

1、jar包：mybatis的和Spring的一共26个

[commons-logging-1.1.1.jar](#)

[javassist-3.17.1-GA.jar](#)

[log4j-1.2.17.jar](#)

[log4j-api-2.0-rc1.jar](#)

[log4j-core-2.0-rc1.jar](#)

[mybatis-3.2.7.jar](#)

[mybatis-spring-1.2.2.jar](#)

[mysql-connector-java-5.1.7-bin.jar](#)

[slf4j-api-1.7.5.jar](#)

[slf4j-log4j12-1.7.5.jar](#)

[spring-aop-4.2.4.RELEASE.jar](#)

[spring-aspects-4.2.4.RELEASE.jar](#)

[spring-beans-4.2.4.RELEASE.jar](#)

[spring-context-4.2.4.RELEASE.jar](#)

[spring-core-4.2.4.RELEASE.jar](#)

[spring-expression-4.2.4.RELEASE.jar](#)

[spring-jdbc-4.2.4.RELEASE.jar](#)

[spring-orm-4.2.4.RELEASE.jar](#)

[spring-tx-4.2.4.RELEASE.jar](#)

[spring-web-4.2.4.RELEASE.jar](#)

[asm-3.3.1.jar](#)

[cglib-2.2.2.jar](#)

[com.springsource.com.mchange.v2.c3p0-0.9.1.2.jar](#)

[com.springsource.org.aopalliance-1.0.0.jar](#)

[com.springsource.org.apache.commons.logging-1.1.1.jar](#)

[com.springsource.org.aspectj.weaver-1.6.8.RELEASE.jar](#)

## 2、创建Spring的配置文件

### 1、导入Spring约束

```
<beans xmlns="http://www.springframework.org/schema/beans"
        xmlns:context="http://www.springframework.org/schema/context"
        xmlns:p="http://www.springframework.org/schema/p"
        xmlns:aop="http://www.springframework.org/schema/aop"
        xmlns:tx="http://www.springframework.org/schema/tx"
        xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
        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/aop
http://www.springframework.org/schema/aop/spring-aop-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">
```

### 2、扫描相应的包

```
<context:property-placeholder location="classpath:jdbc.properties"/>    扫描数据源的包
```

### 3、管理数据源 使用c3p0

```
<bean
    id="dataSource" class="com.mchange.v2.c3p0.ComboPooledDataSource">
    <property name="jdbcUrl" value="${jdbc.url}"></property>
    <property name="driverClass" value="${jdbc.driver}"></property>
    <property name="user" value="${jdbc.username}"></property>
    <property name="password" value="${jdbc.password}"></property>
</bean>
```

#### 4、sqlSessionFactory工厂管理

```
<bean id="sqlSessionFactory" class="org.mybatis.spring.SqlSessionFactoryBean">
    <property name="dataSource" ref="dataSource"></property>

    <!-- 加载Mybatis核心配置文件 -->

    <property name="configLocation" value="classpath:sqlMapConfig.xml"></property>

    <!-- 定义别名 -->

    <property name="typeAliasesPackage" value="com.itheima.domain"></property>
</bean>

5、把工厂注入dao

<bean id="userDao" class="com.itheima.dao.impl.UserMapperImpl">
    <property name="sqlSessionFactory" ref="sqlSessionFactory"></property>
</bean>
```

#### 3、创建DAO接口，创建DAO实现类继承 extends SqlSessionDaoSupport

#### 4、实现接口方法

#### 5、写测试类调用DAO

//加载Spring配置文件

```
ApplicationContext app = new
ClassPathXmlApplicationContext("applicationContext.xml");
```

//获取dao对象

```
UserMapper userMapper = (UserMapper) app.getBean("userDao");
```

//调用方法

```
User user = userMapper.findUserByID(1);
```