

INDEX

SR NO	NAME OF THE EXPERIMENT	DATE	FACULTY SIGN	CO
1	<p>Introduction to Android Application Components and UI Controls:</p> <ol style="list-style-type: none"> 1. Write a program to demonstrate activity life cycle. 2. Create an Employee Registration form using Linear Layout and Relative Layout. 3. Design a screen that displays the frame image and write a quote on that. 4. Create an android application that displays an image using frame layout and when the user clicks on that image another image should be displayed on the screen. 5. Create an android application to add two numbers and display result in Toast Message and AlertDialog. 6. Create an Android application for the student registration form using the relative layout. Display the entered details on the second page using explicit intent. 7. Create an application to implement implicit intent with functionality to open camera, Gallery, Contact, Dial, Browser. 8. Create an application which has two buttons. When the user clicks on the first button the first fragment will be displayed and when the user clicks on the second button the second fragment will be displayed. 9. Write an Android application with five checkboxes to list the 5 subjects of your class and radio buttons to display gender. Display the selected subject name when you click any one of the checkboxes and gender in the alert dialog box. 10. Create a basic calculator to perform arithmetic operations with divide-by-zero validation. (using Alert box). 11. Create an Android application to demonstrate List View using an array adapter. 12. Create a mobile application for a currency converter. Use a spinner for selecting the currency. 13. Write an application to increase font size using seekbar. 14. Create an Android application to demonstrate progressbar. 			CO1

2	Database Connectivity: 1. Create an Android application to read and write content in internal storage. 2. Create an Android application to read and write content in external storage. 3. Write an android program for shared preference to store value in name-value pairs. 4. Create a login form with a remember me checkbox. Save the username and password if the checkbox is checked using shared preference and show the welcome page when the login button is clicked. 5. Create an Android application to insert, update, select, and delete records from the student table using SQLite Database. 6. Write a program to create a user registration form, after registration data will be inserted in the SQLite database, and design an activity that displays that information. 7. Android Program to perform CRUD operation using real time database Firebase.			CO2
3	Animation, Multimedia and Location Based Services: 1. Write an Android application to play, pause, and stop an audio file. 2. Write an Android application to play a video with Media controller. 3. Create an android application that applies different animations on an image. 4. Create an Android application to implement frame animation. 5. Create an Android application to display the current location of your device (display longitude and latitude values). 6. Create an Android application that displays the current location of your device from longitude and latitude values (Reverse Geocoding). 7. Create an Android application that accepts longitude and latitude from the user and marks that location on google map.			CO3
4	REST API integration: 1. Create an Android application to demonstrate JSON data parsing using OkHttp (you can use https://api.github.com/users JSON data). 2. Create an Android application to demonstrate JSON data parsing using Volley (you can use https://api.github.com/users JSON data).			CO4

	3. Create an Android application to demonstrate JSON data parsing using Retrofit (you can use https://api.github.com/users JSON data).			
5	<p>Introduction to Dart and Flutter:</p> <ol style="list-style-type: none"> 1. Write a Flutter program to demonstrate Text widget and its properties. 2. Write a Flutter program to display dog names (demonstrate stateless widget and column widgets). 3. Write a Flutter program that allows the user to enter a city in a text field and displays city name (demonstrate stateful widget). 4. Write a Flutter program to change the background color (demonstrate stateful widget). 5. Write a Flutter Program to display fruit list using ListView. 6. Write a Flutter program to demonstrate navigation (user should be navigated from first screen to second screen). 7. Write a Flutter program to design a Login form using TextField, Check Box, Buttons, Drop down, Switch etc.. 			CO5
6	<p>Data Handling in Flutter:</p> <ol style="list-style-type: none"> 1. Write a Flutter program based on RestAPI to fetch data. 2. Write a flutter program to demonstrate JSON serialization and Deserialization. 3. Write a flutter program to perform CRUD operations using sqflite. 			CO6

Module 1:**Introduction to Android Application Components and UI Controls:****1. Write a program to demonstrate activity life cycle.****MainActivity.java:**

```
package com.example.myapplication;

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.os.Bundle;
import android.util.Log;
import android.widget.Toast;

public class MainActivity extends AppCompatActivity {

    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_main);
        Log.d("lifecycle", "onCreate invoked");
        Toast.makeText(getApplicationContext(), "onCreate invoked",
Toast.LENGTH_LONG).show();
    }

    @Override
    protected void onStart() {
        super.onStart();
        Log.d("lifecycle", "onStart invoked");
        Toast.makeText(getApplicationContext(), "onStart invoked", Toast.LENGTH_LONG).show();
    }

    @Override
    protected void onResume() {
        super.onResume();
        Log.d("lifecycle", "onResume invoked");
        Toast.makeText(getApplicationContext(), "onResume invoked",
Toast.LENGTH_LONG).show();
    }

    @Override
    protected void onPause() {
        super.onPause();
        Log.d("lifecycle", "onPause invoked");
        Toast.makeText(getApplicationContext(), "onPause invoked", Toast.LENGTH_LONG).show();
    }
}
```

```
}

@Override
protected void onStop() {
    super.onStop();
    Log.d("lifecycle", "onStop invoked");
    Toast.makeText(getApplicationContext(), "onStop invoked", Toast.LENGTH_LONG).show();
}

@Override
protected void onRestart() {
    super.onRestart();
    Log.d("lifecycle", "onRestart invoked");
    Toast.makeText(getApplicationContext(), "onRestart invoked",
Toast.LENGTH_LONG).show();
}

@Override
protected void onDestroy() {
    super.onDestroy();
    Log.d("lifecycle", "onDestroy invoked");
    Toast.makeText(getApplicationContext(), "onDestroy invoked",
Toast.LENGTH_LONG).show();
}
```

Output:

My Application > app > build.gradle.kts (app) > android

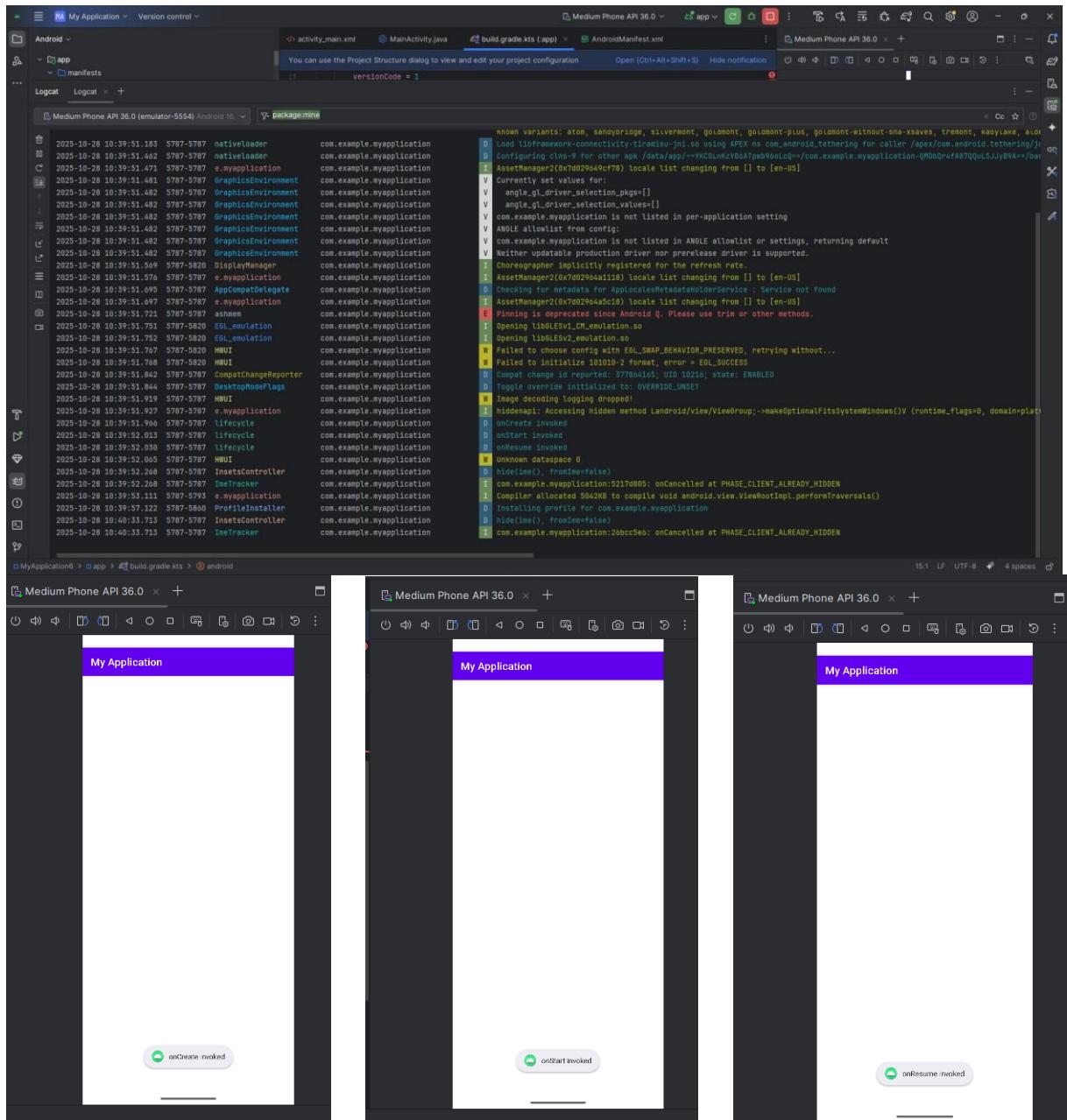
Medium Phone API 36.0 > app > AndroidManifest.xml > Medium Phone API 36.0 > +

Android > app > manifests > Logcat > Logcat > + > package:mine

Medium Phone API 36.0 (emulator-5554) Android 16. > package:mine

```
beginning of main
beginning of system
beginning of crash
2025-10-28 10:39:50.885 2925-2300 Finsky com.android.vending
2025-10-28 10:39:51.887 2925-2298 Finsky com.android.vending
2025-10-28 10:39:50.984 5787-5787 e.myapplication com.example.myapplication
2025-10-28 10:39:51.112 5787-5787 e.myapplication com.example.myapplication
2025-10-28 10:39:51.113 5787-5787 e.myapplication com.example.myapplication

[2] ItemStore: getIItemNames RPC failed for item com.example.myapplication
[0] ItemStore: getIItemNames RPC failed for item com.example.myapplication
I Late-enabling Xcheck:jni
I Using CollectorTypeCMC GC.
W Unexpected CPU variant for x86: x86_64.
Known variants: atom sandymobile silvermont goldmont goldmont-plus goldmont-without-sha-xsvases tremont kabylake alder Lake
D Load framework-connectivity-tiramisu-jni.so using APEX ns com_android_tethering for caller /apex/com.android.tethering/
D Configuring ctsn-9 for other apk /data/app/_YQCOLNkYQDAtpw90LcLw==/com.example.myapplication-QM0Qr4FAE7QQuLJjy89A==/base.apk
I AssetManager@2(Ox7d0294a9f7e8) locale list changing from [] to [en-US]
V Currently set values for:
V   _angle_gl_driver_selection_pkgs={}
V   _angle_gl_driver_selection_values={}
V com.example.myapplication is not listed in per-application setting
V ANGLE allowlist from config:
V com.example.myapplication is not listed in ANGLE allowlist or settings, returning default
V Neither updatable production driver nor preloaded driver is supported.
I Choreographer implicitly registered for the refresh rate.
I AssetManager@2(Ox7d0294a1118) locale list changing from [] to [en-US]
D Checking for metadata for ApplicableMetadataHolderService : Service not found
I AssetManager@2(Ox7d0294a5c1b) locale list changing from [] to [en-US]
E Planning is deprecated since Android Q. Please use trim or other methods.
I Opening libLESv1_CMulation.so
I Opening libLESv2_CMulation.so
W Failed to choose config with EGL_SWAP_BEHAVIOR_PRESERVED, retrying without...
W Failed to initialize 10010-2 format, error = EGL_SUCCESS
D Compat change id reported: 77884160; UID: 0; PID: 0; state: ENABLED
D Toggle override enabled to: OVERRIDE_UNSET
W Image decoding logging dropped!
I hidnappe: Accessing hidden method Landroid/view/ViewGroup;::makeOptionalFitsSystemWindows()V (runtime_flags=0, domain=platform)
D onCreate invoked
D onStart invoked
D onResume invoked
```



2. Create an Employee Registration form using Linear Layout and Relative Layout.

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/tvHeader"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="Employee Registration"
        android:textSize="24sp"
        android:textStyle="bold"
        android:layout_gravity="center"
        android:paddingBottom="16dp"/>
    <RelativeLayout android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:layout_marginBottom="16dp">
        <EditText android:id="@+id/etEmployeeId"
            android:layout_width="match_parent"
            android:layout_height="wrap_content"
            android:layout_alignParentEnd="true"
            android:layout_marginStart="16dp"
            android:hint="Enter Employee ID"
            android:minHeight="48dp"
            android:inputType="text" />
        <EditText
            android:id="@+id/etName"
            android:layout_width="match_parent"
            android:layout_height="wrap_content"
            android:layout_below="@+id/etEmployeeId"
            android:layout_alignParentEnd="true"
            android:layout_marginStart="16dp"
            android:hint="Enter Name"
            android:inputType="textPersonName"
            android:minHeight="48dp" />
        <EditText android:id="@+id/etDepartment"
            android:layout_width="match_parent"
            android:layout_height="wrap_content"
            android:layout_alignParentEnd="true"
            android:layout_marginStart="16dp"
            android:layout_below="@+id/etName"
            android:hint="Enter Department"
```

```
        android:inputType="text"
        android:minHeight="48dp" />
<EditText android:id="@+id/etEmail"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:layout_alignParentEnd="true"
        android:layout_marginStart="16dp"
        android:layout_below="@+id/etDepartment"
        android:hint="Enter Email"
        android:inputType="textEmailAddress"
        android:minHeight="48dp" />
<EditText
        android:id="@+id/etPhone"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:layout_alignParentEnd="true"
        android:layout_marginStart="16dp"
        android:layout_below="@+id/etEmail"
        android:hint="Enter Phone Number"
        android:inputType="phone"
        android:minHeight="48dp" />
</RelativeLayout>
<TextView
        android:id="@+id/tvgender"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="Gender"
        android:layout_marginStart="16dp"
        android:textSize="20sp"
        android:textStyle="bold"
        android:paddingBottom="16dp"/>

<RadioGroup
        android:id="@+id/group_gender"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_marginStart="16dp">

    <RadioButton android:id="@+id/radioButton"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_marginEnd="20dp"
        android:text="Male" />
    <RadioButton
        android:id="@+id/radioButton2"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="Female"
        android:layout_marginEnd="20dp"/>
    <RadioButton android:id="@+id/radioButton3"
        android:layout_width="wrap_content"
```

```
        android:layout_height="wrap_content"
        android:text="Transgender" />

    </RadioGroup>
    <TextView
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="Hobbies"
        android:textSize="10pt"
        android:textStyle="bold"
        android:id="@+id/TextView2"
        android:layout_margin="15dp"/>
    <RelativeLayout
        android:layout_width="match_parent"
        android:layout_height="wrap_content">
        <CheckBox
            android:id="@+id/checkBox"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:text="Cricket"
            android:layout_marginLeft="10dp"/>

        <CheckBox
            android:id="@+id/checkBox2"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:text="Dance"
            android:layout_toRightOf="@+id/checkBox"
            android:layout_marginLeft="2dp"/>

        <CheckBox
            android:id="@+id/checkBox3"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:text="Football"
            android:layout_toRightOf="@+id/checkBox2"
            android:layout_marginLeft="2dp"/>
    </RelativeLayout>

    <Button
        android:id="@+id	btnSubmit"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_gravity="center_horizontal"
        android:layout_marginTop="20dp"
        android:background="@color/purple_200"
        android:minHeight="32dp"
        android:text="Submit"
        android:textColor="#311B92"
        tools:ignore="TouchTargetSizeCheck" />
</LinearLayout>
```

MainActivity.java:

```
package com.example.myapplication;

import android.content.Intent;
import android.os.Bundle;
import android.view.View;
import android.widget.Button;
import android.widget.CheckBox;
import android.widget.EditText;
import android.widget.RadioButton;
import android.widget.RadioGroup;
import android.widget.Toast;

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 {
    Button submit;
    EditText id, phone;
    RadioGroup gender;
    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);

        setContentView(R.layout.activity_main);
        submit=findViewById(R.id.btnSubmit);
        id=findViewById(R.id.etEmployeeId);
        phone=findViewById(R.id.etPhone);
        gender=findViewById(R.id.group_gender);
        CheckBox cricket=findViewById(R.id.checkBox);
        CheckBox dance=findViewById(R.id.checkBox2);
        CheckBox football=findViewById(R.id.checkBox3);

        submit.setOnClickListener(new View.OnClickListener() {
            @Override
            public void onClick(View v) {
                String result="";
                Intent i=new Intent(MainActivity.this,
com.example.myapplication.MainActivity2.class);
                i.putExtra("id",id.getText().toString());
                i.putExtra("phone",phone.getText().toString());
                int radioId=gender.getCheckedRadioButtonId();
                RadioButton selected=findViewById(radioId);
                i.putExtra("gender",selected.getText().toString());
                if(cricket.isChecked())
                    result+=" Cricket";
                if(dance.isChecked())

```

```
        result+=" Dance";
        if(football.isChecked())
            result+=" Football";
        i.putExtra("hobbies",result);

        Toast.makeText(getApplicationContext(),"Hobbies:"+result,Toast.LENGTH_LONG).show();
        startActivity(i);
    }
});
```

MainActivity2.java:

```
package com.example.myapplication;

import android.os.Bundle;
import android.widget.TextView;

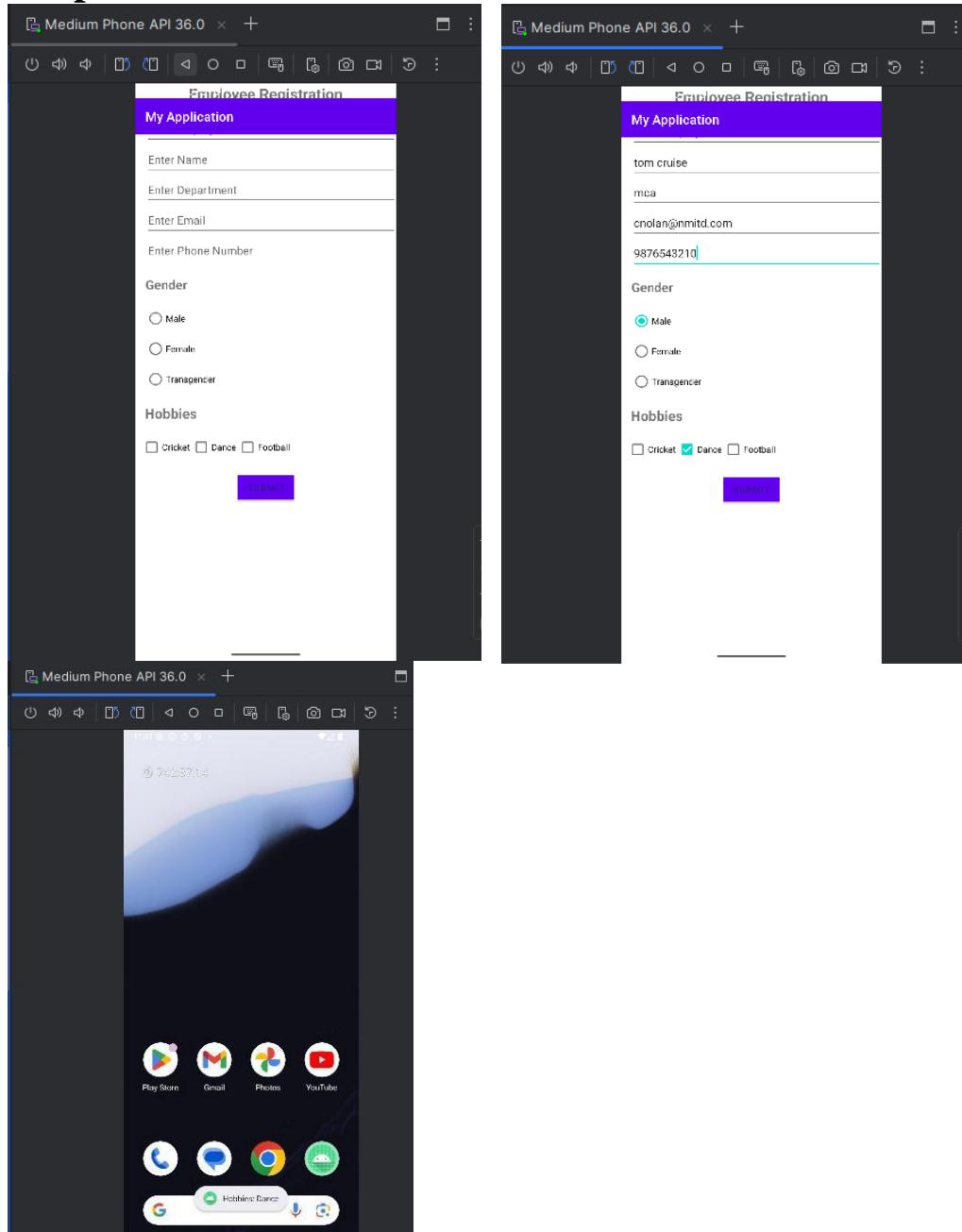
import androidx.appcompat.app.AlertDialog;
import androidx.appcompat.app.AppCompatActivity;

public class MainActivity2 extends AppCompatActivity {
    TextView display;
    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);

        setContentView(R.layout.activity_main);
        display=findViewById(R.id.TextView2);
        String id=getIntent().getStringExtra("id");
        String phone=getIntent().getStringExtra("phone");
        String gender=getIntent().getStringExtra("gender");
        String hobbies=getIntent().getStringExtra("hobbies");
        display.setText("Employee Id: "+id+"\nPhone number: "+ phone+"\nGender: "+gender+"\nhobbies: "+hobbies);
        AlertDialog.Builder builder=new AlertDialog.Builder(MainActivity2.this);
        builder.setCancelable(true);
        builder.setMessage("id: "+id+"\nPhone no.: "+phone+"\nGender: "+gender+"\nHobbies: "+hobbies);
        builder.setTitle("Employee details");
        builder.show();

    }
}
```

Output:



3. Design a screen that displays the frame image and write a quote on that.

Activity_main.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:id="@+id/main"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    android:background="#FFFFFF"
    tools:context=".MainActivity">

    <!-- Frame image as background -->
    <ImageView
        android:id="@+id/frameImage"
        android:layout_width="match_parent"
        android:layout_height="match_parent"
        android:scaleType="centerCrop"
        android:src="@drawable/lewis" />

    <!-- Quote text displayed on top -->
    <TextView
        android:id="@+id/quoteText"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_gravity="center"
        android:text="The best way to predict the future is to create it."
        android:textColor="#FFFFFF"
        android:textSize="22sp"
        android:padding="24dp"
        android:background="#80000000"
        android:gravity="center"
        android:textStyle="italic"
        android:fontFamily="sans-serif-medium" />

</FrameLayout>
```

MainActivity.java:

```
package com.example.myapplication;

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 androidx.appcompat.app.AppCompatActivity;
import android.os.Bundle;
```

```
import android.widget.ImageView;
import android.widget.TextView;

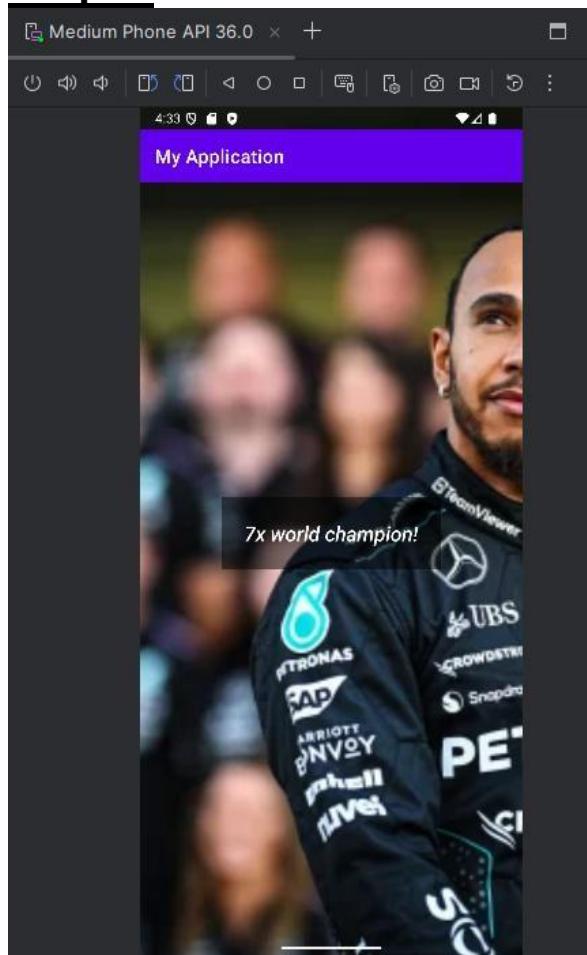
public class MainActivity extends AppCompatActivity {

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

        // Reference views (optional, if you want to change them dynamically)
        ImageView frameImage = findViewById(R.id.frameImage);
        TextView quoteText = findViewById(R.id.quoteText);

        // Optional: change text or image programmatically
        quoteText.setText("7x world champion!");
        // frameImage.setImageResource(R.drawable.another_frame); // Example if you want to
        swap the image
    }
}
```

Output:

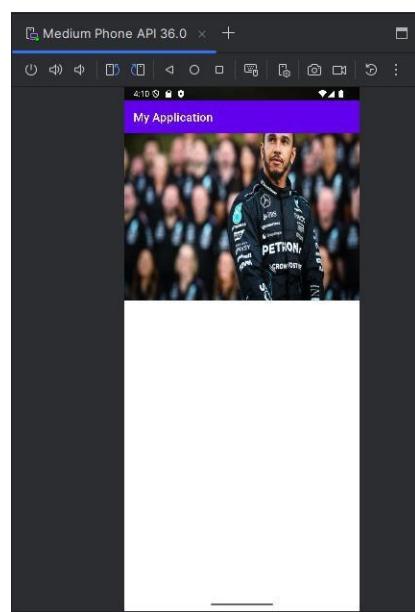
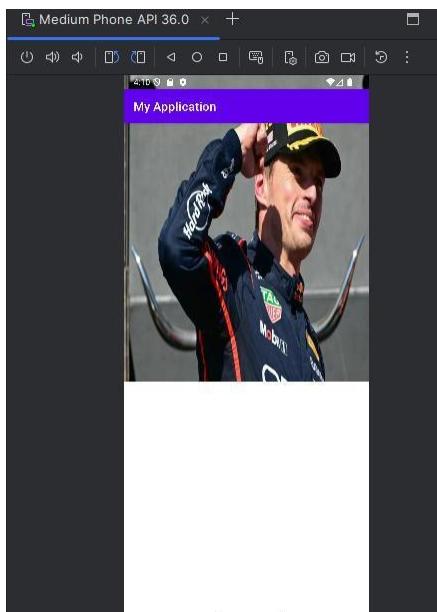


4. **Create an android application that displays an image using frame layout and when the user clicks on that image another image should be displayed on the screen.**

```
Activity_main.xml:  
plugins {  
    alias(libs.plugins.android.application)  
}  
  
android {  
    namespace = "com.example.myapplication"  
    compileSdk = 36 // ■ updated from 35 → 36  
  
    defaultConfig {  
        applicationId = "com.example.myapplication"  
        minSdk = 24  
        targetSdk = 36 // ■ updated from 35 → 36  
        versionCode = 1  
        versionName = "1.0"  
  
        testInstrumentationRunner = "androidx.test.runner.AndroidJUnitRunner"  
    }  
  
    buildTypes {  
        release {  
            isMinifyEnabled = false  
            proguardFiles(  
                getDefaultProguardFile("proguard-android-optimize.txt"),  
                "proguard-rules.pro"  
            )  
        }  
    }  
  
    compileOptions {  
        // ■ Use Java 17 (recommended for new Android Gradle Plugin versions)  
        sourceCompatibility = JavaVersion.VERSION_17  
        targetCompatibility = JavaVersion.VERSION_17  
    }  
}  
  
dependencies {  
    implementation(libs.appcompat)  
    implementation(libs.material)  
    implementation(libs.activity)  
    implementation(libs.constraintlayout)  
  
    testImplementation(libs.junit)  
    androidTestImplementation(libs.ext.junit)  
    androidTestImplementation(libs.espresso.core)  
}
```

```
MainActivity.java:  
package com.example.myapplication;  
  
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 androidx.appcompat.app.AppCompatActivity;  
  
import android.os.Bundle;  
import android.view.View;  
import android.widget.ImageView;  
  
public class MainActivity extends AppCompatActivity {  
    ImageView image1, image2;  
  
    @Override  
    protected void onCreate(Bundle savedInstanceState) {  
        super.onCreate(savedInstanceState);  
        setContentView(R.layout.activity_main);  
        image1 = findViewById(R.id.imageView1);  
        image2 = findViewById(R.id.imageView2);  
        //Set OnClickListener using anonymous class:  
        image1.setOnClickListener(new View.OnClickListener() {  
            @Override  
            public void onClick(View view) {  
                image1.setVisibility(View.GONE);  
                image2.setVisibility(View.VISIBLE);  
            }  
        });  
        //Set OnClickListener using anonymous class:  
        image2.setOnClickListener(new View.OnClickListener() {  
            @Override  
            public void onClick(View view) {  
                image1.setVisibility(View.VISIBLE);  
                image2.setVisibility(View.GONE);  
            }  
        });  
    }  
}
```

output:



5. Create an android application to add two numbers and display result in Toast Message and AlertDialog.

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">

    <EditText
        android:id="@+id/Num1"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:ems="10"
        android:hint="Enter number 1"
        android:layout_marginTop="100dp"
        android:inputType="number" />

    <EditText
        android:id="@+id/Num2"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:ems="10"
        android:hint="Enter number 2"
        android:inputType="number" />

    <Button
        android:id="@+id/button"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_gravity="center"
        android:padding="50dp"
        android:text="Add" />

</LinearLayout>
```

MainActivity.java:

```
package com.example.myapplication;

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 androidx.appcompat.app.AppCompatActivity;
import android.os.Bundle;
import android.view.View;
import android.widget.Button;
import android.widget.EditText;
import android.widget.Toast;
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 {

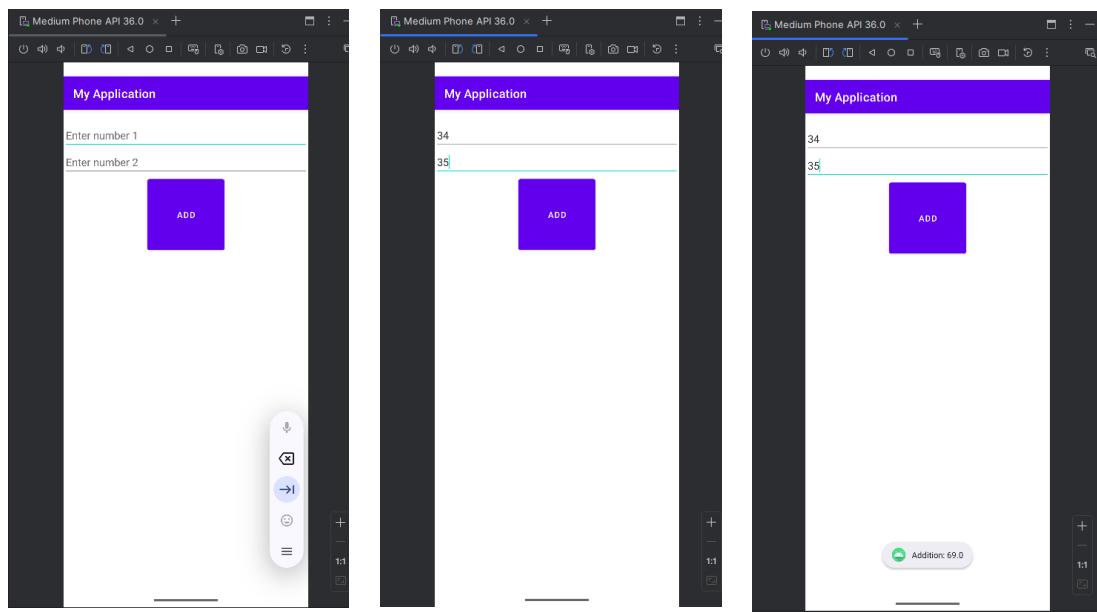
    // UI elements
    private EditText number1;
    private EditText number2;
    private Button addButton;

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

        // Initialize views
        number1 = findViewById(R.id.Num1);
        number2 = findViewById(R.id.Num2);
        addButton = findViewById(R.id.button);

        // Set click listener
        addButton.setOnClickListener(new View.OnClickListener() {
            @Override
            public void onClick(View v) {
                double num1 = Double.parseDouble(number1.getText().toString());
                double num2 = Double.parseDouble(number2.getText().toString());
                double sum = num1 + num2;

                Toast.makeText(
                    getApplicationContext(),
                    "Addition: " + sum,
                    Toast.LENGTH_LONG
                ).show();
            }
        });
    }
}
```

output:

6. Create an Android application for the student registration form using the relative layout. Display the entered details on the second page using explicit intent.

MODULE 1 : PRACTICAL 6

activity_main.xml:

```
<?xml version="1.0" encoding="utf-8"?>
<RelativeLayout
    xmlns:android="http://schemas.android.com/apk/res/android"
    android:id="@+id/main"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    android:padding="16dp">

    <TextView
        android:id="@+id/tvHeader"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="Student Registration"
        android:textSize="24sp"
        android:textStyle="bold"
        android:layout_centerHorizontal="true"
        android:paddingBottom="24dp"/>

    <EditText
        android:id="@+id/etName"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:hint="Enter Name"
        android:inputType="textPersonName"
        android:minHeight="48dp"
        android:layout_below="@+id/tvHeader"
        android:layout_marginTop="16dp"/>

    <EditText
        android:id="@+id/etRollNo"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:hint="Enter Roll Number"
        android:inputType="number"
        android:minHeight="48dp"
        android:layout_below="@+id/etName"
        android:layout_marginTop="16dp"/>

    <EditText
        android:id="@+id/etCourse"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:hint="Enter Course"
        android:inputType="text"
```

```
        android:minHeight="48dp"
        android:layout_below="@+id/etRollNo"
        android:layout_marginTop="16dp"/>

    <Button
        android:id="@+id	btnSubmit"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="Submit"
        android:layout_below="@+id/etCourse"
        android:layout_centerHorizontal="true"
        android:layout_marginTop="24dp"/>

</RelativeLayout>
activity_main2.xml:
<?xml version="1.0" encoding="utf-8"?>
<RelativeLayout
    xmlns:android="http://schemas.android.com/apk/res/android"
    android:id="@+id/displayLayout"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    android:padding="16dp">

    <TextView
        android:id="@+id/tvDisplayHeader"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="Student Details"
        android:textSize="24sp"
        android:textStyle="bold"
        android:layout_centerHorizontal="true"
        android:paddingBottom="24dp"/>

    <TextView
        android:id="@+id/tvName"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="Name:"
        android:layout_below="@+id/tvDisplayHeader"
        android:textSize="18sp"/>

    <TextView
        android:id="@+id/tvRollNo"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="Roll No:"
        android:layout_below="@+id/tvName"
        android:layout_marginTop="16dp"
        android:textSize="18sp"/>

    <TextView
```

```
        android:id="@+id/tvCourse"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="Course:"
        android:layout_below="@+id/tvRollNo"
        android:layout_marginTop="16dp"
        android:textSize="18sp"/>

    </RelativeLayout>
>MainActivity.java:
package com.example.myapplication;

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 androidx.appcompat.app.AppCompatActivity;
import android.content.Intent;
import android.os.Bundle;
import android.view.View;
import android.widget.Button;
import android.widget.EditText;

public class MainActivity extends AppCompatActivity {

    EditText etName, etRollNo, etCourse;
    Button btnSubmit;

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

        etName = findViewById(R.id.etName);
        etRollNo = findViewById(R.id.etRollNo);
        etCourse = findViewById(R.id.etCourse);
        btnSubmit = findViewById(R.id.btnSubmit);

        btnSubmit.setOnClickListener(new View.OnClickListener() {
            @Override
            public void onClick(View view) {
                String name = etName.getText().toString();
                String rollNo = etRollNo.getText().toString();
                String course = etCourse.getText().toString();

                Intent intent = new Intent(MainActivity.this, MainActivity2.class);
                intent.putExtra("name", name);
                intent.putExtra("rollNo", rollNo);
            }
        });
    }
}
```

