Spring Tutorial



This spring tutorial provides in-depth concepts of Spring Framework with simplified examples. It was **developed by Rod Johnson in 2003**. Spring framework makes the easy development of JavaEE application.

It is helpful for beginners and experienced persons.

Spring Framework

Spring is a *lightweight* framework. It can be thought of as a *framework of frameworks* because it provides support to various frameworks such as [Struts](https://www.javatpoint.com/struts-2-tutorial)

, [Hibernate](https://www.javatpoint.com/hibernate-tutorial)

, Tapestry, [EJB](https://www.javatpoint.com/ejb-tutorial)

, [JSF](https://www.javatpoint.com/jsf-tutorial)

, etc. The framework, in broader sense, can be defined as a structure where we find solution of the various technical problems.

The Spring framework comprises several modules such as IOC, AOP, DAO, Context, ORM, WEB MVC etc. We will learn these modules in next page. Let's understand the IOC and Dependency Injection first.

33.5M

643

Difference between JDK, JRE, and JVM

Inversion Of Control (IOC) and Dependency Injection

These are the design patterns that are used to remove dependency from the programming code. They make the code easier to test and maintain. Let's understand this with the following code:

1. **class** Employee{
2. Address address;
3. Employee(){
4. address=**new** Address();
5. }
6. }

In such case, there is dependency between the Employee and Address (tight coupling). In the Inversion of Control scenario, we do this something like this:

1. **class** Employee{
2. Address address;
3. Employee(Address address){
4. **this**.address=address;
5. }
6. }

Thus, IOC makes the code loosely coupled. In such case, there is no need to modify the code if our logic is moved to new environment.

In Spring framework, IOC container is responsible to inject the dependency. We provide metadata to the IOC container either by XML file or annotation.

Advantage of Dependency Injection

* makes the code loosely coupled so easy to maintain
* makes the code easy to test

Advantages of Spring Framework

There are many advantages of Spring Framework. They are as follows:

1) Predefined Templates

Spring framework provides templates for JDBC, Hibernate, JPA etc. technologies. So there is no need to write too much code. It hides the basic steps of these technologies.

Let's take the example of JdbcTemplate, you don't need to write the code for exception handling, creating connection, creating statement, committing transaction, closing connection etc. You need to write the code of executing query only. Thus, it save a lot of JDBC code.

2) Loose Coupling

The Spring applications are loosely coupled because of dependency injection.

3) Easy to test

The Dependency Injection makes easier to test the application. The EJB or Struts application require server to run the application but Spring framework doesn't require server.

4) Lightweight

Spring framework is lightweight because of its POJO implementation. The Spring Framework doesn't force the programmer to inherit any class or implement any interface. That is why it is said non-invasive.

5) Fast Development

The Dependency Injection feature of Spring Framework and it support to various frameworks makes the easy development of JavaEE application.

6) Powerful abstraction

It provides powerful abstraction to JavaEE specifications such as [JMS](https://www.javatpoint.com/jms-tutorial)

, [JDBC](https://www.javatpoint.com/java-jdbc)

, JPA and JTA.

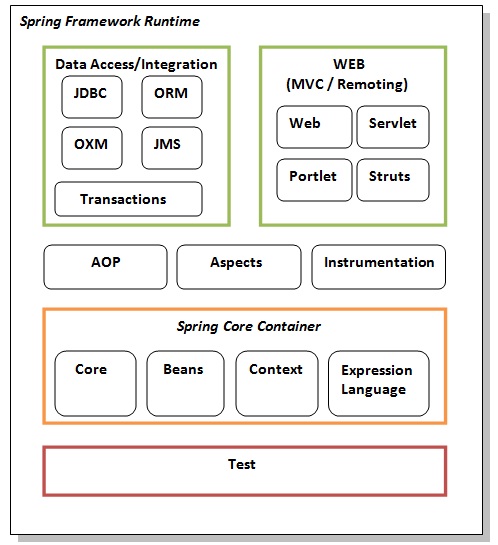
7) Declarative support

It provides declarative support for caching, validation, transactions and formatting.

Spring Modules

1. [Spring Modules](https://www.javatpoint.com/spring-modules)
2. [Test](https://www.javatpoint.com/spring-modules#sptest)
3. [Spring Core Container](https://www.javatpoint.com/spring-modules#spcore)
4. [AOP, Aspects and Instrumentation](https://www.javatpoint.com/spring-modules#spaop)
5. [Data Access / Integration](https://www.javatpoint.com/spring-modules#spda)
6. [Web](https://www.javatpoint.com/spring-modules#spweb)

The Spring framework comprises of many modules such as core, beans, context, expression language, AOP, Aspects, Instrumentation, JDBC, ORM, OXM, JMS, Transaction, Web, Servlet, Struts etc. These modules are grouped into Test, Core Container, AOP, Aspects, Instrumentation, Data Access / Integration, Web (MVC / Remoting) as displayed in the following diagram.



Test

This layer provides support of testing with JUnit and TestNG.

Spring Core Container

The Spring Core container contains core, beans, context and expression language (EL) modules.

Core and Beans

These modules provide IOC and Dependency Injection features.

OOPs Concepts in Java

Context

This module supports internationalization (I18N), EJB, JMS, Basic Remoting.

Expression Language

It is an extension to the EL defined in JSP. It provides support to setting and getting property values, method invocation, accessing collections and indexers, named variables, logical and arithmetic operators, retrieval of objects by name etc.

AOP, Aspects and Instrumentation

These modules support aspect oriented programming implementation where you can use Advices, Pointcuts etc. to decouple the code.

The aspects module provides support to integration with AspectJ.

The instrumentation module provides support to class instrumentation and classloader implementations.

Data Access / Integration

This group comprises of JDBC, ORM, OXM, JMS and Transaction modules. These modules basically provide support to interact with the database.

Web

This group comprises of Web, Web-Servlet, Web-Struts and Web-Portlet. These modules provide support to create web application.

Creating spring application in Eclipse IDE

1. [Creating spring application in Eclipse](https://www.javatpoint.com/example-of-spring-application-in-eclipse)
2. [Steps to create spring application in Eclipse](https://www.javatpoint.com/example-of-spring-application-in-eclipse#steps)

Here, we are going to create a simple application of spring framework using eclipse IDE. Let's see the simple steps to create the spring application in Eclipse IDE.

* **create the java project**
* **add spring jar files**
* **create the class**
* **create the xml file to provide the values**
* **create the test class**

Steps to create spring application in Eclipse IDE

Let's see the 5 steps to create the first spring application using eclipse IDE.

1) Create the Java Project

Go to **File** menu - **New** - **project** - **Java Project**. Write the project name e.g. firstspring - **Finish**. Now the java project is created.

2) Add spring jar files

There are mainly three jar files required to run this application.

Prime Ministers of India | List of Prime Minister of India (1947-2020)

* **org.springframework.core-3.0.1.RELEASE-A**
* **com.springsource.org.apache.commons.logging-1.1.1**
* **org.springframework.beans-3.0.1.RELEASE-A**

For the future use, You can download the required jar files for spring core application.

[download the core jar files for spring](https://static.javatpoint.com/src/sp/spcorejars.zip)

[download the all jar files for spring including aop, mvc, j2ee, remoting, oxm, etc.](https://static.javatpoint.com/src/sp/springjars.zip)

To run this example, you need to load only spring core jar files.

To load the jar files in eclipse IDE, **Right click on your project** - **Build Path** - **Add external archives** - **select all the required jar files** - **finish.**.

3) Create Java class

In such case, we are simply creating the Student class have name property. The name of the student will be provided by the xml file. It is just a simple example not the actual use of spring. We will see the actual use in Dependency Injection chapter. To create the java class, **Right click on src** - **New**- **class** - **Write the class name e.g. Student** - **finish**. Write the following code:

1. **package** com.javatpoint;
3. **public** **class** Student {
4. **private** String name;
6. **public** String getName() {
7. **return** name;
8. }
10. **public** **void** setName(String name) {
11. **this**.name = name;
12. }
14. **public** **void** displayInfo(){
15. System.out.println("Hello: "+name);
16. }
17. }

This is simple bean class, containing only one property name with its getters and setters method. This class contains one extra method named displayInfo() that prints the student name by the hello message.

4) Create the xml file

To create the xml file click on src - new - file - give the file name such as applicationContext.xml - finish. Open the applicationContext.xml file, and write the following code:

1. <?xml version="1.0" encoding="UTF-8"?>
2. <beans
3. xmlns="http://www.springframework.org/schema/beans"
4. xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
5. xmlns:p="http://www.springframework.org/schema/p"
6. xsi:schemaLocation="http://www.springframework.org/schema/beans
7. http://www.springframework.org/schema/beans/spring-beans-3.0.xsd">
9. <bean id="studentbean" **class**="com.javatpoint.Student">
10. <property name="name" value="Vimal Jaiswal"></property>
11. </bean>
13. </beans>

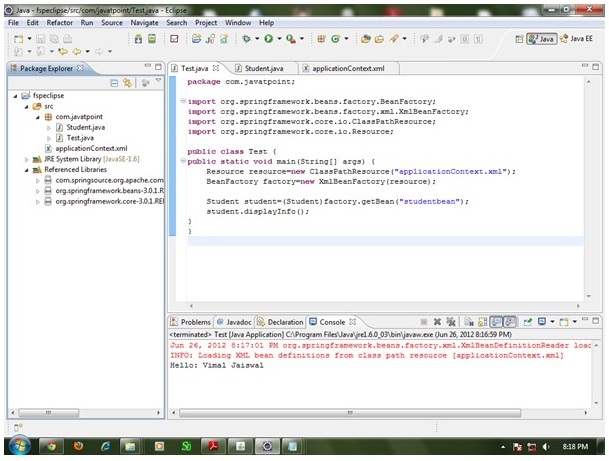
The **bean** element is used to define the bean for the given class. The **property** subelement of bean specifies the property of the Student class named name. The value specified in the property element will be set in the Student class object by the IOC container.

5) Create the test class

Create the java class e.g. Test. Here we are getting the object of Student class from the IOC container using the getBean() method of BeanFactory. Let's see the code of test class.

1. **package** com.javatpoint;
3. **import** org.springframework.beans.factory.BeanFactory;
4. **import** org.springframework.beans.factory.xml.XmlBeanFactory;
5. **import** org.springframework.core.io.ClassPathResource;
6. **import** org.springframework.core.io.Resource;
8. **public** **class** Test {
9. **public** **static** **void** main(String[] args) {
10. Resource resource=**new** ClassPathResource("applicationContext.xml");
11. BeanFactory factory=**new** XmlBeanFactory(resource);
13. Student student=(Student)factory.getBean("studentbean");
14. student.displayInfo();
15. }
16. }

Now run this class. You will get the output Hello: Vimal Jaiswal.



IoC Container

1. [IoC Container](https://www.javatpoint.com/ioc-container)
2. [Using BeanFactory](https://www.javatpoint.com/ioc-container)
3. [Using ApplicationContext](https://www.javatpoint.com/ioc-container)

The IoC container is responsible to instantiate, configure and assemble the objects. The IoC container gets informations from the XML file and works accordingly. The main tasks performed by IoC container are:

* to instantiate the application class
* to configure the object
* to assemble the dependencies between the objects

There are two types of IoC containers. They are:

1. **BeanFactory**
2. **ApplicationContext**

Difference between BeanFactory and the ApplicationContext

The org.springframework.beans.factory.**BeanFactory** and the org.springframework.context.**ApplicationContext** interfaces acts as the IoC container. The ApplicationContext interface is built on top of the BeanFactory interface. It adds some extra functionality than BeanFactory such as simple integration with Spring's AOP, message resource handling (for I18N), event propagation, application layer specific context (e.g. WebApplicationContext) for web application. So it is better to use ApplicationContext than BeanFactory.

Using BeanFactory

The XmlBeanFactory is the implementation class for the BeanFactory interface. To use the BeanFactory, we need to create the instance of XmlBeanFactory class as given below:

Difference between JDK, JRE, and JVM

1. Resource resource=**new** ClassPathResource("applicationContext.xml");
2. BeanFactory factory=**new** XmlBeanFactory(resource);

The constructor of XmlBeanFactory class receives the Resource object so we need to pass the resource object to create the object of BeanFactory.

Using ApplicationContext

The ClassPathXmlApplicationContext class is the implementation class of ApplicationContext interface. We need to instantiate the ClassPathXmlApplicationContext class to use the ApplicationContext as given below:

1. ApplicationContext context =
2. **new** ClassPathXmlApplicationContext("applicationContext.xml");

The constructor of ClassPathXmlApplicationContext class receives string, so we can pass the name of the xml file to create the instance of ApplicationContext.

Dependency Injection in Spring

1. [Dependency Injection in Spring](https://www.javatpoint.com/dependency-injection-in-spring)
2. [Dependency Lookup](https://www.javatpoint.com/dependency-injection-in-spring#dl)
3. [Dependency Injection](https://www.javatpoint.com/dependency-injection-in-spring#di)

Dependency Injection (DI) is a design pattern that removes the dependency from the programming code so that it can be easy to manage and test the application. Dependency Injection makes our programming code loosely coupled. To understand the DI better, Let's understand the Dependency Lookup (DL) first:

Dependency Lookup

The Dependency Lookup is an approach where we get the resource after demand. There can be various ways to get the resource for example:

1. A obj = **new** AImpl();

In such way, we get the resource(instance of A class) directly by new keyword. Another way is factory method:

1. A obj = A.getA();

This way, we get the resource (instance of A class) by calling the static factory method getA().

HTML Tutorial

Alternatively, we can get the resource by JNDI (Java Naming Directory Interface) as:

1. Context ctx = **new** InitialContext();
2. Context environmentCtx = (Context) ctx.lookup("java:comp/env");
3. A obj = (A)environmentCtx.lookup("A");

There can be various ways to get the resource to obtain the resource. Let's see the problem in this approach.

Problems of Dependency Lookup

There are mainly two problems of dependency lookup.

* **tight coupling** The dependency lookup approach makes the code tightly coupled. If resource is changed, we need to perform a lot of modification in the code.
* **Not easy for testing** This approach creates a lot of problems while testing the application especially in black box testing.

Dependency Injection

The Dependency Injection is a design pattern that removes the dependency of the programs. In such case we provide the information from the external source such as XML file. It makes our code loosely coupled and easier for testing. In such case we write the code as:

1. **class** Employee{
2. Address address;
4. Employee(Address address){
5. **this**.address=address;
6. }
7. **public** **void** setAddress(Address address){
8. **this**.address=address;
9. }
11. }

In such case, instance of Address class is provided by external souce such as XML file either by constructor or setter method.

Two ways to perform Dependency Injection in Spring framework

Spring framework provides two ways to inject dependency

* By Constructor
* By Setter method

Upcoming topics in Spring Dependency Injection

[Dependency Injection by constructor](https://www.javatpoint.com/spring-tutorial-dependency-injection-by-constructor)

Let's see how we can inject dependency by constructor.

[Dependency Injection by setter method](https://www.javatpoint.com/spring-tutorial-dependency-injection-by-setter-method)

Let's see how we can inject dependency by setter method.

Dependency Injection by Constructor Example

1. [Dependency Injection by constructor](https://www.javatpoint.com/spring-tutorial-dependency-injection-by-constructor)
2. [Injecting primitive and string-based values](https://www.javatpoint.com/spring-tutorial-dependency-injection-by-constructor#ips)

We can inject the dependency by constructor. The **<constructor-arg>** subelement of **<bean>** is used for constructor injection. Here we are going to inject

1. primitive and String-based values
2. Dependent object (contained object)
3. Collection values etc.

Injecting primitive and string-based values

Let's see the simple example to inject primitive and string-based values. We have created three files here:

* Employee.java
* applicationContext.xml
* Test.java

**Employee.java**

It is a simple class containing two fields id and name. There are four constructors and one method in this class.

1. **package** com.javatpoint;
3. **public** **class** Employee {
4. **private** **int** id;
5. **private** String name;
7. **public** Employee() {System.out.println("def cons");}
9. **public** Employee(**int** id) {**this**.id = id;}
11. **public** Employee(String name) {  **this**.name = name;}
13. **public** Employee(**int** id, String name) {
14. **this**.id = id;
15. **this**.name = name;
16. }
18. **void** show(){
19. System.out.println(id+" "+name);
20. }
22. }

Prime Ministers of India | List of Prime Minister of India (1947-2020)

**applicationContext.xml**

We are providing the information into the bean by this file. The constructor-arg element invokes the constructor. In such case, parameterized constructor of int type will be invoked. The value attribute of constructor-arg element will assign the specified value. The type attribute specifies that int parameter constructor will be invoked.

1. <?xml version="1.0" encoding="UTF-8"?>
2. <beans
3. xmlns="http://www.springframework.org/schema/beans"
4. xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
5. xmlns:p="http://www.springframework.org/schema/p"
6. xsi:schemaLocation="http://www.springframework.org/schema/beans
7. http://www.springframework.org/schema/beans/spring-beans-3.0.xsd">
9. <bean id="e" **class**="com.javatpoint.Employee">
10. <constructor-arg value="10" type="int"></constructor-arg>
11. </bean>
13. </beans>

**Test.java**

This class gets the bean from the applicationContext.xml file and calls the show method.

1. **package** com.javatpoint;
3. **import** org.springframework.beans.factory.BeanFactory;
4. **import** org.springframework.beans.factory.xml.XmlBeanFactory;
5. **import** org.springframework.core.io.\*;
7. **public** **class** Test {
8. **public** **static** **void** main(String[] args) {
10. Resource r=**new** ClassPathResource("applicationContext.xml");
11. BeanFactory factory=**new** XmlBeanFactory(r);
13. Employee s=(Employee)factory.getBean("e");
14. s.show();
16. }
17. }

**Output:**10 null

[download this example](https://static.javatpoint.com/src/sp/dicons.zip)

Injecting string-based values

If you don't specify the type attribute in the constructor-arg element, by default string type constructor will be invoked.

1. ....
2. <bean id="e" **class**="com.javatpoint.Employee">
3. <constructor-arg value="10"></constructor-arg>
4. </bean>
5. ....

If you change the bean element as given above, string parameter constructor will be invoked and the output will be 0 10.

**Output:**0 10

You may also pass the string literal as following:

1. ....
2. <bean id="e" **class**="com.javatpoint.Employee">
3. <constructor-arg value="Sonoo"></constructor-arg>
4. </bean>
5. ....

**Output:**0 Sonoo

You may pass integer literal and string both as following

1. ....
2. <bean id="e" **class**="com.javatpoint.Employee">
3. <constructor-arg value="10" type="int" ></constructor-arg>
4. <constructor-arg value="Sonoo"></constructor-arg>
5. </bean>
6. ....

**Output:**10 Sonoo

Constructor Injection with Dependent Object

1. [Constructor Injection with Dependent Object](https://www.javatpoint.com/spring-tutorial-constructor-injection-with-dependent-object)

If there is HAS-A relationship between the classes, we create the instance of dependent object (contained object) first then pass it as an argument of the main class constructor. Here, our scenario is Employee HAS-A Address. The Address class object will be termed as the dependent object. Let's see the Address class first:

**Address.java**

This class contains three properties, one constructor and toString() method to return the values of these object.

1. **package** com.javatpoint;
3. **public** **class** Address {
4. **private** String city;
5. **private** String state;
6. **private** String country;
8. **public** Address(String city, String state, String country) {
9. **super**();
10. **this**.city = city;
11. **this**.state = state;
12. **this**.country = country;
13. }
15. **public** String toString(){
16. **return** city+" "+state+" "+country;
17. }
18. }

**Employee.java**

It contains three properties id, name and address(dependent object) ,two constructors and show() method to show the records of the current object including the depedent object.

1. **package** com.javatpoint;
3. **public** **class** Employee {
4. **private** **int** id;
5. **private** String name;
6. **private** Address address;//Aggregation
8. **public** Employee() {System.out.println("def cons");}
10. **public** Employee(**int** id, String name, Address address) {
11. **super**();
12. **this**.id = id;
13. **this**.name = name;
14. **this**.address = address;
15. }
17. **void** show(){
18. System.out.println(id+" "+name);
19. System.out.println(address.toString());
20. }
22. }

**applicationContext.xml**

The **ref** attribute is used to define the reference of another object, such way we are passing the dependent object as an constructor argument.

Exception Handling in Java - Javatpoint

1. <?xml version="1.0" encoding="UTF-8"?>
2. <beans
3. xmlns="http://www.springframework.org/schema/beans"
4. xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
5. xmlns:p="http://www.springframework.org/schema/p"
6. xsi:schemaLocation="http://www.springframework.org/schema/beans
7. http://www.springframework.org/schema/beans/spring-beans-3.0.xsd">
9. <bean id="a1" **class**="com.javatpoint.Address">
10. <constructor-arg value="ghaziabad"></constructor-arg>
11. <constructor-arg value="UP"></constructor-arg>
12. <constructor-arg value="India"></constructor-arg>
13. </bean>

16. <bean id="e" **class**="com.javatpoint.Employee">
17. <constructor-arg value="12" type="int"></constructor-arg>
18. <constructor-arg value="Sonoo"></constructor-arg>
19. <constructor-arg>
20. <ref bean="a1"/>
21. </constructor-arg>
22. </bean>
24. </beans>

**Test.java**

This class gets the bean from the applicationContext.xml file and calls the show method.

1. **package** com.javatpoint;
3. **import** org.springframework.beans.factory.BeanFactory;
4. **import** org.springframework.beans.factory.xml.XmlBeanFactory;
5. **import** org.springframework.core.io.\*;
7. **public** **class** Test {
8. **public** **static** **void** main(String[] args) {
10. Resource r=**new** ClassPathResource("applicationContext.xml");
11. BeanFactory factory=**new** XmlBeanFactory(r);
13. Employee s=(Employee)factory.getBean("e");
14. s.show();
16. }
17. }

Constructor Injection with Collection Example

1. [Constructor Injection with Collection](https://www.javatpoint.com/spring-tutorial-constructor-injection-with-collection)

We can inject collection values by constructor in spring framework. There can be used three elements inside the **constructor-arg** element.

It can be:

1. **list**
2. **set**
3. **map**

Each collection can have string based and non-string based values.

In this example, we are taking the example of Forum where **One question can have multiple answers**. There are three pages:

1. **Question.java**
2. **applicationContext.xml**
3. **Test.java**

In this example, we are using list that can have duplicate elements, you may use set that have only unique elements. But, you need to change list to set in the applicationContext.xml file and List to Set in the Question.java file.

