



附錄 A Python 基礎語法查詢表

A-1 Python 基本概念

A-1-1 語言語意

Python 程式語言的特色強調簡單閱讀並明確指示。

它是一種「物件導向的高階程式語言」，也是「直譯式程式語言」。任何數字、字串、資料結構、函式、類別、模組等，只要是 Python 直譯器裡的東西皆有它自己的「盒子」，這個盒子就是「Python 物件」。

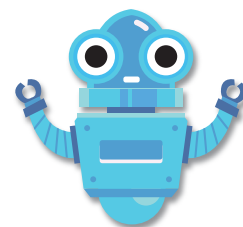
A-1-2 縮排

空白在 Python 中具有非常重要的意義，不同於 R、C++、Java 或 Perl 其他程式語言，行首空白是重要的，又稱為縮進。行首的空白（空格和 tab 鍵）用來決定邏輯行的層次，從而決定語句分組。代表同一層次的程式碼必須有相同的縮進，每一組這樣的語句稱為一個「塊」。

例如下面的程式碼為 for 迴圈的表示法：

```
for x in array:
    if x < pivot:
        less.append(x)
    else:
        greater.append(x)
```

注意：不要混合使用空格鍵和 Tab 鍵來縮排，因為在跨越不同的平台時無法正常工作。





A-1-3 資料型態

在 Python 程式語言中，標準函式庫有一些內建型態，用來處理數值資料、字串、布林值（True、False）以及日期時間資料。

型態	說明
None	Python 中的「null」值，如果一個函式沒有指定回傳值，即為隱式的回傳 None。
bytes	純 ASCII 位元組。
int	整數，如：1、2、3。
float	浮點數，如：3.33。
str	字串，可以用單引號和雙引號括出字串，但撰寫程式時，盡量維持使用同一種，以免錯誤，範例如下： a = 'hey. This is a string.' b = "another way to use string."
bool	布林值，僅有兩個值：True、False。 用於判斷條件敘述句結果為 True（是）或 False（否）。

A-1-4 呼叫函式及物件方法

在 Python 程式語言中，呼叫函式時，可在括號中放入 0 至多個參數，或選擇將傳回值指定給某個變數：

```
result = f(x, y, z)
```

A-1-5 資料輸出

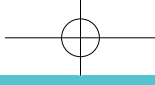
print 命令可以列印指令的項目內容，用法參照下面程式碼：

```
print ('項目 1', '項目 2', '.....')
```

A-1-6 變數及參數傳遞

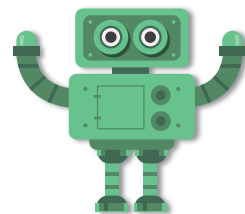
在 Python 程式語言中設定一個變數時，其實是為等號右邊的物件建立參照（reference），以整數 list 為例，將 1、2、3 的 list 指派給 a 這個變數：

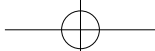
```
a = [1, 2, 3]
```



A-1-7 二元運算子

運算子	名稱	說明
$a + b$	加	a 加 b。
$a - b$	減	a 減 b。
$a * b$	乘	a 乘以 b。
$a ** b$	次方	a 的 b 次方。
a / b	除	a 除以 b。
$a // b$	取得整數的商數	a 除以 b 之商數取整數。
$a \% b$	取得餘數	a 除以 b 之餘數。
$a == b$	等於	a 是否等於 b，如果相等結果為 True。
$a != b$	不等於	a 是否等於 b，如果不相等結果為 True。
$a > b$	大於	a 是否大於 b，如果是結果為 True。
$a < b$	小於	a 是否小於 b，如果是結果為 True。
$a >= b$	大於或等於	a 是否大於或等於 b，如果是結果為 True。
$a <= b$	小於或等於	a 是否小於或等於 b，如果是結果為 True。





A-1-8 條件式

條件式	名稱	說明
not	傳回比較結果相反的值	即比較結果為「True」，傳回「False」，比較結果為「False」，傳回「True」。
and	比較兩個運算元的結果	唯有在兩個運算元的比較結果都是「True」才回傳「True」，否則皆回傳「False」。
or	比較兩個運算元的結果	唯有在兩個運算元的比較結果都是「False」才回傳「False」，否則皆回傳「True」。
if	單向判斷式	檢查條件式結果為「True」時，執行程式區塊命令；若為「False」時則不執行。
elif	判斷式	if 判斷式後可以選擇性寫一或多個 elif，若前面條件式結果皆為「False」時，則執行程式 else 區塊命令。
else	判斷式	if 判斷式後可以選擇性增加 else，若前面條件式結果為「False」時，則執行程式 else 區塊命令。

A-1-9 迴圈

迴圈	名稱	說明
for	for 迴圈	執行固定次數的迴圈，for 迴圈會對一個集合（如：list 或 tuple），進行迭代的動作。
while	while 迴圈	執行不固定次數的迴圈，while 迴圈會一直執行其條件式，直到條件式被評估為 False，或是用 break 命令中斷執行。

想學習更多關於 Python 程式語言的基礎與應用，您可以到 AI4kids 官網 www.ai4kids.ai 的線上課程進行查詢。

