

Tutorial

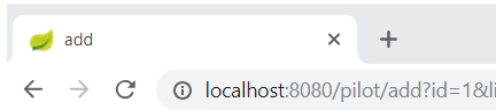
Membuat Controller dan Fungsi Add

Jalankan program dan buka

localhost:8080/pilot/add?id=1&licenseNumber=1234&name=Jono&flyHour=20

- **Pertanyaan 1:** apakah hasilnya? Jika error, tuliskan penjelasan Anda.

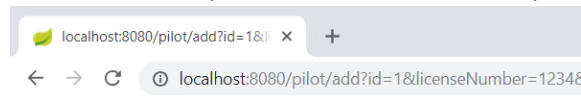
Jawaban : Hasil bisa dijalankan dengan baik.



Data Berhasil Ditambahkan

- localhost:8080/pilot/add?id=1&licenseNumber=1234&name=Jono
Pertanyaan 2: apakah hasilnya? Jika error, tuliskan penjelasan Anda

Jawaban : Hasilnya akan error, karena terdapat value parameter yang tidak dimasukkan



Whitelabel Error Page

This application has no explicit mapping for /error, so you are seeing this

Wed Sep 26 15:35:03 ICT 2018

There was an unexpected error (type=Bad Request, status=400).

Required int parameter 'flyHour' is not present

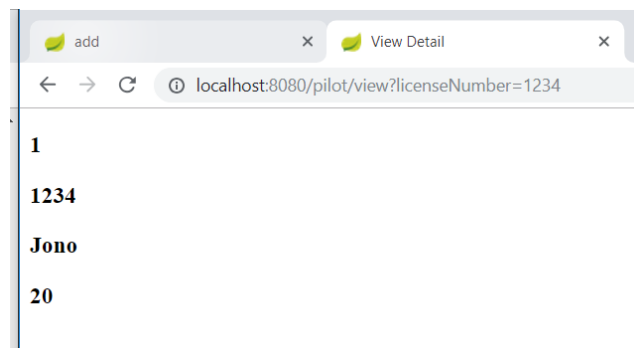
Method View By licenseNumber

localhost:8080/pilot/add?id=1&licenseNumber=1234&name=Jono&flyHour=20

localhost:8080/pilot/view?licenseNumber=1234

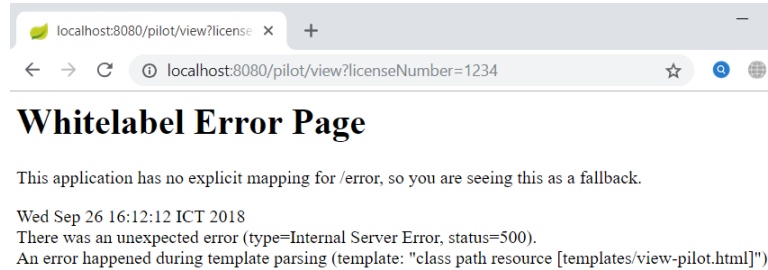
Pertanyaan 3: apakah data Pilot tersebut muncul? Jika tidak, mengapa?

Jawaban : Data Pilot akan muncul.



Coba matikan program dan jalankan kembali serta buka
localhost:8080/pilot/view?licenseNumber=1234

Pertanyaan 4: apakah data Pilot tersebut muncul? Jika tidak, mengapa?



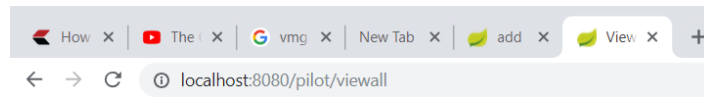
Jawaban: Tidak muncul, karena sebelumnya tidak dilakukan pemasukkan data.

Method View All

localhost:8080/pilot/add?id=1&licenseNumber=1234&name=Jono&flyHour=20

localhost:8080/pilot/viewall

Pertanyaan 5: apakah data Pilot tersebut muncul?



