

新觀念的 Visual Basic.NET 教本

第 1 章 認識 VB.NET

1-1 VB.NET 簡介

VB.NET 簡介

- VB 的起源。
- VB 的演進。
- VB 的特色。
- VB.NET 的誕生。

VB.NET 的幾個版本

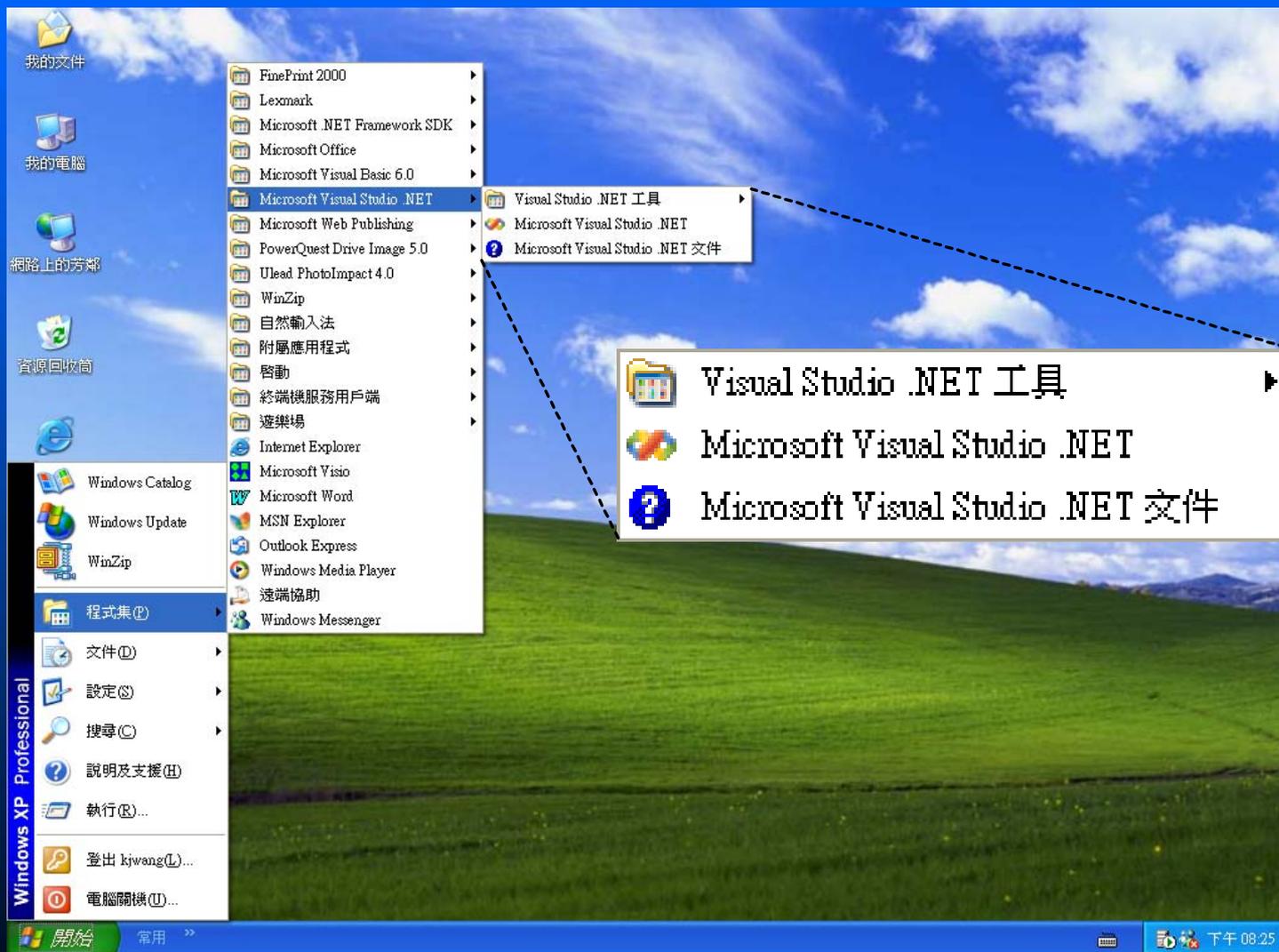
- Visual Basic.NET Standard(標準版)。
- Visual Studio Professional(專業版)。
- Visual Studio .NET Enterprise Developer(企業版)。
- Visual Studio .NET Enterprise Architect(企業建築家版)。

安裝 VB.NET 的系統需求

- CPU
- 作業系統
- 硬碟空間
- 記憶體

1-2 進入VB.NET

Visual Studio.NET 群組 (1)



Visual Studio.NET 群組 (2)

The screenshot shows the Visual Studio .NET help application. The title bar reads "Visual Studio .NET Combined Collection - Visual Basic 和 Visual C#". The menu bar includes "檔案(F)", "編輯(E)", "檢視(V)", "工具(T)", "視窗(W)", and "說明(H)". The address bar shows the URL "ms-help://MS.VSCC/MS.MSDNVS.1028/Vbcon/html".

