

建立VB.NET應用程式

資科系
林偉川

1

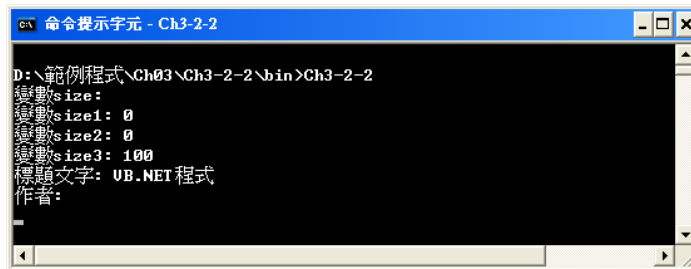
使用VB.NET開發應用程式

- VB.NET應用程式開發是使用Visual Studio.NET整合開發環境，其基本的步驟如下所示：
 - Step 1：新增專案。
 - Step 2：建立表單介面。
 - Step 3：設定控制項屬性。
 - Step 4：撰寫程式碼。
 - Step 5：編譯與執行。

2

第一個VB.NET主控台應用程式

- 主控台應用程式是早期BASICA、GWBASIC和QuickBasic在MS-DOS作業系統以文字模式執行的應用程式，在Windows作業系統是在「命令提示字元」視窗中執行，如下圖所示：



```
命令提示字元 - Ch3-2-2
D:\範例程式\Ch03\Ch3-2-2\bin>Ch3-2-2
變數 size:
變數 size1: 0
變數 size2: 0
變數 size3: 100
標題文字: VB.NET 程式
作者:
```

3

第一個VB.NET Windows應用程式

- Windows應用程式是在Windows作業系統下執行的圖形介面應用程式，例如：Word、記事本、小畫家或Visual Studio.NET本身都屬於Windows應用程式。
- Windows應用程式的執行需視使用者的操作而定，也就是依觸發的事件來執行適當的處理，稱為事件驅動程式設計。

4

第一個VB.NET Windows應用程式

- 步驟1：新增VB.NET的Windows應用程式專案
- 步驟2：在表單新增控制項
- 步驟3：設定控制項屬性
- 步驟4：新增其它的控制項
- 步驟5：在控制項新增事件處理程序
- 步驟6：編譯與執行Windows應用程式

5

VB.NET的程式架構

- VB.NET應用程式的程式碼是儲存在附檔名.vb的檔案，主要分為兩種：標準模組(Modules)和類別(Classes)檔，VB.NET程式碼就是置於這兩種檔案中。
- 在Visual Studio.NET的VB.NET專案所建立的主控台與Windows應用程式：
 - **主控台應用程式**：一個擁有Sub Main主程式的標準模組。
 - **Windows應用程式**：表單類別。

6

VB.NET的程式架構-標準模組

- VB.NET的標準模組是**延續舊版VB的程式架構**，主控台應用程式的程式碼都是位於標準模組中，在新增VB.NET專案時，預設的檔案名稱為**Module1.vb**，其程式架構如下所示：

```
Module Module1
    Sub Main()
        .....
    End Sub
End Module
```

7

VB.NET的程式架構

- 類別是用來建立物件(Object)，屬於物件導向程式設計的觀念，只需將類別視為Windows應用程式架構，在新增VB.NET專案時，預設的檔案名稱是Form1.vb，如下所示：

```
Public Class Form1
    Inherits System.Windows.Forms.Form
    .....
    Private Sub Button1_Click()
    End Sub
End Class
```

8

輸出與輸入

- 主控台應用程式和Windows應用程式最明顯的差異，就是使用介面的資料輸出與輸入：
 - 主控台應用程式：輸出與輸入是在命令提示字元輸入和輸出的文字內容。
 - Windows應用程式：表單控制項。

9

輸出與輸入-主控台

- 主控台應用程式的輸出與輸入是使用System.Console類別的方法，在程式碼中可以省略System，它是主控台應用程式的標準輸入、輸出。

方法	說明
<code>Console.Write(String)</code>	將字串 String 送到標準輸出，也就是「命令提示字元」視窗
<code>Console.WriteLine(String)</code>	同 Write 方法，只是在最後加上換行符號
<code>Console.Read()</code>	從標準輸入讀取一個字元直到使用者按下 <code>Enter</code> 鍵，傳回值是整數
<code>Console.ReadLine()</code>	類似 Read 方法，只是從標準輸入讀取整個字串

VB.NET程式的程式敘述

- **程式敘述(Statements)**：VB.NET程式是由程式敘述所組成，一系列程式敘述如同英文的一個句子，內含多個運算式、運算子或關鍵字，如下所示：

```
Dim balance As Integer = 1000
interest = balance * rate
Console.WriteLine("第一個VB.NET應用程式")
```

11

VB.NET程式的程式區塊

- **程式區塊 (Blocks)**：程式區塊是由多個程式敘述組成，它是位在Sub/End Sub、Function/End Function和If/End If等擁有End關鍵字間的程式碼，如下所示：

```
Sub Main

End Sub
```

