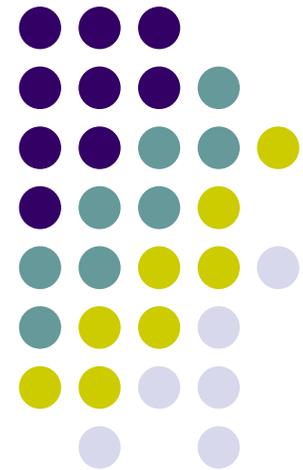


# Automation of Information Management

---

資訊管理自動化

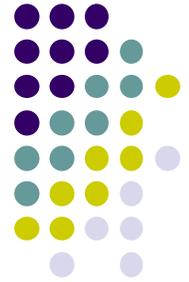




# 辦公室個人資訊管理工具

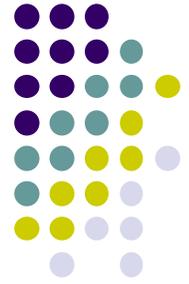
- Microsoft Office
  - 文書排版工具：Word
  - 試算圖表工具：Excel
  - 流程圖繪製工具：Visio
  - 小型資料庫工具：Access
  - 網頁製作工具：FrontPage
  - 印刷出版工具：Publisher
  - 簡報軟體：PowerPoint
  - 個人事務管理工具：Outlook
  - 專案管理工具：Project
  - 圖形處理工具：Picture Manager
  - 快速筆記工具：One Note
  - etc

# 文書排版工具 - Word



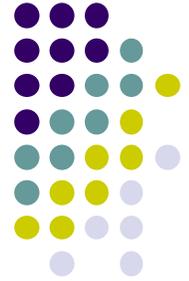
- 排版觀念
  - 文件版面格式統一設定，不重複相同的設定動作
  - 排版 ≠ 打字
- 檔案管理
  - 檔案資訊
  - 存檔格式
  - 列印設定
- 文字編輯
  - 剪下、複製、貼上、選擇性貼上
  - 搜尋、取代

# Word (cont.)



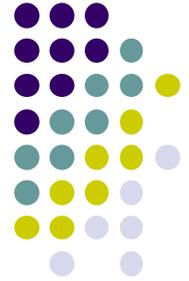
- 頁面檢視
  - 文件檢視模式、螢幕比例
  - 輔助顯示：尺規(Ruler)、格線(Grid)、功能窗格
  - 視窗配置：畫面分割、螢幕捲動、視窗切換
  - 巨集設定
- 常用功能
  - 字型設定
    - 字體、上下標、底線、全形與半形文字
  - 段落設定
    - 段落間距、排列方式、定位點
  - 樣式設定
    - 內文、標題、

# Word (cont.)

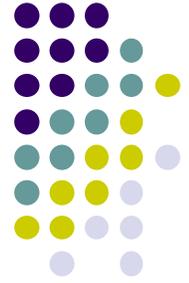


- 版面配置 → 格式排版
  - 版面設定
    - 直書或橫書格式
    - 版面邊界
    - 分隔設定
      - 分欄觀念—同一頁中分成數欄
      - 分頁觀念—同樣的版面中換至新頁
      - 分節觀念—同一份文件中存在不同版面格式
    - 段落分欄
  - 段落與排版概念
    - 縮排、段落間距、定位點、對齊方式
    - 項目符號及編號段落
  - 頁面背景、框線與網底的設定

# Word (cont.)



- 格式排版
  - 文件樣式與格式設定
    - 新增與修改樣式
    - 套用樣式與清除格式
  - 文件排版與目錄製作
- 參考資料
  - 製作目錄
    - 分節版面製作與分節頁碼設定
    - 本文目錄：設定內文標題樣式
    - 圖表目錄：設定圖表標號樣式
  - 註腳
  - 標記：標號(圖表目錄)、交互參照(參考引文)
  - 文件索引



# Word (cont.) – 插入物件

- 製作表格
  - 儲存格與行列概念
  - 儲存格、行與列的增加、刪除與合併
  - 表格分割(含跨頁概念)與合併
  - 表格與文字轉換：以TAB控制自動產生表格
  - 儲存格簡易計算公式
- 圖表製作
  - 直接啟動Excel圖表功能
- 特殊符號
- 連結：超連結、書籤、交互參照



