**CHAPTER 4: OPERATORS**

**Relational Operators**

- Relational operators always result in a boolean value (true or false).

- There are six relational operators: >, >=, <, <=, ==, and !=. The last two (== and !=) are sometimes referred to as equality operators.

- When comparing characters, Java uses the Unicode value of the character as the numerical value.

- Equality operators

- There are two equality operators: == and !=.

- Four types of things can be tested: numbers, characters, booleans, and reference variables.

- When comparing reference variables, == returns true only if both references refer to the same object.

**instanceof Operator**

- instanceof is for reference variables only, and checks for whether the object is of a particular type.

- The instanceof operator can be used only to test objects (or null) against class types that are in the same class hierarchy.

- For interfaces, an object passes the instanceof test if any of its superclasses implement the interface on the right side of the instanceof operator.

**Arithmetic Operators**

- There are four primary math operators: add, subtract, multiply, and divide.

- The remainder operator (%), returns the remainder of a division.

- Expressions are evaluated from left to right, unless you add parentheses, or unless some operators in the expression have higher precedence than others.

- The \*, /, and % operators have higher precedence than + and -.

**String Concatenation Operator**

- If either operand is a String, the + operator concatenates the operands.

- If both operands are numeric, the + operator adds the operands.

**Increment/Decrement Operators**

- Prefix operators (++ and --) run before the value is used in the expression.

- Postfix operators (++ and --) run after the value is used in the expression.

Eg : int a,b;

A=2;

b=a++; //a=3, b=2 (postfix)

b=++a; // a=3, b=3 (prefix)

- In any expression, both operands are fully evaluated before the operator is applied.

- Variables marked final cannot be incremented or decremented.

**Ternary (Conditional Operator)**

- Returns one of two values based on whether a boolean expression is true or false.

- Returns the value after the ? if the expression is true.

- Returns the value after the : if the expression is false.

X=(Boolean exp) ? value if true : value if false

**Logical Operators**

- The exam covers six "logical" operators: &, |, ^, !, &&, and ||.

- Logical operators work with two expressions (except for !) that must resolve to boolean values.

- The && and & operators return true only if both operands are true.

- The || and | operators return true if either or both operands are true.

- The && and || operators are known as short-circuit operators.

- The && operator does not evaluate the right operand if the left operand is false.

- The || does not evaluate the right operand if the left operand is true.

- The & and | operators always evaluate both operands.

- The ^ operator (called the "logical XOR"), returns true if exactly one operand is true.

- The ! operator (called the "inversion" operator), returns the opposite value of the boolean operand it precedes.