

第七章 視窗應用程式開發



7.1 建立視窗應用程式專案

7.3 控制項編輯與命名

7.2 整合開發環境介紹

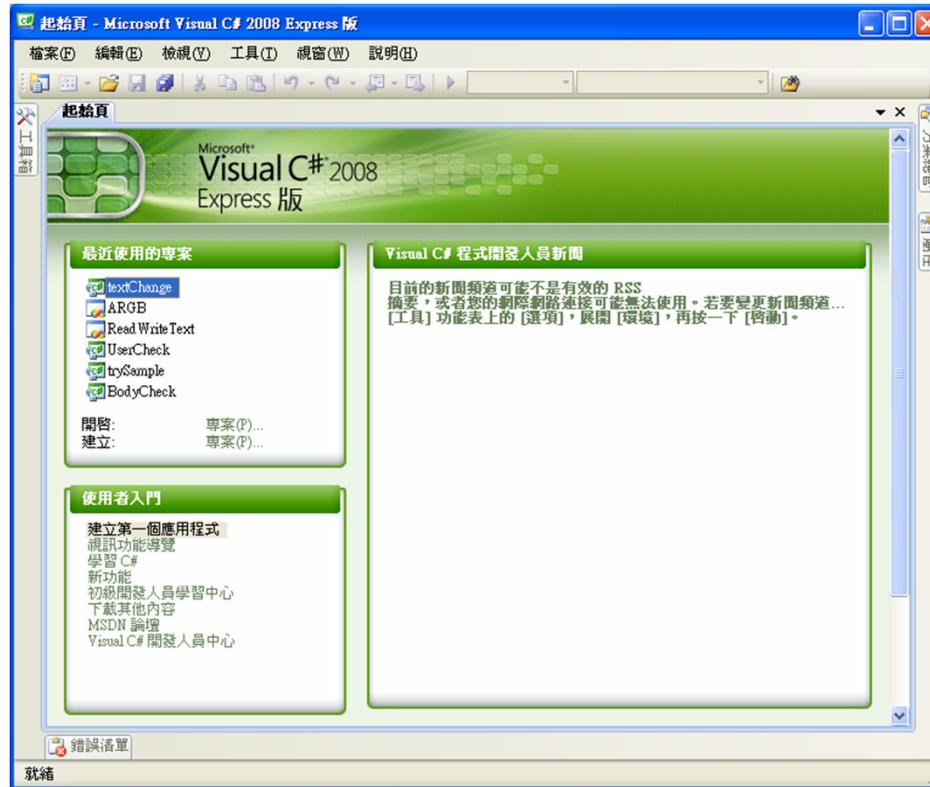
7.4 視窗應用程式的開發

備註：可依進度點選小節



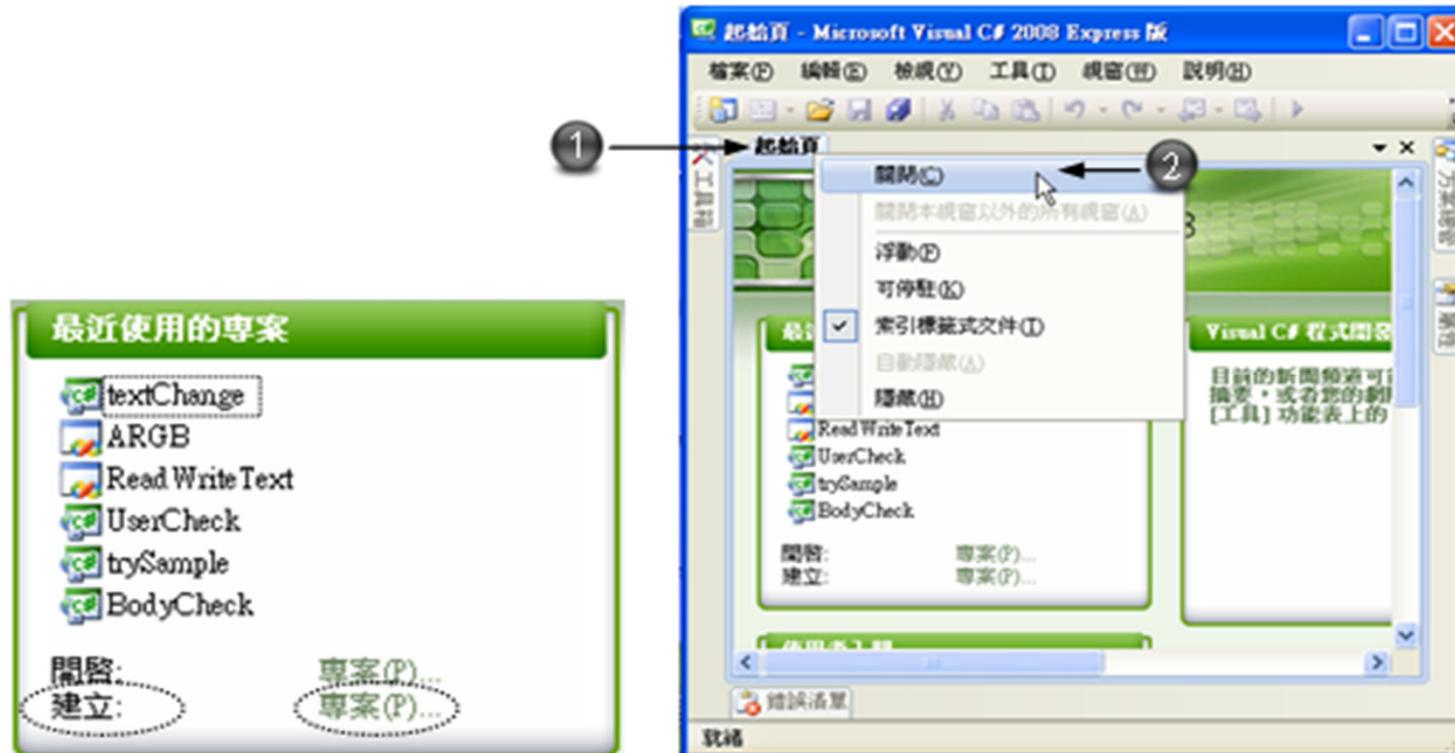
7.1 建立視窗應用程式專案

■ 執行【開始 / 所有程式(R) / Microsoft Visual C# 2008 Express Edition】會進入 Visual C# 2008 起始頁畫面。





若希望能顯示起始頁，執行功能表的【視窗(W)/重設視窗配置(R)】即可重新打開起始頁。其中「最近使用的專案」窗格用來顯示最近更新過的專案清單，讓您由此窗格直接建立新專案，或直接開啟舊有的專案。





7.1.1 如何新增專案

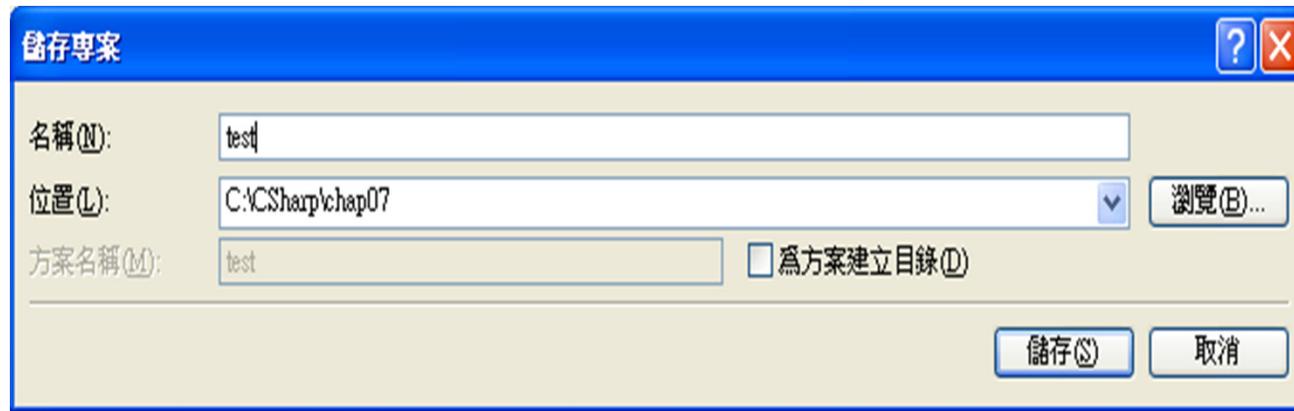
- 在左上圖「最近使用的專案」窗格中最後一行點選「建立：」的鈕，或是執行功能表【檔案(F)/新增專案(P)】指令，就會開啟「新增專案」對話方塊，供我們對新專案做相關的設定。





7.1.2 如何儲存專案

- 執行【檔案(F)/全部儲存(L)】或按 全部儲存圖示進行儲存專案，接著出現下圖「儲存專案」視窗，在「名稱(N)」請輸入指定的專案名稱是**test**；在「位置(L)」設定專案的儲存位置是「**C:\CSharp\chap07**」。
- 完成後按 **儲存(S)** 鈕儲存專案。路徑未建立，系統預設建在**C:\Documents and Settings\使用者名稱\My Documents\Visual Studio 2008\ Projects**的資料夾