The left sidebar shows a tree view of the help content. Under "Visual Studio .NET", there are items like "Visual Studio .NET 簡介", "取得協助", "範例和逐步解說", and "使用 Visual Studio .NET 開發". Under "Visual Basic 和 Visual C#", there are items like "快速入門", "升級應用程式", "管理專案", "建立應用程式", "存取資料", "元件程式設計", "參考", and "範例". Other categories include "Visual C++", "JScript", ".NET Framework", and "MSDN Library".

The main content area is titled "Visual Basic 和 Visual C#" and contains the following text:

Visual Basic 和 Visual C#

本章節的文件將討論 Visual Basic 或 Visual C# 中的應用程式開發。兩種程式語言都適用於本章節中所涵蓋的主題。由兩種中的任一種語言撰寫的應用程式都是建立在 Common Language Runtime 的服務上，且會充分利用 .NET Framework 類別庫 (Class Library)。

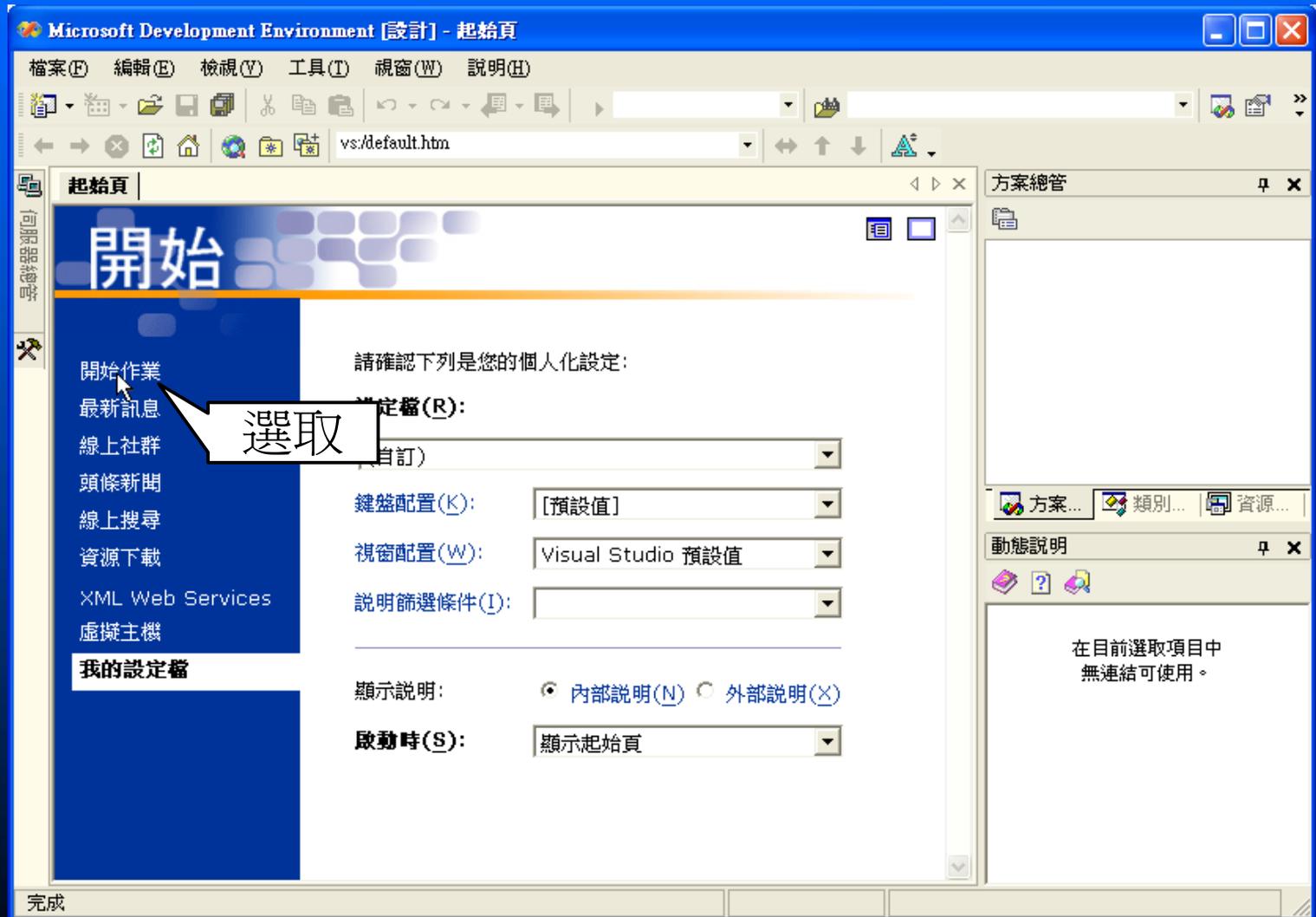
本章節內容

- [快速入門](#)
藉由列出新功能以及引導您瞭解版本升級問題和應用程式設計決定，來協助您開始工作。
- [存取資料](#)
討論如何將資料存取功能加入您的應用程式中。

