

## Membuat Class Model

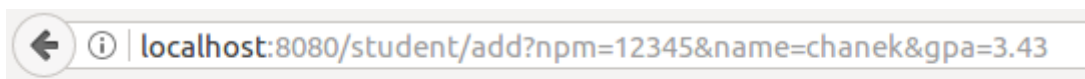
1. Method selectStudent setelah berhasil di implementasikan:

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

Pada metode berikut, mengiterasikan arraylist untuk menemukan mahasiswa dengan NPM yang sama dengan yang dimasukkan dalam parameter, jika ada yang sama kemudian *return* dari mahasiswa yang didapatkan, jika tidak maka NULL menandakan bahwa tidak ada mahasiswa tersebut.

## Membuat Service

1. localhost:8080/student/add?npm=12345&name=chanek&gpa=3.43  
Pertanyaan 1: apakah hasilnya? Jika error, tuliskan penjelasan Anda.



## Data berhasil ditambahkan

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

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



## Whitelabel Error Page

This application has no explicit mapping for /error, so you are seeing this as a fallback.

Sat Sep 23 00:46:58 WIB 2017

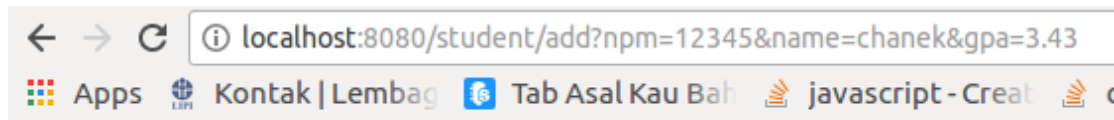
There was an unexpected error (type=Not Found, status=404).

No message available

Terjadi Whitelabel Error Page (type=Bad Request, status=400), dibutuhkan parameter 'gpa' untuk RequestParam GET.

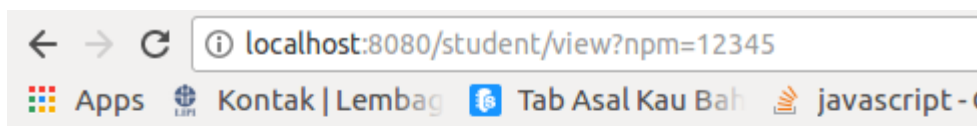
## Method View by NPM

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



## Data berhasil ditambahkan

lalu buka  
localhost:8080/student/view?npm=12345



**NPM = 12345**

**Name = chanek**

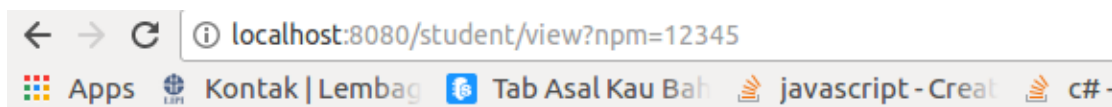
**GPA = 3.43**

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

Data student yang dimaksud muncul sesuai dengan yang sebelumnya  
dimasukkan pada parameter

**npm=12345&name=chanek&gpa=3.43**

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



## Whitelabel Error Page

This application has no explicit mapping for /error, so you are seeing this as a fallback.

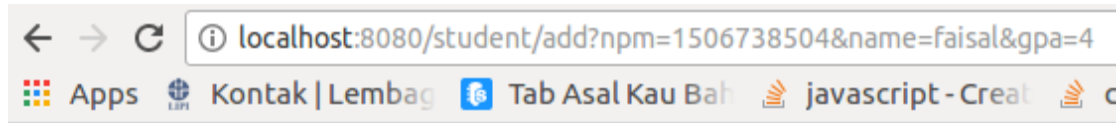
Sat Sep 23 21:33:40 WIB 2017

There was an unexpected error (type=Internal Server Error, status=500).

Exception evaluating SpringEL expression: "student.npm" (view:7)

Pertanyaan 4: apakah data Student tersebut muncul? Jika tidak, mengapa?  
Tidak. Data student ada pada session setiap kali springboot dijalankan.  
Ketika springboot dimatikan, maka session akan diperbaharui sepenuhnya.

