

**Embedded System Project Laboratory 嵌入式
系統專題實驗
實驗（一）**

Android 之 Activity 切換

授課教師：曾嘉影 老師

班 級：I3B

座 號：54

姓 名：黃子睿

題目 1：

Android 之Activity 切換

實驗內容及問題說明

試以範例步驟練習完成可以在 Activity 間傳遞參數的程式。

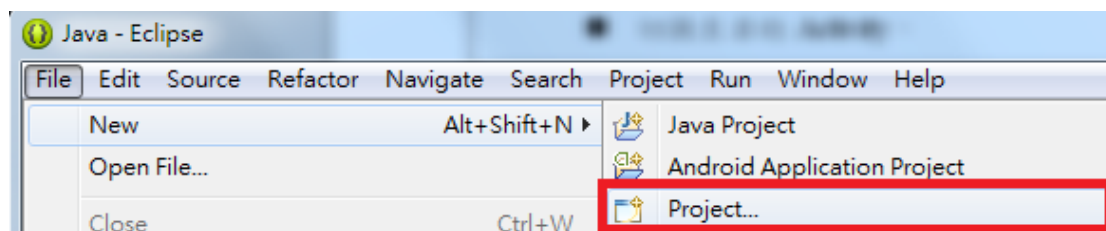
(完成與範例相同功能程式給基本 60 分)

