Tutorial 2

APAP

­­Latihan Project Hello World

1. Ganti baris tersebut menjadi @RequestMapping(“/hello123”)

Pertanyaan:

* Apakah compile error? **Tidak**
* Jika tidak, stop Spring Boot yang sedang berjalan, run kembali dan buka localhost: 8080/hello apa yang terjadi? **Terdapat “Whitelabel Error Page”**

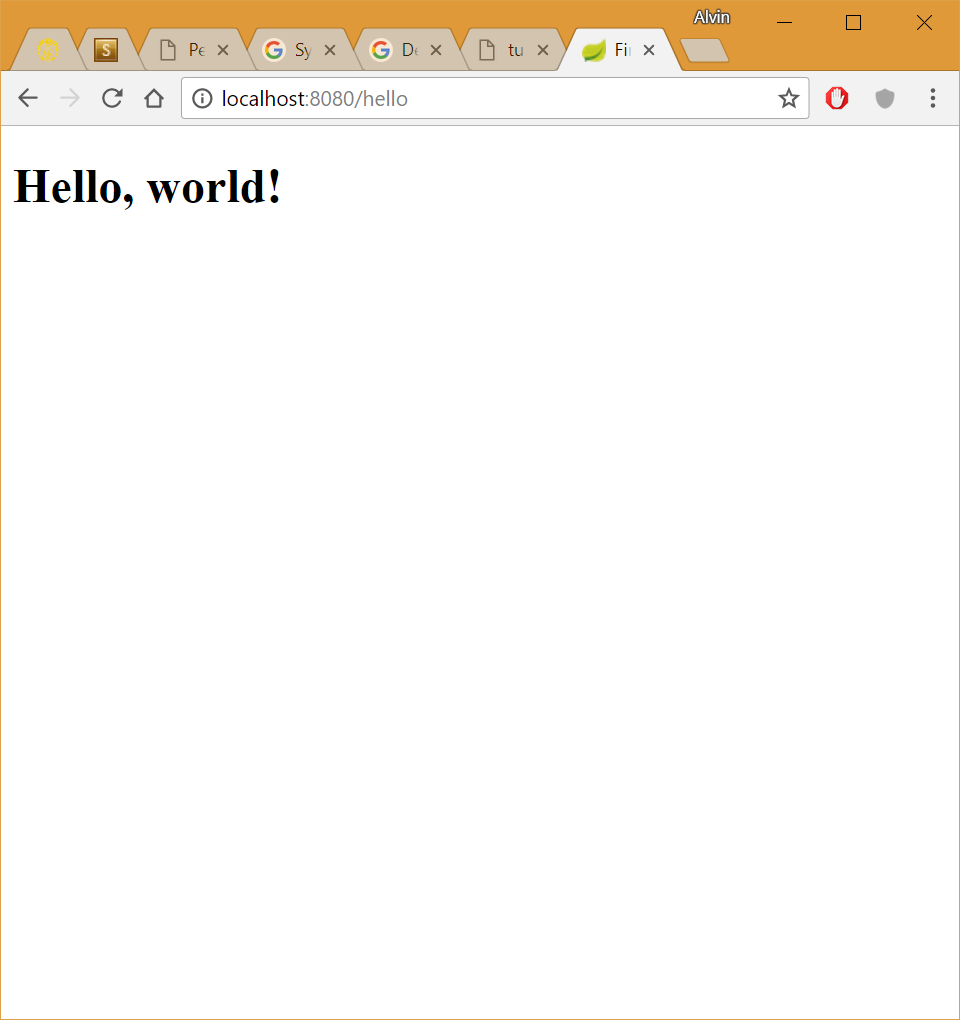


* Kembalikan RequestMapping menjadi /hello **Done**

2. Ganti nama method index() dengan nama method hello()

Pertanyaan:

* Apakah compile error? **Tidak**
* Jika tidak, Stop Spring Boot yang sedang berjalan, run kembali dan buka localhost:8080/hello apakah page hello sebelumnya masih muncul? **Tidak**

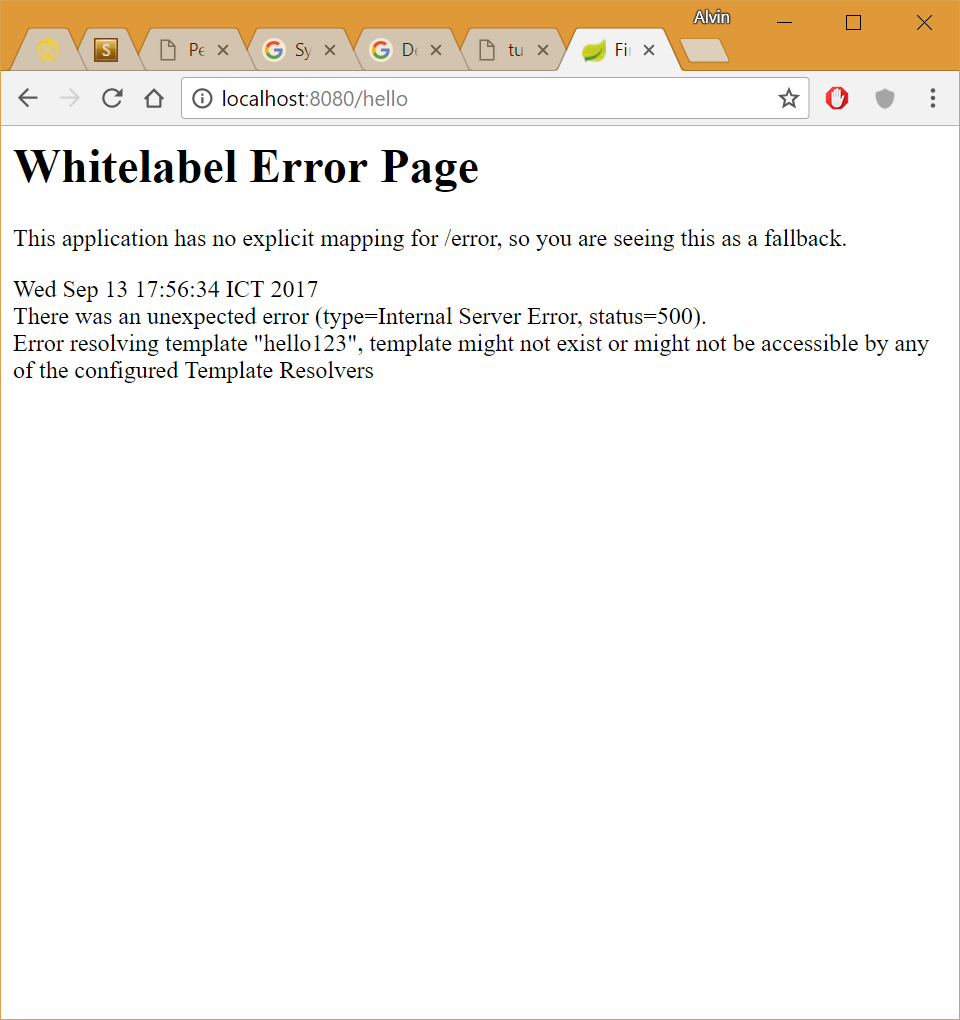


* Kembalikan nama method menjadi index() **Done**

3. Ganti string return type menjadi return “hello123”

Pertanyaan:

* Apakah compile error? **Tidak**
* Jika tidak, Stop Spring Boot yang sedang berjalan, run kembali dan buka localhost:8080/hello apakah page hello sebelumnya masih muncul? **Tidak. Muncul “Whitelabel Error Page”**



* Kembalikan return type menjadi “hello” **Done**

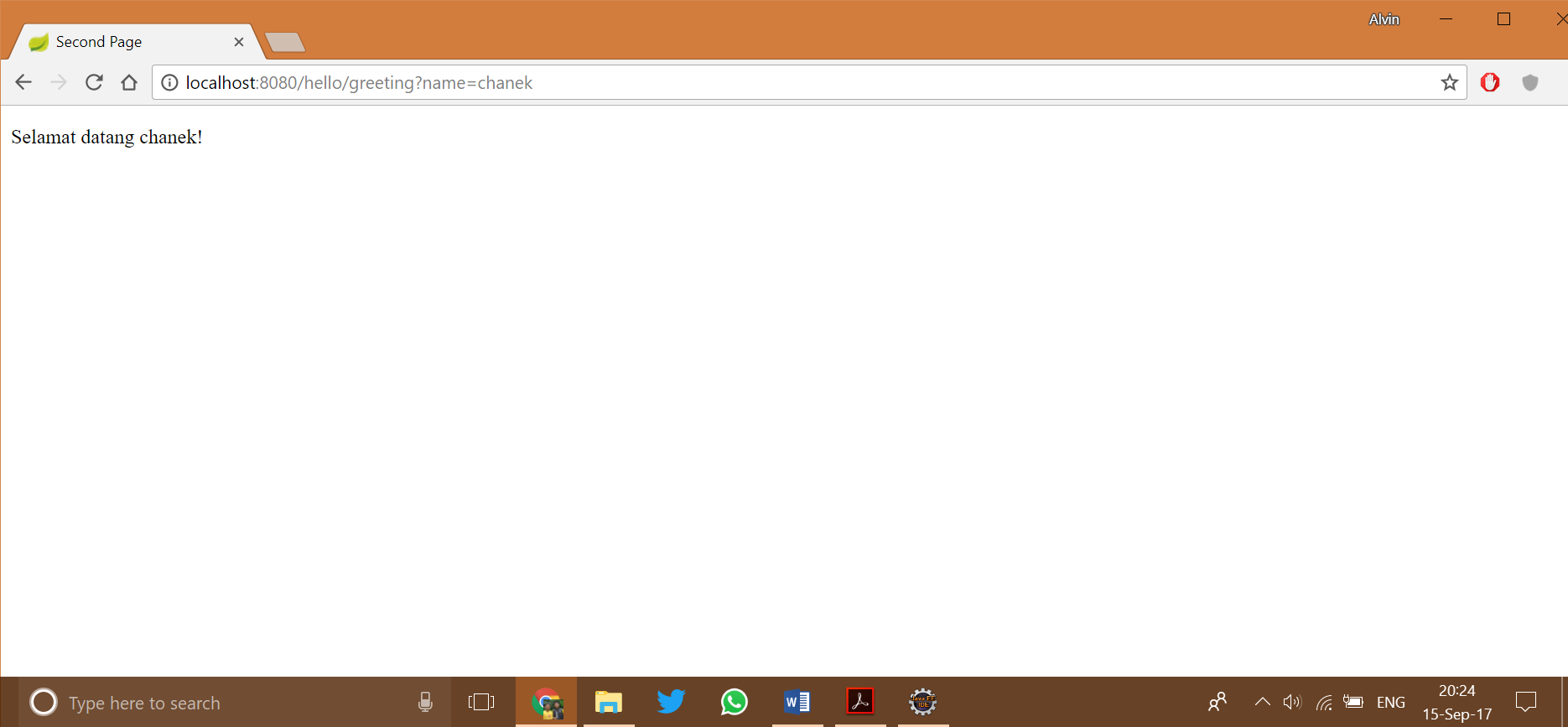
­­Latihan Request Parameter

1. Ubah nilai anotasi RequestMapping dari “/greeting” menjadi “/hello/greeting”

Buka localhost:8080/hello/greeting?name=chanek

Pertanyaan:

* Apakah hasilnya? **Muncul “Selamat datang chanek!”**

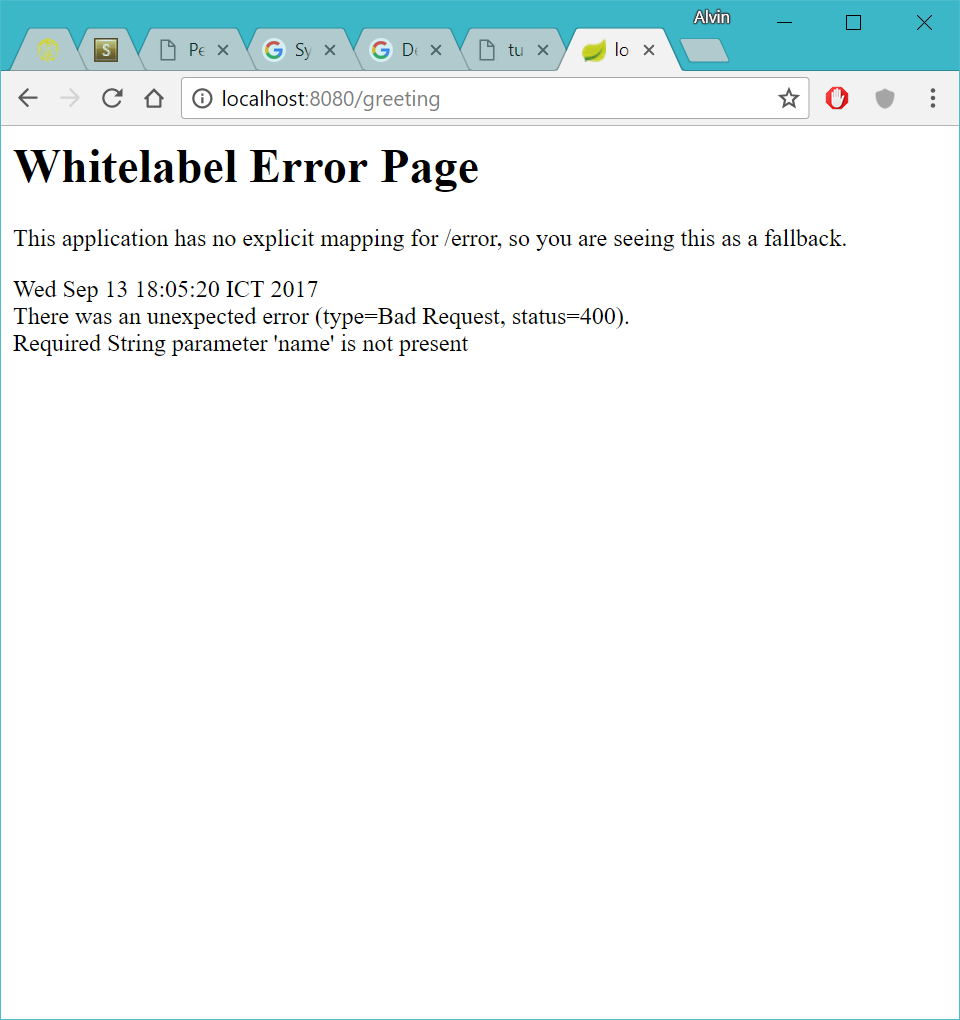


* Kembalikan request map menjadi @RequestMapping(“/greeting”) **Done**

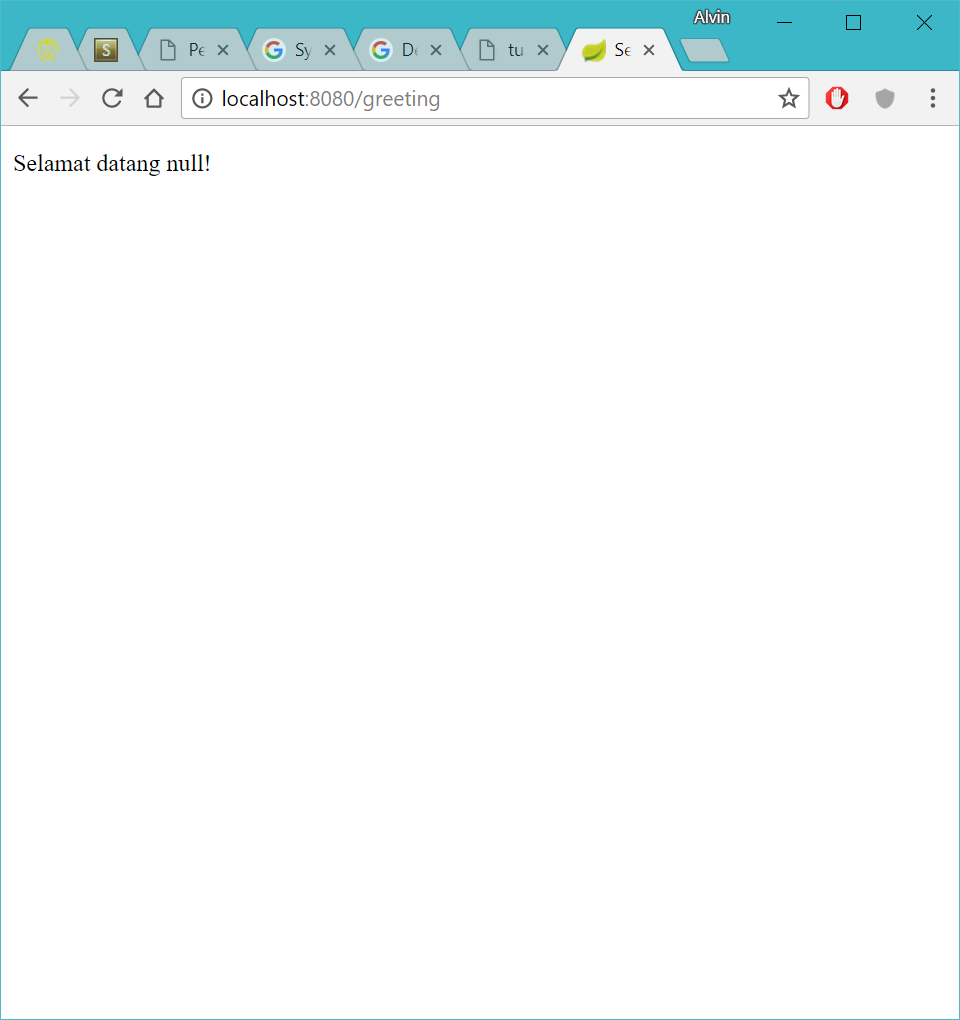
2. Akses localhost:8080/greeting

Pertanyaan:

* Apakah hasilnya? **Terdapat “Whitelabel Error Page”**



* Ubah header method greeting menjadi seperti berikut
  + public String greeting (@RequestParam(value = “name”, required = false) String name, Model model) **Done**
  + Stop Spring Boot yang sedang berjalan, run kembali, buka localhost:8080/greeting **Done**
  + Apakah hasilnya? **Menampilkan “Selamat datang null!”**



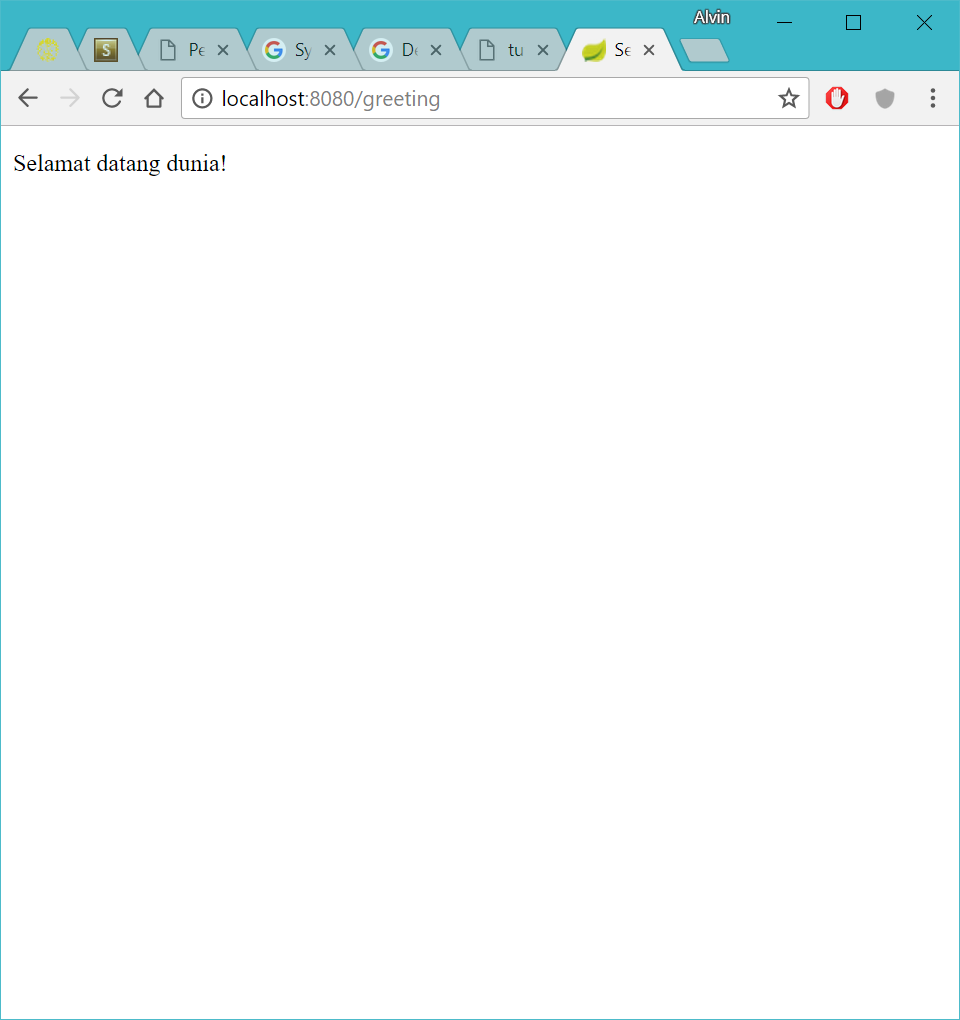
3. Ubah header method greeting menjadi seperti berikut

public String greeting (@RequestParam(value = “name”, required = false, defaultValue = “dunia”) String name, Model model)

Stop Spring Boot yang sedang berjalan, run kembali, dan buka localhost:8080/greeting.

