

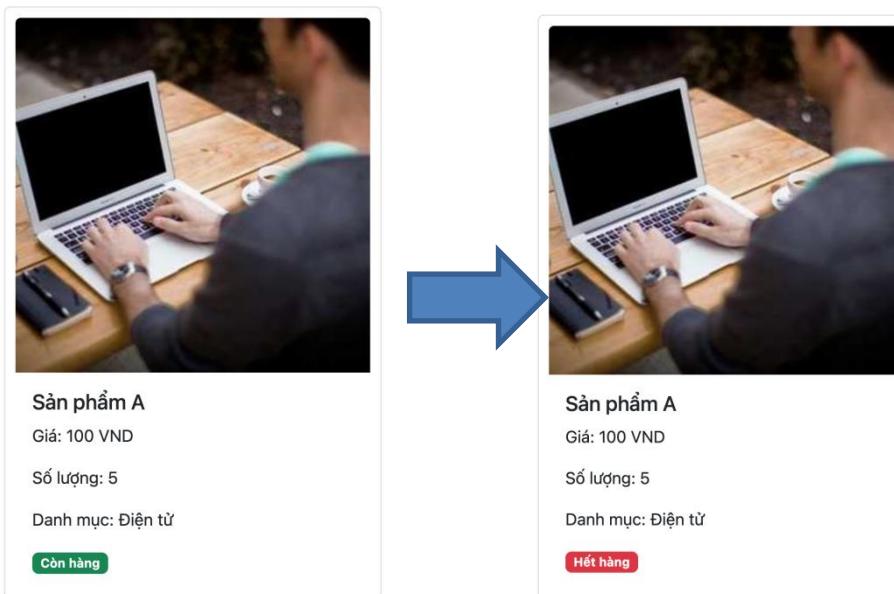
MỤC TIÊU:

Kết thúc bài thực hành này bạn có khả năng

- ✓ Hiểu và vận dụng v-if, v-show, v-else, v-else-if trong Vue.js.
- ✓ Thực hành các kỹ thuật hiển thị danh sách bằng v-for với object, destructuring, và template.
- ✓ Sử dụng các logic điều kiện và hiển thị phức tạp trong ứng dụng quản lý sản phẩm.

NỘI DUNG**PHẦN I:****Bài 1 (2 điểm) :** Hiển thị trạng thái sản phẩm bằng v-if và v-show**Yêu cầu:**

1. Sản phẩm cần có ít nhất các thuộc tính: id, name, price, quantity, category, inStock.
2. Sử dụng v-if để hiển thị thông báo “Sản phẩm đã hết hàng” nếu inStock là false.
3. Sử dụng v-show để hiển thị chi tiết sản phẩm (tên, giá, số lượng) khi sản phẩm còn hàng.

**Hướng dẫn:**

Bước 1: Tạo 1 file mới trong thư mục **Labs/Lab-3/Bai1.vue**.

```
<template>
<div class="card">
    
    <div class="card-body">
        <h5 class="card-title">{{ product.name }}</h5>
        <p class="card-text">Giá: {{ product.price }} VND</p>
        <p class="card-text">Số lượng: {{ product.quantity }}</p>
        <p class="card-text">Danh mục: {{ product.category }}</p>
        <span
            class="badge"
            :class="{
                'bg-success': product.inStock,
                'bg-danger': !product.inStock,
            }"
        >
            {{ product.inStock ? "Còn hàng" : "Hết hàng" }}
        </span>
    </div>
</div>
</template>
<script setup>
import { reactive } from "vue";
const product = reactive({
    id: 1,
    name: "Sản phẩm A",
    price: 100,
    quantity: 5,
    category: "Điện tử",
    inStock: false,
    description: "Mô tả A",
    imageUrl: "https://picsum.photos/id/1/300/300",
});
</script>
```

Bước 2: Nhúng Component **Labs/Lab-3/Bai1.vue** ở **App.vue**

```
<script setup>
import Bai1 from "./Labs/Lab-3/Bai1.vue";
</script>
<template>
<div id="app">
    <Bai1 />
</div>
</template>
```

Bài 2 (2 điểm): Sử dụng v-else-if để phân loại số lượng sản phẩm

Yêu cầu:

1. Dựa vào dữ liệu bài 1, sử dụng v-else-if để phân loại số lượng sản phẩm.
2. Nếu quantity > 10, hiển thị “Số lượng lớn”.
3. Nếu quantity từ 1 đến 10, hiển thị “Số lượng vừa phải”.
4. Nếu quantity bằng 0, hiển thị “Hết hàng”.

