# Skill Based Lab and Mobile Computing Lab

# Subject Code: MCAL34

A Practical Journal Submitted in Fulfilment

of the Degree of

**MASTER**

**In**

**COMPUTER APPLICATION**

**Year 2023-2024**

By

Ravishankar Jaiswal

**129211**

Semester- III

Under the Guidance of

**Prof. Dynaneshwar Deore**


Institute of Distance and Open Learning

Vidya Nagari, Kalina, Santacruz East – 400098.

University of Mumbai

**PCP Center**

[Satish Pradhan Dyanasadhana College, Thane]

# Institute of Distance and Open Learning,

**Vidyanagari, Kalina, Santacruz (E) -400098**

# CERTIFICATE

This to certify that, **Mr. Ravishankar Jaiswal** appearing **Master in Computer Application (Semester II) Application ID: 129211** has satisfactorily completed the prescribed practical of **MCAL34- Skill Based Lab and Mobile Computing Lab** as laid down by the University of Mumbai for the academic year 2023-24

| Teacher in charge | Examiners | Coordinator |
|---|---|---|
| Prof. Dyaneshwar Deore | | IDOL, MCA<br>University of Mumbai |

Date: -

Place: -

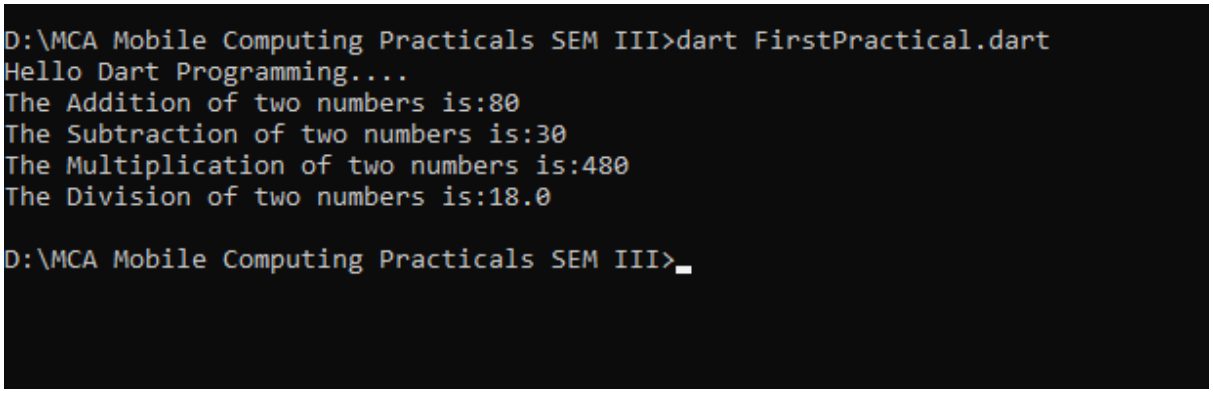
# INDEX:

**Practical 1: Write a program to demonstrate dart function and perform addition, multiplication, division and subtraction.**

**Code:**

```
void add(int a, int b){
        var c = a+b;
        print("The Additio0n of two numbers is:$c");
}
void sub(int a, int b){
        var c = a-b;
        print("The Subtraction of two numbers is:$c");
}
void mul(int a, int b){
        var c = a*b;
        print("The Multiplication of two numbers is:$c");
}
void div(int a, int b){
        var c = a/b;
        print("The Division of two numbers is:$c");
}
void main(){
        print("Hello Dart Programming....");
        add(50,30);
        sub(90,60);
        mul(8,60);
        div(90,5);
}
```

**Output:**

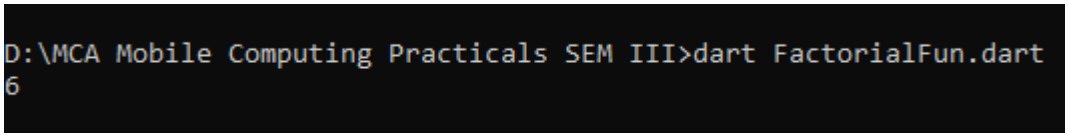



**Practical 2: Write a Dart program to calculate factorial of given number using recursive function.**

**Code:**

```
void main(){
        print(factorial(3));
}
factorial(number){

        if(number<=0){
        //termination case
        return 1;
        }
        else{
        return(number*factorial(number-1));
        //function invokes itself
        }
}
```

