第一章习题答案

1.填空题

（1）ORM

（2）Hibernate、MyBatis

（3）Apache

（4）API接口层、数据处理层、基础支撑层

（5）SqlSessionFactory

2.选择题

（1）B

（2）B

（3）D

（4）C

（5）D

3.思考题

（1）

Ⅰ.代码繁琐

使用JDBC编程时，代码量较大，尤其是当数据表字段较多时，代码显得繁琐、累赘并且使开发人员的工作量增加。

Ⅱ.表关系维护复杂

数据表之间存在各种关系，包括一对一、一对多、多对多、级联等。如果采用JDBC编程的方式维护数据表之间的关系，过程较为复杂并且容易出错。

Ⅲ.硬编码

当使用JDBC编程时，SQL语句都是硬编码到Java程序中，如果改变SQL语句，那么需要重新编译Java代码，不利于系统后期的维护。

Ⅳ.性能问题

在批量处理数据的时候，JDBC编程存在效率低下的问题，此时，程序将向数据库发送大批量的同类SQL语句请求，浪费数据库资源，影响运行效率

（2）

MyBatis的功能架构由三层组成，包括API接口层、数据处理层、基础支撑层。

API接口层：提供给外部使用的接口API，开发人员通过这些API来操纵数据库。API接口层接收到调用请求时，它会调用数据处理层来完成具体的数据处理。

数据处理层：负责具体的SQL查找、SQL解析、SQL执行和执行结果映射处理等。它主要的功能是根据调用的请求完成一次数据库操作。

基础支撑层：负责最基础的功能支撑，包括连接管理、事务管理、配置加载、缓存处理等，MyBatis将这些共用的功能抽取出来作为最基础的组件，为上层的数据处理层提供支持。

4.编程题

Ⅰ.建库表、实体类Student,与书中例子相同。

Ⅱ.建StudentDao接口

1. package com.qfedu.dao;
2. import java.util.List;
3. import com.qfedu.pojo.Student;
4. public interface StudentDao {
5. //查询所有学生
6. List<Student> SelectByIdAll();
7. }

Ⅲ.StudentMapper.xml

1. <?xml version="1.0" encoding="UTF-8" ?>
2. <!DOCTYPE mapper
3. PUBLIC "-//mybatis.org//DTD Mapper 3.0//EN" "http://mybatis.org/dtd/mybatis-3-mapper.dtd">
4. <mapper namespace="com.qfedu.dao.StudentDao">
5. <select id="SelectByIdAll" resultType="com.qfedu.pojo.Student">
6. select\* from student
7. </select>
8. </mapper>

Ⅳ. mybatis-config.xml

1. <?xml version="1.0" encoding="UTF-8" ?>
2. <!DOCTYPE configuration
3. PUBLIC "-//mybatis.org//DTD Config 3.0//EN" "http://mybatis.org/dtd/mybatis-3-config.dtd">
4. <configuration>
5. <!--配置环境 -->
6. <environments default="mysql">
7. <environment id="mysql">
8. <transactionManager type="JDBC"/>
9. <dataSource type="POOLED">
10. <property name="driver" value="com.mysql.jdbc.Driver"/>
11. <property name="url" value="jdbc:mysql://localhost:3306/chapter01"/>
12. <property name="username" value="root"/>
13. <property name="password" value="root"/>
14. </dataSource>
15. </environment>
16. </environments>
17. <!-- 配置映射文件的位置 -->
18. <mappers>
19. <mapper resource="com/qfedu/dao/impl/StudentMapper.xml"/>
20. </mappers>
21. </configuration>

Ⅴ.书写测试类TestSelectStudent

1. package com.qfedu.test;
2. import java.io.IOException;
3. import java.io.InputStream;
4. import java.util.List;
5. import org.apache.ibatis.io.Resources;
6. import org.apache.ibatis.session.SqlSession;
7. import org.apache.ibatis.session.SqlSessionFactory;
8. import org.apache.ibatis.session.SqlSessionFactoryBuilder;
9. import com.qfedu.dao.StudentDao;
10. import com.qfedu.pojo.Student;
11. public class TestSelectStudent {
12. public static void main(String[] args) {
13. //读取配置文件
14. String resource = "mybatis-config.xml";
15. try {
16. InputStream is = Resources.getResourceAsStream(resource);
17. //创建SQLSessionFactory对象
18. SqlSessionFactory factory = new SqlSessionFactoryBuilder().build(is);
19. SqlSession session = factory.openSession();
20. StudentDao sd = session.getMapper(StudentDao.class);
21. List<Student> stu = sd.SelectByIdAll();
22. for (Student student : stu) {
23. System.out.println(student);
24. }
25. } catch (IOException e) {
26. e.printStackTrace();
27. }
28. }
29. }