

變數與運算式

Chih-Chung Hsu (許志仲)
Institute of Data Science
National Cheng Kung University
<https://cchsu.info>



```
= 22  
int(X)  
int(type(X))  
= "This is a string"  
int(X)  
int(type(X))
```

int

on1\Scripts\python

s\PyCharm Community Edi

```
sys; print('Python  
h.extend(['C:\\py
```

platform))

onsole: starting.

Value and the Data Type
changes during the code
Execution

3.7.3 (v3.7.3:ef4

[MSC v.1916 64 bit (AMD

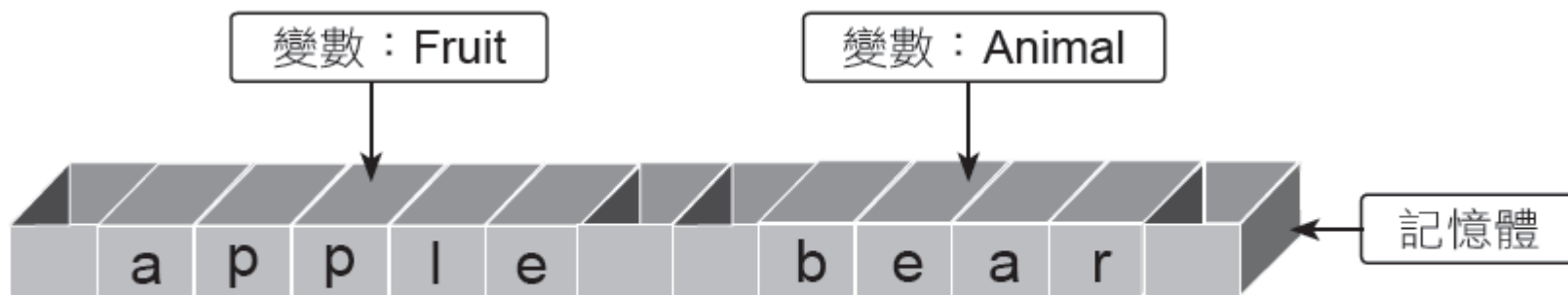
```
file('C:/python/t
```

```
'int'  
a string  
'str'
```

變數

■ 認識變數

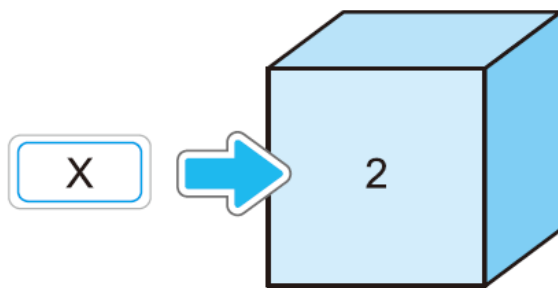
- 當資料儲存於記憶體時，電腦會記住該記憶體的位置，以便要使用時才可以取出。但電腦的地址是一個複雜且隨機的數字，例如「65438790」，程式設計者怎麼可能會記得此地址呢？更何況有很多地址要記憶。解決的方法是給予這些地址一個有意義的名稱，取代無意義的數字地址，就可輕鬆取得電腦中的資料了！這些取代數字地址的名稱就是「變數」。



▲ 以變數取代記憶體地址

變數

- 何謂變數
- 在Python中任何整數、浮點數、字串、變數與函式都是物件，Python的變數可以想像為標籤，可以使用「=」將標籤貼到物件上



- 那我怎麼知道變數存在記憶體哪邊的精確位置？
 - id(variable)

變數

- 變數是將標籤貼在物件上，就指向那個物件，程式在運算過程中，對資料進行處理與運算，就是對變數進行處理與運算，就是對變數所對應的物件進行處理與運算

變數

- 程式中命名變數的方式通常有固定的規則，如成績就用score 表示，加總就用sum 等，再對變數進行運算

sum = score1 + score2

加總就用sum表示

成績就用score表示

基本範例 (變數存取)

1	<code>print(a)</code>
---	-----------------------

- 程式碼解說
 - 第1 行：印出變數a 所參考的物件內容

變數執行結果

- 執行結果

發生 NameError 錯誤。

Traceback (most recent call last):

File "H:/teach/python/ch2-vars-operator/var1.py", line 1, in

<module>

print(a)

NameError: name 'a' is not defined

- 執行結果說明

- 印出變數a之前，變數a並未參考到任何物件，出現NameError表示變數a未定義

變數範例-2

1	a=1
2	print(a, id(a))
3	a='Python'
4	print(a, id(a))

