Android程式設計

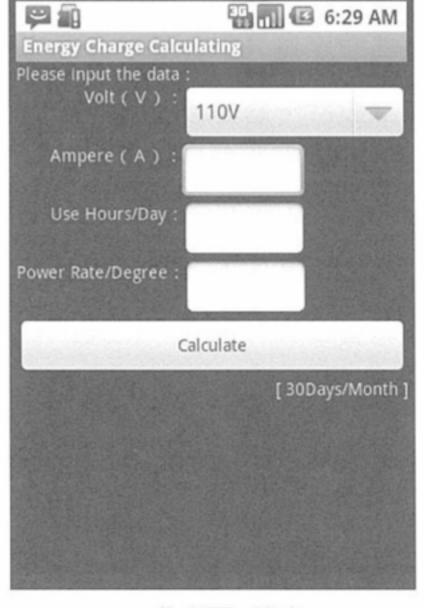
電費計算機

王昱景 Brian Wang brian.wang.frontline@gmail.com

1. 題目說明:

請開啓 COM.TQC.GDD01 專案,設計「電費計算機」應用程式。在選擇伏特、輸入安培及輸入每天使用時數與每度電費後,按下「計算電器每月電費」按鈕,即可在下方顯示金額,並可依模擬器語系變更程式的語系,請依下列題意完成作答。

P 1	日 日 日 日 日 日 日 日 日 日 日 日 日 日 日 日 日 日 日
電費計算機	
請輸入以下資料	後送出:
伏特(V):	110V -
安培(A):	
每天使用時數:	
毎度電費 :	
	計算電器毎月電費
	[月皆以30天計算]



<參考圖-繁體中文>

<參考圖-英文>

2. 設計說明:

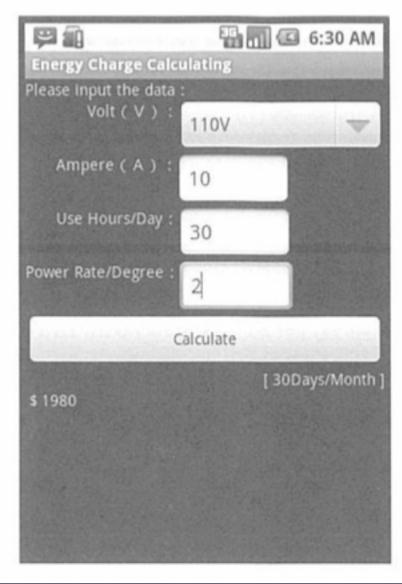
- (1) 專案中已預先設計好 values/strings.xml 及 values-en/strings.xml 檔案, 此兩個檔案變數名稱相同,其中 values/strings.xml 為繁體中文語系時 所使用之檔案, values-en/strings.xml 為英文語系時所使用之檔案。
- (2) 請開啓 main.xml 檔案,依參考圖利用 RelativeLayout 製作出電費計算機的主要畫面,且位置需對齊如參考圖。
- (3) 參考圖中的【伏特(V)】為一 Spinner, string 名稱為 option_v, 變數 名稱為 input_v, 變數陣列為 option_vs, item 有兩個為【110V】和【220V】。

- (4) 請設計一個變數名稱為 input_a 的 EditText, 供使用者輸入【安培(A)】, 且 inputType= numberDecimal。
- (5)請設計一個變數名稱為 input_hr 的 EditText,供使用者輸入【每天使用時數】,且限制輸入數值(整數)。
- (6)請設計一個變數名稱為 input_m 的 EditText,供使用者輸入【每度電費】,且限制輸入數值(整數)。
- (7)請設計一個 string 名稱為 btn 和變數名稱為 submit 的 Button, string 值設定為【計算電器每月電費】。
- (8)請設計一個變數名稱為 respond 的 TextView, text 值預設為空,於點選「計算電器每月電費」後,於此控制項顯示總電費。
- (9) 點選「計算電器每月電費」的按鈕時,計算出電費總數,顯示【\$xxx】, 將總電費代入 xxx 中。
- (10) 電費公式:電費 =(瓦數/1000)*(每天使用時數*30天)*每度電費。
- (11) W (瓦數) = V (伏特) * A (安培)
- (12) 將模擬器的語系切換為 en 英文, 再次執行程式, 界面以英文呈現。將模擬器的語系切換為 zh-TW 繁體中文, 再次執行程式, 界面以繁體中文呈現。

3. 執行結果參考畫面:

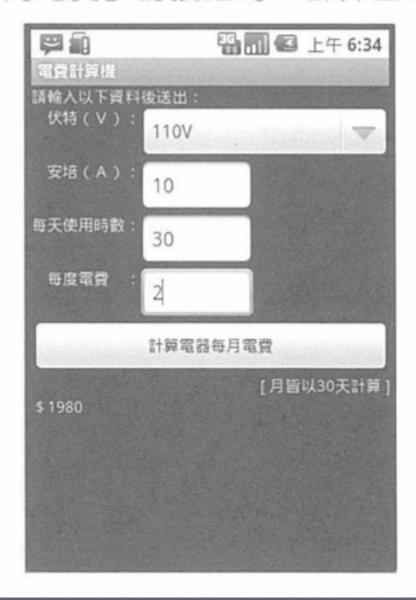
(1) 將模擬器的語系切換為 en 英文, 再次執行程式, 界面以英文呈現。點選「Calculate」的按鈕時, 計算出電費總數。





(2) 將模擬器的語系切換為 zh-TW 繁體中文,再次執行程式,界面以繁體中文呈現。點選「計算電器每月電費」的按鈕時,計算出電費總數。





4. 自行測試程式是否達成下列結果:

- (1) 如參考圖所示,畫面元件依 res/layout/main.xml 的定義正確顯示。
- (2) 變數名稱為 input_a 的 EditText, 其 inputType= numberDecimal。
- (3) 變數名稱為 input_hr 及 input_m 的 EditText, 需限制輸入數值。
- (4) 變數名稱為 response 的 TextView, text 值預設為空。

- (5) 點選「計算電器每月電費」的按鈕,依公式計算出電費總數。
- (6) 依據模擬器的語系變更,可切換程式為繁體中文或英文語系。

5. 評分項目:

項			配	分	得	分
All Discourses and the restriction of the pro-	畫面元件依 res/layout/main.xml 的定義正確顯示,各個物件及其屬性設定與參考圖相同		4			
(2) 變數名稱為 inp numberDecimal	2) 變數名稱為 input_a 的 EditText, 其 inputType= numberDecimal		2	2		
(3) 變數名稱為 input_hr 及 input_m 的 EditText,需限制輸入數值		4	4			
(4) 變數名稱為 response 的 TextView, text 值預設為空		2	2			
(5) 點選「計算電器每月電費」的按鈕,依公式計算出電費總數並正確顯示		2	4			
(6) 依據模擬器的語系變更,可切換程式為繁體中文 或英文語系		2	4			
總		分	2	.0		

PowerCalculate

- 開啟和執行 Android 專案
- 建立元件的版面配置
- 建立 Activity 活動類別

開啟和執行 Android 專案

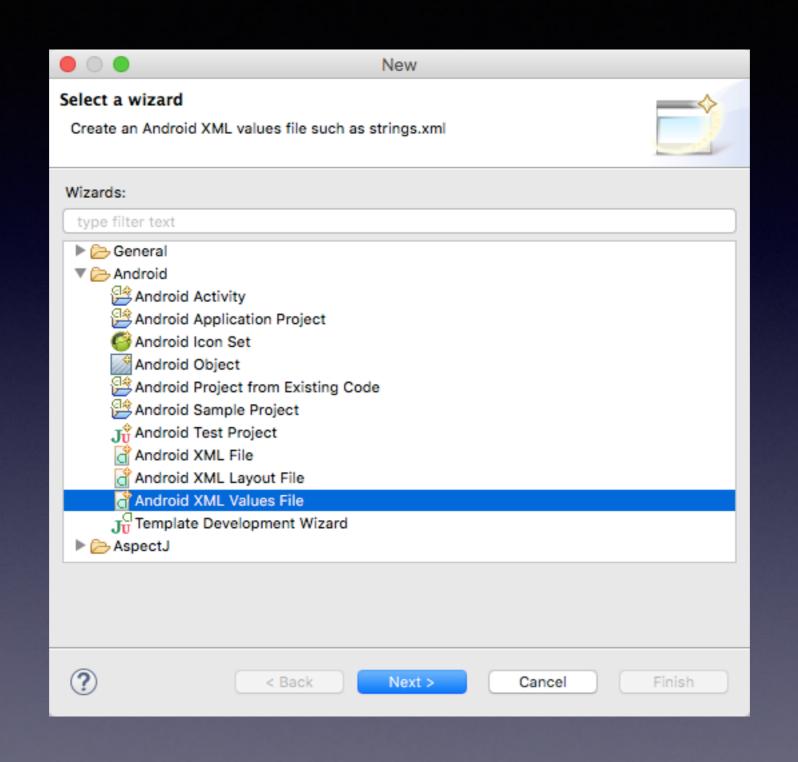
- 請啟動 Eclipse IDE
- 建立 Android 專案
 - Project Name: PowerCalculate
 - Build Target: Android 4
 - Package Name: tw.edu.vnu.powercalculate

