

Nama : Izzatunnisa Arwadi

NPM : 1806269726

Kelas : Ekstensi

1. Java SE

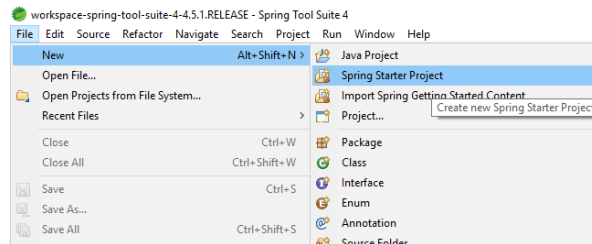
Melakukan download JDK serta instalasi. Setelah instalasi selesai dilakukan setting javapath

2. Spring Tool Suite

Melakukan download STS for windows dan melakukan extract sehingga STS dapat dijalankan

3. Membuat project baru

- Spring starter project



Untuk memulai project spring diawali dengan klik File >> New >> Spring Starter Project

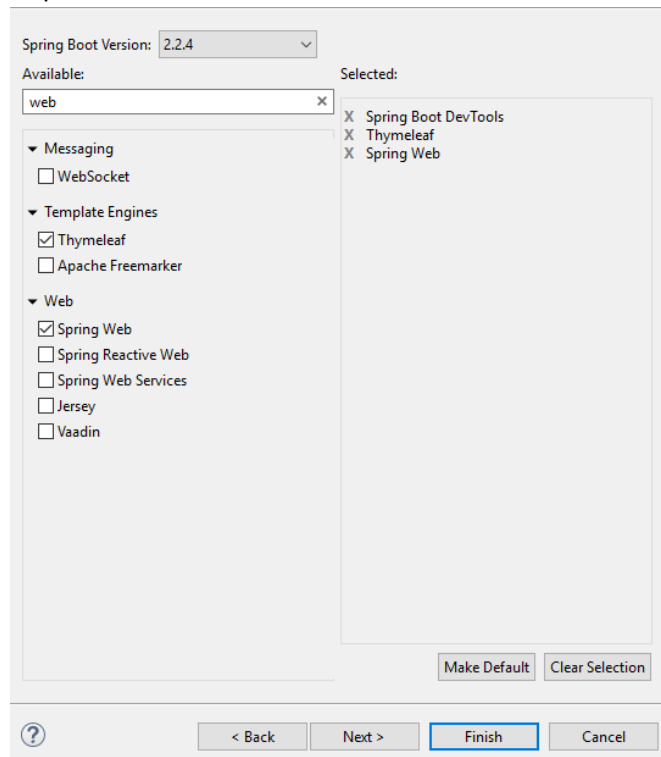
- Penamaan file

New Spring Starter Project

A screenshot of the 'New Spring Starter Project' dialog box. The 'Service URL' is set to 'https://start.spring.io'. The 'Name' field contains 'tutorial-02'. The 'Use default location' checkbox is checked. The 'Location' field shows the path 'C:\Users\ASUS\Documents\workspace-spring-tool-suite-4-4.5.1.REL'. The 'Type' is 'Maven', 'Packaging' is 'Jar', 'Java Version' is '8', and 'Language' is 'Java'. The 'Group' is 'com.example', 'Artifact' is 'tutorial-02', 'Version' is '0.0.1-SNAPSHOT', 'Description' is 'Demo project for Spring Boot', and 'Package' is 'com.example.demo'. At the bottom, there are buttons for '< Back', 'Next >', 'Finish', and 'Cancel'. The 'Next >' button is highlighted.

Setelah itu dilakukan penamaan project. Pada tutorial kali ini project name dinamakan tutorial-02