1

1234

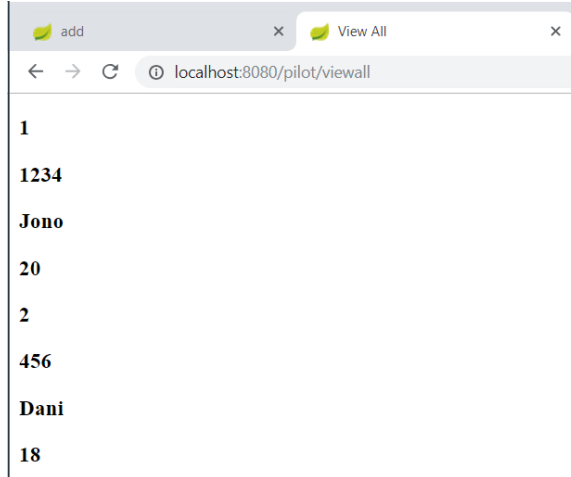
Jono

20

Coba tambahkan data Pilot lainnya dengan NPM yang berbeda, lalu buka localhost:8080/pilot/viewall

Pertanyaan 6: Apakah semua data Pilot muncul?

Jawab : Ya semua data muncul.



Latihan

1. Pada PilotController tambahkan sebuah method view Pilot dengan menggunakan Path Variable. Misalnya, Anda ingin memasukkan data seorang Pilot yang memiliki licenseNumber 101, untuk melihat data yang baru dimasukkan tersebut dapat mengakses halaman

localhost:8080/pilot/view/license-number/101.

Jika licenseNumber tidak diberikan atau tidak ditemukan kembalikan halaman error yang berisi informasi bahwa nomor licenseNumber kosong atau tidak ditemukan.

```

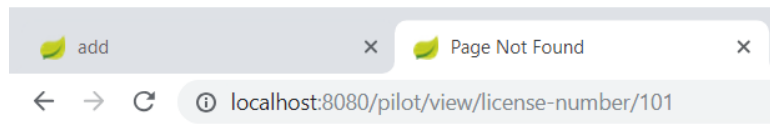
@RequestMapping(value = { "/pilot/view/license-number/", "/pilot/view/license-number/{licenseNumber}" })
public String viewPath(@PathVariable Optional<String> licenseNumber, Model model) {
    String temp = "error";

    if (licenseNumber.isPresent()) {
        PilotModel archive = pilotService.getPilotDetailByLicenseNumber(licenseNumber.get());
        if (archive == null) {
            model.addAttribute("erorinfo", "License Number Tidak ditemukan");
        }

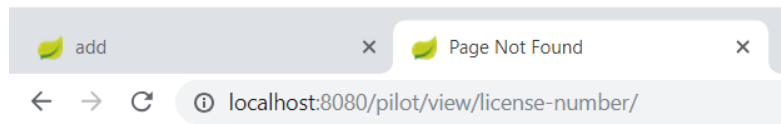
        else {
            model.addAttribute("pilot", archive);
            temp = "view-pilot";
        }
    }

    else {
        model.addAttribute("erorinfo", "License Number Tidak dimasukkan");
    }
    return temp;
}

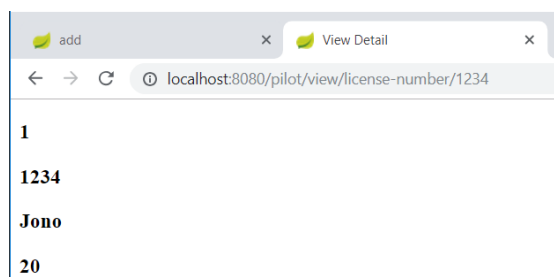
```



