

Lab 3 Understanding

These lessons helped me understand how Vue handles logic, structure, and communication in a clean way. Computed properties stood out because they let Vue handle calculations for us instead of putting that logic directly in the template. For example, instead of combining the brand and product name in the HTML, a computed property like `title` does that work. The same idea applies to `image` and `inStock`, which automatically update based on the selected variant. I like that Vue only recalculates these values when the data they depend on changes, which makes the app more efficient.

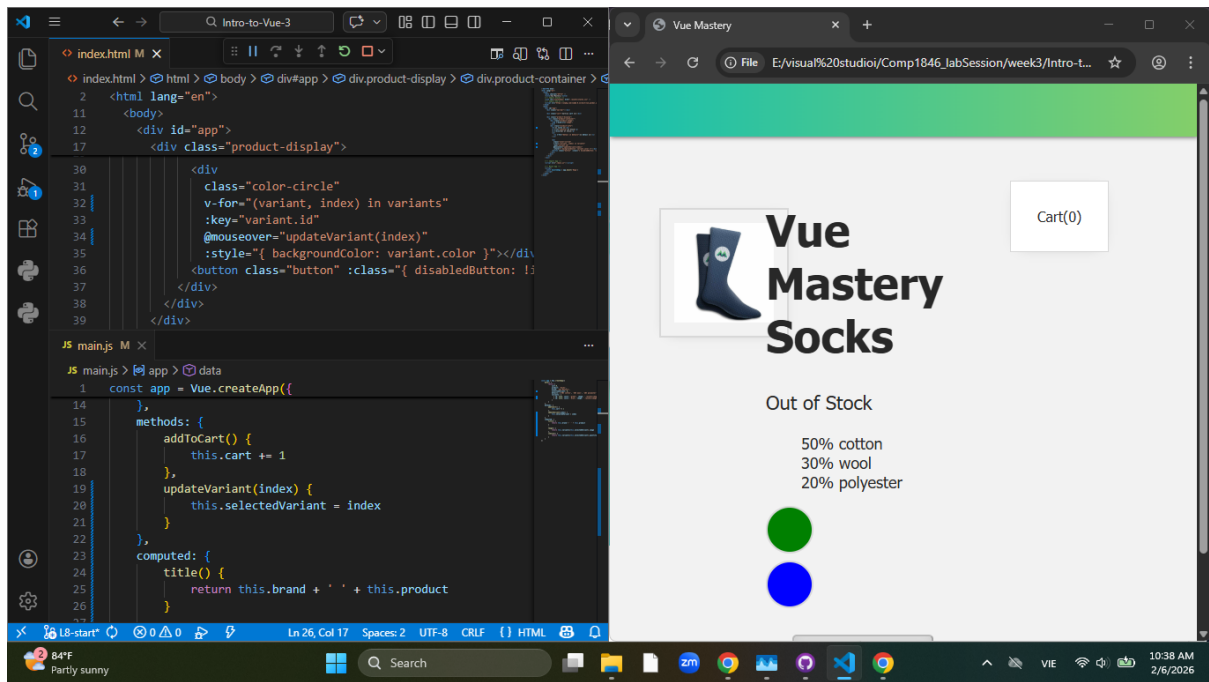
Breaking the app into components also made a lot more sense after this lesson. Moving everything into a product-display component keeps product-related code organized and reusable. Each product component works independently, even though they all share the same structure. That made it clearer why components are such a big part of Vue apps.

Props helped me understand how data flows from a parent to a child component. Since components are isolated, they can't directly access parent data. Passing the `premium` value as a prop allows the product component to decide whether shipping is free or not without breaking that separation.

