

# 資料型別

蔡尚融

2018-02-26

# 資料型別

- 所有的資料型態與函數在 Python 3 中，都是一種物件 (Object) ，函數也是一種資料型態。
- 抽象概念上，程式執行時的資料交換或傳遞是透過變數。
- 變數命名 (variable name) ，可以用的子元為 大小寫字母 (A, B, ..., a, b, ...) 、  
數字 (1, 2, ...) 、底線 (\_) ；但第一個字元不可為數字，亦不可為關鍵字 (keyword) 。

# 指派陳述

- Python 用符號 = 表示指派變數與值，不是數學上的相等。

- 指派變數值 (assignment)

```
>>> message = 'This is a text.'
```

```
>>> year = 2017
```

```
>>> sqrt_2 = 1.4142135623730951
```

- Python 3 的關鍵字：

False, None, True, and, as, assert, break, class, continue, def, del, elif, else, except, finally, for, from, global, if, import, in, is, lambda, nonlocal, not, or, pass, raise, return, try, while, with, yield.

# 變數與型態

- 值 (value)：使程式運作的基本資料。  
數值、字串、物件。
- type() 函式，檢查型態。

```
>>> type('hello, world')  
<class 'str'>  
>>> pi = 3.14159  
<class 'float'>  
>>> year = 2017  
>>> type(year)  
<class 'int'>  
>>> type('year')  
<class 'str'>
```

# 陳述與表示式

- 陳述 (statement)：單一的指令、函數呼叫，或指派變數值。

```
>>> import math
>>> sqrt_2 = math.sqrt(2)
>>> year = 2017
>>> print(year)
```

- 表示式 (expression)：結合數值、變數、運算子等，以能求得數值的表達。

```
>>> 2
2
>>> 2 + sqrt_2
3.414213562373095
```

# 算術運算子 (arithmetic operator)

- 算術運算子為二元運算，需要兩個運算元。
- 運算元  $x$  與  $y$  可為整數 (int)、浮點數 (float)。

運算子	用途	語法
+	加	$x + y$
-	減	$x - y$
*	乘	$x * y$
**	指數	$x ** y$
/	除	$x / y$
//	商取整數	$x // y$
%	取餘數	$x \% y$

- 補充：單元運算只要一個運算元，例如轉為負值用  $-$  運算子。

# 運算優先權

- 括號 ( ) 有最高優先權。
- 指數 (exponentiation) ，運算子 \*\*  

```
>>> 1 + 2**3
```

```
9
```
- 乘除先於加減。
- 兩個同優先權運算時，先遇到先處理。

# 字串運算

- 基本上字串是不能運算的；但是運算子 + 與 \* 例外。
- 運算子 + 用於字串連接。

```
>>> str1 = 'Learing '  
>>> str2 = 'Python! '  
>>> str1 + str2  
'Learing Python'
```

- 運算子 \* 用於重複出現。

```
>>> str3 = 'Test!'  
>>> str3 * 4  
'Test!Test!Test!Test!'
```



## 隨堂練習：

1. 變數 `va` 與 `vb` 由指派陳述給值，變數 `vc` 由 `va` 與 `vb` 加法二元運算再指派給值，則此三者的變數型態分別為何。

```
va = 3
```

```
vb = 5.0
```

```
vc = va + vb
```

2. 由兩個給定字串 `str1` 與 `str2` 合成下列結果。

- ▶ 給定的字串：

```
str1 = 'Learing '
```

```
str2 = 'Python! '
```

- ▶ 結果：

```
'Learing Python! Learing Python! Learing Python! '
```