

TRAINING CONTENTS

Module	Session No.	Topics	No. of class hours
Introduction to React.js	1	<p>Module 1: Introduction to React.js</p> <p>Lecture:</p> <ul style="list-style-type: none"> ○ Overview of JavaScript and ES6 features ○ Introduction to React and its core concepts ○ Setting up the development environment <p>Practical:</p> <ul style="list-style-type: none"> ○ Install Node.js and create-react-app ○ Create a simple React application ○ Understanding JSX and rendering elements 	3
	2	<p>Module 1: Introduction to React.js</p> <p>Lecture:</p> <ul style="list-style-type: none"> ○ Components and Props ○ State and Lifecycle methods <p>Practical:</p> <ul style="list-style-type: none"> ○ Create functional and class components ○ Passing data through props 	3
Handling Events	3	<p>Module 2: Handling Events</p> <p>Lecture:</p> <ul style="list-style-type: none"> ○ Handling events in React ○ Conditional rendering and lists <p>Practical:</p> <ul style="list-style-type: none"> ○ Implement event handlers in components ○ Create dynamic lists using map ○ Conditional rendering using ternary operators and logical && 	3
Forms and Controlled Components	4	<p>Module 2: Forms and Controlled Components</p> <p>Lecture:</p> <ul style="list-style-type: none"> ○ Forms and Controlled Components 	3

		<ul style="list-style-type: none"> ○ Lifting State Up <p>Practical:</p> <ul style="list-style-type: none"> ○ Create and manage forms in React ○ Handle form submissions ○ Lift state up to parent components for shared state management 	
State Management with	5	<p>Module 3: State Management with Redux</p> <p>Lecture:</p> <ul style="list-style-type: none"> ○ Introduction to Redux and its core principles ○ Setting up Redux with React ○ <p>Practical:</p> <ul style="list-style-type: none"> ○ Install Redux and React-Redux ○ Create a simple Redux store and actions ○ Connect Redux to React components 	3
	6	<p>Module 3: State Management with Redux</p> <p>Lecture:</p> <ul style="list-style-type: none"> ○ Advanced Redux Concepts ○ Middleware and Redux Thunk <p>Practical:</p> <ul style="list-style-type: none"> ○ Implement middleware in Redux ○ Use Redux Thunk for async actions ○ Integrate API calls within Redux actions 	3
React Router	7	<p>Module 4: React Router</p> <p>Lecture:</p> <ul style="list-style-type: none"> ○ Introduction to client-side routing ○ React Router basics <p>Practical:</p> <ul style="list-style-type: none"> ○ Set up React Router in a project ○ Create routes and navigation 	3

	8	<p>Module 4: React Router</p> <p>Lecture:</p> <ul style="list-style-type: none"> ○ Dynamic routing and route parameters ○ Nested routes and redirects <p>Practical:</p> <ul style="list-style-type: none"> ○ Implement dynamic routes with parameters ○ Create nested routes ○ Use Redirect for programmatic navigation 	3
Styling in React	9	<p>Module 5: Styling in React</p> <p>Lecture:</p> <ul style="list-style-type: none"> ○ CSS Modules and styled-components ○ Responsive design principles <p>Practical:</p> <ul style="list-style-type: none"> ○ Apply styles using CSS Modules ○ Use styled-components for component-level styling ○ Implement responsive design using media queries 	3
	10	<p>Module 5: Styling in React</p> <p>Lecture:</p> <ul style="list-style-type: none"> ○ Material-UI and theming ○ Customizing Material-UI components <p>Practical:</p> <ul style="list-style-type: none"> ○ Install and set up Material-UI ○ Apply themes and custom styles to components ○ Use Material-UI components to build a UI 	3
Performance Optimization	11	<p>Module 6: Performance Optimization</p> <p>Lecture:</p>	3

		<ul style="list-style-type: none"> ○ Performance bottlenecks in React ○ Code-splitting and lazy loading <p>Practical:</p> <ul style="list-style-type: none"> ○ Identify and fix performance issues ○ Implement code-splitting using React.lazy 	
	12	<p>Module 6: Performance Optimization</p> <p>Lecture:</p> <ul style="list-style-type: none"> ○ Using the React Profiler ○ Optimizing network requests <p>Practical:</p> <ul style="list-style-type: none"> ○ Profile a React application using React Profiler ○ Optimize network requests with efficient data fetching ○ Implement caching strategies for API data 	3
Integrating APIs	13	<p>Module 7: Integrating APIs</p> <p>Lecture:</p> <ul style="list-style-type: none"> ○ Fetch API and Axios ○ Handling API responses and errors <p>Practical:</p> <ul style="list-style-type: none"> ○ Set up Axios in a project ○ Fetch data from an API and display it in components ○ Handle loading states and errors 	3
	14	<p>Module 7: Integrating APIs</p> <p>Lecture:</p> <ul style="list-style-type: none"> ○ Authentication and Authorization ○ JWT tokens and secure API calls <p>Practical:</p> <ul style="list-style-type: none"> ○ Implement authentication with JWT tokens ○ Protect routes using private route components ○ Secure API calls with authentication headers 	3

