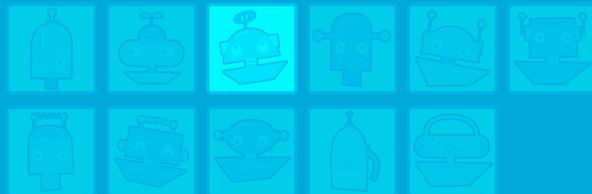


Chapter

03 基本輸出入



輸出與輸入是撰寫程式的第一步，也是所有語言的第一步，本章即是介紹 VB 的基本輸出入。



3-1 Console 類別

Console 類別的部分方法成員，如下圖所示，通常使用在主控台應用程式的輸入與輸出。由於都是公用方法，如下圖的大寫 S (Shared)，所以都是使用「類別.成員」存取這些方法。本單元將介紹 Write、WriteLine、Read 以及 ReadLine 等四個常用方法。

	名稱	描述
⇒ S	Beep()	透過主控台喇叭播放嗶聲。
⇒ S	Beep(Int32, Int32)	以指定之頻率和持續期間透過主控台喇叭播放嗶聲。
⇒ S	Clear()	清除主控台緩衝區及包含顯示資訊的對應主控台視窗。
⇒ S	MoveBufferArea(Int32, Int32, Int32, Int32, Int32, Int32)	將螢幕緩衝區的指定之來源區域複製到指定之目標區域中。
⇒ S	MoveBufferArea(Int32, Int32, Int32, Int32, Int32, Int32, Char, ConsoleColor, ConsoleColor)	將螢幕緩衝區的指定之來源區域複製到指定之目標區域中。



一、Write 方法

Write 方法共有 17 種多載，部分多載如下圖所示。

	Write(Boolean)	將指定布林值 (Boolean) 的文字表示寫入標準輸出資料流。
	Write(Char)	將指定的 Unicode 字元值寫入標準輸出資料流。
	Write(array<Char>^)	將指定的 Unicode 字元陣列寫入標準輸出資料流。
	Write(array<Char>^, Int32, Int32)	將指定的 Unicode 字元子陣列寫入標準輸出資料流。
	Write(Decimal)	將指定 Decimal 值的表示文字寫入標準輸出資料流。
	Write(Double)	將指定的雙精確度浮點數值的文字表示寫入標準輸出資料流。

本單元僅介紹兩種代表性的方法，分別是 Write(Short) 與 Write(String, Object)。

1. Overloads Public Shared Sub Write(Short)

公用方法，可將 Short 型別的資料輸出。例如：

```
Dim a As Short = 5
Console.Write(a)
```

結果是 5。

2. Overloads Public Shared Sub Write(String, Object)

此方法可將指定資料以格式化字串輸出。例如：

```
Dim a As Short = 50000
Dim b As Short = 20000
Console.Write("{0:c} {1:n}", a, b)
```

的輸出如下。

```
NT$50,000.00    20,000.00
```

其中 {0:c} 與 {1:n} 的 0 與 1 表示所要對應的資料順序（上例 a 的順序是 0，b 是 1）；c 與 n 代表輸出格式，常用的輸出格式如右表所示，且大小寫都視為相同。

符號	類別名稱
C	貨幣符號
D	十進位
E	科學記號
F	小數的位數。例如：F5 表示小數取 5 位。
N	顯示千分位
X	十六進位



二、WriteLine 方法

WriteLine 方法的使用方式均同 Write 方法，只是資料輸出之後即跳一列。例如：

```
Console.WriteLine("a")
Console.WriteLine("aa")
```

結果是

```
a
aa
```

而

```
Console.Write("a")
Console.Write("aa")
```

結果是

```
aaa
```

三、Read 方法

此方法的語法為

```
Public Shared Function Read() As Short
```

以下敘述可讀取單一字元，並印出其 ASCII Code。例如：鍵入 a，則印出 97；但若鍵入 abc，亦只印出 97。

```
Dim a As Short
a = Console.Read( )
Console.WriteLine(a)
```

若要輸出所鍵入的字元，則要使用轉型。例如：以下敘述可將所鍵入的字元，以字元的形式輸出。如鍵入 a，可輸出 a。

```
Dim a As Short
Dim b As Char
a = Console.Read( )
b = Chr(a)      'Chr 是轉型函式，轉為字元
Console.WriteLine(b)
```



四、ReadLine 方法

此方法的語法為

```
Public Shared Function ReadLine() As String
```

例如：以下敘述可讀取使用者所輸入的字串。

```
Dim s As String  
s = Console.ReadLine()  
Console.WriteLine(s)
```

輸入的東西不管是阿拉伯數字或中英文，VB 均視為字串，若輸入的東西是數值，且要進行數值運算，則要使用 Val 函數轉為數值。例如：以下敘述可將字串 s 轉為數值並加一後輸出。Val 的功能是將字串轉為數值。

```
Dim s As String  
Dim a As Short  
s = Console.ReadLine()  
a = Val(s) + 1  
Console.WriteLine(a)
```

若未先使用 Val 函數轉型而直接相加，則視為字串相加。請自行鍵入以下程式，並觀察執行結果。

```
Module Module1  
    Sub Main()  
        Dim s1 As String = "12"  
        Dim s2 As String = "34"  
        Dim s As String  
        s = s1 + s2  
        Console.WriteLine(s)  
        Dim a As Short  
        a = Val(s1) + Val(s2)  
        Console.WriteLine(a)  
    End Sub  
End Module
```

**範例 3-1a**

請寫一個程式，可以輸入矩形的長與寬，並輸出周長與面積。

**執行結果**

```
C:\WINDOWS\system32\c... - □ X
請輸入矩形的長:4
請輸入矩形的寬:5
矩形面積為:20
矩形周長為:18
請按任意鍵繼續 . . .
```

程式列印

```
Module Module1
    Sub Main()
        Dim a, b, c, d As Short
        Console.Write(" 請輸入矩形的長:")
        a = Val(Console.ReadLine())
        Console.Write(" 請輸入矩形的寬:")
        b = Val(Console.ReadLine())
        c = a * b
        d = 2 * (a + b)
        Console.Write(" 矩形面積為:")
        Console.WriteLine(c)
        Console.Write(" 矩形周長為:")
        Console.WriteLine(d)
    End Sub
End Module
```

動手做

$x=3$ 、 $y=4$ ：請寫程式計算 $x+y$ 、 $\frac{x+y}{x-y}$ 、 $\sqrt{x^2+y^2}$ 的結果。

❗ 提示：開根號的運算，除了可用次方 (^) 運算子外，亦可使用 `Math.sqrt()` 方法，請自行線上查詢 `Math`。



3-2 Form

Form（表單）如右圖所示，它是一新增 Windows Forms App 專案即存在的容器控制項。因為 Form 的上面可以放置其他的輸出入控制項，所以又稱為容器控制項。



◎ 屬性

Form 最常用的屬性，分別是 Name 及 Text。

Name：物件名稱。所有控制項的物件名稱均使用 Name。

Text：標題。

◎ 事件

Form 最常用的事件分別是 Load、Click、DoubleClick，分別說明如下：

Load：Form 載入時，產生此事件。

Click：Form 被按一下時，產生此事件。

DoubleClick：Form 被按兩下時，產生此事件。

表單內取用表單屬性，必須以 Me 開頭。例如：以下敘述可以設定其 Text 屬性。

```
Me.Text="Hello"
```

範例 3-2a

請設計一程式，滿足以下條件：

1. 於 Form 載入時，使用 Form 的標題向使用者說「Hello」。
2. 當使用者於 Form 按一下時，使用 Form 的標題向使用者問「早安」。
3. 當使用者於 Form 按兩下時，結束程式。





執行結果



程式列印



```
Public Class Form1
    Private Sub Form1_Click(ByVal sender As Object, ByVal e As System.
EventArgs) Handles Me.Click
        Me.Text = " 早安 "
    End Sub

    Private Sub Form1_DoubleClick(ByVal sender As Object, ByVal e As System.
EventArgs) Handles Me.DoubleClick
        End
    End Sub

    Private Sub Form1_Load(ByVal sender As System.Object, ByVal e As System.
EventArgs) Handles MyBase.Load
        Me.Text = "Hello"
    End Sub
End Class
```



補充說明

為了節省篇幅，本書往後各章均將事件內的參數刪除。例如：Click 事件簡化如下：

```
Private Sub Form1_Click() Handles Me.Click
```



3-3 Label

Label 的主要功能是輸出結果，或於設計階段輸出某些提示字元，如下圖所示。



將 Label 安排於 Form 的畫面，Text 屬性預設 Label1，且物件名稱預設為 Label1。



將 Text 屬性修改為「請輸入姓名」。

◎ 屬性

Label 最常用的屬性，分別是 Name 及 Text。

Name：物件名稱。

Text：標題。上圖左的 Text 為 Label1。上圖右則修改為「請輸入姓名」。

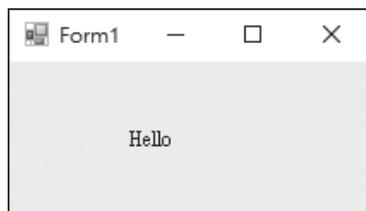
(註 VB6 的 Form、Label 與 Command 標題是 Caption，但 .NET 均統一使用 Text 了。)

