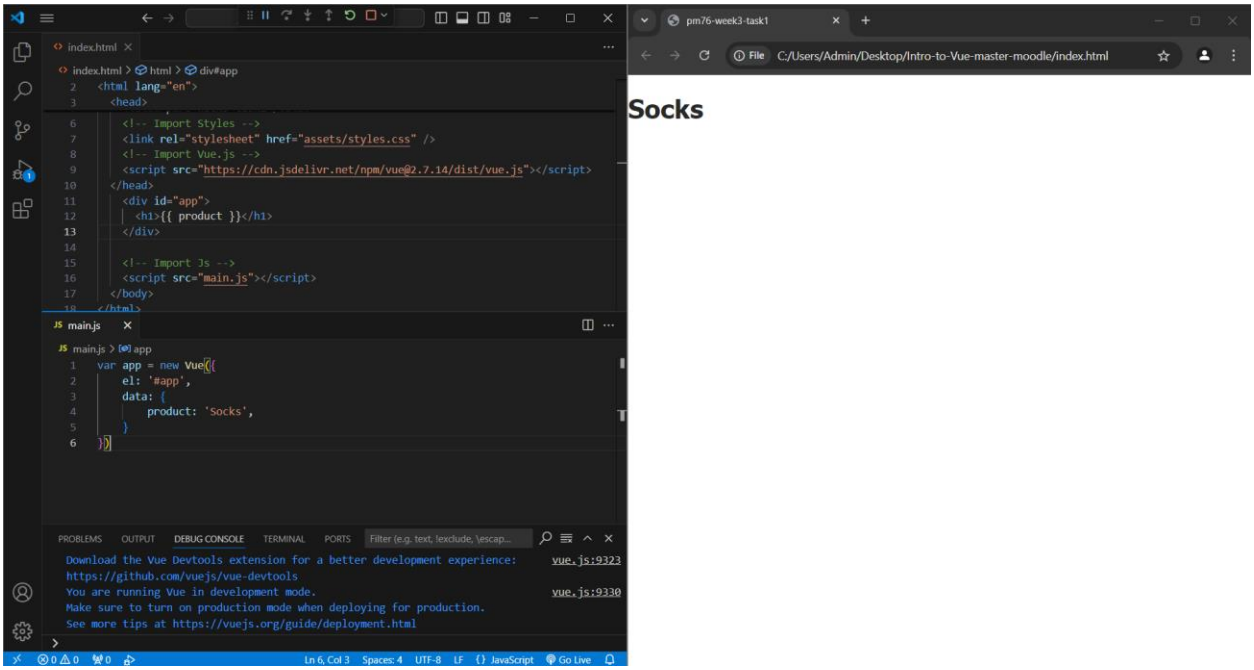
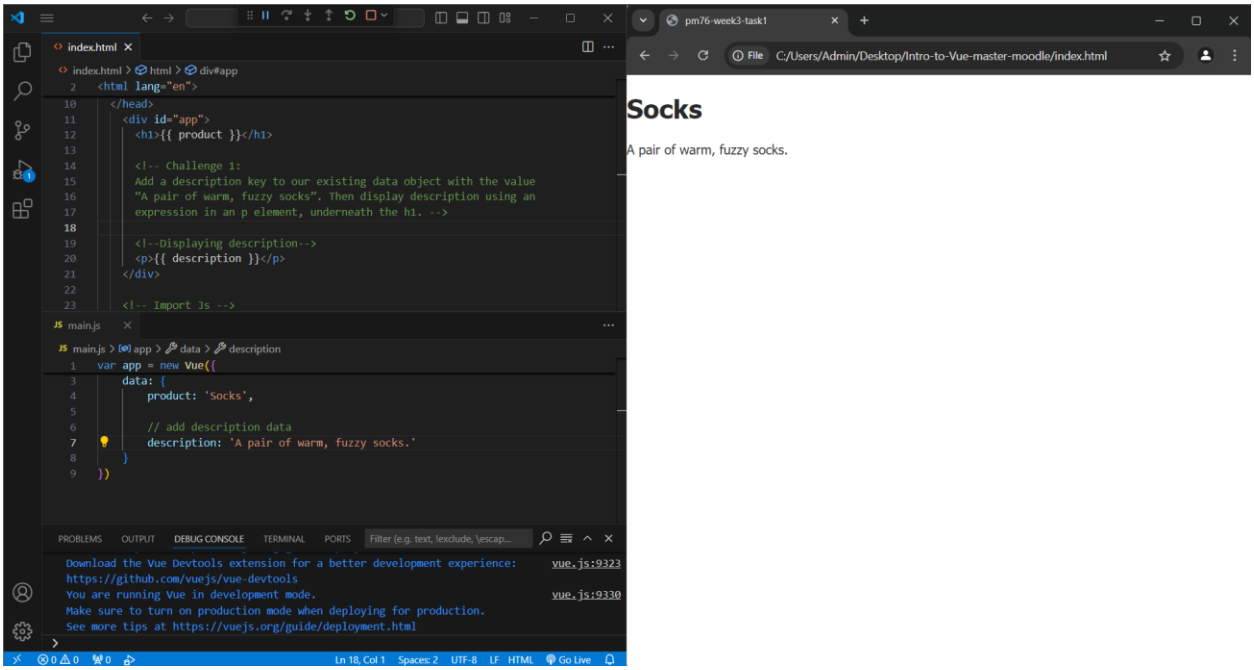
Full Name / Student ID:	Dao Vinh Long / 001322975
Submission Title:	COMP1842 Task 1 & 2
Course Name:	Web programing 2
Course Code:	COMP1842

Task A (Lesson 1-4):

