

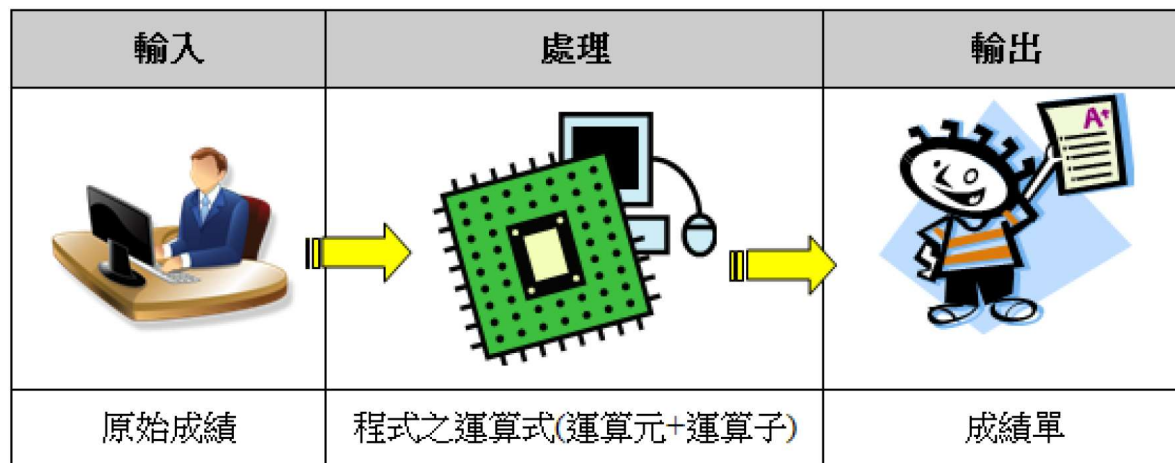
## 5 資料的運算



### 5-1 運算子與函數

電腦處理資料的過程為：輸入—處理—輸出，其中「處理」的程序除了變數賦值、讓資料儲存在電腦記憶體基本處理之外，常見的資料操作有運算子與函數兩種模式。

【示意圖】



#### 5-1-1 資料的操作：運算子

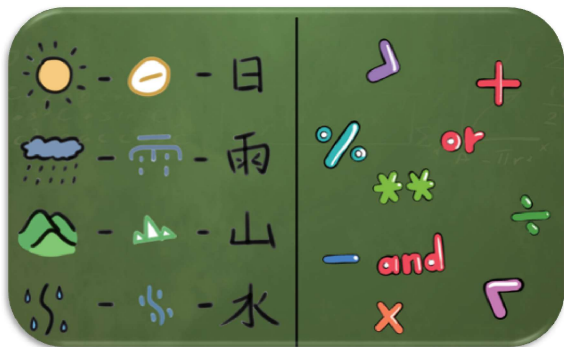
##### 運算式的定義

運算式(Expression)是由運算元(Operand)與運算子(Operator)所組成。

【例如】 $A = B + 1$

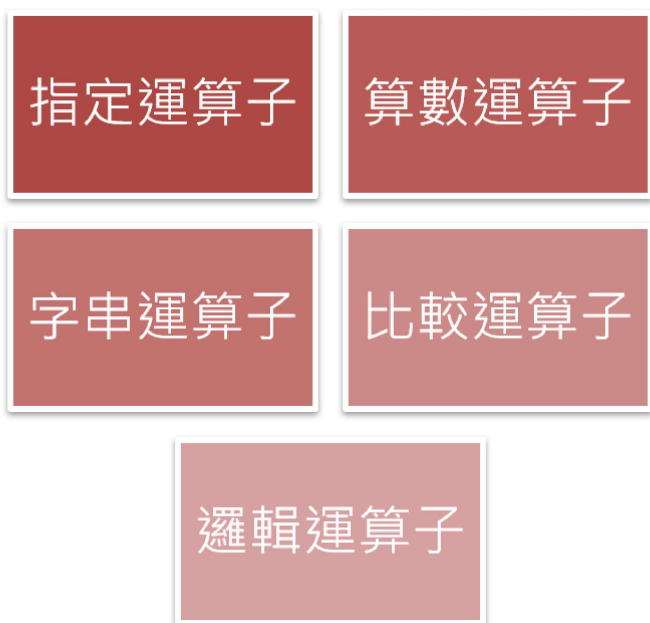
其中「A,B,1」稱為運算元，「=,+」則稱為運算子。

- 「運算式」就像平常用的數學公式一樣。
- 「運算元」不是變數就是常數。
- 「運算子」是一種符號，具有特定的功能，用符號來表示某種意義。
  - 例如：指定運算子，用等號(=)表示。
  - 等號右邊先運算，再將運算結果指定給左邊變數。
  - 變數賦值， $a = 1 + 2$ 。



### 常見的運算子

- 瞭解數字的操作：算數運算子、指定運算子(算數指定運算子)
- 瞭解文字的操作：字串運算子。
- 瞭解布林的操作：比較運算子、邏輯運算子。



## 5-1-2 資料的操作：函式

把常用的功能模組化，以函式(function)的方式撰寫程式，方便日後呼叫使用。

- **內建函式**：Python內建許多必要的函式，可以直接使用，像 `print()`、`type()` 與 `help()` 等。
- **標準函式庫**：有一些常用的Python函式，不可以直接使用，需要 `import`後方可使用，像是數學模組 `math`、或隨機模組 `random`等，稱為標準函式庫。
- **自訂函式**：開發者自己也可以撰寫函式使用。

## 5-2. 數值資料的運算

---

### 5-2-1 算術運算子

算數運算子(Arithmetic Operator)是程式語言中使用率最高的運算子，常用於整數與浮點數進行數字四則運算。

