

## CONTENTS

S. No.	Date	List of Experiments	CO	Page No.	Signature
1.	16/02/2024	Study of Android Studio IDE (Latest)	1	03	
2.	16/02/2024	Creation and Deletion of Virtual Devices	1	06	
3.	16/02/2024	Displaying “Welcome to MCA” in Virtual & Physical	1	11	
4.	20/02/2024	Use the resources - – Integer; String; Array; Boolean; Dimension, Colors in XML and Code Files	1	12	
5.	22/02/2024	Designing Simple Toast	2	15	
6.	23/02/2024	Designing Custom Toast	2	17	
7.	27/02/2024	Designing Dialog – Simple, Positive, Negative, Neutral	2	19	
8.	01/03/2024	Navigation in Android Ex. Login Applications	2	24	
9.	01/03/2024	Designing Simple Calculator Application	2	28	
10.	04/03/2024	Designing User Interface based on Layouts	2	38	
11.	05/03/2024	Applying colors on Activity background (AutoCompleteTextView)	2	46	

## CONTENTS

S. No.	Date	List of Experiments	CO	Page No.	Signature
12.	08/03/2024	Applying colors on Activity background (MultiAutoCompleteTextView)	2	48	
13.	12/03/2024	Applying colors on Activity background (Spinner)	2	51	
14.	21/03/2024	Design “ABCD” using Scroll view. (Adapter)	3	54	
15.	22/03/2024	Design “Country & its flag” using Pictures controls.	3	59	
16.	29/03/2024	Designing Option menu.	3	62	
17.	29/03/2023	Designing Context menu.	3	64	
18.	04/04/2023	Designing Popup menu.	3	66	
19.	15/04/2023	Read, Write and Append Operation on File	4	68	
20.	18/04/2023	Read and Write using Shared Preference	4	72	
21.	22/04/2023	Perform CRUD operations on SQLite Database	4	76	
22.	26/04/2023	Study of Firebase or Cloud Database (Latest)	4	87	

## P1. Study of Android Studio IDE (Latest):-

### Overview :-

- It is based on IntelliJ IDEA, a Java integrated development environment for software, and incorporates its code editing and developer tools.
- It is the official integrated development environment (IDE) for Android application development.
- Android Studio provides more features that enhance our productivity while building Android apps.
- Android Studio was announced on 16th May 2013 at the Google I/O conference as an official IDE for Android app development.
- It started its early access preview from version 0.1 in May 2013. The first stable built version was released in December 2014, starts from version 1.0.

### Features :-

1. Android Studio's code editor provides features like syntax highlighting, code completion, and code refactoring.
2. It includes a fast and feature-rich emulator that enables developers to test their apps on various Android device configurations without needing physical hardware.
3. It integrates seamlessly with ADB, allowing developers to install, debug, and profile their apps directly from the IDE.
4. Android Studio uses the Gradle build system to automate the build process and manage dependencies for Android projects.
5. Android Studio provides built-in support for version control systems like Git, enabling developers to manage their project's source code efficiently.
6. It includes powerful profiler tools for analyzing the performance and behavior of Android apps.
7. It supports a rich ecosystem of plugins and extensions that extend its functionality and cater to specific developer needs.

## **P2.Creation and Deletion of Virtual Devices:-**

### **\*Creation of Virtual Devices:**

#### **1. Open Android Studio:**

- Launch Android Studio on your computer.

#### **2. Open AVD Manager:**

- Click on the "AVD Manager" icon in the toolbar, or go to "Tools" > "AVD Manager" from the top menu.

#### **3. Create Virtual Device:**

- In the AVD Manager window, click on the "Create Virtual Device" button.

#### **4. Choose Device Definition:**

- Select a device definition from the list of available options (e.g., Pixel, Nexus).
- Click "Next" to proceed.

#### **5. Select System Image:**

- Choose a system image for the selected device definition (e.g., Android version and API level). - Click "Next" to continue.

#### **6. Configure Virtual Device:**

- Customize the virtual device configuration settings, such as name, orientation, screen size, and RAM size.
- Click "Finish" to create the virtual device.

#### **7. Verify Creation:**

- Once the virtual device is created, it will appear in the AVD Manager window with its name and specifications.

**\*Deletion of Virtual Devices:****1. Open AVD Manager:**

- Launch Android Studio and open the AVD Manager by clicking on the "AVD Manager" icon in the toolbar or going to "Tools" > "AVD Manager" from the top menu.

**2. Select Virtual Device:**

- In the AVD Manager window, select the virtual device you want to delete from the list of available devices.

**3. Delete Virtual Device:**

- Click on the "Delete" icon (trash can icon) next to the selected virtual device.

**4. Confirm Deletion:**

- A confirmation dialog will appear asking if you're sure you want to delete the virtual device. Click "Delete" to confirm.

**5. Verify Deletion:**

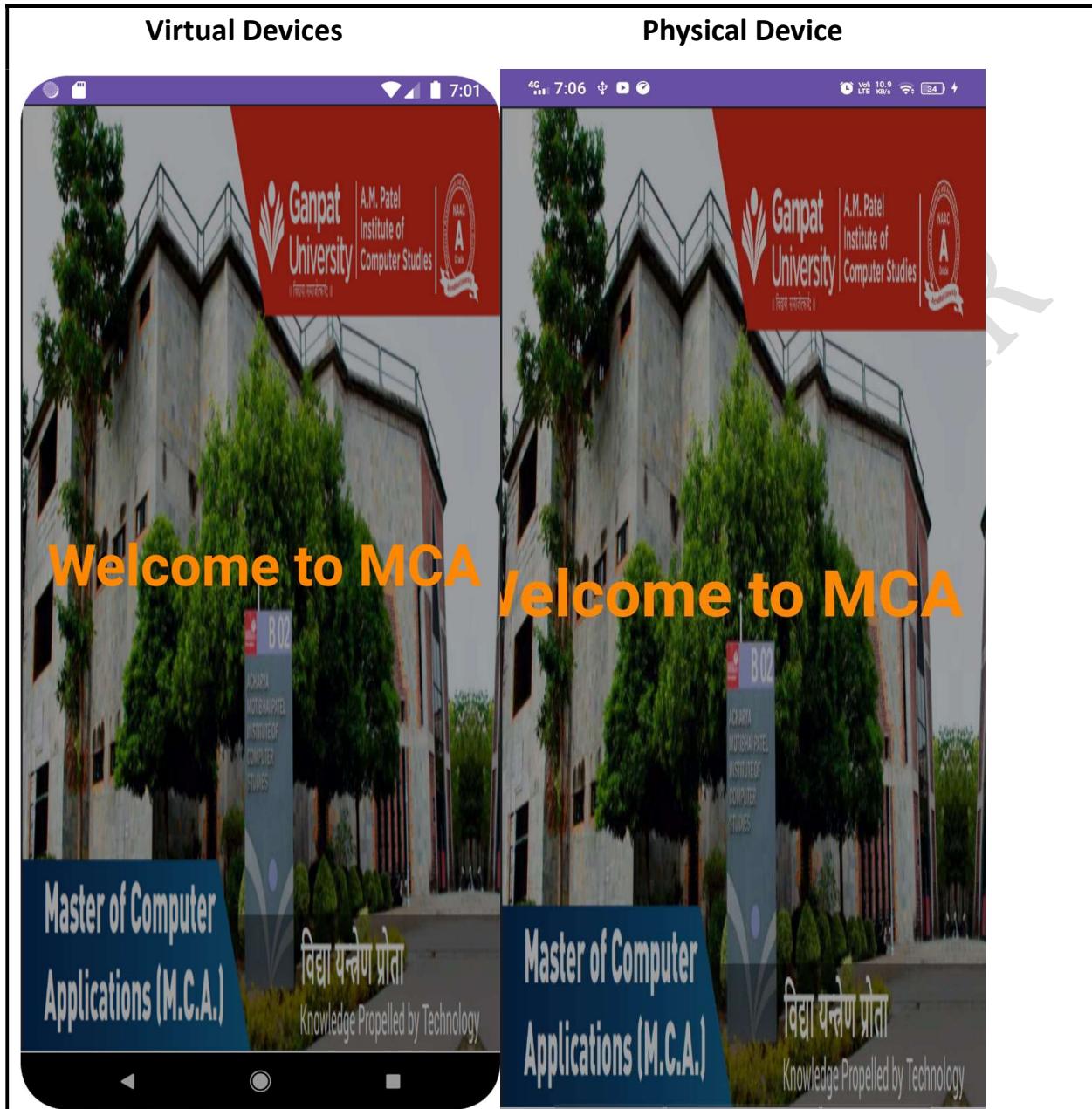
- Once deleted, the virtual device will be removed from the list in the AVD Manager window.

**P3. Displaying “Welcome to MCA” in Virtual & Physical**

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

```
<ImageView  
    android:id="@+id/imageView2"  
    android:layout_width="match_parent"  
    android:layout_height="match_parent"  
    android:layout_marginBottom="2dp"  
    app:layout_constraintBottom_toBottomOf="parent"  
    app:layout_constraintEnd_toEndOf="parent"  
    android:scaleType="fitXY"  
    app:layout_constraintStart_toStartOf="parent"  
    app:layout_constraintTop_toTopOf="parent"  
    app:srcCompat="@drawable/background" />  
  
<View  
    android:id="@+id/blurOverlayView"  
    android:layout_width="match_parent"  
    android:layout_height="match_parent"  
    android:background="#60000000" />  
  
<TextView  
    android:id="@+id/textView2"  
    android:layout_width="405dp"  
    android:layout_height="90dp"  
    android:text="Welcome to MCA"  
    android:textAlignment="center"  
    android:textColor="@android:color/holo_orange_dark"  
    android:textSize="48sp"  
    android:textStyle="bold"  
    app:layout_constraintBottom_toBottomOf="parent"  
    app:layout_constraintEnd_toEndOf="parent"  
    app:layout_constraintHorizontal_bias="1.0"  
    app:layout_constraintStart_toStartOf="parent"  
    app:layout_constraintTop_toTopOf="@+id/imageView2" />  
  
</androidx.constraintlayout.widget.ConstraintLayout>
```



#### Q4. Use the resources – Integer; String; Array; Boolean; Dimension, Colors in XML and Code Files



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="match_parent"
        android:layout_height="wrap_content"
        android:text="Array Values"
        android:paddingTop="10sp"
        android:textStyle="bold"
        android:textAlignment="center"
        android:textSize="28sp"
        app:layout_constraintEnd_toEndOf="parent"
        app:layout_constraintHorizontal_bias="0.16"
        app:layout_constraintStart_toStartOf="parent"
        app:layout_constraintTop_toTopOf="parent" />

    <Spinner
        android:id="@+id/spinner"
        style="@style/SpinnerStyle"
        android:layout_width="match_parent"
        android:layout_height="70dp"
        android:layout_marginEnd="2dp"
        android:entries="@array/myArr"
        android:textSize="100sp"
        app:layout_constraintEnd_toEndOf="parent"
        app:layout_constraintHorizontal_bias="0.0"
        app:layout_constraintStart_toStartOf="parent"
        app:layout_constraintTop_toBottomOf="@+id/textView" />

    <TextView
        android:id="@+id/textView2"
        android:layout_width="409dp"
        android:layout_height="40dp"
        android:layout_marginTop="25dp"
        android:text="String Values "
        android:textAlignment="center"
        android:textSize="28sp"
        android:textStyle="bold"
        app:layout_constraintEnd_toEndOf="parent"
        app:layout_constraintStart_toStartOf="parent"
```

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

<TextView
    android:id="@+id/textView3"
    android:layout_width="match_parent"
    android:layout_height="55dp"
    android:textSize="20sp"
    android:textAlignment="center"
    android:paddingTop="15sp"
    android:textStyle="bold"
    app:layout_constraintEnd_toEndOf="parent"
    android:text="@string/my_string"
    app:layout_constraintStart_toStartOf="parent"
    app:layout_constraintTop_toBottomOf="@+id/textView2" />

<TextView
    android:id="@+id/textView4"
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:layout_marginTop="25dp"
    android:text="Boolean Values"
    android:textAlignment="center"
    android:textSize="28sp"
    android:textStyle="bold"
    app:layout_constraintEnd_toEndOf="parent"
    app:layout_constraintStart_toStartOf="parent"
    app:layout_constraintTop_toBottomOf="@+id/textView3" />

<CheckBox
    android:id="@+id/checkBoxBoolean"
    android:layout_width="50dp"
    android:layout_height="75dp"
    android:layout_marginStart="120dp"
    android:layout_marginEnd="120dp"
    android:textAlignment="center"
    android:textSize="18sp"
    app:layout_constraintEnd_toEndOf="parent"
    app:layout_constraintHorizontal_bias="0.0"
    app:layout_constraintStart_toStartOf="parent"
    app:layout_constraintTop_toBottomOf="@+id/textView4" />

<TextView
    android:id="@+id/textViewBoolean"
    android:layout_width="121dp"
    android:layout_height="75dp"
    android:paddingTop="18sp"
    android:text="False"
    android:textSize="23sp"
    app:layout_constraintEnd_toEndOf="parent"
    app:layout_constraintHorizontal_bias="0.0"
    app:layout_constraintStart_toEndOf="@+id/checkBoxBoolean"
    app:layout_constraintTop_toBottomOf="@+id/textView4" />

<TextView
```

```
        android:id="@+id/textView5"
        android:layout_width="match_parent"
        android:layout_height="50dp"
        android:layout_marginTop="25dp"
        android:text="Integer"
        android:textAlignment="center"
        android:textSize="28sp"
        android:textStyle="bold"
        app:layout_constraintEnd_toEndOf="parent"
        app:layout_constraintStart_toStartOf="parent"
        app:layout_constraintTop_toBottomOf="@+id/checkBoxBoolean" />

<TextView
    android:id="@+id/textView6"
    android:layout_width="match_parent"
    android:layout_height="48dp"
    app:layout_constraintEnd_toEndOf="parent"
    android:textSize="20sp"
    android:textStyle="bold"
    android:textAlignment="center"
    app:layout_constraintStart_toStartOf="parent"
    app:layout_constraintTop_toBottomOf="@+id/textView5" />

<TextView
    android:id="@+id/textView7"
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:layout_marginTop="25dp"
    android:textSize="28sp"
    android:textAlignment="center"
    android:textStyle="bold"

    android:text="Dimension"
    app:layout_constraintEnd_toEndOf="parent"
    app:layout_constraintStart_toStartOf="parent"
    app:layout_constraintTop_toBottomOf="@+id/textView6" />

<TextView
    android:id="@+id/textView8"
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:paddingTop="10sp"
    android:text="5"
    android:textStyle="bold"
    android:textAlignment="center"
    app:layout_constraintEnd_toEndOf="parent"
    app:layout_constraintStart_toStartOf="parent"
    app:layout_constraintTop_toBottomOf="@+id/textView7" />

<TextView
    android:id="@+id/textView9"
    android:layout_width="match_parent"
    android:layout_height="93dp"
    android:layout_marginTop="20dp"
```

```
    app:layout_constraintBottom_toBottomOf="parent"
    app:layout_constraintTop_toBottomOf="@+id/textView8"
    tools:layout_editor_absoluteX="2dp" />

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



MainActivity.java

```
package com.example.practicals4;

import androidx.appcompat.app.AppCompatActivity;
import android.os.Bundle;
import android.widget.CheckBox;
import android.widget.TextView;
import android.widget.CompoundButton;
import android.util.TypedValue;
import androidx.core.content.ContextCompat;

public class MainActivity extends AppCompatActivity {
    CheckBox checkBoxBoolean;
    TextView textViewBoolean, textViewIntegers, textViewDimension;

