

LAB 1: CHỈ THỊ TRONG VUE 3

Bài 1: Hiển thị thông tin sinh viên gồm họ tên, ngày sinh, giới tính, hình ảnh và điểm trung bình.



Yêu cầu:

- Hiển thị thông tin sinh viên bao gồm, họ tên, ngày sinh, phái, điểm

Hướng dẫn:

File index.html

```
<body class="container">
  <div id="app">
    <h5 class="text-center">THÔNG TIN SINH VIÊN</h5>
    
    <br>
    <span>Họ và tên : </span> {{ ten }} <br>
    <span>Ngày sinh : </span> {{ ngaysinh }} <br>
    <span>Phái : </span> {{ phai }} <br>
    <span>Điểm : </span> {{ diem }}
  </div>

  <script src="https://cdnjs.cloudflare.com/ajax/libs/jquery/3.5.1/jquery.slim.min.js"></script>
  <script src="https://cdnjs.cloudflare.com/ajax/libs/popper.js/1.16.1/umd/popper.min.js"></script>
  <script src="https://cdnjs.cloudflare.com/ajax/libs/twitter-bootstrap/4.6.0/js/bootstrap.min.js"></script>
  <script src="https://unpkg.com/vue@3.0.0-beta.12/dist/vue.global.js"></script>
  <script src="main.js"></script>
</body>
</html>
```

File main.js

```
const app = Vue.createApp({
  data() {
    return {
      image: 'img.jpg',
      ten: 'Nguyễn Văn Tèo',
      ngaysinh: '24/3/2004',
      phai: 'Nam',
      diem: 8.5
    }
  }
});

// Mount the app to the #app element
app.mount('#app');
```

Bài 2: Xây dựng danh sách sản phẩm có giao diện như sau:

Image	Product	Code	Available	Price	5 star rating		
	1	Leaf Rake	GDN-0011	March 19, 2016	Leaf rake with 48-inch wooden handle.	19.95	3.2
	2	Garden Cart	GDN-0023	March 18, 2016	15 gallon capacity rolling garden cart	32.99	4.2
	5	Hammer	TBX-0048	May 21, 2016	Curved claw steel hammer	8.9	4.8
	8	Saw	TBX-0022	May 15, 2016	15-inch steel blade hand saw	11.55	3.7
	10	Video Game Controller	GMG-0042	October 15, 2015	Standard two-button video game controller	35.95	4.6

Hướng dẫn:

File index.html

```

<body class="container text-center">
<div id="app">
  <table style="width: 100%" class="table table-bordered">
    <tr>
      <th>Image</th>
      <th>Product</th>
      <th>Code</th>
      <th>Available</th>
      <th>Price</th>
      <th>5 star rating</th>
    </tr>
    <tr v-for="item in list">
      <!-- <td> </td>
      <td>{{item.name}}</td>
      <td>{{item.birth}}</td>
      <td>{{item.gender}}</td>
      <td>{{item.point}}</td> -->
      <td> </td>
      <td>{{item.productId}}</td>
      <td>{{item.productName}}</td>
      <td>{{item.productCode}}</td>
      <td>{{item.releaseDate}}</td>
      <td>{{item.description}}</td>
      <td>{{item.price}}</td>
      <td>{{item.starRating}}</td>
    </tr>
  </table>
</div>
<script src="https://cdnjs.cloudflare.com/ajax/libs/jquery/3.5.1/jquery.slim.min.js"></script>
<script src="https://cdnjs.cloudflare.com/ajax/libs/popper.js/1.16.1/umd/popper.min.js"></script>
<script src="https://cdnjs.cloudflare.com/ajax/libs/twitter-bootstrap/4.6.0/js/bootstrap.min.js"></script>
<script src="https://unpkg.com/vue@3.0.0-beta.12/dist/vue.global.js"></script>
<script src="/main.js"></script>

