Components

Component nedir, fonksiyon olarak nasıl yazılır, nasıl kullanılır (render edilir), isimlendirme kuralı ve yeniden kullanılabilirlik -- basit örneklerle.

Başlangıç 8 dk

Son güncelleme: 18 Ağustos 2026
Çevrimiçi oku: https://www.learnforgex.com/tr/topics/components

---

Components

JSX dersinde React'in söz dizimini gördük. Şimdi JSX'i gerçekten neyin içinde kullandığımıza bakıyoruz: **component**'ler. React'te her şey bir component'ten oluşur -- bu ders, bir component'in ne olduğunu ve nasıl yazılıp kullanıldığını basit örneklerle anlatıyor.

Component Nedir?

Bir component, arayüzün küçük, bağımsız bir parçasıdır -- bir buton, bir kart, bir sayfanın tamamı bile bir component olabilir. React Nedir? dersinde bahsettiğimiz gibi, bir React uygulaması aslında birçok küçük component'in bir araya gelmesiyle oluşur.

Fonksiyon Olarak Component Yazmak

En basit hâliyle, bir component, JSX döndüren bir JavaScript fonksiyonudur:

```
// At its simplest, a component is a JavaScript function that returns JSX.
// The difference: its name starts with an UPPERCASE letter -- React treats
// names starting with a lowercase letter as normal HTML tags like "div",
// "h1", and recognizes names starting with an uppercase letter as a
// component you wrote.
function Welcome() {
  return <h1>Hello!</h1>;
}

console.log(Welcome());
```

Bu kadar. Yeni bir sözdizimi ya da özel bir anahtar kelime yok -- sadece normal bir fonksiyon, geriye JSX döndürüyor.

Component'i Kullanmak (Render Etmek)

Bir component'i kullanmak (buna "render etmek" denir), onu JSX içinde bir etiket gibi yazmaktır:

```
// USING (rendering) a component means writing it into JSX like an HTML tag:
// <Welcome />. A component can also be used inside another component --
// just like App uses Welcome here.
function Welcome() {
  return <h1>Hello!</h1>;
}

function App() {
  return (
    <div>
      <Welcome />
      <p>This is my first page built with React.</p>
    </div>
  );
}

console.log(App());
```

`<Welcome />`, tıpkı `<h1>` ya da `<div>` yazmak gibi -- tek fark, bunun senin yazdığın bir component olması. Bir component, başka bir component'in içinde de kullanılabilir; App'in Welcome'ı kullanması gibi.

Component İsimlendirme Kuralı

React, component'i normal bir HTML etiketinden nasıl ayırt eder? Cevap basit: isim büyük mü küçük mü diye bakarak.

```
// React tells components and normal HTML tags apart by their capitalization:
// - Starts with a lowercase letter ("div", "button") -> a normal HTML tag.
// - Starts with an uppercase letter ("Welcome", "UserCard") -> your component.

function userCard() {
  // Because this function starts with a lowercase letter, even if you write
  // <userCard /> in JSX, React will try to interpret it as an unknown HTML
  // tag rather than as a component -- this is why component names must
  // ALWAYS start with an uppercase letter.
  return <div>Incorrect naming example</div>;
}

function UserCard() {
  // This is correct: a name that starts with an uppercase letter.
  return <div>Correct naming example</div>;
}

console.log(userCard(), UserCard());
```

Küçük harfle başlayan bir isim (`div`, `button`, `userCard`), React tarafından normal bir HTML etiketi sanılır. Bu yüzden component isimleri her zaman büyük harfle başlamalıdır (`Welcome`, `UserCard`). Bunu unutmak, bu dersteki en yaygın hatalardan biri.

Yeniden Kullanılabilir Component'ler

Bir component'in en büyük faydası: bir kez yaz, istediğin kadar kullan.

```
// The biggest benefit of a component: write it ONCE, use it AS MANY TIMES
// AS YOU WANT. We can reuse the same Button component in different places
// over and over -- no need to rewrite the same HTML/CSS each time.
function Button() {
  return <button>Click</button>;
}

function Toolbar() {
  return (
    <div>
      <Button />
      <Button />
      <Button />
    </div>
  );
}

console.log(Toolbar());
```

Aynı `Button` component'i üç kez kullanıldı -- her seferinde aynı HTML'i yeniden yazmaya gerek kalmadı. Bir sonraki derste (Props), aynı component'i her kullanımda FARKLI verilerle nasıl özelleştireceğimizi göreceğiz.

Özet ve Terimler Sözlüğü

Bir component, JSX döndüren bir fonksiyondur. İsmi büyük harfle başlamak zorundadır, JSX içinde bir etiket gibi kullanılır, ve bir kez yazılıp istendiği kadar tekrar kullanılabilir.

Terimler Sözlüğü

Component — Arayüzün küçük, bağımsız bir parçasını tanımlayan, JSX döndüren bir JavaScript fonksiyonu.

Render Etmek — Bir component'i JSX içinde kullanmak (`<Welcome />` gibi).

Function Component — Bir fonksiyon olarak yazılan component (bugünkü React'te en yaygın component türü).