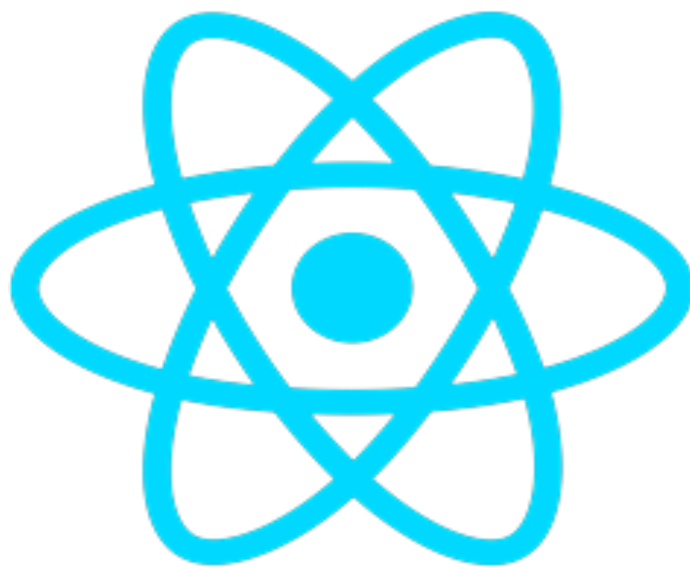


Top 75 React Interview Questions and Answers



1) What is Reactjs?

React is a JavaScript library that makes building user interfaces easy. It was developed by Facebook.

2) Does React use HTML?

No, It uses JSX, which is similar to HTML.

3) When was React first released?

React was first released on March 2013.

4) Give me two most significant drawbacks of React

- Integrating React with the MVC framework like Rails requires complex configuration.
- React require the users to have knowledge about the integration of user interface into MVC framework.

5) State the difference between Real DOM and Virtual DOM

Real DOM

It is updated slowly.

It allows a direct update from HTML.

It wastes too much memory.

Virtual DOM

It updates faster.

It cannot be used to update HTML directly.

Memory consumption is less

6) What is Flux Concept In React?

Facebook widely uses flux architecture concept for developing client-side web applications. It is *not* a framework or a library. It is simply a new kind of architecture that complements React and the concept of Unidirectional Data Flow.

7) Define the term Redux in React

Redux is a library used for front end development. It is a state container for JavaScript applications which should be used for the applications state management. You can test and run an application developed with Redux in different environments.

8) What is the 'Store' feature in Redux?

Redux has a feature called 'Store' which allows you to save the application's entire State at one place. Therefore all its component's State are stored in the Store so that you will get regular updates directly from the Store. The single state tree helps you to keep track of changes over time and debug or inspect the application.

9) What is an action in Redux?

It is a function which returns an action object. The action-type and the action data are always stored in the action object. Actions can send data between the Store and the software application. All information retrieved by the Store is produced by the actions.

10) Name the important features of React

Here, are important features of React.

- ♦ Allows you to use 3rd party libraries
 - ♦ Time-Saving
 - ♦ Faster Development
 - ♦ Simplicity and Composable
 - ♦ Fully supported by
 - ♦ Facebook.
 - ♦ Code Stability with One-directional data binding
- React Components

11) Explain the term stateless components

Stateless components are pure functions that render DOM-based solely on the properties provided to them.

12) Explain React Router

React Router is a routing library which allows you to add new screen flows to your application, and it also keeps URL in sync with what's being shown on the page.

13) What is dispatcher?

A dispatcher is a central hub of app where you will receive actions and broadcast payload to registered callbacks.

14) What is meant by callback function? What is its purpose?

A callback function should be called when `setState` has finished, and the component is re-rendered. As the `setState` is asynchronous, which is why it takes in a second callback function.

15) Explain the term high order component

A higher-order component also shortly known as HOC is an advanced technique for reusing component logic. It is not a part of the React API, but they are a pattern which emerges from React's compositional nature.

16) Explain the Presentational segment

A presentational part is a segment which allows you to renders HTML. The segment's capacity is presentational in markup.

17) What are Props in react js?

Props mean properties, which is a way of passing data from parent to child. We can say that props are just a communication channel between components. It is always moving from parent to child component.

