"VIDYALANKAR SCHOOL OF INFORMATION TECHNOLOGY, WADALA"

AFFILIATED TO

UNIVERSITY OF MUMBAI
CENTRE FOR DISTANCE AND ONLINE EDUCATION (CDOE)

Mobile Computing

SUBMITTED BY

Hrushikesh Sunil Sawant
Application No.: 129788

SUBMITTED IN PARTIAL FUFILLMENT OF THE REQUIREMENTS FOR QUALIFYING MCA PART-II (SEMISTER-III) EXAMINATION

2023 - 2024

Centre for Distance and Online Education

Vidya Nagari, Kalina, Santacruz East – 400098.

CERTIFICATE

This is to certify that **Mr. Sawant Hrushikesh Sunil Sandhya** of **Master in Computer Application** (MCA) Semester III has completed the specified term work in the subject of **Mobile Computing** satisfactorily within this institute as laid down by University of Mumbai during the academic year 2023 to 2024.

(Prof. Ujwala Sav)	_____	_____
Subject In-charge	External Examiner	Coordinator – M.C.A

"VIDYALANKAR SCHOOL OF INFORMATION TECHNOLOGY, WADALA"

AFFILIATED TO

UNIVERSITY OF MUMBAI

CENTRE FOR DISTANCE AND ONLINE EDUCATION (CDOE)

INDEX

4.	**TO STUDY DIFFERENT TYPES OF FILE OPERATIONS AND SHARED PREFERENCES:** A. Program to create a file in a directory and perform following file operations. a. Write into a file b. Read from a file c. Delete a file B. Create a new project and create a login Activity. In this create a login UI asking user email and password with an option of remember me checkbox. Also a button displaying Sign In or Register using shared preferences.		50-60	
5.	**TO STUDY AUDIO, VIDEO AND LOCATION ON ANDROID:** A. Write a program to: a. Record an audio and play it. b. Play a video in a video view. B. Create an application to display the current location of your device. (Latitude & Longitude value). C. Calculate route distance between two locations.		61-72	
6.	**ANDROID PROGRAM BASED ON REST API:** A. Create a basic application that allows you to download HTML from a given web page using HttpURLConnection. B. Create an application to parse the data using JSONObject methods and set it in the Text View's. (Employee name and salary stored in JSON format). C. Write a basic application to (use volley library), create a button and on click of the button a HTTP request will be send to server. The response from the server is then displayed using Toast on the screen.		73-80	
7.	**INTRODUCTION TO DART AND FLUTTER:** Flutter program using layout, widget and state management. A. Create a widget ProductBox that contains the details of the product, such as image, name, price, and description. In the ProductBox widget, we use the following child widgets: Container, Row, Column, Expanded, Card, Text, Image, etc.		81-84	
8.	**INTRODUCTION TO DART AND FLUTTER:** A. Create a registration form using Flutter		85-86	

Practical 1: INTRODUCTION TO ANDROID

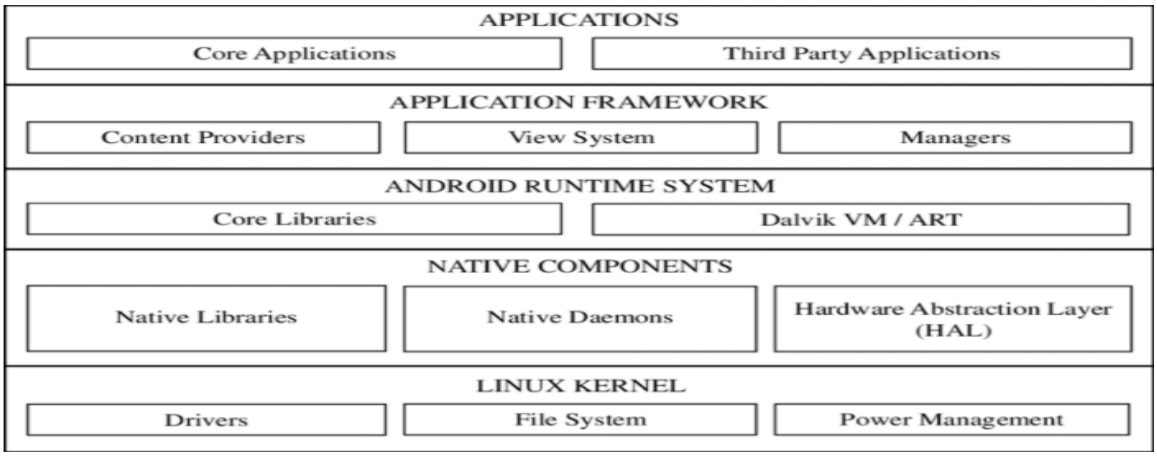
Study android platform, the layers of android and 4 kinds of android components, Understanding the manifest.xml file.

- **Android Platform:**

1. The Android platform is a platform for mobile devices that uses a modified Linux kernel.
2. The Android Platform was introduced by the Open Handset Alliance in November of 2007.
3. Most applications that run on the Android platform are written in the Java programming language.
4. Although most of the applications that run on the Android Platform are written in Java, there is no Java Virtual Machine. Instead, the Java classes are first compiled into what are known as Dalvik Executables and run on the Dalvik Virtual Machine.
5. Android is an open development platform.
6. To create an application for the platform, a developer requires the Android SDK, which includes tools and APIs.

- **Layers Of Android:**

The following are the layers that compose the Android architecture.



- **Application:**

1. Android application is the top layer.
2. All applications are installed on this layer only. Examples of such applications are Contacts Books, Browser, and Games etc.

- **Application Framework:**

1. The Application Framework layer provides many higher-level services to applications in the form of Java classes.
2. Application developers are allowed to make use of these services in their applications.
3. The Android framework includes the following key services:
    a. Activity Manager − Controls all aspects of the application lifecycle and activity stack.
    b. Content Providers − Allows applications to publish and share data with other applications.
    c. Resource Manager − Provides access to non-code embedded resources such as strings, color settings and user interface layouts.
    d. Notifications Manager − Allows applications to display alerts and notifications to the user.
    e. View System − An extensible set of views used to create application user interfaces.

- **Android Runtime and Core Libraries:**

1. Android currently uses Android Runtime (ART) to execute application code.
2. ART is preceded by the Dalvik Runtime that compiled developer code to Dalvik Executable files (Dex files). These execution environments are optimized for the android platform taking into consideration the processor and memory constraints on mobile devices.
3. Some of the core libraries that are present in the Android operating systems are:
    a. android.app − Provides access to the application model and is the cornerstone of all Android applications.
    b. android.content − Facilitates content access, publishing and messaging between applications and application components.
    c. android.database − Used to access data published by content providers and includes SQLite database management classes.
    d. android.opengl − A Java interface to the OpenGL ES 3D graphics rendering API.
    e. android.os − Provides applications with access to standard operating system services including messages, system services and inter-process communication.
    f. android.text − Used to render and manipulate text on a device display.
    g. android.view − The fundamental building blocks of application user interfaces.
    h. android.widget − A rich collection of pre-built user interface components such as buttons, labels, list views, layout managers, radio buttons etc.
    i. android.webkit − A set of classes intended to allow web-browsing capabilities to be built into applications.

- **Linux Kernel:**
    1. The root component of the Android System is the Linux Kernel. It is the foundational piece that enables all of Android's functionality.
    2. The Linux Kernel is a battle-tested piece of software that has been used in developing operating systems for devices of wide range, from supercomputers to small gadgets. It has limited processing abilities like small networked gadgets for the Internet of Things (IoT).
    3. The Linux Kernel can be tweaked to meet the device specifications to make it possible for manufacturers to make Android devices with different capabilities to match user experience.

- Four Kinds of Android Components:

Four Kinds Of Android Components are as follows:

1. **Service**:
    - A service in Android is a background process.
    - Services are typically used for processes that are ongoing or that take a significant period of time.
    - A service doesn't have a user interface, so they are often combined with other components such as activities.
    - A typical example is an app in which an activity starts a service running on user interaction, with the service perhaps uploading data to a web resource.
    - The user can continue to interact with the activity while the service runs because it executes in the background.
2. **Content Providers:**
    - A content provider component supplies data from one application to others on request. Such requests are handled by the methods of the ContentResolver class.
    - The data may be stored in the file system, the database or somewhere else entirely.
    - A content provider is implemented as a subclass of ContentProvider class and must implement a standard set of APIs that enable other applications to perform transactions.
3. **Broadcast Receivers:**
    - The Android system makes various types of broadcasts an app can respond to.
    - Apps can be developed to make these broadcasts, but this is far less likely than to listen for existing broadcasts, at least for first apps.
    - System announcements include information about the device's hardware, such as the battery level, the screen shutting off, the charger being plugged into an outlet, etc.
    - To receive broadcast announcements on Android, apps can use a broadcast receiver.
    - A typical example of this is a battery level widget in which you want to update the display when the battery level changes. In this case, you could use a service class in conjunction with a broadcast receiver to let your app keep listening for announcements in the background.

4. **Activities:**
    - An activity represents a single screen with a user interface,in-short Activity performs actions on the screen.
    - For example, an email application might have one activity that shows a list of new emails, another activity to compose an email, and another activity for reading emails. If an application has more than one activity, then one of them should be marked as the activity that is presented when the application is launched.

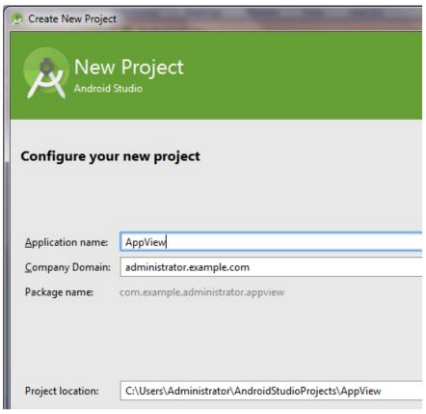
- **Manifest.xml File:**
    1. Every app project must have an AndroidManifest.xml file (with precisely that name) at the root of the project source set.
    2. The manifest file describes essential information about your app to the Android build tools, the Android operating system, and Google Play.
    3. On Android Studio to build app, the manifest file is created for already, and most of the essential manifest elements are added to build the app.
    4. Among many other things, the manifest file is required to declare the following:
        a. The app's package name, which usually matches your code's namespace. The Android build tools use this to determine the location of code entities when building your project. When packaging the app, the build tools replace this value with the application ID from the Gradle build files, which is used as the unique app identifier on the system and on Google Play.
        b. The components of the app, which include all activities, services, broadcast receivers, and content providers. Each component must define basic properties such as the name of its Kotlin or Java class. It can also declare capabilities such as which device configurations it can handle, and intent filters that describe how the component can be started.
        c. The permissions that the app needs in order to access protected parts of the system or other apps. It also declares any permissions that other apps must have if they want to access content from this app.

The hardware and software features the app requires, which affects which devices can install the app from Google Play.

Practical 2: BASIC CONTROLS AND UI COMPONENT

A. Program to demonstrate use of different text control, RadioGroup, RadioButton, Checkbox and Button control by creating Registration form.

Create a new Android project called AppView. By default, it creates main.xml file located in the res/layout folder, which contains a <TextView> element.




```
<LinearLayout
        xmlns:android="http://schemas.android.com/apk/res/android"
        xmlns:tools="http://schemas.android.com/tools".
        android:layout_width="fill_parent"
        android:layout_height="match_parent"
        android:orientation="vertical"
        tools:context=".MainActivity">
        <TextView android:text="@string/hello_world"
        android:layout_width="fill_content"
        android:layout_height="wrap_content" />
</LinearLayout >
```

The TextView is used to display text/caption to the user. This is the most basic View and very frequently used in an application.
The next View is a subclass of TextView and it is EditText. This View allows the user to edit the text displayed.

```
<EditText
android:layout_width="fill_parent"
android:layout_height="wrap_content"
android:id="@+id/txtUserName" />
```

Button represents a push-button widget.

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

ImageButton is similar to Button View except that it displays an image with text.

```
<ImageButton
        android:layout_width="fill_parent"
        android:layout_height="wrap_content"
        android:id="@+id/imgButton"
        android:src="@drawable/abc_ic_menu_copy_mtrl_am_alpha"
/>
```

CheckBox is a type of button that has two states; i.e., checked or unchecked.

```
<CheckBox
        android:layout_width="fill_parent"
        android:layout_height="wrap_content"
        android:id="@+id/chkIndia"
        android:text="India"
/>
<CheckBox
        android:layout_width="fill_parent"
        android:layout_height="wrap_content"
        android:id="@+id/chkUS" style="?android:attr/starStyle"
android:text="US"
/>
```

RadioGroup and RadioButton, both have two states: either checked or unchecked. A RadioGroup is used to group together one or more RadioButton Views, thereby allowing only one RadioButton to be checked within the RadioGroup.

```
<RadioGroup
        android:layout_width="fill_parent"
        android:layout_height="wrap_content"
        android:orientation="vertical"
        android:id="@+id/rdoGroup">
        <RadioButton
                android:id="@+id/rbMale"
                android:layout_width="fill_parent"
                android:layout_height="wrap_content"
```

```
            android:text="Male"/>
        <RadioButton
            android:id="@+id/rbFemale"
            android:layout_width="fill_parent"
            android:layout_height="wrap_content"
            android:text="Female"/>
</RadioGroup>
```

ToggleButton displays checked/unchecked states using a light indicator.

```
<ToggleButton
android:layout_width="fill_parent"
android:layout_height="wrap_content"
android:id="@+id/toggleButton"/>
```

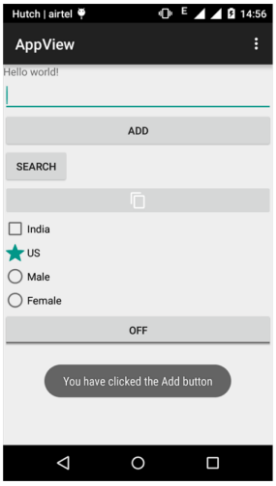
Now, add some code as below to handle View events for elements, like Introduction to Android Button, CheckBox etc.

```
Button buttonAdd = (Button) findViewById(R.id.btnAdd);
buttonAdd.setOnClickListener(new View.OnClickListener()
{
        @Override
        public void onClick(View view)
        {
            DisplayMessage("You have clicked the Add button");
        }
});
```

Write a common method to display the text message as below.

```
private void DisplayMessage(String textMessage) {
        Toast.makeText(getBaseContext(), textMessage, Toast.LENGTH_SHORT).show();
}
```

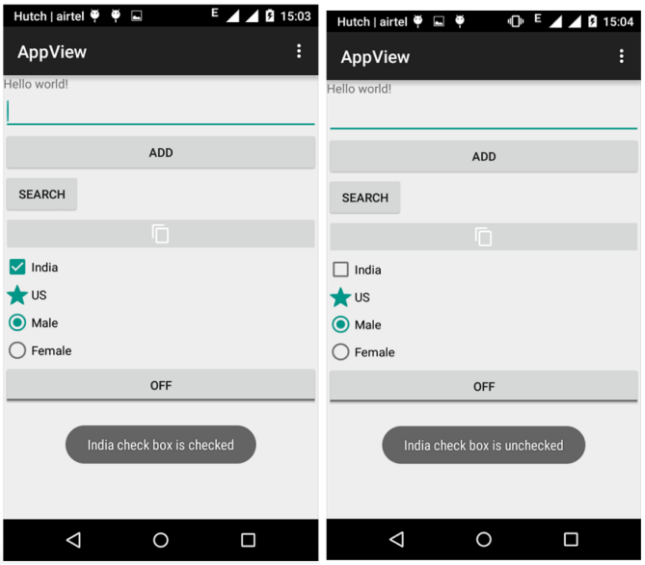
Now, run the application by pressing F11



To handle the View events for element CheckBox, add the below code.

```
CheckBox checkBox = (CheckBox) findViewById(R.id.chkIndia);
checkBox.setOnClickListener(new View.OnClickListener()
{
        @Override
        public void onClick(View view)
        {
        if (((CheckBox) view).isChecked()) DisplayMessage("India check box is checked");
        else DisplayMessage("India check box is unchecked");
        }
});
```

Now, run the application by pressing F11. For example, I have checked the India CheckBox.



Now, uncheck the India CheckBox.

B. Create an application to design simple calculator to perform addition, subtraction, multiplication and division. Show message for divide by zero error using Toast.