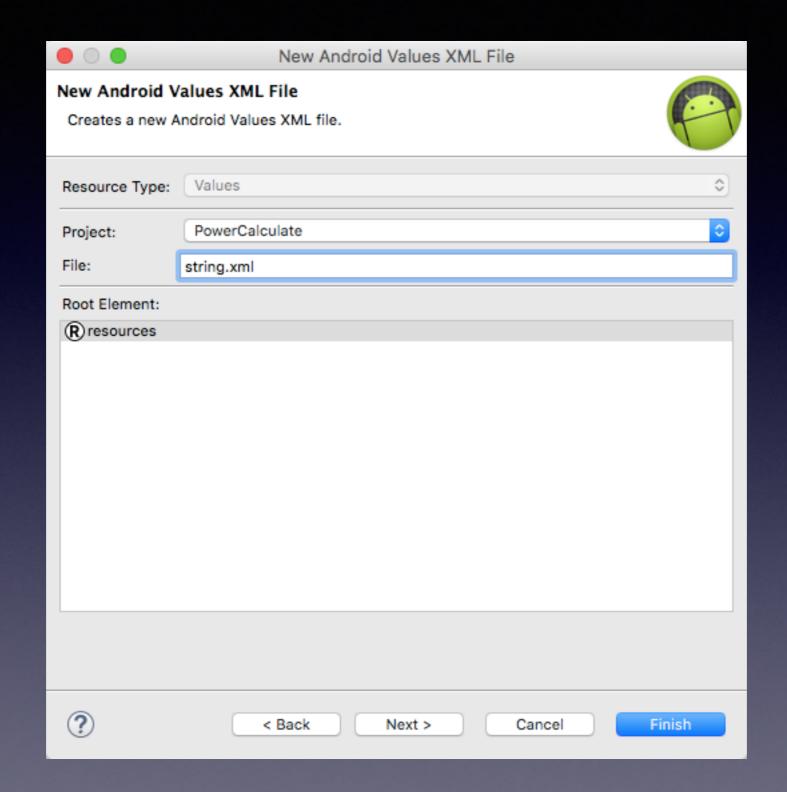
建立元件的版面配置

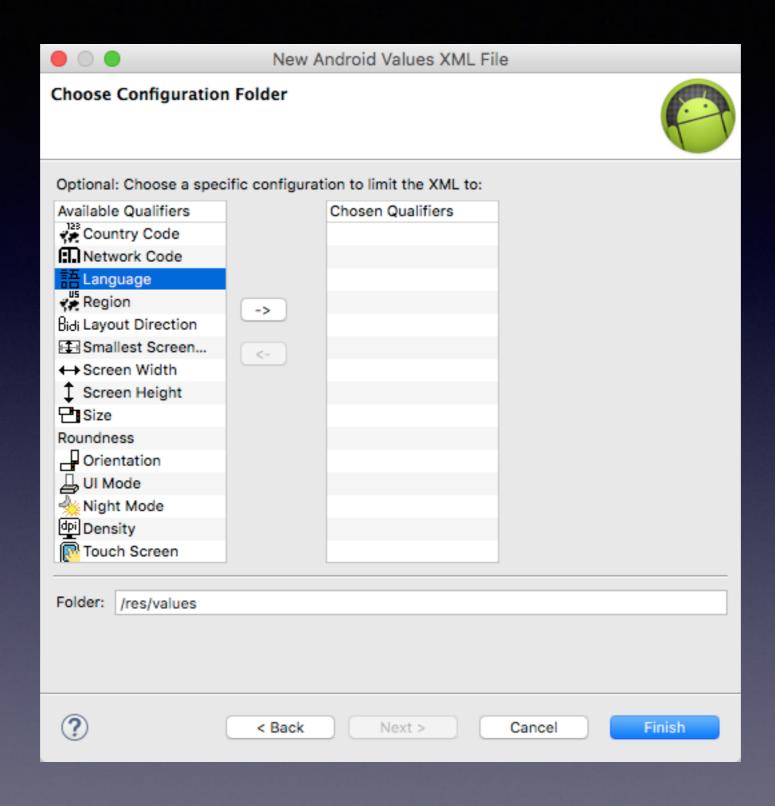
• 刪除 "Hello World" TextView

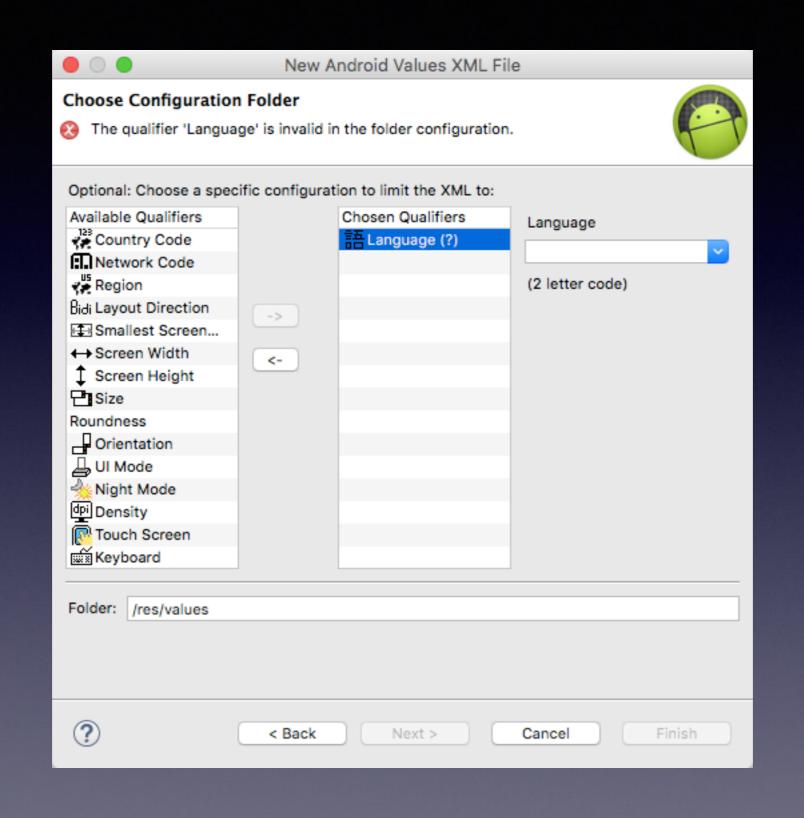
```
    在 RelativeLayout 依序編排 TextViwe、
        TextViwe、Spinner、TextViwe、
        EditText(NumberDecimal)、TextViwe、
        EditText(Number)、TextViwe、
        EditText(Number)、TextViwe、Button 和
        TextViwe 元件
```

