

VB.NET語法建立ASP.NET程式

資科系
林偉川

物件的基本觀念

- ASP.NET是一種伺服器端網頁技術，本身並沒有專屬的程式語法，預設是使用VB.NET語法，VB.NET語言是一種支援.NET Framework的物件導向程式語言。
- ASP.NET主要是使用VB.NET語法和.NET Framework類別的物件，就算讀者不熟悉物件導向程式設計也沒有關係，因為只需了解物件的基本觀念和如何使用。

物件的基本觀念-物件

- 物件為物件導向程式的基礎，簡單的說物件是資料（**Data**）和包含處理資料相關函數（稱為方法）的綜合體，而「類別」（**Class**）是可以用來定義物件和建立物件，在建立好物件後，並不用考慮物件內部處理的程式碼，只需知道物件提供的屬性和方法和如何使用，就可以使用這些物件。
- 例如：ASP.NET可以使用.NET Framework的類別物件提供的**HTTP物件**：Response、Server、Request、Session和Application等物件。

3

物件的基本觀念-屬性

- 物件的屬性是物件的性質和狀態，例如：
 - Response物件的ContentType屬性。
 - Response物件的BufferOutput屬性。

Response.ContentType
Response.BufferOutput

4

物件的基本觀念-方法

- 方法就是物件的處理函數，例如：
Response物件擁有下列方法，如下所示：
`Response.Write()`
`Response.Redirect()`

5

物件的基本觀念-事件

- 事件是指物件回應的事件，例如：**Page**物件的**Load**事件，我們可以建立處理此事件的程序，如下所示：
`Sub Page_Load(Sender As Object, e As EventArgs)`
.....
`End Sub`

6

物件的基本觀念-集合物件

- 集合物件如同其名是一組物件的集合，例如：**Response**物件的集合物件**Cookies**（通常是使用指定屬性取得集合物件），如下所示：

`Response.Cookies("name").Value`

- 程式碼取出集合物件中名為name的Cookie物件，然後使用Value屬性取得其值，事實上，**Application**和**Session**物件都屬於一種集合物件。

7

ASP.NET程式的基本架構

- 在舊版ASP 3.0版使用的程式語法是VBScript或JavaScript等腳本語言，ASP.NET使用的是.NET語言C#、VB.NET或Jscript.NET等，以本書為例是使用「VB.NET」。
- VB.NET是Visual Basic 6的下一個版本，和前一個版本的主要差異在VB.NET是支援.NET Framework的物件導向程式語言。

8

微軟的BASIC家族

- 目前微軟Basic家族包含Visual Basic 6到VB.NET，Office家族提供VBA，客戶端有VBScript。伺服器端VBScript語法的ASP和VB.NET語法的ASP.NET。

9

建立VB.NET語法的ASP.NET程式

- ASP.NET程式的附檔名為.aspx，程式碼直接內嵌於HTML標籤，置於<%...%>符號間，如下：

```
08: <%  
09: ' 變數宣告  
10: Dim beginSize, endSize, i As Integer  
11: beginSize = 4  
12: endSize = 6  
13: ' 設定字型尺寸由小變大  
14: For i = beginSize To endSize  
15: %>  
16: <font size=<%=i%>>  
17: <p>ASP.NET網頁設計範例教本</p></font>  
18: <%  
19: Next i  
20: %>
```

10

建立VB.NET語法的ASP.NET程式-Page指引

- 在ASP.NET程式開頭的第2列指出使用的程式語言，如下所示：

```
02: <%@ Page Language="VB" %>
```

- @ Page是ASP.NET程式的「指引」(Directive)，Language「屬性」(Attribute)定義ASP.NET程式使用的程式語言，以此例為VB，表示使用VB.NET，因為預設值是VB，如果使用VB.NET，也可以省略這一系列程式碼。

11

建立VB.NET語法的ASP.NET程式- 變數宣告和迴圈開始

- 變數宣告和迴圈開始：第8~15列的ASP.NET程式整段程式碼並沒有任何HTML標籤，所以只需在開始和結束加上<%和%>符號，如下所示：

```
08: <%
```

```
09: ' 變數宣告
```

```
10: Dim beginSize, endSize, i As Integer
```

```
11: beginSize = 4
```

```
12: endSize = 6
```

```
13: ' 設定字型尺寸由小變大
```

```
14: For i = beginSize To endSize
```

```
15: %>
```

