1. A Single Page Application (SPA) is a web application that dynamically updates the current page rather than loading entire new pages from the server. Key characteristics include using client-side routing, typically utilizing AJAX for data retrieval, and often having a more seamless user experience compared to traditional multi-page applications (MPAs).

2. SPAs offer faster navigation, smoother user experience, and better performance by minimizing server requests. Examples include Gmail, Facebook, and Twitter.

3. React is a JavaScript library for building user interfaces. JSX is a syntax extension for JavaScript that allows mixing HTML-like code with JavaScript code.

4. JSX is used in React to write component structure in a more declarative and intuitive way. JSX elements are rendered using React's `ReactDOM.render()` method.

5. Yes, JavaScript expressions can be embedded in JSX using curly braces `{}`. For example, `{2 + 2}`.

6. The `className` attribute in JSX is used to specify CSS classes for elements. JSX fragments are used to group multiple elements without adding extra nodes to the DOM.

7. JSX to JavaScript conversion example: `<div>Hello, world!</div>` becomes `React.createElement('div', null, 'Hello, world!');`

8. JavaScript to JSX conversion example: `React.createElement('div', null, 'Hello, world!');` becomes `<div>Hello, world!</div>`

9. A React component is a reusable piece of UI. The two types of components in React are functional components and class components.

10. A functional component in React is defined using a JavaScript function that returns JSX.

11. A class component in React is defined using ES6 class syntax that extends `React.Component` and implements a `render()` method.

12. Props in React components are used to pass data from parent to child components.

13. Props are passed to a React component as attributes. For example, `<MyComponent name="John" />`.

14. State in a React component represents data that can change over time. In a class component, state is initialized in the constructor using `this.state`.

15. State in a React component can be updated using the `setState()` method provided by React.

16. The component lifecycle in React consists of three main phases: Mounting, Updating, and Unmounting.

17. Lifecycle methods in React are methods that are automatically invoked at certain points in a component's lifecycle, such as `componentDidMount()` and `componentDidUpdate()`.

18. The virtual DOM in React is an in-memory representation of the actual DOM. It works by efficiently updating only the parts of the DOM that have changed, rather than re-rendering the entire DOM.

19. The purpose of the virtual DOM in React is to improve performance by minimizing DOM manipulation and updates.

20. React's reconciliation algorithm compares the virtual DOM with the previous virtual DOM to determine the most efficient way to update the actual DOM.

21. External CSS stylesheets can be applied to a React component by importing the stylesheet into the component file.

22. Inline CSS styles can be applied to a React component using the `style` attribute with a JavaScript object containing CSS properties and values.

23. A list of items in React can be rendered using the `map()` method to iterate over an array and return JSX elements for each item.

24. Providing a unique `key` prop when rendering lists in React is important for efficient updating of the virtual DOM and ensuring proper component re-rendering.

25. Content can be conditionally rendered in React using JavaScript expressions within JSX, such as ternary operators or logical && operators.

26. Controlled forms in React are forms where form data is handled by React components, while uncontrolled forms rely on the DOM to handle form data.

27. A hook in React is a function that allows functional components to use state and other React features. `useState` hook is used for managing state, and `useEffect` hook is used for performing side effects in functional components.

28. The Model-View-Controller (MVC) architectural pattern separates an application into three interconnected components: the Model (data), the View (user interface), and the Controller (logic).

29. In the MVC pattern, the Model manages the data and business logic, the View displays the user interface, and the Controller handles user input and updates the Model and View accordingly.

30. Redux is a predictable state container for JavaScript apps, primarily used with React. Its core principles include having a single source of truth (the store), state is read-only, and changes are made with pure functions (reducers).

31. The main components of Redux architecture are the store, actions, reducers, and middleware.

32. Data flow in Redux architecture follows a unidirectional flow: actions are dispatched to reducers, which update the state in the store, triggering re-renders in connected components.

33. Redux action is a plain JavaScript object that represents an intention to change the state, reducers are functions that specify how the state changes in response to actions, and the Redux store holds the state of the application.

34. Middleware in Redux allows for extending Redux's capabilities, such as logging, asynchronous actions, or applying third-party plugins.

35. Different types of testing for React applications include unit testing, integration testing, and end-to-end testing. Testing libraries for React include Jest, Enzyme, and React Testing Library.