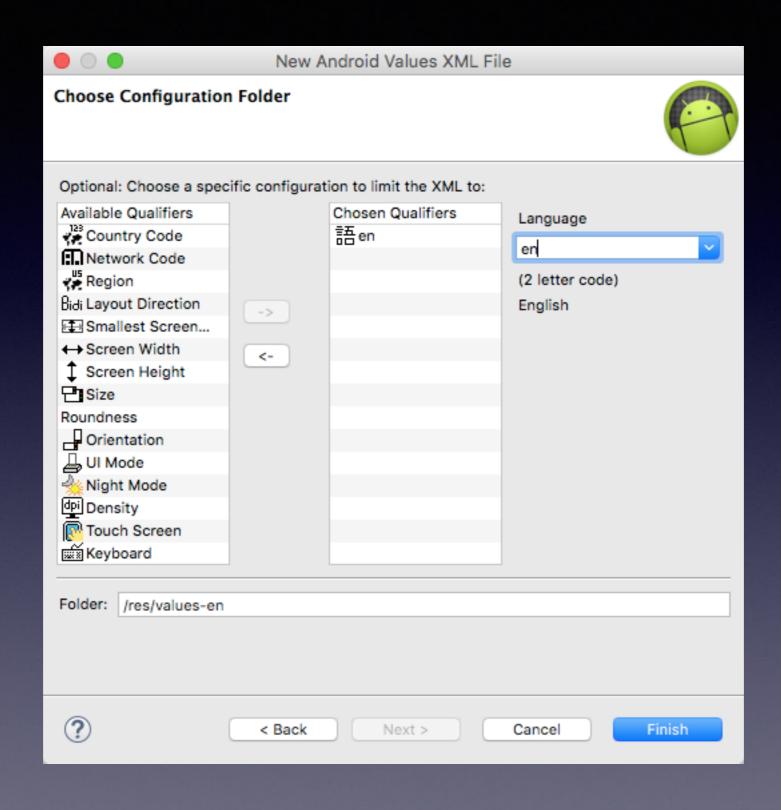
• 增加 en 語系的 strings.xml 設定檔

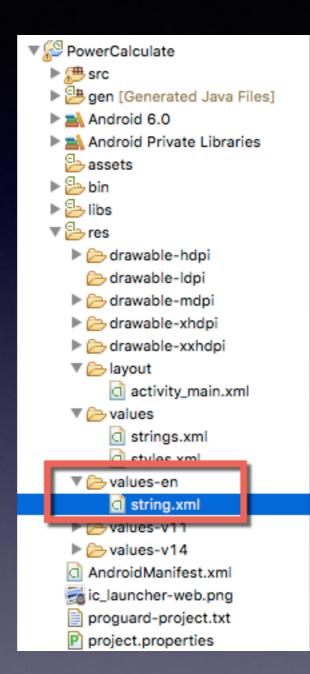










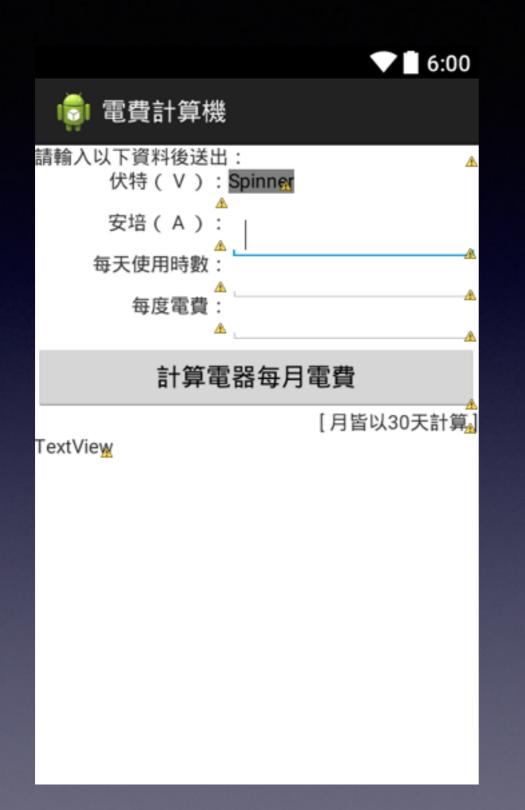


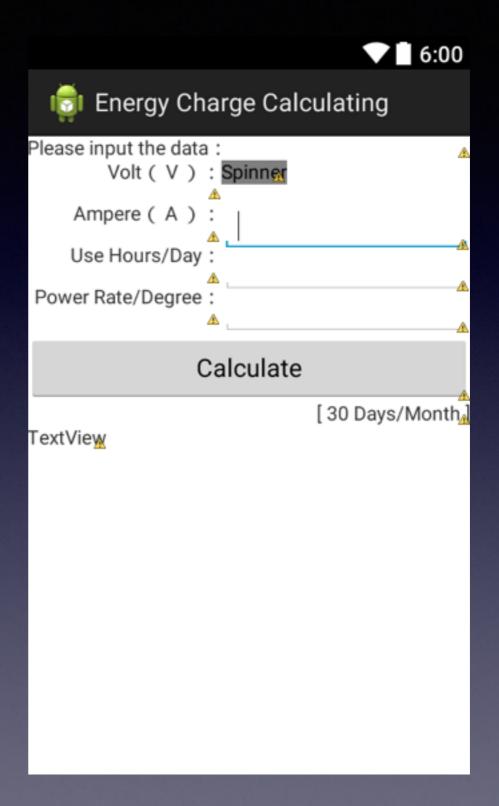
- 刪除 string.xml 的 hello_world,修改 app_name 為"電費計算機",英文為"Energy Charge Calculating"
- 將第一個 TextView 新增 title 的 string 值中文為 "請輸入以下資料後送出:",英文為 "Please input the data:"
- 第二個 TextView 新增 option_v 的 string 值為"伏特(V);"特(V);"

