

Write Up Tutorial 6

Arsitektur dan Pemrograman Aplikasi Perusahaan

Nama : Lintang Matahari Hasani

NPM : 1506689231

1. Tutorial

1.1 Penggunaan For Looping

Tampilan kode for looping pada laman viewall.html

```
<div th:each="student, iterationStatus: ${students}" th:class="${iterationStatus.odd}? 'odd'">
  <h3 th:text="No. ' + ${iterationStatus.count}">No. 1</h3>
  <h3 th:text="NPM = ' + ${student.npm}">Student NPM</h3>
  <h3 th:text="Name = ' + ${student.name}">Student Name</h3>
  <h3 th:text="GPA = ' + ${student.gpa}">Student GPA</h3>

  <h3>Kuliah yang diambil</h3>
  <ul th:each="course, iterationStatus: ${student.courses}">
    <li th:text="${course.name} + '- ' + ${course.credits} + ' sks'" >
      Nama kuliah-X SKS
    </li>
  </ul>
</div>
<!--
```

Tampilan pada browser

All Students

No. 1

NPM = 12345

Name = Joni

GPA = 3.5

Kuliah yang diambil

- MPKI-6 sks

[Update Data](#)
[Delete Data](#)

No. 2

NPM = 12347

Name = Mustafa Kemal Pasha

GPA = 3.9

Kuliah yang diambil

Pertanyaan 1 : Apakah Value yang dihasilkan dari `${iterationStatus.odd}` ?

`${iterationStatus.odd}` menghasilkan value boolean. Nilai true dihasilkan ketika index dari `iterationStatus` merupakan angka ganjil. Nilai false dihasilkan ketika index merupakan angka genap. Hal ini dapat dilihat ketika kode diatas dimodifikasi sebagai berikut

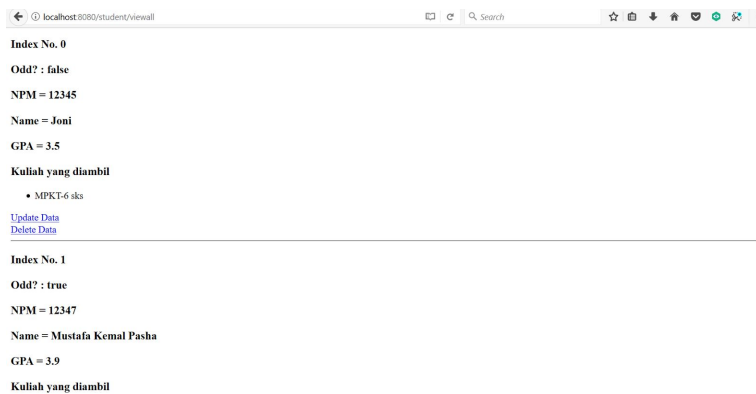
```
<div th:each="student, iterationStatus: ${students}" th:class="${iterationStatus.odd}? 'odd'">
  <h3 th:text="Index No. ' + ${iterationStatus.index}">No. 1</h3>
  <h3 th:text="Odd? : ' + ${iterationStatus.odd}">No. 1</h3>
  <h3 th:text="NPM = ' + ${student.npm}">Student NPM</h3>
  <h3 th:text="Name = ' + ${student.name}">Student Name</h3>
  <h3 th:text="GPA = ' + ${student.gpa}">Student GPA</h3>

  <h3>Kuliah yang diambil</h3>
  <ul th:each="course, iterationStatus: ${student.courses}">
    <li th:text="${course.name} + '- ' + ${course.credits} + ' sks'" >
      Nama kuliah-X SKS
    </li>
  </ul>
</div>
```

Nama : Lintang Matahari Hasani

NPM : 1506689231

Tampilannya adalah sebagai berikut:

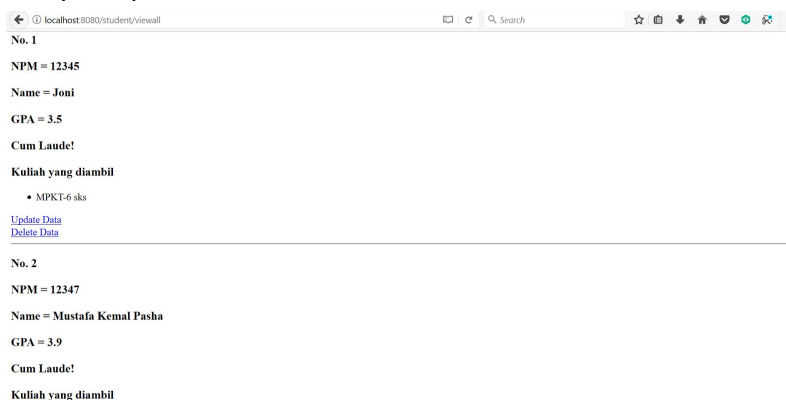


1.2 Penggunaan Conditional Expression pada Thymeleaf

Tampilan kode for loop pada viewall.html

```
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 = ' + ${student.gpa}">Student GPA</h3>
14
15 <h3 th:if="${student.gpa}>=3.49}">Cum Laude!</h3>
16 <h3 th:unless="${student.gpa}>=3.49}">Sangat Memuaskan!</h3>
17
18 <h3>Kuliah yang diambil</h3>
19= <ul th:each="course, iterationStatus: ${student.courses}">
20 <li th:text="${course.name} + '- ' + ${course.credits} + ' sks'" >
21 Nama kuliah-X SKS
22 </li>
23 </ul>
```

Tampilan pada browser



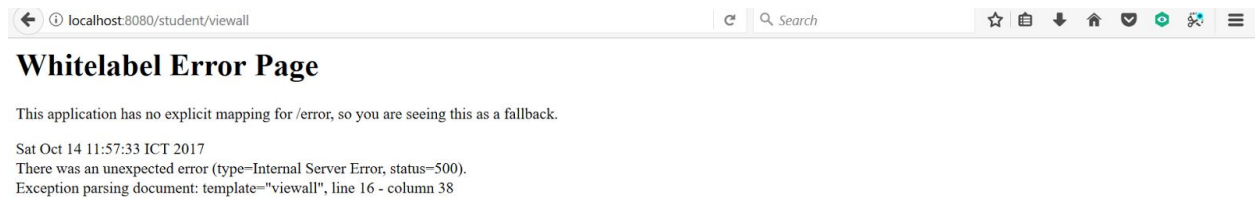
Pertanyaan 2 : Mengapa condition di dalam th:unless sama dengan th:if?

th:if adalah *conditional expression* dalam *thymeleaf* yang bermakna 'JIKA' dan akan mengembalikan nilai true jika condition di dalamnya terpenuhi. th:unless merupakan conditional expression yang bermakna 'KECUALI' dan akan mengembalikan true jika condition di dalamnya tidak terpenuhi atau merupakan komplemennya. Kedua *conditional expression* pada contoh menggunakan condition yang sama agar memenuhi kriteria cum laude untuk gpa ≥ 3.49 dan kriteria sangat memuaskan untuk yang memiliki gpa < 3.49 (komplemen dari kondisi pertama).

Nama : Lintang Matahari Hasani
NPM : 1506689231

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?

Ya, terjadi error pada aplikasi. Error terjadi karena `th:unless` dan `th:if` sama-sama dapat bernilai `true` sehingga terjadi ambiguitas untuk menampilkan status Cum Laude atau Sangat Memuaskan.

