

變數、運算式與輸入輸出

滄海《Python 程式設計初學指引》二版·CH02

§ 1 名詞速查表

| 中文 | English | 一句話定義 | 例 | 易混 |
|------|-----------------------------------|--|---------------------------------|---|
| 變數 | variable | 可變的具名記憶體空間；像把「名牌」貼到資料物件上 | <code>a = 5</code> | 型別隨值動態決定 |
| 型別 | type | 資料的類別，Python 執行時自動判斷； <code>type()</code> 可查 | <code>int/float/str/bool</code> | 同名變數可換型別 |
| 運算式 | expression | 由運算元 (operand) 與運算子 (operator) 組成的式子 | <code>a + 5</code> | — |
| 指定 | assignment <code>=</code> | 把右邊的值指定給左邊的變數 (由右到左) | <code>x = 3</code> | <code>=</code> 指定 vs <code>==</code> 比較 |
| 保留字 | keyword | Python 固定用途的字，不可當變數名 | <code>if for def</code> | 內建函式名也避免 |
| 整數除法 | floor division <code>//</code> | 除完 向下取整 (往更小的整數) | <code>-3//2 → -2</code> | 與 <code>/</code> (浮點) 不同 |
| 取餘數 | modulo <code>%</code> | 除法的餘數 | <code>7 % 3 → 1</code> | — |
| 型別轉換 | type casting | <code>int()</code> <code>float()</code> <code>str()</code> <code>bool()</code> 等強制轉換 | <code>int('5')</code> | <code>int('a') → ValueError</code> |
| 輸入 | <code>input()</code> | 讀鍵盤輸入， 回傳值永遠是字串 | <code>input('?')</code> | 要數字得 <code>int()</code> 轉 |
| 格式化 | <code>%</code> formatting | <code>%d %f %s</code> 等控制 <code>print</code> 輸出格式 | <code>"%.2f" % x</code> | — |

§ 2 核心概念

核心概念

變數 (variable) 是可變的具名記憶體空間。Python 的變數像一張**名牌**：`a = 5` 是把名牌 `a` 貼到資料物件 `5` 上。`=` 是**指定 (assignment)**，由右到左：先算出右邊的值，再貼給左邊的變數。

Python 是**動態型別**：不用宣告型別，`a = 5` 時 `a` 是 `int`、`a = 5.0` 變 `float`、`a = '5'` 變 `str`，型別隨貼到的物件而變，`type()` 可查。同一個名稱可重複指定、可換型別。

運算式 (expression) 由運算元與運算子組成 (`a + 5`)。要記三件事：① 運算子有**優先序** (`()` > `**` > `*` / `//` > `%` > `+` - > 比較 > `and/or/not`，`=` 最低)；② 兩種除法不同 (`/` 給浮點數、`//` 向下取整)；③ `input()` 回傳的永遠是**字串**，要當數字得用 `int()` / `float()` 轉。

§ 3 主要內容

3.1 一、變數與動態型別 (招牌)

```
a = 5          # a 是 int
a = 5.0       # 重新指定 → a 變 float
a = '5'       # 再指定 → a 變 str
print(type(a)) # <class 'str'>
del a        # 刪除變數，之後再用會 NameError
```

變數 = 名牌；= 把名牌貼到資料物件上 (動態型別)



