

Nama : Windiany Lestari Sitorus

Kelas : Ekstensi

NPM : 1606955063

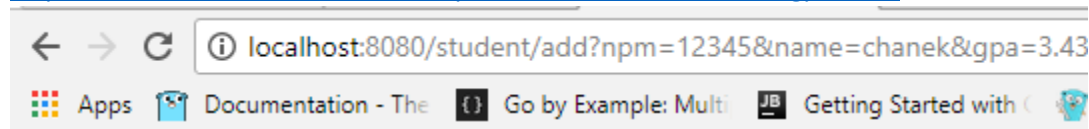
Ringkasan Materi Tutorial 3 APAP

1. Pada tutorial ini lanjut mempelajari konsep MVC menggunakan spring boot, pada model didefinisikan data apa saja yang akan digunakan. Pada service akan didefinisikan method apa saja yang akan digunakan dalam mengelola data yang telah didefinisikan di model. Nantinya service akan menjadi perantara antara controller dan database.
2. Pada tutorial ini juga saya mempelajari CRUD (Create, Update, Delete) juga menampilkan semua data student yang telah di create.
3. Menerapkan kembali path variable.

Tutorial 3 APAP

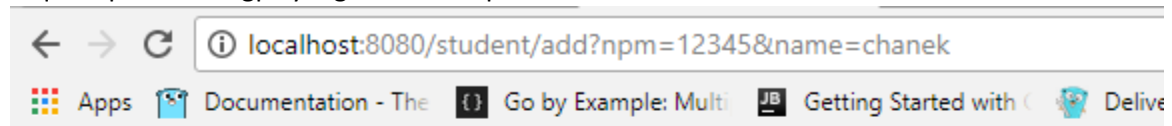
1. Hasilnya tidak error saat menjalankan

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



Data berhasil ditambahkan

2. Akan tetapi saat menjalankan <http://localhost:8080/student/add?npm=12345&name=chanek> hasilnya akan error 400 dikarenakan GET method add yang didefine di controller memiliki request parameter gpa yang bersifat required.



Whitelabel Error Page

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

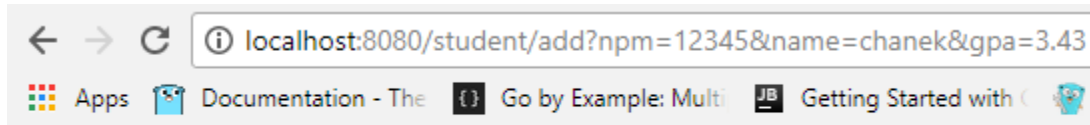
Sat Mar 03 01:37:51 ICT 2018

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

Required double parameter 'gpa' is not present

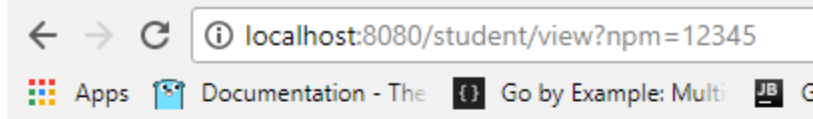
3. Saat dijalankan kembali

<http://localhost:8080/student/add?npm=12345&name=chanek&gpa=3.43> maka hasilnya adalah seperti berikut:



Data berhasil ditambahkan

Dan saat menjalankan <http://localhost:8080/student/view?npm=12345> maka akan tampil data student seperti berikut:

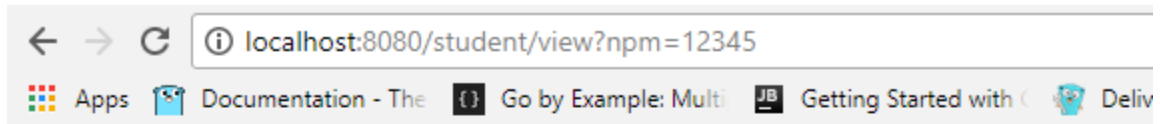


NPM = 12345

Name = chanek

GPA = 3.43

4. Kemudian ketika program diberhentikan dan request URL <http://localhost:8080/student/view?npm=12345> dijalankan, maka akan error 500 karena belum ada data yang ditambahkan ke list studentList, karena disini studentList hanya bersifat static, tidak menyimpan semua data yang telah dijalankan program sebelumnya:



Whitelabel Error Page

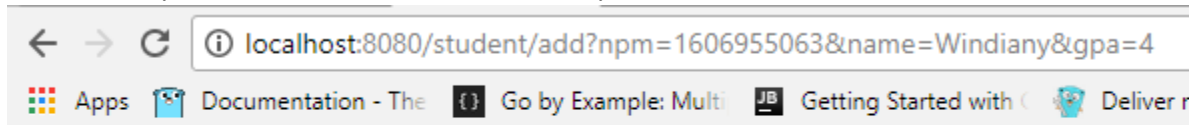
This application has no explicit mapping for /error, so you are seeing this as a fallback.

Sat Mar 03 11:14:57 ICT 2018

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

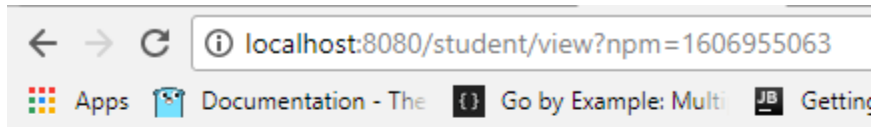
Exception evaluating SpringEL expression: "student.npm" (view:7)

Kemudian saya menambahkan data student lainnya



Data berhasil ditambahkan

Dan menjalankan request view, berhasil

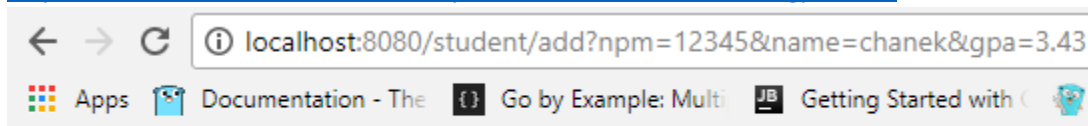


NPM = 1606955063

Name = Windiany

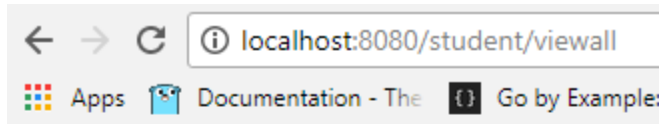
GPA = 4.0

5. Setelah menambahkan method viewAll dan menjalankan <http://localhost:8080/student/add?npm=12345&name=chanek&gpa=3.43>



Data berhasil ditambahkan

Kemudian menjalankan method viewAll <http://localhost:8080/student/viewall> maka data student akan tampil



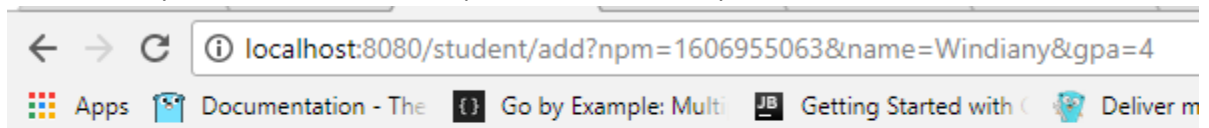
No 1

NPM = 12345

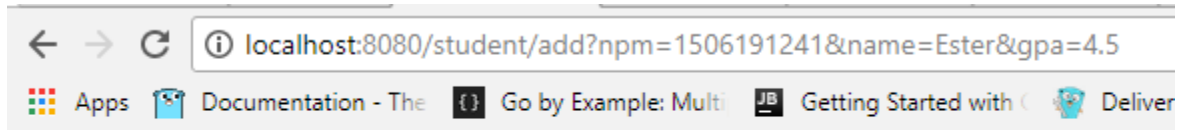
Name chanek

GPA 3.43

6. Kemudian saya menambahkan beberapa data student lainnya

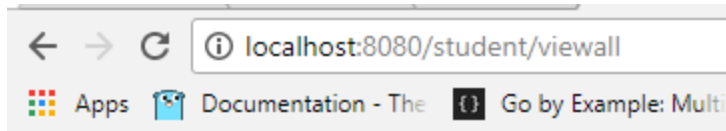


Data berhasil ditambahkan



Data berhasil ditambahkan

Kemudian menjalankan kembali request URL viewall, maka akan tampil seluruh data student yang telah ditambahkan sebelumnya.