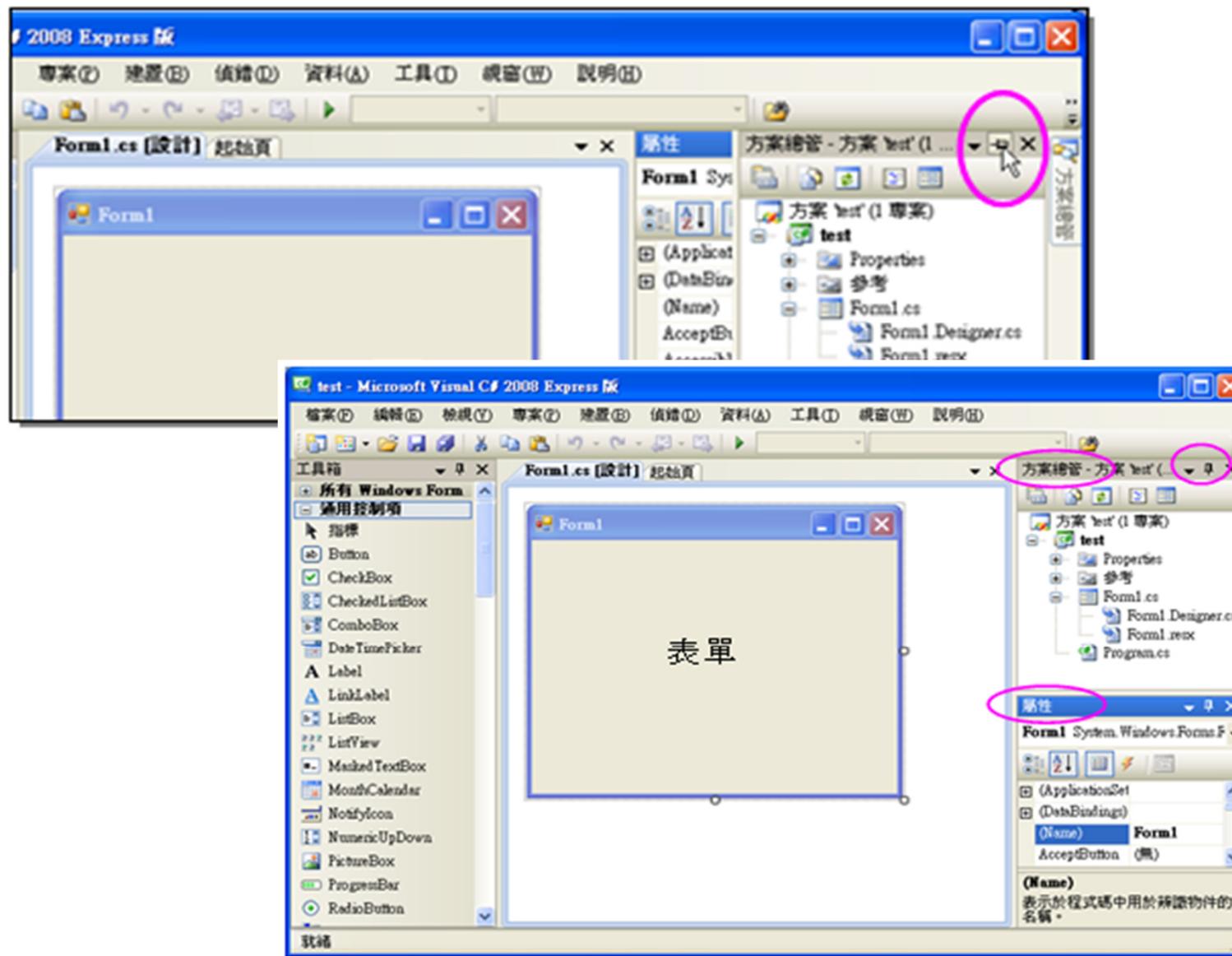




7.1.3 如何固定方案總管視窗

- 1. 若方案總管如下圖採彈跳式未固定，請移動滑鼠到右邊界「方案總管」處按一下，方案總管會由右邊界向左彈出。當滑鼠離開方案總管時又自動彈回隱藏起來。

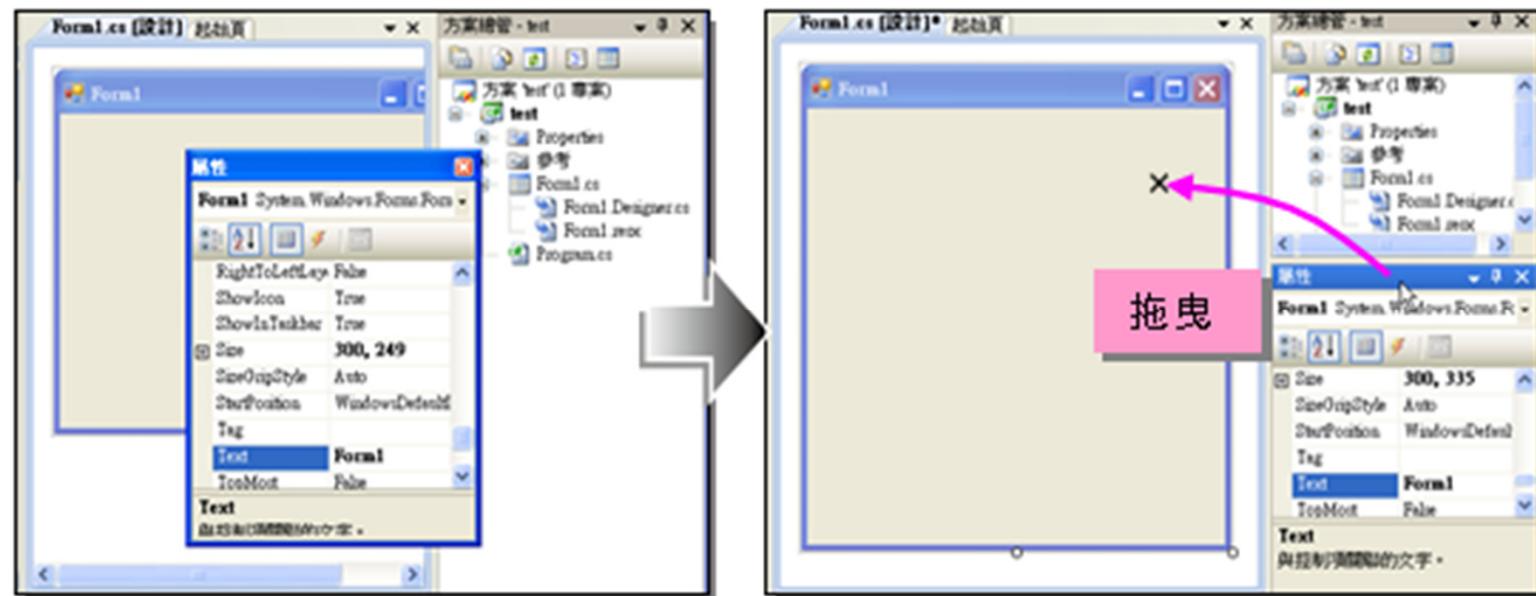






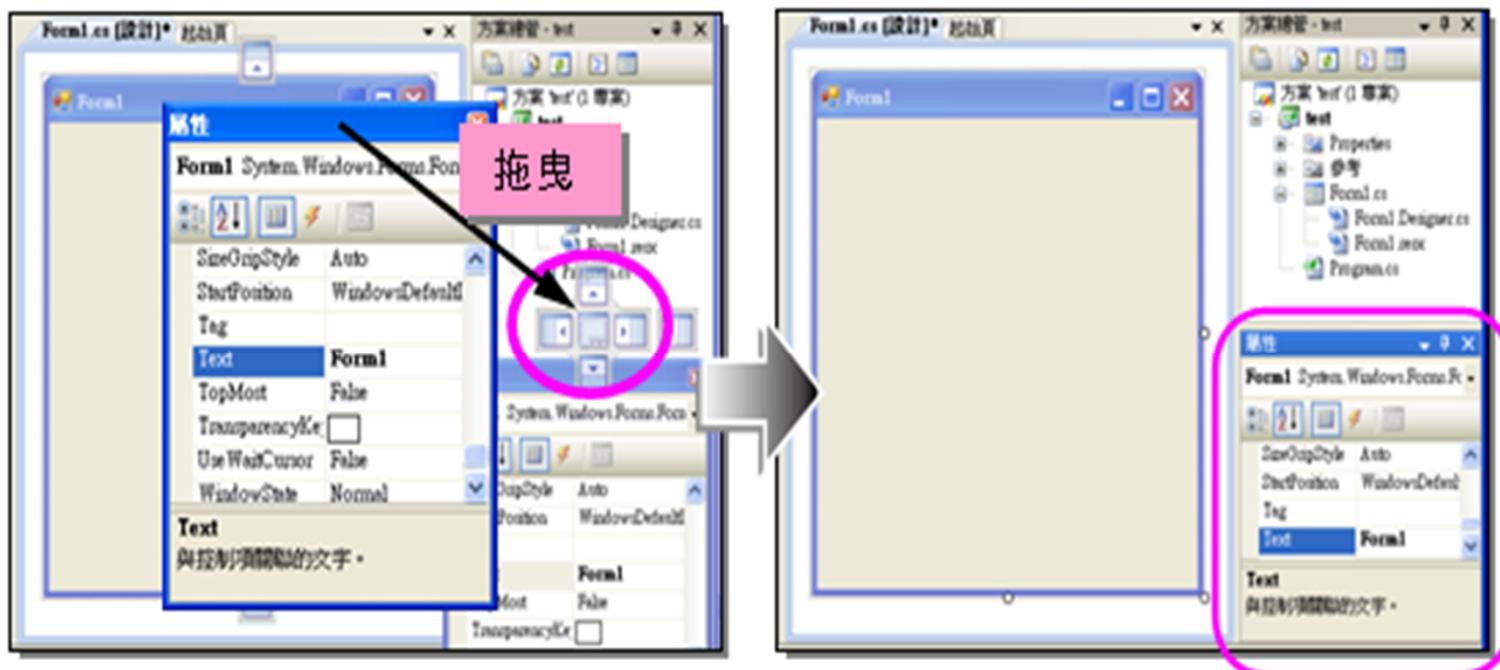
7.1.4 將方案總管和屬性視窗置於右邊界

■ 將屬性視窗變成浮動式視窗





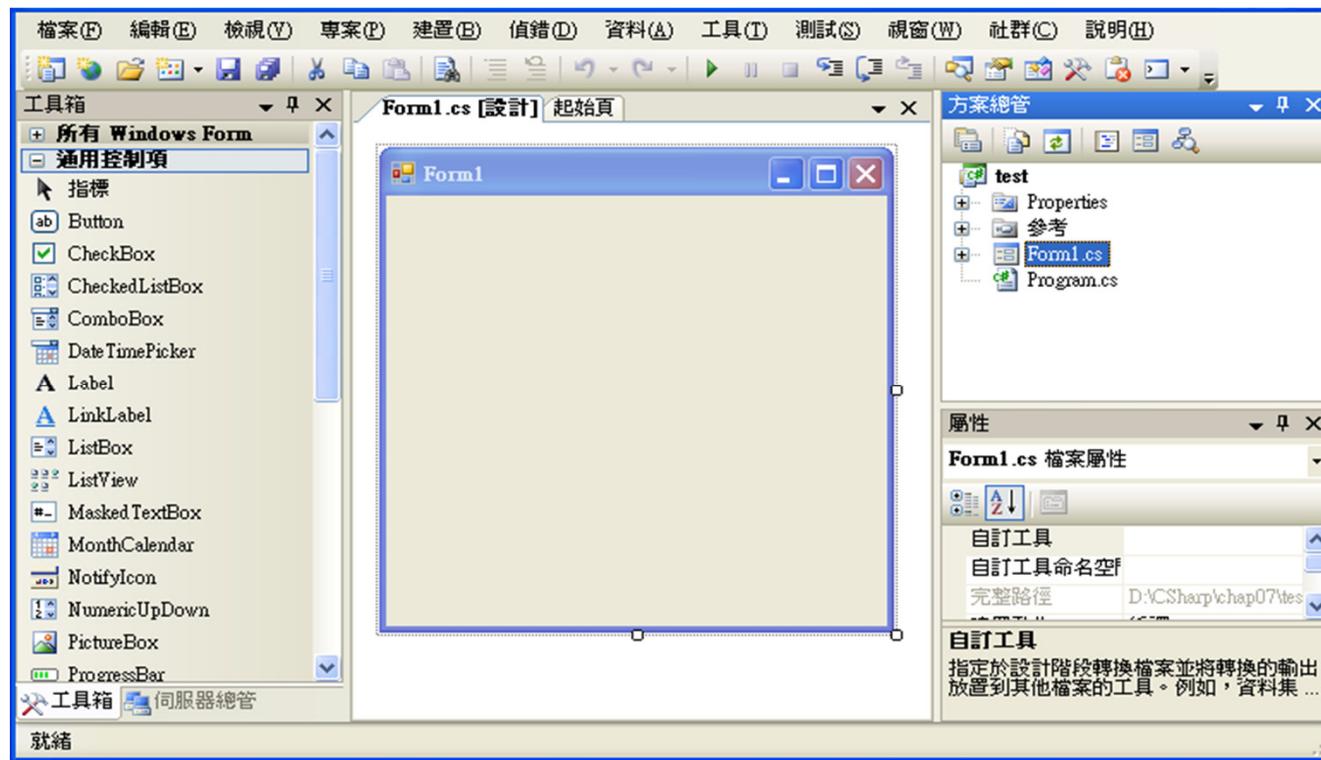
■ 將屬性視窗變成固定式視窗





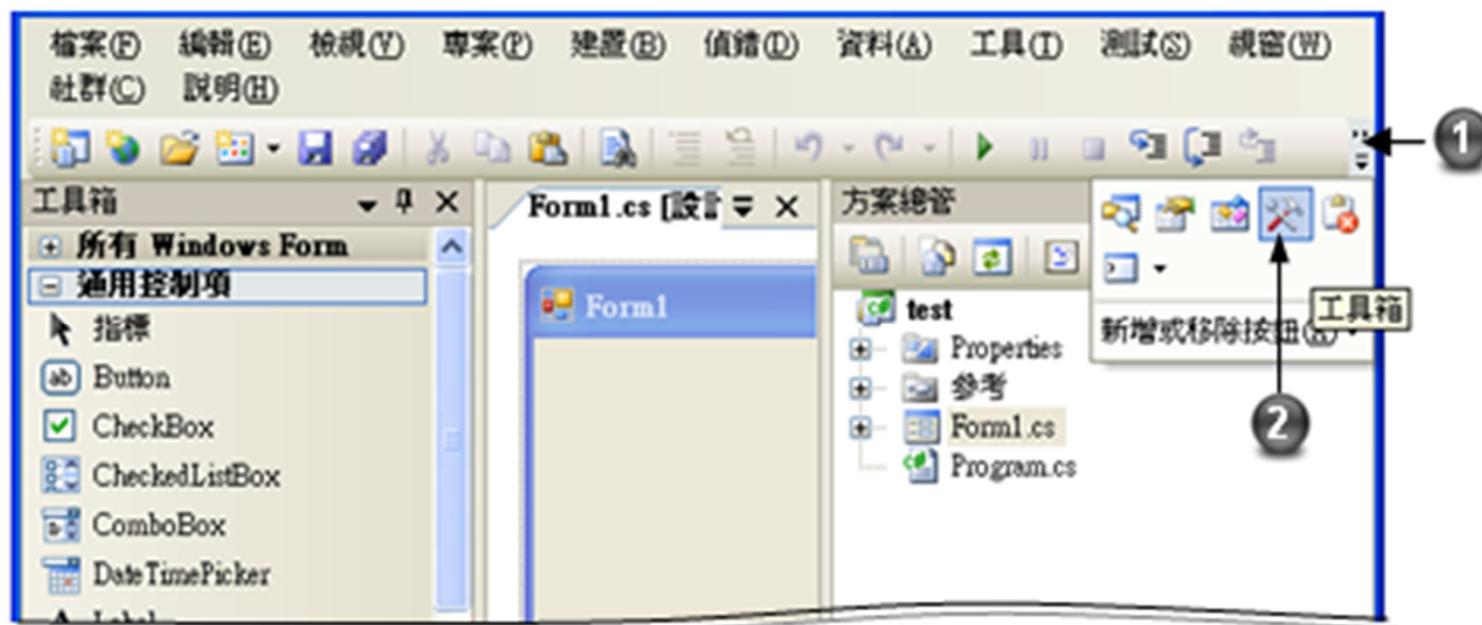
7.1.5 設定工具箱

- 若螢幕解析度設為1024x768，在IDE整合開發環境可在功能表列的正下方看到完整的標準工具列。



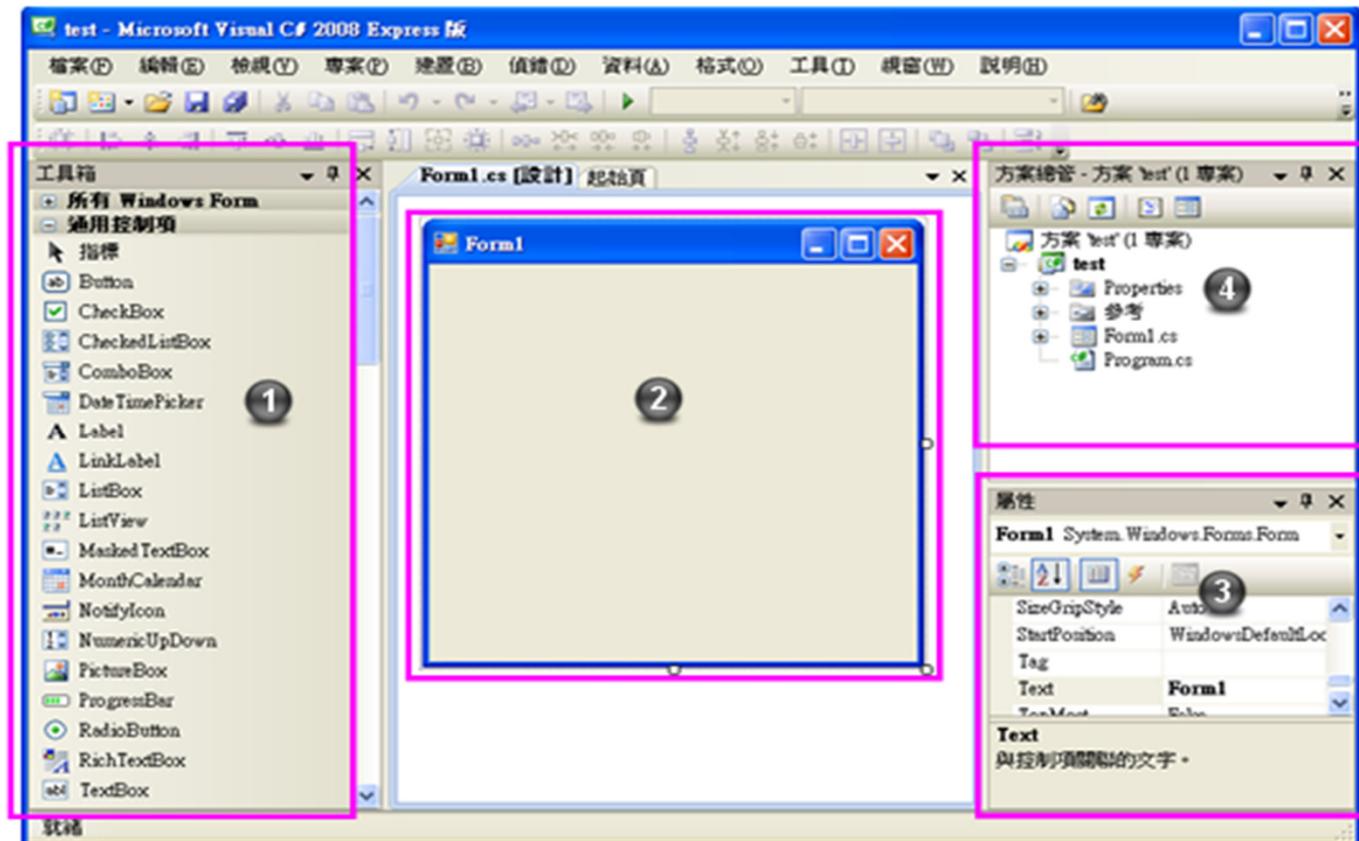


- 螢幕解析度是800x600，導致標準工具列長度太長而無法完整顯現，





