PRACTICAL: 1

AIM:

Introduction to Android and Create “Custom Message” application. That will display “Custom Message” in the middle of the screen in the Black color with the Yellow background.

THEORY:

Android apps can be written using Kotlin, Java, and C++ languages. The Android SDK tools compile your code along with any data and resource files into an APK, an *Android package*, which is an archive file with an .apk suffix. One APK file contains all the contents of an Android app and is the file that Android-powered devices use to install the app.

CODE:

MainActivity.java

```
package com.example.wcmc_pr1_17it015;

import androidx.appcompat.app.AppCompatActivity;

import android.os.Bundle;
import android.view.View;

public class MainActivity extends AppCompatActivity {
   View v1;
   @Override
   protected void onCreate(Bundle savedInstanceState) {
      super.onCreate(savedInstanceState);
      setContentView(R.layout.activity_main);

      v1 = this.getWindow().getDecorView();
      v1.setBackgroundResource(R.color.yellow);
   }
}
```

AndroidManifest.xml

```
<?xml version="1.0" encoding="utf-8"?>
<manifest xmlns:android="http://schemas.android.com/apk/res/android"
   package="com.example.wcmc_pr1_17it015">

   <application
      android:allowBackup="true"
      android:icon="@mipmap/ic_launcher"
      android:label="@string/app_name"
      android:roundIcon="@mipmap/ic_launcher_round"
      android:supportsRtl="true"
      android:theme="@style/AppTheme">
      <activity android:name=".MainActivity">
         <intent-filter>
```

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

            <category android:name="android.intent.category.LAUNCHER" />
        </intent-filter>
    </activity>
</application>

</manifest>
```

activity_main.xml

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

    <TextView
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="Kishan_17IT015"
        app:layout_constraintBottom_toBottomOf="parent"
        app:layout_constraintLeft_toLeftOf="parent"
        app:layout_constraintRight_toRightOf="parent"
        app:layout_constraintTop_toTopOf="parent"
        android:textSize="30sp"
        android:textColor="#000000"/>

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

Colors.xml

```
<?xml version="1.0" encoding="utf-8"?>
<resources>
    <color name="colorPrimary">#008577</color>
    <color name="colorPrimaryDark">#00574B</color>
    <color name="colorAccent">#D81B60</color>
    <color name="yellow">#ffff45</color>
</resources>
```

OUTPUT:

LATEST APPLICATIONS:

Clip layer, Clipboard Action, Native Clipboard, Clip Stack, Copy Bubble etc.

LEARNING OUTCOME:

We can learn how to change the background color in the application adding color in color.xml and add custom message in the app with black color.

PRACTICAL: 2

AIM:

Create an android application to calculate sum of two numbers and gives result in Toast Message.

THEORY:

TextView: TextView is a user interface control which is used to set and display the text to the user based on our requirements.In android, we can create a TextView control in two ways either in XML layout file or create it in Activity file programmatically.
Attributes: android:id, android:capitalize, android:cursorVisible, android:editable, android:gravity etc.

EditText: In Android, EditText is a standard entry widget in android apps. It is an overlay over TextView that configures itself to be editable. EditText is a subclass of TextView with text editing operations. We often use EditText in our applications in order to provide an input or text field, especially in forms. The most simple example of EditText is Login or Sign-in form.
Attributes: android:id, android:capitalize, android:cursorVisible, android:editable, android:gravity etc.

Button: In Android, Button represents a push button. A Push buttons can be clicked or pressed by the user to perform an action. There are different types of buttons used in android such as CompoundButton, ToggleButton, RadioButton. Button is a subclass of TextView class and compound button is the subclass of Button class.

Toast Message: A toast provides simple feedback about an operation in a small popup. It only fills the amount of space required for the message and the current activity remains visible and interactive. Toasts automatically disappear after a timeout.
The android.widget.Toast class is the subclass of java.lang.Object class.

CODE:

MainActivity.java

```
package com.example.wcmc_pr2_17it015;

import androidx.appcompat.app.AppCompatActivity;

import android.os.Bundle;
import android.view.View;
import android.widget.Button;
import android.widget.EditText;
import android.widget.Toast;

public class MainActivity extends AppCompatActivity {
    Button b1 ;
    EditText et1 ;
    EditText et2 ;
    View v1;
```

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

        v1 = this.getWindow().getDecorView();
        v1.setBackgroundResource(R.color.background);

         b1 = findViewById(R.id.b1);
         et1 = findViewById(R.id.et1);
         et2 = findViewById(R.id.et2);

        b1.setOnClickListener(new View.OnClickListener() {
            @Override
            public void onClick(View v) {
                int n1  = Integer.parseInt(et1.getText().toString());
                int n2 = Integer.parseInt((et2.getText().toString()));
                n1 = n1+n2;
                Toast.makeText(getApplicationContext(),"sum:
"+Integer.toString(n1),Toast.LENGTH_LONG).show();
            }
        });
    }
}
```

AndroidManifest.xml

```
<?xml version="1.0" encoding="utf-8"?>
<manifest xmlns:android="http://schemas.android.com/apk/res/android"
    package="com.example.wcmc_pr2_17it015">

    <application
        android:allowBackup="true"
        android:icon="@mipmap/ic_launcher"
        android:label="@string/app_name"
        android:roundIcon="@mipmap/ic_launcher_round"
        android:supportsRtl="true"
        android:theme="@style/AppTheme">
        <activity android:name=".MainActivity">
            <intent-filter>
                <action android:name="android.intent.action.MAIN" />

                <category android:name="android.intent.category.LAUNCHER" />
            </intent-filter>
        </activity>
    </application>
</manifest>
```

activity_main.xml

```
<?xml version="1.0" encoding="utf-8"?>
<androidx.constraintlayout.widget.ConstraintLayout
    xmlns:android="http://schemas.android.com/apk/res/android"
```

```
xmlns:app="http://schemas.android.com/apk/res-auto"
xmlns:tools="http://schemas.android.com/tools"
android:layout_width="match_parent"
android:layout_height="match_parent"
android:id="@+id/p1"
tools:context=".MainActivity">

<TextView
   android:id="@+id/textView0"
   android:layout_width="wrap_content"
   android:layout_height="wrap_content"
   android:layout_marginStart="112dp"
   android:layout_marginLeft="112dp"
   android:layout_marginTop="32dp"
   android:layout_marginEnd="117dp"
   android:layout_marginRight="117dp"
   android:text="Kishan_17IT015"
   android:textSize="25sp"
   android:textColor="#000000"
   app:layout_constraintEnd_toEndOf="parent"
   app:layout_constraintStart_toStartOf="parent"
   app:layout_constraintTop_toTopOf="parent"
   tools:ignore="DuplicateIds,MissingConstraints" />

<TextView
   android:id="@+id/textView1"
   android:layout_width="wrap_content"
   android:layout_height="wrap_content"
   android:layout_marginStart="37dp"
   android:layout_marginLeft="37dp"
   android:layout_marginTop="41dp"
   android:layout_marginEnd="152dp"
   android:layout_marginRight="152dp"
   android:text="Enter First Number: "
   android:textSize="25sp"
   app:layout_constraintEnd_toEndOf="parent"
   app:layout_constraintStart_toStartOf="parent"
   app:layout_constraintTop_toBottomOf="@+id/textView0"
   tools:ignore="MissingConstraints" />

<TextView
   android:id="@+id/textView2"
   android:layout_width="wrap_content"
   android:layout_height="wrap_content"
   android:layout_below="@id/et1"
   android:layout_marginStart="37dp"
   android:layout_marginLeft="37dp"
   android:layout_marginTop="42dp"
   android:layout_marginEnd="117dp"
   android:layout_marginRight="117dp"
   android:text="Enter Second Number: "
```

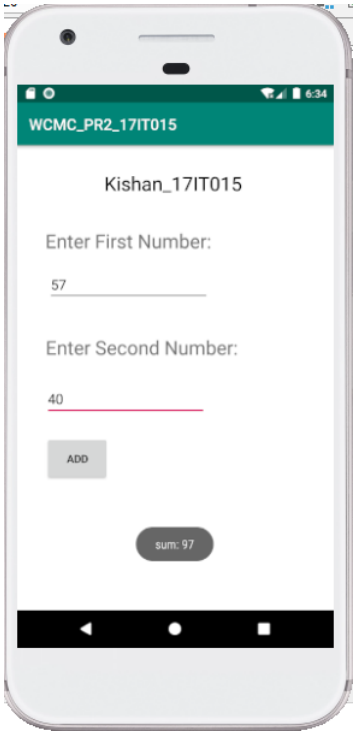
```
    android:textSize="25sp"
    app:layout_constraintEnd_toEndOf="parent"
    app:layout_constraintStart_toStartOf="parent"
    app:layout_constraintTop_toBottomOf="@+id/et1"
    tools:ignore="MissingConstraints" />

<EditText
    android:id="@+id/et1"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:layout_below="@id/textView1"
    android:layout_marginStart="37dp"
    android:layout_marginLeft="37dp"
    android:layout_marginTop="18dp"
    android:layout_marginEnd="161dp"
    android:layout_marginRight="161dp"
    android:ems="10"
    android:inputType="text"
    app:layout_constraintEnd_toEndOf="parent"
    app:layout_constraintHorizontal_bias="1.0"
    app:layout_constraintStart_toStartOf="parent"
    app:layout_constraintTop_toBottomOf="@+id/textView1"
    tools:ignore="MissingConstraints" />

<EditText
    android:id="@+id/et2"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:layout_below="@id/textView2"
    android:layout_marginStart="37dp"
    android:layout_marginLeft="37dp"
    android:layout_marginTop="28dp"
    android:layout_marginEnd="161dp"
    android:layout_marginRight="161dp"
    android:ems="10"
    android:inputType="text"
    app:layout_constraintEnd_toEndOf="parent"
    app:layout_constraintHorizontal_bias="0.0"
    app:layout_constraintStart_toStartOf="parent"
    app:layout_constraintTop_toBottomOf="@+id/textView2"
    tools:ignore="MissingConstraints" />

<Button
    android:id="@+id/b1"
    android:layout_width="83dp"
    android:layout_height="59dp"
    android:layout_below="@id/et2"
    android:layout_marginStart="37dp"
    android:layout_marginLeft="37dp"
    android:layout_marginTop="25dp"
    android:layout_marginEnd="291dp"
```

```
        android:layout_marginRight="291dp"
        android:text="add"
        app:layout_constraintEnd_toEndOf="parent"
        app:layout_constraintStart_toStartOf="parent"
        app:layout_constraintTop_toBottomOf="@+id/et2"
        tools:ignore="MissingConstraints" />

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

Colors.xml

```
<?xml version="1.0" encoding="utf-8"?>
<resources>
    <color name="colorPrimary">#008577</color>
    <color name="colorPrimaryDark">#00574B</color>
    <color name="colorAccent">#D81B60</color>
    <color name="background">#B9AFF5E3</color>
</resources>
```

OUTPUT:



LATEST APPLICATIONS:

G-Mail, Google Drive, Google Classroom, Paytm etc.

LEARNING OUTCOME:

In above application we can create a functionality in which user can enter two numbers and then click on ADD button it will display sum of that number in toast message for to develop this app we use TextView, EditText, Button and Toast Message functionalities of the android.

PRACTICAL 3

AIM: Create an application that will display Toast (Message) on specific interval of time.

THEORY:

Android CountDownTimer class is used to schedule a countdown until a time in the future defined by the user, with regular notifications on intervals along the way. This class is an abstract class whose methods need to be overridden to implement it in our project.

CODE:

// MainActivity.java

```
package com.MyApplication;

import android.os.CountDownTimer;
import android.support.v7.app.AppCompatActivity;
import android.os.Bundle;
import android.widget.Toast;

public class MainActivity extends AppCompatActivity {

   @Override
   protected void onCreate(Bundle savedInstanceState) {
      super.onCreate(savedInstanceState);
      setContentView(R.layout.activity_main);
      new CountDownTimer(50000,5000){

         @Override
         public void onTick(long l) {
            Toast.makeText(getApplicationContext(),”17it015!!",Toast.LENGTH_SHORT).show();
         }
         @Override
         public void onFinish() {}
      }.start();
   }
}
```

// activity_main.xml

```
<?xml version="1.0" encoding="utf-8"?>
<android.support.constraint.ConstraintLayout
xmlns:android="http://schemas.android.com/apk/res/android"
   xmlns:app="http://schemas.android.com/apk/res-auto"
   xmlns:tools="http://schemas.android.com/tools"
   android:layout_width="match_parent"
   android:layout_height="match_parent"
   android:background="#000"
   tools:context=".MainActivity">
```

```
   <TextView
      android:layout_width="wrap_content"
      android:layout_height="wrap_content"
      android:text="Hello World"
      android:textSize="60sp"
      android:textColor="##66ff66"
      app:layout_constraintBottom_toBottomOf="parent"
      app:layout_constraintLeft_toLeftOf="parent"
      app:layout_constraintRight_toRightOf="parent"
      app:layout_constraintTop_toTopOf="parent" />

</android.support.constraint.ConstraintLayout>
```

// **AndroidManifest.xml**

```
<?xml version="1.0" encoding="utf-8"?>
<manifest xmlns:android="http://schemas.android.com/apk/res/android"
   package="com.17IT022_practical3">

   <application
     android:allowBackup="true"
     android:icon="@mipmap/ic_launcher"
     android:label="@string/app_name"
     android:roundIcon="@mipmap/ic_launcher_round"
     android:supportsRtl="true"
     android:theme="@style/AppTheme">
     <activity android:name=".MainActivity">
       <intent-filter>
         <action android:name="android.intent.action.MAIN" />

         <category android:name="android.intent.category.LAUNCHER" />
       </intent-filter>
     </activity>
  </application>

</manifest>
```

OUTPUT:

Here we have selected the timer of 50 seconds so the toast will display for 50 seconds and the interval we use is 5 seconds.

LATEST APPLICATIONS:

Stop watches, Instagram etc.

LEARNING OUTCOME:

In this practical we learn how to display toast message in specific interval of time.

PRACTICAL 4

AIM: Create a temperature converter Application. (Fahrenheit-Celsius).

THEORY:

Android CountDownTimer class is used to schedule a countdown until a time in the future defined by the user, with regular notifications on intervals along the way. This class is an abstract class whose methods need to be overridden to implement it in our project.

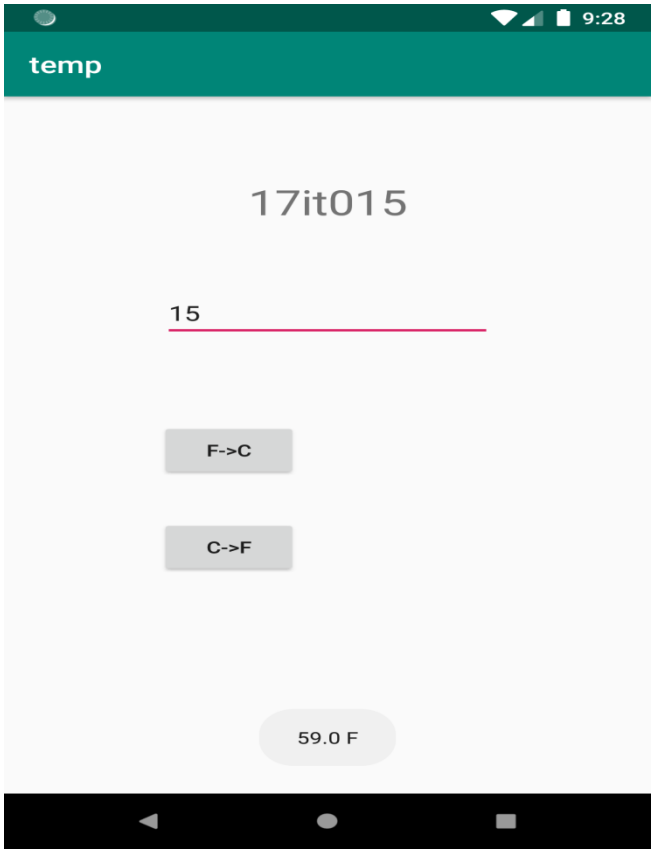
CODE:

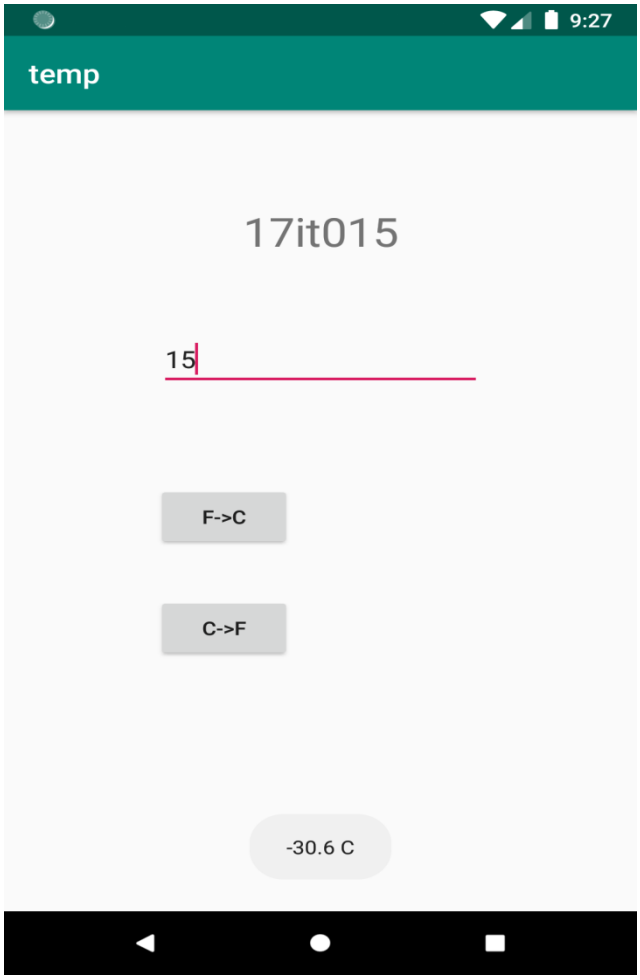
```
// MainActivity.java
package com.example.temp;
import androidx.appcompat.app.AppCompatActivity;
import android.os.Bundle;
import android.view.View;
import android.widget.Button;
import android.widget.EditText;
import android.widget.Toast;
public class MainActivity extends AppCompatActivity {
   EditText temp;
   Button fc;
   Button cf;
   Double d;
   @Override
   protected void onCreate(Bundle savedInstanceState) {
      super.onCreate(savedInstanceState);
      setContentView(R.layout.activity_main);
      temp=(EditText)findViewById(R.id.editText);
      fc=(Button)findViewById(R.id.button);
      cf=(Button)findViewById(R.id.button2);
      fc.setOnClickListener(new View.OnClickListener() {
         @Override
         public void onClick(View v) {
            d=Double.parseDouble(String.valueOf(temp.getText()));
            Double d1=d-32.0;
            Double d2= d1*(9.0/5.0);
            String s=String.valueOf(d2);
            Toast.makeText(MainActivity.this,s+" C",Toast.LENGTH_LONG).show();
         }
      });
      cf.setOnClickListener(new View.OnClickListener() {
         @Override
         public void onClick(View v) {
            d=Double.parseDouble(String.valueOf(temp.getText()));
            Double d3=d*(9.0/5.0)+32.0;
            String s=String.valueOf(d3);
            Toast.makeText(MainActivity.this,s+" F",Toast.LENGTH_LONG).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">
    <TextView
        android:id="@+id/textView"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="17it015"
        android:textSize="30sp"
        app:layout_constraintBottom_toBottomOf="parent"
        app:layout_constraintLeft_toLeftOf="parent"
        app:layout_constraintRight_toRightOf="parent"
        app:layout_constraintTop_toTopOf="parent"
        app:layout_constraintVertical_bias="0.126" />
    <EditText
        android:id="@+id/editText"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:ems="10"
        android:hint="temperature"
        android:inputType="textPersonName"
        app:layout_constraintBottom_toBottomOf="parent"
        app:layout_constraintEnd_toEndOf="parent"
        app:layout_constraintStart_toStartOf="parent"
        app:layout_constraintTop_toBottomOf="@+id/textView"
        app:layout_constraintVertical_bias="0.121" />
    <Button
        android:id="@+id/button"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_marginTop="72dp"
        android:text="F->C"
        app:layout_constraintBottom_toBottomOf="parent"
        app:layout_constraintEnd_toEndOf="parent"
        app:layout_constraintHorizontal_bias="0.306"
        app:layout_constraintStart_toStartOf="parent"
        app:layout_constraintTop_toBottomOf="@+id/editText"
        app:layout_constraintVertical_bias="0.0" />
    <Button
        android:id="@+id/button2"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="C->F"
        app:layout_constraintBottom_toBottomOf="parent"
```