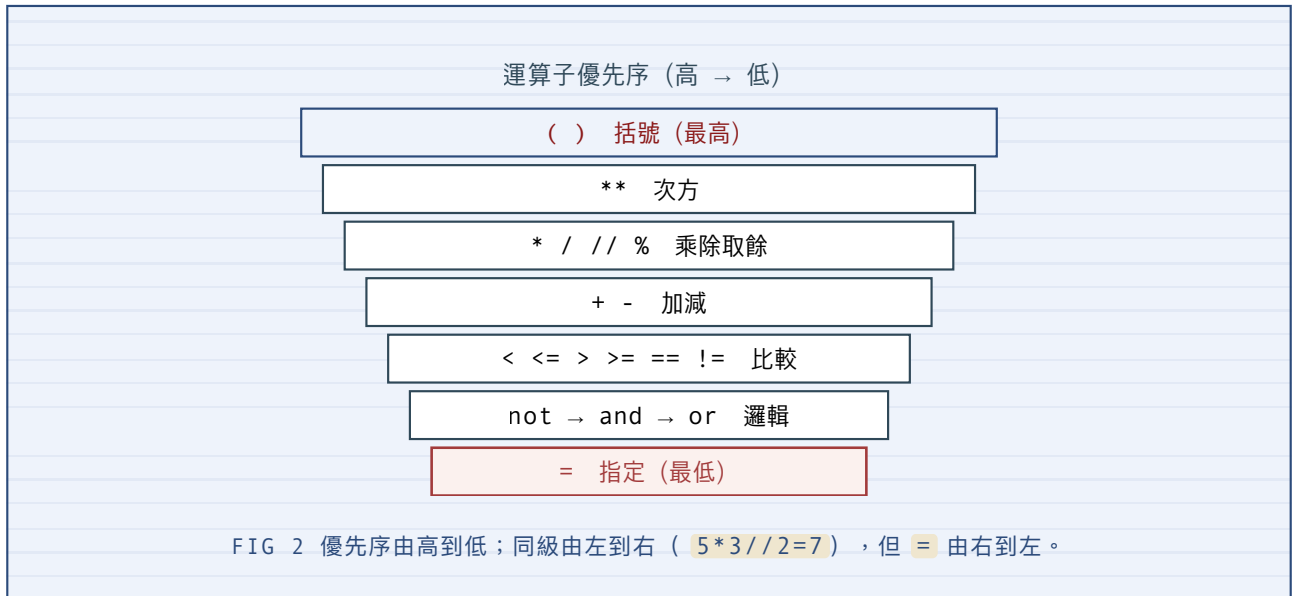
同一個名牌 `a` 可改貼不同型別的物件；`type(a)` 隨之改變

FIG 1 變數是名牌、`=` 是貼名牌；Python 動態型別，型別隨貼到的物件而定。

命名規則：① 字母或底線開頭；② 之後可字母／數字／底線；③ 不可用保留字（`if for while def class import` 等）；④ 區分大小寫、全半形。慣例：有意義的名稱、常數用全大寫提醒自己別改。

3.2 二、運算子與優先序（招牌）

```
7 / 2      # 3.5  一般除法 → 浮點數
7 // 2     # 3    整數除法 → 向下取整
-3 // 2    # -2   向下取整（往更小的整數，不是 -1）
7 % 3      # 1    餘數
2 ** 10    # 1024 次方
5 * 3 // 2 # 7    同優先序由左到右：(5*3)//2
a = b = 10 # 多重指定
a, b = b, a # 交換兩變數（不用暫存變數）
```



數值函式：`abs(a)` 絕對值、`pow(a,b)` 等同 `a**b`、`round()` 四捨五入（**銀行家捨入**，見易錯）。

3.3 三、型別轉換與進位制

```
int('520') # 520 字串→整數（內容須是數字）
int(2.9)   # 2    浮點→整數，小數無條件捨去（不是四捨五入）
float(5)   # 5.0
str(520)   # '520' 數值→字串
bin(246)   # '0b11110110' oct(246)→'0o366' hex(246)→'0xf6'（回傳字串）
```

3.4 四、布林與比較運算

```
True and False # False      or / and / not 三個邏輯運算子
3 < 5          # True
True + True    # 2          布林可當數字：True=1、False=0
int(True)     # 1
'a' in 'cat'   # True      in / not in 判斷存在；is / is not 判斷同物件
```

3.5 五、字串、input 與 print 格式化

```
'Py' + 'thon' # 'Python'    + 串接
'=' * 10      # '===========' * 重複
len('Python') # 6
ord('A'), chr(65) # 65, 'A' 字元↔編碼
print('a', 'b', sep='\n') # sep 設分隔符 (這裡換行)

name = input('你的名字：') # input 回傳「字串」
age = int(input('你的年紀：')) # 要當數字必須 int() 轉
print('%s 今年 %d 歲, BMI=%.2f' % (name, age, 22.86)) # 格式化：%s 字串 %d 整數 %.2f 兩位小數
```

跳脫字元：`\n` 換行、`\t` 定位、`\\` 反斜線、`\'` `\"` 引號。

§ 4 語法與方法速查

```
# 變數與型別
x = 值          type(x)          del x          # 動態型別；del 刪變數
a = b = c = 10  a, b = b, a # 多重指定 / 交換

# 運算子
+ - * /        // [ ](向下取整) % [ ](餘數) ** [ ](次方)
+= -= *= /= //= %= **= # 複合指定
abs(x) pow(a,b) round(x) round(x, 位數)

# 型別轉換
int(x) float(x) str(x) bool(x) # int('a')→ValueError、int(2.9)→2
bin(n) oct(n) hex(n)          # 回傳字串

# 布林 / 比較
and or not      < <= > >= == != is is not in not in

# 字串 / 輸入輸出
s1 + s2      s * n      len(s)      ord(c)      chr(n)
input('提示') # 回字串
print(..., sep=' ', end='\n')
"%d %f %s" % (值, ...) # 格式化；%.2f 兩位、%7d 寬7
```

