

通訊網路實驗

Android & Python Programming
Android Studio

Dept. of Electrical and Computer Engineering (ECE)
National Yang Ming Chiao Tung University

評分標準 & 注意事項

- 出席 30%
- Demo 30%
- 結報 40%
 - e3上有學習單
 - 檔名：學號_姓名_Labx.pdf
 - 交pdf到對應的資料夾中 (期限一週)

課程大綱

- Android Studio Introduction
- Android Studio 入門
- Android Studio Button應用

Demo項目

- Q1 : Button小程式
- Q2 : 改變TextView字體大小
- Q3 : 進階Button小程式
 - ▣ Q1、Q2可建在同一個專案內，Q3自己一個新的專案

Introduction

- 為Android平台開發程式的整合式開發環境
- 供開發者免費使用，在Windows、OS X、Linux平台上均可執行
- 基於Gradle的建構
- 內建Android SDK和AVD管理器、佈局編輯器等



Introduction-元件(View)

- 一個介面上可以有許多個元件，每個元件搭配java code的指令可以完成各類功能
- 常用的元件有TextView、ImageView、Button等

Introduction-xml

- 用來設定畫面以及畫面上元件的屬性



```
<?xml version="1.0" encoding="utf-8"?>
<androidx.constraintlayout.widget.ConstraintLayout 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:layout_width="match_parent"
    android:layout_height="match_parent"
    tools:context=".MainActivity">

    <TextView 加入一個顯示文字的介面元件
        android:layout_width="wrap_content" 文字介面元件的寬度及高度，wrap content代表依
        android:layout_height="wrap_content" 照內容決定，match parent代表填滿可顯示的空間
        android:text="Hello World!"
        app:layout_constraintBottom_toBottomOf="parent"
        app:layout_constraintLeft_toLeftOf="parent"
        app:layout_constraintRight_toRightOf="parent"
        app:layout_constraintTop_toTopOf="parent" /> 代表此元件是沒有結尾標籤的單獨標籤

</androidx.constraintlayout.widget.ConstraintLayout>
```