```
        intent.putExtra("course", course);
        startActivity(intent);

    }
});

}

}

MainActivity2.java:
package com.example.myapplication;

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 androidx.appcompat.app.AppCompatActivity;
import android.os.Bundle;
import android.widget.TextView;

public class MainActivity2 extends AppCompatActivity {

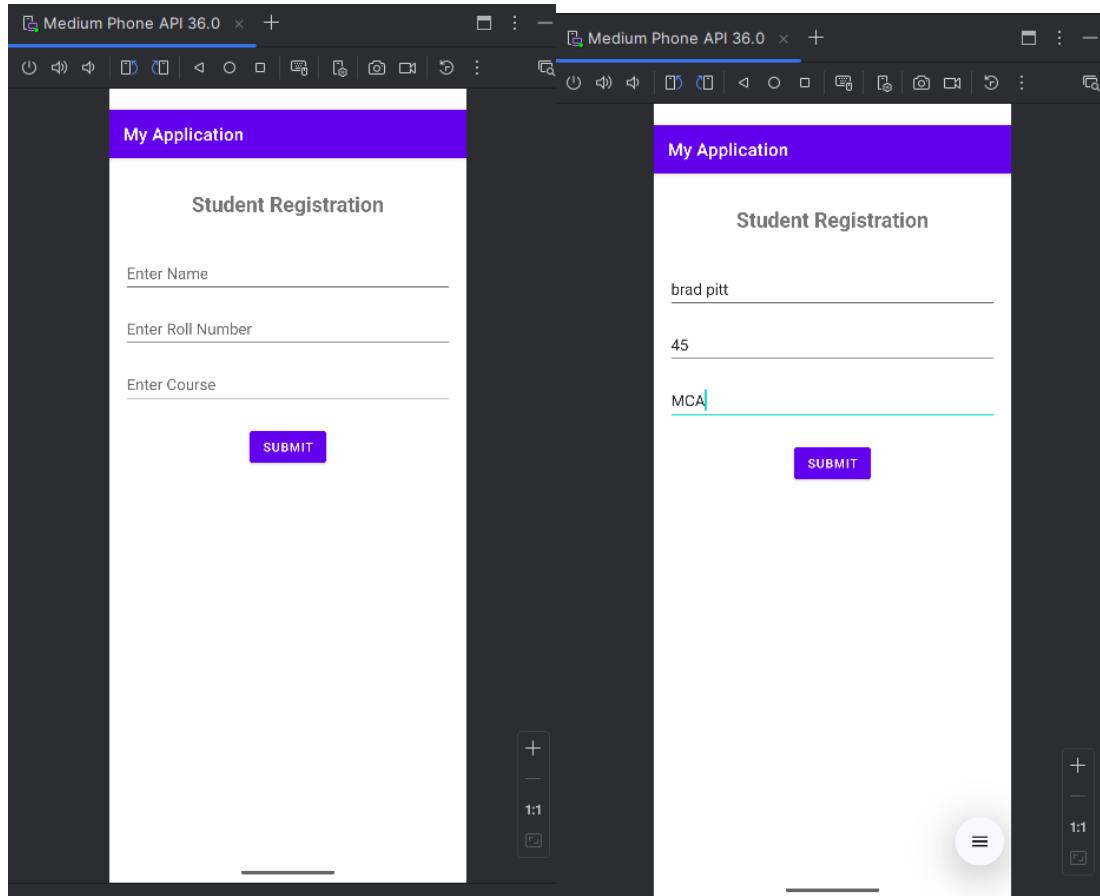
    TextView tvName, tvRollNo, tvCourse;

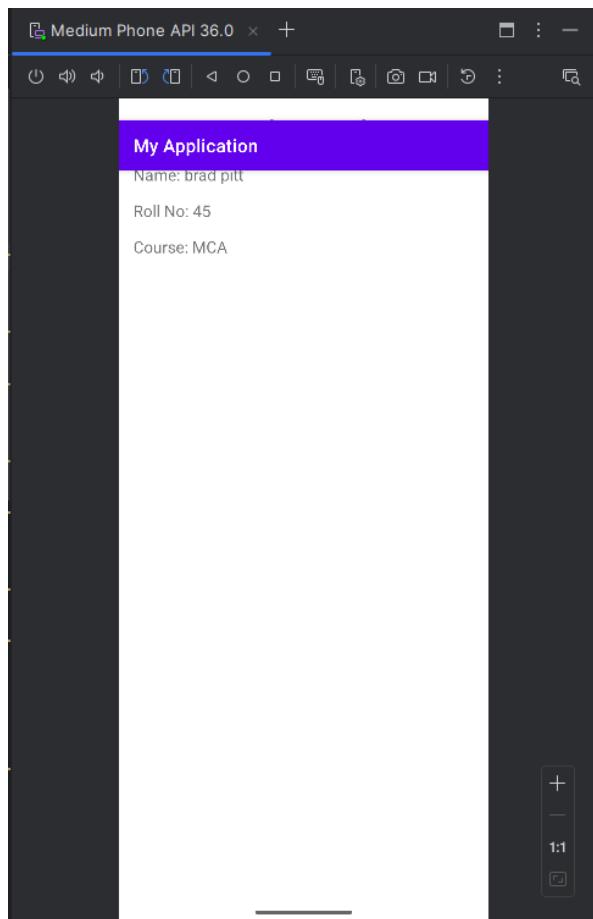
    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_main2);

        tvName = findViewById(R.id.tvName);
        tvRollNo = findViewById(R.id.tvRollNo);
        tvCourse = findViewById(R.id.tvCourse);

        // Get data from intent
        String name = getIntent().getStringExtra("name");
        String rollNo = getIntent().getStringExtra("rollNo");
        String course = getIntent().getStringExtra("course");

        tvName.setText("Name: " + name);
        tvRollNo.setText("Roll No: " + rollNo);
        tvCourse.setText("Course: " + course);
    }
}
```

output:



7. Create an application to implement implicit intent with functionality to open camera, Gallery, Contact, Dial, Browser.

```
package com.example.practical4;

import android.content.Intent;
import android.net.Uri;
import android.os.Bundle;
import android.provider.MediaStore;
import android.view.View;
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;

public class MainActivity extends AppCompatActivity {

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

        setContentView(R.layout.activity_main);
        findViewById(R.id.button2).setOnClickListener(new View.OnClickListener() {
            @Override
            public void onClick(View v) {
                Intent i=new Intent(MediaStore.ACTION_IMAGE_CAPTURE);
                startActivity(i);
            }
        });
        findViewById(R.id.button6).setOnClickListener(new View.OnClickListener() {
            @Override
            public void onClick(View v) {
                Intent i=new Intent(Intent.ACTION_VIEW);
                i.setData(Uri.parse("content://media/external/images/media/"));
                startActivity(i);
            }
        });
        findViewById(R.id.button5).setOnClickListener(new View.OnClickListener() {
            @Override
            public void onClick(View v) {
                Intent i=new Intent(Intent.ACTION_VIEW);
                i.setData(Uri.parse("content://contacts/people/"));
                startActivity(i);
            }
        });
        EditText ed=findViewById(R.id.editTextText2);
        findViewById(R.id.button7).setOnClickListener(new View.OnClickListener() {
```

```
    @Override
    public void onClick(View v) {
        Intent i=new Intent(Intent.ACTION_VIEW);
        i.setData(Uri.parse("tel:"+ed.getText()));
        startActivity(i);
    }
});

findViewById(R.id.button8).setOnClickListener(new View.OnClickListener() {
    @Override
    public void onClick(View v) {

        Intent i=new Intent(Intent.ACTION_VIEW,Uri.parse("http://"+ed.getText()+"/"));
        startActivity(i);
    }
});

}

<?xml version="1.0" encoding="utf-8"?>
<GridLayout 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:columnCount="2"
    android:layout_margin="20dp"
    android:padding="10dp">

<EditText
    android:id="@+id/editTextText2"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:layout_row="0"
    android:layout_column="0"
    android:layout_marginTop="50dp"
    android:ems="10"
    android:hint="Phone number"
    android:inputType="text" />

<Button
    android:id="@+id/button8"
```

```
    android:layout_width="120dp"
    android:layout_height="wrap_content"
    android:layout_row="3"
    android:layout_column="1"
    android:text="Browser" />

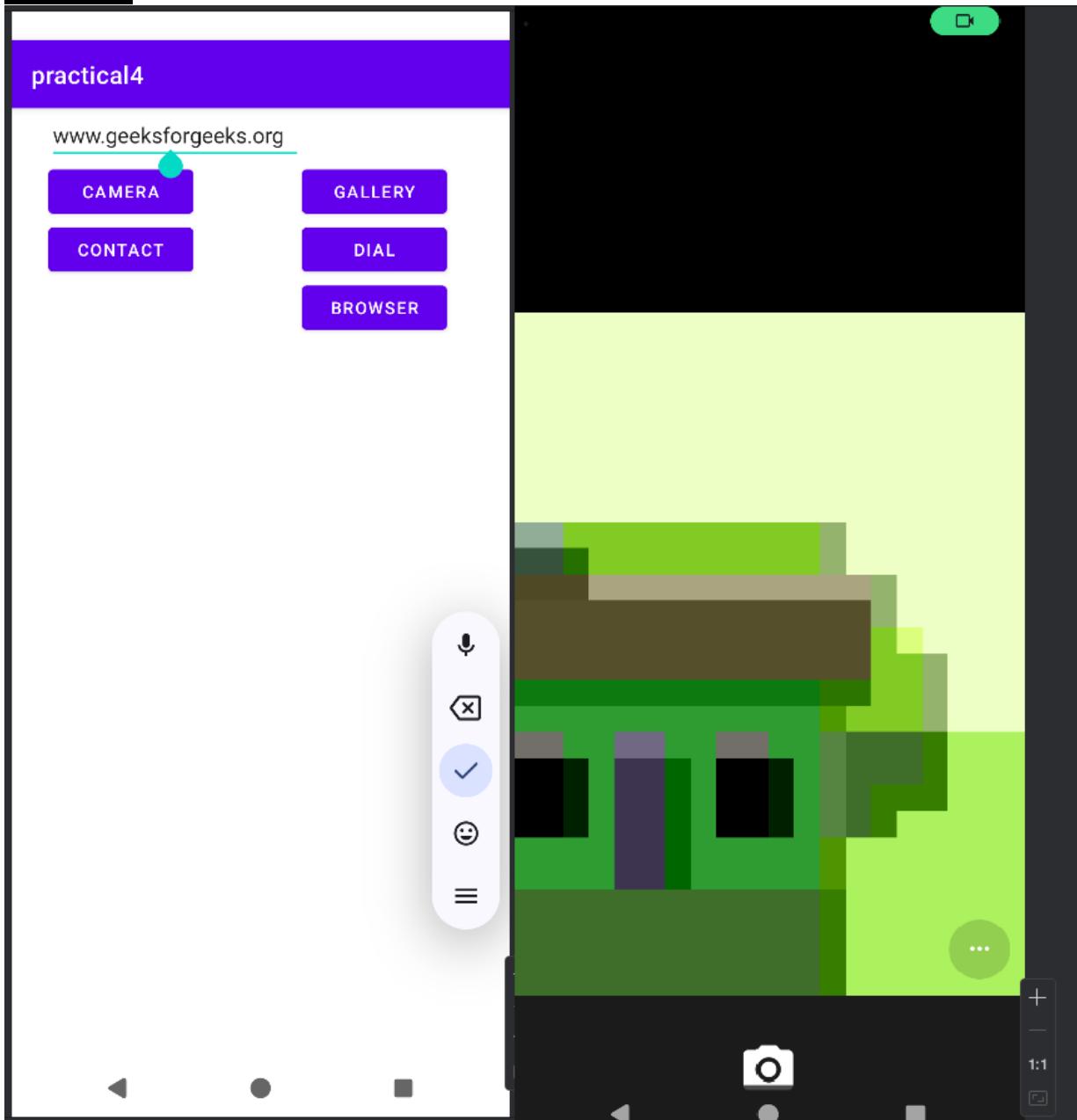
<Button
    android:id="@+id/button7"
    android:layout_width="120dp"
    android:layout_height="wrap_content"
    android:layout_row="2"
    android:layout_column="1"
    android:text="Dial" />

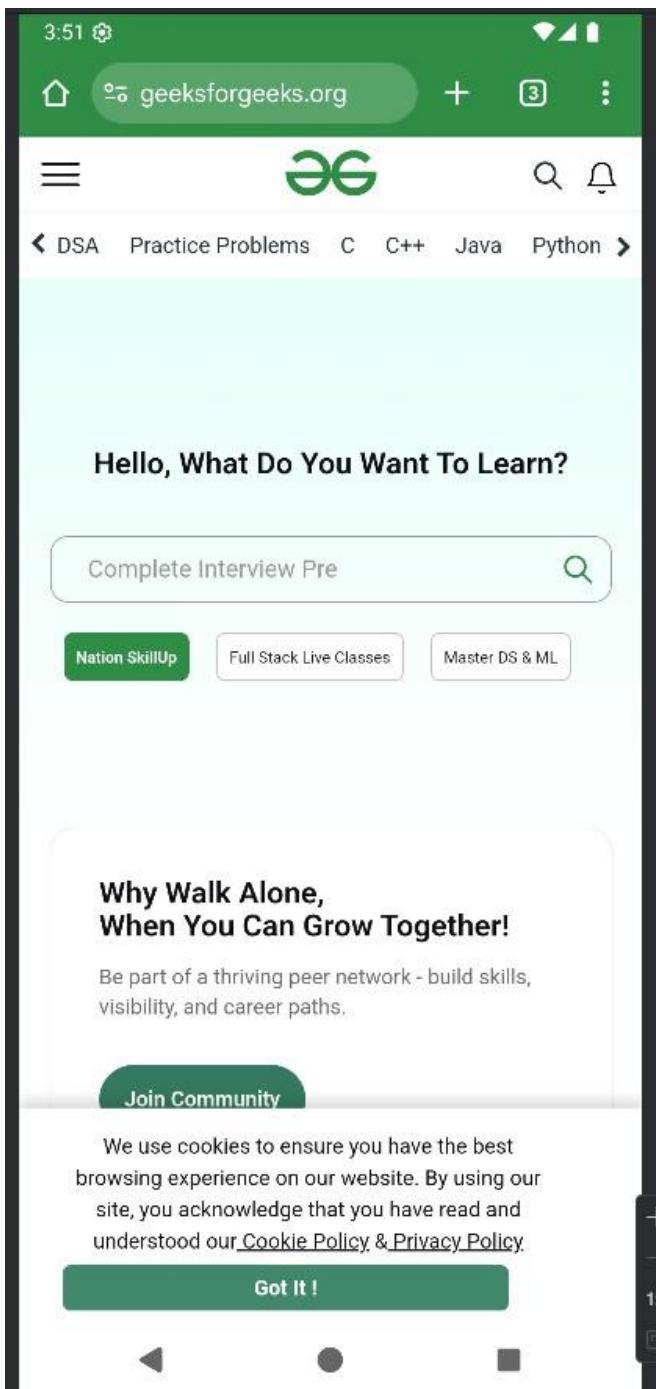
<Button
    android:id="@+id/button6"
    android:layout_width="120dp"
    android:layout_height="wrap_content"
    android:layout_row="1"
    android:layout_column="1"
    android:text="Gallery" />

<Button
    android:id="@+id/button2"
    android:layout_width="120dp"
    android:layout_height="wrap_content"
    android:layout_row="1"
    android:layout_column="0"
    android:text="Camera" />

<Button
    android:id="@+id/button5"
    android:layout_width="120dp"
    android:layout_height="wrap_content"
    android:layout_row="2"
    android:layout_column="0"
    android:text="Contact" />
</GridLayout>
```

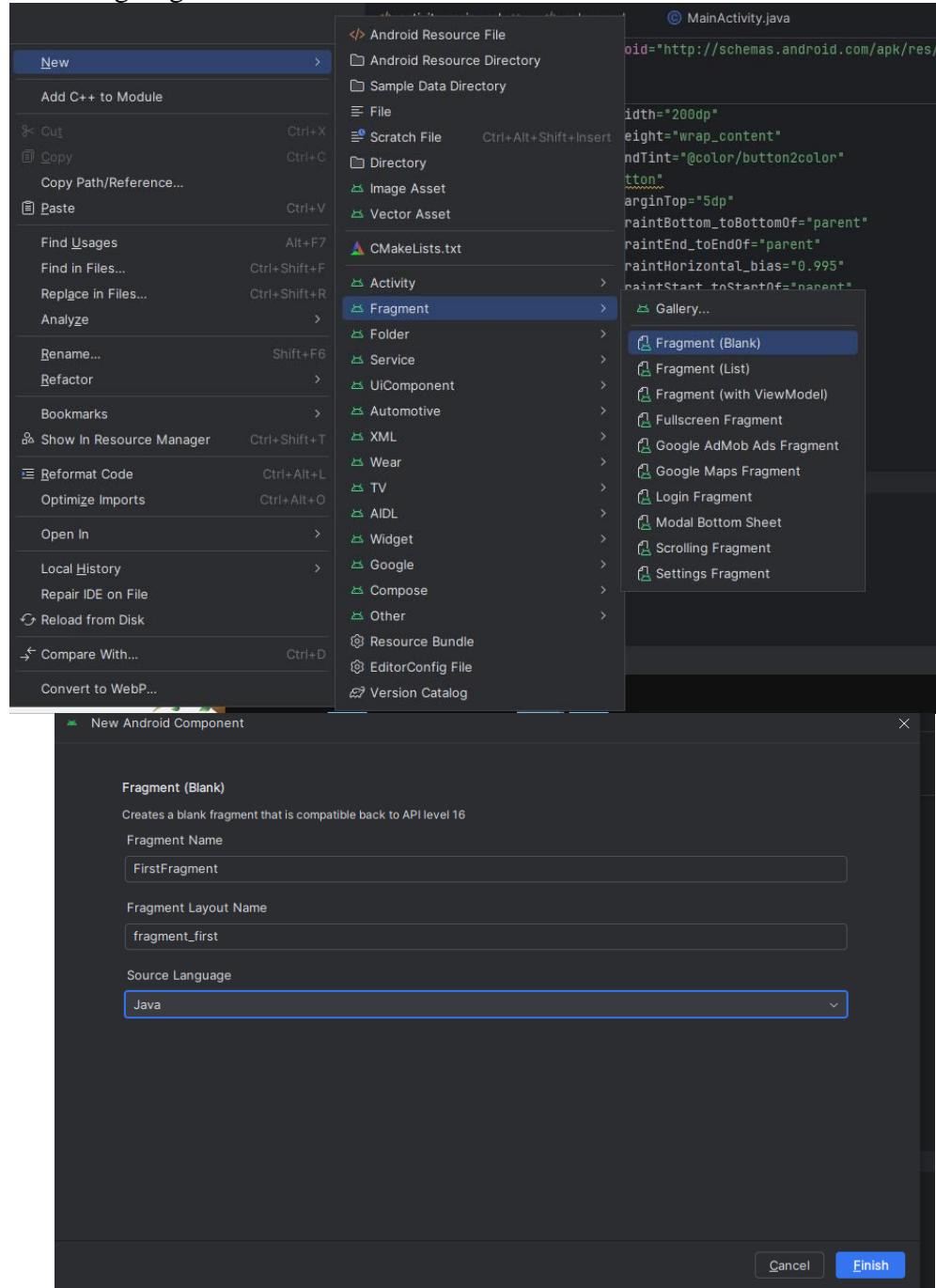
Output:





8. **Create an application which has two buttons. When the user clicks on the first button the first fragment will be displayed and when the user clicks on the second button the second fragment will be displayed.**

//Creating fragments



Code:

```
<?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:orientation="horizontal"
    android:layout_height="match_parent"
    tools:context=".MainActivity">
<LinearLayout
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:orientation="horizontal"
    android:background="@color/fragmenttwocolor">
    <Button
        android:id="@+id/button2"
        android:layout_width="200dp"
        android:layout_height="wrap_content"
        android:layout_marginTop="5dp"
        android:text="Button"
        app:layout_constraintBottom_toBottomOf="parent"
        app:layout_constraintEnd_toEndOf="parent"
        app:layout_constraintHorizontal_bias="0.0"
        android:backgroundTint="@color/button1color"
        app:layout_constraintStart_toStartOf="parent"
        app:layout_constraintTop_toTopOf="parent"
        app:layout_constraintVertical_bias="0.023" />
    <Button
        android:id="@+id/button"
        android:layout_width="200dp"
        android:layout_height="wrap_content"
        android:backgroundTint="@color/button2color"
        android:text="Button"
        android:layout_marginTop="5dp"
        app:layout_constraintBottom_toBottomOf="parent"
        app:layout_constraintEnd_toEndOf="parent"
        app:layout_constraintHorizontal_bias="0.995"
        app:layout_constraintStart_toStartOf="parent"
        app:layout_constraintTop_toTopOf="parent"
        app:layout_constraintVertical_bias="0.023" />
</LinearLayout>
</LinearLayout>
```

```
<?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:orientation="vertical"
    android:layout_height="match_parent"
    tools:context=".MainActivity">
<LinearLayout
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:orientation="horizontal"
    android:background="@color/fragmenttwocolor">
    <Button
        android:id="@+id/button2"
        android:layout_width="200dp"
        android:layout_height="wrap_content"
        android:layout_marginTop="5dp"
        android:text="Button"
        app:layout_constraintBottom_toBottomOf="parent"
        app:layout_constraintEnd_toEndOf="parent"
        app:layout_constraintHorizontal_bias="0.0"
        android:backgroundTint="@color/button1color"
        app:layout_constraintStart_toStartOf="parent"
        app:layout_constraintTop_toTopOf="parent"
        app:layout_constraintVertical_bias="0.023" />

    <Button
        android:id="@+id/button"
        android:layout_width="200dp"
        android:layout_height="wrap_content"
        android:backgroundTint="@color/button2color"
        android:text="Button"
        android:layout_marginTop="5dp"
        app:layout_constraintBottom_toBottomOf="parent"
        app:layout_constraintEnd_toEndOf="parent"
        app:layout_constraintHorizontal_bias="0.995"
        app:layout_constraintStart_toStartOf="parent"
        app:layout_constraintTop_toTopOf="parent"
        app:layout_constraintVertical_bias="0.023" />
</LinearLayout>
<FrameLayout
    android:id="@+id/Frame1"
    android:layout_width="match_parent"
    android:layout_height="match_parent">

    </FrameLayout>
</LinearLayout>
```

```
package com.example.myapplication;

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

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 androidx.fragment.app.FragmentTransaction;

public class MainActivity extends AppCompatActivity {
    Button bt1,bt2;

    @Override
    protected void onCreate(Bundle savedInstanceState) {

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

        bt1=findViewById(R.id.button);
        bt2=findViewById(R.id.button2);

        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;
    });

    bt1.setOnClickListener(new View.OnClickListener() {
        @Override
        public void onClick(View v) {
            FirstFragment firstFragment=new FirstFragment();
            FragmentTransaction
            transaction=getSupportFragmentManager().beginTransaction();
            transaction.replace(R.id.Frame1,firstFragment);
            transaction.commit();
        }
    });
    bt2.setOnClickListener(new View.OnClickListener() {
        @Override
        public void onClick(View v) {
            FragmentTwo fragmentTwo=new FragmentTwo();
            FragmentTransaction
            transaction=getSupportFragmentManager().beginTransaction();

```

```
        transaction.replace(R.id.Frame1,fragmentTwo);
        transaction.commit();

    }
});

}
}
```

```
<?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=".FragmentTwo">
<ImageView
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    android:scaleType="fitXY"
    android:src="@drawable/i5"/>
<TextView
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    android:text="Nature"
    android:textSize="30dp"
    android:textColor="@color/white"/>
</FrameLayout>
```

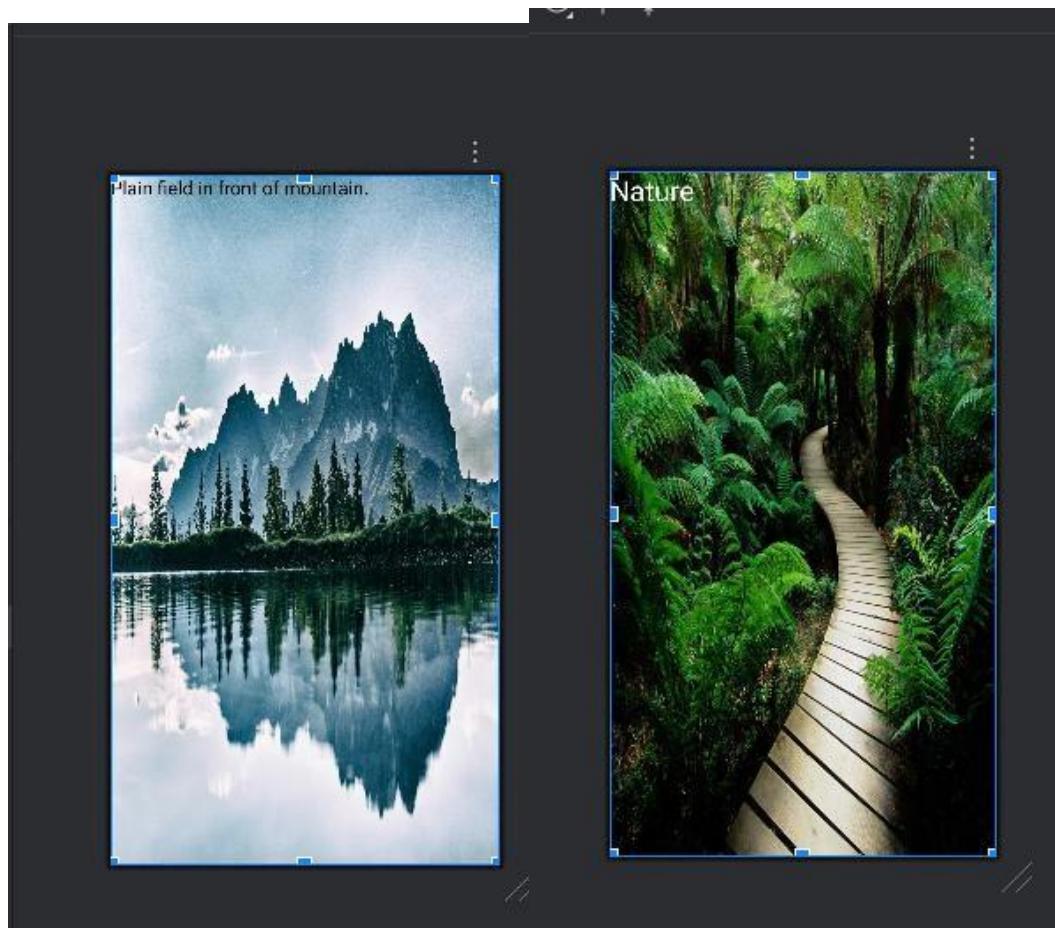
```
<?xml version="1.0" encoding="utf-8"?>
<FrameLayout xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:tools="http://schemas.android.com/tools"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    tools:context=".FirstFragment">
<ImageView
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    android:src="@drawable/i3"
    android:scaleType="fitXY"
/>
<TextView
    android:layout_width="match_parent"
    android:layout_height="match_parent"
```

android:text="Plain field in front of mountain."

```
        android:textSize="20dp"
        android:textColor="@color/black"
    />
</FrameLayout>
```

Output:





9. **Write an Android application with five check boxes to list the 5 subjects of your class and radio buttons to display gender. Display the selected subject name when you click any one of the checkboxes and gender in the alert dialog box.**

Activitymain.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:orientation="vertical"
    android:padding="20dp">

    <TextView
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="Select Your Subjects:"
        android:textSize="18sp"
        android:layout_marginTop="70dp"
        android:layout_marginBottom="10dp" />

    <CheckBox
        android:id="@+id/cbMath"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="Mathematics" />

    <CheckBox
        android:id="@+id/cbScience"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="Science" />

    <CheckBox
        android:id="@+id/cbEnglish"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="English" />

    <CheckBox
        android:id="@+id/cbHistory"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="History" />

    <CheckBox
        android:id="@+id/cbComputer"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="Computer Science" />

    <View
        android:layout_width="match_parent"
```

```
        android:layout_height="1dp"
        android:background="#CCCCCC"
        android:layout_marginVertical="15dp" />

<TextView
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:text="Select Gender:"
    android:textSize="18sp"
    android:layout_marginBottom="10dp" />

<RadioGroup
    android:id="@+id/radioGroupGender"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:orientation="horizontal">

    <RadioButton
        android:id="@+id/rbMale"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="Male" />

    <RadioButton
        android:id="@+id/rbFemale"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="Female"
        android:layout_marginStart="20dp" />
</RadioGroup>

</LinearLayout>
```

MainActivity.java

```
package com.example.myapplication;

import androidx.appcompat.app.AlertDialog;
import androidx.appcompat.app.AppCompatActivity;

import android.os.Bundle;
import android.widget.CheckBox;
import android.widget.RadioButton;
import android.widget.RadioGroup;
import android.widget.Toast;

public class MainActivity extends AppCompatActivity {
    CheckBox cbMath, cbScience, cbEnglish, cbHistory, cbComputer;
    RadioGroup radioGroupGender;

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

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

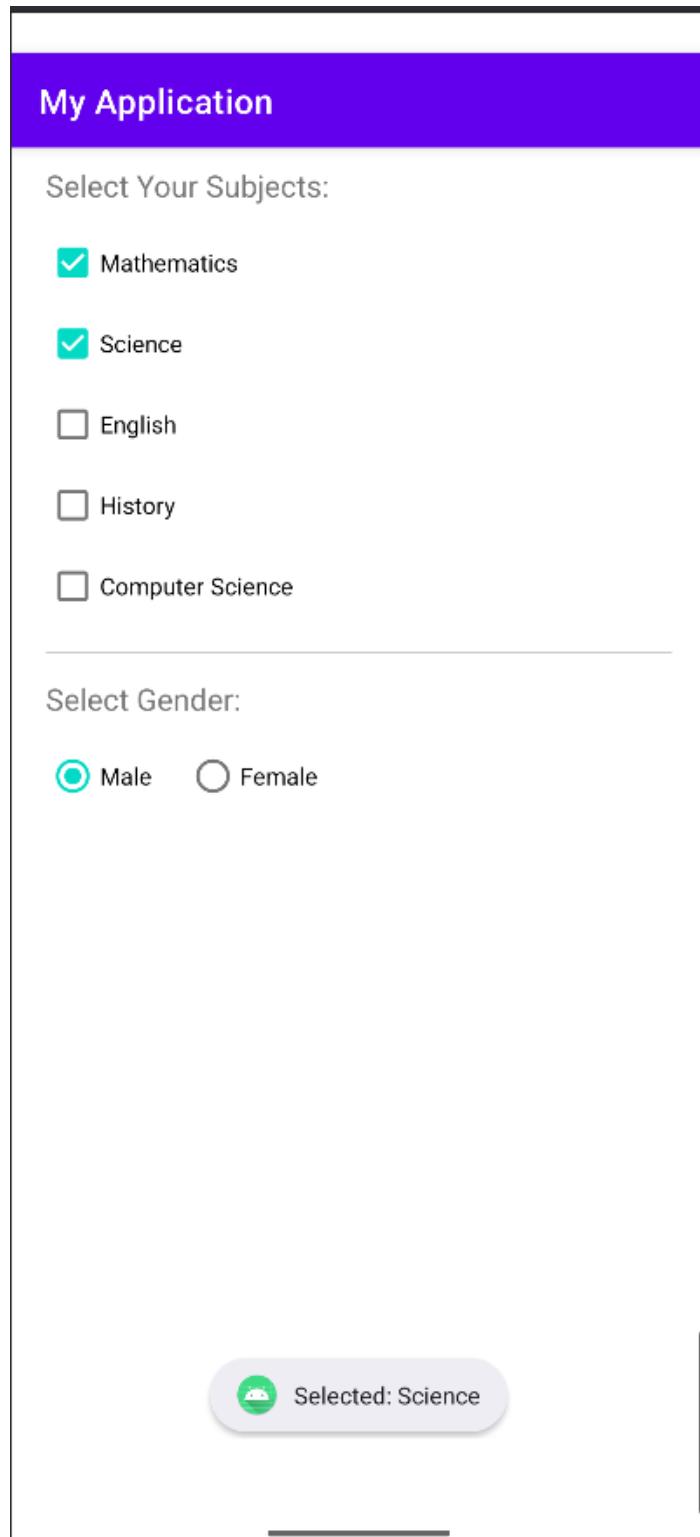
cbMath = findViewById(R.id.cbMath);
cbScience = findViewById(R.id.cbScience);
cbEnglish = findViewById(R.id.cbEnglish);
cbHistory = findViewById(R.id.cbHistory);
cbComputer = findViewById(R.id.cbComputer);
radioGroupGender = findViewById(R.id.radioGroupGender);

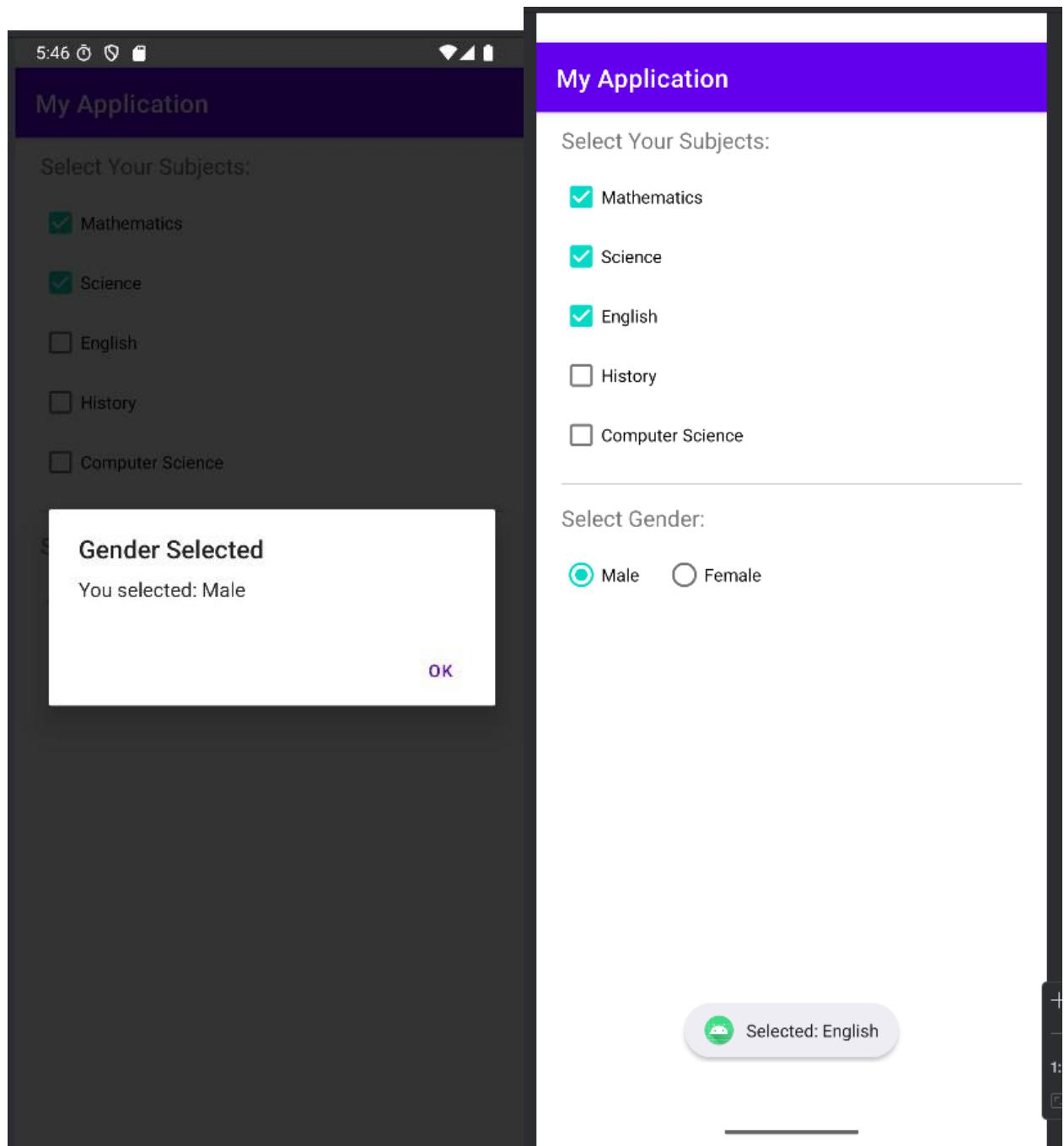
// --- Checkbox listeners ---
cbMath.setOnClickListener(v -> showSubject(cbMath));
cbScience.setOnClickListener(v -> showSubject(cbScience));
cbEnglish.setOnClickListener(v -> showSubject(cbEnglish));
cbHistory.setOnClickListener(v -> showSubject(cbHistory));
cbComputer.setOnClickListener(v -> showSubject(cbComputer));

// --- RadioGroup listener ---
radioGroupGender.setOnCheckedChangeListener((group, checkedId) -> {
    RadioButton selected = findViewById(checkedId);
    if (selected != null) {
        showGenderDialog(selected.getText().toString());
    }
});

private void showSubject(CheckBox cb) {
    if (cb.isChecked()) {
        Toast.makeText(this, "Selected: " + cb.getText(), Toast.LENGTH_SHORT).show();
    }
}

private void showGenderDialog(String gender) {
    new AlertDialog.Builder(this)
        .setTitle("Gender Selected")
        .setMessage("You selected: " + gender)
        .setPositiveButton("OK", null)
        .show();
}
```

