**Python程式語法**

**Python**是一種廣泛使用的[高階程式語言](https://zh.wikipedia.org/wiki/%E9%AB%98%E7%BA%A7%E8%AF%AD%E8%A8%80" \o "高階語言)。Python 的設計哲學強調代碼的[可讀性](https://zh.wikipedia.org/wiki/%E5%8F%AF%E8%AF%BB%E6%80%A7" \o "可讀性)和簡潔的語法（尤其是使用空格縮排劃分代碼塊，而非使用大括弧或者關鍵詞）。相比於 C++ 或 Java，Python 讓開發者能夠用更少的代碼表達想法。不管是小型還是大型程式，該語言都試圖讓程式的結構清晰明了。

Python 擁有[動態型別系統](https://zh.wikipedia.org/wiki/%E9%A1%9E%E5%9E%8B%E7%B3%BB%E7%B5%B1" \o "類型系統)和[垃圾回收](https://zh.wikipedia.org/wiki/%E5%9E%83%E5%9C%BE%E5%9B%9E%E6%94%B6)功能，能夠自動管理記憶體使用，並且支援多種編程範式，包括物件導向、命令式、函數式和程序式編程。其本身擁有一個巨大而廣泛的標準庫。

**語句和控制流**

* if語句： 當條件成立時執行語句塊。經常與else, elif（相當於else if）配合使用。
* for語句： 遍列列表、字串、字典、集合等[疊代器](https://zh.wikipedia.org/wiki/%E8%BF%AD%E4%BB%A3%E5%99%A8" \o "疊代器)，依次處理疊代器中的每個元素。
* while語句：當條件為真時，循環執行語句塊。
* try語句：與except, finally, else配合使用處理在程式執行中出現的異常情況。
* class語句：用於定義類型。
* def語句：用於定義函式和類型的方法。
* pass語句：表示此行為空，不執行任何操作。
* import語句：匯入一個模組或包。常用寫法：from module import name, import module as name, from module import name as anothername

**Print 函式**

　　Python的函式支援[遞迴](https://zh.wikipedia.org/wiki/%E9%80%92%E5%BD%92" \o "遞迴)、預設參數值、可變參數、閉包，但不支援函式[過載](https://zh.wikipedia.org/wiki/%E9%87%8D%E8%BC%89)。為了增強代碼的可讀性，可以在函式後書寫「文件字串」（Documentation Strings，或者簡稱docstrings），用於直譯函式的作用、參數的類型與意義、返回值類型與取值範圍等。

　　函式呼叫時，實參可以如同C語言那樣按照位置與形參符合；也可以按照[命名參數](https://zh.wikipedia.org/w/index.php?title=%E5%91%BD%E5%90%8D%E5%8F%83%E6%95%B8&action=edit&redlink=1)（英語：[Named parameter](https://en.wikipedia.org/wiki/Named_parameter)）形式呼叫，即 param\_name=value 形式的實參。在一個函式呼叫的實參表中，關鍵字引數必須出現在位置參數之後。

　　Python的函式作為[第一類物件](https://zh.wikipedia.org/wiki/%E7%AC%AC%E4%B8%80%E9%A1%9E%E7%89%A9%E4%BB%B6" \o "第一類物件)，具有和普通變數平等的地位。函式一旦定義，即可視作為普通物件，其形參會保留上次呼叫時的值，但在函式新的一次呼叫時會被實參值覆蓋。因此函式的**預設參數值**在連續多次呼叫該函式時，如果不被實參值覆蓋，就會一直保留。

　　Python的函式實參與形參之間的結合是傳遞物件的參照[[8]](https://zh.wikipedia.org/zh-tw/Python#cite_note-8)。這是因為Python的賦值操作是把（變數）名字繫結到物件上。形實結合也是這種方式。如果形參繫結到一個可變的物件，則通過形參對此物件內容的修改，在函式外也是可見的。如果形參繫結到一個不可變的物件，則通過形參是不能修改此物件內容，但可以把形參重新繫結到其它物件上，這並不影響函式外的物件的值。

**數學運算**

　　Python使用與C、Java類別似的運算子，支援整數與浮點數的數學運算。同時還支援複數運算與無窮位數（實際受限於電腦的能力）的整數運算。除了求絕對值函式abs()外，大多數數學函式處於math和cmath模組內。前者用於實數運算，而後者用於複數運算。使用時需要先匯入它們，比如：

　　fractions模組用於支援分數運算；decimal模組用於支援高精度的浮點數運算；第三方庫Sympy用於支援數學符號運算。

　　Python定義求余執行a % b的值處於開區間[0, b)內，如果b是負數，開區間變為(b, 0]。這是一個很常見的定義方式。不過其實它依賴於整除的定義。為了讓方程式：b \* (a // b) + a % b = a恆真，整除執行需要向負無窮小方向取值。比如7 // 3的結果是2，而(-7) // 3的結果卻是-3。這個演算法與其它很多程式語言不一樣，需要注意，它們的整除運算會向0的方向取值。

　　Python允許像數學的常用寫法那樣連著寫兩個比較執行符。比如a < b < c與a < b and b < c等價。C++的結果與Python不一樣，首先它會先計算a < b，根據兩者的大小獲得0或者1兩個值之一，然後再與c進行比較。