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

        //=====for CheckBox=====
        checkBoxBoolean = findViewById(R.id.checkBoxBoolean);
        textViewBoolean = findViewById(R.id.textViewBoolean);
        checkBoxBoolean.setOnCheckedChangeListener(new CompoundButton.OnCheckedChangeListener() {
            @Override
            public void onCheckedChanged(CompoundButton buttonView, boolean isChecked) {
                // Update the text of the TextView based on the checked state
                textViewBoolean.setText(isChecked ? "True" : "False");
            }
        });
        //=====for Integer value=====

        int intValue = getResources().getInteger(R.integer.my_integer);
        textViewIntegers = findViewById(R.id.textView6);
        textViewIntegers.setText(String.valueOf(intValue));

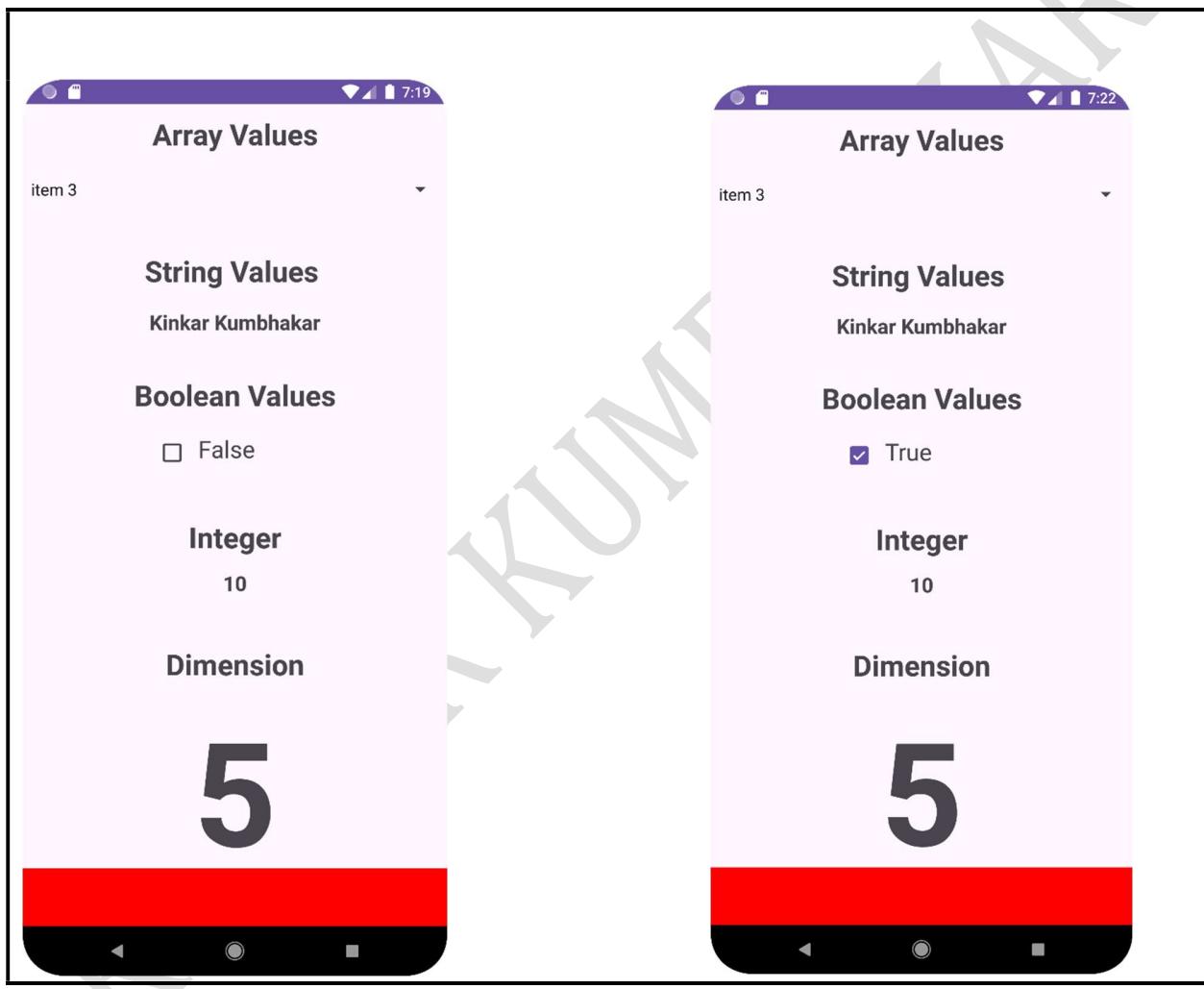
        //=====for Dimension=====

        TextView tv = findViewById(R.id.textView8);
        float textViewDimension = getResources().getDimension(R.dimen.my_dimension);
        tv.setTextSize(TypedValue.COMPLEX_UNIT_PX, textViewDimension);
```

```
=====for Colors=====
```

```
TextView textViewColors = findViewById(R.id.textView9);
int myColor = ContextCompat.getColor(this, R.color.my_color);
textViewColors.setBackgroundColor(myColor);

    }
```





arrays.xml

```
<?xml version="1.0" encoding="utf-8"?>
<resources>
    <array name="myArr">
        <item>item 1</item>
        <item>item 2</item>
        <item>item 3</item>
    </array>
</resources>
```



bools.xml

```
<?xml version="1.0" encoding="utf-8"?>
<resources>
    <bool name="is_enabled">true</bool>
</resources>
```



colors.xml

```
<?xml version="1.0" encoding="utf-8"?>
<resources>
    <color name="black">#FF000000</color>
    <color name="white">#FFFFFF</color>
    <color name="my_color">#FF0000</color>
</resources>
```



dimens.xml

```
<?xml version="1.0" encoding="utf-8"?>
<resources>
    <dimen name="my_dimension">140dp</dimen>
</resources>
```



integers.xml

```
<?xml version="1.0" encoding="utf-8"?>
<resources>
    <dimen name="my_dimension">140dp</dimen>
</resources>
```



strings.xml

```
<resources>
    <string name="app_name">Practicals4</string>
    <string name="my_string">Kinkar Kumbhakar</string>
</resources>
```

## Q5. Designing Simple Toast



activity\_main.xml

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

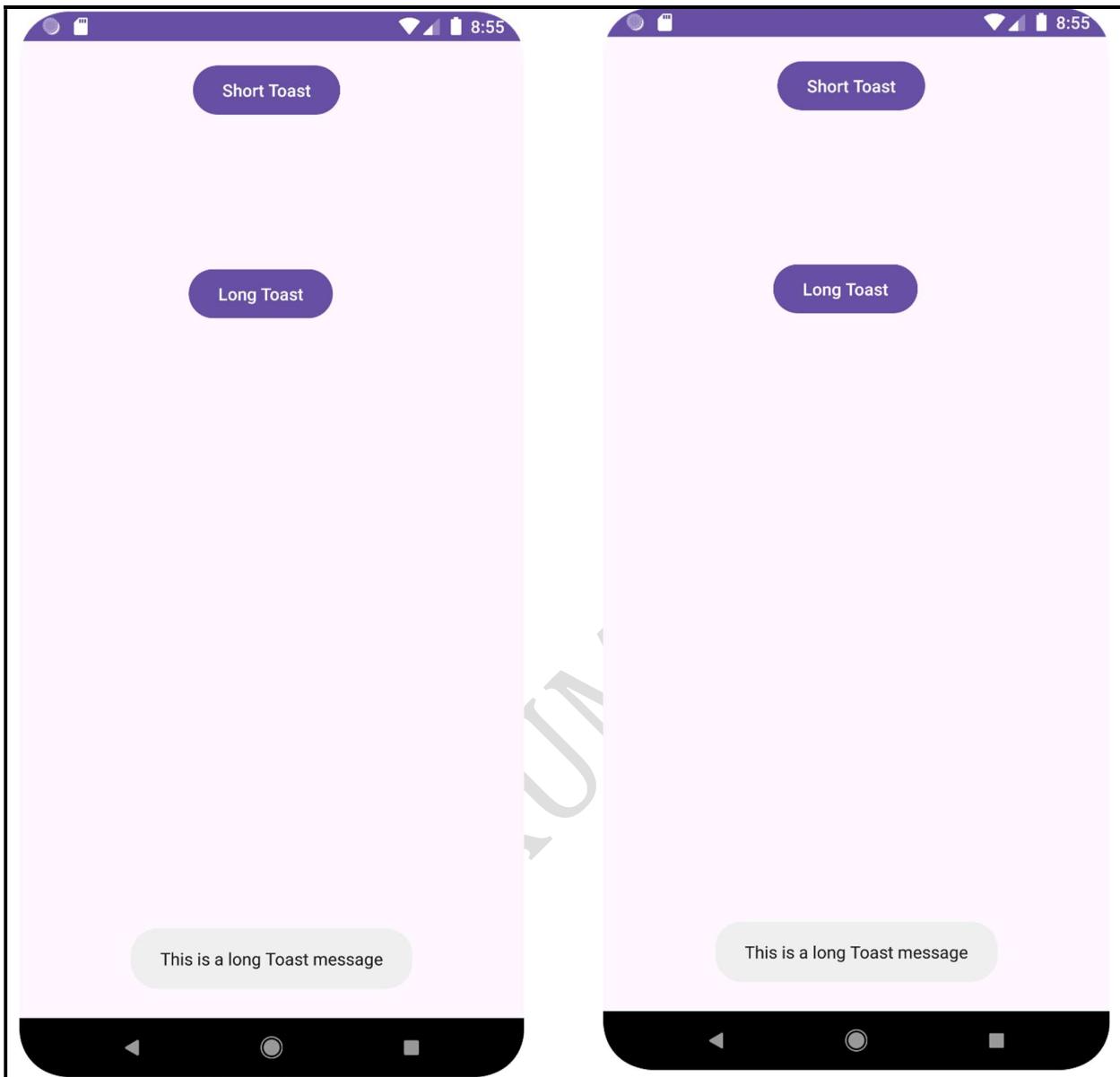
    <Button
        android:id="@+id/button_short_toast"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_alignParentEnd="true"
        android:layout_marginEnd="134dp"
        android:layout_marginBottom="16dp"
        android:onClick="showToast"
        android:text="Short Toast" />
```

```
<Button  
    android:id="@+id/button_long_toast"  
    android:layout_width="wrap_content"  
    android:layout_height="wrap_content"  
    android:layout_below="@+id/button_short_toast"  
    android:layout_alignParentEnd="true"  
    android:layout_marginTop="102dp"  
    android:layout_marginEnd="140dp"  
    android:onClick="showLongToast"  
    android:text="Long Toast" />  
  
</RelativeLayout>
```



MainActivity.java

```
package com.example.myapplication;  
  
import androidx.appcompat.app.AppCompatActivity;  
import android.os.Bundle;  
import android.view.View;  
import android.widget.Toast;  
import android.view.LayoutInflater;  
  
public class MainActivity extends AppCompatActivity {  
  
    @Override  
    protected void onCreate(Bundle savedInstanceState) {  
        super.onCreate(savedInstanceState);  
        setContentView(R.layout.activity_main);  
    }  
    public void showToast(View view) {  
        Toast.makeText(this, "This is a short Toast message", Toast.LENGTH_SHORT).show();  
    }  
  
    public void showLongToast(View view) {  
        Toast.makeText(this, "This is a long Toast message", Toast.LENGTH_LONG).show();  
    }  
}
```



## P6. Designing Custom Toast



activity\_main.xml

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

```
<Button  
    android:id="@+id/customToastBtn"  
    android:layout_width="wrap_content"  
    android:layout_height="wrap_content"  
    android:layout_centerInParent="true"  
    android:textColor="@color/white"  
    android:padding="8sp"  
    android:background="#0A433E"  
    android:text="Show Custom Toast" />  
  
</RelativeLayout>
```



custom\_layout.xml

```
<?xml version="1.0" encoding="utf-8"?>  
<LinearLayout xmlns:android="http://schemas.android.com/apk/res/android"  
    android:id="@+id/custom_toast_container"  
    android:layout_width="wrap_content"  
    android:layout_height="wrap_content"  
    android:padding="16dp"  
    android:background="#E81118"  
    android:orientation="horizontal">  
  
    <TextView  
        android:id="@+id/custom_toast_text"  
        android:layout_width="wrap_content"  
        android:layout_height="wrap_content"  
        android:textColor="#FFFFFF"  
        android:textSize="16sp"  
        android:text="This is a Custom Toast!" />  
  
</LinearLayout>
```



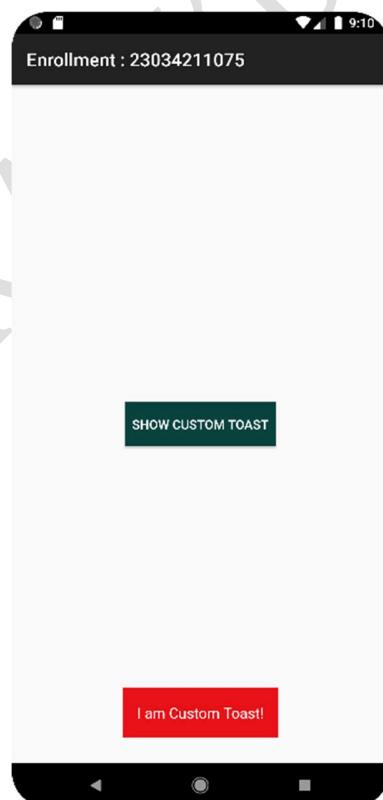
MainActivity.java

```
package com.example.practical6;  
  
import android.os.Bundle;  
import android.view.Gravity;  
import android.view.LayoutInflater;  
import android.view.View;  
import android.widget.Button;  
import android.widget.TextView;  
import android.widget.Toast;  
import androidx.appcompat.app.AppCompatActivity;
```

```
public class MainActivity extends AppCompatActivity {
    Button btn;
    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_main);
        btn = (Button) findViewById(R.id.customToastBtn);
        btn.setOnClickListener(new View.OnClickListener() {
            @Override
            public void onClick(View v) {
                LayoutInflator inflater = getLayoutInflater();
                View layout = inflater.inflate(R.layout.custom_layout, findViewById(R.id.custom_toast_container));

                // Set custom text
                TextView text = layout.findViewById(R.id.custom_toast_text);
                text.setText("I am Custom Toast!");

                // Create custom toast and set properties
                Toast toast = new Toast(getApplicationContext());
                toast.setGravity(Gravity.BOTTOM | Gravity.CENTER_HORIZONTAL, 0, 100);
                toast.setDuration(Toast.LENGTH_SHORT);
                toast.setView(layout);
                toast.show();
            }
        });
    }
}
```



## P7. Designing Dialog – Simple, Positive, Negative, Neutral



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="#F3ECEC"
    tools:context=".MainActivity">

    <Button
        android:id="@+id/simpleDialogBtn"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_centerHorizontal="true"
        android:background="#333DEA"
        android:padding="15sp"
        android:text="Simple Dialog"
        android:textColor="#fff"
        app:layout_constraintBottom_toTopOf="@+id/positiveDialogBtn"
        app:layout_constraintEnd_toEndOf="parent"
        app:layout_constraintHorizontal_bias="0.5"
        app:layout_constraintStart_toStartOf="parent"
        app:layout_constraintTop_toTopOf="parent"
        app:layout_constraintVertical_bias="0.5" />