**Output:**




**Practical 3: Write a dart program to perform user define exception**

**Code:**

```
class EmptyNameException implements Exception {
```

```
        String cause;
        EmptyNameException(this.cause);
}
void displayName(str){
  if(str.length>0){
        print(str);
   }
  else{
        throw new EmptyNameException('Name is empty');
        }
}

void main(){
 var name= ' ';
  try{
        displayName(name);
  }
  on EmptyNameException{
        print('Given name is empty');
   }
print('MY OWN CUSTOM EXCEPTION');

}
```

**Output:**

```
D:\MCA Mobile Computing Practicals SEM III>dart CustomException.dart
Given name is empty
MY OWN CUSTOM EXCEPTION
```

**Practical 4: Write a flutter app to display title ‘Hello IDOL Students”.**

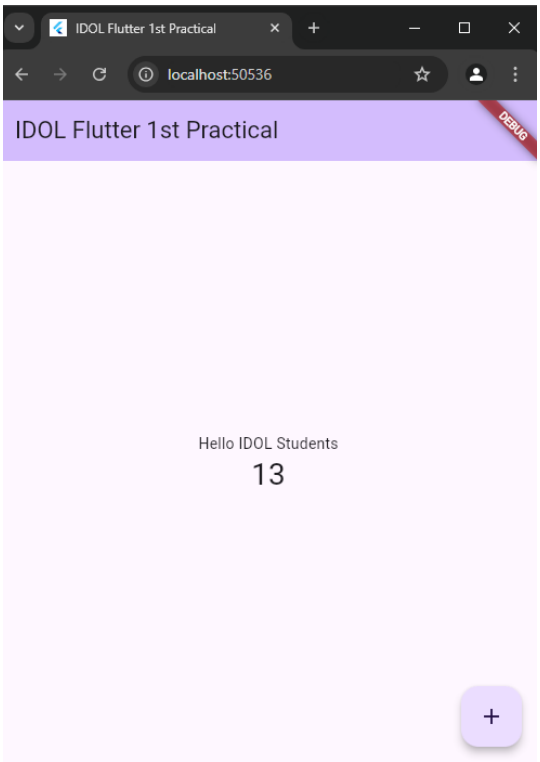
**Code:**

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

```
void main() {
  runApp(const MyApp());
}
class MyApp extends StatelessWidget {
  const MyApp({super.key});

  // This widget is the root of your application.
  @override
  Widget build(BuildContext context) {
    return MaterialApp(
      title: 'IDOL Flutter 1st Practical',
      theme: ThemeData(

        colorScheme: ColorScheme.fromSeed(seedColor: Colors.deepPurple),
        useMaterial3: true,
      ),
      home: const MyHomePage(title: 'IDOL Flutter 1st Practical'),
    );
  }
}

class MyHomePage extends StatefulWidget {
  const MyHomePage({super.key, required this.title});
  final String title;

  @override
  State<MyHomePage> createState() => _MyHomePageState();
}

class _MyHomePageState extends State<MyHomePage> {
  int _counter = 0;

  void _incrementCounter() {
    setState(() {

      _counter++;
    });
  }

  @override
  Widget build(BuildContext context) {

    return Scaffold(
      appBar: AppBar(

        backgroundColor: Theme.of(context).colorScheme.inversePrimary,

        title: Text(widget.title),
      ),
      body: Center(
        child: Column(
```

```
          mainAxisAlignment: MainAxisAlignment.center,
          children: <Widget>[
            const Text(
              'Hello IDOL Students',
            ),
            Text(
              '$_counter',
              style: Theme.of(context).textTheme.headlineMedium,
            ),
          ],
        ),
      ),
      floatingActionButton: FloatingActionButton(
        onPressed: _incrementCounter,
        tooltip: 'Increment',
        child: const Icon(Icons.add),
      ), // This trailing comma makes auto-formatting nicer for build methods.
    );
} }
```

