Complete Code Example

/pom.xml

```
<?xml version="1.0" encoding="UTF-8"?>
<project xmlns="http://maven.apache.org/POM/4.0.0" xmlns:xsi="http://www.w3.org/2001/XMLSch
        xsi:schemaLocation="http://maven.apache.org/POM/4.0.0 https://maven.apache.org/xsd/
        <modelVersion>4.0.0</modelVersion>
        <parent>
                <groupId>org.springframework.boot</groupId>
                <artifactId>spring-boot-starter-parent</artifactId>
                <version>3.0.0-M3</version>
                <relativePath/> <!-- lookup parent from repository -->
        </parent>
        <groupId>com.in28minutes.springboot</groupId>
        <artifactId>myfirstwebapp</artifactId>
        <version>0.0.1-SNAPSHOT</version>
        <name>myfirstwebapp</name>
        <description>Demo project for Spring Boot</description>
        <properties>
                <java.version>18</java.version>
        </properties>
        <dependencies>

                <dependency>
                        <groupId>org.springframework.boot</groupId>
                        <artifactId>spring-boot-starter-web</artifactId>
                </dependency>

                <dependency>
                        <groupId>org.springframework.boot</groupId>
                        <artifactId>spring-boot-starter-validation</artifactId>
                </dependency>
                <dependency>
                        <groupId>org.apache.tomcat.embed</groupId>
                        <artifactId>tomcat-embed-jasper</artifactId>
                        <scope>provided</scope>
                </dependency>

                <dependency>
                        <groupId>jakarta.servlet.jsp.jstl</groupId>
                        <artifactId>jakarta.servlet.jsp.jstl-api</artifactId>
                </dependency>

                <dependency>
                        <groupId>org.eclipse.jetty</groupId>
                        <artifactId>glassfish-jstl</artifactId>
                </dependency>

                <dependency>
                        <groupId>org.webjars</groupId>
                        <artifactId>bootstrap</artifactId>
                        <version>5.1.3</version>
                </dependency>

                <dependency>
                        <groupId>org.webjars</groupId>
                        <artifactId>jquery</artifactId>
                        <version>3.6.0</version>
                </dependency>

                <dependency>
                        <groupId>org.webjars</groupId>
                        <artifactId>bootstrap-datepicker</artifactId>
                        <version>1.9.0</version>
                </dependency>


                <dependency>
                        <groupId>org.springframework.boot</groupId>
                        <artifactId>spring-boot-devtools</artifactId>
                        <scope>runtime</scope>
                        <optional>true</optional>
                </dependency>
                <dependency>
                        <groupId>org.springframework.boot</groupId>
                        <artifactId>spring-boot-starter-test</artifactId>
                        <scope>test</scope>
                </dependency>
        </dependencies>

        <build>
                <plugins>
                        <plugin>
                                <groupId>org.springframework.boot</groupId>
                                <artifactId>spring-boot-maven-plugin</artifactId>
                        </plugin>
                </plugins>
        </build>
        <repositories>
                <repository>
                        <id>spring-milestones</id>
                        <name>Spring Milestones</name>
                        <url>https://repo.spring.io/milestone</url>
                        <snapshots>
                                <enabled>false</enabled>
                        </snapshots>
                </repository>
        </repositories>
        <pluginRepositories>
                <pluginRepository>
                        <id>spring-milestones</id>
                        <name>Spring Milestones</name>
                        <url>https://repo.spring.io/milestone</url>
                        <snapshots>
                                <enabled>false</enabled>
                        </snapshots>
                </pluginRepository>
        </pluginRepositories>

</project>
```

/src/main/java/com/in28minutes/springboot/myfirstwebapp/MyfirstwebappApplication.java

```
package com.in28minutes.springboot.myfirstwebapp;

import org.springframework.boot.SpringApplication;
import org.springframework.boot.autoconfigure.SpringBootApplication;

@SpringBootApplication
public class MyfirstwebappApplication {

        public static void main(String[] args) {
                SpringApplication.run(MyfirstwebappApplication.class, args);
        }

}
```

