

Praktikum 1

Spring Submit Form

1. Pengantar

- Spring Framework adalah framework yang digunakan untuk membangun aplikasi berbasis web dan Java. Dengan menggunakan Spring Framework, mahasiswa dapat membangun aplikasi dengan mudah dan cepat
- Dalam proses pembuatan projek, semua dilakukan menggunakan Netbeans IDE yang terkenal akan keramahan tampilan dan penggunaannya. Nantinya mahasiswa dapat memulai mengerjakan kode-kode aplikasi tersebut
- Praktikum ini akan membantu mahasiswa untuk mempelajari dan menggunakan Spring Framework dan Java

2. Pembahasan

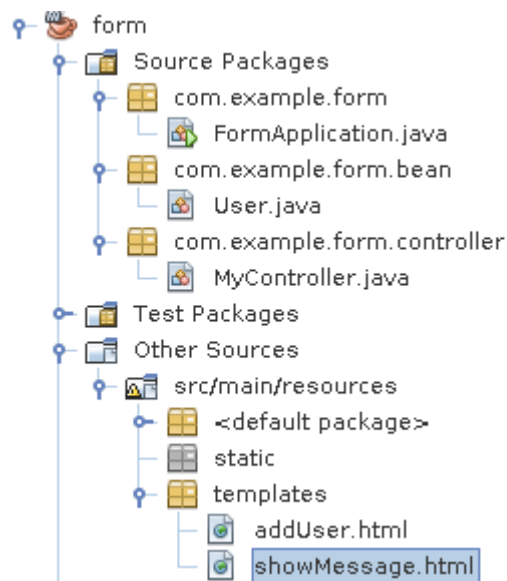
Praktikum ini dimulai dengan pembuatan projek pertama dengan menggunakan Spring Boot Initizr untuk cara mudah mendapatkan dependency. Namun sebelum projek dimulai, kebutuhan NetBeans harus terpenuhi seperti: Tomcat Server/GlassFish, Java EE plugin, Java Web plugin. Jika sudah terpenuhi mahasiswa dapat memulai projeknya.

1. Buatlah projek dengan menggunakan Spring Boot Initizr dari web <https://start.spring.io>, kemudian masukkan nama dan dependency sebagai berikut:

Tak

Project	Maven Project	Gradle Project			
	Java	Kotlin	Groovy		
Language					
Spring Boot	2.2.0 M2	2.2.0 (SNAPSHOT)	2.1.5 (SNAPSHOT)	2.1.4	1.5.20
Project Metadata	Group com.example				
	Artifact form				
	More options				
Dependencies See all	Search dependencies to add		Selected dependencies		
	Web, Security, JPA, Actuator, Devtools...		<div>Web [Web] Servlet web application with Spring MVC and Tomcat</div> <div>Thymeleaf [Template Engines] Thymeleaf templating engine</div>		

2. Kemudian import projek yang telah dibuat ke NetBeans. Lalu buatlah struktur folder dan file sesuai dengan gambar berikut:



3. Setelah itu bukalah User.java dan kemudian masukkan kode berikut ini:

```
package com.example.form.bean;

public class User {
    private String name;
    private String occupation;
```

```
public String getName() {  
    return name;  
}  
  
public void setName(String name) {  
    this.name = name;  
}  
  
public String getOccupation() {  
    return occupation;  
}  
  
public void setOccupation(String occupation) {  
    this.occupation = occupation;  
}  
}
```

4. Lalu bukalah MyController.java, dan masukkan kode berikut ini:

```
package com.example.form.controller;  
  
import com.example.form.bean.User;  
import org.springframework.stereotype.Controller;  
import org.springframework.web.bind.annotation.GetMapping;  
import org.springframework.web.bind.annotation.PostMapping;  
  
@Controller  
public class MyController {  
    @GetMapping("/addUser")  
    public String sendForm(User user) {  
  
        return "addUser";  
    }  
  
    @PostMapping("/addUser")  
    public String processForm(User user) {  
  
        return "showMessage";  
    }  
}
```

5. Tahap terakhir dari kode utama adalah ubah FormApplication.java, lalu masukkan kode berikut.

```
package com.example.form;  
  
import org.springframework.boot.SpringApplication;  
import org.springframework.boot.autoconfigure.SpringBootApplication;  
import org.springframework.boot.web.servlet.support.SpringBootServletInitializer;
```

```
@SpringBootApplication
public class FormApplication extends SpringBootServletInitializer {
    public static void main(String[] args) {
        SpringApplication.run(FormApplication.class, args);
    }
}
```

6. Kemudian buatlah layout dari addUser.html sebagai berikut:

```
<!DOCTYPE HTML>
<html xmlns:th="http://www.thymeleaf.org">
<head>
    <title>Add user</title>
    <meta charset="UTF-8">
</head>
<body>
<h1>Add User</h1>
<form action="#" th:action="@{/addUser}" th:object="${user}" method="post">
    <p>
        Name: <input type="text" th:field="*{name}">
    </p>
    <p>
        Occupation: <input type="text" th:field="*{occupation}">
    </p>
    <p>
        <input type="submit" value="Submit"/> <input type="reset" value="Reset">
    </p>
</form>
</body>
</html>
```

7. Lalu ubahlah showMessage.html dan masukkan kode berikut:

```
<!DOCTYPE HTML>
<html xmlns:th="http://www.thymeleaf.org">
<head>
    <title>Show message</title>
    <meta charset="UTF-8">
</head>
<body>
<h1>Result</h1>
<p th:text="'Name: ' + ${user.name}"></p>
<p th:text="'Occupation: ' + ${user.occupation}"></p>
```

```
<a href="/addUser">Submit another message</a>
</body>
</html>
```

8. Build dan Run projek, dan kemudian buka browser ke: <http://localhost:8080/addUser>

9. Maka kita bisa melihat form untuk memasukkan nama dan pekerjaan

Add User

Name:

Occupation:

Result

Name: Maulana

Occupation: Programmer

[Submit another message](#)

3. Penugasan