



# 黑马程序员

## 《Java EE 企业级应用开发教程

## 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 5 章 MyBatis 的注解开发	计划课时	3 课时
教学引入	前面的章节介绍了 MyBatis 的基本用法、关联映射、动态 SQL 和缓存机制等知识，所有的配置都是基于 XML 文件完成的，但在实际开发中，大量的 XML 配置文件的编写是非常繁琐的，为此，MyBatis 提供了更加简便的基于注解的配置方式。本章将对 MyBatis 的注解开发进行详细讲解。		
教学目标	<ul style="list-style-type: none"> <li>● 使学生掌握基于注解的单表增删改查</li> <li>● 使学生熟悉基于注解的一对一关联查询</li> <li>● 使学生熟悉基于注解的一对多关联查询</li> <li>● 使学生熟悉基于注解的多对多关联查询</li> </ul>		
教学重点	<ul style="list-style-type: none"> <li>● @Select 注解</li> <li>● @Insert 注解</li> <li>● @Update 注解</li> <li>● @Delete 注解</li> <li>● @Param 注解</li> </ul>		
教学难点	<ul style="list-style-type: none"> <li>● 多对多查询</li> </ul>		
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;"><b>第一课时</b></p> <p style="text-align: center;">（@Select 注解、@Insert 注解、@Update 注解、@Delete 注解、@Param 注解）</p> <p>一、创设情景，导入新课</p> <p>我们知道，在代码中，如果给每行代码都写上一定的注释，可以大大提高阅读程序的效率，也有助于开发人员对代码的理解。当然除了 XML 的映射方式外，还可以使用 MyBatis 基于注解的方式来实现数据库的表之间的映射关系，这是为了避免书写大量繁琐的 XML 代码。本节课，将对 @Select 注解、@Insert 注解、@Update 注解等内容进行详细讲解。</p> <p>二、新课讲解</p> <p><b>知识点 1-@Select 注解</b></p> <p>教师通过多媒体演示 PPT 内容讲解 @Select 注解。</p> <p>@Select 注解用于映射查询语句，其作用等同于 XML 配置文件中的 &lt;select&gt; 元素。</p> <p>下面通过一个案例演示 @Select 注解的使用，该案例要求根据员工的 id 查找员工信息，案例具体实现步骤如下。</p> <p>(1) 创建数据表。</p>		



在 mybatis 数据库中创建名为 tb\_worker 的数据表，同时预先插入几条测试数据。

(2) 创建持久化类 Worker。

在项目的 com.itheima.pojo 包下创建持久化类 Worker，在 Worker 类中定义 id、员工姓名、年龄、性别、工号等属性以及属性对应的 getter/setter 方法。

(3) 创建 WorkerMapper 接口。

在项目的 src/main/java 目录下创建 com.itheima.dao 包，并在 com.itheima.dao 包下创建 WorkerMapper 接口，用于编写 @Select 注解映射的 select 查询方法。

(4) 编写配置文件。

在核心配置文件 mybatis-config.xml 中的 <mappers> 元素下引入 WorkerMapper 接口，将 WorkerMapper.java 接口加载到核心配置文件中。

