

# Operadores do Python

Operador	Nome	Explicação	Exemplos
+	Adição	Executa a soma de dois valores	3 + 5 dá 8. 'a' + 'b' dá 'ab'
-	Subtração	Torna um número negativo ou executa subtração de um número menos outro	-5.2 torna um número negativo. 50 - 24 dá 26
*	Multiplicação	Executa a multiplicação de dois números ou retorna a string repetida tantas vezes	2 * 3 dá 6. 'la' * 3 dá 'lalala'.
**	Potência	Retorna x elevado a y	3 ** 4 dá 81 (3 * 3 * 3 * 3)
/	Divisão	Divide x por y	4/3 dá 1 (divisão de inteiros dá um inteiro) 4.0/3 ou 4/3.0 dá 1.3333333333333333
%	Módulo	Retorna o resto da divisão	8%3 dá 2. -25.5%2.25 dá 1.5
<<	Deslocamento de bits a esquerda	Desloca tantos bits a esquerda	2 << 2 dá 8. - 2 é representado em bits por 10 . Deslocar dois bits a esquerda dá 1000 o qual é representado pelo decimal 8
>>	Deslocamento de bits a direita	Desloca tantos bits a direita	11 >> 1 dá 5 - 11 é representado em bits por 1011 . Deslocar um bit a direita dá 101 o qual é representado pelo decimal 5
&	Operador bit a bit AND	Bits configurados nos dois operadores são configurados no resultado	5 & 3 dá 1
	Operador bit a bit OR	Bits configurados em um ou outro operador são configurados no resultado	5   3 dá 7
^	Operador bit a bit XOR	Bits configurados em um ou outro operador, mas não em ambos, são configurados no resultado	5 ^ 3 dá 6
~	Operador bit a bit NOT	Bits configurado no operador não são configurados no resultado e vice-versa	~5 dá -6
<	Menor que	Retorna se x é menor que y. Toda comparação retorna 1 para verdadeiro e 0 para falso. Isto é equivalente as variáveis especiais True e False respectivamente.	5 < 3 dá 0 (False) e 3 < 5 dá 1 (True). Comparações podem ser encadeadas: 3 < 5 < 7 dá True
>	Maior que	Retorna se x é maior que y	5 > 3 retorna True. Se os dois operandos são números, eles são primeiro convertidos para um tipo comum
<=	Menor ou igual a	Retorna se x é menor ou igual a y	x = 3; y = 6; x <= y retorna True
>=	Maior ou igual a	Retorna se x é maior ou igual a y	x = 4; y = 3; x >= 3 retorna True
==	Igual a	Compara se os objetos são iguais	x = 2; y = 2; x == y retorna True.x = 'str'; y = 'stR'; x == y retorna False.x = 'str'; y = 'str'; x == y retorna True.
!=	Diferente de	Compara se os objetos são diferentes	x = 2; y = 3; x != y retorna True.
not	Operador booleano NOT	Se x é True, ele retorna False. Se x é False, ele retorna True.	x = True; not y retorna False.
and	Operador booleano AND	x and y retorna False se x é False	x = False; y = True; x and y retorna Falseuma vez que x é False. Neste caso, Python não avaliará y uma vez que ele sabe que o valor da expressão terá que ser False (pois x é False). Isto é chamado short-circuit evaluation.
or	Operador booleano OR	Se x é True, ele retorna True	x = True; y = False; x or y retornra True. Short-circuit evaluation é aplicado aqui também.

## Precedência dos Operadores

Operador	Descrição
lambda	Expressão Lambda
or	Operador booleano OR
and	Operador booleano AND
not x	Operador booleano NOT
in, not in	teste de membros
is, is not	teste de identidade
<, <=, >, >=, !=, ==	Comparações
	Operador bit a bit OR
^	Operador bit a bit XOR
&	Operador bit a bit AND
<<, >>	Deslocamentos de bits
+, -	Adição e subtração
*, /, %	Multiplicação, Divisão e Resto
+x, -x	Positivo, Negativo
~x	Operador bit a bit NOT
**	Potenciação
x.attribute	Referência a um atributo
x[index]	Subscrição
x[index:index]	Repartição
f(arguments ...)	Chamada de função
(expressions, ...)	Exibição de tupla
[expressions, ...]	Exibição de lista
{key:datum, ...}	Exibição de dicionário
`expressions, ...`	Conversão de string