

# Android程式設計

## 計算BMI值

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「計算 BMI 值」應用程式。在輸入性別、身高及體重後，按下「計算」按鈕，在 Activity 中運用 Intent 傳遞資訊並啓動另一個 Activity，並計算 BMI 值，請依下列題意完成作答。



計算BMI值

計算你/妳的BMI值

男性/女性： ☒ 男性 ☐ 女性

身高(m)：

體重(kg)：

計算

## 2. 設計說明：

- (1) 請設計兩個 Activity，程式執行時，首先進入第一個 Activity (GDD01.java)。
- (2) 請於 GDD01.java 定義兩個僅可單選的 RadioButton，以便選擇男性或女性，兩個 EditText 供使用者輸入身高及體重，一個 Button 控制項其 String 值設定為【計算】，其餘 TextView 控制項顯示如參考圖所示。
- (3) res/layout/main.xml 定義第一個 Activity 所有物件，請依參考圖定義產生使用者畫面。
- (4) 於 Activity1 (GDD01.java) 中，以 EditText 定義文字框，資料型態設定為 decimal，可輸入小數點。於第一個 Activity 點選「計算」後，將所輸入的數值資料帶至第二個 Activity (GDD01\_Child.java)。



- (5) 第二個 Activity 接收資料後即開始計算比對。並以兩個 TextView 分別顯示 BMI 值與建議事項。按「回上一頁」按鈕，結束此 Activity，回到上一個 Activity。
- (6) res/layout/mylayout.xml 定義第二個 Activity 所有物件。
- (7) res/values/strings.xml 記錄第一個 Activity 的變數名稱及內容，res/values/advice.xml 記錄第二個 Activity 的計算結果建議事項，strings.xml 與 advice.xml 已定義好程式內所需使用的文字。
- (8) BMI 值 = 體重 (公斤) / ((身高 (公尺) \* 身高 (公尺))。男性標準：BMI 介於 20~25，女性標準：BMI 介於 18~22。
- (9) 若計算的結果小於最小標準值，則第二個 Activity 顯示【您該多吃點】，介於標準值範圍內，顯示【體型很棒喔】，若計算結果大於最大標準值，顯示【您該少吃些，並多多運動】。

