1. Creating Hello World Application

Steps:
1. Click New Project, the New Project Dialog box appears.
2. Choose Empty Views Activity then click Next.
3. Specify the Name of your project, Select the Language as Java, and Select the SDK as API 24("Nougat",Android 7.0).Click Finish Button.
4. Update the following code in activity_main.xml and MainActivity.java

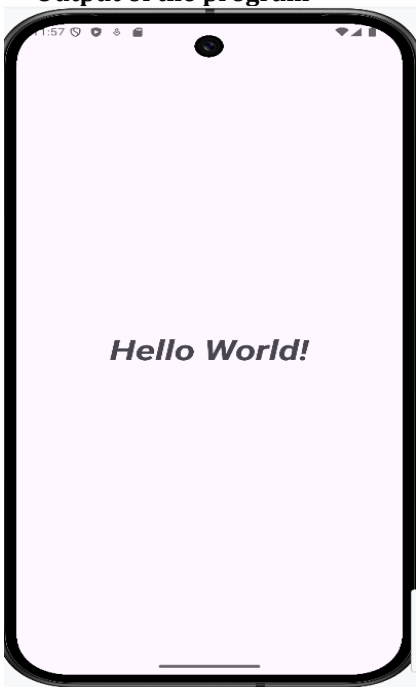
=>Coding part of Activity_main.xml

```
<?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/main"
    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="Hello World!"
        android:textSize="40sp"
        android:textStyle="bold|italic"
        app:layout_constraintBottom_toBottomOf="parent"
        app:layout_constraintEnd_toEndOf="parent"
        app:layout_constraintStart_toStartOf="parent"
        app:layout_constraintTop_toTopOf="parent" />

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

=>Coding part of MainActivity.java

```
package com.example.loginapplication;

import android.os.Bundle;
import androidx.activity.EdgeToEdge;
import androidx.appcompat.app.AppCompatActivity;
import androidx.core.graphics.Insets;
import androidx.core.view.ViewCompat;
import androidx.core.view.WindowInsetsCompat;

public class MainActivity extends AppCompatActivity {

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

=>Output of the program



2. Creating an Application that displays message based on screen orientation

Step 1: Click New Project, the New Project Dialog box appears.
2. Choose Empty Views Activity then click Next.
3. Specify the Name of your project, Select the Language as Java, and Select the SDK as API 24("Nougat",Android 7.0).Click Finish Button.
4. Update the following code in activity_main.xml and MainActivity.java

=>Coding part of 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/main"
    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="This is Portrait"
        android:textSize="40sp"
        app:layout_constraintBottom_toBottomOf="parent"
        app:layout_constraintEnd_toEndOf="parent"
        app:layout_constraintHorizontal_bias="0.496"
        app:layout_constraintStart_toStartOf="parent"
        app:layout_constraintTop_toTopOf="parent"
        app:layout_constraintVertical_bias="0.25" />

    <Button
        android:id="@+id/button"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:onClick="onClick"
        android:text="Launch new activity"
        app:layout_constraintBottom_toBottomOf="parent"
```

```
    app:layout_constraintEnd_toEndOf="parent"
    app:layout_constraintHorizontal_bias="0.497"
    app:layout_constraintStart_toStartOf="parent"
    app:layout_constraintTop_toTopOf="parent"
    app:layout_constraintVertical_bias="0.499" />

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

=>Coding part of MainActivity.java

```
package com.example.orientation;

import android.content.Intent;
import android.os.Bundle;
import android.view.View;
import androidx.activity.EdgeToEdge;
import androidx.appcompat.app.AppCompatActivity;
import androidx.core.graphics.Insets;
import androidx.core.view.ViewCompat;
import androidx.core.view.WindowInsetsCompat;

public class MainActivity extends AppCompatActivity {

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

  public void onClick(View v){
    Intent intent=new Intent(MainActivity.this,Nextactivity.class);
    startActivity(intent);
  }

}
```

Step 2: Create another new empty views activity and give the name as Nextactivity (Go to app>> New>>Activity>>Empty Views Activity)

=>Coding part of Activity_nextactivity.xml

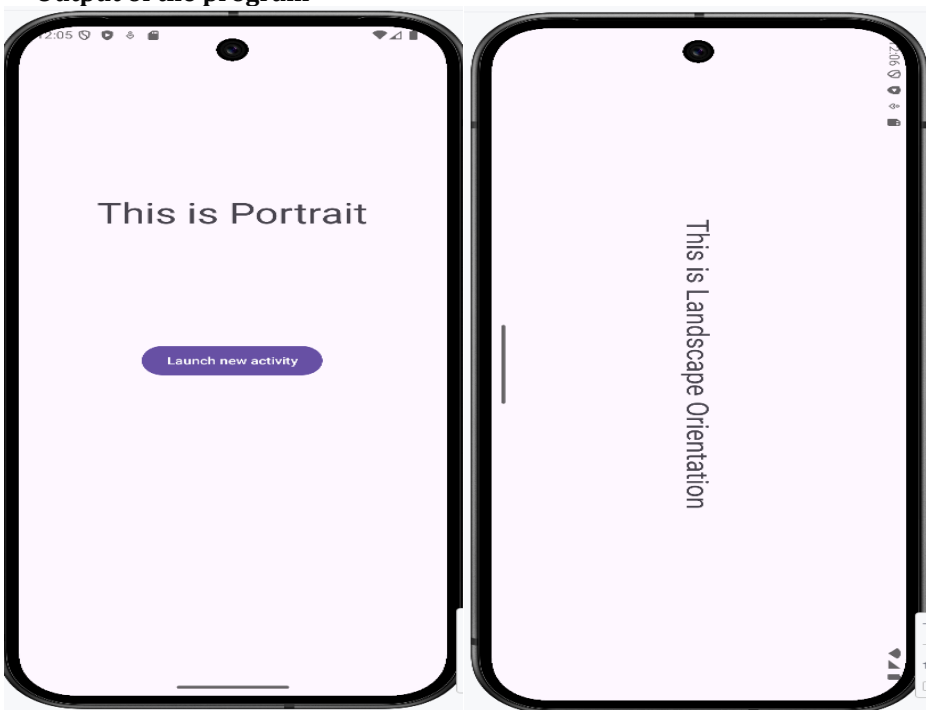
```
<?xml version="1.0" encoding="utf-8"?>
<androidx.constraintlayout.widget.ConstraintLayout
xmlns:android="http://schemas.android.com/apk/res/android"
  xmlns:app="http://schemas.android.com/apk/res-auto"
  xmlns:tools="http://schemas.android.com/tools"
  android:id="@+id/main"
  android:layout_width="match_parent"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

  <TextView
    android:id="@+id/textView3"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:text="This is Landscape Orientation"
    android:textSize="30sp"
    app:layout_constraintBottom_toBottomOf="parent"
    app:layout_constraintEnd_toEndOf="parent"
    app:layout_constraintStart_toStartOf="parent"
    app:layout_constraintTop_toTopOf="parent" />
</androidx.constraintlayout.widget.ConstraintLayout>
```

Step 3: Add the screen orientation values in AndroidManifest.xml

=>Coding part of AndroidManifest.xml

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

    <application
        android:allowBackup="true"
        android:dataExtractionRules="@xml/data_extraction_rules"
        android:fullBackupContent="@xml/backup_rules"
        android:icon="@mipmap/ic_launcher"
        android:label="@string/app_name"
        android:roundIcon="@mipmap/ic_launcher_round"
        android:supportsRtl="true"
        android:theme="@style/Theme.Orientation"
        tools:targetApi="31">
        <activity
            android:name=".Nextactivity"
            android:exported="false"
            android:screenOrientation="landscape"

            />
        <activity
            android:name=".MainActivity"
            android:exported="true"
            android:screenOrientation="portrait"
            >
            <intent-filter>
                <action android:name="android.intent.action.MAIN" />

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

=>Output of the program



3. Create and Application to develop Login window using UI controls

Step 1: Click New Project, the New Project Dialog box appears.
2. Choose Empty Views Activity then click Next.
3. Specify the Name of your project, Select the Language as Java, and Select the SDK as API 24("Nougat",Android 7.0).Click Finish Button.
4. Update the following code in activity_main.xml and MainActivity.java