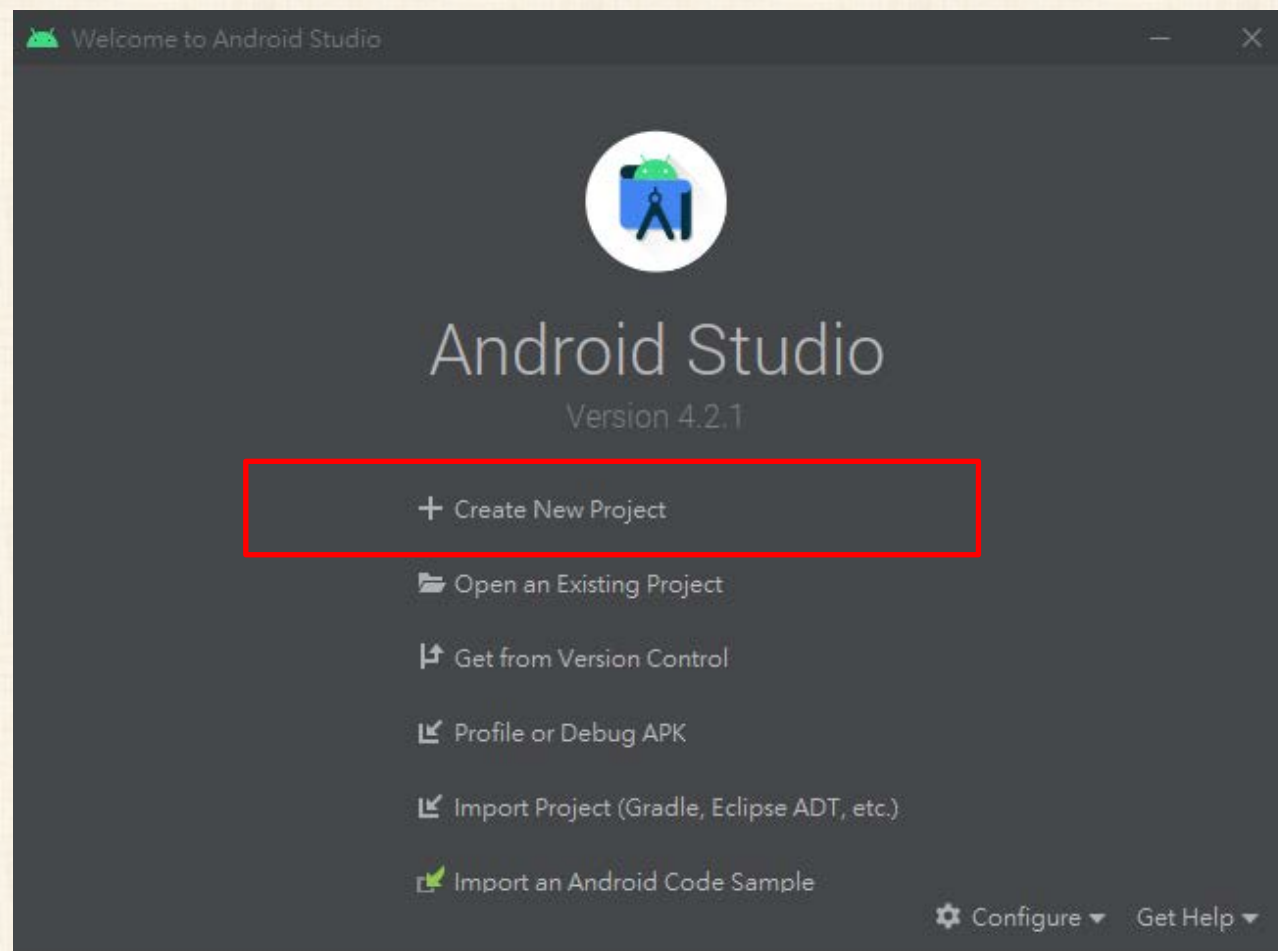

入門

- Build your first app

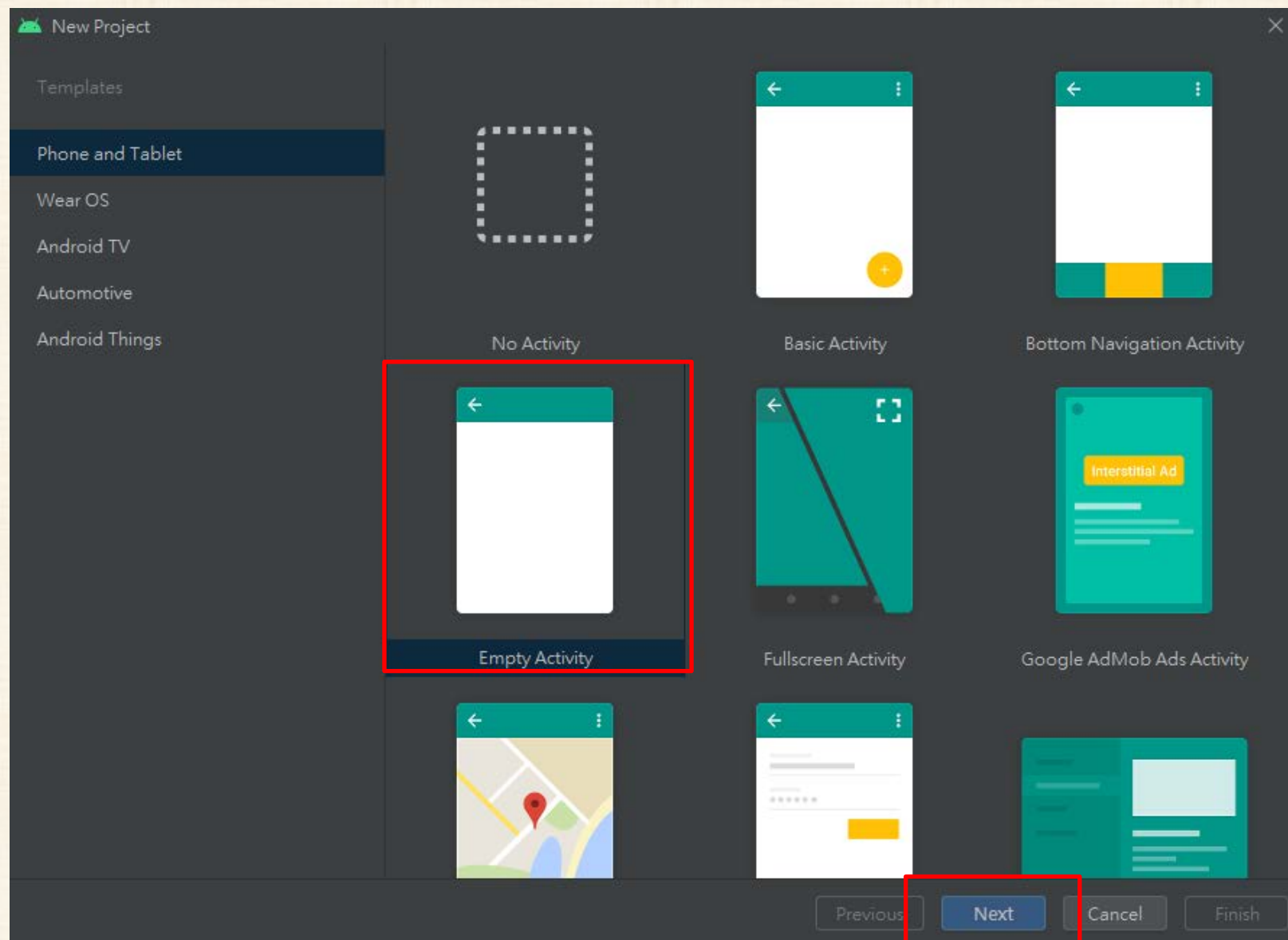
- 步驟：

- 1. Create a new Android Studio project
- 2. !! App名稱請包含自己的學號 !! (App+"學號")
 - 不能有中文字
- 3. AVD建立(Virtual Device)
- 4. RUN

Step 1



也可以用File > new > project 建立新的project



Step 2

New Project

Empty Activity

Creates a new empty activity

Name App309000000

Package name com.example.app309000000

Save location C:\Users\HY\AndroidStudioProjects\App309000000

Language Java

Minimum SDK API 16: Android 4.1 (Jelly Bean)

