Ex. No. 1	**Metric Converter**
Date of Exercise	**04/08/2023**

Aim

To create metric convert application using Java programming language.

Description

Activities

Android system initiates its program with in an Activity starting with a call on onCreate() callback method. There is a sequence of callback methods that start up an activity and a sequence of callback methods.

UI screen components

A typical user interface of an android application consists of action bar and the application content area.

- Main Action Bar
- View Control
- Content Area
- Split Action Bar

Types of layouts

There are many types of layouts. Some of which are listed below

- Linear Layout
- Absolute Layout
- Table Layout
- Frame Layout
- Relative Layout

UI Elements

A View is an object that draws something on the screen that the user can interact with and a ViewGroup is an object that holds other View (and ViewGroup) objects in order to define the layout of the user interface.

• TextView - This control is used to display text to the user.

• EditText - EditText is a predefined subclass of TextView that includes rich editing capabilities.

• Button - A push-button that can be pressed, or clicked, by the user to perform an action. • ImageButton - An ImageButton is an AbsoluteLayout which enables you to specify the exact location of its children. This shows a button with an image (instead of text) that can be pressed or clicked by the user.

• RadioButton - The RadioButton has two states: either checked or unchecked. • CheckBox - An on/off switch that can be toggled by the user. You should use check box when presenting users with a group of selectable options that are not mutually exclusive.

Program

XML code

```
<?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="Metric Converter"

        android:textColor="#0A040B"

        android:textSize="34sp"

        android:textStyle="bold"
        app:layout_constraintEnd_toEndOf="parent"
        app:layout_constraintStart_toStartOf="parent"
        app:layout_constraintTop_toTopOf="parent" />

    <EditText
        android:id="@+id/editTextNumber"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:ems="10"
        android:inputType="number"
        app:layout_constraintBottom_toBottomOf="parent"
        app:layout_constraintEnd_toEndOf="parent"
        app:layout_constraintHorizontal_bias="0.497"
        app:layout_constraintStart_toStartOf="parent"
        app:layout_constraintTop_toTopOf="parent"
        app:layout_constraintVertical_bias="0.157" />

    <TextView
        android:id="@+id/textView2"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="Enter in Kilometers"
        android:textColor="#3F51B5"
        android:textSize="24sp"
        app:layout_constraintBottom_toBottomOf="parent"
        app:layout_constraintEnd_toEndOf="parent"
        app:layout_constraintHorizontal_bias="0.497"
        app:layout_constraintStart_toStartOf="parent"
        app:layout_constraintTop_toTopOf="@+id/textView"
        app:layout_constraintVertical_bias="0.078" />

    <TextView
        android:id="@+id/valueMetres"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="Result"
        android:textSize="20sp"
        app:layout_constraintBottom_toBottomOf="parent"
        app:layout_constraintEnd_toEndOf="parent"
        app:layout_constraintHorizontal_bias="0.498"
        app:layout_constraintStart_toStartOf="parent"
        app:layout_constraintTop_toTopOf="parent"
        app:layout_constraintVertical_bias="0.362" />

    <TextView
        android:id="@+id/textView4"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_marginBottom="4dp"
        android:text="meter"
        android:textSize="24sp"
        app:layout_constraintBottom_toTopOf="@+id/valueMetres"
        app:layout_constraintEnd_toEndOf="parent"
        app:layout_constraintHorizontal_bias="0.498"
        app:layout_constraintStart_toStartOf="parent" />

    <Button
        android:id="@+id/button"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:background="#E91E63"
        android:text="Convert to metres"
        android:textColor="#FBF9FF"
        android:textColorHighlight="#673AB7"
        android:textColorHint="#3F51B5"
        app:iconTint="#FF9800"
        app:layout_constraintBottom_toTopOf="@+id/textView4"
        app:layout_constraintEnd_toEndOf="parent"
        app:layout_constraintStart_toStartOf="parent"
        app:layout_constraintTop_toBottomOf="@+id/editTextNumber"
        app:layout_constraintVertical_bias="0.45" />

    <TextView
        android:id="@+id/textView5"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="Enter in Fahrenheit"
        android:textColor="#009688"
        android:textSize="24sp"
        app:layout_constraintBottom_toBottomOf="parent"
        app:layout_constraintEnd_toEndOf="parent"
        app:layout_constraintHorizontal_bias="0.517"
        app:layout_constraintStart_toStartOf="parent"
        app:layout_constraintTop_toTopOf="@+id/textView"
        app:layout_constraintVertical_bias="0.463" />

    <EditText
        android:id="@+id/editTextNumber2"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:ems="10"
        android:inputType="number"
        app:layout_constraintBottom_toBottomOf="parent"
        app:layout_constraintEnd_toEndOf="parent"
        app:layout_constraintHorizontal_bias="0.514"
        app:layout_constraintStart_toStartOf="parent"
        app:layout_constraintTop_toTopOf="@+id/textView5"
        app:layout_constraintVertical_bias="0.146" />

    <Button
        android:id="@+id/button2"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_marginTop="92dp"
        android:background="#009688"
        android:text="Convert to Celsius"
        app:layout_constraintBottom_toBottomOf="parent"
        app:layout_constraintEnd_toEndOf="parent"
        app:layout_constraintHorizontal_bias="0.496"
        app:layout_constraintStart_toStartOf="parent"
        app:layout_constraintTop_toBottomOf="@+id/textView5"
        app:layout_constraintVertical_bias="0.0" />

    <TextView
        android:id="@+id/valueCelsius"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="result"
        android:textSize="24sp"
        app:layout_constraintBottom_toBottomOf="parent"
        app:layout_constraintEnd_toEndOf="parent"
        app:layout_constraintHorizontal_bias="0.498"
        app:layout_constraintStart_toStartOf="parent"
        app:layout_constraintTop_toTopOf="@+id/button2"
        app:layout_constraintVertical_bias="0.25" />

    <TextView
        android:id="@+id/textView3"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="Enter in Feet"
        android:textColor="#FF5722"
        android:textSize="24sp"
        app:layout_constraintBottom_toBottomOf="parent"
        app:layout_constraintEnd_toEndOf="parent"
        app:layout_constraintHorizontal_bias="0.501"
        app:layout_constraintStart_toStartOf="parent"
        app:layout_constraintTop_toBottomOf="@+id/valueCelsius"
        app:layout_constraintVertical_bias="0.112" />

    <EditText
        android:id="@+id/editTextNumber3"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:ems="10"
        android:inputType="number"
        app:layout_constraintBottom_toBottomOf="parent"
        app:layout_constraintEnd_toEndOf="parent"
        app:layout_constraintHorizontal_bias="0.485"
        app:layout_constraintStart_toStartOf="parent"
        app:layout_constraintTop_toBottomOf="@+id/textView3"
        app:layout_constraintVertical_bias="0.151" />

    <Button
        android:id="@+id/button3"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="onvert to inches"
        app:layout_constraintBottom_toBottomOf="parent"
        app:layout_constraintEnd_toEndOf="parent"
        app:layout_constraintHorizontal_bias="0.284"
        app:layout_constraintStart_toStartOf="parent"
        app:layout_constraintTop_toBottomOf="@+id/editTextNumber3"
        app:layout_constraintVertical_bias="0.578" />

    <TextView
        android:id="@+id/valueInches"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="result"
        android:textSize="24sp"
        app:layout_constraintBottom_toBottomOf="parent"
        app:layout_constraintEnd_toEndOf="parent"
        app:layout_constraintHorizontal_bias="0.643"
        app:layout_constraintStart_toStartOf="parent"
        app:layout_constraintTop_toTopOf="parent"
        app:layout_constraintVertical_bias="0.937" />

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

Java Code

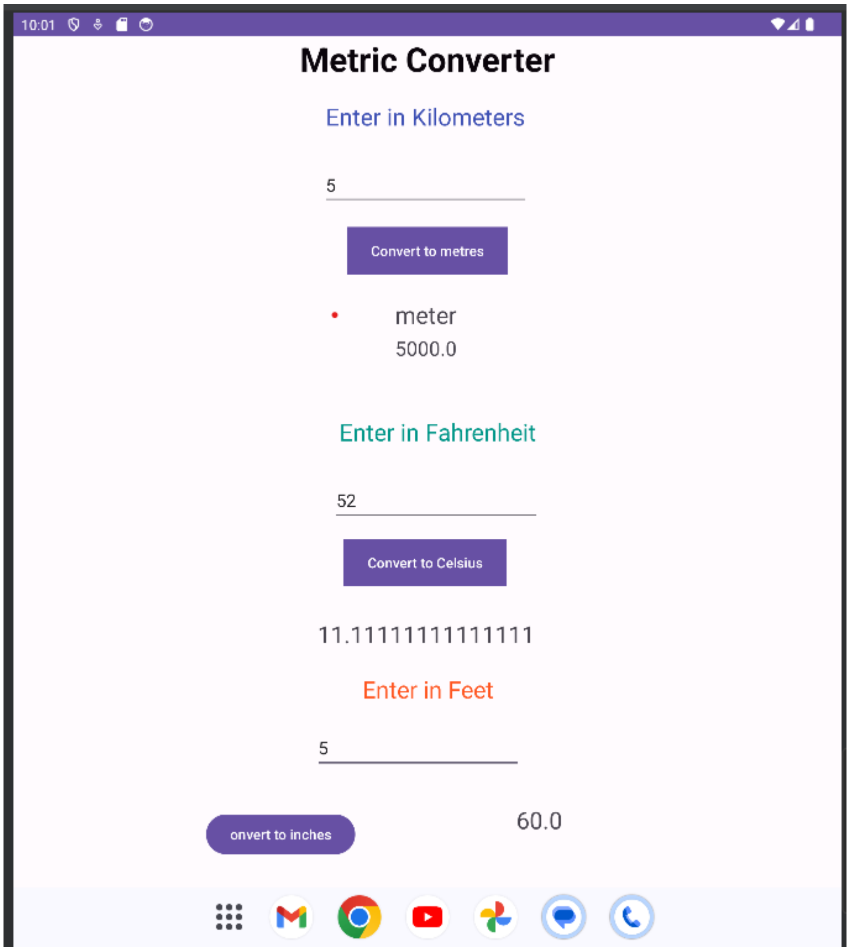
```
package com.example.myapplicationunitconverter;
```

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

import android.annotation.SuppressLint;

import android.os.Bundle;

import android.view.View;

import android.widget.Button;

import android.widget.EditText;

import android.widget.TextView;

public class MainActivity extends AppCompatActivity {

    // Declaring widgets

    EditText editText, editText2, editText3;

    TextView textView, textView2, textView4,textView3, valueMetres, textView5, valueCelsius, valueInches;

    Button button, button2, button3;

    @SuppressLint("MissingInflatedId")

    @Override

    protected void onCreate(Bundle savedInstanceState) {

        super.onCreate(savedInstanceState);

        setContentView(R.layout.activity_main);

        //Instantiating widgets

        textView = findViewById(R.id.textView);

        textView2 = findViewById(R.id.textView2);

        textView3 = findViewById(R.id.textView3);
```

```
textView4 = findViewById(R.id.textView4);

textView5 = findViewById(R.id.textView5);

valueMetres = findViewById(R.id.valueMetres);

valueCelsius = findViewById(R.id.valueCelsius);

valueInches = findViewById(R.id.valueInches);

editText = findViewById(R.id.editTextNumber);

editText2 = findViewById(R.id.editTextNumber2);

editText3 = findViewById(R.id.editTextNumber3);

button = findViewById(R.id.button);

button2 = findViewById(R.id.button2);

button3 = findViewById(R.id.button3);

// Adding a click event

button.setOnClickListener(new View.OnClickListener() {

    @Override

    public void onClick(View v) {

        ConvertFromKiloToMetre();

    }

});

button2.setOnClickListener(new View.OnClickListener() {

    @Override

    public void onClick(View v) {
            ConvertFromFahrenheitToCelsius();

        }

    });

    button3.setOnClickListener(new View.OnClickListener() {

        @Override

        public void onClick(View v) {

            ConvertFromFeetToInches();

        }

    });

}

@SuppressLint("SetTextI18n")

private void ConvertFromFeetToInches() {

    String ValueEnteredInFeet = editText3.getText().toString();

    double feet = Double.parseDouble(ValueEnteredInFeet);

    double inches = feet * 12;

    valueInches.setText("" + inches);

}

@SuppressLint("SetTextI18n")

private void ConvertFromFahrenheitToCelsius() {

    String ValueEnteredInFahrenheit = editText2.getText().toString();

    double fahrenheit = Double.parseDouble(ValueEnteredInFahrenheit);

    double celsius = (fahrenheit-32) * 5/9;
```

```
        valueCelsius.setText("" + celsius);

    }

    @SuppressLint("SetTextI18n")

    private void ConvertFromKiloToMetre() {

        String ValueEnteredInKilo = editText.getText().toString();

        double kilo = Double.parseDouble(ValueEnteredInKilo);

        double metre = kilo*1000;

        valueMetres.setText("" + metre);

    }

}
```

Output Screenshots



10:01

Metric Converter

Enter in Kilometers

5

Convert to metres

meter

5000.0

Enter in Fahrenheit

52

Convert to Celsius

11.11111111111111

Enter in Feet

5

onvert to inches

60.0

Result

Thus, the experiment for metric convert application using Java programming language has been coded, compiled and executed successfully.

Ex. No. 2	Student Information System
Date of Exercise	18/08/2023

Aim

To design a Student Information system using activity and intent concepts.

Description

Android activity lifecycle

In Android, an activity is referred to as one screen in an application. It is very similar to a single window of any desktop application. An Android app consists of one or more screens or activities. Each activity goes through various stages or a lifecycle and is managed by activity stacks. So when a new activity starts, the previous one always remains below it. There are four stages of an activity.

- If an activity is in the foreground of the screen i.e at the top of the stack, then it is said to be active or running. This is usually the activity that the user is currently interacting with.
- If an activity has lost focus and a non-full-sized or transparent activity has focused on top of your activity. In such a case either another activity has a higher position in multi-window mode or the activity itself is not focusable in the current window mode. Such activity is completely alive.
- If an activity is completely hidden by another activity, it is stopped or hidden. It still retains all the information, and as its window is hidden thus it will often be killed by the system when memory is needed elsewhere.
- The system can destroy the activity from memory by either asking it to finish or simply killing its process. When it is displayed again to the user, it must be completely restarted and restored to its previous state.

Intents and types

This process of taking users from one application to another is achieved by passing the Intent to the system. Intents, in general, are used for navigating among various activities within the same application, but note, is not limited to one single application, i.e., they can be utilized from moving from one application to another as well.

Intents could be Implicit, for instance, calling intended actions, and explicit as well, such as opening another activity after some operations like onClick or anything else.

There are two types of intents in android

- Implicit Intent

  Implicit Intent doesn't specify the component. In such a case, intent provides information on available components provided by the system that is to be invoked. For example, you may write the following code to view the webpage.
- Explicit

  Explicit Intent specifies the component. In such a case, intent provides the external class to be invoked.