    <Button
        android:id="@+id/positiveDialogBtn"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_centerHorizontal="true"
        android:background="#333DEA"
        android:padding="15sp"
        android:text="Positive Dialog"
        android:textColor="#fff"
        app:layout_constraintBottom_toTopOf="@+id/negativeDialogBtn"
        app:layout_constraintEnd_toEndOf="parent"
        app:layout_constraintHorizontal_bias="0.5"
        app:layout_constraintStart_toStartOf="parent"
        app:layout_constraintTop_toBottomOf="@+id/simpleDialogBtn"
        app:layout_constraintVertical_bias="0.5" />

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

```
        android:layout_centerHorizontal="true"
        android:background="#333DEA"
        android:padding="15sp"
        android:text="Negative Dialog"
        android:textColor="#fff"
        app:layout_constraintBottom_toTopOf="@+id/neutralDialogBtn"
        app:layout_constraintEnd_toEndOf="parent"
        app:layout_constraintHorizontal_bias="0.5"
        app:layout_constraintStart_toStartOf="parent"
        app:layout_constraintTop_toBottomOf="@+id/positiveDialogBtn"
        app:layout_constraintVertical_bias="0.5" />

    <Button
        android:id="@+id/neutralDialogBtn"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_centerHorizontal="true"
        android:background="#333DEA"
        android:padding="15sp"
        android:text="Neutral Dialog"
        android:textColor="#fff"
        app:layout_constraintBottom_toBottomOf="parent"
        app:layout_constraintEnd_toEndOf="parent"
        app:layout_constraintHorizontal_bias="0.5"
        app:layout_constraintStart_toStartOf="parent"
        app:layout_constraintTop_toBottomOf="@+id/negativeDialogBtn"
        app:layout_constraintVertical_bias="0.5" />

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



MainActivity.java

```
package com.example.practical7;

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

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

public class MainActivity extends AppCompatActivity {
    Button simpleBtn, posBtn, negBtn, neuBtn;
    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_main);
        simpleBtn = (Button) findViewById(R.id.simpleDialogBtn);
```

```
posBtn = (Button) findViewById(R.id.positiveDialogBtn);
negBtn = (Button) findViewById(R.id.negativeDialogBtn);
neuBtn = (Button) findViewById(R.id.neutralDialogBtn);

simpleBtn.setOnClickListener(new View.OnClickListener() {
    @Override
    public void onClick(View v) {
        showDialog("Simple Dialog", "I am Simple Dialog");
    }
});

posBtn.setOnClickListener(new View.OnClickListener() {
    @Override
    public void onClick(View v) {
        showDialogWithPositiveBtn("Positive Dialog", "I am Positive Dialog");
    }
});

negBtn.setOnClickListener(new View.OnClickListener() {
    @Override
    public void onClick(View v) {
        showDialogWithNegativeBtn("Negative Dialog", "I am Negative Dialog");
    }
});

neuBtn.setOnClickListener(new View.OnClickListener() {
    @Override
    public void onClick(View v) {
        showDialogWithNeutralBtn("Neutral Dialog", "I am Neutral Dialog");
    }
});

private void showDialogWithNeutralBtn(String neutralDialog, String iAmNeutralDialog) {
    AlertDialog.Builder builder = new AlertDialog.Builder(this);
    builder.setTitle(neutralDialog)
        .setMessage(iAmNeutralDialog)
        .setNeutralButton("Dismiss", new DialogInterface.OnClickListener() {
            @Override
            public void onClick(DialogInterface dialog, int which) {
                dialog.dismiss();
            }
        });
    AlertDialog dialog = builder.create();
    dialog.show();
}

private void showDialogWithNegativeBtn(String negativeDialog, String iAmNegativeDialog) {
    AlertDialog.Builder builder = new AlertDialog.Builder(this);
    builder.setTitle(negativeDialog)
        .setMessage(iAmNegativeDialog)
        .setNegativeButton("Cancel", new DialogInterface.OnClickListener() {
```

```
@Override  
public void onClick(DialogInterface dialog, int which) {  
    dialog.dismiss();  
}  
});  
AlertDialog dialog = builder.create();  
dialog.show();  
}  
  
private void showDialogWithPositiveBtn(String positiveDialog, String iAmPositiveDialog) {  
    AlertDialog.Builder builder = new AlertDialog.Builder(this);  
    builder.setTitle(positiveDialog)  
        .setMessage(iAmPositiveDialog)  
        .setPositiveButton("OK", new DialogInterface.OnClickListener() {  
            @Override  
            public void onClick(DialogInterface dialog, int which) {  
                dialog.dismiss();  
            }  
        });  
    AlertDialog dialog = builder.create();  
    dialog.show();  
}  
  
private void showDialog(String simpledialog, String iAmSimpleDialog) {  
    AlertDialog.Builder builder = new AlertDialog.Builder(this);  
    builder.setTitle(simpledialog)  
        .setMessage(iAmSimpleDialog)  
        .setPositiveButton("OK", new DialogInterface.OnClickListener() {  
            @Override  
            public void onClick(DialogInterface dialog, int which) {  
                dialog.dismiss();  
            }  
        });  
    AlertDialog dialog = builder.create();  
    dialog.show();  
}
```



## P8. Navigation in Android Ex. Login Applications



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="#D3CCCC"
    tools:context=".MainActivity">
```

```
<Button  
    android:id="@+id/button"  
    android:layout_width="wrap_content"  
    android:layout_height="wrap_content"  
    android:background="#EF0606"  
    android:padding="12sp"  
    android:text="Go to Login Page"  
    android:textColor="#fff"  
    android:textSize="20sp"  
    app:layout_constraintBottom_toBottomOf="parent"  
    app:layout_constraintEnd_toEndOf="parent"  
    app:layout_constraintHorizontal_bias="0.5"  
    app:layout_constraintStart_toStartOf="parent"  
    app:layout_constraintTop_toTopOf="parent"  
    app:layout_constraintVertical_bias="0.5" />  
</androidx.constraintlayout.widget.ConstraintLayout>
```



activity\_login.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="#D3CCCC"  
    tools:context=".MainActivity">  
  
<Button  
    android:id="@+id/button"  
    android:layout_width="wrap_content"  
    android:layout_height="wrap_content"  
    android:background="#EF0606"  
    android:padding="12sp"  
    android:text="Go to Login Page"  
    android:textColor="#fff"  
    android:textSize="20sp"  
    app:layout_constraintBottom_toBottomOf="parent"  
    app:layout_constraintEnd_toEndOf="parent"  
    app:layout_constraintHorizontal_bias="0.5"  
    app:layout_constraintStart_toStartOf="parent"  
    app:layout_constraintTop_toTopOf="parent"  
    app:layout_constraintVertical_bias="0.5" />  
</androidx.constraintlayout.widget.ConstraintLayout>
```



## MainActivity.java

```
package com.example.practical8;

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

public class MainActivity extends AppCompatActivity {
    Button btn;
    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_main);
        btn = (Button) findViewById(R.id.button);
        btn.setOnClickListener(new View.OnClickListener() {
            @Override
            public void onClick(View v) {
                Intent i = new Intent(MainActivity.this, LoginActivity.class);
                startActivity(i);
            }
        });
    }
}
```

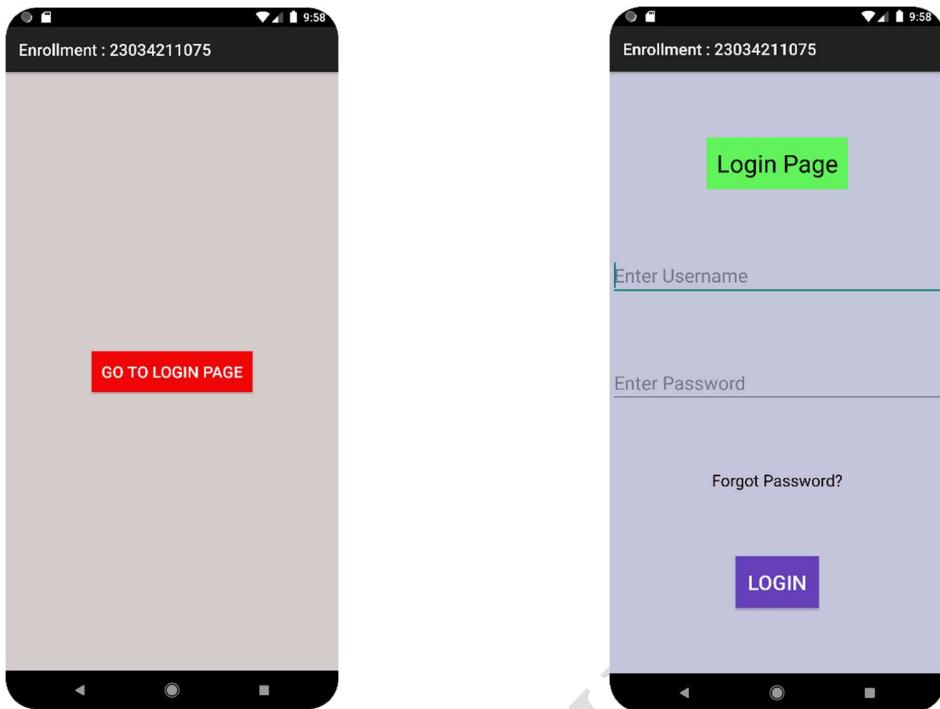


## LoginActivity.java

```
package com.example.practical8;
import androidx.appcompat.app.AppCompatActivity;
import android.os.Bundle;

public class LoginActivity extends AppCompatActivity {

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



## P9. Designing Simple Calculator Application



activity\_main.xml

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

    <TextView
        android:id="@+id/solution_tv"
        android:layout_width="match_parent"
        android:layout_height="match_parent"
        android:layout_above="@+id/result_tv"
        android:layout_marginStart="16dp"
        android:layout_marginTop="16dp"
        android:layout_marginEnd="16dp"
        android:layout_marginBottom="16dp"
        android:textAlignment="textEnd"
        android:textColor="@color/black"
        android:textSize="32dp" />
```

```
<TextView  
    android:layout_width="match_parent"  
    android:layout_height="wrap_content"  
    android:id="@+id/result_tv"  
    android:text="0"  
    android:textSize="64dp"  
    android:textAlignment="textEnd"  
    android:textColor="@color/black"  
    android:layout_above="@+id/buttons_layout"  
    android:layout_margin="16dp" />  
  
<LinearLayout  
    android:layout_width="match_parent"  
    android:layout_height="wrap_content"  
    android:orientation="vertical"  
    android:layout_alignParentBottom="true"  
    android:background="#F1F1F1"  
    android:paddingVertical="16dp"  
    android:id="@+id/buttons_layout">  
  
<LinearLayout  
    android:layout_width="match_parent"  
    android:layout_height="wrap_content"  
    android:gravity="center"  
    android:orientation="horizontal">  
  
<com.google.android.material.button.MaterialButton  
    android:layout_width="72dp"  
    android:layout_height="72dp"  
    app:cornerRadius="36dp"  
    style="@style/Widget.MaterialComponents.ExtendedFloatingActionButton"  
    android:textSize="32dp"  
    android:textColor="@color/white"  
    android:layout_margin="12dp"  
    android:id="@+id/button_c"  
    android:backgroundTint="#F44336"  
    android:text="C" />  
<com.google.android.material.button.MaterialButton  
    android:layout_width="72dp"  
    android:layout_height="72dp"  
    app:cornerRadius="36dp"  
    style="@style/Widget.MaterialComponents.ExtendedFloatingActionButton"  
    android:textSize="32dp"  
    android:textColor="@color/white"  
    android:layout_margin="12dp"  
    android:id="@+id/button_open_bracket"  
    android:backgroundTint="#A1A1A1"  
    android:text="(" />  
<com.google.android.material.button.MaterialButton  
    android:layout_width="72dp"  
    android:layout_height="72dp"  
    app:cornerRadius="36dp"
```

```
style="@style/Widget.MaterialComponents.ExtendedFloatingActionButton"
    android:textSize="32dp"
    android:textColor="@color/white"
    android:layout_margin="12dp"
    android:backgroundTint="#A1A1A1"
    android:id="@+id/button_close_bracket"
    android:text=")" />
<com.google.android.material.button.MaterialButton
    android:layout_width="72dp"
    android:layout_height="72dp"
    app:cornerRadius="36dp"
    style="@style/Widget.MaterialComponents.ExtendedFloatingActionButton"
    android:textSize="32dp"
    android:textColor="@color/white"
    android:layout_margin="12dp"
    android:id="@+id/button_divide"
    android:backgroundTint="#FF9800"
    android:text="/" />

</LinearLayout>
<LinearLayout
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:gravity="center"
    android:orientation="horizontal">

    <com.google.android.material.button.MaterialButton
        android:layout_width="72dp"
        android:layout_height="72dp"
        app:cornerRadius="36dp"
        style="@style/Widget.MaterialComponents.ExtendedFloatingActionButton"
        android:textSize="32dp"
        android:textColor="@color/white"
        android:layout_margin="12dp"
        android:id="@+id/button_7"
        android:text="7" />
    <com.google.android.material.button.MaterialButton
        android:layout_width="72dp"
        android:layout_height="72dp"
        app:cornerRadius="36dp"
        style="@style/Widget.MaterialComponents.ExtendedFloatingActionButton"
        android:textSize="32dp"
        android:textColor="@color/white"
        android:layout_margin="12dp"
        android:id="@+id/button_8"
        android:text="8" />
    <com.google.android.material.button.MaterialButton
        android:layout_width="72dp"
        android:layout_height="72dp"
        app:cornerRadius="36dp"
        style="@style/Widget.MaterialComponents.ExtendedFloatingActionButton"
        android:textSize="32dp"
        android:textColor="@color/white"
        android:layout_margin="12dp"
        android:id="@+id/button_9"
        android:text="9" />

```

```
    android:id="@+id/button_9"
    android:text="9"/>
<com.google.android.material.button.MaterialButton
    android:layout_width="72dp"
    android:layout_height="72dp"
    app:cornerRadius="36dp"
    style="@style/Widget.MaterialComponents.ExtendedFloatingActionButton"
    android:textSize="32dp"
    android:textColor="@color/white"
    android:layout_margin="12dp"
    android:id="@+id/button_multiply"
    android:backgroundTint="#FF9800"
    android:text="*" />

</LinearLayout>
<LinearLayout
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:gravity="center"
    android:orientation="horizontal">

    <com.google.android.material.button.MaterialButton
        android:layout_width="72dp"
        android:layout_height="72dp"
        app:cornerRadius="36dp"
        style="@style/Widget.MaterialComponents.ExtendedFloatingActionButton"
        android:textSize="32dp"
        android:textColor="@color/white"
        android:layout_margin="12dp"
        android:id="@+id/button_4"
        android:text="4" />
    <com.google.android.material.button.MaterialButton
        android:layout_width="72dp"
        android:layout_height="72dp"
        app:cornerRadius="36dp"
        style="@style/Widget.MaterialComponents.ExtendedFloatingActionButton"
        android:textSize="32dp"
        android:textColor="@color/white"
        android:layout_margin="12dp"
        android:id="@+id/button_5"
        android:text="5" />
    <com.google.android.material.button.MaterialButton
        android:layout_width="72dp"
        android:layout_height="72dp"
        app:cornerRadius="36dp"
        style="@style/Widget.MaterialComponents.ExtendedFloatingActionButton"
        android:textSize="32dp"
        android:textColor="@color/white"
        android:layout_margin="12dp"
        android:id="@+id/button_6"
        android:text="6" />
    <com.google.android.material.button.MaterialButton
        android:layout_width="72dp"
        android:layout_height="72dp"
```