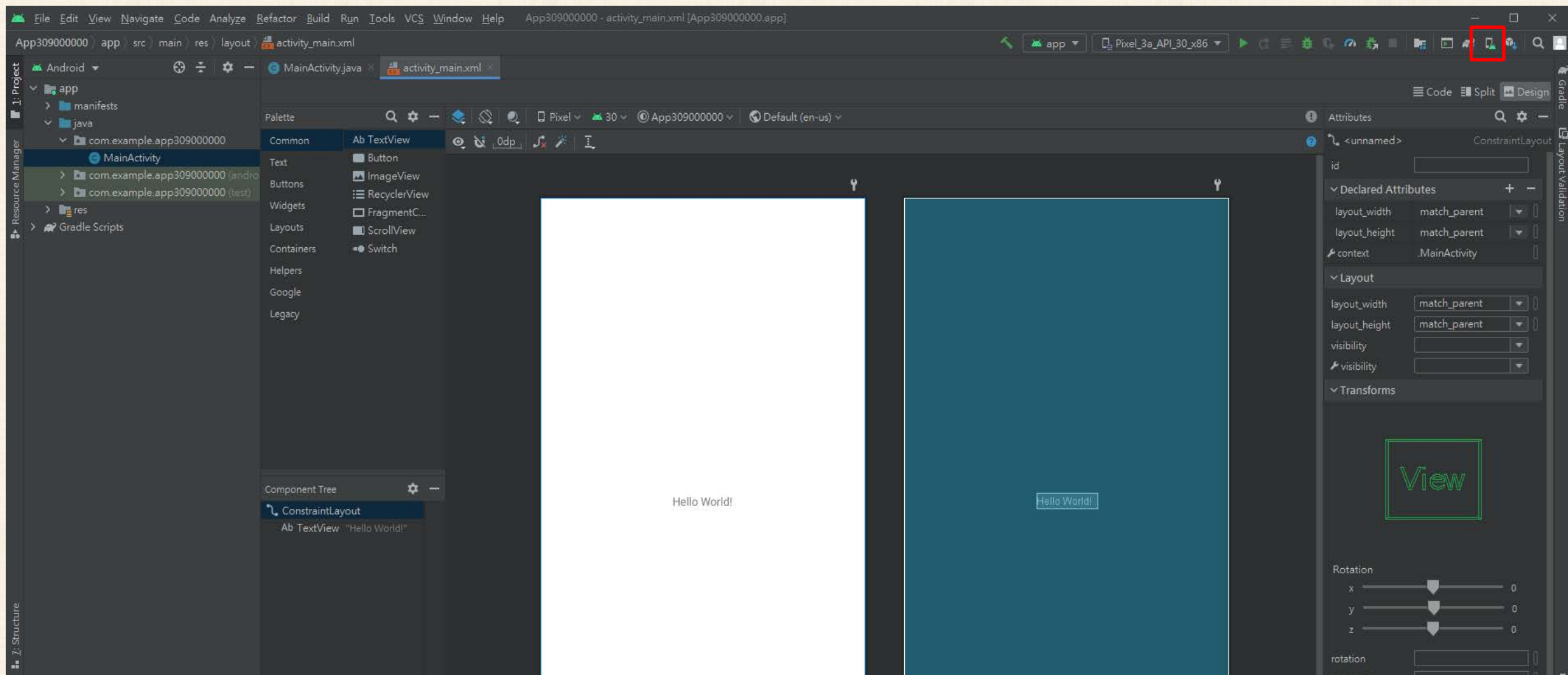
i Your app will run on approximately 99.8% of devices.
[Help me choose](#)

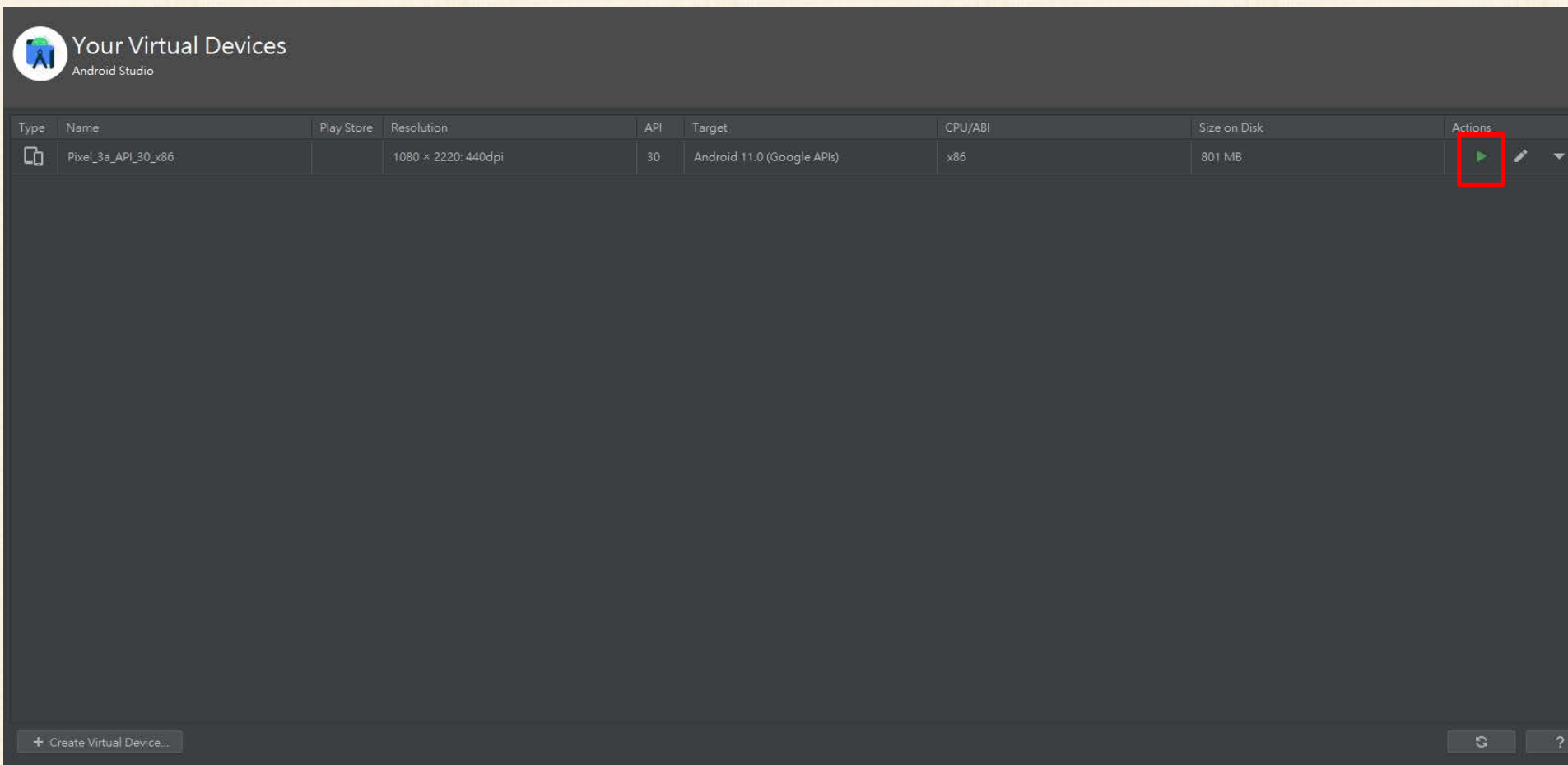
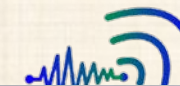
☐ Use legacy android.support libraries **?**
Using legacy android.support libraries will prevent you from using the latest Play Services and Jetpack libraries