No 1

NPM = 12345

Name chanek

GPA 3.43

No 2

NPM = 1606955063

Name Windiany

GPA 4.0

No 3

NPM = 1506191241

Name Ester

GPA 4.5

Latihan

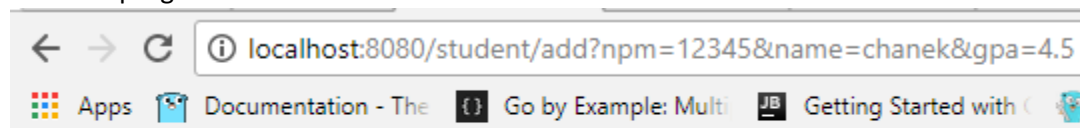
1. Pada StudentController tambahkan method view dengan menggunakan path variable. Pada request mapping akan ditambahkan valuenya menjadi /student/view dan student/view/{npm} kemudian akan dilakukan pengecekan npm, apabila npm nya ada pada path variable maka akan di return student dengan npm yang ada pada parameter. Apabila npm nya tidak sesuai, maka akan return error message. Hal yang sama juga terjadi apabila pada request URL tidak disertakan npm. Kemudian akan diredirect ke halaman view.html

```
@RequestMapping(value= {"/student/view/", "student/view/{npm}"})
public String viewByNpm(@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("errorMsg", npm.get() + " not found");
        }
    }else {
        model.addAttribute("errorMsg", "Please add NPM in your parameter");
    }
    return "view";
}
```

Kemudian di file view.html ditambahkan pengecekan variable student, apabila variable student exist maka yang ditampilkan data student, apabila tidak maka akan ditampilkan error message.

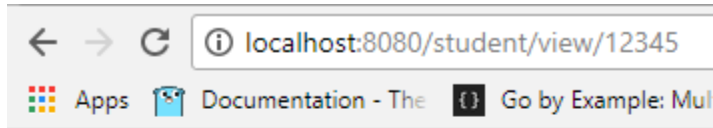
```
1 <!DOCTYPE html>
2 <html xmlns:th="http://www.thymeleaf.org">
3 <head>
4 <title>View Student By NPM</title>
5 </head>
6 <body>
7 <div th:if="${student}">
8 <h3 th:text="'NPM = ' + ${student.npm}">Student NPM</h3>
9 <h3 th:text="'Name = ' + ${student.name}">Student Name</h3>
10 <h3 th:text="'GPA = ' + ${student.gpa}">Student GPA</h3>
11 </div>
12 <div th:unless="${student}">
13 <h3 th:text="${errorMsg}">Error Message</h3>
14 </div>
15 </body>
16 </html>
```

Jalankan program dan tambahkan data student.



Data berhasil ditambahkan

Jalankan <http://localhost:8080/student/view/12345> untuk test path variable yang telah di define pada request mapping.

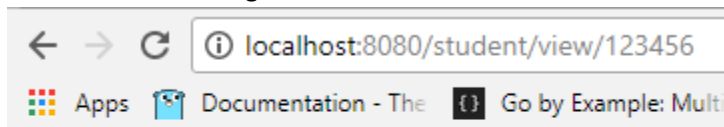


NPM = 12345

Name = chanek

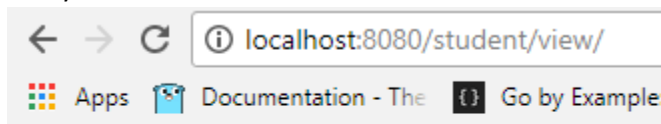
GPA = 4.5

Kemudian test view dengan npm student yang tidak ditambahkan sebelumnya, maka akan muncul error message 123456 not found.



123456 not found

Sementara apabila npm tidak disertakan di request URL maka akan muncul error message untuk menyertakan NPM.



Please add NPM in your parameter

2. Untuk method delete, maka yang pertama sekali saya lakukan adalah define method delete pada interface StudentService.

```
package com.example.tutorial3.service;

import java.util.List;

public interface StudentService {
    StudentModel selectStudent(String npm);

    List<StudentModel> selectAllStudents();

    void addStudent(StudentModel student);

    boolean deleteStudent(String npm);
}
```

Kemudian override method tersebut di InMemoryStudentService.

```
@Override
public boolean deleteStudent(String npm) {
    for(StudentModel student : studentList) {
        if(student.getNpm().equals(npm)) {
            studentList.remove(student);
            return true;
        }
    }
    return false;
}
```

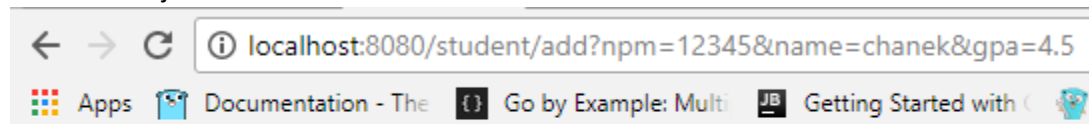
Kemudian di StudentController tambahkan request mapping untuk method delete, apabila NPM exist pada request URL maka akan dilakukan penghapusan data student. Apabila data student berhasil dihapus maka akan muncul message "Data student berhasil dihapus". Akan tetapi apabila data tidak berhasil dihapus, berarti NPM not found dan tidak ada data yang dihapus. Dan apabila NPM tidak ada pada request URL, maka akan muncul message "Please add NPM in your parameter". Kemudian akan di direct ke delete.html.

```
@RequestMapping(value= {"student/delete/", "student/delete/{npm}"})
public String deleteByNpm(@PathVariable Optional<String> npm, Model model) {
    if(npm.isPresent()) {
        boolean result = studentService.deleteStudent(npm.get());
        if(result == true) {
            model.addAttribute("message", "Data student berhasil dihapus");
        }else {
            model.addAttribute("message", npm.get() + " not found");
        }
    }else {
        model.addAttribute("message", "Please add NPM in your parameter");
    }
    return "delete";
}
```

Dan pada delete.html di define variable \${message} untuk menampung hasil dari StudentController.

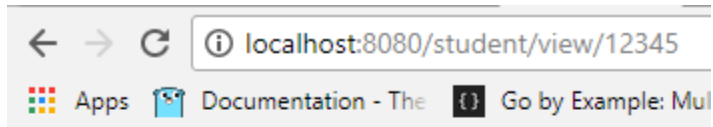
```
<!DOCTYPE html>
<html xmlns:th="http://www.thymeleaf.org">
<head>
<title>Delete</title>
</head>
<body>
    <h2 th:text="${message}"></h2>
</body>
</html>
```

Maka saat dijalankan add



Data berhasil ditambahkan

Kemudian view

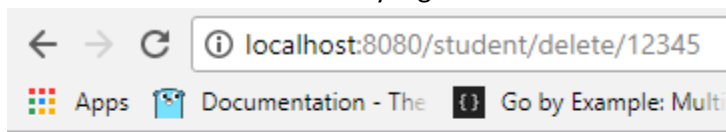


NPM = 12345

Name = chanek

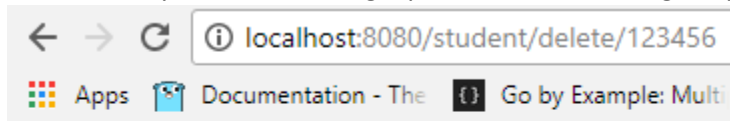
GPA = 4.5

Kemudian delete data student yang ditambahkan



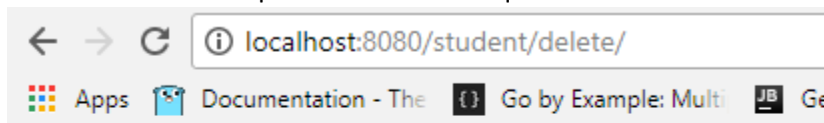
Data student berhasil dihapus

Kemudian saya mencoba menghapus data student dengan npm secara acak.



123456 not found

Tidak memasukkan npm saat delete di request URL



Please add NPM in your parameter