範例 3-3a

請設計一程式，滿足以下條件：

1. 於 Form 載入時，使用 Label 的標題向使用者說「Hello」。
2. 當使用者於 Form 按一下時，使用 Label 的標題向使用者問「早安」。
3. 當使用者於 Form 按兩下時，結束程式。

執行結果





程式列印

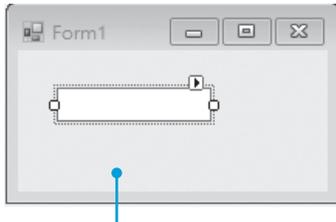


```
Public Class Form1
    Private Sub Form1_Click() Handles Me.Click
        Label1.Text = " 早安 "
    End Sub
    Private Sub Form1_DoubleClick() Handles Me.DoubleClick
        End
    End Sub
    Private Sub Form1_Load() Handles MyBase.Load
        Label1.Text = "Hello"
    End Sub
End Class
```



3-4 TextBox

TextBox 的主要功能是輸入資料，如下圖所示。



於 Form 安排一個 TextBox 的畫面，其 Text 屬性預設值是空白。



安排一個 Label 及 TextBox 的畫面，Label 通常是輸入提示或輸出資料，TextBox 則是輸入物件。

◎ 屬性

TextBox 最常用的屬性，分別是 Name、Text 及 Multiline。

Name：物件名稱。所有控制項的物件名稱均使用 Name。

Text：TextBox 之值。上圖右的 Text 已修改為 5。

Multiline：是否允許多列，系統預設為 False；若要允許輸入多列，則應修改此屬性為 True。



◎ 事件

TextBox 的常用事件分別是 Change、MouseMove，分別說明如下：

TextChanged：TextBox 的值改變時，產生此事件。

MouseMove：滑鼠指標經過 TextBox 上方時，產生此事件，通常亦可當作輸入提示。

◎ 字串與數值

TextBox 與 Label 控制項的 Text 屬性的型別都是字串，所以若有程式如下：

```
a = TextBox1.Text  
b = TextBox2.Text  
c = a + b
```

則其結果將會是字串相加。若要進行數值相加，則應先轉為數值，轉為數值的方法有很多種，常用的是 Val() 函式與物件導向的 Integer.Parse() 方法。例如：

```
a = Val(TextBox1.Text)  
b = Val(TextBox2.Text)  
c = a + b
```

或

```
a = Integer.Parse (TextBox1.Text)  
b = Integer.Parse (TextBox2.Text)  
c = a + b
```

都是數值的加法。同理，運算結果若為數值，且要由 TextBox 或 Label 控制項輸出，亦要先轉為字串，轉為字串常用的方式有 Str() 函式與 ToString() 方法。如以下敘述：

```
Label1.Text=Str(c)
```

或

```
Label1.Text=c.ToString()
```



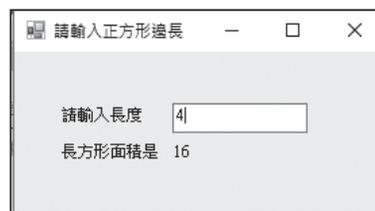
範例 3-4a



請寫一程式，滿足以下條件：

1. 可以於 TextBox 輸入文字或數字。
2. 輸入資料的過程，其值改變時均由 Label 輸出 TextBox 值的平方。
3. 當滑鼠指標指向 TextBox 時，可於表單標題顯示「請輸入正方形邊長」的提示。

表單配置與輸出結果



程式列印



```
Public Class Form1
    Private Sub TextBox1_MouseMove() Handles TextBox1.MouseMove
        Me.Text = "請輸入正方形邊長"
    End Sub
    Private Sub TextBox1_TextChanged() Handles TextBox1.TextChanged
        Dim a, b As Single
        a = Val(TextBox1.Text)
        b = a ^ 2
        Label3.Text = b.ToString()
    End Sub
End Class
```

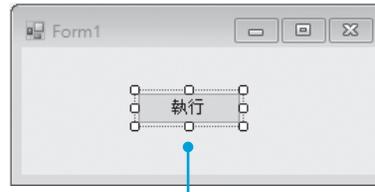


3-5 Button

Button 的主要功能是提供一個按鈕，供使用者執行某項功能，如下圖所示。



安排一個 Button，其 Text 屬性預設值是 Button1，Name 亦為 Button1。



將 Text 屬性值改為「執行」的畫面。

◎ 屬性

Button 最常用的屬性，分別是 Name、Text。

Name：物件名稱。所有控制項的物件名稱均使用 Name。

Text：Button 之標題，上圖右則為「執行」。

◎ 事件

Button 的常用事件分別是 Click。

Click：Button 被按一下時，執行此事件。

範例 3-5a

請寫一程式，滿足以下條件：

1. 可以輸入兩個數字。
2. 安排四個 Button 按鈕，其標題分別是 +、-、* 及 /。
3. 按一下以上四個按鈕，則可由 Label 輸出以上兩個數字相加、減、乘及除的結果。





表單配置與執行結果



程式列印



```
Public Class Form1
    Private Sub Button1_Click() Handles Button1.Click
        Dim a, b, c As Short
        a = Integer.Parse(TextBox1.Text) 'a=Val(Textbox1.text)
        b = Integer.Parse(TextBox2.Text)
        c = a + b
        Label1.Text = "+"
        Label3.Text = c.ToString()
    End Sub
    Private Sub Button2_Click() Handles Button2.Click
        Dim a, b, c As Short
        a = Integer.Parse(TextBox1.Text) 'a=Val(Textbox1.text)
        b = Integer.Parse(TextBox2.Text)
        c = a - b
        Label1.Text = "-"
        Label3.Text = c.ToString()
    End Sub
    Private Sub Button3_Click() Handles Button3.Click
        Dim a, b, c As Short
        a = Integer.Parse(TextBox1.Text) 'a=Val(Textbox1.text)
        b = Integer.Parse(TextBox2.Text)
```



```
        c = a * b
        Label11.Text = "*"
        Label13.Text = c.ToString()
    End Sub
    Private Sub Button4_Click() Handles Button4.Click
        Dim a, b, c As Short
        a = Integer.Parse(TextBox1.Text) 'a=Val(Textbox1.text)
        b = Integer.Parse(TextBox2.Text)
        c = a / b
        Label11.Text = "/"
        Label13.Text = c.ToString()
    End Sub
    Private Sub Button5_Click() Handles Button5.Click
        End
    End Sub
End Class
```

◎ 物件縮碼

物件的命名通常是依照物件的產生順序命名，例如 `TextBox1`、`TextBox2`、`TextBox3`，此種方式通常到程式設計階段都會忘記每一控制項的功能，造成不斷翻閱表單與程式的困擾，為了解決此一問題，通常必須將物件按照其功能重新命名。而物件名稱若能保有類別的部分名稱，則可提高程式可讀性，所以遂有物件縮碼的構想。例如：`TextBox` 先以 `txt` 為開頭命名，`Label` 先以 `lbl` 為開頭命名，`Button` 先以 `btn` 開頭命名。其次，才依照此物件的功能接續命名。例如：`Button1` 的功能是相加，所以命名為 `btnadd`，`Button2` 的功能是相減，所以是 `btnsub`，其餘依此類推。所以以上程式，通常撰寫如下。



```
Public Class Form1
    Private Sub btnadd_Click() Handles btnadd.Click
        Dim a, b, c As Short
        a = Val(txta.Text) ' 使用函式
        b = Val(txtb.Text)
        c = a + b
        lblop.Text = "+"
        lblresult.Text = c.ToString()
    End Sub
    Private Sub btnsub_Click() Handles btnsub.Click
        Dim a, b, c As Short
        a = Val(txta.Text) ' 使用函式
        b = Val(txtb.Text)
        c = a - b
        lblop.Text = "-"
        lblresult.Text = c.ToString()
    End Sub
    Private Sub btnmul_Click() Handles btnmul.Click
        Dim a, b, c As Short
        a = Val(txta.Text) ' 使用函式
        b = Val(txtb.Text)
        c = a * b
        lblop.Text = "*"
        lblresult.Text = c.ToString()
    End Sub
    Private Sub btndiv_Click() Handles btndiv.Click
        Dim a, b, c As Short
        a = Val(txta.Text) ' 使用函式
        b = Val(txtb.Text)
        c = a / b
        lblop.Text = "/"
        lblresult.Text = c.ToString()
    End Sub
    Private Sub btnend_Click() Handles btnend.Click
        End
    End Sub
End Class
```



◎ 共用事件

若事件之間有許多共同的程式片段，則可使用「共用事件」處理。例如：上面的四個按鈕，都有一些程式共用，則可以寫成以下敘述。請於 Button2~4 的 Click 事件均點選 Button1_Click。

```
Public Class Form1
Private Sub Button1_Click(ByVal sender As System.Object) Handles Button1.
Click, Button2.Click, Button3.Click, Button4.Click
    Dim n As Char
    Dim a, b As Short
    Dim c As Single
    n = CType(sender, Button).Text
    a = Val(TextBox1.Text)
    b = Val(TextBox2.Text)
    Select Case n
        Case "+"
            c = a + b
        Case "-"
            c = a - b
        Case "*"
            c = a * b
        Case "/"
            c = a / b
    End Select
    Label1.Text = sender.Text
    Label3.Text = c.ToString()
End Sub
End Class
```

以上程式的 Handles 後面跟著 4 個事件，如以下敘述：

```
Private Sub Button1_Click ()Handles Button1.Click, Button2.Click, Button3.
Click, Button4.Click
```

