Tutorial 2 – Arsitektur Pemograman

Pada tutorial 2 ini akan berlatih menggunakan framework Spring Boot. Outline:

● Membuat dan menggunakan Controller pada Spring Boot - project VIRAL

● Menerapkan routing pada project Spring Boot

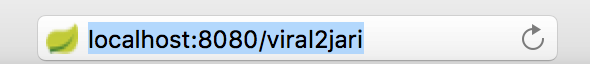
● Latihan

**Latihan Project Viral**

1.Ganti baris @RequestMapping(“/viral”) menjadi @RequestMapping(“/viral2jari”). Kemudian run kembali.

a.Apakah terjadi compile error? Tidak

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





404 mengartikan tidak ditemukannya page untuk melakukan request ke URL. Ketika melakukan @RequestMapping(“/viral2jari”) dan melakuakn render ke localhost:8080/viral terjadi eror karena alamat dari method yang mereturn page viral.html adalah @RequestMapping(“/viral2Jari”) sedangkan ketika memanggil localhost:8080/viral Request Mapping tidak memiliki method dalam alamat viral.

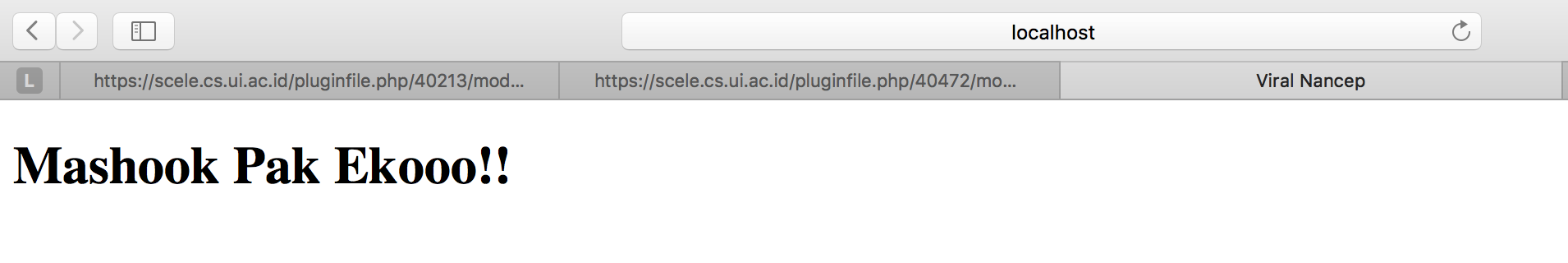
c.Apakah Kegunaan @RequestMapping(“/viral”)

dengan kata lain mapping sebuah request adalah sebuah alamat yang dimiliki oleh method,Setiap method akan dipanggil dengan menggunakan RequestMapping.

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

a.Apakah terjadi compile error? Tidak

b.Coba akses localhost:8080/viral, berikan alasan dan jelaskan apa yang terjadi. Kembalikan menjadi index()

Tidak terjadi eror, dikarenakan method tetap dipanggil oleh Request Mapping viral.

3.Ganti string return type menjadi menjadi return “viral2jari”

a.Apakah terjadi compile error? Tidak   
  
b. Coba akses localhost:8080/viral, berikan alasan dan jelaskan apa yang terjadi.

 Terjadi Eror, Karena tidak ditemukan html pada return dari method yang dipanggil oleh @RequestMapping(“/viral”)

1. Apakah kegunaan return type pada method controller tersebut?

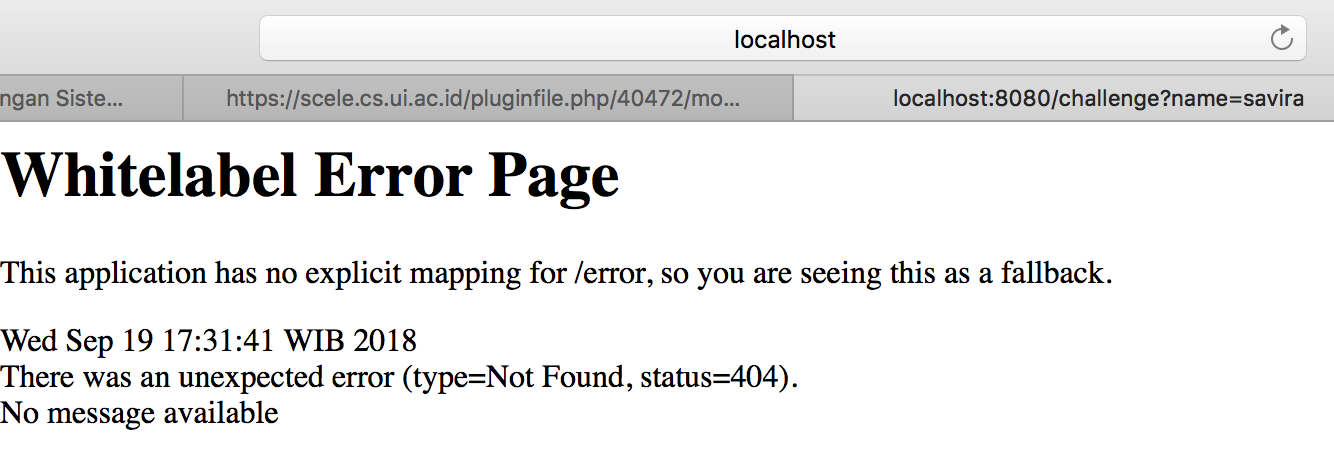
Kegunaan dari return type adalah memanggil view (file html) yang akan ditampilkan

