

CH8 物件簡介與Application物件

物件簡介與Application物件

8.1 物件簡介

8.2 Application物件常用的屬性

8.3 Application物件常用的方法

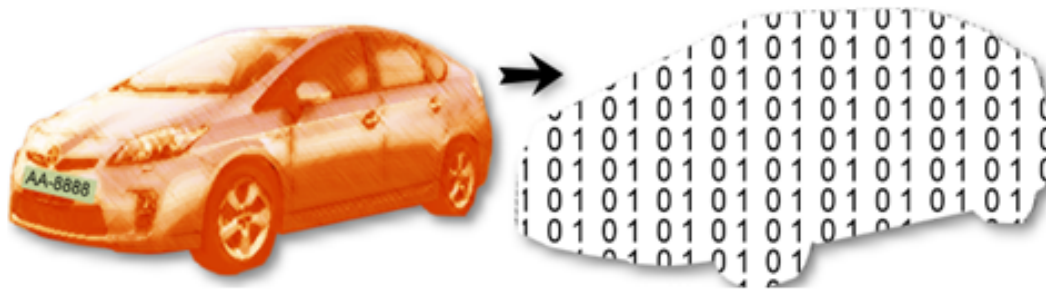
8.4 Application物件常用的事件

8.1 物件簡介

8.1.1 物件是什麼？

- 物件是外界真實物品的**抽象對應**，例如：轎車、貨車、休旅車、警車 ... 等都對應「車子」物件。
- 物件應該包含**狀態**或稱**屬性**(汽車的排氣量、車種、顏色...)，也包含**行為**(發動、加速、剎車...)。
- 物件具有**可識別**的特性(例如汽車的車牌號碼)。
- **特性類似的物件**可以歸類成同一個**類別(Class)**，例如車牌號碼為AA-8888的轎車和BB-9999貨車同屬於「車子」類別，兩輛車雖然都屬於「車子」類別，但是各為不同的物件。

真實物品抽象化成物件



汽車物件

車牌號碼：AA-8888

車 種：轎車

排 氣 量：1800

顏 色：紅

發動、加速、剎車、
換檔...等行為

- 在物件導向程式中，物件是由類別所建立的實體 (Instance)。
- 物件導向程式設計(Object Priented Programming, OOP)就是模擬真實世界所發展出來的概念，適合用來發展大型的程式。
- Excel VBA是符合物件導向程式的設計理念。

8.1.2 物件的屬性

- 物件抽象化過程中會將真實物件的各種特徵，改以各種型態的資料來描述，這些資料就稱為物件的屬性(Property)。
- 譬如各種轎車都是屬於汽車這個類別的物件，但是因同類別的各物件具有相同屬性其屬性值卻不同，如：廠牌、排氣量、顏色...等，所以能夠區分出不同的物件。

8.1.2 物件的屬性

Continue...

- 在設計程式時，每個物件都具有**Name**(物件名稱)**屬性**且必須是**唯一**，才能在程式中識別。
- 例如：汽車的**車牌號碼**就像是物件**Name**屬性，**車牌號碼**必須是**唯一**，根據車牌號碼來寄送罰單就不會出錯。
- 譬如：物件名稱為「myCar」，若要將該物件的**Color**屬性設為「紅色」，其寫法如下：
myCar.Color = "紅色"

8.1.3 物件的方法

- 在物件抽象化的過程，會將真實物件的各種**行為(功能)**，改以各種**方法(Method)**來達成。
- 例如：汽車具有發動、加速、剎車...等**行為**，在汽車物件中就有對應的**方法**來完成該行為(功能)。
- 在**呼叫方法**執行時，可**傳遞訊息**來指定動作，例如：物件名稱為myCar，提供ChangeSpeed(number)方法來換檔，若要將myCar物件車速切換到2檔，其寫法如下：
myCar.ChangeSpeed(2)

8.1.4 物件的事件

- **事件(Event)**其實也是物件的一種**方法**，只是**事件**是由物件本身、系統或其他物件來**觸動執行**。
- VBA針對各種物件**定義一系列事件**，每個物件擁有的事件不盡相同，可以針對被觸發事件要處理的程式碼寫在該事件程序中。
- 例如：在按鈕物件上按一下滑鼠左鍵時，會觸動按鈕的**Click事件**，可將使用者按鈕按下時要處理的程式碼寫在Click事件程序中。

8.1.5 物件導向程式的特性

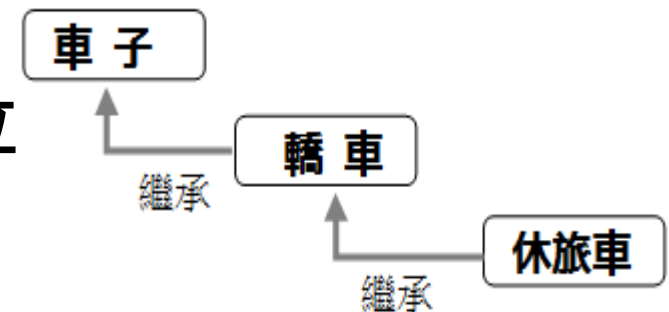
- 在物件導向程式中物件具有**封裝**、**繼承**和**多型**的三個基本特性，可改進程序導向程式缺乏**程式資料安全**、**程式再利用**、**維護困難**的問題。
- 改進**程式資料安全**：**封裝 (Encapsulation)**
- 改進**程式再利用**：**繼承 (Inheritance)**
- 改進**維護困難**：**多型 (Polymorphism)**

8.1.5 物件導向程式的特性--封裝 (Encapsulation)

- 所謂「封裝」就是將物件中的**資料和方法**加以**保護**，外界必須**透過介面**或執行物件中的**方法**，才能存取物件中的資料。
- 外界**不用知道**物件內如何運作，**透過介面**就能順利運用物件，能避免物件的資料被不當存取。
- 例如：我們可以讀取物件的屬性、設定屬性，甚至執行物件的方法。

8.1.5 物件導向程式的特性--繼承 (Inheritance)

- 所謂「繼承」就是指物件可以利用現有的類別(父類別)來建立新的類別(子類別)。
- 子類別可繼承父類別的屬性和方法，也可新增自己的屬性和方法，或覆寫父類別屬性和方法。
- 物件具有此特性所以程式碼可以重複使用，加快了程式開發的速度。
- 類別間會因為繼承形成階層式的關係。例如：用「車子」類別建立「轎車」類別，再用「轎車」類別建立「休旅車」類別，關係如下：



8.1.5 物件導向程式的特性--多型 (Polymorphism)

「多型」就是透過物件的單一介面，可以使用多元的存取方式來操作物件的屬性和方法。物件具有多型的特性，使物件操作富有彈性。例如：依照引數的個數或資料型別的不同來呼叫下列myCar物件的ChangeSpeed()三種方法：

myCar.ChangeSpeed(2)

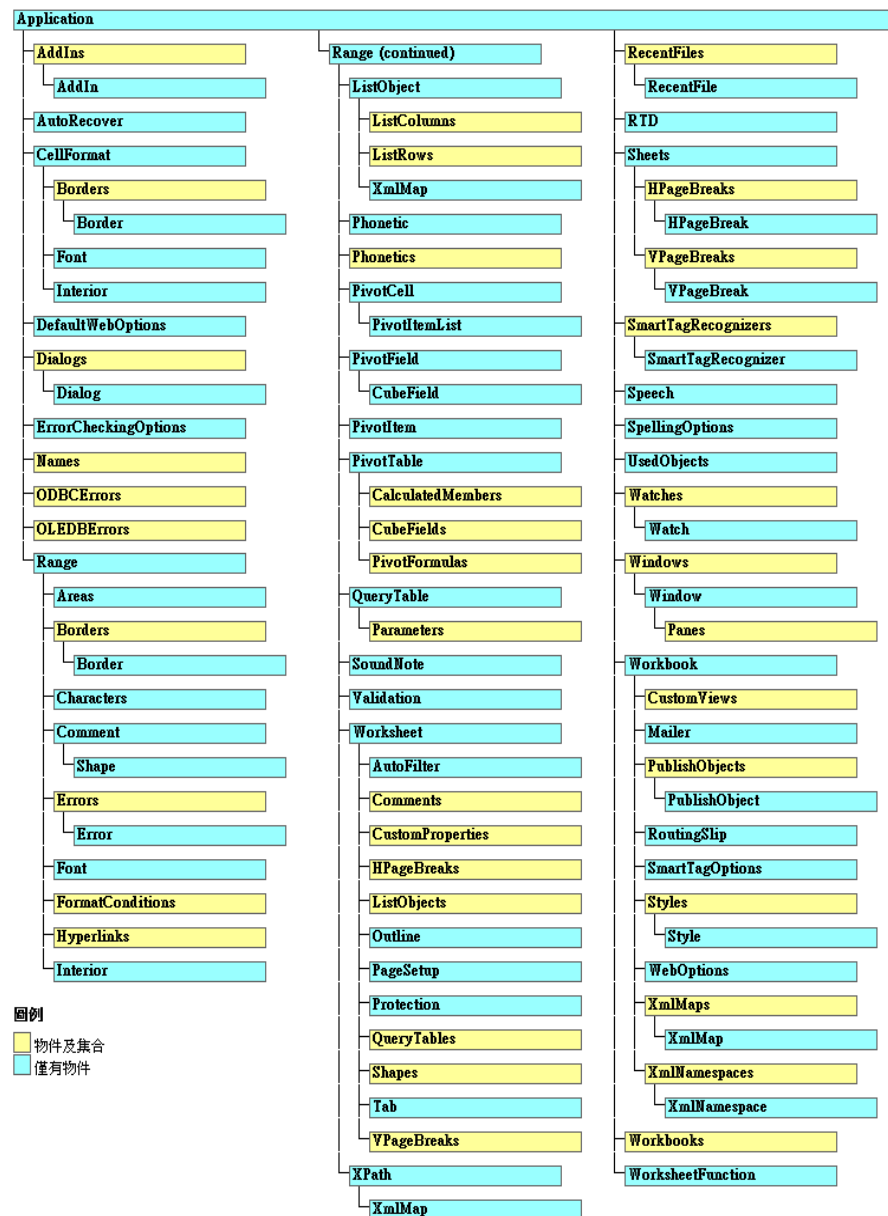
myCar.ChangeSpeed (2, Auto)

