

Tutorial Routing dan Controller pada Project Spring Boot

Ringkasan

Anotasi `@Controller` menandakan bahwa sebuah kelas merupakan sebuah Controller yang merupakan salah satu komponen pada pattern MVC. Controller akan merespon HTTP Requests dan mengembalikan View.

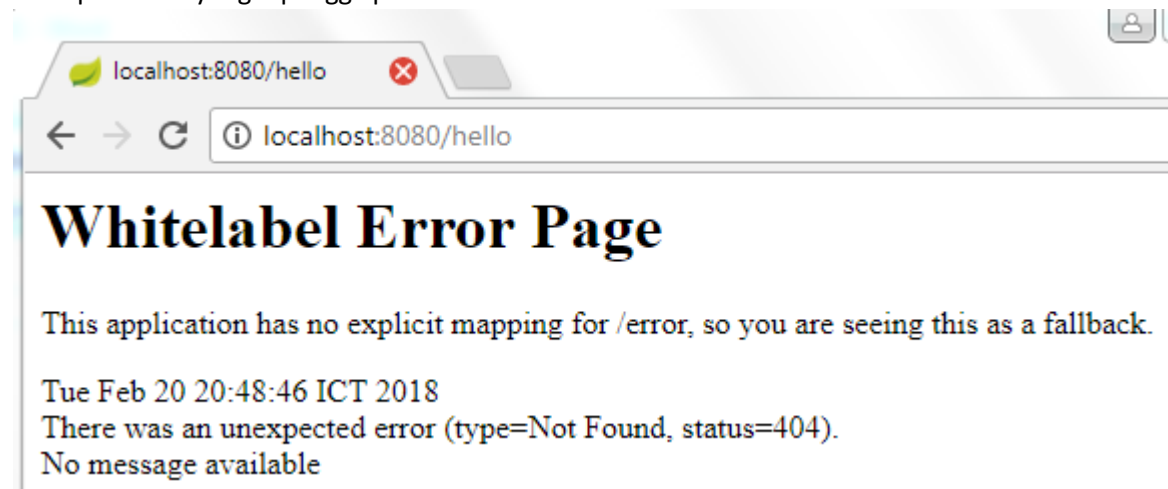
Request Parameter (Query String Spring Boot dapat mengolah query string. Parameter dikirim melalui GET request. Method kemudian menangkap parameter name dari query string dengan menambahkan parameter `@RequestParam(value = "[namaparameter]")`. Objek Model digunakan untuk passing nilai ke view.

Path Variable merupakan cara lain untuk passing suatu data dari URL. 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.

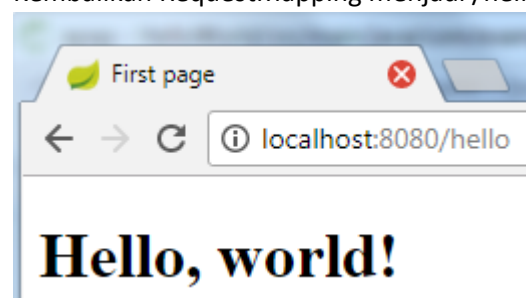
Latihan Project Hello World

1. Ganti baris tersebut menjadi `@RequestMapping("/hello123")` 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 pada anotasi request mapping yang didaftarkan adalah hello namun diganti menjadi hello123 sedangkan pada url masih tetap `/hello`. Anotasi request mapping merupakan url yang dipanggil pada browser.

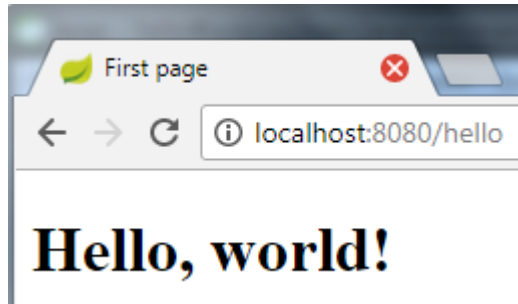


Kembalikan RequestMapping menjadi `/hello` dan hasilnya kembali seperti keadaan semula.



