INTERVIEW QUESTIONS OF REACT JS

1. What is ReactJS?

ReactJS is an open-source JavaScript library developed by Facebook for building user interfaces, particularly for single-page applications. It allows developers to create reusable UI components that manage their state efficiently.

2. What are the main features of React?

The main features of React include:

- **Component-Based Architecture**: UI is built using independent, reusable components.
- **Virtual DOM**: React creates a virtual representation of the DOM, which improves performance by minimizing direct updates to the actual DOM.
- **JSX**: A syntax extension that allows mixing HTML with JavaScript, making the code more readable.
- **One-Way Data Binding**: Data flows in one direction, making it easier to track changes and manage the application state.

3. What is JSX?

JSX (JavaScript XML) is a syntax extension for JavaScript that allows developers to write HTML-like code within JavaScript. JSX makes it easier to create and visualize the structure of the UI components in React.

4. What is the Virtual DOM?

The Virtual DOM is an in-memory representation of the actual DOM. When a component's state changes, React updates the Virtual DOM first and then efficiently reconciles the changes with the actual DOM. This process minimizes the performance overhead associated with direct DOM manipulation.

5. What are components in React?

Components are the building blocks of a React application. They can be functional or class-based and are used to encapsulate UI logic and rendering. Each component can maintain its own state and can be reused across the application.

6. What is the difference between state and props?

- **State**: State is a local data storage that is managed within a component. It can be changed or updated, and when it does, the component re-renders.
- **Props**: Props (short for properties) are read-only attributes passed from parent components to child components. They are used to pass data and event handlers to child components.

7. What are hooks in React?

Hooks are special functions introduced in React 16.8 that allow developers to use state and other React features in functional components. Common hooks include `useState` (for managing state) and `useEffect` (for managing side effects).

8. What is the useEffect hook?

The `useEffect` hook is used to perform side effects in functional components. It runs after the render and can be used for tasks like data fetching, subscriptions, or manual DOM manipulations. You can control when it runs by passing dependencies as an array.

9. What is Redux, and how does it work with React?

Redux is a state management library that helps manage application state in a predictable way. It works with React by providing a central store for the application's state, allowing components to access and update the state via actions and reducers. React-Redux is the official binding library that connects Redux with React.

---

10. What are controlled components?

Controlled components are React components that do not maintain their own state. Instead, their value is controlled by React state, making it easier to manage and validate user input. This is often used with form elements, where the input value is set via props and updated through events.