

# SPRING FRAMEWORK COURSE

## DEPENDENCY INJECTION (PART 2)



By the expert: Eng. Ubaldo Acosta



**SPRING FRAMEWORK COURSE**

[www.globalmentoring.com.mx](http://www.globalmentoring.com.mx)

# WAYS OF DEPENDENCY INJECTION

There are two common ways to perform dependency injection:

- By the constructor of the class
- By means of the property setter method

The configuration of the injection can be done in two ways:

- Explicitly from the xml file (descriptor)
- By annotations, directly in the Java class



# INYECCIÓN DE DEPENDENCIAS POR CONSTRUCTOR

```
applicationContext.xml
1  <?xml version="1.0" encoding="UTF-8"?>
2  <beans xmlns="http://www.springframework.org/schema/beans"
16    <bean id="hibernateDao" class="beans.dao.HibernateDao">
17      <constructor-arg ref="dataSource"/>
18      <constructor-arg value="10"/>
19    </bean>
22  </beans>
```

This DataSource bean must also be defined

```
HibernateDao.java
15  public class HibernateDao {
16
17      private int timeout;
18      private JdbcTemplate jdbcTemplate;
19
20      public HibernateDao(DataSource ds, int timeout) {
21          this.timeout = timeout;
22          this.jdbcTemplate = new JdbcTemplate(ds);
23      }
24
```

# INJECTION OF DEPENDENCIES BY SETTER

```
applicationContext.xml  %  
1  <?xml version="1.0" encoding="UTF-8"?>  
2  <beans xmlns="http://www.springframework.org/schema/beans"  
11  <bean id="hibernateDao" class="beans.dao.HibernateDao">  
12    <property name="dataSource" ref="dataSource" />  
13    <property name="timeout" value="30" />  
14  </bean>  
15  
16  </beans>  
17
```

```
HibernateDao.java  %  
12  public class HibernateDao {  
13  
14    private int timeout;  
15    private JdbcTemplate jdbcTemplate;  
16  
17    public void setTimeout(int timeout) {  
18      this.timeout = timeout;  
19    }  
20  
21    public void setDataSource(DataSource dataSource) {  
22      this.jdbcTemplate = new JdbcTemplate( dataSource );  
23    }  
24
```

# ADVANTAGES OF DEPENDENCY INJECTION

- Reduction of dependencies between objects
- The code is self-documented, since it describes its dependencies
- A dependency configuration can be applied in one place
- Simpler unit tests
- Reduces complexity and directs programming towards the use of interfaces and not towards specific classes



# TYPES OF COLLECTION INJECTION

Collection Element	Use
<list>	Injects a list of values, allows duplicates.
<set>	Injects a list of values, does not allow duplicates.
<map>	Injects a collection of type map (key-value), where the index and the value can be of any type.
<props>	Injects a collection of type properties (name-value), where both values must be a String

# ALCANCE DE BEANS EN SPRING

Alcance	Significado
singleton	Generates only one bean instance defined by Spring container (default)
prototype	It allows instantiating the defined bean any number of times (one per user)
request	The bean has the HTTP request scope. It is only valid in the web context, such as Spring MVC.
session	The bean has the HTTP session scope. It is only valid in the web context, such as Spring MVC.
global-session	It is only valid in the context of portlets. The bean has a global reach HTTP session.

**SPRING FRAMEWORK COURSE**

[www.globalmentoring.com.mx](http://www.globalmentoring.com.mx)

**ONLINE COURSE**

# **SPRING FRAMEWORK**

---

By: Eng. Ubaldo Acosta



**SPRING FRAMEWORK COURSE**

[www.globalmentoring.com.mx](http://www.globalmentoring.com.mx)