1 500586224

Python 3136064

範例執行結果

- 執行結果說明
 - 變數a 參考整數物件「1」時，id(a) 為500586224，
 - a 參考字串物件「Python」時，id(a) 為3136064，
 - 表示變數a 經由「=」參考到不同的物件，執行時決定變數參考到不同物件，變數不需要事先宣告資料型別就可以使用

範例-3

1	<code>x = 1</code>
2	<code>y = x</code>
3	<code>print(id(x),id(y))</code>
4	<code>print(x,y)</code>

■ 執行結果

493311728 493311728

1 1

範例-3執行結果

- 執行結果說明
 - 變數x 參考整數物件「1」時，id(x) 為493311728，
 - 變數y 參考到變數x 所參考物件時，id(y) 為493311728，
 - 表示變數y 經由「=」參考到變數x 所參考的物件，變數x 與變數y 參考到相同物件

那麼，具體來說，如何設定變數？

■ 建立變數

■ 新增變數

- Python 變數不需宣告就可以使用，語法為：`變數名稱 = 變數值`
- 如果多個變數具有相同變數值，可以一起指定變數值，例如變數 a、b、c 的值皆為 20，其宣告方式為：
`a = b = c = 20`
- 也可以在同一列中指定多個變數，「變數」之間以「,」分隔，「值」之間也以「,」分隔。例如變數 age 的值為 18，name 的值為「林大山」：

`age, name = 18, "林大山"`

■ 刪除變數

- 如果變數不再使用，可以將變數刪除以節省記憶體。刪除變數的語法為：

`del 變數名稱`

變數命名規則

- 第一個字母必須是大小寫字母、_、中文。
- 大小寫字母、數字、_、中文組成變數名稱。
 - 英文字母大小寫視為不同變數名稱。
- 變數名稱不能與 Python 內建的保留字相同。Python 的保留字有：

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	

Comment 註解

■ 註解

■ 單行註解

- Python可在程式碼中加入「#」做為單行註解，使用方式有兩種：第一種是位於程式列起始處，該行程式都不會執行，例如：

```
#fruit變數為最喜歡吃的水果名稱
```

```
fruit = "香蕉"
```

■ 多行註解

- 在註解的區塊前後加入三個單引號 (') 或三個雙引號 (""") 作為多行註解。許多程式設計者會使用多行註解來說明程式用途、作者等，例如：

```
"""
```

```
本程式可計算使用者 BMI 值提供使用者參考。
```

```
使用者輸入身高及體重後會顯示 BMI 值。
```

```
設計者：文淵閣工作室
```

```
"""
```

資料型態

■ 數值型態

- Python 數值資料型態有整數 (int)、浮點數 (float) 及布林值 (bool)。
 - 整數：int，是指不含小數點的數值。
 - 浮點數：float，是指包含小數點的數值。
 - 布林值：bool，此種資料型態只有兩個值：True 及 False (注意「T」及「F」是大寫)。此種變數通常是在條件運算中使用，程式可根據布林變數的值判斷要進行何種運作。

字串型態

- Python 字串資料型態 (str) 是變數值以一對雙引號 (「"」) 或單引號 (「'」) 包
- 起來，例如：`str1 = "這是字串 "`
- 如果字串要包含引號 (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！"'    #str2變數值為'小明說："你好！"'
```

```
str3 = "小華說：'早安！'"    #str3變數值為「小華說：'早安！'」
```

Escape char

- 若字串需含有特殊字元如 Tab、換行等，可在字串中使用脫逸字元：脫逸字元是以「\」為開頭，後面跟著一定格式的字元代表特定意義的特殊字元。下表為Python 常用的脫逸字元：

脫逸字元	意義	脫逸字元	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	響鈴
\b	後退鍵 (BackSpace)	\f	換頁
\x	以十六進位表示字元	\o	以八進位表示字元

取得資料的型別

- type 命令

- type 命令會取得項目的資料型態，如果使用者不確定某些項目的資料型態，可用 type 命令確認，語法為： `type(項目)`

- 例如：

```
print(type(56))    #輸出為 <class 'int'>
```

```
print(type(56.0))  #輸出為 <class 'float'>
```

```
print(type("How are you?")) #輸出為 <class 'str'>
```

```
print(type(True))  #輸出為 <class 'bool'>
```

資料型態轉換

- 資料型態轉換

- Python 具有簡單的資料型態自動轉換功能：如果是整數與浮點運算，系統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    #num1為 12.8，浮點數
```