Below the main content is a "搜尋結果" (Search Results) section with a table with columns "標題" (Title), "位置" (Location), and "順序" (Order).

A callout box with a white background and black border is overlaid on the "快速入門" link. It contains the text: "這些都是 VB.NET 提供的電子文件。" (These are the electronic files provided by VB.NET).

建立應用程式專案



上機：建立Windows應用程式專案

1. 在VB.NET的起始頁中，按下「新增專案」鈕 (或是選擇功能表的「檔案>>新增>>專案」)，接著會出現以下「新增專案」視窗，如圖-4。
2. 在「新增專案」視窗中，依序完成：
 - 1) 選取「專案類型」裡的「Visual Basic專案」。
 - 2) 選取「範本」裡的「Windows應用程式」。
 - 3) 在「名稱」欄位中輸入專案名稱(例如「第一章練習一」)。
 - 4) 在「位置」欄位中輸入資料夾位置(例如「C:\VB教本」)。
 - 5) 按下「確定」鈕。

上機：建立Windows應用程式專案

3. 稍候一下，VB.NET的開發環境中便會出現抬頭為“Form1”的表單視窗，如圖-6。
4. 最後讓我們儲存這個專案，並且回到起始頁。操作過程是：選取功能表的「檔案>>全部儲存」（作用是儲存專案之相關檔案），接著再選取功能表的「檔案>>關閉方案」（回到起始頁）。

上機：建立主控台應用程式專案

1. 按下起始頁的「新增專案」鈕，(或是選擇功能表的「檔案>>新增>>專案」)，接著在「新增專案」視窗中，依序完成：(參閱圖-7)
 - 1) 選取「專案類型」裡的「Visual Basic專案」。
 - 2) 選取「範本」裡的「主控台應用程式」。
 - 3) 在「名稱」欄位中輸入專案名稱(例如「第一章練習二」)。
 - 4) 在「位置」欄位中輸入資料夾位置(例如「C:\VB教本」)。
 - 5) 按下「確定」鈕。

上機：建立主控台應用程式專案

2. 稍候一下，VB.NET的開發環境中會出現抬頭為“Module1.vb”程式模組視窗，如圖-8。
3. 最後選取功能表的「檔案>>全部儲存」儲存專案，接著再選取「檔案>>關閉方案」關閉專案回到起始頁。

上機：開啓前面練習所建立的專案

1. 在起始頁中，按下「開啓專案」鈕(或是選擇功能表的「檔案>>開啓>>專案」)，接著會出現「開啓專案」視窗。
2. 在「開啓專案」視窗中，將資料夾設定於「C:\VB教本\第一章練習一」(設定方法之一是在「檔案名稱」欄位中輸入 C:\VB教本\第一章練習一，接著按下 Enter 鍵)，然後選取「第一章練習一」，按下「開啓」鈕，如圖-9。

上機：開啓前面練習所建立的專案

3. 接著我們可以看到VB.NET重新開啓了剛才所建立的「第一章練習一」專案，螢幕中央會出現抬頭為“Form1”視窗。(註：如果沒有看到這個視窗，可以雙按「方案總管」裡面的Form1.vb加以開啓，[如圖-10](#))
4. 接下來我們要開啓「第一章練習二」專案，首先選取功能表的「檔案>>關閉方案」關閉「第一章練習一」專案，然後再選取功能表的「檔案>>開啓>>專案」，接著會出現「開啓專案」視窗。