即代表共用 Button1_Click 程序。其次

```
n = CType(sender, Button).Text
```



Ctype 是型別轉換函式，可將來源的 sender 從 Object 型別轉為 Button 型別，再取對應到 Button 的 Text 屬性，以便進一步判別應進行何項運算。

動手做

1. 假設某次考試成績資料如右：55、66、77、88、99

- (1) 請寫一程式輸入以上資料。
- (2) 計算總和與平均。
- (3) 輸出總和與平均。

2. 假設某次考試成績資料如右：

請寫一程式：

- (1) 可輸入以上資料。
(請先不要使用後面的迴圈)
- (2) 計算每一人平均並輸出。
- (3) 計算每科平均並輸出。

座號	姓名	國文	英文	平均
1	AA	22	33	
2	BB	44	55	
3	CC	66	77	
平均				

3. 自由落體運動中，垂直上拋初速以 v 表示，垂直距離以 y 表示，則 y 與初速 v 、時間 t 的關係為 $y = vt - (10 \times t^2) / 2$ ，請寫一程式，可輸入 v 與 t 而求得垂直落下的距離。

$y(m)$	$v (m/s)$	$t(s)$
	10	1
	10	2
	0	1
	0	2

範例 3-5b

請寫一程式，滿足以下條件：

1. 可以輸入兩個座標。
2. 計算此兩點座標距離。
3. 輸出此兩點距離。

表單配置

物件的命名應有意義，程式撰寫與維護才會輕鬆。例如：本例座標的輸入採用 txtx1、txty1、txtx2、txty2，資料輸出採用 lblout。



輸出結果



Form1

第一點座標

第二點座標

其距離為 Label4

Form1

第一點座標

第二點座標

其距離為 5

程式列印



```
Private Sub Button1_Click() Handles Button1.Click
    Dim x1, x2, y1, y2 As Short
    Dim d As Single
    x1 = Val(txtx1.Text)
    y1 = Val(tyty1.Text)
    x2 = Val(txtx2.Text)
    y2 = Val(tyty2.Text)
    d = ((x1 - x2) ^ 2 + (y1 - y2) ^ 2) ^ (1 / 2)
    lblout.Text = Str(d)
End Sub
```

動手做



- 輸入三角形三邊長 a 、 b 、 c ，求其面積。（已知邊長，則面積 $= \sqrt{d(d-a)(d-b)(d-c)}$ ，其中 $d = \frac{a+b+c}{2}$ ，本例假設所輸入的三邊長可圍成三角形，例如：輸入 3、4、5 得到面積 6）。
- 輸入三個座標點，求三角形面積。例如：輸入 (3, 0)、(0, 4)、(0, 0)，將可得面積 6。
! 提示：已知三角形頂點座標，則其面積可由行列式法求得，公式如下且可推廣至任何凸多邊形。

$$\frac{1}{2} \begin{vmatrix} x_1 & x_2 & x_3 & x_1 \\ y_1 & y_2 & y_3 & y_1 \end{vmatrix} = \frac{1}{2} (x_1y_2 + x_2y_3 + x_3y_1 - x_2y_1 - x_3y_2 - x_1y_3)$$



範例 3-5c

寫一個程式，可以輸入一個一元二次方程式，並求其解。（本例假設所輸入的方程式有二解，例如： $2x^2 - 7x + 3 = 0$ 。其解為 $x_1 = 0.5$ 、 $x_2 = 3$ ）



演算法則



1. 設有一元二次方程式： $ax^2 + bx + c = 0$ 。
2. 令 $d = \sqrt{b^2 - 4ac}$ 。
3. 則方程式二解分別為 $x_1 = \frac{-b+d}{2a}$ 、 $x_2 = \frac{-b-d}{2a}$ 。

表單配置與輸出結果



程式列印



```
Private Sub Button1_Click() Handles Button1.Click
    Dim a, b, c As Short
    Dim d As Single
    Dim x1, x2 As Single
    'input
    a = Val(txta.Text)
    b = Val(txtb.Text)
    c = Val(txtc.Text)
    'process
    d = b ^ 2 - 4 * a * c
    x1 = (-b + d ^ (1 / 2)) / (2 * a) '小心，括號漏了就錯
```



```

x2 = (-b - d ^ (1 / 2)) / (2 * a)
'output
lblx1.Text = Str(x1)
lblx2.Text = Str(x2)
End Sub

```

動手做

寫一個程式，可以輸入一個二元一次方程式，並求其解。（本例假設所輸入的方程式恰有一解）

! 提示：本例演算法如下

①設二元一次方程式： $a_1x + b_1y = c_1$ 、 $a_2x + b_2y = c_2$ 。

②令 $d = \begin{vmatrix} a_1 & b_1 \\ a_2 & b_2 \end{vmatrix} = a_1b_2 - a_2b_1$ 。

③則其解分別是 $x = \frac{\begin{vmatrix} c_1 & b_1 \\ c_2 & b_2 \end{vmatrix}}{d} = (c_1b_2 - c_2b_1) / d$ 、 $y = \frac{\begin{vmatrix} a_1 & c_1 \\ a_2 & c_2 \end{vmatrix}}{d} = (a_1c_2 - a_2c_1) / d$ 。

④例如： $3x + y = 5$ 、 $x - 2y = -3$ ，則其解為 $x = 1$ 、 $y = 2$ 。

範例 3-5d

計數器。請設計一程式，安排一個 Label 與 Button，每當使用者按一下 Button，則 Label 的顯示值增加一。

執行結果



程式列印



本例的變數 `c` 一定要宣告在方法外面，才可保留至下一次此變數被呼叫。

```
Public Class Form1
    Dim c As Integer
    Private Sub Button1_Click() Handles Button1.Click
        c += 1
        Label1.Text = c
    End Sub
    Private Sub Form1_Load() Handles MyBase.Load
        c = 0
    End Sub
End Class
```

其次，亦可宣告 `s` 為公用（或譯為靜態）變數（`Static`）如以下程式，其執行結果亦相同。

```
Public Class Form1
    Private Sub Button2_Click() Handles Button2.Click
        Static s As Short = 0
        s = s + 1
        Label2.Text = s.ToString()
    End Sub
End Class
```

動手做



請寫一程式，共包含四個按鈕，可分別遞增一、遞增十、遞減一，及遞減十。

範例 3-5e



請寫一程式，滿足以下條件：

1. 可以輸入兩個數字。
2. 交換此兩個數字。
3. 輸出交換的結果。



輸出結果



Form1 window showing input fields for '輸入第一數' (3) and '輸入第二數' (4). Buttons: 交換, 結束.

Form1 window showing input fields for '輸入第一數' (4) and '輸入第二數' (3). Buttons: 交換, 結束.

程式列印



```
Public Class Form1
    Private Sub Button1_Click() Handles Button1.Click
        ' 宣告
        Dim a, b, t As Short
        'input
        a = Val(txta.Text)
        b = Val(txtb.Text)
        'process
        t = a
        a = b
        b = t
        'output
        txta.Text = Str(a)
        txtb.Text = Str(b)
    End Sub
End Class
```

程式說明



兩個數要交換，就如同兩隻手的東西要交換。所以先設定兩隻手的名稱分別為 a 與 b，現要交換其內容，則其方法如下：

1. 先找來第 3 隻手 t。
2. 將 a 手的東西交給 t。

```
t=a
```



3. 將 b 手的東西交給 a。

```
a=b
```

4. 將 t 手的東西交給 a，而完成兩隻手上東西的交換。

```
a=t
```

5. 其次，電腦並無法同時拋出兩隻手的東西，且同時接住另一隻手的東西，所以以下敘述當然得到錯誤的結果。

```
a=b
```

```
b=a
```

動手做



數字右旋或左旋。請安排四個 Label 與一個按鈕，每當使用者按一下按鈕，可將 Label1 的值交給 Label2，Label2 交給 Label3，Label3 交給 Label4，Label4 的值交給 Label1，此即為右旋。



3-6 PictureBox

PictureBox（圖片盒）允許載入點陣圖（*.bmp）、圖示（*.ico）、中繼檔（Metafile，*.wmf）、*.jpeg、可攜式網路圖檔（*.png）及 *.gif 等型態的圖形檔案。其常用屬性如下：

◎ Image 屬性

用於設定所要顯示的圖形檔。於程式設計階段點選 image 屬性，畫面如右圖。





◎ 本機資源

使用本機硬碟的檔案，稱為本機資源。請於上圖點選「本機資源／匯入」，再點選所要的圖檔，即可出現上圖。若要於程式執行階段載入本機資源檔案圖檔，則應該使用 `Image.FromFile` 方法，如以下敘述：

```
PictureBox1.Image=Image.FromFile("d:\vbbook\gwosheng.bmp")
```

或是使用 `Bitmap` 類別的建構函式，如以下敘述：

```
PictureBox1.Image=New Bitmap("d:\vbbook\gwosheng.bmp")
```

都可載入圖檔。其次，若要省略圖檔路徑，則應將圖檔複製到專案資料夾下的 `bin/Debug` 資料夾。

◎ 資源檔

若於執行階段需要不斷改變圖檔來源，為了提高程式執行效率與簡化程式撰寫，請先將圖檔匯入專案資源檔，如右圖所示。（請於右圖點選「專案資源檔」，並先匯入所有的圖檔）。待於程式設計階段，若要載入專案資源檔，則程式如下：（請留意副檔名與路徑都已經省略）



```
PictureBox1.Image = My.Resources.g1
```

