

Course Title : UML 2.0 **Duration : 2 days**

The Unified Modeling Language (UML) is the industry-standard notation for object-oriented modeling. UML consists of a number of diagrams to help understand, visualize, communicate and specify various aspects of requirements and design decisions to fulfill these requirements.

Program Objectives

This two-day program introduces the participants to UML modeling. It covers the basic principles of using UML for system analysis and design. With the help of small case studies and examples, it provides practical insight into the important UML diagrams.

The focus of this program is UML 2.0 and not the principles of object-oriented analysis and design. If you are looking for an in-depth training program on object-oriented analysis and design, and the principles of good object-oriented design, etc., then you should consider our four-day training program called "Object-Oriented Analysis and Design" (OOAD).

The difference between learning OOAD and learning UML is like the difference between learning music and learning music notation. Mastering music notation does not make one a master musician. Similarly, mastering UML does not make one a master object-oriented designer. This course just teaches you UML 2.0.

Audience

This course is suitable for programmers, designers, and project managers who are involved in a software project involving programming in any object-oriented language, and modeling using UML. The participants should have familiarity with core object-oriented concepts, and have some background in programming in any object-oriented programming language.

Day-wise Break-up

Day	Module	Topic
Day 1	Module 1	Getting Started
	Module 2	Use Case Diagrams
	Module 3	Activity Diagrams
	Module 4	Class Diagrams and Object Diagrams
	Module 5	State Machine Diagrams
Day 2	Module 6	Sequence Diagrams and Communication Diagrams
	Module 7	Component Diagrams
	Module 8	Other UML Diagrams

Course Outline

Module 1: Getting Started

- Introduction to UML
- Overview of commonly used UML tools

Module 2: Use Case Diagrams

- Actors and use cases
- Drawing use case diagrams
- Generalization and other relationships

Module 3: Activity Diagrams

- Drawing simple activity diagrams
- Activity diagrams with partitions
- Send signal, receive signal, time signal
- Sub-activities
- Activity parameters

Module 4: Class Diagrams and Object Diagrams

- Creating classes
- Association, aggregations and composition relationships
- Generalization relationships
- Abstract classes and interfaces
- Dependency relationships
- Drawing object diagrams to illustrate a class diagram

Module 5: State Machine Diagrams

- State modeling
- Events and actions
- Guard conditions

Module 6: Sequence Diagrams and Communication Diagrams

- Understanding classes and their responsibilities
- Drawing sequence diagrams
- Object messages
- Self messages
- Conditions and loops
- Communication diagrams

Module 7: Component Diagrams

- Components and their relationships
- Provided and required interfaces
- Ports

Module 8: Other UML Diagrams

- Deployment Diagrams
- Package Diagrams
- Composite Structure Diagrams
- Timing Diagrams
- Interaction Overview Diagrams