

## Lesson 8 :

Cart (0)

# Vue MasterySocks

Out of Stock



50% cotton  
30% wool  
20% polyester



S  
M  
L  
XL

Add to Cart

28°C Nắng rải rác

File Edit Selection View Go Run Terminal Help

lesson 8

index.html

```

<!DOCTYPE html>
<html lang="en">
  <head>
    <meta charset="UTF-8" />
    <title>Vue Mastery</title>
    <!-- Import CSS -->
    <link rel="stylesheet" href="/assets/style.css" />
    <!-- Import JS -->
    <script src="https://unpkg.com/vue@3.0.11/dist/vue.global.js"></script>
  </head>
  <body>
    <div id="app">
      <div class="nav-bar"></div>
      <div class="cart">Cart {{ cart }}</div>
      <div class="product-display">
        <div class="product-container">
          <div class="product-image">
            
          </div>
          <div class="product-info">
            <h1>{{ brand }} {{ product }}</h1>
            <p v-if="inStock">In Stock</p>
            <p v-else>Out of Stock</p>
            <ul>
              <li v-for="detail in details" :key="detail">{{ detail }}</li>
            </ul>
            <div v-for="variant in variants" :key="variant.id" @mouseover="updateImage(variant.image)">
              <button class="color-circle"
                     :style="{ background-color: variant.color }">
                </button>
            </div>
            <ul>
              <li v-for="size in sizes" :key="size">{{ size }}</li>
            </ul>
            <div>
              <button class="disabled" :disabled="!inStock" @click="addToCart">Add to Cart</button>
            </div>
          </div>
        </div>
      </div>
    </div>
  </body>

```

main.js

```

const app = Vue.createApp({
  data() {
    return {
      cart: 0,
      brand: "Vue Mastery",
      product: "Socks",
      image: "assets/images/socks_blue.jpg",
      inStock: false,
      details: ["50% cotton", "30% wool", "20% polyester"],
      variants: [
        { id: 1, color: "green", image: "assets/images/socks_green.jpg" },
        { id: 2, color: "blue", image: "assets/images/socks_blue.jpg" }
      ],
      sizes: ["S", "M", "L", "XL"]
    };
  },
  methods: {
    addToCart() {
      this.cart += 1;
    },
    removeFromCart() {
      if (this.cart > 0) {
        this.cart -= 1;
      }
    },
    updateImage(variantImage) {
      this.image = variantImage;
    }
  }
});
app.mount("#app");

```

Nắng rải rác

File Edit Selection View Go Run Terminal Help

lesson 8

index.html

```

<!DOCTYPE html>
<html lang="en">
  <head>
    <meta charset="UTF-8" />
    <title>Vue Mastery</title>
    <!-- Import CSS -->
    <link rel="stylesheet" href="/assets/style.css" />
    <!-- Import JS -->
    <script src="https://unpkg.com/vue@3.0.11/dist/vue.global.js"></script>
  </head>
  <body>
    <div id="app">
      <div class="nav-bar"></div>
      <div class="cart">Cart {{ cart }}</div>
      <div class="product-display">
        <div class="product-container">
          <div class="product-image">
            
          </div>
          <div class="product-info">
            <h1>{{ brand }} {{ product }}</h1>
            <p v-if="inStock">In Stock</p>
            <p v-else>Out of Stock</p>
            <ul>
              <li v-for="detail in details" :key="detail">{{ detail }}</li>
            </ul>
            <div v-for="variant in variants" :key="variant.id" @mouseover="updateImage(variant.image)">
              <button class="color-circle"
                     :style="{ background-color: variant.color }">
                </button>
            </div>
            <ul>
              <li v-for="size in sizes" :key="size">{{ size }}</li>
            </ul>
            <div>
              <button class="disabled" :disabled="!inStock" @click="addToCart">Add to Cart</button>
            </div>
          </div>
        </div>
      </div>
    </div>
  </body>

