

# 國立中壢高商 103 學年度 精進優質計畫

桃園縣資訊教師研習講義

## 用 Visual Basic 開發 Android App 專題研習

資深電腦圖書作家 陳會安老師

103/11/25

# 用 Visual Basic 開發 Android App 專題研習

## 壹、活動主旨

- 一、Basic4android (B4A) 是一套整合開發環境 IDE，可以讓 Visual Basic 語言的開發者直接使用 BASIC 語言來開發 Android App，B4A 能夠自動將開發者撰寫的 BASIC 程式碼編譯轉換成 Java 程式碼來建立原生的 Android 應用程式。
- 二、透過以 Basic4android 開發 Android App 的分享，協助參與教師以現有的 Visual Basic 程式設計經驗，轉而開發 Android App。

## 貳、研習課程

時間	議程
8:40~9:00	報到
9:00~10:30	Android 與 B4A 基礎和開發環境安裝設定－建立第一個 Android App 和 AVD 的使用
10:30~10:40	茶水時間
10:40~12:10	B4A 的 Visual Basic 語法
12:10~13:00	午餐&午休
13:00~13:50	使用 B4A 的 GUI 設計工具
13:50~14:00	茶水時間
14:00~14:50	GUI 元件與活動
14:50~15:00	茶水時間
15:00~15:50	繪圖、檔案處理與 <b>正式版 B4A 函數庫</b> ←(正式版與試用版是最大的差異)
15:50~16:00	茶水時間
16:00~17:00	意圖、服務與狀態列通知

## 參、主講人：陳會安老師

學歷：美國猶他州州立大學電腦碩士資深專業電腦圖書作家

經歷：美商 PH 出版經理、專業電腦書作者

著作：用 Visual Basic 您也能開發 Android App--使用 Basic4android)

用實例學 Visual Basic 2013 / 2012 程式設計

JavaScript 與 jQuery 網頁設計範例教本

Java SE 7 與 Android 4.x 程式設計範例教本

PHP+ MySQL 與 jQuery Mobile 跨行動裝置網站開發

SQL Server 2012 資料庫設計與開發實務

...等超過 30 本著作

**GOTOP** 用 Visual Basic 您也能開發 Android App



## Android的基礎

- Android是一套使用Linux作業系統為基礎開發的Open Source作業系統，主要是針對手機等行動裝置使用的作業系統，現在Android已經逐漸擴充到平板電腦、筆電和其他領域，例如：電子書閱讀器、MP4播放器和Internet電視等。
- 目前擁有Android作業系統的是非營利組織的開放式手持裝置聯盟，Google公司則在幕後全力支持Android作業系統的開發計劃，並且在Android作業系統整合Google的Gmail、Youtube、Google地圖和Google Play等服務，作為主要的獲利來源。

**GOTOP** 用 Visual Basic 您也能開發 Android App



## Android的版本

Android版本	釋出日期	代號
1.5	2009/4/30	Cupcake (紙杯蛋糕)
1.6	2009/9/15	Donut (甜甜圈)
2.0/2.1	2009/10/26	Eclair (閃電泡芙，法式奶油夾心甜點)
2.2	2010/5/20	Froyo (冷凍乳酪)
2.3	2010/12/6	Gingerbread (薑餅)
3.0/3.1/3.2	2011/2/22	Honeycomb (蜂窩)
4.x	2011/10/19	Ice Cream Sandwich (冰淇淋三明治)
4.1/4.2/4.3	2012/6/28,10/29, 2013/7/24	Jelly Bean (雷根糖)
4.4	2013/9/3	KitKat (奇巧巧克力)

GOTOP 用 Visual Basic 您也能開發 Android App

## Android的系統架構

Android App

- Android 作業系統的系統架構像是在蛋糕店購買的一個多層蛋糕，在行動裝置的硬體和使用者之間是**Android軟體堆疊**（Android Software Stack），如右圖所示：

The diagram illustrates the Android Software Stack as a multi-layered cake. The layers from top to bottom are:

- 應用程式 (Applications)**: Includes 瀏覽器 (Browser), 聯絡人 (Contacts), 日曆 (Calendar), and 其他 (Others). This layer is highlighted with a red box.
- 應用程式框架 (Application Framework)**: Includes 活動管理 (Activity Management), 資源管理 (Resource Management), 內容提供者 (Content Provider), and 其他 (Others).
- 函數庫 (Libraries)**: Includes WebKit, OpenGL, SQLite, and 其他 (Others).
- Android 執行環境 (Android Execution Environment)**: Includes Dalvik VM 虛擬機器 (Dalvik VM Virtual Machine) and 核心Java函數庫 (Core Java Libraries).
- Linux核心 (Linux Kernel)**: Includes 顯示驅動 (Display Driver), 相機驅動 (Camera Driver), Wifi 驅動 (Wifi Driver), and 其他 (Others).

GOTOP 用 Visual Basic 您也能開發 Android App

## Android App的組成元件

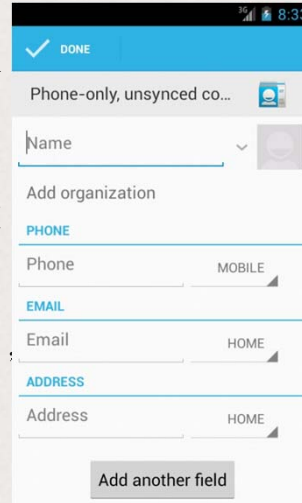
如同網頁中的超連結

- Android App的組成有四種元件：
  - 活動 (Activity)
  - 內容提供者 (Content Provider)
  - 廣播接收器 (Broadcast Receiver)
  - 服務 (Service)

The diagram shows a central green Android robot with four dashed lines connecting it to four circular components: Activity, Intent, Service, and Content Provider. A Broadcast Receiver component is also shown at the bottom left. A pink callout bubble points to the Intent component with the text "如同網頁中的超連結" (Like a hyperlink on a website).

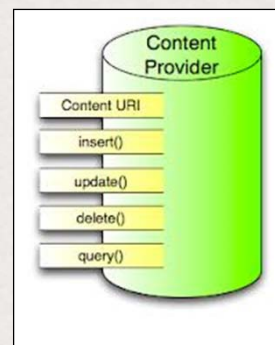
## 活動 (Activities)

- 活動是Android App與使用者互動的元件，用來定義使用者經驗。它也是**唯一可以讓使用者看到的元件**，即使用介面的螢幕畫面，一個活動就是使用者在行動裝置上看到的單一螢幕畫面。
- Android App可以建立一至多個活動來處理App所需的不同互動，如同Web網站擁有多頁網頁。



## 內容提供者 (Content Providers)

- 內容提供者是在不同Android App之間分享資料的介面，它是一組封裝的資料，提供客製化API來進行讀寫。例如：聯絡人應用程式並沒有儲存任何聯絡人資料，它是透過內容提供者取得聯絡人資訊：姓名、地址和電話等，換句話說，其他需要使用聯絡人資料的Android App，也都可以透過同一個內容提供者來存取聯絡人資料。



## 廣播接收器 ( Broadcast Receivers )

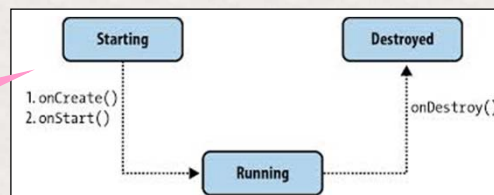
- 廣播接收器顧名思義是**用來接收廣播且做出回應**，這是**Android實作系統層級的廣播與回應機制**，事實上，Android系統本身就會常常發出廣播，例如：接到來電、收到簡訊、啟動相機裝置、時區改變、系統開機、電池剩餘電量過低或使用者選擇偏好語言時，Android系統都會發出廣播。



## 服務 ( Services )

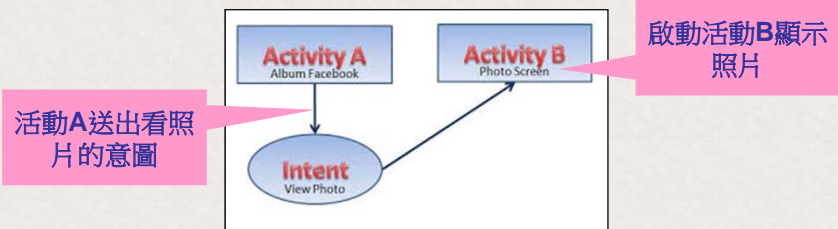
- 服務是在背景執行的行程，可以執行和活動一樣的工作，只是沒有使用介面。例如：在播放背景音樂時，之所以不會打斷我們發送簡訊或收發電子郵件，因為它是在背景執行的服務，才能讓音樂播放不中斷。Android作業系統本身就內建許多系統服務，我們可以直接使用API來使用這些服務，例如：定位服務。

服務的生命周期



## 意圖 (Intents)

- 意圖是一個啟動其他Android活動、服務和廣播接收器的非同步訊息。意圖可以告訴Android作業系統我想作什麼？執行什麼動作？作業系統是使用意圖篩選 (Intent Filters) 來找出可以處理的元件，例如：啟動其他活動、告訴指定服務可以啟動或停止與送出廣播。



## Android App的主要開發方式

- 方法一：Eclipse IDE+Android SDK(原生)
- 方法二：HTML5方案(Hybrid)-瀏覽器  
– jQuery Mobile+PhoneGap



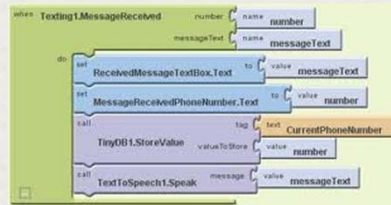
<http://fju.app.appery.io/Weather.html>  
<http://fju1.app.appery.io/>

## Android App開發使用的程式語言

- 目前教學上，Android App開發使用的程式語言主要有：
  - Java語言(物件導向程式語言)
  - App Inventor(拼圖)
  - HTML5+JavaScript(網頁程式語言)
  - **Basic4android+BASIC**

C++語言(RAD Studio XE 6)

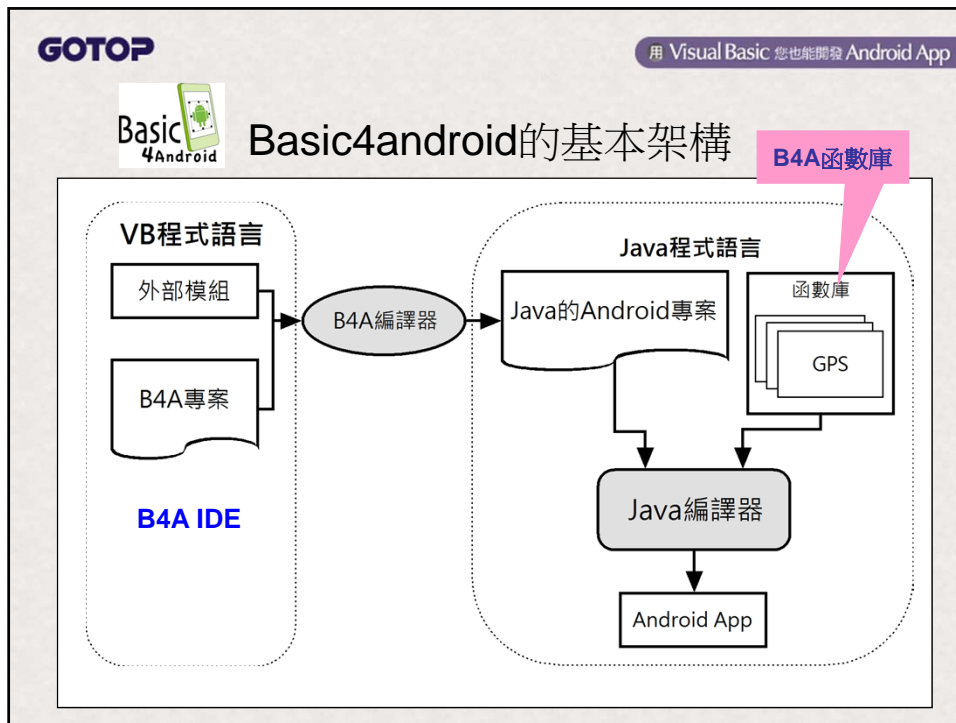
C#語言(Xamarin)



