

# **2. TOUCH VB**

\*\*\*\*\*

## **2.1 程式的開發流程**

## **2.2 動手開發一個 Visual Basic 程式**

## 2.1 程式的開發流程

撰寫程式是為了解決人們的需求，例如我們寫一個計算圓面積的小程式，將來我們需要計算圓面積時，只須開啓程式，而不必自行去按計算機，不但省時又便利，當然這只是一個小程式，未來我們會接觸到更大更複雜的程式，它所帶來的便利與迅速，非人工可以比擬。

在此我們舉第一章計算圓面積的程式來做例子解說程式開發的流程

### (1)分析問題及需求

以圓面積的例子，最大的差別就是在於圓週率 $\pi$ 並非一個可以整除的數，而是 3.14159.....，所以求出來的圓面積小數點後面必須有幾位數，這便是客戶的『需求』。

### (2)設計演算法或流程圖

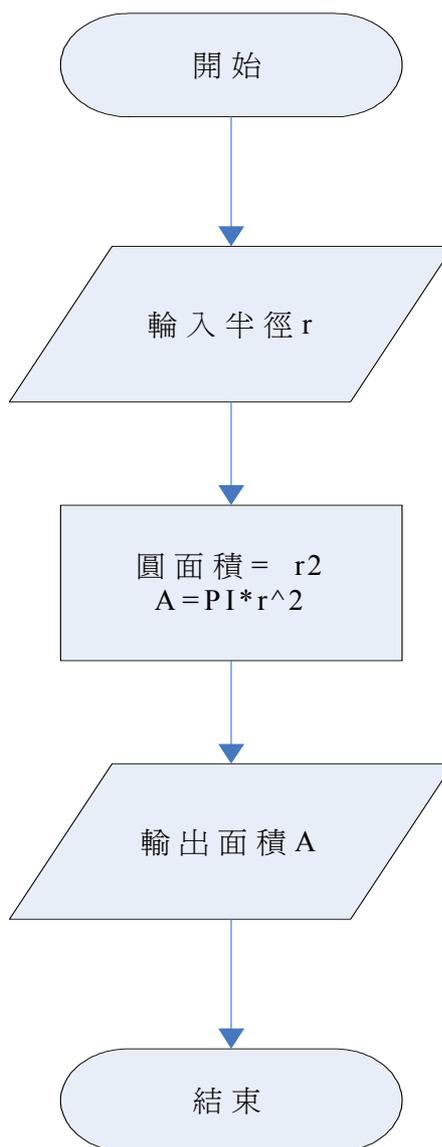
當問題以及需求確定後，程式該如何寫便是程式設計師的責任了，預先將程式的設計流程安排出來，可避免程式撰寫過程中發生錯誤，也可簡化除錯的過程，並且對最後的驗證步驟節省許多時間。

演算法並不是一種真正的程式語言，而是一種來說明如何解決問題方法或步驟，可以使用任何一種程式語言的敘述，並沒有硬性規定。例如以下為計算圓面積程式的演算法及流程圖，讀者可以相互比較選一種較容易懂的方法來使用，在後面的章節將會詳細介紹流程圖。

#### (1)演算法

1. 使用者輸入半徑
2. 圓面積 = 圓周率 \* 半徑平方
3. 輸出面積

## (2) 流程圖



## (3) 設計使用者介面

使用者介面設計的好壞，嚴重影響使用者使用程式的便利性，VB 所擁有的圖形化界面使程式設計師在設計使用者介面時，較為迅速，也可將設計出來的介面立即交給客戶確認是否合適，不合適的話可以立即修改。

如下節範例所示，即可輕易辨別使用者介面設計的好壞。

## (4) 撰寫程式碼

現在我們已經擁有解決問題的流程(演算法或流程圖)，並且也將使用者介面完成了，我們現在必須開始撰寫程式碼，在撰寫程式途中『善用註解』可以幫助你不少的忙，在變數的命名上盡量採取有意義的命名，否則日後需要維護或修改程式碼時，將不知從何處下手。

## (5) 驗證程式的正確性

撰寫完程式後，我們必須核對程式的功能如同當初撰寫時的『需求』相同，核對的方法有很多，可以照著流程圖一步一步的來做測試，以確保程式功能達到預期的要求。例如求圓面積的程式，我們可以用一組數據輸入程式，看看其結果與我們按計算機算出來的結果是否相同，但是程式的問題有時是在使用中才會發生，使用者可能會以不正確的方式使用程式，假設我們要求使用者輸入一個『數字』可是使用者可能會不小心按到 A，造成程式發生錯誤，或是無法預知的結果，程式設計師有責任在設計程式時，去避免這類型的錯誤。

## 2.2 動手開發一個 Visual Basic 程式

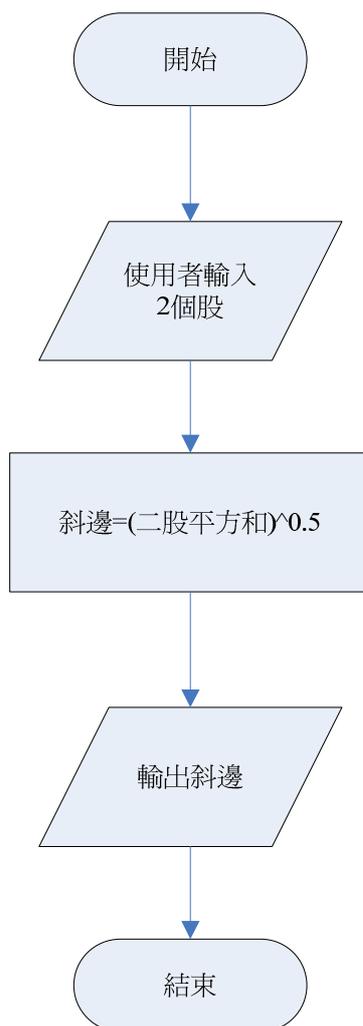
前面已經有一個計算圓面積的範例程式，現在請讀者跟著我們一起動手來寫一個『畢氏定理』的小程式，照著上一章節所介紹開發程式的流程，親自動手撰寫，學習程式就和學習數學是一樣的，必須親自動手才能夠了解其中的奧妙之處。

