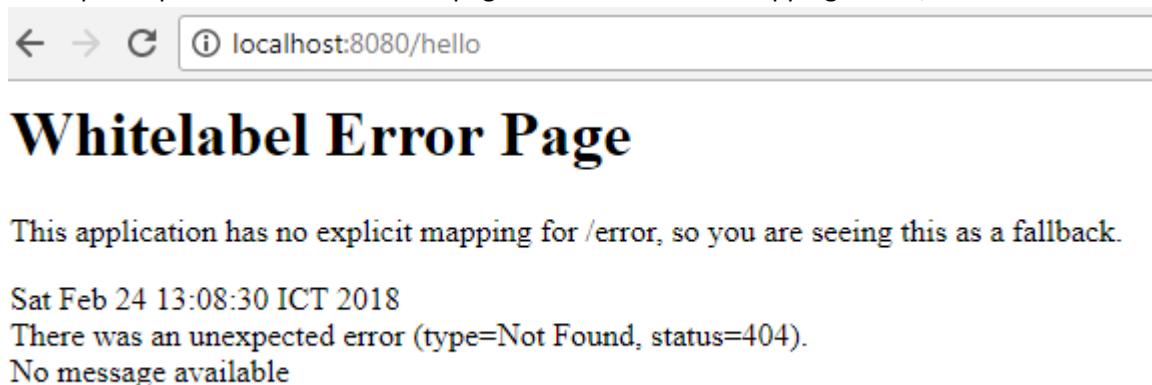
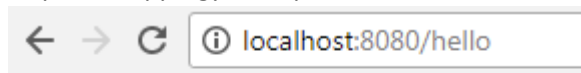


Latihan Project Hello World

1. Tidak compile error
Ditampilkan pesan "Whitelabel error page" karena tidak ada mapping untuk /hello

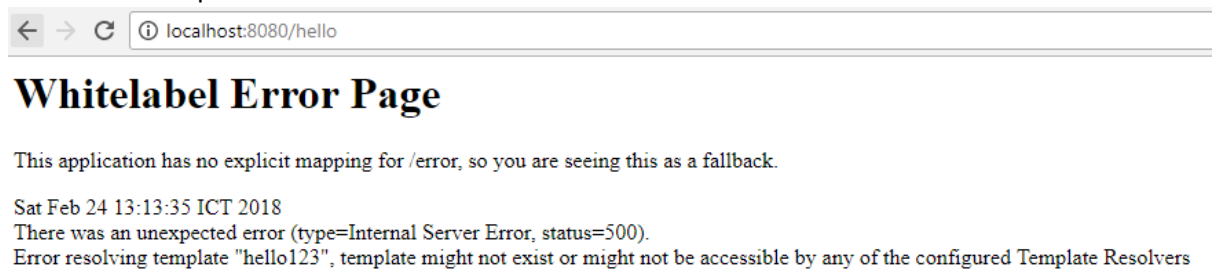


2. Tidak compile error
Iya, masih muncul karena yang diganti adalah nama method index() menjadi hello() dan request mappingnya tetap diakses melalui /hello sehingga masih muncul Hello, world!



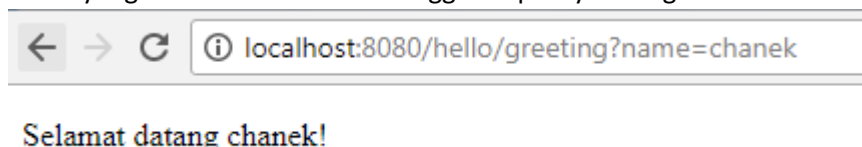
Hello, world!

3. Tidak compile error
Tidak muncul karena String tersebut menandakan nama file html yang akan diakses. File .html yang tersedia adalah hello.html sementara string yang dikembalikan hello123 sehingga muncul error seperti di bawah.



Latihan Request Parameter

1. Hasilnya : Selamat datang chanek!
Karena pemanggilan di website sama dengan definisi di RequestMapping dan parameter nama yang diberikan chanek. Sehingga outputnya sebagai berikut.



2. Membuka localhost:8080/greeting dan menghasilkan pesan “Whitelabel Error Page” karena diakses di web dengan /greeting sementara requestMapping greeting membutuhkan parameter.



Whitelabel Error Page

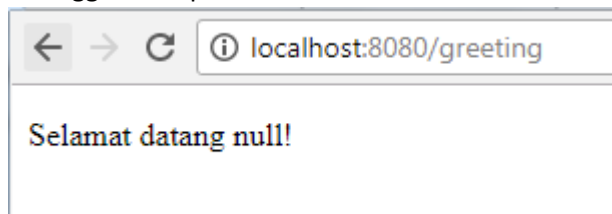
This application has no explicit mapping for /error, so you are seeing this as a fallback.

Sat Feb 24 13:20:33 ICT 2018

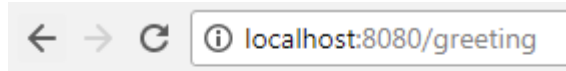
There was an unexpected error (type=Bad Request, status=400).

Required String parameter 'name' is not present

Setelah diubah menjadi Selamat datang null! Karena ditambahkan parameter required = false artinya mapping greeting dapat diakses tanpa parameter dan value parameternya tidak sehingga ditampilkan null



3. Hasilnya : Selamat datang dunia! Karena ditambahkan parameter defaultValue=dunia sehingga ketika mengakses /greeting tanpa parameter maka ditampilkan dunia.

