

6

Operators

What is Operator

- › Operator is a symbol to perform operation.
- › Operator receives one or more operands (values) and returns one value.
- › Ex: +, -, *, /, == etc.

Classification of Operators

- › Arithmetical Operators
- › Assignment Operators
- › Increment and Decrement Operators
- › Comparison Operators
- › Logical Operators
- › Concatenation Operator
- › Ternary Operator

Arithmetical Operators

- › Used to perform arithmetical operations on the numbers
- › + Addition
- › - Subtraction
- › * Multiplication
- › / Division

- › % Remainder

Assignment Operators

- › Used to perform arithmetical operations on the numbers
- › = Assigns to
- › += Add and Assigns to
- › -= Subtract and Assigns to
- › *= Multiply and Assigns to
- › /= Divide and Assigns to
- › %= Remainder Assigns to

Increment / Decrement Operators

- › Used to perform arithmetical operations on the numbers
- › It returns the incremented / decremented value and also overwrites the value of variable.
- › n++ Post-Incrementation (First it returns value; then increments)
- › ++n Pre-Incrementation (First it increments value; then returns)
- › n-- Post-Decrementation (First it returns value; then decrements)
- › --n Pre-Decrementation (First it decrements value; then returns)

Comparison Operators

- › Used to compare two values and return true / false, based on the condition.
- › == equal to
- › != not equal to
- › < less than
- › > greater than

- › <= less than or equal to
- › >= greater than or equal to

Logical Operators

- › Checks both operands (Boolean) and returns true / false.

& Logical And (Both operands should be true)

Evaluates both operands,
even if left-hand operand returns false.

&& Conditional And (Both operands should be true)

Doesn't evaluate right-hand operand,
if left-hand operand returns false.

| Logical Or (At least any one operand should be true)

Evaluates both operands,
even if left-hand operand returns true.

|| Conditional Or (At least any one operand should be true)

Doesn't evaluate right-hand operand,
if left-hand operand returns true.

^ Logical Exclusive Or - XOR (Any one operand only should be true)

Evaluates both operands.

! Negation (true becomes false; False becomes true)

Concatenation Operator

- › Attaches second operand string at the end of first operand string and returns the combined string.

+ "string1" + "string2" returns "string1string2" (as string)
 "string1" + number returns "stringnumber" (as string)

number + "string1" returns "numberstring" (as string)

Ternary Conditional Operator

- › It evaluates the given Boolean value;
- › Returns first expression (consequent) if true;
- › Returns second expression (alternative) if false.

? : (condition)? consequent : alternative

Operator Precedence

Category	Operator
Postfix	() [] ++ --
Unary	+ - !
Multiplicative	* / %
Additive	+ -
Relational	< <= > >=
Equality	== !=
Logical AND	&&
Logical OR	
Conditional	?:
Assignment	= += -= *= /= %=

- Operators are evaluated based on the order of precedence.
- highest precedence appear at the top of the table