

Tutorial 2 – Intro to Spring Boot

Latihan Project Hello World

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

Pertanyaan:

Apakah compile error?

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

Jawab : Program dapat compile, namun ketika browser dibuka dengan perintah tersebut, terdapat error seperti berikut



Kembalikan RequestMapping menjadi `/hello`

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

Pertanyaan:

Apakah compile error?

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

Jawab : Program dapat dicompile, dan ketika browser dibuka dengan perintah tersebut, halaman yang keluar seperti berikut (sama seperti yang keluar ketika page hello sebelumnya)



Hello, world!

Kembalikan nama method menjadi `index()`

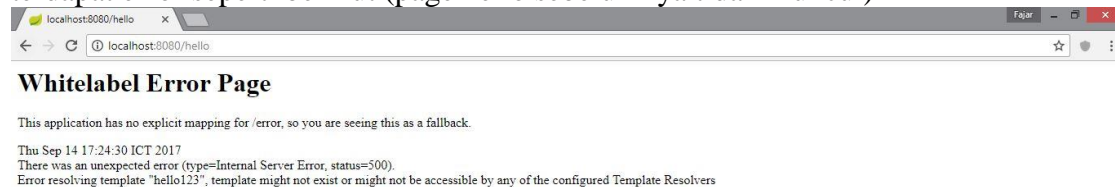
3. Ganti string return type menjadi `return "hello123";`

Pertanyaan:

Apakah compile error?

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

Jawab : Program dapat compile, namun ketika browser dibuka dengan perintah tersebut, terdapat error seperti berikut (page hello sebelumnya tidak muncul)



Kembalikan return type menjadi `"hello"`

Latihan Request Parameter

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

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

Pertanyaan: apakah hasilnya?

Jawab : Program dapat compile, dan ketika browser dibuka dengan perintah tersebut, muncul halaman seperti berikut

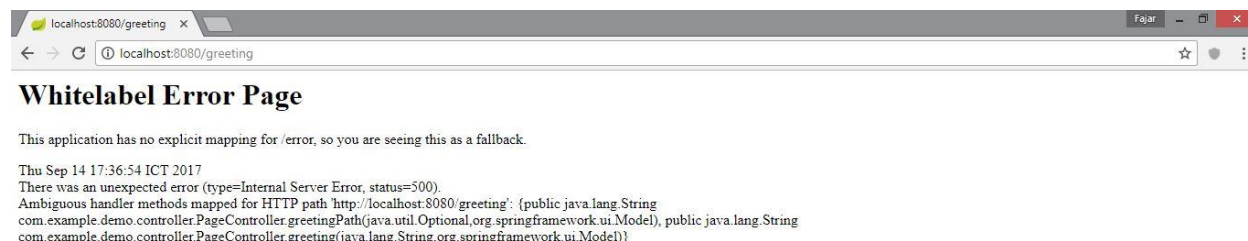


Kembalikan request map menjadi @RequestMapping("/greeting")

2. Akses **localhost: 8080/greeting**

Pertanyaan: Apakah hasilnya?

Jawab : Muncul error ketika browser dibuka dengan perintah tersebut, seperti berikut



Ubah header method greeting menjadi seperti berikut :

```
public String greeting ( @RequestParam ( value = "name" , required = false ) String
name , Model model)
```

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

Pertanyaan: Apakah hasilnya?

Jawab : Program dapat compile, setelah browser dibuka dengan perintah tersebut, muncul halaman seperti berikut (keluar "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?

Jawab : Program dapat dicompile, dan ketika browser dibuka dengan perintah tersebut, muncul halaman seperti berikut (keluar “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?

Jawab : Karena pada baris tersebut, terdapat **th:text** milik Thymeleaf yang akan mengganti seluruh teks yang ada ditempat atribut tersebut berada. Pada berkas greeting.html tersebut, **th:text** akan mengganti seluruh text yang ada pada tag p menjadi value yang ada pada atribut tersebut (sehingga tulisan “Sapaan untuk user” tidak akan muncul)

Latihan Path Variable

1. Akses localhost:8080/greeting/

Pertanyaan: Apa hasilnya?

Jawab : Program dapat dicompile, dan ketika browser dibuka dengan perintah tersebut, muncul halaman berikut (keluar “Selamat datang dunia!”)



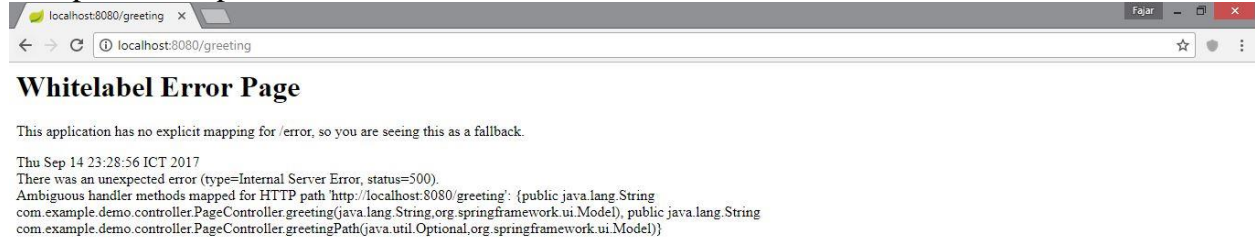
Ubah method greetingPath menjadi seperti berikut

```
@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/

Pertanyaan: Apa hasilnya?

Jawab : Program dapat compile, namun ketika browser dibuka dengan perintah tersebut, terdapat error seperti berikut



Akses **localhost:8080/greeting/chanek**

Pertanyaan: Apa hasilnya?

Jawab : Program dapat compile, dan ketika browser dibuka dengan perintah tersebut, akan muncul tampilan seperti berikut (keluar “Selamat datang chanek!”)



Latihan

1. Buatlah satu halaman baru untuk melakukan operasi perkalian.
2. Halaman ini menerima **2 buah** parameter dari query string yaitu a dan b.
3. Pada view tampilkan hasil perkalian a dan b.
4. Jika a atau b tidak ada, nilai defaultnya 0.
5. Contoh hasil tampilannya

localhost:8080/perkalian/ → perkalian yg tidak didefinisikan akan mengeluarkan nilai a dan b menjadi nilai default yaitu 0



Perkalian

0 x 0 = 0

localhost:8080/perkalian?a=8 → perkalian jika diketahui nilai a=8 dan b tidak didefinisikan (defaultnya = 0)



Perkalian

8 x 0 = 0

localhost:8080/perkalian?a=4&b=5 → perkalian 4 dgn 5



Perkalian

4 x 5 = 20

localhost:8080/perkalian?a=19&b=97 → perkalian 19 dgn 97



Perkalian

19 x 97 = 1843

(NOTE : write-up ada pada file txt, source code terlampir)