Fildza Larissa – 1506727495
APAP – B

METHOD SELECT STUDENT

```
@Override
public StudentModel selectStudent(String npm) {
    for (int i = 0; i < studentList.size(); i++) {
        String studentNpm = studentList.get(i).getNpm();
        if (studentNpm.equals(npm)) {
            return studentList.get(i);
        }
    }
    return null;
}
```

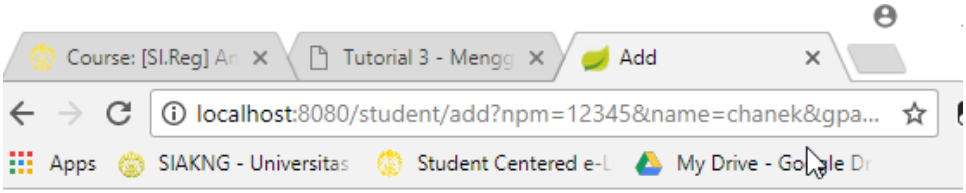
Method diatas merupakan method untuk mengambil student berdasarkan npm yang telah diberikan. Pertama karena kumpulan student diletakan didalam arraylist maka pertama kali akan mengiterasi arraylist, setelah itu menyamakan npm student pada index ke i dengan npm yang dimasukan dalam parameter, jika sama maka akan dikembalikan student tersebut namun jika tidak sama akan dikembalikan nilai null.

MEMBUAT SERVICE

Buka localhost:8080/student/add?npm=12345&name=chanek&gpa=3.43

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

Jawab :



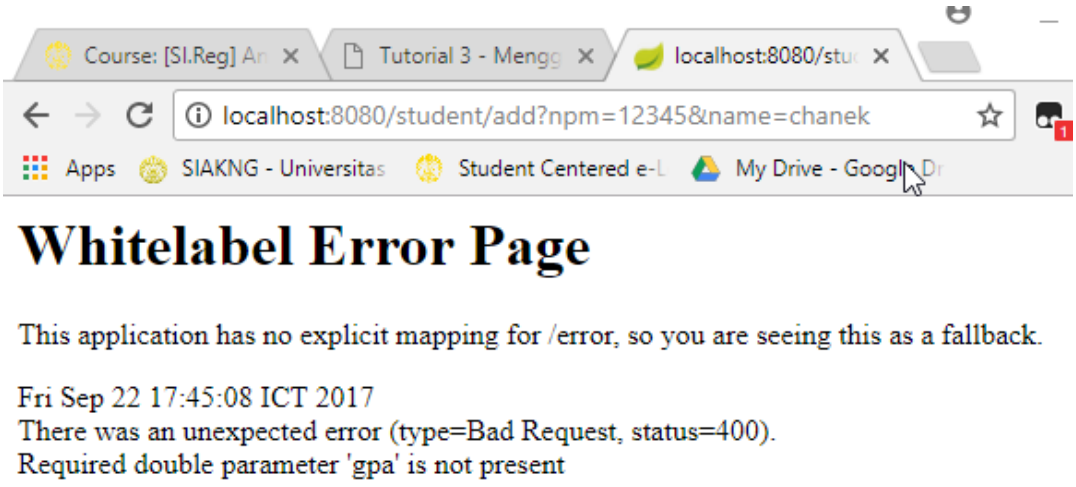
Data berhasil ditambahkan

Buka localhost:8080/student/add?npm=12345&name=chanek

Pertanyaan 2: apakah hasilnya? Jika error, tuliskan penjelasan Anda.

Jawab : Whitelabel Error Page dengann tipe error Bad Request dan status 400

Hal ini terjadi karena pada parameter yang dimasukan tidak terdapat nilai gpa sedangkan untuk RequestParam GET dibutuhkan parameter gpa dan controller hal itu harus dituliskan karena requirednya adalah true.