試著練習其他 Android 操作，修改、新增功能則再加分

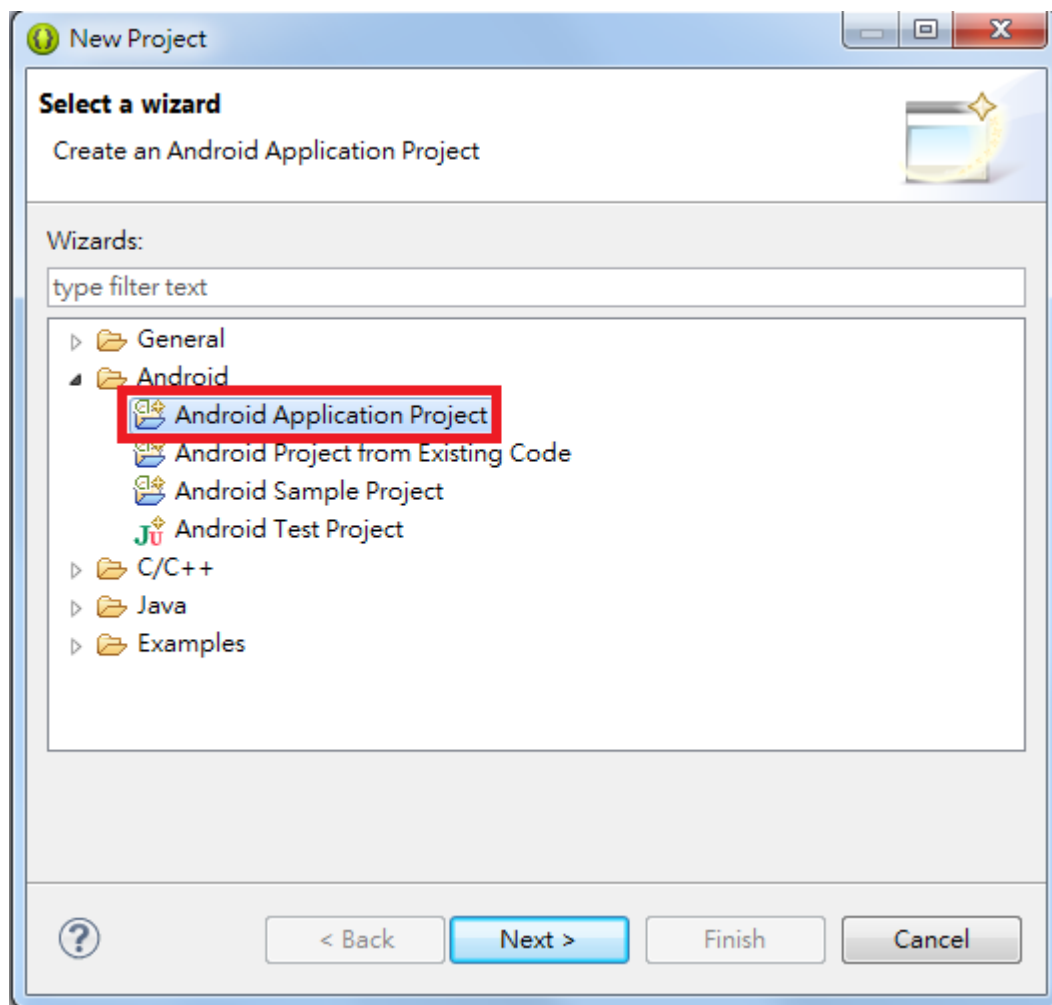
(如更改背景、修改字體大小、顏色、多個分頁…等)。

實驗程式及輸出畫面

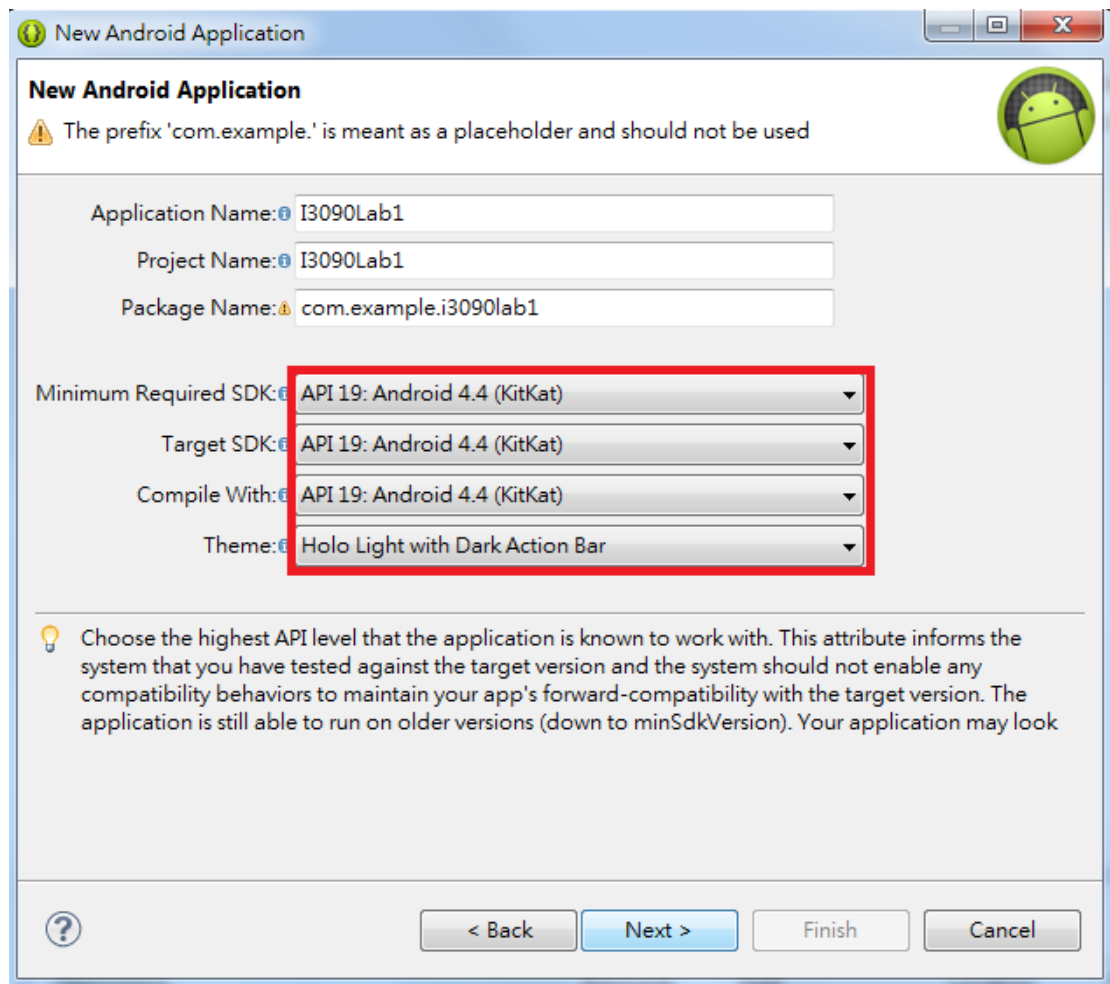
1. 開啟一個新的專案，File->Project



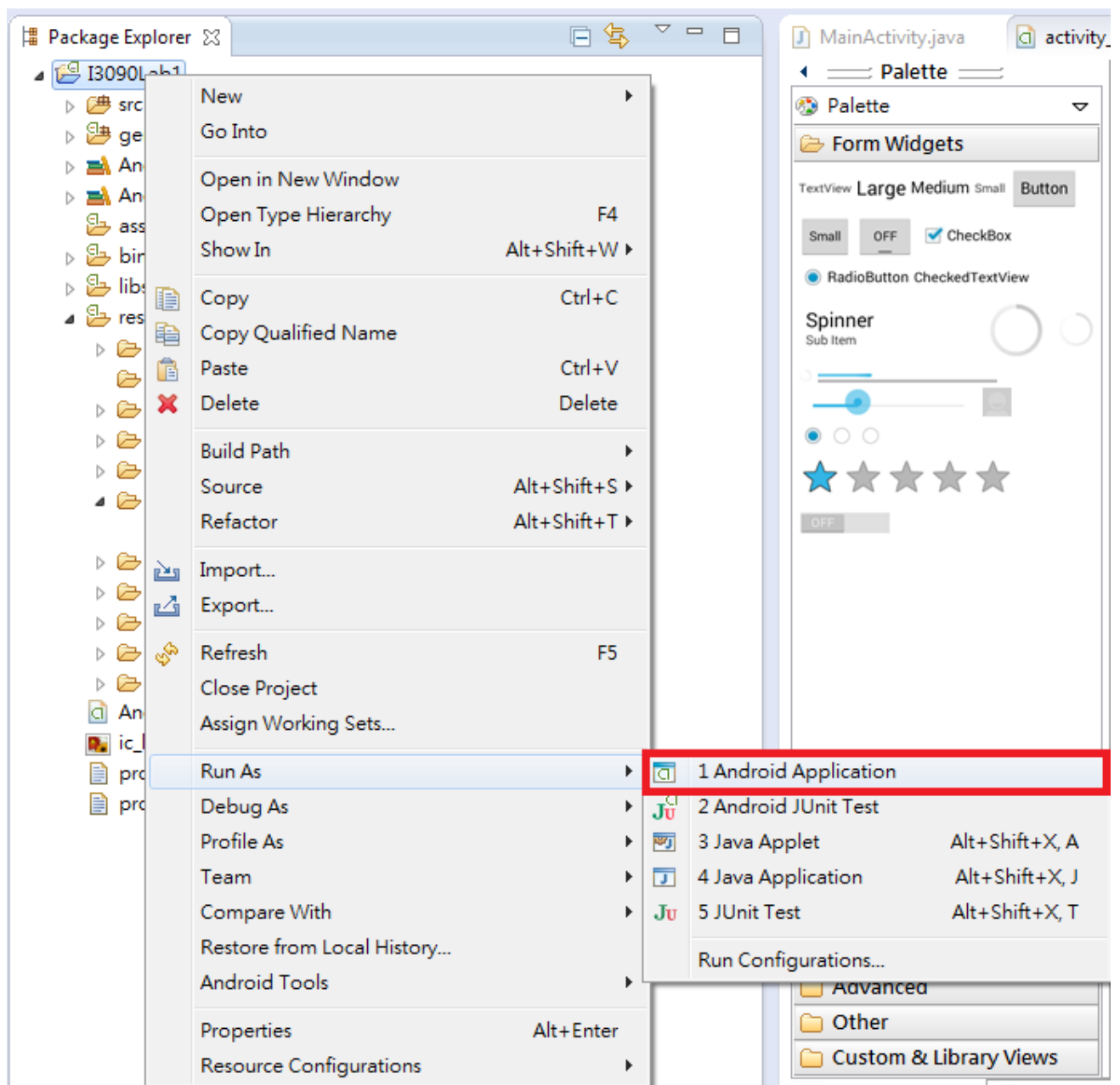
2. 選擇Android Application Project

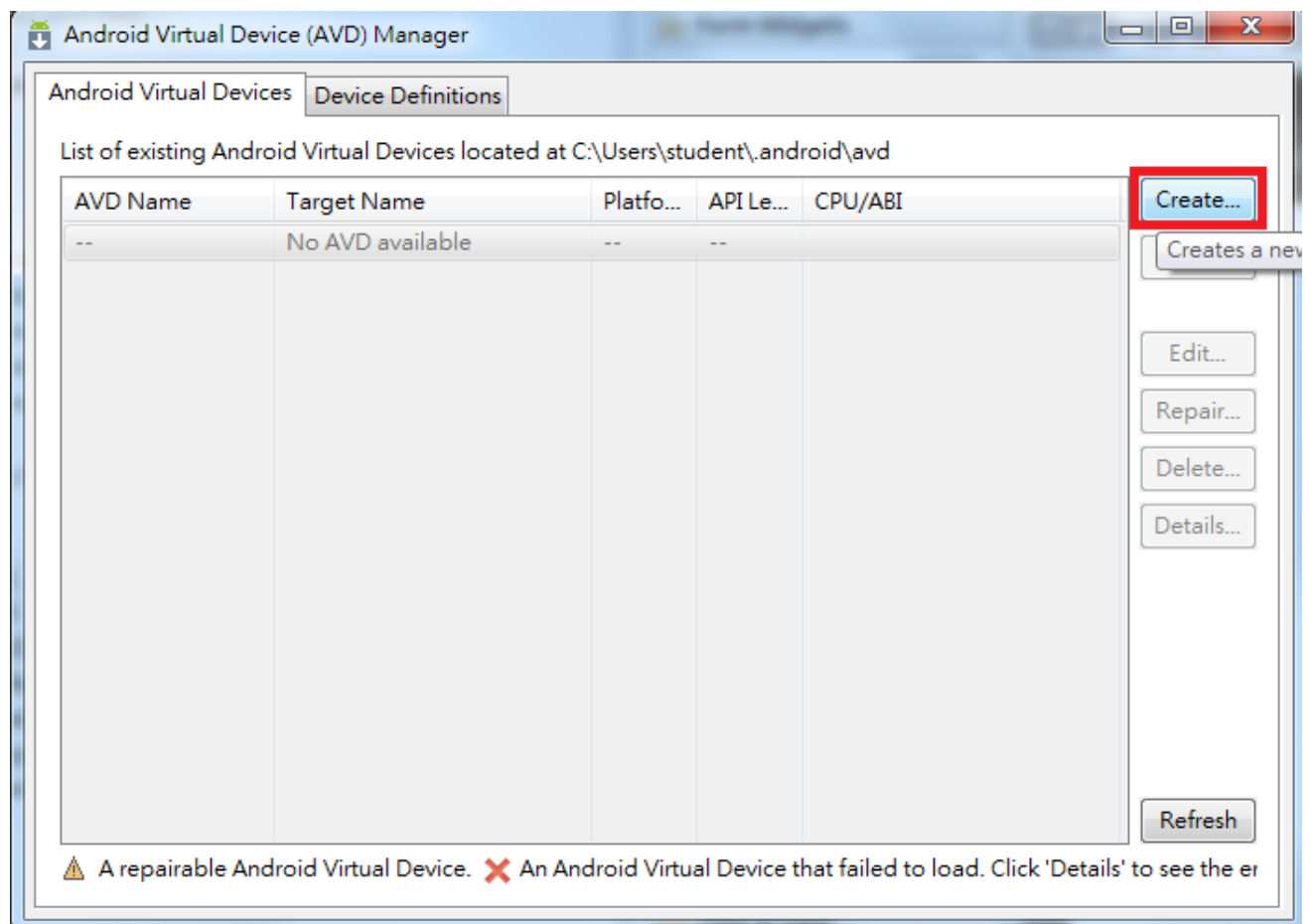


3. 如圖設定



4. 接著執行看看，在專案上按滑鼠右鍵，Run As->Android Application





Create new Android Virtual Device (AVD) ✕

AVD Name:

Device: 5.1" WVGA (480 × 800: mdpi) ▾

Target: Android 4.4.2 - API Level 19 ▾

CPU/ABI: ARM (armeabi-v7a) ▾

Keyboard: ☒ Hardware keyboard present

Skin: Skin with dynamic hardware controls ▾

Front Camera: None ▾

Back Camera: None ▾

Memory Options: RAM: VM Heap:

Internal Storage: MiB ▾

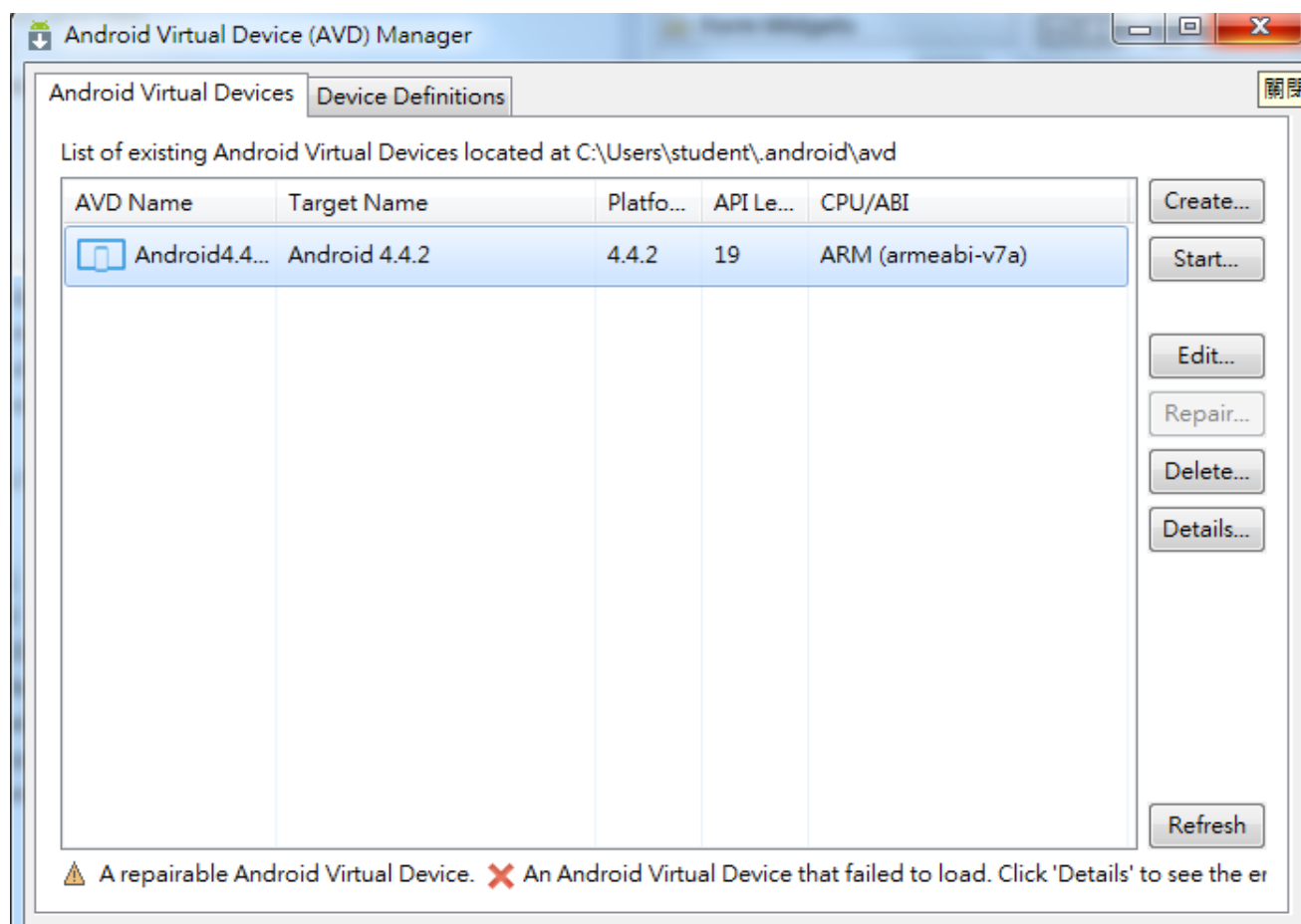
SD Card:

☒ Size: MiB ▾

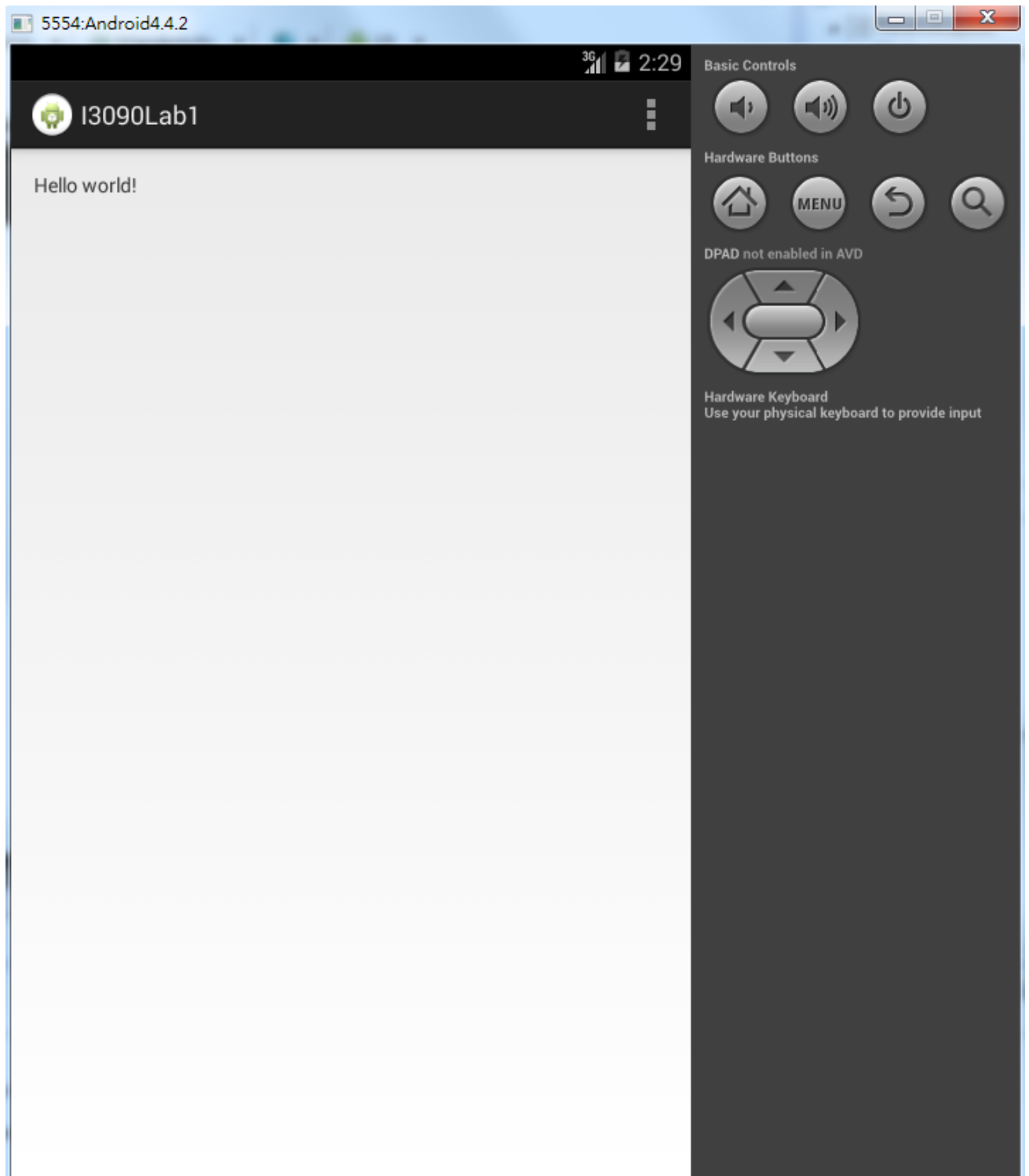
☐ File: Browse...

