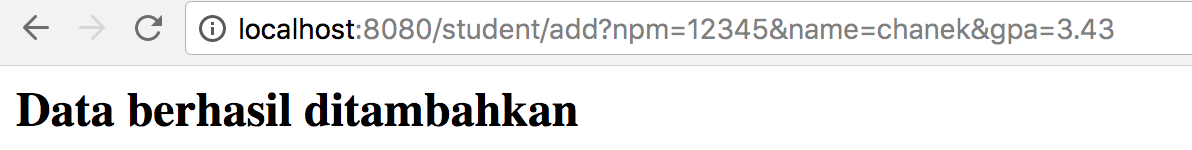
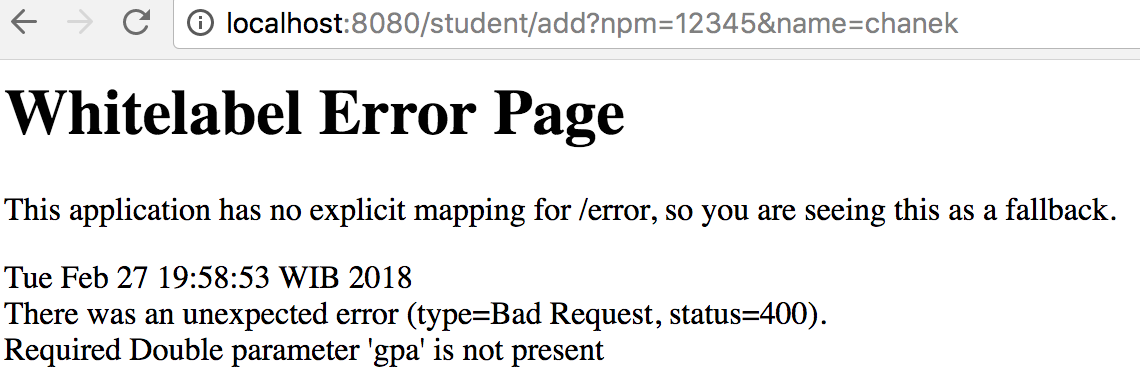
Pada tutorial 3 ini saya mempelajari implementasi dari konsem MVC pada spring boot. Fokus dari tutorial 3 yaitu pada model dan service, dimana model merupakan suatu objek yang merepresentasikan sesuatu. Sedangkan service yaitu suatu layer yang menjadi mediator antara controller dan database. Kemudian controller berisi method-method yang telah dideklarasikan pada service kemudian dioverride dengan implementasi setiap methodnya. Pada tutorial kali ini untuk menyimpan datanya belum menggunakan database, akan tetapi menggukan list. List yang dibuat yaitu list of StudentModel.

1. Jalankan program dan buka, localhost:8080/student/add?npm=12345&name=chanek&gpa=3.43. apakah hasilnya? Jika error, tuliskan penjelasan Anda.

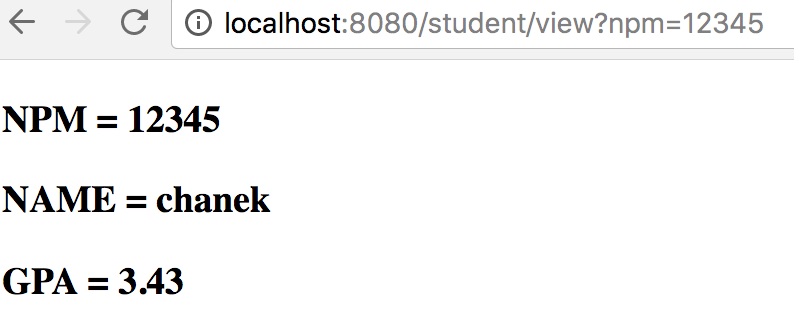
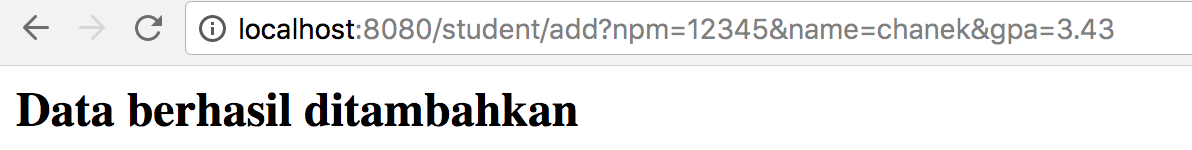


1. Jalankan program dan buka, localhost:8080/student/add?npm=12345&name=chanek. apakah hasilnya? Jika error, tuliskan penjelasan Anda.