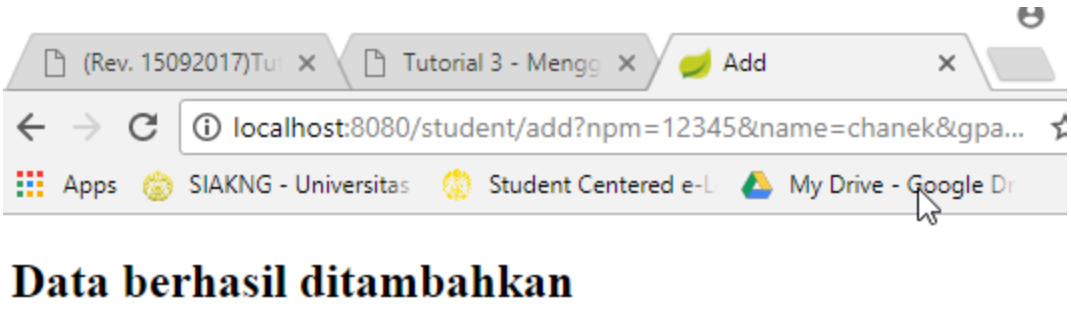


METHOD VIEW BY NPM

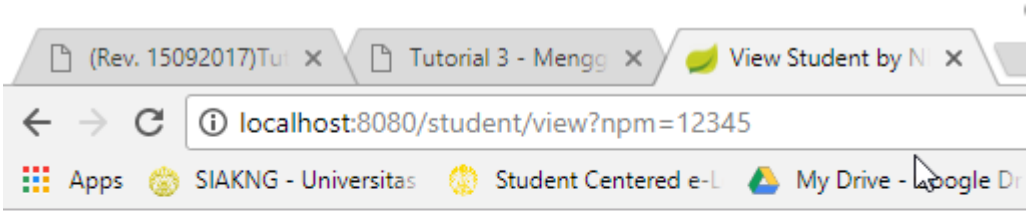
Jalankan dan buka localhost:8080/student/add?npm=12345&name=chanek&gpa=3.43 lalu buka localhost:8080/student/view?npm=12345

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

Jawab : Ya data muncul



Fildza Larissa – 1506727495
APAP – B



NPM =12345

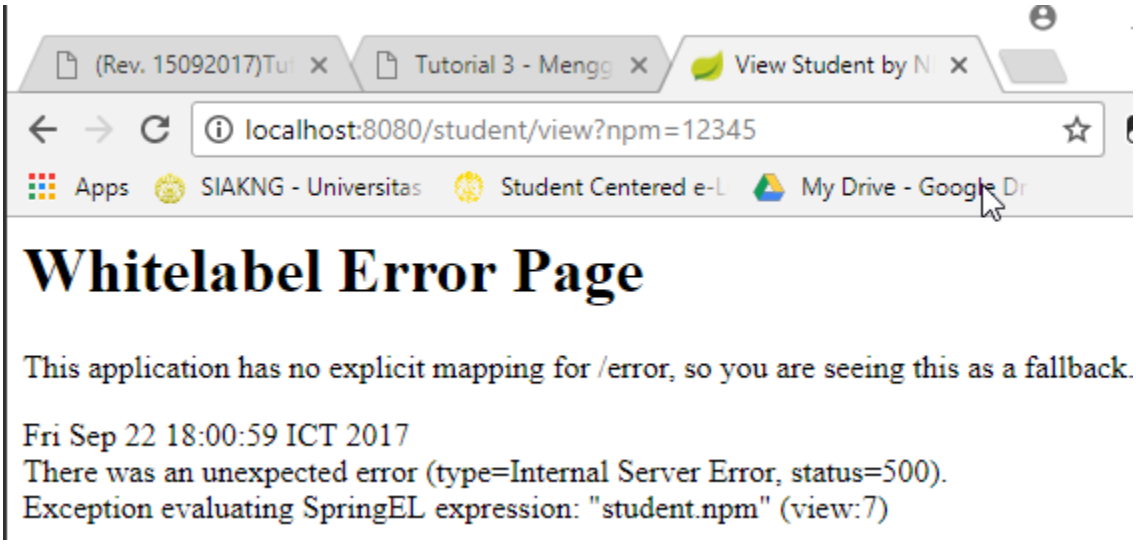
Name =chanek

GPA =3.43

matikan program dan jalankan kembali serta buka localhost:8080/student/view?npm=12345

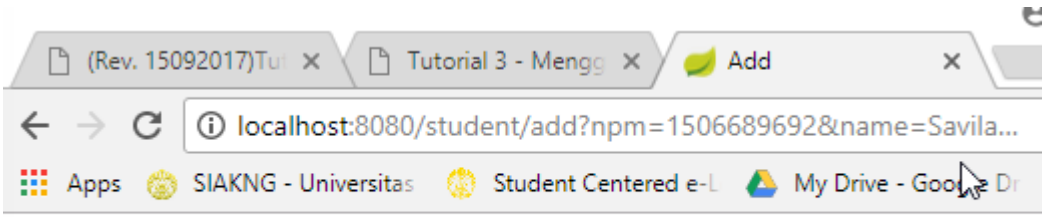
Pertanyaan 4: apakah data Student tersebut muncul? Jika tidak, mengapa?

