

ASP.NET Core MVC



- [Güncel PDF'i İndir](#)
- [En güncel eğitimlerimiz için www.vebende.com.tr](http://www.vebende.com.tr) ziyaret edin

Eğitim Süresi

- **Süre:** 10 gün
- **Ders Süresi:** 50 dakika
- **Eğitim Saati:** 10:00 - 17:00
- Eğitim formatında eğitimler 50 dakika + 10 dakika moladır. 12:00-13:00 saatleri arasında 1 saat yemek arasındaki verilir. Günde toplam 6 saat eğitim verilir. 10 günlük formatta 60 saat eğitim verilmektedir.
- Eğitimler uzaktan eğitim formatında tasarlanmıştır. Her eğitim için teams linkleri gönderilir. Katılımcılar bu linklere girerek eğitimlere katılırlar. Ayrıca farklı remote çalışma araçları da eğitmen tarafından tüm katılımlara sunulur. Katılımcılar bu araçları kullanarak eğitimlere katılırlar.

- Eğitim içeriğinde github ve codespace kullanılır. Katılımcılar bu platformlar üzerinden örnek projeler oluşturur ve eğitimle birlikte eğitimlerde sorulan sorulara ve taleplere uygun içeriğe cevap verir. Katılımcılar bu araçlarla eğitimlerde sorulan sorulara ve taleplere uygun içeriğe cevap verir.
- Eğitim yapay zeka destekli kendi kendine öğrenme formasyonu ile tasarlanmıştır. Katılımcılar eğitim boyunca kendi kendine öğrenme formasyonu ile eğitimlere katılırlar. Bu eğitim formatı sayesinde tüm katılımcılar gelecek tüm yaşamlarında kendilerini güncellemeye devam edebilecekler ve her türlü sorunun karşısında çözüm bulabilecekleri yeteneklere sahip olacaklardır.

İleri Düzey C++ Geliştirme

C++, yüksek performanslı uygulamalar geliştirmek için kullanılan güçlü ve esnek bir programlama dilidir. Sistem programlamasından oyun motorlarına, finansal uygulamalardan bilimsel hesaplamalara kadar geniş bir kullanım alanına sahiptir. Derleyici optimizasyonları sayesinde düşük seviyeli kontrol imkanı sunarken, nesne yönelimli programlama (OOP) ile modülerlik ve yeniden kullanılabilirlik sağlar.

İleri Düzey C++ Geliştirme eğitimi, dilin derinlemesine özelliklerini ele alarak katılımcılara kapsamlı bir yetkinlik kazandırmayı hedefler. Eğitimde, bellek yönetimi, şablon programlama, çoklu iş parçacığı yönetimi ve modern C++ standartları gibi ileri seviye konular işlenir.

Eğitimin temel kazanımları şunlardır:

- **Bellek Yönetimi:** Akıllı işaretçiler (smart pointers), bellek sızıntılarının önlenmesi ve verimli bellek kullanımı.
- **Şablon Programlama:** Generic programlama, şablon metaprogramlama (TMP) ve ileri seviye STL kullanımı.
- **Çoklu İş Parçacığı (Multithreading):** Paralel programlama, eşzamanlılık (concurrency) ve senkronizasyon mekanizmaları.
- **Modern C++ Standartları:** C++11, C++14, C++17 ve C++20 ile gelen yenilikler ve en iyi uygulamalar.
- **Performans Optimizasyonu:** Düşük gecikmeli ve yüksek verimli kod yazımı için optimizasyon teknikleri.

Bu eğitim, C++ dilini profesyonel düzeyde kullanmak isteyen geliştiriciler için kapsamlı bir rehber sunar. Güncel C++ standartlarını ve endüstri en iyi uygulamalarını içeren içerikleriyle, katılımcılara ileri düzey yazılım geliştirme yetkinlikleri kazandırır.

