**Membuat​ ​Class​ ​Model**

1. Method selectStudent setelah berhasil di implementasikan:



Pada metode berikut, mengiterasikan arraylist untuk menemukan mahasiswa dengan NPM yang sama dengan yang dimasukan dalam parameter, jika ada yang sama kemudian *return* dari mahasiswa yang didapatkan, jika tidak maka NULL menandakan bahwa tidak ada mahasiswa tersebut.

**Membuat Service**

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

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



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

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

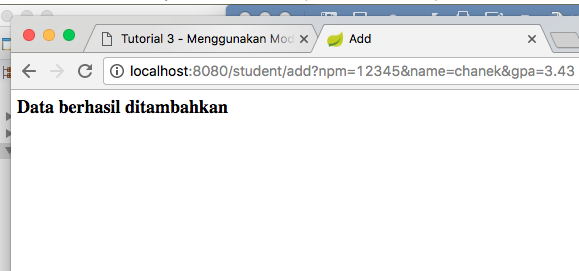


Terjadi Whitelabel Error Page (type=Bad Request, status=400), dibutuhkan parameter ‘gpa’ untuk RequestParam GET

**Method​ ​View​ ​by​ ​NPM**

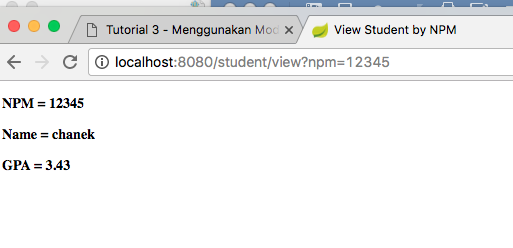
1. Jalankan program dan buka

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



lalu buka

localhost:8080/student/view?npm=12345



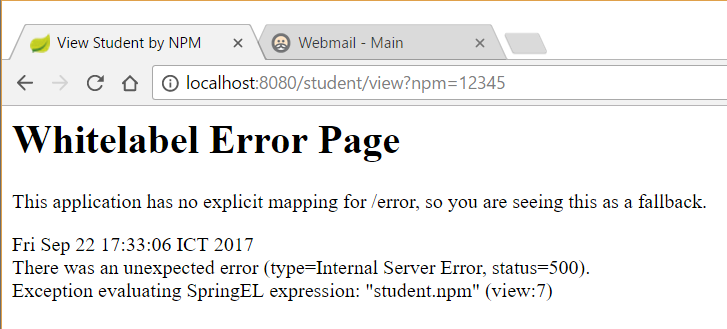
Pertanyaan​ ​3​: apakah data Student tersebut muncul? Jika tidak, mengapa?

Muncul.

Data student yang dimaksud muncul sesuai dengan yang sebelumnya dimasukkan pada parameter

**npm=12345&name=chanek&gpa=3.43**

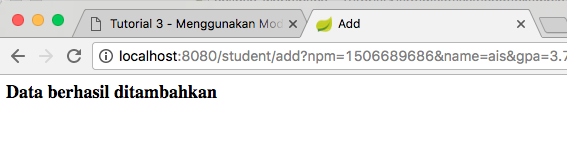
1. Coba matikan program dan jalankan kembali serta buka localhost:8080/student/view?npm=12345



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

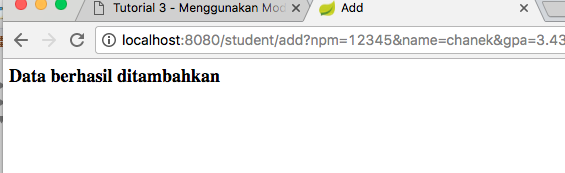
Tidak. Data student ada pada session setiap kali springboot dijalankan. Ketika springboot dimatikan, maka session akan diperbaharui sepenuhnya.