```
app:layout_constraintEnd_toEndOf="parent"
        app:layout_constraintHorizontal_bias="0.306"
        app:layout_constraintStart_toStartOf="parent"
        app:layout_constraintTop_toBottomOf="@+id/button"
        app:layout_constraintVertical_bias="0.158" />
</androidx.constraintlayout.widget.ConstraintLayout>
// AndroidManifest.xml
<?xml version="1.0" encoding="utf-8"?>
<manifest xmlns:android="http://schemas.android.com/apk/res/android"
    package="com.example.temp">
    <application
        android:allowBackup="true"
        android:icon="@mipmap/ic_launcher"
        android:label="@string/app_name"
        android:roundIcon="@mipmap/ic_launcher_round"
        android:supportsRtl="true"
        android:theme="@style/AppTheme">
        <activity android:name=".MainActivity">
            <intent-filter>
                <action android:name="android.intent.action.MAIN" />
                <category android:name="android.intent.category.LAUNCHER" />
            </intent-filter>
        </activity>
    </application>
</manifest>
```

OUTPUT:


9:28
temp
17it015
15
F->C
C->F
59.0 F


9:27
temp
17it015
15
F->C
C->F
-30.6 C

LATEST APPLICATIONS:

AccuWeather, Dark Sky-Hyperlocal Weather etc.

LEARNING OUTCOME:

In this practical we learn use of buttons and toast a message here we have created a application to convert temperature from celcius to faranheit and viceversa.

PRACTICAL 5

AIM: Create a login application with following features: 1. Successful Login message in TextView with Green background if Username & password is correct 2. Failure message in TextView with Red background if Username or password is incorrect. 3. Disable Login Button after three wrong login attempts. 4. Close application if user selects Cancel Button.

THEORY:

TextView: In Android, TextView displays text to the user and optionally allows them to edit it programmatically. TextView is a complete text editor, however basic class is configured to not allow editing but we can edit it. View is the parent class of TextView. Being a subclass of view the text view component can be used in your app's GUI inside a ViewGroup, or as the content view of an activity. Attributes: android:id, android:capitalize, android:cursorVisible, android:editable, android:gravity etc.

EditText: In Android, EditText is a standard entry widget in android apps. It is an overlay over TextView that configures itself to be editable. EditText is a subclass of TextView with text editing operations. We often use EditText in our applications in order to provide an input or text field, especially in forms. The most simple example of EditText is Login or Sign-in form. Attributes: android:id, android:capitalize, android:cursorVisible, android:editable, android:gravity etc.

Button: In Android, Button represents a push button. A Push buttons can be clicked or pressed by the user to perform an action. There are different types of buttons used in android such as CompoundButton, ToggleButton, RadioButton. Button is a subclass of TextView class and compound button is the subclass of Button class.

Toast Message: A toast provides simple feedback about an operation in a small popup. It only fills the amount of space required for the message and the current activity remains visible and interactive. Toasts automatically disappear after a timeout. The android.widget.Toast class is the subclass of java.lang.Object class.

CODE:

```
// MainActivity.java
package com.example.a17it015_login;

import androidx.appcompat.app.AppCompatActivity;

import android.graphics.Color;
import android.os.AsyncTask;
import android.os.Bundle;
import android.view.View;
import android.widget.Button;
import android.widget.EditText;
import android.widget.TextView;
import android.widget.Toast;

public class MainActivity extends AppCompatActivity {
    EditText user1,pass,left;
```

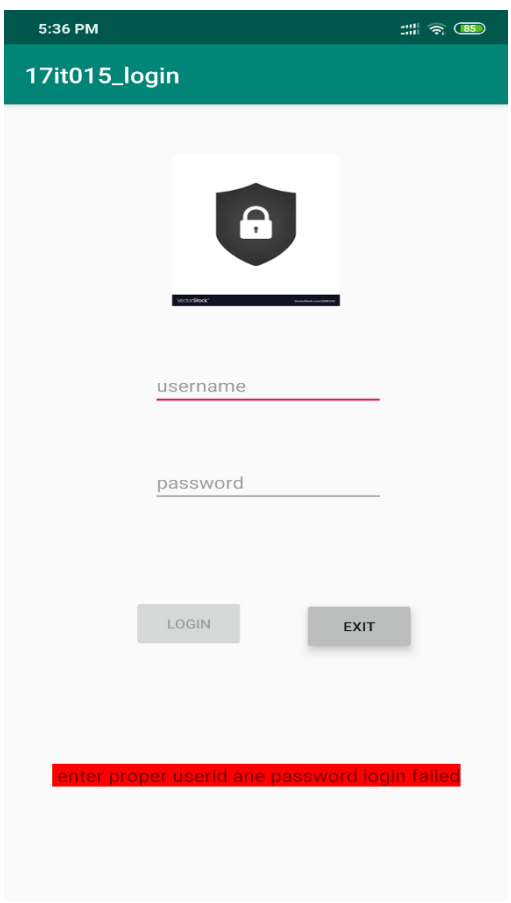
```
    Button b1,b2;
    TextView t1;
    int count=3;
    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_main);
        user1=findViewById(R.id.editText);
        pass=findViewById(R.id.editText3);
        t1=findViewById(R.id.result);
        b1=findViewById(R.id.button);
        b2=findViewById(R.id.button2);
        b1.setOnClickListener(new View.OnClickListener() {
            @Override
            public void onClick(View v) {
                String user = user1.getText().toString();
                String pass1=pass.getText().toString();
                if(user.equals("admin") && pass1.equals("password") ){
                    t1.setText("Successful login");
                    t1.setBackgroundColor(Color.GREEN);
                }
                else {
                    count--;
                    t1.setText(" enter proper userid ane password login failed");
                    t1.setBackgroundColor(Color.RED);
                    String cnt=String.valueOf(count);
                    Toast.makeText(MainActivity.this,cnt+"Attempt
Remaining",Toast.LENGTH_LONG).show();
                    if(count==0)
                    {
                        b1.setEnabled(false);
                    }
                }

            }
        });
        b2.setOnClickListener(new View.OnClickListener() {
            @Override
            public void onClick(View v) {
                finish();
                System.exit(0);
            }
        });
    }
}
// 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"
```

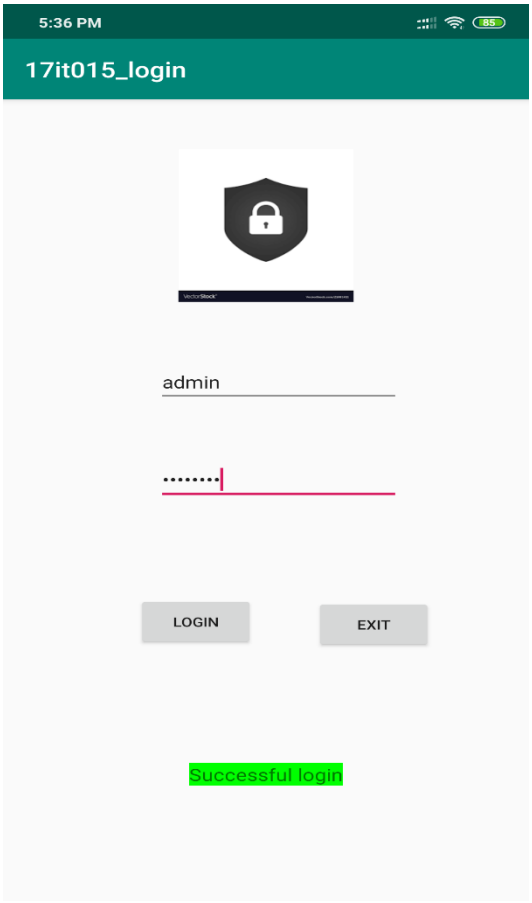
```
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    android:background="#00353131"
    tools:context=".MainActivity">

    <EditText
        android:id="@+id/editText"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:ems="10"
        android:hint="username"
        android:inputType="textPersonName"
        app:layout_constraintBottom_toBottomOf="parent"
        app:layout_constraintEnd_toEndOf="parent"
        app:layout_constraintHorizontal_bias="0.545"
        app:layout_constraintStart_toStartOf="parent"
        app:layout_constraintTop_toTopOf="parent"
        app:layout_constraintVertical_bias="0.344" />

    <EditText
        android:id="@+id/editText3"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:ems="10"
        android:hint="password"
        android:inputType="textPassword"
        app:layout_constraintBottom_toBottomOf="parent"
        app:layout_constraintEnd_toEndOf="parent"
        app:layout_constraintHorizontal_bias="0.545"
        app:layout_constraintStart_toStartOf="parent"
        app:layout_constraintTop_toBottomOf="@+id/editText"
        app:layout_constraintVertical_bias="0.116" />

    <Button
        android:id="@+id/button"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_marginTop="80dp"
        android:layout_marginEnd="76dp"
        android:layout_marginRight="76dp"
        android:text="login"
        app:layout_constraintBottom_toBottomOf="parent"
        app:layout_constraintEnd_toEndOf="parent"
        app:layout_constraintHorizontal_bias="0.437"
        app:layout_constraintStart_toStartOf="parent"
        app:layout_constraintTop_toBottomOf="@+id/editText3"
        app:layout_constraintVertical_bias="0.026" />

    <ImageView
        android:id="@+id/imageView5"
        android:layout_width="181dp"
```

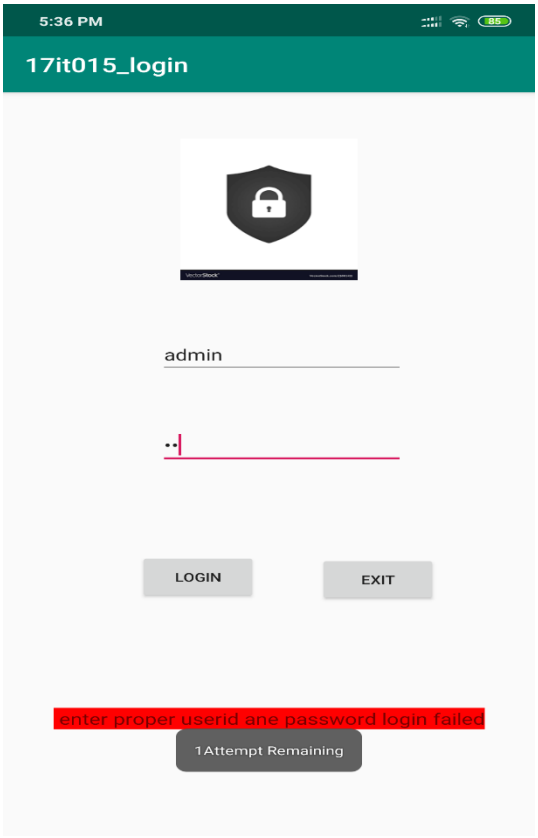
```
        android:layout_height="141dp"
        app:layout_constraintBottom_toTopOf="@+id/editText"
        app:layout_constraintEnd_toEndOf="parent"
        app:layout_constraintStart_toStartOf="parent"
        app:layout_constraintTop_toTopOf="parent"
        app:layout_constraintVertical_bias="0.463"
        app:srcCompat="@drawable/lock" />

    <TextView
        android:id="@+id/result"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:textSize="18sp"
        app:layout_constraintBottom_toBottomOf="parent"
        app:layout_constraintEnd_toEndOf="parent"
        app:layout_constraintStart_toStartOf="parent"
        app:layout_constraintTop_toBottomOf="@+id/button" />

    <Button
        android:id="@+id/button2"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="exit"
        app:layout_constraintBottom_toBottomOf="parent"
        app:layout_constraintEnd_toEndOf="parent"
        app:layout_constraintHorizontal_bias="0.764"
        app:layout_constraintStart_toStartOf="parent"
        app:layout_constraintTop_toBottomOf="@+id/editText3"
        app:layout_constraintVertical_bias="0.277" />

</androidx.constraintlayout.widget.ConstraintLayout>
// AndroidManifest.xml
<?xml version="1.0" encoding="utf-8"?>
<manifest xmlns:android="http://schemas.android.com/apk/res/android"
    package="com.example.a17it015_login">

    <application
        android:allowBackup="true"
        android:icon="@mipmap/ic_launcher"
        android:label="@string/app_name"
        android:roundIcon="@mipmap/ic_launcher_round"
        android:supportsRtl="true"
        android:theme="@style/AppTheme">
        <activity android:name=".MainActivity">
            <intent-filter>
                <action android:name="android.intent.action.MAIN" />

