

Complete ReactJS Topics and Practice Questions

Before starting the React concept, below to be cleared.

1. Document Object Model (DOM)

- DOM Tree Structure
- Selecting Elements:
 - getElementById()
 - getElementsByClassName()
 - getElementsByTagName()
 - querySelector()
 - querySelectorAll()
- Modifying Elements:
 - innerHTML, textContent, innerText
 - setAttribute(), getAttribute()
 - classList methods
 - style property
- Creating and Removing Elements:
 - createElement()
 - appendChild(), insertBefore()
 - removeChild(), remove()
- Element Properties and Methods

2. Event Handling

- Event Types:
 - Mouse Events (click, mouseover, mouseout, etc.)
 - Keyboard Events (keydown, keyup, keypress)
 - Form Events (submit, change, input, focus, blur)
 - Window Events (load, resize, scroll)

- `addEventListener()`
- `removeEventListener()`
- Event Object
- Event Propagation (Bubbling and Capturing)
- Event Delegation
- `preventDefault()`
- `stopPropagation()`

3. Browser Object Model (BOM)

- Window Object
- Location Object
- History Object
- Navigator Object
- Screen Object
- Working with Frames

4. Web Storage APIs

- Local Storage (`localStorage`)
- Session Storage (`sessionStorage`)
- Cookies
- Storage Events

5. Testing and Debugging

- Console Methods
- Debugging Tools
- Browser DevTools
- Breakpoints
- Unit Testing
- Testing Frameworks (Jest, Mocha)
- Test-Driven Development (TDD)

6. Performance Optimization

- Code Optimization Techniques
- Memory Management
- Minimizing DOM Manipulation
- Event Delegation
- Lazy Loading
- Code Splitting

Core React Fundamentals (React Start here)

1. React Introduction (0.5 day) | 📺 YouTube: [React JS Intro](#) | 📖 Docs: [docs](#)

- What is React?
- React History
- React Philosophy (Declarative, Component-Based)
- React vs Other Frameworks
- CDN vs Build Tools Setup
- React Features & Advantages Over Other Framework
- SPA

2. JSX Syntax (0.5 day) | 📺 YouTube: [JSX in React - Codevolution](#) | 📖 Docs: [docs](#)

- JX Elements
- Expressions in JSX
- Attributes & Props in JSX
- Fragments
- Comments & Whitespace Rules

3. Components (1 day) | 📺 YouTube: [React Components](#) | 📖 Docs: [docs](#)

- Functional Components
- Class Components (Legacy)
- Component Naming Conventions
- Default & Named Exports
- Organizing Component Files

4. Props Timeline: (0.5 day) | YouTube: [React Props](#) | Docs: [docs](#)

- Passing Props
- Prop Types & Validation
- Default Props
- Prop Drilling
- Children Prop

5. State

- State in Class Components
- useState Hook
- State Initialization
- Updating State Patterns
- Derived State

6. Event Handling

- Handling Events in React
- Synthetic Events
- Event Binding Techniques
- Passing Arguments to Handlers

7. Conditional Rendering

- if / else
- Ternary Operator
- Logical &&
- Guard Clauses

8. Lists & Keys

- Rendering Lists with map()
- Unique Keys
- Key Best Practices

9. Forms

- Controlled Components

- Uncontrolled Components & Refs
- Form Validation Basics
- React Hook Form / Formik (Overview)

10. Component Lifecycle (Class)

- Mounting Phase
- Updating Phase
- Unmounting Phase
- Error Phase

11. React Hooks Basics

- useState
- useEffect
- Rules of Hooks
- Cleaning up Effects

12. Advanced Hooks

- useRef
- useMemo
- useCallback
- useReducer
- useEffect
- useDeferredValue
- useTransition

13. Custom Hooks

- Naming Conventions (use*)
- Sharing Logic via Hooks
- Parameterization & Return Values

14. Context API

- Creating Context
- Provider & Consumer

- useContext Hook
- Avoiding Prop Drilling

15. Performance Optimization

- React.memo
- PureComponent
- Memoization (useMemo / useCallback)
- Lazy Loading & Suspense
- Code Splitting

16. Error Boundaries

- componentDidCatch
- getDerivedStateFromError
- Fallback UI Patterns

17. Portals

- Rendering Outside Parent DOM Tree
- Modal & Tooltip Use-Cases

18. Refs & DOM Access

- Creating Refs
- Forwarding Refs
- useImperativeHandle

19. Routing

- react-router Basics
- BrowserRouter, HashRouter
- Links & NavLink
- Route Params & Query Strings
- Nested & Dynamic Routes
- Redirects & Navigation Guards

