

# E

## ViLLE程式語言 學習工具的安裝 與使用

- ◆ E-1 ViLLE 程式語言學習工具的基礎
- ◆ E-2 ViLLE 程式語言學習工具的安裝
- ◆ E-3 ViLLE 程式語言學習工具的使用
- ◆ E-4 參考文獻

## E-1 ViLLE 程式語言學習工具的基礎

程式設計（Programming）是資訊科學一門相當重要的課程，也是數十年來電腦資訊教育上最大的挑戰，有相當多的研究證明程式語言模擬工具（也稱為視覺化學習工具）可以幫助初學者學習程式設計。

有鑑於此，筆者在評估現有眾多輔助學習工具軟體後，在本書導入程式語言視覺化學習工具 ViLLE，以便幫助初學者學習程式設計，建立正確的程式設計觀念。

### E-1-1 ViLLE 學習工具簡介

ViLLE 是芬蘭 Turku 大學開發的程式語言教學工具，這是由 Teemu Rajala、Erkki Kaila 和 Mikko Laakso 使用 Java 語言開發的程式語言視覺化教學工具。

ViLLE 是一套與程式語言種類無關（Language-independent）的視覺化學習工具（The Visual Learning Tool），提供程式設計的抽象觀點來模擬程式的執行，可以讓使用者一行一行的追蹤程式執行，並且監視程式輸出和變數的狀態。

在教學上，ViLLE 適用在課堂或網路線上輔助教學，也可以讓初學者自學，同時支援多種程式語言的學習，包含：Java、C++、PHP、JavaScript 和 Python 等（筆者已經自行新增 Visual Basic 語言）。

ViLLE 是一套模擬程式執行的軟體工具，可以實際模擬原始程式碼的執行，顯示程式執行時的呼叫堆疊和變數狀態，並且使用視覺化的方式一步一步前進或後退來檢視每一行原始程式碼的執行過程。

不只如此，ViLLE 本身已經內建有數十個教學範例程式，包含程式語言基本語法的變數、運算式、條件敘述、迴圈結構、程序與函數和陣

列等，可以幫助初學者學習基本語法，和各種流程控制結構的執行流程，讓初學者可以使用電腦方式來思考與學習程式設計。

### **E-1-2 ViLLE 學習工具的特點**

ViLLE 視覺化學習工具的功能十分強大，其主要特點的說明，如下所示：

- 與程式語言種類無關：ViLLE 可以使用多種程式語言來模擬程式的執行，因為程式設計的觀念並沒有差別，只是使用的程式語言不同，初學者可以透過檢視不同程式語言的執行過程，來加強程式設計觀念的建立。
- 同時檢視不同程式語言的同步執行：ViLLE 在模擬程式執行時，可以選擇 2 種程式語言來同時模擬執行，例如：Visual Basic 和 C++，能夠讓初學者了解程式設計的相同處，和比較程式語言的語法差異。
- 彈性控制程式的模擬執行：在 ViLLE 模擬執行介面擁有動畫工具列的控制按鈕，除了能夠自動模擬執行或以一次一行方式來模擬執行外，還可以後退來一行一行執程式碼，幫助初學者來回檢視程式執行過程的變數狀態。
- 彈出式問題視窗：ViLLE 提供單選題的彈出式問題視窗，能夠在程式模擬執行的過程中，在執行指定程式碼行時彈出問題視窗，加強初學者程式設計的學習效果。
- 呼叫堆疊的狀態：ViLLE 在模擬程序與函數呼叫的執行過程中，每一次的呼叫，都會使用獨立視窗顯示程序與函數呼叫的堆疊狀態，除了顯示程序與函數的參數與區域變數值外，當結束程序與函數執

行後，還會顯示函數的傳回值，然後關閉視窗來模擬結束程序與函數的執行。

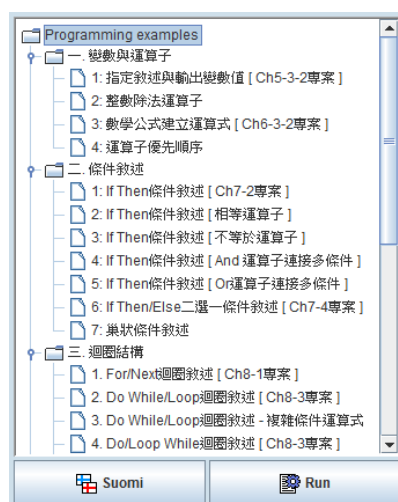
- 使用一行一行的方式來視覺化執行程式碼：ViLLE 的模擬執行方式是使用色彩顯示原始程式碼行來標示程式執行的目前進度，不只如此，它還會使用不同色彩標示前一行程式碼，讓初學者更容易追蹤程式的執行。

