Operators in Java

Symbols that tells compiler to perform some operation.

expression:

$$Sum = a + b$$

operands

Types of Operators

Arithmetic Operators (Binary/Unary/Ternary)
Relational Operators
Logical Operators
Bitwise Operators
Assignment Operators

Arithmetic Operators

Binary (2 operands)
+
−
*
/
% (modulo)

unary (1 operand)
++ (increment)
−− (Decrement)

pre increment	post increment way
++a	a++
1) value change	1) value use
2) value use	2) value change

Relational operators

$$==$$
$$!=$$
$$>$$
$$<$$
$$>=$$
$$<=$$

$S.O.P (A > B)$ ✗
↑
cannot find symbol
error

$S.O.P ((A > B))$ ✓

Logical operators

&& AND   & — Bitwire AND

|| OR

!   NOT

Assignment Operator

$$=$$

$+=$ $a = a + 10$

$-=$ $a = a - 10$

$*=$ $a = a * 10$

$/=$ $a = a/10$

Operator Precedence

Operators precedence determines the order in which the operator is an expression are evaluated.

For Operators	Precedence
postfix increment & Decrement	++ --
prefix increment & decrement and unary	++, --, +, -, ~, !
multiplicative	*, /, %
additive	+, -
shift	<<, >>, >>>
relational	<, >, <=, >=, instanceof
equality	==, !=
bitwise AND	&
bitwise exclusive OR	^
bitwise inclusive OR	|
logical AND	&&
logical OR	||
ternary	?:
assignment	=, !=, -=, /=, %=, &=, ^=, !=, <<=, >>=, >>>=

Associativity of Operators

If an expression has two operators with similar precedence, the expression is evaluated according to its associativity (either left to right, or right to left).

Precedence	Associativity
++, --, .	left to right
++, --, +, -, ~, !	right to left
*, /, %	left to right
+, -	left to right
<<, >>, >>>	left to right
<, >, <=, >=, instanceof	left to right
equality ==, !=	left to right
&	left to right
^	left to right
|	left to right
&&	left to right
||	right to left
?:	
=, +=, -=, *=, /=, %=, &=, ^=, |=	right to left