Chapter 3：物件的指派與運算

本章重點

將介紹程式重要的物件概念，程式的所有東西與世間萬物一樣，每一樣都是物件。以物件的概念為基礎，才能真正了解程式碼的意義。

準備材料

可上網的電腦

學習目標

1. 如何建立 Python 物件。
2. 將物件指派給變數。
3. 了解變數的命名與初始化。
4. 了解物件型別 (又稱之為資料型別) 的重要性。
5. 認識算術運算式、邏輯運算式。

3-1. 程式的物件

生活中充滿許多物件 (object)，不同物件有不同的功能和特性：

- 簡單物件。有些簡單物件有其特定的操作方法，例如：程式中的 "python" 這個字串。有些物件沒有操作方法，例如：整數 9。
- 複雜物件。複雜物件有其特定的操作方法，例如：生活中的『 掃地機器人 』，有掃地、吸塵、拖地等多種功能。

物件有名稱較容易溝通與理解：

- 使用好名稱。例如：『 會掃地的機器設備 』稱為掃地機器人。
- 同一類的物件不同的命名。例如：有 2 顆蘋果，1 顆叫 Alice，1 顆叫 Bob。
- 物件有明顯的特徵或功能。例如：電視、手機、冰箱等。

TIP	多個物件可以組成新的物件。例如：把可可加入牛奶中，可以稱為為 Cocoamilk。

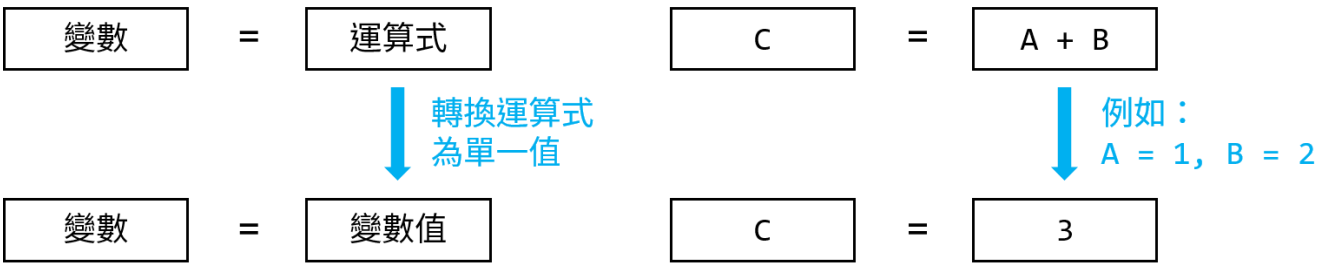
3-1-1. 物件的命名

程式用來描述事件發生的詳細情境，程式中每一個東西都是物件。

- 在程式語言中，可使用變數 (variable) 來幫物件命名。

數學 vs. 程式設計

- 數學，等號左邊和右邊相等。例如：2 * x = 3 * y (程式無法理解這種數學恆等式)。
- 在程式語言裡，等號的意思是指派或指定，有等號的稱之為指派敘述。



TIP	Python 中，『 = 』是指派，『 == 』比較相等的運算。

在程式裡，等號左邊只能是變數名稱，右邊是運算式或變數值，也都是物件。

電腦如何理解程式？

- 必須提供電腦可理解的敘述及對應的步驟，電腦才能進行明確的動作。

```
>>> a = 3
>>> b = 2
>>> a = b + x
Traceback (most recent call last):
  File "<ipython-input-5-1d7f9c34d141>", line 1, in <module>
    a = b + x
NameError: name 'x' is not defined
```

```
>>> a = 3
>>> b = 2
>>> x = a - b
>>> x
1
```

注意：若在 IPython 中，指派敘述不會顯示結果，必須自己呼叫變數來檢查變數值。

3-1-2. 變數：物件的名牌

物件的屬性與方法：

- 每一個物件都有對應的資料屬性 (attribute) 和與它互動的操作方法 (method)。
- 例如：你可以定義『 腳踏車 』物件。
  - ✓ 長、寬、高、顏色、幾個輪子，稱為物件的資料屬性。
  - ✓ 讓人騎乘、沒停好會倒下、改變腳踏車的顏色，這些互動方式稱作物件的操作方法。

例如：'summer' 這個文字在 Python 是字串物件。 Python 內建 (built-in) 有一個字串物件的操作方法，可以將每一個字元轉成大寫。

```
>>> 'summer'.upper()
SUMMER
```

Python 的物件包含了：

- 屬性 (attribute)：從屬性中可以得知該物件相關的值、資料、特性。
- 方法 (method)：該種物件專屬的操作 (operations) ，可對物件執行特定的動作。