Output:



10. Create a basic calculator to perform arithmetic operations with divide-by-zero validation. (using Alert box).

```
<?xml version="1.0" encoding="utf-8"?>
<LinearLayout
    xmlns:android="http://schemas.android.com/apk/res/android"
    android:orientation="vertical"
    android:padding="20dp"
    android:gravity="center"
    android:layout_width="match_parent"
    android:layout_height="match_parent">

    <EditText
        android:id="@+id/num1"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:hint="Enter first number"
        android:inputType="numberDecimal" />

    <EditText
        android:id="@+id/num2"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:hint="Enter second number"
        android:inputType="numberDecimal"
        android:layout_marginTop="10dp" />

    <TextView
        android:id="@+id/result"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:text="Result: "
        android:textSize="18sp"
        android:layout_marginTop="20dp" />

    <LinearLayout
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:orientation="horizontal"
        android:gravity="center"
        android:layout_marginTop="20dp">

        <Button
            android:id="@+id/btnAdd"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:text="+" />

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

```
        android:layout_height="wrap_content"
        android:text="-"
        android:layout_marginStart="10dp" />

    <Button
        android:id="@+id(btnMul"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="×"
        android:layout_marginStart="10dp" />

    <Button
        android:id="@+id	btnDiv"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="÷"
        android:layout_marginStart="10dp" />
    </LinearLayout>
</LinearLayout>
```

MainActivity.java

```
package com.example.myapplication;

import androidx.appcompat.app.AlertDialog;
import androidx.appcompat.app.AppCompatActivity;

import android.os.Bundle;
import android.view.View;
import android.widget.Button;
import android.widget.EditText;
import android.widget.TextView;
```

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

```
    EditText num1, num2;
    TextView result;
    Button btnAdd, btnSub, btnMul, btnDiv;

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

        num1 = findViewById(R.id.num1);
        num2 = findViewById(R.id.num2);
        result = findViewById(R.id.result);
```

```
btnAdd = findViewById(R.id.btnAdd);
btnSub = findViewById(R.id.btnSub);
btnMul = findViewById(R.id.btnMul);
btnDiv = findViewById(R.id.btnDiv);

btnAdd.setOnClickListener(v -> calculate('+'));
btnSub.setOnClickListener(v -> calculate('-'));
btnMul.setOnClickListener(v -> calculate('*'));
btnDiv.setOnClickListener(v -> calculate('/'));
}

private void calculate(char operator) {
    String n1 = num1.getText().toString();
    String n2 = num2.getText().toString();

    if (n1.isEmpty() || n2.isEmpty()) {
        showAlert("Please enter both numbers.");
        return;
    }

    double a = Double.parseDouble(n1);
    double b = Double.parseDouble(n2);
    double res = 0;

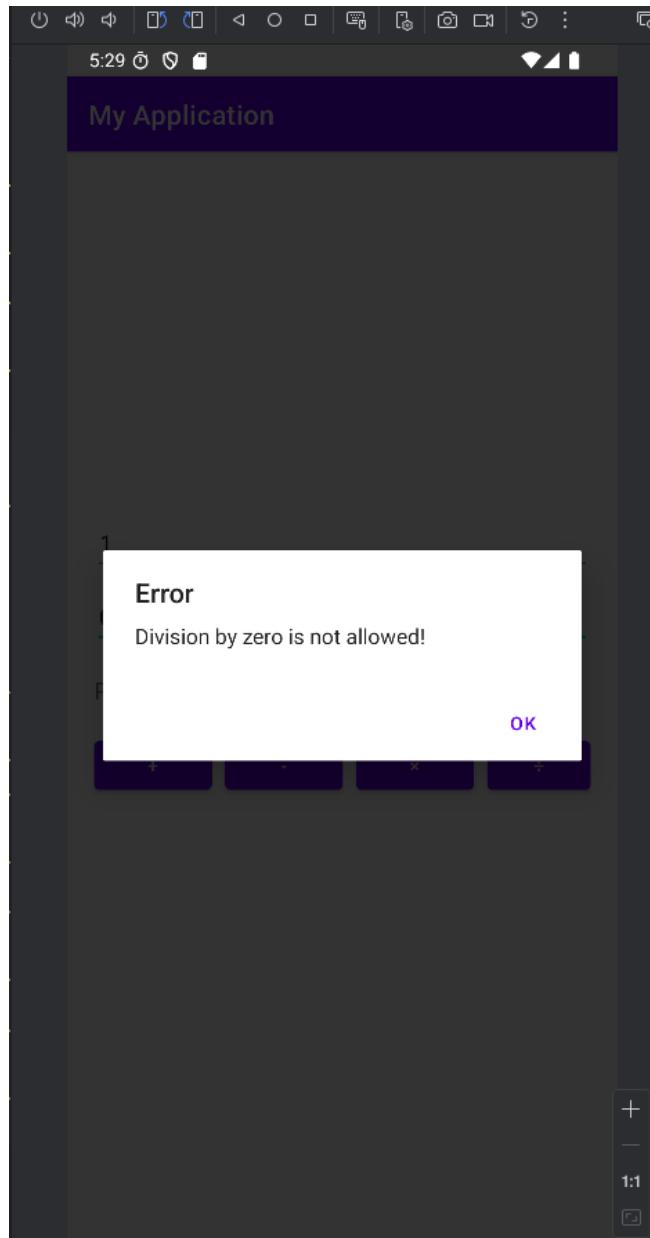
    switch (operator) {
        case '+':
            res = a + b;
            break;
        case '-':
            res = a - b;
            break;
        case '*':
            res = a * b;
            break;
        case '/':
            if (b == 0) {
                showAlert("Division by zero is not allowed!");
                return;
            } else {
                res = a / b;
            }
            break;
    }

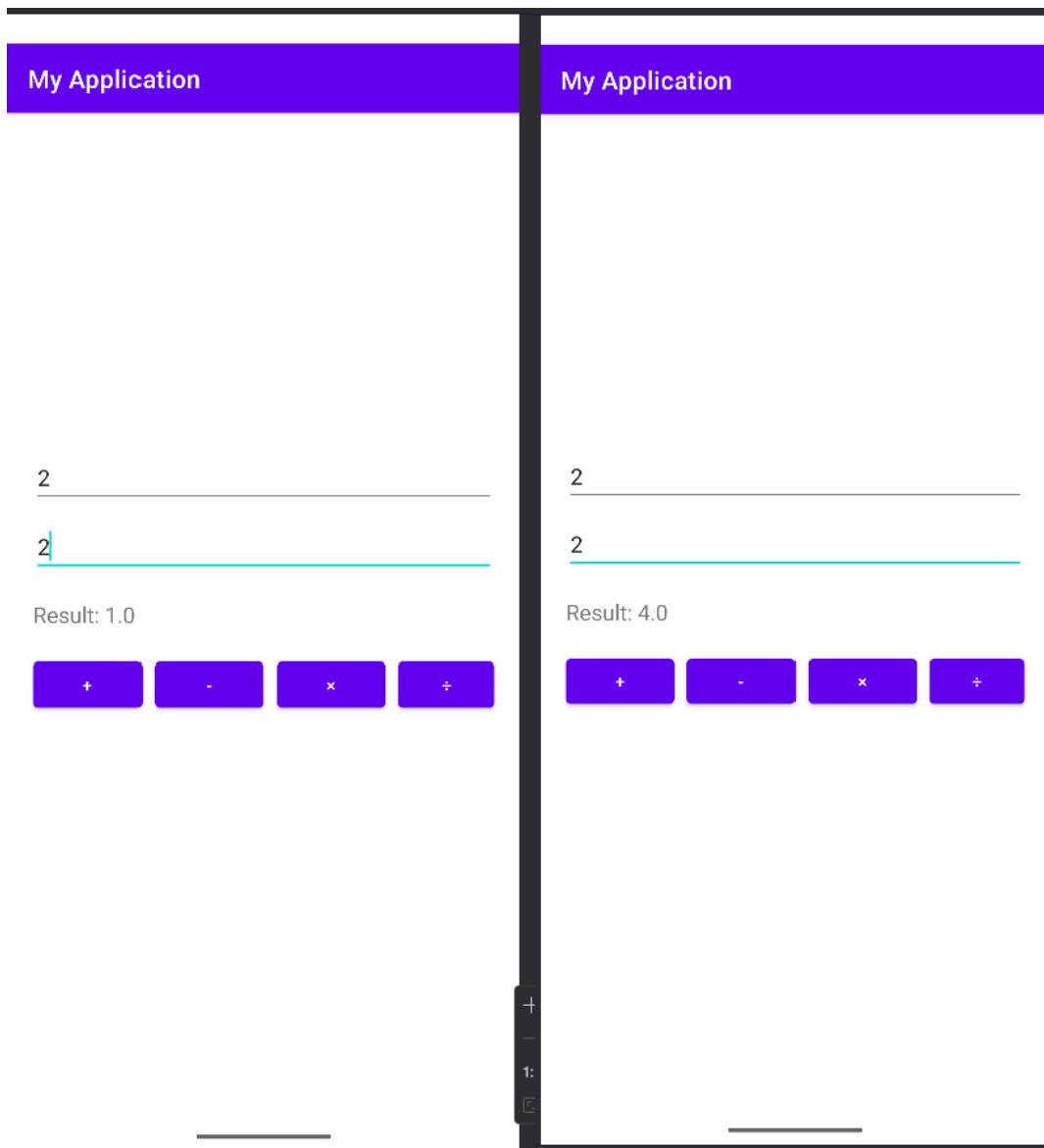
    result.setText("Result: " + res);
}

private void showAlert(String message) {
    AlertDialog.Builder builder = new AlertDialog.Builder(this);
    builder.setTitle("Error");
    builder.setMessage(message);
```

```
        builder.setPositiveButton("OK", null);
        builder.show();
    }
}
```

Output:





11. Create an Android application to demonstrate List View using an array adapter.

Code:

```
Activity_main.xml
<?xml version="1.0" encoding="utf-8"?>
<LinearLayout xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:app="http://schemas.android.com/apk/res-auto"
    xmlns:tools="http://schemas.android.com/tools"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    tools:context=".MainActivity"
    android:orientation="vertical"
    android:padding="16dp">
    <ListView
        android:id="@+id/listView"
        android:layout_width="match_parent"
        android:layout_height="wrap_content" />
</LinearLayout>
```

MainActivity.java

```
package com.example.myapplication;

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 androidx.appcompat.app.AppCompatActivity;

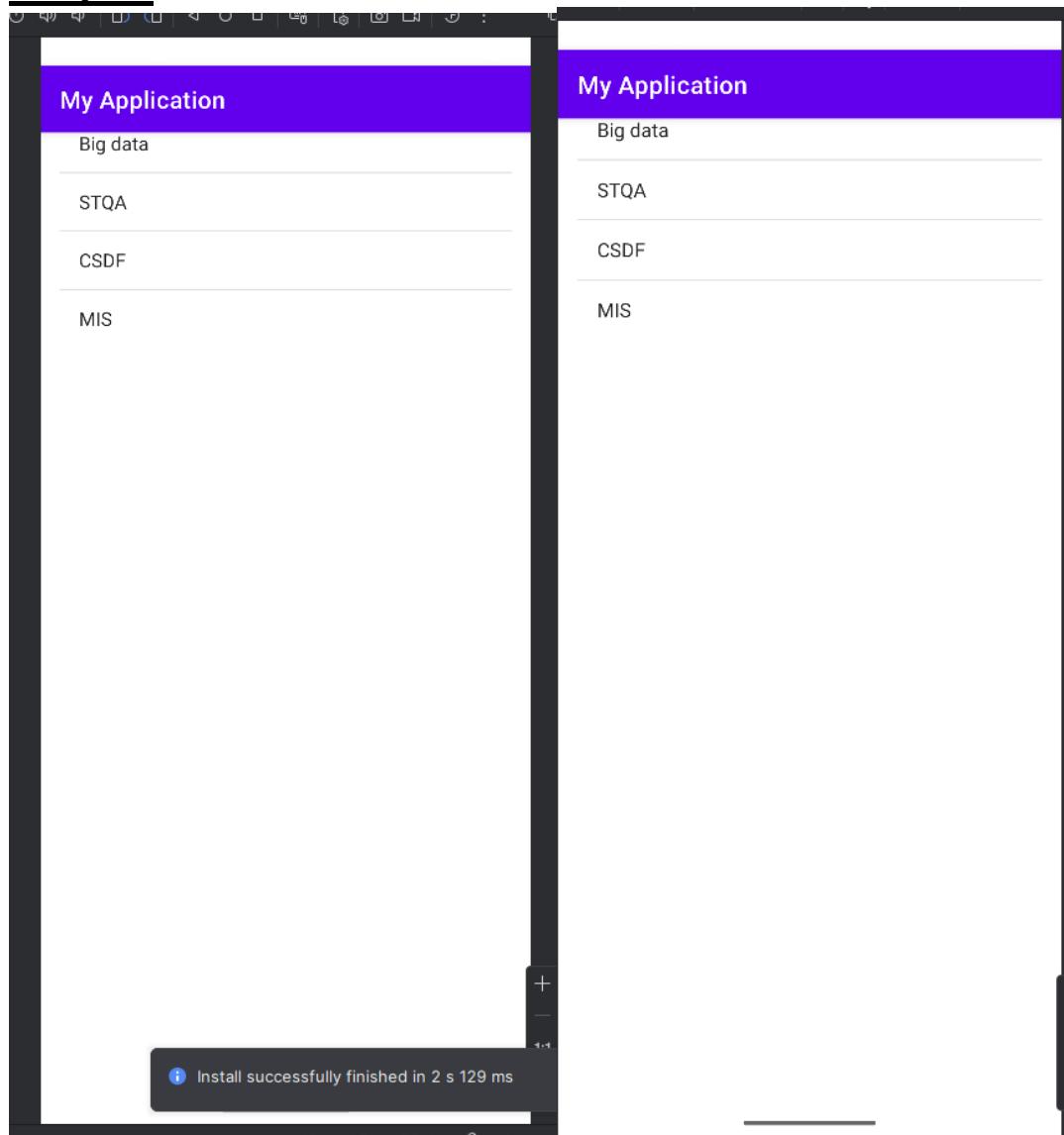
import android.os.Bundle;
import android.widget.ArrayAdapter;
import android.widget.ListView;

public class MainActivity extends AppCompatActivity {

    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_main);
        ListView listView = findViewById(R.id.listView);
        String[] sub = {"Mobile computing", "Big data", "STQA", "CSDF", "MIS"};
        ArrayAdapter<String> adapter = new ArrayAdapter<>(this,
            android.R.layout.simple_list_item_1, sub);
```

```
        listView.setAdapter(adapter);  
    }  
}
```

Output:



12. Create a mobile application for a currency converter. Use a spinner for selecting the currency.

```
package com.example.myapplication;

import android.os.Bundle;
import android.view.View;
import android.widget.ArrayAdapter;
import android.widget.Button;
import android.widget.EditText;
import android.widget.ImageButton;
import android.widget.Spinner;
import android.widget.TextView;

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 {

    Spinner sp;
    TextView tx;
    ImageButton ib;
    EditText et;
    Spinner sp1;
    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_main);
        ib=findViewById(R.id.imageButton);
        sp=findViewById(R.id.spinner);
        sp1=findViewById(R.id.spinner2);
        tx=findViewById(R.id.textView);
        et=findViewById(R.id.editTextText);
        //                                         ArrayAdapter<CharSequence>
        adapter=ArrayAdapter.createFromResource(this,R.array.currencies,
        android.R.layout.simple_spinner_item);
        //     adapter.setDropDownViewResource(android.R.layout.simple_spinner_dropdown_item);
        //     sp.setAdapter(adapter);
        //     sp1.setAdapter(adapter);

        ib.setOnClickListener(new View.OnClickListener() {
            @Override
            public void onClick(View v) {
                String fromcurr=sp.getSelectedItem().toString();
                String tocurr=sp1.getSelectedItem().toString();
                double amount=Double.parseDouble(et.getText().toString());
            }
        });
    }
}
```

```
        double conversionrate=getConversionRate(fromcurr,tocurr);
        double result=amount*conversionrate;
        tx.setText(String.valueOf(result));

    }

});

}

private double getConversionRate(String fromcurr,String tocurr){
    if(fromcurr.equals("USD")&&tocurr.equals("EUR")){
        return 0.86;

    } else if (fromcurr.equals("EUR")&&tocurr.equals("USB")) {
        return 1.16;
    }
    else if (fromcurr.equals("USD")&&tocurr.equals("INR")) {
        return 87.67;
    }
    else if (fromcurr.equals("INR")&&tocurr.equals("USD")) {
        return 0.011;
    }
    else if (fromcurr.equals("EUR")&&tocurr.equals("INR")) {
        return 101.79;
    }
    else if (fromcurr.equals("INR")&&tocurr.equals("EUR")) {
        return 0.0098;
    }
    else if (fromcurr.equals("GBP")&&tocurr.equals("INR")) {
        return 117.99;
    }
    else if (fromcurr.equals("INR")&&tocurr.equals("GBP")) {
        return 0.0085 ;
    }

    else if (fromcurr.equals("JPY")&&tocurr.equals("INR")) {
        return 0.59 ;
    }
    else if (fromcurr.equals("JPY")&&tocurr.equals("GBP")) {
        return 0.0050;
    }else if (fromcurr.equals("JPY")&&tocurr.equals("USD")) {
        return 0.0067 ;
    }
    else if (fromcurr.equals("USD")&&tocurr.equals("JPY")) {
        return 148.40 ;
    }
    else if (fromcurr.equals("INR")&&tocurr.equals("JPY")) {
        return 1.69 ;
    }
    else if (fromcurr.equals("GBP")&&tocurr.equals("JPY")) {
        return 199.74 ;
    }

}
```

```
    return 0.0;  
  
}  
  
}
```

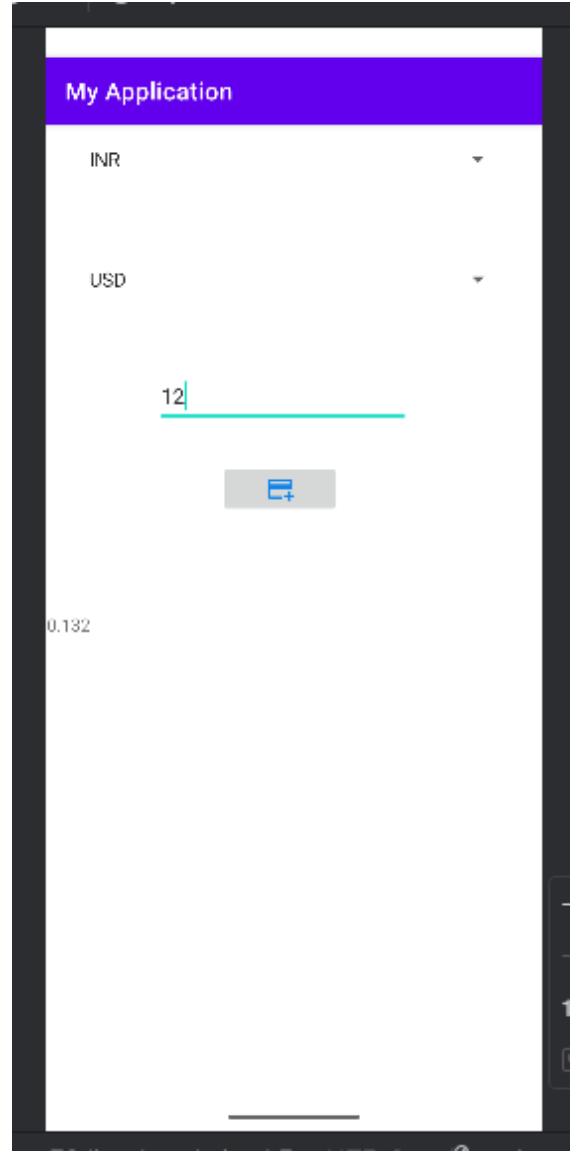
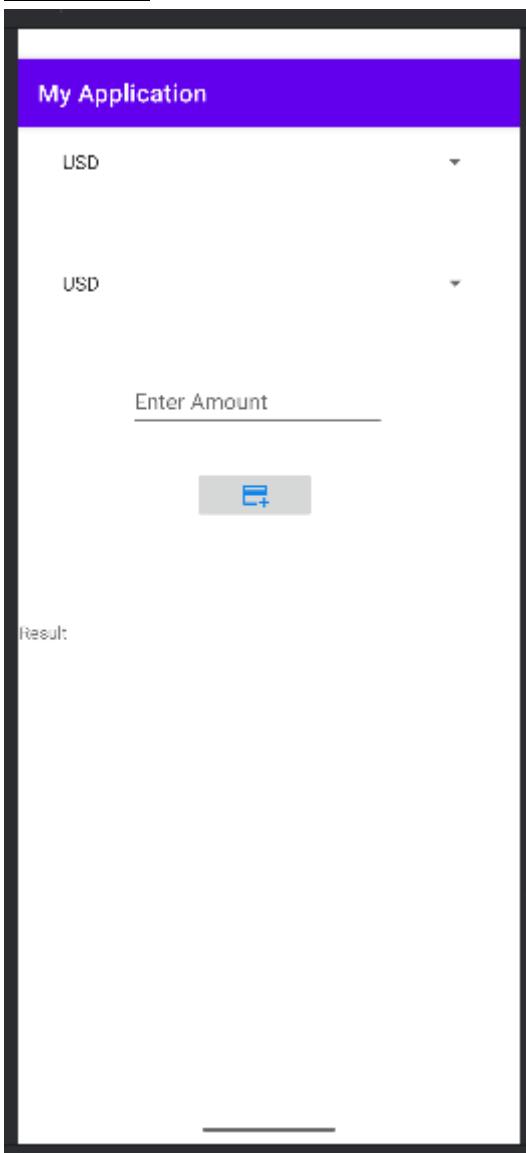
Xml file:

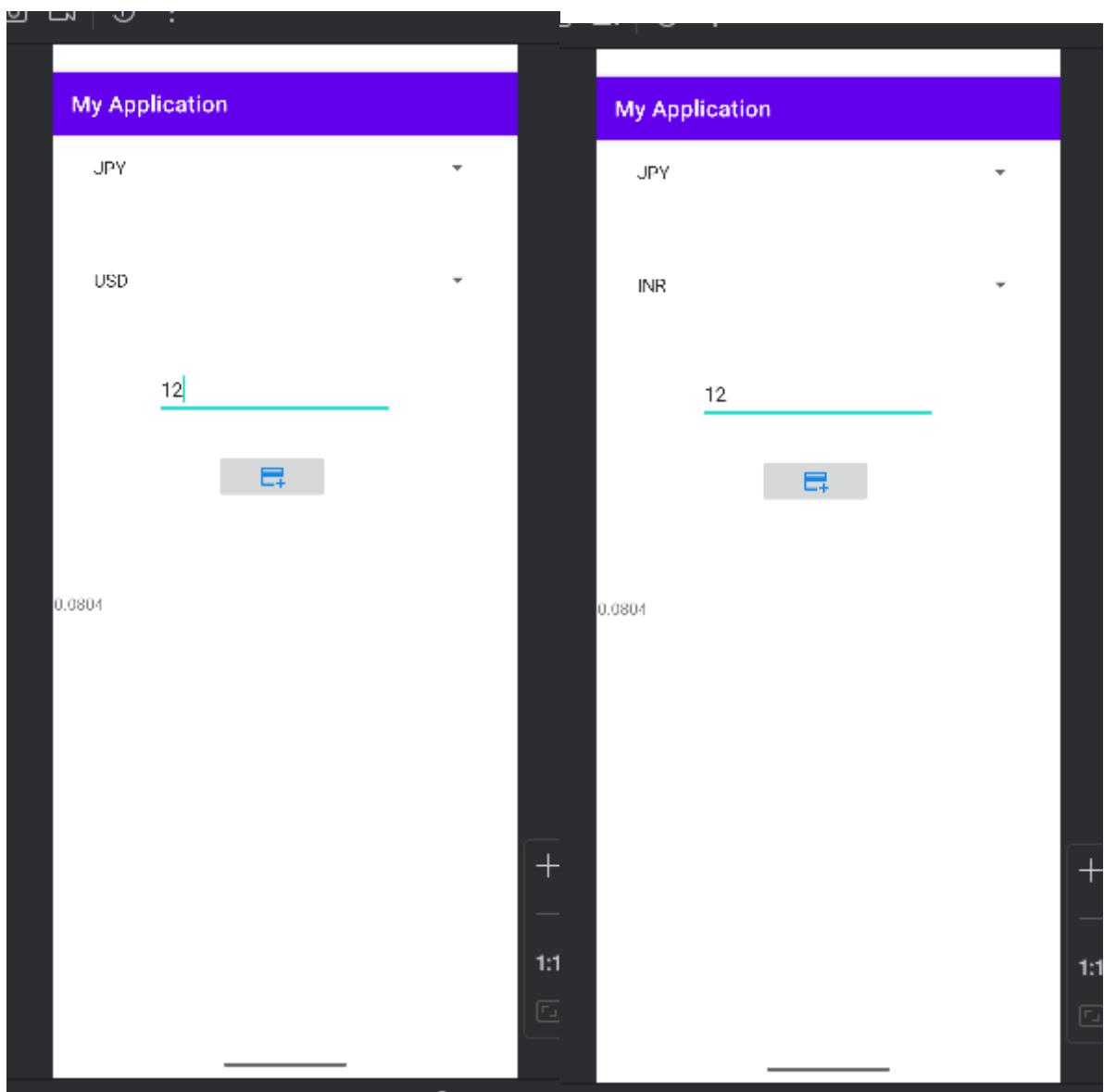
```
<?xml version="1.0" encoding="utf-8"?>  
<androidx.constraintlayout.widget.ConstraintLayout  
    xmlns:android="http://schemas.android.com/apk/res/android"  
    xmlns:app="http://schemas.android.com/apk/res-auto"  
    xmlns:tools="http://schemas.android.com/tools"  
    android:id="@+id/main"  
    android:layout_width="match_parent"  
    android:layout_height="match_parent"  
    tools:context=".MainActivity">  
  
<Spinner  
    android:id="@+id/spinner2"  
    android:layout_width="354dp"  
    android:layout_height="48dp"  
    android:entries="@array/currencies"  
    android:minHeight="48dp"  
    app:layout_constraintBottom_toBottomOf="parent"  
    app:layout_constraintEnd_toEndOf="parent"  
    app:layout_constraintHorizontal_bias="0.491"  
    app:layout_constraintStart_toStartOf="parent"  
    app:layout_constraintTop_toTopOf="parent"  
    app:layout_constraintVertical_bias="0.213" />  
  
<Spinner  
    android:id="@+id/spinner"  
    android:layout_width="354dp"  
    android:layout_height="48dp"  
    android:entries="@array/currencies"  
    app:layout_constraintBottom_toBottomOf="parent"  
    app:layout_constraintEnd_toEndOf="parent"  
    app:layout_constraintHorizontal_bias="0.491"  
    app:layout_constraintStart_toStartOf="parent"  
    app:layout_constraintTop_toTopOf="parent"  
    app:layout_constraintVertical_bias="0.098" />  
  
<EditText  
    android:id="@+id/editTextText"  
    android:layout_width="wrap_content"
```

```
        android:layout_height="wrap_content"
        android:ems="10"
        android:hint="Enter Amount"
        android:inputType="number"
        android:minHeight="48dp"
        app:layout_constraintBottom_toBottomOf="parent"
        app:layout_constraintEnd_toEndOf="parent"
        app:layout_constraintHorizontal_bias="0.452"
        app:layout_constraintStart_toStartOf="parent"
        app:layout_constraintTop_toTopOf="parent"
        app:layout_constraintVertical_bias="0.325" />
```

```
<ImageButton
    android:id="@+id/imageButton"
    android:layout_width="100dp"
    android:layout_height="wrap_content"
    app:layout_constraintBottom_toBottomOf="parent"
    app:layout_constraintEnd_toEndOf="parent"
    app:layout_constraintHorizontal_bias="0.463"
    app:layout_constraintStart_toStartOf="parent"
    app:layout_constraintTop_toTopOf="parent"
    app:layout_constraintVertical_bias="0.414"
    app:srcCompat="@drawable/outline_add_card_24"
    tools:ignore="SpeakableTextPresentCheck,TouchTargetSizeCheck" />
```

```
<TextView
    android:id="@+id/textView"
    android:layout_width="400dp"
    android:layout_height="wrap_content"
    android:text="Result"
    app:layout_constraintBottom_toBottomOf="parent"
    app:layout_constraintEnd_toEndOf="parent"
    app:layout_constraintHorizontal_bias="0.097"
    app:layout_constraintStart_toStartOf="parent"
    app:layout_constraintTop_toTopOf="parent"
    app:layout_constraintVertical_bias="0.542" />
</androidx.constraintlayout.widget.ConstraintLayout>
```

Output:



13. Write an application to increase font size using seekbar.

```
<?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/editTextText"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:ems="10"
        android:inputType="text"
        android:text="Name"
        app:layout_constraintBottom_toBottomOf="parent"
        app:layout_constraintEnd_toEndOf="parent"
        app:layout_constraintStart_toStartOf="parent"
        app:layout_constraintTop_toTopOf="parent"
        app:layout_constraintVertical_bias="0.023"
        app:layout_editor_absoluteX="123dp"
        app:layout_editor_absoluteY="111dp"/>

    <SeekBar
        android:id="@+id/seekBar3"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:max="100"
        android:min="0"
        app:layout_constraintBottom_toBottomOf="parent"
        app:layout_constraintEnd_toEndOf="parent"
        app:layout_constraintStart_toStartOf="parent"
        app:layout_constraintTop_toTopOf="parent"
        app:layout_constraintVertical_bias="0.155" />
</androidx.constraintlayout.widget.ConstraintLayout>
```

.JAVA File:

```
package com.example.myapplication;

import android.os.Bundle;
import android.widget.SeekBar;
import android.widget.TextView;
```

```
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 {
    TextView tv;
    SeekBar sb;
    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);

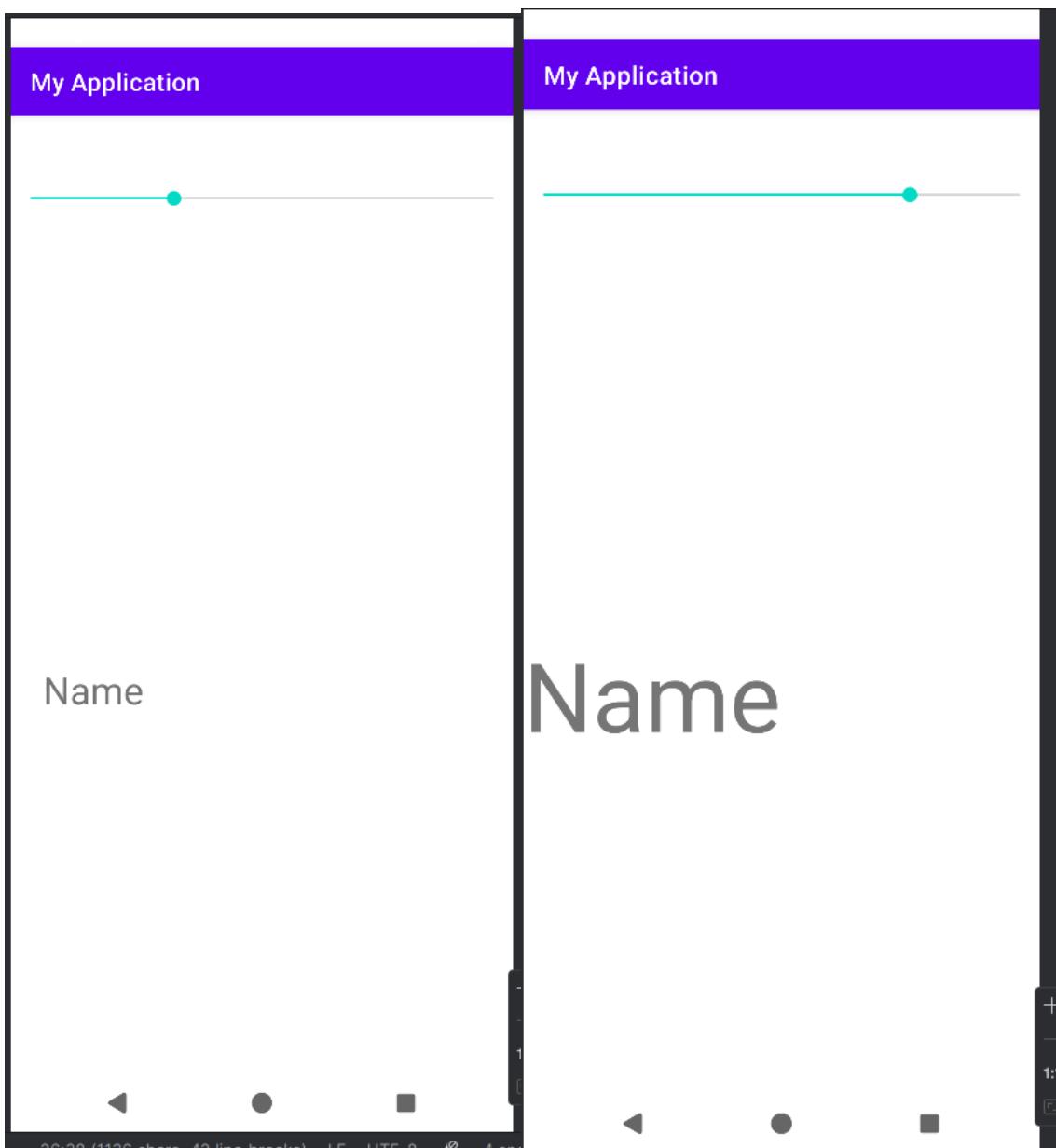
        setContentView(R.layout.activity_main);
        tv=findViewById(R.id.editTextText);
        sb=findViewById(R.id.seekBar3);
        sb.setOnSeekBarChangeListener(new SeekBar.OnSeekBarChangeListener() {
            @Override
            public void onProgressChanged(SeekBar seekBar, int progress, boolean fromUser) {
                tv.setTextSize(progress);
            }

            @Override
            public void onStartTrackingTouch(SeekBar seekBar) {
            }

            @Override
            public void onStopTrackingTouch(SeekBar seekBar) {
            }
        });

    }
}
```

Output:



14. Create an Android application to demonstrate progressbar.

MainActivity.java

```
package com.example.myapplication;

import androidx.appcompat.app.AppCompatActivity;
import android.os.Bundle;
import android.os.Handler;
import android.view.View;
import android.widget.Button;
import android.widget.ProgressBar;

public class MainActivity extends AppCompatActivity {

    ProgressBar progressBar;
    Button btnStart;
    Handler handler = new Handler();
    int progressStatus = 0;

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

        progressBar = findViewById(R.id.progressBar);
        btnStart = findViewById(R.id.btnStart);

        btnStart.setOnClickListener(v -> {
            progressBar.setVisibility(View.VISIBLE);
            progressStatus = 0;

            new Thread(() -> {
                while (progressStatus < 100) {
                    progressStatus += 10;
                    handler.post(() -> progressBar.setProgress(progressStatus));
                    try {
                        Thread.sleep(500); // delay for effect
                    } catch (InterruptedException e) {
                        e.printStackTrace();
                    }
                }
            }).start();
        });
    }