Program

XML code

(Main_activity)

```
<?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:id="@+id/i1"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    android:background="#FFFFFF"
    tools:context=".MainActivity">

    <ImageButton
        android:id="@+id/examButton"
        android:layout_width="84dp"
        android:layout_height="75dp"
        android:backgroundTint="#068DCA"
        app:layout_constraintBottom_toBottomOf="parent"
        app:layout_constraintEnd_toEndOf="parent"
        app:layout_constraintHorizontal_bias="0.498"
        app:layout_constraintStart_toStartOf="parent"
        app:layout_constraintTop_toTopOf="parent"
        app:layout_constraintVertical_bias="0.945"
        app:srcCompat="@android:drawable/ic_menu_edit" />

    <EditText
        android:id="@+id/editTextText"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:ems="10"
        android:hint="Name"
        android:inputType="text"
        app:layout_constraintBottom_toBottomOf="parent"
        app:layout_constraintEnd_toEndOf="parent"
        app:layout_constraintHorizontal_bias="0.497"
        app:layout_constraintStart_toStartOf="parent"
        app:layout_constraintTop_toTopOf="parent"
        app:layout_constraintVertical_bias="0.237" />

    <TextView
        android:id="@+id/textView3"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:capitalize="sentences"
        android:text="Student Information System"
        android:textSize="24sp"
        android:textStyle="bold"
        app:layout_constraintBottom_toBottomOf="parent"
        app:layout_constraintEnd_toEndOf="parent"
        app:layout_constraintHorizontal_bias="0.495"
        app:layout_constraintStart_toStartOf="parent"
        app:layout_constraintTop_toTopOf="parent"
        app:layout_constraintVertical_bias="0.062" />

    <TextView
        android:id="@+id/textView4"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="Enter student information"
        android:textSize="20sp"
        android:textStyle="italic"
        app:layout_constraintBottom_toBottomOf="parent"
        app:layout_constraintEnd_toEndOf="parent"
        app:layout_constraintHorizontal_bias="0.497"
        app:layout_constraintStart_toStartOf="parent"
        app:layout_constraintTop_toTopOf="parent"
        app:layout_constraintVertical_bias="0.142" />

    <EditText
        android:id="@+id/editTextNumberDecimal"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:ems="10"
        android:hint="Register Number"
        android:inputType="numberDecimal"
        app:layout_constraintBottom_toBottomOf="parent"
        app:layout_constraintEnd_toEndOf="parent"
        app:layout_constraintHorizontal_bias="0.497"
        app:layout_constraintStart_toStartOf="parent"
        app:layout_constraintTop_toTopOf="parent"
        app:layout_constraintVertical_bias="0.362" />
```

```
<EditText
    android:id="@+id/editTextText2"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:ems="10"
    android:hint="Department"
    android:inputType="text"
    app:layout_constraintBottom_toBottomOf="parent"
    app:layout_constraintEnd_toEndOf="parent"
    app:layout_constraintHorizontal_bias="0.497"
    app:layout_constraintStart_toStartOf="parent"
    app:layout_constraintTop_toTopOf="parent"
    app:layout_constraintVertical_bias="0.478" />

<EditText
    android:id="@+id/editTextNumber"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:ems="10"
    android:hint="Sub 1 Marks"
    android:inputType="number"
    app:layout_constraintBottom_toBottomOf="parent"
    app:layout_constraintEnd_toEndOf="parent"
    app:layout_constraintHorizontal_bias="0.497"
    app:layout_constraintStart_toStartOf="parent"
    app:layout_constraintTop_toTopOf="parent"
    app:layout_constraintVertical_bias="0.597" />

<EditText
    android:id="@+id/editTextNumber2"
```

```
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:ems="10"
        android:hint="Sub 2 Marks"
        android:inputType="number"
        app:layout_constraintBottom_toBottomOf="parent"
        app:layout_constraintEnd_toEndOf="parent"
        app:layout_constraintHorizontal_bias="0.492"
        app:layout_constraintStart_toStartOf="parent"
        app:layout_constraintTop_toTopOf="parent"
        app:layout_constraintVertical_bias="0.698" />

    <EditText
        android:id="@+id/editTextNumber3"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:ems="10"
        android:hint="Sub 3 Marks"
        android:inputType="number"
        app:layout_constraintBottom_toBottomOf="parent"
        app:layout_constraintEnd_toEndOf="parent"
        app:layout_constraintHorizontal_bias="0.492"
        app:layout_constraintStart_toStartOf="parent"
        app:layout_constraintTop_toTopOf="parent"
        app:layout_constraintVertical_bias="0.807" />
</androidx.constraintlayout.widget.ConstraintLayout>
```

(activity_2)

```
<?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:id="@+id/info"
android:layout_width="match_parent"
android:layout_height="match_parent"
tools:context=".Activity2">

<TextView
    android:id="@+id/textView"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:text="Exams"
    android:textColor="#177AC8"
    android:textSize="24sp"
    android:textStyle="bold"
    app:layout_constraintBottom_toBottomOf="parent"
    app:layout_constraintEnd_toEndOf="parent"
    app:layout_constraintHorizontal_bias="0.498"
    app:layout_constraintStart_toStartOf="parent"
    app:layout_constraintTop_toTopOf="parent"
    app:layout_constraintVertical_bias="0.081"
    tools:ignore="MissingConstraints" />

<TextView
    android:id="@+id/textView2"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:text="text"
    android:textSize="20sp"
    app:layout_constraintBottom_toBottomOf="parent"
    app:layout_constraintEnd_toEndOf="parent"
    app:layout_constraintHorizontal_bias="0.567"
```

```
        app:layout_constraintStart_toStartOf="parent"
        app:layout_constraintTop_toTopOf="parent"
        app:layout_constraintVertical_bias="0.174"
        tools:ignore="MissingConstraints" />

    <TextView
        android:id="@+id/textView5"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="Name"
        android:textSize="20sp"
        android:textStyle="bold"
        app:layout_constraintBottom_toBottomOf="parent"
        app:layout_constraintEnd_toEndOf="parent"
        app:layout_constraintHorizontal_bias="0.206"
        app:layout_constraintStart_toStartOf="parent"
        app:layout_constraintTop_toTopOf="parent"
        app:layout_constraintVertical_bias="0.161" />

    <TextView
        android:id="@+id/Regno"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="TextView"
        android:textSize="20sp"
        app:layout_constraintBottom_toBottomOf="parent"
        app:layout_constraintEnd_toEndOf="parent"
        app:layout_constraintHorizontal_bias="0.58"
        app:layout_constraintStart_toStartOf="parent"
        app:layout_constraintTop_toTopOf="parent"
        app:layout_constraintVertical_bias="0.279" />
```

```
<TextView
    android:id="@+id/textView7"
    android:layout_width="62dp"
    android:layout_height="16dp"
    android:text="TextView"
    android:textSize="16sp"
    app:layout_constraintBottom_toBottomOf="parent"
    app:layout_constraintEnd_toEndOf="parent"
    app:layout_constraintHorizontal_bias="0.578"
    app:layout_constraintStart_toStartOf="parent"
    app:layout_constraintTop_toTopOf="parent"
    app:layout_constraintVertical_bias="0.366" />

<TextView
    android:id="@+id/textView8"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:text="TextView"
    android:textSize="20sp"
    app:layout_constraintBottom_toBottomOf="parent"
    app:layout_constraintEnd_toEndOf="parent"
    app:layout_constraintHorizontal_bias="0.58"
    app:layout_constraintStart_toStartOf="parent"
    app:layout_constraintTop_toTopOf="parent"
    app:layout_constraintVertical_bias="0.448" />

<TextView
    android:id="@+id/textView9"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
```

```
        android:text="TextView"
        android:textSize="20sp"
        app:layout_constraintBottom_toBottomOf="parent"
        app:layout_constraintEnd_toEndOf="parent"
        app:layout_constraintHorizontal_bias="0.58"
        app:layout_constraintStart_toStartOf="parent"
        app:layout_constraintTop_toTopOf="parent"
        app:layout_constraintVertical_bias="0.563" />

    <TextView
        android:id="@+id/textView10"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="TextView"
        android:textSize="20sp"
        app:layout_constraintBottom_toBottomOf="parent"
        app:layout_constraintEnd_toEndOf="parent"
        app:layout_constraintHorizontal_bias="0.58"
        app:layout_constraintStart_toStartOf="parent"
        app:layout_constraintTop_toTopOf="parent"
        app:layout_constraintVertical_bias="0.666" />

    <TextView
        android:id="@+id/textView11"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="TextView"
        app:layout_constraintBottom_toBottomOf="parent"
        app:layout_constraintEnd_toEndOf="parent"
        app:layout_constraintHorizontal_bias="0.209"
        app:layout_constraintStart_toStartOf="parent"
        app:layout_constraintTop_toTopOf="parent"
        app:layout_constraintVertical_bias="0.287" />

    <TextView
        android:id="@+id/textView12"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="TextView"
        app:layout_constraintBottom_toBottomOf="parent"
        app:layout_constraintEnd_toEndOf="parent"
        app:layout_constraintHorizontal_bias="0.209"
        app:layout_constraintStart_toStartOf="parent"
        app:layout_constraintTop_toTopOf="parent"
        app:layout_constraintVertical_bias="0.367" />

    <TextView
        android:id="@+id/textView13"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="TextView"
        tools:layout_editor_absoluteX="74dp"
        tools:layout_editor_absoluteY="328dp" />

    <TextView
        android:id="@+id/textView14"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="TextView"
        tools:layout_editor_absoluteX="72dp"
        tools:layout_editor_absoluteY="402dp" />
    <TextView
        android:id="@+id/textView15"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="TextView"
        tools:layout_editor_absoluteX="77dp"
        tools:layout_editor_absoluteY="476dp" />
</androidx.constraintlayout.widget.ConstraintLayout>
```

Java code

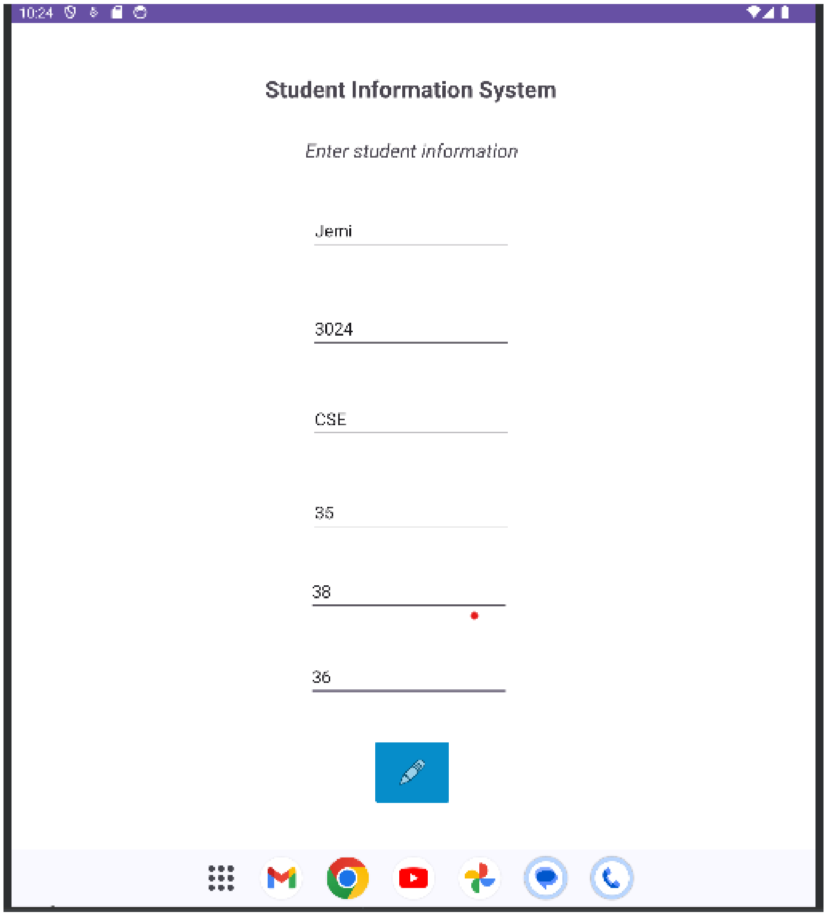
(Main_activity file)

```
package com.example.studentinformationapp;

import androidx.annotation.Nullable;
import androidx.appcompat.app.AppCompatActivity;

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

public class MainActivity extends AppCompatActivity {
    EditText e1, e2, e3, e4, e5, e6;
    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_main);

        e1= findViewById(R.id.editTextText);
        e2= findViewById(R.id.editTextNumberDecimal);
        e3= findViewById(R.id.editTextText2);
        e4= findViewById(R.id.editTextNumber);
        e5= findViewById(R.id.editTextNumber2);
        e6= findViewById(R.id.editTextNumber3);
        ImageButton button = findViewById(R.id.examButton);
        button.setOnClickListener(new View.OnClickListener() {
            @Override
            public void onClick(View v) {
                String str1 = e1.getText().toString();
                String str2 = e2.getText().toString();
                String str3 = e3.getText().toString();
                String str4 = e4.getText().toString();
                String str5 = e5.getText().toString();
                String str6 = e6.getText().toString();
                Intent i = new Intent(v.getContext(), Activity2.class);
                i.putExtra("message_key1", str1);
                i.putExtra("message_key2", str2);
                i.putExtra("message_key3", str3);
                i.putExtra("message_key4", str4);
                i.putExtra("message_key5", str5);
                i.putExtra("message_key6", str6);
                startActivity(i);
            }
        });

    }
}
```

(activity2 file)

```
package com.example.studentinformationapp;

import androidx.appcompat.app.AppCompatActivity;
import android.annotation.SuppressLint;
import android.content.Intent;
import android.os.Bundle;
import android.widget.EditText;
import android.widget.TextView;

public class Activity2 extends AppCompatActivity {
    TextView t1, t2, t3, t4, t5, t6;
    @SuppressLint("MissingInflatedId")
    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_2);
        t1 = findViewById(R.id.textView2);
        t2 = findViewById(R.id.Regno);
        t3 = findViewById(R.id.textView7);
        t4 = findViewById(R.id.textView8);
        t5 = findViewById(R.id.textView9);
        t6 = findViewById(R.id.textView10);
        Intent intent = getIntent();
        String str1 = intent.getStringExtra("message_key1");
        String str2 = intent.getStringExtra("message_key2");
        String str3 = intent.getStringExtra("message_key3");
        String str4 = intent.getStringExtra("message_key4");
        String str5 = intent.getStringExtra("message_key5");
        String str6 = intent.getStringExtra("message_key6");
        t1.setText(str1);
        t2.setText(str2);
        t3.setText(str3);
        t4.setText(str4);
        t5.setText(str5);
        t6.setText(str6);

    }
}
```

Output



Student Information System

Enter student information

Jemi

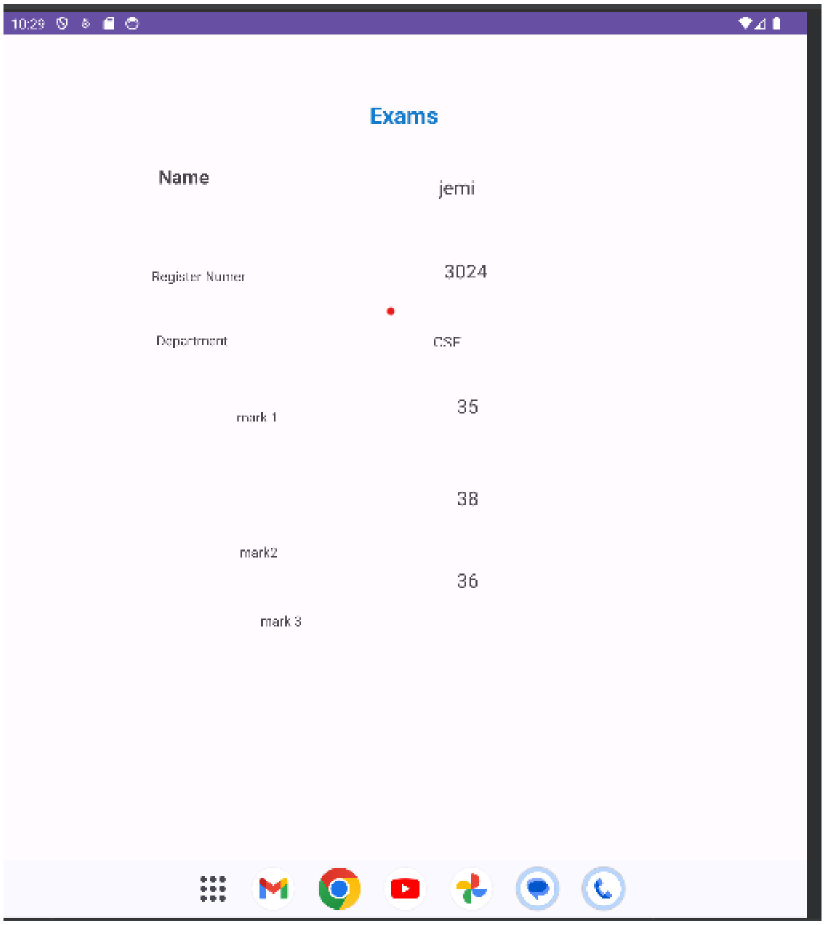
3024

CSE

35

38

36



Exams

Name jemi

Register Numer 3024

Department CSE

mark 1 35

38

mark2

36

mark 3

Result

The program for Student Information System is successfully implemented and verified.

