

2018

Java EE框架 ---MyBatis

Java EE framework --MyBatis

王磊

部门：研发部

CONTENTS



下载jar包



搭建MyBatis开发环境



配置sqlMapConfig.xml



部署运行


从官网下载MyBatis jar包

mybatis – MyBatis 3 | Introductio

www.mybatis.org/mybatis-3/

淘宝购物 百度一下 爱淘宝 天猫商城 京东商城 一号店

mybatis

 **MyBatis**

Last Published: 12 March 2018 | Version: 3.4.7-SNAPSHOT

REFERENCE DOCUMENTATION

Introduction

Getting Started

Configuration XML

Mapper XML Files

Dynamic SQL

Java API

SQL Builder Class

Logging

PROJECT DOCUMENTATION

Project Information

CI Management

Dependencies

Dependency Information

Distribution Management

About

Issue Management

Licenses

Mailing Lists

Plugin Management

Plugins

Team

Introduction

What is MyBatis?

MyBatis is a first class persistence framework with support for custom SQL, stored procedures and advanced mappings. MyBatis eliminates almost all of the JDBC code and manual setting of parameters and retrieval of results. MyBatis can use simple XML or Annotations for configuration and map primitives, Map interfaces and Java POJOs (Plain Old Java Objects) to database records.

Help make this documentation better...






If you find this documentation lacking in any way, or missing documentation for a feature, then the best thing to do is learn about it and then write the documentation yourself!

Sources of this manual are available in xdoc format at [project's Git](#) Fork the repository, update them and send a pull request.

You're the best author of this documentation, people like you have to read it!

Translations

Users can read about MyBatis in following translations:

-  English
-  Español
-  日本語
-  한국어
-  简体中文

Do you want to read about MyBatis in your own native language? File an issue providing patches with your mother tongue documentation!