myCar.ChangeSpeed (2, Turbo)

8.1.6 Excel的物件模型

- 在Excel中，物件如右圖，在「物件模型」階層結構中，被有系統地架構起來，因為Excel物件眾多僅列出常用的物件。
- Excel最常用的物件有：**Application**、**Workbook**、**Sheet**、**Range...**等。

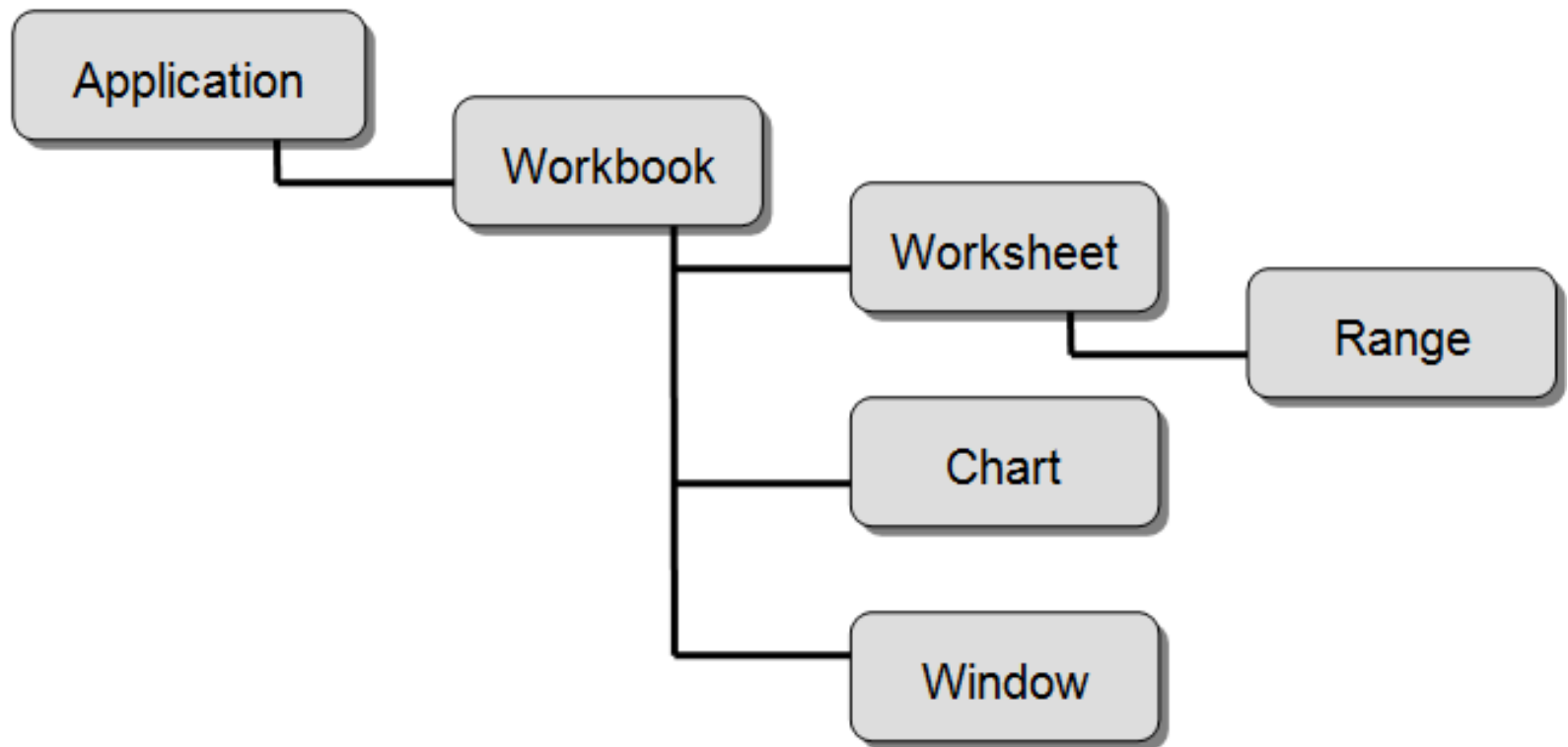
Microsoft Excel 物件模型



8.2 Application 物件常用的屬性

- 當Excel應用程式被執行時，在VBA中就會建立一個Application物件，來代表整個Excel應用程式，所以Application物件等於Excel應用程式。
- 當設定Application物件的屬性值或執行方法時，會影響到整個Excel應用程式。
- Application物件就像是個大容器，其中可以包含多個活頁簿(WorkBook)、工作表(Worksheet)...等物件。

- Excel VBA中的物件間有階層式關係，如下圖即為常用物件的隸屬關係：



8.2.1 設定Excel環境的常用屬性 Continue...

若要指定物件的屬性值，其語法為：

語法：

物件名稱.屬性名稱 = 屬性值

1. Caption屬性

Caption屬性設定Excel應用程式的標題欄文字內容，例如：設定標題欄文字內容為「薪資系統」，寫法為：

`Application.Caption = "薪資系統"`

8.2.1 設定Excel環境的常用屬性 Continue...

2. DisplayStatusBar

DisplayStatusBar屬性可以設定Excel應用程式是否顯示狀態列，屬性值有True(顯示)、False(不顯示)兩種。

例如：設定要顯示狀態列，寫法為：

```
Application.DisplayStatusBar = True
```

3. StatusBar

StatusBar屬性可以設定Excel應用程式的狀態列文字內容，若要恢復狀態列的預設文字狀態，則可以將屬性值設為False。

```
Application.StatusBar = "讀檔中"
```

8.2.1 設定Excel環境的常用屬性 Continue...

4. WindowState

WindowState屬性可以設定Excel應用程式的視窗狀態。屬性值有xlMaximized(視窗最大化)、xlMinimized(視窗最小化)、xlNormal (手動)三種。

例如：將Excel應用程式縮小到工作列，寫法為：

```
Application.WindowState = xlMinimized
```

5. DisplayFullScreen

DisplayFullScreen屬性可以設定為全螢幕模式，將Excel應用程式視窗最大化。屬性值為True(全螢幕模式)、False(一般模式，預設值)。

```
Application.DisplayFullScreen = True
```

8.2.1 設定Excel環境的常用屬性 Continue...

6. DisplayAlerts

DisplayAlerts屬性可以設定Excel應用程式在關閉視窗等操作時，**是否顯示提示訊息或警告**。

例如：提醒尚未儲存檔案、是否確定刪除工作表。

屬性值有True(顯示提示訊息或警告)、False(不顯示提示訊息或警告)兩種。

例如：將Excel應用程式的提示訊息和警告功能關閉。
寫法為：

```
Application.DisplayAlerts = False
```

8.2.1 設定Excel環境的常用屬性 Continue...

7. EnableEvents

EnableEvents屬性可以設定Excel應用程式**是否允許觸動事件**，屬性值有True(**預設值**、啟用事件觸發)、False(暫停事件觸發)兩種。

例如：修改儲存格內容會觸動工作表的Change事件，如果要連續修改資料又不要一直觸動事件時，就可以先設EnableEvents屬性值為False，等修改完資料後，再設回屬性值為True。

```
Application.EnableEvents = False
```

8.2.1 設定Excel環境的常用屬性 Continue...