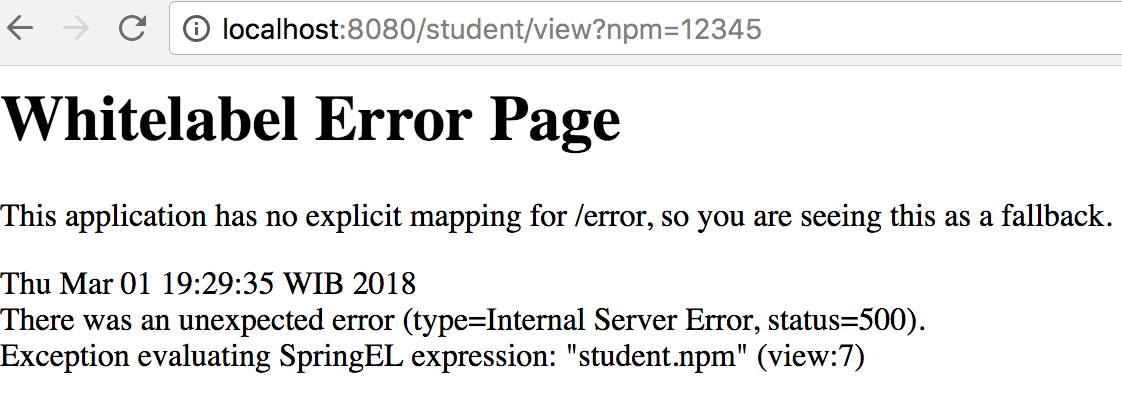


Error dikarenakan parameter yang dibutuhkan sebanyak 3 dan dari 3 parameter itu *required*. Sedangkan yang dijalankan hanya memberikan 2 parameter saja (npm dan name), seharusnya 3 yaitu npm, name, dan gpa.

1. Jalankan program dan buka, localhost:8080/student/add?npm=12345&name=chanek&gpa=3.43. lalu buka localhost:8080/student/view?npm=12345. apakah data Student tersebut muncul? Jika tidak, mengapa?

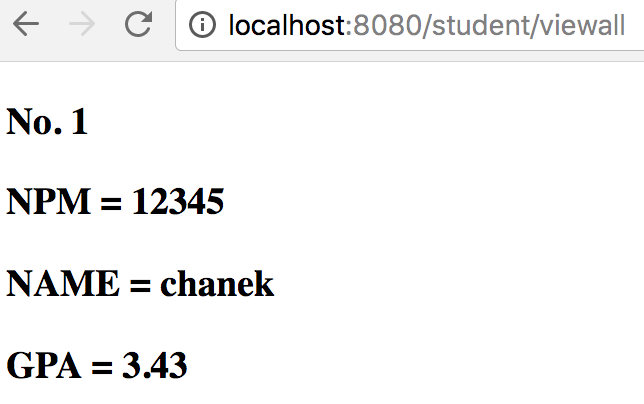


1. Coba matikan program dan jalankan kembali serta buka localhost:8080/student/view?npm=12345.

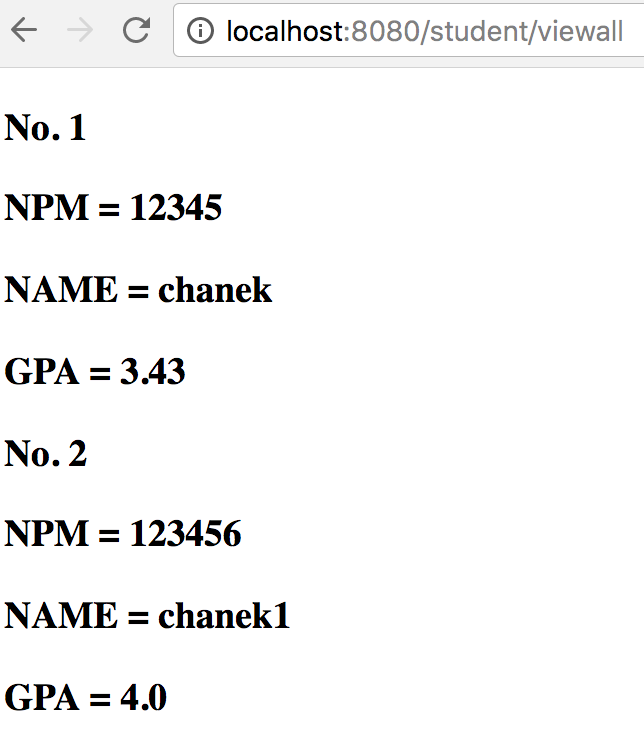
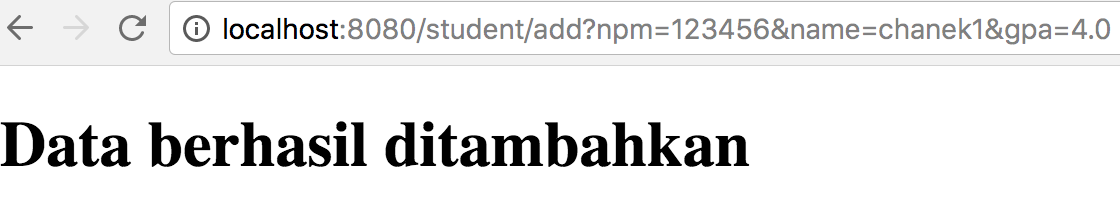


Dikarenakan ketika program dimatikan dan dijalankan kembali, maka list of StudentModel kembali kosong, maka ketika kita mencari data tidak akan ditemukan (return null).