1. Coba tambahkan data Student lainnya dengan NPM yang berbeda.

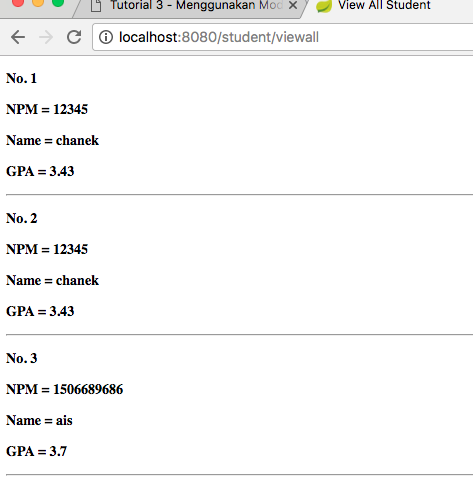


**Method​ ​View​ ​All**

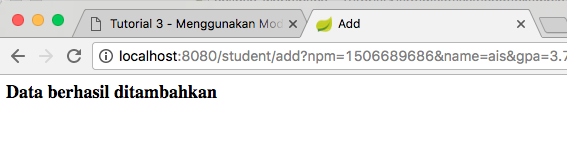
1. Jalankan program dan buka localhost:8080/student/add?npm=12345&name=chanek&gpa=3.43

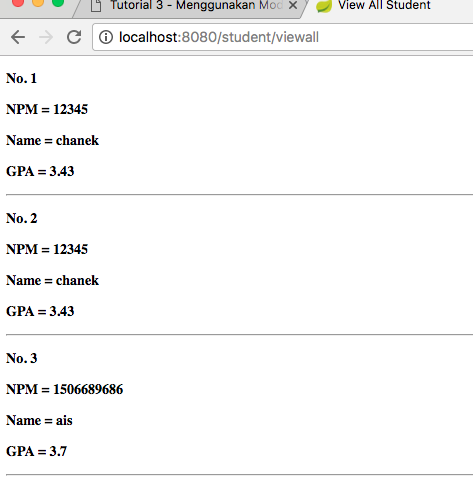


lalu buka localhost:8080/student/viewall



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

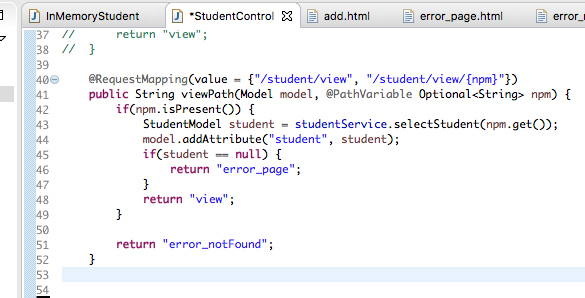
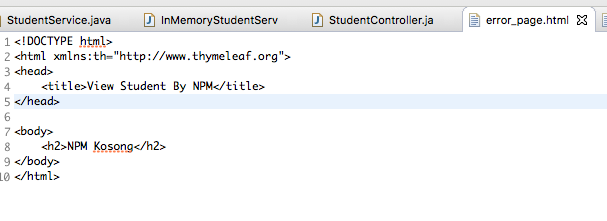
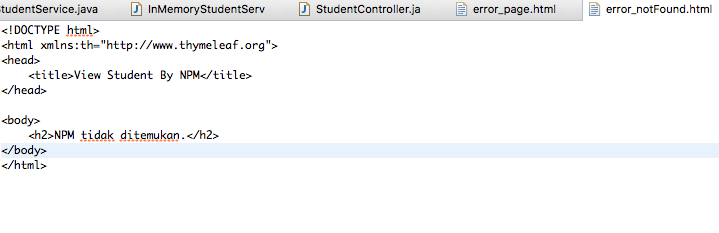


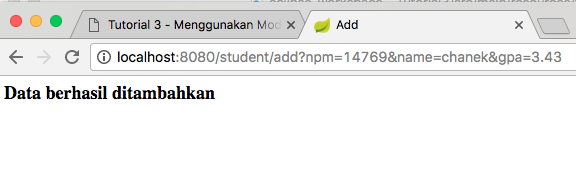
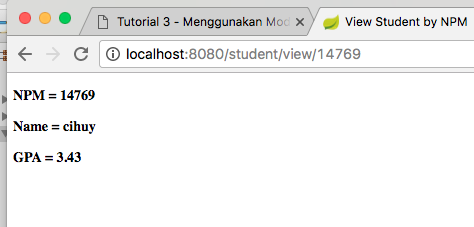


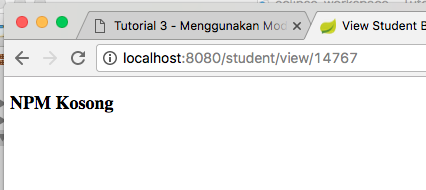
Pertanyaan​ ​6​: Apakah semua data Student muncul?

