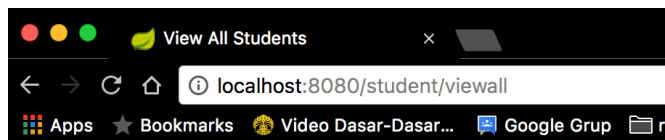


Pertanyaan 1: Apakah value yang dihasilkan dari `${iterationStatus.odd}` ? value yang dihasilkan berupa Boolean

Penggunaan Conditional Expression pada Thymeleaf

kondisi tertentu. Berikut ini cara yang dapat dilakukan.

1. Tambahkan kode baris berikut ini pada file **viewall.html** .
2. Coba run aplikasi Anda, lalu akses **localhost:8080/student/viewall** . Tampilan akan seperti gambar di bawah ini



All Students

No. 1

NPM = 123

Name = aduhai

GPA = 2.9

Sangat Memuaskan!

Kuliah yang diambil

- MPKT-6 sks

[Delete Data](#)

[Update Data](#)

No. 2

NPM = 124

Name = hoho

GPA = 3.6

Cum Laude!

Kuliah yang diambil

- PSP-4 sks
- SDA-3 sks

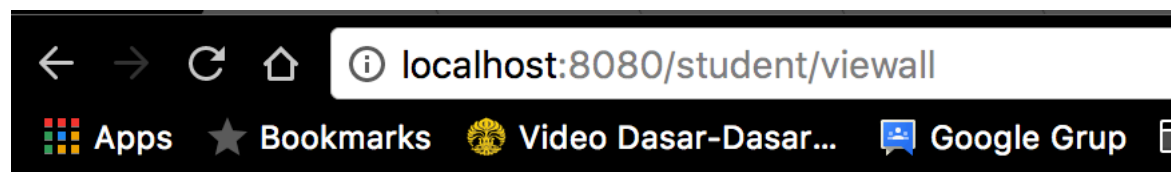
[Delete Data](#)

[Update Data](#)

Pertanyaan 2 : Mengapa *condition* di dalam **th:unless** sama dengan di dalam **th:if** ?
Jelaskan jawaban Anda.

th:if hasilnya merupakan negasi dari th:unless. Jika th:unless menghasilkan nilai true, jika condition yang diinput adalah false.

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?



Whitelabel Error Page

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

Sat Oct 21 02:21:24 WIB 2017

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

Exception parsing document: template="viewall", line 15 - column 42

Error. Hal ini karena th:if Karena th:if student.gpa >= 3.49 bernilai true, sedangkan seharusnya th:unless yang merupakan negasinya bernilai yaitu bernilai true. Namun, karena nilai yang diinput pada th:unless student.gpa <=3.49 yang bernilai true dan menghasilkan nilai false, maka menghasilkan error.

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*.

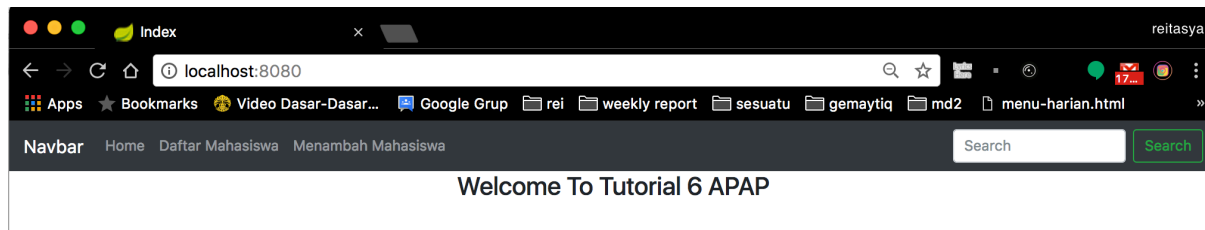
```
<h3 th:text="({{student.gpa}} >=3.49)? 'Cum Laude!' : 'Sangat Memuaskan!'"></h3>
```

Static file

1. Unduh bootstrap pada <http://getbootstrap.com/docs/4.0/getting-started/download/>

Reitasya Devina
1406623663

2. *Extract folder* zip yang Anda unduh ke dalam folder **static** yang terdapat pada folder **src/main/resources** .
3. Buat halaman **index.html** menjadi *source code* dibawah ini
4. Akses halaman index dari aplikasi Anda. Contoh tampilan adalah sebagai berikut.

