



黑马程序员

《Java EE 企业级应用开发教程 （SSM）（第2版）》

教学设计

课程名称： Java EE 企业级应用开发教程

授课年级： XXXX 年级

授课学期： 第 X 学期

教师姓名： 某某老师

2021年6月



课题名称	第2章 MyBatis 的核心配置	计划课时	6 课时
教学引入	通过上一章的学习，我们已经对 MyBatis 框架有了一个初步的了解，但是要想熟练地使用 MyBatis 框架进行实际开发，只会简单的配置是不行的，还需要对框架中的核心对象、核心配置文件以及映射文件有更深入的了解。本章将针对 MyBatis 核心对象、核心配置文件和映射文件进行详细讲解。		
教学目标	<ul style="list-style-type: none"> ● 使学生了解 MyBatis 核心对象的作用 ● 使学生掌握 MyBatis 核心配置文件及其元素的使用 ● 使学生掌握 MyBatis 映射文件及其元素的使用 		
教学重点	<ul style="list-style-type: none"> ● SqlSessionFactory 对象 ● <mappers>元素 ● MyBatis 映射文件中的常用元素 ● <select>元素 ● <insert>元素 ● <update>元素 ● <delete>元素 ● <resultMap>元素 		
教学难点	无		
教学方式	课堂教学以 PPT 讲授为主，并结合多媒体进行教学		
教学过程	<p style="text-align: center;">第一课时</p> <p style="text-align: center;">(SqlSessionFactoryBuilder 对象、SqlSessionFactory 对象、SqlSessionFactory 对象、MyBatis 配置文件的主要元素、<properties>元素)</p> <p>一、创设情景，导入新课</p> <p>构建楼房的基础是打好地基，有了坚固的地基才能够进行下一步的建设。同样在实际开发中，也需要一个“地基”，为项目的开发提供后台数据支持。MyBatis 是一个持久层框架，它便充当了一个“地基”的角色。本节课，将对 SqlSessionFactoryBuilder 对象、SqlSessionFactory 对象、SqlSessionFactory 对象、<properties>元素等进行详细讲解。</p> <p>二、新课讲解</p> <p>知识点 1-SqlSessionFactoryBuilder 对象</p> <p>教师通过多媒体演示 PPT 内容讲解 SqlSessionFactoryBuilder 对象。</p> <p>所有的 MyBatis 应用都是以 SqlSessionFactory 对象为中心，而 SqlSessionFactoryBuilder 就是 SqlSessionFactory 的构造者。</p> <p>SqlSessionFactoryBuilder 通过 build() 方法构建 SqlSessionFactory 对象，SqlSessionFactoryBuilder 提供了多个重载的 build() 方法。</p> <p>SqlSessionFactoryBuilder 构建 SqlSessionFactory 对象的 build() 方法按照配置信息的传入方式，可以分为三种形式，具体如下：</p> <p>(1) build(InputStream inputStream, String environment, Properties properties)</p>		



- (2) build(Reader reader, String environment, Properties properties)
- (3) build(Configuration config)

知识点 2-SqlSessionFactory 对象

教师讲解 SqlSessionFactory 对象。

SqlSessionFactory 是 MyBatis 框架中十分重要的对象，用于创建 SqlSession 对象，所有的 MyBatis 应用都是以 SqlSessionFactory 为对象中心。在 SqlSessionFactoryBuilder 构建 SqlSessionFactory 对象之后，就可以使用 SqlSessionFactory 对象调用 SqlSessionFactory 的 openSession() 方法创建 SqlSession 对象，SqlSessionFactory 有多个重载的 openSession() 方法。

接下来讲 SqlSessionFactory 的 openSession() 方法。

知识点 3-SqlSession 对象

教师讲解 SqlSession 对象。

SqlSession 是 MyBatis 框架中另一个重要的对象，它是应用程序与持久层之间执行交互操作的一个单线程对象，主要作用是执行持久化操作，类似于 JDBC 中的 Connection。SqlSession 对象包含了执行 SQL 操作的方法，由于其底层封装了 JDBC 连接，所以可以直接使用 SqlSession 对象来执行已映射的 SQL 语句。