License Number Tidak ditemukan



License Number Tidak dimasukkan



2. Tambahkan fitur untuk melakukan update flyHour dari Pilot berdasarkan licenseNumber. Misalnya, setelah melakukan add Pilot pada soal nomor 1, cobalah untuk mengubah flyHour objek Pilot tersebut menjadi 100 dengan mengakses halaman `localhost:8080/pilot/update/license-number/101/fly-hour/100`. Tampilkan sebuah halaman yang memberikan informasi bahwa data tersebut telah berhasil diubah. Jika id tidak

diberikan atau tidak ditemukan kembalikan halaman error yang berisi informasi bahwa id kosong atau tidak ditemukan dan proses update dibatalkan

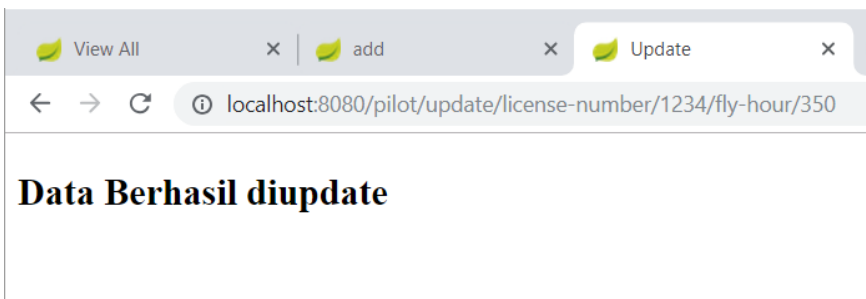
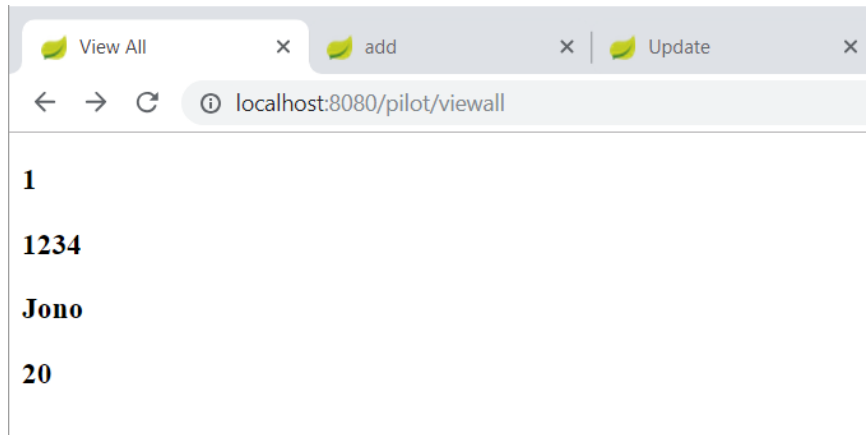
```
@RequestMapping("/pilot/update/license-number/{licenseNumber}/fly-hour/{flyHour}")
public String updatePath (@PathVariable Optional<String>licenseNumber,
    @PathVariable Optional<Integer>flyHour,Model model) {
    if (licenseNumber.isPresent() & flyHour.isPresent()) {
        PilotModel archive = pilotService.getPilotDetailByLicenseNumber(licenseNumber.get());
        if (archive == null) {
            model.addAttribute("updateinfo", "License Number Tidak ditemukan, Data Batal diupdate !");
        }

        else {
            archive.setFlyHour(flyHour.get());
            model.addAttribute("updateinfo", "Data Berhasil diupdate");
        }
    }

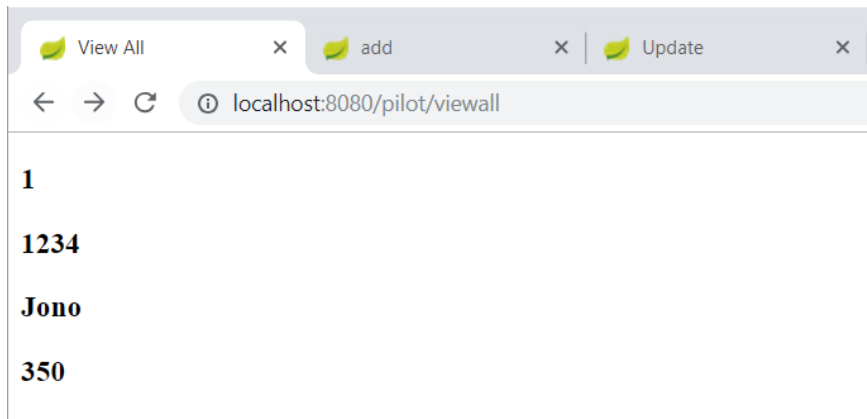
    else {
        model.addAttribute("updateinfo", "License Number Tidak dimasukkan, Data Batal diupdate !");
    }

    return "infoupdate";
}
```