上機：開啓前面練習所建立的專案

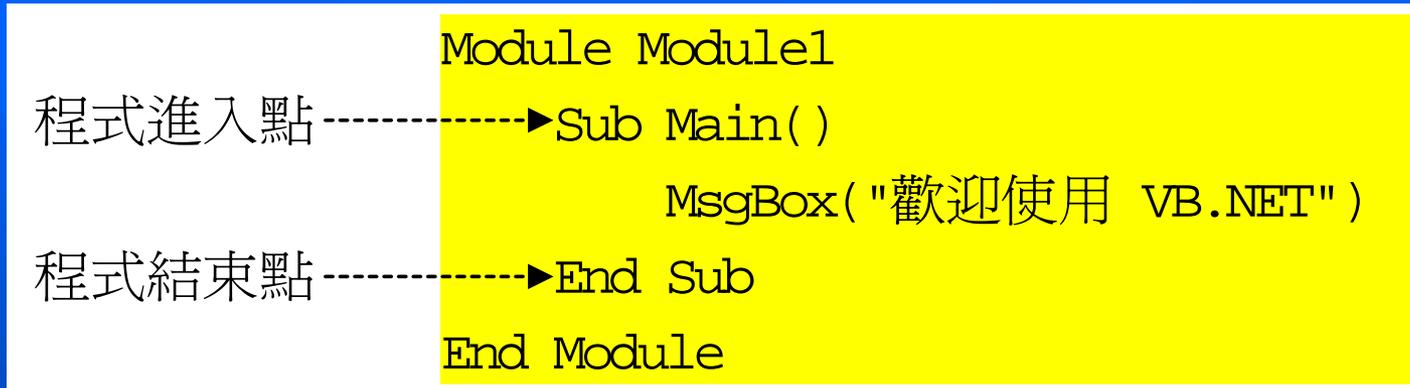
5. 在「開啓專案」視窗中，將資料夾設定於「C:\VB教本\第一章練習二」，然後選取「第一章練習二」，按下「開啓」鈕，[如圖-11](#)。
6. 接著我們可以看到VB.NET重新開啓了剛才所建立的「第一章練習二」專案，螢幕中央會出現抬頭為“Module1.vb”的視窗。(註：如果沒有看到這個視窗，可以雙按「方案總管」裡面的Module1.vb加以開啓，[如圖-12](#))。
7. 選取功能表的「檔案>>關閉方案」關閉「第一章練習二」專案，回到起始頁。

1-3 使用「主控台應用程式」

上機：輸入指令

- 目的：在「主控台應用程式」專案中輸入程式。
- 1. 建立「主控台應用程式」專案(過程如p.22 的上機)或是開啓先前所建立的「主控台應用程式」專案。
- 2. 在Module1.vb視窗(稱爲「程式模組」視窗)中，於Sub Main() 與End Sub之間輸入「MsgBox(”歡迎使用VB.NET”)」，如圖-13。
- 3. 按下 F5 執行程式，結果顯示一訊息窗。
- 4. 按下以上訊息窗的「確定」鈕即可結束主控台應用程式，回到VB.NET的開發環境。

上機：輸入指令



■ 程式解說：

- 以上程式啓動後，會從Sub Main() 開始執行，接下來執行我們所撰寫的指令(也就是MsgBox這一行)，而MsgBox這一行指令會顯示一個訊息窗(如上圖)，當我們按下這個訊息窗的「確定」鈕之後，該訊息窗被關閉，然後執行到End Sub，而End Sub是程式的結束點，執行至此即結束主控台應用程式。

上機：認識Console.WriteLine

1. 延續上一個專案，在MsgBox這行指令之前輸入一行程式，使得程式成爲：

```
Module Module1
```

```
    Sub Main()
```

```
        Console.WriteLine("歡迎使用 VB.NET")
```

```
        MsgBox("歡迎使用 VB.NET")
```

```
    End Sub
```

```
End Module
```

上機：認識Console.WriteLine

2. 按下 F5 執行程式，結果如圖-15。
3. 按下訊息窗的「確定」鈕結束程式，接著會回到VB.NET開發環境。接下來請修改程式 -- 將MsgBox這一行指令刪除，程式變成：

```
Module Module1
```

```
    Sub Main()
```

```
        Console.WriteLine("歡迎使用 VB.NET")
```

```
    End Sub
```

```
End Module
```

上機：認識 Console.WriteLine

4. 按下 F5 執行程式，結果只看到螢幕閃一下，主控台視窗就不見了，沒看到 Console.WriteLine 所顯示的訊息。
5. 讓我們繼續修改程式，在 Console.WriteLine 之後增加一行指令，使整個程式變成：

```
Module Module1
```

```
    Sub Main()
```

```
        Console.WriteLine("歡迎使用 VB.NET")
```

```
        Console.ReadLine()
```

```
    End Sub
```

```
End Module
```

上機：認識Console.WriteLine

6. 按下 F5 執行程式，結果又可以看到 Console.WriteLine 所顯示的內容了，[如圖-16](#)。
7. 請在主控台視窗中按下 Enter 鍵，接著會執行到 End Sub 結束程式。
 - 程式解說
 - 步驟4：為什麼沒有看到 Console.WriteLine 所顯示的訊息呢？因為程式執行 Console.WriteLine 之後，緊接著來到 End Sub，而這是程式的結束點。
 - 步驟5：Console.ReadLine 指令其作用是讓主控台視窗等待使用者輸入資料後按下 Enter 鍵。