Emulation Options: ☐ Snapshot ☒ Use Host GPU

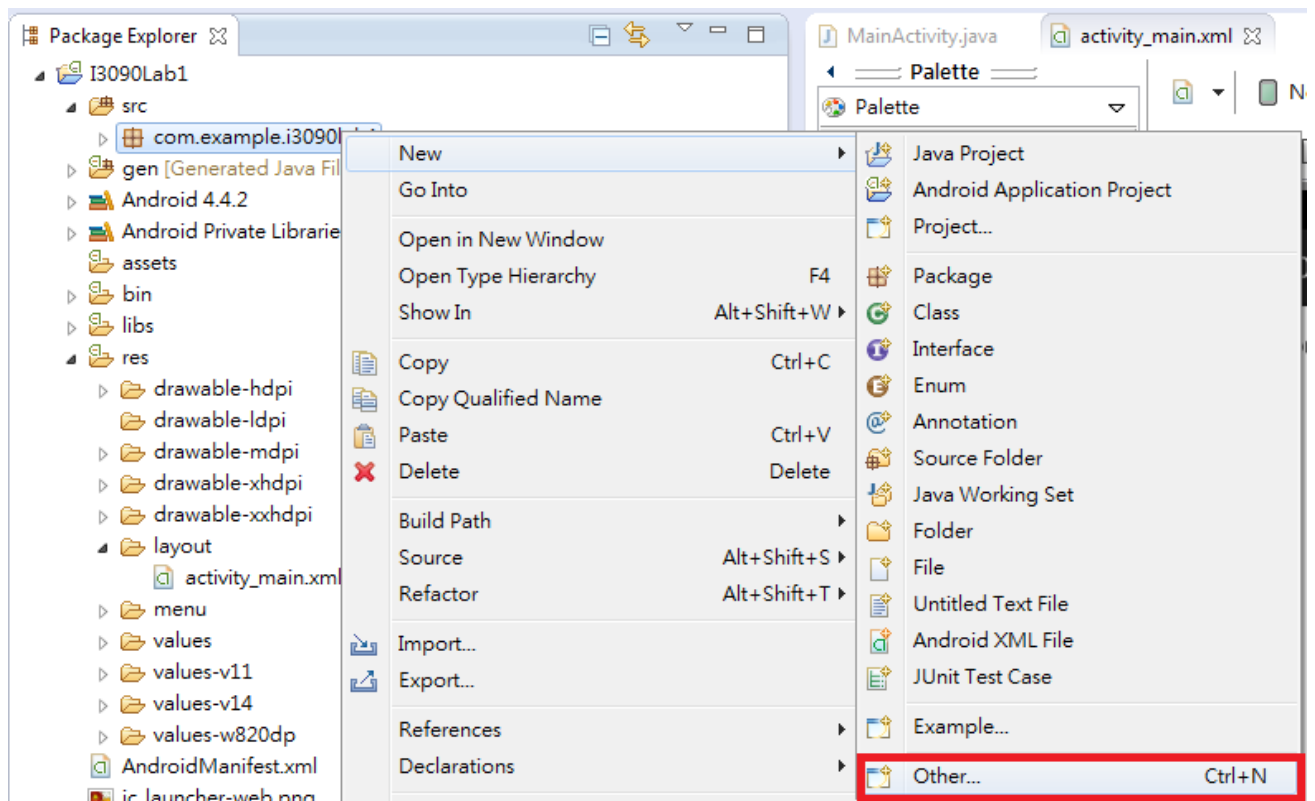
☐ Override the existing AVD with the same name