螢幕解析度設為1024x768，是初學者用來撰寫 Visual C# 2008 程式的最佳整合開發環境(簡稱IDE)。



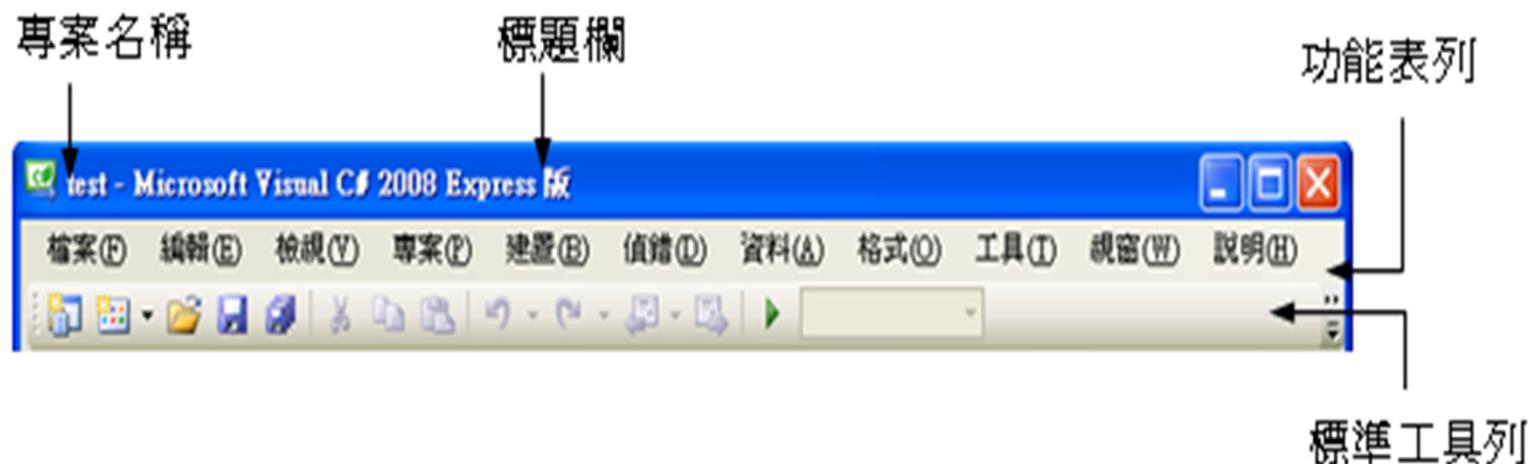


7.2 整合開發環境介紹

7.2.1 標題欄

■ 標題欄

標題欄中會顯示目前編輯的專案名稱以及程式語言名稱和版本。





7.2.2 功能表列

屬性	說明
檔案(E)	提供檔案存取、列印和專案新增等指令。
編輯(E)	提供復原、複製、尋找...等和編輯相關的指令。
檢視(V)	提供開啓或關閉 IDE 的各種視窗。
專案(P)	提供加入專案中表單、控制項、元件...等的指令。
建置(B)	提供方案和專案建置、重整的指令。
偵錯(D)	提供程式啟動、逐行執行...等和程式除錯相關的指令。
資料(A)	提供產生資料集和預覽資料的指令。
格式(O)	提供控制項的對齊、順序、鎖定的指令。
工具(T)	提供新增工具和設定 IDE 環境的相關指令。
視窗(W)	提供視窗各種顯示方式的相關指令。
說明(H)	提供輔助說明。



7.2.3 標準工具列