- **Activity_main.xml:**

```
<?xml version="1.0" encoding="utf-8"?>
<AbsoluteLayout 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">
   <EditText
      android:id="@+id/editText4"
      android:layout_width="wrap_content"
      android:layout_height="wrap_content"
      android:layout_x="73dp"
      android:layout_y="119dp"
      android:ems="10"
      android:hint="Enter first number"
      android:inputType="numberDecimal"
      tools:layout_editor_absoluteX="101dp"
      tools:layout_editor_absoluteY="87dp" />
   <EditText
      android:id="@+id/editText5"
      android:layout_width="wrap_content"
      android:layout_height="wrap_content"
      android:layout_x="76dp"
      android:layout_y="198dp"
      android:ems="10"
      android:hint="Enter second number"
      android:inputType="numberDecimal"
      tools:layout_editor_absoluteX="101dp"
      tools:layout_editor_absoluteY="207dp" />
   <Button
      android:id="@+id/button"
      android:layout_width="wrap_content"
      android:layout_height="wrap_content"
      android:layout_x="64dp"
      android:layout_y="316dp"
      android:text="ADD"
```

```
        tools:layout_editor_absoluteX="164dp"
        tools:layout_editor_absoluteY="335dp" />
    <TextView
        android:id="@+id/textView"
        android:layout_width="81dp"
        android:layout_height="38dp"
        android:layout_x="135dp"
        android:layout_y="520dp"
        android:text="Result"
        tools:layout_editor_absoluteX="173dp"
        tools:layout_editor_absoluteY="545dp" />
    <Button
        android:id="@+id/button2"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_x="204dp"
        android:layout_y="318dp"
        android:text="Sub" />
    <Button
        android:id="@+id/button3"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_x="65dp"
        android:layout_y="402dp"
        android:text="Mult" />
    <Button
        android:id="@+id/button4"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_x="205dp"
        android:layout_y="398dp"
        android:text="Div" />
</AbsoluteLayout>
```

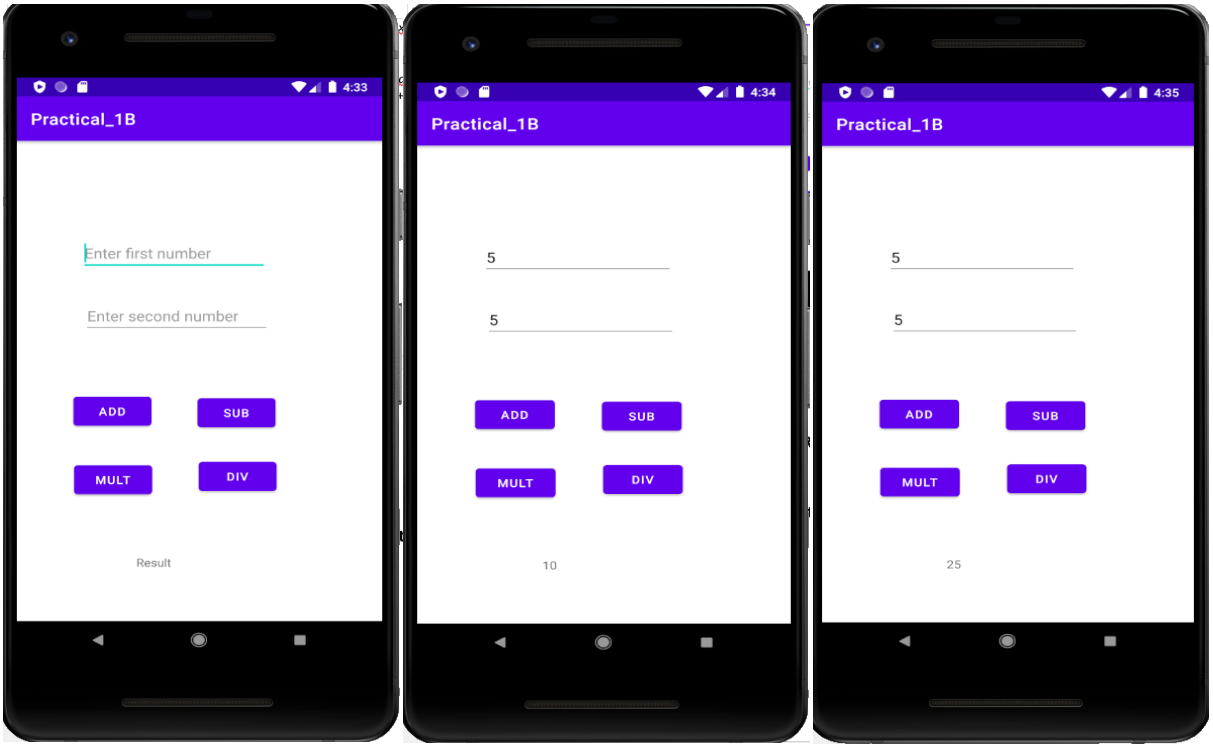
- **MainActivity.java:**

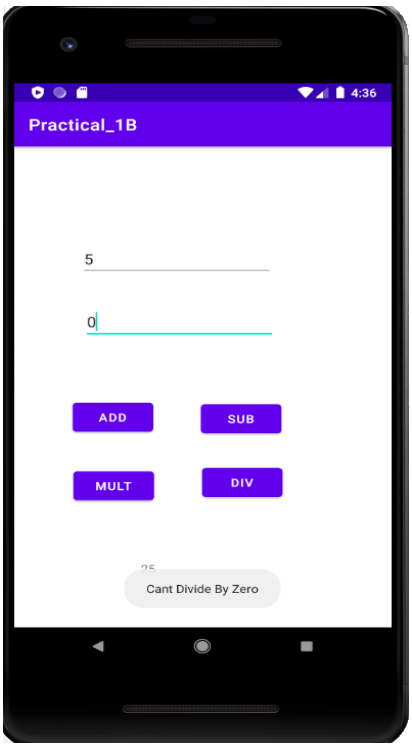
```
package com.example.practical_1b;
import androidx.appcompat.app.AppCompatActivity;
import android.os.Bundle;
import android.view.View;
import android.widget.Button;
```

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

public class MainActivity extends AppCompatActivity {
   Button ad,sb,mu,di;
   EditText n1,n2;
   TextView tv;
   @Override
   protected void onCreate(Bundle savedInstanceState) {
      super.onCreate(savedInstanceState);
      setContentView(R.layout.activity_main);
      ad=findViewById(R.id.button);
      sb=findViewById(R.id.button2);
      mu=findViewById(R.id.button3);
      di=findViewById(R.id.button4);
      n1=findViewById(R.id.editText4);
      n2=findViewById(R.id.editText5);
      tv=findViewById(R.id.textView);
      ad.setOnClickListener(
            new View.OnClickListener() {
               @Override
               public void onClick(View v) {
                  int n3=0;
                  String num1=n1.getText().toString();
                  String num2=n2.getText().toString();
                  n3=Integer.parseInt(num1)+Integer.parseInt(num2);
                  tv.setText(" "+n3);
               }
            }
      );
      sb.setOnClickListener(
            new View.OnClickListener() {
               @Override
               public void onClick(View v) {
                  int n4=0;
                  String num1=n1.getText().toString();
                  String num2=n2.getText().toString();
                  n4=Integer.parseInt(num1)-Integer.parseInt(num2);
                  tv.setText(" "+n4);
```

```
            }
        }
    );
    mu.setOnClickListener(
        new View.OnClickListener() {
            @Override
            public void onClick(View v) {
                int n4=0;
                String num1=n1.getText().toString();
                String num2=n2.getText().toString();
                n4=Integer.parseInt(num1)*Integer.parseInt(num2);
                tv.setText(" "+n4);

            }
        }
    );
    di.setOnClickListener(
        new View.OnClickListener() {
            @Override
            public void onClick(View v) {
                int n4=0;
                String num1=n1.getText().toString();
                String num2=n2.getText().toString();
                if(Integer.parseInt(num2)==0)
                {
                    Toast.makeText(MainActivity.this,"Cant Divide By
Zero",Toast.LENGTH_LONG).show();
                }
                else {
                    n4 = Integer.parseInt(num1) / Integer.parseInt(num2);
                    tv.setText(" " + n4);
                }
            }
        }
    );
  }
}
```

- **Output:**


4:33
Practical_1B
Enter first number
Enter second number
ADD
SUB
MULT
DIV
Result
4:34
Practical_1B
5
5
ADD
SUB
MULT
DIV
10
4:35
Practical_1B
5
5
ADD
SUB
MULT
DIV
25


4:36
Practical_1B
5
0
ADD
SUB
MULT
DIV
Cant Divide By Zero

C. Program to demonstrate use of Spinner, AutoCompleteTextView, multiline text and TextView control by creating Feedback form.

- **activity_main.xml:**

```
<?xml version="1.0" encoding="utf-8"?>
<android.support.constraint.ConstraintLayout
xmlns:android="http://schemas.android.com/apk/res/android"
xmlns:app="http://schemas.android.com/apk/res-auto"
xmlns:tools="http://schemas.android.com/tools"
android:layout_width="match_parent"
android:layout_height="match_parent"
tools:context=".MainActivity">
Basic Controls and UI
Component
<LinearLayout
android:layout_width="match_parent"
android:layout_height="wrap_content"
android:orientation="vertical"
tools:ignore="MissingConstraints"
android:layout_marginLeft="20dp"
android:layout_marginRight="30dp">
<TextView
android:id="@+id/textView"
android:layout_width="match_parent"
android:layout_height="wrap_content"
android:text="Thank You For Visiting Nexon Auto Shop"
android:layout_marginBottom="40dp"
android:layout_marginTop="30dp"
android:textColor="@color/black"
android:textSize="20sp"
android:textStyle="bold"
/>
<LinearLayout
android:layout_width="match_parent"
android:layout_height="wrap_content"
android:orientation="horizontal"
android:layout_marginBottom="40dp">
<TextView
android:id="@+id/textView2"
android:layout_width="90dp"
```

```
android:layout_height="wrap_content"
android:layout_weight="1"
android:text="Name"
android:textSize="15sp"/>
<EditText
android:id="@+id/editTextTextPersonName"
android:layout_width="wrap_content"
android:layout_height="wrap_content"
android:layout_weight="1"
android:ems="10"
android:inputType="textPersonName" />
</LinearLayout>
<LinearLayout
android:layout_width="match_parent"
android:layout_height="wrap_content"
android:orientation="horizontal"
android:layout_marginBottom="40dp">
<TextView
android:id="@+id/textView4"
android:layout_height="wrap_content"
android:layout_width="150dp"
android:layout_weight="1"
android:text="Country"
android:textSize="15sp"/>
<Spinner
android:id="@+id/spinner"
android:layout_width="match_parent"
android:layout_height="wrap_content"
android:layout_weight="1" />
</LinearLayout>
<LinearLayout
android:layout_width="match_parent"
android:layout_height="wrap_content"
android:orientation="horizontal"
android:layout_marginBottom="40dp">
<TextView
android:id="@+id/textView5"
android:layout_width="100dp"
android:layout_height="wrap_content"
android:layout_weight="1"
```

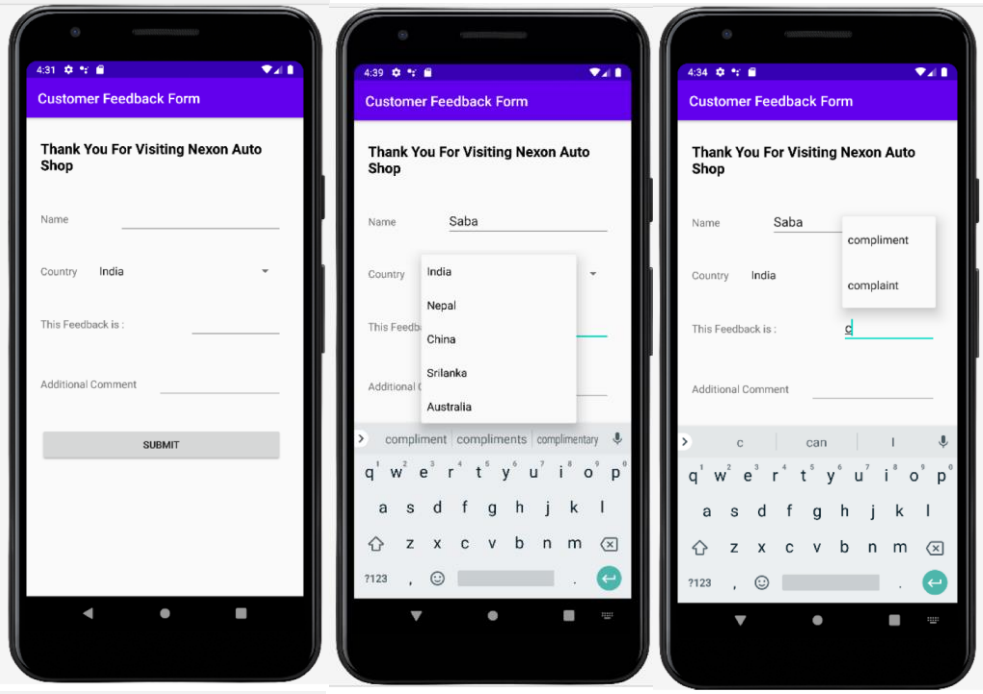
```
android:text="This Feedback is :"
android:textSize="15sp"/>
<AutoCompleteTextView
android:id="@+id/autoCompleteTextView"
android:layout_width="wrap_content"
android:layout_height="wrap_content"
android:layout_weight="1" />
</LinearLayout>
<LinearLayout
android:layout_width="match_parent"
android:layout_height="wrap_content"
android:orientation="horizontal"
android:layout_marginBottom="40dp">
<TextView
android:id="@+id/textView6"
android:layout_width="200dp"
android:layout_height="wrap_content"
android:layout_weight="1"
android:text="Additional Comment"
android:textSize="15sp"/>
<EditText
android:id="@+id/editTextTextMultiLine"
android:layout_width="wrap_content"
android:layout_height="wrap_content"
android:layout_weight="1"
android:ems="10"
android:gravity="start|top"
android:inputType="textMultiLine" />
</LinearLayout>
Basic Controls and UI
Component
<Button
android:id="@+id/button"
android:layout_width="match_parent"
android:layout_height="wrap_content"
android:text="Submit" />
</LinearLayout>
</android.support.constraint.ConstraintLayout>
```

- **MainActivity.java:**

```
package com.example.customerfeedbackform;
import android.support.v7.app.AppCompatActivity;
import android.os.Bundle;
import android.view.View;
import android.widget.AdapterView;
import android.widget.AdapterView.OnItemSelectedListener;
import android.widget.ArrayAdapter;
import android.widget.Spinner;
import android.widget.Toast;
import android.widget.AutoCompleteTextView;
public class MainActivity extends AppCompatActivity implements
OnItemSelectedListener{
String[] Country={"India","Nepal","China","Srilanka","Australia"};
String[] feedback = { "suggestion", "compliment", "complaint",
"other"};
@Override
protected void onCreate(Bundle savedInstanceState) {
super.onCreate(savedInstanceState);
setContentView(R.layout.activity_main);
Spinner spin = (Spinner) findViewById(R.id.spinner);
spin.setOnItemSelectedListener(this);
ArrayAdapter aa = new
ArrayAdapter(this,android.R.layout.simple_spinner_item,Country);
ArrayAdapter<String> adapter = new ArrayAdapter<String>(this,
android.R.layout.simple_dropdown_item_1line, feedback);
AutoCompleteTextView actv =
(AutoCompleteTextView)findViewById(R.id.autoCompleteTextView);
actv.setThreshold(1);
actv.setAdapter(adapter);
Basic Controls and UI
Component
aa.setDropDownViewResource(android.R.layout.simple_spinner_dropdo
wn_item);
//Setting the ArrayAdapter data on the Spinner
spin.setAdapter(aa);
}
@Override
public void onItemSelected(AdapterView<?> adapterView, View view,
int i, long l) {
}
```

```
@Override
public void onNothingSelected(AdapterView<?> adapterView) {
}
}
```

- **Output:**



Practical 3: DATA BASE CONNECTIVITY

A. Program to Create android program to demonstrate the use of Internal Storage.

- **Activity_main.xml:**

```
<?xml version="1.0" encoding="utf-8"?>
<android.support.constraint.ConstraintLayout
xmlns:android="http://schemas.android.com/apk/res/android"
xmlns:app="http://schemas.android.com/apk/res-auto"
xmlns:tools="http://schemas.android.com/tools"
android:layout_width="match_parent"
android:layout_height="match_parent"
tools:context=".MainActivity">
<LinearLayout
android:layout_width="match_parent"
android:layout_height="match_parent"
android:orientation="vertical"
android:layout_marginRight="20sp"
android:layout_marginLeft="20sp">

<TextView
android:id="@+id/textView"
android:layout_width="match_parent"
android:layout_height="wrap_content"
android:text="Internal Storage"
android:layout_marginTop="20sp"
android:layout_marginBottom="20sp"
android:textSize="40sp"
android:textColor="@color/black"
android:textStyle="bold"
/>

<EditText
android:id="@+id/editTextTextPersonName"
android:layout_width="match_parent"
android:layout_height="wrap_content"
android:ems="10"
android:inputType="textPersonName"
android:hint="Enter Text"
```

```
android:layout_marginTop="20sp"
android:layout_marginBottom="20sp"
android:textSize="30sp"/>

<TextView
android:id="@+id/textView2"
android:layout_width="match_parent"
android:layout_height="wrap_content"
android:hint="Read"
android:layout_marginTop="20sp"
android:layout_marginBottom="20sp"
android:textSize="30sp"/>

<LinearLayout
android:layout_width="match_parent"
android:layout_height="match_parent"
android:orientation="horizontal">

<Button
android:id="@+id/button"
android:layout_width="wrap_content"
android:layout_height="wrap_content"
android:layout_weight="1"
android:text="Save" />

