**1. When might you use React.PureComponent?**

**A.** when you do not want your component to have props

**B.** when you have sibling components that need to be compared

**C.** when you want a default implementation of `shouldComponentUpdate()

**D.** when you do not want your component to have state

**2. You have written the following code but nothing is rendering. How do you fix this problem?**

const Heading = () => {

<h1>Hello!</h1>;

};

**A.** Add a render function.

**B.** Change the curly braces to parentheses or add a return statement before the h1 tag.

**C.** Move the h1 to another component.

**D.** Surround the h1 in a div.

**3. Which option is correct for State vs Props**

**A.** Props is something that the parent doesn’t need and decide to throw around among other parents; State is

the parent’s favorite child and something the component wants to nurture.

**B.** Props is a copy of real DOM; State is the definition of the real DOM.

**C.** Props get passed to the component using naming conventions, like a function parameter; State is managed

within the component and holds some information that may change over the lifetime of the component.

**D.** Prop is managed within the component and holds some information that may change over the lifetime of the

component; State gets passed to the component, like a function parameter

**4. What package contains the render() function that renders a React element tree to the DOM?**

**A.**React

**B.**ReactDOM

**C.**Render

**D.**DOM

**5. Which of the following click event handlers will allow you to pass the name of the person to be**

**hugged?**

class Huggable extends React.Component {

hug(id) {

console.log("hugging " + id);

}

render() {

let name = "kitteh";

let button = // Missing Code

return button;

}

}

**A.** <button onClick={(name) => this.hug(name)}>Hug Button</button>

**B.** <button onClick={this.hug(e, name)}>Hug Button</button>

**C.** <button onClick={(e) => hug(e, name)}>Hug Button</button>

**D.** <button onClick={(e) => this.hug(name,e)}>Hug Button</button>

**6. Which answer best describes a function component?**

**A.** A function component is required to create a React component.

**B.** A function component accepts a single props object and returns a React element.

**C.** A function component is the only way to create a component.

**D.** A function component is the same as a class component.

**7. What is the name of the tool used to take JSX and turn it into createElement calls?**

**A.** JSX Editor

**B.** ReactDOM

**C.** Browser Buddy

**D.** Babel

**8. If you see the following import in a file, what is being used for state management in the component?**

import React, {useState} from 'react';

**A.** React Hooks

**B.** stateful components

**C.** math

**D.** class components

**9. If you created a component called Dish and rendered it to the DOM, what type of element would be**

**rendered?**

function Dish() {

return <h1> Mac and Cheese</h1>;

}

ReactDOM.render(

<Dish />

document.getElementById('root')

)

**A.** div

**B.** section

**C.** component

**D.** h1

**10. Which of the following lifecycle methods does not get triggered on the component’s initial render?**

**A.** componentWillMount()

**B.** componentWillReceiveProps()

**C.** render()

**D.** componentDidMount()

**11. What does this React element look like given the following function?**

React.createElement(

"h1",

null,

"What's happening?"

);

**A.** <h1 props={null}>What's happening?</h1>

**B.** <h1 id="component">What's happening?</h1>

**C.** <h1>What's happening?</h1>

**D.** <h1 id="element">What's happening?</h1>

**Interview Questions 15m**

1. **What are controlled components?**

[**https://tr.reactjs.org/docs/forms.html#controlled-components**](https://tr.reactjs.org/docs/forms.html#controlled-components)

**[https://medium.com/frkntplglu/controlled-vs-uncontrolled-component-](https://medium.com/frkntplglu/controlled-vs-uncontrolled-component-hangisini-kullanal%C4%B1m-c3637d95d304" \l ":~:text=Basit%C3%A7e%20bir%20componentimizde%20form%20elemanlar%C4%B1m%C4%B1z%C4%B1n,y%C3%B6netiyorsak%20bu%20controlled%20component%20oluyor.)**

1. **What is a higher order component?**

[**https://tr.reactjs.org/docs/higher-order-components.html**](https://tr.reactjs.org/docs/higher-order-components.html)

Parametre olarak bir component alan, istediğimiz değerleri componentin argümanları (props) ile birlikte yeni bir component’e aktaran ve bu componenti geriye döndüren bir metot yazdık. Yani bir HOC yazmış olduk.

**3. How Virtual-DOM is more efficient than Dirty checking?**