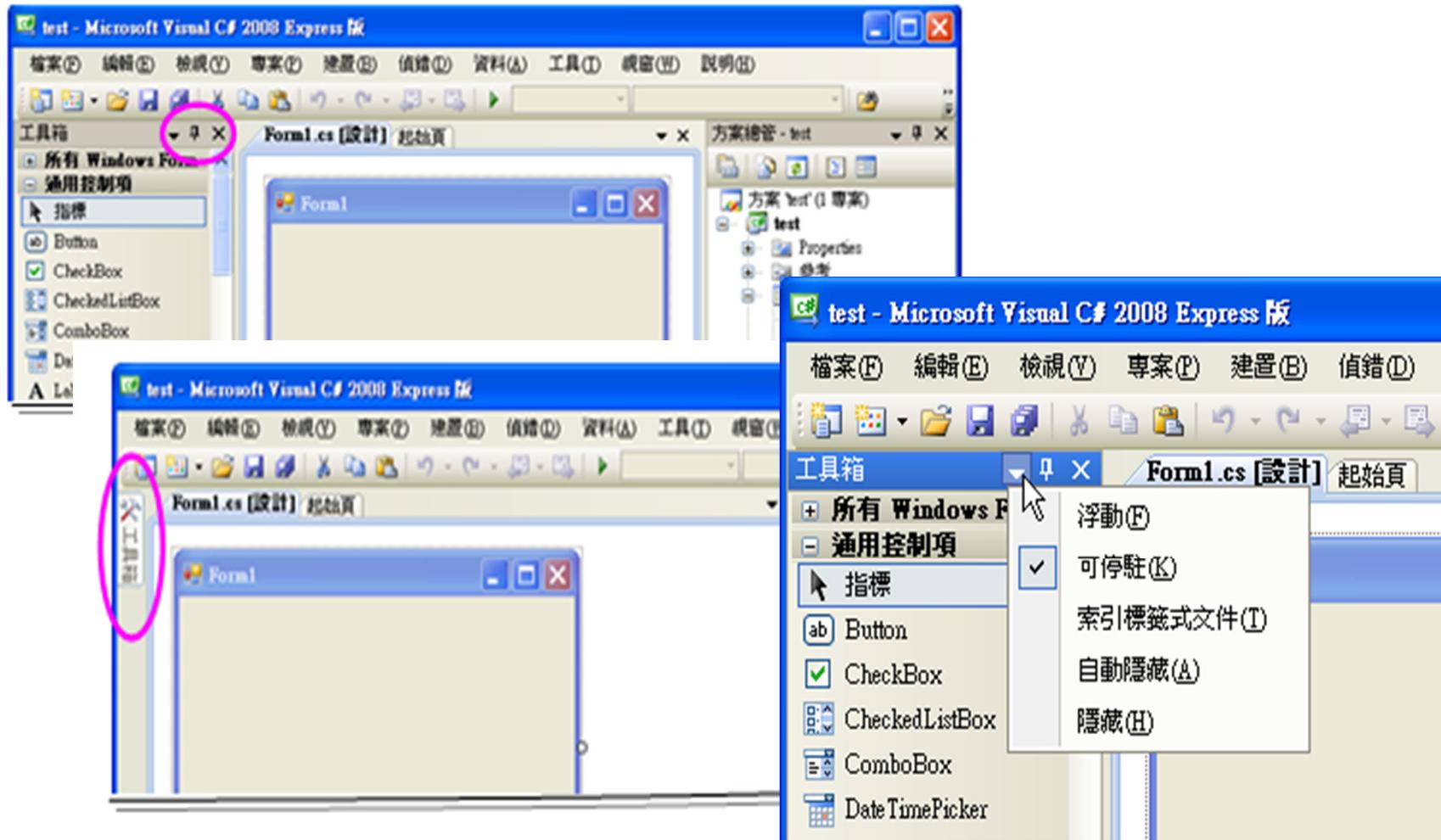
圖示	對應的功能表指令	功能說明
	[檔案/新增專案]	新增專案或空白方案。
	[檔案/加入新項目]	新增一個表單、類別...等項目。
	[檔案/開啟檔案]	開啟現有的專案。
	[檔案/儲存]	儲存目前編輯中的檔案。
	[檔案/全部儲存]	儲存方案中全部的檔案。
	[編輯/剪下]	將選取的物件或文字剪到剪貼簿中。
	[編輯/複製]	將選取物件或文字複製到剪貼簿中。
	[編輯/貼上]	將剪貼簿中的物件或文字複製到目前的位置。
	[編輯/復原]	取消前一個編輯動作。
	[編輯/取消復原]	復原前一個取消的編輯動作。
	[檢視/向後巡覽]	回到前面編輯的位置。



	[檢視/向前巡覽]	回到執行 [向後巡覽] 指令的編輯位置
	[偵錯/開始偵錯]	執行程式 (快速鍵為 F5)。
	[編輯/尋找和取代/ 檔案中尋找]	尋找指定的字串。
	[檢視/方案總管]	開啟方案總管視窗。
	[檢視/屬性視窗]	開啟屬性視窗。
	[檢視/物件瀏覽器]	開啟物件瀏覽器視窗。
	[檢視/工具箱]	開啟工具箱。
	[檢視/其他視窗/起始頁]	開啟起始頁。