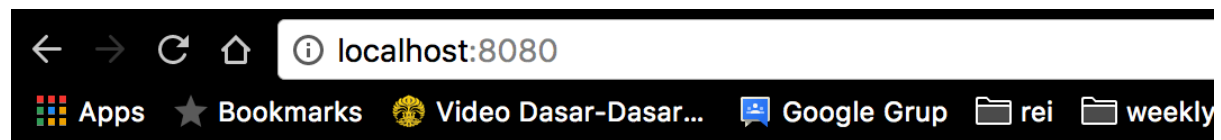


Fragment

Fragment merupakan sebuah fitur dalam Thymeleaf yang memungkinkan *developer* untuk me- *reuse source code* dari halaman HTML-nya. Fitur ini memungkinkan Anda untuk memecah halaman HTML menjadi bagian-bagian kecil dan bagian tersebut dapat digunakan oleh halaman HTML lain. *Code reuse* ini berguna ketika *developer* menggunakan komponen-komponen yang sering dipakai di berbagai bagian dalam aplikasi yang dibuatnya. Contohnya adalah pembuatan *header* dan *footer*. Thymeleaf juga sudah menyediakan dokumentasi mengenai *fragment* yang dapat diakses pada <http://www.thymeleaf.org/doc/articles/layouts.html>

Berikut ini adalah cara penggunaan *fragment* : 1. Buat sebuah *folder* dalam *folder templates* dengan nama **fragments** 2. Buat sebuah halaman HTML bernama **fragment.html** yang berisi *source code* seperti dibawah ini. File HTML tersebut dibuat pada folder **fragments** yang baru Anda buat sebelumnya

3. Pada *source code* diatas, **th:fragment** merepresentasikan nama *fragment* dari bagian yang Anda ingin jadikan sebuah *fragment*.
4. Buat halaman **index.html** menjadi *source code* dibawah ini
5. Coba jalankan halaman index yang Anda buat. Contoh tampilan yang keluar adalah sebagai berikut



- [Home](#)
- [Daftar Mahasiswa](#)
- [Menambah Mahasiswa](#)

Welcome To Tutorial 6 APAP

Mata Kuliah APAP

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

- `th:replace="fragments/fragment :: header"` -> untuk memasukkan header dari file `header.html` pada folder `fragments`
- `th:replace="fragments/fragment :: footer"` -> untuk memasukkan footer dari file `footer.html` pada folder `fragments`

Error 404 Not Found Handler

Berikut ini adalah cara untuk membuat *error 404 not found handler*:

1. Buat sebuah *folder* dalam *folder templates* dengan nama **error**
2. Buat sebuah halaman HTML bernama **404.html** yang berisi *source code* seperti dibawah ini. File HTML tersebut dibuat pada folder **error** yang baru Anda buat sebelumnya
3. Akses suatu halaman yang tidak tersedia dari aplikasi Anda. Misalnya halaman dengan mapping **/teacher** . Contoh tampilan yang keluar adalah sebagai berikut



Halaman tidak ditemukan

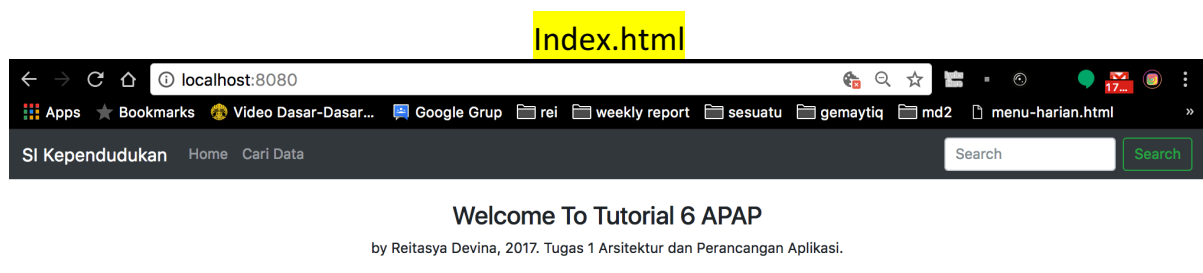
Pertanyaan 6 : Apa *handler* dengan metode ini juga dapat berlaku bagi *error* lain seperti ***error 500: internal server error*** ?