Eğitim Hedefi

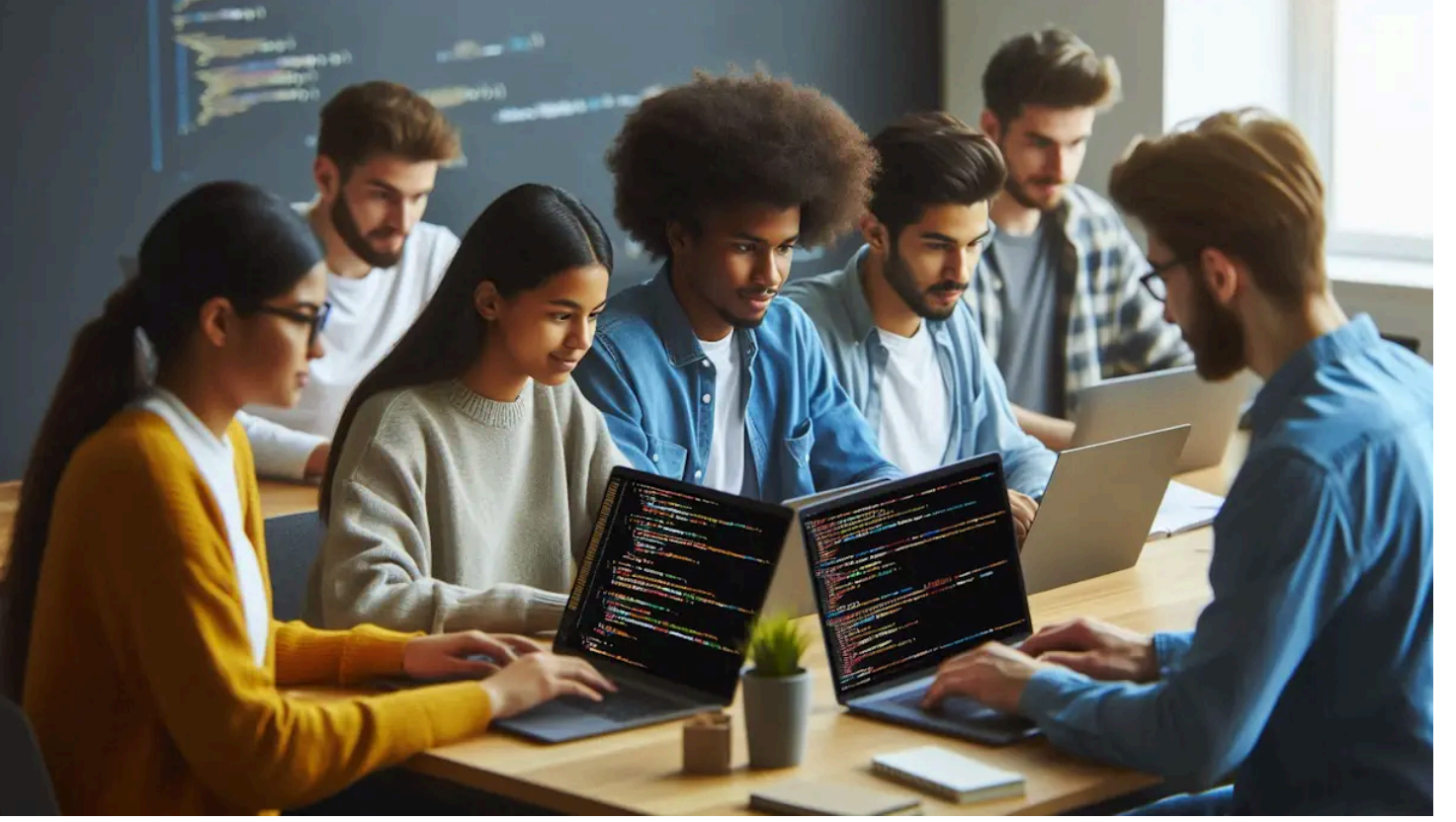
Bu eğitim, **C++ programlama dili** kullanarak yüksek performanslı, güvenilir ve ölçeklenebilir yazılımlar geliştirmeyi öğretmeyi amaçlamaktadır. Katılımcılar, **bellek yönetimi**, **nesne yönelimli programlama (OOP)**, **şablon programlama**, **çoklu iş parçacığı yönetimi**, **modern C++ standartları** ve **performans optimizasyon teknikleri** gibi ileri seviye konulara hakim olacaklardır.

