

## TUTORIAL 2

### Menggunakan Routing dan Controller dalam Project Spring Boot

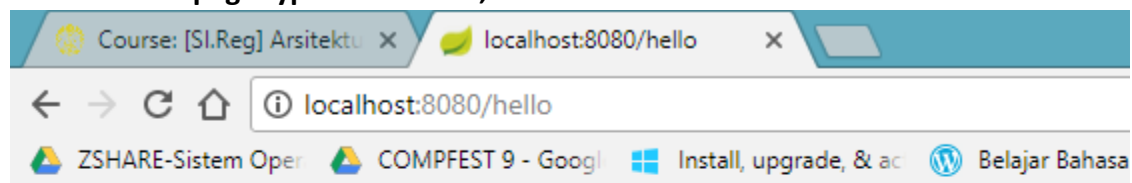
#### Latihan Project Hello World

1. Ganti baris tersebut menjadi `@RequestMapping("/hello123")`. Apakah compile error?  
**Tidak compile error.**

```
5  
6  
7 @RequestMapping("/hello123")  
8 public String index() {  
9     return "hello";  
10 }  
11 }
```

Run kembali dan buka localhost:8080/hello apa yang terjadi?

**Muncul error page type=Not Found, status=404**



### Whitelabel Error Page

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

Sat Sep 16 21:55:10 ICT 2017

There was an unexpected error (type=Not Found, status=404).

No message available

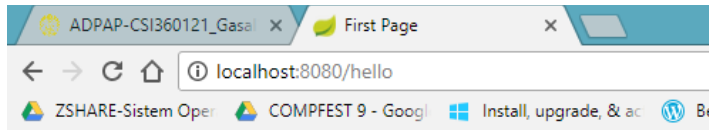
Kembalikan RequestMapping menjadi /hello

2. Ganti nama method `index()` dengan nama method `hello()`. Apakah compile error?  
**Tidak compile error.**

```
25  
26 @RequestMapping("/hello")  
27 public String hello() {  
28     return "hello";  
29 }  
30 }
```

Run kembali dan buka localhost:8080/hello apakah page sebelumnya masih muncul?

**Ya, page "Hello, world!" masih muncul.**



**Hello, world!**

Kembalikan nama method menjadi index().

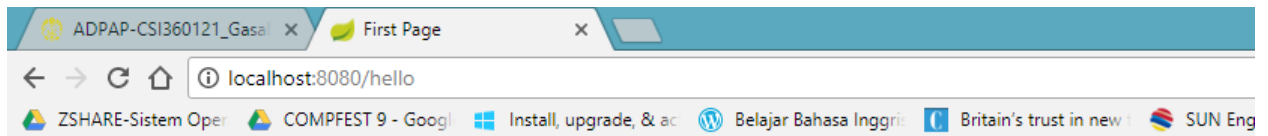
3. Ganti string return type menjadi return "hello123"; apakah compile error?

**Tidak compile error.**

```
26 @RequestMapping("/hello")
27 public String index() {
28     return "hello123";
29 }
```

Run kembali dan buka localhost:8080/hello apakah page hello sebelumnya masih muncul?

**Tidak, muncul error page pada internal server dengan status 500. Template might not exist.**



## Whitelabel Error Page

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

Fri Sep 15 11:17:22 ICT 2017

There was an unexpected error (type=Internal Server Error, status=500).

Error resolving template "hello123", template might not exist or might not be accessible by any of the configured Template Resolvers

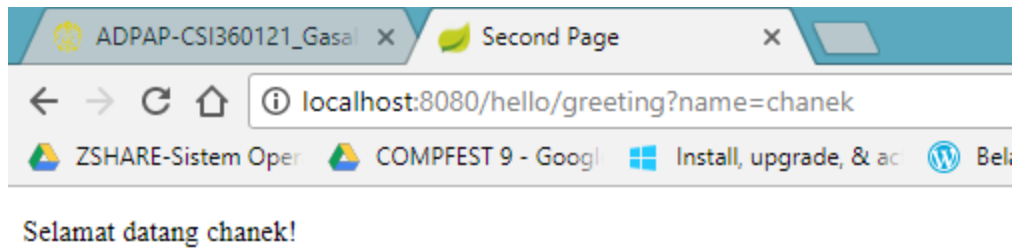
Menandakan apakah String yang di-return tersebut?

