

Ch3 Visual Basic 資料型態與運算

本章將會介紹：

- 基本資料型態
- 變數宣告
- 變數的活動範圍
- 算術運算子與運算式
- 基本指令敘述

復習：

復習前一章的內容。前一章我們初步地 view 了所謂的「物件」及「事件導向程式」。整理一下大致如下：

當一專案(程式)啟動時，會先啟動「啟動物件」(ex:Form1)。在讀入 Form1 時，會觸發 Form1_Load () 事件並執行預先設計好的程式。若按下 Command Button(ex: cmd_1)，會觸發 cmd_1_click () 事件並執行預先設計好的程式。

結論：

- 我們可以稱每觸發一事件都會執行一段「程序(式)」。
- 一個表單模組(Form Module)下可以有許多程序。

基本資料型態

Visual Basic 語言的基本資料型態有如下列數種。對於不同的資料如 10 (整數) “逢甲”(字串) 12/31/96 (時間) 36.67 (單精度) 454747858 (長整數) 4.8x10e3 (雙精度)有不同的記憶空間與範圍。

以下表格顯示所支援的資料型態，以及儲存空間大小與範圍。

(截自 MSDN Library)

| 資料型態 | 儲存空間大小 | 範圍 |
|-------------------------------|--------|---|
| Byte | 1 個位元組 | 0 到 255 可用於儲存 2 進位的資料 |
| Boolean | 2 個位元組 | True 或 False |
| Integer | 2 個位元組 | -32,768 到 32,767 存放整數用。 |
| Long (long integer) | 4 個位元組 | -2,147,483,648 到 2,147,483,647 存放長整數用。 |
| Single (單精度浮點數) | 4 個位元組 | 負數時從 -3.402823E38 到 -1.401298E-45；正數時從 1.401298E-45 到 3.402823E38 |
| Double | 8 個位元組 | 負數時從 -1.79769313486231E308 到 |

| | | |
|----------------------------|--------------------|--|
| (雙精度浮點數) | | -4.94065645841247E-324；正數時從 4.94065645841247E-324 到 1.79769313486232E308 |
| Currency (大範圍的整數) | 8 個位元組 | 從 -922,337,203,685,477.5808 到 922,337,203,685,477.5807 一個精明確定的資料 型態,常用於貨幣計量。 |
| Decimal | 14 個位元組 | 沒有小數點時為 +/-79,228,162,514,264,337,593,543,950,335，而小 數點右邊有 28 位數時為 +/-7.9228162514264337593543950335；最小的非 零值為 +/-0.000000000000000000000001。 |
| Date | 8 個位元組 | January 1, 100 到 December 31, 9999 存放日期 資料 |
| Object | 4 個位元組 | 任何 Object 引用 |
| String (固定長度字串) | 字串長度 | 1 到大約 65,400 存放非數質的資料 |
| Variant(數值) Variant(字元) | 16 個位元組 22 個位元組 | 任何數值，最大可達 Double 的範圍。宣告後未 指定為何種資料型態。 |

Variant 特性簡介

1. 宣告變數未指定資料型態時,則預設為此型態。
2. 建立 Variant 變數在未指定初始值前,內容為空。
3. 將資料定義為 Variant,此變數可儲存系統所有資料型態,不必由程式碼來做任何的型態間轉換。
4. 系統在幫我們做資料型態轉換的同時,會消耗系統資源。故除非不必要,盡量少用之。

變數宣告

語法：Dim 變數名稱 As 資料型態

EX:

Dim A As String

Dim B As Single

Dim C As Double

Dim i,j,k As Integer

練習

預備知識：

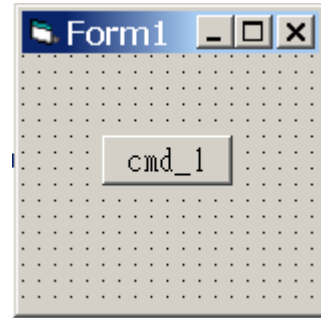
-符號「&」為串連字串之功能。

-符號「_」(底線)為程式段行之功能。當程式段過長時可以使用此功能讓程式段行以方便閱讀。

-字串要以引號(“”)引起來,否則將會認定為一變數。

建立一表單,加入 1 個 Command Button 控制項,並命名為 cmd_1,如又圖。加入如下之程式碼

```
Private Sub cmd_1_Click()  
    Dim strA As String  
    Dim strB As String  
    Dim intA As Integer  
    strA = ABC  
    strB = "ABC"  
    intA = ABC  
    MsgBox "strA=" & strA & _  
        ";intA=" & strB & _  
        ";intA=" & intA  
  
End Sub
```



