# MyBatis简介

## 认识MyBatis

1）MyBatis是主流的Java持久层框架之一，它与Hibernate一样，也是一种ORM

（Object/Relational Mapping，即对象关系映射）框架。

2）MyBatis本事Apache的一个开源项目，原名iBatis，后改名为MyBatis。

3）MyBatis是一个基于Java的持久层框架。MyBatis提供的持久层框架包括SQL Maps

和Data Access Object（DAD）它几乎消除了所有的JDBC代码和参数的手工设置以及

结果集的检索。MyBatis使用简单的XML或注解用于配置和原始映射，将接口和Java

的POJOs（Plain Old Java Objects，普通的Java对象）映射成数据库中的记录。

## Mybatis与Hibernaete

Mybatis与Hibernate都是当前比较流行的基于Java的持久层框架，都是一种ORM （Object/Relational Mapping）框架。但两者差异如下：

1. MyBatis是一个半自动映射框架，因为MyBatis需要手动匹配POJO、SQL和映射关系；

而Hibernate是一个全表映射（全自动映射）框架，只需要提供POJO和映射关系即

可。

1. MyBatis是一个小巧、方便、高效、简单、直接、半自动化的持久层框架；Hibernate 是一个强大、方便、高效、复杂、间接、全半自动化的持久层框架。

这两个框架各有优缺点，开发者应根据实际应用选择。

# MyBatis开发环境的搭建

## 下载

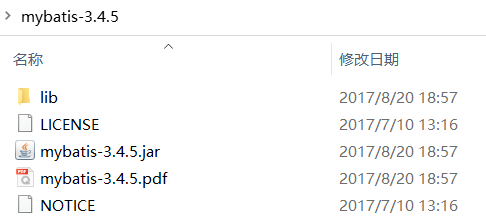
可以从以下地址下载对应版本的MyBatis，我下载的是3.4.5

**https://github.com/mybatis/mybatis-3/releases**

在下载时只需要下载mybatis-3.4.5.zip即可。

## 目录结构

1. 下载解压缩后，其目录结构如下：



1. mybatis-3.4.5.jar是MyBatis的核心包
2. mybatis-3.4.5.pdf是MyBatis手册
3. lib目录中的jar包是MyBatis的依赖包

## 安装

在使用MyBatis框架时需要将它的核心包（mybatis-3.4.5.jar）和依赖包(lib目录中的 所有jar包)复制到WEB-INF/lib目录中即可。

# MyBatis的工作原理

了解MyBatis的工作原理对于程序的理解很有帮助，MyBatis工作原理如下：  
 1）读取MyBatis配置文件。MyBatis配置文件是mybatis-config.xml，它配置了MyBatis 的运行环境、数据库连接信息等。

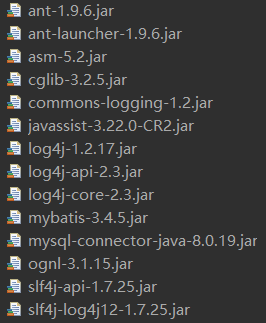
2）加载映射文件。映射文件指的是SQL映射文件，通常是XML格式的文件，该文件中配置了操作数据库的SQL语句，需要在MyBatis配置文件mybatis-config.xml中加载。Mybatis-config.xml 配置文件可以加载多个映射文件，每个映射文件对应数据库中的一张表。

1. 构造会话工厂。通过MyBatis的环境等配置信息构造会话工厂SqlBeanFactory。
2. 创建SQL会话对象。由会话工厂创建SqlSession对象，该对象中包含了执行SQL语 句的所有方法。
3. Executor执行器。MyBatis底层定义了一个Executor接口来操作数据库，它将根据 SqlSession传递的参数动态地生成需要执行的SQL语句，同时会负责查询缓冲的维 护。
4. MapperStatement对象。在Executor接口的的执行方法中有一个MappedStatement 类型的参数，该参数是对映射信息的封装，用于存储要映射的的SQL语句的id、参 数等信息。
5. 输入参数映射。输入参数可以是Map、List等集合类型，也可以是基本数据类型和 POJO类型。输入参数映射过程类似于JDBC对preparedStatement对象设置参数的过 程。
6. 输出结果映射。输出结果类型可以是Map、List等集合类型，也可以是基本类型和 POJO类型。输出结果映射过程类似于JDBC对结果集的解析过程。

