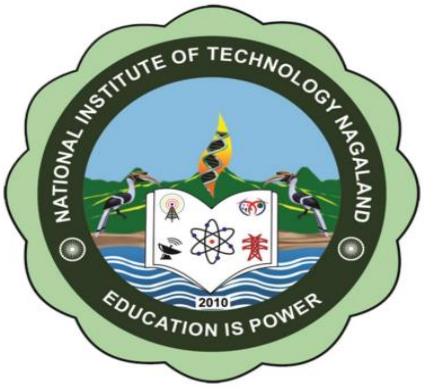


Mobile and Pervasive Computing Laboratory Record (CS403)

Submitted by

Name :- SAKSHAM SINGHAL

Registration no. :- 2018105184

Semester :- 7th

Department :- CSE

Course In-Charge:

Dr. Lithungo Murry

Ms. Bidyarani

Mr. Yuga Sai

Index

1.	Write a simple android program to create a Form for new student enrolled in your institute.
2.	Simple android program to mimic the function of calculator.
3.	Create a login form with landing page.
4.	To create stopwatch app in android studio
5.	To make an alarm clock app
6.	To make image viewer app in android studio

ANDROID DEVELOPMENT

ASSIGNMENT – 1

Write a simple android program to create a Form for new student enrolled in your institute, which must include some basic information of the student like – Student ID, student name (First name, Last name), F/Name, Sex, Email, D.O.B, Address (P/C).

Algorithm:

1. Create the new project-> select empty activity.
2. Go to design tab and add 'Text' widget from the palette options for the form fields according to the requirement, for example, plaintext for Name, Father's Name and id; Postal Address for address; E-mail for E-mail, Password for Password and Date for Date of Birth fields respectively.
3. Add radio button group from Palette->Buttons->RadioGroup.
4. Add individual selection radio button for male and female selection in the radio button group.
5. Edit the text attributes of the radio button to male and female respectively.
6. Add simple button from Buttons from palette for submit button.
7. Now, we have added all the required field and buttons to our design.
8. Add required margin between different fields like top, left, right and bottom to the fields.
9. Match the width of text fields to the parent by selecting 'layout_width' to 'match_parent' in xml.
10. Change the 'hint' attributed of all the form field one by one according to their purpose, for example, Id, Name, Father's Name, Address, E-mail, Password, Date of Birth.
11. Edit the text attribute of submit button at last to 'Submit'. Set it's allcaps attributes in xml to 'False'.
12. Change the background color attribute of the app in xml to 'grey' adding grey color hex code in the colors.xml file.
13. Add some padding to each form field say 15dp.
14. Set 'textColorHint' attribute inn xml for form field to 'dimgray' color whose hex codes are set in the colors.xml file.
15. Set 'textColor' attribute for the form fields to black in the xml code.
16. Set ButtonTint attribute for button selection color of the button for male and female to 'dimgray'.
17. Hence, our app design is completed.
18. Now, enable usb debugging on mobile phone after enabling developer mode on android smartphone.
19. Connect the phone through a usb cable to pc. Select the connected device in android studio screen.
20. Run the created app by playing the 'Run' button on top right or 'Shift+f10' shortcut key.

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"
    android:background="@color/grey"
    android:padding="15dp"
    tools:context=".MainActivity">

    <EditText
        android:id="@+id/editTextTextPersonName"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:layout_marginTop="20dp"
        android:ems="10"
        android:hint="id"
        android:textColor="@color/black"
        android:textColorHint="@color/dimgray"
        android:inputType="textPersonName"
        app:layout_constraintEnd_toEndOf="parent"
        app:layout_constraintHorizontal_bias="0.502"
        app:layout_constraintStart_toStartOf="parent"
        app:layout_constraintTop_toTopOf="parent" />

    <EditText
        android:id="@+id/editTextTextPersonName2"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:layout_marginTop="15dp"
        android:ems="10"
        android:hint="Name"
        android:textColor="@color/black"
        android:textColorHint="@color/dimgray"
```

```
        android:inputType="textPersonName"
        app:layout_constraintEnd_toEndOf="parent"
        app:layout_constraintHorizontal_bias="0.502"
        app:layout_constraintStart_toStartOf="parent"
        app:layout_constraintTop_toBottomOf="@+id/editTextTextPersonName" />

    <EditText
        android:id="@+id/editTextTextPersonName3"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:layout_marginTop="15dp"
        android:ems="10"
        android:hint="Father's Name"
        android:textColor="@color/black"
        android:textColorHint="@color/dimgray"
        android:inputType="textPersonName"
        app:layout_constraintEnd_toEndOf="parent"
        app:layout_constraintHorizontal_bias="0.502"
        app:layout_constraintStart_toStartOf="parent"
        app:layout_constraintTop_toBottomOf="@+id/editTextTextPersonName2" />

    <EditText
        android:id="@+id/editTextNumberPassword2"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:layout_marginTop="15dp"
        android:ems="10"
        android:hint="Password"
        android:inputType="numberPassword"
        android:textColor="@color/black"
```

```
        android:textColorHint="@color/dimgray"
        app:layout_constraintEnd_toEndOf="parent"
        app:layout_constraintHorizontal_bias="0.0"
        app:layout_constraintStart_toStartOf="parent"
        app:layout_constraintTop_toBottomOf="@+id/editTextTextEmailAddress2" />

    <EditText
        android:id="@+id/editTextDate"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:layout_marginTop="15dp"
        android:ems="10"
        android:hint="Date of Birth"
        android:inputType="date"
        android:textColor="@color/black"
        android:textColorHint="@color/dimgray"
        app:layout_constraintEnd_toEndOf="parent"
        app:layout_constraintHorizontal_bias="0.0"
        app:layout_constraintStart_toStartOf="parent"
        app:layout_constraintTop_toBottomOf="@+id/editTextNumberPassword2" />

    <Button
        android:id="@+id/button"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_marginLeft="20dp"
        android:layout_marginTop="15dp"
        android:backgroundTint="@color/dark_grey"
        android:text="Submit"
        android:textColor="@color/black"
```

```
        android:textAllCaps="false"
        app:layout_constraintStart_toStartOf="parent"
        app:layout_constraintTop_toBottomOf="@+id/editTextDate" />

    <EditText
        android:id="@+id/editTextTextEmailAddress2"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:layout_marginTop="15dp"
        android:ems="10"
        android:hint="E-mail"
        android:inputType="textEmailAddress"
        android:textColor="@color/black"
        android:textColorHint="@color/dimgray"
        app:layout_constraintEnd_toEndOf="parent"
        app:layout_constraintHorizontal_bias="0.0"
        app:layout_constraintStart_toStartOf="parent"
        app:layout_constraintTop_toBottomOf="@+id/editTextTextPostalAddress" />

    <EditText
        android:id="@+id/editTextTextPostalAddress"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:layout_marginTop="15dp"
        android:ems="10"
        android:hint="Address"
        android:inputType="textPostalAddress"
        android:textColor="@color/black"
        android:textColorHint="@color/dimgray"
        app:layout_constraintEnd_toEndOf="parent"
```

```
        app:layout_constraintHorizontal_bias="1.0"
        app:layout_constraintStart_toStartOf="parent"
        app:layout_constraintTop_toBottomOf="@+id/radioGroup" />

    <RadioGroup
        android:id="@+id/radioGroup"
        android:layout_width="398dp"
        android:layout_height="36dp"
        android:layout_marginTop="15dp"
        android:orientation="horizontal"
        android:gravity="center"
        app:layout_constraintEnd_toEndOf="parent"
        app:layout_constraintHorizontal_bias="0.555"
        app:layout_constraintStart_toStartOf="parent"
        app:layout_constraintTop_toBottomOf="@+id/editTextTextPersonName3">

        <RadioButton
            android:id="@+id/radioMale"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:checked="false"
            android:text="  Male"
            android:padding="5dp"
            android:buttonTint="@color/dimgray"
            android:textColor="@color/black" />

        <RadioButton
            android:id="@+id/radioFemale"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
```

```
            android:checked="false"

            android:text="Female"

            android:layout_marginLeft="10dp"

            android:padding="5dp"

            android:buttonTint="@color/dimgray"

            android:textColor="@color/black" />

    </RadioGroup>




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

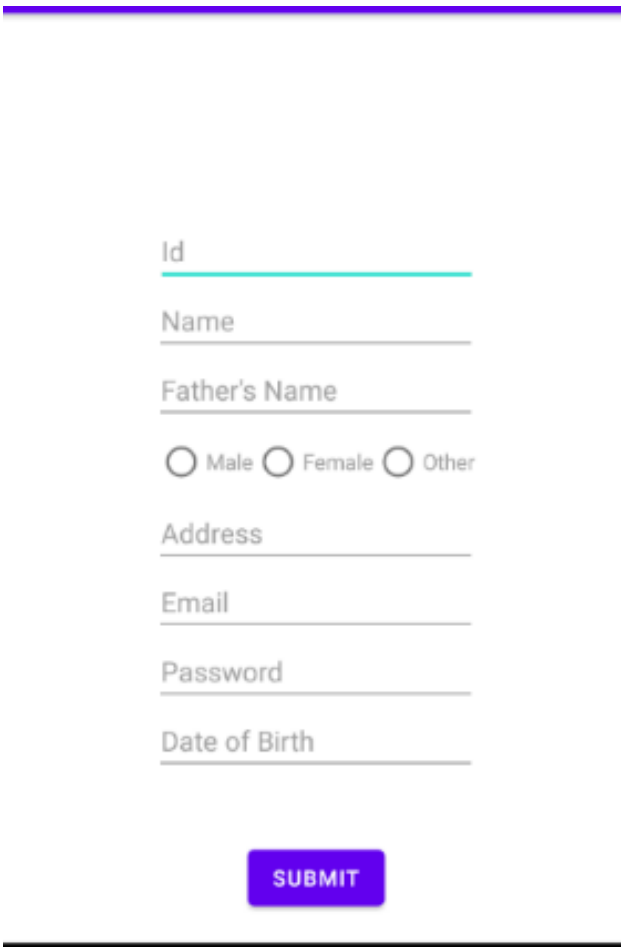
colors.xml:

```
<?xml version="1.0" encoding="utf-8"?>
<resources>
    <color name="purple_200">#FFBB86FC</color>
    <color name="purple_500">#FF6200EE</color>
    <color name="purple_700">#FF3700B3</color>
    <color name="teal_200">#FF03DAC5</color>
    <color name="teal_700">#FF018786</color>
    <color name="black">#FF000000</color>
    <color name="white">#FFFFFFFF</color>
    <color name="grey">    #DCDCDC</color>
    <color name="dark_grey">#A9A9A9</color>
    <color name="dimgray">#696969</color>
</resources>
```

MainActivity.java:

```
package com.example.lab1_2018105181;

import androidx.appcompat.app.AppCompatActivity;

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

public class MainActivity extends AppCompatActivity {

    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_main);
        getSupportActionBar().setBackgroundDrawable(new
ColorDrawable(Color.parseColor("#4169e1")));
    }
}
```

Output :-



ANDROID DEVELOPMENT

ASSIGNMENT – 2

Write a simple android program to mimic the function of a calculator, which must include some basic arithmetic operation like addition, subtraction, multiplication and division.