8. FileDialog

FileDialog屬性可以顯示系統內建和檔案相關的對話方塊，可用的參數值有msoFileDialogOpen(開啟舊檔)、msoFileDialogSaveAs(儲存檔案)、msoFileDialogFolderPicker(瀏覽)等三種。

後面必須使用Show()方法來顯示對話方塊。

例如：開啟「開啟舊檔」對話方塊，寫法為：

```
Dim dialog As Object      ' 宣告 dialog 物件
' 指定 dialog 等於開啟舊檔對話方塊
Set dialog = Application.FileDialog(msoFileDialogOpen)
Dim pick As Boolean       ' 宣告 pick 布林變數
pick = dialog.Show        ' 顯示開啟舊檔對話方塊
If pick Then              ' 如果按確定鈕
    MsgBox "選取的檔案是： " & dialog.SelectedItems(1)
End If
```

8.2.1 設定Excel環境的常用屬性

9. Dialogs

Dialogs屬性可以開啟系統內建的對話方塊，可用的屬性值很多可以由清單中選用，然後使用Show方法來開啟對話方塊。

例如：開啟列印對話方塊，寫法為：

```
Application.Dialogs(xlDialogPrint).Show
```

10. Interactive

Interactive屬性可中止接受使用者在工作表的操作動作，屬性值False(接受，預設值)、True(不接受)。

在執行程序或巨集前先將Interactive屬性值設為False，待執行後再重設屬性值為True。



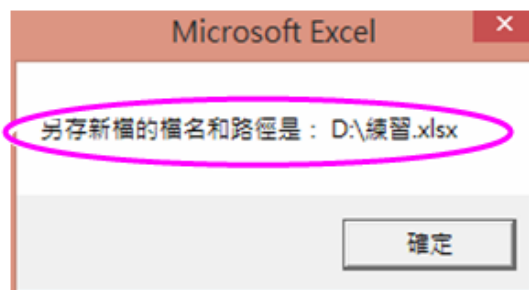
實作 FileName : Application 屬性 1.xlsm

按 <設定>鈕會設定 Excel 狀態為：標題欄文字為 B1 儲存格內容、狀態列顯示 B2 儲存格內容、顯示「儲存檔案」對話方塊並顯示另存的檔名、Excel 視窗大小為手動。按 <復原>鈕會設定 Excel 視窗最大化，並恢復狀態列的預設文字。

► 輸出要求

Application屬性 - Application實作				
	A	B	C	D
1	標題欄文字：	Application實作	設定	
2	狀態列文字：	實作中...	復原	

實作中...



► 解題技巧

- 要顯示「儲存檔案」對話方塊，可以使用 **Application** 的 **FileDialog** 屬性，參數值設為 **msoFileDialogSaveAs**。當輸入左小括弧時就會顯示清單，從中選取 **msoFileDialogSaveAs** 項目即可。

```
Set dialog = Application.FileDialog(  
    FileDialog(FileDialogType As MsoFileDialogType) As FileDialog  
        ☐ msoFileDialogFolderPicker  
        ☐ msoFileDialogOpen  
        ☐ msoFileDialogSaveAs
```


Step 3 編寫程式碼

FileName: Application 屬性 1.xlsm (工作表 1 程式碼)

```
01 Private Sub CommandButton1_Click()  
02     Application.Caption = Range("B1").Value      '設定標題欄文字內容  
03     Application.DisplayStatusBar = True           '顯示狀態列  
04     Application.StatusBar = Range("B2").Value     '設定狀態列顯示文字內容  
05     Dim dialog As Object      '宣告 dialog 物件  
06     '指定 dialog 等於儲存檔案對話方塊  
07     Set dialog = Application.FileDialog(msoFileDialogSaveAs)  
08     Dim pick As Boolean       '宣告 pick 布林變數  
09     pick = dialog.Show        '顯示儲存檔案對話方塊  
10     If pick Then             '如果按確定鈕  
11         MsgBox "另存新檔的檔名和路徑是： " & dialog.SelectedItems(1)  
12     End If  
13     Application.WindowState = xlNormal           '設定視窗大小為手動  
14 End Sub  
15  
16 Private Sub CommandButton2_Click()  
17     Application.WindowState = xlMaximized        '設定視窗最大化  
18     Application.StatusBar = False                '恢復狀態列預設文字狀態  
19 End Sub
```

► 隨堂測驗

將上面實作的開啟「儲存檔案」對話方塊，改為「**開啟舊檔**」對話方塊並顯示選取的檔案的檔名，然後顯示「**列印**」對話方塊。

8.2.2 指定物件的常用屬性

Continue...

1. Workbooks

Workbooks屬性可以取得Excel應用程式中的活頁簿集合。

例如：新增一個活頁簿然後顯示活頁簿的總數：

```
Application.Workbooks.Add
```

```
MsgBox “活頁簿數量：” & _
```

```
Application.Workbooks.Count
```

8.2.2 指定物件的常用屬性

Continue...

2. **ActiveWorkbook**

ActiveWorkbook屬性，可以取得Excel應用程式**目前作用中活頁簿**。

例如：顯示目前作用中活頁簿的Name屬性：

```
MsgBox Application.ActiveWorkbook.Name
```

8.2.2 指定物件的常用屬性

Continue...

3. Worksheets

使用Worksheets屬性可以取得目前作用活頁簿中，所有工作表的集合。

例：在第1個工作表後面新增一個工作表，然後設該工作表的名稱為「新增工作表」，寫法為：

```
Worksheets.Add After := Worksheets(1)  
Worksheets(2).Name = "新工作表"
```

8.2.2 指定物件的常用屬性

Continue...

4. Sheets

Sheets屬性可以取得目前作用活頁簿中，所有**工作表**和**圖表工作表**的集合。

指定工作表可用索引值，例如：Sheets(1)表示第一個工作表。

也可用工作表名稱，例如：Sheets("工作表1")。

8.2.2 指定物件的常用屬性

Continue...

5. **ActiveSheet**

ActiveSheet屬性可以取得**目前作用中工作表**，如果沒有作用中工作表時傳回值為Nothing。

MsgBox ActiveWorkbook.ActiveSheet.Name

8.2.2 指定物件的常用屬性

Continue...

6. Columns

Columns屬性可以取得目前作用工作表中，所有垂直欄的集合。

例如：選取F欄(索引值為6)的寫法為：

```
Application.Columns(6).Select  
Application.Columns( "F" ).Select
```

選取1~3欄的寫法為：

```
Application.Columns("A:C").Select
```


8.2.2 指定物件的常用屬性

Continue...

7. Rows

Rows屬性可以取得目前作用工作表中，所有**水平列的集合**。

例如刪除第6列，選取3~6列的寫法為：

```
Application.Rows(6).Delete
```

```
Application.Rows("3:6").Select
```

8.2.2 指定物件的常用屬性

Continue...

8. Range

Range屬性可以取得目前作用工作表中的一個儲存格，或是儲存格範圍。

例如：清除指定A2 ~ D7儲存格的值，寫法為：

```
Application.Worksheets( "工作表1" )  
    .Range("A2:D7").ClearContents
```

8.2.2 指定物件的常用屬性

Continue...

9. Cells

使用Cells屬性可以取得目前作用工作表中的**所有儲存格的集合**。

例如：指定C6儲存格的值，寫法為：

```
Application.Cells(6, 3).Value = "薪資 "
```

10. ActiveCell

使用ActiveCell屬性可以取得**目前作用中儲存格**，其資料型別為Range。

例如：將目前作用中儲存格的字型指定為粗體字：

```
ActiveCell.Font.Bold = True
```

8.2.2 指定物件的常用屬性

11. Selection

使用Selection屬性可取得目前選取的物件，選取的物件可能是儲存格範圍、圖表...等等。

例如：若選取儲存格範圍，就將儲存格的背景色指定為紅色，寫法：

```
If TypeName(Selection) = "Range"  
    Selection.Interior.Color = RGB(255, 0, 0)  
End If
```

8.2.3 物件變數的宣告

- 物件也可以用Dim宣告成物件變數，物件型態可以是Workbook、Worksheet、Range...等物件。
- 然後用Set敘述來指定物件變數的參照位置，其語法為：

語法：

```
Dim 物件變數名稱 As 物件型態
```

```
Set 物件變數名稱 = 物件
```

8.2.3 物件變數的宣告

Continue...

