Tyson Martin

COM-391

Prof. Estey

Activity 2

03/16/2025

**Angular & Music App Guide**

**Part 1: Angular Components, Forms, and Data Binding**

Setting Up Angular Application

1. Create a new Angular application called simpleapp with routing and CSS enabled.
2. Set up your Visual Studio Code workspace and open the project.
3. Serve the Angular application locally so it opens automatically in your browser.
4. Install Bootstrap and Popper.js to style the application.
5. Update angular.json to include Bootstrap’s CSS and JS in the appropriate arrays.

Creating Shop Component with Reactive Form

1. Generate a new component called shop.
2. Add two properties: one for a question (e.g., “What’s your name?”) and one for the answer (default to “unknown”).
3. Use a model-driven (reactive) form with a form control bound to the answer input field.
4. Add an onSubmit method to log and store the submitted name.
5. Display the answer in a label only if it is not “unknown”.
6. Import ReactiveFormsModule in the app module.

Creating Info Component with Template Form

1. Generate a new component called info.
2. Add three properties: quantity, a product list, and selectedProduct.
3. Use a template-driven form to bind user input to these properties using two-way data binding.
4. Add submit and reset buttons to trigger methods that log or reset the input data.
5. Pass the answer from shop to info using the @Input() decorator.
6. Conditionally show the info component only when a name has been entered.
7. Import FormsModule in the app module.

**Research Questions**

1. **What is @Input (used in info.component.ts)?**

The @Input() decorator in Angular allows a child component to receive data from its parent component. This is part of Angular’s component communication mechanism. When a parent component binds a value to a property using [propertyName]="value", the child can access that value if the property is decorated with @Input. For example, in this project, the ShopComponent passes a name to InfoComponent using @Input() so it can greet the user dynamically.

1. **What is [value] (used in info.component.html)?**

[value] is Angular’s property binding syntax. It binds a DOM element’s value attribute to a component’s property or expression. In a dropdown list (<select>), [value]="product" binds the current option to the product variable, so when a user selects it, Angular knows which item was selected. This is dynamic compared to hardcoding the value attribute.

1. **What is [(ngModel)] (used in info.component.html)?**

[(ngModel)] is Angular’s two-way data binding syntax. It binds the input field or control to a component variable, keeping both in sync. If the user changes the input in the form, the component’s property automatically updates. Similarly, if the component updates the property programmatically, the input field reflects the change. This is particularly useful for forms and user input.