



JAVASCRIPT KULLANARAK REACT'E GİRİŞ EĞİTİMİ 3 GÜN



Digital Vizyon
Akademi

www.digitalvizyon.net



İçindekiler

Eğitim Hakkında.....	3
Neler Öğreneceksiniz?	3
Ön Koşullar	4
Kimler Katılmalı.....	4
Outline	4
Package Manager (npm/Yarn)	4
Compiler Setup	5
ES6/ES2015.....	5
Project Setup (Create React App)	5
Best Practices (Code Organization & Conventions) React Overview	6
Elements	6
JSX.....	6
Virtual DOM Components	6
Props.....	6
Events	6
Hooks.....	7
State.....	7
Side Effects & Lifecycle	7
Lists	8
Component Architecture	8
Forms.....	8
HTTP	8
Routing (React Router)	9
Build & Deploy (as time permits)	9

Eđitim Hakkında

JavaScript kullanarak React'e giriş eğitimi, React framework'üne adım atmak isteyen herkes için mükemmel bir başlangıç noktasıdır. Bu eğitim, web geliştirme deneyimi olmayanların bile React'i öğrenmesine yardımcı olacak şekilde tasarlanmıştır. Eğitim, React'in temel kavramlarını ve JavaScript kullanarak nasıl uygulanabileceğini ayrıntılı bir şekilde ele alır.

Eđitimin ilk bölümünde, React'in temelleri ve neden popüler bir framework olduđu hakkında bilgi veriyoruz. Katılımcılar, React'in neden kullanışlı olduđunu ve neden birçok geliştiricinin tercih ettiđini anlayacaklardır. Ardından, React component yapılarının nasıl oluşturulduđu ve nasıl birleştirildiđi konusunda derinlemesine bir inceleme yapıyoruz. Bu bölümde, JSX (JavaScript XML) sözdizimini de tanıtılır ve React bileşenlerinin nasıl JavaScript koduyla birlikte kullanıldıđı gösterilir.

Eđitimin bir sonraki aşamasında, state ve props kavramları ele alınır. Katılımcılar, React bileşenlerine nasıl state ve props verileri aktarabileceklerini ve bu verileri nasıl güncelleyebileceklerini öğrenirler. Bu bölümde, bileşenlerin yeniden render edilmesini tetikleyen state deđişiklikleri ve props geçişleri gibi konular da ele alınır.

React ile dinamik web uygulamaları geliştirmek için, katılımcılar ayrıca React Router gibi popüler araçları ve Redux gibi state yönetimi kütüphanelerini de öğrenirler. React Router, tek sayfa uygulamalarının yönlendirme işlemlerini kolaylaştırırken, Redux, uygulamanın genel durumunun yönetimine yardımcı olur. Bu ek araçlar, React ile geliştirilen uygulamaların daha verimli olmasına olanak sağlar.

Eđitim, pratik yapma fırsatı da sunar. Ayrıca katılımcılar, öğrendikleri kavramları kullanarak küçük örnek projeler geliştirir ve uygulama yapısı, bileşenlerin birleştirilmesi ve veri yönetimi gibi konuları pekiştirirler. Ayrıca, eğitimin sonunda, katılımcılar tarafından geliştirilen projelerin nasıl test edileceđi ve yayına alınacağı gibi konulara da deđiniyoruz.

JavaScript kullanarak React'e giriş eğitimi, React'i öğrenmek isteyen her seviyeden geliştirici için uygundur. Eğitim, öğrencilere hem teorik bilgi hem de pratik beceriler kazandırır ve React kullanarak web geliştirme kariyerlerine sağlam bir başlangıç yapmalarını sağlar.

Neler Öğreneceksiniz?

JavaScript kullanarak React'e giriş eğitiminde şu konuları öğrenebilirsiniz:

- React Framework'ünün temel kavramları ve kullanımı
- React component yapılarının nasıl oluşturulduđu
- State ve Props gibi React'in temel bileşenlerinin nasıl kullanılabilceđi
- Dinamik web uygulamalarının nasıl geliştirilebileceđi
- React ile ilgili popüler araçlar ve teknolojiler, örneğin React Router ve Redux
- Uygulamanın verilerinin yönetimi ve yapılandırması



- Uygulamanın test etme, geliştirme ve dağıtma süreçleri
- Performans ve güvenlik testleri yapma
- Uygulamanın nasıl dağıtılabileceği.

Bu eğitim, React'e giriş yapmak isteyen ve JavaScript kullanmayı öğrenmek isteyen herkes için faydalı olabilir. Eğitim, uygulamalı çalışma imkanı sunarak teorik bilgilerin pratikte nasıl uygulanabileceğini gösterir.

