

# 流程控制(二)重複結構



- ◆ 學習 For...Next 迴圈的使用時機和用法
- ◆ 學習巢狀 For...Next 迴圈的用法
- ◆ 學習取得工作表資料的最下列和最右欄
- ◆ 學習 Do...Loop 和 For...Next 迴圈的差異
- ◆ 學習 Do While...Loop 和 Do...Loop While 迴圈
- ◆ 學習 Do Until...Loop 和 Do...Loop Until 迴圈
- ◆ 學習前測式和後測式 Do...Loop 迴圈的差異
- ◆ 學習 While...Wend 和 Do...Loop 無窮迴圈
- ◆ 學習在儲存格中寫入公式和函數

## 5.1 前言

「結構化程式」是程式設計的基本原則，而程式是由循序結構、選擇結構和重複結構三種基本邏輯架構組成。其中「循序結構」是程式的最基本結構，是由上到下依序逐行執行敘述構成。至於上一章介紹的「選擇結構」會依照條件，分別執行不同流程的敘述區段。

設計程式時，常碰到需將某個程式區段重複執行多次，此時就需要「重複結構」來完成。「重複結構」俗稱為「迴圈」(Loop)，譬如： $1 + 2 + 3 + \dots + 1000$  計算其總和、從 A1 到 A100 連續讀取儲存格內資料等，這些重複地執行煩人的固定性工作，可透過「重複結構」來完成，即使執行多少次也不會出錯。VBA 常用的重複結構：有 For...Next 和 Do...Loop 兩種迴圈，將於本章逐一介紹。

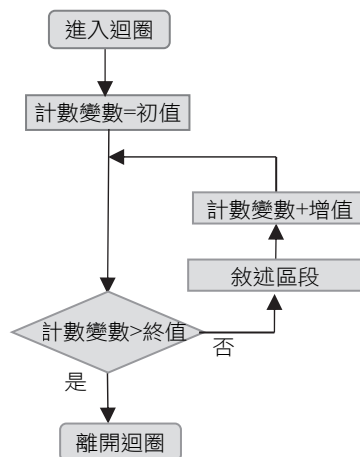
## 5.2 For...Next 迴圈

程式中若有某個「敘述區段」需要反覆執行指定的次數時，就可以使用 For...Next 迴圈來完成。For...Next 迴圈是由計數變數、初值、終值以及增值所構成。For...Next 迴圈的語法如下：

語法：

```
For 計數變數 = 初值 To 終值 [Step 增值]
    ⋮
    敘述區段
    ⋮
[Exit For]
Next
```

流程圖：



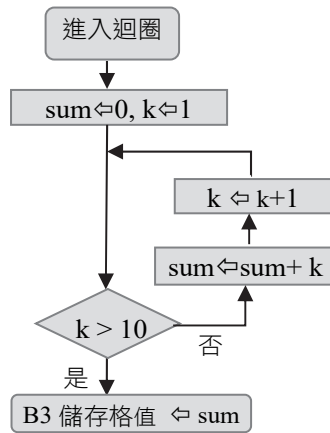
► 說明

1. For...Next 迴圈中的計數變數必須是數值資料型別的變數，而初值、終值和增值則可以為數值變數、數值常值或數值運算式。增值不能為零，若省略不寫則預設增值為 1。
2. For...Next 迴圈中，若初值小於終值，增值必須為正，當計數變數大於終值時離開迴圈；若初值大於終值時，則增值必須為負值，計數變數在小於終值時離開迴圈。
3. 如果想要提早離開 For...Next 迴圈，則可以使用 Exit For 敘述配合 If 條件式來判斷是否提前離開 For 迴圈。
4. For ...Next 迴圈內的敘述，應按<Tab>鍵來讓敘述區段內縮以方便閱讀。

► 簡例

1. 將  $1 + 2 + 3 + \dots + 10$  的總和存入 B3 儲存格。

① 流程圖



假設：

1. 計數變數 k
2. 初值 = 1
3. 終值 = 10

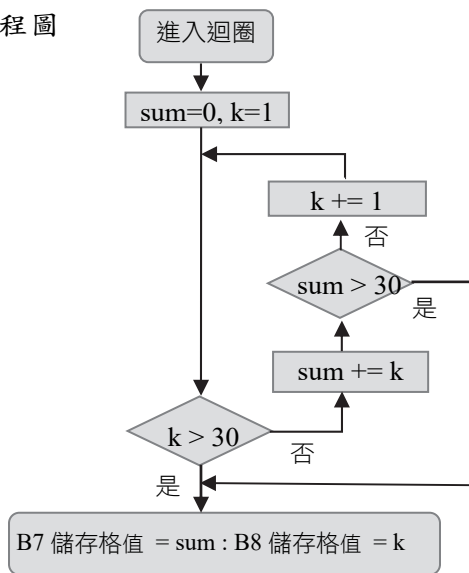
② 程式碼

```

Dim k As Integer, sum As Integer
sum = 0
For k = 1 To 10 Step 1
    sum = sum + k
Next
Range("B3").Value = sum
  