§ 5 常見錯誤

常見錯誤

- `input()` 當數字直接算 → 它回的是字串，`input()+1` 會 `TypeError` 或字串相加；要 `int(input())`。
- `int('Python')` → `ValueError`；只有內容是數字的字串能轉。
- `int(2.9)` 以為會四捨五入 → 其實無條件捨去得 2；要四捨五入用 `round()`。
- `round()` 不是單純四捨五入（銀行家捨入）：`round(2.5)→2`、`round(0.5)→0`、`round(1.5)→2`（逢 5 進到偶數）。
- `//` 對負數：向下取整往更小取，`-3//2 → -2`（不是 -1）。
- `=` 與 `==` 混用：`=` 是指定、`==` 是比較。
- 除以 0 → `ZeroDivisionError`。
- 用到沒定義／拼錯的變數名 → `NameError`；用保留字或內建函式名當變數會出問題（如把 `print` 當變數名會讓 `print()` 失效）。
- 不同型別硬運算 → `TypeError`（如 `'3' + 5`）。

§ 6 練習題

練習 1 (一般題)：動態型別與 type()

引導步驟

1. 依序 `a = 5`、印 `type(a)`；再 `a = 5.0`、印；再 `a = '5'`、印。
2. 觀察同一名稱型別怎麼變。

| |
|--|
| |
| |
| |
| |
| |
| |
| |
| |
| |
| |

解答

```
a = 5; print(type(a)) # <class 'int'>
a = 5.0; print(type(a)) # <class 'float'>
a = '5'; print(type(a)) # <class 'str'>
```

練習 2 (重要題)：兩種除法與優先序

引導步驟

1. 印 `7/2`、`7//2`、`-3//2`、`7%3`。
2. 印 `5 * 3 // 2` 與 `2 ** 3 ** 2`，預測再驗證。

| |
|--|
| |
| |
| |
| |
| |
| |
| |
| |
| |
| |

解答

```
print(7/2, 7//2, -3//2, 7%3)    # 3.5 3 -2 1
print(5 * 3 // 2)               # 7 ((5*3)//2, 同級左到右)
print(2 ** 3 ** 2)             # 512 (** 由右到左: 2**(3**2)=2**9)
```

練習 3（一般題）：交換兩變數

引導步驟

1. `a, b = 1, 2`。
2. 用 `a, b = b, a` 交換，印出結果（不用暫存變數）。

| |
|--|
| |
| |
| |
| |
| |
| |
| |
| |
| |
| |

解答

```
a, b = 1, 2
a, b = b, a
print(a, b)    # 2 1
```


練習 5（一般題）：格式化輸出

引導步驟

1. 給 `name='Peter'`、`bmi=22.857`。
2. 用 `%` 格式化印「Peter 的 BMI 是 22.86」（兩位小數）。

| |
|--|
| |
| |
| |
| |
| |
| |
| |
| |
| |
| |

解答

```
name = 'Peter'; bmi = 22.857
print('%s 的 BMI 是%.2f' % (name, bmi)) # Peter 的 BMI 是 22.86
```

§ 7 自我檢核

- 能說出變數是「名牌貼物件」、Python 動態型別，並用 `type()` 查型別。
- 知道變數命名規則（字母/底線開頭、不可用保留字、區分大小寫）。
- 能分辨 `/`（浮點）與 `//`（向下取整），並算對負數 `-3//2=-2`。
- 記得運算子優先序（`()>*>*/% >+-`，`=` 最低），同級由左到右。
- 會用複合指定 `+=` 等與 `abs/pow/round`；知道 `round` 是銀行家捨入。
- 能做型別轉換 `int/float/str/bool`，知道 `int(2.9)=2`（捨去）、`int('a')` `ValueError`。
- 會用布林與比較運算（`and/or/not`、`== != is in`）。
- 知道 `input()` 回字串、要 `int()` 轉；會用 `print` 的 `%` 格式化與 `sep`。

□ 認得常見錯誤代碼：NameError／TypeError／ValueError／ZeroDivisionError／SyntaxError。