### 3. 執行結果參考畫面：

- (1) 於第一個 Activity 中，以 EditText 定義文字框，資料型態設定為 decimal，可輸入小數點。

計算你/妳的BMI值

男性/女性： ☐ 男性 ☒ 女性

身高(m)： 1.58

體重(kg)：

1 2 3 4 5 6 7 8 9 0

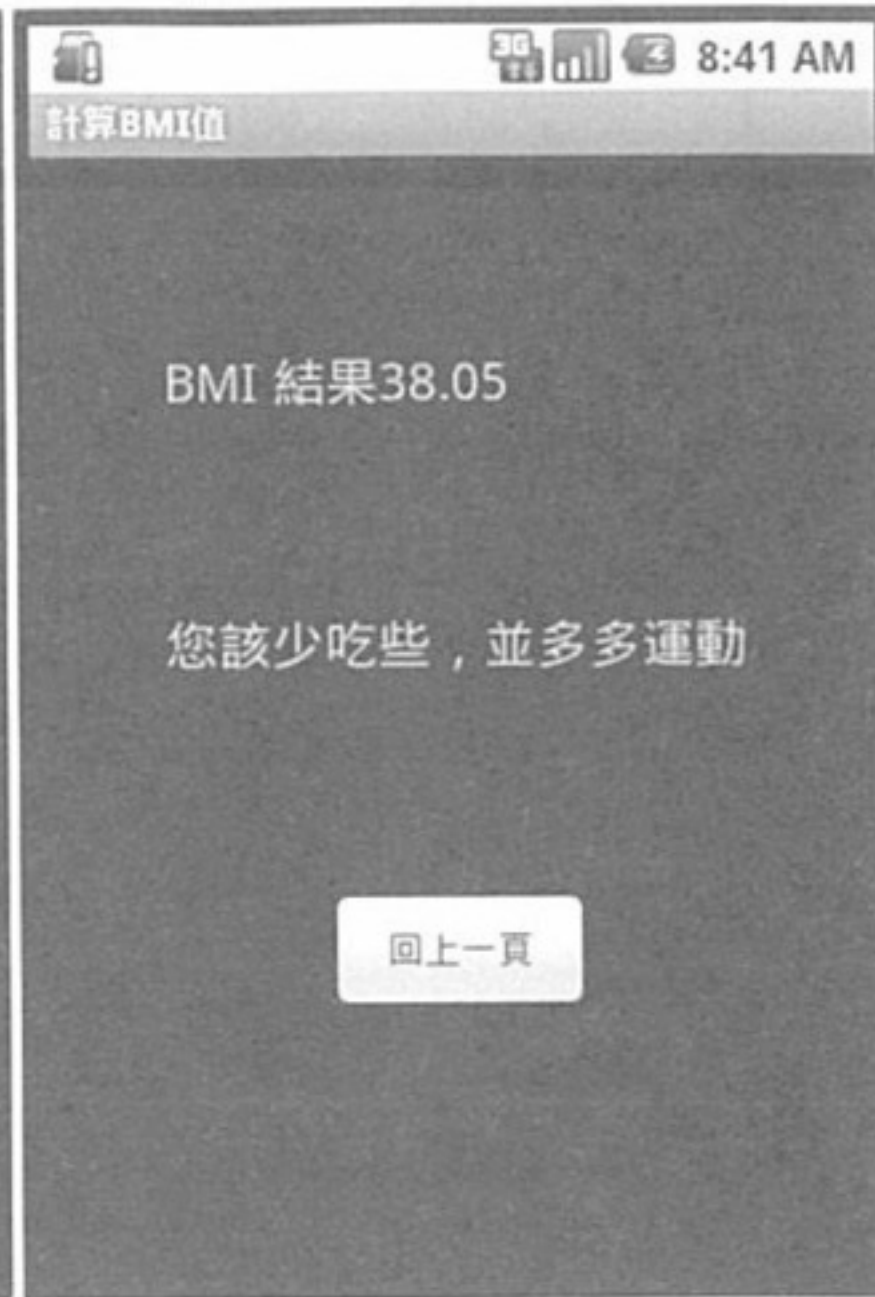
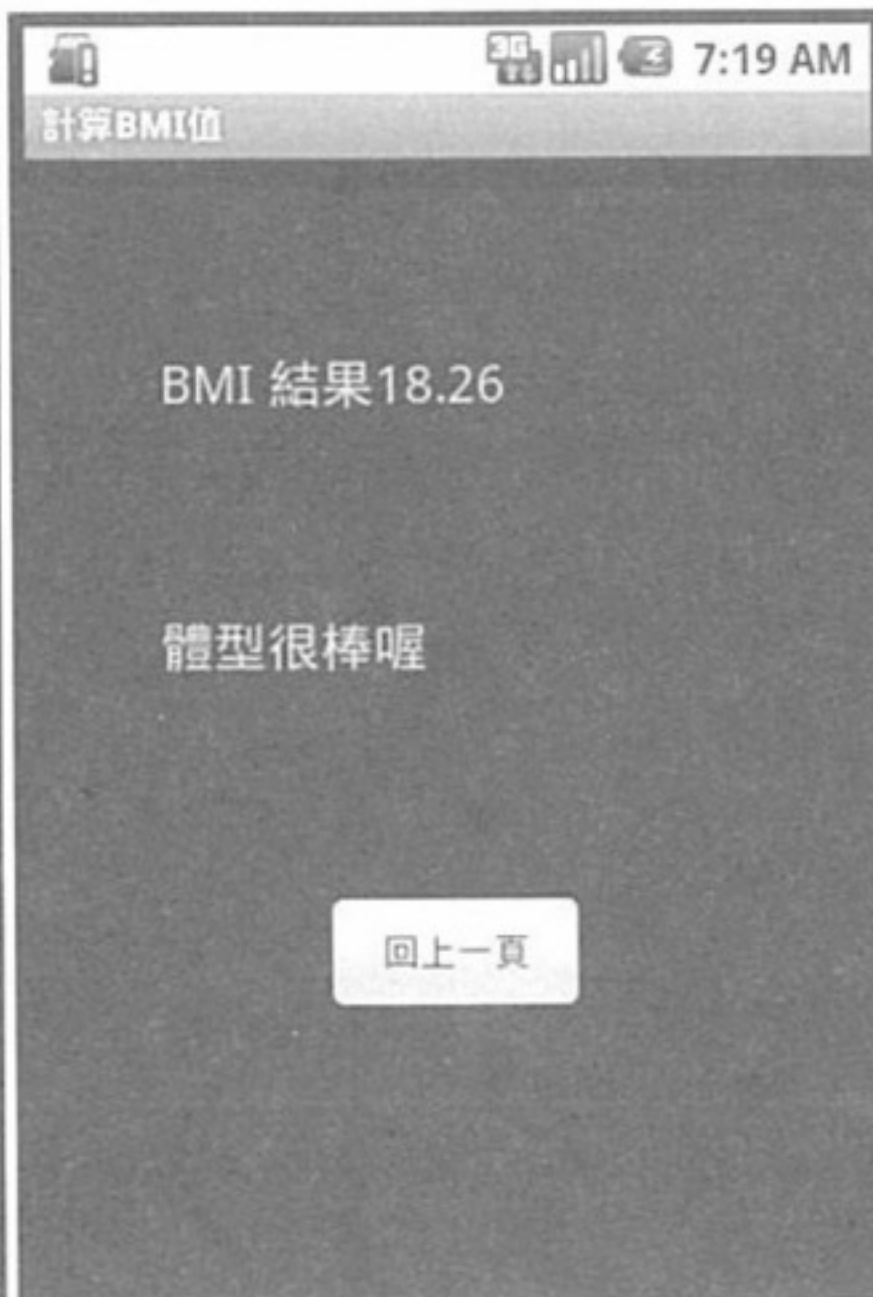
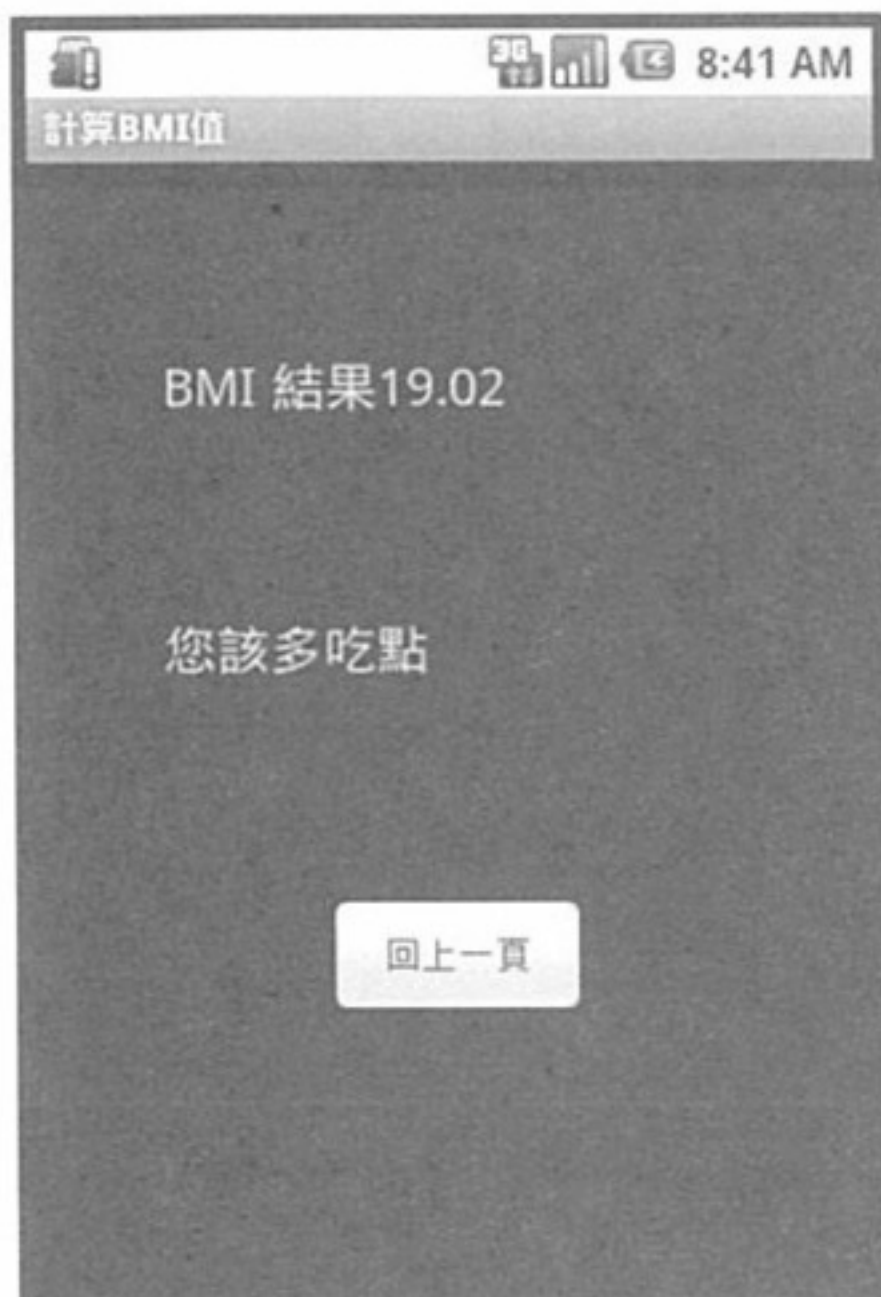
@ # \$ % " \* - / ( )

ALT ! " ' : ; , ? DEL

英符 ?123 \_ . ↵



(2) 若計算的結果小於最小標準值，則第二個 Activity 顯示【您該多吃點】，介於標準值範圍內，顯示【體型很棒喔】，若計算結果大於最大標準值，顯示【您該少吃些，並多多運動】。



