

## 一，spring配置c3p0连接池

第一步：导入jar包



mchange-common...5.jar  
609.41KB



c3p0-0.9.5.4.jar  
488.12KB

第二步：创建spring配置文件，在里面配置连接池

```
<?xml version="1.0" encoding="UTF-8"?>
<beans xmlns="http://www.springframework.org/schema/beans"
    xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
    xmlns:context="http://www.springframework.org/schema/context"
    xmlns:jdbc="http://www.springframework.org/schema/jdbc"
    xmlns:jee="http://www.springframework.org/schema/jee"
    xmlns:tx="http://www.springframework.org/schema/tx"
    xmlns:util="http://www.springframework.org/schema/util"
    xmlns:task="http://www.springframework.org/schema/task"
    xsi:schemaLocation="
        http://www.springframework.org/schema/beans
        http://www.springframework.org/schema/beans/spring-beans-4.0.xsd
        http://www.springframework.org/schema/context
        http://www.springframework.org/schema/context/spring-context-4.0.xsd
        http://www.springframework.org/schema/jdbc
        http://www.springframework.org/schema/jdbc/spring-jdbc-4.0.xsd
        http://www.springframework.org/schema/jee
        http://www.springframework.org/schema/jee/spring-jee-4.0.xsd
        http://www.springframework.org/schema/tx
        http://www.springframework.org/schema/tx/spring-tx-4.0.xsd
        http://www.springframework.org/schema/util
        http://www.springframework.org/schema/util/spring-util-4.0.xsd
        http://www.springframework.org/schema/task
        http://www.springframework.org/schema/task/spring-task-4.0.xsd"
    default-lazy-init="false">
    <!-- 配置c3p0的连接池 -->
    <bean id="dataSource"
class="com.mchange.v2.c3p0.ComboPooledDataSource">
    <!-- 注入属性值 -->
    <property name="driverClass" value="com.mysql.cj.jdbc.Driver"></property>
    <property name="jdbcUrl" value="jdbc:mysql://localhost:3306/hb1?
serverTimezone=UTC"></property>
    <property name="user" value="root"></property>
```

```

        <property name="password" value="AQL271422"> </property>
    </bean>
</beans>

```

## 二，dao使用jdbcTemplate

1，创建service和dao实体对象，配置service和dao对象，在service中注入dao对象。

```

<!-- 创建 service和dao对象，在service中注入dao的对象-->
    <bean id="userDao" class="cn.java.c3p0.UserDao"> </bean>
    <bean id="userService " class="cn.java.c3p0.UserService">
        <!-- 注入dao对象 -->
        <property name="userDao" ref="userDao"> </property>
    </bean>

```

2，创建jdbcTemplate对象，把模板对象注入到dao中去

```

<!-- 创建jdbcTemplate对象 -->
    <bean id="jdbcTemplate"
class="org.springframework.jdbc.core.JdbcTemplate">
        <!-- 把dataSource传递到模板对象里面 -->
        <property name="dataSource" ref="dataSource"> </property>
    </bean>

```

3，template对象

```

package cn.java.c3p0;
import org.springframework.jdbc.core.JdbcTemplate;
public class UserDao {
    //得到JdbcTemplate对象
    private JdbcTemplate jdbcTemplate;
    public JdbcTemplate getJdbcTemplate() {
        return jdbcTemplate;
    }
    public void setJdbcTemplate(JdbcTemplate jdbcTemplate) {
        this.jdbcTemplate = jdbcTemplate;
    }
    //添加操作
    public void add(){

    }
}

```

## 三，具体的代码：

userdao对象：

```

package cn.java.c3p0;
import org.springframework.jdbc.core.JdbcTemplate;
public class UserDao {
    //得到JdbcTemplate对象
    private JdbcTemplate jdbcTemplate;

```

```

public JdbcTemplate getJdbcTemplate() {
    return jdbcTemplate;
}
public void setJdbcTemplate(JdbcTemplate jdbcTemplate) {
    this.jdbcTemplate = jdbcTemplate;
}
//添加操作
public void add(){
    String sqlString = "insert into anqili values(?,?)";
    jdbcTemplate.update(sqlString,"88","66");
}
}

```

userservice对象:

```

package cn.java.c3p0;
public class UserService {
private UserDao userDao;
public void setUserDao(UserDao userDao) {
    this.userDao = userDao;
}
public void add(){
    userDao.add();
}
}

```

applicationcontext配置文件详细信息:

```

<?xml version="1.0" encoding="UTF-8"?>
<beans xmlns="http://www.springframework.org/schema/beans"
    xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
    xmlns:context="http://www.springframework.org/schema/context"
    xmlns:jdbc="http://www.springframework.org/schema/jdbc"
    xmlns:jee="http://www.springframework.org/schema/jee"
    xmlns:tx="http://www.springframework.org/schema/tx"
    xmlns:util="http://www.springframework.org/schema/util"
    xsi:schemaLocation="
        http://www.springframework.org/schema/beans
        http://www.springframework.org/schema/beans/spring-beans-4.0.xsd
        http://www.springframework.org/schema/context
        http://www.springframework.org/schema/context/spring-context-4.0.xsd
        http://www.springframework.org/schema/jdbc
        http://www.springframework.org/schema/jdbc/spring-jdbc-4.0.xsd
        http://www.springframework.org/schema/jee
        http://www.springframework.org/schema/jee/spring-jee-4.0.xsd
        http://www.springframework.org/schema/tx
        http://www.springframework.org/schema/tx/spring-tx-4.0.xsd
        http://www.springframework.org/schema/util
        http://www.springframework.org/schema/util/spring-util-4.0.xsd

```

```

http://www.springframework.org/schema/task
http://www.springframework.org/schema/task/spring-task-4.0.xsd
    default-lazy-init="false">
    <!-- 配置c3p0的连接池 -->
    <bean id="dataSource"
class="com.mchange.v2.c3p0.ComboPooledDataSource">
        <!-- 注入属性值 -->
        <property name="driverClass" value="com.mysql.cj.jdbc.Driver">
</property>
        <property name="jdbcUrl"
            value="jdbc:mysql://localhost:3306/hb1?serverTimezone=UTC">
</property>
        <property name="user" value="root"></property>
        <property name="password" value="AQL271422"></property>
    </bean>
    <!-- 创建 service和dao对象，在service中注入dao的对象-->
    <bean id="userDao" class="cn.java.c3p0.UserDao">
    <!-- 注入template对象 -->
    <property name="jdbcTemplate" ref="jdbcTemplate"></property>
    </bean>
    <bean id="userService" class="cn.java.c3p0.UserService">
        <!-- 注入dao对象 -->
        <property name="userDao" ref="userDao"></property>
    </bean>
    <!-- 创建jdbcTemplate对象 -->
    <bean id="jdbcTemplate"
class="org.springframework.jdbc.core.JdbcTemplate">
        <!-- 把dataSource传递到模板对象里面 -->
        <property name="dataSource" ref="dataSource"></property>
    </bean>
</beans>

```

测试代码:

```

package cn.java.testSet;
import org.junit.Test;
import org.springframework.context.ApplicationContext;
import org.springframework.context.support.ClassPathXmlApplicationContext;
import cn.java.c3p0.UserService;
import cn.java.jdbcTemplate.user;
import cn.java.zhujie.aop.Book;
public class TestTemplate {
    @Test
    public void testTemplate(){
        ApplicationContext context =
            new ClassPathXmlApplicationContext("applicationcontext.xml");
        UserService book = (UserService)context.getBean("userService");
        book.add();
    }
}

```

} }