觀察按下 cmd_1 後執行結果。

討論：

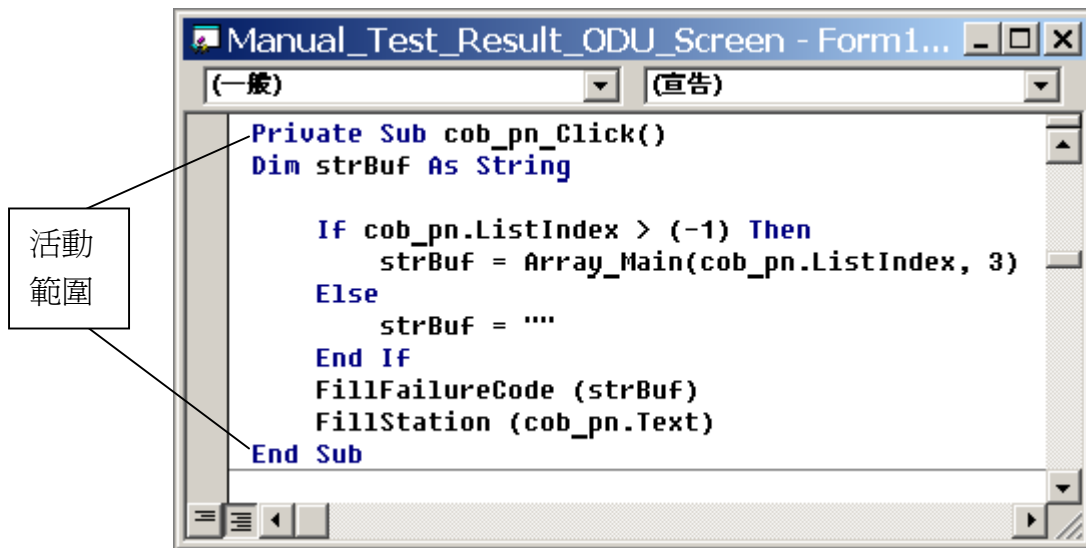
1. 當「ABC」沒有用引號引起來時,系統自動將「ABC」宣告為 Variant 資料型態之變數。起始值為空值(empty)
2. intA assign 一空值,show 出來的結果是 0。

變數的活動範圍

(1)程序層次變數

活動範圍：該變數所宣告的程序內

Ex:



此段程式中的 strBUF 的活動範圍就從 Private Sub cob_pn_Click()開始到

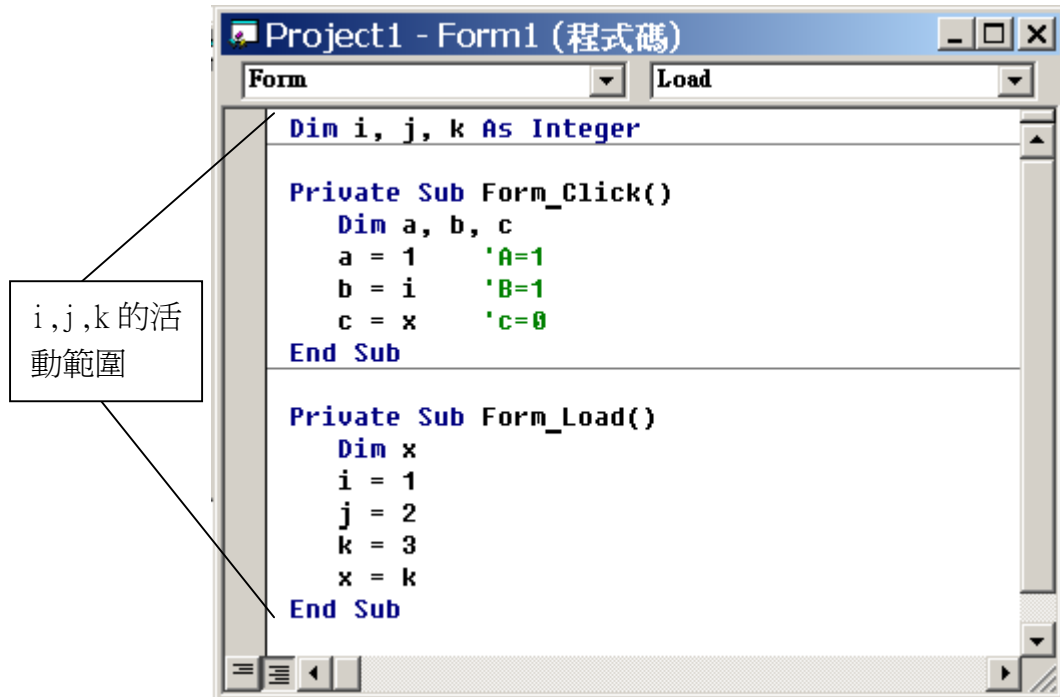
End Sub 結束。

PS: 假設希望 cob_pn_Click() 每次執行後的 strBUF 可以保留的話，可以將 Dim 改成為 Static, 亦即宣告 strBUF 為”靜態變數”。

