

Nama : Budi Indrawan
NPM : 1606954754

Tutorial Model & Service pada Project Spring Boot

A. Ringkasan Materi

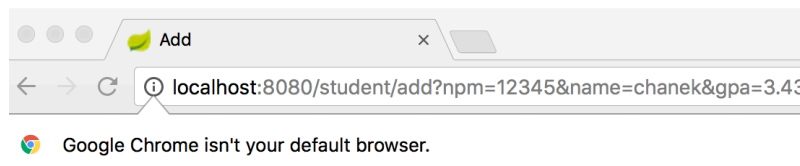
- Model digunakan untuk menyimpan informasi data student dan mempresentasikannya.
- Service adalah interface yang digunakan untuk mendefinisikan method apa saja yang harus implementasi dan digunakan untuk memanipulasi Data.

B. Jawaban Tutorial

❖ Membuat Controller dan Fungsi Add

Pertanyaan 1. Apakah hasilnya?

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

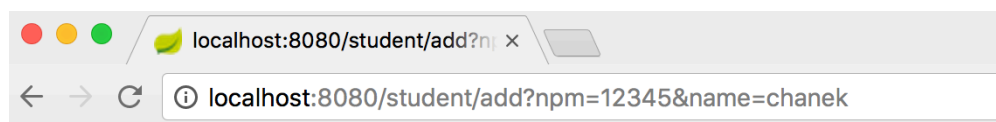


Data berhasil ditambahkan

Pertanyaan 2. Apakah hasilnya?

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

Alasan terjadi error adalah karena parameter yang di add kurang, tidak ada nilai gpa.



Whitelabel Error Page

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

Tue Feb 27 20:26:16 WIB 2018

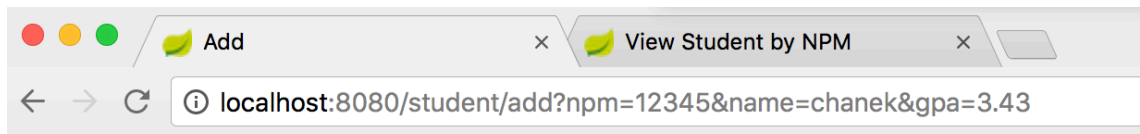
There was an unexpected error (type=Bad Request, status=400).

Required double parameter 'gpa' is not present

Nama : Budi Indrawan
NPM : 1606954754

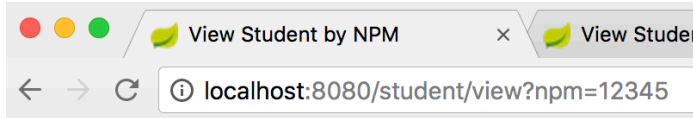
❖ Method View by NPM

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



Data berhasil ditambahkan

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



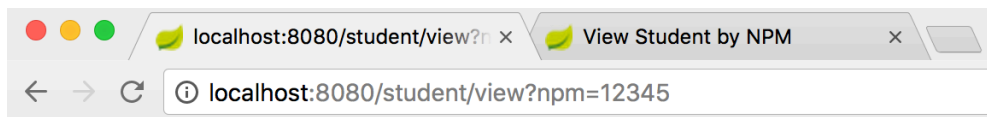
NPM = 12345

Name = chanek

GPA = 3.43

Pertanyaan 3 : Apakah data Student tersebut muncul? Ya

Membuka 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.

Tue Feb 27 20:42:47 WIB 2018

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

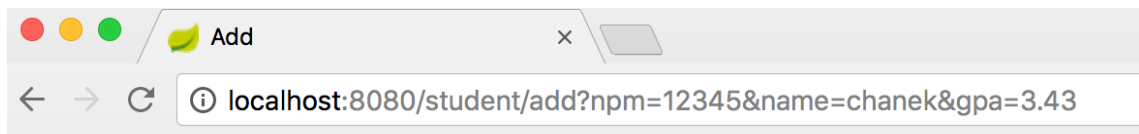
Exception evaluating SpringEL expression: "student.npm" (view:7)

Pertanyaan 4 : Apakah data Student tersebut muncul? Tidak, dikarenakan setelah dimatikan program dan di running kembali, data pada List telah terhapus sehingga diperlukan proses add kembali jika ingin ditampilkan.

Nama : Budi Indrawan
NPM : 1606954754

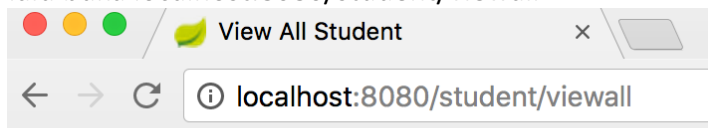
❖ Method View All

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



Data berhasil ditambahkan

lalu buka localhost:8080/student/viewall



No. = 1

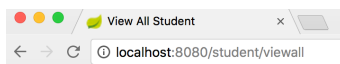
NPM = 12345

Name = chanek

GPA = 3.43

Pertanyaan 5 : apakah data Student tersebut muncul? Ya

Menambahkan 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 = 12355

Name = cekek

GPA = 2.43

No. = 3

NPM = 12365

Name = bonek

GPA = 1.43

Pertanyaan 6 : Apakah semua data Student muncul? Ya

Nama : Budi Indrawan
NPM : 1606954754

Latihan

1. Pada StudentController menambahkan method view dengan Path Variable

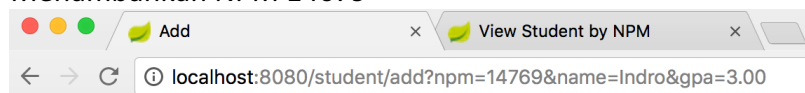
```
@RequestMapping(value = {"/student/view","student/view/{npm}"})
public String studentPath (@PathVariable Optional<String> npm, Model model) {
    if (npm.isPresent()) {
        StudentModel student = studentService.selectStudent(npm.get());
        if (student !=null) {
            model.addAttribute("student",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

    return "view";
}
```

Menambahkan file noData.html untuk menampilkan halaman jika NPM yang dimasukan atau tidak dimasukan NPM akan muncul tampilan "Data Tidak Ada".

