



## **Thymeleaf – Presentation Layer Design**

### **Ringkasan Materi Tutorial 5**

Pada tutorial 5, hal yang dipelajari yaitu cara penggunaan for looping. Penggunaan for looping kali ini dengan cara menambahkan ekspresi iteration status. Iteration status ini berada dalam atribut th:each yang berfungsi untuk melakukan iterasi data mahasiswa yang ada pada database dan menampilkan pada halaman viewall.html. lalu, di tutorial ini mempelajari cara penggunaan conditional expression menggunakan th:if dan th:unless (sama seperti if dan else). Ekspresi kondisi digunakan untuk menyeleksi status dari suatu mahasiswa apakah 'Cum Laude!' atau 'Sangat Memuaskan!'. Selanjutnya hal yang dipelajari yaitu fragment pada thymeleaf. Fragment ini berfungsi untuk memecah halaman HTML menjadi bagian-bagian kecil sehingga bagian tersebut nantinya dapat digunakan oleh halaman HTML yang lain. Fragment nya ada header dan footer. Fragment tersebut akan dipanggil pada halaman index.html, viewall.html, dan form-update.html. Error handler juga dipelajari. Error handler berguna untuk menangani error dengan mengembalikan halaman html yang berisi informasi error tersebut. Dalam hal ini, error yang ditangani adalah error 400 dan 500.

### **Pertanyaan 1:**

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

### **Jawab :**

Tidak ada perubahan dari tampilan, namun akan membuat class baru yang akan menampilkan nilai odd pada tiap perulangan ganjil. IterationStatus yaitu suatu variable yang akan berulang sebanyak data yang ada di table student. itterationStatus adalah ekspresi singkat untuk menggambarkan if else.

### **Pertanyaan 2:**

Mengapa condition di dalam th:unless sama dengan di dalam th:if? Jelaskan jawaban Anda

### **Jawab :**

Karena th:unless menghasilkan data yang merupakan kebalikan dari kondisi yang diberikan. Pada case ini, kondisi  $GPA \geq 3.49$  berarti kondisinya adalah  $GPA \leq 3.49$



### 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?

### Jawab :

Terjadi error, karena if atau unless tidak dapat memproses karakter <

### 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

### Jawab :

Penulisan conditional expression menggunakan ternary operator dan hasilnya tetap sama

```
<!DOCTYPE html>
<html xmlns:th="http://www.thymeleaf.org">
  <head>
    <title>View All Students</title>
  </head>
  <body>
    <h1>All Students</h1>

    <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 = ' + ${student.gpa}">Student GPA</h3>
      <h3 th:text="${student.gpa} >= 3.49 ? 'Cum Laude!' : 'Sangat Memuaskan!'"></h3>
    </div>
  </body>
</html>
```

### Pertanyaan 5:

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

### Jawab :

Baris kode `replace` menandakan bahwa pada blok `div` tersebut akan digantikan dengan kode dari folder `fragments`, file `fragment.html` pada blok yang memiliki nama `fragment="header"` atau `fragment="footer"`



## Tutorial 05

### Arsitektur dan Pemrograman Aplikasi Perusahaan

#### Semester Genap 2017/2018

---

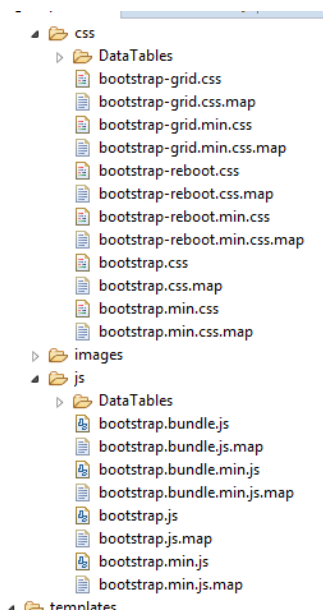
Pertanyaan 6: Apa handler dengan metode ini juga dapat berlaku bagi error lain seperti error 500: internal server error?

**Jawab :**

Berlaku. Error 500 tersebut dapat dipanggil sesuai dengan setting pemanggilannya.

