MODULE-2 TW-03 TEAM LEAD VERSION

TEAMWORK

CLARUSWAY
WAY TO REINVENT YOURSELF

Meeting Agenda

- Icebreaking
- Workshop Activities - Tuesday
- Teamwork Activities - Friday
  - Questions
  - Interview Questions
- Video of the week
- Retro meeting
- Case study / project

Teamwork Schedule

Ice-breaking 10m

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

Workshop Activities (Tuesday) 90m

- Our expectation is the completion and understanding of the Hospital Appointment application.
- Product List *(Optional Project)*

Team Work Activities (Friday) 90m

Ask Questions 20m

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'`

Answer: D

2. 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>`

Answer: B

3. When do you use `useLayoutEffect`?

A. to optimize for all devices
B. to complete the update
C. to change the layout of the screen
D. when you need the browser to paint before the effect runs

Answer: B

4. 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>; }`

Answer: B

5. 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

Answer: C

6. What is the children prop?

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

Answer: D

7. Which library does the `fetch()` function come from?

A. FetchJS
B. ReactDOM
C. No library. `fetch()` is supported by most browsers.
D. React

Answer: C

8. What is Axios in the context of React.js?

A. A styling library
B. A state management tool
C. A JavaScript library for making HTTP requests
D. A routing library

Answer: C

9. How do you install Axios in a React project?

A. npm install react-axios
B. npm install axios
C. npm install react-http
D. npm install http-axios-react

Answer: B

10. How do you handle errors in an Axios request?

A. Using the error method
B. Using the fail method
C. Using the handleError method
D. Using the catch method

Answer: D

11. How can you pass parameters in an Axios POST request in React?

A. Using the data property
B. Using the params property
C. Using the body property
D. Using the payload property

Answer: A

12. In React, what is a common approach to managing Axios requests in functional components?

A. Utilizing the useEffect hook
B. Using the componentDidMount lifecycle method
C. Embedding Axios calls directly in the component body
D. Creating a separate class for Axios requests

Answer: A

13. What is the primary advantage of using Axios over the traditional fetch API in React?

A. Axios provides a simpler syntax
B. Axios is faster in making HTTP requests
C. Axios has built-in support for interceptors and cancel tokens
D. Axios is the official HTTP library recommended by the React team

Answer: C

Interview Questions 15m

1. What are Custom Hooks?

Answer :
A Custom Hook is a function in Javascript whose name begins with 'use' and which calls other hooks. It is a part of React v16.8 hook update and permits you for reusing the stateful logic without any need for component hierarchy restructuring.

In almost all of the cases, custom hooks are considered to be sufficient for replacing render props and HoCs (Higher-Order components) and reducing the amount of nesting required. Custom Hooks will allow you for avoiding multiple layers of abstraction or wrapper hell that might come along with Render Props and HoCs.

The disadvantage of Custom Hooks is it cannot be used inside of the classes.is unnecessary to bind 'this' inside the constructor when using an arrow function. This prevents bugs caused by the use of 'this' in React callbacks.

2. What are the rules that must be followed while using React Hooks?

Answer :
There are 2 rules which must be followed while you code with Hooks:

- React Hooks must be called only at the top level. It is not allowed to call them inside the nested functions, loops, or conditions.
- It is allowed to call the Hooks only from the React Function Components.

3. What is Axios, and how does it simplify making HTTP requests in React applications?

Answer :

Axios is a third-party JavaScript library that simplifies making HTTP requests in both browsers and Node.js. In React, Axios is often preferred for its simplicity, ease of use, and features like automatic JSON parsing, request/response interception, and better error handling compared to the native fetch function.

4. How do you handle asynchronous operations with fetch in a React component?

Answer :
You typically use fetch within the componentDidMount lifecycle method or with the useEffect hook in functional components. You can then handle the response asynchronously using .then() and .catch() or with async/await.

Example using fetch with useEffect in a functional component:

```
import { useEffect } from 'react';

function MyFunctionalComponent() {
  useEffect(() => {
    fetch('https://api.example.com/data')
      .then(response => response.json())
      .then(data => console.log(data))
      .catch(error => console.error('Error:', error));
  }, []);
  // ... rest of the component
}
```

☕

Coffee Break 10m

☕

Retro Meeting on a personal and team level 5m

Ask the questions below:

- What went well?
- What went wrong?
- What is the improvement areas?

Case study/Project 15m

TEAMWORK - Tutorial App - Edit (20.00-21.30)

Closing 5m

- Next week's plan
- QA Session