                <category android:name="android.intent.category.LAUNCHER" />
            </intent-filter>
        </activity>
    </application>
```

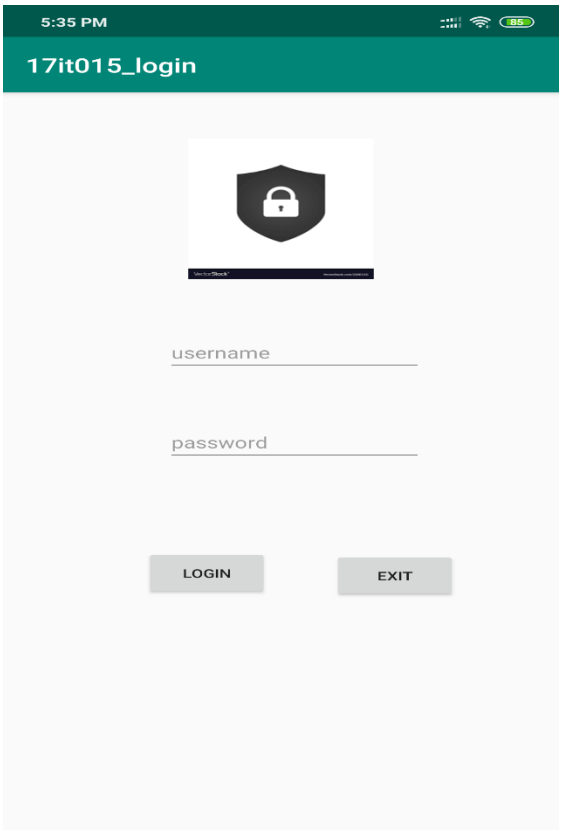
```
</manifest>
```

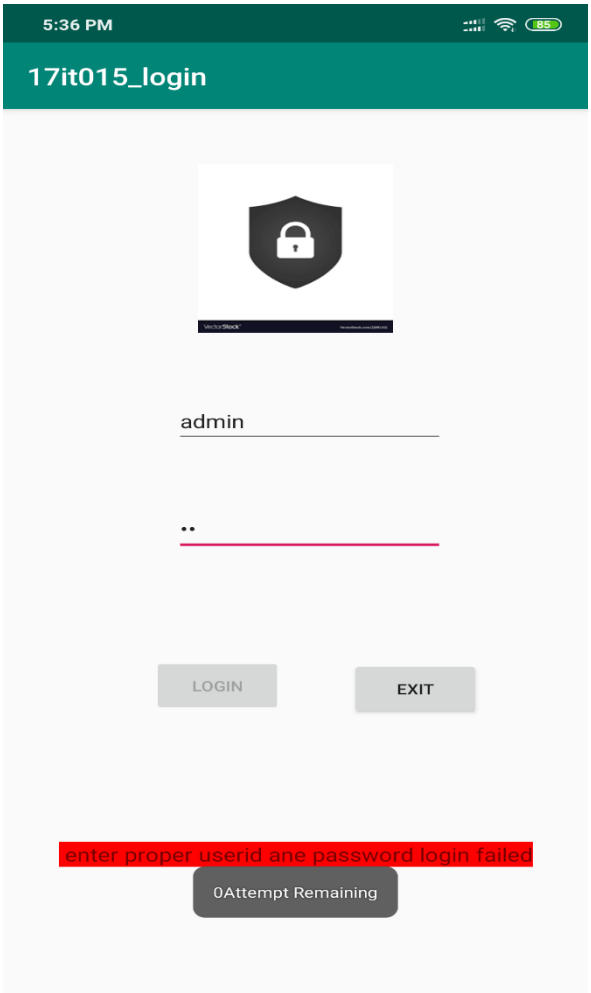
OUTPUT:

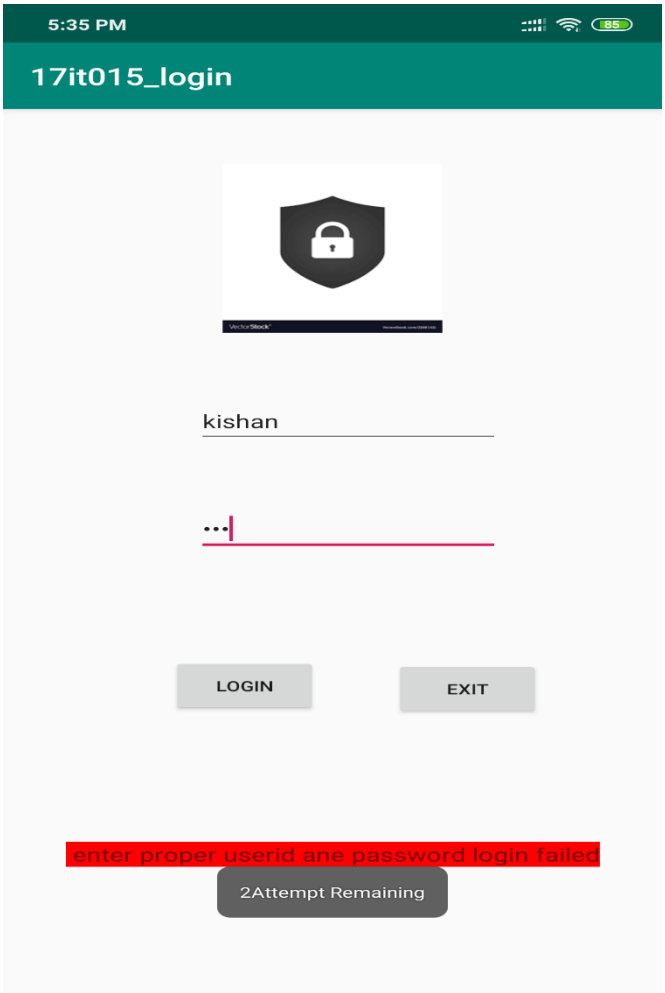





5:36 PM
17it015_login
admin
LOGIN
EXIT
enter proper userid ane password login failed
1Attempt Remaining


5:35 PM
17it015_login
username
password
LOGIN
EXIT


5:36 PM
17it015_login
admin
LOGIN
EXIT
enter proper userid ane password login failed
0Attempt Remaining


5:35 PM
17it015_login
kishan
LOGIN
EXIT
enter proper userid ane password login failed
2Attempt Remaining

LATEST APPLICATIONS:

Instagram, Facebook, Helo, Amazon, Flipkart, etc.

LEARNING OUTCOME:

In this practical we learnt how to create button ,textview , toast message, text field and also how to put an image on login page .I also learnt validation.

PRACTICAL 6

AIM:

Create an application which turns ON or OFF Torch/Flashlight of Camera.

THEORY:

Switch: Switch is a two-state toggle switch widget that can select between two options. It is used to display checked and unchecked state of a button providing slider control to user. Switch is a subclass of CompoundButton. It is basically an off/on button which indicates the current state of Switch. It is commonly used in selecting on/off in Sound, Bluetooth, Wi-Fi etc.

Permission: Specifies a system permission that the user must grant in order for the app to operate correctly. Permissions are granted by the user when the application is installed (on devices running Android 5.1 and lower) or while the app is running (on devices running Android 6.0 and higher).

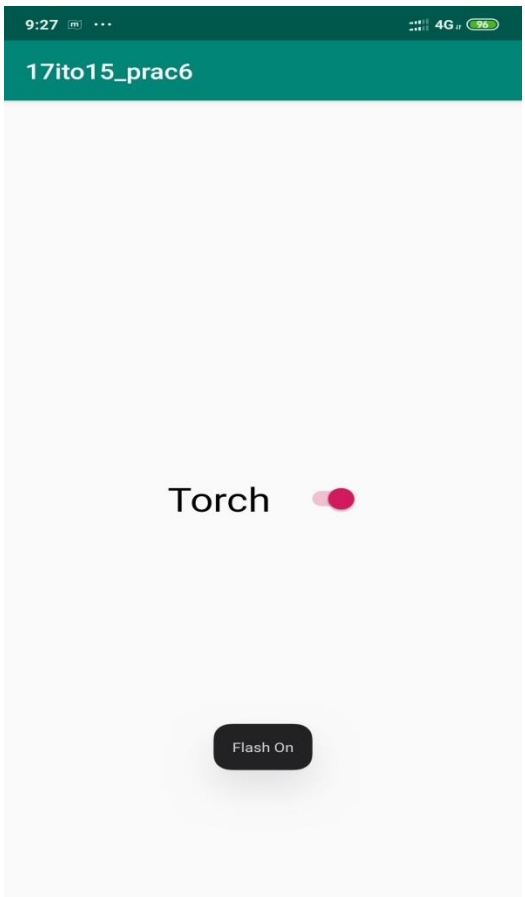
CODE:

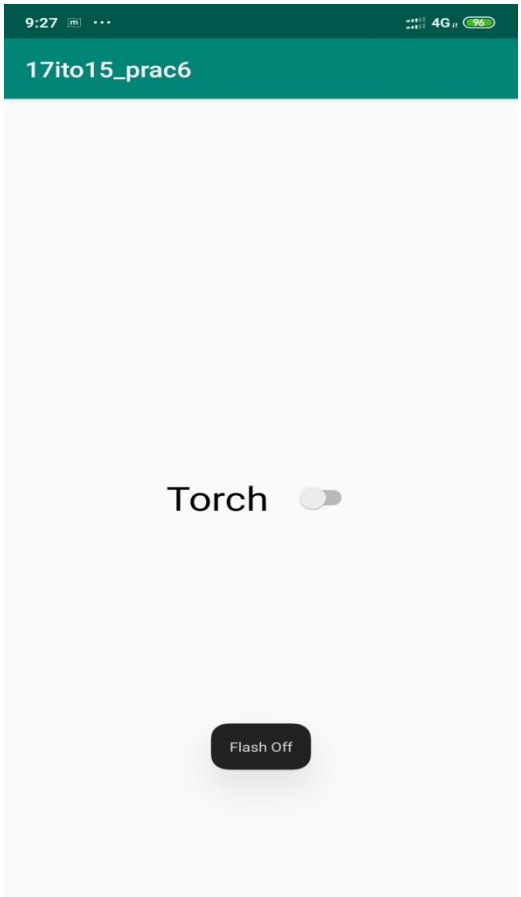
```
// MainActivity.java
package com.example.a17ito15_prac6;
import androidx.annotation.RequiresApi;
import androidx.appcompat.app.AppCompatActivity;
import android.content.Context;
import android.graphics.Camera;
import android.hardware.camera2.CameraAccessException;
import android.hardware.camera2.CameraManager;
import android.os.Build;
import android.os.Bundle;
import android.widget.CompoundButton;
import android.widget.Switch;
import android.widget.Toast;
import java.security.Policy;
public class MainActivity extends AppCompatActivity {
Switch s;
   @Override
   protected void onCreate(Bundle savedInstanceState) {
      super.onCreate(savedInstanceState);
      setContentView(R.layout.activity_main);
      s=(Switch) findViewById(R.id.switch1);
      s.setOnCheckedChangeListener(new CompoundButton.OnCheckedChangeListener() {
         @RequiresApi(api = Build.VERSION_CODES.M)
         @Override
         public void onCheckedChanged(CompoundButton buttonView, boolean isChecked) {
            if(s.isChecked())
            {
               flashLightOn();
            }
            else
            {
```

```
                flashLightOff();
            }
        }
    });
}
@RequiresApi(api = Build.VERSION_CODES.M)
private void flashLightOn() {
    CameraManager cameraManager = (CameraManager)
getSystemService(Context.CAMERA_SERVICE);

    try {
        String cameraId = cameraManager.getCameraIdList()[0];
        cameraManager.setTorchMode(cameraId, true);
        Toast.makeText(MainActivity.this,"Flash On",Toast.LENGTH_LONG).show();
    } catch (CameraAccessException e) {
    }
}
@RequiresApi(api = Build.VERSION_CODES.M)
private void flashLightOff() {
    CameraManager cameraManager = (CameraManager)
getSystemService(Context.CAMERA_SERVICE);
    try {
        String cameraId = cameraManager.getCameraIdList()[0];
        cameraManager.setTorchMode(cameraId, false);
        Toast.makeText(MainActivity.this,"Flash Off",Toast.LENGTH_LONG).show();
    } catch (CameraAccessException e) {
    }
}
}
// 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">
    <Switch
        android:id="@+id/switch1"
        android:layout_width="145dp"
        android:layout_height="55dp"
        android:text="Torch"
        android:textSize="36sp"
        app:layout_constraintBottom_toBottomOf="parent"
        app:layout_constraintEnd_toEndOf="parent"
        app:layout_constraintStart_toStartOf="parent"
        app:layout_constraintTop_toTopOf="parent" />

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

```
// AndroidManifest.xml
<?xml version="1.0" encoding="utf-8"?>
<manifest xmlns:android="http://schemas.android.com/apk/res/android"
    package="com.example.a17ito15_prac6">

    <application
        android:allowBackup="true"
        android:icon="@mipmap/ic_launcher"
        android:label="@string/app_name"
        android:roundIcon="@mipmap/ic_launcher_round"
        android:supportsRtl="true"
        android:theme="@style/AppTheme">
        <activity android:name=".MainActivity">
            <intent-filter>
                <action android:name="android.intent.action.MAIN" />
                <category android:name="android.intent.category.LAUNCHER" />
            </intent-filter>
        </activity>
    </application>
    <uses-permission android:name="android.permission.CAMERA" />
    <uses-feature android:name="android.hardware.camera" />
</manifest>
```

OUTPUT:





LATEST APPLICATIONS:

Huji,vivavideo etc…

LEARNING OUTCOME:

In this practical we learn how to display toast message in application and also how to use Switch and also how to take permission for using camera function.

PRACTICAL 7

AIM:

Create an application that will change color of the screen, based on selected options from the menu.

THEORY:

Option menu: Android Option Menus are the primary menus of android. They can be used for settings, search, delete item etc.Here, we are going to see two examples of option menus. First, the simple option menus and second, options menus with images.Here, we are inflating the menu by calling the inflate() method of MenuInflater class. To perform event handling on menu items, you need to override onOptionsItemSelected().

Toast: Andorid Toast can be used to display information for the short period of time. A toast contains message to be displayed quickly and disappears after sometime.The android.widget.Toast class is the subclass of java.lang.Object class.You can also create custom toast as well for example toast displaying image. You can visit next page to see the code for custom toast.

CODE:

```
// MainActivity.java
package com.example.a17it015_prac7;
import androidx.appcompat.app.AppCompatActivity;
import androidx.constraintlayout.widget.ConstraintLayout;
import android.graphics.Color;
import android.os.Bundle;
import android.view.ContextMenu;
import android.view.Menu;
import android.view.MenuItem;
import android.widget.Toast;
public class MainActivity extends AppCompatActivity {
   ConstraintLayout  current;
   @Override
   protected void onCreate(Bundle savedInstanceState) {
      super.onCreate(savedInstanceState);
      setContentView(R.layout.activity_main);
      current = (ConstraintLayout)findViewById(R.id.lay);
   }
   public boolean onCreateOptionsMenu(Menu menu) {
     // super.onCreateContextMenu(menu, v, menuInfo);
    //  MenuInflater inflater = getMenuInflater();
      getMenuInflater().inflate(R.menu.option_menu, menu);
      return true;
   }
   public boolean onOptionsItemSelected(MenuItem item) {

      switch (item.getItemId()) {
         case R.id.item1:
            current.setBackgroundColor(Color.RED);
```

```
            Toast.makeText(getApplicationContext(),"RED color
",Toast.LENGTH_SHORT).show();
            return true;
         case R.id.item2:
            current.setBackgroundColor(Color.GREEN);
            Toast.makeText(getApplicationContext(),"GREEN
color",Toast.LENGTH_SHORT).show();
            return true;
         case R.id.item3:
            current.setBackgroundColor(Color.BLUE);
            Toast.makeText(getApplicationContext(),"BLUE
color",Toast.LENGTH_SHORT).show();
            return true;
         default:
            return super.onContextItemSelected(item);
      }
   }
}
```

// **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:id="@+id/lay"
   android:layout_width="match_parent"
   android:layout_height="match_parent"
   tools:context=".MainActivity">

  <TextView
     android:layout_width="wrap_content"
     android:layout_height="wrap_content"
     android:text="To see the menu click the dots above in actionbar"
     android:textSize="18sp"
     app:layout_constraintBottom_toBottomOf="parent"
     app:layout_constraintLeft_toLeftOf="parent"
     app:layout_constraintRight_toRightOf="parent"
     app:layout_constraintTop_toTopOf="parent"
     app:layout_constraintVertical_bias="0.181" />
</androidx.constraintlayout.widget.ConstraintLayout>
```

// AndroidManifest.xml

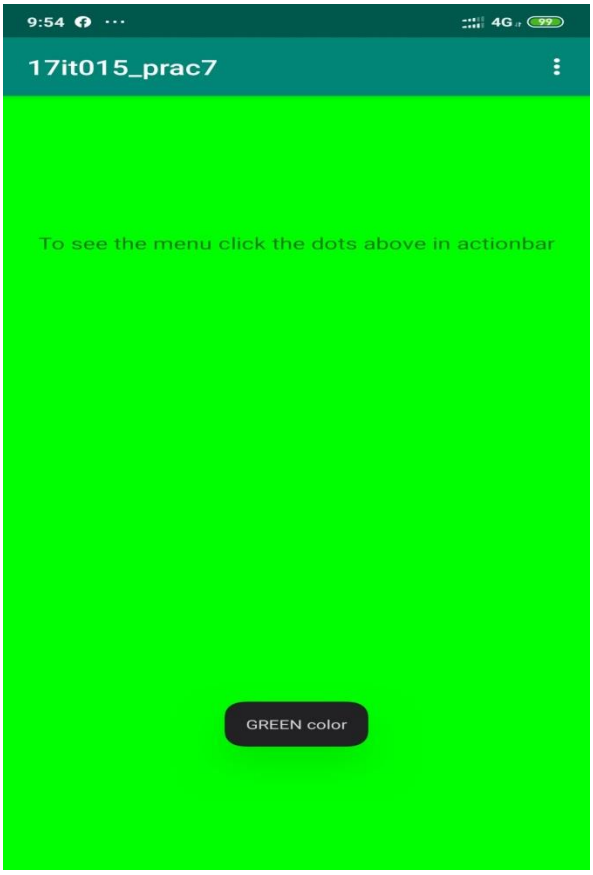
```
<?xml version="1.0" encoding="utf-8"?>
<manifest xmlns:android="http://schemas.android.com/apk/res/android"
   package="com.example.a17it015_prac7">
   <application
      android:allowBackup="true"
      android:icon="@mipmap/ico"
      android:label="@string/app_name"
      android:roundIcon="@mipmap/ic_launcher_round"
      android:supportsRtl="true"
      android:theme="@style/AppTheme">
```

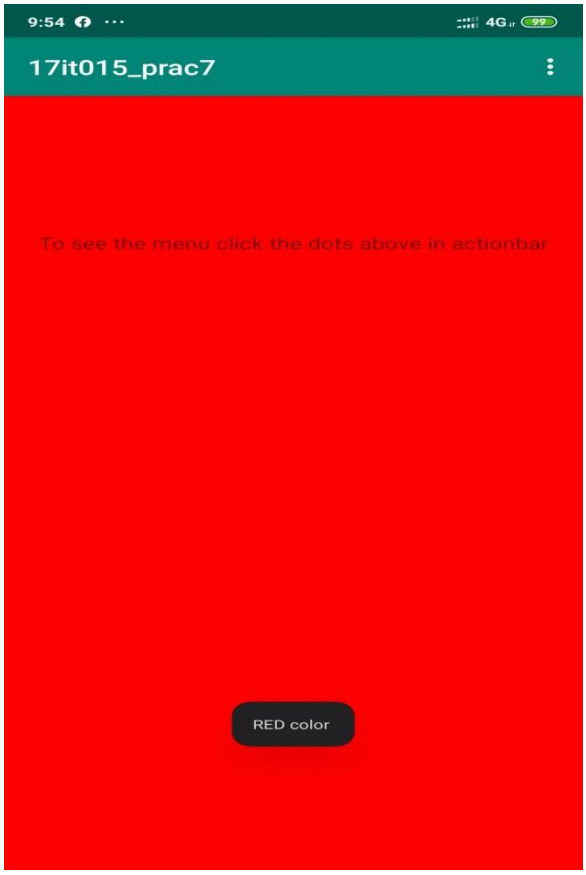
```
        <activity android:name=".MainActivity">
            <intent-filter>
                <action android:name="android.intent.action.MAIN" />

                <category android:name="android.intent.category.LAUNCHER" />
            </intent-filter>
        </activity>
    </application>

</manifest>
```

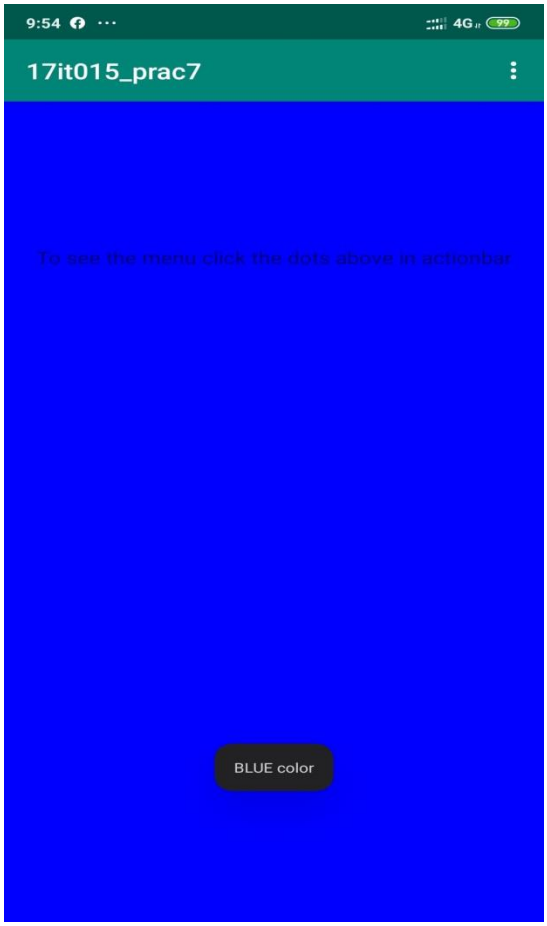
// **option_menu.xml**

```
<?xml version="1.0" encoding="utf-8"?>
<menu xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:app="http://schemas.android.com/apk/res-auto">
    <item android:id="@+id/item1"
        android:title="Red"/>

    <item android:id="@+id/item2"
        android:title="Green"/>

    <item android:id="@+id/item3"
        android:title="Blue"/>

</menu>
```

OUTPUT:







LATEST APPLICATIONS:

Snapseed,GFX tool,crickbuzz

LEARNING OUTCOME:

In this practical we learn how to display toast message in application and also how to use option menu and how to add actions on their items list.

PRACTICAL: 8

AIM:

Create an application with the help of fragment.

THEORY:

Fragment: - In Android, Fragment is a part of an activity which enable more modular activity design. It will not be wrong if we say a fragment is a kind of sub-activity. It represents a behavior or a portion of user interface in an Activity. We can combine multiple Fragments in Single Activity to build a multi panel UI and reuse a Fragment in multiple Activities. We always need to embed Fragment in an activity and the fragment lifecycle is directly affected by the host activity's lifecycle.

Need of Fragments in Android: - Before the introduction of Fragment's we can only show a single Activity on the screen at one given point of time so we were not able to divide the screen and control different parts separately. With the help of Fragment's, we can divide the screens in different parts and controls different parts separately.

By using Fragments, we can comprise multiple Fragments in a single Activity. Fragments have their own events, layouts and complete life cycle. It provides flexibility and also removed the limitation of single Activity on the screen at a time.

CODE:

// MainActivity.java

