

Write-Up Tutorial 6: Menggunakan Thymeleaf Sebagai Presentation Layer

Ringkasan

Di dalam tutorial ini saya melanjutkan tutorial 5 yang telah saya kerjakan. Kali ini saya mempelajari cara menggunakan Thymeleaf sebagai presentation layer. Hal-hal yang saya pelajari diantaranya penggunaan for looping, penggunaan conditional expression, static file, fragment, dan error handler. Selain itu, terdapat latihan menggunakan DataTables dan membuat fragment header yang dinamis.

Pada penggunaan for looping, saya mempelajari bahwa for looping dapat digunakan ketika menampilkan object yang berbentuk List. Selain itu, setiap iterasi dapat dilakukan pengecekan menggunakan status iterasi yang diberi nama pada th:each (bisa berupa iterationStatus atau iterStat).

Pada penggunaan conditional expression, saya mempelajari bahwa Thymeleaf dapat menggunakan conditional expression untuk menampilkan view tergantung kondisi yang muncul. Misalnya kita menampilkan status mahasiswa, apakah dia cum laude atau tidak, dengan menggunakan th:if dan th:unless. Selain itu, dapat juga digunakan ternary operator atau switch case untuk menampilkan beberapa input tertentu.

Pada penggunaan static file, saya mempelajari bahwa Thymeleaf dapat menggunakan ekstensi local seperti file css dan JavaScript. Ekstensi-ekstensi tersebut disimpan dalam folder static yang ada di folder resources. Folder static ini juga berguna menyimpan framework seperti Bootstrap dan DataTables.

Pada penggunaan fragment, saya mempelajari bahwa Thymeleaf dapat menggunakan fragment yang disimpan di dalam folder resources/templates/fragments. Fragment ini berisi bagian-bagian dari aplikasi yang dapat ditampilkan di dalam aplikasi melalui th:insert atau th:replace atau attribute lainnya yang mendukung fungsi dari fragment.

Pada penggunaan error handler, saya mempelajari bahwa Thymeleaf dapat menggunakan folder error pada template sehingga page yang keluar dapat berupa error page yang telah kita kustomisasi ketimbang whitelabel error page yang ditampilkan oleh Thymeleaf secara default.

Pada latihan dengan menggunakan DataTables, saya memasukkan resource DataTables ke dalam folder static untuk kemudian dipakai di template viewall. Mengikuti example dari DataTables yang ada di website DataTables, saya berhasil menggunakan DataTables untuk menampilkan daftar mahasiswa yang terdaftar di aplikasi tutorial ini.

Pada penggunaan dynamic header, saya masih merasa kebingungan ketika membaca instruksinya. Hal ini karena fragment bernama header ditujukan kepada elemen-elemen yang ada di body, bukan di title yang ada di header. Akhirnya saya memodifikasi title yang ada menggunakan switch case dan if clause dengan kondisi yang diambil dari URL yang sedang aktif untuk menampilkan judul page secara dinamis.

Jawaban dari Setiap Pertanyaan

1. `{iterationStatus.odd}` menandai baris dengan nomor urut ganjil. Misalkan kita menuliskan kode seperti di bawah ini

```
th:class="{iterationStatus.odd}? 'odd'"
```

Sukrian Syahnel Putra

1306402066

ADPAP - B

Maka untuk setiap row dengan nomor urut ganjil (1, 3, 5, 7, dst.) div yang bersangkutan akan memiliki kelas "odd".

Catatan: setiap iterasi pada Thymeleaf dimulai dari NOMOR URUT 0 yang merupakan bilangan genap, sehingga data yang dituliskan kelas "odd" ini bisa saja data dengan URUTAN genap (urutan ke-2, ke-4, ke-6, dst.) dibandingkan data dengan URUTAN ganjil (urutan ke-1, ke-3, ke-5, dst.)

2. Condition di dalam th:unless sama dengan th:if karena kedua statement adalah kontradiksi antara satu dan yang lainnya, dimana th:unless merupakan invers dari th:if. th:if akan mengeluarkan input jika kondisi di dalam if clause terpenuhi. Sementara itu, th:unless tidak akan mengeluarkan input jika kondisi di dalam unless clause terpenuhi.

3. Terjadi error ketika diproses. Hal ini terjadi karena value dari attribute "th:unless" tidak boleh mengandung karakter "<" atau "kurang dari".

4. Line th:text di dalam <h3> yang menampilkan "cum laude" diubah menjadi sebagai berikut:

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

5. th:replace="fragments/fragment :: header" menukar isi elemen di dalam div menjadi isi elemen fragment bernama header yang ada di dalam fragments/fragment.html. Sementara itu th:replace="fragments/fragment :: footer" menukar isi elemen di dalam div menjadi isi elemen fragment bernama footer yang ada di dalam fragments/fragment.html.

6. Handler ini juga berlaku untuk error lainnya, seperti error 500: internal server error dan error 400: bad request. Isinya juga bisa disesuaikan dengan kebutuhan aplikasi.