**Output:**



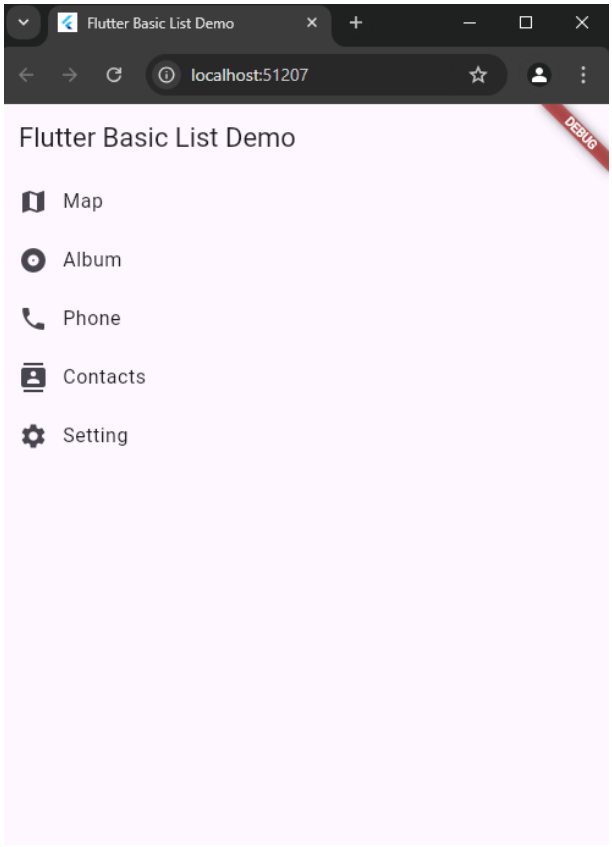

**Practical 5: Demonstrate flutter app to display list of built in icons along with list.**

**Code:**

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

void main() => runApp(MyApp());
class MyApp extends StatelessWidget{
  @override
```

