



THYMELEAF

ĐÀO TẠO CHUYÊN GIA LẬP TRÌNH

CYBERSOFT.EDU.VN



MYC</>DER

timviec*it*.com

Nội dung

- ☐ Tổng quan về Thymeleaf.
- ☐ Cấu hình Thymeleaf.
- ☐ Binding dữ liệu trong Thymeleaf.
- ☐ Một số thuộc tính của Thymeleaf.
- ☐ Dàn layout sử dụng Thymeleaf Layout Dialect.

CYBERSOFT

ĐÀO TẠO CHUYÊN GIA LẬP TRÌNH

Tạo controller - view

HomeController

```
@Controller
public class HomeController {
    @GetMapping("/")
    public String index() {
        return "home/index";
    }
}
```

File **html** phải khai báo
xmlns:th=<http://www.thymeleaf.org>
để sử dụng thymeleaf nếu không khi
chạy lên chương trình sẽ báo lỗi.

index.html

```
<!DOCTYPE html>
<html xmlns:th="http://www.thymeleaf.org">
<head>
<meta charset="UTF-8">
<title>home page</title>
</head>
<body>
    <h1>HOME PAGE</h1>
</body>
</html>
```

Thymeleaf



- ❑ **Thymeleaf** là một Java Template Engine có nhiệm vụ xử lý và generate ra các file HTML, XML, v.v...
- ❑ **Thymeleaf** kết hợp **dữ liệu** và **template** + **quy tắc** để sinh ra file HTML chứa đầy đủ thông tin.
- ❑ **Thymeleaf** có thể làm việc với cả hai môi trường Web và môi trường không phải Web.
- ❑ Cần khai báo thuộc tính **xmlns:th=**<http://www.thymeleaf.org> cho trang html nếu muốn sử dụng Thymeleaf.

Thư viện sử dụng

porm.xml

```
<dependency>
  <groupId>org.thymeleaf</groupId>
  <artifactId>thymeleaf</artifactId>
  <version>3.0.11.RELEASE</version>
</dependency>

<dependency>
  <groupId>org.thymeleaf</groupId>
  <artifactId>thymeleaf-spring5</artifactId>
  <version>3.0.11.RELEASE</version>
</dependency>
```

Dependency
cho Thymeleaf.

Dependency hỗ
trợ cho Spring 5.

Cấu hình Thymeleaf

❑ Các bước cấu hình:

- ✓ **Bước 1:** Tải thư viện Thymeleaf cho dự án.
- ✓ **Bước 2:** Tạo bean **TemplateResolver** thiết lập các thông tin prefix, suffix để chỉ dẫn folder chứa các page view bên trong thư mục webapp.
- ✓ **Bước 3:** Tạo bean **SpringTemplateEngine** chỉ định loại TemplateEngine sẽ sử dụng cho dự án.
- ✓ **Bước 4:** Tạo bean **ViewResolver** thực hiện mapping giữa các file view với view name trả về từ controller.

TemplateResolver

- ❑ Bean **TemplateResolver** thực hiện thiết lập các thông tin prefix, suffix để chỉ dẫn folder chứa các page view bên trong thư mục webapp.

```
@Bean
public ITemplateResolver templateResolver() {
    SpringResourceTemplateResolver templateResolver = new SpringResourceTemplateResolver();
    // Khai báo thư mục chứa các view file
    templateResolver.setPrefix( "/WEB-INF/templates/" );
    // Khai báo phần mở rộng của view file
    templateResolver.setSuffix( ".html" );
    // Khai báo mã nguồn html đích => Khi chạy chương trình sẽ render ra mã nguồn HTML5
    templateResolver.setTemplateMode( "HTML5" );
    /*
     * Thuộc tính cacheable đánh dấu cache cho view,
     * nếu bạn để là true thì hệ thống sẽ lấy file view ban đầu và lưu cache,
     * mọi thay đổi của file view sẽ không được load lên mà phải restart lại server.
     */
    templateResolver.setCacheable(false);
    return templateResolver;
}
```