- 用Set敘述來指定物件變數的參照來源，並不是真的將物件指定給變數，而是將物件參照(物件的記憶體位置)指定給物件變數。
- 用Dim宣告r為儲存格物件，並用Set指定為A1:B2儲存格範圍，寫法為：

```
Dim r As Range
```

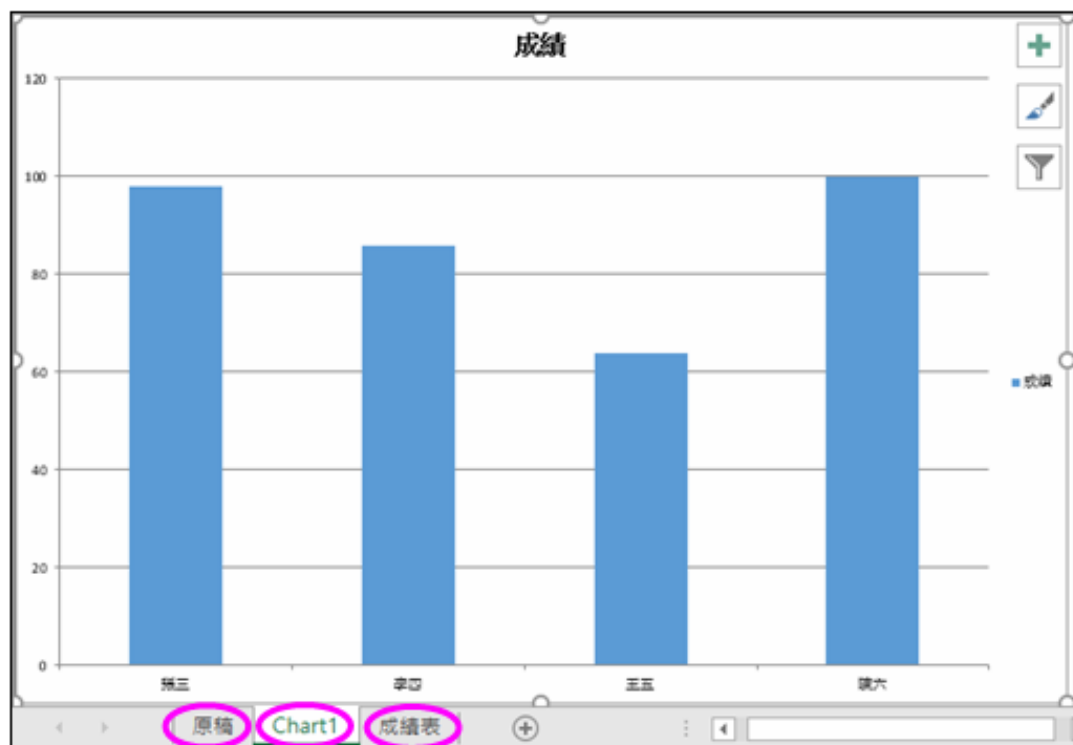
```
Set r = Range("A1:B2")
```



實作 FileName：Application 屬性 2.xlsm

目前作用工作表名為「原稿」，在其後新增一個名為「成績表」的工作表。接著將「原稿」工作表的成績複製到「成績表」工作表中。最後以「成績表」工作表中的成績作為資料來源，新增一個圖表。

► 輸出要求



► 解題技巧

- 用Application的Worksheets.Add方法來新增工作表，並用After參數指定位在目前工作表後面。
- 要指定AB兩欄，可以使用Columns("A:B")來表示。然後用Copy方法來複製兩欄的資料。
- 先用Select方法來選取 "成績表" 工作表的A1儲存格，表示資料要貼上的位置，然後用Paste方法將複製的資料貼上。
- 先用Select方法來選取 "成績表" 工作表的A1:B5儲存格範圍，作為圖表的資料來源。然後用Sheets的Add方法，並將Type參數設為xlSheetType.xlChart來新增一個圖表。

Step 3 編寫程式碼

FileName: Application 屬性 2.xlsm (工作表 1 程式碼)

```
01 Private Sub CommandButton1 Click()
```

02	Worksheets("原稿").Select	'選取"原稿"使成為目前作用工作表
----	-------------------------	-------------------

03	Application.Worksheets.Add After:=ActiveSheet	'新增工作表在目前工作表後面
----	---	----------------

04	Application.ActiveSheet.Name = "成績表"	'設新增工作表的名稱為"新增"
----	--------------------------------------	-----------------

05	Worksheets("原稿").Columns("A:B").Copy	'複製'原稿'工作表的 AB 兩欄
----	--------------------------------------	-------------------

06	Worksheets("成績表").Range("A1").Select	'選取"成績表"工作表的 A1 儲存格
----	--------------------------------------	---------------------

07	Worksheets("成績表").Paste	'將複製的 AB 兩欄貼到"成績表"工作表
----	-------------------------	-----------------------

08	Worksheets("成績表").Range("A1:B5").Select	'選取資料來源儲存格範圍
----	---	--------------

09 Application.Sheets.Add Type:=xlSheetType.xlChart '新增一個 Sheet 其格式為圖表

10 End Sub

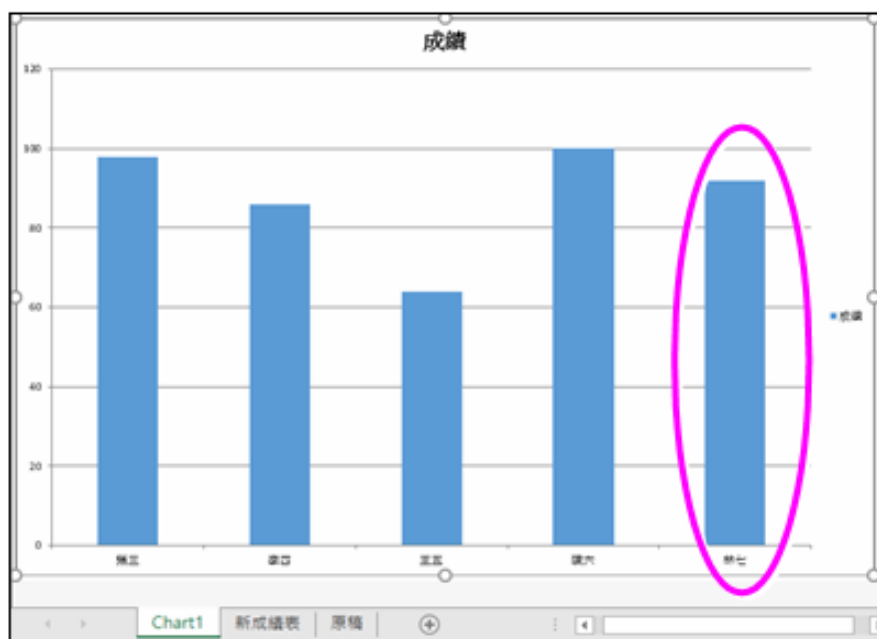
► 隨堂測驗

目前作用工作表名稱為「原稿」，在前面新增一個名為「新成績表」的工作表。接著將「原稿」工作表的成績複製到「新成績表」工作表中。用 Cells 屬性來指定 A6 和 B6 儲存格的值，分別為 "林七" 和 92。設定標題欄的背景色為綠色(RGB(0, 150, 0))，字體為粗體。最後以「新成績表」工作表中的成績作為資料來源，新增一個圖表。

粗體字

	A	B	C
1	姓名	成績	← 加底色
2	張三	98	
3	李四	86	
4	王五	64	
5	陳六	100	
6	林七	92	← 新增資料
7			

Chart1 新成績表 原稿



8.2.4 設定Excel計算的常用屬性 Continue...

1. Calculation

- Calculation屬性可以設定對儲存格內**公式的重算方式**，屬性值有xlCalculationManual(手動重算)、xlCalculationAutomatic(自動重算)、xlCalculation Semiautomatic (除運算列表為手動外其餘為自動重算)三種。
- 例如：將Excel應用程式設為手動重算以提升效率，寫法為：

`Application.Calculation = xlCalculationManual`

8.2.4 設定Excel計算的常用屬性 Continue...

2. CalculateBeforeSave

- 當Calculation屬性值設為xlCalculationManual (手動重算)時，可以使用CalculateBeforeSave屬性指定存檔前是否再重算儲存格內公式。
- 屬性值有：True(存檔前重算)、False(存檔前不重算)兩種。
- 例如：將Excel應用程式設為存檔前必須重算儲存格內公式，寫法為：

`Application.CalculateBeforeSave = True`

8.2.4 設定Excel計算的常用屬性 Continue...

3. ScreenUpdating

- ScreenUpdating屬性可以設定Excel應用程式是否**更新畫面**，屬性值有True(更新畫面)、False(不更新畫面)兩種。
- 停止畫面更新可以**加快程式的執行效率**，以及減少畫面的跳動。
- 例如關閉Excel應用程式更新畫面功能，來加快程式的執行效率，寫法為：

`Application.ScreenUpdating = False`

8.2.4 設定Excel計算的常用屬性 Continue...

4. WorksheetFunction

- WorksheetFunction屬性可以引用Excel工作表的函數。
- 在VBA程式中如果要引用Excel工作表的函數，除非VBA已經內建該函數，否則必須透過WorksheetFunction屬性。
- 例如：計算L2到L5儲存格的平均，寫法為：
`Range("L7").Value = Application.
WorksheetFunction.Average(Range("L2:L5"))`

8.3 Application 物件常用的方法

8.3.1 Application物件常用的方法

Continue...

1. Calculate

當Application的Calculation屬性值設為手動重算時，可以使用Calculate方法來重算活頁簿中的公式。

Application.Calculate

Worksheets("工作表1").Calculate

2. Speech.Speak

Speech.Speak方法可用語音讀出字串參數，例如：要讀出「Hello你好」寫法為：

Application.Speech.Speak("Hello 你好")

8.3.1 Application物件常用的方法

Continue...

3. Goto

Application的Goto方法，可**選取**已開啟活頁簿的**任意工作表中的儲存格**，其語法為：

語法：

```
Application.Goto ( Reference, [ Scroll])
```

Reference引數是指定儲存格範圍。

Scroll引數是設定是否**捲動螢幕**，引數值為False(預設值)不捲動；若為True時則會捲動。

例：選取並捲動到第二個工作表的A50儲存格：

```
Application.Goto
```

```
Reference:=Worksheets(2).Range("A50"), Scroll:=True
```


8.3.1 Application物件常用的方法

Continue...