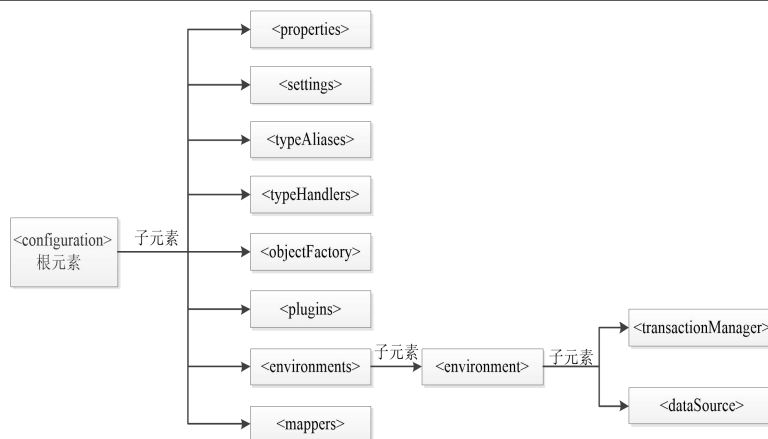
SqlSession 对象中包含了很多方法，常用方法有<T> T selectOne(String statement)、int insert(String statement, Object parameter)、int update(String statement)、int delete(String statement)、void commit()、void rollback() 等，分别表示查询方法、插入方法、更新方法、删除方法、提交事务和回滚事务的方法。

每一个线程都应该有一个自己的 SqlSession 对象，并且该对象不能共享。SqlSession 对象是线程不安全的，因此其使用范围最好在一次请求或一个方法中，绝不能将其放在类的静态字段、对象字段或任何类型的管理范围（如 Servlet 的 HttpSession）中使用。SqlSession 对象使用完之后，要及时的关闭，SqlSession 对象通常放在 finally 块中关闭。

知识点 4-MyBatis 配置文件的主要元素

教师讲解 MyBatis 配置文件的主要元素。

MyBatis 核心配置文件中的主要元素如图所示。



知识点 5-`<properties>`元素

教师讲解`<properties>`元素。

`<properties>`是一个配置属性的元素，该元素的作用是读取外部文件的配置信息。例如，数据库连接参数等属性，原本是配置在核心配置文件内部，通过`<properties>`元素，可以让系统读取外部文件的属性值。

`<properties>`元素具体使用方式如下。

- (1) 在项目的 `src/main/resources` 目录下，创建一个名称为 `db.properties` 的配置文件，该文件主要用于配置数据库的连接信息。
- (2) 在 MyBatis 核心配置文件 `mybatis-config.xml` 中使用 `<properties... />` 元素引入 `db.properties` 文件，以获取数据库连接信息。
- (3) 在 MyBatis 核心配置文件 `mybatis-config.xml` 中使用 `<property... />` 元素配置数据库驱动、连接数据库的 URL、连接数据库的用户名和连接数据库的密码等数据库连接参数。

三、归纳总结

教师回顾本节课所讲的内容，并通过测试题的方式引导学生解答问题并给予指导。

四、布置作业

教师通过高校教辅平台 (<http://tch.ityxb.com>) 布置本节课作业以及下节课的预习作业。

第二课时

(`<settings>`元素、`<typeAliases>`元素、`<environments>`元素、`<mappers>`元素)

一、复习巩固

教师通过上节课作业的完成情况，对学生吸收不好的知识点进行再次巩固讲解。

二、通过直接导入的方式导入新课

掌握了 MyBatis 框架的三个核心对象以及配置文件包含的主要元素等，接下来就需要学习配置文件中具体的元素，如`<settings>`元素、`<typeAliases>`元素以及`<environments>`元素等。

