
app:layout_constraintHorizontal_bias="0.647"
app:layout_constraintStart_toStartOf="parent"
app:layout_constraintTop_toTopOf="parent"
app:layout_constraintVertical_bias="0.46" />
```

```
<Button
android:id="@+id/divbtn"
android:layout_width="90dp"
android:layout_height="40dp"
android:text="DIV"
app:layout_constraintBottom_toBottomOf="parent"
app:layout_constraintEnd_toEndOf="parent"
app:layout_constraintHorizontal_bias="1.0"
app:layout_constraintStart_toEndOf="@+id/mulbtn"
app:layout_constraintStart_toStartOf="parent"
app:layout_constraintTop_toTopOf="parent"
app:layout_constraintVertical_bias="0.464" />
```

```
<TextView
android:id="@+id/tans"
android:layout_width="165dp"
android:layout_height="27dp"
app:layout_constraintBottom_toBottomOf="parent"
app:layout_constraintEnd_toEndOf="parent"
app:layout_constraintHorizontal_bias="0.382"
app:layout_constraintStart_toStartOf="parent"
app:layout_constraintTop_toTopOf="parent"
app:layout_constraintVertical_bias="0.344" />
```

```
<androidx.constraintlayout.widget.Guideline
android:id="@+id/guideline4"
android:layout_width="wrap_content"
android:layout_height="wrap_content"
android:orientation="vertical"
app:layout_constraintGuide_begin="20dp" />
```

```
<TextView
android:id="@+id/textView2"
android:layout_width="wrap_content"
android:layout_height="wrap_content"
android:text="ENTER THE FIRST VALUE"
app:layout_constraintBottom_toBottomOf="parent"
app:layout_constraintEnd_toEndOf="parent"
app:layout_constraintEnd_toStartOf="@+id/text1"
app:layout_constraintHorizontal_bias="0.5"
```

```
app:layout_constraintStart_toStartOf="parent"
app:layout_constraintTop_toTopOf="parent"
app:layout_constraintVertical_bias="0.179" />
```

```
<TextView
android:id="@+id/textView3"
android:layout_width="163dp"
android:layout_height="32dp"
android:layout_marginStart="32dp"
android:layout_marginLeft="32dp"
android:text="ENTER THE SECOND VALUE"
app:layout_constraintBottom_toTopOf="@+id/tans"
app:layout_constraintStart_toStartOf="parent"
app:layout_constraintTop_toBottomOf="@+id/textView2"
app:layout_constraintVertical_bias="0.517" />
```

```
<EditText
android:id="@+id/text2"
android:layout_width="98dp"
android:layout_height="50dp"
android:ems="10"
android:inputType="textPersonName"
app:layout_constraintBottom_toTopOf="@+id/tans"
app:layout_constraintEnd_toEndOf="parent"
app:layout_constraintHorizontal_bias="0.168"
app:layout_constraintStart_toEndOf="@+id/textView3"
app:layout_constraintTop_toBottomOf="@+id/text1"
app:layout_constraintVertical_bias="0.51" />
```

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

MainActivity.java

```
package com.e11.myapplication10;
import android.os.Bundle;
import androidx.appcompat.app.AppCompatActivity;
import android.view.View;
import android.widget.Button;
import android.widget.EditText;
import android.widget.TextView;

public class MainActivity extends AppCompatActivity {
    Button btn1,btn2,btn3,btn4;
    EditText t1, t2;
    TextViewans;
    float d;
    protected void onCreate(Bundle savedInstanceState) {
```



```

super.onCreate(savedInstanceState);
setContentView(R.layout.activity_main);
btn1 = findViewById(R.id.addbtn);
btn2=findViewById(R.id.subbtn);
btn3=findViewById(R.id.mulbtn);
btn4=findViewById(R.id.divbtn);
t1 = findViewById(R.id.text1);
t2 = findViewById(R.id.text2);
ans=findViewById(R.id.tans);

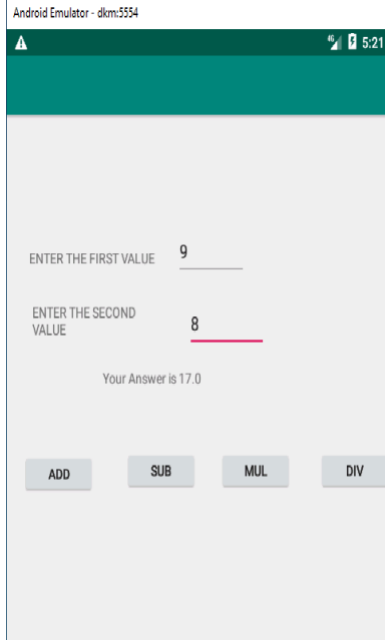
btn1.setOnClickListener(new View.OnClickListener() {
@Override
public void onClick(View view) {
d = Integer.parseInt(t1.getText().toString()) + Integer.parseInt(t2.getText().toString());
ans.setText("Your Answer is " + d);
}
});
btn2.setOnClickListener(new View.OnClickListener() {
@Override
public void onClick(View view) {
d = Integer.parseInt(t1.getText().toString()) - Integer.parseInt(t2.getText().toString());
ans.setText("Your Answer is " + d);
}
});
btn3.setOnClickListener(new View.OnClickListener() {
@Override
public void onClick(View view) {
d = Integer.parseInt(t1.getText().toString()) * Integer.parseInt(t2.getText().toString());
ans.setText("Your Answer is " + d);
}
});
btn4.setOnClickListener(new View.OnClickListener() {
@Override
public void onClick(View view) {
d = Integer.parseInt(t1.getText().toString()) / Integer.parseInt(t2.getText().toString());
ans.setText("Your Answer is " + d);
}
});
}
}
}

```

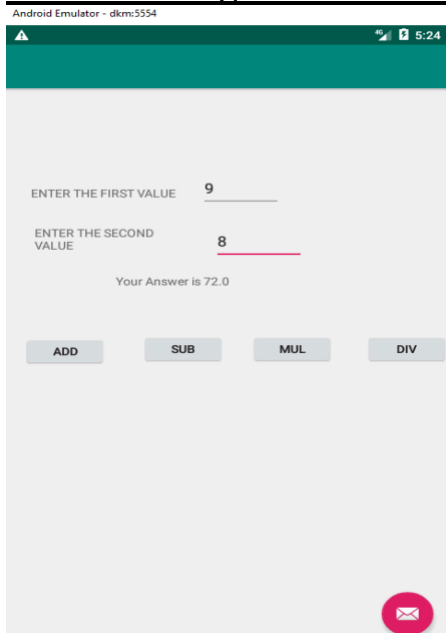
OUTPUT:

After selecting ADD button:

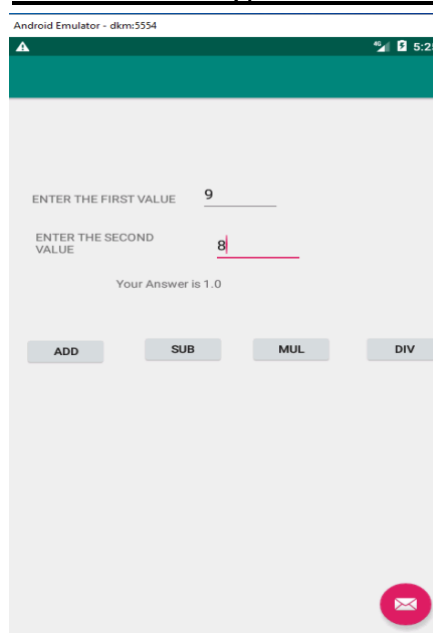
After selecting SUB button:



After selecting MUL button:



After selecting DIV button:



RESULT:

Thus the android application using Text View, Edit Text, Button has been executed successfully.

Ex.No.2**RADIOBUTTON,RADIOGROUP****AIM:**

To Develop a Android Application using Radio Group and Radio Button.

PROCEDURE:

Step 1: Start the process

Step 2: Create the new project by opening the android studio and then click on the file--> new-->new project.

Step 3: Select the basic activity and choose next option

Step 4: Change the Application Name and press Finish Button

Step 5: Next Going to Design layout by selecting content - main.xml

Step 6: Place the appropriate control for Developing the Application

Step 7: Change the constraints and required properties for the controls.

S.No.	Control	Control values	Properties & values
1	Button	Radio Button (RED)	ID (red)
2	Button	Radio Button (BLUE)	ID (blu)
3	Button	Radio Button (GREEN)	ID (gre)
4	Button	Button (FAVORITE COLOR)	ID (but)
5	Button	Radio Button (YELLOW)	ID (yel)
6	Button	Radio Button (MAGENTA)	ID (mag)
7	Button	Radio Button (BLACK)	ID (bla)
8	Button	Button (BACKGROUND COLOR)	ID (but2)
9	Text	Text View ()	ID (TextView)

Step 8: Select res--> values --> String.xml and add the String values

Step 9: Enter Java coding for the Application by select Mainactivity.java

Step 10: Stop the process.

PROGRAM:

Content_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"
    app:layout_behavior="@string/appbar_scrolling_view_behavior"
    tools:context=".MainActivity"
    tools:showIn="@layout/activity_main">
    <RadioGroup
        android:id="@+id/radioGroup"
        android:layout_width="270dp"
        android:layout_height="203dp"
        app:layout_constraintBottom_toBottomOf="parent"
        app:layout_constraintEnd_toEndOf="parent"
        app:layout_constraintHorizontal_bias="0.0"
        app:layout_constraintStart_toStartOf="parent"
        app:layout_constraintTop_toTopOf="parent"
        app:layout_constraintVertical_bias="0.0">
        <RadioButton
            android:id="@+id/red"
            android:layout_width="234dp"
            android:layout_height="49dp"
            android:text="RED" />
        <RadioButton
            android:id="@+id/blu"
            android:layout_width="234dp"
            android:layout_height="49dp"
            android:text="BLUE" />
        <RadioButton
            android:id="@+id/gre"
            android:layout_width="234dp"
            android:layout_height="49dp"
            android:text="GREEN" />
        <Button
            android:id="@+id/but"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
```

```
android:text="FAVOURITE COLOR" />
</RadioGroup>
```

```
<RadioGroup
android:layout_width="269dp"
android:layout_height="260dp"
android:layout_marginTop="32dp"
app:layout_constraintEnd_toEndOf="parent"
app:layout_constraintHorizontal_bias="0.039"
app:layout_constraintStart_toStartOf="parent"
app:layout_constraintTop_toBottomOf="@+id/radioGroup">
```

```
<RadioButton
android:id="@+id/yel"
android:layout_width="109dp"
android:layout_height="49dp"
android:text="YELLOW" />
```

```
<RadioButton
android:id="@+id/man"
android:layout_width="match_parent"
android:layout_height="49dp"
android:text="MAGENTA" />
```

```
<RadioButton
android:id="@+id/bla"
android:layout_width="match_parent"
android:layout_height="49dp"
android:text="BLACK" />
```

```
<Button
android:id="@+id/but2"
android:layout_width="wrap_content"
android:layout_height="wrap_content"
android:text="BACK GROUND" />
```

```
<TextView
android:id="@+id/textView1"
android:layout_width="match_parent"
android:layout_height="50dp" />
```

```
</RadioGroup>
```

