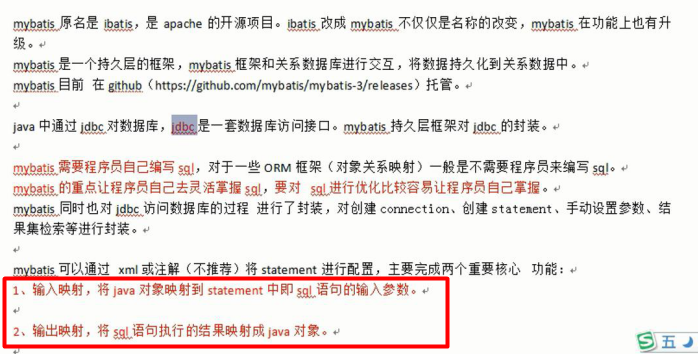
# Mybatis

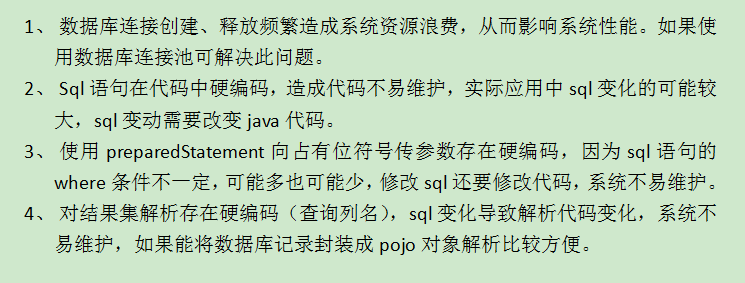
## mybatis介绍

mybatis是一个持久层的框架，是和关系型数据库交互的。



mybatis不是一个完全的ORM框架，因为需要我们自己编写sql

## JDBC在开发中的问题



## 框架原理



## 入门

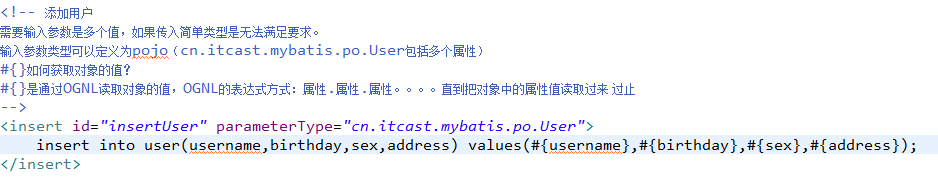
### 环境搭建

导入jar，配置日志文件

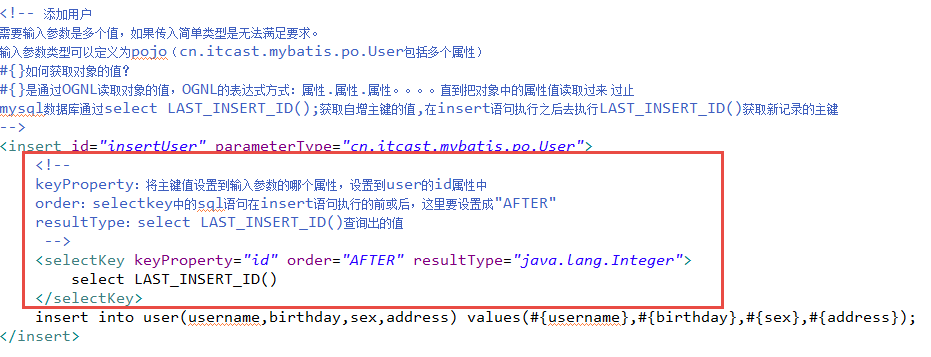
### 配置

1. SqlMapConfig.xml配置数据源等
2. Mapper.xml配置sql语句等

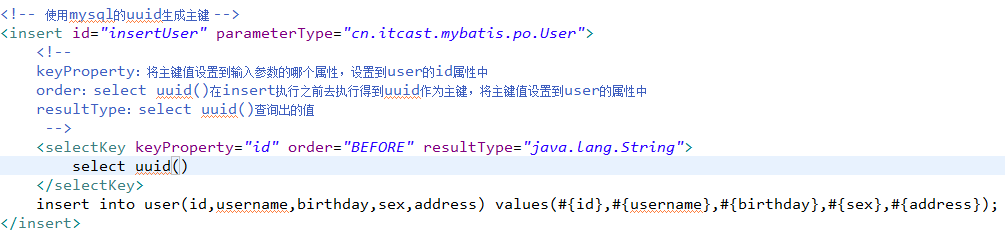
#### 添加用户



#### 数据库自增的返回



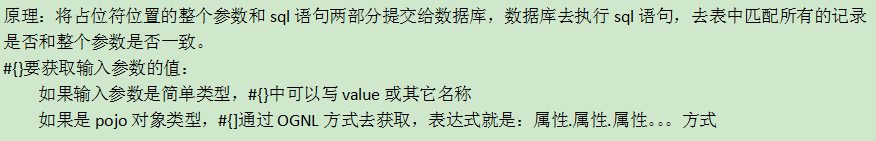