```
    app:cornerRadius="36dp"
    style="@style/Widget.MaterialComponents.ExtendedFloatingActionButton"
    android:textSize="32dp"
    android:textColor="@color/white"
    android:layout_margin="12dp"
    android:id="@+id/button_plus"
    android:backgroundTint="#FF9800"
    android:text="+" />

</LinearLayout>
<LinearLayout
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:gravity="center"
    android:orientation="horizontal">

    <com.google.android.material.button.MaterialButton
        android:layout_width="72dp"
        android:layout_height="72dp"
        app:cornerRadius="36dp"
        style="@style/Widget.MaterialComponents.ExtendedFloatingActionButton"
        android:textSize="32dp"
        android:textColor="@color/white"
        android:layout_margin="12dp"
        android:id="@+id/button_1"
        android:text="1" />
    <com.google.android.material.button.MaterialButton
        android:layout_width="72dp"
        android:layout_height="72dp"
        app:cornerRadius="36dp"
        style="@style/Widget.MaterialComponents.ExtendedFloatingActionButton"
        android:textSize="32dp"
        android:textColor="@color/white"
        android:layout_margin="12dp"
        android:id="@+id/button_2"
        android:text="2" />
    <com.google.android.material.button.MaterialButton
        android:layout_width="72dp"
        android:layout_height="72dp"
        app:cornerRadius="36dp"
        style="@style/Widget.MaterialComponents.ExtendedFloatingActionButton"
        android:textSize="32dp"
        android:textColor="@color/white"
        android:layout_margin="12dp"
        android:id="@+id/button_3"
        android:text="3" />
    <com.google.android.material.button.MaterialButton
        android:layout_width="72dp"
        android:layout_height="72dp"
        app:cornerRadius="36dp"
        style="@style/Widget.MaterialComponents.ExtendedFloatingActionButton"
        android:textSize="32dp"
        android:textColor="@color/white"
        android:layout_margin="12dp"
        android:id="@+id/button_4"
        android:text="4" />
</LinearLayout>
```

```
    android:id="@+id/button_minus"
    android:backgroundTint="#FF9800"
    android:text="-" />

</LinearLayout>
<LinearLayout
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:gravity="center"
    android:orientation="horizontal">

    <com.google.android.material.button.MaterialButton
        android:layout_width="72dp"
        android:layout_height="72dp"
        app:cornerRadius="36dp"
        style="@style/Widget.MaterialComponents.ExtendedFloatingActionButton"
        android:textSize="20dp"
        android:textColor="@color/white"
        android:layout_margin="12dp"
        android:id="@+id/button_ac"
        android:backgroundTint="#F44336"
        android:text="AC" />
    <com.google.android.material.button.MaterialButton
        android:layout_width="72dp"
        android:layout_height="72dp"
        app:cornerRadius="36dp"
        style="@style/Widget.MaterialComponents.ExtendedFloatingActionButton"
        android:textSize="32dp"
        android:textColor="@color/white"
        android:layout_margin="12dp"
        android:id="@+id/button_0"
        android:text="0" />
    <com.google.android.material.button.MaterialButton
        android:layout_width="72dp"
        android:layout_height="72dp"
        app:cornerRadius="36dp"
        style="@style/Widget.MaterialComponents.ExtendedFloatingActionButton"
        android:textSize="32dp"
        android:textColor="@color/white"
        android:layout_margin="12dp"
        android:id="@+id/button_dot"
        android:text"." />
    <com.google.android.material.button.MaterialButton
        android:layout_width="72dp"
        android:layout_height="72dp"
        app:cornerRadius="36dp"
        style="@style/Widget.MaterialComponents.ExtendedFloatingActionButton"
        android:textSize="32dp"
        android:textColor="@color/white"
        android:layout_margin="12dp"
        android:id="@+id/button_equals"
        android:backgroundTint="#FF9800"
        android:text="=" />
</LinearLayout>
```

```
</LinearLayout>  
</RelativeLayout>
```



MainActivity.java

```
package easy.tuto.easycalculator;  
  
import androidx.appcompat.app.AppCompatActivity;  
  
import android.os.Bundle;  
import android.view.View;  
import android.widget.TextView;  
  
import com.google.android.material.button.MaterialButton;  
import org.mozilla.javascript.Context;  
import org.mozilla.javascript.Scriptable;  
  
public class MainActivity extends AppCompatActivity implements View.OnClickListener{  
  
    TextView resultTv,solutionTv;  
    MaterialButton buttonC,buttonBrackOpen,buttonBrackClose;  
    MaterialButton buttonDivide,buttonMultiply,buttonPlus,buttonMinus,buttonEquals;  
    MaterialButton button0,button1,button2,button3,button4,button5,button6,button7,button8,button9;  
    MaterialButton buttonAC,buttonDot;  
  
    @Override  
    protected void onCreate(Bundle savedInstanceState) {  
        super.onCreate(savedInstanceState);  
        setContentView(R.layout.activity_main);  
        resultTv = findViewById(R.id.result_tv);  
        solutionTv = findViewById(R.id.solution_tv);  
  
        assignId(buttonC,R.id.button_c);  
        assignId(buttonBrackOpen,R.id.button_open_bracket);  
        assignId(buttonBrackClose,R.id.button_close_bracket);  
        assignId(buttonDivide,R.id.button_divide);  
        assignId(buttonMultiply,R.id.button_multiply);  
        assignId(buttonPlus,R.id.button_plus);  
        assignId(buttonMinus,R.id.button_minus);  
        assignId(buttonEquals,R.id.button_equals);  
        assignId(button0,R.id.button_0);  
        assignId(button1,R.id.button_1);  
        assignId(button2,R.id.button_2);  
        assignId(button3,R.id.button_3);  
        assignId(button4,R.id.button_4);  
        assignId(button5,R.id.button_5);  
        assignId(button6,R.id.button_6);  
    }  
}
```

```
assignId(button7,R.id.button_7);
assignId(button8,R.id.button_8);
assignId(button9,R.id.button_9);
assignId(buttonAC,R.id.button_ac);
assignId(buttonDot,R.id.button_dot);

}

void assignId(MaterialButton btn,int id){
    btn = findViewById(id);
    btn.setOnClickListener(this);
}

@Override
public void onClick(View view) {
    MaterialButton button =(MaterialButton) view;
    String buttonText = button.getText().toString();
    String dataToCalculate = solutionTv.getText().toString();

    if(buttonText.equals("AC")){
        solutionTv.setText("");
        resultTv.setText("0");
        return;
    }
    if(buttonText.equals("=")){
        solutionTv.setText(resultTv.getText());
        return;
    }
    if(buttonText.equals("C")){
        dataToCalculate = dataToCalculate.substring(0,dataToCalculate.length()-1);
    }else{
        dataToCalculate = dataToCalculate+buttonText;
    }
    solutionTv.setText(dataToCalculate);

    String finalResult = getResult(dataToCalculate);

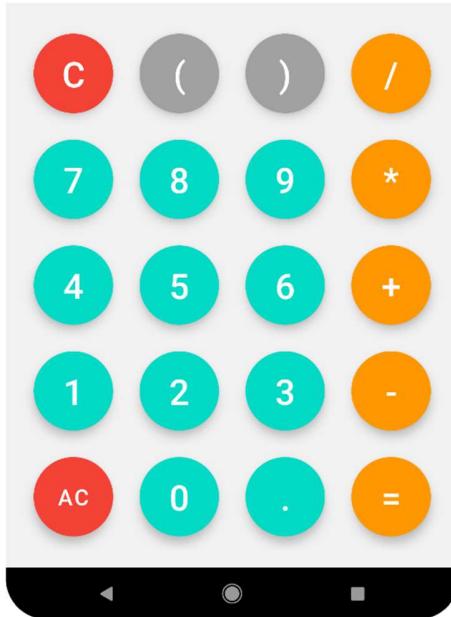
    if(!finalResult.equals("Err")){
        resultTv.setText(finalResult);
    }
}

String getResult(String data){
    try{
        Context context = Context.enter();
        context.setOptimizationLevel(-1);
        Scriptable scriptable = context.initStandardObjects();
        String finalResult = context.evaluateString(scriptable,data,"Javascript",1,null).toString();
    }
}
```

```
if(finalResult.endsWith(".0")){
    finalResult = finalResult.replace(".0","");
}
return finalResult;
}catch (Exception e){
    return "Err";
}
}
```

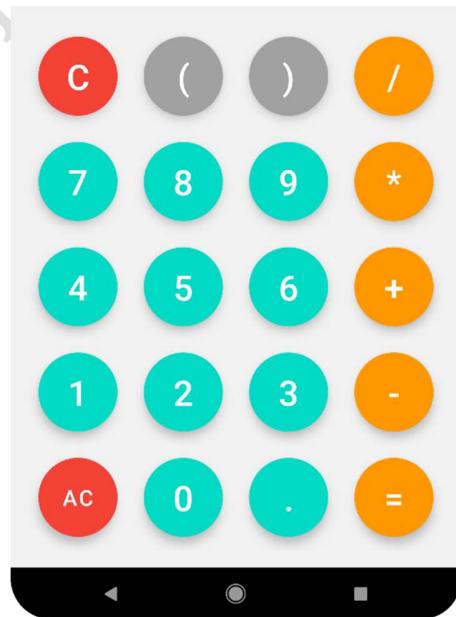
8\*9

72



8\*9/966

0.074534161  
49068323



## P11. Applying colors on Activity background (AutoCompleteTextView)



activity\_main.xml

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

    <AutoCompleteTextView
        android:id="@+id/autoCompleteTextView"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:background="@color/white"
        android:hint="Enter Color name"
        android:textColorHint="@color/black"
        android:textAlignment="center"
        android:textSize="25sp"
        app:layout_constraintBottom_toBottomOf="parent"
        app:layout_constraintTop_toTopOf="parent"
        app:layout_constraintVertical_bias="0.5" />

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



MainActivity.java

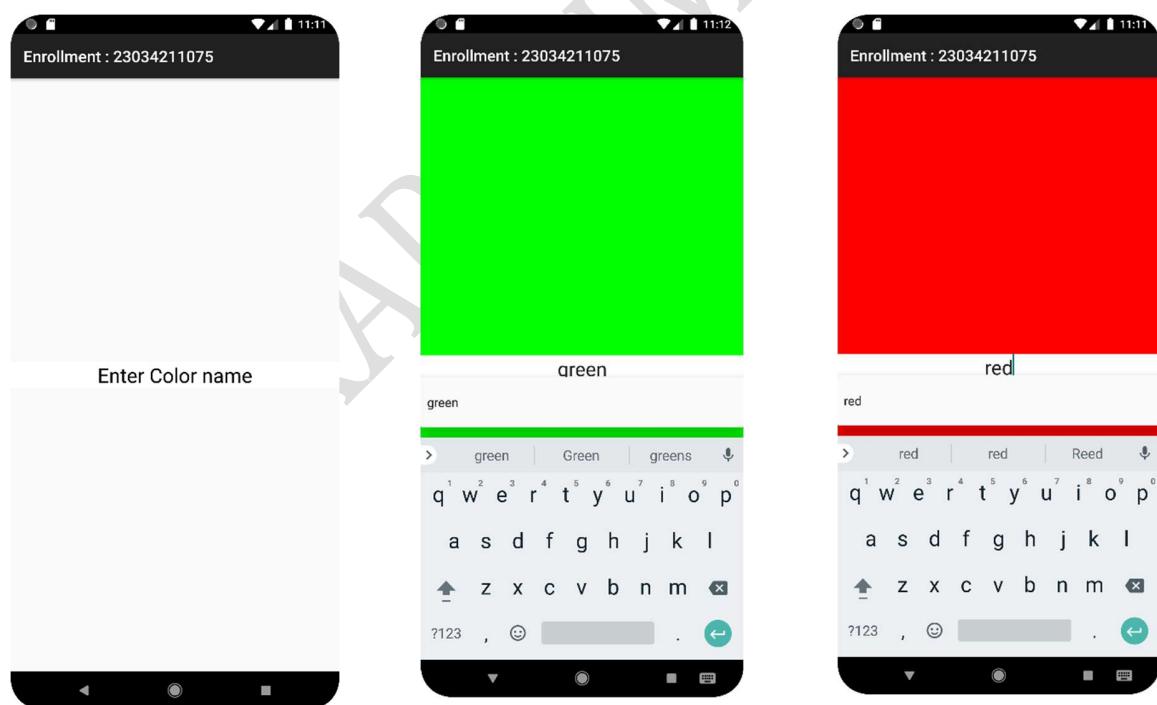
```
package com.example.practical11;

import androidx.appcompat.app.AppCompatActivity;
import androidx.constraintlayout.widget.ConstraintLayout;

import android.graphics.Color;
import android.os.Bundle;
import android.widget.ArrayAdapter;
import android.widget.AutoCompleteTextView;

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

```
actv = (AutoCompleteTextView) findViewById(R.id.autoCompleteTextView);
String[] colors = {"red", "green", "yellow"};
ArrayAdapter<String> ad = new ArrayAdapter<String>(this, android.R.layout.simple_dropdown_item_1line, colors);
actv.setAdapter(ad);
ConstraintLayout consLayout = findViewById(R.id.constraintLayout);
actv.setOnItemClickListener((parent, view, position, id) -> {
    String selectedcolor = (String) parent.getItemAtPosition(position);
    int color;
    switch(selectedcolor){
        case "red":
            color = Color.RED;
            break;
        case "green":
            color = Color.GREEN;
            break;
        case "yellow":
            color = Color.YELLOW;
            break;
        default:
            color = Color.WHITE;
    }
    consLayout.setBackgroundColor(color);
});
```



## P12. Applying colors on Activity background (MultiAutoCompleteTextView)



activity\_main.xml

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

    <MultiAutoCompleteTextView
        android:id="@+id/multi"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:hint="Enter Color names"
        android:background="@color/white"
        android:textColorHint="@color/black"
        android:textAlignment="center"
        android:textSize="25sp"
        android:completionThreshold="1"
        app:layout_constraintBottom_toBottomOf="parent"
        app:layout_constraintTop_toTopOf="parent"
        app:layout_constraintVertical_bias="0.5" />

