Sukrian Syahnel Putra
1306402066
ADPAP - B

Write-Up Tutorial 3: Menggunakan Model dan Service pada Project Spring Boot

Ringkasan

Di dalam tutorial ini saya mempelajari kegunaan Model dan Service serta implementasi konsep MVC (model-view-controller) dan integrasi kedua elemen tersebut dengan Controller yang ada di Project Spring Boot.

Secara garis besar, Model berfungsi sebagai representasi sebuah objek dan menyimpan berbagai macam informasi mengenai objek tersebut. Misalnya, di dalam tutorial ini seorang mahasiswa direpresentasikan sebagai sebuah objek dengan atribut-atribut berupa nama, NPM, dan GPA dari mahasiswa tersebut.

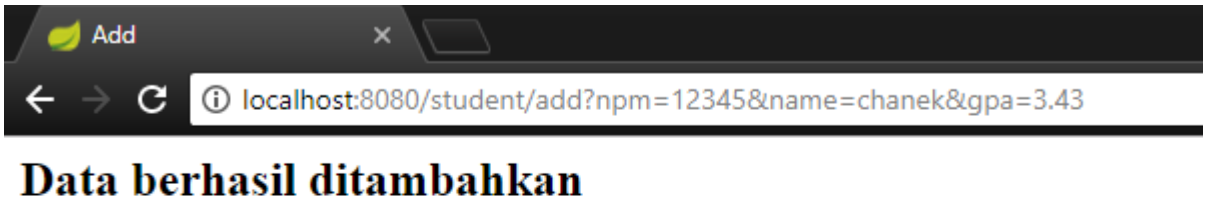
Service merupakan jembatan yang menghubungkan antara Model dan Controller yang berisi daftar tindakan yang dapat dilakukan pada sebuah Model. Di dalam tutorial ini, beberapa service yang dapat dilakukan adalah menambahkan, melihat dan menghapus data mahasiswa.

Controller berfungsi menjalankan service tersebut dan menampilkan ke dalam bentuk View. Segala aktivitas di dalam Service dijalankan melalui sebuah Controller untuk kemudian ditampilkan ke web page.

Di dalam tutorial ini hal-hal yang telah saya pelajari antara lain membuat Model, membuat Service, dan mengimplementasikan setiap Service yang dapat dilakukan pada Model.

Jawaban dari setiap pertanyaan

1. Setelah menerapkan metode add , ketika kita menjalankan program dan membuka localhost:8080/student/add?npm=12345&name=chanek&gpa=3.43 di web browser, apa hasilnya?
   Jawaban: Saya menerima halaman “add” dengan normal seperti tertera pada screenshot berikut.



2. Ketika kita menjalankan program dan membuka localhost:8080/student/add?npm=12345&name=chanek di web browser, apa hasilnya?
   Jawaban: Saya menerima sebuah page error 400 yang mengindikasikan “bad request”. Hal ini terjadi karena parameter yang diperlukan, yaitu GPA, tidak dimasukkan. Berikut penjelasan page error 400 tersebut:

Sukrian Syahnel Putra
1306402066
ADPAP - B

Whitelabel Error Page

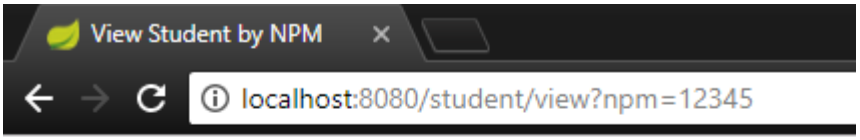
This application has no explicit mapping for /error, so you are seeing this as a fallback.

Fri Sep 22 16:03:37 ICT 2017

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

Required double parameter 'gpa' is not present

3. Setelah menerapkan metode “view”, kemudian menjalankan program, kemudian membuka localhost:8080/student/add?npm=12345&name=chanek&gpa=3.43 lalu localhost:8080/student/view?npm=12345, apakah data student muncul?
   Jawaban: Data student berhasil muncul di web page, seperti tertera pada gambar berikut.