(5) 编写测试类。

为了验证上述配置，可以在测试类 MyBatisTest 中，编写测试方法 findWorkerByIdTest()。

### 知识点 2-@Insert 注解

教师讲解 @Insert 注解。

@Insert 注解用于映射插入语句，其作用等同于 XML 配置文件中的 <insert> 元素。

下面通过一个案例演示 @Insert 注解的使用，该案例要求实现员工信息的插入，案例具体实现步骤如下。

(1) 添加插入接口。

在 WorkerMapper 接口中添加向 tb\_worker 数据表插入数据的方法 insertWorker()，并在方法上添加 @Insert 注解。

(2) 编写测试类。

为了验证上述配置，可以在测试类 MyBatisTest 中，编写测试方法 insertWorkerTest()。

### 知识点 3-@Update 注解

教师讲解 @Update 注解。

@Update 注解用于映射更新语句，其作用等同于 XML 配置文件中的 <update> 元素。

下面通过一个案例演示 @Update 注解的使用，该案例要求实现员工信息的修改，案例具体实现步骤如下。

(1) 添加更新接口。

在 WorkerMapper 接口中添加更新 tb\_worker 表中数据的方法，并在方法上添加 @Update 注解。

(2) 编写测试类。

为了验证上述配置，可以在测试类 MyBatisTest 中，编写测试方法 updateWorkerTest()。

### 知识点 4-@Delete 注解



教师讲解@Delete 注解。

@Delete 注解用于映射删除语句，其作用等同于 XML 配置文件中的<delete>元素。

下面通过一个案例演示@Delete 注解的使用，该案例要求实现员工信息的删除，案例具体实现步骤如下。

(1) 添加删除接口。

在 WorkerMapper 接口中添加删除数据库中数据的方法，并在方法上添加@Delete 注解。

(2) 编写测试类。

为了验证上述配置，可以在测试类 MyBatisTest 中，编写测试方法 deleteWorkerTest()。

### 知识点 5-@Param 注解

教师讲解@Param 注解。

@Param 注解的功能是指定 SQL 语句中的参数，通常用于 SQL 语句中参数比较多的情况。

下面通过一个案例演示@Param 注解的使用，该案例要求根据员工的 id 和姓名查询员工信息，案例具体实现步骤如下。

(1) 添加查询接口。

在 WorkerMapper 接口中添加多条件查询的方法，具体代码如下所示。

```
@Select("select * from tb_worker where id = #{param01}
        and name = #{param02}")
Worker selectWorkerByIdAndName(@Param("param01") int id,
                                @Param("param02") String name);
```

(2) 编写测试类。

为了验证上述配置，可以在测试类 MyBatisTest 中，编写测试方法 selectWorkerByIdAndNameTest()。

### 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

### (一对一查询、一对多查询、多对多查询、案例：基于 MyBatis 注解的学生管理程序)

#### 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 的基于注解的单表增删改查后，接下来继续学习 MyBatis 的基于注解的关联查询等。

#### 三、新课讲解



### 知识点 1-一对一查询

教师讲解一对一查询。

MyBatis 中使用@One 注解实现数据表的一对一关联查询，其作用等同于 XML 配置文件中的<association>元素。

接下来，以 4.2 节中使用的 tb\_idcard 和 tb\_person 数据表为例，详细讲解基于注解@One 实现 tb\_idcard 和 tb\_person 数据表之间的一对一关联查询，具体步骤如下。

(1) 本案例使用 4.2 节中的 IdCard 类和 Person 类作为持久类。

(2) 创建接口。

在项目的 com.itheima.dao 包下创建 IdCardMapper 接口，在该接口中编写 selectIdCardById() 方法，通过 id 查询人员对应的身份证信息。

在项目的 com.itheima.dao 包下创建 PersonMapper 接口，在该接口中编写 selectPersonById()，通过 id 查询人员信息。

注意：在@Results 注解中，可以包含多个@Result 注解，一个@Result 注解完成实体类中一个属性和数据表中一个字段的映射。

在@Result 注解中，有 column、property 和 one 三个属性，它们的含义分别如下所示。

- property 属性用来指定关联属性，这里为 card。
- column 属性用来指定关联的数据库表中的字段，这里为 card\_id。
- one 属性用来指定数据表之间属于哪种关联关系，通过@One 注解表明数据表 tb\_idcard 和 tb\_person 之间是一对一关联关系。

(3) 编写配置文件。

在核心配置文件 mybatis-config.xml 中的<mappers>元素下引入 IdCardMapper 和 PersonMapper 接口。

需要注意的是，由于 mybatis-config.xml 文件中的扫描方式是从上往下扫描，所以<mappers>元素下引入 IdCardMapper 和 PersonMapper 接口的位置，必须在引入 IdCardMapper.xml 和 PersonMapper.xml 文件位置前面，否则程序将会首先读取到引入的 IdCardMapper.xml 和 PersonMapper.xml 文件，程序将会报错。