**Muncul**

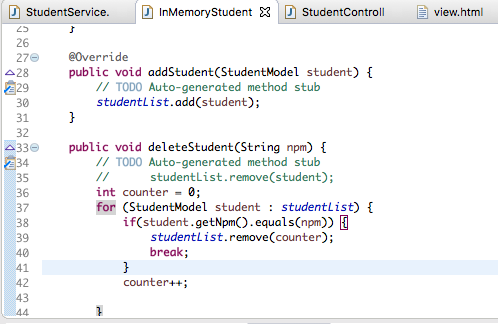
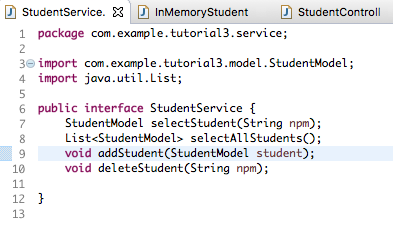
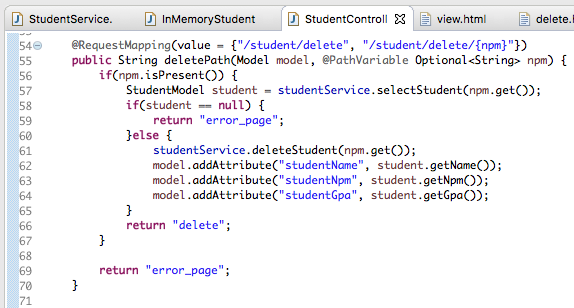
**Latihan**

1. Pada StudentController ​tambahkan sebuah method view Student dengan menggunakan Path Variable​.  
     
   Method yang ditambahkan  
     
     
   bikin halaman error\_page.html dan error\_notFound.html  
     
     
     
     
   Tambahkan npm 14769

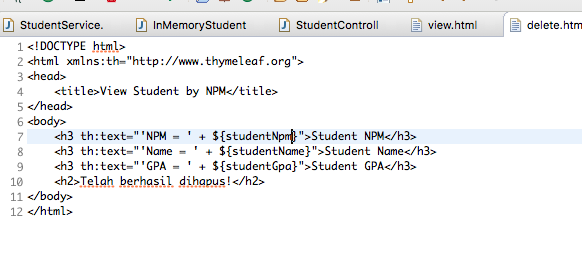
  
Cek apakah student dengan npm tersebut sudah masuk  


Ketika menulis npm yg tidak terdaftar maka akan menampilkan halaman berikut

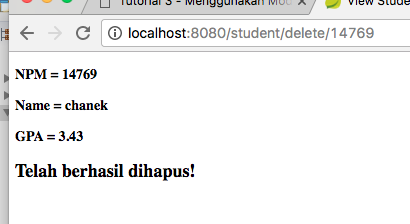
1. Tambahkan fitur untuk melakukan delete Student berdasarkan NPM. Misalnya, setelah melakukan add Student pada soal nomor 1, cobalah untuk melakukan delete data tersebut dengan mengakses halaman localhost:8080/student/delete/14769. ​Tampilkan sebuah halaman yang memberikan informasi bahwa data tersebut telah berhasil dihapus.  
   Buat method baru deleteStudent

  
  
  
Buat method deletePath pada StudentController  
  


Halaman html yang akan ditampilkan:



Contoh halaman html yang akan ditampilkan



**Lesson Learned**  
Dalam tutorial 3 kali ini, saya belajar cara pengimplementasian interface yang baik menggunakan MVC. Saya belajar mengenaik class Model apa saja yang terdapat dalam MVC. Selain itu, saya mengimplementasikan cara menggunakan java util. Saya belajar cara penggunaan looping pada view. Selain itu, Saya melakukan requestparam dan pathvariable untuk penyimpanan data untuk sprinboot