#### uuid



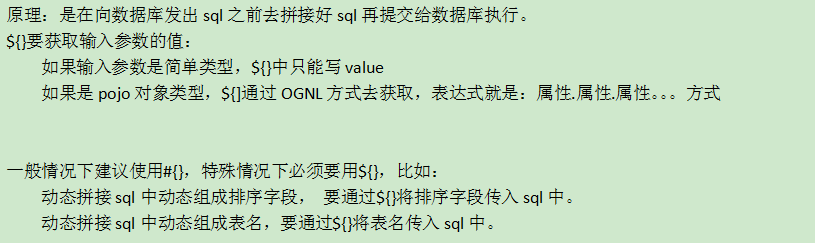
### 总结

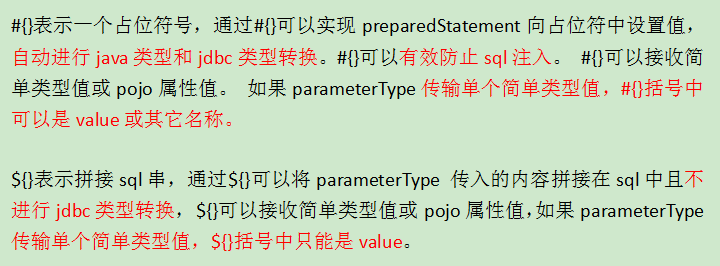
#### #{}和${}

#{}：表示一个占位符，避免sql注入



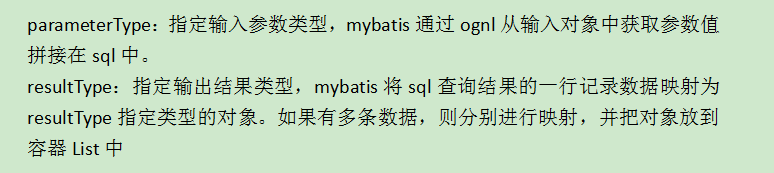
${}：表示sql拼接符





#### parameterType和resultType





## Mybatis开发Dao

### 传统方式

存在问题

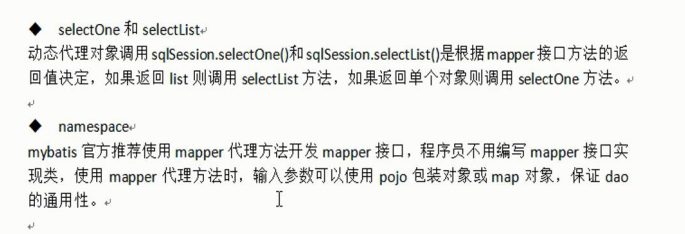
代码重复 硬编码

### mapper代理

规范

1. mapper.xml  
   namespace：接口全路径  
   接口方法的名称需要和statement的id一致  
   parameterType：与接口方法输入参数类型一致  
   接口方法的返回值类型和statement中的resultType类型一致
2. SqlMapConfig.xml引入mapper.xml
3. 测试 不需要自己实现类

