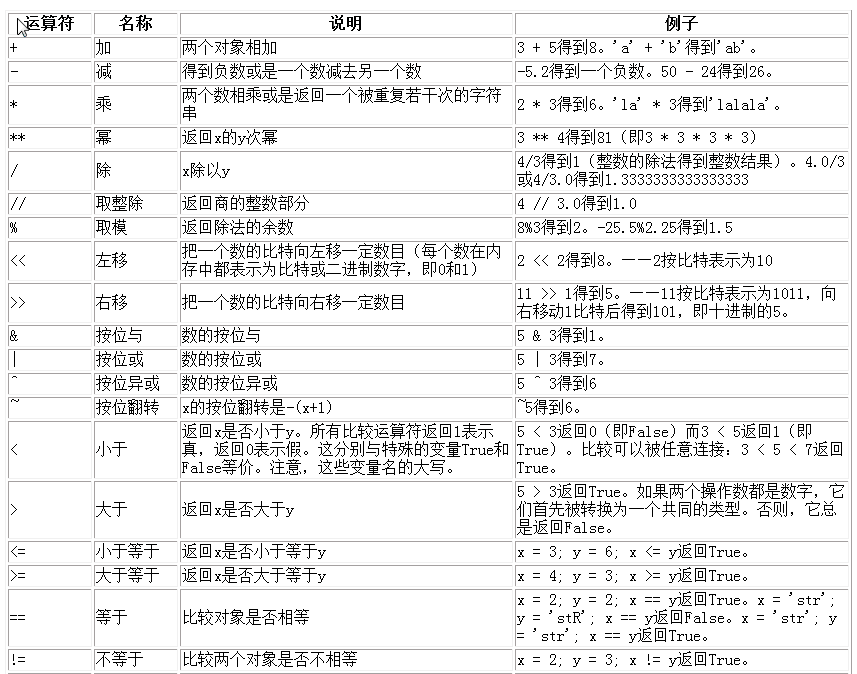
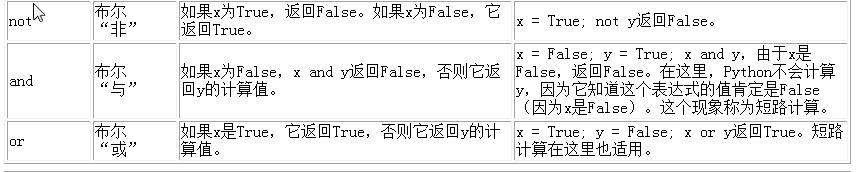
一、简介。

　　咱们编写的大多数语句（逻辑行）都包含**表达式**。一个简单的表达式例子如2 + 3。一个表达式可以分解为运算符和操作数。*运算符* 的功能是完成某件事，它们由如+这样的符号或者其他特定的关键字表示。运算符需要数据来进行运算，这样的数据被称为 *操作数* 。在这个例子中，2和3是操作数。

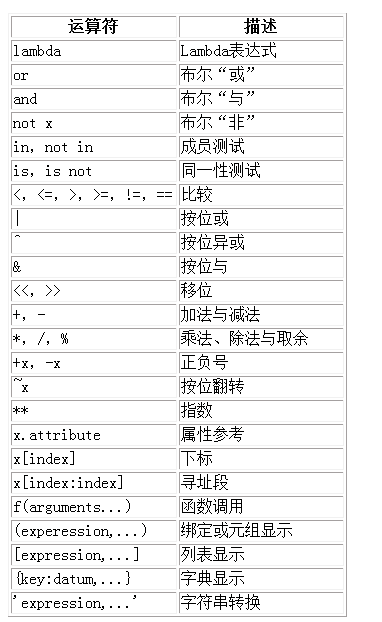
二、Python中的运算符。





三、运算符的优先级。

　　下面这张表（与Python参考手册中的那个表一模一样）已经顾及了完整的需要。事实上，我建议你使用圆括号来分组运算符和操作数，以便能够明确地指出运算的先后顺序，使程序尽可能地易读。例如，2 + (3 \* 4)显然比2 + 3 \* 4清晰。与此同时，圆括号也应该正确使用，而不应该用得过滥（比如2 + (3 + 4)）。



import math

math.abs(x) 專用於把int變絕對值

math.fabs(x) 專用於把float變絕對值

**sum(iterable[, start])**

Sums start and the items of an iterable from left to right and returns the total. start defaults to 0. The iterable‘s items are normally numbers, and the start value is not allowed to be a string.

For some use cases, there are good alternatives to sum(). The preferred, fast way to concatenate a sequence of strings is by calling ''.join(sequence). To add floating point values with extended precision, see math.fsum(). To concatenate a series of iterables, consider using itertools.chain().