

Atikah Luthfiana

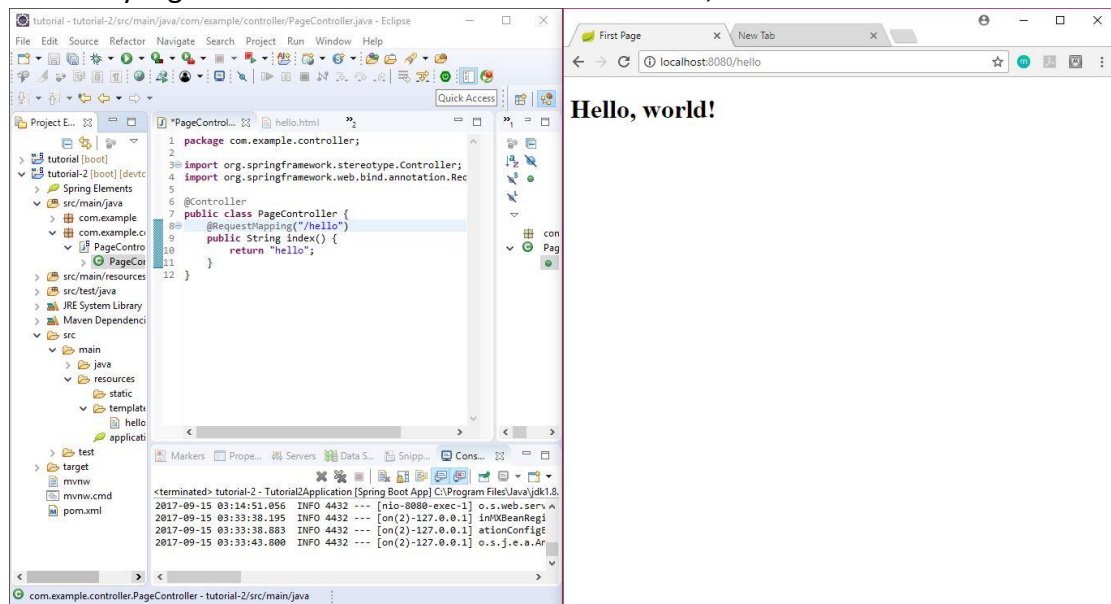
1506689250

ADPAP B

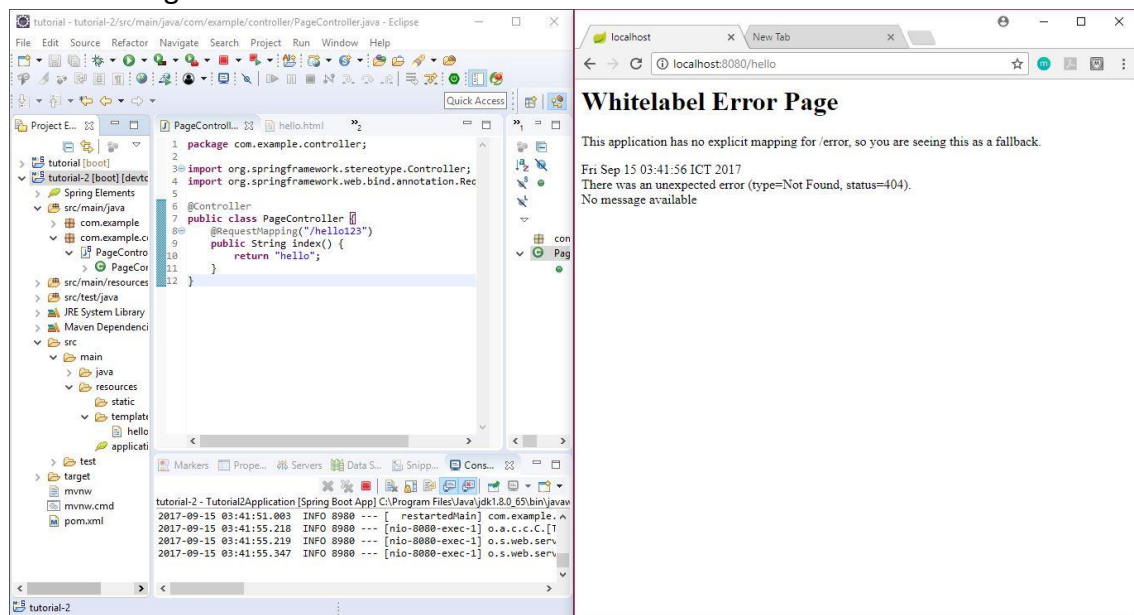
## Tutorial 2

### 1. Hello, World!

- a. Anotasi biasa digunakan untuk memberikan info tambahan. Contohnya, `@Controller` berguna untuk memberitahukan bahwa `class PageController` merupakan kelas `Controller`. `@RequestMapping` merupakan salah satu bentuk anotasi yang paling sering digunakan, yang berfungsi untuk melakukan *mapping request* dari web ke sebuah `class` atau `method` tertentu. Dibawah ini dibuat *request mapping* untuk web *url hello* yang kemudian akan mencetak tulisan "Hello, World!".