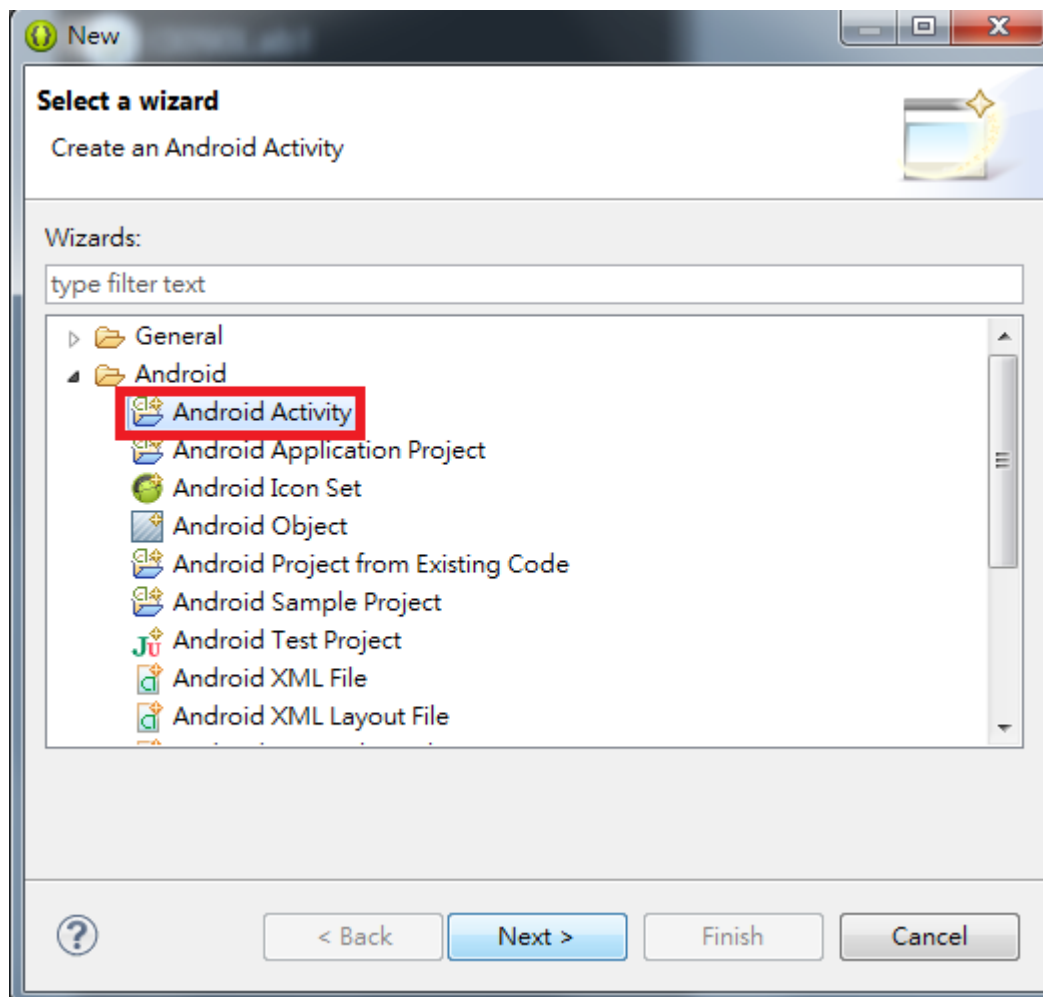
5. 正常會是如圖所示



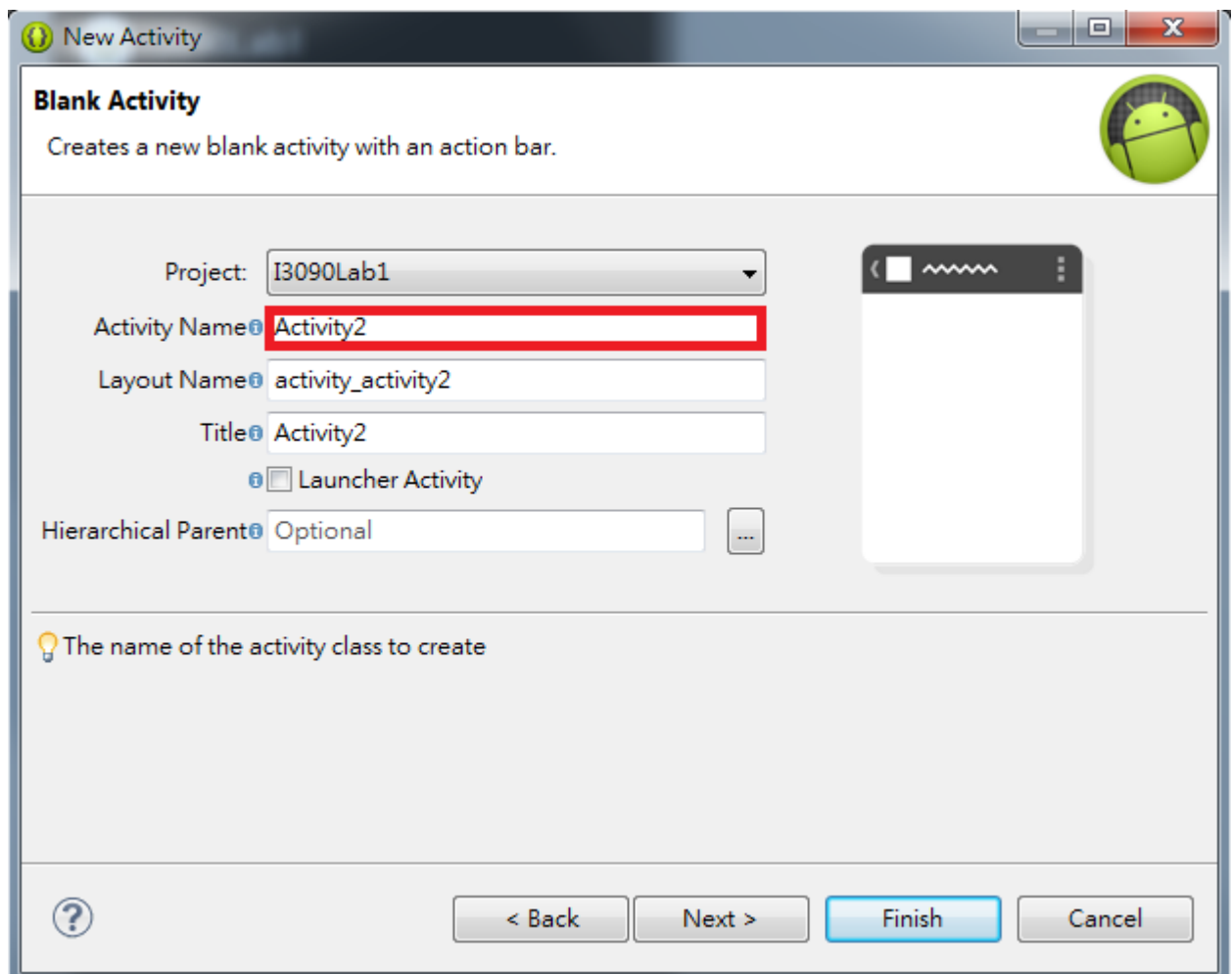
6. Package 上滑鼠右鍵，New->Other



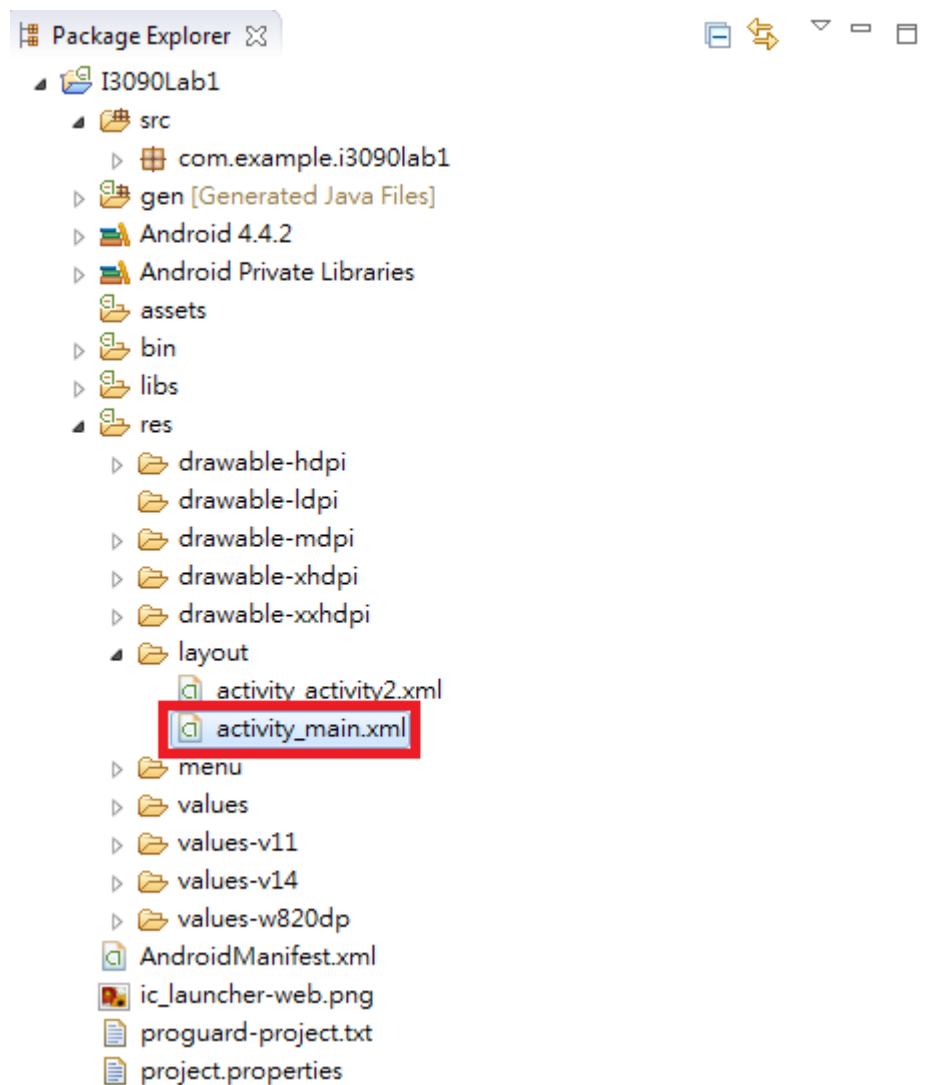
7. 選擇Android Activity



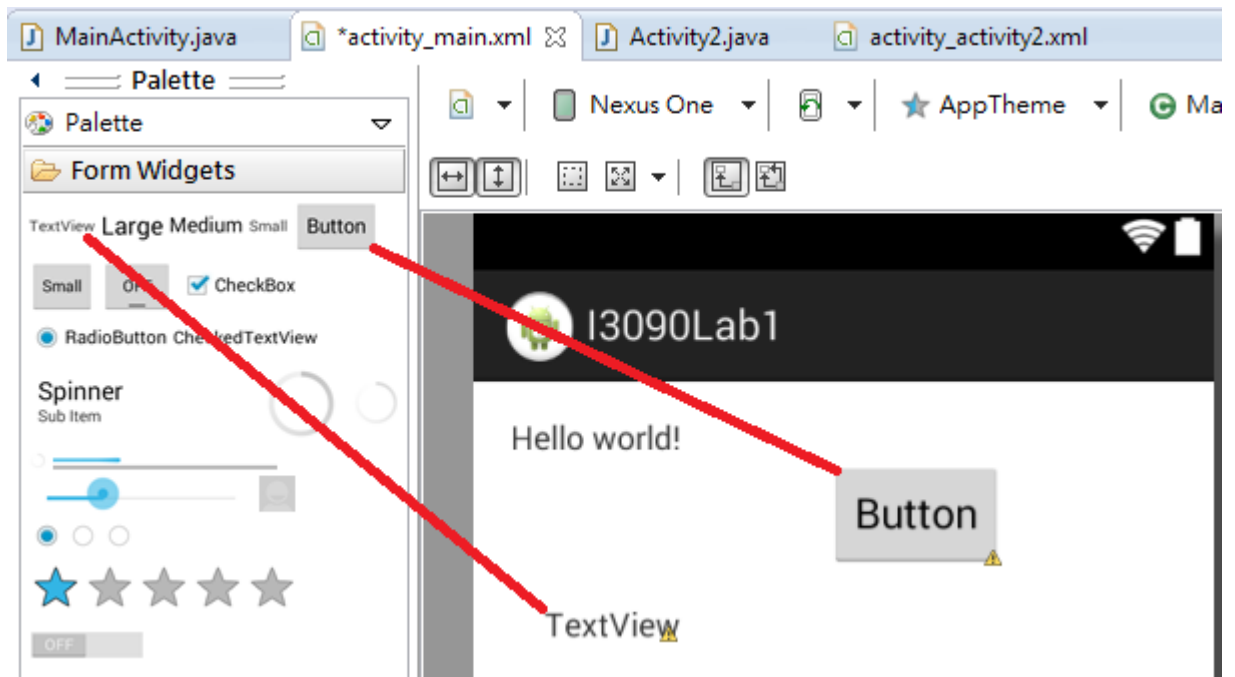
8. 打上 Activity 名字後 Finish



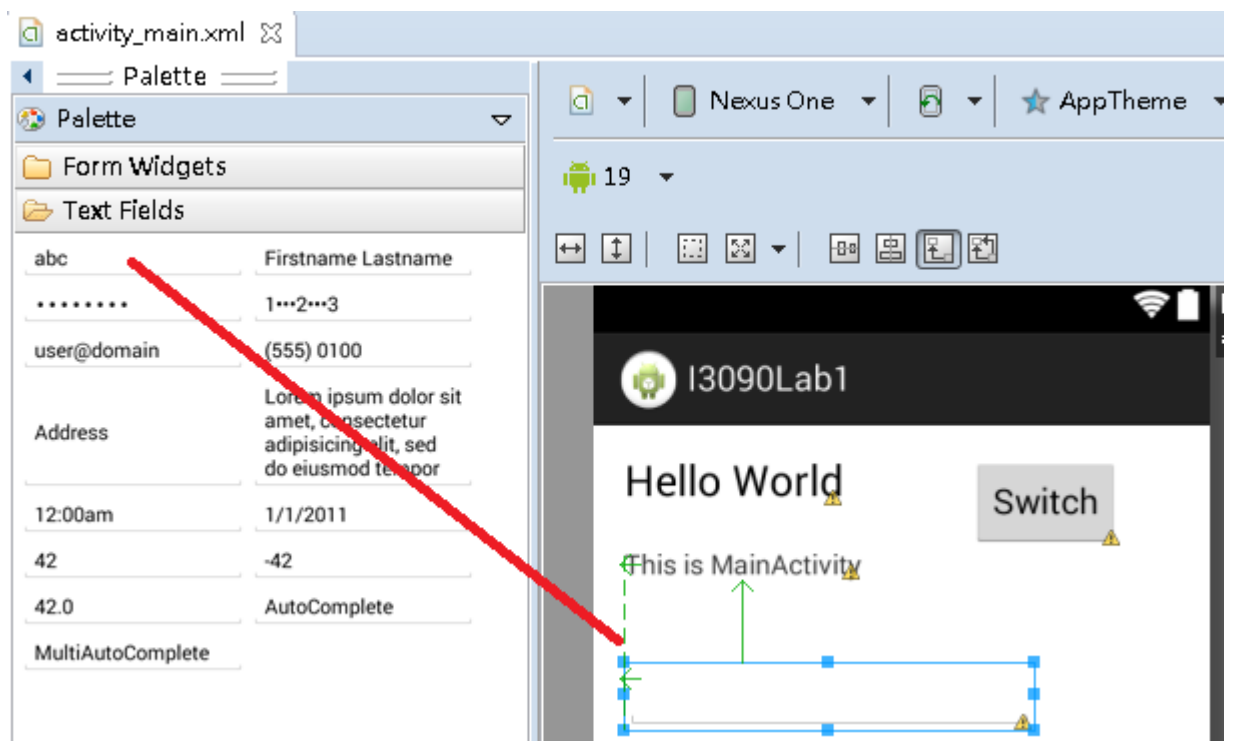
9. 開啟 layout 底下的 activity_main.xml



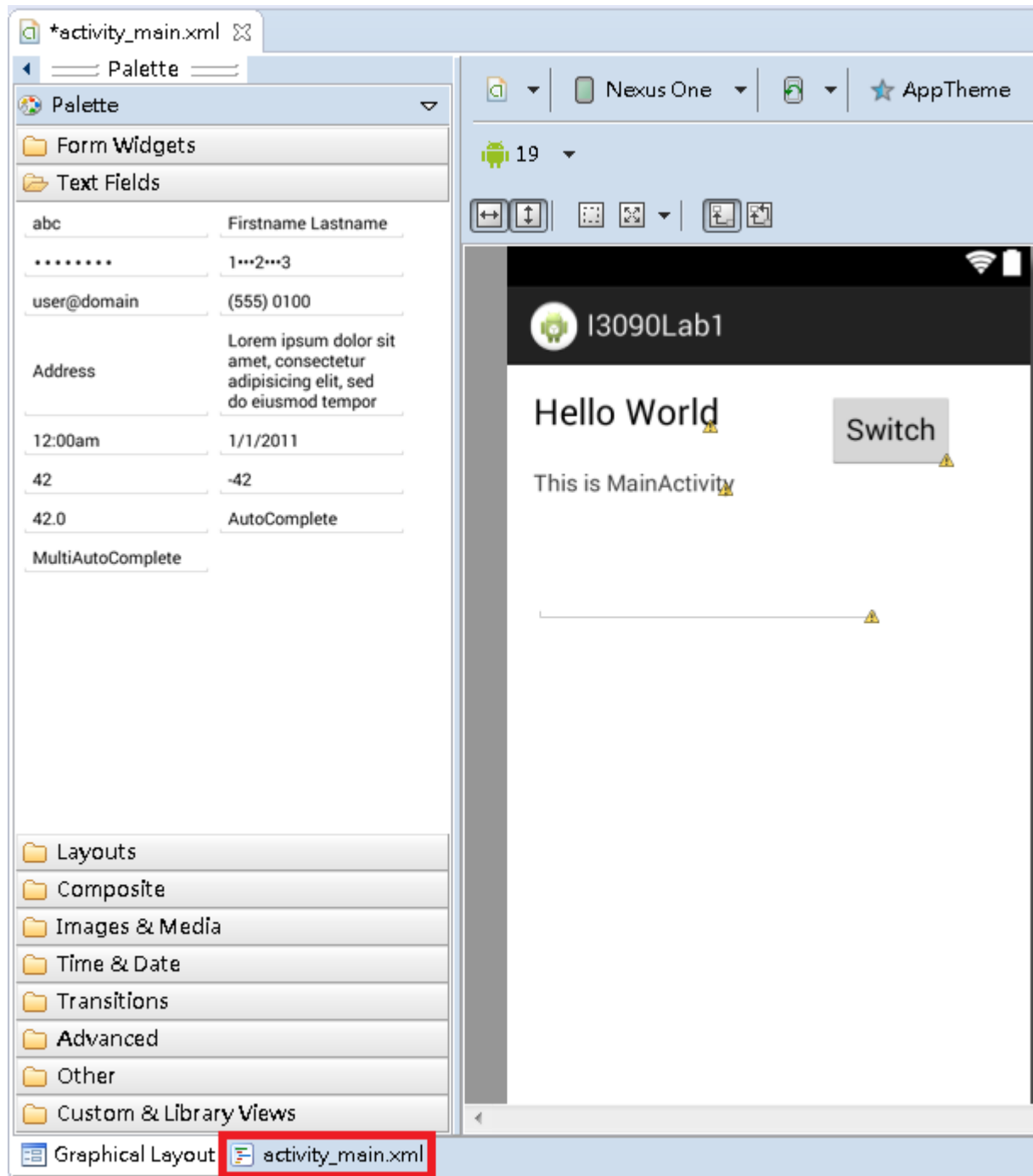
10. 從左邊直接把元件拉到預覽畫面，就完成了元件的新增



11. 切換到Text Fields，然後如圖拉出一個輸入框



12. 視窗底下可以切換分頁，如圖切換到程式碼的部分



13. 綠色的是這個元件的名字，紅色的是需要修改的地方

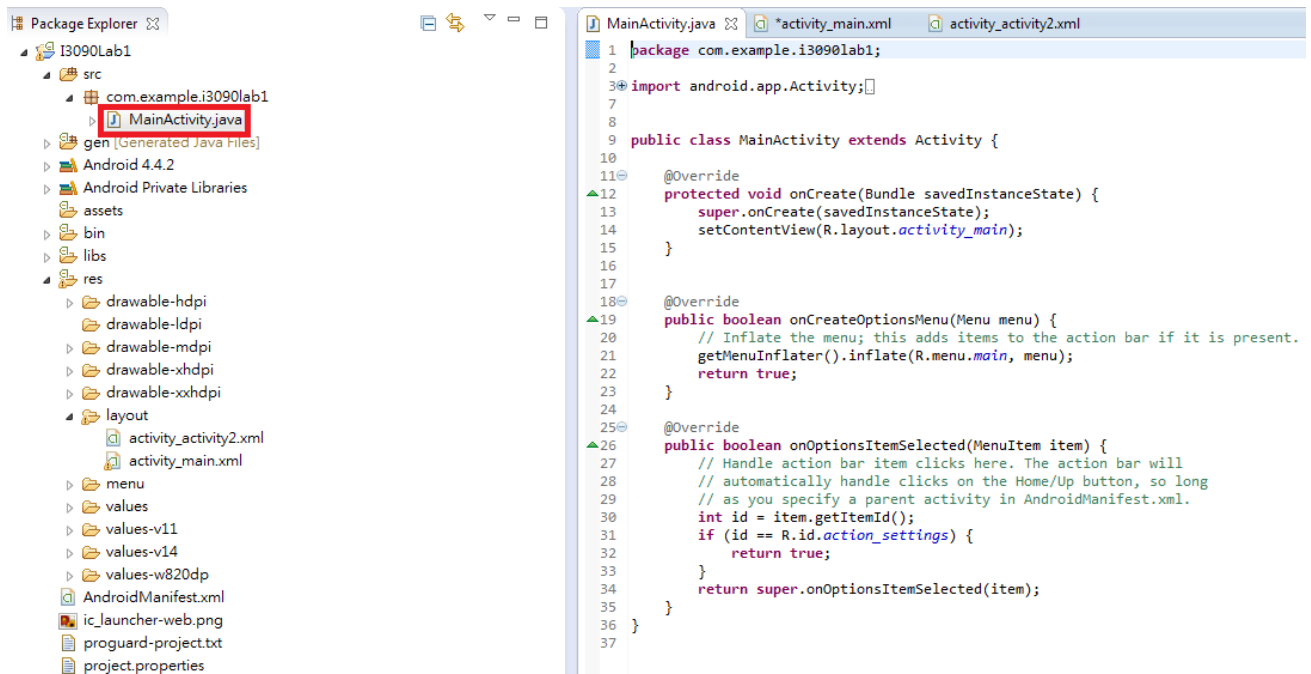
上面修改成"Switch"，下面修改成"This is MainActivity"

```

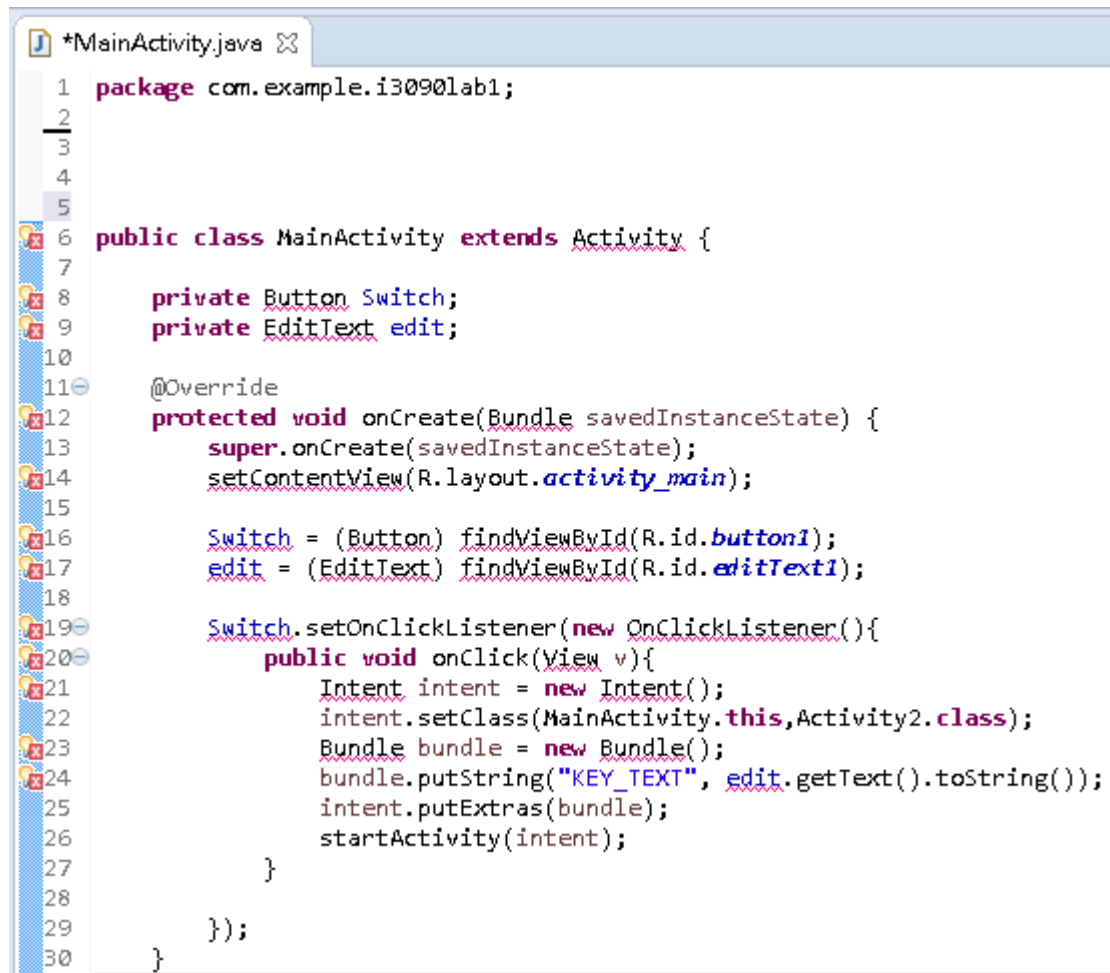
11 <EditText
12     android:id="@+id/editText1"
13     android:layout_width="wrap_content"
14     android:layout_height="wrap_content"
15     android:layout_alignLeft="@+id/textView2"
16     android:layout_below="@+id/textView2"
17     android:layout_marginTop="41dp"
18     android:ems="10" />
19
20 <Button
21     android:id="@+id/button1"
22     android:layout_width="wrap_content"
23     android:layout_height="wrap_content"
24     android:layout_alignParentRight="true"
25     android:layout_marginRight="30dp"
26     android:text="Switch" />
27
28 <TextView
29     android:id="@+id/textView2"
30     android:layout_width="wrap_content"
31     android:layout_height="wrap_content"
32     android:layout_below="@+id/button1"
33     android:text="This is MainActivity" />

