TRƯỜNG ĐẠI HỌC HỌC VĂN LANG

**KHOA CÔNG NGHỆ THÔNG TIN**

A red and black shield with white logo

Description automatically generated

**TÀI LIỆU THỰC HÀNH**

**SPRING BOOT**

**CHƯƠNG 03: FORM DỮ LIỆU SPRING BOOT**

Giảng viên biên soạn: Ths. Nguyễn Minh Tân

2024

**BUỔI 3: FORM DỮ LIỆU SPRING BOOT**

1. **MỤC TIÊU**

* Khởi tạo trang Web đầu tiên
* Khởi tạo trang web hiển thị thông tin sinh viên
* Tạo Templates Thymeleaf
* Spring Boot Validation
* Controller
* Bootstrap 5.3.0

1. **NỘI DUNG THỰC HÀNH:**

**Khởi tạo trang web hiển thị tên sinh viên**

Xây dựng form nhập thông tin sinh viên, có chứa các kiểm tra ràng buộc sau:

* Tên bắt buộc nhập.
* Tuổi phải là Số, có độ giá trị độ tuổi từ 18 đến 100.
* Khoa phải là Chữ

Sau khi nhập thông tin sinh viên. Trang sẽ trả về kết quả thông tin sinh viên.

Để xây dựng một form nhập thông tin sinh viên với các ràng buộc kiểm tra trên ta có thể sử dụng Spring Boot, Thymeleaf cùng với Spring Validation. Dưới đây là các bước để thực hiện điều này:

**Bước 1: Thêm Dependencies vào pom.xml**

Thêm **Dependency Spring Boot Validation (Spring Boot Starter Validation)** và **Jakarta Validation** vào trong file **pom.xml** và trong thẻ **dependencies**.

|  |
| --- |
| <dependencies>  *<!--Các Dependency khác-->*  *<!--Thêm phần Dependency ràng buộc-->*  <dependency>  <groupId>org.springframework.boot</groupId>  <artifactId>spring-boot-starter-validation</artifactId>  </dependency>  <dependency>  <groupId>jakarta.validation</groupId>  <artifactId>jakarta.validation-api</artifactId>  </dependency>  </dependencies> |

Sau khi thêm **Dependency** ta tiến thành chuột phải chọn **Maven ->** chọn **Reload project** để dự án tải và cập nhật **Spring Boot Validation** vào dự án.

A screenshot of a computer

Description automatically generated

**Bước 2: Tạo package model và xác định class SinhVien.java:**

Tạo package **model** tại **src/main/java/com.vanlang.demo**. Nhấn chuột phải vào **com.vanlang.demo**, chọn **New → Package** → đặt tên là **model. Trong package model** class **SinhVien.java** với các Validation phù hợp:

A screen shot of a computer

Description automatically generated

Code mẫu class SinhVien.java

A screen shot of a computer program

Description automatically generated

**Bước 3: Thêm Controller**

Tạo thêm **SinhVienController.java** để xử lý các yêu cầu GET và POST, bao gồm Validation.

A screen shot of a computer

Description automatically generated

Code mẫu class **SinhVienController.java**

A screenshot of a computer program

Description automatically generated

**Bước 4: Tạo Templates Thymeleaf**

Trong package **templates** tạo thêm directory **sinhvien**. Trong directory **sinhvien** tạo thêm các trang hiển thị html: ‘**form-sinhvien.html’**, ‘**result-sinhvien.html’**,

Code mẫu form nhập liệu ‘**form-sinhvien.html’**:

A screenshot of a computer program

Description automatically generated

Code mẫu trang hiển thị kết quả sinh viên ‘**result-sinhvien.html’**:

A screenshot of a computer program

Description automatically generated

**KẾT QUẢ THAM KHẢO:**

Trang nhập sinh viên Trang hiển thị kết quả sinh viên

A screenshot of a computer

Description automatically generatedA screenshot of a computer

Description automatically generated

**Yêu cầu bổ sung**

Tạo trang Layout dùng chung, sử dụng Bootstrap cho Layout và sửa lại tất cả các trang Thêm sinh viên, Hiển thị kết quả sinh viên.

**Hướng dẫn:**

Để tạo một trang layout dùng chung cho các trang khác trong ứng dụng Spring Boot và Thymeleaf, có thể sử dụng cơ chế layout của Thymeleaf. Điều này sẽ duy trì nhất quán và dễ dàng quản lý các thay đổi trên toàn bộ ứng dụng.