#### Latihan

1. Mengubah tampilan localhost:8080/student/viewall menjadi lengkap dengan kolom-kolom yang diperlukan
  - a) Menampilkan data mahasiswa ke dalam bentuk DataTable. Download DataTable dan masukkan ke folder static seperti pada gambar dibawah ini



- b) DataTable dapat dijalankan dengan memanggil JQuery. Pada bagian head viewall.html lakukan pemanggilan file referensi

## Tutorial 05

### Arsitektur dan Pemrograman Aplikasi Perusahaan

### Semester Genap 2017/2018

```
viewall.html index.html form-add.html form-update... fragment.html 404.html 500.html
<!DOCTYPE html>
<html xmlns:th="http://www.thymeleaf.org">
  <head>
    <title>View All Students</title>
    <script type="text/javascript" charset="utf8" src="/js/DataTables/jquery-3.2.1.min.js"></script>
    <link rel="stylesheet" type="text/css" href="/css/DataTables/jquery.dataTables.min.css"></link>
    <script type="text/javascript" charset="utf8" src="/js/DataTables/jquery.dataTables.min.js"></script>
    <link rel="stylesheet" type="text/css" href="/css/bootstrap.min.css"></link>
    <script>
      $(document).ready( function () {
        $('#table_id').DataTable();
      } );
    </script>
  </head>
  <body>
```

c) Lalu tambahkan kolom-kolom yang diinginkan

```
<body>
  <div th:replace="fragments/fragment :: header"></div>
  <h1>All Students</h1>

  <table id="table_id" class="table table-bordered" style="width:100%">
    <thead>
      <tr>
        <th>No</th>
        <th>NPM</th>
        <th>Name</th>
        <th>GPA</th>
        <th>Cum laude</th>
        <th>Delete</th>
        <th>Update</th>
      </tr>
    </thead>
    <tbody>
```

d) Menuliskan code berikut pada halaman form-add.html untuk menambahkan data mahasiswa ke dalam table

## Tutorial 05

### Arsitektur dan Pemrograman Aplikasi Perusahaan

### Semester Genap 2017/2018

```
<!DOCTYPE HTML>
<html xmlns="http://www.w3.org/1999/xhtml"
      xmlns:th="http://www.thymeleaf.org">
<head>

<title>Add student</title>
<meta http-equiv="Content-Type" content="text/html; charset=UTF-8" />
<link rel="stylesheet" type="text/css" href="/css/bootstrap.min.css"></link>
</head>

<body>
<div th:replace="fragments/fragment :: header"></div>

<h2 class="page-header">Add Student</h2>
<nav class="navbar navbar-default">

<form action="/student/add/submit" method="get">
<div class="form-group">
<label for="npm">NPM</label> <input type="text" class="form-control" name="npm" />
</div>
<div>
<label for="name">Name</label> <input type="text" class="form-control" name="name" />
</div>
<div>
<label for="gpa">GPA</label> <input type="text" class="form-control" name="gpa" />
</div>

<div>
<button type="submit" class="btn btn-success" name="action" value="add">Add</button>
</div>
</form>
</nav>
```

Berikut merupakan tampilan saat menambahkan data mahasiswa

APAP EKSTENSI [Home](#) [Daftar Mahasiswa](#) [Menambah Mahasiswa](#)

## Add Student

NPM

Name

GPA

Data berhasil ditambahkan ke table

**Data berhasil ditambahkan**

Gambar dibawah merupakan halaman viewall berisi semua data mahasiswa dan data mahasiswa dengan nama Mutiarani Ramadhan yang baru saja ditambahkan



## Tutorial 05

### Arsitektur dan Pemrograman Aplikasi Perusahaan

### Semester Genap 2017/2018

APAP EKSTENSI Home Daftar Mahasiswa Menambah Mahasiswa						
All Students						
Show 10 entries		Search:				
No	NPM	Name	GPA	Cum laude	Delete	Update
1	140102920	Mutiarani Ramadhan	3.8	Cum Laude!	Delete	Update
2	1406557503	Desy Ulfasari	2.75	Sangat Memuaskan!	Delete	Update
3	1406623083	Dwi Arfi Faradina Mawardi	3.0	Sangat Memuaskan!	Delete	Update
4	1502839	Shabrina	3.2	Sangat Memuaskan!	Delete	Update
5	16038938	Rinjani Samosir	3.5	Cum Laude!	Delete	Update
6	1606954703	Aisyah Sadiyah	3.25	Sangat Memuaskan!	Delete	Update
7	1606954716	Akhda Khairul Zadid	3.0	Sangat Memuaskan!	Delete	Update
8	1606954735	Arri Kurniawan	2.75	Sangat Memuaskan!	Delete	Update
9	1606954741	Ayu Musfita Nainggolan	3.0	Sangat Memuaskan!	Delete	Update
10	1606954754	Budi Indrawan	3.5	Cum Laude!	Delete	Update