## Basic4android簡介

- Basic4android (B4A) 是以色列Anywhere Software公司開發的整合開發環境，這是針對Android平台開發的一套簡單且功能強大的快速應用程式開發工具 (Rapid Application Development, RAD)。
- Basic4android是一套Android App的整合開發環境IDE，可以讓Visual Basic語言的使用者直接使用BASIC語言來開發Android App，B4A能夠將開發者撰寫的BASIC程式碼編譯轉換成Java程式碼來建立原生Android App，換句話說，其執行效能並不會打折扣。





GOTOP 用 Visual Basic 您也能開發 Android App

### 為什麼Basic4android比較簡單

- 將Java物件導向程式設計降為VB語言的物件基礎程式設計(B4A本身也提供有物件導向語法)。
- 使用函數庫簡化Android權限設定，和擴充新功能，新版Android功能是使用B4A函數庫來擴充。
- 每一個函數庫就是一至多個物件，可以存取屬性和呼叫方法來使用所需的功能(原生是使用物件導向的繼承和覆寫等語法)，例如：GPS、資料庫、網路存取(JSON)等。

最新B4J免費開發工具，使用和B4A的相同觀念，可以讓使用者以VB語言開發跨Windows、Mac和Linux的視窗應用程式(使用JavaFx)。

## Basic4android VS App Inventor

- App Inventor的開發目的主要是針對沒有程式設計經驗的使用者，使用簡單的拼圖來建立程式，VB開發者仍然需要學習全新語言，再加上目前業界幾乎不可能使用此工具開發App，所以純萃只有教學用途，而且很難和Android SDK接軌。
- Basic4android是使用BASIC語言，可以延續已有VB程式設計經驗，再加上其架構是對應Android SDK，且最終會編譯成Java程式碼，不只可以獨立開發App，更可以進一步作為橋樑來學習原生Java的Android App開發。所以，本書架構事實上和Android SDK圖書十分相似。



## Basic4android的版本

- **Basic4android企業版**：提供2年免費升級服務，和Basic4android社群的完整存取權限。
- **Basic4android標準版**：提供2個月免費升級服務，和Basic4android社群的完整存取權限。
- **Basic4android測試版**：Basic4android免費試用版本，提供30天免費試用，一個限制功能版本的Basic4android，只能使用核心函數庫來建立基本Android App，而且沒有Basic4android社群的完整存取權限，只能下載部分B4A範例專案。

**GOTOP** 用 Visual Basic 您也能開發 Android App

Basic4Android 的官網

<http://www.basic4ppc.com/> ppc是Pocket PC

Anywhere Software Basic4Android

Home Screenshots Showcase Features Download Purchase Online Community Documentation Contact Us

Basic4Android is the simplest and most powerful Rapid Application Development (RAD) tool available for the Android platform

- ✓ Basic4Android includes all features needed to develop great, real-world Android applications
- ✓ Compiled applications are native Android applications, there is no extra run-time or any dependencies
- ✓ Our community with over 40k developers is waiting to help you get started

Download the free trial version

Among our customers: IBM, hp, GE

News: Basic4Android v3.00 is released!

Customer Spotlight: NASA

**GOTOP** 用 Visual Basic 您也能開發 Android App

下載與安裝B4A開發環境

- 步驟一：下載與安裝JDK  
– JDK 7以上版本
- 步驟二：下載與安裝ADT Bundle  
– 最新4.4，本書使用4.22版，目標版本是4.0
- 步驟三：下載與安裝Basic4android  
– B4A 2.71測試版
- 步驟四：啟動與設定Basic4android
- 步驟五：建立Android模擬器AVD

下載與安裝B4A開發環境.ppt



## 建立第一個AndroidApp

- 步驟一：新增與儲存B4A的Android App專案
- 步驟二：啟動Android模擬器
- 步驟三：編譯與執行Android App

建立第一個AndroidApp.ppt



## 使用B4A開發Android App的基本步驟

- **Step 1. 新增B4A專案：**建立B4A專案和替每一個B4A專案新增專屬的資料夾。
- **Step 2. 建立圖形使用介面：**開啟GUI設計工具建立圖形使用介面，依照規劃的版面配置新增所需的介面元件來編排使用介面。
- **Step 3. 設定介面元件屬性：**設定元件屬性值來調整尺寸、字型、色彩和外觀等。
- **Step 4. 建立和撰寫事件處理程序的程式碼：**依照活動、服務或介面元件觸發的事件，例如：**Button**的**Click**事件，建立所需的事件處理程序和撰寫處理的程式碼。
- **Step 5. 編譯與執行Android App：**在Windows開發電腦編譯Android App，和連接實機或啟動Android模擬器來測試執行。

其建立步驟和VB視窗應用程式沒有什麼不同

## B4A活動模組的程式結構

- Sub Process\_Globals
- End Sub
- Sub Globals
- End Sub
- Sub Activity\_Create(FirstTime As Boolean)
- End Sub
- Sub Activity\_Resume
- End Sub
- Sub Activity\_Pause (UserClosed As Boolean)
- End Sub

## B4A活動模組的程式結構-說明

- **Process\_Globals**程序：定義整個App各活動模組都能存取的全域變數；Globals是目前活動模組可以存取的全域變數。
- **Activity\_Create(FirstTime As Boolean)**程序：當活動建立時呼叫的程序，程序參數FirstTime判斷是否是第1次建立活動。
- **Activity\_Resume**程序：在建立活動呼叫Activity\_Create後馬上呼叫的程序來進入執行狀態，或從背景暫停狀態返回前景執行狀態時呼叫。
- **Activity\_Pause(UserClosed As Boolean)**程序：當活動從前景執行狀態進入背景暫停狀態時呼叫此程序，程序參數UserClosed是布林值，可以判斷是否是使用者操作才進入背景。



## B4A的命名規則

- 名稱不能使用B4A語言的關鍵字和B4A內建物件的名稱（不區分英文大小寫，例如：因為內建File物件，所以命名不可以是File或file）。
- 名稱必須是英文字母開頭，不能包含句點「.」、運算子、括號或空白，只能是英文字母A-Z、a-z、數字0-9和底線「\_」。
- 名稱不區分英文字母大小寫，abc、Abc、aBc、abC、ABc和ABC等都是同一個名稱。
- 在宣告的有效範圍內須唯一。

## B4A的關鍵字

- B4A語言的關鍵字和其他程式語言有些不同，它是指核心函數庫（**Core Library**）的**Keywords**型態成員，除了語法關鍵字外，還包含一些存取內建物件屬性、數學、檔案處理、對話方塊等方法的成員，例如：**Msgbox**(訊息視窗)和**Sender**(事件來源物件)等。

## B4A的註解

- B4A的註解文字是使用「'」符號開始的列，或在程式列上此符號之後的文字內容（**不支援Visual Basic語言的REM關鍵字**），如下所示：

```
' 大家好
```

```
Dim size1,size2 As Int ' 變數宣告
```

## 程式敘述的結束

- B4A可以使用「:」符號標示程式敘述的結束，我們可以在同一列程式碼包含2個程式敘述，第1個敘述是宣告size變數，第2個指定size變數值，如下所示：

```
Dim size As Int : size = 10
```

- 不同於**Visual Basic**語言的「:」符號是標示程式敘述的結束，**B4A**隱含是一個換行字元（**Carriage Return**），加上「:」表示是換行成二列程式碼，而非程式敘述結束。

### 太長的程式列

- B4A的單行程式碼如果太長，基於程式編排的需要，我們可以將它分成兩列來排列，此時，請在第1列程式碼的最後加上「\_」符號，如下所示：

```
Dim size1,size2,size3,size4,size5, _
size6,size7,size8,size9 As Int
```

**不支援VS 2010之後版本的隱含字串連接**

### 基本資料型態(源於Java語言)

- B4A語言支援8種基本資料型態，如下表所示：

資料型態	說明	範圍最小值	範圍最大值
Boolean	布林	False	True
Byte	位元組8Bits	-128	127
Short	短整數16Bits	-32768	32767
Int	整數16Bits	-2147483648	2147483647
Long	長整數64Bits	-9223372036854775808	9223372036854775807
Float	單精度浮點數 32Bits	1.4E-45	3.4028235 E38
Double	雙精度浮點數 64Bits	2.2250738585072014 E - 308	1.7976931348623157 E 308
Char	字元	N/A	N/A



**GOTOP** 用 Visual Basic 您也能開發 Android App

### 基本資料型態(對比VB語言)

- B4A語言支援8種基本資料型態，如下表所示：

B4A資料型態	VB資料型態
Boolean	Boolean
Byte	Byte
Short	Short
Int	Integer
Long	Long
Float	Single
Double	Double
Char	Char

**GOTOP** 用 Visual Basic 您也能開發 Android App

### 非基本資料型態

- 除了基本資料型態的其他資料型態，例如：使用 **Type** 關鍵字宣告的自訂型態、函數庫的物件、陣列或字串等都屬於非基本資料型態。其中陣列與字串型態的簡單說明，如下表所示：

資料型態	說明
String	字串，一種字元陣列
Array	陣列，相同資料型態的變數集合，使用索引值來存取

## 變數宣告

- 在B4A是使用Dim關鍵字宣告變數，As關鍵字指定資料型態，**B4A不支援Visual Basic語言的常數**。如下所示：

```
Dim Sum As Float
```

```
Dim size1,size2 As Int
```

```
Dim Capital, Interest, Rate As Double
```

- 程式碼宣告1個Float型態、2個整數Int型態和3個Double型態的變數。B4A在同一列宣告的多個變數可以分別指定不同的資料型態，如下所示：

```
Dim name As String,score As Int
```

## 變數的初值

- 我們可以在宣告變數同時指定變數的初值，如下所示：

```
Dim size3 As Int = 100
```

```
Dim Grade1 = 56.7, Grade2 As Float = 78.5
```

```
Dim i = 0, j = 2, k = 15 As Int
```

```
Dim name = "Joe" As String, rate = 1.1 As Double, flag  
= True As Boolean
```

- 程式碼在宣告多個變數的同時就指定初值，「=」等號可以在型態之後，也可以在As關鍵字之前。

### 自動(隱含)資料型態轉換

- B4A會自動依需求執行型態轉換，例如：自動將數字轉換成字串；字串轉換成數字。

### 明確資料型態轉換

- B4A明確執行型態轉換的作法是直接將它指定給欲轉換型態的變數，如下所示：  
`Dim intGrade1, intGrade2 As Int`  
`Dim dblGrade1 = 56.7, dblGrade2 = 78.5 As Double`
- 將Double變數型態明確轉換成Int變數時，只需使用指定敘述指定變數值即可，如下所示：  
`intGrade1 = dblGrade1`  
`intGrade2 = dblGrade2`
- 程式碼將Double型態的值轉換成整數型態的值，即變數intGrade1和intGrade2的值。

GOTOP 用 Visual Basic 您也能開發 Android App

### 算術運算式

- 算術運算式就是使用算術運算子建立的運算式，其說明**(VB語言的指數運算子是^)**如下表所示：

運算子	說明	運算式範例
Power	指數	Power(7, 2) = 49，即7 <sup>2</sup>
-	負號	-7
*	乘法	2 * 7 = 14
/	除法	7 / 2 = 3.5
Mod	餘數	7 Mod 2 = 1
+	加法	7 + 2 = 9
-	減法	7 - 2 = 5

GOTOP 用 Visual Basic 您也能開發 Android App

### 關係運算式

- 關係運算子沒有優先順序的分別，通常是使用在迴圈和條件敘述作為判斷條件，如下表所示：

運算子	說明	運算式範例	結果
=	等於	6 = 4	False
<>	不等於	6 <> 4	True
<	小於	6 < 4	False
>	大於	6 > 4	True
<=	小於等於	6 <= 4	False
>=	大於等於	6 >= 4	True

