



Object-Oriented Analysis with UML 2.0

Duration: 3 Days

Course Description

Object orientation has become the predominant paradigm for virtually all modern software development. This course builds on basic concepts to apply object-oriented principles to all phases of the software development life cycle, with particular emphasis on analysis and design. Students will participate in case studies and focused exercises. The Unified Modeling Language (UML) is introduced and is used as the common language in all examples and exercises.

Audience

This course is intended for software developers, programmers, and analysts who are familiar with, and experienced in, software development methodologies and who will be using object orientation and UML in upcoming projects.

Prerequisites

The students must have experience in software development and some familiarity with basic concepts of object orientation. It is recommended that the students take an Object Orientation Overview course or has equivalent experience.

Course Outline

Outline

Overview of Object-Oriented Analysis and Design

- Concepts of object orientation
- Object-oriented software development process

Gathering requirements

Structuring and documenting requirements with use cases

- Use cases and actors
- Use case relationships
- Use case diagram

Behavior Analysis

- Documenting behavior with object interaction diagrams
- Sequence diagrams
- Communication diagrams
- Timing diagrams
- Interaction overview diagrams
- Selecting the right diagram type
- Interaction frames

Object activity analysis

- Actions and activities
- Activity diagrams

Finite state analysis

- Identifying object states



Sterlink Training

Object-Oriented Curriculum

Structure Analysis

- Describing the static model in UML
- Discovering classes and objects
- Responsibility identification
- Expanding object knowledge with CRC cards
- Class diagram
- Structuring class diagrams
- Object diagram and scenarios
- Describing class associations in UML
- Aggregation and composition
- Evaluating an object model

- Object states and transitions
- Documenting object states with UML state machine diagrams

Object Design

- Generalization
- Encapsulation
- Interface, types and roles
- Persistent objects

Applying analysis patterns

Model transitions

Best practices and pitfalls

Case study

For more information about this and other offerings

Please contact Sterlink Data Systems at (416) 859-6470 or email us at training@sterlink.ca

Sterlink Data Systems
985 Audley Road South
Ajax, Ontario L1Z 1N6
www.sterlink.ca

