

## COURSE GLOSSARY

# Introduction to Java

**);** A punctuation character that terminates a statement in Java and indicates the end of a single executable instruction.

**Arithmetic operator:** Operators such as `+`, `-`, `*`, and `/` used to perform mathematical calculations on numeric values, including unary increments (`++`) and decrements (`--`).

**Array:** A fixed-size ordered collection that stores multiple values of the same type under a single variable name and provides indexed access to each element.

**Assignment operator (`=`):** The symbol used to store the value on the right-hand side into the variable on the left-hand side of the expression.

**boolean:** A primitive type that represents one of two logical values: true or false, commonly used for conditions and comparisons.

**Case-sensitive:** A characteristic of Java where uppercase and lowercase letters are treated as distinct, so identifiers like `main` and `MAIN` are different.

**char:** A primitive type that stores a single Unicode character and is written in single quotes, such as `'A'` or `'$'`.

**For loop:** A loop construct that iterates over the elements of a sequence (vector, list, or other iterable), assigning each element in turn to a loop variable and executing the loop body for each element.

**geom\_point:** A ggplot2 layer that draws individual points for each observation, creating a scatterplot when `x` and `y` aesthetics are specified

**Comparison operator:** Operators like `>`, `<`, `>=`, `<=`, `==`, and `!=` that compare values and produce a boolean result indicating the comparison outcome.

**double:** A primitive floating-point type in Java used to store 64-bit decimal numbers for greater precision in fractional values.

**float:** A primitive floating-point type in Java that stores 32-bit decimal numbers and uses less memory but offers less precision than double.

**Index:** A zero-based integer that identifies the position of an element within an array, where the first element is at index 0.

**int:** A primitive numeric type in Java used to store 32-bit whole numbers without a fractional component.

it emphasizes portability, strong typing, and a large standard library.

**Java:** A high-level, object-oriented programming language widely used for building web, desktop, enterprise, and Android applications

**long:** A primitive numeric type in Java used to store 64-bit whole numbers for very large integer values, typically suffixed with `L` when written as a literal.

**Method:** A named block of code inside a class that performs a specific task, may accept parameters, and can return a value to the caller.

**Primitive:** One of Java's eight basic built-in data types (`byte`, `short`, `int`, `long`, `float`, `double`, `char`, `boolean`) that store simple values and are not objects.

**`public static void main(String[] args):`** The standard main method signature that serves as the entry point for a Java application, where the Java Virtual Machine begins program execution.

**Semicolon (`;`)**

**String concatenation:** The operation of joining two or more strings (or a string and other values), typically using the `+` operator, producing a combined string.

**String:** A non-primitive Java type representing a sequence of characters enclosed in double quotes and providing built-in methods for manipulation such as `length()` and `toLowerCase()`.

**`System.out.println:`** A built-in Java call that prints the given text or value to the standard output (console) followed by a newline.

**Variable:** A named storage location in a program that holds a value of a specified type and can be read or updated during program execution.