Eğitim sonunda katılımcılar:

- **C++ dilinin temel yapı taşlarını derinlemesine anlayacak,**
- **Bellek yönetimi tekniklerini (RAII, akıllı işaretçiler, manuel bellek yönetimi) etkin şekilde kullanabilecek,**
- **Şablon programlama ve metaprogramlama tekniklerini uygulayabilecek,**
- **Çoklu iş parçacığı ve paralel programlama yöntemleriyle performans artıracak,**
- **Modern C++ (C++11, C++14, C++17, C++20) özelliklerini öğrenerek kodlarını daha verimli hale getirebilecek,**
- **Verimli algoritmalar ve veri yapıları ile optimize edilmiş yazılımlar geliştirebileceklerdir.**

Bu eğitim, **C++'a yeni başlayan geliştiriciler** için olduğu kadar, **mevcut bilgilerini ileri seviyeye taşımak isteyen profesyoneller** için de uygundur.

Eğitim İçeriği



İleri Düzey C++ ile Yüksek Performanslı Yazılım Geliştirme

1. Giriş ve Kurulum

- C++ Nedir? Kullanım Alanları ve Avantajları
- C ile C++ Arasındaki Temel Farklar
- Gerekli Yazılımlar: GCC, Clang, Visual Studio, Code::Blocks Kurulumu
- İlk C++ Programının Yazılması ("Hello, World!")
- C++ Derleyicileri ve IDE Kullanımı

2. Giriş/Çıkış İşlemleri

- cin, cout, cerr, clog Kullanımı
- Formatlı Giriş/Çıkış İşlemleri
- getline() Fonksiyonu ile String Okuma
- Dosya Okuma/Yazma İşlemleri (fstream, ifstream, ofstream)

3. Veri Türleri, Operatörler ve Tip Dönüşümleri

- Temel Veri Türleri (int, float, double, char, bool, string)
- Sabitler ve const Kullanımı
- Operatörler (Aritmetik, Mantıksal, Karşılaştırma, Atama, Bitwise)
- Tip Dönüşümleri (static_cast, dynamic_cast, const_cast, reinterpret_cast)

4. Diziler ve Bellek Yönetimi

- Tek Boyutlu ve Çok Boyutlu Diziler
- Karakter Dizileri (char[] ve string Kullanımı)
- Pointer ve Dizi İlişkisi
- std::vector ile Klasik Dizilerin Karşılaştırılması

5. Kontrol Yapıları

- Koşul İfadeleri: if-else, switch-case
- Döngüler: for, while, do-while
- break, continue ve goto Kullanımı
- Örnek Uygulama: FizzBuzz

6. Pointer, Fonksiyonlar ve Yapılar

- İşaretçiler ve Bellek Yönetimi
- new ve delete Operatörleri ile Dinamik Bellek Yönetimi
- struct ve union Kullanımı
- Call by Value vs. Call by Reference
- Fonksiyonlarda Varsayılan Parametreler ve inline Kullanımı

7. Fonksiyon ve Operatör Aşırı Yükleme

- Fonksiyon Overloading (Aşırı Yükleme)
- *Operatör Aşırı Yükleme (+, -, *, /, [], ())
- Arkadaş (friend) Fonksiyonlar

8. Şablonlar (Templates) ve Standart Şablon Kütüphanesi (STL)

- Şablon Fonksiyonlar (Function Templates)
- Şablon Sınıflar (Class Templates)
- STL ile Şablon Kullanımı

9. Constructor, Destructor ve Kalıtım (Inheritance)

- Constructor ve Destructor Mantığı
- Parametrelili ve Parametresiz Constructor Kullanımı
- Kalıtım Türleri (public, private, protected)
- Base ve Derived Class İlişkileri

10. Polimorfizm, Soyut Sınıflar ve Sanal Fonksiyonlar

- Fonksiyon Override ve Overload Arasındaki Farklar
- Sanal Fonksiyonlar (virtual)
- Soyut Sınıflar ve Saf Sanal Fonksiyonlar (= 0)