### E-1-3 ViLLE 教學範例程式的說明

ViLLE 視覺化學習工具不只是一個模擬程式，更內建多種程式語法的教學範例程式，可以讓初學者馬上使用 ViLLE 學習程式設計。

#### 教學範例目錄

在 ViLLE 內建數十個英文與芬蘭語版本的教學範例，包含：變數、指定敘述、運算子、條件敘述、迴圈結構、程序與函數和陣列等，如下圖所示：

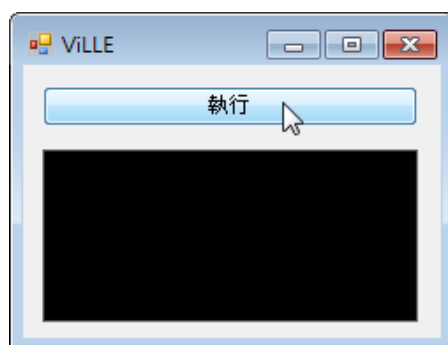


上述圖例的範例是筆者中文化英文版的教學範例（在 ViLLE 使用介面部分仍是英文介面），在結合本書範例專案（即項目後各節的專案）後，可以讓讀者搭配學習工具來學習 Visual Basic 語言的基本語法。

## Visual Basic 教學範例程式

本書主要是說明如何使用 Visual Basic 建立 Windows 視窗應用程式，使用事件驅動程式設計建立 Windows 應用程式，而非控制台應用程式，所以沒有主程序 Main()。

為了讓 ViLLE 可以模擬執行 Visual Basic 程式，筆者是使用一個測試用途的 Windows Form 專案來建立 Visual Basic 視窗應用程式，如下圖所示：



按上方的【執行】鈕，可以執行 Button1\_Click()事件處理程序，其執行結果是輸出至下方 lblOutput 標籤控制項，換句話說，在 ViLLE 教學範例執行的 Button1\_Click()事件處理程序，相當於是其他程式語言的主程式。

## E-2 ViLLE 程式語言學習工具的安裝

ViLLE 程式語言學習工具是一個 Java Applet 應用程式，在安裝 Java 執行環境（Java SE Runtime Environment，JRE）後，就可以在瀏覽程式執行 ViLLE 程式語言學習工具。

### E-2-1 安裝 Java 執行環境

在瀏覽程式執行 ViLLE 程式語言學習工具前，我們需要在 Windows 作業系統安裝 JRE，即 JRE 6（Java SE Runtime Environment 6），其下載網址為：<http://java.sun.com/javase/downloads/index.jsp>。

請建立 Internet 連線昇陽網站下載 JRE 6，在本節是使用 Update 23 為例，其下載程式檔案名稱為【jre-6u23-windows-i586.exe】。在 Windows 作業系統安裝 JRE 的步驟，如下所示：

Step 1：請按二下【jre-6u23-windows-i586.exe】程式檔案，稍等一下，可以看到歡迎安裝的精靈畫面。



Step 2：選下方中間紅色【授權合約】超連接，可以顯示授權合約，按【安裝】鈕同意授權且開始安裝 JRE，可以看到目前的安裝進度。



Step 3：稍等一下，等到安裝完成，可以看到成功安裝的精靈畫面。



Step 4：按【關閉】鈕完成 JRE 的安裝。

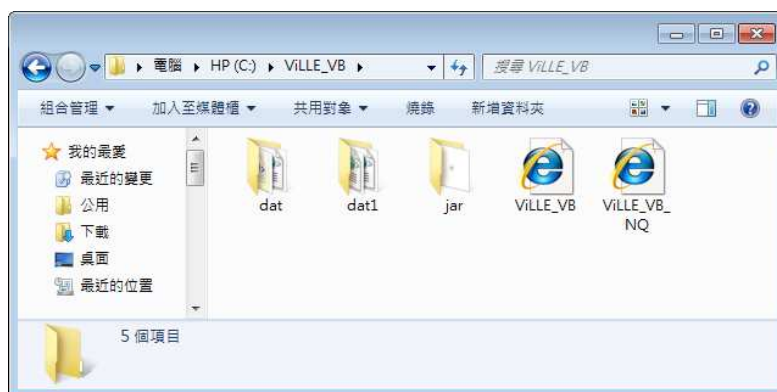
## E-2-2 安裝與啟動 ViLLE 程式語言學習工具

ViLLE 程式語言學習工具並沒有安裝程式，我們只需在 Windows 作業系統安裝 JRE 後，就可以在 Internet Explorer 啟動 ViLLE 程式語言學習工具。

