Q1. What is React, and what advantages does it offer over other front-end frameworks?

Sample Answer: React is a JavaScript library for building user interfaces, mainly focused on the view layer.React is used for building fast and interactive user interfaces (UI), especially for single-page applications (SPAs). The key advantages are It uses a virtual DOM for faster updates. You can adopt React gradually without rewriting an entire code base.

Q2. What is JSX?

Sample Answer: JSX, or JavaScript XML, is a syntax extension for JavaScript commonly used with React. It allows developers to write HTML-like code directly within JavaScript, making it easier to create and manage user interface components in a more readable and intuitive way.

Q3. What are React Components?

Sample Answer: React components are reusable building blocks for the user interface. They allow you to split the UI into independent, reusable pieces, making development more manageable.

Q4. What are ‘Functional Components’ and ‘Class Components’ in React?

Sample Answer: Functional components are simpler components written as functions that return JSX. On the other hand, class components use ES6 classes and have access to lifecycle methods.

Q5. What are React Hooks? Can you explain useState and useEffect?

Sample Answer: React Hooks are functions that enable functional components to manage state and lifecycle features. useState is a Hook that enables state variables to be used in functional components. It returns an array containing two elements: the current state value and a function to alter it. On the other hand, useEffect is used to handle side effects, such as fetching data, setting up subscriptions, or manually modifying the DOM. By default, it runs after every render.

Q6. What is state management, and how do you manage ‘state’ in React?

Sample Answer: In React, state management refers to the process of handling and updating data within an application and its components. For functional components, I utilize the useState hook to manage state, while in class components, I rely on the this.setState method to update the state.

Q7. What is the Virtual DOM, and how does it work?

Sample Answer: The Virtual DOM (Document Object Model) is a memory-resident, lightweight copy of the actual DOM that enables JavaScript frameworks such as React to efficiently update the user interface. To enhance performance, frameworks can identify minimal changes and implement them efficiently by comparing the Virtual DOM with the real DOM.

Q8. How does React handle events?

Sample Answer: React handles events using a synthetic event system, which is a wrapper around the browser’s native events. This system ensures consistent behavior across different browsers and improves performance by reusing event objects through event pooling (in older versions). React events are also named using camelCase (e.g., onClick instead of onclick) and work similarly to native DOM events but with added cross-browser consistency.

Q9. What is the difference between props and state in React?

Sample Answer: In React, props and state are both basic JavaScript objects that contain information to affect the output of render. However, they serve distinct purposes. Props are read-only properties that are transferred from a parent component to a child. Whereas a state is a local variable that is administered by a component and can be modified over time.

Q10. What is the purpose of keys in React?

Sample Answer: Keys in React are essential for the efficient identification and tracking of changes in lists of elements. React employs keys as distinct identifiers for each item when rendering a list. This enables React to ascertain which items have been added, removed, or reordered, and to update the DOM accordingly.

Q11. What are the different lifecycle methods in React?

Sample Answer: React’s component lifecycle is a series of methods that manage a component’s life from creation to destruction. Lifecycle methods include componentDidMount, componentDidUpdate, and componentWillUnmount for class components, and useEffect for functional components.

Q12. What is the method of handling page submissions within React?

Sample Answer: Page submissions can be managed in React by employing state to manage form data. Upon submission of the form, an event handler is called, which typically validates the form data before transmitting it to the server or executing another action. The event handler alters the state in response to user input and mitigates the default form submission by utilising React’s controlled components to manage the process.

Q13. Which methods can be used to maximize performance outputs in React-based systems?

Sample Answer: Developers for React-based systems can use several methods to minimise unneeded rendering, enhance load times, and maximise resource use to increase system performance. Some important ways to boost performance are listed below:

Using React’s PureComponent for Class Components: By only updating when the component’s props or state have changed, PureComponent automatically runs the shouldComponentUpdate function to stop needless re-renders.

For functional components, implement React.memo: A higher-order component called React.memo guarantees that functional components only re-render when their props change. This, hence, lowers the number of renders by memoizing them.

Lazy loading: This function lets components load only when required, therefore boosting application speed and performance by lowering the initial load time.

Reduce Unneeded Re-renders Using Class Components: By using shouldComponentUpdate, you can explicitly determine when a component should re-render, therefore lowering needless updates and enhancing general performance.

Q14. What is the purpose of React Router?

Sample Answer: React Router is a library that offers features for routing and navigation in React applications. It allows the development of single-page applications (SPAs) that render various components according to the URL, without necessitating a complete page refresh. User experience can be improved by enabling seamless transitions between various perspectives.

Q15. How does React work?

React creates a virtual DOM. When state changes in a component it firstly runs a "diffing" algorithm, which identifies what has changed in the virtual DOM. The second step is reconciliation, where it updates the DOM with the results of the difference.

Q16. Explain the use of render method in React?

React renders HTML to the web page by using a function called render(). The purpose of the function is to display the specified HTML code inside the specified HTML element. In the render() method, we can read props and state and return our JSX code to the root component of our app.

Q17. What is higher-order component in React?

Higher-order components or HOC is the advanced method of reusing the component functionality logic. It simply takes the original component and returns the enhanced component. HOC are beneficial as they are easy to code and read. Also, helps to get rid of copying the same logic in every component.

Q18. What is React Router?

React Router is a standard library for routing in React. It enables the navigation among views of various components in a React Application, allows changing the browser URL, and keeps the UI in sync with the URL.

To install react router type the following command. "npm i react-router-dom"

Q19. What are hooks in React?

Hooks are a new addition in React 16.8. They let developers use state and other React features without writing a class. Hooks doesn’t violate any existing React concepts. Instead, Hooks provide a direct API to react concepts such as props, state, context, refs and life-cycle

Q20. Explain the useState hook in React?

The most used hook in React is the useState() hook. Using this hook we can declare a state variable inside a function but only one state variable can be declared using a single useState() hook. Whenever the useState() hook is used, the value of the state variable is changed and the new variable is stored in a new cell in the stack.

Syntax:

const [state, setState] = useState(initialState);

state: The current state value.

setState: A function used to update the state value.

initialState: The initial value of the state.

Q21. Explain the useEffect hook in React?

The useEffect hook in React eliminates the side effect of using class based components. It is used as an alternative to componentDidUpdate() method. The useEffect hook accepts two arguments where second argument is optional.

useEffect(function, dependency)

The dependency decides when the component will be updated again after rendering.