

MyBatis



作成者:周东

作成日:2019.08.30

Neusoft

目录

- ◆ 简介
- ◆ 配置
- ◆ Query
- ◆ CRUD

简介

MyBatis是支持普通SQL查询, 存储过程和高级映射的优秀持久层框架。MyBatis消除了几乎所有的JDBC代码和参数的手工设置以及对结果的检索封装。MyBatis可以使用简单的XML或注解用于配置和原始映射, 将接口和Java的POJO (Plain Old Java Objects, 普通的Java对象) 映射成数据库中的记录。

配置

创建MyBatis配置文件 - conf.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">
      <transactionManager type="JDBC" />
      <dataSource type="POOLED">
        <property name="driver" value="com.mysql.jdbc.Driver" />
        <property name="url" value="jdbc:mysql://localhost:3306/mybatis" />
        <property name="username" value="root" />
        <property name="password" value="123456" />
      </dataSource>
    </environment>
  </environments>
</configuration>
```

配置

创建持久化类 User.java

```
public class User {  
    private String id;  
    private String name;  
    private String password;  
    private Date createTime;  
    private Date expireTime;  
    ...getters/setters  
}
```

配置

查询SQL映射文件 - userMapper.xml

```
<?xml version="1.0" encoding="UTF-8" ?>
<!DOCTYPE mapper PUBLIC "-//mybatis.org//DTD Mapper 3.0//EN"
"http://mybatis.org/dtd/mybatis-3-mapper.dtd">
<mapper namespace="com.mybatis_test.userMapper">
  <select id="getUser" parameterType="int" resultType="com. mybatis_test.User">
    select * from users where id=#{id}
  </select>
</mapper>
```

配置

在conf.xml文件中注册userMapper.xml文件

```
<mappers>  
  <mapper resource="com/mybatis_test/userMapper.xml" />  
</mappers>
```

Query

查询

```
public static void main(String[] args) throws Exception{
    String resource = "conf.xml";
    Reader reader = Resources.getResourceAsReader(resource);
    SqlSessionFactory sessionFactory = new
    SqlSessionFactoryBuilder().build(reader);
    SqlSession session = sessionFactory.openSession();
    String statement = "com.mybatis_test.userMapper.getUser";
    User user = session.selectOne(statement, 1);
    System.out.println(user);
    session.close();
}
```


CRUD

插入SQL映射文件 - userMapper.xml

```
<?xml version="1.0" encoding="UTF-8" ?>
<!DOCTYPE mapper PUBLIC "-//mybatis.org//DTD Mapper 3.0//EN"
"http://mybatis.org/dtd/mybatis-3-mapper.dtd">
<mapper namespace="com.mybatis_test.userMapper">
    <insert id="addUser" parameterType="com.mybatis_test.User">
        insert into users (id, name, age) values (#{id}, #{name}, #{age})
    </insert>
</mapper>
```

CRUD

插入

```
public static void main(String[] args) throws Exception{
    String resource = "conf.xml";
    Reader reader = Resources.getResourceAsReader(resource);
    SqlSessionFactory sessionFactory = new
    SqlSessionFactoryBuilder().build(reader);
    SqlSession session = sessionFactory.openSession();
    String statement = "com.mybatis_test.userMapper.addUser";
    int insert = session.insert(statement, new User("00001", "张一", 25));
    session.commit();
    System.out.println(insert);
    session.close();
}
```

CRUD

更新SQL映射文件 - userMapper.xml

```
<?xml version="1.0" encoding="UTF-8" ?>
<!DOCTYPE mapper PUBLIC "-//mybatis.org//DTD Mapper 3.0//EN"
"http://mybatis.org/dtd/mybatis-3-mapper.dtd">
<mapper namespace="com.mybatis_test.userMapper">
    <update id="updateUser" parameterType="com.mybatis_test.User">
        update users set name=#{name}, age=#{age} where id=#{id}
    </update>
</mapper>
```

CRUD

更新

```
public static void main(String[] args) throws Exception{
    String resource = "conf.xml";
    Reader reader = Resources.getResourceAsReader(resource);
    SqlSessionFactory sessionFactory = new
    SqlSessionFactoryBuilder().build(reader);
    SqlSession session = sessionFactory.openSession();
    String statement = "com.mybatis_test.userMapper.updateUser";
    int update = session.update(statement, new User("00001", "王五", 26));
    session.commit();
    System.out.println(update);
    session.close();
}
```

CRUD

删除SQL映射文件 - userMapper.xml

```
<?xml version="1.0" encoding="UTF-8" ?>
<!DOCTYPE mapper PUBLIC "-//mybatis.org//DTD Mapper 3.0//EN"
"http://mybatis.org/dtd/mybatis-3-mapper.dtd">
<mapper namespace="com.mybatis_test.userMapper">
    <delete id="deleteUser" parameterType="int">
        delete from users where id = #{id}
    </delete>
</mapper>
```

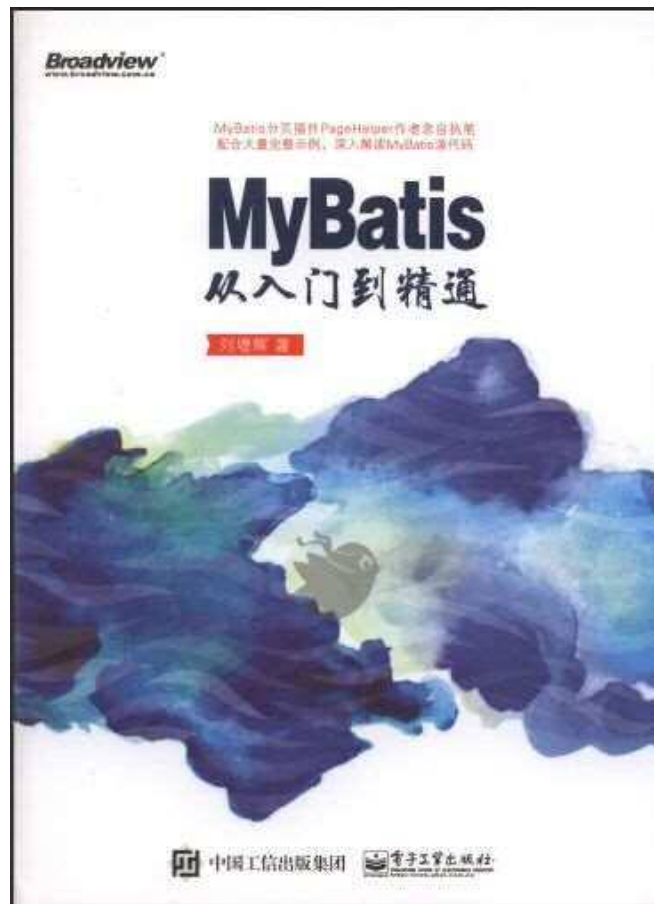
CRUD

删除

```
public static void main(String[] args) throws Exception{
    String resource = "conf.xml";
    Reader reader = Resources.getResourceAsReader(resource);
    SqlSessionFactory sessionFactory = new
    SqlSessionFactoryBuilder().build(reader);
    SqlSession session = sessionFactory.openSession();
    String statement = "com.mybatis_test.userMapper.deleteUser";
    int delete = session.delete(statement, "00001");
    session.commit();
    System.out.println(delete);
    session.close();
}
```

参考资料

MyBatis从入门到精通



参考资料

Neusoft

Neusoft

Beyond Technology