**不支援VB語言的Is, IsNot和Like運算子**

## 布林運算式

- 當迴圈和條件敘述的判斷條件不只一個時，我們可以使用布林運算子連接多個關係運算式來建立更複雜的判斷條件，其說明如下表所示：

運算子	運算式範例	結果
Not (A)	Not (5 > 3)	False
Not (B)	Not (4 <= 2)	True
A AND B	5 > 3 AND 4 <= 2	False
A OR B	5 > 3 OR 4 <= 2	True

**B4A不支援VB語言的Xor, AndAlso和OrElse運算子  
VB Not A ; B4A是Not (A)**

## 字串連接運算子

- B4A的字串連接運算子是「&」，可以連接多個字串變數或字串值，其說明如下所示：

運算子	說明	運算式範例
&	字串連接	"ab" & "cd" = "abcd"

**B4A語言的換行符號是CRLF關鍵字  
" " 符號是Quote**

## If/Then/Else If條件敘述

- If/Then/Else If條件敘述新增Else If關鍵字（**VB語言在Else If之間沒有空格**）建立多選一條件敘述，如下：

```
If dayValue = 1 Then
    out = out & "星期日" & CRLF
Else If dayValue = 2 Then
    out = out & "星期一" & CRLF
Else If dayValue = 3 Then
    out = out & "星期二" & CRLF
...
Else If dayValue = 7 Then
    out = out & "星期六" & CRLF
Else
    out = out & "無法分辨是星期幾" & CRLF
End If
```

## Select/Case條件敘述

- **Select/Case**多選一條件敘述可以依照符合條件執行不同區塊的程式碼(**VB是Select Case**)，如下所示：

```
Select dayValue
    Case 1: out = out & "周日假期" & CRLF
    Case 2, 3, 4, 5, 6
        out = out & "工作天" & CRLF
    Case 7: out = out & "周末假日" & CRLF
    Case Else
        out = out & "無法分辨是星期幾" & CRLF
End Select
```

## For/Next迴圈

- For/Next迴圈可以執行固定次數的迴圈，使用Step值來增加或減少計數器，如果Step為1可以省略Step關鍵字(**Next**之後不支援加上變數*i*，**Next i**)。例如：使用For/Step/Next迴圈，每次增加1，執行1加到10的迴圈，如下所示：

```
For i = 1 To 10
    sum = sum + i
    out = out & i & " "
Next
out = out & CRLF & "1加到10: " & sum & CRLF
```

## Exit關鍵字

- 在迴圈尚未到達結束條件時，我們可以使用Exit關鍵字強迫跳出迴圈來結束迴圈執行，B4A只需使用Exit關鍵字；**Visual Basic**是**Exit Do**或**Exit For**，如下所示：

```
For counter = 1 To 100
    If counter > 10 Then Exit
    sum = sum + counter
    out = out & counter & " "
Next
```

## B4A不支援後測式迴圈

- **B4A不支援後測式迴圈**，我們可以使用Do/Loop迴圈，然後在無窮迴圈使用If條件配合Exit關鍵字來中斷迴圈執行來建立後測式迴圈，如下所示：

```
counter = 1 : sum = 0
Do While counter >= 1
    sum = sum + counter
    out = out & counter & " "
    counter = counter + 1
    If counter > 10 Then Exit
Loop
out = out & CRLF & "1加到10: " & sum & CRLF
```

## Continue關鍵字

- B4A的Continue關鍵字可以在迴圈尚未執行完前，馬上執行下一次迴圈，如下所示：

```
For counter = 0 to 100
    .....
    If counter Mod 2 = 0 Then Continue
    .....
Next
```

- B4A只需Continue關鍵字；**Visual Basic語言是Continue Do或Continue For。**



## 例外處理敘述

- B4A支援和Visual Basic語言相同的Try/Catch/End Try，不過，**B4A不支援Finally程式區塊**，如下所示：

```
Try
  Dim In As InputStream
  In = File.OpenInput(File.DirInternal, "MissingFile.txt")
Catch
  lblOutput.Text = LastException.Message
End Try
```

- 在Catch使用LastException預建物件取得進一步的例外資訊，即Message屬性。

## 宣告一維陣列

- B4A陣列同樣也是使用Dim關鍵字宣告，我們可以在宣告時指定陣列尺寸。例如：一維整數陣列的宣告，如下所示：

```
Dim Names(3) As String
```

- 程式碼宣告3個元素的Names陣列，括號是元素數（**Visual Basic陣列是最大索引值**），其索引值是從0開始，範圍是0~2共3個元素。
- 我們可以使用變數指定陣列尺寸，如下所示：

```
Dim arrSize As Int : arrSize = 3
Dim Names(arrSize) As String
```

## 一維陣列初始設定

- B4A陣列的初值需要使用Array關鍵字來宣告，如下所示：

```
Dim Grades() As Int
```

```
Grades = Array As Int(53, 76, 85)
```

- 程式碼宣告一維整數陣列Grades但沒有指定尺寸，其尺寸是Array As後方型態Int括號中的初值個數，陣列索引的最大值就是初值個數減一，即2。

## For/Next迴圈走訪一維陣列

- 我們可以使用For/Next迴圈走訪陣列的每一個元素，如下所示：

```
For i = 0 To Names.Length - 1
```

```
    out = out & Names(i) & "-" & Grades(i) &  
    CRLF
```

```
Next
```

- 程式碼使用陣列物件的Length屬性取得陣列尺寸，減一就是最大索引值，可以從0至最大索引值來一一取出每一個元素。

### 建立沒有參數和傳回值的程序

- 如果Sub宣告的程序沒有參數，在宣告時不加上空括號，呼叫時一定不能使用空括號（**如果有空括號，B4A編譯器會誤認為是陣列**），例如：更改App標題文字的ChangeTitle程序，如下所示：

```
Sub ChangeTitle  
    Activity.Title = "建立B4A程序"  
End Sub
```

- 程序在Sub關鍵字之後是程序名稱，因為沒有參數，可以加上空括號，即ChangeTitle()，也可以不加。在Sub/End Sub之間是程序的程式區塊。

### 呼叫沒有參數和傳回值的程序

- 呼叫程序只需使用ChangeTitle（**B4A不支援Visual Basic語言的Call關鍵字**），不可以使用ChangeTitle()，如下所示：

```
ChangeTitle
```

### 建立擁有參數沒有傳回值的程序

- 在程序名稱後的括號可以加上傳入參數，如果不只一個，請使用「,」號分隔，例如：顯示輸出結果的PrintResult程序，如下所示：

```
Sub PrintResult(str As String)
    lblOutput.Text = str
End Sub
```

- 程序擁有一個參數str，其資料型態是String。

### 呼叫擁有參數沒有傳回值的程序

- 因為程序擁有參數，所以在呼叫時，除了程序名稱，還需要使用括號來指定傳入值，如下所示：

```
PrintResult(out)
```

- 程序呼叫傳入的參數值就是變數out的值。

### 建立擁有參數和傳回值的程序

- **B4A**的程序也是函數一樣可以傳回值，所以可以指定傳回值型態，例如：計算利息的**CallInterest**程序，如下所示：

```
Sub CallInterest(capital As Double, rate As Double) As Double
    Dim amount As Double
    amount = capital * rate / 100
    Return amount
End Sub
```

- 程序擁有2個參數，在**As**關鍵字之後是傳回值型態，**B4A**程序可以使用**Return**關鍵字來傳回值。

### 呼叫擁有參數和傳回值的程序

- 因為程序有傳回值，在呼叫程序時通常是位在運算式指定敘述的右邊，如下所示：

```
interest = CallInterest(capital, rate)
```

不支援VB語言的Function/End Function

## 傳值呼叫

- **B4A**是依資料型態區分傳值或傳址呼叫。傳值呼叫是指將變數的值傳入程序，所以不會更改傳入的變數值，B4A基本資料型態的變數和String型態都是使用傳值呼叫，如下所示：

```
Sub CallByVal(value As Int, str As String)
    value = 45
    str = "江小魚"
End Sub
```

## 傳址呼叫

- 傳址呼叫是將變數實際儲存位址傳入程序，B4A的非基本資料型態（String除外）參數都是使用傳址呼叫，它是將複製的參考傳入程序，可以更改參數成員（例如：陣列元素值），但不能更變數本身的參考（即陣列變數本身），如下所示：

```
Sub CallByRef(Tests() As Int)
    Dim MyArr(1) As Int
    MyArr(0) = 30
    Tests(0) = 50
    Tests = MyArr
End Sub
```

## 區域範圍

- 在B4A程序內宣告的變數，只能在此程序的程式區塊中使用，其他程序並不能存取，稱為區域變數(變數a和c)，如下所示：

```
Sub SubA()  
    Dim a, c As Int  
    a = 3  
    c = 10  
End Sub
```

## 全域變數範圍-Process\_Globals程序

- Process\_Globals程序：在此程序宣告的變數稱為行程變數，可以在行程中B4A專案的所有模組存取，不過，不是所有型態都可以宣告成行程變數，例如：介面元件的物件就不可以，如下所示：

```
Sub Process_Globals  
    Dim g As Int : g = 5  
End Sub
```

### 全域變數範圍-Globals程序

- **Globals**程序：在此程序宣告的變數稱為活動變數，可以在目前活動模組的所有程序來存取，如下所示：

Sub Globals

Dim lblOutput As Label

Dim a As Int : a = 1

Dim b As Int : b = 2

End Sub

### 變數範圍的注意事項

- 在**Process\_Globals**和**Globals**程序宣告全域變數後，其他程序就算宣告同名變數，存取的仍然是全域變數，因為除了第**1**次宣告變數外，在**B4A**重複宣告同名變數，當轉換成**Java**程式碼後，程式碼只是初始變數成為預設值，並不會建立全新變數。



## B4A的集合物件

- B4A支援兩種集合物件，如下所示：
  - List物件
  - Map物件
- B4A有很多物件的方法都是使用List和Map物件作為參數或傳回值，例如：檔案處理和建立選擇對話方塊。



## GUI工具-建立兒童數學訓練

- 步驟一：啟動B4A新增專案
- 步驟二：啟動Android模擬器
- 步驟三：開啟GUI設計工具連接Android模擬器
- 步驟四：在GUI設計工具新增介面元件
- 步驟五：在GUI設計工具設定介面元件的屬性
- 步驟六：自動產生程式碼與儲存版面配置檔
- 步驟七：編輯事件處理程序的程式碼
- 步驟八：編譯與執行Android App

**GOTOP** 用 Visual Basic 您也能開發 Android App

### B4A的GUI介面元件 VS VB控制項

B4A介面元件	Visual Basic控制項
Activity	Form
Button	Button
CheckBox	CheckBox
EditText	TextBox
ImageView	PictureBox
Label	Label
ListView	ListView
Panel	Panel
RadioButton	RadioButton
ScrollView	N/A
SeekBar	TrackBar(類似)
Spinner	N/A
TabHost	TabControl
ToggleButton	N/A
WebView	WebBrowser

**GOTOP** 用 Visual Basic 您也能開發 Android App

### GUI介面元件的共同屬性

屬性	說明
Name	元件名稱，建議使用有意義的名稱，而且在開頭使用3個字元來標示元件種類，例如：lbl、btn、edt等
Type	元件型態的唯讀屬性
Event Name	處理事件的事件處理程序名稱，預設與Name屬性值相同。事件處理程序名稱的格式是此屬性值加上「_」符號，再加上事件名稱，例如：Event Name屬性值Button1的Click事件處理程序名為Button1_Click
Parent	父元件名稱（有些介面元件是容器元件，可以包含子元件），最上層是Activity活動
Left	與父元件左方邊界的X軸座標，單位是dip
Top	與父元件上方邊界的Y軸座標，單位是dip
Width	元件寬度，單位是dip
Height	元件高度，單位是dip
Enabled	此元件是否可以使用，True是可以；否為不可以
Visible	此元件是否可見，True是可見；否為隱藏
Tag	此屬性是用來儲存額外資訊，以簡化程式設計