Ön Koşullar

JavaScript kullanarak React'e giriş eğitimine katılmak için şu ön koşullar gerekmektedir:

- Temel JavaScript bilgisi: Eğitimde kullanılacak olan dillerden biri JavaScript olduğundan, JavaScript'in temel kavramlarını anlamanız gerekmektedir.
- HTML ve CSS bilgisi: React uygulamaları, HTML ve CSS kullanarak tasarım ve düzenleme yapar. Bu dillerin temel kavramlarını anlamanız gerekmektedir.
- Programlama deneyimi: Eğitim boyunca bir uygulama geliştireceksiniz, bu nedenle programlama ile ilgili deneyiminiz olması avantaj sağlayacaktır.

Eğer bu ön koşulları karşılamıyorsanız, öncelikle bu konuları öğrenmeniz ve pratik yapmanız önerilir. Böylece eğitimden en iyi şekilde yararlanabilirsiniz. -reacte-giris-egitimi

Kimler Katılmalı

JavaScript kullanarak React'e giriş eğitimine katılmak isteyen şu kişiler için uygundur:

- Web geliştirmeye ilgi duyan başlangıç seviyesi programcılar
- HTML, CSS ve JavaScript'i temel düzeyde bilenler
- Dinamik ve interaktif web uygulamaları yapmak isteyenler
- React'in temel kavramlarını öğrenmek isteyenler
- JavaScript'i kullanarak uygulamalar geliştirmek isteyenler
- Front-end geliştirme kariyerine adım atmak isteyenler

Eğitim, React'i kullanarak dinamik ve etkileşimli web uygulamaları geliştirmek isteyen herkes için faydalı olacaktır. Başlangıç seviyesi programcılar için de uygun bir eğitimidir, ancak programlama deneyimi olanlar daha hızlı ve verimli bir şekilde eğitimden yararlanabilirler.

Outline

Package Manager (npm/Yarn)

- npm
 - What is js?
 - What is npm?



- Using npm
- npm Scripts

- Yarn
 - Why Yarn?
 - npm Yarn
 - Installation
 - Using Yarn

Compiler Setup

- Setup (Babel)

ES6/ES2015

- Classes
- Scope (var, let, const)
- Arrow Functions
- Modules
- Template Literals
- Default, Rest, Spread
- Default
- Rest
- Spread
- Destructuring
- Optional Parameters
- assign()
- Object Initializer

Project Setup (Create React App)

- Create new Project
- Folder Structure
- Browser Support
- Styles and Assets
- Dependencies



Best Practices (Code Organization & Conventions) React Overview

- Why React?
- What it is?
- Why it is useful?
- Angular, React Compared
- Web application architectures
- Server-side web application architecture
- Single-page web application architecture
- React Architecture

Elements

- Hello World in JavaScript
- Hello World in React

JSX

- Replacing createElement
- Embedding Expressions
- Specifying Attributes

Virtual DOM Components

- Creating an Element
- Create a Function Component
- Rendering a Component
- Creating a Class Component
- Composing & Reuse

Props

- Read-only
- String Literals vs. Expressions
- Function vs. Class Components

Events

- Listening/Subscribing/Wiring to an Event



- In Vanilla JavaScript
- In React: Function Component
- In React: Class Component
- Binding
 - Why Binding is Necessary?
 - Class Method
 - Arrow Function
- Passing Parameters
 - Using Arrow Functions
 - Using Bind
- Handling Events
 - Using Arrow Functions
 - Using Bind
 - Synthetic Events

Hooks

- Definition
- Why Hooks?
- No Breaking Changes
- Hooks API: useState, useEffect, useRef, useContext
- Rules of Hooks

State

- Definition
- State in Function Components
- useState Hook
- State in Class Components
- Mutating state with setState
- Using State Correctly
- Data Flows Down

Side Effects & Lifecycle

- useEffect Hook
- What does useEffect do?
- Using useEffect
- useEffect Demo
- What are Lifecycle Methods
- Understanding Mounting



- Common Less Common Methods
- Using Lifecycle Methods
- Conditional Rendering
- If, else
- Conditional Operator (?)
- Logical (&&) Operator

Lists

- In Vanilla JavaScript: for loop, forEach, array.map
- In React: using Elements, Components
- Why Keys are Needed

Component Architecture

- Reuse
- Component Communication
- Design Patterns
 - Container and Presentation Components
 - Composition Inheritance

Forms

- Controlled Components
- Reuse of Change Logic across Multiple Inputs
- Handling Form Submission
- Controlling Other Form Elements: select, textarea, number
- Validation
- Uncontrolled Components

HTTP

- Axios library
- Fetch API
- Using with React (HTTP GET)
- Refactoring for Reuse
- HTTP POST, PUT, DELETE



Routing (React Router)

- Installation
- Basics
- Handling Not Found (404) (as time permits)
- Parameters (Url & Query) (as time permits)
- Nesting (as time permits)

Build & Deploy (as time permits)

- Building a React Application for Production
- Deploying a React Application
- Serving Apps with Client-Side Routing
- Customizing Environment Variables