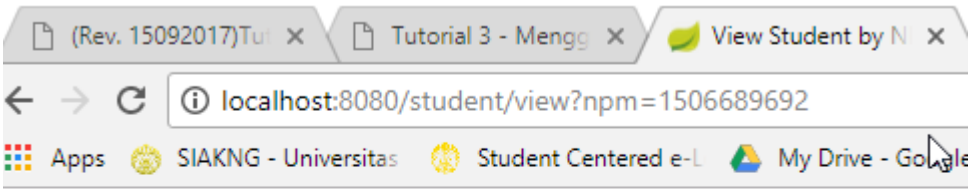
Jawab : Tidak, karena program dimatikan terlebih dahulu setelah itu di run kembali namun tidak melakukan add pada data sedangan kita hanya dapat melihat data setelah mengadd data dan setelah itu view akan mengambil data dari add yang telah dilakukan



Tambahkan data student lainnya dengan NPM yang berbeda

Fildza Larissa – 1506727495
APAP – B

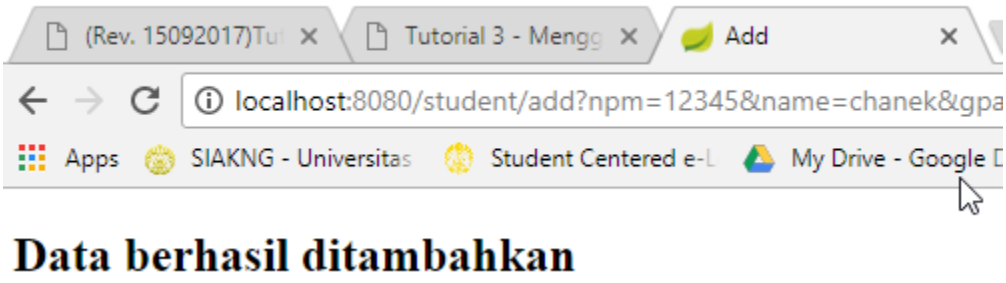




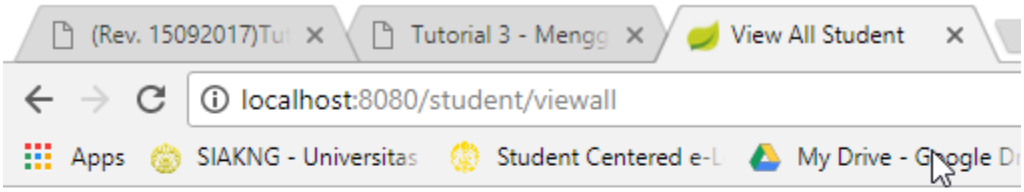
METHOD VIEW ALL

Jalankan program dan buka localhost:8080/student/add?npm=12345&name=chanek&gpa=3.43 lalu buka localhost:8080/student/viewall

Pertanyaan 5: apakah data Student tersebut muncul?



Fildza Larissa – 1506727495
APAP – B



No. 1

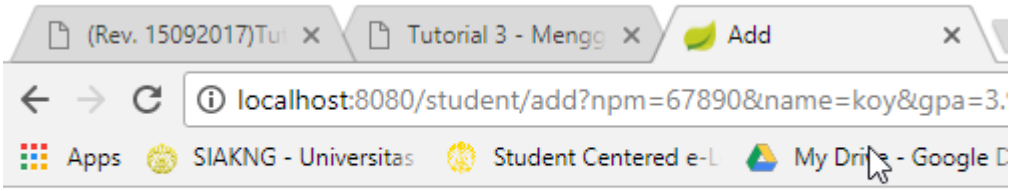
NPM = 12345

Nama = chanek

GPA = 3.43

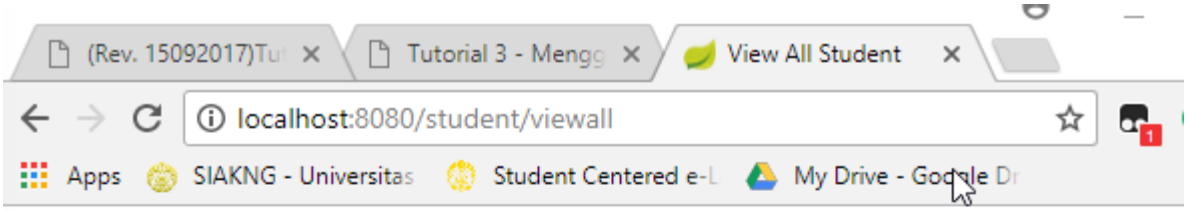
Coba tambahkan data Student lainnya dengan NPM yang berbeda, lalu buka localhost:8080/student/viewall

Pertanyaan 6: Apakah semua data Student muncul?



Data berhasil ditambahkan

Fildza Larissa – 1506727495
APAP – B



No. 1

NPM = 12345

Nama = chanek

GPA = 3.43

No. 2

NPM = 67890

Nama = koy

GPA = 3.99

LATIHAN

1. Pada StudentController tambahkan sebuah method view Student dengan menggunakan Path Variable. Misalnya, Anda ingin memasukkan data seorang Student yang memiliki NPM 14769, untuk melihat data yang baru dimasukkan tersebut dapat mengakses halaman localhost:8080/student/view/14769.
   Jika nomor NPM tidak diberikan atau tidak ditemukan kembalikan halaman error yang berisi informasi bahwa nomor NPM kosong atau tidak ditemukan.