Ex. No. 3	**Employee Management System**
Date of Exercise	**01/09/2023**

Aim

To design an Employee Management System using Room database.

Description

Room is a persistence library that provides an abstraction layer over the SQLite database to allow a more robust database. With the help of room, we can easily create the database and perform CRUD operations very easily.

Components of Room

The three main components of the room are Entity, Database, and DAO.

Entity: Entity is a modal class that is annotated with @Entity. This class is having variables that will be our columns and the class is our table.

Database: It is an abstract class where we will be storing all our database entries which we can call Entities.

DAO: The full form of DAO is a Database access object which is an interface class with the help of it we can perform different operations in our database.

Program

XML code

Java code

(AppDatabase)

```
package com.example.exp3_roomdatabase;
import android.content.Context;
import androidx.room.Database;
import androidx.room.Room;
import androidx.room.RoomDatabase;
@Database(entities = {Student.class},version=1,exportSchema = false)
public abstract class AppDatabase extends RoomDatabase{
    static AppDatabase appDatabase;
    public abstract StudentDao studentDao();
    public static AppDatabase getAppDatabase(Context context){
        if(appDatabase==null){
            appDatabase = Room.databaseBuilder(context, AppDatabase.class,
                    "studDatabase").build();
        }
        return appDatabase;
    }
}
```

(MainActivity)

```
package com.example.exp3_roomdatabase;
import androidx.annotation.NonNull;
import androidx.appcompat.app.AppCompatActivity;
import androidx.lifecycle.ViewModelProvider;
import android.content.Intent;
import android.os.Bundle;
import android.view.ContextMenu;
import android.view.MenuItem;
import android.view.View;
import android.widget.AdapterView;
import android.widget.ArrayAdapter;
import android.widget.ListView;
import androidx.room.Room;
import com.google.android.material.floatingactionbutton.FloatingActionButton;
import java.util.ArrayList;
import java.util.concurrent.Executors;
public class MainActivity extends AppCompatActivity {
    ListView listView;
    FloatingActionButton floatingActionButton;
    ArrayList<String> studentDetailsList;
    ArrayAdapter<String> stringArrayAdapter;
    private StudentViewModel studentViewModel;
    String name;
    int registerNumber;
    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_main);
        listView = findViewById(R.id.listViewId);
        floatingActionButton = findViewById(R.id.floatingActionButtonId);
        studentDetailsList = new ArrayList<>();
        stringArrayAdapter = new
ArrayAdapter<String>(this,android.R.layout.simple_list_item_1,studentDetailsList);
        stringArrayAdapter.notifyDataSetChanged();
        studentViewModel = new
                ViewModelProvider(this).get(StudentViewModel.class);
        studentViewModel.getAllRecords().observe(this,students -> {
            studentDetailsList.clear();
            for (Student student: students){
```

```
            studentDetailsList.add(student.getRegisterNumber()+" "+student.getStudentName());
          }
          listView.setAdapter(stringArrayAdapter);
        });
        registerForContextMenu(listView);
        floatingActionButton.setOnClickListener(new View.OnClickListener(){
          @Override
          public void onClick(View view){
            startActivity(new
                    Intent(getApplicationContext(),StudentActivity.class));
          }
        });
      }
      @Override
      public void onCreateContextMenu(ContextMenu menu, View v,
                                ContextMenu.ContextMenuInfo menuInfo){
        super.onCreateContextMenu(menu,v,menuInfo);
        getMenuInflater().inflate(R.menu.menu_student,menu);
        AdapterView.AdapterContextMenuInfo info =
(AdapterView.AdapterContextMenuInfo)menuInfo;
        String data[] = stringArrayAdapter.getItem(info.position).split(" ");
            registerNumber = Integer.parseInt((data[0]));
        name = data[1];
      }
      @Override
      public boolean onContextItemSelected(@NonNull MenuItem item){
        if(item.getItemId()==R.id.deleteID){
          Executors.newSingleThreadExecutor().execute(new Runnable(){
            @Override
            public void run(){
              studentViewModel.deleteRecords(new
                      Student(registerNumber, name));
            }
          });
        }
        else if (item.getItemId()==R.id.updateId){
// Executors.newSingleThreadExecutor().execute(new Runnable() {
// @Override
// public void run() {
// studentViewModel.updateRecord(new
//           Student(registerNumber,name));
// }
// });
            Intent intent = new Intent(getApplicationContext(), StudentActivity.class);
            intent.putExtra("registerNumber", registerNumber);
            intent.putExtra("studentName", name);
            startActivity(intent);
        }
        return true;
    }
}
```

(Student)

```
package com.example.exp3_roomdatabase;
import androidx.annotation.NonNull;
import androidx.room.Entity;
import androidx.room.PrimaryKey;
@Entity
public class Student {
    @PrimaryKey
    @NonNull
    int registerNumber;
    @NonNull
    String studentName;
    public Student(int registerNumber, String studentName){
        this.registerNumber = registerNumber;
        this.studentName=studentName;
    }
    public int getRegisterNumber(){return registerNumber;}
    public void setRegisterNumber(int registerNumber){this.registerNumber=registerNumber;}
    public String getStudentName(){return studentName;}
    public void setStudentName(String studentName){this.studentName=studentName;}
}
```

(StudentActivity)

```
package com.example.exp3_roomdatabase;
import androidx.appcompat.app.AppCompatActivity;
import androidx.lifecycle.ViewModelProvider;
import android.content.Intent;
import android.os.Bundle;
import android.view.View;
import android.widget.Button;
import android.widget.EditText;
import java.util.concurrent.Executors;
```

```
public class StudentActivity extends AppCompatActivity {
    EditText editTextRegNo,editTextName;
    Button buttonAdd,buttonDone;
    StudentViewModel studentViewModel;
    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_student);
        editTextName = findViewById(R.id.editTextNameId);
        editTextRegNo = findViewById(R.id.editTextRegNoId);
        buttonAdd = findViewById(R.id.buttonAddId);
        buttonDone = findViewById(R.id.buttonDoneId);
        studentViewModel = new
                ViewModelProvider(this).get(StudentViewModel.class);
        Intent intent = getIntent();
        if (intent!=null){
            int registerNumber = intent.getIntExtra("registerNumber",-1);
            String studentName = intent.getStringExtra("studentName");
            if(registerNumber!=-1 && studentName!=null){
                //Pre-fill the UI elements with existing data for editing
                editTextRegNo.setText(String.valueOf(registerNumber));
                editTextName.setText(studentName);
            }
        }
        buttonDone.setOnClickListener(new View.OnClickListener(){
            @Override
            public void onClick(View view){
                finish();
            }
        });
        buttonAdd.setOnClickListener(new View.OnClickListener(){
            @Override
            public void onClick(View view){
                int regNo=Integer.parseInt(editTextRegNo.getText().toString());
                String studentName = editTextName.getText().toString();
                Executors.newSingleThreadExecutor().execute(new Runnable(){
                    @Override
                    public void run(){
                        //check if the new registration number already exists
                        Student existingStudent = studentViewModel.getStudentByRegisterNumber(regNo);
                        if (existingStudent!=null){
                            existingStudent.setStudentName(studentName);
                            studentViewModel.updateRecord(existingStudent);
                        }
                        else {
                            studentViewModel.insertRecord(new
                                    Student(regNo, studentName));
                        }
                    }
                });
            }
        });
    }
}
```

(StudentDao)

```
package com.example.exp3_roomdatabase;
import androidx.lifecycle.LiveData;
import androidx.room.Dao;
import androidx.room.Insert;
import androidx.room.Query;
import androidx.room.Delete;
import androidx.room.Update;
import java.util.List;
@Dao
public interface StudentDao {
    @Insert
    void insert(Student student);
    @Delete
    void delete(Student... students);
    @Update
    void update(Student student);
    @Query("select * from student")
    LiveData<List<Student>> getAllStudents();
    @Query("SELECT * FROM student WHERE registerNumber = :regNo")
    Student getStudentByRegisterNumber(int regNo);
}
```

(StudentRepository)

```
package com.example.exp3_roomdatabase;
import android.app.Application;
import androidx.lifecycle.LiveData;
import java.util.List;
public class StudentRepository {
    private StudentDao studentDao;//DAO interface
    private LiveData<List<Student>> listLiveData;
    public StudentRepository(Application application){
        AppDatabase appDatabase = AppDatabase.getAppDatabase(application);
        studentDao = appDatabase.studentDao();//our database name
        listLiveData = studentDao.getAllStudents();
    }
    public void insertRecord(Student student){studentDao.insert(student);}
    public void deleteRecords(Student... students){studentDao.delete(students);}
    public void updateRecord(Student student){studentDao.update(student);
    }
    public Student getStudentByRegisterNumber(int regNo){
        return studentDao.getStudentByRegisterNumber(regNo);
    }
    public LiveData<List<Student>> getAllStudentsInfo(){ return
            studentDao.getAllStudents();}
}
```

(StudentViewModel)

```
package com.example.exp3_roomdatabase;
import android.app.Application;
import androidx.annotation.NonNull;
import androidx.lifecycle.AndroidViewModel;
import androidx.lifecycle.LiveData;
import java.util.List;
public class StudentViewModel extends AndroidViewModel {
    private StudentRepository studentRepository;
    public StudentViewModel(@NonNull Application application){
        super(application);
        studentRepository = new StudentRepository(application);
    }
    public LiveData<List<Student>> getAllRecords(){
        return studentRepository.getAllStudentsInfo();
    }
    public void insertRecord(Student
                                student){studentRepository.insertRecord(student);}
    public void deleteRecords(Student...
                                students){studentRepository.deleteRecords(students);}
    public void updateRecord(Student
                                student){studentRepository.updateRecord(student);}
    public Student getStudentByRegisterNumber(int regNo){
        return studentRepository.getStudentByRegisterNumber(regNo);
    }
}
```

XML- activity_main

```
<?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">

    <ListView
        android:id="@+id/listViewId"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        app:layout_constraintBottom_toBottomOf="parent"
        app:layout_constraintEnd_toEndOf="parent"
        app:layout_constraintHorizontal_bias="0.0"
        app:layout_constraintStart_toStartOf="parent"
        app:layout_constraintTop_toTopOf="parent"
        app:layout_constraintVertical_bias="0.04" />
    <com.google.android.material.floatingactionbutton.FloatingActionButton
        android:id="@+id/floatingActionButtonId"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:clickable="true"
        app:layout_constraintBottom_toBottomOf="parent"
        app:layout_constraintEnd_toEndOf="parent"
        app:layout_constraintHorizontal_bias="0.905"
        app:layout_constraintStart_toStartOf="parent"
        app:layout_constraintTop_toTopOf="parent"
        app:layout_constraintVertical_bias="0.666"
        app:srcCompat="@android:drawable/ic_input_add" />
</androidx.constraintlayout.widget.ConstraintLayout>
```