SpringTemplateEngine

- ❑ Bean **SpringTemplateEngine** chỉ định loại TemplateEngine sẽ sử dụng cho dự án.

```
@Bean
public SpringTemplateEngine templateEngine() {
    SpringTemplateEngine templateEngine = new SpringTemplateEngine();
    templateEngine.setTemplateResolver( templateResolver() );
    return templateEngine;
}
```


ViewResolver

- ❑ **ThymeleafViewResolver** sẽ thực hiện mapping giữa các file view với view name trả về từ controller.
- ❑ **Ví dụ:** Trong controller trả về “index” thì nó sẽ hiểu là trả về file **index.html** trong folder **/WEB-INF/templates**.

```
@Bean
public ViewResolver viewResolver() {
    ThymeleafViewResolver viewResolver = new ThymeleafViewResolver();
    viewResolver.setTemplateEngine( templateEngine() );
    return viewResolver;
}
```

Biểu thức trong Thymeleaf

❑ Có 5 loại biểu thức cơ bản trong Thymeleaf:

`$ {...}`: Biểu thức lấy giá trị của attribute.

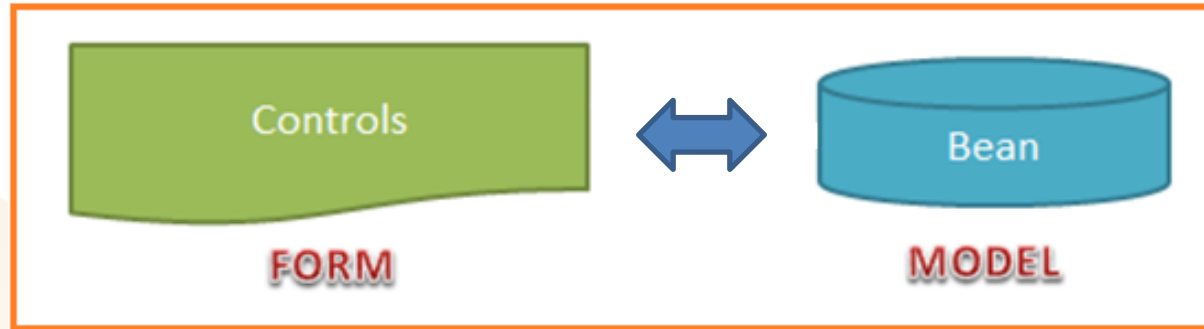
`* {...}`: Biểu thức lấy giá trị của thuộc tính trong đối tượng.

`# {...}`: Biểu thức lấy message (Sử dụng cho đa ngôn ngữ).

`@ {...}`: Lấy đường dẫn URL dựa theo context của server.

