

Tutorial 6 – Thymeleaf – Presentation Layer

Penggunaan For Looping

```
1 <!DOCTYPE html>
2 <html xmlns:th="http://www.thymeleaf.org">
3 <head>
4 <title>View All Students</title>
5 </head>
6 <body>
7 <h1>All Students</h1>
8
9 <div th:each="student, iterationStatus: ${students}" th:class="${iterationStatus.odd}?'odd'">
10 <h3 th:text="No. ' + ${iterationStatus.count}>No. 1</h3>
11 <h3 th:text="NPM = ' + ${student.npm}>Student NPM</h3>
12 <h3 th:text="Name = ' + ${student.name}>Student Name</h3>
13 <h3 th:text="GPA = ' + ${student.gpa}>Student GPA</h3>
14 <h3>Kuliah yang diambil</h3>
15 <ul th:each="course, iterationStatus: ${student.courses}">
16 <li th:text="${course.name} + ' - ' + ${course.credits} + ' sks'">
17 Nama kuliah-X SKS
18 </li>
19 </ul>
20 <a th:href="/student/delete/" + ${student.npm}> Delete Data</a><br/>
21 <a th:href="/student/update/" + ${student.npm}> Update Data</a><br/>
22 <hr/>
23 </div>
24 </body>
25 </html>
```

Pertanyaan 1:

Apakah value yang dihasilkan dari `${iterationStatus.odd}` ?

➔ Ketika dilihat dari Inspect element, class yang keluar memiliki value odd atau ganjil

Penggunaan Conditional Expression pada Thymeleaf

1. Tambahkan code baris berikut ini pada file **viewall.html**

```
<h3 th:unless="${student.gpa}>3.49">Sangat Memuaskan!</h3>
<h3>Kuliah yang diambil</h3>
```

2. Coba run aplikasi Anda, lalu akses **localhost:8080/student/viewall** . Tampilan akan seperti gambar di bawah ini.

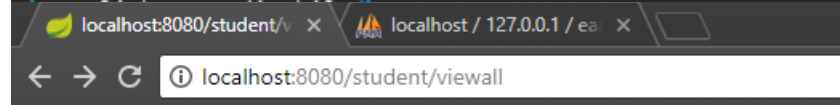
All Students		
No. 1	No. 2	No. 3
NPM = 123	NPM = 124	NPM = 125
Name = Fajar	Name = Ammar	Name = Adit
GPA = 3.3	GPA = 3.25	GPA = 3.15
Sangat Memuaskan!	Sangat Memuaskan!	Sangat Memuaskan!
Kuliah yang diambil	Kuliah yang diambil	Kuliah yang diambil
<ul style="list-style-type: none">MPKT-6 sks	<ul style="list-style-type: none">PSP-4 sksSDA-3 sks	
Delete Data	Delete Data	Delete Data
Update Data	Update Data	Update Data

Pertanyaan 2 : Mengapa *condition* di dalam **th:unless** sama dengan di dalam **th:if** ?
Jelaskan jawaban Anda.

➔ Karena unless dapat diibaratkan fungsi 'else' dalam pemrograman seperti biasa. Atau menyerupai negasi dari if (seperti if not), sehingga kondisi yang diberikan harus sama.

Pertanyaan 3: Coba ubah *condition* pada `th:unless=${student.gpa}>=3.49}` menjadi `th:unless=${student.gpa}<=3.48}`, lalu run kembali aplikasi Anda dan mengakses halaman yang sama. Apakah terjadi *error*? Jika iya, kenapa? Jelaskan alasan Anda. Jika tidak *error*, apakah hasilnya sesuai?

```
<h3 th:if="${student.gpa}>=3.49}">Cum Laude!</h3>
<h3 th:unless="${student.gpa}<=3.48}">Sangat Memuaskan!</h3>
```



Whitelabel Error Page

This application has no explicit mapping for /error, so you are seeing this as a fallback.