20. State Management Libraries

- Redux Core Concepts (Store, Actions, Reducers)
- Redux Toolkit Basics
- Context vs Redux
- MobX Overview
- Recoil / Zustand / Jotai

21. Server State & Data Fetching

- Fetch API / axios
- React Query (TanStack Query)
- SWR
- Caching & Revalidation

22. Asynchronous Patterns

- Promises & Async/Await in React
- useEffect Data Fetching Pattern
- AbortController

23. Code Splitting & Suspense

- React.lazy
- Suspense Fallbacks
- Dynamic Imports

24. Server-Side Rendering & Hydration

- Principles of SSR
- Next.js Pages & App Router
- Data Fetching (getServerSideProps, getStaticProps)
- Streaming & React Server Components

25. Styling in React

- Inline Styles
- CSS Modules
- Styled-Components / Emotion
- Tailwind CSS Integration

- CSS-in-JS Patterns

26. Animation & Gestures

- CSS Transitions
- React Spring
- Framer Motion
- React Transition Group

27. Testing

- Jest Basics
- React Testing Library
- Snapshot Testing
- Enzyme (Legacy)
- Cypress / Playwright E2E

28. Accessibility (a11y)

- ARIA Attributes
- Semantic HTML in JSX
- Focus Management

29. TypeScript with React

- Typing Props & State
- Generics in Components
- Hooks with TypeScript

30. Internationalization (i18n)

- react-i18next Basics
- Formatting & Pluralization

31. Progressive Web Apps

- Service Workers with CRA / Vite
- Offline Strategies

33. DevTools & Debugging

- React Developer Tools
- Profiler Tab
- Browser DevTools Integration

34. Build Tools & Deployment

- CRA vs Vite vs Next.js
- Webpack & Babel Basics
- Environment Variables
- Deploying to Vercel / Netlify / S3

35. Advanced Patterns

- Higher-Order Components (HOC)
- Render Props
- Compound Components
- State Reducer Pattern
- Controlled vs Uncontrolled Patterns

36. New & Upcoming Features

- Concurrent Rendering
- Offscreen Component API
- useSyncExternalStore
- React 19 Actions & Server Components

ReactJS Practice Questions by Category

Basic Practice

1. Build a Counter Component
2. Create a Toggle Switch
3. Implement a Dynamic Greeting using Props
4. Render a List of Users from Static Array
5. Controlled Input Form

6. Conditional Display of Components
7. Build a Simple Accordion
8. Create a Star Rating Component
9. Build a Stopwatch with useState & setInterval
10. Basic Todo List (Add/Delete)

Intermediate Practice

1. Fetch Data from Public API and Render Table
2. Build a Search Filter with Debounce
3. Implement Pagination Component
4. Create a Multi-Step Form with Validation
5. Theme Switcher using Context API
6. Build Image Carousel with Autoplay
7. Implement Infinite Scrolling List
8. Create Custom Hook for Local Storage
9. Implement Drag & Drop Kanban Board
10. Build a Weather Dashboard with React Router

Advanced Practice

1. Build a Redux Toolkit Shopping Cart
2. Implement Authentication Flow with Context + JWT
3. Real-time Chat App using WebSockets
4. Build a Data Grid with Virtualization
5. Implement Optimistic UI Updates with React Query
6. Create a Performance Profiler Dashboard
7. Implement Error Boundary with Logging
8. Create a Reusable Modal using Portals & HOC

9. Build a File Uploader with Progress Bar

Coding Challenges

1. Deep Clone React Tree Utility
2. Implement a Custom usePrevious Hook
3. Build a Debounced Autocomplete Component
4. Implement a Light / Dark Mode Persisted in Local Storage
5. Create a Scroll-to-Top Button with useEffect
6. Build a Dynamic Form Renderer from JSON Schema
7. Implement an Image Lazy Loader using IntersectionObserver
8. Build a Virtualized Chat Window with 10k Messages
9. Create a Calendar Date Picker
10. Implement a Chess Board with Move Validation

Real-World Project Ideas

1. Expense Tracker with Charts
2. Movie Search App using TMDB API
3. E-Commerce Storefront (Redux + Stripe)
4. Kanban Task Manager (Drag & Drop)
5. Blogging Platform with Next.js & Markdown
6. Recipe Finder with Filters
7. Cryptocurrency Price Dashboard
8. Social Media Feed with Infinite Scroll
9. Travel Booking Interface
10. Fitness Progress Tracker