### 安裝 ViLLE 程式語言學習工具

---

ViLLE 並不需要安裝，只需將 ViLLE 相關程式檔案複製至指定資料夾，例如：「C:\ViLLE\_VB」資料夾，如下圖所示：



在上述圖例的 2 頁 HTML 網頁檔案內嵌有 ViLLE 的 Java Applet 程式，其說明如下所示：

- ViLLE\_VB.html：執行 ViLLE 的 Java Applet，在模擬執行學習範例時會顯示彈出式視窗的測驗問題。
- ViLLE\_VB\_NQ.html：ViLLE 只會單純執行範例程式，並不會顯示彈出式視窗的測驗問題。

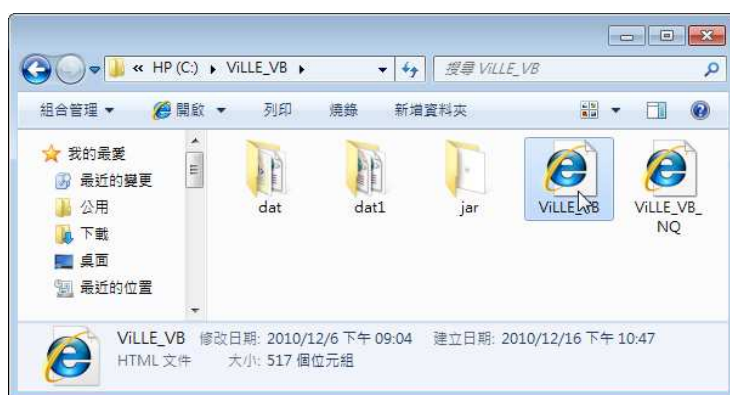
### 啟動 ViLLE 程式語言學習工具

---

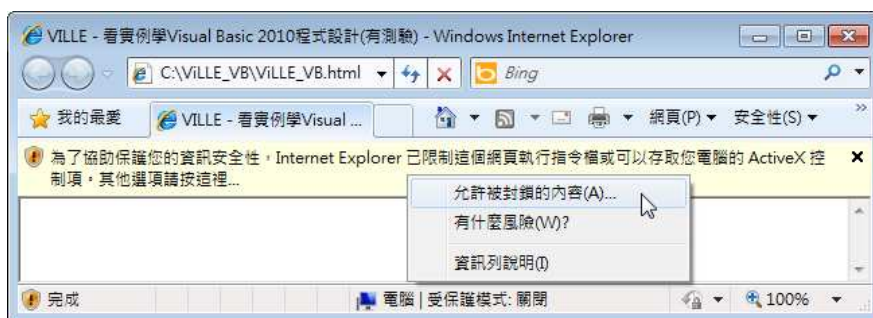


在複製 ViLLE 和安裝 JRE 後，我們就可以在 Internet Explorer 載入網頁來啟動 ViLLE 程式語言學習工具，其步驟如下所示：

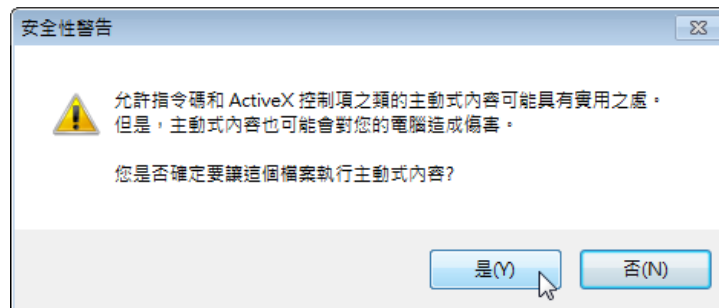
Step 1：請開啟 ViLLE 應用程式所在的「C:\ViLLE\_VB」資料夾，如下圖所示：



Step 2：按一下【ViLLE\_VB.html】或【ViLLE\_VB\_NQ.html】的 HTML 網頁，可以啟動 Internet Explorer，在上方看到一個資訊安全性警告的訊息列，如下圖所示：



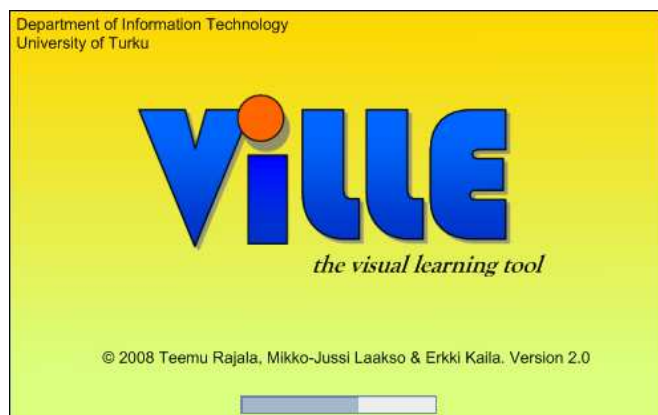
Step 3：在訊息列上執行右鍵快顯功能表的【允許被封鎖的內容】指令，可以看到一個「安全性警告」視窗。



