

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

Jawaban: Value yang dihasilkan menyatakan dia ganjil atau tidak berupa Boolean.

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

Jawaban: Condition akan masuk ke `unless` apabila condition dalam `if` tidak dipenuhi. Jika gpa tidak lebih besar dari 3.49 maka akan ditampilkan teks lain selain `cum laude`!

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?

Jawaban: Error, karena hasil dari `if` dan `unless`nya mengeluarkan hasil yang sama

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.

Jawaban: `<tr th:text="${student.gpa}>=3.49}? 'Ya' : 'Tidak' "></tr>`

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

Jawaban: Bagian tersebut di replace dengan fragment tersebut.

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

Jawaban: Tidak, karena itu hanya berlaku untuk Not Found.

Latihan

1. DataTable

Untuk mengimplementasikan DataTable, panggil CDN dari DataTable, JQuery, dan JavaScript dari DataTable.

```
<link rel="stylesheet" type="text/css" th:href="@{https://cdn.datatables.net/1.10.16/css/jquery.dataTables.min.css}"></link>
<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>
```

Kemudian panggil fungsi DataTable dengan Script

```
<script>
    $(document).ready( function () {
        $('#example').DataTable();
    } );
</script>
```

Setelah itu masukkan id pada table, pada case ini id adalah example

```
<table id="example" 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>
```

Setelah itu masukkan kode untuk menggenerate data

```
<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>
```

Hasil jadi:

No.	NPM	Name	GPA	Cum Laude	Update	Delete
1	123	Chanak	4.0	Cum Laude!	Update Data	Delete Data
2	1234	jessica	3.5	Cum Laude!	Update Data	Delete Data

2. Fragment

Untuk membuat fragment yang dinamis, langkah-langkah sebagai berikut:

- Mengganti kode title pada fragment.html sehingga dapat menerima input model title dari controller

```
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>
```

- Setelah itu, pada controller ditambahkan atribut title berisi String sehingga nantinya akan ditampilkan pada setiap view

```
@RequestMapping("/")
public String index (Model model)
{
    model.addAttribute("title", "Index");
    return "index";
}
```

- Setelah itu pada setiap view ditambahkan div untuk mengganti title dengan title yang sesuai dimasukkan pada controller

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

Lesson learned

Pada tutorial ini, saya belajar For Looping dengan menggunakan th:each dan Conditional dengan menggunakan th:if pada Thymeleaf. Selain itu saya juga belajar mengenai DataTable dan Fragment. Dimana DataTable berguna untuk memudahkan dalam mengambil data sekaligus mempercantik tampilan dari table, sementara fragment mempermudah layouting serta membuat tampilan lebih dinamis.