Hướng dẫn:

Bước 1: Tạo 1 file mới trong thư mục **Labs/Lab-3/Bai2.vue**.

```
<template>
<div class="card">
    
    <div class="card-body">
        <h5 class="card-title">{{ product.name }} - {{ product.category }}</h5>
        <p v-if="product.quantity > 10" class="text-success">Số lượng lớn</p>
        <p v-else-if="product.quantity > 0" class="text-warning">Số lượng vừa phải</p>
        <p v-else class="text-danger">Hết hàng</p>
        <p>Giá: {{ product.price }} VND</p>
    </div>
</div>
</template>

<script setup>
import { reactive } from "vue";
const product = reactive({
    id: 1,
    name: "Sản phẩm A",
    price: 100,
    quantity: 5,
    category: "Điện tử",
    inStock: false,
    description: "Mô tả A",
    imageUrl: "https://picsum.photos/id/1/300/300",
});
</script>
```

Bước 2: Nhúng Component **Labs/Lab-3/Bai2.vue**. Ở **App.vue**

```
<script setup>
import Bai2 from "./Labs/Lab-3/Bai2.vue";
</script>
```

<template>

```
<div id="app">
    <Bai2 />
</div>
</template>
```

PHẦN II:

Bài 3 (3 điểm): Hiển thị danh sách sản phẩm

Yêu cầu:

2. Hiển thị danh sách các sản phẩm, mỗi sản phẩm có các thuộc tính: id, name, price, quantity, category, inStock.
3. Sử dụng v-for để lặp qua danh sách sản phẩm và hiển thị tên, giá và trạng thái inStock (còn hàng/hết hàng).
4. Sử dụng destructuring để lấy các thuộc tính từ object sản phẩm.

Danh sách sản phẩm

#	Ảnh	Tên	Giá	Tình trạng
1		Laptop	15000000 VND	Còn hàng
2		Phone	8000000 VND	Còn hàng
3		Headphone	2000000 VND	Hết hàng

Hướng dẫn:

Bước 1: Tạo 1 file mới trong thư mục **Labs/Lab-3/Bai3.vue**.

```
<template>
<div class="container mt-5">
<h2>Danh sách sản phẩm</h2>
<table class="table">
    <thead>
```

```

<tr>
  <th scope="col">#</th>
  <th scope="col">Ảnh</th>
  <th scope="col">Tên</th>
  <th scope="col">Giá</th>
  <th scope="col">Tình trạng</th>
</tr>
</thead>
<tbody>
<tr v-for="({ id, name, price, inStock, image }, index) in products" :key="id">
  <th scope="row">{{ index + 1 }}</th>
  <td></td>
  <td>{{ name }}</td>
  <td>{{ price }} VND</td>
  <td>
    <span v-if="inStock" class="badge bg-success">Còn hàng</span>
    <span v-else class="badge bg-danger">Hết hàng</span>
  </td>
</tr>
</tbody>
</table>
</div>
</template>

<script setup>
import { reactive } from "vue";
const products = reactive([
  {
    id: 1,
    name: "Laptop",
    price: 15000000,
    quantity: 5,
    category: "Electronics",
    inStock: true,
    image: "https://picsum.photos/id/1/300/300",
  },
  {
    id: 2,
    name: "Phone",
    price: 8000000,
    quantity: 12,
    category: "Electronics",
    inStock: true,
    image: "https://picsum.photos/id/2/300/300",
  },
  {
    id: 3,
    name: "Headphone",
    price: 2000000,
    quantity: 0,
    category: "Accessories",
    inStock: false,
    image: "https://picsum.photos/id/3/300/300",
  },
]);

```

</script>

BƯỚC 2: Nhúng Component **Labs/Lab-3/Bai3.vue** ở **App.vue**

```
<script setup>

import Bai3 from "./Labs/Lab-3/Bai3.vue";

</script>

<template>

<div id="app">
    <Bai3 />
</div>

</template>
```

