

6. Những toán tử cơ bản trong Python

Về cơ bản những toán tử trong Python cũng khá giống với các ngôn ngữ lập trình khác như C/C++/Java. Ngoài ra Python còn bổ sung thêm một số toán tử khá mới mà tôi sẽ trình bày ở phần dưới đây. Nhưng trước khi đi vào chi tiết các toán tử, các bạn cần hiểu toán tử là gì.

Toán tử là các cấu trúc có thể thao tác giá trị của các toán hạng.

Ví dụ với phép toán $2 + 3$, thì ở đây 2 và 3 là các toán hạng và $+$ là toán tử.

Dưới đây là các loại toán tử cơ bản sử dụng trong Python cùng với ý nghĩa và ví dụ cho mỗi toán tử. Những toán tử đó có thể áp dụng tới tất cả các toán hạng là kiểu số.

Loại toán tử	Toán tử	Ý nghĩa	Ví dụ
Toán tử số học	$+$, $-$	Phép cộng, phép trừ	100-1
	$*$, $/$, $\%$	Phép nhân, chia, module	<pre>>>> 30 %4 2</pre>
	$**$	Mũ	<pre>>>> 2**4 16</pre>
	$//$	Truncation Division (còn gọi là floordivision hay floor division). Kết quả của phép chia này là một phần của kết quả, nghĩa là phần phân thập phân bị cắt ngắn (nếu có). Nếu cả hai số chia và số bị chia là số nguyên, kết quả cũng sẽ là một số nguyên. Nếu một trong hai số chia hoặc số bị chia là một float, kết quả sẽ là kết quả cắt ngắn như một float. Nhưng nếu một trong các toán hạng là số âm, kết quả sẽ được làm tròn.	<pre>>>> 12//4 3 >>> 12.1//4 3.0 >>> -12.1//4 -4.0</pre>
Toán tử so sánh	$<$, $<=$, $>$, $>=$	So sánh nhỏ hơn, nhỏ hơn hoặc bằng, lớn hơn, lớn hơn hoặc bằng	<pre>>>> 2<=34 True</pre>

			<pre>>>> 2 > 34 False</pre>
	==	So sánh bằng	<pre>>>> 2 == 2 True</pre>
	!= hoặc <>	So sánh khác (không bằng)	<pre>>>> 2 <> 2 False</pre>
Toán tử gán	=, +=, -=, *=, /=, %= **=, //=	Phép gán, cộng và gán, Trừ và gán, nhân và gán, Chia và gán, chia lấy phần dư và gán, Mũ và gán, floor division và gán	<pre>>>> c = 2+3 >>> print (c) 5 >>> c += 1 >>> print (c) 6</pre>
Toán tử thao tác bit	~, &, ^	Thao tác với bit ứng với phép 'OR', 'AND', và 'XOR'	<pre>>>> bin(0b1011^0b0101) '0b1110'</pre>
	<<, >>	Toán tử dịch trái, dịch phải nhị phân	<pre>>>> 12<<2 48</pre>
	~	Là một thao tác lật bit: $\sim x + x = -1$	<pre>>>> a = -1234 >>> print (~a+a) -1</pre>

Toán tử logic	or, and, not	Phép toán 'hoặc', 'và', 'phủ định'	<pre>>>> not (True or False) False</pre>
Toán tử membership	in, not in	Kiểm tra có tồn tại trong tập hay không.	<pre>>>> 1 in [1,2,3] True >>> 1 not in [1,2,3] False >>> 34 in [1,2,3] False</pre>
Toán tử 1 ngôi (toán tử đơn vị)	+, -	+X, -X: Dấu dương, âm đại số	<pre>>>> +4 4 >>> -4 -4</pre>

Thứ tự ưu tiên của các toán tử

Thứ tự ưu tiên của các toán tử là điều rất quan trọng các bạn cần nắm được khi thực hiện các phép toán kết hợp nhiều toán tử. Các toán tử được liệt kê trong bảng theo thứ tự ưu tiên từ cao đến thấp.

	Toán tử	Mô tả
1	**	
2	~ + -	Đảo bit(lấy phần bù); phép cộng, trừ một ngôi
3	~ / % //	
4	+ -	Toán tử cộng, trừ

5	>> <<	
6	&	
7	^	
8	<= < > >=	
9	<> == !=	
10	= %= /= //= -= += *= **=	
11	in not in	
12	not or and	

Kết Luận

Như vậy là chúng ta đã có cái nhìn rõ ràng và chi tiết hơn về những toán tử cơ bản được sử dụng trong Python cũng như thứ tự ưu tiên của chúng. Trong bài tiếp theo chúng tôi sẽ giới thiệu về toán tử điều kiện và toán tử lặp trong Python, liệu chúng có khác biệt gì so với các ngôn ngữ C/C++ hay Java ?