H

Full Name: Email:

Test Name:

Taken On:

Time Taken:

Work Experience:

Invited by:

Invited on:

Tags Score:

Kenneth Choi

kennethichoi@gmail.com

MBA: React/Redux

1 Aug 2019 17:40:06 PDT

14 min 7 sec/ 25 min

1 years

Jeff

1 Aug 2019 17:35:54 PDT

Advanced 15/25

Essential 76.25/95

Flux 0/15

JSX 5/5

Javascript 15/15

React Class Components 47.5/55

React Functional Components 10/15

React Lifecycle Methods 15/15

React Render 20/30

React State 26.25/30

ReactJS 72.5/85

Redux Actions 5/5

Redux Reducers 5/5

Redux Thunk 5/5

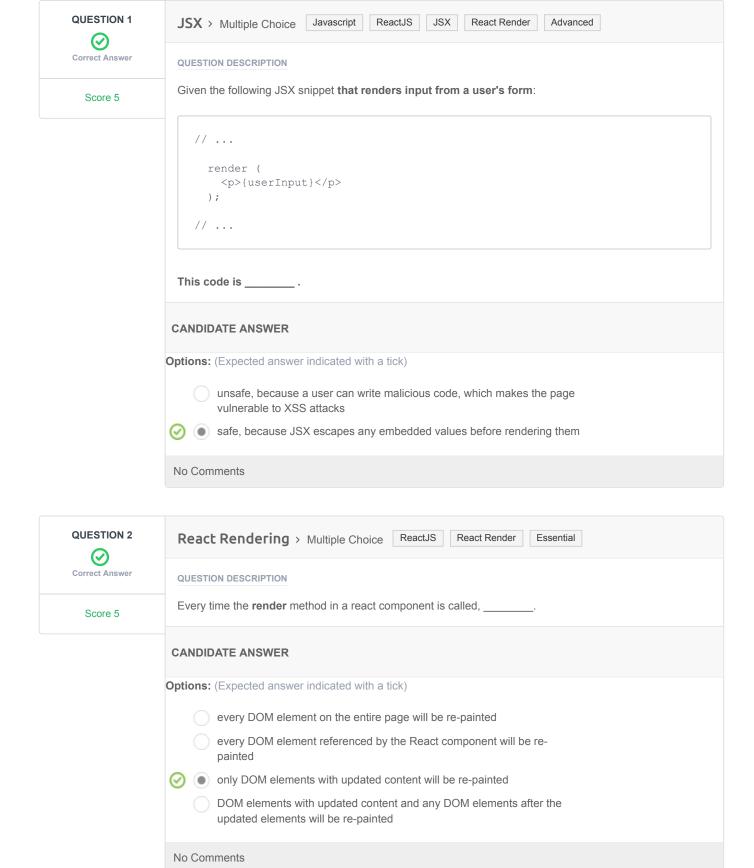
76% 91/120

scored in MBA: React/Redux in 14 min 7 sec on 1 Aug 2019 17:40:06 PDT

Recruiter/Team Comments:

	Question Description	Time Taken	Score	Status
Q1	JSX > Multiple Choice	38 sec	5/ 5	Ø
Q2	React Rendering > Multiple Choice	42 sec	5/ 5	⊘
Q3	React Debugging > Multiple Choice	1 min 1 sec	5/ 5	⊘
Q4	React Lifecycle Methods > Multiple Choice	16 sec	5/ 5	⊘
Q5	React Components > Multiple Choice	8 sec	5/ 5	⊘
Q6	React State > Multiple Choice	20 sec	5/ 5	Ø