**Return pada method tersebut menandakan file html yang dituju.**

## Latihan Request Parameter

1. Ubah nilai anotasi RequestMapping dari "/greeting" menjadi "/hello/greeting" buka localhost:8080/hello/greeting?name=chanek . Apakah hasilnya?

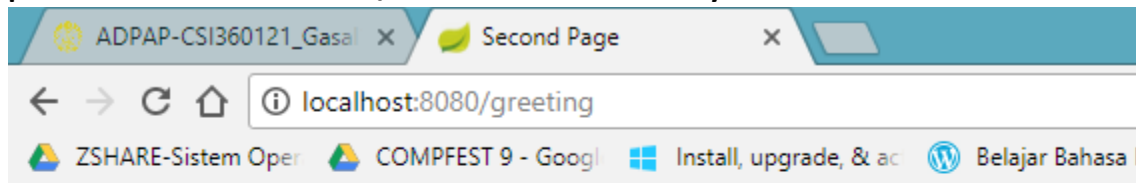
**Page greeting.html tetap muncul dengan menampilkan "Selamat datang chanek".**



Kembalikan request map.

2. Akses localhost:8080/greeting. Apakah hasilnya?

**Muncul unexpected error dengan type=Bad Request, status=400. Karena string parameter 'name' tidak ada/tidak dimasukkan nilainya.**



## Whitelabel Error Page

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

Fri Sep 15 11:24:29 ICT 2017

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

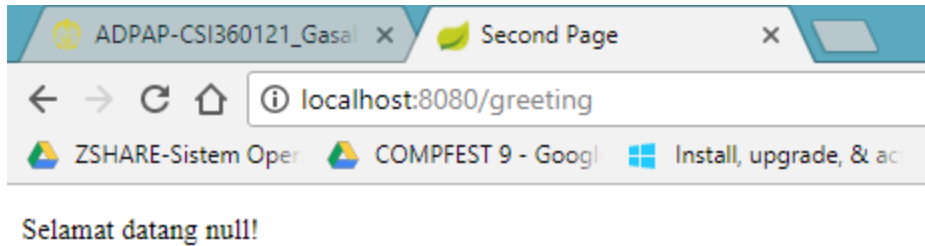
Required String parameter 'name' is not present

Ubah header method greeting menjadi seperti berikut.

```
@RequestMapping("/greeting")
public String greeting(@RequestParam(value = "name", required = false) String name,
    Model model) {
    model.addAttribute("name", name);
    return "greeting";
}
```

Stop spring boot, run kembali, buka localhost:8080/greeting. Apakah hasilnya?

**Page greeting.html ditampilkan tetapi nilai 'name' adalah null.**

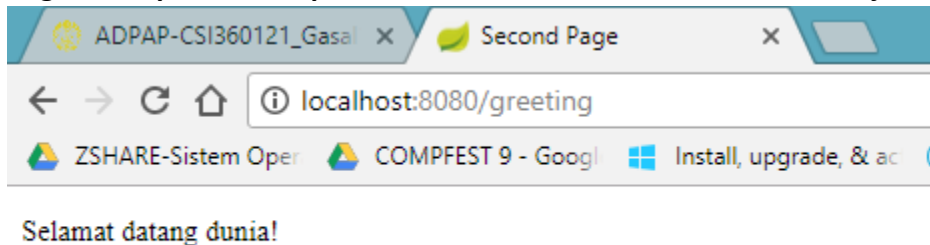


3. Ubah header method greeting menjadi seperti berikut.

```
@RequestMapping("/greeting")
public String greeting(@RequestParam(value = "name", required = false, defaultValue = "dunia") String name,
    Model model) {
    model.addAttribute("name", name);
    return "greeting";
}
```

Stop, run kembali, buka localhost:8080/greeting. Apakah hasilnya?

**Page ditampilkan, dan parameter 'name' diisi secara default menjadi 'dunia'.**



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

```
<body>
  <p th:text="'Selamat datang ' + ${name} + '!'"> Sapaan untuk user</p>
</body>
```

Mengapa tulisan "Sapaan untuk user" tidak pernah muncul?