```
  Widget build(BuildContext context){
    final appTitle = 'Flutter Basic List Demo';
    return MaterialApp(
      title: appTitle,
      home: Scaffold(
        appBar: AppBar(
          title: Text(appTitle),
        ),
        body: ListView(
          children:<Widget>[
            ListTile(
              leading: Icon(Icons.map),
              title: Text('Map'),
            ),
            ListTile(
              leading: Icon(Icons.album),
              title: Text('Album'),
            ),
            ListTile(
              leading: Icon(Icons.phone),
              title: Text('Phone'),
            ),
            ListTile(
              leading: Icon(Icons.contacts),
              title: Text('Contacts'),
            ),
            ListTile(
              leading: Icon(Icons.settings),
              title: Text('Setting'),
            ),
          ],
        ),
      ),
    );
  }
}
```

**Output:**

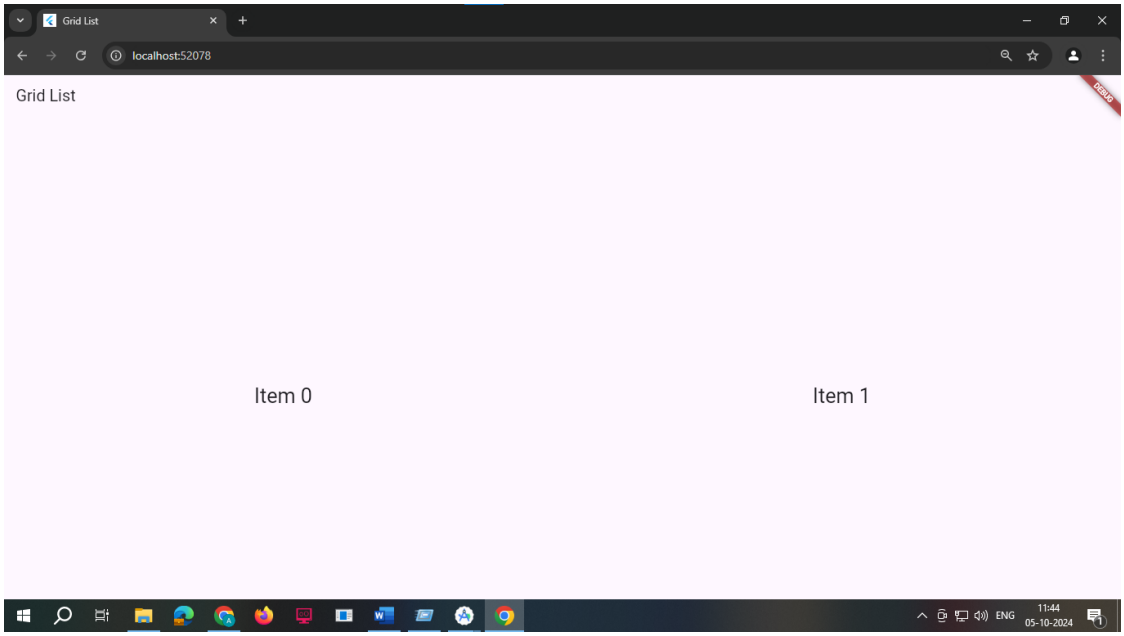


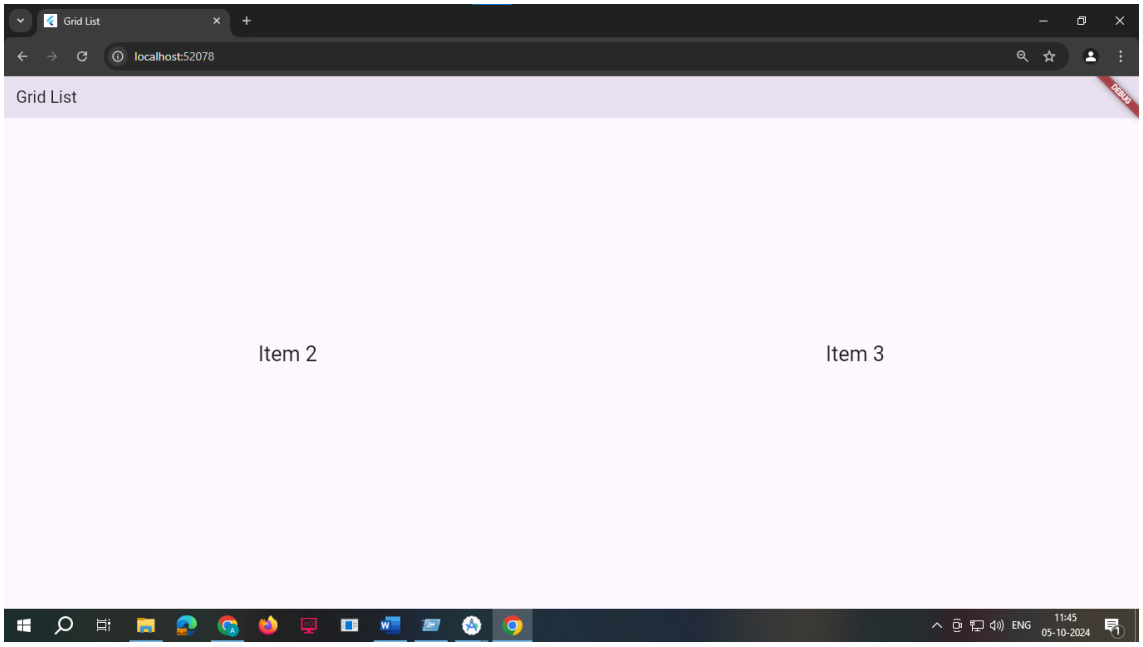

**Practical 6: To demonstrate grid layout using flutter**

**Code:**

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

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

  @override
  Widget build(BuildContext context){
    const title = "Grid List";

    return MaterialApp(
      title: title,
      home: Scaffold(
        appBar: AppBar(
          title: const Text(title),
        ),
        body: GridView.count(
          crossAxisCount: 2,
          children: List.generate(100, (index)
    {
    return Center(
    child: Text(
    'Item $index',
    style: Theme.of(context).textTheme.headlineMedium
    ),
    );
    })
        ),
      ),
    );
  }
}
```