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



MainActivity.java

```
package com.example.practical12;

import androidx.appcompat.app.AppCompatActivity;
import androidx.constraintlayout.widget.ConstraintLayout;

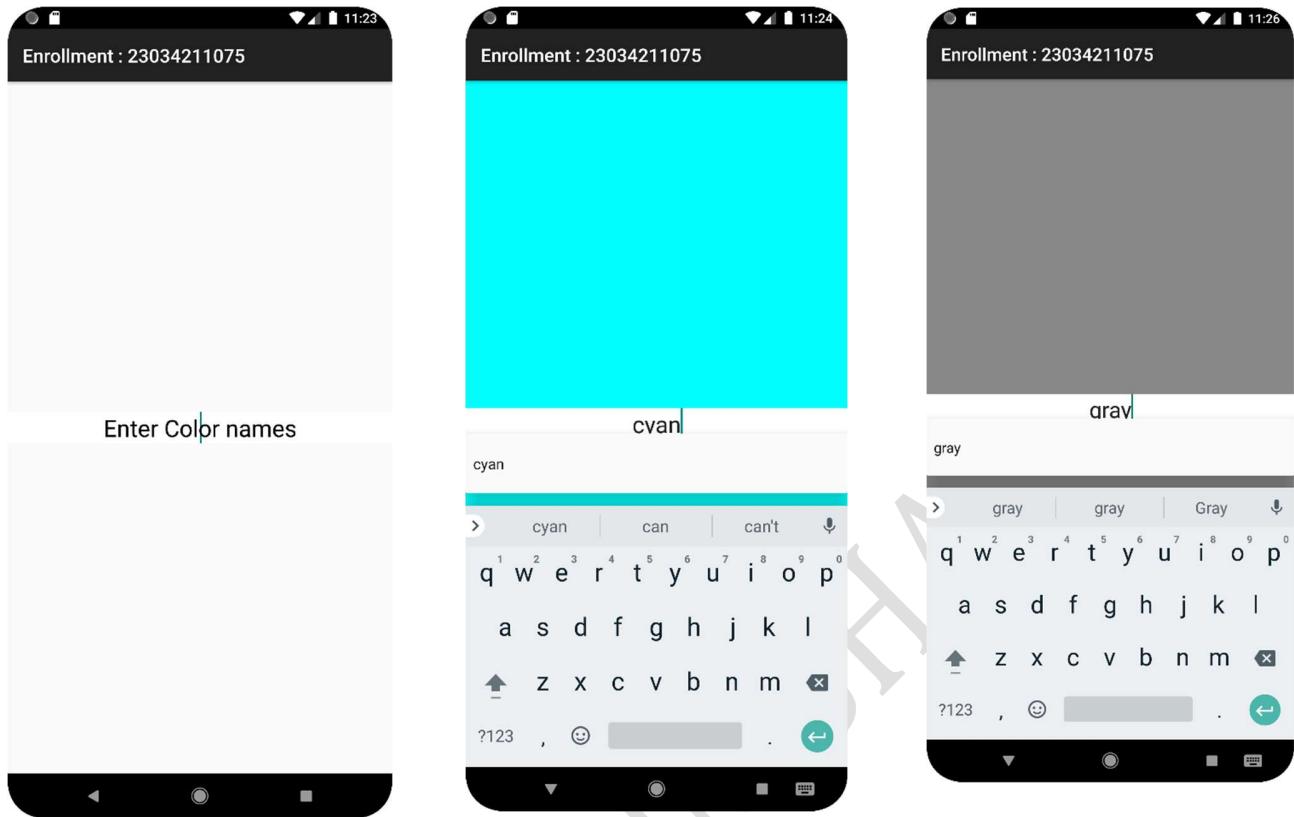
import android.graphics.Color;
import android.os.Bundle;
import android.widget.ArrayAdapter;
import android.widget.MultiAutoCompleteTextView;

public class MainActivity extends AppCompatActivity {
    MultiAutoCompleteTextView mactv;
    ConstraintLayout conslay;
    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_main);
        mactv = (MultiAutoCompleteTextView) findViewById(R.id.multi);
```

```
conslayt = findViewById(R.id.constraintLayout);
String[] colors = {"cyan", "blue", "green", "gray"};
ArrayAdapter<String> ad = new ArrayAdapter<>(this, android.R.layout.simple_dropdown_item_1line, colors);
mactv.setAdapter(ad);
mactv.setTokenizer(new MultiAutoCompleteTextView.CommaTokenizer());

mactv.setOnItemClickListener((parent, view, position, id) -> {
    String selectedColor = (String) parent.getItemAtPosition(position);
    int color;
    switch (selectedColor){
        case "cyan":
            color = Color.CYAN;
            break;
        case "blue":
            color = Color.BLUE;
            break;
        case "green":
            color = Color.GREEN;
            break;
        case "gray":
            color = Color.GRAY;
            break;
        default:
            color = Color.DKGRAY;
    }
    conslayt.setBackgroundColor(color);
});

}
}
```



### P13. Applying colors on Activity background (Spinner)

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="#E8E3E3"
    android:id="@+id/constraintlayout"
    tools:context=".MainActivity">

    <Spinner
        android:id="@+id/spinner"
        android:layout_width="409dp"
        android:layout_height="wrap_content"
        android:entries="@array/colors" />
```

```
        app:layout_constraintBottom_toBottomOf="parent"
        app:layout_constraintEnd_toEndOf="parent"
        app:layout_constraintHorizontal_bias="0.5"
        app:layout_constraintStart_toStartOf="parent"
        app:layout_constraintTop_toTopOf="parent"
        app:layout_constraintVertical_bias="0.5" />

<TextView
    android:id="@+id/textView"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:text="Select Colors"
    android:textSize="30sp"
    app:layout_constraintBottom_toTopOf="@+id/spinner"
    app:layout_constraintEnd_toEndOf="parent"
    app:layout_constraintHorizontal_bias="0.5"
    app:layout_constraintStart_toStartOf="parent"
    app:layout_constraintTop_toTopOf="parent"
    app:layout_constraintVertical_bias="0.5" />
</androidx.constraintlayout.widget.ConstraintLayout>
```



MainActivity.java

```
package com.example.practical13;

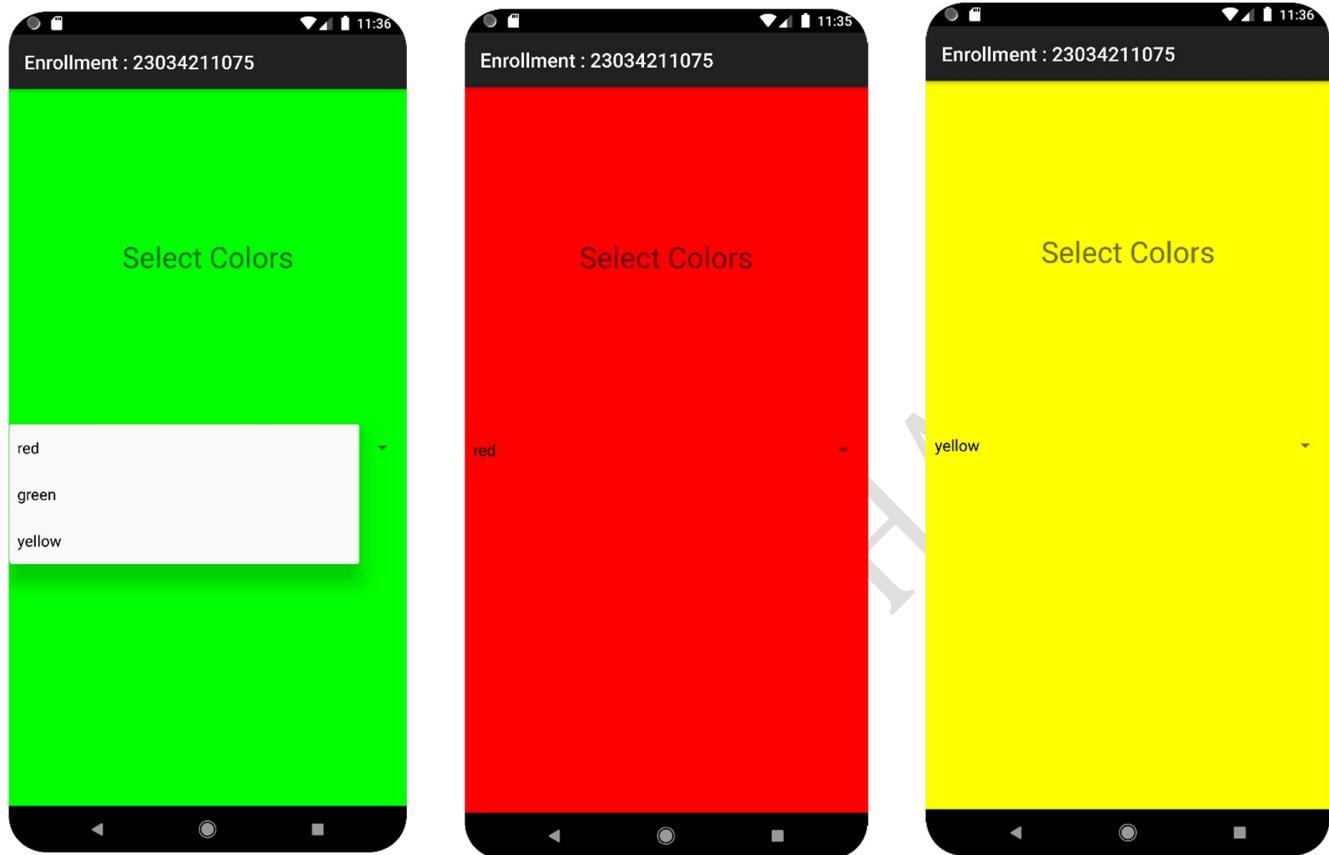
import androidx.appcompat.app.AppCompatActivity;
import androidx.constraintlayout.widget.ConstraintLayout;

import android.graphics.Color;
import android.os.Bundle;
import android.view.View;
import android.widget.AdapterView;
import android.widget.ArrayAdapter;
import android.widget.Spinner;

public class MainActivity extends AppCompatActivity {
    ConstraintLayout cns;
    Spinner sp;
    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_main);
        cns = (ConstraintLayout) findViewById(R.id.constraintlayout);
        sp = (Spinner) findViewById(R.id.spinner);
        cns.setBackgroundColor(Color.WHITE);
        ArrayAdapter<CharSequence> ad = ArrayAdapter.createFromResource(this, R.array.colors,
            android.R.layout.simple_spinner_dropdown_item);
        sp.setAdapter(ad);
```

```
sp.setOnItemSelectedListener(new AdapterView.OnItemSelectedListener() {
    @Override
    public void onItemSelected(AdapterView<?> parent, View view, int position, long id) {
        String selectedcolor = (String) parent.getItemAtPosition(position);
        switch (selectedcolor){
            case "red":
                cns.setBackgroundColor(Color.RED);
                break;
            case "green":
                cns.setBackgroundColor(Color.GREEN);
                break;
            case "yellow":
                cns.setBackgroundColor(Color.YELLOW);
                break;
        }
    }

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



#### P14. Design “ABCD” using Scroll view. (Adapter)

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">

    <androidx.recyclerview.widget.RecyclerView
```

```
    android:id="@+id/recyclerView"
    android:layout_width="match_parent"
    android:layout_height="match_parent" />

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



item\_letter.xml

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

    <TextView
        android:id="@+id/letterTextView"
        android:layout_width="match_parent"
        android:layout_height="match_parent"
        android:gravity="center"
        android:padding="8dp"
        android:textColor="@android:color/black"
        android:textSize="64sp"
        android:textStyle="bold" />

</LinearLayout>
```



activity\_letter\_adapter.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=".LetterAdapter">

    <TextView
        android:id="@+id/letterTextView"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:padding="8dp"
        android:text="A"
        android:textColor="@android:color/black"
        android:textSize="24sp"
        app:layout_constraintBottom_toBottomOf="parent"
```

```
    app:layout_constraintEnd_toEndOf="parent"
    app:layout_constraintHorizontal_bias="0.5"
    app:layout_constraintStart_toStartOf="parent"
    app:layout_constraintVertical_bias="0.5" />

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



MainActivity.java

```
package com.example.practical14;

import androidx.appcompat.app.AppCompatActivity;
import androidx.recyclerview.widget.LinearLayoutManager;
import androidx.recyclerview.widget.RecyclerView;
import android.os.Bundle;
import java.util.ArrayList;
import java.util.Arrays;
import java.util.List;

public class MainActivity extends AppCompatActivity {
    private RecyclerView recyclerView;
    private LetterAdapter mAdapter;
    private List<String> mLetters = new ArrayList<>(Arrays.asList("A", "B", "C", "D", "E", "F", "G", "H", "I", "J", "K", "L", "M", "N",
    "O", "P", "Q", "R", "S", "T", "U", "V", "W", "X", "Y", "Z"));

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

        recyclerView = findViewById(R.id.recyclerView);
        mAdapter = new LetterAdapter(this, mLetters);

        recyclerView.setLayoutManager(new LinearLayoutManager(this));
        recyclerView.setAdapter(mAdapter);
    }
}
```



LetterAdapter.java

```
package com.example.practical14;

import android.content.Context;
import android.view.LayoutInflater;
import android.view.View;
import android.view.ViewGroup;
import android.widget.TextView;
import androidx.annotation.NonNull;
import androidx.recyclerview.widget.RecyclerView;
import java.util.List;

public class LetterAdapter extends RecyclerView.Adapter<LetterAdapter.ViewHolder> {
    private Context mContext;
    private List<String> mLetters;

    public LetterAdapter(Context context, List<String> letters) {
        mContext = context;
        mLetters = letters;
    }

    @NonNull
    @Override
    public ViewHolder onCreateViewHolder(@NonNull ViewGroup parent, int viewType) {
        View view = LayoutInflater.from(mContext).inflate(R.layout.item_letter, parent, false);
        return new ViewHolder(view);
    }

    @Override
    public void onBindViewHolder(@NonNull ViewHolder holder, int position) {
        holder.bind(mLetters.get(position));
    }

    @Override
    public int getItemCount() {
        return mLetters.size();
    }

    public class ViewHolder extends RecyclerView.ViewHolder {
        TextView letterTextView;

        public ViewHolder(@NonNull View itemView) {
            super(itemView);
            letterTextView = itemView.findViewById(R.id.letterTextView);
        }

        public void bind(String letter) {
            letterTextView.setText(letter);
        }
    }
}
```

**P15. Design “Country & its flag” using Pictures controls.**

activity\_main.xml

```
<!-- activity_main.xml -->
<RelativeLayout xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:tools="http://schemas.android.com/tools"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    tools:context=".MainActivity">

    <androidx.recyclerview.widget.RecyclerView
        android:id="@+id/recyclerView"
        android:layout_width="match_parent"
        android:layout_height="match_parent" />

</RelativeLayout>
```



item\_country.xml

```
<RelativeLayout xmlns:android="http://schemas.android.com/apk/res/android"
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:padding="16dp">

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

    <ImageView
        android:id="@+id/flagImage"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_below="@id/countryName"
        android:layout_marginTop="10dp"/>

</RelativeLayout>
```



MainActivity.java

```
package com.example.practical15;

import androidx.appcompat.app.AppCompatActivity;
import androidx.recyclerview.widget.LinearLayoutManager;
import androidx.recyclerview.widget.RecyclerView;
import android.os.Bundle;
import android.widget.ImageView;
import android.widget.TextView;

public class MainActivity extends AppCompatActivity {
    TextView countryTv;
    ImageView flagImageView;
    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_main);
        RecyclerView recyclerView = findViewById(R.id.recyclerView);
        recyclerView.setLayoutManager(new LinearLayoutManager(this));
        // Sample data - country names and flag images
        String[] countryNames = {"India", "United States"};
        int[] flagImages = {R.drawable.india, R.drawable.us};

        CountryAdapter adapter = new CountryAdapter(countryNames, flagImages);
```

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



CountryAdapter.java

```
package com.example.practical15;

import android.view.LayoutInflater;
import android.view.View;
import android.view.ViewGroup;
import android.widget.ImageView;
import android.widget.TextView;

import androidx.annotation.NonNull;
import androidx.recyclerview.widget.RecyclerView;

public class CountryAdapter extends RecyclerView.Adapter<CountryAdapter.CountryViewHolder> {
    private String[] countryNames;
    private int[] flagImages;

    public CountryAdapter(String[] countryNames, int[] flagImages) {
        this.countryNames = countryNames;
        this.flagImages = flagImages;
    }

    @NonNull
    @Override
    public CountryViewHolder onCreateViewHolder(@NonNull ViewGroup parent, int viewType) {
        View view = LayoutInflater.from(parent.getContext()).inflate(R.layout.item_country, parent, false);
        return new CountryViewHolder(view);
    }

    @Override
    public void onBindViewHolder(@NonNull CountryViewHolder holder, int position) {
        holder.countryName.setText(countryNames[position]);
        holder.flagImage.setImageResource(flagImages[position]);
    }

    @Override
    public int getItemCount() {
        return countryNames.length;
    }

    static class CountryViewHolder extends RecyclerView.ViewHolder {
        TextView countryName;
        ImageView flagImage;

        CountryViewHolder(@NonNull View itemView) {

```

```
super(itemView);
countryName = itemView.findViewById(R.id.countryName);
flagImage = itemView.findViewById(R.id.flagImage);
}
}
}
```



### P16. Designing Option menu.



```
<?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">

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



## MainActivity.java

```
package com.example.practical16;

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

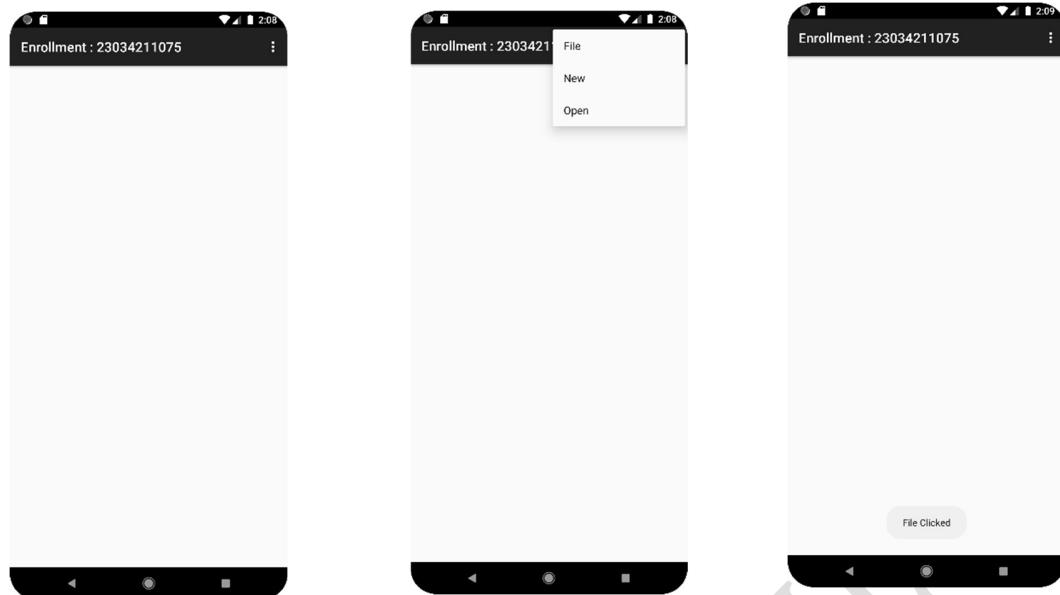
import android.os.Bundle;
import android.view.Menu;
import android.view.MenuItem;
import android.widget.Toast;

public class MainActivity extends AppCompatActivity {

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

    @Override
    public boolean onCreateOptionsMenu(Menu menu) {
        getMenuInflater().inflate(R.menu.menu, menu);
        return super.onCreateOptionsMenu(menu);
    }

    @Override
    public boolean onOptionsItemSelected(@NonNull MenuItem item) {
        int itemId = item.getItemId();
        if (itemId == R.id.mnFile) {
            Toast.makeText(this, "File Clicked", Toast.LENGTH_SHORT).show();
            return true;
        } else if (itemId == R.id.mnNew) {
            Toast.makeText(this, "New Clicked", Toast.LENGTH_SHORT).show();
            return true;
        } else if (itemId == R.id.mnOpen) {
            Toast.makeText(this, "Open Clicked", Toast.LENGTH_SHORT).show();
            return true;
        }
        return super.onOptionsItemSelected(item);
    }
}
```



### P17. Designing Context menu.

activity\_main.xml

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

    <TextView
        android:id="@+id/textView"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="Long press here for options"
        android:textSize="24sp"
        android:padding="16dp"/>

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



MainActivity.java

```
package com.example.practical17;

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

import android.os.Bundle;
import android.view.ContextMenu;
import android.view.MenuItem;
import android.view.View;
import android.widget.TextView;
import android.widget.Toast;

public class MainActivity extends AppCompatActivity {
    TextView tv;
    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_main);
        tv = (TextView) findViewById(R.id.textView);

        registerForContextMenu(tv);
    }

    @Override
    public void onCreateContextMenu(ContextMenu menu, View v, ContextMenu.ContextMenuItemInfo menuInfo) {
        super.onCreateContextMenu(menu, v, menuInfo);
        getMenuInflater().inflate(R.menu.menu, menu );
    }

    @Override
    public boolean onContextItemSelected(@NonNull MenuItem item) {
        int itemId = item.getItemId();
        if (itemId == R.id.action_edit) {
            Toast.makeText(this, "Edit clicked", Toast.LENGTH_SHORT).show();
            return true;
        } else if (itemId == R.id.action_delete) {
            Toast.makeText(this, "Delete clicked", Toast.LENGTH_SHORT).show();
            return true;
        } else if (itemId == R.id.action_paste) {
            Toast.makeText(this, "Paste clicked", Toast.LENGTH_SHORT).show();
            return true;
        }
        return super.onContextItemSelected(item);
    }
}
```



### P18. Designing Popup menu.

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="Show Popup Menu"
        app:layout_constraintBottom_toBottomOf="parent"
        app:layout_constraintEnd_toEndOf="parent"
        app:layout_constraintHorizontal_bias="0.5"
        app:layout_constraintStart_toStartOf="parent"
        app:layout_constraintTop_toTopOf="parent"
        app:layout_constraintVertical_bias="0.5" />

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



MainActivity.java

```
package com.example.practical18;

import androidx.appcompat.app.ActionBar;
import androidx.appcompat.app.AppCompatActivity;

import android.os.Bundle;
import android.view.MenuItem;
import android.view.View;
import android.widget.Button;
import android.widget.PopupMenu;
import android.widget.Toast;

public class MainActivity extends AppCompatActivity {
    Button btn;

    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_main);
        btn = (Button) findViewById(R.id.button);

        btn.setOnClickListener(new View.OnClickListener() {
            @Override
            public void onClick(View v) {
                showPopupMenu(v);
            }
        });
    }

    private void showPopupMenu(View v) {
        PopupMenu pm = new PopupMenu(this, v);
        pm.inflate(R.menu.menu);

        pm.setOnMenuItemClickListener(new PopupMenu.OnMenuItemClickListener() {
            @Override
            public boolean onMenuItemClick(MenuItem item) {
                int itemId = item.getItemId();
                return true;
            }
        });
    }
}
```

```
if (itemId == R.id.action_copy) {  
    Toast.makeText(MainActivity.this, "Copy clicked", Toast.LENGTH_SHORT).show();  
    return true;  
} else if (itemId == R.id.action_paste) {  
    Toast.makeText(MainActivity.this, "Paste clicked", Toast.LENGTH_SHORT).show();  
    return true;  
} else if (itemId == R.id.action_cancel) {  
    Toast.makeText(MainActivity.this, "Cancel clicked", Toast.LENGTH_SHORT).show();  
    return true;  
}  
}  
});  
pm.show();  
}  
}
```



### P19. Read, Write and Append Operation on 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">

    <Button
        android:id="@+id	btnRead"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:onClick="read_file"
        android:text="Read"
        android:textSize="20sp"
        app:layout_constraintBottom_toTopOf="@+id	btnWrite"
        app:layout_constraintEnd_toEndOf="parent"
        app:layout_constraintHorizontal_bias="0.5"
        app:layout_constraintStart_toStartOf="parent"
        app:layout_constraintTop_toBottomOf="@+id	edtv"
        app:layout_constraintVertical_bias="0.5" />

    <Button
        android:id="@+id	btnWrite"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:onClick="write_file"
        android:text="Write"
        android:textSize="20sp"
        app:layout_constraintBottom_toTopOf="@+id	btnAppend"
        app:layout_constraintEnd_toEndOf="parent"
        app:layout_constraintHorizontal_bias="0.5"
        app:layout_constraintStart_toStartOf="parent"
        app:layout_constraintTop_toBottomOf="@+id	btnRead"
        app:layout_constraintVertical_bias="0.5" />

    <Button
        android:id="@+id	btnAppend"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:onClick="append_file"
        android:text="Append"
        android:textSize="20sp"
        app:layout_constraintBottom_toBottomOf="parent"
        app:layout_constraintEnd_toEndOf="parent"
        app:layout_constraintHorizontal_bias="0.5"
        app:layout_constraintStart_toStartOf="parent"
        app:layout_constraintTop_toBottomOf="@+id	btnWrite"
        app:layout_constraintVertical_bias="0.5" />

    <EditText
        android:id="@+id	edtv"
        android:layout_width="365dp"
        android:layout_height="111dp"
        android:ems="10"
        android:inputType="text"
        app:layout_constraintBottom_toTopOf="@+id	btnRead"
        app:layout_constraintEnd_toEndOf="parent"
```

```
    app:layout_constraintHorizontal_bias="0.5"
    app:layout_constraintStart_toStartOf="parent"
    app:layout_constraintTop_toTopOf="parent"
    app:layout_constraintVertical_bias="0.5" />

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



MainActivity.java

```
package com.example.pra19;

import androidx.appcompat.app.AppCompatActivity;

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

import java.io.DataInputStream;
import java.io.FileInputStream;
import java.io.FileNotFoundException;
import java.io.FileOutputStream;
import java.io.IOException;

public class MainActivity extends AppCompatActivity {
    FileInputStream fis;
    FileOutputStream fos;
    String s;
    EditText edt;
    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_main);
        edt = (EditText) findViewById(R.id.edtv);
    }

    public void read_file(View view) {
        try {
            fis = openFileInput("demofile.txt");
            DataInputStream dis = new DataInputStream(fis);
            StringBuffer sb = new StringBuffer();
            String s=null;
            while((s=dis.readLine()) != null)
            {
                sb.append(s);
            }
            edt.setText(sb.toString());
            dis.close();
        }
```

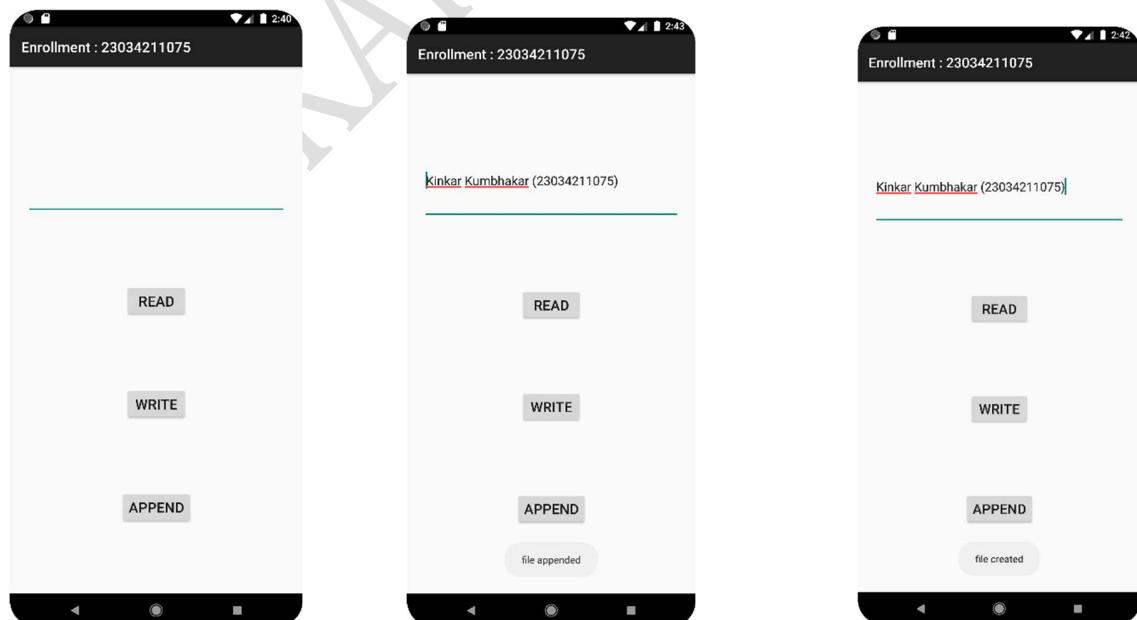
```
    fis.close();
    Toast.makeText(this, "Get Text", Toast.LENGTH_SHORT).show();
} catch (IOException e) {
    e.printStackTrace();
}

}

public void write_file(View view) {
    try {
        fos = openFileOutput("demofile.txt", MODE_PRIVATE);
        String s = edt.getText().toString();
        fos.write(s.getBytes());
        fos.close();
        Toast.makeText(this, "file created", Toast.LENGTH_SHORT).show();
    } catch (IOException e) {
        e.printStackTrace();
    }
}

public void append_file(View view) {
    try {
        fos = openFileOutput("demofile.txt", MODE_APPEND);
        String s = edt.getText().toString();
        fos.write(s.getBytes());
        fos.close();
        Toast.makeText(this, "file appended", Toast.LENGTH_SHORT).show();
    } catch (IOException e) {
        e.printStackTrace();
    }
}

}
```



## P20. Read and Write using Shared Preference



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/tvName"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="My Name"
        android:textSize="35sp"
        app:layout_constraintBottom_toTopOf="@+id(edtName)"
        app:layout_constraintEnd_toEndOf="parent"
        app:layout_constraintHorizontal_bias="0.5"
        app:layout_constraintStart_toStartOf="parent"
        app:layout_constraintTop_toTopOf="parent"
        app:layout_constraintVertical_bias="0.5" />

    <EditText
        android:id="@+id(edtName"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:ems="10"
        android:hint="Name"
        android:inputType="text"
        android:textSize="30sp"
        app:layout_constraintBottom_toTopOf="@+id/tvRoll"
        app:layout_constraintEnd_toEndOf="parent"
        app:layout_constraintHorizontal_bias="0.5"
        app:layout_constraintStart_toStartOf="parent"
        app:layout_constraintTop_toBottomOf="@+id/tvName"
        app:layout_constraintVertical_bias="0.5" />

    <TextView
        android:id="@+id/tvRoll"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="My RollNo"
        android:textSize="35sp"
        app:layout_constraintBottom_toTopOf="@+id(edtRoll"
        app:layout_constraintEnd_toEndOf="parent"
        app:layout_constraintHorizontal_bias="0.5"
        app:layout_constraintStart_toStartOf="parent"
        app:layout_constraintTop_toBottomOf="@+id(edtName"
```

```
    app:layout_constraintVertical_bias="0.5" />

<EditText
    android:id="@+id/edtRoll"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:ems="10"
    android:hint="Roll no"
    android:inputType="text"
    android:textSize="30sp"
    app:layout_constraintBottom_toTopOf="@+id/checkBox"
    app:layout_constraintEnd_toEndOf="parent"
    app:layout_constraintHorizontal_bias="0.5"
    app:layout_constraintStart_toStartOf="parent"
    app:layout_constraintTop_toBottomOf="@+id/tvRoll"
    app:layout_constraintVertical_bias="0.5" />