◎ SizeMode

用來設定圖形的顯示模式，如右表所示。

成員名稱	說明
Normal	影像放置在 <code>PictureBox</code> 的左上角。如果影像大於包含它的 <code>PictureBox</code> ，就會裁剪影像。
StretchImage	<code>PictureBox</code> 內的影像會延伸或縮小，以調整成最適合 <code>PictureBox</code> 的大小。
AutoSize	將 <code>PictureBox</code> 的大小調整成與其所包含影像的大小相等。
CenterImage	如果 <code>PictureBox</code> 大於影像，影像即置中顯示。如果影像大於 <code>PictureBox</code> ，圖片即放在 <code>PictureBox</code> 的中央，而外緣被裁剪。
Zoom	不論是增大或縮小，影像大小都維持大小比例。



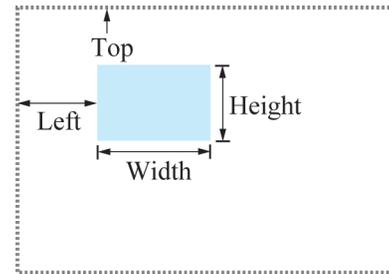
例如：以下程式可於執行階段設定為 Normal 顯示。

```
PictureBox1.SizeMode = PictureBoxSizeMode.Normal
```

◎ Left,Top

控制項的位置 (x , y) 屬性，如右圖所示。若於程式執行階段改變其值，則有動畫的效果。例如：以下程式，可將控制項向右移動 10 個單位。

```
PictureBox1.Left = PictureBox1.Left + 10
```



◎ Height,Width

控制項的高度與寬度屬性，如右圖所示。若於程式執行階段改變其值，則有動畫的效果。例如：

```
PictureBox1.Width= PictureBox1.Width+50
```

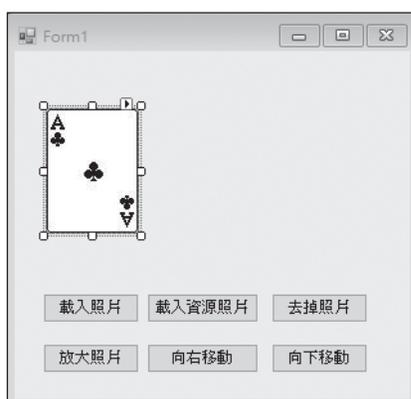
則可將控制項的大小增加 50 個單位。

範例 3-6a

請設計一個程式，滿足以下條件：

1. 可以使用兩個按鈕改變圖檔顯示內容。例如：點選「洪國勝」，則出現洪國勝的照片，點選「蔡依林」，則出現蔡依林的照片。
2. 可以使用兩個 Button 任意調整 PictureBox 的大小，及可以使用四個按鈕分別控制控制項的上、下、左、右移動。

表單配置與執行結果





程式列印



```
Public Class Form1
    Private Sub Button1_Click() Handles Button1.Click
        PictureBox1.Image = Image.FromFile("d:\vbbook\gwosheng.bmp")
    End Sub
    Private Sub Button2_Click() Handles Button1.Click
        PictureBox1.Image = My.Resources.g1
    End Sub
    Private Sub Button3_Click() Handles Button3.Click
        PictureBox1.Image = Nothing
    End Sub
    ' 放大
    Private Sub Button4_Click() Handles Button4.Click
        PictureBox1.Width = PictureBox1.Width + 50
        PictureBox1.Height = PictureBox1.Height + 50
    End Sub
    ' 向右
    Private Sub Button5_Click() Handles Button5.Click
        PictureBox1.Location = New Point(PictureBox1.Location.X + 10,
        PictureBox1.Location.Y)
        'PictureBox1.Left = PictureBox1.Left + 10
    End Sub
    ' 向下
    Private Sub Button6_Click() Handles Button6.Click
        PictureBox1.Top = PictureBox1.Top + 50
    End Sub
End Class
```



3-7 Timer

Timer（時間）控制項通常用來作為與時間有關的功能。例如：您希望若干時間自動執行一些事情，此時即可使用 Timer 控制項。其最主要的屬性為 Interval，單位是 ms（千分之一秒）。其次，最常用的事件是 Tick。例如：若設定

```
Timer1.Interval=1000
Timer1.Enabled=True
```

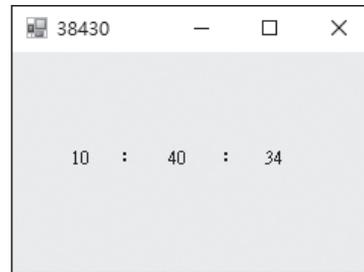
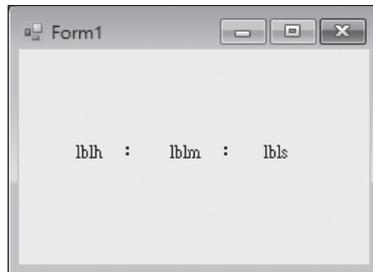
則程式將每隔 1 秒，自動執行 Timer1_Timer() 事件，直到 Timer1.Enable=False。

範例 3-7a

試設計一個數字時鐘或計時器。



表單配置與執行結果



演算說明

本例將一天的時間，定義為 0 到 $(60 \times 60 \times 24 - 1)$ 。則秒針 s、分針 m、時針 h 分別如下：

```
t = (t + 1) Mod (24 * 60 * 60)
s = t Mod 60
m = (t \ 60) Mod 60
h = t \ 60 \ 60
```



程式列印



```
Public Class Form1
    Dim t As Long
    Private Sub Timer1_Tick() Handles Timer1.Tick
        Dim h, m, s As Short
        t = (t + 1) Mod (24 * 60 * 60)
        s = t Mod 60
        m = (t \ 60) Mod 60
        h = t \ 60 \ 60
        lbls.Text = s.ToString()
        lblm.Text = m.ToString()
        lblh.Text = h.ToString()
    End Sub
    Private Sub Form1_Load() Handles MyBase.Load
        t = (Date.Now.Hour * 60 * 60 + Date.Now.Minute * 60 + Date.Now.Second)
        Me.Text = t
        Timer1.Interval = 1000
        Timer1.Enabled = True
    End Sub
End Class
```

動手做



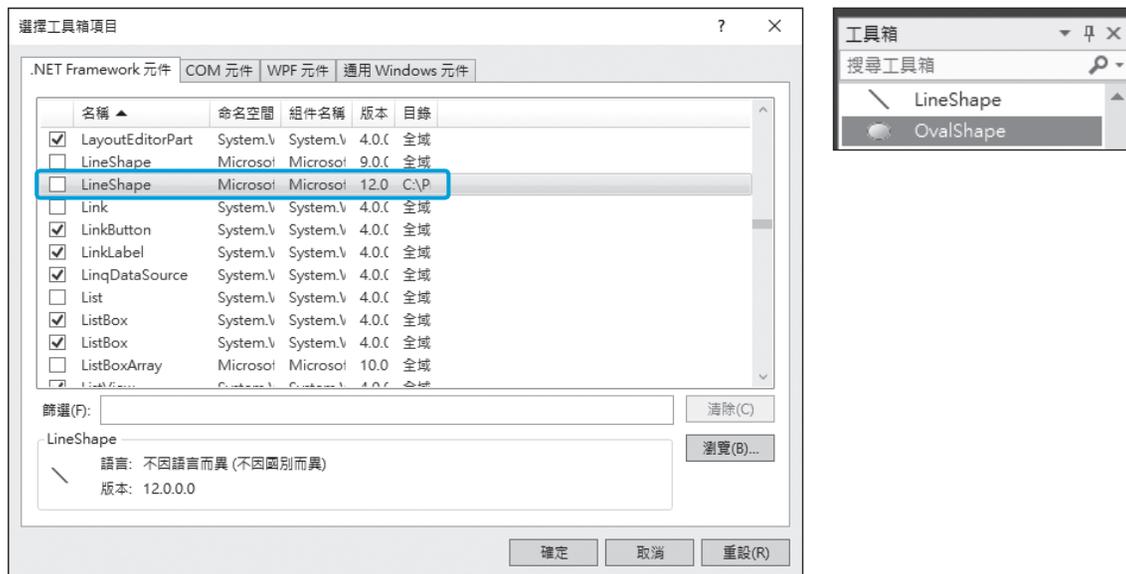
1. 請設計一程式，於表單載入時讓 PictureBox 由小逐漸變大，當超過邊界時，亦可變回最小。
2. 請設計一程式，於表單載入時讓 PictureBox 逐漸向右移動，當跑到最右邊時，亦可回到最左邊。
3. 計時器。請設計一個程式，當按鈕被按時，開始計時，再按一下時，傳回計時時間。



3-8 LineShape

LineShape（線段）位於 Visual Basic PowerPacks，通常用來繪製直線，但是 VB2010 以後就不是標準控制項，請先於技術支援搜尋 LineShape，然後下載，並安裝。再於工具列點選快顯功能表的「選擇項目」（如下圖左），並勾選 LineShape，即可於工具箱顯示此控制項，如下圖右。

註 下圖左若看不到勾號，請用滑鼠將名稱右移。



◎ 屬性

$x1$, $y1$, $x2$, $y2$ ：此 4 個屬性用來設定直線的兩端點座標。使用者於設計階段可用滑鼠指著直線兩旁的黑點，拖曳直線至適當位置。執行階段則可重設 ($x1$, $y1$) , ($x2$, $y2$) 而改變直線的位置。

