

Updated on 02/01/2025

Sign up

React training

4 days (28 hours)

Presentation

With our React training, fully master the Web maximize your users' experience and improve the fluidity and quality of your Web Apps.

With React training, you'll learn how to exploit one of the JavaScript libraries most widely used by developers worldwide. From basic concepts such as ES6, ESNext and the virtual DOM, to Reactjs and its components, this training course will help you master React.

Learn how to optimize your digital products thanks to memoization and the notion of purity, well as how to manage authentication and contexts.

Our React training course will also enable you to create modern Web Apps efficiently, thanks in particular to best practices in testing and debugging.

Like all our programs, our course is based on the latest version of this library, [ReactJS 19](#).

Objectives

- Master the functionalities of the latest version of React, Meta's JavaScript technology
- Developing web applications with React
- Create complex web components quickly
- SinglePage application with React.JS and TypeScript
- Optimizing the performance of a React application
- Using Redux for status management

Target audience

- Web developers
- Lead Developer

Prerequisites

- Basic knowledge of HTML / CSS / JS

Software requirements

- Machine
 - Minimum 8 GB RAM (16 GB recommended)
 - Windows 11, MacOS or Linux
- Latest version of [React](#) installed
- An IDE such as Visual Studio Code

React training program

Introduction

- Reminders, from ES6 to ESNext
- JavaScript, DOM and VirtualDOM
- Limitations of JavaScript applications
- Ecosystem of modern JavaScript web frameworks
- The new ReactJS vision
- Development environment: IDE, tools and plug-ins

Introducing ReactJS

- Virtual Dom, a choice of performance
- Create-react-app VS Vite
- State components, simple rendering function, performance
- JavaScript or JSX
- Understanding JSX
- Creating view components
- Props initialization
- Understanding the passage of props
- Lifting up state - advantages and disadvantages
- React and CSS: css-in-js, CSS modules or styled-component and styling scope

ESNEXT, ES2015 and TypeScript

- TypeScript: Creation and configuration with ESLint (TSLint)
- Maintain the same code standards and qualities throughout the project
- Setting up husky and lint-staged

Components

- Fragments
- Breaking down an application by component
- Props VS state
- Component Data Flow: information flows between components
- Reusable components: control and transfer of ownership
- Form component control: proprietary solution or use of libraries (Formik, react-hook-form)
- Form validation with yup
- DOM manipulation
- Portals

Hooks

- useState
- useEffect - understanding the life cycle and unsubscribes
- useEffect
- useReducer
- Separate component logic to increase reusability
- Creating a customHook
- useRef

Performance optimization

- Understanding memoization and the notion of purity
- useCallback
- useMemo
- React Memo
- Suspense, in the Components section
- Server Components, in the Components section
- useTransition, in the Hooks section
- useDeferredValue, in the Performance Optimization section

REACT 19 NEW FEATURES

- Support for HTML metadata tags
- Support for <link> style sheets
- <Context> as a provider
- Support for <scripts> async
- Improving the React Compiler
- Server Components
- Actions
- Web Components integration
- Asset Loading
- New hooks: use(), useFormStatus(), useActionState() and useOptimistic()
- Passage de ref en tant que Prop

ROUTING WITH REACT ROUTER

- The SPA principle
- Installation of react-router (v7)
- Route creation
- Route configuration and organization
- The different navigation modes
- Redirection and History
- Page 404
- Data fetching with React-Router
- upgrade from v6 to v7, flag v7_startTransition, imports react-router and react-router/dom

Authentication

- Private Route and Redirection
- Layout creation by user role
- Token persistence via customHook creation

REACT CONTEXT

- Definition and use cases
- Creating multiple contexts
- Using different contexts

COMPONENT LIBRARIES

- Using a SemanticUI/MaterialUI/AntDesign/React-Bootstrap library
- How can I use a library without creating technical debt?

Animation

- React Transition Group
- Animations and Transitions

Internationalization

- i18n integration with React-intl (the most popular library for language management)

REDUX

- Understanding Redux
- Install redux-toolkit
- Actions, creators and dispatchers
- Using providers and component injection to organize a Store
- DataFetching with RTK Query

- Integrating redux into our project
- Redux VS Context? When to use which one?

Companies concerned

This course is aimed at both individuals and companies, large or small, wishing to train their teams in a new advanced computer technology, or to acquire specific business knowledge or modern methods.

Positioning on entry to training

Positioning at the start of training complies with Qualiopi quality criteria. As soon as registration is finalized, the learner receives a self-assessment questionnaire which enables us to assess his or her estimated level of proficiency in different types of technology, as well as his or her expectations and personal objectives for the training to come, within the limits imposed by the selected format. This questionnaire also enables us to anticipate any connection or security difficulties within the company (intra-company or virtual classroom) which could be problematic for the follow-up and smooth running of the training session.

Teaching methods

Practical course: 60% Practical, 40% Theory. Training material distributed in digital format to all participants.

Organization

The course alternates theoretical input from the trainer, supported by examples, brainstorming sessions and group work.

Validation

At the end of the session, a multiple-choice questionnaire verifies the correct acquisition of skills.

Sanction

A certificate will be issued to each trainee who completes the course.