NPM = 12345

Name = chanek

GPA = 3.43

4. Setelah menjalankan instruksi sebelumnya, kemudian mematikan dan menjalankan kembali program dan membuka localhost:8080/student/view?npm=12345, apakah data student muncul?
   Jawaban: Data student tersebut tidak muncul. Hal ini terjadi karena data student yang sebelumnya tidak di-store ke dalam database eksternal dan hanya disimpan ke dalam List yang ada di dalam class, sehingga data bersifat sementara. Hal ini menyebabkan data tersebut dihapus ketika program dimatikan atau di-restart.
5. Setelah mengimplementasi metode view all di tutorial, kemudian menjalankan program dan membuka localhost:8080/student/add?npm=12345&name=chanek&gpa=3.43 kemudian localhost:8080/student/viewall, apakah data student tersebut muncul?
   Jawaban: Data student muncul secara normal, seperti terlihat pada gambar berikut.

Sukrian Syahnel Putra
1306402066
ADPAP - B

No. 1

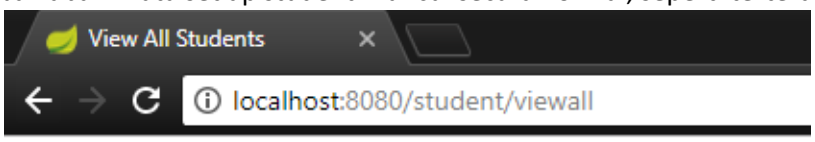
NPM = 12345

Name = chanek

GPA = 3.43

---

6. Tambahkan data student yang lainnya, kemudian jalankan localhost:8080/student/viewall. Apakah data semua student muncul?
   Jawaban: Data setiap student muncul secara normal, seperti tertera pada gambar berikut.



No. 1

NPM = 12345

Name = chanek

GPA = 3.43

No. 2

NPM = 11234

Name = kubis

GPA = 3.33

No. 3

NPM = 15320

Name = wortel

GPA = 3.54

Sukrian Syahnel Putra
1306402066
ADPAP - B
Method selectStudent

Berikut merupakan pseudocode dari metode selectStudent yang saya implementasikan dalam class InMemoryStudentService:

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

Penjelasan fitur delete yang dibuat:

- Karena di dalam service belum ada metode untuk menghapus daftar student, maka hal pertama yang saya lakukan adalah membuat metode di interface StudentService untuk kemudian di-implement oleh class InMemoryStudentService.

```
boolean removeStudent(String npm);
```

- Setelah saya membuat metode di interface, saya kemudian mengimplementasi metode tersebut di class InMemoryStudentService.

```
@Override
public boolean removeStudent(String npm) {
	StudentModel target = this.selectStudent(npm);
	if (target == null) {
		return false;
	}
	studentList.remove(target);
	return true;
}
```

- Pada metode di atas, saya menerapkan boolean untuk return type karena hasil penghapusan ini perlu ditampilkan di web page nantinya. Jika return dari metode adalah true, hal ini menyatakan bahwa data tersebut ada dalam sistem dan berhasil dihapus. Sementara itu, jika data yang ingin dihapus tidak ditemukan, maka akan me-return boolean false. Pencarian data student menggunakan metode selectStudent yang telah diterapkan di class ini.
- Setelah itu, saya membuat metode baru di class StudentController untuk menjalankan metode removeStudent yang telah dibuat.

```
//	Delete using Path Variable
	@RequestMapping(value = {"/student/delete", "/student/delete/{npm}"})
	public String deletePath(@PathVariable String npm, Model model) {
		model.addAttribute("npm", npm);
		boolean deleted = studentService.removeStudent(npm);
		if (!deleted) {
			return "notfound";
		}
		return "delete";
```