**Output:**




**Practical 7: Write an android app to design basic user interface.**

**Code:**

MainActivity.java:

```
package com.example.myuserapplication;

import androidx.appcompat.app.AppCompatActivity;

import android.os.Bundle;

public class MainActivity extends AppCompatActivity {

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

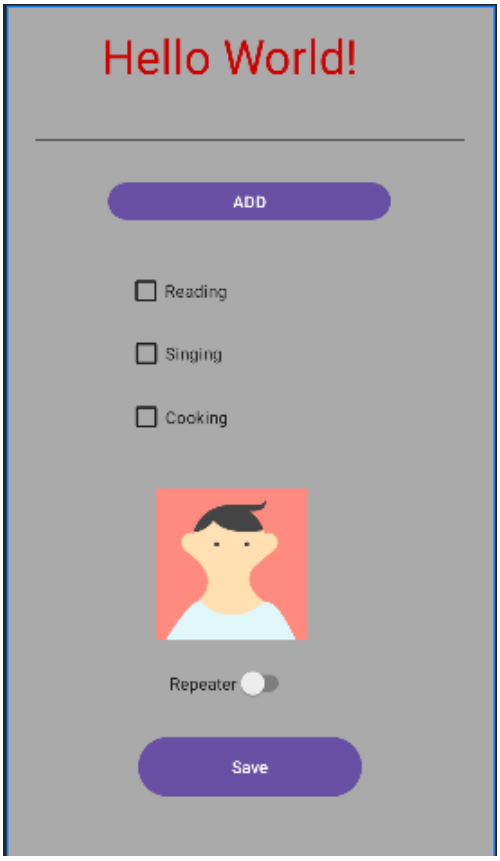
activity_main.xml:

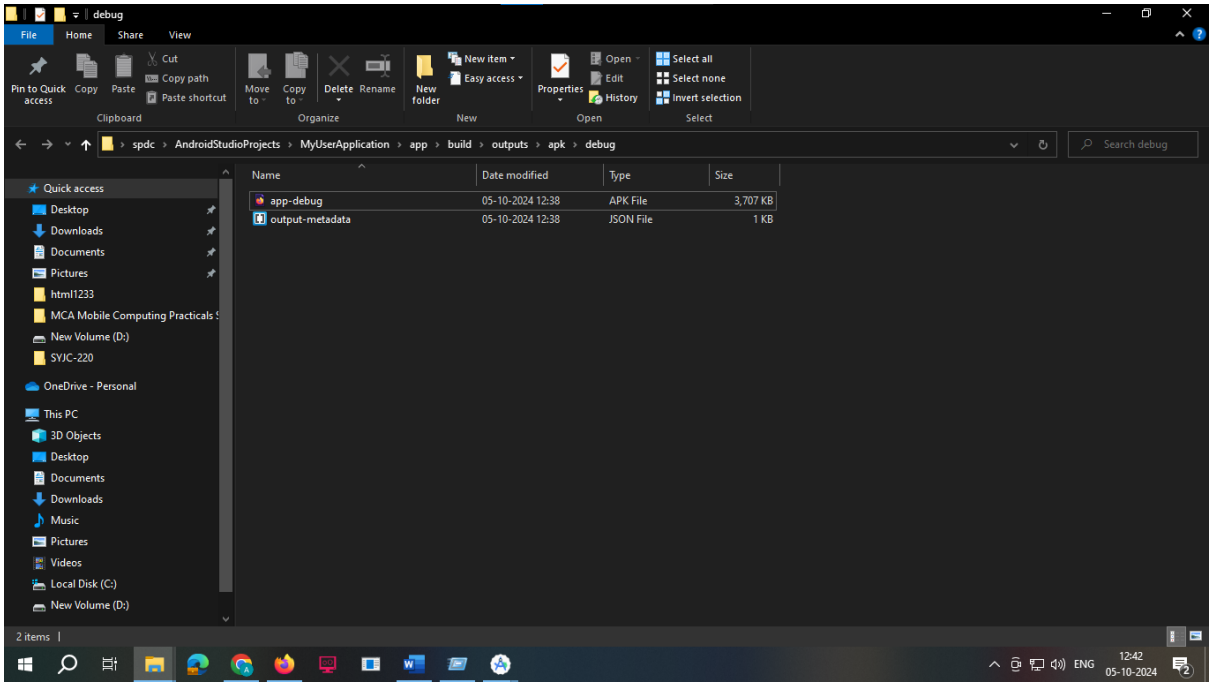
```
<?xml version="1.0" encoding="utf-8"?>
<androidx.constraintlayout.widget.ConstraintLayout
xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:app="http://schemas.android.com/apk/res-auto"
    xmlns:tools="http://schemas.android.com/tools"
    android:layout_width="match_parent"
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