**GOTOP** 用 Visual Basic 您也能開發 Android App

### 事件種類

介面元件	支援的事件種類
Activity	Click、LongClick、Touch、KeyPress、KeyUp、WindowFocusChanged
AutoCompleteEditText	ItemClick、TextChanged、EnterPressed、FocusChanged
Button	Click、LongClick、Down、Up
CheckBox	CheckedChange
EditText	EnterPressed、FocusChanged、TextChanged
HorizontalScrollView	ScrollChanged
ImageView	Click、LongClick
Label	Click、LongClick
ListView	ItemClick、ItemLongClick
Panel	Click、LongClick、Touch
RadioButton	CheckedChange
ScrollView	ScrollChanged
SeekBar	ValueChanged
Spinner	ItemClick
TabHost	Click、LongClick、TabChanged
ToggleButton	CheckedChange
WebView	PageFinished、OverrideUrl、UserAndPasswordRequired

**GOTOP** 用 Visual Basic 您也能開發 Android App

### 建立事件處理程序

- 我們可以在**B4A**專案新增事件處理程序的程式碼，實務上，主要是從**GUI**設計工具自動產生元件宣告和事件處理程序，當然，在程式碼編輯器一樣可以自行輸入程式碼來建立事件處理程序。
  - 在**GUI**設計工具產生事件處理程序
  - 在程式碼編輯器自行輸入事件處理程序

## 介面元件的事件處理程序

- B4A和Visual Basic語言都是使用事件驅動程式設計來建立應用程式，整個程式結構就是在處理各種事件的事件處理程序，當我們使用GUI設計工具新增介面元件（如同Visual Basic控制項）後，例如：新增名為myButton按鈕元件，就可以在Main活動模組建立處理Click事件的事件處理程序，如下所示：

Event Name屬性  
值, 不是Name屬性

```
Sub myButton_Click
    MsgBox("按下按鈕!", "")
End Sub
```

## Spinner清單元件

- Android提供兩種清單元件：Spinner和ListView元件，可以顯示一長串清單項目來供使用者選擇。
- Spinner元件類似Windows作業系統的下拉式清單方塊，它是一個單選的清單元件。



**GOTOP** 用 Visual Basic 您也能開發 Android App

### 建立Spinner清單元件

- 在B4A的GUI設計工具可以執行「**Add View>Spinner**」指令新增**Spinner**元件，不過，沒有提供介面來新增選項，我們需要使用程式碼呼叫**Add()**方法來新增選項，如下所示：

```
Dim spnType As Spinner
...
spnType.Add("美金")
spnType.Add("日元")
spnType.Add("人民幣")
```

Add()方法新增選項

**GOTOP** 用 Visual Basic 您也能開發 Android App

### Spinner清單元件的ItemClick事件

- Spinner**元件是一個下拉式清單方塊，當使用者選擇選項後，就觸發**ItemClick**事件，在事件處理程序處理選擇，如下所示：

```
Sub spnType_ItemClick(Position As Int, Value As Object)
    Select Position
        Case 0 : rate = 29.9 '美金
        Case 1 : rate = 0.2949 '日元
        Case 2 : rate = 4.861 '人民幣
    End Select
End Sub
```

**GOTOP** 用 Visual Basic 您也能開發 Android App  
專案: Lesson4\ListView清單介面\匯率轉換程式II.b4a

## ListView清單介面

- **ListView**元件是各種行動裝置作業系統相當常見的使用介面，實務上，如果整個活動只擁有一個**ListView**元件，此時的**ListView**元件是一個選單用途的元件，而且，通常就是**Android App**的主選單。



**GOTOP** 用 Visual Basic 您也能開發 Android App

## 建立單行項目的ListView元件

- 在**B4A**的**GUI**設計工具可以執行「**Add View>ListView**」指令新增**ListView**元件，不過，**B4A**沒有提供介面來新增選項，我們需要自行使用程式碼來新增選項，依需求可以建立單行和兩行內容的項目。
- **ListView**元件呼叫**AddSingleLine()**方法新增**ListView**元件的單行選項，參數是選項名稱的字串，如下所示：

```
Dim ListView1 As ListView  
...  
ListView1.AddSingleLine("陳會安")
```

**AddSingleLine()**方法  
新增選項

## ListView 元件單行項目的傳回值

- ListView 元件的項目還可以指定所需的傳回值，如下所示：

```
Dim lstType As ListView
```

```
...
```

```
lstType.AddSingleLine2("美金", 29.9)
```

```
lstType.AddSingleLine2("日元", 0.2949)
```

```
lstType.AddSingleLine2("人民幣", 4.861)
```

AddSingleLine2() 方法  
新增選項

- **AddSingleLine2()** 方法新增 ListView 元件的單行選項，第 2 個參數指定傳回值，可以呼叫 **GetItem()** 方法或在 **ItemClick** 事件處理程序取得此傳回值，以此例的傳回值是匯率。

## 單行項目 ListView 元件的 ItemClick 事件

- ListView 元件當使用者選擇選項後，就會觸發 ItemClick 事件，如下所示：

```
Sub ListView1_ItemClick(Position As Int, Value As Object)
```

```
    Dim amount As Double = edtAmount.Text
```

```
    lblOutput.Text = "兌換成新台幣的金額: " & (amount * Value)
```

```
End Sub
```

- 第 1 個參數是選項索引位置（從 0 開始），第 2 個參數是選項值，**AddSingleLine()** 方法新增選項，其值是參數的選項名稱；**AddSingleLine2()** 方法新增選項，其值是第 2 個參數 Value 的傳回值。

**GOTOP** 專案:Lesson4\訊息視窗\猜數字遊戲.b4a 用 Visual Basic 您也能開發 Android App

### 訊息視窗

- B4A支援源於Visual Basic的Msgbox訊息視窗和Android原生的Toast彈跳訊息框。



**GOTOP** 用 Visual Basic 您也能開發 Android App

### Msgbox訊息視窗

- 訊息視窗可以在對話方塊顯示一段訊息和一個OK按鈕，例如：在Android App建立一個名為「猜數字遊戲」的訊息視窗，顯示開始遊戲的訊息文字，如下所示：  
**Msgbox**("請輸入數字開始遊戲", "猜數字遊戲")
- 第1參數是訊息文字，第2個參數是標題文字。確認訊息視窗是使用**Msgbox2()**方法，如下所示：  
**Dim result As Int**  
**result = Msgbox2("共猜: " & time & \_**  
**"次, 按[確定]鈕再玩一次.", "猜數字遊戲", \_**  
**"確定", "", "結束", Null)**



## Msgbox訊息視窗-取得使用者操作

- **Msgbox2()**方法的傳回值是使用者按下了哪一個按鈕，我們可以使用**If**條件來判斷使用者的操作，如下所示：

```
If result = DialogResult.POSITIVE Then
```

```
...
```

```
End If
```

- 條件判斷是否按下**POSITIVE**鈕，**DialogResult.CANCEL**是**CANCEL**鈕；**DialogResult.NEGATIVE**是**NAGATIVE**鈕。

## Toast彈跳訊息框

- **Toast**訊息是在行動裝置顯示一個彈跳訊息框，可以在此訊息框顯示一段訊息文字，且只保留一段時間。例如：使用**Toast**彈跳訊息框顯示目前猜測的數字太大，如下所示：

```
ToastMessageShow(num & " 數字太大...",  
False)
```

- **ToastMessageShow()**方法是關鍵字，第1個參數是顯示的訊息字串，在第2個參數決定顯示時間，值**False**較短（2秒）；**True**顯示時間較長（3秒）。

**GOTOP** 專案:Lesson4\選項選單\溫度轉換程式.b4a 用 Visual Basic 您也能開發 Android App

### 選項選單

- Android的選項選單（Options Menu）是當使用者按下行動裝置的【MENU】鍵後，在下方顯示的選單。



Options Menu

**GOTOP** 用 Visual Basic 您也能開發 Android App

### 新增選項選單

- B4A的Activity物件是活動本身，此物件提供3個方法來新增選項選單，第1個方法是建立單純文字項目的選項，如下所示：  
`Activity.AddMenuItem("轉成攝氏", "ConvertC")`
- AddMenuItem()**方法可以新增選單的選項，第1個參數是選項名稱，第2個參數是事件名稱。

## 選項的事件處理

- 當使用者選擇選項，就會產生Click事件，呼叫名為ConvertC\_Click的事件處理程序（ConvertC是第2個參數的事件名稱Event Name），如下所示：

```
Sub ConvertC_Click  
...  
End Sub
```



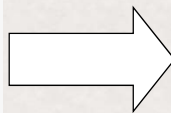
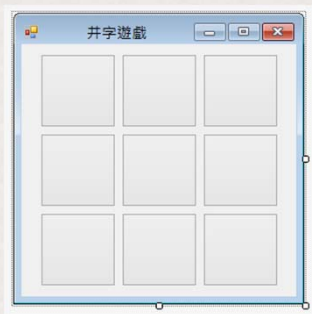
## B4A模組種類

- **活動模組 (Activity Modules)**：每一個活動都擁有專屬活動模組，負責處理活動的生命周期，**對比VB的Form表單，多活動對比多表單**。
- **程式碼模組 (Code Modules)**：程式碼模組的內容是程式碼，它是一個工具箱用來支援Android App開發，在程式碼模組之中不允許擁有任何活動，**對比VB的Module模組**。
- **服務模組 (Service Modules)**：每一個Android服務都擁有對應的服務模組，這是一些在背景執行的工作。
- **類別模組 (Class Modules)**：類別模組是Basic4android 2.0版新增的模組，這是一種建立物件導向程式設計類別的模組。

### VB專案 → B4A專案(步驟)

- **Step 1**：參考VB表單介面的控制項，在B4A使用GUI工具建立對應的使用介面，建議使用相同的Name屬性值。
- **Step 2**：使用B4A的GUI工具建立介面元件所需的事件處理程序。
- **Step 3**：複製事件處理程式碼，從Form1.vb程式檔案至B4A的事件處理程序。
- **Step 4**：將VB程式碼修改成B4A的VB程式碼。

### VB專案 → B4A專案(使用介面)



專案:VS2012\井字遊戲.sln

專案:Lesson4\井字遊戲\井字遊戲.b4a

### VB專案 → B4A專案(全域變數)

```
Public Class Form1
    Dim turn As Boolean = True
    Dim board(2, 2) As Integer

    Private Sub Form1_Load(sender As Object, e As EventArgs) Handles MyBase.Load
        ' 初始陣列元素值為-3
        For i As Integer = 0 To 2
            For j As Integer = 0 To 2
                board(i, j) = -3
            Next j
        Next i
    End Sub
End Class
```

```
Sub Globals
    Dim turn As Boolean = True
    Dim board(3, 3) As Int
    Dim Button1 As Button : Dim Button2 As Button
    Dim Button3 As Button : Dim Button4 As Button
    Dim Button5 As Button : Dim Button6 As Button
    Dim Button7 As Button : Dim Button8 As Button
    Dim Button9 As Button
End Sub
```