BorderWidth：線條寬度。

BorderColor：線條顏色。

BorderStyle：線條樣式。



範例 3-8a

請用直線物件製作考試分數長條圖。



表單配置與輸出結果



程式列印



```
Public Class Form1
    Dim l As Short
    Private Sub Form1_Load() Handles MyBase.Load
        lin.BorderWidth = 5
        lin.BorderColor = Color.Red
        l = lin.X2 - lin.X1
    End Sub
    Private Sub Button1_Click() Handles Button1.Click
        lin.X2 = lin.X1 + (Val(TextBox1.Text) / 100) * l
    End Sub
End Class
```

動手做



同上範例，製作直立長條圖。



範例 3-8b

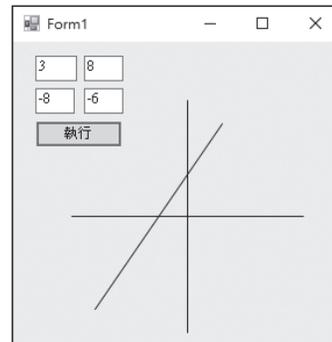
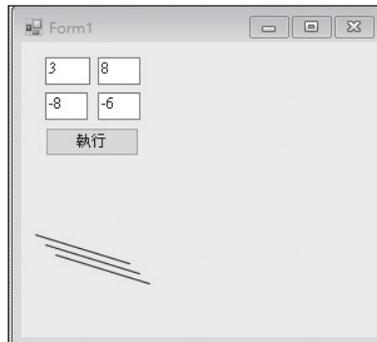
請寫一程式，可以輸入任意兩點座標，且繪出其直線。



表單配置與輸出結果



1. 配置 4 個 TextBox，Name 分別修改為 txtx1、txty1、txtx2、txty2，功能為輸入直線兩端點座標，如下圖左。
2. 配置 3 個 LineShape，Name 分別修改為 linx、liny 及 lin，可分別繪出 x 軸、y 軸及對應直線，如下圖右。



演算法則

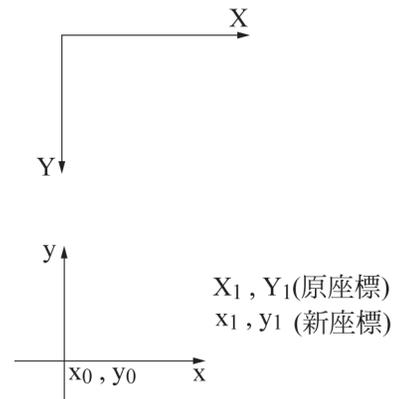


表單的座標系統 (X, Y) 如右上圖大寫所示，原點 (0, 0) 在左上角，X 軸向右遞增，Y 軸向下遞增，單位是像素。但是數學繪圖的座標系統 (x, y)，如右下圖小寫所示，原點通常在中間，且 x 軸向右遞增，y 軸向上遞增，且單位長度通常遠大於像素。所以，我們要有 3 個動作，調整以上因素，分別是平移至 (x0, y0)：

```
x0 = 100
y0 = 100
X1 = x0+x1
Y1 = y0+y1
```

調整 y 軸方向：

```
dy=-1
```





及以經驗值調整水平與垂直放大倍數。

```
sx = 10
sy = 10
X1=x0+x1*sy
Y1=y0+y1*sy*dy
```

程式列印

```
Public Class Form1
    Dim x0, y0 As Short
    Private Sub Button1_Click() Handles Button1.Click
        Dim x1, y1, x2, y2 As Short
        Dim sx, sy As Short
        ' 水平與垂直放大倍數
        sx = 10
        sy = 10
        'y 座標方向調整
        Dim dy As Short = -1
        x1 = x0 + Val(txtx1.Text) * sx
        y1 = y0 + Val(tyty1.Text) * sy * dy
        x2 = x0 + Val(txtx2.Text) * sx
        y2 = y0 + Val(tyty2.Text) * sy * dy
        lin.X1 = x1
        lin.Y1 = y1
        lin.X2 = x2
        lin.Y2 = y2
    End Sub
    Private Sub Form1_Load() Handles MyBase.Load
        Me.Height = 300
        Me.Width = 300
        ' 座標原點
        x0 = 150
        y0 = 150
        'x 軸
```



```
linx.X1 = x0 - 100
linx.Y1 = y0
linx.X2 = x0 + 100
linx.Y2 = y0
'y 軸
liny.X1 = x0 + 0
liny.Y1 = y0 + 100
liny.X2 = x0 + 0
liny.Y2 = y0 - 100
End Sub
End Class
```

範例 3-8c

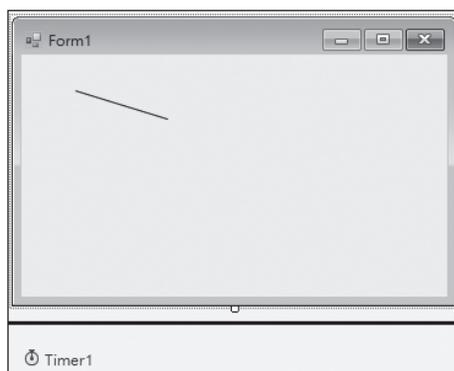
示範製作秒針。



表單配置與執行結果



1. 配置 1 個 LineShape，修改其 Name 為 lin。
2. 配置 1 個 Timer，修改其 Interval 屬性為 1000，Enabled 屬性為 True。





程式列印



程式解說請看專題製作。

```
Public Class Form1
    Private Sub Timer1_Tick() Handles Timer1.Tick
        Dim s As Short
        s = Date.Now.Second
        Dim x0, y0, x1, y1, x2, y2 As Short
        x0 = 150
        y0 = 150
        x1 = x0
        y1 = y0
        Dim s1 As Short = 10
        Dim d As Single
        d = (-2 * Math.PI * s / 60) + Math.PI / 2
        x2 = s1 * Math.Cos(d)
        y2 = s1 * Math.Sin(d)
        Dim sx, sy, dy As Short
        sx = 10
        sy = 10
        dy = -1
        x2 = x0 + x2 * sx
        y2 = y0 + y2 * sy * dy
        lin.X1 = x0
        lin.Y1 = y0
        lin.X2 = x2
        lin.Y2 = y2
        Me.Text = Now
    End Sub
End Class
```



3-9 OvalShape

OvalShape（圓與橢圓）控制項位於 Visual Basic PowerPacks（下載與安裝方式請看 3-8 節的 LineShape），通常用來顯示圓與橢圓，其常用屬性如下：

Location.X ,Location.Y：控制項的左上角座標。

Size.Width,Size.Height：控制項的寬度與高度。

FillStyle 屬性：填充的樣式，常用的有透明（Transparent）、填滿（Solid）。

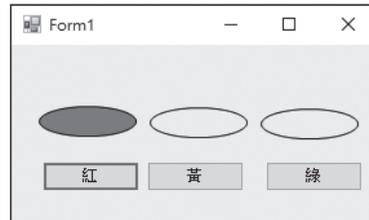
FillColor：填充的顏色。

範例 3-9a

紅綠燈設計。請設計一程式，使用 3 個按鈕，其名稱分別是紅、黃、綠，控制 3 個 OvalShape 控制項，其顏色分別是 Red、Yellow、Green，當使用者按紅燈時，紅燈亮；按黃燈時，黃燈亮；按綠燈時，綠燈亮。



執行結果



程式列印

```
Public Class Form1
    ' 紅
    Private Sub btnred_Click() Handles btnred.Click
        ovsred.FillColor = Color.Red
        ovsred.FillStyle = PowerPacks.FillStyle.Solid
        ovsyel.FillStyle = PowerPacks.FillStyle.Transparent
        ovsgre.FillStyle = PowerPacks.FillStyle.Transparent
    End Sub
```



```
' 黃
Private Sub btnyel_Click() Handles btnyel.Click
    ovsred.FillStyle = PowerPacks.FillStyle.Transparent
    ovsyel.FillColor = Color.Yellow
    ovsyel.FillStyle = PowerPacks.FillStyle.Solid
    ovsgre.FillStyle = PowerPacks.FillStyle.Transparent
End Sub
' 綠
Private Sub btngre_Click() Handles btngre.Click
    ovsred.FillStyle = PowerPacks.FillStyle.Transparent
    ovsyel.FillStyle = PowerPacks.FillStyle.Transparent
    ovsgre.FillColor = Color.Green
    ovsgre.FillStyle = PowerPacks.FillStyle.Solid
End Sub
End Class
```

動手做



1. 請設計一程式，可以自動展示 OvalShape 物件的填滿樣式（每秒換一種顏色），與填滿顏色的變化。
2. 請設計一程式，可以展示 OvalShape 物件的填滿顏色。



3-10 Random 類別

日常生活有猜拳、擲骰子及電腦開獎等活動，這些遊戲都必須藉由亂數的產生，VB 則提供 Random 類別，讓使用者運用。關於 Random 類別的建構函式、方法，請看以下說明：

◎ 建構函式

Random 類別的建構函式共有兩種多載，分別是 Public Sub Random() 與 Public Sub New(Integer)，前者是使用系統時間當亂數的種子，後者是自己給一個特殊的整數當亂數種子。