常用的數學運算符號有七種：加、減、乘、除、次方、餘數與商數。

表 2-7 算術運算子

運算子	說明	舉例	結果
+	加	$A=5+2$	$A=7$
-	減	$A=5-2$	$A=3$
*	乘	$A=5*2$	$A=10$
/	浮點除法	$A=5/2$	$A=2.5$
//	整數除法 ( 去除小數點 )	$A=5//2$	$A=2$
%	相除後求餘數	$A=5\%2$	$A=1$
**	次方	$A=5**2$	$A=25$

- 當成計算機直接做計算。
- 變數賦值後再來做計算。

範例：七種算數運算子運算

```
1  ## 算術運算子：變數運算
2  a = 11
3  b = 3.8
4
5  a + b
6  a - b
7  a * b
8  a/b
9  a//b
10 a%b
11 a**b
```

### 隨堂練習6: 統計人數

假期結束了，方老師問班上同學們：誰在寒假裡看過『上陽賦』？有一半同學舉手。又問：誰看過『哲仁皇后』？有五分之二的同學舉手。方老師發現，又7位同學兩部都看過。張小青去爺爺家，沒有網路，兩部劇都沒有看過。所以，他問老師：這兩部都沒看過的同學，要不要舉手讓您數一數呢？方老師笑說：不用了，我已經知道答案了。請問如果班上有50位同學，那麼多少人兩部都沒看過呢。

```
1  #|echo: false
2  #|eval: false
3
4  ## Input Data
5  A = 50/2; B = 50 * 2/5; A_B = 7
6  ## Process Data
7  C = A + B - A_B
8  D = 50 - C
```

```

9  ## Output Data
10 print("兩部都沒看的", D)

```

## 5-2-2 算數指定運算子

### 指定運算子

- 指定運算子就是變數賦值中的等號 `=`，將等號右邊的值指定給左側的變數名字（名字 = 值）。
- 指定運算子右側可以是常數、變數或運算式，最終都會指定給左邊的變數名字。