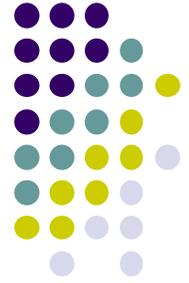
# Word (cont.) – 插入物件

- 插入方程式
  - 新版：直接選取方程式按鈕
  - 舊版：物件→方程式編輯器3.0(與2003版文件相容)
- 插入圖案
  - 自選圖案(jpg, png, bmp, wmf, emf, gif ...)
  - 內建圖形、美工圖案(線上圖形庫)
  - 點選圖片或按右鍵進行格式編輯
    - 裁剪模式
    - 縮放大小：等比例、任意變形...等
    - 圖文位置：與文字排列、覆蓋文字、方形排列...等
    - 其他效果：旋轉、光線、解析度...等



# Word (cont.) – 插入物件

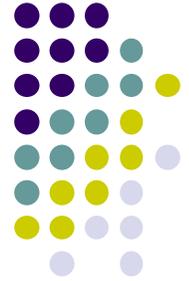
- 頁首及頁尾
  - 頁首、頁尾、頁碼、日期...
- 頁面
  - 封面、空白頁
  - 快速分頁(Ctrl + Enter)
- 文字方塊
  - 設置版面中的版面
  - 將圖、表、文字結合成一個物件
  - 格式設計：具備圖層功能，可當作圖案處理



# Word (cont.)

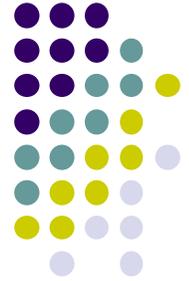
- 郵件(大量重複性資料)處理
  - 建立信封與標籤
  - 合併列印
    - 製作列印清單表格(可搭配Excel)
    - 選取清單
    - 製作文件(編輯文件內容並插入合併欄位)
    - 編輯清單(選擇要插入的清單資料)
    - 完成合併→編輯合併文件或列印
- 校閱文件
  - 追蹤修訂、新增註解、字數檢查、簡繁體轉換...等
  - 連結筆記(One Note)→文字方塊貼圖筆記工具

# 流程圖表工具 - Visio

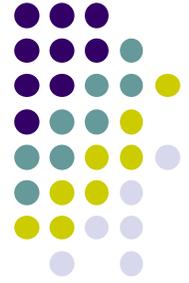


- 流程圖概念
  - 流程圖
  - 組織圖
  - 系統架構圖
  - 建置規劃圖
  - 排程圖
- 物件與群組概念
  - 類別圖形(Stencil)
  - 物件美工修飾
  - 群組順序

# Visio (cont.)

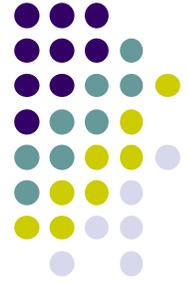


- 圖形物件調整
  - 圖形的旋轉
  - 圖形的對齊
  - 圖形的聯結
  - 圖形的排列
- 出圖頁面設定
  - 背景設定
  - 頁面大小調整
  - 複製圖形至其他軟體



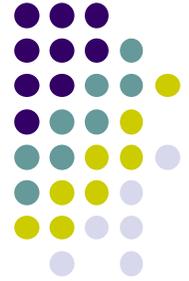
# 試算分析工具 - Excel

- 檔案管理
  - 檔案資訊
  - 存檔格式
  - 版面設定
  - 列印設定與列印範圍
- 儲存格資料編輯
  - 剪下、複製、貼上、選擇性貼上
  - 搜尋、取代
  - 工作表複製與刪除
  - 儲存格填滿與清除



# Excel (cont.)

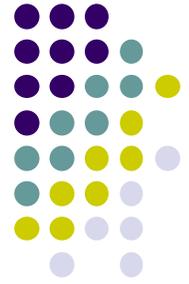
- 檢視頁面
  - 版面模式
  - 頁首(Header)與頁尾(Footer)
  - 螢幕比例
  - 工具列
  - 資料編輯列與工作窗格
- 工作視窗調整
  - 凍結窗格
  - 隱藏視窗
  - 新增視窗與並排視窗



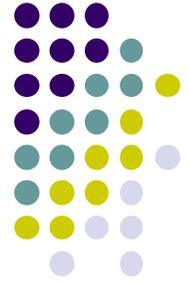
# Excel (cont.)