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/mylayout.xml`、`res/layout/main.xml` 的定義正確顯示。
- (2) 正確使用 `res/values/strings.xml`、`res/values/advice.xml`。
- (3) 第一個 Activity 中，以 `EditText` 定義文字框，資料型態設定為 `decimal`，可輸入小數點。
- (4) 程式將輸入資訊由 Activity 傳入另一個 Activity，在第二個 Activity 計算結果，顯示 BMI 值與建議事項。
- (5) 點選「回上一頁」按鈕，結束此 Activity，回到上一個 Activity。



## 5. 評分項目：

項	目	配 分	得 分
(1)	畫面元件依 res/layout/mylayout.xml、 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)	正確使用 res/values/strings.xml、 res/values/advice.xml	2	
(3)	第一個 Activity 中，以 EditText 定義文字框，資 料型態設定為 decimal，可輸入小數點	2	
(4)	程式將輸入資訊由 Activity 傳入另一個 Activity，在第二個 Activity 計算結果，顯示正 確 BMI 值與建議事項	8	
(5)	點選「回上一頁」按鈕，結束此 Activity，回到 上一個 Activity	4	
總	分	20	



Activity

# 啟動 Activity

透過呼叫 `startActivity()`、將 `Intent` (描述要啟動的 Activity) 傳給它，您可以啟動另一個 Activity。意圖會指出您要啟動的那個 Activity，或描述您要執行的動作類型 (而讓系統為您選取適當的 Activity，甚至可以是來自不同應用程式的 Activity)。意圖也可以攜帶少量的資料給已啟動的 Activity 使用。

在您自己的應用程式內運作時，通常只要啟動已知的 Activity。建立意圖並明確定義您要啟動的 Activity (使用類別名稱)，可以達成此目的。例如，以下示範 Activity 如何啟動另一個名為 `SignInActivity` 的 Activity：

```
Intent intent = new Intent(this, SignInActivity.class);
startActivity(intent);
```

不過，您的應用程式也希望可以執行其他動作，例如使用您 Activity 中的資料以傳送電子郵件、文字訊息或狀態更新。在此情況下，您的應用程式可能就沒有專屬的 Activity 來執行這類動作。因此，您可以改為運用裝置上其他應用程式提供的 Activity。讓這些 Activity 為您執行所需的動作。這正是意圖寶貴的地方－您可以建立意圖，在其中描述您要執行的動作，然後系統會從另一個應用程式啟動適當的 Activity。如果有多個 Activity 都可以處理意圖，則使用者可以選取要使用哪個 Activity。例如，如果要讓使用者傳送電子郵件訊息，您可以建立下列意圖：

```
Intent intent = new Intent(Intent.ACTION_SEND);
intent.putExtra(Intent.EXTRA_EMAIL, recipientArray);
startActivity(intent);
```

加入意圖的 `EXTRA_EMAIL` 額外值是一個字串陣列，由應該要寄送電子郵件的電子郵件地址所組成。電子郵件應用程式回應此意圖時，它會讀取額外值中提供的字串陣列，並將這些內容放置在編寫電子郵件表單的「收件人」欄位。在此情況下，電子郵件應用程式的 Activity 會啟動，並且會在使用者完成後，繼續您的 Activity。

## 啟動 Activity 以取得結果

有時候，您會想要收到由您啟動 Activity 的結果。如要接收結果，請透過呼叫 `startActivityForResult()` (而非 `startActivity()`) 以啟動 Activity。如要接收後續 Activity 的結果，請實作 `onActivityResult()` 回呼方法。後續 Activity 完成時，會將 `Intent` 中的結果傳回到您的 `onActivityResult()` 方法。

例如，您可能會讓使用者在聯絡人中挑選一位，讓您的 Activity 可以針對該聯絡人的資訊進行一些處理。以下示範如何建立這類意圖，並處理結果：

```
private void pickContact() {
    // Create an intent to "pick" a contact, as defined by the content provider URI
    Intent intent = new Intent(Intent.ACTION_PICK, Contacts.CONTENT_URI);
    startActivityForResult(intent, PICK_CONTACT_REQUEST);
}

@Override
protected void onActivityResult(int requestCode, int resultCode, Intent data) {
    // If the request went well (OK) and the request was PICK_CONTACT_REQUEST
    if (resultCode == Activity.RESULT_OK && requestCode == PICK_CONTACT_REQUEST) {
        // Perform a query to the contact's content provider for the contact's name
        Cursor cursor = getContentResolver().query(data.getData(),
            new String[] {Contacts.DISPLAY_NAME}, null, null, null);
        if (cursor.moveToFirst()) { // True if the cursor is not empty
            int columnIndex = cursor.getColumnIndex(Contacts.DISPLAY_NAME);
            String name = cursor.getString(columnIndex);
            // Do something with the selected contact's name...
        }
    }
}
```

此範例顯示您應該在您的 `onActivityResult()` 方法中使用的基本邏輯，以便處理 Activity 結果。第一個條件會檢查要求是否成功 — 如果成功，則 `resultCode` 會是 `RESULT_OK` —，以及此結果回應的要求是否為已知的要求 — 範例中的 `requestCode` 符合以 `startActivityForResult()` 傳送的第二個參數。程式碼在這裡透過查詢 `Intent` (`data` 參數) 中傳回的資料，來處理 Activity 結果。

其中的過程是，`ContentResolver` 針對內容供應程式執行查詢，所傳回的 `Cursor` 可以讀取查詢到的資料。如需詳細資訊，請參閱[內容供應程式](#)。

如需關於使用意圖的詳細資訊，請參閱[意圖](#)和[意圖篩選器](#)。



## 關閉 Activity

呼叫 Activity 的 `finish()` 方法，可以關閉此 Activity。您也可以呼叫 `finishActivity()`，關閉您之前啟動的個別 Activity。

**注意：**大多數情況，您不應使用這些方法明確地結束 Activity。如同下一節所討論的 Activity 生命週期，Android 系統會為您管理 Activity 的生命週期，所以您不需要結束您自己的 Activity。呼叫這些方法對使用者體驗有負面的影響，只有在您十分確定不希望使用者返回 Activity 的此執行個體時，才加以呼叫。

Intent 意圖

# 意圖和意圖篩選器

`Intent` 是可用來向另一個[應用程式元件](#)要求動作的傳訊物件。雖然意圖有幾種方式可加速元件間的通訊，但共有三種基本使用案例：

- 如何啟動 `Activity`：

`Activity` 代表應用程式中的單一畫面。您可以將 `Intent` 傳送至 `startActivity()` 來啟動 `Activity` 的新執行個體。`Intent` 可描述要啟動的 `Activity` 並攜帶任何必要資料。

如果您想要在 `Activity` 完成時收到結果，請呼叫 `startActivityForResult()`。`Activity` 的 `onActivityResult()` 回呼中的個別 `Intent` 物件，就是 `Activity` 收到的結果。如需詳細資訊，請參閱 [Activity 指南](#)。

- 如何啟動服務：

`Service` 是可以在背景中執行操作的元件，但沒有使用者介面。您可以將 `Intent` 傳送至 `startService()` 來啟動服務以執行一次性操作 (例如下載檔案)。`Intent` 可描述要啟動的服務並攜帶任何必要資料。

如果服務是採用主從介面設計，您可以將 `Intent` 傳送至 `bindService()` 來繫結至另一個元件的服務。如需詳細資訊，請參閱[服務指南](#)。