**Question.java**

This class contains three properties, two constructors and displayInfo() method that prints the information. Here, we are using List to contain the multiple answers.

Java Try Catch

1. **package** com.javatpoint;
3. **import** java.util.Iterator;
4. **import** java.util.List;
6. **public** **class** Question {
7. **private** **int** id;
8. **private** String name;
9. **private** List<String> answers;
11. **public** Question() {}
12. **public** Question(**int** id, String name, List<String> answers) {
13. **super**();
14. **this**.id = id;
15. **this**.name = name;
16. **this**.answers = answers;
17. }
19. **public** **void** displayInfo(){
20. System.out.println(id+" "+name);
21. System.out.println("answers are:");
22. Iterator<String> itr=answers.iterator();
23. **while**(itr.hasNext()){
24. System.out.println(itr.next());
25. }
26. }
28. }

**applicationContext.xml**

The list element of constructor-arg is used here to define the list.

1. <?xml version="1.0" encoding="UTF-8"?>
2. <beans
3. xmlns="http://www.springframework.org/schema/beans"
4. xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
5. xmlns:p="http://www.springframework.org/schema/p"
6. xsi:schemaLocation="http://www.springframework.org/schema/beans
7. http://www.springframework.org/schema/beans/spring-beans-3.0.xsd">
9. <bean id="q" **class**="com.javatpoint.Question">
10. <constructor-arg value="111"></constructor-arg>
11. <constructor-arg value="What is java?"></constructor-arg>
12. <constructor-arg>
13. <list>
14. <value>Java is a programming language</value>
15. <value>Java is a Platform</value>
16. <value>Java is an Island of Indonasia</value>
17. </list>
18. </constructor-arg>
19. </bean>
21. </beans>

**Test.java**

This class gets the bean from the applicationContext.xml file and calls the displayInfo method.

1. **package** com.javatpoint;
3. **import** org.springframework.beans.factory.BeanFactory;
4. **import** org.springframework.beans.factory.xml.XmlBeanFactory;
5. **import** org.springframework.core.io.ClassPathResource;
6. **import** org.springframework.core.io.Resource;
8. **public** **class** Test {
9. **public** **static** **void** main(String[] args) {
10. Resource r=**new** ClassPathResource("applicationContext.xml");
11. BeanFactory factory=**new** XmlBeanFactory(r);
13. Question q=(Question)factory.getBean("q");
14. q.displayInfo();
16. }
17. }

Constructor Injection with Non-String Collection (having Dependent Object) Example

1. [Constructor Injection with Non-String Collection](https://www.javatpoint.com/spring-tutorial-constructor-injection-with-non-string-collection)

If we have dependent object in the collection, we can inject these information by using the **ref** element inside the **list**, **set** or **map**.

In this example, we are taking the example of Forum where **One question can have multiple answers**. But Answer has its own information such as answerId, answer and postedBy. There are four pages used in this example:

1. **Question.java**
2. **Answer.java**
3. **applicationContext.xml**
4. **Test.java**

In this example, we are using list that can have duplicate elements, you may use set that have only unique elements. But, you need to change list to set in the applicationContext.xml file and List to Set in the Question.java file.

**Question.java**

This class contains three properties, two constructors and displayInfo() method that prints the information. Here, we are using List to contain the multiple answers.

Features of Java - Javatpoint

1. **package** com.javatpoint;
3. **import** java.util.Iterator;
4. **import** java.util.List;
6. **public** **class** Question {
7. **private** **int** id;
8. **private** String name;
9. **private** List<Answer> answers;
11. **public** Question() {}
12. **public** Question(**int** id, String name, List<Answer> answers) {
13. **super**();
14. **this**.id = id;
15. **this**.name = name;
16. **this**.answers = answers;
17. }
19. **public** **void** displayInfo(){
20. System.out.println(id+" "+name);
21. System.out.println("answers are:");
22. Iterator<Answer> itr=answers.iterator();
23. **while**(itr.hasNext()){
24. System.out.println(itr.next());
25. }
26. }
28. }

**Answer.java**

This class has three properties id, name and by with constructor and toString() method.

1. **package** com.javatpoint;
3. **public** **class** Answer {
4. **private** **int** id;
5. **private** String name;
6. **private** String by;
8. **public** Answer() {}
9. **public** Answer(**int** id, String name, String by) {
10. **super**();
11. **this**.id = id;
12. **this**.name = name;
13. **this**.by = by;
14. }
16. **public** String toString(){
17. **return** id+" "+name+" "+by;
18. }
19. }

**applicationContext.xml**

The **ref** element is used to define the reference of another bean. Here, we are using **bean** attribute of **ref** element to specify the reference of another bean.

1. <?xml version="1.0" encoding="UTF-8"?>
2. <beans
3. xmlns="http://www.springframework.org/schema/beans"
4. xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
5. xmlns:p="http://www.springframework.org/schema/p"
6. xsi:schemaLocation="http://www.springframework.org/schema/beans
7. http://www.springframework.org/schema/beans/spring-beans-3.0.xsd">
9. <bean id="ans1" **class**="com.javatpoint.Answer">
10. <constructor-arg value="1"></constructor-arg>
11. <constructor-arg value="Java is a programming language"></constructor-arg>
12. <constructor-arg value="John"></constructor-arg>
13. </bean>
15. <bean id="ans2" **class**="com.javatpoint.Answer">
16. <constructor-arg value="2"></constructor-arg>
17. <constructor-arg value="Java is a Platform"></constructor-arg>
18. <constructor-arg value="Ravi"></constructor-arg>
19. </bean>
21. <bean id="q" **class**="com.javatpoint.Question">
22. <constructor-arg value="111"></constructor-arg>
23. <constructor-arg value="What is java?"></constructor-arg>
24. <constructor-arg>
25. <list>
26. <ref bean="ans1"/>
27. <ref bean="ans2"/>
28. </list>
29. </constructor-arg>
30. </bean>
32. </beans>

**Test.java**

This class gets the bean from the applicationContext.xml file and calls the displayInfo method.

1. **package** com.javatpoint;
3. **import** org.springframework.beans.factory.BeanFactory;
4. **import** org.springframework.beans.factory.xml.XmlBeanFactory;
5. **import** org.springframework.core.io.ClassPathResource;
6. **import** org.springframework.core.io.Resource;
8. **public** **class** Test {
9. **public** **static** **void** main(String[] args) {
10. Resource r=**new** ClassPathResource("applicationContext.xml");
11. BeanFactory factory=**new** XmlBeanFactory(r);
13. Question q=(Question)factory.getBean("q");
14. q.displayInfo();
16. }
17. }

Constructor Injection with Map Example

1. [Constructor Injection with Map Example](https://www.javatpoint.com/spring-tutorial-constructor-injection-with-map)

In this example, we are using **map** as the answer that have answer with posted username. Here, we are using key and value pair both as a string.

Like previous examples, it is the example of forum where **one question can have multiple answers**.

**Question.java**

This class contains three properties, two constructors and displayInfo() method to display the information.

Difference between JDK, JRE, and JVM

1. **package** com.javatpoint;
2. **import** java.util.Iterator;
3. **import** java.util.Map;
4. **import** java.util.Set;
5. **import** java.util.Map.Entry;
7. **public** **class** Question {
8. **private** **int** id;
9. **private** String name;
10. **private** Map<String,String> answers;
12. **public** Question() {}
13. **public** Question(**int** id, String name, Map<String, String> answers) {
14. **super**();
15. **this**.id = id;
16. **this**.name = name;
17. **this**.answers = answers;
18. }
20. **public** **void** displayInfo(){
21. System.out.println("question id:"+id);
22. System.out.println("question name:"+name);
23. System.out.println("Answers....");
24. Set<Entry<String, String>> set=answers.entrySet();
25. Iterator<Entry<String, String>> itr=set.iterator();
26. **while**(itr.hasNext()){
27. Entry<String,String> entry=itr.next();
28. System.out.println("Answer:"+entry.getKey()+" Posted By:"+entry.getValue());
29. }
30. }
31. }

**applicationContext.xml**

The **entry** attribute of **map** is used to define the key and value information.

1. <?xml version="1.0" encoding="UTF-8"?>
2. <beans
3. xmlns="http://www.springframework.org/schema/beans"
4. xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
5. xmlns:p="http://www.springframework.org/schema/p"
6. xsi:schemaLocation="http://www.springframework.org/schema/beans
7. http://www.springframework.org/schema/beans/spring-beans-3.0.xsd">
9. <bean id="q" **class**="com.javatpoint.Question">
10. <constructor-arg value="11"></constructor-arg>
11. <constructor-arg value="What is Java?"></constructor-arg>
12. <constructor-arg>
13. <map>
14. <entry key="Java is a Programming Language"  value="Ajay Kumar"></entry>
15. <entry key="Java is a Platform" value="John Smith"></entry>
16. <entry key="Java is an Island" value="Raj Kumar"></entry>
17. </map>
18. </constructor-arg>
19. </bean>
21. </beans>

**Test.java**

This class gets the bean from the applicationContext.xml file and calls the displayInfo() method.

1. **package** com.javatpoint;
3. **import** org.springframework.beans.factory.BeanFactory;
4. **import** org.springframework.beans.factory.xml.XmlBeanFactory;
5. **import** org.springframework.core.io.ClassPathResource;
6. **import** org.springframework.core.io.Resource;
8. **public** **class** Test {
9. **public** **static** **void** main(String[] args) {
10. Resource r=**new** ClassPathResource("applicationContext.xml");
11. BeanFactory factory=**new** XmlBeanFactory(r);
13. Question q=(Question)factory.getBean("q");
14. q.displayInfo();
16. }
17. }

Constructor Injection with Non-String Map (having dependent Object) Example

1. [Constructor Injection with Non-String Map](https://www.javatpoint.com/spring-tutorial-constructor-injection-with-non-string-map)

In this example, we are using **map** as the answer that have Answer and User. Here, we are using key and value pair both as an object. Answer has its own information such as answerId, answer and postedDate, User has its own information such as userId, username, emailId.

Like previous examples, it is the example of forum where **one question can have multiple answers**.

**Question.java**

This class contains three properties, two constructors and displayInfo() method to display the information.

31M

727

OOPs Concepts in Java

1. **package** com.javatpoint;
2. **import** java.util.Iterator;
3. **import** java.util.Map;
4. **import** java.util.Set;
5. **import** java.util.Map.Entry;
7. **public** **class** Question {
8. **private** **int** id;
9. **private** String name;
10. **private** Map<Answer,User> answers;
12. **public** Question() {}
13. **public** Question(**int** id, String name, Map<Answer, User> answers) {
14. **super**();
15. **this**.id = id;
16. **this**.name = name;
17. **this**.answers = answers;
18. }

21. **public** **void** displayInfo(){
22. System.out.println("question id:"+id);
23. System.out.println("question name:"+name);
24. System.out.println("Answers....");
25. Set<Entry<Answer, User>> set=answers.entrySet();
26. Iterator<Entry<Answer, User>> itr=set.iterator();
27. **while**(itr.hasNext()){
28. Entry<Answer, User> entry=itr.next();
29. Answer ans=entry.getKey();
30. User user=entry.getValue();
31. System.out.println("Answer Information:");
32. System.out.println(ans);
33. System.out.println("Posted By:");
34. System.out.println(user);
35. }
36. }
37. }

**Answer.java**

1. **package** com.javatpoint;
3. **import** java.util.Date;
5. **public** **class** Answer {
6. **private** **int** id;
7. **private** String answer;
8. **private** Date postedDate;
9. **public** Answer() {}
10. **public** Answer(**int** id, String answer, Date postedDate) {
11. **super**();
12. **this**.id = id;
13. **this**.answer = answer;
14. **this**.postedDate = postedDate;
15. }
17. **public** String toString(){
18. **return** "Id:"+id+" Answer:"+answer+" Posted Date:"+postedDate;
19. }
20. }

**User.java**

1. **package** com.javatpoint;
3. **public** **class** User {
4. **private** **int** id;
5. **private** String name,email;
6. **public** User() {}
7. **public** User(**int** id, String name, String email) {
8. **super**();
9. **this**.id = id;
10. **this**.name = name;
11. **this**.email = email;
12. }
14. **public** String toString(){
15. **return** "Id:"+id+" Name:"+name+" Email Id:"+email;
16. }
17. }

**applicationContext.xml**

The **key-ref** and **value-ref** attributes of entry **element** is used to define the reference of bean in the map.

1. <?xml version="1.0" encoding="UTF-8"?>
2. <beans
3. xmlns="http://www.springframework.org/schema/beans"
4. xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
5. xmlns:p="http://www.springframework.org/schema/p"
6. xsi:schemaLocation="http://www.springframework.org/schema/beans
7. http://www.springframework.org/schema/beans/spring-beans-3.0.xsd">
9. <bean id="answer1" **class**="com.javatpoint.Answer">
10. <constructor-arg value="1"></constructor-arg>
11. <constructor-arg value="Java is a Programming Language"></constructor-arg>
12. <constructor-arg value="12/12/2001"></constructor-arg>
13. </bean>
14. <bean id="answer2" **class**="com.javatpoint.Answer">
15. <constructor-arg value="2"></constructor-arg>
16. <constructor-arg value="Java is a Platform"></constructor-arg>
17. <constructor-arg value="12/12/2003"></constructor-arg>
18. </bean>
20. <bean id="user1" **class**="com.javatpoint.User">
21. <constructor-arg value="1"></constructor-arg>
22. <constructor-arg value="Arun Kumar"></constructor-arg>
23. <constructor-arg value="arun@gmail.com"></constructor-arg>
24. </bean>
25. <bean id="user2" **class**="com.javatpoint.User">
26. <constructor-arg value="2"></constructor-arg>
27. <constructor-arg value="Varun Kumar"></constructor-arg>
28. <constructor-arg value="Varun@gmail.com"></constructor-arg>
29. </bean>
31. <bean id="q" **class**="com.javatpoint.Question">
32. <constructor-arg value="1"></constructor-arg>
33. <constructor-arg value="What is Java?"></constructor-arg>
34. <constructor-arg>
35. <map>
36. <entry key-ref="answer1" value-ref="user1"></entry>
37. <entry key-ref="answer2" value-ref="user2"></entry>
38. </map>
39. </constructor-arg>
40. </bean>
42. </beans>

**Test.java**

This class gets the bean from the applicationContext.xml file and calls the displayInfo() method to display the information.

1. **package** com.javatpoint;
3. **import** org.springframework.beans.factory.BeanFactory;
4. **import** org.springframework.beans.factory.xml.XmlBeanFactory;
5. **import** org.springframework.core.io.ClassPathResource;
6. **import** org.springframework.core.io.Resource;
8. **public** **class** Test {
9. **public** **static** **void** main(String[] args) {
10. Resource r=**new** ClassPathResource("applicationContext.xml");
11. BeanFactory factory=**new** XmlBeanFactory(r);
13. Question q=(Question)factory.getBean("q");
14. q.displayInfo();
16. }
17. }

Inheriting Bean in Spring

1. [Inheriting Bean in Spring](https://www.javatpoint.com/spring-tutorial-inheriting-bean-in-spring)

By using the **parent** attribute of **bean**, we can specify the inheritance relation between the beans. In such case, parent bean values will be inherited to the current bean.

Let's see the simple example to inherit the bean.

**Employee.java**

This class contains three properties, three constructor and show() method to display the values.

1. **package** com.javatpoint;
3. **public** **class** Employee {
4. **private** **int** id;
5. **private** String name;
6. **private** Address address;
7. **public** Employee() {}
9. **public** Employee(**int** id, String name) {
10. **super**();
11. **this**.id = id;
12. **this**.name = name;
13. }
14. **public** Employee(**int** id, String name, Address address) {
15. **super**();
16. **this**.id = id;
17. **this**.name = name;
18. **this**.address = address;
19. }
21. **void** show(){
22. System.out.println(id+" "+name);
23. System.out.println(address);
24. }
26. }

**Address.java**

1. **package** com.javatpoint;
3. **public** **class** Address {
4. **private** String addressLine1,city,state,country;
6. **public** Address(String addressLine1, String city, String state, String country) {
7. **super**();
8. **this**.addressLine1 = addressLine1;
9. **this**.city = city;
10. **this**.state = state;
11. **this**.country = country;
12. }
13. **public** String toString(){
14. **return** addressLine1+" "+city+" "+state+" "+country;
15. }
17. }

**applicationContext.xml**

1. <?xml version="1.0" encoding="UTF-8"?>
2. <beans
3. xmlns="http://www.springframework.org/schema/beans"
4. xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
5. xmlns:p="http://www.springframework.org/schema/p"
6. xsi:schemaLocation="http://www.springframework.org/schema/beans
7. http://www.springframework.org/schema/beans/spring-beans-3.0.xsd">
9. <bean id="e1" **class**="com.javatpoint.Employee">
10. <constructor-arg value="101"></constructor-arg>
11. <constructor-arg  value="Sachin"></constructor-arg>
12. </bean>
14. <bean id="address1" **class**="com.javatpoint.Address">
15. <constructor-arg value="21,Lohianagar"></constructor-arg>
16. <constructor-arg value="Ghaziabad"></constructor-arg>
17. <constructor-arg value="UP"></constructor-arg>
18. <constructor-arg value="USA"></constructor-arg>
19. </bean>
21. <bean id="e2" **class**="com.javatpoint.Employee" parent="e1">
22. <constructor-arg ref="address1"></constructor-arg>
23. </bean>
25. </beans>

**Test.java**

This class gets the bean from the applicationContext.xml file and calls the show method.

Java Try Catch

1. **package** com.javatpoint;
3. **import** org.springframework.beans.factory.BeanFactory;
4. **import** org.springframework.beans.factory.xml.XmlBeanFactory;
5. **import** org.springframework.core.io.ClassPathResource;
6. **import** org.springframework.core.io.Resource;
8. **public** **class** Test {
9. **public** **static** **void** main(String[] args) {
10. Resource r=**new** ClassPathResource("applicationContext.xml");
11. BeanFactory factory=**new** XmlBeanFactory(r);
13. Employee e1=(Employee)factory.getBean("e2");
14. e1.show();
16. }
17. }

Dependency Injection by setter method

1. [Dependency Injection by constructor](https://www.javatpoint.com/spring-tutorial-dependency-injection-by-setter-method)
2. [Injecting primitive and string-based values](https://www.javatpoint.com/spring-tutorial-dependency-injection-by-setter-method#ips)

We can inject the dependency by setter method also. The **<property>** subelement of **<bean>** is used for setter injection. Here we are going to inject

1. primitive and String-based values
2. Dependent object (contained object)
3. Collection values etc.

Injecting primitive and string-based values by setter method

Let's see the simple example to inject primitive and string-based values by setter method. We have created three files here:

* Employee.java
* applicationContext.xml
* Test.java

**Employee.java**

It is a simple class containing three fields id, name and city with its setters and getters and a method to display these informations.

1. **package** com.javatpoint;
3. **public** **class** Employee {
4. **private** **int** id;
5. **private** String name;
6. **private** String city;
8. **public** **int** getId() {
9. **return** id;
10. }
11. **public** **void** setId(**int** id) {
12. **this**.id = id;
13. }
14. **public** String getName() {
15. **return** name;
16. }
17. **public** **void** setName(String name) {
18. **this**.name = name;
19. }
21. **public** String getCity() {
22. **return** city;
23. }
24. **public** **void** setCity(String city) {
25. **this**.city = city;
26. }
27. **void** display(){
28. System.out.println(id+" "+name+" "+city);
29. }
31. }

**applicationContext.xml**

We are providing the information into the bean by this file. The property element invokes the setter method. The value subelement of property will assign the specified value.

How to find Nth Highest Salary in SQL

1. <?xml version="1.0" encoding="UTF-8"?>
2. <beans
3. xmlns="http://www.springframework.org/schema/beans"
4. xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
5. xmlns:p="http://www.springframework.org/schema/p"
6. xsi:schemaLocation="http://www.springframework.org/schema/beans
7. http://www.springframework.org/schema/beans/spring-beans-3.0.xsd">
9. <bean id="obj" **class**="com.javatpoint.Employee">
10. <property name="id">
11. <value>20</value>
12. </property>
13. <property name="name">
14. <value>Arun</value>
15. </property>
16. <property name="city">
17. <value>ghaziabad</value>
18. </property>
20. </bean>
22. </beans>

**Test.java**

This class gets the bean from the applicationContext.xml file and calls the display method.

1. **package** com.javatpoint;
3. **import** org.springframework.beans.factory.BeanFactory;
4. **import** org.springframework.beans.factory.xml.XmlBeanFactory;
5. **import** org.springframework.core.io.\*;
7. **public** **class** Test {
8. **public** **static** **void** main(String[] args) {
10. Resource r=**new** ClassPathResource("applicationContext.xml");
11. BeanFactory factory=**new** XmlBeanFactory(r);
13. Employee e=(Employee)factory.getBean("obj");
14. s.display();
16. }
17. }

**Output:**20 Arun ghaziabad

Setter Injection with Dependent Object Example

1. [Setter Injection with Dependent Object](https://www.javatpoint.com/spring-tutorial-setter-injection-with-dependent-object)

Like Constructor Injection, we can inject the dependency of another bean using setters. In such case, we use **property** element. Here, our scenario is **Employee HAS-A Address**. The Address class object will be termed as the dependent object. Let's see the Address class first:

**Address.java**

This class contains four properties, setters and getters and toString() method.

1. **package** com.javatpoint;
3. **public** **class** Address {
4. **private** String addressLine1,city,state,country;
6. //getters and setters
8. **public** String toString(){
9. **return** addressLine1+" "+city+" "+state+" "+country;
10. }

**Employee.java**

It contains three properties id, name and address(dependent object) , setters and getters with displayInfo() method.

1. **package** com.javatpoint;
3. **public** **class** Employee {
4. **private** **int** id;
5. **private** String name;
6. **private** Address address;
8. //setters and getters
10. **void** displayInfo(){
11. System.out.println(id+" "+name);
12. System.out.println(address);
13. }
14. }

**applicationContext.xml**

The **ref** attribute of **property** elements is used to define the reference of another bean.