Bài 4: Hiển thị danh sách sản phẩm với template và điều kiện**Yêu cầu:**

1. Trong file **Labs/Lab-3/Bai4.vue**, hiển thị danh sách sản phẩm và trạng thái inStock.
2. Sử dụng v-for và template để lặp qua các sản phẩm và hiển thị chi tiết sản phẩm.
3. Sử dụng v-if để kiểm tra trạng thái hàng tồn kho (inStock), và v-for để lặp qua tất cả các thuộc tính của từng sản phẩm.
4. Tính tổng giá trị hàng tồn kho
5. Lọc sản phẩm theo danh mục

Quản lý sản phẩm

Lọc theo danh mục

Tất cả

Tổng giá trị hàng tồn kho: 171000000 VND



Laptop - Electronics

Còn hàng

Giá: 15000000 VND

Mô tả: Laptop thế hệ mới



Phone - Electronics

Còn hàng

Giá: 8000000 VND

Mô tả: Điện thoại thông minh



Headphone - Accessories

Hết hàng

Giá: 2000000 VND

Mô tả: Tai nghe chất lượng cao

Hướng dẫn:

Bước 1: Tạo 1 file mới trong thư mục **Labs/Lab-3/Bai4.vue**.

```
<template>
<div class="container">
<h2>Quản lý sản phẩm</h2>
<!-- Dropdown Lọc sản phẩm -->
<div class="mb-3">
<label for="categoryFilter">Lọc theo danh mục</label>
    <select v-model="selectedCategory" class="form-select" id="categoryFilter">
        <option value="">Tất cả</option>
        <option value="Electronics">Electronics</option>
        <option value="Accessories">Accessories</option>
    </select>
</div>
<!-- Tổng giá trị sản phẩm còn trong kho -->
<div class="alert alert-info">Tổng giá trị hàng tồn kho: {{ totalStockValue }} VND</div>
<!-- Danh sách sản phẩm -->
<div class="row">
<template
    v-for="product in filteredProducts"
    :key="product.id"
    class="product-template"
>
<div class="col-md-4">
    <div class="card">
        
        <div class="card-body">
```

```

<h5 class="card-title">{{ product.name }} - {{ product.category }}</h5>
<p v-if="product.inStock" class="text-success">Còn hàng</p>
<p v-else class="text-danger">Hết hàng</p>
<p>Giá: {{ product.price }} VND</p>
<p>Mô tả: {{ product.description }}</p>
</div>
</div>
</template>
</div>
</div>
</template>
<script setup>
import { reactive, computed, ref } from "vue";
const selectedCategory = ref(""); // Lọc theo danh mục
const products = reactive([
    {
        id: 1,
        name: "Laptop",
        price: 15000000,
        quantity: 5,
        category: "Electronics",
        inStock: true,
        description: "Laptop thế hệ mới",
        imageUrl: "https://picsum.photos/id/1/150/150",
    },
    {
        id: 2,
        name: "Phone",
        price: 8000000,
        quantity: 12,
        category: "Electronics",
        inStock: true,
        description: "Điện thoại thông minh",
        imageUrl: "https://picsum.photos/id/2/150/150",
    },
    {
        id: 3,
        name: "Headphone",
        price: 2000000,
        quantity: 0,
        category: "Accessories",
        inStock: false,
        description: "Tai nghe chất lượng cao",
        imageUrl: "https://picsum.photos/id/3/150/150",
    },
]);
// Tính tổng giá trị hàng tồn kho
const totalStockValue = computed(() => {
    return products.reduce((total, product) => {
        return product.inStock ? total + product.price * product.quantity : total;
    }, 0);
});
// Lọc sản phẩm theo danh mục
const filteredProducts = computed(() => {

```

```
if (!selectedCategory.value) {
    return products;
}
return products.filter((product) => product.category === selectedCategory.value);
});
```

Bước 2: Nhúng Component Labs/Lab-3/Bai4.vue. ở App.vue

```
<script setup>

import Bai4 from "./Labs/Lab-3/Bai4.vue";

</script>

<template>
<div id="app">
    <Bai4 />
</div>
</template>
```

**** Giảng viên có thể cho các bài tập phù hợp phục vụ assignment.**

***** Yêu cầu nộp bài:**

SV nén file (*hoặc share thư mục google drive*) bao gồm các yêu cầu đã thực hiện trên, nộp LMS đúng thời gian quy định của giảng viên. KHÔNG NỘP BÀI COI NHƯ KHÔNG CÓ ĐIỂM

--- Hết ---