Laborator 5: MVC Model View Controller

1. Creați un proiect maven folosind archetype maven-archetype-webapp

2. Adaugați dependențele

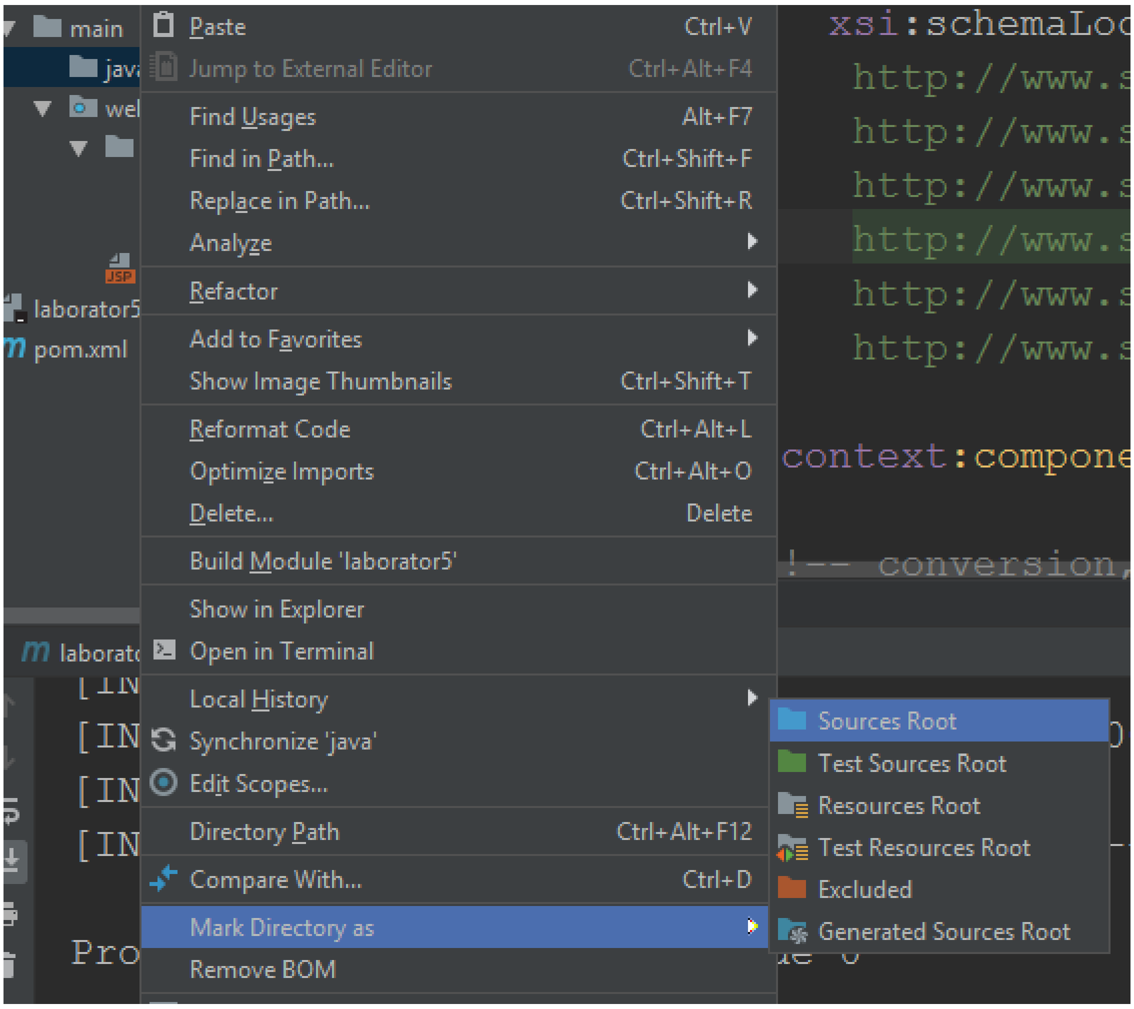
```
<properties>
    <project.build.sourceEncoding>UTF-
8</project.build.sourceEncoding>
    <maven.compiler.source>1.7</maven.compiler.source>
    <maven.compiler.target>1.7</maven.compiler.target>
    <spring.version>5.1.3.RELEASE</spring.version>
    <servlet.version>4.0.1</servlet.version>
    <thymeleaf.version>3.0.9.RELEASE</thymeleaf.version>
    <lombok.version>1.18.2</lombok.version>
  </properties>
    <dependency>
      <groupId>org.springframework</groupId>
      <artifactId>spring-webmvc</artifactId>
      <version>${spring.version}</version>
    </dependency>
    <dependency>
      <groupId>javax.servlet</groupId>
      <artifactId>javax.servlet-api</artifactId>
      <version>${servlet.version}</version>
    </dependency>
    <dependency>
      <groupId>org.thymeleaf</groupId>
      <artifactId>thymeleaf</artifactId>
      <version>${thymeleaf.version}</version>
    </dependency>
    <dependency>
      <groupId>org.thymeleaf</groupId>
      <artifactId>thymeleaf-spring5</artifactId>
      <version>${thymeleaf.version}</version>
    </dependency>
    <dependency>
      <groupId>org.projectlombok</groupId>
      <artifactId>lombok</artifactId>
      <version>${lombok.version}</version>
    </dependency>
```