- 第三個 TextView 新增 option_a 的 string 值為 "安培(A):",英文為 "Ampere(A):"
- 第四個 TextView 新增 option_hr 的 string 值為 "每天使用時數:",英文為"Use Hours/Day:"
- 第五個 TextView 新增 option_m 的 string 值為 "每 度電費:",英文為 "Power Rate/Degree:"

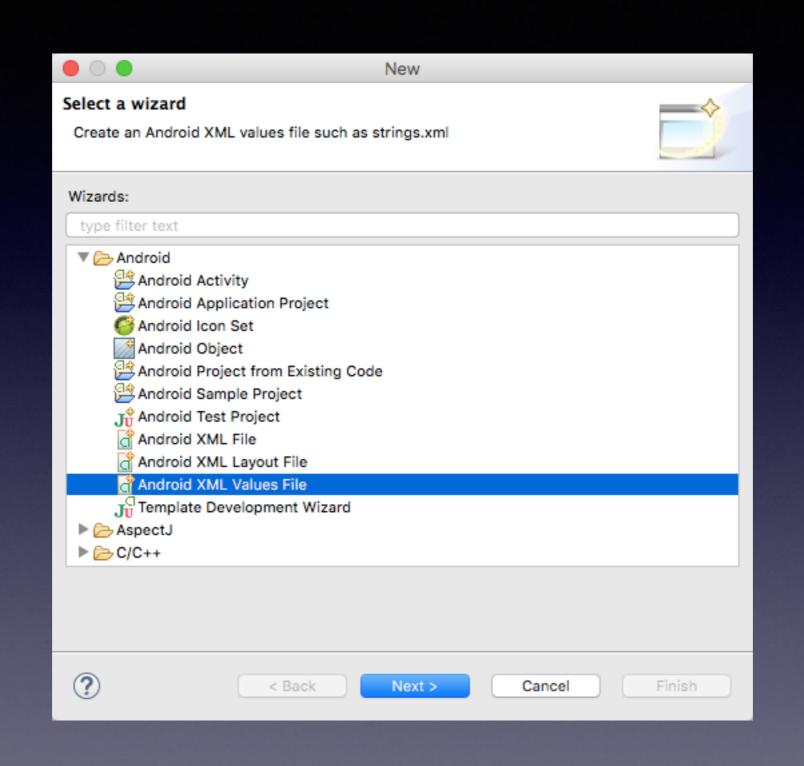
- 第六個 TextView 新增 note 的 string 值為 "[月皆以30天計算]",英文為 "[30 Days/Month]"
- 將第七個 TextView 的 id 設定為 respond,並清空 text
- 將 Spinner 的 id 設定為 input_v

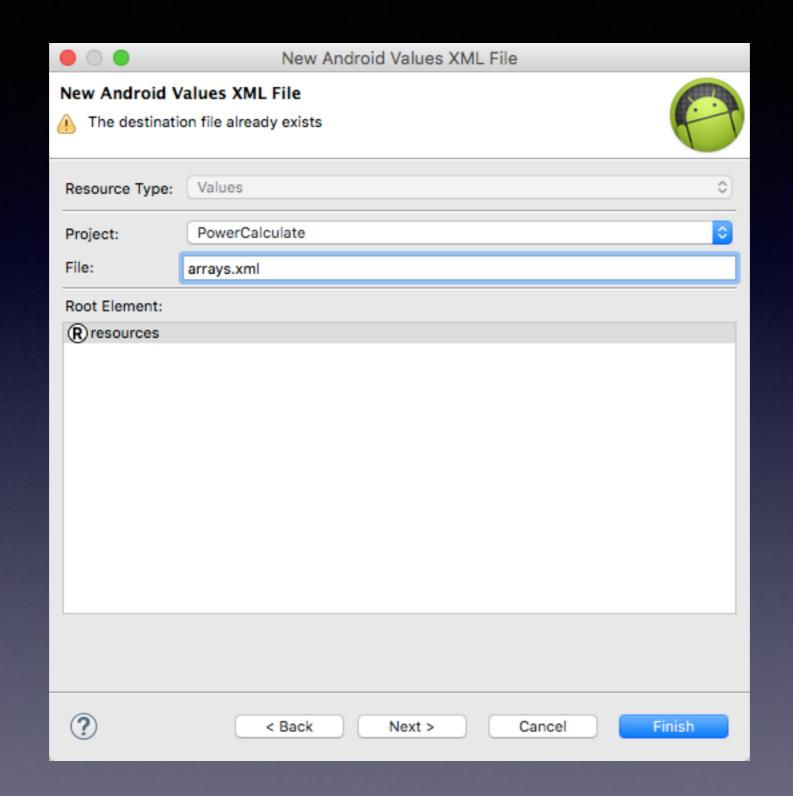
- 將第一個 EditText 的 id 設定為 input_a
- 將第二個 EditText 的 id 設定為 input_hr
- 將第三個 EditText 的 id 設定為 input_m
- 將 Button 的 id 設定為 submit, on Click 設定為 calculate_Click, 並新增 btn 的 string 值中文為 "計算電器每月電費", 英文為 "Calculate"

