react nedir

React, Facebook tarafından geliştirilmiş bir JavaScript kütüphanesidir. Kullanıcı arayüzlerini oluşturmak için kullanılır ve özellikle tek sayfa uygulamaları (SPA) oluşturmak için tercih edilir.

React, kullanıcı arayüzünü birbirinden bağımsız bileşenler şeklinde tasarlamayı sağlayan bileşen tabanlı bir yapıya sahiptir. Bu bileşenler, verileri özelleştirilmiş bir şekilde işleyebilir ve farklı bileşenler arasında veri iletişimini sağlayabilir. Bu sayede, karmaşık kullanıcı arayüzleri kolayca tasarlanabilir ve yönetilebilir.

React, sanal DOM (Document Object Model) yapısını kullanarak performansı artırır. Bu sayede, kullanıcı arayüzünde yapılan herhangi bir değişiklik, önce sanal DOM üzerinde yapılır ve ardından gerçek DOM üzerinde uygulanır. Bu sayede, arayüzde gereksiz yenilemelerin önüne geçilir ve uygulamanın performansı artırılır.

React, açık kaynak kodlu bir projedir ve sürekli olarak geliştirilmektedir. Bu sayede, geniş bir geliştirici topluluğu tarafından desteklenir ve kolayca özelleştirilebilir.

react nasıl kullanılır??

React kullanmak için, öncelikle bir geliştirme ortamı kurmanız gerekiyor. Bu amaçla, Node.js ve npm (Node Package Manager) kurulu olan bir bilgisayara ihtiyacınız olacak. Node.js ve npm'nin yüklü olduğundan emin olduktan sonra, bir metin düzenleyicisi (örneğin, Visual Studio Code) kullanarak React uygulamanızı oluşturabilirsiniz.

1. React Projesi Oluşturma: Yeni bir klasör oluşturun ve terminalden bu klasöre gidin. Ardından, terminalde aşağıdaki komutu çalıştırarak yeni bir React projesi oluşturun:

npx create-react-app my-app

1. Proje Dosyalarını Düzenleme: Proje dosyalarını düzenlemek için metin düzenleyicinizi açın ve "src" klasöründe bulunan "App.js" dosyasını düzenleyin. Bu dosya, uygulamanızın ana bileşeni olacaktır.
2. Uygulamayı Başlatma: React uygulamanızı başlatmak için terminalde aşağıdaki komutu çalıştırın:

npm start

Bu komut, uygulamanızı başlatacak ve tarayıcınızda "[http://localhost:3000](http://localhost:3000/)" adresinde gösterilecektir.

1. Bileşenler Oluşturma: React'ta, kullanıcı arayüzü bileşenleri şeklinde tasarlanır. Yeni bir bileşen oluşturmak için, "src" klasörüne gidin ve yeni bir dosya oluşturun. Dosya adını, bileşen adıyla aynı yapın ve ".js" uzantısı ekleyin. Ardından, bileşeninizi düzenleyin.

Örneğin, "MyComponent.js" adında bir bileşen oluşturmak isterseniz, "src" klasörüne gidin ve "MyComponent.js" adında bir dosya oluşturun. Dosyayı açın ve aşağıdaki kodu ekleyin:

import React from 'react';

function MyComponent() {

return (

<div>

<h1>Merhaba Dünya!</h1>

<p>Bu benim ilk React bileşenim.</p>

</div>

);

}

export default MyComponent;

Bu kod, "MyComponent" adında bir bileşen oluşturacak ve bir başlık ve bir paragraf içerecektir. Ardından, bileşeninizi kullanmak için "App.js" dosyasına gidin ve aşağıdaki kodu ekleyin:

import React from 'react';

import MyComponent from './MyComponent';

function App() {

return (

<div>

<MyComponent />

</div>

);

}

export default App;

Bu kod, "MyComponent" bileşenini "App" bileşenine ekleyecek ve "Merhaba Dünya!" başlığı ve "Bu benim ilk React bileşenim." paragrafını gösterecektir.

Bu şekilde, React kullan

React Framework üzerinde Typescript, HTML ve CSS??

