

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.

```
package com.javatpoint;

public class Address {
    private String city;
    private String state;
    private String country;

    public Address(String city, String state, String country) {
        super();
        this.city = city;
        this.state = state;
        this.country = country;
    }

    public String toString(){
        return city+" "+state+" "+country;
    }
}
```

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.

```
package com.javatpoint;

public class Employee {
    private int id;
    private String name;
    private Address address;//Aggregation

    public Employee() {System.out.println("def cons");}

    public Employee(int id, String name, Address address) {
        super();
        this.id = id;
        this.name = name;
        this.address = address;
    }

    void show(){
        System.out.println(id+" "+name);
        System.out.println(address.toString());
    }
}
```

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.

```
<?xml version="1.0" encoding="UTF-8"?>
<beans
  xmlns="http://www.springframework.org/schema/beans"
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xmlns:p="http://www.springframework.org/schema/p"
  xsi:schemaLocation="http://www.springframework.org/schema/beans
    http://www.springframework.org/schema/beans/spring-beans-
3.0.xsd">

  <bean id="a1" class="com.javatpoint.Address">
    <constructor-arg value="ghaziabad"></constructor-arg>
    <constructor-arg value="UP"></constructor-arg>
    <constructor-arg value="India"></constructor-arg>
  </bean>

  <bean id="e" class="com.javatpoint.Employee">
    <constructor-arg value="12" type="int"></constructor-arg>
    <constructor-arg value="Sonoo"></constructor-arg>
    <constructor-arg>
      <ref bean="a1"/>
    </constructor-arg>
  </bean>

</beans>
```

Test.java

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

```
package com.javatpoint;

import org.springframework.beans.factory.BeanFactory;
import org.springframework.beans.factory.xml.XmlBeanFactory;
import org.springframework.core.io.*;
```

```
public class Test {  
    public static void main(String[] args) {  
  
        Resource r=new ClassPathResource("applicationContext.xml");  
        BeanFactory factory=new XmlBeanFactory(r);  
  
        Employee s=(Employee)factory.getBean("e");  
        s.show();  
  
    }  
}
```

[download this example \(developed using MyEclipse IDE\)](#)

[download this example \(developed using Eclipse IDE\)](#)

[← prev](#)

[next →](#)

Please Share



Learn Latest Tutorials

