# spring笔记

spring就是对象工厂,以后我们程序中用到的对象全部由spring创建

工厂模式:使用第三方的类(工厂类)产生我们需要的对象,为了方便解耦合

人吃苹果!

简单工厂:

1. 定义产品接口,所有产品必须实现该接口
2. 定义工厂接口,接口提供方法返回产品的接口类型,提供参数以便产生不同产品对象
3. 编写工厂实现类通过判断参数从而产生不同的产品对象

工厂方法模式:总体的思想就是一个产品单独对应一个工厂

简单工厂+反射:

springIOC基本使用

1. 复制springjar分类中的核心包到项目中
2. 编写spring主配置文件(IOC容器)
3. 在配置文件里面配置那些对象需要交给IOC容器管理

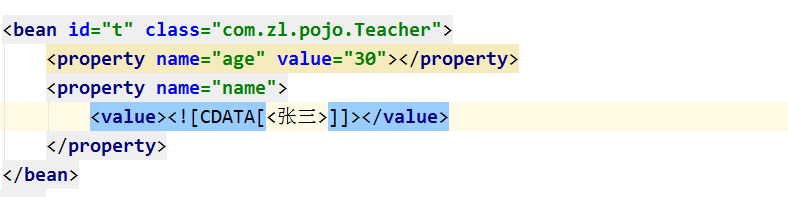
IOC:控制反转,反转资源获取的方向,之前我们需要一个对象需要我们自己new对象,先全部对象由spring帮我们创建,我们用的时候从IOC容器获取

关于bean的生命周期:

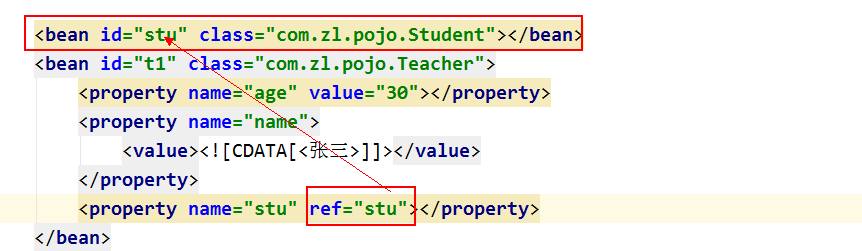
1. 当启动IOC容器的时候,默认的spring会帮我们创建IOC容器中的所有bean对象
2. IOC容器里面的对象默认是单例状态
3. spring帮我们创建对象的时候默认调用的是无参构造方法