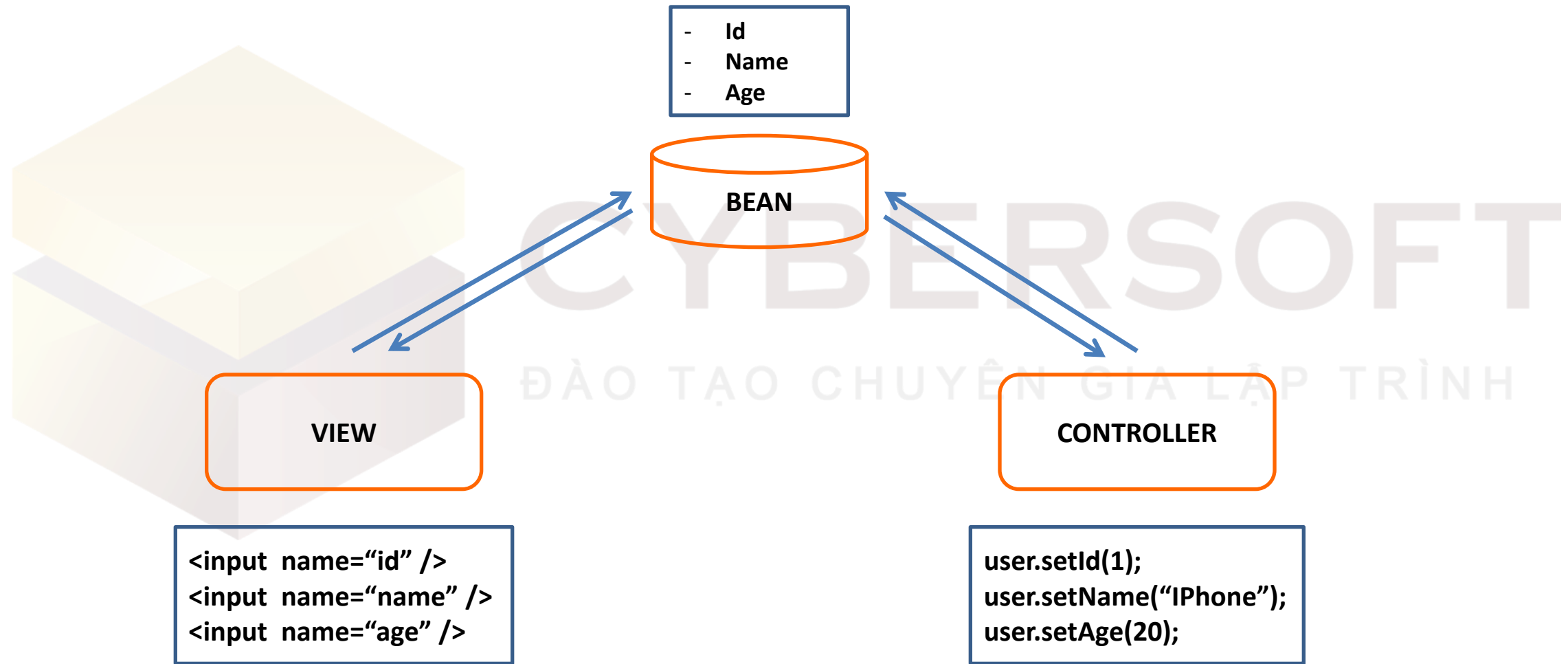
`~ {...}`: Biểu thức phân đoạn.

Data Binding là gì?



- ✓ **Data binding** (ràng buộc dữ liệu) là kỹ thuật dùng để **tạo gắn kết dữ liệu giữa tầng view và dữ liệu trong các bean ở tầng controller**.
- ✓ Khi thay đổi dữ liệu trong bean thì dữ liệu trên các điều khiển cũng thay đổi theo.
- ✓ Ràng buộc dữ liệu có thể là **1 chiều** hoặc **2 chiều**:
 - ✓ **Chiều lên**: chuyển dữ liệu từ các điều khiển vào các thuộc tính của bean.
 - ✓ **Chiều về**: hiển thị dữ liệu từ các thuộc tính của bean lên các điều khiển của form.