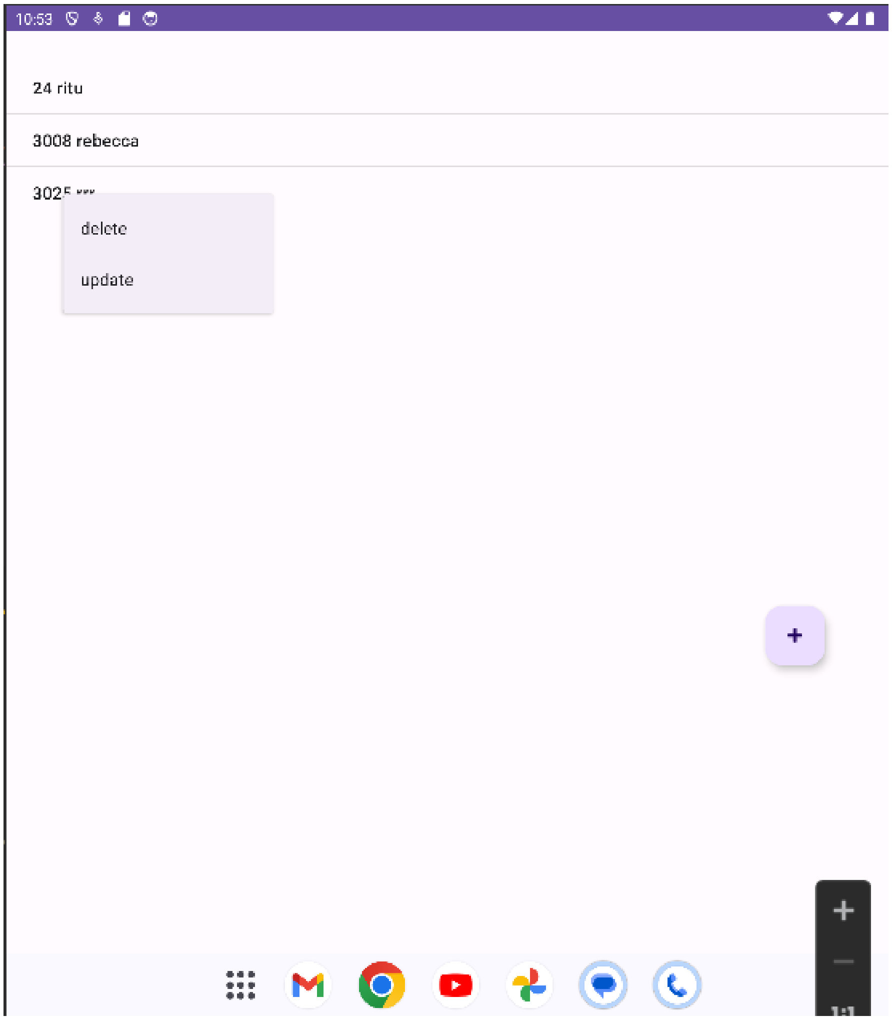
(Activity_student)

```
<?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=".StudentActivity">
<Button
   android:id="@+id/buttonDoneId"
   android:layout_width="wrap_content"
   android:layout_height="wrap_content"
   android:text="DONE"
   app:layout_constraintBottom_toBottomOf="parent"
   app:layout_constraintEnd_toEndOf="parent"
   app:layout_constraintStart_toStartOf="parent"
   app:layout_constraintTop_toTopOf="parent" />
<EditText
   android:id="@+id/editTextNameId"
   android:layout_width="wrap_content"
   android:layout_height="wrap_content"
   android:ems="10"
   android:inputType="text"
   app:layout_constraintBottom_toBottomOf="parent"
   app:layout_constraintEnd_toEndOf="parent"
   app:layout_constraintHorizontal_bias="0.264"
   app:layout_constraintStart_toEndOf="@+id/textView"
   app:layout_constraintTop_toTopOf="parent"
   app:layout_constraintVertical_bias="0.067" />
<EditText
   android:id="@+id/editTextRegNoId"
   android:layout_width="wrap_content"
   android:layout_height="wrap_content"
   android:ems="10"
   android:inputType="text"
   app:layout_constraintBottom_toBottomOf="parent"
   app:layout_constraintEnd_toEndOf="parent"
   app:layout_constraintHorizontal_bias="0.156"
   app:layout_constraintStart_toEndOf="@+id/textView2"
   app:layout_constraintTop_toTopOf="parent"
   app:layout_constraintVertical_bias="0.15" />
<Button
   android:id="@+id/buttonAddId"
   android:layout_width="wrap_content"
   android:layout_height="wrap_content"
   android:text="ADD"
```

```
        app:layout_constraintBottom_toBottomOf="parent"
        app:layout_constraintEnd_toEndOf="parent"
        app:layout_constraintStart_toStartOf="parent"
        app:layout_constraintTop_toTopOf="parent"
        app:layout_constraintVertical_bias="0.339" />
    <TextView
        android:id="@+id/textView"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="Name"
        android:textSize="20dp"
        app:layout_constraintBottom_toBottomOf="parent"
        app:layout_constraintEnd_toEndOf="parent"
        app:layout_constraintHorizontal_bias="0.068"
        app:layout_constraintStart_toStartOf="parent"
        app:layout_constraintTop_toTopOf="parent"
        app:layout_constraintVertical_bias="0.066" />
    <TextView
        android:id="@+id/textView2"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="Identification Number"
        android:textSize="20dp"
        app:layout_constraintBottom_toBottomOf="parent"
        app:layout_constraintEnd_toEndOf="parent"
        app:layout_constraintHorizontal_bias="0.069"
        app:layout_constraintStart_toStartOf="parent"
        app:layout_constraintTop_toTopOf="@+id/textView"
        app:layout_constraintVertical_bias="0.088" />
</androidx.constraintlayout.widget.ConstraintLayout>
```

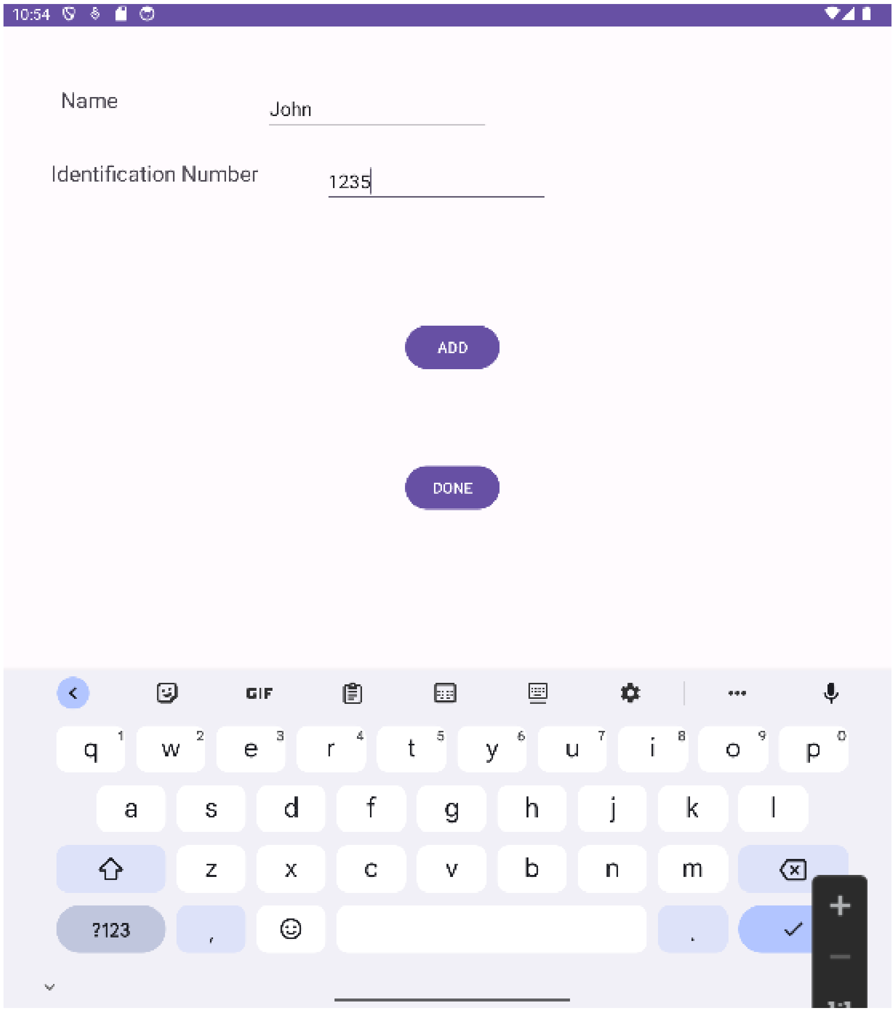
Output



24 ritu

3008 rebecca

3025



Name John

Identification Number 1235

Result

The program for Employee Management System is successfully coded using SQLite database.

Ex. No. 4	**Contact App using Recycler View Adapter**
Date of Exercise	**15/09/2023**

Aim

To create a contact app using Recycler View Adapter in Android Studio.

Description

RecyclerView is a ViewGroup added to the android studio as a successor of the GridView and ListView. It is an improvement on both of them and can be found in the latest v-7 support packages. It has been created to make possible construction of any lists with XML layouts as an item which can be customized vastly while improving on the efficiency of ListViews and GridViews. This improvement is achieved by recycling the views which are out of the visibility of the user. For example, if a user scrolled down to a position where items 4 and 5 are visible; items 1, 2, and 3 would be cleared from the memory to reduce memory consumption.

- The Card Layout: The card layout is an XML layout which will be treated as an item for the list created by the RecyclerView.
- The ViewHolder: The ViewHolder is a java class that stores the reference to the card layout views that have to be dynamically modified during the execution of the program by a list of data obtained either by online databases or added in some other way.
- The Data Class: The Data class is a custom java class that acts as a structure for holding the information for every item of the RecyclerView.

The Adapter: The adapter is the main code responsible for RecyclerView. It holds all the important methods dealing with the implementation of RecylcerView. The basic methods for a successful implementation are:

- onCreateViewHolder: which deals with the inflation of the card layout as an item for the RecyclerView.
- onBindViewHolder: which deals with the setting of different data and methods related to clicks on particular items of the RecyclerView.
- getItemCount: which Returns the length of the RecyclerView.
- onAttachedToRecyclerView: which attaches the adapter to the RecyclerView.

Program

XML code

(activity_main)

```
<?xml version="1.0" encoding="utf-8"?>
<androidx.constraintlayout.widget.ConstraintLayout
xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:app="http://schemas.android.com/apk/res-auto"
    xmlns:tools="http://schemas.android.com/tools"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    tools:context=".MainActivity">
    <androidx.recyclerview.widget.RecyclerView
        android:id="@+id/recyclerViewId"
        android:layout_width="409dp"
        android:layout_height="729dp"
        app:layout_constraintEnd_toEndOf="parent"
        app:layout_constraintHorizontal_bias="0.0"
        app:layout_constraintStart_toStartOf="parent"
        app:layout_constraintTop_toTopOf="parent" />
</androidx.constraintlayout.widget.ConstraintLayout>
```

(contact_details)

```
<?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="wrap_content">
    <ImageView
        android:id="@+id/imageView"
        android:layout_width="100dp"
        android:layout_height="100dp"
        app:layout_constraintBottom_toBottomOf="parent"
        app:layout_constraintHorizontal_bias="0.051"
        app:layout_constraintStart_toStartOf="parent"
        app:layout_constraintTop_toTopOf="parent"
        app:layout_constraintVertical_bias="1.0"
        tools:srcCompat="@tools:sample/avatars" />
    <TextView
        android:id="@+id/nameId"
        android:layout_width="265dp"
        android:layout_height="32dp"
        android:gravity="center"
        android:text="Name"
        android:textSize="25sp"
        app:layout_constraintBottom_toTopOf="@+id/numberId"
        app:layout_constraintHorizontal_bias="0.534"
        app:layout_constraintStart_toEndOf="@+id/imageView"
        app:layout_constraintTop_toTopOf="@+id/imageView"
        app:layout_constraintVertical_bias="0.047" />
    <TextView
        android:id="@+id/numberId"
        android:layout_width="269dp"
        android:layout_height="30dp"
        android:gravity="center"
        android:text="Number"
        android:textSize="25sp"
        app:layout_constraintBottom_toBottomOf="parent"
        app:layout_constraintEnd_toEndOf="parent"
        app:layout_constraintHorizontal_bias="0.57"
        app:layout_constraintStart_toEndOf="@+id/imageView"
        app:layout_constraintTop_toBottomOf="@+id/nameId"
        app:layout_constraintVertical_bias="0.1" />
</androidx.constraintlayout.widget.ConstraintLayout>
```

Java code (main_activity)

```
package com.example.exp4_contactview;
import androidx.appcompat.app.AppCompatActivity;
import androidx.recyclerview.widget.LinearLayoutManager;
import androidx.recyclerview.widget.RecyclerView;

import android.annotation.SuppressLint;
import android.os.Bundle;
import java.util.Arrays;
public class MainActivity extends AppCompatActivity {
    RecyclerView recyclerView;
    ContactAdapter contactAdapter;
    @SuppressLint("MissingInflatedId")
    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_main);
        recyclerView = findViewById(R.id.recyclerViewId);
        contactAdapter = new ContactAdapter(this,
                Arrays.asList("John", "Ria", "Maria"),
                Arrays.asList("123","456","789"),
                new int[]{R.drawable.p1, R.drawable.p2, R.drawable.p3}
        );
        recyclerView.setAdapter(contactAdapter);
        recyclerView.setLayoutManager(new LinearLayoutManager( this));
    }
}
```

(contactAdapter)

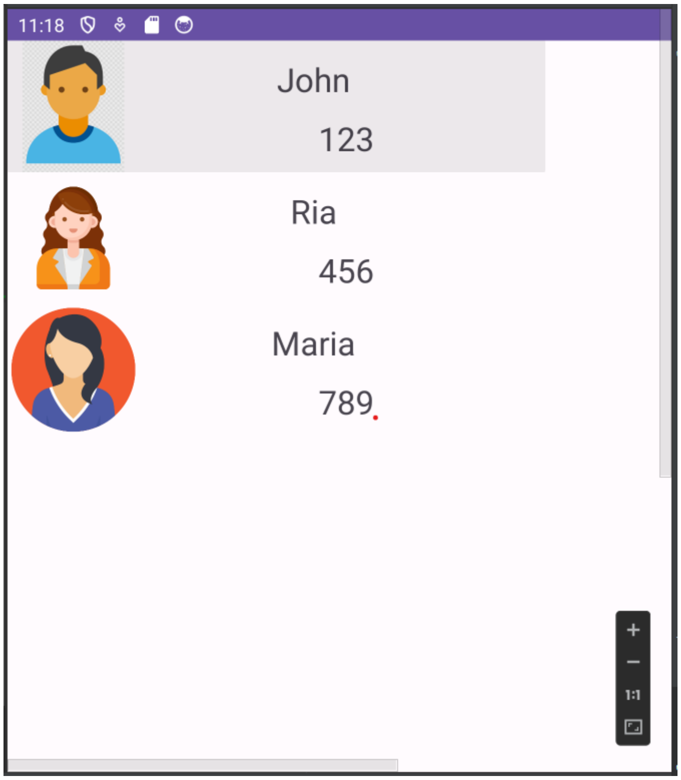
```
package com.example.exp4_contactview;
```

```
import androidx.appcompat.app.AppCompatActivity;
import androidx.recyclerview.widget.LinearLayoutManager;
import androidx.recyclerview.widget.RecyclerView;
import android.annotation.SuppressLint;
import android.os.Bundle;
import java.util.Arrays;
public class MainActivity extends AppCompatActivity {
    RecyclerView recyclerView;
    ContactAdapter contactAdapter;
    @SuppressLint("MissingInflatedId")
    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_main);
        recyclerView = findViewById(R.id.recyclerViewId);
        contactAdapter = new ContactAdapter(this,
                Arrays.asList("John", "Ria", "Maria"),
                Arrays.asList("123","456","789"),
                new int[]{R.drawable.p1, R.drawable.p2, R.drawable.p3}
        );
        recyclerView.setAdapter(contactAdapter);
        recyclerView.setLayoutManager(new LinearLayoutManager( this));
    }
}
```

(contactViewHolder)

```
package com.example.exp4_contactview;

import android.content.Context;
import android.content.Intent;
import android.net.Uri;
import android.view.View;
import android.widget.ImageView;
import android.widget.TextView;
import androidx.annotation.NonNull;
import androidx.recyclerview.widget.RecyclerView;
import java.util.List;
public class ContactViewHolder extends RecyclerView.ViewHolder implements
View.OnClickListener{
    public ImageView imageView;
    public TextView nameTextView, numberTextView;
    List<String> numberList;
    public ContactViewHolder(@NonNull View itemView, List<String> numberList){
        super(itemView);
        itemView.setOnClickListener(this);
        imageView = itemView.findViewById(R.id.imageView);
        nameTextView = itemView.findViewById(R.id.nameId);
        numberTextView = itemView.findViewById(R.id.numberId);
        this.numberList = numberList;
    }
    @Override
    public void onClick(View view){
        int position = getLayoutPosition();
        String element = numberList.get(position);
        Context context = view.getContext();
        context.startActivity(new Intent(Intent.ACTION_DIAL, Uri.parse( "tel:"+element)));
    }
}
```

Output



Result

The program for creating a contact app using Recycler View Adapter is successfully coded and verified.

Ex No: 5	CREATE A MEDIA PLAYER APPLICATION
Date:	12/10/2023

AIM:

To create a Media Player Application.

DESCRIPTION:

A RecyclerView is a popular and flexible widget in Android that is used to display a scrollable list or grid of items in a more efficient and memory-friendly way compared to the older ListView. It is a fundamental component for creating dynamic and interactive user interfaces in Android apps.

