

Java Design Patterns Eğitimi

Süre: 4 Gün

Açıklama

Java Design Patterns Eğitimi, Java tasarım kalıpları, UML ve uygulama mimarisi konularını ele alan 4 günlük bir eğitim programıdır.

Nesne yönelimli ilkeler, Gang of Four kalıpları, entegrasyon ve altyapı kalıpları ile anti-pattern konuları incelenir. Sayfadaki Java EE sürüm kapsamını katılım öncesinde teyit edebilirsiniz.

Tasarım modellerinde, her bir bileşenin sorumluluğu rol tarafından belirlenir. Tasarım modeli dokümantasyonunda yer alan kurallar, geliştirme ekiplerinin programlama amaçlarını iletmelerini ve tüm Java geliştirme topluluğu için bir referans noktası sağlamalarını kolaylaştırır.

Bu eğitimde neler öğreneceksiniz?

- Java EE 5 ve Java EE 6'nın model tabanlı özellikleri arasındaki farklar
- Java EE ortamının her katmanında ilgili modelleri uygulama
- Katmanlar arası iletişimi iyileştirmek için kodları yeniden düzenleme
- Model tabanlı geliştirmeyi uygulama mimarisiyle ilişkilendirme
- Nesneye dayalı program geliştirme ilkelerini ve tasarım kurallarını uygulama
- Java'ya özel kodlama sorunlarını çözmek için iyi bilinen modelleri uygulama

Eğitim İçeriği

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

Katılım Koşulları

Katılım koşulları için bu sayfadaki İletişim Formu üzerinden bilgi alabilirsiniz.