3. Completați fișierul web.xml și spring-mvc-servlet.xml.

```
<web-app>
  <display-name>Archetype Created Web Application</display-name>

  <servlet>
    <servlet-name>dispatcher</servlet-name>
    <servlet-
class>org.springframework.web.servlet.DispatcherServlet</servlet-
class>
    <init-param>
      <param-name>contextConfigLocation</param-name>
      <param-value>/WEB-INF/spring-mvc-servlet.xml</param-value>
    </init-param>
    <load-on-startup>1</load-on-startup>
  </servlet>


  <servlet-mapping>
    <servlet-name>dispatcher</servlet-name>
    <url-pattern>/</url-pattern>
  </servlet-mapping>
</web-app>
```

```
<bean
      class="org.springframework.web.servlet.view.InternalResourc
eViewResolver">
      <property name="prefix" value="/WEB-INF/view/" />
      <property name="suffix" value=".jsp" />
</bean>
```

4. În directorul main creați un nou director denumit java. Folosiți pentru acest director opțiunea mark directory as Sources root.



5. Adăugați pachetul com.apbdoo.lab5.controllers
6. În pachetul com.apbdoo.lab5 creați clasa com.apbdoo.lab5.controllers.IndexController.
7. Adnotați clasa @Controller

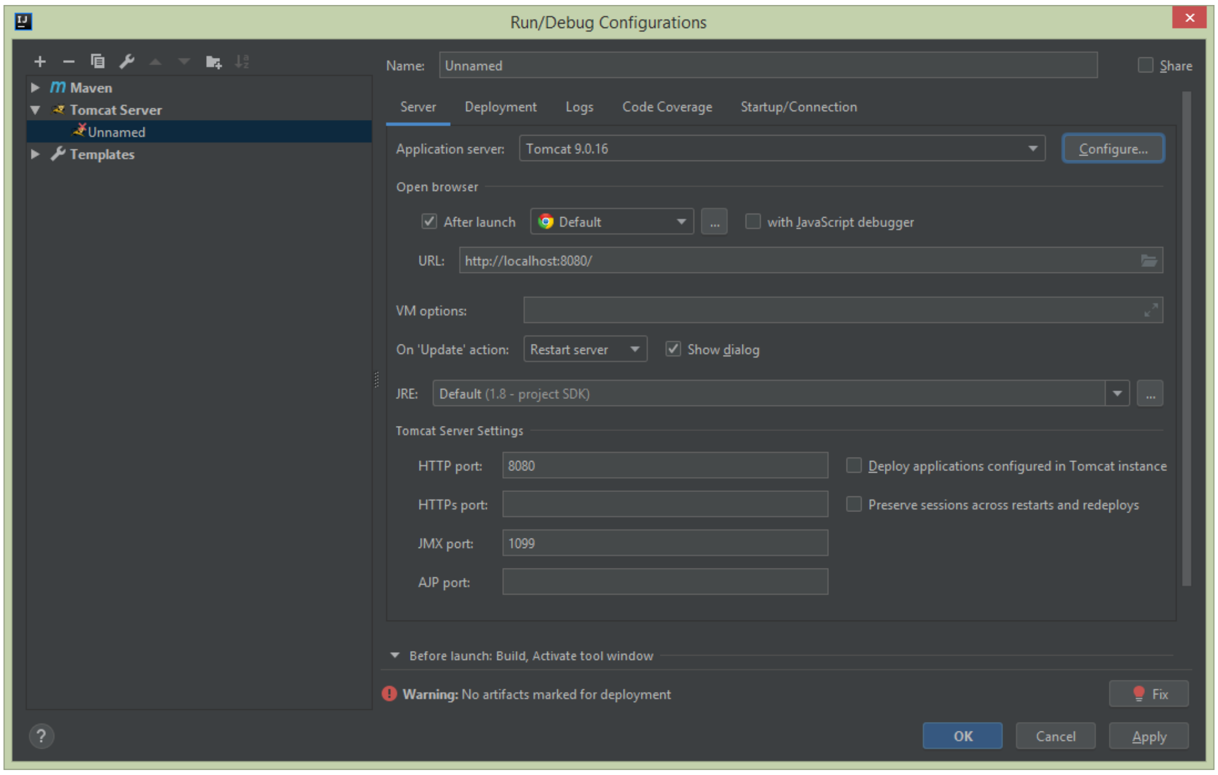
```
package com.apbdoo.lab5.controllers;

import org.springframework.stereotype.Controller;

@Controller
public class IndexController {

}
```