- b. Ketika dilakukan perubahan pada request mapping menjadi `/hello123` yang terjadi adalah sebagai berikut:



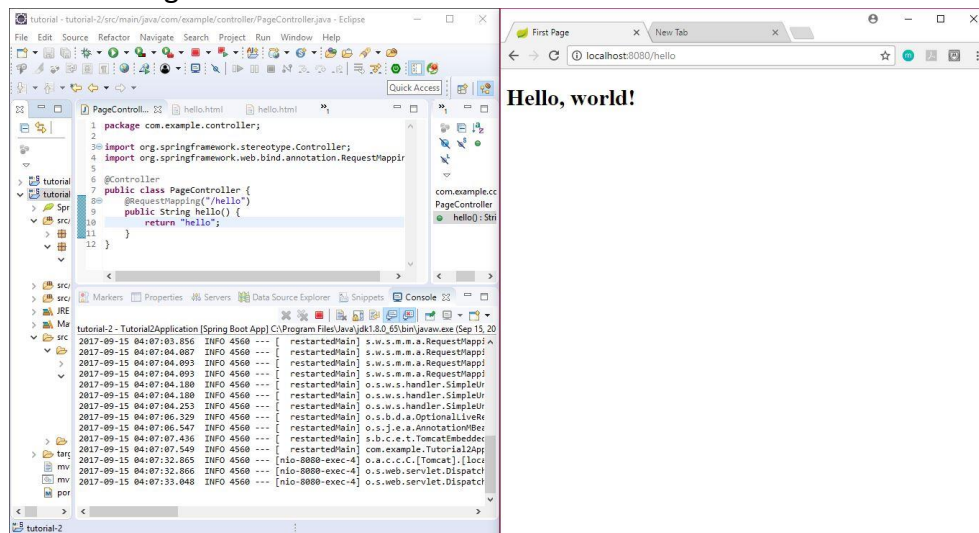
Atikah Luthfiana

1506689250

ADPAP B

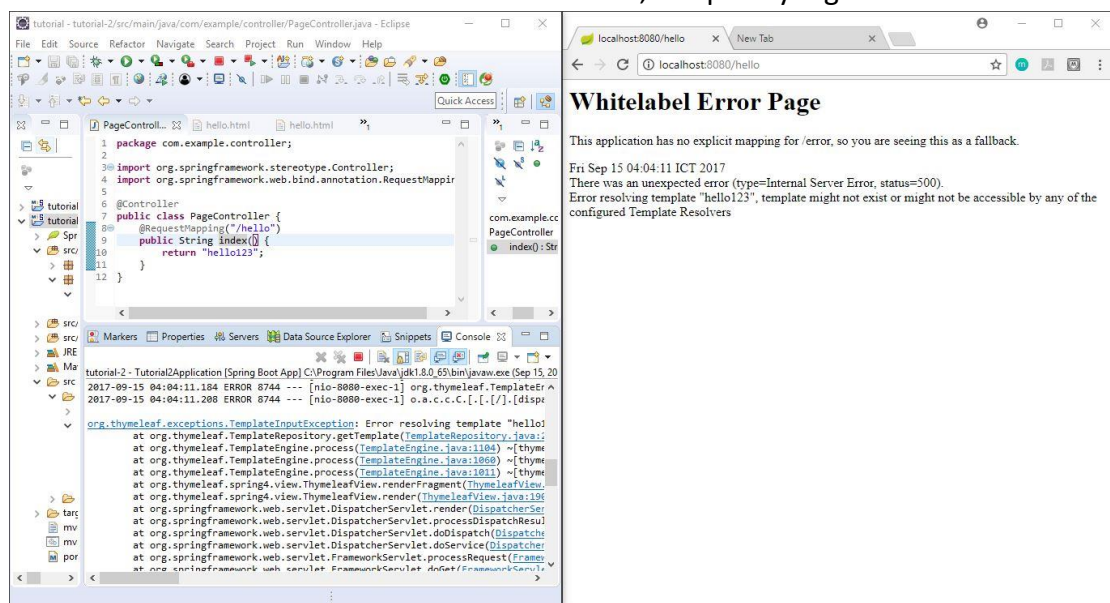
Karena fungsi utama dari *request mapping* adalah untuk melakukan *mapping request* dari web ke sebuah *class* atau *method* tertentu, perubahan pada url yang dituliskan setelah tulisan *@RequestMapping* didalam tanda () akan mengalami *error*. Hal ini dikarenakan, *mapping* yang dilakukan gagal karena alamat url pada anotasi dan *return* dari *method* berbeda. Ketika *RequestMapping* melakukan *mapping* ke alamat url hello123, tidak ada yang dapat ditampilkan karena tidak ada method yang melakukan *return* ke alamat url hello123. Sehingga tidak ada pesan yang ditampilkan.

- c. Ketika dilakukan pengubahan nama method menjadi hello(), keluaran yang dihasilkan adalah sebagai berikut:



Perubahan nama *method* tidak mempengaruhi proses *mapping* yang dilakukan oleh anotasi *@RequestMapping*. *RequestMapping* akan sukses selama nama *url* dalam anotasi dan *return* dari method sama.