<Button
android:id="@+id/button2"
android:layout_width="wrap_content"
android:layout_height="wrap_content"
android:layout_weight="1"
android:text="Load" />
</LinearLayout>
</LinearLayout>
</android.support.constraint.ConstraintLayout>
```

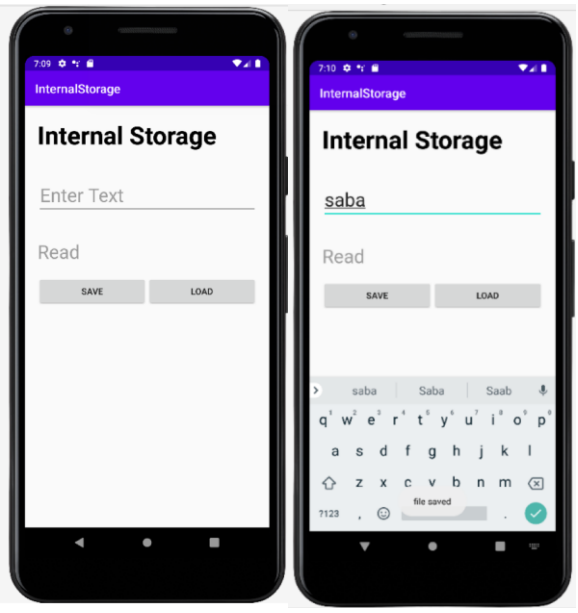
- **MainActivity.java:**

```
package com.example.internalstorage;
import android.support.v7.app.AppCompatActivity;
import android.os.Bundle;
```

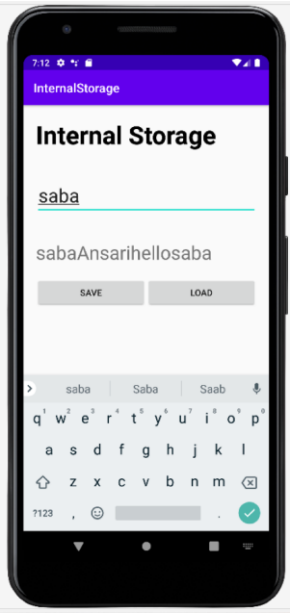
```
import android.view.View;
import android.widget.Button;
import android.widget.EditText;
import android.widget.TextView;
import android.widget.Toast;
import java.io.FileInputStream;
import java.io.FileOutputStream;

public class MainActivity extends AppCompatActivity {
Button b1,b2;
TextView tv;
EditText ed1;
String data;
private String file = "mydata";

@Override
protected void onCreate(Bundle savedInstanceState) {
super.onCreate(savedInstanceState);
setContentView(R.layout.activity_main);
b1=(Button)findViewById(R.id.button);
b2=(Button)findViewById(R.id.button2);
ed1=(EditText)findViewById(R.id.editTextTextPersonName);
tv=(TextView)findViewById(R.id.textView2);
b1.setOnClickListener(new View.OnClickListener() {
@Override
public void onClick(View v) {
data=ed1.getText().toString();
try {
FileOutputStream fOut =
openFileOutput(file,MODE_APPEND);
fOut.write(data.getBytes());
fOut.close();
Toast.makeText(getBaseContext(),"file
saved",Toast.LENGTH_SHORT).show();
}
catch (Exception e) {
// TODO Auto-generated catch block
e.printStackTrace();
}
}
```

```
});
b2.setOnClickListener(new View.OnClickListener() {
@Override
public void onClick(View v) {
try {
FileInputStream fin = openFileInput(file);
int c = 0;
String t="";
while( (c= fin.read()) != -1) t += Character.toString((char) c);
tv.setText(t);
Toast.makeText(getBaseContext(),"file
read",Toast.LENGTH_SHORT).show();
catch(Exception e){
}
}
}
}); }
}
```

- **Output:**



Once we click on load button, all the data saved in internal strorage file will be displayed, As we are using Append mode ,it is showing previous data also.



To view the file in internal storage click on Device File Explorer->Data->Data->com.example.internalstorage->files->myData

B. Program to Create android program to demonstrate the use of external Storage.

- **AndroidManifest.xml:**

```
<?xml version="1.0" encoding="utf-8"?>
<manifest xmlns:android="http://schemas.android.com/apk/res/android"
package="com.example.externalstrorage">
<uses-permission
android:name="android.permission.WRITE_EXTERNAL_STORAGE" />
<uses-permission
android:name="android.permission.READ_EXTERNAL_STORAGE" />
<application
android:allowBackup="true"
android:icon="@mipmap/ic_launcher"
android:label="@string/app_name"
android:roundIcon="@mipmap/ic_launcher_round"
android:supportsRtl="true"
android:theme="@style/Theme.ExternalStrorage">
<activity
android:name=".ViewData"
android:exported="false" />
<activity
android:name=".MainActivity"
android:exported="true">
<intent-filter>
<action android:name="android.intent.action.MAIN" />
<category
android:name="android.intent.category.LAUNCHER" />
</intent-filter>
</activity>
</application>
</manifest>
```

Once we have added the permission open activity_main.xml file from \res\layout folder path and write the code as shown below.

activity_main.xml

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

```
android:layout_height="match_parent"
tools:context=".ViewData">
<LinearLayout
android:layout_width="match_parent"
android:layout_height="match_parent"
android:orientation="vertical" android:layout_marginLeft="20sp"
android:layout_marginRight="20sp">
<TextView
android:id="@+id/textView3"
android:layout_width="match_parent"
android:layout_height="wrap_content"
android:text="Retreiving Saved Text Data"
android:layout_marginTop="30dp"
android:layout_marginBottom="30dp"/>
<TextView
android:id="@+id/textView4"
android:layout_width="match_parent"
android:layout_height="wrap_content"
android:text="TextView"
android:layout_marginTop="30dp"
android:layout_marginBottom="30dp"/>
<Button
android:id="@+id/button"
android:layout_width="match_parent"
android:layout_height="wrap_content"
android:text="Retreive Private data"
android:layout_marginTop="30dp"
android:layout_marginBottom="30dp"
android:onClick="showPrivateData"/>
<Button
android:id="@+id/button5"
android:layout_width="match_parent"
android:layout_height="wrap_content"
android:text="Retreive Public Data"
android:layout_marginTop="30dp"
android:layout_marginBottom="30dp"
android:onClick="showPublicData"/>
<Button
android:id="@+id/button6"
android:layout_width="match_parent"
```

```
android:layout_height="wrap_content"
android:text="Home"
android:layout_marginTop="30dp"
android:layout_marginBottom="30dp"
android:onClick="back"/>
</LinearLayout>
</androidx.constraintlayout.widget.ConstraintLayout>
```

- **Create new Empty Activity: Activity_View_Data.xml**

```
<?xml version="1.0" encoding="utf-8"?>
<androidx.constraintlayout.widget.ConstraintLayout
xmlns:android="http://schemas.android.com/apk/res/android"
xmlns:app="http://schemas.android.com/apk/res-auto"
xmlns:tools="http://schemas.android.com/tools"
android:layout_width="match_parent"
android:layout_height="match_parent"
tools:context=".ViewData">
<LinearLayout
android:layout_width="match_parent"
android:layout_height="match_parent"
android:orientation="vertical" android:layout_marginLeft="20sp"
android:layout_marginRight="20sp">
<TextView
android:id="@+id/textView3"
android:layout_width="match_parent"
android:layout_height="wrap_content"
android:text="Retreiving Saved Text Data"
android:layout_marginTop="30dp"
android:layout_marginBottom="30dp"/>
<TextView
android:id="@+id/textView4"
android:layout_width="match_parent"
android:layout_height="wrap_content"
android:text="TextView"
android:layout_marginTop="30dp"
android:layout_marginBottom="30dp"/>
<Button
android:id="@+id/button"
android:layout_width="match_parent"
android:layout_height="wrap_content"
```

```
android:text="Retreive Private data"
android:layout_marginTop="30dp"
android:layout_marginBottom="30dp"
android:onClick="showPrivateData"/>
<Button
android:id="@+id/button5"
android:layout_width="match_parent"
android:layout_height="wrap_content"
android:text="Retreive Public Data"
android:layout_marginTop="30dp"
android:layout_marginBottom="30dp"
android:onClick="showPublicData"/>
Data Base Connectivity
<Button
android:id="@+id/button6"
android:layout_width="match_parent"
android:layout_height="wrap_content"
android:text="Home"
android:layout_marginTop="30dp"
android:layout_marginBottom="30dp"
android:onClick="back"/>
</LinearLayout>
</androidx.constraintlayout.widget.ConstraintLayout>
```

- **MainActivity.java:**

```
package com.example.externalstrorage;
import static android.os.Environment.DIRECTORY_DOWNLOADS;
import androidx.appcompat.app.AppCompatActivity;
import android.Manifest;
import android.content.Intent;
import android.os.Bundle;
import android.os.Environment;
import android.view.View;
import android.widget.EditText;
import android.widget.Toast;
import androidx.core.app.ActivityCompat;
import java.io.File;
import java.io.FileOutputStream;
import java.io.IOException;
public class MainActivity extends AppCompatActivity {
```

```
private int EXTERNAL_STORAGE_PERMISSION_CODE = 23;
EditText editText;
@Override
protected void onCreate(Bundle savedInstanceState) {
super.onCreate(savedInstanceState);
setContentView(R.layout.activity_main);
editText = (EditText) findViewById(R.id.editTextTextMultiLine);
}
public void savePublicly(View view) {
// Requesting Permission to access External Storage
ActivityCompat.requestPermissions(this, new
String[]{Manifest.permission.READ_EXTERNAL_STORAGE},
EXTERNAL_STORAGE_PERMISSION_CODE);
String editTextData = editText.getText().toString();
// getExternalStoragePublicDirectory() represents root of external storage, we are using
DOWNLOADS
// We can use following directories: MUSIC, PODCASTS,
ALARMS, RINGTONES, NOTIFICATIONS, PICTURES, MOVIES
File folder =
Environment.getExternalStoragePublicDirectory(DIRECTORY_DOWNL
OADS);
// Storing the data in file with name as sabaext.txt
File file = new File(folder, "sabaext.txt");
writeTextData(file, editTextData);
editText.setText("");
}
public void savePrivately(View view) {
String editTextData = editText.getText().toString();
// Creating folder with name ExternalStorage
File folder = getExternalFilesDir("ExternalStorage");
// Creating file with name xyz.txt
File file = new File(folder, "xyz.txt");
writeTextData(file, editTextData);
editText.setText("");
}
public void viewInformation(View view) {
// Creating an intent to start a new activity
Intent intent = new Intent(MainActivity.this, ViewData.class);
startActivity(intent);
}
```

```
private void writeTextData(File file, String data) {
FileOutputStream fileOutputStream = null;
try {
fileOutputStream = new FileOutputStream(file);
fileOutputStream.write(data.getBytes());
Toast.makeText(this, "Done" + file.getAbsolutePath(),
Toast.LENGTH_SHORT).show();
} catch (Exception e) {
e.printStackTrace();
} finally {
if (fileOutputStream != null)
{
try {
fileOutputStream.close();
} catch (IOException e) {
e.printStackTrace();
}
}
}
}}
```

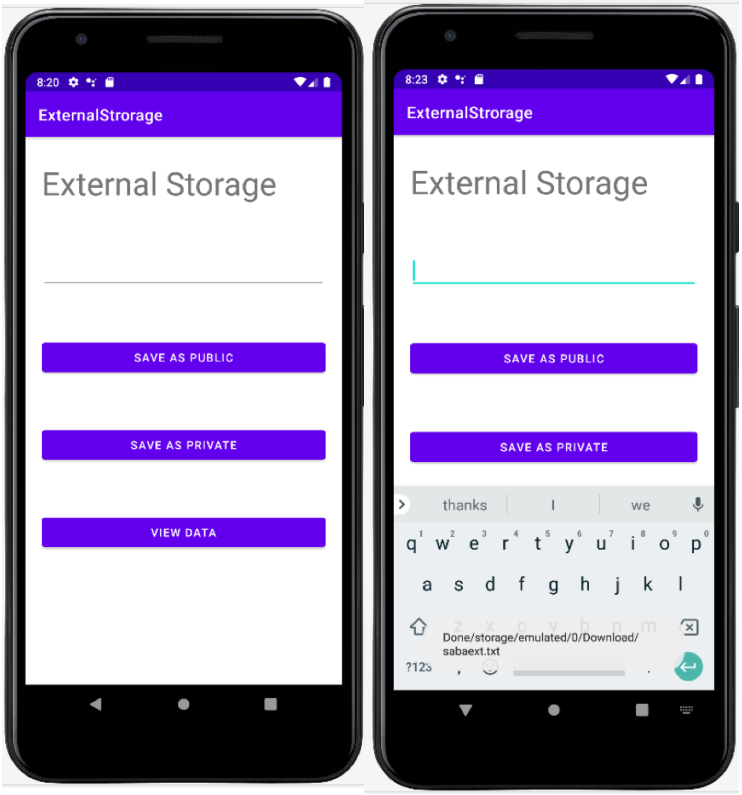
- **ViewData.java:**

```
package com.example.externalstrorage;
import static android.os.Environment.DIRECTORY_DOWNLOADS;
import androidx.appcompat.app.AppCompatActivity;
import android.content.Intent;
import android.os.Bundle;
import android.os.Environment;
import android.view.View;
import android.widget.TextView;
import java.io.File;
import java.io.FileInputStream;
import java.io.IOException;
public class ViewData extends AppCompatActivity {
TextView textView;
@Override
protected void onCreate(Bundle savedInstanceState) {
super.onCreate(savedInstanceState);
setContentView(R.layout.activity_view_data);
textView = (TextView) findViewById(R.id.textView4);
```

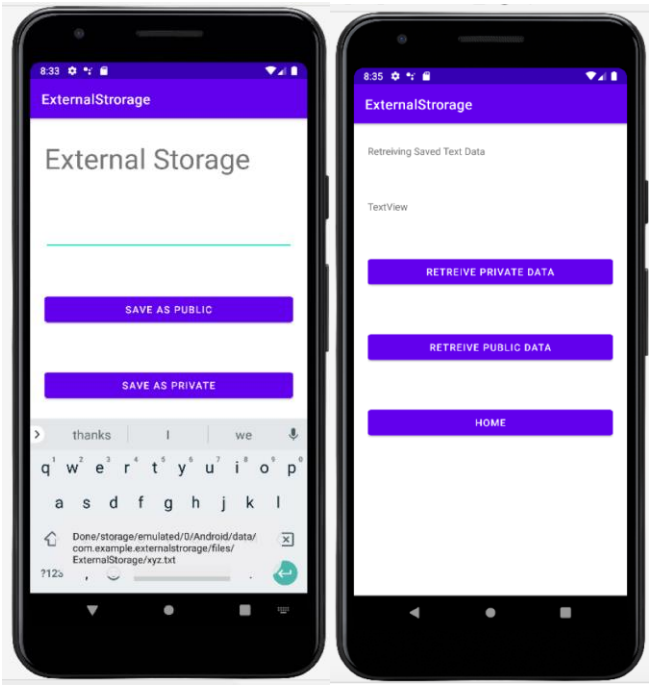
```
Data Base Connectivity
}
public void showPublicData(View view) {
// Accessing the saved data from the downloads folder
File folder =
Environment.getExternalStoragePublicDirectory(DIRECTORY_DOWNL
OADS);
// sabaext represent the file data that is saved publicly
File file = new File(folder, "sabaext.txt");
String data = getdata(file);
if (data != null) {
textView.setText(data);
} else {
textView.setText("No Data Found");
}
}
public void showPrivateData(View view) {
// ExternalStorage represent the folder name to access privately saved
data
File folder = getExternalFilesDir("ExternalStorage");
// xyz.txt is the file that is saved privately
File file = new File(folder, "xyz.txt");
String data = getdata(file);
if (data != null) {
textView.setText(data);
} else {
textView.setText("No Data Found");
}
}
public void back(View view) {
Intent intent = new Intent(ViewData.this, MainActivity.class);
startActivity(intent);
}
// getdata() is the method which reads the data
// the data that is saved in byte format in the file
private String getdata(File myfile) {
FileInputStream fileInputStream = null;
try {
fileInputStream = new FileInputStream(myfile);
int i = -1;
```