(2) 模組層次變數:

活動範圍：宣告變數所在位置的整個模組。

EX:



練習:

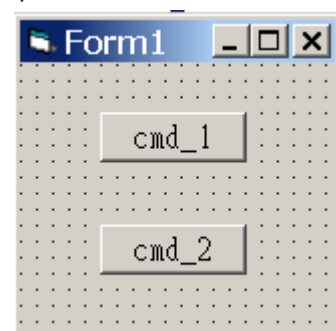
預備知識：

一表單在載入時所會發生的 event：

1. **Initialize** (當表單物件產生時引發)。
2. **Load** (每一次表單被載入記憶體時引發)。
3. **Activate** (當使用者在應用程式中切換到不同表單時引發)。
4. **GotFocus** (沒有貼任何可以獲得輸入焦點的控制項的表單，才可以收到這個事件)
5. **LostFocus** (沒有貼任何可以失去輸入焦點的控制項的表單，才可以收到這個事件)。
6. **Deactivate** (當表單失去輸入焦點時引發)。
7. **QueryUnload** (當表單要自記憶體移除之前引發)。
8. **Unload** (當表單要自記憶體移除之後引發)
9. **Terminate** (當表單物件被設定成 **Nothing** 時引發)。

常用的為：Load, Activate, Unload。

建立一表單, 加入 2 個 Command 控制項, 各命名為 cmd_1, cmd_2。如右圖



加入下列程式碼：

```
Dim i, j, k As Integer

Private Sub cmd_1_Click()
    Dim a As Integer
    a = k + i
    MsgBox a
End Sub

Private Sub cmd_2_Click()
    MsgBox "x=" & x
End Sub

Private Sub Form_Load()
    Dim x As Integer
    i = 1
    j = 2
    k = 3
    x = k
End Sub
```

觀察按下 cmd_1 和 cmd_2 後的結果。

討論：

- i 和 k 都是模組層次的變數,可以遊走於各程序之間。
- x 系為程序層次變數,在其它程序中將重新定義。

算術運算子與運算式

運算子有分成：

1. 數值運算式
2. 關係運算式(見第 4 章)
3. 邏輯運算式(見第 4 章)

數值運算式是由算術運算子和數值型態資料所組成。VB6 的算術運算子整理如下

| 運算子 | 功能 | 數學運算式 | VB 運算式 |
|-----|-------|--------------|--------|
| ^ | 指數 | X^3 | X^3 |
| + | 正號 | +A | +A |
| - | 負號 | -A | -A |
| * | 乘號 | $A \times B$ | A*B |
| / | 浮點數除號 | $A \div B$ | A/B |
| \ | 整數除號 | | A\B |

| | | | |
|-----|----|-----|---------|
| Mod | 餘數 | | A Mod B |
| + | 加號 | A+B | A+B |
| - | 減號 | A-B | A-B |

範例：

a=10 , b=4 , c=3 , d=25. 下列 VB 運算式各代表何款數學運算式？

(1) $(a^b+5)*c \rightarrow (10^4 + 5) \times 3$

(2) $a/c*d+d/a \rightarrow \frac{10}{3 \times 25} + \frac{25}{10}$

(3) $a \setminus c \rightarrow (10/25)$ 無條件捨去法 = 0

(4) $a \text{ Mod } b \rightarrow (10/4)$ 取餘數 = 2

練習：

製作一表單，上面加入 4 個 command 按鈕，各命名為 cmd_1,cmd_2,cmd_3

令 X=2.1 , Z=1.2, W=1.4, N=1.01, T=1.02

cmd_1 按下去後以 Message Box show 出 $X^2 + 3X$

cmd_2 按下去後以 Message Box show 出 $\frac{(X^2 + Z^{\frac{1}{3}})W^4}{N^{24} + T^{\frac{2}{5}}}$

cmd_3 按下去後以 Message Box show 出 X^{Z^3}

Ans：

```

Project1 - Form1 (程式碼)
Form Load
Dim X, Z, W, N, T As Double
Private Sub cmd_1_Click()
    MsgBox X ^ 2 + 3 * X
End Sub

Private Sub cmd_2_Click()
    MsgBox (X ^ 2 + Z ^ (1 / 3)) * W ^ 4 / (N ^ 24 + T ^ (2 / 5))
End Sub

Private Sub cmd_3_Click()
    MsgBox X ^ (Z ^ 3)
End Sub

Private Sub Form_Load()
    X = 2.1: Z = 1.2: W = 1.4: N = 1.01: T = 1.02
End Sub