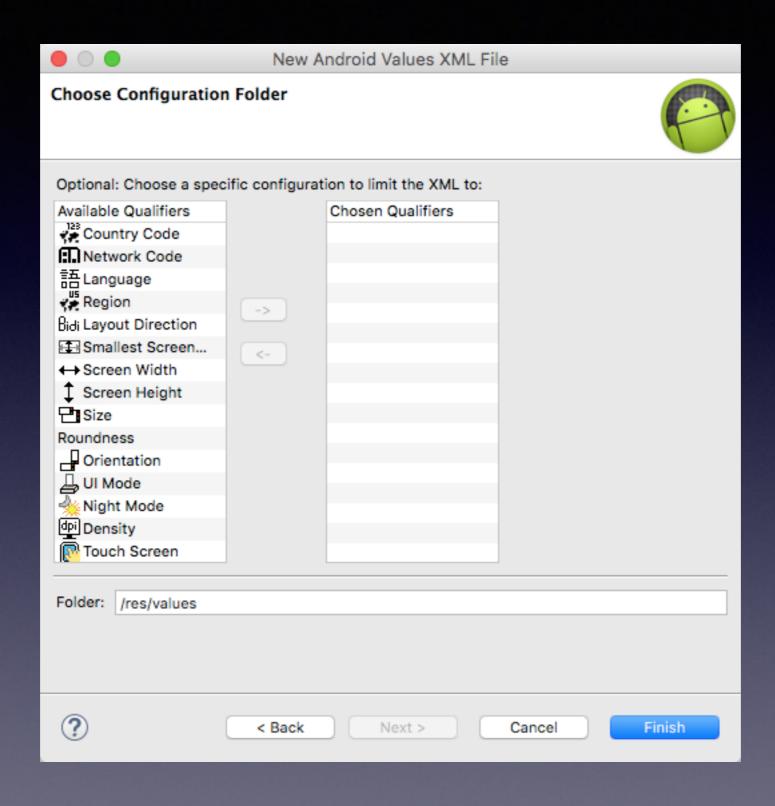


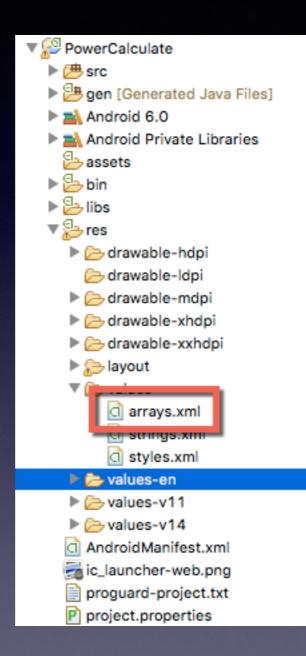


• 增加 arrays.xml 設定檔

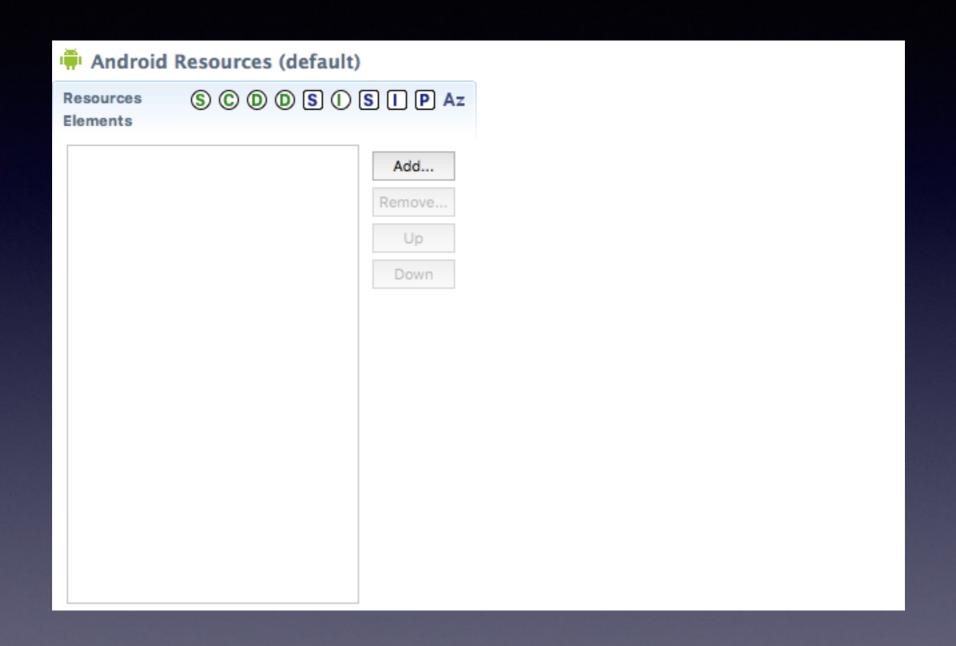


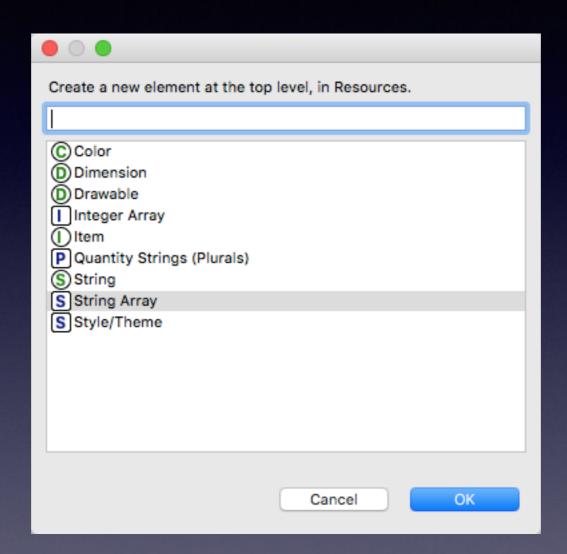


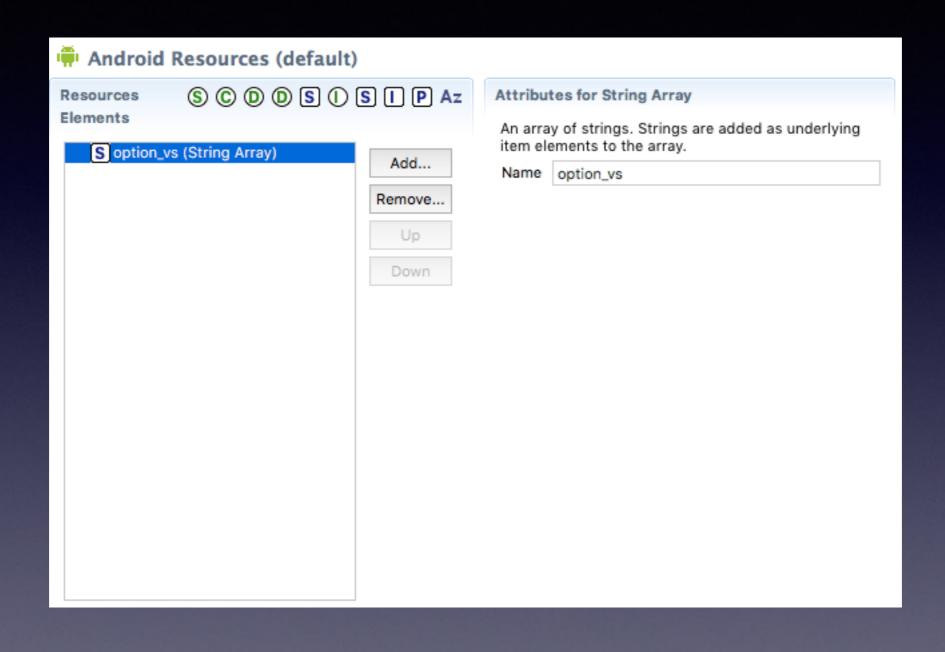




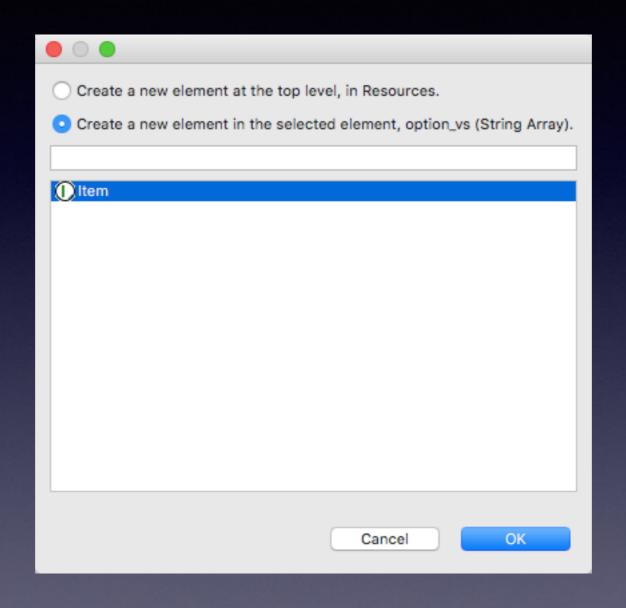
• 新增 option_vs 的 String Array

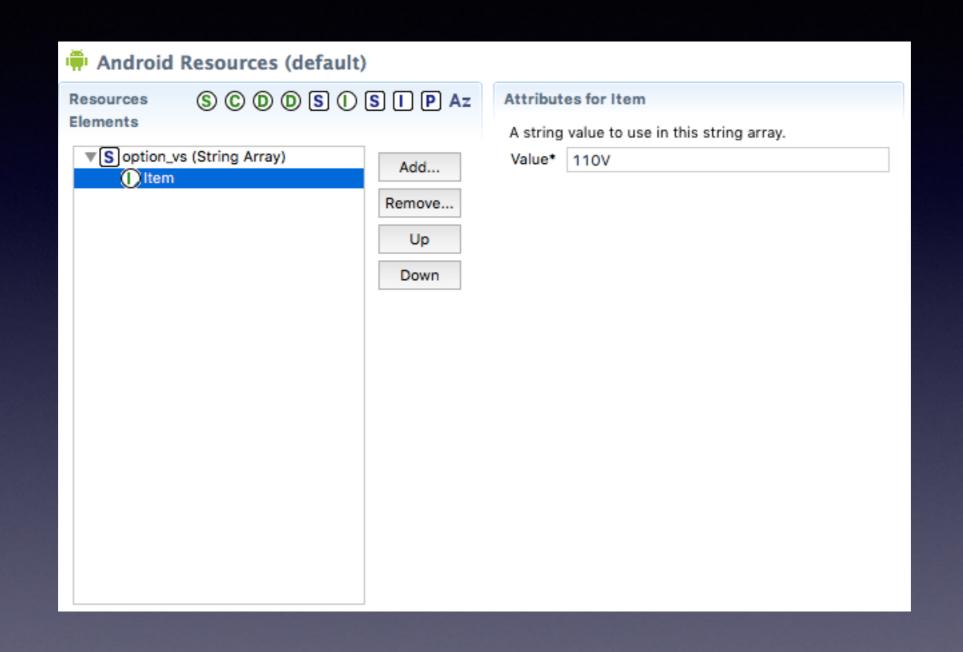






• 新增 110V 和 220V 的 Item





res / layout / activity_main.xml

```
<RelativeLayout xmlns:android="http://schemas.android.com/apk/res/android"</pre>
   xmlns:tools="http://schemas.android.com/tools"
    android:layout_width="match_parent"
    android:layout_height="match_parent" >
    <TextView
        android:id="@+id/textView1"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:layout_alignParentLeft="true"
        android:layout_alignParentTop="true"
        android:text="@string/title" />
    <TextView
        android:id="@+id/textView5"
        android:layout_width="wrap_content"
        android:layout_height="30dp"
       android:layout_alignRight="@+id/textView4"
        android:layout_below="@+id/textView4"
        android:gravity="right"
        android:text="@string/option_m" />
    <TextView
        android:id="@+id/textView2"
        android:layout_width="140dp"
        android:layout_height="30dp"
        android:layout_alianParentLeft="true"
        android:layout_below="@+id/textView1"
        android:gravity="right"
        android:text="@string/option_v" />
```

```
<TextView
    android:id="@+id/textView3"
    android:layout_width="wrap_content"
    android:layout_height="30dp"
    android:layout_alignRight="@+id/textView2"
    android:layout_below="@+id/textView2"
    android:gravity="right"
    android:text="@string/option_a" />
<TextView
    android:id="@+id/textView4"
    android:layout_width="wrap_content"
    android:layout_height="30dp"
    android:layout_alianRight="@+id/textView3"
    android:layout_below="@+id/textView3"
    android:gravity="right"
    android:text="@string/option_hr" />
<Spinner
    android:id="@+id/input_v"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:layout_alignTop="@+id/textView2"