- d. Ketika return dari method index tersebut diubah, tampilan yang dihasilkan adalah:



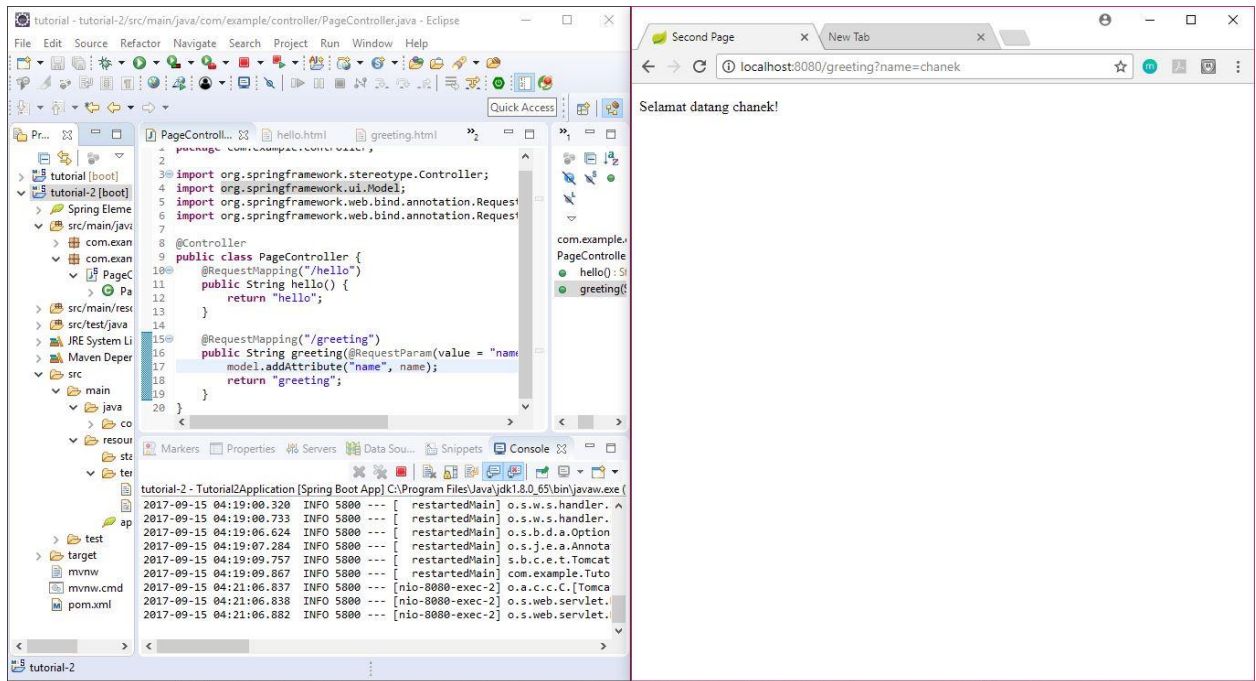
Atikah Luthfiana

1506689250

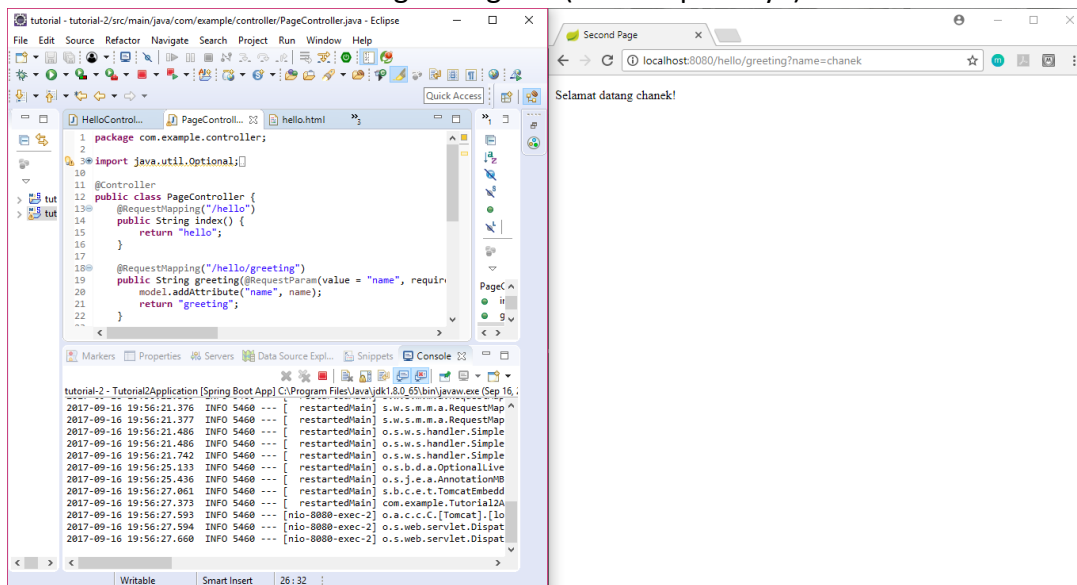
ADPAP B

Error ini terjadi disebabkan oleh hal yang sama pada nomor 1. b, yaitu nama *url* yang telah dideklarasikan pada anotasi tidak sama dengan *return* dari *method* yang dituju untuk melakukan proses *mapping*. Disini, alamat url hello123 tidak ada/belum dibuat. Sehingga tidak ada *url* yang dituju untuk *method* tersebut melakukan *mapping*.