```
StringBuffer buffer = new StringBuffer();
while ((i = fileInputStream.read()) != -1) {
buffer.append((char) i);
}
return buffer.toString();
} catch (Exception e) {
e.printStackTrace();
} finally {
if (fileInputStream != null) {
try {
fileInputStream.close();
} catch (IOException e) {
e.printStackTrace();
}
}
}
return null;
}
}
```

- **Output:**

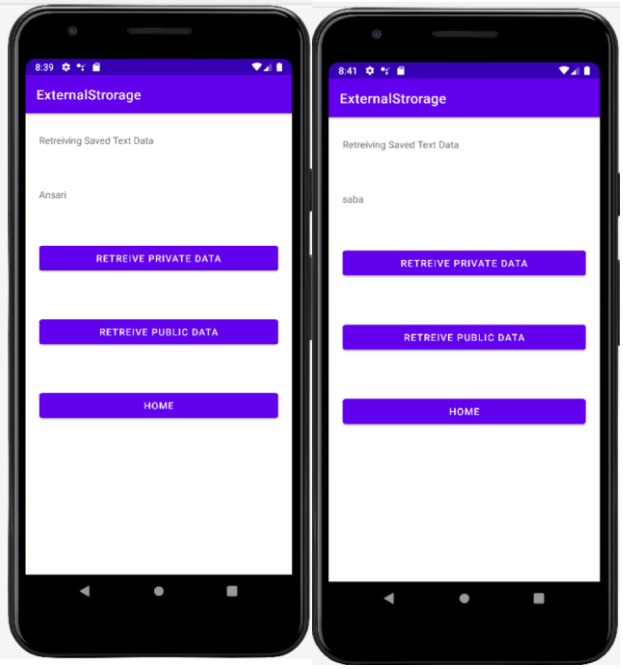


when Clicked on Save as Public button:
When Clicked on Save As Private Button:



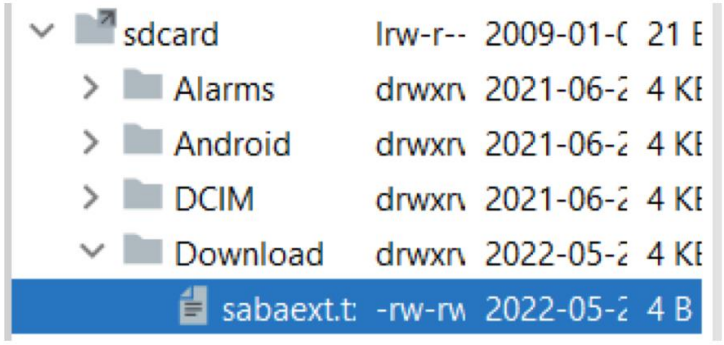
When clicked on View Data button is clicked another activity that view activity is started.

When clicked on retrieve Private data Button:



When we clicked on Retrieve public data button:

To View data saved as publicly click on Device File Explorer->SD Card->Downloads and you will see sabaext.txt file.



To View data saved as privately click on Device File Explorer->Storage->Android->Data->com.examples.externalstorage(your Application folder)->Files. In files folder you can see folder is created

C. Create an activity with all possible UI Components (Edit Text, Text view, Radio Button, Check Box, and Rating Bar). Get the values from the components and display it on the next activity.

- **activity_main.xml:**

```
<?xml version="1.0" encoding="utf-8"?>
<ScrollView xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:app="http://schemas.android.com/apk/res-auto"
    xmlns:tools="http://schemas.android.com/tools"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    android:background="#2596BE"
    tools:context=".MainActivity">
    <LinearLayout
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:orientation="vertical">
        <Space
            android:layout_width="match_parent"
            android:layout_height="32dp" />
        <TextView
            android:id="@+id/textView"
            android:layout_width="match_parent"
            android:layout_height="53dp"
            android:text="REGISTRATION"
            android:textAlignment="center"
            android:textColor="@android:color/white"
            android:textSize="30sp"
            android:textStyle="bold" />
        <Space
            android:layout_width="match_parent"
            android:layout_height="38dp" />
        <EditText
            android:id="@+id/name"
            android:layout_width="match_parent"
            android:layout_height="wrap_content"
            android:backgroundTint="@android:color/white"
            android:ems="10"
            android:hint="Enter Name"
            android:inputType="textPersonName"
```

```
    android:minHeight="48dp"
    android:textColor="@color/white" />
<Space
    android:layout_width="match_parent"
    android:layout_height="22dp" />
<EditText
    android:id="@+id/roll"
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:backgroundTint="@color/white"
    android:ems="10"
    android:hint="Enter Roll No"
    android:inputType="number"
    android:minHeight="48dp"
    android:textColor="@android:color/white" />
<Space
    android:layout_width="match_parent"
    android:layout_height="22dp" />
<EditText
    android:id="@+id/dept"
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:backgroundTint="@android:color/white"
    android:ems="10"
    android:hint="Enter Department"
    android:inputType="textPersonName"
    android:minHeight="48dp"
    android:textColor="@android:color/white" />
<Space
    android:layout_width="match_parent"
    android:layout_height="22dp" />
<TextView
    android:id="@+id/textView2"
    android:layout_width="match_parent"
    android:layout_height="29dp"
    android:minHeight="48dp"
    android:text="Select Shift:"
    android:textColor="@android:color/white"
    android:textSize="18sp" />
<RadioGroup
```

```
        android:layout_width="155dp"
        android:layout_height="wrap_content"
        android:id="@+id/radiogrp">
        <RadioButton
            android:id="@+id/ra1"
            android:layout_width="match_parent"
            android:layout_height="wrap_content"
            android:text="Morning"
            android:textColor="@android:color/white"
            android:textSize="16sp" />
        <RadioButton
            android:id="@+id/ra2"
            android:layout_width="match_parent"
            android:layout_height="wrap_content"
            android:text="Afternoon"
            android:textColor="@android:color/white"
            android:textSize="16sp" />
    </RadioGroup>
    <Space
        android:layout_width="match_parent"
        android:layout_height="22dp" />
    <TextView
        android:id="@+id/textView3"
        android:layout_width="match_parent"
        android:layout_height="27dp"
        android:minHeight="48dp"
        android:text="Select Hobbies:"
        android:textColor="@android:color/white"
        android:textSize="18sp" />
    <CheckBox
        android:id="@+id/dan"
        android:layout_width="128dp"
        android:layout_height="wrap_content"
        android:text="Dance"
        android:textColor="@android:color/white" />
    <CheckBox
        android:id="@+id/sin"
        android:layout_width="128dp"
        android:layout_height="wrap_content"
        android:text="Sing"
```

```
    android:textColor="@android:color/white" />
<CheckBox
    android:id="@+id/pla"
    android:layout_width="128dp"
    android:layout_height="wrap_content"
    android:text="Play"
    android:textColor="@android:color/white" />
<Space
    android:layout_width="match_parent"
    android:layout_height="22dp" />
<TextView
    android:id="@+id/textView4"
    android:layout_width="match_parent"
    android:layout_height="29dp"
    android:minHeight="48dp"
    android:text="Rate Me:"
    android:textColor="@android:color/white"
    android:textSize="18sp" />
<RatingBar
    android:id="@+id/ratingBar"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content" />
<Space
    android:layout_width="match_parent"
    android:layout_height="35dp" />
<Button
    android:id="@+id/reg"
    android:layout_width="355dp"
    android:layout_height="59dp"
    android:backgroundTint="@android:color/white"
    android:insetLeft="60dp"
    android:onClick="onClick"
    android:text="Register"
    android:textColor="#2596BE"
    android:textSize="18sp"
    android:textStyle="bold"
    tools:ignore="OnClick" />
<Space
    android:layout_width="match_parent"
    android:layout_height="35dp" />
```

```
    </LinearLayout>
</ScrollView>
```

- **activity_display_data.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"
    android:background="#2596BE"
    tools:ignore="SpeakableTextPresentCheck">
    <LinearLayout
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:orientation="vertical">
        <Space
            android:layout_width="match_parent"
            android:layout_height="37dp" />
        <TextView
            android:id="@+id/textView5"
            android:layout_width="match_parent"
            android:layout_height="wrap_content"
            android:text="REGISTRATION SUCCESS"
            android:textAlignment="center"
            android:textColor="@android:color/white"
            android:textSize="30sp"
            android:textStyle="bold" />
        <Space
            android:layout_width="match_parent"
            android:layout_height="58dp" />
        <TextView
            android:id="@+id/t1"
            android:layout_width="match_parent"
            android:layout_height="31dp"
            android:minHeight="48dp"
            android:textAlignment="center"
            android:textColor="@android:color/white"
            android:textSize="25sp"
            android:textStyle="bold" />
        <Space
```

```
    android:layout_width="match_parent"
    android:layout_height="37dp" />
<TextView
    android:id="@+id/t2"
    android:layout_width="match_parent"
    android:layout_height="31dp"
    android:minHeight="48dp"
    android:textAlignment="center"
    android:textColor="@android:color/white"
    android:textSize="25sp"
    android:textStyle="bold" />
<Space
    android:layout_width="match_parent"
    android:layout_height="37dp" />
<TextView
    android:id="@+id/t3"
    android:layout_width="match_parent"
    android:layout_height="31dp"
    android:minHeight="48dp"
    android:textAlignment="center"
    android:textColor="@android:color/white"
    android:textSize="25sp"
    android:textStyle="bold" />
<Space
    android:layout_width="match_parent"
    android:layout_height="37dp" />
<TextView
    android:id="@+id/t4"
    android:layout_width="match_parent"
    android:layout_height="31dp"
    android:minHeight="48dp"
    android:textAlignment="center"
    android:textColor="@android:color/white"
    android:textSize="25sp"
    android:textStyle="bold" />
<Space
    android:layout_width="match_parent"
    android:layout_height="37dp" />
<TextView
    android:id="@+id/t9"
```

```
        android:layout_width="match_parent"
        android:layout_height="31dp"
        android:minHeight="48dp"
        android:text="Hobbies:"
        android:scrollbars = "vertical"
        android:textAlignment="center"
        android:textColor="@android:color/white"
        android:textSize="25sp"
        android:textStyle="bold" />
    <TextView
        android:id="@+id/t5"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:minHeight="48dp"
        android:scrollbarAlwaysDrawVerticalTrack="true"
        android:textAlignment="center"
        android:textColor="@android:color/white"
        android:textSize="25sp"
        android:textStyle="bold" />
    <Space
        android:layout_width="match_parent"
        android:layout_height="37dp" />
    <TextView
        android:id="@+id/t6"
        android:layout_width="match_parent"
        android:layout_height="31dp"
        android:minHeight="48dp"
        android:textAlignment="center"
        android:textColor="@android:color/white"
        android:textSize="25sp"
        android:textStyle="bold" />
  </LinearLayout>
</ScrollView>
```

- **MainActivity.java:**

```
package com.example.practical3_a;
import androidx.appcompat.app.AppCompatActivity;
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.RatingBar;
import android.widget.TextView;
import java.util.ArrayList;
public class MainActivity extends AppCompatActivity {
Button send_button;
RatingBar ratingBar;
RadioGroup rg1;
RadioButton rb1,rb2;
CheckBox c1,c2,c3;
EditText object1,object2,object3;
String shift;
ArrayList<String> hobby = new ArrayList<>();
   @Override
   protected void onCreate(Bundle savedInstanceState) {
      super.onCreate(savedInstanceState);
      setContentView(R.layout.activity_main);
      send_button = findViewById(R.id.reg);
      object1 =  findViewById(R.id.name);
      object2 = findViewById(R.id.roll);
      object3 =  findViewById(R.id.dept);
      ratingBar = findViewById(R.id.ratingBar);
      rg1 = findViewById(R.id.radiogrp);
      c1 = findViewById(R.id.dan);
      c2 = findViewById(R.id.sin);
      c3 = findViewById(R.id.pla);
      send_button.setOnClickListener(new View.OnClickListener() {
         public void onClick(View v) {
            String name = object1.getText().toString();
            String roll = object2.getText().toString();
            String dept = object3.getText().toString();
            String rate = String.valueOf(ratingBar.getRating());
            rb1 = findViewById(R.id.ra1);
            rb2 = findViewById(R.id.ra2);
            if (rb1.isChecked()) {
               shift = "Morning";
```

```
            }
            if (rb2.isChecked()) {
               shift = "Afternoon";
            }
            if (c1.isChecked()) {
               hobby.add(c1.getText().toString());
            }
            if (c2.isChecked()) {
               hobby.add((String) c2.getText());
            }
            if (c3.isChecked()) {
               hobby.add((String) c3.getText());
            }
            Intent intent = new Intent(getApplicationContext(), DisplayData.class);
            intent.putExtra("nam", name);
            intent.putExtra("roll", roll);
            intent.putExtra("dept", dept);
            intent.putExtra("rat", rate);
            intent.putExtra("shi", shift);
            intent.putExtra("hob", hobby);
            startActivity(intent);
         }
      } );  }}
```

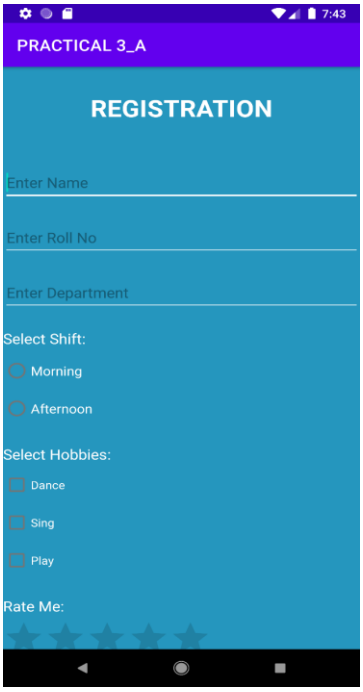
- **DisplayData.java:**

```
package com.example.practical3_a;
import androidx.appcompat.app.AppCompatActivity;
import android.content.Intent;
import android.os.Bundle;
import android.text.method.ScrollingMovementMethod;
import android.widget.ArrayAdapter;
import android.widget.ListView;
import android.widget.TextView;
import java.util.ArrayList;
public class DisplayData extends AppCompatActivity {
TextView tv1,tv2,tv3,tv4,tv6,tv5;
   @Override
   protected void onCreate(Bundle savedInstanceState) {
      super.onCreate(savedInstanceState);
      setContentView(R.layout.activity_display_data);
```

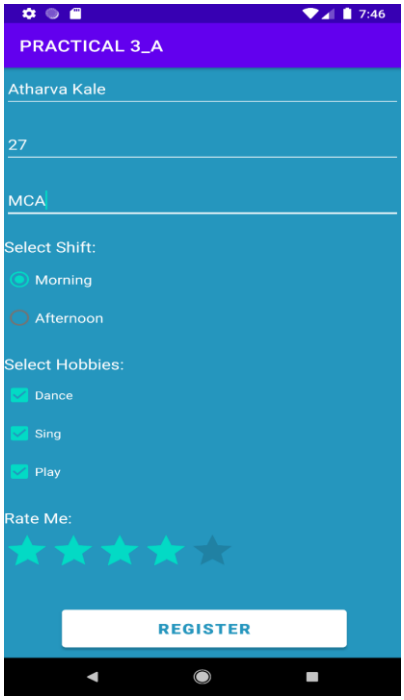
```
        tv1=findViewById(R.id.t1);
        tv2=findViewById(R.id.t2);
        tv3=findViewById(R.id.t3);
        tv4=findViewById(R.id.t4);
        tv5=findViewById(R.id.t5);
        tv6=findViewById(R.id.t6);
        Intent intent = getIntent();
        String str1 = intent.getStringExtra("nam");
        tv1.setText("Name: "+str1);
        String str2 = intent.getStringExtra("roll");
        tv2.setText("Roll No: "+str2);
        String str3 = intent.getStringExtra("dept");
        tv3.setText("Department: "+str3);
        String str4 = intent.getStringExtra("shi");
        tv4.setText("Shift: "+str4);
        ArrayList<String> myList = (ArrayList<String>) getIntent().getSerializableExtra("hob");
        tv5.setText("");
        for (int j = 0; j < myList.size(); j++){
            tv5.append( myList.get(j) + "\n");
        }
        tv5.setMovementMethod(new ScrollingMovementMethod());
        String str5 = intent.getStringExtra("rat");
        tv6.setText("Rating: "+str5);
    }
}
```

- **Output:**

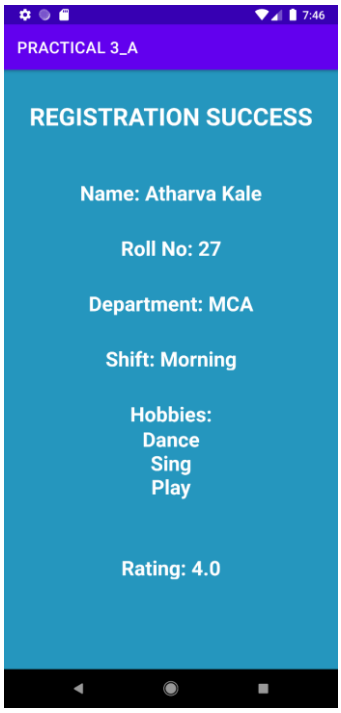
GUI



After Filling Data



Display Data



D. Create a login form. For successful login, display welcome page and in case of failure display alert box indicating error message and attempts made. Disable submit button after 3 wrong attempts and display the alert message indicating the same.