```
package com.example.fragment_17it015;
import android.os.Bundle;
import android.view.View;
import android.widget.Button;
import android.widget.Toast;
import androidx.appcompat.app.AppCompatActivity;

public class MainActivity extends AppCompatActivity {
   private Button button;
   @Override
   protected void onCreate(Bundle savedInstanceState) {
      super.onCreate(savedInstanceState);
      setContentView(R.layout.activity_main);
      button = findViewById(R.id.button);
      button.setOnClickListener(new View.OnClickListener() {
         @Override
         public void onClick(View view) {
            Toast.makeText(getApplicationContext(),"Main Activity
Toast",Toast.LENGTH_SHORT).show();
         }
      });
   }
}
```

```
// MyFragment.java
        package com.example.fragment_17it015;
        import android.os.Bundle;
        import android.view.LayoutInflater;
        import android.view.View;
        import android.view.ViewGroup;
        import android.widget.Button;
        import android.widget.Toast;
        import androidx.annotation.NonNull;
        import androidx.annotation.Nullable;
        import androidx.fragment.app.Fragment;

        public class MyFragment extends Fragment {
           private View view;
           private Button button;
           @Override
           public void onCreate(@Nullable Bundle savedInstanceState) {
              super.onCreate(savedInstanceState);

           }

           @Nullable
           @Override
           public View onCreateView(@NonNull LayoutInflater inflater, @Nullable ViewGroup
        container, @Nullable Bundle savedInstanceState) {

              view = inflater.inflate(R.layout.fragment_my,container,false);
              button = view.findViewById(R.id.button);

              button.setOnClickListener(new View.OnClickListener() {
                 @Override
                 public void onClick(View view) {
                    Toast.makeText(getActivity(),"Fragment
        Activity",Toast.LENGTH_SHORT).show();
                 }
              });

              return view;
           }
}
// activity_mail.xml
<?xml version="1.0" encoding="utf-8"?>
<LinearLayout xmlns:android="http://schemas.android.com/apk/res/android"
   xmlns:app="http://schemas.android.com/apk/res-auto"
   xmlns:tools="http://schemas.android.com/tools"
   android:layout_width="match_parent"
   android:layout_height="match_parent"
   tools:context=".MainActivity"
   android:gravity="center"
   android:orientation="vertical">
```

```
    <Button
        android:id="@+id/button"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="Click me !!"
        android:textSize="30sp"
        android:background="#000"
        android:textColor="#FFF"
        android:padding="20dp"/>

    <fragment
        android:id="@+id/fragment"
        android:name="com.example.fragment_17it015.MyFragment"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:layout_marginTop="50dp"
        android:layout_marginHorizontal="20dp"/>
</LinearLayout>
```

// fragment_my.xml

```
        <?xml version="1.0" encoding="utf-8"?>
        <FrameLayout xmlns:android="http://schemas.android.com/apk/res/android"
            xmlns:tools="http://schemas.android.com/tools"
            android:layout_width="match_parent"
            android:layout_height="match_parent"
            tools:context="com.example.fragment_17it015.MyFragment">
            <LinearLayout
                android:layout_width="match_parent"
                android:layout_height="match_parent"
                android:orientation="vertical"
                android:gravity="center"
                android:background="#19fff8"
                >
                <TextView
                    android:layout_width="match_parent"
                    android:layout_height="wrap_content"
                    android:text="17IT015"
                    android:textAlignment="center"
                    android:textSize="50sp"
                    android:textColor="#000"/>
                <Button
                    android:id="@+id/button"
                    android:layout_width="wrap_content"
                    android:layout_height="wrap_content"
                    android:text="Click Me  !!"
                    android:textSize="20sp"
                    android:padding="15dp"
                    android:background="#FFF"
                    android:layout_margin="50dp" />
            </LinearLayout>

        </FrameLayout>
```

OUTPUT:


10:14
fragment_17it015
CLICK ME !!
17IT015
CLICK ME !!


10:15
fragment_17it015
CLICK ME !!
17IT015
CLICK ME !!
Fragment Activity

LATEST APPLICATIONS:

Application like Google play store, Movies, Music and many more which shows content in tab layout uses fragment for display in different content for different tabs.

LEARNING OUTCOME:

1. Working with android fragment.
2. How android fragment activity is different from MainActivity.
3. Use of fragment in more than 1 activity.

PRACTICAL 9

AIM:

Create an application with the help of web view.

THEORY:

EditText: In android, EditText is a user interface control which is used to allow the user to enter or modify the text. While using EditText control in our android applications, we need to specify the type of data the text field can accept using input Type attribute. For example, if it accepts plain text, then we need to specify the inputType as “text”. In case if EditText field is for password, then we need to specify the inputType as “textPassword”.

Permission: Specifies a system permission that the user must grant in order for the app to operate correctly. Permissions are granted by the user when the application is installed (on devices running Android 5.1 and lower) or while the app is running (on devices running Android 6.0 and higher).\

WebView: WebView is a view that display web pages inside your application. You can also specify HTML string and can show it inside your application using WebView. WebView makes turns your application to a web application.

CODE:

```
// MainActivity.java
package com.example.wcmc_17it015_prac_9;
import androidx.appcompat.app.AppCompatActivity;
import android.os.Bundle;
import android.view.View;
import android.webkit.WebView;
import android.webkit.WebViewClient;
import android.widget.Button;
import android.widget.EditText;

public class MainActivity extends AppCompatActivity {
    WebView w;
    EditText t1;
    Button b1;
    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_main);
        b1 = findViewById(R.id.button);
        t1 = findViewById(R.id.editText);
        w = findViewById(R.id.webview);
        w.loadUrl("http://kishandalsania.ml/");
        b1.setOnClickListener(new View.OnClickListener() {
            @Override
            public void onClick(View v) {
                w.getSettings().setLoadsImagesAutomatically(true);
```

```
            w.getSettings().setJavaScriptEnabled(true);
            w.setScrollBarStyle(View.SCROLLBARS_INSIDE_OVERLAY);
            w.loadUrl(t1.getText().toString());
        }
      });
   }
}
```

// **activity_main.xml**

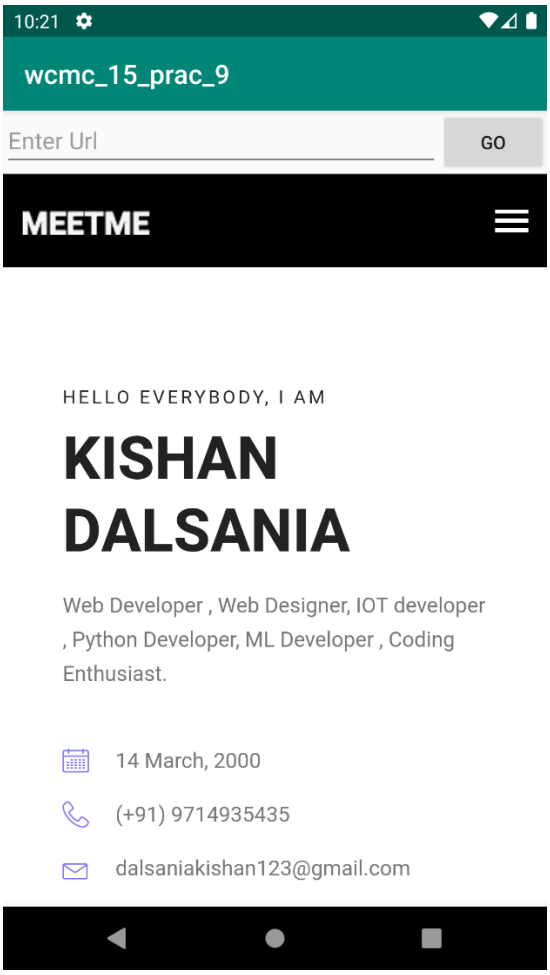
```
<?xml version="1.0" encoding="utf-8"?>
<LinearLayout xmlns:android="http://schemas.android.com/apk/res/android"
   xmlns:app="http://schemas.android.com/apk/res-auto"
   xmlns:tools="http://schemas.android.com/tools"
   android:layout_width="match_parent"
   android:layout_height="match_parent"
   android:orientation="vertical"
   tools:context=".MainActivity">
   <LinearLayout
      android:layout_width="match_parent"
      android:layout_height="wrap_content"
      android:orientation="horizontal">
      <EditText
         android:id="@+id/editText"
         android:layout_width="330dp"
         android:layout_height="wrap_content"
         android:hint="Enter Url"
         android:inputType="textPersonName" />
      <Button
         android:id="@+id/button"
         android:layout_width="wrap_content"
         android:layout_height="wrap_content"
         android:text="Go" />
   </LinearLayout>
   <WebView
      android:id="@+id/webview"
      android:layout_width="match_parent"
      android:layout_height="match_parent"
      android:layout_marginBottom="4dp" />
</LinearLayout>
```

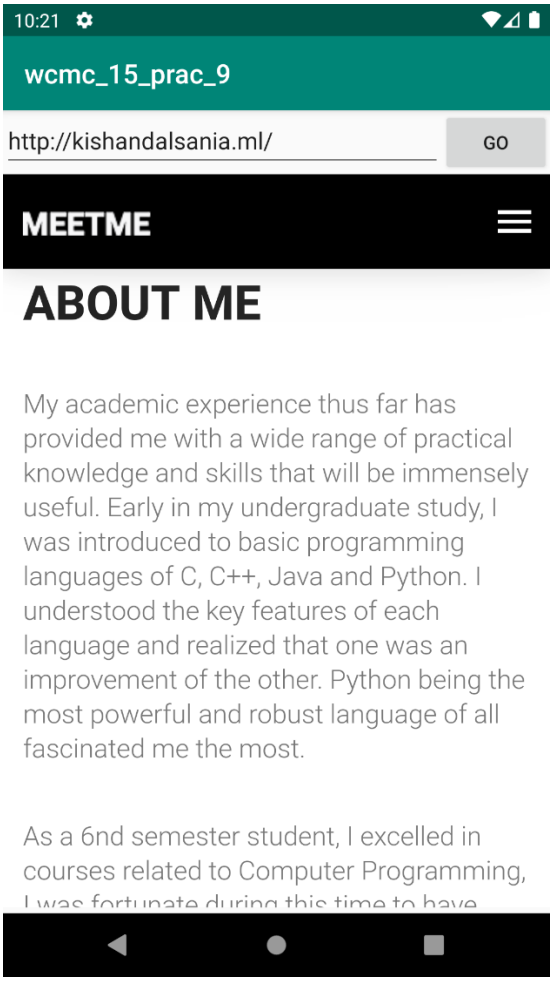
// **AndroidManifest.xml**

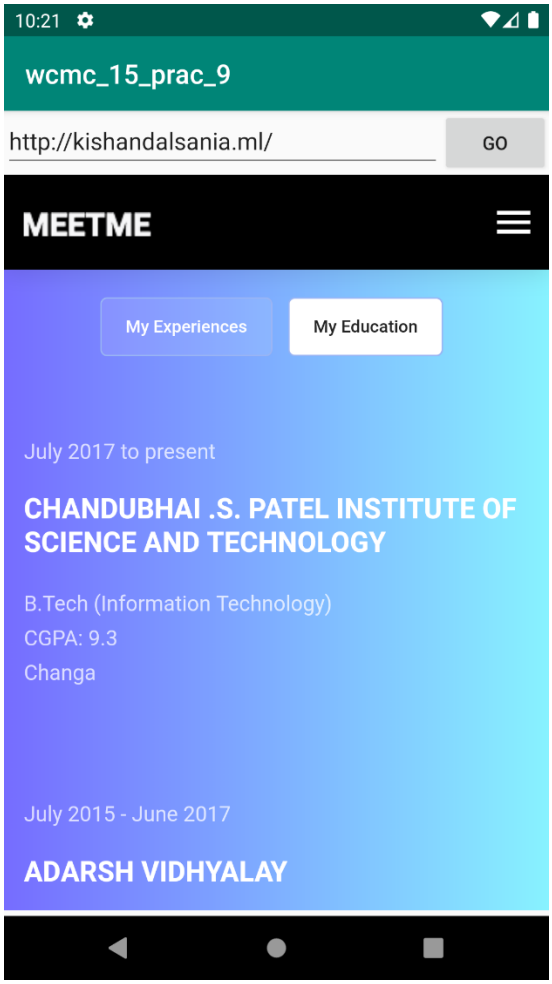
```
<?xml version="1.0" encoding="utf-8"?>
<manifest xmlns:android="http://schemas.android.com/apk/res/android"
   package="com.example.wcmc_17it015_prac_9">
   <uses-permission android:name="android.permission.INTERNET"/>
   <application
      android:allowBackup="true"
      android:icon="@mipmap/ic_launcher"
      android:label="@string/app_name"
      android:roundIcon="@mipmap/ic_launcher_round"
      android:supportsRtl="true"
      android:theme="@style/AppTheme">
      <activity android:name=".MainActivity">
```

```
        <intent-filter>
            <action android:name="android.intent.action.MAIN" />
            <category android:name="android.intent.category.LAUNCHER" />
        </intent-filter>
    </activity>
</application>

</manifest>
```

OUTPUT:







LATEST APPLICATIONS:

Quora , Reactnative , ArcadeHero etc…

LEARNING OUTCOME:

In this practical we learn how to display web pages inside our application and also we turn our application into Web application with the help of WebView and also how to take permission for using internet.

PRACTICAL 10

AIM:

Create an application with the help of database.

THEORY:

SQLite : SQLite is an open-source relational database i.e. used to perform database operations on android devices such as storing, manipulating or retrieving persistent data from the database. It is embedded in android bydefault. So, there is no need to perform any database setup or administration task. SQLiteOpenHelper class provides the functionality to use the SQLite database. openOrCreateDatabase() this method is used to create a database.

execSQL() : This method not only insert data , but also used to update or modify already existing data in database using bind arguments.

rawQuery() : directly accepts an SQL select statement as input. The method returns 'Cursor' object which points to one row of the query result.

CODE:

// MainActivity.java

```
package com.example.practical10;
import androidx.appcompat.app.AppCompatActivity;
import android.content.Context;
import android.content.Intent;
import android.database.sqlite.SQLiteDatabase;
import android.os.Bundle;
import android.provider.ContactsContract;
import android.view.View;
import android.widget.Button;
import android.database.Cursor;

public class MainActivity extends AppCompatActivity {

   Button in,del,viewa;
   SQLiteDatabase db;

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

      in=findViewById(R.id.button5);
      del=findViewById(R.id.button6);
      viewa=findViewById(R.id.button7);

      db=openOrCreateDatabase("myDb", Context.MODE_PRIVATE,null);
      db.execSQL("CREATE TABLE IF NOT EXISTS student(name VARCHAR,rollno
VARCHAR);");
```

```
        in.setOnClickListener(new View.OnClickListener() {
            @Override
            public void onClick(View v) {
                startActivity(new Intent(MainActivity.this, Insert.class));
            }
        });

        del.setOnClickListener(new View.OnClickListener() {
            @Override
            public void onClick(View v) {
                startActivity(new Intent(MainActivity.this, Delete.class));
            }
        });

        viewa.setOnClickListener(new View.OnClickListener() {
            @Override
            public void onClick(View v) {
                startActivity(new Intent(MainActivity.this,Viewa.class));
            }
        });
    }
}
```

//Insert.java

```
package com.example.practical10;

import androidx.appcompat.app.AppCompatActivity;

import android.content.Context;
import android.database.SQLException;
import android.database.sqlite.SQLiteDatabase;
import android.os.Bundle;
import android.view.View;
import android.widget.EditText;
import android.widget.Toast;
import android.widget.Button;


public class Insert extends AppCompatActivity {

    EditText name,rollno;
    Button insert;
    SQLiteDatabase db;

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

        name=findViewById(R.id.editText);
        rollno=findViewById(R.id.editText2);
```

```
        insert=findViewById(R.id.button);

        db=openOrCreateDatabase("myDb", Context.MODE_PRIVATE,null);

        insert.setOnClickListener(new View.OnClickListener() {
            @Override
            public void onClick(View v) {

                if (name.getText().toString().trim().length()==0 ||
rollno.getText().toString().trim().length()==0){
                    Toast.makeText(getApplicationContext(),"Please Enter Details",
Toast.LENGTH_SHORT).show();
                }
                else
                {
                    try{
                        db.execSQL("INSERT INTO student
VALUES('"+name.getText()+"','"+rollno.getText()+"');");
                        Toast.makeText(getApplicationContext(),"Inserted
Successfully",Toast.LENGTH_SHORT).show();
                    }
                    catch (SQLException e){
                        Toast.makeText(getApplicationContext(),"Operation Not
Successful",Toast.LENGTH_SHORT).show();
                    }
                }
            }
        });
    }
}
```

//Delete.java

```
package com.example.practical10;

import androidx.appcompat.app.AppCompatActivity;
import android.content.Context;
import android.database.Cursor;
import android.database.sqlite.SQLiteDatabase;
import android.os.Bundle;
import android.view.View;
import android.widget.Button;
import android.widget.EditText;
import android.widget.Toast;


public class Delete extends AppCompatActivity {

    EditText name,rollno;
    Button delete;
    SQLiteDatabase db;

    @Override
```

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

      name=findViewById(R.id.editText);
      rollno=findViewById(R.id.editText2);
      delete=findViewById(R.id.button);

      db=openOrCreateDatabase("myDb", Context.MODE_PRIVATE,null);

      delete.setOnClickListener(new View.OnClickListener() {
         @Override
         public void onClick(View v) {

            if (name.getText().toString().trim().length()==0 ||
rollno.getText().toString().trim().length()==0){
               Toast.makeText(getApplicationContext(),"Please Enter Details",
Toast.LENGTH_SHORT).show();
            }

            Cursor c=db.rawQuery("SELECT * FROM student WHERE
name='"+name.getText()+"'", null);
            if(c.moveToFirst())
            {
               db.execSQL("DELETE FROM student WHERE name='"+name.getText()+"'");
               Toast.makeText(getApplicationContext(),"Deleted
Successfully",Toast.LENGTH_SHORT).show();
            }
            else
            {
               Toast.makeText(getApplicationContext(),"Operation Not
Successful",Toast.LENGTH_SHORT).show();
            }

         }
      });

   }
}
```

