

Menu

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

In this example you will learn how to set a bean property via constructor 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.         User(String name, int age, String country)
10.         {
11.             this.name=name;
12.             this.age=age;
13.             this.country=country;
14.         }
15.
16.         public String toString() {
17.             return name + " is " + age + " years old, living in " + country;
18.         }
19.     }

```

The `User` bean class has three attributes viz. `name`, `age` and `country`. All the three attributes are set thru constructor 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 constructor 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.         <constructor-arg value="Eswar" />
09.         <constructor-arg value="24"/>
10.         <constructor-arg value="India"/>
11.     </bean>
12. </beans>

```

The `constructor-arg` element within the `bean` element is used to set the property value thru constructor injection. Since there is only one constructor in the `User` bean class, this code will work fine. When there is more than one constructor with the same number of arguments, then the following ambiguities will occur. Consider the following code.

```

01. User( int age, String country)
02. {
03.     this.age=age;
04.     this.country=country;
05. }
06.
07. User(String name, String country)
08. {
09.     this.name=name;
10.     this.country=country;
11. }

```

The bean configuration file.

```

1. <bean id="user" class="com.vaannila.User" >
2.     <constructor-arg value="24"/>
3.     <constructor-arg value="India"/>
4. </bean>

```

Now which constructor do you think will be invoked? The first one with the `int` and the `String` argument, right? But for your surprise it will call the second constructor with both `String` arguments. Though we know the first argument is of type `int` and the second argument is of type `String`, spring interprets both as `String` arguments. To avoid this confusion you need to specify the `type` attribute of the `constructor-arg` element. Now with the following bean configuration, the first constructor will be invoked.

```

1. <bean id="user" class="com.vaannila.User" >
2.     <constructor-arg type="int" value="24"/>
3.     <constructor-arg type="java.lang.String" value="India"/>
4. </bean>

```

Now consider this case. We have the following constructors in the `User` bean class.

```

01. User(String name, int age)
02. {
03.     this.name=name;
04.     this.age=age;

```

Subscribe

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

```
05. }  
06.  
07. User( int age, String country)  
08. {  
09.     this.age=age;  
10.     this.country=country;  
11. }
```

The bean configuration file.

```
1. <bean id="user" class="com.vaannila.User" >  
2.     <constructor-arg type="int" value="24"/>  
3.     <constructor-arg type="java.lang.String" value="India"/>  
4. </bean>
```

Now which constructor do you think will be called? The second constructor, right? But again for your surprise the first constructor will be called, this is because the order in which the arguments appear in the bean configuration file will not be considered while invoking the constructor. To solve this problem you can use the `index` attribute to specify the constructor argument index.

Here is the bean configuration file after adding the index attribute.

```
1. <bean id="user" class="com.vaannila.User" >  
2.     <constructor-arg index="0" type="int" value="24"/>  
3.     <constructor-arg index="1" type="java.lang.String" value="India"/>  
4. </bean>
```

Now as expected, the second constructor will be invoked.

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

Source : [Download](#)

Source + Lib : [Download](#)