

TUTORIAL

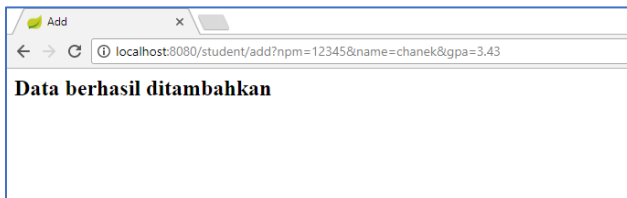
Write-up implement method selectStudent() ialah sebagai berikut. Saya melakukan looping terhadap List studentList untuk memilih student berdasarkan npm yang ada.

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

Jawaban Pertanyaan

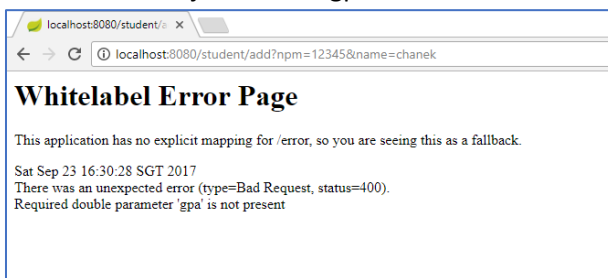
1. Hasilnya "Data berhasil ditambahkan".

Hal tersebut terjadi karena pada @RequestParam required = true dan dalam link localhost:8080/student/add?npm=12345&name=chanek&gpa=3.43 , npm-nama-gpa tertulis lengkap sehingga data berhasil ditambahkan.

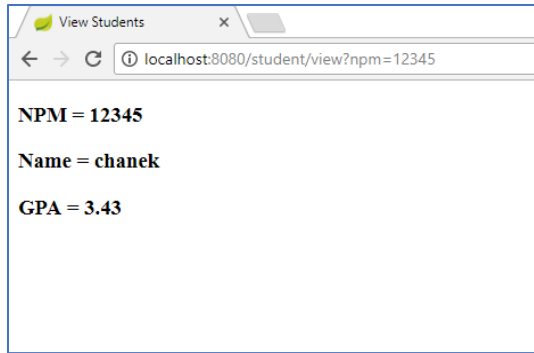


2. Hasilnya Whitelabel Error Page.

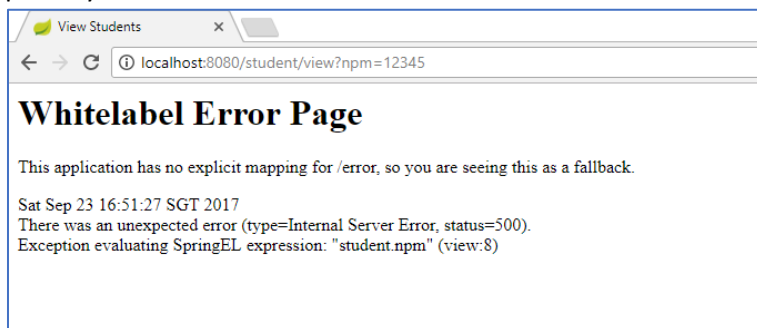
Hal tersebut terjadi karena gpa tidak ditulis sementara requirednya di set true.



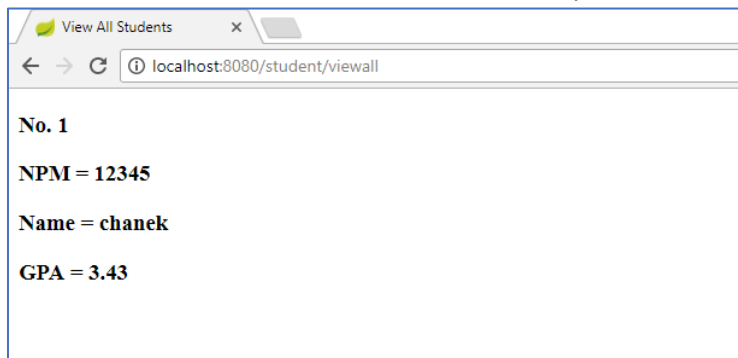
3. Muncul NPM, Name, dan GPA yang dimasukkan di dalam url. Hal tersebut terjadi karena localhost:8080/student/view?npm=12345 dijalankan setelah localhost:8080/student/add?npm=12345&name=chanek&gpa=3.43.



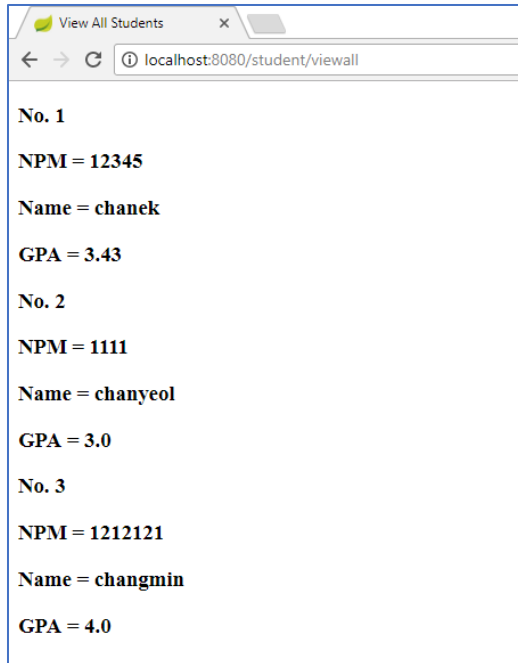
4. Tidak muncul, karena sebelumnya tidak ditambahkan data siswa (npm, nama, gpa) seperti pada pertanyaan nomor 3.



