

# Spring 回顾笔记 (二)----Spring IOC 容器

作者: [pride](#)

原文链接: <https://ld246.com/article/1543541366059>

来源网站: [链滴](#)

许可协议: [署名-相同方式共享 4.0 国际 \(CC BY-SA 4.0\)](#)

- 在 Spring 的 IOC 容器里配置 Bean

在 xml 文件中通过 bean 节点来配置 bean

```
<!-- 通过全类名的方式配置 bean -->  
<bean id="car" class="com.spring.factorybean.Car" >
```

id: Bean 的名称。

-<font color = #3399FF>在 IOC 容器中必须是唯一的</font>

-若 id 没有指定, Spring 自动将权限定性类名作为 Bean 的名字

-id可以指定多个名字, 名字之间可用逗号、分号、或空格分隔

在 Spring IOC 容器读取 Bean 配置创建 Bean 实例之前, 必须对它进行实例化. 只有在容器实例化后, 可以从 IOC 容器里获取 Bean 实例并使用.

Spring 提供了两种类型的 IOC 容器实现.

BeanFactory: IOC 容器的基本实现.

ApplicationContext: 提供了更多的高级特性. 是 BeanFactory 的子接口.

BeanFactory 是 Spring 框架的基础设施, 面向 Spring 本身; ApplicationContext 面向使用 Spring 框架的开发者, 几乎所有的应用场合都直接使用 ApplicationContext 而非底层的 BeanFactory

无论使用何种方式, 配置文件时相同的.

- 来具体说下ApplicationContext

ApplicationContext 的主要实现类:

ClassPathXmlApplicationContext: 从类路径下加载配置文件

FileSystemXmlApplicationContext: 从文件系统中加载配置文件

<font color = #3399FF>ApplicationContext 在初始化上下文时就实例化所有单例的 Bean。 </font>

那么怎么从 IOC 容器中获取 Bean -->调用 ApplicationContext 的 getBean() 方法;

- 依赖注入的方式

-属性注入

属性注入即通过 <font color = #3399FF>setter</font> 方法注入Bean 的属性值或依赖的对象

属性注入使用 <property> 元素, 使用 name 属性指定 Bean 的属性名称, value 属性或 <value> 节点指定属性值

属性注入是实际应用中常用的注入方式

```
<!-- 通过全类名的方式配置 bean -->  
<bean id="car" class="com.spring.factorybean.Car" >  
<property name="name" value="benchi" ></property>  
<property name="speed" value="10" ></property>  
<property name="price" value="1000000" ></property>  
</bean>
```

-构造器注入

通过构造方法注入Bean 的属性值或依赖的对象, 它保证了 Bean 实例在实例化后就可以使用。

构造器注入在<constructor-arg>元素里声明属性, <constructor-arg> 中没有 name 属性

按索引匹配入参:

```
<bean id="car" class="com.spring.helloworld.Car">  
<constructor-arg value="奥迪" index="0"></constructor-arg>  
<constructor-arg value="长春一汽" index="1"></constructor-arg>  
<constructor-arg value="500000" index="2"></constructor-arg>  
</bean>
```

按类型匹配入参:

```
<bean id="car" class="com.spring.helloworld.Car">  
<constructor-arg value="奥迪" type="java.lang.String"/>  
<constructor-arg value="长春一汽" type="java.lang.String"/>  
<constructor-arg value="500000" type="double"/>  
</bean>
```

-工厂方法注入 (很少使用, 不推荐)