

### A. Ringkasan Materi

Pada tutorial ini digunakan pattern MVC. Untuk tutorial 2 hanya digunakan Controller dan View dengan menggunakan file html. Pada file controller digunakan anotasi @controller untuk mendeklarasikan bahwa file tersebut merupakan controller.

Pada controller terdapat method-method yang akan dijalankan dimana return dari method-method tersebut adalah nama file view yang merupakan file html.

Pada setiap method terdapat anotasi @RequestMapping("/http\_url") yang digunakan untuk mapping method sehingga untuk pemanggilan method tersebut cukup dengan menggunakan url "/http\_url" yang dideklarasikan pada anotasi @RequestMapping.

Anotasi @RequestParam digunakan untuk menangkap parameter yang dikirimkan melalui GET Request pada URL. Pada anotasi @RequestParam terdapat properties seperti value yaitu nama dari parameter yang dimasukkan, required digunakan untuk menentukan apakah parameter adalah syarat untuk dapat mengakses method tersebut atau tidak, defaultValue untuk menyimpan nilai/value jika parameter tidak dimasukkan ketika required false.

Anotasi @PathVariable digunakan juga untuk menangkap parameter. Hanya saja parameter yang dikirimkan bukan melalui GET request melainkan dari path url yang diakses pada URL. Contoh URL adalah sebagai berikut /greeting/parameter. Properties @PathVariable adalah name. Parameter dapat dibuat optional dan dapat digunakan method isPresent untuk mengetahui apakah parameter dimasukkan dalam URL yang diakses. Terdapat juga method get untuk memanggil parameter yang diinputkan.

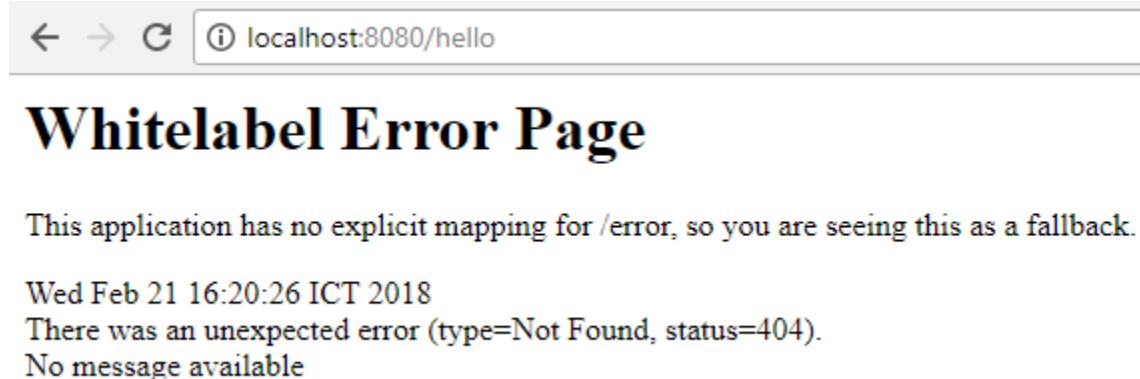
### B. Jawaban Tutorial

Latihan Project Hello World

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

Pertanyaan : Apakah compile error?

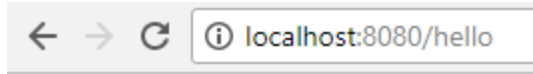
**Jawab** : Tidak tapi tampilan yang dihasilkan adalah sebagai berikut.



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

Pertanyaan : Apakah compile error?

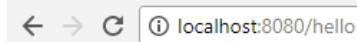
**Jawab** : Tidak



# Hello, world!

3. Ganti string return type menjadi return "hello123";  
Pertanyaan : Apakah compile error?

**Jawab :** Tidak tapi tampilan yang dihasilkan adalah sebagai berikut.



## Whitelabel Error Page

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

Wed Feb 21 16:22:29 ICT 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

4. Pertanyaan : Menandakan apakah String yang di-return tersebut?

**Jawab :** String menandakan nama file html yang akan ditampilkan sehingga string yang di-return harus sama dengan nama file html yang ditampilkan.

### Latihan Request Parameter

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

**Jawab :**



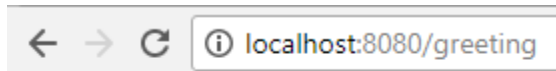
Selamat datang chanek!

2. Akses localhost:8080/greeting  
Ubah header method greeting menjadi seperti berikut

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

Pertanyaan: Apakah hasilnya?

**Jawab :**



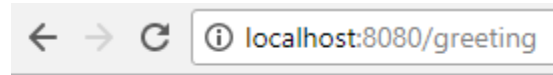
Selamat datang null!

3. Akses localhost:8080/greeting  
Ubah header method greeting menjadi seperti berikut

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

Pertanyaan: Apakah hasilnya?

**Jawab :**



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?

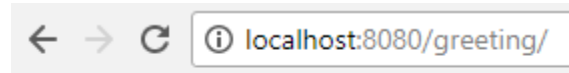
**Jawab :** Karena tulisan tersebut otomatis di-replace dengan text yang ada pada tag p.

Latihan Path Variable:

1. Akses localhost:8080/greeting/

Pertanyaan: Apa hasilnya?

**Jawab :**



Selamat datang dunia!

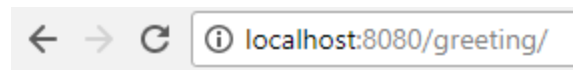
2. Ubah method greetingPath menjadi seperti berikut

```
@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";
}
```

3. Akses localhost:8080/greeting/

Pertanyaan: Apa hasilnya?

**Jawab :**

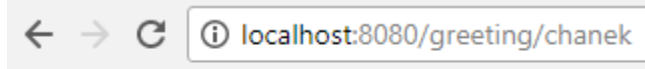


Selamat datang apap!

4. Akses localhost:8080/greeting/chaneK

Pertanyaan: Apa hasilnya?

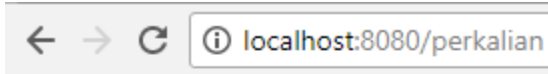
Jawab :



Selamat datang chanek!

## Latihan

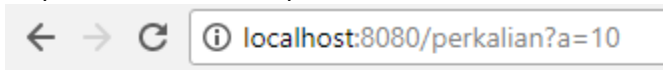
1. <http://localhost:8080/perkalian>



## Perkalian

0 x 0 = 0

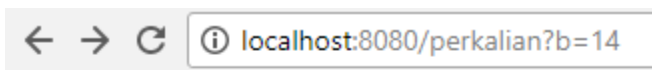
2. <http://localhost:8080/perkalian?a=10>



## Perkalian

10 x 0 = 0

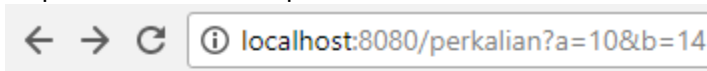
3. <http://localhost:8080/perkalian?b=14>



## Perkalian

0 x 14 = 0

4. <http://localhost:8080/perkalian?a=10&b=14>



## Perkalian

10 x 14 = 140