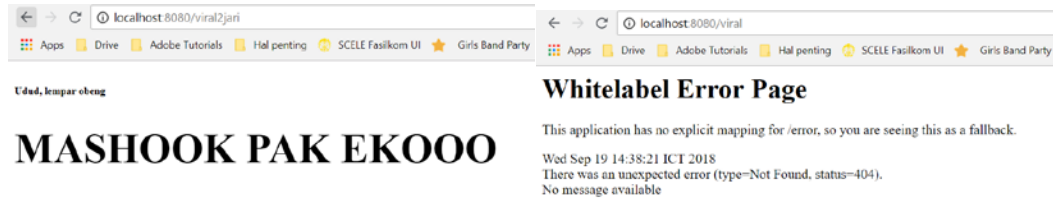


## A. Latihan Project Viral

1. Ganti baris `@RequestMapping("/viral")` menjadi `@RequestMapping("/viral2jari")`. Kemudian run kembali.



- a. Apakah terjadi compile error?

**Jawaban : Tidak.**

- b. Coba akses localhost:8080/viral, berikan alasan dan jelaskan apa yang terjadi.

**Jawaban : Terjadi error karena ketika ada request untuk membuka halaman viral, tidak ada halaman yang dirujuk oleh /viral.**

- c. Apakah Kegunaan `@RequestMapping("/viral")`?

**Jawaban : Memetakan request halaman viral ke method yang dimaksud oleh request sehingga dapat memunculkan halaman viral yang diminta oleh user.**

Kembalikan menjadi `@RequestMapping("/viral")`

2. Ganti nama method `index()` menjadi nama method `viral()`



- a. Apakah terjadi compile error?

**Jawaban : Tidak.**

- b. Coba akses localhost:8080/viral, berikan alasan dan jelaskan apa yang terjadi.

Kembalikan menjadi `index()`.

**Jawaban : Halaman dapat dibuka dengan baik. Hal ini bukan dipengaruhi oleh nama dari methodnya, melainkan method tersebut telah ditunjuk sebagai method untuk melakukan request mapping (routing menuju halaman yang dimaksud)**

3. Ganti string return type menjadi menjadi return `"viral2jari"`



- a. Apakah terjadi compile error?

**Jawaban : Tidak**

b. Coba akses localhost:8080/viral, berikan alasan dan jelaskan apa yang terjadi.

**Jawaban : Terjadi error karena ketika method index dijalankan, method tidak menemukan halaman bernama viral2jari.**

c. Apakah kegunaan return type pada method controller tersebut?

**Jawaban : Sebagai nama file .html yang akan dirender oleh program.**

4. Kembalikan return type menjadi "viral"

## B. Latihan Request Parameter

Setiap ada perubahan pada kode, selalu usahakan run ulang

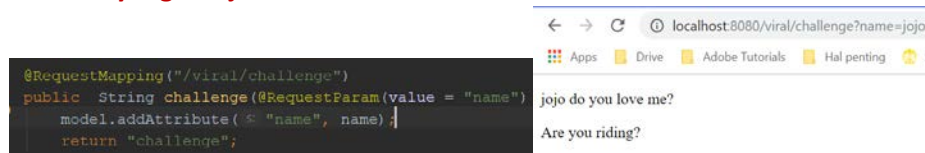
1. Mengapa tulisan "In My feeling" pada thymeleaf challenge tidak pernah muncul Di browser?

**Jawaban : Karena teks tidak di-override oleh th:text.**

2. Ubah nilai anotasi Request Mapping "/"challenge" menjadi "/viral/challenge".

Deskripsikan hasil pada browser dan kembalikan ke bentuk sebelumnya!

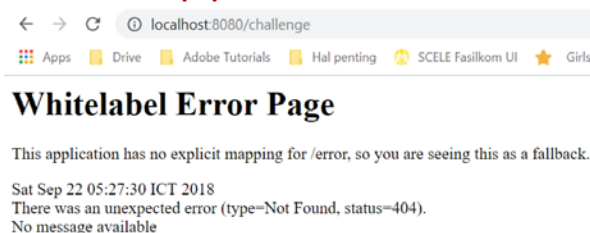
**Jawaban : challenge menjadi method yang mengarahkan /viral/challenge menuju halaman yang dituju.**



3. Akses localhost:8080/challenge.

a. Deskripsikan hasilnya

**Jawaban : Terjadi Error 404 Not Found karena path /challenge tidak dihandle oleh method apapun.**

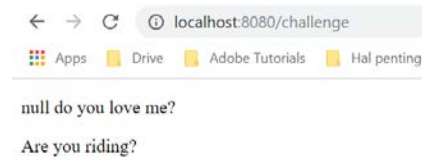


b. Ubah methodnya menjadi seperti di bawah ini, deskripsikan lalu kembalikan ke bentuk awal!