7.2.4 工具箱





7.2.5 方案總管

- 方案（Solution）就像是一個容器，它可以包含多個專案（Project），而一個專案通常含有多個項目。
- 項目可以是檔案和專案的其他部分，如參考、資料連接或資料夾。
- Visual C# 2008 提供一個「方案總管」視窗，提供整個方案圖形檢視畫面，協助您在開發應用程式時管理專案和檔案。
- 若在IDE開發環境看不到方案總管視窗，可執行【檢視(V)方案總管(P)】指令，或直接點選「標準」工具列中的 方案總管圖示來開啟該視窗。

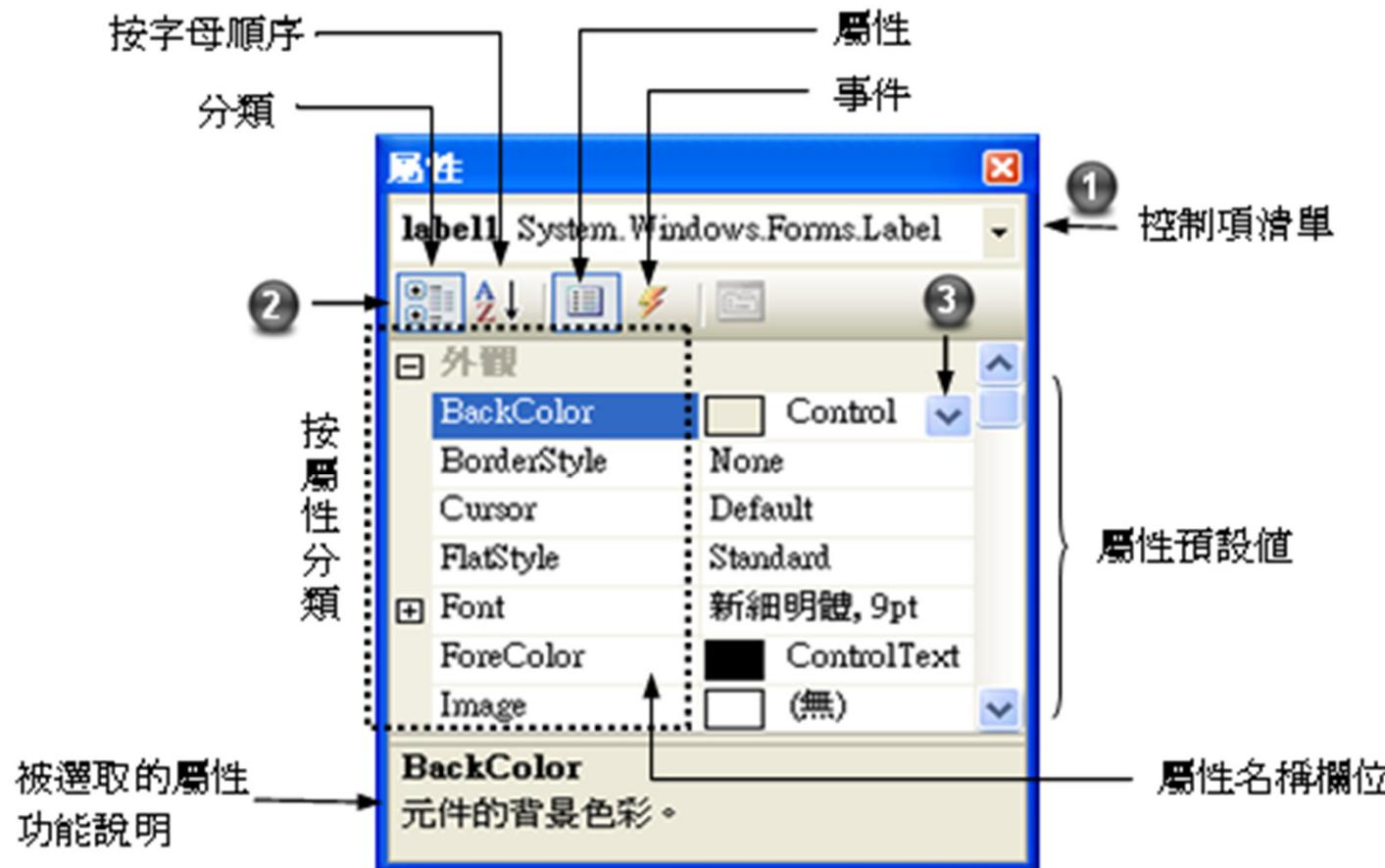


方案總管工具列圖示說明

	選取屬性窗格
	顯示所有的檔案
	重新整理
	程式碼檢視(撰寫程式)
	設計工具檢視(表單物件)