e) Berikut merupakan code untuk mengedit data mahasiswa

```
<title>Edit Student</title>
<meta http-equiv="Content-Type" content="text/html; charset=UTF-8" />
<link rel="stylesheet" type="text/css" href="/css/bootstrap.min.css"></link>
</head>
<body>
  <div th:replace="fragments/fragment :: header"></div>
  <h1 class="page-header">Edit Student</h1>
  <nav class="navbar navbar-default">
    <form action="/student/update/submit" method="post">
      <div class="form-group">
        <label for="npm">NPM</label> <input type="text" class="form-control" name="npm" readonly="true" th:value="$
      </div>
      <div>
        <label for="name">Name</label> <input type="text" class="form-control" name="name" th:value="{student.name
      </div>
      <div>
        <label for="gpa">GPA</label> <input type="text" class="form-control" name="gpa" th:value="{student.gpa}" /
      </div>
      <div>
        <button type="submit" name="action" class="btn btn-primary" value="save">Update</button>
      </div>
    </form>
  </nav>
</body>
</html>
```

Berikut merupakan tampilan halaman mengedit data sebelum diedit



## Tutorial 05

### Arsitektur dan Pemrograman Aplikasi Perusahaan

### Semester Genap 2017/2018

APAP EKSTENSI Home Daftar Mahasiswa Menambah Mahasiswa

## Edit Student

NPM

Name

GPA

Dan ini setelah diedit pada bagian GPA

APAP EKSTENSI Home Daftar Mahasiswa Menambah Mahasiswa

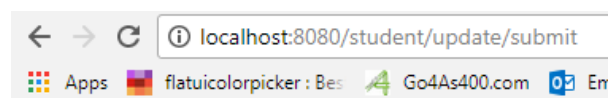
## Edit Student

NPM

Name

GPA

Data berhasil diedit



**Data berhasil diedit**

Berikut merupakan tampilan viewall beserta data yang berhasil diedit



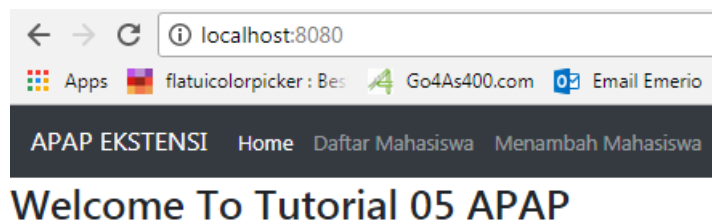
## Tutorial 05

### Arsitektur dan Pemrograman Aplikasi Perusahaan

### Semester Genap 2017/2018

APAP EKSTENSI Home Daftar Mahasiswa Menambah Mahasiswa							Search
All Students							Search:
Show 10 entries							
No	NPM	Name	GPA	Cum laude	Delete	Update	
1	140102920	Mutiarani Ramadhan	3.95	Cum Laude!	Delete	Update	
2	1406557503	Desy Ulfasari	2.75	Sangat Memuaskan!	Delete	Update	
3	1406623083	Dwi Arfi Faradina Mawardi	3.0	Sangat Memuaskan!	Delete	Update	
4	1502839	Shabrina	3.2	Sangat Memuaskan!	Delete	Update	
5	16038938	Rinjani Samosir	3.5	Cum Laude!	Delete	Update	
6	1606954703	Aisyah Sadiyah	3.25	Sangat Memuaskan!	Delete	Update	
7	1606954716	Akhda Khairul Zadid	3.0	Sangat Memuaskan!	Delete	Update	
8	1606954735	Arri Kurniawan	2.75	Sangat Memuaskan!	Delete	Update	
9	1606954741	Ayu Musfita Nainggolan	3.0	Sangat Memuaskan!	Delete	Update	
10	1606954754	Budi Indrawan	3.5	Cum Laude!	Delete	Update	