12

建立VB.NET語法的ASP.NET程式- 內含於HTML標籤

- 內含於HTML標籤：第16列的標籤指定字型尺寸，在標籤屬性size加上ASP.NET程式碼，這個屬性屬於動態變數，隨著迴圈產生不同的字型尺寸，同樣是位在<%和%>符號間，如下所示：

16: <font size=<%=i%>>

<%= i%>寫法相當於Response.write()方法的另一種輸出

13

建立VB.NET語法的ASP.NET程式- 迴圈結束

- 迴圈結束：最後第18~20列屬於迴圈結束的程式碼，因為前後都有不同的HTML標籤，這列程式碼也需要使用<%和%>符號標示出來：

18: <%

19: Next i

20: %>

14

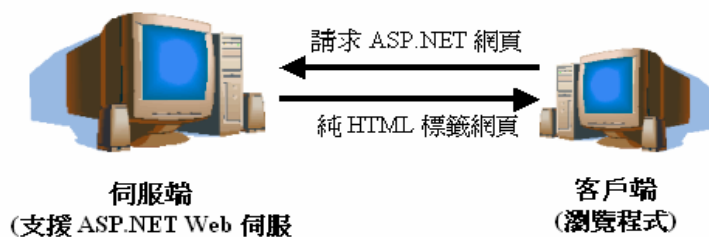
儲存成ASP.NET程式

- ASP.NET程式檔案內容也擁有HTML標籤，只是其副檔名為【.aspx】，如果使用【記事本】儲存檔案時，預設儲存的副檔名為【.txt】

15

ASP.NET程式的執行過程

- ASP.NET程式為伺服器端網頁技術，我們需要將程式檔案公佈到Web伺服器後才能執行ASP.NET程式，其執行的過程，如下圖所示：



16

ASP.NET程式的執行過程-執行

- 執行ASP.NET程式需要將ASP.NET程式檔案公佈或複製到IIS主目錄【C:\inetpub\wwwroot】，也可以建立成虛擬目錄。
- 啓動Internet Explorer瀏覽程式，在【網址】欄輸入URL網址，以此例如下所示：
http://domain_name/Ch03/Ch3-2-2.aspx
- domain_name爲網站的網域名稱，如果伺服器位在同一台電腦，可以使用localhost，如下：
<http://localhost/Ch03/Ch3-2-2.aspx>

17

ASP.NET程式的撰寫風格

- ASP.NET程式的撰寫風格即VB.NET語法的撰寫風格，能夠讓程式更容易了解和維護，例如：程式碼縮排可以反應出程式碼的邏輯和迴圈架構，註解文字可以讓程式更容易明白其目的。

18

使用註解文字

- 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，而且在維護上也可以提供更多的資訊，VB.NET的程式註解是以「'」符號或Rem指令開始的列，或程式列之後此符號後的文字內容，如下所示：

' 註解文字

Rem 註解文字

Response.Write("<h2>大家好!</h2>") ' 使用Response物件

19

太長的程式碼

- 如果程式碼的長度太長，基於程式編排的需要可以將它分成兩列，此時請在第一列程式碼的最後加上「& _」符號，如下所示：

Response.Write("<h2>大家好!這是比較長的程式碼," & _
"所以需要分爲兩列.</h2>")

20

VB.NET變數的資料型態

- VB.NET的變數並沒有強調一定需要宣告其資料型態，**如果沒有宣告資料型態，預設的資料型態是Object**，Object是所有VB.NET資料型態的基礎資料型態。VB.NET的資料型態，如下：

資料型態	說明	位元組	範圍
Boolean	布林值	2	True 或 False
Byte	正整數	1	0~255
Char	字元	2	0~65535
Short	短整數	2	-32,768~32,767
Integer	整數	4	-2,147,483,648~2,147,483,647

21

VB.NET變數的資料型態

Long	長整數	8	-9,223,372,036,854,775,808~9,223,372,036,854,775,807
Single	單精度的浮點數	4	負值範圍為 -3.4028235E+38~1.401298E-45，正值的範圍為 1.401298E-45~3.4028235E+38
Double	雙精度的浮點數	8	負值範圍為 -1.79769313486231570E+308~4.94065645841246544E-324，正值範圍為 4.94065645841246544E-324~1.79769313486231570E+308

22

VB.NET變數的資料型態

Decimal	數值	16	0~+/- 79,228,162,514,264,337,593,543,950,335 沒有小數，0~+/- 7.9228162514264337593543950335 帶 28 位小數，最小的非零值為 +/- 0.00000000000000000000000000000001(+/- 1E-28)
Date	日期	8	0001 年 1 月 1 日~9999 年 12 月 31 日
String	字串	依平台	0~2 百萬 Unicode 字元
Object	物件	4	物件型態變數可以儲存各種資料型態 的值