TIP	目前，我們只介紹了字串物件的方法，簡單的物件沒有屬性，例如：字串與整數。

3-1-3. 物件名稱

所有物件都可給予一個名稱，這物件名稱就是變數。

- 例如：將整數 1 指派給變數 a。正確是說以變數 a 做為整數物件 1 的名稱。

```
>>> a = 1
>>> a
1
```

- 例如：將字串 "hello" 指派給變數 greeting。正確是說以變數 greeting 做為字串物件 "hello" 的名稱，再呼叫 upper() 的方法將字串轉成大寫。

```
>>> greeting = 'hello'
>>> greeting.upper()
HELLO
```

3-2. 變數的指派與命名規則

用等號 = 來連結變數名稱和對應的物件，這個連結動作叫指派或指定 (assignment)。等號的左邊為變數名稱，右邊為物件。

- 右邊的物件可以是單一物件，也可以是一個運算式 (運算式的結果一定能為一個物件值)。
- 例如：1 和 2 都是物件，1 + 2 為一個運算式物件，最後的值為 3，3 也是物件。

```
>>> a = 1 + 2
>>> a
3
```

TIP 變數只是物件的名稱，物件才是主角，變數代表了一個有資料型別的物件。

3-2-1. 變數的指派

Python 變數不需宣告就可使用，語法為：

```
變數名稱 = 變數值 (某一個物件)
```

例如：變數的值為整數 80。

```
>>> score = 80    # score 的資料型別為整數 (int)
```

使用變數時不必指定資料型別，Python 會根據變數值設定資料型別，例如：上述 score 的資料型別為整數 (int)。又例如：下面的 fruit 的型別為字串 (str)。

```
>>> fruit = '香蕉'    # fruit 的資料型別為字串 (str)
```

可以一起指定多個變數，具有相同的值，例如：

```
>>> a = b = c = 20
```

也可以在同一列指定多個變數，以『 , 』分隔，例如：

```
>>> age, name = 18, '王小明'
```

若變數不再使用，刪除變數可以節省記憶體，語法為：

```
del 變數名稱
```

例如：刪除變數 score。

```
>>> del score
```

3-2-2. 變數的命名規則

Python 的變數命名必須遵守，否則程式會發生錯誤。

- 變數名稱必須以字母（a-z 或是 A-Z）或是底線（_）做開頭，不能是數字。
- 從第二個字開始，可以是字母、數字或是底線。
- 英文大小寫視為不同的變數名稱，例如：apple 與 Apple 是不同的變數名稱。
- 不能是程式保留字，例如：

acos	and	array	asin	assert	atan
break	class	close	continue	cos	Data
def	del	e	elif	else	except
exec	exp	fabs	float	finally	floor
for	from	global	if	import	in
input	int	is	lambda	log	log10
not	open	or	pass	pi	print
raise	range	return	sin	sqrt	tan
try	type	while	write	zeros	

結論

本章重點：

1. 如何建立（初始化）變數。
2. 變數名稱有基本規則，不能任意命名。
3. 每一個物件都有對應的屬性（attribute），有些簡單物件則沒有屬性。
4. 物件有對應的操作方法（method），這些操作方法可控制物件執行相對應的動作。
5. 使用等號（=）來連結變數名稱和對應的物件主體。

3-3. 資料型別

3-3-1. 數值、布林與字串型別

整數、浮點數

整數 (int) 與浮點數 (float) 是數值資料。整數不包含小數點，浮點數包含小數點。例如：

```
>>> num1 = 34    # 整數
>>> num2 = 67.83    # 浮點數
```

若整數要指定為浮點數型別，可加上小數點符號。例如：

```
>>> num3 = 34.0    # 浮點數
```

布林值

布林值 (bool) 用來表示邏輯，內容為 **True** 及 **False** (注意『 T 』與『 F 』是大寫)，通常在條件運算中使用，程式可根據值來判斷要進行何種運算。例如：

```
>>> flag = True    # 布林值
```

字串

字串 (str) 是以一對雙引號 (") 或單引號 (') 包含起來的內容。例如：

```
>>> str1 = '這是字串'
```

若字串要包含引號本身 (雙引號或單引號)，可使用另一種引號包住字串。例如：

```
>>> str2 = '小明說："你好！ "'    # 變數值為『 小明說："你好！ " 』
```

若字串需要特殊字元（引號）或格式（如 Tab、換行等），可在字串中使用脫逸字元：

拖曳字元	意義	拖曳字元	意義
\'	單引號『 ' 』	\"	雙引號『 " 』
\\	反斜線『 \ 』	\n	換行
\r	游標移到列首	\t	Tab 鍵
\v	垂直定位	\a	嗶聲
\b	移除前字元	\f	換頁
\x	以 16 進位表示字元	\o	以 8 進位表示字元