Data Binding

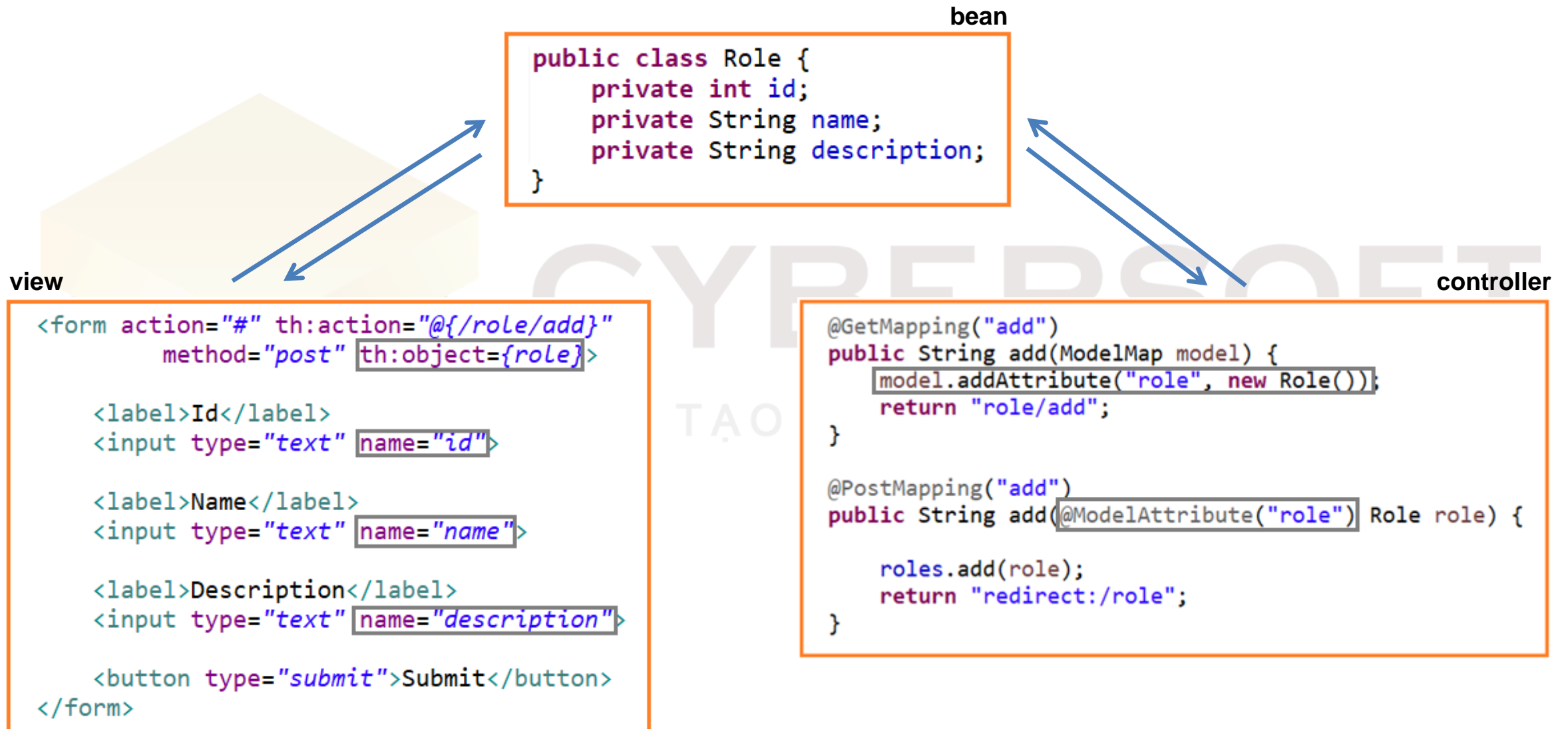


Form trong Thymeleaf

- ❑ **th:object** được dùng để chỉ định đối tượng cần truy xuất.
- ❑ **th:field** được dùng để ràng buộc dữ liệu với các thuộc tính trong bean.
- ❑ **th:errors** được dùng để hiển thị lỗi validation cho các field.
- ❑ **th:action** được dùng để khai báo url.

```
<form method="post" th:object="${role}" th:action="@{/admin/role/add}" action="#">
  <div class="form-group">
    <label>Tên quyền</label>
    <input type="text" th:field="*{name}" class="form-control" />
    <span class="text-danger" th:if="${#fields.hasErrors('name')}" th:errors="*{name}"></span>
  </div>
  <div class="form-group">
    <button type="submit" class="btn btn-success">Lưu lại</button>
    <a class="btn btn-secondary" th:href="@{/admin/role}">Quay lại</a>
  </div>
</form>
```

