Your grade: 100%

Your latest: **100%** • Your highest: **100%** • To pass you need at least 70%. We keep your highest score.

Next item →

1. Which of the following describes JSX? 1 / 1 point

- (•) JavaScript XML, a syntax extension for JavaScript
- ( ) An error object
- ( ) A JavaScript library
- ( ) An add-on for JavaScript

Correct
Correct! JavaScript extends JSX to let you write HTML-like code inside of JavaScript. This code makes it easier for you to describe the structure of UI components.

2. What tool quickly sets up a new React project with current features like fast refresh and current module replacement? 1 / 1 point

- ( ) CRA
- ( ) Webpack
- ( ) Babel
- (•) Vite

Correct
Correct! JSX Vite allows you to set up new React apps quickly. It gives you a quick development server with hot module replacement and fast refresh, which makes it simple to make current React apps.

3. Which describes "one-way binding" in React? 1 / 1 point

- (•) Data can only flow from parent components to child components.
- ( ) Data can only flow from child components to parent components.
- ( ) Data can flow both ways between components.
- ( ) Data can't flow between components.

Correct
Correct! React only allows data to move from parent components to child components. Child components can't change their parent's data sent to them via props, so React's employs unidirectional data flow.

4. What role does a Virtual DOM have in React? 1 / 1 point

- ( ) It provides each React component its own DOM.
- (•) It changes just the parts of the DOM that need updating.
- ( ) It replaces the browser's native DOM with a virtual copy.
- ( ) It lets React components directly change the DOM.

Correct
Correct! The Virtual DOM in React is a small copy of the real DOM stored in memory. When you make changes to the UI, the Virtual DOM compares them to the real DOM and only updates the parts that changed. This makes the re-rendering process faster.

5. How does React send data from one class component to another? 1 / 1 point

- ( ) By using hooks
- ( ) By adding data from outside files
- (•) By using props
- ( ) By using global variables

Correct
Correct! In React, data can be sent from one class component to another using props. With props, parent components can send data to child components, which lets different parts of the program talk to each other.

6. In addition to componentDidMount(), which method does React invoke to mount a class component? 1 / 1 point

- ( ) componentWillUnmount()
- (•) componentWillMount()
- ( ) getDerivedStateFromProps()
- ( ) componentDidUpdate()

Correct
Correct! The two methods React uses when mounting a class component include componentWillMount() and componentDidMount().

7. Which method does React call when updating props to reflect that update in the component's state? 1 / 1 point

- (•) getDerivedStateFromProps()
- ( ) render()
- ( ) componentDidUpdate()
- ( ) getSnapshotBeforeUpdate()

Correct
Correct! You use the getDerivedStateFromProps() method if you have updated props and you want to reflect that update in the component's state.

8. What happens in React when you attempt to change a state object directly? 1 / 1 point

- ○ It issues a warning but doesn't change anything in the application.
- ○ It changes the state and re-renders the component.
- ◉ It doesn't change the state or cause a re-render.
- ○ Issues an error, and the application stops.

Correct
Correct! When you change the state object directly in React, you skip over the systems meant to discover the change in the state.

9. Which type of syntax does React use to update how the UI should behave when data or states change? 1 / 1 point

- ○ Imperative
- ◉ Declarative
- ○ JSX
- ○ HTML

Correct
Correct! React uses declarative syntax because it lets developers describe the UI state they want, and React updates the DOM to fit that state.

10. Which of the following shows JSX syntax? 1 / 1 point

- ○
```
const MyComponent = () => {
 <h1>Hello, World!</h1>
 <p>This is a JSX component.</p>
};
```
- ○
```
const MyComponent = () => {
 return (
   <h1>Hello, World!</h1>
   <p>This is a JSX component.</p>
 )
};
```
- ○
```
const MyComponent = () => {
 <>
   <h1>Hello, World!</h1>
   <p>This is a JSX component.</p>
 </>
};
```
- ◉
```
const MyComponent = () => {
 return (
   <>
     <h1>Hello, World!</h1>
     <p>This is a JSX component.</p>
   </>
 );
}
```

Correct
Correct! JSX syntax requires one common parent element. It can be a fragment or a tag.