What is React?

React is an open-source JavaScript Library use to build Dynamic and interactive user interfaces. It is developed at Facebook in 2011. Currently most widely used JS library for front-end development. Used to create single page application (page does not re-load) and if simplifies the creation of SPA by using reusable components.

What are the key-features of React.js?

React utilizes a virtual representation of the DOM, allowing efficient updates by minimizing direct manipulation of the actual DOM, resulting in improved performance.

React structures user interfaces as modular, reusable components, promoting a more Architecture: maintainable and scalable approach to building applications.

JSX (JavaScript XML): JSX is a syntax extension for JavaScript used in React, allowing developers to write HTML-like code within JavaScript, enhancing readability and maintainability.

React have a declarative programming style(JSX), where developers focus on "what" the Ul should look like and React handles the "how" behind the scenes. This simplify the code.

Hooks are functions that enable functional components to manage state and lifecycle features, providing a more concise and expressive way to handle component logic.

What is the role of React in software development? V.IMP.

React is a JavaScript library. React is used to simply the creation of complex UI application.

What is single page application?

A Single Page Application (SPA) is a web application that have only one single web page.

Whenever user do some action on the website, then in response content is dynamically updated without refreshing or loading a new page.

Advantage of React?

Simple to build Single Page Application (by using Components)

React is cross platform and open source (Free to use)

Lightweight and very fast (Virtual DOM)

Disadvantages of react?

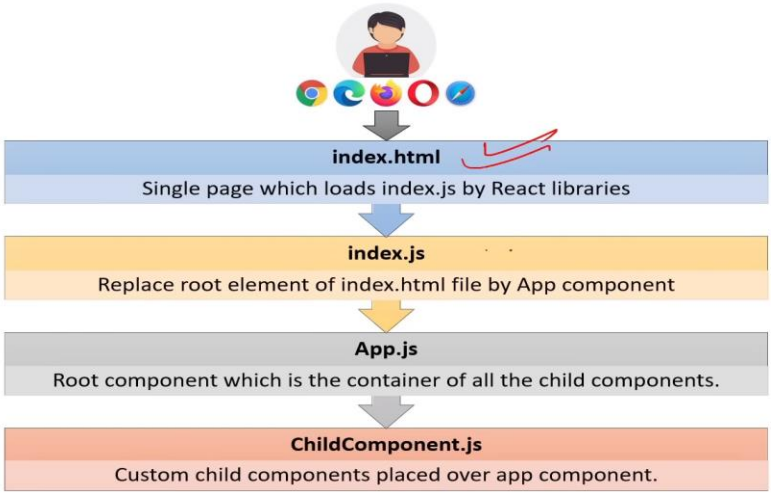
React is not a good choice for very small applications.

What is the difference between React and Angular?

Q. What is the difference between React and Angular?

React	Angular
React and Angular both are used to create single page UI applications using components.	
1. React is a JavaScript library	Angular is a complete framework.
2. React uses a **virtual DOM** which makes it faster.	Angular uses a real DOM
3. React is smaller in size and **lightweight** and therefore faster sometime.	Angular is bigger because it is a complete framework.
4. React depends on external libraries for many complex features, so developer has to write many lines of code for complex functionalities.	Since Angular is a complete framework, therefore it provide **built-in support** for features like routing, forms, validation, and HTTP requests.

Q. How React App load and display the components in browser? V. IMP.



Virtual DOM?

React uses a virtual DOM to efficiently update the UI without re-render the entire page, which helps improve performance and make the application more responsive.

DOM	Virtual DOM
1. DOM is actual representation of the webpage.	Virtual DOM is lightweight copy of the DOM.
2. Re-renders the entire page when updates occur.	Re-renders only the changed parts efficiently.
3. Can be slower, especially with frequent updates.	Optimized for faster rendering.
4. Suitable for static websites and simple applications	Ideal for dynamic and complex single-page applications with frequent updates

What are React Components? V.IMP.

In React, a component is a reusable building block for creating user interfaces.

What is npm? What is the role of node_modules folder?

NPM is used to manage the dependencies for your React project, including the React library itself.

What is the role of public folder in React?

Public folder contains static assets that are served directly to the user's browser, such as images, fonts, and the index.html file.

Why className is used over class attribute in React?