- 如果系統無法自動進行資料型態轉換，就需以資料型態轉換命令強制轉換。Python 強制資料型態轉換命令有：
 - int()：強制轉換為整數資料型態。
 - float()：強制轉換為浮點數資料型態。
 - str()：強制轉換為字串資料型態。

輸出與輸入用法 (進階)

- Print 輸出命令

- print 命令能列印指定項目的內容，語法為：

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

- **項目1, 項目2, …**：print 命令可以一次列印多個項目資料，項目之間以逗號「,」分開。
 - **sep**：分隔字元，如果列印多個項目，項目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 命令會列印在下一列。

Print的參數格式化

```
print (項目 % (參數列))
```

- 「%」參數格式化
 - print 命令支援參數格式化功能，語法為：
 - 常用的參數有：

參數	意義
%%	在字串中顯示「%」。
%d	以整數資料型態輸出。
%f	以浮點數資料型態輸出。
%s	以字串資料型態輸出。
%e 或 %E	以科學記號方式輸出。

Print的參數格式化 (cont')

- 即以「%s」代表字串、「%d」代表整數、「%f」代表浮點數，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，若少於 5 位數，會在數字左方填入空白字元(若大於 5 位數則會全部列印)。
 - %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 個字元則會全部列印)。
 - %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」字元。

輸入資料到程式裡面

- input 輸入命令

- input 命令的語法為：`變數 = input([提示字串])`

- 「提示字串」是輸出一段提示訊息，告知使用者如何輸入。輸入資料時，當使用者按下 Enter 鍵後就視為輸入結束，input 命令會將使用者輸入的資料存入變數中。

- 輸入資料的型態為字串

- 使用 input 命令時，使用者輸入的資料型態為字串。初學者最容易產生的錯誤是當使用者輸入數字型態的資料時，程式設計者會將其視為數值資料型態而直接進行數值運算，於是發生資料型態不符合的錯誤。

運算式

- 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：「 $2 + 3$ 」中的「 $+$ 」是運算子，「 2 」及「 3 」是運算元。
- 運算子依據運算元的個數分為單元運算子及二元運算子：
 - 單元運算子：只有一個運算元，例如「 -100 」中的「 $-$ 」（負）、「 $\text{not } x$ 」中的「 not 」等，單元運算子是位於運算元的左方。
 - 二元運算子：具有兩個運算元，例如「 $100 - 30$ 」中的「 $-$ 」（減）、「 $x \text{ and } y$ 」中的「 and 」，二元運算子是位於兩個運算元的中間。

運算子列表

- 算術運算子

- 用於執行一般數學運算的運算子稱為「算術運算子」。

運算子	意義	範例	範例結果
+	兩運算元相加	12+3	15
-	兩運算元相減	12-3	9
*	兩運算元相乘	12*3	36
/	兩運算元相除	32/5	6.4
%	取得餘數	32%5	2
//	取得整除的商數	32//5	6
**	(運算元 1) 的 (運算元 2) 次方	7**2	$7^2 = 49$

比較運算子

■ 比較運算子

- 比較運算子會比較兩個運算式，若比較結果正確，就傳回 True，若比較結果錯誤，就傳回 False。設計者可根據比較結果，進行不同處理程序。

運算子	意義	範例	範例結果
==	運算式 1 是否等於運算式 2	(6+9==2+13) (8+9==2+13)	True False
!=	運算式 1 是否不等於運算式 2	(8+9!=2+13) (6+9!=2+13)	True False
>	運算式 1 是否大於運算式 2	(8+9>2+13) (6+9>2+13)	True False
<	運算式 1 是否小於運算式 2	(5+9<2+13) (8+9<2+13)	True False
>=	運算式 1 是否大於或等於運算式 2	(6+9>=2+13) (3+9>=2+13)	True False
<=	運算式 1 是否小於或等於運算式 2	(3+9<=2+13) (8+9<=2+13)	True False

邏輯運算子

■ 邏輯運算子

- 邏輯運算子通常是結合多個比較運算式來綜合得到最終比較結果，用於較複雜的比較條件。

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都是 True 時，才傳回 True ，其餘情況皆傳回 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都是 False 時，才傳回 False ，其餘情況皆傳回 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