18) Explain yield catchphrase in JavaScript

The yield catchphrase is utilized to delay and resume a generator work, which is known as yield catchphrase.

19) Name two types of React component

Two types of react Components are:

- ♦ Function component
- ♦ Class component

20) Explain synthetic event in React js

Synthetic event is a kind of object which acts as a cross-browser wrapper around the browser's native event. It also helps us to combine the behaviors of various browser into signal API.

21) What is React State?

It is an object which decides how a specific component renders and how it behaves. The state stores the information which can be changed over the lifetime of a React component.

22) How can you update state in react js?

A state can be updated on the component directly or indirectly.

23) Explain the use of the arrow function in React

The arrow function helps you to predict the behavior of bugs when passed as a callback. Therefore, it prevents bug caused by this all together.

24) State the main difference between Props and State

The main difference the two is that the State is mutable and Props are immutable.

25) Explain pure components in React js

Pure components are the fastest components which can replace any component with only a render(). It helps you to enhance the simplicity of the code and performance of the application.

26) What kind of information controls a segment in React?

There are mainly two sorts of information that control a segment: State and Props

- ◆ State: State information that will change, we need to utilize State.
- ◆ Props: Props are set by the parent and which are settled all through the lifetime of a part.

27) What is 'create-react-app'?

'create-react-app' is a command-line tool which allows you to create one basic react application.

28) Explain the use of 'key' in react list

Keys allow you to provide each list element with a stable identity. The keys should be unique.

29) What are children prop?

Children props are used to pass component to other components as properties.

30) Explain error boundaries?

Error boundaries help you to catch Javascript error anywhere in the child components. They are most used to log the error and show a fallback UI.

31) What is the use of empty tags ?

Empty tags are used in React for declaring fragments.

32) Explain strict mode

StrictMode allows you to run checks and warnings for react components. It runs only on development build. It helps you to highlight the issues without rendering any visible UI.

33) What are reacted portals?

Portal allows you to render children into a DOM node. **CreatePortalmethod** is used for it.

34) What is Context?

React context helps you to pass data using the tree of react components. It helps you to share data globally between various react components.

35) What is the use of Webpack?

Webpack in basically is a module builder. It is mainly runs during the development process.

36) What is Babel in React js?

Babel, is a JavaScript compiler that converts latest JavaScript like ES6, ES7 into plain old ES5 JavaScript that most browsers understand.

37) How can a browser read JSX file?

If you want the browser to read JSX, then that JSX file should be replaced using a JSX transformer like Babel and then send back to the browser.

38) What are the major issues of using MVC architecture in React?

Here are the major challenges you will face while handling MVC architecture:

- DOM handling is quite expensive
- Most of the time applications were slow and inefficient
- Because of circular functions, a complex model has been created around models and ideas

39) What can be done when there is more than one line of expression?

At that time a multi-line JSX expression is the only option left for you.

40) What is the reduction?

The reduction is an application method of handling State.

41) Explain the term synthetic events

It is actually a cross-browser wrapper around the browser's native event. These events have interface `stopPropagation()` and `preventDefault()`.

42) When should you use the top-class elements for the function element?

If your element does a stage or lifetime cycle, we should use top-class elements.

43) How can you share an element in the parsing?

Using the State, we can share the data.

44) Explain the term reconciliation

When a component's state or props change then rest will compare the rendered element with previously rendered DOM and will update the actual DOM if it is needed. This process is known as reconciliation.

45) How can you re-render a component without using setState() function?

You can use forceUpdate() function for re-rendering any component.

46) Can you update props in react?

You can't update props in react js because props are read-only. Moreover, you can not modify props received from parent to child.

47) Explain the term 'Restructuring.'

Restructuring is extraction process of array objects. Once the process is completed, you can separate each object in a separate variable.

48) Can you update the values of props?

It is not possible to update the value of props as it is immutable.

49) Explain the meaning of Mounting and Demounting

- The process of attaching the element to the DCOM is called mounting.
- The process of detaching the element from the DCOM is called the demounting process.

50) What is the use of 'props-types' library?

'Prop-types' library allows you to perform runtime type checking for props and similar object in a recent application.