# 使用Eclipse开发MyBatis入门程序

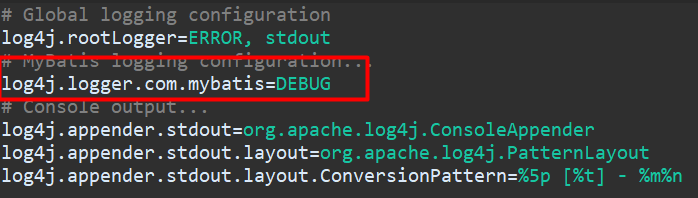
1. **创建Web应用，并添加相关JAR包。**

在Eclipse中创建一个名为ch6的Web应用，将MyBatis的核心JAR包，依赖JAR包以及MySQL数据库驱动JAR包一起复制到/WEB-INF/lib目录下。添加后的lib目录如下图所示：



1. **创建日志文件**

MyBatis默认使用log4j输出日志信息，如果开发者需要查看控制台输出的SQL语句，那么需要在classpath路径下配置其日志文件。在ch6应用的src目录下创建log4j.properties文件，其内容如下：

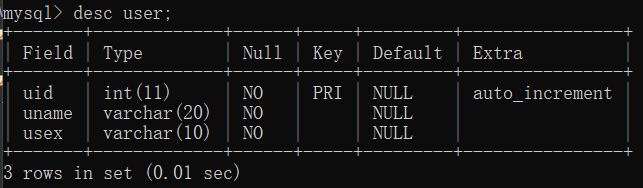


该配置文件内容不需要开发者手写，可以从MyBatis使用手册中的Logging小节复制，然后进行简单修改。

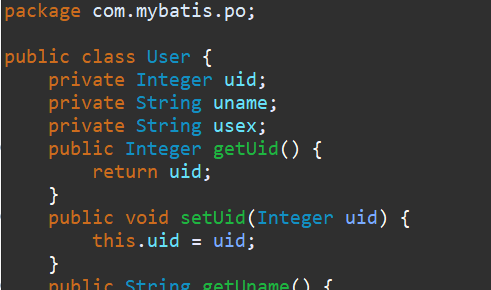
在日志文件中配置了全局的日志配置、MyBatis的日志配置和控制台输出，其中MyBatis的日志配置用于将com.mybatis包下的所有类的日志记录级别设置为DEBUG。