   a. Membuat method view dengan menggunakan path

```
@RequestMapping ("/student/view/{npm}")
public String viewpath(Model model, @PathVariable Optional<String> npm) {
    if(npm.isPresent()) {
        StudentModel student = studentService.selectStudent(npm.get());
        if (student != null) {
            model.addAttribute("student", student);
            return "view";
        }else {
            return "errorNotFound";
        }

    }
    return "errorEmpty";
}
```

Method diatas mengecek apakah variable path telah dimasukan, jika npm telah dimasukan pada variable path maka mengecek apakah npm tersebut terdapat pada data, jika iya maka akan mereturn ke view.html jika tidak ke html errorNotfound. Jika ternyata npm tidak dimasukkan pada viarble oath maka akan mengembalikan ke html errorEmpty.

b. Membuat html jika npm tidak ditemukan

StudentCont... | errorEmpty.html | errorNotFoun...

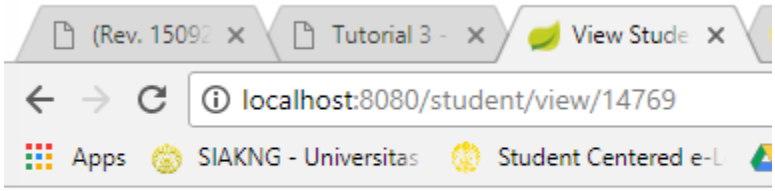
```
<!DOCTYPE html>
<html xmlns:th="http://www.thymeleaf.org">
    <head>
        <title>View Student by NPM</title>
    </head>
    <body>
        <h3>Nomor NPM Tidak Ditemukan</h3>
    </body>
</html>
```

c. Membuat html jika npm tidak dimasukkan pada parameter

StudentCont... | errorEmpty.html | errorNotFoun...

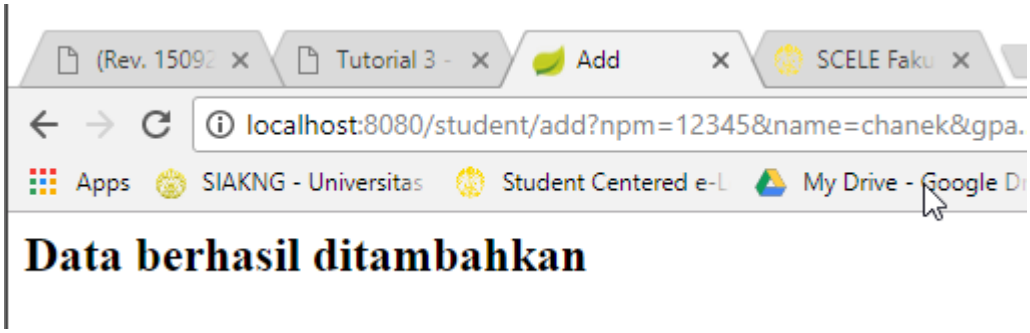
```
<!DOCTYPE html>
<html xmlns:th="http://www.thymeleaf.org">
    <head>
        <title>View Student by NPM</title>
    </head>
    <body>
        <h3>Nomor NPM Kosong</h3>
    </body>
</html>
```

d. Localhost:8080/student/view/14769 (sebelum menambahkan student baru)
Maka akan memunculkan nomor NPM tidak ditemukan karena belum ada data yang dimasukkan dan tidak cocok dengan semua data.