Sun Oct 15 17:02:28 ICT 2017

There was an unexpected error (type=Internal Server Error, status=500).

Exception parsing document: template="viewall", line 15 - column 38

➔ Ya, terjadi error. Error yang terjadi karena keluaran akan sama dengan kondisi *if* yang ada pada baris sebelumnya, menunjukkan conditionnya sama sehingga terjadi error.

Pertanyaan 4: Tuliskan cara lain Anda dengan penulisan *conditional expression* yang berbeda untuk mendapatkan hasil yang sama seperti gambar di atas. Hint: Gunakan *ternary operator*.

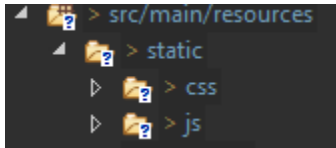
➔

```
<!-- Cara lain penulisan conditional expression utk hasil yg sama dengan 2 baris diatas-->
<h3 th:text="${student.gpa}>=3.49 ? 'Cum Laude!' : 'Sangat Memuaskan!'"></h3>
```

All Students		
No. 1	No. 2	
NPM = 123	NPM = 124	
Name = Fajar	Name = Ammar	No. 3
GPA = 3.3	GPA = 3.25	NPM = 125
Sangat Memuaskan!	Sangat Memuaskan!	Name = Adit
Kuliah yang diambil	Kuliah yang diambil	GPA = 3.15
<ul style="list-style-type: none">MPKT-6 sks	<ul style="list-style-type: none">PSP-4 sksSDA-3 sks	Sangat Memuaskan!
Delete Data	Delete Data	Kuliah yang diambil
Update Data	Update Data	Delete Data
>	>	Update Data
>	>	>

Static file

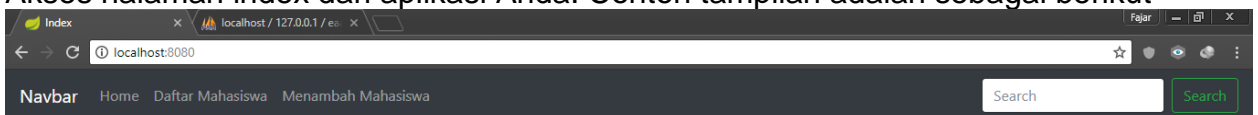
1. Unduh bootstrap pada <http://getbootstrap.com/docs/4.0/getting-started/download/>
2. *Extract folder* zip yang Anda unduh ke dalam folder **static** yang terdapat pada folder **src/main/resources**



3. Buat halaman **index.html** menjadi *source code* dibawah ini

```
9 <!DOCTYPE html>
10 <html xmlns:th = "http://www.thymeleaf.org">
11 <head>
12 <title> Index </title>
13 <link rel = "stylesheet" href = "/css/bootstrap.min.css" />
14 </head>
15 <body>
16 <nav class = "navbar navbar-expand-lg navbar-dark bg-dark">
17 <a class = "navbar-brand" href = "#" >Navbar </a>
18 <button class="navbar-toggler" type="button" data-toggle="collapse" data-target = "#navbarSupportedContent" aria-controls = "navbarSupportedContent" >
19 <span class = "navbar-toggler-icon" ></span>
20 </button>
21
22 <div class = "collapse navbar-collapse" id = "navbarSupportedContent">
23 <ul class = "navbar-nav mr-auto">
24 <li class = "nav-item" >
25 <a class = "nav-link" href = "/" > Home </a>
26 </li>
27 <li class = "nav-item" >
28 <a class="nav-link" href = "/student/viewall" > Daftar Mahasiswa </a>
29 </li>
30 <li class = "nav-item" >
31 <a class = "nav-link" href = "/student/add" > Menambah Mahasiswa </a>
32 </li>
33 </ul>
34
35 <form class = "form-inline my-2 my-lg-0">
36 <input class="form-control mr-sm-2"
37 type="text" placeholder="Search"
38 aria-label = "Search" />
39 <button class = "btn btn-outline-success my-2 my-sm-0" type = "submit" >
40 Search
41 </button>
42 </form>
43 </div>
44 </nav>
45 <center>
46 <h3> Welcome To Tutorial 6 APAP </h3>
47 </center>
```