- 格式設定
  - 儲存格格式設定
    - 資料格式
    - 對齊方式
    - 字型設定
    - 外框與圖樣
  - 欄列格式調整
    - 欄寬或列高之自動調整與手動調整
    - 欄列之隱藏
  - 工作表隱藏、複製與刪除

# Excel (cont.)



- 插入物件
  - 新增欄列及儲存格
  - 分頁功能
  - 統計圖表繪製
    - 圓餅圖：呈現數值間的比例關係
    - 直條圖、折線圖：橫軸座標彼此獨立
    - 散佈圖：橫軸座標彼此相依
    - 區域圖：呈現數列各分量對應於橫軸的比例
    - 雷達圖(風花圖)：呈現極座標模式比較距離關係
    - 盒鬚圖：呈現數據的統計值(如初始值、終止值與平均)
    - 混合圖：呈現不同值域的主副軸，混合直條與折線
  - 趨勢線
    - 迴歸分析：找出通過數據分布區的期望公式



# Excel (cont.)

- 基本函數之使用
  - 文字函數
  - 數學函數
  - 統計函數
  - 其他
- 試算工具
  - 試算表之建立
  - 跨頁試算功能設定
  - 報表製作與應用
- 資料分析
  - 匯入外部資料
  - 自動排序
  - 資料篩選與樞紐分析
  - 清單製作



# 試算分析應用

- 解聯立方程式
  - $AX=C \rightarrow A^{-1}AX=A^{-1}C \rightarrow IX=A^{-1}C$ 
    - 其中，X為求解向量，I為單位矩陣， $A^{-1}$ 為A的反矩陣
  - 應用MINVERSE函數與MMULT函數計算
  - 使用INDEX函數定義矩陣中元素值
- 計算定積分
  - 計算曲線下面積
  - 將曲線下面積切割為許多矩形或梯形後將這些小面積累加起來
- 從數據分佈尋找趨勢線(迴歸分析)