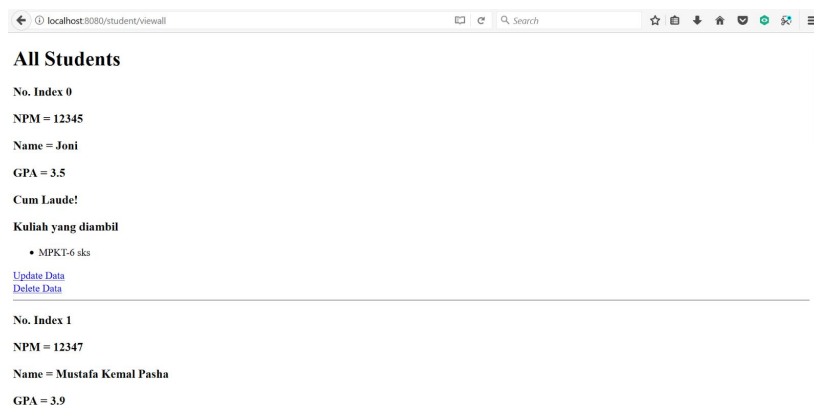


Pertanyaan 4: Tuliskan cara lain Anda dengan penulisan conditional expression yang berbeda untuk mendapatkan hasil yang sama seperti gambar di atas. *Hint: Gunakan ternary operator*

Untuk menampilkan hasil yang sama, dapat conditional expression berikut ini dapat digunakan

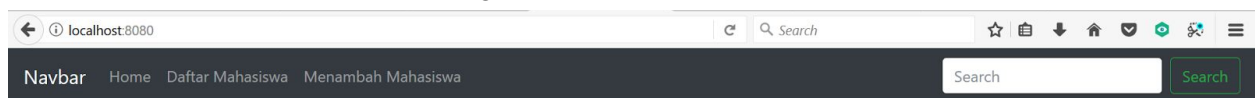
```
<h3 th:text="${student.gpa}>=3.49}? 'Cum Laude!' : 'Sangat Memuaskan!'">Predikat</h3>
```

Tampilan pada browser



1.3 Static File

Tampilan pada browser setelah mengimplementasikan Bootstrap



Welcome To Tutorial 6 APAP

Nama : Lintang Matahari Hasani
NPM : 1506689231

1.4 Fragment

Tampilan kode fragment.html

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

Tampilan kode index.html

```
Dashboard viewall.html index.html StudentController.java fragment.html
1 <!DOCTYPE html>
2 <html xmlns:th="http://www.thymeleaf.org">
3   <head>
4     <title>Index</title>
5   </head>
6   <body>
7     <div th:replace="fragments/fragment :: header"></div>
8     <h2>Welcome To Tutorial 6 APAP</h2>
9     <div th:replace="fragments/fragment :: footer"></div>
10  </body>
11 </html>
```

Tampilan pada browser

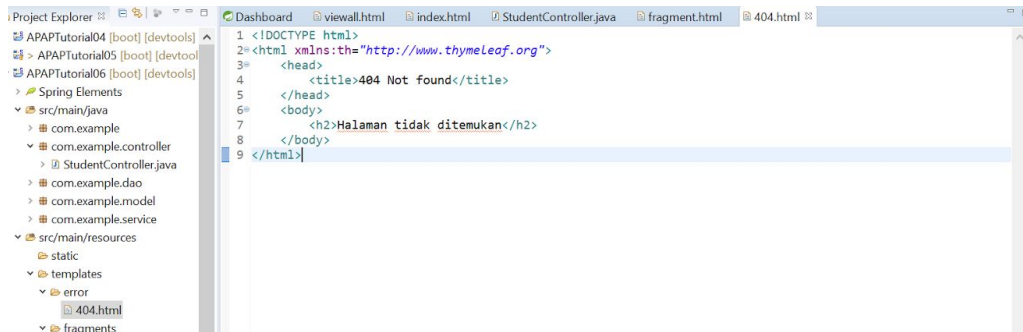


Pertanyaan 5 : Apa yang dimaksud dengan `th:replace="fragments/fragment :: header"` dan `th:replace="fragments/fragment :: footer"` pada file `index.html` yang Anda buat?