```

14. 開啟MainActivity.java

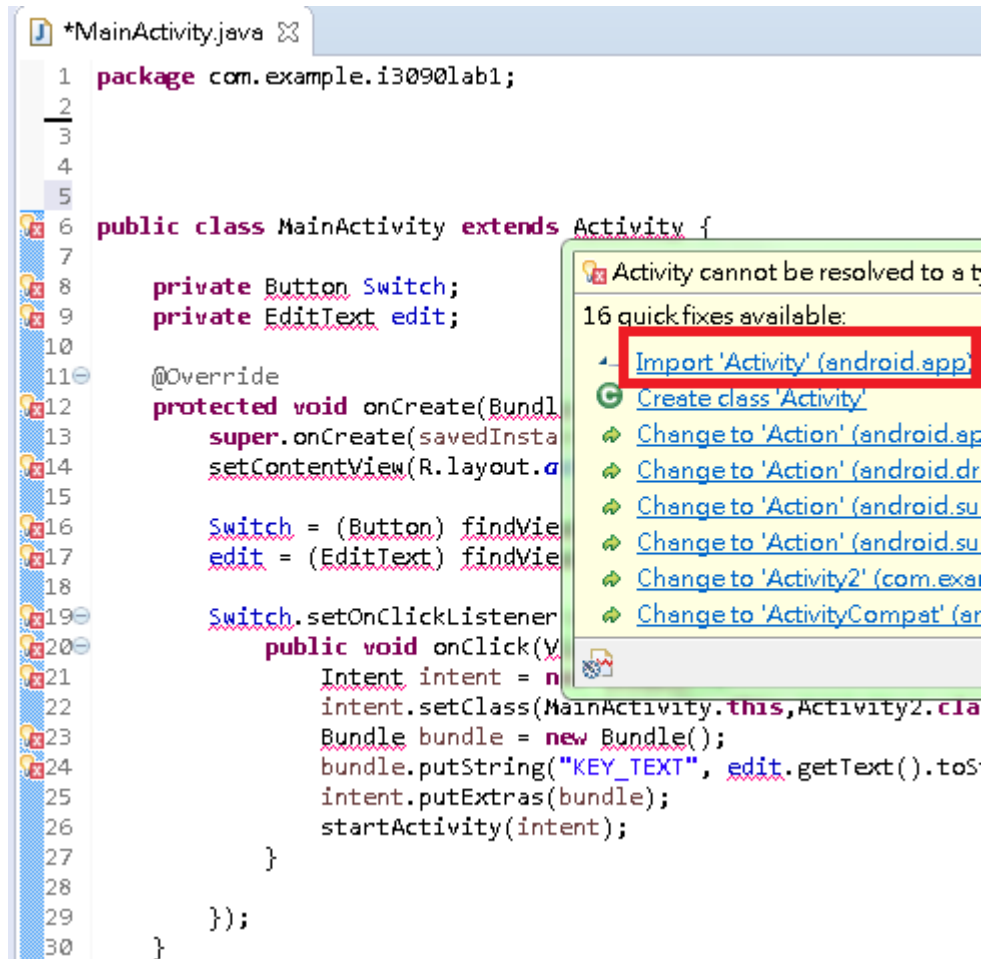


15. 如圖輸入程式碼



```
1 package com.example.i3090lab1;
2
3
4
5
6 public class MainActivity extends Activity {
7
8     private Button Switch;
9     private EditText edit;
10
11     @Override
12     protected void onCreate(Bundle savedInstanceState) {
13         super.onCreate(savedInstanceState);
14         setContentView(R.layout.activity_main);
15
16         Switch = (Button) findViewById(R.id.button1);
17         edit = (EditText) findViewById(R.id.editText1);
18
19         Switch.setOnClickListener(new OnClickListener(){
20             public void onClick(View v){
21                 Intent intent = new Intent();
22                 intent.setClass(MainActivity.this, Activity2.class);
23                 Bundle bundle = new Bundle();
24                 bundle.putString("KEY_TEXT", edit.getText().toString());
25                 intent.putExtras(bundle);
26                 startActivity(intent);
27             }
28         });
29     }
30 }
```

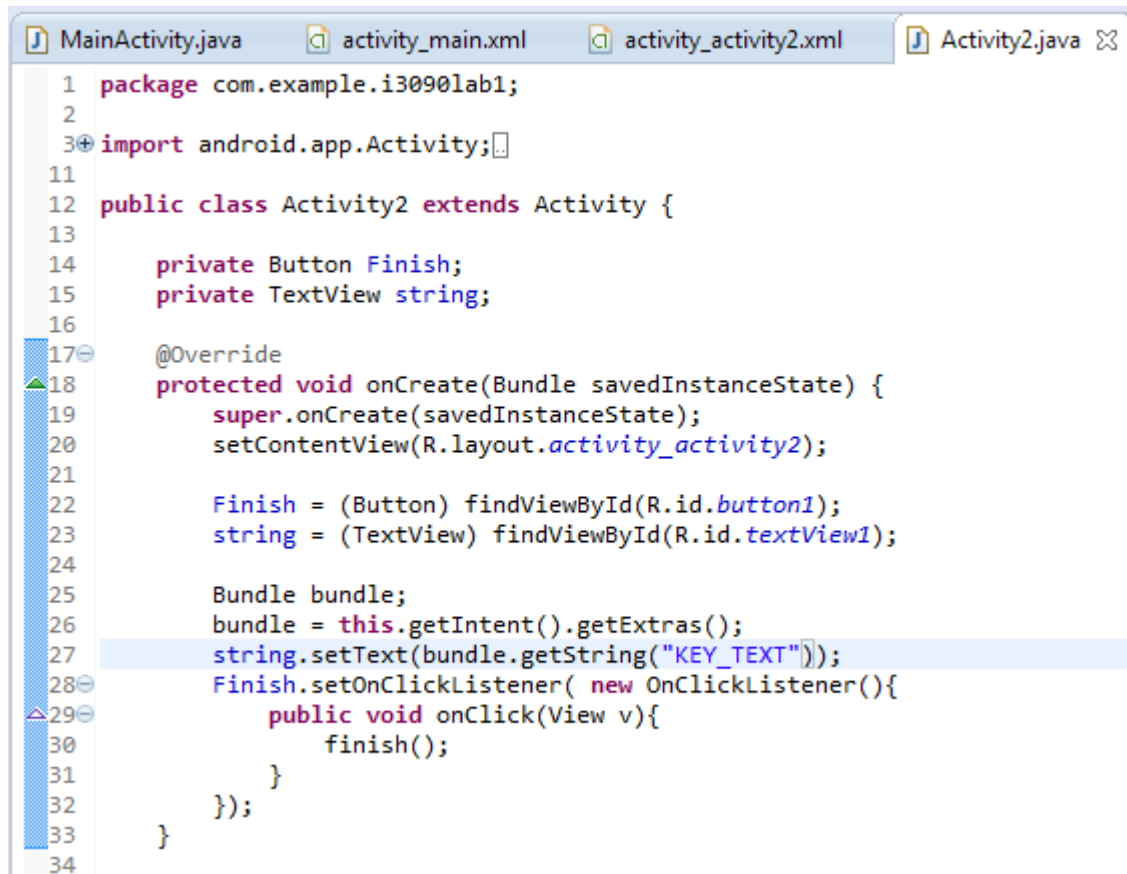
16. 滑鼠移動到紅線上，出現 quick fixes 視窗時，如圖選擇 Import ，
沒有 Import 的選項就跳過不要選