33.1M

751

Features of Java - Javatpoint

1. <?xml version="1.0" encoding="UTF-8"?>
2. <beans
3. xmlns="http://www.springframework.org/schema/beans"
4. xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
5. xmlns:p="http://www.springframework.org/schema/p"
6. xsi:schemaLocation="http://www.springframework.org/schema/beans
7. http://www.springframework.org/schema/beans/spring-beans-3.0.xsd">
9. <bean id="address1" **class**="com.javatpoint.Address">
10. <property name="addressLine1" value="51,Lohianagar"></property>
11. <property name="city" value="Ghaziabad"></property>
12. <property name="state" value="UP"></property>
13. <property name="country" value="India"></property>
14. </bean>
16. <bean id="obj" **class**="com.javatpoint.Employee">
17. <property name="id" value="1"></property>
18. <property name="name" value="Sachin Yadav"></property>
19. <property name="address" ref="address1"></property>
20. </bean>
22. </beans>

**Test.java**

This class gets the bean from the applicationContext.xml file and calls the displayInfo() method.

1. **package** com.javatpoint;
3. **import** org.springframework.beans.factory.BeanFactory;
4. **import** org.springframework.beans.factory.xml.XmlBeanFactory;
5. **import** org.springframework.context.ApplicationContext;
6. **import** org.springframework.context.support.ClassPathXmlApplicationContext;
7. **import** org.springframework.core.io.ClassPathResource;
8. **import** org.springframework.core.io.Resource;
10. **public** **class** Test {
11. **public** **static** **void** main(String[] args) {
12. Resource r=**new** ClassPathResource("applicationContext.xml");
13. BeanFactory factory=**new** XmlBeanFactory(r);
15. Employee e=(Employee)factory.getBean("obj");
16. e.displayInfo();
18. }
19. }

Setter Injection with Collection Example

1. [Setter Injection with Collection](https://www.javatpoint.com/spring-tutorial-setter-injection-with-collection)

We can inject collection values by setter method in spring framework. There can be used three elements inside the **property** element.

It can be:

1. **list**
2. **set**
3. **map**

Each collection can have string based and non-string based values.

In this example, we are taking the example of Forum where **One question can have multiple answers**. There are three pages:

1. **Question.java**
2. **applicationContext.xml**
3. **Test.java**

In this example, we are using list that can have duplicate elements, you may use set that have only unique elements. But, you need to change list to set in the applicationContext.xml file and List to Set in the Question.java file.

**Question.java**

This class contains three properties with setters and getters and displayInfo() method that prints the information. Here, we are using List to contain the multiple answers.

History of Java

1. **package** com.javatpoint;
2. **import** java.util.Iterator;
3. **import** java.util.List;
5. **public** **class** Question {
6. **private** **int** id;
7. **private** String name;
8. **private** List<String> answers;
10. //setters and getters
12. **public** **void** displayInfo(){
13. System.out.println(id+" "+name);
14. System.out.println("answers are:");
15. Iterator<String> itr=answers.iterator();
16. **while**(itr.hasNext()){
17. System.out.println(itr.next());
18. }
19. }
21. }

**applicationContext.xml**

The list element of constructor-arg is used here to define the list.

1. <?xml version="1.0" encoding="UTF-8"?>
2. <beans
3. xmlns="http://www.springframework.org/schema/beans"
4. xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
5. xmlns:p="http://www.springframework.org/schema/p"
6. xsi:schemaLocation="http://www.springframework.org/schema/beans
7. http://www.springframework.org/schema/beans/spring-beans-3.0.xsd">
9. <bean id="q" **class**="com.javatpoint.Question">
10. <property name="id" value="1"></property>
11. <property name="name" value="What is Java?"></property>
12. <property name="answers">
13. <list>
14. <value>Java is a programming language</value>
15. <value>Java is a platform</value>
16. <value>Java is an Island</value>
17. </list>
18. </property>
19. </bean>
21. </beans>

**Test.java**

This class gets the bean from the applicationContext.xml file and calls the displayInfo method.

1. **package** com.javatpoint;
3. **import** org.springframework.beans.factory.BeanFactory;
4. **import** org.springframework.beans.factory.xml.XmlBeanFactory;
5. **import** org.springframework.core.io.ClassPathResource;
6. **import** org.springframework.core.io.Resource;
8. **public** **class** Test {
9. **public** **static** **void** main(String[] args) {
10. Resource r=**new** ClassPathResource("applicationContext.xml");
11. BeanFactory factory=**new** XmlBeanFactory(r);
13. Question q=(Question)factory.getBean("q");
14. q.displayInfo();
16. }
17. }

Setter Injection with Non-String Collection (having Dependent Object) Example

1. [Setter Injection with Non-String Collection](https://www.javatpoint.com/spring-tutorial-setter-injection-with-non-string-collection)

If we have dependent object in the collection, we can inject these information by using the **ref** element inside the **list**, **set** or **map**. Here, we will use list, set or map element inside the **property** element.

In this example, we are taking the example of Forum where **One question can have multiple answers**. But Answer has its own information such as answerId, answer and postedBy. There are four pages used in this example:

1. **Question.java**
2. **Answer.java**
3. **applicationContext.xml**
4. **Test.java**

In this example, we are using list that can have duplicate elements, you may use set that have only unique elements. But, you need to change list to set in the applicationContext.xml file and List to Set in the Question.java file.

**Question.java**

This class contains three properties, two constructors and displayInfo() method that prints the information. Here, we are using List to contain the multiple answers.

Prime Ministers of India | List of Prime Minister of India (1947-2020)

1. **package** com.javatpoint;
3. **import** java.util.Iterator;
4. **import** java.util.List;
6. **public** **class** Question {
7. **private** **int** id;
8. **private** String name;
9. **private** List<Answer> answers;
11. //setters and getters
13. **public** **void** displayInfo(){
14. System.out.println(id+" "+name);
15. System.out.println("answers are:");
16. Iterator<Answer> itr=answers.iterator();
17. **while**(itr.hasNext()){
18. System.out.println(itr.next());
19. }
20. }
22. }

**Answer.java**

This class has three properties id, name and by with constructor and toString() method.

1. **package** com.javatpoint;
3. **public** **class** Answer {
4. **private** **int** id;
5. **private** String name;
6. **private** String by;
8. //setters and getters
10. **public** String toString(){
11. **return** id+" "+name+" "+by;
12. }
13. }

**applicationContext.xml**

The **ref** element is used to define the reference of another bean. Here, we are using **bean** attribute of **ref** element to specify the reference of another bean.

1. <?xml version="1.0" encoding="UTF-8"?>
2. <beans
3. xmlns="http://www.springframework.org/schema/beans"
4. xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
5. xmlns:p="http://www.springframework.org/schema/p"
6. xsi:schemaLocation="http://www.springframework.org/schema/beans
7. http://www.springframework.org/schema/beans/spring-beans-3.0.xsd">
9. <bean id="answer1" **class**="com.javatpoint.Answer">
10. <property name="id" value="1"></property>
11. <property name="name" value="Java is a programming language"></property>
12. <property name="by" value="Ravi Malik"></property>
13. </bean>
14. <bean id="answer2" **class**="com.javatpoint.Answer">
15. <property name="id" value="2"></property>
16. <property name="name" value="Java is a platform"></property>
17. <property name="by" value="Sachin"></property>
18. </bean>
20. <bean id="q" **class**="com.javatpoint.Question">
21. <property name="id" value="1"></property>
22. <property name="name" value="What is Java?"></property>
23. <property name="answers">
24. <list>
25. <ref bean="answer1"/>
26. <ref bean="answer2"/>
27. </list>
28. </property>
29. </bean>
31. </beans>

**Test.java**

This class gets the bean from the applicationContext.xml file and calls the displayInfo method.

1. **package** com.javatpoint;
3. **import** org.springframework.beans.factory.BeanFactory;
4. **import** org.springframework.beans.factory.xml.XmlBeanFactory;
5. **import** org.springframework.core.io.ClassPathResource;
6. **import** org.springframework.core.io.Resource;
8. **public** **class** Test {
9. **public** **static** **void** main(String[] args) {
10. Resource r=**new** ClassPathResource("applicationContext.xml");
11. BeanFactory factory=**new** XmlBeanFactory(r);
13. Question q=(Question)factory.getBean("q");
14. q.displayInfo();
16. }
17. }

Setter Injection with Map Example

1. [Setter Injection with Map Example](https://www.javatpoint.com/spring-tutorial-setter-injection-with-map)

In this example, we are using **map** as the answer for a question that have answer as the key and username as the value. Here, we are using key and value pair both as a string.

Like previous examples, it is the example of forum where **one question can have multiple answers**.

**Question.java**

This class contains three properties, getters & setters and displayInfo() method to display the information.

History of Java

1. **package** com.javatpoint;
2. **import** java.util.Iterator;
3. **import** java.util.Map;
4. **import** java.util.Set;
5. **import** java.util.Map.Entry;
7. **public** **class** Question {
8. **private** **int** id;
9. **private** String name;
10. **private** Map<String,String> answers;
12. //getters and setters
14. **public** **void** displayInfo(){
15. System.out.println("question id:"+id);
16. System.out.println("question name:"+name);
17. System.out.println("Answers....");
18. Set<Entry<String, String>> set=answers.entrySet();
19. Iterator<Entry<String, String>> itr=set.iterator();
20. **while**(itr.hasNext()){
21. Entry<String,String> entry=itr.next();
22. System.out.println("Answer:"+entry.getKey()+" Posted By:"+entry.getValue());
23. }
24. }
25. }

**applicationContext.xml**

The **entry** attribute of **map** is used to define the key and value information.

1. <?xml version="1.0" encoding="UTF-8"?>
2. <beans
3. xmlns="http://www.springframework.org/schema/beans"
4. xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
5. xmlns:p="http://www.springframework.org/schema/p"
6. xsi:schemaLocation="http://www.springframework.org/schema/beans
7. http://www.springframework.org/schema/beans/spring-beans-3.0.xsd">
9. <bean id="q" **class**="com.javatpoint.Question">
10. <property name="id" value="1"></property>
11. <property name="name" value="What is Java?"></property>
12. <property name="answers">
13. <map>
14. <entry key="Java is a programming language"  value="Sonoo Jaiswal"></entry>
15. <entry key="Java is a Platform" value="Sachin Yadav"></entry>
16. </map>
17. </property>
18. </bean>
20. </beans>

**Test.java**

This class gets the bean from the applicationContext.xml file and calls the displayInfo() method.

1. **package** com.javatpoint;
3. **import** org.springframework.beans.factory.BeanFactory;
4. **import** org.springframework.beans.factory.xml.XmlBeanFactory;
5. **import** org.springframework.core.io.ClassPathResource;
6. **import** org.springframework.core.io.Resource;
8. **public** **class** Test {
9. **public** **static** **void** main(String[] args) {
10. Resource r=**new** ClassPathResource("applicationContext.xml");
11. BeanFactory factory=**new** XmlBeanFactory(r);
13. Question q=(Question)factory.getBean("q");
14. q.displayInfo();
16. }
17. }

Setter Injection with Non-String Map (having dependent Object) Example

1. [Setter Injection with Non-String Map](https://www.javatpoint.com/spring-tutorial-setter-injection-with-non-string-map)

In this example, we are using **map** as the answer that have Answer and User. Here, we are using key and value pair both as an object. Answer has its own information such as answerId, answer and postedDate, User has its own information such as userId, username, emailId.

Like previous examples, it is the example of forum where **one question can have multiple answers**.

**Question.java**

This class contains three properties, getters & setters and displayInfo() method to display the information.

33.8M

733

Exception Handling in Java - Javatpoint

1. **package** com.javatpoint;
2. **import** java.util.Iterator;
3. **import** java.util.Map;
4. **import** java.util.Set;
5. **import** java.util.Map.Entry;
7. **public** **class** Question {
8. **private** **int** id;
9. **private** String name;
10. **private** Map<Answer,User> answers;
12. //getters and setters

15. **public** **void** displayInfo(){
16. System.out.println("question id:"+id);
17. System.out.println("question name:"+name);
18. System.out.println("Answers....");
19. Set<Entry<Answer, User>> set=answers.entrySet();
20. Iterator<Entry<Answer, User>> itr=set.iterator();
21. **while**(itr.hasNext()){
22. Entry<Answer, User> entry=itr.next();
23. Answer ans=entry.getKey();
24. User user=entry.getValue();
25. System.out.println("Answer Information:");
26. System.out.println(ans);
27. System.out.println("Posted By:");
28. System.out.println(user);
29. }
30. }
31. }

**Answer.java**

1. **package** com.javatpoint;
3. **import** java.util.Date;
5. **public** **class** Answer {
6. **private** **int** id;
7. **private** String answer;
8. **private** Date postedDate;
9. **public** Answer() {}
10. **public** Answer(**int** id, String answer, Date postedDate) {
11. **super**();
12. **this**.id = id;
13. **this**.answer = answer;
14. **this**.postedDate = postedDate;
15. }
17. **public** String toString(){
18. **return** "Id:"+id+" Answer:"+answer+" Posted Date:"+postedDate;
19. }
20. }

**User.java**

1. **package** com.javatpoint;
3. **public** **class** User {
4. **private** **int** id;
5. **private** String name,email;
6. **public** User() {}
7. **public** User(**int** id, String name, String email) {
8. **super**();
9. **this**.id = id;
10. **this**.name = name;
11. **this**.email = email;
12. }
14. **public** String toString(){
15. **return** "Id:"+id+" Name:"+name+" Email Id:"+email;
16. }
17. }

**applicationContext.xml**

The **key-ref** and **value-ref** attributes of entry **element** is used to define the reference of bean in the map.

1. <?xml version="1.0" encoding="UTF-8"?>
2. <beans
3. xmlns="http://www.springframework.org/schema/beans"
4. xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
5. xmlns:p="http://www.springframework.org/schema/p"
6. xsi:schemaLocation="http://www.springframework.org/schema/beans
7. http://www.springframework.org/schema/beans/spring-beans-3.0.xsd">
9. <bean id="answer1" **class**="com.javatpoint.Answer">
10. <property name="id" value="1"></property>
11. <property name="answer" value="Java is a Programming Language"></property>
12. <property name="postedDate" value="12/12/2001"></property>
13. </bean>
14. <bean id="answer2" **class**="com.javatpoint.Answer">
15. <property name="id" value="2"></property>
16. <property name="answer" value="Java is a Platform"></property>
17. <property name="postedDate" value="12/12/2003"></property>
18. </bean>
20. <bean id="user1" **class**="com.javatpoint.User">
21. <property name="id" value="1"></property>
22. <property name="name" value="Arun Kumar"></property>
23. <property name="email" value="arun@gmail.com"></property>
24. </bean>
25. <bean id="user2" **class**="com.javatpoint.User">
26. <property name="id" value="2"></property>
27. <property name="name" value="Varun Kumar"></property>
28. <property name="email" value="Varun@gmail.com"></property>
29. </bean>
31. <bean id="q" **class**="com.javatpoint.Question">
32. <property name="id" value="1"></property>
33. <property name="name" value="What is Java?"></property>
34. <property name="answers">
35. <map>
36. <entry key-ref="answer1" value-ref="user1"></entry>
37. <entry key-ref="answer2" value-ref="user2"></entry>
38. </map>
39. </property>
40. </bean>
42. </beans>

**Test.java**

This class gets the bean from the applicationContext.xml file and calls the displayInfo() method to display the information.

1. **package** com.javatpoint;
3. **import** org.springframework.beans.factory.BeanFactory;
4. **import** org.springframework.beans.factory.xml.XmlBeanFactory;
5. **import** org.springframework.core.io.ClassPathResource;
6. **import** org.springframework.core.io.Resource;
8. **public** **class** Test {
9. **public** **static** **void** main(String[] args) {
10. Resource r=**new** ClassPathResource("applicationContext.xml");
11. BeanFactory factory=**new** XmlBeanFactory(r);
13. Question q=(Question)factory.getBean("q");
14. q.displayInfo();
16. }
17. }

Difference between constructor and setter injection

1. [Difference between constructor and setter injection](https://www.javatpoint.com/difference-between-constructor-and-setter-injection)

There are many key differences between constructor injection and setter injection.

1. **Partial dependency**: can be injected using setter injection but it is not possible by constructor. Suppose there are 3 properties in a class, having 3 arg constructor and setters methods. In such case, if you want to pass information for only one property, it is possible by setter method only.
2. **Overriding**: Setter injection overrides the constructor injection. If we use both constructor and setter injection, IOC container will use the setter injection.
3. **Changes**: We can easily change the value by setter injection. It doesn't create a new bean instance always like constructor. So setter injection is flexible than constructor injection.

Autowiring in Spring

Autowiring feature of spring framework enables you to inject the object dependency implicitly. It internally uses setter or constructor injection.

Autowiring can't be used to inject primitive and string values. It works with reference only.

Advantage of Autowiring

It requires the **less code** because we don't need to write the code to inject the dependency explicitly.

Disadvantage of Autowiring

No control of programmer.

00:00/11:48

10 Sec

22.9M

430

Prime Ministers of India | List of Prime Minister of India (1947-2020)

It can't be used for primitive and string values.

Autowiring Modes

There are many autowiring modes:

|  |  |  |
| --- | --- | --- |
| **No.** | **Mode** | **Description** |
| 1) | no | It is the default autowiring mode. It means no autowiring bydefault. |
| 2) | byName | The byName mode injects the object dependency according to name of the bean. In such case, property name and bean name must be same. It internally calls setter method. |
| 3) | byType | The byType mode injects the object dependency according to type. So property name and bean name can be different. It internally calls setter method. |
| 4) | constructor | The constructor mode injects the dependency by calling the constructor of the class. It calls the constructor having large number of parameters. |
| 5) | autodetect | It is deprecated since Spring 3. |

Example of Autowiring

Let's see the simple code to use autowiring in spring. You need to use autowire attribute of bean element to apply the autowire modes.

1. <bean id="a" **class**="org.sssit.A" autowire="byName"></bean>

Let's see the full example of autowiring in spring. To create this example, we have created 4 files.

1. **B.java**
2. **A.java**
3. **applicationContext.xml**
4. **Test.java**

**B.java**

This class contains a constructor and method only.

1. **package** org.sssit;
2. **public** **class** B {
3. B(){System.out.println("b is created");}
4. **void** print(){System.out.println("hello b");}
5. }

**A.java**

This class contains reference of B class and constructor and method.

1. **package** org.sssit;
2. **public** **class** A {
3. B b;
4. A(){System.out.println("a is created");}
5. **public** B getB() {
6. **return** b;
7. }
8. **public** **void** setB(B b) {
9. **this**.b = b;
10. }
11. **void** print(){System.out.println("hello a");}
12. **void** display(){
13. print();
14. b.print();
15. }
16. }

**applicationContext.xml**

1. <?xml version="1.0" encoding="UTF-8"?>
2. <beans
3. xmlns="http://www.springframework.org/schema/beans"
4. xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
5. xmlns:p="http://www.springframework.org/schema/p"
6. xsi:schemaLocation="http://www.springframework.org/schema/beans
7. http://www.springframework.org/schema/beans/spring-beans-3.0.xsd">
9. <bean id="b" **class**="org.sssit.B"></bean>
10. <bean id="a" **class**="org.sssit.A" autowire="byName"></bean>
12. </beans>

**Test.java**

This class gets the bean from the applicationContext.xml file and calls the display method.

1. **package** org.sssit;
2. **import** org.springframework.context.ApplicationContext;
3. **import** org.springframework.context.support.ClassPathXmlApplicationContext;
4. **public** **class** Test {
5. **public** **static** **void** main(String[] args) {
6. ApplicationContext context=**new** ClassPathXmlApplicationContext("applicationContext.xml");
7. A a=context.getBean("a",A.**class**);
8. a.display();
9. }
10. }

Output:

b is created

a is created

hello a

hello b

1) byName autowiring mode

In case of byName autowiring mode, bean id and reference name must be same.

It internally uses setter injection.

1. <bean id="b" **class**="org.sssit.B"></bean>
2. <bean id="a" **class**="org.sssit.A" autowire="byName"></bean>

But, if you change the name of bean, it will not inject the dependency.

Let's see the code where we are changing the name of the bean from b to b1.

1. <bean id="b1" **class**="org.sssit.B"></bean>
2. <bean id="a" **class**="org.sssit.A" autowire="byName"></bean>

2) byType autowiring mode

In case of byType autowiring mode, bean id and reference name may be different. But there must be only one bean of a type.

It internally uses setter injection.

1. <bean id="b1" **class**="org.sssit.B"></bean>
2. <bean id="a" **class**="org.sssit.A" autowire="byType"></bean>

In this case, it works fine because you have created an instance of B type. It doesn't matter that you have different bean name than reference name.

But, if you have multiple bean of one type, it will not work and throw exception.

Let's see the code where are many bean of type B.

1. <bean id="b1" **class**="org.sssit.B"></bean>
2. <bean id="b2" **class**="org.sssit.B"></bean>
3. <bean id="a" **class**="org.sssit.A" autowire="byName"></bean>

In such case, it will throw exception.

3) constructor autowiring mode

In case of constructor autowiring mode, spring container injects the dependency by highest parameterized constructor.

If you have 3 constructors in a class, zero-arg, one-arg and two-arg then injection will be performed by calling the two-arg constructor.

1. <bean id="b" **class**="org.sssit.B"></bean>
2. <bean id="a" **class**="org.sssit.A" autowire="constructor"></bean>

4) no autowiring mode

In case of no autowiring mode, spring container doesn't inject the dependency by autowiring.

1. <bean id="b" **class**="org.sssit.B"></bean>
2. <bean id="a" **class**="org.sssit.A" autowire="no"></bean>

Dependency Injection with Factory Method in Spring

Spring framework provides facility to inject bean using factory method. To do so, we can use two attributes of bean element.

1. **factory-method:** represents the factory method that will be invoked to inject the bean.
2. **factory-bean:** represents the reference of the bean by which factory method will be invoked. It is used if factory method is non-static.

A method that returns instance of a class is called **factory method**.

1. **public** **class** A {
2. **public** **static** A getA(){//factory method
3. **return** **new** A();
4. }
5. }

Factory Method Types

There can be three types of factory method:

1) A **static factory method** that returns instance of **its own** class. It is used in singleton design pattern.

31M

727

OOPs Concepts in Java

1. <bean id="a" **class**="com.javatpoint.A" factory-method="getA"></bean>

2) A **static factory method** that returns instance of **another** class. It is used instance is not known and decided at runtime.