*** 最常用的脫逸字元為：\n，其餘幾乎很少用到的。

例如：

```
>>> str3 = '大家好！ \n 歡迎光臨！'    # 『 歡迎光臨！ 』會顯示於第二列
```

type() 會取得物件的資料型別，語法為：

```
type(物件)
```

例如：

```
>>> type(56)    # int
>>> type("How are you?")    # str
>>> type(True)    # bool
```

NoneType 型別

『 沒有值的物件 』也是物件，以 None 表示，型別是 NoneType。

例如：你參加考試，寫完交卷後，卻莫名考卷遺失了，此時老師既不能記你零分，也不能說你沒來考試，只能說你這次考試沒有分數。

None 型別通常出現在使用函式（function），用來表示函式沒有傳回值（return）。

```
>>> type(None)    # NoneType
```

3-3-2. 資料型別轉換

物件的資料型別非常重要，相同的資料型別才能運算。

資料型別自動轉換

Python 具有簡單的自動轉換功能：若是整數與浮點數運算，會先將整數轉換為浮點數再運算，運算結果為浮點數。例如：

```
>>> num1 = 5 + 7.8    # 結果為浮點數 12.8
```

若是數值與布林值運算，布林值會先轉換為數值，True 轉換為 1，False 轉換為 0。例如：

```
>>> num2 = 5 + True    # 結果為整數 6
```

資料型別轉換函數

若系統無法自動傳換，就需要以型別轉換函數強制轉換。常見的有：

- int()：強制轉換為整數。
- float()：強制轉換為浮點數。
- str()：強制轉換為字串。

例如：整數與字串作 + 運算會產生錯誤。

```
>>> num3 = 23 + '67'    # 錯誤，型別不同無法運算
```

將字串轉換為整數，就可以正常：

```
>>> num3 = 23 + int('67')    # 正確，結果為 90
```

以 print() 列印字串時，若將字串和數值組合會產生錯誤：

```
>>> score = 60
>>> print('小明的成績為 ' + score)    # 錯誤，數值無法自動轉換為字串
```

將數值轉換為字串，再進行組合，就可正常執行：

```
>>> score = 60
>>> print('我的成績為 ' + str(score))    # 正確，結果為『 我的成績為 60 』
```

結論

本章重點：

1. Python 有 5 種基本物件，是整數 (int)、浮點數 (float)、布林 (bool)、字串 (str)、NoneType。
2. 物件運算時要注意型別，必要時需要型別轉換。
3. 可用 type() 函式確認物件的型別。

3-4. 輸出與輸入函式

3-4-1. print()：列印輸出內容

print() 能在螢幕上列印指定的物件。語法為：

```
print(物件 1 [, 物件 2, ……, sep = '分隔字元', end = '結束字元'])
```

- 物件 1, 物件 2, ……：print() 可以一次列印多個物件資料，物件之間以逗點『 , 』分開。
- sep：分隔字元，如果列印多個物件，物件之間以分隔符號區隔，預設值為一個空白字元『 " " 』。
- end：結束字元，列印完畢後自動加入的字元，預設值為換列字元『 "\n" 』，所以下一次執行 print() 會列印在下一列。

例如：

```
print('多吃水果')
print(100, '多吃水果', 60)
print(100, '多吃水果', 60, sep = '&')
print(100, '多吃水果', 60, sep = '&', end = '')
print('多喝水')
```

```
多吃水果
100 多吃水果 60
100&多吃水果&60
100&多吃水果&60 多喝水
```

程式設計師的思考

學會使用 help(函式名稱) 來查詢各種函式的用法，例如：help(print) 可查詢 print() 的詳細的參數設定。若是查詢模組內的函式，例如：sleep()，就需要先匯入 time 模組，語法為 help(sleep) 或 help(time.sleep)，參見 Chapter 2『 匯入模組 』。

參數格式化：%

若要以特定的格式來輸出文字或數值，可用 print() 搭配字串的 % 參數格式化，即以『 %s 』代表字串、『 %d 』代表整數、『 %f 』代表浮點數。語法為：

```
print('欲顯示的文字含 %型別參數' % 資料物件或變數)
```

以 % 的參數格式化方式列印字串與整數。若有多個資料，則可用小括號包起來的 tuple，例如：(name, score)。(tuple 是一種容器物件，容器用來擺放多個物件)

```
name = '王小明'
score = 80
print('%s 的成績為 %d' % (name, score))
```

```
王小明 的成績為 80
```