### 2.2.1 問題分析

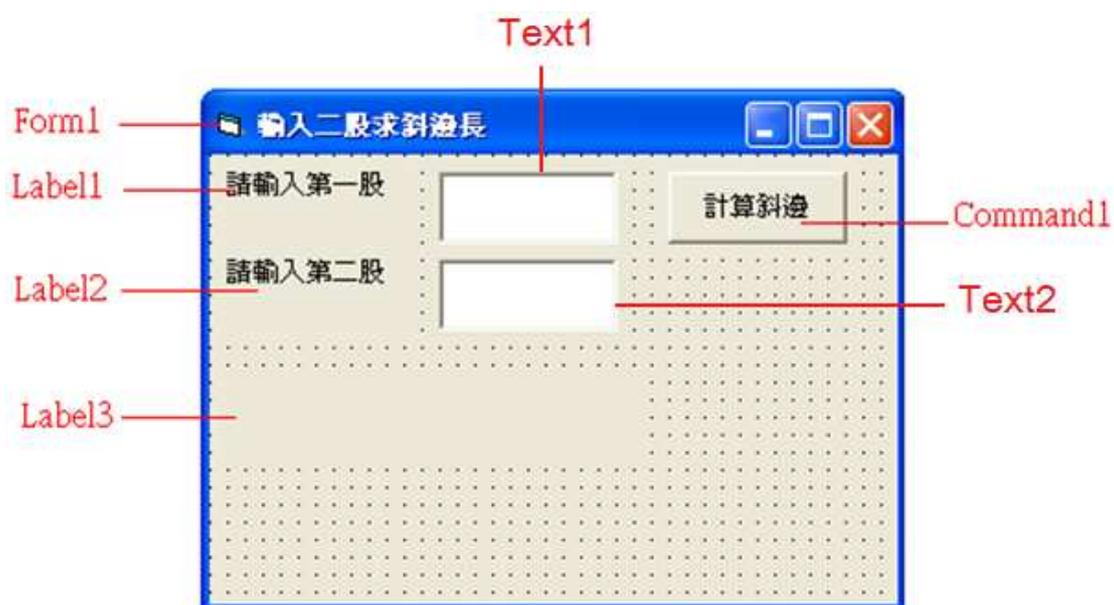
我們要了解程式要做哪些工作，例如是要輸入二股求斜邊，或是輸入斜邊和其中一股，求另一股。若確定要輸入二股求斜邊時，那麼程式應該有以下的能力。

1. 可以輸入二個股
2. 輸出斜邊

### 2.2.2 設計流程



### 2.2.3 設計使用者介面

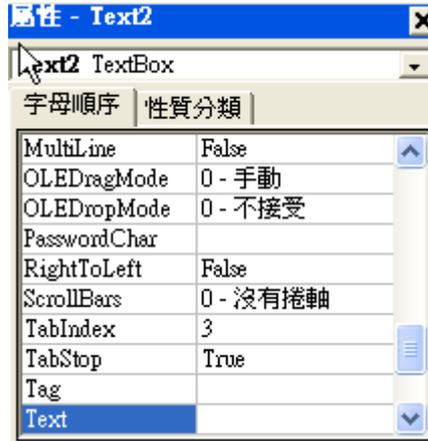
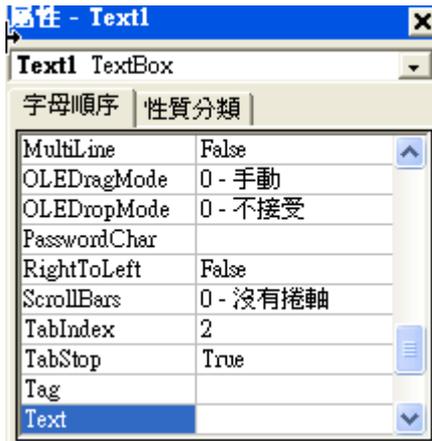


屬性 - Form1	
Form1 Form	
字母順序	性質分類
(Name)	Form1
Appearance	1 - 立體
AutoRedraw	False
BackColor	<input type="checkbox"/> &H8000000F&
BorderStyle	2 - 大小可調整
Caption	輸入二股求斜邊長
ClipControls	True
ControlBox	True
DrawMode	13 - Copy Pen
DrawStyle	0 - 實線

屬性 - Label1	
Label1 Label	
字母順序	性質分類
(Name)	Label1
Alignment	0 - 靠左對齊
Appearance	1 - 立體
AutoSize	False
BackColor	<input type="checkbox"/> &H8000000F&
BackStyle	1 - 不透明
BorderStyle	0 - 沒有框線
Caption	請輸入第一股
DataField	
DataFormat	

屬性 - Label2	
Label2 Label	
字母順序	性質分類
(Name)	Label2
Alignment	0 - 靠左對齊
Appearance	1 - 立體
AutoSize	False
BackColor	<input type="checkbox"/> &H8000000F&
BackStyle	1 - 不透明
BorderStyle	0 - 沒有框線
Caption	請輸入第二股
DataField	
DataFormat	

