

## MỤC TIÊU:

Kết thúc bài thực hành này bạn có khả năng thao tác với:

- ✓ Conditional rendering và List rendering trong VueJS
- ✓ Thực hành được v-if, v-else, v-else-if, v-show
- ✓ Phân biệt trường hợp sử dụng v-if và v-show
- ✓ Tìm hiểu v-for và các thành phần mở rộng

## NỘI DUNG

### PHẦN I: Conditional Rendering trong VueJS

**Bài 1 (2 điểm):** Nhập vào điểm trung bình của học sinh và in ra xếp loại tương ứng. Yêu cầu:

- 🚩 Nếu ĐTB < 5.0 thì xếp loại yếu
- 🚩 Nếu  $5.0 \leq \text{ĐTB} < 6.5$  thì xếp loại trung bình
- 🚩 Nếu  $6.5 \leq \text{ĐTB} < 8.0$  thì xếp loại khá
- 🚩 Nếu  $8.0 \leq \text{ĐTB} < 9$  thì xếp loại giỏi.
- 🚩 Nếu  $\text{ĐTB} \geq 9.0$  thì xếp loại xuất sắc.

**Nhập điểm của bạn:**

9

Xuất sắc

**Bài 2 (3 điểm):** Viết chương trình nhập một tháng bất kỳ và in ra mùa tương ứng.

Yêu cầu:

- 🚩 Sử dụng v-if, v-else-if...
- 🚩 Tháng < 1 hoặc >12 thì in ra thông điệp "Vui lòng nhập tháng từ 1 đến 12!"

Nhập tháng của bạn:

Mùa Xuân



Nhập tháng của bạn:

Vui lòng nhập tháng từ 1 đến 12!

Gợi ý tham khảo:

```
<template>
  <div class="container">
    <h3>Nhập tháng của bạn:</h3>
    <input type="number" v-model.number="month" />
    <p v-if="month < 1 || month > 12" style="color: red">Vui lòng nhập tháng từ 1 đến 12!</p>
    <p v-else-if="month >= 1 && month <= 3">Mùa Xuân</p>
    <p v-else-if="month <= 6">Mùa Hè</p>
    <p v-else-if="month <= 9">Mùa Thu</p>
    <p v-else>Mùa Đông</p>
  </div>
</template>

<script setup>
import { ref } from 'vue';

const month = ref('1');
</script>
```

## PHẦN II: Thao tác với List Rendering

**Bài 3 (3 điểm):** Hiển thị danh sách thông tin bài viết gồm hình ảnh, tiêu đề, nội dung mô tả như hình dưới đây. Yêu cầu:

- Sử dụng lại Bài 4 Lab 4
- Thay thế cách hiển thị danh sách bài viết bằng cách sử dụng **v-for**

## Kiến thức sức khỏe cộng đồng



### 8 loại rau củ quả giàu canxi

Canxi là khoáng chất cần thiết đối với cơ thể người. Có nhiều cách để bổ sung canxi, trong đó bổ sung qua đường ăn uống là cách tốt nhất. Có 8 loại rau củ quả giàu canxi...

[Xem chi tiết](#)


### Các loại gia vị tốt cho sức khỏe

Một số loại gia vị cung cấp nhiều polyphenol chống oxy hóa cao hơn các loại gia vị quen thuộc khác. Với một lượng nhỏ các món ăn đã thêm hương thơm và vị hấp dẫn...

[Xem chi tiết](#)


### 9 loại đậu bổ dưỡng nên dùng nhiều

Đậu lăng, đậu nành, đậu phộng, đậu Hà Lan giàu chất xơ, protein cùng nhiều vitamin và khoáng chất giúp giảm lượng đường trong máu, tốt cho tim...

