1Z0-808 Exam Topic Reviewer

TopicId: 1008
Topic: Java Operators and Precedence

August 5, 2025

Topic 1008: Java Operators and Precedence

Thinking Like the Compiler: The Rules of Evaluation

Operators are the verbs of Java; they perform actions. Simple expressions like `2 + 3` are easy, but the exam will present complex expressions and ask for the exact result. To answer correctly, you can't just guess the order of operations—you must know it. This topic is all about operator precedence (which operator goes first) and associativity (what to do in a tie). Getting this right is pure precision.

The Most Important Operators for the Exam

Unary Operators: `++` and `--`

This is a guaranteed exam topic. You must master the difference between pre- and post-increment/decrement.

- **Post-Increment (`x++`):** The expression evaluates to the *original* value of `x`, and *then* `x` is incremented.
- **Pre-Increment (`++x`):** `x` is incremented *first*, and the expression evaluates to the *new* value of `x`.

```
int a = 5;
int b = a++; // Step 1: b is assigned the original value of a (5).
             // Step 2: a is incremented to 6.
             // Result: a is 6, b is 5.

int c = 5;
int d = ++c; // Step 1: c is incremented to 6.
             // Step 2: d is assigned the new value of c (6).
             // Result: c is 6, d is 6.
```

Logical Operators and Short-Circuiting: `&&` and `||`

This is another critical concept. The short-circuiting operators can prevent the right-hand side of an expression from being executed.

- **`&&` (AND):** If the left side is `false`, the result is automatically `false`, and the right side is **never evaluated**.
- **`||` (OR):** If the left side is `true`, the result is automatically `true`, and the right side is **never evaluated**.

This is crucial when the right-hand side has a side effect, like an increment operator.

```
int x = 10;
// The left side (x > 10) is false. The right side (x++ > 10)
// is never executed. x is not incremented.
if (x > 10 && x++ > 10) {
    // not reached
}
System.out.println(x); // Prints 10
```

```
// The non-short-circuiting version (&) evaluates both sides always.
int y = 10;
if (y > 10 & y++ > 10) {
    // not reached
}
System.out.println(y); // Prints 11 (y++ was executed)
```

Other Key Operators

- **Integer Division:** An `int` divided by an `int` always results in a truncated `int`. `7 / 2` is 3.
- **Assignment `=`:** Has very low precedence and is right-associative. `x = y = 5;` is valid; it assigns `5` to `y`, and then assigns the result of that (which is 5) to `x`.
- **Equality `==`:** For objects, this compares memory addresses, not content. Use `.equals()` for meaningful comparison of object values.

Operator Precedence: A Simplified Guide

You don't need to memorize the entire official chart. Focus on these groups, from highest to lowest precedence.

Precedence	Operators	Associativity
1 (Highest)	Post-unary: `expr++`, `expr--`	Left-to-right
2	Pre-unary: `++expr`, `--expr`, `+`, `-`, `!`	Right-to-left
3	Multiplicative: `*`, `/`, `%`	Left-to-right
4	Additive: `+`, `-`	Left-to-right
5	Relational: `>`, `<`, `>=`, `<=`, `instanceof`	Left-to-right
6	Equality: `==`, `!=`	Left-to-right
7	Logical AND: `&&`	Left-to-right
8	Logical OR: `||`	Left-to-right
9	Ternary: `? :`	Right-to-left
10 (Lowest)	Assignment: `=`, `+=`, `-=`, etc.	Right-to-left

Golden Rule: When you see a complex expression, don't guess. Either apply the precedence rules strictly or, even better, observe how parentheses `()` are used. Parentheses override all precedence rules. For example, in `3 + 4 * 5`, multiplication happens first (result 23). In `(3 + 4) * 5`, addition happens first (result 35).

Key Takeaways for the 1Z0-808 Exam

- **Master Increment/Decrement:** Know the difference between `x++` and `++x` cold. Trace variables step-by-step.
- **Watch for Short-Circuits:** In `&&` and `||` expressions, check if the right-hand side has a side effect that might not happen.

- **Know the Precedence Tiers:** At a minimum, know that unary operators are high, then arithmetic, then relational, then logical, and finally assignment is last.
- **Trust Parentheses:** Parentheses dictate the order of evaluation, period.