23

VB.NET變數的宣告

- VB.NET的命名原則，如下所示：
 - 不可以使用VB.NET的保留字或系統的物件名稱。
 - 必需是英文字母或底線「_」開頭，如果以底線開頭，至少需要一個英文字母或數字。
 - 名稱長度不可超過16383個字元，不區分英文大小寫，abc和ABC代表同一個名稱。
 - 名稱中間不能有句點「.」或空白，只能是英文字母、數字和底線。
 - 在宣告的有效範圍內需唯一。

24

變數的宣告-範例

- VB.NET元素名稱的範例，如下所示：

abc, ab_123, size1, _123, _abc ' 合法名稱

_, 123abc ' 不合法名稱，只有底線或數字開頭

25

變數的宣告-宣告變數

- 變數在程式碼扮演的角色是用來儲存程式執行中的一些暫存資料，在VB.NET程式碼是使用【Dim】指令宣告所需的變數，其指令如下：

`Dim balance` ' 變數balance的資料型態是Object

- 程式碼宣告一個變數，因為沒有指定資料型態，預設的資料型態為Object，可以儲存各種資料型態的值。如果宣告的變數需要指定型態，我們需要使用As指令宣告變數的資料型態，如下：

`Dim size1,size2 As Integer` ' 變數都是Integer資料型態

26

變數的宣告-指定初值

- 如果同一列的2個變數屬於不同資料型態，我們可以分別宣告其資料型態，如下所示：

```
Dim name As String,size3 As Integer
```

- 如果需要，還可以在宣告時同時指定變數的初值，如下所示：

```
Dim name As String ,size3 As Integer = 100
```

27

指定敘述

- 在VB.NET宣告變數後，因為變數是用來儲存資料，所以需要指定或更改變數值，使用的是指定敘述「=」等號，例如：

```
balance = 1000
```

```
size1 = 500
```

```
size2 = 400
```

```
name = "陳會安"
```

- 變數balance是Object資料型態，可以儲存各種資料型態的資料，可以再次在程式碼使用指定敘述設定變數成其它值，如下所示：

```
balance = "1000"
```

28

常數的宣告與使用

- 程式碼的**常數**是使用一個**常數名稱**取代**固定的數字或字串**，VB.NET本身已經擁有一些**內建常數**，我們也可以自行建立所需的常數，如下所示：

```
Const PI = 3.1415926
```

- 在VB.NET程式碼就可以直接使用這個常數計算圓面積。

29

一維陣列-宣告

- 一維陣列對比前述的信箱就是單排信箱，VB.NET的陣列通常是固定大小的陣列，也就是在宣告時需要指定陣列大小，宣告陣列的語法，如下所示：

```
Dim array_name(size-1) As type_name
```

- 語法的array_name為**陣列名稱**、size是**陣列大小**、type_name是**資料型態**，括號是最大索引值，**需要減一**，例如：宣告字串陣列name，如下：

```
Dim name(4) As String
```
- 陣列的大小為**5**，因為從**0**起算所以括號值為**4**。

30

一維陣列-存取

- 使用指定敘述指定陣列值，其程式碼如下所示：
`name(0) = "陳會安"`
`name(1) = "江小魚"`
`name(2) = "張無忌"`
`name(3) = "楊過"`
`name(4) = "小龍女"`
- 程式碼可以看出使用名稱加上索引值來指定陣列元素，至於取出陣列值的程式碼，如下所示：
`userName = name(3)`

31

多維陣列

- 多維陣列擁有多個索引，對比前述的信箱就是多排信箱，二維陣列擁有2個索引，三維陣列有3個，依序類推。
- 只需是表格編排的資料，我們就可以建立多維陣列來儲存這些資料，例如：大樓住戶名稱和管理費的二維陣列，其宣告如下所示：

`Dim home(2,1) As String`

home(0,0)	home(0,1)
home(1,0)	home(1,1)
home(2,0)	home(2,1)

32

動態陣列

- 動態陣列是指陣列大小允許在程式執行時變更其大小，即動態更改陣列的尺寸。

33

動態陣列-宣告

- VB.NET的陣列可以使用【ReDim】指令在程式執行時重新更改陣列大小，不過ReDim並不能宣告陣列，只能更改現存的陣列大小，如下：
ReDim name(2)
- 程式碼使用ReDim指令變更陣列大小為3，現在陣列擁有3個元素，如果陣列內已經儲存內容，為了保留原陣列的內容，需要使用**Preserve**指令，其宣告如下所示：
ReDim Preserve name(4)

34

動態陣列-相關函數

- 在動態陣列的處理上，在程式更改大小後，我們需要取得目前**陣列索引**的範圍，此時可以使用函數取得陣列的索引範圍，相關函數說明，如下所示：
 - **Lbound()**函數：取得陣列的最小索引，變數 **size** 的值是陣列的最小索引，如右所示：**size = Lbound(name)**
 - **Ubound()**函數：取得陣列的最大索引，變數 **size** 的值是陣列的最大索引，如右所示：**size = Ubound(name)**

35

VB.NET的運算子

- 在VB.NET指定敘述的等號右邊若為運算式或是條件運算式，這都是使用「運算子」(Operator)和「運算元」(Operand)所組成，VB.NET擁有完整的算術 (Arithmetic)、關係 (Relational)、連結 (Concatenation) 和邏輯 (Logical) 運算子。一些範例如下所示：
 - A + B - 1**
 - A >= B**
 - A > B And A > 1**
- 運算式變數**A**、**B**和數值**1**都屬於運算元，「+」、「-」、「>=」和「>」為運算子。

36

運算子的優先順序

- 如果運算子沒有優先順序的差異，運算式由左到右依序的執行。
- "("括號內比括號外先執行。
- 當運算式擁有多種運算子時，**算術運算子最先**，接著**關係運算子**，最後是**邏輯運算子**。
- **關係運算子擁有相同的優先順序**。
- 算術運算子的優先順序，如下所示：
^指數 > (-)負號 > */乘除 > \整數除法 > Mod > +-加減 > &
- 邏輯運算子的優先順序，如下所示：
Not > And, AndAlso > Or, OrElse > Xor

37

算術運算子

運算子	說明	運算式範例
^	指數	$5^2 = 25$
-	負號	-7
*	乘法	$5 * 6 = 30$
/	除法	$7 / 2 = 3.5$
\	整數除法	$7 \setminus 2 = 3$
Mod	餘數	$7 \bmod 2 = 1$
+	加法	$4 + 3 = 7$
-	減法	$4 - 3 = 1$

8

字串連結運算子

- VB.NET字串資料型態的變數可以使用字串連結運算子「&」來連結，其優先順序是在算術運算子之後，但在關係運算子之前：

運算子	說明	運算式範例
&	字串連結	"ab" & "cd" = "abcd" "ASP.NET 網頁" & "設計" = "ASP.NET 網頁設計"

39

關係運算子

運算子	說明	運算式範例	結果
=	等於	6 = 3	False
<>	不等於	6 <> 3	True
<	小於	6 < 3	False
>	大於	6 > 3	True
<=	小於等於	6 <= 3	False
>=	大於等於	6 >= 3	True
Is	物件比較	比較兩個變數是否指向同一個物件	

40

邏輯運算子-說明

運算子	說明
Not	非，傳回運算元相反的值
And	且，連結的兩個運算元都為 True，則運算式為 True
Or	或，連結的 2 個運算元中，任一個為 True，則運算式為 True
Xor	連結的 2 個運算元中，只有一個為真，不同時都為 True 則運算式為 True，否則為 False
AndAlso	2 個運算元都為 True 才為 True，如果第 1 個運算元已經 False，就不用計算第 2 個運算元，傳回 False
OrElse	連結的 2 個運算元中，任一個為 True，則運算式為 True，否則為 False，如果第一個為 True，就不用計算第 2 個運算元，傳回 True

41

邏輯運算子-範例

邏輯運算子	完整的運算式	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
A Xor B	5 > 3 Xor 4 <= 2	True
A Eqv B	5 > 3 Eqv 4 <= 2	False
A AndAlso B	5 > 3 AndAlso 4 <= 2	False
A OrElse B	5 > 3 OrElse 4 <= 2	True，以此例第二個運算元根本不用計算

42

邏輯運算子-真假值表

A	B	Not A	Not B	A And B	A Or B	A Xor B
True	True	False	False	True	True	False
True	False	False	True	False	True	True
False	True	True	False	False	True	True
False	False	True	True	False	False	False

43

指定運算子

運算子	範例	相當的運算式	說明
=	x = y	N/A	指定敘述
^=	x ^= y	x = x ^ y	指數
+=	x += y	x = x + y	數字相加
-=	x -= y	x = x - y	減法
*=	x *= y	x = x * y	乘法
/=	x /= y	x = x / y	除法
\=	x \= y	x = x \ y	整數除法
&=	x &= y	x = x & y	字串連結

44