

Pelajaran yang saya dapat kali ini adalah bagaimana cara menggunakan Request Parameter dan Path Variable dengan Java Spring Boot. Request parameter berfungsi untuk mengambil variable yang dimasukkan, kita dapat menentukan apakah variable tersebut *required* atau tidak. Variable tersebut dapat kita olah datanya. Path variable berfungsi sama seperti request parameter, namun berupa *placeholder* dan kita tidak bisa menentukan *required* atau tidaknya.

### Latihan Project Hello World

1. Ganti baris tersebut menjadi `@RequestMapping("/hello123")`

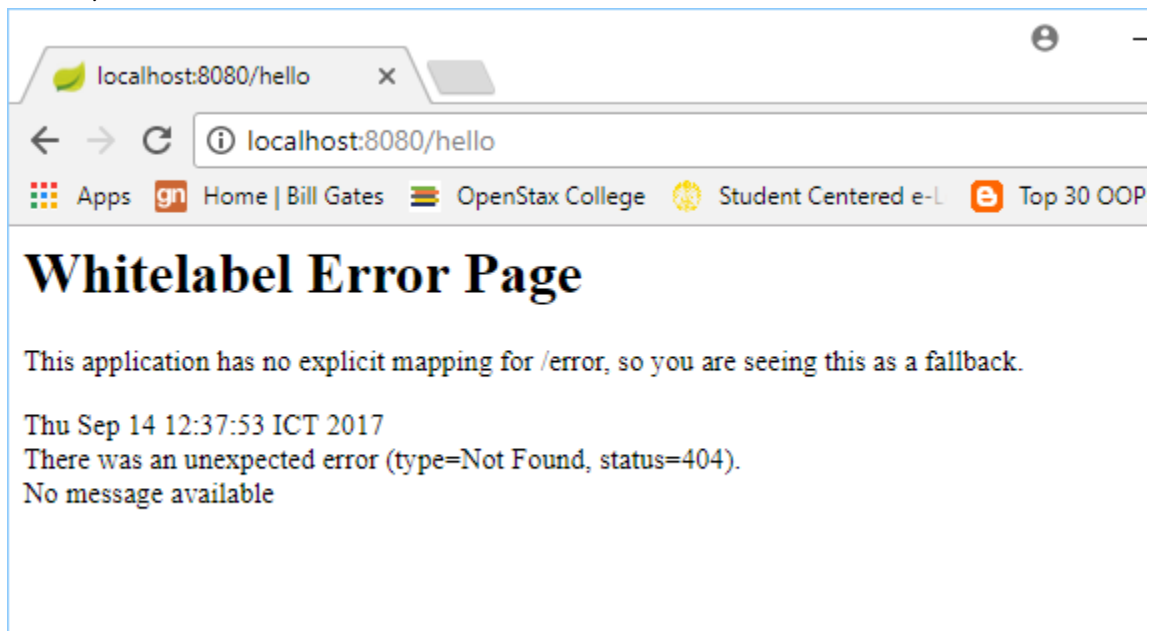
```
11  
12 @Controller  
13 public class PageController  
14 {  
15     @RequestMapping("/hello123")  
16     public String index ()  
17     {  
18         return "hello";  
19     }  
20 }
```

Pertanyaan: Apakah compile error?

Tidak.

Jika tidak, stop Spring Boot yang sedang berjalan, run kembali dan buka localhost: 8080/hello apa yang terjadi?

Terjadi error karena `@RequestMapping` berubah menjadi `/hello123`, sementara yang kita tuliskan adalah `/hello`.



Kembalikan `RequestMapping` menjadi `/hello`

May Iffah Rizki

1506757661

APAP – A

```
1 @Controller
2 public class PageController
3 {
4     @RequestMapping("/hello")
5     public String index ()
6     {
7         return "hello";
8     }
9 }
```