12

VB.NET程式的End

- **End程式敘述結束程式**：VB.NET應用程式的進入點是Sub Main或啟動物件，如果以程式碼結束VB.NET程式是使用End程式敘述，當程式執行到此指令，就會馬上結束應用程式的執行，如下所示：

```
Private Sub Button2_Click(ByVal sender As
    System.Object, ByVal e _
        As System.EventArgs) Handles
    Button1.Click
    End
End Sub
```

13

VB.NET程式的程式註解

- **程式註解 (Comments)**：程式註解是程式設計上很重要的部分，良好的註解不但能夠輕易了解程式的目的，在維護上也可以提供更多的資訊，VB.NET程式的註解是以**REM指令**或**'符號**開始的列，或程式列上此符號之後的內容，如下所示：

```
REM 變數的宣告
' 大家好
Dim size1,size2 As Integer ' 變數的宣告
```

14

VB.NET程式之太長的程式碼

- **太長的程式碼**：如果VB.NET的程式碼太長，基於程式編排的需要，我們可以將它分為兩列，此時在第一列程式碼的**最後加上"_"符號**，如下所示：

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

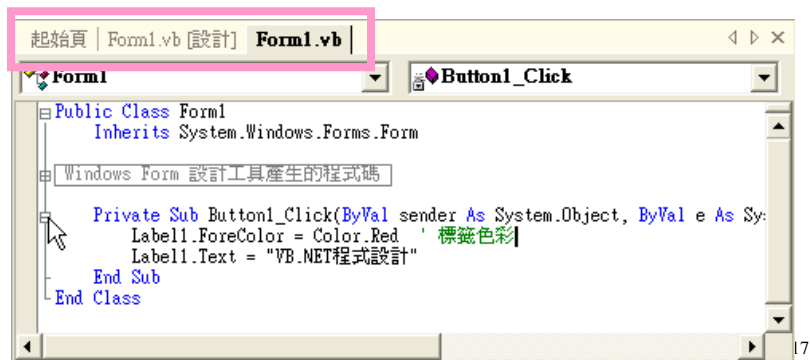
15

VB.NET程式的列印和顯示常數

VBA 常數	VB.NET 的列舉常數	說明
vbBack	ControlChars.Back	退格鍵(Backspace)
vbTab	ControlChars.Tab	定位字元(Tab)
vbCr	ControlChars.Cr	換行字元(Carriage Return)
vbLf	ControlChars.Lf	復行字元(Line Feed)
vbCrLf	ControlChars.CrLf	即 vbCr+vbLf
vbNewLine	ControlChars.NewLine	和 vbCrLf 擁有相同功能
vbNullChar	ControlChars.NullChar	0 值的字元

切換檔案與輸入程式碼

- 在編輯視窗輸入的程式碼將自動縮排和使用不同色彩標示，在上方的檔名標籤可以切換編輯的程式碼檔案，如下圖所示：

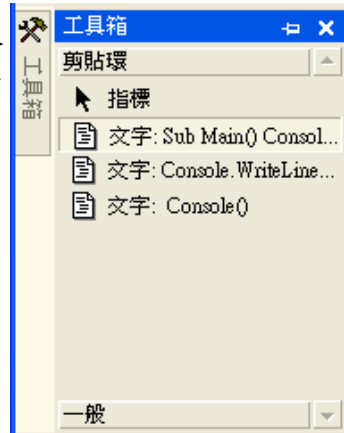


IntelliSense智慧程式碼輸入

- VB.NET的程式碼編輯視窗提供智慧指令提示，可以在輸入程式碼時，顯示物件屬性、函數語法等提示訊息。
 - 物件**屬性**或**方法清單**
 - **函數**或**方法**的使用提示
 - 自動完成指令

剪貼環功能

- 在「工具箱」視窗【剪貼環】標籤的功能類似剪貼簿，對於在程式碼編輯視窗複製、剪下的文字，都會保留在此標籤，如右圖所示：



19

執行VB.NET應用程式

- 在Visual Studio.NET建置和編譯VB.NET專案後，在各VB.NET專案資料夾的bin子資料夾會有此專案編譯成的執行檔，副檔名為.exe，我們可以直接在Windows作業系統執行EXE執行檔。

－執行主控台應用程式

- 在「命令提示字元」視窗執行EXE執行檔

－執行Windows應用程式

- 請執行「開始/執行」指令，在「執行」對話方塊選取執行的EXE執行檔。

20

列印程式碼

- 如果需要繳交作業或進程式碼除錯，或是研究Visual Studio.NET自動產生的控制項程式碼，可以將程式碼列印成報表。
- Visual Studio.NET提供列印功能可以執行「檔案/列印」指令，列印程式碼編輯視窗中展開的程式碼(隱藏的程式碼區塊並不會列印)。