React Framework üzerinde TypeScript, HTML ve CSS kullanmak oldukça kolaydır. React projeleri, TypeScript, HTML ve CSS dosyalarını içerebilecek şekilde yapılandırılabilir.

TypeScript Kullanımı: TypeScript, JavaScript'in bir üst kümesidir ve statik tiplere sahip olması nedeniyle yazılım geliştirme sürecini daha kolay ve güvenli hale getirir. React projelerinde TypeScript kullanmak için, projenizin başlatılmasında "--template typescript" argümanını kullanabilirsiniz.

luaCopy code

npx create-react-app my-app --template typescript

Bu komut, yeni bir React projesi oluşturur ve TypeScript şablonunu kullanır.

HTML ve CSS Kullanımı: React projelerinde HTML ve CSS dosyaları kullanmak, componentlerinizin görünümünü daha ayrıntılı bir şekilde özelleştirmenizi sağlar. HTML dosyalarını JSX (JavaScript Syntax Extension) formatında kullanabilirsiniz. JSX, JavaScript ve HTML arasında bir köprü görevi görür. CSS dosyalarını ise, "styled-components" veya "CSS Modules" gibi kütüphaneler kullanarak bileşenlere uygulayabilirsiniz.

Örneğin, "App.js" dosyanızda bir div bileşeni oluşturup içine HTML kodu ekleyebilirsiniz:

javascriptCopy code

import React from 'react'; import './App.css'; function App() { return ( <div> <h1>Merhaba Dünya!</h1> <p>Bu benim ilk React bileşenim.</p> </div> ); } export default App;

Bu kod, bir başlık ve bir paragraf içeren bir div bileşeni oluşturacaktır. Ayrıca, stil dosyası olarak "App.css" dosyasını da kullanır.

CSS dosyasında, bileşeninize özgü stilleri tanımlayabilirsiniz:

cssCopy code

div { background-color: yellow; color: red; }

Bu kod, div bileşeninin arka plan rengini sarı, metin rengini ise kırmızı olarak ayarlar.

Styled Components Kullanımı: Styled Components, React projelerinde CSS stilini yönetmek için kullanabileceğiniz bir kütüphanedir. Bileşenlerinizin görünümünü CSS kodu yazarak değil, JavaScript kodu yazarak belirleyebilirsiniz.

Örneğin, "App.js" dosyanızda bir bileşen oluşturabilir ve bu bileşene özgü stil kodlarını Styled Components ile belirleyebilirsiniz:

React Framework üzerinde TypeScript, HTML ve CSS kullanmak oldukça kolaydır. React projeleri, TypeScript, HTML ve CSS dosyalarını içerebilecek şekilde yapılandırılabilir.

TypeScript Kullanımı: TypeScript, JavaScript'in bir üst kümesidir ve statik tiplere sahip olması nedeniyle yazılım geliştirme sürecini daha kolay ve güvenli hale getirir. React projelerinde TypeScript kullanmak için, projenizin başlatılmasında "--template typescript" argümanını kullanabilirsiniz.

npx create-renpx create-react-app my-app --template typescriptact-app my-app --template typescript-app my-app --template typescript

Bu komut, yeni bir React projesi oluşturur ve TypeScript şablonunu kullanır.

HTML ve CSS Kullanımı: React projelerinde HTML ve CSS dosyaları kullanmak, componentlerinizin görünümünü daha ayrıntılı bir şekilde özelleştirmenizi sağlar. HTML dosyalarını JSX (JavaScript Syntax Extension) formatında kullanabilirsiniz. JSX, JavaScript ve HTML arasında bir köprü görevi görür. CSS dosyalarını ise, "styled-components" veya "CSS Modules" gibi kütüphaneler kullanarak bileşenlere uygulayabilirsiniz.

Örneğin, "App.js" dosyanızda bir div bileşeni oluşturup içine HTML kodu ekleyebilirsiniz:

import React from 'react';

import './App.css';

function App() {

return (

<div>

<h1>Merhaba Dünya!</h1>

<p>Bu benim ilk React bileşenim.</p>

</div>

);

}

export default App;Bu kod, bir başlık ve bir paragraf içeren bir div bileşeni oluşturacaktır. Ayrıca, stil dosyası olarak "App.css" dosyasını da kullanır.