<p>Deployment and CI/CD</p>	<p>15</p>	<p>Module 8: Deployment and CI/CD Lecture:</p> <ul style="list-style-type: none"> ○ Deployment strategies for React apps ○ Introduction to CI/CD pipelines <p>Practical:</p> <ul style="list-style-type: none"> ○ Deploy a React app to a cloud platform ○ Set up a CI/CD pipeline using GitHub Actions ○ Automate testing and deployment processes 	<p>3</p>
	<p>16</p>	<p>Module 8: Deployment and CI/CD Lecture:</p> <ul style="list-style-type: none"> ○ Monitoring and maintaining deployed apps <p>Practical:</p> <ul style="list-style-type: none"> ○ Implement monitoring tools ○ Set up error tracking ○ Maintain and update deployed applications 	<p>3</p>
<p>Animations in React</p>	<p>17</p>	<p>Module 9: Animations in React Lecture:</p> <ul style="list-style-type: none"> ○ Introduction to animations in React ○ Using CSS transitions and animations <p>Practical:</p> <ul style="list-style-type: none"> ○ Implement simple CSS transitions ○ Use keyframe animations for dynamic effects ○ Create a loading spinner 	<p>3</p>
	<p>18</p>	<p>Module 9: Animations in React Lecture:</p> <ul style="list-style-type: none"> ○ Advanced animation techniques ○ React Spring and Framer Motion <p>Practical:</p> <ul style="list-style-type: none"> ○ Integrate React Spring for complex animations 	<p>3</p>

		<ul style="list-style-type: none"> ○ Use Framer Motion for advanced motion effects ○ Create interactive UI elements with animations 	
Context API	19	<p>Module 10: Context API</p> <p>Lecture:</p> <ul style="list-style-type: none"> ○ Introduction to Context API ○ When to use Context vs. Redux <p>Practical:</p> <ul style="list-style-type: none"> ○ Create a Context in React ○ Use Context Provider and Consumer ○ Manage global state with Context API 	3
	20	<p>Module 10: Context API</p> <p>Lecture:</p> <ul style="list-style-type: none"> ○ Advanced Context API patterns ○ Optimizing context updates <p>Practical:</p> <ul style="list-style-type: none"> ○ Implement nested contexts ○ Optimize context performance ○ Use React.useContext for better readability 	3
Error Boundaries	21	<p>Module 11: Error Boundaries</p> <p>Lecture:</p> <ul style="list-style-type: none"> ○ Introduction to Error Boundaries ○ Handling errors in React components <p>Practical:</p> <ul style="list-style-type: none"> ○ Create an Error Boundary component ○ Use Error Boundaries in a React application ○ Display custom error messages 	3
	22	<p>Module 11: Error Boundaries</p> <p>Lecture:</p> <ul style="list-style-type: none"> ○ Advanced error handling techniques ○ Integrating error tracking tools <p>Practical:</p>	3

		<ul style="list-style-type: none"> ○ Implement error logging with Sentry ○ Create a fallback UI for errors ○ Manage error state in components 	
Custom Hooks	23	<p>Module 12: Custom Hooks</p> <p>Lecture:</p> <ul style="list-style-type: none"> ○ Introduction to custom hooks ○ When to create custom hooks <p>Practical:</p> <ul style="list-style-type: none"> ○ Create custom hooks for reusable logic ○ Use custom hooks in components ○ Manage side effects with custom hooks 	3
	24	<p>Module 12: Custom Hooks</p> <p>Lecture:</p> <ul style="list-style-type: none"> ○ Advanced custom hooks ○ Performance optimization with hooks <p>Practical:</p> <ul style="list-style-type: none"> ○ Implement memorization in custom hooks ○ Optimize hooks for better performance ○ Share logic between components with custom hooks 	3
Server-Side Rendering (SSR)	25	<p>Module 13: Server-Side Rendering (SSR)</p> <p>Lecture:</p> <ul style="list-style-type: none"> ○ Introduction to SSR with React ○ Benefits and use cases of SSR <p>Practical:</p> <ul style="list-style-type: none"> ○ Set up a basic SSR project ○ Implement SSR ○ Pre-render pages for better SEO 	3

<p>Security Best Practices</p>	<p>26</p>	<p>Module 14: Security Best Practices Lecture:</p> <ul style="list-style-type: none"> ○ Introduction to security in React ○ Preventing common security issues <p>Practical:</p> <ul style="list-style-type: none"> ○ Implement security measures in React ○ Secure API calls and data storage ○ Use security libraries and tools 	<p>3</p>
	<p>27</p>	<p>Module 14: Security Best Practices Lecture:</p> <ul style="list-style-type: none"> ○ Advanced security techniques ○ Security testing and auditing <p>Practical:</p> <ul style="list-style-type: none"> ○ Perform security testing on a React app ○ Audit security practices ○ Implement advanced security measures 	<p>3</p>
<p>Total Sessions</p>	<p>27</p>	<p>Total class hours</p>	<p>81 hours</p>
<p>Mentorship Sessions</p>	<p>28</p>	<p>CV Writing: Crafting an effective CV, structuring for impact, tailoring CVs for job roles, common mistakes, and ATS optimization</p>	<p>3</p>
	<p>29</p>	<p>Job Interview: Resume-based and behavioral interviews, common interview questions, STAR method, body language, and mock interview practice</p>	<p>4</p>
	<p>30</p>	<p>Industrial Life: Workplace culture, professional ethics, career growth, communication skills, handling work pressure, and networking strategies</p>	<p>3</p>

Total Mentorship Sessions	3	Total mentorship class hours	10