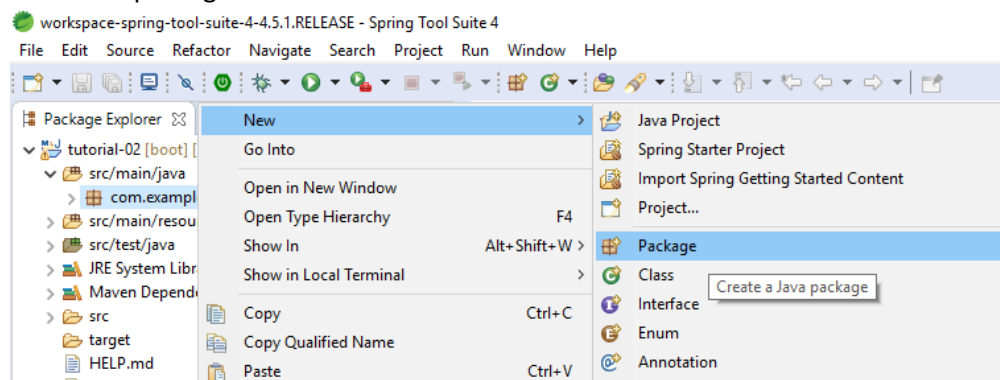
- Dependencies



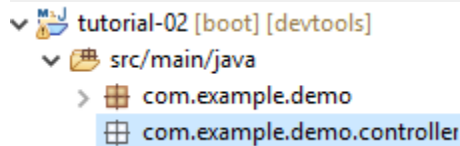
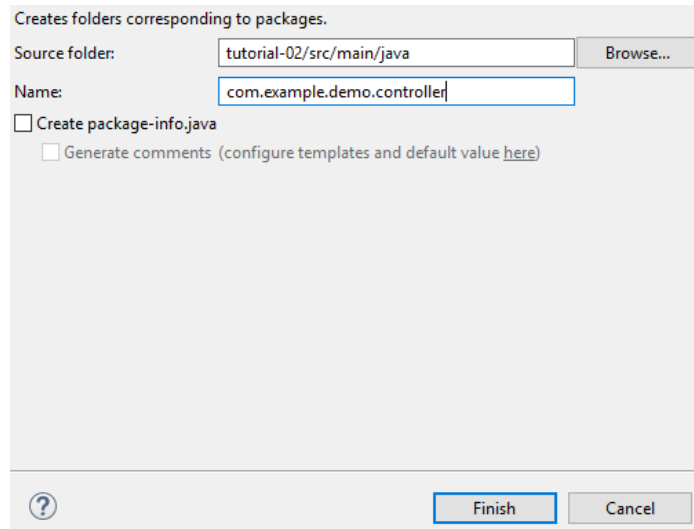
Akan ditampilkan pilihan dependencies yang akan digunakan pada project. Pada tutorial kali ini dependencies yang digunakan adalah DevTools, Thymeleaf, dan Web

#### 4. Project Hello

- Membuat package controller

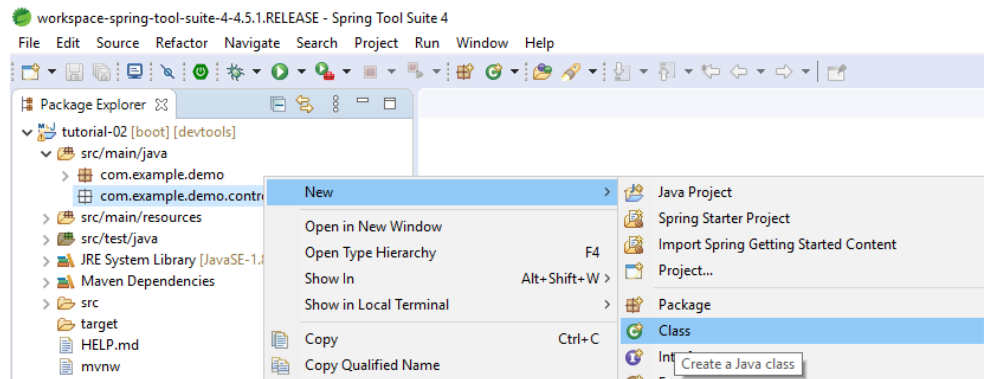


Setelah project baru berhasil dibuat, dilakukan pembuatan package baru pada com.example.demo.



Package tersebut diberi nama `com.example.demo.controller`. Apabila package telah berhasil dibuat maka akan dimunculkan pada treeview project.

- **Buat Class PageController.java**



Dilakukan pembuatan class pada package yang telah dibuat sebelumnya.