屬性 - Label3	
Label3 Label	
字母順序	性質分類
(Name)	Label3
Alignment	0 - 靠左對齊
Appearance	1 - 立體
AutoSize	False
BackColor	<input type="checkbox"/> &H8000000F&
BackStyle	1 - 不透明
BorderStyle	0 - 沒有框線
Caption	
DataField	
DataFormat	



## 2.2.4 編輯程式碼

---

```
Private Sub Command1_Click()
```

```
Dim a, b As Integer '變數宣告，a、b 代表二股
```

```
Dim c As Single '變數宣告，c 代表斜邊，斜邊不一定是整數，故宣告為 single
```

```
a = Val(Text1.Text) '將第一股的值以數值型態指定給變數 a
```

```
b = Val(Text2.Text) '將第二股的值以數值型態指定給變數 b
```

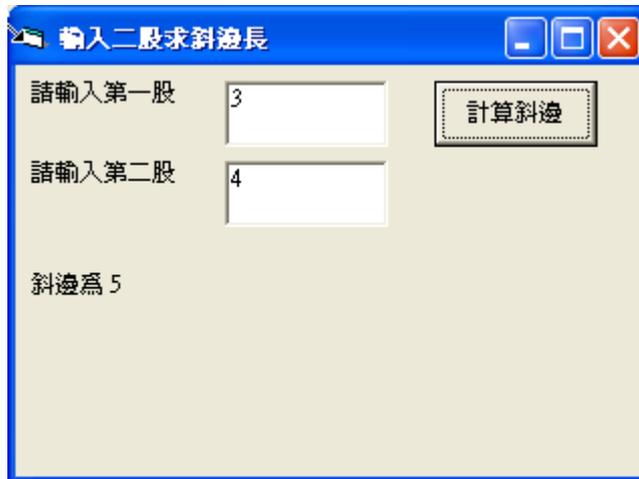
```
c = (a ^ 2 + b ^ 2) ^ 0.5 '計算斜邊
```

```
Label3.Caption = "斜邊為" + Str(c) '輸出斜邊(因為要和字串一起輸出，
    所以必須將數值轉成字串)
```

```
End Sub
```