CSS dosyasında, bileşeninize özgü stilleri tanımlayabilirsiniz:

div {

background-color: yellow;

color: red;

}Bu kod, div bileşeninin arka plan rengini sarı, metin rengini ise kırmızı olarak ayarlar.

Styled Components Kullanımı: Styled Components, React projelerinde CSS stilini yönetmek için kullanabileceğiniz bir kütüphanedir. Bileşenlerinizin görünümünü CSS kodu yazarak değil, JavaScript kodu yazarak belirleyebilirsiniz.

Örneğin, "App.js" dosyanızda bir bileşen oluşturabilir ve bu bileşene özgü stil kodlarını Styled Components ile belirleyebilirsiniz:

javascriptCopy code

import React from 'react';

import styled from 'styled-components';

const StyledDiv = styled.div`

background-color: yellow;

color: red;

`;

function App() {

return (

<StyledDiv>

<h1>Merhaba Dünya!</h1>

<p>Bu benim ilk React bileşenim.</p>

</StyledDiv>

);

}

export default App;Bu kod, "StyledDiv" adında bir bileşen oluşturacak ve bileşene özgü stil kodlarını içerecektir.

Formun Üstü

Regenerate response

Formun Altı

CSS nedir

CSS (Cascading Style Sheets), web sayfalarının tasarımını ve görünümünü belirlemek için kullanılan bir web teknolojisidir. HTML'in aksine, CSS sadece web sayfalarının tasarımı ve görünümüyle ilgilenir, içeriği değil. CSS, web sayfalarındaki yazı tipi, renk, boyut, arka plan, düzen, sınır ve diğer görsel öğelerin tarzını belirlemek için kullanılır.

CSS, web tasarımcılarına ve geliştiricilere web sayfalarını daha kolay ve esnek bir şekilde düzenleme ve tasarlama imkanı verir. CSS ayrıca, bir web sitesindeki tüm sayfaların aynı tarzı paylaşmasına izin verir, bu da sitenin genel görünümünün tutarlı olmasını sağlar.

React ne işe yarar

React, kullanıcı arayüzleri oluşturmak için kullanılan bir JavaScript kütüphanesidir. React, bir web sitesinin veya uygulamanın farklı bölümlerindeki bileşenlerin (component) oluşturulması ve yönetimi için kullanılabilir. Bu sayede uygulamanın herhangi bir kısmı değiştirildiğinde, diğer kısımların etkilenmemesi sağlanır.

React, birçok geliştirici tarafından popüler bir seçim haline gelmiştir, çünkü performansı yüksektir ve bileşenlerin oluşturulması, düzenlenmesi ve yönetimi için çok esnek bir yapı sunar. Ayrıca, React bileşenlerini tekrar kullanılabilir kılarak, kod tekrarını azaltabilir ve yazılım geliştirme sürecini hızlandırabilir.

React, Facebook tarafından geliştirilmiştir ve çok sayıda büyük ölçekli web sitesinde kullanılmaktadır. Ayrıca, birçok topluluk üyesi tarafından desteklenmektedir ve sürekli olarak güncellenmektedir.

Html

HTML (HyperText Markup Language), web sayfalarının yapısını ve içeriğini belirlemek için kullanılan bir web teknolojisidir. HTML, web sayfalarında kullanılan tüm metin, görüntü, bağlantı, video, ses ve diğer içeriklerin düzenlenmesi ve organize edilmesi için kullanılır.

HTML, web sayfalarında kullanılan içeriği, etiketler ve özellikler aracılığıyla yapılandırır. Etiketler, içeriği belirli bir formatta gösterir. Özellikler ise etiketlere özelleştirilmiş özellikler ekler.

Web tarayıcıları, web sayfasını açarken HTML kodunu yorumlar ve sayfayı görüntüler. HTML aynı zamanda, web tarayıcıları tarafından indekslenerek arama motorlarında daha kolay bulunmasına yardımcı olur.

HTML, web tasarımında temel bir bileşen olarak kabul edilir ve web geliştiricileri tarafından kullanılır. Web sayfalarının oluşturulması için en önemli araçlardan biridir.

Veri ne yaapmka istiyoruz tarlasa kullanan state olur :

Hook reoks reaj js api istek cevap