- 如何傳送廣播：

廣播是指任何應用程式都可接收的訊息。系統會傳送各種系統事件廣播，例如系統開機或裝置開始充電。您可以將 `Intent` 傳送至 `sendBroadcast()`、`sendOrderedBroadcast(Intent, String)` 或 `sendStickyBroadcast()`，以向其他應用程式傳送廣播。



# 意圖類型

意圖類型分為兩種：

- **明確意圖**：可依名稱 (完整類別名稱) 指定要啟動的元件。一般情況下，您會使用明確意圖啟動您應用程式中的元件，這是因為您知道 Activity 的類別名稱或您想要啟動的服務。例如，為回應使用者動作而啟動新的 Activity，或啟動服務以在背景下載檔案。
- **隱含意圖**：不會指定特定元件，而會宣告要執行的一般動作，讓另一個應用程式的元件來處理它。例如，如果您想要向使用者顯示地圖上的某個位置，可以使用隱含意圖，要求另一個支援應用程式在地圖上顯示指定的位置。

當您建立明確意圖以啟動 Activity 或服務時，系統會立即啟動 **Intent** 物件中指定的應用程式元件。

當您建立隱含意圖時，Android 系統會比較意圖內容和裝置上其他應用程式的**宣示說明檔案**中宣告的「意圖篩選器」，以找出要啟動的適當元件。如果意圖和意圖篩選器相符，系統會啟動該元件，並將 **Intent** 物件傳送給它。如果有多個意圖篩選器符合意圖，系統會顯示對話方塊，供使用者挑選要使用的應用程式。

意圖篩選器是應用程式宣示說明檔案中的運算式，可指定元件要接收的意圖類型。例如，藉由宣告 Activity 的意圖篩選器，可讓其他應用程式使用特定意圖類型直接啟動您的 Activity。同樣地，如果您「不」為 Activity 宣告任何意圖篩選器，就只能以明確意圖啟動它。

**注意：**為了確保您的應用程式安全，請一律使用明確意圖啟動 **Service**，並且不要宣告服務的意圖篩選器。使用隱含意圖啟動服務會危害安全性，原因在於您無法確定哪個服務會回應意圖，而且使用者無法得知系統會啟動哪項服務。從 Android 5.0 (API 級別 21) 開始，如果您使用隱含意圖呼叫 **bindService()**，系統都會擲回例外狀況。

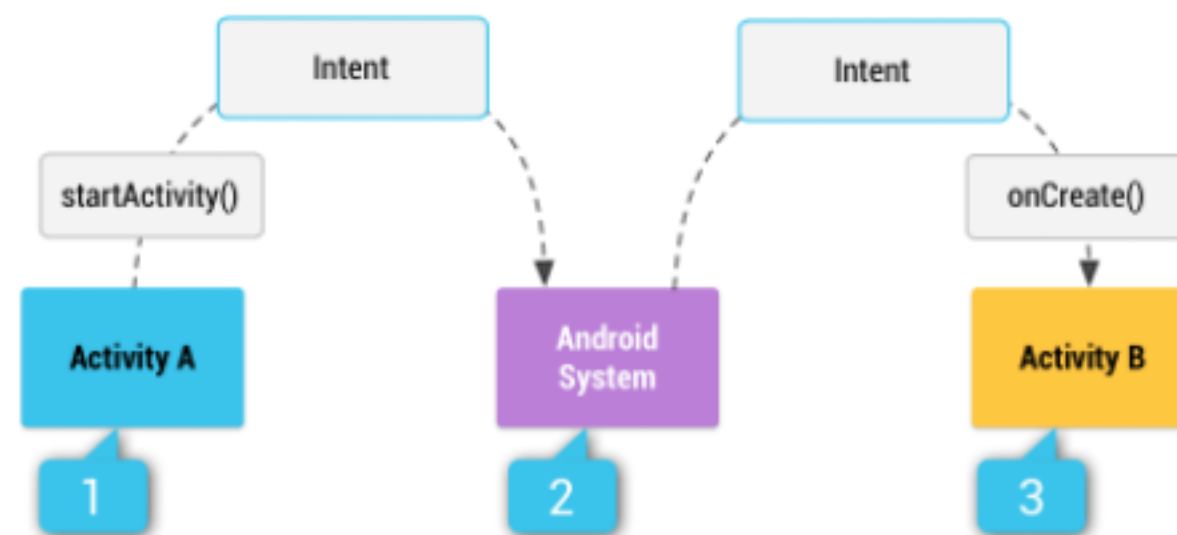


圖 1.說明如何透過系統傳送隱含意圖以啟動另一個 Activity：[1] Activity A 會建立含有動作描述的**Intent**並傳送至 **startActivity()**。[2] Android 系統會搜尋所有應用程式，以找出符合該意圖的意圖篩選器。找到相符項目時，[3] 系統會呼叫其 **onCreate()** 方法，並將 **Intent** 傳送給它來啟動相符的 Activity (Activity B)。

## 明確意圖範例

您用來啟動特定應用程式元件的就是明確意圖，例如應用程式中的特定 Activity 或服務。如要建立明確意圖，請定義 `Intent` 物件的元件名稱 — 其他意圖屬性均為選用性質。

例如，您在應用程式中建置稱為 `DownloadService` 的服務，其設計為從網頁下載檔案，您可以使用下列程式碼來啟動該服務：

```
// Executed in an Activity, so 'this' is the Context
// The fileUrl is a string URL, such as "http://www.example.com/image.png"
Intent downloadIntent = new Intent(this, DownloadService.class);
downloadIntent.setData(Uri.parse(fileUrl));
startService(downloadIntent);
```

`Intent(Context, Class)` 建構函式提供應用程式 `Context` 與元件 (`Class` 物件)。因此，這個意圖會明確啟動應用程式中的 `DownloadService` 類別。

如要進一步瞭解如何建置及啟動服務，請參閱[服務指南](#)。

# BMI

- 開啟和執行 Android 專案
- 建立 activity\_main 元件的版面配置
- 建立 activity\_child 元件的版面配置
- 建立 MainActivity 活動類別
- 建立 ChildActivity 活動類別
- AndroidManifest 設定



# 開啟和執行 Android 專案

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

# 建立 activity\_main 元件的版面配置

- 刪除 “Hello World” TextView
- 將 RelativeLayout 改為 AbsoluteLayout
- 在 AbsoluteLayout 依序垂直編排  
TextViwe (Large) 、TextViwe (Medium) 、  
TextViwe (Medium) 、TextViwe (Medium) 、  
RadioGroup 、RadioButton 、RadioButton 、  
EditText (numberDecimal) 、  
EditText (numberDecimal) 和 Button 元件



## 計算BMI值

計算你/妳的BMI值

男性/女性： ☐ 男性 ☐ 女性

身高(m)：

體重(kg)：

計算



- 刪除 string.xml 的 hello\_world，修改 app\_name 為 “計算BMI值”
- 將第一個 TextView 新增 title 的 string 值中文為 “計算你/妳的BMI值”，並將 layout\_x 和 layout\_y 屬性設定為 45dp
- 將第二個 TextView 新增 sex 的 string 值中文為 “男性/女性：”，將 layout\_x 屬性設定為 15dp，layout\_y 屬性設定為 120dp