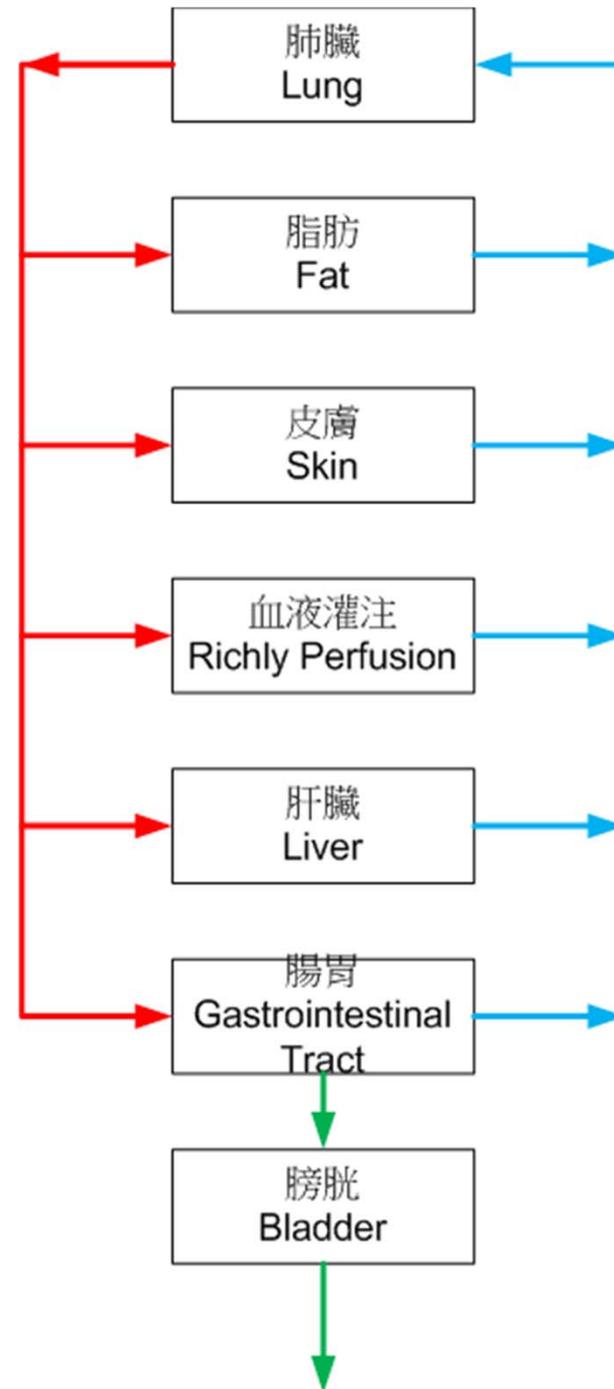
# 藥物動力學計算應用

- 藥物進入人體管道如皮層、食道、呼吸道、血管，會存留在體內的組織或器官中
- 經實驗歸納存留與代謝之係數與濃度可形成上述之聯立方程式， $X$ 為存留在各組織器官中的藥物濃度。
- PBPK模式係一種計算評估環境化學物質在生物體內劑量之研究，模擬由血液將暴露物質帶至全身，在各生理單元進行分布、累積、排出或代謝等時間-濃度的預測與解釋。

# PBPK modelling

- Physiologically based pharmacokinetic:

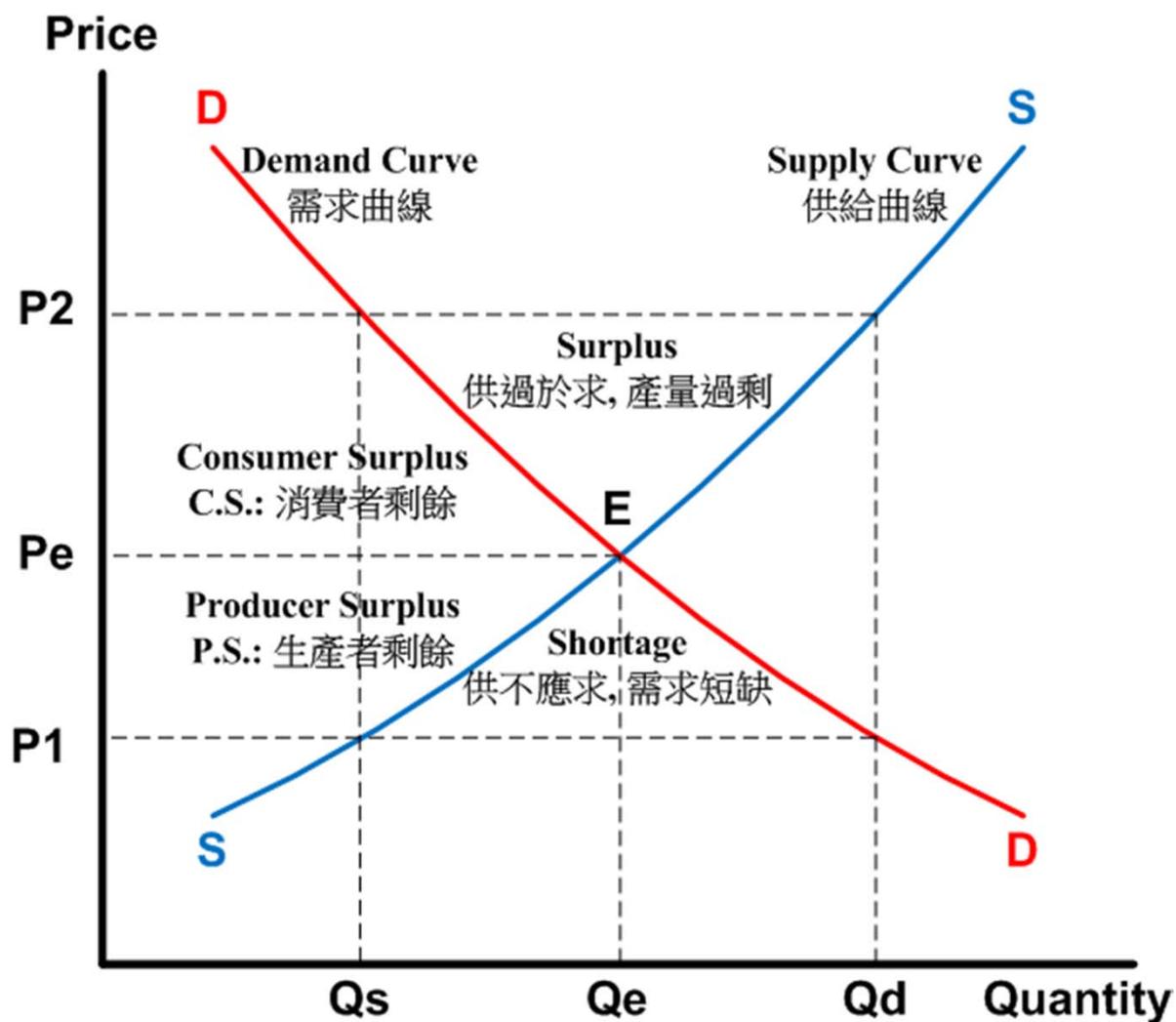
[https://en.wikipedia.org/wiki/Physiologically\\_based\\_pharmacokinetic\\_modelling](https://en.wikipedia.org/wiki/Physiologically_based_pharmacokinetic_modelling)

