

**Introduction to Object-Oriented Analysis and Design with UML using Java**

**Course Number:** JAV-200
**Duration:** 4 days

**Overview**

This OOAD with UML training teaches learners proven, real-world techniques to address the biggest challenge in software development: delivering quality systems that meet requirements – on time and within budget. This hands-on course focuses on practical skills, giving learners the knowledge and confidence to design well-structured software systems.

**Prerequisites**

Some programming experience is required.

**Materials**

All attendees receive comprehensive courseware covering all topics in the course.

**Software Needed on Each Student PC**

* Windows, macOS, or Linux with at least 8GB RAM
* A recent JDK version installed
* The Java tool the students are likely to use after the class (Eclipse or IntelliJ IDEA)
* Microsoft Access, SQL Server, or another relational database accessible via JDBC or ODBC

**Objectives**

* Define object-oriented programming and its role in software development
* Utilize UML diagrams to model software systems
* Compare iterative and agile software development processes
* Analyze use cases to capture system requirements
* Design and implement class diagrams using design patterns
* Evaluate and select appropriate design patterns for specific problems
* Explain the role of frameworks and tiers in software architecture

**Outline**

* Introduction to OOAD
	+ Intro: Fields of Study
	+ Object Orientation Overview
	+ Object-Oriented Concepts
	+ Stating the Case for Object Orientation
* Unified Modeling Language (UML)
	+ Unified Modeling Language Defined
	+ Static Diagrams: Use Case, Class, Package, Component, Deployment
	+ Dynamic Diagrams: Collaboration, Sequence, State Chart, Activity
* The Software Development Process
	+ Software Development Process Overview
	+ Iterative Processes
	+ Agile Processes
	+ Unified Software Development Process
		- Phases
		- Iterations
		- Disciplines (Workflows)
		- Models
		- Use Case Driven, Architecture Centric, Iterative and Incremental
* The Inception Phase
	+ Initial Planning
	+ Business Modeling
	+ Requirements Overview
* Introduction to Use Cases
	+ Overview
	+ Actors
	+ Use Case Details
	+ Create Initial Use Case Model
* Additional Modeling
	+ Domain Modeling
	+ Discovering Your Types
	+ Technology Modeling
	+ Non-functional Requirements
* Elaboration Phase
	+ Initial Planning
	+ Detailing Use Cases
	+ Elaborating Use Cases
	+ Refining Analysis Model
	+ Architectural Use Cases, Detailing Use Cases, Elaborating Use Cases, Identify Analysis Classes
* Elaboration - Design
	+ Dynamic Modeling
	+ Frameworks and Tiers
	+ OO Design Principles
	+ Collaboration Modeling Object Refinement
* Introduction to Design Patterns
	+ Introduction
	+ Exploring a Simple Pattern - Iterator
	+ Design Pattern Background
	+ Examining Collection Traversal, Examining the Iterator Pattern
* Design Patterns: A More Formal Approach
	+ The Gang of Four Description
	+ The GOF Patterns
	+ Discussing the GOF Patterns
* Moving Deeper Into Patterns
	+ Factory Method Pattern
	+ Strategy Pattern
	+ Decorator Pattern
	+ Template Method Pattern
	+ Using Factory Method, Using Strategy, Considering Decorator
* Conclusion