1. <bean id="b" **class**="com.javatpoint.A" factory-method="getB"></bean>

3) A **non-static factory** method that returns instance of **another** class. It is used instance is not known and decided at runtime.

1. <bean id="a" **class**="com.javatpoint.A"></bean>
2. <bean id="b" **class**="com.javatpoint.A" factory-method="getB" factory-bean="a"></bean>

Type 1

Let's see the simple code to inject the dependency by static factory method.

1. <bean id="a" **class**="com.javatpoint.A" factory-method="getA"></bean>

Let's see the full example to inject dependency using factory method in spring. To create this example, we have created 3 files.

1. **A.java**
2. **applicationContext.xml**
3. **Test.java**

**A.java**

This class is a singleton class.

1. **package** com.javatpoint;
2. **public** **class** A {
3. **private** **static** **final** A obj=**new** A();
4. **private** A(){System.out.println("private constructor");}
5. **public** **static** A getA(){
6. System.out.println("factory method ");
7. **return** obj;
8. }
9. **public** **void** msg(){
10. System.out.println("hello user");
11. }
12. }

**applicationContext.xml**

1. <?xml version="1.0" encoding="UTF-8"?>
2. <beans
3. xmlns="http://www.springframework.org/schema/beans"
4. xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
5. xmlns:p="http://www.springframework.org/schema/p"
6. xsi:schemaLocation="http://www.springframework.org/schema/beans
7. http://www.springframework.org/schema/beans/spring-beans-3.0.xsd">
9. <bean id="a" **class**="com.javatpoint.A" factory-method="getA"></bean>
11. </beans>

**Test.java**

This class gets the bean from the applicationContext.xml file and calls the msg method.

1. **package** org.sssit;
2. **import** org.springframework.context.ApplicationContext;
3. **import** org.springframework.context.support.ClassPathXmlApplicationContext;
4. **public** **class** Test {
5. **public** **static** **void** main(String[] args) {
6. ApplicationContext context=**new** ClassPathXmlApplicationContext("applicationContext.xml");
7. A a=(A)context.getBean("a");
8. a.msg();
9. }
10. }

Output:

private constructor

factory method

hello user

Type 2

Let's see the simple code to inject the dependency by static factory method that returns the instance of another class.

To create this example, we have created 6 files.

1. **Printable.java**
2. **A.java**
3. **B.java**
4. **PrintableFactory.java**
5. **applicationContext.xml**
6. **Test.java**

**Printable.java**

1. **package** com.javatpoint;
2. **public** **interface** Printable {
3. **void** print();
4. }

**A.java**

1. **package** com.javatpoint;
2. **public** **class** A **implements** Printable{
3. @Override
4. **public** **void** print() {
5. System.out.println("hello a");
6. }
8. }

**B.java**

1. **package** com.javatpoint;
2. **public** **class** B **implements** Printable{
3. @Override
4. **public** **void** print() {
5. System.out.println("hello b");
6. }
7. }

**PrintableFactory.java**

1. **package** com.javatpoint;
2. **public** **class** PrintableFactory {
3. **public** **static** Printable getPrintable(){
4. //return new B();
5. **return** **new** A();//return any one instance, either A or B
6. }
7. }

**applicationContext.xml**

1. <?xml version="1.0" encoding="UTF-8"?>
2. <beans
3. xmlns="http://www.springframework.org/schema/beans"
4. xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
5. xmlns:p="http://www.springframework.org/schema/p"
6. xsi:schemaLocation="http://www.springframework.org/schema/beans
7. http://www.springframework.org/schema/beans/spring-beans-3.0.xsd">
9. <bean id="p" **class**="com.javatpoint.PrintableFactory" factory-method="getPrintable"></bean>
11. </beans>

**Test.java**

This class gets the bean from the applicationContext.xml file and calls the print() method.

1. **package** org.sssit;
2. **import** org.springframework.context.ApplicationContext;
3. **import** org.springframework.context.support.ClassPathXmlApplicationContext;
4. **public** **class** Test {
5. **public** **static** **void** main(String[] args) {
6. ApplicationContext context=**new** ClassPathXmlApplicationContext("applicationContext.xml");
7. Printable p=(Printable)context.getBean("p");
8. p.print();
9. }
10. }

Output:

hello a

Type 3

Let's see the example to inject the dependency by non-static factory method that returns the instance of another class.

To create this example, we have created 6 files.

1. **Printable.java**
2. **A.java**
3. **B.java**
4. **PrintableFactory.java**
5. **applicationContext.xml**
6. **Test.java**

All files are same as previous, you need to change only 2 files: PrintableFactory and applicationContext.xml.

**PrintableFactory.java**

1. **package** com.javatpoint;
2. **public** **class** PrintableFactory {
3. //non-static factory method
4. **public** Printable getPrintable(){
5. **return** **new** A();//return any one instance, either A or B
6. }
7. }

**applicationContext.xml**

1. <?xml version="1.0" encoding="UTF-8"?>
2. <beans
3. xmlns="http://www.springframework.org/schema/beans"
4. xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
5. xmlns:p="http://www.springframework.org/schema/p"
6. xsi:schemaLocation="http://www.springframework.org/schema/beans
7. http://www.springframework.org/schema/beans/spring-beans-3.0.xsd">
9. <bean id="pfactory" **class**="com.javatpoint.PrintableFactory"></bean>
10. <bean id="p" **class**="com.javatpoint.PrintableFactory" factory-method="getPrintable"
11. factory-bean="pfactory"></bean>
13. </beans>

Output:

hello a

Spring AOP Tutorial

1. [What is AOP?](https://www.javatpoint.com/spring-aop-tutorial)
2. [Where use AOP?](https://www.javatpoint.com/spring-aop-tutorial)
3. [AOP Concepts and Terminology](https://www.javatpoint.com/spring-aop-tutorial)
4. [Spring AOP Implementation](https://www.javatpoint.com/spring-aop-tutorial)

**Aspect Oriented Programming** (AOP) compliments OOPs in the sense that it also provides modularity. But the key unit of modularity is aspect than class.

AOP breaks the program logic into distinct parts (called concerns). It is used to increase modularity by **cross-cutting concerns**.

A **cross-cutting concern** is a concern that can affect the whole application and should be centralized in one location in code as possible, such as transaction management, authentication, logging, security etc.

Why use AOP?

It provides the pluggable way to dynamically add the additional concern before, after or around the actual logic. Suppose there are 10 methods in a class as given below:

31M

727

OOPs Concepts in Java

1. **class** A{
2. **public** **void** m1(){...}
3. **public** **void** m2(){...}
4. **public** **void** m3(){...}
5. **public** **void** m4(){...}
6. **public** **void** m5(){...}
7. **public** **void** n1(){...}
8. **public** **void** n2(){...}
9. **public** **void** p1(){...}
10. **public** **void** p2(){...}
11. **public** **void** p3(){...}
12. }

There are 5 methods that starts from m, 2 methods that starts from n and 3 methods that starts from p.

**Understanding Scenario** I have to maintain log and send notification after calling methods that starts from m.

**Problem without AOP** We can call methods (that maintains log and sends notification) from the methods starting with m. In such scenario, we need to write the code in all the 5 methods.

But, if client says in future, I don't have to send notification, you need to change all the methods. It leads to the maintenance problem.

**Solution with AOP** We don't have to call methods from the method. Now we can define the additional concern like maintaining log, sending notification etc. in the method of a class. Its entry is given in the xml file.

In future, if client says to remove the notifier functionality, we need to change only in the xml file. So, maintenance is easy in AOP.

Where use AOP?

AOP is mostly used in following cases:

* to provide declarative enterprise services such as declarative transaction management.
* It allows users to implement custom aspects.

AOP Concepts and Terminology

AOP concepts and terminologies are as follows:

* Join point
* Advice
* Pointcut
* Introduction
* Target Object
* Aspect
* Interceptor
* AOP Proxy
* Weaving

Join point

Join point is any point in your program such as method execution, exception handling, field access etc. Spring supports only method execution join point.

Advice

Advice represents an action taken by an aspect at a particular join point. There are different types of advices:

* **Before Advice**: it executes before a join point.
* **After Returning Advice**: it executes after a joint point completes normally.
* **After Throwing Advice**: it executes if method exits by throwing an exception.
* **After (finally) Advice**: it executes after a join point regardless of join point exit whether normally or exceptional return.
* **Around Advice**: It executes before and after a join point.

Pointcut

It is an expression language of AOP that matches join points.

Introduction

It means introduction of additional method and fields for a type. It allows you to introduce new interface to any advised object.

Target Object

It is the object i.e. being advised by one or more aspects. It is also known as proxied object in spring because Spring AOP is implemented using runtime proxies.

Aspect

It is a class that contains advices, joinpoints etc.

Interceptor

It is an aspect that contains only one advice.

AOP Proxy

It is used to implement aspect contracts, created by AOP framework. It will be a JDK dynamic proxy or CGLIB proxy in spring framework.

Weaving

It is the process of linking aspect with other application types or objects to create an advised object. Weaving can be done at compile time, load time or runtime. Spring AOP performs weaving at runtime.

AOP Implementations

AOP implementations are provided by:

1. AspectJ
2. Spring AOP
3. JBoss AOP

Spring AOP

Spring AOP can be used by 3 ways given below. But the widely used approach is Spring AspectJ Annotation Style. The 3 ways to use spring AOP are given below:

1. By Spring1.2 Old style (dtd based) (also supported in Spring3)
2. By AspectJ annotation-style

# [Spring AOP AspectJ Annotation Example](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)

1. [[@Before Example](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)](https://www.javatpoint.com/spring-aop-aspectj-annotation-example)
2. [[@After Example](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)](https://www.javatpoint.com/spring-aop-aspectj-annotation-example)
3. [[@AfterReturning Example](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)](https://www.javatpoint.com/spring-aop-aspectj-annotation-example)
4. [[@Around Example](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)](https://www.javatpoint.com/spring-aop-aspectj-annotation-example)
5. [[@AfterThrowing Example](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)](https://www.javatpoint.com/spring-aop-aspectj-annotation-example)

[The](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[Spring Framework](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[recommends you to use](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[Spring AspectJ AOP implementation](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[over the Spring 1.2 old style dtd based AOP implementation because it provides you more control and it is easy to use.](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)

[There are two ways to use Spring AOP AspectJ implementation:](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)

1. [By annotation: We are going to learn it here.](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)
2. [By xml configuration (schema based): We will learn it in next page.](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)

[To understand the aop concepts, its advantage etc. visit here](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)[[AOP Concepts Tutorial](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)](https://www.javatpoint.com/spring-aop-tutorial)

[[download all examples (developed using MyEclipse IDE)](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)](https://static.javatpoint.com/src/sp/aopaspectjannotation.zip)

[Spring AspectJ AOP implementation provides many annotations:](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)

1. **[@Aspect](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[declares the class as aspect.](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)
2. **[@Pointcut](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[declares the pointcut expression.](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)

[The annotations used to create advices are given below:](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)

[26.2M](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)

[537](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)

[HTML Tutorial](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)

1. **[@Before](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[declares the before advice. It is applied before calling the actual method.](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)
2. **[@After](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[declares the after advice. It is applied after calling the actual method and before returning result.](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)
3. **[@AfterReturning](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[declares the after returning advice. It is applied after calling the actual method and before returning result. But you can get the result value in the advice.](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)
4. **[@Around](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[declares the around advice. It is applied before and after calling the actual method.](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)
5. **[@AfterThrowing](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[declares the throws advice. It is applied if actual method throws exception.](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)

[Understanding Pointcut](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)

[Pointcut is an expression language of Spring AOP.](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)

[The](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[@Pointcut](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[annotation is used to define the pointcut. We can refer the pointcut expression by name also. Let's see the simple example of pointcut expression.](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)

1. [@Pointcut("execution(\* Operation.\*(..))")](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)
2. **[private](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)****[void](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[doSomething() {}](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)

[The name of the pointcut expression is doSomething(). It will be applied on all the methods of Operation class regardless of return type.](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)

[Understanding Pointcut Expressions](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)

[Let's try the understand the pointcut expressions by the examples given below:](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)

1. [@Pointcut("execution(public \* \*(..))")](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)

[It will be applied on all the public methods.](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)

1. [@Pointcut("execution(public Operation.\*(..))")](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)

[It will be applied on all the public methods of Operation class.](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)

1. [@Pointcut("execution(\* Operation.\*(..))")](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)

[It will be applied on all the methods of Operation class.](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)

1. [@Pointcut("execution(public Employee.set\*(..))")](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)

[It will be applied on all the public setter methods of Employee class.](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)

1. [@Pointcut("execution(int Operation.\*(..))")](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)

[It will be applied on all the methods of Operation class that returns int value.](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)

[1) @Before Example](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)

[The AspectJ Before Advice is applied before the actual business logic method. You can perform any operation here such as conversion, authentication etc.](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)

[Create a class that contains actual business logic.](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)

*[File: Operation.java](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)*

1. **[package](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[com.javatpoint;](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)
2. **[public](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)****[class](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[Operation{](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)
3. **[public](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)****[void](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[msg(){System.out.println("msg method invoked");}](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)
4. **[public](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)****[int](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[m(){System.out.println("m method invoked");](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[return](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[2;}](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)
5. **[public](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)****[int](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[k(){System.out.println("k method invoked");](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[return](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[3;}](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)
6. [}](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)

[Now, create the aspect class that contains before advice.](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)

*[File: TrackOperation.java](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)*

1. **[package](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[com.javatpoint;](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)
3. **[import](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[org.aspectj.lang.JoinPoint;](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)
4. **[import](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[org.aspectj.lang.annotation.Aspect;](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)
5. **[import](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[org.aspectj.lang.annotation.Before;](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)
6. **[import](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[org.aspectj.lang.annotation.Pointcut;](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)
8. [@Aspect](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)
9. **[public](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)****[class](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[TrackOperation{](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)
10. [@Pointcut("execution(\* Operation.\*(..))")](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)
11. **[public](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)****[void](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[k(){}//pointcut name](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)
13. [@Before("k()")//applying pointcut on before advice](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)
14. **[public](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)****[void](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[myadvice(JoinPoint jp)//it is advice (before advice)](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)
15. [{](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)
16. [System.out.println("additional concern");](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)
17. [//System.out.println("Method Signature: "  + jp.getSignature());](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)
18. [}](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)
19. [}](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)

[Now create the applicationContext.xml file that defines beans.](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)

*[File: applicationContext.xml](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)*

1. [<?xml version="1.0" encoding="UTF-8"?>](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)
2. [<beans xmlns="http://www.springframework.org/schema/beans"](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)
3. [xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)
4. [xmlns:aop="http://www.springframework.org/schema/aop"](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)
5. [xsi:schemaLocation="http://www.springframework.org/schema/beans](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)
6. [http://www.springframework.org/schema/beans/spring-beans.xsd](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)
7. [http://www.springframework.org/schema/aop](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)
8. [http://www.springframework.org/schema/aop/spring-aop.xsd">](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)

11. [<bean id="opBean"](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[class](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[="com.javatpoint.Operation">   </bean>](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)
12. [<bean id="trackMyBean"](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[class](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[="com.javatpoint.TrackOperation"></bean>](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)
14. [<bean](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[class](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[="org.springframework.aop.aspectj.annotation.AnnotationAwareAspectJAutoProxyCreator"></bean>](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)
16. [</beans>](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)

[Now, let's call the actual method.](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)

*[File: Test.java](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)*

1. **[package](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[com.javatpoint;](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)
3. **[import](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[org.springframework.context.ApplicationContext;](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)
4. **[import](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[org.springframework.context.support.ClassPathXmlApplicationContext;](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)
5. **[public](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)****[class](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[Test{](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)
6. **[public](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)****[static](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)****[void](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[main(String[] args){](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)
7. [ApplicationContext context =](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[new](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[ClassPathXmlApplicationContext("applicationContext.xml");](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)
8. [Operation e = (Operation) context.getBean("opBean");](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)
9. [System.out.println("calling msg...");](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)
10. [e.msg();](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)
11. [System.out.println("calling m...");](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)
12. [e.m();](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)
13. [System.out.println("calling k...");](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)
14. [e.k();](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)
15. [}](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)
16. [}](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)

[Output](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)

1. [calling msg...](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)
2. [additional concern](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)
3. [msg() method invoked](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)
4. [calling m...](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)
5. [additional concern](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)
6. [m() method invoked](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)
7. [calling k...](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)
8. [additional concern](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)
9. [k() method invoked](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)

[As you can see, additional concern is printed before msg(), m() and k() method is invoked.](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)

[Now if you change the pointcut expression as given below:](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)

1. [@Pointcut("execution(\* Operation.m\*(..))")](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)

[Now additional concern will be applied for the methods starting with m in Operation class. Output will be as this:](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)

1. [calling msg...](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)
2. [additional concern](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)
3. [msg() method invoked](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)
4. [calling m...](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)
5. [additional concern](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)
6. [m() method invoked](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)
7. [calling k...](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)
8. [k() method invoked](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)

[Now you can see additional concern is not printed before k() method invoked.](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)

[2) @After Example](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)

[The AspectJ after advice is applied after calling the actual business logic methods. It can be used to maintain log, security, notification etc.](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)

[Here, We are assuming that](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[Operation.java](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[,](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[applicationContext.xml](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[and](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[Test.java](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[files are same as given in @Before example.](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)

[Create the aspect class that contains after advice.](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)

*[File: TrackOperation.java](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)*

1. **[package](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[com.javatpoint;](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)
3. **[import](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[org.aspectj.lang.JoinPoint;](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)
4. **[import](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[org.aspectj.lang.annotation.Aspect;](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)
5. **[import](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[org.aspectj.lang.annotation.After;](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)
6. **[import](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[org.aspectj.lang.annotation.Pointcut;](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)
8. [@Aspect](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)
9. **[public](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)****[class](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[TrackOperation{](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)
10. [@Pointcut("execution(\* Operation.\*(..))")](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)
11. **[public](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)****[void](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[k(){}//pointcut name](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)
13. [@After("k()")//applying pointcut on after advice](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)
14. **[public](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)****[void](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[myadvice(JoinPoint jp)//it is advice (after advice)](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)
15. [{](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)
16. [System.out.println("additional concern");](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)
17. [//System.out.println("Method Signature: "  + jp.getSignature());](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)
18. [}](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)
19. [}](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)

[Output](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)

1. [calling msg...](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)
2. [msg() method invoked](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)
3. [additional concern](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)
4. [calling m...](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)
5. [m() method invoked](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)
6. [additional concern](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)
7. [calling k...](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)
8. [k() method invoked](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)
9. [additional concern](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)

[You can see that additional concern is printed after calling msg(), m() and k() methods.](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)

[3) @AfterReturning Example](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)

[By using after returning advice, we can get the result in the advice.](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)

[Create the class that contains business logic.](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)

*[File: Operation.java](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)*

1. **[package](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[com.javatpoint;](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)
2. **[public](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)****[class](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[Operation{](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)
3. **[public](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)****[int](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[m(){System.out.println("m() method invoked");](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[return](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[2;}](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)
4. **[public](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)****[int](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[k(){System.out.println("k() method invoked");](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[return](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[3;}](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)
5. [}](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)

[Create the aspect class that contains after returning advice.](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)

*[File: TrackOperation.java](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)*

1. **[package](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[com.javatpoint;](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)
3. **[import](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[org.aspectj.lang.JoinPoint;](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)
4. **[import](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[org.aspectj.lang.annotation.AfterReturning;](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)
5. **[import](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[org.aspectj.lang.annotation.Aspect;](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)
7. [@Aspect](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)
8. **[public](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)****[class](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[TrackOperation{](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)
9. [@AfterReturning(](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)
10. [pointcut = "execution(\* Operation.\*(..))",](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)
11. [returning= "result")](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)
13. **[public](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)****[void](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[myadvice(JoinPoint jp,Object result)//it is advice (after returning advice)](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)
14. [{](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)
15. [System.out.println("additional concern");](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)
16. [System.out.println("Method Signature: "  + jp.getSignature());](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)
17. [System.out.println("Result in advice: "+result);](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)
18. [System.out.println("end of after returning advice...");](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)
19. [}](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)
20. [}](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)

*[File: applicationContext.xml](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)*

[It is same as given in @Before advice example](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)

*[File: Test.java](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)*

[Now create the Test class that calls the actual methods.](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)

1. **[package](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[com.javatpoint;](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)
3. **[import](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[org.springframework.context.ApplicationContext;](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)
4. **[import](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[org.springframework.context.support.ClassPathXmlApplicationContext;](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)
5. **[public](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)****[class](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[Test{](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)
6. **[public](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)****[static](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)****[void](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[main(String[] args){](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)
7. [ApplicationContext context =](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[new](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[ClassPathXmlApplicationContext("applicationContext.xml");](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)
8. [Operation e = (Operation) context.getBean("opBean");](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)
9. [System.out.println("calling m...");](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)
10. [System.out.println(e.m());](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)
11. [System.out.println("calling k...");](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)
12. [System.out.println(e.k());](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)
13. [}](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)
14. [}](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)

[Output](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)

1. [calling m...](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)
2. [m() method invoked](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)
3. [additional concern](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)
4. [Method Signature:](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[int](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[com.javatpoint.Operation.m()](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)
5. [Result in advice: 2](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)
6. [end of after returning advice...](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)
7. [2](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)
8. [calling k...](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)
9. [k() method invoked](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)
10. [additional concern](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)
11. [Method Signature:](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[int](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[com.javatpoint.Operation.k()](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)
12. [Result in advice: 3](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)
13. [end of after returning advice...](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)
14. [3](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)

[You can see that return value is printed two times, one is printed by TrackOperation class and second by Test class.](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)

[4) @Around Example](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)

[The AspectJ around advice is applied before and after calling the actual business logic methods.](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)

[Here, we are assuming that](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[applicationContext.xml](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[file is same as given in @Before example.](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)

[Create a class that contains actual business logic.](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)

*[File: Operation.java](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)*

1. **[package](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[com.javatpoint;](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)
2. **[public](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)****[class](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[Operation{](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)
3. **[public](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)****[void](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[msg(){System.out.println("msg() is invoked");}](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)
4. **[public](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)****[void](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[display(){System.out.println("display() is invoked");}](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)
5. [}](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)

[Create the aspect class that contains around advice.](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)

[You need to pass the](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[PreceedingJoinPoint](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[reference in the advice method, so that we can proceed the request by calling the proceed() method.](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)

