

I Bean: come crearli e istanziarli



Cosa imparerai

- Come creare ed istanziare un bean

Istanziare un bean

I metadati di configurazione sono sostanzialmente un elenco di informazioni che servono al container per creare uno o più oggetti.

Se utilizziamo i metadati in formato XML, nell'elemento `<bean />` è necessario inserire l'attributo «**class**».

```
<bean id="ordine" class="it.test.service.OrdineService">  
...  
</bean>
```

Questo attributo è obbligatorio in tutti i casi tranne se utilizziamo una classe Factory che si occupa di creare le istanze di bean.

Istanziare un bean

L'attributo **class** può essere utilizzato in due modi:

- ❑ **per specificare il tipo di bean da istanziare.** Questo è il caso più comune e prevede che il container crei il bean chiamando il costruttore del bean;
- ❑ **per indicare la classe che contiene il metodo factory statico che viene richiamato per creare l'oggetto.** Questo caso è meno comune. Il tipo di oggetto restituito dal metodo factory statico può essere la stessa classe o un'altra classe.

Istanziare un bean

Creare l'istanza di un bean tramite il costruttore

È il caso più comune.

In questo caso tutte le classi sono utilizzabili e compatibili con Spring.

Non abbiamo bisogno di implementare interfacce specifiche o metodi particolari.

A seconda del tipo di container utilizzato per un bean, in alcuni casi può essere necessario definire un costruttore default (il costruttore vuoto, senza parametri in ingresso).

```
<bean id="prodotto" class="it.test.service.ProdottoService">  
...  
</bean>  
  
<bean id="ordine" class="it.test.service.OrdineService">  
...  
</bean>
```

Istanziare un bean

Creare l'istanza di un bean tramite un metodo factory statico

In questo caso, l'attributo **class** viene utilizzato per indicare la classe che contiene il **metodo statico**. Per poter creare istanza con questo approccio, è necessario utilizzare l'attributo **factory-method** per specificare il nome del metodo factory da invocare per la creazione del bean.

NOTA: la classe che contiene il metodo factory può essere la classe stessa del bean o un'altra classe.

```
public class BeanFactory {  
    private static OrdineService ordine = new OrdineService();  
  
    public static OrdineService getOrdineInstance() {  
        return ordine;  
    }  
}
```

```
<beans  
  xmlns="http://www.springframework.org/schema/beans"  
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"  
  xsi:schemaLocation="  
    http://www.springframework.org/schema/beans  
    https://www.springframework.org/schema/beans/spring-beans.xsd">  
  
  <bean id="ordine2" class="it.test.BeanFactory" factory-method="getOrdineInstance">  
    <!-- ... -->  
  </bean>  
</beans>
```

```
public class Main {  
    public static void main(String[] args) {  
        ApplicationContext context = new ClassPathXmlApplicationContext("services-xml.xml");  
  
        OrdineService os = context.getBean("ordine2", OrdineService.class);  
  
        System.out.println(os.sayHello());  
    }  
}
```

Istanziare un bean

Creare l'istanza di un bean tramite un metodo factory di istanza

Questo caso è simile al precedente, solo che il metodo richiamato dal container non è un metodo statico ed è definito in un bean appositamente configurato. **In questo caso non si utilizza l'attributo class.**

Gli attributi da utilizzare sono:

- ❑ **factory-bean**, contiene il nome del bean che contiene il metodo che deve essere invocato per creare l'oggetto
- ❑ **factory-method**, contiene il nome del metodo del bean da invocare.

*Il metodo
non è statico*

```
public class BeanFactory {  
    private OrdineService ordine = new OrdineService();  
  
    public OrdineService getOrdineInstance() {  
        return ordine;  
    }  
}
```



```
<beans  
  xmlns="http://www.springframework.org/schema/beans"  
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"  
  xsi:schemaLocation="  
    http://www.springframework.org/schema/beans  
    https://www.springframework.org/schema/beans/spring-beans.xsd">  
  <bean id="beanFactory" class="it.test.BeanFactory">  
  </bean>  
  
  <bean id="ordine2" factory-bean="beanFactory" factory-method="getOrdineInstance">  
  </bean>  
</beans>
```

*Facciamo riferimento al
metodo di un altro bean*

```
public class Main {  
    public static void main(String[] args) {  
        ApplicationContext context = new ClassPathXmlApplicationContext("services-xml.xml");  
  
        OrdineService os = context.getBean("ordine2", OrdineService.class);  
  
        System.out.println(os.sayHello());  
    }  
}
```

Istanziare un bean

...Creare l'istanza di un bean tramite un metodo factory di istanza

È importante considerare anche che una classe factory può contenere anche più di un metodo factory.

L'importante è specificare nell'attributo *factory-method*, il metodo da far invocare al container per la creazione dell'istanza del bean desiderato.

```
public class BeanFactory {  
    private OrdineService ordine = new OrdineService();  
    private ProdottoService prodotto = new ProdottoService();  
  
    public OrdineService getOrdineInstance() {  
        return ordine;  
    }  
  
    public ProdottoService getProdottoInstance() {  
        return prodotto;  
    }  
}
```

Di cosa abbiamo parlato in questa lezione

- Come creare ed istanziare un bean