- **activity_main.xml:**

```
<?xml version="1.0" encoding="utf-8"?>
<androidx.constraintlayout.widget.ConstraintLayout
xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:app="http://schemas.android.com/apk/res-auto"
    xmlns:tools="http://schemas.android.com/tools"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    tools:context=".MainActivity">
    <TextView
        android:id="@+id/textView"
        android:layout_width="182dp"
        android:layout_height="49dp"
        android:text="LOGIN"
        android:textAlignment="center"
        android:textSize="30sp"
        android:textStyle="bold"
        app:layout_constraintBottom_toBottomOf="parent"
        app:layout_constraintEnd_toEndOf="parent"
        app:layout_constraintHorizontal_bias="0.497"
        app:layout_constraintStart_toStartOf="parent"
        app:layout_constraintTop_toTopOf="parent"
        app:layout_constraintVertical_bias="0.043" />
    <EditText
        android:id="@+id/user"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:ems="10"
        android:hint="Enter Username"
        android:inputType="textPersonName"
        app:layout_constraintBottom_toBottomOf="parent"
        app:layout_constraintEnd_toEndOf="parent"
        app:layout_constraintHorizontal_bias="0.497"
        app:layout_constraintStart_toStartOf="parent"
        app:layout_constraintTop_toTopOf="parent"
        app:layout_constraintVertical_bias="0.243"
```

```
    tools:ignore="MissingConstraints" />
  <EditText
    android:id="@+id/pass"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:ems="10"
    android:hint="Enter Password"
    android:inputType="textPassword"
    app:layout_constraintBottom_toBottomOf="parent"
    app:layout_constraintEnd_toEndOf="parent"
    app:layout_constraintHorizontal_bias="0.497"
    app:layout_constraintStart_toStartOf="parent"
    app:layout_constraintTop_toTopOf="parent"
    app:layout_constraintVertical_bias="0.392"
    tools:ignore="MissingConstraints" />
  <Button
    android:id="@+id/log"
    android:layout_width="106dp"
    android:layout_height="52dp"
    android:onClick="onClick"
    android:text="LOGIN"
    app:layout_constraintBottom_toBottomOf="parent"
    app:layout_constraintEnd_toEndOf="parent"
    app:layout_constraintHorizontal_bias="0.498"
    app:layout_constraintStart_toStartOf="parent"
    app:layout_constraintTop_toTopOf="parent"
    app:layout_constraintVertical_bias="0.555"
    tools:ignore="MissingConstraints,OnClick" />
</androidx.constraintlayout.widget.ConstraintLayout>
```

- **activity_welcome_page.xml:**

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

```
    android:id="@+id/textView2"
    android:layout_width="359dp"
    android:layout_height="96dp"
    android:textAlignment="center"
    android:textSize="30sp"
    android:textStyle="bold"
    tools:layout_editor_absoluteX="40dp"
    tools:layout_editor_absoluteY="347dp"
    tools:ignore="MissingConstraints" />
</androidx.constraintlayout.widget.ConstraintLayout>
```

- **MainActivity.java:**

```
package com.example.practical3_b;
import androidx.appcompat.app.AppCompatActivity;
import android.content.Intent;
import android.os.Bundle;
import android.view.View;
import android.widget.Button;
import android.widget.EditText;
import android.widget.Toast;
public class MainActivity extends AppCompatActivity {
   int count=0;
   EditText u,p;
   Button b;
   View bu;
   @Override
   protected void onCreate(Bundle savedInstanceState) {
      super.onCreate(savedInstanceState);
      setContentView(R.layout.activity_main);
      u=findViewById(R.id.user);
      p=findViewById(R.id.pass);
      b=findViewById(R.id.log);
      bu = findViewById(R.id.log);}

   public void onClick(View view) {
      String us = u.getText().toString();
      String pa = p.getText().toString();
      if (count<2)
      {
         if(us.equals("Atharva") && pa.equals("Kale"))
```

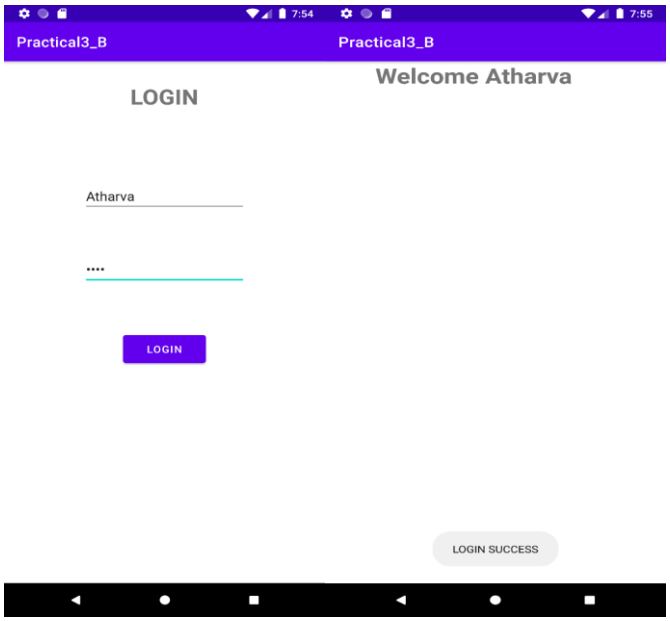
```
        {
            Toast.makeText(MainActivity.this, "LOGIN SUCCESS",
Toast.LENGTH_SHORT).show();
            Intent intent = new Intent(getApplicationContext(), WelcomePage.class);
            intent.putExtra("nam", u.getText().toString());
            startActivity(intent);
        }
        else
        {
            count+=1;
            Toast.makeText(MainActivity.this, "LOGIN FAILED!!",
Toast.LENGTH_SHORT).show();
            Toast.makeText(MainActivity.this, "Remaining Attempts: "+(3-count),
Toast.LENGTH_SHORT).show();
        }
      }
      else
      {
        b.setClickable(false);
        bu.setVisibility(View.GONE);
        Toast.makeText(MainActivity.this, "LOGIN FAILED!! TRY AGAIN AFTER SOME
TIME!!", Toast.LENGTH_SHORT).show();
      } } }
```

- **WelcomePage.java:**

```
package com.example.practical3_b;
import androidx.appcompat.app.AppCompatActivity;
import android.content.Intent;
import android.os.Bundle;
import android.widget.TextView;
public class WelcomePage extends AppCompatActivity {
   @Override
   protected void onCreate(Bundle savedInstanceState) {
      super.onCreate(savedInstanceState);
      setContentView(R.layout.activity_welcome_page);
      TextView t1 = (TextView) findViewById(R.id.textView2);
      Intent intent = getIntent();
      String str1 = intent.getStringExtra("nam");
      t1.setText("Welcome "+str1);
} }
```

- **Output:**

Correct Username and Password



Failed Attempts and Button Disabled:

Practical 4: DATA BASE CONNECTIVITY

A. Program to create a file in a directory and perform following file operations.
   a. Write into a file
   b. Read from a file
   c. Delete a file

- **activity_main.xml:**

```
<?xml version="1.0" encoding="utf-8"?>
<androidx.constraintlayout.widget.ConstraintLayout
xmlns:android="http://schemas.android.com/apk/res/android"
   xmlns:app="http://schemas.android.com/apk/res-auto"
   xmlns:tools="http://schemas.android.com/tools"
   android:layout_width="match_parent"
   android:layout_height="match_parent"
   tools:context=".MainActivity">
   <EditText
      android:id="@+id/to_file"
      android:layout_width="405dp"
      android:layout_height="146dp"
      android:ems="10"
      android:inputType="textMultiLine"
      android:hint="Enter Text To Add To File"
      app:layout_constraintBottom_toBottomOf="parent"
      app:layout_constraintEnd_toEndOf="parent"
      app:layout_constraintHorizontal_bias="0.427"
      app:layout_constraintStart_toStartOf="parent"
      app:layout_constraintTop_toTopOf="parent"
      app:layout_constraintVertical_bias="0.023" />
   <Button
      android:id="@+id/insert"
      android:layout_width="121dp"
      android:layout_height="60dp"
      android:onClick="write"
      android:text="INSERT"
      app:layout_constraintBottom_toBottomOf="parent"
      app:layout_constraintEnd_toEndOf="parent"
      app:layout_constraintStart_toStartOf="parent"
      app:layout_constraintTop_toTopOf="parent"
```

```
    app:layout_constraintVertical_bias="0.302" />
  <Button
    android:id="@+id/del"
    android:layout_width="121dp"
    android:layout_height="60dp"
    android:text="DELETE"
    android:onClick="delete"
    app:layout_constraintBottom_toBottomOf="parent"
    app:layout_constraintEnd_toEndOf="parent"
    app:layout_constraintStart_toStartOf="parent"
    app:layout_constraintTop_toTopOf="parent"
    app:layout_constraintVertical_bias="0.931" />
  <Button
    android:id="@+id/read"
    android:layout_width="121dp"
    android:layout_height="60dp"
    android:text="READ"
    android:onClick="read"
    app:layout_constraintBottom_toBottomOf="parent"
    app:layout_constraintEnd_toEndOf="parent"
    app:layout_constraintStart_toStartOf="parent"
    app:layout_constraintTop_toTopOf="parent"
    app:layout_constraintVertical_bias="0.701" />
  <TextView
    android:id="@+id/fro_file"
    android:layout_width="405dp"
    android:layout_height="124dp"
    android:text="TextView"
    app:layout_constraintBottom_toBottomOf="parent"
    android:singleLine="false"
    app:layout_constraintEnd_toEndOf="parent"
    app:layout_constraintStart_toStartOf="parent"
    app:layout_constraintTop_toTopOf="parent" />
</androidx.constraintlayout.widget.ConstraintLayout>
```

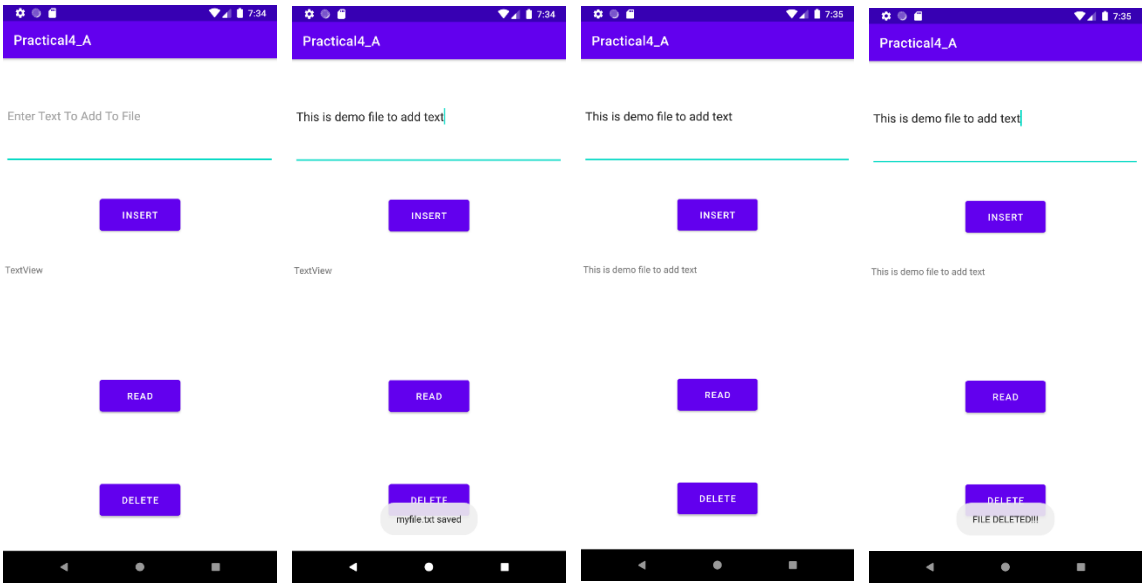
- **MainActivity.java:**

```
package com.example.practical4_a;
import androidx.appcompat.app.AppCompatActivity;
import android.content.Context;
import android.os.Bundle;
```

```
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.FileNotFoundException;
import java.io.FileOutputStream;
import java.io.IOException;
import java.io.InputStream;
import java.io.InputStreamReader;
import java.io.OutputStreamWriter;
public class MainActivity extends AppCompatActivity {
   Button ins,red,del;
   TextView t1;
   EditText e1;
   @Override
   protected void onCreate(Bundle savedInstanceState) {
      super.onCreate(savedInstanceState);
      setContentView(R.layout.activity_main);

      t1=findViewById(R.id.fro_file);
      e1=findViewById(R.id.to_file);
      ins=findViewById(R.id.insert);
      red=findViewById(R.id.read);
      del=findViewById(R.id.del);
   }
   public void write(View v) {
      String filename="myfile.txt";
      String data=e1.getText().toString();
FileOutputStream fos;
      try {
         fos = openFileOutput(filename, Context.MODE_APPEND);
         //default mode is PRIVATE, can be APPEND etc.
         fos.write(data.getBytes());
         fos.close();

         Toast.makeText(getApplicationContext(),filename + " saved",
```

```
                Toast.LENGTH_LONG).show();
        } catch (FileNotFoundException e) {Toast.makeText(getApplicationContext(),"FILE NOT
FOUND",
                Toast.LENGTH_LONG).show();;}
        catch (IOException e) {e.printStackTrace();}
    }
    public void delete(View v) {
        deleteFile("myfile.txt");
        Toast.makeText(getApplicationContext(),"FILE DELETED!!!",
Toast.LENGTH_LONG).show();
    }
    public void read(View view) {
        StringBuffer stringBuffer = new StringBuffer();
        try {
            //Attaching BufferedReader to the FileInputStream by the help of InputStreamReader
            BufferedReader inputReader = new BufferedReader(new
InputStreamReader(openFileInput("myfile.txt")));
            String inputString;
            //Reading data line by line and storing it into the stringbuffer
            while ((inputString = inputReader.readLine()) != null) {
                stringBuffer.append(inputString + "\n");
            }
            t1.setText(stringBuffer.toString());
        } catch (FileNotFoundException e)
        {Toast.makeText(getApplicationContext(),"FILE NOT FOUND",
Toast.LENGTH_LONG).show();}
        catch (IOException e) {
            e.printStackTrace();
        }
    }
    }
```

- **Output:**


Practical4_A
Enter Text To Add To File
INSERT
TextView
READ
DELETE
Practical4_A
This is demo file to add text
INSERT
TextView
READ
DELETE
myfile.txt saved
Practical4_A
This is demo file to add text
INSERT
This is demo file to add text
READ
DELETE
Practical4_A
This is demo file to add text
INSERT
This is demo file to add text
READ
DELETE
FILE DELETED!!!

B. Create a new project and create a login Activity. In this create a login UI asking user email and password with an option of remember me checkbox. Also a button displaying Sign In or Register using shared preferences

- **activity_main.xml:**

```
<?xml version="1.0" encoding="utf-8"?>
<androidx.constraintlayout.widget.ConstraintLayout
xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:app="http://schemas.android.com/apk/res-auto"
    xmlns:tools="http://schemas.android.com/tools"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    tools:context=".MainActivity">
    <TextView
        android:id="@+id/textView"
        android:layout_width="182dp"
        android:layout_height="49dp"
        android:text="LOGIN"
        android:textAlignment="center"
        android:textSize="30sp"
        android:textStyle="bold"
        app:layout_constraintBottom_toBottomOf="parent"
        app:layout_constraintEnd_toEndOf="parent"
        app:layout_constraintHorizontal_bias="0.497"
        app:layout_constraintStart_toStartOf="parent"
        app:layout_constraintTop_toTopOf="parent"
        app:layout_constraintVertical_bias="0.043" />
    <EditText
        android:id="@+id/pass"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:ems="10"
        android:hint="Enter Password"
        android:inputType="textPassword"
        android:minHeight="48dp"
        app:layout_constraintBottom_toBottomOf="parent"
        app:layout_constraintEnd_toEndOf="parent"
        app:layout_constraintHorizontal_bias="0.497"
        app:layout_constraintStart_toStartOf="parent"
        app:layout_constraintTop_toTopOf="parent"
```

```
    app:layout_constraintVertical_bias="0.341"
    tools:ignore="MissingConstraints" />
  <Button
    android:id="@+id/log"
    android:layout_width="106dp"
    android:layout_height="52dp"
    android:onClick="onClick"
    android:text="LOGIN"
    app:layout_constraintBottom_toBottomOf="parent"
    app:layout_constraintEnd_toEndOf="parent"
    app:layout_constraintHorizontal_bias="0.498"
    app:layout_constraintStart_toStartOf="parent"
    app:layout_constraintTop_toTopOf="parent"
    app:layout_constraintVertical_bias="0.586"
    tools:ignore="MissingConstraints,OnClick" />
  <CheckBox
    android:id="@+id/share"
    android:layout_width="145dp"
    android:layout_height="48dp"
    android:text="Remember Me"
    app:layout_constraintBottom_toBottomOf="parent"
    app:layout_constraintEnd_toEndOf="parent"
    app:layout_constraintHorizontal_bias="0.375"
    app:layout_constraintStart_toStartOf="parent"
    app:layout_constraintTop_toTopOf="parent"
    app:layout_constraintVertical_bias="0.459" /
  <EditText
    android:id="@+id/user"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:ems="10"
    android:hint="Enter E-Mail"
    android:inputType="textEmailAddress"
    android:minHeight="48dp"
    app:layout_constraintBottom_toBottomOf="parent"
    app:layout_constraintEnd_toEndOf="parent"
    app:layout_constraintHorizontal_bias="0.497"
    app:layout_constraintStart_toStartOf="parent"
    app:layout_constraintTop_toTopOf="parent"
    app:layout_constraintVertical_bias="0.239"
```

```
    tools:ignore="MissingConstraints" />
</androidx.constraintlayout.widget.ConstraintLayout>
```

- activity_display.xml:

```
<?xml version="1.0" encoding="utf-8"?>
<androidx.constraintlayout.widget.ConstraintLayout
xmlns:android="http://schemas.android.com/apk/res/android"
   xmlns:app="http://schemas.android.com/apk/res-auto"
   xmlns:tools="http://schemas.android.com/tools"
   android:layout_width="match_parent"
   android:layout_height="match_parent"
   tools:context=".Display">
   <TextView
      android:id="@+id/textView2"
      android:layout_width="355dp"
      android:layout_height="113dp"
      android:textAlignment="center"
      android:textSize="30sp"
      android:textStyle="bold"
      app:layout_constraintBottom_toBottomOf="parent"
      app:layout_constraintEnd_toEndOf="parent"
      app:layout_constraintStart_toStartOf="parent"
      app:layout_constraintTop_toTopOf="parent" />
   <Button
      android:id="@+id/button"
      android:layout_width="wrap_content"
      android:layout_height="wrap_content"
      android:text="Clear Data"
      android:onClick="onClick"
      app:layout_constraintBottom_toBottomOf="parent"
      app:layout_constraintEnd_toEndOf="parent"
      app:layout_constraintHorizontal_bias="0.498"
      app:layout_constraintStart_toStartOf="parent"
      app:layout_constraintTop_toTopOf="parent"
      app:layout_constraintVertical_bias="0.865" />
</androidx.constraintlayout.widget.ConstraintLayout>
```

