# BeanFactory与FactoryBean

## BeanFactory

是工厂类（接口），用于管理bean的工厂，是Spring IOC的核心，用于实例化、定位、配置对象，以及建立对象间的关系。

### BeanFactory实例化Bean的过程：

1. 调用bean的默认构造方法（或指定的构造方法），生成bean1。
2. 将配置文件中配置的属性值注入bean1，生成bean2。
3. 如bean实现了InitalizingBean接口，则执行afterPropertiesSet()方法，生成bean3。
4. 如配置文件配置了init-method属性，则执行指定的方法，生成bean4。

构造函数🡪set🡪InitalizingBean#afterPropertiesSet()🡪init-method

## FactoryBean

通常来讲，根据XML的<bean>节点中配置的class路径，Spring可以通过反射机制来实例化。

如:

<bean id=”beanA” class=”org.leo.BeanA”>

<property name=”userName” value=”leo”/>

</bean>

但实际工作中，实例化一个对象并不如此简单。比如要在实例化的时候做一些复杂的处理。这就用上FactoryBean了。

比如最常用的Quartz框架，我们在配置调度任务的时候，配置的都是FactoryBean。

FactoryBean隐藏了许多实例化的复杂细节，方便上层调用。

如果在<bean>中配置的是FactoryBean，那么getBean()返回的其实是FactoryBean#getObject()返回的对象。

如：

<bean id=”beanA” class=”org.leo.BeanAFactoryBean”/>

getBean(“beanA”)返回的其实是BeanAFactoryBean#getObject()返回的对象，至于是什么，就由BeanAFactoryBean来决定了。

要想返回BeanFactory的实例，应该getBean(“&beanA”)。