```

2.  $1 + 2 + 3 + \dots + k \leq 30$ ，總和超過 30 就離開迴圈，求 k 和 sum。

① 流程圖



假設：

1. 計數變數 k
2. 初值 = 1
3. 終值 = 10

② 程式碼

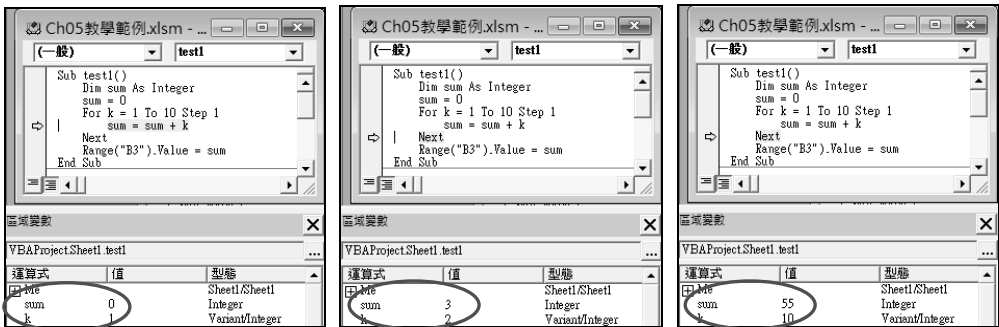
```

Dim sum As Integer, k As Integer
sum = 0
For k = 1 To 10 Step 1
    sum = sum + k
    If sum > 30 Then Exit For
Next
Range("B7").Value = sum
Range("B8").Value = k
  
```

► 隨堂測驗

1. 試將  $11 + 9 + \dots + 3 + 1 = ?$  的運算式，使用 For...Next 迴圈敘述來撰寫程式？
2. 試將  $0.5 + 1.0 + \dots + 4.5 + 5 = ?$  的運算式，使用 For...Next 迴圈敘述來撰寫程式？

在前面第二章中我們介紹了程式除錯的方法，我們藉此來了解 For...Next 敘述的運作過程。先在 VBA 程式編輯器中執行功能表【檢視/區域變數視窗】指令，來開啟區域變數視窗。然後按 <F8> 鍵來逐行執行程式，就可以透過區域變數視窗來觀察 For...Next 迴圈的運作流程，以及相關變數的變化情形。



**實作** FileName : For.xlsxm

設計一個迴圈測試程式，可輸入 For...Next 迴圈的初值、終值和增值，執行後會計算出總和並顯示。

► 輸出要求

	A	B	C
1	初值(整數):	1	執行
2	終值(整數):	100	
3	增值(整數):	1	
4	總和:	5050	

	A	B	C
1	初值(整數):	100	執行
2	終值(整數):	1	
3	增值(整數):	-5	
4	總和:	1050	

► 解題技巧

Step 1 建立輸出入介面

1. 新增活頁簿並以「For」為新活頁簿名稱。
2. 在工作表 1 中建立如下表格，和 ActiveX 命令按鈕控制項：

	A	B	C
1	初值(整數)：		執行
2	終值(整數)：		
3	增值(整數)：		
4	總和：		

### Step 2 問題分析

1. 從儲存格讀取使用者輸入的初值、終值和增值，並分別存放在 start\_num(初值)、end\_num(終值)和 step\_num (增值)變數中。
2. 將 start\_num(初值)、end\_num(終值)和 step\_num (增值)變數，代入 For...Next 迴圈中計算出總和。

```
For i = start_num To end_num Step step_num
    sum += i
Next
```

### Step 3 編寫程式碼

FileName: For.xlsm (工作表 1 程式碼)