17. 接著開啟 `activity_activity2.xml`，依照剛才步驟拉完元件再修改程式碼

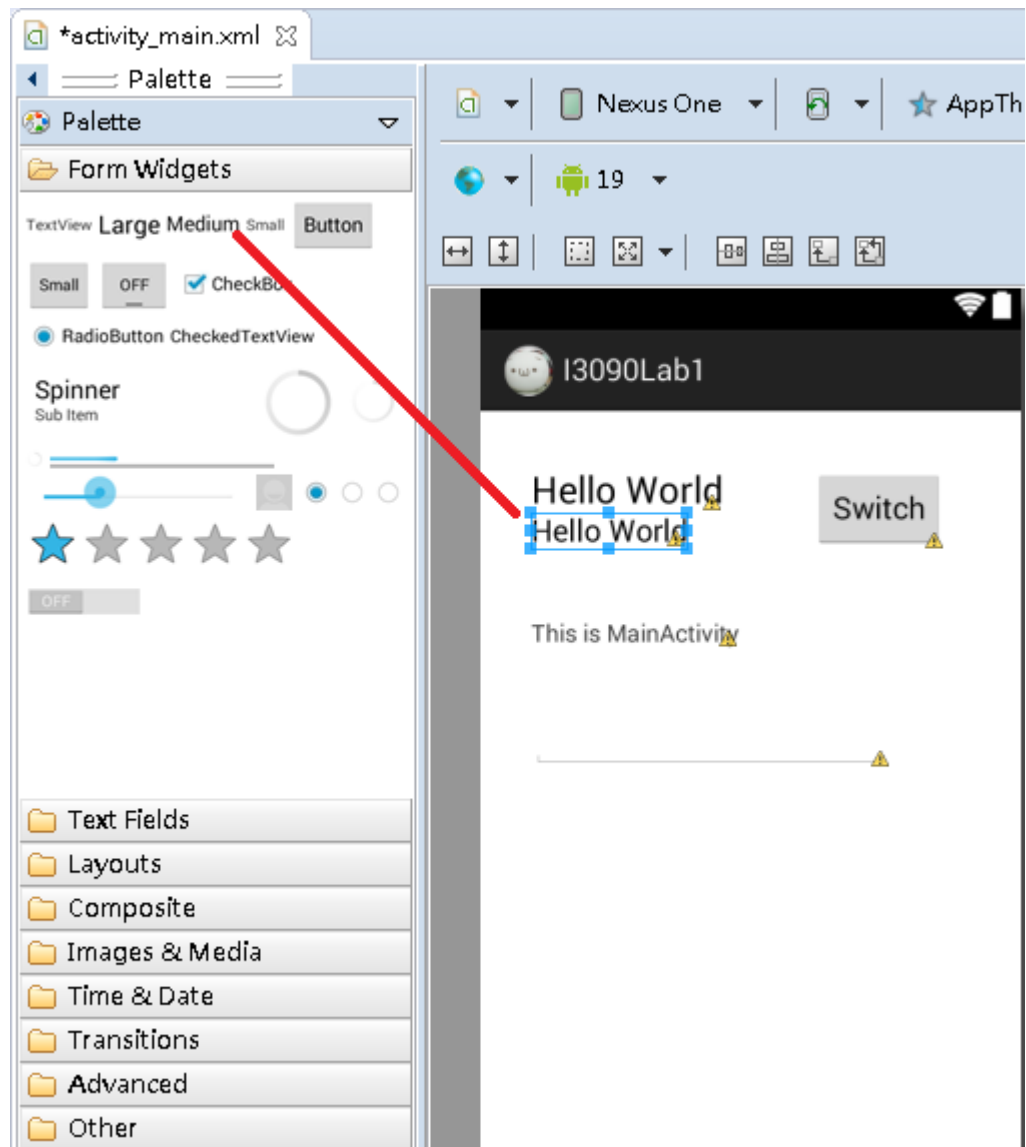
```
*MainActivity.java  *activity_main.xml  *activity_activity2.xml  ⌵
1  <RelativeLayout xmlns:android="http://schemas.android.com/apk/res/android"
2      xmlns:tools="http://schemas.android.com/tools"
3      android:layout_width="match_parent"
4      android:layout_height="match_parent"
5      android:paddingBottom="@dimen/activity_vertical_margin"
6      android:paddingLeft="@dimen/activity_horizontal_margin"
7      android:paddingRight="@dimen/activity_horizontal_margin"
8      android:paddingTop="@dimen/activity_vertical_margin"
9      tools:context="com.example.i3090lab1.Activity2" >
10
11      <TextView
12          android:id="@+id/textView1"
13          android:layout_width="wrap_content"
14          android:layout_height="wrap_content"
15          android:text="NULL" />
16
17      <TextView
18          android:id="@+id/textView2"
19          android:layout_width="wrap_content"
20          android:layout_height="wrap_content"
21          android:layout_alignLeft="@+id/textView1"
22          android:layout_below="@+id/button1"
23          android:layout_marginTop="72dp"
24          android:text="This is Activity2" />
25
26      <Button
27          android:id="@+id/button1"
28          android:layout_width="wrap_content"
29          android:layout_height="wrap_content"
30          android:layout_alignParentRight="true"
31          android:layout_alignTop="@+id/textView1"
32          android:layout_marginRight="28dp"
33          android:text="Return" />
```

18. 如圖修改 Activity2.java 的程式碼

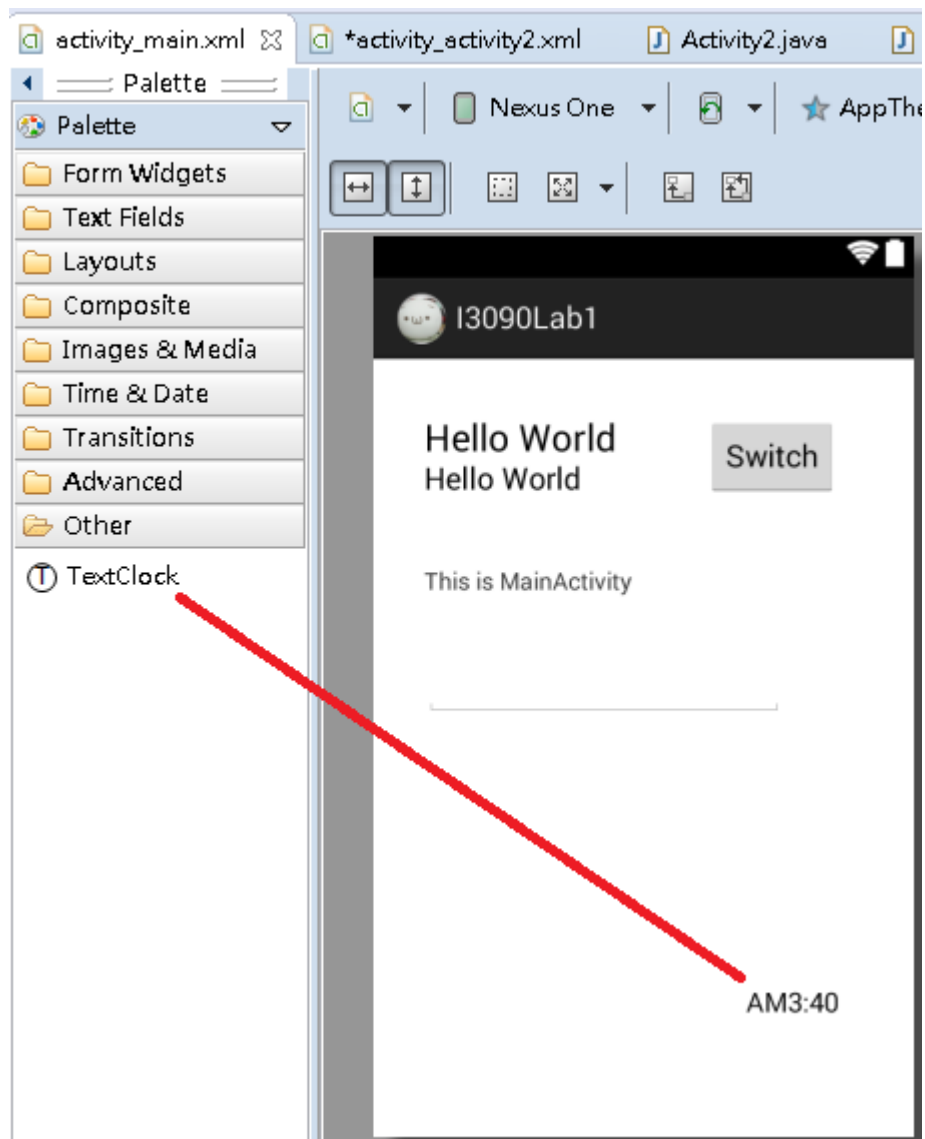


```
1 package com.example.i3090lab1;
2
3 import android.app.Activity;
11
12 public class Activity2 extends Activity {
13
14     private Button Finish;
15     private TextView string;
16
17     @Override
18     protected void onCreate(Bundle savedInstanceState) {
19         super.onCreate(savedInstanceState);
20         setContentView(R.layout.activity_activity2);
21
22         Finish = (Button) findViewById(R.id.button1);
23         string = (TextView) findViewById(R.id.textView1);
24
25         Bundle bundle;
26         bundle = this getIntent().getExtras();
27         string.setText(bundle.getString("KEY_TEXT"));
28         Finish.setOnClickListener( new OnClickListener(){
29             public void onClick(View v){
30                 finish();
31             }
32         });
33     }
34 }
```

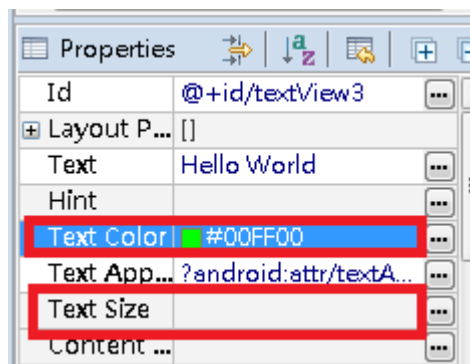
19. 從左邊直接把元件拉到預覽畫面，就完成了中型字體的新增



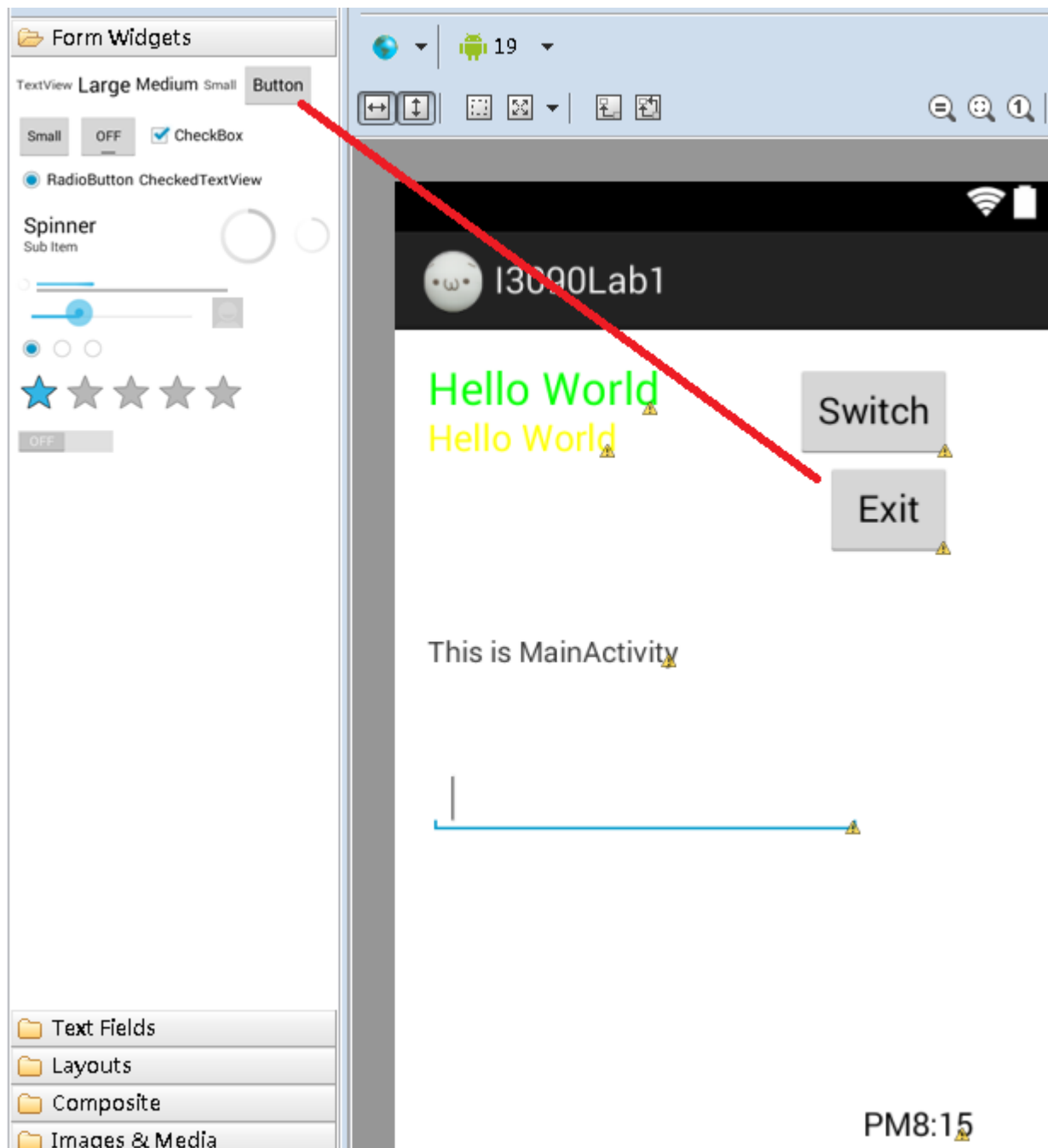
20. 從左邊直接把元件拉到預覽畫面，就完成了文字時鐘的新增



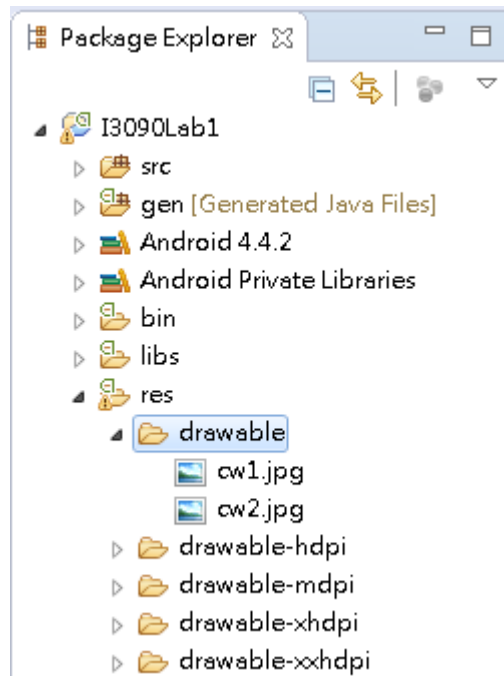
21. 點選字體可以從屬性修改顏色和大小



22. 同之前步驟新增一個離開按鈕



23. 先在res目錄底下新增一個drawable資料夾，並把背景圖放進去



24. 在activity_main.xml 加上紅色那行程式碼

```
activity_main.xml
1 <RelativeLayout xmlns:android="http://schemas.android.com/apk/res/android"
2   xmlns:tools="http://schemas.android.com/tools"
3   android:background="@drawable/cw1"
4   android:layout_width="match_parent"
5   android:layout_height="match_parent"
6   android:paddingBottom="@dimen/activity_vertical_margin"
7   android:paddingLeft="@dimen/activity_horizontal_margin"
8   android:paddingRight="@dimen/activity_horizontal_margin"
9   android:paddingTop="@dimen/activity_vertical_margin"
10  tools:context="com.example.i3090lab1.MainActivity" >
```