### 複合指定運算子

- 「複合指定運算子」(Compound Assignment Operators)：指定運算子搭配某個運算子
- 算術指定運算子，為指定運算子和算術運算子的綜合。

表 2-10 算術運算子範例

運算子	說明	範例	將範例進行縮寫
<code>+=</code>	加	<code>A=A+2</code>	<code>A += 2</code>
<code>-=</code>	減	<code>A=A-2</code>	<code>A -= 2</code>
<code>*=</code>	乘	<code>A=A*2</code>	<code>A *= 2</code>
<code>/=</code>	浮點除法	<code>A=A/2</code>	<code>A /= 2</code>
<code>//=</code>	整數除法（去除小數點）	<code>A=A//2</code>	<code>A //= 2</code>
<code>%=</code>	相除後求餘數	<code>A=A%2</code>	<code>A %= 2</code>
<code>**=</code>	次方	<code>A=A**2</code>	<code>A **= 2</code>

## 5-2-3 運算的優先順序

製作算數運算式時，要小心兩個觀念：「運算子優先順序」(operator precedence)與「運算子結合規則」(operator associativity)。

- 「運算的優先順序」和數學習慣相同。
  - 先乘除後加減是大家耳熟能詳的優先順序
  - 次方運算子優先權最高、加減運算子則最低。
- 「運算子結合規則」通常是「左到右」運算。
- 「調整運算優先順序」可以使用小括號 `()`。
  - 小括號用來明確規定那個運算要先做。
  - 不能用中括號或大括號表示優先順序。
  - 不需要死背運算子的「優先順序」與「結合規則」。
- 當我們不確定、有疑慮時，請記得多請教Google大神。

範例一	乘除先運算	接著加減運算
$F=2+3*5-14/7$	$F=2+15-2$	$F=15$

範例二	括號先運算	接著求餘數運算
$F=(2+3)\%4$	$F=5\%4$	$F=1$

=> 在數學運算中，括號優先權最高，其次是\*\*指數 → %餘數 → //商 → /除法 → \*乘法 → + 加法 → -減法。 \*\*

- 在處理一個多運算子的運算式時，有一些規則和步驟是必須遵守的：
  - 當遇到一個運算式時，先區分運算子與運算元。
  - 依照運算子的優先順序作整理動作。
  - 將各運算子根據其結合順序進行運算。

**範例：**請利用Python計算 $(23+32)\times 4$ ，以及9開三次方？

```

1  ## 數字運算的優先順序，不用背，多利用小括號。
2  23 + 32 * 4
3  ((23 + 32) * 4)
4  (9**1/3)
5  (9**(1/3))

```

**加分題：**小猴子摘水果

有一天，一隻小猴子來到果園裡，牠走到蘋果樹下，看到滿樹的蘋果，就爬上去摘了2個。又看到不遠的地方有一個梨子樹，又大又黃，就丟掉蘋果，爬到梨子樹摘了1個梨子。正準備吃的時候，看到前方又紅又大的桃子樹，就高高興興地跑到桃樹上，摘了3個桃子。這時傳來小鳥的叫聲，鳥在香蕉樹上，小猴子當然就立馬飛奔去摘香蕉，最後吃了6根香蕉。請問這隻猴子總共摘了多少個水果，手中最後的水果是什麼？請你用Python記錄一下！

在 Python 中，变量会根据程序运行而发生变化，以小猴子每次摘的水果数量为例，可以看出 `mc` 变量每次都会更换成一个水果的名称。`i` 变量会随摘的水果数变化而变化。

