Janis Gonzalez

WEB 425 Angular with TypeScript

June 29, 2023

Discussion 6.1 Input and Output Properties

Angular is a popular JavaScript framework that is used for being able to create dynamic web applications. One of the key features is the ability to easily exchange data between components using input and output properties.

Input properties allow data to be passed into a component from the parent component. By using the @Input decorator, a component can define a property that can be set by the parent component (@Input() and @Output() properties). This allows for flexible and reusable components, as the parent component can customize the behavior or appearance of the child component by setting different values for its input properties (@Input() and @Output() properties). Output properties, on the other hand, enable a component to emit events to its parent component. This is done using the @Output decorator and the EventEmitter class (EventEmitter). The child component can then define an output property and emit events using the emit() method of the EventEmitter (EventEmitter). The parent component can then bind to this output property and listen for the emitted events, allowing it to react to it and perform the appropriate actions needed (EventEmitter). The @Input() property is writable, meanwhile the @Output() property is observable (@Input() and @Output() properties). The combination of input and output properties creates a powerful communication mechanism between components in Angular. It promotes the principle of component-based architecture and thus enables the building of modular, reusable, and highly interactive applications (@Input() and @Output() properties).

By leveraging input and output properties, developers can create components that are decoupled from each other, making it easier to understand, test, and maintain the code. It also facilitates collaboration among its team members, as many different developers can work on separate components independently and then integrate them easily using the input and output properties (@Input() and @Output() properties). Angular's input and output properties provide a powerful means of communication between components, enhancing the flexibility and reusability of Angular applications (@Input() and @Output() properties).

References

*@Input() and @Output() properties*. Angular. (n.d.-b). https://docs.angular.lat/guide/inputs-outputs

*EventEmitter*. Angular. (n.d.-a). https://angular.io/api/core/EventEmitter