51. What is React?

React is a JavaScript library focused on creating declarative user interfaces (UIs) using a component-based concept. It's used for handling the view layer and can be used for web and mobile apps.

52) What is useState() in React ?

The React useState Hook allows us to track state in a function component.

State generally refers to data or properties that need to be tracking in an application.

To use the useState Hook, we first need to import it into our component.

useState accepts an initial state and returns two values:

- The current state.
- A function that updates the state.

53) What is JSX?

JSX stands for JavaScript XML. It allows us to write HTML inside JavaScript and place them in the DOM without using functions like `appendChild()` or `createElement()`.

As stated in the official docs of React, JSX provides syntactic sugar for `React.createElement()` function.

54) What are the differences between controlled and uncontrolled components?

A Controlled Component is one that takes its current value through props and notifies changes through callbacks like `onChange`.

A Uncontrolled Component is one that stores its own state internally, and you query the DOM using a ref to find its current value when you need it.

55) When rendering a list what is a key and what is it's purpose?

Keys help React identify which items have changed, are added, or are removed. Keys should be given to the elements inside the array to give the elements a stable identity. The best way to pick a key is to use a string that uniquely identifies a list item among its siblings.

56) Why do React Hooks make use of refs?

Earlier, refs were only limited to class components but now it can also be accessible in function components through the `useRef` Hook in React.

The refs are used for:

- Managing focus, media playback, or text selection.

57) What is React Hooks?

Hooks allow function components to have access to state and other React features. Because of this, class components are generally no longer needed.

56) Differentiate between States and Props.

The major differences between States and Props are given below.

SN	Props	State
1.	Props are read-only.	State changes can be asynchronous.
2.	Props are immutable.	State is mutable.
3.	Props allow you to pass data from one component to other components as an argument.	State holds information about the components.
4.	Props can be accessed by the child component.	State cannot be accessed by child components.

57) What is an event in React?

An event is an action which triggers as a result of the user action or system generated event like a mouse click, loading of a web page, pressing a key, window resizes, etc. In React, the event handling system is very similar to handling events in DOM elements. The React event handling system is known as Synthetic Event, which is a cross-browser wrapper of the browser's native event.

58) What is Virtual DOM?

Virtual DOM is just a copy of the original DOM kept in the memory and synced with the real DOM by libraries such as ReactDOM. Virtual DOM has the same properties that of the Real DOM, but it lacks the power to directly change the content of the screen.

The Virtual DOM (VDOM) is an in-memory representation of Real DOM. The representation of a UI is kept in memory and synced with the "real" DOM. It's a step that happens between the render function being called and the displaying of elements on the screen. This entire process is called reconciliation.

59) What is context?

Context provides a way to pass data through the component tree without having to pass props down manually at every level. For example, authenticated user, locale preference, UI theme need to be accessed in the application by many components.

Eg: `const { Provider, Consumer } = React.createContext(defaultValue)`

60) What is children prop?

Children is a prop (`this.props.children`) that allow you to pass components as data to other components, just like any other prop you use. Component tree put between component's opening and closing tag will be passed to that component as children prop.

61) What are Life Cycle Methods in React?