• MyBatis jar包 •

jar 包内容

名称



lib

依赖包



LICENSE



mybatis-3.2.7.jar

核心包



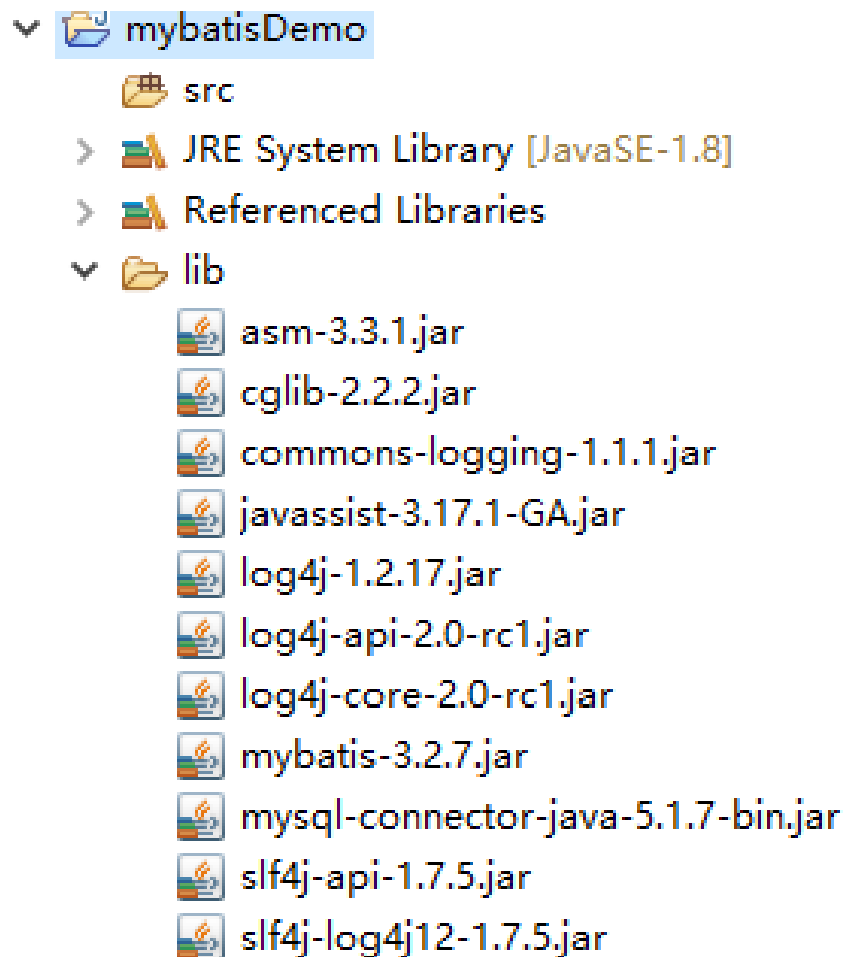
mybatis-3.2.7.pdf



NOTICE

一、创建Java项目

- 1、新建Java Project
- 2、引入Mybatis jar包
- 3、构建路径



二、创建javabean

创建javabean包，编写javabean

```
package com.xk.javabean;

public class User {
    private String id;
    private String username;
    private String password;
    private String grade;
    private String email;
    public String getId() {}
    public void setId(String id) {}
    public String getUsername() {}
    public void setUsername(String username) {}
    public String getPassword() {}
    public void setPassword(String password) {}
    public String getGrade() {}
    public void setGrade(String grade) {}
    public String getEmail() {}
    public void setEmail(String email) {}
    @Override
    public String toString() {
        return "User [id=" + id + ", username=" + username + ", password="
            + password + ", grade=" + grade + ", email=" + email + "];"
    }
}
```

三、编写Mybatis核心配置文件

编写Mybatis核心配置文件sqlMapConfig.xml

```
<?xml version="1.0" encoding="UTF-8" ?>
<!DOCTYPE configuration
PUBLIC "-//mybatis.org//DTD Config 3.0//EN"
"http://mybatis.org/dtd/mybatis-3-config.dtd">
<configuration>
  <environments default="development">
    <environment id="development">
      <!-- 使用jdbc事务管理 -->
      <transactionManager type="JDBC" />
      <!-- 数据库连接池 -->
      <dataSource type="POOLED">
        <property name="driver" value="com.mysql.jdbc.Driver" />
        <property name="url"
          value="jdbc:mysql://localhost:3306/usermanager?characterEncoding=utf-8" />
        <property name="username" value="root" />
        <property name="password" value="root" />
      </dataSource>
    </environment>
  </environments>
</configuration>
```

四、在src下创建log4j日志

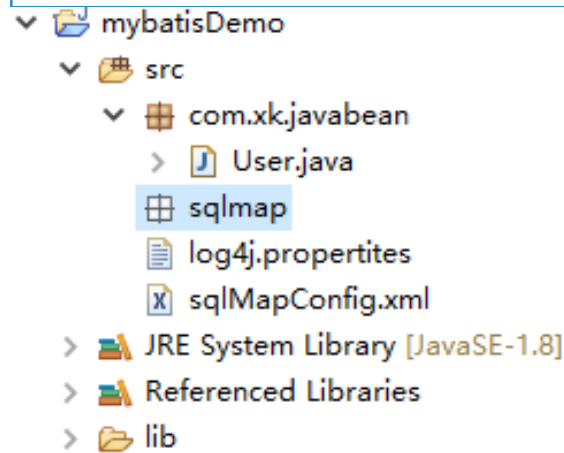
在src下创建log4j日志

- ✓ mybatisDemo
 - ✓ src
 - ✓ com.xk.javabean
 - > User.java
 - log4j.properties
 - sqlMapConfig.xml
 - > JRE System Library [JavaSE-1.8]
 - > Referenced Libraries
 - > lib

```
1 # Global logging configuration
2 log4j.rootLogger=DEBUG, stdout
3 # Console output...
4 log4j.appender.stdout=org.apache.log4j.ConsoleAppender
5 log4j.appender.stdout.layout=org.apache.log4j.PatternLayout
6 log4j.appender.stdout.layout.ConversionPattern=%5p [%t] - %m%n
```