4. Run

如果要呼叫使用其他Excel檔案中的巨集，則要使用Application的Run方法。其常用語法為：

語法：

```
Application.Run "'檔案名稱'!巨集名稱", 引數串列
```

Excel檔案名稱前後要用 ' 括住，用 ! 接巨集名稱。

引數串列數量可以為0~30個。

執行時會開啟該Excel檔案，再執行指定巨集程序

例如執行ch08測試.xlsm檔中的Test巨集程序，並傳遞一個字串引數，寫法為

```
Application.Run "'ch08測試.xlsm'!Test", "執行巨集程序"
```

8.3.1 Application物件常用的方法

Continue...

5. Wait

Wait方法可以暫停巨集執行，直到指定的時間到時才繼續執行接在後面的敘述。

例如：要暫停到下午3:30，程式寫法為：

```
Application.Wait "15:30:00"
```

8.3.1 Application物件常用的方法

Continue...

6. OnTime

OnTime方法可以暫停指定的時間，然後再執行指定的巨集。

例如暫停6秒鐘後才執行Test巨集程序，寫法為：

```
Application.OnTime Now + TimeValue("00:00:06"), "Test"
```

8.3.1 Application物件常用的方法

Continue...

7. OnKey

- OnKey方法讓使用者按下**指定按鍵**時，可以**執行指定的巨集**，通常寫在WorkBook的**Open事件**中。
- 例如使用者按 <Ctrl> + <Shift> + 鍵時就執行Test巨集程序，寫法為：

```
Application.OnKey "^+{b}", "Test"
```

- 如果要取消指定按鍵執行巨集程序時，通常寫在WorkBook的**BeforeClose事件**中，寫法為：

```
Application.OnKey "^+{b}", ""
```

8.3.1 Application物件常用的方法

Continue...

8. InputBox

Application的InputBox方法可以顯示輸入對話方塊，讓使用者輸入指定資料型態的資料。

其語法為：

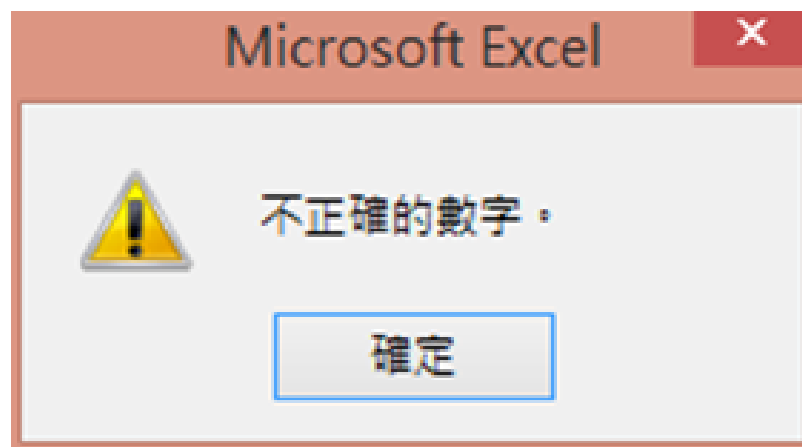
語法：

```
傳回值 = Application.InputBox(Prompt:= "提示" [, Title:= "標題"]  
                                [, Default:= 預設值] [, Type:= 型態值])
```

8.3.1 Application物件常用的方法

Continue...

- **Type**引數可以設定輸入值的型態，Type有多種設定值：0(公式)、1(數字)、2(字串；預設值)、4(True 或 False)、8(儲存格為 Range 物件)、16(錯誤值；例如 #N/A)、64(數值陣列)。使用者如果輸入不是指定資料型態，會顯示錯誤訊息。



8.3.1 Application物件常用的方法

Continue...

如果使用者按 取消 鈕，則傳回值為False。

例如顯示輸入方塊讓使用者輸入數值，如果按確定 鈕就顯示輸入數值，寫法為：

```
num = Application.InputBox("輸入數值：", Type:=1)
```

```
If num <> False Then
```

```
    Range("C13").Value = num
```

```
End If
```

8.3.1 Application物件常用的方法

Continue...

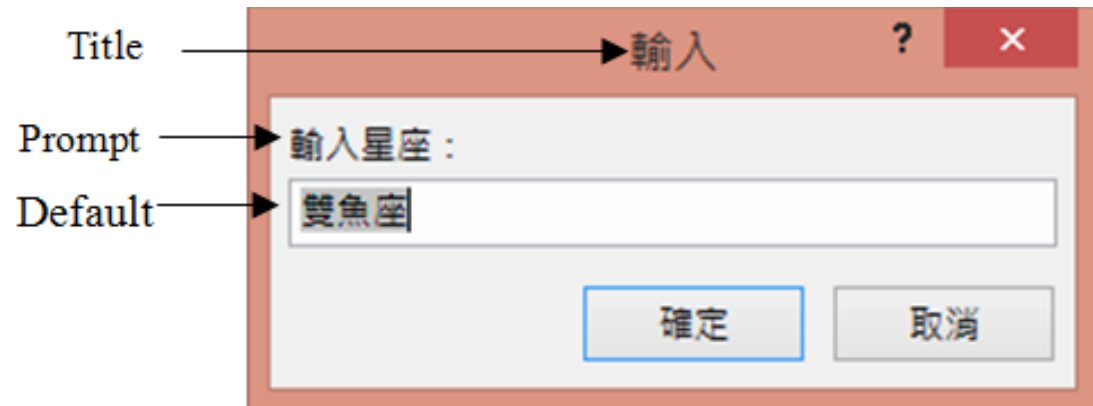
例：顯示輸入方塊讓使用者輸入字串，標題為“輸入”，預設值為“雙魚座”，如果按確定鈕就顯示輸入字串，寫法為：

```
star = Application.InputBox( "輸入星座：" , _  
    Title:="輸入", Default:="雙魚座", Type:=2)
```

```
If star <> False Then
```

```
    Range("E13").Value = star
```

```
End If
```



8.3.1 Application物件常用的方法

Continue...

例如：顯示輸入方塊讓使用者輸入公式，如果按 確定 鈕就顯示公式計算結果，寫法為：

```
myFormula = Application.InputBox("輸入公式：", _  
Default:="=Max($L$2:$L$5)", Type:=0)
```

```
If myFormula <> False Then
```

```
    Range("L8").Value = myFormula
```

```
End If
```

8.3.1 Application物件常用的方法

Continue...

Type引數設定值可以指定多種輸入值的型態，

例如使用者可以輸入數值、字串和布林值時，可以指定Type設定值為7(1+2+4)。

例如：顯示輸入方塊讓使用者輸入數值或字串，如果按 鈕就顯示輸入的資料，程式寫法為：

```
num = Application.InputBox( "輸入數值或字串：" , _  
    Type:=3)
```

```
If num <> False Then
```

```
    Range("H13").Value = num
```

```
End If
```

8.3.1 Application物件常用的方法

Continue...

10. Union :

Application的Union方法，可取得多個儲存格範圍的聯集範圍。其語法為：

語法：

```
儲存格範圍 = Application.Union ( Arg1, Arg2, [Arg3, ..., Arg30])
```

Arg1~Arg30引數為儲存格範圍，最少要有2個儲存格範圍最多為30個，傳回值所有儲存格範圍的聯集。

例：選取A1:C5和B3:F4的儲存格範圍，寫法為：

```
Dim rng As Range
```

```
Set rng = Application.Union(Range("A1:C5"), Range("B3:F4"))
```

```
rng.Select    '同時A1:C5和B3:F4儲存格範圍
```

8.3.1 Application物件常用的方法

Continue...

11. GetOpenFilename

- Application的GetOpenFilename方法會開啟「開啟舊檔」對話方塊，並傳回使用者選取的檔案名稱，但不會開啟該檔案。其語法為：

語法：

```
檔案 = Application.GetOpenFilename ([FileFilter:= "篩選準則 串列"]  
                                     [, FilterIndex:= 篩選條件索引值] [, MultiSelect:= 複選])
```

- FileFilter引數是由提示字串和萬用字元檔案篩選準則為一組的字串所組成，其中是由逗號分隔。
- 只列出副檔名為xlsx的活頁簿檔案，FileFilter引數為 "活頁簿檔案, *.xlsx"。

8.3.1 Application物件常用的方法

Continue...

- 例如列出副檔名為xlsx、xlsm、csv的檔案，FileFilter引數為 "活頁簿檔案, *.xlsx, 啟用巨集活頁簿檔案, *.xlsm, CSV格式, *.cvs" 。
- 如果省略FileFilter引數則預設為列出所有檔案(*.*)。
- 例如讓使用者選取一個活頁簿檔案(.xlsx)並開啟，寫法為：

```
Dim fName As Variant
```

```
fName = Application.GetOpenFilename(  
    "活頁簿檔案, *.xlsx ")
```

```
If fName <> False Then Workbooks.Open(fName)
```

8.3.1 Application物件常用的方法

Continue...

