Experiment 3: Simple Tap Counter

1. Objective

Create an app that counts the number of times a button is tapped. This experiment will reinforce understanding of event listeners, UI updates, and maintaining state within an app.

2. Steps to Complete the Application

1. Design the Layout:

   Open activity_main.xml.

   Use a ConstraintLayout or LinearLayout for simplicity.

   Add a Button for the tap action.

   Include a TextView to display the counter.

2. Configure the Button and TextView:

   Set up the button with an id and text indicating the tap action.

   Initialize the TextView with text to display the initial counter state (e.g., "Taps: 0").

```
<Button
    android:id="@+id/btnTap"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:text="Tap me!" />

<TextView
```

```
    android:id="@+id/tvCounter"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:text="Taps: 0"
    android:textSize="24sp" />
```

3. Write the Logic in MainActivity:

   In MainActivity.java, define a member variable to hold the tap count.

   Set up an OnClickListener for the button to increment the tap count and update the TextView on each tap.

```
public class MainActivity extends AppCompatActivity {

    Button btnTap;
    TextView tvCounter;
    int tapCount = 0;

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

        btnTap = findViewById(R.id.btnTap);
        tvCounter = findViewById(R.id.tvCounter);

        btnTap.setOnClickListener(new View.OnClickListener() {
            @Override
            public void onClick(View v) {
                tapCount++;  // Increment the tap count
                tvCounter.setText("Taps: " + tapCount); // Update the TextView
            }
        });
    }
}
```

3. Explanation

The Button is set with an OnClickListener which is a callback that executes when the button is clicked.

Inside the OnClickListener, we increment the tapCount variable by 1 for each tap.

The TextView (tvCounter) is updated with the new tapCount value each time the button is clicked, providing real-time feedback to the user.

This exercise demonstrates how to handle user interactions and dynamically update the UI in response to those interactions.