RecyclerView is a versatile and efficient UI component in Android that facilitates the creation of dynamic, scrollable lists or grids of items. Its key advantages include efficient item recycling, flexibility in layout management, support for item animations, and ease of use with adapters, making it a crucial element for creating responsive and memory-efficient Android apps.

Key Components of the RecyclerView:

Adapter- The Adapter is the bridge between your data source and the RecyclerView. It's responsible for creating ViewHolders and binding data to them. To use the RecyclerView, you need to create a custom Adapter class that extends RecyclerView.Adapter. This class defines how your data is presented in the RecyclerView and how to respond to user interactions.

ViewHolder- ViewHolders are lightweight objects that hold references to the views within each item of the RecyclerView. They play a crucial role in optimizing performance by recycling views as the user scrolls. By caching references to views, ViewHolders reduce the need for findViewById calls, which can be resource-intensive.

PROGRAM:

XML Code-

activity_main.xml

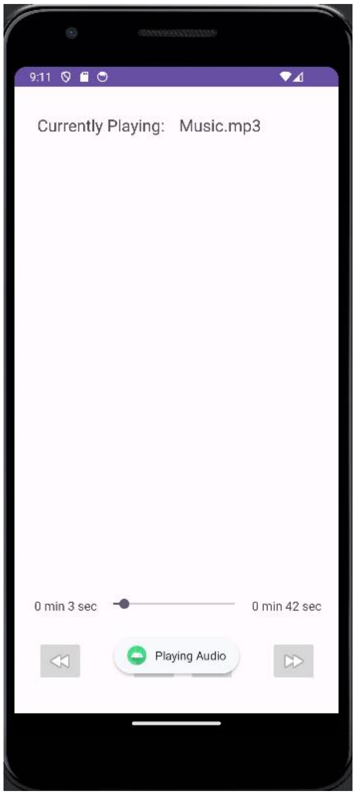
```
<?xml version="1.0" encoding="utf-8"?>
<androidx.constraintlayout.widget.ConstraintLayout
    xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:app="http://schemas.android.com/apk/res-auto"
    xmlns:tools="http://schemas.android.com/tools"
    android:layout_width="match_parent"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

    <TextView
        android:id="@+id/cp_label"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="@string/currently_playing"
        android:textSize="20sp"
        app:layout_constraintBottom_toBottomOf="parent"
        app:layout_constraintEnd_toEndOf="parent"
        app:layout_constraintHorizontal_bias="0.116"
        app:layout_constraintStart_toStartOf="parent"
        app:layout_constraintTop_toTopOf="parent"
        app:layout_constraintVertical_bias="0.045" />

    <TextView
        android:id="@+id/s_name"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text=""
        android:textSize="20sp"
        app:layout_constraintBottom_toBottomOf="parent"
        app:layout_constraintEnd_toEndOf="parent"
        app:layout_constraintHorizontal_bias="0.158"
        app:layout_constraintStart_toEndOf="@+id/cp_label"
        app:layout_constraintTop_toTopOf="parent"
        app:layout_constraintVertical_bias="0.045" />

    <ImageButton
        android:id="@+id/fast_reverse"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        app:layout_constraintBottom_toBottomOf="parent"
        app:layout_constraintEnd_toEndOf="parent"
        app:layout_constraintHorizontal_bias="0.081"
        app:layout_constraintStart_toStartOf="parent"
        app:layout_constraintTop_toTopOf="parent"
        app:layout_constraintVertical_bias="0.947"
        app:srcCompat="@android:drawable/ic_media_rew"
        android:contentDescription="@string/previous" />

    <ImageButton
        android:id="@+id/fast_forward"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:contentDescription="@string/forward"
        app:layout_constraintBottom_toBottomOf="parent"
        app:layout_constraintEnd_toEndOf="parent"
        app:layout_constraintHorizontal_bias="0.896"
        app:layout_constraintStart_toEndOf="@+id/fast_reverse"
        app:layout_constraintTop_toTopOf="parent"
        app:layout_constraintVertical_bias="0.947"
        app:srcCompat="@android:drawable/ic_media_ff" />

    <ImageButton
        android:id="@+id/pause_btn"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_marginStart="48dp"
        android:contentDescription="@string/pause"
        app:layout_constraintBottom_toBottomOf="parent"
        app:layout_constraintEnd_toStartOf="@+id/fast_forward"
        app:layout_constraintHorizontal_bias="0.079"
        app:layout_constraintStart_toEndOf="@+id/fast_reverse"
        app:layout_constraintTop_toTopOf="parent"
        app:layout_constraintVertical_bias="0.947"
        app:srcCompat="@android:drawable/ic_media_pause" />

    <ImageButton
        android:id="@+id/play_btn"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:contentDescription="@string/play"
        app:layout_constraintBottom_toBottomOf="parent"
        app:layout_constraintEnd_toStartOf="@+id/fast_forward"
        app:layout_constraintHorizontal_bias="0.748"
        app:layout_constraintStart_toEndOf="@+id/fast_reverse"
        app:layout_constraintTop_toTopOf="parent"
        app:layout_constraintVertical_bias="0.947"
        app:srcCompat="@android:drawable/ic_media_play" />

    <TextView
        android:id="@+id/current_time"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="@string/_0_min_0_sec"
        android:textSize="15sp"
        app:layout_constraintBottom_toBottomOf="parent"
        app:layout_constraintEnd_toEndOf="parent"
        app:layout_constraintHorizontal_bias="0.076"
        app:layout_constraintStart_toStartOf="parent"
        app:layout_constraintTop_toTopOf="parent"
        app:layout_constraintVertical_bias="0.837" />

    <TextView
        android:id="@+id/song_time"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="@string/_0min_0sec"
        android:textSize="15sp"
        app:layout_constraintBottom_toBottomOf="parent"
        app:layout_constraintEnd_toEndOf="parent"
        app:layout_constraintHorizontal_bias="0.901"
        app:layout_constraintStart_toEndOf="@+id/current_time"
        app:layout_constraintTop_toTopOf="parent"
        app:layout_constraintVertical_bias="0.837" />

    <SeekBar
        android:id="@+id/seekBar"
        android:layout_width="180dp"
        android:layout_height="16dp"
        app:layout_constraintBottom_toBottomOf="parent"
        app:layout_constraintEnd_toStartOf="@+id/song_time"
        app:layout_constraintHorizontal_bias="0.433"
        app:layout_constraintStart_toEndOf="@+id/current_time"
        app:layout_constraintTop_toTopOf="parent"
        app:layout_constraintVertical_bias="0.832" />
</androidx.constraintlayout.widget.ConstraintLayout>
```

Java Code-

MainActivity.java

```
package com.example.madex5;
import androidx.appcompat.app.AppCompatActivity;
import android.media.MediaPlayer;
import android.os.Bundle;
import android.os.Handler;
import android.view.View;
import android.widget.ImageButton;
import android.widget.SeekBar;
import android.widget.TextView;
import android.widget.Toast;
import java.util.concurrent.TimeUnit;

public class MainActivity extends AppCompatActivity {
    private ImageButton f_btn,b_btn,play_btn,pause_btn;
    private MediaPlayer mplayer;
    private TextView song_name, current_time, song_time;
    private SeekBar song_progress;
    private static int otime=0, stime=0, etime=0, ftime=5000, btime=5000;
    private Handler hdlr=new Handler();
    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_main);
        f_btn=findViewById(R.id.fast_forward);
        b_btn=findViewById(R.id.fast_reverse);
        play_btn=findViewById(R.id.play_btn);
        pause_btn=findViewById(R.id.pause_btn);
        song_name=findViewById(R.id.s_name);
        current_time=findViewById(R.id.current_time);
        song_time=findViewById(R.id.song_time);
        song_name.setText("Music.mp3");
        mplayer=MediaPlayer.create(this,R.raw.music);
        song_progress=findViewById(R.id.seekBar);
        song_progress.setClickable(false);
        pause_btn.setClickable(false);
        play_btn.setOnClickListener(new View.OnClickListener(){

            @Override
            public void onClick(View view) {
                Toast.makeText(MainActivity.this,"Playing Audio", Toast.LENGTH_SHORT).show();
                mplayer.start();
                etime=mplayer.getDuration();
                stime=mplayer.getCurrentPosition();
                if (otime==0){
                    song_progress.setMax(etime);
                    otime=1;
                }
                song_time.setText(String.format("%d min %d sec",
TimeUnit.MILLISECONDS.toMinutes(etime),TimeUnit.MILLISECONDS.toSeconds(etime)-
TimeUnit.MINUTES.toSeconds(TimeUnit.MILLISECONDS.toMinutes(etime))));
                current_time.setText(String.format("%d min %d sec",
TimeUnit.MILLISECONDS.toMinutes(stime),TimeUnit.MILLISECONDS.toSeconds(stime)-
TimeUnit.MINUTES.toSeconds(TimeUnit.MINUTES.toMinutes(stime))));
                song_progress.setProgress(stime);
                hdlr.postDelayed(UpdateSongTime, 100);
                pause_btn.setEnabled(true);
                play_btn.setEnabled(false);
            }
        });
        pause_btn.setOnClickListener(new View.OnClickListener(){

            @Override
            public void onClick(View view) {
                mplayer.pause();
                pause_btn.setEnabled(false);
                play_btn.setEnabled(true);
                Toast.makeText(getApplicationContext(),"Audio
Paused",Toast.LENGTH_SHORT).show();
            }
        });
        f_btn.setOnClickListener(new View.OnClickListener(){

            @Override
            public void onClick(View view) {
                if ((stime+ftime)<=etime){
                    stime=stime+ftime;
                    mplayer.seekTo(stime);
                }
                else{
                    Toast.makeText(getApplicationContext(),"Forward Jumping is not
Possible",Toast.LENGTH_SHORT).show();
                }
                if(!play_btn.isEnabled()){
                    play_btn.setEnabled(true);
                }
            }
        });
        b_btn.setOnClickListener(new View.OnClickListener(){
            @Override
            public void onClick(View view) {
                if((stime-btime)>0){
                    stime=stime-btime;
                    mplayer.seekTo(stime);
                }
                else{
                    Toast.makeText(getApplicationContext(),"Backward Jumpimg is Not
Possible",Toast.LENGTH_SHORT).show();
                }
                if (!play_btn.isEnabled()){
                    play_btn.setEnabled(true);
                }
            }
        });
    }
    private Runnable UpdateSongTime=new Runnable(){

        @Override
        public void run() {
            stime=mplayer.getCurrentPosition();
            current_time.setText(String.format("%d min %d
sec",TimeUnit.MILLISECONDS.toMinutes(stime),TimeUnit.MILLISECONDS.toSeconds(stime)-
TimeUnit.MINUTES.toSeconds(TimeUnit.MILLISECONDS.toMinutes(stime))));
            song_progress.setProgress(stime);
            hdlr.postDelayed(this,100);
        }
    };
}
```

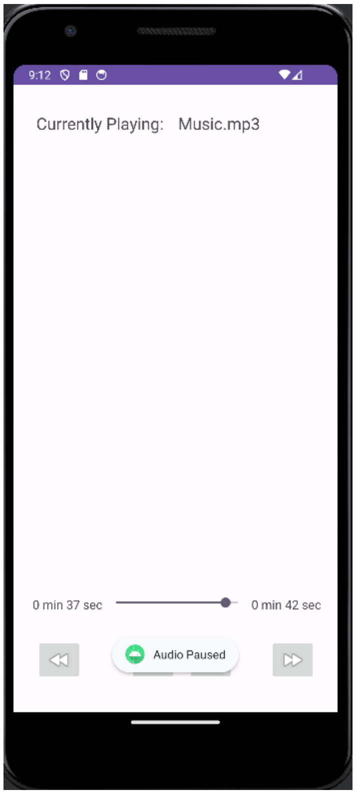
OUTPUT:



Currently Playing: Music.mp3

0 min 3 sec 0 min 42 sec

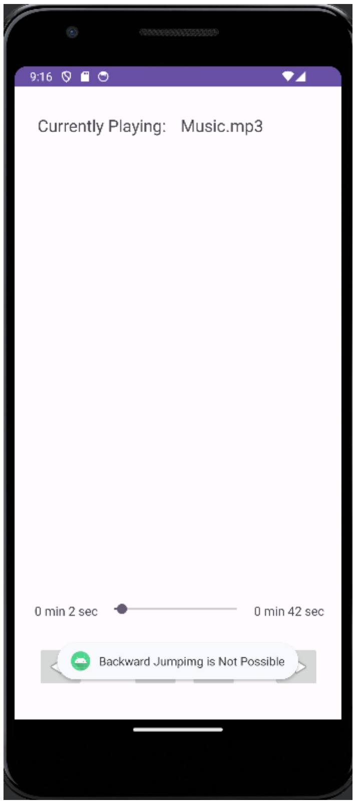
Playing Audio



Currently Playing: Music.mp3

0 min 37 sec 0 min 42 sec

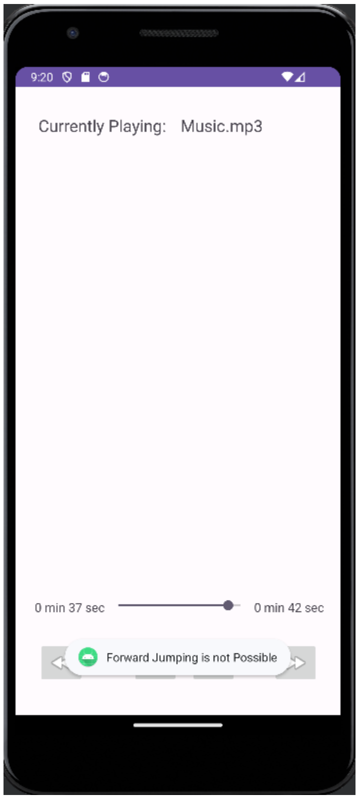
Audio Paused



Currently Playing: Music.mp3

0 min 2 sec 0 min 42 sec

Backward Jumpimg is Not Possible



Currently Playing: Music.mp3

0 min 37 sec 0 min 42 sec

Forward Jumping is not Possible

RESULT:

Thus, the experiment for creating a Media Player Application has been coded, compiled, and executed successfully.

Ex No: 6	CREATE A SMS APPLICATION
Date:	19/10/2023

AIM:

To create a SMS Application.

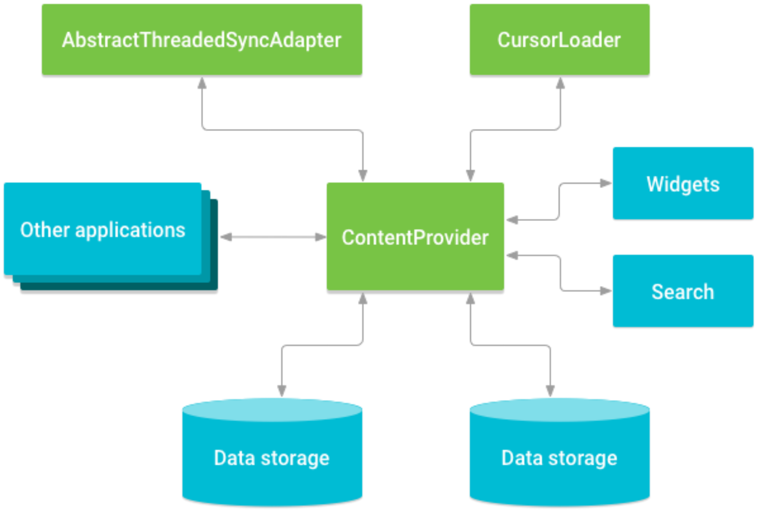
DESCRIPTION:

A Content provider manages access to a central repository of data. A provider is part of an Android application, which often provides its own UI for working with the data. However, content providers are primarily used by other applications, which access the provider using a provider client object. Together, providers and provider clients offer a consistent, standard interface to data that also handles interprocess communication and secure data access.