Form binding



List binding trong Thymeleaf

- ❑ Thuộc tính **th:if** dùng kiểm tra điều kiện.
- ❑ Thuộc tính **th:text** dùng để in đoạn văn bản.
- ❑ Thuộc tính **th:each** dùng để lặp một danh sách.

```
<tr th:if="${products.empty}">
    <td colspan="3"> No Products Available</td>
</tr>
<tr th:each="item : ${products}">
    <td th:text="${itemStat.index + 1}">1</td>
    <td th:text="${item.name}">Onions</td>
    <td th:text="${item.price}">2.41</td>
</tr>
```

Url trong Thymeleaf

❑ Truyền tham số

```
<a th:href="@{/order/details(id=3)}">Details</a>
```



```
<a th:href="/order/details?id=3">Details</a>
```

```
<a th:href="@{/order/details(id=3,action='show_all')}">Details</a>
```



```
<a href="/order/details?id=3&action=show_all">Details</a>
```

```
<a th:href="@{/product/edit/{id}(id=3)}">Edit</a>
```



```
<a th:href="/product/edit/3)">Edit</a>
```


Url trong Thymeleaf

- ❑ Sử dụng thuộc tính **th:href** hoặc **th:src** cho các url trong trang web.
- ❑ Phải có biểu thức url liên kết **@{}**.

Link href

```
<link rel="stylesheet" th:href="@{/css/font-awesome.min.css}">  
<link rel="stylesheet" th:href="@{/css/bootstrap.min.css}">
```

Link src

```
<script th:src="@{/js/jquery.min.js}"></script>  
<script th:src="@{/js/popper.min.js}"></script>  
<script th:src="@{/js/bootstrap.min.js}"></script>
```

Biểu thức điều kiện

- ❑ Biểu thức **th:if** kiểm tra điều kiện true.
- ❑ Biểu thức **th:unless** kiểm tra điều kiện false.

```
<td>
  <span th:if="{teacher.gender == 'F'}">Female</span>
  <span th:unless="{teacher.gender == 'F'}">Male</span>
</td>
```

- ❑ Biểu thức **switch case**.

```
<td th:switch="{user.role}">
  <span th:case="'admin'">administrator</span>
  <span th:case="'manager'">manager</span>
  <span th:case="'member'">member</span>
  <span th:case="*">NO ROLE YET!</span>
</td>
```

FT

ĐÀO TẠO CHUYÊN GIA LẬP TRÌNH

th:class

- ❑ **th:class** được dùng để thay thế class có sẵn bằng class mới trong thẻ html.

```
<p class="user-class" th:class="${isAdmin} ? admin-class : user-class">  
    Some Text 1  
</p>
```



```
<p class="admin-class">  
    Some Text 1  
</p>
```

- ❑ **th:classappend** được dùng để thêm class vào thẻ html.

```
<p class="base-class" th:classappend="${isAdmin} ? admin-class : user-class">  
    Some Text 1  
</p>
```



```
<p class="base-class admin-class">  
    Some Text 1  
</p>
```

th:style

- ❑ **th:style** được dùng để thay thế style có sẵn bằng class mới trong thẻ html.

```
<p style="color: blue;" th:style = "${isAdmin} ? 'color: blue' : 'color: black' ">  
    Some Text 1  
</p>
```



```
<p style="color: blue">  
    Some Text 1  
</p>
```

- ❑ **th:styleappend** được dùng để thêm style vào thẻ html.

```
<p style="background: #eee;" th:styleappend = "${isAdmin} ? 'color: blue' : 'color: black' ">  
    Some Text 1  
</p>
```



```
<p style="background: #eee; color: blue">  
    Some Text 1  
</p>
```

Bài tập

❑ Sử dụng Spring MVC và Thymeleaf thực hành CRUD cho danh sách quyền với thông tin mô tả như sau:

❖ **Role:**

- **Id:** Mã quyền.
- **Name:** Tên quyền.
- **Description:** Mô tả thông tin quyền.