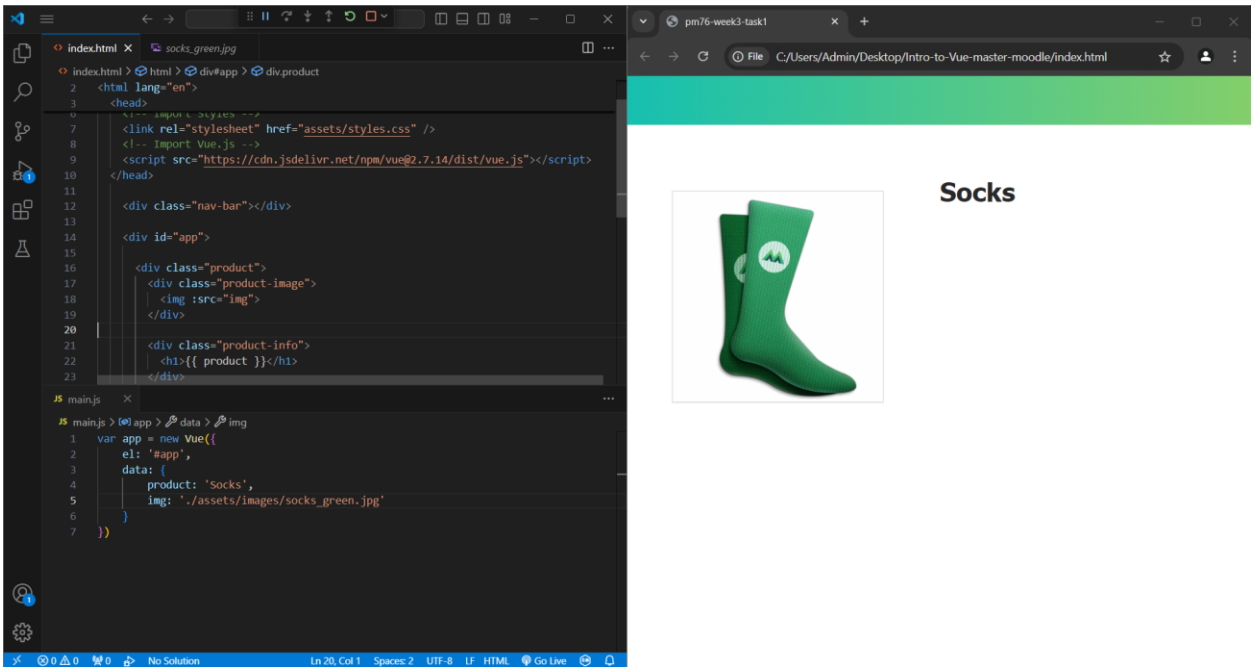
Lesson 1: The Vue Instance



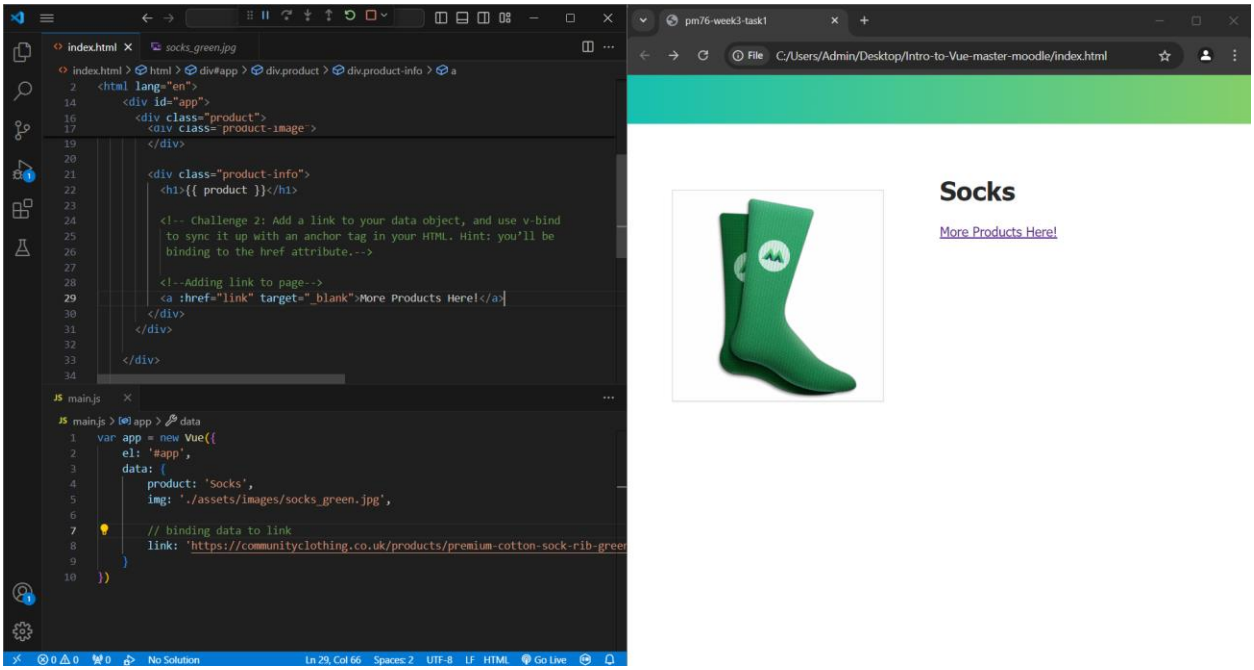
Challenge 1:



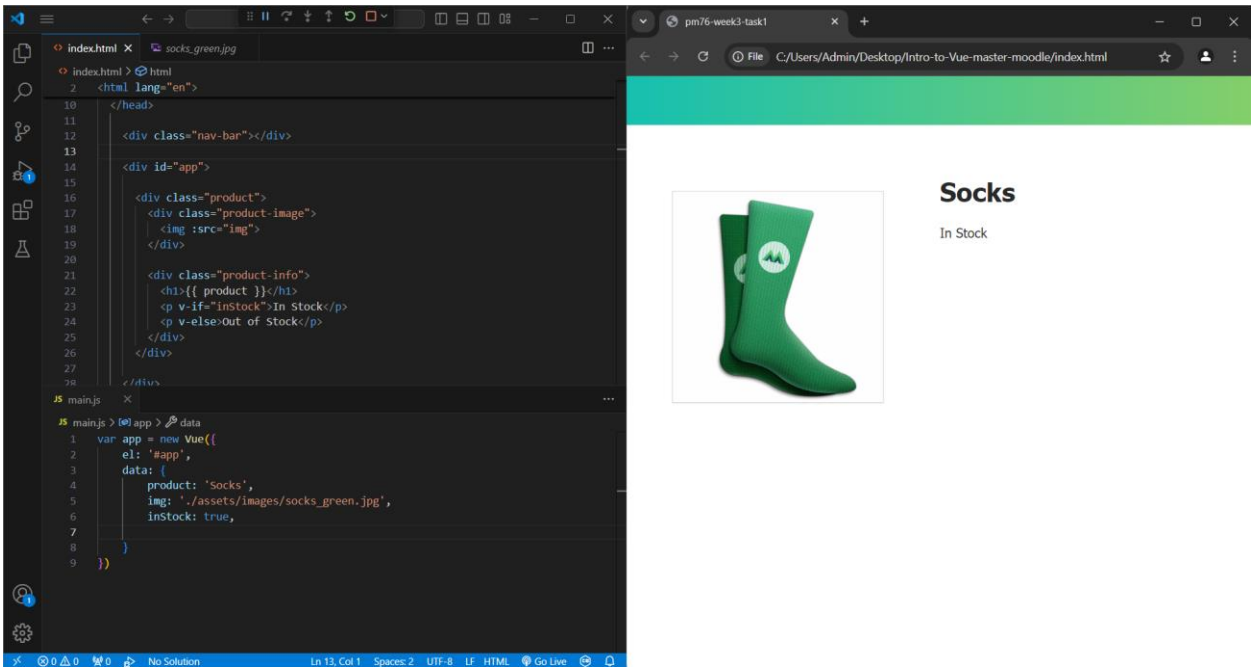
Lesson 2: Attribute Binding



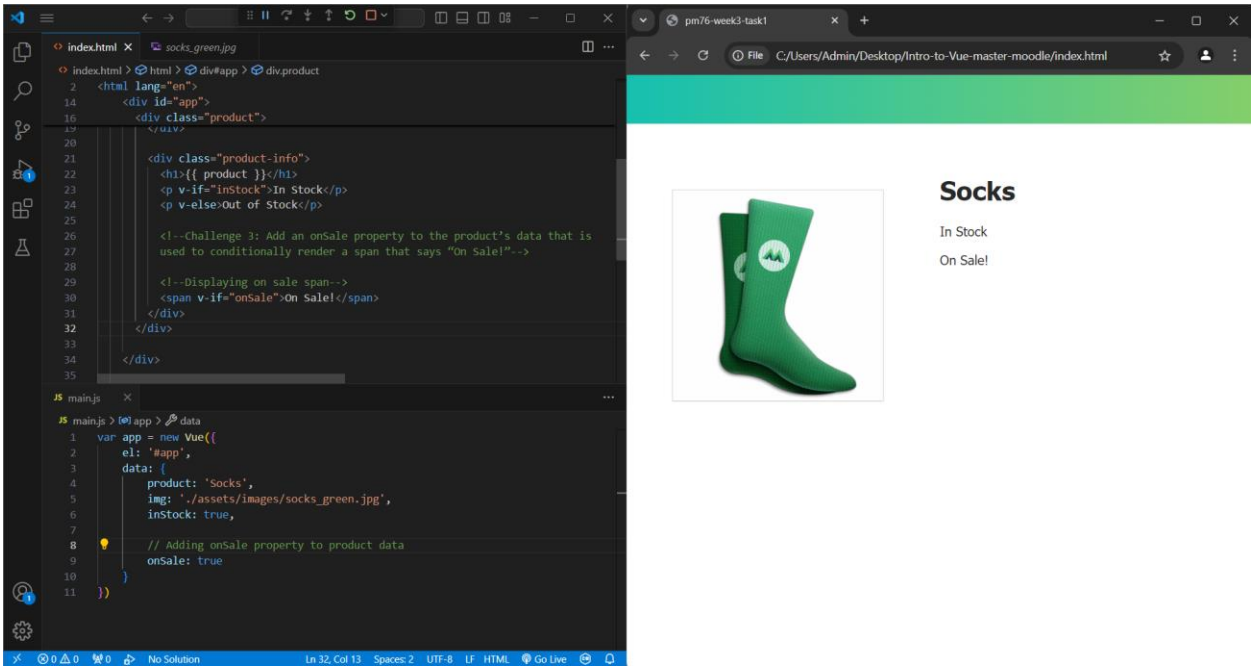
Challenge 2:



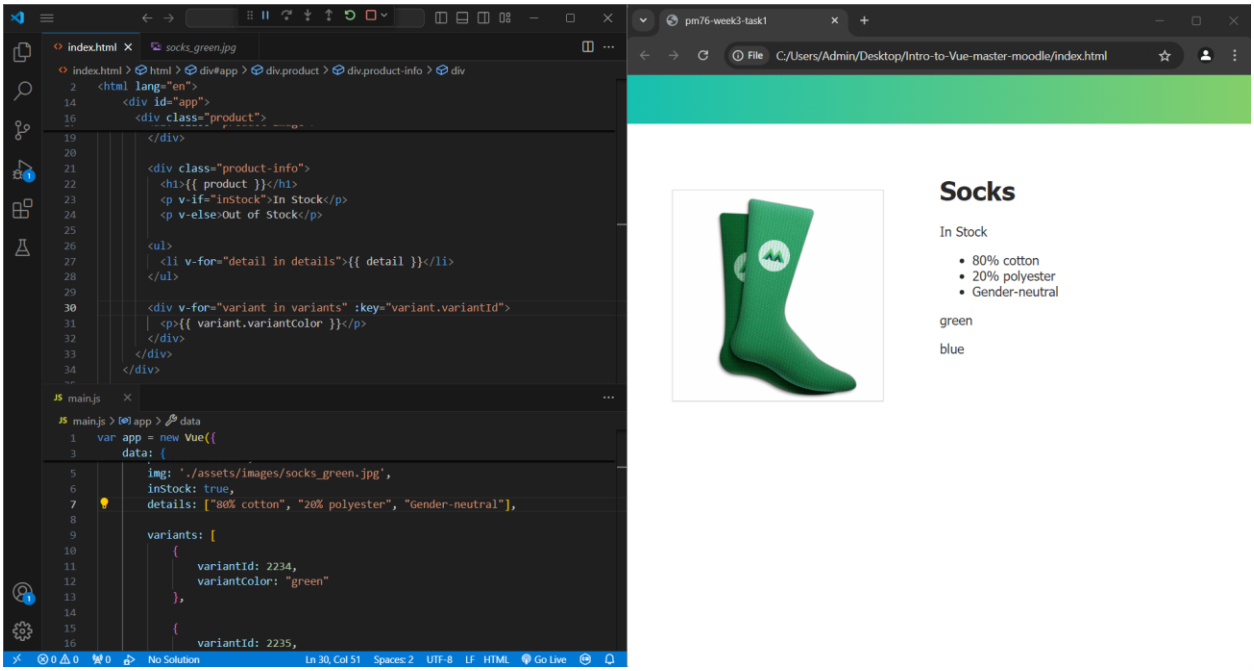
Lesson 3: Conditional Rendering



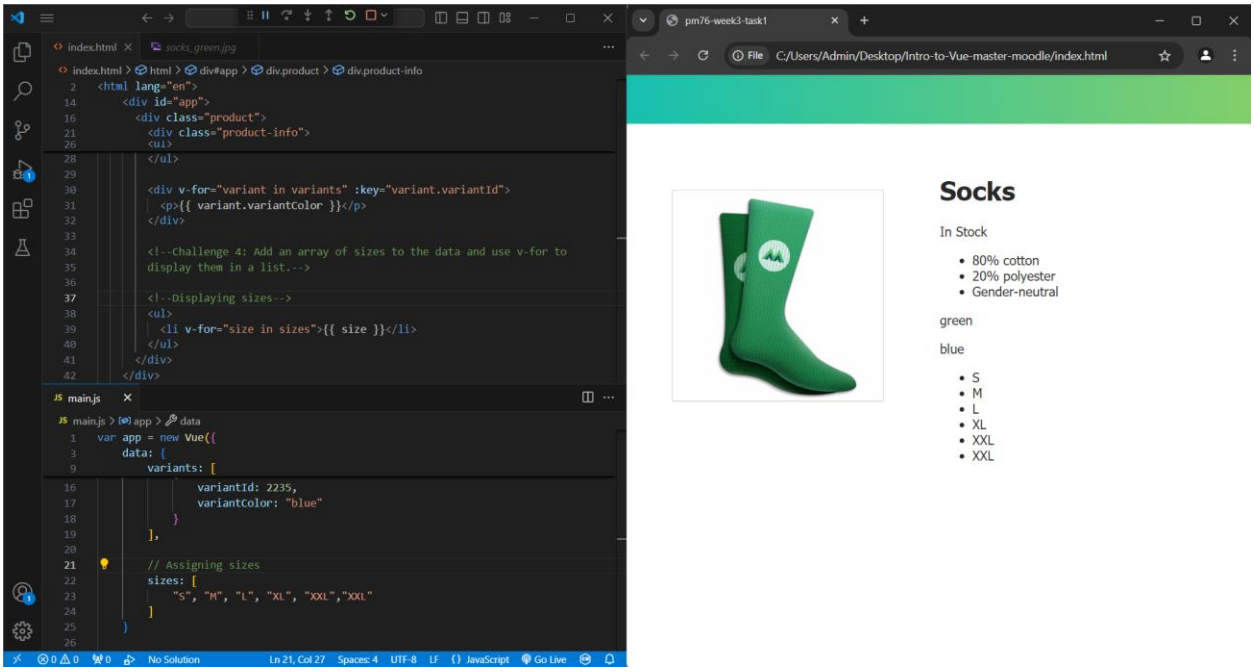
Challenge 3



Lesson 4: List Rendering



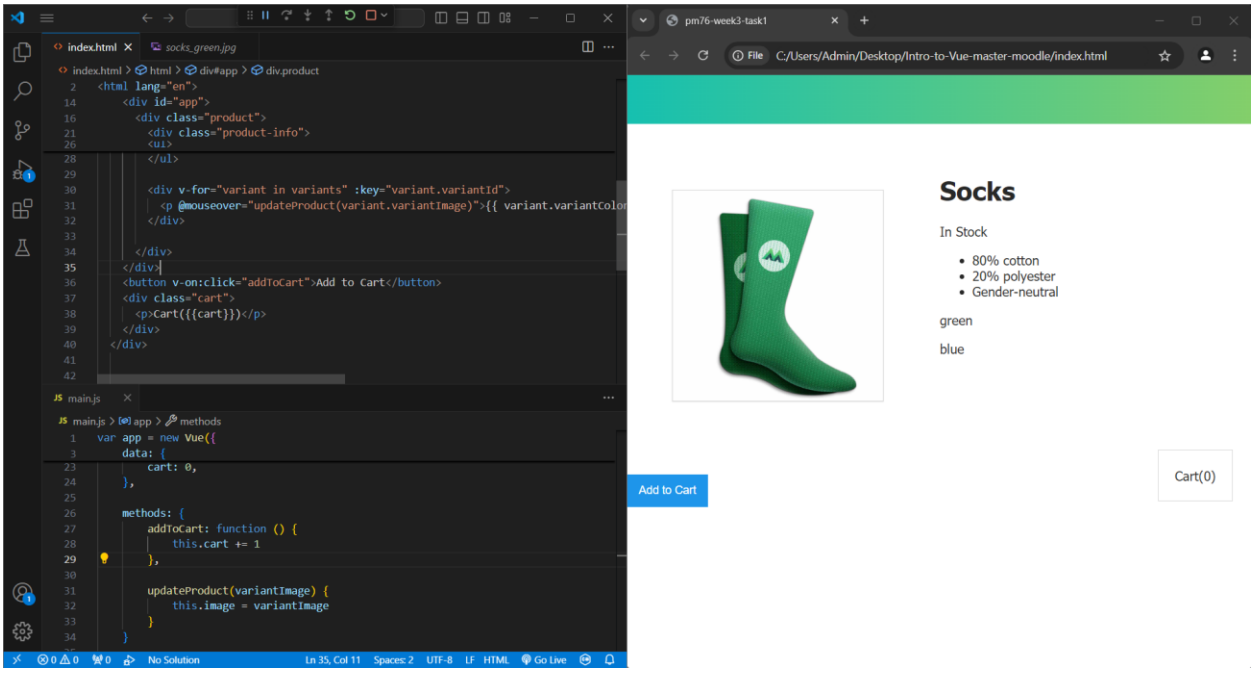
Challenge 4:



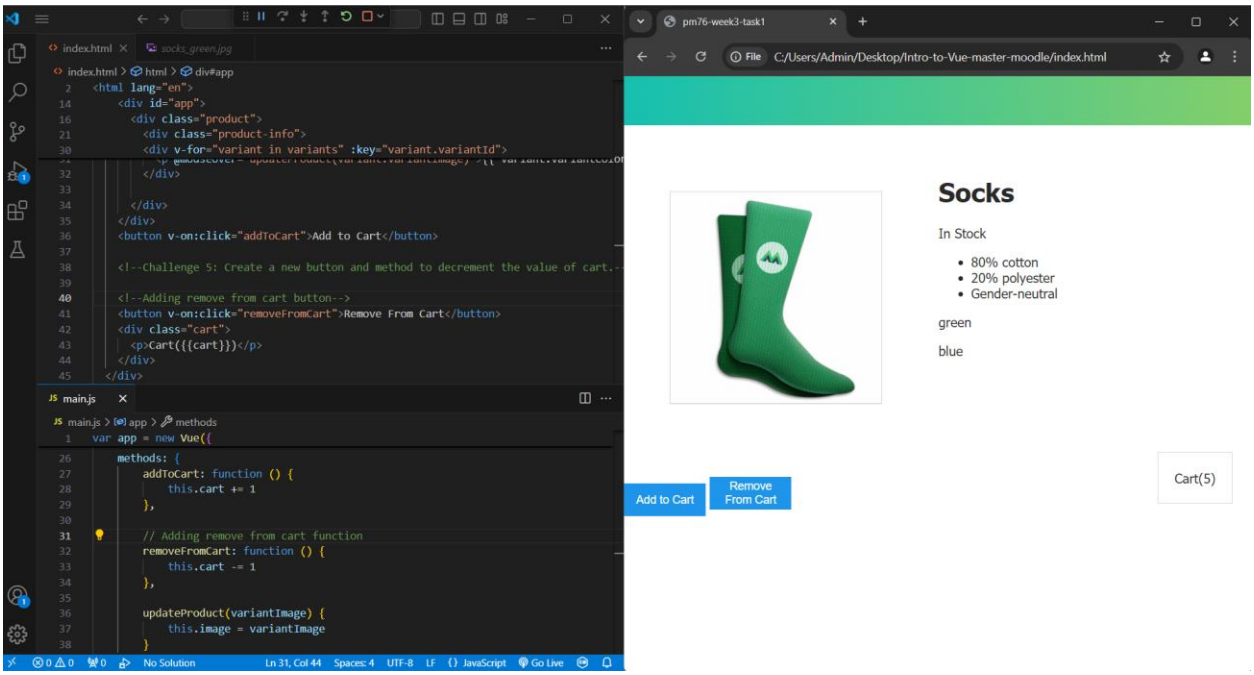
Task 1 Evaluation: From task 1 (lesson 1-4), I have understood how to create and manage a Vue instance. Bind HTML attributes (such as href, src, etc.) to the data properties in Vue instance using the v-bind directive. This makes the app more dynamic and responsive to changes in data. v-if, v-else, and v-show are used to control conditional elements. rendering lists of items using the v-for allow to dynamically display arrays or objects and manage list data effectively.

Task 2: Lesson 5-7

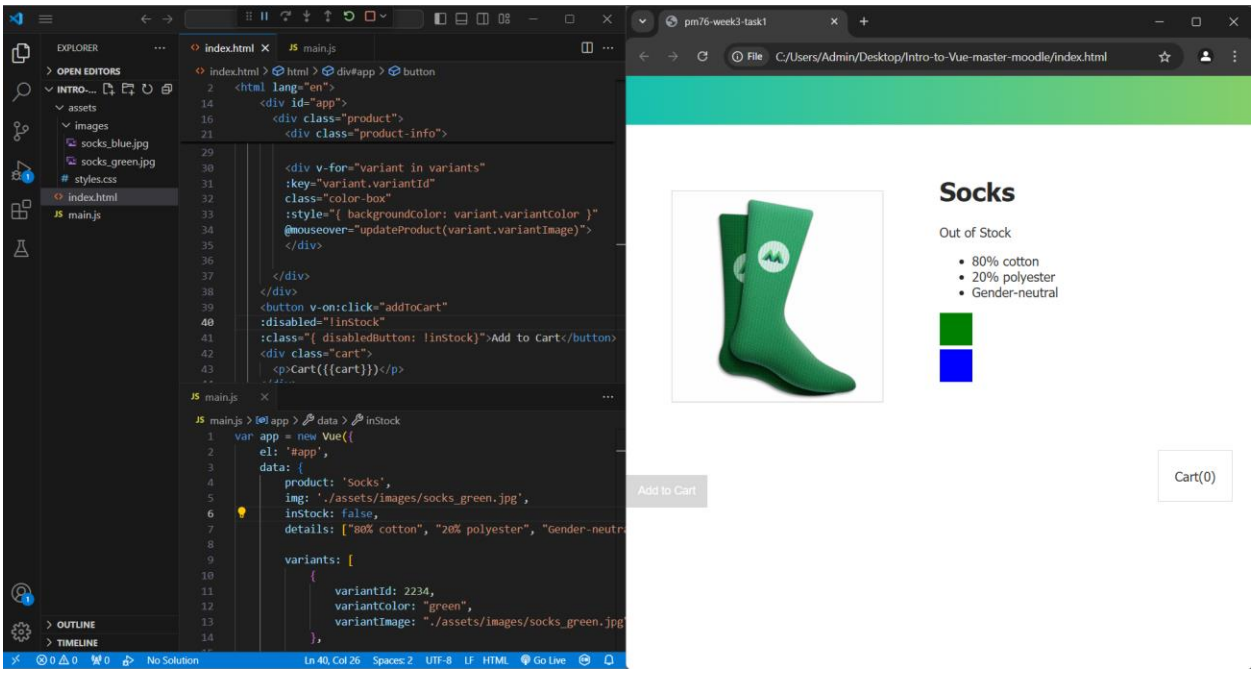
Lesson 5: Event Handling



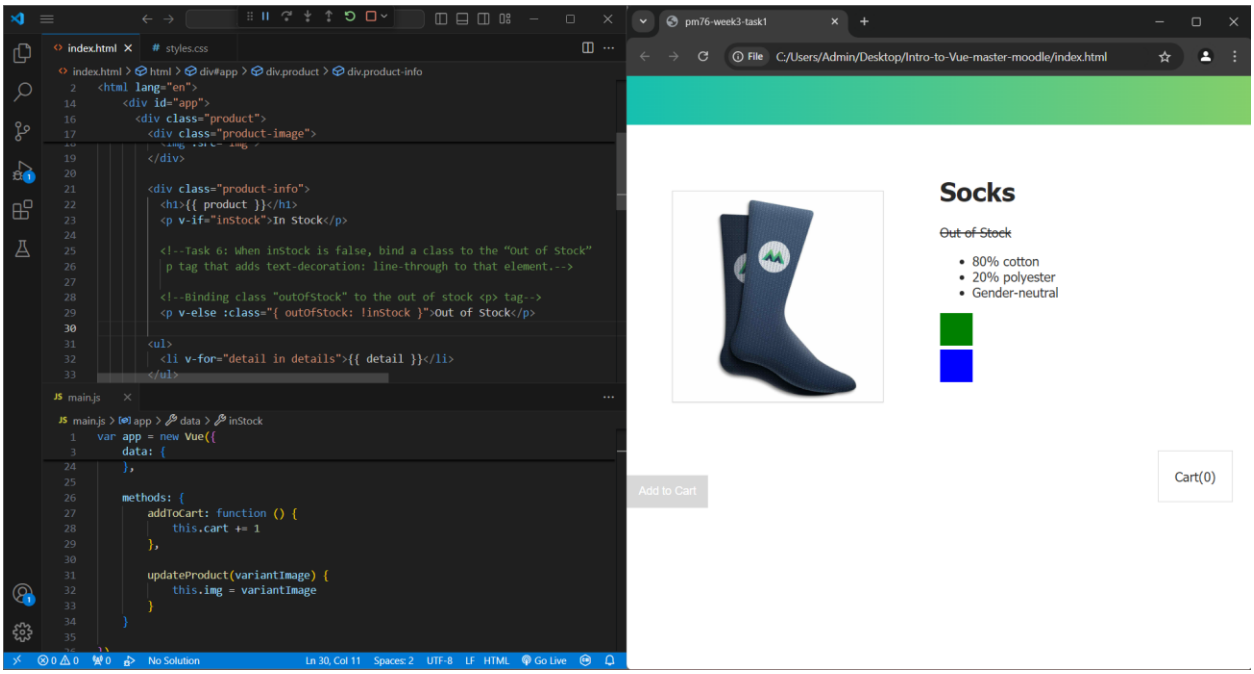
Challenge 5:



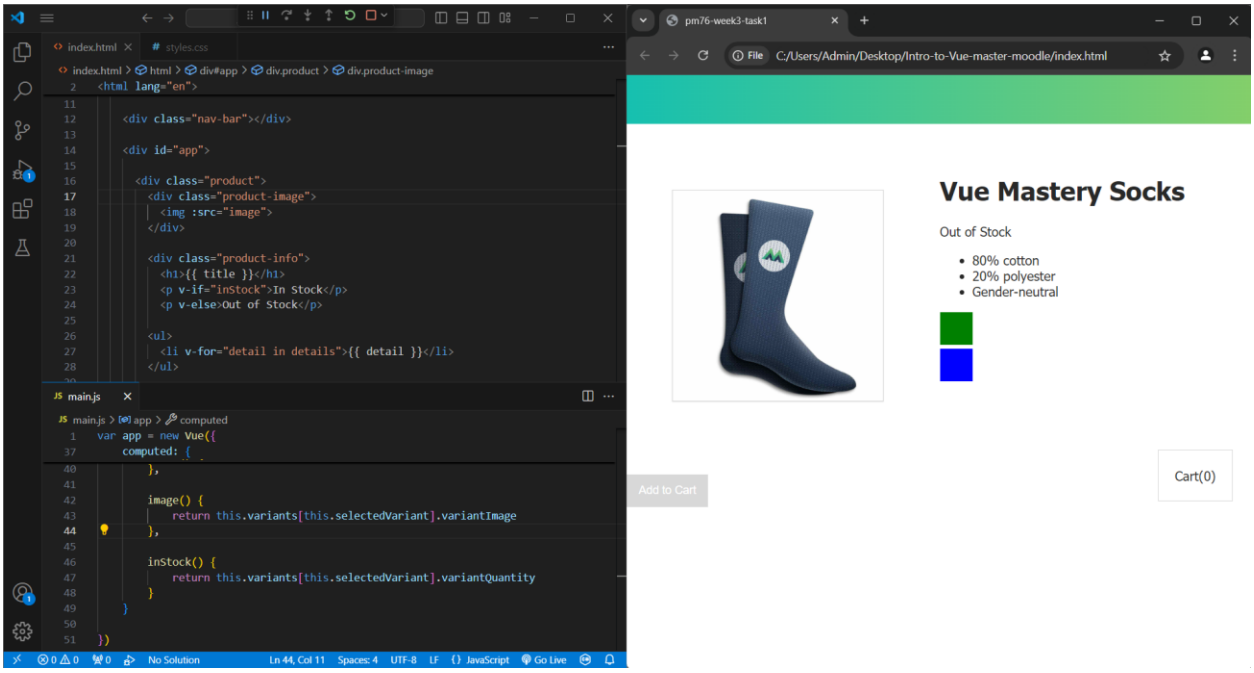
Lesson 6: Class & Style Binding



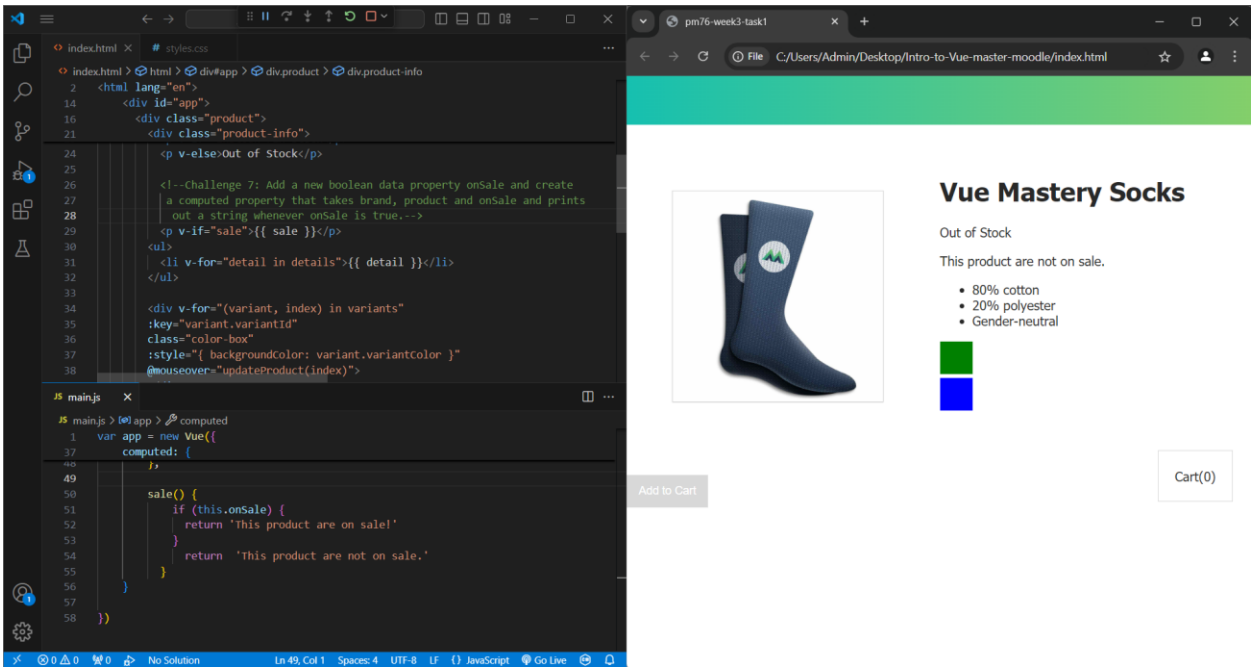
Challenge 6:



Lesson 7: Computed Properties

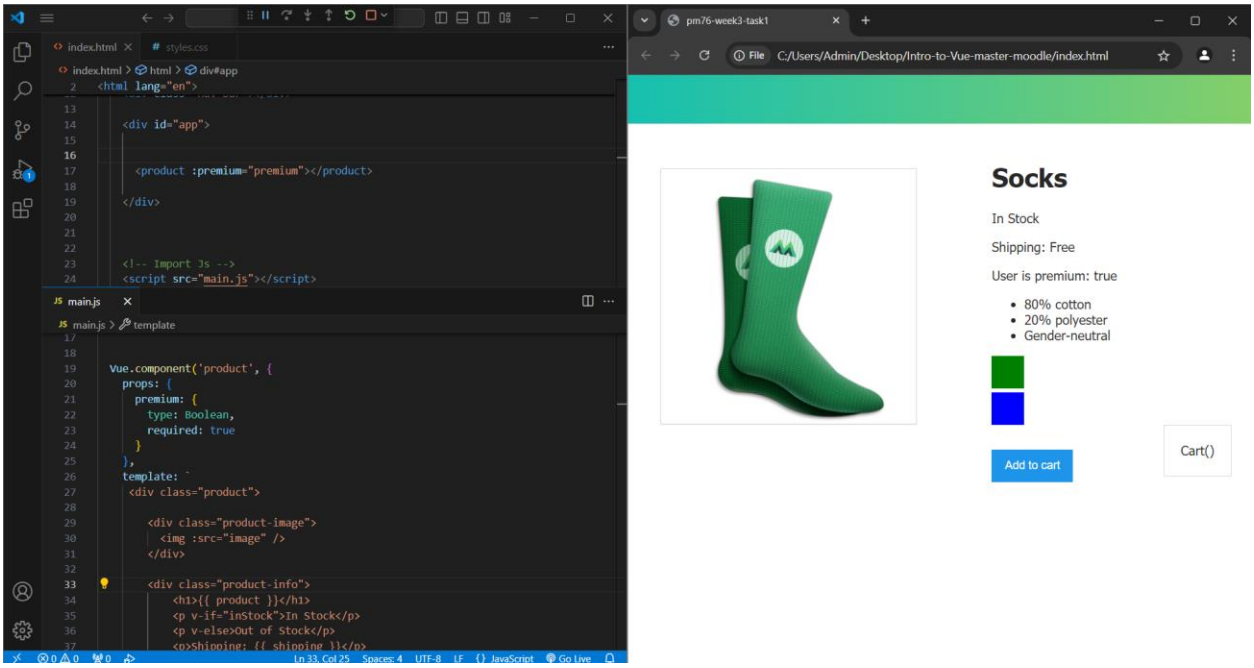


Challenge 7:



Task 2 Evaluation: After doing task 2 (Lesson 5-7) Lessons and Challenges can help me gains an understanding of how to manage user interactions like clicks, keypresses using Vue's event binding (v-on or @). And bind CSS classes and inline style. Finally, use Vue's computed properties to automatically updated when dependent data changes.

