

TUTORIAL 6 APAP

Pada tutorial kali ini membahas mengenai Persentation Layer

PENGUNAAN FOR LOOPING

PERTANYAAN 1

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

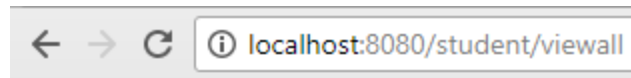
Fungsi nya adalah memberikan value berupa angka-angka yang ganjil

PENGUNAAN CONDITIONAL EXPRESSION PADA THYMELEAF

1. Menambahkan kode seperti berikut pada viewall.html

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

2. Mengakses halaman localhost:8080/student/viewall



All Students

No. 1

NPM = 123

Name = Mia

GPA = 3.2

SangatMemuaskan!

[Delete Data](#)

[Update Data](#)

Kuliah yang diambil

- MPKT-6sks

No. 2

NPM = 124

Name = Ririn

GPA = 3.9

Cum Laude!

[Delete Data](#)

[Update Data](#)

Kuliah yang diambil

- PSP-4sks
- SDA-3sks

TUTORIAL 6 APAP

PERTANYAAN 2

Condition th:unless sama dengan condition di dalam th:if karena kondisi didalam unless merupakan kondisi negatif dari apa yang ada di dalam kondisi if

th:if="{student.gpa}>=3.49}">Cum Laude!

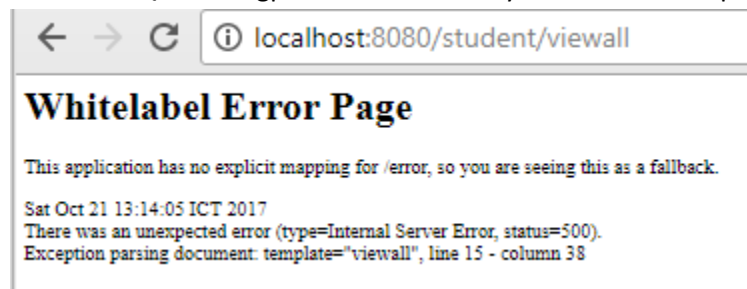
Maka th:unless="{student.gpa}>=3.49}">SangatMemuaskan

Sehingga jika student.gpa < 3.49 maka akan masuk ke dalam kondisi unless

PERTANYAAN 3

Mengubah kondidi pada th:unless="{student.gpa}>=3.49} menjadi

th:unless="{student.gpa}<=3.48 akan menyebabkan error seperti berikut :



Hal ini dikarenakan nilai dari attribute th:unless berasosiasi dengan elemen bertipe null yang tidak boleh mengandung karakter < , selain itu kondisi yang diberikan antara if dan unless berbeda sehingga tidak dapat dioperasikan

PERTANYAAN 4

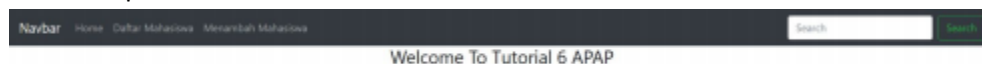
Cara lain untuk menuliskan conditional expression

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

STATIC FILE

1. BOOTSTRAP

- Mengekstrak folder zip bootstrap yang diunduh ke dalam folder static yang terdapat pada src/main/resources
- Membuat halaman index.html dengan menambahkan <link rel="stylesheet" href="/css/bootstrap.min.css"/>
- Hasil Tampilan



2. FRAGMENT

Fragment memungkinkan untuk me-reuse code dari halaman HTML nya dengan memecah halam HTML menjadi bagian-bagian kecil dan dapat digunakan oleh halaman HTML lainnya.

- Membuat folder fragments di dalam folder templates

TUTORIAL 6 APAP

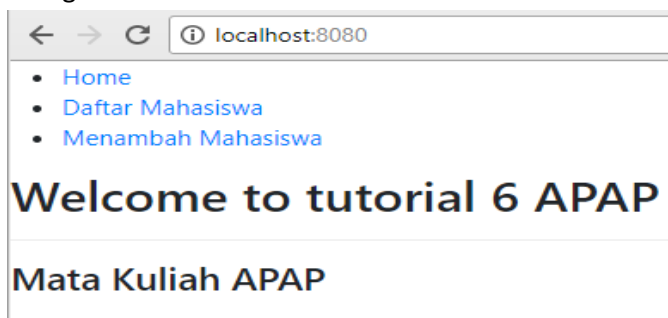
- b. Membuat halaman fragment.html

```
<!DOCTYPE html>
<html xmlns:th="http://www.thymeleaf.org">
<head>
<title>Fragment</title>
</head>
<body>
<div th:fragment="header">
<ul>
<li><a href="/">Home</a></li>
<li><a href="/student/viewall">Daftar Mahasiswa</a></li>
<li><a href="/student/add">Menambah Mahasiswa</a></li>
</ul>
</div>
<div th:fragment="footer">
<hr/>
<h4>Mata Kuliah APAP</h4>
</div>
</body>
</html>
```