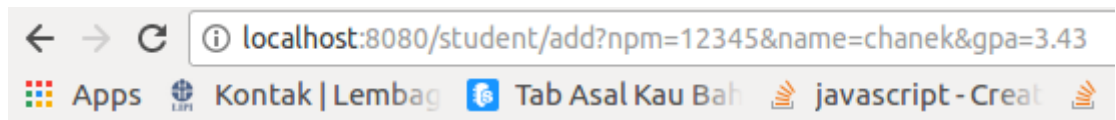
3. Coba tambahkan data Student lainnya dengan NPM yang berbeda.



**Data berhasil ditambahkan**

## Method View All

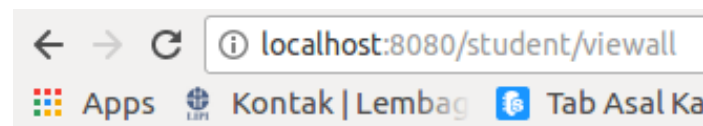
1. Jalankan program dan buka localhost:8080/student/add?npm=12345&name=chanek&gpa=3.43



**Data berhasil ditambahkan**

lalu buka localhost:8080/student/viewall

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



**No. 1**

**NPM = 12345**

**Name = chanek**

**GPA = 3.43**

---

**No. 2**

**NPM = 1506738504**

**Name = faisal**

**GPA = 4.0**

---

Pertanyaan 6: Apakah semua data Student muncul?  
**Muncul**

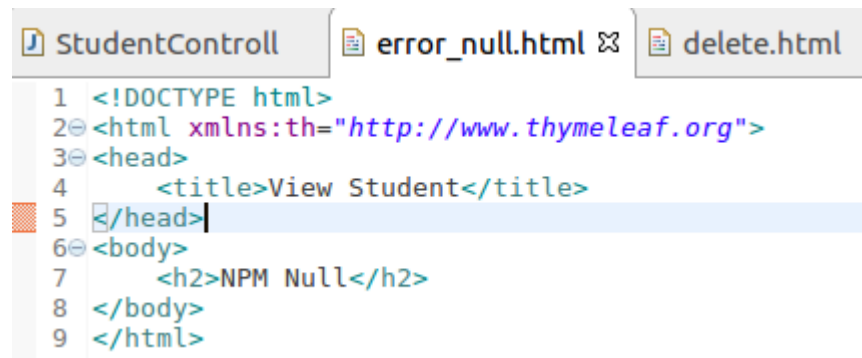
## Latihan

1. Pada StudentController tambahkan sebuah method view Student dengan menggunakan Path Variable.

Method yang ditambahkan

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

membuat halaman error\_null.html dan error\_notFound.html

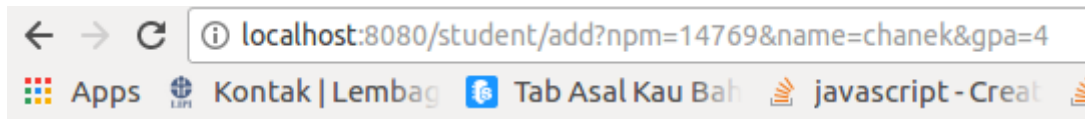


```
1 <!DOCTYPE html>  
2 <html xmlns:th="http://www.thymeleaf.org">  
3 <head>  
4     <title>View Student</title>  
5 </head>  
6 <body>  
7     <h2>NPM Null</h2>  
8 </body>  
9 </html>
```



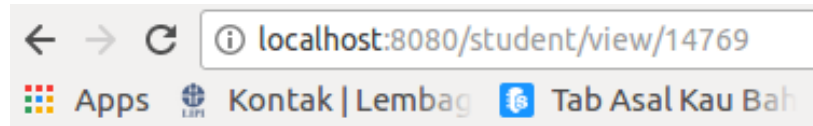
```
1 <!DOCTYPE html>  
2 <html xmlns:th="http://www.thymeleaf.org">  
3 <head>  
4     <title>View Student</title>  
5 </head>  
6 <body>  
7     <h2>NPM Not Found</h2>  
8 </body>  
9 </html>
```