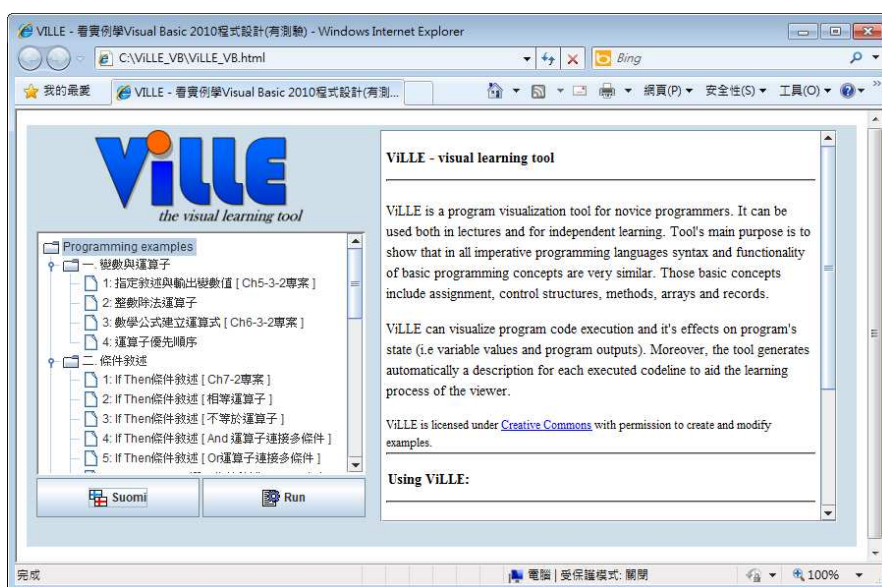
Step 4：按【是】鈕允許網頁執行 Java Applet，接著再看到一個無法驗證數位簽名的「警告-安全」視窗。



Step 5：按【執行】鈕確認執行 Java Applet，可以看到 ViLLE 版權頁。



Step 6：稍等一下，可以顯示執行 ViLLE 的網頁內容，如下圖所示：



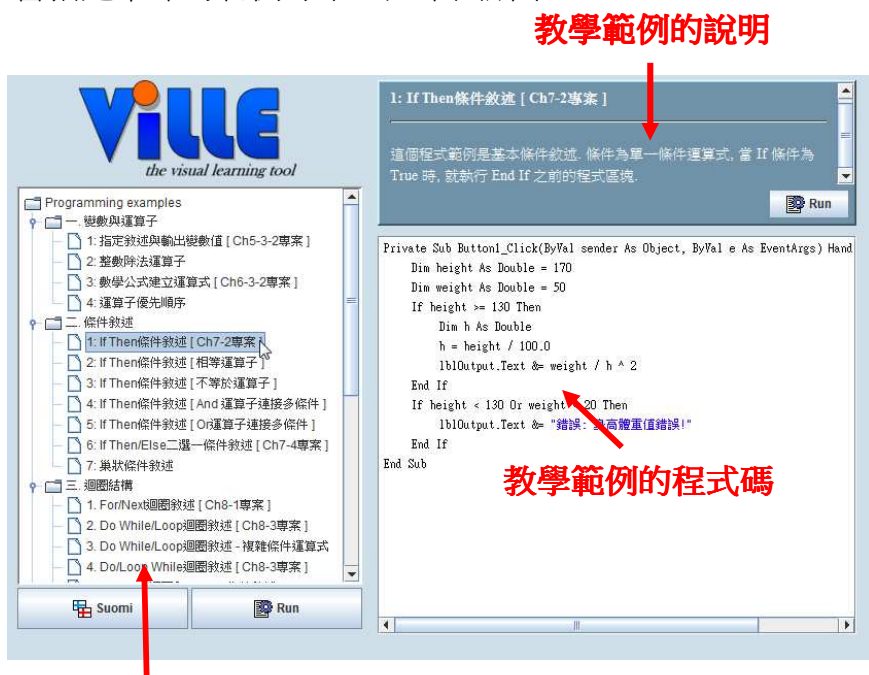
上述圖例是 ViLLE 學習工具 Java Applet 的執行畫面。結束 ViLLE 請按視窗右上角的【X】鈕關閉 Internet Explorer，即可結束 ViLLE 的執行。