```

File main.js với khai báo list như sau:

list:

```

{
  "productId": 1,
  "productName": "Leaf Rake",
  "productCode": "GDN-0011",
  "releaseDate": "March 19, 2016",
  "description": "Leaf rake with 48-inch wooden handle.",
  "price": 19.95,
  "starRating": 3.2,
  "imageUrl":
"http://openclipart.org/image/300px/svg_to_png/26215/Anonymous_Leaf_Rake.png"
},
{
  "productId": 2,
  "productName": "Garden Cart",
  "productCode": "GDN-0023",

```

```
    "releaseDate": "March 18, 2016",
    "description": "15 gallon capacity rolling garden cart",
    "price": 32.99,
    "starRating": 4.2,
    "imageUrl":
"http://openclipart.org/image/300px/svg_to_png/58471/garden_cart.png"
  },
  {
    "productId": 5,
    "productName": "Hammer",
    "productCode": "TBX-0048",
    "releaseDate": "May 21, 2016",
    "description": "Curved claw steel hammer",
    "price": 8.9,
    "starRating": 4.8,
    "imageUrl":
"http://openclipart.org/image/300px/svg_to_png/73/rejon_Hammer.png"
  },
  {
    "productId": 8,
    "productName": "Saw",
    "productCode": "TBX-0022",
    "releaseDate": "May 15, 2016",
    "description": "15-inch steel blade hand saw",
    "price": 11.55,
    "starRating": 3.7,
    "imageUrl":
"http://openclipart.org/image/300px/svg_to_png/27070/egore911_saw.png"
  },
  {
    "productId": 10,
    "productName": "Video Game Controller",
    "productCode": "GMG-0042",
    "releaseDate": "October 15, 2015",
```

```

    "description": "Standard two-button video game controller",
    "price": 35.95,
    "starRating": 4.6,
    "imageUrl": "http://openclipart.org/image/300px/svg_to_png/120337/xbox-controller_01.png"
  }
]

```

Bài 3: To-Do-List đơn giản

To-Do List

Add

Hình a

Yêu cầu:

- Xây dựng một ứng dụng To-Do List đơn giản cho phép người dùng thêm, sửa, và xóa các công việc cần làm.
- Sử dụng: v-for, v-model, v-bind, v-on.

Add

Học javascript	Delete
Học Vue 3	Delete
Học PHP	Delete
Học Typescript	Delete

Hình b: thêm việc

To-Do List

Add

Học javascript	Delete
Học Vue 3	Delete
Học Typescript	Delete

Hình b: xóa việc

Hướng dẫn:

File index.html

```
<body class="container">
  <div id="app">
    <h1 class="text-center my-4">To-Do List</h1>

    <div class="input-group mb-3">
      <input v-model="newTask" type="text" class="form-control" placeholder="Add a new task" aria-label="Add a new task">
      <div class="input-group-append">
        <button class="btn btn-primary" @click="addTask">Add</button>
      </div>
    </div>

    <ul class="list-group">
      <li v-for="(task, index) in tasks" :key="index" class="list-group-item d-flex justify-content-between align-items-center">
        <input v-model="task.text" class="form-control" @blur="updateTask(index)" />
        <button class="btn btn-danger" @click="deleteTask(index)">Delete</button>
      </li>
    </ul>
  </div>
  <script src="https://unpkg.com/vue@3.0.0-beta.12/dist/vue.global.js"></script>

  <script src="main.js"></script>
</body>
```

File main.js

```
const { createApp } = Vue;

createApp({
  data() {
    return {
      newTask: '',
      tasks: []
    };
  },
  methods: {
    addTask() {
      if (this.newTask.trim() !== '') {
        this.tasks.push({ text: this.newTask });
        this.newTask = ''; // Clear the input field
      }
    },
    deleteTask(index) {
      this.tasks.splice(index, 1);
    },
    updateTask(index) {
      if (this.tasks[index].text.trim() === '') {
        this.deleteTask(index);
      }
    }
  }
}).mount('#app');
```

Bài 4: Quản lý sản phẩm**Product Management**

Product Name

Product Price

Add Product

Search product by name

Yêu cầu:

- Tạo một ứng dụng quản lý sản phẩm với khả năng thêm, chỉnh sửa, xóa sản phẩm, và tìm kiếm sản phẩm theo tên.
- **Sử dụng:**
 - ✓ v-model: Liên kết hai chiều với dữ liệu đầu vào của người dùng.
 - ✓ v-bind: Liên kết thuộc tính của các phần tử HTML với dữ liệu trong Vue.
 - ✓ v-if: Hiển thị hoặc ẩn các phần tử dựa trên điều kiện.
 - ✓ v-on: Xử lý sự kiện DOM.
 - ✓ v-for: Lặp qua các mục trong danh sách.
 - ✓ v-mouse: Xử lý các sự kiện chuột.

Hướng dẫn**Dữ liệu**

```

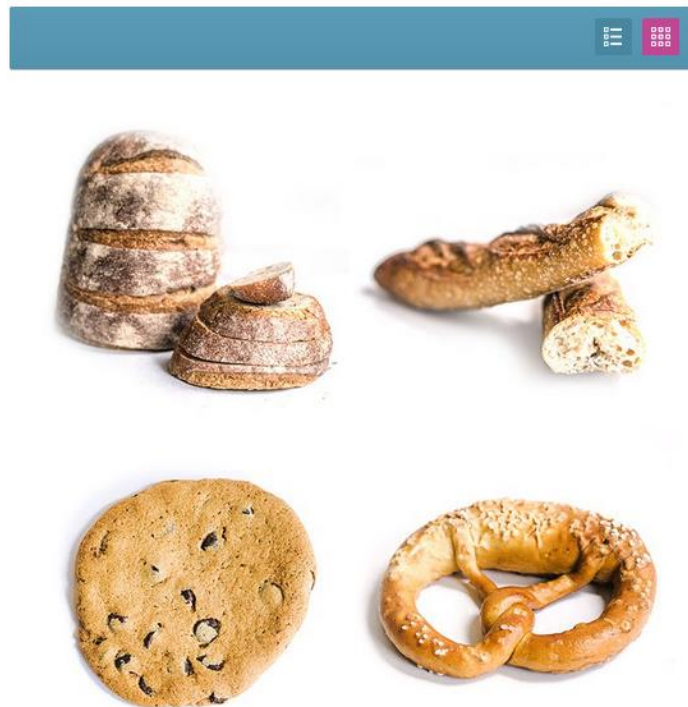
data() {
  return {
    product: {
      name: '',
      price: 0
    },
    products: [],
    searchQuery: '',
    editIndex: null
  };
},

