第九章习题答案

1.填空题

（1）JDBCTemplate

（2）queryForObject()、queryForList()

（3）update()、batchUpdate

（4）execute()

（5）JdbcDaoSupport

2.选择题

（1）B

（2）D

（3）C

（4）A

（5）B

3.思考题

（1）

答：Spring 的JDBC模块由4个包组成，它们分别是core（核心包）、object（对象包）、dataSource（数据源包）和support（支持包），这些包相互协作，共同支撑了Spring的JDBC功能。

（2）

答：Ⅰ.通过直接注入JdbcTemplate的方式

Ⅱ.通过继承JdbcDaoSupport类的方式

4.编程题

Ⅰ.创建Teacher.java实体类

1. package pojo;
2. public class Teacher {
3. private int tid;
4. private String tname;
5. private String age;
6. private String course;
7. public Teacher() {
8. super();
9. // TODO Auto-generated constructor stub
10. }
11. public Teacher(int tid, String tname, String age, String course) {
12. super();
13. this.tid = tid;
14. this.tname = tname;
15. this.age = age;
16. this.course = course;
17. }
18. public int getTid() {
19. return tid;
20. }
21. public void setTid(int tid) {
22. this.tid = tid;
23. }
24. public String getTname() {
25. return tname;
26. }
27. public void setTname(String tname) {
28. this.tname = tname;
29. }
30. public String getAge() {
31. return age;
32. }
33. public void setAge(String age) {
34. this.age = age;
35. }
36. public String getCourse() {
37. return course;
38. }
39. public void setCourse(String course) {
40. this.course = course;
41. }
42. @Override
43. public String toString() {
44. return "Teacher [tid=" + tid + ", tname=" + tname + ", age=" + age + ", course=" + course + "]";
45. }
46. }

Ⅱ.TeacherDao.java

1. package dao;
2. import java.util.List;
3. import org.apache.ibatis.annotations.Select;
4. import pojo.Teacher;
5. public interface TeacherDao {
6. //查询数据表teacher中的记录条数
7. @Select("select \* from teacher")
8. public List<Teacher> queryNumber();
9. }

Ⅲ.application.xml

1. <?xml version="1.0" encoding="UTF-8"?>
2. <beans xmlns="http://www.springframework.org/schema/beans"
3. xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
4. xmlns:context="http://www.springframework.org/schema/context"
5. xmlns:aop="http://www.springframework.org/schema/aop"
6. xmlns:tx="http://www.springframework.org/schema/tx"
7. xsi:schemaLocation="http://www.springframework.org/schema/beans
8. http://www.springframework.org/schema/beans/spring-beans.xsd
9. http://www.springframework.org/schema/context
10. http://www.springframework.org/schema/context/spring-context.xsd
11. http://www.springframework.org/schema/aop
12. http://www.springframework.org/schema/aop/spring-aop.xsd
13. http://www.springframework.org/schema/tx
14. http://www.springframework.org/schema/tx/spring-tx.xsd">
15. <context:component-scan base-package="com.qfedu.\*" />
16. <!-- 引入外部 properties 文件 -->
17. <context:property-placeholder location="classpath:jdbc.properties"/>
18. <!-- 注册数据源 -->
19. <bean name="dataSource" class="com.mchange.v2.c3p0.ComboPooledDataSource">
20. <property name="driverClass" value="${jdbc.driverClass}"/>
21. <property name="jdbcUrl" value="${jdbc.jdbcUrl}"/>
22. <property name="user" value="${jdbc.user}"/>
23. <property name="password" value="${jdbc.password}"/>
24. </bean>
25. <!--注册JdbcTemplate类 -->
26. <bean name="jdbcTemplate" class="org.springframework.jdbc.core.JdbcTemplate">
27. <property name="dataSource" ref="dataSource"/>
28. </bean>
29. </beans>

Ⅳ.test测试类

1. package test;
2. import org.springframework.context.ApplicationContext;
3. import org.springframework.context.support.ClassPathXmlApplicationContext;
4. import org.springframework.jdbc.core.JdbcTemplate;
5. public class TestJDBCTemplate {
6. public static void main(String[] args) throws Exception {
7. ApplicationContext context =new ClassPathXmlApplicationContext("application.xml");
8. JdbcTemplate jdbcTemplate= context.getBean("jdbcTemplate",JdbcTemplate.class);
9. //1. 向数据库增加表
10. String createSql = "CREATE TABLE `teacher` (\r\n" +
11. " `tid` int(11) NOT NULL AUTO\_INCREMENT,\r\n" +
12. " `tname` varchar(20) DEFAULT NULL,\r\n" +
13. " `age` varchar(20) DEFAULT NULL,\r\n" +
14. " `course` varchar(20) DEFAULT NULL,\r\n" +
15. " PRIMARY KEY (`tid`)\r\n" +
16. ") ENGINE=InnoDB AUTO\_INCREMENT=2 DEFAULT CHARSET=utf8;";
17. jdbcTemplate.execute(createSql);
18. System.out.println("teacher 表增加成功");
19. //2.向teacher增加一条数据
20. String insertSql = "INSERT INTO `teacher` VALUES (?, ?, ?, ?);";
21. int i = jdbcTemplate.update(insertSql, null, "ZhangSan", "32", "Java");
22. System.out.println("增加数据数量："+i);
23. //3.teacher 中的记录数
24. String sql = "select count(1) from teacher";
25. //调用JDBCTemplate类对象的select()方法
26. Integer integer = jdbcTemplate.queryForObject(sql, Integer.class);
27. System.out.println("teacher 表中的个数：" + integer);
28. }
29. }