

Lec 9 Variable/Constants

This lesson explains the basic building blocks of Java programs, focusing on how code is structured using alphabets, digits, and special symbols to create variables, constants, keywords, statements, and programs. It also covers the rules and conventions for naming in Java, the significance of data types, and the importance of following coding conventions like camel case for clarity and maintainability.

Java Program Structure and Components

Java programs are built using basic elements: alphabets, digits, and special symbols. These are combined to form variables, constants, keywords, classes, and methods. Statements are created by joining these elements, and a program is a collection of statements, similar to how paragraphs are formed from sentences in English.

From Alphabets to Program Elements

In Java, alphabets, digits, and special symbols are used to create meaningful elements like variables (names for storing data), constants (fixed values), keywords

(reserved words with special meaning), classes, and methods.

Writing a Simple Java Program

A simple Java program to add two numbers is written using variables and statements. Each line uses alphabets, digits, and symbols to declare variables, assign values, perform operations, and print results.

Understanding Keywords, Variables, and Constants

Keywords are reserved words in Java with predefined meanings (e.g., int for integer values). Variables are named storage locations whose values can change, while constants are fixed values assigned to variables. Assignment (=) and comparison (==) operators have different roles.

Memory Allocation for Variables

When a variable is created and initialized, a memory cell in RAM is reserved with the variable's name, storing its value. Declaration (reserving the name) and initialization (assigning a value) can be done separately or together.

Data Types and Static Typing in Java

Java requires specifying a data type for every variable, which determines what kind of value it can hold. Data type checking happens at compile time, making Java a statically typed language. Primitive and non-primitive data types exist, and data type errors are caught before running the program.

Variable Naming Rules and Scope

Within a single code block, each variable name must be unique; you cannot declare two variables with the same name in the same block. Variables can be updated multiple times, but redeclaring causes errors. Java enforces strong grammar rules for consistency.

Rules and Conventions for Naming in Java

Java identifiers (names for variables, classes, methods, etc.) can use alphabets, digits, underscores, and dollar signs, but cannot start with a digit or use spaces. Reserved keywords cannot be used as names. Naming conventions (like camel case) help make code readable and maintainable.

Camel Case Convention in Java

The camel case naming convention requires class names to start with uppercase letters and each new word in the name to also start with an uppercase letter. Method names start with a lowercase letter, with each new word capitalized. This helps distinguish between classes, methods, and keywords at a glance.

Recap and Next Steps

The session recaps the importance of understanding the basics—components of a program, naming rules, memory allocation, and conventions. Mastering these fundamentals is essential before moving on to topics like data types, operators, and control statements.