Enhanced Core Java Syllabus

This syllabus provides a detailed and structured roadmap to learn Core Java concepts, suitable for both development and automation testing roles. The plan includes practical sessions and interview preparation to ensure a well-rounded learning experience.

Schedule Overview

- **Beginners Level:** ~60 hours (~1.5–2 months)
- **Intermediate Level:** ~75 hours (~2.5-3 months)

BEGINNERS (50 Hours)

The focus here is on building foundational skills step-by-step.

1. **Introduction to Java (2 Hours)**
   - Overview of Java and its features.
   - Installation of Java and IDE setup (Eclipse).
   - Writing and running the first Java program.
2. **Java Basics (10 Hours)**
   - Object and Class.
   - Variables and data types: Primitive and Non-Primitive.
   - Operators: Arithmetic, relational, logical, and assignment.
   - Access Modifiers: public, private, protected, and default.
   - Keywords: final, static, this, and super.
   - Constructors: Default, parameterized.
3. **Control Statements (6 Hours)**
   - Conditional Statements: if, if-else, and nested if-else.
   - Loops: for, while, and do-while.
   - Jump Statements: break, continue, and return.
4. **Arrays (6 Hours)**
   - Introduction to Arrays: Definition, declaration, and initialization.
   - Single-dimensional Arrays: Accessing and iterating elements.
   - Multi-dimensional Arrays: Matrix operations.
   - Common Array Methods: Sorting, searching, and manipulation.
5. **Strings (6 Hours)**

- String Basics: Immutable and mutable strings.
- String Creation: Using literals and new keyword.
- String Methods: length, substring, concat, indexOf, and split.
- StringBuilder and StringBuffer: Usage and differences.

6. **Object-Oriented Programming (OOPS) (12-16 Hours)**
    - Encapsulation: Getters and setters.
    - Inheritance: Types of inheritance and method overriding.
    - Polymorphism: Overloading and overriding concepts.
    - Abstraction: Abstract classes and methods.
    - Interfaces: Multiple inheritance using interfaces.

INTERMEDIATE (28 Hours)

This stage covers advanced concepts that help in development and testing.

1. **Exception Handling (5 Hours)**
    - Types of Exceptions: Checked and unchecked exceptions.
    - Try-Catch Mechanism: Handling runtime errors.
    - Finally Block: Cleanup operations.
    - Difference Between Throw and Throws.
2. **Collections Framework (12 Hours)**
    - Wrapper Classes: Auto-boxing and unboxing.
    - **List**: Introduction to ArrayList and LinkedList.
    - **Set**: HashSet and TreeSet.
    - **Map**: HashMap and TreeMap.
    - Iteration: Enhanced for-loops and iterators.
    - Comparison of Collection Types: When to use what.
3. **Multithreading (4 Hours)**
    - Creating Threads: Extending Thread and implementing Runnable.
    - Thread Methods: start, sleep, and join.
    - Inter-thread Communication: wait, notify, and notifyAll.
4. **Date and Time API (3 Hours)**
    - Overview of java.util.Date and java.time classes.
    - Formatting and parsing dates.
    - Manipulating dates and times using LocalDate, LocalTime, and LocalDateTime.
5. **File Handling (4 Hours)**

  - Reading and Writing Files: Using FileReader, FileWriter, and BufferedReader.
  - Handling Binary Files.
  - Working with Properties Files.

ADVANCED topics (9 Hours)

6. **Lambda Expressions and Functional Interfaces (3 Hours)**
   - Syntax and usage of lambdas.
   - Common Functional Interfaces: Predicate, Consumer, and Supplier.
   - Basics of Stream API: Filtering and mapping data.
7. **Optional Topics (If Time Allows)**
   - **Annotations**: Basics and custom annotations (1 Hour).
   - **JDBC Basics**: Connecting to databases, executing queries, and handling results (4 Hours).