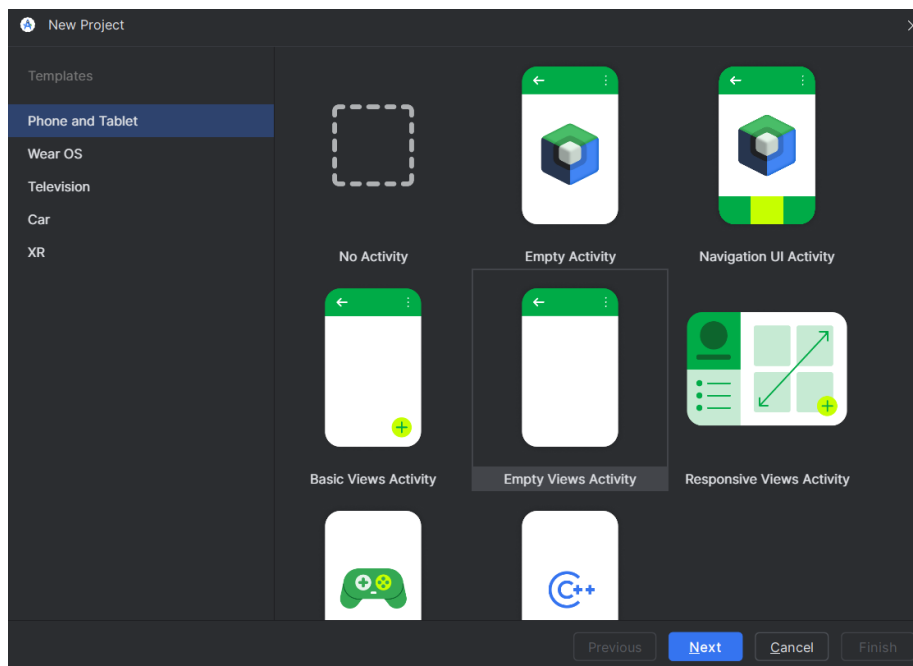


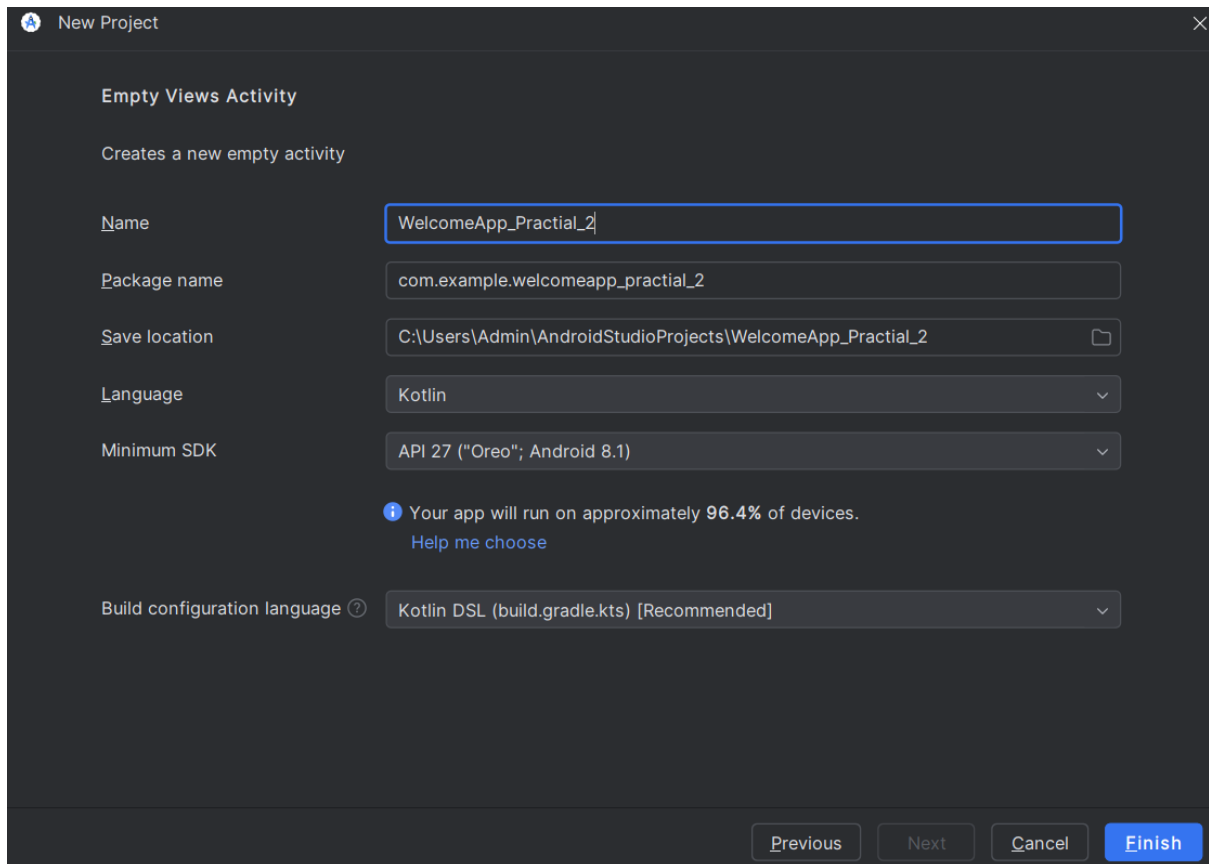
2. Build a Simple Input–Output App (EditText + Validation) Topics covered: EditText, TextView, Button, onClickListener, simple validation Task: • User enters name in EditText • On button click, display “Welcome, <name>!” in TextView • Add validation: if name is empty → show Toast “Please enter a name!”