- **MainActivity.java:**

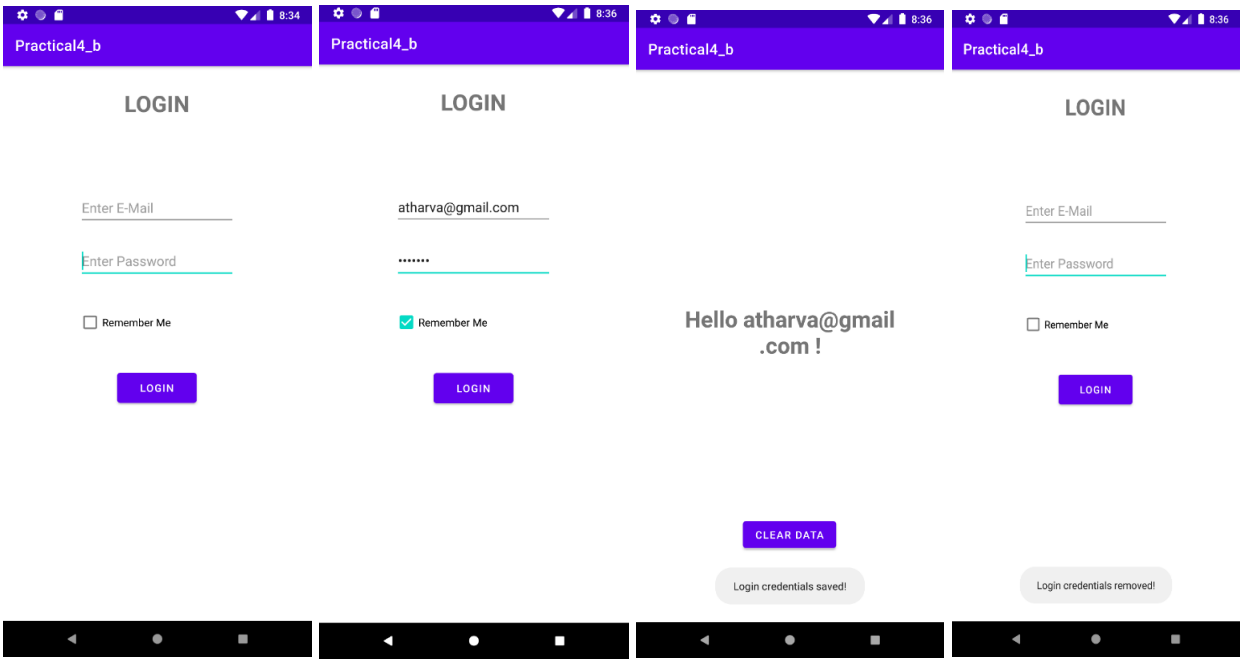
```
package com.example.practical4_b;
import androidx.appcompat.app.AppCompatActivity;
import android.content.Intent;
import android.content.SharedPreferences;
import android.os.Bundle;
import android.view.View;
import android.widget.CheckBox;
import android.widget.EditText;
import android.widget.Toast;
public class MainActivity extends AppCompatActivity {
   EditText user,pass;
   CheckBox cb;
   SharedPreferences sharedPreferences;
   boolean isRemembered;
   @Override
   protected void onCreate(Bundle savedInstanceState) {
      super.onCreate(savedInstanceState);
      setContentView(R.layout.activity_main);
      user = findViewById(R.id.user);
      pass = findViewById(R.id.pass);
      cb = findViewById(R.id.share);
      sharedPreferences = getSharedPreferences("SHARED_PREF", MODE_PRIVATE);
      isRemembered = sharedPreferences.getBoolean("Checkbox", false);
      if(isRemembered){
         Intent intent = new Intent(MainActivity.this,Display.class);
         startActivity(intent);
         finish();
      }
   }
   public void onClick(View view) {
      String name = user.getText().toString();
      boolean checked = cb.isChecked();
      SharedPreferences.Editor editor = sharedPreferences.edit();
      Toast.makeText(MainActivity.this,"Login credentials
saved!",Toast.LENGTH_SHORT).show();
      editor.putString("Name",name);
      editor.putBoolean("Checkbox",checked);
      editor.apply();
```

```
        Intent intent = new Intent(MainActivity.this,Display.class);
        startActivity(intent);
        finish();
    }
    }
```

- **Display.java:**

```
package com.example.practical4_b;
import androidx.appcompat.app.AppCompatActivity;
import android.content.Intent;
import android.content.SharedPreferences;
import android.os.Bundle;
import android.view.View;
import android.widget.TextView;
import android.widget.Toast;
public class Display extends AppCompatActivity {
    SharedPreferences preferences;
    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_display);
        TextView tv;

        tv = findViewById(R.id.textView2);
        preferences=getSharedPreferences("SHARED_PREF",MODE_PRIVATE);
        String name=preferences.getString("Name","");
        tv.setText("Hello "+name+" !");
    }
    public void onClick(View view) {
        SharedPreferences.Editor editor = preferences.edit();
        editor.clear();
        editor.apply();
        Intent intent = new Intent(Display.this, MainActivity.class);
        Toast.makeText(Display.this,"Login credentials
removed!",Toast.LENGTH_SHORT).show();
        startActivity(intent);
        finish();
    }
}
```

- **Output:**


Practical4_b
LOGIN
Enter E-Mail
Enter Password
Remember Me
LOGIN
Practical4_b
LOGIN
atharva@gmail.com
Remember Me
LOGIN
Practical4_b
Hello atharva@gmail .com !
CLEAR DATA
Login credentials saved!
Practical4_b
LOGIN
Enter E-Mail
Enter Password
Remember Me
LOGIN
Login credentials removed!

PRACTICAL 5: Study Audio, Video And Location on Android.

A. Write a program to:
   a. Record an audio and play it.
   b. Play a video in a video view

- **AndroidManifest.xml:**

```
<?xml version="1.0" encoding="utf-8"?>
<manifest xmlns:android="http://schemas.android.com/apk/res/android"
   package="com.example.practical6_a">
   <uses-permission android:name="android.permission.INTERNET" />
   <uses-permission android:name="android.permission.RECORD_AUDIO" />
   <uses-permission android:name="android.permission.WRITE_EXTERNAL_STORAGE"/>
   <uses-permission android:name="android.permission.STORAGE" />
   <application
      android:allowBackup="true"
      android:icon="@mipmap/ic_launcher"
      android:label="@string/app_name"
      android:roundIcon="@mipmap/ic_launcher_round"
      android:supportsRtl="true"
      android:theme="@style/Theme.Practical6_A">
      <activity
         android:name=".MainActivity"
         android:exported="true">
         <intent-filter>
            <action android:name="android.intent.action.MAIN" />
            <category android:name="android.intent.category.LAUNCHER" />
         </intent-filter>
      </activity>
   </application>
</manifest>
```

- **activity_main.xml:**

```
<?xml version="1.0" encoding="utf-8"?>
<androidx.constraintlayout.widget.ConstraintLayout
xmlns:android="http://schemas.android.com/apk/res/android"
   xmlns:app="http://schemas.android.com/apk/res-auto"
   xmlns:tools="http://schemas.android.com/tools"
   android:layout_width="match_parent"
android:layout_height="match_parent"
   tools:context=".MainActivity">
   <Button
      android:id="@+id/button5"
```

```
        android:layout_width="96dp"
        android:layout_height="55dp"
        android:text="PLAY"
        app:layout_constraintBottom_toBottomOf="parent"
        app:layout_constraintEnd_toEndOf="parent"
        app:layout_constraintHorizontal_bias="0.295"
        app:layout_constraintStart_toStartOf="parent"
        app:layout_constraintTop_toTopOf="parent"
        app:layout_constraintVertical_bias="0.318" />
    <Button
        android:id="@+id/button4"
        android:layout_width="96dp"
        android:layout_height="55dp"
        android:text="STOP PLAY"
        app:layout_constraintBottom_toBottomOf="parent"
        app:layout_constraintEnd_toEndOf="parent"
        app:layout_constraintHorizontal_bias="0.717"
        app:layout_constraintStart_toStartOf="parent"
        app:layout_constraintTop_toTopOf="parent"
        app:layout_constraintVertical_bias="0.318" />
    <Button
        android:id="@+id/button3"
        android:layout_width="96dp"
        android:layout_height="55dp"
        android:text="PLAY VIDEO"
        app:layout_constraintBottom_toBottomOf="parent"
        app:layout_constraintEnd_toEndOf="parent"
        app:layout_constraintHorizontal_bias="0.498"
        app:layout_constraintStart_toStartOf="parent"
        app:layout_constraintTop_toTopOf="parent"
        app:layout_constraintVertical_bias="0.458" />
    <Button
        android:id="@+id/button2"
        android:layout_width="96dp"
        android:layout_height="55dp"
        android:text="STOP RECORD"
app:layout_constraintBottom_toBottomOf="parent"
        app:layout_constraintEnd_toEndOf="parent"
        app:layout_constraintHorizontal_bias="0.717"
        app:layout_constraintStart_toStartOf="parent"
        app:layout_constraintTop_toTopOf="parent"
        app:layout_constraintVertical_bias="0.176" />
    <TextView
        android:id="@+id/textView"
        android:layout_width="294dp"
        android:layout_height="37dp"
```

```
        android:text="AUDIO AND VIDEO"
        android:textAlignment="center"
        android:textSize="30sp"
        android:textStyle="bold"
        app:layout_constraintBottom_toBottomOf="parent"
        app:layout_constraintEnd_toEndOf="parent"
        app:layout_constraintHorizontal_bias="0.495"
        app:layout_constraintStart_toStartOf="parent"
        app:layout_constraintTop_toTopOf="parent"
        app:layout_constraintVertical_bias="0.023" />
    <VideoView
        android:id="@+id/videoView"
        android:layout_width="365dp"
        android:layout_height="293dp"
        app:layout_constraintBottom_toBottomOf="parent"
        app:layout_constraintEnd_toEndOf="parent"
        app:layout_constraintHorizontal_bias="0.549"
        app:layout_constraintStart_toStartOf="parent"
        app:layout_constraintTop_toTopOf="parent"
        app:layout_constraintVertical_bias="0.904" />
    <Button
        android:id="@+id/button"
        android:layout_width="96dp"
        android:layout_height="55dp"
        android:text="RECORD"
        app:layout_constraintBottom_toBottomOf="parent"
        app:layout_constraintEnd_toEndOf="parent"
        app:layout_constraintHorizontal_bias="0.295"
        app:layout_constraintStart_toStartOf="parent"
        app:layout_constraintTop_toTopOf="parent"
        app:layout_constraintVertical_bias="0.176" />
</androidx.constraintlayout.widget.ConstraintLayout>
```

- **MainActivity.java:**

```
package com.example.practical6_a;
import static android.Manifest.permission.RECORD_AUDIO;
import static android.Manifest.permission.WRITE_EXTERNAL_STORAGE;
import androidx.appcompat.app.AppCompatActivity;
import androidx.core.app.ActivityCompat;
import androidx.core.content.ContextCompat;
import android.annotation.SuppressLint;
import android.content.pm.PackageManager;
import android.media.MediaPlayer;
import android.media.MediaRecorder;
import android.net.Uri;
import android.os.Bundle;
```

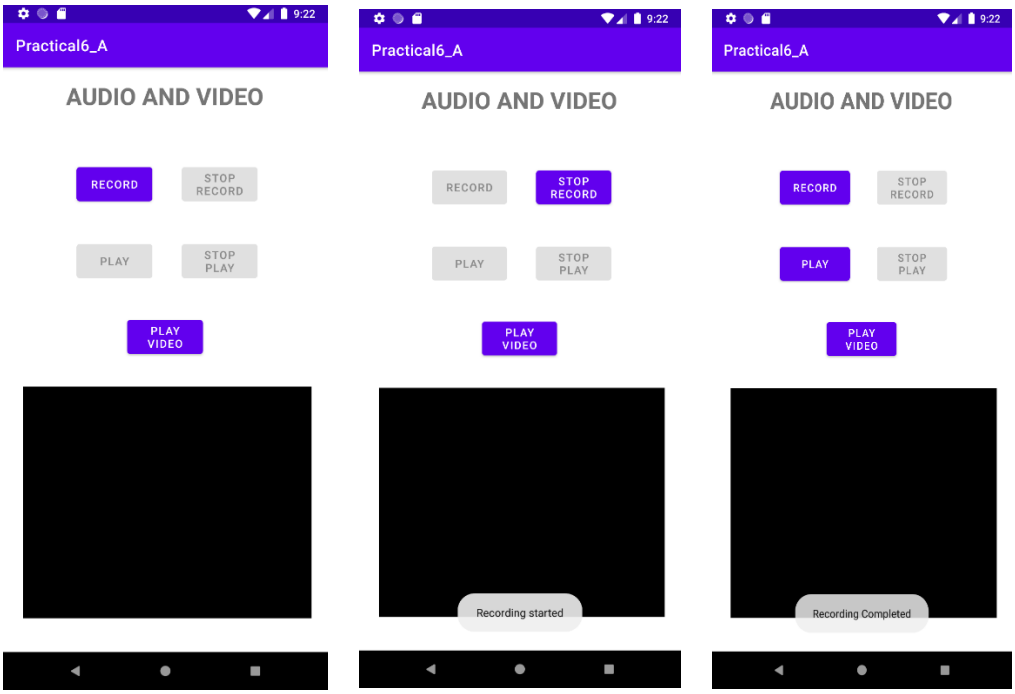
```
import android.os.Environment;
import android.util.Log;
import android.view.View;
import android.widget.Button;
import android.widget.MediaController;
import android.widget.Toast;
import android.widget.VideoView;
import java.io.IOException;
import java.util.Random;
public class MainActivity extends AppCompatActivity {
   Button buttonStart, buttonStop, buttonPlayLastRecordAudio,
playvideo,buttonStopPlayingRecording ;
   String AudioSavePathInDevice = null;
   MediaRecorder mediaRecorder ;
   Random random ;
   VideoView vw;
   String RandomAudioFileName = "ABCDEFGHIJKLMNOP";
   public static final int RequestPermissionCode = 1;
   MediaPlayer mediaPlayer ;
   @Override
   protected void onCreate(Bundle savedInstanceState) {
      super.onCreate(savedInstanceState);
      setContentView(R.layout.activity_main);
      vw = findViewById(R.id.videoView);
      buttonStart = (Button) findViewById(R.id.button);
      buttonStop = (Button) findViewById(R.id.button2);
      buttonPlayLastRecordAudio = (Button) findViewById(R.id.button5);
      buttonStopPlayingRecording = (Button)findViewById(R.id.button4);
      playvideo=findViewById(R.id.button3);
      buttonStop.setEnabled(false);
      buttonPlayLastRecordAudio.setEnabled(false);
      buttonStopPlayingRecording.setEnabled(false);
      random = new Random();
      playvideo.setOnClickListener(new View.OnClickListener() {
         @Override
         public void onClick(View view) {
            Toast.makeText(MainActivity.this, "STARTING VIDEO",
Toast.LENGTH_LONG).show();
            Uri vidUri =
Uri.parse("https://ia800201.us.archive.org/22/items/ksnn_compilation_master_the_internet/ksnn
_compilation_master_the_internet_512kb.mp4");
            vw = findViewById(R.id.videoView);
            vw.setVideoURI(vidUri);
            vw.setMediaController(new MediaController(MainActivity.this));
            vw.requestFocus();
            vw.start();
```

