Zahava Gopin

WEB 425: discussion 6.1

6/26/2023

Angular Input and Output

In Angular, an input property allows data to flow into a component from its parent component. By decorating a property with the @Input() decorator, we make it an input property. This allows the parent component to bind data to the child component's property using property binding syntax in the template. The child component can then use this data for various purposes, such as displaying information or performing calculations. “In a child component or directive, use the @Input() decorator to inform Angular that a property in that component can get its value from its parent component. It's important to remember that the data flow is from the viewpoint of the child component. As a result, a @Input() allows data from the parent component to be input into the child component” (Jain, 2022).

On the other hand, output properties allow a child component to emit events to its parent component. By using the @Output() decorator and an instance of the EventEmitter class, we can define an output property in a child component. This enables the child component to raise events and notify its parent component about certain actions or changes.

“@Input() and @Output() act as the API, or application programming interface, of the child component in that they allow the child to communicate with the parent. Think of @Input() and @Output() like ports or doorways—@Input() is the doorway into the component allowing data to flow in while @Output() is the doorway out of the component, allowing the child component to send data out” (Angular, n.d.).

Resources:

Jain, S. (2022, December 16). *Input and Output Directives in Angular*. Scaler Topics. https://www.scaler.com/topics/angular/input-and-output-directives-in-angular/

‌

*Angular*. (n.d.). Docs.angular.lat. Retrieved June 26, 2023, from https://docs.angular.lat/guide/inputs-outputs

‌