//Viewa.java

```
package com.example.practical10;

import androidx.appcompat.app.AppCompatActivity;

import android.content.Context;
import android.database.Cursor;
import android.database.sqlite.SQLiteDatabase;
import android.os.Bundle;
import android.widget.EditText;
import android.widget.TextView;
import android.widget.Toast;
```

```
public class Viewa extends AppCompatActivity {

   TextView t1;
   SQLiteDatabase db;

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

      t1=findViewById(R.id.textView4);

      db=openOrCreateDatabase("myDb", Context.MODE_PRIVATE,null);

      Cursor c=db.rawQuery("SELECT * FROM student",null);
      if (c.getCount()==0){
         Toast.makeText(getApplicationContext(),"No Data
Found",Toast.LENGTH_SHORT).show();
      }

      StringBuffer result=new StringBuffer();
      while (c.moveToNext()){
         result.append("Name:"+c.getString(0)+"\n");
         result.append("Roll No:"+c.getString(1)+"\n\n");
      }
      t1.setText(result.toString());

   }
}
```

//AndroidManifest.xml

```
<?xml version="1.0" encoding="utf-8"?>
<manifest xmlns:android="http://schemas.android.com/apk/res/android"
 package=" com.example.practical10">

   <application
      android:allowBackup="true"
      android:icon="@mipmap/ic_launcher"
      android:label="@string/app_name"
      android:roundIcon="@mipmap/ic_launcher_round"
      android:supportsRtl="true"
      android:theme="@style/AppTheme">
      <activity android:name=".Viewa"></activity>
      <activity android:name=".Delete" />
      <activity android:name=".Insert" />
      <activity android:name=".MainActivity">
         <intent-filter>
            <action android:name="android.intent.action.MAIN" />

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

```
        </activity>
    </application>

</manifest>
```

//activity_main.xml

```
<?xml version="1.0" encoding="utf-8"?>
<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"
    android:background="#fff"
    >

    <Button
        android:id="@+id/button5"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_marginStart="161dp"
        android:layout_marginLeft="161dp"
        android:layout_marginTop="283dp"
        android:layout_marginEnd="162dp"
        android:layout_marginRight="162dp"
        android:layout_marginBottom="64dp"
        android:background="#FFE5E5"
        android:text="Insert"
        android:textColor="#000"
        app:layout_constraintBottom_toTopOf="@+id/button6"
        app:layout_constraintEnd_toEndOf="parent"
        app:layout_constraintStart_toStartOf="parent"
        app:layout_constraintTop_toTopOf="parent" />

    <Button
        android:id="@+id/button6"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_marginStart="161dp"
        android:layout_marginLeft="161dp"
        android:layout_marginTop="64dp"
        android:layout_marginEnd="162dp"
        android:layout_marginRight="162dp"
        android:layout_marginBottom="64dp"
        android:background="#FFE5E5"
        android:text="Delete"
        android:textColor="#000"
        app:layout_constraintBottom_toTopOf="@+id/button7"
        app:layout_constraintEnd_toEndOf="parent"
        app:layout_constraintStart_toStartOf="parent"
```

```
      app:layout_constraintTop_toBottomOf="@+id/button5" />

   <Button
      android:id="@+id/button7"
      android:layout_width="wrap_content"
      android:layout_height="wrap_content"
      android:layout_marginStart="161dp"
      android:layout_marginLeft="161dp"
      android:layout_marginTop="64dp"
      android:layout_marginEnd="162dp"
      android:layout_marginRight="162dp"
      android:layout_marginBottom="148dp"
      android:background="#FFE5E5"
      android:text="View All"
      android:textColor="#000"
      app:layout_constraintBottom_toBottomOf="parent"
      app:layout_constraintEnd_toEndOf="parent"
      app:layout_constraintStart_toStartOf="parent"
      app:layout_constraintTop_toBottomOf="@+id/button6" />

   <TextView
      android:id="@+id/textView3"
      android:layout_width="wrap_content"
      android:layout_height="wrap_content"
      android:layout_marginStart="159dp"
      android:layout_marginLeft="159dp"
      android:layout_marginTop="95dp"
      android:layout_marginEnd="159dp"
      android:layout_marginRight="159dp"
      android:layout_marginBottom="144dp"
      android:text="Database Data"
      app:layout_constraintBottom_toTopOf="@+id/button5"
      app:layout_constraintEnd_toEndOf="parent"
      app:layout_constraintStart_toStartOf="parent"
      app:layout_constraintTop_toTopOf="parent"
      android:textColor="#B53160"
      android:textSize="30sp"
      />
</androidx.constraintlayout.widget.ConstraintLayout>
```

//activity_viewa.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=".Viewa"
   android:background="#fff"
   >
```

```
    <TextView
        android:id="@+id/textView4"
        android:layout_width="259dp"
        android:layout_height="498dp"
        android:layout_marginStart="48dp"
        android:layout_marginLeft="48dp"
        android:layout_marginTop="35dp"
        android:layout_marginEnd="56dp"
        android:layout_marginRight="56dp"
        android:layout_marginBottom="48dp"
        app:layout_constraintBottom_toBottomOf="parent"
        app:layout_constraintEnd_toEndOf="parent"
        app:layout_constraintStart_toStartOf="parent"
        app:layout_constraintTop_toTopOf="parent"
        android:textSize="10sp"
        android:textColor="#FFE5E5"
        />
</androidx.constraintlayout.widget.ConstraintLayout>
```

//activity_insert.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=".Insert"
    android:background="#fff"
    >

    <EditText
        android:id="@+id/editText"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_marginTop="156dp"
        android:layout_marginEnd="30dp"
        android:layout_marginRight="30dp"
        android:layout_marginBottom="60dp"
        android:ems="10"
        android:hint="Name"
        android:inputType="textPersonName"
        app:layout_constraintBottom_toTopOf="@+id/editText2"
        app:layout_constraintEnd_toEndOf="parent"
        app:layout_constraintTop_toTopOf="parent" />

    <TextView
        android:id="@+id/textView"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_marginStart="58dp"
        android:layout_marginLeft="58dp"
```

```
    android:layout_marginTop="96dp"
    android:layout_marginEnd="56dp"
    android:layout_marginRight="56dp"
    android:text="Name"
    android:textColor="#B53160"
    android:textSize="20sp"
    app:layout_constraintEnd_toStartOf="@+id/editText"
    app:layout_constraintStart_toStartOf="parent"
    app:layout_constraintTop_toTopOf="parent" />

<EditText
    android:id="@+id/editText2"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:layout_marginTop="60dp"
    android:layout_marginEnd="30dp"
    android:layout_marginRight="30dp"
    android:layout_marginBottom="425dp"
    android:ems="10"
    android:hint="Roll No"
    android:inputType="textPersonName"
    app:layout_constraintBottom_toBottomOf="parent"
    app:layout_constraintEnd_toEndOf="parent"
    app:layout_constraintTop_toBottomOf="@+id/editText" />

<TextView
    android:id="@+id/textView2"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:layout_marginStart="58dp"
    android:layout_marginLeft="58dp"
    android:layout_marginTop="145dp"
    android:layout_marginEnd="20dp"
    android:layout_marginRight="20dp"
    android:layout_marginBottom="437dp"
    android:text="Roll No"
    android:textColor="#B53160"
    android:textSize="20sp"
    app:layout_constraintBottom_toBottomOf="parent"
    app:layout_constraintEnd_toStartOf="@+id/editText2"
    app:layout_constraintHorizontal_bias="1.0"
    app:layout_constraintStart_toStartOf="parent"
    app:layout_constraintTop_toBottomOf="@+id/textView" />

<Button
    android:id="@+id/button"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:layout_marginStart="180dp"
    android:layout_marginLeft="180dp"
    android:layout_marginTop="58dp"
```

```
        android:layout_marginEnd="92dp"
        android:layout_marginRight="92dp"
        android:layout_marginBottom="299dp"
        android:background="#FFE5E5"
        android:text="Insert"
        android:textColor="#000"
        app:layout_constraintBottom_toBottomOf="parent"
        app:layout_constraintEnd_toEndOf="parent"
        app:layout_constraintStart_toStartOf="parent"
        app:layout_constraintTop_toBottomOf="@+id/editText2" />
</androidx.constraintlayout.widget.ConstraintLayout>
```

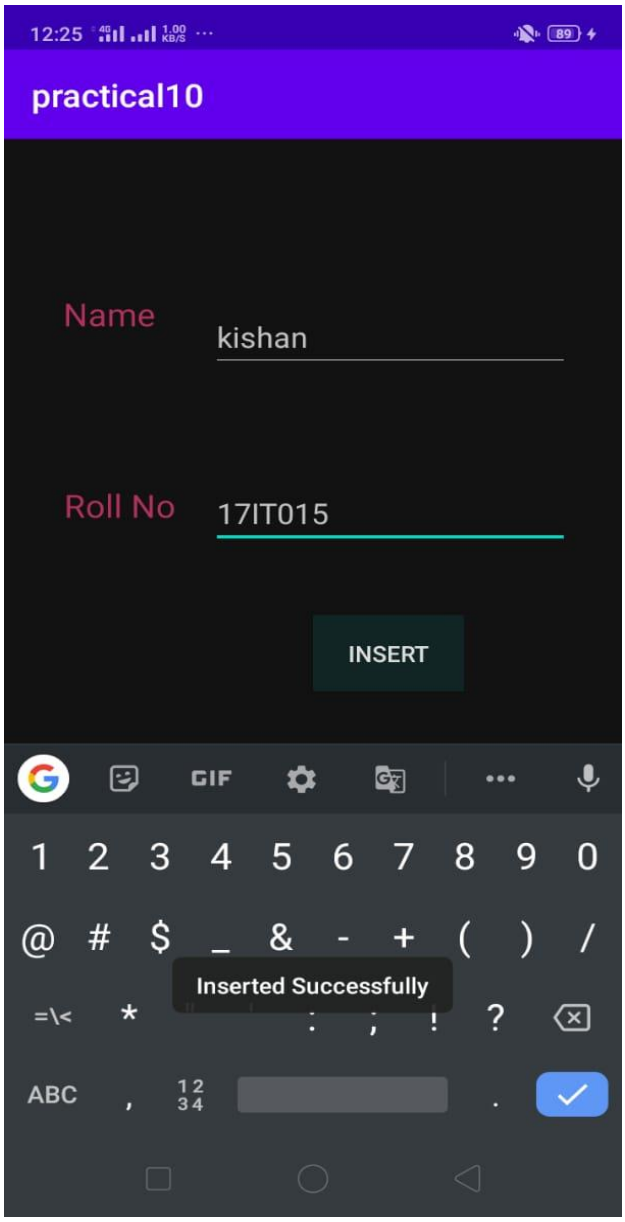
//activity_delete.xml

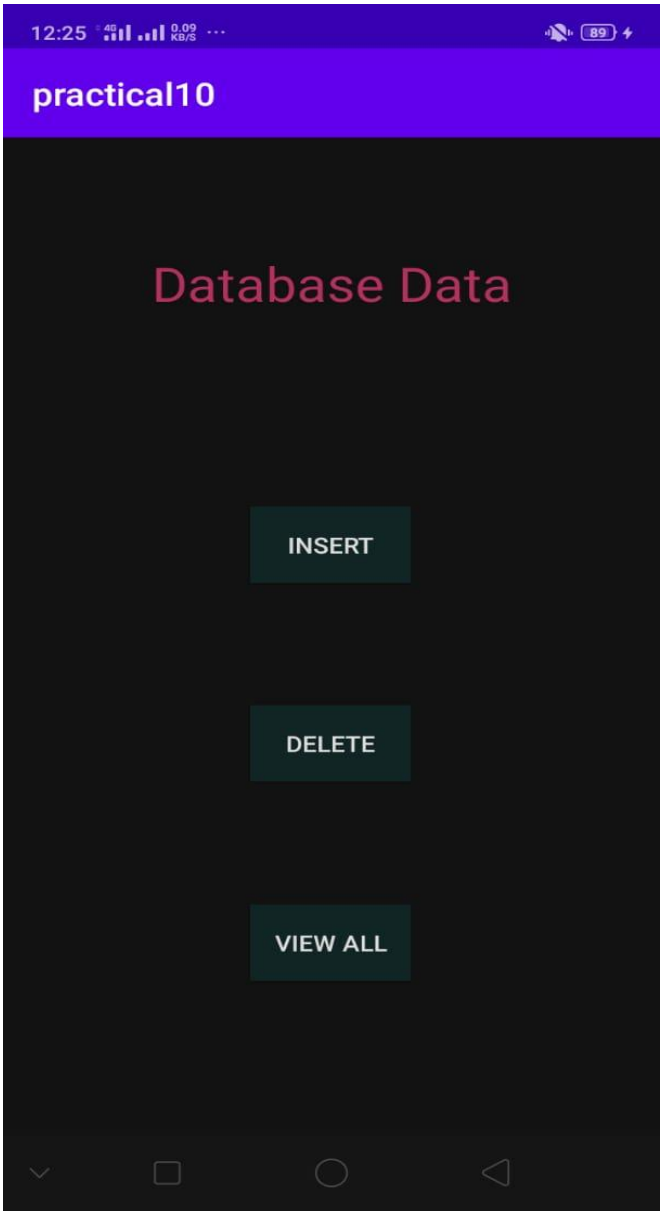
```
<?xml version="1.0" encoding="utf-8"?>
<androidx.constraintlayout.widget.ConstraintLayout
xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:app="http://schemas.android.com/apk/res-auto"
    xmlns:tools="http://schemas.android.com/tools"
    android:layout_width="match_parent"
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

    <EditText
        android:id="@+id/editText"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_marginTop="156dp"
        android:layout_marginEnd="30dp"
        android:layout_marginRight="30dp"
        android:layout_marginBottom="60dp"
        android:ems="10"
        android:hint="Name"
        android:inputType="textPersonName"
        app:layout_constraintBottom_toTopOf="@+id/editText2"
        app:layout_constraintEnd_toEndOf="parent"
        app:layout_constraintTop_toTopOf="parent" />

    <TextView
        android:id="@+id/textView"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_marginStart="58dp"
        android:layout_marginLeft="58dp"
        android:layout_marginTop="96dp"
        android:layout_marginEnd="56dp"
        android:layout_marginRight="56dp"
        android:text="Name"
        android:textColor="#B53160"
        android:textSize="20sp"
        app:layout_constraintEnd_toStartOf="@+id/editText"
```

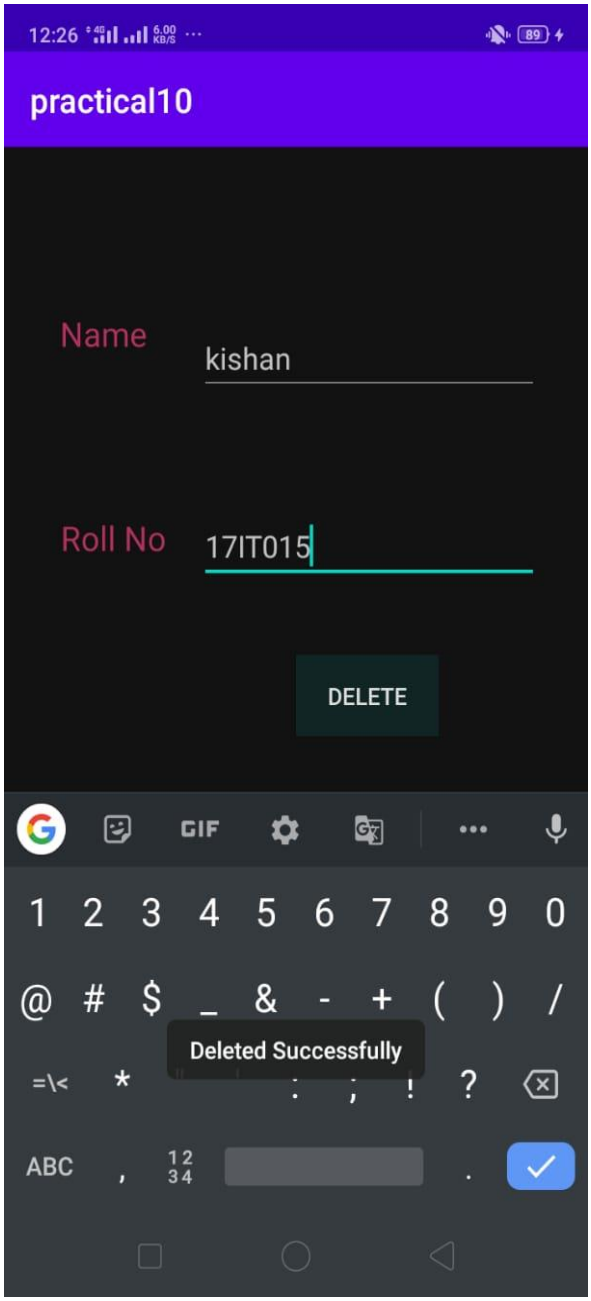
```
    app:layout_constraintStart_toStartOf="parent"
    app:layout_constraintTop_toTopOf="parent" />

  <EditText
    android:id="@+id/editText2"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:layout_marginTop="60dp"
    android:layout_marginEnd="30dp"
    android:layout_marginRight="30dp"
    android:layout_marginBottom="425dp"
    android:ems="10"
    android:hint="Roll No"
    android:inputType="textPersonName"
    app:layout_constraintBottom_toBottomOf="parent"
    app:layout_constraintEnd_toEndOf="parent"
    app:layout_constraintTop_toBottomOf="@+id/editText" />

  <TextView
    android:id="@+id/textView2"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:layout_marginStart="58dp"
    android:layout_marginLeft="58dp"
    android:layout_marginTop="145dp"
    android:layout_marginEnd="20dp"
    android:layout_marginRight="20dp"
    android:layout_marginBottom="437dp"
    android:text="Roll No"
    android:textColor="#B53160"
    android:textSize="20sp"
    app:layout_constraintBottom_toBottomOf="parent"
    app:layout_constraintEnd_toStartOf="@+id/editText2"
    app:layout_constraintHorizontal_bias="1.0"
    app:layout_constraintStart_toStartOf="parent"
    app:layout_constraintTop_toBottomOf="@+id/textView" />

  <Button
    android:id="@+id/button"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:layout_marginStart="180dp"
    android:layout_marginLeft="180dp"
    android:layout_marginTop="58dp"
    android:layout_marginEnd="92dp"
    android:layout_marginRight="92dp"
    android:layout_marginBottom="299dp"
    android:background="#FFE5E5"
    android:text="Delete"
    android:textColor="#000"
    app:layout_constraintBottom_toBottomOf="parent"
```

```
        app:layout_constraintEnd_toEndOf="parent"
        app:layout_constraintStart_toStartOf="parent"
        app:layout_constraintTop_toBottomOf="@+id/editText2" />

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

Output :







LATEST APPLICATIONS:

Altibase , MongoDB , Oracle RDBMS , IBM DB2 , Redis etc…

LEARNING OUTCOME:

In this Practical , we learnt how to use database in android application using SQLite.We also learnt how to create database and table.Then how to insert Data in the database.