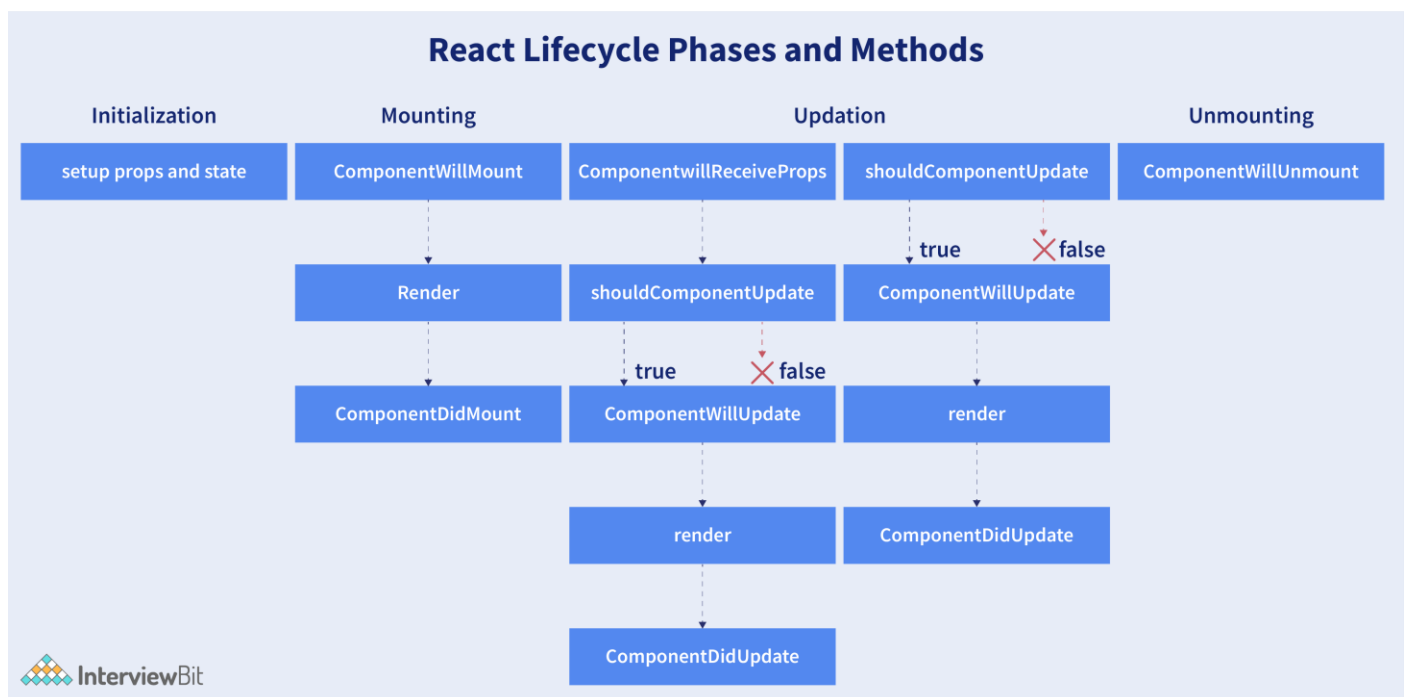
There are four different phases in the lifecycle of React component. They are:

Initialization: During this phase, React component will prepare by setting up the default props and initial state for the upcoming tough journey.

Mounting: Mounting refers to putting the elements into the browser DOM. Since React uses VirtualDOM, the entire browser DOM which has been currently rendered would not be refreshed. This phase includes the lifecycle methods `componentWillMount` and `componentDidMount`.

Updating: In this phase, a component will be updated when there is a change in the state or props of a component. This phase will have lifecycle methods like `componentWillUpdate`, `shouldComponentUpdate`, `render`, and `componentDidUpdate`.

Unmounting: In this last phase of the component lifecycle, the component will be removed from the DOM or will be unmounted from the browser DOM. This phase will have the lifecycle method named `componentWillUnmount`.



62) What are the limitations of React?

The few limitations of React are as given below:

- React is not a full-blown framework as it is only a library.
- The components of React are numerous and will take time to fully grasp the benefits of all.
- It might be difficult for beginner programmers to understand React.
- Coding might become complex as it will make use of inline templating and JSX.

63) What is the difference between Shadow DOM and Virtual DOM?

The Shadow DOM is a browser technology designed primarily for scoping variables and CSS in web components. The Virtual DOM is a concept implemented by libraries in JavaScript on top of browser APIs.

64) What is the difference between React and ReactDOM?

The react package contains `React.createElement()`, `React.Component`, `React.Children`, and other helpers related to elements and component classes. You can think of these as the isomorphic or universal helpers that you need to build components. The react-dom package contains `ReactDOM.render()`, and in react-dom/server we have server-side rendering support with `ReactDOMServer.renderToString()` and `ReactDOMServer.renderToStaticMarkup()`.

65) What is the purpose of ReactTestUtils package?

ReactTestUtils are provided in the with-addons package and allow you to perform actions against a simulated DOM for the purpose of unit testing.

66) What are the core principles of Redux?