## E-3 ViLLE 程式語言學習工具的使用

在成功啟動 ViLLE 程式語言學習工具後，我們就可以選擇欲學習的教學範例，然後執行範例程式的模擬執行動畫。

### E-3-1 範例目錄的主使用介面

在 ViLLE 主使用介面的左邊顯示教學範例的分類目錄，展開分類目錄，可以看到之下的教學範例程式碼項目，項目名稱後的專案章節是對應本書指定章節的範例專案，如下圖所示：

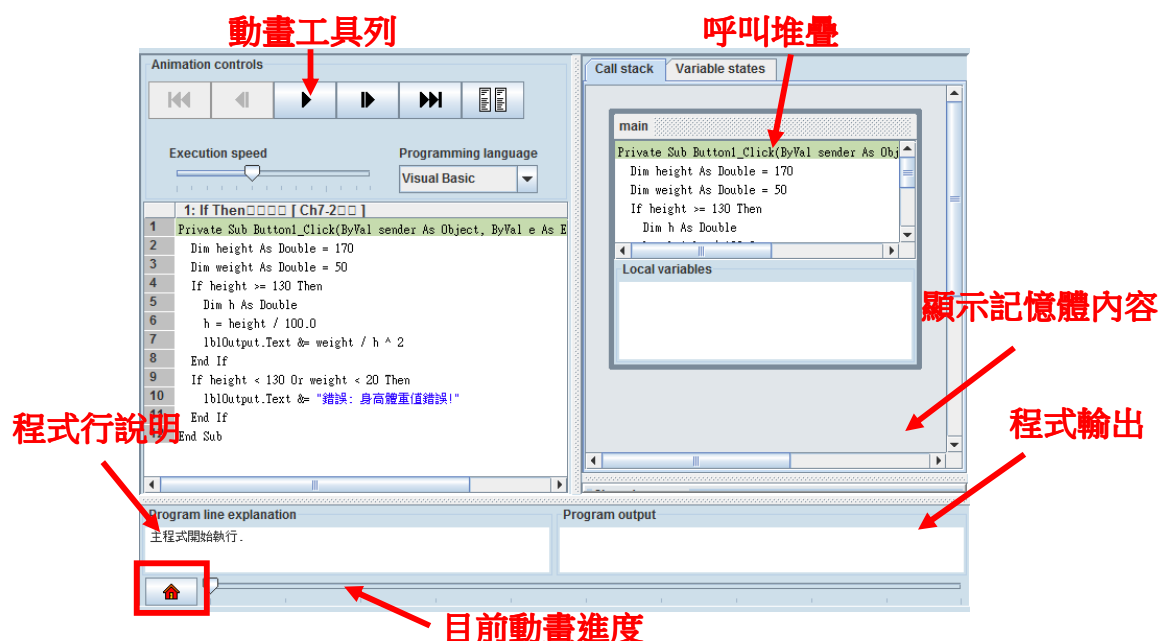


#### 教學範例目錄

在左邊選取範例，就可以在右上方顯示範例說明；右下方是原始程式碼，按下方或右上方的【Run】鈕模擬執行此範例程式碼。

### E-3-2 模擬執行介面

在 ViLLE 主使用介面選取範例程式後，按【Run】鈕執行範例程式，可以看到 ViLLE 的模擬執行介面，如下圖所示：



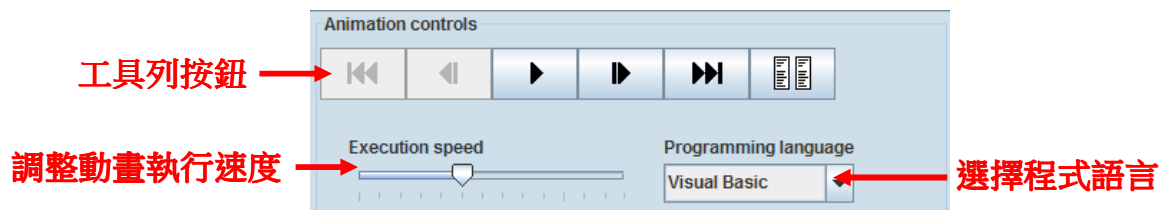
上述圖例的左上方是動畫工具列，右上方標籤頁可以切換顯示呼叫堆疊（Call stack）或變數狀態（Variable states）。在動畫工具列下方是原始程式碼列表，其上的程式名稱因為中文編碼問題，並無法正確顯示中文內容。

在下方左邊是程式行說明（Program line explanation），可以顯示目前模擬執行程式碼行的說明，不過，大部分內容都是英文說明，右下方是程式輸出結果（Program output），即程式的執行結果。