Previous Next Cancel **Finish**

Step 3

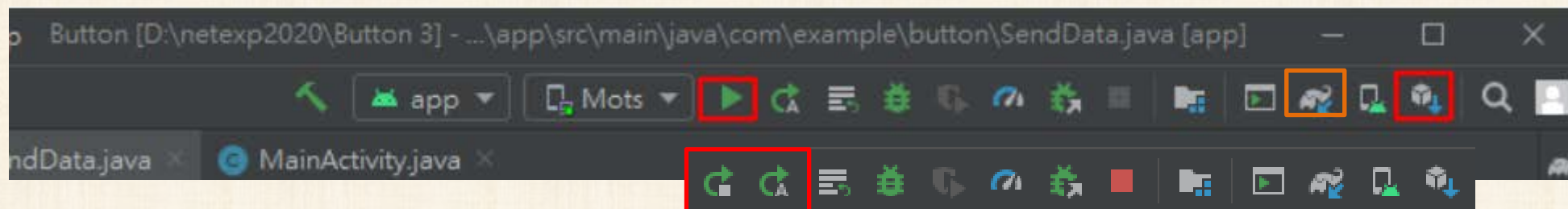
AVD Manager



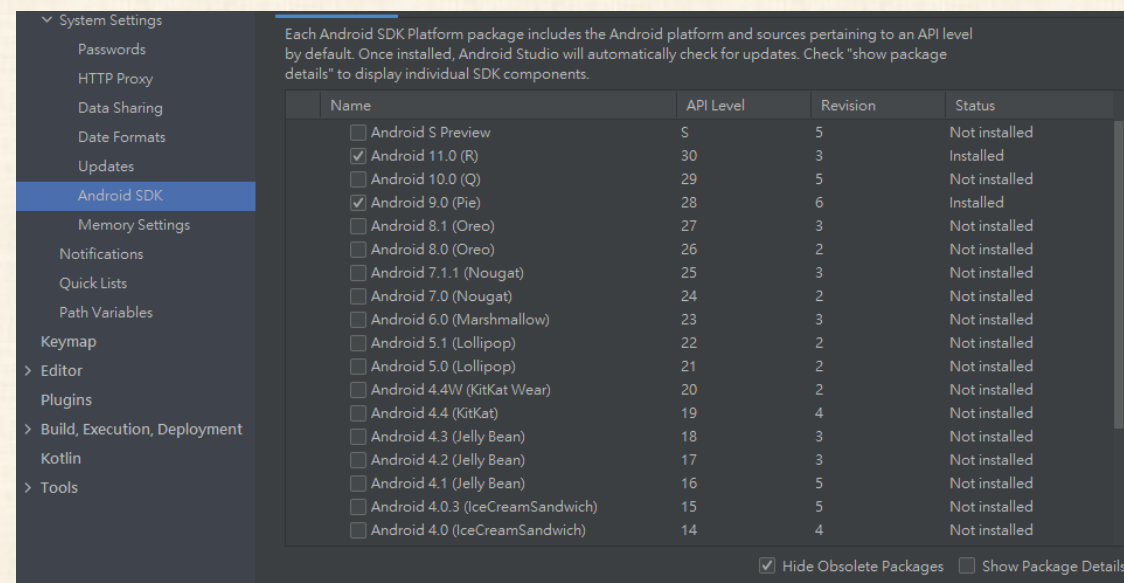


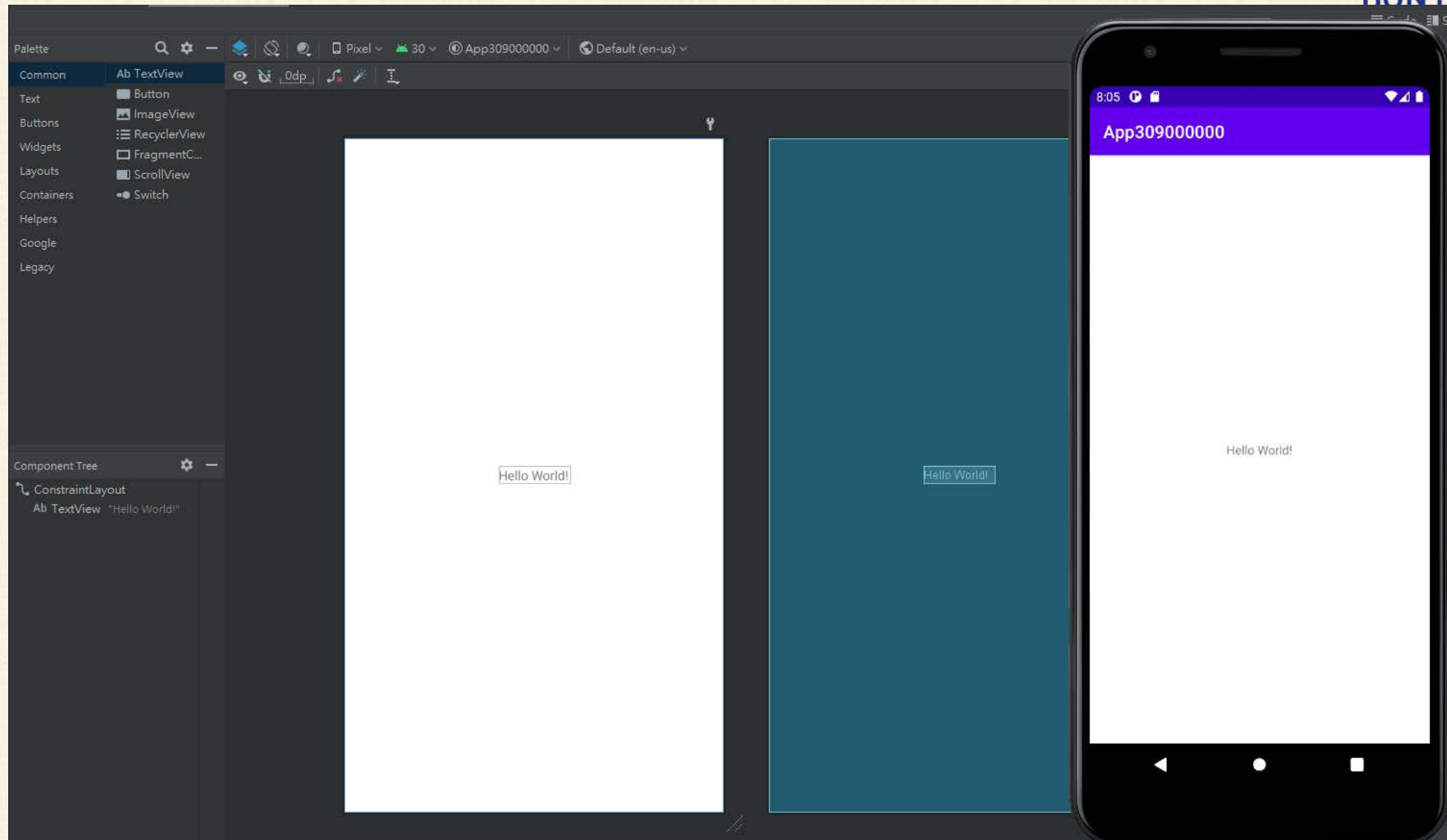
□ 如果遇到需要權限、密碼，打x或略過

Step 4



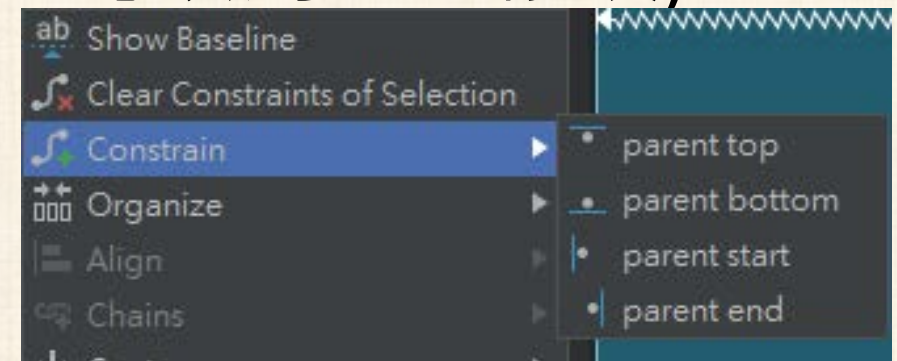
- 點箭頭把App燒入模擬器
- 如果箭頭不能點，點立方體或橘色框框那個





製作元件

- 選取.xml檔的Design介面
- 在左邊Palette選取你要的元件，拖曳到中間的畫面
- 在右邊Attributes可以選取or更改該元件的各類屬性