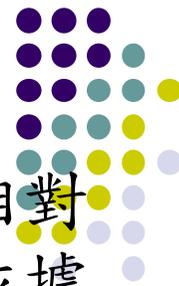




# 市場行銷管理分析

- 供需理論



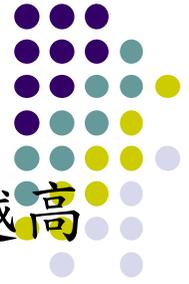


- 供給(supply)

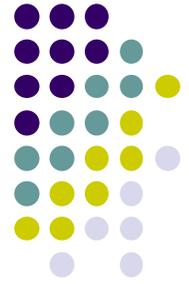
- 指特定市場上在一定時期內，與每一銷售價格相對應，生產者願意且能供應的商品數量。生產量依據利潤最大化條件決定，它在短期往往是固定，但供給量則可以隨時變動。
- 供給定律：假設其他因素不變，當一件物品的相對價格上升時，其供給量會上升，反之亦然。換言之兩者成正比關係，形成為「供給曲線」。

- 需求(demand)

- 指人們有能力購買，並願意購買某商品的慾望。隨價格升降，在每段時間內所願意買某商品的數量。
- 在某價格下，消費者願意購買某貨物的總數量為「需求量」。不同價格下，需求量會不同。價格與需求量的關係，是為「需求曲線」。



- 供需關係
  - 商品價格如果越高，生產者願意生產的數量會越高，消費者願意購買的數量會越低。
- 市場超額需求( Market Excess Demand )
  - 是指市場需求量大於市場供給量的情形。從附圖可知，會造成超額需求，是因為價格太低的緣故。例如價格 $P_1$ ，依需求曲線想要買進的數量是 $Q_d$ ，而供給曲線想要賣出的數量只有 $Q_s$ ，兩者之間的差距即為市場超額需求。
  - 超額需求下市場求過於供，價格必定上漲，最後會達到新的均衡點 $E$ ，這時新的價格為 $P_e$ ，新的成交量為 $Q_e$ 。透過價格的上漲，引導需求量減少，供給量增加，最終達到新的均衡。反之，則為超額供給(Excess Supply)，如價格 $P_2$ 與 $(Q_s, Q_d)$ 關係



# VBA入門

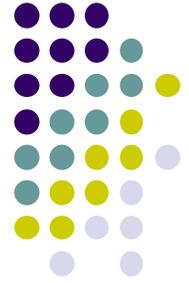
- VBA巨集語言指的就是Visual Basic For Application，主要用做設計Office 應用程式中的自訂功能及自動化而設計。
- 在應用程式中，若包含了VBA程式語言的話，程式開發人員就可以在不同應用程式中，使用共同巨集語言進行程式開發的工作，並自行製作Word、Access、Excel、PowerPoint、FrontPage和Outlook等程式中的解決方案。
- 應用VBA讓需重覆操作的資料有效率的自動化處理