In javaScript, class is a keyword whether JSX is an extension of JavaScript. That's why to avoid any confusion React uses className instead of class. For example, Passing a string as the className prop

What is the role of index.html page in React? V. IMP.

index.html file in a React project is the entry point to the application.

What is the difference between Declarative & Imperative syntax?

Declarative syntax focuses on describing the desired result without specifying the step-by-step process. Imperative syntax involves step by step process to achieve a particular goal.

JSX in React is used to write declarative syntax.

JavaScript has an imperative syntax.

What is the role of App.js file in React? V.IMP.

App.js file contain the root component(App) of React application. App component is like Container for other components. App.js defines the structure, layout, and routing in the application.

What is the role of function and return inside App.js?

The function keyword is used to define a JavaScript function that represents your React component.

The function takes in props as its argument (if needed) and returns JSX.

What is the role of main.js file, ReactDOM and render method in React?

ReactDOM is a JavaScript library which convert your components (which are written in React syntax) to the actual browser DOM.

main.js file is the JavaScript file which renders all the components and replace the root element of index.html file with the newly rendered root element.

What is JSX?

JSX (JavaScript XML) is a syntax extension used by React to write HTML-like code.

JSX is converted to JavaScript with the tools like Babel Because Browsers understand JavaScript not JSX.

What is babel?

Babel in React is used to transpile JSX syntax into regular JavaScript which browser can understand.

What are the advantages of JSX? V. IMP.

1. Improve code readability and writability

2. Error checking in advance (Type safety)

3. Support JavaScript expressions

4. Improve Performance

5. code reusability.

Can browser read a JSX File? What is Babel?

No. Browser will not understand JSX code directly. Babel in React is used to transpile JSX syntax into regular JavaScript which browser can understand.

What is Transpiler and compiler?

A Transpiler is a tool that converts source code from one high-level programming language(JSX) to another high-level programming language(JavaScript). Example: Babel

A compiler is a tool that converts high-level programming language(Java) into a lower- level language(machine code or bytecode).

Is it possible to use JSX without React?

Yes, it's possible to use JSX without React by creating your own transpiler like Babel

What is the role of fragment in React?

In React, a fragment is a way to group multiple children elements without adding extra nodes to the DOM.

What are props in JSX? V.IMP.

props (properties) are a way to pass data from a parent component to a child component.

What is the difference between state and props?

Mutability: State is mutable and can be changed by the component, while props are immutable and passed to the component by its parent.

Ownership: State is owned and managed by the component itself, whereas props are owned by the parent component and passed down to child components.

Usage: State is used to manage dynamic data that affects the component's behavior and rendering, while props are used to pass data and event handlers to child components for configuration and interaction.

What are spread operator?

The spread operator (...) is used to expand or spread an array or object.

Whether React is a Framework or a Librarv? What is the difference?

Library: Developers import the libraries at the top and then used its functions in components. React is commonly referred to as a JavaScript library.

Framework: Developers need to follow a specific structure or pattern defined by the framework.

Angular is a framework.

How React provides Reusability and Composition?

React provides reusability and composition through its component-based architecture.

Reusability: Once you create a component, you can re-use it in different parts of your application or even in multiple projects.

Composition: Composition is creating new and big components by combining existing small components.

What are state, stateless, stateful and state management terms?

"state" refers to the current data of the component.

Stateful or state management means, when a user performs some actions on the Ul, then the React application should be able to update and re-render that data or state on the Ul.

What are stateless components?

If the component's behaviour will be not depend of its state, then it can be called a stateless component.

What are Props n JSX?

props (properties) are a way to pass data from a parent component to a child component.

What is prop drilling in react?

Prop drilling is the process of passing down props through multiple layers of components.

What is the role of src folder in React?

src folder is used to store all the source code of the application which is responsible for the dynamic changes in your web application.

What is the role of export default inside App.js?

Export statement is used to make a component available for import using "import" statement in other files.

What are the Types of React components? What are Functional Components? V. IMP.

Functional components are declared as a JavaScript function. They are stateless component, but with the help of hooks, they can now manage state also.

In how many ways can avoid Prop Drilling? V. IMP.

Using (Context AP) Using Redux

Q. What are Class Components In React? V.IMP.

Class components are defined using JavaScript classes.

They are stateful components by using the lifecycle methods. The render method in a class component is responsible for returning JSX.