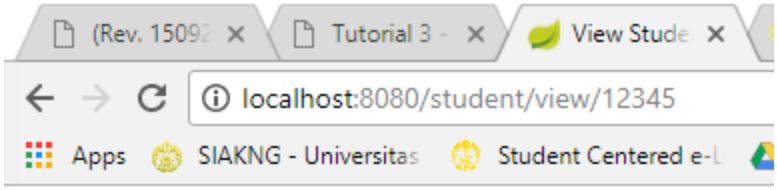


Nomor NPM Tidak Ditemukan

e. Memasukan data student dengan npm 12345



f. localhost:8080/student/view/12345

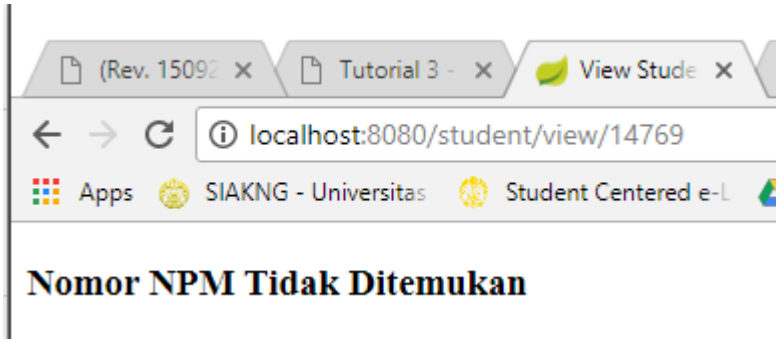


NPM = 12345

Name = chanek

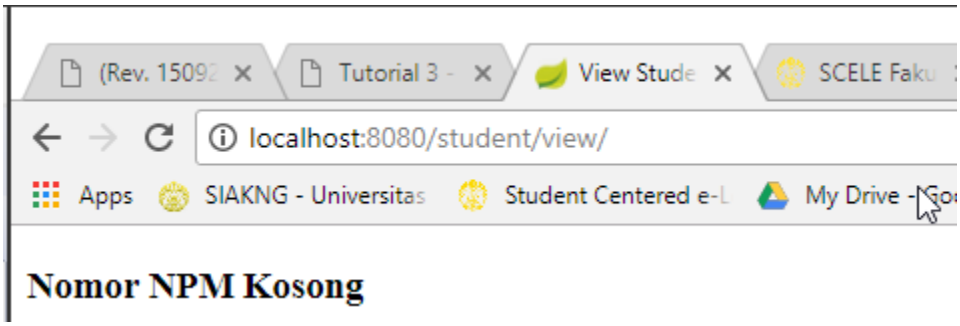
GPA = 3.43

g. Localhots:8080/student/view14769
Akan memunculkan nomor npm tidak ditemukan karena npm tidak cocok dengan npm student-student yang telah dimasukkan.

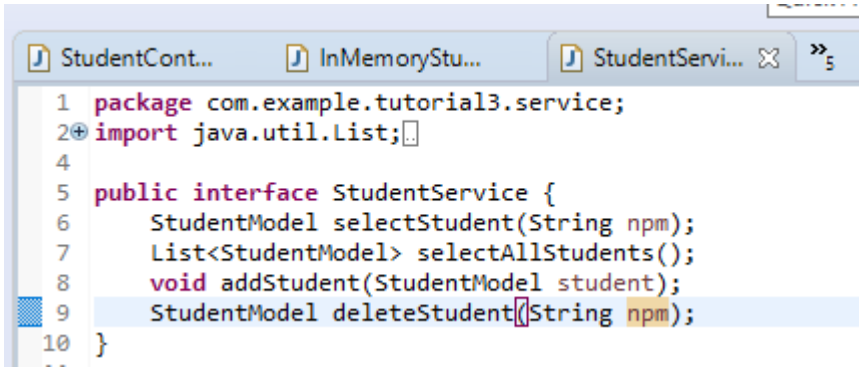


h. localhost:8080/student/view/

Akan memunculkan npm kosong karena npm yang dimasukkan tidak ada.



2. Tambahkan fitur untuk melakukan delete Student berdasarkan NPM. Misalnya, setelah melakukan add Student pada soal nomor 1, cobalah untuk melakukan delete data tersebut dengan mengakses halaman localhost:8080/student/delete/14769. Tampilkan sebuah halaman yang memberikan informasi bahwa data tersebut telah berhasil dihapus.
   Jika nomor NPM tidak diberikan atau tidak ditemukan kembalikan halaman error yang berisi informasi bahwa nomor NPM kosong atau tidak ditemukan dan proses delete dibatalkan.
   a. Menambahkan method delete di interface agar dapat melakukan manipulasi kelas Student.



b. Pada kelas InMemoryStudentService melakukan implementasi method delete. Melakukan iterasi pada seluruh data student yang ada pada arraylist, jika menemukan student ditemukan dengan npm yang dimasukkan maka akan mendelete student tersebut dari arraylist. Method akan mengembalikan student yang dicari sesuai npm.

```
@Override
public StudentModel deleteStudent(String npm) {
    StudentModel student = selectStudent(npm);
    if(student != null) {
        studentList.remove(student);
    }
    return student;
}
```

