# 环境搭建

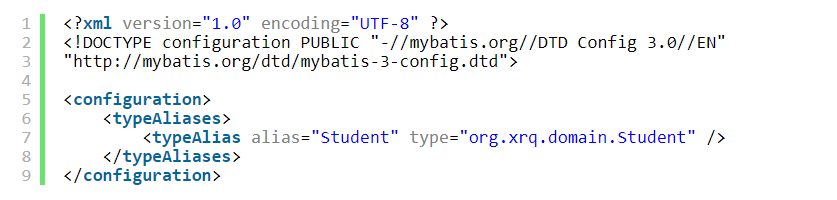
## Jar包准备

1. Spring环境搭建，beans，context，core，expression，common-logging
2. 注解包，javax.annotation jsr250-api 1.0
3. Spring集成Mybatis，MyBatis-Spring-1.x.0.jar
4. jdbc、tx、aop，jdbc 是基本的不多说，用到 tx 和 aop 是因为 Spring 对 MyBatis 事务管理的支持是通过 aop 来实现的
5. aopalliance.jar，这个是使用 Spring AOP 必要的一个 jar 包
6. 数据库连接池，dbcp、c3p0 都可以使用，这里使用的是阿里的 druid

com.alibaba druid 1.1.4

## Mybatis配置文件

只保留别名部分内容，<typeAliases>



## Spring配置文件

Spring-mybatis.xml

配置JDBC连接，数据库连接池，事务管理，注解扫描

# Spring整合Mybatis

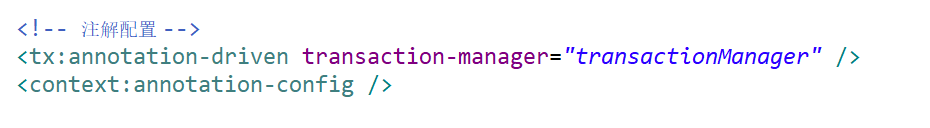
## 工作流程

1. 将配置文件整合到spring-mybatis中，只需加载这个就行。
2. Controller层作为程序入口，调用Service层。使用@Controller
3. Service层主要进行逻辑处理，实例化持久层（DAO）层对象，调用访问数据库方法。使用@Service。
4. 持久层作为访问数据库的直接程序，一般一个方法对应一条sql。在Mybatis中，接口和xml中sql是一一对应的。使用注解@MapperScan

## Spring自动化注解

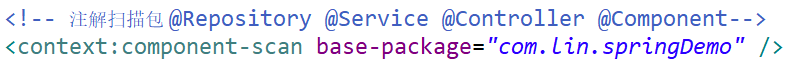
* 1. @Resource @Autowired

依赖注入的注解，实例化对象。



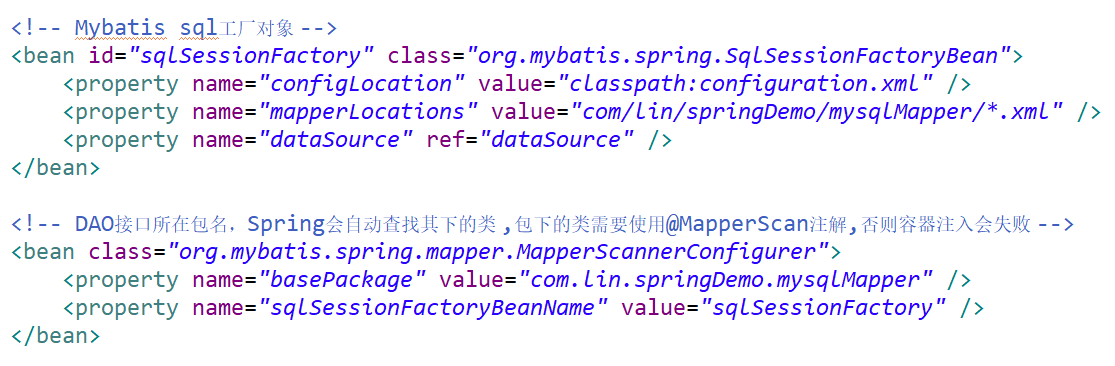
* 1. 组件注解

1. @Repository 注解，对应的是持久层即 Dao 层，其作用是直接和数据库交互，通常来说一个方法对应一条具体的 Sql 语句
2. @Service 注解，对应的是服务层即 Service 层，其作用是对单条 / 多条 Sql 语句进行组合处理，当然如果简单的话就直接调用 Dao 层的某个方法了
3. @Controller 注解，对应的是控制层即 MVC 设计模式中的控制层，其作用是接收用户请求，根据请求调用不同的 Service 取数据，并根据需求对数据进行组合、包装返回给前端
4. @Component 注解，这个更多对应的是一个组件的概念，如果一个 Bean 不知道属于拿个层，可以使用 @Component 注解标注



* 1. Mybatis注解

@MapperScan注解用在mapper接口上，让@Autowired能够实例化该接口，和Mybatis中SqlSessionFactory.getSession.getMapper（\*.class）是一样的。Spring中用在配置文件中的方式在Bean容器中注入了SqlSessionFactory和Session。

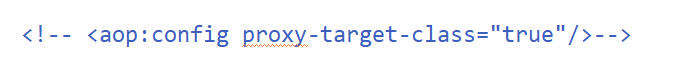


# Spring事务

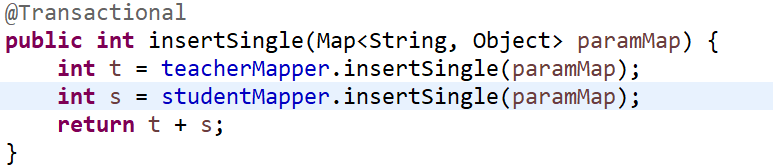
## @Transactional 注解

1. 这个注解用于开启事务管理，注意 @Transactional 注解的使用前提是该方法所在的类是一个 Spring Bean，因此（1）中的 @Service 注解是必须的。换句话说，假如你给方法加了 @Transactional 注解却没有给类加 @Service、@Repository、@Controller、@Component 四个注解其中之一将类声明为一个 Spring 的 Bean，那么对方法的事务管理，是不会起作用的。
2. 将JDK动态代理更换成Cglib代理

JDK动态代理要求该类实现接口，而应用了事务注解的类却没有被spring认为是实现接口，需要阅读AOP源码才明白。



## 具体实例代码



1. 被注解的方法代表一个事务，如果t成功和s成功才代表事务执行成功，才会提交结果。如果有一个不成功，那就表示不成功。

## @Transactional属性

1. @Transactional(propagation = Propagation.REQUIRED)

最重要的先说，propagation 属性表示的是事务的传播特性



1. @Transactional(isolation = Isolation.DEFAULT)

事务的隔离级别



1. @Transactional(readOnly = true)

该事务是否为一个只读事务，配置这个属性可以提高方法执行效率。

1. @Transactional(timeout = 30)

事务超时时间，单位为秒。如果超时就回滚事务。