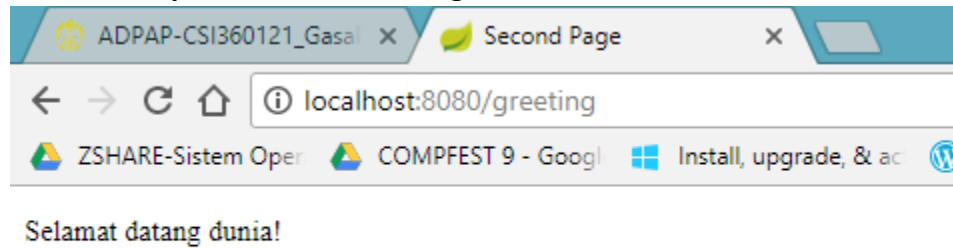
**Karena teks pada tag p tersebut tertumpuk dengan perintah-perintah di controller.**

**Teks tersebut dioverride dengan tampilan akibat command cotroller.**

## Latihan Path Variable

1. Akses localhost:8080/greeting/ apa hasilnya?

**Muncul tampilan “Selamat datang dunia”**

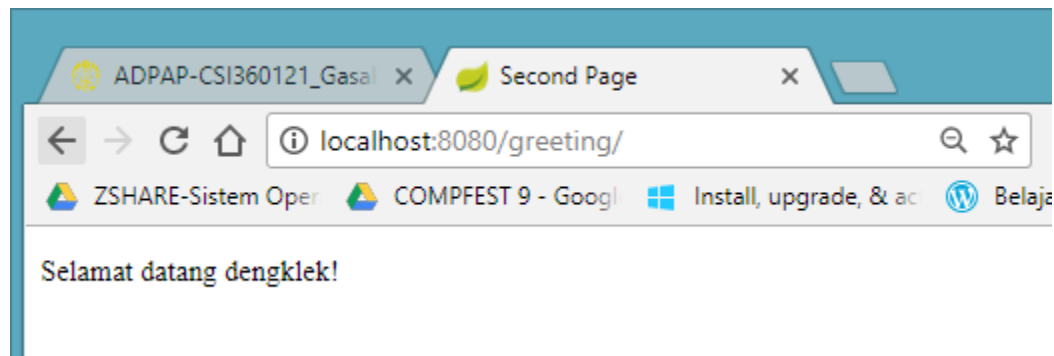


Ubah method greetingPath menjadi seperti berikut.

```
@RequestMapping(value = { "/greeting", "/greeting/{name}" })  
public String greetingPath(@PathVariable Optional<String> name, Model model) {  
    if (name.isPresent()) {  
        model.addAttribute("name", name.get());  
    } else {  
        model.addAttribute("name", "dengklek");  
    }  
    return "greeting";  
}
```

Akses localhost:8080/greeting apa hasilnya?

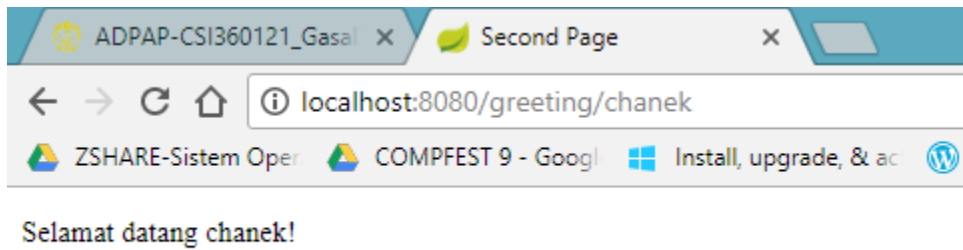
**Muncul page dengan string 'name' bernilai 'dengklek' karena user tidak memasukan nilai 'name'.**



Akses localhost:8080/greeting/chanek apa hasilnya?

**Muncul page dengan tampilan “Selamat datang chanek!”. String 'name' terisi dengan**

nilai 'chanek'.