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

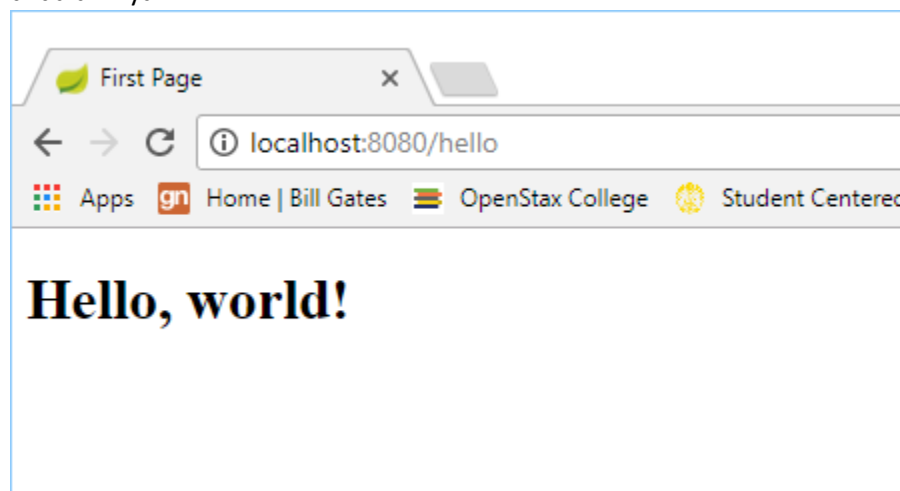
```
12 @Controller
13 public class PageController
14 {
15     @RequestMapping("/hello")
16     public String hello ()
17     {
18         return "hello";
19     }
20 }
```

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?

Masih muncul. Karena penamaan fungsi tidak berpengaruh pada @RequestMapping dan fungsi di dalamnya.



Kembalikan nama method menjadi index()

May Iffah Rizki

1506757661

APAP – A

```
1 @Controller
2 public class PageController
3 {
4     @RequestMapping("/hello")
5     public String index ()
6     {
7         return "hello";
8     }
9 }
```

3. Ganti string return type menjadi return "hello123";

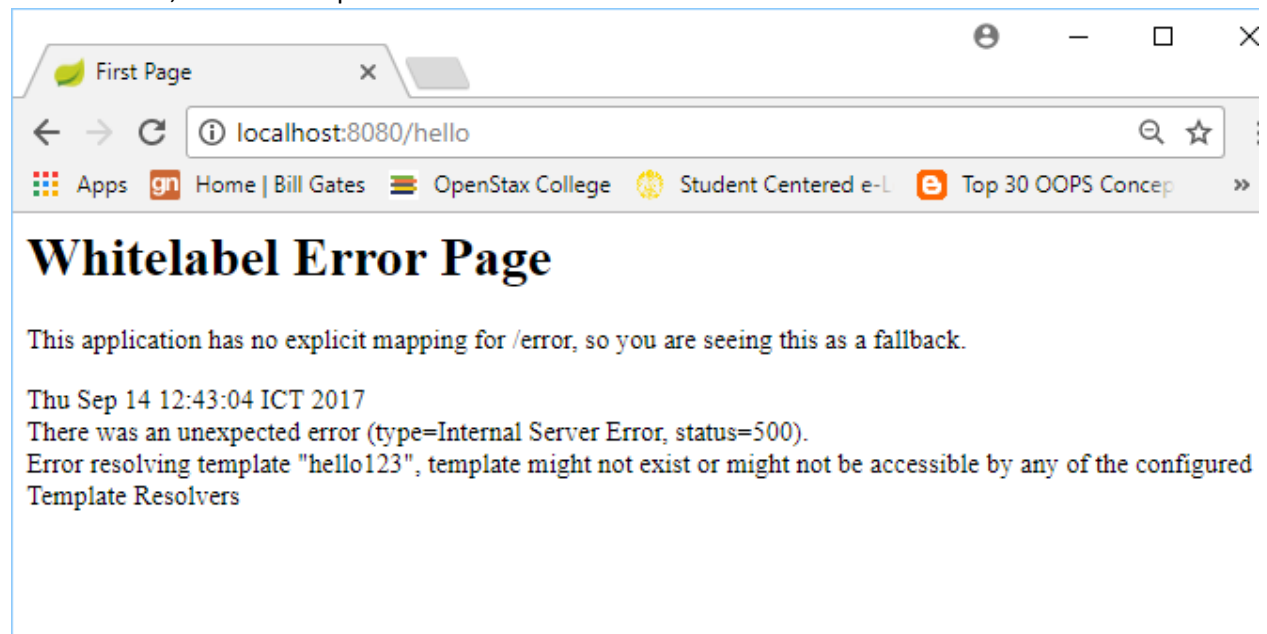
```
12 @Controller
13 public class PageController
14 {
15     @RequestMapping("/hello")
16     public String index ()
17     {
18         return "hello123";
19     }
20 }
```

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, karena terdapat error.