• 五、创建sqlmap文件夹，创建user.xml •

在项目的src文件夹下添加一个“sqlmap”文件夹，然后添加user.xml文件



```
1 <?xml version="1.0" encoding="UTF-8" ?>
2 <!DOCTYPE mapper
3 PUBLIC "-//mybatis.org//DTD Mapper 3.0//EN"
4 "http://mybatis.org/dtd/mybatis-3-mapper.dtd">
5 <!-- 写Sql语句 -->
6 <mapper namespace="test">
7     <!-- 通过ID查询一个用户 -->
8     <select id="findUserById" parameterType="Integer" resultType="com.xk.javabean.User">
9         select * from tuserlogin where id = #{value}
10    </select>
11 </mapper>
```















• 六、在sqlMapConfig.xml中注册user.xml •

打开“sqlMapConfig.xml”，添加如下代码：

```
<?xml version="1.0" encoding="UTF-8" ?>
<!DOCTYPE configuration
PUBLIC "-//mybatis.org//DTD Config 3.0//EN"
"http://mybatis.org/dtd/mybatis-3-config.dtd">
<configuration>
  <environments default="development">
    <environment id="development">
      <!-- 使用jdbc事务管理 -->
      <transactionManager type="JDBC" />
      <!-- 数据库连接池 -->
      <dataSource type="POOLED">
        <property name="driver" value="com.mysql.jdbc.Driver" />
        <property name="url"
          value="jdbc:mysql://localhost:3306/usermanager?characterEncoding=utf-8" />
        <property name="username" value="root" />
        <property name="password" value="root" />
      </dataSource>
    </environment>
  </environments>
  <!-- mappers位置 -->
  <mappers>
    <mapper resource="sqlmap/user.xml"/>
  </mappers>
</configuration>
```

七、创建junit测试单元

在src下创建com.xk.junit测试包

- ▼  mybatisDemo
 - ▼  src
 - ▼  com.xk.javabean
 - >  User.java
 - ▼  com.xk.junit
 - >  MybatisTest.java
 - ▼  sqlmap
 -  user.xml
 -  log4j.properties
 -  sqlMapConfig.xml
 - >  JRE System Library [JavaS
 - >  Referenced Libraries
 - >  JUnit 4
 - ▼  lib

七、创建junit测试单元

```
public class MybatisTest {

    @Test
    public void test() throws Exception {
        //加载核心配置文件
        String resource = "sqlMapConfig.xml";
        InputStream in = Resources.getResourceAsStream(resource);
        //创建SqlSessionFactory
        SqlSessionFactory sqlSessionFactory = new
        SqlSessionFactoryBuilder().build(in);
        //创建SqlSession
        SqlSession sqlSession = sqlSessionFactory.openSession();
        //执行sql语句
        User user = sqlSession.selectOne("test.findUserById", 10);
        System.out.println(user);
    }
}
```

八、观察运行结果

```
DEBUG [main] - Logging initialized using 'class org.apache.ibatis.logging.slf4j.Slf4jImpl' adapter.
DEBUG [main] - PooledDataSource forcefully closed/removed all connections.
DEBUG [main] - PooledDataSource forcefully closed/removed all connections.
DEBUG [main] - PooledDataSource forcefully closed/removed all connections.
DEBUG [main] - PooledDataSource forcefully closed/removed all connections.
DEBUG [main] - Opening JDBC Connection
DEBUG [main] - Created connection 1354011814.
DEBUG [main] - Setting autocommit to false on JDBC Connection [com.mysql.jdbc.JDBC4Connection@50b494a6]
DEBUG [main] - ==> Preparing: select * from tuserlogin where id = ?
DEBUG [main] - ==> Parameters: 10(Integer)
DEBUG [main] - <==          Total: 1
User [id=10, username=陈军, password=123456, grade=1, email=chenj@126.com]
```

id	username	password	grade	email
8	司徒	123456	1	sit@126.com
9	wang	123456	1	wang@126.com
10	陈军	123456	1	chenj@126.com
11	张恒	123456	2	zhang@qq.com
12	刘建军	123456	1	liu@sina.com.cn
13	张扬菲	123456	1	zhangyf@126.com
14	杨杨	123456	1	yang@126.com
15	JavaEE	123456	1	java@163.com
16	oracle	123456	1	oracle@163.com

• 使用MyBatis访问数据库程序成功！ •



在此基础上，进一步拓展CRUD，写出更为复杂的业务逻辑