Name:

Modifiers: ☒ public ☐ package ☐ private ☐ protected  
☐ abstract ☐ final ☐ static

Superclass:

Interfaces:

Which method stubs would you like to create?  
☐ public static void main(String[] args)  
☐ Constructors from superclass  
☒ Inherited abstract methods

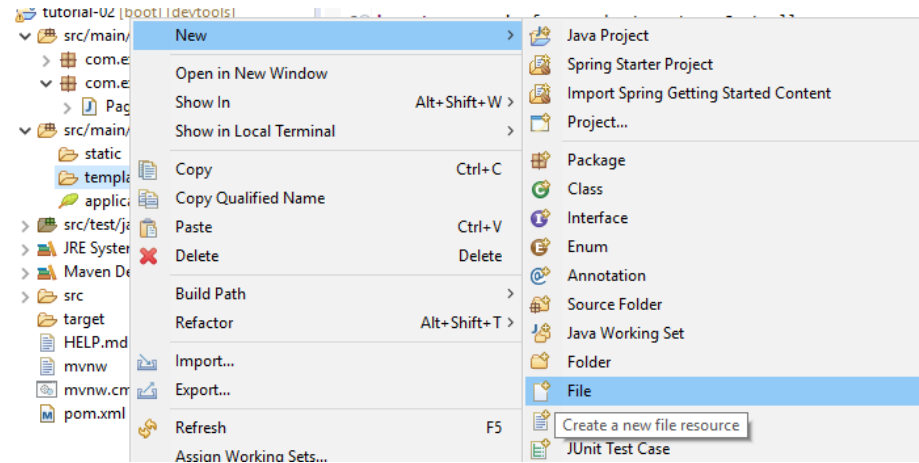
Do you want to add comments? (Configure templates and default value [here](#))  
☐ Generate comments

File tersebut diberi nama PageController

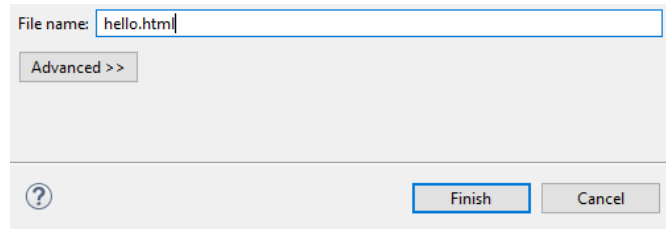
```
*PageController.java
1 package com.example.demo.controller;
2
3 import org.springframework.stereotype.Controller;
4 import org.springframework.web.bind.annotation.RequestMapping;
5
6 @Controller
7 public class PageController {
8     @RequestMapping("/hello")
9     public String index() {
10         return "hello";
11     }
12 }
13
```

Pada file PageController diisikan kode seperti di atas.

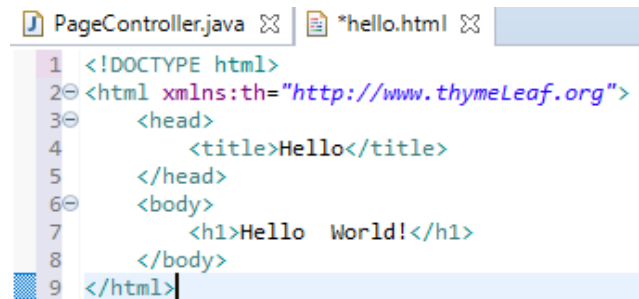
- Buat view hello.html



Dilakukan pembuatan file pada template dengan melakukan klik kanan pada template>new>file

