### **JavaScript and React**

**1. What are JavaScript Promises?** A JavaScript Promise is an object that represents the eventual completion (or failure) of an asynchronous operation and its resulting value.

**2. What is a callback function?** A callback function is a function passed into another function as an argument, which is then invoked inside the outer function to complete some kind of routine or action.

**3. How do you clone an object in JavaScript?** Using Object.assign({}, obj) or the spread operator {...obj} creates a shallow copy. For deep copy, you can use JSON.parse(JSON.stringify(obj)) or a utility like Lodash's cloneDeep.

**4. What is the difference between shallow copy and deep copy?** A shallow copy copies only the first level of properties. A deep copy recursively copies all nested objects.

**5. React components.** React components are the building blocks of a React application. They can be functional or class-based, and they render UI and handle logic.

**6. What is destructuring in JavaScript?** Destructuring is a syntax that allows extracting values from arrays or properties from objects into distinct variables.

**7. What is the use of useRef in React?** useRef is a React Hook that allows you to persist values between renders and directly access or modify DOM elements.

**8. What are setTimeout and setInterval?** setTimeout runs a function once after a delay. setInterval runs a function repeatedly, starting after a delay and repeating at regular intervals.

**9. What is the difference between useState and useEffect hooks in React?** useState manages state in functional components. useEffect performs side effects like data fetching, DOM manipulation, etc.

**10. What is React Router and why is it used?** React Router is a library for routing in React apps. It enables navigation among views and keeps the UI in sync with the URL.

**11. How does the try-catch-finally block work in JavaScript?** It handles exceptions. try contains code that might throw, catch handles errors, and finally executes after try/catch regardless of the outcome.

**12. What is a React Fragment and why is it useful?** A React Fragment allows grouping multiple elements without adding extra nodes to the DOM. It uses <></> or <React.Fragment>.

**13. What are the key features of React?**

* Component-based architecture
* Virtual DOM
* Unidirectional data flow
* JSX syntax
* Hooks

**14. How do you fetch data in React?** Using fetch() or axios in useEffect, often combined with useState to store and render data.

**15. What is the difference between slice() and splice()?** slice() returns a shallow copy of a portion of an array. splice() modifies the array by removing or adding elements.

**16. What is package.json used for?** It holds metadata relevant to the project and manages dependencies, scripts, and project configuration.

**17. How do you select HTML elements using JavaScript?** Using methods like getElementById, querySelector, getElementsByClassName, etc.

**18. How does addEventListener work?** It attaches an event handler to an element without overwriting existing handlers.

**19. What is the difference between synchronous and asynchronous JavaScript?** Synchronous code executes line by line. Asynchronous code (e.g., with Promises or async/await) allows other operations to run while waiting.

**20. How do you remove an element from the DOM using JavaScript?** Using element.remove() or parent.removeChild(child).

**21. What are Promises in JavaScript? How do they work?** Promises represent an async operation. They can be in pending, fulfilled, or rejected states. .then() and .catch() are used for handling results.

**22. Explain the use of break and continue in loops** break exits the loop early, and continue skips the current iteration and moves to the next one.

**23. What is the difference between while and do...while loops?** while checks the condition before the loop body. do...while executes the loop body once before checking the condition.

**24. How does the try-catch-finally block work in JavaScript?** (Repeated - See Q11)

**25. Explain the difference between Array.filter() and Array.find().** filter() returns all matching elements in an array. find() returns the first matching element.

**26. What does Object.keys() and Object.values() return?** Object.keys(obj) returns an array of keys. Object.values(obj) returns an array of values.

**27. What is the difference between .slice() and .splice() methods in an array?** (Repeated - See Q15)

**28. How do you merge two strings in JavaScript?** Using +, concat(), or template literals: "Hello" + "World", or Hello ${world}.

**29. How can you convert a string to an array of characters?** Using split('') method: 'hello'.split('').

**30. What is the difference between state and props?** Props are passed to components and are read-only. State is managed within the component and can be changed using useState.