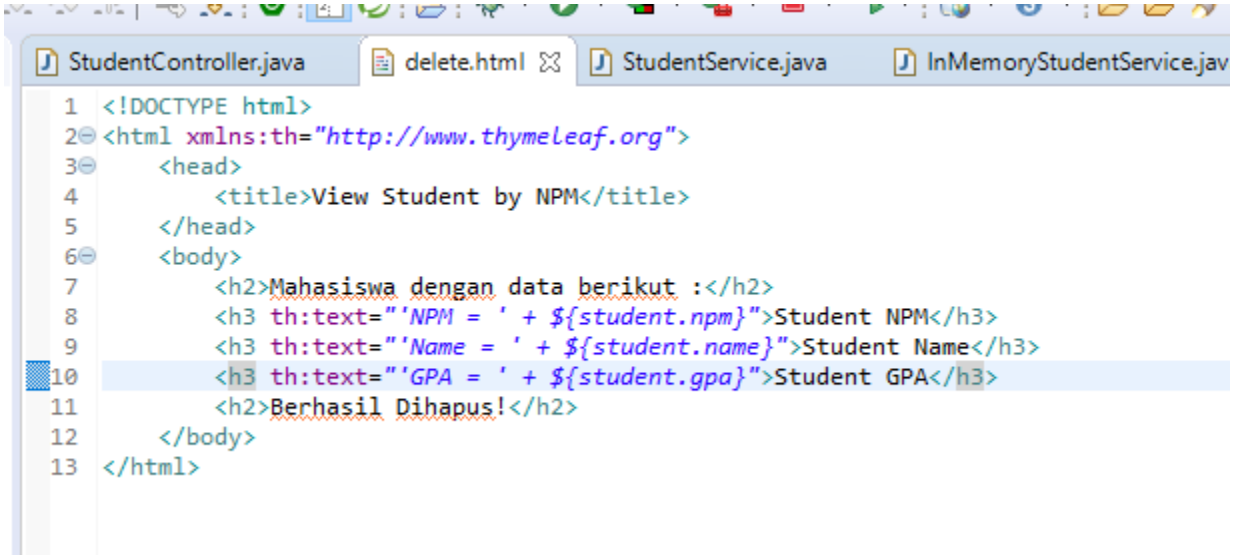
c. Membuat Controller dan fungsi delete

Method diatas mengecek apakah variable path telah dimasukan, jika npm telah dimasukan pada variable path maka mengecek apakah ada di data(ke method delete di InMemoryStudentService), jika variable student yang dihasilkan tidak null ke html delete, jika null ke html errorNotfound. Jika ternyata npm tidak dimasukkan pada variable path maka akan mengembalikan ke html errorEmpty.

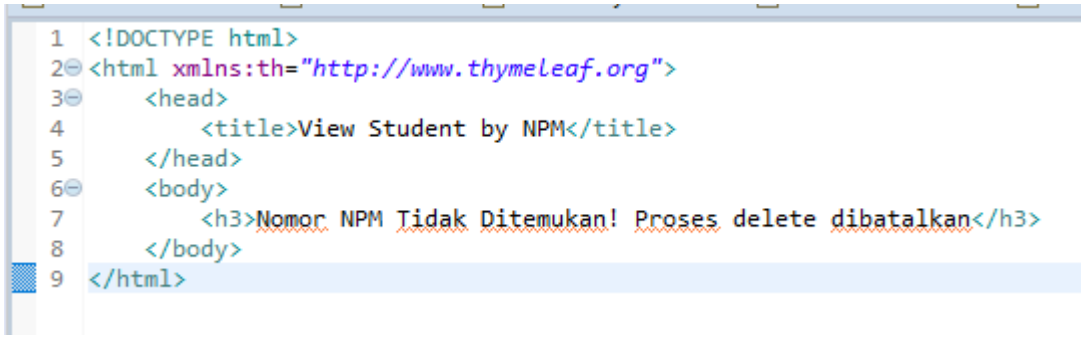
```java
@RequestMapping (value = {"/student/delete", "/student/delete/{npm}"})
public String deletepath(Model model, @PathVariable Optional<String> npm) {
    if(npm.isPresent()) {
        StudentModel student = studentService.deleteStudent(npm.get());
        if (student != null) {
            model.addAttribute("student", student);
            return "delete";
        }else {
            return "errorDelNotFound";
        }
    }else {
        return "errorEmpty";
    }

}
```

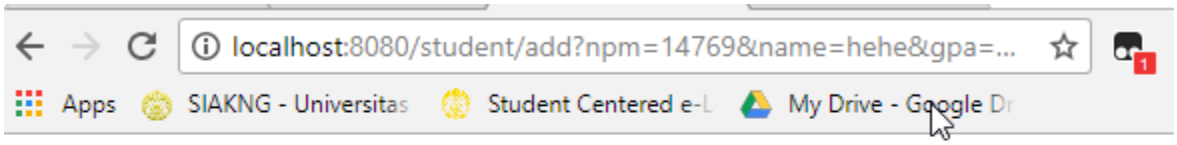
d. Membuat html delete



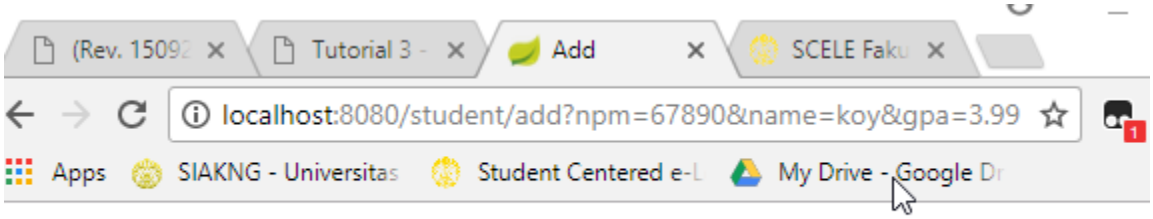
e. Membuat html jika npm tidak ditemukan