=>Coding part of Activity_main.xml

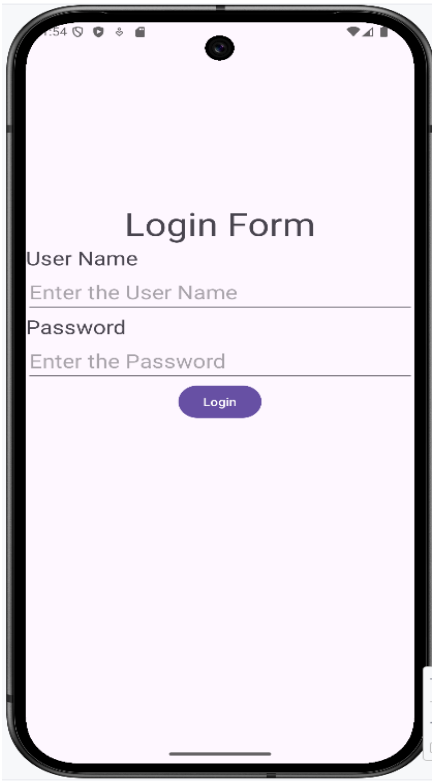
```
<?xml version="1.0" encoding="utf-8"?>
<LinearLayout xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:app="http://schemas.android.com/apk/res-auto"
    xmlns:tools="http://schemas.android.com/tools"
    android:id="@+id/main"
    android:layout_width="match_parent"
    android:layout_height="match_parent"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

    <TextView
        android:id="@+id/tvTitle"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_gravity="center"
        android:layout_marginTop="200dp"
        android:text="Login Form"
        android:textSize="40sp" />

    <TextView
        android:id="@+id/tvUserName"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="User Name"
        android:textSize="24sp" />

    <EditText
        android:id="@+id/etUserName"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:hint="Enter the User Name"
        android:inputType="text"
        android:textSize="24sp" />

    <TextView
        android:id="@+id/tvPassword"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="Password"
        android:textSize="24sp" />

    <EditText
        android:id="@+id/etPassword"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:hint="Enter the Password"
        android:inputType="textPassword"
        android:textSize="24sp" />

    <Button
        android:id="@+id/btnLogin"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="Login"
```

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

</LinearLayout>
```

=>Coding part of MainActivity.java

```
package com.example.loginapplication;

import android.os.Bundle;
import androidx.activity.EdgeToEdge;
import androidx.appcompat.app.AppCompatActivity;
import androidx.core.graphics.Insets;
import androidx.core.view.ViewCompat;
import androidx.core.view.WindowInsetsCompat;

public class MainActivity extends AppCompatActivity {

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

=>Output of the program



4. Create and Application to implement new activity using explicit intent and implicit intent

Step 1: Click New Project, the New Project Dialog box appears.
2. Choose Empty Views Activity then click Next.
3. Specify the Name of your project, Select the Language as Java, and Select the SDK as API 24("Nougat",Android 7.0).Click Finish Button.
4. Update the following code in activity_main.xml and MainActivity.java

=>Coding part of 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:id="@+id/main"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    android:orientation="vertical"
    tools:context=".MainActivity">

    <TextView
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="Explicit Activity"
        android:textSize="30sp"
        android:layout_gravity="center"
        android:layout_marginTop="200dp"
        />

    <Button
        android:id="@+id/btnExplicitContent"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:layout_marginTop="200dp"
        android:onClick="onClick"
        android:text="Explicit Activity" />

</LinearLayout>
```

=>Coding part of MainActivity.java

```
package com.example.implicitexplicit;
import android.content.Intent;
import android.view.View;
import android.os.Bundle;

import androidx.activity.EdgeToEdge;
import androidx.appcompat.app.AppCompatActivity;
import androidx.core.graphics.Insets;
import androidx.core.view.ViewCompat;
import androidx.core.view.WindowInsetsCompat;



public class MainActivity extends AppCompatActivity {

    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        EdgeToEdge.enable(this);
        setContentView(R.layout.activity_main);
        ViewCompat.setOnApplyWindowInsetsListener(findViewById(R.id.main), (v, insets) -> {
            Insets systemBars = insets.getInsets(WindowInsetsCompat.Type.systemBars());
            v.setPadding(systemBars.left, systemBars.top, systemBars.right, systemBars.bottom);
            return insets;
        });
    }
```

```
    public void onClick(View view) {
        Intent intent = new Intent(MainActivity.this, NextActivity.class);
        startActivity(intent);
    }

}
```

Step 2: Create another new empty views activity and give the name as Nextactivity (Go to app>> New>>Activity>>Empty Views Activity)

Coding part of activity_next.xml

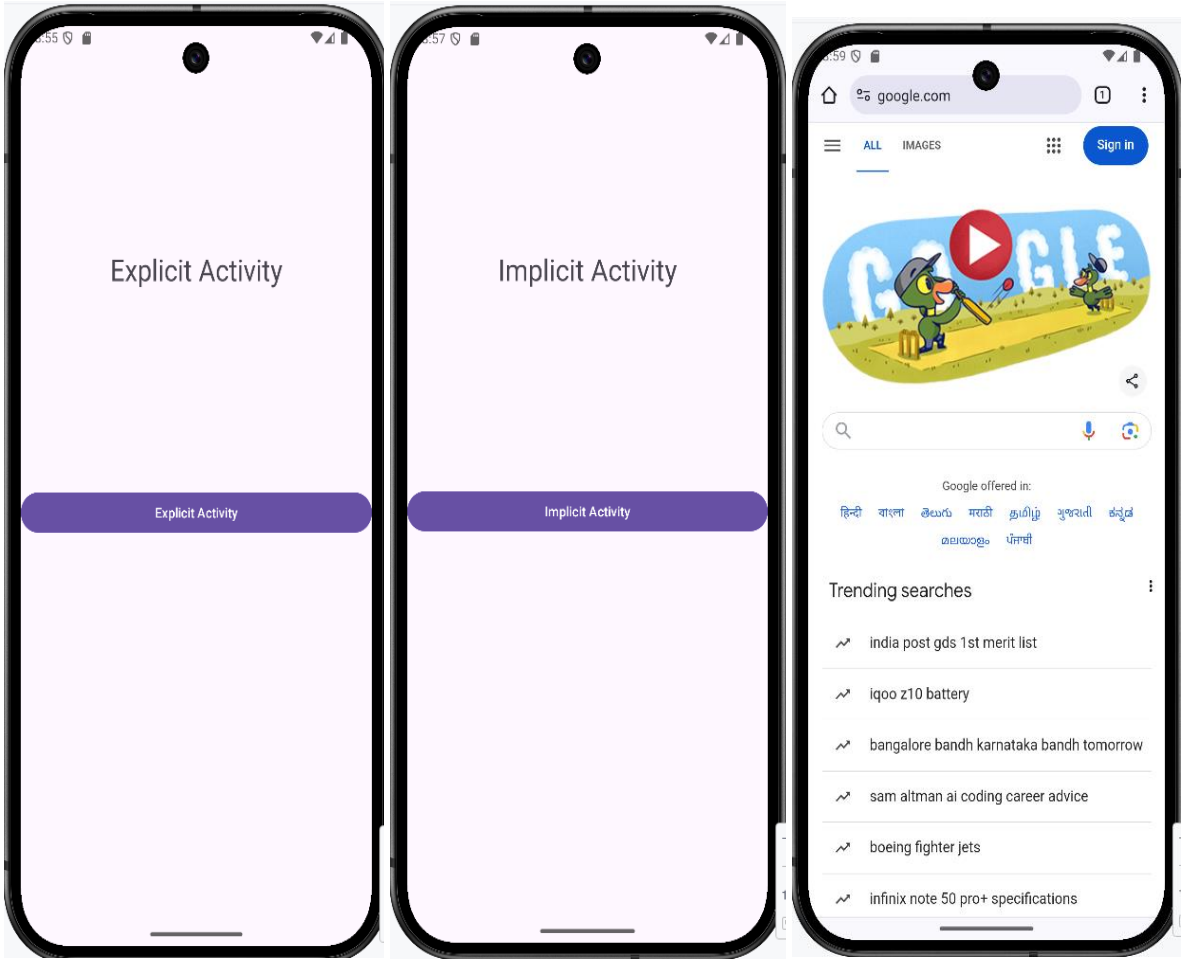
```
<?xml version="1.0" encoding="utf-8"?>
<LinearLayout xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:app="http://schemas.android.com/apk/res-auto"
    xmlns:tools="http://schemas.android.com/tools"
    android:id="@+id/main"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    android:orientation="vertical"
    tools:context=".NextActivity">

    <TextView
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="Implicit Activity"
        android:textSize="30sp"
        android:layout_gravity="center"
        android:layout_marginTop="200dp"
        />

    <Button
        android:id="@+id/btnImplicitContent"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:layout_marginTop="200dp"
        android:onClick="onClick"
        android:text="Implicit Activity" />

</LinearLayout>
```

=>Coding part of NextActivity.java

```
package com.example.implicitexplicit;

import android.os.Bundle;
import androidx.activity.EdgeToEdge;
import androidx.appcompat.app.AppCompatActivity;
import androidx.core.graphics.Insets;
import androidx.core.view.ViewCompat;
import androidx.core.view.WindowInsetsCompat;
import android.content.Intent;
import android.net.Uri;
import android.view.View;


public class NextActivity extends AppCompatActivity {

    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        EdgeToEdge.enable(this);
        setContentView(R.layout.activity_next);
```

```
        ViewCompat.setOnApplyWindowInsetsListener(findViewById(R.id.main), (v, insets) -> {
            Insets systemBars = insets.getInsets(WindowInsetsCompat.Type.systemBars());
            v.setPadding(systemBars.left, systemBars.top, systemBars.right, systemBars.bottom);
            return insets;
        });

    }


    public void onClick(View view) {
        Uri webpage= Uri.parse("http://www.google.com");
        Intent intent = new Intent(Intent.ACTION_VIEW, webpage);
        startActivity(intent);
    }