Pertanyaan:

* Apakah hasilnya? **Menampilkan “Selamat datang dunia!”**



4. Perhatikan bahwa pada berkas greeting.html, tag paragraf yang kita tambahkan adalah sebagai berikut:

<p th:text=”’Selamat datang ‘ + ${name} + ‘!’”>Sapaan untuk user</p>

Pertanyaan:

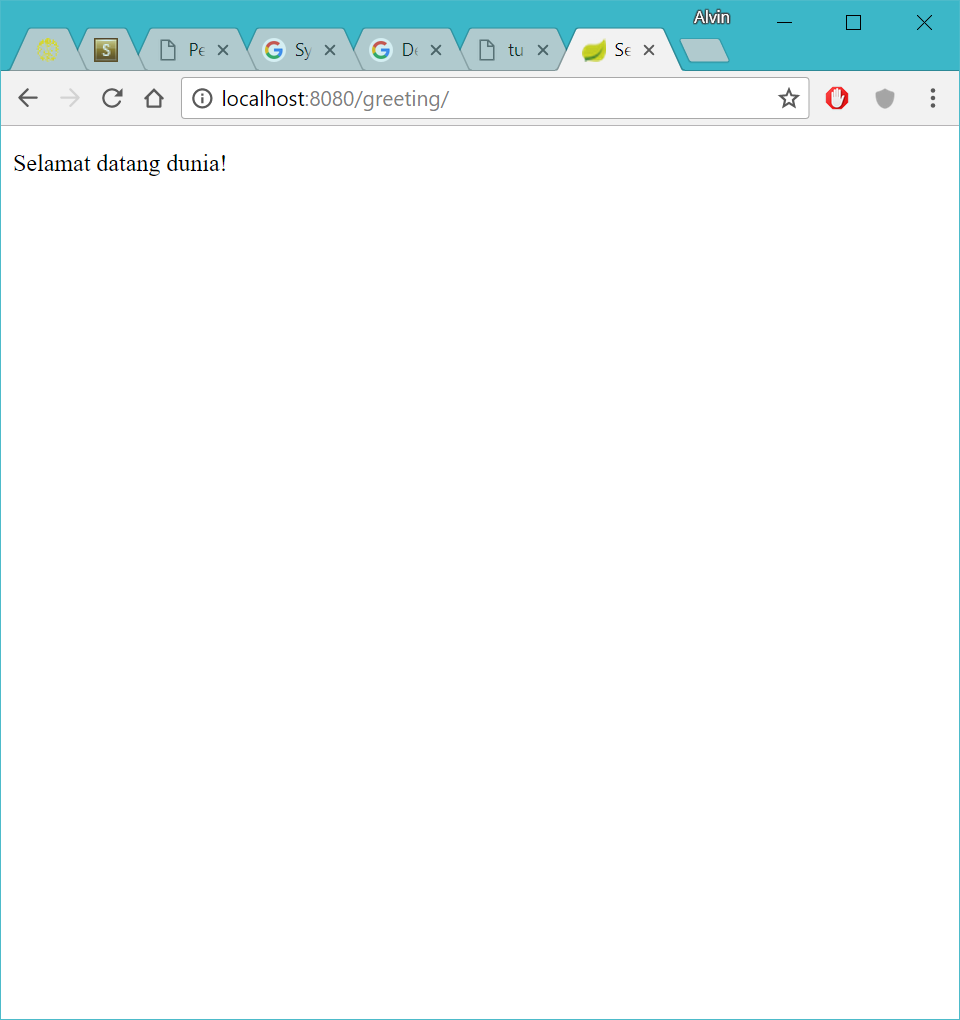
* Mengapa tulisan “Sapaan untuk user” tidak pernah muncul? **Karena th:text milik Thymeleaf akan mengganti seluruh teks yang ada pada tag dimana atribut tersebut ditempatkan. Pada contoh ini, th:text akan mengganti seluruh teks yang ada pada tag p menjadi value yang ada pada atribut tersebut**

­­Latihan Path Variable

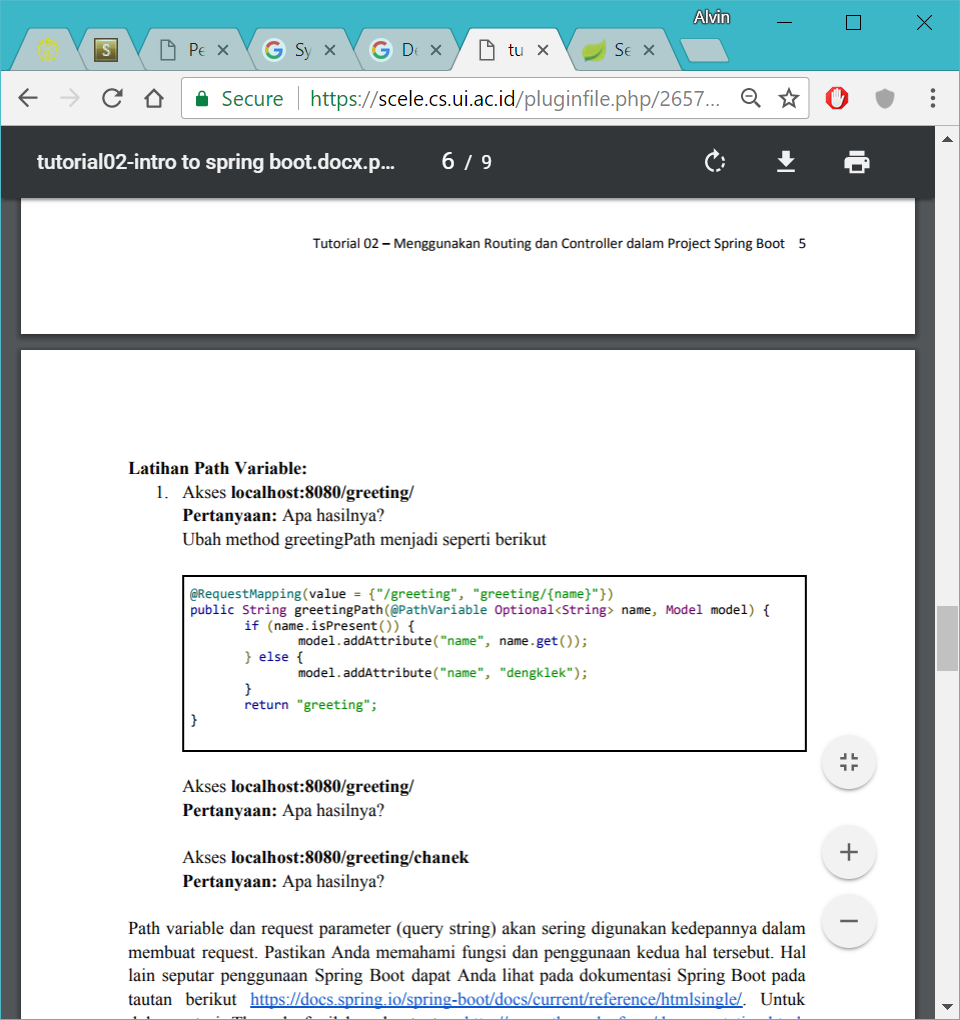
1. Akses localhost:8080/greeting/

Pertanyaan:

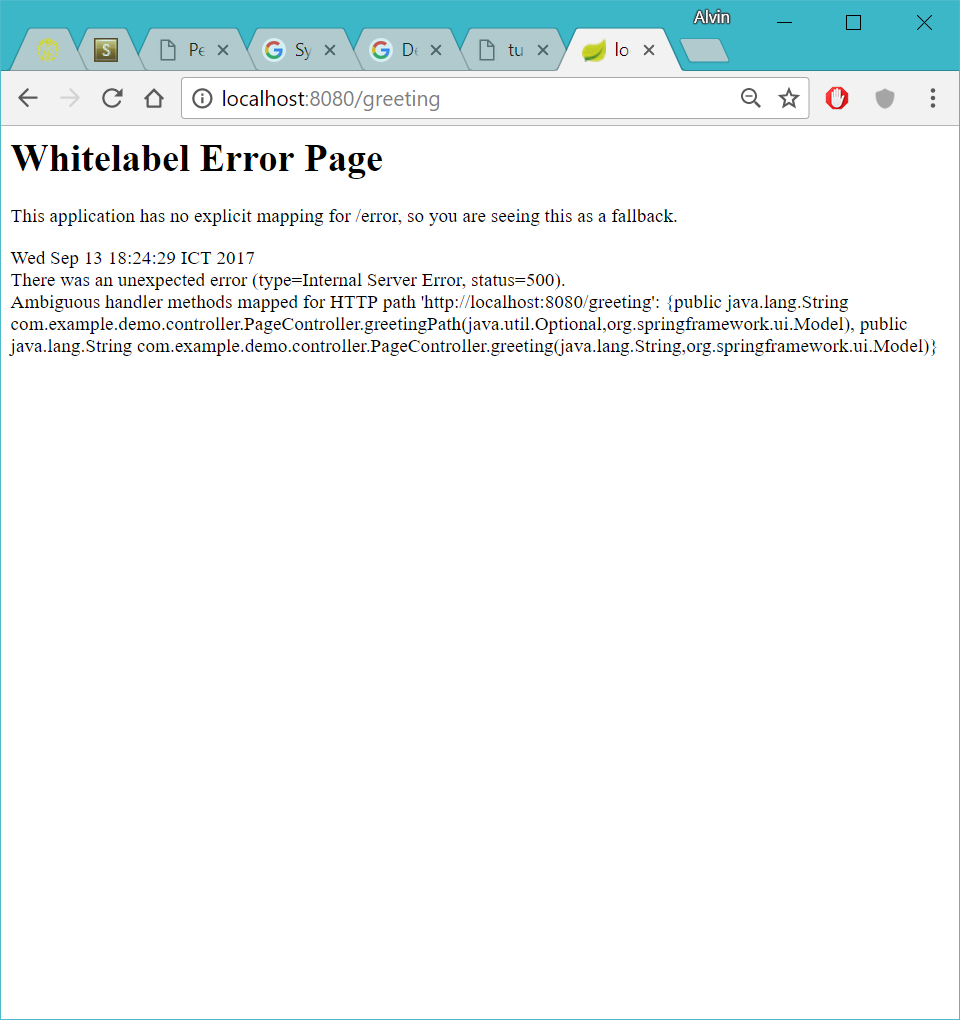
* Apa hasilnya? **Menampilkan “Selamat datang dunia!”**



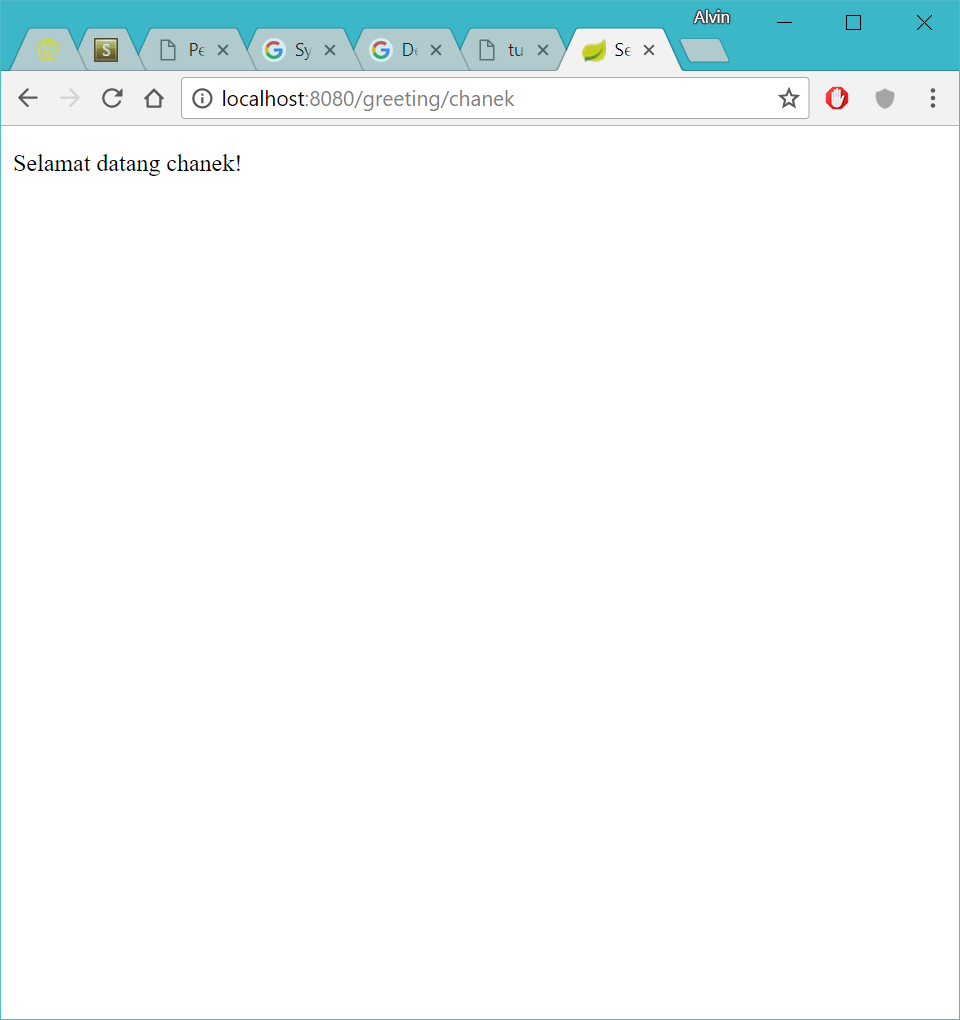
* Ubah method greetingPath menjadi seperti berikut



* + Akses localhost:8080/greeting/
  + Apa hasilnya? **Muncul “Whitelabel Error Page”**