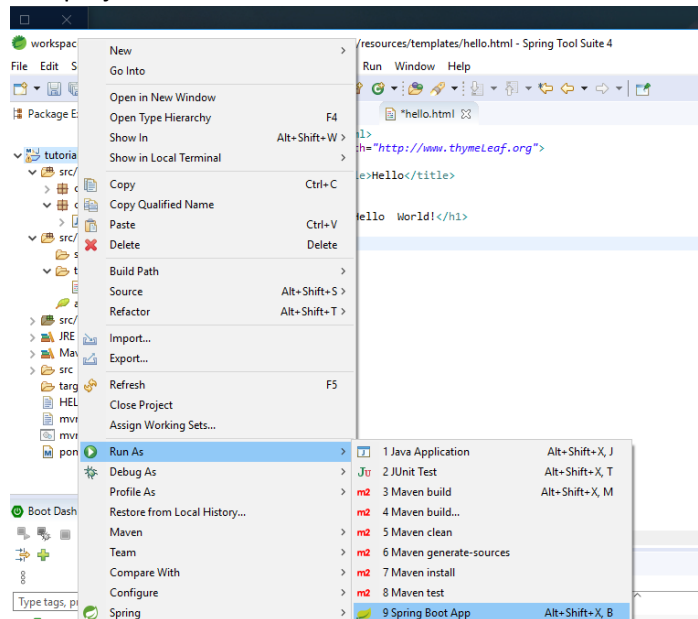


File tersebut diberi nama hello.html

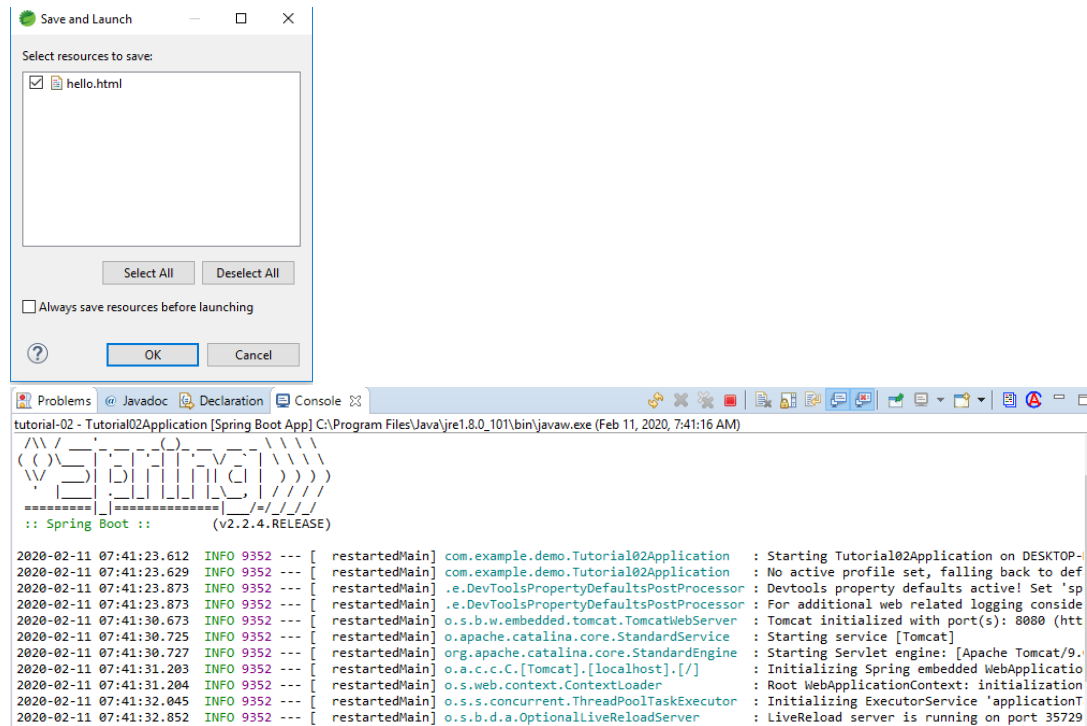


File hello.html diisikan kode seperti di atas.

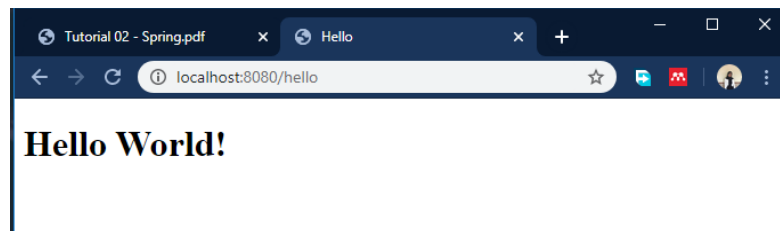
- Run project “Hello”



Untuk menjalankan project hello dilakukan klik kanan pada folder tutorial-02>run as>spring boot app



Akan ditampilkan console seperti di atas.



Setelah itu dilakukan akses pada localhost:8080/hello sehingga dapat menampilkan text Hello World!

### Latihan Project "Hello"

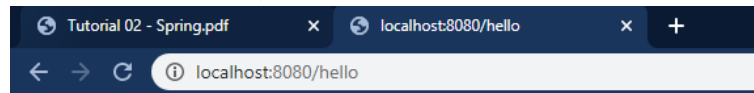
- 1) Mengganti `@RequestMapping("/hello")` menjadi `@RequestMapping("/aloha")` kemudian run

```
public class PageController {
    @RequestMapping("/aloha")
    public String index() {
        return "hello";
    }
}
```

- Tidak terjadi compile error pada console

Pada tampilan console di atas tidak terdapat error ketika dilakukan compile

- Akses localhost:8080/hello



## Whitelabel Error Page

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

Tue Feb 11 07:48:38 ICT 2020

There was an unexpected error (type=Not Found, status=404).