01 Private Sub CommandButton1 Click()

02 Dim start\_num, end\_num, step\_num, sum As Integer

03 start\_num = Range("B1").Value

04 end\_num = Range("B2").Value

05 step\_num = Range("B3").Value

06 sum = 0 '預設總和為 0

07 For i = start\_num To end\_num Step step\_num

08 sum = sum + i

09 Next

10 Range("B4").Value = sum

11 End Sub

### ► 隨堂測驗

將上面實作增加增值的檢查，以避免程式執行時的錯誤。第一增值不能為零、第二當終值大於初值時，增值必須大於零、第三終值小於初值時，增值必須小於零。

	A	B	C
1	初值(整數):	1	執行
2	終值(整數):	10	執行
3	增值(整數):	0	
4	總和:	增值不能為零	

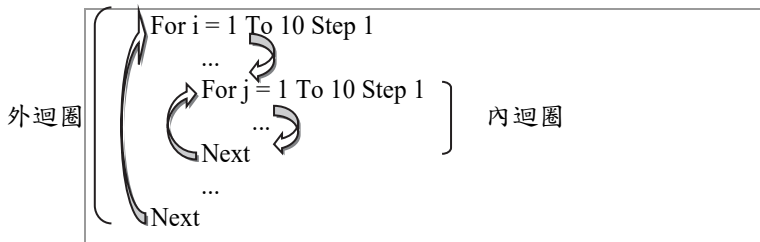
	A	B	C	D	E
1	初值(整數):	1	執行		
2	終值(整數):	10	執行		
3	增值(整數):	-1			
4	總和:	增值小於零時, 終值必須小於初值			

	A	B	C	D	E
1	初值(整數):	10	執行		
2	終值(整數):	1	執行		
3	增值(整數):	1			
4	總和:	增值大於零時, 終值必須大於初值			

### 5.3 巢狀 For...Next 迴圈

#### 5.3.1 巢狀 For...Next 迴圈

所謂「巢狀 For...Next 迴圈」, 就是指一個 For...Next 迴圈內的敘述區段, 還有 For...Next 敘述區段。執行時先進入外迴圈內將內迴圈的敘述區段執行一次, 再回到外迴圈最前面比較計數變數和終值是否超出範圍? 若未超出範圍再進入外迴圈內, 將內迴圈敘述區段的敘述再執行一次, 如此反覆執行到超出範圍才離開外迴圈。使用巢狀迴圈時, 要特別注意每個 For 都必須有對應的 Next, 迴圈彼此之間是不允許相互交錯, 所以編寫時要使用縮排。



巢狀迴圈中的每個 For 迴圈, 都必須使用自己的計數變數, 不可以重複使用。通常巢狀 For 迴圈使用於二維有規則性的表格, 可以逐列逐欄一一寫入或是讀取儲存格的資料。例如: 製作一個九九乘法表, 統計儲存格範圍的數值總和等。

## ► 簡例

每列都顯示五個 "\*" 星號，共三個水平列。可以將 r 設為外部迴圈的計數變數，c 為內部迴圈的計數變數其演算法如下：

- Step ① 設定 r 變數由 1~3 表水平列數，c 變數由 1~5 表每列印出的個數。
- Step ② r = 1 即第 1 列時，c = 1~5 執行 5 次，每次設儲存格 Cells(1, c) 的值为 "\*"。
- Step ③ r = 2 即第 2 列時，c = 1~5 執行 5 次，每次設儲存格 Cells(2, c) 的值为 "\*"。
- Step ④ r = 3 即第 3 列時，c = 1~5 執行 5 次，每次設儲存格 Cells(3, c) 的值为 "\*"。

```
For r = 1 To 3
  For c = 1 To 5
    Cells(r, c).Value = "*"
  Next
Next
```

### 【輸出結果】

	A	B	C	D	E
1	*	*	*	*	*
2	*	*	*	*	*
3	*	*	*	*	*

## 實作

FileName : One2Six.xlsm

設計一個可以顯示如下圖的 1 至 6 漸增的程式。

## ► 輸出要求

	A	B	C	D	E	F
1	1					
2	1	2				
3	1	2	3			
4	1	2	3	4		
5	1	2	3	4	5	
6	1	2	3	4	5	6



► 解題技巧

Step 1 建立輸出入介面

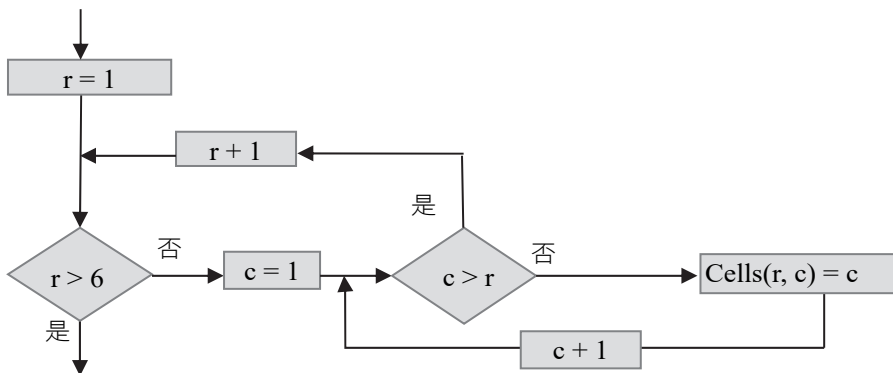
1. 新增活頁簿並以「One2Six」為新活頁簿名稱。
2. 在工作表 1 中建立一個 ActiveX 命令按鈕控制項。