Kembalikan return type menjadi "hello"

May Iffah Rizki

1506757661

APAP – A

```
1  
2 @Controller  
3 public class PageController  
4 {  
5     @RequestMapping("/hello")  
6     public String index ()  
7     {  
8         return "hello";  
9     }  
10 }
```

Pertanyaan: Menandakan apakah String yang di-return tersebut?

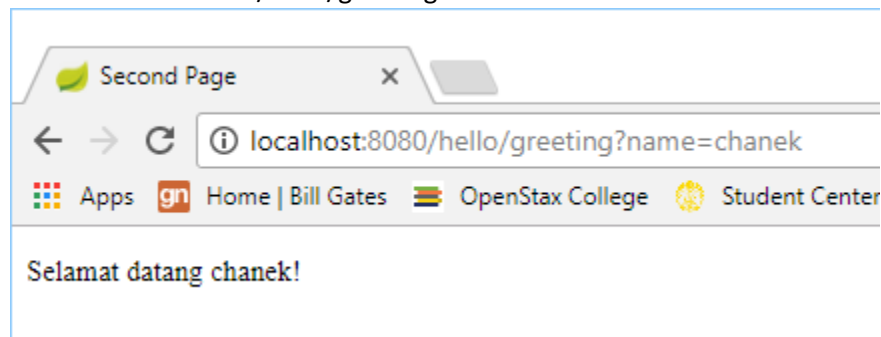
String yang direturn menandakan file hello.html yang ada di templates.

### Latihan Request Parameter

1. Ubah nilai anotasi RequestMapping dari "/greeting" menjadi "/hello/greeting"

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

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



Pertanyaan: apakah hasilnya?

Selamat datang chanek!

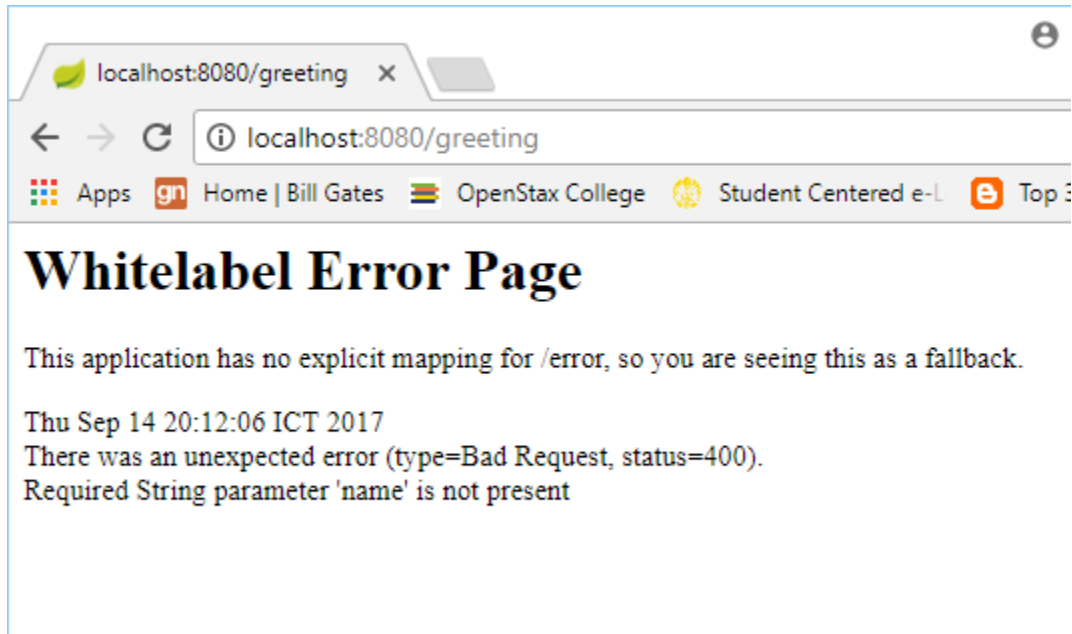
Kembalikan request map menjadi @RequestMapping("/greeting")