Step 1:

Go to File→ New→ New Project→Empty Views Activity→ Next



Click on Project name as “WelcomeApp_Practial_2” → Finish



Step 3: Open the activity_main.xml

1. Go to
app → src → main → res → layout → activity_main.xml
2. Click it.

You will see **Design / Code / Split** tabs at the top.

Click **Split** (so you can see both code + preview) or Click (Alt+Shift+Left). You will see xml code.

Replace it with this given code:

```
<?xml version="1.0" encoding="utf-8"?>
<androidx.constraintlayout.widget.ConstraintLayout
    android:id="@+id/main"
    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:padding="24dp">

    <EditText
        android:id="@+id/nameEditText"
        android:layout_width="0dp"
        android:layout_height="wrap_content"
        android:hint="Enter your name"
        android:inputType="textPersonName"
        app:layout_constraintTop_toTopOf="parent"
```

```

        app:layout_constraintStart_toStartOf="parent"
        app:layout_constraintEnd_toEndOf="parent"/>

<Button
    android:id="@+id/welcomeButton"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:text="Say Welcome"
    app:layout_constraintTop_toBottomOf="@id/nameEditText"
    app:layout_constraintStart_toStartOf="parent"
    app:layout_constraintEnd_toEndOf="parent"
    android:layout_marginTop="16dp"/>

<TextView
    android:id="@+id/resultTextView"
    android:layout_width="0dp"
    android:layout_height="wrap_content"
    android:text=""
    android:textSize="18sp"
    android:textStyle="bold"
    android:gravity="center"
    app:layout_constraintTop_toBottomOf="@id/welcomeButton"
    app:layout_constraintStart_toStartOf="parent"
    app:layout_constraintEnd_toEndOf="parent"
    android:layout_marginTop="24dp"/>

</androidx.constraintlayout.widget.ConstraintLayout>

```

Step 4: Open MainActivity.kt

Find this file:

app → src → main → java → com.example.WecomApp_Practial_1 → MainActivity.kt

Open it.

Add the import at first:

```

import android.widget.Button
import android.widget.EditText
import android.widget.TextView
import android.widget.Toast

```

Then after the:

```

    insets
}

```

Add this code:

```

// 1. Connect XML views to Kotlin
val nameEditText = findViewById<EditText>(R.id.nameEditText)
val welcomeButton = findViewById<Button>(R.id.welcomeButton)
val resultTextView = findViewById<TextView>(R.id.resultTextView)

// 2. Set click listener
welcomeButton.setOnClickListener {