Step 2 問題分析

1. 列(橫向)和欄(縱向)都是由 1 到 6，因此可以使用巢狀迴圈。外圈 r 為水平列(橫向)執行 6 次(r=1~6)，內圈 c 為垂直欄(縱向)由 1 到 6 漸增，所以要將內圈的終值設為 r。

r=1 : c = 1 To 1 ⇒ 顯示 c 值 1 次  
 r=2 : c = 1 To 2 ⇒ 顯示 c 值 2 次  
 r=3 : c = 1 To 3 ⇒ 顯示 c 值 3 次  
 r=4 : c = 1 To 4 ⇒ 顯示 c 值 4 次  
 r=5 : c = 1 To 5 ⇒ 顯示 c 值 5 次  
 r=6 : c = 1 To 6 ⇒ 顯示 c 值 6 次

2. 因為要寫入到儲存格當中，儲存格以 Cells 方式來指定會比較簡便。Cells() 內第一個參數是指定列數，第二個參數是指定欄數，例如：Cells(3, 2)就等於 B3 儲存格。



## Step 3 編寫程式碼

FileName: One2Six.xlsm (工作表 1 程式碼)

```
01 Private Sub CommandButton1_Click()
```

```
02     For r = 1 To 6
```

```
03         For c = 1 To r
```

```
04             Cells(r, c).Value = c
```

```
05         Next
```

```
06     Next
```

```
07 End Sub
```

## ► 隨堂測驗

設計一個可以顯示如右圖的 1 至 6 漸減的程式。

	A	B	C	D	E	F
1	1	2	3	4	5	6
2	1	2	3	4	5	
3	1	2	3	4		
4	1	2	3			
5	1	2				
6	1					

## 5.3.2 取得工作表資料的最下列和最右欄

當我們要處理 Excel 工作表中的資料時，常常需要知道最下一列資料的列號，以及最右一欄資料是第幾欄，以便設定 For ... Next 迴圈的終值。此時可以使用儲存格的 End 屬性，可以由 A1 儲存格開始沿 xlDown 方向找到最下一列，再配合 Row 屬性取得列號。沿 xlToRight 方向找到最右一欄，再配合 Column 屬性取得是第幾欄。End 屬性的詳細用法請參閱第十章第二節的內容。例如：要取得工作表最下一列的列號，和最右一欄是第幾欄，程式寫法為：

```
rNum = Range("A1").End(xlDown).Row
cNum = Range("A1").End(xlToRight).Column
```



**實作** FileName: BackColor.xlsm

在公司存貨月報表上按  鈕，會自動檢查表格的範圍，然後每隔一列填入紫色作為底色，以方便閱讀表格。

► 輸出要求

	A	B	C	D	E	F	G
1	存貨名稱	月初數量	本月進貨數量	本月出貨數量	本月結存數量		
2	巧克力棒隨手杯	1000	500	856	644	填底色	
3	草莓夾心餅乾	2500	1200	2138	1562		
4	巧克力餅乾	1200	400	941	659		
5	純麥蘇打餅乾	800	300	514	586		
6	健康蔬果脆片	600	200	437	363		
7	蔓越莓乾顆粒	1500	400	1345	555		
8	瑞士牛乳餅	700	100	549	251		
9	鮮蝦風味脆餅	400	200	462	138		
10	巧克力酥片	900	300	1098	102		
11							

► 解題技巧

Step 1 建立輸出入介面

1. 新增活頁簿並以「BackColor」為新活頁簿名稱。
2. 將範例 ch05 資料夾 BackColor 資料.xlsx 的表格複製到工作表 1 中，再建立一個 ActiveX 命令按鈕控制項。

	A	B	C	D	E	F	G
1	存貨名稱	月初數量	本月進貨數量	本月出貨數量	本月結存數量		
2	巧克力棒隨手杯	1000	500	856	644	填底色	
3	草莓夾心餅乾	2500	1200	2138	1562		
4	巧克力餅乾	1200	400	941	659		
5	純麥蘇打餅乾	800	300	514	586		
6	健康蔬果脆片	600	200	437	363		
7	蔓越莓乾顆粒	1500	400	1345	555		
8	瑞士牛乳餅	700	100	549	251		
9	鮮蝦風味脆餅	400	200	462	138		
10	巧克力酥片	900	300	1098	102		
11							