摘的过程	摘的水果	摘的个数	摘前水果数量	摘后水果数量
	<code>mc='苹果'</code>	2	<code>i=0</code>	<code>i=2</code>
	<code>mc='梨子'</code>	1	<code>i=2</code>	<code>i=2+1=3</code>
	<code>mc='桃子'</code>	3	<code>i=3</code>	<code>i=3+3=6</code>
	<code>mc='香蕉'</code>	6	<code>i=6</code>	<code>i=6+6=12</code>

```

1  ## 記得要用變數賦值
2  total = 0
3
4  ## 第一回合
5  fruit = "蘋果"; number = 2
6  total = total + number    # 數學的加法
7  print("摘了", fruit, number,"個", "總共", total, "個水果。")
8
9  ## 第二回合
10 fruit = "梨子"; number = 1
11 total += number          # 數學的累加
12 print(f"摘了 {fruit} {number}個 總共{total}個水果。")
13
14 ## 第三回合
15 fruit = "桃子" ; number = 3
16 total = total + number    # 數學的累加
17 print("摘了", fruit, number,"個", "總共", total, "個水果。")
18
19 ## 第四回合
20 fruit = "香蕉" ; number = 6
21 total += number          # 數學的累加
22 print("摘了", fruit, number,"個", "總共", total, "個水果。")

```

## 5-3. 文字資料的運算

### 5-3-1 文字運算子

#### 串接字串

使用「+」串接字串，可以將兩個字串合併成一個字串。

- 「土豆」是「花生」還是「馬鈴薯」？
  - 台灣是「花生」、大陸是「馬鈴薯」。
- 『+』是「相加」還是「串接」？
  - 數字是「相加」、文字是「串接」。

## 複製字串

使用「\*」複製字串，執行「字串2」會複製字串一份串接原來字串的後面，執行「字串3」會複製字串兩份串接原來字串的後面，依此類推。

### 範例：文字運算子的運算

```

1  # 文字運算子
2  "123" + "456"    # 字串連接
3  "123" * 3        # 複製字串
4
5  s1 = "123"
6  s2 = "456"
7  s3 = s1 + s2
8  s4 = s1 * 2

```

### 【隨堂練習7】：真心話大冒險

玩家回答電腦提出的問題，然後用程式把玩家的回答連接起來，並且重複三次輸出。

```

1  #|echo: true
2  #|eval: false
3
4  str1 = input("你生命中最重要的是： ")
5  str2 = input("你想對它說的是： ")
6  (str1 + str2 + "!") * 3
7  =====
8  Out[1]: '程式語言我們相見恨晚!!程式語言我們相見恨晚!!程式語言我們相見恨晚!!'

```


串接多行，使用「\」串接多行。

若在python 中同一行的程式碼過長，在該行最後使用「\」當最後一個字元，就可以寫到下一行，這兩行會被視為同一行。

### 文字的特殊列印，加「\」符號

在處理文字時，有一些特殊的字元無法顯示於螢幕，要變成轉義序列才行。

- 列印時，要在特殊字元前加上反斜線「\」符號。
- 進行特殊的控制，形成跳脫字元(escape character)。

 表2-1 轉義序列

轉義序列	功能說明
\n	換行
\t	跳八格
\\	輸出反斜線
\"	輸出雙引號
\'	輸出單引號

範例：文字的特殊列印

```
1 # 螢幕列印：print函式之轉義
2 print("Bigflower Francis")
3 print("Bigflower \n Francis")    # 跳行
4 print("Bigflower \t Francis")    # 4個空格tab
5
6 ## 特殊符號
7 print("顯示反斜線：\\")
8 print("顯示單引號：\'")
9 print("顯示雙引號：\"")
```

加分題: 猜猜看，印出的結果是什麼（請同學告訴我答案）

```
1 # 成雙成對的雙引號
2 print("Bigflower " Francis")
3 print("Bigflower \" Francis")
4 print("Bigflower ' Francis")
5
6 # 成雙成對的單引號
7 print('Bigflower ' Francis')
8 print('Bigflower " Francis')
9 print('Bigflower \' Francis')
```