}
```

=>Output of the program



// 5 Create an application that displays custom designed Opening Screen

Project creation Steps:
1. Click New Project, the New Project Dialog box appears.
2. Choose Empty Views Activity then click Next.
3. Specify the Name of your project, Select the Language as Java, and Select the SDK as API 24("Nougat",Android 7.0).Click Finish Button.

Next follow the below steps to create background files

Step 1.Create background resources (bg_inner.xml)

a. To create resource file, click app>> res >> drawable >> new >>Drawable Resource File

b. Set filename as bg_inner.xml, root element as shape and then click ok.

Modify the bg_inner.xml file.

Step 2.Add image to the project

a. Copy the image file (Ctrl +C) from the file location

b. Click app>>res>> drawable>> right click Paste(Ctrl+V)

Coding part of bg_inner.xml

```
<?xml version="1.0" encoding="utf-8"?>
<shape xmlns:android="http://schemas.android.com/apk/res/android">
  <gradient
    android:startColor="#84FFFF"
    android:endColor="#f08"
    android:angle="100"/>
  <corners android:radius="20dp"/>
</shape>
```

Step 3. Change the activity_main.xml file and MainActivity.java

=>Coding part of 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:id="@+id/main"
  android:layout_width="match_parent"
  android:layout_height="match_parent"
  tools:context=".MainActivity"
  android:background="@drawable/bg_inner"
  >
  <TextView
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:text="Welcome !!!"
    android:textSize="40sp"
    android:layout_centerInParent="true"
    />
</RelativeLayout>
```

=>Coding part of MainActivity.java

```
package com.example.customdesign;

import android.os.Bundle;

import androidx.activity.EdgeToEdge;
import androidx.appcompat.app.AppCompatActivity;
import androidx.core.graphics.Insets;
import androidx.core.view.ViewCompat;
```

```
import androidx.core.view.WindowInsetsCompat;
import android.content.Intent;
import android.os.Handler;



public class MainActivity extends AppCompatActivity {
    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        EdgeToEdge.enable(this);
        setContentView(R.layout.activity_main);
        ViewCompat.setOnApplyWindowInsetsListener(findViewById(R.id.main), (v, insets) -> {
            Insets systemBars = insets.getInsets(WindowInsetsCompat.Type.systemBars());
            v.setPadding(systemBars.left, systemBars.top, systemBars.right, systemBars.bottom);
            return insets;
        });


        Handler handler=new Handler();
        handler.postDelayed(new Runnable() {
            @Override
            public void run() {
            Intent intent=new Intent(MainActivity.this,NextActivity.class);
            startActivity(intent);
            finish();
            }
        },3000);

    }
}
```

Step 4. Create one more empty views activity and give the name as NextActivity(Go to app>> New>>Activity>>Empty Views Activity) and change the activity_next.xml file and *no changes* in the NextActivity.java file

Coding part of activity_next.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:app2="http://schemas.android.com/apk/res-auto"
    xmlns:tools="http://schemas.android.com/tools"
    android:id="@+id/main"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    tools:context=".NextActivity"
    android:gravity="center"
    android:background="@drawable/android"
    >
    <TextView
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="Welcome to home Page"
        android:textColor="@color/black"
        android:textSize="32sp"
        android:textStyle="bold"
        android:layout_centerInParent="true"
        />
</RelativeLayout>
```

=>Coding part of NextActivity.java

```
package com.example.customdesign;

import android.os.Bundle;
import androidx.activity.EdgeToEdge;
import androidx.appcompat.app.AppCompatActivity;
import androidx.core.graphics.Insets;
import androidx.core.view.ViewCompat;
import androidx.core.view.WindowInsetsCompat;

public class NextActivity extends AppCompatActivity {

    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        EdgeToEdge.enable(this);
        setContentView(R.layout.activity_next);
        ViewCompat.setOnApplyWindowInsetsListener(findViewById(R.id.main), (v, insets) -> {
            Insets systemBars = insets.getInsets(WindowInsetsCompat.Type.systemBars());
            v.setPadding(systemBars.left, systemBars.top, systemBars.right, systemBars.bottom);
            return insets;
        });
    }
}
```

=>Output of the program

6. Create and Application to show all views

Steps:
1. Click New Project, the New Project Dialog box appears.
2. Choose Empty Views Activity then click Next.
3. Specify the Name of your project, Select the Language as Java, and Select the SDK as API 24("Nougat",Android 7.0).Click Finish Button.
4. Update the following code in activity_main.xml and MainActivity.java

=>Coding part of Activity_main.xml

```
<?xml version="1.0" encoding="utf-8"?>
<ScrollView 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/main"
android:layout_width="match_parent"
android:layout_height="match_parent"
tools:context=".MainActivity">

<LinearLayout
   android:layout_width="match_parent"
   android:layout_height="wrap_content"
   android:orientation="vertical"
   android:padding="16dp">

<TextView
   android:id="@+id/textView"
   android:layout_width="wrap_content"
   android:layout_height="wrap_content"
   android:text="This is a TextView"
   android:textSize="18sp"
   android:layout_marginBottom="16dp"/>

<EditText
   android:id="@+id/editText"
   android:layout_width="match_parent"
   android:layout_height="wrap_content"
   android:hint="Enter your text here"
   android:layout_marginBottom="16dp" />

<Button
   android:id="@+id/button"
   android:layout_width="wrap_content"
   android:layout_height="wrap_content"
   android:text="Click Me"
   android:layout_marginBottom="16dp" />

<CheckBox
   android:id="@+id/checkbox"
   android:layout_width="wrap_content"
   android:layout_height="wrap_content"
   android:text="Check Me"
   android:layout_marginBottom="16dp" />
 <TextView
      android:id="@+id/textView1"
      android:layout_width="wrap_content"
      android:layout_height="wrap_content"
      android:text="Which Section You belong to"
      android:textSize="20sp"
      android:layout_marginBottom="16dp"/>
<RadioGroup
   android:id="@+id/radioGroup"
   android:layout_width="wrap_content"
   android:layout_height="wrap_content"
   android:orientation="vertical"
   android:layout_marginBottom="16dp" />

<RadioButton
   android:id="@+id/radioButton1"
   android:layout_width="wrap_content"
   android:layout_height="wrap_content"
   android:text="A section" />

<RadioButton
   android:id="@+id/radioButton2"
   android:layout_width="wrap_content"
   android:layout_height="wrap_content"
```

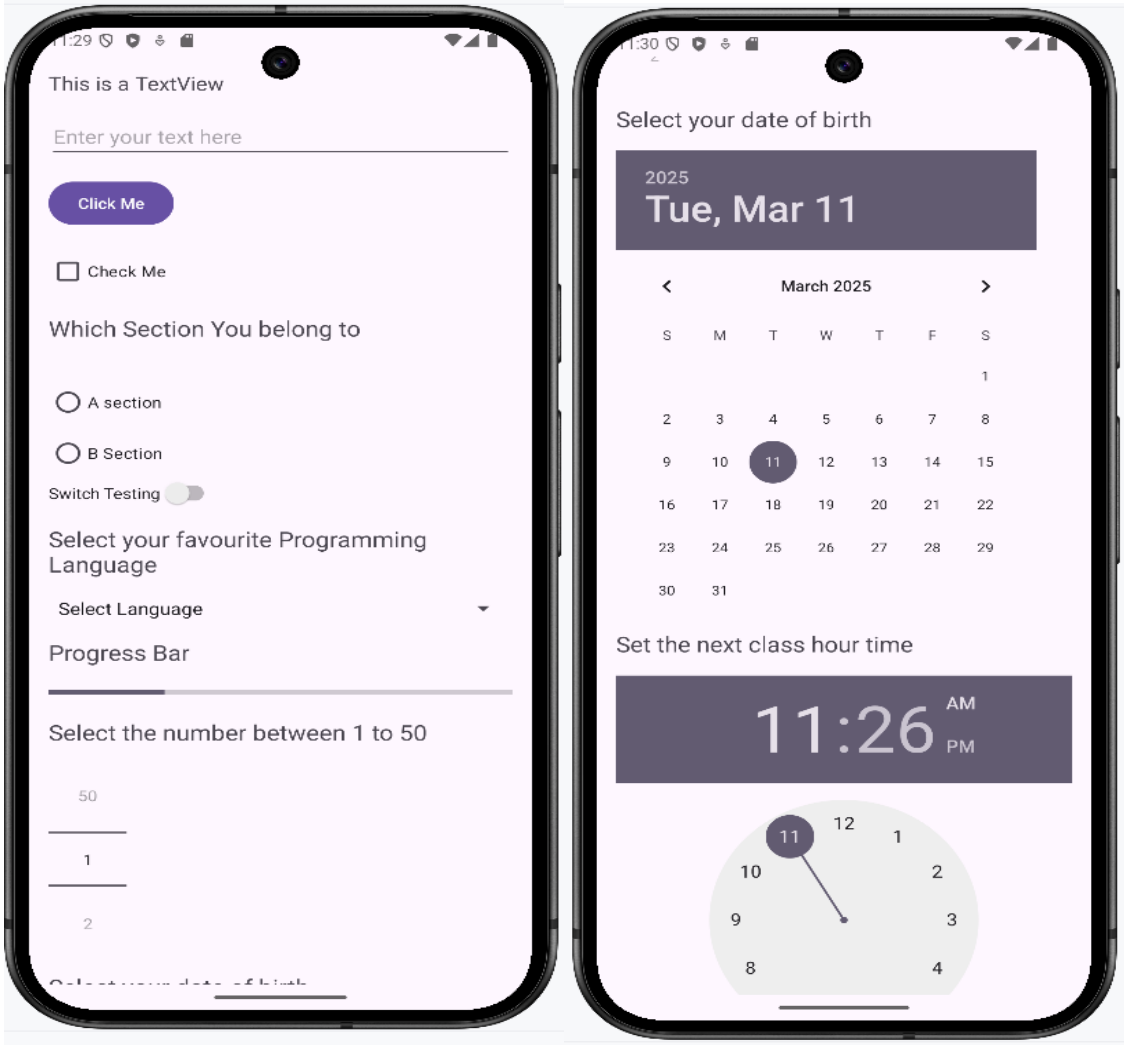
```
    android:text="B Section" />

