

JavaTürk

Tasarım Kalıpları (Design Patterns) Eğitimi

Bilgi Dokümanı

Eğitmen: Akın Kaldıroğlu, egitim@javaturk.org

Amaç: Tasarım Kalıpları (Design Patterns) eğitiminin amacı, kalıp tabanlı düşünme ve programlama yetkinliği kazandırmaktır. GoF'un tasarım kalıplarını ele almak ve en önemli ve sık kullanılanlarını teorik ve pratik olarak öğrenmek ve öğrenilenleri, katılımcıların yardımıyla onların iş alanlarındaki problemlere uygulamak, eğitimin amaçlarındandır.

Açıklama: Bu eğitim online yani internet üzerinden verilecek, ücretli bir eğitimidir. Eğitimde soyut tasarım yapıları UML ile, kod örnekleri ise Java ile verilecektir. Eğitimde soyut tasarım yapıları UML ile, kod örnekleri ise Java ile verilecektir. Eğitim için Java'yı bilmek zorunlu değildir. Eğitimin nesne yapıları üzerine yoğun soyut düşünme ve tasarlama becerilerine odaklanması ve sunuluş şekli, bu eğitimi Java dışındaki nesne-merkezli dilleri kullananlar için de uygun kılmaktadır. Java, C++, C#, Python gibi farklı nesne merkezli dillerle uğraşanların eğitime katılmaları, eğitimi daha zengin ve eğlenceli yapacaktır.

Bu eğitim hakkında ve benimle ilgili bilgi almak için Tasarım Kalıpları seminerimin [sunumuna](#) ve [videosuna](#) bakabilirsiniz. Eğitim ile ilgili sorularınız için egitim@javaturk.org adresine email gönderebilirsiniz.

Süre: 30 saat, haftada 5 saat, toplam 6 hafta.

Ders Saatleri: Pazartesi günleri 20:30 - 22:30 arasında 2 saat ve Cumartesi günleri 10:00 - 13:00 arasında 3 saat

En Yakın Eğitim Dönemi: 9 Mart 2015 Pazartesi - 18 Nisan 2015 Cumartesi

Ücret:

- Kurumsal katılımcılar için 1,250 TL + KDV (Kurumsal katılımcılara istemeleri halinde fatura verilecektir.)

- Bireysel katılımcılar için 1,000 TL + KDV (Lisans öğrencilerine %10 indirim uygulanacaktır. Lisans öğrencilerinin bu indirimden faydalanabilmeleri için, öğrencilik bilgilerini, öğrenci email adreslerini kullanarak egitim@javaturk.org adresine göndermeleri gereklidir.)

Erken kayıt halinde %10 indirim uygulanacaktır. Erken kayıt 25 Şubat Cuma günü saat 17:00'da bitecektir. Erken kayıttan faydalanmak isteyenlerin bu vakte kadar ödemelerini yapmaları gerekmektedir.

Ödeme bilgisi katılımcılarla paylaşılacaktır.

Elde Edilecek Yetkinlikler: Eğitimde, tasarım kalıbı tabanlı soyut düşünme ve formal olarak tasarım yapabilme yetkinlikleri ile tasarım kalıplarını kullanarak programlama yapabilme yetkinlikleri kazanılacaktır. Eğitim sonrasında katılımcılar, daha etkin nesne-merkezli düşünme, tasarım ve programlama yapabilir durumda olacaklardır.

Hedef Kitle: Yazılım tasarımcıları, yazılım mimarları ve adayları, tecrübeli programcılar.

Katılımcı Profili Gereklilikleri: Katılımcıların bir nesne-merkezli dilde tecrübe sahibi olmaları gereklidir. Temel nesne merkezli yapıların, sınıf (class), arayüz (interface), kalıtım (inheritance), polymorphism, generics vb. konularda sağlam bilgi ve tecrübe sahibi olmak gereklidir. Ayrıca katılımcıların UML'e aşina olmaları gerekmekte birlikte çok temel UML notasyon bilgilendirmesi eğitim sırasında yapılacaktır.

Altyapı Gereklilikleri: Mikrofon, hoparlör ve kameraya sahip bir bilgisayar, internet erişimi ve Chrome veya Firefox tarayıcı, Java, C#, C++ ya da Python gibi nesne-merkezli dillerle geliştirme ve çalıştırma ortamı ve PDF okuyucu.

İngilizce Yeterlilik Düzeyi: Eğitim dili Türkçe'dir. Fakat katılımcıların dersten azami ölçüde yararlanabilmesi için okuma düzeyinde İngilizce'ye sahip olmaları beklenmektedir.

Eğitim Ortamı: Eğitim online olarak Gototraining üzerinden yapılacaktır. Katılmak isteyenler bu adresten ya da <http://www.gototraining.com> adresinde "403-372-252" eğitim id bilgisini girerek kayıt yapabilirler.

Eğitim Malzemeleri: Akın Kaldıroğlu'nun ders notları ve örnek uygulamaları. Katılımcılara elektronik formatta dağıtılacaktır. Örnek uygulamalar Java'da Eclipse projesi olarak verilecektir.

Eğitim Faaliyetleri: Konu anlatımı, örnek uygulamalar üzerinden anlatım, katılımcıların örnek uygulama yapması, okuma ve uygulama ödevleri.

Eğitim öncesi Seviye Tespit Sınavı: İsteğe bağlı.

Eğitim sonrası Seviye Tespit Sınavı: İsteğe bağlı.

Katılım Belgesi: Evet

Kaynaklar:

- [GoF Design Patterns](#)
- [Head First Design Patterns](#)
- [Design Patterns Explained](#)
- [C# Design Patterns Essentials](#)
- [Learning Python Design Patterns](#)
- [Applying UML and Patterns](#)
- [Java Tasarım Şablonları ve Yazılım Mimarileri](#)
- [JavaTürk makaleleri](#)
- Makaleler, web kaynakları vs.

Konu Başlıkları:

- Software complexity and change
- Main characteristics of high quality software: cohesion and coupling
- Object-oriented principles: SOLID and GRASP principles: Single responsibility principle, open-closed principle, inheritance vs. delegation, Liskov substitution principle, interface segregation principle, dependency inversion principle, granularity principle, etc.
- Concept of design patterns and managing complexity and change
- GoF design patterns
- **Creational Patterns:** Singleton, Factory, Abstract Factory, Builder, and Prototype
- **Structural Patterns:** Composite, Decorator, Proxy, Façade, Adaptor, Flyweight, and Bridge
- **Behavioral Patterns:** Strategy, Command, Observer, State, Visitor, Iterator, and Mediator, Template Method, Chain of Responsibility, and Memento

Bu dokümanın en son haline [buradan](http://www.javaturk.org) ulaşabilirsiniz.

www.javaturk.org