(4) 编写测试类。

为了验证上述配置，可以在测试类 MyBatisTest 中，编写测试方法 selectPersonByIdTest()。

### 知识点 2-一对多查询

教师讲解一对多查询。

MyBatis 使用@Many 注解实现数据表的一对多关联查询，@Many 注解的作用等同于 XML 配置文件中的<collection>元素。

接下来，以 4.3 节中的 tb\_user 和 tb\_orders 数据表为例，详细讲解基于@Many 注解配置实现 tb\_user 和 tb\_orders 数据表之间的一对多关联查询，具体步骤如下。

(1) 本案例使用 4.3 节中的 Users 类和 Orders 类作为持久类。

(2) 创建接口。

在项目的 com.itheima.dao 包下创建 OrdersMapper 接口，在该接口中编写 selectOrdersByUserId() 方法，通过 user\_id 查询用户对应的订单信息。



在项目的 `com.itheima.dao` 包下创建 `UsersMapper` 接口，在该接口中编写 `selectUserById()` 方法，通过 `id` 查询用户信息。

(3) 编写配置文件。

在核心配置文件 `mybatis-config.xml` 中的 `<mappers>` 元素下引入 `UsersMapper` 和 `OrdersMapper` 接口，将这两个接口加载到核心配置文件中。

需要注意的是，由于 `mybatis-config.xml` 文件中的扫描方式是从上往下扫描，所以 `<mappers>` 元素下引入 `UsersMapper` 和 `OrdersMapper` 接口的位置，必须在引入 `UsersMapper.xml` 和 `OrdersMapper.xml` 文件位置前面，否则程序将会首先读取到引入的 `UsersMapper.xml` 和 `OrdersMapper.xml` 文件，程序将会报错。

(4) 编写测试类。

为了验证上述配置，可以在测试类 `MyBatisTest` 中，编写测试方法 `selectUserByIdTest()`。

### 知识点 3-多对多查询

教师讲解多对多查询。

接下来，以 4.4 节中使用的订单表 `tb_orders`、商品表 `tb_product` 和中间表 `tb_ordersitem` 为例，详细讲解 `tb_orders` 和 `tb_product` 数据表之间基于注解的多对多关联查询，具体步骤如下。

(1) 本案例使用 4.3 节中的 `Orders` 类和 4.4 节中的 `Product` 类作为持久类并在订单持久化类 (`Orders.java`) 中增加商品集合的属性及其对应的 `getter/setter` 方法。

(2) 在项目的 `com.itheima.dao` 包下创建 `ProductMapper` 接口，在该接口编写 `selectProductByOrdersId()` 方法，通过 `user_id` 查询用户对应的订单信息。

(3) 在核心配置文件 `mybatis-config.xml` 中的 `<mappers>` 元素下引入 `ProductMapper` 和 `OrdersMapper` 接口，将这两个接口加载到核心配置文件中。

(4) 为了验证上述配置，可以在测试类 `MyBatisTest` 中，编写测试方法 `selectOrdersByIdTest()`。

### 知识点 4-案例：基于 MyBatis 注解的学生管理程序

教师讲解案例：基于 MyBatis 注解的学生管理程序。

任务目标：

现有一个学生表 `s_student` 和一个班级表 `c_class`，其中，班级表 `c_class` 和学生表 `s_student` 是一对多的关系。

学生表 s_student			
学生 id id	学生姓名 name	学生年龄 age	所属班级 cid
1	张三	18	1
2	李四	18	2
3	王五	19	2
4	赵六	20	1

  

班级表 c_class	
班级 id	班级名称





id	classname
1	一班
2	二班

请使用 MyBatis 注解完成以下几个要求：

(1) MyBatis 注解实现查询操作

根据表 s\_student 和表 c\_class 在数据库分别创建一个学生表 s\_student 和一个班级表 c\_class，并查询 id 为 2 的学生的信息。