Q. What are the 5 differences btw Functional components & Class components? V. IMP.

Functional Component	Class Component
1. Syntax: Defined as a JS function.	Defined as a JS(ES6) class.
2. State: Originally stateless but can now maintain state using hooks.	Can manage local state with this.state.
3. Lifecycle methods: No	Yes
4. Readability: more readable & concise.	Verbose(complex).
5. this keyword: No	Yes (Access props using this.props)

What is Routing and Router in React? V.IMP.

Routing allows you to create a single-page web application with navigation, without refreshing the page.

React Router is a library for handling routing and rendering of different components based on the URL.

What are the roles of <Routes> & <Route> component in React Routing? V. IMP.

The <Routes> component is used as the root container for declaring your collection of routes. The <Route> component is used to define a route and specify the component that should render when the route matches.

What are Route Parameters in React Routing?

Route parameters in React Router are a way to pass dynamic values(data) to the component as part of the URL path.

What is the role of Switch Component in React Routing?

“Switch component ensures that only the first matching <Route> is rendered and rest are ignored.

What are React Hooks? What are the Top React Hooks? V. IMP.

React Hooks are inbuilt functions provided by React that allow functional components to use state and lifecycle features.

useState: useEffect; useContext; useMemo;

What is the role of useState() hook and how it works? V.IMP.

The useState() is a built-in React Hook that allows you for having state variables in functional components.

useState() working: usestate() function accept the initial state value as the parameter and returns an array with two elements:

The first element is the current state value (count in this code).

Second element is the function that is used to update the state (setCount in this code).

The concept of assign array elements to individual variables is called array destructuring.

What is the role of useEffect(). How it works and what is its use? V. IMP

The useEffect Hook in React is used to perform side effects in functional components.

For example, data fetching from API, subscriptions or any other operation that needs to be performed after the component has been rendered. useEffect() function will accept two parameter: (Effect function, dependency array).

What is the meaning of the empty array [] in the useEffect()?

An empty array [] indicates that the effect function should only run once.

What is useref()?

useRef is a used to access or store a DOM element directly.

Why useRef does not cause a re-render?

useRef does not cause a re-render when its .current property changes because it holds a mutable value that doesn't participate in React's rendering lifecycle. It is used for directly accessing DOM elements or storing values that persist across renders.

What is the role of useContext() hook? V.IMP.

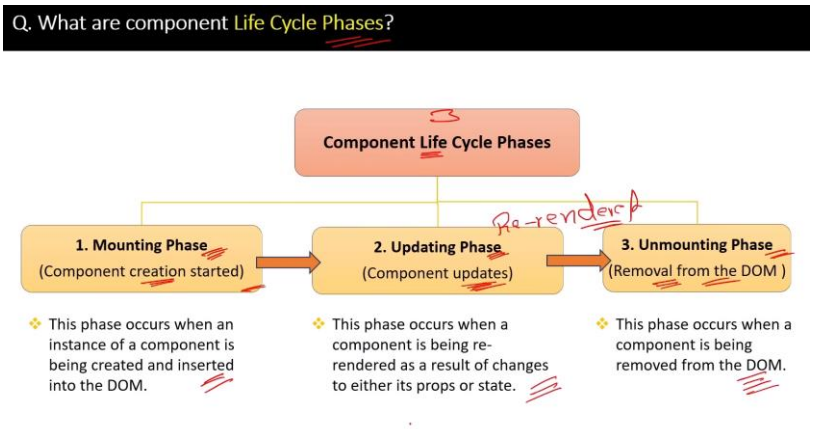
The useContext() hook in React is used to access the value of a context directly within a functional component.