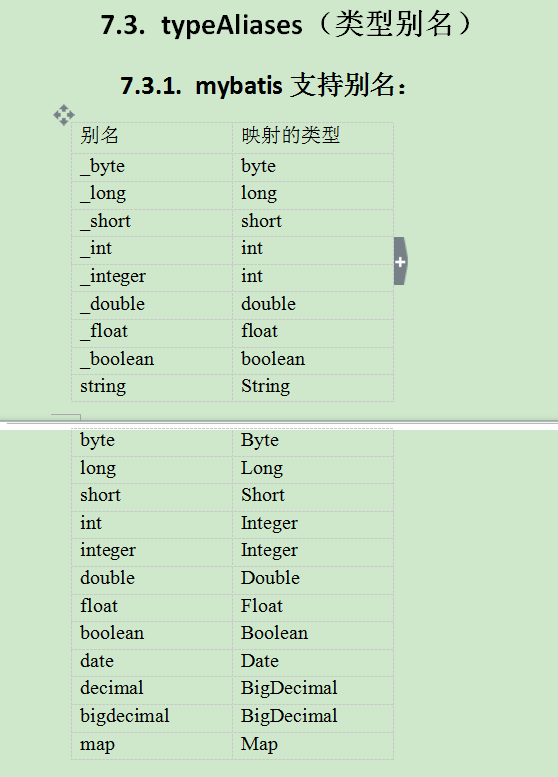
Mapper总结：



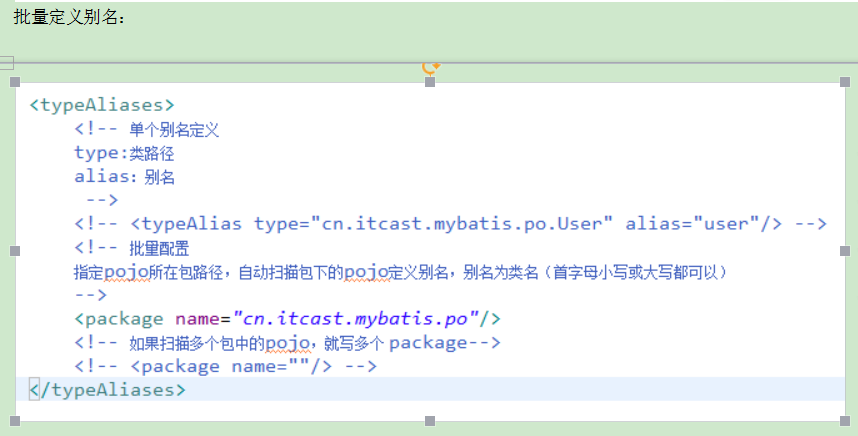
## SqlMapConfig.xml



1. properties可以用于连接数据库信息替换
2. settings  
   
3. typeAliases





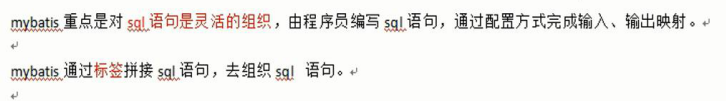


1. mappers

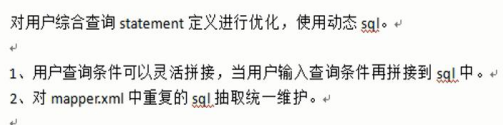


## 动态sql

### 什么是动态sql

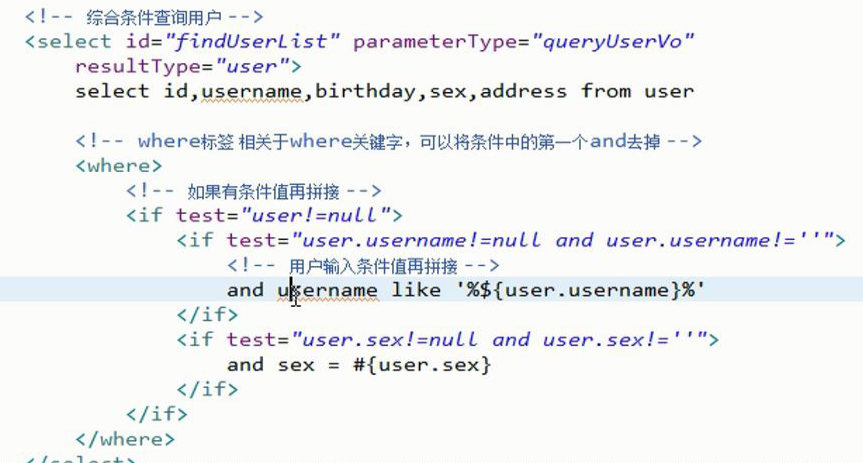


### 需求



### 栗子

解决上面第一个问题



### sql片段

解决第二个问题