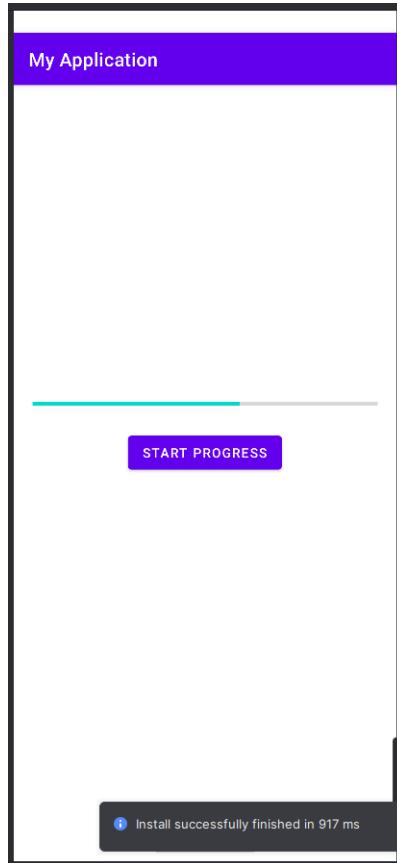
    <?xml version="1.0" encoding="utf-8"?>
<LinearLayout xmlns:android="http://schemas.android.com/apk/res/android"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    android:orientation="vertical"
```

```
    android:gravity="center"
    android:padding="20dp">>

<ProgressBar
    android:id="@+id/progressBar"
    style="?android:attr/progressBarStyleHorizontal"
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:max="100"
    android:progress="0"
    android:visibility="gone"/>

<Button
    android:id="@+id	btnStart"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:text="Start Progress"
    android:layout_marginTop="20dp"/>
</LinearLayout>
```

Output:



Module 2 Database Connectivity:**Practical 1: Create an Android application to read and write content in internal storage.**

```
<?xml version="1.0" encoding="utf-8"?>
<LinearLayout
    xmlns:android="http://schemas.android.com/apk/res/android"
    android:orientation="vertical"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    android:padding="20dp"
    android:gravity="center_horizontal">

    <EditText
        android:id="@+id/editText"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:hint="Enter text here"
        android:minHeight="80dp"
        android:gravity="top"
        android:background="@android:drawable/edit_text" />

    <Button
        android:id="@+id	btnSave"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="Save to Internal Storage"
        android:layout_marginTop="20dp"/>
```

```
<Button  
    android:id="@+id	btnRead"  
    android:layout_width="wrap_content"  
    android:layout_height="wrap_content"  
    android:text="Read from Internal Storage"  
    android:layout_marginTop="10dp"/>  
  
<TextView  
    android:id="@+id/textView"  
    android:layout_width="match_parent"  
    android:layout_height="wrap_content"  
    android:text="File content will appear here"  
    android:textSize="16sp"  
    android:layout_marginTop="20dp"  
    android:gravity="center"/>  
</LinearLayout>
```

```
package com.example.ksr;  
  
import androidx.appcompat.app.AppCompatActivity;  
import android.os.Bundle;  
import android.view.View;  
import android.widget.*;  
import java.io.*;  
  
public class MainActivity2 extends AppCompatActivity {  
  
    EditText editText;  
    Button btnSave, btnRead;
```

```
TextView textView;  
  
String fileName = "myFile.txt";  
  
@Override  
protected void onCreate(Bundle savedInstanceState) {  
    super.onCreate(savedInstanceState);  
    setContentView(R.layout.activity_main);  
  
    editText = findViewById(R.id.editText);  
    btnSave = findViewById(R.id.btnSave);  
    btnRead = findViewById(R.id.btnRead);  
    textView = findViewById(R.id.textView);  
  
    // SAVE BUTTON  
    btnSave.setOnClickListener(new View.OnClickListener() {  
        @Override  
        public void onClick(View v) {  
            String data = editText.getText().toString();  
            if (data.isEmpty()) {  
                Toast.makeText(MainActivity2.this, "Please enter text",  
                        Toast.LENGTH_SHORT).show();  
            } else {  
                writeFile(data);  
            }  
        }  
    });  
  
    // READ BUTTON  
    btnRead.setOnClickListener(new View.OnClickListener() {  
        @Override
```

```
public void onClick(View v) {
    readFromFile();
}

});

}

// Function to Write Data

private void writeToFile(String data) {
    try {
        FileOutputStream fos = openFileOutput(fileName, MODE_PRIVATE);
        fos.write(data.getBytes());
        fos.close();
        Toast.makeText(this, "Data saved successfully!", Toast.LENGTH_SHORT).show();
        editText.setText("");
    } catch (IOException e) {
        e.printStackTrace();
        Toast.makeText(this, "Error saving file!", Toast.LENGTH_SHORT).show();
    }
}

// Function to Read Data

private void readFromFile() {
    try {
        FileInputStream fis = openFileInput(fileName);
        BufferedReader reader = new BufferedReader(new InputStreamReader(fis));
        StringBuilder stringBuilder = new StringBuilder();
        String line;

        while ((line = reader.readLine()) != null) {
            stringBuilder.append(line).append("\n");
        }
    }
}
```

```
        }
```

```
    reader.close();
```

```
    fis.close();
```

```
    textView.setText(stringBuilder.toString());
```

```
} catch (IOException e) {
```

```
    e.printStackTrace();
```

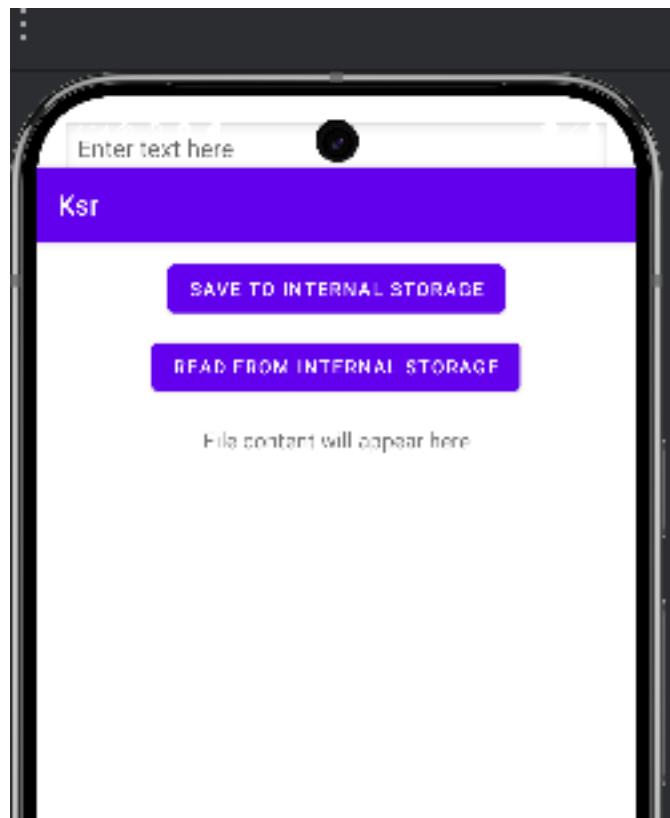
```
    Toast.makeText(this, "Error reading file!", Toast.LENGTH_SHORT).show();
```

```
}
```

```
}
```

```
}
```

Output:



Practical 2: Create an Android application to read and write content in external storage.

```
package com.example.myapplication;
import androidx.appcompat.app.AppCompatActivity;
import android.os.Bundle;
import android.os.Environment;
import android.view.View;
import android.widget.Button;
import android.widget.EditText;
import android.widget.TextView;
import android.widget.Toast;
import java.io.BufferedReader;
import java.io.File;
import java.io.InputStream;
import java.io.OutputStream;
import java.io.IOException;
import java.io.InputStreamReader;

public class MainActivity extends AppCompatActivity {

    private EditText inputText;
    private Button btnWrite, btnLoad;
    private TextView dir;
    private String filename = "hello.txt";
    private String filepath = "MyFileStorage";
    private File extFile;
    private String data = "";

    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_main);
        inputText = findViewById(R.id.input_text);
        btnWrite = findViewById(R.id.btn_write);
        btnLoad = findViewById(R.id.btn_load);
        dir = findViewById(R.id.dir);
        if (!isExternalStorageAvailable() || isExternalStorageReadOnly()) {
            btnWrite.setEnabled(false);
        } else {
            extFile = new File(getExternalFilesDir(filepath), filename);
        }
        getDir();
        btnWrite.setOnClickListener(new View.OnClickListener() {
            @Override
            public void onClick(View view) {
                data = inputText.getText().toString();
                try {
```

```

        FileOutputStream fos = new FileOutputStream(extFile);
        fos.write(data.getBytes());
        // fos.write("Hello".getBytes());
        inputText.getText().clear();
        Toast.makeText(getApplicationContext(), " filename + " saved to external
storage...", Toast.LENGTH_SHORT).show();
        fos.close();
    }
    catch (IOException ex) {
        ex.printStackTrace();
    }
}
});

btnLoad.setOnClickListener(new View.OnClickListener() {
    @Override
    public void onClick(View view) {
        try {
            FileInputStream fis = new FileInputStream(extFile);
            InputStreamReader isr = new InputStreamReader(fis);
            BufferedReader br = new BufferedReader(isr);
            StringBuilder data = new StringBuilder();
            String line;
            while ((line = br.readLine()) != null) {
                data.append("\n").append(line);
            }
            inputText.setText(data);
            Toast.makeText(getApplicationContext(), "Data Retrieved from External File
Successfully...", Toast.LENGTH_SHORT).show();
            fis.close();
        }
        catch (IOException ex) {
            ex.printStackTrace();
        }
    }
});
}

private static boolean isExternalStorageAvailable() {
    String extStorageState = Environment.getExternalStorageState();
    return Environment.MEDIA_MOUNTED.equals(extStorageState);
}

private static boolean isExternalStorageReadOnly() {
    String extStorageState = Environment.getExternalStorageState();
    return Environment.MEDIA_MOUNTED_READ_ONLY.equals(extStorageState);
}

private void getDir()
{
    StringBuilder builder=new StringBuilder();
    builder.append("External File Directories:");
    builder.append(getExternalFilesDir(filepath).getAbsolutePath()).append("\n");
}

```

```
        dir.setText(builder.toString());
    }
}
```

AndroidManifest.xml

```
package com.example.myapplication;
import androidx.appcompat.app.AppCompatActivity;
import android.os.Bundle;
import android.os.Environment;
import android.view.View;
import android.widget.Button;
import android.widget.EditText;
import android.widget.TextView;
import android.widget.Toast;
import java.io.BufferedReader;
import java.io.File;
import java.io.InputStream;
import java.io.OutputStream;
import java.io.IOException;
import java.io.InputStreamReader;

public class MainActivity extends AppCompatActivity {

    private EditText inputText;
    private Button btnWrite, btnLoad;
    private TextView dir;
    private String filename = "hello.txt";
    private String filepath = "MyFileStorage";
    private File extFile;
    private String data = "";

    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_main);
        inputText = findViewById(R.id.input_text);
        btnWrite = findViewById(R.id.btn_write);
        btnLoad = findViewById(R.id.btn_load);
        dir = findViewById(R.id.dir);
        if (!isExternalStorageAvailable() || isExternalStorageReadOnly()) {
            btnWrite.setEnabled(false);
        } else {
            extFile = new File(getExternalFilesDir(filepath), filename);
        }
        getDir();
        btnWrite.setOnClickListener(new View.OnClickListener() {
            @Override
            public void onClick(View view) {
```

```
data = inputText.getText().toString();
try {
    FileOutputStream fos = new FileOutputStream(extFile);
    fos.write(data.getBytes());
    // fos.write("Hello".getBytes());
    inputText.getText().clear();
    Toast.makeText(getApplicationContext(), " filename + " saved to external
storage...", Toast.LENGTH_SHORT).show();
    fos.close();
}
catch (IOException ex) {
    ex.printStackTrace();
}
});
btnLoad.setOnClickListener(new View.OnClickListener() {
    @Override
    public void onClick(View view) {
        try {
            FileInputStream fis = new FileInputStream(extFile);
            InputStreamReader isr = new InputStreamReader(fis);
            BufferedReader br = new BufferedReader(isr);
            StringBuilder data = new StringBuilder();
            String line;
            while ((line = br.readLine()) != null) {
                data.append("\n").append(line);
            }
            inputText.setText(data);
            Toast.makeText(getApplicationContext(), "Data Retrieved from External File
Successfully...", Toast.LENGTH_SHORT).show();
            fis.close();
        }
        catch (IOException ex) {
            ex.printStackTrace();
        }
    }
});
}

private static boolean isExternalStorageAvailable() {
    String extStorageState = Environment.getExternalStorageState();
    return Environment.MEDIA_MOUNTED.equals(extStorageState);
}

private static boolean isExternalStorageReadOnly() {
    String extStorageState = Environment.getExternalStorageState();
    return Environment.MEDIA_MOUNTED_READ_ONLY.equals(extStorageState);
}

private void getDir()
{
    StringBuilder builder=new StringBuilder();
    
```

```
        builder.append("External           File           Directories:  
        ").append(getExternalFilesDir(filepath).getAbsolutePath()).append("\n");  
        dir.setText(builder.toString());  
    }  
}
```

activity_main.xml

```
package com.example.myapplication;  
import androidx.appcompat.app.AppCompatActivity;  
import android.os.Bundle;  
import android.os.Environment;  
import android.view.View;  
import android.widget.Button;  
import android.widget.EditText;  
import android.widget.TextView;  
import android.widget.Toast;  
import java.io.BufferedReader;  
import java.io.File;  
import java.io.FileInputStream;  
import java.io.FileOutputStream;  
import java.io.IOException;  
import java.io.InputStreamReader;
```

```
public class MainActivity extends AppCompatActivity {  
  
    private EditText inputText;  
    private Button btnWrite, btnLoad;  
    private TextView dir;  
    private String filename = "hello.txt";  
    private String filepath = "MyFileStorage";  
    private File extFile;  
    private String data = "";  
  
    @Override  
    protected void onCreate(Bundle savedInstanceState) {  
        super.onCreate(savedInstanceState);  
        setContentView(R.layout.activity_main);  
        inputText = findViewById(R.id.input_text);  
        btnWrite = findViewById(R.id.btn_write);  
        btnLoad = findViewById(R.id.btn_load);  
        dir = findViewById(R.id.dir);  
        if (!isExternalStorageAvailable() || isExternalStorageReadOnly()) {  
            btnWrite.setEnabled(false);  
        }  
        else {  
            extFile = new File(getExternalFilesDir(filepath), filename);  
        }  
        getDir();  
        btnWrite.setOnClickListener(new View.OnClickListener() {
```

```
@Override
public void onClick(View view) {
    data = inputText.getText().toString();
    try {
        FileOutputStream fos = new FileOutputStream(extFile);
        fos.write(data.getBytes());
        // fos.write("Hello".getBytes());
        inputText.getText().clear();
        Toast.makeText(getApplicationContext(), "filename + " saved to external
storage...", Toast.LENGTH_SHORT).show();
        fos.close();
    }
    catch (IOException ex) {
        ex.printStackTrace();
    }
}
});

btnLoad.setOnClickListener(new View.OnClickListener() {
    @Override
    public void onClick(View view) {
        try {
            FileInputStream fis = new FileInputStream(extFile);
            InputStreamReader isr = new InputStreamReader(fis);
            BufferedReader br = new BufferedReader(isr);
            StringBuilder data = new StringBuilder();
            String line;
            while ((line = br.readLine()) != null) {
                data.append("\n").append(line);
            }
            inputText.setText(data);
            Toast.makeText(getApplicationContext(), "Data Retrieved from External File
Successfully...", Toast.LENGTH_SHORT).show();
            fis.close();
        }
        catch (IOException ex) {
            ex.printStackTrace();
        }
    }
});
}

private static boolean isExternalStorageAvailable() {
    String extStorageState = Environment.getExternalStorageState();
    return Environment.MEDIA_MOUNTED.equals(extStorageState);
}

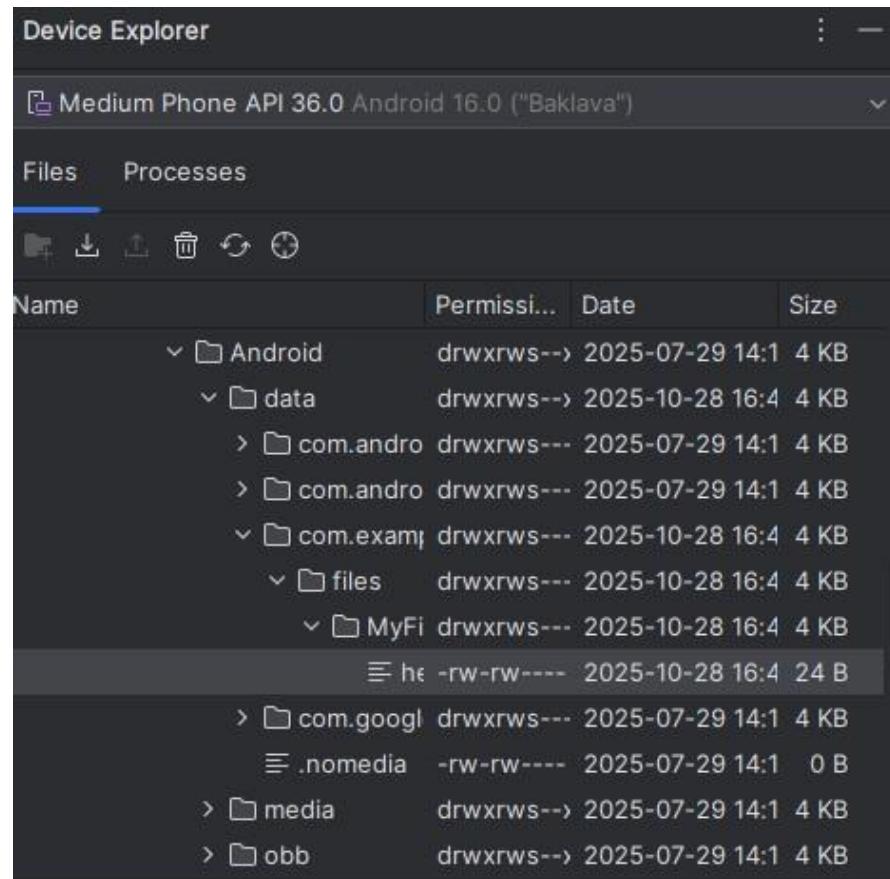
private static boolean isExternalStorageReadOnly() {
    String extStorageState = Environment.getExternalStorageState();
    return Environment.MEDIA_MOUNTED_READ_ONLY.equals(extStorageState);
}

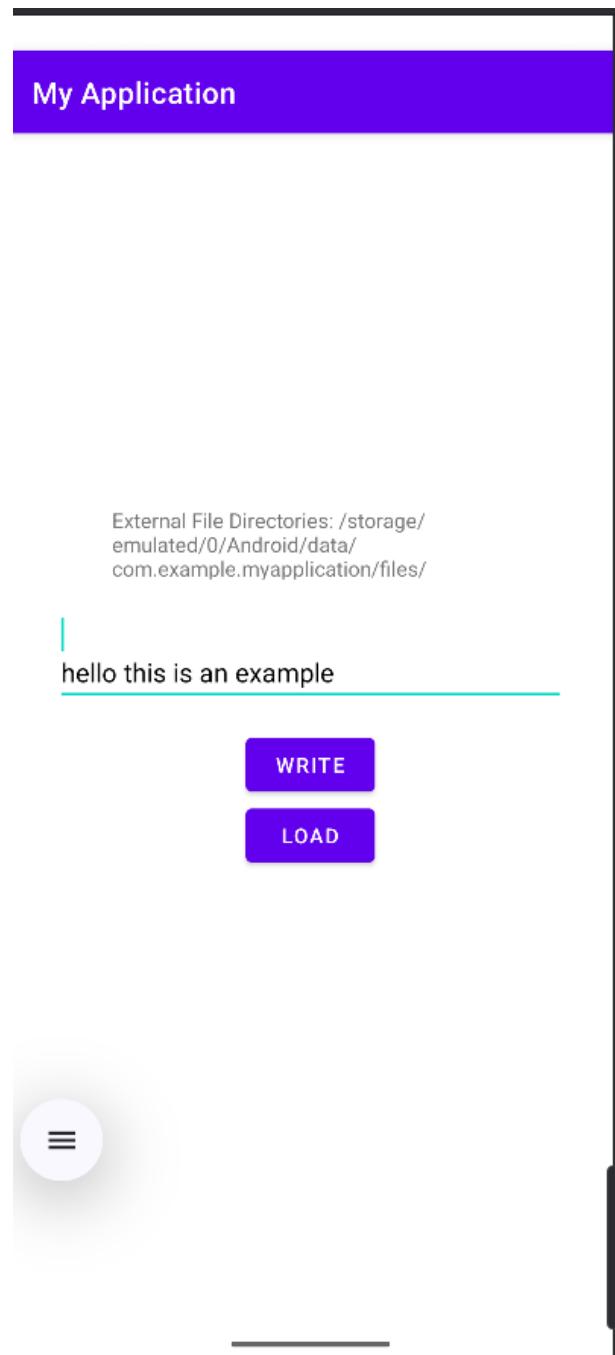
private void getDir()
```

```
{  
    StringBuilder builder=new StringBuilder();  
    builder.append("External  
").append(getExternalFilesDir(filepath).getAbsolutePath()).append("\n");  
    dir.setText(builder.toString());  
}  
}
```

Directories:

Output:





Practical 3: Write an android program for shared preference to store value in name-value pairs.

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

```
        android:padding="20dp"
        android:gravity="center_horizontal">

    <EditText
        android:id="@+id/etName"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:hint="Enter your name"
        android:minHeight="80dp"
        android:gravity="center"/>

    <Button
        android:id="@+id	btnSave"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="Save Name"
        android:layout_marginTop="20dp"/>

    <Button
        android:id="@+id	btnShow"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="Show Saved Name"
        android:layout_marginTop="10dp"/>

    <TextView
        android:id="@+id/tvDisplay"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:text="Your saved name will appear here"
        android:textSize="18sp"
        android:gravity="center"
        android:layout_marginTop="20dp"/>
</LinearLayout>
```

```
package com.example.sharedpreferenceexample;

import androidx.appcompat.app.AppCompatActivity;
import android.os.Bundle;
import android.view.View;
import android.widget.*;
import android.content.SharedPreferences;

public class MainActivity extends AppCompatActivity {

    EditText etName;
    Button btnSave, btnShow;
    TextView tvDisplay;

    SharedPreferences sharedPreferences;
```

```
public static final String MY_PREFS = "MyPrefs";
public static final String KEY_NAME = "userName";

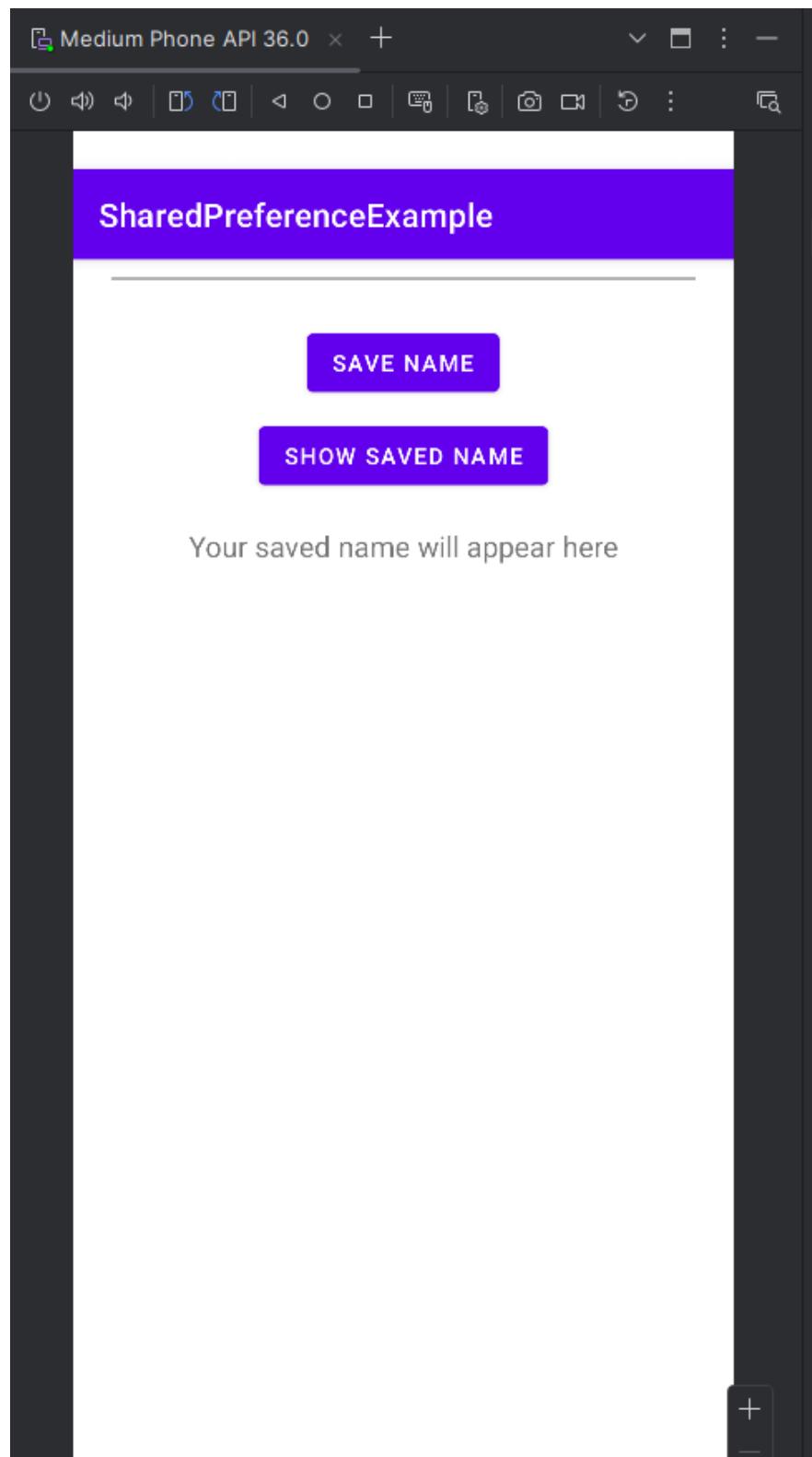
@Override
protected void onCreate(Bundle savedInstanceState) {
    super.onCreate(savedInstanceState);
    setContentView(R.layout.activity_main);

    etName = findViewById(R.id.etName);
    btnSave = findViewById(R.id.btnSave);
    btnShow = findViewById(R.id.btnShow);
    tvDisplay = findViewById(R.id.tvDisplay);

    // Initialize Shared Preferences
    sharedPreferences = getSharedPreferences(MY_PREFS, MODE_PRIVATE);

    // SAVE button click
    btnSave.setOnClickListener(new View.OnClickListener() {
        @Override
        public void onClick(View v) {
            String name = etName.getText().toString();
            if (name.isEmpty()) {
                Toast.makeText(MainActivity.this, "Please enter your name",
                        Toast.LENGTH_SHORT).show();
            } else {
                SharedPreferences.Editor editor = sharedPreferences.edit();
                editor.putString(KEY_NAME, name);
                editor.apply(); // or commit()
                Toast.makeText(MainActivity.this, "Name saved successfully!",
                        Toast.LENGTH_SHORT).show();
                etName.setText("");
            }
        }
    });

    // SHOW button click
    btnShow.setOnClickListener(new View.OnClickListener() {
        @Override
        public void onClick(View v) {
            String savedName = sharedPreferences.getString(KEY_NAME, "No name found");
            tvDisplay.setText("Saved Name: " + savedName);
        }
    });
}
```

Output:

Practical 4: Create a login form with a remember me checkbox. Save the username and password if the checkbox is checked using shared preference and show the welcome page when the login button is clicked.

```
<?xml version="1.0" encoding="utf-8"?>
<LinearLayout
    xmlns:android="http://schemas.android.com/apk/res/android"
    android:orientation="vertical"
    android:gravity="center"
    android:padding="20dp"
    android:layout_width="match_parent"
    android:layout_height="match_parent">

    <TextView
        android:text="Login Form"
        android:textSize="24sp"
        android:textStyle="bold"
        android:layout_marginBottom="30dp"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"/>

    <EditText
        android:id="@+id/etUsername"
        android:hint="Username"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:inputType="textPersonName"/>

    <EditText
        android:id="@+id/etPassword"
        android:hint="Password"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:inputType="textPassword"
        android:layout_marginTop="10dp"/>

    <CheckBox
        android:id="@+id/chkRemember"
        android:text="Remember Me"
        android:layout_marginTop="10dp"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"/>

    <Button
        android:id="@+id/btnLogin"
        android:text="Login"
        android:layout_marginTop="20dp"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"/>
```

```
</LinearLayout>

package com.example.loginsharedprefapp;

import androidx.appcompat.app.AppCompatActivity;
import android.content.Intent;
import android.content.SharedPreferences;
import android.os.Bundle;
import android.view.View;
import android.widget.*;

public class MainActivity extends AppCompatActivity {

    EditText etUsername, etPassword;
    CheckBox chkRemember;
    Button btnLogin;
    SharedPreferences sharedPreferences;

    public static final String PREFS_NAME = "LoginPrefs";
    public static final String KEY_USER = "username";
    public static final String KEY_PASS = "password";
    public static final String KEY_REMEMBER = "remember";

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

        etUsername = findViewById(R.id.etUsername);
        etPassword = findViewById(R.id.etPassword);
        chkRemember = findViewById(R.id.chkRemember);
        btnLogin = findViewById(R.id.btnLogin);

        sharedPreferences = getSharedPreferences(PREFS_NAME, MODE_PRIVATE);

        // Load saved data if Remember Me was checked
        boolean isRemembered = sharedPreferences.getBoolean(KEY_REMEMBER, false);
        if (isRemembered) {
            String savedUser = sharedPreferences.getString(KEY_USER, "");
            String savedPass = sharedPreferences.getString(KEY_PASS, "");
            etUsername.setText(savedUser);
            etPassword.setText(savedPass);
            chkRemember.setChecked(true);
        }

        btnLogin.setOnClickListener(new View.OnClickListener() {
            @Override
            public void onClick(View v) {
                String username = etUsername.getText().toString();
                String password = etPassword.getText().toString();
            }
        });
    }
}
```

```
        if (username.isEmpty() || password.isEmpty()) {
            Toast.makeText(MainActivity.this, "Enter username and password",
Toast.LENGTH_SHORT).show();
        } else {
            SharedPreferences.Editor editor = sharedPreferences.edit();

            if (chkRemember.isChecked()) {
                editor.putString(KEY_USER, username);
                editor.putString(KEY_PASS, password);
                editor.putBoolean(KEY_REMEMBER, true);
            } else {
                editor.clear(); // Remove saved data if unchecked
            }

            editor.apply();

            // Navigate to WelcomeActivity
            Intent intent = new Intent(MainActivity.this, MainActivity2.class);
            intent.putExtra("username", username);
            startActivity(intent);
        }
    });
}
```

```
<?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=".MainActivity2">

    <TextView
        android:id="@+id/tvWelcome"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="TextView"
        tools:layout_editor_absoluteX="66dp"
        tools:layout_editor_absoluteY="48dp" />
</androidx.constraintlayout.widget.ConstraintLayout>
```

```
package com.example.loginsharedprefapp;

import android.os.Bundle;
```

```
import android.widget.TextView;
```

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

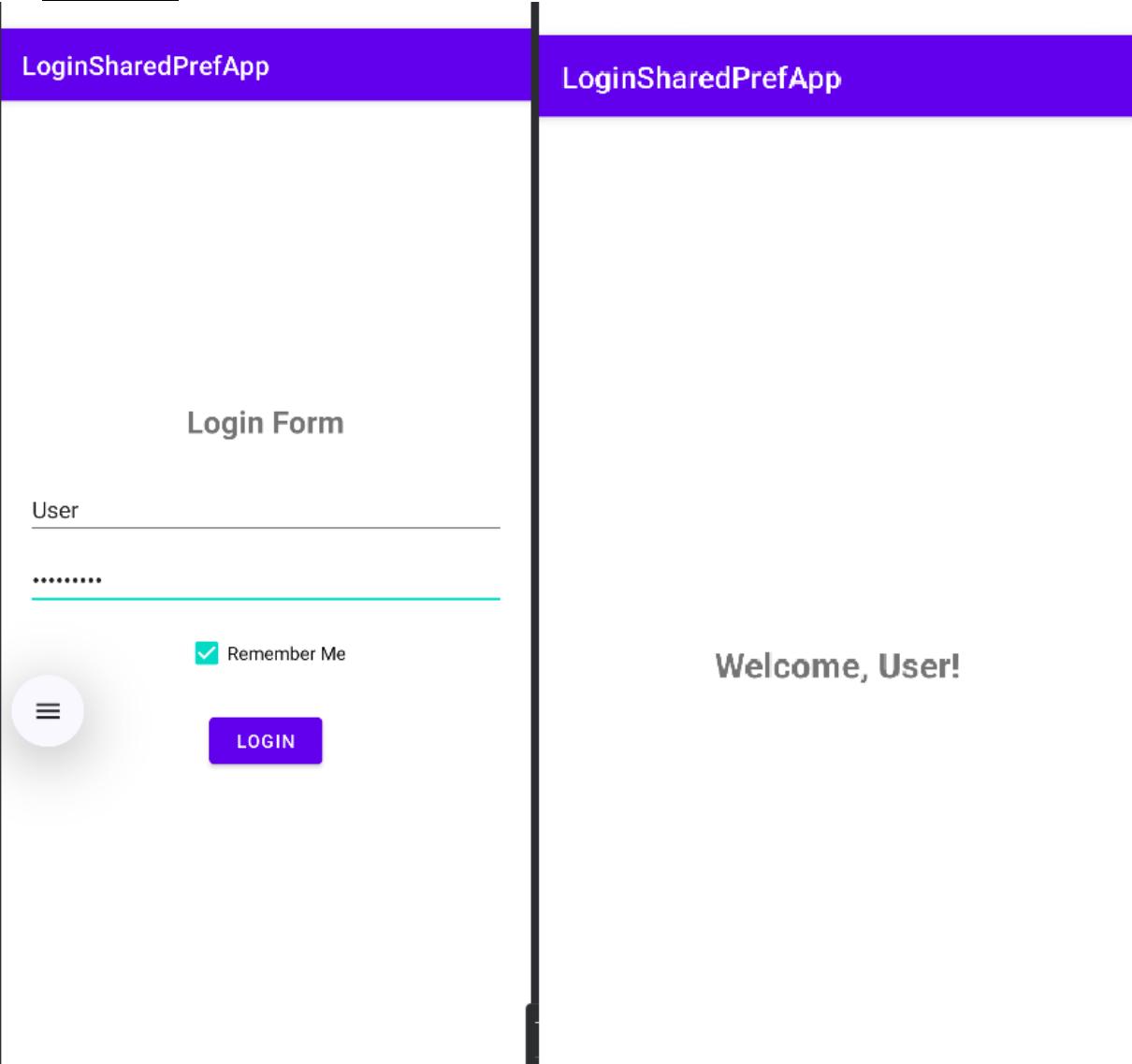
public class MainActivity2 extends AppCompatActivity {

    TextView tvWelcome;

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

        tvWelcome = findViewById(R.id.tvWelcome);

        String username = getIntent().getStringExtra("username");
        tvWelcome.setText("Welcome, " + username + "!");
    }
}
```

Output:

Practical 5: Create an Android application to insert, update, select, and delete records from the student table using SQLite Database.

