

❖ Materi yang dipelajari

Pada tutorial 5 kali ini hal yang dipelajari antara lain cara penggunaan *for looping*. Untuk menggunakan *for looping* pada *project* kali ini adalah 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 menampilkannya pada halaman *viewall.html*. Selanjutnya, dipelajari cara penggunaan *conditional expression* menggunakan *th:if* dan *th:unless* (sama seperti *if* dan *else*). Ekspresi kondisi ini digunakan untuk menyeleksi status dari suatu mahasiswa apakah ‘Cum Laude!’ atau ‘Sangat Memuaskan!’. Berikutnya, hal dipelajari adalah mengenai 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. Pada *project* ini, fragment nya ada 2, yakni *header* dan *footer*. Nantinya, fragment ini akan dipanggil pada halaman *index.html*, *viewall.html*, dan *form-update.html*. Selanjutnya, dipelajari mengenai *error handler* melalui *thymeleaf*. *Error handler* ini berguna jika sebuah *error* muncul maka *error* tersebut nantinya dapat ditangani dengan mengembalikan halaman html yang berisi pemberitahuan *error* apa yang sedang terjadi. Dalam hal ini, *error* yang ditangani adalah *error* 400 dan 500. Terakhir, hal yang dipelajari dalam tutorial kali ini adalah bagaimana cara menggunakan *data tables* untuk menampilkan data mahasiswa pada halaman *viewall.html* agar tampilannya lebih interaktif dan lebih baik.

❖ Pertanyaan

→ Pertanyaan 1

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

Pada halaman *viewall.html* menghasilkan value ‘odd’.

```
<div class="odd">
  <a href="/student/delete/12">Delete Data</a><br />
  <a href="/student/update/12">Update Data</a><br />
  <h3>No. 2</h3>
  <h3>NPM = 12</h3>
  <h3>Name = Iqbal</h3>
  <h3>GPA = 9.0</h3>
  <hr />
</div>
```

→ Pertanyaan 2

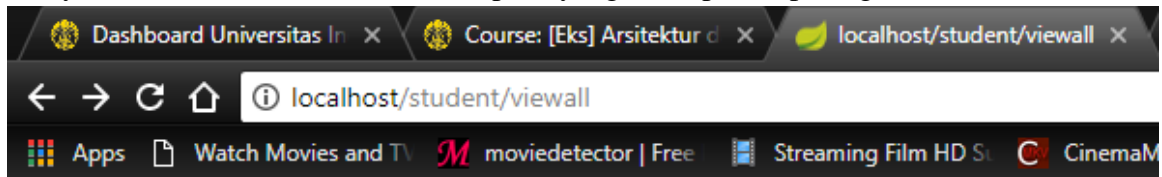
Mengapa *condition* di dalam **th:unless** sama dengan di dalam **th:if** ?

Karena anotasi ‘*th:unless*’ sama seperti anotasi ‘*else*’, sehingga ketika kondisi ≤ 3.49 masuknya ke kondisi ≥ 3.49 . Oleh karena itu, untuk operator perbandingan didalam ‘*th:if*’ dan ‘*th:unless*’ 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?

Iya, karena kondisi ≤ 3.48 masuknya kedalam kondisi ≥ 3.49 sehingga akan menyebabkan *internal server error* seperti yang ditampilkan pada gambar dibawah ini.



Whitelabel Error Page

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

Tue Mar 13 20:26:48 ICT 2018

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

Exception parsing document: template="viewall", line 17 - column 39

→ 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 menggunakan *ternary operator* seperti pada gambar dibawah ini

```
<div th:each="student,iterationStatus: ${students}" th:class="${iterationStatus.odd} ? 'odd' ">
  <a th:href="/student/delete/" + ${student.npm}">Delete Data</a><br/>
  <a th:href="/student/update/" + ${student.npm}">Update Data</a><br/>
  <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>
  <!-- <h3 value="${student.gpa <= 3.48 ? }">Sangat Memuaskan!</h3> -->
  <hr/>
</div>
```

→ Pertanyaan 5

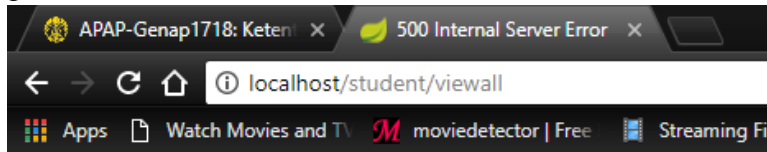
Apa yang dimaksud dengan **th:replace="fragments/fragment :: header"** dan **th:replace="fragments/fragment :: footer"** pada file **index.html** yang Anda buat?

Untuk **th:replace="fragments/fragment :: header"** berarti pada tag ini akan diganti dengan *fragment* yang bernama *header* yang berada pada folder *fragments*, sedangkan untuk **th:replace="fragments/fragment :: footer"** akan diganti dengan *fragment* yang bernama *footer*.

→ Pertanyaan 6

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

Bisa, caranya sama seperti diatas dengan membuat sebuah halaman .html pada folder *error*. Sehingga pada saat terjadi error 500, akan masuk halaman 500.html seperti pada gambar dibawah ini.