2. Akses localhost: 8080/greeting

May Iffah Rizki

1506757661

APAP – A

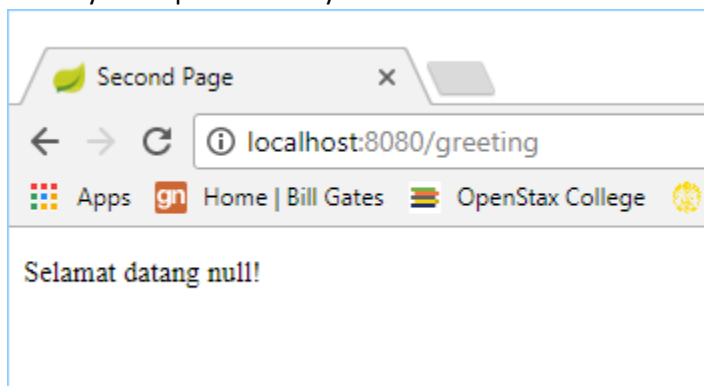


Ubah header method greeting menjadi seperti berikut public String greeting (@RequestParam(value = "name", required = false) String name, Model model)

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

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

Pertanyaan: Apakah hasilnya?



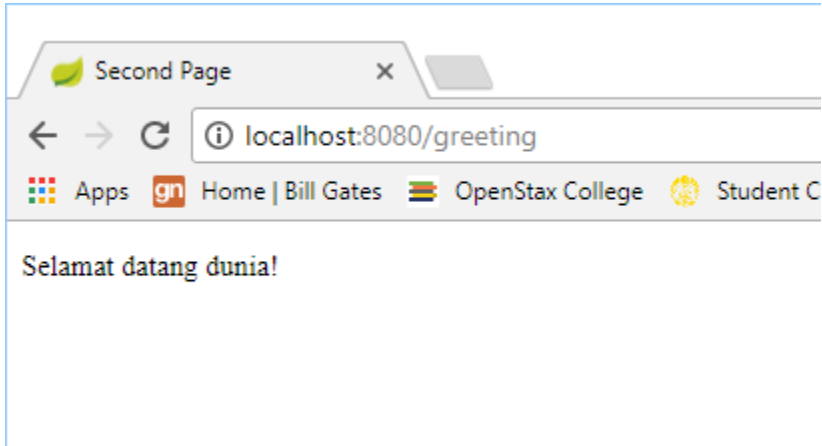
Selamat datang null!

- Ubah header method greeting menjadi seperti berikut public String greeting (@RequestParam(value = "name", required = false, defaultValue = "dunia") String name, Model model)

May Iffah Rizki  
1506757661  
APAP – A

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

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

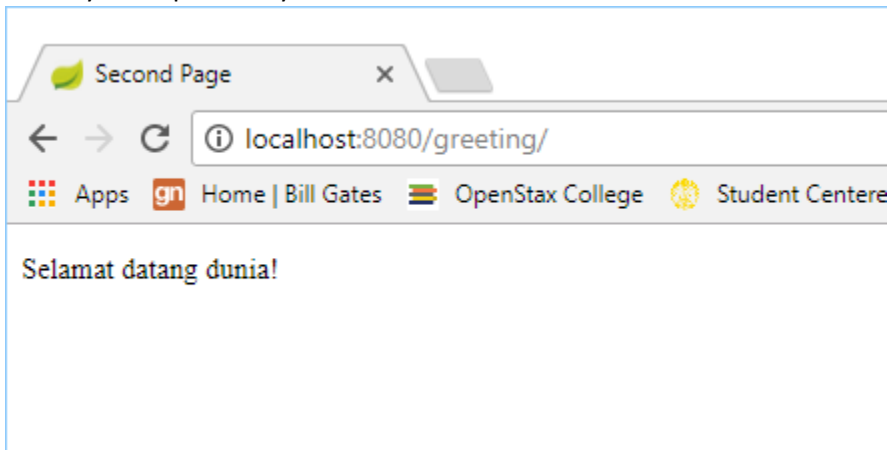


Pertanyaan: apakah hasilnya?  
Selamat datang dunia!

- Perhatikan bahwa pada berkas greeting.html, tag paragraf yang kita tambahkan adalah sebagai berikut:  
Sapaan untuk user  
Pertanyaan: Mengapa tulisan “Sapaan untuk user” tidak pernah muncul?  
Karena hal tersebut hanya menjadi penanda, bukanlah sebuah objek yang ditampilkan di website.

### Latihan Path Variable

- Akses localhost:8080/greeting/  
Pertanyaan: Apa hasilnya?



May Iffah Rizki

1506757661

APAP – A

Selamat datang dunia!