- c. Pada source code diatas, th:fragment merepresentasikan nama fragment dari bagian yang ingin dijadikan sebuah fragment.
- d. Membuat halaman index.html sebagai berikut

```
<!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>
```

- e. Mengakses halaman localhost:8080



PERTANYAAN 5

Yang dimaksud dengan th:replace="fragments/fragment :: header" dan th:replace="fragments/fragment :: footer" pada file index.html adalah semua bagian pada <div> sampai </div> pada fragment.html memiliki nama header dan footer yang akan diimport kepada index.html

TUTORIAL 6 APAP

3. ERROR

- a. Membuat folder error di dalam folder templates
- b. Membuat halaman 404.html yang berisi

```
<!DOCTYPE html>
<html xmlns:th="http://www.thymeleaf.org">
  <head>
    <title>404 Not found</title>
  </head>
  <body>
    <h2>Halaman tidak ditemukan</h2>
  </body>
</html>
```

- c. Mengakses suatu halaman yang tidak tersedia dengan mapping /teacher



Halaman tidak ditemukan

PERTANYAAN 6

Handler dengan metode ini dapat berlaku bagi error lain seperti error 500: internal server error.

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)

TUTORIAL 6 APAP

```
<!DOCTYPE html>
<html xmlns:th="http://www.thymeleaf.org">

  <head>
    <title th:replace="fragments/fragment :: head"></title>
    <link rel="stylesheet" href="/js/jquery.dataTables.min.js"/>
    <link rel="stylesheet" href="/css/jquery.dataTables.min.css"/>
    <link rel="stylesheet" href="/js/dataTables.bootstrap.min.js"/>
    <link rel="stylesheet" href="/css/bootstrap.min.css"/>
    <script>
      $(document).ready(function() {
        $('#students').DataTable();
      });
    </script>
  </head>
  <body>
    <h3>Kuliah yang diambil</h3>

    <table id="students" class="table table-striped table-bordered" cellspacing="0" width="100%">
      <thead>
        <tr>
          <th>No</th>
          <th>NPM</th>
          <th>Name</th>
          <th>GPA</th>
          <th>CumLaude</th>
          <th>Delete</th>
          <th>Update</th>
        </tr>
      </thead>
      <tbody>
        <tr th:each="student, iterationStatus : ${students}">
          <td th:text="${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</a></td>
          <td><a th:href="'/student/update/' + ${student.npm}">Update</a></td>
        </tr>
      </tbody>
    </table>
    <hr/>

  </body>
</html>
```

TUTORIAL 6 APAP

Kuliah yang diambil						
No	NPM	Name	GPA	Cumlaude	Delete	Update
1	123	Mia	3.2	Sangat Memuaskan	Delete	Update
2	124	Ririn	3.9	Cum Laude!	Delete	Update

- 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.

Mengubah isi dari fragment:html

```
<!DOCTYPE html>
<html xmlns:th="http://www.thymeleaf.org">
<head>
<title th:fragment="head" th:if="${httpServletRequest.requestURI == '/'}">Index</title>
<title th:fragment="head" th:if="${httpServletRequest.requestURI == '/course/view/${id_course}'}">View Course</title>
<title th:fragment="head" th:if="${httpServletRequest.requestURI == '/delete'}">Delete</title>
<title th:fragment="head" th:if="${httpServletRequest.requestURI == '/student/delete/${npm}'}">Delete Student</title>
<title th:fragment="head" th:if="${httpServletRequest.requestURI == '/student/add'}">Add Student</title>
<title th:fragment="head" th:if="${httpServletRequest.requestURI == '/student/add/submit'}">Add</title>
<title th:fragment="head" th:if="${httpServletRequest.requestURI == '/student/view'}">View Student by NPM</title>
<title th:fragment="head" th:if="${httpServletRequest.requestURI == '/student/view/${npm}'}">View Student by NPM</title>
<title th:fragment="head" th:if="${httpServletRequest.requestURI == '/student/viewall'}">View All Students</title>
<title th:fragment="head" th:if="${httpServletRequest.requestURI == '/student/update/${npm}'}">Update Student</title>
<title th:fragment="head" th:if="${httpServletRequest.requestURI == '/student/update/submit'}">Update</title>
</head>
<body>
<div th:fragment="header">
<ul>
<li><a href="/">Home</a></li>
<li><a href="/student/viewall">Daftar Mahasiswa</a></li>
<li><a href="/student/add">Menambah Mahasiswa</a></li>
</ul>
</div>
<div th:fragment="footer">
<hr/>
<h4>Mata Kuliah APAP</h4>
</div>
</body>
</html>
```

Memanggil pada setiap file htmlnya dengan menambahkan `th:replace="fragments/fragment :: head"`

Contohnya : form-update

```
</div>
```