7.2.6 屬性視窗

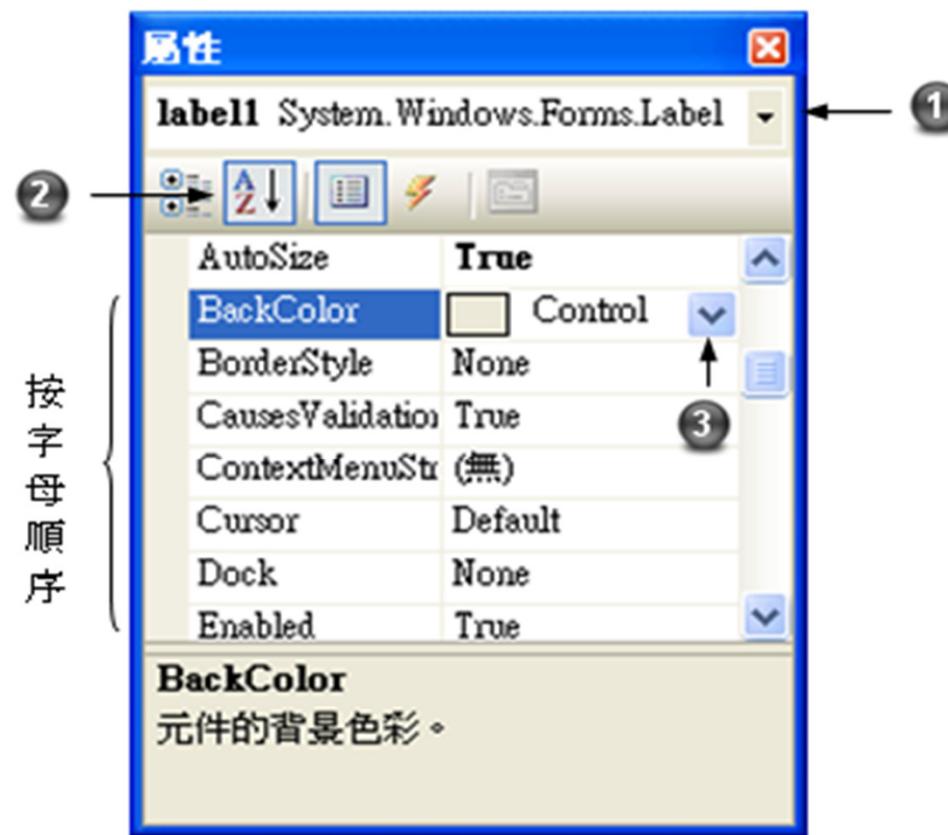


圖一 按屬性分類



屬性視窗 – 按字母分類

按此鈕會依字母順序排列屬性名稱

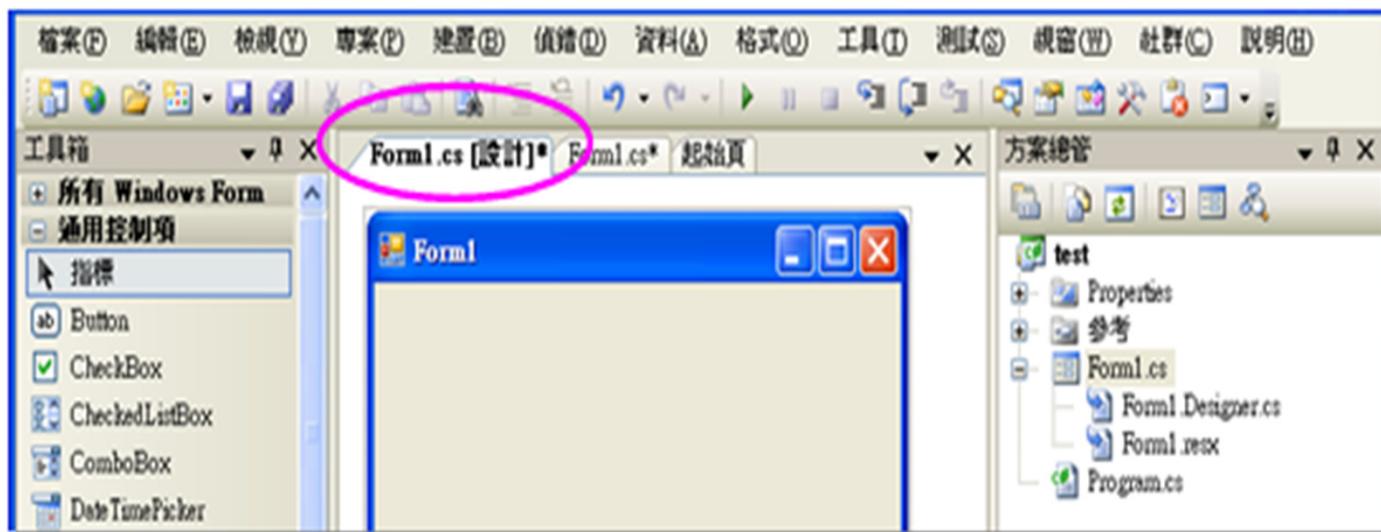


圖二 按字母分類



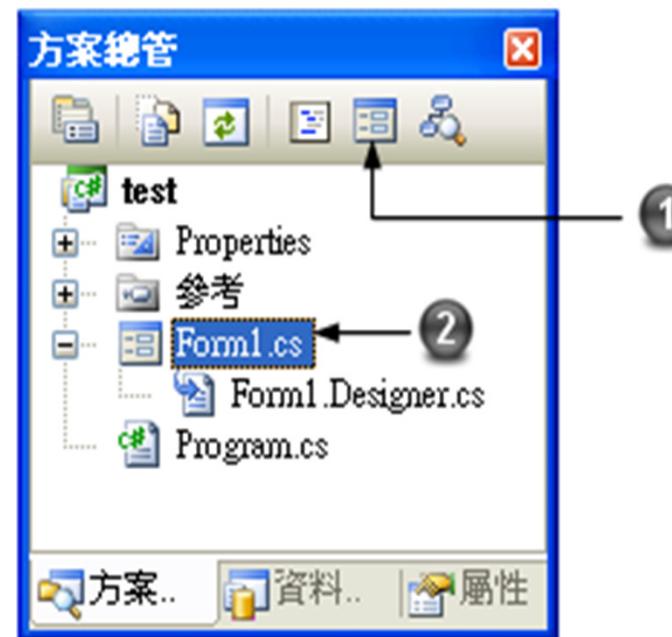
7.2.7 設計工具標籤頁

「設計工具」標籤頁是用來顯示和設計表單。我們可以在表單上新增控制項，作為程式輸出和輸入的介面。





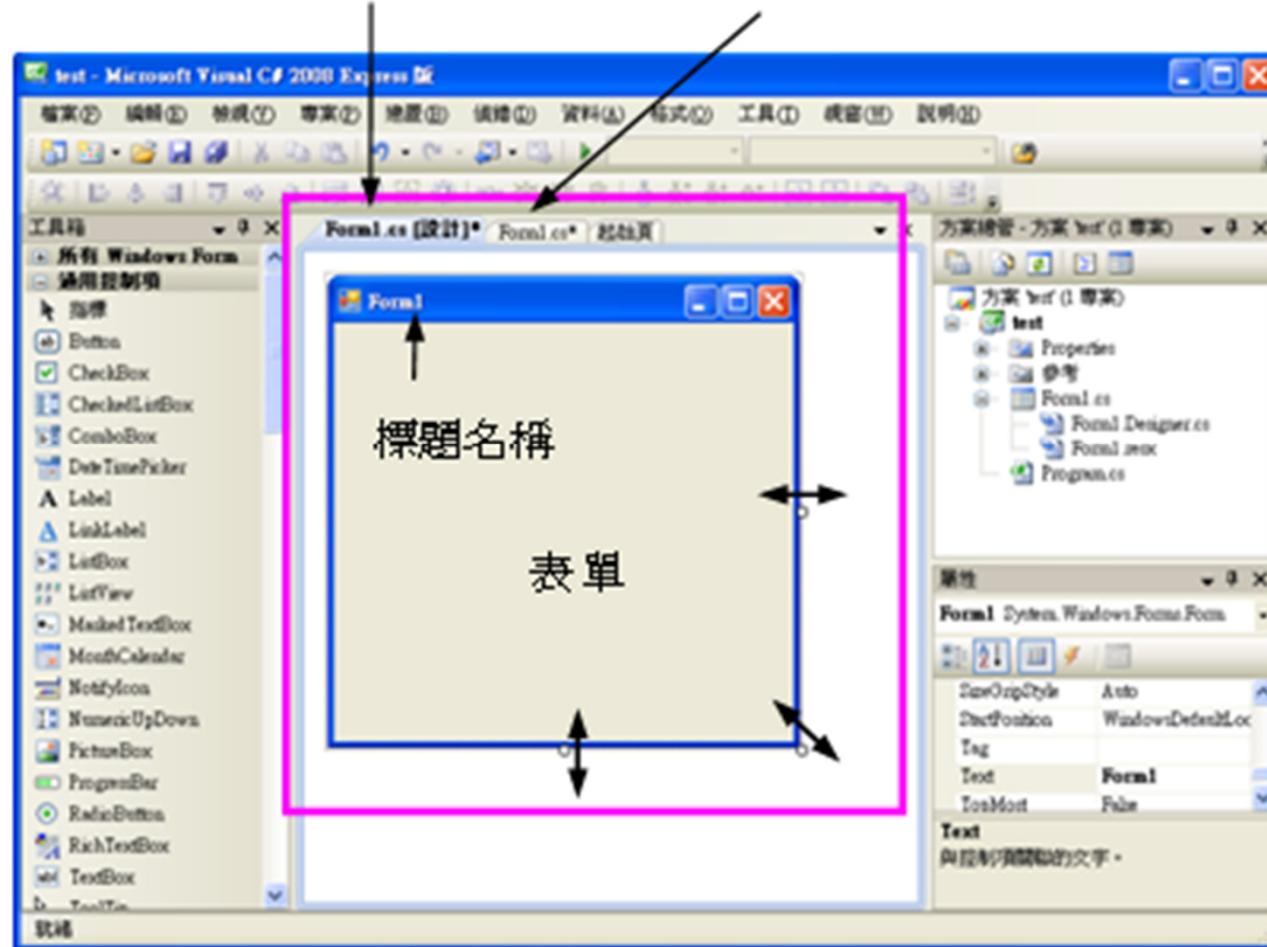
- 若在**IDE**環境下，沒看到「設計工具」標籤頁，可執行功能表的【檢視(V)/設計工具(D)】指令或如下圖選按「方案總管」工具列中的**設計工具檢視圖示**開啟該標籤頁；另外也可在方案總管的**Form1.cs**表單檔（副檔名為.cs）上快按兩下，會在**IDE**正中央出現「設計工具」標籤頁含有表單物件的畫面。





設計工具標籤頁

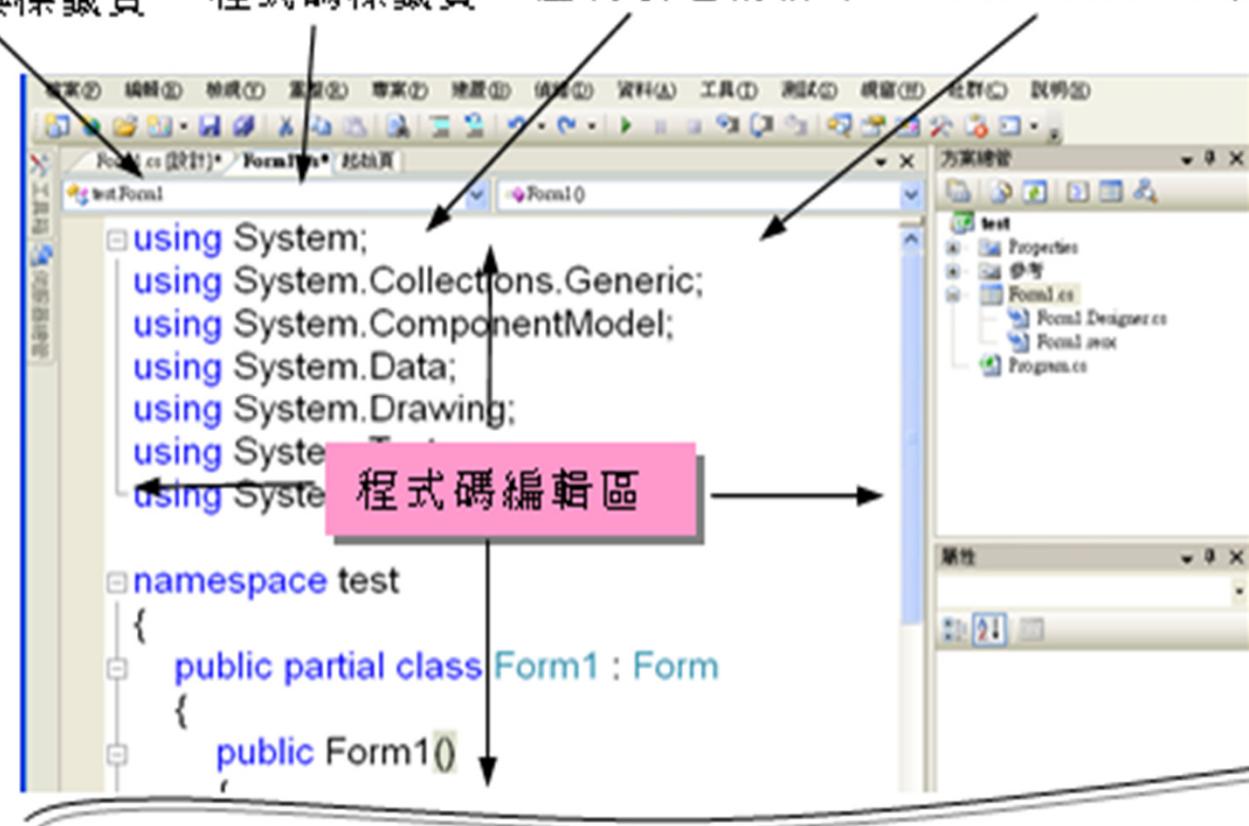
程式碼標籤頁





7.2.8 程式碼標籤頁

設計工具標籤頁 程式碼標籤頁 控制項名稱清單 事件名稱清單(成員清單)



