



Tecnologia + Conoscenza = Innovazione

 Via Copenhagen, 10 Roma

 www.vivasoft.it

 info@vivasoft.it

Vivasoft

Consulenza & Formazione



Vivasoft è un'azienda leader nel settore della tecnologia, specializzata nell'offrire soluzioni innovative basate su Microsoft. Con anni di esperienza nel mercato, siamo orgogliosi di essere partner certificati Microsoft, impegnati a supportare la trasformazione digitale delle imprese. Una gamma completa di prodotti Microsoft e corsi di formazione altamente qualificata, progettati per aiutare le aziende a ottimizzare i loro processi, migliorare la produttività e acquisire competenze avanzate nel mondo della tecnologia.



Modulo 1: Introduzione a React

Cos'è React e perché usarlo

Installazione e configurazione di un ambiente di sviluppo React con create-react-app

La struttura di una app React: componenti, JSX e virtual DOM

Scrivere il primo componente React

Rendering dei componenti nella pagina web

Concetti base di JSX (JavaScript XML)

Uso di ReactDOM.render

Modulo 2: Fondamenti di React - Componenti e Props

Cos'è un componente React e come funziona

Creazione di componenti funzionali e a classe

Differenza tra componenti funzionali e componenti basati su classe

Passaggio di dati tra componenti con le props

Rendering condizionale dei componenti

Gestione di eventi in React

Modulo 3: Stato e Ciclo di Vita dei Componenti

Introduzione allo state in React

Uso dello state in componenti funzionali con l'hook useState

Ciclo di vita dei componenti e l'uso degli hook useEffect e componentDidMount

Gestione degli effetti collaterali con useEffect

Esercizi pratici di gestione dello stato e degli effetti collaterali



Modulo 4: Event Handling e Form in React

Gestione degli eventi in React (click, input, submit, etc.)
Passaggio di dati tra componenti con le props e state
Gestire i form in React: creazione di form e gestione delle informazioni inserite
Validazione dei dati del modulo
Invio dei dati di un modulo con le API
Costruzione di un'applicazione di gestione dei form

Modulo 5: Lavorare con le Liste e le Chiavi

Rendering dinamico di liste di elementi in React
Uso di map() per iterare sugli array
Importanza delle chiavi in una lista
Gestire l'aggiunta, modifica e rimozione di elementi da una lista
Componente per la gestione di una lista di elementi dinamici

Modulo 6: Styling in React

Styling inline con oggetti JavaScript
Utilizzare file CSS per i componenti
Uso di librerie di styling come styled-components e emotion
Introduzione a CSS modules in React
Personalizzazione dei componenti con Tailwind CSS



Modulo 7: Gestione dello Stato Globale - Context API e Redux

Introduzione alla Context API di React
Creazione di un contesto globale per condividere lo stato tra più componenti
Introduzione a Redux per la gestione dello stato globale
Azioni, riduttori e store in Redux
Integrare Redux con React e gestione degli effetti collaterali con redux-thunk
Esercizi pratici con la Context API e Redux

Modulo 8: Routing in React con React Router

Cos'è React Router e come funziona
Creazione di percorsi con <Route> e <Link>
Navigazione programmatica con useHistory e useNavigate
Gestione di rotte con parametri dinamici
Autenticazione e gestione delle rotte protette
Rendering di componenti in base al percorso (Lazy loading)

Modulo 9: Consumo delle API in React

Introduzione a fetch() e axios per fare richieste HTTP
Gestire lo stato di caricamento e gli errori
Lavorare con dati JSON e rendering dei risultati
Autenticazione con token (JWT) e gestione delle sessioni
Creare una piccola applicazione per consumare un'API esterna (ad esempio, API di un clima o di un prodotto)



Modulo 10: Testing in React

Introduzione al testing con Jest e React Testing Library
Scrivere test per i componenti React
Simulazione di eventi e interazioni utente
Testare il rendering di componenti e la logica interna
Mocking e test di componenti con API esterne
Scrittura di test per la gestione dello stato e degli effetti collaterali

Modulo 11: Performance e Ottimizzazione

Principi di ottimizzazione in React: renderizzazione condizionale e memoizzazione
Utilizzare `React.memo()` e `useMemo()` per evitare rendering inutili
Ottimizzare le performance delle liste con virtualizzazione
Lazy loading dei componenti con `React.lazy()` e `Suspense`
Analizzare le performance con React DevTools

Modulo 12: Progetto Finale

Pianificazione e progettazione di un'app React completa
Sviluppo del progetto con l'uso di componenti, routing, API e gestione dello stato
Testing e debugging
Ottimizzazione delle performance
Presentazione del progetto finale e discussione