Algorithm:-

1. Open Android Studio and create New Android Studio Project .Give Application Name ***calculator*** and leave other fields as it is, then click NEXT.
2. Select the Empty Activity and click NEXT.
3. Now we have **activity_main.xml** in which we have to make the layout of our app
4. Select Design->Pallate->Layouts->TableLayout. And customise to make a Tablerow,Tablecol of 5 and 4 respectively i.e 5x4.
5. In each row add four button of format :-
    a. <Button
    b. android:id="@+id/<***button id***>" ----------> Button Id
    c. android:layout_width="30pt"
    d. android:layout_height="30pt"
    e. android:layout_marginRight="1pt"
    f. android:onClick="numberEvent" ----------> Onclick Event
    g. android:layout_weight="1"
    h. android:text=<***Button name***></Button> ------> Button Name
6. From above we have 20 button which we have to edit the {id,onClickEvent,text} field of those buttons according to our requirement and then out layout is prepared .
7. Now we have to add functionality to our app , this we will do in MainActivity.java file of our app.
8. In MainActivity.java we declare
    i. String op : For storing the operator selected by user.
    ii. String oldNumber: For storing the previous operand.
    iii. Boolean isNewOperator : For checking if new operator is selected or not, which has default value of true.
    iv. EditText ed1 : For storing the input from the editText field of app.

9. Open MainActivity.java and we have to implement five onClick event namely:
    i. numberEvent: This event is implemented to read the number from [0-9] and store it in variable ***number.***
    ii. OperatorEvent: This event is implemented to read the operator selected in the app like {+,-,*,/,mod}.
    iii. equalEvent: This event is implemented to calculate the arithemetic operation and showing the result .
    iv. clearEvent: Used to clear the editText field of app.
    v. percentageEvent: Used to calculate the percentage of the number entered.

10. Implementing the numberEvent :
    a. We declare the number variable for storing the input then we the switch statement to assign the number according to id . For example if selected button 7 then our Key will be ''but7'' and we can assign the value of 7 to number.
11. Implementing the operatorEvent:
    a. Firstly we assign the newOperator Value to true and then we save the previous number in oldNumber variable .

b. Then, we switch the id according to {+,-,*,/} and assign the op variable the symbol of the button .

12. Implementing the equalEvent:
    a. Firstly we store the newNumber in variable newNumber
    b. Then,we switch on basis of op{operator} and do the arithemetic operation and store it in the result variable.
    c. Finally , we use setText() to print the result in the editText window.
13. Implementing the clearEvent:
    a. Simply we clear the Screen by using the setText("0") method.
    b. And assigning the newOperator to true;
14. Implementing the percentEvent:
    a. Simply cal the percentage and print it.
15. Finally our app is completed.

XML FILE :

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

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

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

    android:layout_width="match_parent"

    android:layout_height="match_parent"

    android:orientation="vertical"

    tools:context=".MainActivity">

    <EditText

        android:id="@+id/editText"

        android:layout_width="match_parent"

        android:layout_height="wrap_content"

        android:layout_weight="1"

        android:ems="10"

        android:layout_marginBottom="1pt"

        android:gravity="right|center"

        android:inputType="textPersonName"

        android:text="0"

        android:textColor="@color/black"

        android:textSize="40pt" />
```

```
<TableLayout
    android:layout_width="match_parent"
    android:layout_height="wrap_content">
    <TableRow
        android:layout_marginBottom="1pt"
        android:layout_width="match_parent"
        android:layout_height="match_parent">
        <Button
            android:id="@+id/bu7"
            android:layout_width="30pt"
            android:layout_height="30pt"
            android:layout_marginRight="1pt"
            android:onClick="numberEvent"
            android:layout_weight="1"
            android:text="7"></Button>
        <Button
            android:id="@+id/bu8"
            android:layout_width="30pt"
            android:layout_height="30pt"
            android:layout_marginRight="1pt"
            android:onClick="numberEvent"
            android:layout_weight="1"
            android:text="8"></Button>
        <Button
            android:id="@+id/bu9"
            android:layout_width="30pt"
            android:layout_height="30pt"
            android:onClick="numberEvent"
            android:layout_marginRight="1pt"
            android:layout_weight="1"
```

```
            android:text="9"></Button>
        <Button
            android:id="@+id/buDivide"
            android:layout_width="30pt"
            android:layout_height="30pt"
            android:onClick="operatorEvent"
            android:layout_marginRight="1pt"
            android:layout_weight="1"
            android:text="/"
            android:textSize="18sp"
            android:textColor="@color/white"></Button>
    </TableRow>
    <TableRow
        android:layout_marginBottom="1pt"
        android:layout_width="match_parent"
        android:layout_height="match_parent">
        <Button
            android:id="@+id/bu4"
            android:layout_width="30pt"
            android:layout_height="30pt"
            android:layout_marginRight="1pt"
            android:onClick="numberEvent"
            android:layout_weight="1"
            android:text="4"></Button>
        <Button
            android:id="@+id/bu5"
            android:layout_width="30pt"
            android:layout_height="30pt"
            android:onClick="numberEvent"
            android:layout_marginRight="1pt"
```

```
            android:layout_weight="1"
            android:text="5"></Button>
        <Button
            android:id="@+id/bu6"
            android:layout_width="30pt"
            android:layout_height="30pt"
            android:onClick="numberEvent"
            android:layout_marginRight="1pt"
            android:layout_weight="1"
            android:text="6"></Button>
        <Button
            android:id="@+id/buMultiply"
            android:layout_width="30pt"
            android:layout_height="30pt"
            android:layout_marginRight="1pt"
            android:onClick="operatorEvent"
            android:layout_weight="1"
            android:text="*"
            android:textSize="18sp"
            android:textColor="@color/white"></Button>
    </TableRow>
    <TableRow
        android:layout_marginBottom="1pt"
        android:layout_width="match_parent"
        android:layout_height="match_parent">
        <Button
            android:id="@+id/bu1"
            android:layout_width="30pt"
            android:layout_height="30pt"
            android:onClick="numberEvent"
```

```
            android:layout_marginRight="1pt"
            android:layout_weight="1"
            android:text="1"></Button>
        <Button
            android:id="@+id/bu2"
            android:layout_width="30pt"
            android:onClick="numberEvent"
            android:layout_height="30pt"
            android:layout_marginRight="1pt"
            android:layout_weight="1"
            android:text="2"></Button>
        <Button
            android:id="@+id/bu3"
            android:layout_width="30pt"
            android:onClick="numberEvent"
            android:layout_height="30pt"
            android:layout_marginRight="1pt"
            android:layout_weight="1"
            android:text="3"></Button>
        <Button
            android:id="@+id/buSubtract"
            android:layout_width="30pt"
            android:layout_height="30pt"
            android:layout_marginRight="1pt"
            android:onClick="operatorEvent"
            android:layout_weight="1"
            android:textSize="18sp"
            android:text="-"></Button>

    </TableRow>
```

```
<TableRow
    android:layout_marginBottom="1pt"
    android:layout_width="match_parent"
    android:layout_height="match_parent">
    <Button
        android:id="@+id/buDecimal"
        android:layout_width="30pt"
        android:layout_height="30pt"
        android:onClick="numberEvent"
        android:layout_marginRight="1pt"
        android:layout_weight="1"
        android:text="."></Button>
    <Button
        android:id="@+id/bu0"
        android:layout_width="30pt"
        android:onClick="numberEvent"
        android:layout_height="30pt"
        android:layout_marginRight="1pt"
        android:layout_weight="1"
        android:text="0"></Button>
    <Button
        android:id="@+id/buEqual"
        android:layout_width="30pt"
        android:layout_height="30pt"
        android:layout_marginRight="1pt"
        android:layout_weight="1"
        android:onClick="equalEvent"
        android:textSize="18sp"
        android:text="="></Button>
    <Button
```

```
            android:id="@+id/buAdd"
            android:layout_width="30pt"
            android:layout_height="30pt"
            android:layout_marginRight="1pt"
            android:onClick="operatorEvent"
            android:layout_weight="1"
            android:text="+"
            android:textSize="18sp"
            android:textColor="@color/white"></Button>
    </TableRow>
    <TableRow
        android:layout_marginBottom="1pt"
        android:layout_width="match_parent"
        android:layout_height="match_parent">
        <Button
            android:id="@+id/buPercentage"
            android:layout_width="30pt"
            android:layout_height="30pt"
            android:onClick="percentEvent"
            android:layout_marginRight="1pt"
            android:layout_weight="1"
            android:text="%"></Button>
        <Button
            android:id="@+id/buMod"
            android:layout_width="30pt"
            android:layout_height="30pt"
            android:layout_marginRight="1pt"
            android:layout_weight="1"
            android:onClick="operatorEvent"
            android:text="Mod"></Button>
```