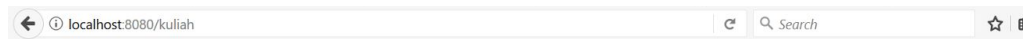
`th:replace="fragments/fragment :: header"` akan mengganti konten/isi dari tag `<div>` yang terkait menjadi potongan kode dari `<div>` dengan `th:fragment="header"` pada `fragment.html`. Hal yang serupa berlaku untuk `th:replace="fragments/fragment :: footer"` yang akan mengganti isi tag `<div>` terkait dengan potongan kode dari `<div>` dengan `th:fragment="footer"` pada `fragment.html`.

1.5 Error 404 Not Found Handler

Tampilan kode 404.html pada folder templates>error



Tampilan pada browser ketika mengakses halaman yang tidak ada



Halaman tidak ditemukan

2. Latihan

2.1 Menggunakan Datatables

Berikut merupakan langkah implementasi penggunaan datatables:

1. Membuat table untuk menampilkan semua student

```
<table class="display">
  <thead>
    <tr>
      <th>No. </th>
      <th>NPM </th>
      <th>Name </th>
      <th>GPA </th>
      <th>Cum Laude </th>
      <th>Update </th>
      <th>Delete</th>
    </tr>
  </thead>
  <tbody>
    <tr th:each="student, iterationStatus: ${students}">
      <td th:text="${iterationStatus.count}">No. </td>
      <td th:text="${student.npm}">Student NPM</td>
      <td th:text="${student.name}">Student Name</td>
      <td th:text="${student.gpa}">Student GPA</td>
      <td th:text="${student.gpa}>3.49? 'Cum Laude!' : 'Sangat Memuaskan!'">Cum Laude</td>
      <td><a th:href = "/student/update/" + ${student.npm}" > Update Data </a></td>
      <td><a th:href = "/student/delete/" + ${student.npm}" > Delete Data </a></td>
    </tr>
  </tbody>
</table>
```

2. Menambahkan link stylesheet, javascript dan jquery yang diperlukan untuk datatables

```
<link rel="stylesheet" type="text/css" th:href="@{https://cdn.datatables.net/1.10.16/css/jquery.dataTables.min.css}">
<script th:src="@{https://ajax.googleapis.com/ajax/libs/jquery/3.1.0/jquery.min.js}"></script>
<script type="text/javascript" charset="utf-8" th:src="@{http://cdn.datatables.net/1.10.16/js/jquery.dataTables.js}">
<script>
$(document).ready( function () {
    $('#table').DataTable();
  });
</script>
```

