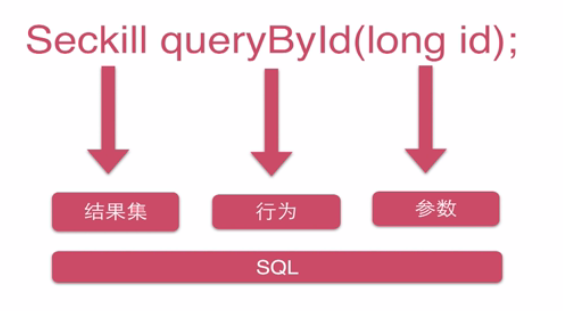
Mybatis整合Spring

整合目标:



1. 更少的编码

只写接口，不写实现



1. 更少的配置-别名



mybatis可以扫描包（就可以识别包下面所有的类）



更少的配置-配置扫描（mybatis提供自动扫描）



更少的配置-dao实现(mybatis提供自动实现DAO接口,整合之后自动注入spring容器)



1. 足够的灵活性



spring整合mybatis实现

1. 在resource目录下创建一个spring文件，在文件下面创建一个spring-dao.xml(所有

dao相关的配置)

其中将数据库连接有关的属性driver，放在了jdbc.properties中

|  |
| --- |
| <?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"*  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"*>  <!-- 配置整合mybatis过程 -->  <!-- 1:配置数据库相关参数  properties的属性:${url}  -->  <context:property-placeholder location=*"classpath:jdbc.properties"*/>  <!-- 2:数据库连接池 -->  <bean id=*"dataSource"* class=*"com.mchange.v2.c3p0.ComboPooledDataSource"*>  <!-- 配置连接池属性 -->  <property name=*"driverClass"* value=*"${driver}"*></property>  <property name=*"jdbcUrl"* value=*"${url}"*></property>  <property name=*"user"* value=*"${jdbc.username}"*></property>  <property name=*"password"* value=*"${password}"*></property>    <!-- 配置连接池的私有属性 -->  <property name=*"maxPoolSize"* value=*"30"*></property>  <property name=*"minPoolSize"* value=*"10"*></property>  <!-- 关闭连接后不自动commit -->  <property name=*"autoCommitOnClose"* value=*"false"*></property>  <!-- 获取连接超时时间 -->  <property name=*"checkoutTimeout"* value=*"1000"*></property>  <!-- 获取当前连接失败重试次数 -->  <property name=*"acquireRetryAttempts"* value=*"2"*></property>  </bean>    <!-- 3:配置sqlsessionfactory对象-->  <bean id=*"sqlSessionFactory"* class=*"org.mybatis.spring.SqlSessionFactoryBean"*>  <!-- 注入数据库连接池 -->  <property name=*"dataSource"* ref=*"dataSource"*></property>  <!-- 配置mybatis全局配置文件 -->  <property name=*"configLocation"* value=*"classpath:mybatis-config.xml"*></property>  <!-- 扫描entity包，使用别名 -->  <property name=*"typeAliasesPackage"* value=*"edu.hubu.seckill.entity"*></property>  <!-- 扫描sql配置文件：mapper需要的xml文件 -->  <property name=*"mapperLocations"* value=*"classpath:mapper/\*.xml"*></property>  </bean>    <!-- 4:配置扫描dao接口包，动态实现Dao接口，并注入到spring容器中 -->  <bean class=*"org.mybatis.spring.mapper.MapperScannerConfigurer"*>  <!-- 注入sqlSessionFactory -->  <property name=*"sqlSessionFactoryBeanName"* value=*"sqlSessionFactory"*></property>  <!-- 给出扫描DAO接口包 -->  <property name=*"basePackage"* value=*"edu.hubu.seckill.dao"*></property>  </bean>  </beans> |