### Spring原理

1.Spring特点：

a.轻量级：完整的Spring框架可以在很小的JAR文件中发布，开销小；

b.控制反转：当应用了IOC一个对象依赖的其他对象会通过被动的方式传递，而不是这

个对象自己创建或者查找对象；

c.面向切面：将应用业务逻辑和系统服务分开；

d.框架集合：Spring可以将简单的组件配置、组合成复杂的应用，Spring提供了很多基础

功能(事务管理、持久化框架集成等)，将应用逻辑的开发留给开发者。

2.Spring实例化Bean流程：

a.Spring容器读取Bean配置信息；

b.根据Bean注册表实例化Bean；

c.将Bean实例放到Spring容器中；

d.使用Bean。

3.Spring Bean作用域：

Spring3中为Bean定义了5中作用域，分别为singleton(单例)、prototype(原型)、request、

session、global session：

a.Singleton：单例模式，容器中只会存在一个共享的Bean实例，无论有多少个Bean引

用它，始终指向同一对象，该模式在多线程下是不安全的；

b.prototype：原型模式，每次通过Spring容器获取prototype定义的bean时，容器都将

创建一个新的Bean实例，每个Bean实例都有属于自己的属性和状态；

c.request：在一次Http请求中，容器会返回该Bean的同一实例，而对不同的Http请求

会产生新的Bean，Http请求结束时该Bean的实例会被销毁；

d.session：在一次Http Session中，容器会返回该Bean的同一实例，而对不同的session

请求会创建新的实例，该bean只在当前session有效；

e.global Session：在一个全局的 Http Session 中，容器会返回该 Bean 的同一个实例，

仅在使用 portlet context 时有效。

4.Spring Bean的生命周期：

a.实例化一个Bean，也就是我们常说的new；

b.安装Spring上下文对实例化的Bean进行配置，也就是IOC注入；