### 【隨堂練習8】中文字串的（特殊）列印

請印出下面的詩

春眠不覺曉，處處聞啼鳥。

夜來風雨聲，花落知多少。

作者"孟浩然" 詩名"春曉"

```
1 #|echo: true
2 #|eval: false
```



```

3
4 s = '春眠不覺曉，處處聞啼鳥。\\n\\
5 夜來風雨聲，花落知多少。\\n\\
6 \\t作者"孟浩然" 詩名"春曉"'

```

## 5-3-2 索引與切片

### 1. 索引(index)表示物件的位置。

索引告訴我們物件在哪裡。如果要存取某物件時，可以使用中括號[]，並在裡面指出是那個位置包含了我們想要的物件。即[位置索引]。

- 正數索引值：由左到右索引，從0開始。
- 負數索引值：由右到左索引，從-1開始。

#### 範例：索引 - 挑選單一元素

0	1	2	3	4	5	6	7	8	9	10	11
P	y	t	h	o	n		基	礎	必	修	課
-12	-11	-10	-9	-8	-7	-6	-5	-4	-3	-2	-1

```

1  ## 文字索引
2  s = "Python 基礎必修課"
3
4  print(s[0])
5  print(s[1])
6  print(s[2])
7
8  print(s[-1])
9  print(s[-2])
10 print(s[-3])

```

### 2. 切片(slice)，挑選特定範圍物件所採取的方法。

索引告訴我們物件在哪裡，如果要挑選特定範圍的物件時，可以使用中括號[]，並在裡面透過一個冒號(:)字元，將分割範圍(slicing range)標明。

[起始索引：結束索引]。

- 起始索引(start)，切片從這個索引開始。
  - 沒有起始索引值，切片是從頭開始。
- 冒號：告訴電腦，我們正在切片。
- 結束索引(stop)，切片到這個索引結束。
  - 沒有結束索引值，切片是到尾結束。

[起始索引：結束索引：間隔索引]

- 間隔索引(step)，是差值。

#### 範例: 切片-挑選多個元素

0	1	2	3	4	5	6	7	8	9	10	11
P	y	t	h	o	n		基	礎	必	修	課
-12	-11	-10	-9	-8	-7	-6	-5	-4	-3	-2	-1
輸出結果						說明					
s[:]	Python 基礎必修課					字串元素從頭擷取到尾					
s[7:]	基礎必修課					字串由註標值 7 擷取到字串結尾					
s[:6]	Python					字串元素從頭擷取到註標值 5					
s[5:8]	n 基					字串元素從註標值 5 擷取到 7					
s[9:6:-1]	必礎基					字串元素從註標值 9 逆向擷取到 7					
s[::-1]	課修必礎基 nohtyP					字串從尾到頭逆向擷取					
s[::-2]	課必基 nhy					字串間隔 1 個元素逆向擷取					

#### 【加分題】: 字串切片- 秘密約定

小方收到小青發來的一段文字，和一組無規律數字(209, 1-3, 20, 102, 201, 181-183, 31, 195, 210, 221)

小方這是我給你寫的一封密信，其中包含著一個我和你的神秘約定，就看你能不能解讀出來了，下午放學後可以仔細研究一下，呵呵。提示一下，你用我給你的一組數字，依次提取這段文字中對應文字，串連起來就差不多了，我想得周到吧。這個約定到底和什麼事情有關呢？我可不喜歡你用笨方法一個一個地數著文字去找，我希望能用你學習的Python語言，編制一個小程序來解讀出我們之間的秘密。我們到底是不是來電閨蜜，就看電腦水平了。末了說一句：我的小小影子，記得要不見不散呦！

```

1 letter = "小方這是我給你寫的一封密信，其中包含著一個我和你的神秘約定，就看你能不能解讀出來了，下午放
2
3 print(letter[209] + letter[1:3] + letter[20] + letter[102] + letter[201] + letter[181:183] + letter[31] + letter[195] + letter[210] + letter[221])

```

=> 小方這個周末我們看電影！