

## Practical – 7

Here's a basic implementation of a temperature converter application using Android and Kotlin. The app will allow the user to convert temperatures between Celsius and Fahrenheit.

### 1. Create a New Android Project

Open Android Studio.

Create a new project with Empty Activity.

Name your project, for example, TemperatureConverter.

Choose Kotlin as the programming language.

---

### 2. Layout File (activity\_main.xml)

Replace the content of your activity\_main.xml file with the following:

<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/inputTemperature"

android:layout\_width="match\_parent"

android:layout\_height="wrap\_content"

android:hint="Enter temperature"

android:inputType="numberDecimal"

android:layout\_marginBottom="16dp"/>

<RadioGroup

```
    android:id="@+id/radioGroup"  
    android:layout_width="wrap_content"  
    android:layout_height="wrap_content"  
    android:orientation="horizontal"  
    android:layout_marginBottom="16dp">
```

<RadioButton

```
    android:id="@+id/radioToCelsius"  
    android:layout_width="wrap_content"  
    android:layout_height="wrap_content"  
    android:text="To Celsius" />
```

<RadioButton

```
    android:id="@+id/radioToFahrenheit"  
    android:layout_width="wrap_content"  
    android:layout_height="wrap_content"  
    android:text="To Fahrenheit" />
```

</RadioGroup>

<Button

```
    android:id="@+id/convertButton"  
    android:layout_width="match_parent"  
    android:layout_height="wrap_content"  
    android:text="Convert"  
    android:layout_marginBottom="16dp"/>
```

```
<TextView
    android:id="@+id/resultTextView"
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:text="Result will be displayed here"
    android:textSize="16sp" />
</LinearLayout>
```

Kotlin file - \_\_\_\_\_

### 3. Main Activity (MainActivity.kt)

Replace the content of your MainActivity.kt file with the following:

```
package com.example.temperatureconverter
```

```
import android.os.Bundle
```

```
import android.widget.*
```

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

```
class MainActivity : AppCompatActivity() {
    override fun onCreate(savedInstanceState: Bundle?) {
        super.onCreate(savedInstanceState)
        setContentView(R.layout.activity_main)

        val inputTemperature = findViewById<EditText>(R.id.inputTemperature)
        val radioToCelsius = findViewById<RadioButton>(R.id.radioToCelsius)
```

```
val radioToFahrenheit =
findViewById<RadioButton>(R.id.radioToFahrenheit)

val convertButton = findViewById<Button>(R.id.convertButton)
val resultTextView = findViewById<TextView>(R.id.resultTextView)

convertButton.setOnClickListener {
    val input = inputTemperature.text.toString()

    if (input.isEmpty()) {
        Toast.makeText(this, "Please enter a temperature",
Toast.LENGTH_SHORT).show()
        return@setOnClickListener
    }

    val temperature = input.toDouble()
    val result = if (radioToCelsius.isChecked) {
        (temperature - 32) * 5 / 9 // Convert to Celsius
    } else if (radioToFahrenheit.isChecked) {
        (temperature * 9 / 5) + 32 // Convert to Fahrenheit
    } else {
        null
    }

    result?.let {
        resultTextView.text = "Converted Temperature: %.2f".format(it)
    } ?: run {
```

```
        Toast.makeText(this, "Please select a conversion type",
        Toast.LENGTH_SHORT).show()

    }

}

}
```

---

#### 4. Run the Application

Connect an Android device or use an emulator.

Build and run the application.

Enter a temperature, select a conversion type (To Celsius or To Fahrenheit), and press Convert to see the result.

---

#### Features

Converts temperatures between Celsius and Fahrenheit.

User-friendly interface with input validation.

Displays the result with two decimal precision.

Feel free to enhance this app further by adding additional features or refining the UI.