

# Android程式設計

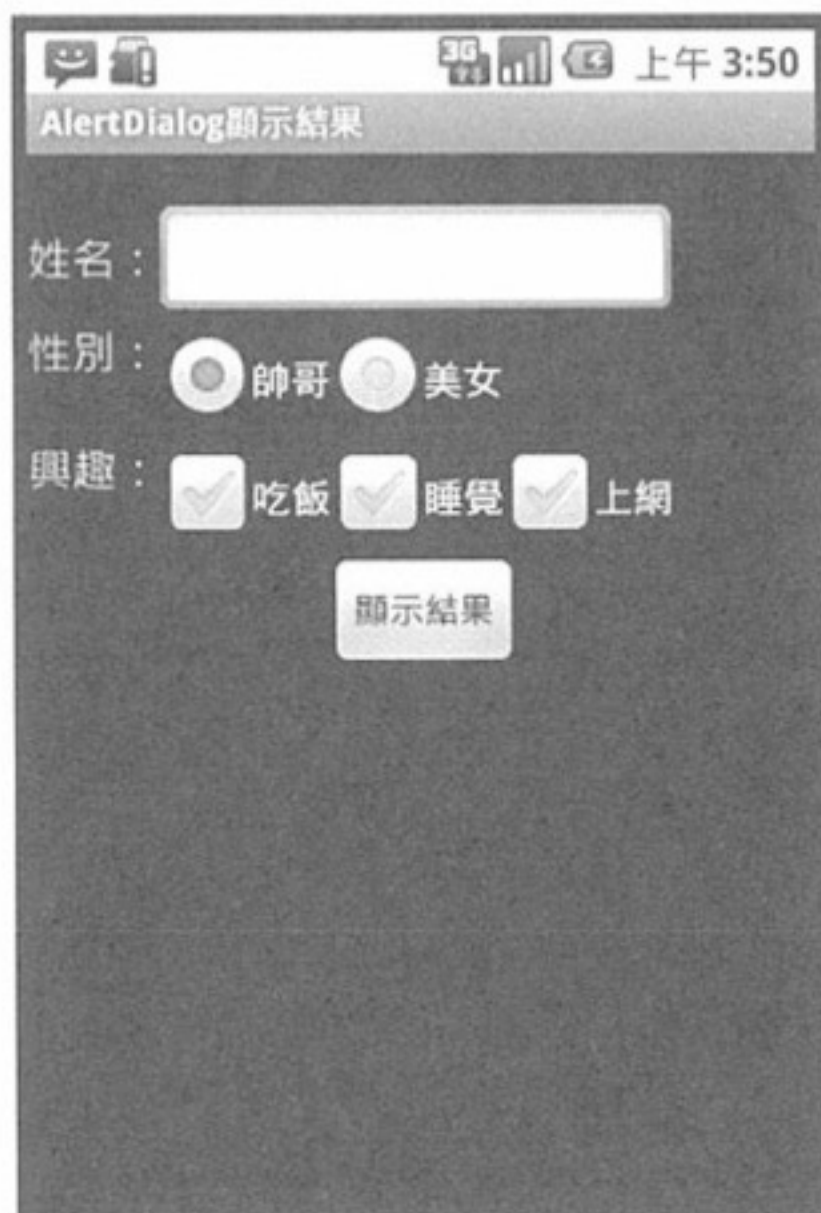
## AlertDialog 顯示結果

王昱景 Brian Wang

[brian.wang.frontline@gmail.com](mailto:brian.wang.frontline@gmail.com)

### 1. 題目說明：

請開啓 **COM.TQC.GDD01** 專案，設計「AlertDialog 顯示結果」應用程式。此一個簡易的表單，內含姓名、性別與興趣欄位，當使用者填寫完表單內的資料後，按下【顯示結果】按鈕，此時程式會收集表單內填寫的資訊，以跳出 AlertDialog 的方式呈現表單輸入結果，請依下列題意完成作答。



The screenshot shows an Android application window with the title "AlertDialog顯示結果". The status bar at the top indicates the time is 3:50 AM. The app interface has a dark background. It contains a form with the following elements:

- 姓名:** A text input field.
- 性別:** Two radio buttons, "帥哥" (selected) and "美女".
- 興趣:** Three checkboxes, "吃飯", "睡覺", and "上網", all of which are checked.
- 顯示結果:** A button located below the interest section.