<Switch
    android:id="@+id/switch1"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:text="Switch Testing"
    android:layout_marginBottom="16dp"/>
<TextView
        android:id="@+id/textView2"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="Select your favourite Programming Language"
        android:textSize="20sp"
        android:layout_marginBottom="16dp"/>
<Spinner
    android:id="@+id/spinner"
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:layout_marginBottom="16dp"/>
<TextView
        android:id="@+id/textView3"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="Progress Bar"
        android:textSize="20sp"
        android:layout_marginBottom="16dp"/>
<ProgressBar
        android:id="@+id/progressBar"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        style="@style/Widget.AppCompat.ProgressBar.Horizontal"
        android:progress="25"
        android:layout_marginBottom="16dp" />
<TextView
        android:id="@+id/textView4"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="Select the number between 1 to 50"
        android:textSize="20sp"
        android:layout_marginBottom="16dp"/>
 <NumberPicker
        android:id="@+id/numberPicker1"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_marginBottom="16dp"/>
 <TextView
        android:id="@+id/textView5"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="Select your date of birth"
        android:textSize="20sp"
        android:layout_marginBottom="16dp"/>
 <DatePicker
    android:id="@+id/datePicker"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:layout_marginBottom="16dp"/>
<TextView
        android:id="@+id/textView6"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="Set the next class hour time"
        android:textSize="20sp"
        android:layout_marginBottom="16dp"/>
```

```
    <TimePicker
        android:id="@+id/timePicker"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_marginBottom="16dp"/>

    </LinearLayout>

</ScrollView>
```

=>Coding part of MainActivity.java

```
package com.example.mainactivity;

import android.os.Bundle;
import android.widget.ArrayAdapter;
import android.widget.Spinner;

import androidx.activity.EdgeToEdge;
import androidx.appcompat.app.AppCompatActivity;
import androidx.core.graphics.Insets;
import androidx.core.view.ViewCompat;
import androidx.core.view.WindowInsetsCompat;

import java.util.ArrayList;
import android.widget.NumberPicker;

public class MainActivity extends AppCompatActivity {

    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        EdgeToEdge.enable(this);
        setContentView(R.layout.activity_main);
        ViewCompat.setOnApplyWindowInsetsListener(findViewById(R.id.main), (v, insets) -> {
            Insets systemBars = insets.getInsets(WindowInsetsCompat.Type.systemBars());
            v.setPadding(systemBars.left, systemBars.top, systemBars.right, systemBars.bottom);
            return insets;
        });
        Spinner spinner = findViewById(R.id.spinner);
        ArrayList<String> arrayList = new ArrayList<>();
        arrayList.add("Select Language");
        arrayList.add("JAVA");
        arrayList.add("ANDROID");
        arrayList.add("C Language");
        arrayList.add("CPP Language");
        arrayList.add("Python Programming");

        ArrayAdapter<String> arrayAdapter = new ArrayAdapter<String>(this,
                android.R.layout.simple_spinner_item, arrayList);
         arrayAdapter.setDropDownViewResource(android.R.layout.simple_spinner_dropdown_item);
        spinner.setAdapter(arrayAdapter);


        NumberPicker numberPicker1 = findViewById(R.id.numberPicker1);
        numberPicker1.setMinValue(1);
        numberPicker1.setMaxValue(50);

    }
}
```

=> Output of the program



7 Create a menu in application

Steps:
1. Click New Project, the New Project Dialog box appears.
2. Choose Empty Views Activity then click Next.
3. Specify the Name of your project, Select the Language as Java, and Select the SDK as API 24("Nougat",Android 7.0).Click Finish Button.

<u>Steps to be followed to create the menu</u>

Step1. Create Android resource directory by clicking res>>new>>Android Resource directory, give the name as menu

Step2. Right click menu folder click new>> Menu Resource File, give the name of the file as menus.

Add the following code in menus.xml

```
<?xml version="1.0" encoding="utf-8"?>
<menu xmlns:android="http://schemas.android.com/apk/res/android">
   <item android:id="@+id/php"  android:title="PHP"/>
   <item android:id="@+id/java"  android:title="JAVA"/>
   <item android:id="@+id/csharp"  android:title="C#"/>
</menu>
```

Step 3. =>Coding part of Activity_main.xml (Remove Helloworld textview control)

```
<?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/main"
   android:layout_width="match_parent"
   android:layout_height="match_parent"
   tools:context=".MainActivity">
</androidx.constraintlayout.widget.ConstraintLayout>
```

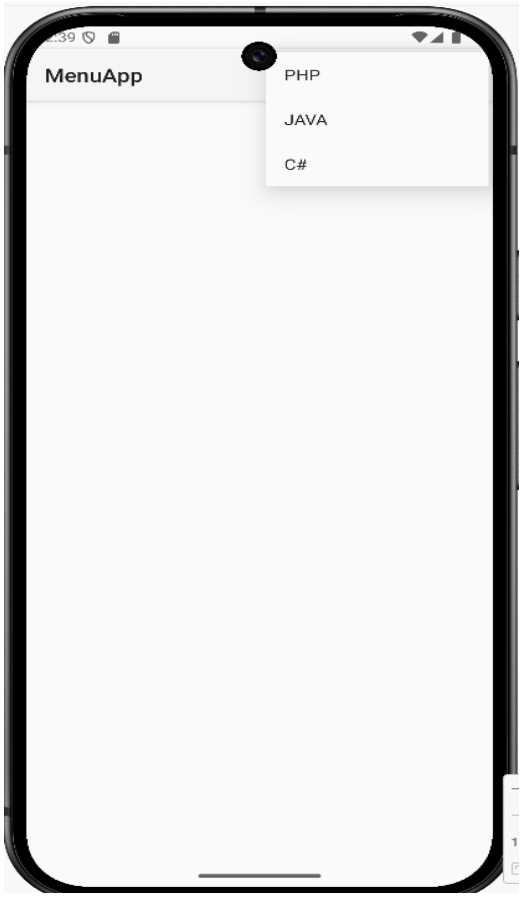
Step 4. =>Coding part of MainActivity.java

```
package com.example.menuapp;

import android.os.Bundle;
import androidx.activity.EdgeToEdge;
import androidx.appcompat.app.AppCompatActivity;
import androidx.core.graphics.Insets;
import androidx.core.view.ViewCompat;
import androidx.core.view.WindowInsetsCompat;
import android.view.Menu;
import android.view.MenuInflater;
import android.view.MenuItem;
import android.widget.Toast;


public class MainActivity extends AppCompatActivity {

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

   @Override
   public boolean onCreatePanelMenu(int featureId,Menu menu)
   {
      MenuInflater inflater=getMenuInflater();
      inflater.inflate(R.menu.menus,menu);
      return true;
   }
   @Override
   public boolean onOptionsItemSelected(MenuItem item)
   {
      if(item.getItemId()==R.id.php) {
      Toast.makeText(this, "Php Page", Toast.LENGTH_SHORT).show();
      }
      if(item.getItemId()==R.id.java) {
         Toast.makeText(this, "Java Page", Toast.LENGTH_SHORT).show();
      }
      if(item.getItemId()==R.id.csharp) {
         Toast.makeText(this, "C# Page", Toast.LENGTH_SHORT).show();
      }
      return true;
   }
}
```

Step 5. Change the theme value AndroidManifest.xml file

```
<?xml version="1.0" encoding="utf-8"?>
<manifest xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:tools="http://schemas.android.com/tools">
    <application
        android:allowBackup="true"
        android:dataExtractionRules="@xml/data_extraction_rules"
        android:fullBackupContent="@xml/backup_rules"
        android:icon="@mipmap/ic_launcher"
        android:label="@string/app_name"
        android:roundIcon="@mipmap/ic_launcher_round"
        android:supportsRtl="true"
        android:theme="@style/Theme.AppCompat.Light"
        tools:targetApi="31">
        <activity
            android:name=".MainActivity"
            android:exported="true">
            <intent-filter>
                <action android:name="android.intent.action.MAIN" />

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

=>Output of the program



8. Create an application to read/ write the Local data.

Steps:
1. Click New Project, the New Project Dialog box appears.
2. Choose Empty Views Activity then click Next.
3. Specify the Name of your project, Select the Language as Java, and Select the SDK as API 24("Nougat",Android 7.0).Click Finish Button.
4. Update the following code in activity_main.xml and MainActivity.java

=>Coding part of 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:id="@+id/main"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    android:orientation="vertical"
    tools:context=".MainActivity">

