

## Tutorial 2

### Menggunakan Routing dan Controller dalam Project Spring Boot

#### ▪ Latihan Project Hello World

1. Apakah *compile error*? Tidak terjadi *compile error*.

Jika tidak, stop Spring Boot yang sedang berjalan, run kembali dan buka **localhost: 8080/hello** apa yang terjadi? Halaman menampilkan Whitelabel Error Page karena *controller* tidak memproses halaman dengan url /hello.

---

### Whitelabel Error Page

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

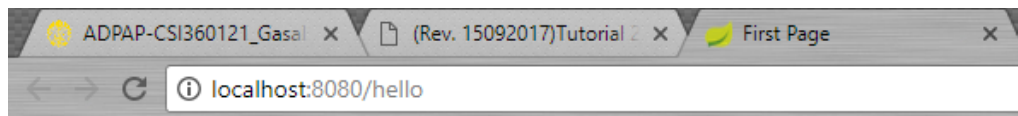
Wed Sep 13 08:26:51 ICT 2017

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

No message available

2. Apakah *compile error*? Tidak terjadi *compile error*.

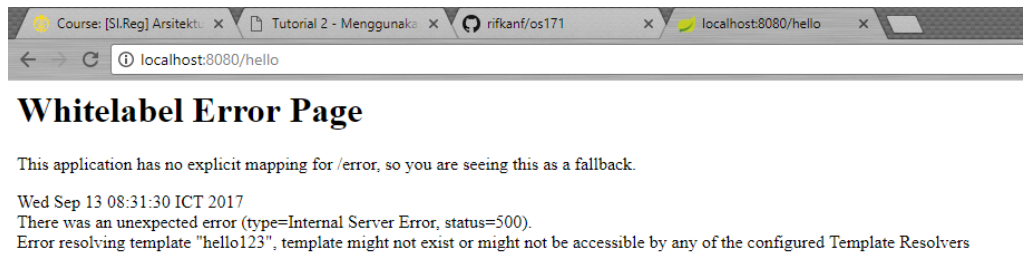
Jika tidak, Stop Spring Boot yang sedang berjalan, run kembali dan buka **localhost:8080/hello** apakah page hello sebelumnya masih muncul? Halaman kembali menampilkan *page* hello sebelum terjadi error.



## Hello, world!

3. Apakah *compile error*? Tidak terjadi *compile error*.

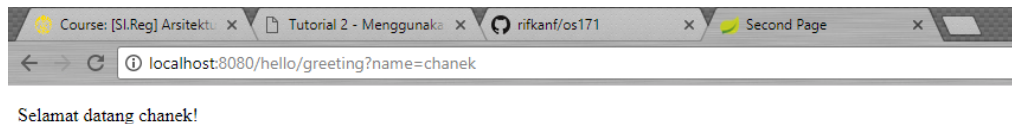
Jika tidak, Stop Spring Boot yang sedang berjalan, run kembali dan buka **localhost: 8080/hello** apakah page hello sebelumnya masih muncul? Tidak muncul. Halaman menampilkan Whitelabel Error Page karena *controller* tidak menemukan halaman hello123.



**Menandakan apakah String yang di-return tersebut?** template yang akan diakses oleh controller.

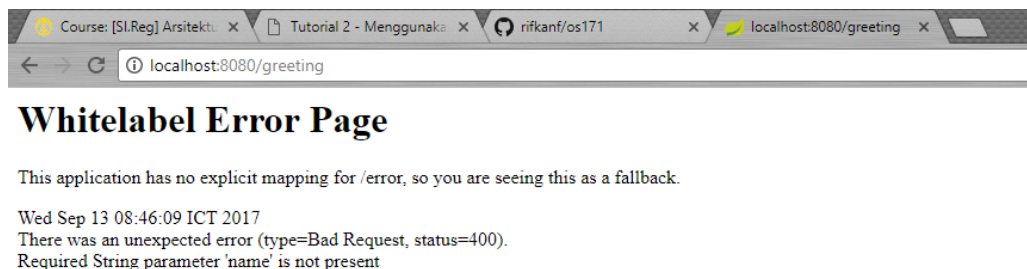
## ▪ Latihan Request Parameter

1. **Apakah hasilnya?** Halaman menampilkan tampilan sebagai berikut.



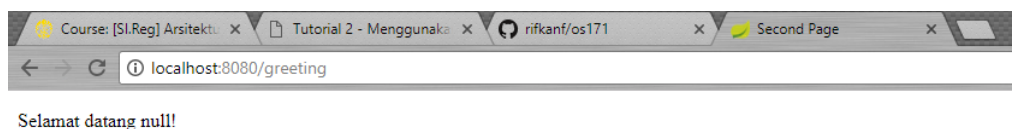
2. **Akses localhost: 8080/greeting**

**Apakah hasilnya?** Halaman menampilkan Whitelabel Error Page karena tidak ada *value* dari parameter name yang didapatkan.