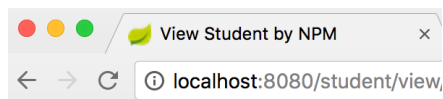
```
1 <!DOCTYPE html>
2 <html xmlns:th="http://www.thymeleaf.org">
3     <head>
4         <title>View Student by NPM</title>
5     </head>
6     <body>
7         <h3> Data tidak ada</h3>
8     </body>
9 </html>
```

Menambahkan NPM 14679



Data berhasil ditambahkan

Mengakses laman localhost:8080/student/view/14769



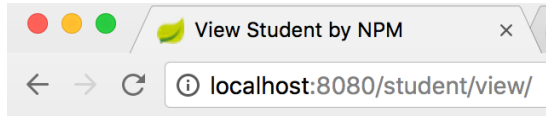
NPM = 14769

Name = Indro

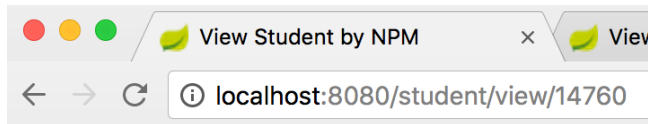
GPA = 3.0

Nama : Budi Indrawan
NPM : 1606954754

Jika NPM tidak disertakan atau NPM salah maka akan muncul tampilan tersebut



Data tidak ada



Data tidak ada

2. Menambahkan fitur untuk melakukan *delete* Student berdasarkan NPM dengan tahapan sbb:

Menambahkan method delete pada interface StudentService

```
1 package com.example.tutorial3.service;
2
3 import java.util.List;
4 import java.util.Optional;
5
6 import com.example.tutorial3.model.StudentModel;
7
8 public interface StudentService {
9     StudentModel selectStudent(String npm);
10
11     StudentModel selectStudent(Optional <String> npm);
12
13     List<StudentModel> selectAllStudents();
14
15     void addStudent(StudentModel student);
16
17     void deleteStudent(StudentModel student);
18 }
19
20
```

Menambahkan implementasi delete pada InMemoryStudentService

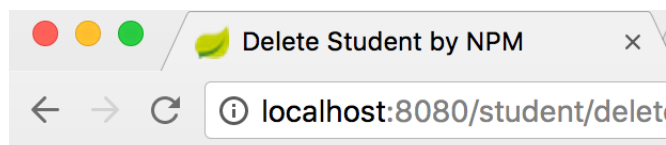
```
@Override
public void deleteStudent(StudentModel student) {
    // TODO Auto-generated method stub
    studentList.remove(student);
}
```

Nama : Budi Indrawan
NPM : 1606954754

Menambahkan method delete pada StudentController

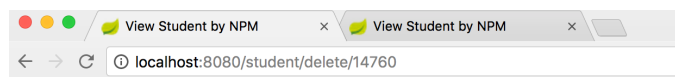
```
@RequestMapping(value = {"/student/delete", "student/delete/{npm}"})  
public String deletePath (@PathVariable Optional<String> npm, Model model) {  
    if (npm.isPresent()) {  
        StudentModel student = studentService.selectStudent(npm.get());  
        if (student != null) {  
          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

Mencoba menghapus data NPM yang di add pada soal no.1 dengan mengakses laman localhost:8080/student/delete/14769

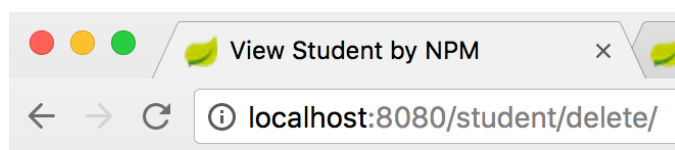


Student with NPM 14769deleted

Jika nomor NPM tidak diberikan atau tidak ditemukan kembalikan halaman *error* yang berisi informasi bahwa nomor NPM kosong atau tidak ditemukan



Data tidak ada



Data tidak ada

Nama : Budi Indrawan
NPM : 1606954754

C. Penjelasan Method selectStudent yang di implementasikan

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

@Override
public StudentModel selectStudent(Optional<String> npm) {
    // TODO Auto-generated method stub
    for (int i=0; i<studentList.size();i++) {
        if(studentList.get(i).getNpm().equals(npm)) {
            return studentList.get(i);
        }
    }
    return null;
}
```

Method selectStudent terdapat 2 dimana method pertama di implementasikan untuk @RequestMapping yang memiliki 1 value, sedangkan selectStudent dengan Optional di implementasikan untuk @RequestMapping yang memiliki 1 value.

Method selectStudent ini akan mengembalikan data student jika parameter yang di input ada pada studentList dan mengembalikan nilai Null apabila data tidak ditemukan.

D. Penjelasan Fitur Delete

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

return "delete";
}
```

Method delete diatas digunakan untuk menghapus data pada studentList sesuai dengan npm yang ada pada parameter. Method akan mengembalikan informasi bahwa data telah berhasil dihapus jika npm pada parameter ada pada studentList dan akan memberikan informasi Data tidak ditemukan apabila npm tidak diinput atau ditemukan pada studentList.