

# Java Design Patterns

## Description

This Java Patterns course reviews common and emerging patterns specific to Java SDK and EE development. You'll learn the depth and evolution of pattern-based techniques in Java,

In design patterns, the responsibility of each component is identified by role.

The conventions of design pattern documentation make it easier for development teams to communicate their programming intentions and provide a reference point for the entire Java development community.

### Delegates will learn how to

- Distinguish between Java EE 5 and Java EE 6 pattern-based features.
- Implement relevant patterns in each tier of the Java EE environment.
- Re-factor code to improve inter-tier communications.
- Relate pattern-based development to an implementation architecture.
- Apply object-oriented principles and design guidelines.
- Implement well-known patterns to Java-specific code problems.

---

### Reviewing Object-Oriented Principles in Java

- Identify selected design patterns
- Describe how OO principles apply to Java
- Interpret Unified Modeling Language (UML) notation and create UML diagrams
- Describe how OO concepts apply to Java
- List the goals of an OO language

## Reviewing Gang of Four Patterns

- List key behavioral, creational and structural patterns
- Apply the Facade pattern
- Apply the Composite pattern
- Review the Model-View-Controller (MVC) patterns
- Apply the Observer pattern
- Apply the Strategy pattern

## Implementing Patterns in Java

- Describe how patterns, idioms and refactoring differ from each other
- Use implementation patterns designed for Java
- List forces affecting class, state, and behavioral patterns

## Exploring Changes in Java EE Technology

- Describe the design goals of the Java EE model
- Describe improvements in the Java EE 6 model

## Implementing Integration Patterns

- Review Java EE integration changes that apply design patterns
- Describe design patterns for the integration tier
- Identify use cases for applying integration tier patterns

## Implementing Patterns in Business Components

- Describe design patterns for the business tier
- Describe the role of an enterprise bean

## Implementing Infrastructural Patterns in Java EE

- Describe the role of infrastructural Java EE patterns
- Describe the Singleton pattern
- Describe the Bean Locator pattern
- Describe the Resource Binder pattern
- Describe the Service Starter pattern

## Implementing More Infrastructure Patterns

- Describe the Context Holder pattern

- Describe the Dependency Injection Extender pattern
- Describe the Thread Tracker pattern
- Describe how Java EE interceptors work
- Describe the Payload Extractor pattern

### **Exploring Anti-Patterns**

- Describe Business Tier AntiPatterns
- Describe the Law of Leaky Abstractions
- Describe Presentation Tier AntiPatterns
- Describe Integration Tier AntiPatterns
- Define AntiPatterns

### **Selecting Patterns for Architecture**

- Define the roles of architect, designer, and developer
- List guidelines for applying patterns to an architectural solution
- Describe the relationship between design patterns and architecture