React

REACT JS INTERVIEW QUESTIONS

>>>

Q- What is React?

React is an open-source front-end JavaScript library that is used for building user interfaces, especially for single-page applications. It is used for handling the view layer for web and mobile apps. React was created by Jordan Walke, a software engineer working for Facebook. React was first deployed on Facebook's News Feed in 2011 and on Instagram in 2012.

Q-What are the major features of React?

The major features of React are:

- It uses VirtualDOM instead of RealDOM considering that RealDOM manipulations are expensive.
- Supports server-side rendering.
- Follows Unidirectional data flow or data binding.
- Uses reusable/composable UI components to develop the view.

Q- What are Pure Components?

React.PureComponent is exactly the same as *React.Component* except that it handles the shouldComponentUpdate() method for you. When props or state changes, *PureComponent* will do a shallow comparison on both props and state. *Component* on the other hand won't compare current props and state to next out of the box. Thus, the component will re-render by default whenever shouldComponentUpdate is called.

Q-What is the difference between state and props?

Both *props* and *state* are plain JavaScript objects. While both of them hold information that influences the output of render, they are different in their functionality with respect to components. Props get passed to the component similar to function parameters whereas state is managed within the component similar to variables declared within a function.

Q- What are props in React?

Props are inputs to components. They are single values or objects containing a set of values that are passed to components on creation using a naming convention similar to HTML-tag attributes. They are data passed down from a parent component to a child component.

The primary purpose of props in React is to provide following component functionality:

1. Pass custom data to your component.
2. Trigger state changes.
3. Use via this.props.reactProp inside component's render() method.

For example, let us create an element with reactProp property:
<Element reactProp={'1'} />
This reactProp (or whatever you came up with) name then becomes a property attached to React's native props object which originally already exists on all components created using React library.
props.reactProp

Q- What is the purpose of the callback function as an argument of setState()

The callback function is invoked when setState is finished and the component gets rendered. Since setState() is asynchronous the callback function is used for any post action.

Note: It is recommended to use the lifecycle method rather than this callback function.

```
setState({ name: 'John' }, () => console.log('The name has updated and component re-rendered'))
```

Q- What are synthetic events in React?

SyntheticEvent is a cross-browser wrapper around the browser's native event. Its API is the same as the browser's native event, including stopPropagation() and preventDefault(), except the events work identically across all browsers.