e.g. id、text、顏色等
- 在畫面中的元件上點選右鍵可以藉由center對齊畫面，
constrain則用來固定該元件的位置 (也可以用code修改)



activity_main.xml

MainActivity.java activity_main.xml

Palette

Common Ab TextView

Text Button

Buttons ImageView

Widgets RecyclerView

Layouts FragmentC...

Containers ScrollView

Helpers Switch

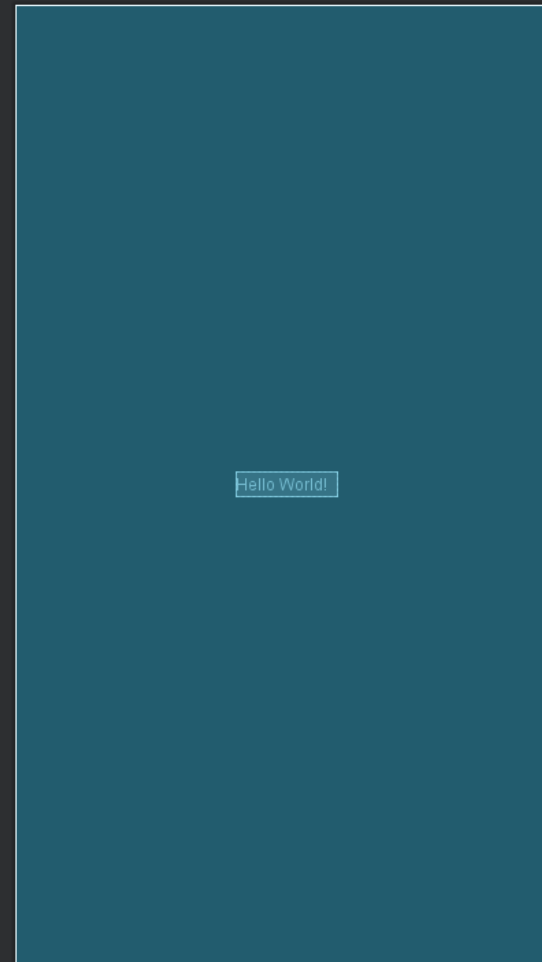
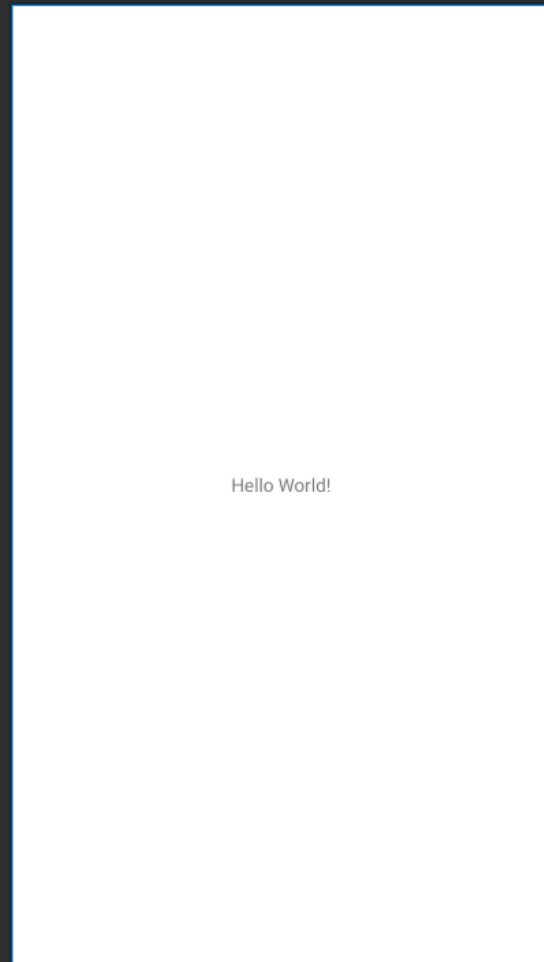
Google

