spring学习笔记—IOC容器介绍

|  |  |
| --- | --- |
|  | 🢂 内容概览 |
|  | Why：此文档用来做什么？它存在的意义是什么？为解决什么问题？   |  | | --- | |  |   What：当前包含了那些内容？   |  | | --- | |  |   How：此文档应如何参考？   |  | | --- | |  |   Who：此文档适用于那些人员阅读参考？   |  | | --- | |  | |

目录

[1 IOC容器 3](#_Toc478739760)

[1.1 BeanFactory和ApplicationContext接口的区别？ 3](#_Toc478739761)

[1.2 什么是Bean? 3](#_Toc478739762)

[1.3 什么是IOC容器? 3](#_Toc478739763)

[1.4 如何配置元数据？ 4](#_Toc478739764)

# IOC容器

## BeanFactory和ApplicationContext接口的区别？

1. org.springframework.beans包和org.springframework.context包下面包含了IOC容器的基本功能；
2. BeanFactory接口提供了用来管理所有类型对象的配置机制；
3. ApplicationContext接口作为BeanFactory接口的子类，增加了对AOP特性的集成、资源文件的处理、事件的发布以及应用层具体context的实现类，比如：WebApplicationContext；
4. 简而言之，BeanFactory提供了配置框架和基本功能，ApplicationContext添加了更多企业级功能；

## 什么是Bean?

1. bean是指使用IOC容器管理的对象，其初始化、配置、集成都是由IOC容器控制；
2. 一个bean是IOC容器中的一个对象，bean和依赖关系通过配置元数据描述；容器管理bean也依据配置元数据；
3. org.springframework.context.ApplicationContext接口代表了IOC容器，负责bean的初始化、配置、集成；
4. spring提供了xml、注解、代码三种方式来配置元数据；

## 什么是IOC容器?

1. 通常，在单实例应用中，使用ClassPathXmlApplicationContext或者FileSystemXmlApplicationContext来创建容器对象；使用xml配置元数据，同时可以通过少量xml配置指定使用注解或者代码方式配置元数据；
2. 在大多数应用场景中，一般都不会直接在代码中创建IOC容器对象，而是通过配置融合到具体的应用中，比如，在web应用中，通过在web.xml中配置几行代码，就可以配置好容器的创建；

|  |
| --- |
| 容器工作过程 |

## 如何配置元数据？

* 1. Xml方式；
  2. 注解方式；
  3. Java Config方式；（推荐）