    <TextView
        android:id="@+id/textView"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="Hello World!"
        android:textColor="@android:color/holo_red_dark"
        android:textSize="40dp"
        app:layout_constraintBottom_toBottomOf="parent"
        app:layout_constraintEnd_toEndOf="parent"
        app:layout_constraintHorizontal_bias="0.412"
        app:layout_constraintStart_toStartOf="parent"
        app:layout_constraintTop_toTopOf="parent"
        app:layout_constraintVertical_bias="0.023" />

    <EditText
        android:id="@+id/editTextText"
        android:layout_width="371dp"
        android:layout_height="45dp"
        android:ems="10"
        android:inputType="text"
        app:layout_constraintEnd_toEndOf="parent"
        app:layout_constraintStart_toStartOf="parent"
```

```
    tools:layout_editor_absoluteY="77dp" />

<Button
    android:id="@+id/button"
    android:layout_width="239dp"
    android:layout_height="39dp"
    android:text="ADD"
    app:layout_constraintEnd_toEndOf="parent"
    app:layout_constraintHorizontal_bias="0.497"
    app:layout_constraintStart_toStartOf="parent"
    tools:layout_editor_absoluteY="146dp" />

<CheckBox
    android:id="@+id/checkBox"
    android:layout_width="208dp"
    android:layout_height="45dp"
    android:text="Reading"
    app:layout_constraintEnd_toEndOf="parent"
    app:layout_constraintStart_toStartOf="parent"
    tools:layout_editor_absoluteY="219dp" />

<CheckBox
    android:id="@+id/checkBox2"
    android:layout_width="206dp"
    android:layout_height="44dp"
    android:text="Singing"
    app:layout_constraintEnd_toEndOf="parent"
    app:layout_constraintHorizontal_bias="0.497"
    app:layout_constraintStart_toStartOf="parent"
    tools:layout_editor_absoluteY="272dp" />

<CheckBox
    android:id="@+id/checkBox3"
    android:layout_width="209dp"
    android:layout_height="46dp"
    android:text="Cooking"
    app:layout_constraintBottom_toBottomOf="parent"
    app:layout_constraintEnd_toEndOf="parent"
    app:layout_constraintHorizontal_bias="0.504"
    app:layout_constraintStart_toStartOf="parent"
    app:layout_constraintTop_toTopOf="parent"
    app:layout_constraintVertical_bias="0.474" />

<ImageView
    android:id="@+id/imageView"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    tools:layout_editor_absoluteX="127dp"
    tools:layout_editor_absoluteY="408dp"
    tools:srcCompat="@tools:sample/avatars" />
```