## LATIHAN

1. Buatlah satu halaman baru dengan melakukan operasi perkalian.

```
PageController.java  greeting.html  latihan.html  hello.html
1 <!DOCTYPE html>
2 <html xmlns:th="http://www.thymeleaf.org">
3   <head>
4     <title>Second Page</title>
5   </head>
6   <body>
7     <h1>Perkalian</h1>
8     <h2 th:text="${a} + ' X ' + ${b} + ' = ' + ${result}"> Perkalian 2 Angka</h2>
9   </body>
10 </html>
```

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

```
@RequestMapping("/perkalian")
public String perkalian(@RequestParam(value = "a", required = false, defaultValue = "0") int a,
    @RequestParam(value = "b", required = false, defaultValue = "0") int b, Model model) {
```

3. Pada view tampilkan hasil perkalian a dan b.

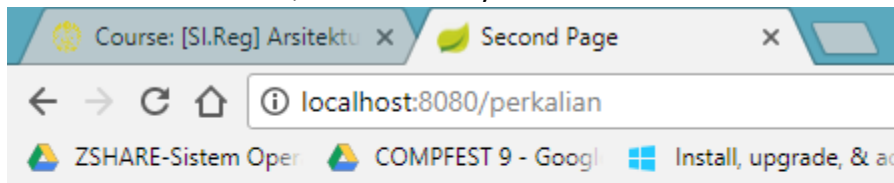
```
PageController.java  greeting.html  latihan.html  hello.html
1 <!DOCTYPE html>
2 <html xmlns:th="http://www.thymeleaf.org">
3   <head>
4     <title>Second Page</title>
5   </head>
6   <body>
7     <h1>Perkalian</h1>
8     <h2 th:text="${a} + ' X ' + ${b} + ' = ' + ${result}"> Perkalian 2 Angka</h2>
9   </body>
10 </html>

@RequestMapping("/perkalian")
public String perkalian(@RequestParam(value = "a", required = false, defaultValue = "0") int a,
    @RequestParam(value = "b", required = false, defaultValue = "0") int b, Model model) {
    model.addAttribute("a", a);
    model.addAttribute("b", b);

    int result = a * b;
    model.addAttribute("result", result);

    return "latihan";
}
```

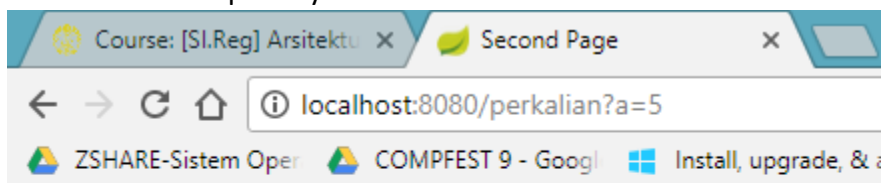
4. Jika a atau b tidak ada, nilai defaultnya 0.



## Perkalian

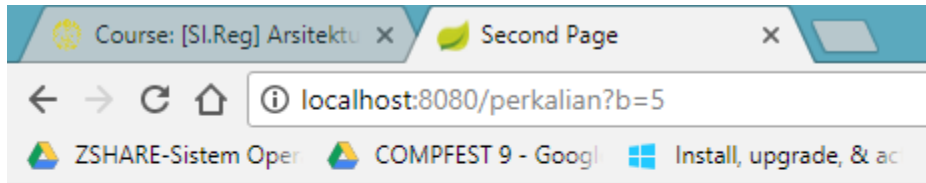
$$0 \times 0 = 0$$

5. Contoh hasil tampilannya



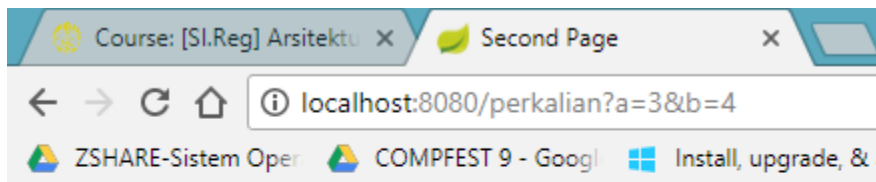
## Perkalian

$$5 \times 0 = 0$$



## Perkalian

$$0 \times 5 = 0$$



## Perkalian

$$3 \times 4 = 12$$

### Ilmu yang didapat pada tutorial ini:

Minggu ini saya belajar tentang implementasi method GET dengan request pada URL. Terdapat 2 cara request yaitu request parameter dan request path variable. Saya juga belajar tentang apa itu controller beserta perintah-perintahnya, syntax pada spring boot serta dokumentasi thymeleaf.