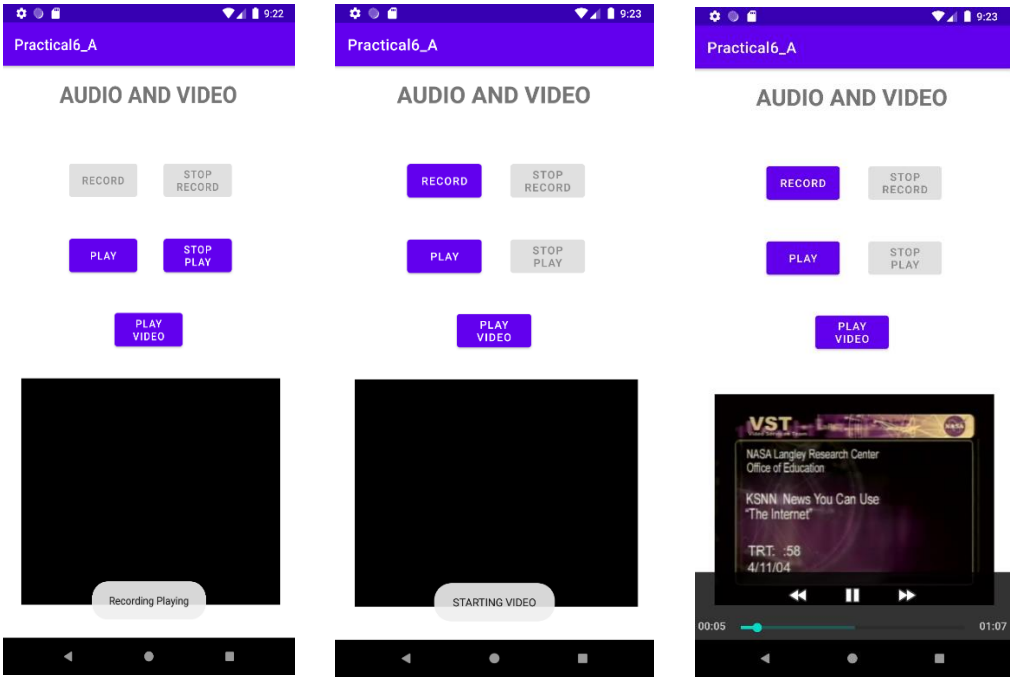
```
        }
    });
    buttonStart.setOnClickListener(new View.OnClickListener() {
        @Override
        public void onClick(View view) {
            if(checkPermission()) {
                AudioSavePathInDevice =
                        Environment.getExternalStorageDirectory().getAbsolutePath() + "/" +
                                CreateRandomAudioFileName(5) + "AudioRecording.3gp";
                MediaRecorderReady();
                try {
                    mediaRecorder.prepare();
                    mediaRecorder.start();
                    Toast.makeText(MainActivity.this, "Recording started",
Toast.LENGTH_LONG).show();
                } catch (IllegalStateException e) {
                    // TODO Auto-generated catch block
                    e.printStackTrace();
                } catch (IOException e) {
                    // TODO Auto-generated catch block
                    e.printStackTrace();
                }
                buttonStart.setEnabled(false);
                buttonStop.setEnabled(true);
            } else {
                requestPermission();
            }
        }
    });
    buttonStop.setOnClickListener(new View.OnClickListener() {
        @Override
        public void onClick(View view) {
            mediaRecorder.stop();
            buttonStop.setEnabled(false);
            buttonPlayLastRecordAudio.setEnabled(true);
            buttonStart.setEnabled(true);
            buttonStopPlayingRecording.setEnabled(false);
            Toast.makeText(MainActivity.this, "Recording Completed",
                    Toast.LENGTH_LONG).show();
        }
    });
    buttonPlayLastRecordAudio.setOnClickListener(new View.OnClickListener() {
        @Override
        public void onClick(View view) throws IllegalArgumentException,
                SecurityException, IllegalStateException {
            buttonStop.setEnabled(false);
```

```
            buttonStart.setEnabled(false);
            buttonStopPlayingRecording.setEnabled(true);
            mediaPlayer = new MediaPlayer();
            try {
               mediaPlayer.setDataSource(AudioSavePathInDevice);
               mediaPlayer.prepare();
            } catch (IOException e) {
               e.printStackTrace();
            }
            mediaPlayer.start();
            Toast.makeText(MainActivity.this, "Recording Playing",
                  Toast.LENGTH_LONG).show();
         }
      });
      buttonStopPlayingRecording.setOnClickListener(new View.OnClickListener() {
         @Override
         public void onClick(View view) {
            buttonStop.setEnabled(false);
            buttonStart.setEnabled(true);
            buttonStopPlayingRecording.setEnabled(false);
            buttonPlayLastRecordAudio.setEnabled(true);

            if(mediaPlayer != null){
               mediaPlayer.stop();
               mediaPlayer.release();
               MediaRecorderReady();
            }
         }
      });
   }
   public void MediaRecorderReady(){
      mediaRecorder=new MediaRecorder();
      mediaRecorder.setAudioSource(MediaRecorder.AudioSource.MIC);
      mediaRecorder.setOutputFormat(MediaRecorder.OutputFormat.THREE_GPP);
      mediaRecorder.setAudioEncoder(MediaRecorder.AudioEncoder.AMR_NB);
      mediaRecorder.setOutputFile(AudioSavePathInDevice);
   }
   public String CreateRandomAudioFileName(int string){
      StringBuilder stringBuilder = new StringBuilder( string );
      int i = 0 ;
      while(i < string ) {
         stringBuilder.append(RandomAudioFileName.
               charAt(random.nextInt(RandomAudioFileName.length())));
         i++ ;
      }
      return stringBuilder.toString();
```

```
    }
    private void requestPermission() {
        ActivityCompat.requestPermissions(MainActivity.this, new
                String[]{WRITE_EXTERNAL_STORAGE, RECORD_AUDIO},
RequestPermissionCode);
    }
    public void onRequestPermissionsResult(int requestCode, String permissions[], int[]
grantResults) {
        super.onRequestPermissionsResult(requestCode, permissions, grantResults);
        switch (requestCode) {
            case RequestPermissionCode:
                if (grantResults.length > 0) {
                    boolean StoragePermission = grantResults[0] ==
                            PackageManager.PERMISSION_GRANTED;
                    boolean RecordPermission = grantResults[1] ==
                            PackageManager.PERMISSION_GRANTED;

                    if (StoragePermission && RecordPermission) {
                        Toast.makeText(MainActivity.this, "Permission Granted",
                                Toast.LENGTH_LONG).show();
                    } else {
                        Toast.makeText(MainActivity.this, "Permission Denied",
Toast.LENGTH_LONG).show();
                    }
                }
                break;
        }
    }
    public boolean checkPermission() {
        int result = ContextCompat.checkSelfPermission(getApplicationContext(),
                WRITE_EXTERNAL_STORAGE);
        int result1 = ContextCompat.checkSelfPermission(getApplicationContext(),
                RECORD_AUDIO);
        return result == PackageManager.PERMISSION_GRANTED &&
                result1 == PackageManager.PERMISSION_GRANTED;
    }
}
```

- **Output:**


Practical6_A
AUDIO AND VIDEO
RECORD
STOP RECORD
PLAY
STOP PLAY
PLAY VIDEO
Recording started
Recording Completed


Practical6_A
AUDIO AND VIDEO
RECORD
STOP RECORD
PLAY
STOP PLAY
PLAY VIDEO
Recording Playing
STARTING VIDEO
NASA Langley Research Center
Office of Education
KSNN News You Can Use
"The Internet"
TRT: :58
4/11/04
00:05
01:07

B. Create an application to display the current location of your device. (Latitude & Longitude value).

- **AndroidManifest.xml:**

```
<?xml version="1.0" encoding="utf-8"?>
<manifest xmlns:android="http://schemas.android.com/apk/res/android"
   package="com.example.practical6b">
   <uses-permission android:name="android.permission.ACCESS_FINE_LOCATION"/>
   <uses-permission android:name="android.permission.ACCESS_COARSE_LOCATION" />
   <application
      android:allowBackup="true"
      android:icon="@mipmap/ic_launcher"
      android:label="@string/app_name"
      android:roundIcon="@mipmap/ic_launcher_round"
      android:supportsRtl="true"
      android:theme="@style/Theme.Practical6B">
      <activity
         android:name=".MainActivity"
         android:exported="true">
         <intent-filter>
            <action android:name="android.intent.action.MAIN" />
            <category android:name="android.intent.category.LAUNCHER" />
         </intent-filter>
      </activity>
   </application>
</manifest>
```

- **activity_main.xml:**

```
<?xml version="1.0" encoding="utf-8"?>
<LinearLayout xmlns:android="http://schemas.android.com/apk/res/android"
   xmlns:tools="http://schemas.android.com/tools"
   android:layout_width="match_parent"
   android:layout_height="match_parent"
   android:orientation="vertical"
   android:gravity="center"
   tools:context=".MainActivity">
   <TextView
      android:id="@+id/showLocation"
      android:layout_width="wrap_content"
      android:layout_height="wrap_content"
      android:hint="Location"
      android:textSize="24sp" />
   <Button
      android:id="@+id/btnGetLocation"
```

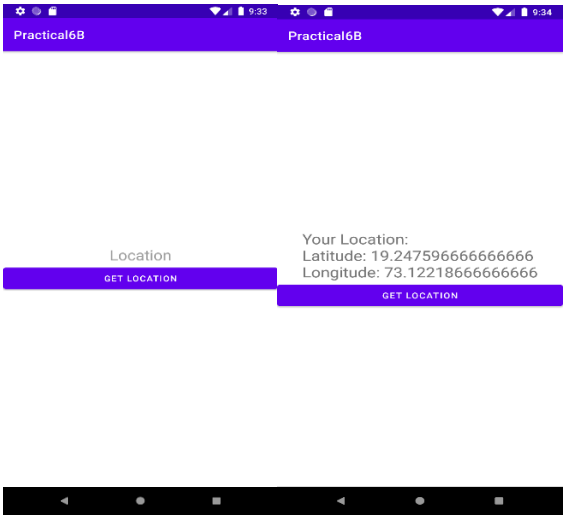
```
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:text="Get Location" />
</LinearLayout>
```

- **MainActivity.java:**

```
package com.example.practical6b;
import android.Manifest;
import android.app.AlertDialog;
import android.content.Context;
import android.content.DialogInterface;
import android.content.Intent;
import android.content.pm.PackageManager;
import android.location.Location;
import android.location.LocationManager;
import android.provider.Settings;
import android.support.v4.app.*;
import android.support.*;
import android.os.Bundle;
import android.view.View;
import android.widget.Button;
import android.widget.TextView;
import android.widget.Toast;
import androidx.appcompat.app.AppCompatActivity;
import androidx.core.app.ActivityCompat;
public class MainActivity extends AppCompatActivity {
   private static final int REQUEST_LOCATION = 1;
   Button btnGetLocation;
   TextView showLocation;
   LocationManager locationManager;
String latitude, longitude;
   @Override
   protected void onCreate(Bundle savedInstanceState) {
      super.onCreate(savedInstanceState);
      setContentView(R.layout.activity_main);
      ActivityCompat.requestPermissions( this,
            new String[] {Manifest.permission.ACCESS_FINE_LOCATION},
REQUEST_LOCATION);
      showLocation = findViewById(R.id.showLocation);
      btnGetLocation = findViewById(R.id.btnGetLocation);
      btnGetLocation.setOnClickListener(new View.OnClickListener() {
         @Override
         public void onClick(View v) {
            locationManager = (LocationManager)
getSystemService(Context.LOCATION_SERVICE);
            if (!locationManager.isProviderEnabled(LocationManager.GPS_PROVIDER)) {
```

```
                OnGPS();
            } else {
                getLocation();
            }
        }
    });
}
private void OnGPS() {
    final AlertDialog.Builder builder = new AlertDialog.Builder(this);
    builder.setMessage("Enable GPS").setCancelable(false).setPositiveButton("Yes", new
DialogInterface.OnClickListener() {
        @Override
        public void onClick(DialogInterface dialog, int which) {
            startActivity(new Intent(Settings.ACTION_LOCATION_SOURCE_SETTINGS));
        }
    }).setNegativeButton("No", new DialogInterface.OnClickListener() {
        @Override
        public void onClick(DialogInterface dialog, int which) {
            dialog.cancel();
        }
    });
    final AlertDialog alertDialog = builder.create();
    alertDialog.show();
}
private void getLocation() {
    if (ActivityCompat.checkSelfPermission(
            MainActivity.this,Manifest.permission.ACCESS_FINE_LOCATION) !=
PackageManager.PERMISSION_GRANTED && ActivityCompat.checkSelfPermission(
            MainActivity.this, Manifest.permission.ACCESS_COARSE_LOCATION) !=
PackageManager.PERMISSION_GRANTED) {
        ActivityCompat.requestPermissions(this, new
String[]{Manifest.permission.ACCESS_FINE_LOCATION}, REQUEST_LOCATION);
    } else {
        Location locationGPS =
locationManager.getLastKnownLocation(LocationManager.GPS_PROVIDER);
        if (locationGPS != null) {
            double lat = locationGPS.getLatitude();
            double longi = locationGPS.getLongitude();
            latitude = String.valueOf(lat);
            longitude = String.valueOf(longi);
            showLocation.setText("Your Location: " + "\n" + "Latitude: " + latitude + "\n" +
"Longitude: " + longitude);
        } else {
            Toast.makeText(this, "Unable to find location.", Toast.LENGTH_SHORT).show();
        }}}}
```

- **Output:**


Practical6B
Location
GET LOCATION
Practical6B
Your Location:
Latitude: 19.247596666666666
Longitude: 73.12218666666666
GET LOCATION

PRACTICAL 6: Android Programs Based On REST API

A. Create a basic application that allows you to download HTML from a given web page using HttpURLConnection.

- **activity_main.xml:**

```
<?xml version="1.0" encoding="utf-8"?>
<androidx.constraintlayout.widget.ConstraintLayout
xmlns:android="http://schemas.android.com/apk/res/android"
   xmlns:app="http://schemas.android.com/apk/res-auto"
   xmlns:tools="http://schemas.android.com/tools"
   android:layout_width="match_parent"
   android:layout_height="match_parent"
   tools:context=".MainActivity">
   <Button
      android:id="@+id/button"
      android:layout_width="wrap_content"
      android:layout_height="wrap_content"
      android:text="SUBMIT"
      android:onClick="onClick"
      app:layout_constraintBottom_toBottomOf="parent"
      app:layout_constraintEnd_toEndOf="parent"
      app:layout_constraintHorizontal_bias="0.498"
      app:layout_constraintStart_toStartOf="parent"
      app:layout_constraintTop_toTopOf="parent"
      app:layout_constraintVertical_bias="0.174" />
   <EditText
      android:id="@+id/get_url"
      android:layout_width="358dp"
      android:layout_height="41dp"
      android:ems="10"
      android:hint="Enter URL"
      android:inputType="textPersonName"
      android:textAlignment="center"
      android:textStyle="bold"
      app:layout_constraintBottom_toBottomOf="parent"
      app:layout_constraintEnd_toEndOf="parent"
      app:layout_constraintHorizontal_bias="0.497"
      app:layout_constraintStart_toStartOf="parent"
```

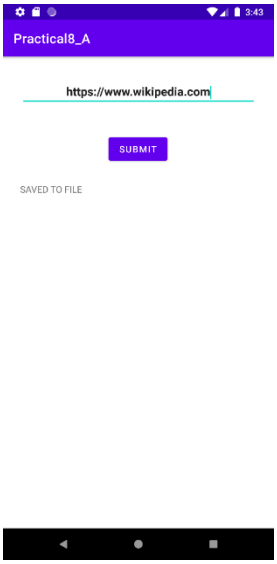
```
      app:layout_constraintTop_toTopOf="parent"
      app:layout_constraintVertical_bias="0.051" />
   <TextView
      android:id="@+id/textView"
      android:layout_width="359dp"
      android:layout_height="482dp"
      android:text=""
      android:scrollbars = "vertical"
      app:layout_constraintBottom_toBottomOf="parent"
      app:layout_constraintEnd_toEndOf="parent"
      app:layout_constraintHorizontal_bias="0.497"
      app:layout_constraintStart_toStartOf="parent"
      app:layout_constraintTop_toTopOf="parent"
      app:layout_constraintVertical_bias="0.816" />
</androidx.constraintlayout.widget.ConstraintLayout>
```

- **MainActivity.java:**

```
   @Override
   protected void onCreate(Bundle savedInstanceState) {
      super.onCreate(savedInstanceState);
      setContentView(R.layout.activity_main);
      e1=findViewById(R.id.get_url);
      b=findViewById(R.id.button);
      t1 = findViewById(R.id.textView);
   }
   public void onClick(View view) throws IOException {
      Thread thread = new Thread(new Runnable() {
         @Override
         public void run() {
            try  {
               String link = e1.getText().toString();
               //String link ="http://www.google.com";
               URL url = new URL(link);
               HttpURLConnection conn = (HttpURLConnection) url.openConnection();
               conn.connect();
               InputStream is = conn.getInputStream();
               BufferedReader reader = new BufferedReader(new InputStreamReader(is, "UTF-
8"));
               String webPage = "",data="";
               while ((data = reader.readLine()) != null){
                  webPage += data + "\n";
               }
               String filename="download.txt";
               FileOutputStream fos;
```

```
                try {
                    fos = openFileOutput(filename, Context.MODE_APPEND);
                    //default mode is PRIVATE, can be APPEND etc.
                    fos.write(webPage.getBytes());
                    fos.close();
                    t1.setText("SAVED TO FILE");
                } catch (FileNotFoundException e)
{Toast.makeText(getApplicationContext(),"FILE NOT FOUND",
                        Toast.LENGTH_LONG).show();;}
                catch (IOException e) {e.printStackTrace();}
            } catch (Exception e)
            {
                e.printStackTrace();
            } }});
        thread.start();
    }}
```

- **Output:**



B. Create an application to parse the data using JSONObject methods and set it in the Text View's. (Employee name and salary stored in JSON format).