## 关联查询

mybatis实现表与表之间的查询，将查询结果。。。

### 一对一

association



### 一对多

collection



### 多对多

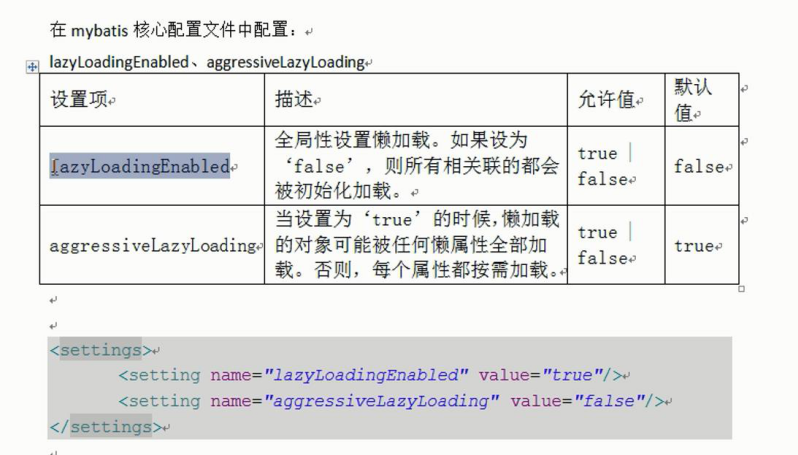
collection中套collection套association



### 延迟加载

如果刚开始仅仅查询一些简单地数据能够满足需求时，尽量使用单表查询

当用户需要查询关联信息时，我们再关联查询关联信息，叫做延迟加载







## 一级缓存和二级缓存

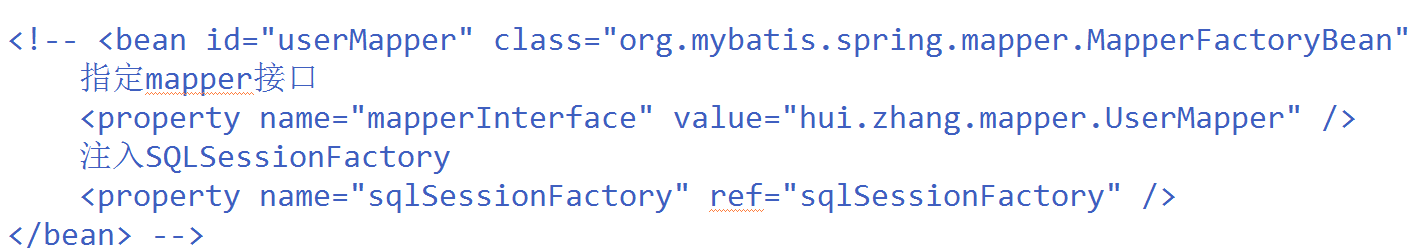
一级缓存：sqlSession范围内

二级缓存：mapper的namespace

## mybatis与spring整合

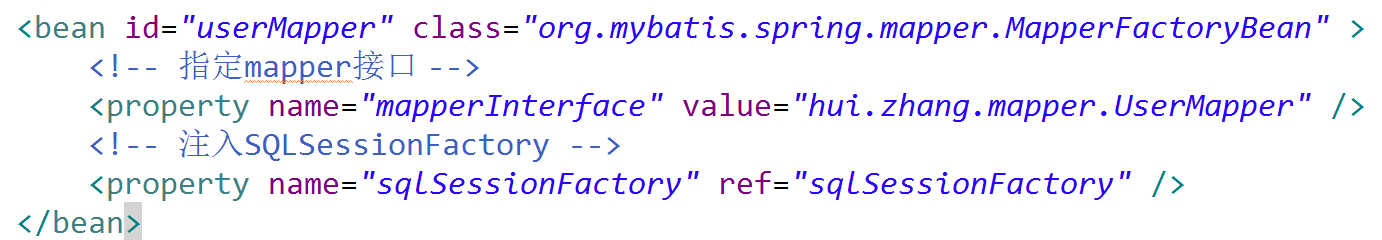