No message available

Akan tetapi pada browser terlihat bahwa terdapat error pada saat mapping. Hal ini dikarenakan adanya pemanggilan `@RequestMapping("/aloha")` pada code sedangkan yang diakses pada browser adalah `localhost:8080/hello`.


- Kegunaan `@RequestMapping("/hello")`  
`@RequestMapping("/hello")` memiliki fungsi sebagai penanda untuk URL yang digunakan untuk melakukan akses URL (mapping halaman hello)

## 2) Mengganti method index() menjadi nama method hello

```
public String hello() {
```

- Tidak terjadi compile error pada console

Pada tampilan console di atas tidak terdapat error ketika dilakukan compile

- 

Tidak terjadi error, hal ini dikarenakan penamaan method tidak memiliki kesamaan dengan method lainnya sehingga program dapat tetap berjalan.

- ```
return "aloha";
```

- ```

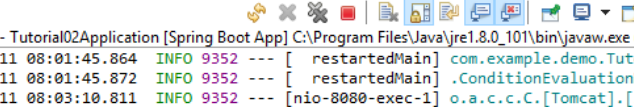
tutorium-02 - Tutorium02Application [Spring Boot App] C:\Program Files\Java\jdk-8.0.101\bin\java.exe (Feb 11, 2020, 7:41:16 AM)

:: Spring Boot ::
(v2.2.4.RELEASE)

2020-02-11 08:01:44.134 INFO 9352 --- [ restartedMain] com.example.demo.Tutorial02Application : Starting Tutorial02Application on DESKTOP-
2020-02-11 08:01:44.135 INFO 9352 --- [ restartedMain] com.example.demo.Tutorial02Application : No active profile set, falling back to def
2020-02-11 08:01:45.361 INFO 9352 --- [ restartedMain] o.s.b.w.embedded.tomcat.TomcatWebServer : Tomcat initialized with port(s): 8080 (htt
2020-02-11 08:01:45.364 INFO 9352 --- [ restartedMain] o.apache.catalina.core.StandardService : Starting service [Tomcat]
2020-02-11 08:01:45.366 INFO 9352 --- [ restartedMain] org.apache.catalina.core.StandardEngine : Starting Servlet engine: [Apache Tomcat/9.
2020-02-11 08:01:45.430 INFO 9352 --- [ restartedMain] o.a.c.c.c.[Tomcat].[localhost].[/] : Initializing Spring embedded WebApplication
2020-02-11 08:01:45.431 INFO 9352 --- [ restartedMain] o.s.web.context.ContextLoader : Root WebApplicationContext: initialization
2020-02-11 08:01:45.580 INFO 9352 --- [ restartedMain] o.s.s.concurrent.ThreadPoolTaskExecutor : Initializing ExecutorService 'applicationT
2020-02-11 08:01:45.765 INFO 9352 --- [ restartedMain] o.s.b.d.a.OptionalLiveReloadServer : LiveReload server is running on port 35729
2020-02-11 08:01:45.843 INFO 9352 --- [ restartedMain] o.s.b.w.embedded.tomcat.TomcatWebServer : Tomcat started on port(s): 8080 (http) wit
2020-02-11 08:01:45.864 INFO 9352 --- [ restartedMain] com.example.demo.Tutorial02Application : Started Tutorial02Application in 1.846 sec
2020-02-11 08:01:45.872 INFO 9352 --- [ restartedMain] .ConditionEvaluationDeltaLoggingListener : Condition evaluation unchanged