```

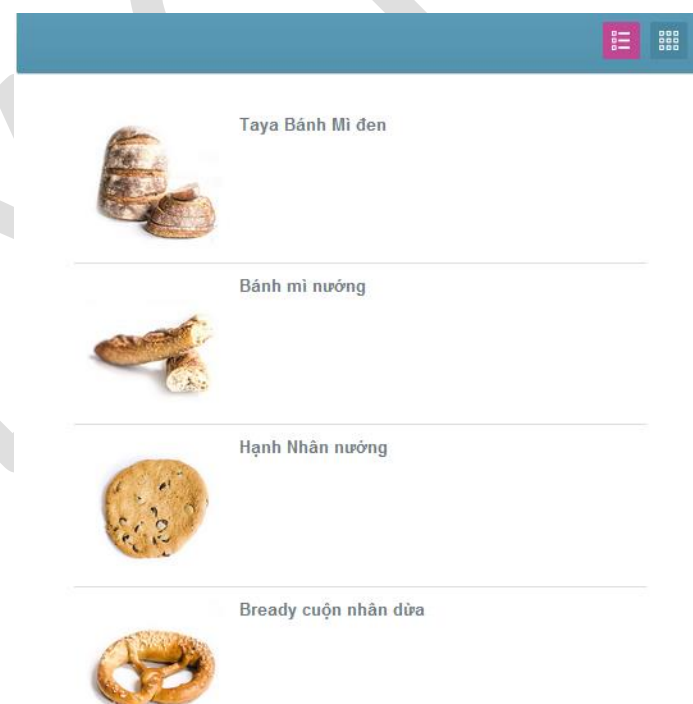
Sử dụng computed để filter sản phẩm

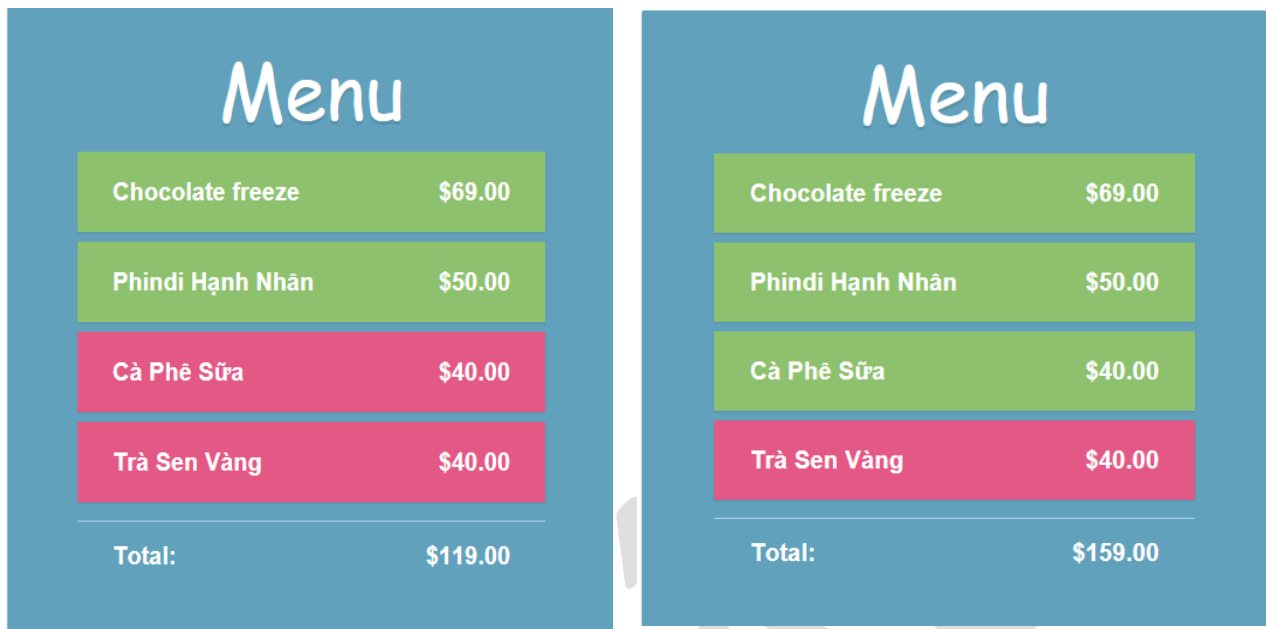
Các phương thức thêm, xóa, sửa thực hiện trong methods