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

String.Xml

```
<resources>
<string name="app_name">My Appradio</string>
<string name="action_settings">Settings</string>
<string name="red">RED</string>
<string name="blu">BLUE</string>
<string name="gre">GREEN</string>
<string name="yel">YELLOW</string>
<string name="man">MAGENTA</string>
<string name="bla">BLACK</string>
</resources>
```

MainActivity.java

```
package com.ex1.myappradio;

import android.graphics.Color;
import android.os.Bundle;

import androidx.appcompat.app.AppCompatActivity;

import android.view.View;
import android.widget.Button;
import android.widget.RadioButton;
import android.widget.TextView;
import android.widget.Toast;

public class MainActivity extends AppCompatActivity {
    RadioButton r1,r2,r3,r4,r5,r6;
    TextView t1;
    String selcolor;
    Button submit,submit2;
    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_main);
        r1=findViewById(R.id.red);
        r2=findViewById(R.id.blu);
        r3=findViewById(R.id.gre);
        r4=findViewById(R.id.yel);
        r5=findViewById(R.id.man);
        r6=findViewById(R.id.bla);
        t1=findViewById(R.id.textView1);
        submit=findViewById(R.id.but);
        submit2=findViewById(R.id.but2);
        submit.setOnClickListener(new View.OnClickListener() {
            @Override
            public void onClick(View view) {
                if(r1.isChecked())
```

```

        {
        selcolor=r1.getText().toString();
        }
        else if(r2.isChecked())
        {
        selcolor=r2.getText().toString();
        }
        else if(r3.isChecked())
        {
        selcolor=r3.getText().toString();
        }
        Toast.makeText(getApplicationContext(),selcolor,Toast.LENGTH_LONG).show();

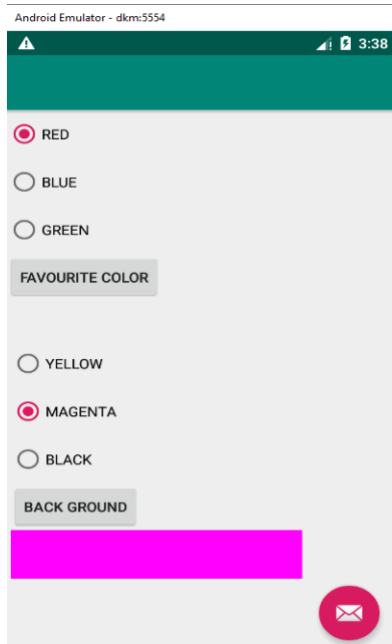
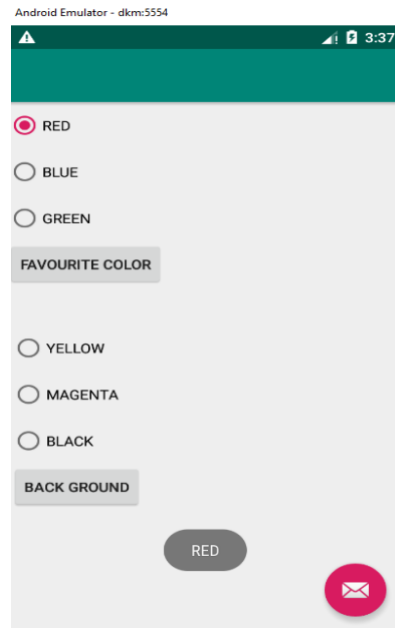
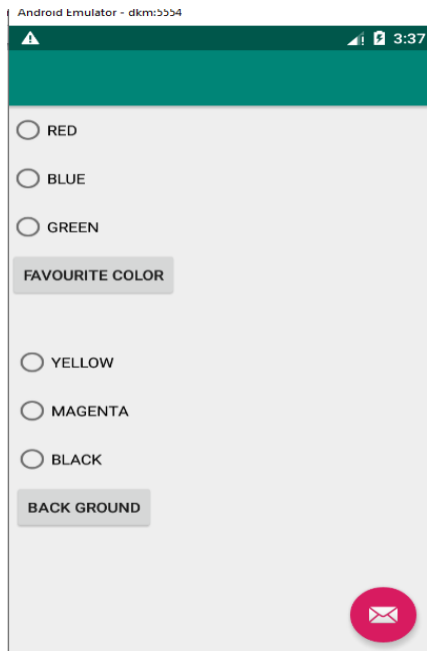
    }
});
submit2.setOnClickListener(new View.OnClickListener() {
    @Override
    public void onClick(View view) {
    if(r4.isChecked())
    {
    t1.setBackgroundColor(Color.YELLOW);

    }
    else if(r5.isChecked())
    {
    t1.setBackgroundColor(Color.MAGENTA);
    }
    else if(r6.isChecked())
    {
    t1.setBackgroundColor(Color.BLACK);
    }
    }

});
}
}

```

OUTPUT:



RESULT:

Thus the android application using Radio Group and Radio Button has been executed Successfully.

Ex. No. 3**CHECKBOX****AIM:**

To Develop a Android Application using CheckBox.

PROCEDURE:

Step 1: Start the proces.

Step 2: Create a new project by opening the Android Studio and then click on File --> new --> New project.

Step 3: Select the Basic Activity and choose next option

Step 4: Change the Application name and Pres Finish Button.

Step 5: Next Going to Design Layout by selecting the content main.xml

Step 6: Place the appropriate controls for Developing the application

Step 7: Change the constraints and required properties for the controls

S.No.	Control	Control Values	Properties & Values
1	Text	TextView (TOPPINGS)	ID (TextView)
2	Button	Check Box (WHIPPED CREAM)	ID (Check Box)
3	Button	Check Box (CHOCOLATE)	ID (Check Box2)
4	Text	TextView (QUANTITY)	ID (TextView2)
5	Button	Button (-)	ID (button)
6	Text	TextView (0)	ID (TextView3)
7	Button	Button (+)	ID (button2)
8	Text	TextView (ORDER SUMMARY)	ID (TextView4)
9	Button	Button (ORDER)	ID (button3)
10	Text	TextView ()	ID (TextView5)

Step 8: Enter Java coding for the application by selecting MainActivity.java

Step 9: Stop the process.

PROGRAM:

Content 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"
    app:layout_behavior="@string/appbar_scrolling_view_behavior"
    tools:context=".MainActivity"
    tools:showIn="@layout/activity_main">
```

```
<TextView
    android:id="@+id/textView"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:text="TOPPINGS"
    android:textAppearance="@style/TextAppearance.AppCompat.Large"
    app:layout_constraintBottom_toBottomOf="parent"
    app:layout_constraintEnd_toEndOf="parent"
    app:layout_constraintHorizontal_bias="0.207"
    app:layout_constraintStart_toStartOf="parent"
    app:layout_constraintTop_toTopOf="parent"
    app:layout_constraintVertical_bias="0.056" />
```

```
<CheckBox
    android:id="@+id/whipped_checkbox"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:text="Whipped Cream"
    android:textAppearance="@style/TextAppearance.AppCompat.Medium"
    app:layout_constraintBottom_toBottomOf="parent"
    app:layout_constraintEnd_toEndOf="parent"
    app:layout_constraintHorizontal_bias="0.187"
    app:layout_constraintStart_toStartOf="parent"
    app:layout_constraintTop_toTopOf="parent"
    app:layout_constraintVertical_bias="0.133" />
```

```
<CheckBox
    android:id="@+id/choco_checkbox"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:text="Chocolate"
    android:textAppearance="@style/TextAppearance.AppCompat.Medium"
    app:layout_constraintBottom_toBottomOf="parent"
    app:layout_constraintEnd_toEndOf="parent"
```

```
app:layout_constraintHorizontal_bias="0.165"  
app:layout_constraintStart_toStartOf="parent"  
app:layout_constraintTop_toBottomOf="@+id/whipped_checkbox"  
app:layout_constraintVertical_bias="0.049" />
```

```
<TextView  
android:id="@+id/textView2"  
android:layout_width="wrap_content"  
android:layout_height="wrap_content"  
android:text="QUANTITY"  
android:textAppearance="@style/TextAppearance.AppCompat.Large"  
app:layout_constraintBottom_toBottomOf="parent"  
app:layout_constraintEnd_toEndOf="parent"  
app:layout_constraintHorizontal_bias="0.178"  
app:layout_constraintStart_toStartOf="parent"  
app:layout_constraintTop_toBottomOf="@+id/whipped_checkbox"  
app:layout_constraintVertical_bias="0.171" />
```