Legacy

Component Tree

ConstraintLayout

Ab TextView "Hello World!"



Attributes

<unnamed> ConstraintLayout

id

Declared Attributes

layout_width match_parent

layout_height match_parent

context .MainActivity

Layout

layout_width match_parent

layout_height match_parent

visibility

visibility

Transforms

View

Rotation

x 0

y 0

z 0

rotation

rotationX

rotationY

scaleX

scaleY

translationX

translationY

alpha

Common Attributes

Q1

- Button 小程式
- 新增一個TextView，Text改為Android Lab1 Demo，id=showtext
- 新增一個Button，Text改為Demo，id=demobutton
- 點擊Button後會顯示"Pass!"
- Hint: 下一頁的code要打在MainActivity.java

MainActivity.java

```
import android.os.Bundle;
import android.view.View;
import android.widget.Button;
import android.widget.TextView;

public class MainActivity extends AppCompatActivity {
    TextView showtext;
    Button demobutton;

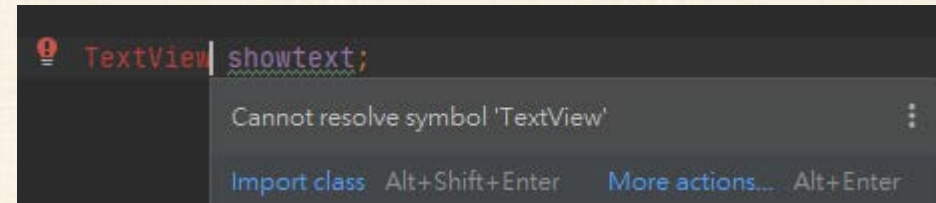
    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_main);
        showtext = (TextView)findViewById(R.id.showtext);
        demobutton = (Button)findViewById(R.id.demobutton);
        demobutton.setOnClickListener(new View.OnClickListener() {
            @Override
            public void onClick(View v) {
                showtext.setText("Pass!");
            }
        });
    }
}
```

宣告變數型別、名稱

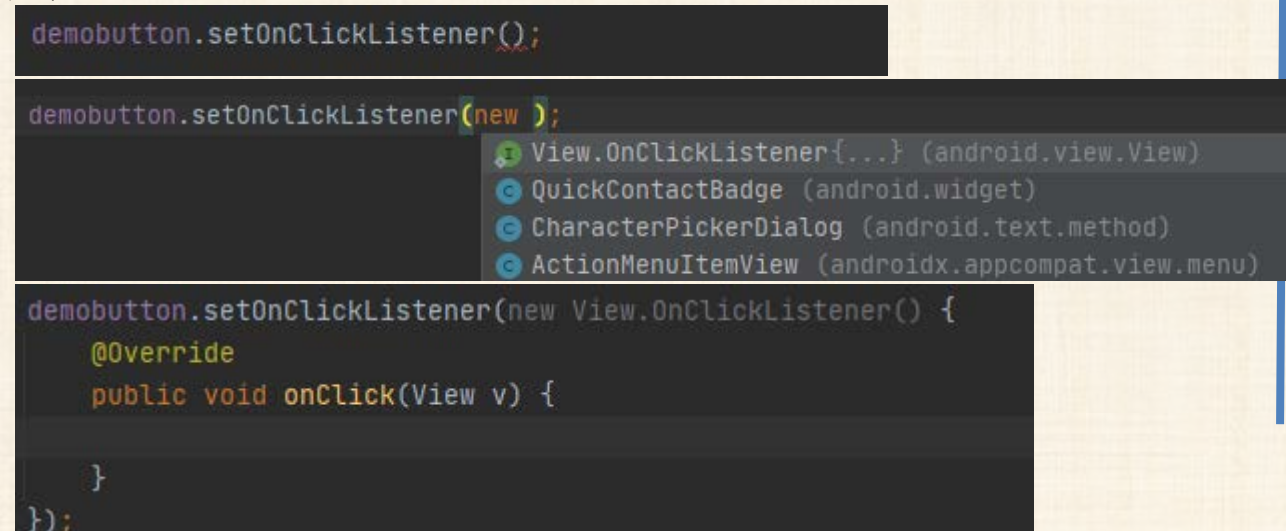
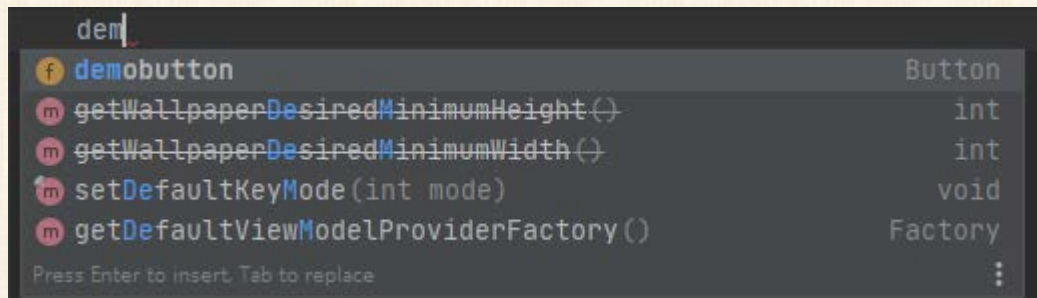