Redux follows three fundamental principles:

- i. Single source of truth: The state of your whole application is stored in an object tree within a single store. The single state tree makes it easier to keep track of changes over time and debug or inspect the application.
- ii. State is read-only: The only way to change the state is to emit an action, an object describing what happened. This ensures that neither the views nor the network callbacks will ever write directly to the state.
- iii. Changes are made with pure functions: To specify how the state tree is transformed by actions, you write reducers. Reducers are just pure functions that take the previous state and an action as parameters, and return the next state.

67) How to access Redux store outside a component?

You just need to export the store from the module where it created with `createStore()`. Also, it shouldn't pollute the global window object.

68) What is the difference between React context and React Redux?

You can use Context in your application directly and is going to be great for passing down data to deeply nested components which what it was designed for. Whereas Redux is much more powerful and provides a large number of features that the Context API doesn't provide. Also, React Redux uses context internally but it doesn't expose this fact in the public API.

69) What is the difference between component and container in React Redux?

Component is a class or function component that describes the presentational part of your application. Container is an informal term for a component that is connected to a Redux store. Containers subscribe to Redux state updates and dispatch actions, and they usually don't render DOM elements; they delegate rendering to presentational child components.

70) What is the difference between HTML and React event handling?

i. In HTML, the event name usually represents in *lowercase* as a convention:

```
<button onclick='activateLasers() '>
```

Whereas in React it follows *camelCase* convention:

```
<button onClick={activateLasers}>
```

ii. In HTML, you can return `false` to prevent default behavior:

```
<a href='#' onclick='console.log("The link was clicked."); return false;' />
```

Whereas in React you must call `preventDefault()` explicitly:

```
function handleClick(event) {  
  event.preventDefault()  
  console.log('The link was clicked.')}
```

iii. In HTML, you need to invoke the function by appending `()` Whereas in react you should not append `()` with the function name. (refer "activateLasers" function in the first point for example)

72) What is Lifting State Up in React?

When several components need to share the same changing data then it is recommended to *lift the shared state up* to their closest common ancestor. That means if two child components share the same data from its parent, then move the state to parent instead of maintaining local state in both of the child components.

73) What are fragments?

It's a common pattern in React which is used for a component to return multiple elements. *Fragments* let you group a list of children without adding extra nodes to the DOM.

```
render() {  
  return (  
    <React.Fragment>  
      <ChildA />  
      <ChildB />  
      <ChildC />  
    </React.Fragment>  
  )  
}
```

There is also a *shorter syntax*, but it's not supported in many tools:

```
render() {  
  return (  
    <>  
      <ChildA />  
      <ChildB />  
      <ChildC />  
    </>  
  )  
}
```

74) Why fragments are better than container divs?

Below are the list of reasons,

- Fragments are a bit faster and use less memory by not creating an extra DOM node. This only has a real benefit on very large and deep trees.
- Some CSS mechanisms like *Flexbox* and *CSS Grid* have a special parent-child relationships, and adding divs in the middle makes it hard to keep the desired layout.
- The DOM Inspector is less cluttered.

75) Why should we not update the state directly?

If you try to update the state directly then it won't re-render the component.

```
//Wrong
this.state.message = 'Hello world'
```

Instead use `setState()` method. It schedules an update to a component's state object. When state changes, the component responds by re-rendering.

```
//Correct
this.setState({ message: 'Hello World' })
```

Note: You can directly assign to the state object either in *constructor* or using latest javascript's class field declaration syntax.

76) Differentiate React Hooks vs Classes.

React Hooks	Classes
It is used in functional components of React.	It is used in class-based components of React.
It will not require a declaration of any kind of constructor.	It is necessary to declare the constructor inside the class component.
It does not require the use of <code>this</code> keyword in state declaration or modification.	Keyword <code>this</code> will be used in state declaration (<code>this.state</code>) and in modification (<code>this.setState()</code>).
It is easier to use because of the <code>useState</code> functionality.	No specific function is available for helping us to access the state and its corresponding <code>setState</code> variable.
React Hooks can be helpful in implementing Redux and context API.	Because of the long setup of state declarations, class states are generally not preferred.

77) What is Jest?

Jest is a JavaScript unit testing framework created by Facebook based on Jasmine and provides automated mock creation and a jsdom environment. It's often used for testing components.

.....End.....