2. Ganti nama method `index()` dengan nama method `hello()` Pertanyaan: Apakah compile error? Tidak. Compile error.

Jika tidak, Stop Spring Boot yang sedang berjalan, run kembali dan buka `localhost:8080/hello` apakah page hello sebelumnya masih muncul? Masih muncul karena yang dipanggil adalah request mapping bukan nama method.



3. Ganti string return type menjadi `return "hello123";` Pertanyaan: Apakah compile error? Ya, compile error. Hasilnya error karena return type pada fungsi tersebut menandakan nama file html yang dituju, jika diganti menjadi `hello123` maka terjadi error karena nama file yang ada adalah `hello`.



Kembalikan return type menjadi `"hello"` sehingga hasilnya kembali lagi seperti pada keadaan semula.



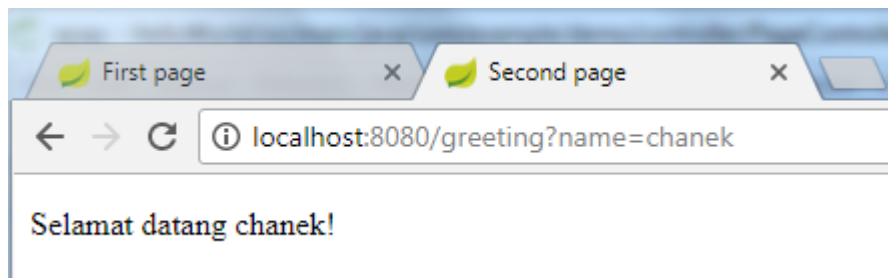
Pertanyaan: Menandakan apakah String yang di-return tersebut? Nama file html yang dipanggil.
Request Parameter (Query String)

```
package com.example.demo.controller;

import org.springframework.stereotype.Controller;
import org.springframework.ui.Model;
import org.springframework.web.bind.annotation.RequestMapping;
import org.springframework.web.bind.annotation.RequestParam;

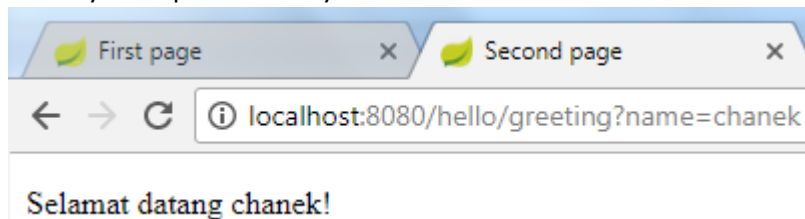
@Controller
public class PageController {
    @RequestMapping("/hello")
    public String index()
    {
        return "hello";
    }

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



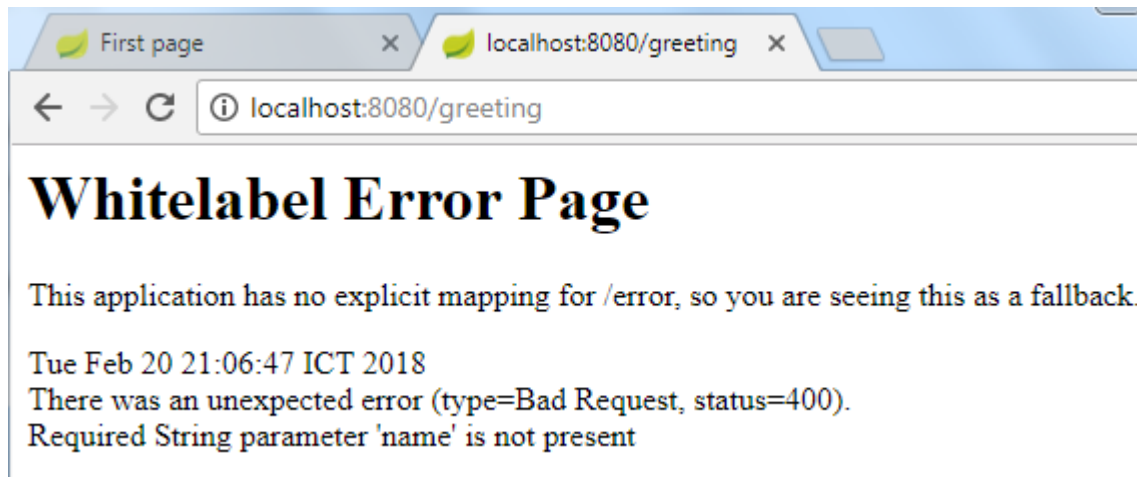
Latihan Request Parameter

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") maka hasilnya sama.

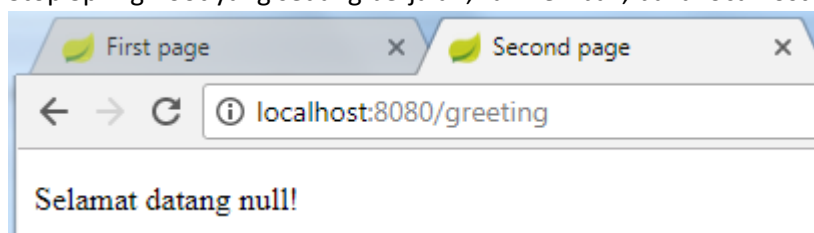
2. Akses localhost:8080/greeting
Pertanyaan: Apakah hasilnya? Error seperti pada gambar di bawah ini karena fungsi tersebut membutuhkan parameter namun pada url tidak diberikan nilai parameter sehingga error.



Ubah header method greeting menjadi seperti berikut

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

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



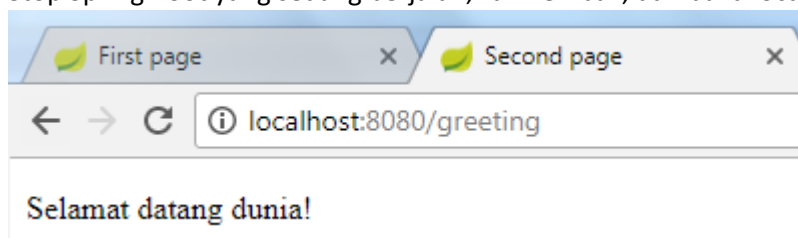
Pertanyaan:

Apakah hasilnya? Tidak error karena nilai parameter tidak required (required = false) tetapi namanya menjadi null karena tidak diberikan isi parameter dan tidak diberikan nilai default value.

- Ubah header method greeting menjadi seperti berikut

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

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



Pertanyaan: apakah hasilnya? Hasilnya tidak error karena meskipun nilai parameter tidak diberikan, sudah ada nilai default value yang akan ditampilkan jika nilai parameter tidak diberikan.

- 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? Penggunaan dependency thymeleaf tidak bisa menggunakan html murni sehingga pada tag html ada tambahan misalnya tag th.

Path Variable

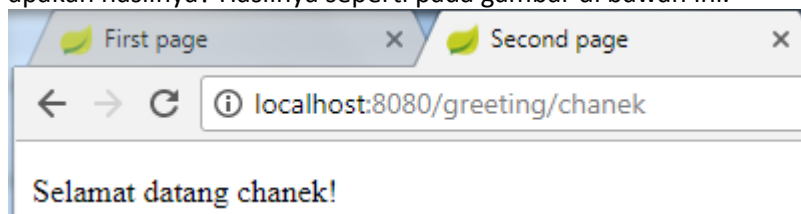
1. PageController tambahkan import berikut

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

Dan tambahkan method greetingPath pada PageController