輸入程式碼的時候

- 看到紅線或紅字
 - 在有錯的地方按 Alt + Enter 或是點紅色燈泡可以使用系統自動修正
 - 可以自動import class等



- 出現提示按tab可以快速修改

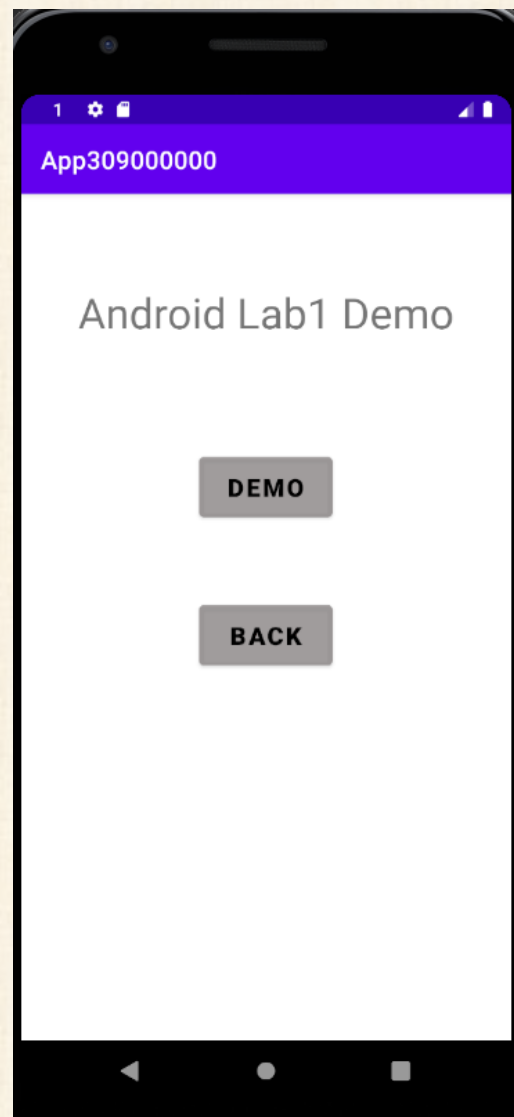


Q2

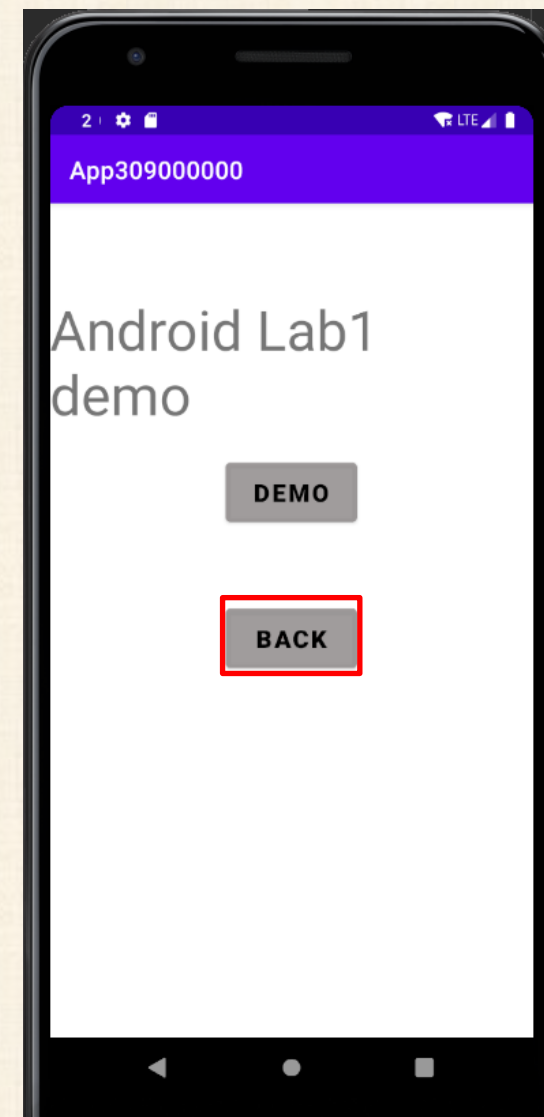
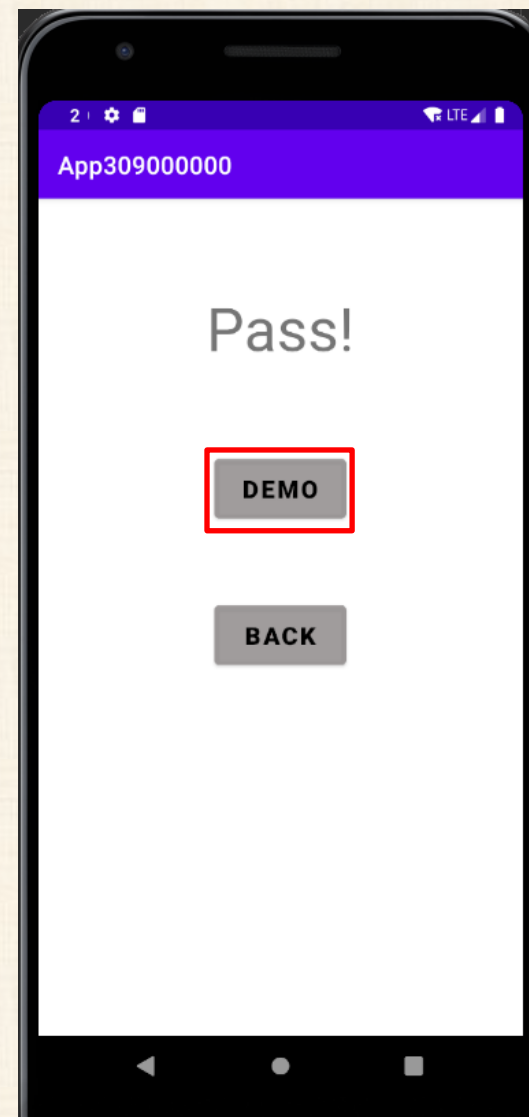
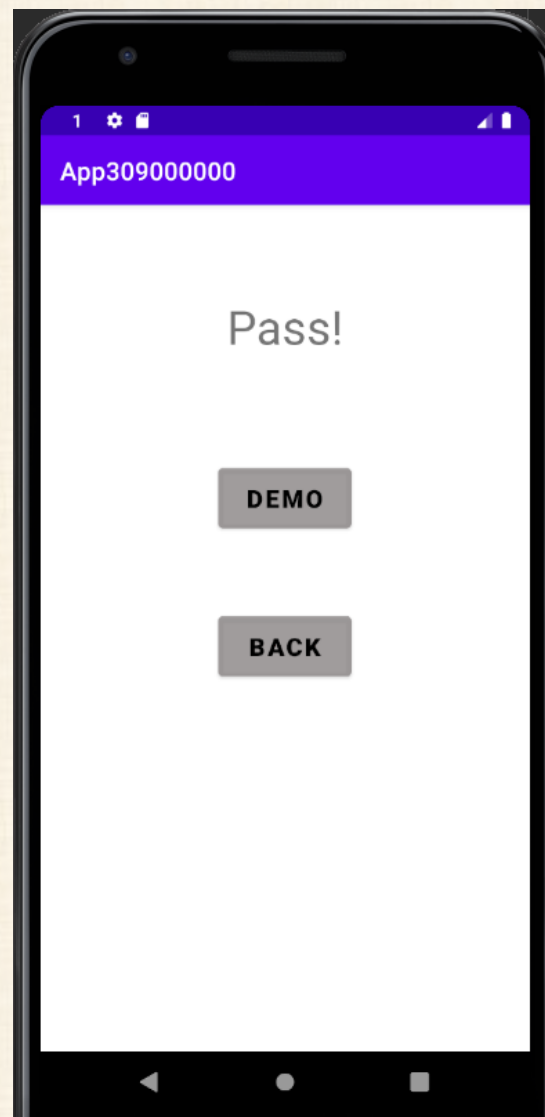
- 在Q1的基礎上，每按一次Demo後”Pass!”字體會變大一點
- 新增一個Back Button，按下後會顯示”Android Lab1 demo”，且每按一次Back Button顯示的字體會變小一點
- Hint: 1. `size = (int)showtext.getTextSize()`