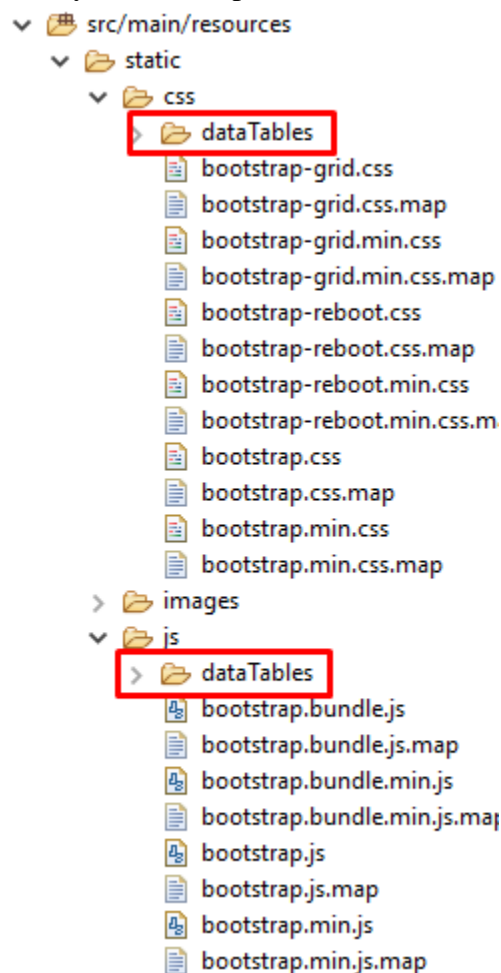


Halaman tidak ditemukan

❖ Latihan

→ Latihan 1

a. Library datatables pada folder src/main/resources/static



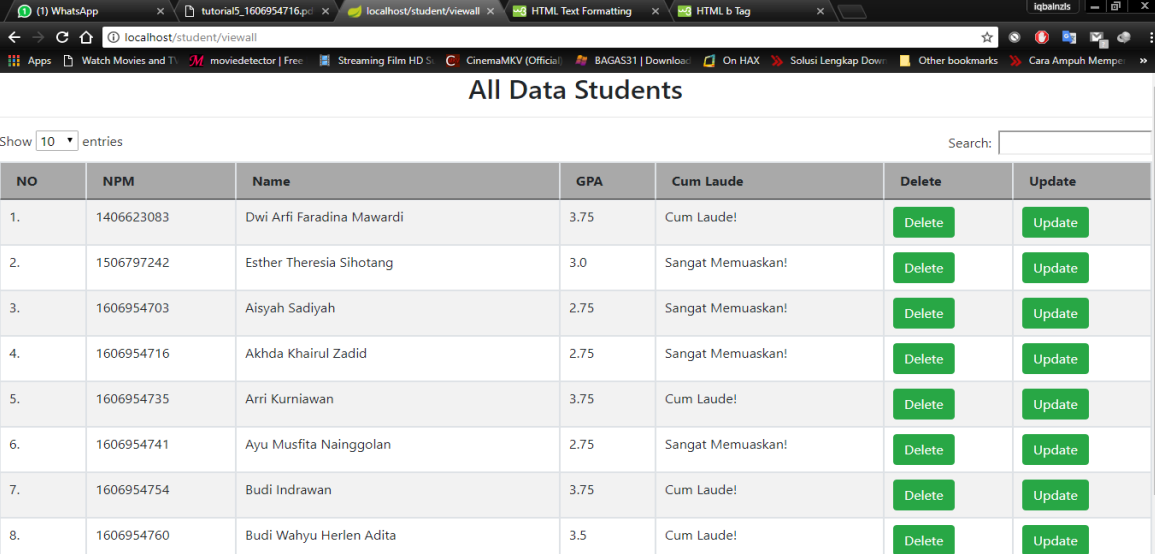
Pertama yang harus dilakukan adalah mendownload *library* datatables pada halaman <https://datatables.net/download/>. Setelah itu, ekstrak dan letakan library tersebut kedalam folder `src/main/resources/static`. Langkah berikutnya merubah tampilan halaman `viewall.html` dengan menggunakan datatables. Untuk code nya seperti pada gambar dibawah ini.