---

## 2.2.5 執行程式並進行測試



若結果正確則可將程式存檔，若發現有錯誤則再次進行修正，直到結果正確為止

## 2.2.6 存檔

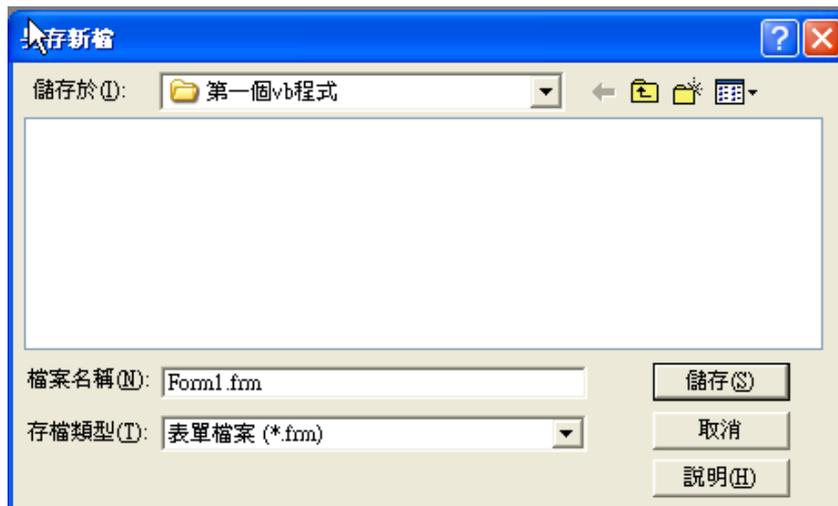
如欲存檔，請按主視窗上的 ，即可儲存專案



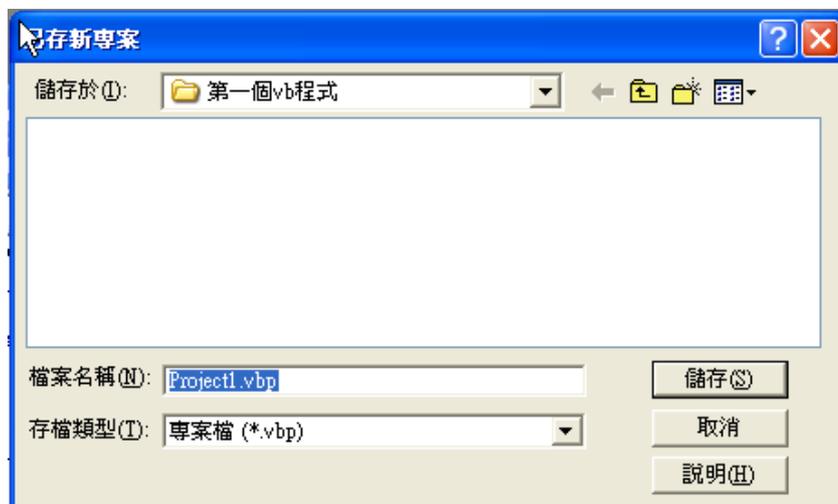
1. 按下存檔後請選擇一個欲存放檔案的目錄



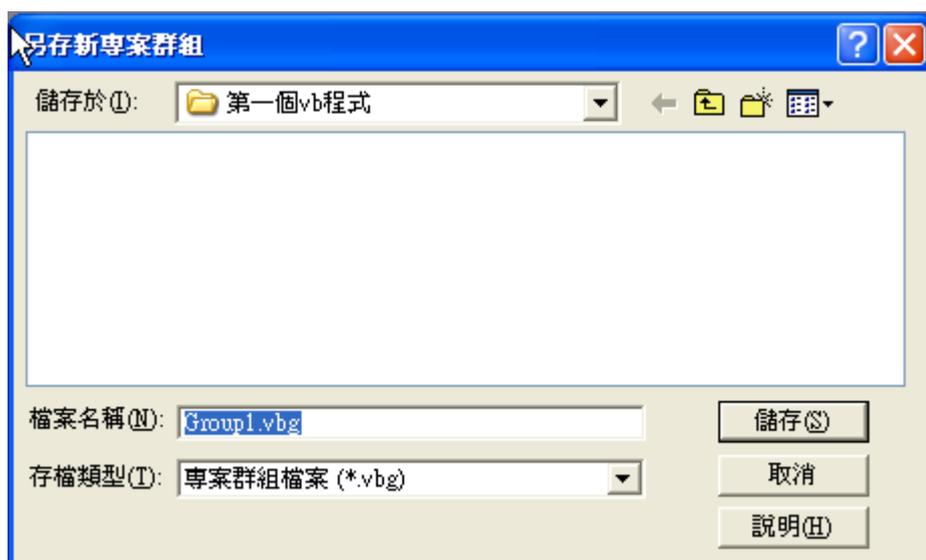
2. 存檔時, 可將 Form1 改成較易辨認的檔名



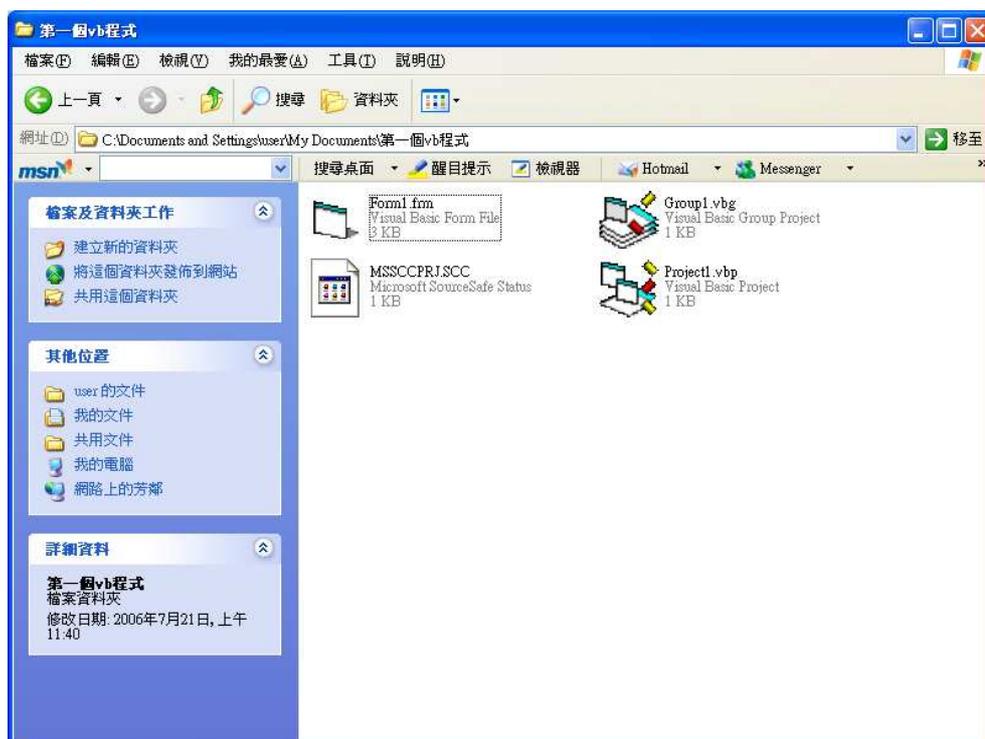
3. 儲存專案檔, 也可以將 Project1 改成喜歡的檔名



4. 儲存專案群組, Group1 也可以更改成自己喜歡的名字



5.存檔完後，可到檔案所在的資料夾查看是否有正確儲存

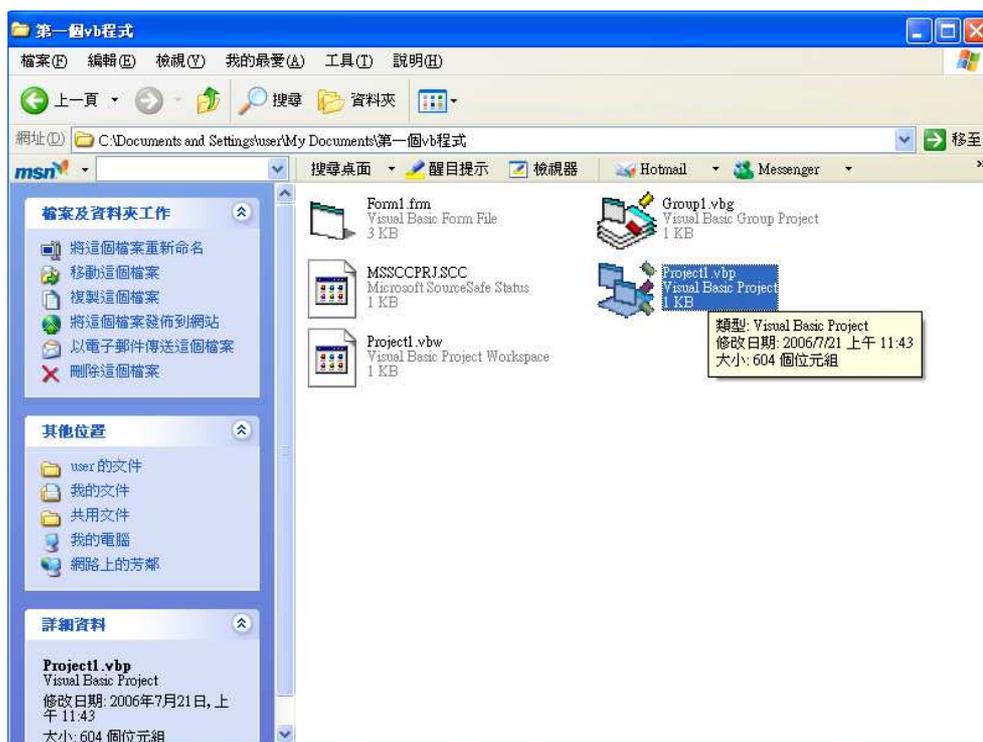


### 2.2.7 開啓舊檔

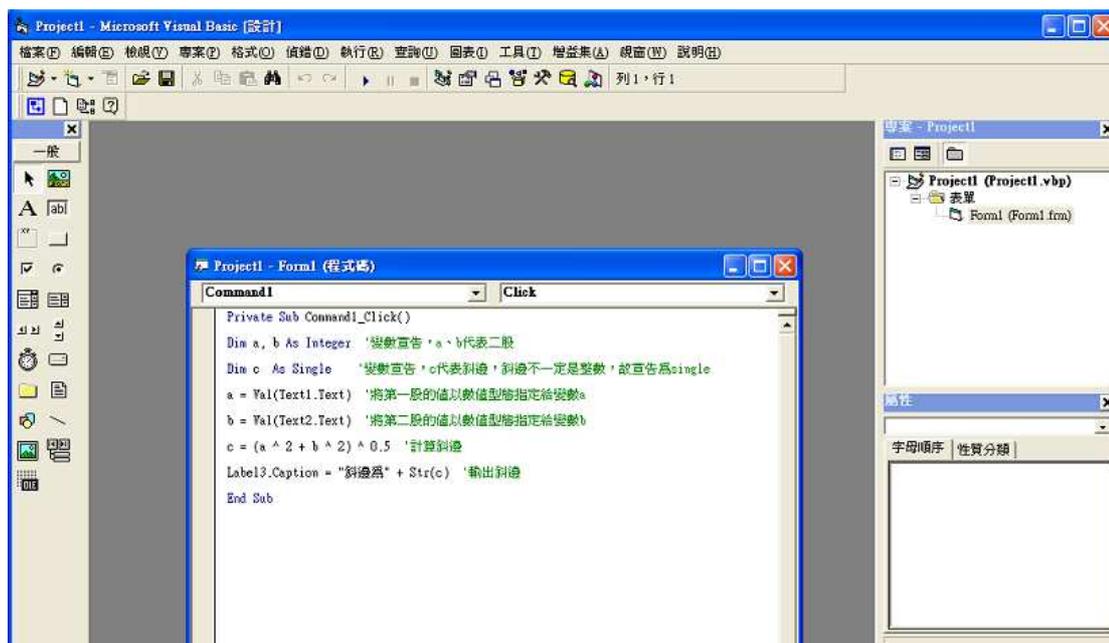
有二種方式可以開啓已存檔的程式，請詳見以下說明

1.直接從檔案開啓

(1).用滑鼠左鍵點二下副檔名為 vbp 的專案檔

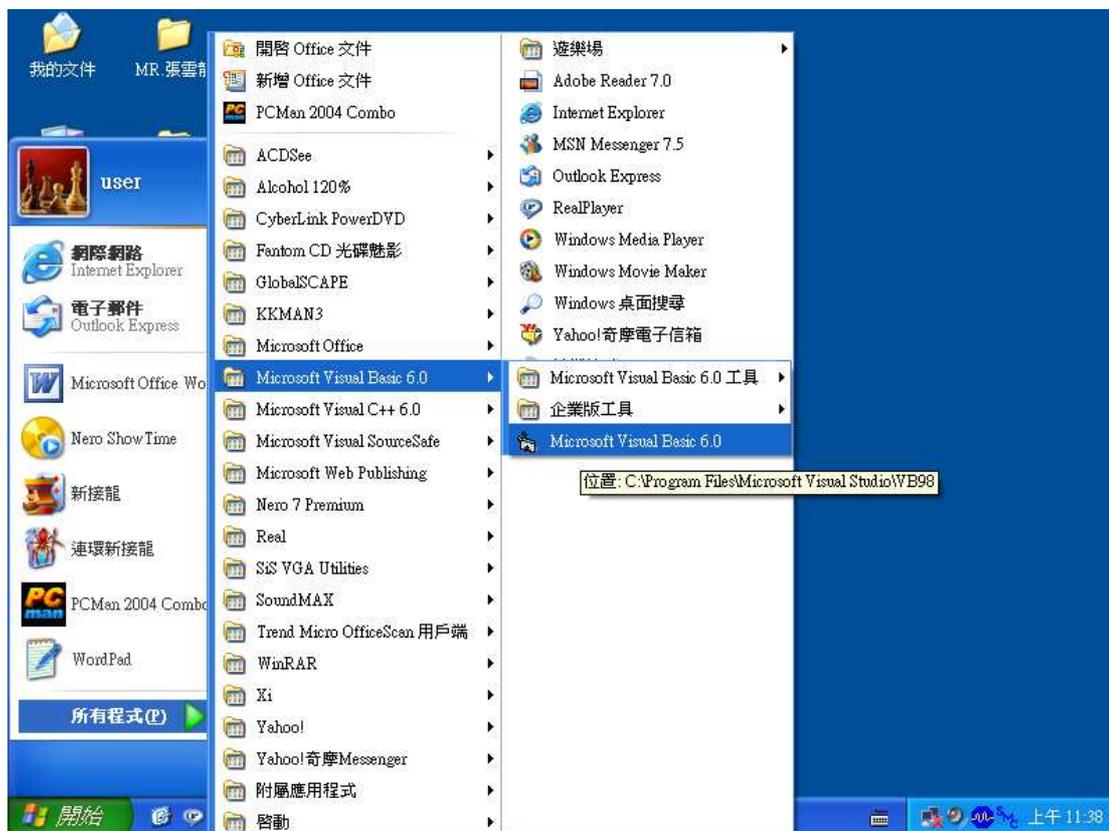