Berikut merupakan tampilan home



## 2. Membuat fragment header yang dinamis.

Membuat code untuk header di halaman index.html dimasukkan pada th:fragment="header" seperti berikut ini pada fragment.html



```
<!DOCTYPE html>
<html xmlns:th="http://www.thymeleaf.org">
<head>
<title>Fragment</title>
</head>
<body>
<div th:fragment="header">
<nav class="navbar navbar-expand-lg navbar-dark bg-dark">
<a class="navbar-brand" href="#">APAP EKSTENSI</a>
<button class="navbar-toggler" type="button" data-toggle="collapse"
data-target="#navbarSupportedContent"
aria-controls="navbarSupportedContent" aria-expanded="false"
aria-label="Toggle navigation">
<span class="navbar-toggler-icon"></span>
</button>
<div class="collapse navbar-collapse" id="navbarSupportedContent">
<ul class="navbar-nav mr-auto">
<li class="nav-item" th:classappend="${#httpServletRequest.getRequestURI() == '/' ? 'active':''}">
<a class="nav-link" href="/">Home</a>
</li>
<li class="nav-item" th:classappend="${#httpServletRequest.getRequestURI() == '/student/viewall' ? 'active':''}">
<a class="nav-link" href="/student/viewall">Daftar Mahasiswa</a>
</li>
<li class="nav-item" th:classappend="${#httpServletRequest.getRequestURI() == '/student/add' ? 'active':''}">
<a class="nav-link" href="/student/add">Menambah Mahasiswa</a>
</li>
</ul>
<form class="form-inline my-2 my-lg-0">
<input class="form-control mr-sm-2"
type="text">
</form>
</div>
```

Pada bagian footer tuliskan code berikut.

```
<div th:fragment="footer">
<hr/>
<h4>Mata Kuliah APAP</h4>
</div>
```

Lalu, header dan footer dapat diletakkan pada halaman index, daftar mahasiswa, menambahkan mahasiswa. Seperti berikut ini.

```
> <div class="collapse navbar-collapse" id="navbarSupportedContent">
> <ul class="navbar-nav mr-auto">
> <li class="nav-item" th:classappend="${#httpServletRequest.getRequestURI() == '/' ? 'active':''}">
> <a class="nav-link" href="/">Home</a>
> </li>
> <li class="nav-item" th:classappend="${#httpServletRequest.getRequestURI() == '/student/viewall' ? 'active':''}">
> <a class="nav-link" href="/student/viewall">Daftar Mahasiswa</a>
> </li>
> <li class="nav-item" th:classappend="${#httpServletRequest.getRequestURI() == '/student/add' ? 'active':''}">
> <a class="nav-link" href="/student/add">Menambah Mahasiswa</a>
> </li>
> </ul>
```

Bertujuan agar halaman web berjalan dengan dinamis. Jadi, saat user klik home maka tampilannya akan aktif pada menu home dan berlaku hal yang sama untuk menu daftar mahasiswa dan menambahkan mahasiswa.



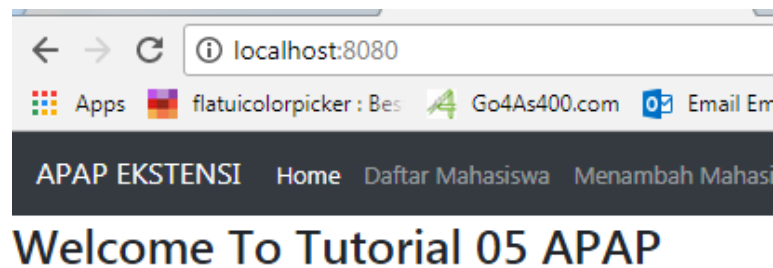
## Tutorial 05

### Arsitektur dan Pemrograman Aplikasi Perusahaan

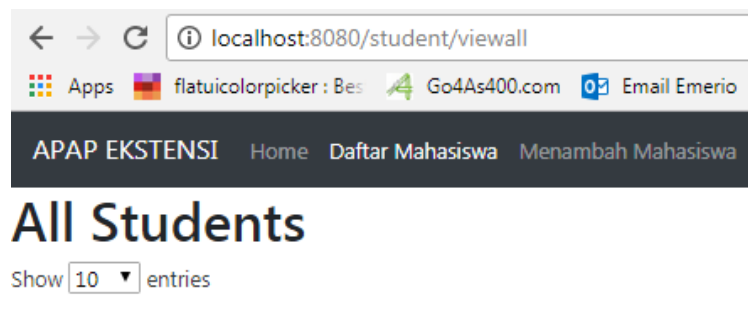
#### Semester Genap 2017/2018

---

Aktif pada menu **Home**



Aktif pada menu **Daftar Mahasiswa**



Aktif pada menu **Menambah Mahasiswa**