[Xem chi tiết](#)

### Sử dụng v-for:

```
<template>
  <div class="container mt-5">
    <h1 class="text-center">Kiến thức sức khỏe cộng đồng</h1>
    <div class="row">
      <div class="col-sm-4 mb-4" v-for="(item, index) in items" :key="index">
        <div class="card">
          
          <div class="card-body">
            <h3 class="card-title">{{ item.title }}</h3>
            <p class="card-text">{{ item.content }}</p>
            <button class="btn btn-info">Xem chi tiết</button>
          </div>
        </div>
      </div>
    </div>
  </div>
</template>
```

### Bài 4 (3 điểm): Xây dựng các chức năng CRUD cho quản lý học sinh với các trường Họ tên, Điểm, Ngày sinh. Yêu cầu:

- ✚ Tạo Form: Phần này cho phép thêm hoặc sửa thông tin học sinh. Khi nhấn nút "Thêm", dữ liệu mới sẽ được thêm vào danh sách. Nếu đang trong chế độ chỉnh sửa, nút "Cập nhật" (thay cho nút Thêm) sẽ hiển thị và học sinh tương ứng sẽ được cập nhật.
- ✚ Danh sách: Danh sách hiển thị các học sinh đã được thêm vào. Bạn có thể chỉnh sửa hoặc xóa từng học sinh từ danh sách này.

✚ Có thể sử dụng sự kiện **@submit.prevent** thay vì **@click** ngăn chặn hành vi tải lại trang web khi người dùng bấm 'Thêm'

### Thêm học sinh

Họ tên:

Điểm:

Ngày sinh:

Thêm

### Danh sách học sinh

Họ và tên	Điểm	Ngày sinh		
Nguyễn Chí Hùng	8	2006-01-01	Sửa	Xóa
Phạm Thị Lan	9	2006-05-15	Sửa	Xóa

### Hướng dẫn:

1. Tạo **Bai4.vue** gồm 2 phần trong template

Phần **Form** để thêm hoặc sửa thông tin học sinh:

```

<form @submit.prevent="submitForm" class="col-sm-4">
  <h3>Thêm học sinh</h3>
  <div class="mb-3 mt-3">
    <label for="name">Họ tên:</label>
    <input type="text" class="form-control" v-model="student.name" id="name" required />
  </div>

  <div class="mb-3">
    <label for="score">Điểm:</label>
    <input type="number" max="10" min="0" class="form-control" v-model="student.score" id="score" required />
  </div>

  <div class="mb-3">
    <label for="dob">Ngày sinh:</label>
    <input type="date" class="form-control" v-model="student.dob" id="dob" required />
  </div>

  <button type="submit" class="btn btn-success">{{ isEditing ? 'Cập nhật' : 'Thêm' }}</button>
</form>

```

Phần **Danh sách** để hiển thị các học sinh đã được thêm vào, kèm các nút Sửa Xóa:

```
<h3>Danh sách học sinh</h3>
<table class="table table-hover">
  <thead>
    <tr>
      <th>Họ và tên</th>
      <th>Điểm</th>
      <th>Ngày sinh</th>
      <th></th>
      <th></th>
    </tr>
  </thead>
  <tbody>
    <tr v-for="(stu, index) in students" :key="index">
      <td>{{ stu.name }}</td>
      <td>{{ stu.score }}</td>
      <td>{{ stu.dob }}</td>
      <td>
        <button class="btn btn-warning" @click="editStudent(index)">Sửa</button>
      </td>
      <td>
        <button class="btn btn-danger" @click="deleteStudent(index)">Xóa</button>
      </td>
    </tr>
  </tbody>
</table>
```

## 2. Xử lý dữ liệu

```
<script setup>
import { ref } from 'vue'

// Dữ liệu mẫu ban đầu
const students = ref([
  { name: 'Nguyễn Chí Hùng', score: 8, dob: '2006-01-01' },
  { name: 'Phạm Thị Lan', score: 9, dob: '2006-05-15' }
])

// Biến quản lý trạng thái form
const student = ref({
  name: '',
  score: null,
  dob: ''
})

let isEditing = ref(false)
let editingIndex = ref(null)
```

```
function submitForm() {
  if (isEditing.value) {
    // Cập nhật thông tin học sinh
    // { ...student.value } sẽ tạo một bản sao mới của đối tượng student.value
    students.value[editingIndex.value] = { ...student.value }
    isEditing.value = false
    editingIndex.value = null
  } else {
    // Thêm học sinh mới
    students.value.push({ ...student.value })
  }
  resetForm()
}

function editStudent(index) {
  student.value = { ...students.value[index] }
  isEditing.value = true
  editingIndex.value = index
}

function deleteStudent(index) {
  students.value.splice(index, 1)
}

function resetForm() {
  student.value = {
    name: '',
    score: null,
    dob: ''
  }
}
</script>
```

**\*\* Giảng viên có thể cho thêm 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 ---