

TUTORIAL 5 : Thymeleaf – Presentation Layer Design

Latihan :

1. Membuat sebuah fragment navigation bar yang dinamis. Dimana, aspek dinamis tersebut terdapat pada title navbar yang sebelumnya hanya bertuliskan APAP akan berubah sesuai dengan nama page yang sedang diakses.

Hal yang dilakukan :

- 1) Menambahkan variable title pada navbar di fragment.html

```
<nav th:fragment="navbar(title)" class="navbar navbar-expand-lg navbar-dark bg-dark">
```

- 2) Pada fragment.html, dalam tag body, dan didalam tag nav, terdapat tag a yang sebelumnya menghasilkan tulisan "APAP". Ambil nilai title yang didapat dari baris navbar diatas kepada tag a tersebut. Dan hapus tulisan "APAP"

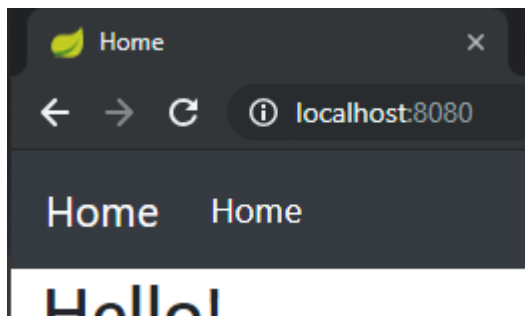
```
<a class="navbar-brand" href="#" th:text="${title}"></a>
```

- 3) Selanjutnya pada masing-masing view, ditambahkan nilai title disamping navbar pada tag nav

Contoh : pada view home

```
<body>  
<nav th:replace="fragments/fragment :: navbar('Home')"></nav>  
<div class="container-fluid">
```

- 4) Secara otomatis isi tag a (title navbar) akan berubah di tiap page sesuai dengan nilai title yang dimasukkan.

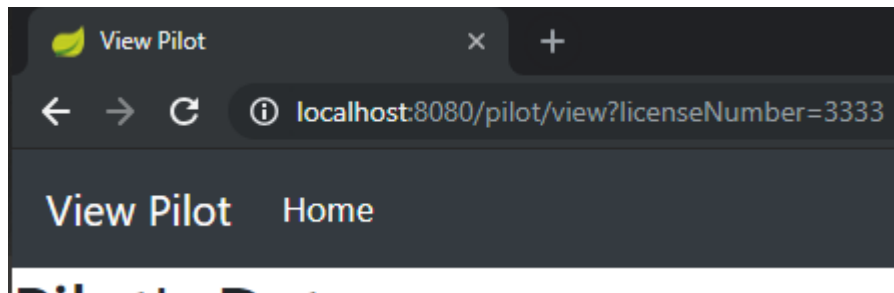


- 5) Contoh lain ketika di page view pilot

Pada view-pilot.html

```
<body>  
<nav th:replace="fragments/fragment :: navbar('View Pilot')"></nav>  
<h2>Pilot's Data</h2>
```

Pada tampilan :



2. Melakukan insert banyak flight sekaligus

Hal yang dilakukan :

- 1) Membuat method baru untuk menambah row (baris) di PilotController

```
@RequestMapping(value = "/flight/add/{licenseNumber}", params = {"addRow"}, method = RequestMethod.POST)
private String addRow(@ModelAttribute PilotModel pilot, Model model) {
    pilot.getPilotFlight().add(new FlightModel());
    model.addAttribute("pilot", pilot);
    return "addFlight";
}
```

- 2) Membuat method baru untuk remove (menghapus) row di PilotController

```
@RequestMapping(value = "/flight/add/{licenseNumber}", method = RequestMethod.POST, params = {"removeRow"})
private String removeRow(@ModelAttribute PilotModel pilot, final BindingResult bindingResult,
    final HttpServletRequest req, Model model) {
    final Integer rowId = Integer.valueOf(req.getParameter("removeRow"));
    pilot.getPilotFlight().remove(rowId.intValue());
    model.addAttribute("pilot", pilot);
    return "addFlight";
}
```

- 3) Memodifikasi method add pada PilotController

```
@RequestMapping(value = "/flight/add/{licenseNumber}", method = RequestMethod.GET)
private String addFlight(@PathVariable(value = "licenseNumber") String licenseNumber, Model model) {
    PilotModel pilot = pilotService.getPilotDetailByLicenseNumber(licenseNumber);
    ArrayList<FlightModel> list = new ArrayList<FlightModel>();
    list.add(new FlightModel());
    pilot.setPilotFlight(list);
    model.addAttribute("pilot", pilot);
    return "addFlight";
}
```