CYBERSOFT

ĐÀO TẠO CHUYÊN GIA LẬP TRÌNH

Thymeleaf layout

❑ Các bước cấu hình:

- ✓ **Bước 1:** Tải thư viện **Thymeleaf Layout Dialect**.
- ✓ **Bước 2:** Tạo bean **LayoutDialect**.
- ✓ **Bước 3:** Cấu hình **SpringTemplateEngine** sử dụng bean vừa tạo để dựng layout.

CYBERSOFT
ĐÀO TẠO CHUYÊN GIA LẬP TRÌNH

Thymeleaf layout

```
<dependency>
  <groupId>org.thymeleaf</groupId>
  <artifactId>thymeleaf</artifactId>
  <version>3.0.11.RELEASE</version>
</dependency>
<dependency>
  <groupId>org.thymeleaf</groupId>
  <artifactId>thymeleaf-spring5</artifactId>
  <version>3.0.11.RELEASE</version>
</dependency>
<dependency>
  <groupId>nz.net.ultraq.thymeleaf</groupId>
  <artifactId>thymeleaf-layout-dialect</artifactId>
  <version>2.3.0</version>
</dependency>
```

FT
TRÌNH

Thymeleaf layout

```
@Bean
public LayoutDialect layoutDialect() {
    return new LayoutDialect();
}

@Bean
public SpringTemplateEngine templateEngine() {
    SpringTemplateEngine templateEngine = new SpringTemplateEngine();
    templateEngine.setTemplateResolver( templateResolver() );
    templateEngine.addDialect(layoutDialect());
    return templateEngine;
}
```


Fragment

- ❑ **Fragment** (mảnh) là một phần nhỏ trong một template. Thymeleaf cho phép nhúng các fragment của template này vào một template khác.
- ❖ Các cách để nhúng một fragment vào trong template:
- ✓ Sử dụng **th:replace**: Tính năng này sẽ thay thế thẻ html bằng fragment.

```
<div class="child" xmlns:th="http://www.thymeleaf.org" th:fragment="demo">  
  <button class="btn btn-success">Click</button>  
</div>
```

Định nghĩa
fragment

Nhúng vào
template

```
<section class="parent">  
  <div th:replace="fragments/test :: demo"></div>  
</section>
```

