

Ringkasan

Pada tutorial ini, saya mempelajari untuk membuat aplikasi java web menggunakan framework spring, membuat *controller* dari suatu halaman web yang terdiri dari method-method. Adapaun hal lainnya yg dilakukan adalah :

- RequestMapping: yang berfungsi untuk pemanggilan URL
- RequestParameter: untuk menambahkan parameter pada request URL
- PathVariable: untuk passing data dari URL

Latihan Project Hello World

1. Ganti baris tersebut menjadi `@RequestMapping("/hello123")` Pertanyaan: Apakah compile error? **Tidak error tetapi tidak menampilkan halaman web**
Jika tidak, stop Spring Boot yang sedang berjalan, run kembali dan buka localhost:8080/hello apa yang terjadi? Kembalikan RequestMapping menjadi /hello

Whitelabel Error Page

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

Tue Feb 20 21:01:25 WIB 2018

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

No message available

2. Ganti nama method `index()` dengan nama method `hello()` Pertanyaan: Apakah compile error? **Tidak error, halaman web tetap dapat ditampilkan**
Jika tidak, Stop Spring Boot yang sedang berjalan, run kembali dan buka localhost:8080/hello apakah page hello sebelumnya masih muncul? Kembalikan nama method menjadi `index()`

Hello, World!

3. Ganti string return type menjadi `return "hello123";`
Pertanyaan: Apakah compile error? **Tidak error tetapi tidak menampilkan halaman web**
Jika tidak, Stop Spring Boot yang sedang berjalan, run kembali dan buka localhost:8080/hello apakah page hello sebelumnya masih muncul? Kembalikan return type menjadi `"hello"`
Pertanyaan: Menandakan apakah String yang di-return tersebut? Menandakan file html yang akan dijelaskan

Whitelabel Error Page

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

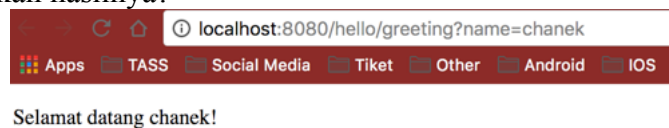
Tue Feb 20 21:26:01 WIB 2018

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

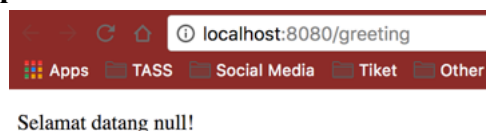
Latihan Request Parameter

1. Ubah nilai anotasi RequestMapping dari "/greeting" menjadi "/hello/greeting"
Buka **localhost:8080/hello/greeting?name=chanek**
Pertanyaan: apakah hasilnya?

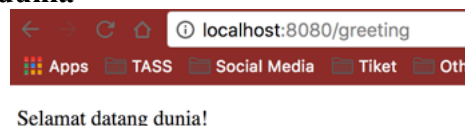


Kembalikan request map menjadi @RequestMapping("/greeting")

2. Akses localhost:8080/greeting Pertanyaan: Apakah hasilnya? 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? **Halaman web tetap berjalan dengan diperbolehkan value dari name = 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? **Walaupun nilai pada parameter name tidak dipanggil pada link, value dari name tetap tampil pada halaman web dengan defaultValue yang telah di set yaitu dunia**



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

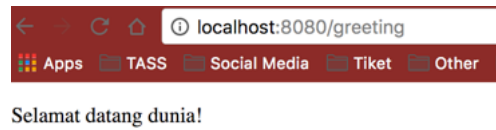
<p th:text=”Selamat datang ‘ + \${name} + ‘!”>Sapaan untuk user</p>

Pertanyaan: Mengapa tulisan “Sapaan untuk user” tidak pernah muncul?

Dikarenakan menggunakan thymeleaf yang merupakan template engine untuk spring dan menggunakan tag xml dimana tidak akan mengeksekusi yang berada diluar tags

Latihan Path Variable:

1. Akses localhost:8080/greeting/
Pertanyaan: Apa hasilnya?



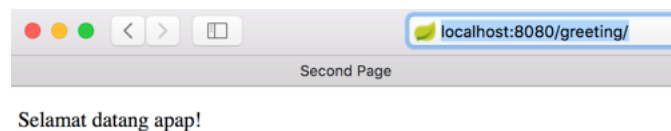
Ubah method greetingPath menjadi seperti berikut

```
@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", "apap");  
    }  
    return "greeting"; }  
import import java.util.Optional;
```

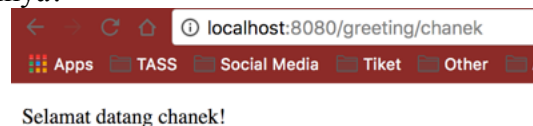
dan tambahkan

```
import import java.util.Optional;
```

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



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



Latihan

File HTML

```
<!DOCTYPE html>
<html xmlns:th="http://www.thymeleaf.org">
  <head>
    <title> Perkalian </title>
  </head>
  <body>
    <p><h3>Perkalian</h3></p>
    <p th:text="${a} + 'x' + ${b} + '=' + ${hasil}"></p>
  </body>
</html>
```

File java class

```
public class PerkalianController {

    @RequestMapping("/perkalian")
    public String perkalian (@RequestParam(value = "a", required = false) Integer a,
        @RequestParam(value = "b", required = false) Integer b, Model model) {
        int hasil;
        hasil=0;
        if (a == null && b == null) {
            a=0;
            b=0;

        }else if(a==null){
            a=0;
        }
        else if(b==null){
            b=0;
        }
        model.addAttribute("a", a);
        model.addAttribute("b", b);
        model.addAttribute("hasil", a*b);
        return "perkalian";
    }
}
```

Output



Perkalian

0x0=0



Perkalian

10x10=100