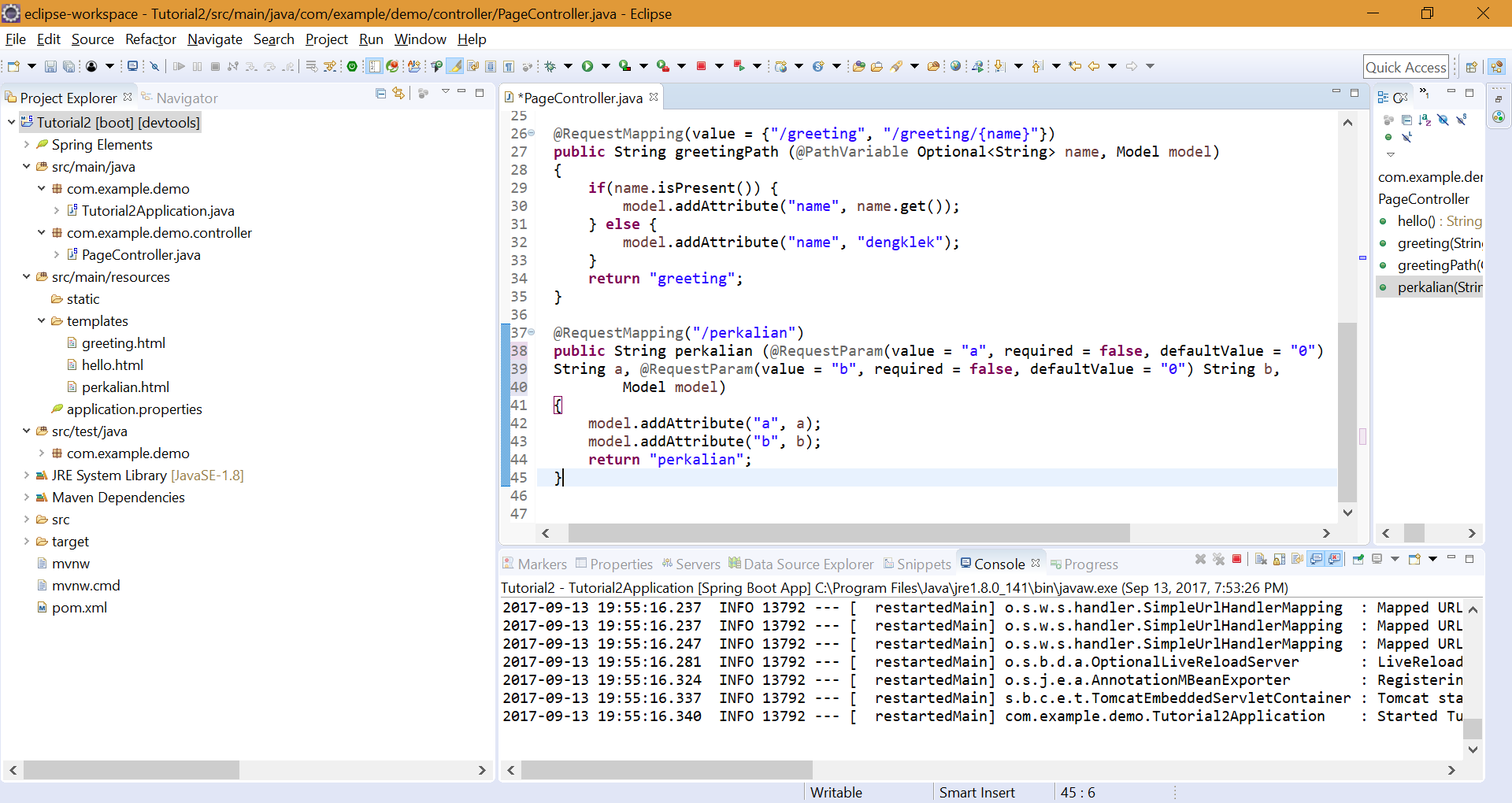
* + Akses localhost:8080/greeting/chanek
  + Apa hasilnya? **Muncul “Selamat datang chanek!”**