Step 2 問題分析

1. 使用 End 屬性取得表格的最下列列號，以及最右欄是第幾欄
2. 因為是隔列填底色，所以 For 迴圈中的增值設為 2。
3. 儲存格範圍可以使用 Range(Cells(列 1, 欄 1), Cells(列 2, 欄 2))來表示，其中 Cells(列 1, 欄 1)為左上角儲存格、Cells(列 2, 欄 2) 為右下角儲存格。
4. 使用儲存格的 Interior.Color 屬性可以設定底色，詳細用法請參閱第十章。

## Step 3 編寫程式碼

FileName: BackColor.xlsm (工作表 1 程式碼)

01 Private Sub CommandButton1\_Click()

02 Dim rNum As Integer

03 Dim cNum As Integer

04 rNum = Range("A1").End(xlDown).Row '取得最下列是第幾列

05 cNum = Range("A1").End(xlToRight).Column '取得最右欄是第幾欄

06 For r = 2 To rNum Step 2 '從第二列開始隔列填色

07 Range(Cells(r, 1), Cells(r, cNum)).Interior.Color = RGB(255, 0, 255)


08 Next

09 End Sub

## ► 隨堂測驗

將上面實作改成隔欄填入底色。

	A	B	C	D	E	F	G
1	存貨名稱	月初數量	本月進貨數量	本月出貨數量	本月結存數量	填底色	
2	巧克力棒隨手杯	1000	500	856	644		
3	草莓夾心餅乾	2500	1200	2138	1562		
4	巧克力餅乾	1200	400	941	659		
5	純麥蘇打餅乾	800	300	514	586		
6	健康蔬果脆片	600	200	437	363		
7	蔓越莓乾顆粒	1500	400	1345	555		
8	瑞士牛乳餅	700	100	549	251		
9	鮮蝦風味脆餅	400	200	462	138		
10	巧克力酥片	900	300	1098	102		
11							



### 實作

FileName: Print.xlsm

本實作學習如何修改巨集產生的程式碼變成符合本實作的程式碼。本實作要求將工作表中 1~10 個班級所有同學的六科成績，按  鈕就會將 1~10 個班級，逐班顯示所有同學的六科成績。建議先透過功能區「資料」標籤頁的「自動篩選」鈕，點按班級的  下拉鈕，如下圖選擇第一班，便會由成績工作表中篩選出並顯示第一班所有同學的六科成績，將此過程錄製成巨集，再將巨集自動產生的程式碼複製給  按鈕 Command Button1 的 Click 事件程序中，然後在 Click 事件程序中修改相關程式碼，變成可列印 1-10 個班級所有同學六科成績。

► 輸出要求

	A	B	C	D	E	F	G	H	I	J	K
1	班級	座號	姓名	程式設計	計算機概論	會計學	經濟學	記帳實務	專題製作		
2	1	1	姍姍	64	73	89	78	44	80	列印	
3	1	2	怡臻	84	85	80	88	86	98		
4	1	3	蓓庭	84	90	60	92	72	95		
5	1	4	采婷	78	74	49	82	58	93		


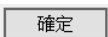

► 解題技巧

Step 1 建立輸出入介面

1. 新增活頁簿並以「Print」為新活頁簿名稱。
2. 將範例 ch05 資料夾中 Print 資料.xlsx 中的資料複製到工作表 1，建立一個 ActiveX 命令按鈕控制項，並設定其屬性值：

	A	B	C	D	E	F	G	H	I	J	K
1	班級	座號	姓名	程式設計	計算機概論	會計學	經濟學	記帳實務	專題製作		
2	1	1	姍姍	64	73	89	78	44	80	列印	
3	1	2	怡臻	84	85	80	88	86	98		
4	1	3	蓓庭	84	90	60	92	72	95		
5	1	4	采婷	78	74	49	82	58	93		
6	1	5	苡倫	90	74	72	90	74	97		
7	1	6	妤凡	74	59	37	82	78	92		
8	1	7	于茹	80	83	49	88	64	97		
9	1	8	巧均	60	38	33	66	50	85		
10	1	9	雅雯	62	84	35	78	50	95		
11	1	10	子庭	88	83	46	88	88	97		
12	2	1	采庭	92	82	79	86	84	100		
13	2	2	映瑄	80	96	68	86	82	98		
14	2	3	心潔	82	75	72	88	82	100		

Step 2 分析問題

1. 自動篩選和列印的程式碼，我們可以使用錄製巨集的方式將程式碼錄下。操作步驟如下：
  - ① 在「開發人員」索引標籤頁中按  鈕，預設會在 Module1 中建立巨集 1 程序。我們使用預設值因為只是要所錄製下來的程式碼，按  鈕開始錄製巨集。
  - ② 點按 A1 儲存格，然後在「資料」索引標籤頁中按  自動篩選鈕，開啟自動篩選的功能。

