

# Kod Analizi ve Yazılım Kalite Eğitimi

Süre: 2 Gün · Eğitim türü: Sanal Sınıf / Online

## Açıklama

**Kod Analizi ve Yazılım Kalite Eğitimi**, yazılım ekiplerinin daha güvenli, sürdürülebilir, ölçülebilir ve bakımı kolay uygulamalar geliştirmelerine yardımcı olmak amacıyla hazırlanmıştır. İki günlük online eğitimde Clean Code prensipleri, statik ve dinamik kod analizi, kod metrikleri, code review ve güvenli yazılım geliştirme uygulamaları bütüncül bir yaklaşımla ele alınır. Katılımcılar; kod kokularını ve yapısal sorunları tespit etmeyi, analiz sonuçlarını doğru yorumlamayı ve kod kalitesi kontrollerini geliştirme süreçlerine dahil etmeyi öğrenir. Kurumsal ekiplerin ortak kodlama standartları oluşturmaya katkı sağlayan eğitim, teknik borcun azaltılması ve yazılım kalitesinin sürekli takip edilebilir hale getirilmesi için uygulanabilir bir çerçeve sunar. **Kimler Katılmalı?**

- Yazılım geliştiriciler
- Yazılım mimarları ve teknik liderler
- Test mühendisleri ve kalite güvence uzmanları
- DevOps ve DevSecOps ekipleri
- Yazılım kalitesi süreçlerinden sorumlu yöneticiler
- Kod inceleme ve kalite standartlarını geliştirmek isteyen profesyoneller

## Eğitim Sonunda Kazanılacak Yetkinlikler

- Clean Code prensiplerini yazılım projelerine uygulama
- Statik ve dinamik kod analizi yöntemlerini ayırt etme
- Kod kokularını ve yapısal problemleri belirleme
- Kod metriklerini ve analiz raporlarını yorumlama
- Etkili code review süreçleri oluşturma
- Teknik borcu görünür ve yönetilebilir hale getirme
- Kalite kontrollerini CI/CD süreçlerine entegre etme
- Kurumsal kodlama standartlarının geliştirilmesine katkıda bulunma

## Eğitim İçeriği

### 1. Yazılım Kalitesine Kurumsal Bakış

- Yazılım kalitesinin proje ve kurum üzerindeki etkisi
- Sürdürülebilir yazılım geliştirme yaklaşımı

- Kalite risklerinin geliştirme sürecinin erken aşamalarında tespit edilmesi
- Kurumsal kodlama standartları ve kalite politikaları
- Yazılım kalitesi, maliyet ve teknik borç ilişkisi

## **2. Clean Code ve Sürdürülebilir Kodlama**

- Clean Code prensipleri
- Okunabilir ve bakımı kolay kod yazma
- Kod tekrarını azaltma yöntemleri
- Anlamlı isimlendirme ve doğru yapılandırma
- Kod kokularının tanımlanması
- Refactoring gereksinimlerinin belirlenmesi

## **3. Statik Kod Analizi**

- Statik kod analizi nedir?
- Kodu çalıştırmadan hata ve kalite sorunlarını tespit etme
- Kod kokuları, yapısal problemler ve olası güvenlik açıkları
- Statik analiz kurallarının ve eşik değerlerinin belirlenmesi
- Analiz raporlarının yorumlanması
- Statik kod analizinin geliştirme süreçlerine entegrasyonu

## **4. Dinamik Kod Analizi ve Çalışma Zamanı Değerlendirmesi**

- Statik ve dinamik kod analizi arasındaki farklar
- Uygulamaların çalışma zamanı davranışlarının incelenmesi
- Performans, kaynak kullanımı ve bellek problemlerinin değerlendirilmesi
- Çalışma zamanı hatalarının ve risklerinin tespit edilmesi
- Bulguların kalite iyileştirme sürecine aktarılması

## **5. Kod Metrikleri ve Kalite Standartları**

- Kod karmaşıklığı ve bağımlılık ölçümleri
- Okunabilirlik ve bakım kolaylığı göstergeleri
- Test edilebilirlik ve yeniden kullanılabilirlik
- Teknik borcun ölçülmesi ve önceliklendirilmesi
- Kurumsal kodlama standartlarının oluşturulması
- Ölçülebilir yazılım kalite hedeflerinin belirlenmesi

## **6. Güvenli Kod Geliştirme**

- Güvenli yazılım geliştirme prensipleri
- Kod seviyesinde güvenlik yaklaşımı
- Yaygın güvenlik hataları ve riskleri

- Güvenlik kontrollerinin yazılım geliştirme yaşam döngüsüne eklenmesi
- Kalite ve güvenlik kontrollerinin birlikte yönetilmesi

