

Nama : Windiany Lestari Sitorus

Kelas : Ekstensi

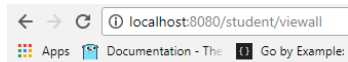
NPM : 1606955063

## Penggunaan For Looping

### Pertanyaan 1

1. Value yang dihasilkan `${iterationStatus.odd}` adalah menghasilkan `class='odd'` dengan `iterationStatus` yang ganjil.

Hasil tampilan pada browser akan sama saja, tidak ada perubahan.



### All Students

No. 1

NPM = 123

Name = ddsf

GPA = 3.0

[Delete Data](#)

[Update Data](#)

---

No. 2

NPM = 456

Name = Windiany Lestari Sitoruss

GPA = 4.0

[Delete Data](#)

[Update Data](#)

---

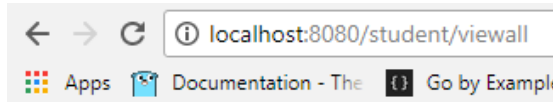
Akan tetapi apabila di inspect element akan ditemukan saat `iterationStatus` ke 2 (akan tetapi indexing dimulai dari 0) dengan index 1 = ganjil akan dicreate `class = 'odd'` sesuai code.

```
<div>
  <h3>No. 1</h3>
  <h3>NPM = 123</h3>
  <h3>Name = ddsf</h3>
  <h3>GPA = 3.0</h3>
  <a href="/student/delete/123"> Delete Data</a>
  <br>
  <a href="/student/update/123"> Update Data</a>
  <br>
  <hr>
</div>
<div class="test"> == $0
  <h3>No. 2</h3>
  <h3>NPM = 456</h3>
  <h3>Name = Windiany Lestari Sitoruss</h3>
  <h3>GPA = 4.0</h3>
  <a href="/student/delete/456"> Delete Data</a>
  <br>
  <a href="/student/update/456"> Update Data</a>
  <br>
  <hr>
</div>
```

## Penggunaan Conditional Expression pada Thymeleaf

### Pertanyaan 2

2. `th:unless` merupakan negative counterpart dari `th:if`, seperti not (!). Sehingga condition dalam `th:if` dan `th:unless` sama, karena sudah dibedakan dengan adanya `th:unless`. Apabila condition dalam `th:unless` diubah dengan kebalikan condition `th:if`, maka conditionnya akan sama dengan `th:if` dan akan menghasilkan error. `Th:unless` dapat berdiri sendiri tanpa `th:if`. Condition dalam `th:unless` juga tidak harus selalu sama dengan `th:if`, asalkan tidak memiliki makna yang sama dengan `th:if`.



### All Students

No. 1

**NPM = 123**

**Name = ddsf**

**GPA = 3.0**

**Sangat Memuaskan!**

[Delete Data](#)

[Update Data](#)

---

No. 2

**NPM = 456**

**Name = Windiany Lestari Sitoruss**

**GPA = 4.0**

**Cum Laude!**

[Delete Data](#)

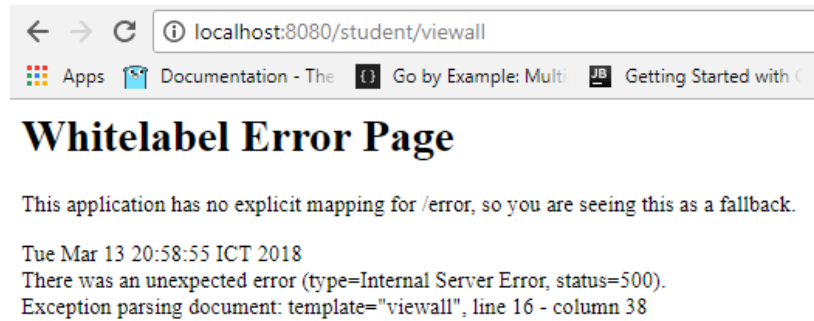
[Update Data](#)

---

### Pertanyaan 3

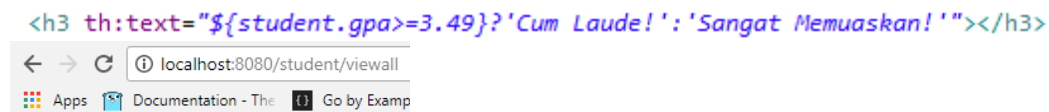
3. Terjadi error karena condition dalam `th:unless` menggambarkan hal yang sama dengan `th:if`. Pada `th:unless=${student.gpa<=3.48}` memiliki arti apabila `student.gpa` tidak lebih kecil sama dengan 3.48 sehingga target value nya adalah `student.gpa` lebih besar sama

dengan 3.49. Sementara hal yang sama juga terjadi akan di `th:if=${student.gpa}>=3.49}`. Hal ini menyebabkan error karena condition yang dicover memiliki arti yang sama.



#### Pertanyaan 4

4. Cara lain penulisan conditional expression menggunakan ternary operator adalah:



### All Students

No. 1

NPM = 123

Name = ddsf

GPA = 3.0

Sangat Memuaskan!