Tambahkan npm 14769



## Data berhasil ditambahkan

Cek apakah student dengan npm tersebut sudah masuk dengan pathVariable

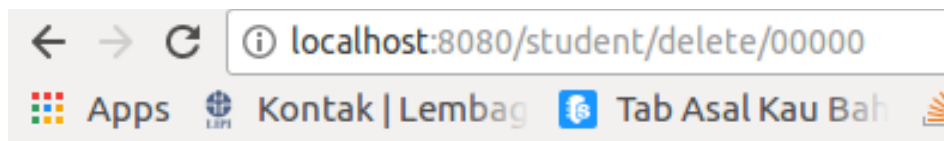


**NPM = 14769**

**Name = chanek**

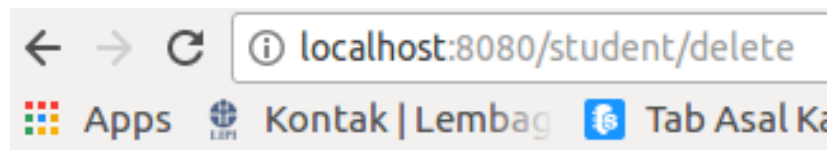
**GPA = 4.0**

Ketika menulis npm yg tidak terdaftar maka akan menampilkan halaman berikut



## NPM Not Found

Ketika tidak menulis npm maka akan menampilkan halaman berikut



## NPM Null

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.

Buat method baru deleteStudent

```
@Override
public void deleteStudent(String npm) {
    // TODO Auto-generated method stub
    for (int i = 0; i < studentList.size(); i++) {
        if (studentList.get(i).getNpm().equals(npm)) {
            studentList.remove(i);
            break;
        }
    }
}
```

```
Project Explore *InMemoryStuden StudentService. ✖
1 package com.example.tutorial3.service;
2
3 import java.util.List;
4
5
6
7 public interface StudentService {
8     StudentModel selectStudent(String npm);
9
10    List<StudentModel> selectAllStudents();
11
12    void addStudent(StudentModel student);
13    void deleteStudent(String npm);
14 }
15
```

Buat method deletePath pada StudentController

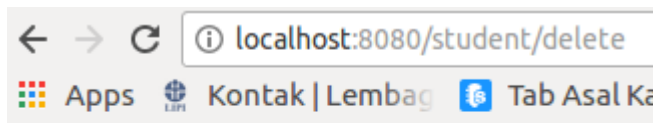
```
@RequestMapping(value = {"/student/delete", "/student/delete/{npm}"})
public String deletePath(Model model, @PathVariable Optional<String> npm) {
    if (npm.isPresent()) {
        StudentModel student = studentService.selectStudent(npm.get());
        if (student == null) {
            return "error_notFound";
        }
        model.addAttribute("studentName", student.getName());
        model.addAttribute("studentNpm", student.getNpm());
        model.addAttribute("studentGpa", student.getGpa());
        return "delete";
    }
    return "error_null";
}
```

Halaman html yang akan ditampilkan:

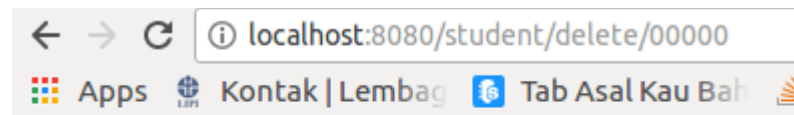
```
Project Explore StudentControll delete.html ✖ »
1 <!DOCTYPE html>
2 <html xmlns:th="http://www.thymeleaf.org">
3 <head>
4     <title>Delete Student</title>
5 </head>
6 <body>
7     <h3 th:text="'NPM = ' + ${studentNpm}">Student NPM</h3>
8     <h3 th:text="'Name = ' + ${studentName}">Student Name</h3>
9     <h3 th:text="'GPA = ' + ${studentGpa}">Student GPA</h3>
10    <h2>Telah berhasil dihapus!</h2>
11 </body>
12 </html>
```

Faisal Satrio Priatmadji  
1506738504  
ADPAP - C

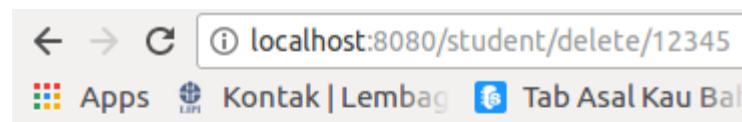
Contoh halaman html yang akan ditampilkan



## **NPM Null**



## **NPM Not Found**



**NPM = 12345**

**Name = chanek**

**GPA = 3.43**

**Telah berhasil dihapus!**

## **Ringkasan**

Pada tutorial ini, saya belajar mengenai class Model dalam MVC .  
Bagaimana penyimpanan data pada Springboot, serta bagaimana  
menggunakan pathVariable dan requestParam. Saya juga mempelajari  
penggunaan List dalam Springboot.