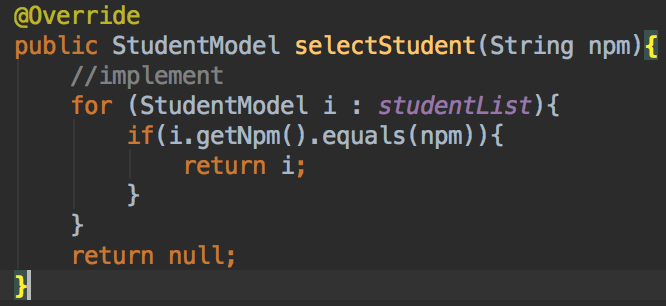
1. Jalankan program dan buka localhost:8080/student/add?npm=12345&name=chanek&gpa=3.43 lalu buka localhost:8080/student/viewall. apakah data Student tersebut muncul?



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



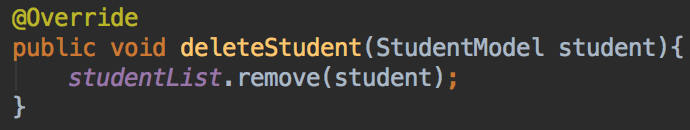
1. Implementasi dari method selectStudent().



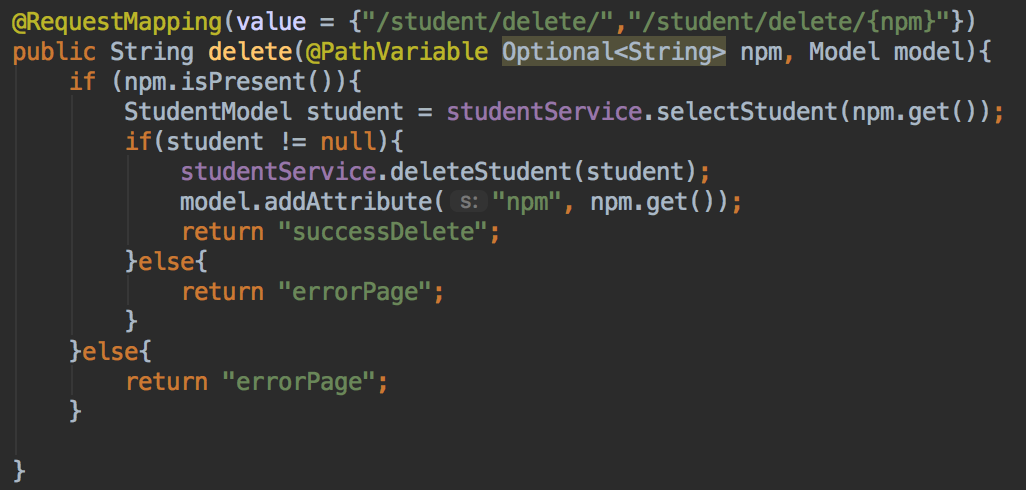
1. Pembuatan fitur *delete* dengan cara membuat *service* deleteStudent(StudentModel student) pada file StudentService.java.



Kemudian di*override* pada InMemoryStudentService.java, dimana saat *override* diimplementasikan isi dari methodnya.



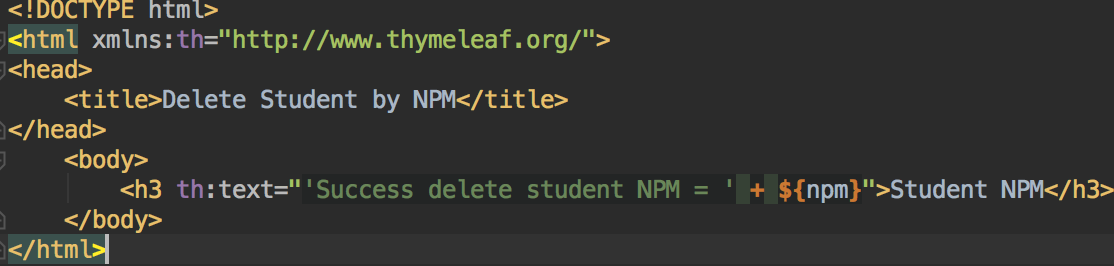
Setelah itu, membuat method pada controller nya difile StudentController.java.



Pada method delete diatas, terdapat 2 return yang berbeda, yaitu successDelete dan errorPage. Maka buat 2 file successDelete.html dan errorPage.html pada folder template.



Isi dari file successDelete.html



Sedangkan isi dari file errorPage.html

