Tài liệu học Thymeleaf cơ bản (Phiên bản v6)

# I. Tóm lược cú pháp thường gặp

|  |  |  |  |
| --- | --- | --- | --- |
| Chủ đề/Mục đích | Cú pháp | Giải thích | Lưu ý / Mẹo |
| Hiển thị nội dung | th:text="${...}" | Hiển thị giá trị bên trong văn bản | Thymeleaf tự escape, tránh XSS |
| Vòng lặp danh sách | th:each="item : ${list}" | Lặp qua tập hợp trong model | item không cần `${}` vì là biến mới |
| Hiển thị có điều kiện | th:if="${cond}" / th:unless="${cond}" | Hiển thị hoặc ẩn theo điều kiện | th:unless là phủ định th:if |
| Link động | th:href="@{/path/{id}(id=${val})}" | Gán URL động với tham số | Tự thêm context path |
| Ảnh động | th:src="@{/img/logo.png}" | Gán URL động cho ảnh | Tương tự th:href |
| Giá trị input | th:value="${val}" | Gán giá trị cho thuộc tính value | Dùng input readonly hoặc form |
| Form binding | th:object="${obj}" + th:field="\*{field}" | Binding form đến object trong model | \*{} sử dụng context của th:object |
| Thêm class điều kiện | th:classappend="${cond} ? 'c1' : 'c2'" | Append CSS class dựa trên cond | th:class thay cho class toàn phần |
| Biến tạm | th:with="var=${expr}" | Khai báo biến tạm trong template | Chỉ dùng trong scope chứa th:with |
| Tách layout | th:replace="~{layout/master :: frag}" | Chèn fragment từ layout | DRY, tái sử dụng layout |
| Include fragment | th:include="~{common/footer}" | Nhúng fragment | Không thay thế container |

# II. Ví dụ minh họa chi tiết (Controller + HTML + Giải thích)

## 1. th:text - Hiển thị nội dung

Controller:

@GetMapping("/hello")

public String hello(Model model) {

model.addAttribute("ten", "Nguyễn Văn A");

return "hello";

}

HTML:

<span th:text="${ten}">Giá trị mặc định</span>

## 2. th:each - Vòng lặp danh sách

Controller:

@GetMapping("/products")

public String showProducts(Model model) {

List<Product> list = List.of(...);

model.addAttribute("productList", list);

return "product-list";

}

HTML:

<tr th:each="p : ${productList}">...

<td th:text="${p.name}"></td>

</tr>

## 3. th:if / th:unless - Hiển thị có điều kiện

Controller:

@GetMapping("/status")

public String status(Model model) {

model.addAttribute("user", u);

return "status";

}

HTML:

<p th:if="${user.active}">Active</p>

<p th:unless="${user.active}">Inactive</p>

## 4. th:href - Link động

Controller:

@GetMapping("/profile/{id}")

public String prof(@PathVariable Long id, Model model) {

model.addAttribute("u", userService.findById(id));

return "profile";

}

HTML:

<a th:href="@{/user/{id}(id=${u.id})}">Profile</a>

Giải thích: `{id}` đóng vai trí placeholder, `(id=${u.id})` gán giá trị; @{} thêm context path.

## 5. th:value - Gán giá trị input

Controller:

@GetMapping("/form")

public String form(Model model) {

model.addAttribute("user", new User());

return "form";

}

HTML:

<input type="text" th:value="${user.name}"/>

## 6. th:object + th:field - Form binding

Controller:

@GetMapping("/edit")

public String editForm(Model model) {

model.addAttribute("user", userService.get());

return "edit";

}

HTML:

<form th:object="${user}" th:action="@{/save}" method="post">

<input th:field="\*{name}"/>

</form>

Giải thích: `th:object` đặt context object; `\*{name}` tương đương `${user.name}`.

## 7. th:classappend - Thêm class điều kiện

Controller:

@GetMapping("/dash")

public String dash(Model model){...}

HTML:

<div th:classappend="${user.admin} ? 'admin' : 'user'"></div>

## 8. th:with - Biến tạm

Controller:

@GetMapping("/welcome")

...

HTML:

<div th:with="fullName=${user.firstName + ' ' + user.lastName}">...

<p th:text="${fullName}"></p>

</div>

## 9. th:replace - Tách layout

Controller:

@GetMapping("/home")

public String home(){return "home";}

HTML:

<div th:replace="~{layout/master :: header}"></div>

Sample layout/master.html:  
<html th:fragment="layout">  
 <head th:fragment="header">...</head>  
</html>

## 10. th:include - Include fragment

Controller:

@GetMapping("/page")

public String page(){return "page";}

HTML:

<footer th:include="~{common/footer}"></footer>

Sample common/footer.html:  
<footer th:fragment="footer">...</footer>

# III. Phụ lục: Layout & Fragment mẫu

layout/master.html:

<!DOCTYPE html>

<html xmlns:th="http://www.thymeleaf.org" th:fragment="layout">

<head>

<title th:insert="~{::title}">Default Title</title>

</head>

<body>

<header th:fragment="header">Header content</header>

<main th:fragment="content">Main content</main>

<footer th:fragment="footer">Footer content</footer>

</body>

</html>

common/footer.html:

<div th:fragment="footer">

<p>&copy; 2025 Công ty ABC</p>

</div>