```
    <Switch
        android:id="@+id/switch1"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="Repeater"
        app:layout_constraintBottom_toBottomOf="parent"
        app:layout_constraintEnd_toEndOf="parent"
        app:layout_constraintHorizontal_bias="0.448"
        app:layout_constraintStart_toStartOf="parent"
        app:layout_constraintTop_toTopOf="@+id/textView"
        app:layout_constraintVertical_bias="0.79" />

    <Button
        android:id="@+id/button2"
        android:layout_width="189dp"
        android:layout_height="58dp"
        android:text="Save"
        app:layout_constraintEnd_toEndOf="parent"
        app:layout_constraintStart_toStartOf="parent"
        tools:layout_editor_absoluteY="615dp" />

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

**Output:**




**Practical 8: Creating android program to demonstrate the use of Internal Storage**

**Code:**

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
```

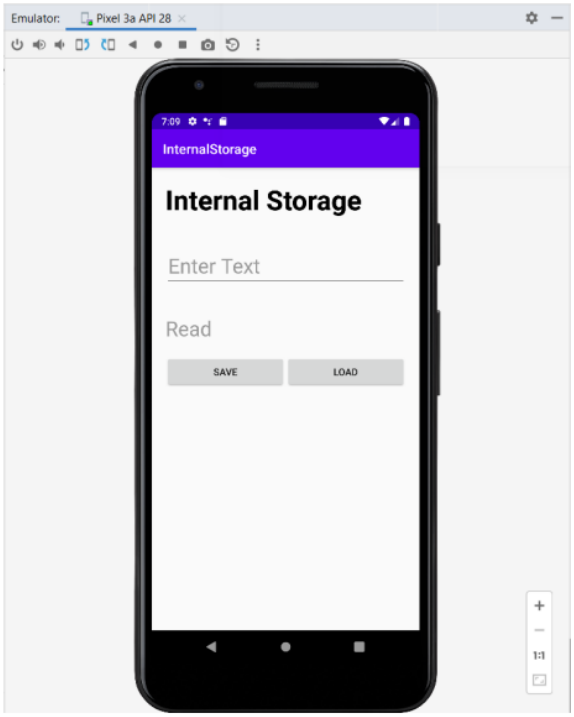
```
 android:id="@+id/button"
 android:layout_width="wrap_content"
 android:layout_height="wrap_content"
 android:layout_weight="1"
 android:text="Save" />
<Button
 android:id="@+id/button2"
 android:layout_width="wrap_content"
 android:layout_height="wrap_content"
 android:layout_weight="1"
 android:text="Load"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

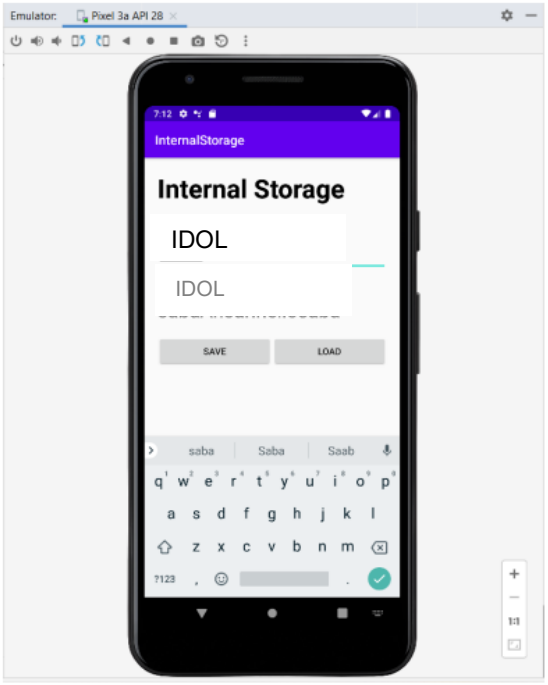
**MainActivity.java**

```
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:**


Emulator: Pixel 3a API 28
7:09
InternalStorage
Internal Storage
Enter Text
Read
SAVE
LOAD


Emulator: Pixel 3a API 28
7:12
InternalStorage
Internal Storage
IDOL
IDOL
SAVE
LOAD
saba
Saba
Saab