三、新课讲解

知识点 1-`<settings>`元素



教师讲解<settings>元素。

<settings>元素主要用于改变 MyBatis 运行时的行为，如开启二级缓存、开启延迟加载等。

接下来讲<settings>元素中的常见配置参数。

知识点 2-<typeAliases>元素

教师讲解<typeAliases>元素。

核心配置文件若要引用一个 POJO 实体类，需要输入 POJO 实体类的全限定类名，而 POJO 实体类的全限定类名比较冗长，如果直接输入 POJO 实体类的全限定类名，很容易拼写错误。这时，可以使用<typeAliases>元素为核心配置文件中的 POJO 实体类设置一个简短的别名，通过 MyBatis 的核心配置文件与映射文件相关联，减少全限定类名的冗余，以简化操作。

如果有多个全限定类需要设置别名，有以下两种方式可以完成设置。

(1) 在<typeAliases>元素下，使用多个<typeAlias>元素为每一个全限定类逐个配置别名。

(2) 通过自动扫描包的形式自定义别名。

除了可以使用<typeAliases>元素为实体类自定义别名外，MyBatis 框架还为许多常见的 Java 类型（如数值、字符串、日期和集合等）提供了相应的默认别名，如 byte 对应的别名为_byte、long 对应的别名为_long、Byte 对应的别名为 byte 等。

知识点 3-<environments>元素

教师讲解<environments>元素。

MyBatis 的运行环境信息包括事务管理器和数据源。在 MyBatis 的核心配置文件中，MyBatis 通过<environments>元素定义一个运行环境。

<environments>元素有两个子元素，<transactionManager>元素和<dataSource>元素。<transactionManager>元素用于配置运行环境的事务管理器；<dataSource>元素用于配置运行环境的数据源信息。

在 MyBatis 中，<transactionManager>元素可以配置两种类型的事务管理器，分别是 JDBC 和 MANAGED。

对于数据源的配置，MyBatis 框架提供了 UNPOOLED、POOLED 和 JNDI 三种数据源类型，具体如下。

1. UNPOOLED

UNPOOLED 表示数据源为无连接池类型。配置此数据源类型后，程序在每次被请求时会打开和关闭数据库连接。UNPOOLED 适用于对性能要求不高的简单应用程序。

2. POOLED

POOLED 表示数据源为连接池类型。POOLED 数据源利用“池”的概念将 JDBC 连接对象组织起来，节省了在创建新的连接对象时需要初始化和认证的时间。

3. JNDI

JNDI 表示数据源可以在 EJB 或应用服务器等容器中使用。

知识点 4-<mappers>元素

教师讲解<mappers>元素。



在 MyBatis 的核心配置文件中，<mappers>元素用于引入 MyBatis 映射文件。映射文件包含了 POJO 对象和数据表之间的映射信息，MyBatis 通过核心配置文件中的<mappers>元素找到映射文件并解析其中的映射信息。

通过<mappers>元素引入映射文件的方法有 4 种，具体如下。

1. 使用类路径引入
2. 使用本地文件路径引入
3. 使用接口类引入
4. 使用包名引入

四、归纳总结

教师回顾本节课所讲的内容，并通过测试题的方式引导学生解答问题并给予指导。

五、布置作业

教师通过高校教辅平台（<http://tch.ityxb.com>）布置本节课作业以及下节课的预习作业。

第三课时

（MyBatis 映射文件中的常用元素、

<select>元素、<insert>元素、<update>元素、<delete>元素、<sql>元素）

一、复习巩固

教师通过上节课作业的完成情况，对学生吸收不好的知识点进行再次巩固讲解。

二、通过直接导入的方式导入新课

掌握了 MyBatis 配置文件包含的主要元素等，接下学习映射文件中具体的元素，如<select>元素、<insert>元素以及<sql>元素等。

三、新课讲解

知识点 1-MyBatis 映射文件中的常用元素

教师讲解 MyBatis 映射文件中的常用元素。

在 MyBatis 映射文件中，<mapper>元素是映射文件的根元素，其他元素都是它的子元素。

接下来讲 MyBatis 映射文件中的常用元素。

知识点 2-<select>元素

教师讲解<select>元素。

<select>元素用来映射查询语句，它可以从数据库中查询数据并返回。

接下来讲<select>元素的常用属性。

知识点 3-<insert>元素

教师讲解<insert>元素。

<insert>元素用于映射插入语句，在执行完<insert>元素中定义的 SQL 语句后，会返回插入记录的数量。

很多时候，执行插入操作后，需要获取插入成功的数据生成的主键值，不同类型数据库获取主键值的方式不同，下面分别对支持主键自动增长的数据库获取主键值和不支持主键自动增长的数据库获取主键值的方式进行介绍。

