spring 的架构:

spring框架的核心组件:core,context,beans

关键是把对象之间的依赖关系用配置文件来管理,依赖注入机制,注入关系在IOC容器中管理,ioc容器就是被bean包裹的对象,Spring正是把对象包装在bean中达到对这些对象的管理,及额外操作.

context就是发现每个bean之间的关系,建立关系并且维护好关系.context就是一个bean关系的容器,关系集合又叫ioc容器

core是为发现,建立,维护每个bean之间关系的一系列工具

bean创建时工厂模式.

beanfactory的接口 有可列表的,之间有集成关系的,能够自动装配规则的.

在spring的配置文件中定义一个<bean/>节点 在spring内部就会被转化成beandefinition对象,以后所有的操作都在这个对象内完成

applicationContext是context的顶级父类以继承了beanfactory,说明主体对象是bean,

实现了resourceLoader接口,使得applicationcontext可以读取任何外部资源

applicationcontext主要创建一个应用环境,利用beanfactory创建bean对象,保存对象关系白表,捕获各种事件.

spring的设计模式;

工厂模式,单例模式,模板模式,策略模式,

代理模式,jdk动态代理(接口实现)和cglib代理

jdk和cglib代理方式的使用使用了策略模式,

依赖注入方式:构造器依赖注入,setter方法依赖注入,通过使用构造器注入实现强制依赖注入,通过setter方法实现可选依赖注入.

给spring容器配置元数据,xml配置文件,基于注解的配置,基于java的配置.

spring bean的作用域:

singleton:在每个ioc容器中只有一个实例;

prototype:一个bean的定义可以有多个实例;

request:每次http请求都会创建一个实例;

session:在每一次会话,一个bean对应一个实例;

global-session:全局httpsession中,一个bean对应一个实例.

spring的单例bean不是线程安全的.

bean生命周期方法:setup,容器加载bean的时候被调用,

teardown,容器卸载类的时候被调用,

bean标签的两个重要属性:init+method destory-method

通过这两个属性可以自己定制初始化和销毁的方法,有相应的注解postConstruct,predestory

spring 注入java集合

list set map props

自动装配的五种方式:

no bytype byname constructor autodetect(先用构造器,如果无法实现,再用type)

@configuration 用来标记类可以作为一个bean的定义,被ioc容器使用

@bean 此方法返回一个对象,作为一个bean注册金spring应用上下文

开启注解扫描

<context:annotation-config/>

spring访问hibernate

1.控制反转 hibernate template和callback

2.继承hibernatedaosupport提供的aop拦截器

applicationcontext:

1.默认初始化所有的singleton,也可以取消预初始化

2.继承messagesource,支持国际化

3.实现resourceloader接口,进行资源访问,访问url和文件

4.事件机制

5.可以加载多个配置文件

6.以申明方式启动并创建spring容器

7.载入多个上下文

beanfactory采用延迟加载注入bean编程方式创建手动注册,ac是启动时全部创建申明方式创建自动注册,