

Tutorial 3 Writeup

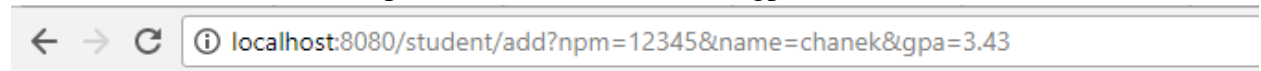
A. Rangkuman

Tutorial minggu ini membahas praktik materi MVC (Model-View-Controller) pada Spring Boot. Arsitektur MVC akan memisahkan aspek aplikasi sesuai logiknya. Pada tutorial kali ini, saya menggunakan arsitektur aplikasi dengan Model, View, Service dan Controller. Controller berfungsi untuk mengatur flow aplikasi dan mengolah input, Model berfungsi sebagai representasi objek Student, View berfungsi untuk menampilkan user interface dan Service berfungsi untuk menjalankan proses aplikasi.

B. Tutorial

1. Jalankan program dan buka

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



Data berhasil ditambahkan

2. Jalankan program dan buka

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



Whitelabel Error Page

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

Fri Sep 22 16:22:35 ICT 2017

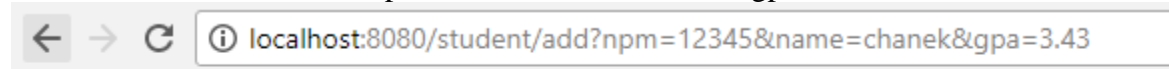
There was an unexpected error (type=Bad Request, status=400).

Required double parameter 'gpa' is not present

Terjadi error karena tidak terdapat parameter “gpa” pada masukan yang diberikan, sehingga objek baru Student tidak dapat dibuat.

3. 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



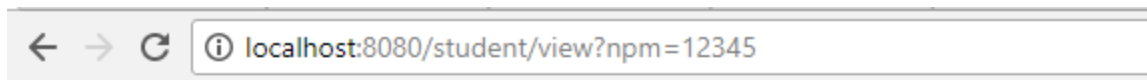
NPM = 12345

Name = chanek

GPA = 3.43

Data student muncul karena objek berhasil dibuat, sehingga detail student tersebut dapat dilihat melalui path /student/view

4. 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.

Fri Sep 22 16:34:10 ICT 2017

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

Exception evaluating SpringEL expression: "student.npm" (view:8)

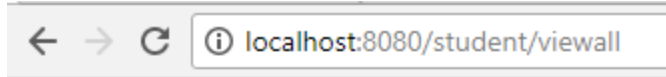
Halaman tersebut tidak bisa diakses karena objek student dengan npm 12345 belum dibuat dengan path student/add/.