- 將第三個 TextView 新增 height 的 string 值中文為“身高(m):”，將 layout\_x 屬性設定為 15dp，layout\_y 屬性設定為 170dp
- 將第四個 TextView 新增 weight 的 string 值中文為“體重(kg):”，將 layout\_x 屬性設定為 15dp，layout\_y 屬性設定為 220dp
- 將 RadioGroup 的 id 設定為 sex，將 layout\_x 屬性設定為 120dp，layout\_y 屬性設定為 110dp，並將 orientation 屬性設定為 horizontal

- 將第一個 RadioButton 的 id 設定為 male，新增 male 的 string 值中文為“男性”，將 checked 屬性設定為 false
- 將第二個 RadioButton 的 id 設定為 female，新增 female 的 string 值中文為“女性”
- 將第一個 EditText 的 id 設定為 height，將 layout\_width 屬性設定為 145dp，並將 layout\_x 屬性設定為 100dp，layout\_y 屬性設定為 160dp



- 將第二個 EditText 的 id 設定為 weight，將 layout\_width 屬性設定為 145dp，並將 layout\_x 屬性設定為 100dp，layout\_y 屬性設定為 210dp
- 將 Button 的 id 設定為 submit，新增 submit 的 string 值中文為“計算”，將 layout\_width 屬性設定為 70dp，並將 layout\_x 屬性設定為 135dp，layout\_y 屬性設定為 260dp

```
1 <AbsoluteLayout 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
6   <TextView
7       android:id="@+id/respond"
8       android:layout_width="wrap_content"
9       android:layout_height="wrap_content"
10      android:layout_x="45dp"
11      android:layout_y="45dp"
12      android:text="@string/title"
13      android:textAppearance="?android:attr/textAppearanceLarge" />
14
15   <TextView
16       android:id="@+id/msg"
17       android:layout_width="wrap_content"
18       android:layout_height="wrap_content"
19       android:layout_x="15dp"
20       android:layout_y="120dp"
21       android:text="@string/sex"
22       android:textAppearance="?android:attr/textAppearanceMedium" />
23
24   <TextView
25       android:id="@+id/textView3"
26       android:layout_width="wrap_content"
27       android:layout_height="wrap_content"
28       android:layout_x="15dp"
29       android:layout_y="170dp"
30       android:text="@string/height"
31       android:textAppearance="?android:attr/textAppearanceMedium" />
32
```

```
33 <TextView
34     android:id="@+id/textView4"
35     android:layout_width="wrap_content"
36     android:layout_height="wrap_content"
37     android:layout_x="15dp"
38     android:layout_y="220dp"
39     android:text="@string/weight"
40     android:textAppearance="?android:attr/textAppearanceMedium" />
41
42 <RadioGroup
43     android:id="@+id/sex"
44     android:layout_width="wrap_content"
45     android:layout_height="wrap_content"
46     android:layout_x="120dp"
47     android:layout_y="110dp"
48     android:orientation="horizontal" >
49
50     <RadioButton
51         android:id="@+id/male"
52         android:layout_width="wrap_content"
53         android:layout_height="wrap_content"
54         android:checked="false"
55         android:text="@string/male" />
56
57     <RadioButton
58         android:id="@+id/female"
59         android:layout_width="wrap_content"
60         android:layout_height="wrap_content"
61         android:text="@string/female" />
62
63 </RadioGroup>
64
```



```
65     <EditText
66         android:id="@+id/height"
67         android:layout_width="145dp"
68         android:layout_height="wrap_content"
69         android:layout_x="100dp"
70         android:layout_y="160dp"
71         android:inputType="numberDecimal" />
72
73     <EditText
74         android:id="@+id/weight"
75         android:layout_width="145dp"
76         android:layout_height="wrap_content"
77         android:layout_x="100dp"
78         android:layout_y="210dp"
79         android:ems="10"
80         android:inputType="numberDecimal" />
81
82     <Button
83         android:id="@+id/submit"
84         android:layout_width="70dp"
85         android:layout_height="wrap_content"
86         android:layout_x="135dp"
87         android:layout_y="260dp"
88         android:text="@string/submit" />
89
90
91 </AbsoluteLayout>
```

```
1 <?xml version="1.0" encoding="utf-8"?>
2 <resources>
3
4     <string name="app_name">計算BMI值</string>
5
6     <string name="title">計算你/妳的BMI值</string>
7
8     <string name="sex">男性/女性：</string>
9
10    <string name="height">身高 (m)：</string>
11
12    <string name="weight">體重 (kg)：</string>
13
14    <string name="male">男性</string>
15
16    <string name="female">女性</string>
17
18    <string name="submit">計算</string>
19
20    <string name="err_msg_sex">請選擇性別！</string>
21
22    <string name="err_msg_height">請輸入身高！</string>
23
24    <string name="err_msg_weight">請輸入體重！</string>
25
26 </resources>
```

# 建立 activity\_child 元件的版面配置

- 將 RelativeLayout 改為 AbsoluteLayout
- 在 AbsoluteLayout 依序垂直編排  
TextViwe (Medium) 、TextViwe (Medium) 、  
TextViwe (Medium) 和 Button 元件





計算BMI值

Medium Text

Medium Text

回上一頁



- 將第一個 TextView 的 id 設定為 respond，將 layout\_x 屬性設定為 50dp，layout\_y 屬性設定為 72dp
- 將第二個 TextView 的 id 設定為 msg，將 layout\_x 屬性設定為 50dp，layout\_y 屬性設定為 172dp
- 將 Button 的 id 設定為 back，新增 back 的 string 值中文為“回上一頁”，將 layout\_width 屬性設定為 100dp，並將 layout\_x 屬性設定為 110dp，layout\_y 屬性設定為 280dp

```
1 <?xml version="1.0" encoding="utf-8"?>
2 <AbsoluteLayout xmlns:android="http://schemas.android.com/apk/res/android"
3     android:id="@+id/AbsoluteLayout1"
4     android:layout_width="match_parent"
5     android:layout_height="match_parent" >
6
7     <TextView
8         android:id="@+id/respond"
9         android:layout_width="wrap_content"
10        android:layout_height="wrap_content"
11        android:layout_x="50dp"
12        android:layout_y="72dp"
13        android:text="Medium Text"
14        android:textAppearance="?android:attr/textAppearanceMedium" />
15
16    <TextView
17        android:id="@+id/msg"
18        android:layout_width="wrap_content"
19        android:layout_height="wrap_content"
20        android:layout_x="50dp"
21        android:layout_y="172dp"
22        android:text="Medium Text"
23        android:textAppearance="?android:attr/textAppearanceMedium" />
24
25    <Button
26        android:id="@+id/back"
27        android:layout_width="100dp"
28        android:layout_height="wrap_content"
29        android:layout_x="110dp"
30        android:layout_y="280dp"
31        android:text="@string/back" />
32
33 </AbsoluteLayout>
```



```
1 <?xml version="1.0" encoding="utf-8"?>
2 <resources>
3
4     <string name="back">回上一頁</string>
5
6     <string name="result">BMI 結果</string>
7
8     <string name="bmi_heavy">您該少吃些，並多多運動</string>
9
10    <string name="bmi_light">您該多吃點</string>
11
12    <string name="bmi_average">體型很棒喔</string>
13
14 </resources>
```