*[File: TrackOperation.java](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)*

1. **[package](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[com.javatpoint;](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)
2. **[import](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[org.aspectj.lang.ProceedingJoinPoint;](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)
3. **[import](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[org.aspectj.lang.annotation.Around;](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)
4. **[import](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[org.aspectj.lang.annotation.Aspect;](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)
5. **[import](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[org.aspectj.lang.annotation.Pointcut;](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)
7. [@Aspect](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)
8. **[public](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)****[class](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[TrackOperation](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)
9. [{](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)
10. [@Pointcut("execution(\* Operation.\*(..))")](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)
11. **[public](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)****[void](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[abcPointcut(){}](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)
13. [@Around("abcPointcut()")](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)
14. **[public](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[Object myadvice(ProceedingJoinPoint pjp)](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[throws](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[Throwable](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)
15. [{](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)
16. [System.out.println("Additional Concern Before calling actual method");](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)
17. [Object obj=pjp.proceed();](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)
18. [System.out.println("Additional Concern After calling actual method");](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)
19. **[return](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[obj;](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)
20. [}](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)
21. [}](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)

*[File: Test.java](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)*

[Now create the Test class that calls the actual methods.](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)

1. **[package](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[com.javatpoint;](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)
2. **[import](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[org.springframework.context.ApplicationContext;](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)
3. **[import](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[org.springframework.context.support.ClassPathXmlApplicationContext;](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)
4. **[public](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)****[class](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[Test{](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)
5. **[public](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)****[static](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)****[void](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[main(String[] args){](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)
6. [ApplicationContext context =](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[new](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[classPathXmlApplicationContext("applicationContext.xml");](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)
8. [Operation op = (Operation) context.getBean("opBean");](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)
9. [op.msg();](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)
10. [op.display();](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)
11. [}](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)
12. [}](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)

[Output](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)

1. [Additional Concern Before calling actual method](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)
2. [msg() is invoked](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)
3. [Additional Concern After calling actual method](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)
4. [Additional Concern Before calling actual method](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)
5. [display() is invoked](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)
6. [Additional Concern After calling actual method](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)

[You can see that additional concern is printed before and after calling msg() and display methods.](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)

[5) @AfterThrowing Example](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)

[By using after throwing advice, we can print the exception in the TrackOperation class. Let's see the example of AspectJ AfterThrowing advice.](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)

[Create the class that contains business logic.](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)

*[File: Operation.java](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)*

1. **[package](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[com.javatpoint;](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)
2. **[public](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)****[class](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[Operation{](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)
3. **[public](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)****[void](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[validate(](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[int](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[age)](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[throws](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[Exception{](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)
4. **[if](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[(age<18){](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)
5. **[throw](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)****[new](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[ArithmeticException("Not valid age");](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)
6. [}](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)
7. **[else](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[{](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)
8. [System.out.println("Thanks for vote");](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)
9. [}](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)
10. [}](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)
12. [}](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)

[Create the aspect class that contains after throwing advice.](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)

[Here, we need to pass the Throwable reference also, so that we can intercept the exception here.](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)

*[File: TrackOperation.java](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)*

1. **[package](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[com.javatpoint;](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)
2. **[import](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[org.aspectj.lang.JoinPoint;](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)
3. **[import](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[org.aspectj.lang.annotation.AfterThrowing;](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)
4. **[import](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[org.aspectj.lang.annotation.Aspect;](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)
5. [@Aspect](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)
6. **[public](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)****[class](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[TrackOperation{](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)
7. [@AfterThrowing(](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)
8. [pointcut = "execution(\* Operation.\*(..))",](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)
9. [throwing= "error")](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)
11. **[public](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)****[void](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[myadvice(JoinPoint jp,Throwable error)//it is advice](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)
12. [{](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)
13. [System.out.println("additional concern");](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)
14. [System.out.println("Method Signature: "  + jp.getSignature());](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)
15. [System.out.println("Exception is: "+error);](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)
16. [System.out.println("end of after throwing advice...");](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)
17. [}](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)
18. [}](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)

*[File: applicationContext.xml](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)*

[It is same as given in @Before advice example](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)

*[File: Test.java](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)*

[Now create the Test class that calls the actual methods.](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)

1. **[package](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[com.javatpoint;](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)
3. **[import](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[org.springframework.context.ApplicationContext;](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)
4. **[import](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[org.springframework.context.support.ClassPathXmlApplicationContext;](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)
5. **[public](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)****[class](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[Test{](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)
6. **[public](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)****[static](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)****[void](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[main(String[] args){](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)
7. [ApplicationContext context =](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[new](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[ClassPathXmlApplicationContext("applicationContext.xml");](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)
8. [Operation op = (Operation) context.getBean("opBean");](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)
9. [System.out.println("calling validate...");](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)
10. **[try](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[{](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)
11. [op.validate(19);   }catch(Exception e){System.out.println(e);}](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)

# [Spring Expression Language (SPEL) Tutorial](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)

1. [[SPEL](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)](https://www.javatpoint.com/spring-expression-language-tutorial)
2. [[SPEL API](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)](https://www.javatpoint.com/spring-expression-language-tutorial)
3. [[Hello SPEL Example](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)](https://www.javatpoint.com/spring-expression-language-tutorial)
4. [[Other Examples of SPEL](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)](https://www.javatpoint.com/spring-expression-language-tutorial)

**[SpEL](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[is an exression language supporting the features of querying and manipulating an object graph at runtime.](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)

[There are many expression languages available such as JSP EL, OGNL, MVEL and JBoss EL. SpEL provides some additional features such as method invocation and string templating functionality.](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)

[SpEL API](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)

[The SpEL API provides many interfaces and classes. They are as follows:](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)

* [Expression interface](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)
* [SpelExpression class](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)
* [ExpressionParser interface](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)
* [SpelExpressionParser class](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)
* [EvaluationContext interface](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)
* [StandardEvaluationContext class](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)

[Hello SPEL Example](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)

1. **[import](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[org.springframework.expression.Expression;](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)
2. **[import](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[org.springframework.expression.ExpressionParser;](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)
3. **[import](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[org.springframework.expression.spel.standard.SpelExpressionParser;](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)
5. **[public](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)****[class](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[Test {](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)
6. **[public](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)****[static](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)****[void](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[main(String[] args) {](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)
7. [ExpressionParser parser =](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[new](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[SpelExpressionParser();](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)

[Operators in SPEL](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)

1. [[Examples of Operators in SPEL](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)](https://www.javatpoint.com/operators-in-spel)

[We can use many operators in SpEL such as arithmetic, relational, logical etc. There are given a lot of examples of using different operators in SpEL.](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)

[Examples of using operators in SPEL](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)

1. **[import](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[org.springframework.expression.ExpressionParser;](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)
2. **[import](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[org.springframework.expression.spel.standard.SpelExpressionParser;](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)

5. **[public](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)****[class](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[Test {](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)
6. **[public](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)****[static](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)****[void](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[main(String[] args) {](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)
7. [ExpressionParser parser =](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[new](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[SpelExpressionParser();](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)
9. [//arithmetic operator](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)
10. [System.out.println(parser.parseExpression("'Welcome SPEL'+'!'").getValue());](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)
11. [System.out.println(parser.parseExpression("10 \* 10/2").getValue());](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)
12. [System.out.println(parser.parseExpression("'Today is: '+ new java.util.Date()").getValue());](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)
14. [//logical operator](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)
15. [System.out.println(parser.parseExpression("true and true").getValue());](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)
17. [//Relational operator](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)
18. [System.out.println(parser.parseExpression("'sonoo'.length()==5").getValue());](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)
19. [}](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)
20. [}](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)
22. [Expression exp = parser.parseExpression("'Hello SPEL'");](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)
23. [String message = (String) exp.getValue();](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)
24. [System.out.println(message);](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)
25. [//OR](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)
26. [//System.out.println(parser.parseExpression("'Hello SPEL'").getValue());](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)
27. [}](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)
28. [}](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)

[Other SPEL Example](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)

[Let's see a lot of useful examples of SPEL. Here, we are assuming all the examples have been written inside the main() method.](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)

[OOPs Concepts in Java](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)

[Using concat() method with String](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)

1. [ExpressionParser parser =](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[new](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[SpelExpressionParser();](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)
2. [Expression exp = parser.parseExpression("'Welcome SPEL'.concat('!')");](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)
3. [String message = (String) exp.getValue();](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)
4. [System.out.println(message);](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)

[Converting String into byte array](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)

1. [Expression exp = parser.parseExpression("'Hello World'.bytes");](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)
2. **[byte](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[[] bytes = (](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[byte](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[[]) exp.getValue();](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)
3. **[for](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[(](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[int](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[i=0;i<bytes.length;i++){](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)
4. [System.out.print(bytes[i]+" ");](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)
5. [}](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)

[Getting length after converting string into bytes](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)

1. [Expression exp = parser.parseExpression("'Hello World'.bytes.length");](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)
2. **[int](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[length = (Integer) exp.getValue();](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)
3. [System.out.println(length);](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)

[Converting String contents into uppercase letter](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)

1. [Expression exp = parser.parseExpression("new String('hello world').toUpperCase()");](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)
2. [String message = exp.getValue(String.](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[class](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[);](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)
3. [System.out.println(message);](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)
4. [//OR](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)
5. [System.out.println(parser.parseExpression("'hello world'.toUpperCase()").getValue());](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)
6. [System.out.println("calling validate again...");](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)
8. **[try](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[{](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)
9. [op.validate(11);](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)
10. [}](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[catch](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[(Exception e){System.out.println(e);}](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)
11. [}](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)
12. [}](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)

[Output](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)

1. [calling validate...](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)
2. [Thanks](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[for](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[vote](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)
3. [calling validate again...](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)
4. [additional concern](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)
5. [Method Signature:](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[void](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[com.javatpoint.Operation.validate(](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[int](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[)](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)
6. [Exception is: java.lang.ArithmeticException: Not valid age](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)
7. [end of after throwing advice...](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)
8. [java.lang.ArithmeticException: Not valid age](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)

# [Variable in SPEL | StandardEvaluationContext](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)

1. [[Using Variable in SPEL](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)](https://www.javatpoint.com/using-variable-in-spel)
2. [[StandardEvaluationContext class](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)](https://www.javatpoint.com/using-variable-in-spel)
3. [[Example of Using variable in SPEL](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)](https://www.javatpoint.com/using-variable-in-spel)

[In SpEL, we can store a value in the variable and use the variable in the method and call the method. To work on variable, we need to use](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[StandardEvaluationContext](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[class.](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)

[Example of Using variable in SPEL](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)

**[Calculation.java](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**

1. **[public](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)****[class](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[Calculation {](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)
2. **[private](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)****[int](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[number;](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)
3. **[public](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)****[int](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[getNumber() {](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)
4. **[return](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[number;](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)
5. [}](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)
6. **[public](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)****[void](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[setNumber(](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[int](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[number) {](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)
7. **[this](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[.number = number;](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)
8. [}](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)
9. **[public](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)****[int](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[cube(){](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)
10. **[return](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[number\*number\*number;](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)
11. [}](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)
12. [}](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)

**[Test.java](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**

1. **[import](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[org.springframework.expression.ExpressionParser;](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)
2. **[import](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[org.springframework.expression.spel.standard.SpelExpressionParser;](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)
3. **[import](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[org.springframework.expression.spel.support.StandardEvaluationContext;](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)
5. **[public](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)****[class](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[Test {](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)
6. **[public](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)****[static](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)****[void](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[main(String[] args) {](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)
7. [Calculation calculation=](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[new](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[Calculation();](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)
8. [StandardEvaluationContext context=](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[new](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[StandardEvaluationContext(calculation);](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)
10. [ExpressionParser parser =](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[new](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[SpelExpressionParser();](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)
11. [parser.parseExpression("number").setValue(context,"5");](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)
13. [System.out.println(calculation.cube());](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)
14. [}](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)
15. [}](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)

[Spring MVC Tutorial](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)

[A Spring MVC is a Java framework which is used to build web applications. It follows the Model-View-Controller design pattern. It implements all the basic features of a core spring framework like Inversion of Control, Dependency Injection.](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)

[A Spring MVC provides an elegant solution to use MVC in spring framework by the help of](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[DispatcherServlet](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[. Here,](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[DispatcherServlet](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[is a class that receives the incoming request and maps it to the right resource such as controllers, models, and views.](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)

[Spring Web Model-View-Controller](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)

[](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)

* **[Model](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[- A model contains the data of the application. A data can be a single object or a collection of objects.](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)
* **[Controller](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[- A controller contains the business logic of an application. Here, the @Controller annotation is used to mark the class as the controller.](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)
* **[View](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[- A view represents the provided information in a particular format. Generally, JSP+JSTL is used to create a view page. Although spring also supports other view technologies such as Apache Velocity, Thymeleaf and FreeMarker.](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)
* **[Front Controller](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[- In Spring Web MVC, the DispatcherServlet class works as the front controller. It is responsible to manage the flow of the Spring MVC application.](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)

[Understanding the flow of Spring Web MVC](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)

[](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)

* [As displayed in the figure, all the incoming request is intercepted by the DispatcherServlet that works as the front controller.](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)
* [The DispatcherServlet gets an entry of handler mapping from the XML file and forwards the request to the controller.](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)
* [The controller returns an object of ModelAndView.](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)
* [The DispatcherServlet checks the entry of view resolver in the XML file and invokes the specified view component.](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)

[Advantages of Spring MVC Framework](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)

[Let's see some of the advantages of Spring MVC Framework:-](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)

* **[Separate roles](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[- The Spring MVC separates each role, where the model object, controller, command object, view resolver, DispatcherServlet, validator, etc. can be fulfilled by a specialized object.](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)
* **[Light-weight](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[- It uses light-weight servlet container to develop and deploy your application.](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)
* **[Powerful Configuration](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[- It provides a robust configuration for both framework and application classes that includes easy referencing across contexts, such as from web controllers to business objects and validators.](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)
* **[Rapid development](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[- The Spring MVC facilitates fast and parallel development.](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)
* **[Reusable business code](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[- Instead of creating new objects, it allows us to use the existing business objects.](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)
* **[Easy to test](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[- In Spring, generally we create JavaBeans classes that enable you to inject test data using the setter methods.](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)
* **[Flexible Mapping](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[- It provides the specific annotations that easily redirect the page.](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)

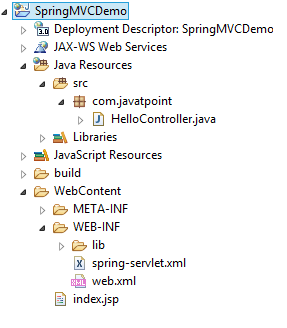
[Spring Web MVC Framework Example](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)

[Let's see the simple example of a Spring Web MVC framework. The steps are as follows:](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)

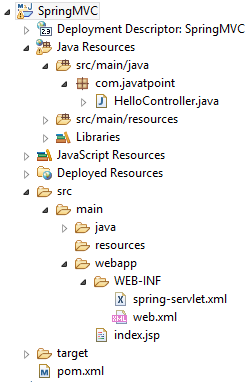
[Exception Handling in Java - Javatpoint](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)

* [Load the spring jar files or add dependencies in the case of Maven](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)
* [Create the controller class](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)
* [Provide the entry of controller in the web.xml file](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)
* [Define the bean in the separate XML file](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)
* [Display the message in the JSP page](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)
* [Start the server and deploy the project](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)

[Directory Structure of Spring MVC](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)

[](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)

[Directory Structure of Spring MVC using Maven](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)

[](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)

[Required Jar files or Maven Dependency](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)

[To run this example, you need to load:](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)

* [Spring Core jar files](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)
* [Spring Web jar files](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)
* [JSP + JSTL jar files (If you are using any another view technology then load the corresponding jar files).](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)

**[Download Link:](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[[Download all the jar files for spring including JSP and JSTL](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)](https://static.javatpoint.com/src/sp/springjars.zip)[.](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)

[If you are using Maven, you don't need to add jar files. Now, you need to add maven dependency to the pom.xml file.](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)

[1. Provide project information and configuration in the pom.xml file.](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)

**[pom.xml](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**

1. **[<project](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[xmlns="http://maven.apache.org/POM/4.0.0" xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)
2. [xsi:schemaLocation="http://maven.apache.org/POM/4.0.0 http://maven.apache.org/maven-v4\_0\_0.xsd"](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[>](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**
3. **[<modelVersion>](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[4.0.0](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[</modelVersion>](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**
4. **[<groupId>](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[com.javatpoint](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[</groupId>](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**
5. **[<artifactId>](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[SpringMVC](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[</artifactId>](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**
6. **[<packaging>](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[war](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[</packaging>](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**
7. **[<version>](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[0.0.1-SNAPSHOT](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[</version>](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**
8. **[<name>](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[SpringMVC Maven Webapp](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[</name>](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**
9. **[<url>](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[http://maven.apache.org](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[</url>](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**
10. **[<dependencies>](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**
11. **[<dependency>](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**
12. **[<groupId>](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[junit](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[</groupId>](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**
13. **[<artifactId>](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[junit](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[</artifactId>](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**
14. **[<version>](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[3.8.1](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[</version>](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**
15. **[<scope>](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[test](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[</scope>](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**
16. **[</dependency>](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**
18. [<!-- https://mvnrepository.com/artifact/org.springframework/spring-webmvc -->](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)
19. **[<dependency>](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**
20. **[<groupId>](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[org.springframework](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[</groupId>](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**
21. **[<artifactId>](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[spring-webmvc](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[</artifactId>](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**
22. **[<version>](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[5.1.1.RELEASE](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[</version>](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**
23. **[</dependency>](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**
25. [<!-- https://mvnrepository.com/artifact/javax.servlet/javax.servlet-api -->](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)
26. **[<dependency>](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**
27. **[<groupId>](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[javax.servlet](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[</groupId>](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**
28. **[<artifactId>](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[servlet-api](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[</artifactId>](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**
29. **[<version>](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[3.0-alpha-1](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[</version>](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**
30. **[</dependency>](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**
32. **[</dependencies>](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**
33. **[<build>](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**
34. **[<finalName>](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[SpringMVC](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[</finalName>](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**
35. **[</build>](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**
36. **[</project>](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**

[2. Create the controller class](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)

[To create the controller class, we are using two annotations @Controller and @RequestMapping.](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)

[The @Controller annotation marks this class as Controller.](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)

[The @Requestmapping annotation is used to map the class with the specified URL name.](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)

**[HelloController.java](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**

1. **[package](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[com.javatpoint;](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)
2. **[import](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[org.springframework.stereotype.Controller;](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)
3. **[import](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[org.springframework.web.bind.annotation.RequestMapping;](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)
4. [@Controller](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)
5. **[public](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)****[class](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[HelloController {](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)
6. [@RequestMapping("/")](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)
7. **[public](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[String display()](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)
8. [{](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)
9. **[return](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**["index";](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)
10. [}](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)
11. [}](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)

[3. Provide the entry of controller in the web.xml file](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)

[In this xml file, we are specifying the servlet class DispatcherServlet that acts as the front controller in Spring Web MVC. All the incoming request for the html file will be forwarded to the DispatcherServlet.](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)

**[web.xml](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**

1. **[<?xml](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[version="1.0" encoding="UTF-8"](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[?>](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**
2. **[<web-app](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" xmlns="http://java.sun.com/xml/ns/javaee" xsi:schemaLocation="http://java.sun.com/xml/ns/javaee http://java.sun.com/xml/ns/javaee/web-app\_3\_0.xsd" id="WebApp\_ID" version="3.0"](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[>](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**
3. **[<display-name>](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[SpringMVC](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[</display-name>](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**
4. **[<servlet>](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**
5. **[<servlet-name>](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[spring](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[</servlet-name>](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**
6. **[<servlet-class>](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[org.springframework.web.servlet.DispatcherServlet](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[</servlet-class>](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**
7. **[<load-on-startup>](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[1](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[</load-on-startup>](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**
8. **[</servlet>](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**
9. **[<servlet-mapping>](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**
10. **[<servlet-name>](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[spring](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[</servlet-name>](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**
11. **[<url-pattern>](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[/](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[</url-pattern>](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**
12. **[</servlet-mapping>](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**
13. **[</web-app>](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**

[4. Define the bean in the xml file](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)

[This is the important configuration file where we need to specify the View components.](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)

[The context:component-scan element defines the base-package where DispatcherServlet will search the controller class.](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)

[This xml file should be located inside the WEB-INF directory.](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)

**[spring-servlet.xml](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**

1. **[<?xml](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[version="1.0" encoding="UTF-8"](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[?>](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**
2. **[<beans](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[xmlns="http://www.springframework.org/schema/beans"](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)
3. [xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)
4. [xmlns:context="http://www.springframework.org/schema/context"](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)
5. [xmlns:mvc="http://www.springframework.org/schema/mvc"](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)
6. [xsi:schemaLocation="](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)
7. [http://www.springframework.org/schema/beans](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)
8. [http://www.springframework.org/schema/beans/spring-beans.xsd](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)
9. [http://www.springframework.org/schema/context](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)
10. [http://www.springframework.org/schema/context/spring-context.xsd](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)
11. [http://www.springframework.org/schema/mvc](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)
12. [http://www.springframework.org/schema/mvc/spring-mvc.xsd"](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[>](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**
14. [<!-- Provide support for component scanning -->](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)
15. **[<context:component-scan](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[base-package="com.javatpoint"](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[/>](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**
17. [<!--Provide support for conversion, formatting and validation -->](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)
18. **[<mvc:annotation-driven/>](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**
20. **[</beans>](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**

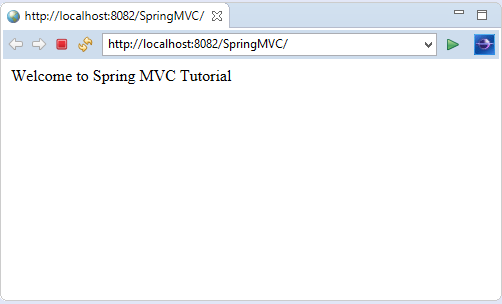
[5. Display the message in the JSP page](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)

[This is the simple JSP page, displaying the message returned by the Controller.](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)

**[index.jsp](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**

1. **[<html>](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**
2. **[<body>](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**
3. **[<p>](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[Welcome to Spring MVC Tutorial](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[</p>](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**
4. **[</body>](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**
5. **[</html>](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**

