Java Complete Online Course

Course Objective

The purpose of this course is to provide a comprehensive understanding of Java programming, starting from the fundamental concepts of Core Java to the more complex topics covered in Advanced Java. This course is designed to build a solid foundation in object-oriented programming and gradually advance towards developing real-world web applications using Java-based technologies.

Whether the learner is a student, a beginner programmer, or an IT professional seeking upskilling, this course provides a step-by-step pathway to becoming a competent Java developer ready for real-world challenges.

Course Outline

Core Java

This module provides a foundational understanding of the Java programming language.

1. Introduction to Java

- History and Features of Java
- Java Virtual Machine (JVM), JRE, and JDK
- Installing Java and Setting Up Environment
- Writing First Java Program

2. Java Basics

- Data Types, Variables, and Keywords
- Operators and Expressions
- Control Statements (if, if-else, switch)
- Loops (for, while, do-while)
- Input and Output (Scanner, BufferedReader)

3. Object-Oriented Programming (OOP) in Java

- Classes and Objects
- Constructors (Default, Parameterized)
- this keyword
- Inheritance (Single, Multilevel, Hierarchical)
- Polymorphism (Method Overloading & Overriding)
- Abstraction (Abstract Classes & Interfaces)
- Encapsulation
- Static & Final Keyword

4. Packages and Access Modifiers

- Built-in Packages (java.lang, java.util)
- User-defined Packages
- import Statement
- Access Specifiers: private, protected, public, default

5. Arrays and Strings

- Single and Multidimensional Arrays
- String Class and StringBuffer
- StringBuilder and String Methods
- String Comparison and Immutability

6. Exception Handling

- Try, Catch, Finally
- Throw and Throws
- Custom Exceptions
- Checked vs Unchecked Exceptions

7. Java Collections Framework

- List, Set, Map Interfaces
- ArrayList, LinkedList, HashSet, TreeSet
- HashMap, LinkedHashMap, TreeMap
- Iterator and ListIterator
- Comparable and Comparator

8. Multithreading and Concurrency

- Thread Class and Runnable Interface
- Thread Life Cycle
- Thread Synchronization
- Inter-thread Communication

9. File I/O (Java IO and NIO)

- File, FileReader, FileWriter
- BufferedReader, BufferedWriter
- Serialization and Deserialization
- Java NIO Basics (Path, Files, Channels, ByteBuffer)

Advanced Java

10. GUI Programming with Swing

- Introduction to AWT and Swing
- JFrame, JPanel, JButton, JLabel, etc.
- Layout Managers
- Event Handling

11. Java Database Connectivity (JDBC)

- JDBC Architecture
- Connecting to Database (MySQL/Oracle)

- Statement, PreparedStatement, ResultSet
- Transactions and Batch Processing

12. Java Servlets

- Servlet Life Cycle
- doGet() and doPost() methods
- Request and Response Objects
- Session Management
- ServletConfig and ServletContext

13. Java Server Pages (JSP)

- JSP Lifecycle
- JSP Scripting Elements
- JSP Directives and Implicit Objects
- JSTL and Custom Tags

14. JavaBeans

- What are JavaBeans?
- Properties, Getters, Setters
- Introspection and Serialization

15. MVC Architecture

- Model-View-Controller Pattern
- Building a Simple MVC Web Application
- Integration with JSP & Servlets

16. Introduction to Spring Framework (Optional but useful)

- Overview of Spring Core (IOC, DI)
- Spring MVC Basics
- Spring Boot Introduction

17. Build Tools and Version Control

- Apache Maven or Gradle
- Git and GitHub Basics

18. Unit Testing in Java

- JUnit 5 Basics
- Writing and Running Test Cases
- Mockito for Mocking

19. Design Patterns in Java (Singleton, Factory, Observer, etc.)

20. Java 8+ Features: Lambda Expressions, Streams, Functional Interfaces, Optional, Method References, etc.

Project

The goal of this project is to bridge the gap between conceptual understanding and real-world application. Starting from core Java fundamentals such as object-oriented programming, file handling, and collections, and progressing to advanced topics including JDBC, Servlets, JSP, and web-based application development, each project has been carefully selected to reinforce essential skills and encourage hands-on experience.

Total Duration of the Course

- **Total duration:** 2 months.
- **Days:** Monday to Friday (1 hr daily)
- **Mode:** Online

Certificate

- A course completion certificate will be awarded once you have successfully completed this course.

Wishing you a **Smooth Coding** ahead!