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

**Jawaban : Parameter 'name' tidak harus diisi sehingga apabila kita tidak memberikan nilai untuk parameternya, hasilnya berupa halaman di mana isian**

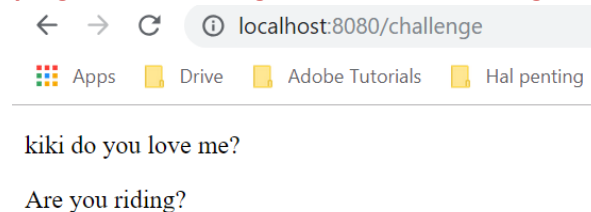
yang harus diisi dengan nilai akan diisi dengan "null".



c. Ubah methodnya menjadi seperti di bawah ini, deskripsikan lalu kembalika ke bentuk awal!

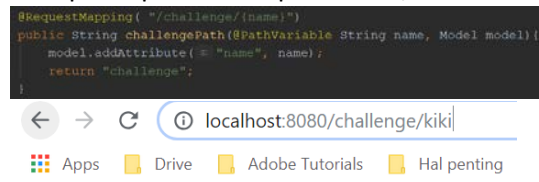
```
@RequestMapping("/challenge")
public String challenge(@RequestParam(value = "name", required = false, defaultValue = "kiki") String name,
                        Model model){
    model.addAttribute("name", name);
    return "challenge";
}
```

**Jawaban : Parameter 'name' tidak harus diisi sehingga apabila kita tidak memberikan nilai untuk parameternya, hasilnya berupa halaman di mana isian yang harus diisi dengan nilai akan diisi dengan default value-nya.**



### C. Latihan Path Variable

Setiap ada perubahan pada kode, selalu usahakan run ulang

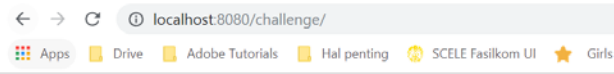


kiki do you love me?

Are you riding?

1. Akses localhost:8080/challenge/  
Apakah hasil dari halaman tersebut?

**Jawaban : Error 400 Bad Request karena method challengePath seharusnya dirujuk oleh path /challenge/{nama}**



## Whitelabel Error Page

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

Fri Sep 21 19:01:58 ICT 2018

There was an unexpected error (type=Bad Request, status=400).

Required String parameter 'name' is not present

### 2. Ubah method challengePath menjadi seperti berikut

```
@RequestMapping(value= {"/challenge", "challenge/{name}"})
public String challengePath(@PathVariable Optional<String> name, Model model){
    if (name.isPresent()){
        model.addAttribute("name", name.get());
    }else{
        model.addAttribute("name", "KB");
    }
    return "challenge";
}
```

Akses localhost:8080/challenge/

Apakah hasil dari halaman tersebut?

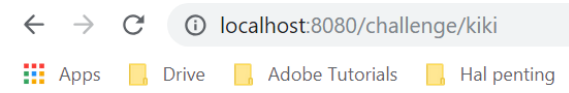
**Jawaban : Terjadi internal server error karena terdapat dua method yang melakukan request mapping ke path /challenge sehingga menjadi ambigu.**

```
@java.lang.IllegalStateException: Ambiguous handler methods mapped for HTTP path 'http://localhost:8080/challenge/':
```

### 3. Akses localhost:8080/challenge/kiki

Apakah hasil dari halaman tersebut?

**Jawaban : Request diarahkan menuju method challengePath (tidak terjadi ambiguitas seperti nomor sebelumnya) karena hanya method itu yang melakukan handling terhadap /challenge/{name}**



kiki do you love me?

Are you riding?

## Latihan

1. Buatlah sebuah halaman baru sebagai viral generator
2. Halaman ini menerima 2 buah parameter berupa dua angka
3. Pada view menampilkan viral generator berupa kata yang memiliki spesifikasi sebagai berikut:
  - a. jika tidak ada parameter hanya menampilkan kata "hm"
  - b. parameter pertama mengindikasikan jumlah huruf 'm' pada kata "hm"
  - c. default kata untuk a = 0 atau a=1 atau b=0 atau b=1 adalah "hm"
  - d. parameter kedua mengindikasikan kelipatan kata yang dihasilkan pada poin b
  - e. perkata dipisahkan oleh satu spasi

Jawaban :

```

@RequestMapping("/generator")
public String viralgenerator(@RequestParam(value = "a", required = false, defaultValue = "0") String a, @RequestParam(value = "b", required = false, defaultValue = "0") String b, Model model){
    //value dari request dijadikan integer
    int mMultiplier = Integer.parseInt(a);
    int hmMultiplier = Integer.parseInt(b);

    //kalau value 0, diberi default masing - masing multiplier
    //adalah 1
    if(mMultiplier==0){mMultiplier=1;}
    if(hmMultiplier==0){hmMultiplier=1;}

    String fWord = "h";

    //karakter "h" ditambahkan menurut jumlah "m" nya
    for (int i =1; i<=mMultiplier; i++){
        fWord+="m";
    }

    //satu kata sudah dibuat
    //menambahkan <hmMultiplier - 1> kata selanjutnya
    String result = fWord;
    for (int j =1; j<=hmMultiplier-1; j++){
        result+=(" " + fWord);
    }

    //agar dapat dirender html
    model.addAttribute("a", a);
    model.addAttribute("b", b);
    model.addAttribute("result", result);
    return "generator";
}

```