```
@RequestMapping("/greeting/{name}")
public String greetingPath(@PathVariable String name, Model model) {
    model.addAttribute("name", name);
    return "greeting";
}
```

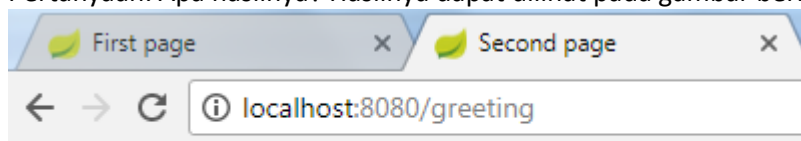
2. Stop Spring Boot yang sedang berjalan, run kembali, dan buka localhost:8080/greeting/chanek apakah hasilnya? Hasilnya seperti pada gambar di bawah ini.



Latihan Path Variable

1. Akses localhost:8080/greeting/

Pertanyaan: Apa hasilnya? Hasilnya dapat dilihat pada gambar berikut ini.



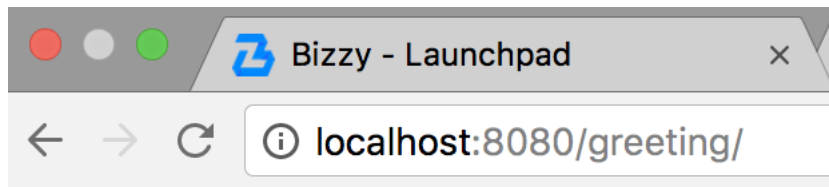
Selamat datang dunia!

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", "apap");
    }
    return "greeting";
}
```