```
<Button  
android:id="@+id/button"  
android:layout_width="wrap_content"  
android:layout_height="wrap_content"  
android:onClick="decrement"  
android:text="-"  
app:layout_constraintBottom_toBottomOf="parent"  
app:layout_constraintEnd_toEndOf="parent"  
app:layout_constraintHorizontal_bias="0.123"  
app:layout_constraintStart_toStartOf="parent"  
app:layout_constraintTop_toBottomOf="@+id/whipped_checkbox"  
app:layout_constraintVertical_bias="0.286" />
```

```
<Button  
android:id="@+id/button2"  
android:layout_width="wrap_content"  
android:layout_height="wrap_content"  
android:onClick="increment"  
android:text="+"  
app:layout_constraintBottom_toBottomOf="parent"  
app:layout_constraintEnd_toEndOf="parent"  
app:layout_constraintHorizontal_bias="0.548"  
app:layout_constraintStart_toEndOf="@+id/button"  
app:layout_constraintTop_toTopOf="parent"  
app:layout_constraintVertical_bias="0.419" />
```

```
<TextView  
android:id="@+id/quantity_text_view"  
android:layout_width="55dp"
```

```
android:layout_height="34dp"
android:text="0"
app:layout_constraintBottom_toBottomOf="parent"
app:layout_constraintEnd_toEndOf="parent"
app:layout_constraintHorizontal_bias="0.463"
app:layout_constraintStart_toStartOf="parent"
app:layout_constraintTop_toBottomOf="@+id/whipped_checkbox"
app:layout_constraintVertical_bias="0.291" />
```

```
<TextView
android:id="@+id/textView4"
android:layout_width="wrap_content"
android:layout_height="wrap_content"
android:text="ORDER SUMMARY"
android:textAppearance="@style/TextAppearance.AppCompat.Large"
app:layout_constraintBottom_toBottomOf="parent"
app:layout_constraintEnd_toEndOf="parent"
app:layout_constraintHorizontal_bias="0.147"
app:layout_constraintStart_toStartOf="parent"
app:layout_constraintTop_toBottomOf="@+id/quantity_text_view"
app:layout_constraintVertical_bias="0.136" />
```

```
<Button
android:id="@+id/button3"
android:layout_width="wrap_content"
android:layout_height="wrap_content"
android:onClick="submitOrder"
android:text="ORDER"
app:layout_constraintBottom_toBottomOf="parent"
app:layout_constraintEnd_toEndOf="parent"
app:layout_constraintHorizontal_bias="0.727"
app:layout_constraintStart_toStartOf="parent"
app:layout_constraintTop_toBottomOf="@+id/button2"
app:layout_constraintVertical_bias="0.104" />
```

```
<TextView
android:id="@+id/order_summary_text_view"
android:layout_width="187dp"
android:layout_height="126dp"
app:layout_constraintBottom_toBottomOf="parent"
app:layout_constraintEnd_toEndOf="parent"
app:layout_constraintHorizontal_bias="0.147"
app:layout_constraintStart_toStartOf="parent"
app:layout_constraintTop_toBottomOf="@+id/quantity_text_view"
app:layout_constraintVertical_bias="0.338" />
</androidx.constraintlayout.widget.ConstraintLayout>
```

MainActivity.java

```
package com.e11.myapplicationcheck;
import android.os.Bundle;
import com.google.android.material.floatingactionbutton.FloatingActionButton;
import com.google.android.material.snackbar.Snackbar;
import androidx.appcompat.app.AppCompatActivity;
import androidx.appcompat.widget.Toolbar;
import android.view.View;
import android.view.Menu;
import android.view.MenuItem;
import android.widget.CheckBox;
import android.widget.TextView;
import android.widget.Toast;

public class MainActivity extends AppCompatActivity {
    int quantity=0,price_per_cup=5;

    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_main);
        View des=new TextView(getApplicationContext());
        des.setVisibility(View.GONE);
    }

    public void submitOrder(View view)
    {
        createOrderSummary();
        Toast toast=Toast.makeText(getApplicationContext(),"Thank You!",
        Toast.LENGTH_SHORT);
        toast.show();
    }

    private int calculatePrice(int quantity,int price_per_cup)
    {
        return quantity*price_per_cup;
    }

    private void createOrderSummary()
    {
        CheckBox iswhipped=findViewById(R.id.whipped_checkbox);
        CheckBox ischocolate=findViewById(R.id.choco_checkbox);
        int total_price=calculatePrice(quantity,price_per_cup);
        String report="Name: SIVAGAMI.R"+"\\n"+"Add Whipped
        Cream?" +iswhipped.isChecked()+"\\n"+"Add
        Chocolate?" +ischocolate.isChecked()+"\\n"+"Quantity:" +quantity+"\\n"+"Price:$"+total_price;
        TextView reportTextView=findViewById(R.id.order_summary_text_view);
        reportTextView.setText(" "+report);
    }
}
```

```

    }

    public void decrement(View view)
    {
    if(quantity<=0)
    quantity=0;
    else
    quantity -=1;
    displayMessage(quantity);
    }

    public void increment(View view)
    {
    quantity +=1;
    displayMessage(quantity);
    }
    private void displayMessage(intvalue)
    {
    TextViewquantityTextView=findViewById(R.id.quantity_text_view);
    quantityTextView.setText(""+value);

    }

    private void displayPrice(intnumber)
    {
    String price="Total of $" +number+"\n"+"Thanks!";
    displayMessage(price);

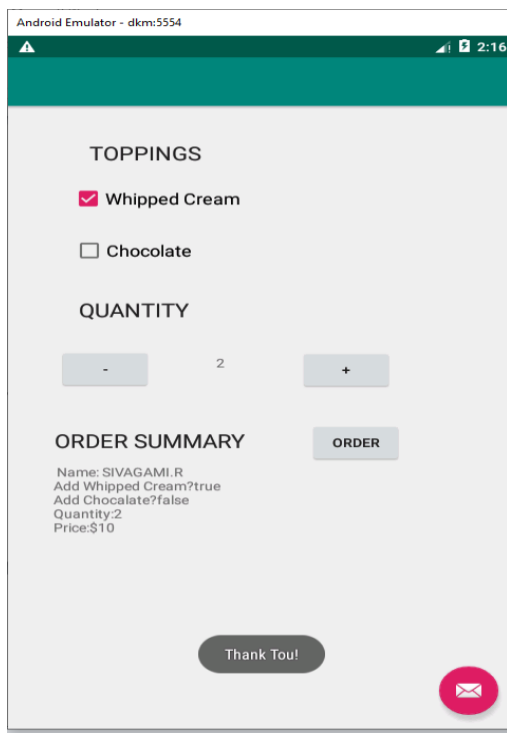
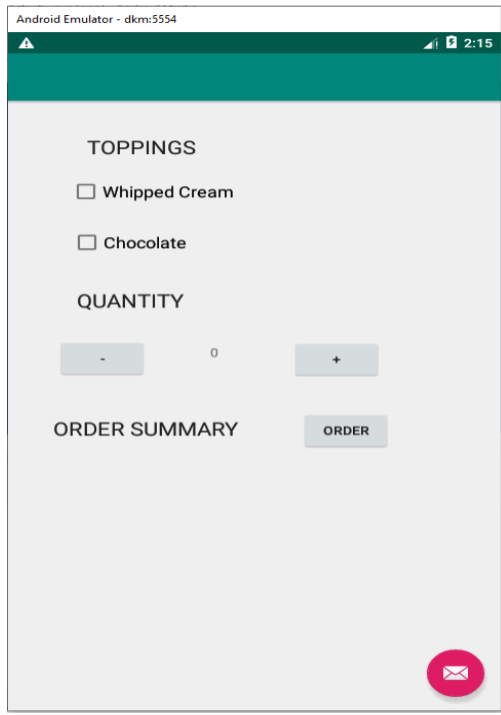
    }

    private void displayMessage(String msg)
    {
    TextViewmsgTextView=findViewById(R.id.order_summary_text_view);
    msgTextView.setText(msg);
    }
}

```

OUTPUT:

3.CHECK BOX



RESULT:

Thus the Android Application using checkbox has been executed successfully.

Ex. No: 4

DIALOG BOX

AIM:

To Develop a Android Application using Dialog Box.

PROCEDURE:

Step 1: Start the process.

Step 2: Create the New project by opening the Android Studio and then click on File -->new --> New project.

Step 3: Select the Basic Activity and Choose next option

Step 4: Change the application name and press finish button

Step 5: Design the Layout by selecting Content_main.xml

Step 6: Place the appropriate controls for Developing the Applications

Step 7: Change the constraints and required properties for the controls.

S.No.	Control	Control values	Properties & Values
1	Button	ImageButton ()	ID (ImageButton)
2	Layout	Guideline(Horizontal)	ID (guideline)

Step 8: Enter Java coding for the Application by select MainActivity.java

Step 9: Stop the process.

PROGRAM:**Content 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"
app:layout_behavior="@string/appbar_scrolling_view_behavior"
tools:context=".MainActivity"
tools:showIn="@layout/activity_main">

<ImageButton
android:id="@+id/imageButton"
android:layout_width="140dp"
```



```
android:layout_height="132dp"
app:layout_constraintBottom_toBottomOf="parent"
app:layout_constraintEnd_toEndOf="parent"
app:layout_constraintHorizontal_bias="0.457"
app:layout_constraintStart_toStartOf="parent"
app:layout_constraintTop_toTopOf="parent"
app:layout_constraintVertical_bias="0.369"
app:srcCompat="@mipmap/ic_launcher_round" />
```

```
<androidx.constraintlayout.widget.Guideline
android:id="@+id/guideline"
android:layout_width="wrap_content"
android:layout_height="wrap_content"
android:orientation="horizontal"
app:layout_constraintGuide_begin="20dp" />
</androidx.constraintlayout.widget.ConstraintLayout>
```