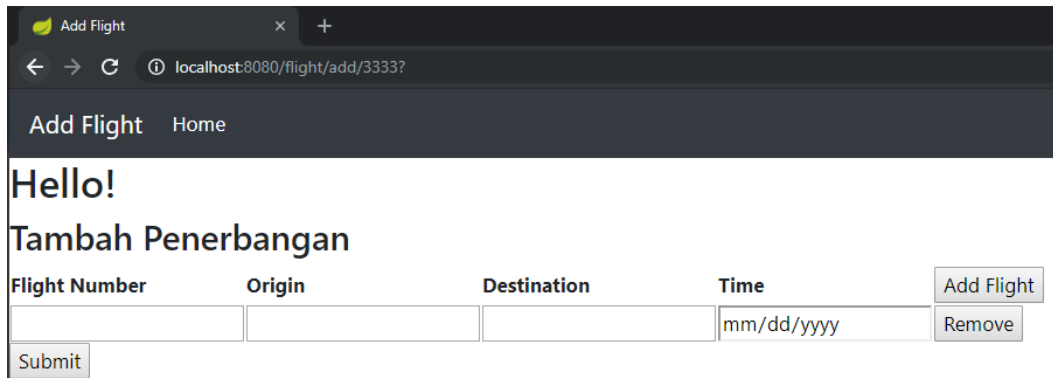
- 4) Membuat method addFlightSubmit yang akan dipakai untuk menyimpan penerbangan dengan nomor lisensi pilot yang ditentukan

```
@RequestMapping(value = "/flight/add/{licenseNumber}", method = RequestMethod.POST, params = {"save"})
private String addFlightSubmit (@ModelAttribute PilotModel pilot) {
    PilotModel currPilot = pilotService.getPilotDetailByLicenseNumber(pilot.getLicenseNumber());
    for (FlightModel flight : pilot.getPilotFlight()) {
        flight.setPilot(currPilot);
        flightService.addFlight(flight);
    }
    return "add";
}
```

- 5) Memodifikasi addFlight.html agar dapat add row, remove row, dan insert multiple

```
<h2>Hello!</h2>
<h3>Tambah Penerbangan</h3>
<form th:action="@{/flight/add/} + ${pilot.licenseNumber}" th:object="${pilot}" method="POST">
  <!-- <input type="hidden" th:field="**{pilot}" /> -->
  <input type="hidden" th:field="**{licenseNumber}" /></input>
  <table>
    <thead>
      <tr>
        <th>Flight Number</th>
        <th>Origin</th>
        <th>Destination</th>
        <th>Time</th>
        <th><button type="submit" name="addRow">Add Flight</button></th>
      </tr>
    </thead>
    <tbody>
      <tr th:each="flight, rowStat : *{pilotFlight}">
        <td><input type="text" th:field="**{pilotFlight[__${rowStat.index}__].flightNumber}" /></td>
        <td><input type="text" th:field="**{pilotFlight[__${rowStat.index}__].origin}" /></td>
        <td><input type="text" th:field="**{pilotFlight[__${rowStat.index}__].destination}" /></td>
        <td><input type="date" th:field="**{pilotFlight[__${rowStat.index}__].time}" /></td>
        <td><button type="submit" name="removeRow" th:value="${rowStat.index}">Remove</button></td>
      </tr>
    </tbody>
  </table>
  <button type="submit" name="save">Submit</button>
</form>
```

6) Sehingga akan menjadi seperti ini dalam tampilan



Add Flight Home

Hello!

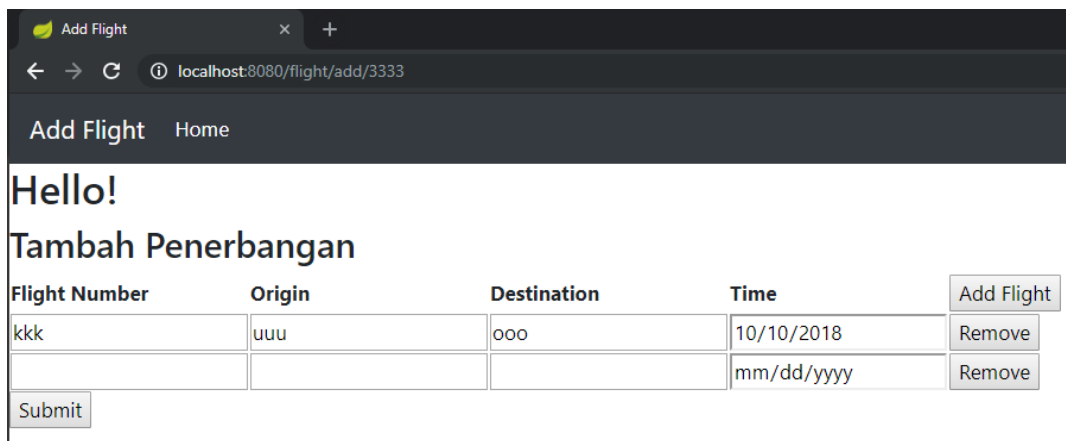
Tambah Penerbangan

Flight Number	Origin	Destination	Time	Add Flight
<input type="text"/>	<input type="text"/>	<input type="text"/>	<input type="text" value="mm/dd/yyyy"/>	Remove

Submit

7) Ketika ditambah row, jumlah row bertambah.

8) Namun pada case ini, hanya bisa dilakukan tambah row ketika baris yang tepat di atasnya telah berisi dan ada nilainya.



Add Flight Home

Hello!

Tambah Penerbangan

Flight Number	Origin	Destination	Time	Add Flight
kkk	uuu	ooo	10/10/2018	Remove
<input type="text"/>	<input type="text"/>	<input type="text"/>	<input type="text" value="mm/dd/yyyy"/>	Remove

Submit

RAFIQA AMINI MULIA
1606917834
APAP – C

Referensi

Tutorial: Thymeleaf + Spring. (2018). *Thymeleaf.org*. Retrieved 11 October 2018, from <https://www.thymeleaf.org/doc/tutorials/3.0/thymeleafspring.html#dynamic-fields>

Thymeleaf Page Layouts - Thymeleaf. (2018). *Thymeleaf.org*. Retrieved 11 October 2018, from <https://www.thymeleaf.org/doc/articles/layouts.html>