Spring整合MyBatis 步骤：

1. 添加mybatis-spring整合jar包

|  |
| --- |
| <dependency>  <groupId>org.mybatis</groupId>  <artifactId>mybatis-spring</artifactId>  <version>1.3.3</version> </dependency> |

1. 在spring配置文件中将第下jar包中的 SqlSessionFactoryBean类纳入spring容器管理 ，此类负责读取mybatis的全局配置文件 ，以及mybatis的映射文件,负责连接数据库

|  |
| --- |
| <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:aop="http://www.springframework.org/schema/aop"  xmlns:mybatis-spring=<http://mybatis.org/schema/mybatis-spring>   xsi:schemaLocation="http://www.springframework.org/schema/beans  http://www.springframework.org/schema/beans/spring-beans.xsd  http://www.springframework.org/schema/context  http://www.springframework.org/schema/context/spring-context.xsd   http://www.springframework.org/schema/aop  http://www.springframework.org/schema/aop/spring-aop.xsd  http://mybatis.org/schema/mybatis-spring  http://mybatis.org/schema/mybatis-spring.xsd">  负责连接数据库  <bean id="datasource" class="org.apache.commons.dbcp2.BasicDataSource" destroy-method="closeConnectionPool">  <property name="driverClassName" value="${driver}"></property>  <property name="password" value="${password}"></property>  <property name="username" value="${username}"></property>  <property name="url" value="${url}"></property> </bean> *<!--整合MyBatis框架  org.mybatis.spring.SqlSessionFactoryBean是spring+mybatis整合包中提供的工具类，用于获得SqlSession对象的  -->* <bean id="sqlSessionFactory" class="org.mybatis.spring.SqlSessionFactoryBean">  <!—注入dataSource到SqlSessionFactoryBean --><property name="dataSource" ref="datasource"></property>  <!—name="mapperLocations" 扫描类路径下com目录下的mapper目录下的所有xml文件 做为Mybatis的映射文件 -->  <property name="mapperLocations" value="classpath:com/mapper/\*.xml"></property>  <!--- name="configLocation" 负责将读取mybatis的全局配置文件 把mybatis的基本配置(如打印sql日志及分页插件加入到mybatis中) -->  <property name="configLocation" value="classpath:mybatis-config.xml"></property> </bean>  *<!--将mybatis的com.mapper包也的所有的接口实现类纳入spring容器管理  在代码中可以用@Autowired注解注入依赖的Mapper -->* <mybatis-spring:scan base-package="com.mapper"></mybatis-spring:scan> |

1. 在程序代码中用@Autowired注解注入依赖的Mapper,调用相应的方法完成数据库的相关操作

|  |
| --- |
| ClassPathXmlApplicationContext context = new ClassPathXmlApplicationContext("application.xml");   *//用Spring整合Mybatis后， 发现我们不需要通过SqlSessionFactory.openSession()打开连接，也不需通过sqlSession.getMapper()得到Mapper接口了 因为我们在配置文件 中配置的 SqlSessionFactoryBean,以及 通过 mybatis-spring扫描 com.mapper包下的接口  com.mapper包下的所有Mppaer接口自动纳入Spring容器管理 了，数据库的连接与关闭都不需要我们手动管理 了，由spring帮我们管理了数据库的开启与关闭* DeptServiceImpl deptService = context.getBean("deptServiceImpl", DeptServiceImpl.class);  List<DeptBean> deptBeanList = deptService.queryData(null);  System.*out*.println(deptBeanList); |