**[Output:](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**

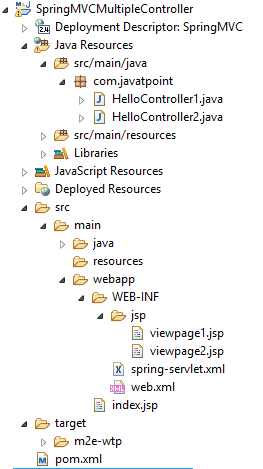
[](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)

[Spring MVC Multiple Controller Example](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)

[In Spring MVC, we can create multiple controllers at a time. It is required to map each controller class with](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[@Controller](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[annotation. Here, we see a Spring MVC example of multiple controllers. The steps are as follows:](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)

* [Load the spring jar files or add dependencies in the case of Maven](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)
* [Create the controller class](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)
* [Provide the entry of controller in the web.xml file](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)
* [Define the bean in the separate XML file](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)
* [Create the other view components](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)
* [Start the server and deploy the project](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)

[Directory Structure of Spring MVC](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)

[](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)

[1. Add dependencies to pom.xml](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)

1. [<!-- https://mvnrepository.com/artifact/org.springframework/spring-webmvc -->](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)
2. **[<dependency>](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**
3. **[<groupId>](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[org.springframework](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[</groupId>](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**
4. **[<artifactId>](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[spring-webmvc](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[</artifactId>](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**
5. **[<version>](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[5.1.1.RELEASE](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[</version>](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**
6. **[</dependency>](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**
8. [<!-- https://mvnrepository.com/artifact/javax.servlet/javax.servlet-api -->](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)
9. **[<dependency>](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**
10. **[<groupId>](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[javax.servlet](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[</groupId>](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**
11. **[<artifactId>](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[servlet-api](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[</artifactId>](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**
12. **[<version>](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[3.0-alpha-1](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[</version>](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**
13. **[</dependency>](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**

[2. Create the request page](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)

[Let's create a simple JSP page containing two links.](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)

**[index.jsp](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**

1. **[<html>](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**
2. **[<body>](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**
3. **[<a](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[href="hello1"](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[>](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[Spring MVC](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[</a>](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[||](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)
4. **[<a](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[href="hello2"](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[>](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[Spring Boot](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[</a>](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**
5. **[</body>](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**
6. **[</html>](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**

[3. Create the controller class](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)

[Let's create two controller classes, where each returns the particular view page.](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)

[Exception Handling in Java - Javatpoint](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)

**[HelloController1.java](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**

1. [package com.javatpoint;](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)
2. [import org.springframework.stereotype.Controller;](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)
3. [import org.springframework.web.bind.annotation.RequestMapping;](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)
4. [@Controller](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)
5. [public class HelloController1 {](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)
6. [@RequestMapping("/hello1")](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)
7. [public String display()](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)
8. [{](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)
9. [return "viewpage1";](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)
10. [}](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)
11. [}](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)

**[HelloController2.java](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**

1. [package com.javatpoint;](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)
2. [import org.springframework.stereotype.Controller;](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)
3. [import org.springframework.web.bind.annotation.RequestMapping;](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)
4. [@Controller](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)
5. [public class HelloController2 {](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)
6. [@RequestMapping("/hello2")](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)
7. [public String display()](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)
8. [{](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)
9. [return "viewpage2";](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)
10. [}](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)
11. [}](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)

[4. Provide the entry of controller in the web.xml file](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)

**[web.xml](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**

1. **[<?xml](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[version="1.0" encoding="UTF-8"](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[?>](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**
2. **[<web-app](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" xmlns="http://java.sun.com/xml/ns/javaee" xsi:schemaLocation="http://java.sun.com/xml/ns/javaee http://java.sun.com/xml/ns/javaee/web-app\_3\_0.xsd" id="WebApp\_ID" version="3.0"](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[>](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**
3. **[<display-name>](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[SpringMVC](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[</display-name>](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**
4. **[<servlet>](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**
5. **[<servlet-name>](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[spring](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[</servlet-name>](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**
6. **[<servlet-class>](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[org.springframework.web.servlet.DispatcherServlet](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[</servlet-class>](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**
7. **[<load-on-startup>](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[1](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[</load-on-startup>](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**
8. **[</servlet>](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**
9. **[<servlet-mapping>](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**
10. **[<servlet-name>](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[spring](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[</servlet-name>](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**
11. **[<url-pattern>](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[/](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[</url-pattern>](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**
12. **[</servlet-mapping>](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**
13. **[</web-app>](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**

[5. Define the bean in the xml file](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)

**[spring-servlet.xml](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**

1. **[<?xml](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[version="1.0" encoding="UTF-8"](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[?>](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**
2. **[<beans](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[xmlns="http://www.springframework.org/schema/beans"](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)
3. [xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)
4. [xmlns:context="http://www.springframework.org/schema/context"](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)
5. [xmlns:mvc="http://www.springframework.org/schema/mvc"](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)
6. [xsi:schemaLocation="](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)
7. [http://www.springframework.org/schema/beans](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)
8. [http://www.springframework.org/schema/beans/spring-beans.xsd](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)
9. [http://www.springframework.org/schema/context](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)
10. [http://www.springframework.org/schema/context/spring-context.xsd](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)
11. [http://www.springframework.org/schema/mvc](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)
12. [http://www.springframework.org/schema/mvc/spring-mvc.xsd"](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[>](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**
14. [<!-- Provide support for component scanning -->](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)
15. **[<context:component-scan](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[base-package="com.javatpoint"](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[/>](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**
17. [<!--Provide support for conversion, formatting and validation -->](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)
18. **[<mvc:annotation-driven/>](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**
19. **[<bean](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[id="viewResolver" class="org.springframework.web.servlet.view.InternalResourceViewResolver"](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[>](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**
20. **[<property](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[name="prefix" value="/WEB-INF/jsp/"](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[></property>](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**
21. **[<property](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[name="suffix" value=".jsp"](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[></property>](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**
22. **[</bean>](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**
23. **[</beans>](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**

[6. Create the other view components](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)

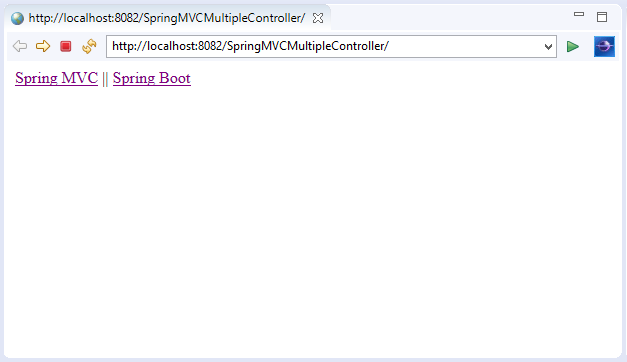
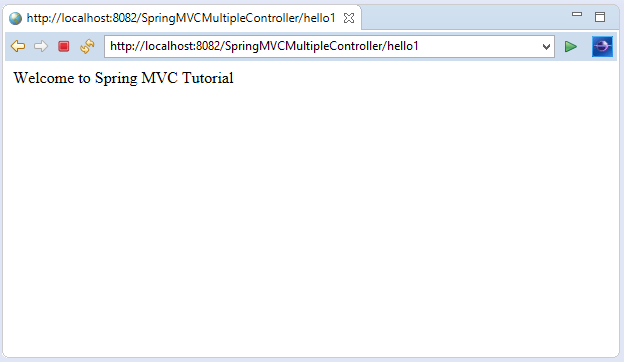
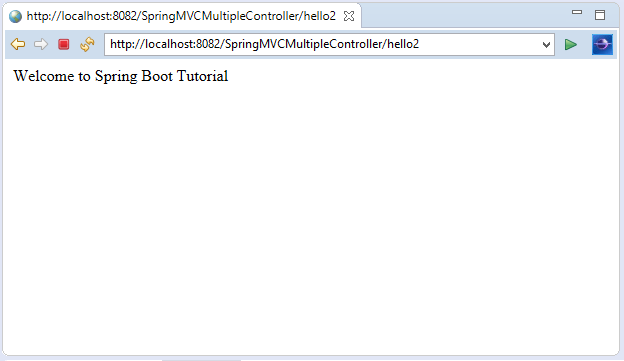
**[viewpage1.jsp](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**

1. **[<html>](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**
2. **[<body>](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**
3. **[<p>](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[Welcome to Spring MVC Tutorial](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[</p>](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**
4. **[</body>](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**
5. **[</html>](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**

**[viewpage1.jsp](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**

1. **[<html>](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**
2. **[<body>](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**
3. **[<p>](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[Welcome to Spring Boot Tutorial](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[</p>](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**
4. **[</body>](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**
5. **[</html>](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**

**[Output:](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**

[  
  
In Spring MVC, the model works a container that contains the data of the application. Here, a data can be in any form such as objects, strings, information from the database, etc.](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)

[It is required to place the](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[Model](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[interface in the controller part of the application. The object of](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[HttpServletRequest](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[reads the information provided by the user and pass it to the](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[Model](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[interface. Now, a view page easily accesses the data from the model part.](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)

[Methods of Model Interface](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)

|  |  |
| --- | --- |
| **[Method](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)** | **[Description](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)** |
| [Model addAllAttributes(Collection<?> arg)](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example) | [It adds all the attributes in the provided Collection into this Map.](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example) |
| [Model addAllAttributes(Map<String,?> arg)](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example) | [It adds all the attributes in the provided Map into this Map.](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example) |
| [Model addAllAttribute(Object arg)](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example) | [It adds the provided attribute to this Map using a generated name.](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example) |
| [Model addAllAttribute(String arg0, Object arg1)](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example) | [It binds the attribute with the provided name.](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example) |
| [Map<String, Object> asMap()](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example) | [It return the current set of model attributes as a Map.](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example) |
| [Model mergeAttributes(Map< String,?> arg)](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example) | [It adds all attributes in the provided Map into this Map, with existing objects of the same name taking precedence.](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example) |
| [boolean containsAttribute(String arg)](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example) | [It indicates whether this model contains an attribute of the given name](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example) |

[Spring MVC Model Example](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)

[Let's create a login page that contains a username and password. Here, we validate the password with a specific value.](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)

[1. Add dependencies to pom.xml](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)

1. [<!-- https://mvnrepository.com/artifact/org.springframework/spring-webmvc -->](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)
2. **[<dependency>](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**
3. **[<groupId>](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[org.springframework](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[</groupId>](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**
4. **[<artifactId>](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[spring-webmvc](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[</artifactId>](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**
5. **[<version>](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[5.1.1.RELEASE](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[</version>](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**
6. **[</dependency>](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**
8. [<!-- https://mvnrepository.com/artifact/javax.servlet/javax.servlet-api -->](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)
9. **[<dependency>](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**
10. **[<groupId>](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[javax.servlet](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[</groupId>](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**
11. **[<artifactId>](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[servlet-api](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[</artifactId>](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**
12. **[<version>](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[3.0-alpha-1](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[</version>](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**
13. **[</dependency>](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**

[2. Create the request page](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)

[Here, we create the login page to receive name and password from the user.](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)

[Prime Ministers of India | List of Prime Minister of India (1947-2020)](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)

**[index.jsp](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**

1. **[<html>](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**
2. **[<body>](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**
3. **[<form](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[action="hello"](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[>](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**
4. [UserName :](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[<input](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[type="text" name="name"](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[/>](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)****[<br><br>](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**
5. [Password :](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[<input](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[type="text" name="pass"](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[/>](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)****[<br><br>](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**
6. **[<input](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[type="submit" name="submit"](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[>](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**
7. **[</form>](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**
8. **[</body>](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**
9. **[</html>](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**

[3. Create the controller class](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)

[In controller class:](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)

* [The](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[HttpServletRequest](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[is used to read the HTML form data provided by the user.](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)
* [The](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[Model](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[contains the request data and provides it to view page.](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)

**[HelloController.java](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**

1. **[package](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[com.javatpoint;](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)
2. **[import](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[javax.servlet.http.HttpServletRequest;](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)
3. **[import](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[org.springframework.stereotype.Controller;](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)
4. **[import](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[org.springframework.ui.Model;](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)
5. **[import](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[org.springframework.web.bind.annotation.RequestMapping;](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)
7. [@Controller](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)
8. **[public](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)****[class](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[HelloController {](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)
10. [@RequestMapping("/hello")](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)
11. **[public](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[String display(HttpServletRequest req,Model m)](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)
12. [{](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)
13. [//read the provided form data](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)
14. [String name=req.getParameter("name");](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)
15. [String pass=req.getParameter("pass");](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)
16. **[if](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[(pass.equals("admin"))](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)
17. [{](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)
18. [String msg="Hello "+ name;](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)
19. [//add a message to the model](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)
20. [m.addAttribute("message", msg);](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)
21. **[return](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**["viewpage";](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)
22. [}](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)
23. **[else](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**
24. [{](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)
25. [String msg="Sorry "+ name+". You entered an incorrect password";](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)
26. [m.addAttribute("message", msg);](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)
27. **[return](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**["errorpage";](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)
28. [}](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)
29. [}](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)
30. [}](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)

[4. Provide the entry of controller in the web.xml file](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)

**[web.xml](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**

1. **[<?xml](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[version="1.0" encoding="UTF-8"](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[?>](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**
2. **[<web-app](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" xmlns="http://java.sun.com/xml/ns/javaee" xsi:schemaLocation="http://java.sun.com/xml/ns/javaee http://java.sun.com/xml/ns/javaee/web-app\_3\_0.xsd" id="WebApp\_ID" version="3.0"](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[>](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**
3. **[<display-name>](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[SpringMVC](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[</display-name>](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**
4. **[<servlet>](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**
5. **[<servlet-name>](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[spring](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[</servlet-name>](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**
6. **[<servlet-class>](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[org.springframework.web.servlet.DispatcherServlet](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[</servlet-class>](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**
7. **[<load-on-startup>](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[1](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[</load-on-startup>](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**
8. **[</servlet>](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**
9. **[<servlet-mapping>](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**
10. **[<servlet-name>](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[spring](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[</servlet-name>](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**
11. **[<url-pattern>](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[/](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[</url-pattern>](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**
12. **[</servlet-mapping>](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**
13. **[</web-app>](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**

[5. Define the bean in the xml file](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)

**[spring-servlet.xml](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**

1. **[<?xml](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[version="1.0" encoding="UTF-8"](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[?>](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**
2. **[<beans](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[xmlns="http://www.springframework.org/schema/beans"](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)
3. [xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)
4. [xmlns:context="http://www.springframework.org/schema/context"](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)
5. [xmlns:mvc="http://www.springframework.org/schema/mvc"](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)
6. [xsi:schemaLocation="](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)
7. [http://www.springframework.org/schema/beans](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)
8. [http://www.springframework.org/schema/beans/spring-beans.xsd](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)
9. [http://www.springframework.org/schema/context](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)
10. [http://www.springframework.org/schema/context/spring-context.xsd](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)
11. [http://www.springframework.org/schema/mvc](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)
12. [http://www.springframework.org/schema/mvc/spring-mvc.xsd"](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[>](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**
14. [<!-- Provide support for component scanning -->](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)
15. **[<context:component-scan](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[base-package="com.javatpoint"](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[/>](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**
17. [<!--Provide support for conversion, formatting and validation -->](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)
18. **[<mvc:annotation-driven/>](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**
19. **[<bean](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[id="viewResolver" class="org.springframework.web.servlet.view.InternalResourceViewResolver"](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[>](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**
20. **[<property](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[name="prefix" value="/WEB-INF/jsp/"](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[></property>](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**
21. **[<property](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[name="suffix" value=".jsp"](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[></property>](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**
22. **[</bean>](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**
23. **[</beans>](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**

[6. Create the other view components](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)

[To run this example, the following view components must be located inside the WEB-INF/jsp directory.](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)

**[viewpage.jsp](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**

1. **[<html>](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**
2. **[<body>](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**
3. [${message}](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)
4. **[</body>](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**
5. **[</html>](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**

**[errorpage.jsp](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**

1. **[<html>](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**
2. **[<body>](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**
3. [${message}](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)
4. **[<br><br>](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**
5. **[<jsp:include](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[page="/index.jsp"](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[></jsp:include>](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**
6. **[</body>](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**
7. **[</html>](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**

**[Output:](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**

# [Spring MVC Model Interface Spring MVC Model Interface Spring MVC Model Interface Spring MVC Model Interface Spring MVC Model InterfaceSpring MVC RequestParam Annotation](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)

[In Spring MVC, the](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[@RequestParam](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[annotation is used to read the form data and bind it automatically to the parameter present in the provided method. So, it ignores the requirement of](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[HttpServletRequest](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[object to read the provided data.](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)

[Including form data, it also maps the request parameter to query parameter and parts in multipart requests. If the method parameter type is Map and a request parameter name is specified, then the request parameter value is converted to a Map else the map parameter is populated with all request parameter names and values.](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)

[Spring MVC RequestParam Example](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)

[Let's create a login page that contains a username and password. Here, we validate the password with a specific value.](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)

[1. Add dependencies to pom.xml](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)

1. [<!-- https://mvnrepository.com/artifact/org.springframework/spring-webmvc -->](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)
2. **[<dependency>](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**
3. **[<groupId>](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[org.springframework](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[</groupId>](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**
4. **[<artifactId>](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[spring-webmvc](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[</artifactId>](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**
5. **[<version>](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[5.1.1.RELEASE](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[</version>](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**
6. **[</dependency>](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**
8. [<!-- https://mvnrepository.com/artifact/javax.servlet/javax.servlet-api -->](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)
9. **[<dependency>](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**
10. **[<groupId>](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[javax.servlet](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[</groupId>](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**
11. **[<artifactId>](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[servlet-api](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[</artifactId>](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**
12. **[<version>](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[3.0-alpha-1](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[</version>](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**
13. **[</dependency>](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**

[2. Create the request page](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)

[It is the login page that receive name and password from the user.](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)

[Exception Handling in Java - Javatpoint](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)

**[index.jsp](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**

1. **[<html>](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**
2. **[<body>](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**
3. **[<form](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[action="hello"](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[>](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**
4. [UserName :](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[<input](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[type="text" name="name"](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[/>](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)****[<br><br>](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**
5. [Password :](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[<input](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[type="text" name="pass"](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[/>](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)****[<br><br>](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**
6. **[<input](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[type="submit" name="submit"](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[>](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**
7. **[</form>](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**
8. **[</body>](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**
9. **[</html>](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**

[3. Create the Controller Class](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)

[In controller class:](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)

* [The @RequestParam is used to read the HTML form data provided by a user and bind it to the request parameter.](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)
* [The Model contains the request data and provides it to view page.](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)

**[HelloController.java](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**

1. **[package](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[com.javatpoint;](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)
2. **[import](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[org.springframework.stereotype.Controller;](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)
3. **[import](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[org.springframework.ui.Model;](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)
4. **[import](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[org.springframework.web.bind.annotation.RequestMapping;](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)
5. **[import](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[org.springframework.web.bind.annotation.RequestParam;](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)
7. [@Controller](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)
8. **[public](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)****[class](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[HelloController {](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)
10. [@RequestMapping("/hello")](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)
11. [//read the provided form data](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)
12. **[public](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[String display(@RequestParam("name") String name,@RequestParam("pass") String pass,Model m)](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)
13. [{](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)
14. **[if](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[(pass.equals("admin"))](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)
15. [{](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)
16. [String msg="Hello "+ name;](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)
17. [//add a message to the model](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)
18. [m.addAttribute("message", msg);](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)
19. **[return](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**["viewpage";](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)
20. [}](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)
21. **[else](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**
22. [{](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)
23. [String msg="Sorry "+ name+". You entered an incorrect password";](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)
24. [m.addAttribute("message", msg);](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)
25. **[return](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**["errorpage";](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)
26. [}](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)
27. [}](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)
28. [}](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)

[4. Create the other view components](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)

[To run this example, the following view components must be located inside the WEB-INF/jsp directory.](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)

**[viewpage.jsp](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**

1. **[<html>](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**
2. **[<body>](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**
3. [${message}](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)
4. **[</body>](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**
5. **[</html>](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**

**[errorpage.jsp](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**

1. **[<html>](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**
2. **[<body>](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**
3. [${message}](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)
4. **[<br><br>](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**
5. **[<jsp:include](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[page="/index.jsp"](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[></jsp:include>](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**
6. **[</body>](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**
7. **[</html>](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**
8. **[</html>](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**

**[Output:](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**

# [Spring MVC RequestParam Annotation Spring MVC RequestParam Annotation Spring MVC RequestParam Annotation Spring MVC RequestParam Annotation Spring MVC RequestParam AnnotationSpring MVC Form Tag Library](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)

[The Spring MVC form tags are the configurable and reusable building blocks for a web page. These tags provide JSP, an easy way to develop, read and maintain.](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)

[The Spring MVC form tags can be seen as data binding-aware tags that can automatically set data to Java object/bean and also retrieve from it. Here, each tag provides support for the set of attributes of its corresponding HTML tag counterpart, making the tags familiar and easy to use.](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)

[Configuration of Spring MVC Form Tag](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)

[The form tag library comes under the spring-webmvc.jar. To enable the support for form tag library, it is required to reference some configuration. So, add the following directive at the beginning of the JSP page:](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)

1. **[<](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[%@ taglib prefix="form" uri="http://www.springframework.org/tags/form" %](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[>](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**

[List of Spring MVC Form Tags](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)

[Let's see some of the frequently used Spring MVC form tags.](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)

[33.1M](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)

[751](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)

[Features of Java - Javatpoint](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)

|  |  |
| --- | --- |
| **[Form Tag](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)** | **[Description](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)** |
| [form:form](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example) | [It is a container tag that contains all other form tags.](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example) |
| [form:input](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example) | [This tag is used to generate the text field.](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example) |
| [form:radiobutton](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example) | [This tag is used to generate the radio buttons.](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example) |
| [form:checkbox](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example) | [This tag is used to generate the checkboxes.](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example) |
| [form:password](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example) | [This tag is used to generate the password input field.](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example) |
| [form:select](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example) | [This tag is used to generate the drop-down list.](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example) |
| [form:textarea](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example) | [This tag is used to generate the multi-line text field.](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example) |
| [form:hidden](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example) | [This tag is used to generate the hidden input field.](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example) |

[The form tag](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)

[The Spring MVC form tag is a container tag. It is a parent tag that contains all the other tags of the tag library. This tag generates an HTML form tag and exposes a binding path to the inner tags for binding.](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)