- 如果有多個篩選準則，可以設定 **FilterIndex** 引數來指定預設的篩選準則，省略時預設為第一個。
- 例如預設為第二個篩選準則，寫法為：

Dim fName As Variant

fName = Application.GetOpenFilename("活頁簿檔案, *.xlsx, 啟用巨集活頁簿檔案, *.xlsm, CSV格式, *.cvs ", FilterIndex:=2)

語法：

```
檔案 = Application.GetOpenFilename ([FileFilter:= "篩選準則 串列"]  
                                     [, FilterIndex:= 篩選條件索引值] [, MultiSelect:= 複選])
```

8.3.1 Application物件常用的方法

Continue...

- **MultiSelect**引數可以設定是否允許複選檔案，引數值為False(預設值)時只能選取一個檔案名稱；若為True時則可以多選取的檔案名稱。
- 例如使用者選取多個活頁簿檔案(.xlsx)並全部開啟：

```
Dim fName As Variant
fName = Application.GetOpenFilename(
    "活頁簿檔案, *.xlsx ", MultiSelect:=True)
If fName <> False Then
    For i = 1 To UBound(fName)
        Workbooks.Open(fName(i))
    Next i
End If
```

8.3.1 Application物件常用的方法

12. Quit

Application的Quit方法可以關閉Excel應用程式。例如將所有開啟的活頁簿存檔後關閉Excel應用程式，寫法為：

```
For Each wb In Application.Workbooks  
    wb.Save  
Next wb  
Application.Quit
```


8.3.2 使用With...End With敘述

Continue...

- 如果物件有多個屬性或方法要同時設定時，可以使用With ... End With敘述結構。
- 使用With敘述不但可以容易閱讀，更重要的是能夠提高程式的執行效率，其語法為：

語法：

```
With 物件名稱  
    [ .屬性 = 屬性值 ]  
    ...  
    [ .方法 ]  
    ...  
End With
```

8.3.2 使用 With...End With 敘述

Continue...

- 例如：設定 Excel 沒有警告訊息就直接結束，寫法為：

```
Application.DisplayAlerts = False  
Application.Quit
```

- 使用 With ... End With 敘述

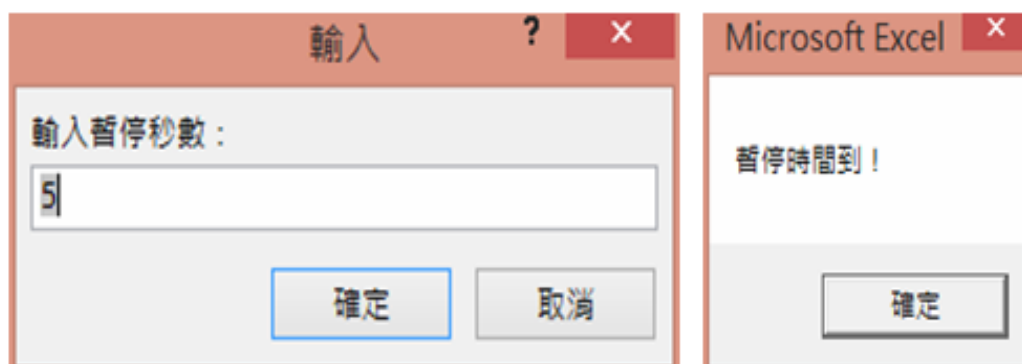
```
With Application           '使用 With 敘述  
    .DisplayAlerts = False  
    .Quit  
End With
```



實作 FileName：Application 方法 .xlsm

程式執行時會出現對話方塊，可以輸入暫停秒數字串(預設值為 5)，秒數到會執行「暫停」巨集程序。「暫停」程序執行時，會讀出「暫停時間到」語音，並顯示「暫停時間到！」訊息方塊。

► 輸出要求



► 解題技巧

- 使用Application的**InputBox**屬性，可以開啟輸入對話方塊。如果輸入的資料不是False(即按取消鈕)，就執行暫停的程式。
- 因為暫停後要執行巨集，所以要使用Application的**Wait**方法。
- 使用Application的**Speech.Speak**方法，可以讀出指定字串的語音。

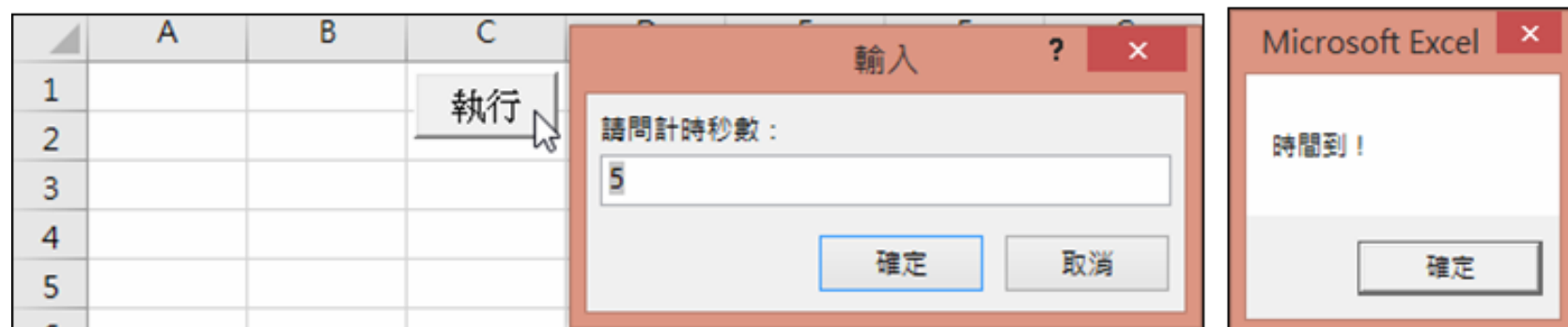
Step 3 編寫程式碼

FileName: Application 方法.xlsm (工作表 1 程式碼)

```
01 Private Sub CommandButton1_Click()  
02     s = Application.InputBox("輸入暫停秒數：", Title:="輸入", Default:="5", Type:=2)  
03     If s <> False Then  
04         Application.Wait Now + TimeValue("00:00:" & s)  
05         Application.Speech.Speak ("暫停時間到")  
06         MsgBox "暫停時間到！"  
07     End If  
08     Application.Goto Range("A1") '選取 A1 儲存格  
09 End Sub
```

► 隨堂測驗

將上面實作改為幾秒鐘後執行 maAlarm 程序，程序中會讀出「時間到」語音，並顯示「時間到！」訊息方塊。



8.4 Application 物件常用的事件

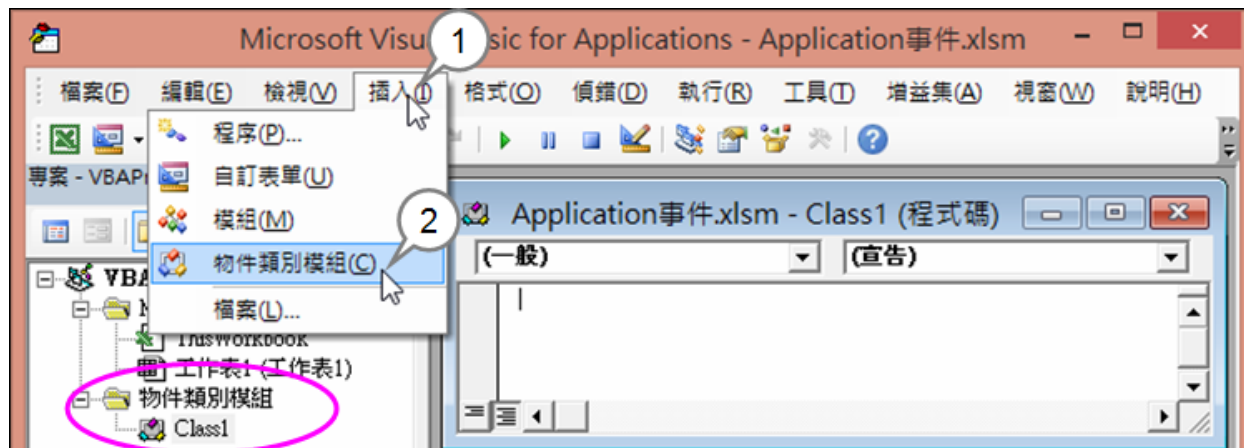
8.4.1 建立Application 物件的步驟

Continue...