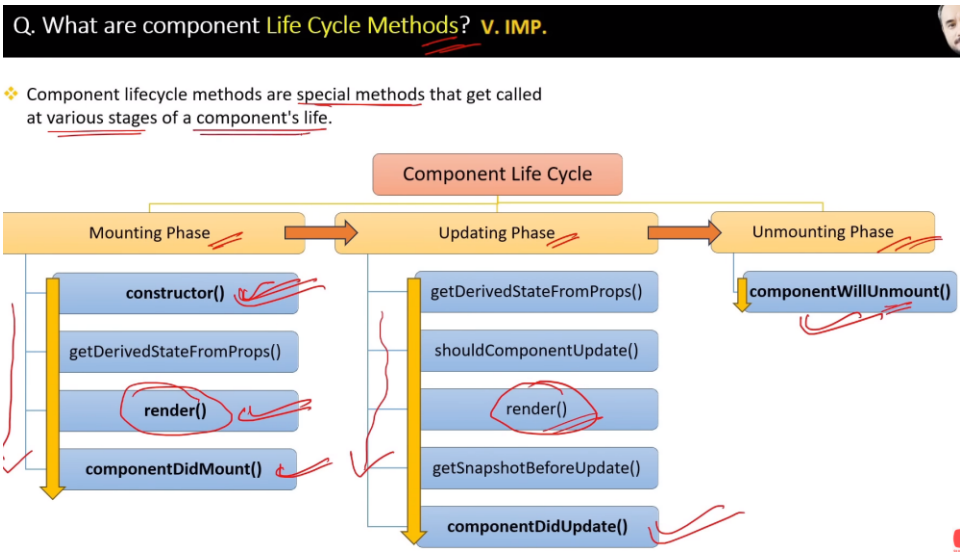
What is createContext() method? What are Provider & Consumer properties?

createContext() function returns an object with Provider and Consumer properties.The Provider property is responsible for providing the context value to all its child components. useContext() method or Consumer property can be used to to access the value of a context directly within a functional component.

What is ref?

it can enable the access to the DOM element directly so we can manipulate it. It can be used so save a value between re-renders in its current property that doesn't cause a re-render while being updated.





What is the role of render() method in component life cycle?

Render() method returns the React elements that will be rendered to the DOM.

How the State can be maintained in a class component? Vv. IMP.

Two step process to maintain state: _this.setState() method is used to update the state. this.state property is used to render the update state in DOM.

What is the role of componentDidMount() method in component life cycle?

componentDidMount() lifecycle method in React is the part of mounting phase and is called after a component has been rendered to the DOM.

Mostly used for side effects. For example, external data fetching or setting up subscriptions.

What are Controlled Components in React? V.IMP.

A controlled component is a component whose form elements (like input fields or checkboxes) are controlled by the state of the application.

What are the Differences btw Controlled & Uncontrolled Components? V. IMP.

Controlled Components	Uncontrolled Components
1. Values are controlled by **React state**.	Values are not controlled by React state.
2. Event handlers **update React state**.	No explicit state update; values can be accessed **directly from the DOM**.
3. Don't depend on useRef().	Commonly uses **useRef()** to access form element values.
4. **Re-renders** on state changes.	Less re-rendering since values are not directly tied to React state.
5. A **recommended** and standard practice for form handling in React.	Useful in certain scenarios but less commonly considered a best practice.

How to handle forms in React?

The preferred and recommended approach for handling forms in React is by using controlled components.

What is Code Splitting in React? V.IMP.

Code splitting is a technique to split the JavaScript bundle into smaller chunks, which are loaded on- demand.

What is the role of Lazy and Suspense methods in React? V. IMP.

React.lazy is a function that allows you to load a component lazily. It enables code splitting meaning component is loaded only when it is required.

What is a Higher-Order Component in React? V. IMP.

A Higher-Order Component is a component which takes another component as an argument and adds extra features to another component.

Q. What are the differences between React & React Native?

React	React Native
1. React is a library	React Native is a framework.
2. React is used for building web interfaces.	React Native is used for building mobile applications.
3. Run on Web browsers.	Run on iOS and Android platforms.
4. HTML and CSS are used for UI.	Native UI components (e.g., View, Text) are used for UI.
5. Deployed as web applications.	Deployed through app stores(e.g., App Store, Google Play).

Q. What are the Top 3 ways to achieve state management? When to use what in React?

1. useState Hook:

- **When to use:** Simple component-level state.
- **Reason:** Ideal for applications having small components and isolated state because it is Lightweight and built into React only.