# Excel VBA基礎-建立巨集



- 檔案→選項→自訂功能區→勾選「開發人員」
- 建立設計按鈕：開發人員→設計模式→ 
- 執行巨集需在低安全模式
- 按鈕屬性：按右鍵點選「內容」
  - 調整按鈕的外觀
  - 注意按鈕名稱，ex, CommandButton1
- 設計程式：按右鍵點選「檢視程式碼」
  - 進入Visual Basic頁面，會出現一段程式區塊，如  

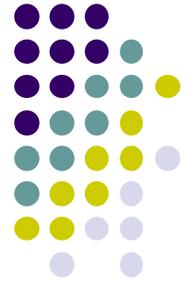
```
Private Sub CommandButton1_Click()  
End Sub
```
  - 在區塊中撰寫程式



## Excel VBA基礎-建立巨集(cont.)

- 檔案→選項→自訂功能區→勾選「開發人員」
- 建立工作頁程式：開發人員→Visual Basic
- 檢視程式碼：
  - 進入Visual Basic頁面，點選工作頁或新增模組，建立如下之程式區塊，如  
Sub VBATest()  
  
End Sub
  - 在區塊中撰寫程式
  - 按綠色執行鍵執行程式
- 此工作頁程式或模組程式可作成巨集

# Excel VBA基礎-儲存格資料存取

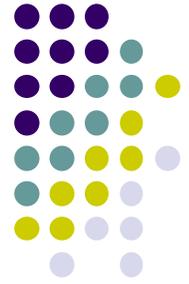


- 存取工作頁Sheet1之儲存格資料
  - `X = Worksheets("Sheet1").Cells(3,5)`  
表示將該分頁第3列第5欄(E3)的資料指定給變數X
  - `Worksheets("Sheet1").Cells(5,3) = "ABC"`  
表示該分頁C5儲存格的值為ABC
  - `Worksheets("Sheet1").Cells(5,6).Formula = ("=SUM(A5:E5)")` 或  
`Worksheets("Sheet1").Range("F5").Formula = ("=SUM(A5:E5)")`  
表示該分頁F5儲存格的值為Excel函數SUM計算A5至E5的加總結果
- 執行：在非設計模式下點擊按鈕

# Excel VBA基礎-資料格式與運算



- 資料格式(等式右邊的資料指定給左邊的變數)
  - 文字：以雙引號引起，ex: “String”
  - 數字：分整數與實數，ex: 10, 10.5
- 宣告變數：Dim 變數名稱 As 資料型態
  - Ex: Dim str As String
  - 常用資料型態如String, Integer, Double
- 變數四則運算
  - 數字運算：+, -, \*, /, mod
  - 文字與字串結合：“Mary” & (10)會得到字串Mary10

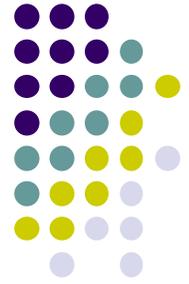


# Excel VBA基礎-基本邏輯判斷

- If 判斷式 Then ... End If
  - 如果if內的判斷式成立, 執行Then後的區塊
- If ... Then ... Else ... End If
  - 如果if內的判斷式成立, 執行Then後的區塊  
否則便執行Else後的區塊
- If ... Elself ... Then ... Else ... End If
  - 如果if內的判斷式成立, 執行Then後的區塊  
否則就進行Elself後的判斷式, 成立時就執行Then  
後面的區塊, 否則便繼續判斷下一個Elself
  - 如果Elself後的判斷式皆不成立, 則執行Else後區塊



- 例如依據某儲存格的文字  
在另一定義相對應的數值
  - `X = Worksheets("Sheet1").Cells(1,1)`  
`If X="A" Then`  
    `Worksheets("Sheet1").Cells(1,2)=90`  
`Elseif X="B" Then`  
    `Worksheets("Sheet1").Cells(1,2)=80`  
`Else`  
    `Worksheets("Sheet1").Cells(1,2)=70`  
`End If`
- 邏輯判斷符號
  - `=, >, >=, <, <=, AND, OR, NOT`
  - `Worksheets("Sheet1").Cells(1,1) = null` 表該儲存格沒有資料