(2) MyBatis 注解实现修改操作

修改 id 为 4 的学生的姓名修改为李雷，年龄修改为 21。

(3) MyBatis 注解实现一对多查询

查询出二班所有学生的信息。

实现步骤：MyBatis 注解实现查询操作。

1. 项目搭建

创建一个名称为 mybatis-demo05 的项目，项目的具体搭建过程请参考 1.3 节。

2. 数据库准备

在名为 mybatis 的数据库中，创建两个数据表，分别为学生表 s\_student 和班级表 c\_class，同时在表中预先插入几条测试数据。

3. POJO 类准备

在项目的 src/main/java 目录下创建 com.itheima.pojo 包，在 com.itheima.pojo 包中创建持久化类 IClass，并在类中定义相关属性和方法，该类用于封装 IClass 对象的 id、班级名称以及关联的学生集合等属性。

在 com.itheima.pojo 包中，创建持久化类 IStudent，并在类中定义相关属性和方法，该类用于封装 IStudent 对象的 id、姓名和年龄等属性。

4. 创建 Mapper 接口

在项目的 src/main/java 目录下创建 com.itheima.dao 包，并在 com.itheima.dao 包下创建 IStudentMapper 接口，用于编写 @Select 注解映射的 select 查询语句。

5. 修改 mybatis-config.xml 核心配置文件

在核心配置文件 mybatis-config.xml 中的 <mappers> 元素下引入 IStudentMapper 接口，将 IStudentMapper 接口加载到核心配置文件中。

```
<mapper class="com.itheima.dao.IStudentMapper"/>
```

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. 编写测试方法

为了验证上述配置，在项目的 src/test/java 目录下创建 Test 文件夹，在 Test 文件夹创建测试类 MyBatisTest，在测试类 MyBatisTest 中编写测试方法 findIStudentByIdTest()。

实现步骤：MyBatis 注解实现修改操作。

1. 修改 Mapper 接口

在 IStudentMapper 接口中添加更新 s\_student 表中数据的方法，并在方



法上添加@Update 注解。

## 2. 编写测试方法

为了验证上述配置，可以在测试类 MyBatisTest 中，编写测试方法 updateIStudentTest()。

实现步骤：MyBatis 注解实现一对多查询。

## 1. 修改 Mapper 接口

(1) 在 IStudentMapper 接口中编写 selectStudentByCid() 方法，通过 cid 查询对应班级中的学生信息。

(2) 在项目的 com.itheima.dao 包下创建 IClassMapper 接口，在该接口中编写 selectClassById() 方法，通过 id 查询班级信息。

## 2. 修改 mybatis-config.xml 核心配置文件

在核心配置文件 mybatis-config.xml 中的<mappers>元素下引入 IClassMapper 接口，将 IClassMapper 接口加载到核心配置文件中。

## 3. 编写测试方法

为了验证上述配置，可以在测试类 MyBatisTest 中，编写测试方法 selectClassByIdTest()。

## 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（上机练习）

**上机一：（考察知识点为@Select 注解、@Insert 注解、@Update 注解、@Delete 注解、@Param 注解、一对一查询、一对多查询、多对多查询、案例：基于 MyBatis 注解的学生管理程序）**

**形式：单独完成**

**题目：**

请使用 MyBatis 注解完成以下几个要求：

### (1) MyBatis 注解实现查询操作

根据表 1 和表 2 在数据库分别创建一个学生表 s\_student 和一个班级表 c\_class，并查询 id 为 2 的学生的信息。

### (2) MyBatis 注解实现修改操作

修改 id 为 4 的学生的姓名修改为李雷，年龄修改为 21。

### (3) MyBatis 注解实现一对多查询

查询出二班所有学生的信息。

注意：创建一个名称为 mybatis-demo05 的项目，项目的具体搭建过程请参考 1.3 节。

表 1 学生表 (s\_student)

学生 id id	学生姓名 uid	学生年龄 age	所属班级 uid
-------------	-------------	-------------	-------------





		1	张三	18	1
		2	李四	18	2
		3	王五	19	2
		4	赵六	20	1
		表 2 班级表（c_class）			
		班级 id	班级名称		
		id	classname		
		1	一班		
		2	二班		
教学后记					