Muhammad Kamal – 1606954905

Tutorial3 – write up

4. implement select student

@Override

public StudentModel selectStudent(String npm) {

//implemet

for(StudentModel itemStudent : studentList)

{

if (itemStudent.getNpm().equals(npm))

{

return itemStudent;

}

}

return null;

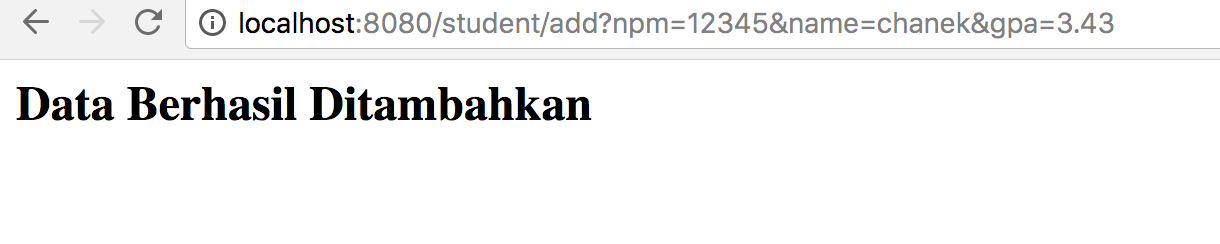
}

* Pertanyaan 1 :

Menjalankan program panggil Controller Add

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

Hasilnya : tidak error, muncul seperti gambar berikut :



* Pertanyaan 2

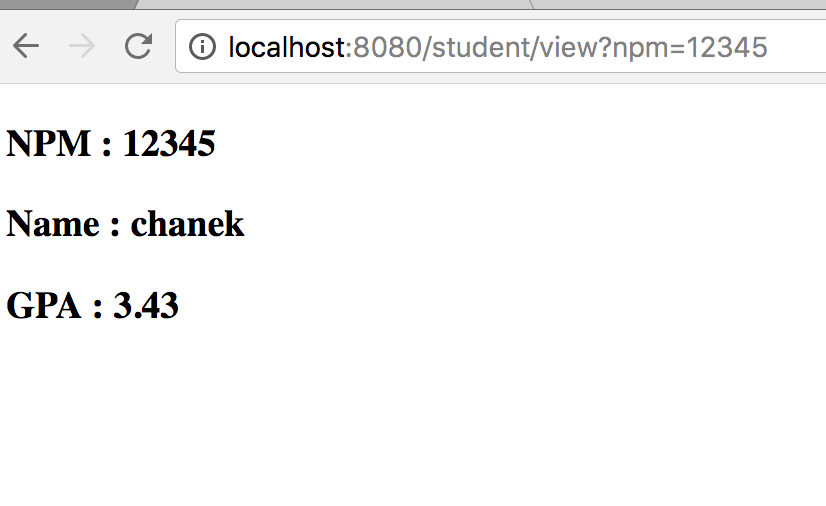
Menjalankan program dan panggil controller add seperti berikut:

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

Hasil : muncul error, karena tidak ada input parameter gpa yang required

* Pertanyaan 3

Hasil : muncul, seperti berikut

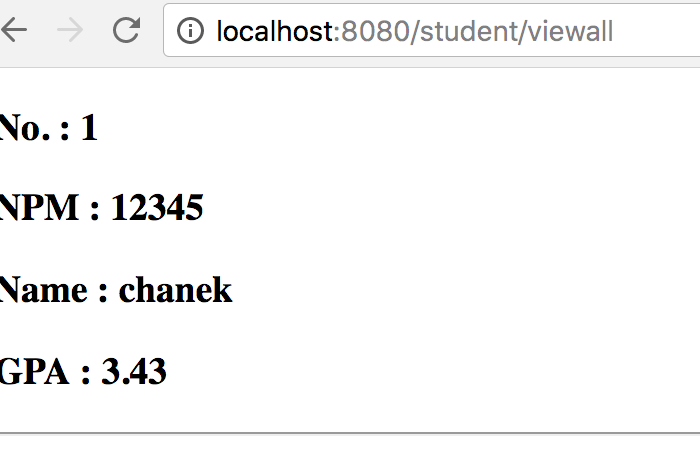


* Pertanyaan 4 – matikan program, dan jalankan kembali. apakah data Student tersebut muncul? Jika tidak, mengapa?

Tidak muncul, karena studentlist belum terisi data, dan ketika view dipanggil, return adalah n]

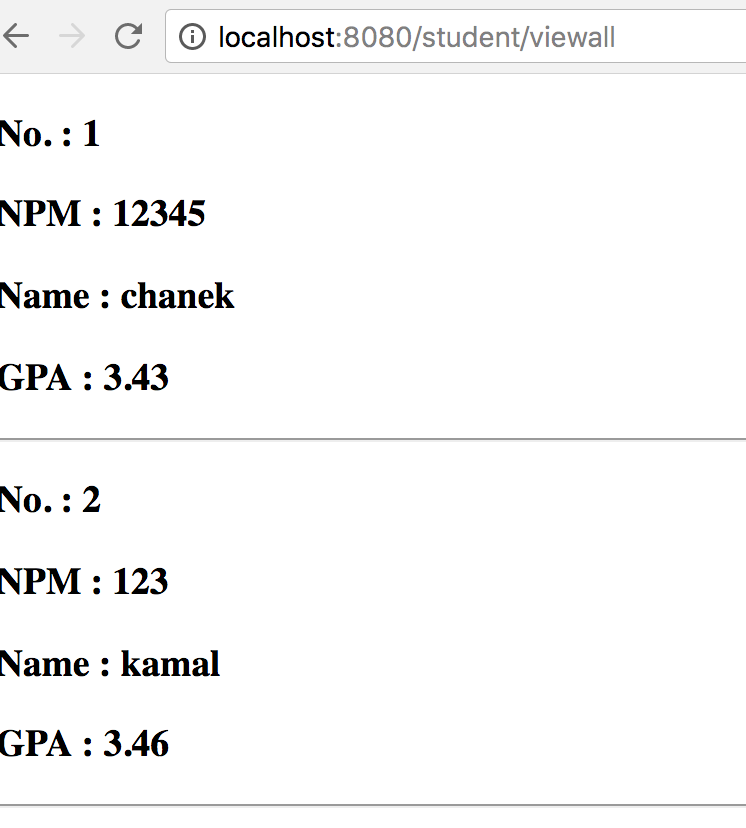
* Pertanyaan 5​: apakah data Student tersebut muncul?

