



JAVA DESIGN PATTERNS AND UML 2.0 EĞİTİMİ

3 GÜN



Digital Vizyon
Akademi

www.digitalvizyon.net



İçindekiler

Eđitim Hakkında.....	3
Neler Öğreneceksiniz?	3
Ön Koşullar	4
Kimler Katılmalı.....	4
Outline	5

Eđitim Hakkında

Java Design Patterns and UML 2.0 Eđitimi bir eđitim programıdır. Java tasarım desenleri, yazılım geliřtirme s¼recinde yinelenmeli olarak kullanılmaktadır. Bunun yanı sıra, belirli bir sorunu ç¼zme için test edilmiřtir. Ayrıca, dođrulanmıř yapıları ifade etmektedir. UML (Unified Modeling Language) ise, yazılım ve sistem modelleme için bir standart dildir. UML 2.0, UML dillerinin son ve en geliřmiř s¼r¼m¼d¼r.

Java Tasarım Desenleri ve UML 2.0 Eđitimi, yazılım geliřtirme s¼reçlerini anlamak ve etkin bir řekilde yönetmek için hayati önem taşıyan yetenekleri kazandırır. Java tabanlı uygulamaların tasarım ve geliřtirme s¼reçlerini desteklemek için tasarım desenlerinin ve UML 2.0'ın nasıl kullanılacağını öđretir.

Tasarım desenleri, yazılım geliřtirme s¼recinde karřılařılan yaygın problemlere ç¼z¼m sađlar. Bu eđitimde, en sık kullanılan Java tasarım desenlerini tanıyacaksınız. Böylece, hangi durumlarda bu desenlerin uygulanabileceđini öđrenecek ve Java dilini kullanarak bu desenleri uygulayabileceksiniz. Tasarım desenleri, kod tekrarını önler, mod¼lerlik sađlar ve yazılımın bakımını ve geniřletilebilirliđini kolaylařtırır.

UML 2.0 ise, yazılım geliřtirme s¼reçlerini modelleme ve görselleřtirme için bir dil ve standardır. UML, yazılım geliřtirme s¼reçlerini belgeler, görselleřtirir ve paylařır. Bunun yanı sıra, eđitimde, UML 2.0'ın sunduđu çeřitli diyagram tipleri hakkında detaylı bilgi edineceksiniz. Bunlar arasında class, sequence, state, activity, use case gibi diyagramlar bulunur.

Eđitim, teorik konuları ve pratik uygulamaları dengeli bir řekilde sunarak öđrencilere somut bir öđrenme deneyimi sađlar. Ayrıca, öđrencilere örnek projeler sunmaktadır. Böylece öđrenilen konuların pratiđe nasıl d¼k¼leceđini görebilirler.

Bu eđitim ayrıca Java tasarım desenlerinin farklı t¼rlerini (Singleton, Factory, Observer gibi) ve UML 2.0'ın farklı diagram t¼rlerini (use case, class, sequence, state gibi) tanımanızı sađlar. Ayrıca, bu desenleri ve diagramları nasıl kullanabileceđinizi öđrenmenizi amaçlar.

Java Design Patterns and UML 2.0 Eđitimi, Java ile yazılım geliřtirme yapmak isteyen veya bu alanda daha da geliřmek isteyen herkes için uygun bir seçenektir.

Neler Öđreneceksiniz?

Java Design Patterns ve UML 2.0 eđitiminde řunları öđrenebilirsiniz:

- Java programlama dillerinde kullanılan tasarım desenleri: Singleton, Factory, Observer, Decorator, gibi.
- Tasarım desenlerinin nasıl seçileceđi ve uygulanacağını
- UML 2.0 kavramları: Use Case, Class, Object, State, Activity, Sequence, Collaboration, Deployment gibi.
- UML 2.0'ın nasıl kullanılacağı ve yazılım sistemlerinin nasıl modelleneceđini.



- Java ile UML 2.0'in uygulanmasına dair pratik örnekleri öğreneceksiniz.

Bu eğitim, Java programlama dilleri kullanarak yazılım sistemlerinin tasarımı ve modellemesi konularında uzmanlaşmak isteyenler için faydalı olabilir.

Ön Koşullar

Java Design Patterns ve UML 2.0 eğitimi için şu ön koşullar gerekmektedir:

- Temel Java programlama bilgisi: Değişkenler, döngüler, fonksiyonlar, nesne yönelimli programlama gibi konuları anlamanız gerekir.
- Yazılım sistemleri hakkında temel bilgi: Veri yapıları, algoritmalar, veri analizi gibi konuları anlamanız faydalı olabilir.
- UML veya benzer bir görselleştirme dili hakkında temel bilgi: UML'nin nasıl kullanılabileceği ve görselleştirme notasyonları hakkında bilgi sahibi olmanız gerekir.

Bu ön koşulların tamamına sahip olmanız durumunda, Java Design Patterns ve UML 2.0 eğitiminde daha verimli bir şekilde öğrenme fırsatı bulabilirsiniz.

Kimler Katılmalı

Java Design Patterns ve UML 2.0 eğitimine katılmasını önerdiğimiz kişiler şunlardır:

- Java programlama dilleri kullanarak yazılım geliştirme yapanlar
- Tasarım desenleri ve UML 2.0'i kullanarak yazılım sistemlerinin tasarımı ve modellemesini öğrenmek isteyenler
- Yazılım mühendisliği, bilgisayar mühendisliği gibi alanlarda eğitim almış olanlar
- Yazılım sistemlerinin tasarımı ve modellemesi konularında uzmanlaşmak isteyen profesyoneller
- UML ve Java programlama dilleri hakkında bilgi sahibi olmak isteyenler

Bu eğitimi tamamlamanız durumunda, Java programlama dilleri kullanarak yazılım sistemlerinin tasarımı ve modellemesi konularında uzmanlaşma fırsatı bulabilirsiniz.



Outline

- Introducing Modeling and the Software Development Process
- Creating Use Case Diagrams
- Creating Use Case Scenarios and Forms
- Creating Activity Diagrams
- Creating Interaction Diagrams
- Creating State Machine Diagrams
- OOP Principles and Design Patterns
- Interface Patterns: Adapter, Facade, Composite, Bridge
- Responsibility Patterns: Singleton, Observer, Mediator, Proxy, Responsibility, Flyweight
- Construction Patterns: Builder, Factory Method, Abstract Factory, Prototype, Memento
- Operation Patterns: Template Method, State, Strategy, Command, Interpreter
- Extension Patterns: Decorator, Iterator, Visitor