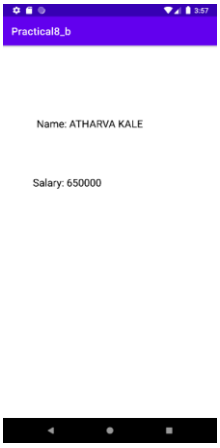
- **activity_main.xml:**

```
<?xml version="1.0" encoding="utf-8"?>
<androidx.constraintlayout.widget.ConstraintLayout
xmlns:android="http://schemas.android.com/apk/res/android"
   xmlns:app="http://schemas.android.com/apk/res-auto"
   xmlns:tools="http://schemas.android.com/tools"
   android:layout_width="match_parent"
   android:layout_height="match_parent"
   tools:context=".MainActivity">
   <TextView
      android:id="@+id/name"
      android:layout_width="248dp"
      android:layout_height="59dp"
      android:layout_centerHorizontal="true"
      android:text="Name"
      android:textColor="#000"
      android:textSize="20sp"
      app:layout_constraintBottom_toBottomOf="parent"
      app:layout_constraintEnd_toEndOf="parent"
      app:layout_constraintHorizontal_bias="0.395"
      app:layout_constraintStart_toStartOf="parent"
      app:layout_constraintTop_toTopOf="parent"
      app:layout_constraintVertical_bias="0.208" />
   <TextView
      android:id="@+id/salary"
      android:layout_width="259dp"
      android:layout_height="61dp"
      android:layout_centerHorizontal="true"
      android:text="Salary"
      android:textColor="#000"
      android:textSize="20sp"
      app:layout_constraintBottom_toBottomOf="parent"
      app:layout_constraintEnd_toEndOf="parent"
      app:layout_constraintHorizontal_bias="0.377"
      app:layout_constraintStart_toStartOf="parent"
      app:layout_constraintTop_toTopOf="parent"
      app:layout_constraintVertical_bias="0.381" />
</androidx.constraintlayout.widget.ConstraintLayout>
```

- **MainActivity.java:**

```
package com.example.practical8_b;
import androidx.appcompat.app.AppCompatActivity;
import android.os.Bundle;
import android.widget.TextView;
import org.json.JSONException;
import org.json.JSONObject;
public class MainActivity extends AppCompatActivity {
   String JSON_STRING = "{\"employee\":{\"name\":\"ATHARVA
KALE\",\"salary\":650000}}";
   String name, salary;
   TextView employeeName, employeeSalary;
   @Override
   protected void onCreate(Bundle savedInstanceState) {
      super.onCreate(savedInstanceState);
      setContentView(R.layout.activity_main);
      employeeName = (TextView) findViewById(R.id.name);
      employeeSalary = (TextView) findViewById(R.id.salary);
      try {
         JSONObject obj = new JSONObject(JSON_STRING);
         JSONObject employee = obj.getJSONObject("employee");
         name = employee.getString("name");
         salary = employee.getString("salary");
        employeeName.setText("Name: "+name);
         employeeSalary.setText("Salary: "+salary);
      } catch (JSONException e) {
         e.printStackTrace();
      }
   }
}
```

- **Output:**



C. Write a basic application to (use volley library), create a button and on click of the button a HTTP request will be send to server. The response from the server is then displayed using Toast on the screen

- **activity_main.xml:**

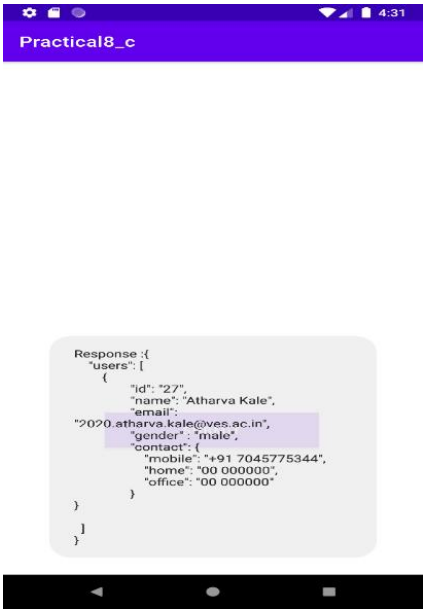
```
<?xml version="1.0" encoding="utf-8"?>
<androidx.constraintlayout.widget.ConstraintLayout
xmlns:android="http://schemas.android.com/apk/res/android"
   xmlns:app="http://schemas.android.com/apk/res-auto"
   xmlns:tools="http://schemas.android.com/tools"
   android:layout_width="match_parent"
   android:layout_height="match_parent"
   tools:context=".MainActivity">
   <Button
      android:id="@+id/buttonRequest"
      android:layout_width="210dp"
      android:layout_height="49dp"
      android:layout_alignParentTop="true"
      android:background="#414af4"
      android:text="SUBMIT"
      android:textColor="#ffffff"
      app:layout_constraintBottom_toBottomOf="parent"
      app:layout_constraintEnd_toEndOf="parent"
      app:layout_constraintHorizontal_bias="0.494"
      app:layout_constraintStart_toStartOf="parent"
      app:layout_constraintTop_toTopOf="parent"
      app:layout_constraintVertical_bias="0.735" />
</androidx.constraintlayout.widget.ConstraintLayout>
```

- **MainActivity.java:**

```
package com.example.practical8_c;
import androidx.appcompat.app.AppCompatActivity;
import android.os.Bundle;
import android.util.Log;
import android.view.View;
import android.widget.Button;
import android.widget.Toast;
import com.android.volley.Request;
import com.android.volley.RequestQueue;
import com.android.volley.Response;
import com.android.volley.VolleyError;
import com.android.volley.toolbox.StringRequest;
import com.android.volley.toolbox.Volley;
public class MainActivity extends AppCompatActivity {
```

```
   private static final String TAG = MainActivity.class.getName();
   private Button btnRequest;
   private RequestQueue mRequestQueue;
   private StringRequest mStringRequest;
   private String url = "https://run.mocky.io/v3/b0f7696f-ac67-4152-9f23-1a4c3dc68b37";
   @Override
   protected void onCreate(Bundle savedInstanceState) {
      super.onCreate(savedInstanceState);
      setContentView(R.layout.activity_main);
      btnRequest = (Button) findViewById(R.id.buttonRequest);
      btnRequest.setOnClickListener(new View.OnClickListener() {
                                 @Override
                                 public void onClick(View v){
                                    sendAndRequestResponse();
                                 }
                              }
      );
   }
   private void sendAndRequestResponse() {
      //RequestQueue initialized
      mRequestQueue = Volley.newRequestQueue(this);
//String Request initialized
      mStringRequest = new StringRequest(Request.Method.GET, url, new
Response.Listener<String>() {
         @Override
         public void onResponse(String response) {
            Toast.makeText(getApplicationContext(),"Response :" + response.toString(),
Toast.LENGTH_LONG).show();//display the response on screen
         }
      }, new Response.ErrorListener() {
         @Override
         public void onErrorResponse(VolleyError error) {
            Log.i(TAG,"Error :" + error.toString());
         }
      });
      mRequestQueue.add(mStringRequest);
   }
}
```

- **Output:**


4:31
Practical8_c
Response :{
"users": [
{
"id": "27",
"name": "Atharva Kale",
"email":
"2020.atharva.kale@ves.ac.in",
"gender" : "male",
"contact": {
"mobile": "+91 7045775344",
"home": "00 000000",
"office": "00 000000"
}
}
]
}

PRACTICAL 7: Introduction to Dart and Flutter

A. Create a widget ProductBox that contains the details of the product, such as image, name, price, and description. In the ProductBox widget, we use the following child widgets: Container, Row, Column, Expanded, Card, Text, Image, etc. This widget contains the following layout

- **main.dart:**

```
import 'package:flutter/material.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 Demo', theme: ThemeData(
      primarySwatch: Colors.blue,
    ),
      home: MyHomePage(title: 'Product layout demo home page'),
    );
  }
}
class MyHomePage extends StatelessWidget {
  MyHomePage({Key? key, required this.title}) : super(key: key);
  final String title;
  @override
  Widget build(BuildContext context) {
    return Scaffold(
        appBar: AppBar(title: Text("Product Listing")),
        body: ListView(
          shrinkWrap: true,
          padding: const EdgeInsets.fromLTRB(2.0, 10.0, 2.0, 10.0),
          children: <Widget>[
            ProductBox(
name: "iPhone",
              description: "If you have a kidney you have a iPhone",
              price: 99999,
```

```
            image: "iPhone.png"
          ),
          ProductBox(
            name: "Pixel",
            description: "Pixel, garbage thing!!",
            price: 50000,
            image: "Pixel.png"
          ),
          ProductBox(
            name: "Laptop",
            description: "Alienware feel the POWER OF ALIEN!!!",
            price: 200000,
            image: "Laptop.png"
          ),
          ProductBox(
            name: "Tablet",
            description: "Ipad the one to eat your money!!!!",
            price: 60000,
            image: "Tablet.png"
          ),
          ProductBox(
            name: "Pendrive",
            description: "Pendrive FAST SMOOTH AND IT SAVES YOUR DATA!!!",
            price: 200,
            image: "Pendrive.png"
          ),
          ProductBox(
            name: "Floppy Drive",
            description: "Floppy drive???!!! Go visit a DOCTOR!!!",
            price: 0,
            image: "Floppy.png"
          ),
        ],
      )
    );
  }
}
class ProductBox extends StatelessWidget {
  ProductBox({Key? key, required this.name, required this.description, required this.price,
required this.image}) :
```

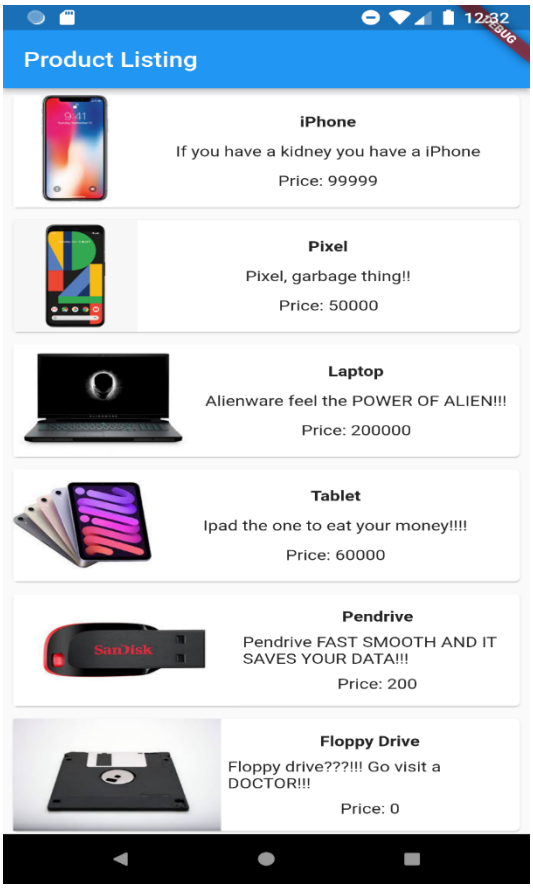
```
      super(key: key);
  final String name;
  final String description;
  final int price;
  final String image;

  Widget build(BuildContext context) {
   return Container(
      padding: EdgeInsets.all(2),
      height: 120,
      child: Card(
         child: Row(
            mainAxisAlignment: MainAxisAlignment.spaceEvenly,
            children: <Widget>[
              Image.asset("assets/" + image),
              Expanded(
                 child: Container(
                    padding: EdgeInsets.all(5),
                    child: Column(
                      mainAxisAlignment: MainAxisAlignment.spaceEvenly,
                      children: <Widget>[
                        Text(
                           this.name, style: TextStyle(
                           fontWeight: FontWeight.bold
                        )
                        ),
                        Text(this.description), Text(
                           "Price: " + this.price.toString()
                        ),
                      ],
                    )
                 )
              )
            ]) )
    );  }}
```

- **pubspec.yaml:**

```
assets:
  - assets/iPhone.png
  - assets/Laptop.png
```

- assets/Pendrive.png
- assets/Pixel.png
- assets/Tablet.png
- assets/Floppy.png

- **Output:**



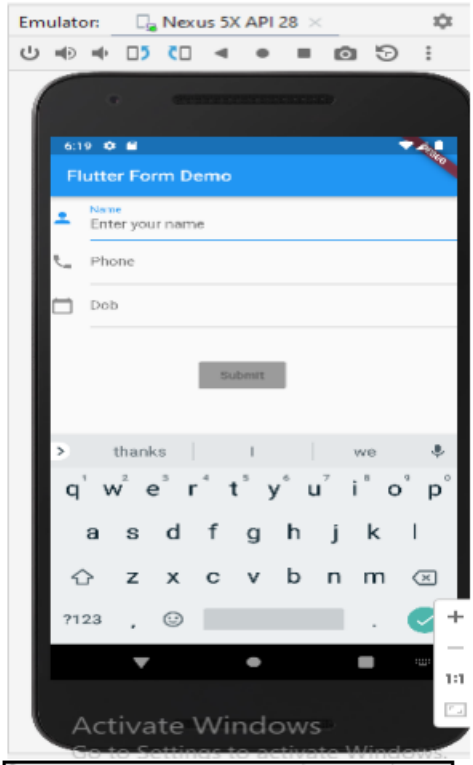
PRACTICAL 8: Introduction to Dart and Flutter:

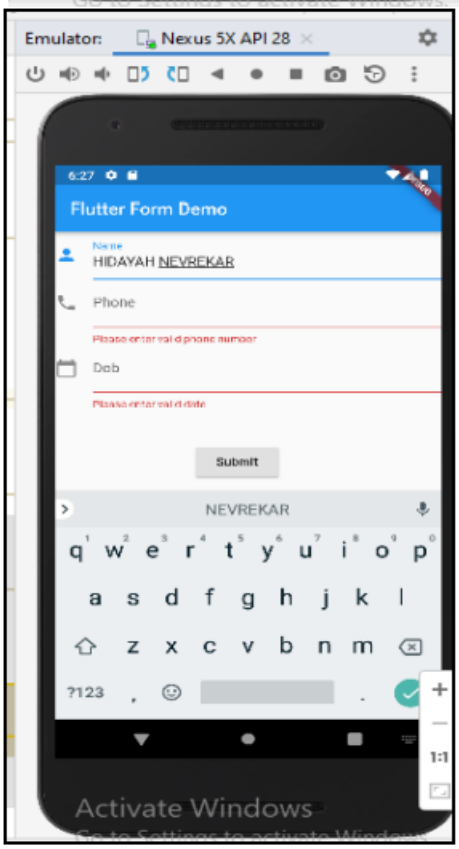
A. Create a registration form using Flutter.

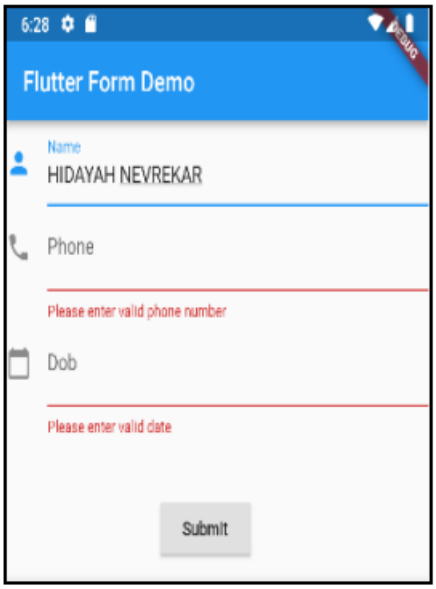
```
class MyCustomFormState extends State<MyCustomForm> {

  final _formKey = GlobalKey<FormState>();

  @override
  Widget build(BuildContext context) {
    // Build a Form widget using the _formKey created above.
    return Form(
      key: _formKey,
      child: Column(
        crossAxisAlignment: CrossAxisAlignment.start,
        children: <Widget>[
          TextFormField(
            decoration: const InputDecoration(
              icon: const Icon(Icons.person),
              hintText: 'Enter your name',
              labelText: 'Name',
            ),  // InputDecoration
          ),  // TextFormField
          TextFormField(
            decoration: const InputDecoration(
              icon: const Icon(Icons.phone),
              hintText: 'Enter a phone number',
              labelText: 'Phone',
            ),  // InputDecoration
          ),  // TextFormField
          TextFormField(
            decoration: const InputDecoration(
              icon: const Icon(Icons.calendar_today),
              hintText: 'Enter your date of birth',
              labelText: 'Dob',
            ),  // InputDecoration
          ),  // TextFormField
          new Container(
              padding: const EdgeInsets.only(left: 150.0, top: 40.0),
              child: new RaisedButton(
                child: const Text('Submit'),
                onPressed: null,
              )),  // RaisedButton, Container
```

```
  ),  // InputDecoration
  validator: (value) {
    if (value!.isEmpty) {
      return 'Please enter valid phone number';
    }
    return null;
  },
),  // TextFormField
TextFormField(
  decoration: const InputDecoration(
    icon: const Icon(Icons.calendar_today),
    hintText: 'Enter your date of birth',
    labelText: 'Dob',
  ),  // InputDecoration
  validator: (value) {
    if (value!.isEmpty) {
      return 'Please enter valid date';
    }
    return null;
  },
),  // TextFormField
Container(
    padding: const EdgeInsets.only(left: 150.0, top: 40.0),
    child: new RaisedButton(
      child: const Text('Submit'),
      onPressed: () {
        // It returns true if the form is valid, otherwise returns false
        if (_formKey.currentState!.validate()) {
          // If the form is valid, display a Snackbar.
          Scaffold.of(context)
              .showSnackBar(SnackBar(content: Text('Data is in processing.')));
        }
      },
    )),  // RaisedButton, Container
```

- **Output:**


Emulator: Nexus 5X API 28
Flutter Form Demo
Name
Enter your name
Phone
Dob
Submit
Activate Windows


Emulator: Nexus 5X API 28
Flutter Form Demo
Name
HIDAYAH NEVREKAR
Phone
Dob
Submit
Activate Windows


Flutter Form Demo
Name
HIDAYAH NEVREKAR
Phone
Please enter valid phone number
Dob
Please enter valid date
Submit