Ubah method `greetingPath` menjadi seperti berikut

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

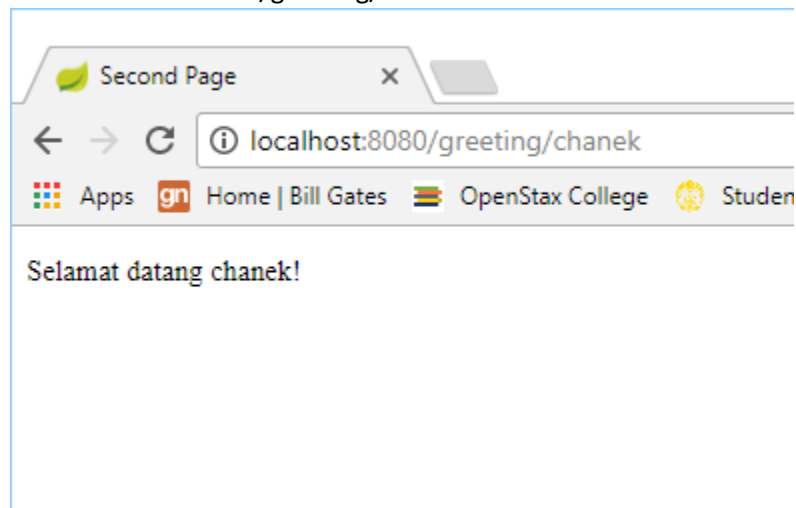
Akses `localhost:8080/greeting/`



Pertanyaan: Apa hasilnya?

Terdapat error.

Akses `localhost:8080/greeting/chanek`



Pertanyaan: Apa hasilnya?

May Iffah Rizki

1506757661

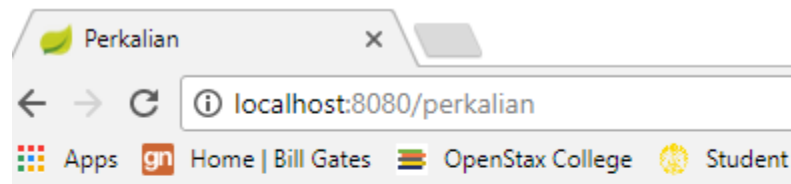
APAP – A

Selamat datang chaneke!

## Latihan

Hasil tampilan:

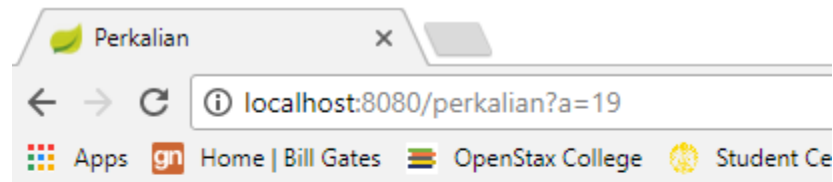
Perkalian



## Perkalian

$$0 \times 0 = 0$$

Perkalian a = 19



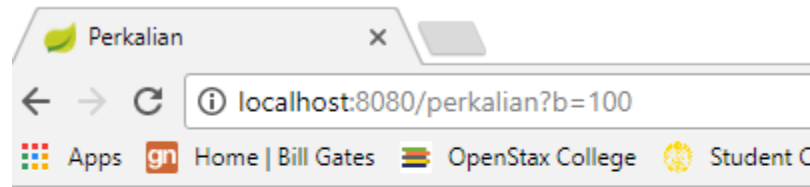
## Perkalian

$$19 \times 0 = 0$$



May Iffah Rizki  
1506757661  
APAP – A

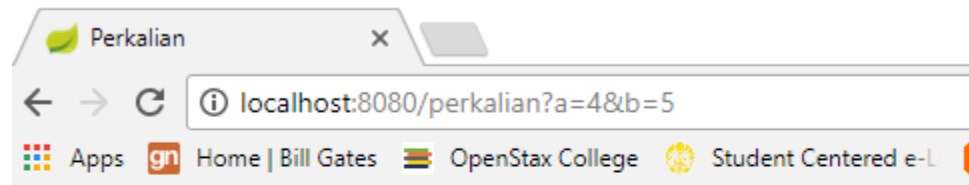
Perkalian b = 100



## Perkalian

$$0 \times 100 = 0$$

Perkalian a = 4 dan b = 5



## Perkalian

$$4 \times 5 = 20$$