MainActivity.java

```
package com.e11.myapplicationdialog;
import android.content.DialogInterface;
import android.os.Bundle;
import com.google.android.material.floatingactionbutton.FloatingActionButton;
import com.google.android.material.snackbar.Snackbar;
import androidx.appcompat.app.AlertDialog;
import androidx.appcompat.app.AppCompatActivity;
import androidx.appcompat.widget.Toolbar;
```

```
import android.view.View;
import android.view.Menu;
import android.view.MenuItem;
import android.widget.Toast;
```

```
public class MainActivity extends AppCompatActivity {
    Toast msg;
```

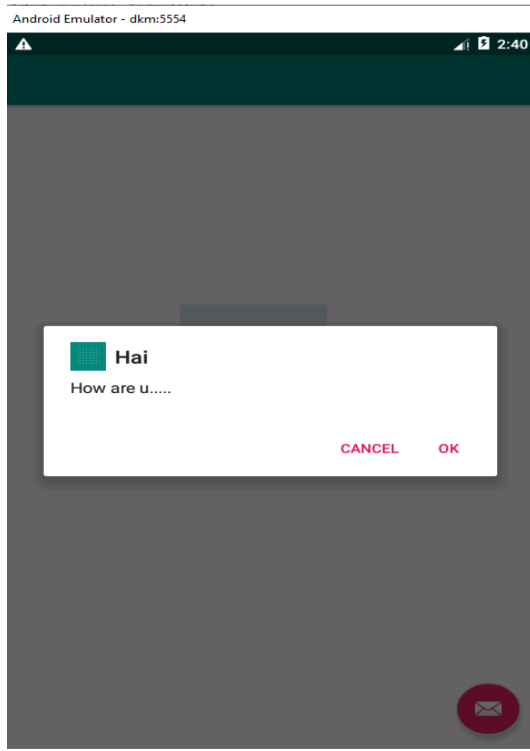
```
protected void onCreate(Bundle savedInstanceState) {
    super.onCreate(savedInstanceState);
    setContentView(R.layout.activity_main);
    AlertDialog.Builder adb=new AlertDialog.Builder(this);
    adb.setTitle("Hai");
    adb.setMessage("How are u.....");
    adb.setPositiveButton("ok", new DialogInterface.OnClickListener() {
```

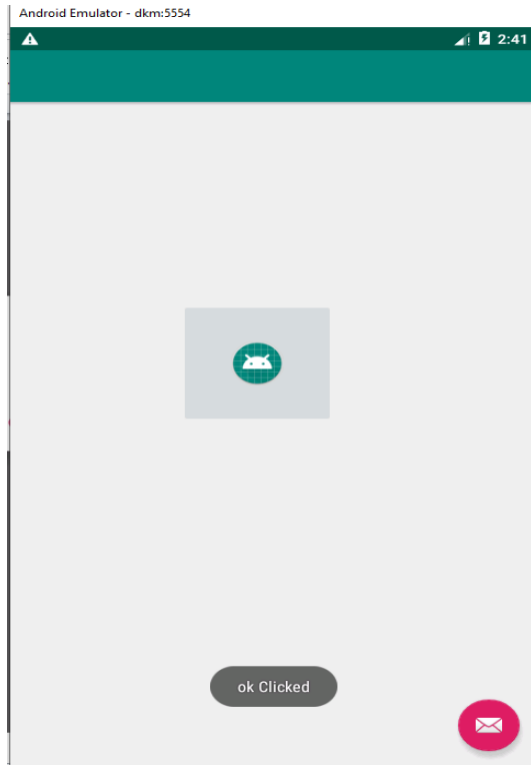
```
public void onClick(DialogInterface dialogInterface, inti) {
    msg=Toast.makeText(MainActivity.this,"ok Clicked",Toast.LENGTH_LONG);
    msg.show();
    }
}
```

```
});  
adb.setNegativeButton("cancel", new DialogInterface.OnClickListener() {  
  
public void onClick(DialogInterface dialogInterface, inti) {  
msg=Toast.makeText(MainActivity.this,"cancel Clicked",Toast.LENGTH_LONG);  
msg.show();  
dialogInterface.cancel();  
}  
});  
adb.setIcon(R.drawable.ic_launcher_background);  
adb.show();  
  
}  
}
```

OUTPUT:

4.DIALOG BOX





RESULT:

Thus the Android Application using Dialog Box has been Executed Successfully.

Ex.No.5**DATE PICKER****AIM:**

To Develop a android Application using Date Picker

PROCEDURE:

Step 1 : Start the process.

Step 2: Create a new Project by opening the Android Studio and then click on File --> new -->new project.

Step 3: Select the basic activity and choose next option.

Step 4: Change the Application name and press finish button.

Step 5: Design the layout by selecting content - main.xml

Step 6: Place the appropriate Controls for Developing the Application

Step 7: Change the constraints and required properties for the controls

S.No.	Control	Control Values	Properties & Values
1	Text	Plain Text ()	ID (Select date) Common Attribute (Hint=Select date...)

Step 8: Enter Java coding for Application by selecting MainActivity.java

Step 9: Stop the process

PROGRAM:**Content 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"
app:layout_behavior="@string/appbar_scrolling_view_behavior"
tools:context=".MainActivity"
tools:showIn="@layout/activity_main">

<EditText
android:id="@+id/SelectDate"
```

```
android:layout_width="wrap_content"
android:layout_height="wrap_content"
android:ems="10"
android:hint="Select date..."
android:inputType="textPersonName"
app:layout_constraintBottom_toBottomOf="parent"
app:layout_constraintEnd_toEndOf="parent"
app:layout_constraintStart_toStartOf="parent"
app:layout_constraintTop_toTopOf="parent"
app:layout_constraintVertical_bias="0.153" />
</androidx.constraintlayout.widget.ConstraintLayout>
```

MainActivity.java

```
package com.e11.myapplicationdatepicker;
```

```
import android.app.DatePickerDialog;
import android.os.Bundle;
import androidx.appcompat.app.AppCompatActivity;
import android.view.View;
import android.widget.DatePicker;
import android.widget.EditText;
import java.util.Calendar;
```

```
public class MainActivity extends AppCompatActivity {
```

```
    EditText date;
```

```
    DatePickerDialog datePickerDialog;
```

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

```
        super.onCreate(savedInstanceState);
```

```
        setContentView(R.layout.activity_main);
```

```
        date=findViewById(R.id.SelectDate);
```

```
        date.setOnClickListener(new View.OnClickListener() {
```

```
            @Override
```

```
            public void onClick(View view) {
```

```
                final Calendar c=Calendar.getInstance();
```

```
                int mYear=c.get(Calendar.YEAR);
```

```
                int mMonth=c.get(Calendar.MONTH);
```

```
                int mDay=c.get(Calendar.DAY_OF_MONTH);
```

```
                datePickerDialog=new DatePickerDialog(MainActivity.this, new
```

```
                DatePickerDialog.OnDateSetListener() {
```

```
                    @Override
```

```
                    public void onDateSet(DatePicker view, int year, int monthOfYear, int dayOfMonth) {
```

```
                        date.setText(dayOfMonth+"/"+(monthOfYear+1)+"/"+year);
```

