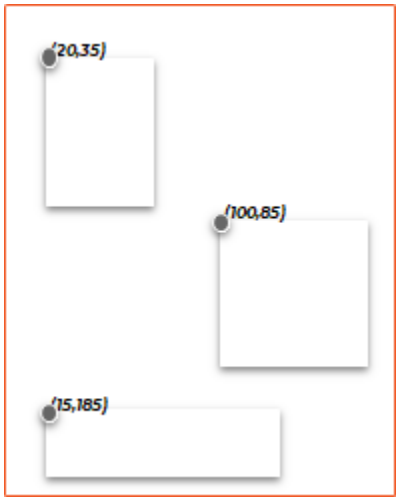
Lab 6: Android AbsoluteLayout

Introduction

Using x/y coordinates, an Absolute Layout allows you to specify the precise placements of its offspring. Compared to other types of layouts that do not contain absolute placement, absolute layouts are less adaptable and more difficult to maintain.

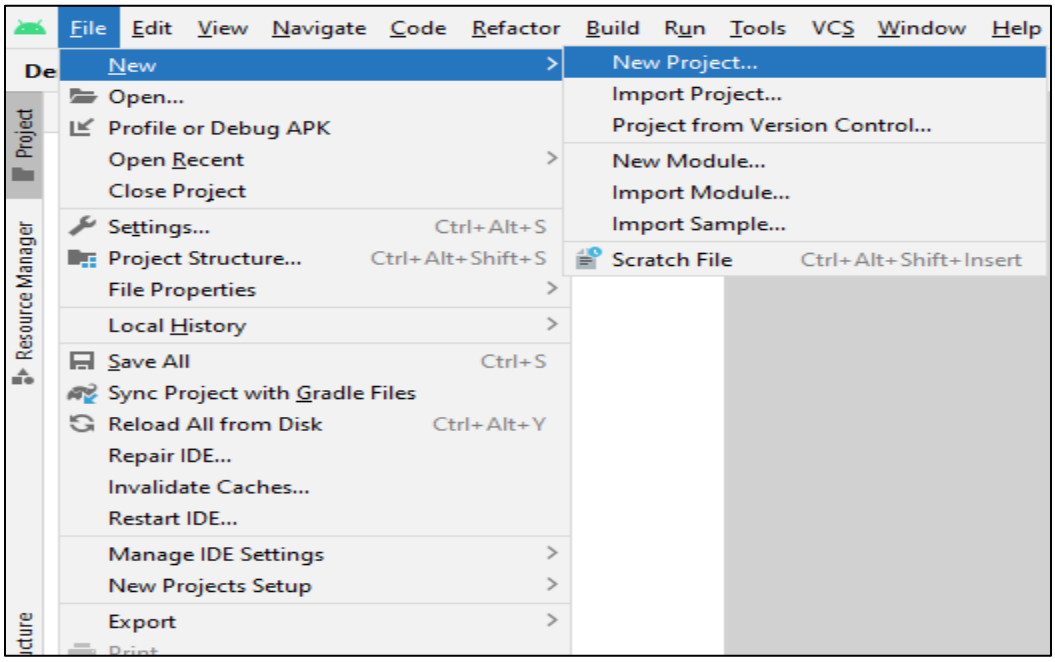


Attribute	Description
id	Used to identify uniquely
layout_x	Used to specify X Coordinate
layout_y	Used to specify Y Coordinate

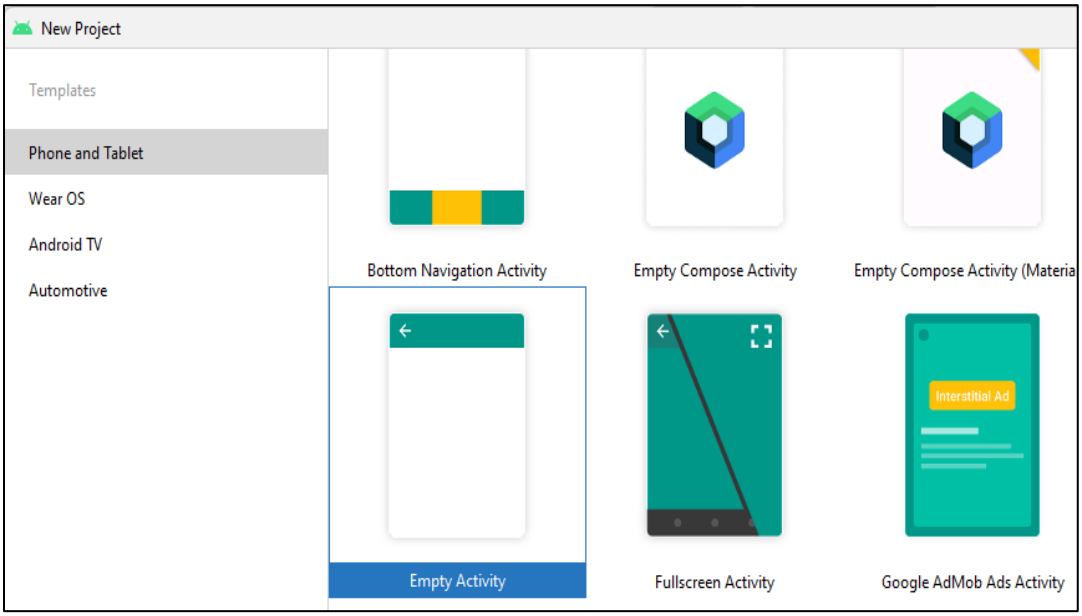
Let’s get Started

You will be guided through a series of easy stages in this exercise to design your own Android application utilising Absolute layout.