5. Data tersebut muncul. Setelah add lalu menampilkan viewall.

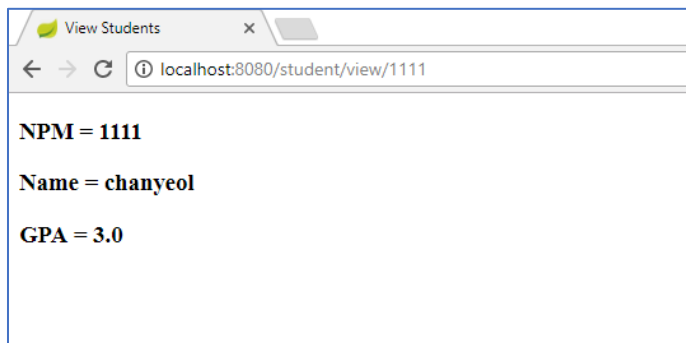


6. Jika ditambahkan data student baru tanpa program diberhentikan, maka saat memanggil viewall hasil yang muncul ialah menampilkan seluruh student.



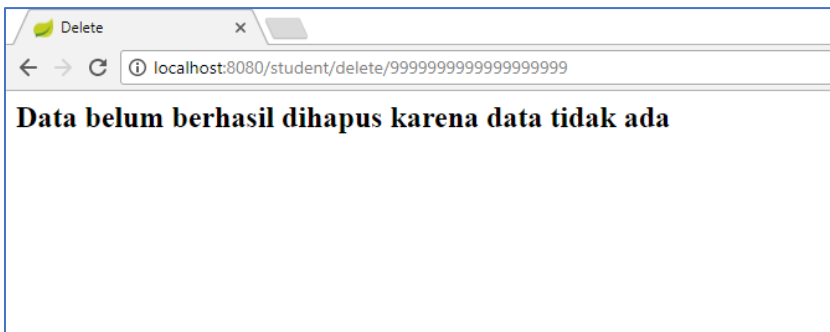
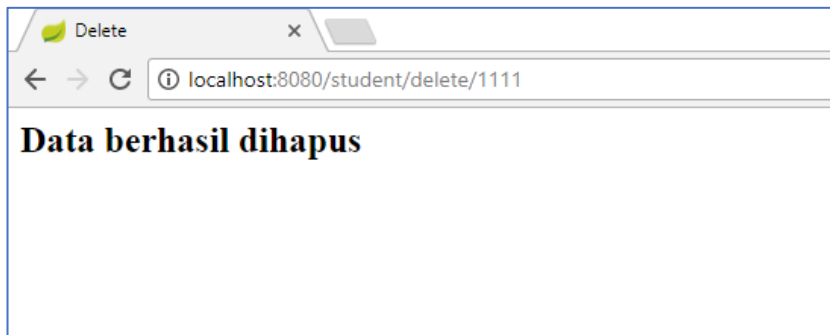
LATIHAN

1. Saat membuat view dengan menggunakan path variable npm, awalnya saya membuat seperti syntax-syntax pada tutorial sebelumnya di mana ada model dan string. Setelah itu saya mengecek apakah npm dalam link url kosong atau tidak. Apabila tidak kosong, maka saya akan memilih/mencari student dengan npm tersebut menggunakan method selectStudent() yang sudah dibuat dalam kelas InMemoryStudentService.java & StudentService.java . Kemudian, saya menambahkan murid tersebut ke dalam model dan selanjutnya saya return halaman view.html . Apabila npm dalam link url kosong, maka saya akan return halaman viewnull.html .



```
@RequestMapping("/student/view/{npm}")
public String view2 (@PathVariable String npm, Model model) {
    if(!npm.isEmpty()) {
        StudentModel student = studentService.selectStudent(npm);
        model.addAttribute("student", student);
        return "view";
    } else {
        return "viewnull";
    }
}
```

2. Saat membuat delete dengan menggunakan path variable npm, saya menggunakan dua alternative value url, yaitu saat npm kosong dan saat npm ada. Apabila npm ada/ present, maka seperti dalam view, saya memilih student dengan npm tersebut menggunakan selectStudent(). Setelah itu, saya mengecek apakah student ini benar-benar ada atau tidak, apabila ada, maka saya akan delete dengan membuat method deleteStudent() pada InMemoryStudentService.java & StudentService.java . Apabila student ternyata tidak ada maka saya akan return halaman deletegagal.html . Nah, jika url npm kosong maka saya akan return halaman deleteempty.html yang berisi keterangan bahwa NPM Masih Kosong.



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