```
                    }
```

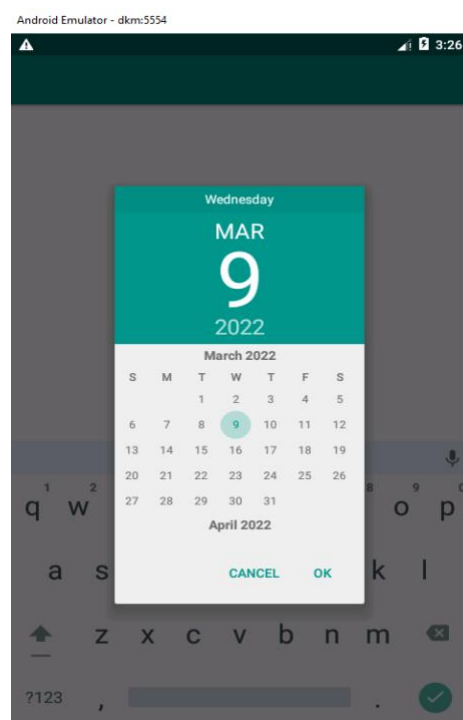
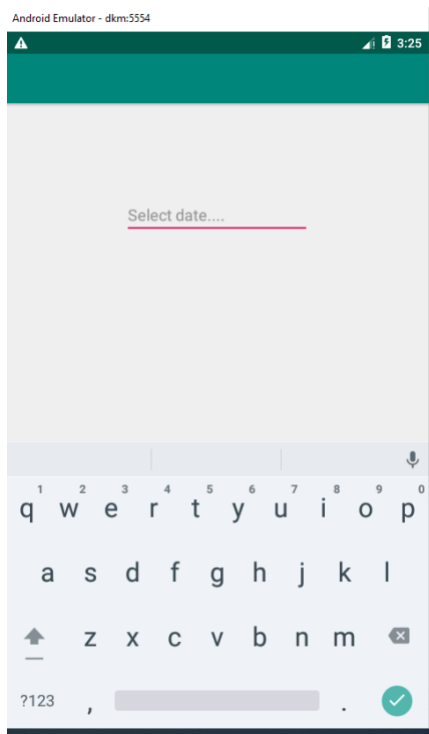
```
                },mYear,mMonth,mDay);
```

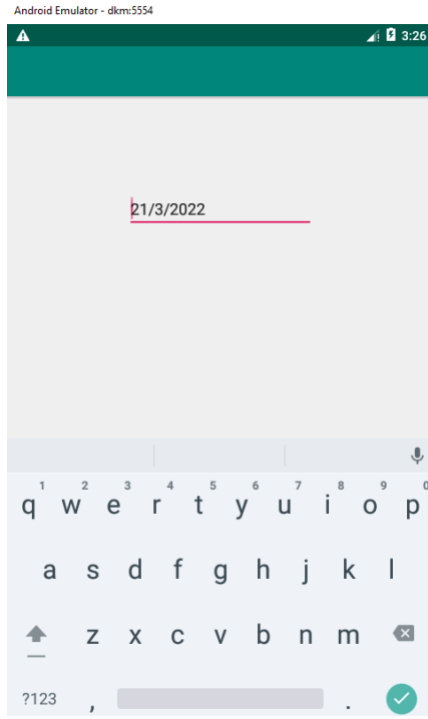
```
                datePickerDialog.show();
```

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

OUTPUT:

5.DATE PICKER





RESULT:

Thus the Android Application using Date Picker has been executed successfully.

Ex.No.6

PROGRESS BAR

AIM:

To Develop a android Application using Progress Bar.

PROCEDURE:

Step 1 : Start the process.

Step 2: Create a new Project by opening the Android Studio and then click on File --> new -->new project.

Step 3: Select the basic activity and choose next option.

Step 4: Change the Application name and press finish button.

Step 5: Design the layout by selecting content - main.xml

Step 6: Place the appropriate Controls for Developing the Application

Step 7: Change the constraints and required properties for the controls

S.No.	Control	Control Values	Properties & Values
1	Text	TextView (PROGRESS BAR)	ID (Text View) ALL ATTRIBUTES (Text Color=#00574B Text size=50dp)
2	Text	TextView (See Your Progress Here...)	ID (Text View2)
3	Widgets	Progress Bar(Horizontal)	ID (P_Bar) COMMON ATTRIBUTES (Style: progressBarStyleHorizontal Max:100 Progress:0)
4	Text	TextView ()	ID (tv)
5	Button	Button (Start)	ID (btn)

Step 8: Enter Java coding for Application by selecting MainActivity.java

Step 9: Stop the process

PROGRAM:

Content 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"
app:layout_behavior="@string/appbar_scrolling_view_behavior"
tools:context=".MainActivity"
tools:showIn="@layout/activity_main">

<TextView
android:id="@+id/textView"
android:layout_width="wrap_content"
android:layout_height="wrap_content"
android:text="PROGRESS BAR"
android:textColor="#00574B"
android:textSize="50dp"
app:layout_constraintBottom_toBottomOf="parent"
```



```
app:layout_constraintEnd_toEndOf="parent"
app:layout_constraintHorizontal_bias="0.496"
app:layout_constraintStart_toStartOf="parent"
app:layout_constraintTop_toTopOf="parent"
app:layout_constraintVertical_bias="0.047" />
```

```
<TextView
android:id="@+id/txtview"
android:layout_width="276dp"
android:layout_height="34dp"
android:text="See Your Progress Here...."
app:layout_constraintBottom_toBottomOf="parent"
app:layout_constraintEnd_toEndOf="parent"
app:layout_constraintHorizontal_bias="0.518"
app:layout_constraintStart_toStartOf="parent"
app:layout_constraintTop_toBottomOf="@+id/textView"
app:layout_constraintVertical_bias="0.06" />
```

```
<ProgressBar
android:id="@+id/p_Bar"
style="?android:attr/progressBarStyleHorizontal"
android:layout_width="232dp"
android:layout_height="24dp"
android:max="100"
android:progress="0"
app:layout_constraintBottom_toBottomOf="parent"
app:layout_constraintEnd_toEndOf="parent"
app:layout_constraintHorizontal_bias="0.636"
app:layout_constraintStart_toStartOf="parent"
app:layout_constraintTop_toBottomOf="@+id/txtview"
app:layout_constraintVertical_bias="0.063" />
```

```
<TextView
android:id="@+id/tv"
android:layout_width="273dp"
android:layout_height="53dp"
app:layout_constraintBottom_toBottomOf="parent"
app:layout_constraintEnd_toEndOf="parent"
app:layout_constraintStart_toStartOf="parent"
app:layout_constraintTop_toBottomOf="@+id/p_Bar"
app:layout_constraintVertical_bias="0.092" />
```

```
<Button
android:id="@+id/show_btn"
android:layout_width="wrap_content"
android:layout_height="wrap_content"
android:text="Start"
```

```
app:layout_constraintBottom_toBottomOf="parent"
app:layout_constraintEnd_toEndOf="parent"
app:layout_constraintHorizontal_bias="0.498"
app:layout_constraintStart_toStartOf="parent"
app:layout_constraintTop_toBottomOf="@+id/tv"
app:layout_constraintVertical_bias="0.125" />
</androidx.constraintlayout.widget.ConstraintLayout>
```

Mainactivity.java

```
package com.ex1.myappprogress;
import android.os.Bundle;
import com.google.android.material.floatingactionbutton.FloatingActionButton;
import com.google.android.material.snackbar.Snackbar;
import androidx.appcompat.app.AppCompatActivity;
import androidx.appcompat.widget.Toolbar;

import android.os.Handler;
import android.view.View;
import android.view.Menu;
import android.view.MenuItem;
import android.widget.Button;
import android.widget.ProgressBar;
import android.widget.TextView;
public class MainActivity extends AppCompatActivity {
    private ProgressBar pbar;
    private int a = 0;
    private TextView textView;
    private Handler handler = new Handler();
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_main);
        textView = findViewById(R.id.tv);
        pbar = findViewById(R.id.p_Bar);
        Button button = findViewById(R.id.show_btn);
        button.setOnClickListener(new View.OnClickListener() {
            @Override
            public void onClick(View view) {
                a = pbar.getProgress();
                new Thread(new Runnable() {
                    @Override
                    public void run() {
                        while (a < 100) {
                            a += 1;
                            handler.post(new Runnable() {
                                @Override
```

```

public void run() {
    pbar.setProgress(a);
    textView.setText(a + "/" + pbar.getMax());
    if (a == 100)
        textView.setText("Your Progress has been completed");
    }
    });

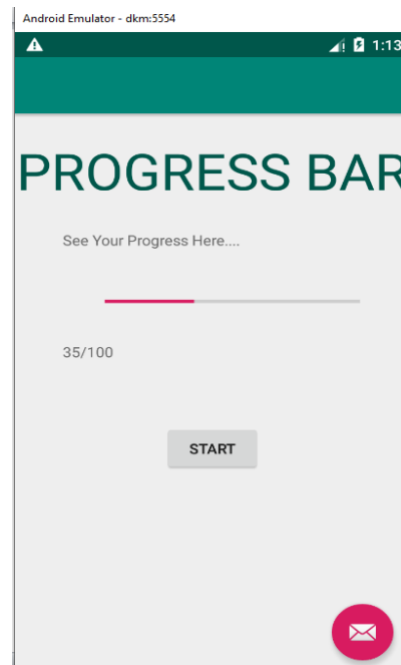
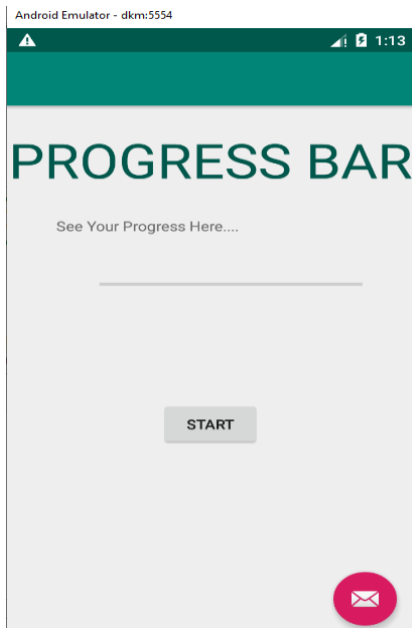
try {
    Thread.sleep(50);
    } catch (InterruptedException e) {
    e.printStackTrace();
    }
    }

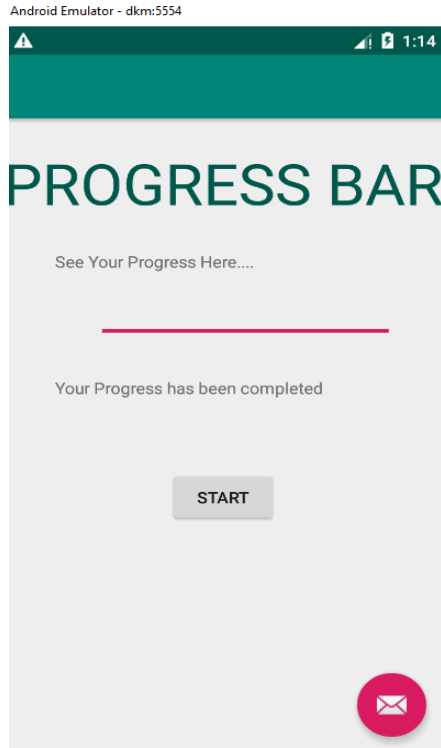
    }).start();
    }
    });
}
}

```

OUTPUT:

6.PROGRESS BAR





RESULT:

Thus the Android Application using Progress Bar has been executed successfully.

Ex.No.7

SPINNER

AIM:

To Develop a android Application using Spinner

PROCEDURE:

Step 1 : Start the process.

