

**React 18**

Vivekanand P V

**Aim:**

* To gain working understanding of the modern component-based front-end design using React.js ecosystem
* To learn the best-practices of front-end design and JavaScript development
* To learn the industry-standard development practices
* To learn state-management using Redux and React Query

**Duration:**

5 days (standard content)

**Prerequisites:**

* Hands-on experience in HTML, CSS, and JavaScript
* Understanding of software engineering principles
* Knowledge of standard design patterns is a bonus

**Essential Reading:**

The aim of essential reading is to bring participants quickly to the pace. Participants are required to go through these quick guides before attending the training.

1. JavaScript basics: <https://developer.mozilla.org/en-US/docs/Learn/Getting_started_with_the_web/JavaScript_basics>
2. DOM (Document Object Model) basics: <https://developer.mozilla.org/en-US/docs/Web/API/Document_Object_Model/Introduction>
3. Arrow functions: <https://developer.mozilla.org/en-US/docs/Web/JavaScript/Reference/Functions/Arrow_functions>
4. Array API: <https://developer.mozilla.org/en-US/docs/Web/JavaScript/Reference/Global_Objects/Array>
5. Promise API: <https://developer.mozilla.org/en-US/docs/Web/JavaScript/Reference/Global_Objects/Promise>

**Lab Setup:**

|  |  |
| --- | --- |
| **Classroom Training** | **Virtual Training** |
| * HD-quality projector or screen with HDMI connectivity * Whiteboard and markers | Access to (one of):   * Zoom Meeting * Skype * Google Meet (Hangout) * Microsoft Teams |

**Environment Setup:**

* Access to Npm/Artifactory
* Node.js v18 and Npm v8 or 9
* Access to GitHub so that participants can clone/download/refer to sample and reference code
* Access to React documentation and other online resources
* Visual Studio Code v1.72+
* Participants should be able to create and run the sample app as per this link: <https://create-react-app.dev/docs/getting-started#quick-start>

**Content**

**Day 1**

**Session 1**

Introduction to component-based front-end development

Project anatomy

React components and composition

Data binding

Embedded expressions

JSX

Component state

Hands-on exercises

**Session 2**

Conditionals

List-rendering

DOM-events

Custom events

Props

One-way binding

Component composition

Content projection

Hands-on exercises

**Day 2**

**Session 1**

Life-cycle methods

Class components

Working with styles

React hooks

Advanced hooks

Render props

Exercises

**Session 2**

Routing

Route props

Route definitions

Parameters in routing

Redirection and navigation

Nested routes

Programmatic navigation

Route organization

Lazy loading

Hands-on exercises

**Day 3**

**Session 1**

Understanding Promise API

Asynchrony in JavaScript

Http communication using Axios

Consuming RESTful webservices with Axios

Hands-on Lab

Next.js core tenets

Using Next.js for static site generation

Using server side rendering

Using incremental static regeneration

**Session 2**

Typical use-case scenarios for SSG, SSR, ISR

Introduction to state-management

Flux architecture

Introduction to Redux workflow

Reducers

Actions and action creators

Redux Toolkit

Slices

Working with redux in function components

**Day 4**

**Session 1**

Component state management techniques

Controlled component technique

Pure and higher order functions

Currying technique

Redux middleware

Context API in React

Hands-on Lab

**Session 2**

Axios instance and abstraction

Http interception

React Query core tenets

Asynchronous store concepts

Hands-on lab

Review of state-management in React

Discussions and Q&A

**Day 5**

**Session 1**

Introduction to component and unit testing

Jest unit testing framework – integration

Typical unit testing scenarios

RTL and component testing

Introduction to forms in React

React Hook Forms – core tenets

Hands-on Lab

**Session 2**

Form state and rendering

Validation and dealing with errors

Nested objects and arrays

Control states

Asynchronous validation

Building React apps

Deployment demonstration (AWS EC2 and S3 static site)

Dockerizing the React apps

Q & A

Further reading

Conclusion