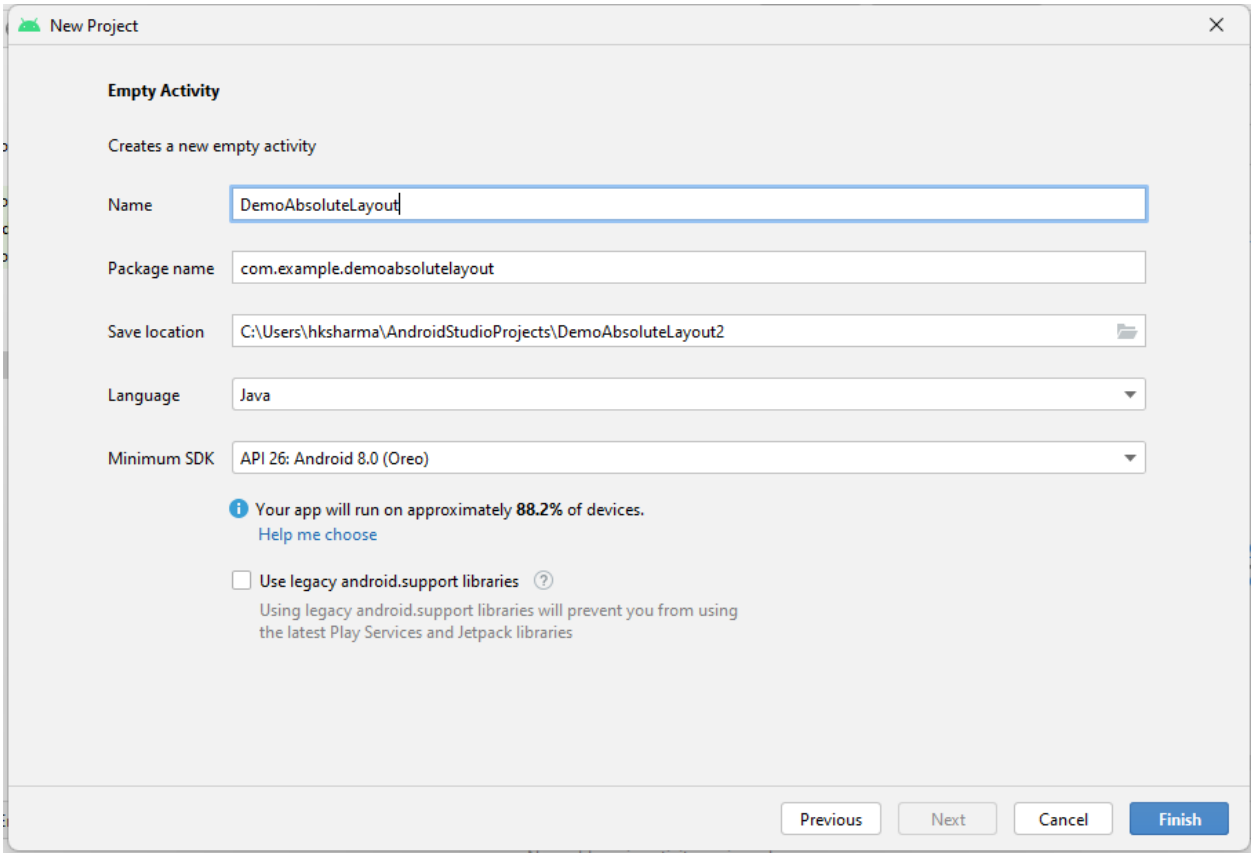
Step 1: Create a New Project in Android Studio as shown below



Step 2: Select Empty Activity as shown below



Step 3: Provide a Project Name as shown below



Step 4: Update MainActivity.java as per the code given below

```
package com.example.demoabsolutelayout;
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);
    }
}
```

Step 5: Update activity_main.xml as per the code given below

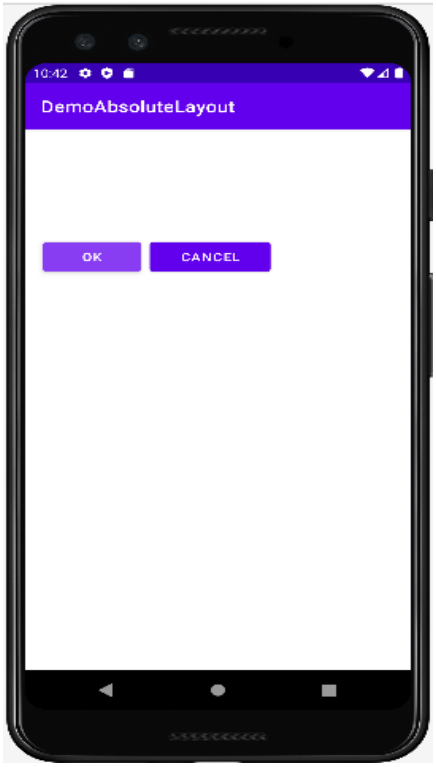
```
<?xml version="1.0" encoding="utf-8"?>
<AbsoluteLayout xmlns:android="http://schemas.android.com/apk/res/android"
    android:layout_width="fill_parent"
    android:layout_height="fill_parent">
```

```
    <Button
        android:layout_width="100dp"
        android:layout_height="wrap_content"
        android:text="OK"
        android:layout_x="50px"
        android:layout_y="361px" />

    <Button
        android:layout_width="123dp"
        android:layout_height="wrap_content"
        android:layout_x="127dp"
        android:layout_y="361px"
        android:text="Cancel" />

</AbsoluteLayout>
```

Step 6: Check Output on Android Emulator and it should look like as given below



Voila!! We have successfully completed this lab.