

# Metadati XML-based



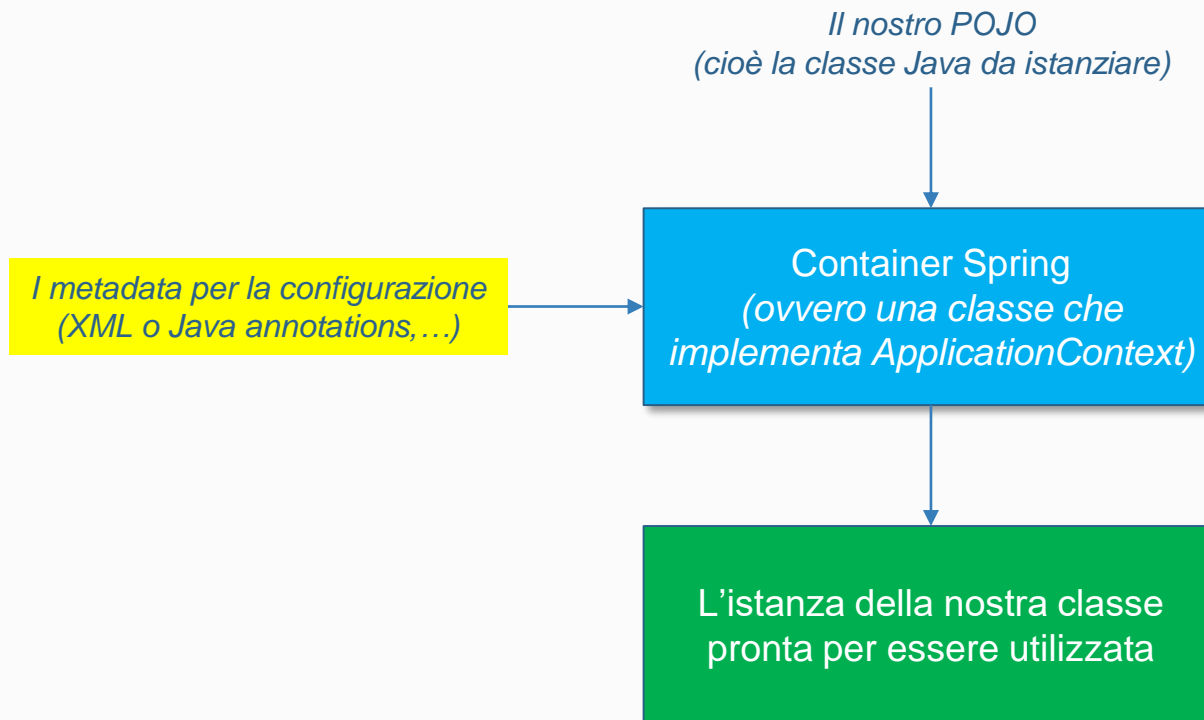
Cosa imparerai

---

- Come configurare i metadati utilizzando XML

# Metadati XML-based

Come spiegato, il container IoC di Spring utilizza i metadati di configurazione per istanziare, configurare ed assemblare gli oggetti.



# Metadati XML-based

I metadati definiscono i bean che compongono l'applicazione e le relazioni tra essi.

## In pratica, cos'è un bean?

Un bean può essere:

- ☐ un oggetto dello strato service (una classe che implementa la business logic)
- ☐ un DAO (Data Access Object)
- ☐ un oggetto dello strato presentation (ad es. una classe per la gestione delle view di una web app)
- ☐ code JMS
- ☐ ecc...

# Metadati XML-based

Per configurare un bean è necessario utilizzare l'elemento **<bean />** inserito all'interno di un elemento padre **<beans />**.

```
<?xml version="1.0" encoding="UTF-8"?>
<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="myBean" class="it.corso.spring.ClienteDAO">
    qui inseriamo le dipendenze da altri bean
    ed eventuali parametri di configurazione
  </bean>

  <bean id="..." class="...">
    ...
  </bean>
</beans>
```

!

L'attributo **id** è una stringa che identifica univocamente il bean all'interno dell'applicazione.

L'attributo **class** definisce il tipo di bean e corrisponde al *fully qualified classname*.

# Metadati XML-based

Spring consente di definire i bean utilizzando più file di configurazione.

Avere più file XML è comodo se vogliamo suddividere i bean per livello logico (ad es. un file per i DAO, uno per il service layer, uno per il presentation layer, ...).

Se utilizziamo più file XML, abbiamo due strade:

1. Dichiarare tutti i file XML nel costruttore del context
2. Raggruppare i file XML in un unico file, tramite l'elemento `<import />` inserito sotto l'elemento `<beans/>` e dichiarare nel costruttore del context solo questo file (questo scenario è interessante perché evita la ricompilazione in caso di aggiunta o aggiornamento dei file XML)

```
<beans>  
  <import resource="myServices.xml"/>  
  <import resource="config/myDao.xml"/>  
  <import resource="/config/myActions.xml"/>
```

qui poi potrai comunque definire altri bean...

```
</beans>
```

# Di cosa abbiamo parlato in questa lezione

- Come configurare i metadati utilizzando XML