Cursor is a Interface whice returns collection of your query data. moveToFirst() is used to point the cursor position from where you want to get data from your cursor. There are methods moveToLast(), moveToNext(), moveToPrevious(), moveToPosition(position) by which you can iterate through your cursor by desired way.

In Android, you can use SmsManager API or devices Built-in SMS application to send SMS's. The permissions for the same are set:

<uses-permission android:name="android.permission.SEND_SMS" />



PROGRAM:

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

```
xmlns:app="http://schemas.android.com/apk/res-auto"
android:layout_width="match_parent"
android:layout_height="match_parent"
tools:context=".MainActivity">

<TextView
   android:id="@+id/txtVw1"
   android:layout_width="134dp"
   android:layout_height="48dp"
   android:gravity="center"
   android:text="@string/compose_to"
   android:textAppearance="?android:attr/textAppearanceMedium"
   android:textSize="18sp"
   app:layout_constraintBottom_toBottomOf="parent"
   app:layout_constraintEnd_toEndOf="parent"
   app:layout_constraintHorizontal_bias="0.059"
   app:layout_constraintStart_toStartOf="parent"
   app:layout_constraintTop_toTopOf="parent"
   app:layout_constraintVertical_bias="0.081" />

<EditText
   android:id="@+id/Cname"
   android:layout_width="195dp"
   android:layout_height="50dp"
   android:layout_marginEnd="72dp"
   android:autofillHints=""
   android:ems="10"
   android:hint="@string/contact_no"
   android:inputType="text"
   app:layout_constraintBottom_toBottomOf="parent"
   app:layout_constraintEnd_toEndOf="parent"
   app:layout_constraintTop_toTopOf="parent"
   app:layout_constraintVertical_bias="0.082" />

<ImageButton
   android:id="@+id/contactbook"
   android:layout_width="58dp"
   android:layout_height="51dp"
   android:contentDescription="@string/contact_book"
   app:layout_constraintBottom_toBottomOf="parent"
   app:layout_constraintEnd_toEndOf="parent"
```

```
        app:layout_constraintHorizontal_bias="0.954"
        app:layout_constraintStart_toStartOf="parent"
        app:layout_constraintTop_toTopOf="parent"
        app:layout_constraintVertical_bias="0.082"
        app:srcCompat="@android:drawable/sym_contact_card" />

    <EditText
        android:id="@+id/sms"
        android:layout_width="290dp"
        android:layout_height="48dp"
        android:autofillHints=""
        android:ems="10"
        android:hint="@string/message"
        android:inputType="text"
        app:layout_constraintBottom_toBottomOf="parent"
        app:layout_constraintEnd_toEndOf="parent"
        app:layout_constraintHorizontal_bias="0.231"
        app:layout_constraintStart_toStartOf="parent"
        app:layout_constraintTop_toTopOf="parent"
        app:layout_constraintVertical_bias="0.538" />

    <ImageButton
        android:id="@+id/send"
        android:layout_width="57dp"
        android:layout_height="49dp"
        android:contentDescription="@string/send"
        app:layout_constraintBottom_toBottomOf="parent"
        app:layout_constraintEnd_toEndOf="parent"
        app:layout_constraintHorizontal_bias="0.898"
        app:layout_constraintStart_toStartOf="parent"
        app:layout_constraintTop_toTopOf="parent"
        app:layout_constraintVertical_bias="0.538"
        app:srcCompat="@android:drawable/ic_menu_send" />


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

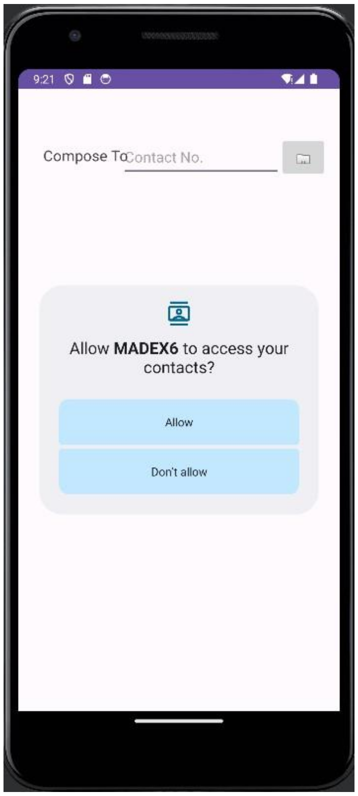
Java Code-

MainActivity.java

```
package com.example.madex6;
```

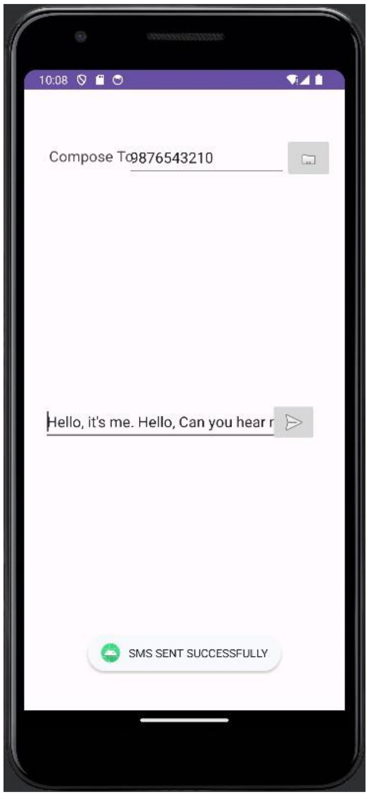
```
import androidx.annotation.NonNull;
import androidx.annotation.Nullable;
import androidx.appcompat.app.AppCompatActivity;
import androidx.core.app.ActivityCompat;
import androidx.core.content.ContextCompat;

import android.Manifest;
import android.content.Intent;
import android.content.pm.PackageManager;
import android.database.Cursor;
import android.net.Uri;
import android.os.Bundle;
import android.provider.ContactsContract;
import android.telephony.SmsManager;
import android.view.View;
import android.widget.EditText;
import android.widget.ImageButton;
import android.widget.TextView;
import android.widget.Toast;

public class MainActivity extends AppCompatActivity {
    EditText msg_send, cn;
    TextView tv;
    ImageButton open, send_msg;
    private static final int CONTACT_PERMISSION_CODE = 1;
    private static final int CONTACT_PICK_CODE = 2;
    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_main);
        cn = findViewById(R.id.Cname);
        msg_send = findViewById(R.id.sms);
        send_msg = findViewById(R.id.send);
        tv = findViewById(R.id.txtVw1);
        open = findViewById(R.id.contactbook);
        open.setOnClickListener(new View.OnClickListener() {
            @Override
            public void onClick(View view) {
                if(checkContactPermission()){
                    pickContactIntent();
                }
                else{
                    requestContactPermission();
                }
            }
        });
        send_msg.setOnClickListener(new View.OnClickListener(){
            @Override
            public void onClick(View view) {
                if (ContextCompat.checkSelfPermission(MainActivity.this,
android.Manifest.permission.SEND_SMS)
                == PackageManager.PERMISSION_GRANTED){
                    sendMessage();
                }
                else{
                    ActivityCompat.requestPermissions(MainActivity.this, new
String[]{android.Manifest.permission.SEND_SMS}, 100);
                }
            }
        });
    }
    private void sendMessage(){
        String phoneno = cn.getText().toString().trim();
        String message = msg_send.getText().toString().trim();
        if(!phoneno.equals("") && !message.equals("")){
            SmsManager smsManager = SmsManager.getDefault();
            smsManager.sendTextMessage(phoneno, null, message, null, null);
            Toast.makeText(this, "SMS SENT SUCCESSFULLY", Toast.LENGTH_LONG).show();
        }
        else{
            Toast.makeText(this, "Type Some Message", Toast.LENGTH_LONG).show();
        }
    }
    private boolean checkContactPermission(){
        boolean result = ContextCompat.checkSelfPermission(
                this,
                android.Manifest.permission.READ_CONTACTS) ==
(PackageManager.PERMISSION_GRANTED
        );
        return result;
    }
    private void requestContactPermission(){
        String[] permissions = {Manifest.permission.READ_CONTACTS};
        ActivityCompat.requestPermissions(this, permissions, CONTACT_PERMISSION_CODE);
    }

    private void pickContactIntent(){
        Intent intent = new Intent(Intent.ACTION_PICK,
ContactsContract.CommonDataKinds.Phone.CONTENT_URI);
        startActivityForResult(intent, CONTACT_PICK_CODE);
    }

    public void onRequestPermissionResult(int requestCode, @NonNull String[] permissions,
@NonNull int[] grantResults){
        super.onRequestPermissionsResult(requestCode, permissions, grantResults);
        if(requestCode == CONTACT_PERMISSION_CODE){
            if(grantResults.length>0 &&
grantResults[0]==PackageManager.PERMISSION_GRANTED){
                pickContactIntent();
            }
            else{
                Toast.makeText(this, "Permission Denied", Toast.LENGTH_SHORT).show();
            }
        }

        if(requestCode == 100 && grantResults.length>0 && grantResults[0] ==
PackageManager.PERMISSION_GRANTED){
            sendMessage();
        }
        else{
            Toast.makeText(this, "Permission Denied", Toast.LENGTH_SHORT).show();
        }
    }
    protected void onActivityResult(int requestCode, int resultCode, @Nullable Intent data){
        super.onActivityResult(requestCode, resultCode, data);
        if(resultCode == RESULT_OK){
            switch (requestCode){
                case CONTACT_PICK_CODE:
                    contactPicked(data);
                    break;
            }
        }
        else{
            Toast.makeText(this, "Failed to Pick Contact", Toast.LENGTH_SHORT).show();
        }
    }

    private void contactPicked(Intent data){
        Cursor cursor = null;

        try{
            String phoneNo = null;
            Uri uri = data.getData();
            cursor = getContentResolver().query(uri, null, null, null, null);
            cursor.moveToFirst();
            int phoneIndex =
cursor.getColumnIndex(ContactsContract.CommonDataKinds.Phone.NUMBER);
            phoneNo = cursor.getString(phoneIndex);
            cn.setText(phoneNo);
        }
        catch(Exception e){
            e.printStackTrace();
        }
    }
}
```

OUTPUT:



Compose To Contact No.

Allow **MADEX6** to access your contacts?



Compose To 9876543210

Hello, it's me. Hello, Can you hear r

SMS SENT SUCCESSFULLY

RESULT:

Thus, the experiment for creating a SMS Application has been coded, compiled, and executed successfully.

Ex No: 7	DOWNLOAD IMAGE USING ASYNCHRONOUS
Date:	26/10/2023

AIM:

To develop an application for downloading image asynchronously using Android Studio.

DESCRIPTION:

The Asynchronous Image Download Application in Android Studio is a user-friendly tool for efficiently downloading images from the internet while maintaining a responsive and seamless user experience. At its core, this application leverages the AsyncTask class, a pivotal component in Android development, to carry out image downloads asynchronously. The AsyncTask class provides several key methods, including doInBackground(), which is used to perform the actual image download from a specified URL in a background thread. This crucial feature ensures that time-consuming network operations do not block the main UI thread, preventing the app from becoming unresponsive.

In addition to doInBackground(), the application also takes advantage of other AsyncTask methods such as onPreExecute(), which allows for any necessary initialization or the display of loading indicators before the background task starts. The onProgressUpdate() method is utilized to update the user interface with progress information during the download process, making it possible to implement features like progress bars. Finally, onPostExecute() plays a crucial role in managing the downloaded image, handling any potential errors, or executing post-processing tasks. Together, these elements work in harmony to create a powerful and efficient image downloading application for Android users.

PROGRAM:

MainActivity.java:

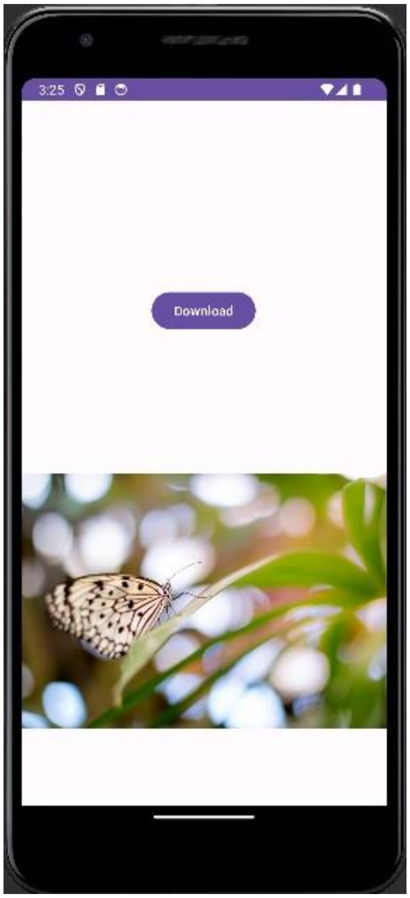
```
package com.example.madex7;
import androidx.appcompat.app.AppCompatActivity;
import android.app.ProgressDialog;
import android.graphics.Bitmap;
import android.graphics.BitmapFactory;
import android.os.AsyncTask;
import android.os.Bundle;
import android.view.View;
import android.widget.Button;
import android.widget.ImageView;
import java.io.IOException;
import java.io.InputStream;
import java.net.HttpURLConnection;
import java.net.URL;
```

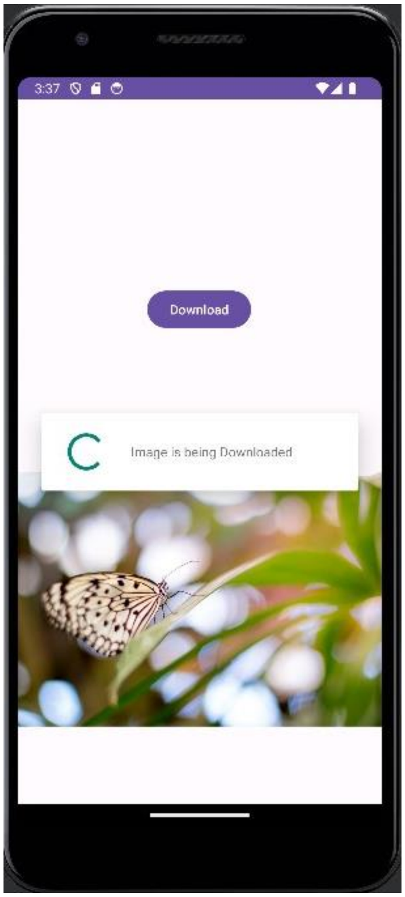
```
public class MainActivity extends AppCompatActivity {
    URL ImageUrl = null;
    InputStream is = null;
    Bitmap bmImg = null;
    ImageView image;
    ProgressDialog p;
    Button download;
    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_main);
        download = findViewById(R.id.button);
        image = findViewById(R.id.imageView);
        download.setOnClickListener(new View.OnClickListener() {
            @Override
            public void onClick(View view) {
                AsyncTaskExample asyncTask = new AsyncTaskExample();
                asyncTask.execute("https://p0.pxfuel.com/preview/875/609/139/butterfly-flower-nature-
plant.jpg");
            }});}
    private class AsyncTaskExample extends AsyncTask<String, String, Bitmap>{
        @Override
        protected void onPreExecute(){
            super.onPreExecute();
            p = new ProgressDialog(MainActivity.this);
            p.setMessage("Image is being Downloaded");
            p.setIndeterminate(false);
            p.setCancelable(false);
            p.show();
        }
        @Override
        protected Bitmap doInBackground(String... strings){
            try{
                ImageUrl = new URL(strings[0]);
                HttpURLConnection conn = (HttpURLConnection) ImageUrl.openConnection();
                conn.setDoInput(true);
                conn.connect();
                is = conn.getInputStream();
                BitmapFactory.Options options = new BitmapFactory.Options();
                options.inPreferredConfig = Bitmap.Config.RGB_565;
                bmImg = BitmapFactory.decodeStream(is, null, options);
            }catch(IOException e){
                e.printStackTrace();
            }
            return bmImg;
        }
        @Override
        protected void onPostExecute(Bitmap bitmap) {
            super.onPostExecute(bitmap);
            if(image != null){
                p.hide();
                image.setImageBitmap(bitmap);
            }
            else{
                p.show();
            }         }     }
}
```

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">
    <Button
        android:id="@+id/button"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="@string/download"
        app:layout_constraintBottom_toBottomOf="parent"
        app:layout_constraintEnd_toEndOf="parent"
        app:layout_constraintHorizontal_bias="0.498"
        app:layout_constraintStart_toStartOf="parent"
        app:layout_constraintTop_toTopOf="parent"
        app:layout_constraintVertical_bias="0.284" />

    <ImageView
        android:id="@+id/imageView"
        android:layout_width="410dp"
        android:layout_height="441dp"
        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="1.0"
        app:srcCompat="@drawable/butterfly"
        android:contentDescription="@string/butterfly" />
</androidx.constraintlayout.widget.ConstraintLayout>
```