Bài 5: Thiết kế layout hiển thị dạng list và grid như sau

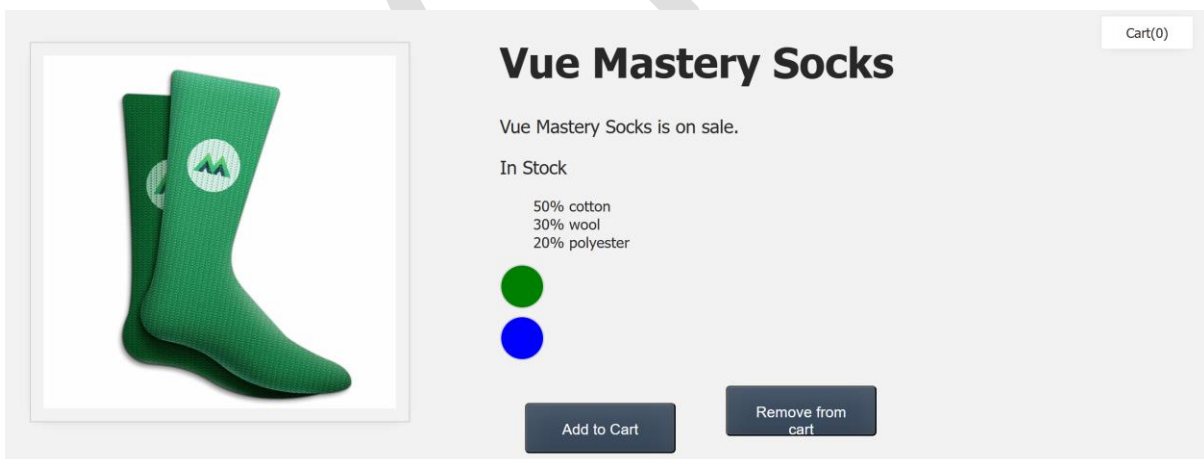


Dạng List



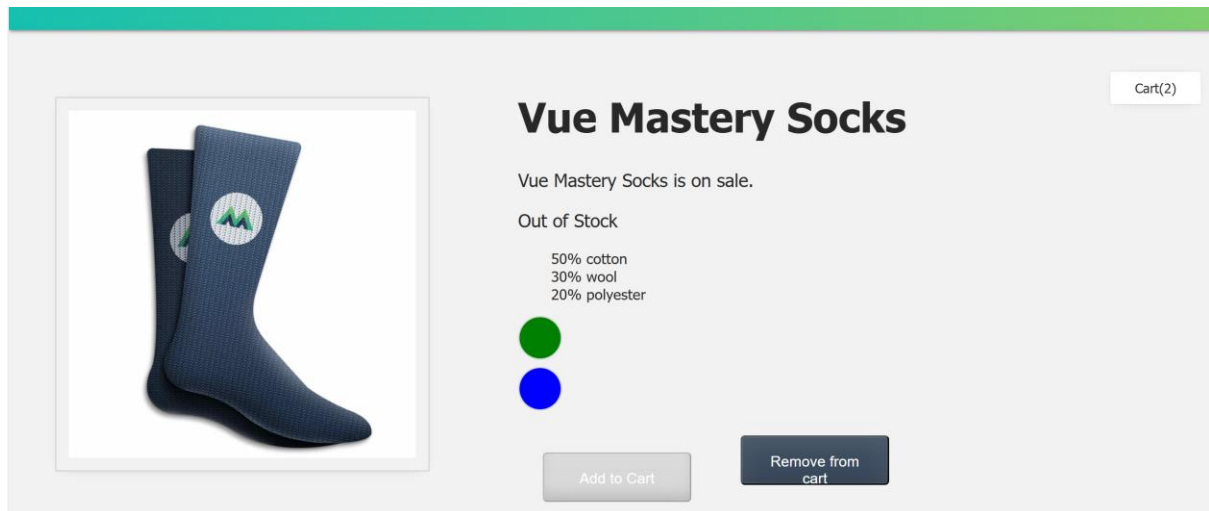
Bài 6: Thiết kế OrderForm như mẫu**Yêu cầu:**

Người dùng chọn sản phẩm nào thì sẽ tự động tính tổng tiền, nếu trả món thì click vào món mình trả, hệ thống tự động cập nhật trừ lại tiền trong tổng tiền, sản phẩm nào được chọn thì hiển thị màu xanh, chưa chọn hoặc trả món thì sẽ màu hồng.

