

Tutorial Routing dan Controller pada Project Spring Boot

Project Hello World

Latihan Project Hello World

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

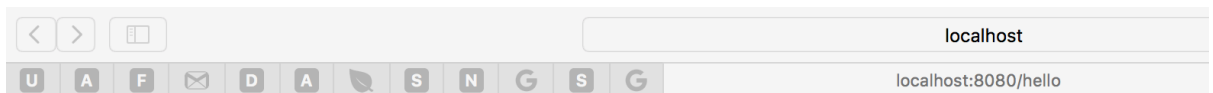
Pertanyaan:

Apakah compile error?

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

Jawab:

error compile, tetapi halaman web tidak ditemukan atau not found



Whitelabel Error Page

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

Wed Feb 21 23:11:34 WIB 2018

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()` Pertanyaan:

Apakah compile error?

Jika tidak, Stop Spring Boot yang sedang berjalan, run kembali dan buka

Jawab:

compile error, dan masih bias berjalan karena nama method tidak berpengaruh kecuali ada nama method yang sama maka compile error

RAHMAWATI PUTRIANASARI
1706106923



Hello, world!

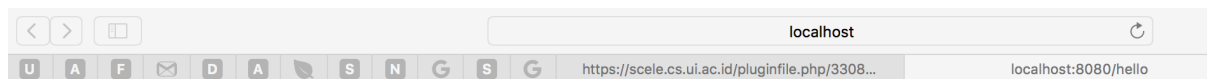
3. Ganti string return type menjadi return "hello123"; Pertanyaan:

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 :

Tidak muncul, Yang error sekarang adalah internal server error



Whitelabel Error Page

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

Wed Feb 21 23:21:36 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

Kembalikan return type menjadi "hello"

Pertanyaan:

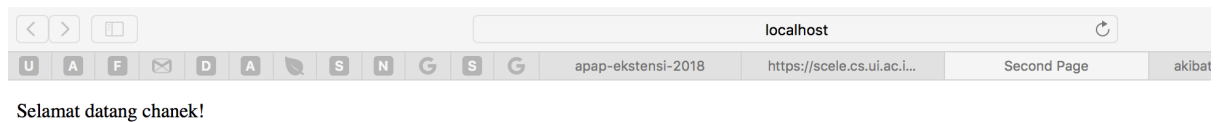
Menandakan apakah String yang di-return tersebut?

Jawab:

String tersebut berarti mereturn nama template yang digunakan dalam request parameter tersebut.

Request Parameter (Query String)

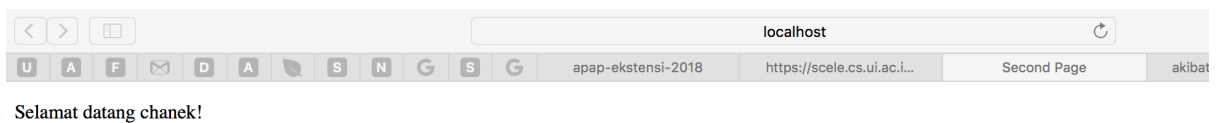
Anda baru saja mengirim parameter melalui GET request. Method greeting kemudian menangkap parameter name dari query string dengan menambahkan parameter `@RequestParam(value = "[namaparameter]")` String name sebagai parameter method greeting. Objek Model digunakan untuk passing nilai ke view. Dalam contoh ini ke view `greeting.html`.



Latihan Request Parameter

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

Jawab:

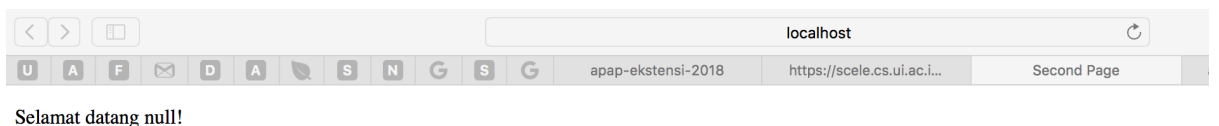


Kembalikan request map menjadi `@RequestMapping("/greeting")`

2. Akses `localhost:8080/greeting`
Pertanyaan: Apakah hasilnya? Stop Spring Boot yang sedang berjalan, run kembali, buka `localhost:8080/greeting` Pertanyaan: Apakah hasilnya?