專案:VS2012\井字遊戲.sln

專案:Lesson4\井字遊戲井字遊戲.b4a

### VB專案 → B4A專案(函數)

```
Function CheckWin()
    Dim total, total1 As Integer
    ' 檢查每一列和每一欄
    For i As Integer = 0 To 2
        total = 0 : total1 = 0
        For j As Integer = 0 To 2
            total += board(i, j) ' 每一列
            total1 += board(j, i) ' 每一欄
        Next j
        If total = 0 Or total1 = 0 Then
            MsgBox("遊戲結束!○贏")
            Return True
        End If
        If total = 3 Or total1 = 3 Then
            MsgBox("遊戲結束!×贏")
            Return True
        End If
    Next i
    ' 檢查對角線
    total = board(0, 0) + board(1, 1) + board(2, 2)
    total1 = board(2, 0) + board(1, 1) + board(0, 2)
    If total = 0 Or total1 = 0 Then
        MsgBox("遊戲結束!○贏")
        Return True
    End If
    If total = 3 Or total1 = 3 Then
        MsgBox("遊戲結束!×贏")
        Return True
    End If
    Return False
End Function
```

```
Sub CheckWin
    Dim total, total1 As Int
    ' 檢查每一列和每一欄
    For i = 0 To 2
        total = 0 : total1 = 0
        For j = 0 To 2
            total = total + board(i, j) ' 每一列
            total1 = total1 + board(j, i) ' 每一欄
        Next
        If total = 0 OR total1 = 0 Then
            MsgBox("遊戲結束!○贏", "")
            Return
        End If
        If total = 3 OR total1 = 3 Then
            MsgBox("遊戲結束!×贏", "")
            Return
        End If
    Next
    ' 檢查對角線
    total = board(0, 0) + board(1, 1) + board(2, 2)
    total1 = board(2, 0) + board(1, 1) + board(0, 2)
    If total = 0 OR total1 = 0 Then
        MsgBox("遊戲結束!○贏", "")
        Return
    End If
    If total = 3 OR total1 = 3 Then
        MsgBox("遊戲結束!×贏", "")
        Return
    End If
End Sub
```

專案:VS2012\井字遊戲.sln

專案:Lesson4\井字遊戲井字遊戲.b4a

**GOTOP** 用 Visual Basic 您也能開發 Android App

## VB專案 → B4A專案(事件處理程序)

專案:VS2012\井字遊戲.sln
專案:Lesson4\井字遊戲\井字遊戲.b4a

**GOTOP** 用 Visual Basic 您也能開發 Android App

## 第二個VB->B4A轉換範例

→

專案:VS2012\拉霸遊戲.sln
專案:Lesson4\拉霸遊戲\拉霸遊戲.b4a

**GOTOP** 用 Visual Basic 您也能開發 Android App

Basic4Android **B4A活動的生命周期**

- Android程式設計的主要工作是撰寫程式碼來回應Android App產生的狀態改變。B4A活動簡化了Android活動的生命周期(沒有停止)，同樣的，我們重視的是不同狀態之間的轉換，如下圖所示：

```

    graph LR
      Start[啟動] -- "1: Activity_Create  
2: Activity_Resume" --> Running[執行]
      Running -- "Activity_Pause" --> Paused[暫停]
      Paused -- "Activity_Resume" --> Running
      Paused -- "Activity.Finish()" --> Removed[移除]
  
```

**GOTOP** 用 Visual Basic 您也能開發 Android App

### 活動生命周期的狀態說明-1

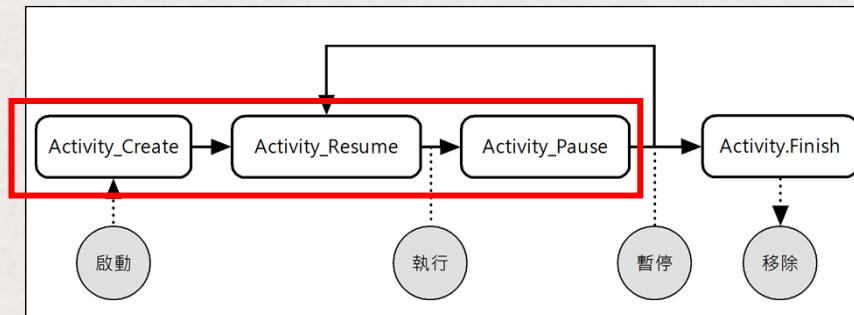
- 啟動狀態**：當Android App不在記憶體中時，執行程式就是進入啟動狀態，B4A會依序呼叫Activity\_Create和Activity\_Resume兩個程序來轉換至執行狀態。
- 執行狀態**：活動位在執行狀態（移至前景）表示目前在螢幕上顯示且與使用者進行互動，Android作業系統在任何時間點都只會有一個處於執行狀態的活動，執行狀態的活動擁有最高權限使用記憶體與資源來提昇程式執行效率，以便更快回應使用者的操作。

## 活動生命周期的狀態說明-2

- **暫停狀態**：暫停狀態的活動是從前景移至背景，此時的App仍然保留在記憶體，只是使用者看不到，B4A會呼叫Activity\_Pause程序，從執行狀態轉換至暫停狀態。
- **移除狀態**：移除狀態的活動是已經刪除的活動(可呼叫Activity.Finish()方法刪除)，表示已經釋放活動佔用的資源且不存在記憶體。活動管理員會自動依據記憶體的使用情況，決定活動是否需要刪除，以便空出更多記憶體讓執行狀態的活動能夠正常的運作。

## 活動生命周期的事件

- 在B4A活動的整個生命期中，共有3個程序會在活動的狀態轉換時呼叫，這些程序可以讓開發者回應狀態改變，撰寫所需的B4A程式碼(B4A簡化了Android活動的生命周期)，如下圖所示：





### Activity\_Create(FirstTime As Boolean)程序

- 當使用者啟動Android App，或更改裝置設定（活動會刪除後再建立），例如：旋轉螢幕，就會建立活動和呼叫Activity\_Create程序，此程序的主要目的是載入版面配置檔，或使用程式碼建立版面配置的使用介面。
- 在此程序擁有1個FirstTime布林參數，可以判斷對於目前的行程來說，活動是否是第1次建立（值True為第1次），通常，我們會使用此參數來初始Process\_Globals程序宣告的行程變數。

### Activity\_Pause(UserClosed As Boolean)程序

- 每一次當活動從前景執行狀態進入背景暫停狀態時，就會呼叫Activity\_Pause程序，或當活動在前景，但使用者更改裝置設定導致活動暫停和移除時，也會呼叫Activity\_Pause程序。
- 程序的參數UserClosed是一個布林值，可以判斷是否是使用者操作才進入背景。

## Activity\_Resume程序

- **Activity\_Resume**程序是在建立活動呼叫**Activity\_Create**後馬上呼叫，然後進入執行狀態，或從背景暫停狀態返回前景執行狀態時呼叫。
- 當我們啟動Android App建立活動，或在目前活動呼叫**StartActivity()**方法來啟動其他活動，建立活動的過程一定會依序呼叫**Activity\_Create**和**Activity\_Resume**程序後，才進入前景的執行狀態。

## 測試活動生命週期的事件

- 在B4A IDE右方【Logs】標籤可以看到呼叫程序的訊息，如下圖所示：

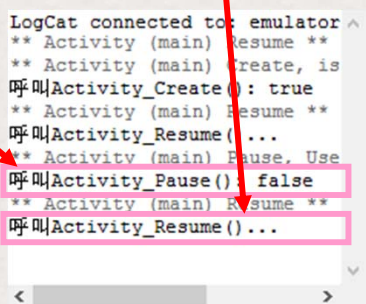
```
LogCat connected to: emulator ^
** Activity (main) Resume **
** Activity (main) Create, is
呼叫Activity_Create(): true
** Activity (main) Resume **
呼叫Activity_Resume()...
```

- 訊息顯示呼叫**Activity\_Create**且是第1次，然後呼叫**Activity\_Resume**，現在活動已經進入**執行**狀態，可以在模擬器看到執行結果，

**GOTOP** 專案:Lesson4\活動生命週期活動生命週期.b4a 用 Visual Basic 您也能開發 Android App

### 測試活動生命週期的事件

- 測試一：請在模擬器按F3鍵執行打電話程式，然後按【返回】鍵離開打電話程式，可以再度看到B4A程式，此時在【Logs】標籤可以看到此活動暫停(參數值為false)，和再度啟動，如下圖所示：

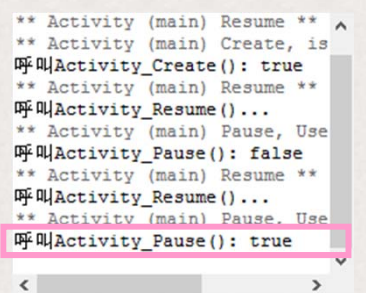


```
LogCat connected to emulator ^
** Activity (main) Resume **
** Activity (main) Create, is
呼叫Activity_Create(): true
** Activity (main) Resume **
呼叫Activity_Resume()...
** Activity (main) Pause, Use
呼叫Activity_Pause(): false
** Activity (main) Resume **
呼叫Activity_Resume()...
```

**GOTOP** 專案:Lesson4\活動生命週期活動生命週期.b4a 用 Visual Basic 您也能開發 Android App

### 測試活動生命週期的事件

- 測試二：現在B4A專案是執行狀態，請按【返回】鍵離開活動，可以在【Logs】標籤看到再次呼叫Activity\_Pause程序，此次的參數值為true，因為是使用者結束活動，如下圖所示：



```
** Activity (main) Resume **
** Activity (main) Create, is
呼叫Activity_Create(): true
** Activity (main) Resume **
呼叫Activity_Resume()...
** Activity (main) Pause, Use
呼叫Activity_Pause(): false
** Activity (main) Resume **
呼叫Activity_Resume()...
** Activity (main) Pause, Use
呼叫Activity_Pause(): true
```

## Canvas物件的基礎

- **Canvas物件可以在其他介面元件或Bitmap物件上繪圖**，當我們初始Canvas物件指定參數的介面元件後，就會建立一個Bitmap物件，並且將介面元件的背景複製到此Bitmap物件來建立Canvas物件，我們所繪出的圖形是繪在此Bitmap物件上。
- Canvas物件可以在**Activity活動、ImageView元件、Panel元件和Bitmap物件**上繪圖，使用Canvas物件繪出的圖形並不會馬上更新使用介面來顯示，我們需要呼叫目標元件的**Invalidate()**方法來更新元件，才能顯示繪出的圖形。

## 建立Canvas畫布 – 宣告Canvas物件和Panel元件

- Canvas物件是一個繪圖物件，提供繪圖方法和將其他介面元件或Bitmap物件建立成畫布，筆者稱為Canvas畫布。
- 首先，我們需要在Globals程序宣告Canvas物件和欲繪圖的Panel元件，如下所示：

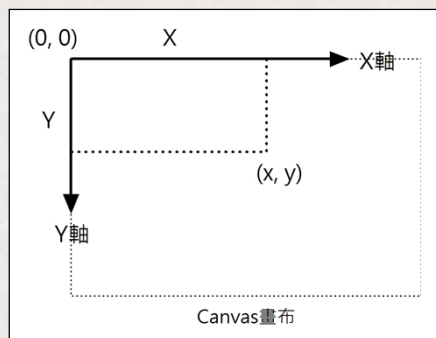
```
Sub Globals  
    Dim pnlGraph As Panel  
    Dim cvsGraph As Canvas  
End Sub
```

### 建立Canvas畫布 - Panel元件的畫布

- 我們準備建立一張Panel元件的畫布，如下所示：  
`cvsGraph.Initialize(pnlGraph)`  
`cvsGraph.DrawColor(Colors.White)`
- 上述程式碼呼叫Initialize()方法初始Canvas物件，參數是Panel元件，然後呼叫DrawColor()方法繪出畫布的背景色彩，以此例是白色。
- 現在，我們可以在活動模組的Activity\_Resume程序呼叫Canvas物件的方法來繪圖。

### Canvas畫布的座標系統

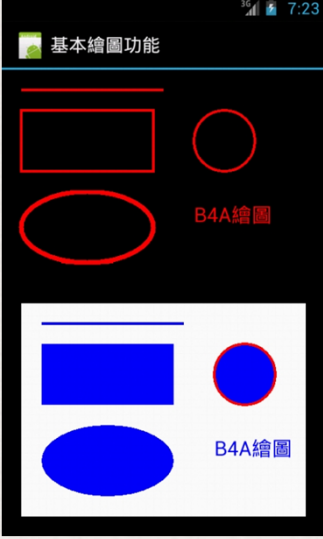
- Canvas畫布是一個長方形區域，即介面元件或Bitmap物件的尺寸，在左上角為原點，其座標是(0, 0)，X軸從左到右；Y軸由上到下，如右圖所示：