右方中間的分享記憶體（Shared memory）可以顯示記憶體狀態，主要是顯示陣列內容，按左下角的  鈕，可以回到主使用介面。

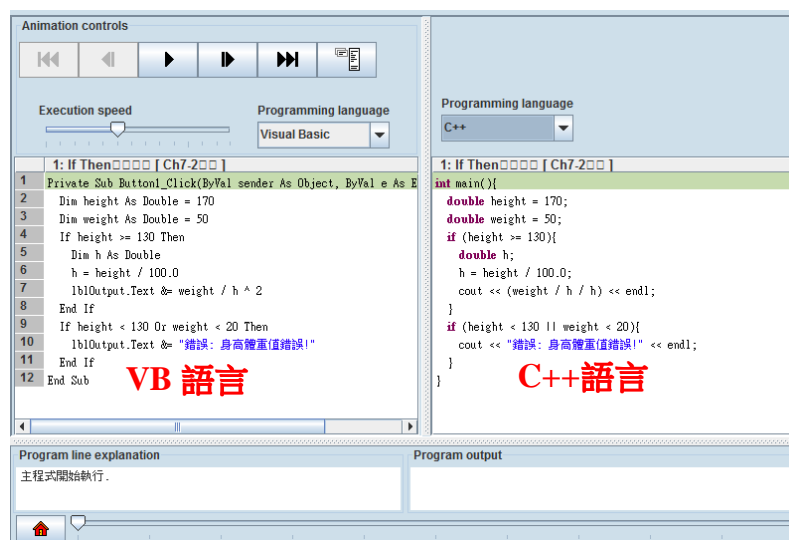
## 動畫工具列

動畫工具列提供按鈕可以控制模擬程式的動畫執行、調整動畫執行速度和使用的程式語言，如下圖所示：



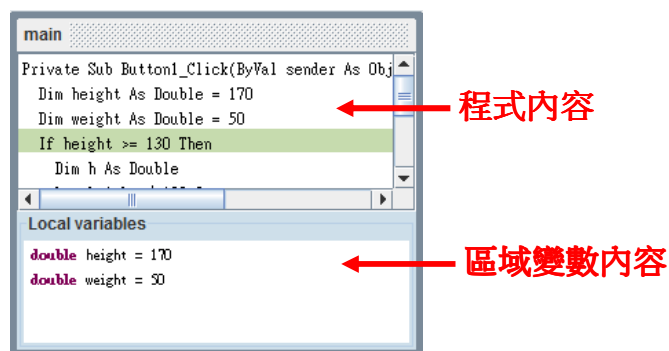
上述工具列按鈕從左至右的說明，如下所示：

- ：此按鈕可以移至原始程式碼的開頭，即第一行程式碼。
- ：當不是自動執行時，按此按鈕移至前一行原始程式碼。
- ：按此按鈕可以切換自動執行原始程式碼，然後在下方調整動畫的執行速度。再按一次按鈕，可以停止自動執行。
- ：當不是自動執行時，按此按鈕移至下一行原始程式碼。
- ：移至原始程式碼的結尾，即最後一行程式碼。
- ：切換是否選擇同時執行兩種程式語言，如下圖所示：



## 呼叫堆疊

在呼叫堆疊視窗可以顯示程序與函數的狀態，因為範例只有一個 Button1\_Click() 事件處理程序，相當於其他程式語言的主程式 main()，所以只顯示一個視窗的程式內容，其標題為 main，如下圖所示：



當程式每一次呼叫程序與函數時，在呼叫堆疊就會顯示一個視窗。視窗分為上下兩個部分，上方是程序與函數的程式碼；下方是此程序與函數的區域變數值。

請注意！因為 ViLLE 是使用 Java 語言開發的教學工具，所以區域變數顯示的資料型態仍然是 Java 語言的資料型態，int 相當於是 Visual Basic 的 Integer；double 相當於是 Double 資料型態，只有 String 是相同的名稱。

### **E-3-3 在 ViLLE 模擬執行原始程式碼**

雖然 ViLLE 和整合開發工具的偵錯功能都可以逐步執行原始程式碼，不過其執行順序並非完全相同，因為偵錯功能是為了程式除錯，所以需先指定中斷點，並且只會執行哪些可執行程式碼，而忽略掉變數宣告或註解等實際上不需要執行的程式碼。