2. `@RequestParam` merupakan anotasi yang digunakan untuk mengambil nilai parameter dari sebuah *url* untuk kemudian bersama dengan `@RequestMapping` parameter yang telah didapat di *mapping* ke argumentasi sebuah *method*.



- a. Mapping berhasil dilakukan karena alamat mapping pada `@RequestMapping` telah di ubah dan return dari method `greeting` ada (ada templatennya)



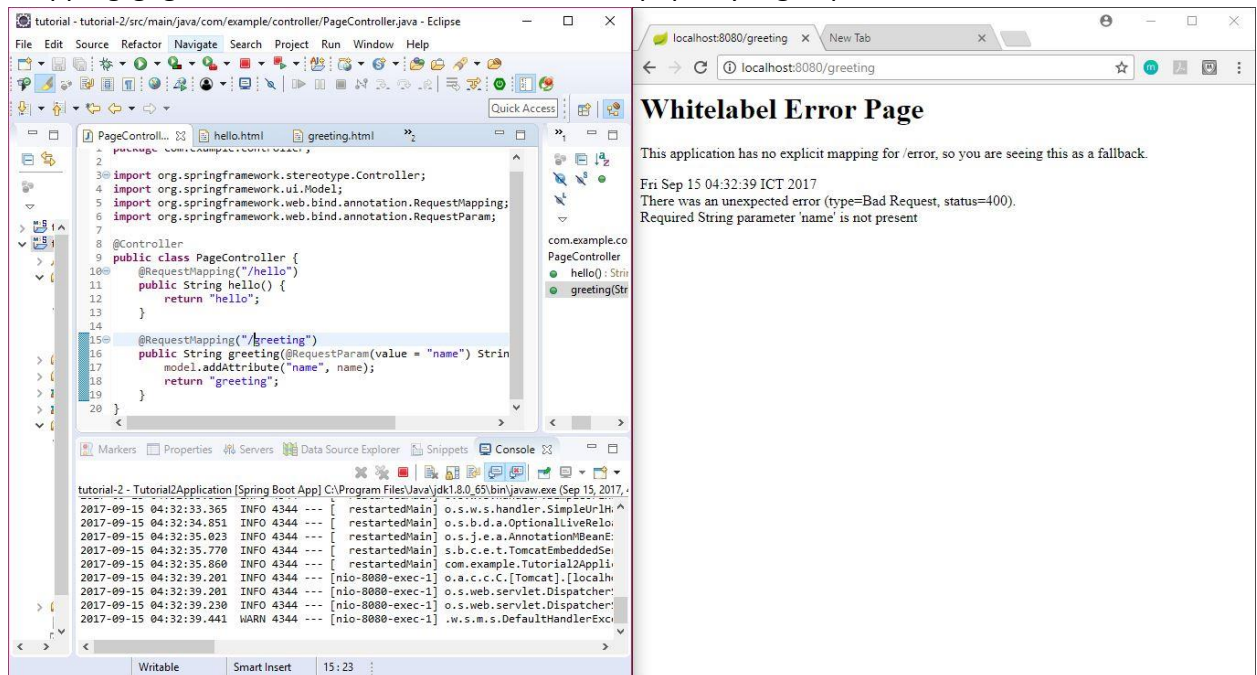


Atikah Luthfiana

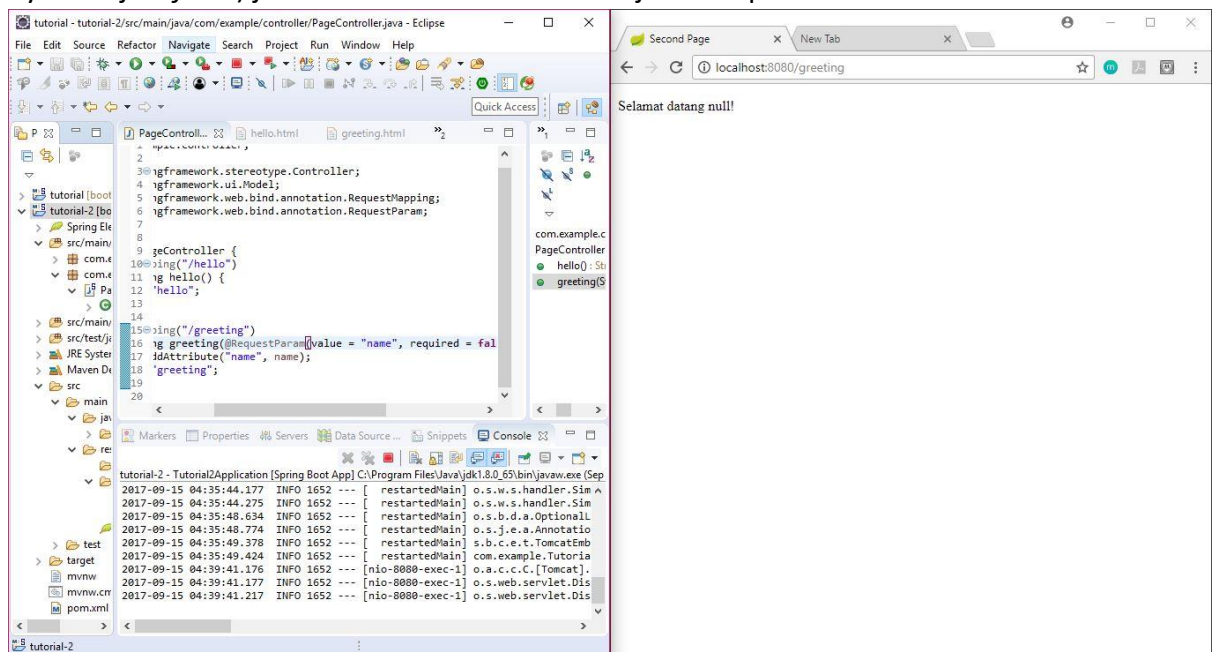
1506689250

ADPAP B

- b. *Mapping* gagal dilakukan karena tidak ada nilai apapun yang dapat diambil.



- c. *Mapping* kini dapat dilakukan namun *request param* tidak mengambil nilai apapun dari url karena tidak ada parameter apapun di url, sehingga keluaran yang dihasilkan akan null. Hal ini dikarenakan pengambilan parameter dari url tidak *required* (telah di-set *required-nya* menjadi *false*) jadi *method* tersebut bisa berjalan tanpa *error*.



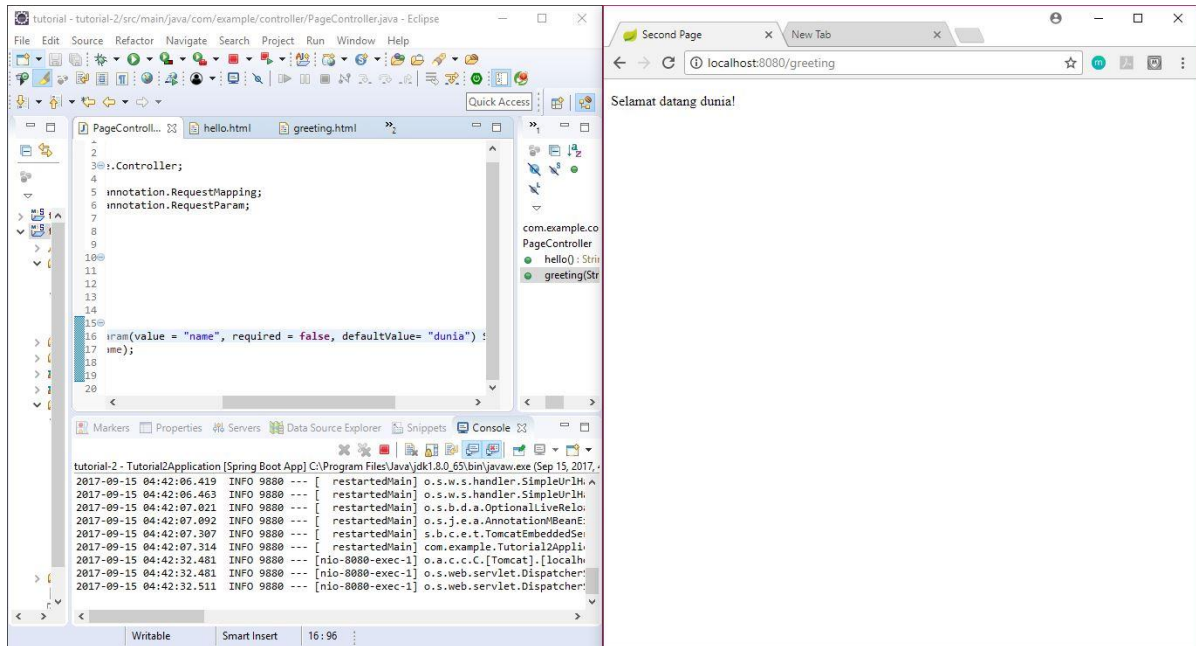
- d. Pada argumentasi *methodnya*, *default value* telah di-set. Sehingga ketika tidak ada parameter yang didapat, maka *requestparam* akan memasukkan nilai default yaitu "dunia". Berbeda dengan bagian 2c, pengambilan nilai parameter dari url belum ada