Ya

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 kolum-kolum 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)

2. Anda tidak harus menggunakan datables versi *server side* (https://datatables.net/examples/server_side/), yang mana lebih baik digunakan untuk data yang sangat banyak. Walaupun demikian, jika anda anda ingin mengimplementasi hal ini sangat diperbolehkan, tetapi tidak ada penambahan nilai untuk hal tersebut.
3. 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.**



Reitasya Devina
1406623663

Viewall.html

localhost:8080/student/viewall

SI Kependudukan Home Cari Data Search Search

All Students

Student GPA Student GPA Student GPA
Show 10 entries Search:

No.	Student NPM	Student Name	Student GPA	Status	Delete	Update
No. 1	NPM = 123	Name = aduhai	GPA = 2.9	Sangat Memuaskan!	Delete	Update
No. 2	NPM = 124	Name = hoho	GPA = 3.6	Cum Laude!	Delete	Update
No. 3	NPM = 125	Name = rei	GPA = 4.0	Cum Laude!	Delete	Update

Showing 1 to 3 of 3 entries Previous 1 Next

by Reitasya Devina, 2017. Tugas 1 Arsitektur dan Perancangan Aplikasi.

Form-add.html

localhost:8080/student/add

SI Kependudukan Home Cari Data

Add Student

NPM 1230

Name sya

GPA 3.99

Save

by Rei

Reitasya Devina
1406623663

update.html



Update Mahasiswa

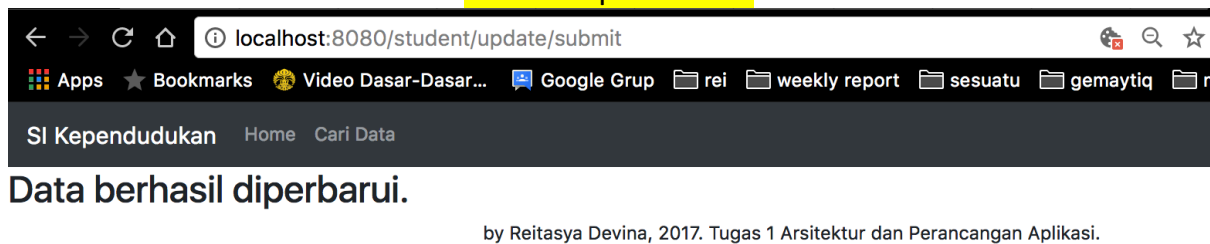
NPM

Name

GPA

by Reitasya Devina, 2017. Tugas 1 Arsitektur dan Perancangan Aplikasi.

success-update.html



Data berhasil diperbarui.

by Reitasya Devina, 2017. Tugas 1 Arsitektur dan Perancangan Aplikasi.

delete.html

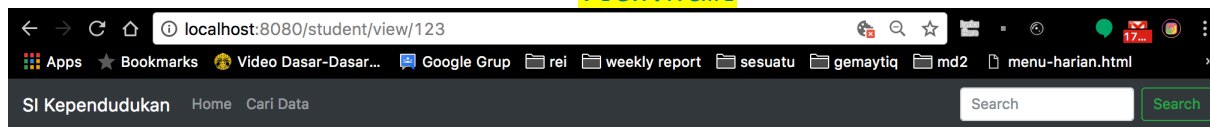


Data berhasil dihapus

by Reitasya Devina, 2017. Tugas 1 Arsitektur dan Perancangan Aplikasi.

Reitasya Devina
1406623663

view.html



Lihat Data Mahasiswa

NPM = 123

Name = aduhai

GPA = 2.9

Kuliah yang diambil

- MPKT-6 sks

by Reitasya Devina, 2017. Tugas 1 Arsitektur dan Perancangan Aplikasi.