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

    <LinearLayout
        android:orientation="vertical"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:padding="20dp"
        android:gravity="center_horizontal">

        <TextView
            android:text="Student Database App"
            android:textSize="24sp"
            android:textStyle="bold"
            android:layout_marginBottom="20dp"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"/>

        <EditText
            android:id="@+id/etId"
            android:hint="Student ID"
            android:inputType="number"
            android:layout_width="match_parent"
            android:layout_height="wrap_content"/>

        <EditText
            android:id="@+id/etName"
            android:hint="Student Name"
            android:layout_width="match_parent"
            android:layout_height="wrap_content"
            android:layout_marginTop="10dp"/>

        <EditText
            android:id="@+id/etCourse"
            android:hint="Course"
            android:layout_width="match_parent"
            android:layout_height="wrap_content"
            android:layout_marginTop="10dp"/>

        <EditText
            android:id="@+id/etMarks"
            android:hint="Marks"
            android:inputType="number"
            android:layout_width="match_parent"
            android:layout_height="wrap_content"
```

```
        android:layout_marginTop="10dp"/>

    <LinearLayout
        android:orientation="horizontal"
        android:gravity="center"
        android:layout_marginTop="20dp"
        android:layout_width="match_parent"
        android:layout_height="wrap_content">

        <Button
            android:id="@+id	btnInsert"
            android:text="Insert"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"/>

        <Button
            android:id="@+id	btnUpdate"
            android:text="Update"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:layout_marginStart="10dp"/>

        <Button
            android:id="@+id	btnDelete"
            android:text="Delete"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:layout_marginStart="10dp"/>
    </LinearLayout>

    <Button
        android:id="@+id	btnView"
        android:text="View All Records"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_marginTop="20dp"/>

</LinearLayout>
</ScrollView>

package com.example.sqlitestudentapp;

import androidx.appcompat.app.AppCompatActivity;
import android.database.Cursor;
import android.os.Bundle;
import android.view.View;
import android.widget.*;

public class MainActivity extends AppCompatActivity {

    EditText etId, etName, etCourse, etMarks;
```

```
Button btnInsert, btnUpdate, btnDelete, btnView;
DatabaseHelper myDB;

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

    myDB = new DatabaseHelper(this);

    etId = findViewById(R.id.etId);
    etName = findViewById(R.id.etName);
    etCourse = findViewById(R.id.etCourse);
    etMarks = findViewById(R.id.etMarks);
    btnInsert = findViewById(R.id.btnInsert);
    btnUpdate = findViewById(R.id.btnUpdate);
    btnDelete = findViewById(R.id.btnDelete);
    btnView = findViewById(R.id.btnView);

   .addData();
    updateData();
    deleteData();
    viewAll();
}

public void addData() {
    btnInsert.setOnClickListener(new View.OnClickListener() {
        @Override
        public void onClick(View v) {
            boolean isInserted = myDB.insertData(
                etName.getText().toString(),
                etCourse.getText().toString(),
                etMarks.getText().toString()
            );
            Toast.makeText(MainActivity.this,
                isInserted ? "Data Inserted" : "Insert Failed",
                Toast.LENGTH_SHORT).show();
        }
    });
}

public void updateData() {
    btnUpdate.setOnClickListener(new View.OnClickListener() {
        @Override
        public void onClick(View v) {
            boolean isUpdated = myDB.updateData(
                etId.getText().toString(),
                etName.getText().toString(),
                etCourse.getText().toString(),
                etMarks.getText().toString()
            );
        }
    });
}
```

);

```
        Toast.makeText(MainActivity.this,
            isUpdated ? "Data Updated" : "Update Failed",
            Toast.LENGTH_SHORT).show();
    }
}

public void deleteData() {
    btnDelete.setOnClickListener(new View.OnClickListener() {
        @Override
        public void onClick(View v) {
            int deletedRows = myDB.deleteData(etId.getText().toString());
            Toast.makeText(MainActivity.this,
                deletedRows > 0 ? "Data Deleted" : "Delete Failed",
                Toast.LENGTH_SHORT).show();
        }
    });
}

public void viewAll() {
    btnView.setOnClickListener(new View.OnClickListener() {
        @Override
        public void onClick(View v) {
            Cursor res = myDB.getAllData();
            if (res.getCount() == 0) {
                showMessage("Error", "No Data Found");
                return;
            }

            StringBuilder buffer = new StringBuilder();
            while (res.moveToNext()) {
                buffer.append("ID: ").append(res.getString(0)).append("\n");
                buffer.append("Name: ").append(res.getString(1)).append("\n");
                buffer.append("Course: ").append(res.getString(2)).append("\n");
                buffer.append("Marks: ").append(res.getString(3)).append("\n\n");
            }

            showMessage("Student Records", buffer.toString());
        }
    });
}

public void showMessage(String title, String message) {
    android.app.AlertDialog.Builder builder = new android.app.AlertDialog.Builder(this);
    builder.setCancelable(true);
    builder.setTitle(title);
    builder.setMessage(message);
    builder.show();
}
}
```

```
package com.example.sqlitestudentapp;

import android.content.ContentValues;
import android.content.Context;
import android.database.Cursor;
import android.database.sqlite.SQLiteDatabase;
import android.database.sqlite.SQLiteOpenHelper;

public class DatabaseHelper extends SQLiteOpenHelper {

    public static final String DATABASE_NAME = "StudentDB";
    public static final String TABLE_NAME = "student_table";
    public static final String COL_1 = "ID";
    public static final String COL_2 = "NAME";
    public static final String COL_3 = "COURSE";
    public static final String COL_4 = "MARKS";

    public DatabaseHelper(Context context) {
        super(context, DATABASE_NAME, null, 1);
    }

    @Override
    public void onCreate(SQLiteDatabase db) {
        db.execSQL("CREATE TABLE " + TABLE_NAME +
                " (ID INTEGER PRIMARY KEY AUTOINCREMENT, NAME TEXT, COURSE
                TEXT, MARKS INTEGER)");
    }

    @Override
    public void onUpgrade(SQLiteDatabase db, int oldVersion, int newVersion) {
        db.execSQL("DROP TABLE IF EXISTS " + TABLE_NAME);
        onCreate(db);
    }

    // INSERT
    public boolean insertData(String name, String course, String marks) {
        SQLiteDatabase db = this.getWritableDatabase();
        ContentValues contentValues = new ContentValues();
        contentValues.put(COL_2, name);
        contentValues.put(COL_3, course);
        contentValues.put(COL_4, marks);
        long result = db.insert(TABLE_NAME, null, contentValues);
        return result != -1; // return true if inserted
    }

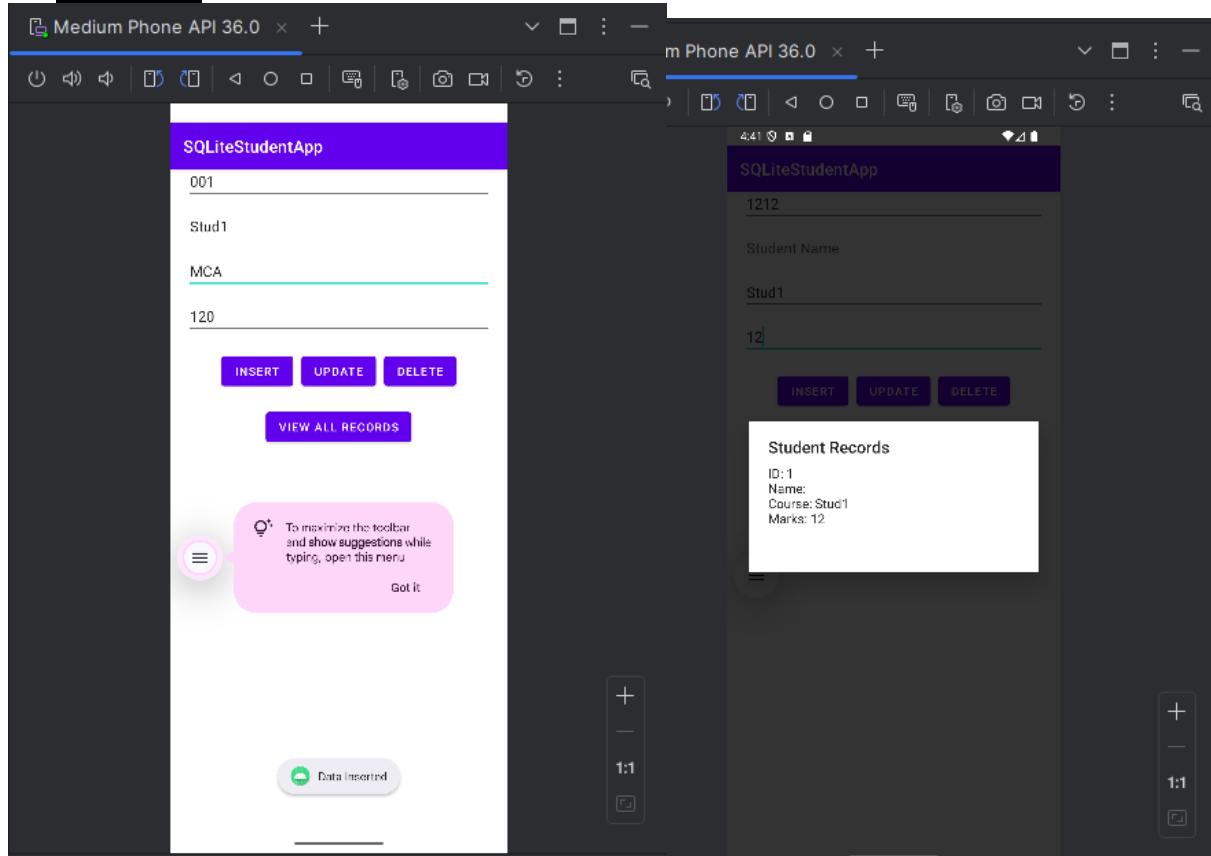
    // UPDATE
    public boolean updateData(String id, String name, String course, String marks) {
        SQLiteDatabase db = this.getWritableDatabase();
        ContentValues contentValues = new ContentValues();
        contentValues.put(COL_2, name);
        contentValues.put(COL_3, course);
```

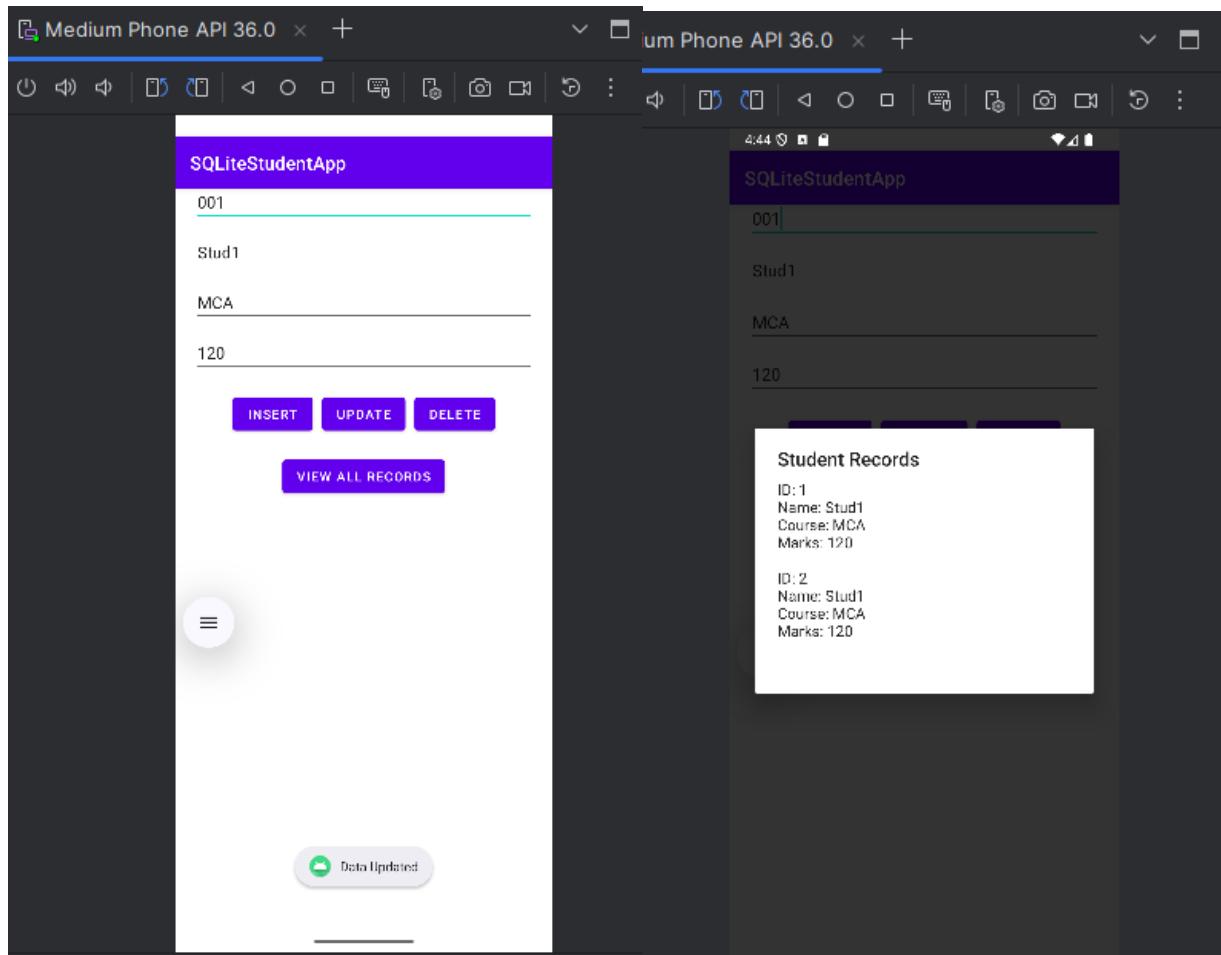
```
contentValues.put(COL_4, marks);
int result = db.update(TABLE_NAME, contentValues, "ID = ?", new String[]{id});
return result > 0;
}

// DELETE
public int deleteData(String id) {
    SQLiteDatabase db = this.getWritableDatabase();
    return db.delete(TABLE_NAME, "ID = ?", new String[]{id});
}

// SELECT ALL
public Cursor getAllData() {
    SQLiteDatabase db = this.getWritableDatabase();
    return db.rawQuery("SELECT * FROM " + TABLE_NAME, null);
}
}
```

Output:





Practical 6: Write a program to create a user registration form, after registration data will be inserted in the SQLite database, and design an activity that displays that information.

```
<?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"
    android:orientation="vertical"
    android:padding="20dp">

    <EditText
        android:id="@+id/etName"
        android:hint="Enter Name"
        android:layout_width="match_parent"
        android:layout_height="wrap_content" />

    <EditText
        android:id="@+id/etEmail"
        android:hint="Enter Email"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:inputType="textEmailAddress"/>

    <EditText
        android:id="@+id/etPhone"
        android:hint="Enter Phone"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:inputType="phone"/>

    <Button
        android:id="@+id/btnRegister"
        android:text="Register"
        android:layout_width="match_parent"
        android:layout_height="wrap_content" />

    <Button
        android:id="@+id/btnViewUsers"
        android:text="View Users"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:layout_marginTop="10dp"/>
</LinearLayout>
```

```
package com.example.userregistrationapp;

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

```
import android.content.Intent;
import android.database.Cursor;
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 {

    DatabaseHelper db;
    EditText etName, etEmail, etPhone;
    Button btnRegister, btnViewUsers;

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

        db = new DatabaseHelper(this);
        etName = findViewById(R.id.etName);
        etEmail = findViewById(R.id.etEmail);
        etPhone = findViewById(R.id.etPhone);
        btnRegister = findViewById(R.id.btnRegister);
        btnViewUsers = findViewById(R.id.btnViewUsers);

        btnRegister.setOnClickListener(v -> {
            String name = etName.getText().toString();
            String email = etEmail.getText().toString();
            String phone = etPhone.getText().toString();

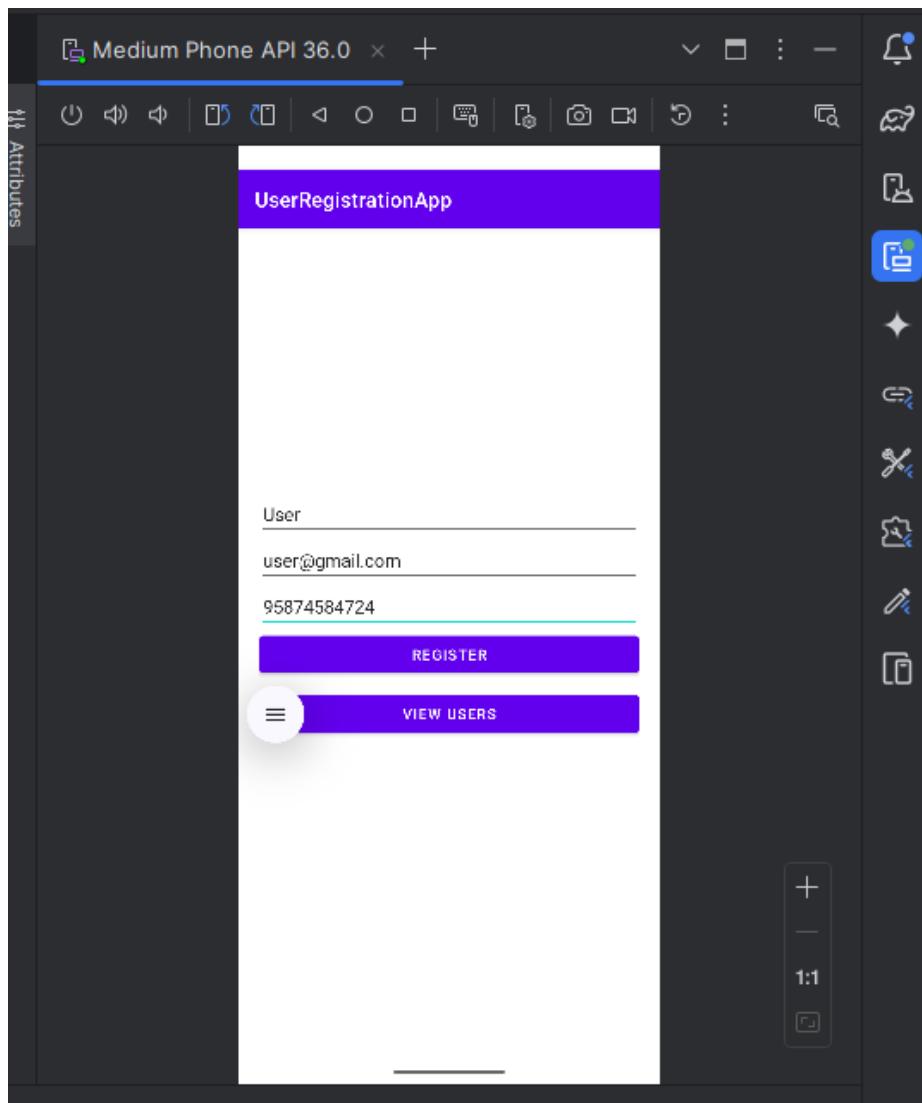
            if (name.isEmpty() || email.isEmpty() || phone.isEmpty()) {
                Toast.makeText(this, "Please fill all fields", Toast.LENGTH_SHORT).show();
            } else {
                boolean inserted = db.insertUser(name, email, phone);
                if (inserted) {
                    Toast.makeText(this, "User Registered Successfully!", Toast.LENGTH_SHORT).show();
                    etName.setText("");
                    etEmail.setText("");
                    etPhone.setText("");
                } else {
                    Toast.makeText(this, "Registration Failed!", Toast.LENGTH_SHORT).show();
                }
            }
        });

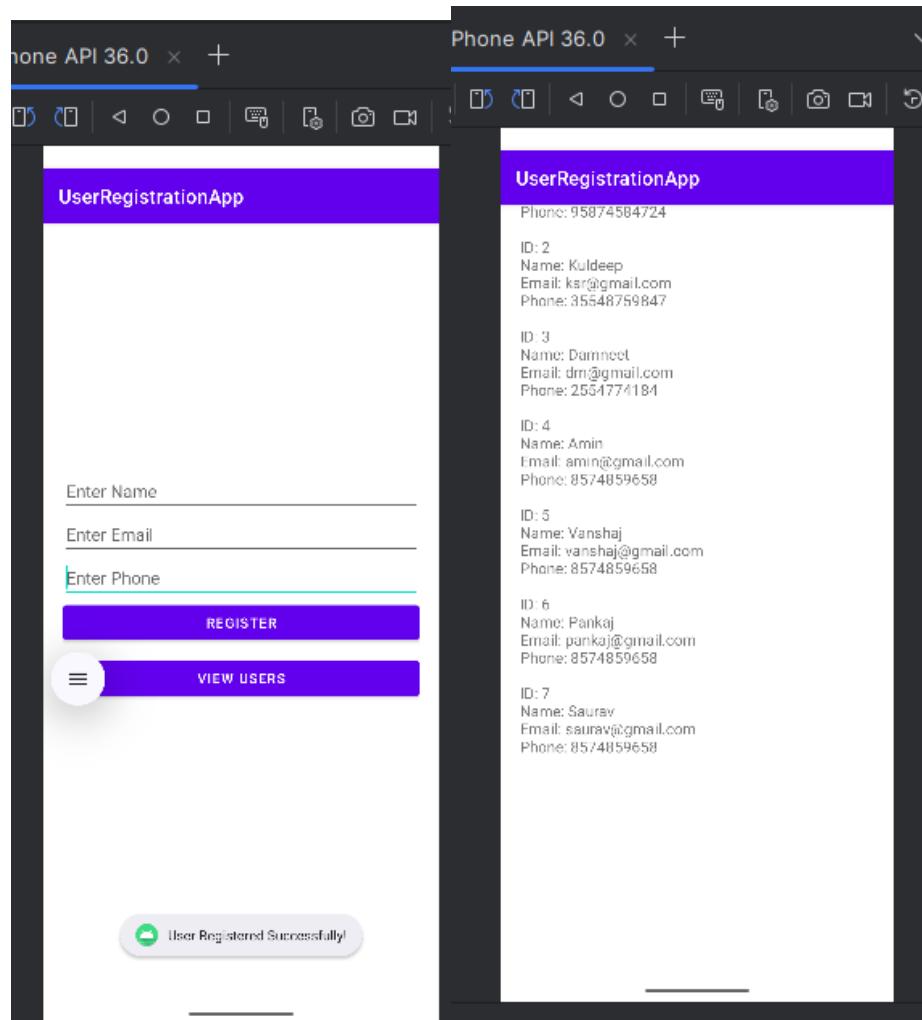
        btnViewUsers.setOnClickListener(v -> {
            Intent intent = new Intent(MainActivity.this, activity_view_users.class);
            startActivity(intent);
        });
    }
}
```

```
        }  
    }  
  
package com.example.userregistrationapp;  
  
import android.content.ContentValues;  
import android.content.Context;  
import android.database.Cursor;  
import android.database.sqlite.SQLiteDatabase;  
import android.database.sqlite.SQLiteOpenHelper;  
  
public class DatabaseHelper extends SQLiteOpenHelper {  
  
    public static final String DATABASE_NAME = "UserDB";  
    public static final String TABLE_NAME = "users";  
    public static final String COL_ID = "id";  
    public static final String COL_NAME = "name";  
    public static final String COL_EMAIL = "email";  
    public static final String COL_PHONE = "phone";  
  
    public DatabaseHelper(Context context) {  
        super(context, DATABASE_NAME, null, 1);  
    }  
  
    @Override  
    public void onCreate(SQLiteDatabase db) {  
        db.execSQL("CREATE TABLE " + TABLE_NAME +  
                  " (" + COL_ID + " INTEGER PRIMARY KEY AUTOINCREMENT, " +  
                  COL_NAME + " TEXT, " +  
                  COL_EMAIL + " TEXT, " +  
                  COL_PHONE + " TEXT)");  
    }  
  
    @Override  
    public void onUpgrade(SQLiteDatabase db, int oldVersion, int newVersion) {  
        db.execSQL("DROP TABLE IF EXISTS " + TABLE_NAME);  
        onCreate(db);  
    }  
  
    // Insert User Data  
    public boolean insertUser(String name, String email, String phone) {  
        SQLiteDatabase db = this.getWritableDatabase();  
        ContentValues values = new ContentValues();  
        values.put(COL_NAME, name);  
        values.put(COL_EMAIL, email);  
        values.put(COL_PHONE, phone);  
        long result = db.insert(TABLE_NAME, null, values);  
        return result != -1;  
    }  
  
    // Get All Users
```

```
public Cursor getAllUsers() {  
    SQLiteDatabase db = this.getReadableDatabase();  
    return db.rawQuery("SELECT * FROM " + TABLE_NAME, null);  
}  
}
```

Output:





Practical 7: Android Program to perform CRUD operation using real time database Firebase.

MainActivity.java

```
package com.example.myapplication;
import androidx.appcompat.app.AppCompatActivity;
import android.os.Bundle;
import android.os.Environment;
import android.view.View;
import android.widget.Button;
import android.widget.EditText;
import android.widget.TextView;
import android.widget.Toast;
import java.io.BufferedReader;
import java.io.File;
import java.io.FileInputStream;
import java.io.FileOutputStream;
import java.io.IOException;
import java.io.InputStreamReader;

public class MainActivity extends AppCompatActivity {

    private EditText inputText;
    private Button btnWrite, btnLoad;
    private TextView dir;
    private String filename = "hello.txt";
    private String filepath = "MyFileStorage";
    private File extFile;
    private String data = "";

    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_main);
        inputText = findViewById(R.id.input_text);
        btnWrite = findViewById(R.id.btn_write);
        btnLoad = findViewById(R.id.btn_load);
        dir = findViewById(R.id.dir);
        if (!isExternalStorageAvailable() || isExternalStorageReadOnly()) {
            btnWrite.setEnabled(false);
        } else {
            extFile = new File(getExternalFilesDir(filepath), filename);
        }
        getDir();
        btnWrite.setOnClickListener(new View.OnClickListener() {
            @Override
            public void onClick(View view) {
                data = inputText.getText().toString();
                try {
                    FileOutputStream fos = new FileOutputStream(extFile);
                    fos.write(data.getBytes());
                    // fos.write("Hello".getBytes());
                    inputText.getText().clear();
                    Toast.makeText(getApplicationContext(), filename + " saved to external storage...",
```

```

Toast.LENGTH_SHORT).show();
        fos.close();
    }
    catch (IOException ex) {
        ex.printStackTrace();
    }
}
});

btnLoad.setOnClickListener(new View.OnClickListener() {
    @Override
    public void onClick(View view) {
        try {
            FileInputStream fis = new FileInputStream(extFile);
            InputStreamReader isr = new InputStreamReader(fis);
            BufferedReader br = new BufferedReader(isr);
            StringBuilder data = new StringBuilder();
            String line;
            while ((line = br.readLine()) != null) {
                data.append("\n").append(line);
            }
            inputText.setText(data);
            Toast.makeText(getApplicationContext(), "Data Retrieved from External File Successfully...", Toast.LENGTH_SHORT).show();
            fis.close();
        }
        catch (IOException ex) {
            ex.printStackTrace();
        }
    }
});
}

private static boolean isExternalStorageAvailable() {
    String extStorageState = Environment.getExternalStorageState();
    return Environment.MEDIA_MOUNTED.equals(extStorageState);
}

private static boolean isExternalStorageReadOnly() {
    String extStorageState = Environment.getExternalStorageState();
    return Environment.MEDIA_MOUNTED_READ_ONLY.equals(extStorageState);
}

private void getDir()
{
    StringBuilder builder=new StringBuilder();
    builder.append("External File");
    builder.append(getExternalFilesDir(filepath).getAbsolutePath()).append("\n");
    dir.setText(builder.toString());
}
}

```

Directories:

POJO.java

```

package com.example.myapplication;
import androidx.appcompat.app.AppCompatActivity;
import android.os.Bundle;
import android.os.Environment;

```

```
import android.view.View;
import android.widget.Button;
import android.widget.EditText;
import android.widget.TextView;
import android.widget.Toast;
import java.io.BufferedReader;
import java.io.File;
import java.io.FileInputStream;
import java.io.FileOutputStream;
import java.io.IOException;
import java.io.InputStreamReader;

public class MainActivity extends AppCompatActivity {

    private EditText inputText;
    private Button btnWrite, btnLoad;
    private TextView dir;
    private String filename = "hello.txt";
    private String filepath = "MyFileStorage";
    private File extFile;
    private String data = "";

    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_main);
        inputText = findViewById(R.id.input_text);
        btnWrite = findViewById(R.id.btn_write);
        btnLoad = findViewById(R.id.btn_load);
        dir = findViewById(R.id.dir);
        if (!isExternalStorageAvailable() || isExternalStorageReadOnly()) {
            btnWrite.setEnabled(false);
        } else {
            extFile = new File(getExternalFilesDir(filepath), filename);
        }
        getDir();
        btnWrite.setOnClickListener(new View.OnClickListener() {
            @Override
            public void onClick(View view) {
                data = inputText.getText().toString();
                try {
                    FileOutputStream fos = new FileOutputStream(extFile);
                    fos.write(data.getBytes());
                    // fos.write("Hello".getBytes());
                    inputText.getText().clear();
                    Toast.makeText(getApplicationContext(), filename + " saved to external storage...", Toast.LENGTH_SHORT).show();
                    fos.close();
                } catch (IOException ex) {
                    ex.printStackTrace();
                }
            }
        });
        btnLoad.setOnClickListener(new View.OnClickListener() {
```

```

@Override
public void onClick(View view) {
    try {
        FileInputStream fis = new FileInputStream(extFile);
        InputStreamReader isr = new InputStreamReader(fis);
        BufferedReader br = new BufferedReader(isr);
        StringBuilder data = new StringBuilder();
        String line;
        while ((line = br.readLine()) != null) {
            data.append("\n").append(line);
        }
        inputText.setText(data);
        Toast.makeText(getApplicationContext(), "Data Retrieved from External File Successfully...", Toast.LENGTH_SHORT).show();
        fis.close();
    }
    catch (IOException ex) {
        ex.printStackTrace();
    }
}
});

private static boolean isExternalStorageAvailable() {
    String extStorageState = Environment.getExternalStorageState();
    return Environment.MEDIA_MOUNTED.equals(extStorageState);
}

private static boolean isExternalStorageReadOnly() {
    String extStorageState = Environment.getExternalStorageState();
    return Environment.MEDIA_MOUNTED_READ_ONLY.equals(extStorageState);
}

private void getDir()
{
    StringBuilder builder=new StringBuilder();
    builder.append("External").append(File.separator);
    builder.append(getExternalFilesDir(filepath).getAbsolutePath()).append("\n");
    dir.setText(builder.toString());
}
}

```

Directories:

XML File

```

package com.example.myapplication;

import androidx.appcompat.app.AppCompatActivity;
import android.os.Bundle;
import android.os.Environment;
import android.view.View;
import android.widget.Button;
import android.widget.EditText;
import android.widget.TextView;
import android.widget.Toast;
import java.io.BufferedReader;

```

```
import java.io.File;
import java.io.FileInputStream;
import java.io.FileOutputStream;
import java.io.IOException;
import java.io.InputStreamReader;

public class MainActivity extends AppCompatActivity {

    private EditText inputText;
    private Button btnWrite, btnLoad;
    private TextView dir;
    private String filename = "hello.txt";
    private String filepath = "MyFileStorage";
    private File extFile;
    private String data = "";

    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_main);
        inputText = findViewById(R.id.input_text);
        btnWrite = findViewById(R.id.btn_write);
        btnLoad = findViewById(R.id.btn_load);
        dir = findViewById(R.id.dir);
        if (!isExternalStorageAvailable() || isExternalStorageReadOnly()) {
            btnWrite.setEnabled(false);
        } else {
            extFile = new File(getExternalFilesDir(filepath), filename);
        }
        getDir();
        btnWrite.setOnClickListener(new View.OnClickListener() {
            @Override
            public void onClick(View view) {
                data = inputText.getText().toString();
                try {
                    FileOutputStream fos = new FileOutputStream(extFile);
                    fos.write(data.getBytes());
                    // fos.write("Hello".getBytes());
                    inputText.getText().clear();
                    Toast.makeText(getApplicationContext(), filename + " saved to external storage...", Toast.LENGTH_SHORT).show();
                    fos.close();
                } catch (IOException ex) {
                    ex.printStackTrace();
                }
            }
        });
        btnLoad.setOnClickListener(new View.OnClickListener() {
            @Override
            public void onClick(View view) {
                try {
                    FileInputStream fis = new FileInputStream(extFile);
                    InputStreamReader isr = new InputStreamReader(fis);
                    BufferedReader br = new BufferedReader(isr);

```

```
        StringBuilder data = new StringBuilder();
        String line;
        while ((line = br.readLine()) != null) {
            data.append("\n").append(line);
        }
        inputText.setText(data);
        Toast.makeText(getApplicationContext(), "Data Retrieved from External File Successfully...", Toast.LENGTH_SHORT).show();
        fis.close();
    }
    catch (IOException ex) {
        ex.printStackTrace();
    }
}
});

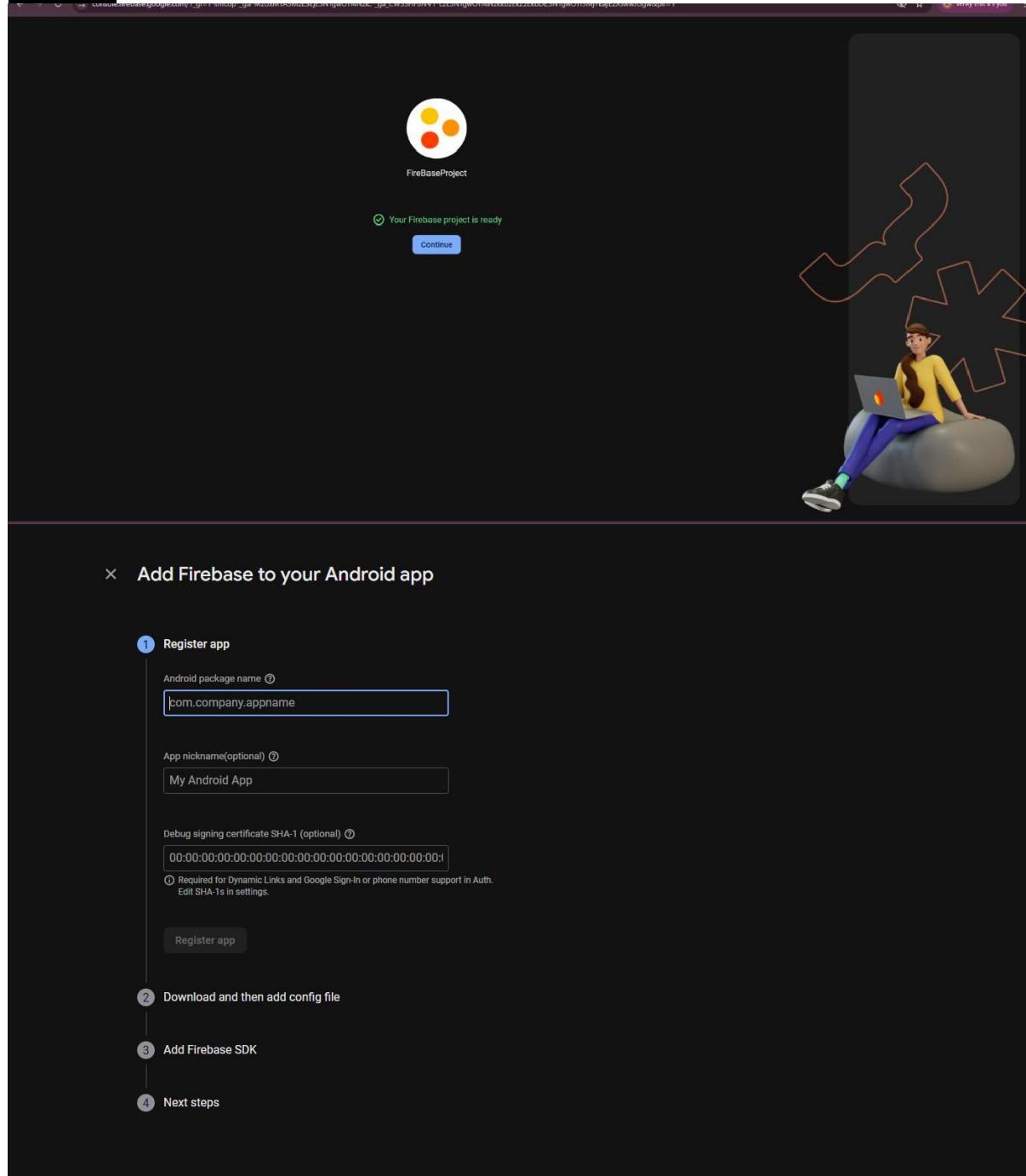
private static boolean isExternalStorageAvailable() {
    String extStorageState = Environment.getExternalStorageState();
    return Environment.MEDIA_MOUNTED.equals(extStorageState);
}

private static boolean isExternalStorageReadOnly() {
    String extStorageState = Environment.getExternalStorageState();
    return Environment.MEDIA_MOUNTED_READ_ONLY.equals(extStorageState);
}

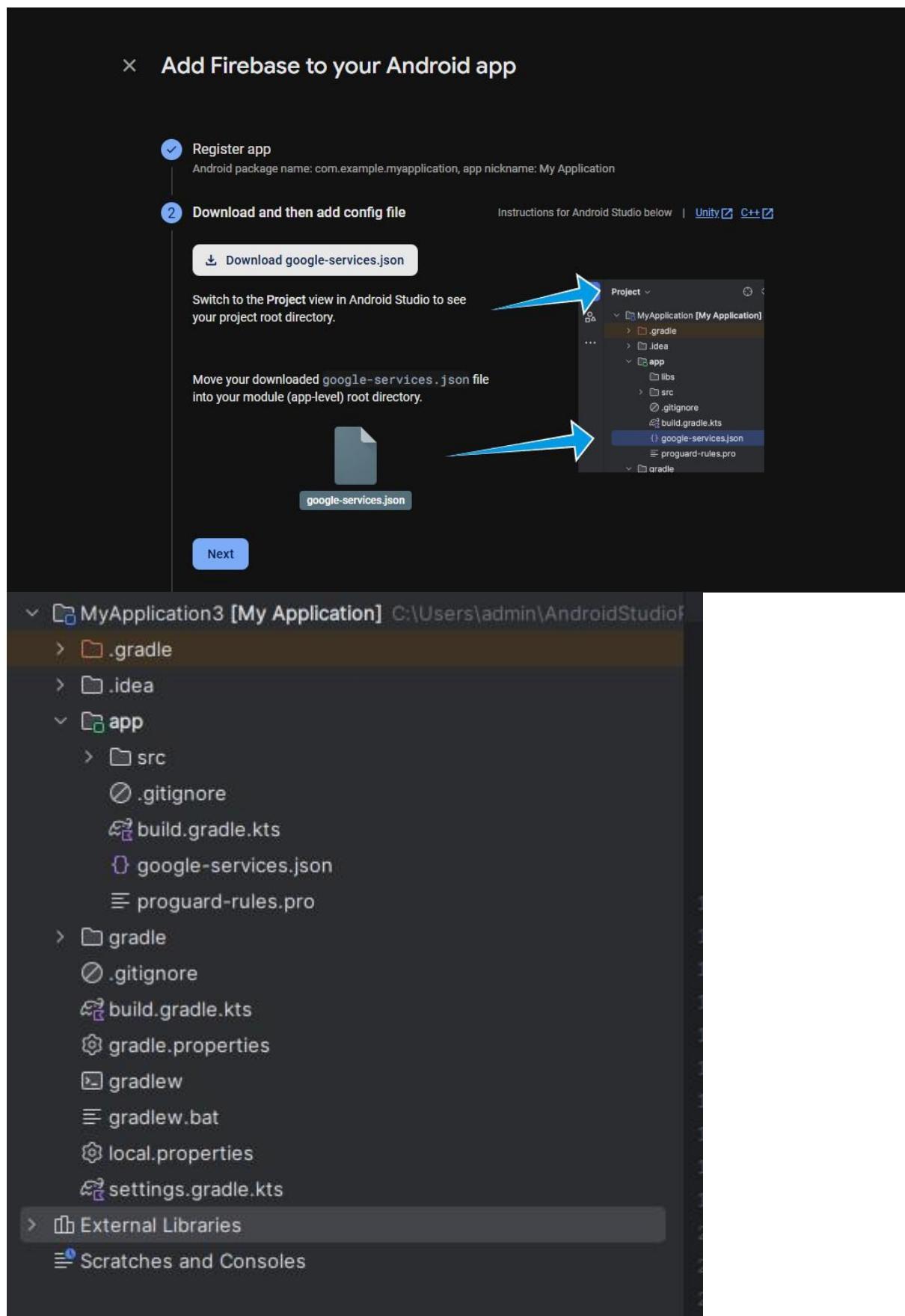
private void getDir()
{
    StringBuilder builder=new StringBuilder();
    builder.append("External File");
    builder.append(getExternalFilesDir(filepath).getAbsolutePath()).append("\n");
    dir.setText(builder.toString());
}
```