```
            <Button
                android:id="@+id/buPower"
                android:layout_width="30pt"
                android:layout_height="30pt"
                android:layout_marginRight="1pt"
                android:layout_weight="1"
                android:onClick="operatorEvent"
                android:textSize="18sp"
                android:text="^"></Button>
            <Button
                android:id="@+id/buClear"
                android:layout_width="30pt"
                android:layout_height="30pt"
                android:layout_marginRight="1pt"
                android:layout_weight="1"
                android:onClick="clearEvent"
                android:text="CE"
                android:textSize="18sp"
                android:textColor="@color/white"></Button>
        </TableRow>
    </TableLayout>
</LinearLayout>
```

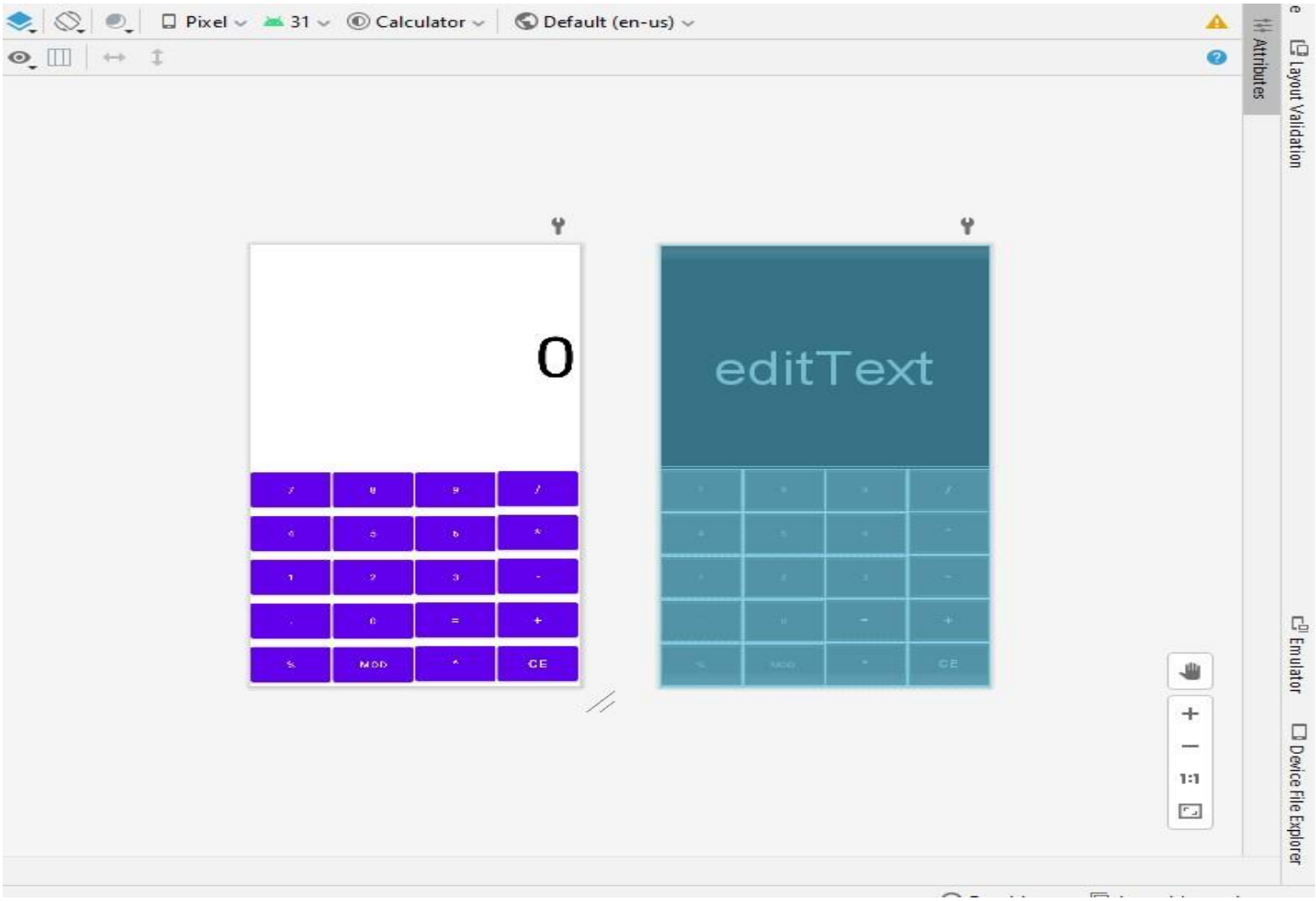
MainActivity.java :-

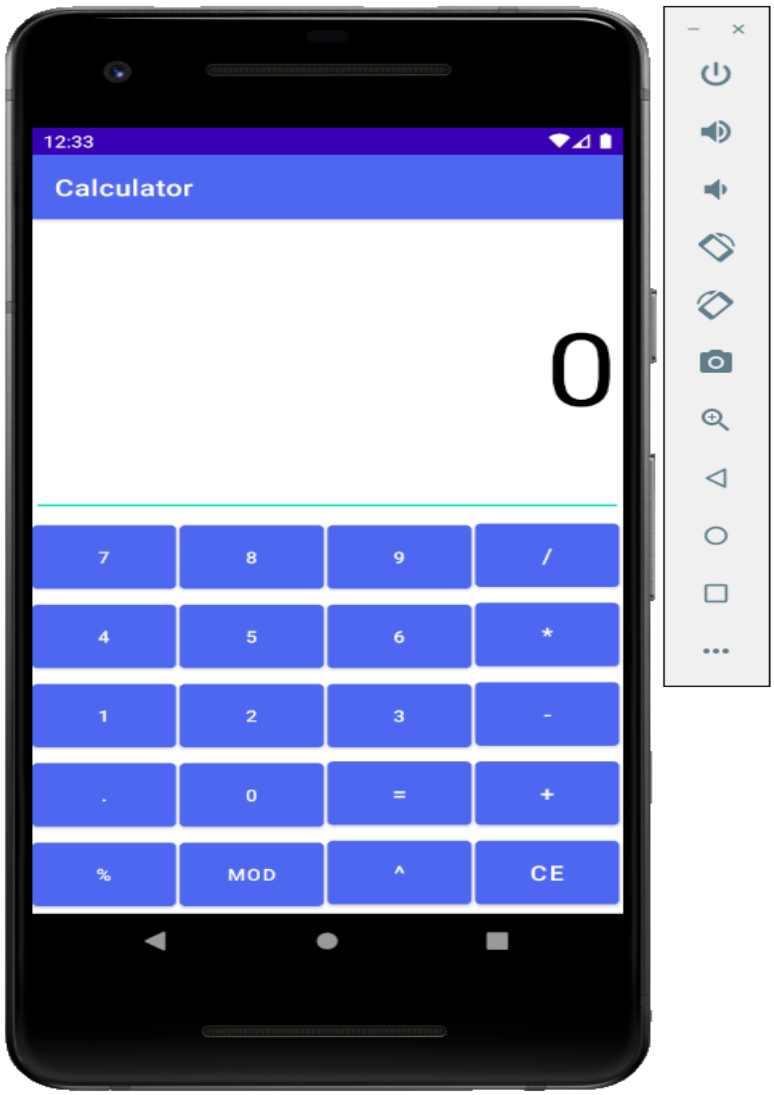
```
package com.example.calculator;
import androidx.appcompat.app.AppCompatActivity;
import android.os.Bundle;
import android.view.View;
import android.widget.EditText;
public class MainActivity extends AppCompatActivity

    String op = "";
    String oldNumber = "";
    boolean isNewOperator = true;
    EditText ed1;
    @Override
    protected void onCreate(Bundle savedInstanceState) {
```

```
    super.onCreate(savedInstanceState);
    setContentView(R.layout.activity_main);
    ed1 = findViewById(R.id.editText);
}
public void numberEvent(View view) {
    if(isNewOperator)
        ed1.setText("");
    isNewOperator = false;
    String number = ed1.getText().toString();
    switch(view.getId()){
        case R.id.bu0:
            number += "0";
            break;

        case R.id.bu1:
            number += "1";
            break;

        case R.id.bu2:
            number += "2";
            break;

        case R.id.bu3:
            number += "3";
            break;

        case R.id.bu4:
            number += "4";
            break;

        case R.id.bu5:
            number += "5";
            break;

        case R.id.bu6:
            number += "6";
            break;

        case R.id.bu7:
            number += "7";
            break;

        case R.id.bu8:
            number += "8";
            break;

        case R.id.bu9:
            number += "9";
            break;

        case R.id.buDecimal:
            number += ".";
            break;

        default:
            throw new IllegalStateException("Unexpected value: " + view.getId());
    }
    ed1.setText(number);
}
public void operatorEvent(View view) {
    isNewOperator = true;
```

```
        oldNumber = ed1.getText().toString();
        switch (view.getId()){
            case R.id.buAdd: op = "+"; break;
            case R.id.buSubtract: op = "-"; break;
            case R.id.buMultiply: op = "*"; break;
            case R.id.buDivide: op = "/"; break;
            case R.id.buPower: op = "^"; break;
            case R.id.buMod: op = "%"; break;
        }
    }
    public void equalEvent(View view) {
        String newNumber = ed1.getText().toString();
        double result = 0.0;
        switch (op){
            case "+":
                result = Double.parseDouble(oldNumber) + Double.parseDouble(newNumber);
                break;
            case "-":
                result = Double.parseDouble(oldNumber) - Double.parseDouble(newNumber);
                break;
            case "*":
                result = Double.parseDouble(oldNumber) * Double.parseDouble(newNumber);
                break;

            case "/":
                result = Double.parseDouble(oldNumber) / Double.parseDouble(newNumber);
                break;

            case "^":
                result =
Math.pow(Double.parseDouble(oldNumber),Double.parseDouble(newNumber));
                break;

            case "%":
                result = Double.parseDouble(oldNumber) % Double.parseDouble(newNumber);
                break;

            default:
                throw new IllegalStateException("Unexpected value: " + op);
        }
        ed1.setText(result + "");
    }
    public void clearEvent(View view) {
        ed1.setText("0");
        isNewOperator = true;
    }
    public void percentEvent(View view) {
        double no = Double.parseDouble(ed1.getText().toString()) / 100;
        ed1.setText(no+"");
        isNewOperator = true;
    }
}
```

OUTPUT :-


Pixel
31
Calculator
Default (en-us)
Attributes
Layout Validation
Emulator
Device File Explorer
0
editText
MOD
CE
1:1



ANDROID DEVELOPMENT

Assignment - 3

Aim : To create a login form with two activity.

Algorithm:

1. Create a new project of loginForm with an empty activity.
2. Add textview for login text, username text and password text.
3. Add E-mail EditText and Password EditText to the layout. Set text hint attribute to “enter email” and “enter password” respectively.
4. Add button to the layout and change its text to “Submit”.
5. Set the different attributes for the added items to the layout to make it look good. For example, margin, padding, constraints, layout width, layout height in their xml file.
6. Create an onclick listener for submit button which when clicked take us to the new activity page.
7. In MainActivity.java get the response from the username and password form field and check if they are not empty. If empty then toast an empty message to the screen else create an instance of new activity and start the activity.
8. Create an empty activity called MainActivity2 which will be opened when submit button is clicked.
9. Add plain text on the main_activity2.xml using the design tab.
10. Now, our app is created, test it on the emulator or android device.

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/textView"
      android:layout_width="wrap_content"
      android:layout_height="wrap_content"
      android:text="@string/login"
      android:layout_marginTop="20dp"
      android:textSize="35sp"
      app:layout_constraintEnd_toEndOf="parent"
      app:layout_constraintStart_toStartOf="parent"
      app:layout_constraintTop_toTopOf="parent" />
   <TextView
      android:id="@+id/textView2"
      android:layout_width="wrap_content"
      android:layout_height="wrap_content"
      android:layout_marginTop="55dp"
      android:layout_marginLeft="25dp"
      android:text="@string/username"
      android:textSize="28sp"
      app:layout_constraintStart_toStartOf="parent"
      app:layout_constraintTop_toBottomOf="@+id/textView" />
```

```
<EditText
   android:id="@+id/inuser"
   android:layout_width="wrap_content"
   android:layout_height="wrap_content"
   android:layout_marginTop="12dp"
   android:layout_marginLeft="25dp"
   android:ems="10"
   android:hint="enter email"
   android:inputType="textEmailAddress"
   android:padding="10dp"
   app:layout_constraintStart_toStartOf="parent"
   app:layout_constraintTop_toBottomOf="@+id/textView2" />
<TextView
   android:id="@+id/textView3"
   android:layout_width="wrap_content"
   android:layout_height="wrap_content"
   android:layout_marginTop="20dp"
   android:layout_marginLeft="25dp"
   android:text="@string/pass"
   android:textSize="28sp"
   app:layout_constraintStart_toStartOf="parent"
   app:layout_constraintTop_toBottomOf="@+id/inuser" />
<EditText
   android:id="@+id/inpass"
   android:layout_width="wrap_content"
   android:layout_height="wrap_content"
   android:layout_marginTop="10dp"
   android:layout_marginLeft="25dp"
   android:ems="10"
   android:hint="enter password"
   android:inputType="textPassword"
   android:padding="10dp"
   app:layout_constraintStart_toStartOf="parent"
   app:layout_constraintTop_toBottomOf="@+id/textView3" />


<LinearLayout
   android:layout_width="match_parent"
   android:layout_height="wrap_content"
   android:orientation="horizontal"
   android:layout_centerHorizontal="true"
   android:gravity="center_horizontal"
   android:layout_marginTop="30dp"
   app:layout_constraintStart_toStartOf="parent"
   app:layout_constraintTop_toBottomOf="@+id/inpass">
   <Button
      android:id="@+id/submit"
```

```
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:onClick="changeactivity"
        android:text="@string/submit"
        android:textAllCaps="false" />
    </LinearLayout>
</androidx.constraintlayout.widget.ConstraintLayout>
```

activity_main2.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=".MainActivity2">