/src/main/java/com/in28minutes/springboot/myfirstwebapp/hello/SayHelloController.java

```
package com.in28minutes.springboot.myfirstwebapp.hello;

import org.springframework.stereotype.Controller;
import org.springframework.web.bind.annotation.RequestMapping;
import org.springframework.web.bind.annotation.ResponseBody;

@Controller
public class SayHelloController {

        //"say-hello" => "Hello! What are you learning today?"

        //say-hello
        // http://localhost:8080/say-hello
        @RequestMapping("say-hello")
        @ResponseBody
        public String sayHello() {
                return "Hello! What are you learning today?";
        }

        @RequestMapping("say-hello-html")
        @ResponseBody
        public String sayHelloHtml() {
                StringBuffer sb = new StringBuffer();
                sb.append("<html>");
                sb.append("<head>");
                sb.append("<title> My First HTML Page - Changed</title>");
                sb.append("</head>");
                sb.append("<body>");
                sb.append("My first html page with body - Changed");
                sb.append("</body>");
                sb.append("</html>");

                return sb.toString();
        }

        //
        // "say-hello-jsp" => sayHello.jsp
        // /src/main/resources/META-INF/resources/WEB-INF/jsp/sayHello.jsp
        // /src/main/resources/META-INF/resources/WEB-INF/jsp/welcome.jsp
        // /src/main/resources/META-INF/resources/WEB-INF/jsp/login.jsp
        // /src/main/resources/META-INF/resources/WEB-INF/jsp/todos.jsp
        @RequestMapping("say-hello-jsp")
        public String sayHelloJsp() {
                return "sayHello";
        }
}
```

/src/main/java/com/in28minutes/springboot/myfirstwebapp/login/WelcomeController.java

```
package com.in28minutes.springboot.myfirstwebapp.login;

import org.springframework.stereotype.Controller;
import org.springframework.ui.ModelMap;
import org.springframework.web.bind.annotation.RequestMapping;
import org.springframework.web.bind.annotation.RequestMethod;
import org.springframework.web.bind.annotation.SessionAttributes;

@Controller
@SessionAttributes("name")
public class WelcomeController {

        @RequestMapping(value="/",method = RequestMethod.GET)
        public String gotoWelcomePage(ModelMap model) {
                model.put("name", "in28minutes");
                return "welcome";
        }
}
```

/src/main/java/com/in28minutes/springboot/myfirstwebapp/todo/Todo.java

```
package com.in28minutes.springboot.myfirstwebapp.todo;

import java.time.LocalDate;

import jakarta.validation.constraints.Size;

//Database (MySQL)
//Static List of todos => Database (H2, MySQL)

public class Todo {

        public Todo(int id, String username, String description, LocalDate targetDate, bool
                super();
                this.id = id;
                this.username = username;
                this.description = description;
                this.targetDate = targetDate;
                this.done = done;
        }

        private int id;
        private String username;

        @Size(min=10, message="Enter atleast 10 characters")
        private String description;
        private LocalDate targetDate;
        private boolean done;

        public int getId() {
                return id;
        }

        public void setId(int id) {
                this.id = id;
        }

        public String getUsername() {
                return username;
        }

        public void setUsername(String username) {
                this.username = username;
        }

        public String getDescription() {
                return description;
        }

        public void setDescription(String description) {
                this.description = description;
        }

        public LocalDate getTargetDate() {
                return targetDate;
        }

        public void setTargetDate(LocalDate targetDate) {
                this.targetDate = targetDate;
        }

        public boolean isDone() {
                return done;
        }

        public void setDone(boolean done) {
                this.done = done;
        }

        @Override
        public String toString() {
                return "Todo [id=" + id + ", username=" + username + ", description=" + des
                                + targetDate + ", done=" + done + "]";
        }

}
```

/src/main/java/com/in28minutes/springboot/myfirstwebapp/todo/TodoController.java

```
package com.in28minutes.springboot.myfirstwebapp.todo;

import java.time.LocalDate;
import java.util.List;

import org.springframework.stereotype.Controller;
import org.springframework.ui.ModelMap;
import org.springframework.validation.BindingResult;
import org.springframework.web.bind.annotation.RequestMapping;
import org.springframework.web.bind.annotation.RequestMethod;
import org.springframework.web.bind.annotation.RequestParam;
import org.springframework.web.bind.annotation.SessionAttributes;

import jakarta.validation.Valid;

@Controller
@SessionAttributes("name")
public class TodoController {

        public TodoController(TodoService todoService) {
                super();
                this.todoService = todoService;
        }

        private TodoService todoService;


        @RequestMapping("list-todos")
        public String listAllTodos(ModelMap model) {
                List<Todo> todos = todoService.findByUsername("in28minutes");
                model.addAttribute("todos", todos);

                return "listTodos";
        }

        //GET, POST
        @RequestMapping(value="add-todo", method = RequestMethod.GET)
        public String showNewTodoPage(ModelMap model) {
                String username = (String)model.get("name");
                Todo todo = new Todo(0, username, "", LocalDate.now().plusYears(1), false);
                model.put("todo", todo);
                return "todo";
        }

        @RequestMapping(value="add-todo", method = RequestMethod.POST)
        public String addNewTodo(ModelMap model, @Valid Todo todo, BindingResult result) {

                if(result.hasErrors()) {
                        return "todo";
                }

                String username = (String)model.get("name");
                todoService.addTodo(username, todo.getDescription(),
                                LocalDate.now().plusYears(1), false);
                return "redirect:list-todos";
        }

        @RequestMapping("delete-todo")
        public String deleteTodo(@RequestParam int id) {
                //Delete todo

                todoService.deleteById(id);
                return "redirect:list-todos";

        }
        @RequestMapping(value="update-todo", method = RequestMethod.GET)
        public String showUpdateTodoPage(@RequestParam int id, ModelMap model) {
                Todo todo = todoService.findById(id);
                model.addAttribute("todo", todo);
                return "todo";
        }

        @RequestMapping(value="update-todo", method = RequestMethod.POST)
        public String updateTodo(ModelMap model, @Valid Todo todo, BindingResult result) {

                if(result.hasErrors()) {
                        return "todo";
                }

                String username = (String)model.get("name");
                todo.setUsername(username);
                todoService.updateTodo(todo);
                return "redirect:list-todos";
        }

}
```

/src/main/java/com/in28minutes/springboot/myfirstwebapp/todo/TodoService.java

```
package com.in28minutes.springboot.myfirstwebapp.todo;

import java.time.LocalDate;
import java.util.ArrayList;
import java.util.List;
import java.util.function.Predicate;

import org.springframework.stereotype.Service;

import jakarta.validation.Valid;

@Service
public class TodoService {

        private static List<Todo> todos = new ArrayList<>();

        private static int todosCount = 0;

        static {
                todos.add(new Todo(++todosCount, "in28minutes","Get AWS Certified",
                                                        LocalDate.now().plusYears(1), false
                todos.add(new Todo(++todosCount, "in28minutes","Learn DevOps",
                                LocalDate.now().plusYears(2), false ));
                todos.add(new Todo(++todosCount, "in28minutes","Learn Full Stack Developmen
                                LocalDate.now().plusYears(3), false ));
        }
        public List<Todo> findByUsername(String username){
                return todos;
        }

        public void addTodo(String username, String description, LocalDate targetDate, bool
                Todo todo = new Todo(++todosCount,username,description,targetDate,done);
                todos.add(todo);
        }

        public void deleteById(int id) {
                //todo.getId() == id
                // todo -> todo.getId() == id
                Predicate<? super Todo> predicate = todo -> todo.getId() == id;
                todos.removeIf(predicate);
        }

        public Todo findById(int id) {
                Predicate<? super Todo> predicate = todo -> todo.getId() == id;
                Todo todo = todos.stream().filter(predicate).findFirst().get();
                return todo;
        }

        public void updateTodo(@Valid Todo todo) {
                deleteById(todo.getId());
                todos.add(todo);
        }
}
```

/src/main/resources/META-INF/resources/WEB-INF/jsp/common/footer.jspf

```
<script src="webjars/bootstrap/5.1.3/js/bootstrap.min.js"></script>
                <script src="webjars/jquery/3.6.0/jquery.min.js"></script>
                <script src="webjars/bootstrap-datepicker/1.9.0/js/bootstrap-datepicker.min.js"></script>

        </body>
</html>
```

/src/main/resources/META-INF/resources/WEB-INF/jsp/common/header.jspf

```
<%@ taglib prefix="c" uri="http://java.sun.com/jsp/jstl/core" %>
<%@ taglib prefix="form" uri="http://www.springframework.org/tags/form" %>


<html>
        <head>
                <link href="webjars/bootstrap/5.1.3/css/bootstrap.min.css" rel="stylesheet" >
                <link href="webjars/bootstrap-datepicker/1.9.0/css/bootstrap-
datepicker.standalone.min.css" rel="stylesheet" >

                <title>Manage Your Todos</title>
        </head>
        <body>
```