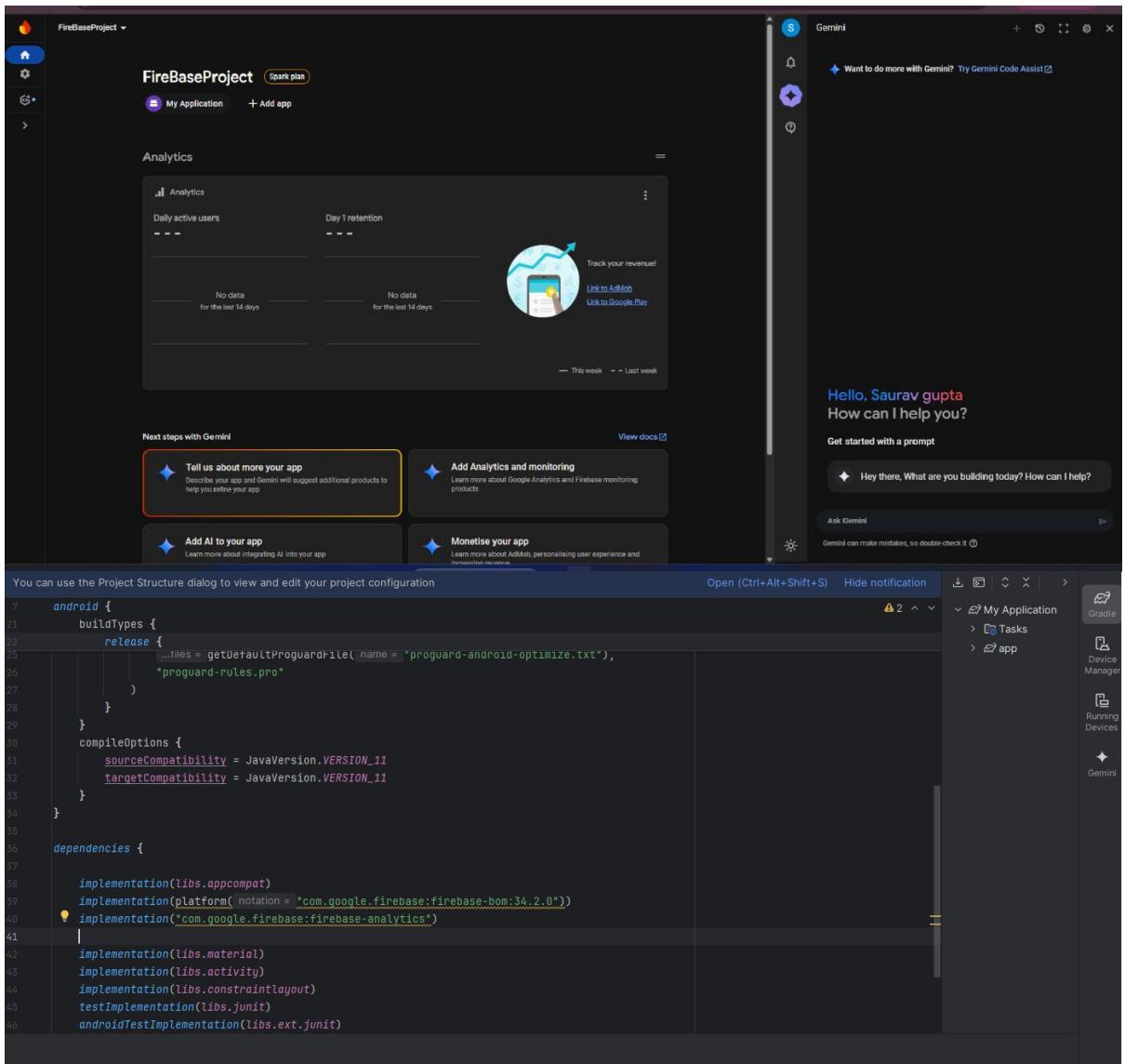
Directories:

Output:

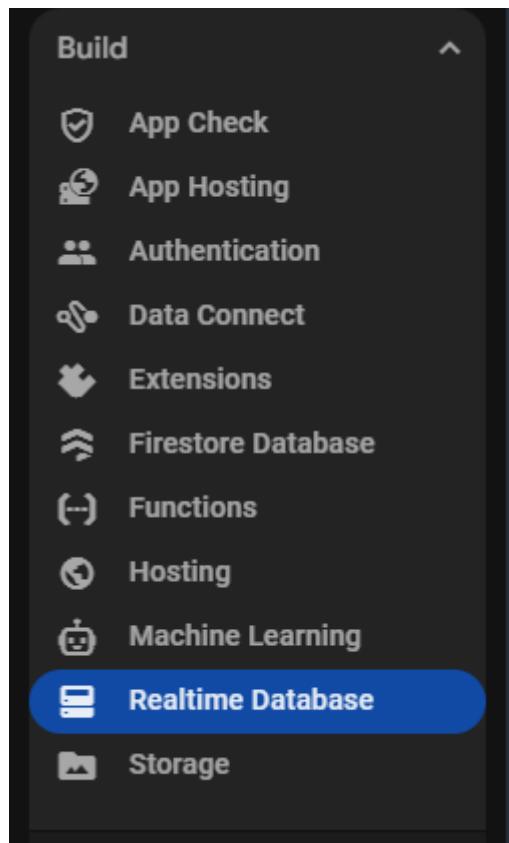


x Add Firebase to your Android app





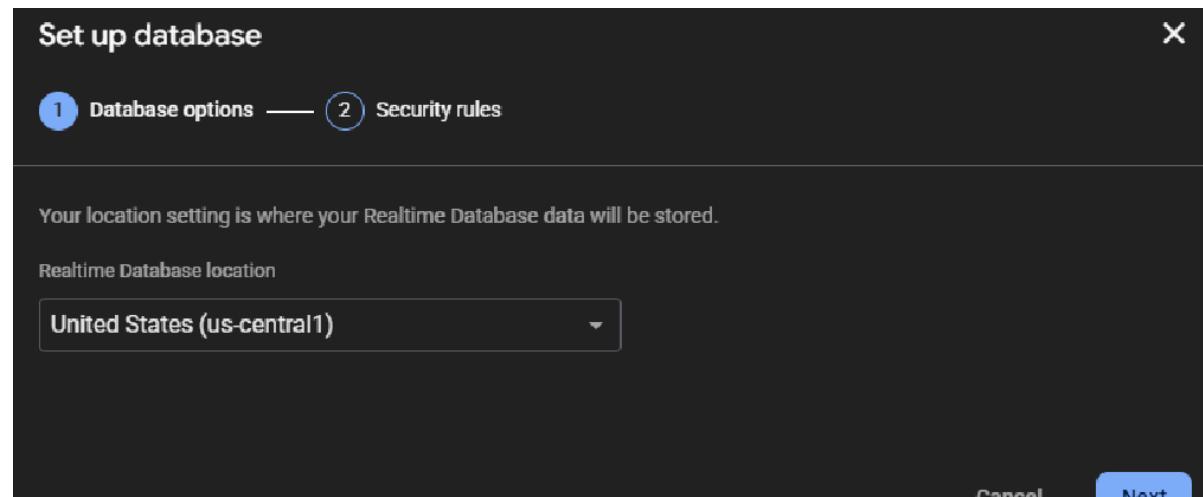
Follow steps and do sync now



Realtime Database
Store and sync data in real time

Create Database Ask Gemini

Is Realtime Database right for you? [Compare databases](#)



Once you have defined your data structure you will have to write rules to secure your data.

[Learn more](#)

Start in **locked mode**

Your data is private by default. Client read/write access will only be granted as specified by your security rules.

Start in **Test mode**

Your data is open by default to enable quick setup. However, you must update your security rules within 30 days to enable long-term client read/write access.

{
 "rules": {
 ".read": "now < 1760639400000", // 2025-10-17
 ".write": "now < 1760639400000", // 2025-10-17
 }
}

💡 The default security rules for test mode allow anyone with your database reference to view, edit and delete all data in your database for the next 30 days

Cancel Enable

★ Default security rules are locked from access

[Learn more](#) [Dismiss](#)

```
1  {  
2    /* Visit https://firebase.google.com/docs/database/security to learn more about security rules. */  
3    "rules": {  
4      ".read": false,  
5      ".write": false  
6    }  
7  }.
```

The screenshot shows the Firebase Rules playground interface. It displays two different security rule configurations:

```
unpublished changes | Publish | Discard | Rules playground
```

Default security rules are locked from access

Top Configuration:

```
1  {
2    /* Visit https://firebase.google.com/docs/database/security to learn more about security rules. */
3    "rules": {
4      ".read": true,
5      ".write": true
6    }
7 }
```

Bottom Configuration:

```
1  {
2    /* Visit https://firebase.google.com/docs/database/security to learn more about security rules. */
3    "rules": {
4      ".read": true,
5      ".write": false
6    }
7 }
```

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">
  <uses-permission 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="My Application"
    android:roundIcon="@mipmap/ic_launcher_round"
    android:supportsRtl="true"
    android:theme="@style/Theme.MyApplication">
    <activity
        android:name=".MainActivity"
        android:exported="true">
        <intent-filter>
            <action android:name="android.intent.action.MAIN" />
        </intent-filter>
    </activity>
  </application>
</manifest>
```

build.gradle.kts (app):

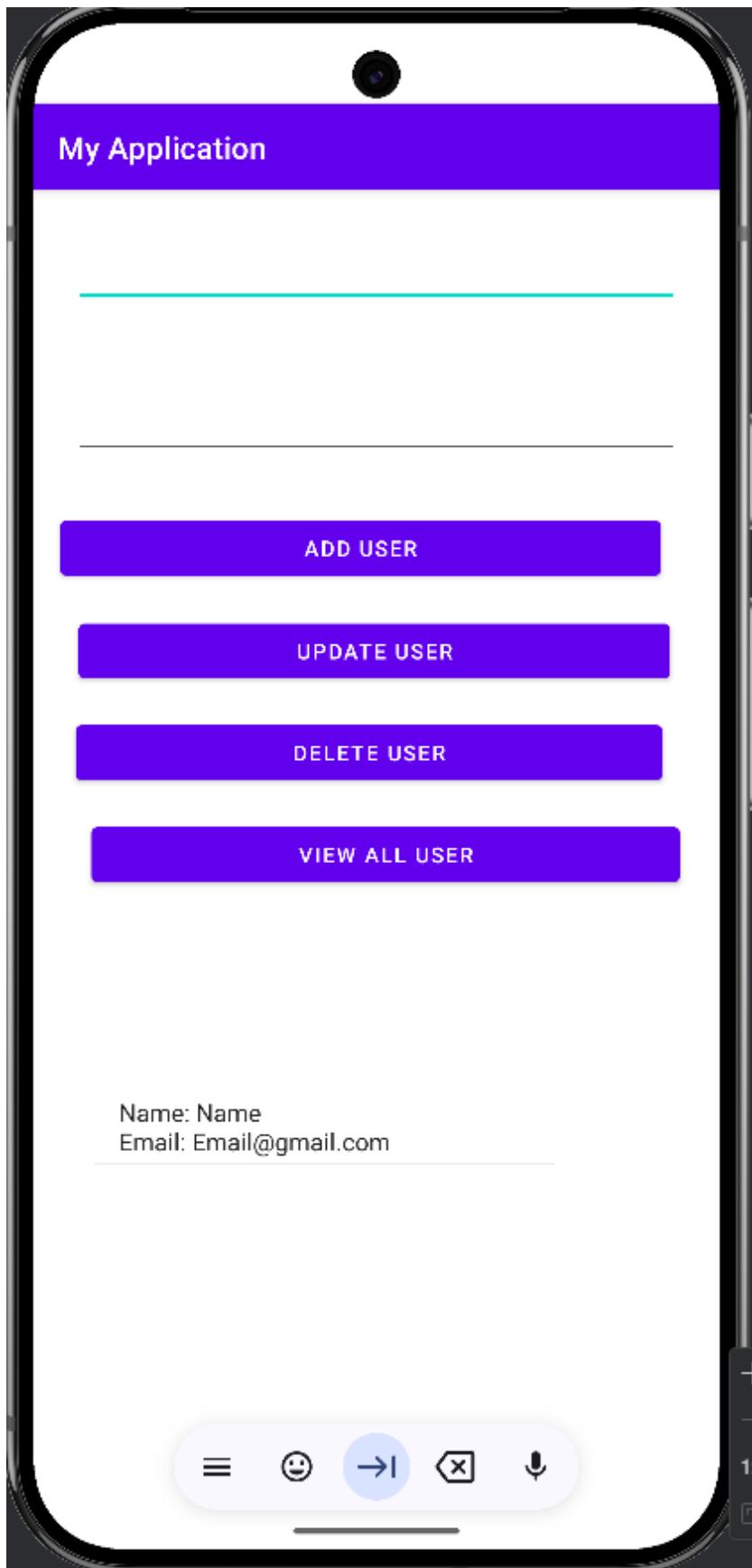
```
plugins {
    id("com.android.application")
    id("org.jetbrains.kotlin.android")
}

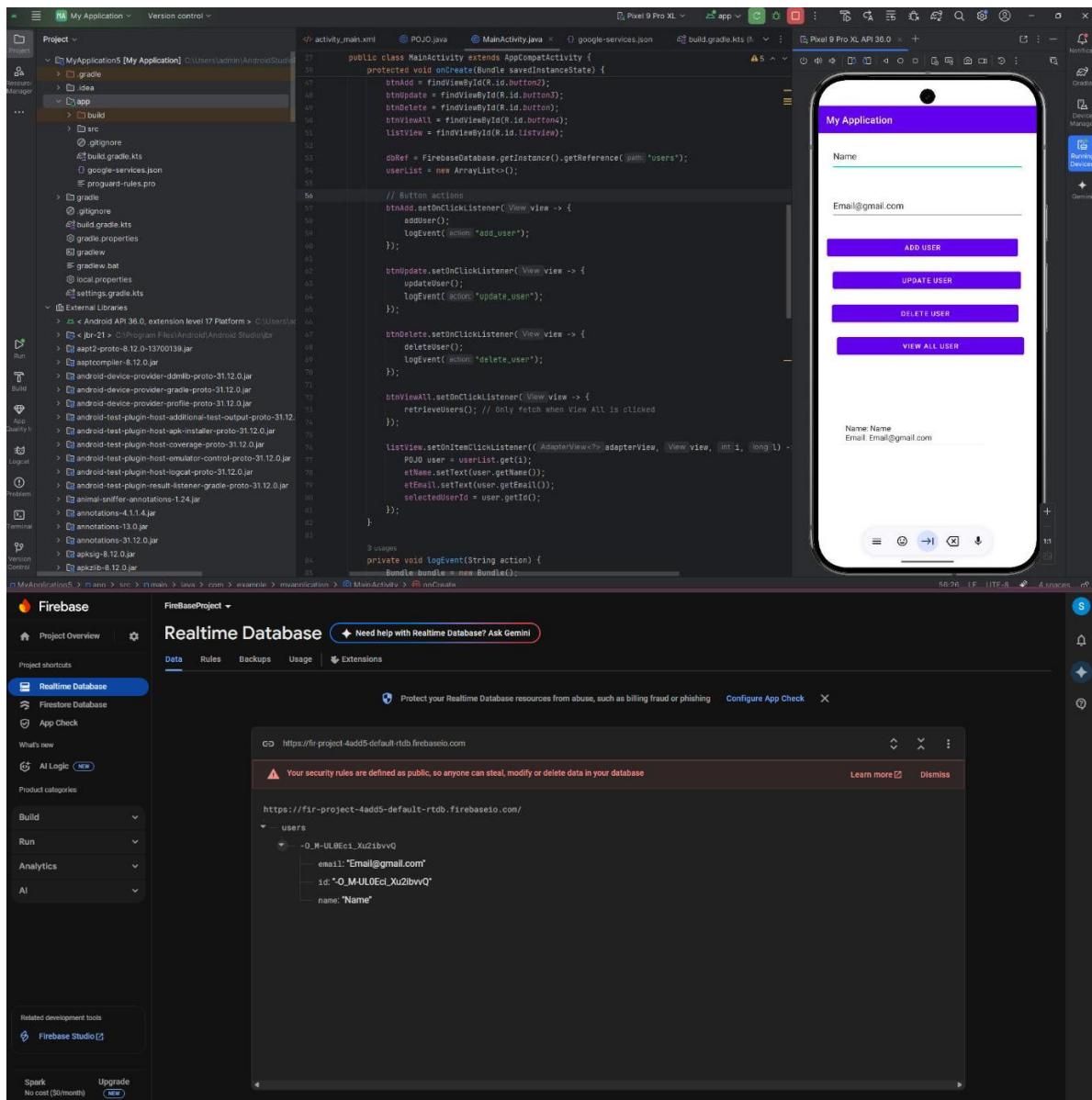
android {
    compileSdkVersion(33)
    targetSdkVersion(33)
    defaultConfig {
        applicationId = "com.example.myapplication"
        minSdkVersion(21)
        versionCode = 1
        versionName = "1.0"
        testInstrumentationRunner = "androidx.test.runner.AndroidJUnitRunner"
    }
    buildTypes {
        release {
            minifyEnabled(false)
            proguardFiles(proguardRulesFile)
        }
    }
}

dependencies {
    implementation("org.jetbrains.kotlin:kotlin-stdlib:1.7.10")
    implementation("com.google.firebase:firebase-database-ktx:22.1.0")
    testImplementation("junit:junit:4.13.2")
    androidTestImplementation("androidx.test.espresso:espresso-core:3.4.0")
}
```

Add this as well in App level build.gradle.kts

```
implementation("com.google.firebaseio:firebase-database")
```





Module 3 Animation, Multimedia and Location Based Services:

- 1. Write an Android application to play, pause, and stop an audio file.**

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:gravity="bottom|center_horizontal"
    android:orientation="horizontal"
    android:background="@drawable/music"
    tools:context=".MainActivity">

    <Button
        android:id="@+id	btnPlay"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_margin="10dp"
        android:backgroundTint="#C6A8A8"
        android:text="Play"
        android:textColor="#01579B" />

    <Button
        android:id="@+id	btnPause"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_margin="10dp"
        android:backgroundTint="#C6A8A8"
        android:text="Pause"
        android:textColor="#01579B" />

    <Button
        android:id="@+id	btnStop"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_margin="10dp"
        android:backgroundTint="#C6A8A8"
        android:text="Stop"
        android:textColor="#01579B" />

</LinearLayout>
```

MainActivity.java

```
package com.example.myapplication;
```

```
import androidx.appcompat.app.AppCompatActivity;
import android.media.MediaPlayer;
import android.os.Bundle;
import android.view.View;
import android.widget.Button;

public class MainActivity extends AppCompatActivity {
    Button play,pause,stop;
    MediaPlayer mp;
    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_main);
        play=findViewById(R.id.btnPlay);
        pause=findViewById(R.id.btnPause);
        stop=findViewById(R.id.btnStop);
        play.setOnClickListener(new View.OnClickListener() {
            @Override
            public void onClick(View v) {
                if(mp==null)
                {
                    mp=MediaPlayer.create(getApplicationContext(),R.raw.song);
                    mp.start();
                }
            }
        });
        pause.setOnClickListener(new View.OnClickListener() {
            @Override
            public void onClick(View v) {
                if(mp!=null)
                {
                    mp.pause();
                }
            }
        });
        stop.setOnClickListener(new View.OnClickListener() {
            @Override
            public void onClick(View v) {
                if(mp!=null)
                {
                    mp.release();
                    mp=null;
                }
            }
        });
    }
}
```

Ouput :



2. Write an Android application to play a video with Media controller.

activity_main.xml

```
<?xml version="1.0" encoding="utf-8"?>
<FrameLayout 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:id="@+id/video_view"
        android:layout_width="match_parent"
        android:layout_marginTop="80dp"
        android:layout_height="match_parent" />
</FrameLayout>
```

MainActivity.java

```
package com.example.myapplication;

import androidx.appcompat.app.AppCompatActivity;
import android.net.Uri;
import android.os.Bundle;
import android.widget.MediaController;
import android.widget.VideoView;

public class MainActivity extends AppCompatActivity {
    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_main);
        VideoView videoView = findViewById(R.id.video_view);
        //Storing video resource in string variable
        String videoPath = "android.resource://" + getPackageName() + "/" + R.raw.video;
        Uri uri = Uri.parse(videoPath);
        videoView.setVideoURI(uri);

        MediaController mediaController = new MediaController(this);
        videoView.setMediaController(mediaController);
        mediaController.setAnchorView(videoView);
    }
}
```

Output :



3. **Create an android application that applies different animations on**

an image.

activity_main.xml

```
<?xml version="1.0" encoding="utf-8"?>
<GridLayout 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="wrap_content"
    android:layout_height="wrap_content"
    android:columnCount="4"
    android:layout_gravity="center"
    tools:context=".MainActivity">

    <!-- this layout holds image which won't
        overflow out of this frame during animations -->
    <FrameLayout
        android:padding="5dp"
        android:layout_gravity="center"
        android:layout_columnSpan="4">

        <ImageView
            android:layout_width="180dp"
            android:layout_height="180dp"
            android:id="@+id/bird"
            android:src="@drawable/bird" />

    </FrameLayout>
    <TextView
        style="@style/animation_title_props"
        android:text="Rotate Animations" />

    <Button
        style="@style/btn_medium_props"
        android:text="Clockwise"
        android:onClick="clockwise"/>

    <Button
        style="@style/btn_medium_props"
        android:text="Anti Clockwise"
        android:onClick="antiClockwise"
        android:layout_width="170dp"/>

    <TextView
        style="@style/animation_title_props"
        android:text="Scale Animations" />

    <Button
        style="@style/btn_medium_props"
        android:text="Expand"
        android:onClick="expand"/>
```

```
<Button
    style="@style/btn_medium_props"
    android:text="Shrink"
    android:onClick="shrink"/>
<TextView
    style="@style/animation_title_props"
    android:text="Translate Animations" />

<Button
    style="@style/btn_min_props"
    android:text="L2R"
    android:onClick="slideL2R"/>

<Button
    style="@style/btn_min_props"
    android:text="T2B"
    android:onClick="slideT2B"/>

<Button
    style="@style/btn_min_props"
    android:text="R2L"
    android:onClick="slideR2L"/>

<Button
    style="@style/btn_min_props"
    android:text="B2T"
    android:onClick="slideB2T"/>

<TextView
    style="@style/animation_title_props"
    android:text="Alpha Animations" />

<Button
    style="@style/btn_medium_props"
    android:text="Fade In"
    android:onClick="fadeIn"/>
<Button
    style="@style/btn_medium_props"
    android:text="Fade Out"
    android:onClick="fadeOut"/>

<TextView
    style="@style/animation_title_props"
    android:text="Mix Animations" />

<Button
    style="@style/btn_medium_props"
    android:text="Expand with Rotation"
    android:onClick="expandWithRotation"/>
```

```
<Button  
    style="@style/btn_medium_props"  
    android:text="Slide with Fade In"  
    android:onClick="slideWithFadeIn"/>  
  
<TextView  
    style="@style/animation_title_props"  
    android:text="Clear Animation" />  
  
<Button  
    style="@style/btn_max_props"  
    android:text="Clear Animation"  
    android:onClick="clearAnimation"/>  
  
</GridLayout>
```

MainActivity.java

```
package com.example.myapplication;  
  
import androidx.appcompat.app.AppCompatActivity;  
import android.os.Bundle;  
import android.view.View;  
import android.view.animation.Animation;  
import android.view.animation.AnimationUtils;  
import android.widget.ImageView;  
import android.widget.Toast;  
  
public class MainActivity extends AppCompatActivity {  
    ImageView bird;  
    Animation animation;  
    @Override  
    protected void onCreate(Bundle savedInstanceState) {  
        super.onCreate(savedInstanceState);  
        setContentView(R.layout.activity_main);  
        bird=findViewById(R.id.bird);  
    }  
  
    public void expand(View view) {  
        animation = AnimationUtils.loadAnimation(MainActivity.this, R.anim.expand);  
        bird.startAnimation(animation);  
        Toast.makeText(this, "Expanding...", Toast.LENGTH_SHORT).show();  
    }  
  
    public void shrink(View view) {  
        animation = AnimationUtils.loadAnimation(MainActivity.this, R.anim.shrink);  
        bird.startAnimation(animation);  
        Toast.makeText(this, "Shrinking Scaling...", Toast.LENGTH_SHORT).show();  
    }  
}
```

```

public void fadeIn(View view) {
    animation = AnimationUtils.loadAnimation(MainActivity.this, R.anim.fade_in);
    bird.startAnimation(animation);
    Toast.makeText(this, "Fading In...", Toast.LENGTH_SHORT).show();
}

public void fadeOut(View view) {
    animation = AnimationUtils.loadAnimation(MainActivity.this, R.anim.fade_out);
    bird.startAnimation(animation);
    Toast.makeText(this, "Fading Out...", Toast.LENGTH_SHORT).show();
}

public void expandWithRotation(View view) {
    animation = AnimationUtils.loadAnimation(MainActivity.this, R.anim.expand_with_rotation);
    bird.startAnimation(animation);
    Toast.makeText(this, "Expanding with Rotation...", Toast.LENGTH_SHORT).show();
}

public void clearAnimation(View view) {
    bird.clearAnimation();
    Toast.makeText(this, "Animation Cleared...", Toast.LENGTH_SHORT).show();
}
}

```

Themes.xml

```

<style name="animation_title_props">
    <item name="android:layout_columnSpan">4</item>
    <item name="android:textColor">@color/black</item>
    <item name="android:textSize">20sp</item>
    <item name="android:paddingTop">10dp</item>
</style>

<!-- small button properties -->
<style name="btn_min_props">
    <item name="android:layout_columnSpan">1</item>
    <item name="android:layout_width">80dp</item>
    <item name="android:layout_margin">1dp</item>
</style>

<!-- medium button properties -->
<style name="btn_medium_props">
    <item name="android:layout_columnSpan">2</item>
    <item name="android:layout_width">165dp</item>
    <item name="android:layout_margin">1dp</item>
</style>

<!-- large button properties -->
<style name="btn_max_props">
    <item name="android:layout_columnSpan">4</item>
    <item name="android:layout_width">wrap_content</item>
    <item name="android:layout_gravity">center</item>
</style>

```

Expand.xml

```
<?xml version="1.0" encoding="utf-8"?>
<set xmlns:android="http://schemas.android.com/apk/res/android">
    <scale
        android:fromXScale="0"
        android:toXScale="2"
        android:fromYScale="0"
        android:toYScale="2"
        android:pivotX="70%"
        android:pivotY="70%"
        android:repeatCount="infinite"
        android:duration="2000" />
</set>
```

Shrink.xml

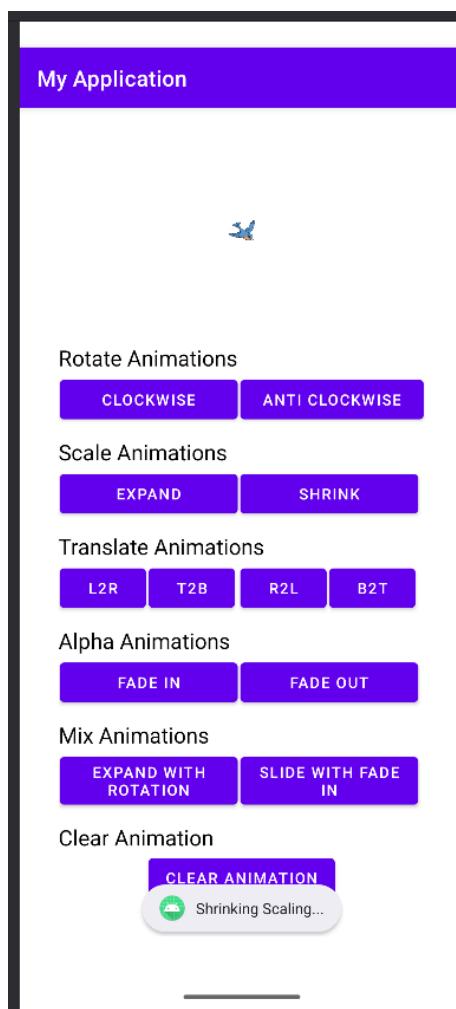
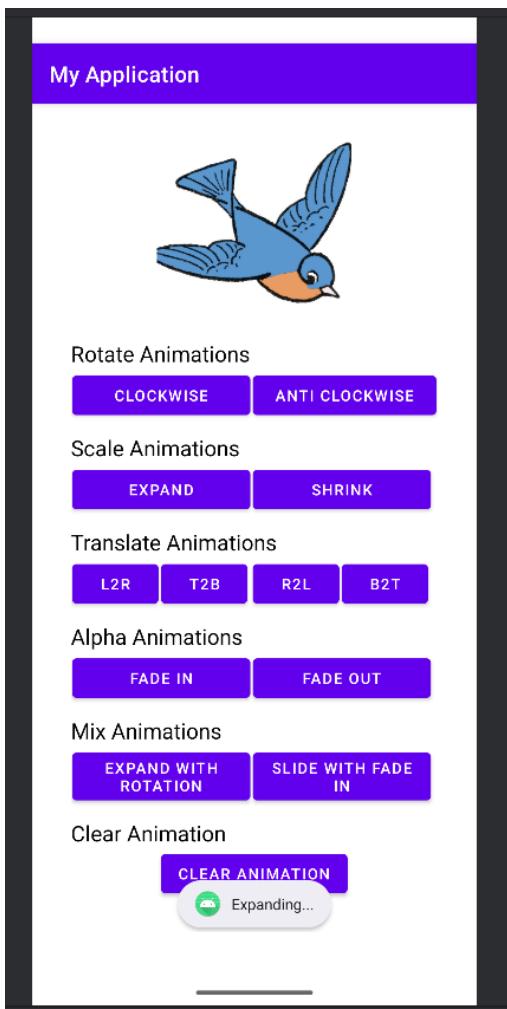
```
<?xml version="1.0" encoding="utf-8"?>
<set xmlns:android="http://schemas.android.com/apk/res/android">
    <scale
        android:fromXScale="1"
        android:toXScale="0"
        android:fromYScale="1"
        android:toYScale="0"
        android:pivotX="50%"
        android:pivotY="50%"
        android:duration="2000" />
</set>
```

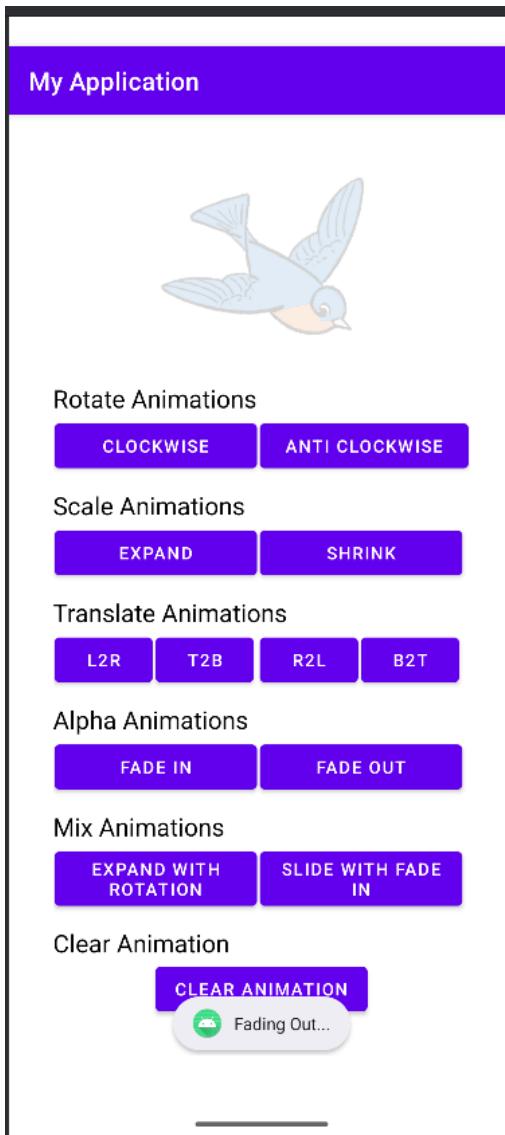
fadeln.xml

```
<?xml version="1.0" encoding="utf-8"?>
<set xmlns:android="http://schemas.android.com/apk/res/android">
    <alpha
        android:fromAlpha="0"
        android:toAlpha="1"
        android:duration="2000" />
</set>
```

fadeout.xml

```
<?xml version="1.0" encoding="utf-8"?>
<set xmlns:android="http://schemas.android.com/apk/res/android">
    <alpha
        android:fromAlpha="1"
        android:toAlpha="0"
        android:duration="2000" />
</set>
```

Output :



4. Create an Android application to implement frame animation.

Running.xml

```
<?xml version="1.0" encoding="utf-8"?>
<animation-list xmlns:android="http://schemas.android.com/apk/res/android">
    <item android:drawable="@drawable/one" android:duration="100" />
    <item android:drawable="@drawable/two" android:duration="100" />
    <item android:drawable="@drawable/three" android:duration="100" />
    <item android:drawable="@drawable/four" android:duration="100" />
</animation-list>
```

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"
    android:gravity="center"
    tools:context=".MainActivity">

    <!-- image view to hold animation frame images -->
    <ImageView
        android:id="@+id/img"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:src="@drawable/running" />