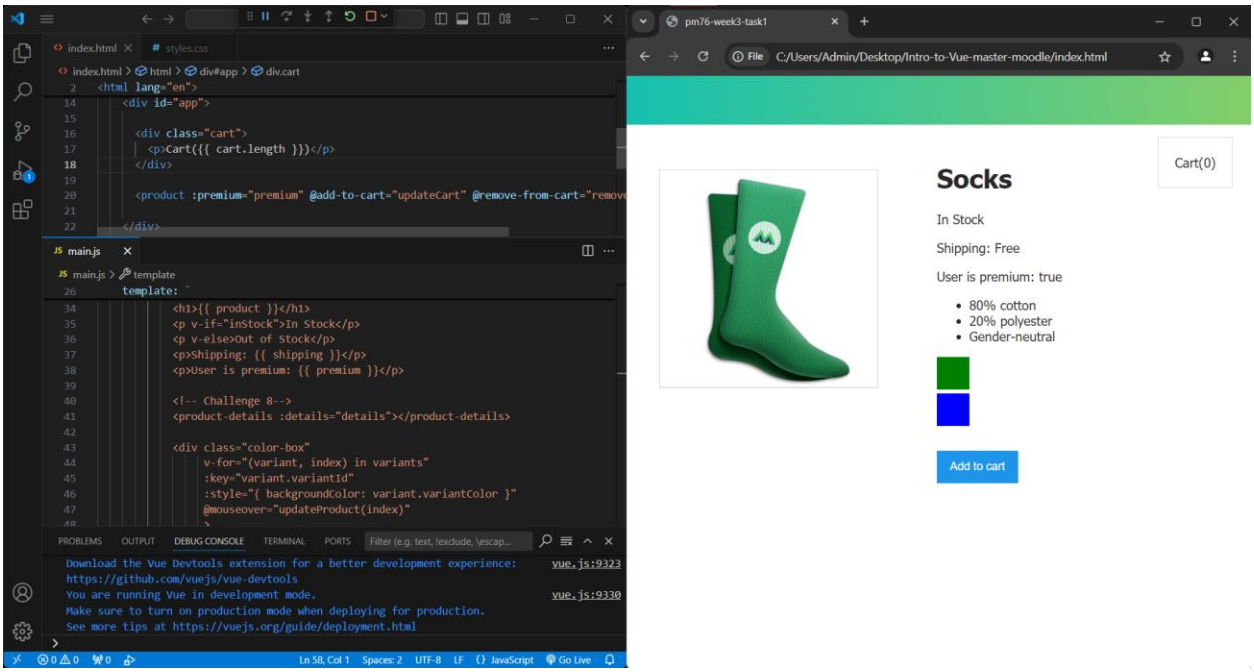
Task


index.html
styles.css
<html lang="en">

<product :premium="premium"></product>

<!-- Import Js -->
<script src="main.js"></script>
main.js
Vue.component('product', {
props: {
premium: {
type: Boolean,
required: true
}
},
template: `

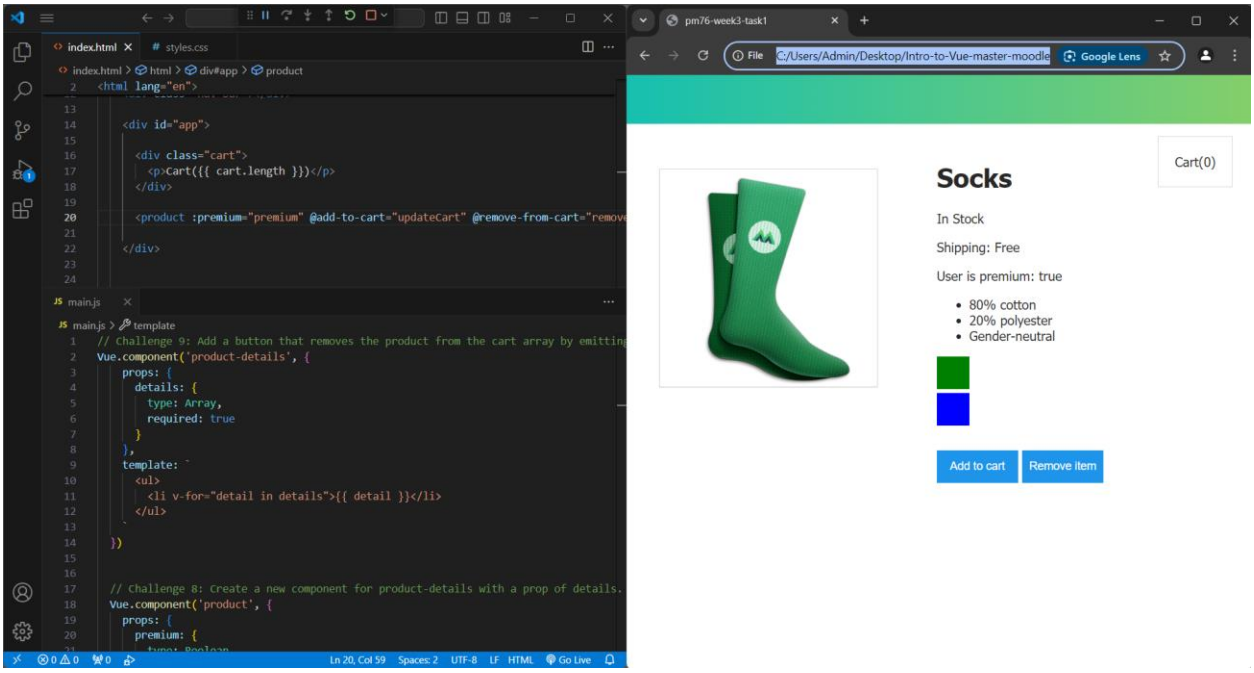
<h1>{{ product }}</h1>
In Stock
<p v-else>Out of Stock
pm76-week3-task1
File C:/Users/Admin/Desktop/Intro-to-Vue-master-moodle/index.html
Socks
In Stock
Shipping: Free
User is premium: true
80% cotton
20% polyester
Gender-neutral
Add to cart
Cart()


index.html
styles.css
<html lang="en">

Cart({{ cart.length }})

<product :premium="premium" @add-to-cart="updateCart" @remove-from-cart="remove

main.js
template:
<h1>{{ product }}</h1>
In Stock
<p v-else>Out of Stock
Shipping: {{ shipping }}
User is premium: {{ premium }}
<!-- Challenge 8-->
<product-details :details="details"></product-details>
<div class="color-box"
v-for="(variant, index) in variants"
:key="variant.variantId"
:style="{ backgroundColor: variant.variantColor }"
@mouseover="updateProduct(index)"
PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS
Download the Vue Devtools extension for a better development experience:
https://github.com/vuejs/vue-devtools
vue.js:9323
You are running Vue in development mode.
Make sure to turn on production mode when deploying for production.
See more tips at https://vuejs.org/guide/deployment.html
vue.js:9330
pm76-week3-task1
File C:/Users/Admin/Desktop/Intro-to-Vue-master-moodle/index.html
Cart(0)
Socks
In Stock
Shipping: Free
User is premium: true
80% cotton
20% polyester
Gender-neutral
Add to cart


index.html
styles.css
<html lang="en">

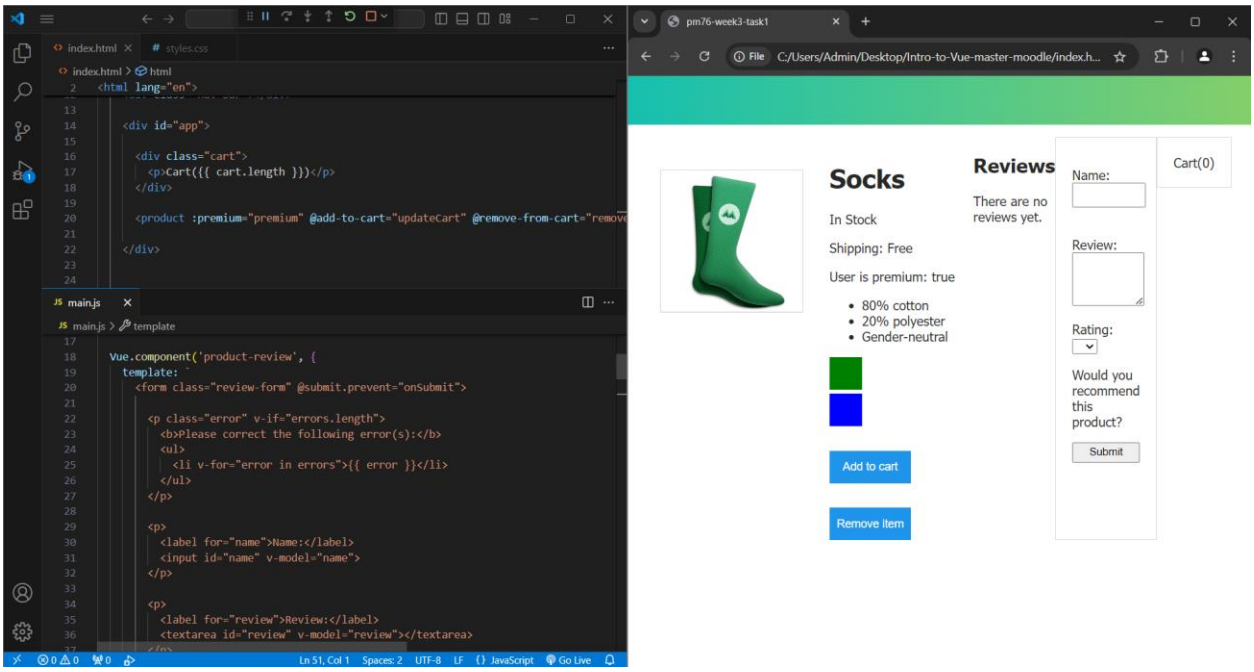
Cart({{ cart.length }})

<product :premium="premium" @add-to-cart="updateCart" @remove-from-cart="remove

main.js
// Challenge 9: Add a button that removes the product from the cart array by emitting
Vue.component('product-details', {
props: {
details: {
type: Array,
required: true
}
},
template:

<li v-for="detail in details">{{ detail }}

})
// Challenge 8: Create a new component for product-details with a prop of details.
Vue.component('product', {
props: {
premium: {
pm76-week3-task1
File C:/Users/Admin/Desktop/Intro-to-Vue-master-moodle
Google Lens
Cart(0)
Socks
In Stock
Shipping: Free
User is premium: true
80% cotton
20% polyester
Gender-neutral
Add to cart
Remove item


index.html
styles.css
<html lang="en">

Cart({{ cart.length }})

<product :premium="premium" @add-to-cart="updateCart" @remove-from-cart="remove

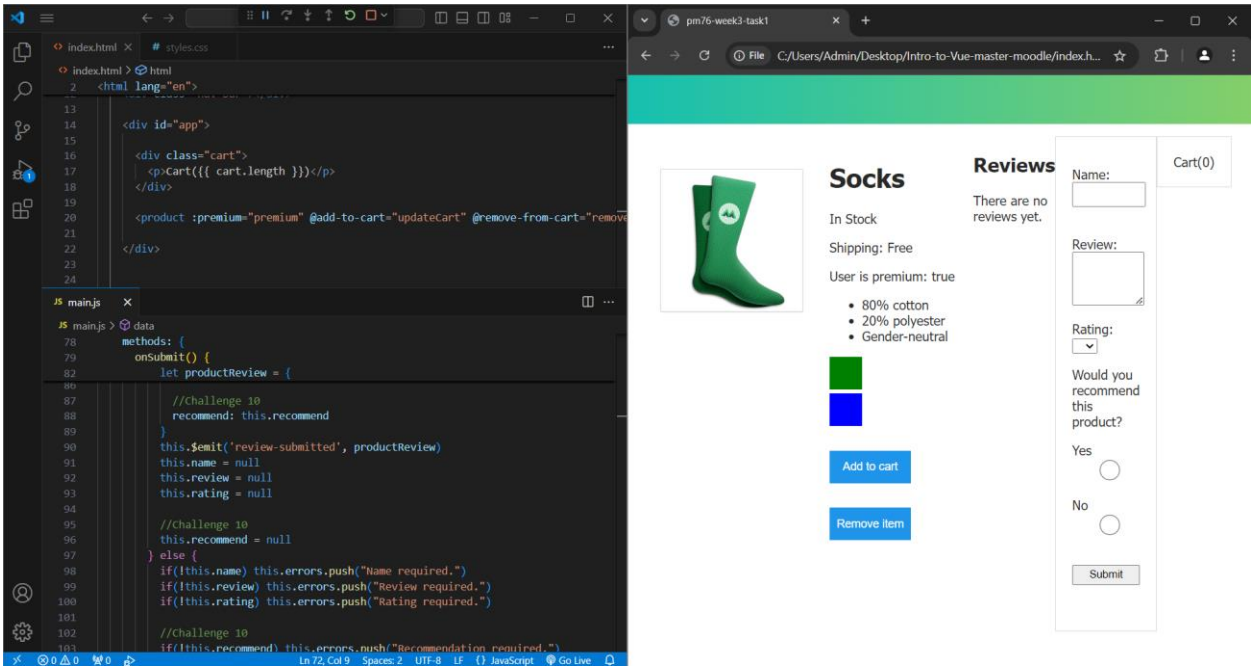
main.js
Vue.component('product-review', {
template:
<form class="review-form" @submit.prevent="onSubmit">

Please correct the following error(s):

<li v-for="error in errors">{{ error }}

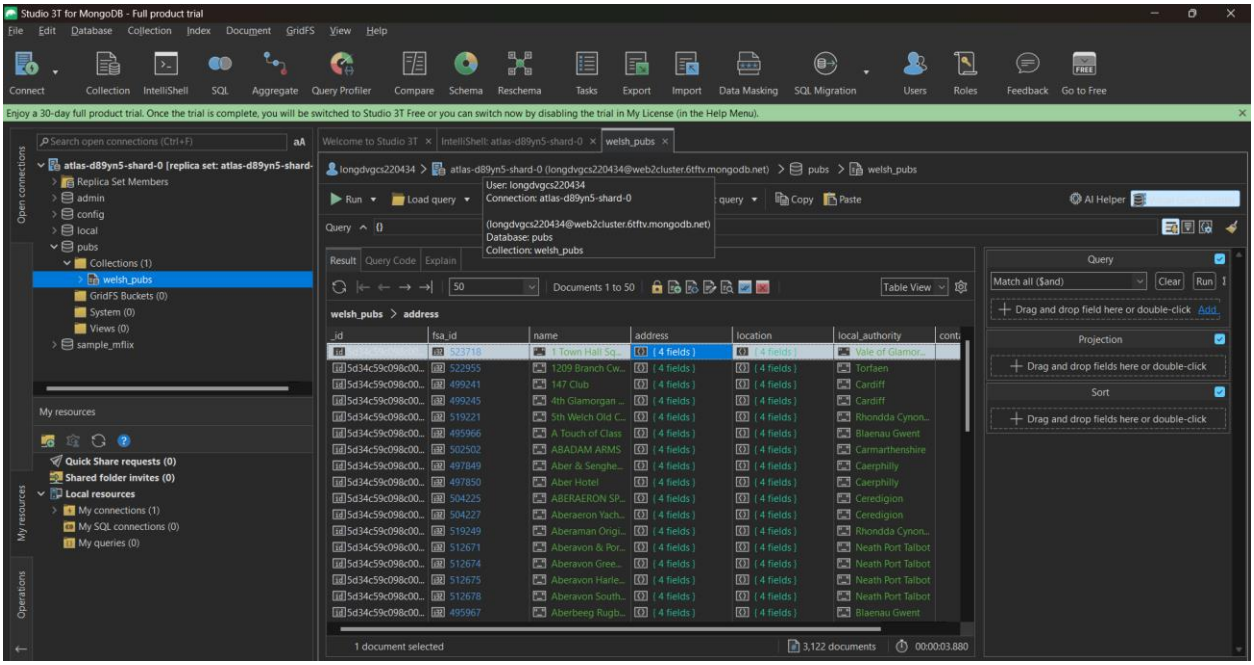
<label for="name">Name:</label>
<input id="name" v-model="name">

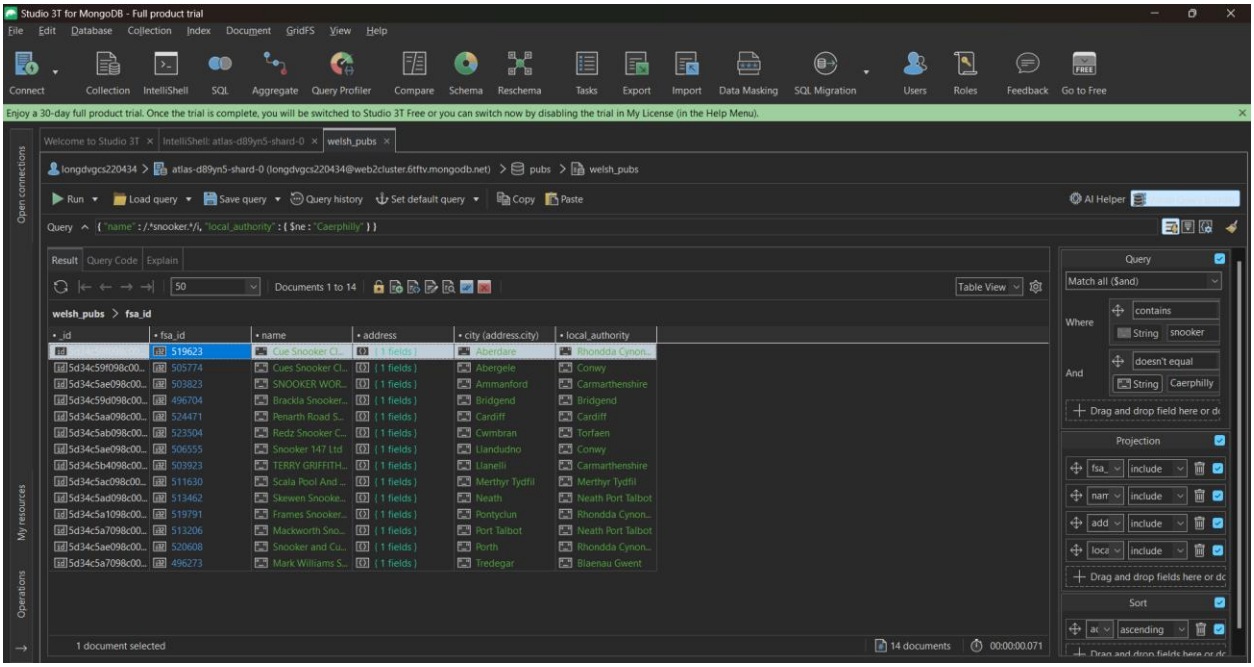
<label for="review">Review:</label>
<textarea id="review" v-model="review"></textarea>
pm76-week3-task1
File C:/Users/Admin/Desktop/Intro-to-Vue-master-moodle/index.h...
Cart(0)
Socks
In Stock
Shipping: Free
User is premium: true
80% cotton
20% polyester
Gender-neutral
Add to cart
Remove item
Reviews
There are no reviews yet.
Name:
Review:
Rating:
Would you recommend this product?
Submit