1. **为数据表创建持久化类**

user表如下：

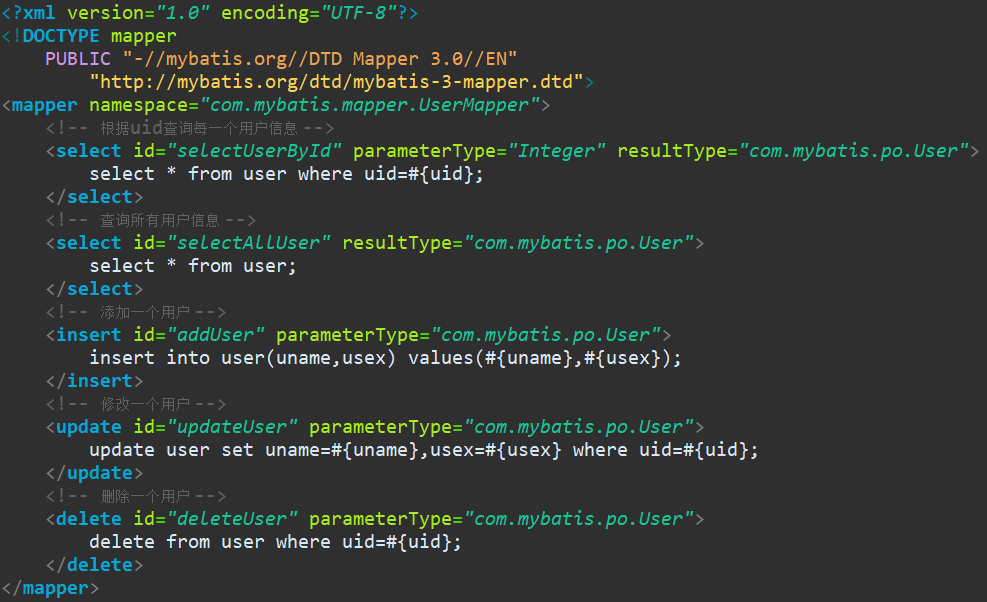


在src目录下创建一个名为com.mybatis.po的包，在该包中创建持久化类User，注意该类中声明的属性要与数据表user的字段一致。代码如下：



**4）创建映射文件**

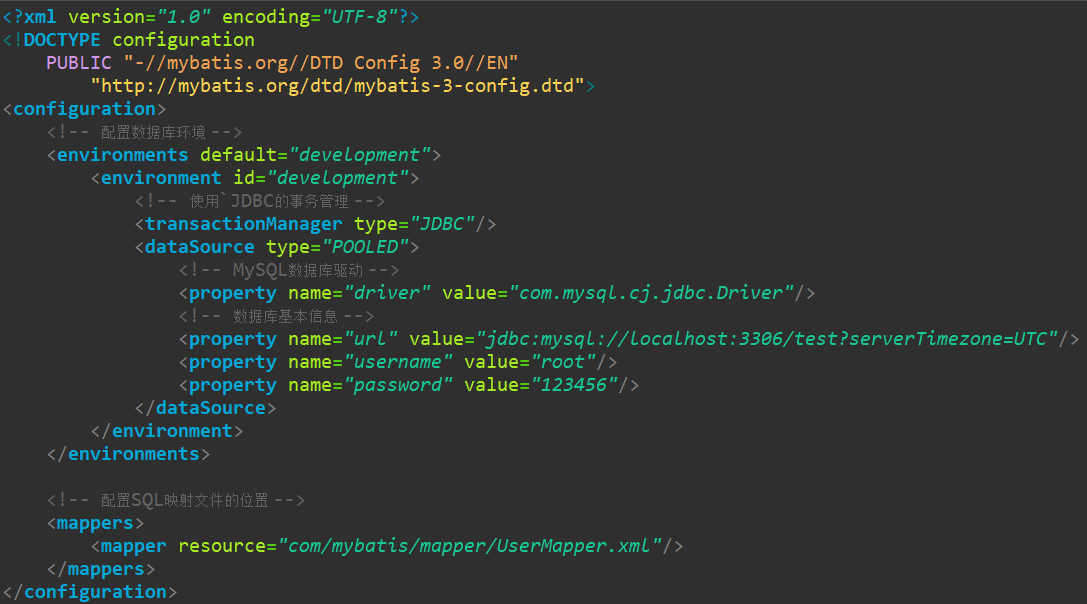
在src目录下创建一个名为com.mybatis.mapper的包，在包中创建映射文件UserMapper.xml文件，对应着user表。代码如下：



在上述映射文件中，<mapper>元素时配置文件的根元素，它包含了一个namespace属性，该属性值通常设置为“包名+SQL映射文件名或接口名”，指定了唯一的命名空间；子元素<select>、<insert>、<update>、<delete>中的信息用于执行查询、添加、修改以及删除操作的配置。在定义的SQL语句中，“#{ }”表示一个占位符，相当于“？”，而“#{uid}”表示该占位符接收参数的名称为uid。