```

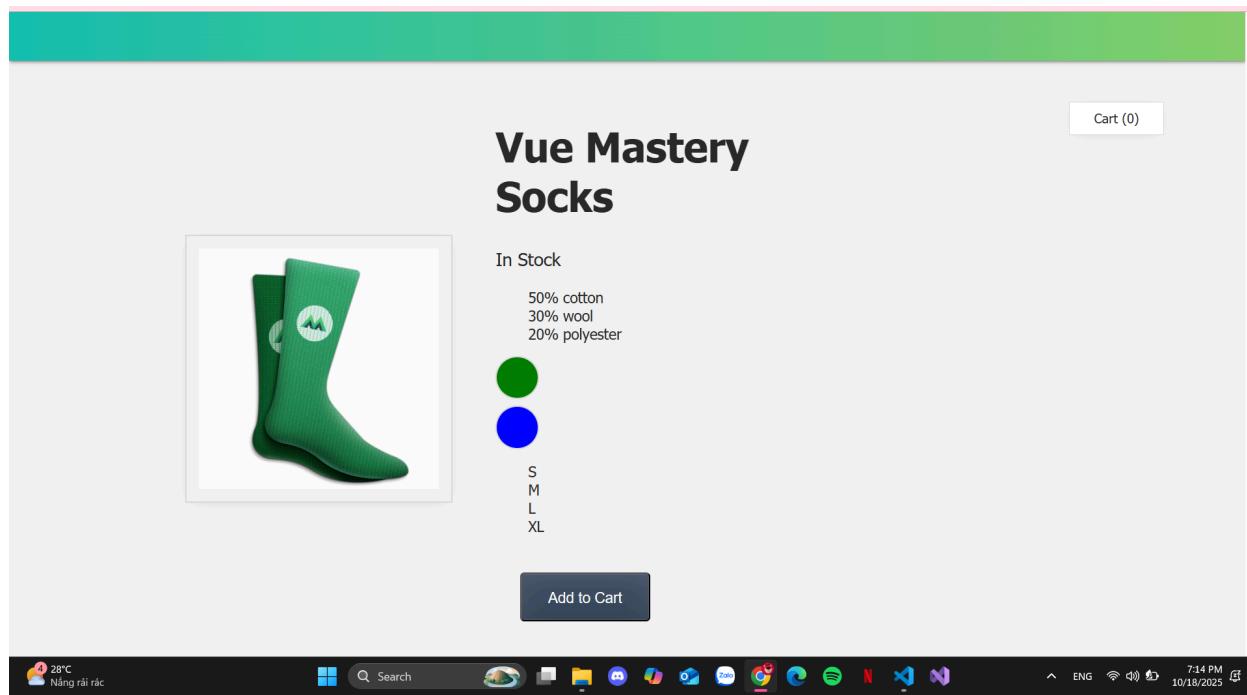
main.js

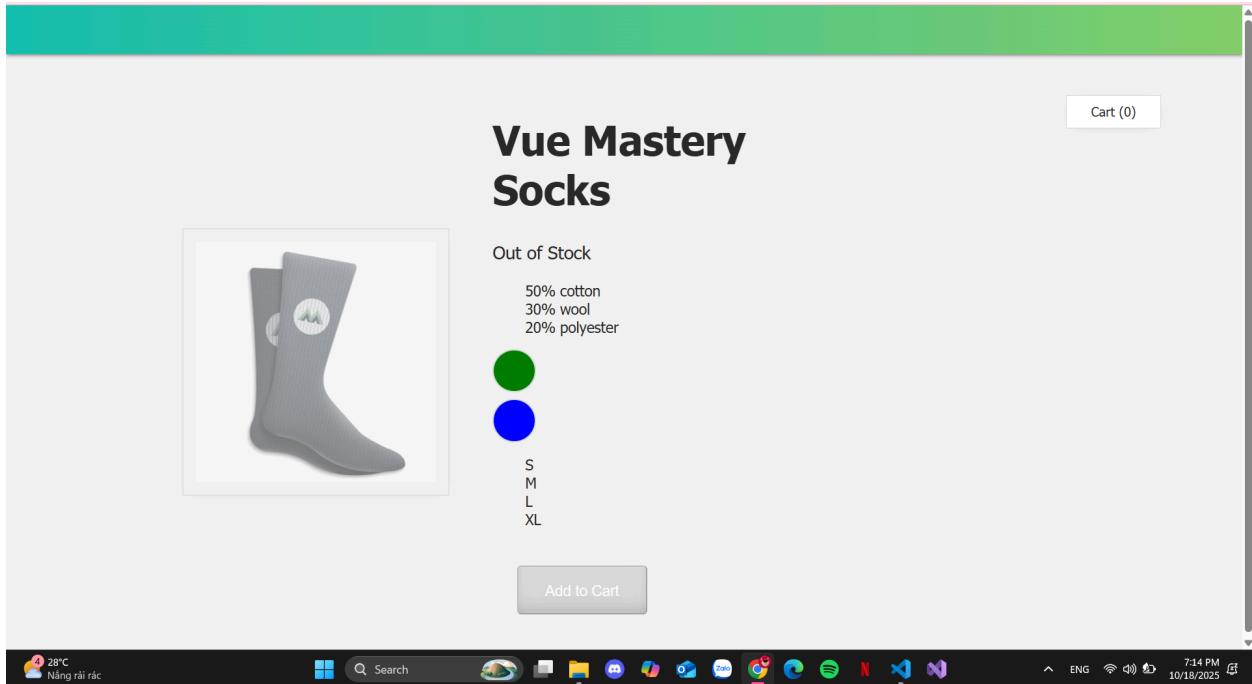
```

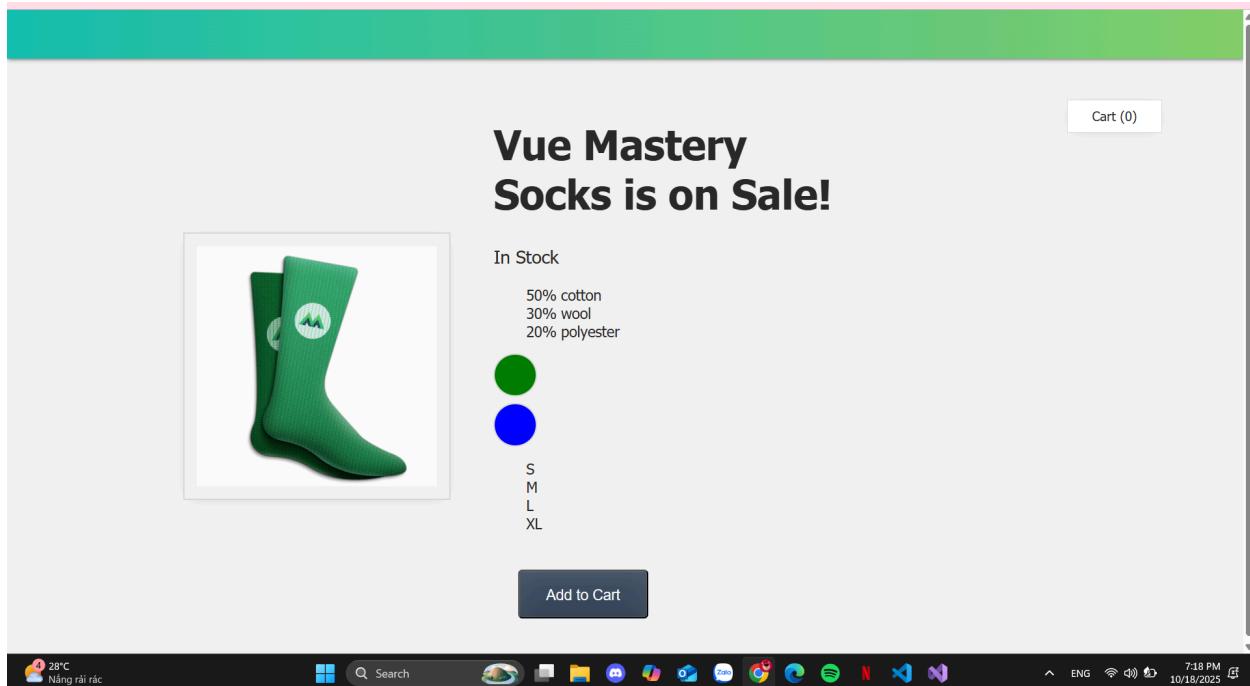
const app = Vue.createApp({
  data() {
    return {
      cart: 0,
      brand: "Vue Mastery",
      product: "Socks",
      image: "assets/images/socks_blue.jpg",
      inStock: false,
      details: ["50% cotton", "30% wool", "20% polyester"],
      variants: [
        { id: 1, color: "green", image: "assets/images/socks_green.jpg" },
        { id: 2, color: "blue", image: "assets/images/socks_blue.jpg" }
      ],
      sizes: ["S", "M", "L", "XL"]
    };
  },
  methods: {
    addToCart() {
      this.cart += 1;
    },
    removeFromCart() {
      if (this.cart > 0) {
        this.cart -= 1;
      }
    },
    updateImage(variantImage) {
      this.image = variantImage;
    }
  }
});
app.mount("#app");