[Delete Data](#)

[Update Data](#)

No. 2

NPM = 456

Name = Windiany Lestari Sitoruss

GPA = 4.0

Cum Laude!

[Delete Data](#)

[Update Data](#)

## Static File

Pada bagian ini diajarkan menggunakan bootstrap.

## Fragment

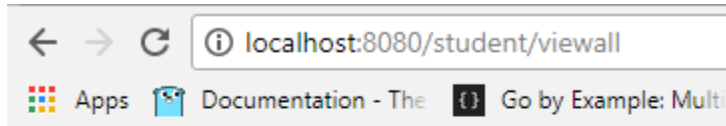
#### Pertanyaan 5

5. `th:replace="fragments/fragment::header"` memiliki arti menggantikan `<div` dengan fragment header yang telah didefine pada `fragments/fragment.html`. Berlaku juga dengan yang footer.

## Error 404 Not Found Handler

Pertanyaan 6

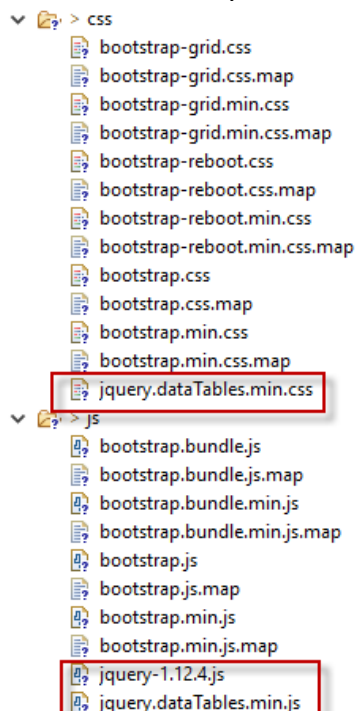
6. Handler ini juga berlaku bagi error lain, terbukti ketika saya mencoba file `500.html` di folder error dapat handle Internal server error yang terjadi. Jadi setiap file html yang dicreatae akan spesifik menhandle sesuai dengan HTML error code.



## Internal Server Error

## Latihan

1. Setelah membuka [https://datatables.net/examples/basic\\_init/zero\\_configuration.html](https://datatables.net/examples/basic_init/zero_configuration.html) menggunakan bagian Basic Initialisation, saya mengunduh file css dan js yang telah disediakan, kemudian menambahkan pada folder css dan js yang telah ditambahkan pada tutorial sebelumnya.



Kemudian menambahkan script css dan js pada viewall.html.

```
<link href="/css/jquery.dataTables.min.css" rel="stylesheet" />
<script type="text/javascript" src="/js/jquery-1.12.4.js"></script>
<script type="text/javascript" src="/js/jquery.dataTables.min.js"></script>
```

Kemudian menambahkan function data table.

```
<script th:inline="javascript">
$(document).ready(function() {
    $('#studentTable').DataTable();
} );
</script>
```

Dan mengubah script html menjadi seperti berikut:

```
<div class="container">
  <table id="studentTable" class="display" style="width:100%">
    <thead>
      <tr>
        <th>No</th>
        <th>NPM</th>
        <th>Name</th>
        <th>GPA</th>
        <th>Status</th>
        <th>Action</th>
      </tr>
    </thead>
    <tbody>
      <tr th:each="student, iterationStatus: ${students}" th:class="${iterationStatus.odd}? 'odd'">
        <td th:text="${iterationStatus.count}"></td>
        <td th:text="${student.npm}"></td>
        <td th:text="${student.name}"></td>
        <td th:text="${student.gpa}"></td>
        <td th:text="${student.gpa}>=3.49}? 'Cum Laude!': 'Sangat Memuaskan!'"></td>
        <td>
          <a th:href="'/student/delete/' + ${student.npm}" > Delete Data </a>
          <a th:href="'/student/update/' + ${student.npm}" > Update Data </a>
        </td>
      </tr>
    </tbody>
  </table>
</div>
```

Maka saat dijalankan di browser akan menjadi:

← → ↻ 📄 localhost:8080/student/viewall 🔍 ☆ 🌐 📱 🗑️ ⋮

📱 Apps 📖 Documentation - The 📄 Go by Example: Multi 📄 Getting Started with 📄 Deliver my data, Mr. J 📄 DB STAGING 📄 Family Gathering - G... 📄 Best Price on 24 Gue... 📄 MIRA PROD 📄 Windi - November 20... »

### All Students

Show 10 entries Search:

No	NPM	Name	GPA	Status	Action
1	11111122	Anastasia Grey	3.95	Cum Laude!	<a href="#">Delete Data</a> <a href="#">Update Data</a>
2	11111134	Selena Gomez	3.1	Sangat Memuaskan!	<a href="#">Delete Data</a> <a href="#">Update Data</a>
3	11111178	Putri Marino	3.99	Cum Laude!	<a href="#">Delete Data</a> <a href="#">Update Data</a>
4	11112222	Kendal Jenner	3.9	Cum Laude!	<a href="#">Delete Data</a> <a href="#">Update Data</a>
5	11112223	Kylie Jenner	3.7	Cum Laude!	<a href="#">Delete Data</a> <a href="#">Update Data</a>
6	11112244	Kim Kardashian	2.4	Sangat Memuaskan!	<a href="#">Delete Data</a> <a href="#">Update Data</a>
7	11112255	Khloe Kardashian	3.5	Cum Laude!	<a href="#">Delete Data</a> <a href="#">Update Data</a>
8	11112266	Kourtney Kardashian	2.9	Sangat Memuaskan!	<a href="#">Delete Data</a> <a href="#">Update Data</a>
9	11112277	Bella Hadid	2.5	Sangat Memuaskan!	<a href="#">Delete Data</a> <a href="#">Update Data</a>
10	11112288	Gigi Hadid	1.3	Sangat Memuaskan!	<a href="#">Delete Data</a> <a href="#">Update Data</a>

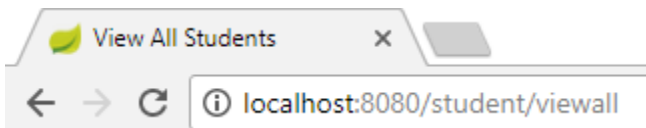
Showing 1 to 10 of 10 entries Previous 1 Next

2. Untuk membuat fragment header di bagian title, maka pada fragment.html ditambahkan variable \$title yang akan menampung nama title untuk setiap page yang akan ditampilkan.

```
<head th:fragment="headFragment(title)">
    <title th:text="${title}">Fragment</title>
    <link href="/css/jquery.dataTables.min.css" rel="stylesheet" />
</head>
```

Kemudian pada file html lainnya tambahkan, untuk menggantikan tag head dengan head yang telah didefine pada fragment.html. Disini variable title akan diganti dengan value title yang anda inginkan.

```
<head th:replace="fragments/fragment :: headFragment(title='View All Students')">
</head>
```



Menurut saya, latihan fragment ini menunjukkan bahwa code dalam fragment dapat di reuse di halaman html yang kita inginkan, sehingga code lebih efisien dan reusable.

Kemudian semua tag head pada file html akan menggunakan fragment header sehingga:

```
viewall.html  fragment.html  view.html
1 <!DOCTYPE html>
2 <html xmlns:th="http://www.thymeleaf.org">
3 <head th:replace="fragments/fragment :: headFragment(title='View Student by NPM')">
4 </head>
```

```
viewall.html  fragment.html  form-add.html
1 <!DOCTYPE HTML>
2 <html xmlns="http://www.w3.org/1999/xhtml"
3 xmlns:th="http://www.thymeleaf.org">
4 <head th:replace="fragments/fragment :: headFragment(title='Add Student')">
5 </head>
```

```
viewall.html  fragment.html  form-add.html  index.html
1 <!DOCTYPE html>
2 <html xmlns:th="http://www.thymeleaf.org">
3 <head th:replace="fragments/fragment :: headFragment(title='Index')">
4 </head>
```

```
viewall.html  fragment.html  form-add.html  index.html  form-update.html
1 <!DOCTYPE HTML>
2 <html xmlns="http://www.w3.org/1999/xhtml"
3 xmlns:th="http://www.thymeleaf.org">
4
5 <head th:replace="fragments/fragment :: headFragment(title='Update Student')">
6 </head>
```

## Ringkasan Materi Tutorial 5 APAP

1. Pada tutorial 5 ini bagian yang paling banyak dibahas adalah mengenai presentation layer atau view. Pada Java Spring Boot biasanya menggunakan template engine Thymeleaf.

2. Belajar penggunaan looping untuk menampilkan data berulang ulang menggunakan `th:each`.
3. Mempelajari conditional expression pada Thymeleaf menggunakan `th:if` dan `th:unless` juga conditional expression. Dan pada bagian ini juga dibahas mengenai condition di dalam `if` dan `unless`.
4. Pada tutorial ini juga diajarkan cara menggunakan bootstrap dalam project.
5. Belajar mengenai fragment, yang dapat digunakan untuk reuse code. Didefinisikan di fragment menggunakan `th:fragment` kemudian akan digunakan di halaman html yang kita inginkan menggunakan `th:replace`. Fragment ini sangat berguna apabila kita sering melakukan hal berulang ulang dalam file html kita.
6. Belajar mengenai error handler, dengan create file dengan nama sesuai error code HTTP, maka akan mampu handle saat error tersebut terjadi. Seperti yang telah dilakukan di tutorial, file `404.html` akan mampu handle error saat kita mengakses halaman yang tidak ada. Sama juga dengan internal server error akan mampu dihandle oleh `500.html`.
7. Pada tutorial ini juga dipelajari mengenai penggunaan data table untuk mengelola data yang ditampilkan.