**GOTOP** 用 Visual Basic 您也能開發 Android App

B4A專案：Lesson5\基本繪圖功能\基本繪圖功能  
.b4a

- **Canvas**物件提供基本繪圖方法，可以讓我們繪出直線、長方形、圓形、橢圓形和文字內容。



**GOTOP** 用 Visual Basic 您也能開發 Android App

繪出直線

- **Canvas**物件是呼叫**DrawLine()**方法繪出直線，如下所示：  
`cvsMain.DrawLine(20dip, 20dip, 160dip, 20dip, Colors.Red, 3dip)`
- 方法的前2個參數是第1個點(x1, y1)座標，第3和4個參數是第2個點(x2, y2)座標，第5個參數是色彩，最後1個參數是線寬度。
- **DrawLine()**方法可以使用第5個參數的色彩和第6個參數的線條寬度，從第1個點座標繪出一條直線至第2個點座標。

## 繪出長方形

- Canvas物件繪出長方形方法是**DrawRect()**方法，不過，它需要建立Rect物件定義長方形尺寸，如下所示：  
`Dim rect1 As Rect`  
`rect1.Initialize(20dip, 40dip, 150dip, 100dip)`
- **Initialize()**方法的前2個參數是長方形左上角座標(x1, y1)，即Rect物件的Left和Top屬性，後2個座標是右下角座標(x2, y2)，即Right和Bottom屬性。繪出長方形如下所示：  
`cvsMain.DrawRect(rect1, Colors.Red, False, 3dip)`
- 方法的第1個參數是長方形尺寸的Rect物件，第2個參數是色彩，第3個參數決定是否填滿，值True是填滿；False只繪出邊框，最後1個參數是線條寬度。

## 繪出圓形

- Canvas物件是使用**DrawCircle()**方法繪出圓形，如下所示：  
`cvsMain.DrawCircle(220dip, 70dip, 30dip, Colors.Red, False, 3dip)`
- 方法的前2個參數是圓心(x, y)座標，第3個參數是半徑，第4個參數是色彩，第5個參數決定是否填滿，值True是填滿；False只繪出邊框，最後1個參數是線條寬度。

## 繪出橢圓形

- **Canvas**物件繪出橢圓形方法是**DrawOval()**方法，我們是在**Rect**物件的範圍內繪出橢圓形，如下所示：

```
Dim rect2 As Rect  
rect2.Initialize(20dip, 120dip, 150dip, 190dip)
```

- 建立**Rect**物件**rect2**後，就可以繪出橢圓形，如下所示：

```
cvsMain.DrawOval(rect2, Colors.Red, False, 5dip)
```

- 方法的第1個參數是橢圓形尺寸的**Rect**物件，第2個參數是色彩，第3個參數決定是否填滿，值**True**是填滿；**False**只繪出邊框，最後1個參數是線條寬度。

## 繪出文字內容

- **Canvas**物件是使用**DrawText()**方法在畫布繪出文字內容，如下所示：

```
cvsMain.DrawText("B4A繪圖", 190dip, 150dip,  
Typeface.DEFAULT, _  
20, Colors.Red, "LEFT")
```

- 方法的第1個參數是繪出的字串，第2和3個參數是繪出位置的參考座標(x, y)，因為繪出座標會因文字對齊方式而不同，第4個參數是字型，以此例為預設字型，第5個參數是文字尺寸，第6個參數是文字色彩，最後1個參數是文字對齊方式，可以是："LEFT"、"CENTER"和"RIGHT"。



B4A專案：繪出圖檔\繪出圖檔\繪出圖檔.b4a

### 載入圖檔

- 我們可以使用Canvas物件的方法來載入圖檔、並且繪出剪裁、旋轉和翻轉圖片效果，不只如此，我們還可以將Canvas畫布繪出的圖形儲存成圖檔。
- 首先載入圖檔建立Bitmap物件Bmp，如下所示：

```
Dim Bmp As Bitmap  
Bmp.Initialize(File.DirAssets,  
"Cow.jpg")
```



### 繪出圖檔

- Canvas物件是使用DrawBitmap()方法繪出圖檔的Bitmap物件，如下所示：

```
Dim DestRect As Rect  
DestRect.Initialize(20dip, 10dip, 20dip + 100dip, 10dip + 100dip)  
cvsGraph.DrawBitmap(Bmp, Null, DestRect)
```
- 方法使用Rect物件定義繪出區域，第1個參數是繪出的Bitmap物件，第2個參數是來源區域的Rect物件，Null表示全部，第3個參數是實際顯示區域的目標Rect物件，如果尺寸比原圖大，就是放大；反之是縮小。

## 繪出剪裁的圖檔

- DrawBitmap()方法的參數有來源和目的Rect物件，如果來源Rect物件只有原圖的部分區域，就是剪裁圖檔。首先定義剪裁的Rect物件，如下所示：

```
Dim SrcRect As Rect  
SrcRect.Initialize(0, 0, Bmp.Width / 2, Bmp.Height)
```

- SrcRect物件尺寸寬度只有原來一半，繪出剪裁圖片，如下所示：

```
DestRect.Left = 150dip  
DestRect.Right = 150dip + 100dip  
cvsGraph.DrawBitmap(Bmp, SrcRect, DestRect)
```

- 程式碼直接更改目標DestRect區域物件的Left和Right屬性值後，呼叫DrawBitmap()方法繪出剪裁圖片，因為有指定來源的SrcRect物件。

## 繪出旋轉的圖檔

- 如同第11-2-3節的方法，DrawBitmap()方法也有繪出旋轉圖形的DrawBitmapRotated()方法，例如：順時鐘方向旋轉70度，如下所示：

```
Dim Rect1 As Rect  
Rect1.Initialize(20dip, 200dip, 20dip + 100dip, 200dip +  
100dip)  
cvsGraph.DrawBitmapRotated(Bmp, Null, Rect1, 70)
```

- 方法的前3個參數和DrawBitmap()方法相同，最後1個參數是旋轉角度（順時鐘方向）。

## 繪出翻轉的圖檔

- 除了旋轉圖檔，Canvas還提供繪出翻轉圖檔的 **DrawBitmapFlipped()** 方法，例如：水平翻轉圖檔，如下所示：  

```
Rect1.Left = 150dip  
Rect1.Right = 150dip + 100dip  
cvsGraph.DrawBitmapFlipped(Bmp, Null, Rect1, False,  
    True)
```
- 方法的前3個參數和DrawBitmap()方法相同，第4個參數是垂直翻轉，值True是翻轉；False為沒有，最後1個參數是水平翻轉，值True是翻轉；False為沒有。

## 將Canvas畫布的圖形儲存成檔案

- 對於Canvas畫布的Bitmap物件，我們可以將它儲存成圖檔，如下所示：  

```
Dim Out As OutputStream  
Out = File.OpenOutput(File.DirRootExternal,  
    "Graph.png", False)  
cvsGraph.Bitmap.WriteToStream(Out, 100, "PNG")  
Out.Close
```
- 程式碼宣告OutputStream輸出串流物件後，呼叫File.OpenOutput()方法開啟輸出串流的Graph.png圖檔，然後呼叫Bitmap物件的WriteToStream()方法將Bitmap物件寫入串流，第1個參數是輸出串流，第2個參數是品質0~100，最後1個參數是圖檔格式"PNG"或"JPEG"。

**GOTOP** 用 Visual Basic 您也能開發 Android App

**B4A專案：Lesson5ImageView元件ImageView元件的使用.b4a**

### 建立ImageView元件

- ImageView元件可以在「Designer」視窗執行「Add View>ImageView」指令來新增，或自行使用程式碼來建立，如下所示：
 

```
Sub Globals
    Dim ImageView1 As ImageView
End Sub

...
ImageView1.Initialize("ImageView1")
```
- 程式碼使用Initialize()方法初始物件，參數是事件名稱。



**GOTOP** 用 Visual Basic 您也能開發 Android App

**B4A專案：Lesson5ImageView元件ImageView元件的使用.b4a**

### 指定ImageView元件顯示的圖片

- 然後使用Bitmap屬性指定顯示的圖片，如下所示：
 

```
igvCow.Bitmap = LoadBitmapSample(File.DirAssets, "Cow.jpg", 100, 100)
```
- 使用LoadBitmapSample()方法（此為關鍵字）載入專案的Cow.jpg圖檔，最後2個參數是圖片尺寸的最大寬度與最大高度，如果圖檔尺寸比較大，就是建立縮圖。
- 如果需要載入全尺寸圖檔，請使用LoadBitmap()方法（此為關鍵字），如下所示：
 

```
igvOutput.Bitmap = LoadBitmap(File.DirAssets, "Snake.jpg")
```

## Android的檔案

- 一般來說，大部分應用程式都需要儲存一些應用程式所需的資料，即**存取永久儲存體 (Persistent Storage)**的儲存裝置來寫入與讀取資料，我們最常使用的儲存種類有：檔案和資料庫。
- 在討論Android檔案時，我們需要先了解Android檔案的命名規則和可以在哪些位置儲存檔案。

## Android的檔案名稱

- Android作業系統的檔案基本命名規則，如下所示：
  - 檔案名稱可以使用a-z、A-Z、0-9、「\_」底線等字元，或使用「+」、「-」、「%」和「&」字元。
  - 檔案名稱不允許空白、「\*」和「?」字元。
  - 名稱區分英文字母大小寫，MyFile.txt和myfile.txt是不同檔案。

## Android檔案的儲存位置

- **File.DirAssets**：此目錄是位在Android App的.apk檔案中，隨著APK檔一起安裝至Android作業系統，我們是在B4A專案的【Files】標籤加入此目錄的檔案，這些檔案是唯讀，不允許寫入資料。
- **File.DirInternal/File.DirInternalCache**：這兩個目錄是儲存在裝置主記憶體的應用程式專屬檔案，其他Android App並無法存取這些檔案，如果Android作業系統需要更多記憶體空間時，位在暫存目錄的檔案會被清除。
- **File.DirRootExternal**：儲存在記憶卡根目錄，通常是指SD卡。
- **File.DirDefaultExternal**：這是儲存在SD卡的應用程式預設目錄，其格式為：`<記憶卡>/Android/data/<套件>/files/`，此目錄如果有需要，就會自動建立，而且其他應用程式也可以存取這些檔案。

## File預建物件-儲存位置與檢查檔案是否存在

- File預建物件提供相關屬性與方法來檢查外部儲存媒體是否可讀或可寫，檔案是否存在，其說明如下表所示：

屬性與方法	說明
File.ExternalReadable	此屬性可以檢查外部儲存媒體是否可讀取，值True為可讀；False為不可
File.ExternalWritable	此屬性可以檢查外部儲存媒體是否可寫入，值True為可寫；False為不可
File.Exists(Dir, FileName)	檢查參數的檔案是否存在，第1個參數是上一節的儲存位置，第2個參數是檔案名稱，傳回True表示存在；False為不存在

**GOTOP** 用 Visual Basic 您也能開發 Android App

## File預建物件-文字檔案讀寫

- **File**預建物件關於文字檔案讀寫的相關方法說明（存取的檔案是位在第1個參數Dir目錄，檔案名稱是第2個參數），如下表所示：

方法	說明
File.WriteString(Dir, FileName, Text)	將參數Text字串寫入參數檔案
File.ReadString(Dir, FileName)	讀取參數檔案傳回檔案內容字串
File.WriteList(Dir, FileName, List)	將參數List物件寫入參數檔案
File.ReadList(Dir, FileName)	讀取參數檔案傳回檔案內容的List物件
File.WriteMap(Dir, FileName, Map)	將參數Map物件寫入參數檔案
File.ReadMap(Dir, FileName)	讀取參數檔案傳回檔案內容的Map物件

