REACT – Interview Answers

1. What are all the hooks you know?
React Hooks allow functional components to use state and lifecycle features. Common hooks include:

- useState – manage component state
- useEffect – handle side effects
- useContext – share data globally
- useReducer – complex state logic
- useRef – access DOM or persist values
- useMemo – optimize performance
- useCallback – memoize functions
- useLayoutEffect – DOM measurements
- useImperativeHandle – expose methods
- useDebugValue – debugging custom hooks

2. Use of Frontend
Frontend is responsible for user interface and user experience. It handles user interactions, form validation, routing, API calls, state management, and displays data received from the backend.

3. Difference between React and JavaScript
JavaScript is a programming language used for logic and functionality. React is a JavaScript library used to build component-based user interfaces.

4. What is CSR (Client Side Rendering)?
In CSR, the browser loads a minimal HTML file and renders content using JavaScript. React fetches data and builds UI on the client side after page load.

5. How do you pass data to the backend?
Data is sent using HTTP requests such as GET, POST, PUT, and DELETE. Frontend uses fetch or axios to send JSON data to backend APIs.

NODE.JS – Interview Answers

1. What is Node.js?
Node.js is a JavaScript runtime built on Chrome’s V8 engine. It allows JavaScript to run on the server and is used for building scalable backend applications.

2. What are imported modules in Node.js?
Modules are reusable blocks of code. • Core modules: fs, http, path, os

- Third-party modules: express, mongoose, dotenv
- Custom modules: user-defined files

3. Difference between Node.js and JavaScript
JavaScript runs in the browser and handles UI logic. Node.js runs JavaScript on the server and handles backend operations.

4. What is a server?
A server is a system that listens for client requests, processes them, and sends responses back over the network.

5. Backend process in Node.js
Client Request → Server → Router → Controller → Business Logic → Database → Response