<Button
    android:id="@+id/button"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:onClick="store_pref"
    android:text="Click here"
    android:textSize="20sp"
    app:layout_constraintBottom_toBottomOf="parent"
    app:layout_constraintEnd_toEndOf="parent"
    app:layout_constraintHorizontal_bias="0.5"
    app:layout_constraintStart_toStartOf="parent"
    app:layout_constraintTop_toBottomOf="@+id/checkBox"
    app:layout_constraintVertical_bias="0.5" />

<CheckBox
    android:id="@+id/checkBox"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:text="Remember Me"
    android:textSize="23sp"
    app:layout_constraintBottom_toTopOf="@+id/button"
    app:layout_constraintEnd_toEndOf="parent"
    app:layout_constraintHorizontal_bias="0.5"
    app:layout_constraintStart_toStartOf="parent"
    app:layout_constraintTop_toBottomOf="@+id/edtRoll"
    app:layout_constraintVertical_bias="0.5" />
</androidx.constraintlayout.widget.ConstraintLayout>
```



MainActivity.java

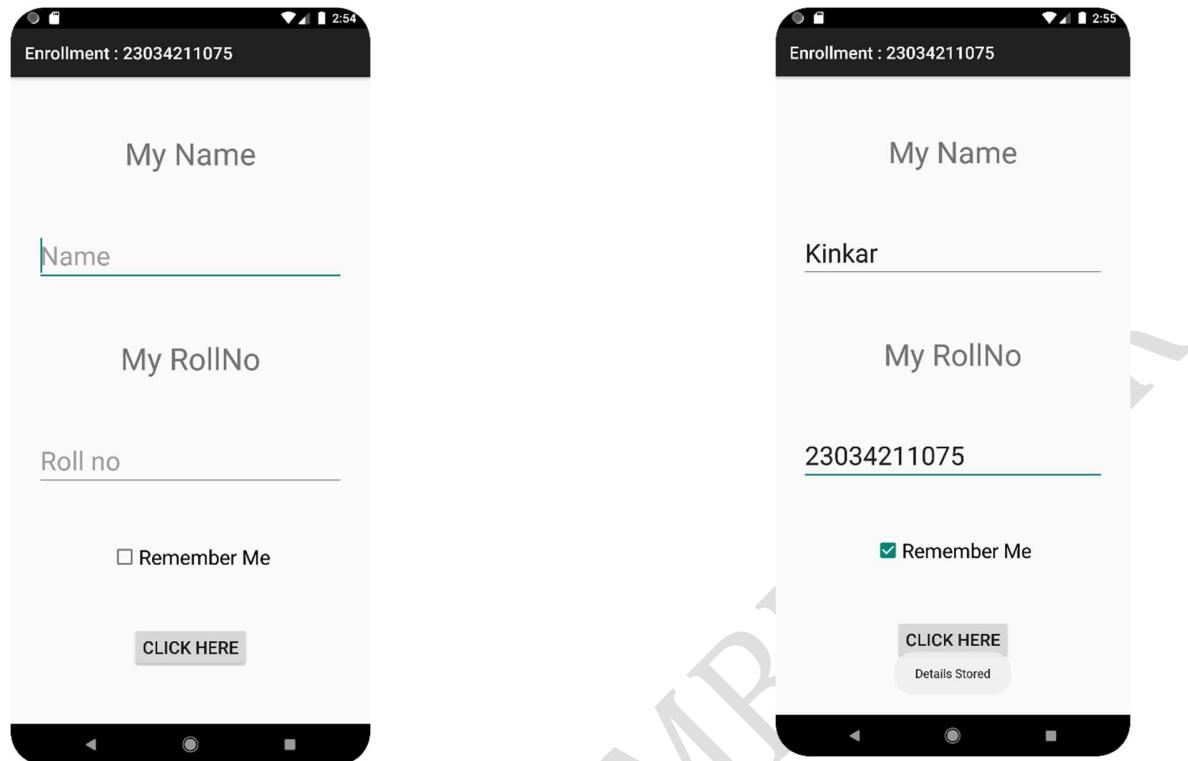
```
package com.example.practical20;
```

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

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 ed1, ed2;
    CheckBox cb;
    String details="Remember";
    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_main);
        ed1 = (EditText) findViewById(R.id.edtName);
        ed2 = (EditText) findViewById(R.id.edtRoll);
        cb = (CheckBox) findViewById(R.id.checkBox);
    }

    public void store_pref(View view) {
        boolean v = cb.isChecked();
        if(v == true)
        {
            SharedPreferences sp = getSharedPreferences(details, MODE_PRIVATE);
            SharedPreferences.Editor sed = sp.edit();
            sed.putString("Nm",ed1.getText().toString());
            sed.putString("Rn",ed2.getText().toString());
            Toast.makeText(this, "Details Stored", Toast.LENGTH_SHORT).show();
            sed.commit();
        }
        else {
            SharedPreferences sp = getSharedPreferences(details, MODE_PRIVATE);
            ed1.setText(sp.getString("Nm", ""));
            ed2.setText(sp.getString("Rn", ""));
            Toast.makeText(this, "Get Details", Toast.LENGTH_SHORT).show();
        }
    }
}
```



### P21. Perform CRUD operations on SQLite Database

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/tvName"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="Name"
        android:textColor="#130505"
        android:textSize="25sp"
        app:layout_constraintBottom_toTopOf="@+id(edtName"
        app:layout_constraintEnd_toEndOf="parent"
        app:layout_constraintHorizontal_bias="0.5"
        app:layout_constraintStart_toStartOf="parent"
        app:layout_constraintTop_toTopOf="parent" />
```

```
    app:layout_constraintVertical_bias="0.5" />

<EditText
    android:id="@+id(edtName"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:ems="10"
    android:hint="Enter Name"
    android:inputType="text"
    android:textSize="20sp"
    app:layout_constraintBottom_toTopOf="@+id/tvAddr"
    app:layout_constraintEnd_toEndOf="parent"
    app:layout_constraintHorizontal_bias="0.5"
    app:layout_constraintStart_toStartOf="parent"
    app:layout_constraintTop_toBottomOf="@+id/tvName"
    app:layout_constraintVertical_bias="0.5" />

<TextView
    android:id="@+id/tvAddr"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:text="Address"
    android:textColor="#130505"
    android:textSize="25sp"
    app:layout_constraintBottom_toTopOf="@+id(edtAddr"
    app:layout_constraintEnd_toEndOf="parent"
    app:layout_constraintHorizontal_bias="0.5"
    app:layout_constraintStart_toStartOf="parent"
    app:layout_constraintTop_toBottomOf="@+id(edtName"
    app:layout_constraintVertical_bias="0.5" />

<EditText
    android:id="@+id(edtAddr"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:ems="10"
    android:hint="Enter Address"
    android:inputType="text"
    android:textSize="20sp"
    app:layout_constraintBottom_toTopOf="@+id/btnSave"
    app:layout_constraintEnd_toEndOf="parent"
    app:layout_constraintHorizontal_bias="0.5"
    app:layout_constraintStart_toStartOf="parent"
    app:layout_constraintTop_toBottomOf="@+id/tvAddr"
    app:layout_constraintVertical_bias="0.5" />

<Button
    android:id="@+id/btnSave"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:background="#0F2DF3"
    android:text="Save"
    android:textColor="#F3EBEB"
    android:textSize="18sp"
```

```
        app:layout_constraintBottom_toTopOf="@+id(btnView"
        app:layout_constraintEnd_toEndOf="parent"
        app:layout_constraintHorizontal_bias="0.5"
        app:layout_constraintStart_toStartOf="parent"
        app:layout_constraintTop_toBottomOf="@+id(edtAddr"
        app:layout_constraintVertical_bias="0.5" />

    <Button
        android:id="@+id	btnView"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:background="#0F2DF3"
        android:text="View"
        android:textColor="#F3EBEB"
        android:textSize="18sp"
        app:layout_constraintBottom_toTopOf="@+id	btnReset"
        app:layout_constraintEnd_toEndOf="parent"
        app:layout_constraintHorizontal_bias="0.5"
        app:layout_constraintStart_toStartOf="parent"
        app:layout_constraintTop_toBottomOf="@+id	btnSave"
        app:layout_constraintVertical_bias="0.5" />

    <Button
        android:id="@+id	btnReset"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:background="#0F2DF3"
        android:text="Reset"
        android:textColor="#F3EBEB"
        android:textSize="18sp"
        app:layout_constraintBottom_toBottomOf="parent"
        app:layout_constraintEnd_toEndOf="parent"
        app:layout_constraintHorizontal_bias="0.5"
        app:layout_constraintStart_toStartOf="parent"
        app:layout_constraintTop_toBottomOf="@+id	btnView"
        app:layout_constraintVertical_bias="0.5" />
</androidx.constraintlayout.widget.ConstraintLayout>
```



activity\_view\_persons.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=".ViewPersons">
```

```
<TextView  
    android:id="@+id/textView"  
    android:layout_width="wrap_content"  
    android:layout_height="wrap_content"  
    android:text="ID"  
    android:textColor="#1E0B0B"  
    android:textSize="25sp"  
    app:layout_constraintBottom_toTopOf="@+id/editID"  
    app:layout_constraintEnd_toEndOf="parent"  
    app:layout_constraintHorizontal_bias="0.5"  
    app:layout_constraintStart_toStartOf="parent"  
    app:layout_constraintTop_toTopOf="parent"  
    app:layout_constraintVertical_bias="0.5" />  
  
<EditText  
    android:id="@+id/editID"  
    android:layout_width="409dp"  
    android:layout_height="wrap_content"  
    android:ems="10"  
    android:inputType="text"  
    app:layout_constraintBottom_toTopOf="@+id	btnShow"  
    app:layout_constraintEnd_toEndOf="parent"  
    app:layout_constraintHorizontal_bias="0.5"  
    app:layout_constraintStart_toStartOf="parent"  
    app:layout_constraintTop_toBottomOf="@+id/textView"  
    app:layout_constraintVertical_bias="0.5" />  
  
<Button  
    android:id="@+id	btnShow"  
    android:layout_width="wrap_content"  
    android:layout_height="wrap_content"  
    android:background="#95F139"  
    android:text="Show"  
    android:textSize="20sp"  
    app:layout_constraintBottom_toTopOf="@+id/textView2"  
    app:layout_constraintEnd_toEndOf="parent"  
    app:layout_constraintHorizontal_bias="0.5"  
    app:layout_constraintStart_toStartOf="parent"  
    app:layout_constraintTop_toBottomOf="@+id/editID"  
    app:layout_constraintVertical_bias="0.5" />  
  
<TextView  
    android:id="@+id/textView2"  
    android:layout_width="wrap_content"  
    android:layout_height="wrap_content"  
    android:text="Name"  
    android:textColor="#1E0B0B"  
    android:textSize="25sp"  
    app:layout_constraintBottom_toTopOf="@+id/editTextName"  
    app:layout_constraintEnd_toEndOf="parent"  
    app:layout_constraintHorizontal_bias="0.5"  
    app:layout_constraintStart_toStartOf="parent"  
    app:layout_constraintTop_toBottomOf="@+id	btnShow"
```

```
    app:layout_constraintVertical_bias="0.5" />

<EditText
    android:id="@+id/editTextName"
    android:layout_width="409dp"
    android:layout_height="wrap_content"
    android:ems="10"
    android:inputType="text"
    app:layout_constraintBottom_toTopOf="@+id/textView3"
    app:layout_constraintEnd_toEndOf="parent"
    app:layout_constraintHorizontal_bias="0.5"
    app:layout_constraintStart_toStartOf="parent"
    app:layout_constraintTop_toBottomOf="@+id/textView2"
    app:layout_constraintVertical_bias="0.5" />

<TextView
    android:id="@+id/textView3"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:text="Address"
    android:textColor="#1E0B0B"
    android:textSize="25sp"
    app:layout_constraintBottom_toTopOf="@+id/editTextAddr"
    app:layout_constraintEnd_toEndOf="parent"
    app:layout_constraintHorizontal_bias="0.5"
    app:layout_constraintStart_toStartOf="parent"
    app:layout_constraintTop_toBottomOf="@+id/editTextName"
    app:layout_constraintVertical_bias="0.5" />

<EditText
    android:id="@+id/editTextAddr"
    android:layout_width="409dp"
    android:layout_height="wrap_content"
    android:ems="10"
    android:inputType="text"
    app:layout_constraintBottom_toTopOf="@+id	btnback"
    app:layout_constraintEnd_toEndOf="parent"
    app:layout_constraintHorizontal_bias="0.5"
    app:layout_constraintStart_toStartOf="parent"
    app:layout_constraintTop_toBottomOf="@+id/textView3"
    app:layout_constraintVertical_bias="0.5" />

<Button
    android:id="@+id	btnback"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:background="@color/black"
    android:text="Back"
    android:textColor="@color/white"
    android:textSize="20sp"
    app:layout_constraintBottom_toTopOf="@+id	btnDelete"
    app:layout_constraintEnd_toEndOf="parent"
    app:layout_constraintHorizontal_bias="0.5"
    app:layout_constraintStart_toStartOf="parent"
```

```
        app:layout_constraintTop_toBottomOf="@+id/editTextAddr"
        app:layout_constraintVertical_bias="0.5" />

    <Button
        android:id="@+id	btnDelete"
        android:layout_width="139dp"
        android:layout_height="50dp"
        android:layout_marginTop="104dp"
        android:background="#F30505"
        android:onClick="removebyID"
        android:text="Delete by Id"
        android:textColor="@color/white"
        android:textSize="20sp"
        app:layout_constraintBottom_toTopOf="@+id	btnUpdate"
        app:layout_constraintEnd_toEndOf="parent"
        app:layout_constraintStart_toStartOf="parent"
        app:layout_constraintTop_toBottomOf="@+id	btnback"
        app:layout_constraintVertical_bias="0.5" />

    <Button
        android:id="@+id	btnUpdate"
        android:layout_width="154dp"
        android:layout_height="52dp"
        android:background="#F30505"
        android:onClick="UpdatebyID"
        android:text="Update by id"
        android:textColor="@color/white"
        android:textSize="20sp"
        app:layout_constraintBottom_toBottomOf="parent"
        app:layout_constraintEnd_toEndOf="parent"
        app:layout_constraintHorizontal_bias="0.5"
        app:layout_constraintStart_toStartOf="parent"
        app:layout_constraintTop_toBottomOf="@+id	btnDelete"
        app:layout_constraintVertical_bias="0.5" />
</androidx.constraintlayout.widget.ConstraintLayout>
```



MainActivity.java

```
package com.example.practical21;

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 implements View.OnClickListener {
    EditText edname, edaddr;
    Button btnsave, btnview, btnreset;
    private DatabaseHelper db;

    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_main);
        edname = (EditText) findViewById(R.id.edtName);
        edaddr = (EditText) findViewById(R.id.edtAddr);
        btnsave = (Button) findViewById(R.id.btnSave);
        btnview = (Button) findViewById(R.id.btnView);
        btnreset = (Button) findViewById(R.id.btnReset);

        db = new DatabaseHelper(this);

        btnsave.setOnClickListener(this);
        btnview.setOnClickListener(this);
        btnreset.setOnClickListener(this);
    }

    private void insert()
    {
        String name = edname.getText().toString().trim();
        String addr = edaddr.getText().toString().trim();
        db.addPerson(name,addr);
        Toast.makeText(this, "Insert Successfully", Toast.LENGTH_SHORT).show();
    }

    @Override
    public void onClick(View v) {
        if(v==btnsave)
        {
            insert();
        }
        if(v==btnview)
        {
            Intent i = new Intent(this,ViewPersons.class);
            startActivity(i);
        }
        if(v==btnreset)
        {
            edaddr.setText("");
            edname.setText("");
        }
    }
}
```