(2).系統即會啓動 VB 來開啓此專案



2.用 VB 開啓舊檔

(1)開啓 VB



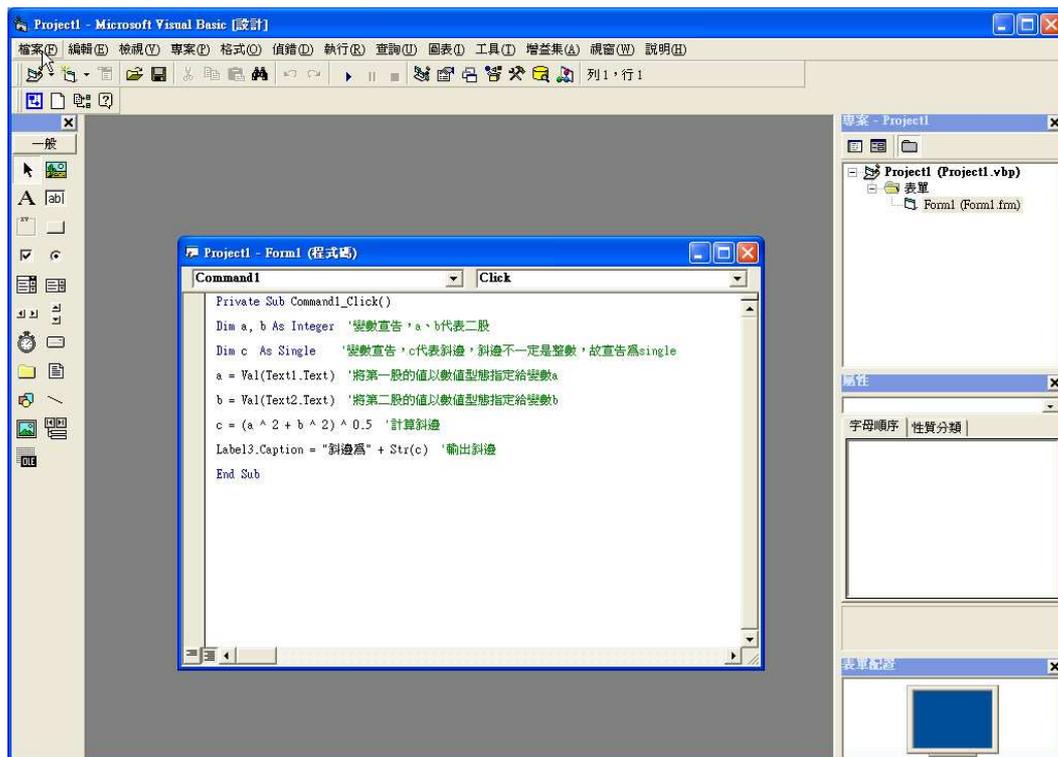
(2)點選『開啓』



(2)選取檔案所在的資料夾，並點選副檔名為 vbp 的專案檔，按下『開啓』

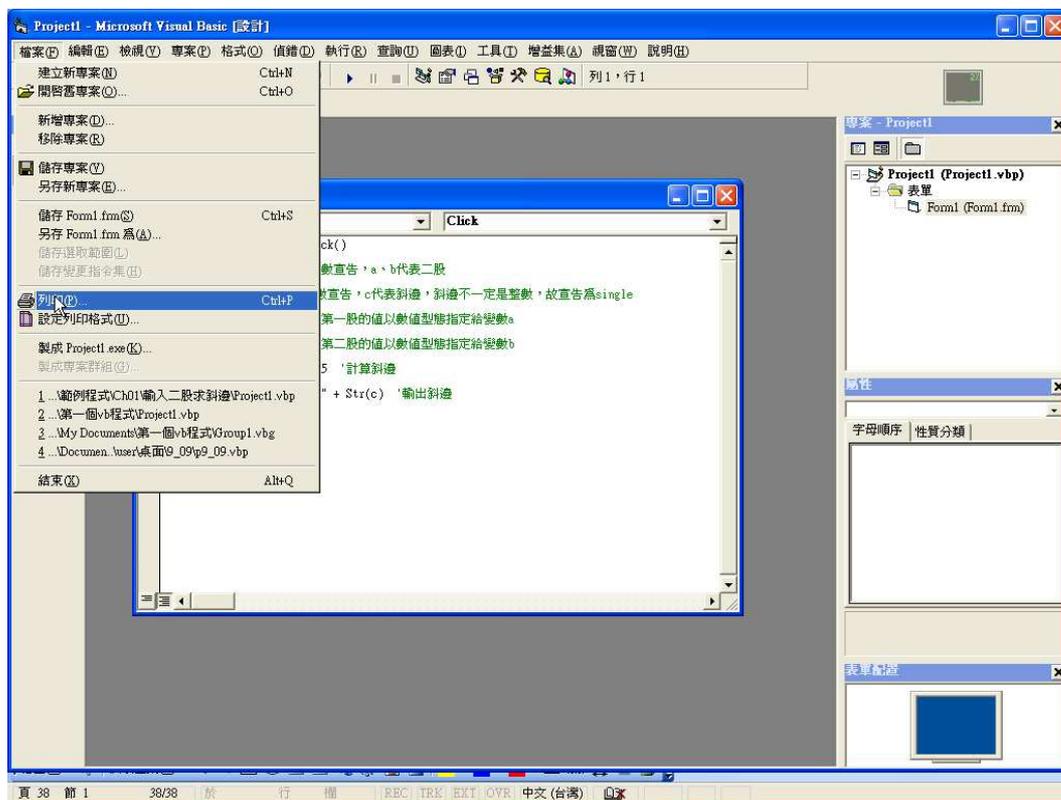


### (3).VB 已經此專案開啓



### 2.2.7 列印表單及程式碼

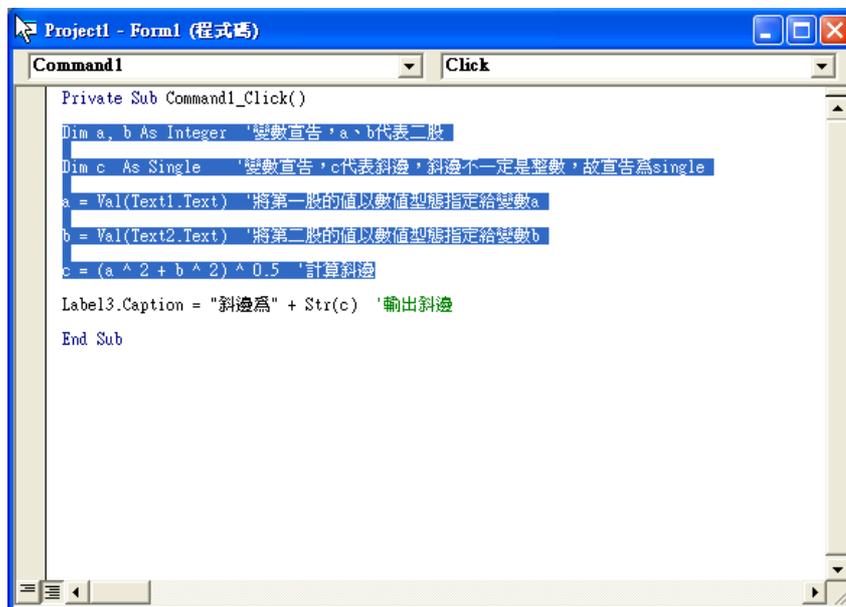
1.開啓欲列印的專案後，選取工作列上的『檔案』→『列印』



## 2.出現列印設定視窗



## 1. 列印出程式碼中反白選取的地方



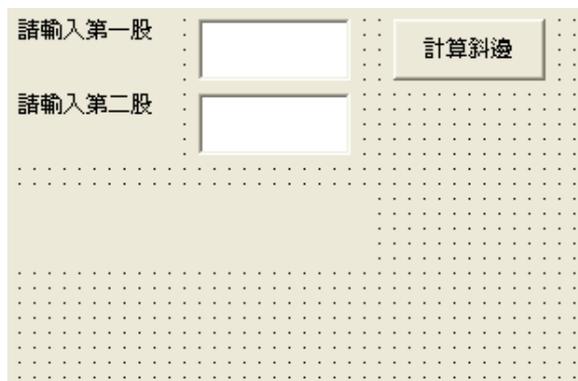
## 2. 列印出目前正在編輯的表單或模組



### 3. 列印目前選取的專案



### 4. 列印出表單的內容



### 5. 列印出我們所撰寫的程式碼

```
Private Sub Command1_Click()
```

```
Dim a, b As Integer '變數宣告，a、b 代表二股
```

```
Dim c As Single '變數宣告，c 代表斜邊，斜邊不一定是整數，故宣告為 single