- 要使用Application物件的事件前，要先建立物件後才能使用。操作步驟如下所示：

1. 建立物件類別模組

執行【插入/物件類別模組】功能，在專案中建立一個預設名稱為**Class1**的物件類別模組。



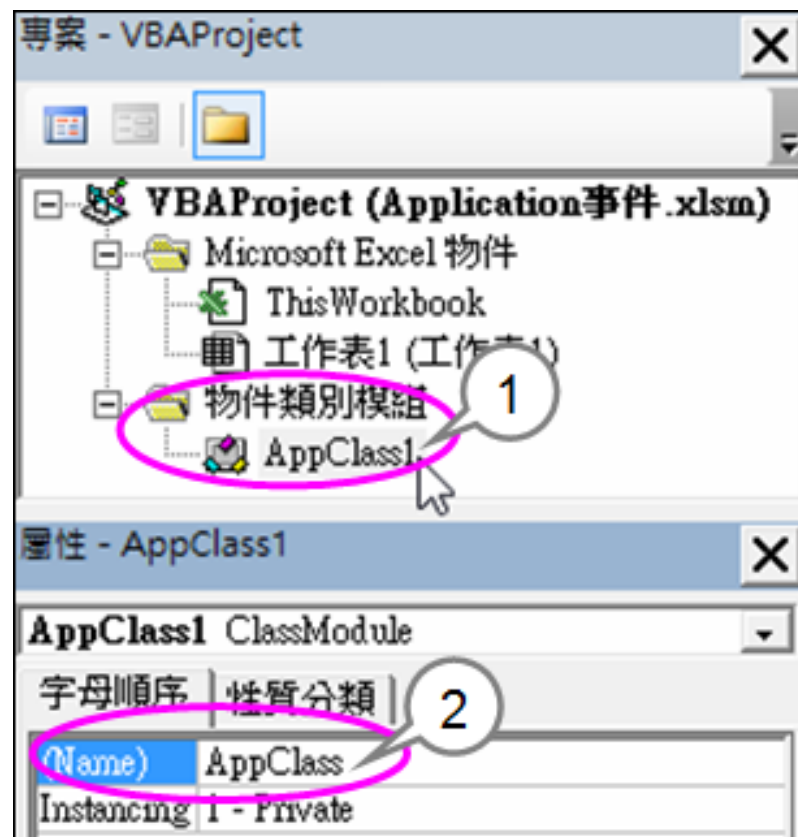
8.4.1 建立Application 物件的步驟

Continue...

2. 修改物件類別模組名稱

執行功能表【檢視/屬性視窗】指令，可以顯示屬性視窗。

先點選Class1物件類別模組後，在屬性視窗中修改Name屬性值，就可以修改模組的名稱。

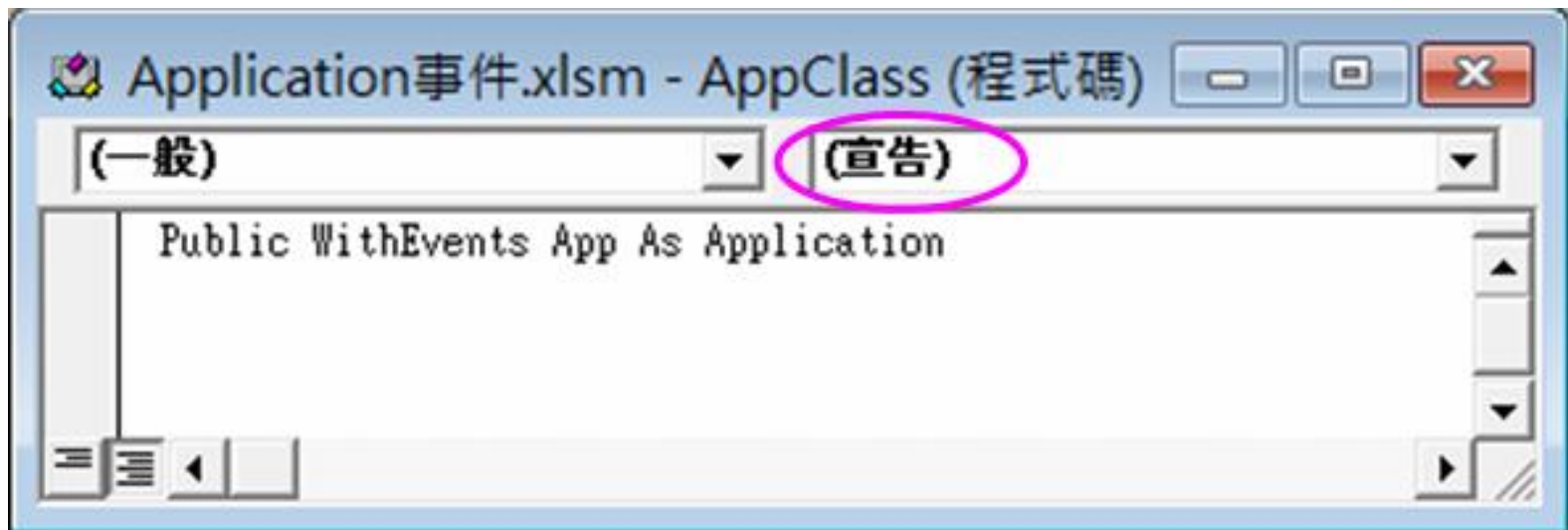


8.4.1 建立Application物件的步驟

Continue...

3. 宣告Application物件

在AppClass物件類別模組的宣告區，使用Public WithEvents敘述來宣告名稱為App的公用Application物件，App物件具有Application物件所有可觸發的事件。



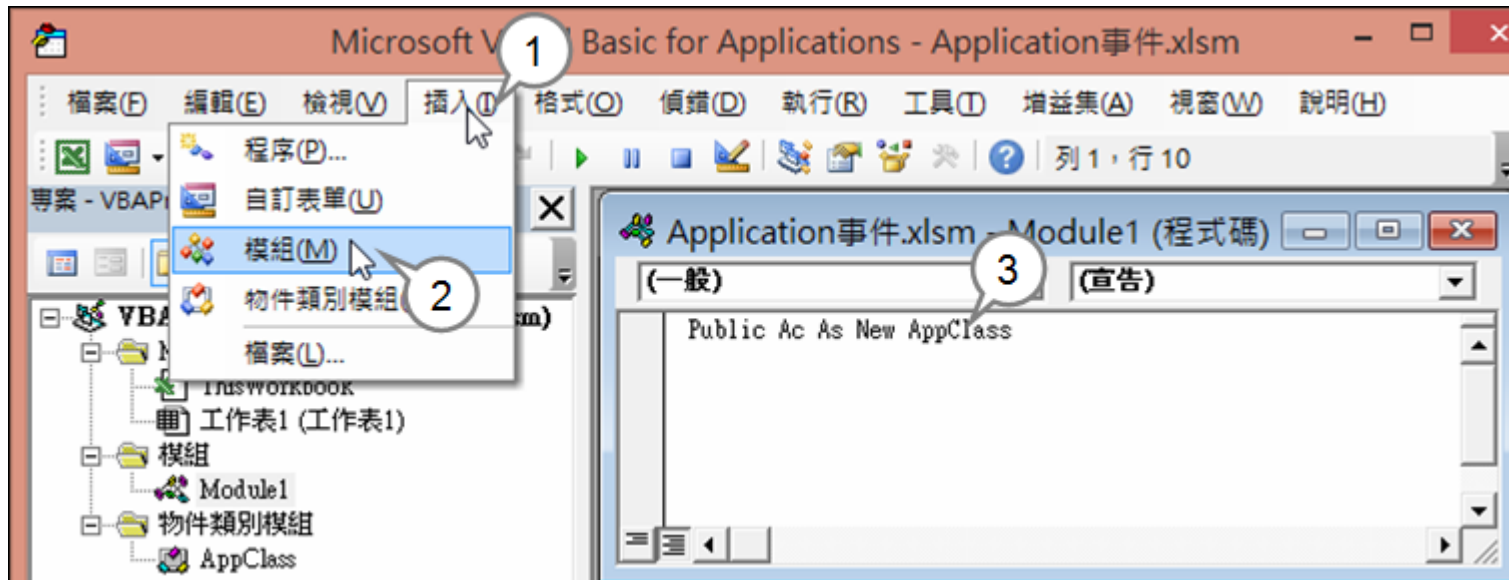
8.4.1 建立Application 物件的步驟

Continue...

4. 建立AppClass物件實體

5. 建立物件的實體前先執行【插入/模組】功能，建立預設名稱為Module1的一般模組。然後在Module1模組的宣告區，用New建立AppClass物件的公用實體Ac，寫法為：

Public Ac As New AppClass



8.4.1 建立Application 物件的步驟

Continue...