b. Halaman `viewall.html`

```
<!DOCTYPE html>
<html xmlns:th="http://www.thymeleaf.org">
<head>
<script src="/js/dataTables/jquery-3.2.1.min.js"></script>
<link rel="stylesheet" type="text/css" href="/css/dataTables/jquery.dataTables.min.css"/>
<link rel="stylesheet" type="text/css" href="/css/bootstrap.css"/>
<script src="/js/dataTables/jquery.dataTables.min.js"></script>
<script>
$(document).ready( function () {
    $('#table').DataTable();
} );
</script>
</head>
<body>
<div th:replace="fragments/fragment :: header"></div>
<h2 align="center">All Data Students</h2><hr/>
<table id="table" class="table table-striped table-bordered" style="width:100%">
<thead>
<th>NO</th>
<th>NPM</th>
<th>Name</th>
<th>GPA</th>
<th>Cum Laude</th>
<th>Delete</th>
<th>Update</th>
</thead>
<tbody>
<div th:each="student, iterationStatus: ${students}">
<tr>
<td><a th:text="${iterationStatus.count} + ' '"/></td>
<td><a th:text="${student.npm}"/></td>
<td><a th:text="${student.name}"/></td>
<td><a th:text="${student.gpa}"/></td>
<td><a th:text="${student.gpa} >= 3.49 ? 'Cum Laude!' : 'Sangat Memuaskan!'"></td>
<td><a th:href="/student/delete/" + ${student.npm}><button type="submit" name="action" value="save" class="btn btn-success">Delete</button></a></td>
<td><a th:href="/student/update/" + ${student.npm}><button type="submit" name="action" value="save" class="btn btn-success">Update</button></a></td>
</tr>
</div>
</tbody>
</table>
<div th:replace="fragments/fragment :: footer"></div>
</body>
</html>
```

Berikutnya buka browser dan akses <http://localhost/student/viewall> dan akan menampilkan halaman seperti gambar dibawah ini.

c. Tampilan halaman `viewall.html`



NO	NPM	Name	GPA	Cum Laude	Delete	Update
1.	1406623083	Dwi Arfi Faradina Mawardi	3.75	Cum Laude!	Delete	Update
2.	1506797242	Esther Theresia Sihotang	3.0	Sangat Memuaskan!	Delete	Update
3.	1606954703	Aisyah Sadiyah	2.75	Sangat Memuaskan!	Delete	Update
4.	1606954716	Akhda Khairul Zadid	2.75	Sangat Memuaskan!	Delete	Update
5.	1606954735	Arri Kurniawan	3.75	Cum Laude!	Delete	Update
6.	1606954741	Ayu Musfita Nainggolan	2.75	Sangat Memuaskan!	Delete	Update
7.	1606954754	Budi Indrawan	3.75	Cum Laude!	Delete	Update
8.	1606954760	Budi Wahyu Herlen Adita	3.5	Cum Laude!	Delete	Update

→ Latihan 2

a. Halaman fragment.html

Dalam tahap ini, *code* untuk *header* yang berada pada halaman index.html dimasukan keladam tag `th:fragment="header"` pada halaman fragment.html. Untuk lebih jelasnya dapat dilihat pada gambar dibawah ini.

```
<!DOCTYPE html>
<html>
<head>
  <title>Fragment</title>
</head>
<body>
  <div th:fragment="header">
    <link rel="stylesheet" href="/css/bootstrap.min.css"/>
    <nav class="navbar navbar-expand-lg navbar-dark bg-dark">
      <a class="navbar-brand" style="position: fixed; align:left"></a>
      <button class="navbar-toggler" type="button" data-toggle="collapse" data-target="#navbarSupportedContent"
        <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="${#httpServletRequest.getRequestURI() == '/' ? 'active':''}">
            <a class="nav-link" th:href="@{/}">Home</a>
          </li>
          <li class="nav-item" th:classappend="${#httpServletRequest.getRequestURI() == '/student/viewall' ? 'active':''}">
            <a class="nav-link" th:href="@{/student/viewall}">Daftar Mahasiswa</a>
          </li>
          <li class="nav-item" th:classappend="${#httpServletRequest.getRequestURI() == '/student/add' ? 'active':''}">
            <a class="nav-link" th: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" placeholder="Search" aria-label="Search"/>
          <button class="btn btn-outline-success my-2 my-sm-0" type="submit">Search</button>
        </form>
      </div>
    </nav>
  </div>

```

Sedangkan untuk bagian footer nya seperti pada gambar dibawah ini.

```
<div th:fragment="footer">
  <hr/>
  <h6 style="position: fixed; bottom: 10; width:100%; text-align: left">Mata Kuliah APAP 2018</h6>
</div>

```

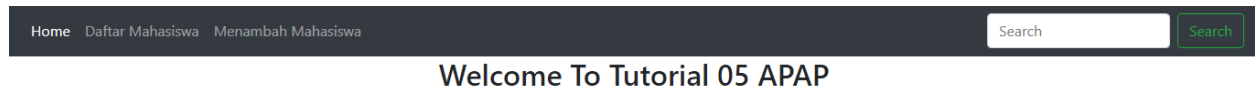
Pada tahap ini, bagian *header* dan *footer* pada fragment sudah dapat ditampilkan pada halaman home, daftar mahasiswa, dan menambah mahasiswa. Tetapi, pada navbar belum aktif sesuai dengan halaman yang dibuka, untuk mengaktifkannya perlu menambahkan code pada setiap navbar seperti pada gambar dibawah ini.

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

```

Setelah itu, akses kembali halaman home dan akan menampilkan gambar dibawah ini.

c. Tampilan halaman index.html



Mata Kuliah APAP 2018

Dapat dilihat pada gambar diatas, saat halaman home terbuka navbar 'Home' juga ikut *ter-highlight* sedangkan navbar yang lain tidak.