Step 2: Create a new Project by opening the Android Studio and then click on File --> new -->new project.

Step 3: Select the basic activity and choose next option.

Step 4: Change the Application name and process press finish button.

Step 5: Open res--> layout -->Activity_main.xml

Step 6: Place the appropriate Controls for Developing the Application

Step 7: Change the constraints and required properties for the controls

S.NO	Controls	Control Values	Properties & Values
1	Container	Spinner	ID (SimpleSpinner)
2	Button	FloatingActionButton	ID (fab)

Step 8: Enter Java coding for Application by selecting MainActivity.java

Step 9: Stop the process

PROGRAM:

Activity_main.xml

```
<?xml version="1.0" encoding="utf-8"?>
<androidx.coordinatorlayout.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"
    tools:context=".MainActivity">

    <com.google.android.material.appbar.AppBarLayout
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:theme="@style/AppTheme.AppBarOverlay">

        </com.google.android.material.appbar.AppBarLayout>

    <Spinner
        android:id="@+id/simpleSpinner"
        android:layout_width="399dp"
        android:layout_height="wrap_content"
        android:layout_centerHorizontal="true"
        android:layout_marginTop="100dp" />
    <include layout="@layout/content_main" />

    <com.google.android.material.floatingactionbutton.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"
app:srcCompat="@android:drawable/ic_dialog_email" />
```

```
<androidx.appcompat.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" />

</androidx.coordinatorlayout.widget.CoordinatorLayout>
```

MainActivity.java

```
package com.ex1.spin07;
import android.os.Bundle;
import com.google.android.material.floatingactionbutton.FloatingActionButton;
import com.google.android.material.snackbar.Snackbar;
import androidx.appcompat.app.AppCompatActivity;
import androidx.appcompat.widget.Toolbar;
import android.view.View;
import android.view.Menu;
import android.view.MenuItem;
import android.widget.AdapterView;
import android.widget.ArrayAdapter;
import android.widget.Spinner;
import android.widget.Toast;

public class MainActivity extends AppCompatActivity implements
AdapterView.OnItemSelectedListener {
    String[] bankNames={"SBI","HDFC","ICICI","AXIS",};

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

        Spinner spin = (Spinner) findViewById(R.id.simpleSpinner);

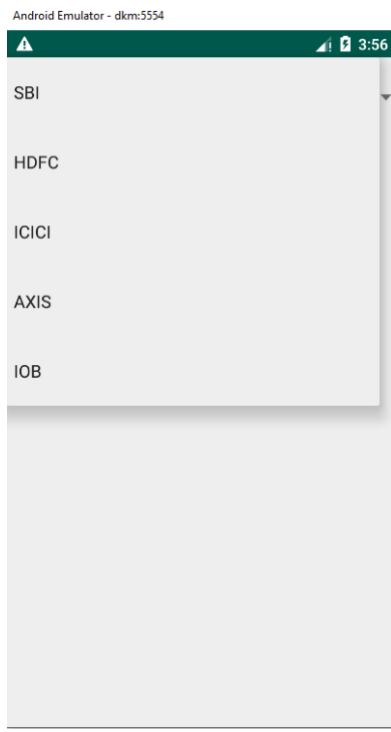
        spin.setOnItemSelectedListener(this);
        ArrayAdapter aa = new ArrayAdapter(this, android.R.layout.simple_spinner_item, bankNames);
        aa.setDropDownViewResource(android.R.layout.simple_dropdown_item_1line);
        spin.setAdapter(aa);
    }
    public void onItemSelected(AdapterView<?> arg0, View arg1, int position, long id) {
        Toast.makeText(getApplicationContext(), bankNames[position], Toast.LENGTH_LONG).show();
    }
}
```

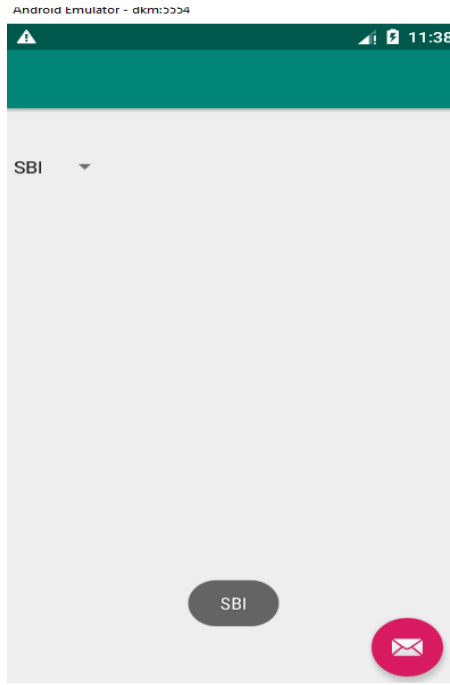
```
public void onNothingSelected(AdapterView<?> arg0)
{

}
}
```

OUTPUT:

7.SPINNER





RESULT:

Thus the Android Application using Spinner has been executed successfully.

Ex.No.8 SIMPLE MENU

AIM:

To Develop a android Application using Simple Menu.

PROCEDURE:

Step 1 : Start the process.

Step 2: Create a new Project by opening the Android Studio and then click on File --> new -->new project.

Step 3: Select the basic activity and choose next option.

Step 4: Change the Application name and process press finish button.

Step 5: Design the Layout by Selecting Content_Main.xml

S.No.	Control	Control Values	Properties & values
1	Text	Text View (Programming Languages)	ID (Text)

Step 6: Click on res folder and create New menu source Directory

Step 7: Create New menu Source file options.xml under Menu folder by clicking on it.

Step 8: Place the Menu control and submenu controls in the design window of the options.xml

Go to res -> Menu control -> In sub menu ->Click option.xml

S.No.	Control	Control Values	Properties & values
1	MenuItem	Item-->Title (C Programming)	ID (i1)
2	MenuItem	Item-->Title (C++ Programming)	ID (i2)
3	MenuItem	Item-->Title (Java Programming)	ID (i3)
4	MenuItem	Item-->Title (Python Programming)	ID (i4)
5	Menu-> MenuItem	Item-->Title (Expert)	ID (i5)
6	MenuItem	Item-->Title (Medium Level)	ID (i6)
7	MenuItem	Item-->Title (Beginners Level)	ID (i7)

Step 9: Add the java code for the Application which contains Menu options

Step 10: Run the Application

Step 11: Stop the process

PROGRAM:

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">
    <TextView
        android:id="@+id/text1"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="Programming Languages"
        app:layout_constraintBottom_toBottomOf="parent"
        app:layout_constraintLeft_toLeftOf="parent"
        app:layout_constraintRight_toRightOf="parent"
```

```
app:layout_constraintTop_toTopOf="parent" />
</androidx.constraintlayout.widget.ConstraintLayout>
```

Options.xml

```
<?xml version="1.0" encoding="utf-8"?>
<menu xmlns:android="http://schemas.android.com/apk/res/android">
<item
android:id="@+id/i1"
android:title="C Programming" />
<item
android:id="@+id/i2"
android:title="C++ Programming" />
<item
android:id="@+id/i3"
android:title="Java" />
<item
android:id="@+id/i4"
android:title="Python">
<menu>
<item
android:id="@+id/i5"
android:title="Expert" />
<item
android:id="@+id/i6"
android:title="medium level" />
<item
android:id="@+id/i7"
android:title="Beginner" />
</menu>
</item>
</menu>
```

MainActivity.java

```
package com.ex1.menuprogram;
import androidx.annotation.NonNull;
import androidx.appcompat.app.AppCompatActivity;
import android.os.Bundle;
import android.view.Menu;
import android.view.MenuInflater;
import android.view.MenuItem;
import android.widget.TextView;
import android.widget.Toast;