```

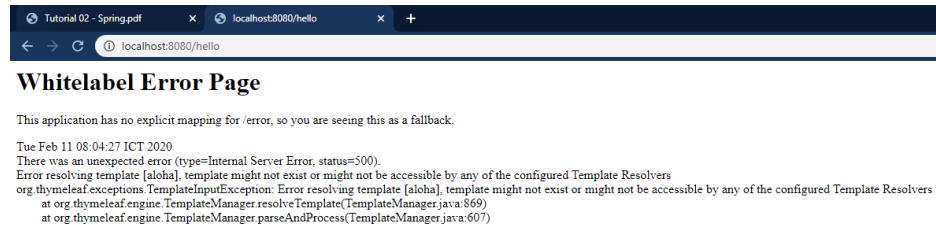
- Akses localhost:8080/hello

- 
- The screenshot shows an IDE with the 'Console' tab selected. The console output includes application startup logs and an exception stack trace. The logs show the application starting successfully on 2020-02-11. The exception is a `org.thymeleaf.exceptions.TemplateInputException` with the message 'Error resolving template [aloha], template not found', which occurred in the `render` method of `ThymeleafView` at line 189.
- ```
tutorial-02 - Tutorial02Application [Spring Boot App] C:\Program Files\Java\jre1.8.0_101\bin\javaw.exe (Feb 11, 2020)
2020-02-11 08:01:45.864 INFO 9352 --- [ restartedMain] com.example.demo.Tutorial02Ap
2020-02-11 08:01:45.872 INFO 9352 --- [ restartedMain] .ConditionEvaluationDeltaLogg
2020-02-11 08:03:10.811 INFO 9352 --- [nio-8080-exec-1] o.a.c.c.C.[Tomcat].[localhost
2020-02-11 08:03:10.813 INFO 9352 --- [nio-8080-exec-1] o.s.web.servlet.DispatcherServ
2020-02-11 08:03:10.827 INFO 9352 --- [nio-8080-exec-1] o.s.web.servlet.DispatcherServ
2020-02-11 08:03:11.049 ERROR 9352 --- [nio-8080-exec-1] org.thymeleaf.TemplateEngine

org.thymeleaf.exceptions.TemplateInputException: Error resolving template [aloha], tem
    at org.thymeleaf.engine.TemplateManager.resolveTemplate(TemplateManager.java:81
    at org.thymeleaf.engine.TemplateManager.parseAndProcess(TemplateManager.java:6
    at org.thymeleaf.TemplateEngine.process(TemplateEngine.java:1098) [thymeleaf-3
    at org.thymeleaf.TemplateEngine.process(TemplateEngine.java:1072) [thymeleaf-3
    at org.thymeleaf.spring5.view.ThymeleafView.renderFragment(ThymeleafView.java:
    at org.thymeleaf.spring5.view.ThymeleafView.render(ThymeleafView.java:189) [th
    at org.springframework.web.servlet.DispatcherServlet.render(DispatcherServlet:
    at org.springframework.web.servlet.DispatcherServlet.processDispatchResult(Dis
    at org.springframework.web.servlet.DispatcherServlet.doDispatch(DispatcherServ
```

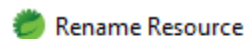
Akan tetapi ketika dilakukan akses localhost:8080/hello pada console muncul beberapa error



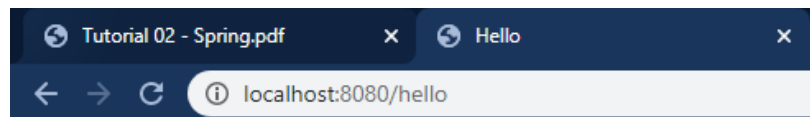


Terjadi error pada program hal ini dikarenakan return atau pengembalian aloha tidak dikenali. Hal ini dikarenakan tidak adanya template dengan nama aloha

- Mengganti hello.html menjadi aloha.html dan akses localhost:8080/hello



Dilakukan penggantian nama hello.html menjadi aloha.html



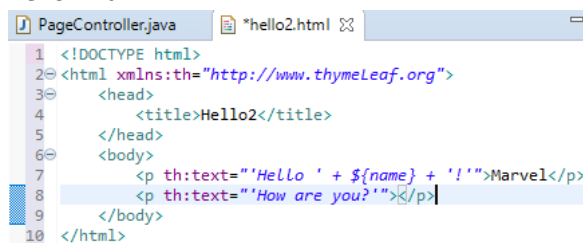
## Hello World!

Ketika dilakukan akses localhost:8080/hello ditampilkan text Hello World!. Hal ini dikarenakan return atau pengembalian aloha dikenali (terdapat file template aloha)

- Kegunaan return method  
Kegunaan dari return method pada file controller berfungsi untuk memanggil file html yang berada di template. Oleh karena itu nama file html serta return pada method harus memiliki nama yang sama.