# 建立 MainActivity 活動類別

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

# 宣告成員變數

```
1 package tw.edu.vnu.bmi;
2
3 import android.app.Activity;
4 import android.content.Intent;
5 import android.os.Bundle;
6 import android.view.View;
7 import android.view.View.OnClickListener;
8 import android.widget.Button;
9 import android.widget.EditText;
10 import android.widget.RadioButton;
11 import android.widget.Toast;
12
13 public class MainActivity extends Activity {
14
15     private RadioButton male;
16     private RadioButton female;
17     private EditText height;
18     private EditText weight;
19     private Button submit;
20
21     @Override
22     protected void onCreate(Bundle savedInstanceState) {
23         super.onCreate(savedInstanceState);
24         setContentView(R.layout.activity_main);
25     }
26 }
```

# onCreate() 方法

```
21 @Override
22 protected void onCreate(Bundle savedInstanceState) {
23     super.onCreate(savedInstanceState);
24     setContentView(R.layout.activity_main);
25
26     male = (RadioButton) findViewById(R.id.male);
27     female = (RadioButton) findViewById(R.id.female);
28     height = (EditText) findViewById(R.id.height);
29     weight = (EditText) findViewById(R.id.weight);
30     submit = (Button) findViewById(R.id.submit);
31
32     submit.setOnClickListener(new OnClickListener() {
33
34         @Override
35         public void onClick(View arg0) {
36             String sex = "";
37
38             if (male.isChecked()) {
39                 sex = "m";
40             } else if (female.isChecked()) {
41                 sex = "f";
42             } else { // 未選擇性別錯誤處理
43                 Toast.makeText(MainActivity.this, getResources().getString(R.string.err_msg_sex), Toast.LENGTH_SHORT).show();
44                 return;
45             }
46         }
47     });
48 }
```



# Button 事件處理方法

```
21 @Override
22 protected void onCreate(Bundle savedInstanceState) {
23     super.onCreate(savedInstanceState);
24     setContentView(R.layout.activity_main);
25
26     male = (RadioButton) findViewById(R.id.male);
27     female = (RadioButton) findViewById(R.id.female);
28     height = (EditText) findViewById(R.id.height);
29     weight = (EditText) findViewById(R.id.weight);
30     submit = (Button) findViewById(R.id.submit);
31
32     submit.setOnClickListener(new OnClickListener() {
33
34         @Override
35         public void onClick(View arg0) {
36             String sex = "";
37
38             if (male.isChecked()) {
39                 sex = "m";
40             } else if (female.isChecked()) {
41                 sex = "f";
42             } else { // 未選擇性別錯誤處理
43                 Toast.makeText(MainActivity.this, getResources().getString(R.string.err_msg_sex), Toast.LENGTH_SHORT).show();
44                 return;
45             }
46         }
47     });
48 }
```

```
47 // 檢查畫面欄位有無輸入值
48 if (height.getText().toString().isEmpty()) {
49     Toast.makeText(MainActivity.this, getResources().getString(R.string.err_msg_height), Toast.LENGTH_SHORT).show();
50     return;
51 }
52 if (weight.getText().toString().isEmpty()) {
53     Toast.makeText(MainActivity.this, getResources().getString(R.string.err_msg_weight), Toast.LENGTH_SHORT).show();
54     return;
55 }
56
57 double h = Double.parseDouble(height.getText().toString());
58 double w = Double.parseDouble(weight.getText().toString());
59
60 Bundle bundle = new Bundle();
61 bundle.putString("sex", sex);
62 bundle.putDouble("height", h);
63 bundle.putDouble("weight", w);
64
65 Intent intent = new Intent(MainActivity.this, ChildActivity.class);
66 intent.putExtras(bundle);
67
68 startActivityForResult(intent, 0);
69 }
70 });
71 }
```

# onActivityResult 方法

```
73 @Override
74 protected void onActivityResult(int requestCode, int resultCode, Intent data) {
75     switch (resultCode) {
76         case RESULT_OK:
77             Bundle bundle = data.getExtras();
78
79             String sex = bundle.getString("Sex");
80             double h = bundle.getDouble("height");
81             double w = bundle.getDouble("weight");
82
83             height.setText(String.valueOf(h));
84             weight.setText(String.valueOf(w));
85
86             if (sex.equals("m")) {
87                 male.setChecked(true);
88             } else {
89                 female.setChecked(true);
90             }
91
92             break;
93         default:
94             break;
95     }
96 }
97
98 }
```

# 建立 ChildActivity 活動類別

- 宣告成員變數
- onCreate() 方法
- Button 事件處理方法
- format() 方法
- getBMI() 方法
- getAdvice() 方法



# 宣告成員變數

```
1 package tw.edu.vnu.bmi;
2
3+ import android.app.Activity;
10
11 public class ChildActivity extends Activity {
12
13     private Intent intent;
14     private TextView respond;
15     private TextView msg;
16     private Button back;
17
18 @Override
19     protected void onCreate(Bundle savedInstanceState) {
20         super.onCreate(savedInstanceState);
21         setContentView(R.layout.activity_child);
```

# onCreate() 方法

```
18  @Override
19  protected void onCreate(Bundle savedInstanceState) {
20      super.onCreate(savedInstanceState);
21      setContentView(R.layout.activity_child);
22
23      respond = (TextView) findViewById(R.id.respond);
24      msg = (TextView) findViewById(R.id.msg);
25      back = (Button) findViewById(R.id.back);
26
27      intent = getIntent();
28      Bundle bundle = intent.getExtras();
29      String sex = bundle.getString("sex");
30      double height = bundle.getDouble("height");
31      double weight = bundle.getDouble("weight");
32
33      respond.setText(getBMI(height, weight));
34      msg.setText(getAdvice(sex, height, weight));
35
36      back.setOnClickListener(new OnClickListener() {
37
38          @Override
39          public void onClick(View arg0) {
40              setResult(RESULT_OK, intent);
41              finish();
42          }
43      });
44  }
```

# Button 事件處理方法

```
18      @Override
19      protected void onCreate(Bundle savedInstanceState) {
20          super.onCreate(savedInstanceState);
21          setContentView(R.layout.activity_child);
22          .....
23          respond = (TextView) findViewById(R.id.respond);
24          msg = (TextView) findViewById(R.id.msg);
25          back = (Button) findViewById(R.id.back);
26          .....
27          intent = getIntent();
28          Bundle bundle = intent.getExtras();
29          String sex = bundle.getString("sex");
30          double height = bundle.getDouble("height");
31          double weight = bundle.getDouble("weight");
32          .....
33          respond.setText(getBMI(height, weight));
34          msg.setText(getAdvice(sex, height, weight));
35          .....
36          back.setOnClickListener(new OnClickListener() {
37              .....
38              @Override
39              public void onClick(View arg0) {
40                  setResult(RESULT_OK, intent);
41                  finish();
42              }
43          });
44      }
```