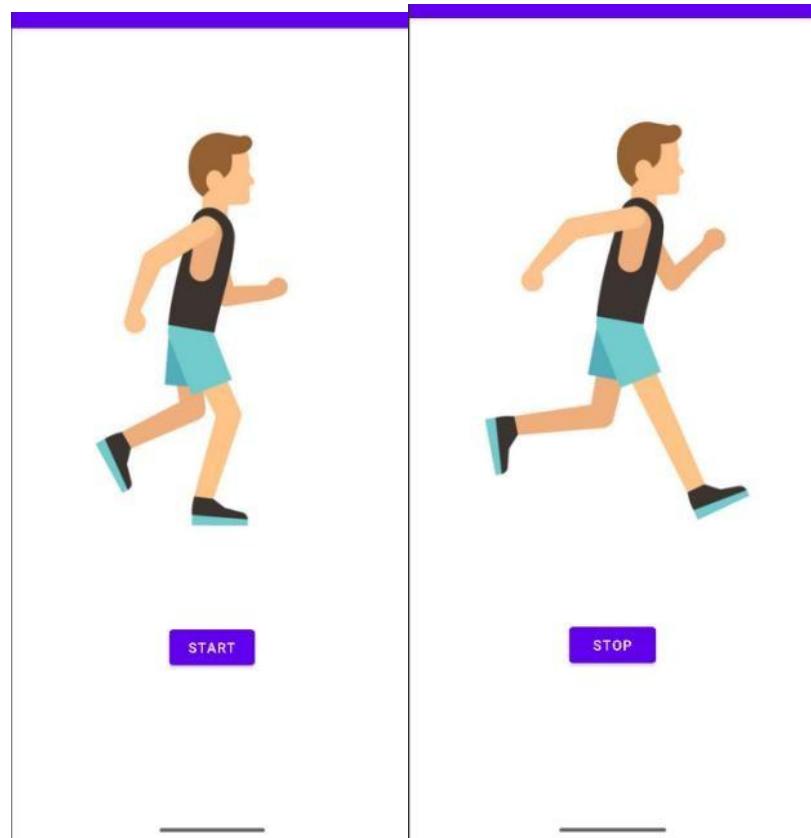
    <!-- button to start or pause animation -->
    <Button
        android:id="@+id/btn_start_stop"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="Start" />
</LinearLayout>
```

MainActivity.java

```
import androidx.appcompat.app.AppCompatActivity;
import android.graphics.drawable.AnimationDrawable;
import android.os.Bundle;
import android.widget.Button;
import android.widget.ImageView;

public class MainActivity extends AppCompatActivity {
    ImageView img;
```

```
Button btnStartStop;
AnimationDrawable animation;
@Override
protected void onCreate(Bundle savedInstanceState) {
    super.onCreate(savedInstanceState);
    setContentView(R.layout.activity_main);
    img = findViewById(R.id.img);
    btnStartStop = findViewById(R.id.btn_start_stop);
    // fetching animation drawable from image view
    animation = (AnimationDrawable) img.getDrawable();
    // starting and stopping animation on button click
    btnStartStop.setOnClickListener(view -> {
        // if running then stop
        if (animation.isRunning()) {
            animation.stop();
            btnStartStop.setText("Start");
            return;
        }
        // else start animation
        animation.start();
        btnStartStop.setText("Stop");
    });
}
```

Output :

5. **Create an Android application to display the current location of your device (display longitude and latitude values).**

AndroidManifest.xml:

```
<uses-permission android:name="android.permission.ACCESS_FINE_LOCATION" />
<uses-permission android:name="android.permission.ACCESS_COARSE_LOCATION" />
<uses-permission android:name="android.permission.INTERNET" />
```

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:orientation="vertical" android:padding="16dp">

    <TextView android:id="@+id/latitude"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_marginTop="20dp"
        android:text="Latitude:"
        android:textSize="24sp"/>

    <TextView android:id="@+id/longitude"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_marginTop="20dp"
        android:text="Longitude:"
        android:textSize="24sp"/>

    <Button
        android:id="@+id/fetch_location_button"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_marginTop="20dp"
        android:text="Fetch Coordinates" />

</LinearLayout>
```

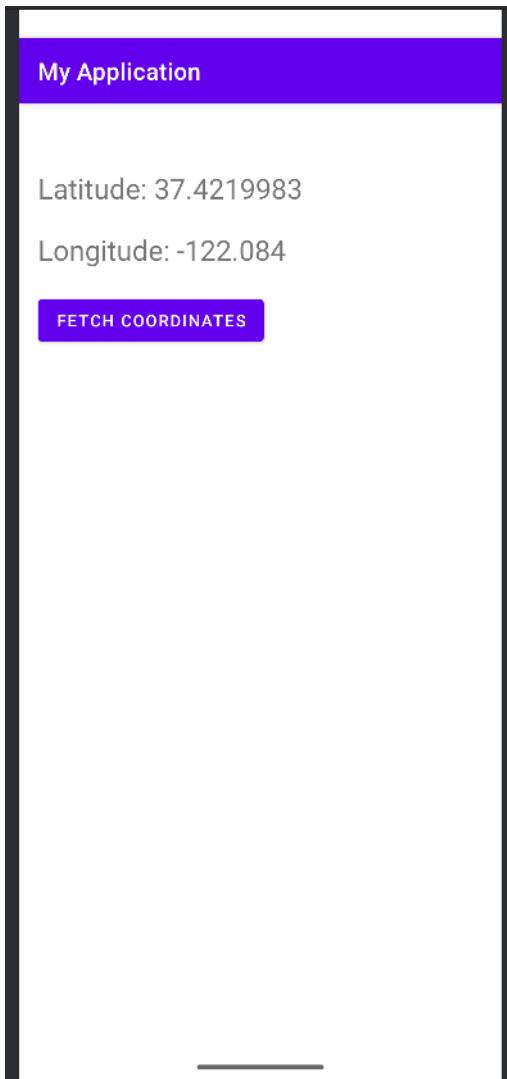
MainActivity.java

```
package com.example.myapplication;

import android.Manifest;
import android.content.pm.PackageManager;
import android.os.Bundle;
import android.widget.Button;
```

```
import android.widget.TextView;  
  
import androidx.activity.EdgeToEdge;  
import androidx.appcompat.app.AppCompatActivity;  
import androidx.core.app.ActivityCompat;  
import androidx.core.graphics.Insets;  
import androidx.core.view.ViewCompat;  
import androidx.core.view.WindowInsetsCompat;  
  
import com.google.android.gms.location.FusedLocationProviderClient;  
import com.google.android.gms.location.LocationServices;  
  
public class MainActivity extends AppCompatActivity {  
  
    TextView latitudeText, longitudeText;  
    FusedLocationProviderClient fusedLocationClient;  
  
    @Override  
    protected void onCreate(Bundle savedInstanceState) {  
        super.onCreate(savedInstanceState);  
        setContentView(R.layout.activity_main);  
  
        latitudeText = findViewById(R.id.latitude);  
        longitudeText = findViewById(R.id.longitude);  
        fusedLocationClient = LocationServices.getFusedLocationProviderClient(this);  
  
        Button fetchButton = findViewById(R.id.fetch_location_button);  
        fetchButton.setOnClickListener(v -> fetchCoordinates());  
    }  
  
    private void fetchCoordinates() {  
        if (ActivityCompat.checkSelfPermission(this,  
Manifest.permission.ACCESS_FINE_LOCATION) !=  
PackageManager.PERMISSION_GRANTED) {  
            ActivityCompat.requestPermissions(this, new  
String[]{Manifest.permission.ACCESS_FINE_LOCATION}, 1);  
            return;  
        }  
  
        fusedLocationClient.getLastLocation()  
        .addOnSuccessListener(this, location -> {  
            if (location != null) {  
                latitudeText.setText("Latitude: " + location.getLatitude());  
                longitudeText.setText("Longitude: " + location.getLongitude());  
            }  
        });  
    }  
}
```

Output:



6. **Create an Android application that displays the current location of your device from longitude and latitude values (Reverse Geocoding).**

Main activity.java

```
package com.example.myapplication;

import android.Manifest;
import android.content.pm.PackageManager;
import android.location.Address;
import android.location.Geocoder;
import android.os.Bundle;
import android.widget.Button;
import android.widget.TextView;

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

import com.google.android.gms.location.FusedLocationProviderClient;
import com.google.android.gms.location.LocationServices;

import java.io.IOException;
import java.util.List;
import java.util.Locale;

public class MainActivity extends AppCompatActivity {

    TextView addressText;
    FusedLocationProviderClient fusedLocationClient;

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

        addressText = findViewById(R.id.address);
        fusedLocationClient = LocationServices.getFusedLocationProviderClient(this);

        Button fetchButton = findViewById(R.id.fetch_address_button);
        fetchButton.setOnClickListener(v -> fetchAddress());
    }

    private void fetchAddress() {
        if (ActivityCompat.checkSelfPermission(this,
                Manifest.permission.ACCESS_FINE_LOCATION) != PackageManager.PERMISSION_GRANTED) {
```

```
ActivityCompat.requestPermissions(this, new
String[]{Manifest.permission.ACCESS_FINE_LOCATION}, 1);
return;
}
fusedLocationClient.getLastLocation()
.addOnSuccessListener(this, location -> {
if (location != null) {
    Geocoder geocoder = new Geocoder(this, Locale.getDefault());
    try {
        List<Address> addresses = geocoder.getFromLocation(location.getLatitude(),
location.getLongitude(), 1);
        if (addresses != null && !addresses.isEmpty()) {
            String address = addresses.get(0).getAddressLine(0);
            addressText.setText("Address: " + address);
        } else {
            addressText.setText("No address found.");
        }
    } catch (IOException e) {
        e.printStackTrace();
        addressText.setText("Error: " + e.getMessage());
    }
}
});
```

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:orientation="vertical" android:padding="16dp">

    <TextView android:id="@+id/address"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_marginTop="50dp"
        android:text="Address will appear here"
        android:textSize="18sp"/>

    <Button
        android:id="@+id/fetch_address_button"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_marginTop="24sp"
        android:text="Fetch Address" />

</LinearLayout>
```

Output :

My Application

Address: 1600 Amphitheatre Pkwy, Mountain
View, CA 94043, USA

FETCH ADDRESS

Module 4 REST API integration:

1. **Create an Android application to demonstrate JSON data parsing using OkHttp (you can use <https://api.github.com/users> JSON data).**

activity_main.xml

```
<?xml version="1.0" encoding="utf-8"?>
<ScrollView
    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">
    <LinearLayout
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:layout_marginTop="80dp"
        android:orientation="vertical">

        <Button
            android:id="@+id	btnFetch"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:layout_marginTop="30dp"
            android:layout_gravity="center_horizontal"
            android:text="Fetch data"/>
        <TextView
            android:id="@+id/result_view"
            android:layout_width="match_parent"
            android:layout_height="wrap_content"
            android:paddingHorizontal="16dp" />
    </LinearLayout>
</ScrollView>
```

MainActivity.java

```
package com.example.myapplication;

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

import android.os.Bundle;
import android.widget.Button;
import android.widget.TextView;
import android.widget.Toast;

import com.google.gson.Gson;
```

```
import java.io.IOException;

import okhttp3.Call;
import okhttp3.Callback;
import okhttp3.OkHttpClient;
import okhttp3.Request;
import okhttp3.Response;

public class MainActivity extends AppCompatActivity {

    TextView resultView;
    Button fetch;
    OkHttpClient client;

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

        resultView = findViewById(R.id.result_view);
        fetch = findViewById(R.id.btnFetch);
        client = new OkHttpClient();

        fetch.setOnClickListener(view -> {
            getWebService();
        });
    }

    private void getWebService() {
        String url = "https://reqres.in/api/users/1";

        Request request = new Request.Builder().url(url).build();

        client.newCall(request).enqueue(new Callback() {
            @Override
            public void onResponse(@NonNull Call call, @NonNull Response response) throws IOException {
                if (response.isSuccessful()) {
                    String jsonResponse = response.body().string();

                    // Parse JSON using Gson
                    Gson gson = new Gson();
                    UserResponse userResponse = gson.fromJson(jsonResponse, UserResponse.class);

                    String display = "ID: " + userResponse.data.id + "\n"
                            + "Name: " + userResponse.data.first_name + " " + userResponse.data.last_name
                            + "\n"
                            + "Email: " + userResponse.data.email + "\n"
                            + "Avatar: " + userResponse.data.avatar;

                    runOnUiThread(() -> resultView.setText(display));
                }
            }
        });
    }
}
```

```
        } else {
            runOnUiThread(() -> Toast.makeText(MainActivity.this, "API Error",
Toast.LENGTH_SHORT).show());
        }
    }

    @Override
    public void onFailure(@NonNull Call call, @NonNull IOException e) {
        runOnUiThread(() -> Toast.makeText(MainActivity.this, "Failed: " + e.getMessage(),
Toast.LENGTH_SHORT).show());
    }
);

}

// Model classes to map the JSON structure
public class UserResponse {
    public Data data;
}

public class Data {
    public int id;
    public String email;
    public String first_name;
    public String last_name;
    public String avatar;
}
}
```

Output :

My Application

FETCH DATA

ID: 1
Name: George Bluth
Email: george.bluth@reqres.in
Avatar: <https://reqres.in/img/faces/1-image.jpg>

2. **Create an Android application to demonstrate JSON data parsing using Volley (you can use <https://api.github.com/users> JSON data).**

activity_main.xml

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

    <ListView
        android:id="@+id/listView"
        android:layout_width="match_parent"
        android:layout_height="match_parent"
        android:textStyle="bold"

        android:layout_marginTop="90dp"/>

</LinearLayout>
```

MainActivity.java

```
package com.example.myapplication;

import android.os.Bundle;
import android.widget.ArrayAdapter;
import android.widget.ListView;
import android.widget.Toast;

import androidx.appcompat.app.AppCompatActivity;

import com.android.volley.Request;
import com.android.volley.RequestQueue;
import com.android.volley.toolbox.JsonArrayRequest;
import com.android.volley.toolbox.Volley;

import org.json.JSONArray;
import org.json.JSONException;
import org.json.JSONObject;

import java.util.ArrayList;

public class MainActivity extends AppCompatActivity {
```

```
private ListView listView;
private ArrayList<String> usernames;
private ArrayAdapter<String> adapter;

private static final String URL = "https://api.github.com/users";

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

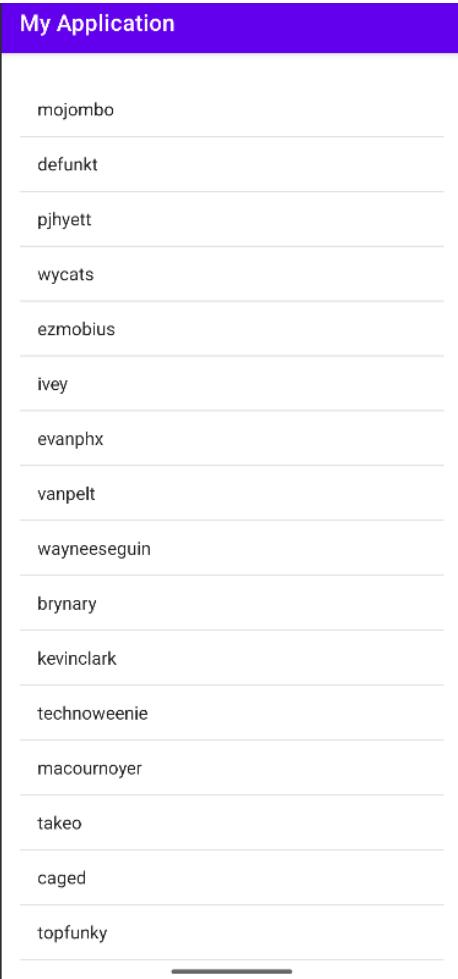
    listView = findViewById(R.id.listView);
    usernames = new ArrayList<>();
    adapter = new ArrayAdapter<>(this, android.R.layout.simple_list_item_1, usernames);
    listView.setAdapter(adapter);

    fetchGitHubUsers();
}

private void fetchGitHubUsers() {
    RequestQueue queue = Volley.newRequestQueue(this);

    JsonArrayRequest jsonArrayRequest = new JsonArrayRequest(Request.Method.GET, URL, null,
        response -> {
            try {
                for (int i = 0; i < response.length(); i++) {
                    JSONObject userObject = response.getJSONObject(i);
                    String login = userObject.getString("login");
                    usernames.add(login);
                }
                adapter.notifyDataSetChanged();
            } catch (JSONException e) {
                Toast.makeText(this, "Parsing error!", Toast.LENGTH_SHORT).show();
            }
        },
        error -> Toast.makeText(this, "Volley error!", Toast.LENGTH_SHORT).show()
    );
    queue.add(jsonArrayRequest);
}
}
```

Output :



3. **Create an Android application to demonstrate JSON data parsing using Retrofit (you can use <https://api.github.com/users> JSON data).**

activity_main.xml

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

    <TextView
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="Using Retrofit"
        android:layout_gravity="center_horizontal"
        android:textSize="16sp"
        android:layout_marginTop="90dp"

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

    <ListView
        android:id="@+id/listView"
        android:layout_width="match_parent"
        android:layout_height="match_parent"
        android:layout_marginTop="30dp"/>

</LinearLayout>
```

MainActivity.java

```
package com.example.myapplication;

import android.os.Bundle;
import android.widget.ArrayAdapter;
import android.widget.ListView;
import android.widget.Toast;

import androidx.appcompat.app.AppCompatActivity;

import java.util.List;
import java.util.ArrayList;

import retrofit2.Retrofit;
import retrofit2.converter.gson.GsonConverterFactory;
import retrofit2.Call;
import retrofit2.Callback;
import retrofit2.Response;
```

```
import retrofit2.http.GET;

public class MainActivity extends AppCompatActivity {

    private ListView listView;
    private ArrayAdapter<String> adapter;
    private ArrayList<String> usernames = new ArrayList<>();

    // --- Step 1: Model class ---
    public class GitHubUser {
        private String login;

        public String getLogin() {
            return login;
        }
    }

    // --- Step 2: API interface ---
    public interface GitHubApi {
        @GET("users")
        Call<List<GitHubUser>> getUsers();
    }

    // --- Step 3: onCreate ---
    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_main);
        listView = findViewById(R.id.listView);

        adapter = new ArrayAdapter<>(this, android.R.layout.simple_list_item_1, usernames);
        listView.setAdapter(adapter);

        fetchGitHubUsers();
    }

    // --- Step 4: Retrofit Call ---
    private void fetchGitHubUsers() {
        Retrofit retrofit = new Retrofit.Builder()
            .baseUrl("https://api.github.com/") // NOTE: ends with /
            .addConverterFactory(GsonConverterFactory.create())
            .build();

        GitHubApi api = retrofit.create(GitHubApi.class);

        Call<List<GitHubUser>> call = api.getUsers();
        call.enqueue(new Callback<List<GitHubUser>>() {
            @Override
            public void onResponse(Call<List<GitHubUser>> call, Response<List<GitHubUser>> response) {
                if (!response.isSuccessful()) {
```

```
        Toast.makeText(MainActivity.this, "Error: " + response.code(),
Toast.LENGTH_SHORT).show();
    return;
}

List<GitHubUser> users = response.body();
for (GitHubUser user : users) {
    usernames.add(user.getLogin());
}
adapter.notifyDataSetChanged();
}

@Override
public void onFailure(Call<List<GitHubUser>> call, Throwable t) {
    Toast.makeText(MainActivity.this, "Failure: " + t.getMessage(),
Toast.LENGTH_SHORT).show();
}
);
}
}
}
```

Output :**My Application**

Using Retrofit

topfunky

anotherjesse

roland

lukas

fanvsfan

tomtt

railsjitsu

nitay

kevwil

KirinDave

jamesgolick

atmos

errfree

mojodna

bmizerany

Module 5 Introduction to Dart and Flutter:

1. **Write a Flutter program to demonstrate Text widget and its properties.**

Code:

```
import 'package:flutter/material.dart';

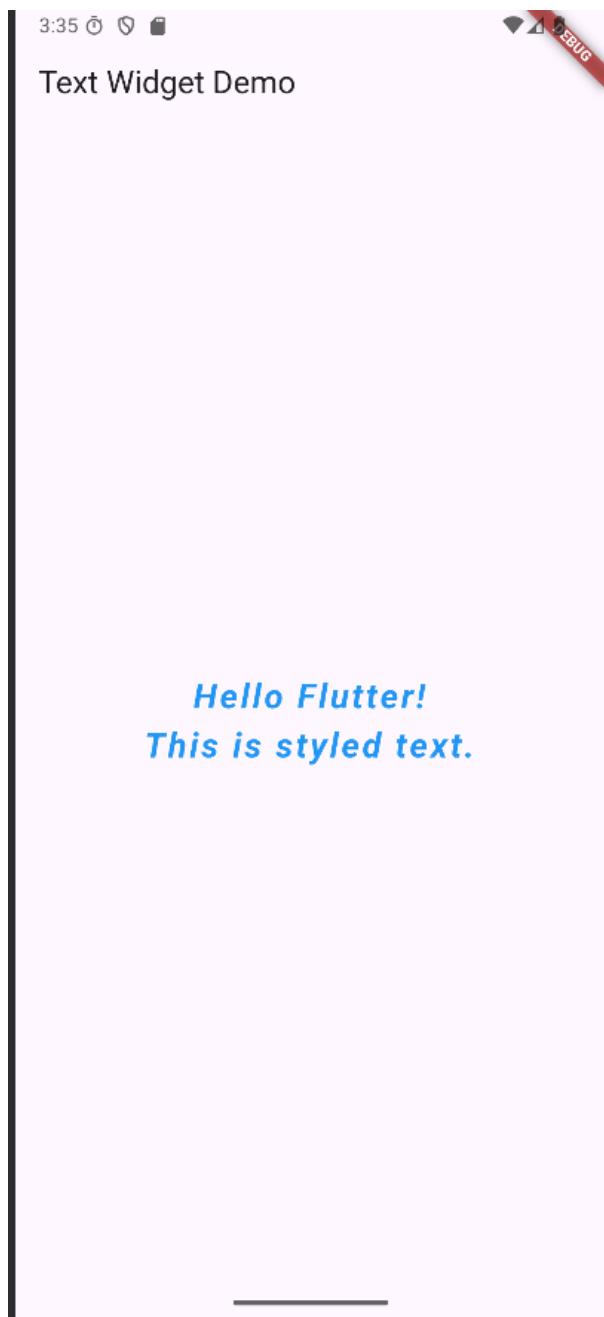
void main() => runApp(const MyApp());

class MyApp extends StatelessWidget {
    const MyApp({super.key});

    @override
    Widget build(BuildContext context) {
        return MaterialApp(
            home: Scaffold(
                appBar: AppBar(title: const Text('Text Widget Demo')),
                body: const Center(
                    child: Text(
                        'Hello Flutter!\nThis is styled text.',
                        textAlign: TextAlign.center,
                        maxLines: 2,
                        overflow: TextOverflow.ellipsis,
                        style: TextStyle(
                            color: Colors.blue,
                            fontSize: 24,
                            fontWeight: FontWeight.bold,
                            fontStyle: FontStyle.italic,
                            letterSpacing: 2,
                        ),
                )));
    }
}
```

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

Output:



2. **Write a Flutter program to display dog names (demonstrate stateless widget and column widgets).**

Code:

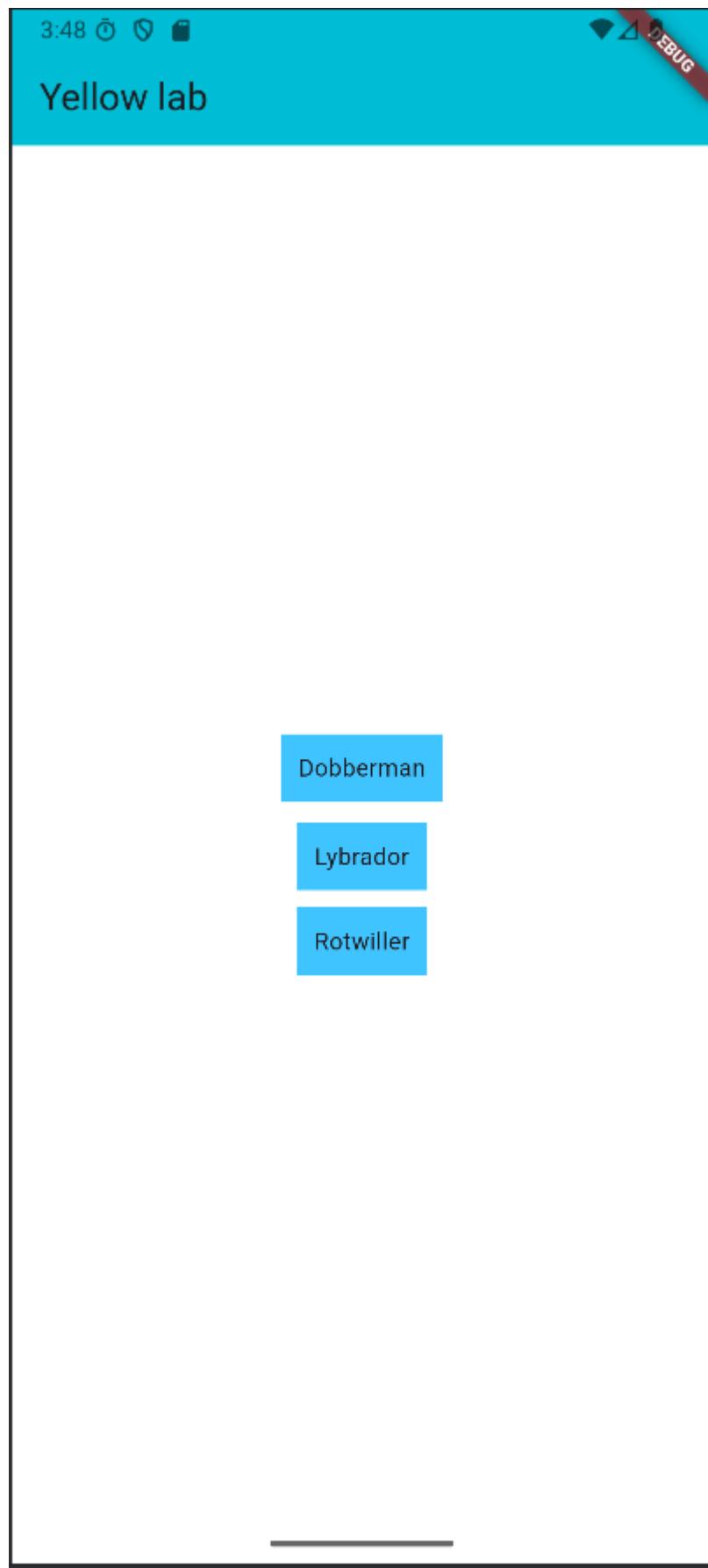
```
import 'package:flutter/material.dart';

void main() {
    runApp(DogApp());
}

class DogApp extends StatelessWidget {

    @override
    Widget build(BuildContext context) {
        return MaterialApp(
            title: 'My Dog App',
            home: Scaffold(backgroundColor: Colors.white,
                appBar: AppBar(backgroundColor: Colors.cyan,
                    title: Text('Yellow lab'),
                ),
                body: Center(
                    child: Column(
                        mainAxisAlignment: MainAxisAlignment.center,
                        children: [
                            DogName('Dobberman'),
                            SizedBox(height: 12.0),
                            DogName('Lybrador'),
                            SizedBox(height: 10.0),
                            DogName('Rotwiller'),
                        ],
                    ),
                ),
            );
    }
}

class DogName extends StatelessWidget {
    final String name;
    const DogName(this.name);
    @override
    Widget build(BuildContext context) {
        return DecoratedBox(decoration: BoxDecoration(color: Colors.lightBlueAccent),
            child: Padding(
                padding: const EdgeInsets.all(10.0),
                child: Text(name),
            ),
        );
    }
}
```

Output:

3. **Write a Flutter program that allows the user to enter a city in a text field and displays city name (demonstrate stateful widget).**

Code:

```
import 'package:flutter/material.dart';

void main() {
    runApp(FavouriteCity());
    /*runApp(
    MaterialApp(
        title: 'Stateful Application Example',
        home: FavouriteCity(),
    )
    );
    */
}

class FavouriteCity extends StatefulWidget {

    @override
    State<StatefulWidget> createState() {
        return _FavoriteCityState();
    }
}

class _FavoriteCityState extends State<FavouriteCity> {
    String nameCity="";
    @override
    Widget build(BuildContext context) {
        debugPrint('Favorite city widget is created.');
        return MaterialApp(
            title: 'Stateful Application Example',
            home:
            Scaffold(
                appBar: AppBar(
                    title: Text('Stateful Application Example'),
                ),
                body: Container(
                    margin: EdgeInsets.all(20.0),
                    child: Column(
                        children: <Widget>[
                            TextField(onSubmitted: (String userInput){
                                setState(() {
                                    debugPrint('setState is called. This tells framework to redraw the favorite city
widget. ');
                                    nameCity=userInput;
                                });
                            },),
                            Padding(padding: EdgeInsets.all(30.0),
                            child: Text(

```

```
        'Your best city is $nameCity',style: TextStyle(fontSize: 20.0),  
        )  
        ),  
        ],  
        ),  
        ),  
        );  
    }  
}
```

Output:

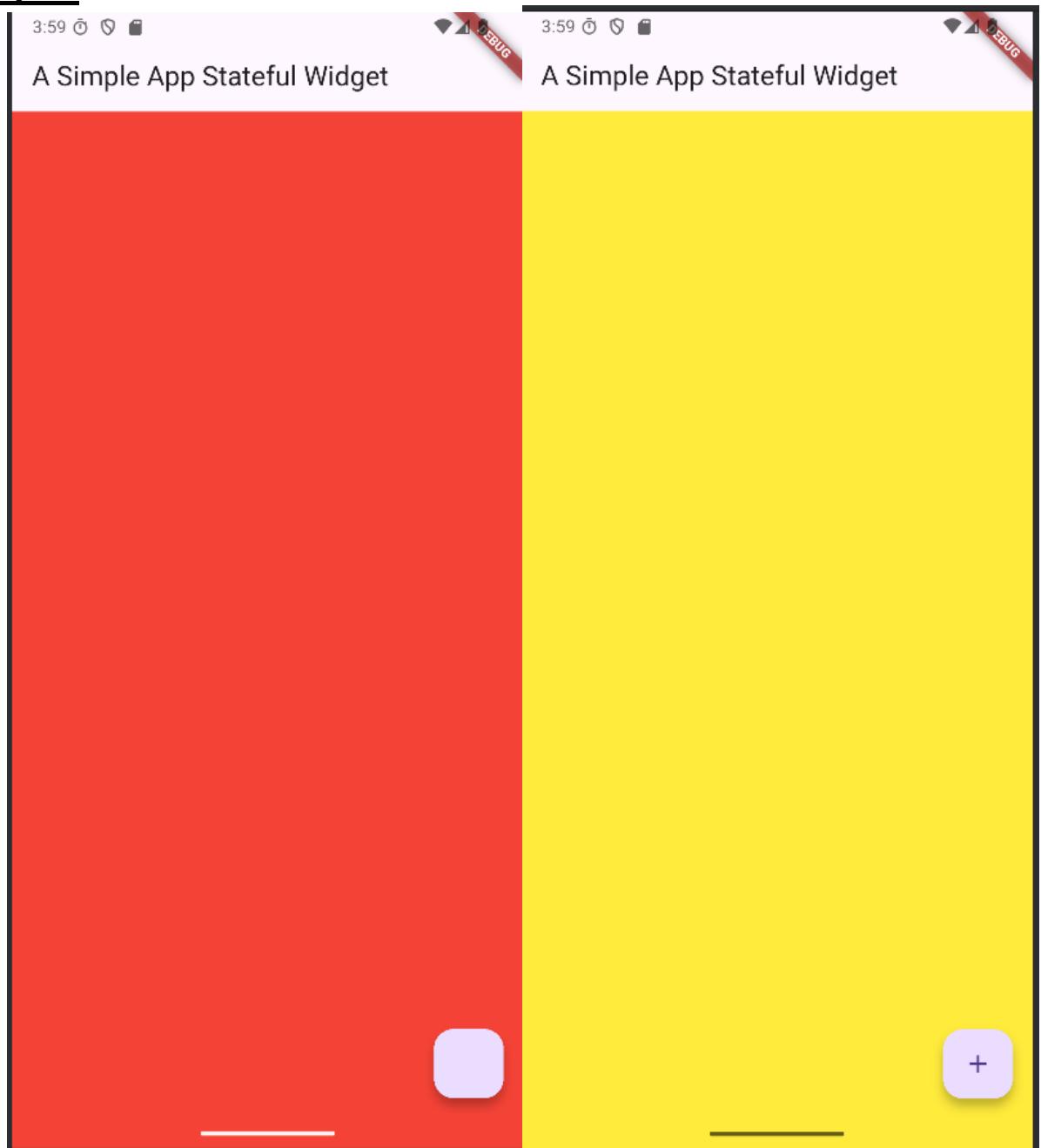


4. **Write a Flutter program to change the background color (demonstrate stateful widget).**

Code:

```
import 'package:flutter/material.dart';

void main() => runApp(MyApp());
class MyApp extends StatefulWidget {
  @override
  _MyState createState() => _MyState();
}
class _MyState extends State<MyApp> {
  Color _containerColor = Colors.yellow;
  void changeColor() {
    setState(() {
      if (_containerColor == Colors.yellow) {
        _containerColor = Colors.red;
        return;
      }
      _containerColor = Colors.yellow;
    });
  }
  @override
  Widget build(BuildContext context) {
    return MaterialApp(
      title: 'Flutter Demo',
      theme: ThemeData(
        primarySwatch: Colors.purple,
      ),
      home: Scaffold(
        appBar: AppBar(title: Text("A Simple App Stateful Widget")),
        body: Container(
          (
            decoration: BoxDecoration(color: _containerColor)
          ),
          floatingActionButton: FloatingActionButton(
            onPressed: changeColor,
            child: Icon(Icons.add),
            tooltip: "Book Here",
          ),
        )));
  }
}
```

Output:

5. Write a Flutter Program to display fruit list using ListView.

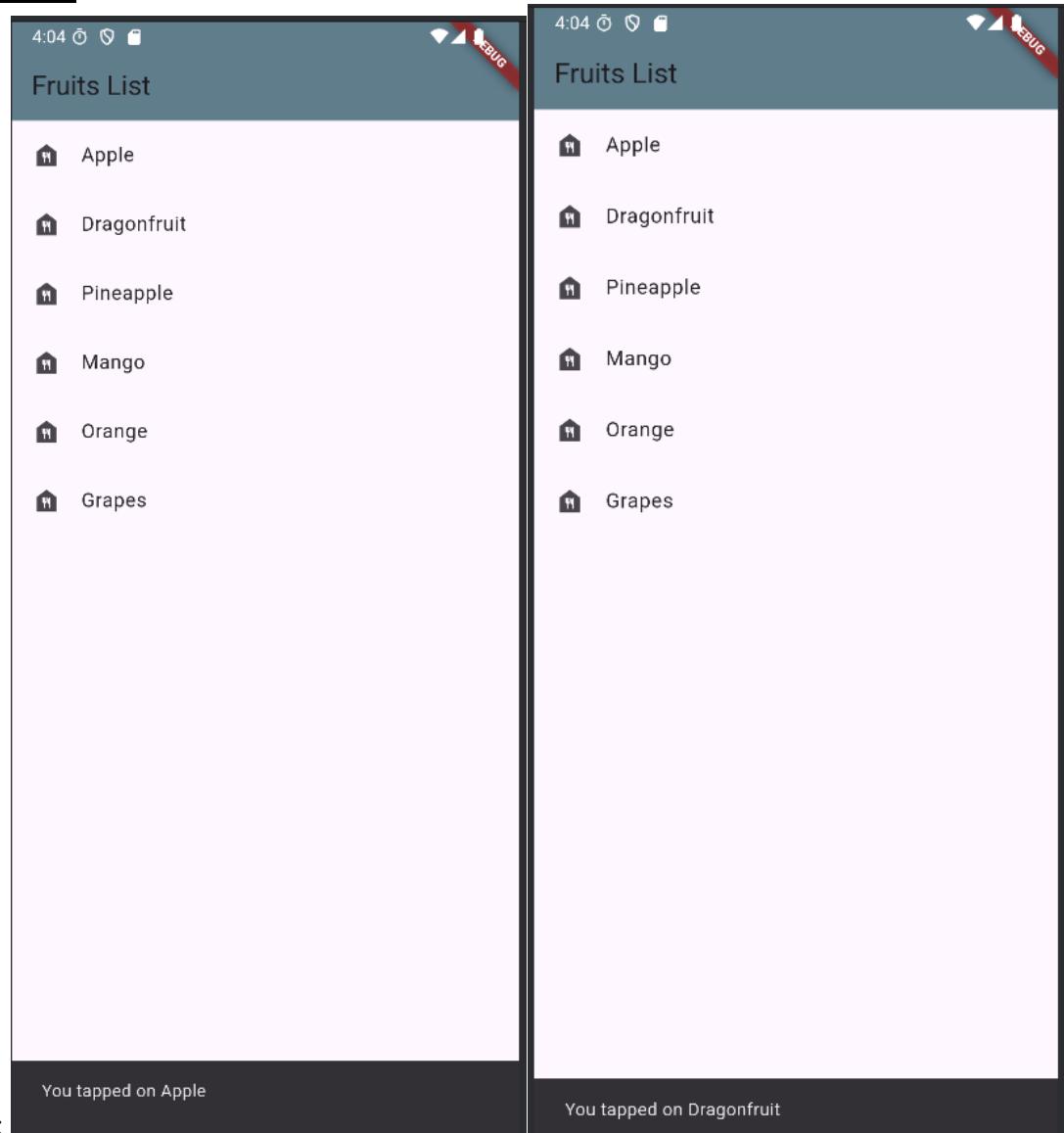
Code:

```
import 'package:flutter/material.dart';
void main() {
  runApp(const MyApp());
}
class MyApp extends StatelessWidget {
  const MyApp({super.key});
  @override
  Widget build(BuildContext context) {
    return MaterialApp(
      title: 'Flutter List Example',
      theme: ThemeData(primarySwatch: Colors.blueGrey),
      home: const MyHomePage(),
    );
  }
}
class MyHomePage extends StatelessWidget {
  const MyHomePage({super.key});

  // A simple List of strings
  final List<String> fruits = const [
    'Apple',
    'Dragonfruit',
    'Pineapple',
    'Mango',
    'Orange',
    'Grapes',
  ];
  @override
  Widget build(BuildContext context) {
    return Scaffold(
      appBar: AppBar(
        title: const Text("Fruits List"),
        backgroundColor: Colors.blueGrey,
      ),
      body: ListView.builder(
        itemCount: fruits.length, // total items
        itemBuilder: (context, index) {
          return ListTile(
            leading: const Icon(Icons.food_bank),
            title: Text(fruits[index]), // show list item
            onTap: () {
              // show selected item in snackbar
              ScaffoldMessenger.of(context).showSnackBar(
                SnackBar(content: Text("You tapped on ${fruits[index]}")),
              );
            },
          );
        },
      ),
    );
  }
}
```

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

Output:



6. **Write a Flutter program to demonstrate navigation (user should be navigated from first screen to second screen).**

Code:

```
import 'package:flutter/material.dart';
```

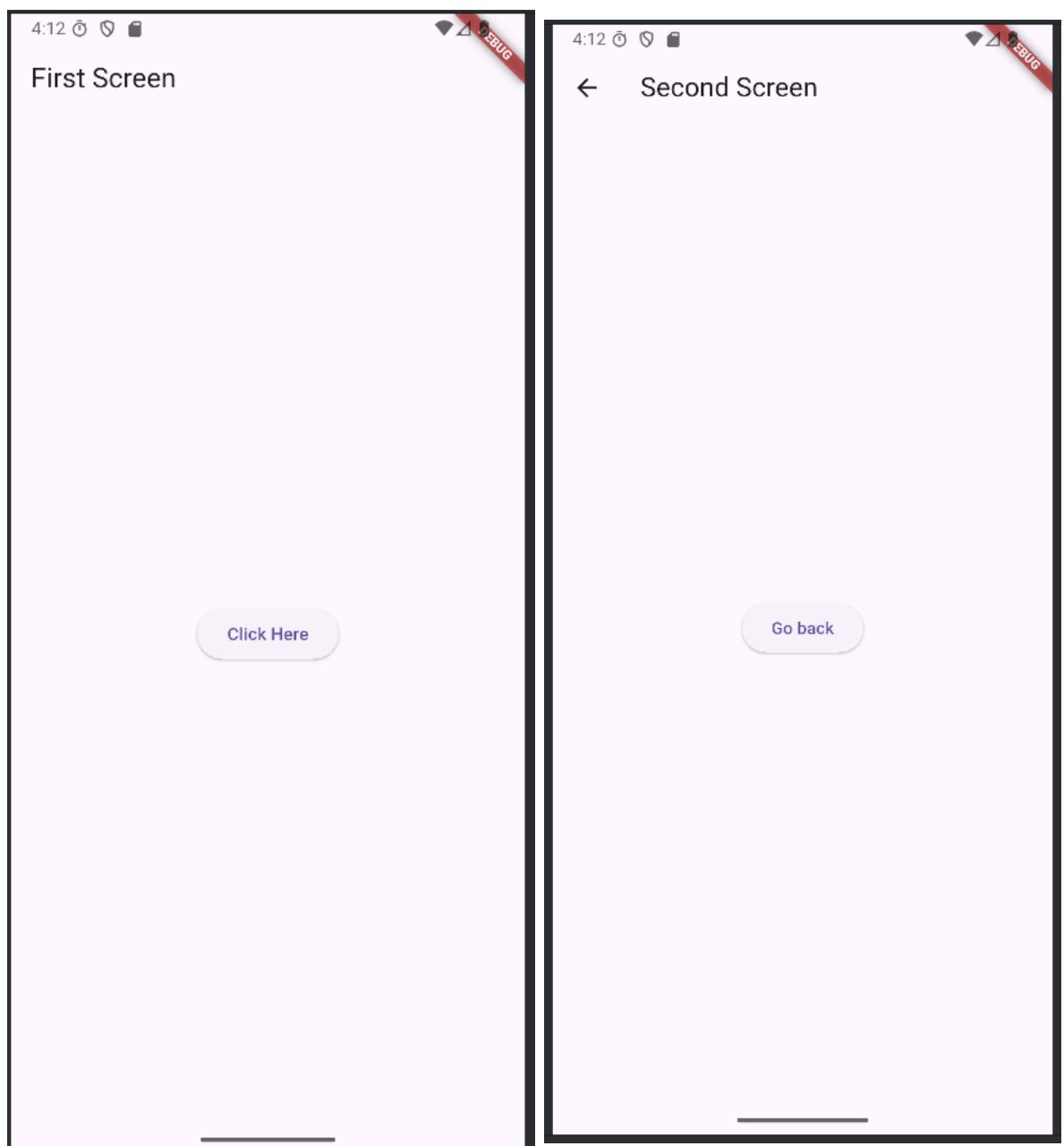
```
void main() {  
  
  runApp(MaterialApp(  
  
    title: 'Flutter Navigation',  
  
    theme: ThemeData(  
  
      // This is the theme of your application.  
  
      primarySwatch: Colors.green,  
  
    ),  
  
    home: FirstRoute(),  
  
  ));  
  
}
```

```
class FirstRoute extends StatelessWidget {  
  
  @override  
  
  Widget build(BuildContext context) {  
  
    return Scaffold(  
  
      appBar: AppBar(  
  
        title: Text('First Screen'),  
  
      ),  
  
      body: Center(  
  
        child: ElevatedButton(  
  
          child: Text('Click Here'),  
  
          onPressed: () {
```

```
        Navigator.push(  
            context,  
            MaterialPageRoute(builder: (context) => SecondRoute()),  
        );  
    },  
),  
),  
);  
}  
}  
  
class SecondRoute extends StatelessWidget {  
    @override  
    Widget build(BuildContext context) {  
        return Scaffold(  
            appBar: AppBar(  
                title: Text("Second Screen"),  
            ),  
            body: Center(  
                child: ElevatedButton(  
                    onPressed: () {  
                        Navigator.pop(context);  
                    },  
                    child: Text('Go back'),  
                ),  
            ),  
        );  
    }  
}
```

```
 );  
 }  
 }
```

Output:



7. Write a Flutter program to design a Login form using TextField, Check Box, Buttons, Drop down, Switch etc.

Code:

```
login_screen.dart
import 'package:flutter/material.dart';
import 'package:login_flutter/home_screen.dart';

class LoginPage extends StatefulWidget {
  @override
  State<LoginPage> createState() => _LoginPageState();
}

class _LoginPageState extends State<LoginPage> {
  bool _rememberMe = false;
  bool _enableNotifications = false;
  String? _selectedUserType;

  final List<String> _userTypes = ['Admin', 'Student', 'Teacher'];
  @override
  Widget build(BuildContext context) {
    return SafeArea(
      child: Scaffold(
        body: Container(
          margin: EdgeInsets.all(24),
          child: Column(
            mainAxisAlignment: MainAxisAlignment.spaceEvenly,
            children: [
              _header(context),
              _inputField(context),
              _extraOptions(context),
              _forgotPassword(context),
              _signup(context),
            ],
          ),
        ),
      );
  }

  _header(context) {
    return Column(
      children: [
        Text(
          "Welcome Back",
          style: TextStyle(fontSize: 40, fontWeight: FontWeight.bold),
        ),
        Text("Enter your credential to login"),
      ],
    );
  }

  _inputField(context) {
    return Column(
      crossAxisAlignment: CrossAxisAlignment.stretch,
      children: [
        TextFormField(
          decoration: InputDecoration(
            labelText: "Email Address",
            hintText: "Enter Email Address",
            prefixIcon: Icon(Icons.email),
          ),
        ),
        TextFormField(
          decoration: InputDecoration(
            labelText: "Password",
            hintText: "Enter Password",
            prefixIcon: Icon(Icons.lock),
          ),
        ),
        Row(
          mainAxisAlignment: MainAxisAlignment.end,
          children: [
            Text("Forgot Password?"),
            Text("Sign Up"),
          ],
        ),
      ],
    );
  }
}
```

```
TextField(  
    decoration: InputDecoration(  
        hintText: "Username",  
        border: OutlineInputBorder(  
            borderRadius: BorderRadius.circular(18),  
            borderSide: BorderSide.none),  
        fillColor: Theme.of(context).primaryColor.withOpacity(0.1),  
        filled: true,  
        prefixIcon: Icon(Icons.person)),  
,  
SizedBox(height: 10),  
TextField(  
    decoration: InputDecoration(  
        hintText: "Password",  
        border: OutlineInputBorder(  
            borderRadius: BorderRadius.circular(18),  
            borderSide: BorderSide.none),  
        fillColor: Theme.of(context).primaryColor.withOpacity(0.1),  
        filled: true,  
        prefixIcon: Icon(Icons.person),  
,  
        obscureText: true,  
,  
SizedBox(height: 10),  
DropdownButtonFormField<String>(  
    value: _selectedUserType,  
    items: _userTypes  
        .map((type) => DropdownMenuItem(  
            value: type,  
            child: Text(type),  
        ))  
        .toList(),  
    decoration: InputDecoration(  
        hintText: "Select User Type",  
        border: OutlineInputBorder(  
            borderRadius: BorderRadius.circular(18),  
            borderSide: BorderSide.none),  
        filled: true,  
        fillColor: Theme.of(context).primaryColor.withOpacity(0.1),  
,  
    onChanged: (value) {  
        setState(() {  
            _selectedUserType = value;  
        });  
    },  
,  
SizedBox(height: 10),  
ElevatedButton(  
    onPressed: () {  
        Navigator.push(  
            context,  
            MaterialPageRoute(builder: (context) => HomePage()),  
        );  
    },
```

```
        },
        child: Text(
            "Login",
            style: TextStyle(fontSize: 20),
        ),
        style: ElevatedButton.styleFrom(
            shape: StadiumBorder(),
            padding: EdgeInsets.symmetric(vertical: 16),
        ),
    ),
),

],
);
}

Widget _extraOptions(context) {
    return Column(
        children: [
            Row(
                children: [
                    Checkbox(
                        value: _rememberMe,
                        onChanged: (value) {
                            setState(() {
                                _rememberMe = value!;
                            });
                        },
                    ),
                    Text("Remember Me"),
                ],
            ),
            Row(
                mainAxisAlignment: MainAxisAlignment.spaceBetween,
                children: [
                    Text("Enable Notifications"),
                    Switch(
                        value: _enableNotifications,
                        onChanged: (value) {
                            setState(() {
                                _enableNotifications = value;
                            });
                        },
                    ),
                ],
            ),
        ],
    );
}

_forgotPassword(context) {
    return TextButton(onPressed: () {}, child: Text("Forgot password?"));
}

_signup(context) {
    return Row(
        mainAxisAlignment: MainAxisAlignment.center,
        children: [

```

```
        Text("Dont have an account? "),
        TextButton(onPressed: () {}, child: Text("Sign Up"))
    ],
);
}
}
```

main.dart

```
import 'package:flutter/material.dart';
import 'package:login/login_screen.dart';
```

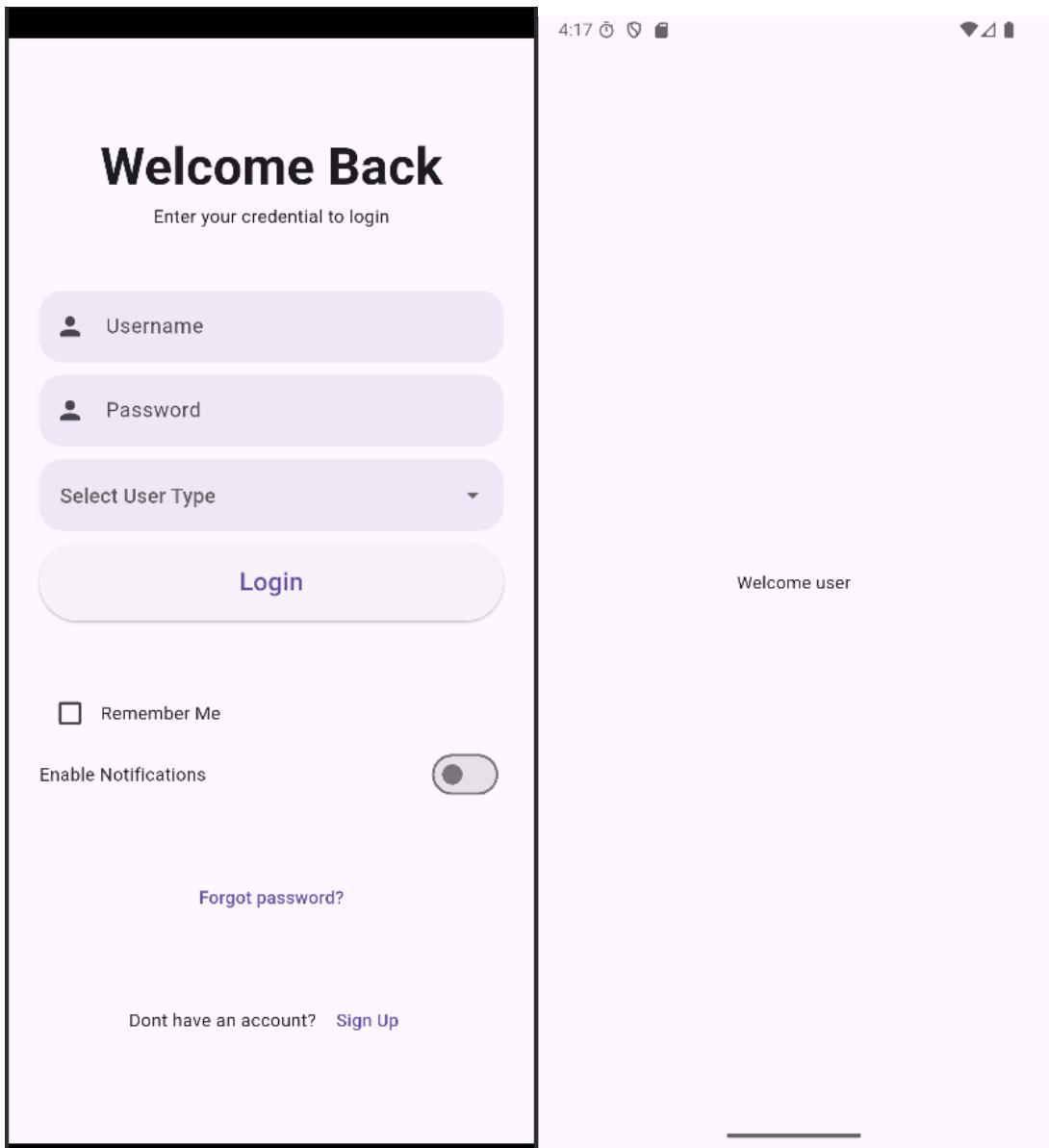
```
void main() => runApp(MyApp());
```

```
class MyApp extends StatelessWidget {
// This widget is the root of your application.
@Override
Widget build(BuildContext context) {
    return MaterialApp(
        title: 'Flutter Login UI',
        debugShowCheckedModeBanner: false,
        home: LoginPage(),
    );
}
}
```

home_screen.dart

```
import 'package:flutter/material.dart';

class HomePage extends StatelessWidget {
@Override
Widget build(BuildContext context) {
    return (Scaffold(
        body: Center(child:Text('Welcome user') ,)
    )
);
}
}
```

Output:

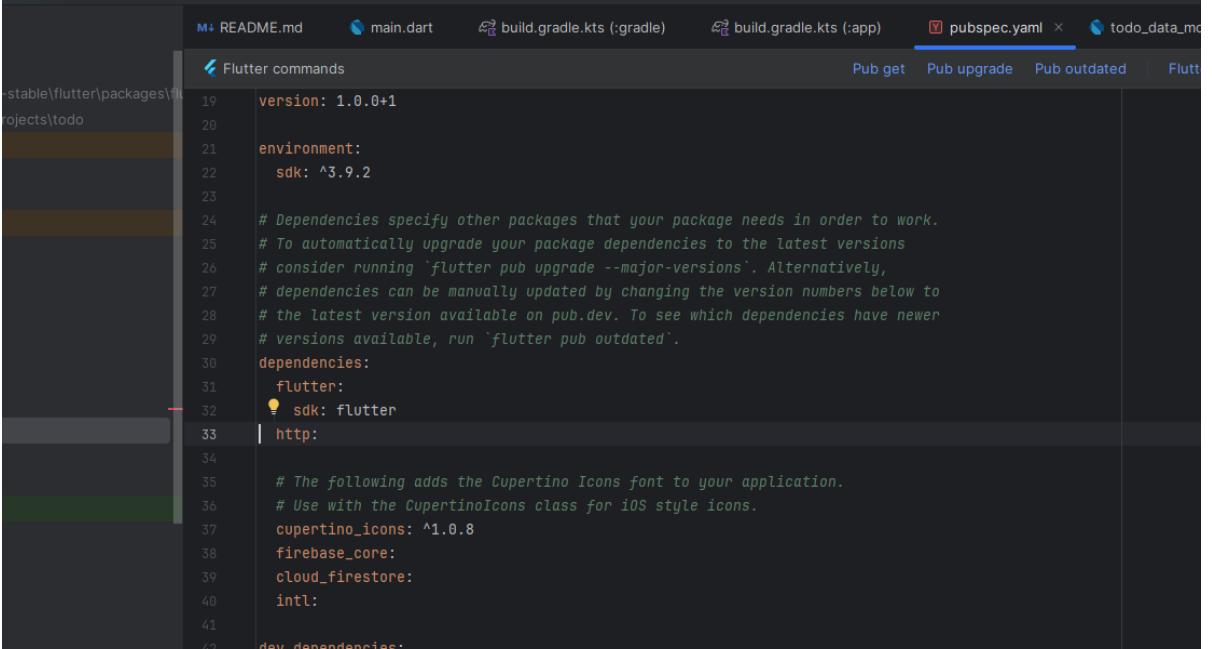
Module 6 Data Handling in Flutter:

1. Write a Flutter program based on RestAPI to fetch data.

Step 1:

Add dependency in pubsec.yaml

https:



```
version: 1.0.0+1
environment:
  sdk: ^3.9.2

# Dependencies specify other packages that your package needs in order to work.
# To automatically upgrade your package dependencies to the latest versions
# consider running `flutter pub upgrade --major-versions`. Alternatively,
# dependencies can be manually updated by changing the version numbers below to
# the latest version available on pub.dev. To see which dependencies have newer
# versions available, run `flutter pub outdated`.
dependencies:
  flutter:
    sdk: flutter
  http:
    # The following adds the Cupertino Icons font to your application.
    # Use with the CupertinoIcons class for iOS style icons.
    cupertino_icons: ^1.0.8
  firebase_core:
  cloud_firestore:
  intl:

dev_dependencies:
```

Step 2:

Code in main.dart file

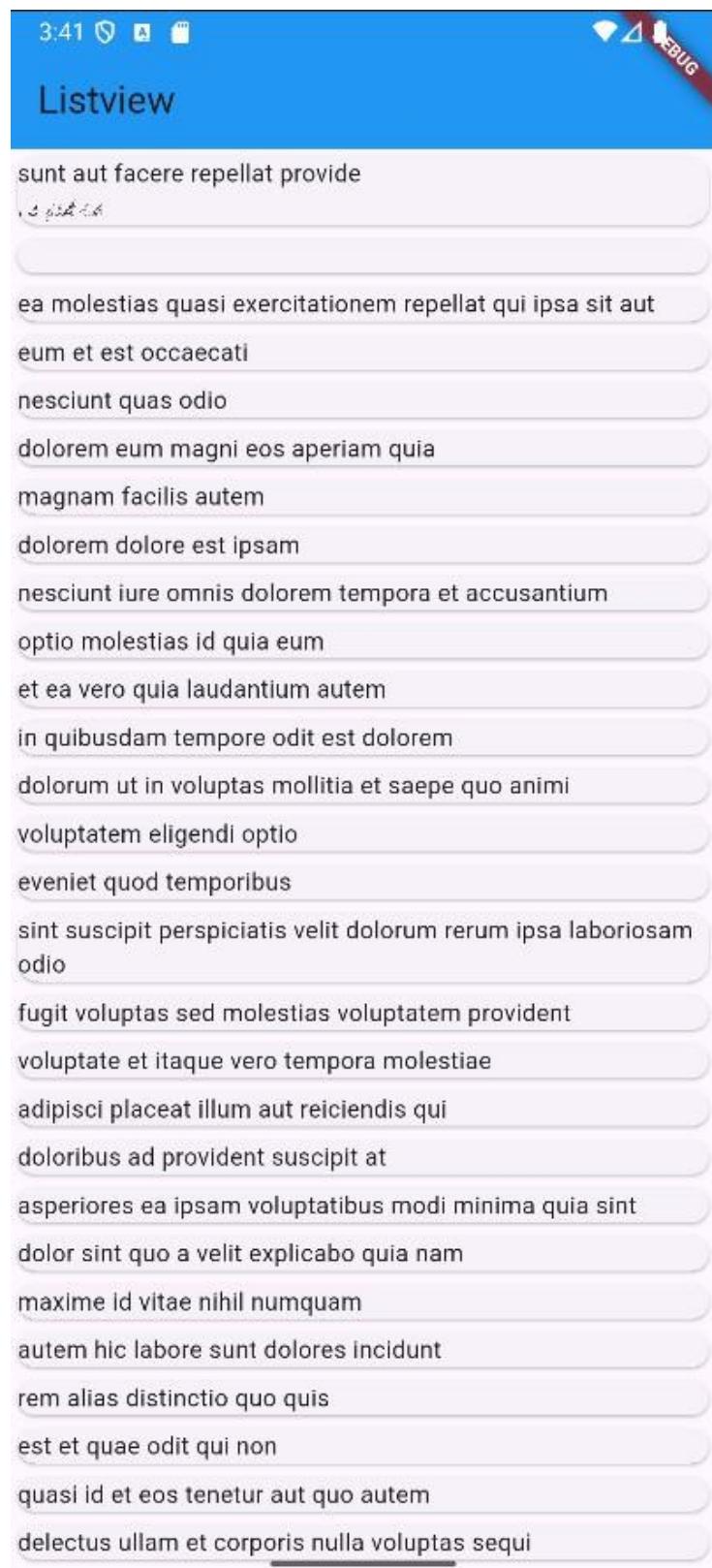
```
import 'dart:async';
import 'dart:convert';
import 'package:flutter/material.dart';
import 'package:http/http.dart' as http;
```

```
void main() {
  runApp(const MaterialApp(
    home: HomePage()
  ));
}
```

```
class HomePage extends StatefulWidget {  
  const HomePage({Key? key}) : super(key: key);  
  
  @override  
  HomePageState createState() => HomePageState();  
}  
  
class HomePageState extends State<HomePage> {  
  late final List data;  
  
  Future<String> getData() async {  
    var response = await http.get(  
      Uri.parse("https://jsonplaceholder.typicode.com/posts"),  
      headers: {  
        "Accept": "application/json"  
      }  
    );  
  
    setState(() {  
      data = json.decode(response.body);  
    });  
    // ignore: avoid_print  
    // print(data[1]["title"]);  
    return "Success!";  
  }  
  
  @override  
  // ignore: must_call_super  
  void initState(){  
    getData();  
  }  
  @override  
  Widget build(BuildContext context){
```

```
return Scaffold(  
    appBar: AppBar(title: const Text("Listview"), backgroundColor: Colors.blue),  
    body: ListView.builder(  
        // ignore: unnecessary_null_comparison  
        itemCount: data.length,  
        itemBuilder: (BuildContext context, int index){  
            return Card(  
                child: Text(data[index]["title"]),  
            );  
        },  
    ),  
);  
}  
}
```

Output:



2. Write a flutter program to demonstrate JSON serialization and Deserialization.

Step 1. Make usermodel.dart in lib file

Add code

```
class UserModel {  
    late String id;  
    late String fullname;  
    late String email;
```

// Map to Object

```
UserModel ({required this.id, required this.fullname, required this.email});  
UserModel.fromMap(Map<String , dynamic> map){  
    this.id = map["id"];  
    this.fullname = map["fullname"];  
    this.email = map["email"];  
}
```

// Object to Map

```
Map <String , dynamic> toMap()  
{  
    return{  
        "id": this.id,  
        "fullname": this.fullname,  
        "email": this.email,  
    };  
}
```

}

Step 2. Main.dart

```
import 'dart:convert';  
import 'package:flutter/material.dart';  
import 'package:jsonapp/usermodel.dart';
```

```
void main() {  
    runApp(const MyApp());  
}
```

```
class MyApp extends StatelessWidget {  
    const MyApp({Key? key}) : super(key: key);  
    @override  
    Widget build(BuildContext context) {  
        return MaterialApp(  
            title: 'Flutter Demo',  
            theme: ThemeData(  
                primarySwatch: Colors.blue,  
,  
                home: HomePage(),
```

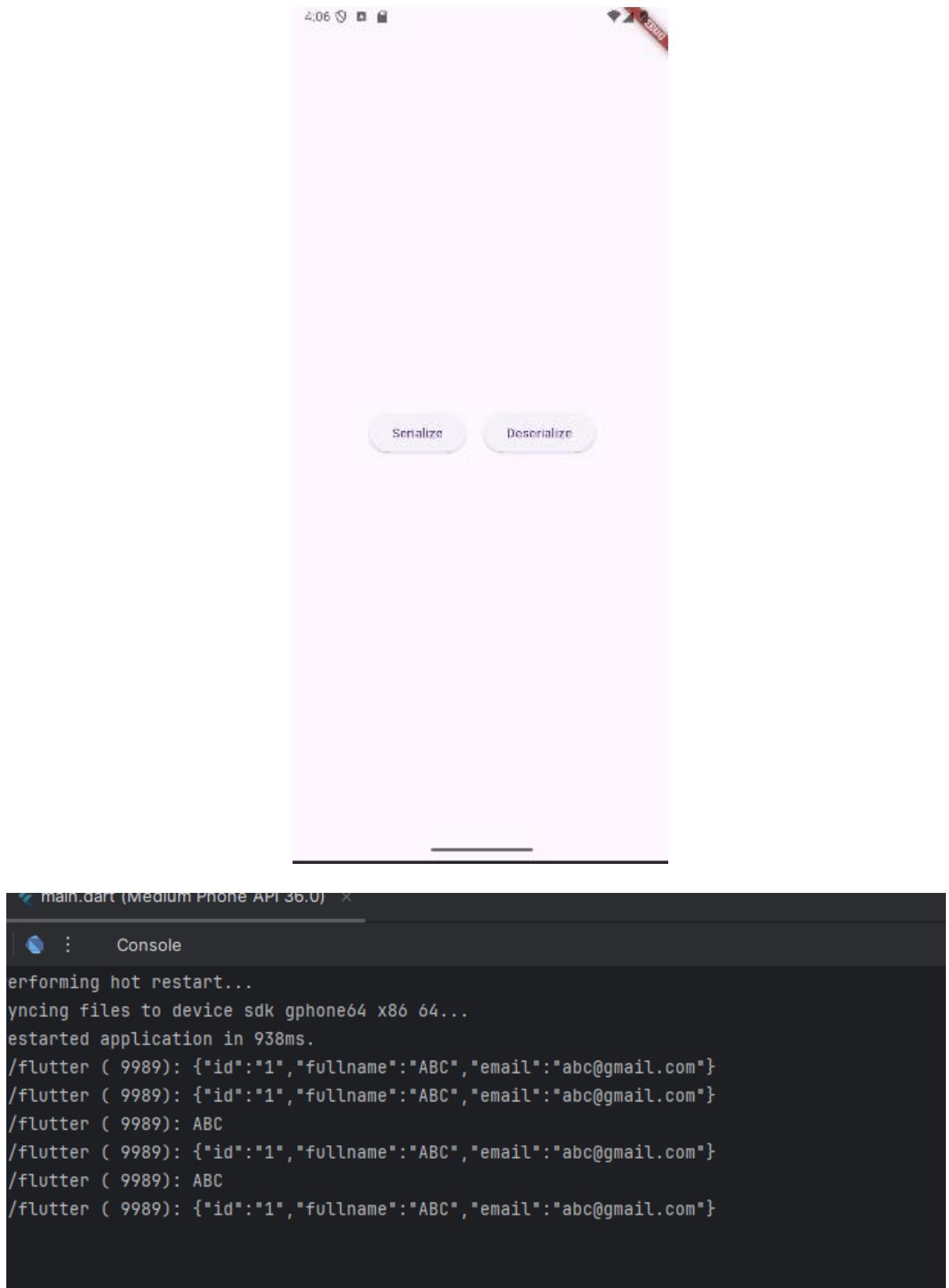
```
        );
    }
}

class HomePage extends StatefulWidget {
    @override
    _HomePageState createState() => _HomePageState();
}

class _HomePageState extends State<HomePage> {
    UserModel userObject = new UserModel(id: "1", fullname: "ABC", email:
        "abc@gmail.com");
    String userJSON = '{"id": "1", "fullname": "ABC", "email": "abc@gmail.com"}';

    @override
    Widget build(BuildContext context) {
        return Scaffold(
            body: Center(
                child: Row(
                    mainAxisAlignment: MainAxisAlignment.center,
                    children: <Widget>[
                        ElevatedButton(
                            onPressed: (){
                                //Serialization
                                Map<String, dynamic> userMap = userObject.toMap();
                                var json = jsonEncode(userMap);
                                print(json.toString());
                            },
                            child: Text("Serialize"),
                        ),
                        SizedBox(width: 20,),
                        ElevatedButton(
                            onPressed: (){
                                var decode = jsonDecode(userJSON);
                                Map<String, dynamic> userMap = decode;
                                UserModel newuser = new UserModel.fromMap(userMap);
                                print(newuser.fullname.toString());
                            },
                            child: Text("Deserialize")
                        ),
                    ],
                ),
            );
    }
}
```

Output:



3. Write a flutter program to perform CRUD operations using sqflite.

Step 1: Add Dependencies in pubspec.yaml
dependencies:

```
flutter:  
  sdk: flutter  
  sqflite: ^2.3.3  
  path_provider: ^2.1.3
```

Step2: Create a Model Class — user_model.dart

```
class User {  
  int? id;  
  String name;  
  String email;  
  
  User({this.id, required this.name, required this.email});  
  
  // Convert a User into a Map.  
  Map<String, dynamic> toMap() {  
    var map = <String, dynamic>{  
      'name': name,  
      'email': email,  
    };  
    if (id != null) map['id'] = id;  
    return map;  
  }  
  
  // Convert a Map into a User  
  User.fromMap(Map<String, dynamic> map)  
    : id = map['id'],  
      name = map['name'],  
      email = map['email'];  
}
```

Step 3: Create the Database Helper — db_helper.dart

```
import 'package:sqflite/sqflite.dart';  
import 'package:path/path.dart';  
import 'user_model.dart';  
  
class DBHelper {  
  static final DBHelper _instance = DBHelper._internal();  
  factory DBHelper() => _instance;  
  DBHelper._internal();  
  
  static Database? _db;  
  
  Future<Database> get db async {  
    _db ??= await initDB();  
  }
```

```
        return _db!;
    }

Future<Database> initDB() async {
    String path = join(await getDatabasesPath(), 'users.db');
    return await openDatabase(path, version: 1, onCreate: _onCreate);
}

Future _onCreate(Database db, int version) async {
    await db.execute("""
        CREATE TABLE users (
            id INTEGER PRIMARY KEY AUTOINCREMENT,
            name TEXT,
            email TEXT
        )
    """);
}

// Insert a new user
Future<int> insertUser(User user) async {
    var dbClient = await db;
    return await dbClient.insert('users', user.toMap());
}

// Get all users
Future<List<User>> getUsers() async {
    var dbClient = await db;
    List<Map<String, dynamic>> maps = await dbClient.query('users');
    return maps.map((user) => User.fromMap(user)).toList();
}

// Update user
Future<int> updateUser(User user) async {
    var dbClient = await db;
    return await dbClient.update('users', user.toMap(),
        where: 'id = ?', whereArgs: [user.id]);
}

// Delete user
Future<int> deleteUser(int id) async {
    var dbClient = await db;
    return await dbClient.delete('users', where: 'id = ?', whereArgs: [id]);
}
```

Step 4: Create the UI — main.dart

```
import 'package:flutter/material.dart';
import 'db_helper.dart';
import 'user_model.dart';
```

```
void main() {
```

```
runApp(MaterialApp(  
    debugShowCheckedModeBanner: false,  
    home: HomeScreen(),  
>);  
>  
  
class HomeScreen extends StatefulWidget {  
    @override  
    State<HomeScreen> createState() => _HomeScreenState();  
>  
  
class _HomeScreenState extends State<HomeScreen> {  
    final DBHelper dbHelper = DBHelper();  
    final TextEditingController nameController = TextEditingController();  
    final TextEditingController emailController = TextEditingController();  
  
    List<User> userList = [];  
  
    @override  
    void initState() {  
        super.initState();  
        refreshUserList();  
>  
  
    void refreshUserList() async {  
        final data = await dbHelper.getUsers();  
        setState(() => userList = data);  
>  
  
    void insertUser() async {  
        if (nameController.text.isNotEmpty && emailController.text.isNotEmpty) {  
            await dbHelper.insertUser(  
                User(name: nameController.text, email: emailController.text));  
            nameController.clear();  
            emailController.clear();  
            refreshUserList();  
>  
        }  
>  
    }  
  
    void updateUser(User user) async {  
        user.name = nameController.text;  
        user.email = emailController.text;  
        await dbHelper.updateUser(user);  
        nameController.clear();  
        emailController.clear();  
        refreshUserList();  
>  
    }  
  
    void deleteUser(int id) async {  
        await dbHelper.deleteUser(id);  
        refreshUserList();  
>
```

```
}

void showUserDialog({User? user}) {
    if (user != null) {
        nameController.text = user.name;
        emailController.text = user.email;
    }

    showDialog(
        context: context,
        builder: (_) => AlertDialog(
            title: Text(user == null ? 'Add User' : 'Edit User'),
            content: Column(
                mainAxisSize: MainAxisSize.min,
                children: [
                    TextField(
                        controller: nameController,
                        decoration: InputDecoration(labelText: 'Name'),
                    ),
                    TextField(
                        controller: emailController,
                        decoration: InputDecoration(labelText: 'Email'),
                    ),
                ],
            ),
            actions: [
                TextButton(
                    child: Text('Cancel'),
                    onPressed: () => Navigator.pop(context),
                ),
                TextButton(
                    child: Text(user == null ? 'Add' : 'Update'),
                    onPressed: () {
                        if (user == null) {
                            insertUser();
                        } else {
                            updateUser(user);
                        }
                        Navigator.pop(context);
                    },
                ),
            ],
        ),
    );
}

@Override
Widget build(BuildContext context) {
    return Scaffold(
        appBar: AppBar(
            title: Text('Flutter CRUD using SQFlite'),

```

```
centerTitle: true,  
),  
body: ListView.builder(  
    itemCount: userList.length,  
    itemBuilder: (context, index) {  
        final user = userList[index];  
        return ListTile(  
            title: Text(user.name),  
            subtitle: Text(user.email),  
            trailing: Row(  
                mainAxisAlignment: MainAxisAlignment.end,  
                children: [  
                    IconButton(  
                        icon: Icon(Icons.edit, color: Colors.blue),  
                        onPressed: () => showUserDialog(user: user)),  
                    IconButton(  
                        icon: Icon(Icons.delete, color: Colors.red),  
                        onPressed: () => deleteUser(user.id!)),  
                ],  
            ),  
        );  
    },  
    floatingActionButton: FloatingActionButton(  
        onPressed: () => showUserDialog(),  
        child: Icon(Icons.add),  
    ),  
);  
}  
}
```

Output:

