Assignment 1

Tittle: Display Hello World on Screen

Name: Tejas Sunil Mate

Class: MCA-III

Roll No: MC232538

Date:

Remark:

Practical No. 1: Develop a program to display Hello World on screen.

Solution:

1. **activity_main.xml**

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



    <TextView
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="Hello World!"
        android:textColor="@color/red"
        android:textSize="50sp"
        app:layout_constraintBottom_toBottomOf="parent"
        app:layout_constraintEnd_toEndOf="parent"
        app:layout_constraintStart_toStartOf="parent"
```

```
  app:layout_constraintTop_toTopOf="parent" />
</androidx.constraintlayout.widget.ConstraintLayout>
```

Output:

Assignment 2

Tittle: Implement Linear and Relative Layout

Name: Tejas Sunil Mate

Class: MCA-III

Roll No: MC232538

Date:

Remark:

Practical No. 2: Develop a program to implement linear layout and Relative layout.

Solution:

1. **Linear Layout:**
   - **A. activity_main.xml**

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

   <Button
      android:id="@+id/btnStartService"
      android:layout_width="300dp"
      android:layout_height="wrap_content"
      android:text="@string/start_service"
      android:textColorLink="#11E9F1"
      android:textSize="30sp" />

   <Button android:id="@+id/btnPauseService"
      android:layout_width="300dp"
      android:layout_height="wrap_content"
      android:textSize="30sp"
      android:text="@string/pause_service"/>
```

```
<Button android:id="@+id/btnStopService"
    android:layout_width="300dp"
    android:layout_height="wrap_content"
    android:textSize="30sp"
    android:text="@string/stop_service"/>
```

B. MainActivity.java

```
package com.example.linearlayout;
import android.os.Bundle;
import android.app.Activity;
public class MainActivity extends Activity {
    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_main);
    }
}
```

Output:

2. **Relative Layout**

   A. **activity_main.xml**

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

   <Button
      android:id="@+id/button1"
      android:layout_width="200dp"
      android:layout_height="100dp"
      android:text="@string/top_left_button"
      android:layout_alignParentStart="true"
      android:layout_alignParentTop="true"
      tools:ignore="RelativeOverlap" />

   <Button
      android:id="@+id/button2"
      android:layout_width="200dp"
      android:layout_height="100dp"
      android:text="@string/top_right_button"
      android:layout_alignParentTop="true"
      android:layout_alignParentEnd="true"
      tools:ignore="RelativeOverlap" />

   <Button
      android:id="@+id/button3"
      android:layout_width="200dp"
      android:layout_height="100dp"
```

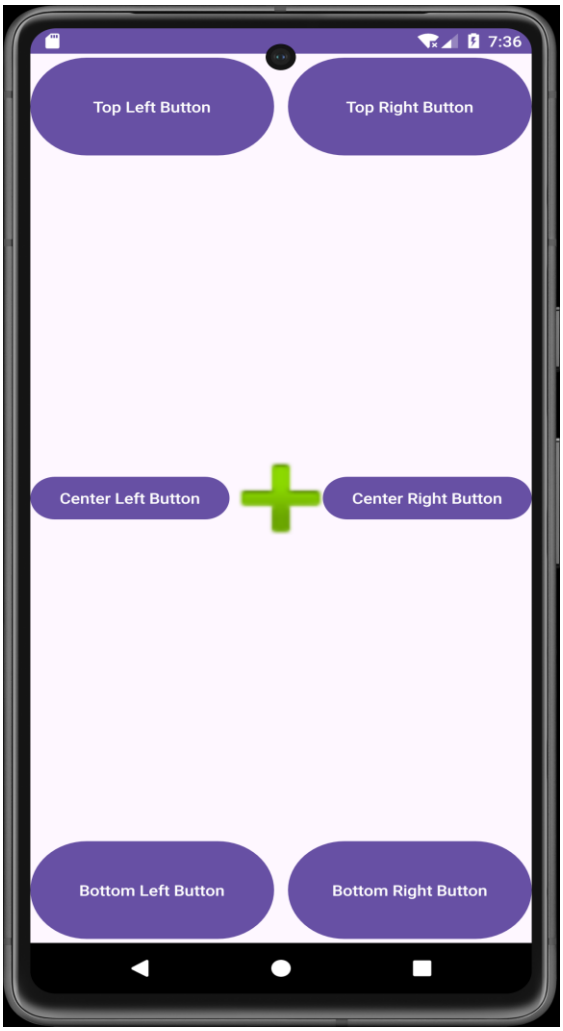
```
    android:text="@string/bottom_left_button"
    android:layout_alignParentStart="true"
    android:layout_alignParentBottom="true"/>

<Button
    android:id="@+id/button4"
    android:layout_width="200dp"
    android:layout_height="100dp"
    android:text="@string/bottom_right_button"
    android:layout_alignParentEnd="true"
    android:layout_alignParentBottom="true"
    tools:ignore="RelativeOverlap" />

<Button
    android:id="@+id/button6"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:text="@string/center_right_button"
    android:layout_centerVertical="true"
    android:layout_alignParentEnd="true"/>

<Button
    android:id="@+id/button7"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:layout_centerVertical="true"
    android:text="@string/center_left_button" />

<ImageView
    android:id="@+id/imageView"
    android:layout_width="120dp"
```

```
        android:layout_height="120dp"
        android:layout_centerHorizontal="true"
        android:layout_centerVertical="true"
        app:srcCompat="@android:drawable/ic_input_add" />
</RelativeLayout>
```

B. MainActivity.java

```
package com.example.relativelayout;

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

public class MainActivity extends AppCompatActivity {

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

Output:


7:36
Top Left Button
Top Right Button
Center Left Button
Center Right Button
Bottom Left Button
Bottom Right Button

Assignment 3

Tittle: Implement frame, table and Relative Layout

Name: Tejas Sunil Mate

Class: MCA-III

Roll No: MC232538

Date:

Remark:

Practical No. 3: Develop a program to implement frame layout, table layout and relative layout.

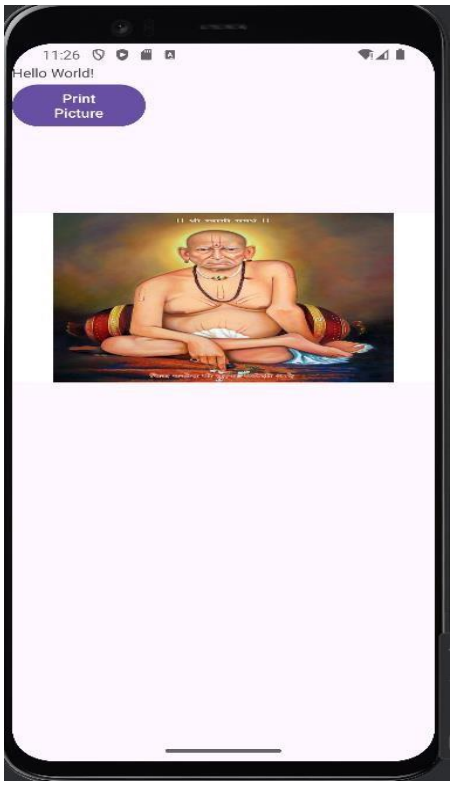
Solution:

1. **Frame Layout:**
   a. **activity_main.xml**

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

<TextView
android:layout_width="wrap_content" android:layout_height="wrap_content"
android:id="@+id/lblComments" android:text="Hello World!"
android:layout_alignParentTop="true" android:layout_alignParentStart="true"/>

<FrameLayout
android:layout_width="wrap_content" android:layout_height="wrap_content"
android:layout_alignStart="@+id/lblComments"
android:layout_below="@+id/lblComments"
android:layout_centerHorizontal="true">
<ImageView
```

android:layout_width="wrap_content" android:layout_height="wrap_content"
android:src="@drawable/swami"/>
<Button
android:layout_width="124dp" android:layout_height="wrap_content"
android:text="Print Picture"/>
</FrameLayout>
</RelativeLayout>

Output:



2. **Table Layout:**
   a. **activity_main.xml**

<?xml version="1.0" encoding="utf-8"?>
<TableLayout xmlns:android="http://schemas.android.com/apk/res/android"
xmlns:app="http://schemas.android.com/apk/res-auto"
xmlns:tools="http://schemas.android.com/tools" android:id="@+id/main"

```
android:layout_width="match_parent" android:layout_height="match_parent"
tools:context=".MainActivity" android:paddingLeft="10dp"
android:paddingRight="10dp">


<TableRow
android:layout_marginTop="30dp">
<TextView
android:layout_width="match_parent" android:layout_height="50dp"
android:background="@color/purple" android:text="Row1"
android:layout_weight="1" android:textAlignment="center" android:padding="10dp"
android:layout_margin="10dp"/>
</TableRow>

<TableRow
android:layout_marginTop="3dp">
<TextView
android:layout_width="match_parent" android:layout_height="55dp"
android:background="@color/purple" android:text="Row2\nCol1"
android:layout_weight="1" android:textAlignment="center" android:padding="10dp"
android:layout_margin="10dp"/>

<TextView
android:layout_width="match_parent" android:layout_height="55dp"
android:background="@color/purple" android:text="Row3\nCol2"
android:layout_weight="2" android:textAlignment="center" android:padding="10dp"
android:layout_margin="10dp"/>
</TableRow>
<TableRow
android:layout_marginTop="10dp">
<TextView
```

```
android:layout_width="match_parent" android:layout_height="55dp"
android:background="@color/purple" android:text="Row4\nCol3"
android:layout_weight="1"

android:textAlignment="center" android:padding="10dp"
android:layout_margin="10dp"/>
<TextView
android:layout_width="match_parent" android:layout_height="55dp"
android:background="@color/purple" android:text="Row5\nCol4"
android:layout_weight="1" android:textAlignment="center" android:padding="10dp"
android:layout_margin="10dp"/>
<TextView
android:layout_width="match_parent" android:layout_height="55dp"
android:background="@color/purple" android:text="Row6\nCol5"
android:layout_weight="1" android:textAlignment="center" android:padding="10dp"
android:layout_margin="10dp"/>
</TableRow>
<TableRow
android:layout_marginTop="30dp">
<TextView
android:layout_width="match_parent" android:layout_height="50dp"
android:background="@color/purple" android:text="Row4"
android:layout_weight="1" android:textAlignment="center" android:padding="10dp"
android:layout_margin="10dp"/>
</TableRow>  </TableLayout>
```

Output:


12:15
3G
Row1
Row2
Col1
Row3
Col2
Row4
Col3
Row5
Col4
Row6
Col5
Row4

Assignment 4

Tittle: User interface for login window.

Name: Tejas Sunil Mate

Class: MCA-III

Roll No: MC232538

Date:

Remark:

Practical No. 4: Develop a program to design User interface for login window.

Solution:

1. **MainActivity.java**

```
package com.example.loginwindow;

import android.os.Bundle;
import androidx.activity.EdgeToEdge;
import androidx.appcompat.app.AppCompatActivity;

public class MainActivity extends AppCompatActivity {

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

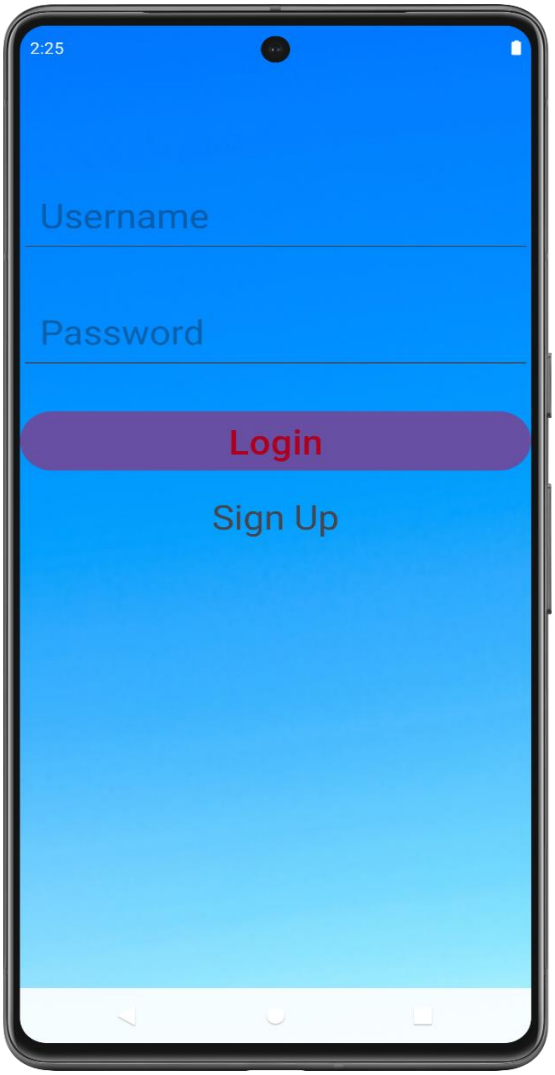
2. **activity_main.xml**

```
<?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"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    android:background="@drawable/imagapp">
```

```
<EditText
    android:id="@+id/email_input"
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:layout_marginTop="130dp"
    android:textSize="30sp"
    android:hint="Username"
    android:inputType="textEmailAddress"
    android:padding="16dp" />

<EditText
    android:id="@+id/password_input"
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:layout_below="@id/email_input"
    android:layout_marginTop="30dp"
    android:textSize="30sp"
    android:hint="Password"
    android:inputType="textPassword"
    android:padding="16dp" />

<Button
    android:id="@+id/login_button"
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:layout_below="@id/password_input"
    android:layout_marginTop="30dp"
    android:text="Login"
    android:textColor="@color/design_default_color_error"
    android:textColorLink="#D1135E"
    android:textSize="30sp" />

<!-- Sign Up Link -->
<TextView
    android:id="@+id/signup_link"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:layout_centerHorizontal="true"
```

```
        android:layout_below="@id/login_button"
        android:layout_marginTop="16dp"
        android:text="Sign Up"
        android:textSize="30sp"
        android:textAlignment="center"/>
</RelativeLayout>
```

Output:



Assignment 5

Tittle: Perform four arithmetic operation like Addition, Subtraction, Multiplication and Division.

Name: Tejas Sunil Mate

Class: MCA-III

Roll No: MC232538

Date:

Remark:

Practical No. 5: Develop a program to perform four arithmetic operation like Addition, Subtraction, Multiplication and Division.

Solution:

1. **MainActivity.java**

```
package com.example.arithematicoperation;

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

public class MainActivity extends AppCompatActivity {

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

        final EditText num1EditText = findViewById(R.id.num1);
        final EditText num2EditText = findViewById(R.id.num2);
        final TextView resultTextView = findViewById(R.id.result_text);

        Button addButton = findViewById(R.id.add_button);
        Button subtractButton = findViewById(R.id.subtract_button);
        Button multiplyButton = findViewById(R.id.multiply_button);
        Button divideButton = findViewById(R.id.divide_button);
```

```
    Button clearButton = findViewById(R.id.clear_button);

    addButton.setOnClickListener(new View.OnClickListener() {
        @Override
        public void onClick(View v) {
            performOperation(Operation.ADD);
        }
    });

    subtractButton.setOnClickListener(new View.OnClickListener() {
        @Override
        public void onClick(View v) {
            performOperation(Operation.SUBTRACT);
        }
    });

    multiplyButton.setOnClickListener(new View.OnClickListener() {
        @Override
        public void onClick(View v) {
            performOperation(Operation.MULTIPLY);
        }
    });

    divideButton.setOnClickListener(new View.OnClickListener() {
        @Override
        public void onClick(View v) {
            performOperation(Operation.DIVIDE);
        }
    });

    clearButton.setOnClickListener(new View.OnClickListener() {
        @Override
        public void onClick(View v) {
            num1EditText.setText("");
            num2EditText.setText("");
            resultTextView.setText("Result will appear here");
        }
    });
}
```

```
private void performOperation(Operation operation) {
    EditText num1EditText = findViewById(R.id.num1);
    EditText num2EditText = findViewById(R.id.num2);
    TextView resultTextView = findViewById(R.id.result_text);

    String num1String = num1EditText.getText().toString();
    String num2String = num2EditText.getText().toString();

    if (num1String.isEmpty() || num2String.isEmpty()) {
        resultTextView.setText("Please enter both numbers.");
        return;
    }

    double num1 = Double.parseDouble(num1String);
    double num2 = Double.parseDouble(num2String);
    double result;

    switch (operation) {
        case ADD:
            result = num1 + num2;
            resultTextView.setText("Result: " + result);
            break;
        case SUBTRACT:
            result = num1 - num2;
            resultTextView.setText("Result: " + result);
            break;
        case MULTIPLY:
            result = num1 * num2;
            resultTextView.setText("Result: " + result);
            break;
        case DIVIDE:
            if (num2 == 0) {
                resultTextView.setText("Error: Division by zero is not allowed.");
            } else {
                result = num1 / num2;
                resultTextView.setText("Result: " + result);
            }
            break;
    }
}
```

```
    private enum Operation {
        ADD, SUBTRACT, MULTIPLY, DIVIDE
    }
}
```

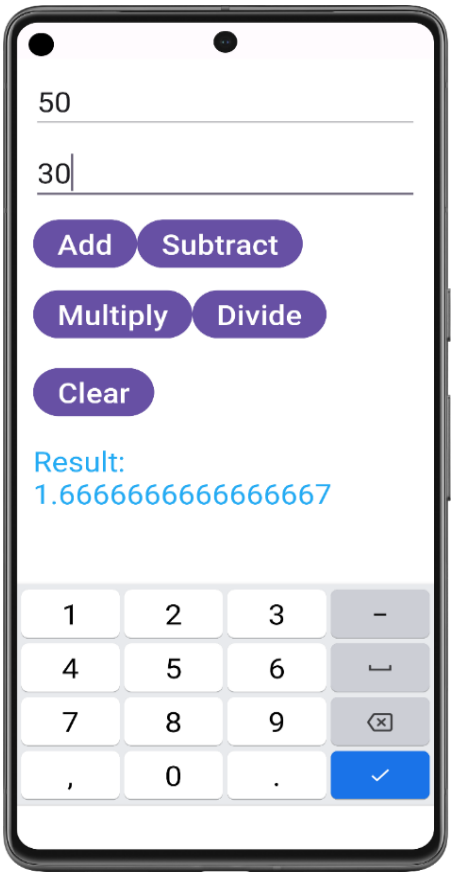
2. activity_main.xml

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

    <!-- First number input -->
    <EditText
        android:id="@+id/num1"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:hint="Enter first number"
        android:textSize="30sp"
        android:inputType="numberDecimal" />

    <!-- Second number input -->
    <EditText
        android:id="@+id/num2"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:layout_below="@id/num1"
        android:layout_marginTop="16dp"
        android:textSize="30sp"
        android:hint="Enter second number"
        android:inputType="numberDecimal" />

    <!-- Addition button -->
    <Button
        android:id="@+id/add_button"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_below="@id/num2"
```

```
    android:textSize="30sp"
    android:layout_marginTop="16dp"
    android:text="Add"
    android:layout_alignParentStart="true"/>

<!-- Subtraction button -->
<Button
    android:id="@+id/subtract_button"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:layout_below="@id/num2"
    android:layout_marginTop="16dp"
    android:textSize="30sp"
    android:layout_toEndOf="@id/add_button"
    android:text="Subtract" />

<!-- Multiplication button -->
<Button
    android:id="@+id/multiply_button"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:layout_below="@id/add_button"
    android:layout_marginTop="16dp"
    android:textSize="30sp"
    android:text="Multiply" />

<!-- Division button -->
<Button
    android:id="@+id/divide_button"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:layout_below="@id/subtract_button"
    android:layout_toEndOf="@id/multiply_button"
    android:layout_marginTop="16dp"
    android:textSize="30sp"
    android:text="Divide" />

<!-- Clear button -->
<Button
    android:id="@+id/clear_button"
```

```
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_below="@id/multiply_button"
        android:layout_marginTop="24dp"
        android:textSize="30sp"
        android:text="Clear"
        android:layout_alignParentStart="true" />

    <TextView
        android:id="@+id/result_text"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_below="@id/clear_button"
        android:layout_marginTop="24dp"
        android:textSize="30sp"
        android:textColor="@color/blue"
        android:text="Result will appear here" />
</RelativeLayout>
```

Output:



Assignment 6

Tittle: Program to implicit intent and explicit intent.

Name: Tejas Sunil Mate

Class: MCA-III

Roll No: MC232538

Date:

Remark:

Practical No. 6: Develop a program to implicit intent and explicit intent.

Solution:-

1. **MainActivity.java**

```
package com.example.intentactivityexample;

import android.content.Intent;
import android.net.Uri;
import android.os.Bundle;
import android.view.View;
import android.widget.Button;

import androidx.appcompat.app.AppCompatActivity;

public class MainActivity extends AppCompatActivity {


    Button explicit_btn, implicit_btn;
    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_main);
```

```
        explicit_btn = (Button)findViewById(R.id.explicit_Intent);
        implicit_btn = (Button) findViewById(R.id.implicit_Intent);

        explicit_btn.setOnClickListener(new View.OnClickListener() {
            @Override
            public void onClick(View v) {

                Intent intent = new  Intent(getBaseContext(), SecondActivity.class);
                startActivity(intent);
            }
        });

        implicit_btn.setOnClickListener(new View.OnClickListener() {
            @Override
            public void onClick(View v) {
                Intent intent = new Intent(Intent.ACTION_VIEW);
                intent.setData(Uri.parse("https://www.youtube.com"));
                startActivity(intent);
            }
        });
    }
}
```

2. **SecondActivity.java**

```
package com.example.intentactivityexample;

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

```
public class SecondActivity extends AppCompatActivity {

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

        Toast.makeText(getApplicationContext(), "We are moved to second
Activity",Toast.LENGTH_LONG).show();
    }
}
```

3. **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"
android:paddingLeft="@dimen/activity_horizontal_margin"
    android:paddingRight="@dimen/activity_horizontal_margin"
    android:paddingTop="@dimen/activity_vertical_margin"
    android:paddingBottom="@dimen/activity_vertical_margin"
tools:context=".MainActivity">

    <TextView
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:textAppearance="?android:attr/textAppearanceMedium"

android:text="@string/if_you_click_on_explicit_example_we_will_navigate_to_second_
activity_within_app_and_if_you_click_on_implicit_example_abhiandroid_homepage_wi
ll_open_in_browser"
```

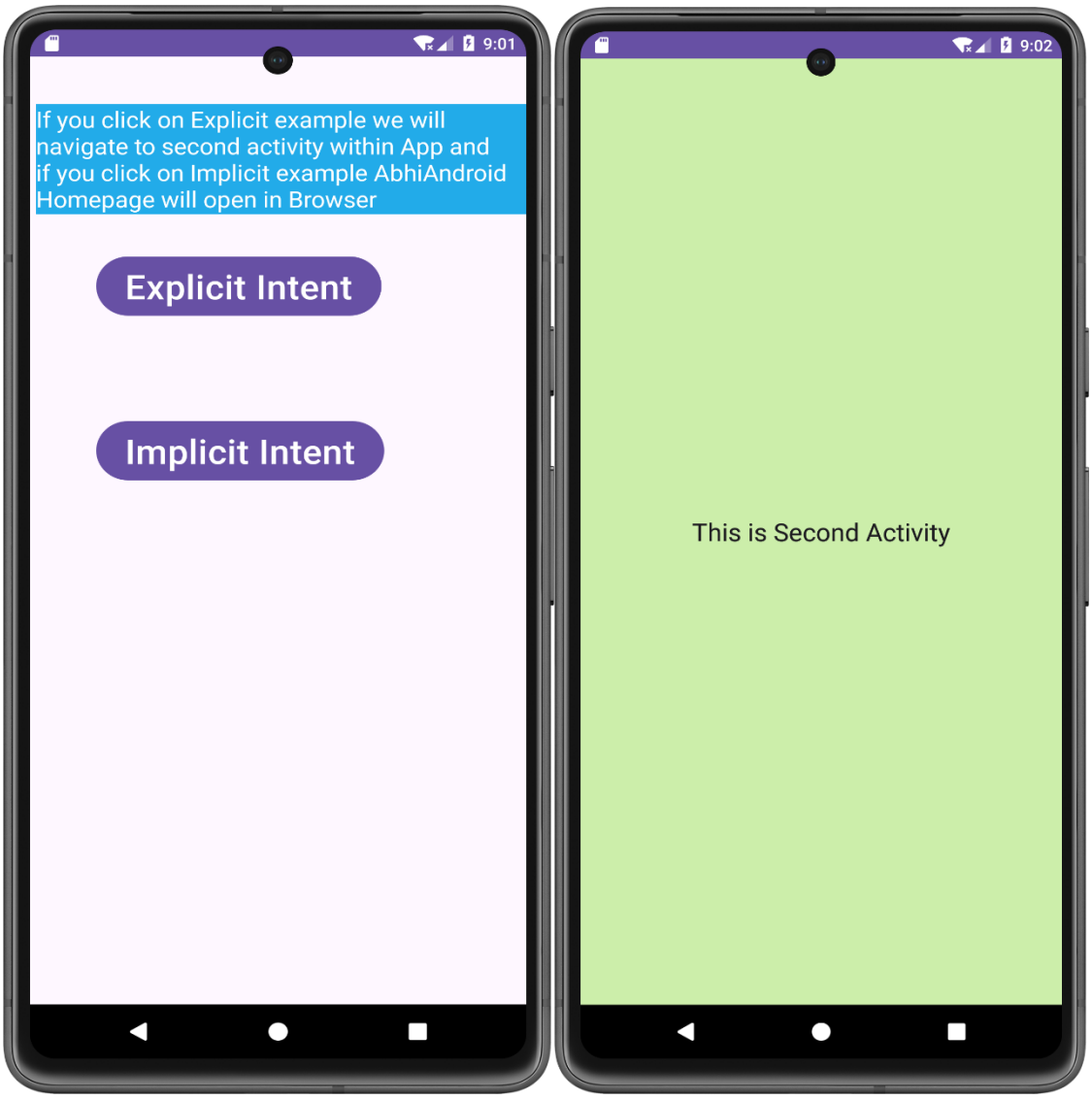
```
    android:id="@+id/textView2"
    android:clickable="false"
    android:layout_alignParentTop="true"
    android:layout_alignParentStart="true"
    android:layout_marginTop="42dp"
    android:layout_marginStart="5dp"
    android:textSize="20sp"
    android:background="#22ABE9"
    android:textColor="#ffffff" />

<Button
    android:id="@+id/explicit_Intent"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:layout_below="@id/textView2"
    android:layout_alignParentStart="true"
    android:layout_centerInParent="true"
    android:layout_marginStart="55dp"
    android:layout_marginTop="34dp"
    android:text="Explicit Intent"
    android:textSize="30sp" />

<Button
    android:id="@+id/implicit_Intent"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:layout_below="@id/explicit_Intent"
    android:layout_marginTop="86dp"
    android:layout_marginStart="55dp"
    android:text="Implicit Intent"
    android:textSize="30sp" />
```

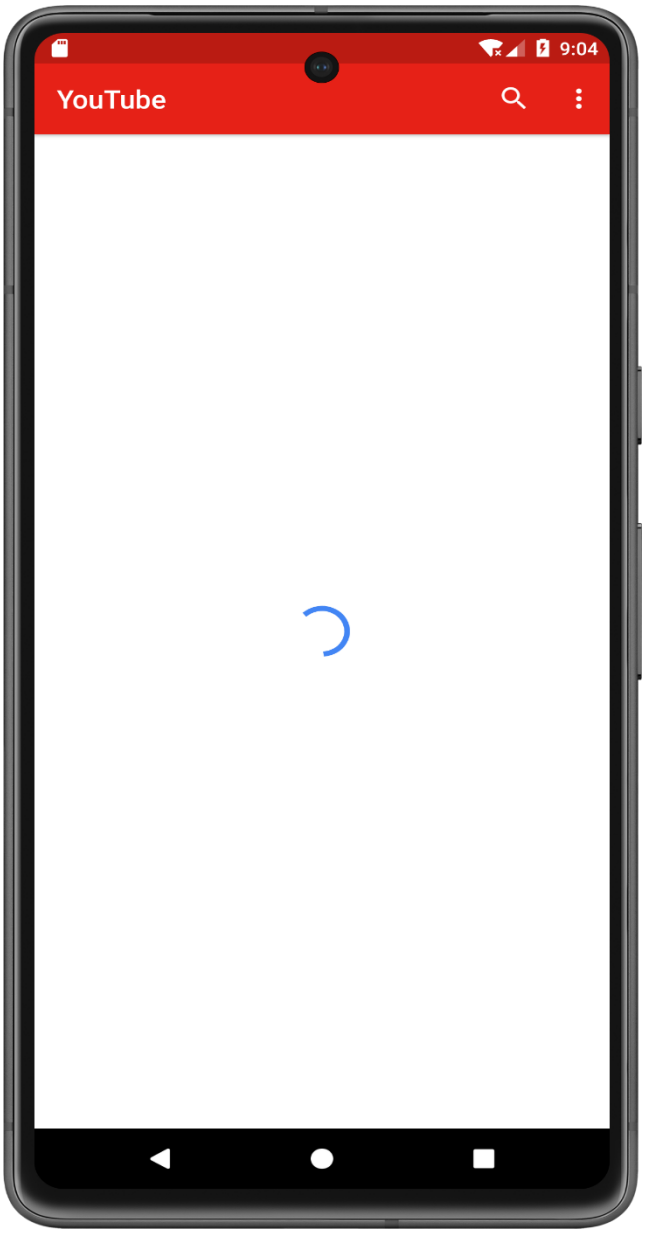
```
</RelativeLayout>
```

4. **activity_second.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"
android:paddingLeft="@dimen/activity_horizontal_margin"
   android:paddingRight="@dimen/activity_horizontal_margin"
   android:paddingTop="@dimen/activity_vertical_margin"
   android:paddingBottom="@dimen/activity_vertical_margin"
   android:background="#CCEEAA"
   tools:context="com.example.intentactivityexample.SecondActivity">

   <TextView
      android:layout_width="wrap_content"
      android:layout_height="wrap_content"
      android:textAppearance="?android:attr/textAppearanceLarge"
      android:text="This is Second Activity"
      android:id="@+id/textView"
      android:layout_centerVertical="true"
      android:layout_centerHorizontal="true" />
</RelativeLayout>
```

Output:


9:01
If you click on Explicit example we will navigate to second activity within App and if you click on Implicit example AbhiAndroid Homepage will open in Browser
Explicit Intent
Implicit Intent
9:02
This is Second Activity


9:04
YouTube

Assignment 7

Tittle: Program to implement fragment

Name: Tejas Sunil Mate

Class: MCA-III

Roll No: MC232538

Date:

Remark:

Practical No. 7: Develop a program to implement fragment.

Solution:

1. **MainActivity.java**

```
package com.example.fragmentexample;

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

public class MainActivity extends AppCompatActivity {

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

```
        if (savedInstanceState == null) {
            // Create a new instance of the fragment
            Fragment exampleFragment = new ExampleFragment();

            // Add the fragment to the container
            FragmentManager fragmentManager = getSupportFragmentManager();
            FragmentTransaction fragmentTransaction =
    fragmentManager.beginTransaction();
            fragmentTransaction.add(R.id.fragment_container, exampleFragment);
            fragmentTransaction.commit();
        }
    }
}
```

2. **activity_main.xml**

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

    <FrameLayout
        android:id="@+id/fragment_container"
        android:layout_width="match_parent"
        android:layout_height="match_parent"
        android:layout_marginTop="150sp"/>
</RelativeLayout>
```

3. **ExampleFragment.java**

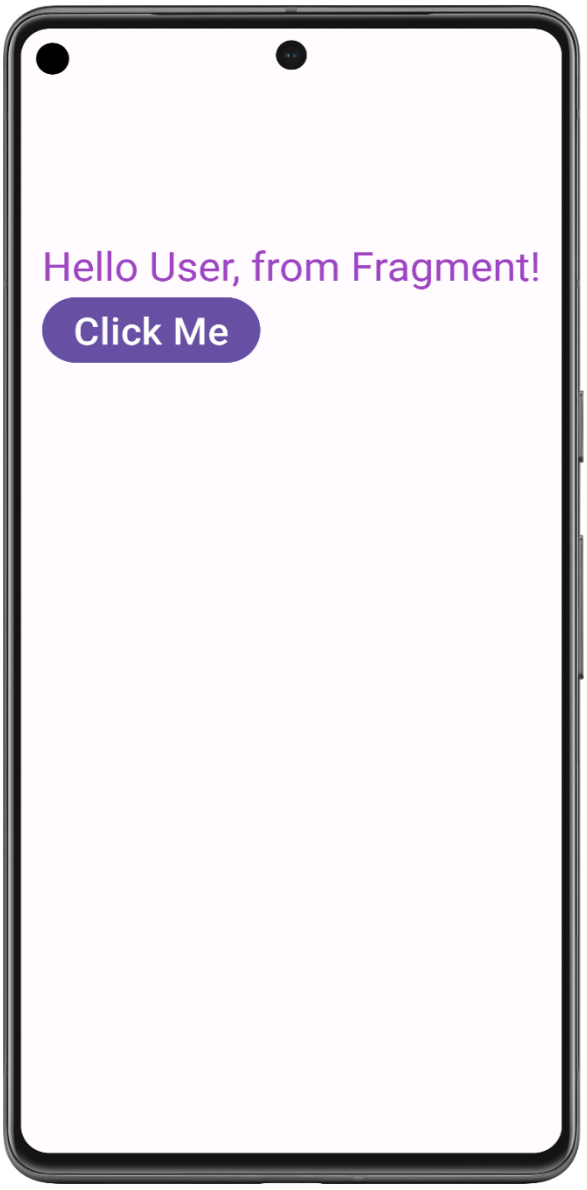
```
package com.example.fragmentexample;

import android.os.Bundle;
import androidx.annotation.NonNull;
import androidx.annotation.Nullable;
import androidx.fragment.app.Fragment;
```

```
import android.view.LayoutInflater;
import android.view.View;
import android.view.ViewGroup;

public class ExampleFragment extends Fragment {

   @Nullable
   @Override
   public View onCreateView(@NonNull LayoutInflater inflater, @Nullable
ViewGroup container,
                            @Nullable Bundle savedInstanceState) {
      // Inflate the layout for this fragment
      return inflater.inflate(R.layout.fragment_example, container, false);
   }
}
```

4. **fragment_example.xml**

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

   <TextView
      android:id="@+id/textView"
      android:layout_width="wrap_content"
      android:layout_height="wrap_content"
      android:text="Hello User, from Fragment!"
      android:textColor="@color/violet"
      android:textSize="32sp" />
   <Button
      android:layout_width="wrap_content"
      android:layout_height="wrap_content"
      android:textSize="30sp"
      android:text="Click Me"/>
</LinearLayout>
```

Output:



Assignment 8

Tittle: Program to implement Checkbox, radio button, Progress bar.

Name: Tejas Sunil Mate

Class: MCA-III

Roll No: MC232538

Date:

Remark:

Practical No. 8: Develop a program to implement Checkbox, radio button, Progress bar.

Solution:

1. **MainActivity.java**

```
package com.example.container;

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

public class MainActivity extends AppCompatActivity {

    private ProgressBar progressBar;
    private Button buttonShowProgress;

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

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


        buttonShowProgress.setOnClickListener(new View.OnClickListener() {
            @Override
            public void onClick(View v) {
                progressBar.setVisibility(View.VISIBLE);
                progressBar.postDelayed(new Runnable() {
                    @Override
                    public void run() {
                        progressBar.setVisibility(View.GONE);
```

```
                }
            }, 3000);
        }
    });
  }
}
```

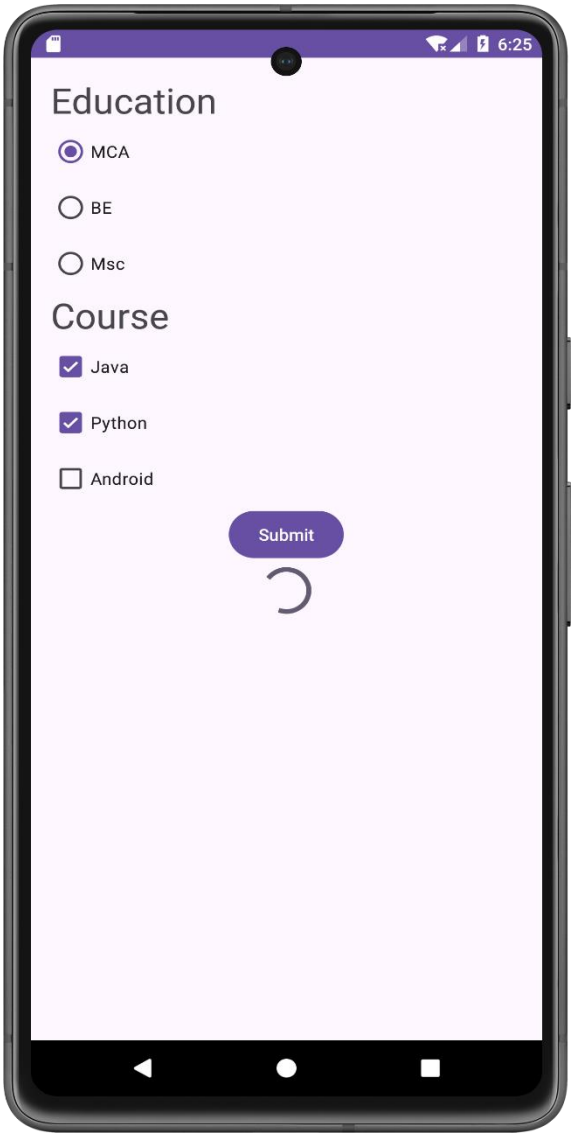
2. activity_main.xml

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

  <TextView
    android:id="@+id/textview1"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:textSize="30sp"
    android:text="Education"/>

  <RadioGroup
    android:id="@+id/radioGroup"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:layout_below="@id/textview1"
    android:orientation="vertical">

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

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

```
    <RadioButton
        android:id="@+id/radioButton3"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="Msc" />
</RadioGroup>

<TextView
    android:id="@+id/textview2"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:textSize="30sp"
    android:layout_below="@id/radioGroup"
    android:text="Course"/>

<CheckBox
    android:id="@+id/checkBox1"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:layout_below="@id/textview2"
    android:text="Java" />
<CheckBox
    android:id="@+id/checkBox2"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:layout_below="@id/checkBox1"
    android:text="Python" />
<CheckBox
    android:id="@+id/checkBox3"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:layout_below="@id/checkBox2"
    android:text="Android" />

<ProgressBar
    android:id="@+id/progressBar"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:layout_below="@id/buttonShowProgress"
    android:layout_centerHorizontal="true"
```

```
        android:visibility="gone" />

    <Button
        android:id="@+id/buttonShowProgress"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="Submit"
        android:layout_below="@id/checkBox3"
        android:layout_centerHorizontal="true" />
</RelativeLayout>
```

Output:



Assignment 9

Tittle: Passing data from one activity to another activity

Name: Tejas Sunil Mate

Class: MCA-III

Roll No: MC232538

Date:

Remark:

Practical No. 9: Passing data from one activity to another activity by using intent.

Solution:

1. **MainActivity.java**

```
package com.example.assigment9;

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

public class MainActivity extends AppCompatActivity {

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

        EditText usernameEditText = findViewById(R.id.editTextText);
        EditText passwordEditText = findViewById(R.id.editTextTextPassword);
        Button loginButton = findViewById(R.id.button);

        loginButton.setOnClickListener(new View.OnClickListener() {
            @Override
            public void onClick(View v) {
                String username = usernameEditText.getText().toString();
                String password = passwordEditText.getText().toString();
```

```
            // Create an Intent to start DisplayActivity
            Intent intent = new Intent(MainActivity.this, Activity2.class);
            intent.putExtra("USERNAME", username);
            intent.putExtra("PASSWORD", password);
            startActivity(intent);
        }
    });
  }
}
```

2. 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"
   android:background="@color/sky_blue"
   tools:context=".MainActivity">

   <EditText
      android:id="@+id/editTextText"
      android:layout_width="wrap_content"
      android:layout_height="wrap_content"
      android:ems="10"
      android:inputType="text"
      android:hint="Name"
      android:textSize="40sp"
      app:layout_constraintBottom_toBottomOf="parent"
      app:layout_constraintEnd_toEndOf="parent"
      app:layout_constraintHorizontal_bias="0.422"
      app:layout_constraintStart_toStartOf="parent"
      app:layout_constraintTop_toTopOf="parent"
      app:layout_constraintVertical_bias="0.123" />

   <EditText
      android:id="@+id/editTextTextPassword"
      android:layout_width="wrap_content"
```

```
            android:layout_height="wrap_content"
            android:ems="10"
            android:hint="Password"
            android:inputType="textPassword"
            android:textSize="40sp"
            app:layout_constraintBottom_toBottomOf="parent"
            app:layout_constraintEnd_toEndOf="parent"
            app:layout_constraintHorizontal_bias="0.422"
            app:layout_constraintStart_toStartOf="parent"
            app:layout_constraintTop_toTopOf="parent"
            app:layout_constraintVertical_bias="0.298" />

        <Button
            android:id="@+id/button"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:text="Login"
            android:textSize="40sp"
            app:layout_constraintBottom_toBottomOf="parent"
            app:layout_constraintEnd_toEndOf="parent"
            app:layout_constraintHorizontal_bias="0.534"
            app:layout_constraintStart_toStartOf="parent"
            app:layout_constraintTop_toTopOf="parent"
            app:layout_constraintVertical_bias="0.658" />
    </androidx.constraintlayout.widget.ConstraintLayout>
```

Activity2.java

```
package com.example.assigment9;

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

public class Activity2 extends AppCompatActivity {

   @Override
   protected void onCreate(Bundle savedInstanceState) {
      super.onCreate(savedInstanceState);
      setContentView(R.layout.activity_2);
```

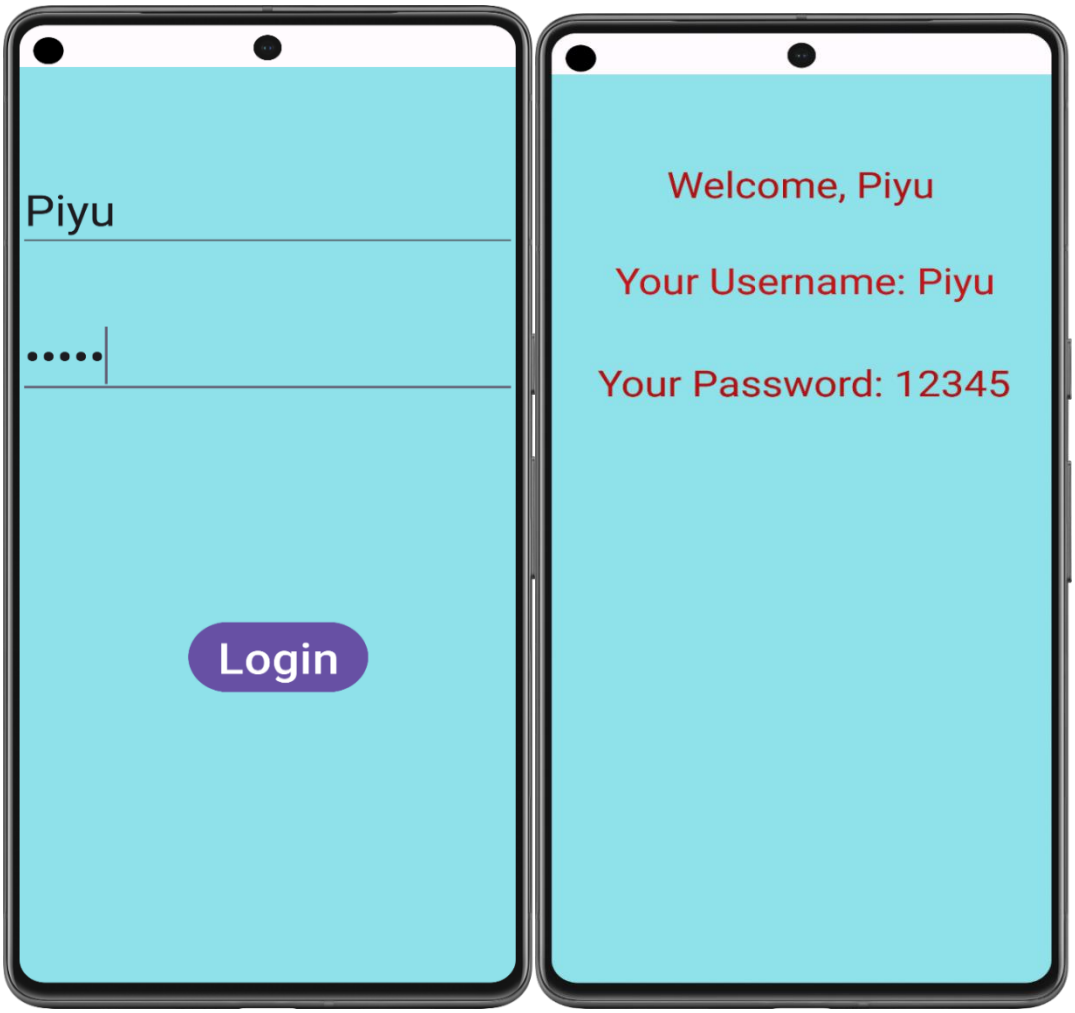
```
        TextView usernameTextView = findViewById(R.id.textView);
        TextView passwordTextView = findViewById(R.id.textView2);
        TextView welcomeTextView = findViewById(R.id.textView3);


        String username = getIntent().getStringExtra("USERNAME");
        String password = getIntent().getStringExtra("PASSWORD");

        welcomeTextView.setText("Welcome, "+username);
        usernameTextView.setText("Your Username: " + username);
        passwordTextView.setText("Your Password: " + password);
    }
}
```

activity_2.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"
    android:background="@color/sky_blue"
    tools:context=".Activity2">

    <TextView
        android:id="@+id/textView"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="TextView"
        android:textColor="#C81414"
        android:textSize="34sp"
        app:layout_constraintBottom_toBottomOf="parent"
        app:layout_constraintEnd_toEndOf="parent"
        app:layout_constraintHorizontal_bias="0.531"
        app:layout_constraintStart_toStartOf="parent"
        app:layout_constraintTop_toTopOf="parent"
```

```
        app:layout_constraintVertical_bias="0.211" />

    <TextView
        android:id="@+id/textView2"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="TextView"
        android:textColor="#B61111"
        android:textSize="34sp"
        app:layout_constraintBottom_toBottomOf="parent"
        app:layout_constraintEnd_toEndOf="parent"
        app:layout_constraintHorizontal_bias="0.531"
        app:layout_constraintStart_toStartOf="parent"
        app:layout_constraintTop_toTopOf="parent"
        app:layout_constraintVertical_bias="0.33" />

    <TextView
        android:id="@+id/textView3"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="TextView"
        android:textColor="#B61111"
        android:textSize="34sp"
        app:layout_constraintBottom_toBottomOf="parent"
        app:layout_constraintEnd_toEndOf="parent"
        app:layout_constraintHorizontal_bias="0.498"
        app:layout_constraintStart_toStartOf="parent"
        app:layout_constraintTop_toTopOf="parent"
        app:layout_constraintVertical_bias="0.099" />
</androidx.constraintlayout.widget.ConstraintLayout>
```

Output:


Piyu
Login
Welcome, Piyu
Your Username: Piyu
Your Password: 12345

Assignment 10

Tittle: Navigation of three activities

Name: Tejas Sunil Mate

Class: MCA-III

Roll No: MC232538

Date:

Remark:

Practical No. 10: Navigation of three activities.

Solution:

1. **MainActivity.java**

```
package com.example.navigation;

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

public class MainActivity extends AppCompatActivity {

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

        Button buttonToActivity2 = findViewById(R.id.button1);
        Button buttonToActivity3 = findViewById(R.id.button2);

        buttonToActivity2.setOnClickListener(new View.OnClickListener() {
            @Override
            public void onClick(View v) {
                Intent intent = new Intent(MainActivity.this, MainActivity2.class);
```

```
            startActivity(intent);
        }
    });

    buttonToActivity3.setOnClickListener(new View.OnClickListener() {
        @Override
        public void onClick(View v) {
            Intent intent = new Intent(MainActivity.this, MainActivity3.class);
            startActivity(intent);
        }
    });
  }
}
```

2. **MainActivity2.java**

```
package com.example.navigation;

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

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

public class MainActivity2 extends AppCompatActivity {

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

        Button buttonToActivity1 = findViewById(R.id.button1);
        Button buttonToActivity3 = findViewById(R.id.button2);

        buttonToActivity1.setOnClickListener(new View.OnClickListener() {
            @Override
```

```
        public void onClick(View v) {
            Intent intent = new Intent(MainActivity2.this, MainActivity.class);
            startActivity(intent);
        }
    });

    buttonToActivity3.setOnClickListener(new View.OnClickListener() {
        @Override
        public void onClick(View v) {
            Intent intent = new Intent(MainActivity2.this, MainActivity3.class);
            startActivity(intent);
        }
    });
  }
}
```

3. **MainActivity3.java**

```
package com.example.navigation;

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

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

public class MainActivity3 extends AppCompatActivity {

    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        EdgeToEdge.enable(this);
        setContentView(R.layout.activity_main3);
        Button buttonToActivity1 = findViewById(R.id.button1);
        Button buttonToActivity2 = findViewById(R.id.button2);

        buttonToActivity1.setOnClickListener(new View.OnClickListener() {
```

```
            @Override
            public void onClick(View v) {
                Intent intent = new Intent(MainActivity3.this, MainActivity.class);
                startActivity(intent);
            }
        });

        buttonToActivity2.setOnClickListener(new View.OnClickListener() {
            @Override
            public void onClick(View v) {
                Intent intent = new Intent(MainActivity3.this, MainActivity2.class);
                startActivity(intent);
            }
        });
    }
}
```

4. **activity_main.xml**

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

   <TextView
      android:id="@+id/textview1"
      android:layout_width="wrap_content"
      android:layout_height="wrap_content"
      android:text="Activity1"
      android:textColor="@color/orange"
      android:textSize="30sp"
      android:textStyle="normal|bold"
      android:layout_marginTop="50dp"/>
   <Button
      android:id="@+id/button1"
      android:layout_width="wrap_content"
      android:layout_height="wrap_content"
      android:text="Activity2"
      android:textSize="30sp"
      android:textStyle="normal|bold"
      android:layout_below="@+id/textview1"/>
```

```
    <Button
        android:id="@+id/button2"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="Activity3"
        android:textSize="30sp"
        android:textStyle="normal|bold"
        android:layout_below="@+id/button1"/>
</RelativeLayout>
```

5. **activity_main2.xml**

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

    <TextView
        android:id="@+id/textview1"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="Activity2"
        android:textSize="30sp"
        android:textColor="@color/red"
        android:textStyle="normal|bold"
        android:layout_marginTop="50dp"/>
    <Button
        android:id="@+id/button1"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="Activity1"
        android:textSize="30sp"
        android:textStyle="normal|bold"
        android:layout_below="@+id/textview1"/>

    <Button
        android:id="@+id/button2"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
```

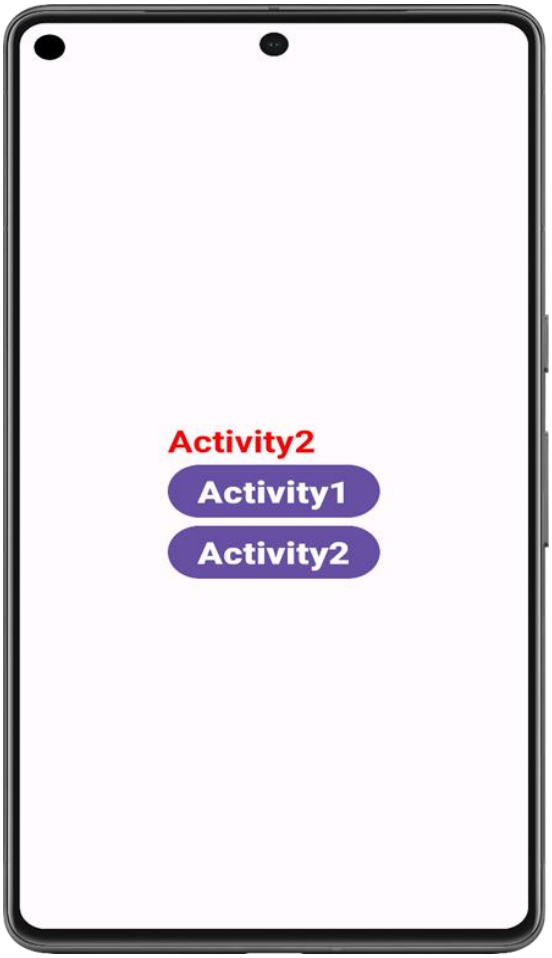
```
        android:text="Activity3"
        android:textSize="30sp"
        android:textStyle="normal|bold"
        android:layout_below="@+id/button1"/>
    </RelativeLayout>
```

6. **activity_main3.xml**

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

      <TextView
        android:id="@+id/textview1"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:gravity="center"
        android:text="Activity3"
        android:textColor="@color/blue"
        android:textSize="30sp"
        android:textStyle="normal|bold"
        android:layout_marginTop="50dp"/>
      <Button
        android:id="@+id/button1"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="Activity1"
        android:textSize="30sp"
        android:textStyle="normal|bold"
        android:layout_below="@+id/textview1"/>

      <Button
        android:id="@+id/button2"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="Activity2"
        android:textSize="30sp"
        android:textStyle="normal|bold"
        android:layout_below="@+id/button1"/>
```

</RelativeLayout>

Output:


Activity2
Activity1
Activity2

Assignment 11

Tittle: Demonstrate different Layouts with different views in android Layouts

Name: Tejas Sunil Mate

Class: MCA-III

Roll No: MC232528

Date:

Remark:

Practical No. 1: Demonstrate different Layouts with different views in android Layouts- ConstraintLayout, RelativeLayout, TableLayout Views- Button, TextView, EditText, WebView, CheckBox, RadioButton, ToggleButton, ImageButton,RatingBar, ProgressBar, SeekBar, VideoView, DatePicker, CalendarView, Spinner
Solution:

1. **activity_main.xml**

```
<androidx.constraintlayout.widget.ConstraintLayout
    xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:app="http://schemas.android.com/apk/res-auto"
    android:layout_width="match_parent"
    android:layout_height="match_parent">

    <TextView
        android:id="@+id/textView"
        android:layout_width="0dp"
        android:layout_height="wrap_content"
        android:text="Welcome!"
        app:layout_constraintTop_toTopOf="parent"
        app:layout_constraintStart_toStartOf="parent"
        app:layout_constraintEnd_toEndOf="parent"/>

    <EditText
        android:id="@+id/editText"
        android:layout_width="0dp"
        android:layout_height="wrap_content"
```

```
        android:hint="Enter text"
        app:layout_constraintTop_toBottomOf="@+id/textView"
        app:layout_constraintStart_toStartOf="parent"
        app:layout_constraintEnd_toEndOf="parent"/>

    <Button
        android:id="@+id/button"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="Submit"
        app:layout_constraintTop_toBottomOf="@+id/editText"
        app:layout_constraintStart_toStartOf="parent"
        app:layout_constraintEnd_toEndOf="parent"/>

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

Assignment 12

Tittle: Make phone call using Intent

Name: Tejas Sunil Mate

Class: MCA-III

Roll No: MC232538

Date:

Remark:

Practical No. 12: Write an android code to make phone call using Intent.

Solution:

A. 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_toTopOf="@+id/callButton"
        app:layout_constraintEnd_toEndOf="parent"
        app:layout_constraintStart_toStartOf="parent"
        app:layout_constraintTop_toTopOf="parent" />
```

```
    <Button
        android:id="@+id/callButton"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="Make Call"
        app:layout_constraintTop_toBottomOf="@+id/textView"
        app:layout_constraintStart_toStartOf="parent"
        app:layout_constraintEnd_toEndOf="parent"
        android:layout_marginTop="16dp" />

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

B. MainActivity.java

```
package com.vishwakarma.phonecall;

import android.Manifest;
import android.content.Intent;
import android.content.pm.PackageManager;
import android.net.Uri;
import android.os.Bundle;
import android.view.View;
import android.widget.Button;
import android.widget.Toast;

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

public class MainActivity extends AppCompatActivity {

    private static final int REQUEST_CALL_PERMISSION = 1;

    @Override
```

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

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

        // Set up the call button
        Button callButton = findViewById(R.id.callButton);
        callButton.setOnClickListener(new View.OnClickListener() {
            @Override
            public void onClick(View v) {
                makePhoneCall();
            }
        });
    }

    private void makePhoneCall() {
        String phoneNumber = "tel:1234567890"; // Replace with your phone number
        Intent callIntent = new Intent(Intent.ACTION_CALL);
        callIntent.setData(Uri.parse(phoneNumber));

        if (ActivityCompat.checkSelfPermission(this,
Manifest.permission.CALL_PHONE) !=
PackageManager.PERMISSION_GRANTED) {
            ActivityCompat.requestPermissions(this, new
String[]{Manifest.permission.CALL_PHONE}, REQUEST_CALL_PERMISSION);
        } else {
            startActivity(callIntent);
        }
    }
```

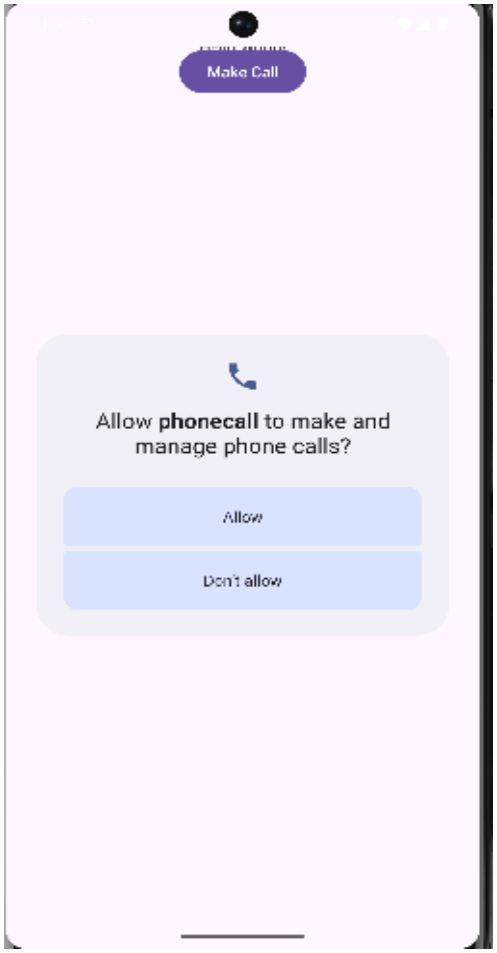
```
    @Override
    public void onRequestPermissionsResult(int requestCode, @NonNull String[]
permissions, @NonNull int[] grantResults) {
        super.onRequestPermissionsResult(requestCode, permissions, grantResults);
        if (requestCode == REQUEST_CALL_PERMISSION) {
            if (grantResults.length > 0 && grantResults[0] ==
PackageManager.PERMISSION_GRANTED) {
                makePhoneCall();
            } else {
                Toast.makeText(this, "Permission DENIED",
Toast.LENGTH_SHORT).show();
            }
        }
    }
}
```

C. Android manifest.xml

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

    <!-- Add the permission here -->
    <uses-permission android:name="android.permission.CALL_PHONE"/>

    <application
        android:allowBackup="true"
        android:dataExtractionRules="@xml/data_extraction_rules"
        android:fullBackupContent="@xml/backup_rules"
        android:icon="@mipmap/ic_launcher"
        android:label="@string/app_name"
        android:roundIcon="@mipmap/ic_launcher_round"
        android:supportsRtl="true"
        android:theme="@style/Theme.Phonecall"
        tools:targetApi="31">
        <activity
            android:name=".MainActivity"
            android:exported="true">
            <intent-filter>
                <action android:name="android.intent.action.MAIN" />
                <category android:name="android.intent.category.LAUNCHER" />
```

```
        </intent-filter>
    </activity>
</application>

</manifest>
```

Output:



Assignment 13

Tittle: Turn ON/OFF Bluetooth

Name: Tejas Sunil Mate

Class: MCA-III

Roll No: MC232538

Date:

Remark:

Practical No. 13: Write an android code to turn ON/OFF Bluetooth.

Solution:

a. 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">

    <Button
        android:id="@+id/toggleBluetoothButton"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="Turn On Bluetooth"
        app:layout_constraintTop_toTopOf="parent"
        app:layout_constraintBottom_toBottomOf="parent"
        app:layout_constraintStart_toStartOf="parent"
        app:layout_constraintEnd_toEndOf="parent"/>

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

b. MainActivity.java

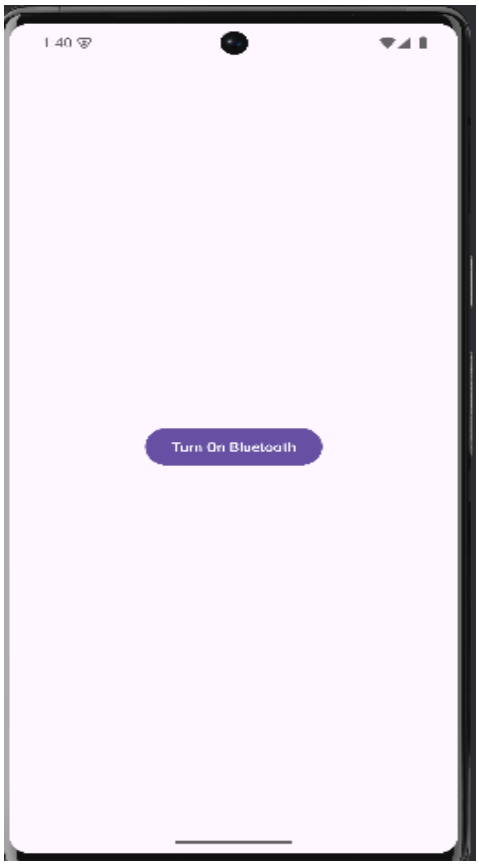
```
package com.vishwakarma.bluetooth;

import android.bluetooth.BluetoothAdapter;
import android.os.Bundle;
import android.view.View;
import android.widget.Button;
import android.widget.Toast;

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

public class MainActivity extends AppCompatActivity {

    private BluetoothAdapter bluetoothAdapter;
    private Button toggleBluetoothButton;

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

        // Set up the button and Bluetooth adapter
        toggleBluetoothButton = findViewById(R.id.toggleBluetoothButton);
        bluetoothAdapter = BluetoothAdapter.getDefaultAdapter();

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

        updateButtonText();
```

```
        // Set up button click listener
        toggleBluetoothButton.setOnClickListener(new View.OnClickListener() {
            @Override
            public void onClick(View v) {
                toggleBluetooth();
            }
        });
    }

    private void toggleBluetooth() {
        if (bluetoothAdapter != null) {
            if (bluetoothAdapter.isEnabled()) {
                bluetoothAdapter.disable();
                Toast.makeText(this, "Bluetooth turned off",
Toast.LENGTH_SHORT).show();
            } else {
                bluetoothAdapter.enable();
                Toast.makeText(this, "Bluetooth turned on",
Toast.LENGTH_SHORT).show();
            }
            updateButtonText();
        } else {
            Toast.makeText(this, "Bluetooth not supported on this device",
Toast.LENGTH_SHORT).show();
        }
    }

    private void updateButtonText() {
        if (bluetoothAdapter != null) {
            if (bluetoothAdapter.isEnabled()) {
                toggleBluetoothButton.setText("Turn Off Bluetooth");
            } else {
                toggleBluetoothButton.setText("Turn On Bluetooth");
            }
        }
    }
}
```

c. Android manifest.xml

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

    <!-- Add Bluetooth permissions -->
    <uses-permission android:name="android.permission.BLUETOOTH"/>
    <uses-permission android:name="android.permission.BLUETOOTH_ADMIN"/>

    <application
        android:allowBackup="true"
        android:dataExtractionRules="@xml/data_extraction_rules"
        android:fullBackupContent="@xml/backup_rules"
        android:icon="@mipmap/ic_launcher"
        android:label="@string/app_name"
        android:roundIcon="@mipmap/ic_launcher_round"
        android:supportsRtl="true"
        android:theme="@style/Theme.Bluetooth"
        tools:targetApi="31">
        <activity
            android:name=".MainActivity"
            android:exported="true">
            <intent-filter>
                <action android:name="android.intent.action.MAIN" />
                <category android:name="android.intent.category.LAUNCHER" />
            </intent-filter>
        </activity>
    </application>

</manifest>
```

Output:


1:40
Turn On Bluetooth

Assignment 14

Tittle: Turn ON/OFF Bluetooth

Name: Tejas Sunil Mate

Class: MCA-III

Roll No: MC232538

Date:

Remark:

Practical No. 14: Write an Android Code to turn on ON/OFF Bluetooth

Solution:

1. **MainActivity.java**

```
package com.vishwakarma.wifi;

import android.content.Context;
import android.net.wifi.WifiManager;
import android.os.Bundle;
import android.view.View;
import android.widget.Button;
import android.widget.Toast;

import androidx.appcompat.app.AppCompatActivity;

public class MainActivity extends AppCompatActivity {

    private WifiManager wifiManager;
    private Button toggleWifiButton;

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

        toggleWifiButton = findViewById(R.id.toggleWifiButton);
        wifiManager = (WifiManager)
getApplicationContext().getSystemService(Context.WIFI_SERVICE);

        updateButtonText();
```

```
        toggleWifiButton.setOnClickListener(new View.OnClickListener() {
            @Override
            public void onClick(View v) {
                toggleWifi();
            }
        });
    }

    private void toggleWifi() {
        if (wifiManager != null) {
            if (wifiManager.isWifiEnabled()) {
                wifiManager.setWifiEnabled(false);
                Toast.makeText(this, "Wi-Fi turned off", Toast.LENGTH_SHORT).show();
            } else {
                wifiManager.setWifiEnabled(true);
                Toast.makeText(this, "Wi-Fi turned on", Toast.LENGTH_SHORT).show();
            }
            updateButtonText();
        } else {
            Toast.makeText(this, "Wi-Fi not supported on this device",
Toast.LENGTH_SHORT).show();
        }
    }

    private void updateButtonText() {
        if (wifiManager != null) {
            if (wifiManager.isWifiEnabled()) {
                toggleWifiButton.setText("Turn Off Wi-Fi");
            } else {
                toggleWifiButton.setText("Turn On Wi-Fi");
            }
        }
    }
}
```

2. **activity_main.xml**

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

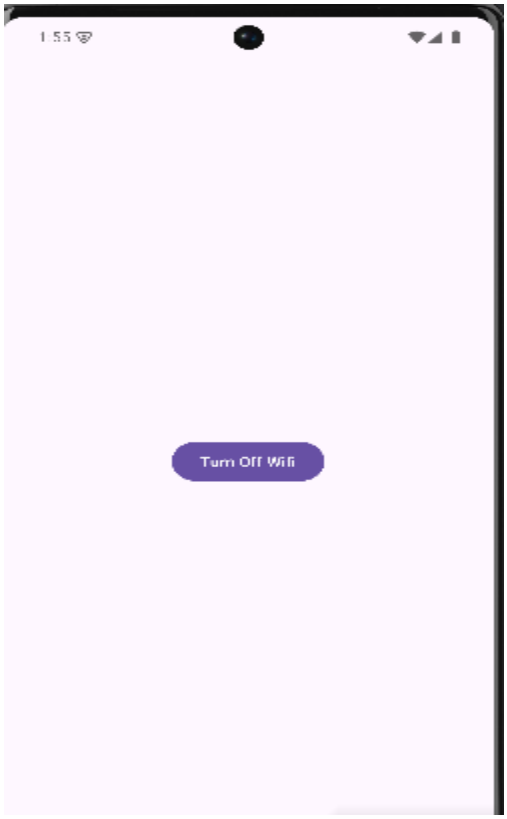
```
    android:layout_height="match_parent">

    <Button
        android:id="@+id/toggleWifiButton"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="Turn On Wi-Fi"
        android:layout_centerInParent="true"/>
</RelativeLayout>
```

3. **AndroidManifest.xml**

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

    <uses-permission android:name="android.permission.ACCESS_WIFI_STATE"/>
    <uses-permission android:name="android.permission.CHANGE_WIFI_STATE"/>

    <application
        android:allowBackup="true"
        android:dataExtractionRules="@xml/data_extraction_rules"
        android:fullBackupContent="@xml/backup_rules"
        android:icon="@mipmap/ic_launcher"
        android:label="@string/app_name"
        android:roundIcon="@mipmap/ic_launcher_round"
        android:supportsRtl="true"
        android:theme="@style/Theme.Wifi"
        tools:targetApi="31">
        <activity
            android:name=".MainActivity"
            android:exported="true">
            <intent-filter>
                <action android:name="android.intent.action.MAIN" />

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

</manifest>

Output:



Assignment 15

Tittle: Design android application for login activity.

Name: Tejas Sunil Mate

Class: MCA-III

Roll No: MC232538

Date:

Remark:

Practical No. 5: Design android application for login activity. Write android code to check login credentials with username=”mca” and password=”android”. Display appropriate toast message to the user.

Solution:

1. **MainActivity.java**

```
package com.example.loginapp;

import android.os.Bundle;

import android.view.View;

import android.widget.Button;

import android.widget.EditText;

import android.widget.Toast;

import androidx.appcompat.app.AppCompatActivity;

public class LoginActivity extends AppCompatActivity {

    private EditText editTextUsername;

    private EditText editTextPassword;

    private Button buttonLogin;

    private static final String VALID_USERNAME = "mca";

    private static final String VALID_PASSWORD = "android";
```

```
    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_login);
        editTextUsername = findViewById(R.id.editTextUsername);
        editTextPassword = findViewById(R.id.editTextPassword);
        buttonLogin = findViewById(R.id.buttonLogin);
        buttonLogin.setOnClickListener(new View.OnClickListener() {
            @Override
            public void onClick(View v) {
                checkLoginCredentials();
            }
        });
    }
    private void checkLoginCredentials() {
        String username = editTextUsername.getText().toString().trim();
        String password = editTextPassword.getText().toString().trim();

        if (username.equals(VALID_USERNAME) &&
password.equals(VALID_PASSWORD)) {
            Toast.makeText(this, "Login Successful!", Toast.LENGTH_SHORT).show();
        } else {
            Toast.makeText(this, "Invalid credentials. Please try again.",
Toast.LENGTH_SHORT).show();
        }
```

```
    }
}
```

2. activity_main.xml

```
<?xml version="1.0" encoding="utf-8"?>
<LinearLayout xmlns:android="http://schemas.android.com/apk/res/android"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    android:orientation="vertical"
    android:padding="16dp">
    <EditText
        android:id="@+id/editTextUsername"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:hint="Username" />
    <EditText
        android:id="@+id/editTextPassword"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:hint="Password"
        android:inputType="textPassword" />
    <Button
        android:id="@+id/buttonLogin"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
```

```
        android:text="Login" />
</LinearLayout>
```

Assignment 16

Tittle: Create a fragment

Name: Tejas Sunil Mate

Class: MCA-III

Roll No: MC232538

Date:

Remark:

Practical No. 16: Create a fragment that has its own UI and enable your activities to communicate with fragments.

Solution:-

1. **MainActivity.java**

```
package com.vishwakarma.fragment;

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

public class MainActivity extends AppCompatActivity implements
MyFragment.FragmentCommunicationListener {

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

        if (savedInstanceState == null) {
            getSupportFragmentManager().beginTransaction()
                    .replace(R.id.fragment_container, new MyFragment())
                    .commit();
        }
    }

    @Override
    public void onButtonClicked(String message) {
        Toast.makeText(this, "Received from Fragment: " + message,
Toast.LENGTH_SHORT).show();
```

```
    }
}
```

2. **MyFragment.java**

```
package com.vishwakarma.fragment;

import android.content.Context; // Import Context
import android.os.Bundle;
import android.view.LayoutInflater;
import android.view.View;
import android.view.ViewGroup;
import android.widget.Button;
import android.widget.TextView;
import androidx.annotation.NonNull;
import androidx.annotation.Nullable;
import androidx.fragment.app.Fragment;

public class MyFragment extends Fragment {

    private TextView textView;
    private Button button;
    private FragmentCommunicationListener listener;

    public interface FragmentCommunicationListener {
        void onButtonClicked(String message);
    }

    @Override
    public void onAttach(@NonNull Context context) {
        super.onAttach(context);
        if (context instanceof FragmentCommunicationListener) {
            listener = (FragmentCommunicationListener) context;
        } else {
            throw new RuntimeException(context.toString() + " must implement
FragmentCommunicationListener");
        }
    }

    @Nullable
    @Override
```

```
    public View onCreateView(@NonNull LayoutInflater inflater, @Nullable ViewGroup
container,
                             @Nullable Bundle savedInstanceState) {
        View view = inflater.inflate(R.layout.fragment_my, container, false);

        textView = view.findViewById(R.id.textView);
        button = view.findViewById(R.id.button);

        button.setOnClickListener(v -> {
            String message = "Hello from Fragment!";
            textView.setText(message);
            if (listener != null) {
                listener.onButtonClicked(message);
            }
        });

        return view;
    }
}
```

3. activity_main.xml

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

    <FrameLayout
        android:id="@+id/fragment_container"
        android:layout_width="match_parent"
        android:layout_height="match_parent"/>
</RelativeLayout>
```

4. Fragment_my.xml

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

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

    <TextView
        android:id="@+id/textView"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="Fragment UI"
        android:textSize="20sp"/>

    <Button
        android:id="@+id/button"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="Send Message"/>
</LinearLayout>
```

Output:

Assignment 17

Tittle: Array Adapter using List View

Name:

Class: MCA-III

Roll No: MC2325

Date:

Remark:

Practical No. 17: Demonstrate Array Adapter using List View to display list of fruits.

Solution:

1. **MainActivity.java**

```
package com.example.arrayadapter;

import android.os.Bundle;
import android.widget.ArrayAdapter;
import android.widget.ListView;
import androidx.appcompat.app.AppCompatActivity;

public class MainActivity extends AppCompatActivity {

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

        ListView listView = findViewById(R.id.listView);

        // Sample data for the ListView
        String[] fruits = {
                "Apple",
                "Banana",
                "Cherry",
                "Date",
                "Grape",
                "Orange",
```

```
            "Pineapple"
        };

        // Create an ArrayAdapter using the custom layout for each item
        ArrayAdapter<String> adapter = new ArrayAdapter<>(
                this,
                R.layout.list_item,  // Use the custom layout
                R.id.textView,       // TextView ID in the custom layout
                fruits
        );

        // Set the adapter to the ListView
        listView.setAdapter(adapter);
    }
}
```

2. **activity_main.xml**

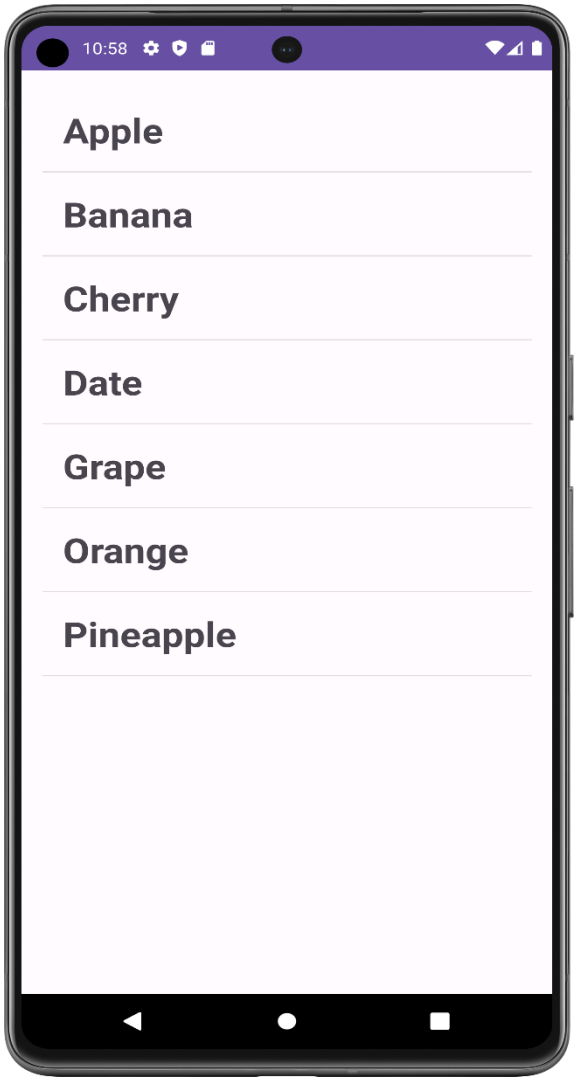
```
<?xml version="1.0" encoding="utf-8"?>
<LinearLayout xmlns:android="http://schemas.android.com/apk/res/android"
    android:orientation="vertical"
    android:layout_width="match_parent"
    android:layout_height="match_parent">

    <ListView
        android:id="@+id/listView"
        android:layout_width="match_parent"
        android:layout_height="match_parent"
        android:padding="16dp"
        android:textSize="40sp" />

</LinearLayout>
```

3. **list_item.xml**

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

```
android:padding="16dp"
android:textStyle="bold"
android:textSize="30sp" />
```

Output:



Assignment 18

Tittle: Alert Dialog Box in android

Name: Tejas Sunil Mate

Class: MCA-III

Roll No: MC232538

Date:

Remark:

Practical No. 18: Write an application to demonstrate Alert Dialog Box in android

Solution:

1. **MainActivity.java**

```
package com.example.alertdialogbox;

import android.content.DialogInterface;
import android.os.Bundle;
import android.text.InputType;
import android.widget.Button;
import android.widget.EditText;
import android.widget.Toast;
import androidx.appcompat.app.AlertDialog;
import androidx.appcompat.app.AppCompatActivity;

public class MainActivity extends AppCompatActivity {

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

        Button buttonShowInputDialog = findViewById(R.id.buttonShowInputDialog);

        buttonShowInputDialog.setOnClickListener(v -> showInputDialog());
    }

    private void showInputDialog() {
        AlertDialog.Builder builder = new AlertDialog.Builder(this);
```

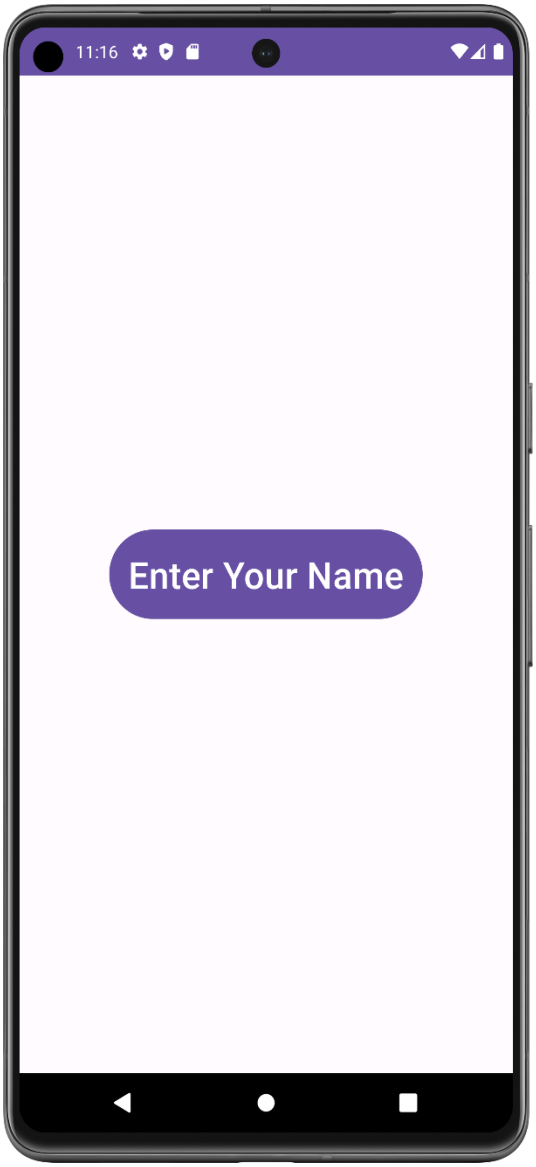
```
        builder.setTitle("Enter Your Name");

        // Set up the input
        final EditText input = new EditText(this);
        input.setInputType(InputType.TYPE_CLASS_TEXT);
        builder.setView(input);

        // Set up the buttons
        builder.setPositiveButton("Submit", new DialogInterface.OnClickListener() {
            @Override
            public void onClick(DialogInterface dialog, int which) {
                String name = input.getText().toString();
                if (!name.isEmpty()) {
                    Toast.makeText(MainActivity.this, "Hello, " + name + "!",
    Toast.LENGTH_SHORT).show();
                } else {
                    Toast.makeText(MainActivity.this, "Please enter a name!",
    Toast.LENGTH_SHORT).show();
                }
            }
        });

        builder.setNegativeButton("Cancel", new DialogInterface.OnClickListener() {
            @Override
            public void onClick(DialogInterface dialog, int which) {
                dialog.cancel();
            }
        });

        // Create and show the AlertDialog
        AlertDialog dialog = builder.create();
        dialog.show();
    }
}
```

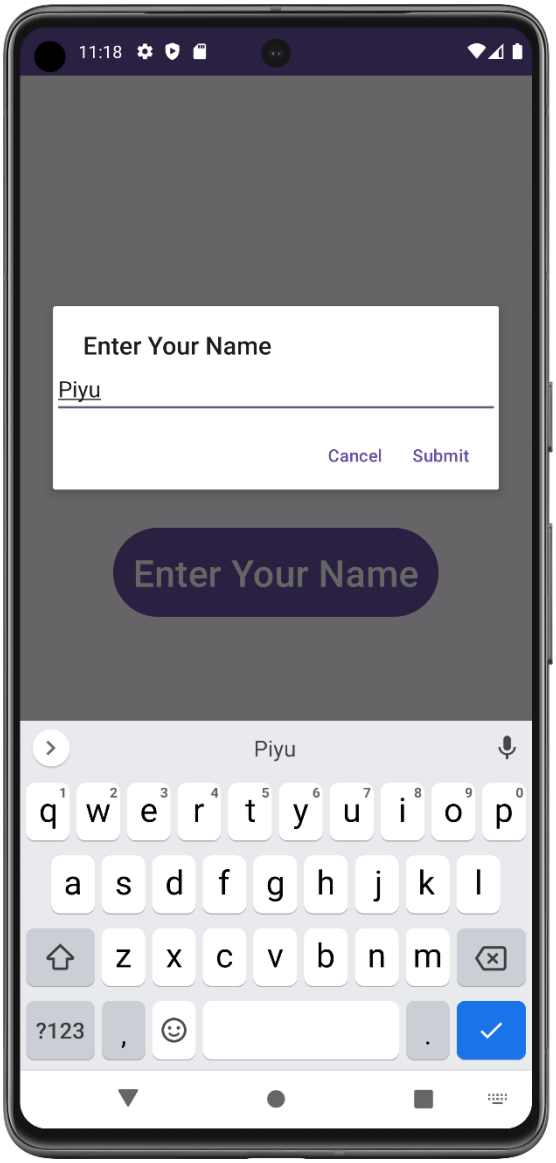
2. activity_main.xml

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

```
    android:orientation="vertical"
    android:gravity="center"
    android:padding="16dp">

    <Button
        android:id="@+id/buttonShowInputDialog"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="@string/enter_your_name"
        android:textSize="30sp"
        android:padding="16dp" />
</LinearLayout>
```

Output:


11:16
Enter Your Name


11:18
Enter Your Name
Piyu
Cancel
Submit
Enter Your Name
Piyu

v

Assignment 19

Tittle: Demonstrate Options Menu, Context Menu and Popup Menu in android

Name: Tejas Sunil Mate

Class: MCA-III

Roll No: MC232538

Date:

Remark:

Practical No. 9: Demonstrate Options Menu, Context Menu and Popup Menu in android

Solution:

1. **Options Menu**
   - **MainActivity.java**
   - **activity_main.xml**
2. **Context Menu**
   - **MainActivity.java**

```
package com.example.contextmenu;

import android.graphics.Color;
import android.os.Bundle;
import android.view.ContextMenu;
import android.view.MenuItem;
import android.view.View;
import android.widget.RelativeLayout;
import android.widget.TextView;
import androidx.appcompat.app.AppCompatActivity;

public class MainActivity extends AppCompatActivity {
   TextView textView;
   RelativeLayout relativeLayout;

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

```
        // Link those objects with their respective id's that we have given in .XML file
        textView = (TextView) findViewById(R.id.textView);
        relativeLayout = (RelativeLayout) findViewById(R.id.relLayout);

        // here you have to register a view for context menu you can register any
    view
        // like listview, image view, textview, button etc
        registerForContextMenu(textView);

    }

    @Override
    public void onCreateContextMenu(ContextMenu menu, View v,
    ContextMenu.ContextMenuInfo menuInfo) {
        super.onCreateContextMenu(menu, v, menuInfo);
        // you can set menu header with title icon etc
        menu.setHeaderTitle("Choose a color");
        // add menu items
        menu.add(0, v.getId(), 0, "Yellow");
        menu.add(0, v.getId(), 0, "Gray");
        menu.add(0, v.getId(), 0, "Cyan");
    }

    // menu item select listener
    @Override
    public boolean onContextItemSelected(MenuItem item) {
        if (item.getTitle() == "Yellow") {
            relativeLayout.setBackgroundColor(Color.YELLOW);
        } else if (item.getTitle() == "Gray") {
            relativeLayout.setBackgroundColor(Color.GRAY);
        } else if (item.getTitle() == "Cyan") {
            relativeLayout.setBackgroundColor(Color.CYAN);
        }
        return true;
    }
}
```

- **activity_main.xml**

```
<?xml version="1.0" encoding="utf-8"?>
<!-- Relative Layout to display all the details -->
<RelativeLayout xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:tools="http://schemas.android.com/tools"
    android:id="@+id/relLayout"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    android:background="#fff"
    android:padding="16dp"
    tools:context=".MainActivity">

    <TextView
        android:id="@+id/textView"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_centerHorizontal="true"
        android:layout_marginTop="20dp"
        android:text="Long press me!"
        android:textColor="#000"
        android:textSize="20sp"
        android:textStyle="bold" />
</RelativeLayout>
```

❖ **Output:**

3. **Popup Menu**

❖ **MainActivity.java**

```
package com.example.popupmenu;

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 button;

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

```
        // Referencing and Initializing the button
        button = (Button) findViewById(R.id.clickBtn);

        // Setting onClick behavior to the button
        button.setOnClickListener(new View.OnClickListener() {
            @Override
            public void onClick(View view) {
                // Initializing the popup menu and giving the reference as current
context
                PopupMenu popupMenu = new PopupMenu(MainActivity.this, button);

                // Inflating popup menu from popup_menu.xml file
                popupMenu.getMenuInflater().inflate(R.menu.popup_menu,
popupMenu.getMenu());
                popupMenu.setOnMenuItemClickListener(new
PopupMenu.OnMenuItemClickListener() {
                    @Override
                    public boolean onMenuItemClick(MenuItem menuItem) {
                        // Toast message on menu item clicked
                        Toast.makeText(MainActivity.this, "You Clicked " +
menuItem.getTitle(), Toast.LENGTH_SHORT).show();
                        return true;
                    }
                });
                // Showing the popup menu
                popupMenu.show();
            }
        });
    }
}
```

❖ **activity_main.xml**

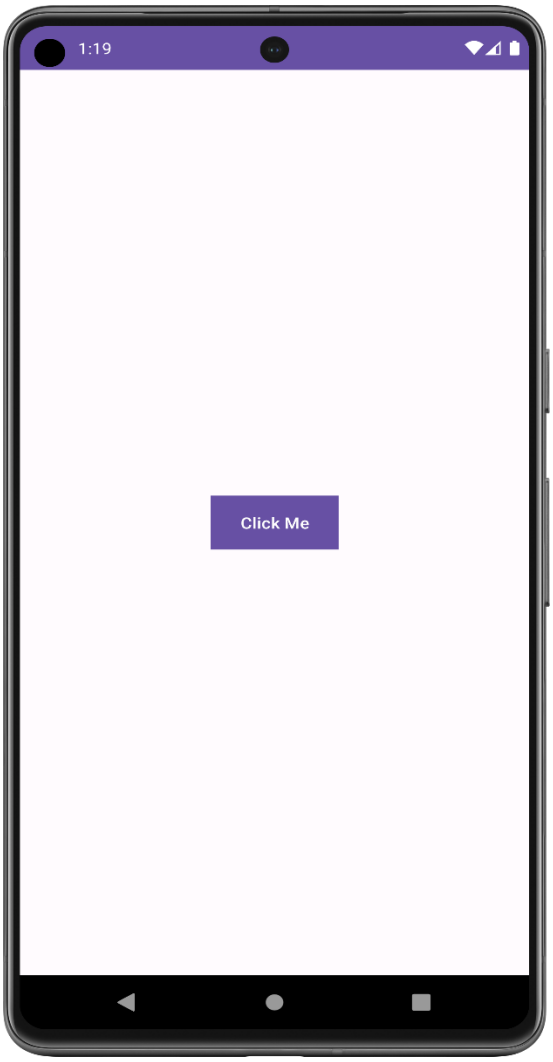
```
<?xml version="1.0" encoding="utf-8"?>
<androidx.constraintlayout.widget.ConstraintLayout
xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:app="http://schemas.android.com/apk/res-auto"
    xmlns:tools="http://schemas.android.com/tools"
    android:layout_width="match_parent"
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

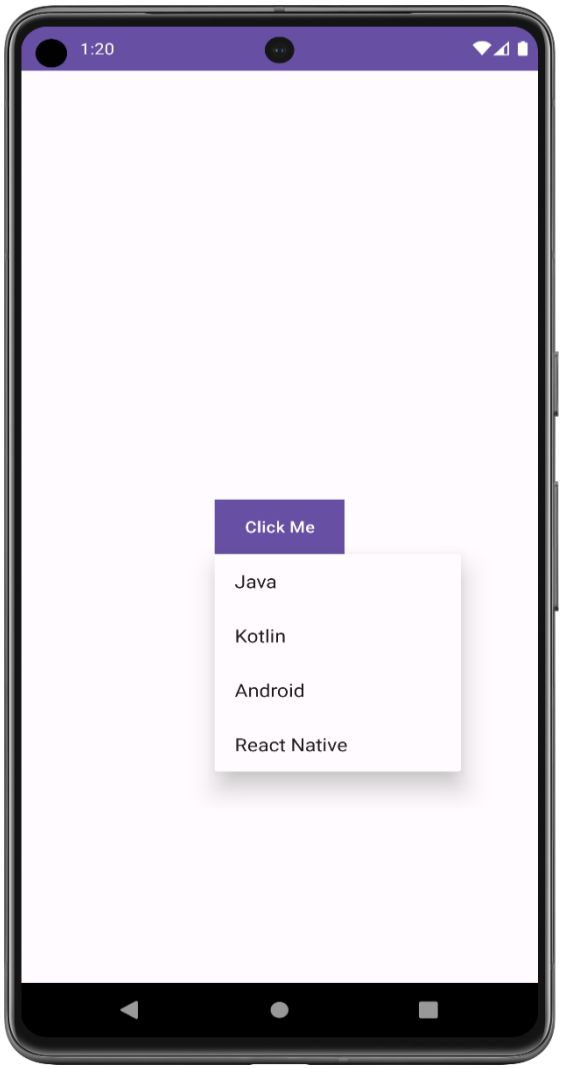
```
    <Button
        android:id="@+id/clickBtn"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:background="#0F9D58"
        android:text="Click Me"
        android:textColor="#ffffff"
        app:layout_constraintBottom_toBottomOf="parent"
        app:layout_constraintLeft_toLeftOf="parent"
        app:layout_constraintRight_toRightOf="parent"
        app:layout_constraintTop_toTopOf="parent" />
</androidx.constraintlayout.widget.ConstraintLayout>
```

- **res/menu/popup_menu.xml**

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

    <item
        android:id="@+id/kotlin"
        android:title="Kotlin" />

    <item
        android:id="@+id/android"
        android:title="Android" />

    <item
        android:id="@+id/react_native"
        android:title="React Native" />
</menu>
```

Output:


1:19
Click Me


1:20
Click Me
Java
Kotlin
Android
React Native

Assignment 20

Tittle: Write an application to produce Notification

Name: Tejas Sunil Mate

Class: MCA-III

Roll No: MC232538

Date:

Remark:

Practical No. 20: Write an application to produce Notification

Solution:

1. **MainActivity.java**

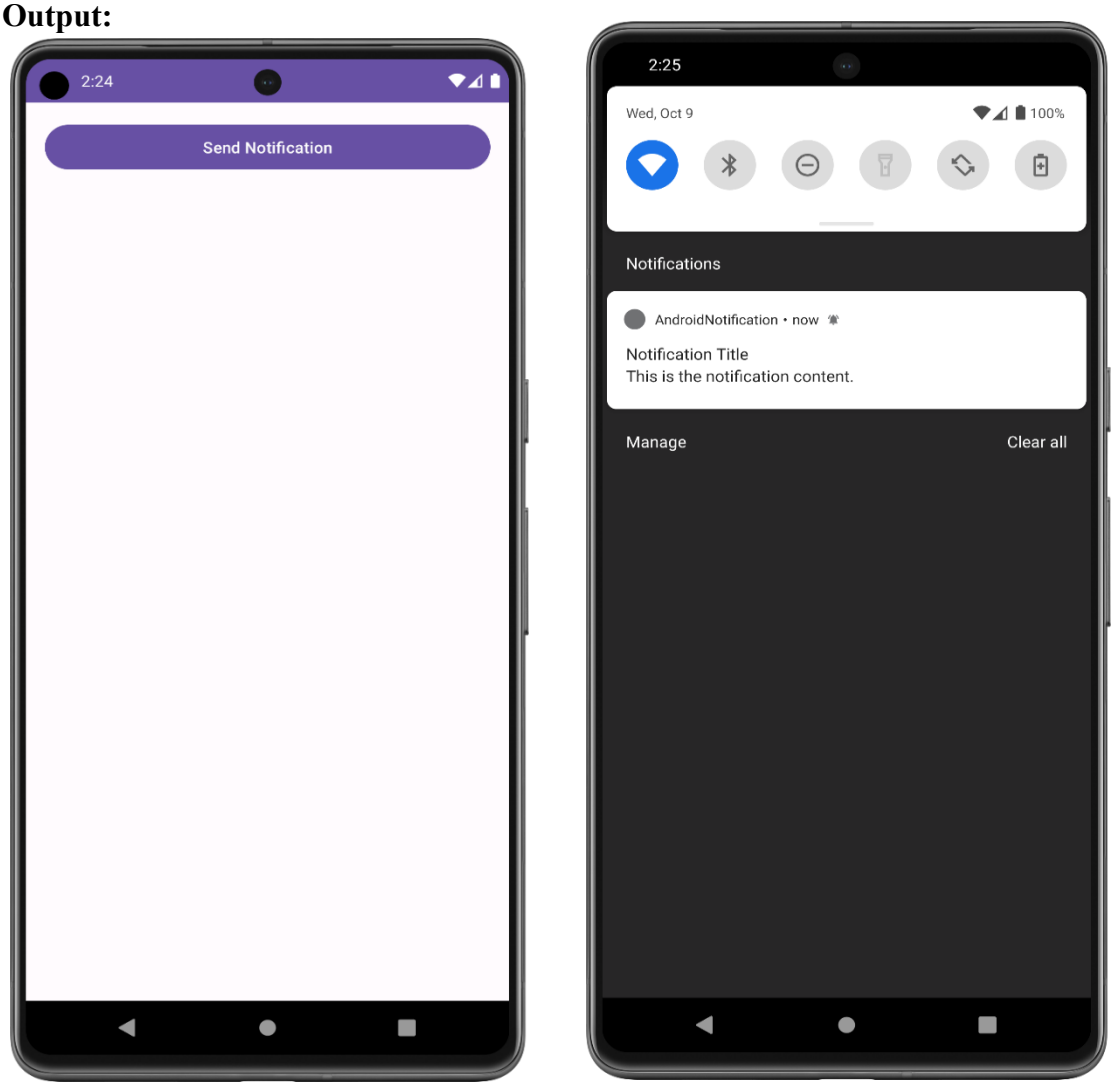
```
package com.example.androidnotification;

import android.app.NotificationChannel;
import android.app.NotificationManager;
import android.app.PendingIntent;
import android.content.Intent;
import android.os.Build;
import android.os.Bundle;
import android.view.View;
import android.widget.Button;
import androidx.appcompat.app.AppCompatActivity;
import androidx.core.app.NotificationCompat;

public class MainActivity extends AppCompatActivity {

    private static final String CHANNEL_ID = "notification_channel";

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

        createNotificationChannel();

        Button buttonNotify = findViewById(R.id.buttonNotify);
        buttonNotify.setOnClickListener(new View.OnClickListener() {
```

```
        @Override
        public void onClick(View v) {
            sendNotification();
        }
    });
}

private void createNotificationChannel() {
    if (Build.VERSION.SDK_INT >= Build.VERSION_CODES.O) {
        NotificationChannel channel = new NotificationChannel(
                CHANNEL_ID,
                "Notification Channel",
                NotificationManager.IMPORTANCE_DEFAULT
        );
        NotificationManager manager = getSystemService(NotificationManager.class);
        manager.createNotificationChannel(channel);
    }
}

private void sendNotification() {
    Intent intent = new Intent(this, MainActivity.class);
    PendingIntent pendingIntent = PendingIntent.getActivity(this, 0, intent, 0);

    NotificationCompat.Builder builder = new NotificationCompat.Builder(this,
CHANNEL_ID)
            .setSmallIcon(R.mipmap.ic_launcher)
            .setContentTitle("Notification Title")
            .setContentText("This is the notification content.")
            .setPriority(NotificationCompat.PRIORITY_DEFAULT)
            .setContentIntent(pendingIntent)
            .setAutoCancel(true);

    NotificationManager notificationManager = (NotificationManager)
getSystemService(NOTIFICATION_SERVICE);
    notificationManager.notify(1, builder.build());
}
}
```

2. **activity_main.xml**

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

    <Button
        android:id="@+id/buttonNotify"
        android:text="@string/send_notification"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"/>
</LinearLayout>
```

Output:



Assignment 21

Tittle: Write an android application using SQLite to create table and perform CRUD operations

Name: Tejas Sunil Mate

Class: MCA-III

Roll No: MC232538

Date:

Remark:

Practical No. 21: Write an android application using SQLite to create table and perform CRUD operations (Example. COURSE table (ID, Name, Duration, Description), perform ADD, UPDATE, DELETE and READ operations)

Solution:

1. **MainActivity.java**

```
package com.example.yourapp;

import androidx.appcompat.app.AppCompatActivity;
import android.os.Bundle;
import android.view.View;
import android.widget.Button;
import android.widget.EditText;
import android.widget.TextView;
import java.util.List;

public class MainActivity extends AppCompatActivity {

    private DBHelper dbHelper;
    private EditText editId, editName, editDuration, editDescription;
    private TextView textViewCourses;

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

        dbHelper = new DBHelper(this);
        editId = findViewById(R.id.editId);
```

```
    editName = findViewById(R.id.editName);
    editDuration = findViewById(R.id.editDuration);
    editDescription = findViewById(R.id.editDescription);
    textViewCourses = findViewById(R.id.textViewCourses);

    Button buttonAdd = findViewById(R.id.buttonAdd);
    Button buttonRead = findViewById(R.id.buttonRead);
    Button buttonUpdate = findViewById(R.id.buttonUpdate);
    Button buttonDelete = findViewById(R.id.buttonDelete);

    buttonAdd.setOnClickListener(view -> addCourse());
    buttonRead.setOnClickListener(view -> readCourses());
    buttonUpdate.setOnClickListener(view -> updateCourse());
    buttonDelete.setOnClickListener(view -> deleteCourse());
}

private void addCourse() {
    String name = editName.getText().toString();
    String duration = editDuration.getText().toString();
    String description = editDescription.getText().toString();
    dbHelper.addCourse(name, duration, description);
    clearFields();
}

private void readCourses() {
    List<String> courses = dbHelper.getAllCourses();
    StringBuilder builder = new StringBuilder();
    for (String course : courses) {
        builder.append(course).append("\n");
    }
    textViewCourses.setText(builder.toString());
}

private void updateCourse() {
    int id = Integer.parseInt(editId.getText().toString());
    String name = editName.getText().toString();
    String duration = editDuration.getText().toString();
    String description = editDescription.getText().toString();
    dbHelper.updateCourse(id, name, duration, description);
    clearFields();
```

```
    }

    private void deleteCourse() {
        int id = Integer.parseInt(editId.getText().toString());
        dbHelper.deleteCourse(id);
        clearFields();
    }

    private void clearFields() {
        editId.setText("");
        editName.setText("");
        editDuration.setText("");
        editDescription.setText("");
    }
}
```

2. DBHelper.java

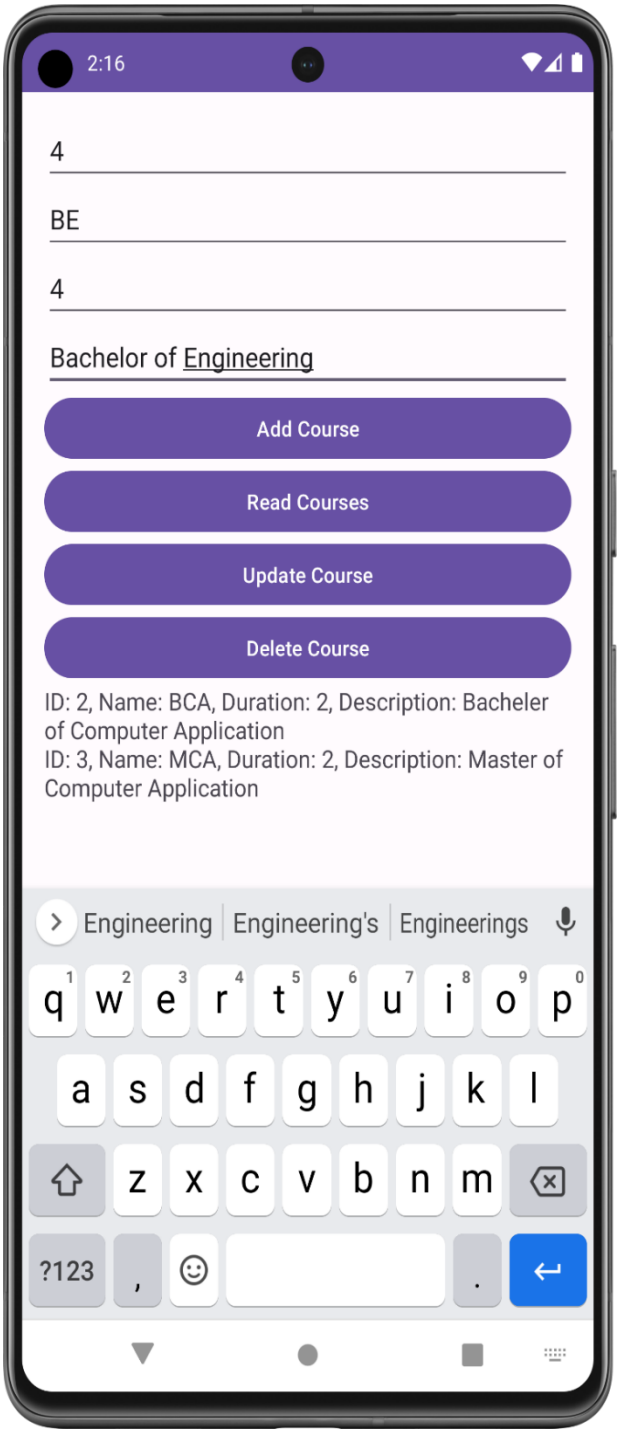
```
package com.example.yourapp;

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

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

public class DBHelper extends SQLiteOpenHelper {

    private static final String DATABASE_NAME = "courses.db";
    private static final int DATABASE_VERSION = 1;
    private static final String TABLE_COURSE = "course";

    private static final String COLUMN_ID = "id";
    private static final String COLUMN_NAME = "name";
    private static final String COLUMN_DURATION = "duration";
    private static final String COLUMN_DESCRIPTION = "description";

    public DBHelper(Context context) {
```

```
        super(context, DATABASE_NAME, null, DATABASE_VERSION);
    }

    @Override
    public void onCreate(SQLiteDatabase db) {
        String CREATE_COURSE_TABLE = "CREATE TABLE " + TABLE_COURSE +
"("
                + COLUMN_ID + " INTEGER PRIMARY KEY AUTOINCREMENT,"
                + COLUMN_NAME + " TEXT,"
                + COLUMN_DURATION + " TEXT,"
                + COLUMN_DESCRIPTION + " TEXT" + ")";
        db.execSQL(CREATE_COURSE_TABLE);
    }

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

    // Add a new course
    public void addCourse(String name, String duration, String description) {
        SQLiteDatabase db = this.getWritableDatabase();
        ContentValues values = new ContentValues();
        values.put(COLUMN_NAME, name);
        values.put(COLUMN_DURATION, duration);
        values.put(COLUMN_DESCRIPTION, description);
        db.insert(TABLE_COURSE, null, values);
        db.close();
    }

    // Read all courses
    public List<String> getAllCourses() {
        List<String> courses = new ArrayList<>();
        SQLiteDatabase db = this.getReadableDatabase();
        Cursor cursor = db.rawQuery("SELECT * FROM " + TABLE_COURSE, null);
        if (cursor.moveToFirst()) {
            do {
                String course = "ID: " + cursor.getInt(0) +
                        ", Name: " + cursor.getString(1) +
```

```
                ", Duration: " + cursor.getString(2) +
                ", Description: " + cursor.getString(3);
            courses.add(course);
        } while (cursor.moveToNext());
    }
    cursor.close();
    db.close();
    return courses;
}

// Update a course
public void updateCourse(int id, String name, String duration, String description) {
    SQLiteDatabase db = this.getWritableDatabase();
    ContentValues values = new ContentValues();
    values.put(COLUMN_NAME, name);
    values.put(COLUMN_DURATION, duration);
    values.put(COLUMN_DESCRIPTION, description);
    db.update(TABLE_COURSE, values, COLUMN_ID + " = ?", new
String[]{String.valueOf(id)});
    db.close();
}

// Delete a course
public void deleteCourse(int id) {
    SQLiteDatabase db = this.getWritableDatabase();
    db.delete(TABLE_COURSE, COLUMN_ID + " = ?", new
String[]{String.valueOf(id)});
    db.close();
}
}
```

3. **activity_main.xml**

Output:


2:16
4
BE
4
Bachelor of Engineering
Add Course
Read Courses
Update Course
Delete Course
ID: 2, Name: BCA, Duration: 2, Description: Bacheler of Computer Application
ID: 3, Name: MCA, Duration: 2, Description: Master of Computer Application
Engineering
Engineering's
Engineerings

Assignment 22

Tittle: Create an Android app, powered by Firebase Realtime database.
Name: Tejas Sunil Mate

Class: MCA-III

Roll No: MC232538

Date:

Remark:

Practical No. 22: Create an Android app, powered by Firebase Realtime database that supports:

- **Adding Data to Firebase Realtime Database,**
- **Retrieving Data from Firebase, and**
- **Deleting data from Firebase database.**

Solution:

1. **User.java**

```
package com.example.q12;
public class User {
   public String name;
   public int age;
   public User() {
   }
   public User(String name, int age) {
      this.name = name;
      this.age = age;
   }}
```

2. **MainActivity.java**

```
package com.example.q12;
import android.os.Bundle;
import android.util.Log;
import android.view.View;
import android.widget.EditText;
import android.widget.TextView;
import android.widget.Toast;
import androidx.appcompat.app.AppCompatActivity;
import com.google.firebase.database.DataSnapshot;
import com.google.firebase.database.DatabaseReference;
```

```
import com.google.firebase.database.FirebaseDatabase;
import java.util.HashMap;
import java.util.Map;
public class MainActivity extends AppCompatActivity {
   // Reference to the Firebase Database
   private static DatabaseReference mDatabase;
   private TextView textViewData;
   // HashMap to store user keys and names for deletion
   private Map<String, String> userKeyMap = new HashMap<>();
   @Override
   protected void onCreate(Bundle savedInstanceState) {
      super.onCreate(savedInstanceState);
      setContentView(R.layout.activity_main);
      // Initialize Firebase Database
      mDatabase = FirebaseDatabase.getInstance().getReference();
      // Initialize TextView for displaying data
      textViewData = findViewById(R.id.textViewData);
   }
   // Method to add data to Firebase
   public void addDataToFirebase(View v) {
      EditText editTextName = findViewById(R.id.editTextName);
      EditText editTextAge = findViewById(R.id.editTextAge);

      String name = editTextName.getText().toString().trim();
      String ageText = editTextAge.getText().toString().trim();
      if (name.isEmpty()) {
         Toast.makeText(this, "Please enter a name",
Toast.LENGTH_SHORT).show();
         return;
      }
      if (ageText.isEmpty()) {
         Toast.makeText(this, "Please enter an age", Toast.LENGTH_SHORT).show();
         return;
      }
      try {
         int age = Integer.parseInt(ageText);
         User user = new User(name, age);
         // Use push() to create a unique key for each user
         mDatabase.child("users").push().setValue(user)
```

```
                .addOnSuccessListener(aVoid -> Toast.makeText(MainActivity.this,
"Data Added", Toast.LENGTH_SHORT).show())
                .addOnFailureListener(e -> Toast.makeText(MainActivity.this, "Failed to
Add Data", Toast.LENGTH_SHORT).show());