# Excel VBA基礎-基本迴圈控制

- 善用迴圈觀念重覆處理相同的動作，  
可在一瞬間處理完大量複製貼上的動作
- for迴圈：利用起始與中止條件控制迴圈
  - 語法：for i = 起始值 to 終止值 ... next i  
For I = 1 To n  
    Worksheets("Sheet1").Cells(I,2) = \_  
Worksheets("Sheet1").Cells(I,1)  
Next I
  - 以上，把Sheet1工作頁第1欄之第1到n列資料\*10後，  
指定給該頁第2欄之第1到n列資料
  - 敘述太長時可用 \_ 底線符號連接兩列程式碼
  - 可利用 Exit For 指令中斷迴圈

# Excel VBA基礎-基本迴圈控制 (Cont.)



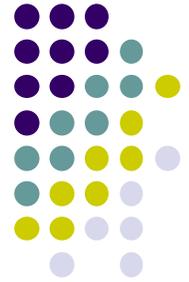
- **while迴圈**：用邏輯條件控制迴圈
  - 語法：while 邏輯條件 ... end while  
While n <= 10  
    Worksheets("Sheet1").Cells(n,2) = n \* 10  
    n = n + 1  
End While
  - 以上，當n<=10時，將n\*10代入Sheet1工作頁第2欄之第n列
  - 可以用Exit While中斷迴圈
  - While迴圈可能在啟動時因邏輯條件不成立而停止

# Excel VBA基礎-基本迴圈控制(Cont.)

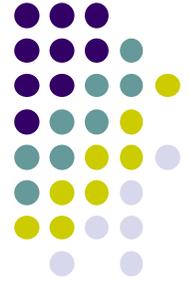


- do迴圈：用邏輯條件允許迴圈至少執行一次
  - 語法：do while 邏輯條件 ... loop  
或 do ... loop until 邏輯條件
  - Do While n <= 10  
Worksheets("Sheet1").Cells(n,2) = n \* 10  
n = n + 1  
Loop
  - Do  
Worksheets("Sheet1").Cells(n,2) = n \* 10  
n = n + 1  
Loop Until n > 10
- 可以用Exit Do中斷迴圈

# Excel VBA基礎-陣列觀念



- 陣列：用一個變數儲存相同型態的資料集合
  - 宣告方式與一般變數相同，常用資料型態如 Integer, Double, String 等
  - 固定長度陣列宣告 → `Dim a(10) As Integer`
  - 變動長度陣列宣告 → `ReDim a(n) As String`  
其中，`n`為整數變數，陣列大小因`n`值而異
  - 不定長度陣列宣告 → `Dim a As Variant`  
應用在取得資料時並不知道會有多少資料，如  
`a = Array(1, 5, 6)` → 定義3個元素之整數陣列  
`a = Split("a-b-c", "-")` → 取得用橫線符號(-)分隔字串後得到的字串集合，`a`自動成為3個元素的陣列
- 陣列的元素從0起算，如`a(0)`表第一個元素
  - 使用 `LBound`及 `UBound`可取得陣列之首尾元素位置，如上例中，`LBound(a)`為0，`UBound(a)`為2



# Excel VBA基礎-陣列應用

- 以下範例取得欄位資料數目進行處理

```
Dim m As Integer
```

```
‘取得第1欄(A1為第一欄第一個儲存格)最後一列有資料的列數
```

```
m = Worksheets("Sheet1").Range("A1").End(xlDown).Row
```

```
ReDim x(m) As Double
```

```
For i = 0 To m-1 ‘陣列長度為m, 起始元素位置為0, 最後一個元素位置為m-1
```

```
    Randomize ‘啟動亂數機制
```

```
    x(i) = Rnd * 1000 Mod 100 + 1 ‘建立一個小於100的亂數
```

```
    If x(i) < Worksheets("Sheet1").Cells(i+1, 1) Then
```

```
        str = "Good"
```

```
    Elseif x(i) >= Worksheets("Sheet1").Cells(i+1, 1) AND x(i) < 100 Then
```

```
        str = "Normal"
```

```
    Else
```

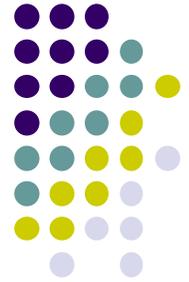
```
        str = "Poor"
```

```
    End If
```

```
    Worksheets("Sheet1").Cells(i+1, 2) = str & "(" & x(i) & ")"
```

```
Next i
```