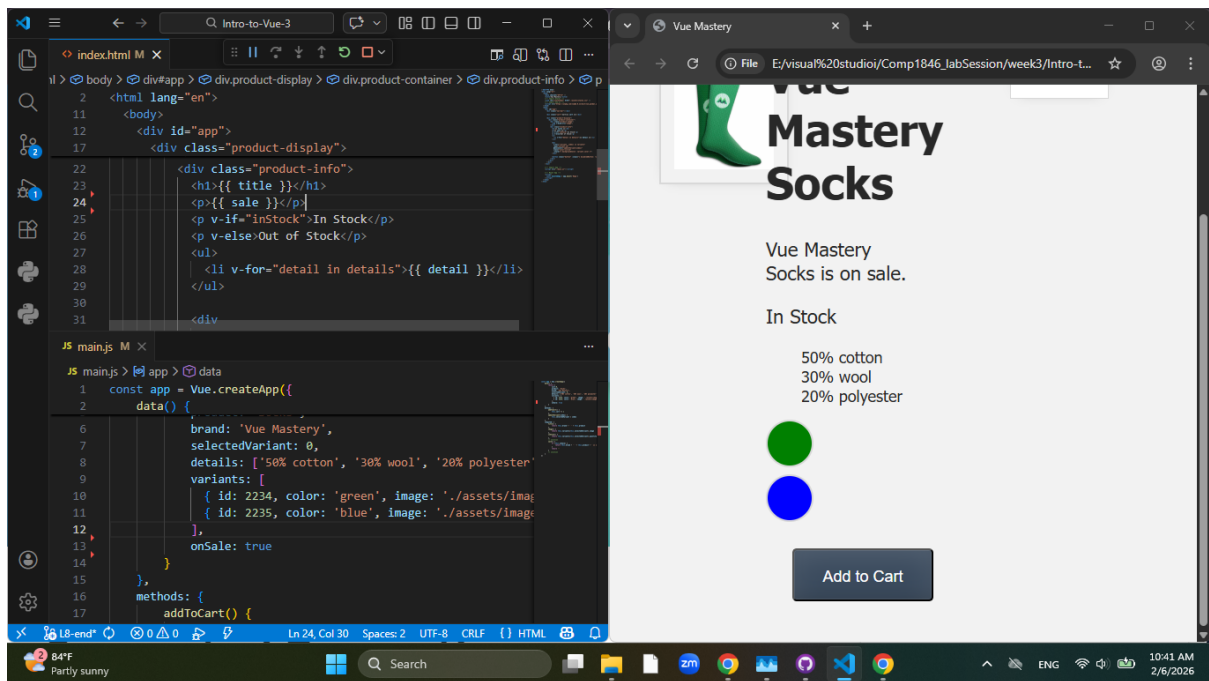
Emitting events finally tied everything together. Because the cart lives in the parent, the product component can't update it directly. Emitting an `add-to-cart` event lets the component notify the parent when the button is clicked. Passing the product ID with the event makes the cart more realistic.

Overall, these lessons showed me how Vue keeps data flow organized and predictable.

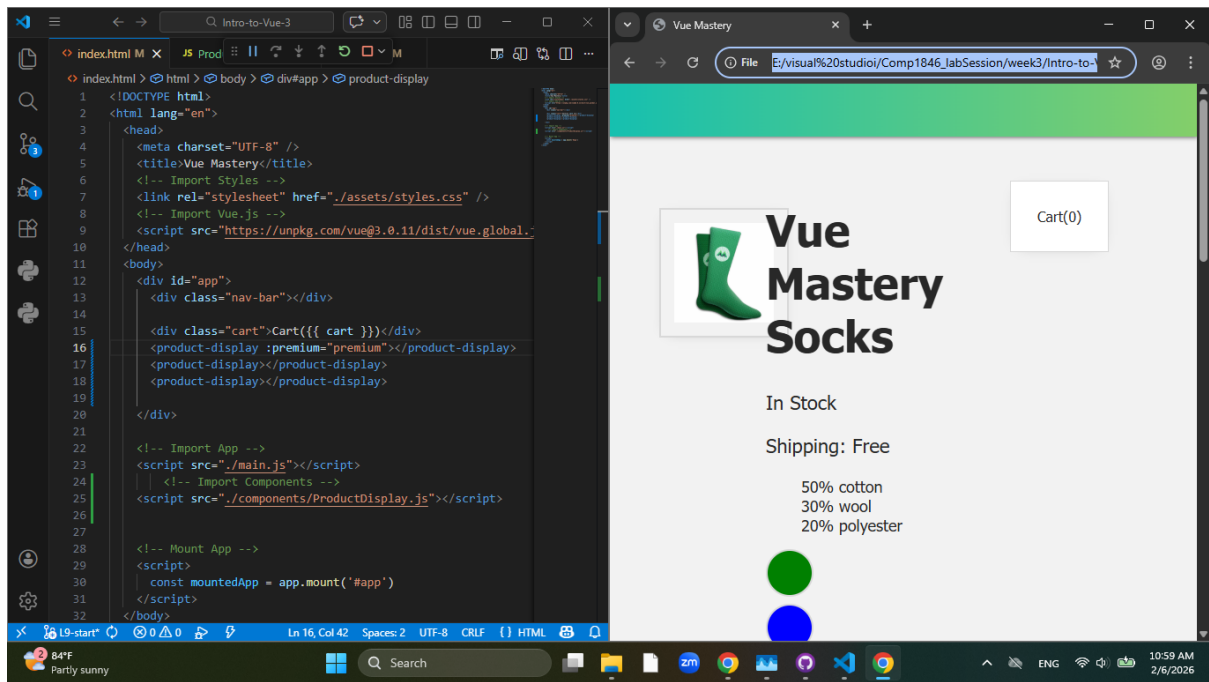
Task of lesson 8 picture:



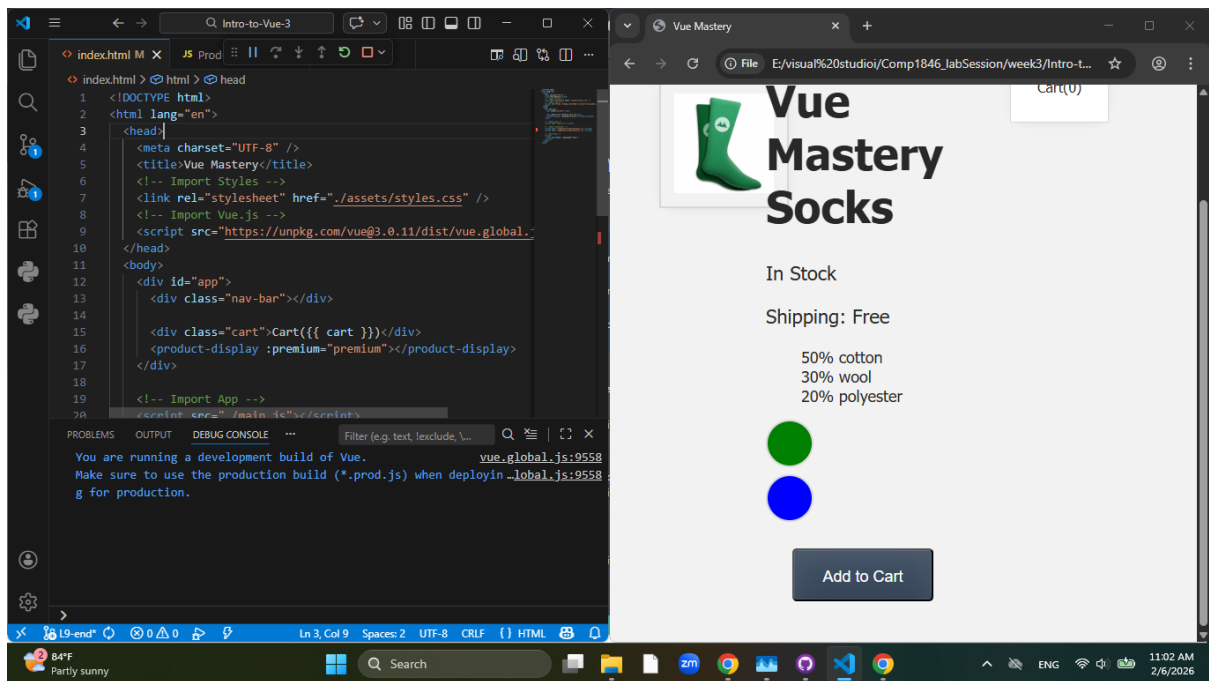
Challenge of lesson 8 picture:



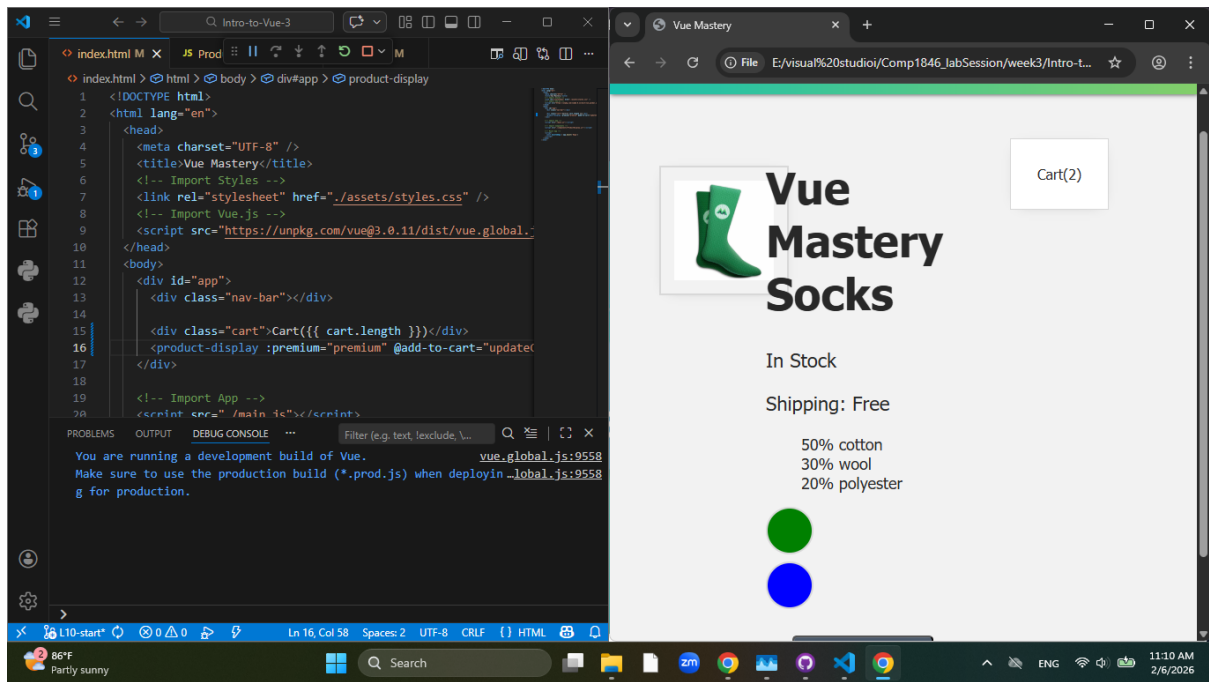
Task of lesson 9 picture:



Challenge of lesson 9 picture:



Task of lesson 10 picture:



Challenge of lesson 10 picture:

