

Nama : Desy Ulfasari
NPM : 1406557503
Tutorial 3 APAP Ekstensi

Deliverables Tutorial 3

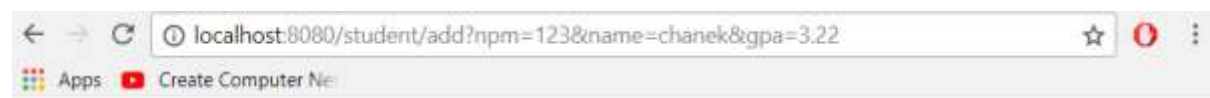
Pada tutorial kali ini, saya mempelajari mengenai model dengan konsep MVC. Model merupakan sebuah objek yang merepresentasikan dan menyimpan informasi terhadap suatu hal. Contoh dari model adalah objek mahasiswa. Objek mahasiswa ini memiliki nama, alamat, tanggal lahir, nomor pokok mahasiswa, dsbnya. Model digunakan untuk merepresentasikan hal-hal tersebut dalam atribut yang dimiliki sebuah model.

Selanjutnya, mempelajari cara membuat service dengan fungsi create and read data dengan konsep MVC, Service adalah suatu layer yang menjadi mediator antara controller dan database. Pada service layer, disimpan business logic yang digunakan untuk mengolah data yang terdapat dalam database. Pengolahan ini meliputi kalkulasi data yang diambil dari database, manipulasi user input kedalam database, dsbnya. Contohnya adalah melakukan kalkulasi IPK sebuah model mahasiswa.

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

Pertanyaan 1: apakah hasilnya? Jika error, tuliskan penjelasan Anda.

Program akan menyimpan data mahasiswa



Data berhasil ditambahkan

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

Pertanyaan 2: apakah hasilnya? Jika error, tuliskan penjelasan Anda.

Akan terjadi error, karena tidak mencantumkan gpa dari mahasiswa. Program controller mewajibkan user untuk mengisi gpa agar program tidak error.



Whitelabel Error Page

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

Sat Mar 03 20:29:02 ICT 2018

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

Required double parameter 'gpa' is not present

Nama : Desy Ulfasari
NPM : 1406557503
Tutorial 3 APAP Ekstensi

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

localhost:8080/student/view?npm=12345

Pertanyaan 3: apakah data Student tersebut muncul? Jika tidak, mengapa?

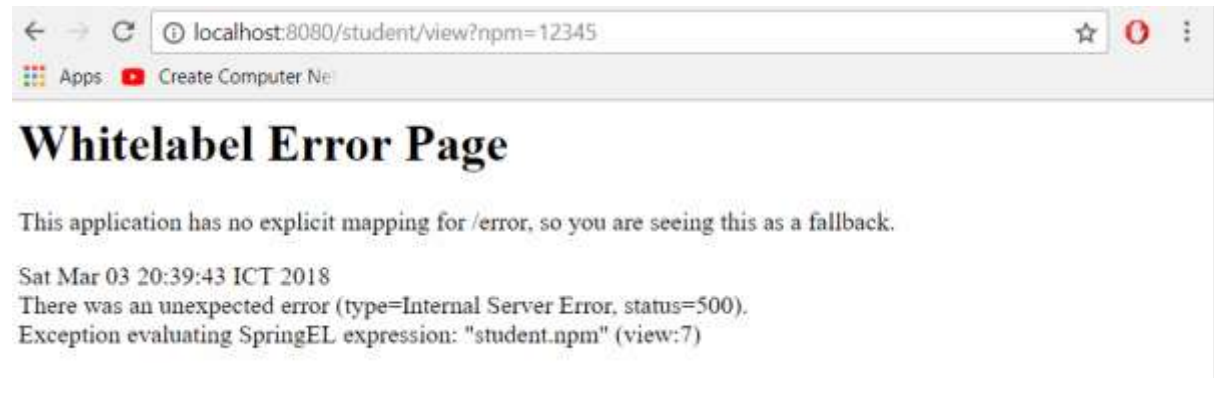
Program akan menampilkan data student



localhost:8080/student/view?npm=12345

Pertanyaan 4: apakah data Student tersebut muncul? Jika tidak, mengapa?

Tidak, karena data di arraylist kosong



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

localhost:8080/student/viewall,

Pertanyaan 5: apakah data Student tersebut muncul?

Nama : Desy Ulfasari
NPM : 1406557503
Tutorial 3 APAP Ekstensi



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

Pertanyaan 6: Apakah semua data Student muncul?



Nama : Desy Ulfasari
NPM : 1406557503
Tutorial 3 APAP Ekstensi

Latihan

1. Pada StudentController tambahkan sebuah method view Student dengan menggunakan Path Variable. Misalnya, Anda ingin memasukkan data seorang Student yang memiliki NPM 14769, untuk melihat data yang baru dimasukkan tersebut dapat mengakses halaman localhost:8080/student/view/14769.

- Method view Student dengan menggunakan Path Variable

```
@RequestMapping(value= {"/student/view/", "/student/view/{npm}"})  
public String view (Model model, @PathVariable Optional <String> npm) {  
    if(npm.isPresent()) {  
        StudentModel student= studentService.selectStudent(npm.get());  
        if(student==null) {  
            model.addAttribute("npm", npm.get());  
            model.addAttribute("statement", "tidak ditemukan");  
        }  
        else {  
            model.addAttribute("student", student);  
            return "view";  
        }  
    }  
    else {  
        model.addAttribute("npm", "");  
        model.addAttribute("statement", "kosong ");  
    }  
      
    return "error|view";  
}
```

- File html untuk view.html dan errorview.html

```
<!DOCTYPE html>  
<html xmlns:th="http://www.thymeleaf.org">  
<head>  
    <title>Error view by NPM</title>  
</head>  
<body>  
    <h2 th:text="'NPM'+${npm}+${statement}">NPM invalid</h2>  
</body>  
</html>  
  
<!DOCTYPE html>  
<html xmlns:th="http://www.thymeleaf.org">  
<head>  
    <title>View Student by NPM</title>  
</head>  
<body>  
    <h3 th:text="'NPM = ' + ${student.npm}">Student NPM</h3>  
    <h3 th:text="'Name = ' + ${student.name}">Student Name</h3>  
    <h3 th:text="'GPA = ' + ${student.gpa}">Student GPA</h3>  
</body>  
</html>
```

- Memasukkan data untuk testing view

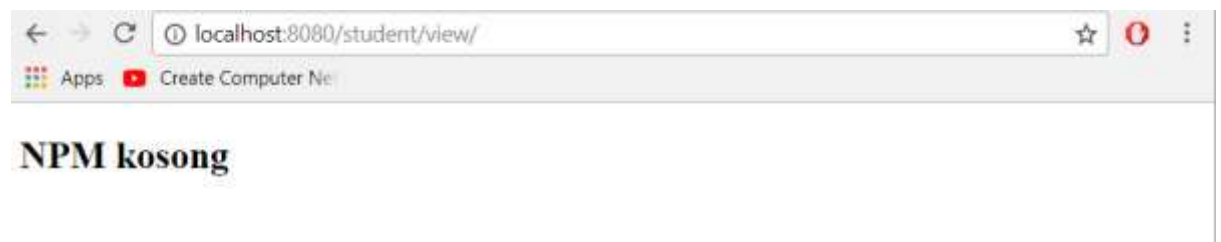
Nama : Desy Ulfasari
NPM : 1406557503
Tutorial 3 APAP Ekstensi



- Tampilan view dengan path variable



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



2. Tambahkan fitur untuk melakukan delete Student berdasarkan NPM. Misalnya, setelah melakukan add Student pada soal nomor 1, cobalah untuk melakukan delete data tersebut dengan mengakses halaman localhost:8080/student/delete/14769.

- Method delete NPM. (testing dengan menghapus pada input data pada nomor sebelumnya)

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

    return "delete";
}
```

- Tampilan view untuk delete Student

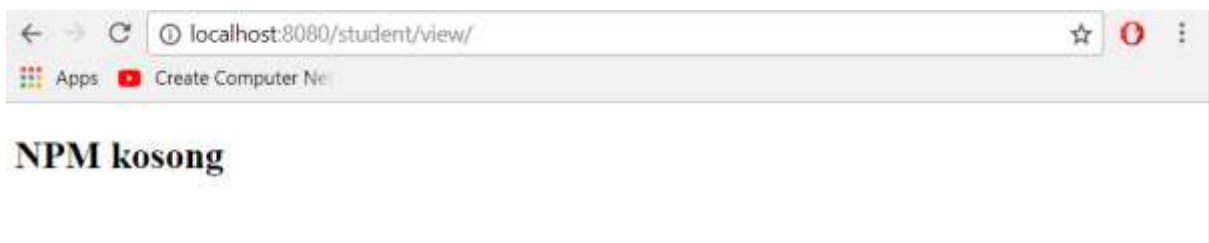
```
<html>
<head>
<title>Delete</title>
</head>
<body>
<h2>Data berhasil dihapus</h2>
</body>
</html>
```

- Menjalankan delete student

Nama : Desy Ulfasari
NPM : 1406557503
Tutorial 3 APAP Ekstensi



Tampilkan sebuah halaman yang memberikan informasi bahwa data tersebut telah berhasil dihapus. Jika nomor NPM tidak diberikan atau tidak ditemukan kembalikan halaman error yang berisi informasi bahwa nomor NPM kosong atau tidak ditemukan dan proses delete dibatalkan.



Method selectStudent yang Anda implementasikan

```
@Override
public StudentModel selectStudent(String npm) {
    for (StudentModel student : studentList) {
        if (student.getNpm ().equals (npm)) {
            return student;
        }
    }
    return null;
}
```

Method tsb digunakan untuk mengambil npm, dengan mengambil npm input yang dicocokkan dengan npm yang ada di arraylist

Penjelasan fitur delete yang Anda buat pada bagian latihan

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

Fitur delete student dengan mengloop isi dari arraylist studentlist. kemudian apabila ada npm input yang sama dengan npm yang ada di dalam array list, maka dengan method remove arraylist, arraylist pada ke i akan di hapus.