Week 14: Research

1. Considering React, what is one-way, or unidirectional data flow? Why is it important?

One-way, or unidirectional data flow is a technique that is mainly found in functional reactive programming. It means that the data has one, and only one way to be transferred to other parts of the application. The data from the parent is known as props. You can only transfer data from parent to child and not vice versa. To update the view, you need to update the model first, and then redraw the view. Unidirectional data flow is important because it has many benefits, such as: Debugging: One-way data flow makes debugging much easier. When the developer knows where the data is coming from and where it is going, they can use tools to find the problems more efficiently; Better control: Having data flow in one direction makes the program less prone to errors and gives the developer more control. It simplifies the debugging process and improves application stability, as data flow is predictable and easier to understand; Efficiency: As the used libraries are wary of the limitations and specifications of the unidirectional flow, extra resources are not wasted, leading to an efficient process. React makes the view redrawing process extremely efficient because it compares not the actual DOM but a virtual DOM it keeps on memory.

2. When does a React component render?

A React component renders when it is first mounted to the DOM, and when it receives new props or state. The render() method returns a React element that describes what the component should look like. React then updates the actual DOM to match the virtual DOM returned by the render() method. The render() method is also responsible for updating the DOM whenever there is a change to internal component data. React uses a diffing algorithm to compare the virtual DOM with the actual DOM and only updates the parts that have changed. This makes rendering in React efficient and fast.

3. What is your favorite thing you learned this week?

Exploring the fundamental concepts and functionality of React has been an engaging experience. Overall, it has been a valuable learning opportunity that has expanded my knowledge and skills in front-end development.

URL:

https://www.geeksforgeeks.org/unidirectional-data-flow/

https://www.educative.io/answers/what-is-unidirectional-data-flow-in-react

https://stackoverflow.com/questions/34565568/two-way-data-binding-angular-vs-one-way-data-flow-react-flux

https://react.dev/reference/react-dom/render

https://legacy.reactjs.org/docs/rendering-elements.html