

Menu

[Struts 1](#)[Struts 2](#)[Spring](#)[Hibernate](#)[Ant](#)[Log4j](#)[Spring](#) > Setter Injection**Spring Setter Injection**

In this example you will learn how to set a bean property via setter injection. Consider the following User bean class.

```

01. package com.vaannila;
02.
03. public class User {
04.
05.     private String name;
06.     private int age;
07.     private String country;
08.
09.     public String getName() {
10.         return name;
11.     }
12.
13.     public void setName(String name) {
14.         this.name = name;
15.     }
16.
17.     public int getAge() {
18.         return age;
19.     }
20.
21.     public void setAge(int age) {
22.         this.age = age;
23.     }
24.
25.     public String getCountry() {
26.         return country;
27.     }
28.
29.     public void setCountry(String country) {
30.         this.country = country;
31.     }
32.
33.     public String toString() {
34.         return name + " is " + age + " years old, living in " + country;
35.     }
36. }

```

The User bean class has three attributes viz. name, age and country. All the three attributes are set using the setter injection. The toString() method of the User bean class is overridden to display the user object.

Here the beans.xml file is used to do spring bean configuration. The following code shows how to set a property value thru setter injection.

```

01. <?xml version="1.0" encoding="UTF-8"?>
02. <beans xmlns="http://www.springframework.org/schema/beans"
03.     xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
04.     xsi:schemaLocation=" http://www.springframework.org/schema/beans
05.         http://www.springframework.org/schema/beans/spring-beans.xsd">
06.
07.     <bean id="user" class="com.vaannila.User" >
08.         <property name="name" value="Eswar" />
09.         <property name="age" value="24"/>
10.         <property name="country" value="India"/>
11.     </bean>
12. </beans>

```

The id attribute of the bean element is used to specify the bean name and the class attribute is used to specify the fully qualified class name of the bean. The property element within the bean element is used to inject property value via setter injection. The name attribute of the property element represents the bean attribute and the value attribute specifies the corresponding property value.

Here we set "Eswar", "24" and "India" for the User bean properties name, age and country respectively. The following Main class is used to get the User bean from the Spring IoC container and display its value to the user.

```

01. package com.vaannila;
02.
03. import org.springframework.context.ApplicationContext;
04. import org.springframework.context.support.
05.     ClassPathXmlApplicationContext;
06.
07. public class Main {
08.
09.     public static void main(String[] args) {
10.         ApplicationContext context = new
11.             ClassPathXmlApplicationContext("beans.xml");
12.         User user = (User)context.getBean("user");
13.         System.out.println(user);
14.     }
15. }

```

Subscribe

[RSS](#)[Email](#)

On executing the Main class the following message gets displayed on the console.

```
1. | Eswar is 24 years old, living in India
```

You can download and try the Spring setter injection example by clicking the Download link below.

Source : [Download](#)

Source + Lib : [Download](#)

[Contact Us](#) | Copyright © 2009 vaannila.com