



Design Patterns in Java Software

Duration: 3 Days

Course Description

Design patterns are recurring solutions to common software problems. Through hands-on workshops and focused class discussions, participants master essential skills by exploring, experimenting, and solving problems by using these recurring solutions. Each student receives a handbook consisting of detailed and carefully crafted, modular guides that provide step-by-step instructions, and essential diagrams and practical examples of using Patterns in Java applications.

Warning signs indicating the need for the application of the appropriate pattern, as well as pitfalls to avoid are highlighted.

Audience

Software developers, designers, technical managers and architects interested in a hands-on exploration of Java SE and EE Design Patterns.

Prerequisites

Students should be experienced programmers with a strong Java background. A basic knowledge of distributed computing, (RMI and/or CORBA/Java, Java EE), and a working knowledge of Servlets, JSPs and EJBs is beneficial.

Course Outline

Pattern-oriented Thinking

Fundamental Patterns

- Interface: Encapsulate a system behavior contract
- Marker Interface: Fulfill a contract without specialized behavior
- Delegation: Use supporting object to implement behavior
- Iterator: Handling members of a collection
- Immutable: Create objects that don't change
- Singleton: Handle one-of-a-kind objects

Behavioral Patterns

- Command
- Observer-observable
- State object
- Strategy
- Template method
- Chain of Responsibility

Presentation-Tier Patterns and Strategies

- Page View Strategy
- View Helper Pattern
- Front Controller Pattern
- Composite View Pattern
- Service to Worker Pattern



Sterlink Training

Object-Oriented Curriculum

- Type-Safe Enumerations: Represent objects which have one value from a finite set (pre and post Java 5)
- Monostate: Handle all-the-same objects
- Dispatcher View Pattern
- Best Practices

Business-Tier Patterns

- Business Delegate
- Service Locator
- Transfer Object
- Business Object
- Session Façade

Creational Patterns

- Factory Method
- Abstract Factory

Integration-Tier Patterns

- Data Access Object
- Service Activator

Structural Patterns

- Adapter
- Composite
- Decorator
- Façade
- Proxy
- Flyweight

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