    <EditText
        android:id="@+id/editTextTextPersonName"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:ems="10"
        android:inputType="textPersonName"
        android:text="@string/act2"
        android:textSize="30sp"
        android:gravity="center"
        app:layout_constraintBottom_toBottomOf="parent"
        app:layout_constraintEnd_toEndOf="parent"
        app:layout_constraintStart_toStartOf="parent"
        app:layout_constraintTop_toTopOf="parent"
        />
</androidx.constraintlayout.widget.ConstraintLayout>
```

MainActivity.java :-

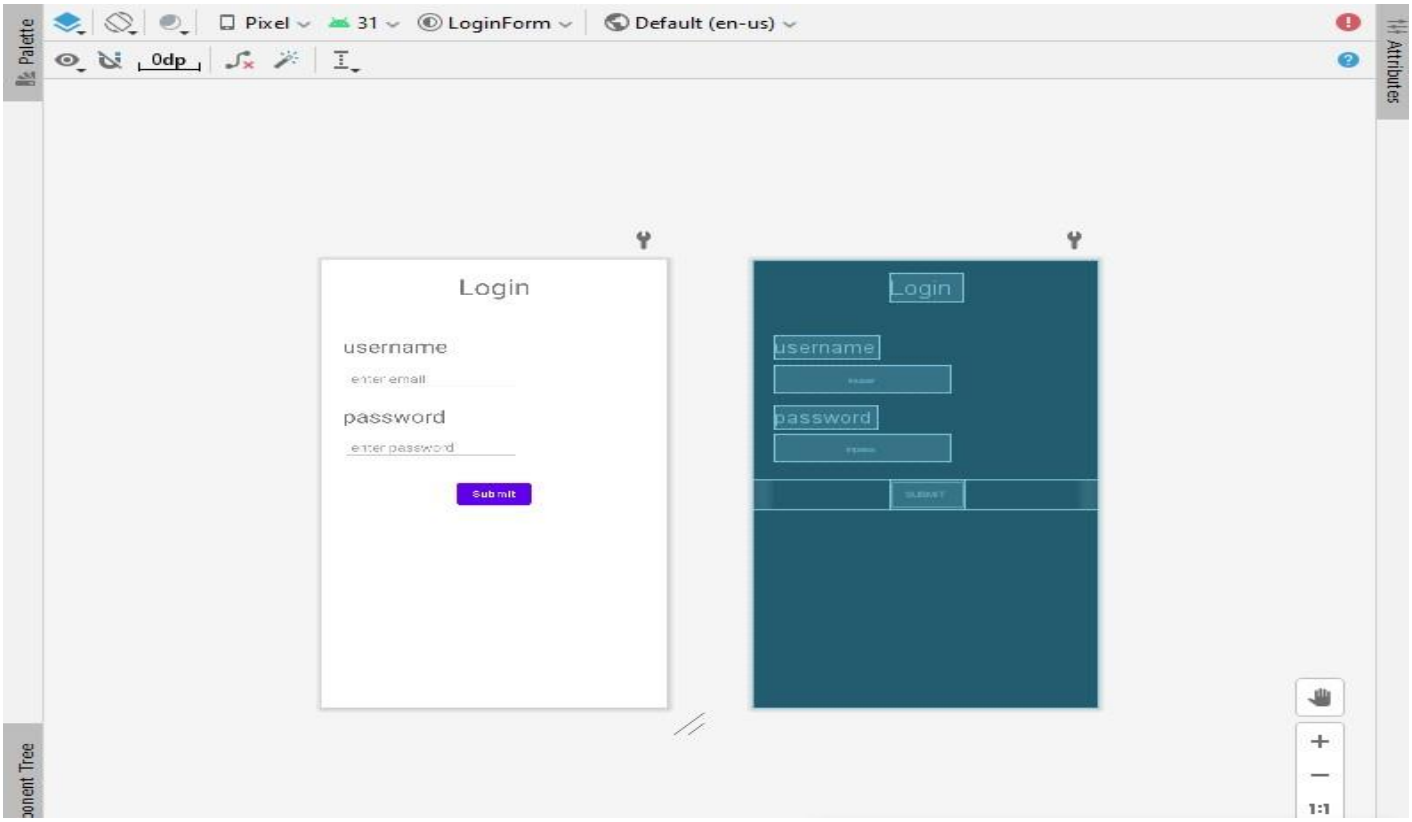
```
<?xml version="1.0" encoding="utf-8"?>
<androidx.constraintlayout.widget.ConstraintLayout
xmlns:android="http://schemas.android.com/apk/res/android"
xmlns:app="http://schemas.android.com/apk/res-auto"
xmlns:tools="http://schemas.android.com/tools"
android:layout_width="match_parent"
android:layout_height="match_parent"     tools:context=".MainActivity2">

    <EditText
        android:id="@+id/editTextTextPersonName"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
```

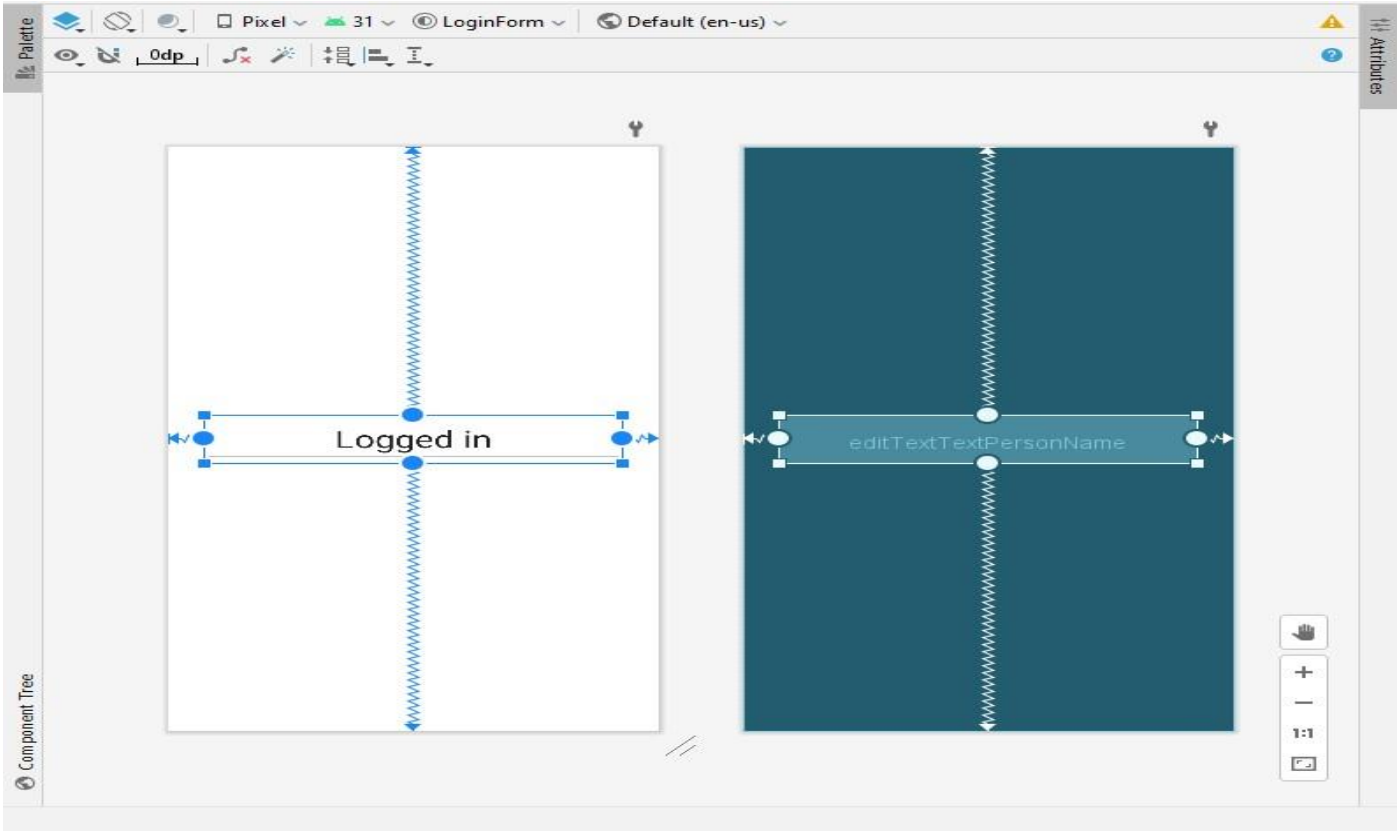
```
        android:ems="10"
        android:inputType="textPersonName"
        android:text="@string/act2"
        android:textSize="30sp"
        android:gravity="center"
        app:layout_constraintBottom_toBottomOf="parent"
        app:layout_constraintEnd_toEndOf="parent"
        app:layout_constraintStart_toStartOf="parent"
        app:layout_constraintTop_toTopOf="parent"
        />
</androidx.constraintlayout.widget.ConstraintLayout>
```

MainActivity2.java :-

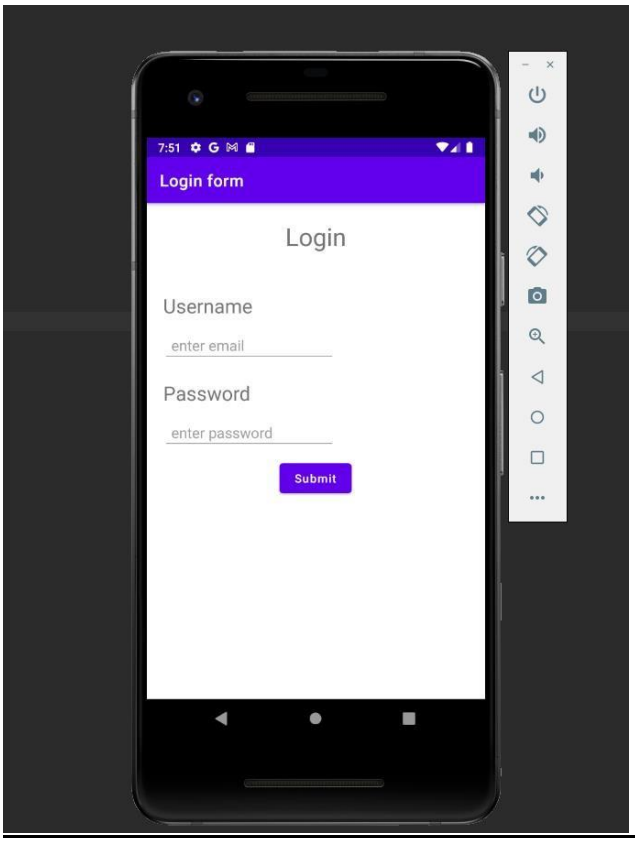
```
package com.example.loginform;

import androidx.appcompat.app.AppCompatActivity;

import android.os.Bundle;