## 2. 設計說明：

- (1) 專案中已設計好 main.xml 作為 Activity 的 Layout，其中的 Widget 元件皆已完成 id 與相關設定。
- (2) 開啓 GDD01.java，設定 Button 的 OnClickListener，進行表單欄位資料收集的動作。
- (3) 判斷姓名欄位是否為空值，如為空值則顯示【姓名：未填】，如不為空值則顯示【姓名：xxx】，將姓名代入 xxx 中。
- (4) 判斷性別欄位的選取值，顯示【性別：xx】，將性別代入 xx 中。



- (5) 判斷興趣欄位的選取值，如有選項被勾選時，顯示【興趣：xx,xx....】，將興趣代入 xx 中，如完全沒有選項被勾選時，則顯示【興趣：未選擇】。
- (6) 資料收集完成後，將收集完成的資訊以 AlertDialog 的方式呈現，並設定 AlertDialog 的 Title 為【結果】。
- (7) 在 AlertDialog 中加入一個「離開」按鈕，用以關閉 AlertDialog 回到主畫面。
- (8) AlertDialog 中的結果，每項資訊需分行顯示，如執行結果參考畫面(1)(2)。

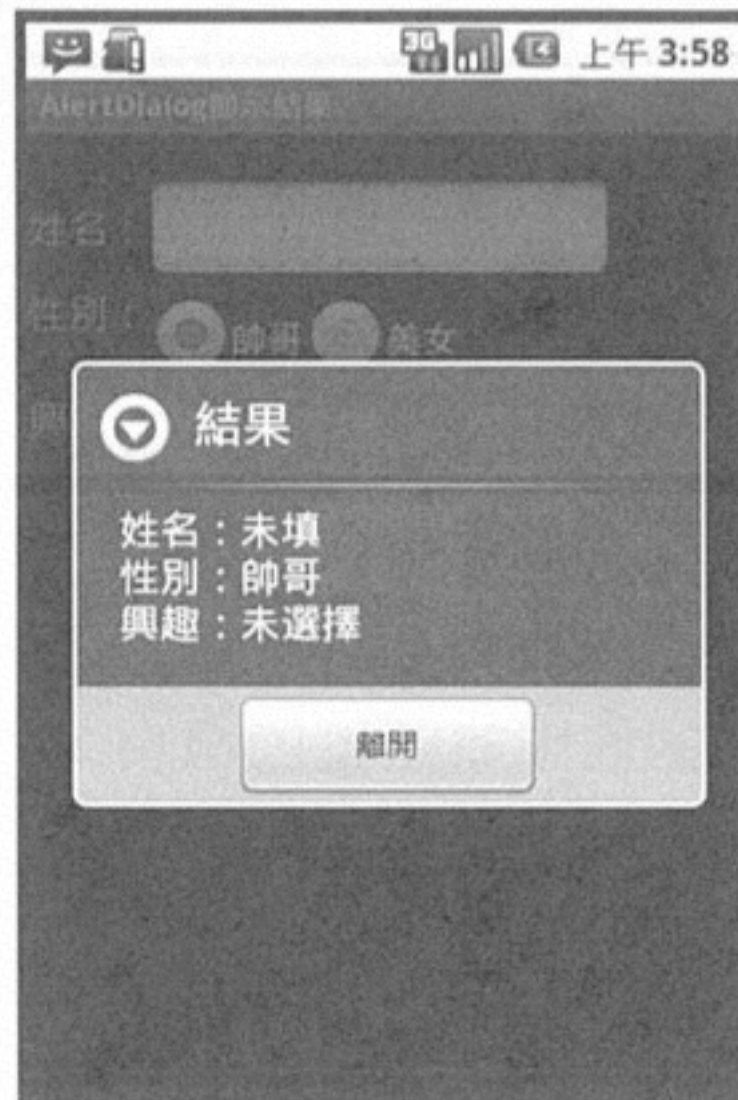
### 3. 執行結果參考畫面：

- (1) 按下【顯示結果】按鈕，此時程式會收集表單內填寫的資訊，以跳出 AlertDialog 的方式呈現表單輸入結果。



- (2) 姓名欄位如為空值則顯示【姓名：未填】。如興趣欄位完全沒有選項被勾選時，則顯示【興趣：未選擇】。





#### 4. 自行測試程式是否達成下列結果：

- (1) 按下「顯示結果」按鈕時，跳出 AlertDialog。
- (2) 當姓名欄位輸入為空值時，AlertDialog 內顯示【姓名：未填】，如不為空值時，則顯示【姓名：xxx】，將姓名代入 xxx 中。
- (3) AlertDialog 內顯示的性別為正確的選項。
- (4) 當興趣選項都未勾選時，AlertDialog 內顯示【興趣：未選擇】，如有勾選時，則顯示【興趣：xx,xx....】，將興趣代入 xx 中。
- (5) AlertDialog 的 Title 設定為【結果】，且有「離開」按鈕以關閉 AlertDialog。
- (6) AlertDialog 中的結果，每項資訊需分行顯示，如執行結果參考畫面(1)(2)。