```

Nắng rải rác







## Lesson 9 :

File Edit Selection View Go Run Terminal Help ← →

lesson 9

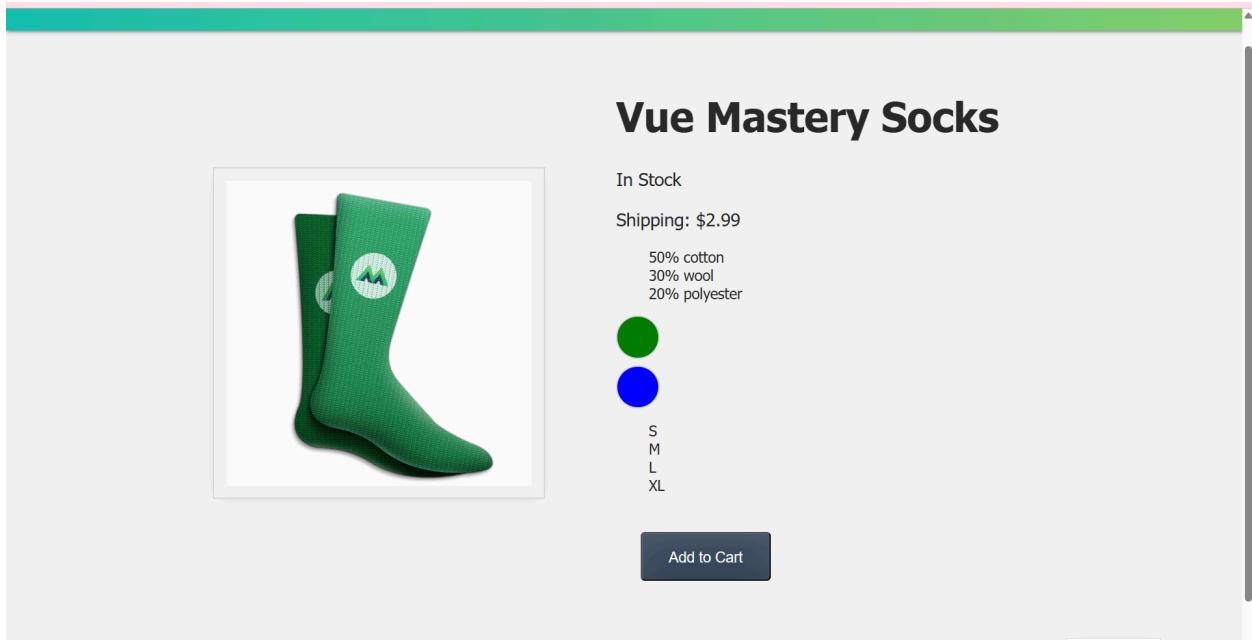
index.html

```
<!DOCTYPE html>
<html lang="en">
  <head>
    <meta charset="UTF-8" />
    <title>Vue Mastery</title>
    <!-- Import CSS -->
    <link rel="stylesheet" href="/assets/styles.css" />
    <!-- Import Vue.js -->
    <script src="https://unpkg.com/vue@0.11/dist/vue.global.js"></script>
  </head>
  <body>
    <div id="app">
      <div class="nav-bar"></div>
      <product-display :premium="premium"></product-display>
      <div class="cart">Cart {{ cart }}</div>
    </div>
    <script src="./components/ProductDisplay.js"></script>
    <script src="./main.js"></script>
  </body>
</html>
```