Atikah Luthfiana

1506689250

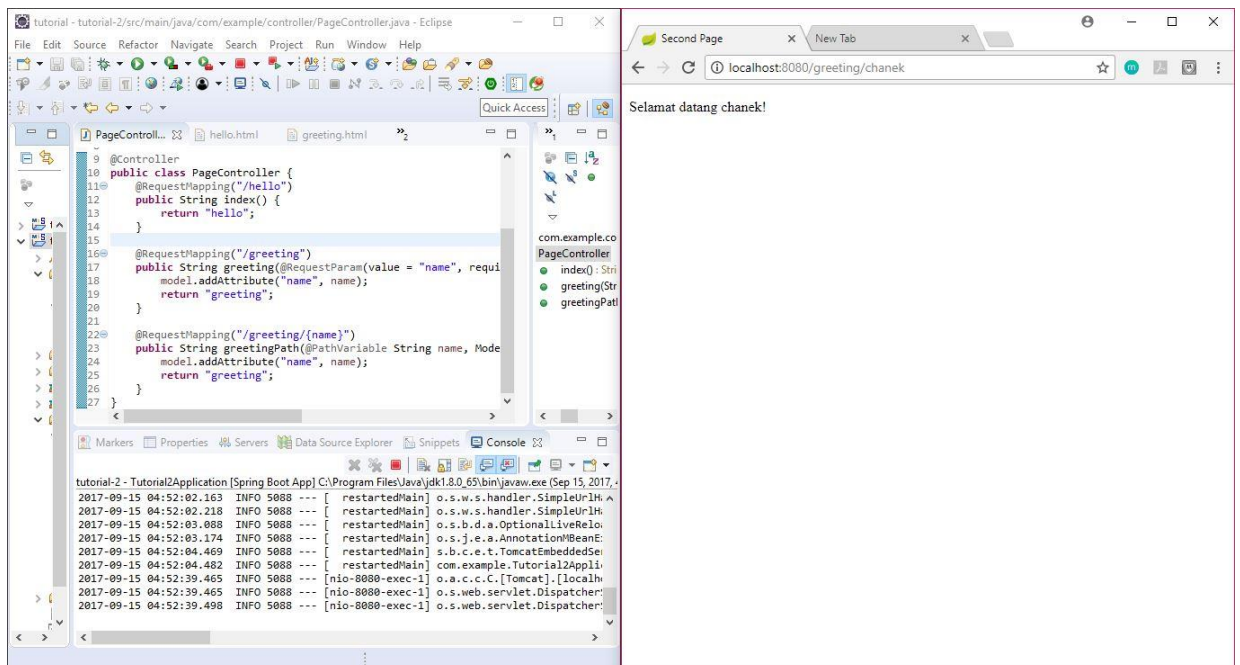
ADPAP B

*default value* yang diinginkan. Sehingga ketika tidak ada param yang didapat maka *requestparam* akan menganggap nilai yang masuk adalah *null*/kosong.



- e. Sapaan untuk user tidak pernah muncul akibat dari adanya `th:text` sehingga seakan-akan `th:tx` menyebabkan proses keluar dari tag `p` sehingga kalimat selanjutnya tidak dijalankan.

### 3. Path Variable



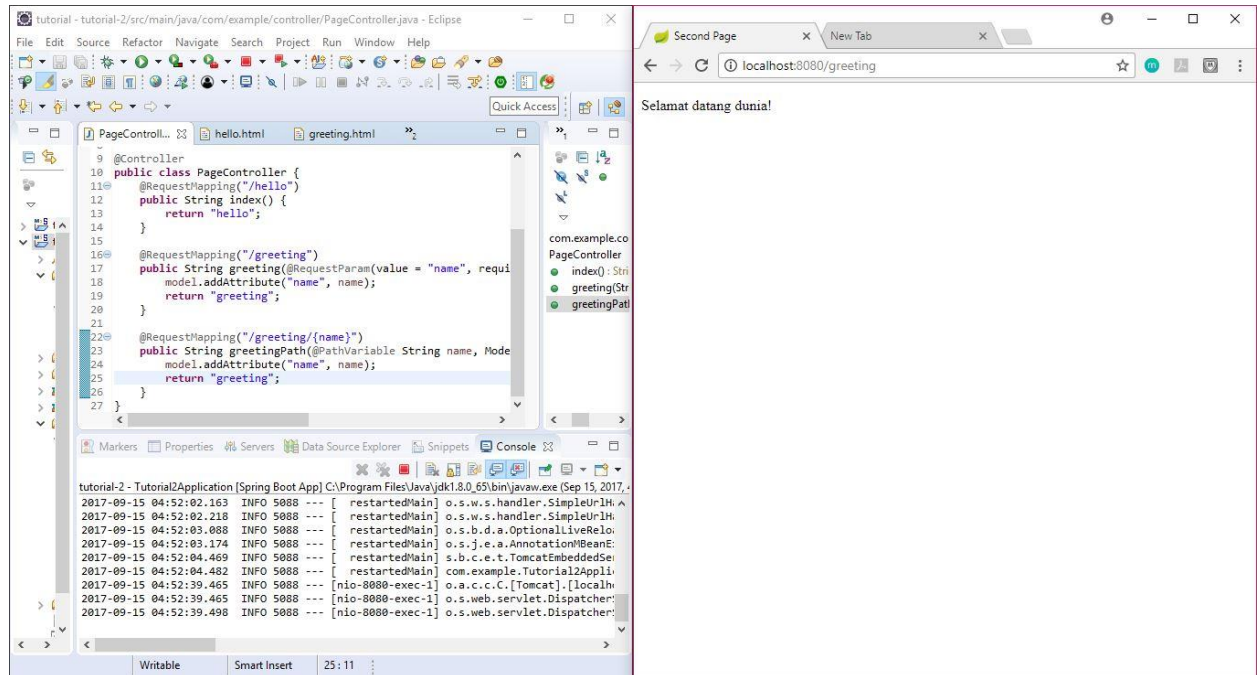
- a. Pada saat *url* hanya `/greeting`, maka *mapping* akan dilakukan ke *method* `greeting`, bukan `greetingPath`. Hal ini dikarenakan *url mapping* yang telah dideklarasikan *method* `greeting` adalah `(/greeting)` sementara *method* `greetingPath` adalah `(/greeting/{nama})`. *Mapping* dari *url* akan diterukan ke *method* yang bersesuaian