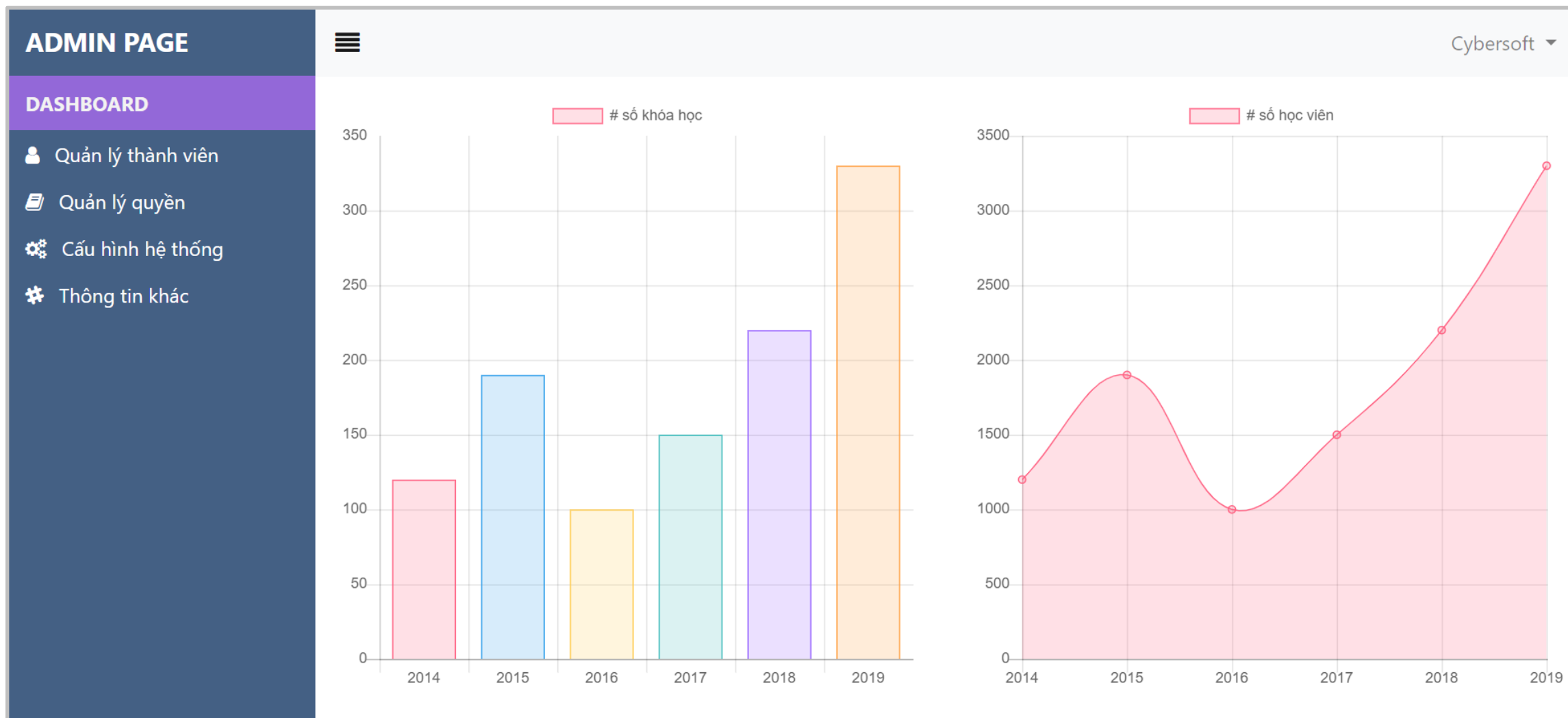
Kết quả

```
<section class="parent">  
  <div class="child">  
    <button class="btn btn-success">Click</button>  
  </div>  
</section>
```

❑ Dàn layout cho trang quản trị sử dụng thymeleaf.

❖ Các bước thực hiện:

- ✓ **Bước 1:** Tách các layout dùng chung thành các fragment (đặt trong thư mục templates/fragments).
- ✓ **Bước 2:** Tạo file default.html làm file layout chính (đặt trong thư mục template/layout).
- ✓ **Bước 3:** Nhúng các fragment vào default.html.
- ✓ **Bước 4:** Tạo các file còn lại (index.html, list.html, ...)



Cấu trúc thư mục



- ▼ 📁 src/main/resources
 - > 📁 static
 - ▼ 📁 templates
 - ▼ 📁 fragments
 - 📄 navbar.html
 - 📄 scripts.html
 - 📄 sidebar.html
 - 📄 styles.html
 - ▼ 📁 home
 - 📄 index.html
 - ▼ 📁 layout
 - 📄 default.html
 - ▼ 📁 user
 - 📄 add.html
 - 📄 edit.html
 - 📄 index.html
 - 🌿 application.properties
 - > 📁 src/test/java
 - > 📖 JRE System Library [JavaSE-1.8]

SOFT
IA LẬP TRÌNH

Style - Script

styles.html

```
<link rel="stylesheet" href="https://maxcdn.bootstrapcdn.com/font-awesome/4.7.0/css/font-awesome.min.css" />
<link rel="stylesheet" xmlns:th="http://www.thymeleaf.org" th:href="@{/css/bootstrap.min.css}" />
<link rel="stylesheet" xmlns:th="http://www.thymeleaf.org" th:href="@{/css/style.css}" />
```

scripts.html

```
<script xmlns:th="http://www.thymeleaf.org" th:src="@{/js/jquery.slim.min.js}"></script>
<script xmlns:th="http://www.thymeleaf.org" th:src="@{/js/popper.min.js}"></script>
<script xmlns:th="http://www.thymeleaf.org" th:src="@{/js/bootstrap.min.js}"></script>
```