- ③ 點按班級的 ▾ 下拉鈕，只勾選「1」項目，然後按  鈕，此時只會顯示 1 班同學的成績資料。

	A	B	C	D	E	F	G	H	I
1	班級	座號	姓名	程式設計	計算機概論	會計學	經濟學	記帳實務	專題製作
	從最小到最大排序(S)				73	89	78	44	80
	從最大到最小	1			85	80	88	86	98
	依色彩排序(O)				90	60	92	72	95
	清除"班級"的篩選(O)				74	49	82	58	93
	依色彩篩選(O)				74	72	90	74	97
	數字篩選(O)	2			59	37	82	78	92
	搜尋				83	49	88	64	97
					38	33	66	50	85
					84	35	78	50	95
					83	46	88	88	97
					82	79	86	84	100
					96	68	86	82	98
					75	72	88	82	100
					98	90	98	92	100
					86	79	86	66	95
					86	69	92	96	100
					41	83	70	46	70
					78	62	78	84	97

- ④ 執行【檔案/列印】指令，印出 1 班同學的成績表。
- ⑤ 在「資料」索引標籤中按  鈕，關閉自動篩選功能。
- ⑥ 在「開發人員」索引標籤中按  鈕，停止錄製巨集。

2. 將巨集 1 中錄製的程式碼複製到 CommandButton1\_Click 事件程序中，下面就是所錄製的巨集程式。

```

01 Range("A1").Select      '選取 A1 儲存格
02 Selection.AutoFilter    '開啟自動篩選
03 ActiveSheet.Range("$A$1:$I$102").AutoFilter Field:=1, Criteria:="1" '篩選條件為 1 班
04 ActiveWindow.SelectedSheets.PrintOut Copies:=1, Collate:=True, _
    IgnorePrintAreas:=False '列印 1 份
05 Selection.AutoFilter    '關閉自動篩選

```

3. 錄製完成的巨集程式，可以再加上 VBA 的語法來增加其功能。以上巨集是列印一個班的程式，我們可以加上 For..Next 迴圈來列印出十個班的成績表。上面第三行程式 Criteria:="1"，是設定篩選條件為"1"，所以必須用 Str 函數將計數變數轉為字串型態。另外 Range("\$A\$1:\$I\$102")指定篩選的儲存格範圍，下次執行時如果學生人數增加會造成錯誤，所以修改為 Range("A:I")。

4. 在專案視窗中的 Module1 上按右鍵，執行【移除 Module1】指令刪除錄製的巨集程式。
5. 命令按鈕的 PrintObject 屬性值設為 False，是指定列印時不印出控制項。

### Step 3 編寫程式碼

```
FileName: Print.xlsm (工作表 1 程式碼)
01 Private Sub CommandButton1_Click()
02     Range("A1").Select      '選取 A1 儲存格
03     Selection.AutoFilter    '開啟自動篩選
04     For i = 1 To 10
05         ActiveSheet.Range("A:I").AutoFilter Field:=1, Criteria1:=Str(i) '篩選條件為 i 字串
06         ActiveWindow.SelectedSheets.PrintOut Copies:=1, Collate:=True, _
07             IgnorePrintAreas:=False      '列印 1 份
08     Next
09     Selection.AutoFilter    '關閉自動篩選
10 End Sub
```

## 5.4 Do ... Loop 迴圈

For...Next 計數迴圈會將迴圈內的「敘述區段」，反覆執行指定的次數。因為 Do ... Loop 迴圈沒有計數變數，是靠條件式來決定是否離開迴圈。所以迴圈的執行次數無法事先預知時，就必須改用 Do...Loop 迴圈敘述來設計。

如果 Do...Loop 條件式迴圈敘述的條件式，置於迴圈的第一行就稱為「前測式迴圈」；如果將條件式放在迴圈的最後一行，就稱為「後測式迴圈」。「前測式迴圈」要先判斷條件式是否成立？如果成立才執行迴圈內的敘述。如果條件式一開始便不成立，則迴圈內的敘述不會被執行。「後測式迴圈」是先執行迴圈內的敘述後才判斷條件式，所以迴圈內的敘述至少會被執行一次。

無論是使用前測式或是後測式條件式迴圈，在迴圈內必須有能夠使條件式不滿足的敘述，如此才能離開迴圈，繼續執行接在迴圈後面的敘述。否則會一直在迴圈內反覆執行形成無窮迴圈，而無法離開迴圈的錯誤。