1. 使用支持主键自动增长的数据库获取主键值

如果使用的数据库支持主键自动增长（如 MySQL 和 SQL Server），那么可



以通过 keyProperty 属性指定 POJO 类的某个属性接收主键返回值(通常会设置到 id 属性上)，然后将 useGeneratedKeys 的属性值设置为 true。

2. 使用不支持主键自动增长的数据库获取主键值

如果使用的数据库不支持主键自动增长(如 Oracle)，或者支持增长的数据库取消了主键自增的规则，可以使用 MyBatis 提供的<selectKey>元素来自定义主键。

知识点 4-<update>元素

教师讲解<update>元素。

<update>元素用于映射更新语句，它可以更新数据库中的数据。在执行完元素中定义的 SQL 语句后，会返回更新的记录数量。

<update>元素属性与<insert>元素属性一致，这里不重复介绍。

知识点 5-<delete>元素

教师讲解<delete>元素。

<delete>元素用于映射删除语句，在执行完<delete>元素中的 SQL 语句之后，会返回删除的记录数量。

接下来讲<delete>元素中可以配置的属性。

知识点 6-<sql>元素

教师讲解<sql>元素。

在一个映射文件中，通常需要定义多条 SQL 语句，这些 SQL 语句的组成可能有一部分是相同的(如多条 select 语句中都查询相同的 id、username 字段)，如果每一个 SQL 语句都重写一遍相同的部分，势必会增加代码量，导致映射文件过于臃肿。

针对以上问题，我们可以在映射文件中使用 MyBatis 所提供的<sql>元素，将这些 SQL 语句中相同的组成部分抽取出来，然后在需要的地方引用。

<sql>元素的作用是定义可重用的 SQL 代码片段，它可以被包含在其他语句中。<sql>元素可以被静态地(在加载参数时)参数化，<sql>元素不同的属性值通过包含的对象发生变化。

四、归纳总结

教师回顾本节课所讲的内容，并通过测试题的方式引导学生解答问题并给予指导。

五、布置作业

教师通过高校教辅平台(<http://tch.ityxb.com>)布置本节课作业以及下节课的预习作业。

第四课时

(<resultMap>元素、案例：员工管理系统)

一、复习巩固

教师通过上节课作业的完成情况，对学生吸收不好的知识点进行再次巩固讲解。

二、通过直接导入的方式导入新课

掌握了 MyBatis 映射文件中的部分元素后，接下学习映射文件中的<resultMap>元素以及本章案例系统。



三、新课讲解

知识点 1-`<resultMap>`元素

教师讲解`<resultMap>`元素。

`<resultMap>`元素表示结果映射集，是 MyBatis 中最重要也是功能最强大的元素。`<resultMap>`元素主要作用是定义映射规则、更新级联以及定义类型转化器等。

默认情况下，MyBatis 程序在运行时会自动将查询到的数据与需要返回的对象的属性进行匹配赋值（数据表中的列名与对象的属性名称完全一致才能匹配成功并赋值）。然而实际开发时，数据表中的列和需要返回的对象的属性可能不会完全一致，这种情况下 MyBatis 不会自动赋值，这时就需要使用`<resultMap>`元素进行结果集映射。

