React Nedir?

React'in ne olduğu, neden ortaya çıktığı, library ile framework arasındaki fark, SPA kavramı ve React'in nerede kullanıldığı -- sade bir dille, hiç kod yazmadan.

Başlangıç 8 dk

Son güncelleme: 6 Ekim 2026
Çevrimiçi oku: https://www.learnforgex.com/tr/topics/what-is-react

---

React Nedir?

Bu, React kursunun ilk dersi. Burada hiç kod yazmayacağız -- önce React'in ne olduğunu, neden ortaya çıktığını ve nerede kullanıldığını basitçe anlayacağız. Sonraki derste bir React projesi kuracağız, ondan sonrasında gerçek kod yazmaya başlayacağız.

React Nedir?

React, kullanıcı arayüzü (UI) oluşturmak için kullanılan bir JavaScript kütüphanesidir. Facebook (şimdiki adıyla Meta) tarafından geliştirilmiştir.

React ile bir web sayfasını küçük, yeniden kullanılabilir parçalara ("component") bölersin. Örneğin bir e-ticaret sitesinde "ürün kartı", "sepet ikonu", "arama kutusu" gibi her parça ayrı bir component olabilir. Bu parçaları birleştirerek tüm sayfayı oluşturursun.

Neden React?

React'ten önce, bir web sayfasındaki bir şeyi değiştirmek (örneğin bir sayaç artırmak) için genelde elle DOM'a (sayfanın HTML yapısına) dokunman gerekiyordu -- hangi elementi bulacağını, ne değiştireceğini kendin yazıyordun. Sayfa büyüdükçe bu, takip edilmesi zor bir hâl alıyordu.

React'te bunun yerine "veri değişince arayüz ne görünmeli?" diye düşünürsün. Veriyi değiştirirsin, React o veriye göre arayüzü otomatik olarak günceller. Bu, kodu daha basit ve daha az hataya açık hâle getirir.

Tarihçe (Kısa)

React ilk olarak 2013 yılında, Facebook tarafından açık kaynak olarak yayınlandı. O zamandan beri hem Facebook/Instagram gibi büyük şirketlerin hem de küçük projelerin en

çok tercih ettiği arayüz kütüphanelerinden biri oldu. Yıllar içinde en büyük değişim "Hooks" ile geldi (2019) -- bunu, bu kursun ilerleyen bir kategorisinde ("Hooks") detaylıca göreceğiz.

Library vs Framework

React kendini bir "library" (kütüphane) olarak tanımlar, "framework" değil. Aradaki fark basit:

- Bir **framework** (örneğin Angular), sana route yönetimi, form yönetimi, HTTP istekleri gibi birçok şeyi hazır ve kendi kurallarıyla sunar. Onun kurallarına göre çalışırsın.
- Bir **library** (React), yalnızca "arayüzü nasıl oluşturursun" sorusuna odaklanır. Routing, form yönetimi gibi diğer ihtiyaçlar için kendi seçtiğin ayrı kütüphaneleri (örneğin React Router) eklersin.

Bu, React'i daha esnek ama aynı zamanda başta biraz daha fazla karar almanı gerektiren bir araç yapar.

React vs Vanilla JavaScript

"Vanilla JavaScript", hiçbir kütüphane kullanmadan, saf JavaScript ile yazılan koda denir. Küçük bir sayfa için vanilla JavaScript yeterli olabilir. Ama sayfa büyüdükçe, hangi elementin ne zaman güncelleneceğini elle takip etmek zorlaşır.

React, bu takibi senin yerine yapar: sen "arayüz şu anda böyle görünmeli" dersin, React hangi kısmın değiştiğini kendisi bulup günceller. Bunu gelecek derslerde ("State ve Event'ler" kategorisinde) canlı örneklerle göreceğiz.

SPA (Single Page Application) Nedir?

React ile yapılan uygulamalar genelde bir "SPA" (Single Page Application, Tek Sayfa Uygulaması) şeklinde çalışır. Bunun anlamı: kullanıcı bir bağlantıya tıkladığında, tarayıcı sayfayı **baştan yeniden yüklemez**. Bunun yerine, JavaScript sayfanın yalnızca gereken kısmını günceller.

Bu, uygulamayı daha hızlı ve daha akıcı hissettirir -- sayfa geçişlerinde o kısa "yenileniyor" beyaz ekranını görmezsin. "Routing" dersinde (React Router ile) bunu gerçek bir örnekle kuracağız.

React Nerede Kullanılır?

React yalnızca web sayfaları için değil. Aynı temel fikirlerle:

- **Web:** Normal web siteleri ve web uygulamaları (bu kursun odak noktası).
- **Mobil:** React Native ile iOS/Android uygulamaları.
- **Masaüstü:** Electron gibi araçlarla masaüstü uygulamaları.

Bu kursta yalnızca web tarafına, yani normal React'e odaklanacağız.

Özet ve Terimler Sözlüğü

React, arayüzleri küçük component'lere bölerek oluşturmanı sağlayan bir JavaScript kütüphanesi. "Veri değişince arayüz otomatik güncellensin" fikrine dayanır. Bir framework değil, bir library'dir -- yani yalnızca arayüz katmanına odaklanır, geri kalanı (routing gibi) senin seçimine bırakır.

Terimler Sözlüğü

React — Kullanıcı arayüzü oluşturmak için kullanılan bir JavaScript kütüphanesi.

Component — Bir arayüzün yeniden kullanılabilir, küçük bir parçası (örneğin bir buton, bir kart).

Library (Kütüphane) — Belirli bir işi (React'te: arayüz oluşturma) yapmana yardımcı olan, ama genel kuralları sana bırakan bir araç.

Framework — Kendi kurallarını ve yapısını dayatan, daha kapsamlı bir araç (örneğin Angular).

SPA (Single Page Application) — Sayfa geçişlerinde tarayıcının tamamen yeniden yüklenmediği, JavaScript'in yalnızca gereken kısmı güncellediği uygulama türü.