5. 評分項目：

項	目	配 分	得 分
(1)	按下「顯示結果」按鈕時，有跳出 AlertDialog	2	
(2)	當姓名欄位輸入為空值時，AlertDialog 內顯示【姓名：未填】，如不為空值時，則顯示【姓名：xxx】，將姓名代入 xxx 中	4	
(3)	AlertDialog 內顯示的性別為正確的選項	4	
(4)	當興趣選項都未勾選時，AlertDialog 內顯示【興趣：未選擇】，如有勾選時，則顯示【興趣：xx,xx....】，將所選擇的興趣代入 xx 中	4	
(5)	AlertDialog 的 Title 設定為【結果】，且有「離開」按鈕以關閉 AlertDialog	4	
(6)	AlertDialog 中的結果，每項資訊需分行顯示，如執行結果參考畫面(1)(2)	2	
總	分	20	



# AlertDialog

- 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: AlertDialog
  - Build Target: Android 4
  - Package Name: tw.edu.vnu.alertdialog



# 建立元件的版面配置

- 刪除 “Hello World” TextView
- 將 RelativeLayout 改為 LinearLayout (Vertical)
- 在 LinearLayout 依序編排 3 個  
LinearLayout (Horizontal) 和 Button 元件

- 在第 1 個 LinearLayout (Horizontal) 依序編排 TextView 和 EditText
- 在第 2 個 LinearLayout (Horizontal) 依序編排 TextView 和 RadioGroup，並刪除 RadioGroup 內的一個 RadioButton
- 在第 3 個 LinearLayout (Horizontal) 依序編排 TextView 和 3 個 CheckBox



## AlertDialog

TextView |

TextView ☐ RadioButton ☐ RadioButton

TextView ☐ CheckBox ☐ CheckBox Ch  
ec  
kB  
ny

Button

- 刪除 string.xml 的 hello\_world，修改 app\_name 為 “AlertDialog顯示結果”
- 將第一個 TextView 新增 name 的 string 值中文為 “姓名：”
- 第二個 TextView 新增 sex 的 string 值為 “性別：”
- 第三個 TextView 新增 interest 的 string 值為 “興趣：”



- 將 RadioGroup 的 orientation 設定為 horizontal
- 第一個 RadioButton 新增 male 的 string 值為 “帥哥”
- 第二個 RadioButton 新增 female 的 string 值為 “美女”

- 將第一個 CheckBox 新增 eat 的 string 值中文為“吃飯”
- 第二個 CheckBox 新增 sleep 的 string 值中文為“睡覺”
- 第三個 CheckBox 新增 internet 的 string 值中文為“上網”



- 將 Button 的 `layout_gravity` 設定為 `center`，並新增 `btn` 的 `string` 值中文為“顯示結果”

