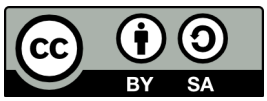
Operators

Java Fundamentals

Libre Education



Operators

An operator is a construct that performs a function to one or more values.

Types of Operators

Operators are grouped by what kind of operation they perform.

Arithmetic Operators

Arithmetic operators perform typical mathematical functions to two values.

1. = (Simple Assignment) - Assigns the value on the right to the value on the left
2. += (Add And Assign) - Adds the two values together and then assigns the value on the left to the result
3. -= (Subtract And Assign) - Subtracts the right value from the left value and then assigns the value on the left to the result
4. *= (Multiply And Assign) - Multiplies the values together and then assigns the value on the left to the result
5. /= (Divide And Assign) - Divided the value on the left by the value on the right and then assigns the value on the left to the result

Unary Operators

Unary operators require and perform a function to one value.

1. + (Unary Plus)
2. - (Unary Minus)
3. + + (Increment) - Increases the value by 1
4. - - (Decrement) - Decreases the value by 1
5. ! (Logical Compliment) - Inverts the value of a boolean

Equality And Relational Operators

Equality and Relational operators check for a condition and return a boolean but do not manipulate values

1. == (Equal To) - Returns true if the two values are equal to each other
2. != (Not Equal To) - Returns true if the two values are not equal to each other
3. ¿ (Greater Than) - Returns true if the value on the left is greater than the value on the right
4. ¡ (Less Than) - Returns true if the value on the left is less than the value on the right

5. ¿= (Greater Than or Equal To) - Returns true if the value on the left is greater than or equal to the value on the right
6. ¡= (Less Than or Equal To) - Returns true of the value on the left is less than or equal to the value on the right
7. instanceof (Instance of) - Compares an object and a class. Returns true if the object provided on the left is an instance of the class provided on the right

Conditional Operators

Conditional operators check two statements or expressions and return a boolean based on the result

1. && (Conditional AND) - Returns true if both statements are true
2. —— (Conditional OR) - Returns true if at least one of the statements are true

Bitwise and Bitshift Operators

Bitwise and Bitshift operators manipulate values at the byte level. They take place between each parallel pair of bits in the value

1. $\sim$ (Unary Bitwise Complement) - Inverts each bit in the value
2. & (Bitwise AND) - If both corresponding bits are 1, the result is 1
3. — (Bitwise Inclusive OR) - If either of the corresponding bits are 1, the result is 1
4. ˆ (Bitwise Exclusive OR) - If both of the corresponding bits are 1, the result is 1

Ternary Operator

? : - The ternary operator provides a shorter way to write and if...then...else statement and takes three operands: an expression and two values. If the expression evaluates to true the first value is returned and if the expression evaluates to false the second value is returned.

Syntax

```
result = expression ? value1 : value2;
```

Example

```
boolean expression = true;
String result = expression ? "The expression is true" : "The expression is
false";
System.out.println(result);
// Output: The expression is true
```

Other Resources

1. Wikipedia (en.wikipedia.org/wiki/Operator_(computer_programming))
2. The Java Tutorials (docs.oracle.com/javase/tutorial/java/nutsandbolts /operators.html)
3. TutorialsPoint (www.tutorialspoint.com/java/java_basic_operators.htm)

   Bitwise and Bitshift Operators
4. Rose India (www.roseindia.net/java/master-java/bitwise-bitshift- operators.shtml)