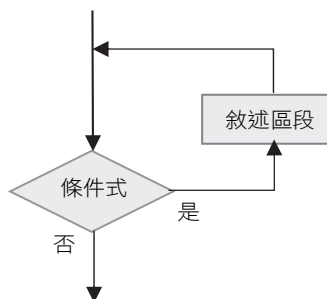
### 5.4.1 Do While 迴圈

Do While 迴圈是<條件式>成立才執行迴圈內的敘述，根據置放<條件式>的位置分成「前測式迴圈」和「後測式迴圈」。

#### 一、前測式迴圈

Do While <條件式> ...Loop 迴圈，將 <條件式> 放置在迴圈的第一行，所以是屬於「前測式迴圈」。當 While 後面的 <條件式> 為 True 時，會如下圖將迴圈內的敘述區段執行一次，然後再回到迴圈的起點，再重新判斷 <條件式> 一次，一直到 <條件式> 為 False 時，才結束迴圈。若要中途離開 Do 迴圈，繼續執行接在迴圈後面的程式碼時，可以使用 Exit Do 敘述。其語法如下：

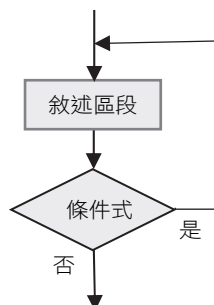
語法：
Do While <條件式>
敘述區段
[Exit Do]
Loop



#### 二、後測式迴圈

至於 Do ... Loop While <條件式> 迴圈，將 <條件式> 放置在迴圈的最後一行，所以是屬於「後測式迴圈」。使用時機是先執行迴圈內的敘述區段一次，才判斷 <條件式> 是否成立時使用。若 <條件式> 為 True，會再執行迴圈內的敘述區段一次，直到 <條件式> 為 False 才結束迴圈。其語法如下：

語法：
Do
敘述區段
[Exit Do]
Loop While <條件式>





Do While...Loop 是前測式迴圈，如果一開始條件就不成立，則迴圈內的敘述區段完全不會被執行就離開迴圈。至於 Do ... Loop While 則是屬於「後測式迴圈」，迴圈內敘述區段至少會執行一次。

### 5.4.2 Do Until 迴圈

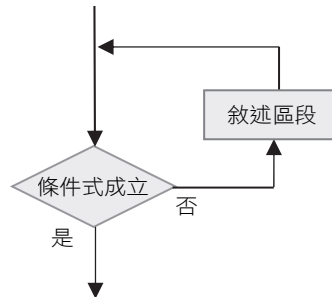
Do While 迴圈是 <條件式> 成立時，才執行迴圈內的敘述區段；而 Do Until 迴圈是 <條件式> 不成立時，才執行迴圈內的敘述區段。根據<條件式>的位置，一樣分成「前測式迴圈」和「後測式迴圈」。

#### 一、前測式迴圈

Do Until...Loop 是以否定方式來做判斷，當<條件式>不成立時，才執行迴圈內的敘述區段；若<條件式>成立，才離開迴圈。Do Until...Loop 迴圈是屬於「前測式迴圈」，其語法為：

```

語法：
Do Until <條件式>
    敘述區段
[Exit Do]
Loop
    
```

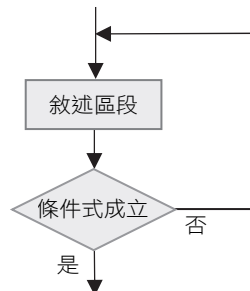


#### 二、後測式迴圈

Do...Loop Until 迴圈也是以否定方式來做判斷，當<條件式>不成立時，才繼續執行迴圈內的敘述區段；若<條件式>成立就離開迴圈。Do...Loop Until 迴圈是屬於「後測式迴圈」，其語法為：

```

語法
Do
    敘述區段
[Exit Do]
Loop Until <條件式>
    
```



Do...Loop 條件式迴圈無論是前測式或後測式迴圈，都可以達到重複執行的效果。撰寫程式時該使用哪個迴圈，應該根據條件和個人的習慣來決定。下面我們用 Do ... Loop 四種不同的條件式迴圈，來撰寫計算  $sum = 1 + 2 + 3 \dots + 10$  程式：

### 一、使用 Do While 迴圈

1.使用 Do While ..... Loop	2. 使用 Do ..... Loop While
<pre>Dim i As Integer, sum As Integer i = 0: sum = 0 Do While i &lt; 10     i = i + 1     sum = sum + i Loop</pre>	<pre>Dim i As Integer, sum As Integer i = 0: sum = 0 Do     i = i + 1     sum = sum + i Loop While i &lt; 10</pre>