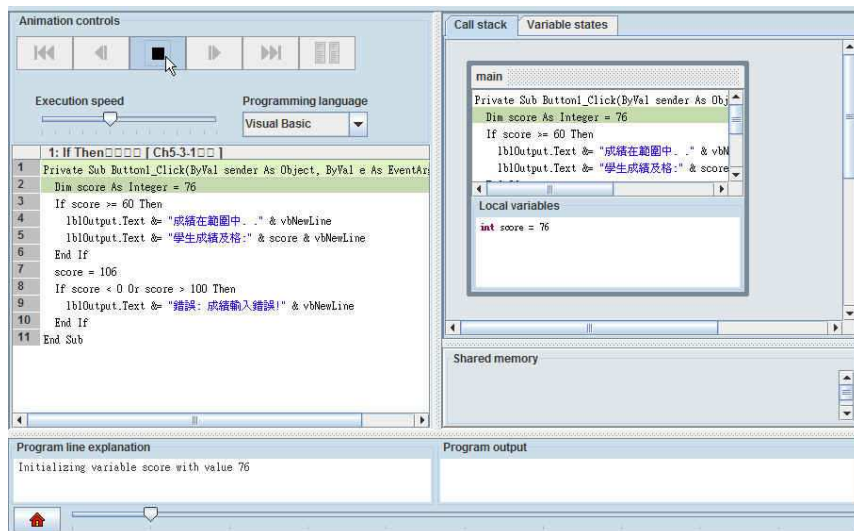
ViLLE 一行一行執行原始程式碼的目的是為了教學，它是以初學者認知的角度來執行程式碼，如同編譯程式編譯程式碼，所有撰寫的程式碼都會一一執行。現在，筆者準備使用一個實例來說明如何在 ViLLE 模擬執行原始程式碼，其步驟如下所示：


**Step 1：**按二下【ViLLE\_VB.html】的 HTML 網頁，就可以啟動 Internet Explorer 執行 ViLLE 的 Java Applet，看到主使用介面。



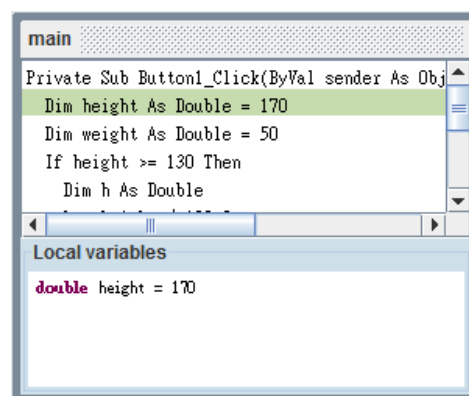


Step 2：展開【條件敘述】的類別目錄後，選【If Then 條件敘述 [ Ch7-2 專案 ]】項目，按游標所在的【Run】鈕進入模擬執行介面。

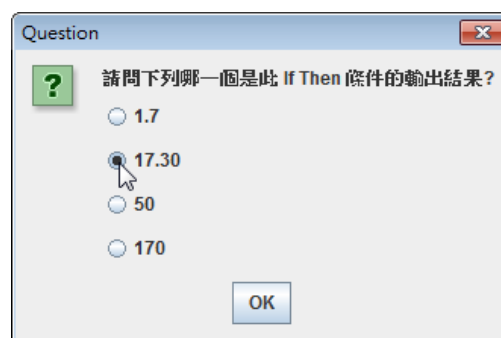


Step 3：在動畫工具列按  鈕自動執行程式碼，可以依序執行原始程式碼，目前執行的程式碼是綠色反白顯示；前一行是淡綠色。

Step 4：當執行至第 2 行時，程式碼宣告變數 **height** 且指定初值 170，在右上方呼叫堆疊的 **main** 視窗（即 **Button1\_Click()** 事件處理程序），同步顯示執行事件處理程序的第 2 行，下方是區域變數 **height** 的值，如下圖所示：



Step 5：當執行到第 4 行時，因為有指定問題，所以顯示一個問題視窗讓使用者回答問題。



Step 6：在輸入答案後，按【OK】鈕，如果答案正確，就顯示正確的訊息視窗。



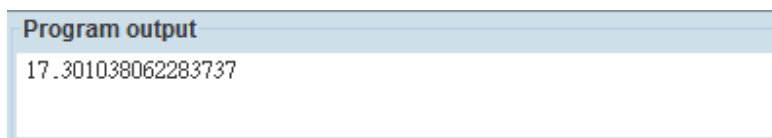
Step 7：如果答案錯誤，就會顯示錯誤的訊息視窗，如下圖所示：



Step 8：按【確定】鈕繼續程式的模擬執行和回答內建問題，直到程式碼的最後一行，就會在最後顯示測驗結果的訊息視窗，顯示問題數和答對題數的比例，以此例是 1 / 2，即 2 題答對 1 題，如下圖所示：