[Syntax](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)

1. **[<form:form](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[action="nextFormPath" modelAttribute=?abc](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[?>](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**

[Spring MVC Form Text Field](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)

[The Spring MVC form text field tag generates an HTML input tag using the bound value. By default, the type of the input tag is text.](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)

[Syntax](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)

1. **[<form:input](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[path="name"](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[/>](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**

[Here,](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[path](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[attribute binds the form field to the bean property.](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)

[The Spring MVC form tag library also provides other input types such as email, date, tel, etc.](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)

[For email:](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)

1. **[<form:input](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[type=?email? path="email"](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[/>](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**

[For date:](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)

1. **[<form:input](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[type=?date? path="date"](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[/>](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**

[Example of Spring MVC Form Text Field](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)

[Let's see an example to create a railway reservation form using form tag library.](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)

[26.8M](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)

[568](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)

[How to find Nth Highest Salary in SQL](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)

[1. Add dependencies to pom.xml file.](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)

1. [<!-- https://mvnrepository.com/artifact/org.springframework/spring-webmvc -->](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)
2. **[<dependency>](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**
3. **[<groupId>](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[org.springframework](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[</groupId>](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**
4. **[<artifactId>](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[spring-webmvc](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[</artifactId>](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**
5. **[<version>](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[5.1.1.RELEASE](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[</version>](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**
6. **[</dependency>](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**
8. [<!-- https://mvnrepository.com/artifact/javax.servlet/javax.servlet-api -->](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)
9. **[<dependency>](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**
10. **[<groupId>](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[javax.servlet](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[</groupId>](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**
11. **[<artifactId>](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[servlet-api](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[</artifactId>](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**
12. **[<version>](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[3.0-alpha-1](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[</version>](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**
13. **[</dependency>](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**
15. [<!-- https://mvnrepository.com/artifact/javax.servlet/jstl -->](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)
16. **[<dependency>](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**
17. **[<groupId>](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[javax.servlet](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[</groupId>](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**
18. **[<artifactId>](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[jstl](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[</artifactId>](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**
19. **[<version>](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[1.2](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[</version>](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**
20. **[</dependency>](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**
21. [<!-- https://mvnrepository.com/artifact/org.apache.tomcat/tomcat-jasper -->](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)
22. **[<dependency>](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**
23. **[<groupId>](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[org.apache.tomcat](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[</groupId>](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**
24. **[<artifactId>](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[tomcat-jasper](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[</artifactId>](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**
25. **[<version>](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[9.0.12](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[</version>](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**
26. **[</dependency>](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**

[2. Create the bean class](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)

[Here, the bean class contains the variables (along setter and getter methods) corresponding to the input field exist in the form.](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)

**[Reservation.java](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**

1. **[package](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[com.javatpoint;](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)
3. **[public](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)****[class](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[Reservation {](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)
5. **[private](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[String firstName;](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)
6. **[private](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[String lastName;](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)
8. **[public](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[Reservation()](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)
9. [{](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)
10. [}](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)
11. **[public](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[String getFirstName() {](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)
12. **[return](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[firstName;](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)
13. [}](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)
14. **[public](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)****[void](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[setFirstName(String firstName) {](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)
15. **[this](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[.firstName = firstName;](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)
16. [}](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)
17. **[public](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[String getLastName() {](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)
18. **[return](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[lastName;](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)
19. [}](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)
20. **[public](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)****[void](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[setLastName(String lastName) {](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)
21. **[this](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[.lastName = lastName;](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)
22. [}](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)
23. [}](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)

[3. Create the controller class](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)

**[ReservationController.java](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**

1. **[package](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[com.javatpoint;](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)
2. **[import](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[org.springframework.stereotype.Controller;](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)
3. **[import](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[org.springframework.ui.Model;](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)
4. **[import](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[org.springframework.web.bind.annotation.ModelAttribute;](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)
5. **[import](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[org.springframework.web.bind.annotation.RequestMapping;](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)
7. [@RequestMapping("/reservation")](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)
8. [@Controller](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)
9. **[public](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)****[class](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[ReservationController {](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)
10. [@RequestMapping("/bookingForm")](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)
11. **[public](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[String bookingForm(Model model)](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)
12. [{](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)
13. [//create a reservation object](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)
14. [Reservation res=](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[new](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[Reservation();](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)
15. [//provide reservation object to the model](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)
16. [model.addAttribute("reservation", res);](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)
17. **[return](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**["reservation-page";](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)
18. [}](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)
19. [@RequestMapping("/submitForm")](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)
20. [// @ModelAttribute binds form data to the object](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)
21. **[public](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[String submitForm(@ModelAttribute("reservation") Reservation res)](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)
22. [{](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)
23. **[return](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**["confirmation-form";](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)
24. [}](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)
25. [}](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)

[4. Provide the entry of controller in the web.xml file](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)

**[web.xml](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**

1. **[<?xml](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[version="1.0" encoding="UTF-8"](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[?>](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**
2. **[<web-app](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" xmlns="http://java.sun.com/xml/ns/javaee" xsi:schemaLocation="http://java.sun.com/xml/ns/javaee http://java.sun.com/xml/ns/javaee/web-app\_3\_0.xsd" id="WebApp\_ID" version="3.0"](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[>](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**
3. **[<display-name>](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[SpringMVC](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[</display-name>](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**
4. **[<servlet>](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**
5. **[<servlet-name>](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[spring](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[</servlet-name>](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**
6. **[<servlet-class>](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[org.springframework.web.servlet.DispatcherServlet](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[</servlet-class>](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**
7. **[<load-on-startup>](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[1](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[</load-on-startup>](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**
8. **[</servlet>](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**
9. **[<servlet-mapping>](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**
10. **[<servlet-name>](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[spring](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[</servlet-name>](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**
11. **[<url-pattern>](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[/](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[</url-pattern>](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**
12. **[</servlet-mapping>](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**
13. **[</web-app>](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**

[5. Define the bean in the xml file](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)

**[spring-servlet.xml](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**

1. **[<?xml](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[version="1.0" encoding="UTF-8"](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[?>](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**
2. **[<beans](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[xmlns="http://www.springframework.org/schema/beans"](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)
3. [xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)
4. [xmlns:context="http://www.springframework.org/schema/context"](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)
5. [xmlns:mvc="http://www.springframework.org/schema/mvc"](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)
6. [xsi:schemaLocation="](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)
7. [http://www.springframework.org/schema/beans](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)
8. [http://www.springframework.org/schema/beans/spring-beans.xsd](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)
9. [http://www.springframework.org/schema/context](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)
10. [http://www.springframework.org/schema/context/spring-context.xsd](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)
11. [http://www.springframework.org/schema/mvc](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)
12. [http://www.springframework.org/schema/mvc/spring-mvc.xsd"](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[>](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**
13. [<!-- Provide support for component scanning -->](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)
14. **[<context:component-scan](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[base-package="com.javatpoint"](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[/>](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**
15. [<!--Provide support for conversion, formatting and validation -->](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)
16. **[<mvc:annotation-driven/>](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**
17. [<!-- Define Spring MVC view resolver -->](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)
18. **[<bean](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[id="viewResolver" class="org.springframework.web.servlet.view.InternalResourceViewResolver"](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[>](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**
19. **[<property](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[name="prefix" value="/WEB-INF/jsp/"](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[></property>](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**
20. **[<property](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[name="suffix" value=".jsp"](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[></property>](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**
21. **[</bean>](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**
22. **[</beans>](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**

[6. Create the requested page](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)

**[index.jsp](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**

1. [<!DOCTYPE html](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[>](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**
2. **[<html>](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**
3. **[<head>](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**
4. **[<title>](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[Railway Registration Form](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[</title>](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**
5. **[</head>](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**
6. **[<body>](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**
7. **[<a](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[href="reservation/bookingForm"](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[>](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[Click here for reservation.](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[</a>](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**
8. **[</body>](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**
9. **[</html>](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**

[7. Create other view components](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)

**[reservation-page.jsp](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**

1. **[<](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[%@ taglib prefix="form" uri="http://www.springframework.org/tags/form" %](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[>](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**
2. [<!DOCTYPE html](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[>](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**
3. **[<html>](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**
4. **[<head>](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**
5. **[<title>](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[Reservation Form](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[</title>](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**
6. **[</head>](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**
7. **[<h3>](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[Railway Reservation Form](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[</h3>](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**
8. **[<body>](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**
9. **[<form:form](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[action="submitForm" modelAttribute="reservation"](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[>](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**
10. [First name:](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[<form:input](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[path="firstName"](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[/>](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**
11. **[<br><br>](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**
12. [Last name:](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[<form:input](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[path="lastName"](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[/>](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**
13. **[<br><br>](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**
14. **[<input](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[type="submit" value="Submit"](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[/>](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**
15. **[</form:form>](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**
16. **[</body>](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**
17. **[</html>](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**

**[Note - The value passed with the @ModelAttribute annotation should be the same to the modelAttribute value present in the view page.](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**

**[confirmation-page.jsp](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**

1. [<!DOCTYPE html](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[>](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**
2. **[<html>](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**
3. **[<body>](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**
4. **[<p>](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[Your reservation is confirmed successfully. Please, re-check the details.](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[</p>](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**
5. [First Name : ${reservation.firstName}](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[<br>](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**
6. [Last Name : ${reservation.lastName}](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)
7. **[</body>](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**
8. **[</html>](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**

**[Output:](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**

# [Spring MVC Form Text Field Spring MVC Form Text Field Spring MVC Form Text Field Spring MVC Form Text FieldSpring MVC Form Radio Button](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)

[The Spring MVC form radio button allows us to choose only one option at a time. This tag renders an HTML input tag of type radio.](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)

[Syntax](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)

1. **[<form:radiobutton](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[path="abc" value="xyz"](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[/>](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**

[Apart from radio button tag, Spring MVC form tag library also contains](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[radiobuttons](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[tag. This tag renders multiple HTML input tags with type radio.](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)

1. **[<form:radiobuttons](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[path="abc" items="${xyz}"](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[/>](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**

[Example of Spring MVC Form Radio Button](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)

[1. Add dependencies to pom.xml](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)

1. [<!-- https://mvnrepository.com/artifact/org.springframework/spring-webmvc -->](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)
2. **[<dependency>](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**
3. **[<groupId>](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[org.springframework](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[</groupId>](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**
4. **[<artifactId>](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[spring-webmvc](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[</artifactId>](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**
5. **[<version>](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[5.1.1.RELEASE](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[</version>](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**
6. **[</dependency>](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**
8. [<!-- https://mvnrepository.com/artifact/javax.servlet/javax.servlet-api -->](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)
9. **[<dependency>](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**
10. **[<groupId>](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[javax.servlet](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[</groupId>](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**
11. **[<artifactId>](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[servlet-api](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[</artifactId>](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**
12. **[<version>](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[3.0-alpha-1](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[</version>](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**
13. **[</dependency>](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**
15. [<!-- https://mvnrepository.com/artifact/javax.servlet/jstl -->](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)
16. **[<dependency>](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**
17. **[<groupId>](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[javax.servlet](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[</groupId>](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**
18. **[<artifactId>](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[jstl](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[</artifactId>](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**
19. **[<version>](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[1.2](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[</version>](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**
20. **[</dependency>](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**
21. [<!-- https://mvnrepository.com/artifact/org.apache.tomcat/tomcat-jasper -->](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)
22. **[<dependency>](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**
23. **[<groupId>](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[org.apache.tomcat](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[</groupId>](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**
24. **[<artifactId>](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[tomcat-jasper](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[</artifactId>](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**
25. **[<version>](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[9.0.12](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[</version>](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**
26. **[</dependency>](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**

[2. Create the bean class](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)

**[Reservation.java](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**

1. [package com.javatpoint;](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)
3. [public class Reservation {](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)
5. [private String firstName;](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)
6. [private String lastName;](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)
7. [private String Gender;](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)
9. [public Reservation()](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)
10. [{](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)
11. [}](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)
12. [public String getFirstName() {](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)
13. [return firstName;](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)
14. [}](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)
15. [public void setFirstName(String firstName) {](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)
16. [this.firstName = firstName;](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)
17. [}](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)
18. [public String getLastName() {](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)
19. [return lastName;](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)
20. [}](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)
21. [public void setLastName(String lastName) {](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)
22. [this.lastName = lastName;](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)
23. [}](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)
24. [public String getGender() {](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)
25. [return Gender;](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)
26. [}](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)
27. [public void setGender(String gender) {](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)
28. [Gender = gender;](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)
29. [}](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)
30. [}](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)

[3. Create the controller class](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)

**[ReservationController.java](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**

[Exception Handling in Java - Javatpoint](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)

1. **[package](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[com.javatpoint;](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)
2. **[import](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[org.springframework.stereotype.Controller;](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)
3. **[import](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[org.springframework.ui.Model;](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)
4. **[import](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[org.springframework.web.bind.annotation.ModelAttribute;](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)
5. **[import](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[org.springframework.web.bind.annotation.RequestMapping;](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)
7. [@RequestMapping("/reservation")](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)
8. [@Controller](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)
9. **[public](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)****[class](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[ReservationController {](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)
10. [@RequestMapping("/bookingForm")](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)
11. **[public](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[String bookingForm(Model model)](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)
12. [{](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)
13. [//create a reservation object](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)
14. [Reservation res=](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[new](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[Reservation();](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)
15. [//provide reservation object to the model](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)
16. [model.addAttribute("reservation", res);](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)
17. **[return](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**["reservation-page";](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)
18. [}](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)
19. [@RequestMapping("/submitForm")](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)
20. **[public](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[String submitForm(@ModelAttribute("reservation") Reservation res)](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)
21. [{](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)
22. **[return](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**["confirmation-form";](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)
23. [}](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)
24. [}](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)

[4. Provide the entry of controller in the web.xml file](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)

**[web.xml](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**

1. **[<?xml](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[version="1.0" encoding="UTF-8"](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[?>](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**
2. **[<web-app](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" xmlns="http://java.sun.com/xml/ns/javaee" xsi:schemaLocation="http://java.sun.com/xml/ns/javaee http://java.sun.com/xml/ns/javaee/web-app\_3\_0.xsd" id="WebApp\_ID" version="3.0"](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[>](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**
3. **[<display-name>](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[SpringMVC](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[</display-name>](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**
4. **[<servlet>](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**
5. **[<servlet-name>](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[spring](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[</servlet-name>](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**
6. **[<servlet-class>](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[org.springframework.web.servlet.DispatcherServlet](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[</servlet-class>](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**
7. **[<load-on-startup>](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[1](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[</load-on-startup>](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**
8. **[</servlet>](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**
9. **[<servlet-mapping>](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**
10. **[<servlet-name>](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[spring](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[</servlet-name>](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**
11. **[<url-pattern>](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[/](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[</url-pattern>](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**
12. **[</servlet-mapping>](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**
13. **[</web-app>](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**

[5. Define the bean in the xml file](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)

**[spring-servlet.xml](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**

1. **[<?xml](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[version="1.0" encoding="UTF-8"](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[?>](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**
2. **[<beans](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[xmlns="http://www.springframework.org/schema/beans"](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)
3. [xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)
4. [xmlns:context="http://www.springframework.org/schema/context"](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)
5. [xmlns:mvc="http://www.springframework.org/schema/mvc"](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)
6. [xsi:schemaLocation="](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)
7. [http://www.springframework.org/schema/beans](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)
8. [http://www.springframework.org/schema/beans/spring-beans.xsd](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)
9. [http://www.springframework.org/schema/context](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)
10. [http://www.springframework.org/schema/context/spring-context.xsd](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)
11. [http://www.springframework.org/schema/mvc](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)
12. [http://www.springframework.org/schema/mvc/spring-mvc.xsd"](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[>](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**
13. [<!-- Provide support for component scanning -->](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)
14. **[<context:component-scan](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[base-package="com.javatpoint"](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[/>](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**
15. [<!--Provide support for conversion, formatting and validation -->](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)
16. **[<mvc:annotation-driven/>](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**
17. [<!-- Define Spring MVC view resolver -->](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)
18. **[<bean](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[id="viewResolver" class="org.springframework.web.servlet.view.InternalResourceViewResolver"](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[>](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**
19. **[<property](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[name="prefix" value="/WEB-INF/jsp/"](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[></property>](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**
20. **[<property](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[name="suffix" value=".jsp"](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[></property>](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**
21. **[</bean>](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**
22. **[</beans>](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**

[6. Create the requested page](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)

**[index.jsp](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**

1. [<!DOCTYPE html](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[>](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**
2. **[<html>](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**
3. **[<head>](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**
4. **[<title>](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[Railway Registration Form](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[</title>](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**
5. **[</head>](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**
6. **[<body>](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**
7. **[<a](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[href="reservation/bookingForm"](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[>](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[Click here for reservation.](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[</a>](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**
8. **[</body>](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**
9. **[</html>](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**

[7. Create the view components](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)

**[reservation-page.jsp](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**

1. **[<](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[%@ taglib prefix="form" uri="http://www.springframework.org/tags/form" %](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[>](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**
2. [<!DOCTYPE html](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[>](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**
3. **[<html>](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**
4. **[<head>](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**
5. **[<title>](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[Reservation Form](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[</title>](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**
6. **[</head>](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**
7. **[<h3>](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[Railway Reservation Form](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[</h3>](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**
8. **[<body>](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**
9. **[<form:form](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[action="submitForm" modelAttribute="reservation"](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[>](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**
10. [First name:](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[<form:input](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[path="firstName"](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[/>](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**
11. **[<br><br>](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**
12. [Last name:](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[<form:input](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[path="lastName"](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[/>](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**
13. **[<br><br>](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**
14. [Gender:](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)
15. [Male](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[<form:radiobutton](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[path="Gender" value="Male"](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[/>](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**
16. [Female](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[<form:radiobutton](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[path="Gender" value="Female"](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[/>](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**
17. **[<br><br>](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**
18. **[<input](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[type="submit" value="Submit"](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[/>](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**
19. **[</form:form>](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**
20. **[</body>](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**
21. **[</html>](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**

**[confirmation-page.jsp](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**

1. [<!DOCTYPE html](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[>](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**
2. **[<html>](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**
3. **[<body>](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**
4. **[<p>](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[Your reservation is confirmed successfully. Please, re-check the details.](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[</p>](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**
5. [First Name : ${reservation.firstName}](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[<br>](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**
6. [Last Name : ${reservation.lastName}](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[<br>](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**
7. [Gender: ${reservation.gender}](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)
8. **[</body>](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**
9. **[</html>](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**

**[Output:](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**

# [Spring MVC Form Radio Button Spring MVC Form Radio Button Spring MVC Form Radio Button Spring MVC Form Radio ButtonSpring MVC Form Checkbox](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)

[The Spring MVC form checkbox facilitates to choose multiple options at the same time. This tag renders an HTML input tag of type checkbox.](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)

[Syntax](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)

1. **[<form:checkbox](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[path="abc" value="element"](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[/>](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**

[Apart from checkbox tag, Spring MVC form tag library also contains](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[checkboxes](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[tag. This tag renders multiple HTML input tags with type checkbox. This tag is used only if you don't want to list all the elements in the view page. In such a case, you can provide the elements at runtime and pass it to the tag. As the user can select multiple options, you need to pass the elements of Array, a List or a Map type.](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)

[Syntax](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)

1. **[<form:checkboxes](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[path="abc" items="${object.elementList}"](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[/>](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**

[Example of Spring MVC Form Check Box](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)

[1. Add dependencies to pom.xml file.](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)

1. [<!-- https://mvnrepository.com/artifact/org.springframework/spring-webmvc -->](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)
2. **[<dependency>](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**
3. **[<groupId>](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[org.springframework](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[</groupId>](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**
4. **[<artifactId>](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[spring-webmvc](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[</artifactId>](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**
5. **[<version>](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[5.1.1.RELEASE](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[</version>](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**
6. **[</dependency>](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**
8. [<!-- https://mvnrepository.com/artifact/javax.servlet/javax.servlet-api -->](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)
9. **[<dependency>](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**
10. **[<groupId>](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[javax.servlet](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[</groupId>](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**
11. **[<artifactId>](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[servlet-api](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[</artifactId>](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**
12. **[<version>](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[3.0-alpha-1](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[</version>](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**
13. **[</dependency>](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**
15. [<!-- https://mvnrepository.com/artifact/javax.servlet/jstl -->](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)
16. **[<dependency>](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**
17. **[<groupId>](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[javax.servlet](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[</groupId>](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**
18. **[<artifactId>](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[jstl](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[</artifactId>](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**
19. **[<version>](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[1.2](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[</version>](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**
20. **[</dependency>](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**
21. [<!-- https://mvnrepository.com/artifact/org.apache.tomcat/tomcat-jasper -->](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)
22. **[<dependency>](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**
23. **[<groupId>](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[org.apache.tomcat](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[</groupId>](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**
24. **[<artifactId>](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[tomcat-jasper](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[</artifactId>](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**
25. **[<version>](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[9.0.12](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[</version>](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**
26. **[</dependency>](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**

[2. Create the bean class](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)

**[Reservation.java](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**

1. **[package](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[com.javatpoint;](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)
3. **[public](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)****[class](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[Reservation {](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)
5. **[private](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[String firstName;](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)
6. **[private](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[String lastName;](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)
7. **[private](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[String Gender;](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)
8. **[private](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[String[] Food;](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)
9. **[public](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[Reservation()](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)
10. [{](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)
11. [}](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)
12. **[public](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[String getFirstName() {](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)
13. **[return](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[firstName;](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)
14. [}](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)
15. **[public](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)****[void](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[setFirstName(String firstName) {](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)
16. **[this](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[.firstName = firstName;](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)
17. [}](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)
18. **[public](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[String getLastName() {](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)
19. **[return](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[lastName;](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)
20. [}](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)
21. **[public](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)****[void](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[setLastName(String lastName) {](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)
22. **[this](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[.lastName = lastName;](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)
23. [}](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)
24. **[public](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[String getGender() {](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)
25. **[return](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[Gender;](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)
26. [}](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)
27. **[public](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)****[void](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[setGender(String gender) {](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)
28. [Gender = gender;](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)
29. [}](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)
30. **[public](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[String[] getFood() {](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)
31. **[return](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[Food;](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)
32. [}](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)
33. **[public](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)****[void](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[setFood(String[] food) {](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)
34. [Food = food;](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)
35. [}](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)
36. [}](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)