    <TextView
        android:id="@+id/tvTitle"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_gravity="center"
        android:layout_marginTop="200dp"
        android:text="Login Form"
        android:textSize="40sp" />

    <TextView
        android:id="@+id/tvUserName"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="User Name"
        android:textSize="24sp" />

    <EditText
        android:id="@+id/etUserName"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:hint="Enter the User Name"
        android:inputType="text"
        android:textSize="24sp" />

    <TextView
        android:id="@+id/tvPassword"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="Password"
        android:textSize="24sp" />

    <EditText
        android:id="@+id/etPassword"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:hint="Enter the Password"
        android:inputType="textPassword"
        android:textSize="24sp" />

    <Button
        android:id="@+id/btnSave"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
```

```
    android:layout_gravity="center"
    android:onClick="onSaveClick"
    android:text="Save" />
  <Button
    android:id="@+id/btnNext"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:layout_gravity="center"
    android:onClick="onNextClick"
    android:text="Next" />

</LinearLayout>
```

=>Coding part of MainActivity.java

```
package com.example.savelocaldata;
import androidx.activity.EdgeToEdge;
import androidx.appcompat.app.AppCompatActivity;
import androidx.core.graphics.Insets;
import androidx.core.view.ViewCompat;
import androidx.core.view.WindowInsetsCompat;

import android.content.Context;
import android.content.Intent;
import android.content.SharedPreferences;
import android.os.Bundle;
import android.view.View;
import android.widget.EditText;
import android.widget.Toast;

public class MainActivity extends AppCompatActivity {
  EditText etUserName,etPassword;

  @Override
  protected void onCreate(Bundle savedInstanceState) {

    super.onCreate(savedInstanceState);
    EdgeToEdge.enable(this);
    setContentView(R.layout.activity_main);

    etUserName = findViewById(R.id.etUserName);
    etPassword = findViewById(R.id.etPassword);
  }

  public void onSaveClick(View view) {

  SharedPreferences sharedPreferences = getSharedPreferences("MyPrefs",
Context.MODE_PRIVATE);
  SharedPreferences.Editor editor = sharedPreferences.edit();
  editor.putString("username",etUserName.getText().toString());
  editor.putString("password", etPassword.getText().toString());
  editor.apply();
  Toast.makeText(getApplicationContext(),"Saved successfully",Toast.LENGTH_LONG).show();
 }

  public void onNextClick(View view) {
    Intent intent = new  Intent(this, NextActivity.class);
    startActivity(intent);
  }
}
```

Next Step. Create one more empty views activity and give the name as NextActivity (Go to app>> New>>Activity>>Empty Views Activity)

Coding part of activity_next.xml

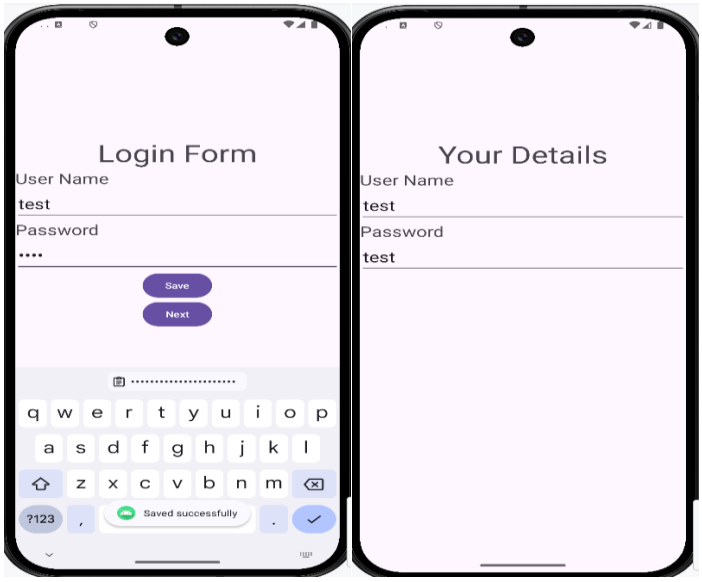
```
<?xml version="1.0" encoding="utf-8"?>
<LinearLayout xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:app="http://schemas.android.com/apk/res-auto"
    xmlns:tools="http://schemas.android.com/tools"
    android:id="@+id/main"
    android:layout_width="match_parent"
    android:layout_height="match_parent"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

    <TextView
        android:id="@+id/tvTitle"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_gravity="center"
        android:layout_marginTop="200dp"
        android:text="Your Details"
        android:textSize="40sp" />

    <TextView
        android:id="@+id/tvUserName"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="User Name"
        android:textSize="24sp" />

    <EditText
        android:id="@+id/etUserName"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:hint="Enter the User Name"
        android:inputType="text"
        android:textSize="24sp" />

    <TextView
        android:id="@+id/tvPassword"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="Password"
        android:textSize="24sp" />

    <EditText
        android:id="@+id/etPassword"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:hint="Enter the Password"
        android:inputType="text"
        android:textSize="24sp" />

</LinearLayout>
```

=>Coding part of NextActivity.java

```
package com.example.savelocaldata;

import android.os.Bundle;
import androidx.activity.EdgeToEdge;
import androidx.appcompat.app.AppCompatActivity;
import androidx.core.graphics.Insets;
import androidx.core.view.ViewCompat;
import androidx.core.view.WindowInsetsCompat;
import android.content.Context;
```

```
import android.content.SharedPreferences;
import android.widget.EditText;

public class NextActivity extends AppCompatActivity {
   EditText etUserName,etPassword;
   @Override
   protected void onCreate(Bundle savedInstanceState) {
      super.onCreate(savedInstanceState);
      EdgeToEdge.enable(this);
      setContentView(R.layout.activity_next);

etUserName = findViewById(R.id.etUserName);
etPassword = findViewById(R.id.etPassword);

SharedPreferences sharedPreferences = getSharedPreferences("MyPrefs",
Context.MODE_PRIVATE);
String username = sharedPreferences.getString("username", "");
String password = sharedPreferences.getString("password", "");
etUserName.setText(username);
etPassword.setText(password);

   }
}
```

=>Output of the program



//9 Create / Read / Write data with database (SQLite)

Steps:
1. Click New Project, the New Project Dialog box appears.
2. Choose Empty Views Activity then click Next.
3. Specify the Name of your project, Select the Language as Java, and Select the SDK as API 24("Nougat",Android 7.0).Click Finish Button.
4. Update the following code in activity_main.xml and MainActivity.java

=>Coding part of Activity_main.xml

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

```
    android:layout_height="wrap_content"
    android:layout_x="50dp"
    android:layout_y="20dp"
    android:text="Student Details"
    android:textSize="30sp" />

<TextView
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:layout_x="20dp"
    android:layout_y="110dp"
    android:text="Enter Rollno:"
    android:textSize="20sp" />

<EditText
    android:id="@+id/Rollno"
    android:layout_width="150dp"
    android:layout_height="wrap_content"
    android:layout_x="175dp"
    android:layout_y="100dp"
    android:inputType="number"
    android:textSize="20sp" />

<TextView
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:layout_x="20dp"
    android:layout_y="160dp"
    android:text="Enter Name:"
    android:textSize="20sp" />

<EditText
    android:id="@+id/Name"
    android:layout_width="150dp"
    android:layout_height="wrap_content"
    android:layout_x="175dp"
    android:layout_y="150dp"
    android:inputType="text"
    android:textSize="20sp" />

<TextView
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:layout_x="20dp"
    android:layout_y="210dp"
    android:text="Enter Marks:"
    android:textSize="20sp" />

<EditText
    android:id="@+id/Marks"
    android:layout_width="150dp"
    android:layout_height="wrap_content"
    android:layout_x="175dp"
    android:layout_y="200dp"
    android:inputType="number"
    android:textSize="20sp" />

<Button
    android:id="@+id/Insert"
    android:layout_width="150dp"
    android:layout_height="wrap_content"
    android:layout_x="25dp"
    android:layout_y="300dp"
    android:onClick="onClickInsert"
    android:text="Insert"
```

```
    android:textSize="30dp" />

  <Button
    android:id="@+id/Delete"
    android:layout_width="150dp"
    android:layout_height="wrap_content"
    android:layout_x="200dp"
    android:layout_y="300dp"
    android:onClick="onClickDelete"
    android:text="Delete"
    android:textSize="30dp" />

  <Button
    android:id="@+id/Update"
    android:layout_width="150dp"
    android:layout_height="wrap_content"
    android:layout_x="25dp"
    android:layout_y="400dp"
    android:onClick="onClickUpdate"
    android:text="Update"
    android:textSize="30dp" />

  <Button
    android:id="@+id/View"
    android:layout_width="150dp"
    android:layout_height="wrap_content"
    android:layout_x="200dp"
    android:layout_y="400dp"
    android:onClick="onClickView"
    android:text="View"
    android:textSize="30dp" />

  <Button
    android:id="@+id/ViewAll"
    android:layout_width="200dp"
    android:layout_height="wrap_content"
    android:layout_x="100dp"
    android:layout_y="500dp"
    android:onClick="onClickViewAll"
    android:text="View All"
    android:textSize="30dp" />