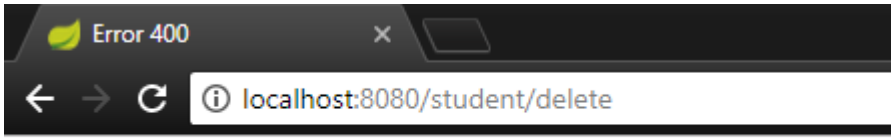
```
		}
```

- Atribut NPM ditambahkan ke dalam model untuk keperluan menampilkan kondisi data dengan NPM yang akan dihapus nanti (berhasil atau data tidak ditemukan).
- Setelah itu, saya membuat dua halaman di resources/templates. Halaman yang pertama adalah delete.html untuk menampilkan kondisi di mana data student berhasil dihapus, dan notfound.html untuk menampilkan kondisi di mana data student tidak ditemukan.
- Ketika saya menjalankan metode delete, ketika saya mencoba membuka localhost:8080/student/delete, program mengembalikan exception berupa MissingPathVariableException dan halaman yang ditampilkan adalah halaman error yang di-generate secara otomatis oleh Spring Boot. Tentunya bukan ini output yang diharapkan.
- Agar output yang muncul berupa halaman peringatan, saya kemudian mengimplementasi sebuah metode untuk meng-handle exception ini.

```
//	Handle missing path variable exceptions
	@ExceptionHandler(MissingPathVariableException.class)
	public @ResponseBody String
handleMissingPathVars(MissingPathVariableException ex) {
		String missingPath = ex.getVariableName();
		if (missingPath.equals("npm")) {
			missingPath = "NPM";
		} else if (missingPath.equals("name")) {
			missingPath = "Nama";
		} else if (missingPath.equals("gpa")) {
			missingPath = "GPA";
		}
		return "<!DOCTYPE html>\r\n" +
				"<html xmlns:th = \"http://www.thymeleaf.org\">\r\n" +
				"	<head>\r\n" +
				"	<title>Error 400</title>\r\n" +
				"	</head>\r\n" +
				"	<body>\r\n" +
				"		<h3>Tolong masukkan path " + missingPath +
".</h3>\r\n" +
				"	</body>\r\n" +
				"</html>";
		}
```

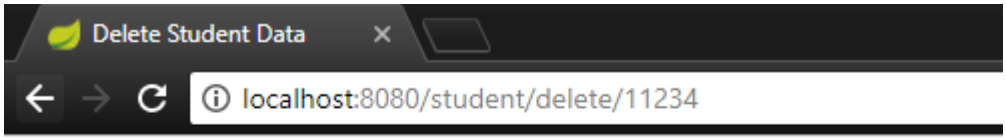
- Implementasi metode ini berfungsi mengubah output yang tadinya auto-generated menjadi output yang kita kehendaki ketika kondisi error muncul. Metode ini diterapkan secara universal untuk semua metode di dalam kelas StudentController, jadi metode-metode lain yang membutuhkan path variable akan mengembalikan metode ini jika muncul MissingPathVariableException.
- Setelah itu, saya menjalankan kembali program. Berikut merupakan screenshot halaman yang muncul ketika dilakukan operasi delete.

Sukrian Syahnel Putra
1306402066
ADPAP - B



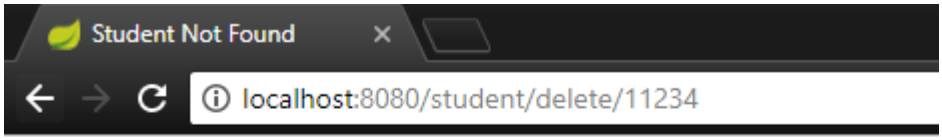
Tolong masukkan path NPM.

Screenshot ketika muncul MissingPathVariableException.



Data mahasiswa dengan NPM 11234 berhasil dihapus!

Screenshot ketika data yang ada di dalam program berhasil dihapus.



Mahasiswa dengan NPM 11234 tidak ditemukan.

Screenshot ketika data yang akan dihapus tidak ditemukan.