https://blog.csdn.net/gongsunjinqian/article/details/52710557

**spring-boot与mybatis整合优化介绍**

2016年09月30日 13:10:40 [ShawnRacine](https://me.csdn.net/gongsunjinqian) 阅读数：15816更多

个人分类： [JAVA](https://blog.csdn.net/gongsunjinqian/article/category/6078422)

版权声明：本文为博主原创文章，未经博主允许不得转载。 https://blog.csdn.net/gongsunjinqian/article/details/52710557

**SpringBoot官方文档**[**http://docs.spring.io/spring-boot/docs/current/reference/htmlsingle/**](http://docs.spring.io/spring-boot/docs/current/reference/htmlsingle/)

关于spring-boot与mybatis整合优化方面的介绍，就是Mybatis-Spring-boot-starter的介绍：

**1、取消spring-mybatis.xml配置**

**①自动检测已存在的Datasource**

之前，需要在spring-mybatis.xml中配置datasource的Bean，现在只需要在application.yml中配置到spring.datasource节点下就可以。因为mybatis-spring-boot支持自动检测已存在的Datasource。

**②将创建并注册SqlSessionFactoryBean实例，并传入Datasource。**

在mybatis中，sqlsession可以有SqlSessionFactory创建；而在mybatis-spring中则需要SqlSessionFactoryBean来创建，并传入datasource。

如：

1. <bean id="sqlSessionFactory" class="org.mybatis.spring.SqlSessionFactoryBean">
2. <property name="configLocation">
3. <value>classpath:mybatis/mapper.xml</value>
4. </property>
5. <property name="dataSource" ref="dataSource" />
6. </bean>

现在，mybatis-spring-boot支持自动创建并注册SqlSessionFactoryBean，所以以上的配置都不需要了。

**③将从SqlSessionFactoryBean中创建并注册SqlSessionTemplate**

SqlSessionTemplate是SqlSession的实现类，较SqlSession的默认实现类DefaultSqlSession来说，是线程安全的。

在mybatis-spring中需要如下配置：

1. <bean id="sqlSession" class="org.mybatis.spring.SqlSessionTemplate">
2. <constructor-arg index="0" ref="sqlSessionFactory" />
3. </bean>

现在，mybatis-spring-boot支持自动创建并注册SqlSessionTemplate，所以不需要以上配置了。

SqlSession对象注入，如下：

1. @Autowired
2. private SqlSession sqlSession;

：：真不知道既然下面④都能注入mappers了，那还要SqlSession对象有什么用。。：：

**④自动扫描mappers，将其关联到SqlSessionTemplate，并将mappers注册到spring容器中，以便注入到我们的beans中。**

默认情况下，mybatis-spring-boot将搜索被@Mapper注释标注的mappers。

文档中描述可以用mybatis-spring提供的@MapperScan标注，但我不会用。

Mybatis-Spring文档中解释@MapperScan注释跟配置MapperScannerConfigurer是同样的效果：

public @interface MapperScan Use this annotation to register MyBatis mapper interfaces when using Java Config. It performs when same work as MapperScannerConfigurer via MapperScannerRegistrar.

1. <bean class="org.mybatis.spring.mapper.MapperScannerConfigurer">
2. <property name="basePackage" value="com.example.mappers" />
3. <property name="sqlSessionFactoryBeanName" value="sqlSessionFactory"></property>
4. </bean>

现在，mybatis-spring-boot支持使用@Mapper注释标注mappers接口类了，所以就不需要上述配置。

：：其实感觉上述配置还是挺好的，不用每个mapper接口都注释@Mapper。。。：：

@Mapper标注使用如下：

1. @Mapper
2. public interface UserMapper {
3. UserInfo queryUser(@Param(value = "userId") int id);
4. }

那么在mybatis-spring-boot中需要配置的是mapper.xml目录：

1. mybatis:
2. mapper-locations: classpath:mapper/\*.xml