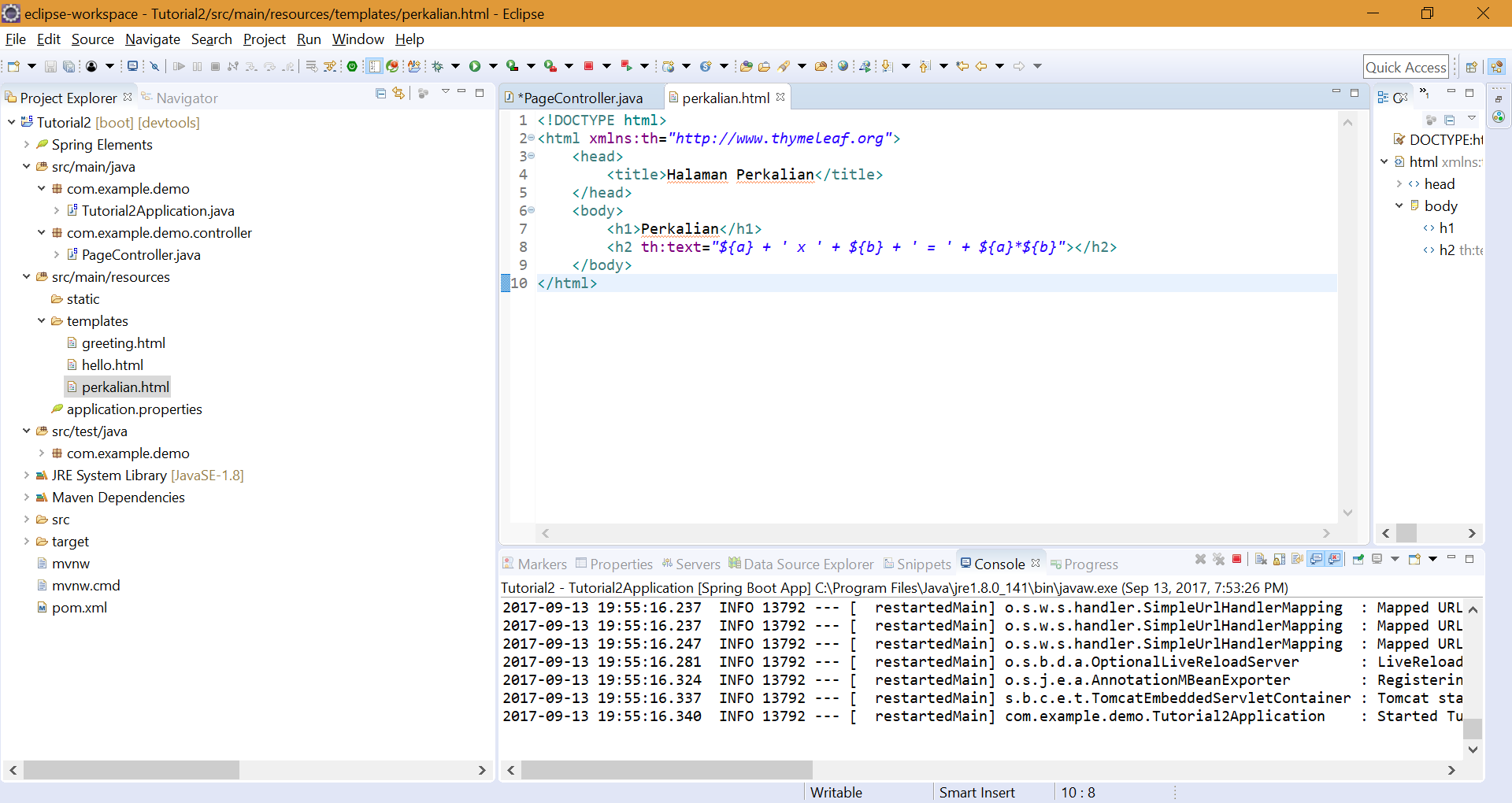
­­Latihan Perkalian

1. Buatlah satu halaman baru untuk melakukan operasi perkalian **Done**

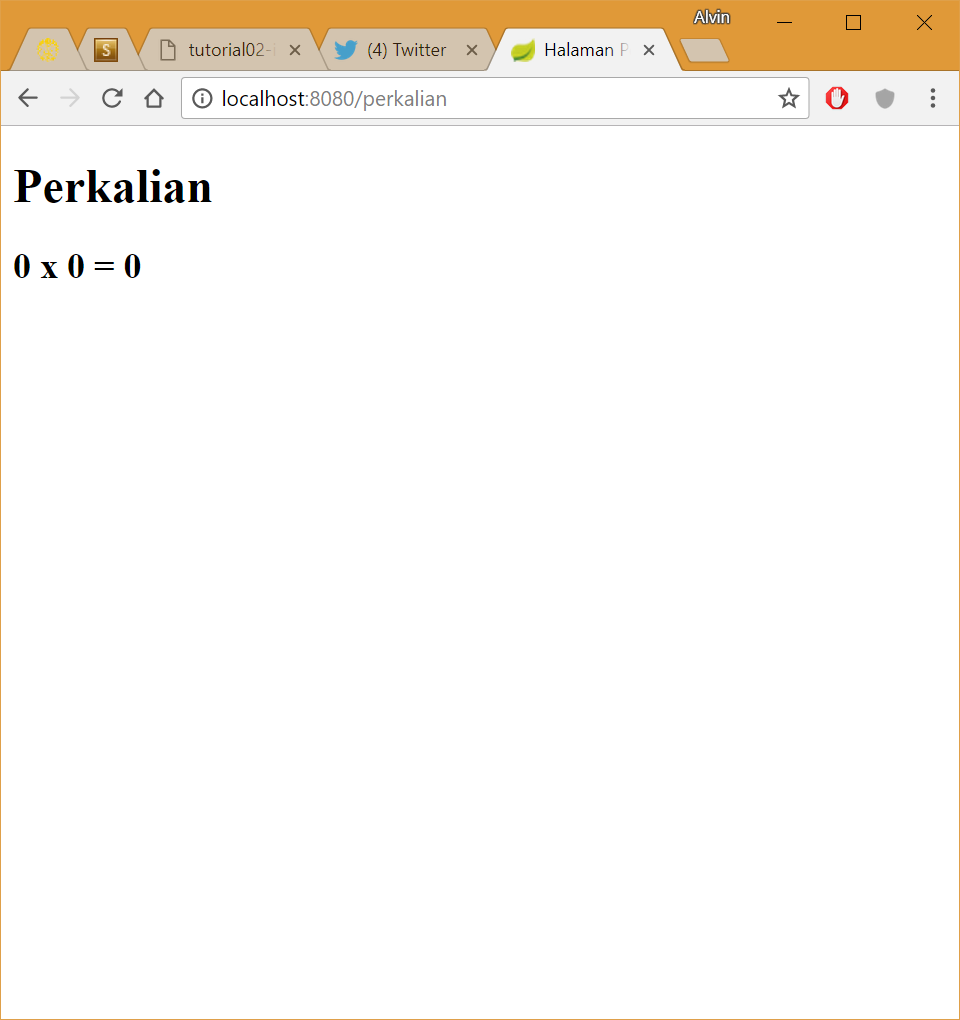
2. Halam ini menerima 2 buah parameter dari query string yaitu a dan b **Done**