## 7. Code Review ve White Box Test Yaklaşımı

- Code review sürecinin amacı ve kapsamı
- Etkili kod inceleme yöntemleri
- Kod inceleme kontrol listelerinin hazırlanması
- White box test yaklaşımı
- Geri bildirim ve düzeltme süreçlerinin yönetilmesi
- Kalite kültürünün yazılım ekibi içinde yaygınlaştırılması

## 8. CI/CD Süreçlerinde Kod Kalitesi Yönetimi

- Sürekli entegrasyon ve kod kalitesi ilişkisi
- Otomatik kalite kontrollerinin CI/CD süreçlerine dahil edilmesi
- Quality gate yaklaşımı
- Kalite takibi ve raporlama
- Erken uyarı mekanizmalarının oluşturulması
- Kurumsal yazılım kalite olgunluğunun artırılması

## Ön Koşullar

Bu eğitimden verimli şekilde yararlanabilmek için katılımcıların temel programlama bilgisine ve yazılım geliştirme süreçleri hakkında genel bilgiye sahip olması önerilir. Belirli bir programlama dili veya kod analiz aracı konusunda ileri düzey deneyim zorunlu değildir.

## Kuruma Özel Kod Analizi ve Yazılım Kalite Eğitimi

Eğitim programı; kurumun kullandığı programlama dilleri, yazılım geliştirme yöntemleri, mevcut kod kalite standartları, proje ihtiyaçları ve ekiplerin deneyim seviyeleri dikkate alınarak özelleştirilebilir. .NET veya Java tabanlı ekipler için örnekler ve uygulamalar kurumun teknoloji altyapısına göre uyarlanabilir. İlgili teknik yetkinlikleri geliştirmek için Kurumsal .NET Eğitimleri ve Kurumsal Java Eğitimleri programlarını inceleyebilirsiniz. Yazılım güvenliğine daha ayrıntılı odaklanmak isteyen ekipler için Güvenli Yazılım Geliştirme Eğitimleri sunulmaktadır. Kod kalite kontrollerini otomatik geliştirme ve dağıtım süreçleriyle birleştirmek için DevOps Eğitimleri, test ve kalite güvence yetkinliklerini geliştirmek için ise ISTQB Test Eğitimleri incelenebilir. BlueMark Academy'nin kurumsal Kod Analizi ve Yazılım Kalite Eğitimi hakkında program, planlama ve teklif bilgisi almak için iletişim formunu doldurabilirsiniz.

### Kod analizi ve yazılım kalite eğitimi nedir?

Kod analizi ve yazılım kalite eğitimi; Clean Code, statik ve dinamik kod analizi, kod metrikleri, code review, güvenli kod geliştirme ve sürekli kalite yönetimi konularını kapsayan kurumsal bir

eğitim programıdır.

### **Kod analizi eğitimi kimler için uygundur?**

Eğitim; yazılım geliştiriciler, yazılım mimarları, teknik liderler, test mühendisleri, kalite güvence uzmanları, DevOps ekipleri ve yazılım kalite süreçlerinden sorumlu yöneticiler için uygundur.

### **Statik kod analizi nedir?**

Statik kod analizi, kaynak kod çalıştırılmadan önce kodun kalite, güvenlik, karmaşıklık ve standartlara uygunluk açısından incelenmesidir. Hataların ve kod kokularının geliştirme sürecinin erken aşamalarında belirlenmesine yardımcı olur.

### **Statik ve dinamik kod analizi arasındaki fark nedir?**

Statik analiz kaynak kodu çalıştırmadan inceler. Dinamik analiz ise uygulamanın çalışma zamanındaki davranışını, performansını, kaynak kullanımını ve ortaya çıkan hataları değerlendirir.

### **Eğitimde Clean Code konuları ele alınıyor mu?**

Evet. Eğitimde okunabilirlik, anlamlı isimlendirme, kod tekrarının azaltılması, kod kokuları, bakım kolaylığı ve sürdürülebilir kodlama gibi temel Clean Code prensipleri ele alınır.

### **Kod Analizi ve Yazılım Kalite Eğitimi kaç gün sürer?**

Eğitim iki gün sürer ve sanal sınıf üzerinden online olarak gerçekleştirilir.

### **Eğitim kuruma özel düzenlenebilir mi?**

Evet. Eğitim içeriği kurumun kullandığı programlama dilleri, geliştirme süreçleri, kalite standartları, proje ihtiyaçları ve katılımcıların deneyim seviyelerine göre özelleştirilebilir.

### **Eğitime katılmak için ön koşul var mı?**

Temel programlama bilgisi ve yazılım geliştirme süreçleri hakkında genel bilgi önerilir. Belirli bir programlama dili veya kod analiz aracında ileri düzey deneyim zorunlu değildir.