% 的參數格式化方式可以精確控制列印位置，讓輸出的資料整齊排列，例如：

- %5d：固定列印 5 個字元，若整數少於 5 位數，會在數字左方填入空白字元 (若大於 5 位數，則全部列印。
- %5s：固定列印 5 個字元，若字串少於 5 個字元，會在字串左方填入空白字元 (若大於 5 個字元，則全部列印。
- %8.2f：固定列印 8 個字元 (含小數點)，小數固定列印 2 位數。若整數少於 5 位數 (8 – 3 = 5)，會在數字左方填入空白字元；若小數少於 2 位數，會在數字右方填入『 0 』。

例如：

```
price = 23.8
print('價格為%8.2f' % price)    # 整數 23 的左方有 3 個空白字元
```

```
價格為    23.80
```

參數格式化：format()

也可以使用字串的 format() 來做格式化，以一對大括號表示參數的位置。語法為：

```
print('欲顯示的文字含 {}'.format(資料物件或變數))
```

例如：以字串的 format() 方法列印字串及整數。

```
name = '王小明'
score = 80
print('{} 的成績為 {}'.format(name, score))
```

```
王小明 的成績為 80
```

- 注意：第一對大括號代表 name 變數，第二對大括號代表 score 變數。

範例：格式化列印

以 print() 函式搭配 % 參數格式化來列印成績單。

3-1 座號佔 2 個字元，姓名、國文、數學、英文都佔 3 個字元。

```
print('姓名   座號  國文  數學  英文')
print('%3s  %2d   %3d   %3d  %3d' % ('林大明', 1, 100, 87, 79))
print('%3s  %2d   %3d   %3d  %3d' % ('陳阿中', 2, 74, 88, 100))
print('%3s  %2d   %3d   %3d  %3d' % ('張小英', 11, 82, 65, 8))
```

```
姓名   座號  國文  數學  英文
林大明   1   100    87   79
陳阿中   2    74    88  100
張小英  11    82    65    8
```

3-4-2. input()：輸入資料

input() 是讓使用者輸入資料。例如：要利用電腦幫忙計算成績，則需先由鍵盤輸入學生成績，並且需要宣告一個變數來暫存資料。語法為：

```
變數 = input(['欲提示的文字'])
```

使用者輸入的資料需要儲存於變數中。例如：讓使用者輸入數學成績，再列印成績。

```
score = input('請輸入數學成績：')    # 輸入完成，需按下 Enter 鍵
print(score)
```

```
請輸入數學成績：86 (輸入完按下 Enter)
86
```

- 注意：由 input() 取得的是字串型別。若變數以後需要計算，則要注意變數的型別。

將輸入的資料轉換型別

input() 取得的資料是字串型別，例如：輸入 5，程式會將此輸入存成字串 '5'。**3-2** 會請你輸入一個整數，然後程式會顯示其平方值。

3-2 輸入一個整數，並顯示其平方值。

```
user_input = input('Enter a integer to find the square of: ')
num = int(user_input)
print(num * num)
```

3-2 一開始的兩行，可以合併成一行，例如：**3-3**。

3-3 輸入一個整數，顯示其平方值（使用 input() 時，同時進行型別轉換）。

```
num = int(input('Enter a number to find the square of: '))
print(num * num)
```

TIP int() 只能轉換原始字串為整數的文字，例如：int('2')。若輸入 'a' 或 '2.1'，則程式會產生錯誤。

要求輸入更多資料

程式可以輸入多項資料。**3-4** 要求輸入 2 個數值，然後顯示這 2 個數值與相乘後的結果。例如：輸入 4.1 和 2.2，程式會顯示 4.1 * 2.2 = 9.02。

3-4 輸入兩個浮點數，顯示其相乘的結果。

```
num1 = float(input('Enter a number: '))
num2 = float(input('Enter another number: '))
print(num1, '*', num2, '=', num1 * num2)
```

- 注意：在 print() 內使用逗點『 , 』分隔物件，則不需考慮物件的型別。

結論

本章重點：

1. 學會使用 print() 帶入多個物件的方法，並顯示結果。
2. 每次使用 input() 時，程式會顯示提示文字並處於等待的輸入的狀態，輸入完成按下 Enter 表示輸入完成。
3. 可以將輸入的資訊轉換成其他適當的型別以便後續處理。

練習 3-4

1. 輸入姓名和年齡，並使用變數分別儲存這兩項資訊，並計算 25 年後的年紀為何？例如：輸入姓名為 Bob，年齡為 10，則程式可顯示 'Hi Bob! In 25 years you will be 35!'。

