

蟒蛇

Python 3.9

Expression



Python 3.8





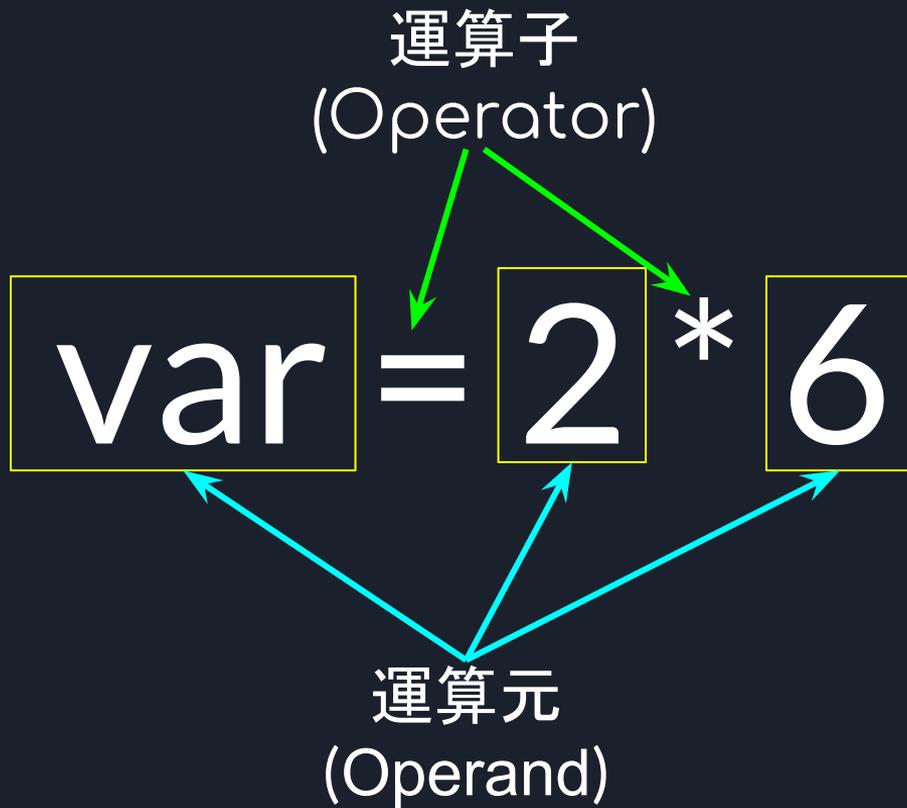
基礎程式設計

Fundamental Programming

Python 3.9 Expression

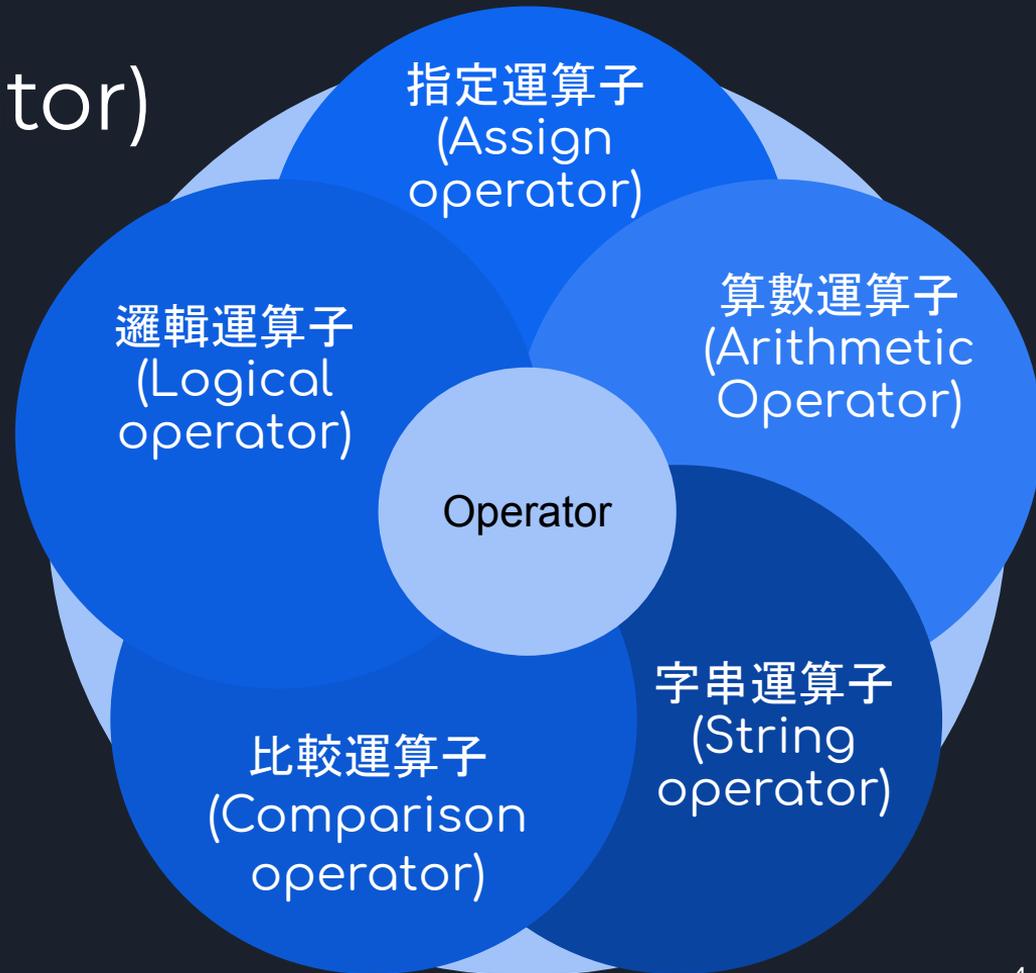
Yung-Chen Chou Ph.D.
iShool, Feng Chia University
Aug. 20, 2021