public class MainActivity extends AppCompatActivity {
    TextView t1;
    @Override
    protected void onCreate(Bundle savedInstanceState) {
```

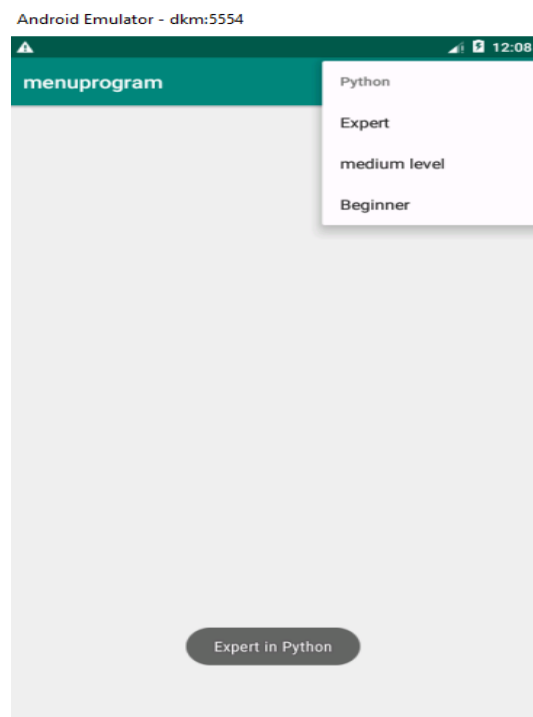
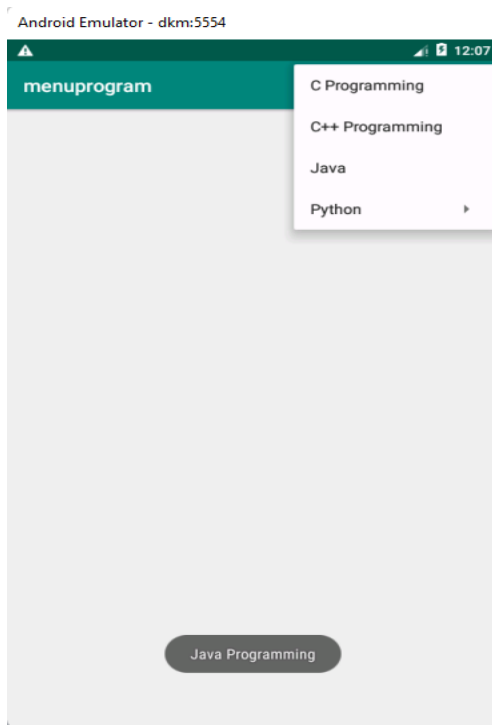
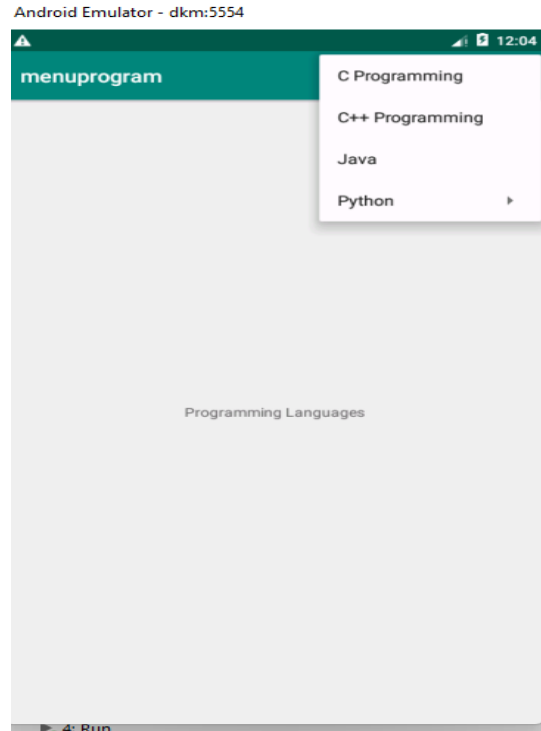
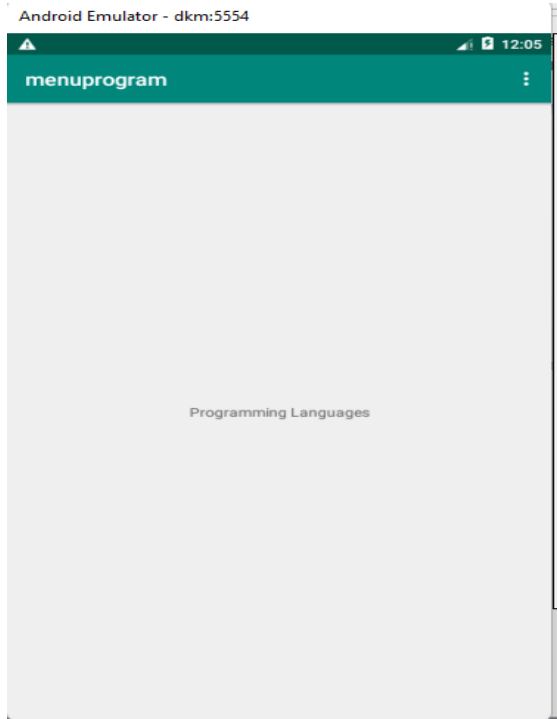
```
super.onCreate(savedInstanceState); setContentView(R.layout.activity_main);
t1=findViewById(R.id.text1);
}
```

```
@Override
public boolean onCreateOptionsMenu(Menu menu) {
    MenuInflater m=getMenuInflater();
    m.inflate(R.menu.options,menu);
    return true;
}
```

```
@Override
public boolean onOptionsItemSelected(@NonNull MenuItem item) {
    switch(item.getItemId()){
    case R.id.i1:
        t1.setText("");
        Toast.makeText(this,"C Programming",Toast.LENGTH_SHORT).show();
        return true;
    case R.id.i2:
        t1.setText("");
        Toast.makeText(this,"C++ Programming",Toast.LENGTH_SHORT).show();
        return true;
    case R.id.i3:
        t1.setText("");
        Toast.makeText(this,"Java Programming",Toast.LENGTH_SHORT).show();
        return true;
    case R.id.i4:
        t1.setText("");
        Toast.makeText(this,"Python Programming",Toast.LENGTH_SHORT).show();
        return true;
    case R.id.i5:
        t1.setText("");
        Toast.makeText(this,"Expert in Python",Toast.LENGTH_SHORT).show();
        return true;
    }
    return true;
}
}
```

OUTPUT:

8. SIMPLE MENU



RESULT:

Thus the Android Application using Simple Menu has been executed successfully.

Ex.No.9**MENU WITH INTENTS****AIM:**

To Develop a android Application using Menu with intents.

PROCEDURE:

Step 1 : Start the process.

Step 2: Create a new Project by opening the Android Studio and then click on File --> new -->new project.

Step 3: Select the basic activity and choose next option.

Step 4: Change the Application name and process press finish button.

Step 5: Design the Layout by Selecting Content_Main.xml

Step 6: Place the appropriate controls for Developing the Applications

Step 7: Change the constraints and required properties for the controls.

S.No.	Control	Control Values	Properties & values
1	Text	Edit Text	ID (nameEt) COMMON ATTRIBUTES (hint:Enter name)
2	Text	Edit Text	ID (emailEt) COMMON ATTRIBUTES (hint:Enter E-mail)
3	Text	Edit Text	ID (PhoneET) COMMON ATTRIBUTES (hint:Enter Phone no)
4	Button	Button (SAVE)	ID (savebtn)

Step 8: Enter Java coding for the Application by select MainActivity.java

Step 9: Select java-->com.blogspot.atifsoftwares.myapplication-->New-->activity-->Empty activity

Step 10: Change te name of the activity as Second activity and press finish button.

Step 11: Place the appropriate controls in activity_second.xml

Step 12: Enter the java coding for the application by selecting second activity.java

Step 13: Stop the process

PROGRAM:

activity_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"
    tools:context=".MainActivity"
    android:orientation="vertical"
    android:gravity="center">
    <EditText
        android:id="@+id/nameEt"
        android:hint="Enter Name"
        android:inputType="text"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"/>
    <EditText
        android:id="@+id/emailEt"
        android:hint="Enter Email"
        android:inputType="text"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"/>
    <EditText
        android:id="@+id/phoneET"
        android:hint="Enter Phone"
        android:inputType="text"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"/>
    <Button
        android:id="@+id/saveBtn"
        android:text="Save"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"/>

</LinearLayout>
```

MainActivity.java

```
package com.ex1.myapplication11;
import android.content.Intent;
import android.os.Bundle;

import com.google.android.material.floatingactionbutton.FloatingActionButton;
import com.google.android.material.snackbar.Snackbar;

import androidx.appcompat.app.ActionBar;
```

```

import androidx.appcompat.app.AppCompatActivity;
import androidx.appcompat.widget.Toolbar;

import android.view.View;
import android.view.Menu;
import android.view.MenuItem;
import android.widget.Button;
import android.widget.EditText;

public class MainActivity extends AppCompatActivity {

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

        ActionBar actionBar=getSupportActionBar();
        actionBar.setTitle("First Activity");

        final EditText mNameEt=findViewById(R.id.nameEt);
        final EditText mEmailEt=findViewById(R.id.emailEt);
        final EditText mPhoneEt=findViewById(R.id.phoneET);

        Button mSavebtn=findViewById(R.id.saveBtn);
        mSavebtn.setOnClickListener(new View.OnClickListener() {
            @Override
            public void onClick(View view) {
                String name=mNameEt.getText().toString();
                String email=mEmailEt.getText().toString();
                String phone=mPhoneEt.getText().toString();

                Intent intent=new Intent(MainActivity.this,secondActivity.class);
                intent.putExtra("NAME",name);
                intent.putExtra("Email",email);
                intent.putExtra("Phone",phone);
                startActivity(intent);

            }
        });
    }
}

```

activity_second.xml

```

<?xml version="1.0" encoding="utf-8"?>
<LinearLayoutxmlns: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:layout_gravity="center"
android:padding="10dp"
tools:context=".secondActivity">
<TextView
android:id="@+id/resultTv"
android:textSize="30sp"
android:textStyle="bold"
android:layout_width="match_parent"
android:layout_height="wrap_content"/>
</LinearLayout>

```

SecondActivity.java

```

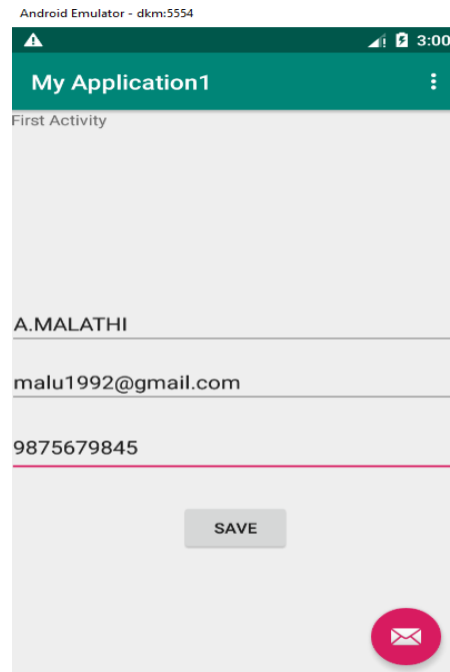
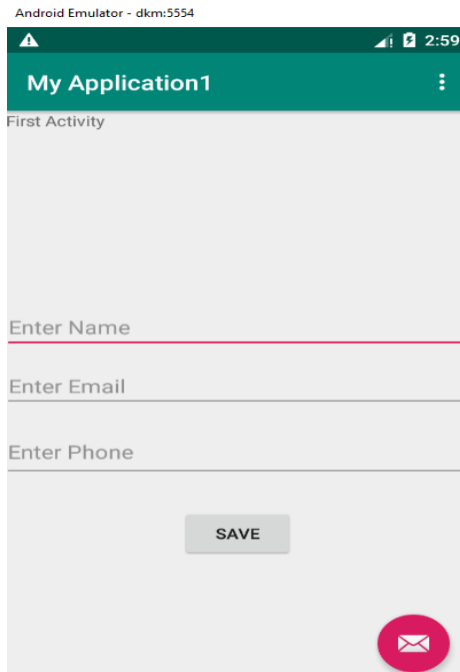
package com.ex1.myapplication11;
import androidx.appcompat.app.ActionBar;
import androidx.appcompat.app.AppCompatActivity;
import android.content.Intent;
import android.os.Bundle;
import android.widget.TextView;
public class secondActivityextends AppCompatActivity {
    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_second);
        ActionBaractionBar=getSupportActionBar();
        actionBar.setTitle("SecondActivity");
        Intent intent=getIntent();
        String name=intent.getStringExtra("NAME");
        String email=intent.getStringExtra("EMMAIL");
        String phone=intent.getStringExtra("PHONE");
        TextViewmResultTv=findViewById(R.id.resultTv);
        mResultTv.setText("Name:"+name+"\nEmail:"+email+"\n Phone:"+phone);

    }
}