註 「多載」：請參閱 7-1 節「方法多載」P241。



◎ 實例方法

本單元將介紹以下 3 個常用的實例方法，分別是 `Next()`、`NextBytes()` 及 `NextDouble()`。

1. Next

`Next()` 共有 3 種多載，分別說明如下：

(1) Public Function Next() As Integer

傳回大於等於 0 且小於 `MaxValue`（註：`MaxValue=2,147,483,647`）的整數。
例如：以下敘述可建立一個 `r` 物件，並產生一個 0 至 `MaxValue` 之間的亂數。

```
Dim r As New Random()  
Console.WriteLine(r.Next())
```

(2) Public Function Next(Integer) As Integer

此方法的語法如下：

```
Public Function Next(ByVal maxValue As Integer) As Integer
```

其中 `maxValue` 必須小於等於 `MaxValue`。其次，本方法可傳回一個整數亂數，此數大於等於 0，且小於所輸入的 `maxValue`。例如：以下敘述可傳回 0，1，2，3 之任一數。

```
Console.WriteLine(r.Next(4)) 'r 物件同上
```

(3) Public Function Next(Integer, Integer) As Integer

此方法的語法如下：

```
Public Function Next(  
    ByVal minValue As Integer,  
    ByVal maxValue As Integer  
) As Integer
```

可傳回一個亂數，此亂數大於等於 `minValue` 且小於 `maxValue`。例如：以下敘述可傳回 1~42（含 42）之間的亂數。

```
Console.WriteLine(r.Next(1, 43)) 'r 物件同上
```



2. NextBytes()

NextBytes() 的語法如下：

```
Public Sub NextBytes(ByVal buffer() As Byte)
```

其中，`buffer` 是要放入亂數的位元組陣列。此方法可將 `buffer` 填入 0~255 之間的整數。例如：

```
Dim r As New Random()  
Dim b As Byte() = New Byte(3) {}  
Dim i As Byte  
r.NextBytes(b)  
For Each i In b  
    Console.WriteLine(i)  
Next
```

將可得 4 個 0~255 之間的亂數。

3. NextDouble()

此方法的語法如下：

```
Public Function NextDouble() As Double
```

可傳回一個浮點亂數，此亂數大於等於 0.0 且小於 1.0。例如：

```
Console.WriteLine(r.NextDouble())
```

將可得一個亂數，此亂數大於等於 0.0 且小於 1.0。

範例 3-10a

示範以上 `Random` 類別的公用屬性以及公用方法。



執行結果

請讀者自行開啟範例檔案 `e3_10a`，並瀏覽執行結果。

**範例 3-10b**

請寫一程式，可以擲兩顆骰子，輸出結果。

**執行結果****程式列印**

```
Public Class Form1
    Private Sub Button1_Click() Handles Button1.Click
        Dim a, b As Integer
        Dim r As New Random
        a = r.Next(1, 7)
        b = r.Next(1, 7)
        lbla.Text = a
        lblb.Text = b
    End Sub
End Class
```

**3-11 對話框**

VB 常用的輸出入對話框有 `InputBox` 函式、`MsgBox` 函式及 `Interaction`. `InputBox` 方法，分別說明如下：

◎ InputBox 函式

`InputBox` 函式可出現一個對話框，並等候使用者輸入文字或按一下按鈕，再回傳包含文字方塊內容的字串。其語法如下：

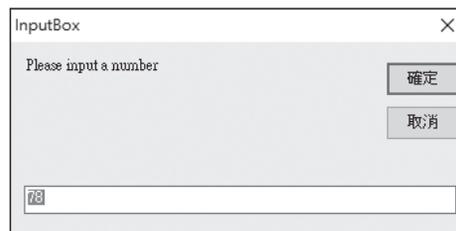


```
Public Function InputBox( _  
    ByVal Prompt As String, _  
    Optional ByVal Title As String = "", _  
    Optional ByVal DefaultResponse As String = "", _  
    Optional ByVal Xpos As Integer = -1, _  
    Optional ByVal YPos As Integer = -1 _  
) As String
```

例如：

```
Dim a As Short  
a = Val(InputBox("Please input a number", "InputBox", "78"))
```

則其輸出如下：



以上程式，若使用方法，則程式如下：

```
a = Val(Interaction.InputBox("Please input a number", "InputBox", "78"))
```

◎ MsgBox 函式

MsgBox 函式可出現一對話框，顯示所要輸出的訊息，其次，此對話框亦可包含確定、取消、終止等按鈕，並回傳使用者所按的結果。其語法如下：

```
Public Function MsgBox( _  
    ByVal Prompt As Object, _  
    Optional ByVal Buttons As MsgBoxStyle = MsgBoxStyle.OKOnly, _  
    Optional ByVal Title As Object = Nothing _  
) As MsgBoxResult
```



1. Prompt：必要項

顯示為對話方塊中訊息的 String 運算式。視使用的字元寬度而定，Prompt 的最大長度是 1024 個字元。如果 Prompt 是由一行以上文字所組成，則您可在每一行之間使用歸位字元（Chr(13)）、換行字元（Chr(10)）或歸位與換行字元的組合（Chr(13) & Chr(10)），來隔開這些行。

2. Buttons：選擇項

數值運算式，這個數值的總和指定要顯示的按鈕數目和類型、要使用的圖示樣式、預設按鈕的識別及訊息方塊的強制回應性。如果省略 Buttons，則預設值為零。

下表所列为 MsgBoxStyle 列舉值，共分五個群組，說明如下：

群組說明	成員	值	描述
1 描述對話方塊中所顯示的按鈕數目和型別	OKOnly	0	僅顯示 [確定] 按鈕。
	OKCancel	1	顯示 [確定] 和 [取消] 按鈕。
	AbortRetryIgnore	2	顯示 [中止]、[重試] 和 [忽略] 按鈕。
	YesNoCancel	3	顯示 [是]、[否] 和 [取消] 按鈕。
	YesNo	4	顯示 [是] 和 [否] 按鈕。
2 描述圖示樣式	RetryCancel	5	顯示 [重試] 和 [取消] 按鈕。
	Critical	16	顯示 [重大訊息] 圖示。
	Question	32	顯示 [警告查詢] 圖示。
	Exclamation	48	顯示 [警告訊息] 圖示。
3 判斷哪個是預設值	Information	64	顯示 [資訊訊息] 圖示。
	DefaultButton1	0	第一個按鈕是預設值。
	DefaultButton2	256	第二個按鈕是預設值。
4 判斷訊息方塊的形式	DefaultButton3	512	第三個按鈕是預設值。
	ApplicationModal	0	應用程式是強制回應的。使用者必須在繼續使用目前應用程式之前先回應訊息方塊。
5 指定訊息方塊視窗是否為前景視窗，以及文字的對齊和方向	SystemModal	4096	系統是強制回應的。所有的應用程式都會暫止，直到使用者對訊息方塊做出回應時為止。
	MsgBoxSetForeground	65536	將訊息方塊視窗指定為前景視窗。
	MsgBoxRight	524288	文字靠右對齊。
	MsgBoxRtlReading	1048576	指定在希伯來文或阿拉伯文的系統上，文字應該以從右到左的書寫方式呈現。

當加入數字來建立 Buttons 引數的最後值時，只能使用每個群組當中的一個數字。



3. Title：選擇項

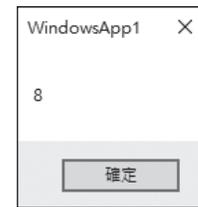
顯示在對話方塊標題列中的 **String** 運算式。如果省略 Title，則會將應用程式名稱放置在標題列中。

4. 函式傳回值：右表所列為此函式傳回值。

例如：以下敘述僅含 Prompt，且 Button 使用預設值 0，所以輸出變數 c 的值，並含一個「確定」按鈕，如右圖。

常數	值
OK	1
Cancel	2
Abort	3
Retry	4
Ignore	5
Yes	6
No	7

```
Dim d,c As Short
c=8
d = MsgBox(c.ToString)
```



當使用者按確定時，其傳回值為 1。又例如：以下敘述。Prompt 為「是否繼續」，Button 為 256 + 16 + 4 = 276，Title 為「標題」。

```
Private Sub Button3_Click() Handles Button3.Click
    Dim msg As String = "是否繼續"
    Dim style As MsgBoxStyle = MsgBoxStyle.DefaultButton2 Or _
        MsgBoxStyle.Critical Or MsgBoxStyle.YesNo
    Dim title As String = "標題"
    Dim response As MsgBoxResult
    response = MsgBox(msg, style, title)
    If response = MsgBoxResult.Yes Then 'User chose Yes.
        'Perform some action.
    Else
        'Perform some other action.
    End If
End Sub
```

結果如右圖。





範例 3-11a

示範上述 InputBox 及 MsgBox 對話框。



執行結果

請自行開啟專案 e3_11a。

◎ Interaction.InputBox 方法

Interaction.InputBox 方法的功能與 InputBox 函式功能相同，用來取代 InputBox 函式，請自行探索。

動手做

請寫程式，可連續用 InputBox 函式輸入兩數，並用 MsgBox 函式輸出結果。

.....