public class MainActivity2 extends AppCompatActivity {

    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_main2);
    }
}
```

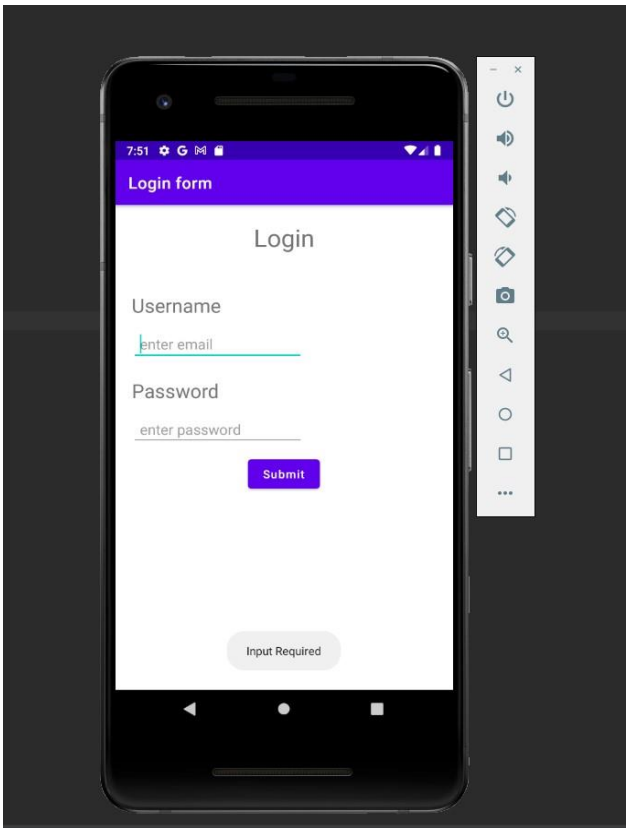
OUTPUT :-


Palette
Pixel
31
LoginForm
Default (en-us)
0dp
Attributes
Login
username
enter email
password
enter password
Submit
Login
username
password
Component Tree
1:1

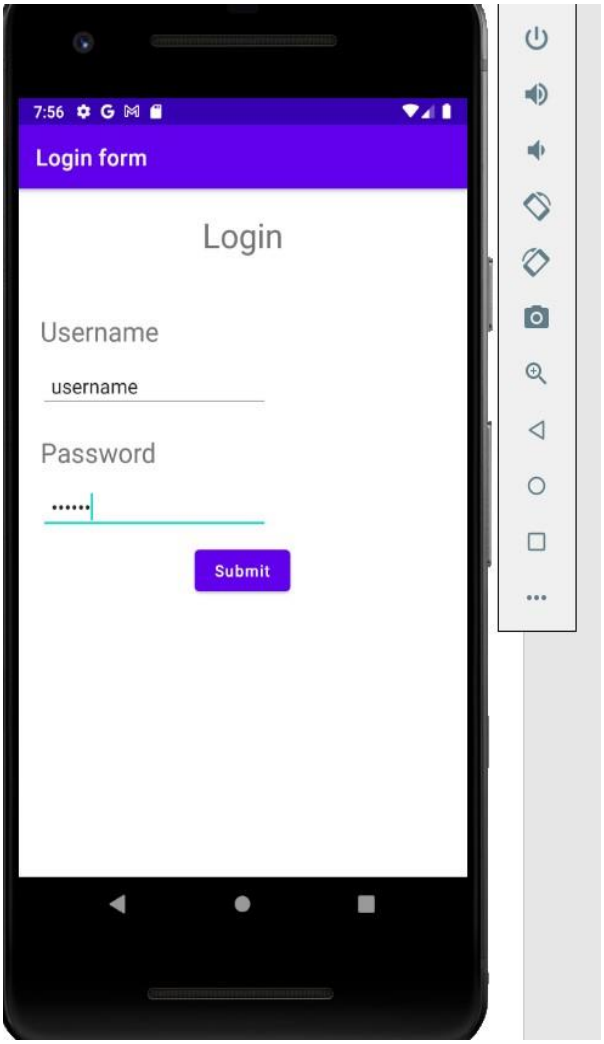

Palette
Pixel
31
LoginForm
Default (en-us)
0dp
Attributes
Logged in
editTextTextPersonName
Component Tree
1:1

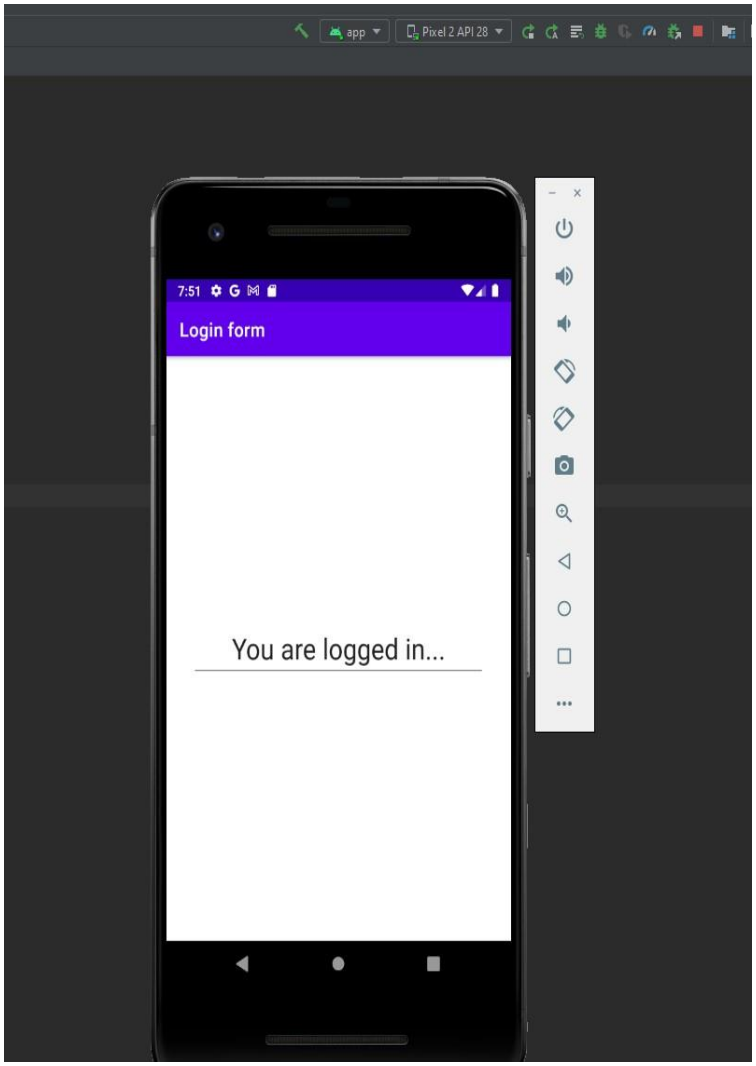
EMULATOR:-





when input details are wrong.





ANDROID LAB

ASSIGNMENT - 4

Aim : Create an alarm clock manager .

Algorithm :

1. Create a project named as Alarm Clock with empty activity.
2. Design the app main activity xml file.
3. Create button for select time, set alarm, cancel alarm with text view for time.
4. Create the on click listener for the button for setting time picker, setting alarm and cancelling alarm write the java code in main activity java file.

5. Create the alarm receiver java file for broadcasting alarm sound and notification on selected time.
6. In the alarm receiver class, create the notification for the selected time with media player object used to create ringtone sound at the selected time for alarming.
7. Set receiver in android manifest file.
8. Create the new destination activity which can be opened when you click on the notification.
9. Now, App is created, run it on the emulator or in mobile device.

XML Code :

activity_main.xml :

```
<?xml version="1.0" encoding="utf-8"?>
<LinearLayout
    xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:tools="http://schemas.android.com/tools"
    xmlns:app="http://schemas.android.com/apk/res-auto"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    android:orientation="vertical"
    tools:context=".MainActivity">
    <TextView
        android:id="@+id/selectedTime"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:layout_marginTop="200dp"
        android:text="05 : 30 PM"
        android:textAlignment="center"
        android:textColor="@color/black"
        android:textSize="46dp" />
    <Button
        android:id="@+id/selectTime"
        android:layout_width="match_parent"
        android:layout_height="70dp"
        android:layout_marginTop="30dp"
        android:layout_marginHorizontal="26dp"
        android:text="Select Time"
        android:textSize="24dp"/>
    <Button
        android:id="@+id/setAlarm"
        android:layout_width="match_parent"
        android:layout_height="70dp"
        android:layout_marginTop="8dp"
        android:layout_marginHorizontal="26dp"
        android:text="Set Alarm"
        android:textSize="24dp"/>
    <Button
        android:id="@+id/cancelAlarm"
        android:layout_width="match_parent"
        android:layout_height="70dp"
        android:layout_marginTop="8dp"
        android:layout_marginHorizontal="26dp"
        android:text="Cancel Alarm"
        android:textSize="24dp"/>
</LinearLayout>
```

Activity_destination.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=".DesstinationActivity">
    <EditText
        android:id="@+id/editTextTextPersonName"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:ems="10"
        android:inputType="textPersonName"
        android:text="Notification Activity"
        app:layout_constraintBottom_toBottomOf="parent"
        app:layout_constraintEnd_toEndOf="parent"
        app:layout_constraintStart_toStartOf="parent"
        app:layout_constraintTop_toTopOf="parent" />
</androidx.constraintlayout.widget.ConstraintLayout>
```

AndroidManifest.xml : - add this line after closing activity tag.

```
<receiver android:name=".AlarmReceiver"/>
```

JAVA CODE :

MainActivity.java :

```
package com.example.alarmclock;
import androidx.appcompat.app.AppCompatActivity;
import androidx.core.app.NotificationManagerCompat;
import android.app.AlarmManager;
import android.app.NotificationChannel;
import android.app.NotificationManager;
import android.app.PendingIntent;
import android.content.Context;
import android.content.Intent;
import android.os.Build;
import android.os.Bundle;
import android.view.View;
import android.widget.Toast;
import com.example.alarmclock.databinding.ActivityMainBinding;
import com.google.android.material.timepicker.MaterialTimePicker;
import com.google.android.material.timepicker.TimeFormat;
import java.util.Calendar;
public class MainActivity extends AppCompatActivity {
    private ActivityMainBinding binding;
    private MaterialTimePicker picker;
    private Calendar calendar;
    private AlarmManager alarmManager;
    private PendingIntent pendingIntent;
    @Override
```

```
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        binding = ActivityMainBinding.inflate(getLayoutInflater());
        setContentView(binding.getRoot());
        createNotificationChannel();
        binding.selectTime.setOnClickListener(new View.OnClickListener() {
            @Override
            public void onClick(View v) {
                showTimePicker();
            }
        });
        binding.setAlarm.setOnClickListener(new View.OnClickListener() {
            @Override
            public void onClick(View v) {
                setAlarm();
            }
        });
        binding.cancelAlarm.setOnClickListener(new View.OnClickListener() {
            @Override
            public void onClick(View v) {
                cancelAlarm();
            }
        });
    }
    private void cancelAlarm() {
        Intent intent = new Intent(this, AlarmReceiver.class);
        pendingIntent = PendingIntent.getBroadcast(this, 0, intent, 0);
        if(alarmManager==null){
            alarmManager = (AlarmManager)
                    getSystemService(Context.ALARM_SERVICE);
        }
        alarmManager.cancel(pendingIntent);
        Toast.makeText(this, "Alarm cancelled...", Toast.LENGTH_SHORT).show();
    }
    private void setAlarm() {
        alarmManager = (AlarmManager) getSystemService(Context.ALARM_SERVICE);
        Intent intent = new Intent(this, AlarmReceiver.class);
        pendingIntent = PendingIntent.getBroadcast(this, 0, intent, 0);
        alarmManager.setInexactRepeating(AlarmManager.RTC_WAKEUP,
calendar.getTimeInMillis(), AlarmManager.INTERVAL_DAY, pendingIntent);
        Toast.makeText(this, "Alarm set successfully...", Toast.LENGTH_SHORT).show();
    }
    private void showTimePicker() {
        picker = new MaterialTimePicker.Builder()
                .setTimeFormat(TimeFormat.CLOCK_12H)
                .setHour(12)
                .setMinute(0)
                .setTitleText("Select Alarm Time")
                .build();
        picker.show(getSupportFragmentManager(), "Experiment");
        picker.addOnPositiveButtonClickListener(new View.OnClickListener() {
            @Override
            public void onClick(View view) {
                if (picker.getHour() > 12) {
                    binding.selectedTime.setText(String.format("%02d", (picker.getHour()
- 12)) + " : " + String.format("%02d", picker.getMinute()) + "PM");
                } else {
                    binding.selectedTime.setText(picker.getHour() + " : " +
picker.getMinute() + " : " + "AM");
                }
                calendar = Calendar.getInstance();
```

```
                calendar.set(Calendar.HOUR_OF_DAY, picker.getHour());
                calendar.set(Calendar.MINUTE, picker.getMinute());
                calendar.set(Calendar.SECOND, 0);
                calendar.set(Calendar.MILLISECOND, 0);
            }
        });
    }
    private void createNotificationChannel() {
        if (Build.VERSION.SDK_INT >= Build.VERSION_CODES.O){
            CharSequence name = "Experiment";
            String description = "Alarm Manager";
            int importance = NotificationManager.IMPORTANCE_HIGH;
            NotificationChannel channel = new NotificationChannel("Experiment", name,
importance);
            channel.setDescription(description);
            NotificationManager notificationManager =
getSystemService(NotificationManager.class);
            notificationManager.createNotificationChannel(channel);
        }
    }
}
```