Bài 7: Thực hiện chức năng như: conditional rendering, list rendering, Event Handling, class & style Binding để thực hiện các tính năng như mẫu:**Yêu cầu:**

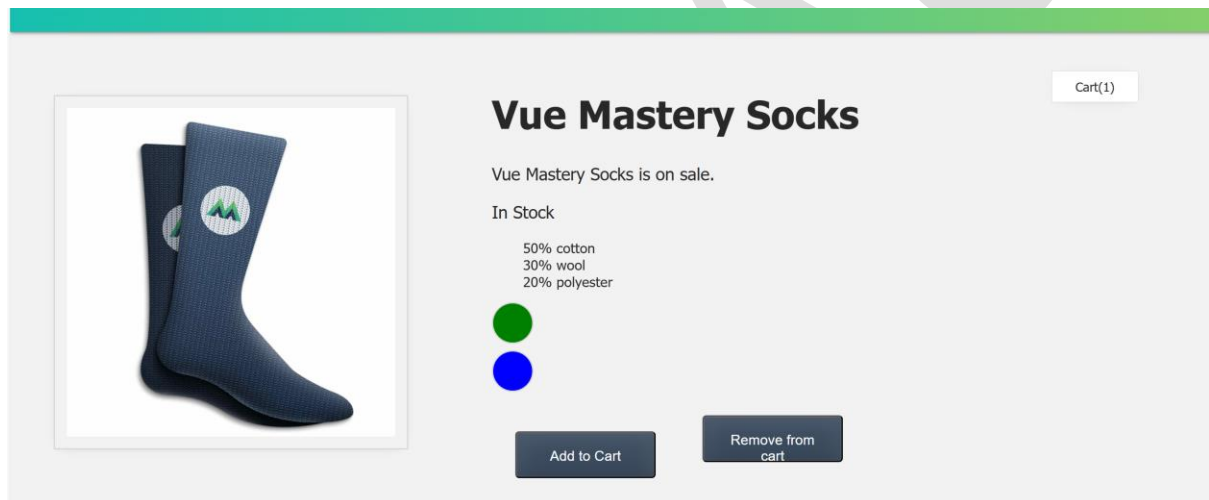
Chọn màu hiển thị socks màu tương ứng

Nếu sản phẩm socks có màu tương ứng hết hàng khi người dùng nhấn Add to cart thì hiển thị Out of Stock và ẩn nút Add to Cart (tức số lượng mua > số lượng hiện có)



Sản phẩm nào đang sale thì hiển thị **on sale**

Nếu Nhấn vào nút Remove from cart thì số lượng được trả lại vào đúng sản phẩm được chọn



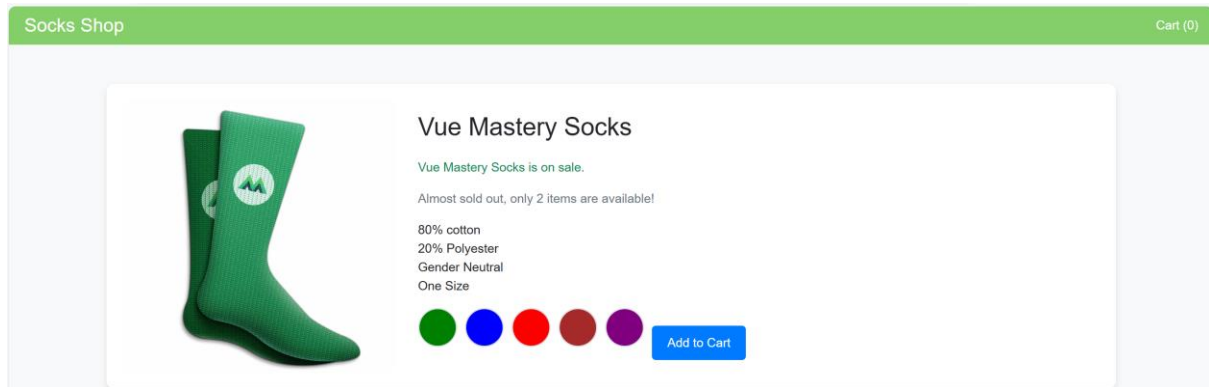
Hướng dẫn:

Dữ liệu

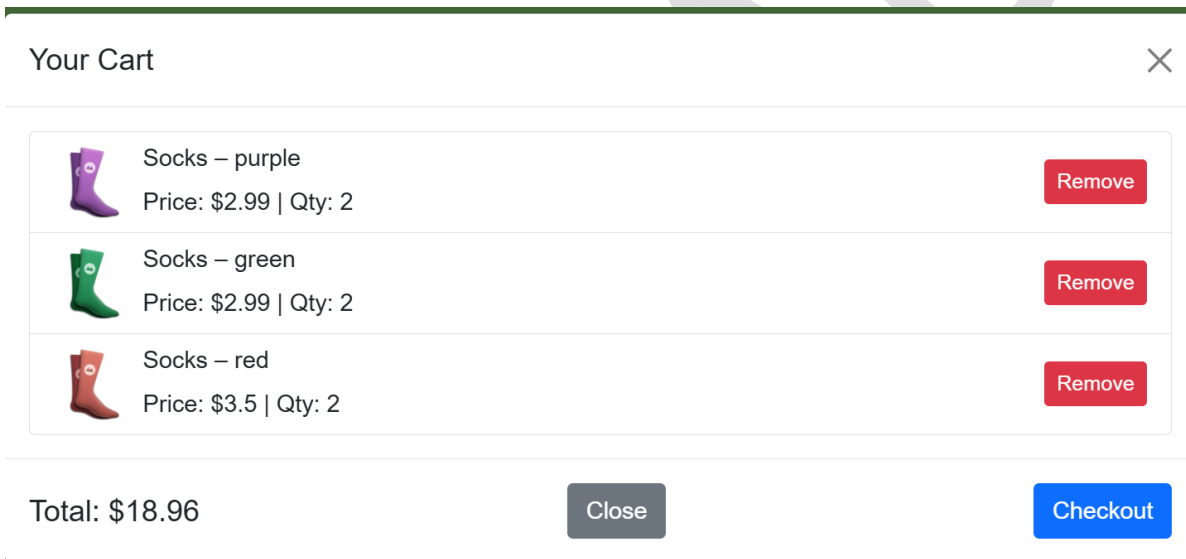
```
data() {
  return {
    cart: 0,
    product: 'Socks',
    brand: 'Vue Mastery',
    selectedVariant: 0,
    details: ['50% cotton', '30% wool', '20% polyester'],
    variants: [
      { id: 2234, color: 'green', image: './assets/images/socks_green.jpg', quantity: 3 },
      { id: 2235, color: 'blue', image: './assets/images/socks_blue.jpg', quantity: 2 },
    ],
    onSale: true
  }
},
```

Phương thức thực hiện thêm, xóa, cập nhật

Computed: thực hiện hiển thị title, image, inStock và sale

Bài 8: Mở rộng từ bài 7**Yêu cầu:**

- Khi hover vào mỗi màu sắc thì tương ứng sẽ hiển thị hình, mà số lượng cho mỗi sock
- Sản phẩm được thêm vào giỏ hàng như sau:



- ✓ Thực hiện chức năng xóa thì sẽ trả về số lượng tồn trong hàng
- Nếu sản phẩm mua đã hết hàng thì sẽ hiện thị “Out of stock”



- Khi remove trong giỏ hàng thì số lượng được cập nhật lại như sau



Hướng dẫn:

Khai báo data như sau

```
const app = Vue.createApp({
  data() {
    return {
      cart: [],
      product: 'Socks',
      brand: 'Vue Mastery',
      selectedVariant: 0,
      details: ['80% cotton', '20% Polyester', 'Gender Neutral', 'One Size'],
      variants: [
        { id: 2234, color: 'green', image: './assets/images/socks_green.jpg', price: 2.99, quantity: 2 },
        { id: 2235, color: 'blue', image: './assets/images/socks_blue.jpg', price: 3.5, quantity: 3 },
        { id: 2236, color: 'red', image: './assets/images/socks_red.jpg', price: 3.5, quantity: 10 },
        { id: 2237, color: 'brown', image: './assets/images/socks_brown.jpg', price: 3.99, quantity: 8 },
        { id: 2238, color: 'purple', image: './assets/images/socks_purple.jpg', price: 2.99, quantity: 10 }
      ],
      onSale: true
    }
  }
})
```