

Java Complete Training Course (For Absolute Beginners)

Beginner Level

Module 1: Introduction to Programming & Java Setup Topics:

- **What is Programming?**
 - What is a Program?
 - How Software Changes the World
 - **Introduction to Java:**
 - Java History, Versions, and Use Cases
 - Java's WORA (Write Once, Run Anywhere) Philosophy
 - **Understanding JDK, JRE, and JVM**
 - **Installing Java and IDE Setup**
 - Install JDK
 - Install and Configure IntelliJ IDEA / Eclipse
 - **Your First Java Program**
 - Structure: Class, Method, Main
 - How Java Code is Compiled and Run (javac + java)
-

Module 2: Java Fundamentals: Variables, Data Types, and Operators

Topics:

- **Variables:**
 - Declaring and Initializing Variables
 - Constants (`final` keyword)
 - **Primitive Data Types:**
 - int, float, double, boolean, char, byte, short, long
 - **Type Casting:**
 - Implicit and Explicit Casting
 - **Operators:**
 - Arithmetic, Relational, Logical, Bitwise, Assignment, Ternary
 - **Expressions and Operator Precedence**
-

Module 3: Input/Output Handling and Flow Control

Topics:

- **Taking User Input:**
 - Using `Scanner` class
- **Output to Console:**
 - `System.out.println`, Formatting Output
- **Control Flow Statements:**
 - if, if-else, if-else-if

- switch-case
 - **Looping Statements:**
 - for, while, do-while
 - **Jump Statements:**
 - break, continue, return
-

Mini-Project : ATM Simulation Console App

- **Description:** Create a simple ATM system with balance checking, withdrawal, deposit options.
 - **Skills:** Input Handling, Conditional Logic, Looping.
-

Module 4: Methods and Modular Programming

Topics:

- **Methods:**
 - Defining and Calling Methods
 - Method Parameters and Return Values
 - Method Overloading
 - **Scope of Variables:**
 - Local, Instance, and Static Scope
 - **Recursion Basics**
-

Module 5: Arrays and Collections Introduction

Topics:

- **Arrays:**
 - One-Dimensional Arrays
 - Two-Dimensional Arrays
 - Common Array Operations (CRUD)
 - **Introduction to Collections:**
 - ArrayList Basics
 - Using Lists vs Arrays
-

Mini-Project : Student Grade Analyzer

- **Description:** Input students' marks, calculate average, highest, and lowest grades.
 - **Skills:** Arrays, Loops, Methods, Conditions.
-

Intermediate Level

Module 6: Object-Oriented Programming (OOP) in Java

Topics:

- **Concepts of OOP:**
 - Abstraction, Encapsulation, Inheritance, Polymorphism
- **Defining Classes and Objects**
- **Fields and Methods**

- Constructors and Overloading Constructors
 - this Keyword Usage
-

Module 7: Advanced OOP Concepts

Topics:

- Inheritance:
 - Single and Multilevel Inheritance
 - Polymorphism:
 - Method Overloading and Overriding
 - Abstract Classes and Methods
 - Interfaces:
 - Defining and Implementing Interfaces
 - Access Modifiers in Detail
 - Final Classes, Methods, and Variables
-

Mini-Project : Library Management System (OOP-based)

- Description: Build a library app to manage book borrowing/returning with classes and inheritance.
 - Skills: OOP Modeling, Inheritance, Interface Implementation.
-

Module 8: Exception Handling

Topics:

- **Types of Exceptions:**
 - Checked and Unchecked
 - **try-catch-finally Blocks**
 - **throw and throws Keyword**
 - **Creating Custom Exceptions**
 - **Nested try-catch Blocks**
-

Module 9: File Handling in Java

Topics:

- **File Reading and Writing:**
 - File, FileReader, BufferedReader
 - FileWriter, BufferedWriter
 - **Handling Directories and Files**
 - **Serialization and Deserialization Basics**
-

Mini-Project : Employee Payroll System (File-Based)

- **Description:** Store and retrieve employee salary data from files.
 - **Skills:** File I/O, Data Persistence, Exception Handling.
-

Advanced Level

Module 10: Java Collections Framework (Deep Dive)

Topics:

- **List Interface:**
 - ArrayList, LinkedList
 - **Set Interface:**
 - HashSet, TreeSet
 - **Map Interface:**
 - HashMap, TreeMap
 - **Iterators and Enhanced For-Loops**
 - **Sorting Collections (Comparable and Comparator Interfaces)**
-

Module 11: Java 8 and Beyond (New Features Overview)

Topics:

- **Lambda Expressions**
 - **Functional Interfaces**
 - **Stream API Basics:**
 - filter(), map(), collect(), sorted()
 - **Optional Class (Avoiding NullPointerException)**
 - **Date and Time API (java.time package)**
-

Module 12: Multithreading and Concurrency

Topics:

- **Introduction to Threads**
 - **Creating Threads:**
 - Extending Thread Class
 - Implementing Runnable Interface
 - **Thread Lifecycle**
 - **Thread Synchronization:**
 - synchronized Keyword
 - Deadlocks and Prevention
 - **Executor Framework Basics**
-

Mini-Project : Multithreaded Bank Simulation

- **Description:** Simulate concurrent bank transactions using Threads.
 - **Skills:** Thread Management, Synchronization.
-

Capstone Projects

Project 1: Student Management System (File + OOP)

- Students Registration and Records Management
 - Save and Load Records from File
-

Project 2: Inventory Management System

- Product Add/Edit/Delete
 - Quantity Update and Report Generation
-

Project 3: Mini Banking Application (Console App)

- Account Creation, Deposit/Withdraw, Balance Checking
- Multithreading for Transaction Simulation