/src/main/resources/META-INF/resources/WEB-INF/jsp/common/navigation.jspf

```
<nav class="navbar navbar-expand-md navbar-light bg-light mb-3 p-1">
        <a class="navbar-brand m-1"
href="https://courses.in28minutes.com">in28minutes</a>
        <div class="collapse navbar-collapse">
                <ul class="navbar-nav">
                        <li class="nav-item"><a class="nav-link" href="/">Home</a></li>
                        <li class="nav-item"><a class="nav-link" href="/list-
todos">Todos</a></li>
                </ul>
        </div>
        <ul class="navbar-nav">
                <li class="nav-item"><a class="nav-link" href="/logout">Logout</a></li>
        </ul>
</nav>
```

/src/main/resources/META-INF/resources/WEB-INF/jsp/listTodos.jsp

```
<%@ include file="common/header.jspf" %>
<%@ include file="common/navigation.jspf" %>
<div class="container">
        <h1>Your Todos</h1>
        <table class="table">
                <thead>
                        <tr>
                                <th>Description</th>
                                <th>Target Date</th>
                                <th>Is Done?</th>
                                <th></th>
                                <th></th>
                        </tr>
                </thead>
                <tbody>
                        <c:forEach items="${todos}" var="todo">
                                <tr>
                                        <td>${todo.description}</td>
                                        <td>${todo.targetDate}</td>
                                        <td>${todo.done}</td>
                                        <td> <a href="delete-todo?id=${todo.id}"
```

```
class="btn btn-warning">Delete</a>   </td>
                                        <td> <a href="update-todo?id=${todo.id}"
class="btn btn-success">Update</a>   </td>
                                </tr>
                        </c:forEach>
                </tbody>
        </table>
        <a href="add-todo" class="btn btn-success">Add Todo</a>
</div>

<%@ include file="common/footer.jspf" %>
```

/src/main/resources/META-INF/resources/WEB-INF/jsp/sayHello.jsp

```
<html>
        <head>
                <title> My first HTML Page - JSP</title>
        </head>
        <body>
                <h1>Heading 1</h1>
                <h2>Heading 2</h2>

                My first html page with body - JSP
        </body>
</html>
```

/src/main/resources/META-INF/resources/WEB-INF/jsp/todo.jsp

```
<%@ include file="common/header.jspf" %>
<%@ include file="common/navigation.jspf" %>

<div class="container">

        <h1>Enter Todo Details</h1>

        <form:form method="post" modelAttribute="todo">

                <fieldset class="mb-3">
                        <form:label path="description">Description</form:label>
                        <form:input type="text" path="description" required="required"/>
                        <form:errors path="description" cssClass="text-warning"/>
                </fieldset>

                <fieldset class="mb-3">
                        <form:label path="targetDate">Target Date</form:label>
                        <form:input type="text" path="targetDate" required="required"/>
                        <form:errors path="targetDate" cssClass="text-warning"/>
                </fieldset>
                <form:input type="hidden" path="id"/>

                <form:input type="hidden" path="done"/>

                <input type="submit" class="btn btn-success"/>

        </form:form>

</div>

<%@ include file="common/footer.jspf" %>

<script type="text/javascript">
        $('#targetDate').datepicker({
            format: 'yyyy-mm-dd'
        });
</script>
```

/src/main/resources/META-INF/resources/WEB-INF/jsp/welcome.jsp

```
<%@ include file="common/header.jspf" %>
<%@ include file="common/navigation.jspf" %>

<div class="container">
        <h1>Welcome ${name}</h1>
        <a href="list-todos">Manage</a> your todos
</div>

<%@ include file="common/footer.jspf" %>
```

/src/main/resources/application.properties

```
#server.port=8081
#sayHello.jsp
#/WEB-INF/jsp/sayHello.jsp

# /WEB-INF/jsp/login.jsp => View Resolver
spring.mvc.view.prefix=/WEB-INF/jsp/
spring.mvc.view.suffix=.jsp
logging.level.org.springframework=info
logging.level.com.in28minutes.springboot.myfirstwebapp=info

spring.mvc.format.date=yyyy-MM-dd
```

/src/test/java/com/in28minutes/springboot/myfirstwebapp/MyfirstwebappApplicationTests.java

```java
package com.in28minutes.springboot.myfirstwebapp;

import org.junit.jupiter.api.Test;
import org.springframework.boot.test.context.SpringBootTest;

@SpringBootTest
class MyfirstwebappApplicationTests {

	@Test
	void contextLoads() {
	}

}
```