

一，spring的jdbctemplate操作

1，spring框架是一个一站式的框架

(1) 针对javaee三层架构，每一层都有自己的解决技术。

(2) 在dao层，使用jdbctemplate。

2，spring为各种支持的持久化技术，都提供了简单的操作模板和回调，如hibernate和mybatis。

ORM持久化技术	模板类
JDBC	org.springframework.jdbc.core.JdbcTemplate
Hibernate5.0	org.springframework.orm.hibernate5.HibernateTemplate
IBatis(MyBatis)	org.springframework.orm.ibatis.SqlMapClientTemplate
JPA	org.springframework.orm.jpa.JpaTemplate

(1) jdbctemplate对jdbc进行封装

3，jdbctemplate使用和dbutils使用很相似，对数据库进行crud操作。

(1) 增加

①导入jdbctemplate的使用jar包。



spring-jdbc-5.0.0.R...E.jar
381.48KB



spring-tx-5.0.0.REL...E.jar
243.05KB

②创建一个对象，设置数据库信息。

③创建一个jdbc模板对象，设置数据源。

④调用jdbctemplate对象里面的方法实现操作。

注意：千万不要忘了导入数据库的jar包：



mysql-connector-ja...6.jar
2.18MB

⑤具体的代码

```
// 1，添加操作
```

```
@Test
```

```
public void add() {
```

```

//2,创建一个对象，设置数据库信息。
DriverManagerDataSource dataSource = new
DriverManagerDataSource();
dataSource.setDriverClassName("com.mysql.cj.jdbc.Driver");
dataSource.setUrl("jdbc:mysql://localhost:3306/hb1?
serverTimezone=UTC");
dataSource.setUsername("root");
dataSource.setPassword("AQL271422");
//3,创建一个jdbc模板对象，设置数据源。
JdbcTemplate jdbcTemplate = new JdbcTemplate(dataSource);
//4调用jdbcTemplate对象里面的方法实现操作。
//表结构如下：create table anqili(id int auto_increment,name varchar(30));
String sqlString = "insert into anqili values(?,?)";
Integer integer = jdbcTemplate.update(sqlString, "1611010201","an");
System.out.println(integer);
}

```

(2) 修改

①写以上的（①-④）四个步骤。

②具体代码

//2，修改操作

```

@Test
public void updates(){
//1,创建一个对象，设置登录数据库的相关信息
DriverManagerDataSource dataSource = new
DriverManagerDataSource();
dataSource.setDriverClassName("com.mysql.cj.jdbc.Driver");
dataSource.setUrl("jdbc:mysql://localhost:3306/hb1?
serverTimezone=UTC");
dataSource.setUsername("root");
dataSource.setPassword("AQL271422");
//2,创建一个jdbc模板对象，设置数据源
JdbcTemplate jdbcTemplate = new JdbcTemplate(dataSource);
//3,调用jdbcTemplate对象里面的方法实现操作
String string = "update anqili set id = 11 where name='an'";
Integer integer = jdbcTemplate.update(string);
System.out.println(integer);
}

```

(3) 删除

①写以上的（①-④）四个步骤。

②具体代码

//3，删除操作

```

@Test
public void deletes(){
//1,创建一个对象，设置登录数据库的相关信息
DriverManagerDataSource dataSource = new

```

```
DriverManagerDataSource();
    dataSource.setDriverClassName("com.mysql.cj.jdbc.Driver");
    dataSource.setUrl("jdbc:mysql://localhost:3306/hb1?
serverTimezone=UTC");
    dataSource.setUsername("root");
    dataSource.setPassword("AQL271422");
    //2),创建一个jdbc模板对象，设置数据源
    JdbcTemplate jdbcTemplate = new JdbcTemplate(dataSource);
    //3),调用jdbctemplate对象里面的方法实现操作
    String string = "delete from anqili where name='an'";
    Integer integer = jdbcTemplate.update(string);
    System.out.println(integer);
}
```