

TW-16 STUDENT VERSION



CLARUSWAY
WAY TO REINVENT YOURSELF

Meeting Agenda

- ▶ Icebreaking
- ▶ Questions
- ▶ Interview Questions
- ▶ Coffee Break
- ▶ Coding Challenge
- ▶ Video of the week
- ▶ Retro meeting
- ▶ Case study / project

Teamwork Schedule

Ice-breaking

10m

- Personal Questions (Study Environment, Kids etc.)
- Any challenges (Classes, Coding, studying, etc.)
- Ask how they're studying, give personal advice.
- Remind that practice makes perfect.

Ask Questions

15m

1. If you want to import just the Component from the React library, what syntax do you use?

- A. import React.Component from 'react'
- B. import [Component] from 'react'
- C. import Component from 'react'
- D. import { Component } from 'react'**

2. How do you fix the syntax error that results from running this code?

```
const person =(firstName, lastName) =>
{
  first: firstName,
  last: lastName
}
console.log(person("Jill", "Wilson"))
```

- A. Wrap the object in parentheses.**
- B. Call the function from another file.
- C. Add a return statement before the first curly brace.
- D. Replace the object with an array.

3. 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

4. Using object literal enhancement, you can put values back into an object. When you log person to the console, what is the output?

```
const name = "Rachel";  
const age = 31;  
const person = { name, age };  
console.log(person);
```

- ☐ A. {{name: "Rachel", age: 31}}
- ☒ B. {name: "Rachel", age: 31}
- ☐ C. {person: "Rachel", person: 31}
- ☐ D. {person: {name: "Rachel", age: 31}}

5. To get the first item from the array ("cooking") using array destructuring, how do you adjust this line?

```
const topics = ["cooking", "art", "history"];
```

- ☐ A. const first = ["cooking", "art", "history"]
- ☐ B. const [] = ["cooking", "art", "history"]
- ☐ C. const [, first] = ["cooking", "art", "history"]
- ☒ D. const [first] = ["cooking", "art", "history"]

6. How do you handle passing through the component tree without having to pass props down manually at every level?

- ☐ A. React Send
- ☐ B. React Pinpoint
- ☐ C. React Router
- ☒ D. React Context

7. Why might you use `useReducer` over `useState` in a React component?

- A. when you want to replace Redux
- B.** when you need to manage more complex state in an app
- C. when you want to improve performance
- D. when you want to break your production app

8. 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

9. What does this React element look like given the following function? (Alternative: Given the following code, what does this React element look like?)

```
React.createElement("h1", null, "What's happening?");
```

- A. `<h1 props={null}>What's happening?</h1>`
- B.** `<h1>What's happening?</h1>`
- C. `<h1 id="component">What's happening?</h1>`
- D. `<h1 id="element">What's happening?</h1>`

10. What do you call the message wrapped in curly braces below?

```
const message = "Hi there";  
const element = <p>{message}</p>;
```

- A. a JS function
- B. a JS element
- C.** a JS expression
- D. a JSX wrapper

11. What is the difference between the click behaviors of these two buttons (assuming that `this.handleClick` is bound correctly)?

- A. `<button onClick={this.handleClick}>Click Me</button>`
- B. `<button onClick={event => this.handleClick(event)}>Click Me</button>`

- A. Button A will not have access to the event object on click of the button.
- B. Button B will not fire the handler `this.handleClick` successfully.
- C. Button A will not fire the handler `this.handleClick` successfully.
- D.** There is no difference.

12. How do you destructure the properties that are sent to the Dish component?

```
function Dish(props) {  
  return (  
    <h1>  
      {props.name} {props.cookingTime}  
    </h1>  
  );  
}
```

- A. `function Dish([name, cookingTime]) { return <h1>{name} {cookingTime}</h1>; }`
- B.** `function Dish({name, cookingTime}) { return <h1>{name} {cookingTime}</h1>; }`
- C. `function Dish(props) { return <h1>{name} {cookingTime}</h1>; }`
- D. `function Dish(...props) { return <h1>{name} {cookingTime}</h1>; }`

13. Why is it important to avoid copying the values of props into a component's state where possible?

- A. because you should never mutate state
- B. because `getDerivedStateFromProps()` is an unsafe method to use
- C.** because you want to allow a component to update in response to changes in the props
- D. because you want to allow data to flow back up to the parent

14. What is the children prop?

- A. a property that adds child components to state
- B.** a special property that JSX creates on components that contain both an opening tag and a closing tag, referencing its contents.
- C. a property that lets you set an array as a property
- D. a property that lets you pass data to child elements

15. Which of these terms commonly describe React applications?

- A.** declarative
- B.** integrated
- C.** closed
- D.** imperative

Text

**Interview Questions****15m**

- 1. What is the difference between state and props?**
- 2. What are the lifecycle methods of ReactJS class components?**
- 3. What are the different phases of ReactJS component lifecycle?**
- 4. What is prop drilling and how can you avoid it?**

Coding Challenge**15m**

There will be no code challenge this week. Students will catch up on any React topic gaps.

**Coffee Break****10m****Case study/Project****15m**

This week there won't be any projects. Students will catch up on any React topic gaps.

Retro Meeting on a personal and team level

10m

Ask the questions below:

- What went well?
- What could be improved?
- What will we commit to do better in the next week?

Closing

5m

-Next week's plan

-QA Session