## DatabaseHelper.java

```
package com.example.practical21;

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

public class DatabaseHelper extends SQLiteOpenHelper {
    public static final String DB_NAME = "PersonsDB";
    public static final String TABLE_NAME = "Persons";
    public static final String COLUMN_ID = "id";
    public static final String COLUMN_NAME = "name";
    public static final String COLUMN_ADD = "addr";

    private static final int DB_VERSION = 1;

    public DatabaseHelper(Context context) {
        super(context, DB_NAME, null, DB_VERSION);
    }

    @Override
    public void onCreate(SQLiteDatabase sd) {
        String sql = "CREATE TABLE " + TABLE_NAME + "(" + COLUMN_ID + " INTEGER PRIMARY KEY AUTOINCREMENT, "
                + COLUMN_NAME + " TEXT, " + COLUMN_ADD + " TEXT)";
        sd.execSQL(sql);
    }

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

    public boolean addPerson(String name, String addr){
        SQLiteDatabase sd = this.getWritableDatabase();
        ContentValues cv = new ContentValues();

        cv.put(COLUMN_NAME,name);
        cv.put(COLUMN_ADD,addr);

        sd.insert(TABLE_NAME,null,cv);
        sd.close();

        return true;
    }

    public Cursor getPerson(int id){
```

```
SQLiteDatabase sd = this.getReadableDatabase();
String sql = "SELECT * FROM " +TABLE_NAME+ " WHERE " +COLUMN_ID+ "=" + id;
Cursor c = sd.rawQuery(sql,null);
return c;
}

public boolean removePersonbyID(int id) {
    SQLiteDatabase db = this.getWritableDatabase();
    db.delete(TABLE_NAME,"id=?",new String[]{String.valueOf(id)});
    return true;
}

public boolean updatePersonbyID(int id, String n, String a) {
    SQLiteDatabase db = this.getWritableDatabase();
    ContentValues cv = new ContentValues();
    cv.put(COLUMN_NAME,n);
    cv.put(COLUMN_ADD,a);
    db.update(TABLE_NAME,cv,"id=?",new String[]{String.valueOf(id)});
    return true;
}
}
```



ViewPersons.java

```
package com.example.practical21;

import androidx.appcompat.app.AppCompatActivity;

import android.annotation.SuppressLint;
import android.content.Intent;
import android.database.Cursor;
import android.os.Bundle;
import android.view.View;
import android.widget.Button;
import android.widget.EditText;
import android.widget.Toast;

public class ViewPersons extends AppCompatActivity {
    EditText edID, edName, edAddr;
    Button btnShow, btnBack;

    private DatabaseHelper db;
    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_view_persons);
        edID = (EditText) findViewById(R.id.edtID);
```

```
edName = (EditText) findViewById(R.id.edtTextName);
edAddr = (EditText) findViewById(R.id.edtTextAddr);
btnBack = (Button) findViewById(R.id.btnback);
btnShow = (Button) findViewById(R.id.btnShow);
db = new DatabaseHelper(this);
btnShow.setOnClickListener(new View.OnClickListener() {
    @Override
    public void onClick(View v) {
        String input = edID.getText().toString().trim();
        if (!input.isEmpty()) {
            showPerson(Integer.parseInt(input));
        } else {
            Toast.makeText(ViewPersons.this, "Please enter an ID", Toast.LENGTH_SHORT).show();
        }
    }
});

btnBack.setOnClickListener(new View.OnClickListener() {
    @Override
    public void onClick(View v) {
        Intent i = new Intent(ViewPersons.this, MainActivity.class);
        startActivity(i);
    }
});

@SuppressLint("Range")
private void showPerson(int i) {
    Cursor c = db.getPerson(i);
    if (c != null && c.moveToFirst()) {
        edID.setText(c.getString(c.getColumnIndex(DatabaseHelper.COLUMN_ID)));
        edName.setText(c.getString(c.getColumnIndex(DatabaseHelper.COLUMN_NAME)));
        edAddr.setText(c.getString(c.getColumnIndex(DatabaseHelper.COLUMN_ADD)));
    } else {
        Toast.makeText(this, "No Record Found", Toast.LENGTH_SHORT).show();
        edID.setText("");
        edName.setText("");
        edAddr.setText("");
    }
}

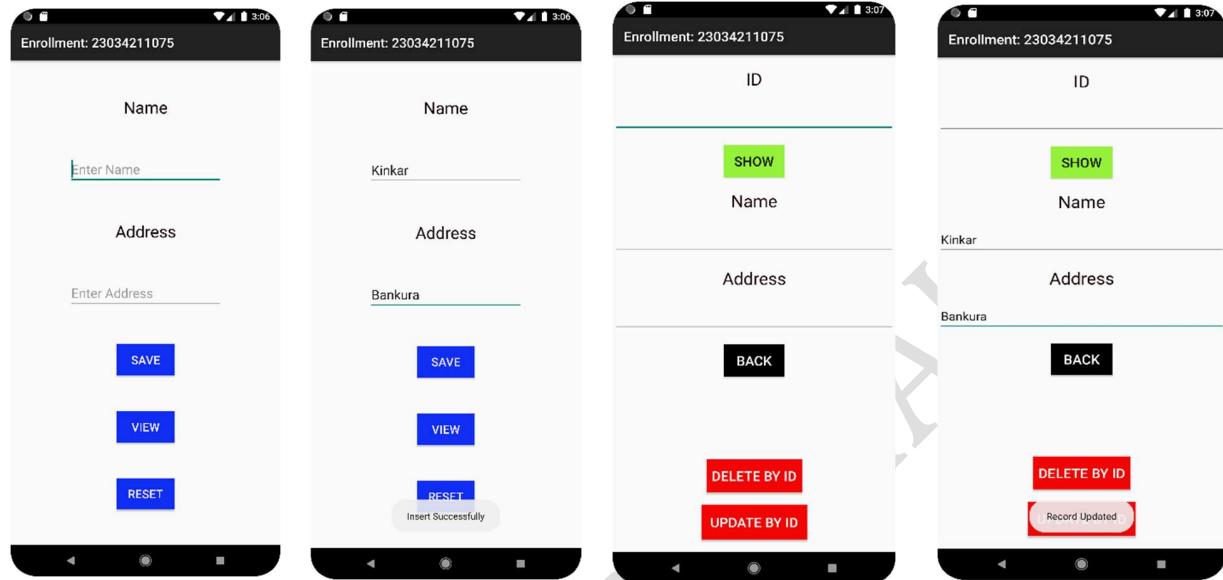
public void removebyID(View view) {
    int id = Integer.parseInt(edID.getText().toString());
    db.removePersonbyID(id);
    Toast.makeText(this, "Record Deleted", Toast.LENGTH_SHORT).show();
    edID.setText("");
}

public void UpdatebyID(View view) {
    int id = Integer.parseInt(edID.getText().toString());
    String n = edName.getText().toString();
    String a = edAddr.getText().toString();
    db.updatePersonbyID(id,n,a);
    Toast.makeText(this, "Record Updated", Toast.LENGTH_SHORT).show();
}
```

```

        edID.setText("");
    }
}

```



## P22. Study of Firebase or Cloud Database (Latest)

activity\_main.xml

```

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

    <TextView
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="Hello World!"
        app:layout_constraintBottom_toBottomOf="parent"
        app:layout_constraintEnd_toEndOf="parent"
        app:layout_constraintStart_toStartOf="parent"
        app:layout_constraintTop_toTopOf="parent" />

</androidx.constraintlayout.widget.ConstraintLayout>

```



## MainActivity.java

```
package com.example.firebaseioexample;

import static android.content.ContentValues.TAG;
import android.os.Bundle;
import android.util.Log;
import androidx.activity.EdgeToEdge;
import androidx.annotation.NonNull;
import androidx.appcompat.app.AppCompatActivity;
import androidx.core.graphics.Insets;
import androidx.core.view.ViewCompat;
import androidx.core.view.WindowInsetsCompat;
import com.google.firebase.database.DataSnapshot;
import com.google.firebase.database.DatabaseError;
import com.google.firebase.database.DatabaseReference;
import com.google.firebase.database.FirebaseDatabase;
import com.google.firebase.database.ValueEventListener;

public class MainActivity extends AppCompatActivity {

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

        // Write a message to the database
        FirebaseDatabase database = FirebaseDatabase.getInstance();

        DatabaseReference myRef_id = database.getReference("id");
        myRef_id.setValue(75);

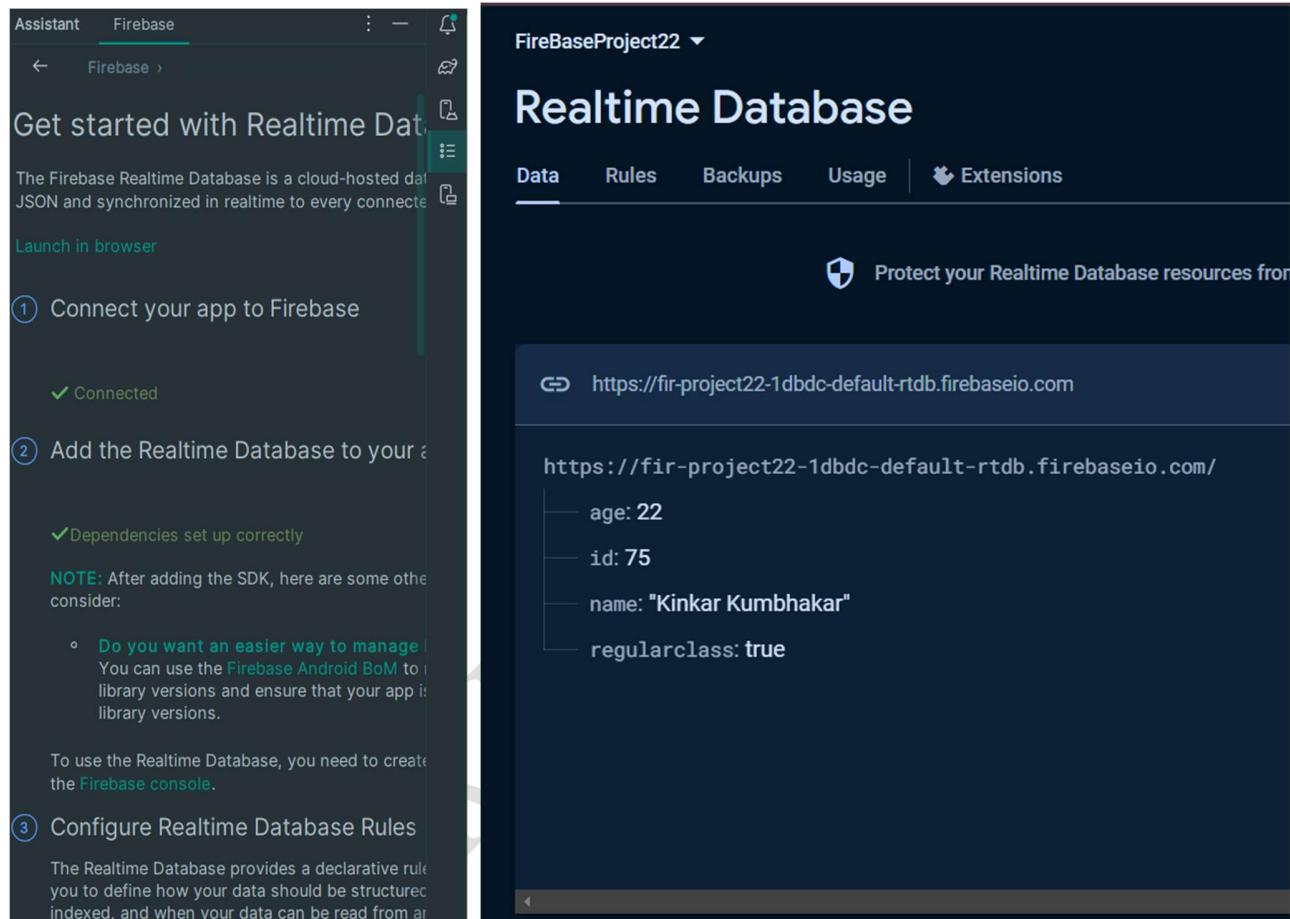
        DatabaseReference myRef_name = database.getReference("name");
        myRef_name.setValue("Kinkar Kumbhakar");
        DatabaseReference myRef_age = database.getReference("age");
        myRef_age.setValue(22);
        DatabaseReference myRef_class = database.getReference("regularclass");
        myRef_class.setValue(true);

        // Read from the database
        myRef_id.addValueEventListener(new ValueEventListener() {
            @Override
            public void onDataChange(DataSnapshot dataSnapshot) {
                // This method is called once with the initial value and again
                // whenever data at this location is updated.
                Long value = dataSnapshot.getValue(Long.class);
                Log.d(TAG, "Value is: " + value);
            }
        });
    }
}
```

```
@Override
public void onCancelled(DatabaseError error) {
    // Failed to read value
    Log.w(TAG, "Failed to read value.", error.toException());
}
});
// Read from the database
myRef_name.addValueEventListener(new ValueEventListener() {
    @Override
    public void onDataChange(DataSnapshot dataSnapshot) {
        // This method is called once with the initial value and again
        // whenever data at this location is updated.
        String value = dataSnapshot.getValue(String.class);
        Log.d(TAG, "Value is: " + value);
    }
    @Override
    public void onCancelled(DatabaseError error) {
        // Failed to read value
        Log.w(TAG, "Failed to read value.", error.toException());
    }
});
// Read from the database
myRef_age.addValueEventListener(new ValueEventListener() {
    @Override
    public void onDataChange(DataSnapshot dataSnapshot) {
        // This method is called once with the initial value and again
        // whenever data at this location is updated.
        Long value = dataSnapshot.getValue(Long.class);
        Log.d(TAG, "Value is: " + value);
    }
    @Override
    public void onCancelled(DatabaseError error) {
        // Failed to read value
        Log.w(TAG, "Failed to read value.", error.toException());
    }
});
// Read from the database
myRef_class.addValueEventListener(new ValueEventListener() {
    @Override
    public void onDataChange(DataSnapshot dataSnapshot) {
        // This method is called once with the initial value and again
        // whenever data at this location is updated.
        boolean value = dataSnapshot.getValue(boolean.class);
        Log.d(TAG, "Value is: " + value);
    }
    @Override
    public void onCancelled(DatabaseError error) {
        // Failed to read value
        Log.w(TAG, "Failed to read value.", error.toException());
    }
});
}

ViewCompat.setOnApplyWindowInsetsListener(findViewById(android.R.id.content), (v, insets) -> {
    Insets systemBars = insets.getInsets(WindowInsetsCompat.Type.systemBars());
```

```
v.setPadding(systemBars.left, systemBars.top, systemBars.right, systemBars.bottom);  
return insets;  
});  
}  
}
```



The screenshot shows two side-by-side views of the Firebase Realtime Database. On the left is the Firebase setup wizard, and on the right is the database console.

**Firebase Setup Wizard:**

- Step 1: Connect your app to Firebase → Connected
- Step 2: Add the Realtime Database to your app → Dependencies set up correctly
- Step 3: Configure Realtime Database Rules → The Realtime Database provides a declarative rule language for you to define how your data should be structured, indexed, and when your data can be read from and written to.

**Realtime Database Console:**

The database URL is <https://fir-project22-1dbdc-default-rtbd.firebaseio.com/>. The data structure shown is:

```
age: 22  
id: 75  
name: "Kinkar Kumbhakar"  
regularclass: true
```