

Spring Beans 自动装配

Beans 自动装配

你已经学会如何使用`<bean>`元素来声明 bean 和通过使用 XML 配置文件中的`<constructor-arg>`和`<property>`元素来注入。

Spring 容器可以在不使用`<constructor-arg>`和`<property>`元素的情况下**自动装配**相互协作的 bean 之间的关系，这有助于减少编写一个大的基于 Spring 的应用程序的 XML 配置的数量。

自动装配模式

下列自动装配模式，它们可用于指示 Spring 容器为来使用自动装配进行依赖注入。你可以使用`<bean>`元素的 **autowire** 属性为一个 bean 定义指定自动装配模式。

模式	描述
no	这是默认的设置，它意味着没有自动装配，你应该使用显式的 bean 定义。在依赖注入章节你已经看到这个了。
<u>byName</u>	由属性名自动装配。Spring 容器看到在 XML 配置文件中 bean 的自尝试匹配，并且将它的属性与在配置文件中被定义为相同名称的 bean 属性。
<u>byType</u>	由属性数据类型自动装配。Spring 容器看到在 XML 配置文件中 bean 的 byType。然后如果它的类型匹配配置文件中的一个确切的 bean 名称。如果存在不止一个这样的 bean，则一个致命的异常将会被抛出。
<u>constructor</u>	类似于 byType，但该类型适用于构造函数参数类型。如果在容器中找不到匹配的 bean，则一个致命错误将会发生。
autodetect	Spring 首先尝试通过 constructor 使用自动装配来连接，如果它不成功，则尝试 byType 自动装配。

可以使用 **byType** 或者 **constructor** 自动装配模式来连接数组和其他类型的集合。

自动装配的局限性

当自动装配始终在同一个项目中使用，它的效果最好。如果通常不使用自动装配，它可能会使开发人员混淆的使用它来连接只有一个或两个 bean 定义。不过，自动装配可以显著减少需要指定的属性或构造器参数，但你应该在使用它们之前考虑到自动装配的局限性和缺点。

限制	描述
重写的可能性	你可以使用总是重写自动装配的 <constructor-arg>和 <property>
原始数据类型	你不能自动装配所谓的简单类型包括基本类型，字符串和类。
混乱的本质	自动装配不如显式装配精确，所以如果可能的话尽可能使用显式装配