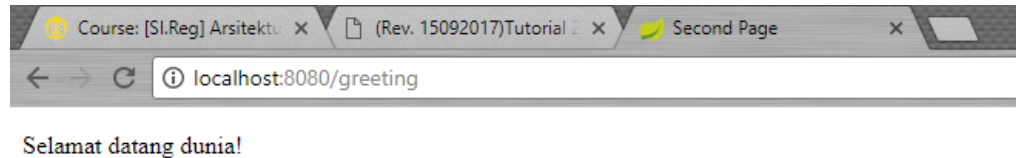


**Stop Spring Boot yang sedang berjalan, run kembali, buka localhost: 8080/greeting**

**Apakah hasilnya?** Halaman menampilkan tampilan sebagai berikut karena parameter tersebut sekarang sudah tidak *required*, maka nilai *default* secara otomatis di-assign *null* oleh sistem.



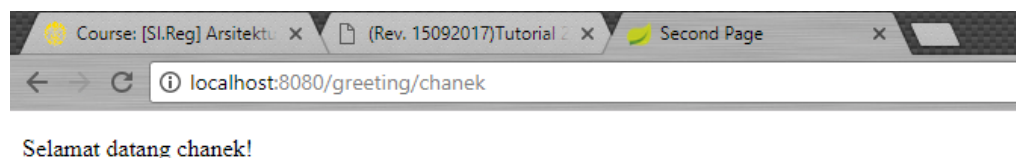
3. **Apakah hasilnya?** Halaman menampilkan tampilan sebagai berikut karena kita sudah kostumisasi nilai *default* sendiri, yaitu “dunia”, jika nilai dari parameternya tidak ada.



4. **Mengapa tulisan “Sapaan untuk user” tidak pernah muncul?** Karena terdapat string “*th:text*” yang merupakan salah satu atribut dari thymeleaf, yang apabila ditambahkan pada suatu tag berfungsi untuk me-*replace* semua tulisan yang berada di dalam tag tersebut dan digantikan dengan *value* yang berada di dalam atribut “*th:text*”.

#### ▪ Path Variable

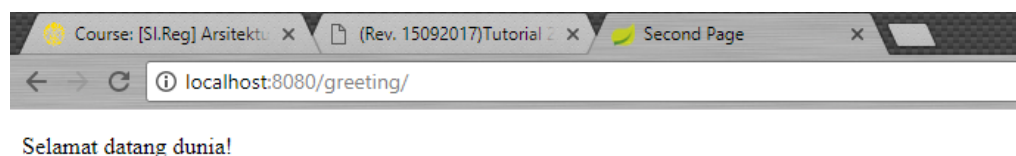
2. **Stop Spring Boot yang sedang berjalan, run kembali, dan buka localhost:8080/greeting/chanek apakah hasilnya?** Halaman menampilkan tampilan sebagai berikut.



#### ▪ Latihan Path Variable

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

**Apa hasilnya?** Halaman menampilkan tampilan sebagai berikut.



**Ubah method greetingPath menjadi seperti berikut**

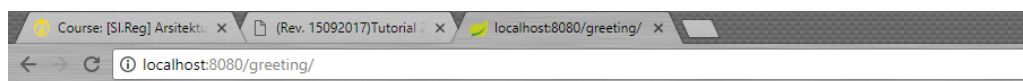
```

@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?** Halaman menampilkan Whitelabel Error Page karena terdapat dua *method* yang dapat digunakan yaitu *greeting* dan *greetingPath*.



### Whitelabel Error Page

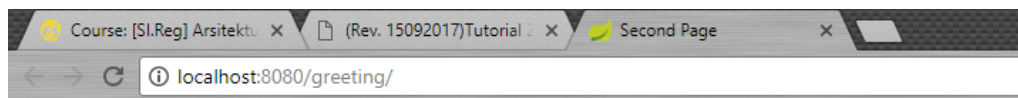
This application has no explicit mapping for /error, so you are seeing this as a fallback.

Fri Sep 15 17:43:26 ICT 2017

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

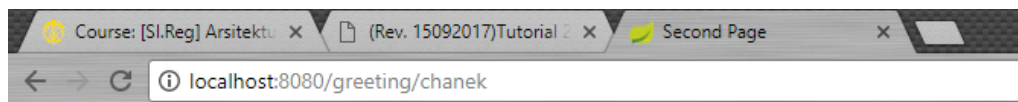
Ambiguous handler methods mapped for HTTP path 'http://localhost:8080/greeting/': {public java.lang.String com.example.demo.controller.PageController.greetingPath(java.util.Optional,org.springframework.ui.Model), public java.lang.String com.example.demo.controller.PageController.greeting(java.lang.String,org.springframework.ui.Model)}

Apabila saya menghapus *method* *greeting*, maka tampilannya akan muncul sebagai berikut.



### Akses localhost:8080/greeting/chanek

**Apa hasilnya?** Halaman akan memunculkan tampilan sebagai berikut.



## **Write-up Tutorial 2:**

Pada tutorial 2 kali ini, saya mempelajari bagaimana cara *controller* mengakses *template* yang nantinya akan ditampilkan pada suatu halaman web. Selain itu, saya juga mempelajari bagaimana melakukan *passing* suatu data dari URL pada Spring Boot, yaitu dengan menggunakan Request Parameter dan Path Variable.