让spring管理SqlSessionFactory、原始dao实现及mapper接口代理对象

### 原始方法



测试类继承SqlSessionDaoSupport，直接getSqlSession()就可以获得sqlSession

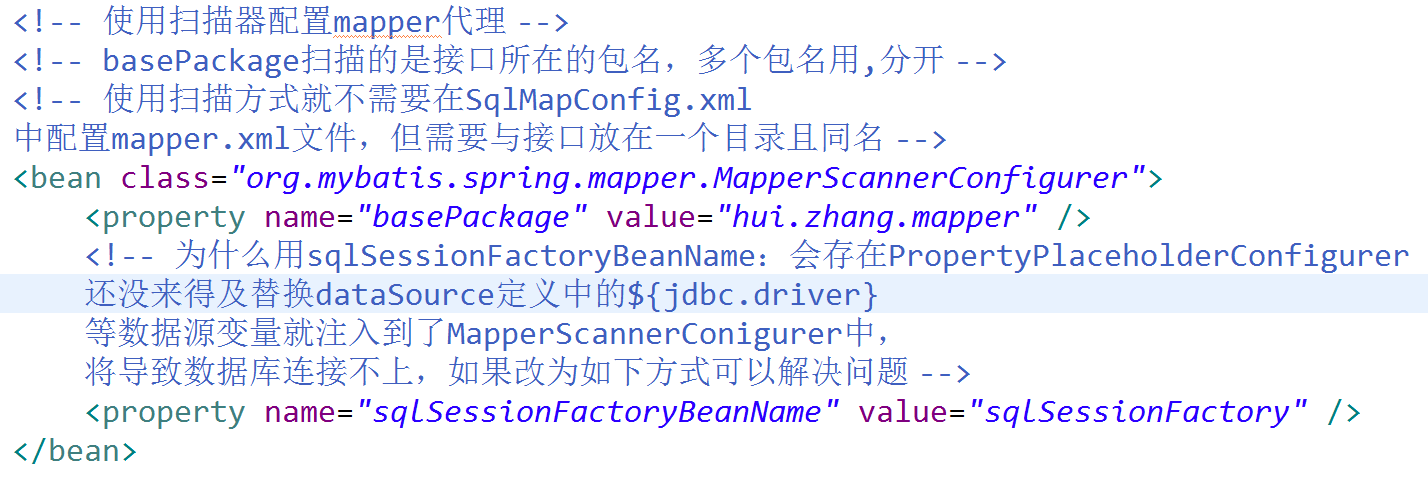
### mapper代理方式一



因为创建多个mapper就不断要注入sqlSessionFactory，麻烦 不建议用

### mapper代理方式二

扫描器



扫描出的接口名是类名首字母小写

## 逆向工程

mybatis官方提供根据数据库表生成执行代码的工具

### 配置逆向工程的配置文件

代码中有详细说明

网络查看：http://blog.csdn.net/h3243212/article/details/50778937

### 运行生成

逆向工程生成的代码只针对单表的crud操作