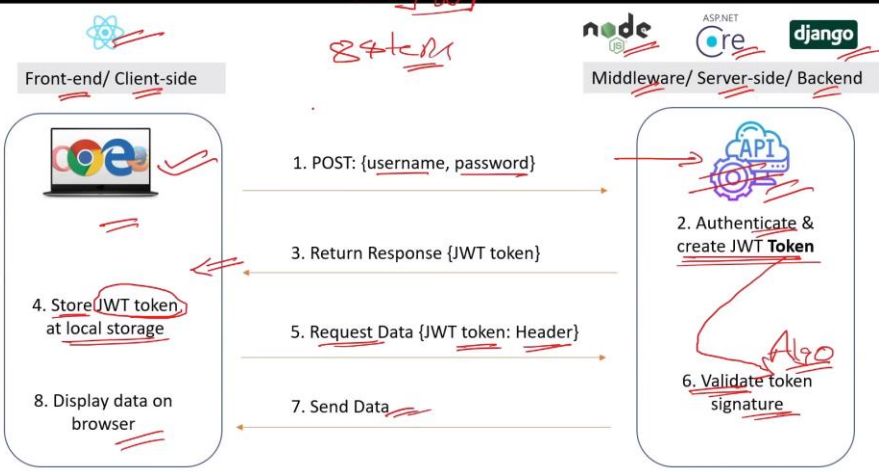
2. Context API:

- **When to use:** Prop drilling avoidance for sharing global data.
- **Reason:** Simplifies data passing through the component tree, reducing the need for manual prop drilling.

3. Redux:

- **When to use:** Large-scale applications with complex state.
- **Reason:** Centralized store and actions provide a predictable state management pattern, aiding in debugging and scalability.

Q. How can you Implement Authentication in a React application? V. IMP.



Q. What is the difference between fetch & axios for api calls in React?

```
fetch("https://api.example.com/data")
  .then((response) => response.json())
  .then((data) => console.log(data))
  .catch((error) => console.error("Error:", error));
```

1. fetch is a **built-in JavaScript function**, so it doesn't require any additional libraries.
2. fetch **returns Promises**, making it easy to work with asynchronous code using async/await syntax.
3. If you want to keep **http requests simple,** fetch is a good choice.

```
// installation command: npm install axios
import axios from "axios";
axios
  .get("https://api.example.com/data")
  .then((response) => console.log(response.data))
  .catch((error) => console.error("Error:", error));
```

1. Axios is a **third-party library,** that simplifies the process of making HTTP requests.
2. Axios allows you to **use interceptors**, which can be good for tasks like request/response logging, authentication, and error handling.
3. If you want to intercept http request/response, or improve error handling then Axios has more features to do it.

Q. How can you Optimize Performance in a React application?

1. Memoization with useMemo and useCallback:	Use this hooks to memoize values and, reducing unnecessary recalculations.
2. Optimizing Renders with React.Fragment:	Use it to avoid unnecessary wrapper elements that could cause additional DOM nodes.
3. Lazy Loading with React.lazy:	Use it to load components lazily, reducing the initial bundle size and improving initial loading performance.
4. Code Splitting:	Employ code splitting to divide your application into smaller chunks that are loaded on demand, improving initial load times.
5. Optimizing Images and Assets:	Compress and optimize images, use responsive images, and leverage lazy loading for images to reduce network and rendering overhead.

What is MVC architecture?

The Model-View-Controller (MVC) framework is an architectural/design pattern that separates an application into three main logical components Model, View, and Controller. Each architectural component is built to handle specific development aspects of an application. It isolates the business, logic, and presentation layer from each other

Explain the building blocks of React?

The five main building blocks of React are:

- **Components:** These are reusable blocks of code that return HTML.
- **JSX:** It stands for JavaScript and XML and allows you to write HTML in React.
- **Props and State:** props are like function parameters and State is similar to variables.
- **Context:** This allows data to be passed through components as props in a hierarchy.
- **Virtual DOM:** It is a lightweight copy of the actual DOM which makes DOM manipulation easier.

What is a key in React?

In React, a key is a special attribute used to uniquely identify elements in a list.

- Keys help react identify which elements were added, changed or removed.
- Without keys, React does not understand the order or uniqueness of each element.
- With keys, React has an idea of which particular element was deleted, edited, and added.
- Keys are generally used for displaying a list of data coming from an API.

What are Custom Hooks?

Custom hooks in React are reusable functions that encapsulate and share logic across multiple components. They allow you to extract component logic into reusable functions, promoting cleaner and more maintainable code.

What is Redux?

Redux is an open-source, JavaScript library used to manage the application state. React uses Redux to build the user interface. It is a predictable state container for JavaScript applications and is used for the entire application's state management.

What are the components of Redux?

- Store: Holds the state of the application.
- Action: The source information for the store.
- Reducer: Specifies how the application's state changes in response to actions sent to the store.