PRACTICAL: 11

AIM:

Creating an application that provides Single Sign-on (SSO) with Chrome Custom Tabs via the AppAuth library, and optionally push managed configuration to provide a user login hint.

THEORY:

Single sign-on: Single sign-on is a property of access control of multiple related, yet independent, software systems. With this property, a user logs in with a single ID and password to gain access to any of several related systems.

GoogleSignInAccount: Class that holds the basic account information of the signed in Google user.

GoogleSignInClient: A client for interacting with the Google Sign In API.

CODE:

```
// MainActivity.java
package com.example.wcmc_15_prac_11;
import androidx.annotation.Nullable;
import androidx.appcompat.app.AppCompatActivity;
import android.content.Intent;
import android.os.Bundle;
import android.util.Log;
import android.view.View;
import android.widget.Toast;
import com.google.android.gms.auth.api.signin.GoogleSignIn;
import com.google.android.gms.auth.api.signin.GoogleSignInAccount;
import com.google.android.gms.auth.api.signin.GoogleSignInClient;
import com.google.android.gms.auth.api.signin.GoogleSignInOptions;
import com.google.android.gms.common.SignInButton;
import com.google.android.gms.common.api.ApiException;
import com.google.android.gms.tasks.Task;
public class MainActivity extends AppCompatActivity {
   private GoogleSignInClient mGoogleSignInClient;
   private GoogleSignInOptions gso;
   @Override
   protected void onCreate(Bundle savedInstanceState) {
      super.onCreate(savedInstanceState);
      setContentView(R.layout.activity_main);
      gso = new
GoogleSignInOptions.Builder(GoogleSignInOptions.DEFAULT_SIGN_IN)
            .requestEmail()
            .build();
      mGoogleSignInClient = GoogleSignIn.getClient(this, gso);
      SignInButton signInButton = findViewById(R.id.sign_in_button);
      signInButton.setSize(SignInButton.SIZE_STANDARD);
      signInButton.setOnClickListener(new View.OnClickListener() {
         @Override
```

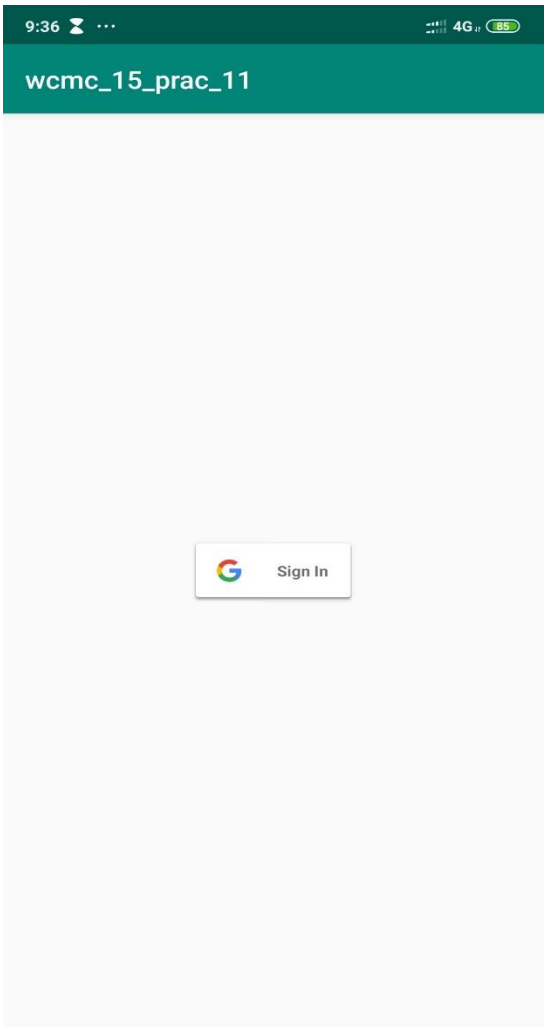
```
            public void onClick(View v) {
                switch (v.getId()) {
                    case R.id.sign_in_button:
                        Intent signInIntent = mGoogleSignInClient.getSignInIntent();
                        startActivityForResult(signInIntent, 9411);
                        break;
                }
            }
        });
    }
    @Override
    protected void onStart() {
        super.onStart();
        GoogleSignInAccount account = GoogleSignIn.getLastSignedInAccount(this);
        if ( account == null) return;
        updateUI(account);
    }
    private void handleSignInResult(Task<GoogleSignInAccount> completedTask) {
        try {
            GoogleSignInAccount account = completedTask.getResult(ApiException.class);
            updateUI(account);
        } catch (ApiException e) {
            Log.e("SSO","signInResult:failed code=" + e.getStatusCode());
        }
    }
    private void updateUI(GoogleSignInAccount account) {
        Toast.makeText(this,"Welcome " +
account.getDisplayName(),Toast.LENGTH_LONG).show();
    }
    @Override
    protected void onActivityResult(int requestCode, int resultCode, @Nullable Intent data)
{
        super.onActivityResult(requestCode, resultCode, data);
        if (requestCode == 9411) {
            Task<GoogleSignInAccount> task =
GoogleSignIn.getSignedInAccountFromIntent(data);
            handleSignInResult(task);
        }
    }
}
```

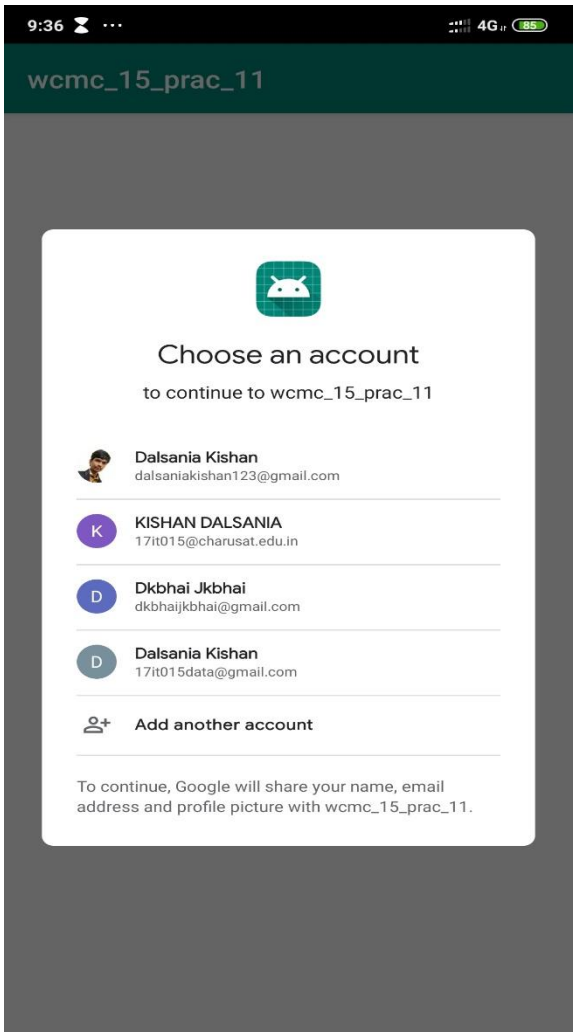
activity_mail.xml

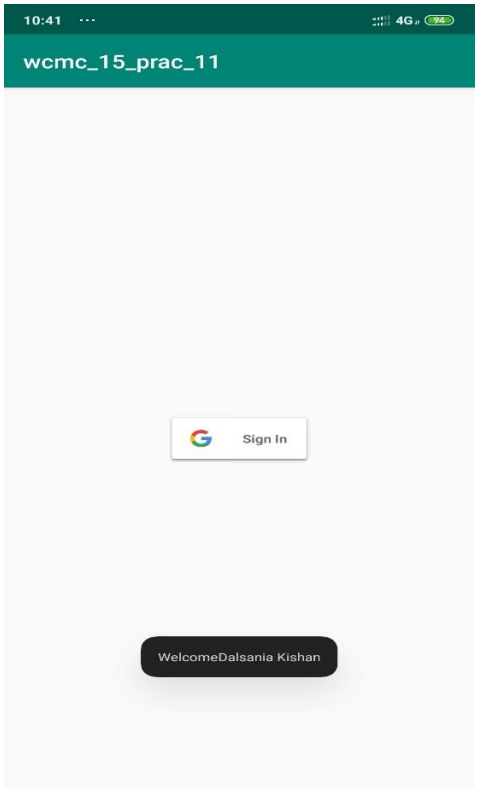
```
<?xml version="1.0" encoding="utf-8"?>
<LinearLayout xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:app="http://schemas.android.com/apk/res-auto"
    xmlns:tools="http://schemas.android.com/tools"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    android:gravity="center"
    tools:context=".MainActivity">
    <com.google.android.gms.common.SignInButton
        android:id="@+id/sign_in_button"
```

```
        android:layout_width="wrap_content"
        android:layout_height="wrap_content" />
    </LinearLayout>
```

OUTPUT:







LATEST APPLICATIONS:

Edmodo, Udemy, Coursera.

LEARNING OUTCOME:

1. Working with google sign in options.
2. Working with android menu layout.

PRACTICAL: 14

AIM:

Create an application that uses end-to-end process of training a machine learning model that can recognize handwritten digit images with TensorFlow and deploy it to an Android app.

THEORY:

The MNIST database of handwritten digits, available from this page, has a training set of 60,000 examples, and a test set of 10,000 examples. It is a subset of a larger set available from NIST. The digits have been size-normalized and centered in a fixed-size image. It is a good database for people who want to try learning techniques and pattern recognition methods on real-world data while spending minimal efforts on preprocessing and formatting.

TensorFlow Lite is a set of tools to help developers run TensorFlow models on mobile, embedded, and IoT devices. It enables on-device machine learning inference with low latency and a small binary size. TensorFlow Lite consists of two main components: The TensorFlow Lite interpreter, which runs specially optimized models on many different hardware types, including mobile phones, embedded Linux devices, and microcontrollers. The TensorFlow Lite converter, which converts TensorFlow models into an efficient form for use by the interpreter, and can introduce optimizations to improve binary size and performance.

TensorFlow Lite is designed to make it easy to perform machine learning on devices, "at the edge" of the network, instead of sending data back and forth from a server. For developers, performing machine learning on-device can help improve:

- Latency*:* there's no round-trip to a server
- Privacy*:* no data needs to leave the device
- Connectivity*:* an Internet connection isn't required
- Power consumption*:* network connections are power hungry

TensorFlow Lite works with a huge range of devices, from tiny microcontrollers to powerful mobile phones.

CODE:

MainActvity.java

```
package com.example.prac_14_wcmc_17it015;

import android.graphics.Bitmap;
import android.os.Bundle;

import android.util.Log;
import android.widget.TextView;
import android.widget.Toast;

import com.nex3z.fingerpaintview.FingerPaintView;
```

```
import java.io.IOException;

import androidx.appcompat.app.AppCompatActivity;
import butterknife.BindView;
import butterknife.ButterKnife;
import butterknife.OnClick;

public class MainActivity extends AppCompatActivity {
    private static final String LOG_TAG = "lele";

    @BindView(R.id.fpv_paint) FingerPaintView mFpvPaint;
    @BindView(R.id.tv_prediction) TextView mTvPrediction;
    @BindView(R.id.tv_probability) TextView mTvProbability;
    @BindView(R.id.tv_timecost) TextView mTvTimeCost;

    private Classifier mClassifier;

    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_main);
        ButterKnife.bind(this);
        init();
    }

    @OnClick(R.id.btn_detect)
    void onDetectClick() {
        if (mClassifier == null) {
            Log.e(LOG_TAG, "onDetectClick(): Classifier is not initialized");
            return;
        } else if (mFpvPaint.isEmpty()) {
            Toast.makeText(this, R.string.please_write_a_digit, Toast.LENGTH_SHORT).show();
            return;
        }

        Bitmap image = mFpvPaint.exportToBitmap(
                Classifier.IMG_WIDTH, Classifier.IMG_HEIGHT);
        Result result = mClassifier.classify(image);
        renderResult(result);
    }

    @OnClick(R.id.btn_clear)
    void onClearClick() {
        mFpvPaint.clear();
        mTvPrediction.setText(R.string.empty);
        mTvProbability.setText(R.string.empty);
        mTvTimeCost.setText(R.string.empty);
    }

    private void init() {
```

```
        try {
            mClassifier = new Classifier(MainActivity.this);
        } catch (IOException e) {
            Toast.makeText(this, e.getMessage().toString(), Toast.LENGTH_LONG).show();
            Log.e(LOG_TAG, "init(): Failed to create Classifier", e);
        }
    }

    private void renderResult(Result result) {
        mTvPrediction.setText(String.valueOf(result.getNumber()));
        mTvProbability.setText(String.valueOf(result.getProbability()));
        mTvTimeCost.setText(String.format(getString(R.string.timecost_value),
                result.getTimeCost()));
    }

}
```

classifier.java

```
package com.example.prac_14_wcmc_17it015;

import android.app.Activity;
import android.content.res.AssetFileDescriptor;
import android.graphics.Bitmap;
import android.os.SystemClock;
import android.util.Log;

import org.tensorflow.lite.Interpreter;

import java.io.FileInputStream;
import java.io.IOException;
import java.nio.ByteBuffer;
import java.nio.ByteOrder;
import java.nio.MappedByteBuffer;
import java.nio.channels.FileChannel;
import java.util.Arrays;

public class Classifier {
    private static final String LOG_TAG = Classifier.class.getSimpleName();

    private static final String MODEL_NAME = "mnist.tflite";

    private static final int BATCH_SIZE = 1;
    public static final int IMG_HEIGHT = 28;
    public static final int IMG_WIDTH = 28;
    private static final int NUM_CHANNEL = 1;
    private static final int NUM_CLASSES = 10;

    private final Interpreter.Options options = new Interpreter.Options();
    private final Interpreter mInterpreter;
    private final ByteBuffer mImageData;
    private final int[] mImagePixels = new int[IMG_HEIGHT * IMG_WIDTH];
    private final float[][] mResult = new float[1][NUM_CLASSES];
```

```
public Classifier(MainActivity activity) throws IOException {
   mInterpreter = new Interpreter(loadModelFile(activity), options);
   mImageData = ByteBuffer.allocateDirect(
         4 * BATCH_SIZE * IMG_HEIGHT * IMG_WIDTH * NUM_CHANNEL);
   mImageData.order(ByteOrder.nativeOrder());
}

public Result classify(Bitmap bitmap) {
   convertBitmapToByteBuffer(bitmap);
   long startTime = SystemClock.uptimeMillis();
   mInterpreter.run(mImageData, mResult);
   long endTime = SystemClock.uptimeMillis();
   long timeCost = endTime - startTime;
   Log.v(LOG_TAG, "classify(): result = " + Arrays.toString(mResult[0])
         + ", timeCost = " + timeCost);
   return new Result(mResult[0], timeCost);
}

private MappedByteBuffer loadModelFile(Activity activity) throws IOException {
   AssetFileDescriptor fileDescriptor = activity.getAssets().openFd(MODEL_NAME);
   FileInputStream inputStream = new FileInputStream(fileDescriptor.getFileDescriptor());
   FileChannel fileChannel = inputStream.getChannel();
   long startOffset = fileDescriptor.getStartOffset();
   long declaredLength = fileDescriptor.getDeclaredLength();
   return fileChannel.map(FileChannel.MapMode.READ_ONLY, startOffset, declaredLength);
}

private void convertBitmapToByteBuffer(Bitmap bitmap) {
   if (mImageData == null) {
      return;
   }
   mImageData.rewind();

   bitmap.getPixels(mImagePixels, 0, bitmap.getWidth(), 0, 0,
         bitmap.getWidth(), bitmap.getHeight());

   int pixel = 0;
   for (int i = 0; i < IMG_WIDTH; ++i) {
      for (int j = 0; j < IMG_HEIGHT; ++j) {
         int value = mImagePixels[pixel++];
         mImageData.putFloat(convertPixel(value));
      }
   }
}

private static float convertPixel(int color) {
   return (255 - (((color >> 16) & 0xFF) * 0.299f
         + ((color >> 8) & 0xFF) * 0.587f
         + (color & 0xFF) * 0.114f)) / 255.0f;
}
```

```
}
```

result.java

```
package com.example.prac_14_wcmc_17it015;

public class Result {

    private final int mNumber;
    private final float mProbability;
    private final long mTimeCost;

    public Result(float[] probs, long timeCost) {
        mNumber = argmax(probs);
        mProbability = probs[mNumber];
        mTimeCost = timeCost;
    }

    public int getNumber() {
        return mNumber;
    }

    public float getProbability() {
        return mProbability;
    }

    public long getTimeCost() {
        return mTimeCost;
    }

    private static int argmax(float[] probs) {
        int maxIdx = -1;
        float maxProb = 0.0f;
        for (int i = 0; i < probs.length; i++) {
            if (probs[i] > maxProb) {
                maxProb = probs[i];
                maxIdx = i;
            }
        }
        return maxIdx;
    }
}
```

App.gradle

```
apply plugin: 'com.android.application'

android {
    compileSdkVersion 28
    defaultConfig {
        applicationId "com.example.prac_14_wcmc_17it015"
        minSdkVersion 15
        targetSdkVersion 28
        versionCode 1
```

```
        versionName "1.0.0"
        testInstrumentationRunner "android.support.test.runner.AndroidJUnitRunner"
    }
    buildTypes {
        release {
            minifyEnabled false
            proguardFiles getDefaultProguardFile('proguard-android.txt'), 'proguard-rules.pro'
        }
    }
    aaptOptions {
        noCompress "tflite"
        noCompress "lite"
    }

    compileOptions {
        sourceCompatibility JavaVersion.VERSION_1_8
        targetCompatibility JavaVersion.VERSION_1_8
    }
}

dependencies {
    implementation fileTree(dir: 'libs', include: ['*.jar'])
    implementation 'com.android.support:appcompat-v7:28.0.0'
    implementation 'com.android.support:support-v4:28.0.0'

    implementation 'com.nex3z:finger-paint-view:0.1.0'
    implementation 'org.tensorflow:tensorflow-lite:1.13.1'

    implementation 'com.jakewharton:butterknife:10.0.0'
    annotationProcessor 'com.jakewharton:butterknife-compiler:10.0.0'

    testImplementation 'junit:junit:4.12'
    androidTestImplementation 'com.android.support.test:runner:1.0.2'
    androidTestImplementation 'com.android.support.test.espresso:espresso-core:3.0.2'
}
```

activity_main.xml

```
<?xml version="1.0" encoding="utf-8"?>
<LinearLayout xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:tools="http://schemas.android.com/tools"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    android:orientation="vertical"
    android:padding="16dp"
    tools:context=".MainActivity">

    <TableLayout
        android:layout_width="match_parent"
        android:layout_height="wrap_content">

        <TextView
            style="@style/ResultText"
```