11. Hata Yönetimi ve İstisna İşleme (Exception Handling)

- Hata Yönetimi İçin try-catch Kullanımı
- Özel Hata Sınıfları (std::exception, std::runtime_error)
- STL Bileşenleri: Algoritmalar, İteratörler, Yardımcı Fonksiyonlar

12. STL Konteynerleri: Vektörler, Map'ler ve Set'ler

- std::vector, std::list ve std::array Kullanımı
- std::map, std::unordered_map ile Anahtar-Değer İlişkileri
- std::set ve std::multiset ile Veri Saklama
- std::valarray Kullanarak Matematiksel İşlemler

Bu eğitim, **C++ dilini profesyonel seviyede kullanmak isteyen geliştiriciler** için hazırlanmıştır ve **performans optimizasyonu, bellek yönetimi ve modern C++ standartlarına hakim olmayı** hedefler.

Eğitim Yöntemi (C++)

- **Teorik Bilgi:** C++ dilinin temel kavramları ve ileri düzey özellikleri detaylı bir şekilde açıklanacaktır.
- **Uygulamalı Örnekler:** Gerçek dünya senaryoları ve C++'ın güçlü özellikleri ile yazılım geliştirme örnekleri üzerinden eğitim verilecektir.
- **Etkileşimli Tartışmalar:** Katılımcılar, C++ dilinin kullanımı, performans optimizasyonları ve yazılım geliştirme süreçleri hakkında soru-cevap şeklinde tartışmalara katılacaktır.
- **Proje Tabanlı Öğrenme:** Eğitimin sonunda, katılımcılar kendi C++ projelerini geliştirerek öğrendiklerini uygulamalı olarak pekiştirecekleri kapsamlı bir proje çalışması yapacaklardır.

Hedef Kitle

1. **Yazılım Geliştiriciler:** C++ ile yüksek performanslı uygulamalar geliştirmek isteyen yazılımcılar.
2. **Sistem Geliştiriciler:** C++'ı sistem programlama, işletim sistemi geliştirme veya donanım entegrasyonu için kullanmak isteyenler.
3. **Oyun Geliştiriciler:** Oyun motorları ve grafik işleme gibi performans gerektiren uygulamalarda C++ kullanmayı öğrenmek isteyenler.
4. **Veri Bilimcileri ve Bilimsel Hesaplamalarla İlgilenenler:** C++'ın verimli bellek yönetimi ve hızlı hesaplama yeteneklerini kullanarak veri analizi ve bilimsel projeler geliştirmek isteyenler.
5. **Yeni Mezunlar ve Stajyerler:** C++ diline giriş yapmak ve modern yazılım geliştirme süreçlerini öğrenmek isteyenler.
6. **Proje Yöneticileri:** C++ projelerinin yazılım geliştirme süreçlerini nasıl yöneteceklerini öğrenmek isteyen yöneticiler.
7. **Ekip Liderleri:** C++ projelerinde takım yönetimi, kod organizasyonu ve yazılım tasarımı konusunda bilgi sahibi olmak isteyen liderler.
8. **Danışmanlar:** C++ tabanlı yazılım çözümleri geliştiren ve bu alanda danışmanlık yapmak isteyen profesyoneller.
9. **Performans Mühendisleri:** Yazılım performansını optimize etmek isteyen mühendisler ve geliştiriciler.

Katılımcılardan Beklentilerimiz (**ASP.NET Core MVC**)

- Katılımcıların temel seviyede C# ve .NET bilgisine sahip olmaları.
- Web teknolojileri (HTML, CSS, JavaScript) hakkında temel bilgi sahibi olmaları önerilir, ancak zorunlu değildir.
- MVC mimarisi ve web uygulamaları geliştirme süreçlerine ilgi duymaları.
- Aktif katılım, soru sorma ve öğrenmeye açık bir yaklaşım sergilemeleri.
-

Eğitim Materyalleri (Eğitmenlere Özel)