**Spring集成mybatis的好处**

集成之后，可以使用spring来管理mybatis（扫描配置文件，事务，日志...），提供扩展功能类（SqlSessionTemplate，SqlSessionDaoSupport）

**SqlSessionTemplate（线程安全的SqlSession）**

SqlSessionTemplate 是 MyBatis-Spring 的核心。 这个类负责管理 MyBatis 的 SqlSession, 调用 MyBatis 的 SQL 方法, 翻译异常。 SqlSessionTemplate 是线程安全的, 可以被多个 DAO 所共享使用。

当调用 SQL 方法时, 包含从映射器 getMapper()方法返回的方法, SqlSessionTemplate 将会保证使用的 SqlSession 是和当前 Spring 的事务相关的。此外,它管理 session 的生命 周期,包含必要的关闭,提交或回滚操作。 SqlSessionTemplate 通常是被用来替代默认的 MyBatis 实现的 DefaultSqlSession , 因为模板可以参与到 Spring 的事务中并且被多个注入的映射器类所使 用时也是线程安全的。

**SqlSessionDaoSupport（方便获取SqlSession）**

SqlSessionDaoSupport 是 一 个 抽象 的支 持 类, 用来 为你 提供 SqlSession 。 调 用 getSqlSession()方法你会得到一个 SqlSessionTemplate,之后可以用于执行 SQL 方法

**Spring集成mybatis的配置**

不同的版本可能不兼容通过maven查看兼容版本

兼容的版本-》spring 3.2.8,mybatis-spring 1.2.2,mybatis3.2.3

**a）下载jar包（maven下载）**

**1）spring自己的jar**

**2）spring集成mybatis的jar**-》mybatis-spring

**3）mybatis的jar**

**b）spring的配置文件**

**1）dataSource-》数据源**

<bean id="myDataSource" class="org.apache.commons.dbcp.BasicDataSource"

destroy-method="close">

<property name="driverClassName" value="${driver}" />

<property name="url" value="${url}" />

<property name="username" value="${uname}" />

<property name="password" value="${pwd}" />

</bean>

**2）SqlSessionFactory-》session工厂**

<bean id="sqlSessionFactory" class="org.mybatis.spring.SqlSessionFactoryBean">

<property name="dataSource" ref="myDataSource" />

配置javabean所在的包

<!-- 配置参数和结果类型bean所在位置，自动设置别名 -->

<property name="typeAliasesPackage" value="org.mybatis.jpetstore.domain" />

<!-- 配置mybatis入口文件 -->

<property name="configLocation" value="classpath:conf/mybatis-config.xml" />

<!-- 自动扫描mybatis的mapping.xml文件 -->

<property name="mapperLocations" value="classpath:com/cyber/vip/dao/\*\*/\*.xml" />

</bean>

<!-- 扫描mapper接口 -->

<bean class="org.mybatis.spring.mapper.MapperScannerConfigurer">

<property name="basePackage" value="com.cdsxt.mybatis.dao" />

</bean>

<bean class="org.mybatis.spring.mapper.MapperScannerConfigurer">

配置dao接口层

        <property name="basePackage" value="org.mybatis.jpetstore.persistence" />

    </bean>

**3）事务处理（和spring的jdbc一致，hibernate需要配置另外的管理器）**

<!-- 定义事务管理器 -->

<bean id="transactionManager"

class="org.springframework.jdbc.datasource.DataSourceTransactionManager">

<property name="dataSource" ref="dataSource" />

</bean>

<!--使用注释事务 -->

<tx:annotation-driven transaction-manager="transactionManager" />

**4）【可选】（如果需要线程安全的session）**

<bean id="sqlSession" class="org.mybatis.spring.SqlSessionTemplate">

<constructor-arg index="0" ref="sqlSessionFactory" />

</bean>

---------------问题----------------------------------

a）集成spring配置事务之后，session默认都是自动提交（因为事务交给了spring管理），可以通过sesssion.getConnection().setAutocommit(false)来设置，手动提交sesssion.getConnection().commit() 提交数据