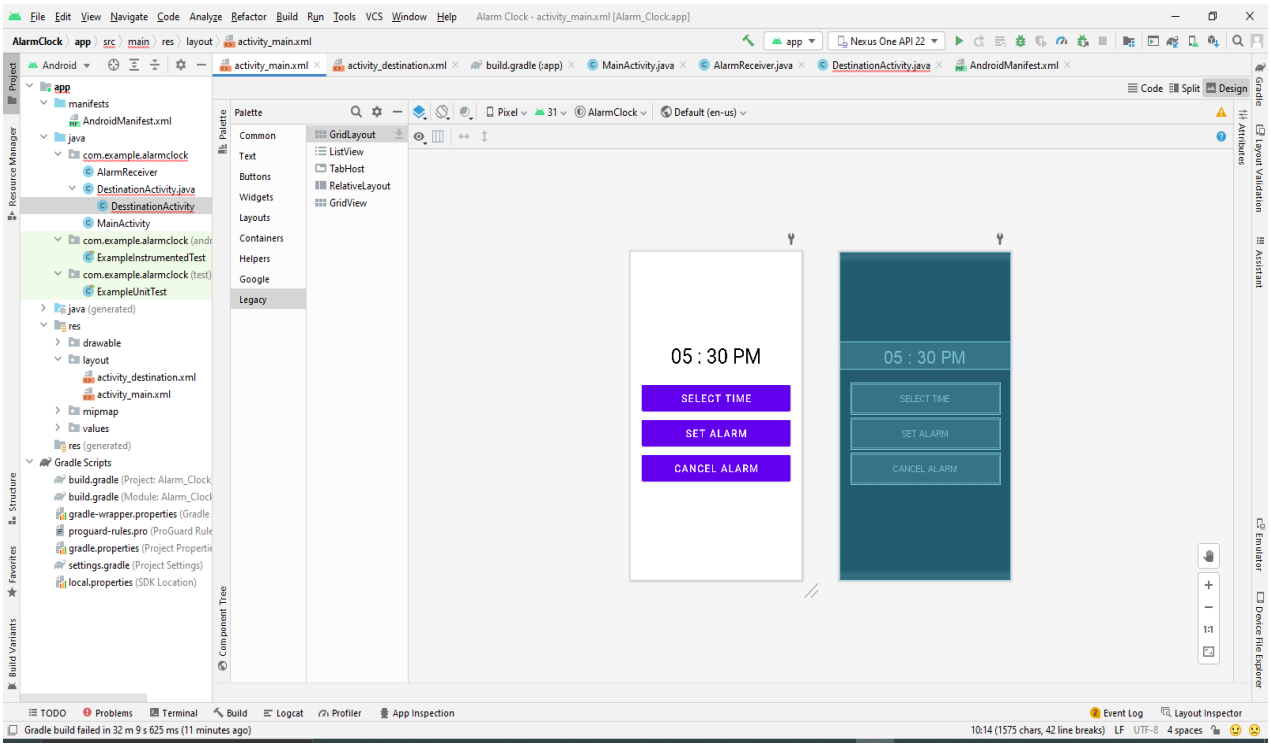
AlarmReceiver.java :

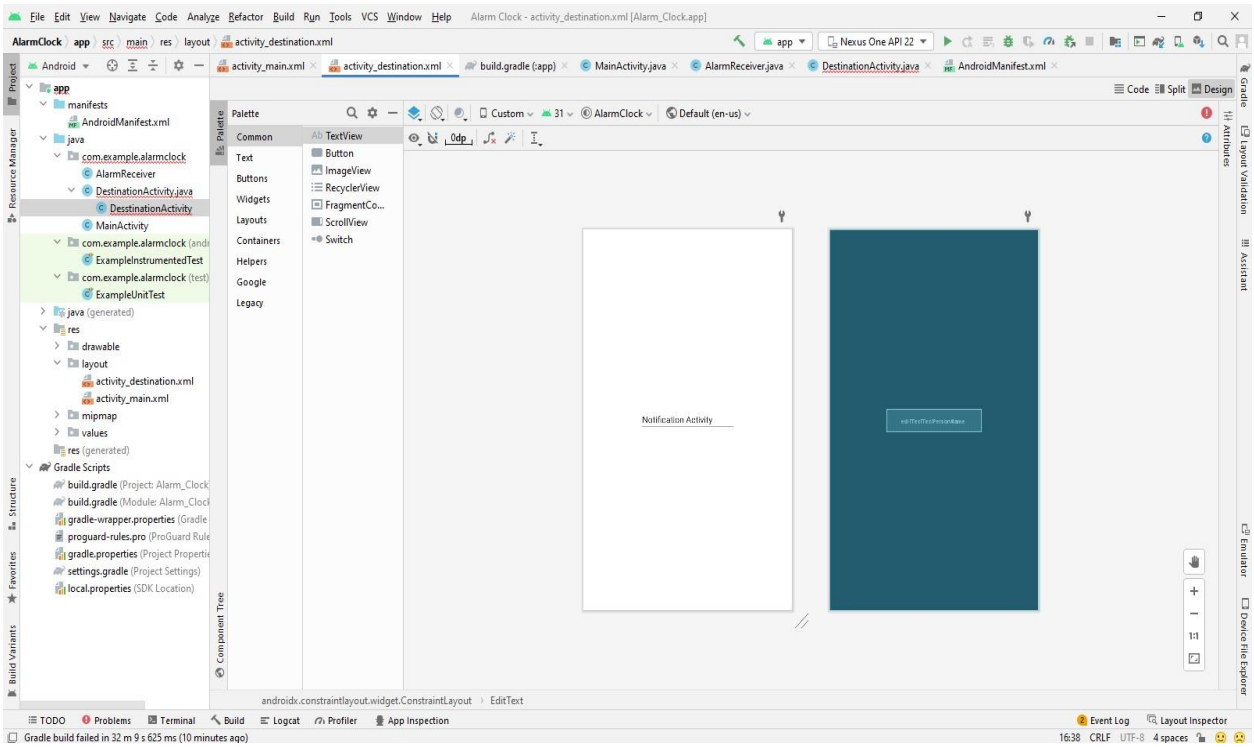
```
package com.example.alarmclock;
import android.app.Notification;
import android.app.NotificationManager;
import android.app.PendingIntent;
import android.content.BroadcastReceiver;
import android.content.Context;
import android.content.Intent;
import android.media.MediaPlayer;
import android.provider.MediaStore;
import android.provider.Settings;
import androidx.core.app.NotificationCompat;
import androidx.core.app.NotificationCompat.Builder;
import androidx.core.app.NotificationManagerCompat;
public class AlarmReceiver extends BroadcastReceiver{
    @Override
    public void onReceive(Context context, Intent intent) {
        Intent i=new Intent(context, DesstinationActivity.class);
        intent.setFlags(Intent.FLAG_ACTIVITY_NEW_TASK | Intent.FLAG_ACTIVITY_CLEAR_TASK);
        PendingIntent pendingIntent = PendingIntent.getActivity(context, 0,i, 0);
        MediaPlayer mediaPlayer = MediaPlayer.create(context,
Settings.System.DEFAULT_RINGTONE_URI);
        mediaPlayer.start();
        NotificationCompat.Builder builder = new NotificationCompat.Builder(context,
"Experiment")
                .setSmallIcon(R.drawable.ic_launcher_background)
                .setContentTitle("wakeUP Alarm Manager")
                .setAutoCancel(true)
                .setDefaults(NotificationCompat.DEFAULT_ALL)
                .setPriority(NotificationCompat.PRIORITY_HIGH)
                .setContentIntent(pendingIntent);
        NotificationManagerCompat notificationManagerCompat =
NotificationManagerCompat.from(context);
        notificationManagerCompat.notify(123, builder.build());
    }
}
```

DestinationActivity.java :

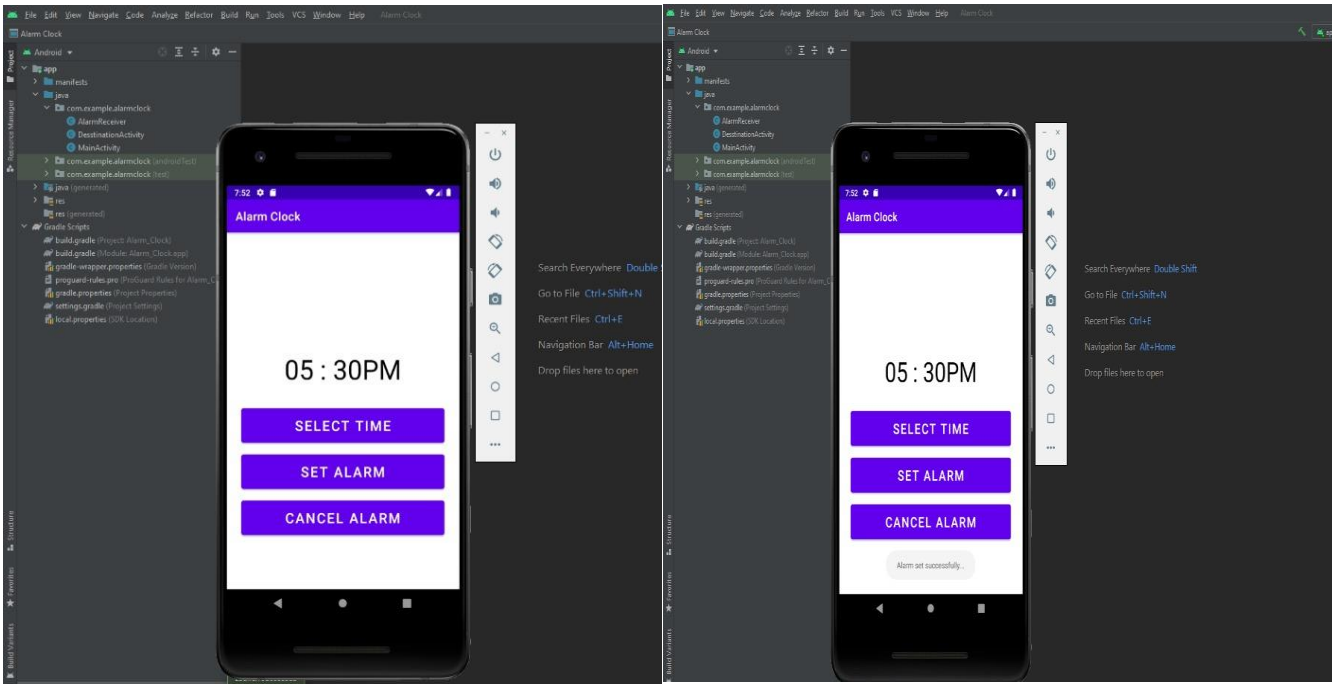
```
package com.example.alarmclock;
import androidx.appcompat.app.AppCompatActivity;
import android.os.Bundle;
public class DesstinationActivity extends AppCompatActivity {
    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_destination); } }
```