Step 9：按【確定】鈕結束模擬程式的執行，可以在下方程式輸出看到執行結果，如下圖所示：



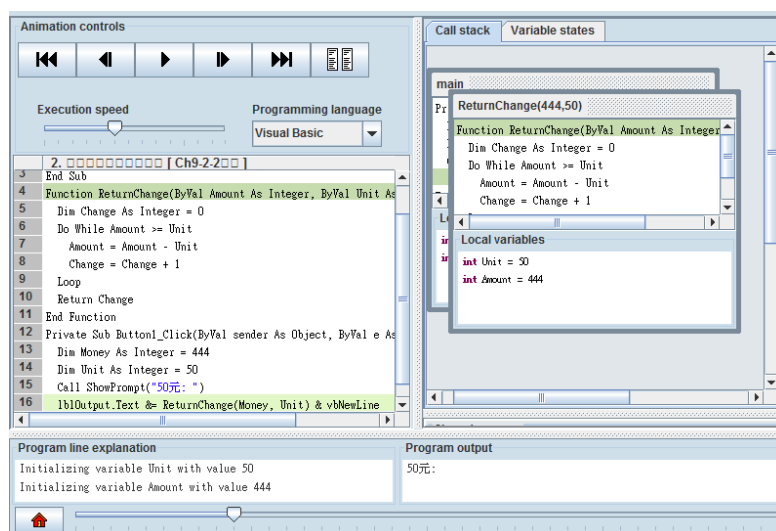
如果在 Internet Explorer 載入【VILLE\_VB\_NQ.html】的 HTML 網頁，就不會顯示執行中的問題視窗，不過，仍然會在最後顯示測驗結果的訊息視窗，此時的答對題數一定是 0。

### E-3-4 再談模擬執行介面

VILLE 如果是模擬執执行程序與函數呼叫的範例程式，在呼叫堆疊會顯示重疊視窗來顯示執行的程序與函數。若是宣告陣列的範例程式，在共享記憶體則會顯示圖形表示的陣列結構。

#### 程序與函數呼叫

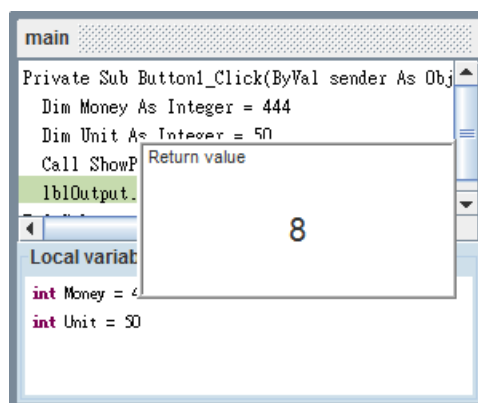
在 VILLE 模擬執行範例程式時，如果執行到程序與函數呼叫時，就會在呼叫堆疊新增一個程序與函數的視窗，標題是程序與函數名稱；內容是程序與函數的程式碼，如下圖所示：



上述圖例在 Button1\_Click()事件處理程序（即主程式 main()）的第 16 行呼叫 ReturnChange()函數，所以在右上方呼叫堆疊顯示 2 個重疊視

窗，後方是 `main`；前方是 `ReturnChange()` 函數，在 `ReturnChange()` 函數視窗的標題列顯示傳入的參數值 444, 50。

此時，程式執行順序也會從呼叫的 `Button1_Click()` 跳至 `ReturnChange()` 函數的程式碼。因為 `ReturnChange()` 函數有傳回值，當執行至第 10 行，就會傳回函數的傳回值，如下圖所示：



上述圖例顯示函數的傳回值是 8，此時執行順序會返回 `Button1_Click()` 事件處理程序的函數呼叫，即第 16 行，然後再繼續執行 `Button1_Click()` 事件處理程序的其他程式碼。

## 陣列結構

ViLLE 模擬執行陣列結構的範例程式時，在共享記憶體視窗會顯示圖形表示的陣列內容，如下圖所示：

Shared memory									
ADDR: #001					ADDR: #002				
"陳會安"	"江小魚"	"陳允傑"	"楊過"	"小龍女"	60	89	75	68	90

上述圖例是 2 個分別擁有 5 個元素的一維整數與字串陣列。如果是二維陣列，例如：2 X 5 的二維整數陣列，如下圖所示：

Shared memory									
ADDR: #001					ADDR: #002		ADDR: #003		
70	79	65	98	60	0	0	0	0	
60	89	75	68	90					

## E-4 參考文獻

1. Rajala, T., Laakso, M.-J., Kaila, E. & Salakoski, T. (2007). VILLE - A language-independent program visualization tool. Proceedings of the Seventh Baltic Sea Conference on Computing Education Research (Koli Calling 2007), Koli National Park, Finland, November 15-18, 2007. Conferences in Research and Practice in Information Technology, Vol. 88, Australian Computer Society. Raymond Lister and Simon, Eds.