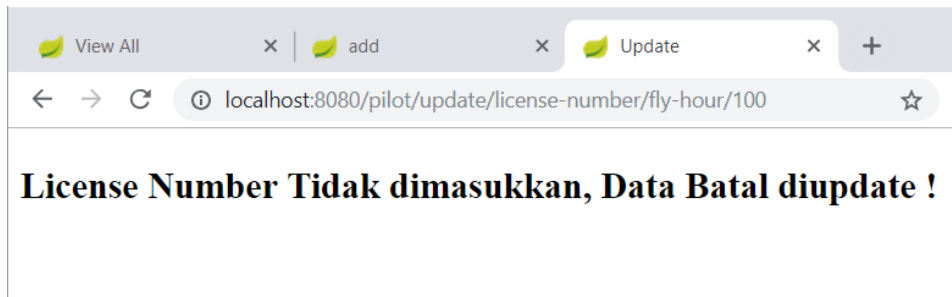
- View awal



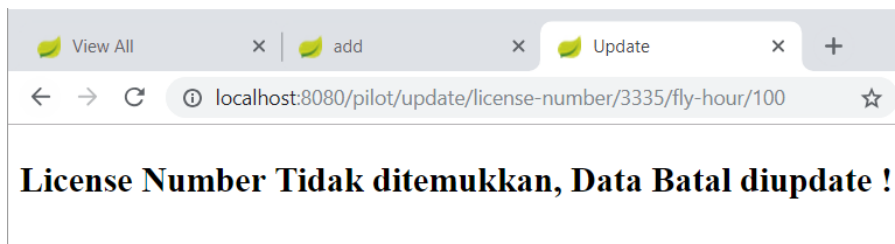
- **View Setelah di update**



- **Jika license-number tidak dimasukkan**



- **Jika license-number tidak ditemukan**



3. Tambahkan fitur untuk melakukan delete Pilot berdasarkan id. Misalnya, setelah melakukan perintah pada soal nomor 1 dan 2, cobalah untuk melakukan delete data tersebut dengan mengakses halaman localhost:8080/pilot/delete/id/1. Tampilkan sebuah halaman yang memberikan informasi bahwa data tersebut telah berhasil dihapus. Jika id tidak diberikan atau tidak ditemukan kembalikan halaman error yang berisi informasi bahwa id kosong atau tidak ditemukan dan proses delete dibatalkan.

```

@RequestMapping(value = {"pilot/delete/id/{id}", "pilot/delete/id/"})
public String deletePath (@PathVariable Optional<String> id , Model model) {
    List<PilotModel> archive = pilotService.getListPilot();
    if (id.isPresent()) {
        PilotModel objekData= pilotService.getPilotDetailById(id.get());
        if (objekData == null) {
            model.addAttribute("infodelete", "[Id Tidak ditemukan, Data Batal diHapus!"]);
        }

        else {
            archive.remove(objekData);
            model.addAttribute("infodelete", "Data Berhasil dihapus");
        }
    }

    else {
        model.addAttribute("infodelete", "Id Tidak dimasukkan, Data Batal diHapus !");
    }

    return "deleteinfo";
}

```