**Latihan** **Request** **Parameter**

1. Mengapa tulisan “In My feeling” pada thymeleaf challenge tidak pernah muncul di browser?

Karena penulisan “In My Feeling” menggunakan format penulisan untuk html pada umumnya.Thymleaf sebagai template engine mempunyai format berbeda dengan html biasa.

2.Ubah nilai anotasi Request Mapping “/challenge” menjadi “/viral/challenge”. Deskripsikan hasil pada browser dan kembalikan ke bentuk sebelumnya!



Hasilnya path /challenge error 404 karena di controller tidak ada request mapping ke path /challenge (sudah diubah menjadi path /viral/challenge. Namun path /viral/challenge menjadi bisa di akses

3.Akses localhost:8080/challenge.

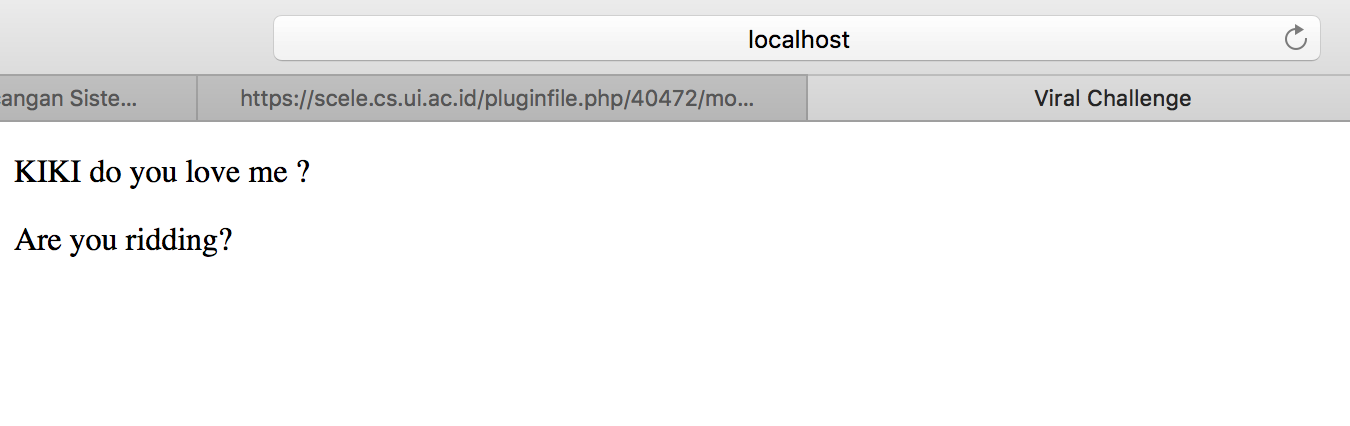
1. Deskripsikan hasilnya

Hasilnya path /challenge error 400 (bad request) karena path yang diakses tidak memiliki parameter string

1. Ubah methodnya menjadi seperti di bawah ini, deskripsikan lalu kembalikan ke bentuk awal!   
   (@RequestParam(value = “name” , required = false)

path /challenge tidak error (request benar) karena parameternya menjadi tidak required

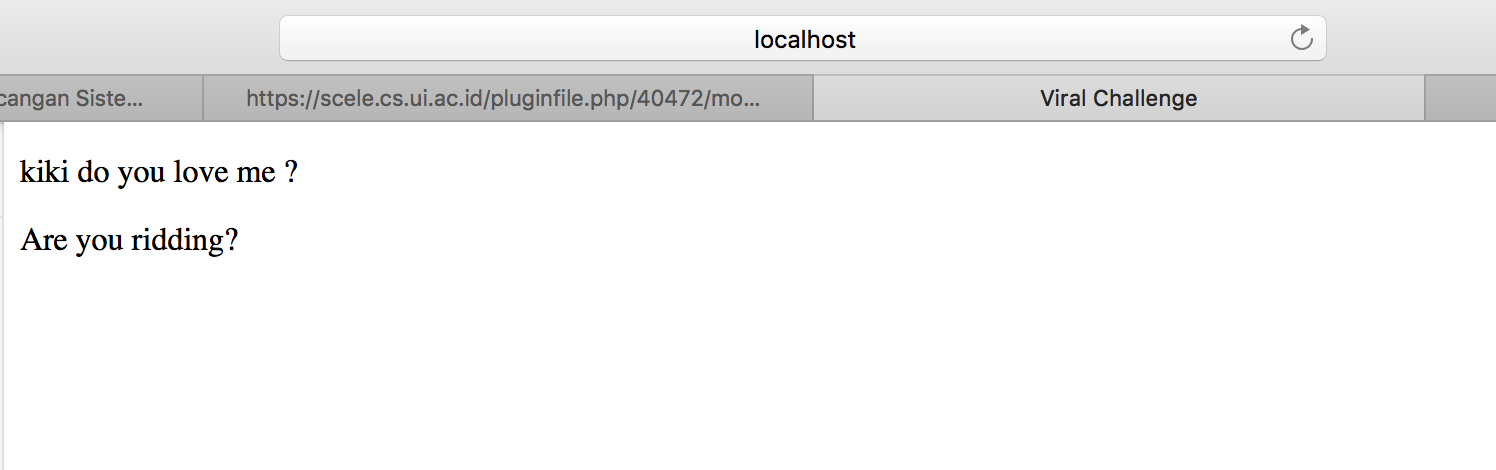
1. Ubah methodnya menjadi seperti di bawah ini, deskripsikan lalu kembalikan ke bentuk awal! 