</AbsoluteLayout>
```

=>Coding part of MainActivity.java

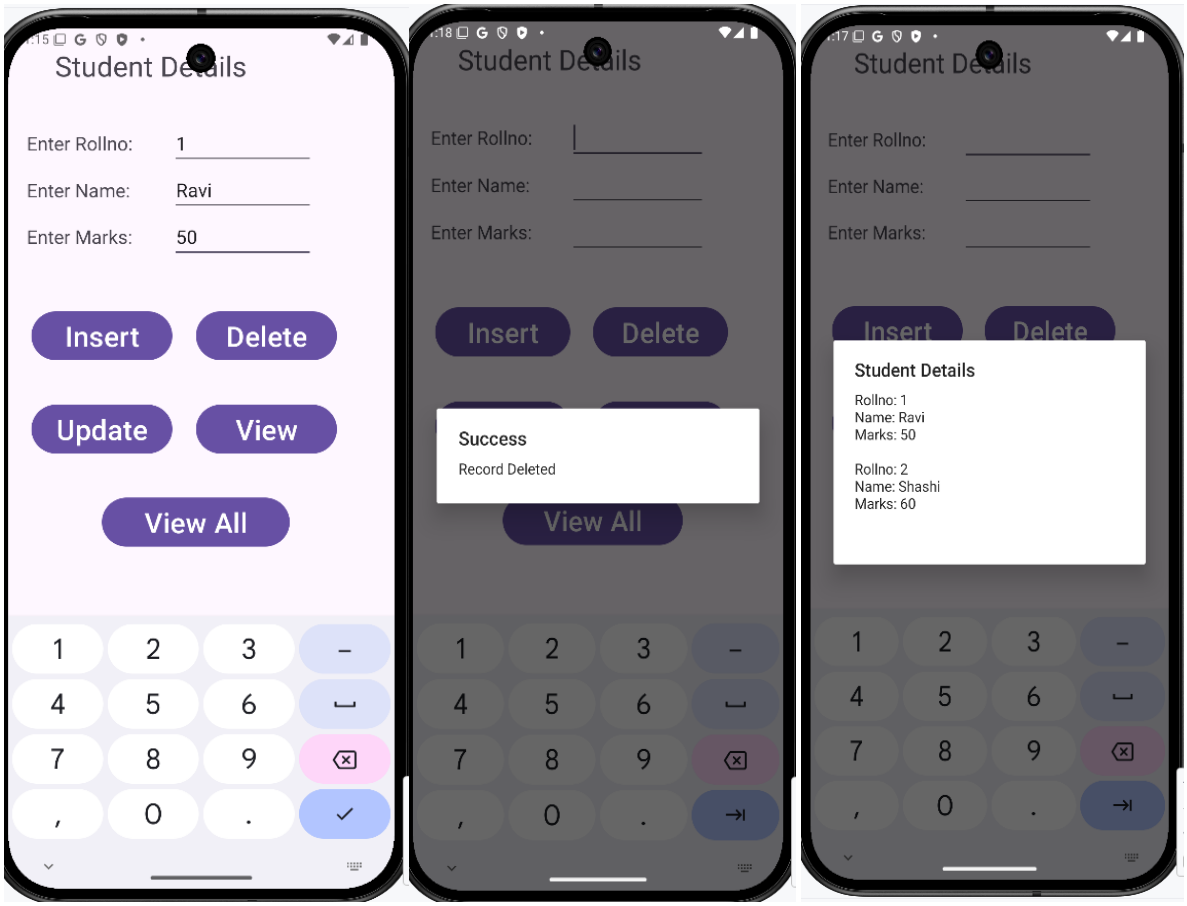
```
package com.example.databaseapp;

import android.app.AlertDialog;
import android.content.Context;
import android.database.Cursor;
import android.database.sqlite.SQLiteDatabase;
import android.os.Bundle;
import android.widget.Button;
import android.widget.EditText;

import androidx.activity.EdgeToEdge;
import androidx.appcompat.app.AppCompatActivity;
import androidx.core.graphics.Insets;
import androidx.core.view.ViewCompat;
import androidx.core.view.WindowInsetsCompat;
import android.view.View;
```

```
public class MainActivity extends AppCompatActivity {

EditText Rollno,Name,Marks;
 SQLiteDatabase db;

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

      Rollno=findViewById(R.id.Rollno);
      Name=findViewById(R.id.Name);
      Marks=findViewById(R.id.Marks);

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

   }
     public void onClickInsert(View view) {

        if (Rollno.getText().toString().trim().length() == 0 ||
            Name.getText().toString().trim().length() == 0 ||
            Marks.getText().toString().trim().length() == 0) {
          showMessage("Error", "Please enter all values");
          return;
        }
        db.execSQL("INSERT INTO student VALUES('" + Rollno.getText() + "','" + Name.getText() +
            "','" + Marks.getText() + "');");

        showMessage("Success", "Record added");
        clearText();

     }

     public void onClickDelete(View view) {

        if(Rollno.getText().toString().trim().length()==0)
        {
          showMessage("Error", "Please enter Rollno");
          return;
        }
        Cursor c=db.rawQuery("SELECT * FROM student WHERE rollno='"+Rollno.getText()+"'",
null);
        if(c.moveToFirst())
        {
          db.execSQL("DELETE FROM student WHERE rollno='"+Rollno.getText()+"'");
          showMessage("Success", "Record Deleted");
        }
        else
        {
          showMessage("Error", "Invalid Rollno");
        }
        clearText();
     }

     public void onClickUpdate(View view){

        if(Rollno.getText().toString().trim().length()==0)
        {
          showMessage("Error", "Please enter Rollno");
          return;
        }
        Cursor c=db.rawQuery("SELECT * FROM student WHERE rollno='"+Rollno.getText()+"'",
```

```
null);
        if(c.moveToFirst()) {
           db.execSQL("UPDATE student SET name='" + Name.getText() + "',marks='" +
Marks.getText() +
                 "' WHERE rollno='"+Rollno.getText()+"'");
           showMessage("Success", "Record Modified");
        }
        else {
           showMessage("Error", "Invalid Rollno");
        }
        clearText();
     }

     public void onClickView(View view) {

        if(Rollno.getText().toString().trim().length()==0)
        {
           showMessage("Error", "Please enter Rollno");
           return;
        }
        Cursor c=db.rawQuery("SELECT * FROM student WHERE rollno='"+Rollno.getText()+"'",
null);
        if(c.moveToFirst())
        {
           Name.setText(c.getString(1));
           Marks.setText(c.getString(2));
        }
        else
        {
           showMessage("Error", "Invalid Rollno");
           clearText();
        }
     }

     public void onClickViewAll(View view) {
        Cursor c=db.rawQuery("SELECT * FROM student", null);
        if(c.getCount()==0)
        {
           showMessage("Error", "No records found");
           return;
        }
        StringBuffer buffer=new StringBuffer();
        while(c.moveToNext())
        {
           buffer.append("Rollno: "+c.getString(0)+"\n");
           buffer.append("Name: "+c.getString(1)+"\n");
           buffer.append("Marks: "+c.getString(2)+"\n\n");
        }
        showMessage("Student Details", buffer.toString());
     }
   public void showMessage(String title,String message)
   {
     AlertDialog.Builder builder=new AlertDialog.Builder(this);
     builder.setCancelable(true);
     builder.setTitle(title);
     builder.setMessage(message);
     builder.show();
   }
   public void clearText()
   {
     Rollno.setText("");
     Name.setText("");
     Marks.setText("");
     Rollno.requestFocus();
```

```
    }
}
```

=>Output of the program



10. Create an application to send SMS

Steps:
1. Click New Project, the New Project Dialog box appears.
2. Choose Empty Views Activity then click Next.
3. Specify the Name of your project, Select the Language as Java, and Select the SDK as API 24("Nougat",Android 7.0).Click Finish Button.
4. Update the following code in activity_main.xml and MainActivity.java

=>Coding part of Activity_main.xml

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

   <LinearLayout
      android:layout_width="match_parent"
      android:layout_height="wrap_content"
      android:layout_marginStart="16dp"
      android:layout_marginTop="32dp"
      android:layout_marginEnd="16dp"
      android:orientation="horizontal">
```

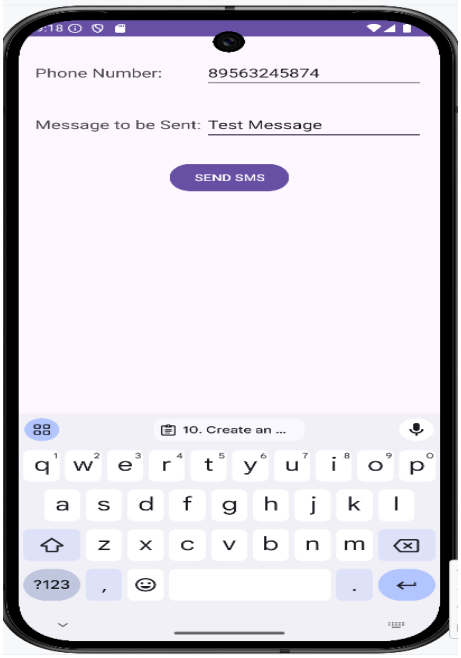
```
    <TextView
        android:id="@+id/textView"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="Phone Number:"
        android:width="165sp"
        android:textSize="18sp" />

    <EditText
        android:id="@+id/etPhoneNumber"
        android:layout_width="0dp"
        android:layout_height="wrap_content"
        android:layout_weight="1"
        android:ems="10"
        android:hint="Enter number"
        android:inputType="phone" />
</LinearLayout>

<LinearLayout
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:layout_marginStart="16dp"
    android:layout_marginTop="32dp"
    android:layout_marginEnd="16dp"
    android:orientation="horizontal">

    <TextView
        android:id="@+id/textView2"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="Message to be Sent:"
        android:width="165sp"
        android:textSize="18sp" />

    <EditText
        android:id="@+id/etMessage"
        android:layout_width="0dp"
        android:layout_height="wrap_content"
        android:layout_weight="1"
        android:hint="Enter message"
        android:inputType="textMultiLine" />

</LinearLayout>

<Button
    android:id="@+id/btnSendSMS"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:layout_gravity="center_horizontal"
    android:layout_marginTop="32dp"
    android:onClick="onClick"
    android:text="SEND SMS" />

</LinearLayout>
```