# format() 方法

```
46 private String format(double num) {  
47     return String.format("%.2f", num);  
48 }  
49  
50 private String getBMI(double height, double weight) {  
51     return getResources().getString(R.string.result) + format(weight / (height * height));  
52 }  
53  
54 private String getAdvice(String sex, double height, double weight) {  
55     double max = sex.equals("m") ? 25 : 22;  
56     double min = sex.equals("m") ? 22 : 18;  
57     double bmi = weight / (height * height);  
58  
59     if (bmi > max) {  
60         return getResources().getString(R.string.bmi_heavy);  
61     } else if (bmi < min) {  
62         return getResources().getString(R.string.bmi_light);  
63     } else {  
64         return getResources().getString(R.string.bmi_average);  
65     }  
66 }  
67  
68 }
```



# getBMI() 方法

```
46 private String format(double num) {  
47     return String.format("%.2f", num);  
48 }  
49  
50 private String getBMI(double height, double weight) {  
51     return getResources().getString(R.string.result) + format(weight / (height * height));  
52 }  
53  
54 private String getAdvice(String sex, double height, double weight) {  
55     double max = sex.equals("m") ? 25 : 22;  
56     double min = sex.equals("m") ? 22 : 18;  
57     double bmi = weight / (height * height);  
58  
59     if (bmi > max) {  
60         return getResources().getString(R.string.bmi_heavy);  
61     } else if (bmi < min) {  
62         return getResources().getString(R.string.bmi_light);  
63     } else {  
64         return getResources().getString(R.string.bmi_average);  
65     }  
66 }  
67  
68 }
```

# getAdvice() 方法

```
46 private String format(double num) {  
47     return String.format("%.2f", num);  
48 }  
49  
50 private String getBMI(double height, double weight) {  
51     return getResources().getString(R.string.result) + format(weight / (height * height));  
52 }  
53  
54 private String getAdvice(String sex, double height, double weight) {  
55     double max = sex.equals("m") ? 25 : 22;  
56     double min = sex.equals("m") ? 22 : 18;  
57     double bmi = weight / (height * height);  
58  
59     if (bmi > max) {  
60         return getResources().getString(R.string.bmi_heavy);  
61     } else if (bmi < min) {  
62         return getResources().getString(R.string.bmi_light);  
63     } else {  
64         return getResources().getString(R.string.bmi_average);  
65     }  
66 }  
67  
68 }
```

# AndroidManifest 設定

- 在 Application Nodes 新增 Activity

Java - BMI/AndroidManifest.xml - Eclipse

File Edit Refactor Source Navigate Search Project Run Window Help

Quick Access Java DDMS

Package Explorer

- appcompat\_v7
- BMI
  - src
    - tw.edu.vnu.bmi
      - ChildActivity.java
      - MainActivity.java
    - gen [Generated Java Files]
    - Android 4.4.2
    - Android Private Libraries
    - Android Dependencies
    - assets
    - bin
    - libs
    - AndroidManifest.xml
    - ic\_launcher-web.png
    - lint.xml
    - proguard-project.txt
    - project.properties

\*BMI Manifest

### Android Manifest

Manifest General Attributes

Defines general information about the AndroidManifest.xml

Package: tw.edu.vnu.bmi [Browse]

Version code: 1

Version name: 1.0 [Browse]

Shared user id: [Browse]

Shared user label: [Browse]

Install location: [Browse]

Manifest Extras

Uses Sdk

Add... Remove... Up Down

Manifest Application Permissions Instrumentation AndroidManifest.xml

Problems Javadoc Declaration Console

Android

AndroidManifest.xml - BMI

171M of 495M Android SDK Content Loader



Java - BMI/AndroidManifest.xml - Eclipse

File Edit Refactor Navigate Search Project Run Window Help

Quick Access

Java DDMS

\*BMI Manifest

Label	@string/app_name	Browse...	Allow clear user data	
Icon	@drawable/ic_launcher	Browse...	Test only	
Logo		Browse...	Backup agent	Browse...
Description		Browse...	Allow backup	true
Permission			Kill after restore	
Process		Browse...	Restore needs application	
Task affinity		Browse...	Restore any version	
Allow task reparenting			Never encrypt	
Has code			Large heap	
Persistent			UI options	Select...
Required for all users			Supports rtl	
Enabled			Restricted account type	Browse...
Debuggable			Required account type	Browse...
VM safe mode				

Application Nodes

> .MainActivity

Add...

Remove...

Up

Down

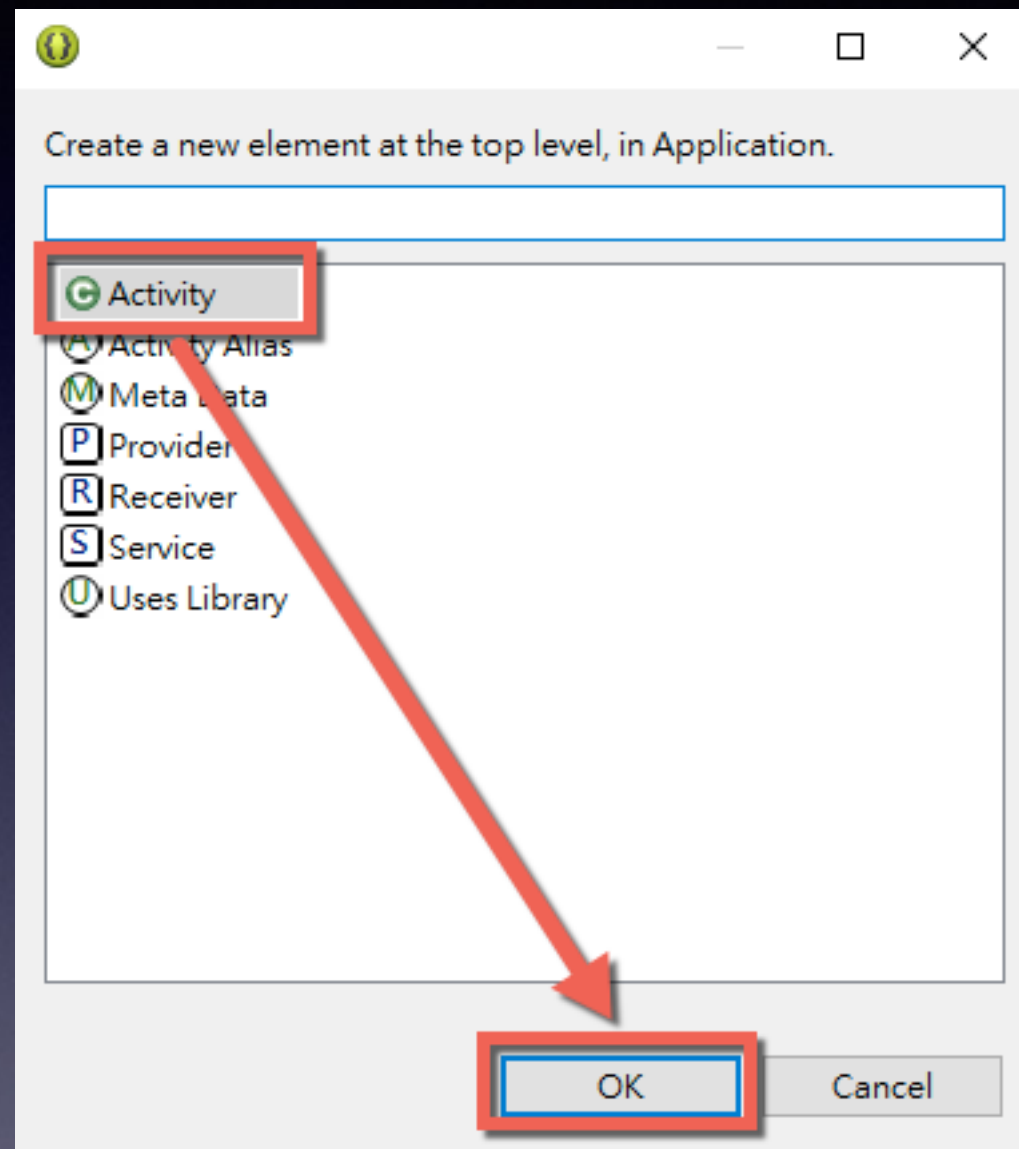
Attributes for .MainActivity

The tag declares an {@link android.app.Activity} class that is available as part of the package's application components, implementing a part of the application's user interface.