        } catch (NumberFormatException e) {
            Toast.makeText(this, "Please enter a valid number for age",
Toast.LENGTH_SHORT).show();
        }
    }
    // Method to retrieve data from Firebase and store keys for deletion
    public void retrieveDataFromFirebase(View v) {
        mDatabase.child("users").get().addOnCompleteListener(task -> {
            if (task.isSuccessful()) {
                StringBuilder retrievedData = new StringBuilder();
                userKeyMap.clear(); // Clear the previous data
                for (DataSnapshot snapshot : task.getResult().getChildren()) {
                    User user = snapshot.getValue(User.class);
                    String key = snapshot.getKey(); // Get the unique key for each user
                    if (user != null && key != null) {
                        retrievedData.append("Name: ").append(user.name)
                                .append(", Age: ").append(user.age)
                                .append("\n");
                        // Store the key and user name for later deletion
                        userKeyMap.put(user.name, key);
                    }
                }
                textViewData.setText(retrievedData.toString());
            } else {
                Log.e("FirebaseData", "Error getting data", task.getException());
                textViewData.setText("Error retrieving data");
            }
        });
    }
    // Method to delete data from Firebase based on user name
    public void deleteDataFromFirebase(View v) {
        EditText editTextName = findViewById(R.id.editTextName);
        String name = editTextName.getText().toString().trim();
        // Validate input
        if (name.isEmpty()) {
```

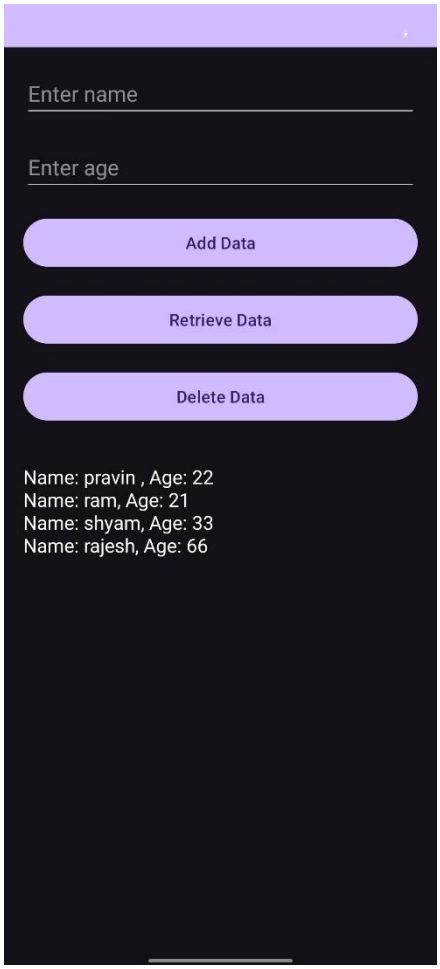
```
            Toast.makeText(this, "Please enter a name to delete",
Toast.LENGTH_SHORT).show();
            return;
        }
        // Get the Firebase key for the user
        String key = userKeyMap.get(name);
        if (key != null) {
            // Delete the user with the retrieved key
            mDatabase.child("users").child(key).removeValue()
                    .addOnSuccessListener(aVoid -> Toast.makeText(MainActivity.this,
"Data Deleted", Toast.LENGTH_SHORT).show())
                    .addOnFailureListener(e -> Toast.makeText(MainActivity.this, "Failed to
Delete Data", Toast.LENGTH_SHORT).show());
        } else {
            Toast.makeText(this, "User not found", Toast.LENGTH_SHORT).show();
        }
    }
}
```

3. **activity_main.xml**

```
<LinearLayout xmlns:android="http://schemas.android.com/apk/res/android"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    android:orientation="vertical"
    android:padding="16dp">
    <!-- Input field for the user's name -->
    <EditText
        android:id="@+id/editTextName"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:hint="Enter name"
        android:layout_marginBottom="16dp"/>
    <!-- Input field for the user's age -->
    <EditText
        android:id="@+id/editTextAge"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:hint="Enter age"
        android:inputType="number"
        android:layout_marginBottom="16dp"/>
```

```
    <!-- Button to add data to Firebase -->
    <Button
        android:id="@+id/btnAddData"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:text="Add Data"
        android:onClick="addDataToFirebase"
        android:layout_marginBottom="16dp"/>

    <!-- Button to retrieve data from Firebase -->
    <Button
        android:id="@+id/btnRetrieveData"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:text="Retrieve Data"
        android:onClick="retrieveDataFromFirebase"
        android:layout_marginBottom="16dp"/>
    <!-- Button to delete data from Firebase -->
    <Button
        android:id="@+id/btnDeleteData"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:text="Delete Data"
        android:onClick="deleteDataFromFirebase"
        android:layout_marginBottom="16dp"/>
    <!-- TextView to display retrieved data -->
    <TextView
        android:id="@+id/textViewData"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:text="Retrieved Data will appear here"
        android:layout_marginTop="16dp"
        android:textSize="16sp"
        android:textColor="@android:color/white"/>
</LinearLayout>
```

Output:


Enter name
Enter age
Add Data
Retrieve Data
Delete Data
Name: pravin , Age: 22
Name: ram, Age: 21
Name: shyam, Age: 33
Name: rajesh, Age: 66

Assignment 23

Tittle: Demonstrate WebView to display the web pages in an android application.
Name: Tejas Sunil Mate

Class: MCA-III

Roll No: MC232538

Date:

Remark:

Practical No. 22: Demonstrate WebView to display the web pages in an android application

Solution:

1. **AndroidManifest.xml**

```
<?xml version="1.0" encoding="utf-8"?>
<manifest xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:tools="http://schemas.android.com/tools">
    <uses-permission android:name="android.permission.INTERNET" />
    <application
        android:allowBackup="true"
        android:dataExtractionRules="@xml/data_extraction_rules"
        android:fullBackupContent="@xml/backup_rules"
        android:icon="@mipmap/ic_launcher"
        android:label="@string/app_name"
        android:roundIcon="@mipmap/ic_launcher_round"
        android:supportsRtl="true"
        android:theme="@style/Theme.Q13"
        tools:targetApi="31">
        <activity
            android:name=".MainActivity"
            android:exported="true">
            <intent-filter>
                <action android:name="android.intent.action.MAIN" />
                <category android:name="android.intent.category.LAUNCHER" />
            </intent-filter>
        </activity>
    </application>
</manifest>
```

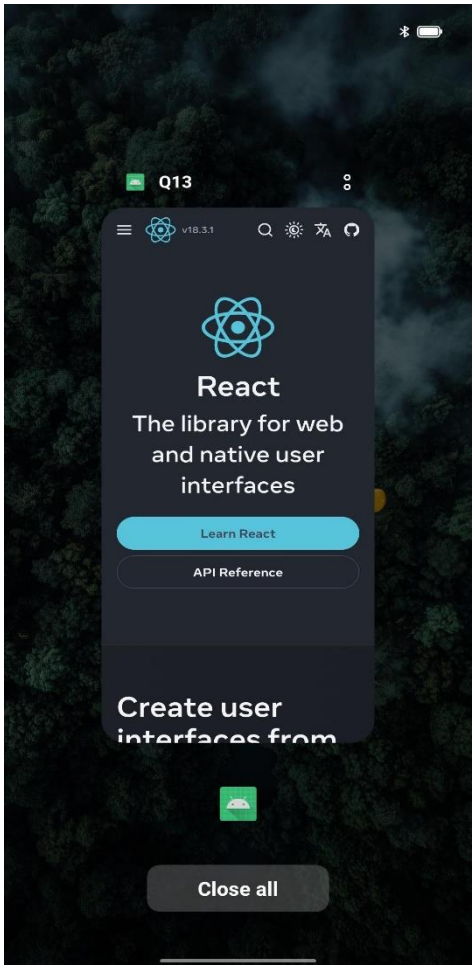
2. **activity_main.xml**

```
<RelativeLayout xmlns:android="http://schemas.android.com/apk/res/android"
   android:layout_width="match_parent"
   android:layout_height="match_parent">
   <WebView
      android:id="@+id/webview"
      android:layout_width="match_parent"
      android:layout_height="match_parent" />
</RelativeLayout>
```

3. **MainActivity.java**

```
package com.example.q13;
import android.os.Bundle;
import android.webkit.WebSettings;
import android.webkit.WebView;
import android.webkit.WebViewClient;
import androidx.appcompat.app.AppCompatActivity;
public class MainActivity extends AppCompatActivity {

   @Override
   protected void onCreate(Bundle savedInstanceState) {
      super.onCreate(savedInstanceState);
      setContentView(R.layout.activity_main);
      WebView webView = findViewById(R.id.webview);
      webView.setWebViewClient(new WebViewClient());
      WebSettings webSettings = webView.getSettings();
      webSettings.setJavaScriptEnabled(true);
      webView.loadUrl("https://www.react.dev");
   }
}
```

Output:


Q13
v18.3.1
React
The library for web and native user interfaces
Learn React
API Reference
Create user
Close all

Assignment 24

Tittle: Write an android app to write JSON data into a file and read JSON data from created file.
Name: Tejas Sunil Mate

Class: MCA-III

Roll No: MC232538

Date:

Remark:

Practical No. 22: Write an android app to write JSON data into a file and read JSON data from created file.

Solution:

1. **activity_main.xml**

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

    <Button
        android:id="@+id/writeJsonButton"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="Write JSON to File"
        android:layout_centerHorizontal="true"
        android:layout_marginTop="100dp"/>

    <Button
        android:id="@+id/readJsonButton"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="Read JSON from File"
        android:layout_below="@id/writeJsonButton"
        android:layout_centerHorizontal="true"
        android:layout_marginTop="20dp"/>

    <TextView
        android:id="@+id/textView"
        android:layout_width="wrap_content"
```

```
        android:layout_height="wrap_content"
        android:text="JSON Data will appear here"
        android:layout_below="@id/readJsonButton"
        android:layout_marginTop="20dp"
        android:layout_centerHorizontal="true"/>
</RelativeLayout>
```

2. **MainActivity.java**

```
package com.example.q14;
import android.os.Bundle;
import android.view.View;
import android.widget.Button;
import android.widget.TextView;
import androidx.appcompat.app.AppCompatActivity;
import org.json.JSONException;
import org.json.JSONObject;
import java.io.FileInputStream;
import java.io.FileOutputStream;
import java.io.IOException;

public class MainActivity extends AppCompatActivity {
    private final String FILE_NAME = "data.json";
    private TextView textView;
    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_main);
        Button writeJsonButton = findViewById(R.id.writeJsonButton);
        Button readJsonButton = findViewById(R.id.readJsonButton);
        textView = findViewById(R.id.textView);
        writeJsonButton.setOnClickListener(new View.OnClickListener() {
            @Override
            public void onClick(View v) {
                writeJsonToFile();
            }
        });
        readJsonButton.setOnClickListener(new View.OnClickListener() {
            @Override
            public void onClick(View v) {
                readJsonFromFile();
```

```
        }
    });
}
private void writeJsonToFile() {
    JSONObject jsonObject = new JSONObject();
    try {
        jsonObject.put("name", "John Doe");
        jsonObject.put("age", 30);
        jsonObject.put("email", "johndoe@example.com");
        jsonObject.put("phone", "123456789");

        String jsonString = jsonObject.toString();
        FileOutputStream fos = openFileOutput(FILE_NAME, MODE_PRIVATE);
        fos.write(jsonString.getBytes());
        fos.close();

        textView.setText("JSON data written to file.");
    } catch (JSONException | IOException e) {
        e.printStackTrace();
    }
}
private void readJsonFromFile() {
    try {
        FileInputStream fis = openFileInput(FILE_NAME);
        int size = fis.available();
        byte[] buffer = new byte[size];
        fis.read(buffer);
        fis.close();

        String jsonString = new String(buffer, "UTF-8");
        textView.setText(jsonString);
    } catch (IOException e) {
        e.printStackTrace();
    }
}
}
```

Output:


Write JSON to File
Read JSON from File
{"name":"John Doe","age":30,"email":"johndoe@example
.com","phone":"123456789"}

Assignment 25

Tittle: Write an application to display a PDF as an image in React app using URL
Name: Tejas Sunil Mate

Class: MCA-III

Roll No: MC232538

Date:

Remark:

Practical No. 25: Write an application to display a PDF as an image in React app using URL

Solution:

1. **app.js**

```
import React from 'react';
import { StyleSheet, View, Dimensions } from 'react-native';
import Pdf from 'react-native-pdf';
import * as FileSystem from 'expo-file-system';


const App = () => {
  // You can specify either a local file or a remote URL
  const pdfUrl = 'https://www.w3.org/WAI/ER/tests/xhtml/testfiles/resources/pdf/dummy.pdf';
  const [pdfFile, setPdfFile] = React.useState(null);


  // Optional: Download PDF to local file system
  React.useEffect(() => {
    const downloadPdf = async () => {
      const downloadResult = await FileSystem.downloadAsync(
```

```
      pdfUrl,
      FileSystem.documentDirectory + 'sample.pdf'
    );
    setPdfFile(downloadResult.uri); // Set the local PDF URI
  };
  downloadPdf();
}, []);

return (
  <View style={styles.container}>
    {pdfFile && (
      <Pdf
        source={{ uri: pdfFile }} // use local file URI here
        onLoadComplete={(numberOfPages) => {
          console.log(`Number of pages: ${numberOfPages}`);
        }}
        onPageChanged={(page) => {
          console.log(`Current page: ${page}`);
        }}
        onError={(error) => {
          console.log(error);
        }}
        style={styles.pdf}
      />
    )}
```

```
    </View>
  );
};
const styles = StyleSheet.create({
  container: {
    flex: 1,
    justifyContent: 'center',
    alignItems: 'center',
  },
  pdf: {
    flex: 1,
    width: Dimensions.get('window').width,
    height: Dimensions.get('window').height,
  },
});
export default App;
```

Assignment 26

Tittle: Develop simple flutter application to open a browser using Android SDK
Name: Tejas Sunil Mate

Class: MCA-III

Roll No: MC232538

Date:

Remark:

Practical No. 26: Develop simple flutter application to open a browser using Android SDK

Solution:

1. **AndroidManifiest.xml**

```
<manifest xmlns:android="http://schemas.android.com/apk/res/android">
  <uses-permission android:name="android.permission.INTERNET" />
  <application
    android:label="myapp"
    android:name="${applicationName}"
    android:icon="@mipmap/ic_launcher">
    <activity
      android:name=".MainActivity"
      android:exported="true"
      android:launchMode="singleTop"
      android:taskAffinity=""
      android:theme="@style/LaunchTheme"

android:configChanges="orientation|keyboardHidden|keyboard|screenSize|smallestSc
reenSize|locale|layoutDirection|fontScale|screenLayout|density|uiMode"
      android:hardwareAccelerated="true"
      android:windowSoftInputMode="adjustResize">
      <meta-data
        android:name="io.flutter.embedding.android.NormalTheme"
        android:resource="@style/NormalTheme"
        />
      <intent-filter>
        <action android:name="android.intent.action.MAIN"/>
        <category android:name="android.intent.category.LAUNCHER"/>
      </intent-filter>
```

```
        </activity>
        <meta-data
            android:name="flutterEmbedding"
            android:value="2" />
    </application>
    <queries>
        <intent>
            <action android:name="android.intent.action.PROCESS_TEXT"/>
            <data android:mimeType="text/plain"/>
        </intent>
    </queries>
</manifest>
```

2. **main.dart**

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

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

class MyApp extends StatelessWidget {
  @override
  Widget build(BuildContext context) {
    return MaterialApp(
      home: Scaffold(
        appBar: AppBar(title: Text('Open Browser Example')),
        body: Center(
          child: ElevatedButton(
            onPressed: () {
              openBrowser('https://flutter.dev');
            },
            child: Text('Open Flutter Website'),
          ),
        ),
      ),
    );
  }

  Future<void> openBrowser(String url) async {
```

```
    Uri uri = Uri.parse(url);
    if (await canLaunchUrl(uri)) {
      await launchUrl(uri, mode: LaunchMode.externalApplication);
    } else {
      throw 'Could not launch $url';
    }
  }
}
```

3. **pubspec.yml**

```
name: myapp
description: "A new Flutter project."
publish_to: 'none'
version: 1.0.0+1
environment:
  sdk: '>=3.4.3 <4.0.0'
dependencies:
  flutter:
    sdk: flutter
  url_launcher: ^6.1.6
  cupertino_icons: ^1.0.6
dev_dependencies:
  flutter_test:
    sdk: flutter
  flutter_lints: ^3.0.0
flutter:
  uses-material-design: true
```

Output: