## Mybatis架构

1. 首先配置一个全局的配置文件sqlMapConfig.xml（名称不固定），在其中配置好数据源，事物以及要加载的Mapper.xml
2. 创建一个SqlSessionFactoryBuild,来创建一个SqlSessionFactroy（会话工厂），作用是创建一个sqlSession
3. 创建一个SqlSession,（面向用户的接口，操作数据库的方法）操作数据库
4. Executor执行（数据库操作的执行器），有两个接口实现（默认执行器和缓存执行器）
5. MappedStatement（mybatis的封装对象）

封装：sql语句(占位符)

1. 到mysql

## 在mapper.xml中的namespace的值

在sqlSession中的方法中的参数String statement，里面的值为namespace + 配置文件中的id,所以在配置namespace要注意

## mybatis开发小结

1、编写SqlMapConfig.xml

2、编写mapper.xml

定义了statement

3、编程通过配置文件创建SqlSessionFactory

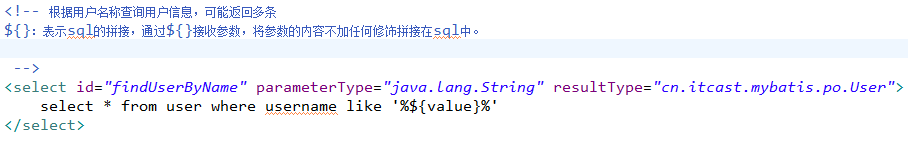
4、通过SqlSessionFactory获取SqlSession

5、通过SqlSession操作数据库

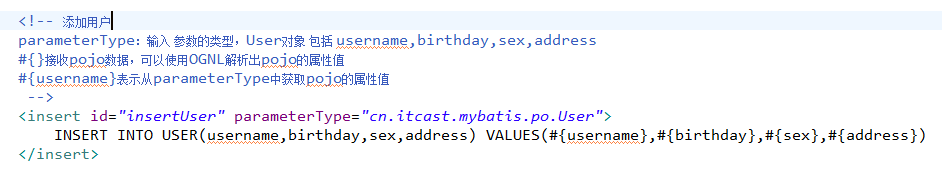
如果执行添加、更新、删除需要调用SqlSession.commit()

6、SqlSesion使用完成要关闭

## 4、使用${ }接收参数



## statement中接收参数



通过使用#{ }便可接收数据，但是#{}中的参数名与方法中的参数的属性名一致

另一种就是通过注解的方法来获取参数

## mapper开发遵守的规则

1. 首先mapper.xml中的namespace需要时mapper.java的全限定名
2. Mapper.xml中statement中的id是mapper.java中的方法名
3. Mapper.xml中statement中的resultType返回值和mapper.java中的返回值必须相同
4. Mapper.xml中statement中的parameterType参数值和mapper.java中的参数值相同

## 6、动态sql

#{}和${}完成输入参数的属性值获取，通过OGNL获取parameterType指定pojo的属性名。

#{}：占位符号，好处防止sql注入

${}：sql拼接符号

if

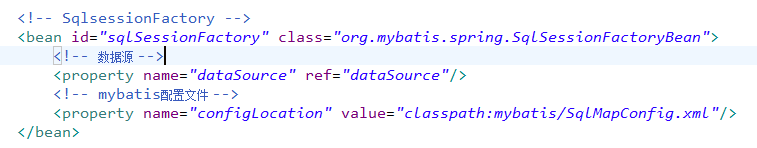
where

foreach

## 7、mybatis和Spring整合开发

在application中配置好数据源datasource,然后再在里面配置好SQLSessionFactory

开发原始Dao（不是自动扫包）



在sqlMapConfig中配置的是一些mapper.xml

在Dao中需要继承SqlSessionDaoSupport，然后获得一格sqlSession来操作数据库

注：因为都是交给spring管理的，所以在配置dao的bean时需要注入sqlSession