3-12 實例探討

一、向量

向量的表示法有直角座標與極座標，兩者的關係如下：

$$\vec{A} = A \angle \theta = a + jb$$
$$\Rightarrow A = \sqrt{a^2 + b^2}$$
$$\theta = \tan^{-1} \frac{b}{a}$$



範例 3-12a

示範將直角座標轉為極座標。



表單配置與執行結果

程式列印

```
Public Class Form1
    Private Sub Button1_Click() Handles Button1.Click
        Dim a As Single = Val(TextBox1.Text)
        Dim b As Single = Val(TextBox2.Text)
        Dim c As Single = Math.Sqrt(a * a + b * b)
        Dim d As Single = Math.Atan(b / a) ' 徑度量
        Dim f As Single = Math.Atan2(b, a) ' 徑度量
        d = Math.Round(d * 180 / Math.PI, 1) ' 度度量, 四捨五入到小數點第一位
        Label2.Text = c.ToString
        Label4.Text = d.ToString
    End Sub
End Class
```

動手做

請寫一程式，可以將極座標轉為直角座標。



二、低通濾波器的頻率響應（※ 高職電子學）

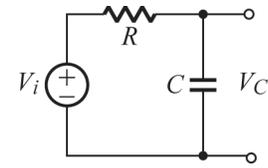
說明：如右圖是一個低通濾波器電路，這個電路包括一個電阻 R 和一個電容器 C ，其電壓增益為輸出電壓 V_0 和輸入電壓 V_i 的比值，可以表示為 $V_0/V_i = 1/(1+j2\pi fRC)$ 。

f ：表示輸入正弦波的頻率，單位為赫茲（Hz）。

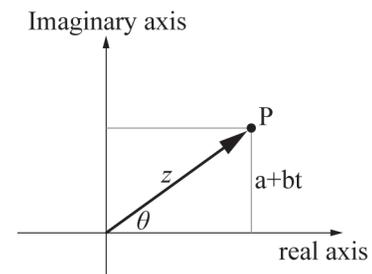
R ：電阻，單位為歐姆（ Ω ）。

C ：電容，單位為法拉（F）。

j ： $\sqrt{-1}$ 。



圖一



圖二

請您寫一個程式：

求出低通濾波器的增益大小 $z = 20 \log_{10} \left| \frac{V_0}{V_i} \right|$ （Amplitude）與相角 θ （Phase）。

輸入：(1) 由使用者輸入電阻值 R ，單位為歐姆（ Ω ）。

(2) 由使用者輸入電容值 C ，單位為法拉（F）。

(3) 由使用者輸入頻率值 f ，單位為赫茲（Hz）。

輸出：(1) 輸出低通濾波器的大小 Z （Amplitude）。

(2) 輸出低通濾波器的相角 θ （Phase）。

例子：輸入電阻值 R ，單位為歐姆 = 1600

輸入電容值 C ，單位為法拉 = 0.000001

輸入頻率值 f ，單位為赫茲 = 1000

濾波器的增益大小 $Z = -20.089\text{dB}$

濾波器的相角 $\theta = -84.319$

**範例 3-12b**

分點座標。已知 $A(x_1, y_1)$ 、 $B(x_2, y_2)$ 為平面上相異兩點，若點 $P(x, y)$ 在直線 AB 上，且 $AP : PB = m : n$ ，則 $P(x, y) = \left(\frac{nx_1 + mx_2}{m+n}, \frac{ny_1 + my_2}{m+n} \right)$ ，請寫一支程式，可以輸入 $A(x_1, y_1)$ 、 $B(x_2, y_2)$ 與 m 、 n ，可求得 P 點座標。

表單配置與執行結果
程式列印

```
Public Class Form1
    Private Sub Button1_Click(sender As Object, e As EventArgs) Handles Button1.Click
        Dim x1, y1, x2, y2, m, n, x, y As Short
        x1 = Integer.Parse(txtx1.Text)
        y1 = Integer.Parse(tyty1.Text)
        x2 = Integer.Parse(txtx2.Text)
        y2 = Integer.Parse(tyty2.Text)
        m = Integer.Parse(txtm.Text)
        n = Integer.Parse(txtn.Text)
        x = (n * x1 + m * x2) / (m + n)
        y = (n * y1 + m * y2) / (m + n)
        lblx.Text = x.ToString()
        lbly.Text = y.ToString()
    End Sub
End Class
```



範例 3-12c

兩點式，兩點可決定一直線。請寫一支程式，可輸入兩個二維座標，並求出此直線方程式。例如：輸入 (2, 1)、(4, 6)，則直線方程式為 $5x - 2y - 8 = 0$ 。



設計步驟



Step 1 ▷ 假設輸入為 $(x_1, y_1), (x_2, y_2)$ 。

Step 2 ▷ 由兩點式，直線方程式為 $y - y_1 = ((y_2 - y_1) / (x_2 - x_1))(x - x_1)$ 。

Step 3 ▷ 令 $a = y_2 - y_1$ ， $b = x_2 - x_1$ 。

Step 4 ▷ 整理上式得方程式為 $ax - by - ax_1 + by_1 = 0$ 。

表單配置與執行結果



程式列印



```
Public Class Form1
    Private Sub Button1_Click(sender As Object, e As EventArgs) Handles Button1.Click
        Dim x1, y1, x2, y2, a, b, c As Short
        x1 = Integer.Parse(txtx1.Text)
        y1 = Integer.Parse(tyty1.Text)
        x2 = Integer.Parse(txtx2.Text)
        y2 = Integer.Parse(tyty2.Text)
        a = y2 - y1
        b = x2 - x1
        c = -a * x1 + b * y1
    End Sub
End Class
```



```

    lbla.Text = a.ToString()
    lblb.Text = -b.ToString()
    lblc.Text = c.ToString()
End Sub
End Class

```

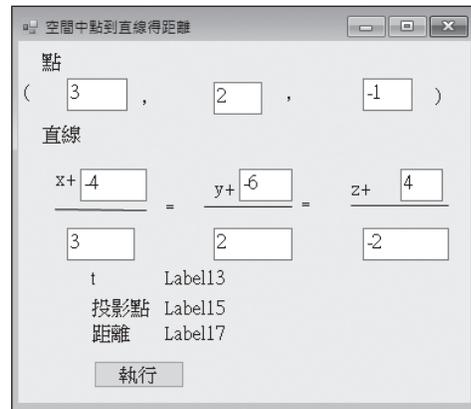
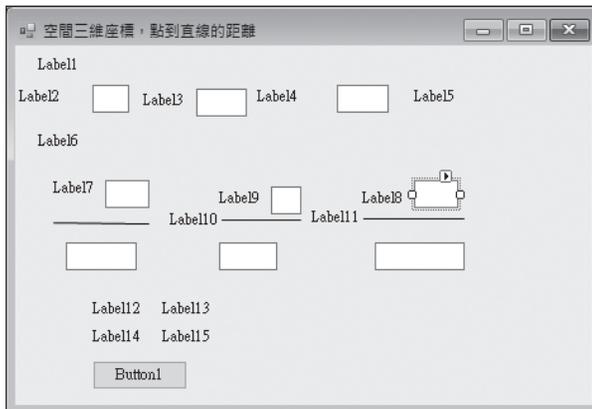
範例 3-12d 空間中點到直線的距離（高中數學）

1. 求空間中一點 $P(a, b, c)$ 在直線 $L: \frac{x+d}{l} = \frac{y+f}{m} = \frac{z+g}{n}$ 的投影點。
2. 點 P 到直線 L 的距離。

操作步驟

Step 1 ▷ 表單控制項初步配置如下圖左。

Step 2 ▷ 修改屬性值如下圖右。



Step 3 ▷ 設 P 在 L 的投影點為 Q ，由 L 之參數式可設 Q 點座標 $Q(\ell t - d, mt - f, nt - g)$ ，且 $\overrightarrow{PQ} = (\ell t - d - a, mt - f - b, nt - g - c)$ 。

Step 4 ▷ L 的方向向量 $\vec{\ell} = (l, m, n)$ 且 $\vec{\ell} \perp \overrightarrow{PQ} \Rightarrow \vec{\ell} \times \overrightarrow{PQ} = 0$
 $\Rightarrow (\ell t - d - a) \times \ell + (mt - f - b) \times m + (nt - g - c) \times n = 0$
 $\Rightarrow \ell^2 t + m^2 t + n^2 t = \ell(d + a) + m(f + b) + n(g + c) = 0$
 $\Rightarrow t = (\ell(d + a) + m(f + b) + n(g + c)) / (\ell^2 + m^2 + n^2)$ 。

Step 5 ▷ 投影點 $Q(\ell t - d, mt - e, nt - f)$ 。

Step 6 ▷ P 到直線 L 的距離 $= |\overrightarrow{PQ}| = \sqrt{(\ell t - d - a)^2 + (mt - f - b)^2 + (nt - g - c)^2}$ 。



執行結果



下圖是 $P(3, 2, -1)$ 在直線 $L: \frac{x-4}{3} = \frac{y-6}{2} = \frac{z+4}{-2}$ 的投影點與距離。

空間中點到直線得距離

點
(3 , 2 , -1)

直線
 $\frac{x-4}{3} = \frac{y-6}{2} = \frac{z+4}{-2}$

t -1
 投影點 Q(1, 4, -2)
 距離 3

執行

程式列印



```
Public Class Form1
    Private Sub Button1_Click(sender As Object, e As EventArgs) Handles Button1.Click
        Dim a, b, c, d, f, g, l, m, n As Short
        Dim t As Single
        Dim dis As Single
        a = Integer.Parse(TextBox1.Text)
        b = Integer.Parse(TextBox2.Text)
        c = Integer.Parse(TextBox3.Text)
        d = Integer.Parse(TextBox4.Text)
        f = Integer.Parse(TextBox5.Text)
        g = Integer.Parse(TextBox6.Text)
        l = Integer.Parse(TextBox7.Text)
        m = Integer.Parse(TextBox8.Text)
        n = Integer.Parse(TextBox9.Text)
        t = (l * (d + a) + m * (f + b) + n * (g + c)) / (l * l + m * m + n * n)
        Label13.Text = t.ToString()
    End Sub
End Class
```



```
Label15.Text = "Q(" + (1 * t - d).ToString() + " , " + (m * t - f).ToString()  
              + " , " + (n * t - g).ToString() + ")"  
Label17.Text = (Math.Sqrt((1 * t - d - a) ^ 2 + (m * t - f - b) ^ 2 + (n * t  
              - g - c) ^ 2)).ToString()  
  
End Sub  
End Class
```