Path /challenge dapat terbuka dengan default Value KIKI yang sudah dituliskan pada controller. Namun, semisal yang di ruting adalah /challenge/savira maka yang akan keluar pada path adalah ‘***savira*** *do you love me ?’*

**Latihan Path Variable**

****

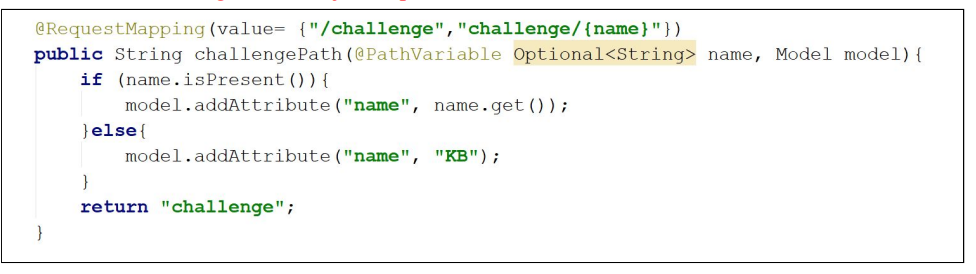
****

Path variable berfungsi sebagai method untuk mengambil input yang terdapat pada address dan pada lab ini value akan dimasukkan kesalah satu atribut di challenge.html

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

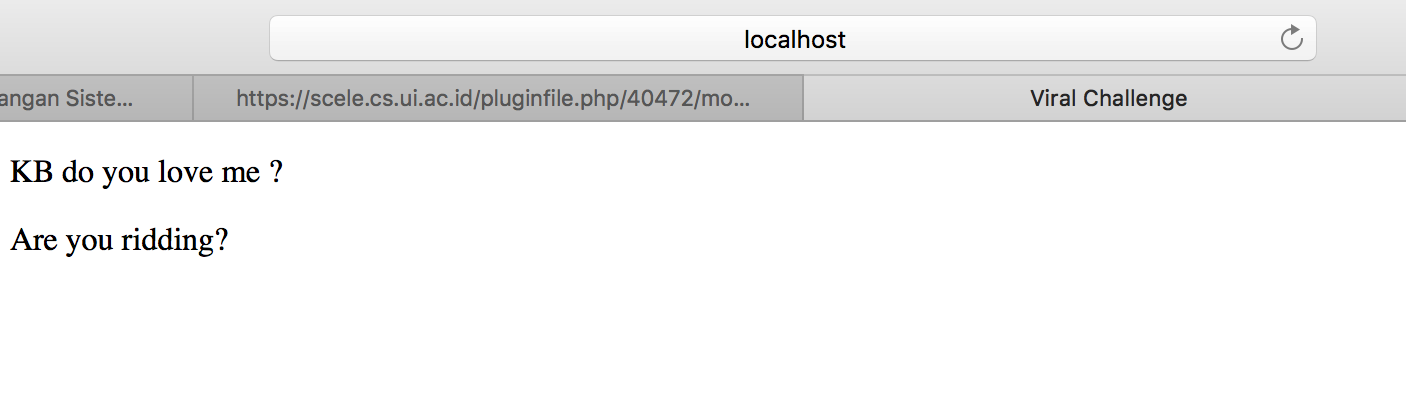
Alamat merouting ke @RequestMapping(“/challenge”) sehingga tidak terjadi eror. @RequestMapping(“/challenge/{nama}”) /{nama} hanyalah alam

2.Ubah method challengePath menjadi seperti berikut

method challengePath() melakukan pengecekan terhadap alamat , ketika tidak ada input value yang dimasukkan ({name}) maka akan return value KB ke halaman HTML. jika ada input value kehalaman challenge,html berupa value yang dimasukkan di alamat.

Untuk pemahaman lebih contoh ditunjukkan dibawah ini:

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



Akses **localhost:8080/challenge/kiki** Apakah hasil dari halaman tersebut?