Atikah Luthfiana

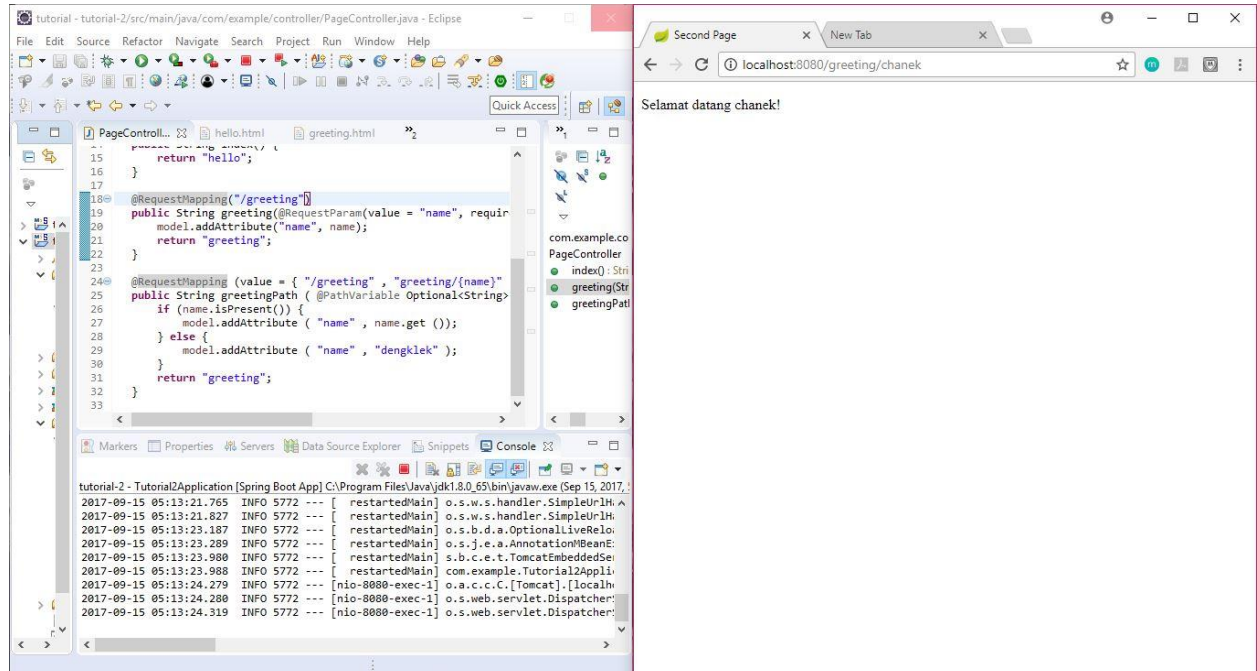
1506689250

ADPAP B

alamat url pada `@RequestMapping`-nya. Sehingga hasil yang dikeluarkan adalah sebagai berikut:



b. Method `greetingPath` kemudian diubah. Setelah dilakukan perubahan pada alamat url, dengan mengisi nama chanek maka hasilnya akan seperti berikut:



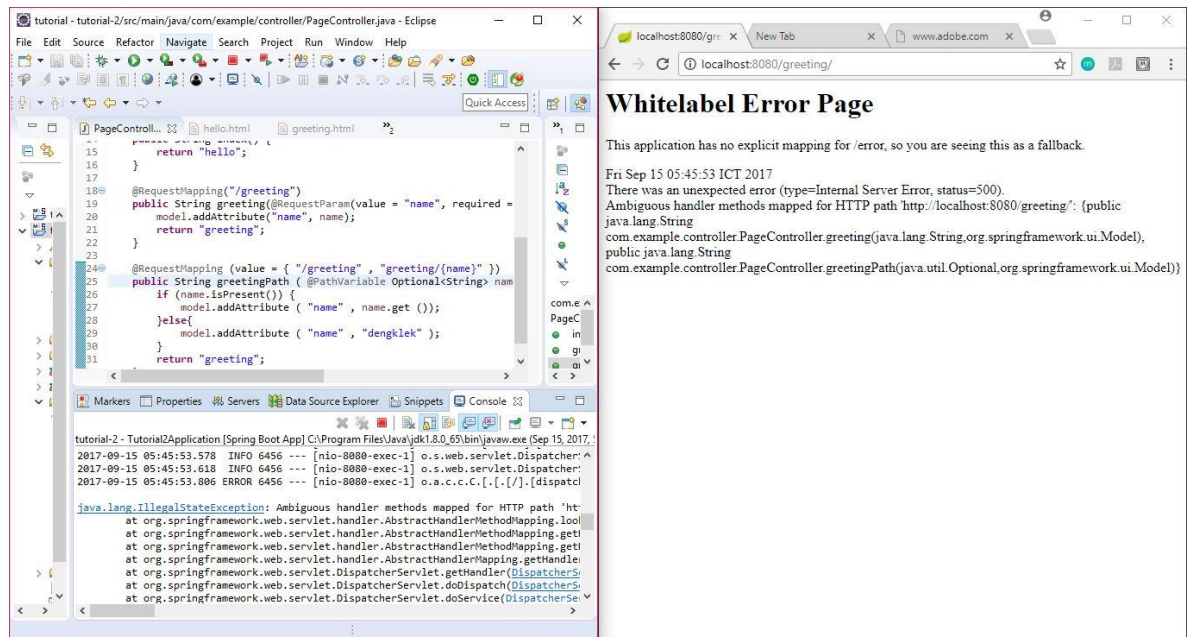
Nama yang dimasukkan dalam url akan dicetak kembali. Hal ini dikarenakan *mapping* yang dilakukan oleh *Path Variable*. Selanjutnya dicoba dengan tidak menambahkan url path apapun (url yang dijalankan adalah `/greeting`). Ketika dijalankan muncul error sebagai berikut:



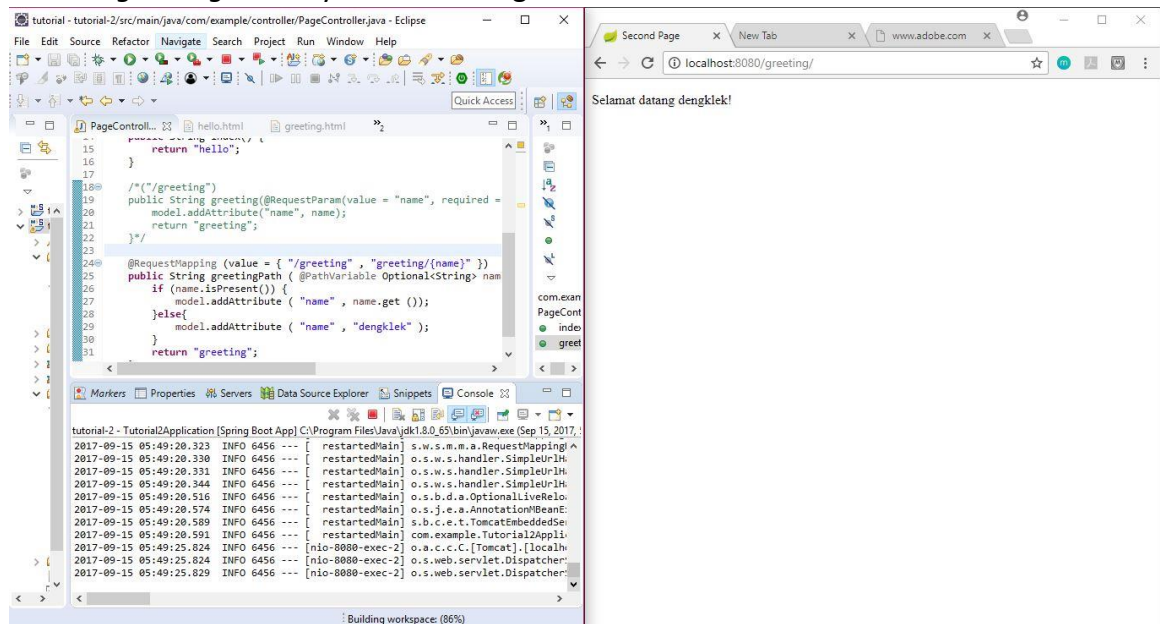
Atikah Luthfiana

1506689250

ADPAP B



Seharusnya, setelah dilakukan *optional mapping*, hasil yang dikeluarkan adalah *default* keluaran yang telah dideklarasikan. Pada *default optional*, ketika tidak ada *path* nama yang ditemukan, maka *default path* nama menjadi dengklek. Yang kemudian dengklek ini akan dicetak dilayar. Namun justru terjadi error. Hal ini dikarenakan ada 2 *mapping url* yang sama, yaitu method `greeting` dan `greetingPath`. Selanjutnya untuk melihat hasil kerja method `greetingPath` saya telah *comment* method `greeting`. Hasilnya adalah sebagai berikut.



Ketika tidak ada name path, tetap dapat berjalan karena telah di-set *optional* untuk *path variable* yang dibuat.

Atikah Luthfiana  
1506689250  
ADPAP B

## **Referensi**

“Spring @RequestMapping @RequestParam @PathVariable Example.” JournalDev, 5 July 2016,  
[www.journaldev.com/3358/spring-requestmapping-requestparam-pathvariable-example](http://www.journaldev.com/3358/spring-requestmapping-requestparam-pathvariable-example).  
Accessed 16 Sept. 2017.