邏輯運算中，Python 的特殊指令 in 與is 運算子

- 「x in y」用於判斷x 是否為y 其中一個元素，若是則回傳True，否則回傳False
- 「x is y」用於判斷兩物件的id 是否相同，若變數x 與變數y 指向相同物件，則回傳True，否則回傳False

in 與is 運算子

運算子	說明	舉例	結果
in	是否包含	<pre>x = 1 y = [1, 2, 3] print(x in y)</pre>	True 註：[1, 2, 3] 為串列，下一章會介紹。
not in	是否不包含	<pre>x = 1 y = [1, 2, 3] print(x not in y)</pre>	False
is	是否為相同物件	<pre>x = [1, 2, 3] y = [1, 2, 3] print(x is y)</pre>	False
is not	是否不為相同物件	<pre>x = [1, 2, 3] y = [1, 2, 3] print(x is not y)</pre>	True

in 與is 運算子

- 「`x == y`」只要`x` 與`y` 數值相同就回傳True
- 「`x is y`」要`x` 與`y` 需要參考到相同物件才回傳True

1	<code>x = [1,2,3]</code>
2	<code>y = [1,2,3]</code>
3	<code>print(id(x),id(y))</code>
4	<code>print(x is y)</code>
5	<code>print(x == y)</code>

in 與is 運算子

- 執行結果

```
5192360 5156576
```

```
False
```

```
True
```

■ 執行結果說明

- 變數x 參考整數串列「1, 2, 3」時，id(x) 為5192360，
- 變數y 參考到整數串列「1, 2, 3」時，id(y) 為5156576，
- 因x 與y 的id 不同，所以「x is y」的結果為False，因x 與y都是整數串列「1, 2, 3」，所以「x == y」的結果為True

複合指定運算子

■ 複合指定運算子

- 複合指定運算子就是為簡化此種敘述產生的運算子，將運算子置於「=」前方來取代重複的變數名稱。例如：

$i += 3$ #即 $i = i + 3$

$i -= 3$ #即 $i = i - 3$

- 複合指定運算子同時做了「執行運算」及「指定」兩件工件。
- 下表是以i 變數值為10 來計算範例結果：

運算子	意義	範例	範例結果
+=	相加後再指定給原變數	$i += 5$	15
-=	相減後再指定給原變數	$i -= 5$	5
*=	相乘後再指定給原變	$i *= 5$	50
/=	相除後再指定給原變數	$i /= 5$	2
%=	相除得到餘數後再指定給原變數	$i \% = 5$	0
//=	相除得到整除商數後再指定給原變數	$i //= 5$	2
**=	做指數運算後再指定給原變數	$i ** = 3$	1000

某些運算子，不僅僅只有加減乘除

- 運算子「+」的功能
- 運算子「+」可用於數值運算，也可用於字串組合，使用時需特別留意運算元的資料型態。
 - 運算子「+」用於數值運算時是計算兩個運算元的總和
 - 運算子「+」用於字串組合時是將兩個運算元的字元組合在一起

運算優先順序

■ 運算子的優先順序

■ 下表為常見運算子的優先順序：

- 優先順序高（數字較小）者先執行運算，同一列中的運算子具有相同的優先順序，優先順序相同時是由左至右運算。

優先順序	運算子
1	() 小括號
2	+ (正數) 、 - (負數)
3	* (乘法) 、 / (除法) 、 % (取餘數) 、 // (取商)
4	+ (加法) 、 - (減法)
5	== 、 != 、 > 、 < 、 >= 、 <=
6	not
7	and
8	or
9	= 、 += 、 -= 、 *= 、 /= 、 %= 、 //= 、 **=

重點整理

- 在本章中介紹了下面的重點：
 - 「變數」顧名思義，是一個隨時可能改變內容的容器名稱，當設計者建立一個變數時，應用程式就會配置一塊記憶體給此變數使用，以變數名稱做為辨識此塊記憶體的標誌，設計者就可在程式中將各種值存入該變數中。
 - 註解的用途是做為程式的說明。
 - Python 數值資料型態有整數 (int)、浮點數 (float) 及布林值 (bool)。
 - Python 字串資料型態 (str) 是變數值以一對雙引號 (「"」) 或單引號 (「'」) 包起來。
 - print 命令能列印指定項目的內容，語法為：

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

重點整理 (cont ')

■input 命令的語法為：

```
變數 = input([提示字串])
```

- 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」中的「+」是運算子，「2」及「3」是運算元。
- 用於執行一般數學運算的運算子稱為「算術運算子」。
- 比較運算子會比較兩個運算式，若比較結果正確，就傳回 **True**，若比較結果錯誤，就傳回 **False**。設計者可根據比較結果，進行不同處理程序。
- 邏輯運算子通常是結合多個比較運算式來綜合得到最終比較結果，用於較複雜的比較條件。
- 複合指定運算子同時做了「執行運算」及「指定」兩件工作。