```

```
a = Val(Text1.Text) '將第一股的值以數值型態指定給變數 a
```

```
b = Val(Text2.Text) '將第二股的值以數值型態指定給變數 b
```

```
c = (a ^ 2 + b ^ 2) ^ 0.5 '計算斜邊
```

```
Label3.Caption = "斜邊為" + Str(c) '輸出斜邊
```

```
End Sub
```

## 6.列印出表單中控制項的屬性(以文字敘述)

---

---

Form1 -1

VERSION 5.00

Begin VB.Form Form1

Caption = “輸入二股求斜邊長”

ClientHeight = 3090

ClientLeft = 60

ClientTop = 450

ClientWidth = 4680

LinkTopic = “Form1”

ScaleHeight = 3090

ScaleWidth = 4680

StartPosition = 3 ‘系統預設值

Begin VB.CommandButton Command1

Caption = “計算斜邊”

Height = 495

Left = 3120

TabIndex = 4

Top = 120

Width = 1215

End

Begin VB.TextBox Text2

Height = 495

Left = 1560

TabIndex = 3

Top = 720

Width = 1215

End

Begin VB.TextBox Text1

Height = 495

Left = 1560

TabIndex = 2

Top = 120

Width = 1215

End

```

Begin VB.Label Label3
    Height = 495
    Left = 120
    TabIndex = 2
    Top = 1560
    Width = 1215
End
Begin VB.Label Label2
    Caption = "請輸入第二股"
    Height = 495
    Left = 120
    TabIndex = 5
    Top = 720
    Width = 1215
End
Begin VB.Label Label1
    Caption = "請輸入第一股"
    Height = 495
    Left = 120
    TabIndex = 0
    Top = 120
    Width = 1215
End

```



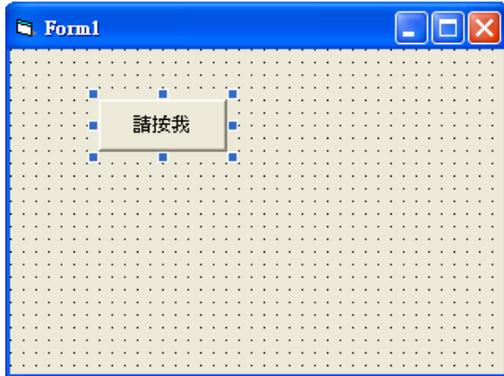
小幫手特區：屬性這麼多，怎麼辦!!!