2. `showtext.setTextSize(TypedValue.COMPLEX_UNIT_PX, size)`

不要忘記在xml的元件要設id、給定位，在java上要用findViewById連結元件

打開app時



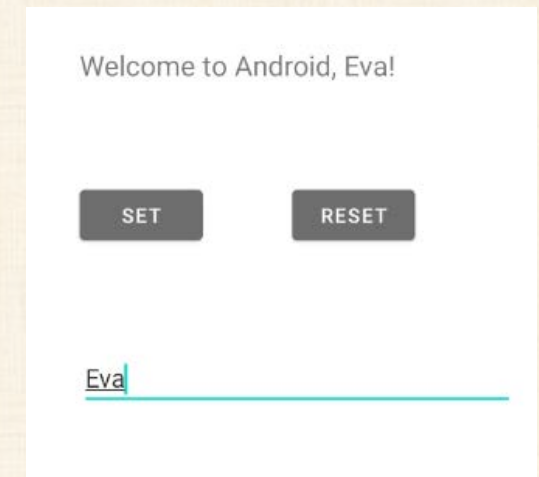
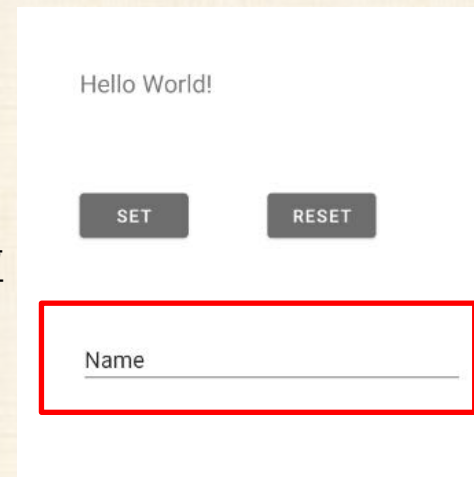
Q1結果



Q3

- 在EditText上輸入名字，按下SET顯示Welcome to Android, “輸入的名字”!
- 按下RESET，顯示Hello World!

這個元件可以用Plain Text拉出來，類別是EditText



- Hint:

EditText Name; //記得宣告元件

.setText(“自己打”+ Name.getText().toString())

只接受文字的形式

不要忘記在xml的元件要設id、給定位，在java上要用findViewById連結元件