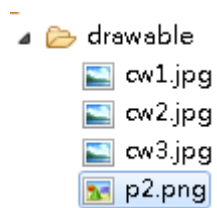
25. 背景圖就新增完成了



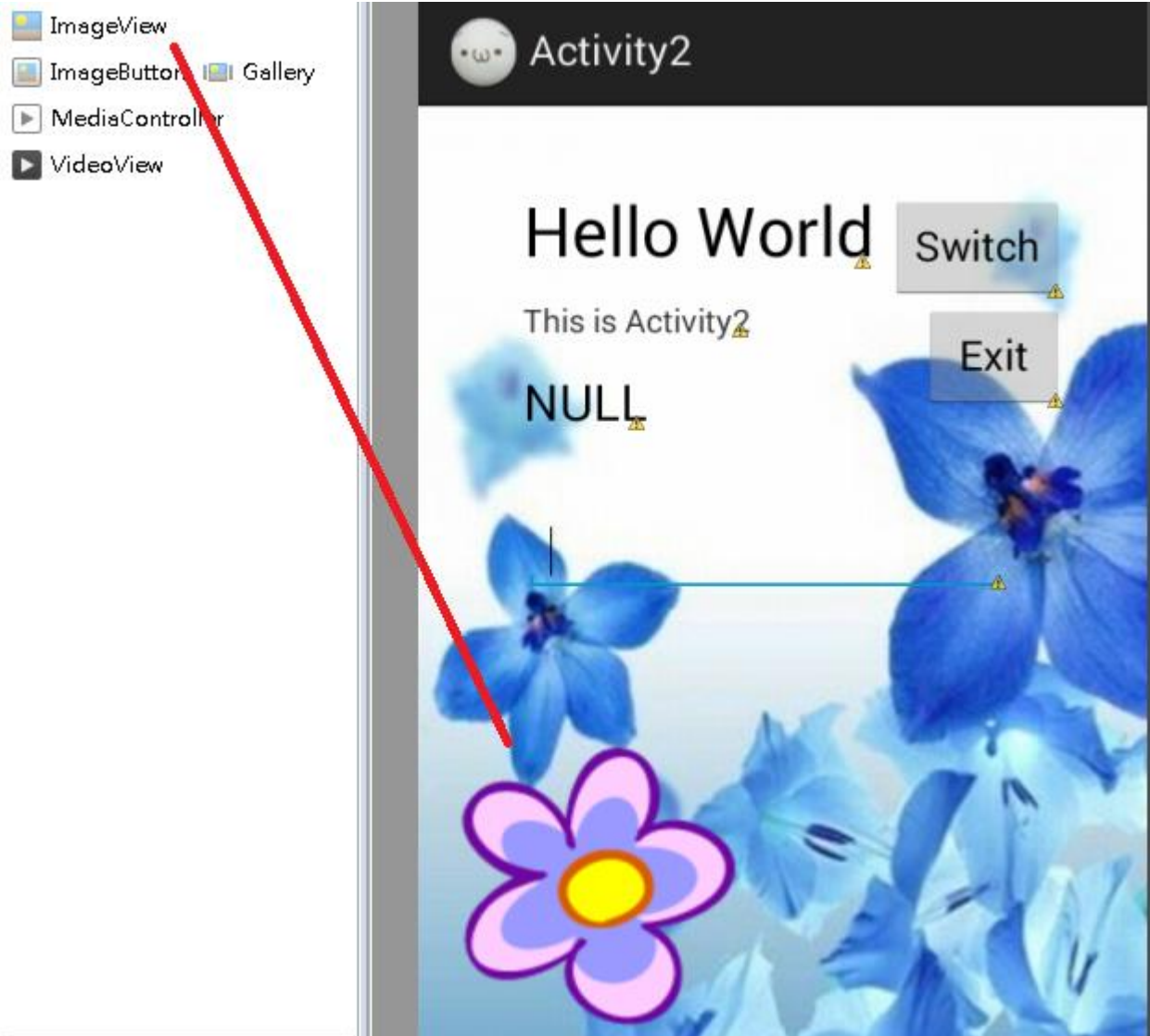
26.activity_activity2.xml 也用同樣方式新增按鈕並修改背景



27. 先在drawable放進新的圖片



28. 接著依照圖示把剛剛放進去的圖片拉近去，就完成圖片新增



29. 如圖修改MainActivity.java程式碼

```

private Button Switch;
private EditText edit;
private Button Finish;
//private TextView string;

@Override
protected void onCreate(Bundle savedInstanceState) {
    super.onCreate(savedInstanceState);
    setContentView(R.layout.activity_main);

    Switch = (Button) findViewById(R.id.button1);
    edit = (EditText) findViewById(R.id.editText1);

    Finish = (Button) findViewById(R.id.button2);

    Switch.setOnClickListener(new OnClickListener(){
        public void onClick(View v){
            Intent intent = new Intent();
            intent.setClass(MainActivity.this,Activity2.class);
            Bundle bundle = new Bundle();
            bundle.putString("KEY_TEXT", edit.getText().toString());
            intent.putExtras(bundle);
            startActivity(intent);
        }
    });

    Finish.setOnClickListener( new OnClickListener(){
        public void onClick(View v){
            finish();
        }
    });
}

```

30. 如圖修改Activity2. java程式碼

```

private Button Switch;
private EditText edit;
private TextView string;
private Button Finish;

@Override
protected void onCreate(Bundle savedInstanceState) {
    super.onCreate(savedInstanceState);
    setContentView(R.layout.activity_main);
    setContentView(R.layout.activity_activity2);

    Switch = (Button) findViewById(R.id.button1);
    Finish = (Button) findViewById(R.id.button2);
    edit = (EditText) findViewById(R.id.editText1);
    string = (TextView) findViewById(R.id.textView1);
    Bundle bundle;
    bundle = this getIntent().getExtras();
    string.setText(bundle.getString("KEY_TEXT"));

    string = (TextView) findViewById(R.id.textView1);

    bundle = this getIntent().getExtras();
    string.setText(bundle.getString("KEY_TEXT"));

    Finish.setOnClickListener( new OnClickListener(){
        public void onClick(View v){
            finish();
        }
    });

    Switch.setOnClickListener(new OnClickListener(){
        public void onClick(View v){
            Intent intent = new Intent();
            intent.setClass(Activity2.this,Activity3.class);
            Bundle bundle = new Bundle();
            bundle.putString("KEY_TEXT", edit.getText().toString());
            intent.putExtras(bundle);
            startActivity(intent);
        }
    });
}

```

31. 依照先前步驟新增Activity3，並修改程式碼

```

private Button Finish;
private TextView string;

@Override
protected void onCreate(Bundle savedInstanceState) {
    super.onCreate(savedInstanceState);
    setContentView(R.layout.activity_activity3);

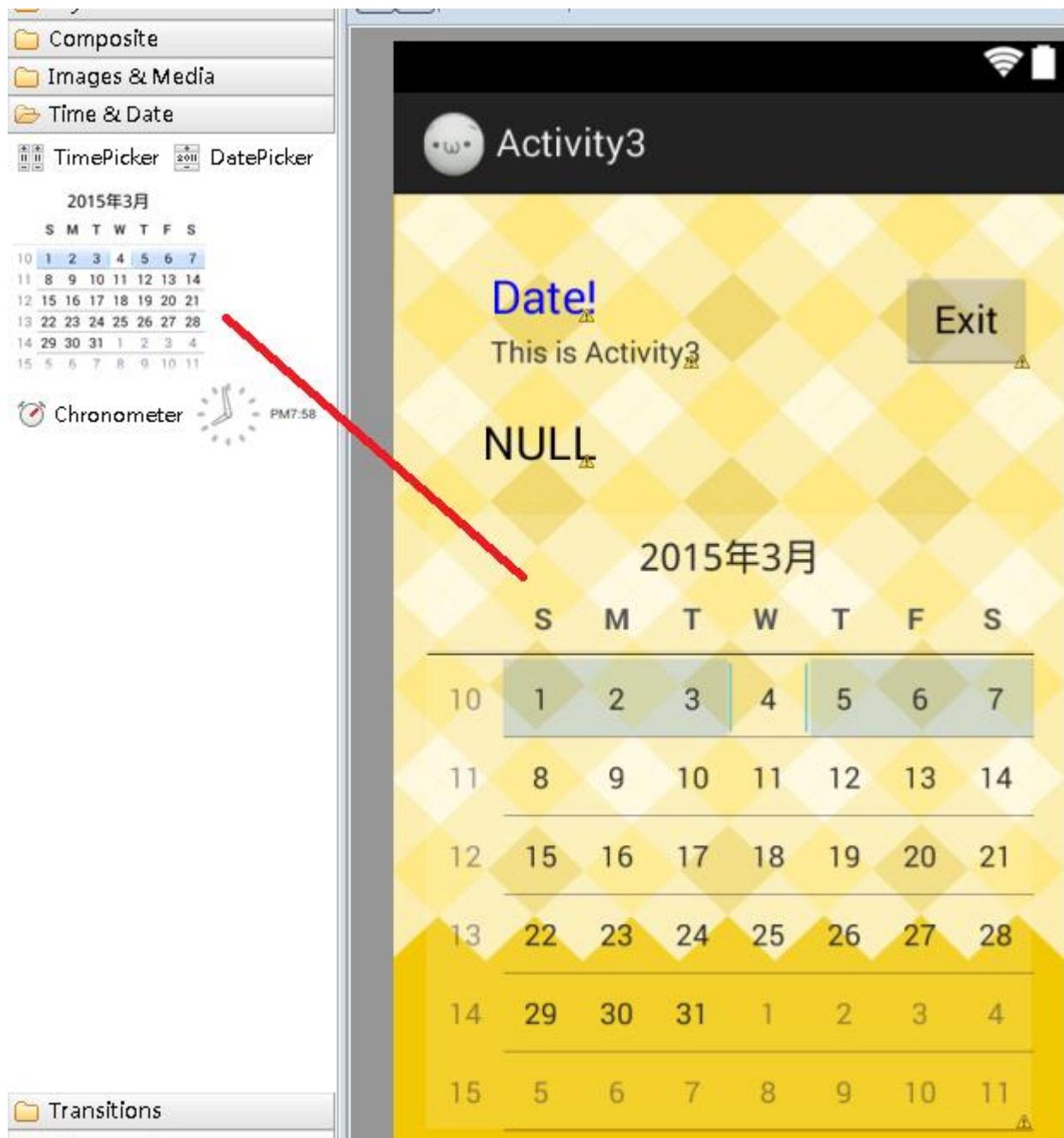
    Finish = (Button) findViewById(R.id.button1);
    string = (TextView) findViewById(R.id.textView1);

    Bundle bundle;
    bundle = this.getIntent().getExtras();
    string.setText(bundle.getString("KEY_TEXT"));

    Finish.setOnClickListener( new OnClickListener(){
        public void onClick(View v){
            finish();
        }
    });
}

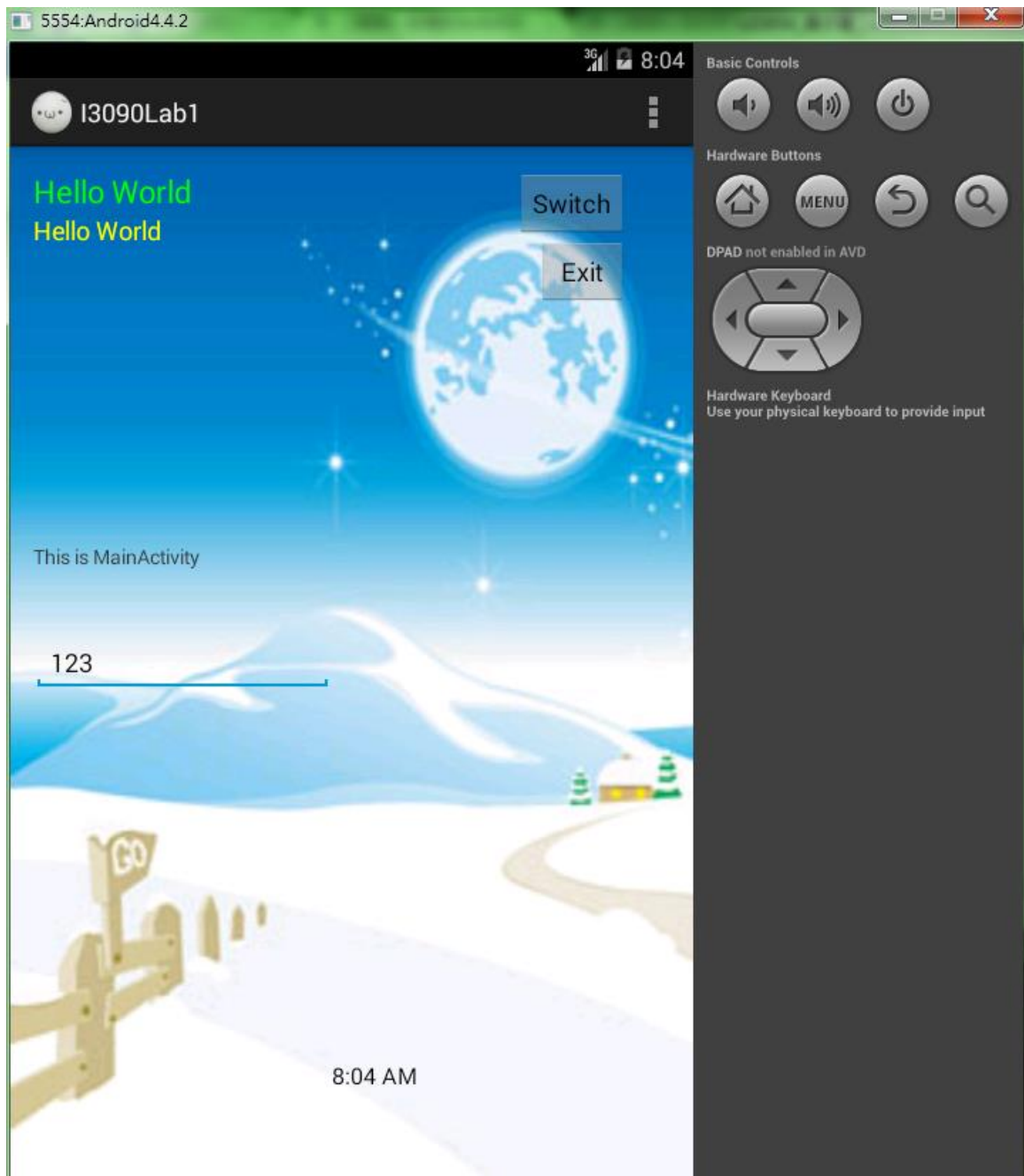
```