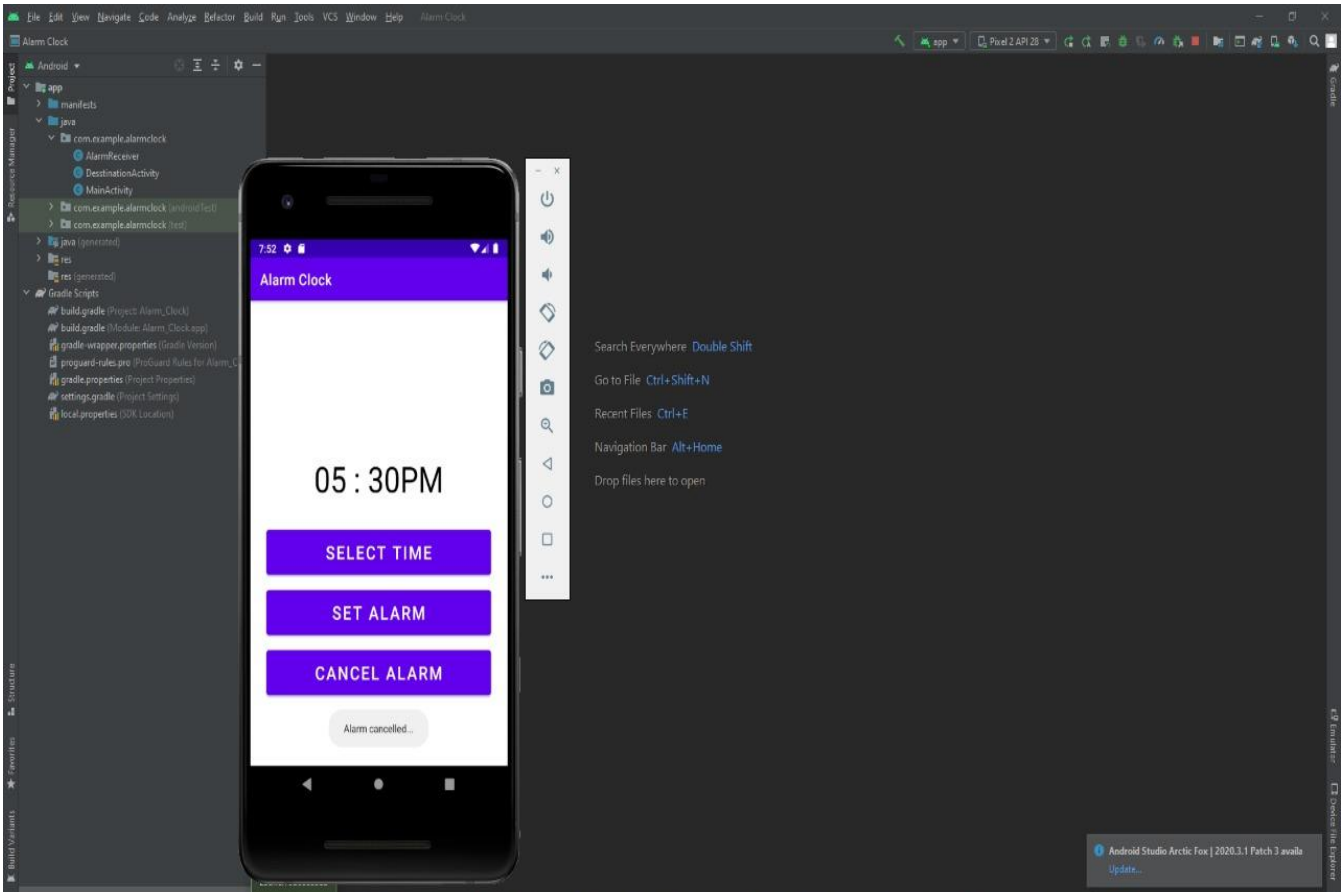
OUTPUT :




Alarm Clock - activity_destination.xml [Alarm_Clock.app]
Notification Activity
androidx.constraintlayout.widget.ConstraintLayout
EditText
Gradle build failed in 32 m 9 s 625 ms (10 minutes ago)

ON EMULATOR :


Alarm Clock
05 : 30PM
SELECT TIME
SET ALARM
CANCEL ALARM
Search Everywhere Double Shift
Go to File Ctrl+Shift+N
Recent Files Ctrl+E
Navigation Bar Alt+Home
Drop files here to open
Alarm Clock
05 : 30PM
SELECT TIME
SET ALARM
CANCEL ALARM
Alarm set successfully..
Search Everywhere Double Shift
Go to File Ctrl+Shift+N
Recent Files Ctrl+E
Navigation Bar Alt+Home
Drop files here to open



ANDROID LAB

ASSIGNMENT - 5

Aim : Create a Stopwatch in Android Studio.

Algorithm :

1. Create a project stopwatch with empty activity.

2. Go to xml code and make the layout, with setting width and height.
3. Make a chronometer, with two image button one start and another stop.
4. Go to drawable create background for the chronometer.
5. Set the width height and size of the layout, buttons, chronometer.
6. Create play, pause and stop button using vector assist from New->Vector Assist menu.
7. Go to java file and create on click listener for button like start, pause.
8. Once started, you can pause the clock and clicking on stop will reset the timer.
9. Write the code and run the app.

XML Code :

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="300dp"
    android:layout_height="300dp"
    android:layout_gravity="center"
    android:background="@drawable/bg_round"
    android:gravity="center"
    tools:context=".MainActivity">

    <Chronometer
        android:id="@+id/chronometer"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:format="00:00:00"
        android:textColor="@color/white"
        android:textSize="40sp"
        android:textStyle="bold" />

    <LinearLayout
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_marginTop="10dp">

        <ImageButton
            android:id="@+id/bt_start"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:background="@android:color/transparent"
            android:src="@drawable/ic_play" />

        <ImageButton
            android:id="@+id/bt_stop"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:layout_marginLeft="20dp"
            android:background="@android:color/transparent"
            android:src="@drawable/ic_stop" />
    </LinearLayout>
</LinearLayout>
```

bg_round.xml :

```
<?xml version="1.0" encoding="utf-8"?>
<shape
    xmlns:android="http://schemas.android.com/apk/res/android">
    android:shape="rectangle"
    <solid
        android:color="@color/purple_500"
    />
    <corners
        android:radius="200dp"
    />
</shape>
```

ic_play.xml :

```
<vector xmlns:android="http://schemas.android.com/apk/res/android"
    android:width="40dp"
    android:height="40dp"
    android:viewportWidth="24"
    android:viewportHeight="24"
    android:tint="?attr/colorControlNormal">
  <path
      android:fillColor="@android:color/white"
      android:pathData="M8,5v14l11,-7z"/>
</vector>
```

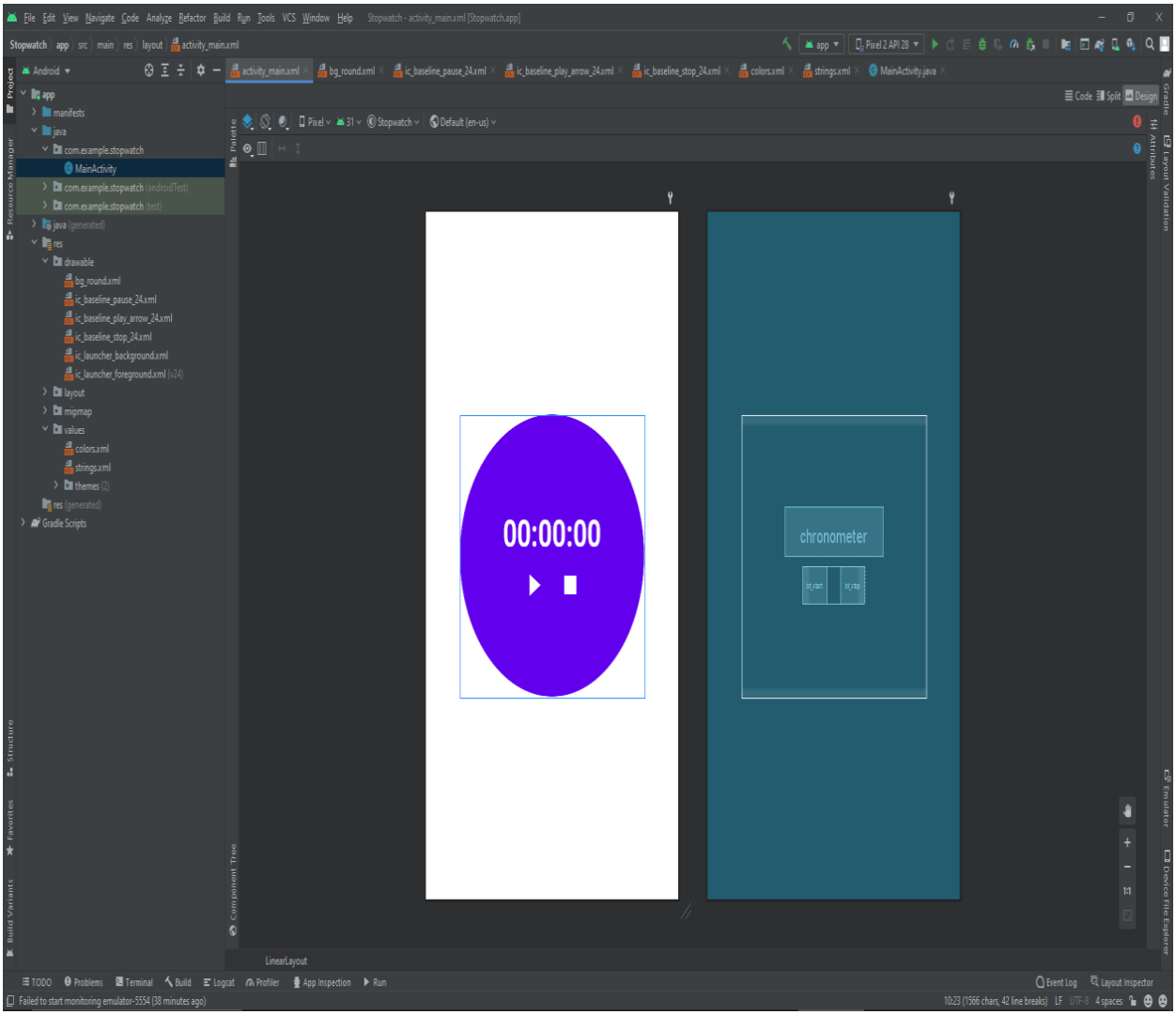
Ic_pause.xml :

```
<vector xmlns:android="http://schemas.android.com/apk/res/android"
    android:width="40dp"
    android:height="40dp"
    android:viewportWidth="24"
    android:viewportHeight="24"
    android:tint="?attr/colorControlNormal">
  <path
      android:fillColor="@android:color/white"
      android:pathData="M6,19h4L10,5L6,5v14zM14,5v14h4L18,5h-4z"/>
</vector>
```

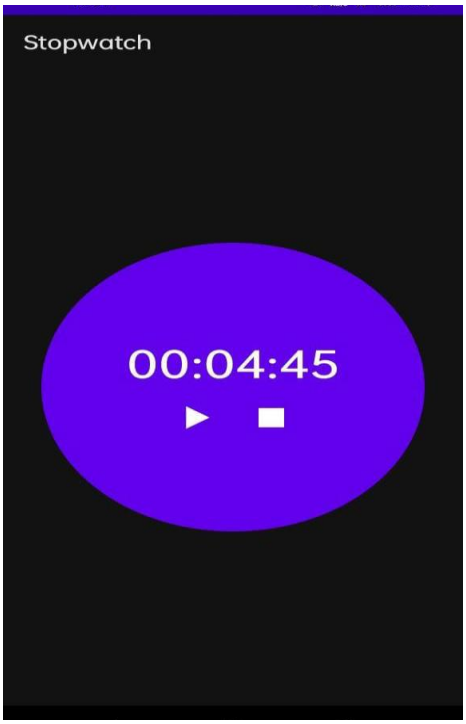
ic_stop.xml :