=>Coding part of MainActivity.java

```
package com.example.sendsms;
import androidx.appcompat.app.AppCompatActivity;
import androidx.core.app.ActivityCompat;
import androidx.core.content.ContextCompat;
import android.content.pm.PackageManager;
import android.os.Bundle;
import android.telephony.SmsManager;
import android.view.View;
```

```
import android.widget.EditText;
import android.widget.Toast;
import android.Manifest;

public class MainActivity extends AppCompatActivity {


   private static final int SMS_PERMISSION_CODE = 101;
   EditText etPhoneNumber, etMessage;
   @Override
   protected void onCreate(Bundle savedInstanceState) {
      super.onCreate(savedInstanceState);
      setContentView(R.layout.activity_main);
      etPhoneNumber = findViewById(R.id.etPhoneNumber);
      etMessage = findViewById(R.id.etMessage);

      if (!checkSMSPermission()) {
         requestSMSPermission();
      }
   }
   private boolean checkSMSPermission() {
       return ContextCompat.checkSelfPermission(this,
           Manifest.permission.SEND_SMS) == PackageManager.PERMISSION_GRANTED;
   }

   private void requestSMSPermission() {
       ActivityCompat.requestPermissions(this, new
           String[]{Manifest.permission.SEND_SMS}, SMS_PERMISSION_CODE);
   }
   public void onClick(View view) {
      String strPhoneNumber = etPhoneNumber.getText().toString();
      String strMessage = etMessage.getText().toString();
      try {
         SmsManager smsManager = SmsManager.getDefault();
         smsManager.sendTextMessage(strPhoneNumber,null,strMessage,null, null);
         Toast.makeText(getApplicationContext(), "Message Sent", Toast.LENGTH_LONG).show();
      } catch (Exception e) {
         Toast.makeText(getApplicationContext(), e.toString(), Toast.LENGTH_LONG).show();
      }
   }
}
```

=>Output of the program



11. Create an application to send Email

Steps:
1. Click New Project, the New Project Dialog box appears.
2. Choose Empty Views Activity then click Next.
3. Specify the Name of your project, Select the Language as Java, and Select the SDK as API 24("Nougat",Android 7.0).Click Finish Button.
4. Update the following code in activity_main.xml and MainActivity.java

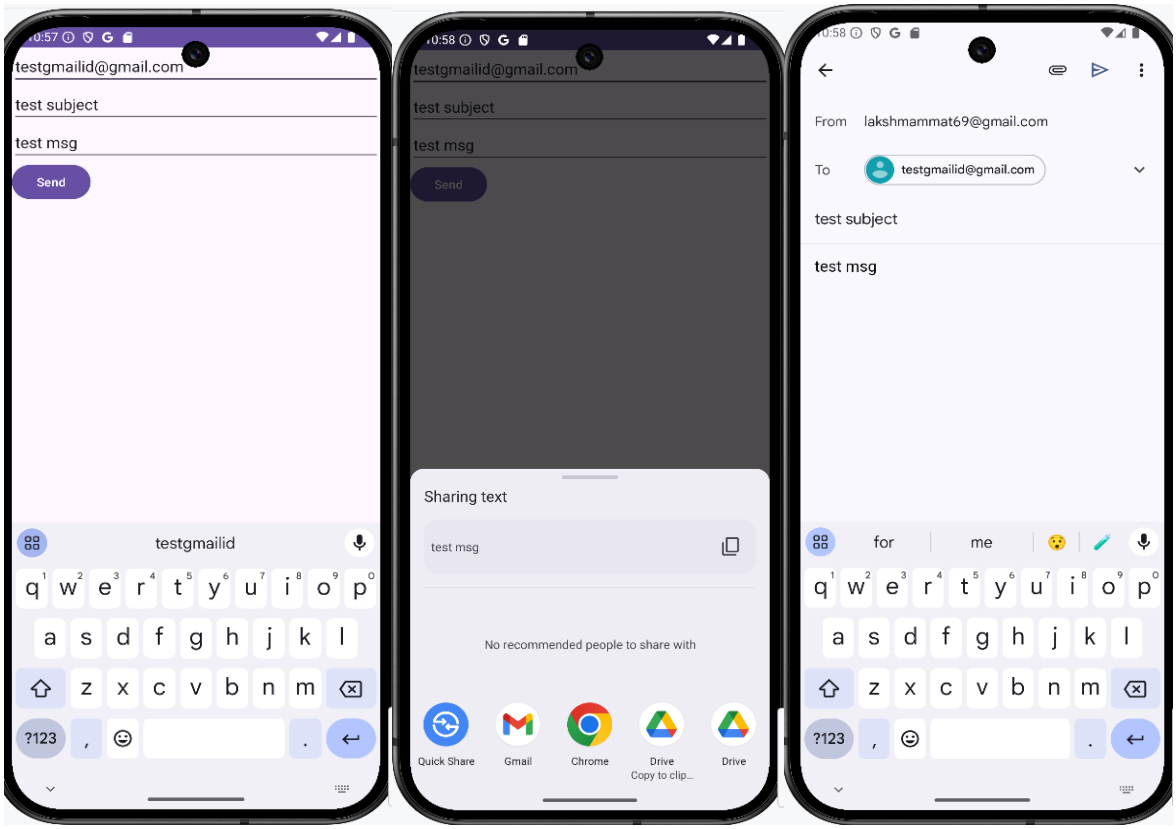
=>Coding part of Activity_main.xml

```
<?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"
   tools:context=".MainActivity">
   <EditText
      android:id="@+id/etTo"
      android:layout_width="match_parent"
      android:layout_height="wrap_content"
      android:hint="To"/>
   <EditText
      android:id="@+id/etSubject"
      android:layout_width="match_parent"
      android:layout_height="wrap_content"
      android:layout_below="@id/etTo"
      android:hint="Subject"/>
   <EditText
      android:id="@+id/etMessage"
      android:layout_width="match_parent"
      android:layout_height="wrap_content"
      android:layout_below="@id/etSubject"
      android:hint="Message"/>
   <Button
      android:id="@+id/btnSend"
      android:layout_width="wrap_content"
      android:layout_height="wrap_content"
      android:onClick="onClick"
      android:layout_below="@id/etMessage"
      android:text="Send"/>
</LinearLayout>
```

=>Coding part of MainActivity.java

```
import androidx.appcompat.app.AppCompatActivity;
import android.content.Intent;
import android.os.Bundle;
import android.view.View;
import android.widget.EditText;
public class MainActivity extends AppCompatActivity {
   EditText etTo, etSubject, etMessage;

   @Override
   protected void onCreate(Bundle savedInstanceState)
   {
      super.onCreate(savedInstanceState);
      setContentView(R.layout.activity_main);
      etTo = findViewById(R.id.etTo);
      etSubject = findViewById(R.id.etSubject);
```

```
        etMessage = findViewById(R.id.etMessage);
    }

    public void onClick(View v){
        String strTo = etTo.getText().toString().trim();
        String strSubject = etSubject.getText().toString().trim();
        String strMessage = etMessage.getText().toString().trim();
        Intent intent = new Intent(Intent.ACTION_SEND);
        intent.setType("text/plain");
        intent.putExtra(Intent.EXTRA_EMAIL, new String[]{strTo});
        intent.putExtra(Intent.EXTRA_SUBJECT, strSubject);
        intent.putExtra(Intent.EXTRA_TEXT, strMessage);
        if (intent.resolveActivity(getPackageManager()) != null)
        {
            startActivity(Intent.createChooser(intent, "Choose an email client"));
        }
    }
}
```



12. Create an Application to Display Map based on the Current/given location

Steps:
1. Click New Project, the New Project Dialog box appears.
2. Choose Empty Views Activity then click Next.
3. Specify the Name of your project, Select the Language as Java, and Select the SDK as API 24("Nougat",Android 7.0).Click Finish Button.
4. Update the following code in activity_main.xml and MainActivity.java

=>Coding part of Activity_main.xml

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

```
    tools:context=".MainActivity">
    <fragment
        android:id="@+id/map"
        android:name="com.google.android.gms.maps.SupportMapFragment"
        android:layout_width="match_parent"
        android:layout_height="match_parent"
        android:layout_alignParentTop="true"
        android:layout_alignParentBottom="true"
        android:layout_alignParentStart="true"
        android:layout_alignParentEnd="true" />
</RelativeLayout>
```