### 5. Request Parameter (Query String)

- hello2.html



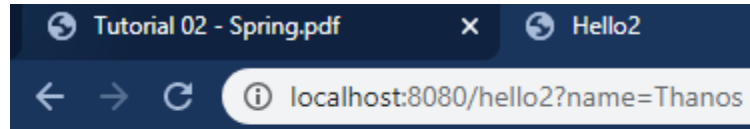
Dilakukan pembuatan file hello2.html pada template dengan isi kode seperti di atas.

- Menambahkan method

```
@RequestMapping("/hello2")
public String hello2(@RequestParam(value = "name") String name, Model model) {
    model.addAttribute("name", name);
    return "hello2";
}
```

Dilakukan penambahan method pada pageController dengan nama method hello2 dan isi kode seperti di atas.

- Akses localhost:8080/hello2?name=Thanos



Hello Thanos!

How are you?

Dilakukan akses pada localhost:8080/hello2?name=Thanos sehingga ditampilkan text:

Hello Thanos!

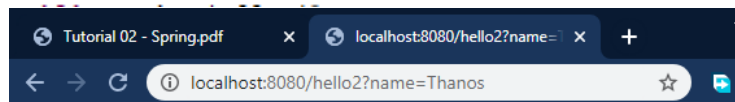
How Are you?

Hal ini dikarenakan adanya pengiriman parameter GET yang berisikan value Thanos dan di lakukan passing value tersebut ke view.

#### Latihan Request Parameter

- 1) Hal ini dikarenakan adanya penggunaan th:text attribute yang menggantikan text marvel menjadi value yang berada dalam tag (parameter pada GET) sehingga value tersebut yang nantinya akan ditampilkan.
- 2) Mengubah anotasi menjadi “/hello/hello2”

```
@RequestMapping("/hello/hello2")
```



## Whitelabel Error Page

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

Tue Feb 11 08:33:31 ICT 2020

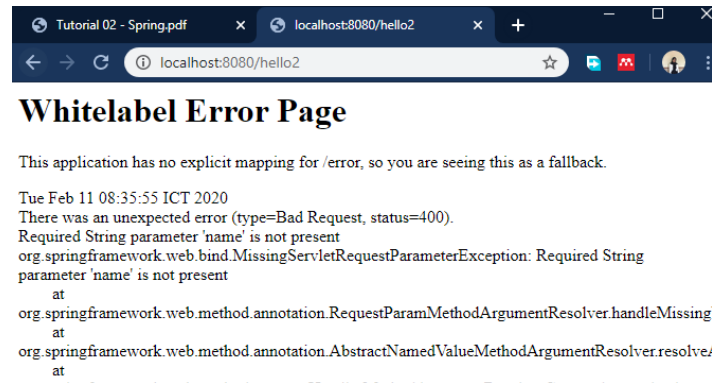
There was an unexpected error (type=Not Found, status=404).

No message available

Ketika dilakukan perubahan request mapping menjadi “/hello/hello2” pada pageController maka akan ditampilkan error pada browser. Hal ini dikarenakan mapping yang dimaksud tidak ada.

### 3) Akses localhost:8080/hello2

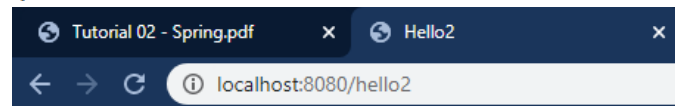
#### - localhost:8080/hello2



Terjadi error pada browser, hal ini dikarenakan tidak adanya parameter atau value yang dilempar ke view sehingga error akan muncul.

#### - Mengubah method

```
@RequestMapping("/hello2")
public String hello2(@RequestParam(value = "name", required = false) String name, Model model) {
    model.addAttribute("name", name);
    return "hello2";
}
```



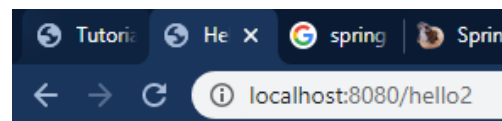
Hello null!

How are you?

Ketika dilakukan pengubahan isi method, akan ditampilkan pada browser tampilan seperti di atas. Ketika tidak ada parameter atau value yang dilempar ke view, maka akan ditampilkan null. Hal ini dikarenakan dilakukannya setting required=false

#### - Mengubah method

```
@RequestMapping("/hello2")
public String hello2(@RequestParam(value = "name", required=false, defaultValue="Thanos") String name, Model model) {
    model.addAttribute("name", name);
    return "hello2";
}
```



Hello Thanos!

How are you?