Jawab:

Karena nilai parameternya null

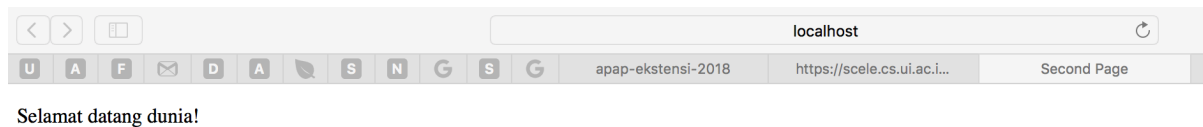


3. Ubah header method greeting menjadi seperti berikut

Stop Spring Boot yang sedang berjalan, run kembali, dan buka

`localhost:8080/greeting`. Pertanyaan: apakah hasilnya?

Jawab:



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

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

Jawab: Karena kita menggunakan Thymeleaf, maka yang digunakan adalah yang ada dalam syntax text tersebut dan mensubstitusi isi dari text yang seharusnya dikeluarkan yaitu Sapaan untuk user. Untuk memudahkan user mengisi variable dari luar.

Path Variable

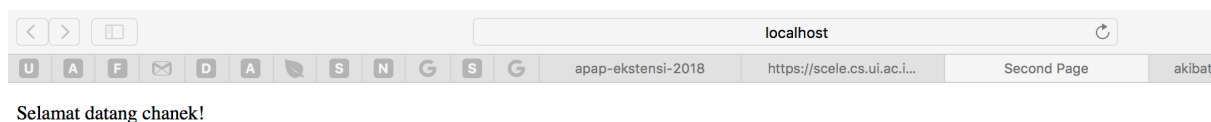
Cara lain untuk passing suatu data dari URL adalah menggunakan path variable. Contoh dari penggunaan path variable misalkan “`/user/delete/{id}`” dimana `id` merupakan path variable. Anda dapat mengirim data melalui path variable ke server pada Spring Boot.

Langkah:

1. PageController tambahkan import berikut:

Dan tambahkan method `greetingPath` pada PageController

2. Stop Spring Boot yang sedang berjalan, run kembali, dan buka `localhost:8080/greeting/chanek` apakah hasilnya?

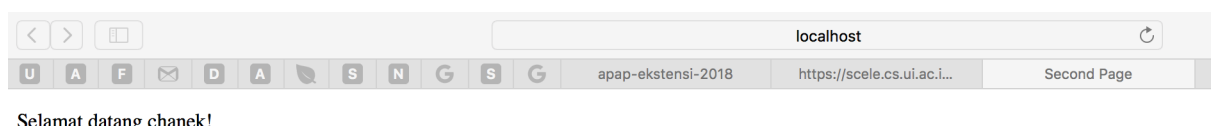


Dalam Spring Boot, `{name}` merupakan path variable yang dapat diolah oleh controller.

Latihan Path Variable:

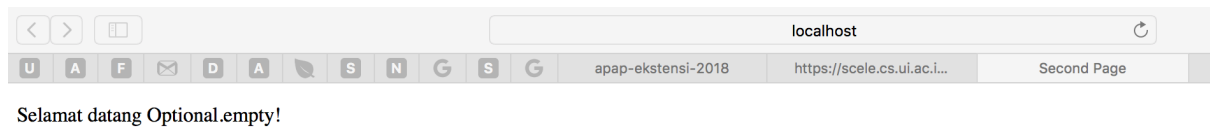
1. Akses `localhost:8080/greeting/`

Pertanyaan: Apa hasilnya?