    android:layout_toRightOf="@+id/textView2"
    android:entries="@array/option_vs" />
<EditText
    android:id="@+id/input_a"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:layout_alignTop="@+id/textView3"
    android:layout_toRightOf="@+id/textView3"
    android:ems="10"
    android:inputType="numberDecimal" >
    <requestFocus />
</EditText>
```

```
<EditText
   android:id="@+id/input_hr"
   android:layout_width="wrap_content"
   android:layout_height="wrap_content"
   android:layout_alignTop="@+id/textView4"
   android:layout_toRightOf="@+id/textView4"
   android: ems="10"
   android:inputType="number" />
<EditText
   android:id="@+id/input_m"
   android:layout_width="wrap_content"
   android:layout_height="wrap_content"
   android:layout_toRightOf="@+id/textView5"
   android:layout_alignTop="@+id/textView5"
   android:ems="10"
   android:inputType="number" />
<Button
   android:id="@+id/submit"
   android:layout_width="match_parent"
   android:layout_height="wrap_content"
   android:layout_alignParentLeft="true"
   android:layout_below="@+id/input_m"
   android:text="@string/btn" />
```

```
<TextView
    android:id="@+id/textView6"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:layout_alignParentRight="true"
    android:layout_below="@+id/submit"
    android:text="@string/note" />

<TextView
    android:id="@+id/respond"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:layout_alignParentLeft="true"
    android:layout_below="@+id/textView6"
    android:text="["]/>
</RelativeLayout>
```

res / values / strings.xml

• res / values-en / strings.xml

res / values / arrays.xml

```
<?xml version="1.0" encoding="utf-8"?>
<resources>
   <string-array name="option_vs">
       <item >110V</item>
       <item >220V</item>
   </string-array>
</resources>
```

建立 Activity 活動類別

- 宣告成員變數
- onCreate() 方法
- setOnClickListener() 事件處理方法

宣告成員變數

```
package com.example.powercalculate;
import android.app.Activity;

public class MainActivity extends Activity {