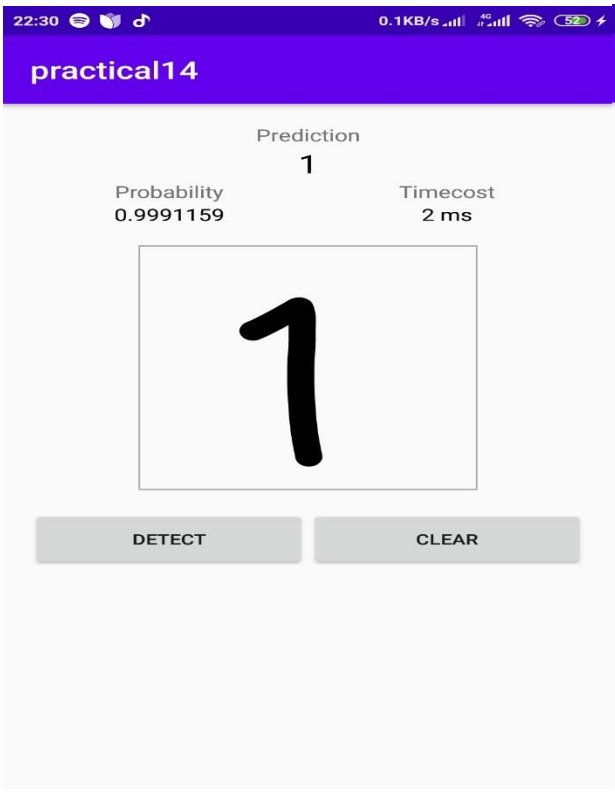
```
        android:text="@string/prediction"/>

    <TextView
        android:id="@+id/tv_prediction"
        style="@style/ResultText"
        android:textSize="24sp"
        android:textColor="@android:color/black"
        android:text="@string/empty"
        tools:text="1"/>

    <TableRow>

        <TextView
            style="@style/ResultText"
            android:text="@string/probability"/>

        <TextView
            style="@style/ResultText"
            android:text="@string/timecost"/>

    </TableRow>

    <TableRow>

        <TextView
            android:id="@+id/tv_probability"
            style="@style/ResultText"
            android:textColor="@android:color/black"
            android:text="@string/empty"
            tools:text="0.9"/>


        <TextView
            android:id="@+id/tv_timecost"
            style="@style/ResultText"
            android:textColor="@android:color/black"
            android:text="@string/empty"
            tools:text="10ms"/>

    </TableRow>

</TableLayout>

<com.nex3z.fingerpaintview.FingerPaintView
    android:id="@+id/fpv_paint"
    android:layout_width="200dp"
    android:layout_height="200dp"
    android:layout_gravity="center"
    android:layout_marginTop="16dp"
    android:foreground="@drawable/shape_rect_border"/>
```

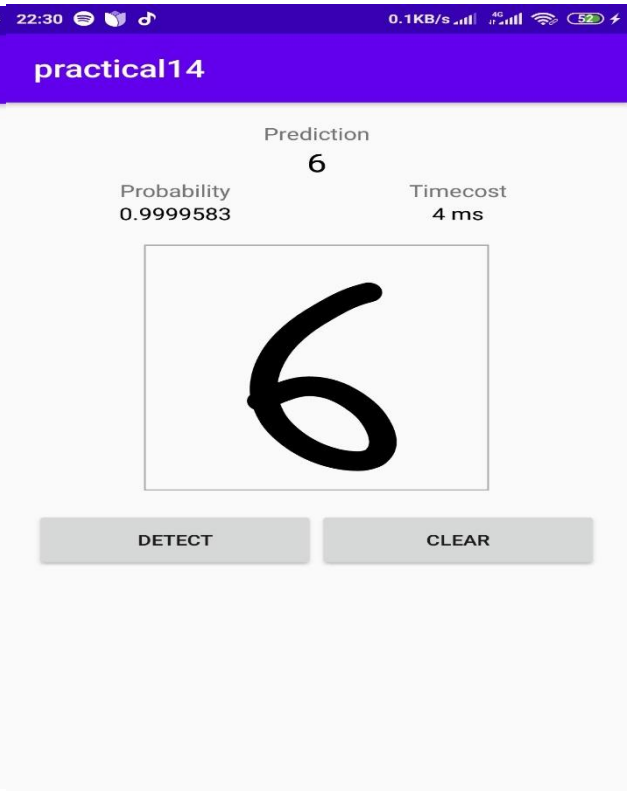
```
    <LinearLayout
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:layout_marginTop="16dp"
        android:orientation="horizontal">

        <Button
            android:id="@+id/btn_detect"
            android:layout_width="match_parent"
            android:layout_height="wrap_content"
            android:layout_weight="1"
            android:text="@string/detect"/>

        <Button
            android:id="@+id/btn_clear"
            android:layout_width="match_parent"
            android:layout_height="wrap_content"
            android:layout_weight="1"
            android:text="@string/clear"/>
    </LinearLayout>
</LinearLayout>
```

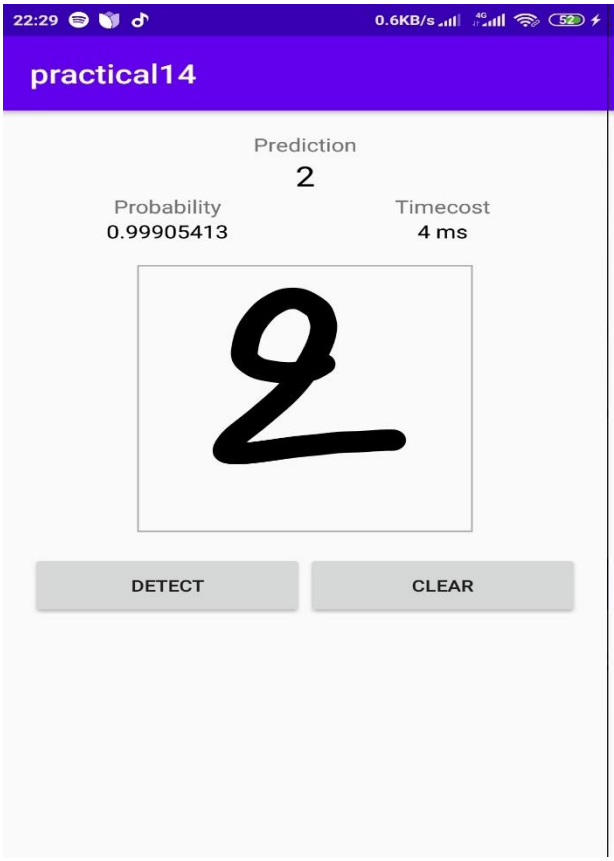
AndroidManifest.xml

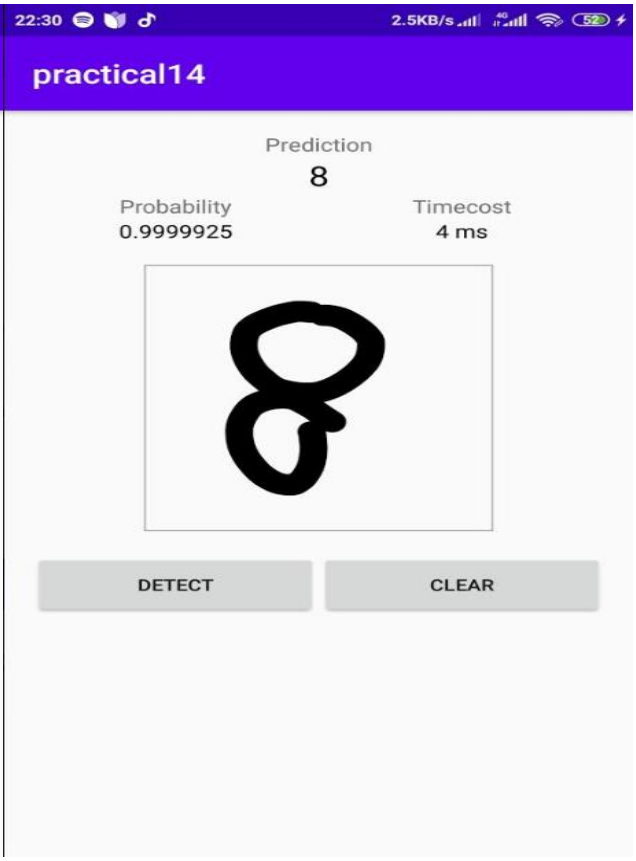
```
<?xml version="1.0" encoding="utf-8"?>
<manifest xmlns:android="http://schemas.android.com/apk/res/android"
    package="com.example.prac_13_wcmc_17it015">

    <application
        android:allowBackup="true"
        android:icon="@mipmap/ic_launcher"
        android:label="@string/app_name"
        android:roundIcon="@mipmap/ic_launcher_round"
        android:supportsRtl="true"
        android:theme="@style/AppTheme">
        <activity android:name=".MainActivity">
            <intent-filter>
                <action android:name="android.intent.action.MAIN" />

                <category android:name="android.intent.category.LAUNCHER" />
            </intent-filter>
        </activity>
    </application>

</manifest>
```

OUTPUT:


22:30
0.1KB/s
practical14
Prediction
1
Probability
0.9991159
Timecost
2 ms
DETECT
CLEAR


22:30
0.1KB/s
practical14
Prediction
6
Probability
0.9999583
Timecost
4 ms
DETECT
CLEAR


22:29
0.6KB/s
practical14
Prediction
2
Probability
0.99905413
Timecost
4 ms
DETECT
CLEAR


22:30
2.5KB/s
practical14
Prediction
8
Probability
0.9999925
Timecost
4 ms
DETECT
CLEAR

LATEST APPLICATIONS:

Netflix, Tinder, Snapchat.

LEARNING OUTCOME:

I learnt how to work with machine learning algorithms in android studio as well as I also learnt how to combine android studio and trainng an algorithm and implement in real application.

PRACTICAL: 13

AIM:

Create an application to play video using YouTube API in PIP mode

THEORY:

PIP is a special type of multi-window mode mainly used for activities that need to be active on screen but should not take up the whole screen space like watching videos, video calls, navigation, etc. It lets the user watch a video in a small window pinned to a corner of the screen (by default bottom right) while navigating between apps or browsing content on the main screen. Android 8.0 (API level 26) and above allows activities to launch in PIP mode.

The PIP window appears in the top-most layer of the screen. You can drag the PIP window to another location using some special toggles. When you tap on the window two special controls appear:

- a full-screen toggle (in the centre of the window) and
- a close button (an "X" in the upper right corner).

CODE:

MainActvity.java

```
package com.example.prac_13_wcmc_17it015;

import androidx.appcompat.app.AppCompatActivity;
import android.app.ActionBar;
import android.app.Notification;
import android.app.PictureInPictureParams;
import android.drm.DrmStore;
import android.graphics.Point;
import android.net.Uri;
import android.os.Build;
import android.os.Bundle;
import android.util.Rational;
import android.view.Display;
import android.view.View;
import android.widget.Button;
import android.widget.MediaController;
import android.widget.VideoView;

public class MainActivity extends AppCompatActivity {

    Button pipbtn;
    String path = "/storage/DCIM/Camera/movie.mp4";
    ActionBar actionBar;
    VideoView video;

    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
```

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

        video = (VideoView)findViewById(R.id.video);
        actionBar = getActionBar();
        MediaController mediaController= new MediaController(this);
        mediaController.setAnchorView(video);
        video.setMediaController(mediaController);
        video.setVideoURI(Uri.parse(path));
        video.requestFocus();
        video.start();

        pipbtn = (Button)findViewById(R.id.pipbtn);

        pipbtn.setOnClickListener(new View.OnClickListener() {
            @Override
            public void onClick(View view) {
                Display display = getWindowManager().getDefaultDisplay();
                Point point = new Point();
                display.getSize(point);
                int width = point.x;
                int height = point.y;
                Rational ratio = null;
                if (Build.VERSION.SDK_INT >= Build.VERSION_CODES.LOLLIPOP) {
                    ratio = new Rational(width,height);
                }
                PictureInPictureParams.Builder pip_builder = null;
                if (Build.VERSION.SDK_INT >= Build.VERSION_CODES.O) {
                    pip_builder = new PictureInPictureParams.Builder();
                }
                if (Build.VERSION.SDK_INT >= Build.VERSION_CODES.O) {
                    pip_builder.setAspectRatio(ratio).build();
                }
                pipbtn.setVisibility(View.INVISIBLE);
                if (Build.VERSION.SDK_INT >= Build.VERSION_CODES.O) {
                    enterPictureInPictureMode(pip_builder.build());
                }

            }
        });
    }
}
```

activity_main.xml

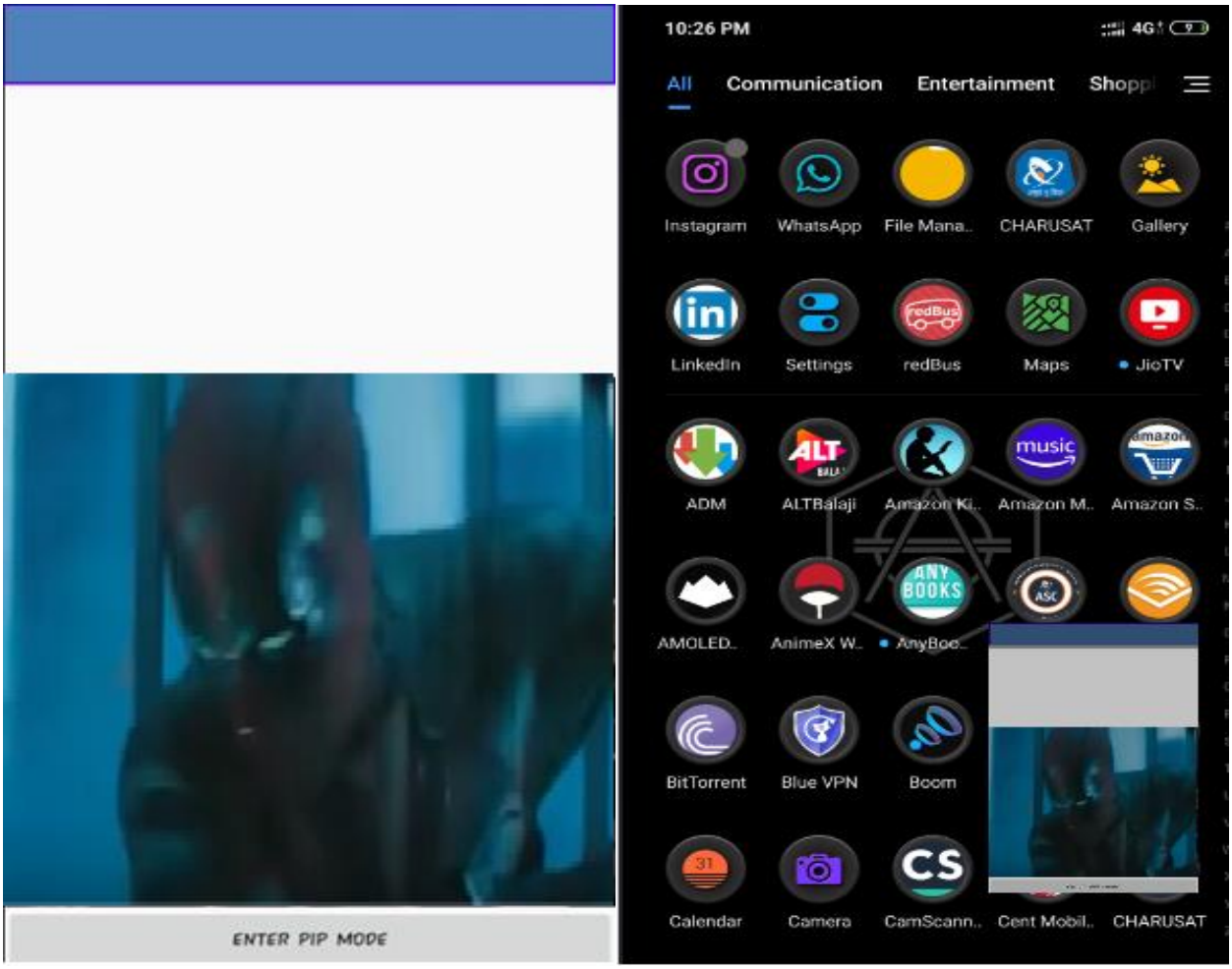
```
<?xml version="1.0" encoding="utf-8"?>
<RelativeLayout
    xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:app="http://schemas.android.com/apk/res-auto"
    xmlns:tools="http://schemas.android.com/tools"
    android:layout_width="match_parent"
    android:layout_height="match_parent"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

```
        android:layout_width="match_parent"
        android:layout_height="match_parent"
        android:id="@+id/video"
        android:layout_above="@id/pipbtn"/>
    <Button
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:text="Enter PIP mode"
        android:layout_alignParentBottom="true"
        android:id="@+id/pipbtn"/>
</RelativeLayout>
```

AndroidManifest.xml

```
<?xml version="1.0" encoding="utf-8"?>
<manifest xmlns:android="http://schemas.android.com/apk/res/android"
    package="com.example.prac_13_wcmc_17it015">

    <application
        android:allowBackup="true"
        android:icon="@mipmap/ic_launcher"
        android:label="@string/app_name"
        android:roundIcon="@mipmap/ic_launcher_round"
        android:supportsRtl="true"
        android:theme="@style/AppTheme">
        <activity android:name=".MainActivity">
            <intent-filter>
                <action android:name="android.intent.action.MAIN" />

                <category android:name="android.intent.category.LAUNCHER" />
            </intent-filter>
        </activity>
    </application>

</manifest>
```

OUTPUT:



LATEST APPLICATIONS:

Zomato, Swiggy , Youtube Vanced.

LEARNING OUTCOME:

I learnt how to work with video in android studio as well as I also learnt pip mode and implement in real application.

PRACTICAL: 14

AIM:

Create an application that uses end-to-end process of training a machine learning model that can recognize handwritten digit images with TensorFlow and deploy it to an Android app.

THEORY:

The MNIST database of handwritten digits, available from this page, has a training set of 60,000 examples, and a test set of 10,000 examples. It is a subset of a larger set available from NIST. The digits have been size-normalized and centered in a fixed-size image. It is a good database for people who want to try learning techniques and pattern recognition methods on real-world data while spending minimal efforts on preprocessing and formatting.

TensorFlow Lite is a set of tools to help developers run TensorFlow models on mobile, embedded, and IoT devices. It enables on-device machine learning inference with low latency and a small binary size. TensorFlow Lite consists of two main components: The TensorFlow Lite interpreter, which runs specially optimized models on many different hardware types, including mobile phones, embedded Linux devices, and microcontrollers. The TensorFlow Lite converter, which converts TensorFlow models into an efficient form for use by the interpreter, and can introduce optimizations to improve binary size and performance.