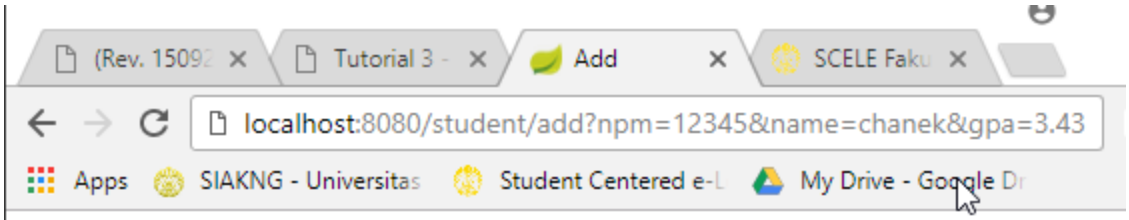
f. Melakukan add student dengan npm 14769



Data berhasil ditambahkan

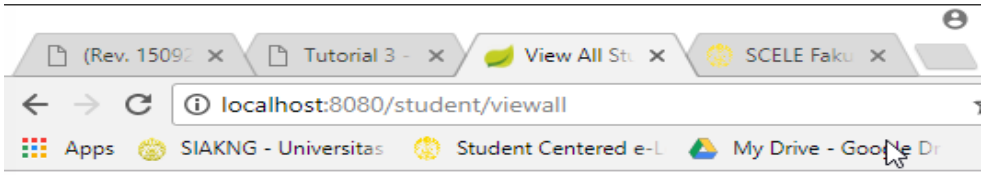


Data berhasil ditambahkan



Data berhasil ditambahkan

g. View data yang telah ditambahkan



No. 1

NPM = 67890

Nama = koy

GPA = 3.99

No. 2

NPM = 14769

Nama = hehe

GPA = 3.88

No. 3

NPM = 12345

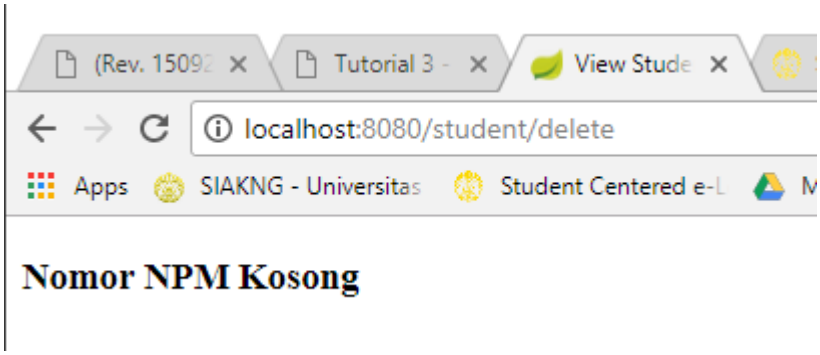
Nama = chanek

GPA = 3.43

---

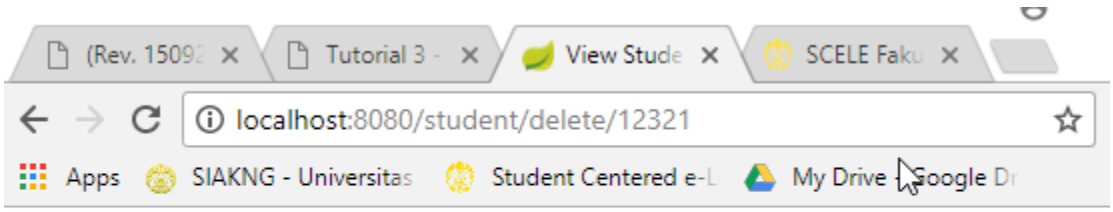
h. Localhost:8080/student/delete

Tidak memasukan nilai di parameter



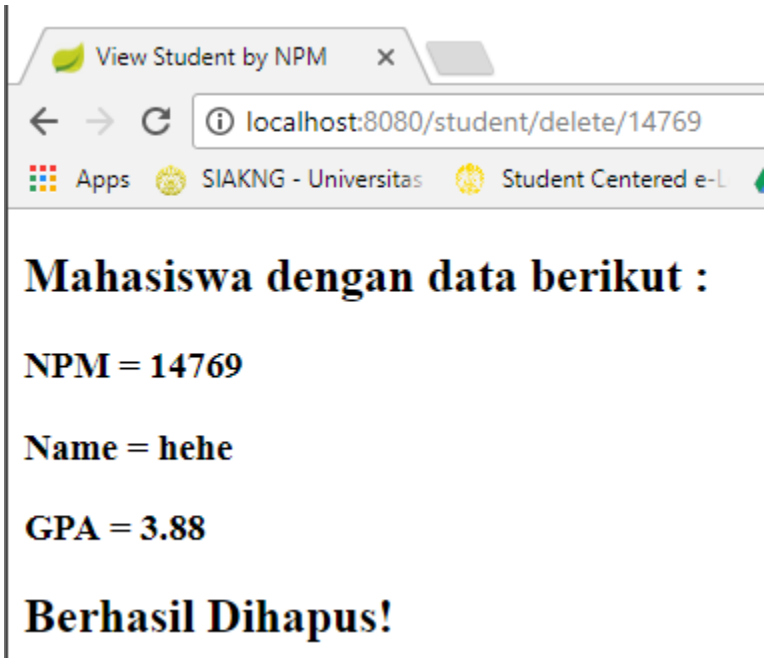
i. Localhost:8080/student/delete/12321

Npm tidak ada di data



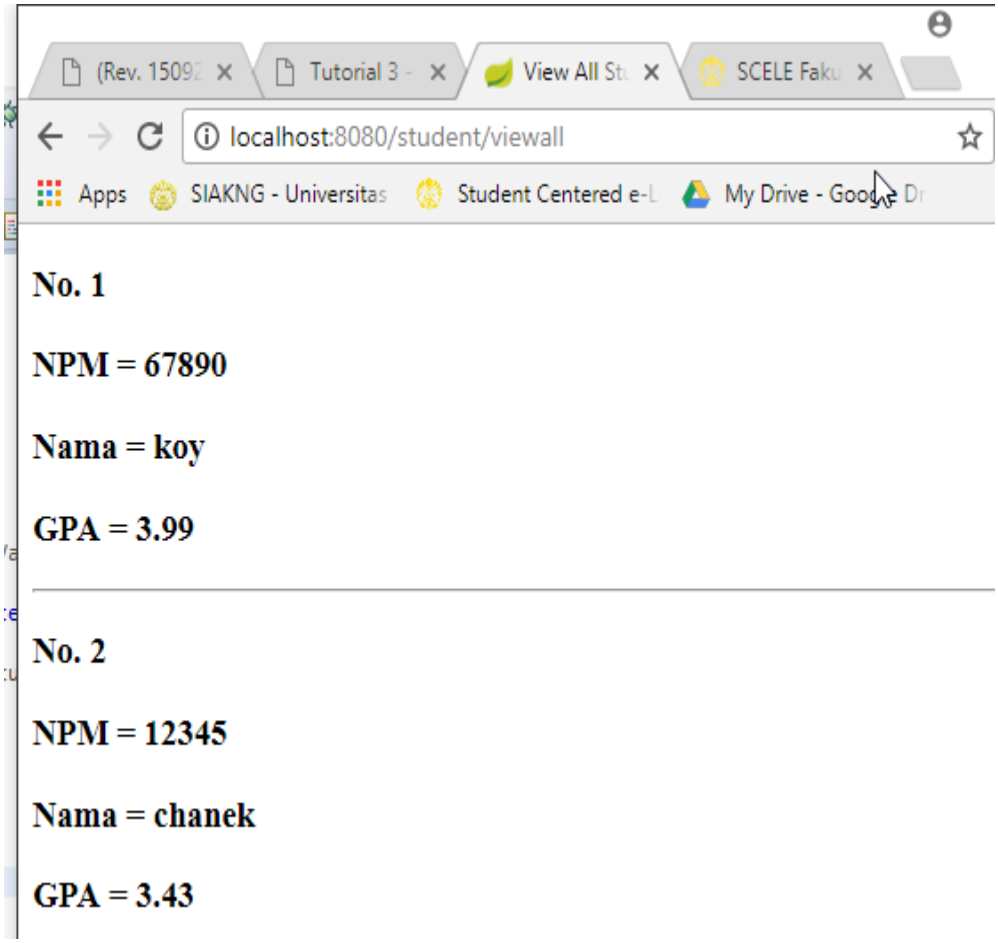
Nomor NPM Tidak Ditemukan! Proses delete dibatalkan

j. Mendelete data : Localhots:8080/student/delete/14769



k. View all data

Fildza Larissa – 1506727495
APAP – B



RINGKASAN MATERI

Pada tutorial 3 ini saya mempelajari mengenai model dan service. Model adalah object yang akan merepresentasikan data akan suatu hal dan pada object tersebut berisi data-data yang berhubungan dengan object yang dibuat. Pada tutorial 3 ini modelnya adalah mahasiswa. Kelas mahasiswa ini akan menyimpan data-data yang berkaitan dengan mahasiswa seperti name, gpa, dan npm. Selain itu, terdapat service yaitu penghubung antara data dengan controller. Pada service terdapat interface yang berisi method apa saja yang akan diimplentasikan seperti pada tutorial kali ini adalah selectstudent, selectallstudent, addstudent, dan deletestudent. Method-method ini nantinya akan digunakan pada kelas InMemoryStudentService yang berisi method-method yang ada pada interface dengan implementasinya.

Selanjutnya controller berfungsi untuk mendapatkan data yang dimasukkan pada url, setelah itu data tersebut akan dibaca dan dilakukan method-method yang ada di service. Contohnya pada tutorial 3 ini adalah memasukkan data student ke dalam arraylist yang ada pada DAO. Setelah itu view berfungsi untuk menampilkan apakah data benar-benar tersimpan. Selain itu, data akan dihapus jika program dimatikan.