    private Spinner input_v;
    private EditText input_a;
    private EditText input_hr;
    private EditText input_m;
    private Button submit;
    private TextView respond;

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

onCreate() 方法

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

    input_v = (Spinner) findViewById(R.id.input_v);
    input_a = (EditText) findViewById(R.id.input_a);
    input_hr = (EditText) findViewById(R.id.input_hr);
    input_m = (EditText) findViewById(R.id.input_m);
    submit = (Button) findViewById(R.id.submit);
    respond = (TextView) findViewById(R.id.respond);
```

電費公式

- 電費=(瓦數/1000)*(每天使用時數*30天)*每度電費
- W(瓦數)=V(伏特)*A(安培)
- ((v * a) / 1000) * (hr * 30) * m

setOnClickListener() 事件處理方法

```
submit.setOnClickListener(new OnClickListener() {
    @Override
    public void onClick(View view) {
        if (input_v.getSelectedItem().toString().isEmpty()) {
            Toast.makeText(MainActivity.this, getResources().getString(R.string.error_v), Toast.LENGTH_SHORT).show();
            return;
        if (input_a.getText().toString().isEmpty()) {
            Toast.makeText(MainActivity.this, getResources().getString(R.string.error_a), Toast.LENGTH_SHORT).show();
            return;
        if (input_hr.getText().toString().isEmpty()) {
            Toast.makeText(MainActivity.this, getResources().getString(R.string.error_hr), Toast.LENGTH_SHORT).show();
            return;
        if (input_m.getText().toString().isEmpty()) {
            Toast.makeText(MainActivity.this, getResources().getString(R.string.error_m), Toast.LENGTH_SHORT).show();
            return;
        }
        int v = input_v.getSelectedItem().toString().equals("110V") ? 110 : 220;
        double a = Double.parseDouble(input_a.getText().toString());
        int hr = Integer.parseInt(input_hr.getText().toString());
        int m = Integer.parseInt(input_m.getText().toString());
        int result = (int) (((v * a) / 1000) * (hr * 30) * m);
        respond.setText("$" + result);
});
```

- 講義、範例程式下載:
- https://github.com/ ycwang812/VNU