OUTPUT:





RESULT:

Thus, the application for downloading image asynchronously using Android studio has been executed successfully and the output has been verified.

Ex No: 8	SWIPE VIEW IN TAB USING VIEW PAGES
Date:	2/11/2023

AIM:

To develop an application with Swipe view in tab with the help of View Page using Android Studio.

DESCRIPTION:

The Swipe View in Tab Using View Pager application in Android Studio offers an intuitive navigation experience. It employs the ViewPager component to enable users to smoothly swipe between various content or pages within distinct tabs. This user-friendly interface combines ViewPager and TabLayout to organize and display content, with each tab representing a different fragment or page. The seamless integration of fragments, an adapter, and a page change listener ensures efficient navigation, making it easy for users to access diverse types of content through horizontal swiping within tabs.

Key Components:

1. ViewPager: ViewPager is a critical component in Android development for creating swipeable interfaces. It allows users to transition between different pages or fragments with a smooth horizontal swipe gesture. In this application, ViewPager is used to manage and display content within tabs.
2. TabLayout: TabLayout is used in conjunction with ViewPager to create tabbed navigation at the top or bottom of the screen, making it easy for users to select the content they want to view. Each tab corresponds to a different fragment or page.
3. Fragments: The application uses fragments to represent the individual content pages that can be swiped between. Each fragment can contain different types of content, such as text, images, or interactive elements.
4. Adapter: An adapter, often a FragmentPagerAdapter or FragmentStatePagerAdapter, is used to manage the fragments displayed within the ViewPager. It provides the necessary data and behavior to efficiently switch between fragments as users swipe through the tabs.
5. Page Change Listener: To respond to user interactions, a page change listener can be implemented to detect when the user swipes to a different tab. This listener allows for custom actions or updates

PROGRAM:

MainActivity.java:

```
package com.example.madex8;

import androidx.annotation.NonNull;
import androidx.appcompat.app.AppCompatActivity;
import androidx.fragment.app.Fragment;
import androidx.viewpager2.widget.ViewPager2;
import android.os.Bundle;
import com.google.android.material.tabs.TabLayout;
import com.google.android.material.tabs.TabLayoutMediator;
import java.util.ArrayList;

public class MainActivity extends AppCompatActivity implements
TabLayoutMediator.TabConfigurationStrategy{
    ViewPager2 viewPager2;
    TabLayout tabLayout;
    ArrayList<String> tabTitles;
    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_main);
        viewPager2 = findViewById(R.id.viewPagerId);
        tabLayout = findViewById(R.id.tabLayout);
        ViewPager2Adapter viewPager2Adapter = new ViewPager2Adapter(this);
        ArrayList<Fragment> fragments = new ArrayList<>();
        fragments.add(new FirstFragment());
        fragments.add(new SecondFragment());
        fragments.add(new ThirdFragment());
        viewPager2Adapter.setFragments(fragments);
        viewPager2.setAdapter(viewPager2Adapter);
        tabTitles = new ArrayList<>();
        tabTitles.add("First");
        tabTitles.add("Second");
        tabTitles.add("Third");
        new TabLayoutMediator(tabLayout, viewPager2, this).attach();
    }
    public void onConfigureTab(@NonNull TabLayout.Tab tab, int position){
        tab.setText(tabTitles.get(position));
    }
}
```

FirstFragment.java:

```
package com.example.madex8;
import androidx.fragment.app.Fragment;
import android.os.Bundle;
import android.view.LayoutInflater;
import android.view.View;
import android.view.ViewGroup;
```

```
public class FirstFragment extends Fragment {
   @Override
   public View onCreateView(LayoutInflater inflater, ViewGroup container, Bundle
savedInstanceState) {
       return inflater.inflate(R.layout.fragment_first, container, false);
   }
}
```

SecondFragment.java:

```
package com.example.madex8;
import androidx.fragment.app.Fragment;
import android.os.Bundle;
import android.view.LayoutInflater;
import android.view.View;
import android.view.ViewGroup;

public class SecondFragment extends Fragment {
   @Override
   public View onCreateView(LayoutInflater inflater, ViewGroup container, Bundle
savedInstanceState) {
       return inflater.inflate(R.layout.fragment_second, container, false);
   }
}
```

ThirdFragment.java:

```
package com.example.madex8;
import androidx.fragment.app.Fragment;
import android.os.Bundle;
import android.view.LayoutInflater;
import android.view.View;
import android.view.ViewGroup;

public class ThirdFragment extends Fragment {
   @Override
   public View onCreateView(LayoutInflater inflater, ViewGroup container, Bundle
savedInstanceState) {
       return inflater.inflate(R.layout.fragment_third, container, false);
   }
}
```

ViewPager2Adapter.java:

```
package com.example.madex8;
import androidx.annotation.NonNull;
import androidx.fragment.app.Fragment;
import androidx.fragment.app.FragmentActivity;
import androidx.viewpager2.adapter.FragmentStateAdapter;
import java.util.ArrayList;

public class ViewPager2Adapter extends FragmentStateAdapter {
    private ArrayList<Fragment> fragments;
    public void setFragments(ArrayList<Fragment> fragments){this.fragments = fragments;}
    public ViewPager2Adapter(@NonNull FragmentActivity fragmentActivity){
        super(fragmentActivity);
    }
    @NonNull
    @Override
    public androidx.fragment.app.Fragment createFragment(int position) {return
fragments.get(position);}
    public int getItemCount(){return fragments.size();}
}
```

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">
    <com.google.android.material.tabs.TabLayout
        android:id="@+id/tabLayout"
        android:layout_width="409dp"
        android:layout_height="wrap_content"
        android:layout_marginStart="1dp"
        android:layout_marginTop="2dp"
        android:layout_marginEnd="1dp"
        android:layout_marginBottom="681dp"
        app:layout_constraintBottom_toBottomOf="parent"
        app:layout_constraintEnd_toEndOf="parent"
        app:layout_constraintStart_toStartOf="parent"
        app:layout_constraintTop_toTopOf="parent"/>
    <androidx.viewpager2.widget.ViewPager2
        android:id="@+id/viewPagerId"
        android:layout_width="409dp"
        android:layout_height="679dp"
        android:layout_marginStart="1dp"
        android:layout_marginTop="1dp"
        android:layout_marginEnd="1dp"
        android:layout_marginBottom="1dp"
        app:layout_constraintBottom_toBottomOf="parent"
        app:layout_constraintEnd_toEndOf="parent"
        app:layout_constraintStart_toStartOf="parent"
        app:layout_constraintTop_toBottomOf="@+id/tabLayout" />
</androidx.constraintlayout.widget.ConstraintLayout>
```

fragment_first.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=".FirstFragment">
    <TextView
        android:layout_width="match_parent"
        android:layout_height="match_parent"
        android:text="@string/firstfragment"
        app:layout_constraintBottom_toBottomOf="parent"
        app:layout_constraintEnd_toEndOf="parent"
        app:layout_constraintStart_toStartOf="parent"
        app:layout_constraintTop_toTopOf="parent" />
</androidx.constraintlayout.widget.ConstraintLayout>
```

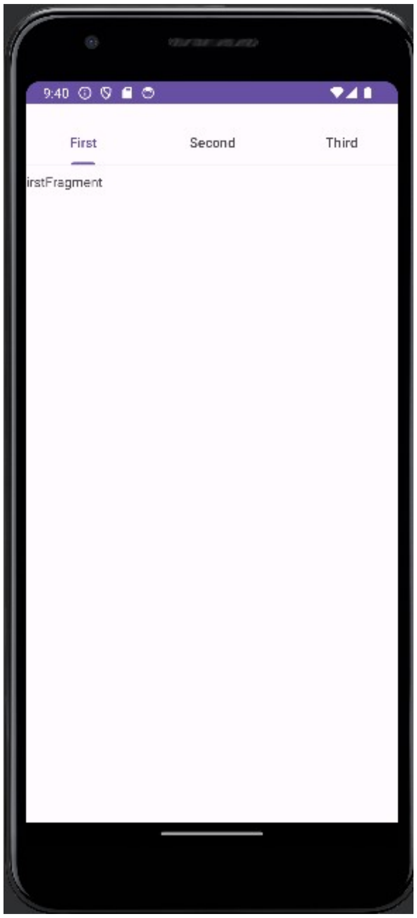
fragment_second.xml:

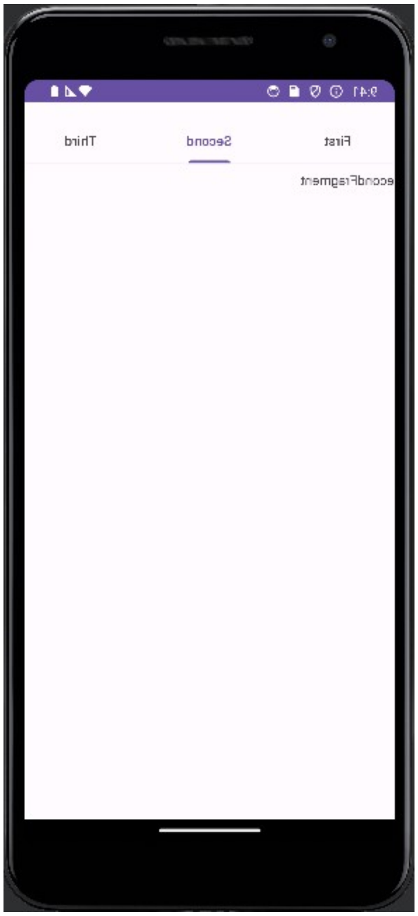
```
<?xml version="1.0" encoding="utf-8"?>
<androidx.constraintlayout.widget.ConstraintLayout
    xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:app="http://schemas.android.com/apk/res-auto"
    xmlns:tools="http://schemas.android.com/tools"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    tools:context=".SecondFragment">
    <TextView
        android:layout_width="match_parent"
        android:layout_height="match_parent"
        android:text="@string/secondfragment"
        app:layout_constraintBottom_toBottomOf="parent"
        app:layout_constraintEnd_toEndOf="parent"
        app:layout_constraintStart_toStartOf="parent"
        app:layout_constraintTop_toTopOf="parent" />
</androidx.constraintlayout.widget.ConstraintLayout>
```

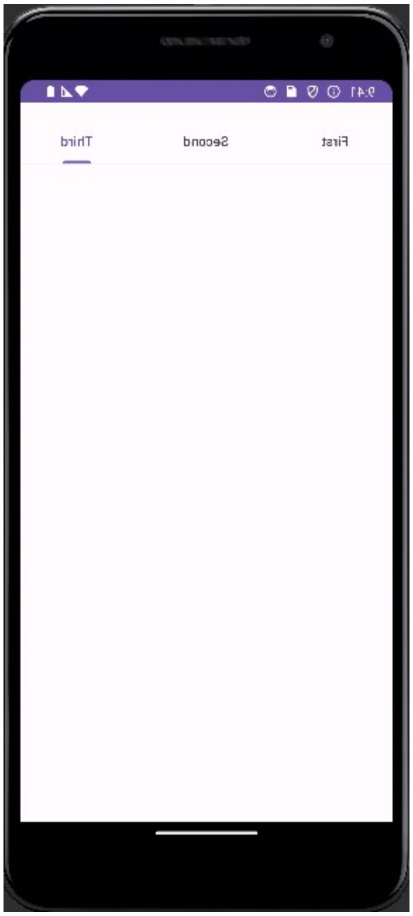
fragment_third.xml:

```
<?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=".ThirdFragment">
    <androidx.recyclerview.widget.RecyclerView
        android:id="@+id/recyclerView"
        android:layout_width="409dp"
        android:layout_height="729dp"
        android:layout_marginStart="1dp"
        android:layout_marginTop="1dp"
        android:layout_marginEnd="1dp"
        android:layout_marginBottom="1dp"
        app:layout_constraintBottom_toBottomOf="parent"
        app:layout_constraintEnd_toEndOf="parent"
        app:layout_constraintStart_toStartOf="parent"
        app:layout_constraintTop_toTopOf="parent" />
</androidx.constraintlayout.widget.ConstraintLayout>
```

OUTPUT:


9:40
First
Second
Third
irstFragment


9:41
Third
Second
First
econdFragment



RESULT:

Thus, the application with swipe view in tab using view pages with the help of Android studio has been executed successfully and the output has been verified.

Ex No: 9	USING ANDROID ML KIT TO EXTRACT TEXT FROM IMAGE
Date:	9/11/2023

AIM:

To develop an application with android ML Kit to extract text from image using android studio.

DESCRIPTION:

The Text Extraction from Image application in Android utilizes the ML Vision framework to extract text from images efficiently. It employs the `FirebaseVisionImage` class to process images, `FirebaseVisionTextRecognizer` to initiate text recognition, and asynchronous callbacks to handle the results. The extracted text and its metadata are encapsulated within the `FirebaseVisionText` class, providing a user-friendly solution for accurate and seamless text extraction from images.

Key Components:

1. ML Vision Text Recognition: ML Vision's Text Recognition API is the core component used for detecting and extracting text from images. It employs machine learning models to recognize text and its structure within the image.