**Bước 1: Tạo Layout Chính**

Tạo một file mới tên là **layout.html** trong đường dẫn **src/main/resources/templates**

File này sẽ chứa cấu trúc **layout chính** cho các trang khác:

Code mẫu trang ‘**layout.html’**:

A screenshot of a computer program

Description automatically generated

Code tham khảo

|  |
| --- |
| <!DOCTYPE html> <html xmlns:th="http://www.thymeleaf.org" xmlns:layout="http://www.ultraq.net.nz/thymeleaf/layout"> <head>  <meta charset="UTF-8">  <title>Layout</title>  <link rel="stylesheet" href="https://cdn.jsdelivr.net/npm/bootstrap@5.3.0/dist/css/bootstrap.min.css"> </head> <body>  <nav class="navbar navbar-expand-lg navbar-dark bg-primary">  <div class="container-fluid"><a class="navbar-brand" href="#">Sinh Viên</a>  <div class="collapse navbar-collapse" id="navbarSupportedContent">  <ul class="navbar-nav me-auto mb-2 mb-lg-0">  <li class="nav-item"><a class="nav-link active" aria-current="page" href="/sinhvien">Nhập Sinh Viên</a>  </li>  </ul>  </div>  </div>  </nav>  <div class="container mt-4">  <section layout:fragment="content"> &lt;!&ndash; Content will be replaced by each specific page &ndash;&gt;  </section>  </div>  <script src="https://cdn.jsdelivr.net/npm/bootstrap@5.3.0/dist/js/bootstrap.bundle.min.js"></script> </body> </html> |

**Bước 2: Sử Dụng Layout Trong Các Trang Khác**

Form Sinh Viên ‘**form-sinhvien.html’**:

A screenshot of a computer program

Description automatically generated

Code tham khảo

|  |
| --- |
| <!DOCTYPE html> <html xmlns:th="http://www.thymeleaf.org" xmlns:layout="http://www.ultraq.net.nz/thymeleaf/layout"  layout:decorate="~{layout}"> <head><title>Form Sinh Viên</title></head> <body> <section layout:fragment="content"><h1>Form Sinh Viên</h1>  <form th:action="@{/sinhvien}" th:object="${sinhVien}" method="post" class="needs-validation" novalidate>  <div class="mb-3">  <label for="name" class="form-label">Tên</label>  <input type="text" class="form-control" id="name" placeholder="Enter your name" th:field="\*{ten}">  <p th:if="${#fields.hasErrors('ten')}" th:errors="\*{ten}" style="color: red;"></p>  </div>   <div class="mb-3">  <label for="age" class="form-label">Tuổi</label>  <input type="text" class="form-control" id="age" placeholder="Enter your age" th:field="\*{tuoi}">  <p th:if="${#fields.hasErrors('tuoi')}" th:errors="\*{tuoi}" style="color: red;"></p>  </div>   <div class="mb-3">  <label for="faculty" class="form-label">Tên</label>  <input type="text" class="form-control" id="faculty" placeholder="Enter your faculty" th:field="\*{khoa}">  <p th:if="${#fields.hasErrors('khoa')}" th:errors="\*{khoa}" style="color: red;"></p>  </div>   <p>  <button type="submit" class="btn btn-primary">Submit</button>  </p>   </form> </section> </body> </html> |

Trang Kết Quả ‘**result-sinhvien.html’**:

A screenshot of a computer program

Description automatically generated

Code tham khảo

|  |
| --- |
| <!DOCTYPE html> <html xmlns:th="http://www.thymeleaf.org" xmlns:layout="http://www.ultraq.net.nz/thymeleaf/layout"  layout:decorate="~{layout}"> <head>  <title>Kết quả</title>  <link rel="stylesheet" href="https://cdn.jsdelivr.net/npm/bootstrap@5.3.0/dist/css/bootstrap.min.css"> </head> <body>  <section layout:fragment="content">  <h1 th:text="${message}"></h1>  <table class="table table-striped table-hover">  <thead class="bg-light">  <tr>  <th>Tên</th>  <th>Tuổi</th>  <th>Khoa</th>  </tr>  </thead>  <tbody>  <tr>  <td th:text="${sinhVien.ten}">Tên mặc định</td>  <td th:text="${sinhVien.tuoi}">Tuổi mặc định</td>  <td th:text="${sinhVien.khoa}">Khoa mặc định</td>  </tr>  </tbody>  </table>  </section> </body> </html> |

**Bước 3: Cài đặt dependency Thymeleaf Layout Dialect vào pom.xml**

**Lưu ý:** Đảm bảo rằng đã cài đặt **thymeleaf-layout-dialect** trong dự án của bạn để sử dụng tính năng **layout** của **Thymeleaf**:

|  |
| --- |
| <dependency>  <groupId>nz.net.ultraq.thymeleaf</groupId>  <artifactId>thymeleaf-layout-dialect</artifactId>  </dependency> |

A screen shot of a computer code

Description automatically generated

Sau khi thêm **Dependency** ta tiến thành nhấp chuột phải → chọn **Maven** → chọn **Reload project** để dự án tải và cập nhật **Thymeleaf Layout Dialect** vào dự án.

**Kết quả tham khảo:**

* Trang form

A screenshot of a computer

Description automatically generated

* Trang result:

A blue and white text box

Description automatically generated