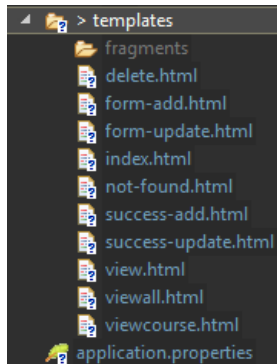
4. Akses halaman index dari aplikasi Anda. Contoh tampilan adalah sebagai berikut



Welcome To Tutorial 6 APAP

Fragment

1. Buat sebuah *folder* dalam *folder* **templates** dengan nama **fragments**



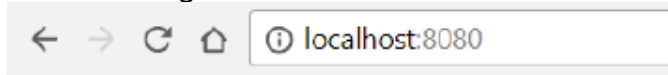
2. Buat sebuah halaman HTML bernama **fragment.html** yang berisi *source code* seperti dibawah ini. File HTML tersebut dibuat pada folder **fragments** yang baru Anda buat sebelumnya

```
1 <!DOCTYPE html>
2 <html xmlns:th = "http://www.thymeleaf.org">
3 <head>
4 <title> Fragment </title>
5 </head>
6
7 <body>
8 <div th:fragment = "header">
9 <ul>
10 <li><a href = "/" > Home </a></li>
11 <li><a href = "/student/viewall" > Daftar Mahasiswa </a></li>
12 <li><a href = "/student/add" > Menambah Mahasiswa </a></li>
13 </ul>
14 </div>
15
16 <div th:fragment = "footer">
17 <hr/>
18 <h4> Mata Kuliah APAP </h4>
19 </div>
20 </body>
21 </html>
```

3. Pada *source code* diatas, **th:fragment** merepresentasikan nama *fragment* dari bagian yang Anda ingin jadikan sebuah *fragment*
4. Buat halaman **index.html** menjadi *source code* dibawah ini

```
<!-- index.html setelah ditambah fragment di fragment.html -->
<!DOCTYPE html>
<html xmlns:th = "http://www.thymeleaf.org">
<head>
<title> Index </title>
</head>
<body>
<div th:replace = "fragments/fragment :: header" ></div>
<h2> Welcome To Tutorial 6 APAP </h2>
<div th:replace = "fragments/fragment :: footer" ></div>
</body>
</html>
```

5. Coba jalankan halaman index yang Anda buat. Contoh tampilan yang keluar adalah sebagai berikut



- [Home](#)
- [Daftar Mahasiswa](#)
- [Menambah Mahasiswa](#)

Welcome To Tutorial 6 APAP

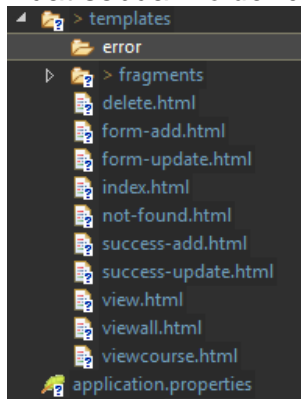
Mata Kuliah APAP

Pertanyaan 5 : Apa yang dimaksud dengan `th:replace="fragments/fragment :: header"` dan `th:replace="fragments/fragment :: footer"` pada file `index.html` yang Anda buat?