2. FirebaseVisionImage: To process images, the application utilizes the FirebaseVisionImage class to convert the image data into a format compatible with ML Vision's Text Recognition.

3. FirebaseVisionTextRecognizer: The FirebaseVisionTextRecognizer class is responsible for initializing and executing the text recognition process. It's used to analyze the image and extract the textual content.

4. Callbacks: Asynchronous processing is crucial, and the application utilizes callbacks to handle the results from the text recognition process. Callbacks like addOnSuccessListener and addOnFailureListener are used to manage successful text extraction or errors.

5. FirebaseVisionText: This class encapsulates the results of text recognition, including information about the detected text, its position in the image, and any associated metadata.

PROGRAM:

<u>MainActivity.java:</u>

```
package com.example.madex9;

import 79ndroid.annotation.NonNull;
import 79ndroid.annotation.Nullable;
import 79ndroid.appcompat.app.AppCompatActivity;
import android.Manifest;
import android.content.Intent;
import android.content.pm.PackageManager;
import android.graphics.Bitmap;
import android.os.Bundle;
import android.provider.MediaStore;
import android.view.View;
import android.widget.ImageView;
import android.widget.TextView;
import android.widget.Toast;
import com.google.android.gms.tasks.OnFailureListener;
import com.google.android.gms.tasks.OnSuccessListener;
import com.google.android.gms.tasks.Task;
import com.google.firebase.ml.vision.FirebaseVision;
import com.google.firebase.ml.vision.common.FirebaseVisionImage;
import com.google.firebase.ml.vision.text.FirebaseVisionText;
import com.google.firebase.ml.vision.text.FirebaseVisionTextRecognizer;

public class MainActivity extends AppCompatActivity {
    ImageView imageView;
    TextView textView;
    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_main);
        imageView = findViewById(R.id.imageId);
        textView = findViewById(R.id.textId);
        if(checkSelfPermission(android.Manifest.permission.CAMERA)!=
PackageManager.PERMISSION_GRANTED){
            requestPermissions(new String[]{Manifest.permission.CAMERA},101);
        }
    }
    public void doProcess(View view){
        Intent intent = new Intent(MediaStore.ACTION_IMAGE_CAPTURE);
        startActivityForResult(intent, 101);
    }
    @Override
    protected void onActivityResult(int requestCode, int resultCode, @Nullable Intent data){
        super.onActivityResult(requestCode, resultCode, data);
        Bundle bundle = data.getExtras();
        Bitmap bitmap = (Bitmap) bundle.get(“data”);
        imageView.setImageBitmap(bitmap);
        FirebaseVisionImage firebaseVisionImage = FirebaseVisionImage.fromBitmap(bitmap);
        FirebaseVision firebaseVision = FirebaseVision.getInstance();
        FirebaseVisionTextRecognizer firebaseVisionTextRecognizer =
firebaseVision.getOnDeviceTextRecognizer();
        Task<FirebaseVisionText> task =
firebaseVisionTextRecognizer.processImage(firebaseVisionImage);
        task.addOnSuccessListener(new OnSuccessListener<FirebaseVisionText>() {
            @Override
            public void onSuccess(FirebaseVisionText firebaseVisionText) {
                String s = firebaseVisionText.getText();
                textView.setText(s);
            }
        });
        task.addOnFailureListener(new OnFailureListener() {
            @Override
            public void onFailure(@NonNull Exception e) {
                Toast.makeText(getApplicationContext(), e.getMessage(),
Toast.LENGTH_LONG).show();
            }
        });
    }
}
```

activity_main.xml:

```
<?xml version=”1.0” encoding=”utf-8”?>
<81ndroid.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:id=”@+id/imageId”
        android:layout_width=”411dp”
        android:layout_height=”297dp”
        android:contentDescription=”@string/image”
        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” />
    <TextView
        android:id=”@+id/textId”
        android:layout_width=”393dp”
        android:layout_height=”57dp”
        android:layout_marginStart=”11dp”
        android:layout_marginTop=”25dp”
        android:layout_marginEnd=”11dp”
        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=”0.493” />
    <Button
        android:id=”@+id/buttonId”
        android:layout_width=”match_parent”
        android:layout_height=”wrap_content”
        android:onClick=”doProcess”
        android:text=”@string/click_me”
        app:layout_constraintBottom_toBottomOf=”parent”
        app:layout_constraintEnd_toEndOf=”parent”
        app:layout_constraintHorizontal_bias=”1.0”
        app:layout_constraintStart_toStartOf=”parent”
        app:layout_constraintTop_toTopOf=”parent”
        app:layout_constraintVertical_bias=”0.434” />
</82ndroid.constraintlayout.widget.ConstraintLayout>
```

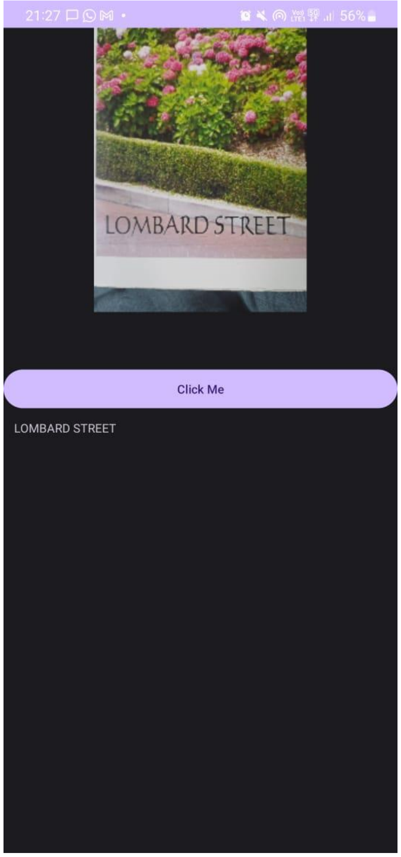
build.gradle (:app):

```
dependencies {

    implementation ‘androidx.appcompat:appcompat:1.6.1’
    implementation ‘com.google.android.material:material:1.10.0’
    implementation ‘androidx.constraintlayout:constraintlayout:2.1.4’
    implementation ‘com.google.firebase:firebase-ml-vision:24.1.0’
    implementation ‘com.google.android.gms:play-services-vision:20.1.3’
    testImplementation ‘junit:junit:4.13.2’
    androidTestImplementation ‘androidx.test.ext:junit:1.1.5’
    androidTestImplementation ‘androidx.test.espresso:espresso-core:3.5.1’
}
```

AndroidManifest.xml:

```
<uses-feature
        android:name=”android.hardware.camera”
        android:required=”false”/>
    <uses-permission android:name=”android.permission.CAMERA”/>
```

OUTPUT:



LOMBARD STREET

RESULT:

Thus, the application that extracts text from images has been created with the help of Android studio and the output has been verified.

Ex No: 10	**GETTING CURRENT LOCATION USING GEOCODER**
Date:	16/11/2023

AIM:

To develop an application that gets the current location with the help of geocoder using Android Studio.

DESCRIPTION:

The Getting Current Location Using Geocoder application in Android Studio employs the Geocoder class to efficiently retrieve and convert the device's geographic coordinates into human-readable location information. Leveraging the LocationManager and LocationListener,it continuously monitors and updates the device's location, while managing permissions to ensure compliance with Android's security standards. The resulting location data is seamlessly integrated into the application's user interface, enhancing the user experience for location-basedservices.

Key Components:

1. Geocoder Class: The Geocoder class is a fundamental component in this application, used to translate geographic coordinates into human-readable location data, such as the city, country, and address.
2. LocationManager: To access the device's current location, the LocationManager class is employed. It facilitates location updates and provides access to the device's GPS or network-based location data.
3. LocationListener: A LocationListener is used to monitor changes in the device's location. It triggers events when the location is updated, allowing the application to react dynamically to location changes.
4. Permissions: To access the device's location information, the application requests and manages the necessary permissions, ensuring that it adheres to Android's security protocols.
5. UI Integration: The application typically integrates this information into its userinterface, displaying the current location details in a user-friendly format.

PROGRAM:

Main Activity.java

```
package com.example.madex10;
import static android.Manifest.permission.ACCESS_FINE_LOCATION;
import androidx.annotation.NonNull;
    import androidx.appcompat.app.AppCompatActivity;
    import androidx.core.app.ActivityCompat;
    import androidx.core.content.ContextCompat;
    import android.content.Context;
    import android.content.pm.PackageManager;
    import android.location.Address;
    import android.location.Geocoder;
    import android.location.Location;
    import  android.location.LocationListener;
    import android.location.LocationManager;
    import android.os.Bundle;
    import android.util.Log;
    import android.view.View;
    import android.widget.Button;
    import android.widget.TextView;
    import android.widget.Toast;
    import java.util.List;
    import java.util.Locale;

    public class MainActivity extends AppCompatActivity implements LocationListener{
      Button btnShowLocation;
      LocationManager locationManager;
      private double latitude;
      private double longitude;
      TextView edit_Country,edit_State,edit_City,edit_Pincode;
      @Override
      protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_main);
        edit_Country = findViewById(R.id.editCountry);
        edit_State=findViewById(R.id.editState);
        edit_City=findViewById(R.id.editCity);
        edit_Pincode=findViewById(R.id.editPincode);
        btnShowLocation=findViewById(R.id.btnShowLocation);
        btnShowLocation.setOnClickListener(new View.OnClickListener() {
            @Override
            public void onClick(View v) {
                if (ContextCompat.checkSelfPermission(MainActivity.this,
ACCESS_FINE_LOCATION)!= PackageManager.PERMISSION_GRANTED){
                    ActivityCompat.requestPermissions(MainActivity.this,new
String[]{ACCESS_FINE_LOCATION},1);
                }else{
                    detectCurrentLocation();
                }
            }
        });
    }
    private void detectCurrentLocation(){
        Toast.makeText(this,"Getting your current location", Toast.LENGTH_SHORT).show();
        locationManager=(LocationManager)
getSystemService(Context.LOCATION_SERVICE);
        if (ActivityCompat.checkSelfPermission(this, ACCESS_FINE_LOCATION)
!=PackageManager.PERMISSION_GRANTED &&
ActivityCompat.checkSelfPermission()){
            return;
        }
        locationManager.requestLocationUpdates(LocationManager.GPS_PROVIDER,0,0,this);
    }
    @Override
    public void onLocationChanged(Location location){
```

```
latitude = location.getLatitude();
```

```
    longitude = location.getLongitude();
    findAddress();
}
private void findAddress(){
    Geocoder geocoder;
    List<Address> addresses;
    geocoder = new Geocoder(this, Locale.getDefault());
    try{
        addresses= geocoder.getFromLocation(latitude,longitude,1);
        String country = addresses.get(0).getCountryName();
        String state = addresses.get(0).getAdminArea();
        String city = addresses.get(0).getLocality();
        String Pincode = addresses.get(0).getPostalCode();
        edit_Country.setText(country);
        edit_State.setText(state);
        edit_City.setText(city);
        edit_Pincode.setText(Pincode);
        Log.d("City",city);
        Log.d("State",state);
        Log.d("Country",country);
        Log.d("Pincode",String.valueOf(Pincode));
    }catch(Exception e){
        Toast.makeText(this,""+e.getMessage(),Toast.LENGTH_SHORT).show();
    }
}
@Override
public void onStatusChanged(String provider, int status, Bundle extras){
}
@Override
public void onProviderEnabled(String provider){
    }
    @Override
    public void onProviderDisabled(String provider){
        Toast.makeText(this,"Please turn on Location",Toast.LENGTH_SHORT).show();
    }
    @Override
    public void onRequestPermissionsResult(int requestCode,@NonNull String[]
permissions,@NonNull int[] grantResults){
        super.onRequestPermissionsResult(requestCode,permissions,grantResults);
        if(requestCode == 1){
            if(grantResults.length > 0 && grantResults[0] ==
PackageManager.PERMISSION_GRANTED){
                detectCurrentLocation();
            }else{
                Toast.makeText(this,"Permission Denied",Toast.LENGTH_SHORT).show();
            }
        }
    }
}
```

Main_activity.xml:

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

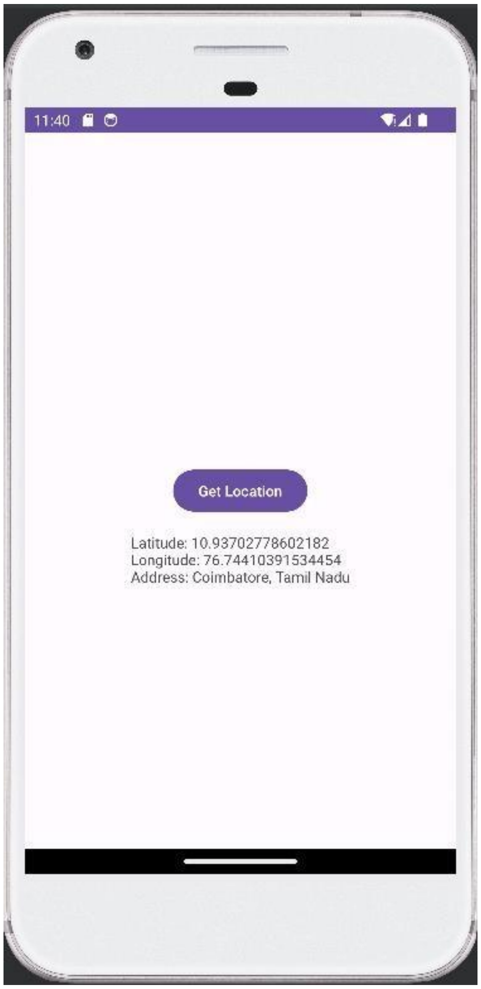
```
android:orientation="vertical">

<TextView
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:hint="Country"
    android:textColor="#000"
    android:textSize="20sp"
    android:id="@+id/editCountry"/>

<TextView
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:hint="State"
    android:textColor="#000"
    android:textSize="20sp"
    android:id="@+id/editState" />

<TextView
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:hint="City"
    android:textColor="#000"
    android:textSize="20sp"
    android:id="@+id/editCity"/>

<TextView
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:hint="Pincode"
```

```
            android:textColor="#000"
            android:textSize="20sp"
            android:id="@+id/editPincode"/>
        <Button
            android:layout_width="match_parent"
            android:layout_height="wrap_content"
            android:text="Get Location"
            android:id="@+id/btnShowLocation"/>
    </LinearLayout>
</RelativeLayout>
```

Manifest.xml:

```
<?xml version="1.0" encoding="utf-8"?>
<manifest xmlns:android="http://schemas.android.com/apk/res/android"
    package="com.example.ex10">
    <uses-permission android:name="android.permission.INTERNET"/>
    <uses-permission android:name="android.permission.ACCESS_FINE_LOCATION"/>
    <uses-permission android:name="android.permission.ACCESS_COARSE_LOCATION"/>
    <application
        android:allowBackup="true"
        android:icon="@mipmap/ic_launcher"
        android:label="ex10"
        android:roundIcon="@mipmap/ic_launcher_round"
        android:supportsRtl="true"
        android:theme="@style/Theme.Ex10">
        <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>
```

OUTPUT:



Latitude: 10.93702778602182
Longitude: 76.74410391534454
Address: Coimbatore, Tamil Nadu

RESULT:

Thus, the application which gets the current location using geocoder with the help of Android studio has been executed successfully and the output has been verified.