```

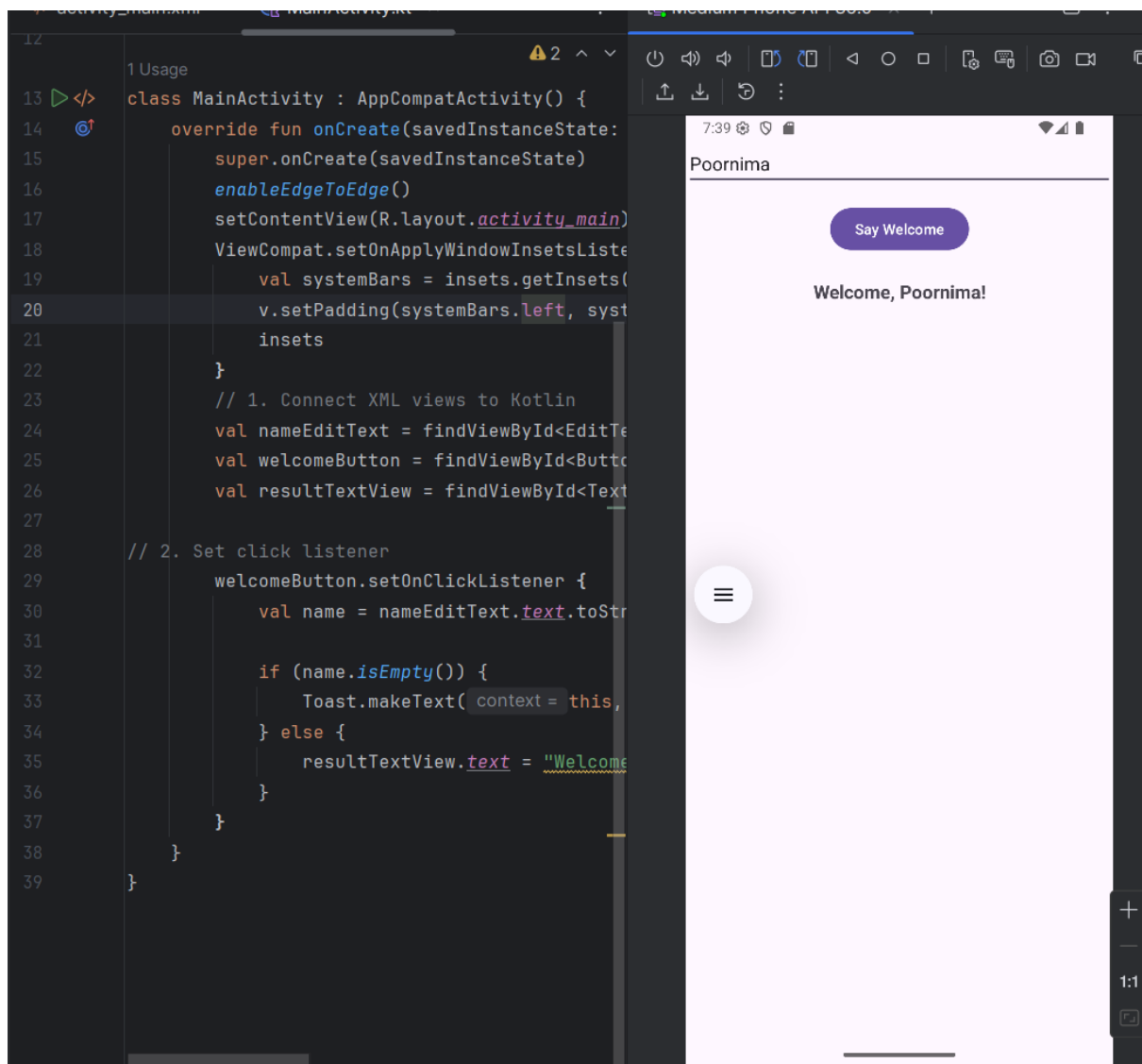
```

        val name = nameEditText.text.toString().trim()

        if (name.isEmpty()) {
            Toast.makeText(this, "Please enter a name!",
                Toast.LENGTH_SHORT).show()
        } else {
            resultTextView.text = "Welcome, $name!"
        }
    }
}

```

Output (Virtual Phone):




Step 6: Connection this app in Actual phone (your phone)

1. Open your YOUR phone

Unlock it.

2. Open the “Settings” app

It's a gear  icon (on your home screen or app drawer).

3. Scroll and tap “About device”

On OPPO, it is usually at the **top** of settings.

4. Tap “Version”

Inside About Device.

5. Tap “Build number” 7 times

Your phone will say:

“You are now a developer!”

6. Go back → Additional Settings

Scroll and find:

Additional Settings OR

7. Search and Tap “Developer Options”

Now you will see:

Developer Options (only appears after enabling it)

8. Turn ON USB Debugging

Scroll down and enable:

 **USB Debugging**

Tap **OK** to allow it.

Now connect you're YOUR phone to your laptop

Use your USB cable → as soon as you connect, your phone will show:

“Allow USB debugging?”

Tap:

✓ **Always allow from this computer**

✓ **Allow**

✓ **USB Mode MUST be “File Transfer”**

Once your Phone is connected

✓ Select your phone:

At the top device dropdown, choose for example:

👉 OPPO CPH2557

(It should show a green phone icon)

✓ Then click the green RUN  button.

Your app will install and start on your phone.

Output (Real phone):

9:07



Poornima 🙇

Say Welcome

Welcome, Poornima 🙇!

