

Bean的生命周期

□

1. 创建Bean工厂[继承BeanFactoryAware接口，可以返回创建当前Bean的工厂名]
2. 实例化应用上下文ApplicationContext。应用上下文类似一个监听器，去全局监听所有需要配置的JavaBean， 传输javaBean到Bean工厂。（调用ApplicationContextware可以打印出应用的上下文的名称）
3. 创建Bean[实例化]（调用无参的构造方法）
4. 传递参数
5. 为Bean赋上名字（为id赋值）， [继承BeanNameAware打印名字]Aware关怀
6. 前置处理器BeanPostprocessor[接口]类似过滤器，单独继承接口写一个类[aop]
7. 启动Bean， 配置init-method="方法名"， 重写该方法， 定制自己的方法。或者继承InitializingBean接口， 重写方法。
8. 前置处理器BeanPostprocessor[接口]
9. 使用Bean
10. 关闭Bean并销毁。[调用DisposableBean接口的Destory方法]/或者自定定制销毁方法（配置destroy-method="方法名"）。

完整的Bean的生命周期就是这样，但是使用的时候我们接触的步骤只是： 3->4->5->6->8->9

完整的生命周期还是需要知道的。