Akses localhost:8080/greeting/

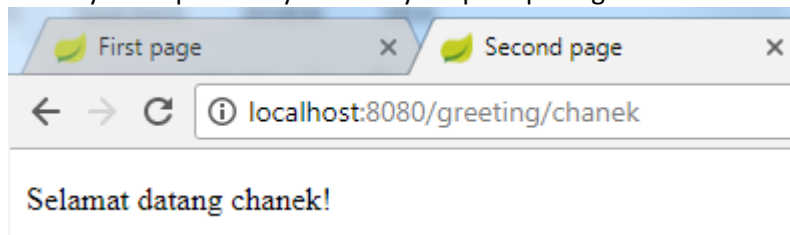
Pertanyaan: Apa hasilnya? Hasilnya seperti pada gambar di bawah ini.



Selamat datang apap!

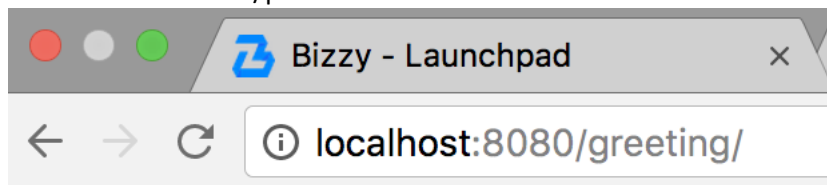
Akses localhost:8080/greeting/chanek

Pertanyaan: Apa hasilnya? Hasilnya seperti pada gambar di bawah ini.



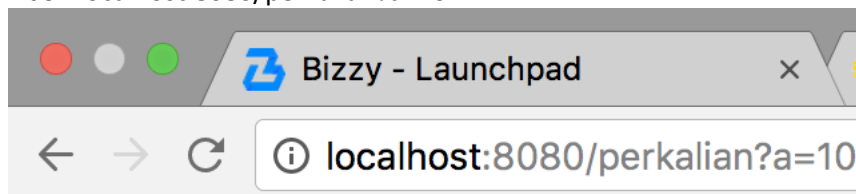
Latihan

1. Hasil localhost:8080/perkalian



Selamat datang apap!

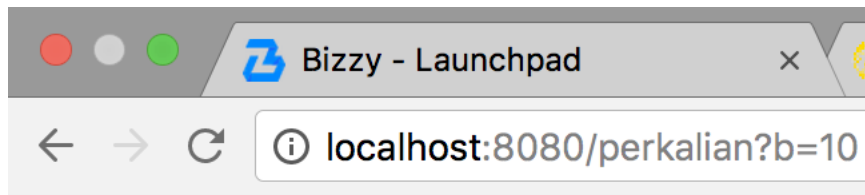
2. Hasil localhost:8080/perkalian?a=10



Perkalian

$$10 \times 0 = 0$$

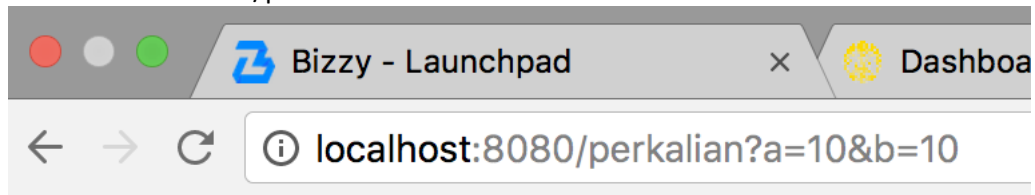
3. Hasil localhost:8080/perkalian?b=10



Perkalian

$$0 \times 10 = 0$$

4. Hasil localhost:8080/perkalian?a=10&b=10



Perkalian

$$10 \times 10 = 100$$