其實 VB 的屬性雖多，但在屬性視窗下方會有一個地方專門用來告知使用者

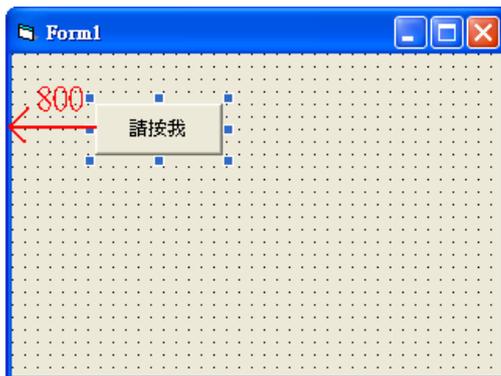


這個屬性功能為何，只要用心去嘗試幾次，自然知道其中的奧妙之處，以下我們將以圖解的方式介紹常用的幾個屬性。

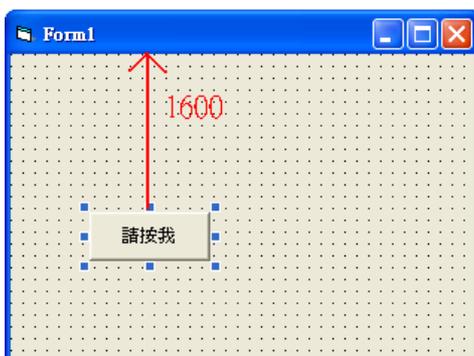
1.Caption (標題)：用以改變 Command1、Form、Label 上所顯示的文字



2.Left(距離表單最左方有多遠)



4. Top(距離表單最上方有多遠)



## 5. Width(寬)



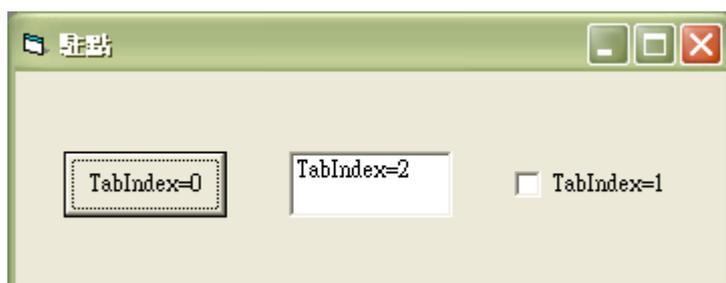
## 6. Height(高)



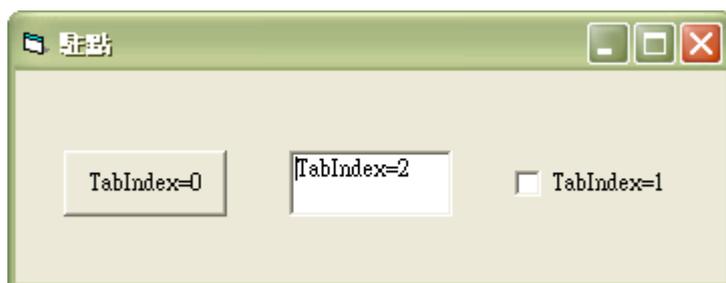
7. TabIndex(駐點)：當我們在使用程式時，雖然能夠同時聽歌、上網，但我們同一時間只能用使滑鼠或鍵盤去控制其中一個程式，而我們所控制的程式在那一段時間內便取得駐點。而在表單中同時存在許多控制項，當您按下 Tab 鍵時，會跳往下一個駐點，若將性質相近的控制項駐點設定為連續的數值時(0、1、2、3、4、5...)，操做起來也較方便。



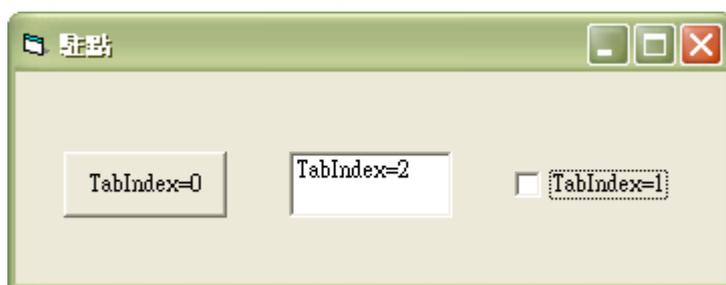
剛開始駐點在 TabIndex=0



按下 Tab 鍵後，駐點就跑到 TabIndex=1



再按一次 Tab 鍵，駐點就跑到 TabIndex=2 了



※所以我們如果把有相關聯的控制項駐點設為鄰近的值，將有助於操作上的便利性，例如輸入完帳號，按下 Tab 鍵，我們當然希望直接跳到輸入密碼的格子，But~有些控制項 TabIndex 並沒有作用，像是 label、frame . . . 等等，因為它們不能被輸入

8. Text(文字)：此屬性是文字方塊(textbox)特有的，您可以發現文字方塊並沒有 caption 屬性，其特性和 caption 類似，都是將欲顯示的資訊輸入其中，即可顯示在螢幕上。