

Front end Developer Interview E-Book

JavaScript Interview Topics

1. Variables & Data Types

- var vs let vs const
- JavaScript Data Types

2. Modern JavaScript Operators

- Ternary Operator (condition ? expr1 : expr2)
- Nullish Coalescing Operator (??)
- Exponential Operator (**)

3. Looping Statements

- for...in
- for...of
- forEach()
- map()

4. Functions

- Fat Arrow Functions (=>)
- Rest Parameter vs Arguments
- Callback Function
- Higher-Order Functions
- Function as First-Class Citizens

5. Object Methods & Properties

- Object.keys()
- Object.values()
- Object.entries()
- Object.freeze() & Object.seal()
- Object.is()
- Object.getOwnPropertyNames()

DOM Methods

- `document.createElement()`
- `document.appendChild()`

6. Arrays & Their Methods

- **Basic Methods:**
 - `push()`, `pop()`, `shift()`, `unshift()`, `slice()`, `splice()`
 - `join()`, `toString()`
- **Searching & Transforming:**
 - `indexOf()`, `at()`, `flat()`, `concat()`
 - `sort()`, `reverse()`, `includes()`, `fill()`
- **Filtering & Reducing:**
 - `filter()`, `find()`, `some()`, `every()`, `reduce()`

Logic-Based JavaScript Questions

1. Finding the First Non-Repeating Character

2. Removing Duplicate Values from an Array

3. Checking for an Empty Array

7. String Methods

- Common string methods (`trim()`, `split()`, `replace()`, etc.)

8. DOM & BOM Objects

- Dialog Boxes: `alert()`, `prompt()`, `confirm()`

9. JavaScript Execution & Event Loop

- **Synchronous vs Asynchronous Execution**
- **Callback Functions & Callback Hell** (Issue & Solution)
- **Promises** (chaining & `.then()`/`.catch()`)
- **Async/Await** (syntactic sugar over promises)

10. Object-Oriented Programming (OOPs) in JavaScript

- Prototyping
- Call, Apply, and Bind Methods

11. Session Management

- Local Storage vs Session Storage

12. Deep Copy vs Shallow Copy

- How JavaScript handles copying objects

13. JavaScript Engine & Working Mechanism

- How JS Works?
 - Event Loop & Microtasks
-

Advanced JavaScript Concepts

1. Currying

2. Event Loop

3. Closures

4. Debouncing & Throttling

5. Hoisting

6. Array/Object Destructuring

7. addEventListener Method

8. setTimeout() vs setInterval()

React Interview Topics

1. Introduction to React

- Why React over Angular?
- Advantages of React

2. React Basics

- JSX
- Babel
- Folder Structure (npx create-react-app myapp)

3. Virtual DOM & Reconciliation

- How Virtual DOM Works?

4. Package Managers

- npm & yarn

5. React Components

- Functional Components (Stateless)
- Class Components (Stateful)
- Lifecycle Methods

6. React Data Flow

- State & Props
- One-Way Data Binding

7. React Routing

- react-router-dom

8. Webpack & Single Page Applications (SPA)

9. React Hooks

- useState()
- useEffect()
- useRef()
- useReducer()
- Context API (useContext()) - Prop Drilling Solution
- Custom Hooks

10. Performance Optimization

- useMemo()
- useCallback()
- Lazy Loading

11. Forms & Components

- Controlled vs Uncontrolled Components

12. Higher-Order Components (HOC)

13. React Lifecycle Phases

- **Mounting:** componentDidMount(), constructor(), render(), getDerivedStateFromProps()
- **Updating:** shouldComponentUpdate(), getSnapshotBeforeUpdate(), componentDidUpdate()
- **Unmounting:** componentWillUnmount()

14. CRUD Operations in React

15. Redux & State Management

- Why Redux?
 - Redux Toolkit
 - `mapStateToProps()` & `mapDispatchToProps()`
-

JavaScript - Additional Topics

1. Execution Context & Scope

- Global Execution Context
- Function Execution Context
- Block Scope vs Function Scope
- Lexical Scope

2. JavaScript Memory Management

- **Garbage Collection**
- **Memory Leaks & How to Avoid Them**

3. WeakMap & WeakSet

- Difference between Map & WeakMap
- WeakMap & WeakSet advantages

4. Generators & Iterators

- `function*` & `yield`

5. Set & Map Objects

- `new Set()` vs `new Map()`
- How they improve performance

6. Symbol Data Type

- Unique identifiers in JavaScript

7. Error Handling in JavaScript

- `try...catch...finally`

- throw statement
- onerror in JavaScript

8. Event Delegation

- How events bubble up and why it's useful

9. JavaScript Modules (ES6)

- import & export
- Named Exports vs Default Exports

10. Web Workers

- Running JavaScript in background threads

11. Fetch API vs Axios

- fetch() basics and error handling
- Difference between fetch() and axios

React - Additional Topics

1. React Fiber Architecture

- How it improves rendering performance

2. Server-Side Rendering (SSR) vs Client-Side Rendering (CSR)

- Next.js for SSR
- React Hydration

3. React Refs (useRef())

- Forwarding Refs
- Handling DOM elements in functional components

4. Compound Components Pattern

- Example of designing reusable components

5. Error Boundaries in React

- componentDidCatch()
- How to prevent UI crashes

6. React Portals

- Rendering components outside of the main DOM tree

7. React Context vs Redux

- When to use Context API instead of Redux

8. Server Components in React (React 18+ Feature)

- Differences from traditional components
-

BONUS - General Web Topics (Good for Interviews)

- HTTP vs HTTPS
- REST API vs GraphQL
- CORS (Cross-Origin Resource Sharing)
- Cookies vs Local Storage vs Session Storage
- Debouncing & Throttling Use Cases
- Performance Optimization