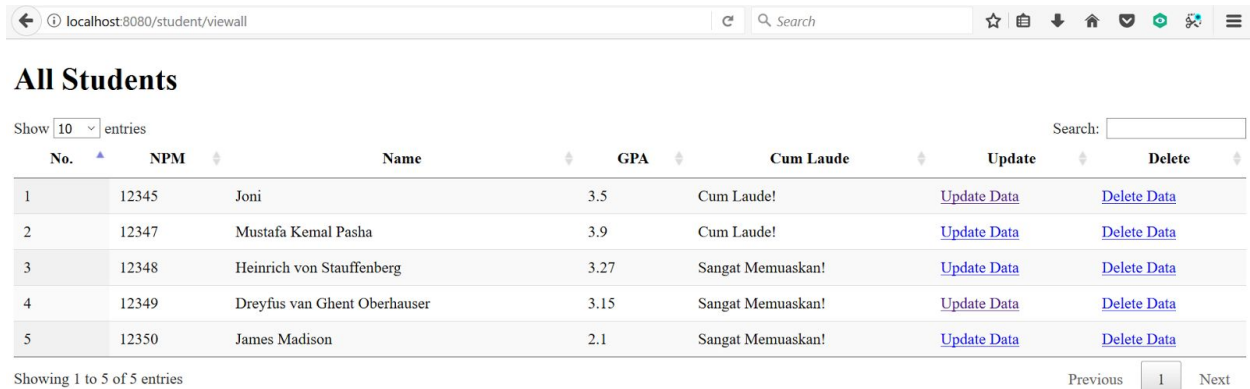
Nama : Lintang Matahari Hasani

NPM : 1506689231

3. Menambahkan id table sesuai dengan id pada jquery datatables

```
<table id="table" class="display">
  <thead>
```

Tampilan pada browser



No.	NPM	Name	GPA	Cum Laude	Update	Delete
1	12345	Joni	3.5	Cum Laude!	Update Data	Delete Data
2	12347	Mustafa Kemal Pasha	3.9	Cum Laude!	Update Data	Delete Data
3	12348	Heinrich von Stauffenberg	3.27	Sangat Memuaskan!	Update Data	Delete Data
4	12349	Dreyfus van Ghent Oberhauser	3.15	Sangat Memuaskan!	Update Data	Delete Data
5	12350	James Madison	2.1	Sangat Memuaskan!	Update Data	Delete Data


Showing 1 to 5 of 5 entries

Previous 1 Next

2.2 Dynamic Fragment

Berikut merupakan langkah implementasi dynamic fragment untuk title di setiap view.

1. Memodifikasi kode fragment.html yang memuat fragment untuk title yang dinamis.



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

2. Memodifikasi Controller (StudentController.java) dengan menambahkan model pada setiap method yang mengembalikan view. Model ini akan menyimpan String title untuk nama setiap laman pada view.

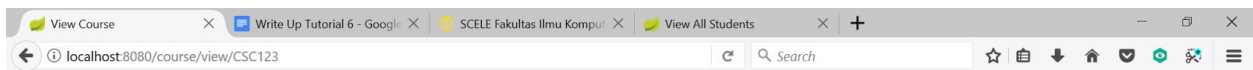
src/main/java/com/example/controller/StudentController.java - Eclipse

```
1 package com.example.controller;
2
3 import java.util.List;
4
5
6
7
8
9
10
11
12
13
14
15
16
17 @Controller
18 public class StudentController
19 {
20     @Autowired
21     StudentService studentDAO;
22
23     @RequestMapping("/")
24     public String index (Model model)
25     {
26         model.addAttribute("title", "Index");
27         return "index";
28     }
29
30     @RequestMapping("/index")
31     public String indexPath (Model model)
32     {
33         model.addAttribute("title", "Index");
34         return "index";
35     }
36
37
38     @RequestMapping("/student/add")
39     public String add (Model model)
40     {
41         model.addAttribute("title", "Add Student");
42         return "form-add";
43     }
44 }
```

3. Mengubah semua tag <title> pada semua view menjadi
<div th:replace="fragments/fragment :: title">

```
<head>
    <div th:replace="fragments/fragment :: title"></div>
</head>
```

Contoh tampilan title pada view view-course.html



3. Lesson Learned

Pada tutorial ini, saya mempelajari mengenai bagaimana cara mengimplementasikan For Looping dan Conditional Expression pada thymeleaf. Pada for looping th:each dapat digunakan untuk melakukan looping terhadap semua object yang diberikan pada view. Untuk menggunakan conditional expression, kita dapat menggunakan th:if yang mengembalikan nilai true bila condition terpenuhi, th:unless yang mengembalikan true bila condition tidak terpenuhi, dan ternary operator.

Selain itu, saya juga belajar cara menggunakan bootstrap sebagai front-end framework pada spring boot project dan mengaplikasikan fragment untuk layouting view yang dinamis. Fragment dapat diimplementasikan dengan membuat folder dan file fragment yang merupakan layout/template yang digunakan oleh semua view, seperti, title, header. Dan footer.

Pada tutorial ini saya juga belajar mengenai penggunaan datatables untuk menampilkan table yang lebih interaktif. Untuk menggunakan datatables, kita dapat menambahkan link css dan script terkait pada view dan memodifikasi tampilan view dengan menggunakan table yang memiliki id sesuai dengan jquery datatables.