Q7	React Lifecycle Methods > Multiple Choice	8 sec	5/ 5	Ø
Q8	React Props > Multiple Choice	2 min 30 sec	0/5	8
Q9	Redux Actions and Reducers > Multiple Choice	30 sec	5/ 5	⊘
Q10	Redux Thunk > Multiple Choice	23 sec	5/ 5	⊘
Q11	Redux > Multiple Choice	20 sec	5/ 5	⊘
Q12	React Component Methods > Multiple Choice	35 sec	5/ 5	⊘
Q13	React Diffing > Multiple Choice	1 min 8 sec	0/5	8
Q14	React Lifecycle Methods > Multiple Choice	1 sec	5/ 5	⊘
Q15	Flux > Sentence Completion	2 min 14 sec	0/ 15	8
Q16	React Diffing > Multiple Choice	6 sec	5/ 5	⊘
Q17	Redux Store > Multiple Choice	54 sec	3.75/ 5	⊘
Q18	Redux State Shape > Multiple Choice	38 sec	5/ 5	⊘
Q19	React > Multiple Choice	42 sec	5/ 5	⊘
Q20	React Best Practices > Multiple Choice	16 sec	5/ 5	⊘
Q21	React Stateful Components > Multiple Choice	25 sec	2.5/ 5	⊘
Q22	Redux > Multiple Choice		5/ 5	⊘





Score 5

React Debugging > Multiple Choice ReactJS React Class Components

QUESTION DESCRIPTION

The following component should show a number and a button. When a user clicks the button, the number should increment. In its current state, the component is throwing an error:

What is wrong with this component?

```
import React, { Component } from 'react';
class Counter extends Component {
  constructor() {
   super();
   this.state = {
     num: 0,
    };
 handleClick() {
   const num = this.state.num + 1;
    this.setState({ num });
 render() {
   return (
     <div>
       <h1>{ this.state.num }</h1>
       <button onClick={this.handleClick()}>Add 1
     </div>
    );
export default Counter;
```

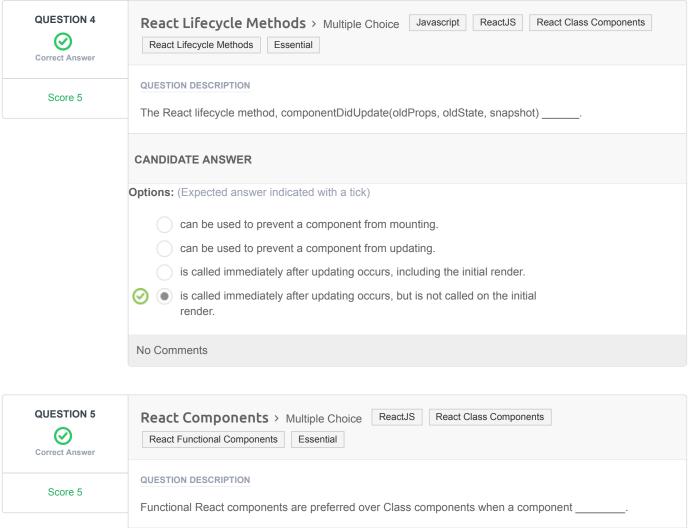
CANDIDATE ANSWER

Options: (Expected answer indicated with a tick)

- A. There is no 'componentDidCatch()' to prevent errors in the component.
- B. 'handleClick()' should be updating props instead of state.
- C. 'handleClick()' is not bound to the instance of the component.
- D. 'handleClick()' is being called on every render, which triggers a rerender, resulting in an infinite loop.



E. Both C and D need fixed before this component will work.





Score 5

React State > Multiple Choice ReactJS React Class Components

QUESTION DESCRIPTION

Given the following react component, what is wrong with the onClick function in the button?

React State

Essential

class Example extends React.Component { constructor() { super(); this.state = { color: 'orange' }; render() { return (<div> <h1>Color: {this.state.color}</h1> <button onClick={() => (this.state.color = 'blue')}> Change color to blue </button>

CANDIDATE ANSWER

);

</div>

Options: (Expected answer indicated with a tick)

- Nothing. The onClick function will update the state and trigger a re-render.
- Context. The onClick function needs to be bound to the instance of the class inside the constructor.
- Wrong variable. It should be `this.props.color = 'blue'`.

