React Interview Questions and Answers

Basic React Questions

1. What is React?

React is a JavaScript library for building user interfaces. It allows developers to create reusable UI components and manage application state efficiently.

2. What are the key features of React?

- Component-Based Architecture
- Virtual DOM
- One-Way Data Binding
- JSX (JavaScript XML)
- Performance Optimization
- Hooks

3. What is JSX?

JSX (JavaScript XML) is a syntax extension for JavaScript that allows writing HTML-like code inside JavaScript.

```
const element = <h1>Hello, World!</h1>;
```

4. What is Virtual DOM in React?

Virtual DOM is a lightweight copy of the real DOM. React updates the Virtual DOM first, compares differences (diffing), and then efficiently updates the real DOM.

5. What is a React Component?

A React component is a reusable piece of UI. It can be a **functional** or **class-based** component.

```
function Welcome() {
  return <h1>Hello, React!</h1>;
}
```

6. Difference between Class and Functional Components?

Feature	Functional Component	Class Component
Syntax	Function-based	Class-based
State	Uses `useState` Hook	Uses `this.state`
Lifecycle Methods	Uses Hooks	Uses lifecycle methods
Performance	Faster	Slightly slower

7. What are Props in React?

Props (Properties) are read-only inputs passed from a parent to a child component.

```
function Greeting(props) {
  return <h1>Hello, {props.name}!</h1>;
}
```

8. What is State in React?

State is an object that stores component-specific data that can change over time.

```
const [count, setCount] = useState(0);
```

9. What is the difference between Props and State?

Feature	Props	State
Mutability	Immutable	Mutable
Scope	Passed from parent to child	Managed within the component
Accessibility	Read-only	Can be modified

10. What is the use of `useState` Hook?

`useState` allows functional components to manage state.

```
const [count, setCount] = useState(0);
```

11. What is `useEffect` Hook?

`useEffect` performs side effects in function components.

```
useEffect(() => {
  console.log("Component Mounted");
}, []);
```

12. What is Conditional Rendering in React?

Conditional rendering is used to render UI based on conditions.

```
{isLoggedIn ? <Dashboard /> : <Login />}
```

13. What is React Router?

React Router is used for navigation in React applications.

```
<Route path="/home" element={<Home />} />
```

14. What are React Fragments?

Fragments allow returning multiple elements without adding extra DOM nodes.

```
<>
  <h1>Hello</h1>
  <p>World</p>
</>
```

15. What is the Context API?

Context API allows passing data deep in the component tree without props.

```
const ThemeContext = createContext("light");
```

Advanced React Questions

51. What is Redux in React?

Redux is a state management library for React applications.

```
const store = createStore(reducer);
```

52. What is Redux Thunk?

Redux Thunk is a middleware for handling async logic in Redux.

53. What is React Query?

React Query is a data-fetching library that helps in caching and synchronization.

54. How to optimize performance in React?

- Use Memoization (`useMemo`)
- Use Lazy Loading (`React.lazy`)
- Avoid Unnecessary Renders (`React.memo`)
- Optimize Lists with `key`

55. What is the difference between `useEffect` and `useLayoutEffect` ?

Feature	useEffect	useLayoutEffect
Execution Time	Runs after render	Runs before paint
Use Case	Data fetching	DOM updates

56. How to handle API requests in React?

Use `fetch` or `axios` .

```
useEffect(() => {
  fetch("https://api.example.com/data")
    .then(response => response.json())
    .then(data => setData(data));
}, []);
```

57. What is Suspense in React?

Suspense lets components wait for data before rendering.

```
<Suspense fallback={<Loading />}>
  <MyComponent />
</Suspense>
```

58. What are React Portals?

Portals allow rendering components outside the DOM hierarchy.

```
ReactDOM.createPortal(<Child />, document.getElementById("modal"));
```

59. What is Server-Side Rendering (SSR) in React?

SSR pre-renders React pages on the server before sending them to the browser.

60. What is Static Site Generation (SSG) in React?

SSG generates static HTML at build time (e.g., Next.js `getStaticProps`).

61. What is Reconciliation in React?

Reconciliation is the process React uses to update the DOM efficiently.

62. What is the difference between controlled and uncontrolled components?

Controlled components use state, whereas uncontrolled components use refs.

63. What is React.memo?

React.memo prevents unnecessary re-renders.

```
export default React.memo(MyComponent);
```

64. What is useCallback in React?

`useCallback` memoizes callback functions.

```
const memoizedCallback = useCallback(() => doSomething(a, b), [a, b]);
```

65. What is useMemo in React?

`useMemo` memoizes computed values.

```
const value = useMemo(() => computeExpensiveValue(a, b), [a, b]);
```

66. How do you handle large lists in React?

Use **React Virtualized** to optimize rendering of large lists.

67. What is the difference between useReducer and useState?

`useReducer` is used for complex state logic, while `useState` is for simple state.

68. How do you handle authentication in React?

Use **JWT, Firebase Auth, or OAuth** for authentication.

69. How do you implement animations in React?

Use **React Spring or Framer Motion**.

```
<motion.div animate={{ x: 100 }} />
```