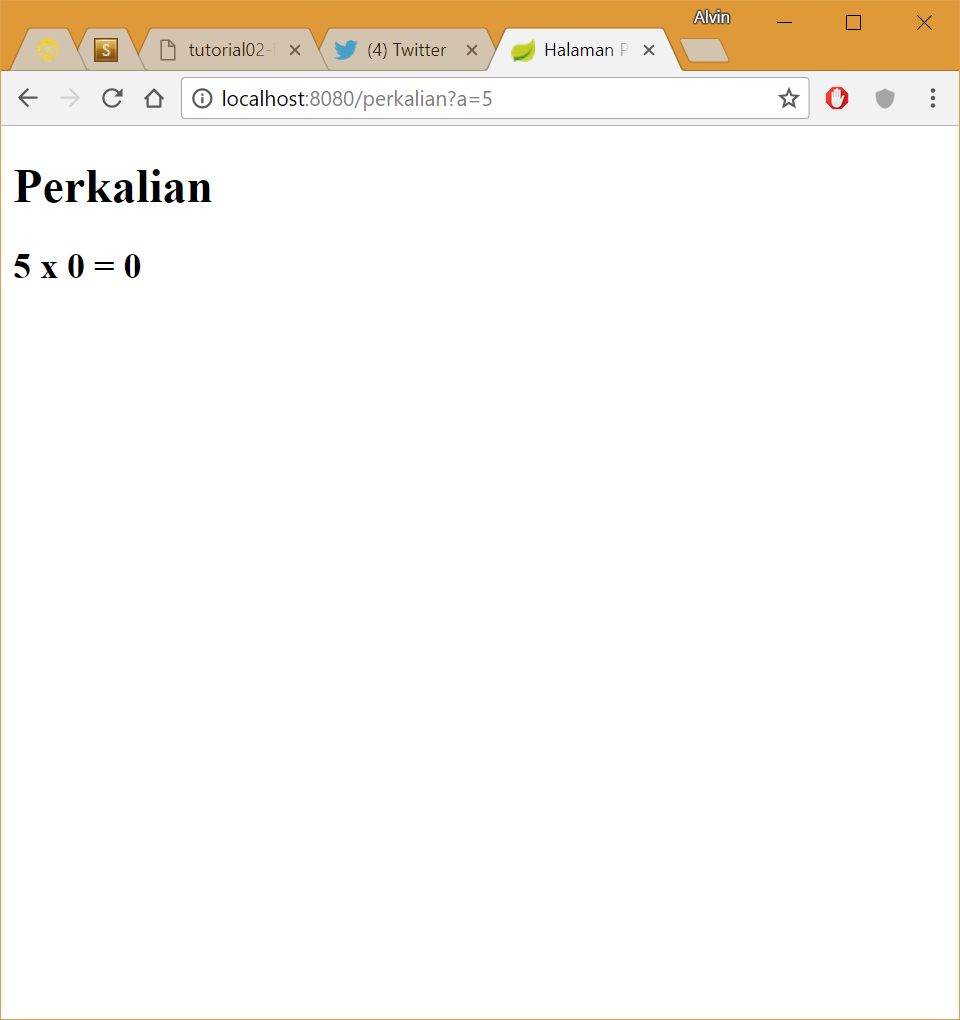


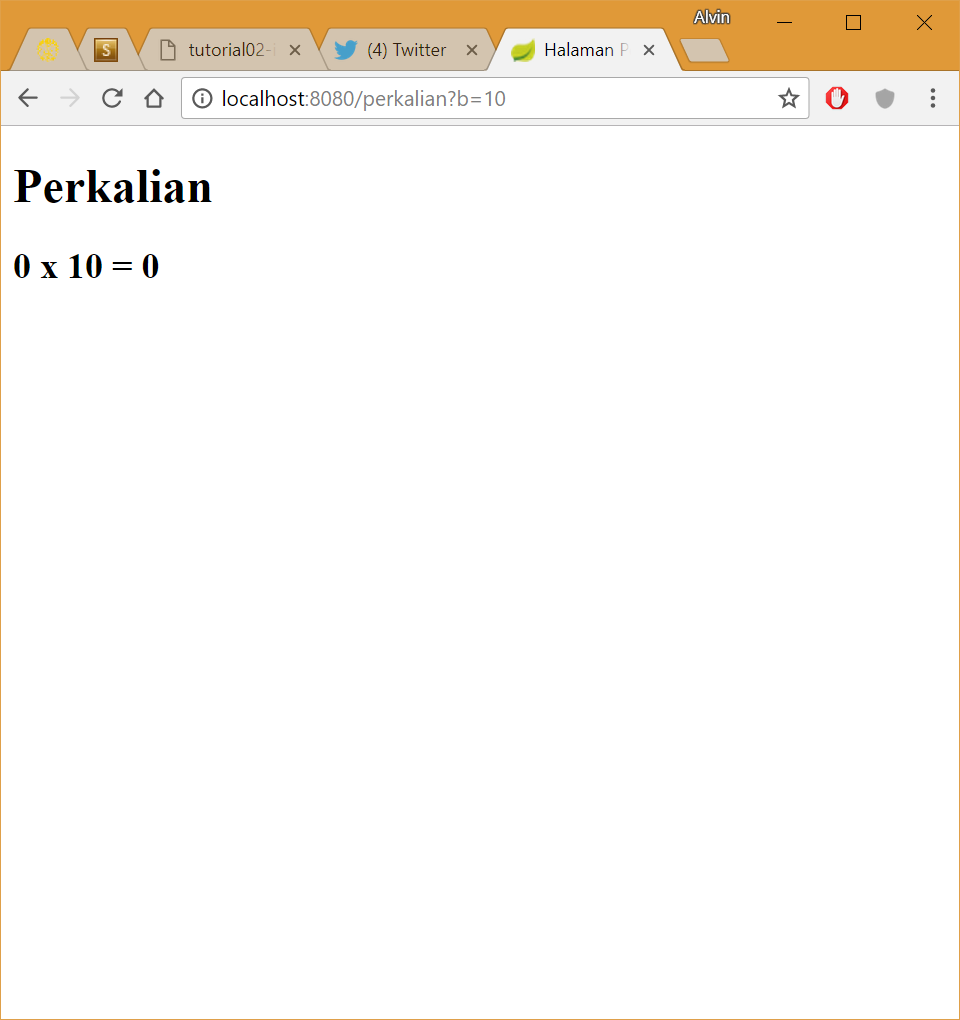
3. Pada view tampilkan hasil perkalian a dan b **Done**

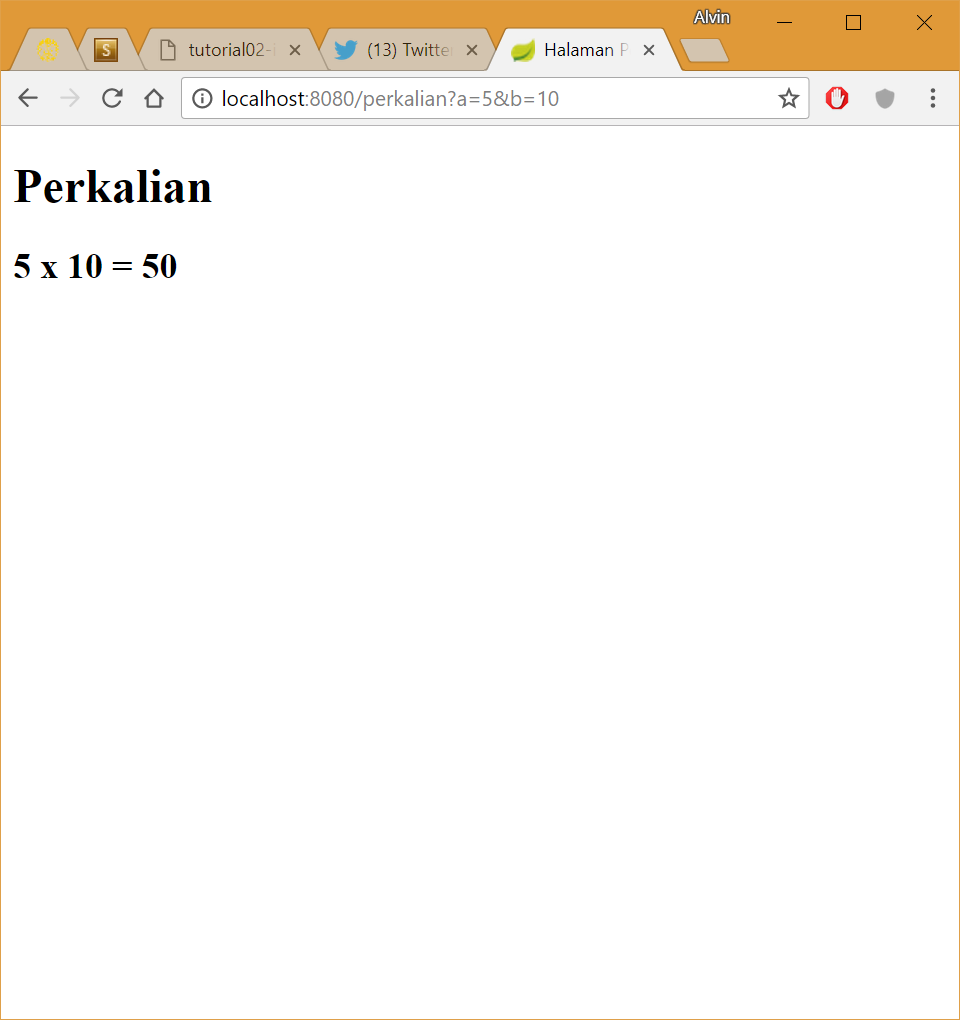


4. Jika a atau b tidak ada, nilai defaultnya 0 **Done**









Lessons learned:

* Yang saya dapat pelajari dari pengerjaan tutorial ini adalah saya dapat mengerti bagaimana cara menggunakan Spring Boot dan mengerti bagaimana Spring Boot dapat bekerja.
* Saya dapat mempelajari bagaimana cara untuk melakukan Request Parameter dan cara untuk melakukan Path Variable pada Spring Boot untuk sebuah website.