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



Data berhasil ditambahkan

lalu buka localhost:8080/student/viewall,



No. 1

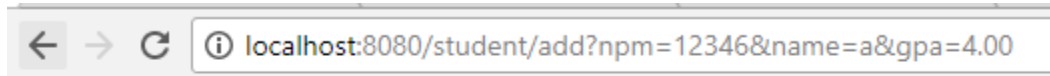
NPM = 12345

Name = chanek

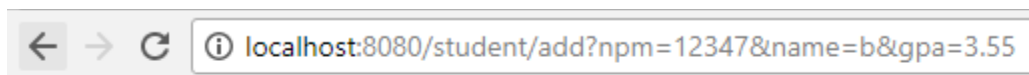
GPA = 3.43

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

Menambahkan student

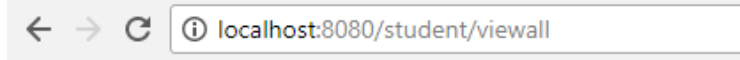


Data berhasil ditambahkan



Data berhasil ditambahkan

/student/viewall untuk menampilkan list student yang telah dimasukkan



No. 1

NPM = 12345

Name = chanek

GPA = 3.43

No. 2

NPM = 12346

Name = a

GPA = 4.0

No. 3

NPM = 12347

Name = b

GPA = 3.55

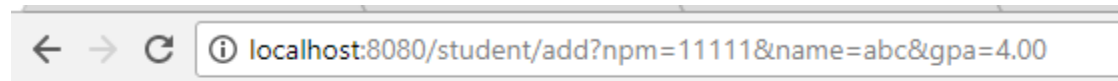
C. Latihan

1. Method view student

```
@RequestMapping("/student/view/{npm}")
public String viewPath(Model model, @PathVariable String npm) {
    StudentModel student = studentService.selectStudent(npm);
    if(student == null) {
        return "notfound";
    }

    else {
        model.addAttribute("student", student);
        return "view";
    }
}
```

Memasukkan data student



Data berhasil ditambahkan

localhost:8080/student/view/11111



NPM = 11111

Name = abc

GPA = 4.0

2. Fitur delete student

StudentController.java

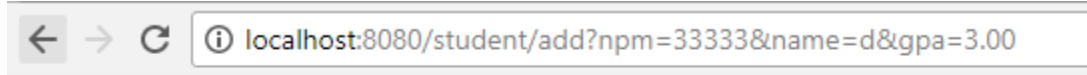
```
@RequestMapping("/student/delete/{npm}")
public String delete(Model model, @PathVariable String npm) {
    StudentModel student = studentService.selectStudent(npm);

    if(student == null) {
        return "notfound";
    }
    else{
        studentService.deleteStudent(student);
        return "delete";
    }
}
```

InMemoryStudentService.java

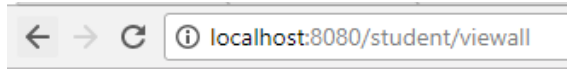
```
@Override
public void deleteStudent(StudentModel student) {
    studentList.remove(student);
}
```

Tambah data student baru



Data berhasil ditambahkan

Lihat detail semua student di localhost:8080/student/viewall



No. 1

NPM = 12346

Name = a

GPA = 4.0

No. 2

NPM = 22222

Name = b

GPA = 3.0

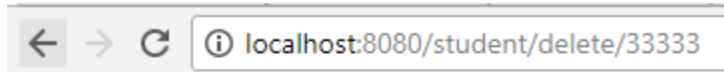
No. 3

NPM = 33333

Name = d

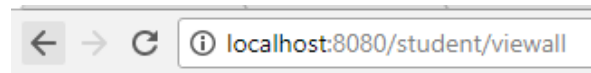
GPA = 3.0

Delete student dengan NPM 33333 tersebut



Data berhasil dihapus

Cek kembali list semua student



No. 1

NPM = 12346

Name = a

GPA = 4.0

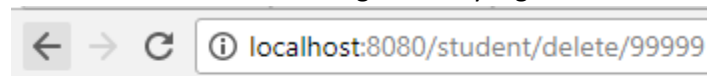
No. 2

NPM = 22222

Name = b

GPA = 3.0

Mencoba delete student dengan NPM yang tidak ada di list semua student



Data tidak ditemukan

D. Penjelasan pembuatan fitur pada latihan:

- Method *view student* dengan *path variable*
Saya membuat method *viewPath* yang mengambil parameter model dan String NPM. Lalu saya membuat objek *StudentModel* yang mengecek apakah ada *student* yang memiliki NPM tersebut. Jika *student* dengan NPM tersebut ada, maka akan *redirect* ke halaman *view.html* yang berisi detail informasi *student*.
- Method *delete student* berdasarkan NPM
Pertama, saya menambahkan *method* baru *deleteStudent* di *interface* *StudentService*. Kemudian saya meng-*override method* tersebut di *file* *InMemoryStudentService.java*. Setelah itu, saya mengatur *RequestMapping path* yang diinginkan di *file* *StudentController.java* dan menentukan parameter yang akan diterima yaitu NPM. Lalu saya membuat kondisi jika terdapat *student* dengan NPM yang dimasukkan di *path*, maka datanya akan dihapus. Namun jika tidak terdapat *student* yang sesuai dengan NPM yang dimasukkan maka akan *redirect* ke halaman *notfound.html*