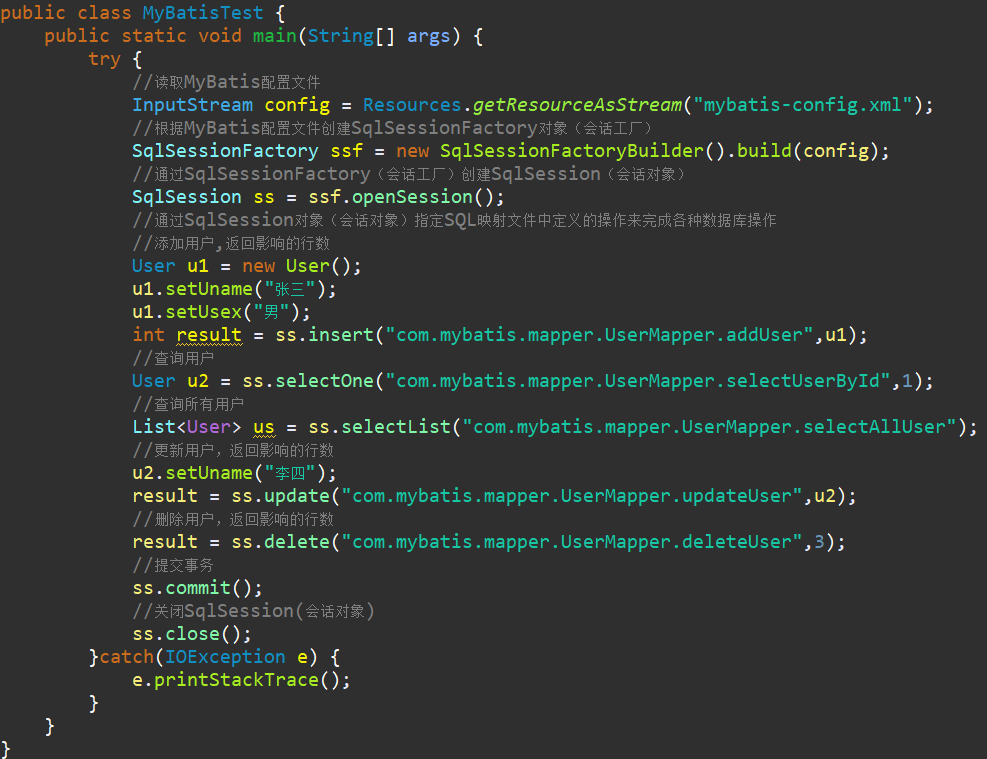
1. **创建MyBatis的全局配置文件**

在scr目录下创建MyBatis的核心配置文件mybatis-config.xml，在该文件中配置了数据库环境和映射文件的位置，具体内容如下：

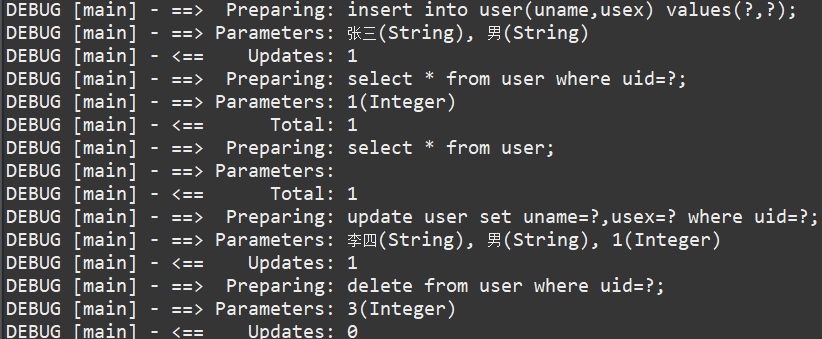


1. **创建测试类**

在src目录下创建一个名为com.mybatis.test包，在该包中创建MyBatisTest测试类。在测试类中首先使用输入流读取配置文件，然后根据配置信息构建SqlSessionFactory对象（会话工厂）。再通过SqlSessionFactory对象创建SqlSession对象（会话对象），并使用SqlSession对象的方法执行数据库操作。



运行结果如下：



**MyBatis与Spring的整合：**

1. 搭建Spring开发环境
2. 搭建MyBatis开发环境
3. 导入MyBatis与Spring整合的中间包
4. 导入数据库驱动包
5. 导入数据源所需要的JAR包

**使用MyBatis Generator插件自动生成映射文件：**

使用MyBatis Generator插件自动生成MyBatis所需要的DAO接口、实体模型类、 Mapper映射文件会省去很多时间，将生成的代码复制到项目工程中即可，这样可以把精力更多的放在业务逻辑上。

MyBatis Generator自动生成代码有3种常用方法，一是使用命令行、二是使用Eclipse 插件、三是使用Maven插件。在这里主要讲解使用命令行的方式。

（1）准备相关的JAR包。

需要准备的JAR包是mysql-connector-java-5.1.45-bin.jar和 mybatis-generator-core-1.3.6.jar（可以从 http:mvnrepository.com/artifact/org.mybatis.generator/mybatis-generator-core/1. 3.6下载）

1. 在某磁盘根目录新建一个文件目录，例如D:\generator，并将准备的JAR包复 制到generator目录中。另外，在generator目录下创建src文件夹用于存放生 成的相关代码文件。
2. 在generator目录中创建配置文件，例如D:\generator\generator.xml，该文件 中的内容为：
3. 打开命令行，进入D:\generator，输入命令

java -jar mybatis-generator.core-1.3.6.jar-configfile generator.xml-overwrite

即可在D:\generator\src目录中看到生成的.java文件。