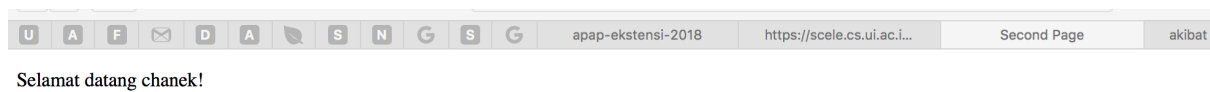


Ubah method `greetingPath` menjadi seperti berikut

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



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

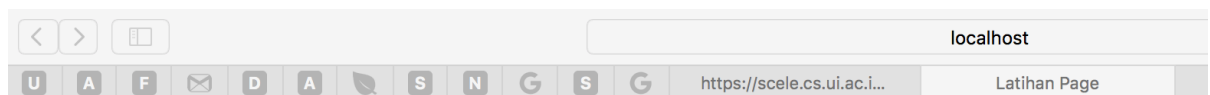


Path variable dan request parameter (query string) akan sering digunakan kedepannya dalam membuat request. Pastikan Anda memahami fungsi dan penggunaan kedua hal tersebut. Hal lain seputar penggunaan Spring Boot dapat Anda lihat pada dokumentasi Spring Boot pada tautan berikut <https://docs.spring.io/spring-boot/docs/current/reference/htmlsingle/> . Untuk dokumentasi Thymeleaf silakan ke tautan <http://www.thymeleaf.org/documentation.html> . Materi Thymeleaf lebih detail akan dibahas pada perkuliahan.

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

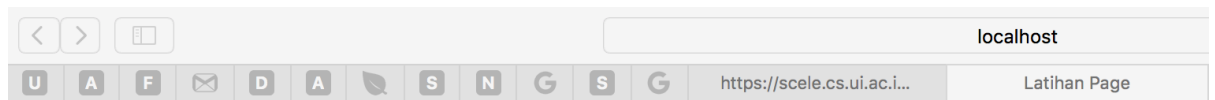


Perkalian

0 x 0 = 0

- • localhost:8080/perkalian ?a=10

RAHMAWATI PUTRIANASARI
1706106923



Perkalian

10 x 0 = 0

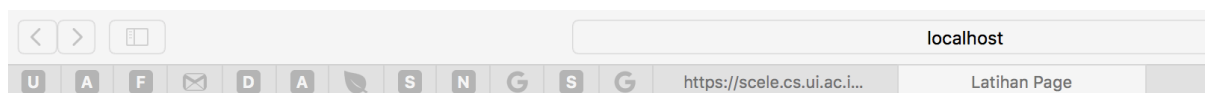
- localhost:8080/perkalian ?b=10



Perkalian

0 x 10 = 0

localhost:8080/perkalian?a=10&b=10



Perkalian

10 x 10 = 100

code:

```
package com.example.demo.controller;  
  
import java.util.Optional;  
  
import org.springframework.stereotype.Controller;
```

```
import org.springframework.ui.Model;

import org.springframework.web.bind.annotation.PathVariable;

import org.springframework.web.bind.annotation.RequestMapping;

import org.springframework.web.bind.annotation.RequestParam;


@Controller

public class PageController {


    @RequestMapping("/hello")

    public String index()

    {

        return "hello";

    }


//    @RequestMapping("/greeting")

//    public String

greeting(@RequestParam(value="name",required=false,defaultValue="dunia") String

name, Model model)

//    {

//        model.addAttribute("name",name);

//        return "greeting";

//    }


    @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","apap");

    }

    return "greeting";

}

@RequestMapping("/greeting")

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

{

    model.addAttribute("name",name);

    return "greeting";

}

@RequestMapping("/perkalian")

public String getitem(

    @RequestParam(value="a",defaultValue="0") Integer aa,

    @RequestParam(value="b",defaultValue="0") Integer bb,

    Model model)

{

    model.addAttribute("aa",aa);

    model.addAttribute("bb",bb);

}
```


RAHMAWATI PUTRIANASARI
1706106923

```
        model.addAttribute("cc",aa*bb);  
  
        return "perkalian";  
    }  
  
}
```