=>Coding part of MainActivity.java

```
package com.example.myapplication;
import android.os.Bundle;
import android.widget.Toast;
import androidx.annotation.NonNull;
import androidx.appcompat.app.AppCompatActivity;
import com.google.android.gms.maps.CameraUpdateFactory;
import com.google.android.gms.maps.GoogleMap;
import com.google.android.gms.maps.OnMapReadyCallback;
import com.google.android.gms.maps.SupportMapFragment;
import com.google.android.gms.maps.model.LatLng;
import com.google.android.gms.maps.model.MarkerOptions;
public class MainActivity extends AppCompatActivity implements OnMapReadyCallback {
    private GoogleMap mMap;
    private double latitude = 0.0;
    private double longitude = 0.0;
    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_main);
        // Obtain the SupportMapFragment and get notified when the map is ready to be used.
        SupportMapFragment mapFragment = (SupportMapFragment)
                getSupportFragmentManager()
                        .findFragmentById(R.id.map);
        if (mapFragment != null) {
            mapFragment.getMapAsync(this);
        } else {
            Toast.makeText(this, "Map Fragment Not Found", Toast.LENGTH_SHORT).show();
        }
    }
    @Override
    public void onMapReady(@NonNull GoogleMap googleMap) {
        mMap = googleMap;
        // Add a marker at current or given location and move the camera
        LatLng location = new LatLng(latitude, longitude);
        mMap.addMarker(new MarkerOptions().position(location).title("Marker"));
        mMap.moveCamera(CameraUpdateFactory.newLatLngZoom(location, 15));
    }
}
```

Coding part of AndroidManifest.xml

```
<?xml version="1.0" encoding="utf-8"?>
<manifest xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:tools="http://schemas.android.com/tools">
    <usespermission android:name="android.permission.ACCESS_FINE_LOCATION" />
    <usespermission android:name="android.permission.ACCESS_COARSE_LOCATION" />
    <usespermission android:name="android.permission.INTERNET" />

    <application
```

```
    android:allowBackup="true"
    android:dataExtractionRules="@xml/data_extraction_rules"
    android:fullBackupContent="@xml/backup_rules"
    android:icon="@mipmap/ic_launcher"
    android:label="@string/app_name"
    android:roundIcon="@mipmap/ic_launcher_round"
    android:supportsRtl="true"
    android:theme="@style/Theme.MyApplication"
    tools:targetApi="31">
    <metadata
      android:name="com.google.android.geo.API_KEY"
      android:value="YOUR_API_KEY_HERE" />

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

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

</manifest>
```

13. Create an Application with Login module. Check User name and password. On successful login change textview "Login Successful". On Login fail alert using Toast "Login Fail"

Steps:
1. Click New Project, the New Project Dialog box appears.
2. Choose Empty Views Activity then click Next.
3. Specify the Name of your project, Select the Language as Java, and Select the SDK as API 24("Nougat",Android 7.0).Click Finish Button.
4. Update the following code in activity_main.xml and MainActivity.java

=>Coding part of 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:id="@+id/main"
  android:layout_width="match_parent"
  android:layout_height="match_parent"
  android:orientation="vertical"
  tools:context=".MainActivity">

  <TextView
    android:id="@+id/tvTitle"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:layout_gravity="center"
    android:layout_marginTop="200dp"
    android:text="Login Form"
    android:textSize="40sp" />

  <TextView
    android:id="@+id/tvUserName"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:text="User Name"
    android:textSize="24sp" />
```

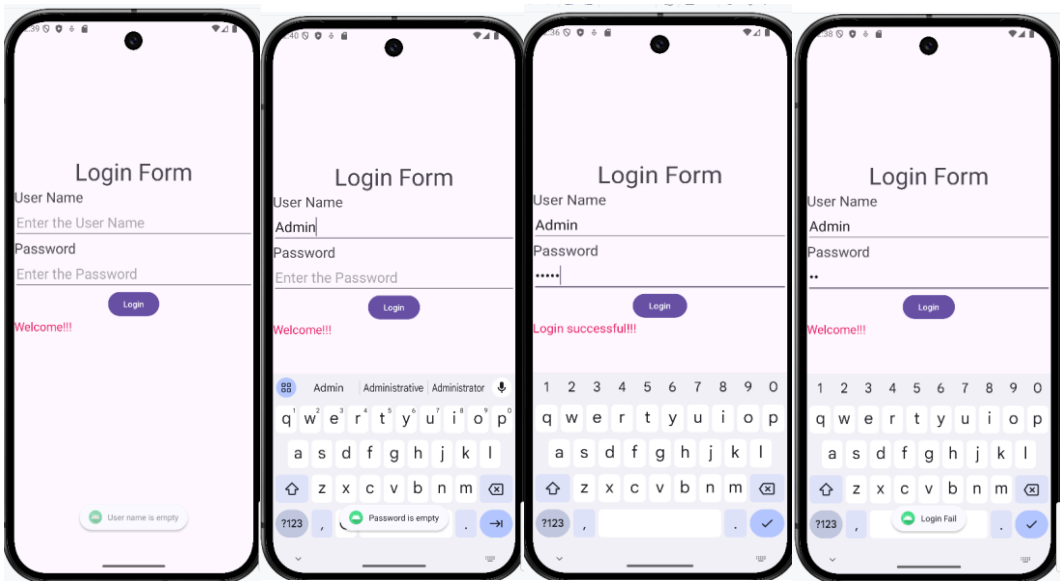
```
    <EditText
        android:id="@+id/etUserName"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:hint="Enter the User Name"
        android:inputType="text"
        android:textSize="24sp" />

    <TextView
        android:id="@+id/tvPassword"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="Password"
        android:textSize="24sp" />

    <EditText
        android:id="@+id/etPassword"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:hint="Enter the Password"
        android:inputType="textPassword"
        android:textSize="24sp" />

    <Button
        android:id="@+id/btnLogin"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_gravity="center"
        android:onClick="onClick"
        android:text="Login" />

    <TextView
        android:id="@+id/tvMessage"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="Welcome!!!"
        android:textColor="#E91E63"
        android:textSize="20sp" />

</LinearLayout>
```

=>Coding part of MainActivity.java

```
package com.example.loginapplication;

import android.os.Bundle;
import android.widget.EditText;
import android.widget.TextView;
import android.widget.Toast;
import android.view.View;
import androidx.activity.EdgeToEdge;
import androidx.appcompat.app.AppCompatActivity;
import androidx.core.graphics.Insets;
import androidx.core.view.ViewCompat;
import androidx.core.view.WindowInsetsCompat;

public class MainActivity extends AppCompatActivity {

    EditText etUserName,etPassword;
    TextView tvMessage;


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

```
        EdgeToEdge.enable(this);
        setContentView(R.layout.activity_main);

        etUserName=(EditText)findViewById(R.id.etUserName);
        etPassword=(EditText)findViewById(R.id.etPassword);
        tvMessage=(TextView)findViewById(R.id.tvMessage);
    }

public void onClick(View v)
{

    tvMessage.setText("Welcome!!!");
    if(etUserName.getText().toString().isEmpty())
    {
        Toast.makeText(this,"User name is empty", Toast.LENGTH_LONG).show();
        return;
    }
    if(etPassword.getText().toString().isEmpty())
    {
        Toast.makeText(this,"Password is empty", Toast.LENGTH_LONG).show();
        return;
    }
    if(etUserName.getText().toString().equals("Admin") &&
etPassword.getText().toString().equals("Admin"))
    {
        tvMessage.setText("Login successful!!!");
    }
    else
    {
        Toast.makeText(this, "Login Fail", Toast.LENGTH_LONG).show();
    }
}
}
```

=>OUTPUT OF THE PROGRAM



14. Learn to deploy Android applications

Steps to Deploy an Android Application

1. Prepare App (use Program 1 Hello world for this program)

2. Generate Signed APK (Android Package Kit):

a. In Android Studio, navigate to Build > Generate Signed Bundle/APK.

b. Follow the prompts to create a new keystore or use an existing one. A keystore is a binary file that contains a set of private keys.

c. Configure the build type (release) and signing configuration.

d. Generate the signed APK file.

3. **Test your signed APK**:

a. Before distributing your app, test the signed APK to ensure that the signing process didn't introduce any issues.

b. Install the APK on various devices and perform thorough testing.

c. Release on Google Play Console:

d. Sign in to the Google Play Console (https://play.google.com/apps/publish).

e. Create a new app entry if this is your first release or select an existing app.

f. Complete all the required information for the app listing, including the title, description, screenshots, and categorization

g. Upload your signed APK file.

h. Set pricing and distribution options.

i. Optimize your store listing for search and conversion.
Once everything is set, click the "Publish" button to release your app to the Google Play Store.

4. Other Distribution Channels (Optional):

- Besides Google Play, you can distribute your app through other channels such as Amazon Appstore, Samsung Galaxy Store, or third party app marketplaces.
- Each distribution channel may have its own requirements and submission process, so be sure to follow their guidelines.

5. Monitor and Update:

- Keep an eye on user feedback and app performance metrics through the Google Play Console.
- Regularly update your app to fix bugs, add new features, and improve user experience based on feedback.