動手做

(以下演算法，請參考高中數學空間向量)

1. 於空間座標中，求兩平行直線的距離。例如：

$$L_1 : \frac{x+1}{2} = \frac{y-1}{2} = \frac{z}{1} \text{ 與 } L_2 : \frac{x-1}{2} = \frac{y}{2} = \frac{z+2}{1} \text{ 之間的距離是 } 3。$$

2. 於空間座標中，請寫一程式，可求兩歪斜線的距離。例如：

$$L_1 : \frac{x-3}{2} = \frac{y-1}{2} = z-2 \text{ 與 } L_2 : \frac{x-4}{-2} = y+4 = \frac{z-1}{2} \text{ 的距離是 } 3。$$

3. 於空間座標中，請寫一程式，可求兩直線的交點座標。例如：

$$L_1 : \frac{x+1}{3} = \frac{y-1}{4} = \frac{z-1}{2} \text{ 與 } L_2 : \frac{x+2}{2} = \frac{y+1}{3} = \frac{z+3}{3} \text{ 的交點座標為 } (2, 5, 3)。$$

4. 於空間向量中，請寫一程式，可以輸入兩個向量，並求其內積、外積與所張平行四邊形面積。例如： $\vec{a} = (1, 2, 3)$ ， $\vec{b} = (3, -2, -1)$ ，則其內積為 -4 ，外積為 $\vec{a} \times \vec{b}$ 是 $(4, 10, -8)$ ，所張平行四邊形面積是 $6\sqrt{5} = 13.4$ 。

5. 於空間座標中，請寫一程式，可以輸入三角形的三頂點座標，並求其面積。例如：

$$A(2, 0, 0), B(0, 2, 0), C(0, 0, 1), \text{ 則其所圍三角形面積為 } \sqrt{6} = 2.44。$$

Chapter 3 課後習題

1. 請設計一程式，可以計算兩個以標準式 $(3+4i)$ 表示的複數的加、減、乘、除等運算。

假設 $P = a + bi$ ， $Q = c + di$

(1) $P + Q = (a + c) + (b + d)i$

(2) $P - Q = (a - c) + (b - d)i$

(3) $P \times Q = (ac - bd) + (ad + bc)i$

(4) $P / Q = ((ac + bd) + (bc - ad)i) / (a \times a + d \times d)$

2. 請設計一程式，可以計算兩個以極式 $(5 \angle 30^\circ)$ 表示的複數的加、減、乘、除等運算。

假設 $P = r_1 \angle \theta_1$ ， $Q = r_2 \angle \theta_2$

(1) $P + Q$ 與 $P - Q$ 請先將極式轉為標準式，再相加減，最後再轉為極式。

(2) $P \times Q = (r_1 \times r_2) \angle (\theta_1 + \theta_2)$

(3) $P / Q = (r_1 / r_2) \angle (\theta_1 - \theta_2)$

3. 寫一個程式，可以輸入一個三元一次方程式，並求其解（本例假設所輸入的方程式恰有一解）。

! 提示：解三元一次方程式的演算法如下：

(1) 設二元一次方程式如下：

$$a_1x + b_1y + c_1z = d_1, \quad a_2x + b_2y + c_2z = d_2, \quad a_3x + b_3y + c_3z = d_3。$$

(2) 令 $f = \begin{vmatrix} a_1 & a_2 & a_3 \\ b_1 & b_2 & b_3 \\ c_1 & c_2 & c_3 \end{vmatrix} = a_1b_2c_3 + a_2b_3c_1 + a_3b_1c_2 - a_3b_2c_1 - b_3c_2a_1 - c_3b_1a_2。$

(3) 則其解分別是 $x = \frac{\begin{vmatrix} d_1 & b_1 & c_1 \\ d_2 & b_2 & c_2 \\ d_3 & b_3 & c_3 \end{vmatrix}}{f}$ ， $y = \frac{\begin{vmatrix} a_1 & d_1 & c_1 \\ a_2 & d_2 & c_2 \\ a_3 & d_3 & c_3 \end{vmatrix}}{f}$ ， $z = \frac{\begin{vmatrix} a_1 & b_1 & d_1 \\ a_2 & b_2 & d_2 \\ a_3 & b_3 & d_3 \end{vmatrix}}{f}。$

(4) 例如： $3x + y + z = 5$ ， $2x - 2y + z = 1$ ， $x - y + 2z = 2$

則其解為 $x = 1$ ， $y = 1$ ， $z = 1$ ，此稱為克拉馬公式。

4. 請分四次輸入 1 個 0 到 9 的整數，並將它合併為 1 個整數。例如：輸入 1，輸入 2，輸入 3，輸入 4，則輸出 1234。

Chapter 3 課後習題

- 同上習題，請先輸入 3 個 0 到 9 的整數，再輸入兩個 0 到 9 的整數，並將其合併為 1 個浮點數。例如：輸入 1，輸入 2，輸入 3，輸入 4，輸入 5，則輸出 123.45。
- 請輸入 1 個 4 位數，並將其分解輸出如 $a_1 \times 10^3 + a_2 \times 10^2 + a_3 \times 10^1 + a_4$ 。例如：輸入 1234，則輸出 $1 \times 10^3 + 2 \times 10^2 + 3 \times 10^1 + 4$ 。
- 請輸入一個 0 到 15 的整數，並將其轉為 2 進位輸出。例如：輸入 12，則輸出 $(12)_{10} = (1100)_2$ 。（請不要用迴圈與陣列）
- ※ 低通濾波器電路， $R = 1600\Omega$ ， $C = 0.000001F$ ， $f = 1000Hz$ ， $Z = -20.089dB$ ， $\theta = -84.319$ ，請寫一個程式，計算電壓增益。

演算法則

$$\frac{v_0}{v_1} = \frac{1}{1+j2\pi fRC} = \frac{1-j2\pi fRC}{1+(2\pi fRC)^2} = \frac{1}{1+(2\pi fRC)^2} - j \frac{2\pi fRC}{1+(2\pi fRC)^2}$$

$$\text{令 } a = 2\pi fRC, \text{ 則上式簡化為: } \frac{1}{1+a^2} - j \frac{a}{1+a^2}$$

$$\text{再令 } b = \frac{1}{1+a^2}, d = \frac{a}{1+a^2}, \text{ 則上式簡化為: } b - jd$$

$$\text{上式的大小為: } g = \sqrt{b^2 + d^2}$$

$$\text{電壓增益的大小} = 20 \log_{10} \left| \frac{V_0}{V_1} \right| = 20 \log_{10} g$$

$$\text{電壓增益相角} = \tan^{-1} \frac{-d}{b}$$

Chapter 3 課後習題

- ※ 9. RLC 串聯電路， $R = 300\Omega$ ， $L = 10\text{mH}$ ， $C = 0.1\mu\text{F}$ ， $V_s = 2.8\sin(2\pi f t)$ ， $\pi = 3.14$ ， $f = 9140\text{Hz}$ ，請寫一程式，計算 \vec{Z} 、 \vec{I} 與 \vec{V}_L 。答案既然是向量，請含大小與角度。

演算法則

$$V_p = 2.8 \quad (V_p = \text{峰值}) \quad , \quad R = 300 \quad , \quad L = 10\text{m} \quad , \quad C = 0.1\mu \quad , \quad \text{PI} = \text{Math.PI}$$

$$X_L = 2\pi f L \quad \text{且} \quad \vec{X}_L = X_L \angle 90^\circ \quad , \quad X_C = \frac{1}{2\pi f C} \quad \text{且} \quad \vec{X}_C = X_C \angle -90^\circ$$

$$\vec{Z} = \vec{R} + \vec{X}_L + \vec{X}_C = R \angle 0^\circ + X_L \angle 90^\circ + X_C \angle -90^\circ = R \angle 0^\circ + (X_L - X_C) \angle 90^\circ$$

$$\Rightarrow Z = \sqrt{R^2 + (X_L - X_C)^2} \quad , \quad Z_A = \tan^{-1}\left(\frac{X_L - X_C}{R}\right) \quad (\text{角度})$$

$$\vec{V} = \frac{V_p}{\sqrt{2}} \angle 0^\circ \Rightarrow V = \frac{V_p}{\sqrt{2}} \quad , \quad V_A = 0 \quad (\text{VA 是電壓角度})$$

$$\vec{I} = \frac{\vec{V}}{\vec{Z}} \quad \Rightarrow I = \frac{V}{Z} \quad , \quad I_A = V_A - Z_A \quad (\text{IA 是電流角度})$$

$$\vec{V}_L = \vec{I} \times X_L \Rightarrow V_L = I \times X_L \quad , \quad V_{L_A} = I_A + 90^\circ \quad (\text{V}_{L_A} \text{ 是電感角度})$$