Sidebar

sidebar.html

```
<div id="side-bar" xmlns:th="http://www.thymeleaf.org" th:fragment="sidebar">
  <div class="logo">ADMIN PAGE</div>
  <ul class="list-group rounded-0">
    <li class="dashboard">DASHBOARD</li>
    <li><a href="#"> <i class="fa fa-user mr-2"></i>
      Quản lý thành viên
    </a></li>
    <li><a href="#"> <i class="fa fa-book mr-2"></i>
      Quản lý quyền
    </a></li>
    <li><a href="#"> <i class="fa fa-cogs mr-2"></i>
      Cấu hình hệ thống
    </a></li>
    <li><a href="#"> <i class="fa fa-slack mr-2"></i>
      Thông tin khác
    </a></li>
  </ul>
</div>
```

Navbar

```
<nav class="navbar navbar-expand-sm navbar-light bg-light w-100"
  xmlns:th="http://www.thymeleaf.org" th:fragment="navbar">
  <a class="navbar-brand" href="#"><i class="fa fa-align-justify"></i></a>
  <button class="navbar-toggler d-lg-none" type="button"
    data-toggle="collapse" data-target="#collapsibleNavId"
    aria-controls="collapsibleNavId" aria-expanded="false"
    aria-label="Toggle navigation"></button>
  <div class="collapse navbar-collapse" id="collapsibleNavId">
    <ul class="navbar-nav ml-auto mt-2 mt-lg-0">
      <li class="nav-item dropdown"><a
        class="nav-link dropdown-toggle" href="#" id="dropdownId"
        data-toggle="dropdown" aria-haspopup="true" aria-expanded="false">
        Cybersoft </a>
      <div class="dropdown-menu dropdown-menu-right"
        aria-labelledby="dropdownId">
        <a class="dropdown-item" href="">Thông tin cá nhân</a> <a
          class="dropdown-item" href="#">Cài đặt</a> <a
            class="dropdown-item" href="#">Thoát</a>
      </div></li>
    </ul>
  </div>
</nav>
```

Template

default.html

```
<!DOCTYPE html>
<html xmlns:th="http://www.thymeleaf.org"
      xmlns:layout="http://www.ultraq.net.nz/thymeleaf/layout">
<head>
  <title>Trang chủ</title>
  <!-- Required meta tags -->
  <meta charset="utf-8" />
  <meta name="viewport" content="width=device-width, initial-scale=1, shrink-to-fit=no" />
  <th:block th:replace="fragments/styles" />
</head>

<body>

  <div class="d-flex justify-content-between">
    <!-- SIDE BAR -->
    <div th:replace="fragments/sidebar :: sidebar"></div>

    <div id="admin-wrapper">
      <!-- HEADER -->
      <div th:replace="fragments/navbar :: navbar"></div>

      <!-- CONTENT -->
      <section id="admin-content" class="p-3" layout:fragment="content">

      </section>
    </div>
  </div>

  <th:block th:replace="fragments/scripts" />
</body>

</html>
```


Trang chủ

home/index.html

```
<!DOCTYPE html>
<html xmlns:th="http://www.thymeleaf.org"
      xmlns:layout="http://www.ultraq.net.nz/web/thymeleaf/layout"
      layout:decorator="layout/default">
  <head>
    <title>Trang chủ nè</title>
  </head>
  <body>
    <section layout:fragment="content">
      <h2>TRANG CHỦ</h2>
    </section>
  </body>
</html>
```

Tổng kết

- ✓ Tổng quan về Thymeleaf.
- ✓ Cấu hình Thymeleaf.
- ✓ Binding dữ liệu trong Thymeleaf.
- ✓ Một số thuộc tính của Thymeleaf.
- ✓ Dàn layout sử dụng Thymeleaf Layout Dialect.

CYBERSOFT
ĐÀO TẠO CHUYÊN GIA LẬP TRÌNH