Mutating state. It should call `this.setState` to update the state.

QUESTION 7 Correct Answer	React Lifecycle Methods > Multiple Choice ReactJS React Lifecycle Methods React Class Components Essential		
Score 5 QUESTION DESCRIPTION The lifecycle method, componentDidMount(),			
	CANDIDATE ANSWER		
	Options: (Expected answer indicated with a tick)		
	will be called every time the component re-renders.		
	is only called after a component has finished mounting, but not when it re- renders.		
	is called after a component has finished mounting AND when it rerenders.		
	is called when updated props are passed down to a component.		
	No Comments		



Score 0

React Props > Multiple Choice ReactJS

React Render

React Class Components

Advanced

QUESTION DESCRIPTION

Given the following React components:

```
// color-title.jsx
import React from 'react';
class ColorTitle extends React.Component {
 constructor(props) {
   super (props);
   this.state = {
     color: this.props.color
   };
 render() {
     <h1>Color: {this.state.color}</h1>
   );
export default ColorTitle;
```

Upon the initial render, the parent container passes down the following props: { color: 'blue' }, and the <h1> inside of the ColorTitle component will have the text "Color: blue". Assuming that the ColorTitle component is NOT unmounted, what color will the <h1> text be after the new props: { color: 'yellow' } are passed down, and why?

CANDIDATE ANSWER

Options: (Expected answer indicated with a tick)

- "Color: yellow", because changing the state of the parent component will re-render its children, causing the ColorTitle to un-mount and re-mount with the updated props.
- "Color: yellow", because passing down new props into the ColorTitle component will update its state.
- "Color: blue", because 'this.changeColor' needs to be bound to the class instance in the constructor.
- 'Color: blue", because passing down new props into the ColorTitle component will not update its state.

Redux Actions and Reducers > Multiple Choice Redux Redux Actions Redux Reducers Essential QUESTION DESCRIPTION When an action is dispatched, what happens in regards to the reducers? // Example Action const exampleAction = { type: "DO_THINGS", payload: { key: 'val' }, }; CANDIDATE ANSWER Options: (Expected answer indicated with a tick) The action is deconstructed and compared with each reducer's state until

there is a match for the data contained in the payload.

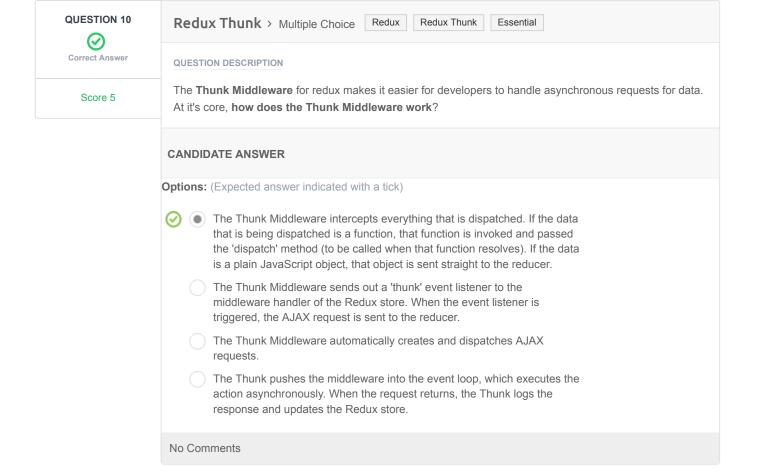
The action is sent to the components' this.setState() method.