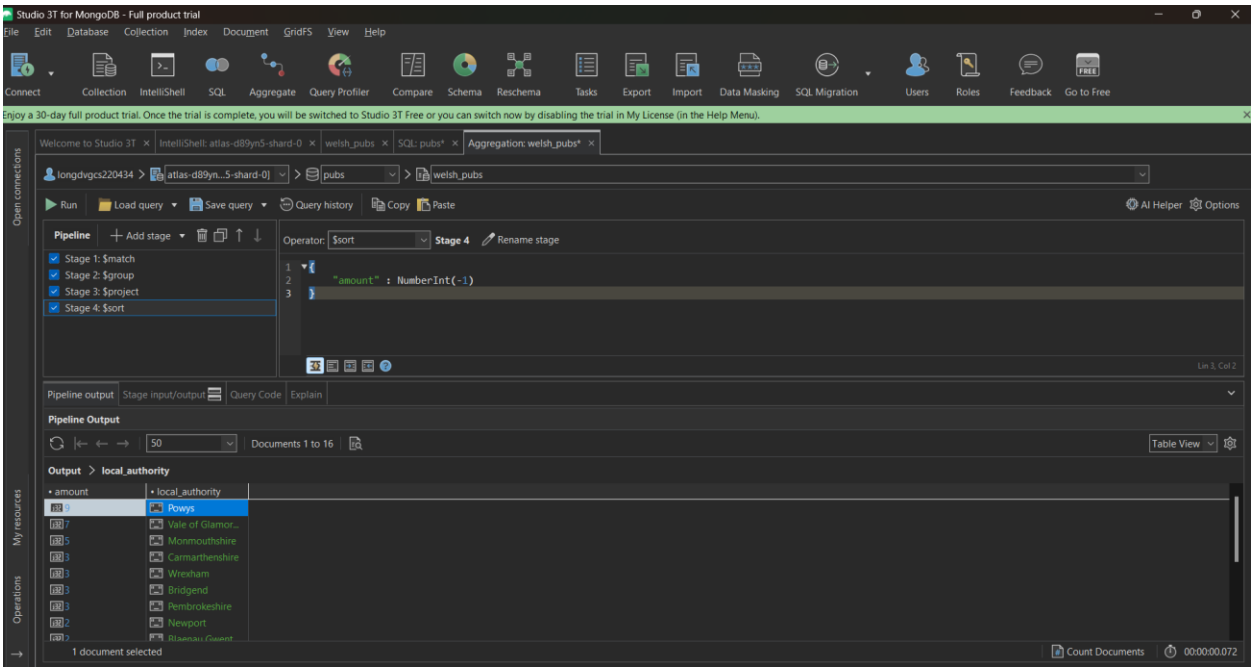


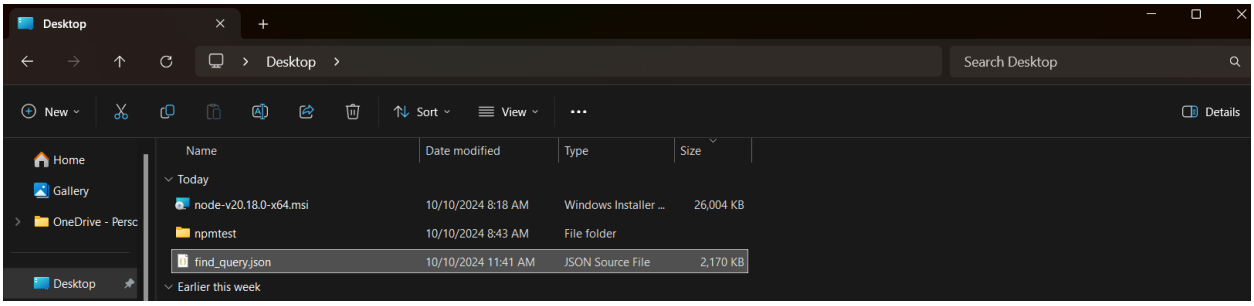
Evaluation:

In lesson 8, I learned how to create and use Vue components. Components help break down the user interface into smaller pieces, making the app more manageable. For example, I can pass data to these components using props, allowing different parts of the app to communicate with each other. This helps manage the UIs by dividing them into sections with their own functionality. In lesson 9, I learned about communicating events between components. I used $emit to send custom events from child to parent components, allowing the parent to listen for these events and respond. In lesson 10, I learned how to work with forms in Vue using the v-model directive for two-way data binding. For example, I could bind input fields directly to data in the Vue instance. Additionally, I learned how to handle form submissions and perform validation making data management more efficient when processing user inputs. These lessons deepen my understanding of building interactive Vue.js applications with components, communication between components, and handling user input.


Studio 3T for MongoDB - Full product trial
File Edit Database Collection Index Document GridFS View Help
Connect Collection IntelliShell SQL Aggregate Query Profiler Compare Schema Reschema Tasks Export Import Data Masking SQL Migration Users Roles Feedback Go to Free
Enjoy a 30-day full product trial. Once the trial is complete, you will be switched to Studio 3T Free or you can switch now by disabling the trial in My License (in the Help Menu).
Welcome to Studio 3T
IntelliShell: atlas-d89yn5-shard-0
welshpubs
longdvgcs220434 > atlas-d89yn5-shard-0 (longdvgcs220434@web2cluster.6tftv.mongodb.net) > pubs > welshpubs
User: longdvgcs220434
Connection: atlas-d89yn5-shard-0
(longdvgcs220434@web2cluster.6tftv.mongodb.net)
Database: pubs
Collection: welshpubs
Run Load query Copy Paste AI Helper
Result Query Code Explain
Documents 1 to 50
Table View
welshpubs > address
1 document selected
3,122 documents
00:00:03.880
Query
Match all ($and)
Clear Run
Drag and drop field here or double-click
Projection
Sort


Studio 3T for MongoDB - Full product trial
File Edit Database Collection Index Document GridFS View Help
Connect Collection IntelliShell SQL Aggregate Query Profiler Compare Schema Reschema Tasks Export Import Data Masking SQL Migration Users Roles Feedback Go to Free
Enjoy a 30-day full product trial. Once the trial is complete, you will be switched to Studio 3T Free or you can switch now by disabling the trial in My License (in the Help Menu).
Welcome to Studio 3T
IntelliShell: atlas-d89yn5-shard-0
welshpubs
longdvgcs220434 > atlas-d89yn5-shard-0 (longdvgcs220434@web2cluster.6tftv.mongodb.net) > pubs > welshpubs
Run Load query Save query Query history Set default query Copy Paste AI Helper
Query { "name" : /.*snooker.*/i, "local_authority" : { $ne : "Caerphilly" } }
Result Query Code Explain
Documents 1 to 14
Table View
welshpubs > fsa_id
1 document selected
14 documents
00:00:00.071
Query
Match all ($and)
Where contains String snooker
And doesn't equal String Caerphilly
Projection
Sort
ascending





Through the exploration of MongoDB basics, I gained valuable insights into managing NoSQL databases efficiently. I learned how to create collections from a .Json file, enabling the seamless import of structured data into MongoDB. Reviewing collections in various views enhanced my understanding of how data can be visualized and interpreted differently depending on the selected view. The use of the Visual Query Builder proved essential in constructing and executing queries without needing advanced knowledge of MongoDB query syntax, allowing for a more intuitive approach to data retrieval. Moreover, I developed the ability to update data directly within a collection, showcasing MongoDB's flexibility.

In addition, using SQL within MongoDB aggregation broadened my skills by enabling SQL queries on a NoSQL platform. I learned to run SQL queries using the SQL Query Tool, export those queries to the Aggregation Editor, and modify them as needed, which improved my proficiency in transforming SQL logic into MongoDB's aggregation framework.

Lastly, I acquired practical experience in importing and exporting data. Importing document data from a .csv file and exporting it to .Json files or new collections enhanced my data manipulation capabilities, crucial for database management and integration in real-world applications.

Appendix

Git Link

https://github.com/VinhLong47/WebProgrammingLab