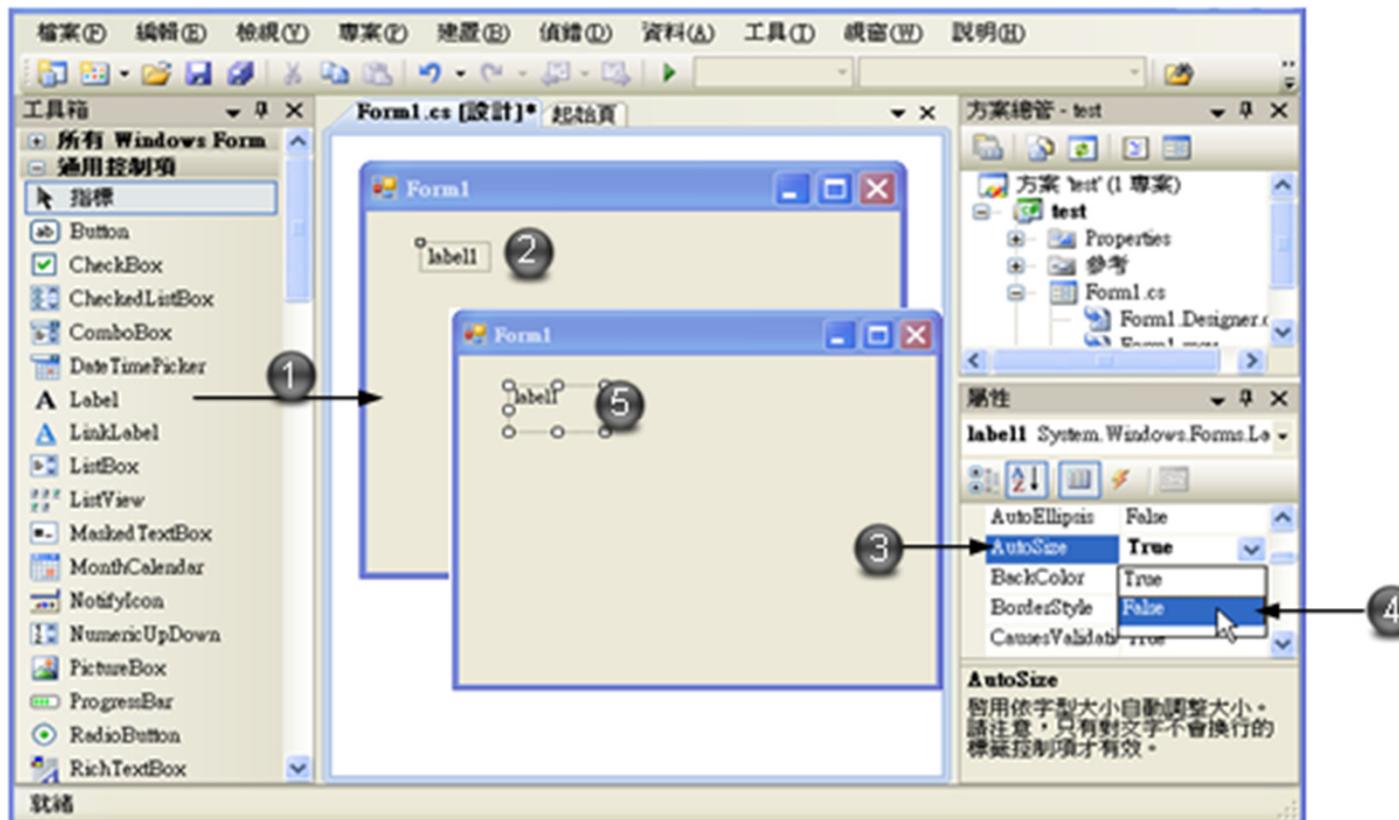


7.3 控制項編輯與命名





7.3.1 如何建立控制項





7.3.2 如何選取控制項

- 當你要對表單或表單中控制項的屬性作修改時，必須先選取表單或控制項變成作用物件成為作用控制項。
- 點選方式：
 - ① 單選：移動滑鼠到表單或控制項上按一下，使其當控制項四周有小白框出現表示該控制項被選取。
 - ② 單選：另一種方式連續按 **Tab** 鍵，可以在各控制項和表單間輪流切換選取。
 - ③ 多選：使用工具箱的 **Ctrl 指標** 指標工具將欲選取的多個控制項框住。
 - ④ 多選：先按住 **Ctrl** 或 **Shift** 鍵，再用滑鼠點選，也可選取多個控制項。



7.3.3 控制項的排列

功能	說明
對齊(A)	讓選取的控制項靠右、左…對齊。
設成相同大小(M)	可以讓選取的控制項調整成大小相同。
水平間距(H)	可以調整選取的控制項間的水平距離。
垂直間距(V)	可以調整選取的控制項間的垂直距離。
物件層次(O)	可以調整控制項的上下順序。
鎖定控制項(L)	鎖定已調整妥當的控制項，使其無法移動。此指令對全部的控制項有效，當點選控制項時，控制項的左上角出現鎖頭。



7.3.4 如何刪除控制項

■ 刪除控制項的方法

- ① 先選取要刪除的控制項或框住欲刪除多個控制項
- ② 滑鼠右鍵由快顯功能表中選取【刪除(D)】

或直接按鍵盤的 **Del** 鍵即可。

- 若欲復原直接按標準工具列的 復原鈕。



7.3.5 控制項的命名

控制項中文名稱	控制項英文名稱	字首	實例
表單	Form	frm	frmScore
標籤	Label	lbl	lblUserName
按鈕	Button	btn	btnDelete
文字方塊	TextBox	txt	txtPassWord
圖片方塊	PictureBox	pic	picMan
影像清單	ImageList	img	imgFlyBird
核取方塊	CheckBox	chk	chkGrade
清單方塊	ListBox	lst	lstBook
超連結標籤	LinkLabel	lnk	lnkIP
豐富文字方塊	RichTextBox	rtx	rtxInput



7.4 視窗應用程式的開發

7.4.1 視窗應用程式的開發步驟

視窗環境下開發一個應用程式的步驟：

Step1 分析問題

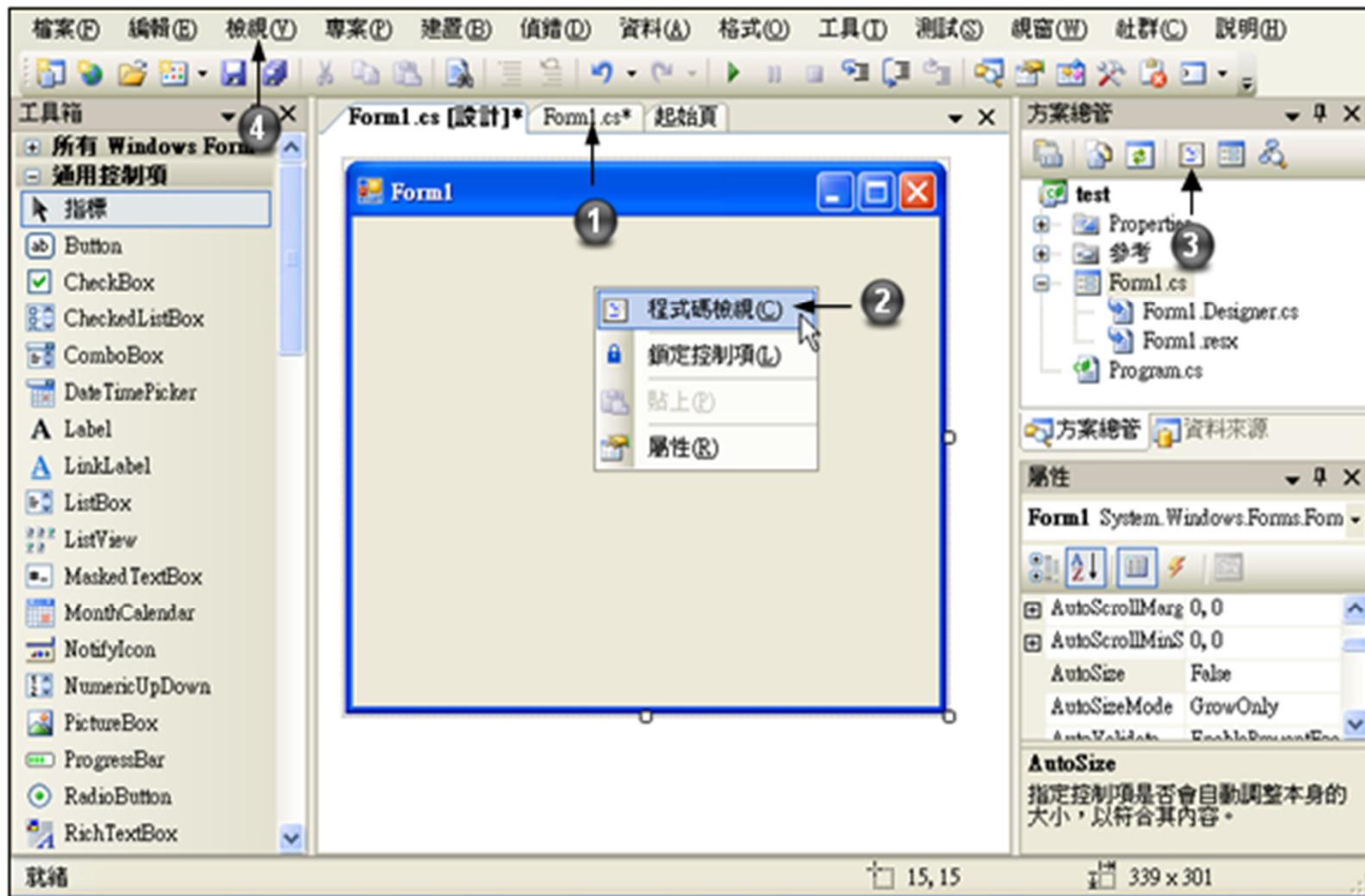
Step2 設計輸出入介面

Step3 撰寫程式碼



■ 撰寫程式方式有下列四種：

- ① 直接選取「**Form1.cs***」。
- ② 在控制項或表單物件上壓滑鼠右鍵，由快顯功能表中選取【**程式碼檢視(C)**】。
- ③ 在方案總管視窗選取  程式碼檢視圖示。
- ④ 執行功能表的【**檢視(V)/程式碼(C)**】指令。