DI:依赖注入,在配置bean的同时给对象注入一些初始值

1. 注入普通类型(基本数据类型以及String)的值



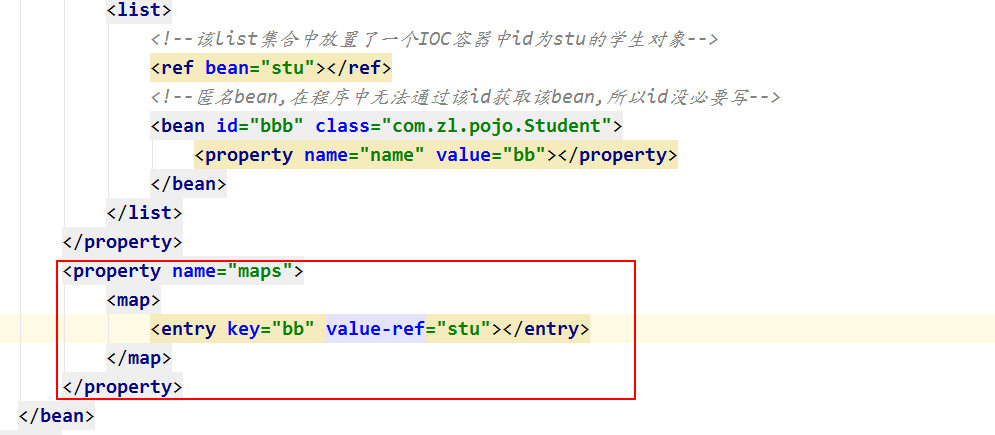
1. 给对象类型的属性注入值



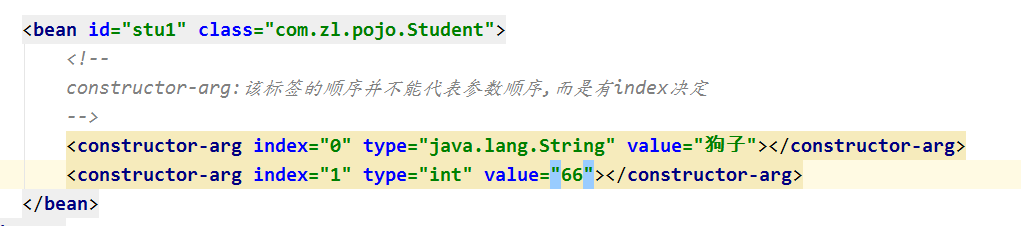
1. 给list集合类型属性注入值



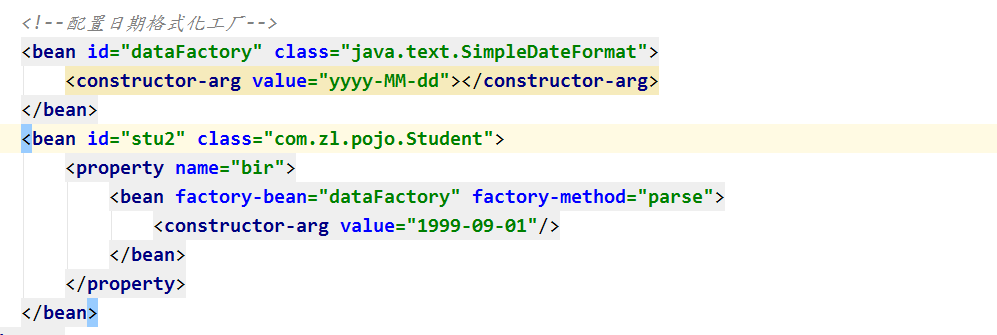
1. 给map集合注入值



1. 通过指定构造方法创建对象



1. 给date类型的属性注入值



自动扫描:会自动把指定的类(通过注解)注册成bean方到IOC容器中

1. 导入核心包以及AOP相关的jar包
2. 在需要自动放到IOC容器中的类上面添加注解

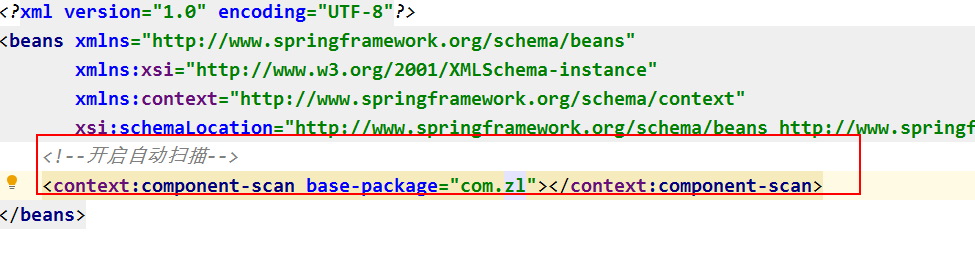
@Component: 基本注解, 标识了一个受 Spring 管理的组件

@Respository: 标识持久层组件

@Service: 标识服务层(业务层)组件

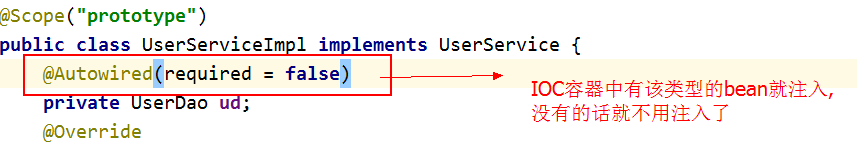
@Controller: 标识表现层组件

1. 指定spring扫描那些包



自动装配:@Autowired,会自动的把IOC容器里面的bean注入给有该注解是属性

1. 默认使用了该注解的属性,在IOC容器中必须有一个与之对应类型的bean



1. 首先spring会根据类型注入,通过类型无法确定注入那个bean的时候根据ID注入,自动注入id和属性名称一致的bean
2. 使用@Autowired注解给属性自动装配的时候,即使属性是私有的也不需要提供getset方法
3. 通过@Qualifier指定注入那个id的bean

注意:

1. 通过注解放到IOC容器里面的bean id默认是类名首字母小写
2. 通过通过注解参数指定该bean 的id
3. 可以同@Scope注解设置bean的作用域范围

## spring和mybatis整合

把mybatis里面涉及到的对象交给spring管理

整合要素:

1. 连接池对象交给spring(配置在spring的配置文件里面)
2. SqlSessionFactory对象
3. 通过spring给我们生成dao层接口实现类,并且自动注册成bean,放到IOC容器里面

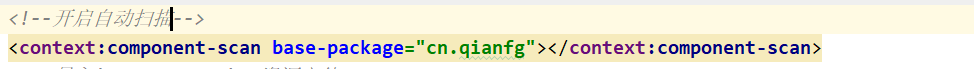
整合步骤:

1. 导入相关jar包
   * 1. spring核心
     2. springAOP
     3. spring事务管理
     4. spring和mybatis整合的中间包
     5. mybatis相关jar包
     6. 使用C3P0数据库连接池
     7. 数据库驱动包
2. 编写配置文件
   * 1. 编写mybatis主配置文件

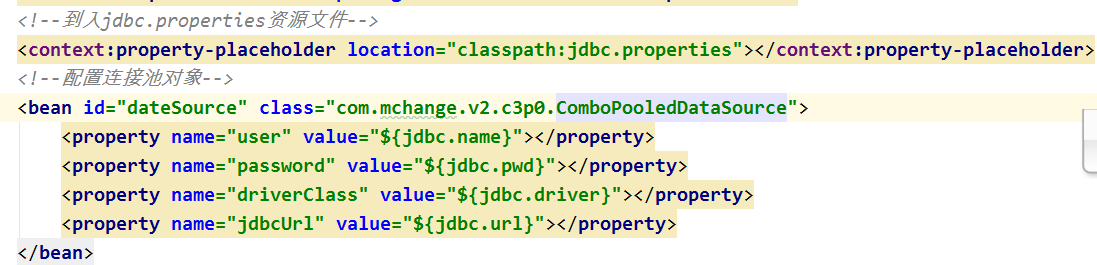


* + 1. 编写spring主配置文件

1. 开启注解扫描



1. 配置数据库连接池



1. 配置SqlSessionFactoryBean对象



1. 配置



## spring和servlet

整合步骤:

1. 导入springweb(springMVC)模块相关的jar包
2. 在web.xml里面配置,让tomcat启动的时候帮我们创建IOC容器



1. 创建BaseServlet