### 二、使用 Do Until 迴圈

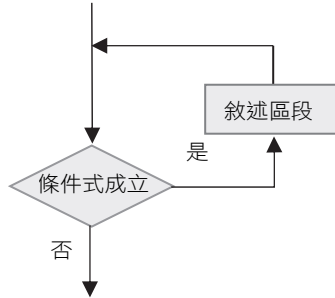
1. 使用 Do Until ..... Loop	2. 使用 Do ..... Loop Until
<pre>Dim i As Integer, sum As Integer i = 0: sum = 0 Do Until i &gt;= 10     i = i + 1     sum = sum + i Loop</pre>	<pre>Dim i As Integer, sum As Integer i = 0: sum = 0 Do     i = i + 1     sum = sum + i Loop Until i &gt;= 10</pre>

#### 5.4.3 While...Wend 迴圈

While...Wend 迴圈和 Do While...Loop 迴圈類似，都是屬於「前測式迴圈」。當 While 後面的條件式結果為 True 時，會將迴圈內的敘述區段執行一次，然後再回到迴圈的起點，再重新判斷條件式一次，當條件式結果為 False 時，才離開迴圈。要特別注意的是 While...Wend 迴圈沒有中途離開迴圈的敘述，所以如果需要中途跳離迴圈，請改用其他的迴圈敘述。其語法如下：

```

語法
While <條件式>
    敘述區段
Wend
    
```



► 簡例

用 While...Wend 迴圈來計算  $sum = 1 + 2 + 3 \dots + 10$  :

```

Dim i As Integer, sum As Integer
i = 0: sum = 0
While i < 10
    i = i + 1
    sum = sum + i
Wend
    
```

**實作** FileName : Sum.xlsm

從 A2 儲存格開始逐一向下讀取數值，直到沒有數值為止，計算出的總計在 B2 儲存格中顯示。

► 輸出要求

	A	B	C
1	數值	總計	計算
2	↓	45	258
3		36	
4		82	
5		71	
6		24	
7			

► 解題技巧

Step 1 建立輸出入介面

1. 新增活頁簿並以「Sum」為新活頁簿名稱。

2. 在工作表 1 中建立如下的表格，和一個 ActiveX 命令按鈕控制項。

	A	B	C
1	數值	總計	計算
2			

### Step 2 問題分析

1. 因為不知道使用者會輸入多少的數值，所以要使用前測式 Do While... Loop 迴圈，當儲存格內有數值才加入總計中，否則就結束迴圈。
2. 使用 IsEmpty() 函數來檢查儲存是否為空白，如果傳回值為 True 表為空白；傳回值為 False 表有資料。
3. 因為 Do ... Loop 迴圈不像 For 迴圈會自動增值，所以要在迴圈內寫程式將變數加一，來讀取下一個儲存格的數值。

### Step 3 編寫程式碼

FileName: Sum.xlsm (工作表 1 程式碼)

01 Sub test()

02 Dim i, sum As Integer

03 i = 2: sum = 0

04 Do While IsEmpty(Cells(i, 1)) = False

05 sum = sum + Cells(i, 1).Value

06 i = i + 1

07 Loop

08 Range("B2").Value = sum

09 End Sub

### ► 隨堂測驗

將上面實作改成向右讀取儲存格數值，並改用 Do Until... Loop 前測式迴圈來編寫程式。

	A	B	C
1	數值	45	36
2	總計	81	計算

	A	B	C	D	E	F	G
1	數值	45	36	82	71	24	
2	總計	258	計算				

## ► 輸出要求



## ► 解題技巧

## Step 1 建立輸出入介面

1. 新增活頁簿並以「CheckScore」為新活頁簿名稱。
2. 在工作表 1 中建立如下的表格，和一個 ActiveX 命令按鈕控制項。

	A	B	C	D	E	F	G
1	座號	國文	英語	數學	物理	歷史	檢查
2	1	99		77	66	55	
3	2		70	80	90	100	
4	3	97	96	95		93	

## Step 2 問題分析

1. 因為是矩形的儲存格範圍，所以要使用巢狀 Do ... Loop 迴圈來逐格檢查。
2. 因為 Do ... Loop 為無窮迴圈，所以一定要有 Exit Do 敘述，才能跳離迴圈。當到第 5 列就離開外層迴圈，到第 7 欄就離開內層迴圈。
3. 使用 IsEmpty() 函數檢查是否有輸入資料，當傳回值為 True 表沒有資料；False 則表有資料。
4. 使用 Cells() 的 Address 屬性，可以取得該儲存格的位址，例如 B2 儲存格位址為「\$B\$2」。