

Object-Oriented Analysis with UML 2.0

Duration: 4 Days

Course Description

Learn how to use Object-Oriented techniques to analyze real-world requirements and to design solutions that are ready to code. Students learn how to identify and design objects, classes, and their relationships to each other, which includes links, associations, and inheritance. A strong emphasis is placed on diagram notation for use cases, class and object representation, links and associations, and object messages. This course utilizes UML 2.0 notation.

Audience

This course is intended for Analysts, designers, and programmers responsible for applying OO techniques in their software engineering projects.

Prerequisites

Familiarity with structured techniques such as functional decomposition is helpful.

Course Outline

Objects

- Objects and Classes
- Responsibilities
- Composite Classes
- Operations and Methods
- Visibility and Scope
- Inheritance

Advanced Objects

- Abstract Classes
- Polymorphism
- Interfaces
- Templates

Classes and Their Relationships

- Associations and Multiplicity
- Qualified Associations
- Association Classes

New Models in UML 2.0

- Composite Structure Diagrams
- Timing Diagrams
- Interaction Overview Diagrams

Use Cases

- Use Cases and Actors
- Include, Extend and Generalization
- Narrative

Process

- Process
- Risk Management
- Test and Reviews
- Refactoring
- The Unified Process, Agile Processes



Sterlink Training

Object-Oriented Curriculum

- Composition and Aggregation

Sequence Diagrams

- Sequence Diagram Notation
- Interaction Frames
- Creating and Destroying Objects
- Synchronous & Asynchronous

Communication Diagrams

- Notation
- Purpose and Use

State Machine Diagrams

- States and their notation
- Transitions and Guards
- Registers and Actions
- Superstates and Substates
- Concurrent States

Activity Diagrams

- Activity Notation
- Decisions and Merges
- Forks and Joins
- Iteration
- Partitions
- Parameters and Pins
- Expansion Regions

Package, Component, and Deployment Diagrams

The Project

Domain Analysis

- The Domain Perspective
- Data Dictionary
- Finding the Objects
- Responsibilities, Collaborators, and Attributes
- CRC Cards
- Class Models
- Use Case Models

Requirements and Specification

- The Goals
- Understand the Problem
- Specify a Solution
- Prototyping

Design of Objects

- Initial Factoring
- Cohesion and Coupling
- Inheritance
- Object Creation
- Interactions, Links and Associations

Refactoring

- Refactoring
- Clues and Cues
- A Few Refactoring Patterns

Best practices and pitfalls

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