- ➔ Bagian div dari `fragment.html` yang memiliki nama `header` dan `footer` akan diimport ke `index.html`

Error 404 Not Found Handler

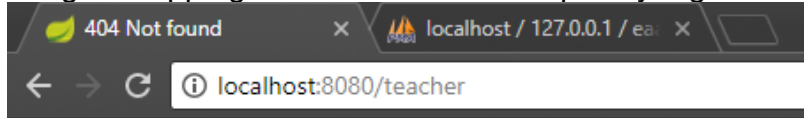
1. Buat sebuah *folder* dalam *folder templates* dengan nama **error**



2. Buat sebuah halaman HTML bernama **404.html** yang berisi *source code* seperti dibawah ini. File HTML tersebut dibuat pada folder **error** yang baru Anda buat sebelumnya

```
1 <!DOCTYPE html>
2 <html xmlns:th = "http://www.thymeleaf.org">
3   <head>
4     <title> 404 Not found </title>
5   </head>
6
7   <body>
8     <h2> Halaman tidak ditemukan </h2>
9   </body>
10 </html>
```

3. Akses suatu halaman yang tidak tersedia dari aplikasi Anda. Misalnya halaman dengan mapping **/teacher**. Contoh tampilan yang keluar adalah sebagai berikut



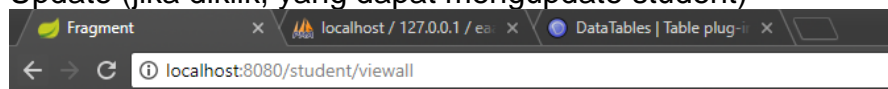
Halaman tidak ditemukan

Pertanyaan 6 : Apa *handler* dengan metode ini juga dapat berlaku bagi *error* lain seperti **error 500: internal server error** ?

- ➔ Tidak, error ini hanya handle kasus halaman yang tidak ditemukan, error lain seperti bad request tidak dihandle dalam metode ini.

Latihan

1. Pada tutorial, anda sudah mencoba untuk memperlihatkan apakah student tersebut *cum laude* atau tidak sampai ke bagian view. Pada latihan kali ini, anda diminta untuk mengubah view tersebut menggunakan DataTables. DataTables adalah plugin jquery *library*, yang dapat membuat sebuah tabel lebih interaktif dan lebih baik (<https://datatables.net/>). Oleh karena itu, gunakan data tabel untuk mengubah tampilan **localhost:8080/student/viewall**. Dengan kolom-kolom yang harus ada sebagai berikut:
 1. No
 2. NPM
 3. Name
 4. GPA
 5. Cum laude
 6. Delete (jika diklik, maka dapat menghapus student)
 7. Update (jika diklik, yang dapat mengupdate student)



All Students

No	NPM	Name	GPA	Cum laude	Delete	Update
1	123	Fajar	3.3	Sangat Memuaskan!	Delete Data	Update Data
2	124	Ammar	3.25	Sangat Memuaskan!	Delete Data	Update Data
3	125	Adit	3.15	Sangat Memuaskan!	Delete Data	Update Data

```
1 <!DOCTYPE html>
2 <html xmlns:th="http://www.thymeleaf.org">
3 <head>
4 <title th:replace="fragments/fragment :: head"></title>
5 <script src="https://ajax.googleapis.com/ajax/libs/jquery/3.2.1/jquery.min.js"></script>
6 <link rel="stylesheet" type="text/css" href="/css/datatables.min.css"></link>
7 <script type="text/javascript" src="/js/datatables.min.js"></script>
8 </head>
9 <body>
10 <h1>All Students</h1>
11 <table id="viewallStudents" class="display">
12 <thead>
13 <tr>
14 <th>No</th>
15 <th>NPM</th>
16 <th>Name</th>
17 <th>GPA</th>
18 <th>Cum laude</th>
19 <th>Delete</th>
20 <th>Update</th>
21 </tr>
22 </thead>
23 <tbody>
24 <tr th:each="student, iterationStatus: ${students}" th:class="${iterationStatus.odd}?'odd'">
25 <td th:text="${iterationStatus.count}">No. 1</td>
26 <td th:text="${student.npm}">Student NPM</td>
27 <td th:text="${student.name}">Student Name</td>
28 <td th:text="${student.gpa}">Student GPA</td>
29 <td th:if="${student.gpa}>=3.49">Cum Laude</td>
30 <td th:unless="${student.gpa}>=3.49">Sangat Menakutkan</td>
31 <!-- jika lain kondisi conditional expression ini akan sama dengan 2 baris diatas -->
32 <td><h3 th:text="${student.gpa}>=3.49 ? 'Cum Laude' : 'Sangat Menakutkan'"></h3> -->
33 <td><a th:href="/student/delete/" + ${student.npm}">Delete Data</a></td>
34 <td><a th:href="/student/update/" + ${student.npm}">Update Data</a></td>
35 </tr>
36 </tbody>
37 </table>
38 <script>
39 $(document).ready(function(){
40 $('#viewallStudents').DataTable();
41 });
42 </script>
43 </body>
44 </html>
```

2. Pada tutorial, Anda sudah mencoba membuat *fragment header* dan *footer*. Tetapi, *header* dan *footer* pada tutorial tersebut tidak dinamis. Data HTML pada *header* atau *footer* akan sama di setiap *view* yang memakai *fragment* tersebut. Pada latihan kali ini, Anda diminta untuk membuat sebuah *fragment header* yang dinamis. Dimana, aspek dinamis tersebut terdapat pada **title yang diletakkan pada *fragment header***, bukan setiap page HTML yang ada. Lebih jauh, *title* tersebut akan berubah-ubah sesuai dengan *page* yang ada. Silahkan implementasi dengan kebutuhan tersebut, dan jika sudah berhasil, **pastikan *header* tersebut dimuat pada setiap *view* yang ada di *project* ini.**

```
1 <!DOCTYPE html>
2 <html xmlns:th="http://www.thymeleaf.org">
3 <head>
4 <title th:fragment="head" th:if="${httpServletRequest.requestURI == '/'}">Index</title>
5 <title th:fragment="head" th:if="${httpServletRequest.requestURI == '/delete'}">Delete</title>
6 <title th:fragment="head" th:if="${httpServletRequest.requestURI == '/student/delete/${npm}'}">Delete Student</title>
7 <title th:fragment="head" th:if="${httpServletRequest.requestURI == '/student/add/submit'}">Add student</title>
8 <title th:fragment="head" th:if="${httpServletRequest.requestURI == '/student/update/submit'}">Update student</title>
9 <title th:fragment="head" th:if="${httpServletRequest.requestURI == '/student/add'}">Add</title>
10 <title th:fragment="head" th:if="${httpServletRequest.requestURI == '/student/update/${npm}'}">Update</title>
11 <title th:fragment="head" th:if="${httpServletRequest.requestURI == '/student/view'}">View Student by NPM</title>
12 <title th:fragment="head" th:if="${httpServletRequest.requestURI == '/student/view/${npm}'}">View Student by NPM</title>
13 <title th:fragment="head" th:if="${httpServletRequest.requestURI == '/student/viewall'}">View All Students</title>
14 <title th:fragment="head" th:if="${httpServletRequest.requestURI == '/course/view/${id_course}'}">View Course</title>
15 </head>
16 <body>
17 <div th:fragment="header">
18 <ul>
19 <li><a href="/"> Home </a></li>
20 <li><a href="/student/viewall"> Daftar Mahasiswa </a></li>
21 <li><a href="/student/add"> Menambah Mahasiswa </a></li>
22 </ul>
23 </div>
24 <div th:fragment="footer">
25 <hr/>
26 <h4> Mata Kuliah APAP </h4>
27 </div>
28 </body>
29 </html>
```

Write-Up

Pada tutorial 6 ini saya belajar mengenai presentation layer, dimana saya belajar menggunakan Conditional Expression pada Thymeleaf, serta fitur fragments pada Thymeleaf agar halaman HTML dapat digunakan kembali (*reuse*).