TensorFlow Lite is designed to make it easy to perform machine learning on devices, "at the edge" of the network, instead of sending data back and forth from a server. For developers, performing machine learning on-device can help improve:

- Latency*:* there's no round-trip to a server
- Privacy*:* no data needs to leave the device
- Connectivity*:* an Internet connection isn't required
- Power consumption*:* network connections are power hungry

TensorFlow Lite works with a huge range of devices, from tiny microcontrollers to powerful mobile phones.

CODE:

MainActvity.java

```
package com.example.prac_14_wcmc_17it015;

import android.graphics.Bitmap;
import android.os.Bundle;

import android.util.Log;
import android.widget.TextView;
import android.widget.Toast;

import com.nex3z.fingerpaintview.FingerPaintView;
```

```
import java.io.IOException;

import androidx.appcompat.app.AppCompatActivity;
import butterknife.BindView;
import butterknife.ButterKnife;
import butterknife.OnClick;

public class MainActivity extends AppCompatActivity {
    private static final String LOG_TAG = "lele";

    @BindView(R.id.fpv_paint) FingerPaintView mFpvPaint;
    @BindView(R.id.tv_prediction) TextView mTvPrediction;
    @BindView(R.id.tv_probability) TextView mTvProbability;
    @BindView(R.id.tv_timecost) TextView mTvTimeCost;

    private Classifier mClassifier;

    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_main);
        ButterKnife.bind(this);
        init();
    }

    @OnClick(R.id.btn_detect)
    void onDetectClick() {
        if (mClassifier == null) {
            Log.e(LOG_TAG, "onDetectClick(): Classifier is not initialized");
            return;
        } else if (mFpvPaint.isEmpty()) {
            Toast.makeText(this, R.string.please_write_a_digit, Toast.LENGTH_SHORT).show();
            return;
        }

        Bitmap image = mFpvPaint.exportToBitmap(
                Classifier.IMG_WIDTH, Classifier.IMG_HEIGHT);
        Result result = mClassifier.classify(image);
        renderResult(result);
    }

    @OnClick(R.id.btn_clear)
    void onClearClick() {
        mFpvPaint.clear();
        mTvPrediction.setText(R.string.empty);
        mTvProbability.setText(R.string.empty);
        mTvTimeCost.setText(R.string.empty);
    }

    private void init() {
```

```
        try {
            mClassifier = new Classifier(MainActivity.this);
        } catch (IOException e) {
            Toast.makeText(this, e.getMessage().toString(), Toast.LENGTH_LONG).show();
            Log.e(LOG_TAG, "init(): Failed to create Classifier", e);
        }
    }

    private void renderResult(Result result) {
        mTvPrediction.setText(String.valueOf(result.getNumber()));
        mTvProbability.setText(String.valueOf(result.getProbability()));
        mTvTimeCost.setText(String.format(getString(R.string.timecost_value),
                result.getTimeCost()));
    }

}
```

classifier.java

```
package com.example.prac_14_wcmc_17it015;

import android.app.Activity;
import android.content.res.AssetFileDescriptor;
import android.graphics.Bitmap;
import android.os.SystemClock;
import android.util.Log;

import org.tensorflow.lite.Interpreter;

import java.io.FileInputStream;
import java.io.IOException;
import java.nio.ByteBuffer;
import java.nio.ByteOrder;
import java.nio.MappedByteBuffer;
import java.nio.channels.FileChannel;
import java.util.Arrays;

public class Classifier {
    private static final String LOG_TAG = Classifier.class.getSimpleName();

    private static final String MODEL_NAME = "mnist.tflite";

    private static final int BATCH_SIZE = 1;
    public static final int IMG_HEIGHT = 28;
    public static final int IMG_WIDTH = 28;
    private static final int NUM_CHANNEL = 1;
    private static final int NUM_CLASSES = 10;

    private final Interpreter.Options options = new Interpreter.Options();
    private final Interpreter mInterpreter;
    private final ByteBuffer mImageData;
    private final int[] mImagePixels = new int[IMG_HEIGHT * IMG_WIDTH];
    private final float[][] mResult = new float[1][NUM_CLASSES];
```

```
public Classifier(MainActivity activity) throws IOException {
   mInterpreter = new Interpreter(loadModelFile(activity), options);
   mImageData = ByteBuffer.allocateDirect(
         4 * BATCH_SIZE * IMG_HEIGHT * IMG_WIDTH * NUM_CHANNEL);
   mImageData.order(ByteOrder.nativeOrder());
}

public Result classify(Bitmap bitmap) {
   convertBitmapToByteBuffer(bitmap);
   long startTime = SystemClock.uptimeMillis();
   mInterpreter.run(mImageData, mResult);
   long endTime = SystemClock.uptimeMillis();
   long timeCost = endTime - startTime;
   Log.v(LOG_TAG, "classify(): result = " + Arrays.toString(mResult[0])
         + ", timeCost = " + timeCost);
   return new Result(mResult[0], timeCost);
}

private MappedByteBuffer loadModelFile(Activity activity) throws IOException {
   AssetFileDescriptor fileDescriptor = activity.getAssets().openFd(MODEL_NAME);
   FileInputStream inputStream = new FileInputStream(fileDescriptor.getFileDescriptor());
   FileChannel fileChannel = inputStream.getChannel();
   long startOffset = fileDescriptor.getStartOffset();
   long declaredLength = fileDescriptor.getDeclaredLength();
   return fileChannel.map(FileChannel.MapMode.READ_ONLY, startOffset, declaredLength);
}

private void convertBitmapToByteBuffer(Bitmap bitmap) {
   if (mImageData == null) {
      return;
   }
   mImageData.rewind();

   bitmap.getPixels(mImagePixels, 0, bitmap.getWidth(), 0, 0,
         bitmap.getWidth(), bitmap.getHeight());

   int pixel = 0;
   for (int i = 0; i < IMG_WIDTH; ++i) {
      for (int j = 0; j < IMG_HEIGHT; ++j) {
         int value = mImagePixels[pixel++];
         mImageData.putFloat(convertPixel(value));
      }
   }
}

private static float convertPixel(int color) {
   return (255 - (((color >> 16) & 0xFF) * 0.299f
         + ((color >> 8) & 0xFF) * 0.587f
         + (color & 0xFF) * 0.114f)) / 255.0f;
}
```

```
}
```

result.java

```
package com.example.prac_14_wcmc_17it015;

public class Result {

    private final int mNumber;
    private final float mProbability;
    private final long mTimeCost;

    public Result(float[] probs, long timeCost) {
        mNumber = argmax(probs);
        mProbability = probs[mNumber];
        mTimeCost = timeCost;
    }

    public int getNumber() {
        return mNumber;
    }

    public float getProbability() {
        return mProbability;
    }

    public long getTimeCost() {
        return mTimeCost;
    }

    private static int argmax(float[] probs) {
        int maxIdx = -1;
        float maxProb = 0.0f;
        for (int i = 0; i < probs.length; i++) {
            if (probs[i] > maxProb) {
                maxProb = probs[i];
                maxIdx = i;
            }
        }
        return maxIdx;
    }
}
```

App.gradle

```
apply plugin: 'com.android.application'

android {
    compileSdkVersion 28
    defaultConfig {
        applicationId "com.example.prac_14_wcmc_17it015"
        minSdkVersion 15
        targetSdkVersion 28
        versionCode 1
```

```
        versionName "1.0.0"
        testInstrumentationRunner "android.support.test.runner.AndroidJUnitRunner"
    }
    buildTypes {
        release {
            minifyEnabled false
            proguardFiles getDefaultProguardFile('proguard-android.txt'), 'proguard-rules.pro'
        }
    }
    aaptOptions {
        noCompress "tflite"
        noCompress "lite"
    }

    compileOptions {
        sourceCompatibility JavaVersion.VERSION_1_8
        targetCompatibility JavaVersion.VERSION_1_8
    }
}

dependencies {
    implementation fileTree(dir: 'libs', include: ['*.jar'])
    implementation 'com.android.support:appcompat-v7:28.0.0'
    implementation 'com.android.support:support-v4:28.0.0'

    implementation 'com.nex3z:finger-paint-view:0.1.0'
    implementation 'org.tensorflow:tensorflow-lite:1.13.1'

    implementation 'com.jakewharton:butterknife:10.0.0'
    annotationProcessor 'com.jakewharton:butterknife-compiler:10.0.0'

    testImplementation 'junit:junit:4.12'
    androidTestImplementation 'com.android.support.test:runner:1.0.2'
    androidTestImplementation 'com.android.support.test.espresso:espresso-core:3.0.2'
}
```

activity_main.xml

```
<?xml version="1.0" encoding="utf-8"?>
<LinearLayout xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:tools="http://schemas.android.com/tools"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    android:orientation="vertical"
    android:padding="16dp"
    tools:context=".MainActivity">

    <TableLayout
        android:layout_width="match_parent"
        android:layout_height="wrap_content">

        <TextView
            style="@style/ResultText"
```

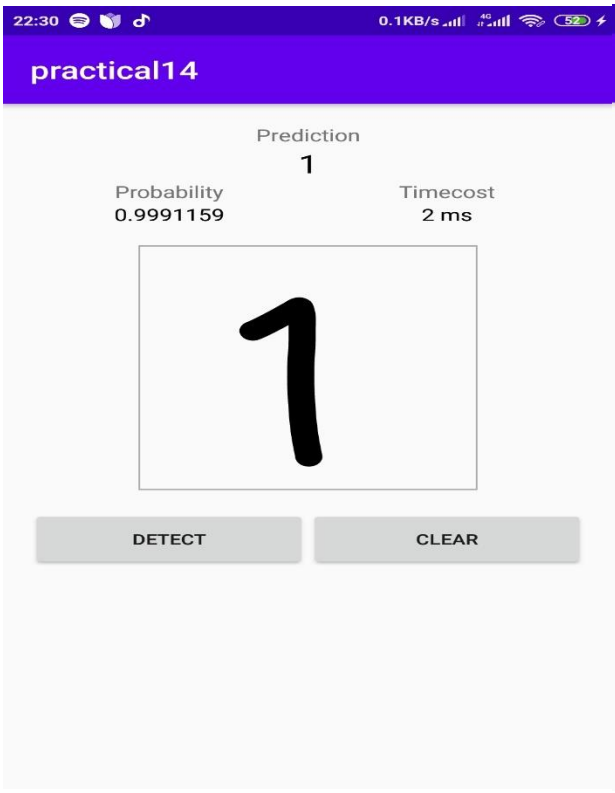
```
        android:text="@string/prediction"/>

    <TextView
        android:id="@+id/tv_prediction"
        style="@style/ResultText"
        android:textSize="24sp"
        android:textColor="@android:color/black"
        android:text="@string/empty"
        tools:text="1"/>

    <TableRow>

        <TextView
            style="@style/ResultText"
            android:text="@string/probability"/>

        <TextView
            style="@style/ResultText"
            android:text="@string/timecost"/>

    </TableRow>

    <TableRow>

        <TextView
            android:id="@+id/tv_probability"
            style="@style/ResultText"
            android:textColor="@android:color/black"
            android:text="@string/empty"
            tools:text="0.9"/>


        <TextView
            android:id="@+id/tv_timecost"
            style="@style/ResultText"
            android:textColor="@android:color/black"
            android:text="@string/empty"
            tools:text="10ms"/>

    </TableRow>

</TableLayout>

<com.nex3z.fingerpaintview.FingerPaintView
    android:id="@+id/fpv_paint"
    android:layout_width="200dp"
    android:layout_height="200dp"
    android:layout_gravity="center"
    android:layout_marginTop="16dp"
    android:foreground="@drawable/shape_rect_border"/>
```

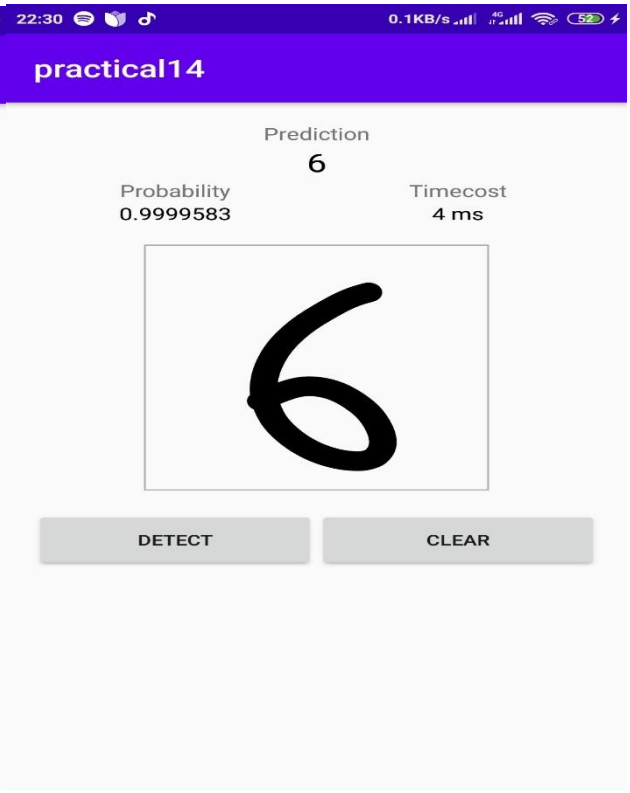
```
    <LinearLayout
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:layout_marginTop="16dp"
        android:orientation="horizontal">

        <Button
            android:id="@+id/btn_detect"
            android:layout_width="match_parent"
            android:layout_height="wrap_content"
            android:layout_weight="1"
            android:text="@string/detect"/>

        <Button
            android:id="@+id/btn_clear"
            android:layout_width="match_parent"
            android:layout_height="wrap_content"
            android:layout_weight="1"
            android:text="@string/clear"/>
    </LinearLayout>
</LinearLayout>
```

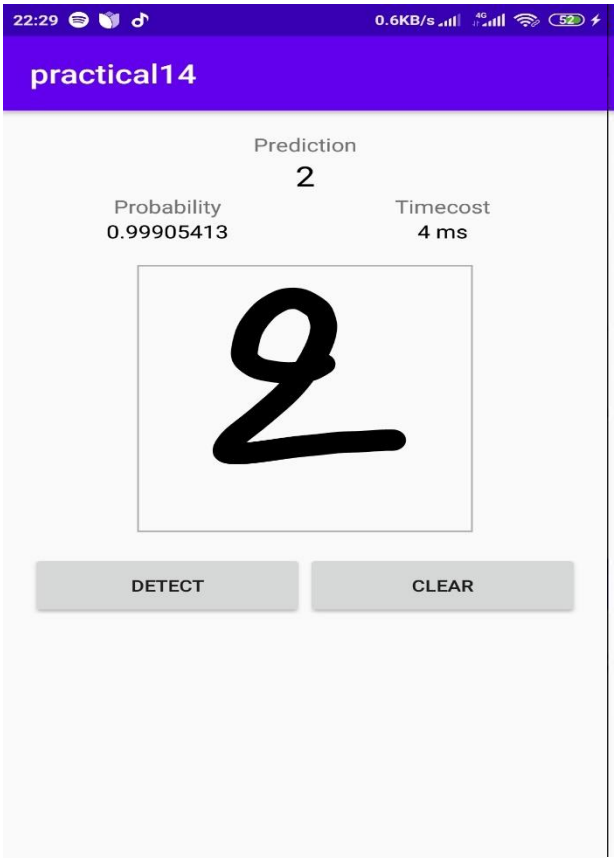
AndroidManifest.xml

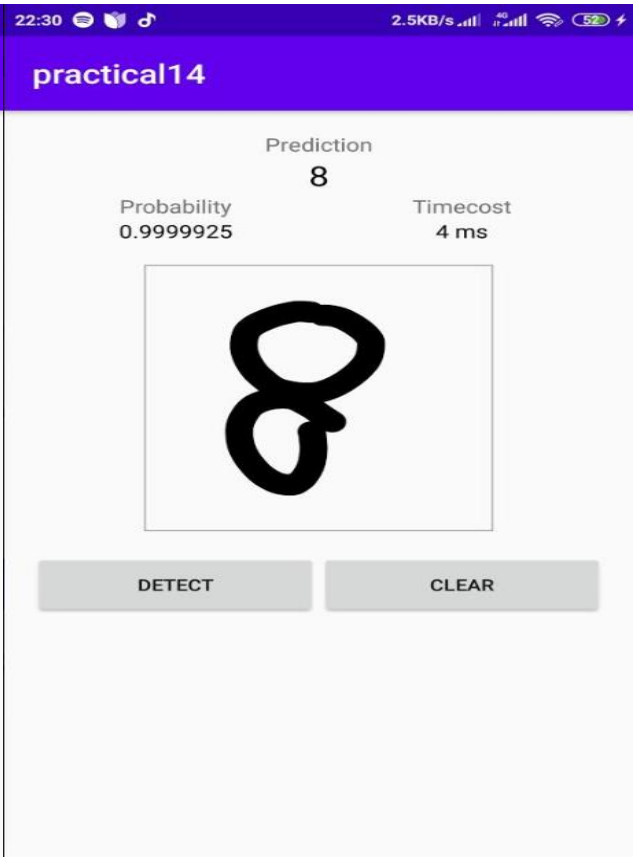
```
<?xml version="1.0" encoding="utf-8"?>
<manifest xmlns:android="http://schemas.android.com/apk/res/android"
    package="com.example.prac_13_wcmc_17it015">

    <application
        android:allowBackup="true"
        android:icon="@mipmap/ic_launcher"
        android:label="@string/app_name"
        android:roundIcon="@mipmap/ic_launcher_round"
        android:supportsRtl="true"
        android:theme="@style/AppTheme">
        <activity android:name=".MainActivity">
            <intent-filter>
                <action android:name="android.intent.action.MAIN" />

                <category android:name="android.intent.category.LAUNCHER" />
            </intent-filter>
        </activity>
    </application>

</manifest>
```

OUTPUT:


22:30
0.1KB/s
practical14
Prediction
1
Probability
0.9991159
Timecost
2 ms
DETECT
CLEAR


22:30
0.1KB/s
practical14
Prediction
6
Probability
0.9999583
Timecost
4 ms
DETECT
CLEAR


22:29
0.6KB/s
practical14
Prediction
2
Probability
0.99905413
Timecost
4 ms
DETECT
CLEAR


22:30
2.5KB/s
practical14
Prediction
8
Probability
0.9999925
Timecost
4 ms
DETECT
CLEAR

LATEST APPLICATIONS:

Netflix, Tinder, Snapchat.

LEARNING OUTCOME:

I learnt how to work with machine learning algorithms in android studio as well as I also learnt how to combine android studio and trainng an algorithm and implement in real application.

PROJECT

Github link :- https://github.com/Dalsania-Kishan/WCMC_Project