接下来，通过一个具体的案例演示使用`<resultMap>`元素进行结果集映射，具体步骤如下。

- (1) 在名称为 mybatis 的数据库中，创建一个 t_student 表，并插入几条测试数据。
- (2) 在项目 src/main/java 目录的下创建 com.itheima.pojo 包，并在 com.itheima.pojo 包中创建实体类 Student，用于封装学生信息。在类中定义 id、name 和 age 属性，以及属性的 getter/setter 方法和 toString() 方法。
- (3) 在项目的 src/main/java 目录下创建 com.itheima.mapper 包，并在该包中创建映射文件 StudentMapper.xml，并在映射文件中编写映射查询语句。StudentMapper.xml 映射文件主要用于实现 SQL 语句和 Java 对象之间的映射，使 SQL 语句查询出来的关系型数据能够被封装成 Java 对象。
- (4) 在核心配置文件 mybatis-config.xml 中，引入 StudentMapper.xml，将 StudentMapper.xml 映射文件加载到程序中。
- (5) 在项目的 src/test/java 目录下的 Test 包中创建测试类 MyBatisTest，在测试类中，编写测试方法 findAllStudentTest()，用于测试`<resultMap>`元素实现查询结果的映射。

多学一招：使用工具类创建 SqlSession 对象

知识点 2-案例：员工管理系统

教师讲解案例：员工管理系统。

案例目标：

本案例要求在数据库中创建一个 employee 表，并利用本章所学知识完成一个员工管理系统，该系统需要实现以下几个功能。

1. 根据 id 查询员工信息。
2. 新增员工信息。
3. 根据 id 修改员工信息。
4. 根据 id 删除员工信息。

员工表 (employee) 如下：

员工表 (employee)			
员工编号 (id)	商品名称 (name)	员工年龄 (age)	员工职位 (position)



1	张三	20	员工
2	李四	18	员工
3	王五	35	经理

实现步骤：

1. 搭建项目

创建一个名称为 mybatisdemo 的项目，项目的具体搭建过程请参考 1.3 节。

2. 数据准备

在 mybatis 数据库中创建 employee 表，并在 employee 表中插入几条数据。

3. POJO 类准备

在 项目 src/main/java 目 录 下 创 建 com.itheima.pojo 包，在 com.itheima.pojo 包下创建持久化类 Employee，并在类中声明 id（编号）、name（姓名）、age（年龄）和 position（职位）属性，以及属性对应的 getter/setter 方法。

4. 编写映射文件

在 项目 src/main/java 目 录 下 创 建 com.itheima.mapper 包，在 com.itheima.mapper 包下创建映射文件 EmployeeMapper.xml，该文件主要用于实现 SQL 语句和 Java 对象之间的映射。

5. 修改 mybatis-config.xml 核心配置文件

在 mybatis-config.xml 映 射 文 件 的 <mappers> 元 素 下 添 加 EmployeeMapper.xml 映射文件路径的配置，用于将 EmployeeMapper.xml 映射文件加载到程序中。

6. 编写 MyBatisUtils 工具类

在 项目 src/main/java 目 录 下 创 建 com.itheima.utils 包，在 com.itheima.utils 包下创建 MyBatisUtils 工具类，该类用于封装读取配置文件信息的代码。

7. 编写测试类

在 项目 src/test/java 目 录 下 创 建 Test 包，在 Test 包下创建测试类，该类主要用于程序测试。

四、归纳总结

教师回顾本节课所讲的内容，并通过测试题的方式引导学生解答问题并给予指导。

五、布置作业

教师通过高校教辅平台（<http://tch.ityxb.com>）布置本节课作业以及下节课的预习作业。

第五、六课时（上机练习）

上机一：（考察知识点为 SqlSessionFactoryBuilder 对象、SqlSessionFactory 对象、SqlSession 对象、MyBatis 配置文件的主要元素、<properties>元素、<settings>元素、<typeAliases>元素、<environments>元素、<mappers>元素、MyBatis 映射文件中的常用元素、<select>元素、<insert>元素、<update>元素、<delete>元素、<sql>元素、<resultMap>元素、案例：员工管理系统）

形式：单独完成

题目：

要求根据员工表在数据库中创建一个 employee 表，并利用本章所学知识完



	<p>成一个员工管理系统，该系统需要实现以下几个功能。</p> <ol style="list-style-type: none">1. 根据 id 查询员工信息。2. 新增员工信息。3. 根据 id 修改员工信息。4. 根据 id 删除员工信息。 <p>注意：创建一个名称为 mybatisdemo 的项目，项目的具体搭建过程请参考 1.3 节。</p>
教学后记	