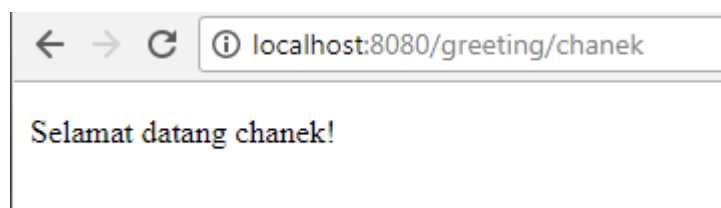


Selamat datang dunia!

4. Tulisan “Sapaan untuk user” tidak pernah muncul karena menggunakan elemen thymeleaf(th). Thymeleaf ini hanya menampilkan teks yang ada thtext:”Teks yang ditambahkan”. Untuk menampilkan teks “Sapaan untuk user” kita dapat meng-take out elemen th tersebut.

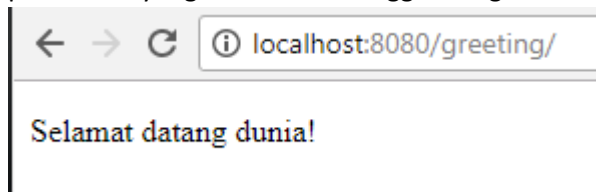
Path Variable

2. Setelah menambahkan method greetingPath dan mengakses /greeting/chanek maka ditampilkan adalah “Selamat datang Chanek” karena parameter yang diberikan chanek

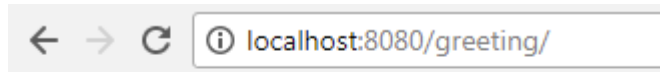


Latihan Path Variable

1. Akses localhost:8080/greeting/ hasilnya seperti gambar di bawah ini karena tidak ada parameter yang diberikan sehingga menghasilkan nilai default yaitu dunia.

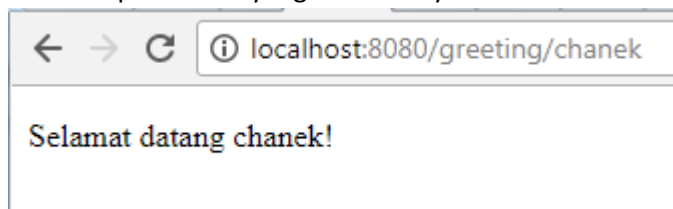


Setelah mengubah greetingPath dan mengakses localhost:8080/greeting hasilnya seperti gambar di bawah ini. Karena di kode tersebut dibuat kondisi jika tidak ada parameter maka atribut nama akan berisi nilai apap



Selamat datang apap!

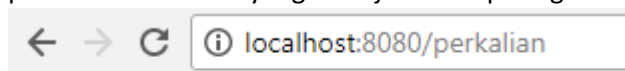
Akses localhost:8080/greeting/chanek hasilnya seperti gambar di bawah ini. Karena parameter yang diberikan adalah chanek, dan jika parameter ada maka akan ditampilkan nilai dari parameter yang diberikan yaitu chanek.



Latihan

Membuat halaman perkalian dengan optional parameter a dan b

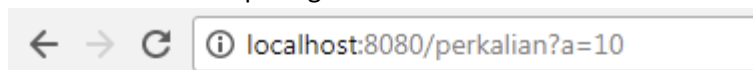
1. Jika diakses tanpa parameter maka menghasilkan nilai 0 karena nilai default dari kedua parameter adalah 0 yang ditunjukkan seperti gambar di bawah ini.



Perkalian

$$0 \times 0 = 0$$

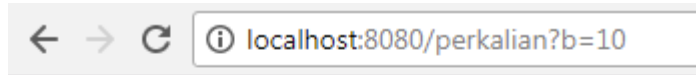
2. Jika diakses dengan parameter a bernilai 10, maka hasilnya adalah 0 karena parameter b bernilai default 0 seperti gambar bawah ini.



Perkalian

$$10 \times 0 = 0$$

3. Jika diakses dengan parameter b bernilai 10, maka hasilnya adalah 0 karena parameter a bernilai default 0 seperti gambar di bawah ini.



Perkalian

0 x 10 = 0

4. Jika diakses dengan parameter a bernilai 10 dan parameter b bernilai 10 maka akan menghasilkan nilai 100 seperti ditunjukkan gambar di bawah ini.



Perkalian

10 x 10 = 100

Ringkasan

Di dalam praktikum ini saya belajar tentang konsep MVC yang ada di Spring Boot. Dalam membuat project ada beberapa hal yang perlu diperhatikan sebagai berikut:

1. Menentukan routing dengan menggunakan notasi `@RequestMapping`. Notasi tersebut didefinisikan di atas method yang akan dipanggil. Jadi ketika di web kita mengakses url sesuai dengan `RequestMapping` yang didefinisikan maka akan dijalankan method yang di bawahnya.
2. Method yang dijalankan dapat di set memiliki parameter atau tidak. Untuk mendefinisikan parameter dapat menggunakan `RequestParamater` dan diikuti model untuk meng-set atribut dari parameter. Selain itu dalam mendefinisikan parameter kita juga dapat menggunakan `PathVariable` dan diikuti model untuk meng-set atribut dari parameter.
3. Kita juga dapat menentukan tambahan kondisi untuk parameter yang kita buat seperti `required = false` (artinya method tersebut dapat diakses dengan parameter dan juga dapat diakses tanpa parameter). Kemudian ada `defaultValue` untuk menentukan nilai default parameter ketika tidak ada parameter yang dikirimkan.
4. Kita dapat menentukan nilai dari attribut dengan menggunakan `model.addAttribute("name", name)`. Attribute name tersebut akan dikirimkan ke view dengan nama file sesuai String yang direturn oleh method.
5. Di view membuat tampilan html. Di file tersebut kita dapat menangkap parameter yang dikirimkan dari Contoller melalui `model.addAttribute` dengan menggunakan elemen html `th:text`