```
<vector xmlns:android="http://schemas.android.com/apk/res/android"
    android:width="40dp"
    android:height="40dp"
    android:viewportWidth="24"
    android:viewportHeight="24"
    android:tint="?attr/colorControlNormal">
  <path
      android:fillColor="@android:color/white"
      android:pathData="M6,6h12v12H6z"/>
</vector>
```

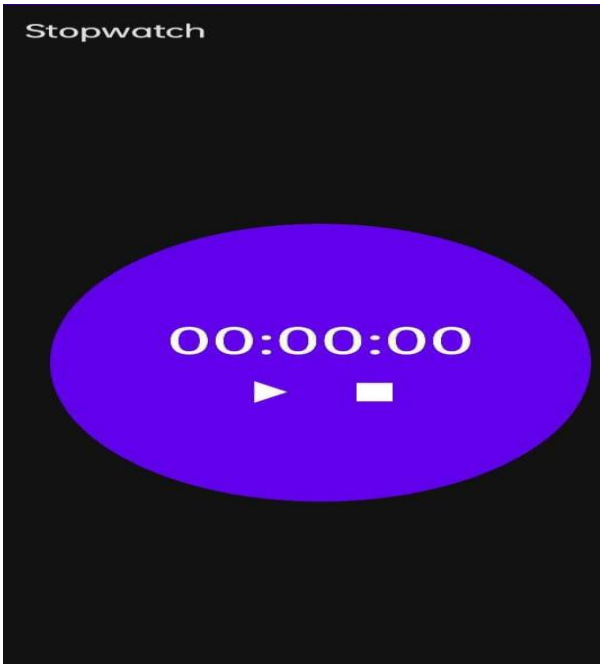
OUTPUT :



ON EMULATOR :



Pause :



Stop :

ANDROID LAB

ASSIGNMENT - 6

Aim : Create an Image Viewer in Android Studio.

Algorithm :

1. Create the new project image viewer and select empty activity.
2. Create a text view in main activity xml and write text for opening window of app, like, this is an image viewer.
3. Create the new empty activity in the java folder of the project name this activity as ImageViewerActivity.
4. In the xml code of the ImageViewerActivity create an image view setting it's width height and scaleType= "center", centerInParent = "True".
5. Go to ImageViewerActivity.java file and write the java code for viewing an image in our activity.

6. Our app is created and now we can run it.

XML CODE :

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:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:gravity="center"
        android:text="This is an Image Viewer"
        android:textSize="30sp"
        app:layout_constraintBottom_toBottomOf="parent"
        app:layout_constraintLeft_toLeftOf="parent"
        app:layout_constraintRight_toRightOf="parent"
        app:layout_constraintTop_toTopOf="parent"
    />
</androidx.constraintlayout.widget.ConstraintLayout>
```

activity_image_viewer :

```
<?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">

    <ImageView

        android:layout_width="match_parent"

        android:layout_height="match_parent"

        android:layout_centerInParent="true"

        android:scaleType="centerInside"

        android:id="@+id/imageView"
```

```
    />

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

AndroidManifest.xml :

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

<manifest xmlns:android="http://schemas.android.com/apk/res/android"
package="com.example.imgviewer">

    <application

        android:allowBackup="true"

        android:icon="@mipmap/ic_launcher"

        android:label="@string/app_name"

        android:roundIcon="@mipmap/ic_launcher_round"

        android:supportsRtl="true"

        android:theme="@style/Theme.ImgViewer">

        <activity

            android:name=".ImageViewerActivity"

            android:exported="true" >

            <intent-filter >

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

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

                <data android:mimeType="image/*"/>

            </intent-filter>

        </activity>

        <activity

            android:name=".MainActivity"

            android:exported="true">

            <intent-filter>

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

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

            </intent-filter>

        </activity>
```

```
    </application>
</manifest>
```

JAVA CODE :

MainActivity.java :

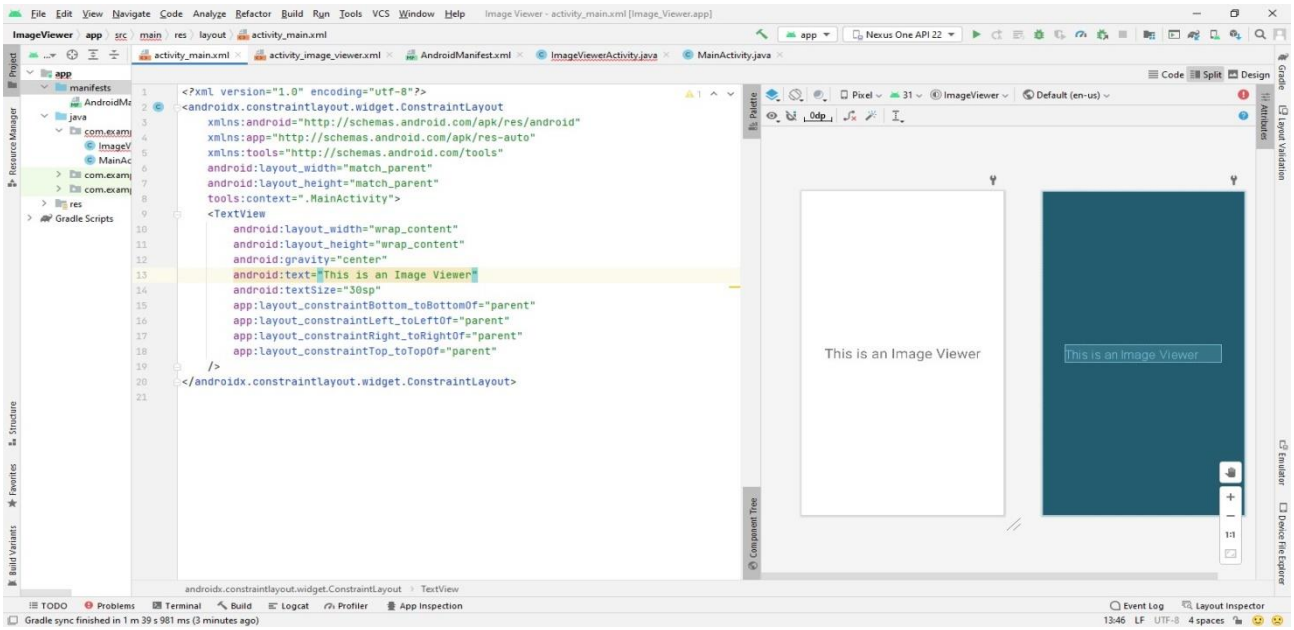
```
package com.example.imageviewer;
import androidx.appcompat.app.AppCompatActivity;
import android.os.Bundle;
public class MainActivity extends AppCompatActivity {
    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_main);
    }
}
```

ImageViewerActivity.java :

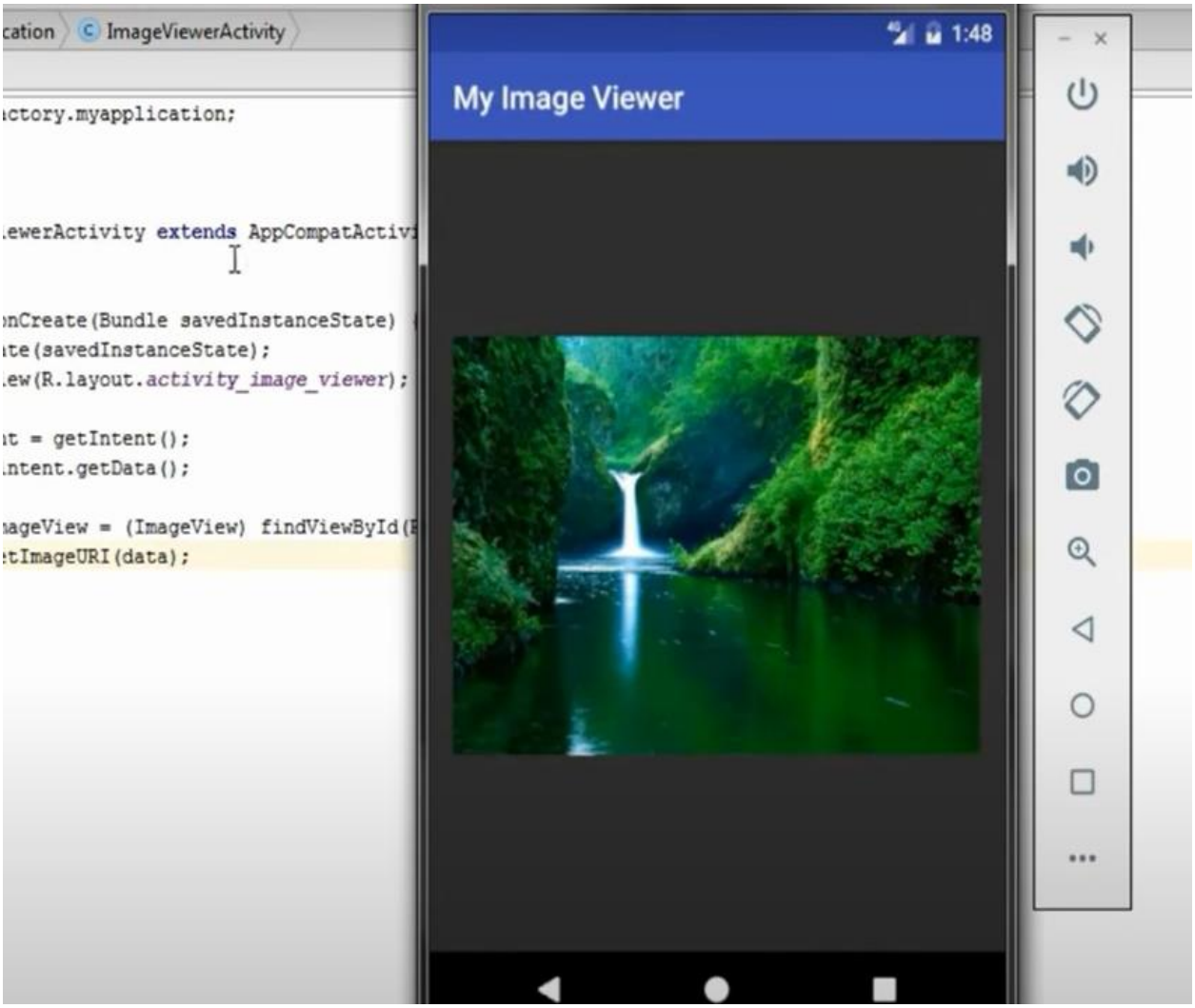
```
package com.example.imageviewer;
import androidx.appcompat.app.AppCompatActivity;
import android.content.Intent;
import android.net.Uri;
import android.os.Bundle;
import android.widget.ImageView;
public class ImageViewerActivity extends AppCompatActivity {
    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_image_viewer);
        Intent intent = getIntent();
        Uri data = intent.getData();
        ImageView imageView = findViewById(R.id.imageView);
```

```
        imageView.setImageURI(data);

    }

}
```

OUTPUT :



Emulator :


ImageViewerActivity
ctory.myapplication;
ewerActivity extends AppCompatActivi
nCreate(Bundle savedInstanceState)
te(savedInstanceState);
ew(R.layout.activity_image_viewer);
t = getIntent();
ntent.getData();
ageView = (ImageView) findViewById(
tImageURI(data);
1:48
My Image Viewer

----------end----------