Ya muncul, seperti berikut:



* Pertanyaan 6​: Apakah semua data Student muncul?

Masukan data berbeda, lalu panggil viewall

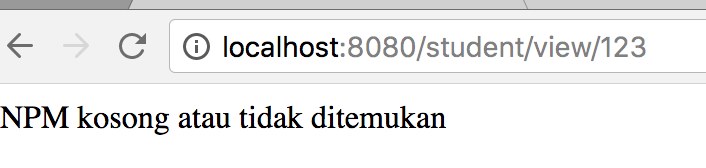


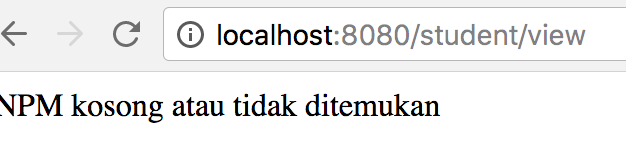
**Latihan**

1. **Tambah method view dengan menggunakan path variable**

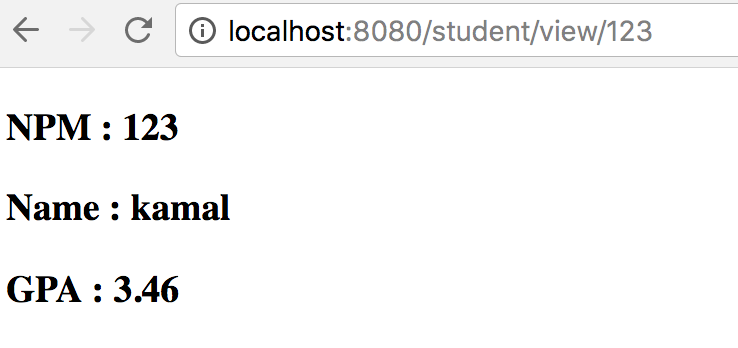
Npm kosong atau tidak ditemukan

Result





NPM ditemukan



Result diatas dihasilkan dari method baru yang dibuat dengan mengunakan patvariable “npm” yang diberikan tanda optional untuk menghandle apabila tidak memasukkan nomor npm. efek dari penambahan method dengan request mapping “student/view” dan “student/view/{npm}” sebaiknya merename request mapping yang di tutorial menggunakan requestparam, karena akan bentrok dan ambigu error apabila requestmapping yang sebelumnya dibuat untuk view menggunakan requestparam npm

Berikut potongan code untuk pathvariable view by npm

// @RequestMapping("/student/view") --> remark since ambigouse error countered after create path variable method with same "view" action method

@RequestMapping("/student/viewold")

public String view(Model model, @RequestParam(value="npm", required=true) String npm) {

StudentModel student = studentService.selectStudent(npm);

//System.out.println(student);

model.addAttribute("student", student);

return "view";

}

@RequestMapping(value = {"/student/view", "student/view/{npm}"})

public String view(@PathVariable Optional<String> npm, Model model) {

if (npm.isPresent())

{

StudentModel student = studentService.selectStudent(npm.get());

System.out.println(student);

if (student != null)

{

System.out.println("cari view view");

model.addAttribute("student", student);

return "view";

}

else

{

System.out.println("cari view not found");

return "viewnotfound";

}

}

else {

System.out.println("cari view not found");

return "viewnotfound";

}

}

Selanjutnya adalah menambahkan html template “viewnotfound”

1. Tambah fitur delete berdasarkan NPM
2. Tambahkan methd interface deleteStudent di dalam studentService
3. Implement method interface tersebut di dalam InMemoryStudentService
4. Buat baru html dengan nama delete.html untuk memberikan notifikasi data telah terhapus
5. Tambahkan method di dalam controller, seperti berikut

/\*\*\*

\* method to delete student from collections student list

\*/

@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)

{

// if student exist, execute service delete

studentService.deleteStudent(npm.get());

return "delete";

}

else

{

return "viewnotfound";

}

}

else {

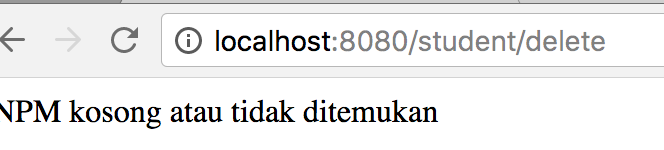
return "viewnotfound";

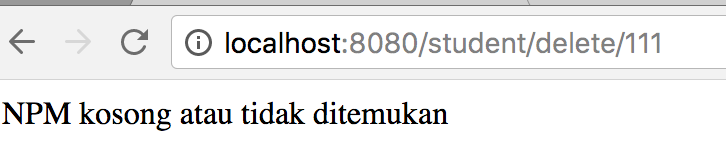
}

}

Result delete :

Apabila npm tidak dimasukkan atau tidak ditemukan, akan muncul halaman berikut





Apabila ditemukan, akan muncul halaman informasi delete berhasil seperti berikut