[3. Create the controller class](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)

**[ReservationController.java](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**

[Features of Java - Javatpoint](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)

1. **[package](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[com.javatpoint;](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)
2. **[import](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[org.springframework.stereotype.Controller;](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)
3. **[import](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[org.springframework.ui.Model;](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)
4. **[import](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[org.springframework.web.bind.annotation.ModelAttribute;](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)
5. **[import](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[org.springframework.web.bind.annotation.RequestMapping;](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)
7. [@RequestMapping("/reservation")](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)
8. [@Controller](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)
9. **[public](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)****[class](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[ReservationController {](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)
10. [@RequestMapping("/bookingForm")](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)
11. **[public](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[String bookingForm(Model model)](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)
12. [{](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)
13. [//create a reservation object](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)
14. [Reservation res=](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[new](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[Reservation();](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)
15. [//provide reservation object to the model](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)
16. [model.addAttribute("reservation", res);](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)
17. **[return](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**["reservation-page";](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)
18. [}](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)
19. [@RequestMapping("/submitForm")](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)
20. **[public](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[String submitForm(@ModelAttribute("reservation") Reservation res)](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)
21. [{](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)
22. **[return](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**["confirmation-form";](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)
23. [}](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)
24. [}](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)

[4. Provide the entry of controller in the web.xml file](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)

**[web.xml](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**

1. **[<?xml](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[version="1.0" encoding="UTF-8"](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[?>](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**
2. **[<web-app](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" xmlns="http://java.sun.com/xml/ns/javaee" xsi:schemaLocation="http://java.sun.com/xml/ns/javaee http://java.sun.com/xml/ns/javaee/web-app\_3\_0.xsd" id="WebApp\_ID" version="3.0"](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[>](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**
3. **[<display-name>](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[SpringMVC](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[</display-name>](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**
4. **[<servlet>](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**
5. **[<servlet-name>](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[spring](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[</servlet-name>](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**
6. **[<servlet-class>](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[org.springframework.web.servlet.DispatcherServlet](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[</servlet-class>](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**
7. **[<load-on-startup>](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[1](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[</load-on-startup>](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**
8. **[</servlet>](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**
9. **[<servlet-mapping>](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**
10. **[<servlet-name>](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[spring](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[</servlet-name>](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**
11. **[<url-pattern>](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[/](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[</url-pattern>](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**
12. **[</servlet-mapping>](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**
13. **[</web-app>](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**

[5. Define the bean in the xml file](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)

**[spring-servlet.xml](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**

1. **[<?xml](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[version="1.0" encoding="UTF-8"](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[?>](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**
2. **[<beans](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[xmlns="http://www.springframework.org/schema/beans"](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)
3. [xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)
4. [xmlns:context="http://www.springframework.org/schema/context"](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)
5. [xmlns:mvc="http://www.springframework.org/schema/mvc"](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)
6. [xsi:schemaLocation="](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)
7. [http://www.springframework.org/schema/beans](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)
8. [http://www.springframework.org/schema/beans/spring-beans.xsd](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)
9. [http://www.springframework.org/schema/context](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)
10. [http://www.springframework.org/schema/context/spring-context.xsd](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)
11. [http://www.springframework.org/schema/mvc](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)
12. [http://www.springframework.org/schema/mvc/spring-mvc.xsd"](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[>](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**
13. [<!-- Provide support for component scanning -->](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)
14. **[<context:component-scan](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[base-package="com.javatpoint"](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[/>](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**
15. [<!--Provide support for conversion, formatting and validation -->](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)
16. **[<mvc:annotation-driven/>](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**
17. [<!-- Define Spring MVC view resolver -->](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)
18. **[<bean](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[id="viewResolver" class="org.springframework.web.servlet.view.InternalResourceViewResolver"](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[>](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**
19. **[<property](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[name="prefix" value="/WEB-INF/jsp/"](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[></property>](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**
20. **[<property](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[name="suffix" value=".jsp"](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[></property>](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**
21. **[</bean>](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**
22. **[</beans>](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**

[6. Create the requested page](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)

**[index.jsp](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**

1. [<!DOCTYPE html](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[>](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**
2. **[<html>](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**
3. **[<head>](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**
4. **[<title>](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[Railway Registration Form](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[</title>](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**
5. **[</head>](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**
6. **[<body>](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**
7. **[<a](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[href="reservation/bookingForm"](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[>](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[Click here for reservation.](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[</a>](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**
8. **[</body>](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**
9. **[</html>](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**

[7. Create the view components](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)

**[reservation-page.jsp](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**

1. **[<](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[%@ taglib prefix="form" uri="http://www.springframework.org/tags/form" %](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[>](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**
2. [<!DOCTYPE html](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[>](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**
3. **[<html>](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**
4. **[<head>](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**
5. **[<title>](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[Reservation Form](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[</title>](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**
6. **[</head>](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**
7. **[<h3>](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[Railway Reservation Form](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[</h3>](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**
8. **[<body>](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**
9. **[<form:form](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[action="submitForm" modelAttribute="reservation"](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[>](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**
10. [First name:](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[<form:input](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[path="firstName"](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[/>](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**
11. **[<br><br>](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**
12. [Last name:](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[<form:input](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[path="lastName"](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[/>](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**
13. **[<br><br>](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**
14. [Gender:](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)
15. [Male](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[<form:radiobutton](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[path="Gender" value="Male"](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[/>](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**
16. [Female](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[<form:radiobutton](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[path="Gender" value="Female"](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[/>](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**
17. **[<br><br>](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**
18. [Meals:](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)
19. [BreakFast](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[<form:checkbox](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[path="Food" value="BreakFast"](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[/>](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**
20. [Lunch](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[<form:checkbox](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[path="Food" value="Lunch"](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[/>](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**
21. [Dinner](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[<form:checkbox](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[path="Food" value="Dinner"](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[/>](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**
22. **[<br><br>](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**
23. **[<input](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[type="submit" value="Submit"](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[/>](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**
24. **[</form:form>](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**
25. **[</body>](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**
26. **[</html>](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**

**[confirmation-page.jsp](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**

1. **[<](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[%@taglib prefix="c" uri="http://java.sun.com/jsp/jstl/core"%](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[>](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**
2. [<!DOCTYPE html](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[>](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**
3. **[<html>](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**
4. **[<body>](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**
5. **[<p>](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[Your reservation is confirmed successfully. Please, re-check the details.](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[</p>](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**
6. [First Name : ${reservation.firstName}](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[<br>](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**
7. [Last Name : ${reservation.lastName}](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[<br>](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**
8. [Gender: ${reservation.gender}](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[<br>](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**
9. [Meals:](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)
10. **[<ul>](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**
11. **[<c:forEach](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[var="meal" items="${reservation.food}"](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[>](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**
12. **[<li>](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[${meal}](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[</li>](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**
13. **[</c:forEach>](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**
14. **[</ul>](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**
15. **[</body>](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**
16. **[</html>](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**

**[Output:](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**

# [Spring MVC Form Checkbox Spring MVC Form Checkbox Spring MVC Form Checkbox Spring MVC Form CheckboxSpring MVC Form Drop-Down List](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)

[The Spring MVC form drop-down list contains the list of elements. This tag generates an HTML select element. It allows data binding to the selected element.](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)

[Syntax](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)

1. **[<form:select](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[path="name"](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[>](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**

[Here are some other tags used to select the options.](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)

[option tag](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)

[This tag generates an HTML option tag. Each tag contains a value that can be selected by the user.](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)

[Syntax](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)

1. **[<form:option](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[value="abc" label="xyz"](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[/>](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**

[options tag](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)

[This tag generates a list of HTML option tags. Each tag contains a list of elements selected by the user.](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)

[14.6M](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)

[233](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)

[Triggers in SQL (Hindi)](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)

[Syntax](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)

1. **[<form:options](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[items="${elementList}" itemValue="abc" itemLabel="xyz"](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[/>](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**

[Example of Spring MVC Form Drop-Down List](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)

[1. Add dependencies to pom.xml file.](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)

1. [<!-- https://mvnrepository.com/artifact/org.springframework/spring-webmvc -->](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)
2. **[<dependency>](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**
3. **[<groupId>](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[org.springframework](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[</groupId>](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**
4. **[<artifactId>](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[spring-webmvc](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[</artifactId>](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**
5. **[<version>](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[5.1.1.RELEASE](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[</version>](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**
6. **[</dependency>](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**
8. [<!-- https://mvnrepository.com/artifact/javax.servlet/javax.servlet-api -->](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)
9. **[<dependency>](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**
10. **[<groupId>](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[javax.servlet](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[</groupId>](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**
11. **[<artifactId>](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[servlet-api](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[</artifactId>](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**
12. **[<version>](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[3.0-alpha-1](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[</version>](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**
13. **[</dependency>](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**
15. [<!-- https://mvnrepository.com/artifact/javax.servlet/jstl -->](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)
16. **[<dependency>](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**
17. **[<groupId>](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[javax.servlet](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[</groupId>](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**
18. **[<artifactId>](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[jstl](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[</artifactId>](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**
19. **[<version>](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[1.2](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[</version>](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**
20. **[</dependency>](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**
21. [<!-- https://mvnrepository.com/artifact/org.apache.tomcat/tomcat-jasper -->](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)
22. **[<dependency>](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**
23. **[<groupId>](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[org.apache.tomcat](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[</groupId>](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**
24. **[<artifactId>](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[tomcat-jasper](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[</artifactId>](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**
25. **[<version>](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[9.0.12](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[</version>](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**
26. **[</dependency>](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**

[2. Create the bean class](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)

**[Reservation.java](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**

1. **[package](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[com.javatpoint;](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)
3. **[public](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)****[class](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[Reservation {](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)
5. **[private](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[String firstName;](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)
6. **[private](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[String lastName;](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)
7. **[private](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[String Gender;](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)
8. **[private](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[String[] Food;](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)
9. **[private](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[String cityFrom;](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)
10. **[private](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[String cityTo;](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)
11. **[public](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[Reservation()](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)
12. [{](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)
13. [}](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)
14. **[public](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[String getFirstName() {](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)
15. **[return](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[firstName;](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)
16. [}](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)
17. **[public](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)****[void](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[setFirstName(String firstName) {](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)
18. **[this](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[.firstName = firstName;](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)
19. [}](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)
20. **[public](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[String getLastName() {](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)
21. **[return](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[lastName;](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)
22. [}](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)
23. **[public](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)****[void](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[setLastName(String lastName) {](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)
24. **[this](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[.lastName = lastName;](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)
25. [}](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)
26. **[public](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[String getGender() {](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)
27. **[return](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[Gender;](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)
28. [}](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)
29. **[public](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)****[void](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[setGender(String gender) {](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)
30. [Gender = gender;](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)
31. [}](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)
32. **[public](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[String[] getFood() {](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)
33. **[return](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[Food;](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)
34. [}](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)
35. **[public](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)****[void](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[setFood(String[] food) {](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)
36. [Food = food;](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)
37. [}](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)
38. **[public](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[String getCityFrom() {](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)
39. **[return](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[cityFrom;](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)
40. [}](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)
41. **[public](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)****[void](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[setCityFrom(String cityFrom) {](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)
42. **[this](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[.cityFrom = cityFrom;](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)
43. [}](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)
44. **[public](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[String getCityTo() {](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)
45. **[return](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[cityTo;](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)
46. [}](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)
47. **[public](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)****[void](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[setCityTo(String cityTo) {](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)
48. **[this](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[.cityTo = cityTo;](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)
49. [}](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)
50. [}](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)

[3. Create the controller class](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)

**[ReservationController.java](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**

1. **[package](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[com.javatpoint;](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)
2. **[import](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[org.springframework.stereotype.Controller;](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)
3. **[import](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[org.springframework.ui.Model;](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)
4. **[import](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[org.springframework.web.bind.annotation.ModelAttribute;](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)
5. **[import](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[org.springframework.web.bind.annotation.RequestMapping;](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)
7. [@RequestMapping("/reservation")](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)
8. [@Controller](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)
9. **[public](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)****[class](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[ReservationController {](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)
10. [@RequestMapping("/bookingForm")](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)
11. **[public](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[String bookingForm(Model model)](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)
12. [{](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)
13. [//create a reservation object](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)
14. [Reservation res=](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[new](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[Reservation();](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)
15. [//provide reservation object to the model](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)
16. [model.addAttribute("reservation", res);](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)
17. **[return](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**["reservation-page";](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)
18. [}](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)
19. [@RequestMapping("/submitForm")](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)
20. **[public](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[String submitForm(@ModelAttribute("reservation") Reservation res)](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)
21. [{](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)
22. **[return](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**["confirmation-form";](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)
23. [}](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)
24. [}](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)

[4. Provide the entry of controller in the web.xml file](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)

**[web.xml](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**

1. **[<?xml](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[version="1.0" encoding="UTF-8"](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[?>](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**
2. **[<web-app](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" xmlns="http://java.sun.com/xml/ns/javaee" xsi:schemaLocation="http://java.sun.com/xml/ns/javaee http://java.sun.com/xml/ns/javaee/web-app\_3\_0.xsd" id="WebApp\_ID" version="3.0"](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[>](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**
3. **[<display-name>](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[SpringMVC](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[</display-name>](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**
4. **[<servlet>](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**
5. **[<servlet-name>](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[spring](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[</servlet-name>](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**
6. **[<servlet-class>](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[org.springframework.web.servlet.DispatcherServlet](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[</servlet-class>](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**
7. **[<load-on-startup>](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[1](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[</load-on-startup>](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**
8. **[</servlet>](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**
9. **[<servlet-mapping>](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**
10. **[<servlet-name>](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[spring](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[</servlet-name>](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**
11. **[<url-pattern>](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[/](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[</url-pattern>](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**
12. **[</servlet-mapping>](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**
13. **[</web-app>](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**

[5. Define the bean in the xml file](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)

**[spring-servlet.xml](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**

1. **[<?xml](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[version="1.0" encoding="UTF-8"](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[?>](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**
2. **[<beans](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[xmlns="http://www.springframework.org/schema/beans"](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)
3. [xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)
4. [xmlns:context="http://www.springframework.org/schema/context"](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)
5. [xmlns:mvc="http://www.springframework.org/schema/mvc"](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)
6. [xsi:schemaLocation="](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)
7. [http://www.springframework.org/schema/beans](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)
8. [http://www.springframework.org/schema/beans/spring-beans.xsd](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)
9. [http://www.springframework.org/schema/context](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)
10. [http://www.springframework.org/schema/context/spring-context.xsd](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)
11. [http://www.springframework.org/schema/mvc](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)
12. [http://www.springframework.org/schema/mvc/spring-mvc.xsd"](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[>](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**
13. [<!-- Provide support for component scanning -->](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)
14. **[<context:component-scan](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[base-package="com.javatpoint"](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[/>](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**
15. [<!--Provide support for conversion, formatting and validation -->](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)
16. **[<mvc:annotation-driven/>](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**
17. [<!-- Define Spring MVC view resolver -->](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)
18. **[<bean](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[id="viewResolver" class="org.springframework.web.servlet.view.InternalResourceViewResolver"](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[>](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**
19. **[<property](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[name="prefix" value="/WEB-INF/jsp/"](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[></property>](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**
20. **[<property](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[name="suffix" value=".jsp"](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[></property>](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**
21. **[</bean>](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**
22. **[</beans>](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**

[6. Create the requested page](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)

**[index.jsp](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**

1. [<!DOCTYPE html](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[>](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**
2. **[<html>](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**
3. **[<head>](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**
4. **[<title>](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[Railway Registration Form](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[</title>](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**
5. **[</head>](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**
6. **[<body>](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**
7. **[<a](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[href="reservation/bookingForm"](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[>](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[Click here for reservation.](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[</a>](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**
8. **[</body>](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**
9. **[</html>](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**

[7. Create the view components](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)

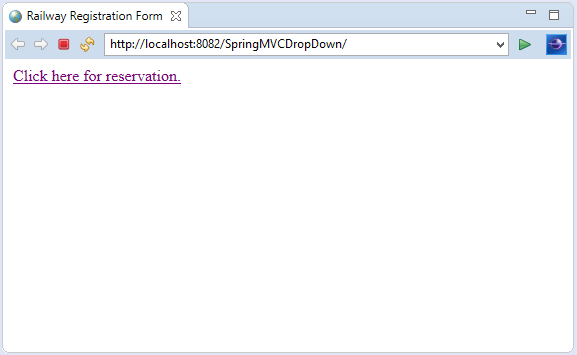
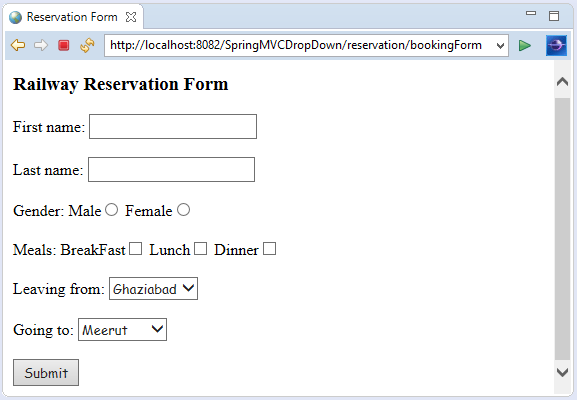
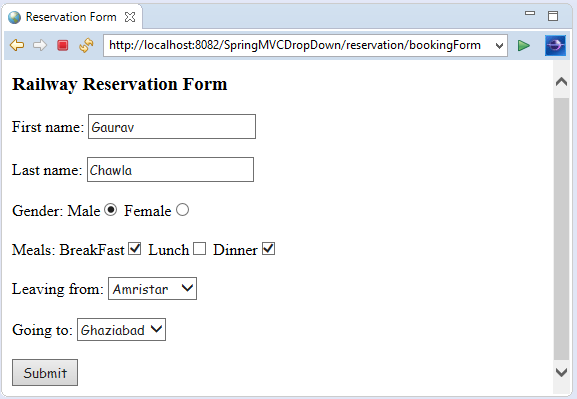
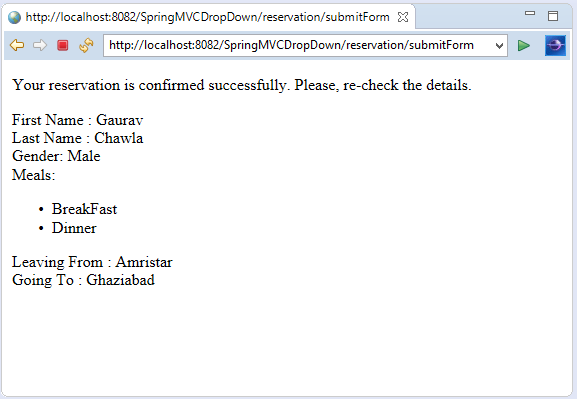
**[reservation-page.jsp](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**

1. **[<](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[%@ taglib prefix="form" uri="http://www.springframework.org/tags/form" %](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[>](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**
2. [<!DOCTYPE html](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[>](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**
3. **[<html>](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**
4. **[<head>](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**
5. **[<title>](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[Reservation Form](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[</title>](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**
6. **[</head>](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**
7. **[<h3>](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[Railway Reservation Form](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[</h3>](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**
8. **[<body>](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**
9. **[<form:form](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[action="submitForm" modelAttribute="reservation"](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[>](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**
10. [First name:](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[<form:input](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[path="firstName"](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[/>](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**
11. **[<br><br>](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**
12. [Last name:](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[<form:input](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[path="lastName"](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[/>](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**
13. **[<br><br>](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**
14. [Gender:](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)
15. [Male](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[<form:radiobutton](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[path="Gender" value="Male"](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[/>](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**
16. [Female](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[<form:radiobutton](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[path="Gender" value="Female"](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[/>](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**
17. **[<br><br>](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**
18. [Meals:](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)
19. [BreakFast](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[<form:checkbox](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[path="Food" value="BreakFast"](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[/>](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**
20. [Lunch](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[<form:checkbox](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[path="Food" value="Lunch"](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[/>](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**
21. [Dinner](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[<form:checkbox](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[path="Food" value="Dinner"](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[/>](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**
22. **[<br><br>](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**
23. [Leaving from:](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[<form:select](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[path="cityFrom"](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[>](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**
24. **[<form:option](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[value="Ghaziabad" label="Ghaziabad"](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[/>](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**
25. **[<form:option](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[value="Modinagar" label="Modinagar"](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[/>](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**
26. **[<form:option](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[value="Meerut" label="Meerut"](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[/>](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**
27. **[<form:option](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[value="Amristar" label="Amristar"](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[/>](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**
28. **[</form:select>](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**
29. **[<br><br>](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**
30. [Going to:](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[<form:select](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[path="cityTo"](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[>](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**
31. **[<form:option](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[value="Ghaziabad" label="Ghaziabad"](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[/>](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**
32. **[<form:option](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[value="Modinagar" label="Modinagar"](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[/>](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**
33. **[<form:option](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[value="Meerut" label="Meerut"](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[/>](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**
34. **[<form:option](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[value="Amristar" label="Amristar"](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[/>](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**
35. **[</form:select>](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**
36. **[<br><br>](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**
37. **[<input](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[type="submit" value="Submit"](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[/>](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**
38. **[</form:form>](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**
39. **[</body>](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**
40. **[</html>](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**

**[confirmation-page.jsp](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**

1. **[<](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[%@taglib prefix="c" uri="http://java.sun.com/jsp/jstl/core"%](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[>](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**
2. [<!DOCTYPE html](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[>](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**
3. **[<html>](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**
4. **[<body>](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**
5. **[<p>](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[Your reservation is confirmed successfully. Please, re-check the details.](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[</p>](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**
6. [First Name : ${reservation.firstName}](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[<br>](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**
7. [Last Name : ${reservation.lastName}](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[<br>](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**
8. [Gender: ${reservation.gender}](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[<br>](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**
9. [Meals:](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)
10. **[<ul>](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**
11. **[<c:forEach](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[var="meal" items="${reservation.food}"](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[>](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**
12. **[<li>](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[${meal}](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[</li>](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**
13. **[</c:forEach>](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**
14. **[</ul>](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**
15. [Leaving From : ${reservation.cityFrom}](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**[<br>](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**
16. [Going To : ${reservation.cityTo}](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)
17. **[</body>](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**
18. **[</html>](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**

**[Output:](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)**

[  
  
  
](https://www.javatpoint.com/spring-aop-aspectj-xml-configuration-example)