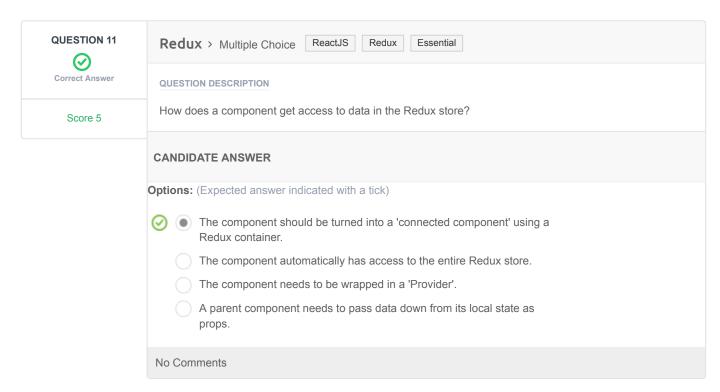
The action is sent ONLY to the reducer with a case that matches the

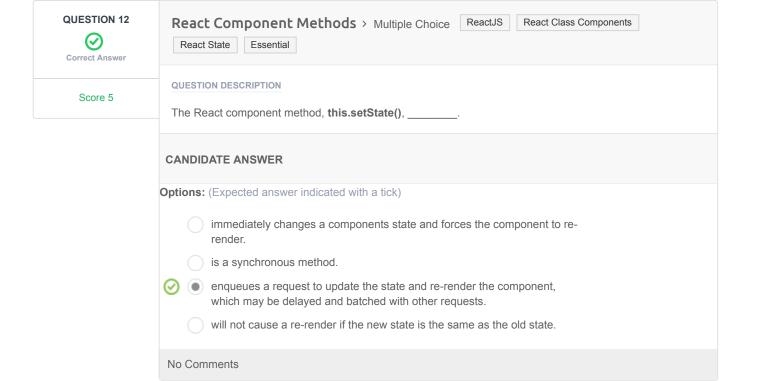
The action is sent to EVERY reducer, but new slices of state are only returned for reducers that are designed to match the action's `type`.

No Comments

action's `type`.







Wrong Answer

Score 0

React Diffing > Multiple Choice ReactJS React Functional Components

React Render

QUESTION DESCRIPTION

Advanced

Given the following react components:

```
// parent.jsx
import React from 'react';
import MyFootercomponent from './my_footer_component';
const Parent = ({ num }) => (
  <article>
   <heading>
     <h1>Title: </h1>
     {p>{num}
   </heading>
   >
     Numbers are very important....
    <MyFooterComponent />
  </article>
export default Parent;
```

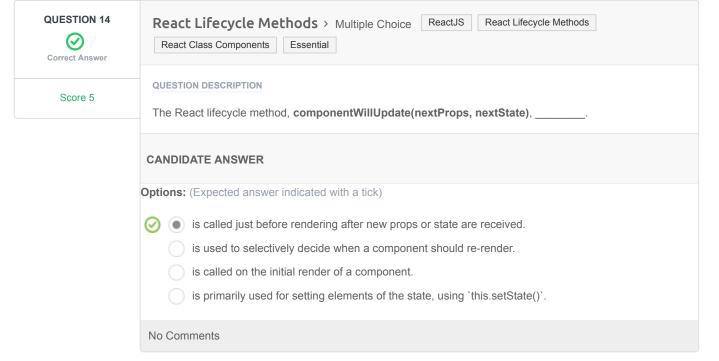
```
// my_footer_component.jsx
export default () => (
 Words are also important...
);
```

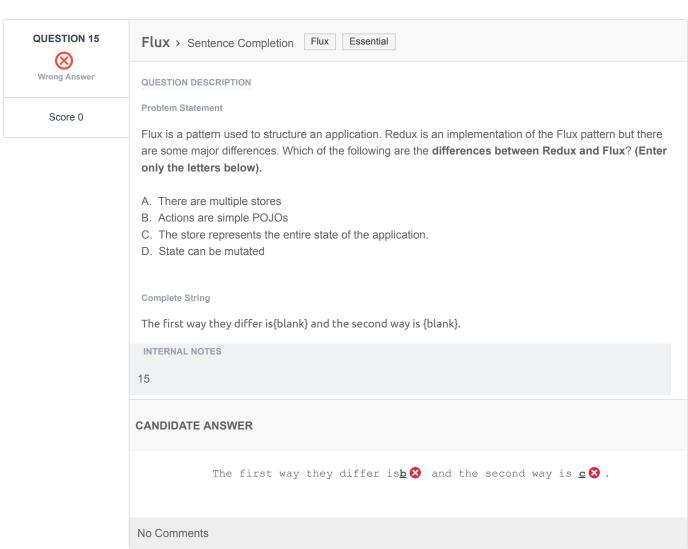
When a new num is passed down as props to the Parent component, what happens to the <MyFooterComponent /> component?