3-5. 運算式

數學都是由『 1 + 1 = 2 』開始，『 1 + 1 』就是典型的運算式。其中用來指定資料做哪一種運算的是『 運算子 』，進行運算的資料稱為『 運算元 』。例如：『 2 + 3 』中，『 + 』是運算子，『 2 』與『 3 』是運算元。

運算子依據運算元的個數，分為單元運算子與二元運算子：

- 單元運算子：只有一個運算元。例如：『 -100 』中的『 - 』(負)、『 not x 』中的『 not 』(取布林值 x 的反向)，單元運算子位於運算元的前方。
- 二元運算子：具有兩個運算元。例如：『 100 - 30 』中的『 - 』(減)、『 x and y 』中的『 and 』(且的運算) 等，二元運算子位於兩個運算元的中間。

3-5-1. 算數運算子

算數運算子：用於執行一般數學運算，稱之為算數運算式，結果只為整數或浮點數。

運算子	意義	範例	範例結果
+	兩個運算元相加	12 + 3	15
-	兩個運算元相減	12 - 3	9
*	兩個運算元相乘	12 * 3	36
/	兩個運算元相除，取商	32 / 5	6.4
%	取得餘數	32 % 5	2
//	兩個運算元相除，取整數商	32 // 5	6
**	(運算元 1) 的 (運算元 2) 次方	2 ** 3	$2^3 = 8$

注意：『 / 』、『 % 』、『 // 』與除法有關，第二個運算元不能為 0，否則會出現『 ZeroDivisionError 』的錯誤。

3-5-2. 邏輯運算子

邏輯運算子：結合多個結果為布林值的比較運算式 == (等於)、> (大於)、< (小於)、>= (大於等於)、<= (小於等於)，再經過邏輯運算得到最終的比較結果，稱之為邏輯運算式，結果也是布林值。

運算子	意義	範例	範例結果
not	傳回與原來比較結果相反的值，即比較結果是 True，就傳回 False；比較結果是 False，就傳回 True。	not(3 > 5) not(5 > 3)	True False
and	只有兩個運算元的比較結果是 True 時，才傳回 True，其餘情況都傳回 False。	(5 > 3) and (9 > 6) (5 > 3) and (9 < 6) (5 < 3) and (9 > 6) (5 < 3) and (9 < 6)	True False False False
or	只有兩個運算元的比較結果是 False 時，才傳回 False，其餘情況都傳回 True。	(5 > 3) or (9 > 6) (5 > 3) or (9 < 6) (5 < 3) or (9 > 6) (5 < 3) or (9 < 6)	True True True False

3-5-3. 複合指定運算子

在程式中，某些變數值常需要做某種規律性的改變，例如：在迴圈中需將計數變數做特定增量。一般的做法是將變數值進行運算後在指定給原來的變數，例如：下面將變數 i 的值增加 3。

```
>>> i = i + 3
```

複合指定運算子：可將算術運算子置於『 = 』的前方來取代重複的變數。例如：

```
>>> i += 3    # 即 i = i + 3
>>> i -= 3    # 即 i = i - 3
```

注意：複合指定運算子同時做了『 執行運算 』與『 指派 』兩件工作。

下表是以 i 變數值為 10，來計算的範例結果：

運算子	意義	範例	範例結果
+=	相加後，再指派給原變數	i += 5	15
-=	相減後，再指派給原變數	i -= 5	5
*=	相乘後，再指派給原變數	i *= 5	50
/=	相除後，再指派給原變數	i /= 5	2.0
%=	取得餘數後，再指派給原變數	i %= 5	0
//=	取得整除的商數後，再指派給原變數	i //= 5	2
**=	做指數運算後，再指派給原變數	i **= 5	100000

範例：計算總分及平均成績

3-5 讓使用者輸入三科成績後，計算總分及平均。

```
nat = input('請輸入國文成績： ')
math = input('請輸入數學成績： ')
eng = input('請輸入英文成績： ')
summation = int(nat) + int(math) + int(eng)    # 輸入值需型別轉換為整數
average = summation / 3
print('成績總分: %d，平均成績: %5.2f' % (summation, average))
```

```
請輸入國文成績: 85

請輸入數學成績: 92

請輸入英文成績: 82
成績總分: 259，平均成績: 86.33
```

結論

本章重點：

1. 程式的運算有兩種，算數運算式與邏輯運算式 (邏輯運算式用於條件判斷，參見 Chapter 6)。
2. 對不同型別的數值物件進行數學運算，要留意其運算結果可能是整數或浮點數。

練習 3-5

1. 編寫程式要求輸入兩個數字，將這兩個數字分別儲存於變數 b 與變數 e，並顯示 b^e 的運算結果。