

**Tribhuvan University**  
**Faculty of Humanities and Social Sciences**



**Lab report on:**  
**Mobile Programming Lab 1:**

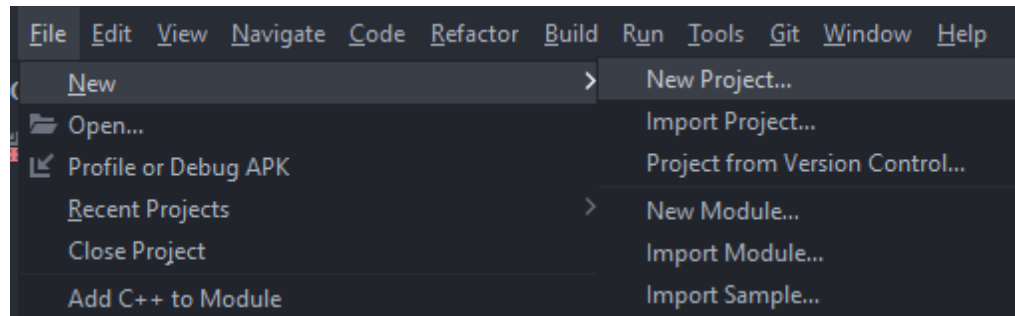
**Submitted to:**  
Mr. Diwas Khatri ,  
Department of Computer Application,  
Himalaya College of Engineering,  
Chyasal,Lalitpur

**Submitted by:**  
Sujal Gurung  
Roll no: 34  
BCA III/II  
July 3, 2024

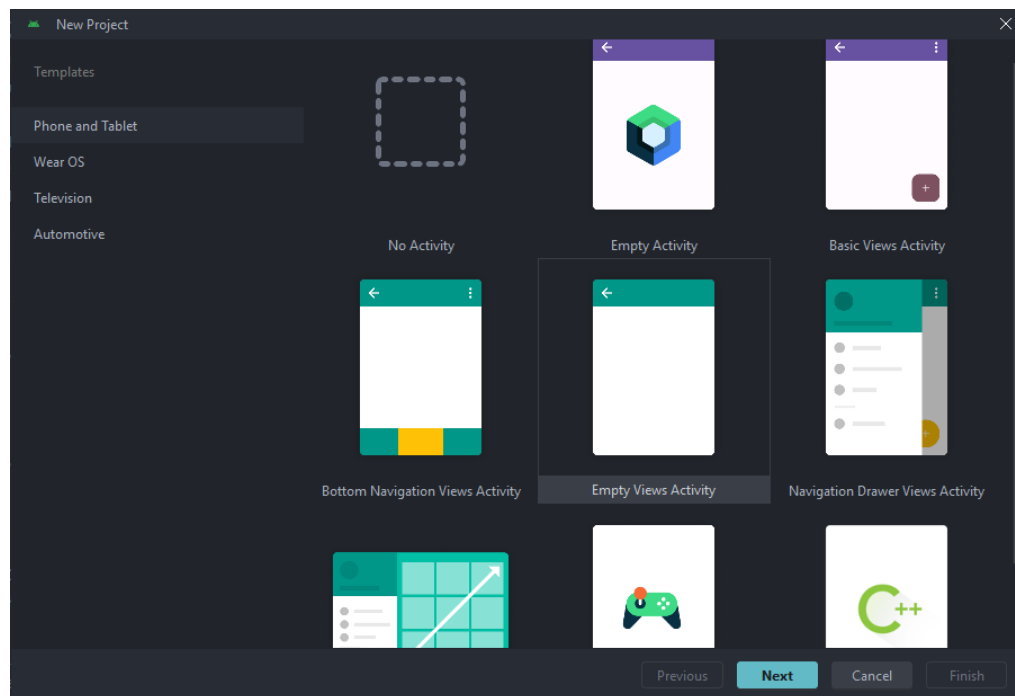
# 1 Demonstrate the setup & installation of Android Project with Java [2 marks]