Dilakukan pengubahan isi method dengan menggunakan defaultValue dan hasil pada browser ditampilkan seperti di atas. Muncul value Thanos yang dijadikan nilai default ketika parameter atau value tidak ada.

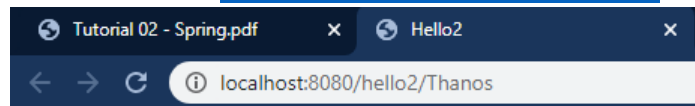
## 6. Path variable

- Menambahkan method helloPath

```
@RequestMapping("/hello2/{name}")
public String helloPath(@PathVariable String name, Model model) {
    model.addAttribute("name", name);
    return "hello2";
}
```

Dilakukan penambahan method baru yang dinamakan helloPath dan diisi dengan kode seperti di atas.

- Melakukan akses <http://localhost:8080/hello2/Thanos>



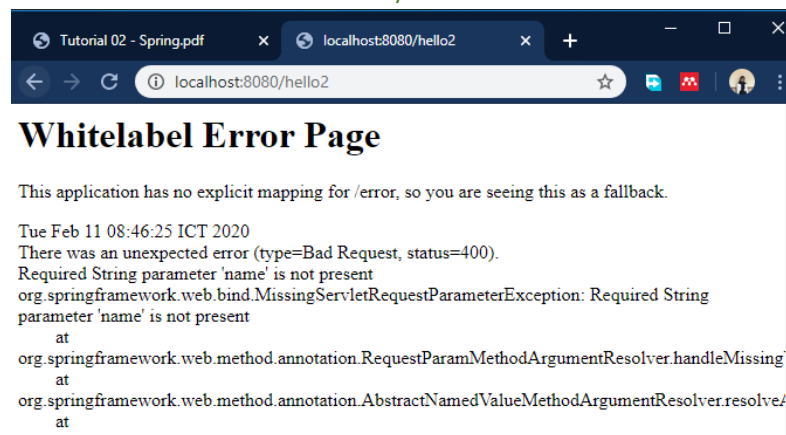
Hello Thanos!

How are you?

Dilakukan akses <http://localhost:8080/hello2/Thanos> sehingga menampilkan tampilan seperti di atas.

### Latihan Path Variable

- Melakukan akses localhost:8080/hello2



Ketika dilakukan akses localhost:8080/hello2 terdapat error. Hal ini dikarenakan tidak adanya penanganan ketika parameter tidak ada.

- Mengubah method helloPath

```
@RequestMapping(value={"/hello2", "/hello2/{name}"})
public String helloPath(@PathVariable Optional<String> name, Model model) {
    if(name.isPresent()) {
        model.addAttribute("name", name.get());
    } else {
        model.addAttribute("name", "Phoenix");
    }
    return "hello2";
}
```

Dilakukan pengubahan isi pada method helloPath dengan isi kode seperti di atas.

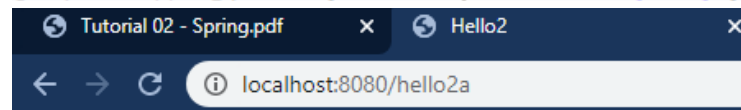
- Melakukan akses localhost:8080/hello2



Ketika dilakukan akses pada localhost:8080/hello2 terdapat ambiguous yakni adanya 2 nama mapping yang sama dengan method berbeda.

Sehingga pada request mapping dilakukan perubahan value.

```
@RequestMapping(value={"/hello2a", "/hello2/{name}"})
```

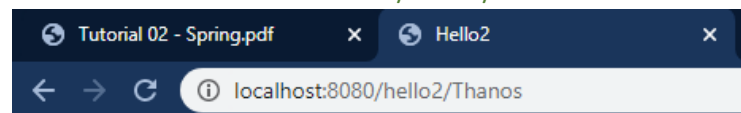


Hello Phoenix!

How are you?

Ketika dilakukan akses pada value yang diganti namanya, akan dimunculkan tampilan dengan value name Phoenix. Hal ini dikarenakan sebelumnya dilakukannya set addAttribute ketika name tidak diinputkan (parameter tidak ada).

- Melakukan akses localhost:8080/hello2/Thanos



Hello Thanos!

How are you?

Ketika dilakukan akses localhost:8080/hello2/Thanos akan ditampilkan value name Thanos. Hal ini dikarenakan adanya set addAttribute dengan menggunakan value pada parameter.