**GOTOP** 用 Visual Basic 您也能開發 Android App

B4A專案：Lesson5\行動Memo\行動Memo(II).b4a

## WriteString()方法寫入字串

- 對於文字檔案，我們可以使用**WriteString()**方法將字串寫入檔案，如下所示：  

```
File.WriteString(File.DirInternal, "note" & slot & ".txt", content)
```
- 方法的第1個參數是Android儲存位置的目錄，第2個參數是檔案名稱，可以將最後1個參數的content字串寫入檔案。

**GOTOP**

用 Visual Basic 您也能開發 Android App

B4A專案：Lesson5\行動Memo\行動Memo(II).b4a

## ReadString()方法讀取字串

- 讀取檔案內容是使用**ReadString()**方法，如下所示：  
`content = File.ReadString(File.DirInternal, "note" & cslot & ".txt")`
- `content`變數是取得的檔案內容，方法第1個參數是第10-1-1節儲存位置的目錄，第2個參數是檔案名稱。
- 在讀取前，我們可以呼叫**File.Exists()**方法檢查檔案是否存在，如下所示：

```
If File.Exists(File.DirInternal,"note" & cslot & ".txt") Then  
...  
End If
```

**GOTOP**

用 Visual Basic 您也能開發 Android App

B4A專案：Lesson5\讀寫List物件\學生課程註冊.b4a

## 建立寫入檔案的List物件

- 對於程式使用的**List**物件（清單資料），我們可以直接讀寫**List**物件來處理文字檔案，使用的是**WriteList()**與**ReadList()**方法。
- 例如：將學生註冊課程姓名清單的**List**物件寫入文字檔案。首先建立**List**物件，如下所示：

```
Dim Students As List  
Students.Initialize()  
Students.Add(edtName.Text)
```

- 程式碼宣告和初始**List**物件後，新增一位學生資料至**List**物件。



**GOTOP**

用 Visual Basic 您也能開發 Android App

B4A專案：Lesson5\讀寫List物件\學生課程註冊.b4a

### WriteList()方法寫入List物件

- 然後將List物件寫入學生註冊資料的檔案，如下所示：

```
File.WriteList(File.DirRootExternal, myFile,  
Students)
```

- 方法的第1個參數是Android儲存位置的目錄，第2個參數是檔案名稱，可以將最後1個參數的List物件Students寫入檔案。

**GOTOP**

用 Visual Basic 您也能開發 Android App

B4A專案：Lesson5\讀寫List物件\學生課程註冊.b4a

### ReadList()方法讀取List物件

- 讀取檔案是使用ReadList()方法，如下所示：

```
Students = File.ReadList(File.DirRootExternal,  
myFile)
```

- 程式碼可以取得檔案內容的List物件Students，方法的第1個參數是Android儲存位置的目錄，第2個參數是檔案名稱。

GOTOP

用 Visual Basic 您也能開發 Android App

B4A專案：Lesson5\讀寫List物件\學生課程註冊.b4a

### 將讀取List物件建立ListView元件的項目

- 在讀取List物件後，就可以將List物件元素新增成為ListView元件的項目，如下所示：

```
ListView1.Clear()  
For i = 0 To Students.Size - 1  
    ListView1.AddSingleLine(Students.Get(i))  
Next
```

- 程式碼呼叫Clear()方法清除項目資料後，使用For/Next迴圈走訪List物件的每一個元素，和呼叫AddSingleLine()方法加入成為ListView元件的清單項目。

GOTOP

用 Visual Basic 您也能開發 Android App

B4A專案：Lesson\讀寫Map物件\我的最愛.b4a

### 建立寫入檔案的Map物件

- 對於程式使用的Map物件（鍵值資料），如同List物件，我們也可以直接讀寫Map物件來處理文字檔案，使用的是WriteMap()與ReadMap()方法。
- 例如：將我的最愛書籤的網站名稱（鍵）和URL網址（值）建立成Map物件，和寫入文字檔案。首先建立Map物件，如下所示：

```
Dim items As Map  
items.Initialize()  
items.Put(edtTitle.Text, edtURL.Text)
```

- 程式碼宣告和初始Map物件後，新增一個書籤的網站名稱和URL網址資料至Map物件。

### WriteMap()方法寫入Map物件

- 然後寫入Map物件至檔案，如下所示：  
`File.WriteMap(File.DirRootExternal, "mapItems.txt", items)`
- 方法的第1個參數是Android儲存位置的目錄，第2個參數是檔案名稱，可以將最後1個參數的Map物件items寫入檔案。

### ReadMap()方法讀取Map物件

- 讀取檔案是使用ReadMap()方法，如下所示：  
`items = File.ReadMap(File.DirRootExternal, "mapItems.txt")`
- 程式碼可以取得檔案內容Map物件items，方法的第1個參數是Android儲存位置的目錄，第2個參數是檔案名稱。

**GOTOP** 用 Visual Basic 您也能開發 Android App

**B4A專案：Lesson\讀寫Map物件\我的最愛.b4a**

### 將讀取的Map物件建立ListView元件項目

- 在讀取Map物件後，就可以將Map物件元素新增成為ListView元件的項目，如下所示：

```
ListView1.Clear()
For i = 0 To items.Size - 1
    ListView1.AddSingleLine2(items.GetKeyAt(i),
    items.GetValueAt(i))
Next
```
- 程式碼呼叫Clear()方法清除項目資料後，使用For/Next迴圈走訪Map物件的每一個元素，和呼叫AddSingleLine2()方法加入成為ListView元件的清單項目，包含項目名稱和值。

**GOTOP** 用 Visual Basic 您也能開發 Android App

 **Basic4android的線上教學資源**

<http://www.basic4ppc.com/android/documentation.html>

**Anywhere Software** Basic4Android

Home | Screenshots | Showcase | Features | Download | Purchase | Online Community | Documentation | About Us | search

**Basic4Android Beginner's Guide (v2.6) and Code snippets: (Download)**  
**Basic4android User's Guide (v1.3): (Download)**  
Try the search feature in the upper right corner. It works!

Internal keywords can be found in the Core library documentation under the Keywords type.

German: Folgende Tutorials und Libraries sind auch in Deutscher Sprache verfügbar: German Tutorials, Keywords, Drawings, Files, Views, GPS, HTTP, Dialogs, Collections, XMLSax, Serial, Network, JSON, TTS, MediaPlayer, RegularExpressions, StrangUnits, Camera, Jpeg, Phone, SQL.  
French: Quelques tutoriels sont disponibles en français: French Tutorials  
Chinese: 中文教程請參閱: Chinese Tutorials  
Russian: Уроки на русском: Russian Tutorials  
Thai: บทเรียนภาษาไทย: Thai Tutorials

**Tutorials and examples:**

- IDE tips (+video tutorial)
- Getting started with remote compilation (video tutorial)
- Setting the emulator and Hello world application
- Debugging applications

 Basic4android book

**GOTOP** 用 Visual Basic 您也能開發 Android App

### 意圖與意圖篩選

- Android應用程式送出意圖的訊息需要經過Android作業系統來判斷接收者是誰，它是使用意圖篩選 (Intent Filters) 找出有能力處理的活動或內建應用程式，然後才將訊息送給接收者，如右圖所示：

意圖 (Intents) 是一個啟動其他 Android 活動、服務和廣播接收器的系統訊息，可以告訴 Android 作業系統我想作什麼？

**GOTOP** 用 Visual Basic 您也能開發 Android App

專案: Lesson6\建立多活動\建立多活動App.b4a

### 建立多活動App-

#### 步驟一：建立Main主活動

- 建立多活動App的第一步是建立B4A專案和輸入程式碼，如下所示：

```

Sub Globals
    Dim Button1 As Button
End Sub

Sub Activity_Create(FirstTime As Boolean)
    Activity.Title = "建立多活動App"
    ...
End Sub

Sub Button1_Click
    StartActivity("Second")
End Sub
    
```

啟動Second活動

**GOTOP** 用 Visual Basic 您也能開發 Android App

專案:Lesson6\建立多活動\建立多活動App.b4a

### 建立多活動App-

#### 步驟二：新增Second活動模組

- B4A專案擁有2個活動模組，我們需要新增名為Second的第2個活動模組，如下所示：

```
Sub Globals
    Dim Button1 As Button
End Sub

Sub Activity_Create(FirstTime As Boolean)
    Activity.Title = "第二個活動"
    ...
End Sub

Sub Button1_Click
    StartActivity("Main")
End Sub
```

啟動Main活動

**GOTOP** 用 Visual Basic 您也能開發 Android App

專案:Lesson6\建立多活動\建立多活動App.b4a

### 建立多活動App-

#### 步驟三：編譯和執行多活動App

- 現在，我們已經完成2個活動的Android App，其編譯和執行結果如下圖所示：



**GOTOP** 用 Visual Basic 您也能開發 Android App  
專案:Lesson6\傳遞資料至其他活動\我的最愛.b4a

### 傳遞資料至其他活動-B4A的作法

- 程式是使用ListView元件顯示網站名稱清單，點選項目，就可以啟動第2個活動，和在WebView元件顯示選取網站的首頁，如下圖所示：



**GOTOP** 用 Visual Basic 您也能開發 Android App  
專案:Lesson6\傳遞資料至其他活動\我的最愛.b4a

### 傳遞資料至其他活動-Process\_Globals程序

- 資料是從Main傳遞至WebView活動，所以是在Main活動模組宣告傳遞資料的變數，如下所示：  
**Sub Process\_Globals**  
**Dim URL As String**  
**End Sub**
- 程序宣告名為URL的行程變數。在第2個WebVew活動可以使用活動名稱加上變數名稱Main.URL取得Main活動的行程變數值，如下所示：  
**Dim URL As String**  
**URL = Main.URL**

## 意圖的動作類型(準備作什麼)

- B4A在呼叫Initialize()方法初始Intent意圖物件時，可以在第1個參數指定使用的動作類型，第2個參數是URI。活動常用的動作類型說明，如下表所示：

動作類型	說明
ACTION_VIEW	顯示資料給使用者檢視
ACTION_EDIT	顯示資料給使用者編輯
ACTION_CALL	使用自動撥號打電話
ACTION_SEND	寄送簡訊或電子郵件
ACTION_PICK	從資料來源選取一個項目資料
ACTION_APPWIDGET_UPDATE	更新應用程式小工具 (App Widget)，詳見第13-3節的說明
ACTION_MAIN	啟動如同是程式進入點的主程式

## 意圖的URI(操作所需的資料)

- 萬用資源識別URI是用來定位Android系統的資源，幫助Intent意圖物件的動作可以取得或找到操作的資料。Android常用的URI說明，如下所示：
  - URL網址：URI可以直接使用URL網址，如下所示：  
<http://www.google.com.tw/>
  - 地圖位置：GPS定位的座標值 (GeoPoint格式)，如下所示：  
<geo:25.04692437135412,121.5161783959678>
  - 電話號碼：指定撥打的電話號碼，如下所示：  
<tel:123456789>
  - 寄送簡訊：寄送簡訊給指定的電話號碼，如下所示：  
<sms:123456789>



## 使用隱含意圖啟動瀏覽器

- 在活動啟動內建瀏覽器是建立ACTION\_VIEW動作，和URI為URL網址的Intent物件，如下所示：  
Dim i As Intent  
i.Initialize(i.ACTION\_VIEW, "http://www.google.com.tw")  
StartActivity(i)
- 程式碼宣告Intent物件變數後，呼叫Initialize()方法初始Intent物件，第1個參數是動作，第2個參數是URI字串，最後呼叫StartActivity()方法啟動活動，參數是Intent物件。

## 使用隱含意圖啟動打電話程式

- 在活動啟動內建撥號程式是使用ACTION\_VIEW動作，URI為電話號碼，如下所示：  
Dim i As Intent  
i.Initialize(i.ACTION\_VIEW, "tel:123456789")  
StartActivity(i)