Introduction



運算子 (Operator)

- 將數值或變數進行運算，需要使用



指定運算子 (Assign Operator)

- 用等號(=)表示,意思是等號右邊先運算,再將運算結果指定給左邊的變數

運算子	範例	結果
+=	a += 2	a = a + 2
-=	a -= 2	a = a - 2
*=	a *= 2	a = a * 2
/=	a /= 2	a = a / 2
**=	a **= 2	a = a ** 2
//=	a //= 2	a = a // 2
%=	a %= 2	a = a % 2

算術運算子 (Arithmetic Operator)

- 在程式中乘號不可以忽略，先乘除後加減，使用小括號括起來的部分優先計算

運算子	說明	範例	結果
+	相加	7+2	9
-	相減	7-2	5
*	相乘	7*2	14
/	相除	7/2	3.5
**	指數	7**2	49
//	整除數	7//2	3
%	除法餘數	7%2	1

算術運算子 (Arithmetic Operator)

運算子	說明	範例	縮寫範例
<code>+=</code>	加	<code>v1 = v1 + 1</code>	<code>v1 += 1</code>
<code>-=</code>	減	<code>v1 = v1 - 1</code>	<code>v1 -= 1</code>
<code>*=</code>	乘	<code>v1 = v1 * 2</code>	<code>v1 *= 2</code>
<code>/=</code>	除	<code>v1 = v1 / 2</code>	<code>v1 /= 2</code>
<code>**=</code>	次方	<code>v1 = v1 ** 2</code>	<code>v1 **= 2</code>
<code>//=</code>	整除數	<code>v1 = v1 // 2</code>	<code>v1 //= 2</code>
<code>%=</code>	除法餘數	<code>v1 = v1 % 5</code>	<code>v1 %= 5</code>

比較運算子 (Comparison operator)

運算子	說明	範例	結果
>	大於	$a > b$	True
<	小於	$a < b$	False
>=	大於或等於	$a >= b$	True
<=	小於或等於	$a <= b$	False
==	等於	$a == 2$	False
!=	不等於	$a != 2$	True

假設 $a = 10, b = 5$

邏輯運算子 (Logical operator)

a	b	a and b	a or b	not a	not b
True	True	True	True	False	False
True	False	False	True	False	True
False	True	False	True	True	False
False	False	False	False	True	True

- 邏輯運算子有三種運算子

- 而且 (**and**)
- 或者 (**or**)
- 非 (**not**)