# Excel VBA基礎-副程式概念



- Sub與Function

- 無傳回值副程式 – 代入引數計算, 使用Call呼叫  
Sub sub01(變數 As 資料型態)  
    計算傳入之變數資料, 運算完未傳回計算結果  
End Sub

- 有傳回值函式 – 傳回值為函式名, 使用=取得值  
Function 函式名(變數 As 資料型態, ...) As 資料型態  
    函式名 = 變數運算, 傳回計算結果值  
End Function

- 副程式之引數

- 輸入之引數可不只一個, 亦可無引數
- Excel VBA之副程式引數屬於傳遞位址, 亦可做為回傳值使用 → 引數值可在計算後改變

# Excel VBA基礎-副程式範例



## ● 主程式部分

```
Sub subtest()
```

```
Dim x As Double, y As Double, x_y As Double
```

```
Dim z As Variant
```

```
x = Worksheets("Analysis").Cells(5, 6)
```

```
y = Worksheets("Analysis").Cells(5, 7)
```

```
z = Array(1.4, 2.6, 3.1, 3.2, 6.1, 3.2, 6.6)
```

```
x_y = sub01(x, y)
```

```
Call sub02("Analysis", 5, 8, x_y)
```

```
For i = 0 To UBound(z)
```

```
    Worksheets("Analysis").Cells(i + 10, 6) = z(i)
```

```
Next i
```

```
Call sub03(z)
```

```
For i = 0 To UBound(z)
```

```
    Worksheets("Analysis").Cells(i + 10, 7) = z(i)
```

```
Next i
```

```
End Sub
```

說明：

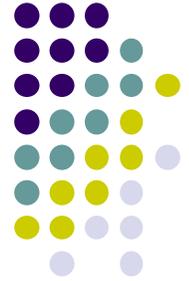
宣告實數變數x, y, x\_y及陣列變數z, 其中x, y值從Analysis工作頁之儲存格F5及G5 (即(5,6)及(5,7))輸入;而z之資料由Array定義

呼叫函式sub01使其值回傳給變數x\_y

呼叫副程式sub02將x\_y值寫入Analysis工作頁儲存格H5中 (即(5,8))輸出陣列z之原始值

呼叫副程式sub03對z做排序輸出陣列z排序後的值

# Excel VBA基礎-副程式範例



## ● 副程式部分

```
Function sub01(a As Double, b As Double) As Double
```

```
sub01 = a + b
```

```
End Function
```

```
Sub sub02(sheet As String, m As Integer, n As Integer, x As Double)
```

```
Worksheets(sheet).Cells(m, n) = x
```

```
End Sub
```

```
Sub sub03(x As Variant)
```

```
Dim n As Integer
```

```
n = UBound(x)
```

```
For i = 0 To n - 2
```

```
    For j = i + 1 To n
```

```
        If x(j) > x(i) Then
```

```
            tmp = x(j)
```

```
            x(j) = x(i)
```

```
            x(i) = tmp
```

```
        End If
```

```
    Next j
```

```
Next i
```

```
End Sub
```

說明：

sub01: 輸入x, y相加得到sub01回傳

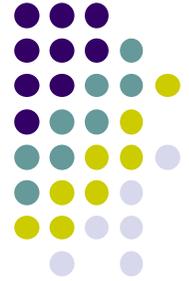
sub02: 輸入工作頁名稱sheet, 儲存格位置(m,n), 輸出值x, 將x寫到指定儲存格

sub03: 輸入陣列x進行氣泡排序法, 將陣列元素由大排到小, 排序後陣列x元素的位置重新改變



# 資料庫工具 - Access

- 資料庫基本概念
- 資料檔案建立
  - 建立空白資料庫
  - 自Excel匯入資料
- 資料格式觀念
  - 數字：整數與浮點數
  - 文字：字元與字串
  - 日期
  - 空值(null)的概念



# Access (cont.)

- 關聯式資料表觀念
  - 主鍵(Primary Key)
  - 外來鍵(Foreign Key)
  - 唯一值(Unique)
- 資料正規化(Normalize)
  - 分析資料功能
  - 建立資料關聯圖
- 查詢資料表
  - 建立查詢
  - 結構化查詢語法(SQL)概念