CANDIDATE ANSWER

Options: (Expected answer indicated with a tick)

- <MyFooterComponent /> WILL unmount, re-mount, re-render, and the coinciding DOM element WILL be re-painted.
- <MyFooterComponent /> WILL re-render and the coinciding DOM element WILL be re-painted.
- <MyFooterComponent /> WILL re-render, but the coinciding DOM element WILL NOT be re-painted.
 - <MyFooterComponent /> WILL NOT re-render, and the coinciding DOM element WILL NOT be re-painted.





 \bigcirc Correct Answer

Score 5

React Diffing > Multiple Choice React Render

QUESTION DESCRIPTION

ReactJS

Advanced

Given the following React component:

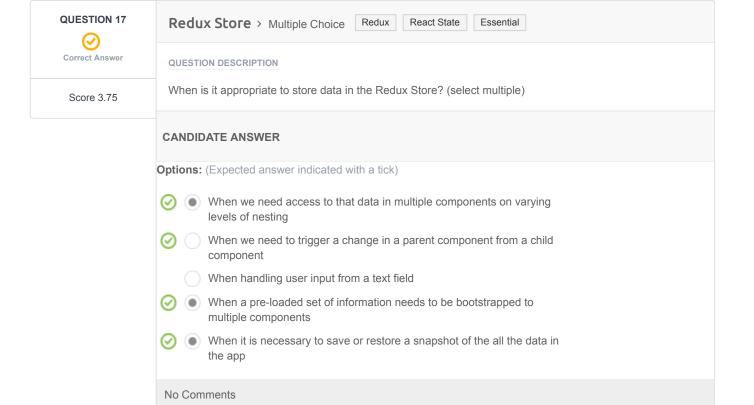
```
import React from 'react';
import MyFootercomponent from './MyFooterComponent';
export default (\{num\}) => (
 <article>
   <h2>{num}</h2>
     Numbers are very important....
   <MyFooterComponent />
  </article>
```

Every time a new num is passed down in the props, which elements/components will be re-painted in the DOM?

CANDIDATE ANSWER

Options: (Expected answer indicated with a tick)

- every DOM element (the <article> and its children) will be re-painted on
- the <h2> element and every element/component after the <h2> (and <MyFooterComponent />) will be re-painted on the page
- only the <h2> element will be repainted
 - the <h2> element and the <MyFooterComponent /> child component will be re-painted





Score 5

Redux State Shape > Multiple Choice Redux

QUESTION DESCRIPTION

React State

Essential

What about the below Redux state shape does not follow best practices? (Select Multiple)

```
posts: {
   1: {
     id: 1,
     author: { username: 'user1', name: 'User 1' },
     body: 'post1',
     comments: [
       {
         id: 1,
         author: { username: 'user2', name: 'User 2' },
         comment: 'comment1'
       },
         id: 2,
         author: { username: 'user1', name: 'User 1' },
         comment: 'comment2'
     ]
   }
}
```

CANDIDATE ANSWER

Options: (Expected answer indicated with a tick)

- The state shape isn't flat.
 - The 'posts' slice of state should return an array containing post objects.
- The 'posts' slice of state should be nested under an 'entities' key to mark it as a table in the database.
- There should additionally be a 'users' slice of state and a 'comments' slice of state to prevent duplicating data.