5. 指定Ac實體等於Application物件

建立一個Initialize程序，程序中用Set指定Ac實體的App等於Application物件，執行該程序後就可以觸動Application物件的所有事件。

該程序和實體的名稱使用者都可以自定，寫法為：

```
Sub Initialize()
```

```
    Set Ac.App = Application
```

```
End Sub
```

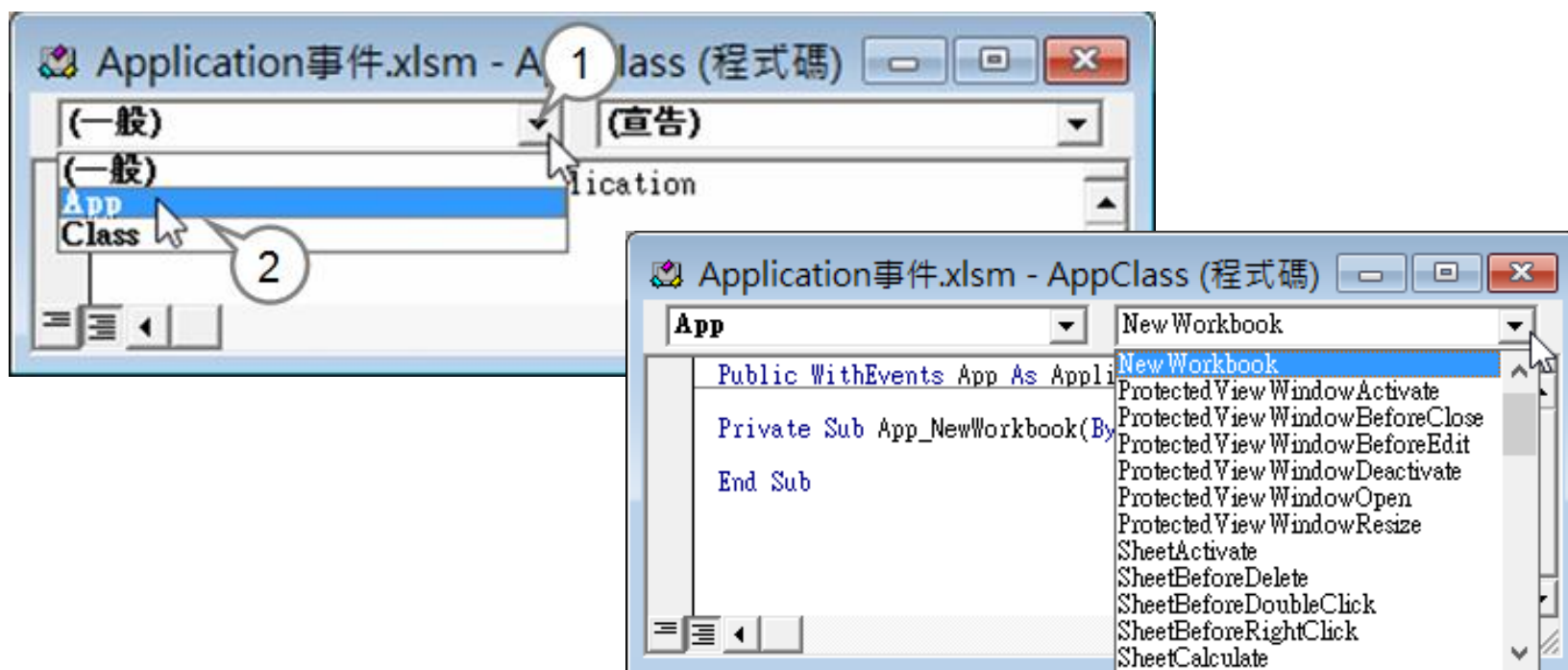
8.4.1 建立Application物件的步驟

Continue...

6. 建立Application物件事件

在物件清單中選取App物件，此時會自動建立App物件的預設事件NewWorkbook。

若要建立其他的事件，可由事件清單中選取。



8.4.1 建立Application物件的步驟

Continue...

7. 完成Application物件的App_NewWorkbook事件程序：

在App_NewWorkbook事件中，撰寫該事件發生時要執行的敘述。事件中的參數Wb代表所新增的活頁簿，例如在事件中顯示所新增活頁簿的名稱，其寫法如下：

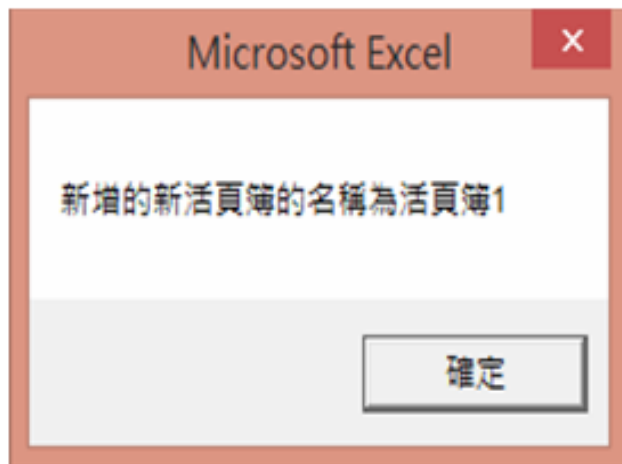
```
Private Sub App_NewWorkbook(ByVal Wb As Workbook)
    MsgBox "新增的新活頁簿的名稱為" & Wb.Name
End Sub
```



實作 FileName：Application 事件.xlsm

設計一個巨集，執行時會新增一個活頁簿，新增時會觸動 NewWorkbook 事件，在該事件中顯示新活頁簿的名稱。

► 輸出要求



► 解題技巧

- 先依照前面建立Application 物件的七個步驟，**建立AppClass物件模組和App_NewWorkbook事件**。
- 在命令按鈕的Click事件程序中，先呼叫一般模組中的**Initialize程序**，來建立Application物件實體Ac，以便觸動Application物件的所有事件。
- 使用**Application.Workbooks.Add敘述**，來新增一個活頁簿。
- 新增活頁簿時會觸動**App_NewWorkbook事件程序**，用MsgBox顯示新活頁簿的名稱。

Step 3 編寫程式碼

FileName: Application 事件.xlsm (AppClass 程式碼)

```
01 Public WithEvents App As Application
02
03 Private Sub App_NewWorkbook(ByVal Wb As Workbook)
04     MsgBox "新增的新活頁簿的名稱為" & Wb.Name
05 End Sub
```

FileName: Application 事件.xlsm (Module1 程式碼)

```
01 Public Ac As New AppClass
02
03 Sub Initialize()
04     Set Ac.App = Application
05 End Sub
```

FileName: Application 事件.xlsm (工作表 1 程式碼)

```
01 Private Sub CommandButton1_Click()
02     Call Initialize
03     Application.Workbooks.Add 新增活頁簿
04 End Sub
```


8.4.2 Application物件的常用事件

Continue...

1. NewWorkbook

當我們用Application的Add方法新增一個活頁簿時，就會觸動NewWorkbook事件。

例如：在NewWorkbook事件中，顯示新活頁簿名稱的寫法如下：

```
Private Sub App_NewWorkbook(ByVal Wb As Workbook)
    MsgBox "新增" & Wb.Name & "活頁簿" , , _
        " NewWorkbook事件"
End Sub
```

8.4.2 Application物件的常用事件

Continue...

2. SheetActivate

當我們**切換工作表**時，就會觸動Application的SheetActivate事件，其中**參數Sh**代表**目前作用工作表**。例如在SheetActivate事件中，顯示目前作用工作表名稱的寫法如下：

```
Private Sub App_SheetActivate(ByVal Sh As Object)
    MsgBox "目前作用工作表是：" & Sh.Name, , _
        "SheetActive事件"
End Sub
```

8.4.2 Application物件的常用事件

Continue...

3. SheetChange

當我們修改工作表的儲存格內容時，就會觸動SheetChange事件，其中參數Sh代表目前作用工作表，參數Target代表修改的儲存格。

例如在SheetChange事件中，顯示修改的儲存格位置的寫法如下：

```
Private Sub App_SheetChange(ByVal Sh As Object, _  
                             ByVal Target As Range)  
    MsgBox Target.Address & "儲存格被修改", , Sh.Name  
End Sub
```

8.4.2 Application物件的常用事件

Continue...

4. WorkbookBeforeClose

關閉活頁簿前會先觸動本事件，其中參數Wb代表要關閉的活頁簿，參數Cancel代表是否要取消關閉，若要取消關閉可以設Cancel參數值為True。

例如在WorkbookBeforeClose事件中，詢問是否要關閉活頁簿，若按否鈕就取消關閉活頁簿：

```
Private Sub App_WorkbookBeforeClose(_  
    ByVal Wb As Workbook, Cancel As Boolean)  
    Dim R As Integer  
    R = MsgBox( "是否要關閉活頁簿？" , vbYesNo, _  
        "WorkbookBeforeClose事件")  
    If R = vbNo Then Cancel = True  
End Sub
```

8.4.2 Application物件的常用事件

Continue...

5. WorkbookBeforePrint

當列印文件前會先觸動本事件，參數Wb代表要列印的活頁簿，參數Cancel代表是否要取消列印。

例如在WorkbookBeforePrint事件中，詢問是否要列印文件，若按 否 鈕就取消列印：

```
Private Sub App_WorkbookBeforePrint(_  
    ByVal wb As Workbook, Cancel As Boolean)  
    Dim R As Integer  
    R = MsgBox( "是否要列印文件？" , vbYesNo, _  
        "WorkbookBeforePrint事件")  
    If R = vbNo Then Cancel = True  
End Sub
```

► 隨堂測驗

繼續上面的實作，若修改活頁簿內就詢問是否要儲存檔案，若按
鈕就儲存活頁簿。當關閉活頁簿時會詢問是否要關閉活頁簿，若按
鈕就取消關閉活頁簿。

是(Y)

否(N)

