

# Operators in TypeScript

## 1) Arithmetic Operators

These operators perform basic mathematical operations.

- + (Addition): Adds two numbers →  $10 + 5 = 15$
- - (Subtraction): Subtracts one number from another →  $10 - 5 = 5$
- \* (Multiplication): Multiplies two numbers →  $10 * 5 = 50$
- / (Division): Divides one number by another →  $10 / 5 = 2$
- % (Modulus): Returns the remainder of division →  $10 \% 3 = 1$
- \*\* (Exponentiation): Raises a number to a power →  $2 ** 3 = 8$

## 2) Assignment Operators

Used to assign values to variables.

- $+= \rightarrow x += 5$  (same as  $x = x + 5$ )
- $-= \rightarrow x -= 5$  (same as  $x = x - 5$ )
- $*= \rightarrow x *= 5$  (same as  $x = x * 5$ )
- $/= \rightarrow x /= 5$  (same as  $x = x / 5$ )
- $%= \rightarrow x \%= 5$  (same as  $x = x \% 5$ )

## 3) Increment & Decrement Operators

Used to increase or decrease a value by 1.

- $++$  (Increment)
  - $x++$  (Post-increment: First use the value, then increase it)
  - $++x$  (Pre-increment: First increase, then use the value)
- $--$  (Decrement)
  - $y--$  (Post-decrement: First use the value, then decrease it)
  - $--y$  (Pre-decrement: First decrease, then use the value)

## 4) Relational/Comparison Operators

Used to compare values and return true or false.

- $<$  (Less than) →  $10 < 20 \rightarrow \text{true}$

- > (Greater than) → 10 > 20 → false
- <= (Less than or equal to) → 10 <= 10 → true
- >= (Greater than or equal to) → 20 >= 15 → true
- == (Equality check, only compares value) → 10 == "10" → true
- != (Not equal) → 10 != 20 → true
- === (Strict equality, compares both value and type) → 10 === "10" → false
- !== (Strict inequality) → 10 !== "10" → true

## 5) Logical Operators

Used to combine multiple conditions.

- && (AND) → Returns true if **both** conditions are true
  - Example: (x > 5 && x < 15) → true only if x is between 5 and 15
- || (OR) → Returns true if **at least one** condition is true
  - Example: (x > 10 || y < 5) → true if any condition is true
- !(NOT) → Reverses the condition (true → false, false → true)
  - Example: !(x > 5) → If x > 5 is true, ! makes it false

## 6) Ternary Operator (Conditional Operator)

A shortcut for if-else.

- **Syntax:**

```
condition ? value_if_true : value_if_false;
```

- **Example:**

```
let age = 18;
let result = age >= 18 ? "Adult" : "Minor";
console.log(result); // "Adult"
```

If age is 18 or more, it prints "Adult", otherwise "Minor".