32.Activity3依照先前步驟修改，接著如圖拉出日期

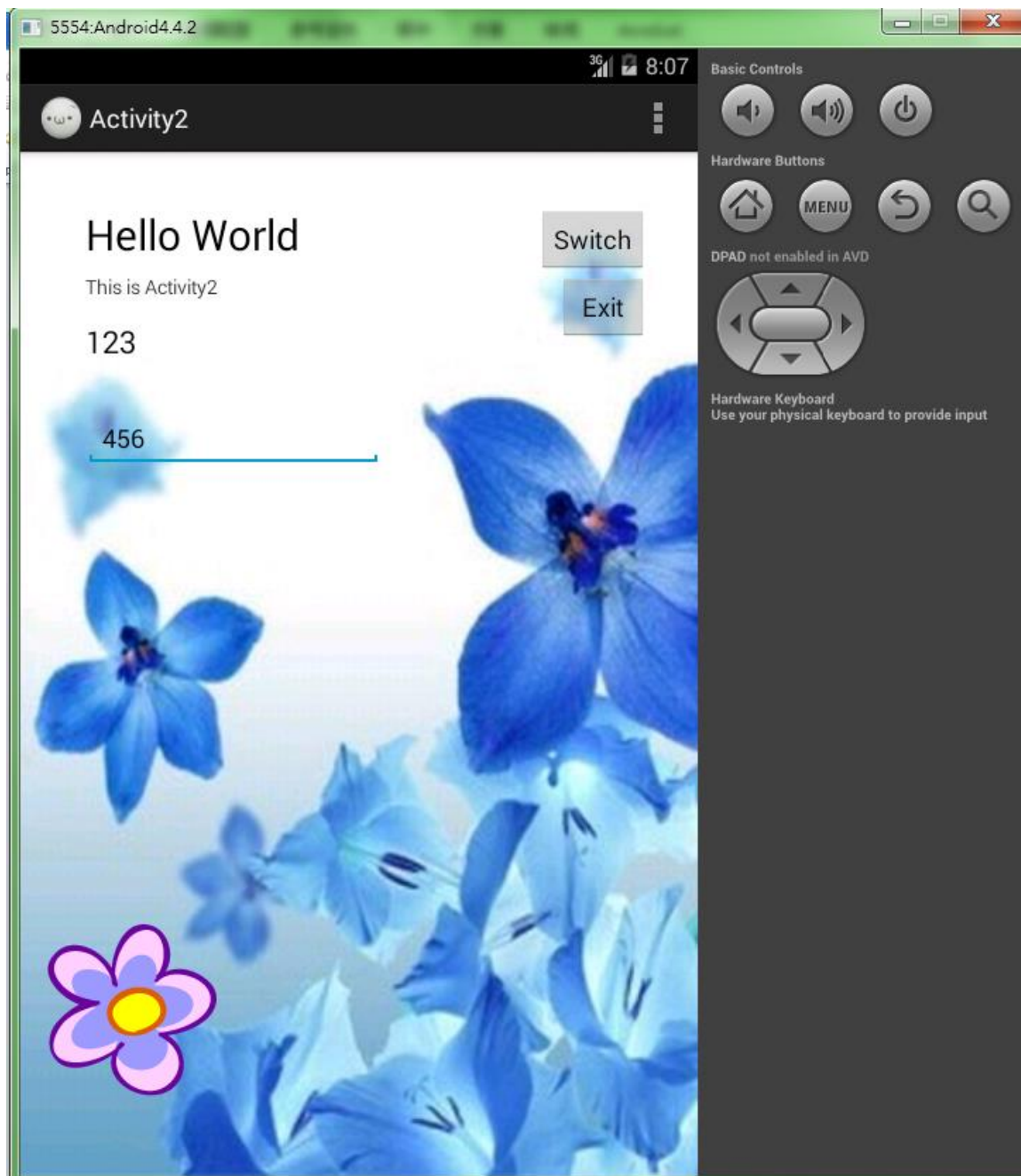


實驗結果

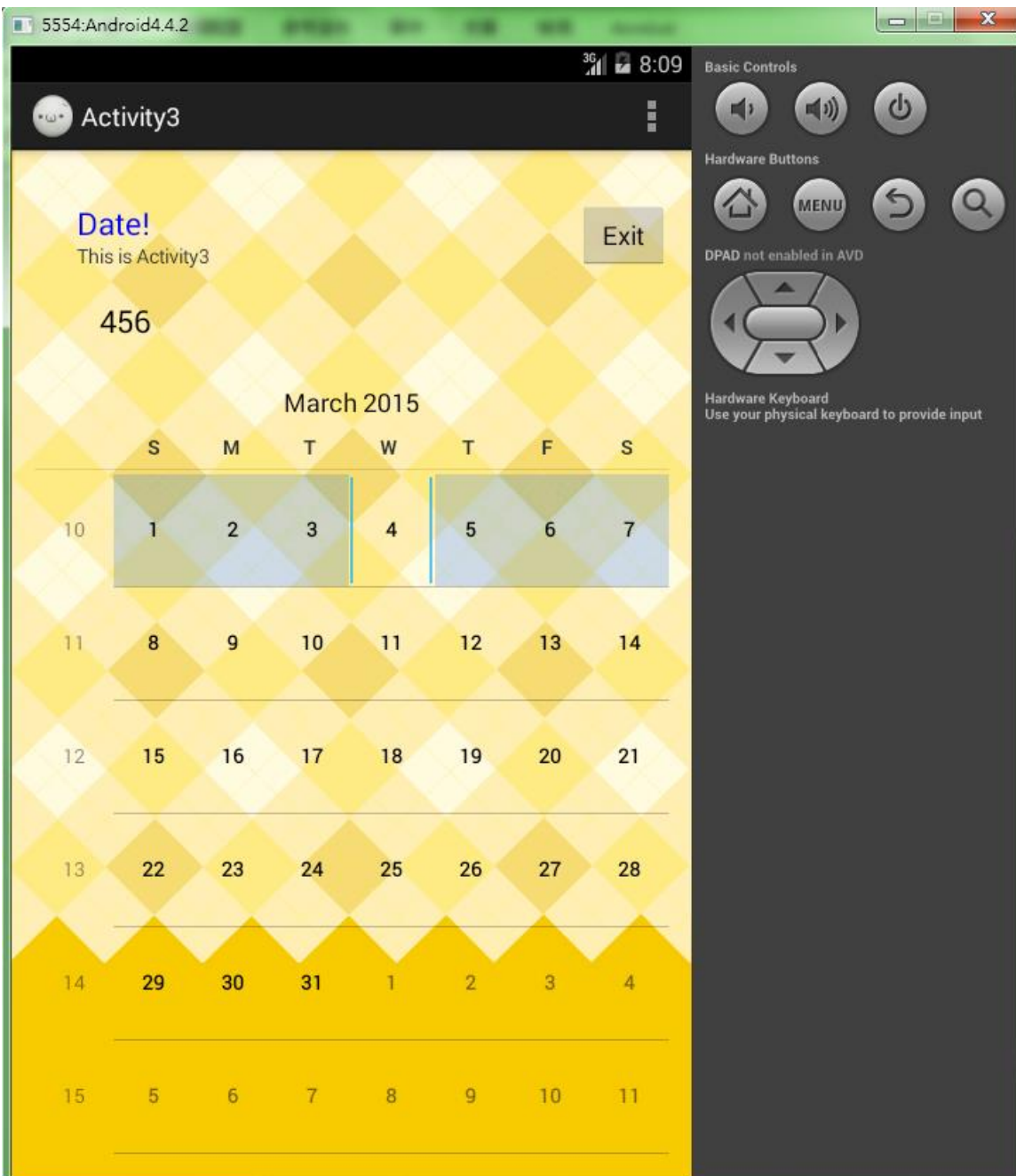
這是 MainActivity 的執行畫面，在輸入框裡輸入123，接著按下Switch
就會跑到Activity2，按下Exit就會返回上一頁(離開)



Activity2負責接收 MainActivity 傳送過來的資料，並且顯示，接著在輸入框裡輸入456，在按下Switch後會跑去Activity3，同樣的按Exit可以返回上一頁



Activity3負責接收 Activity2 傳送過來的資料，並且顯示，最後按 Exit可以返回上一頁



實驗紀錄

在實驗中遇到一些問題

Q: 專案不能匯入?

Ans: 匯出的時候檔名重複，以至於造成匯出的檔案出錯，接著就不能匯入，這問題超級重要，重做了兩次後在各種試驗下才了解。

Q: 怎麼匯出跟匯入專案?

Ans: 有可能因為檔名重複造成匯出失敗所以想到以下方法

匯出:File-Export-General-Archive File-next-Select All-To
archive file-Finish

這樣一來名稱重複也不會匯出失敗

接著要匯入的話先把檔案解壓縮

匯入:File-Import-General-Existing Projects into
Workspace-next-選擇解壓縮後的資料夾-Finish

如此一來就不會發生錯誤了

Q: 在步驟4結束後，沒辦法跑出步驟5的圖?

Ans: 去翻了Lab 0的講義，才發現到原來是少了新增AVD裝置，之後照著Lab 0的設定在用一次就成功了。

Q: 不小心刪掉activity2 怎麼加回來?

Ans: 重新看步驟在照著做一次就可以加回來了。

Q: 程式碼沒顯示行數?

Ans: Google後才找到設定

Window-Preferences-General-Editors-Text Editors-Show line numbers 打勾。

Q: 其他功能怎麼做出來?

Ans: 自己先動手試著亂做，如果不行做出來就請教同學或google。

思考分析

第一次做嵌入式系統專題實驗，實際實驗後學到了很多東西，像是一個介面會有一個xml檔案，透過拉元件的方式可以輕易產生一個功能，接著該元件的程式碼會馬上出現在xml檔裡面。而透過更改裡面的程式碼，可以進行簡易的字串或屬性變更，自己也有試著拉出其他元件，不過有些元件功能看不太懂還要再慢慢嘗試。然後是java寫程式的部分，這次的程式可以在Activity間傳遞參數，有點看不太懂，不過自己有試著去修改一些程式碼更改功能，讓我想到之後的實驗或許會有更多不同的變化令人期待。接著上網google找了相關資料，練習修改背景圖，自己也用元件拉出日曆並新增第三個Activity和圖片，看到班上很多人做出許多不同功能，想著如果大家能一起討論新增各種不同功能一定會成為非常有趣的手機程式。