## 使用隱含意圖啟動程式傳送簡訊

- 在活動可以啟動內建簡訊工具來傳送簡訊，它是使用ACTION\_VIEW動作，URI為簡訊收件者的電話號碼，如下所示：

```
Dim i As Intent
```

```
i.Initialize(i.ACTION_VIEW, "sms:123456789")
```

```
i.PutExtra("sms_body", "這是一封測試簡訊")
```

```
StartActivity(i)
```

- 程式碼呼叫Intent物件PutExtra()方法來附加資料，第1個參數是資料名稱，第2個參數是資料值，以此例是指定簡訊內容。

## 使用明確意圖啟動App的其他活動

- 明確意圖需要在StartActivity()方法參數的Intent意圖物件指明目標活動的活動類別，或進一步指明元件名稱來啟動指定的Android App，而不是讓Android作業系統依據動作類別和URI來選擇適合的Android App。
- 使用明確意圖啟動其他活動需要使用SetComponent()方法指定目標的元件名稱，如下所示：

```
Dim i As Intent
```

```
i.Initialize("", "")
```

```
i.SetComponent("com.android.calculator2/.Calculator")
```

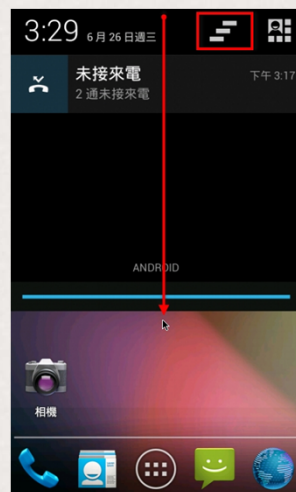
```
StartActivity(i)
```

## 使用明確意圖啟動Google地圖

- 在活動啟動內建Google地圖是使用ACTION\_VIEW動作，URI為GPS座標值，不過稍有不同，如下所示：  
`Dim i As Intent`  
`Dim lat, lon As String`  
`lat = "25.04692437135412"`  
`lon = "121.5161783959678"`  
`i.Initialize(i.ACTION_VIEW, "geo:" & lat & "," & lon & "?q=" & lat & "," & lon)`  
`i.SetComponent("googlemaps")`  
`StartActivity(i)`
- URI字串的最後重複加上q參數的GPS座標值，以便在Google地圖加上標記，呼叫SetComponent()方法指明元件名稱是Google地圖。

## 狀態列通知的基礎

- 狀態列 (Status Bar) 是行動裝置最上方的一條橫向長條區域，平板電腦Android 3.x是位在最下方，Android 4.x版作業系統的狀態列是位在最上方。
- 「通知服務」 (Notification Service) 是一種系統服務，可以在狀態列顯示可向下捲動的通知訊息，例如：未接來電，如右圖所示：



## 建立Notification物件的狀態列通知

- Notification物件建立狀態列通知十分簡單，不過，請注意！我們一定需要指定圖示和訊息內容，才能正確顯示通知。首先宣告Notification物件變數，如下所示：

```
Dim n As Notification
```

```
n.Initialize()
```

- 程式碼呼叫Initialize()方法初始Notification物件。

## 指定狀態列通知的圖示

- 然後使用Icon屬性指定顯示的圖示，如下所示：

```
n.Icon = "w1"
```

- Icon屬性值是圖示檔名稱，不含副檔名且區分英文字母大小寫，我們需要自行將圖示檔複製到B4A專案的「\Objects\res\drawable」目錄，和將屬性值改為唯讀，否則當編譯時，B4A會自動刪除圖示檔，我們也可以指定成“icon”，即App本身的應用程式圖示，如下所示：

```
n.Icon = "icon"
```

## 指定狀態列通知的通知訊息

- 接著指定通知訊息，和點選通知開啟的活動，如下所示：

```
n.SetInfo("BMI計算機", "BMI值太高, 體重稍重...", BMI)
```

- **SetInfo()**方法的第1個參數是標題文字，第2個參數是訊息內容，最後1個參數是點選通知開啟的活動物件名稱，如果程式碼位在Main模組且點選開啟的也是Main活動，最後1個參數值就是空字串""。

## 顯示狀態列通知

- 現在，我們已經指定圖示和通知訊息，最後就可以呼叫**Notify()**方法顯示通知，如下所示：

```
n.Notify(1)
```

- 方法的參數是整數的通知編號，此編號就是用來在之後更新或取消通知。

## 更新狀態列通知

- 對於已經顯示的通知，我們可以更新圖示和通知訊息文字後，再送出一次，如下所示：

```
n.Icon = "w2" ————— 更改圖示  
n.SetInfo("BMI計算機", "BMI值太高, 體重過重...",  
BMI)  
n.Notify(1)
```

- 通知更新圖示和訊息文字後，因為Notify()方法的參數值是同一個通知編號，換句話說，就是顯示更新的狀態列通知。

## 取消狀態列通知

- 當點選狀態列通知開啟目標活動時，我們可以在此活動的Activity\_Resume程序取消此通知，如下所示：

```
Sub Activity_Resume  
Dim n As Notification  
n.Initialize()  
n.Cancel(1) ————— 取消通知  
End Sub
```

- 程序建立Notification物件後，呼叫Cancel()方法取消通知，參數是通知編號。

**GOTOP** 用 Visual Basic 您也能開發 Android App

專案:Lesson6\從狀態列通知傳遞資料\四則計算機.b4a

### 從狀態列通知傳遞資料至活動- 傳遞字串資料

- 在建立狀態列通知的訊息內容時，可以額外傳遞一個字串至啟動活動，使用的是**SetInfo2()**方法，如下所示：
 

```
Dim n As Notification
n.Initialize()
n.Icon = "icon"
n.AutoCancel = True
n.SetInfo2("四則計算機", "除以0的錯誤...", "第2個運算元不能為0", Error)
n.Notify(1)
```

自動取消通知

傳遞的字串
- 圖示是應用程式圖示，指定**AutoCancel**屬性值為**True**，表示點選通知就自動取消通知，在**SetInfo2()**方法的第3個參數是傳遞字串，可以傳遞至最後一個參數的**Error**活動。

**GOTOP** 用 Visual Basic 您也能開發 Android App

專案:Lesson6\從狀態列通知傳遞資料\四則計算機.b4a

### 從狀態列通知傳遞資料至活動- 取得傳遞字串

- 在**Activity\_Resume**程序取得狀態列通知傳遞字串，如下所示：
 

```
Sub Activity_Resume
Dim i As Intent
i = Activity.GetStartingIntent()
If i.HasExtra("Notification_Tag") Then
    lblOutput.Text = i.GetExtra("Notification_Tag")
End If
End Sub
```

取得啟動活動的 Intent 物件

是否有參數
- 呼叫**Activity.GetStartingIntent()**方法取得啟動此活動的**Intent**物件，**If**條件呼叫**HasExtra()**方法，參數是資料名稱**"Notification\_Tag"**，如果存在，就呼叫**GetExtra()**方法取出傳遞字串。

## 服務

- 服務是在背景執行的行程，可以執行和活動一樣的工作，只是沒有使用介面。
- 例如：在播放背景音樂時，之所以不會打斷我們發送簡訊或收發電子郵件，因為它是在背景執行的服務，才能讓音樂播放不中斷。

## 服務的生命周期

- 服務的生命周期幾乎不會被目前存在的活動影響，因為它是在背景執行工作，所以沒有任何活動的使用介面，也不能使用對話方塊（**Toast**彈跳訊息框除外），通常都是使用狀態列通知，作為與使用者互動的管道。
- 服務不會暫停和重新啟動，當使用者旋轉螢幕，也不會重新建立服務（只會重新建立活動），因此，在活動模組撰寫的程式碼，一旦活動進入背景，程式也隨之暫停，但是，服務就可以持續執行，換句話說，對於哪些沒有使用介面的任務與工作，使用服務來實作比起活動還來的簡單。



## 服務模組的範本程式碼

- 服務模組範本程式碼的程序是服務的生命周期，它是作業系統依序呼叫的一些程序，如下所示：

```
Sub Process_Globals
End Sub

Sub Service_Create
End Sub

Sub Service_Start (StartingIntent As Intent)
End Sub

Sub Service_Destroy
End Sub
```

## 服務模組的程序說明-1

- **Process\_Globals**程序：在此程序宣告服務模組的全域變數（即行程變數），服務模組沒有**Globals**程序，因為服務不支援活動物件，我們在服務模組宣告的全域變數，相當於是在活動模組宣告行程變數，可以讓其他模組存取。
- **Service\_Create**程序：當服務第1次啟動時就會呼叫此程序，我們可以在此程序初始行程變數，當服務啟動後，它會一直存在直到活動模組呼叫**StopService()**方法停止服務，或整個行程被刪除。

## 服務模組的程序說明-2

- **Service\_Start**程序：當每一次呼叫StartService()或StartServiceAt()方法時就會呼叫此程序，將App的行程移至前景狀態，作業系統不會刪除此行程，直到程序結束執行為止。如果是排程工作，定時在間隔時間執行的工作，請在此程序呼叫StartServiceAt()方法建立排程執行的服務。
- **Service\_Destroy**程序：當在活動呼叫StopService()方法時，就是呼叫此程序來終止服務且停止服務的執行，我們需要再次呼叫StartService()方法來啟動服務，然後依序呼叫Service\_Create()和Service\_Start()方法來執行服務。

## Service\_Start()程序

- Service\_Start()程序使用StartServiceAt()方法排程自動在下一秒啟動服務，方法第1個參數是服務模組名稱，空字串是目前服務，第2個參數指定間隔時間是目前時間的下一秒，最後1個參數是當裝置如果在睡眠時是否啟動服務，True為是；False要等到裝置恢復才啟動，如下：

```
Sub Service_Start(StartingIntent As Intent)
```

```
    Counter = Counter + 1
```

```
    If isRun Then
```

```
        StartServiceAt("", DateTime.Now +  
        DateTime.TicksPerSecond, True)
```

```
    End If
```

```
End Sub
```

排程啟動服務

## 在服務模組Process\_Globals使用行程變數

- Counter和IsRun變數都是全域的行程變數，在活動模組是使用Timer物件來定時取得Counter變數值，IsRun變數是用來控制服務是否執行。

### Sub Process\_Globals

```
Dim Counter As Int = 0
```

```
Dim isRun As Boolean
```

```
End Sub
```

行程變數

## Service\_Create()程序

- 在TimeService服務模組的Service\_Create()程序建立狀態列通知，可以讓使用者透過通知訊息來開啟可以關閉服務的活動，如下所示：

### Sub Service\_Create

```
Dim n As Notification
```

```
n.Initialize()
```

```
n.Icon = "icon"
```

```
n.AutoCancel = True
```

```
n.SetInfo("計數器服務", "停止計數器服務...", Main)
```

```
n.Notify(1)
```

```
End Sub
```

建立與初始  
Notification物件

指定圖示

指定訊息

**GOTOP** 用 Visual Basic 您也能開發 Android App

### 在活動模組Main建立Timer物件

- **Timer**物件可以在指定間隔時間產生**Tick**事件，我們可以使用**Timer**物件取代很長次數的迴圈和實作動畫效果。

```
Sub Process_Globals
    Dim Timer1 As Timer
    Timer1.Initialize("Timer", 1000)
End Sub
...
Sub Timer_Tick
    lblOutput.Text = TimeService.Counter
End Sub
```

建立與初始Timer物件

顯示計數

**GOTOP** 用 Visual Basic 您也能開發 Android App

### 在活動模組Main啟動和停止服務

```
Sub Button1_Click
    Timer1.Enabled = True
    TimeService.isRun = True
    StartService(TimeService)
End Sub

Sub Button2_Click
    Timer1.Enabled = False
    TimeService.isRun = False
    CancelScheduledService(TimeService)
    StopService(TimeService)
End Sub
```

啟動計時器物件

啟動服務

取消排程啟動服務

停止服務