```

OUTPUT:

9.MENU WITH INTENTS



RESULT:

Thus the Android Application using Menu with intents has been executed successfully.

Ex.No.10**FILES****AIM:**

To Develop a android Application using Files.

PROCEDURE:

Step 1 : Start the process.

Step 2: Create a new Project by opening the Android Studio and then click on File --> new -->new project.

Step 3: Select the basic activity and choose next option.

Step 4: Change the Application name and process press finish button.

Step 5: Design the Layout by Selecting Content_Main.xml

Step 6: Place the appropriate controls for Developing the Applications

Step 7: Change the constraints and required properties for the controls.

S.No.	Control	Control Values	Properties & values
1	Text	Edit Text ()	ID (user input)
2	Button	Button (Write)	ID (write_button)
3	Button	Button (Read)	ID (read_button)
4	Text	Text View (File Content)	ID (TextView2)
5	Text	Text View ()	ID (Content1)

Step 8: Enter Java coding for the Application by select MainActivity.java

Step 9: Stop the process

PROGRAM:**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">
```

```
<EditText  
android:id="@+id/userInput"  
android:layout_width="wrap_content"  
android:layout_height="wrap_content"  
android:ems="10"  
android:inputType="textPersonName"  
app:layout_constraintBottom_toBottomOf="parent"  
app:layout_constraintEnd_toEndOf="parent"  
app:layout_constraintStart_toStartOf="parent"  
app:layout_constraintTop_toTopOf="parent"  
app:layout_constraintVertical_bias="0.122" />
```

```
<Button  
android:id="@+id/write_button"  
android:layout_width="wrap_content"  
android:layout_height="wrap_content"  
android:text="WRITE"  
app:layout_constraintBottom_toBottomOf="parent"  
app:layout_constraintEnd_toEndOf="parent"  
app:layout_constraintHorizontal_bias="0.498"  
app:layout_constraintStart_toStartOf="parent"  
app:layout_constraintTop_toBottomOf="@+id/userInput"  
app:layout_constraintVertical_bias="0.12" />
```

```
<Button  
android:id="@+id/read_button"  
android:layout_width="wrap_content"  
android:layout_height="wrap_content"  
android:text="READ"  
app:layout_constraintBottom_toBottomOf="parent"  
app:layout_constraintEnd_toEndOf="parent"  
app:layout_constraintHorizontal_bias="0.498"  
app:layout_constraintStart_toStartOf="parent"  
app:layout_constraintTop_toBottomOf="@+id/write_button"  
app:layout_constraintVertical_bias="0.111" />
```

```
<TextView  
android:id="@+id/content1"  
android:layout_width="wrap_content"  
android:layout_height="wrap_content"  
app:layout_constraintBottom_toBottomOf="parent"  
app:layout_constraintEnd_toEndOf="parent"  
app:layout_constraintHorizontal_bias="0.498"  
app:layout_constraintStart_toStartOf="parent"
```

```
app:layout_constraintTop_toBottomOf="@+id/read_button"
app:layout_constraintVertical_bias="0.36"
tools:text="" />
```

```
<TextView
android:id="@+id/textView3"
android:layout_width="wrap_content"
android:layout_height="wrap_content"
android:text="File content"
app:layout_constraintBottom_toBottomOf="parent"
app:layout_constraintEnd_toEndOf="parent"
app:layout_constraintStart_toStartOf="parent"
app:layout_constraintTop_toBottomOf="@+id/read_button"
app:layout_constraintVertical_bias="0.072" />
```

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

Main Acitivity.java

```
package com.ex1.myapplication10;
```

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

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

```
    Button read,write;
    EditText userInput;
    TextView filecontent;
    private String filename="demoFile.txt";
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_main);
        read=findViewById(R.id.read_button);
        write=findViewById(R.id.write_button);
        userInput=findViewById(R.id.userInput);
        filecontent=findViewById(R.id.content1);
        read.setOnClickListener(this);
```

```

write.setOnClickListener(this);
    }
public void printMessage(String m){
Toast.makeText(this,m, Toast.LENGTH_LONG).show();
    }
public void onClick(View view) {
    Button b = (Button) view;
    String b_text = b.getText().toString();
switch (b_text.toLowerCase()) {
case "write": {
writeData();
break;

    }
case "read": {
readData();
break;

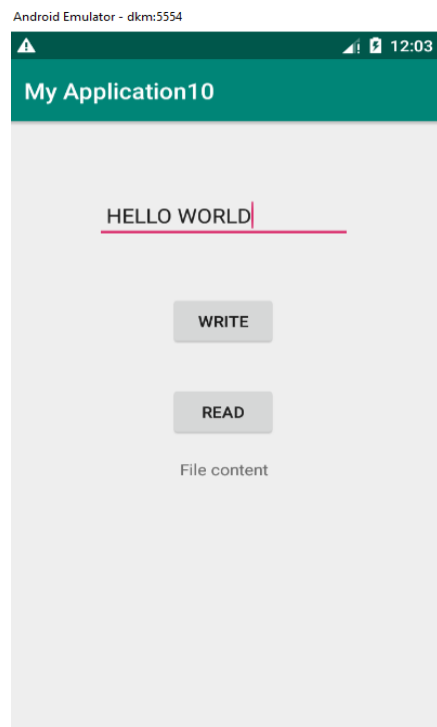
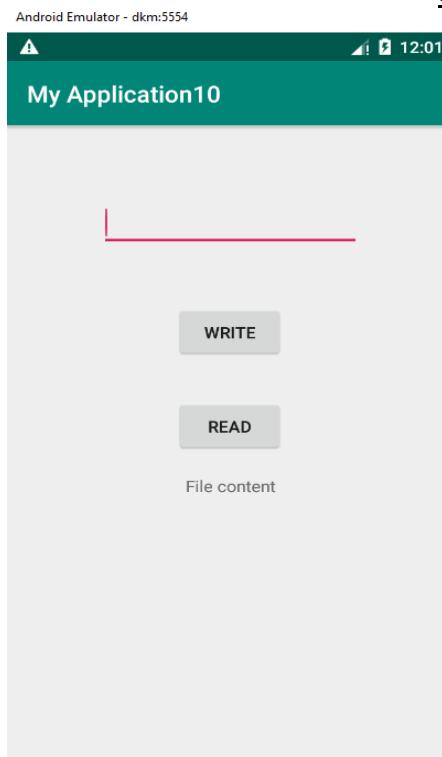
    }
    }
}
private void writeData(){
try{
FileOutputStream fos=openFileOutput(filename, Context.MODE_PRIVATE);
    String data=userInput.getText().toString();
fos.write(data.getBytes());
fos.flush();
fos.close();
    }catch (IOException e){
e.printStackTrace();
    }
userInput.setText(" ");
printMessage("Writing to file"+filename+"completed...");
}
private void readData(){
try {
FileInputStream fin = openFileInput(filename);
inta;
StringBuilder temp = new StringBuilder();
while ((a = fin.read()) != -1) {
temp.append((char) a);
    }
filecontent.setText(temp.toString());
fin.close();
    }catch (IOException e)
    {
e.printStackTrace();
    }
printMessage("reading to file"+filename+"completed....");
}

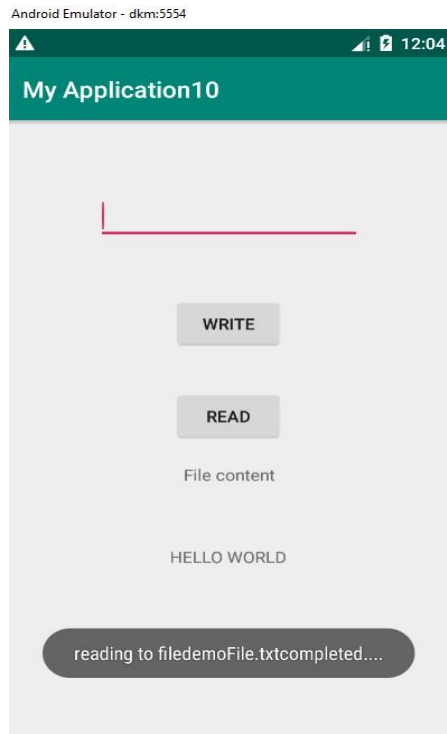
```

```
}  
}
```

OUTPUT:

10. WORKING WITH FILES





RESULT:

Thus the Android Application using Files has been executed successfully.