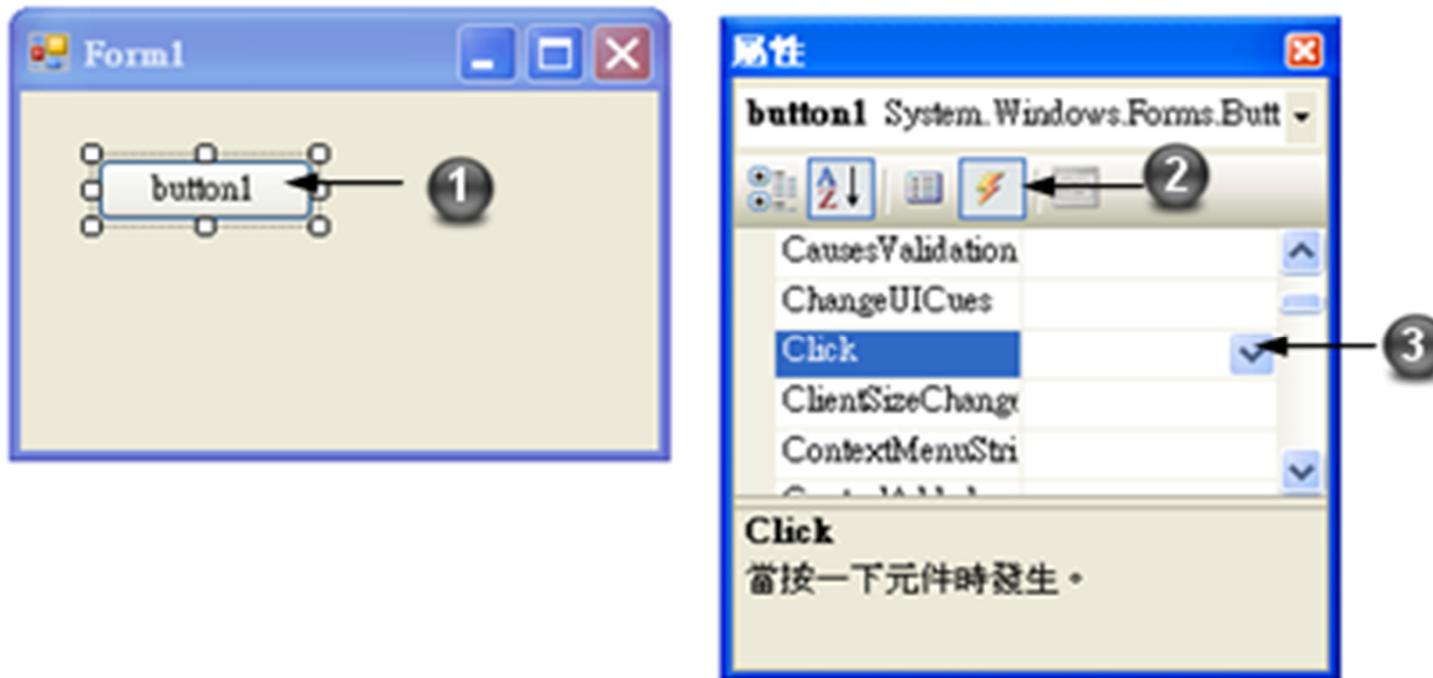
■ 當使用上面其中一種方式便可進入「程式碼檢視」模式：

```
using System;
using System.Collections.Generic;
using System.ComponentModel;
using System.Data;
using System.Drawing;
using System.Text;
using System.Windows.Forms;

namespace test
{
    public partial class Form1 : Form
    {
        public Form1()
        {
            InitializeComponent();
        }
    }
}
```



7.4.2 如何撰寫控制項的事件處理函式





- ① 先選取要撰寫事件的物件，請選取 **button1** 按鈕。
- ② 在屬性視窗的 事件圖示鈕按一下切換到事件清單。
- ③ 在 Click 清單快按兩下即進入下圖 **button1_Click** 事件處理函式。

The screenshot shows the Visual Studio code editor with the file `Form1.cs` open. The code editor displays the following C# code:

```
namespace test
{
    public partial class Form1 : Form
    {
        public Form1()
        {
            InitializeComponent();
        }

        private void button1_Click(object sender, EventArgs e)
        {
        }
    }
}
```

The `button1_Click` method is highlighted with a pink dotted selection. The status bar at the bottom right of the code editor shows the number 37.



The diagram illustrates the relationship between a Windows Form and its associated C# code. On the left, a screenshot of the Visual Studio IDE shows the 'Form1.cs [設計]' tab selected, displaying a Windows Form titled 'Form1' with two buttons, 'button1' and 'button2'. A large grey arrow points from this form towards the code editor on the right. The code editor window is titled 'Form1.cs [設計]*' and shows the following C# code:

```
namespace test
{
    public partial class Form1 : Form
    {
        public Form1()
        {
            InitializeComponent();
        }

        private void button1_Click(object sender, EventArgs e)
        {

        }

        private void button2_Click(object sender, EventArgs e)
        {
        }
    }
}
```

The event handler for 'button2_Click' is highlighted with a pink dashed rectangle.



範例演練

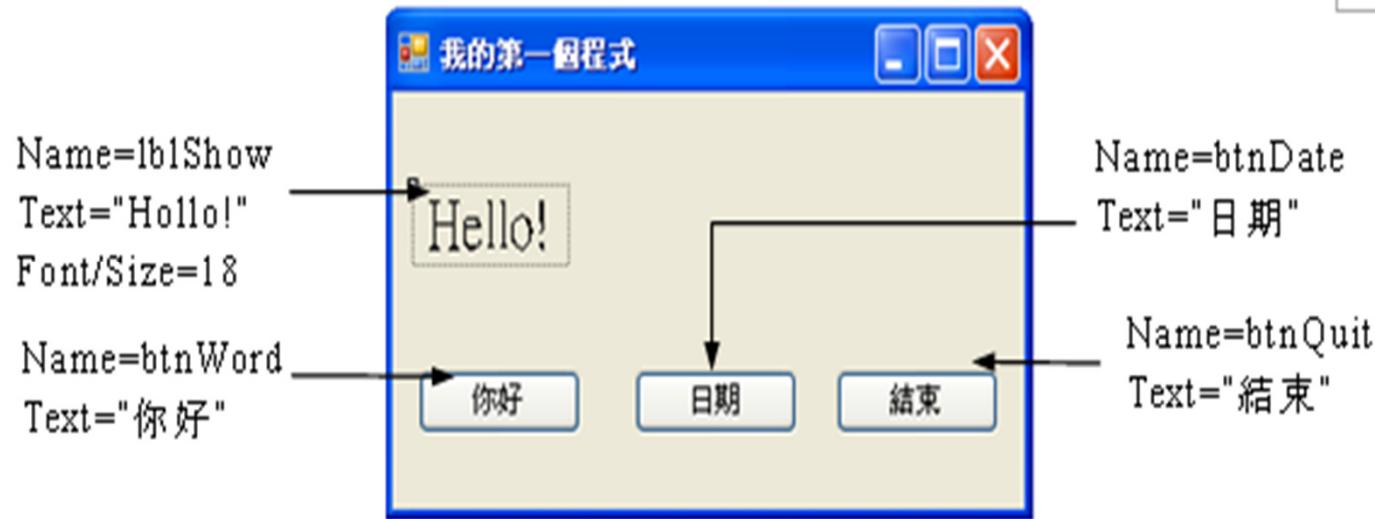
使用上面介紹的視窗應用程式設計步驟，撰寫一個簡單的程式。其要求如下：

- ① 程式開始執行時表單的標題欄會顯示”我的第一個程式”以及表單上方出現 “Hello” 打招呼訊息，表單下方出現三個按鈕分別為 、、 三個按鈕。
- ② 當你按 鈕，顯示 “你好” ，背景為淺藍色(Aqua)。
- ③ 當你按 鈕，顯示今天日期，背景為巧克力色(Chocolate)。
- ④ 當你按 鈕，結束程式執行。



輸出結果





操作步驟：參閱本書 P7-28 ~ P7-37 頁



```
// FileName : hello.sln
01 using System;
02 using System.Collections.Generic;
03 using System.ComponentModel;
04 using System.Data;
05 using System.Drawing;
06 using System.Text;
此部份程式碼為C#自動產生，為節省篇幅，後面範例除非有必要顯示，否則將不顯示。
07 using System.Windows.Forms;
08
09 namespace hello
10 {
11     public partial class Form1 : Form
12     {
13         public Form1()
14         {
15             InitializeComponent();
16         }
17     }
```



```
18  private void btnWord_Click(object sender, EventArgs e)
19  {
20      lblShow.BackColor = Color.Aqua;
21      lblShow.Text = "你好!";
22  }
23
24  private void btnDate_Click(object sender, EventArgs e)
25  {
26      lblShow.BackColor = Color.Chocolate;
27      lblShow.Text = DateTime.Now.ToString();
28  }
29
30  private void btnQuit_Click(object sender, EventArgs e)
31  {
32      Application.Exit();
33  }
34 }
35 }
```