- 可以使用邏輯運算子連結多個條件

舉例	a 值	結果	說明
$((a > 60) \text{ and } (a < 80))$	70	True	條件 (70 > 60) 為 True , 條件 (70 < 80) 為 True , 經 and (而且) 運算 結果為 True
$((a > 60) \text{ and } (a < 80))$	60	False	條件 (60 > 60) 為 False , 條件 (60 < 80) 為 True , 經 and (而且) 運算 結果為 False
$((a > 60) \text{ or } (a < 80))$	60	True	條件 (60 < 80) 為 True , 經 or (或者) 運算 只要有一個條件為 True , 整體結果就為 True
not (a > 60)	60	True	條件 (60 > 60) 為 False , 經 not (非) 運算 結果變成 True

in 與 is 運算子

- 「`a in b`」用於判斷 `a` 是否為 `b` 其中一個元素，若是則回傳 `True`，否則回傳 `False`
- 「`a is b`」用於判斷兩物件的 `id` 是否相同，若變數 `a` 與變數 `b` 指向相同物件，則回傳 `True`，否則回傳 `False`
- 「`a == b`」只要 `a` 與 `b` 數值相同就回傳 `True`
- 「`a is b`」要 `a` 與 `b` 需要參考到相同物件才回傳 `True`

運算子	說明	舉例	結果
<code>in</code>	是否包含	<code>a = 1</code> <code>b = [1, 2, 3]</code> <code>print(a in b)</code>	<code>True</code>
<code>not in</code>	是否 不包含	<code>a = 1</code> <code>b = [1, 2, 3]</code> <code>print(a not in b)</code>	<code>False</code>
<code>is</code>	是否為 相同物件	<code>a = [1, 2, 3]</code> <code>b = [1, 2, 3]</code> <code>print(a is b)</code>	<code>False</code>
<code>is not</code>	是否 不為 相同物件	<code>a = [1, 2, 3]</code> <code>b = [1, 2, 3]</code> <code>print(a is not b)</code>	<code>True</code>

```
>>> a = [1, 2, 3]
>>> b = [1, 2, 3]
>>> print( id(a) )
140249591464448
>>> print( id(b) )
140249590236224
>>> print( a == b )
True
>>> print( a is b )
False
```

不相同

位元運算子 (Bitwise operator)

- Python先將數值轉換成二進位表示，從右到左將數字的每個位元進行位元運算，獲得最後的結果

運算元1	運算元2	運算元1 & 運算元2	運算元1 運算元2	運算元1 ^ 運算元2	~ 運算元2
1	1	1	1	0	0
1	0	0	1	1	1
0	1	0	1	1	0
0	0	0	0	0	1

運算子	說明
&	位元 and 運算，當兩個都 True才將該位元設定 True, 其他都 False Ex: a = 5 & 4
	位元 or 運算，當兩個都 False 才將該位元設定為 False, 其他情況都 True Ex:
^	位元 互斥或運算，當一個為 True 同時另外一個為 False時才將該位元設定為 True, 其他都設定為 False

Example:

0 1 1 0 \Leftarrow 6

1 1 0 1 \Leftarrow 13

& 0 1 0 0 \Leftarrow 4

Example:

0 1 1 0 \Leftarrow 6

1 1 0 1 \Leftarrow 13

| 1 1 1 1 \Leftarrow 15

Example: XOR

0 1 1 0 \Leftarrow 6

1 1 0 1 \Leftarrow 13

^ 1 0 1 1 \Leftarrow 11

位元運算子 (Bitwise operator)

運算元	語法	說明
~ (取位元反相)	$a = \sim 5$	5的二進制 0101, 則 ~ (0101)會得到 1010, 開頭的1表示為負數, 所以得到的結果為 $a = -6$
<< (左移)	運算元1 << 運算元2	將運算元1 依運算元2 指定的位元數向左移動, 右邊補 0
>> (右移)	運算元1 >> 運算元2	將運算元1 依運算元2 指定的位元數向右移動, 左邊補 0

Example: $6 \ll 2$

0 0 0 1 1 0 \leftarrow 6

0 1 1 0 0 0 \leftarrow 6 $\ll 2$ 後的結果為 24

乘法

Example: $6 \gg 2$

0 0 0 1 1 0 \leftarrow 6

0 0 0 0 0 1 \leftarrow 6 $\gg 2$ 後的結果為 1

除法

運算子優先順序

1 (高)	()
2	a()
3	**
4	正負號+ -
5	* / // %
6	+ -
7	> < >= <= == !=
8	not
9	and
10 (低)	or