```

基本指令敘述

VisualBasic 常用的指令敘述，如：指定敘述(=)、顯示指令(Print),以及註解指令(' or Rem)。

指令序述： =

- (1) 功能：將等後右側運算式的結果置於左側變數內。
- (2) ex: Sum = a + b

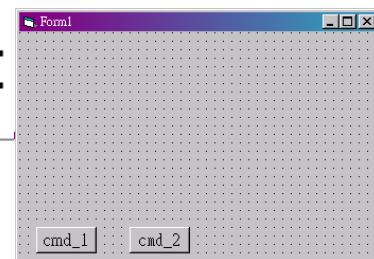
顯示敘述： Print

- (1) 語法：Print [運算式] [;|,] [運算式]
- (2) 功能：將運算式內容顯示於螢幕
- (3) ex: Print a,b 會印出 a,b 變數內容,且 a、b 間各相距一個列印區間
ex: Print "3*4=" ;3*4 會印出 3*4=12
PS: 「;」、「,」只能用於 Print 敘述的字串處理。而「&」可以用在一般字串處理及 Print 敘述裡。
- (4) Tab(n)函數：n ∈ 正整數。由目前游標所在處開始算起,游標往右移 n 個位置。
- (5) Spc(n)函數：n ∈ 正整數。在目前游標位置往後插入 n 個空白。
- (6) Cls 方法：將目前的畫面清除。

範例:

設計一表單加入 2 個 command button,如圖。並輸入如下之程式碼:

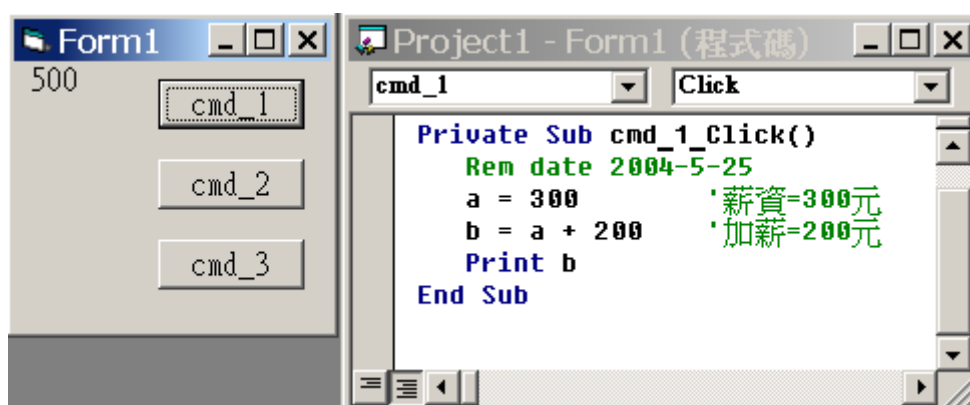
```
Private Sub cmd_1_Click()  
    Cls  
    Print "','的範例:strA", "strB"  
    Print "','的範例:strA"; "strB"  
    Print "'Tab'的範例:strA"; Tab(20); "strB"  
    Print "'Spc'的範例:strA"; Spc(20); "strB"  
End Sub  
Private Sub cmd_2_Click()  
    Print "無Cls"  
    Print  
End Sub
```



文字註解 Rem or '(單引號)

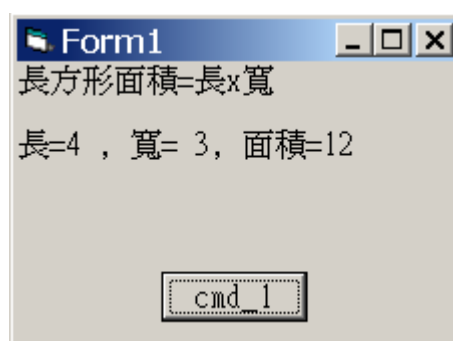
VB 提供 2 種文字註解的方法：Rem 和 '。文字註解的主要做用是提高程式的可讀性，進而讓程式易於偵錯、維護。電腦執行時，會略過註解敘述那行。

範例：



上述的程式碼以 “Rem” 來註明程式的日期，以單引號在適當的程式碼處加上說明，藉以提升程式的可讀性。

練習：請建立一個專案，畫面如右。按下 cmd_1 後會計算出長為 4、寬為 3 的長方形面積。



練習：試設計一個可以計算圓球的表面積、體積的程式。

Hint:面積： $4\pi r^2$ ；體積： $\frac{4}{3}\pi r^3$ 。其中圓週率=3.14159