Score 5

QUESTION DESCRIPTION

Below is a React component that displays a number and a button. Every time the button is clicked, the number should increase by 1. In its current state, it does not work as intended.

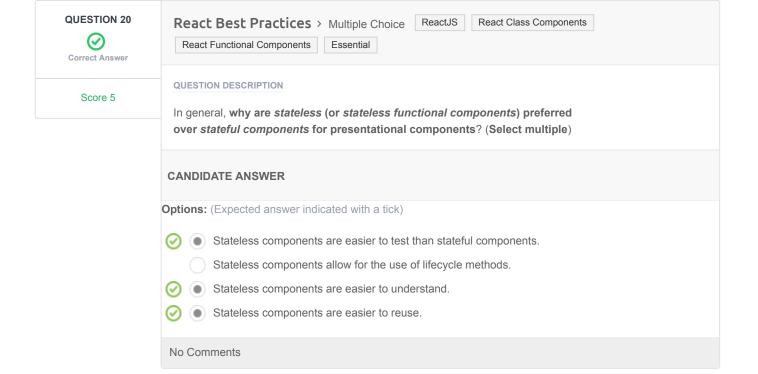
What needs to be changed in order for this component to work as planned?

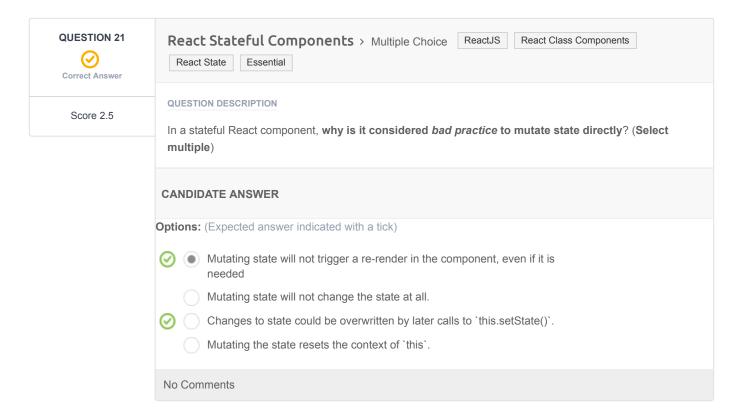
```
import React, { Component } from 'react';
class Counter extends Component {
  constructor() {
   super();
   this.state = {
     num: 0,
    };
 handleClick() {
   this.setState({ num: this.state.num + 1 });
  render() {
   return (
        <h1>{ this.state.num }</h1>
       <button onClick={this.handleClick}>Add 1</button>
      </div>
    );
export default Counter;
```

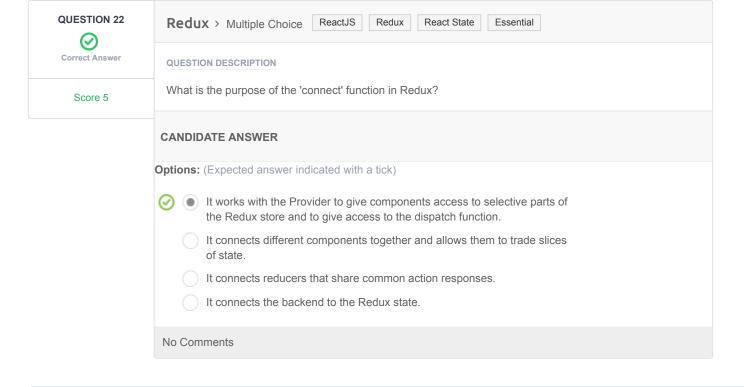
CANDIDATE ANSWER

Options: (Expected answer indicated with a tick)

- handleClick() should be updating props instead of updating state.
- handleClick() needs to be bound to the instance of the class.
 - Inside of handleClick(), 'this.state.num + 1' needs to be performed outside of the object being passed into 'this.setState()'
 - 'onClick' should be 'onMouseDown'.







PDF generated at: 2 Aug 2019 00:54:58 UTC