



**İLERİ SEVİYE REACT
EĞİTİMİ
5 GÜN**



Digital Vizyon
Akademi

www.digitalvizyon.net

Eđitim Hakkında

İleri Seviye React Eđitim Programı, modern web uygulamaları geliřtirmede uzmanlařmak isteyen yazılım geliřtiriciler ve front-end mühendisleri için tasarlanmış kapsamlı bir programdır. React, günümüzün en popüler JavaScript kütüphanelerinden biri olup, kullanıcı arayüzleri oluşturmak için güçlü ve esnek bir araç sunar. Bu eđitim, React kütüphanesinin ileri seviye özelliklerini, performans optimizasyon tekniklerini, state yönetimini ve modern web geliřtirme yöntemlerini kapsayarak, katılımcıların derinlemesine bilgi sahibi olmalarını amaçlamaktadır.

Eđitim boyunca, katılımcılar JavaScript bundlers, mikro frontend yapıları, hata ayıklama teknikleri, state yönetimi, TypeScript kullanımı ve clean code prensipleri gibi konularda derinlemesine bilgi edineceklerdir. Program, gerçek dünya senaryoları ve uygulamalı örneklerle desteklenerek, katılımcıların öğrendiklerini pratiđe dökümlerine olanak tanır. Eđitim, beř gün boyunca yoğun bir içerik sunarak, katılımcıların ileri seviye React bilgilerini pekiřtirmelerini ve profesyonel projelerinde kullanabilecekleri beceriler kazanmalarını sağlar.

İlk olarak, JavaScript bundlers ve performans optimizasyonu konularına odaklanılacaktır. Katılımcılar, Rollup, webpack, parcel ve Snowpack gibi popüler JavaScript bundlers'ın performanslarını deđerlendirmeyi, bundle size'ı minimize etmek için etkili yöntemleri ve bileřenlerin bađımlılıklarını yönetmeyi öğreneceklerdir. Bu konular, modern web uygulamalarının performansını artırmak için kritik öneme sahiptir.

Daha sonra, mikro frontend yapısı ve framework bađımsızlık konuları işlenecektir. Katılımcılar, mikro frontend yapısının nasıl oluşturulacađını ve farklı framework'ler kullanarak nasıl bađımsız yapılar geliřtirilebileceđini keřfedeceklerdir. Next.js ve Vite.js gibi araçların kullanımı da bu konular arasında yer alacaktır.

Hata ayıklama ve performans optimizasyonu üzerinde durulacak diđer bir konudur. Katılımcılar, bileřenlerde hata ayıklama araç ve yöntemlerini öğrenip, gereksiz re-render'ların nasıl önleneceđini ve useState ile useEffect gibi React hook'larının nasıl kullanılacađını öğreneceklerdir. Bu konular, geliřtiricilerin daha etkili ve verimli kod yazmalarına yardımcı olacaktır.

State yönetimi ve bileřen optimizasyonu konuları da eđitimde yer alacaktır. Redux, Redux Slice ve Redux Toolkit kullanımı gibi ileri seviye state yönetimi teknikleri, bileřenlerin küçük parçalara ayrılması ve high order components'ların oluşturulması gibi konular ele alınacaktır. useRef ve diđer React hook'larının kullanımı da bu günün önemli konuları arasında yer alacaktır.

Son olarak, TypeScript, testler ve clean code prensipleri işlenecektir. Katılımcılar, TypeScript kullanımı, React 18 ile eklenen yeni hook'lar, frontend unit test yapısı ve WebSocket kullanımı gibi konuları öğreneceklerdir. Clean code yapısının önemi ve nasıl oluşturulacađı da bu günün konuları arasında yer alacaktır. Eđitim, katılımcıların öğrendiklerini pratiđe dökerek geliřtirdikleri uygulamaları sunmaları ve gerçek dünya senaryoları üzerinden problemlerin tartıřılması ile sona erecektir.



Neler Öğreneceksiniz

- JavaScript bundlers (Rollup, webpack, parcel, Snowpack) ve performans değerlendirmesi.
- Bundle size'ı minimize etmek için etkili yöntemler ve bileşen bağımlılıklarını yönetme teknikleri.
- Mikro frontend yapısının oluşturulması ve framework bağımsız yapı geliştirme yöntemleri.
- Next.js ve Vite.js kullanımı.
- Bileşenlerde hata ayıklama araç ve yöntemleri, performans optimizasyonu ve gereksiz re-render'ların önlenmesi.
- useState ve useEffect gibi React hook'larının kullanımı.
- Redux, Redux Slice ve Redux Toolkit kullanımı.
- Bileşenlerin küçük parçalara ayrılması ve high order components'ların oluşturulması.
- useRef kullanımı ve diğer React hook'larının kullanımı.
- TypeScript kullanımı, React 18 ile eklenen yeni hook'lar ve frontend unit test yapısı.
- WebSocket kullanımı ve clean code yapısının önemi.

Kimler Katılmalı

- React konusunda temel bilgiye sahip yazılım geliştiriciler
- Front-end mühendisleri
- Web uygulamaları geliştiren yazılım mühendisleri
- JavaScript geliştiricileri
- Yazılım projelerinde performans ve optimizasyon konularına ilgi duyan profesyoneller
- TypeScript kullanarak daha güvenli ve ölçeklenebilir kod yazmak isteyenler
- Modern web geliştirme yöntemlerini öğrenmek isteyen tüm yazılım meraklıları

Ön Koşullar

- Temel React bilgisi ve deneyimi
- JavaScript konularında ileri seviye bilgi
- Temel HTML ve CSS bilgisi
- Modern web geliştirme araçları (Node.js, npm/yarn) hakkında bilgi
- Git ve versiyon kontrol sistemleri hakkında temel bilgi
- TypeScript konusunda temel bilgi (tercihen)

Eğitim İçeriği

Gün 1: JavaScript Bundlers ve Performans Optimizasyonu

- JavaScript bundlers (Rollup, webpack, parcel, Snowpack) nedir ve performansları nasıl değerlendirilir?
- Bundle size'ı minimize etmek için etkili yöntemler ve örnek çözümler.
- Bileşenlerin bağımlılıkları ve yüklenmeleri için best practice'ler.



Gün 2: Mikro Frontend Yapısı ve Framework Bağımsızlık

- Mikro frontend yapısı nedir ve nasıl oluşturulur?
- Framework bağımsız yapı oluşturulabilir mi? React, Vue, Angular gibi farklı framework'ler kullanarak nasıl yapı oluşturulur?
- Next.js ve Vite.js kullanımının değerlendirilmesi.

Gün 3: Hata Ayıklama ve Performans Optimizasyonu

- Bileşenlerde hata ayıklama araç ve yöntemlerinin kullanımı ve örneklendirilmesi.
- Performans ile ilgili gereksiz re-render'ların önlenmesi.
- useState ve useEffect gibi React hook'ları kullanımı.

Gün 4: State Yönetimi ve Bileşen Optimizasyonu

- Redux, Redux Slice ve Redux Toolkit kullanımı.
- Bileşenlerin küçük parçalara ayrılması ve high order components'ların oluşturulması.
- useRef kullanım alanları ve useEffect gibi React hook'ları kullanım durumları.

Gün 5: TypeScript, Testler ve Clean Code

- TypeScript harici export edilmiş paketlerin kullanılması.
- React 18 ile eklenen hook'ların kullanımları.
- Frontend unit test yapısı ve WebSocket kullanımı.
- Clean code yapısının oluşturulması ve önemi.