

Nama : Izzatunnisa Arwadi

NPM : 1806269726

Kelas : Ekstensi

1. Buatlah fragment navigation bar sehingga yang semula hanya bertuliskan "APAP" menjadi berubah-ubah (dinamis) sesuai dengan nama halaman yang sedang diakses. Pastikan Anda mengimplementasikannya pada setiap view yang ada pada project

- Fragment

```
<title th:text="${pageTitle}"></title>
```

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

Pada Fragment.html dilakukan penambahan code seperti di atas

- Method controller

```
@RequestMapping("/")
private String home(Model model) {
    String pageTitle = "APAP";
    model.addAttribute("pageTitle", pageTitle);
    return "home";
}
```

Pada method controller dilakukan pembuatan variable untuk mengisi tulisan yang dinamis dan variable tersebut dimasukkan ke dalam model.addAttribute

- Tampilan

The screenshot shows two side-by-side web views. The left view, titled 'APAP Home', features a 'Welcome!' message, a 'Tambah Pilot' button, a form to add a pilot, and a 'Cari Pilot' section with a search form. The right view, titled 'View Pilot Home', displays pilot details (License Number: 12345, Name: Caramia, Fly Hour: 20, Status: Pilot Junior) and a table of flights.

| No. | Flight Number | Origin | Destination | Time | Action |
|-----|---------------|---------|-------------|------------|--------|
| 1 | 202 | Jakarta | Granada | 2019-11-26 | Hapus |

Pada kedua tampilan di atas, tampilan pada fragment navigation bar telah berubah secara dinamis

2. Anda telah mempelajari cara meng-handle multiple checkbox untuk dapat melakukan banyak delete sekaligus. Pada latihan ini Anda diminta untuk mengimplementasi banyak insert untuk flight.

- Code

```
<h3>Tambah Penerbangan</h3>
<form th:action="@{/flight/add}" th:object="${pilot}" method="POST">
    <input type="hidden" th:field="**{licenseNumber}"></input>
    <div>
        <table>
            <thead>
                <tr>
                    <th>Flight Number</th>
                    <th>Origin</th>
                    <th>Destination</th>
                    <th>Time</th>
                </tr>
            </thead>
            <tbody>
                <tr>
                    <td><input class="form-control" type="text" th:field="**{pilotFlight[__${rowStat.index}__].flightNumber}" />
                    <td><input class="form-control" type="text" th:field="**{pilotFlight[__${rowStat.index}__].origin}" />
                    <td><input class="form-control" type="text" th:field="**{pilotFlight[__${rowStat.index}__].destination}" />
                    <td><input class="form-control" type="date" th:field="**{pilotFlight[__${rowStat.index}__].time}" />
                </td></td>
            </tr>
        </tbody>
    </table>
    <div>
        <button type="submit" name="addFlight">Add Flight</button>
    </div>
    <button type="submit" name="submit">Submit</button>
</form>
```

Dilakukan pengeditan pada viewFlight.html

```
@RequestMapping(value = "/flight/add/{licenseNumber}", method = RequestMethod.GET)
private String add(@PathVariable(value = "licenseNumber") String licenseNumber, Model model) {
    FlightModel flight = new FlightModel();
    PilotModel pilot = pilotService.getPilotDetailByLicenseNumber(licenseNumber);

    ArrayList<FlightModel> pilotFlight = new ArrayList<>();

    pilotFlight.add(flight);
    pilot.setPilotFlight(pilotFlight);
    flight.setPilot(pilot);

    String pageTitle = "Add Flight";

    model.addAttribute("pageTitle", pageTitle);
    model.addAttribute("pilot", pilot);
    model.addAttribute("flight", flight);

    return "addFlight";
}

@RequestMapping(value = "/flight/add", method = RequestMethod.POST, params = { "addFlight" })
private String addRow(@ModelAttribute PilotModel pilot, BindingResult bindingResult, Model model) {
    FlightModel flightBaru = new FlightModel();

    pilot.getPilotFlight().add(flightBaru);

    model.addAttribute("pilot", pilot);

    return "addFlight";
}

@RequestMapping(value = "/flight/add", method = RequestMethod.POST, params = { "addFlight" })
private String addRow(@ModelAttribute PilotModel pilot, BindingResult bindingResult, Model model) {
    FlightModel flightBaru = new FlightModel();

    pilot.getPilotFlight().add(flightBaru);

    model.addAttribute("pilot", pilot);

    return "addFlight";
}
```

Dilakukan penambahan method serta modifikasi isi method.

- Tampilan

Tutorial 05 - Thymeleaf.pdf x Add Flight x +

localhost:8080/flight/add/54321

Add Flight Home

Welcome

Tambah Penerbangan

| Flight Number | Origin | Destination | Time | |
|---------------------------------------|--------|-------------|------------|------------|
| 203 | Jeju | Osaka | 12/02/2020 | Add Flight |
| <input type="button" value="Submit"/> | | | | |

Dilakukan input data tambah penerbangan, ketika ingin menambahkan flight baru klik add flight

Welcome

Tambah Penerbangan

| Flight Number | Origin | Destination | Time | |
|---------------------------------------|--------|-------------|------------|------------|
| 203 | Jeju | Osaka | 12/02/2020 | Add Flight |
| 403 | Solo | Jakarta | 28/11/2019 | |
| <input type="button" value="Submit"/> | | | | |

Ketika add flight di klik akan ditampilkan new row setelah itu data di submit

View Pilot Home

Data Pilot

Pilot License Number: 54321

Name: Scarlett

Fly Hour: 24

Status: Pilot Junior

| No. | Flight Number | Origin | Destination | Time | Action |
|-----|---------------|----------|-------------|------------|---|
| 1 | 800 | Surabaya | Jakarta | 2020-02-11 | <input checked="" type="checkbox"/> Hapus |
| 2 | 203 | Jeju | Osaka | 2020-02-11 | <input checked="" type="checkbox"/> Hapus |
| 3 | 403 | Solo | Jakarta | 2019-11-27 | <input checked="" type="checkbox"/> Hapus |

Flights

Kedua data yang telah diinputkan telah masuk ke dalam database