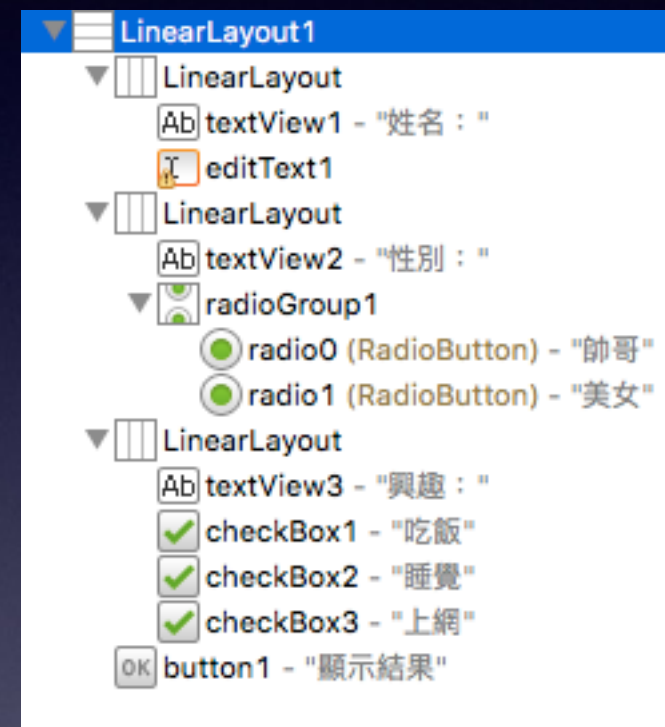
Android 5:10

### AlertDialog顯示結果

姓名：

性別： ☐ 帥哥 ☐ 美女

興趣： ☐ 吃飯 ☐ 睡覺 ☐ 上網



```
<LinearLayout xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:tools="http://schemas.android.com/tools"
    android:id="@+id/LinearLayout1"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    android:orientation="vertical"
    tools:context="${relativePackage}.${activityClass}" >

    <LinearLayout
        android:layout_width="match_parent"
        android:layout_height="wrap_content" >

        <TextView
            android:id="@+id/textView1"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:text="@string/name" />

        <EditText
            android:id="@+id/editText1"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:layout_weight="1"
            android:ems="10"
            android:inputType="textPersonName" >

            <requestFocus />
        </EditText>
    </LinearLayout>

    <LinearLayout
        android:layout_width="match_parent"
        android:layout_height="wrap_content" >
```



```
<TextView
    android:id="@+id/textView2"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:text="@string/sex" />

<RadioGroup
    android:id="@+id/radioGroup1"
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:orientation="horizontal" >

    <RadioButton
        android:id="@+id/radio0"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="@string/male" />

    <RadioButton
        android:id="@+id/radio1"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="@string/female" />
</RadioGroup>
</LinearLayout>

<LinearLayout
    android:layout_width="match_parent"
    android:layout_height="wrap_content" >

    <TextView
        android:id="@+id/textView3"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="@string/interest" />
```

```
<CheckBox
    android:id="@+id/checkBox1"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:text="@string/eat" />

<CheckBox
    android:id="@+id/checkBox2"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:text="@string/sleep" />

<CheckBox
    android:id="@+id/checkBox3"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:text="@string/internet" />
</LinearLayout>

<Button
    android:id="@+id/button1"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:layout_gravity="center"
    android:text="@string/btn" />

</LinearLayout>
```

```
<?xml version="1.0" encoding="utf-8"?>
<resources>

    <string name="app_name">AlertDialog顯示結果</string>

    <string name="name">姓名:</string>

    <string name="sex">性別:</string>
    <string name="interest">興趣:</string>

    <string name="btn">顯示結果</string>

    <string name="male">帥哥</string>

    <string name="female">美女</string>

    <string name="eat">吃飯</string>

    <string name="sleep">睡覺</string>

    <string name="internet">上網</string>

</resources>
```



# 建立 Activity 活動類別

- 宣告成員變數
- onCreate() 方法

# 宣告成員變數

```
package com.example.alertdialog;

import java.util.ArrayList;
import java.util.List;

import android.app.Activity;
import android.app.AlertDialog;
import android.content.DialogInterface;
import android.os.Bundle;
import android.view.View;
import android.view.View.OnClickListener;
import android.widget.Button;
import android.widget.CheckBox;
import android.widget.EditText;
import android.widget.RadioButton;

public class MainActivity extends Activity {

    private EditText nameText;
    private RadioButton rbtn0;
    private RadioButton rbtn1;
    private CheckBox checkBox1;
    private CheckBox checkBox2;
    private CheckBox checkBox3;
    private Button showbtn;
    private AlertDialog.Builder alertDialog;

    @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);

    nameText = (EditText) findViewById(R.id.editText1);
    rbtn0 = (RadioButton) findViewById(R.id.radio0);
    rbtn1 = (RadioButton) findViewById(R.id.radio1);

    checkBox1 = (CheckBox) findViewById(R.id.checkBox1);
    checkBox2 = (CheckBox) findViewById(R.id.checkBox2);
    checkBox3 = (CheckBox) findViewById(R.id.checkBox3);

    alertDialog = new AlertDialog.Builder(this);
    alertDialog.setTitle("結果");
    alertDialog.setOnCancelListener(new DialogInterface.OnCancelListener() {

        @Override
        public void onCancel(DialogInterface dialog) {
            // 關閉 AlertDialog
        }
    });
}
```



```
showbtn = (Button) findViewById(R.id.button1);
showbtn.setOnClickListener(new OnClickListener() {

    @Override
    public void onClick(View v) {
        StringBuilder sb = new StringBuilder();
        sb.append("姓名 : ");

        if (nameText.getText().toString().trim().isEmpty()) {
            sb.append("未填").append("\n");
        } else {
            sb.append(nameText.getText().toString()).append("\n");
        }

        sb.append("性別 : ");

        if (rbtn0.isChecked()) {
            sb.append("帥哥").append("\n");
        } else {
            sb.append("美女").append("\n");
        }

        sb.append("興趣 : ");
        List<String> list = new ArrayList<String>();

        if (checkBox1.isChecked()) {
            list.add("吃飯");
        }
        if (checkBox2.isChecked()) {
            list.add("睡覺");
        }
        if (checkBox3.isChecked()) {
            list.add("上網");
        }
    }
});
```

```
        if (list.size() != 0) {  
            for (int i = 0; i < list.size(); i++) {  
                String s = list.get(i);  
                sb.append(s);  
  
                if (i < list.size() - 1) {  
                    sb.append(", ");  
                }  
            }  
        } else {  
            sb.append("未選擇");  
        }  
        alertDialog.setMessage(sb.toString()).show();  
    }  
});  
}
```

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