8. Adăugați o metodă adnotată @RequestMapping

```
@RequestMapping("/")
public String showIndex(){
      return "index";
}
```

9. Creați directoru WEB-INF/VIEWS și copiați în acest director fișierul index.jsp.

10. Adăugați o configurație pentru a rula aplicația pe un server Tomcat


Run/Debug Configurations

Maven
Tomcat Server
Unnamed
Templates

Name: Unnamed
Share

Server | Deployment | Logs | Code Coverage | Startup/Connection

Application server: Tomcat 9.0.16 Configure...

Open browser
☑ After launch Default ... ☐ with JavaScript debugger
URL: http://localhost:8080/

VM options:

On 'Update' action: Restart server ☑ Show dialog

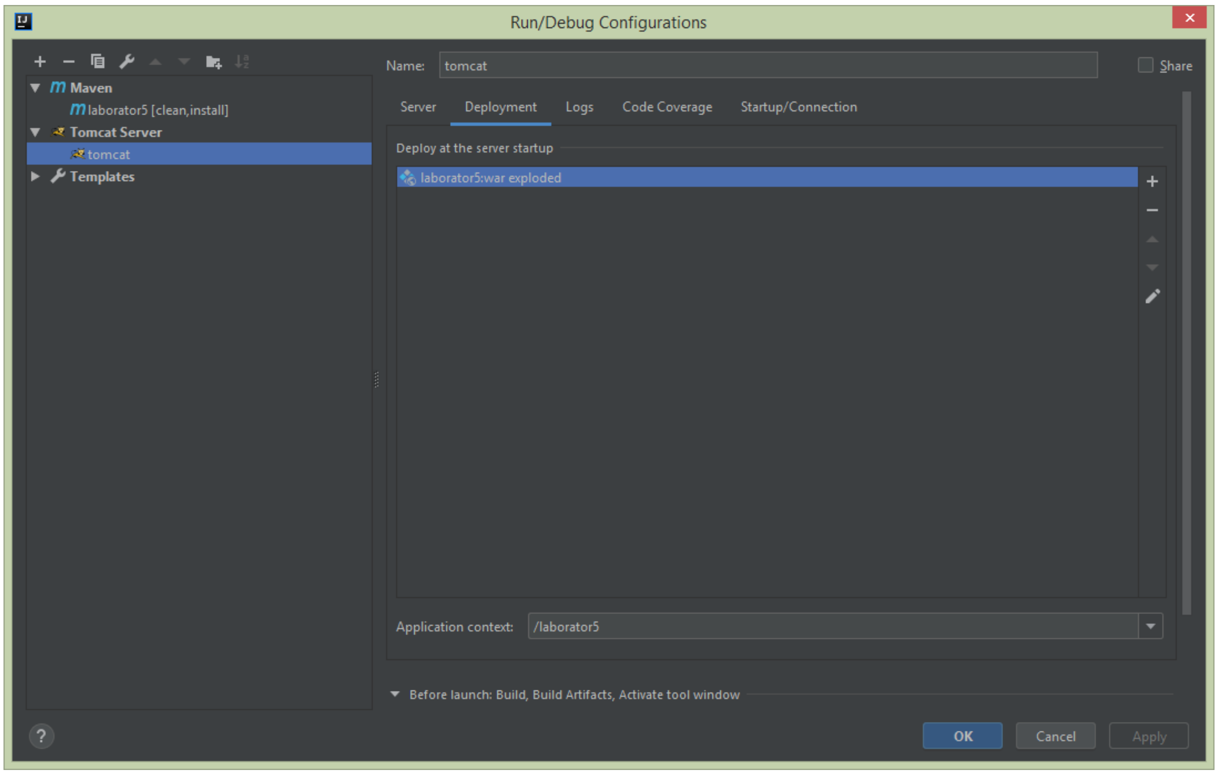
JRE: Default (1.8 - project SDK)

Tomcat Server Settings
HTTP port: 8080 ☐ Deploy applications configured in Tomcat instance
HTTPs port: ☐ Preserve sessions across restarts and redeploys
JMX port: 1099
AJP port:

Before launch: Build, Activate tool window

Warning: No artifacts marked for deployment Fix

OK Cancel Apply


Run/Debug Configurations

Maven
laborator5 [clean,install]
Tomcat Server
tomcat
Templates

Name: tomcat
Share

Server | Deployment | Logs | Code Coverage | Startup/Connection

Deploy at the server startup
laborator5:war exploded

Application context: /laborator5

Before launch: Build, Build Artifacts, Activate tool window

OK Cancel Apply

11. Testați într-un browser adresa

```
http://localhost:8080/laborator5/
```

12. Adăugți în fișierul web.xml configurarea bean-ului org.thymeleaf.spring5.SpringTemplateEngine.

```
<bean class="org.thymeleaf.spring5.view.ThymeleafViewResolver">
        <property name="characterEncoding" value="UTF-8" />
        <property name="templateEngine">
            <bean
class="org.thymeleaf.spring5.SpringTemplateEngine">
                <property name="dialects">
                    <set>
                        <bean
class="org.thymeleaf.spring5.dialect.SpringStandardDialect" />
                    </set>
                </property>
                <property name="templateResolvers">
                    <set>
<bean
class="org.thymeleaf.templateresolver.ServletContextTemplateResol
ver">
               <constructor-arg ref="servletContext"/>
               <property name="cacheable" value="false" />
               <property name="prefix" value="/WEB-INF/views/" />
               <property name="suffix" value=".html" />
               <property name="templateMode" value="HTML5" />
                        </bean>
                    </set>
                </property>
            </bean>
        </property>
    </bean>
```

13. Adăugați pachetul model și clasa Hello cu atributul name.

```
package com.apbdoo.lab5.model;
import lombok.Data;

@Data
public class Hello {
    private Long id;
    private String name;
}
```

14. Adăugați în directorul views templateul formName.html:

```
<!DOCTYPE HTML>
<html xmlns:th="http://www.thymeleaf.org">
<head>
    <title>Getting Started: Handling Form Submission</title>
    <meta http-equiv="Content-Type" content="text/html;
charset=UTF-8" />
</head>
<body>
<h1>Form</h1>
<form action="#" th:action="@{/greeting}" th:object="${hello}"
method="post">
    <p>Id: <input type="text" th:field="*{id}" /></p>
    <p>Message: <input type="text" th:field="*{name}" /></p>
    <p><input type="submit" value="Submit" /> <input type="reset"
value="Reset" /></p>
</form>
<form action="#" th:action="@{/hello}"  method="post">
    <p>Id: <input type="text" name="name" /></p>
    <p><input type="submit" value="Submit" /> <input type="reset"
value="Reset" /></p>
</form>
</body>
</html>
```

15. În clasa IndexController adăugați metoda:

```
@RequestMapping("/showForm")
public String showInputForm(Model model){
        model.addAttribute("hello",new Hello());
        return "formName";
}
```

16. Testați în browser:

```
http://localhost:8080/laborator5/showForm
```

17. Adăugați în IndexController metoda

```
@PostMapping("/greeting")
public String greetingSubmit(@ModelAttribute Hello hello) {
        return "result";
}
```

18. Adăugați în directorul views templatul result:

```
<!DOCTYPE HTML>
<html xmlns:th="http://www.thymeleaf.org">
<head>
   <title>Getting Started: Handling Form Submission</title>
   <meta http-equiv="Content-Type" content="text/html; charset=UTF-8" />
</head>
<body>
<h1>Result</h1>
<p th:text="'id: ' + ${hello.id}" />
<p th:text="'content: ' + ${hello.name}" />
<a th:href="@{'/showForm'}" >Submit another message</a>
</body>
</html>
```

19. Adăugați în pagina home un link către formular.

20. Adăugați un nou endpoint: hello/name=Word.

```
@RequestMapping(value="/hello", method = RequestMethod.GET )
@ResponseBody
public String showHelloWord(@RequestParam String name){

       return "Hello " + name;
}
```

21. Testați fără parametru name și http://localhost:8080/laborator5/hello și adăugați required = false.

```
@RequestMapping(value="/hello", method = RequestMethod.GET )
@ResponseBody
public String showHelloWord(@RequestParam(required = false) String name){

       return "Hello " + name;
}
```

22. Adaugați clasa HelloController. De ce este necesară adnotarea @RequestMapping?

```
@Controller
@RequestMapping("hello")
public class HelloController {
@RequestMapping(value="/hello/{name}", method = RequestMethod.GET )
@ResponseBody
public String showHelloWord(@PathVariable String name){
        return "hello " + name;
}

@RequestMapping("/showForm")
public String showInputForm(Model model){
        model.addAttribute("hello",new Hello());
        return "formName";
    }
}
```