ProductDisplay.js

```
const ProductDisplay = {
  template: `
    <div>
      <button>Add to Cart</button>
    </div>
  </template>
  computed: {
    ...mapState(['cart']),
    ...mapGetters(['getVariant']),
    ...mapActions(['addCart']),
    ...mapMutations(['updateCart'])
  },
  data() {
    return {
      brand: "Vue Mastery",
      product: "Socks",
      selectedVariant: 0,
      details: ["50% cotton", "30% wool", "20% polyester"],
      variants: [
        { id: 2234, color: "green", image: "/assets/images/socks_green.jpg", quantity: 50 },
        { id: 2235, color: "blue", image: "/assets/images/socks_blue.jpg", quantity: 0 }
      ],
      sizes: ["S", "M", "L", "XL"]
    };
  },
  methods: {
    addCart() {
      this.$root.updateCart(1);
    },
    updateVariant(index) {
      this.selectedVariant = index;
    },
    computed: {
      title() {
        return this.brand + " " + this.product;
      },
      image() {
        return this.variants[this.selectedVariant].image;
      },
      inStock() {
        return this.variants[this.selectedVariant].quantity;
      },
      shipping() {
        if (this.premium) {
          return "free";
        }
      }
    }
};
```

index.html



The screenshot shows a Microsoft Edge browser window with the following details:

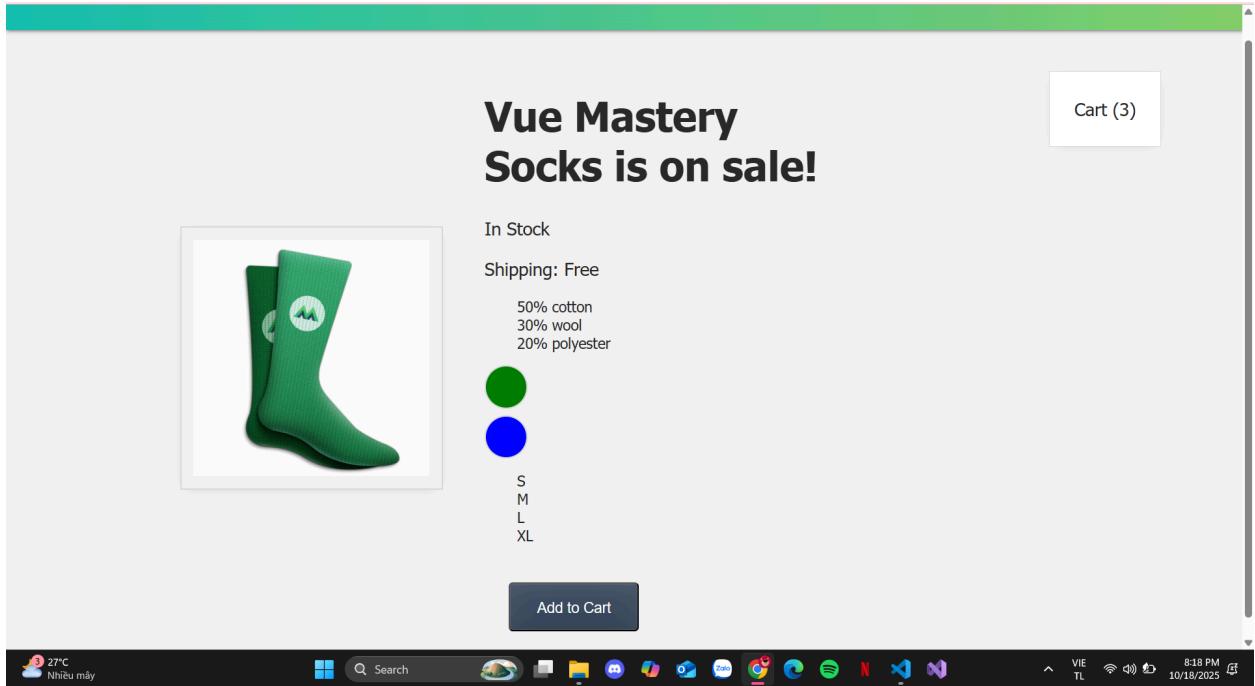
- Title Bar:** "Những ngày mưa..." and "27°C".
- Address Bar:** "Search" and "lesson 9".
- Left Sidebar (Explorer):** Shows project files including "index.html", "index.js", "index.css", "ProductDetails.js", "ProductDisplay.js", and "main.js".
- Right Sidebar (Components):** Shows components like "ProductDisplay" and "ProductDetails".
- Bottom Status Bar:** Includes icons for battery, signal, and network, along with "818 PM" and "10/18/2025".