## 1.1 Project setup steps

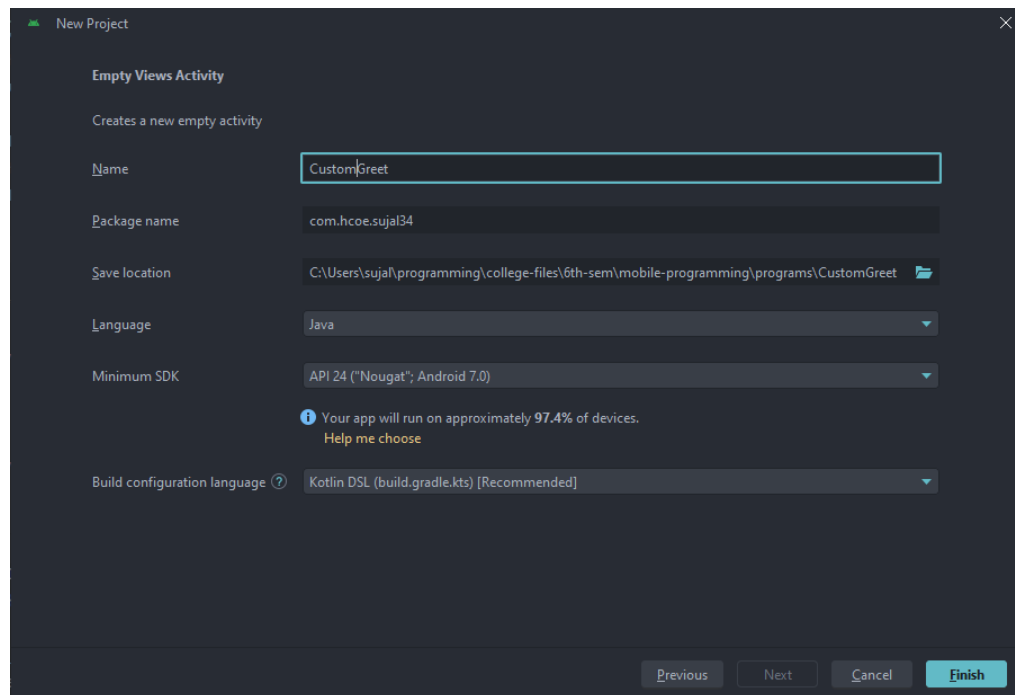
1. From the top menu, go to File → New → New Project...



2. From the given project templates, select Empty Views Activity to use the old XML-based View layouts in your project.



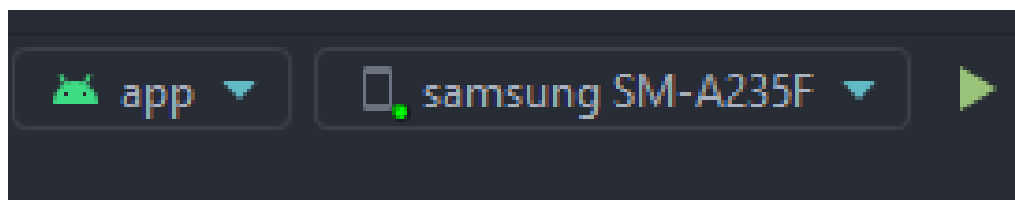
3. Fill in your app details, such as its display name, package (for uniquely identifying it). Make sure to choose Java as the language



4. Done. Android Studio will auto-generate the required files and code to create a basic app

## 1.2 App install steps

1. Apps can be installed on an actual android device or a virtual emulator.
  - To use an actual device, connect it to the computer using a USB cable. Ensure USB debugging & Install apps over USB are enabled inside the device's Developer Settings.
  - To use a virtual emulator, configure it using the top right ribbon beside the run button
2. Use the green Run button to install and open the app



## 2 Develop an android application that prints "Hello World" on the bottom of the page [4 marks]

### 2.1 MainActivity.java

```
package com.hcoe.sujal34;

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;

public class MainActivity extends AppCompatActivity {

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

### 2.2 activity\_main.xml

```
<?xml version="1.0" encoding="utf-8"?>
<LinearLayout
    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:gravity="bottom|center"
    tools:context=".MainActivity">

    <TextView
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="Hello World!"
        android:layout_marginBottom="20dp" />

</LinearLayout>
```



## 2.3 Output

**3 Develop an android application w/ 2 activities: MainActivity & GreetActivity. [5 marks]**

MainActivity should contain a textbox and button w/ label "submit".

When clicked on submit, GreetActivity should open w/ message "Hello {name}" where name is submitted from MainActivity.