Name*	.MainActivity	Browse...
Theme		Browse...
Label	@string/app_name	Browse...

Manifest Application Permissions Instrumentation AndroidManifest.xml

95M of 495M Android SDK Content Loader



Java - BMI/AndroidManifest.xml - Eclipse

File Edit Refactor Navigate Search Project Run Window Help

Quick Access

Java DDMS

\*BMI Manifest

Label	@string/app_name	Browse...	Allow clear user data	
Icon	@drawable/ic_launcher	Browse...	Test only	
Logo		Browse...	Backup agent	Browse...
Description		Browse...	Allow backup	true
Permission			Kill after restore	Console (Android)
Process		Browse...	Restore needs application	
Task affinity		Browse...	Restore any version	
Allow task reparenting			Never encrypt	
Has code			Large heap	
Persistent			UI options	Select...
Required for all users			Supports rtl	
Enabled			Restricted account type	Browse...
Debuggable			Required account type	Browse...
VM safe mode				

Application Nodes

Main Activity

Activity

Add...

Remove...

Up

Down

Attributes for Activity

The tag declares an {@link android.app.Activity} class that is available as part of the package's application components, implementing a part of the application's user interface.

Name\*

Theme

Label

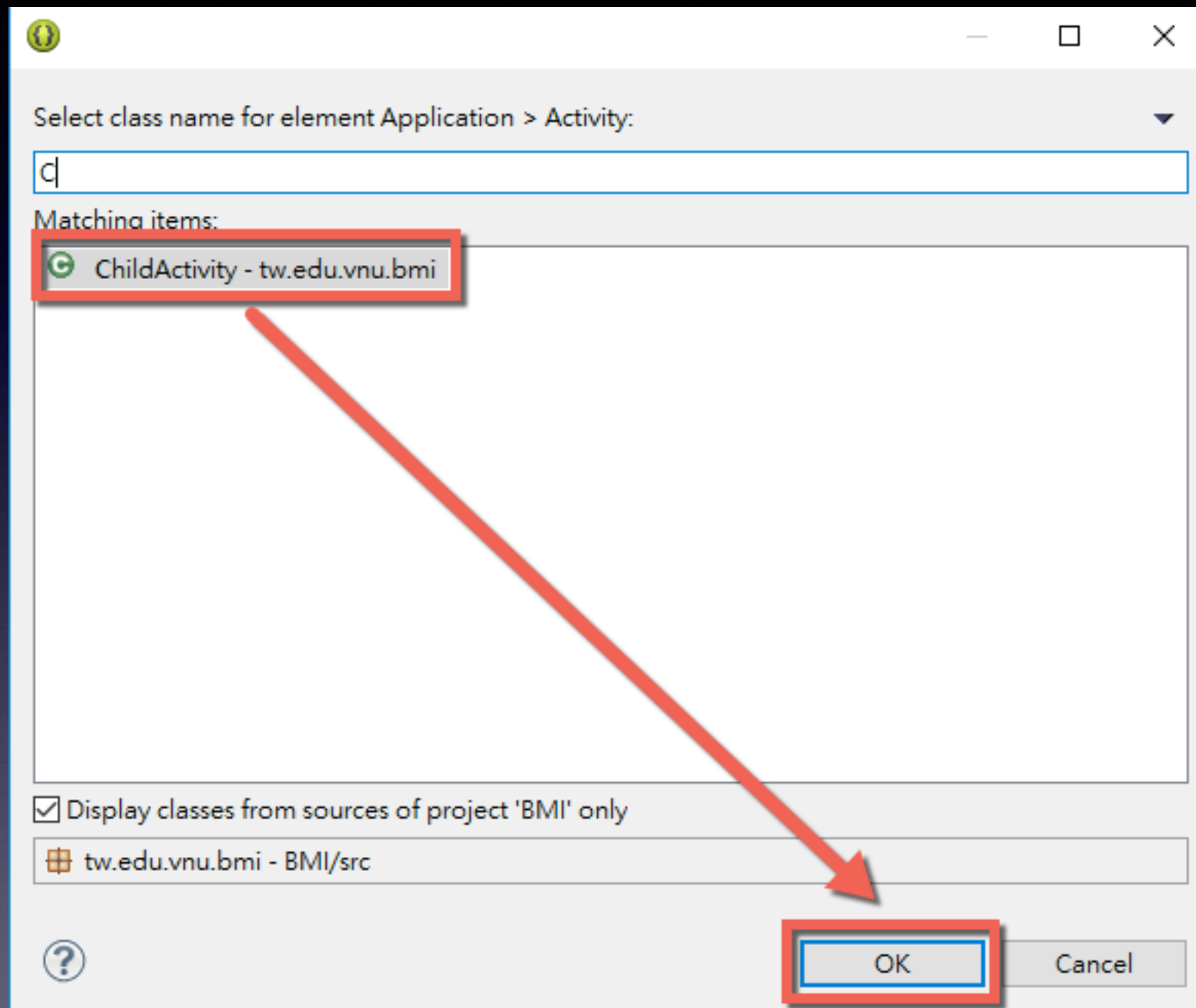
Browse...

Browse...

Browse...

Manifest Application Permissions Instrumentation AndroidManifest.xml

81M of 495M Android SDK Content Loader





```
1 <?xml version="1.0" encoding="utf-8"?>
2 <manifest xmlns:android="http://schemas.android.com/apk/res/android"
3     package="tw.edu.vnu.bmi"
4     android:versionCode="1"
5     android:versionName="1.0" >
6
7     <uses-sdk
8         android:minSdkVersion="9"
9         android:targetSdkVersion="21" />
10
11 <application
12     android:allowBackup="true"
13     android:icon="@drawable/ic_launcher"
14     android:label="@string/app_name"
15     android:theme="@style/AppTheme" >
16     <activity
17         android:name=".MainActivity"
18         android:label="@string/app_name" >
19         <intent-filter>
20             <action android:name="android.intent.action.MAIN" />
21
22             <category android:name="android.intent.category.LAUNCHER" />
23         </intent-filter>
24     </activity>
25     <activity android:name="ChildActivity"></activity>
26 </application>
27
28 </manifest>
```

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