The main content area displays two code snippets:

```
index.html
<!DOCTYPE html>
<html lang="en">
  <head>
    <meta charset="UTF-8" />
    <title>Product Details</title>
    <script src="https://unpkg.com/vue@3"></script>
    <link rel="stylesheet" href="/assets/styles.css" />
  </head>
  <body>
    <div id="app">
      <nav-bar>
        <cart>({{ cart }})</cart>
      </nav-bar>
      <product-display>
        <premium>
          @add-to-cart="updateCart"
        </premium>
      </product-display>
    </div>
    <!-- Component scripts -->
    <script src="/components/ProductDetails.js"></script>
    <script src="/components/ProductDisplay.js"></script>
    <!-- Main Vue app -->
    <script src="main.js"></script>
  </body>
</html>
```

```
main.js
const ProductDisplay = {
  template: `
    <div>
      <button
        class="button"
        class="disabled" @click="addToCart"
        :disabled="!inStock"
        @click="addToCart"
      >
        Add to Cart
      </button>
    </div>
  </div>
  `,
  data() {
    return {
      brand: "Hue Mastery",
      product: "Socks",
      onSale: true,
      selectedVariant: 0,
      details: ["90% cotton", "10% wool", "10% polyester"],
      variants: [
        { id: 2234, color: "green", image: "/assets/images/socks_green.jpg", quantity: 50 },
        { id: 2235, color: "blue", image: "/assets/images/socks_blue.jpg", quantity: 0 }
      ],
      sizes: ["S", "M", "L", "XL"]
    };
  },
  methods: {
    addToCart() {
      this.$emit("add-to-cart");
    },
    updateVariant(index) {
      this.selectedVariant = index;
    }
  },
  computed: {
    title() {
      return this.onSale
        ? `${this.brand} ${this.product} is on sale!`
        : `${this.brand} ${this.product}`;
    },
    image() {
      return this.variants(this.selectedVariant).image;
    },
    inStock() {
      return this.variants(this.selectedVariant).quantity > 0;
    },
    shipping() {
      return this.premium ? "Free" : "$2.99";
    }
  }
};

ProductDisplay.methods.addCart = () =>
```



## Lesson 10 :

```

index.html ...
main.js ...
ProductDetails.js ...
ProductDisplay.js ...
main.js ...

index.html > ...
<!DOCTYPE html>
<html lang="en">
  <head>
    <meta charset="UTF-8" />
    <title>Vue Product Details</title>
    <script src="https://unpkg.com/vue@3"></script>
    <link rel="stylesheet" href="/assets/styles.css" />
  </head>
  <body>
    <div id="app">
      <div class="nav-bar">
        <product-display
          :premium="premium"
          @add-to-cart="updateCart">
        </product-display>
      </div>
      <!-- Component scripts -->
      <script src="/components/ProductDetails.js"></script>
      <script src="/components/ProductDisplay.js"></script>
    </div>
    <!-- Main Vue app -->
    <script src="/main.js"></script>
  </body>
</html>

```

```

main.js > ...
1  const app = Vue.createApp({
2    data() {
3      return {
4        cart: 0,
5        premium: true
6      };
7    },
8    methods: {
9      updateCart() {
10        this.cart += 1;
11      }
12    }
13  });
14
15  app.component("product-details", ProductDetails);
16  app.component("product-display", ProductDisplay);
17
18  app.mount("#app");
19
20
21
22
23
24
25
26
27
28
29
30
31
32
33
34

```

The screenshot shows a Vue.js application running in a browser. The left side displays the `index.html` file, which includes the main Vue app component and its dependencies. The right side shows the rendered application with two components: `ProductDetails` and `ProductDisplay`.

```
index.html >
<!DOCTYPE html>
<html lang="en">
  <head>
    <meta charset="UTF-8" />
    <title>Vue Product Details</title>
    <script src="https://unpkg.com/vue@3"></script>
    <link rel="stylesheet" href="/assets/styles.css" />
  </head>
  <body>
    <div id="app">
      <div class="nav-bar"></div>
      <div>
        <div class="cart">
          <p>Cart {{ cart }}</p>
        </div>
        <div>
          <product-display>
            <premium>
              <#add-to-cart="updateCart">
            </premium>
          </product-display>
        </div>
        <!-- Component scripts -->
        <script src="./components/ProductDetails.js"></script>
        <script src="./components/ProductDisplay.js"></script>
      </div>
      <!-- Main Vue app -->
      <script src="/main.js"></script>
    </div>
  </body>
</html>
```

```
ProductDetails > ProductDisplay.js ● | mang
components > ProductDisplay.js > @ ProductDisplay > template
  1  const ProductDisplay = (
  2    data() {
  3      return {
  4        brand: "Vue Mastery",
  5        product: "Socks",
  6        onSale: true,
  7        selectedVariant: 0,
  8        details: ["50% cotton", "30% wool", "20% polyester"],
  9        variants: [
 10          { id: 2234, color: "green", image: "./assets/images/socks_green.jpg", quantity: 50 },
 11          { id: 2235, color: "blue", image: "./assets/images/socks_blue.jpg", quantity: 0 }
 12        ],
 13        sizes: ["S", "M", "L", "XL"]
 14      };
 15    }
 16    methods: {
 17      addToCart() {
 18        this.$emit("add-to-cart");
 19      },
 20      updateVariant(index) {
 21        this.selectedVariant = index;
 22      }
 23    },
 24    computed: {
 25      title() {
 26        return this.onSale
 27          ? `${this.brand} ${this.product} is on sale!`
 28          : `${this.brand} ${this.product}`;
 29      },
 30      image() {
 31        return this.variants[this.selectedVariant].image;
 32      },
 33      instock() {
 34        return this.variants[this.selectedVariant].quantity > 0;
 35      },
 36      shipping() {
 37        return this.premium ? "Free" : "$2.99";
 38      }
 39    }
 40  
```

**Vue Mastery**  
**Socks is on sale!**

In Stock

Shipping: Free

50% cotton  
30% wool  
20% polyester

S  
M  
L  
XL

Add to Cart

# Vue Mastery

## Socks is on sale!

In Stock



Shipping: Free

50% cotton  
30% wool  
20% polyester



S  
M  
L  
XL

Add to Cart

Cart (4): [2234, 2234, 2234, 2234]

27°C Nhiều mây

Search

VIE TL 8:36 PM 10/18/2025

File Edit Selection View Go Run Terminal Help ← →

lesson 10

EXPLORER

index.html

```
<!DOCTYPE html>
<html lang="en">
  <head>
    <meta charset="UTF-8" />
    <title>Vue Product Details</title>
    <script src="https://unpkg.com/vue@3"></script>
    <link rel="stylesheet" href="/assets/styles.css" />
  </head>
  <body>
    <div id="app">
      <div class="nav-bar"></div>
      <div class="product">
        <h1>Product Details</h1>
        <p>Cart ({{ cart.length }}): {{ cart.join(', ') }}</p>
        <div>
          <product-display
            :premium="premium"
            @add-to-cart="updateCart"
          ></product-display>
        </div>
        <!-- Component scripts -->
        <script src="/components/ProductDetails.js"></script>
        <script src="/components/ProductDisplay.js"></script>
      </div>
      <!-- Main Vue app -->
      <script src="/main.js"></script>
    </div>
  </body>
</html>
```

ProductDetails.js

```
const app = Vue.createApp({
  data() {
    return {
      cart: [],
      premium: true
    };
  },
  methods: {
    updateCart(id) {
      this.cart.push(id);
    }
  }
});

app.component('product-details', ProductDetails);
app.component('product-display', ProductDisplay);
app.mount('#app');
```

main.js

ProductDisplay.js

VS Code

27°C Nhiều mây

Search

VIE TL 8:36 PM 10/18/2025

**Vue Mastery**  
**Socks is on sale!**

In Stock



Shipping: Free

50% cotton  
30% wool  
20% polyester



**Add to Cart**      **Remove from Cart**

Cart (0): []

File Edit Selection View Go Run Terminal Help ↻ → lesson 10

EXPLORER    JS index.html    JS main.js

```

<!-- index.html -->
<html lang="en">
  <head>
    <meta charset="UTF-8" />
    <title>Vue Product Details</title>
    <script src="https://unpkg.com/vue@3.0.0/dist/vue.global.prod.js"></script>
    <link rel="stylesheet" href="/assets/styles.css" />
  </head>
  <body>
    <div id="app">
      <div class="nav-bar"></div>
      <div class="cart">
        <p>Cart: {{ cart.length }}: {{{ cart.join(',') }}}</p>
      </div>
      <product-display
        :premium="premium"
        @add-to-cart="updateCart"
        @remove-from-cart="removeFromCart">
      </product-display>
    </div>
    <!-- Component scripts -->
    <script src="/components/ProductDetails.js"></script>
    <script src="/components/ProductDisplay.js"></script>
    <!-- Main Vue app -->
    <script src="/main.js"></script>
  </body>
</html>

```

JS main.js

```

const app = Vue.createApp({
  data() {
    return {
      cart: [],
      premium: true
    }
  },
  methods: {
    updateCart(id) {
      this.cart.push(id);
    },
    removeFromCart(id) {
      const index = this.cart.indexOf(id);
      if (index !== -1) {
        this.cart.splice(index, 1);
      }
    }
  }
});
app.component('product-details', ProductDetails);
app.component('product-display', ProductDisplay);
app.mount('#app');

```

Search      27°C Nhiêu mây

VIE TL 8:40 PM 10/18/2025

```

<ProductDetails>
  <ProductDisplay :product="product" :cart="cart" :onSale="onSale" :image="image" :inStock="inStock" :shipping="shipping" :cartIsEmpty="cartIsEmpty" />
</ProductDetails>

```

```

const ProductDisplay = () => {
  template: `
    <div>
      <div>
        <div>
          <div></div>
          <div>
            <strong>${product.title}</strong>
            ${product.onSale ? ` ${product.onSale ? "On Sale" : ""}` : ""}
            ${product.inStock ? "In Stock" : "Out of Stock"}
            ${product.shipping ? "Free Shipping" : "Standard Shipping $2.99"}
            ${product.cartIsEmpty ? "Add to Cart" : "Remove from Cart"} ${product.variantId}
          </div>
        </div>
      </div>
    </div>
  `,
  props: {
    product: {
      type: Object,
      required: true
    },
    cart: {
      type: Array,
      required: true
    }
  },
  computed: {
    onSale() {
      return this.product.onSale ? ` ${this.product.onSale ? "On Sale" : ""}` : ""
    },
    image() {
      return this.product.variantId ? this.product.variants[this.product.variantId].image : ""
    },
    inStock() {
      return this.product.variantId ? this.product.variants[this.product.variantId].inStock : false
    },
    shipping() {
      return this.product.variantId ? ` ${this.product.variants[this.product.variantId].shipping}` : ""
    },
    cartIsEmpty() {
      return this.cart.length === 0
    }
  },
  methods: {
    addCart() {
      this.$emit("add-to-cart", this.variantId)
    },
    removeFromCart() {
      this.$emit("remove-from-cart", this.variantId)
    },
    updateVariant(index) {
      this.selectedVariant = index
    }
  }
}

```

In lessons 8, 9 and 10, I felt like I had a better understanding of how Vue components work and how parts of the application can communicate with each other. In lesson 8, I learned how to pass data between components using props and emit, which was confusing at first but with practice, I gradually got the hang of parent-child interactions. In lesson 9, I learned more about computed properties and methods, which made the code cleaner and easier to maintain. At that time, I really felt that Vue was quite “smart” because it could update the interface automatically when data changed.

In lesson 10, I learned how to separate component files like `ProductDisplay.js` and `ProductDetails.js`, which makes the project cleaner and more professional. Although I still encountered small errors when connecting files, after a few fixes, I understood the problem lies in the declaration and import. Overall, these three lessons helped me feel more confident in building basic Vue applications and I feel more interested in programming interactive interfaces.