

第四章 MyBatis 关联映射

课程 目标

- 01 - 掌握动态SQL主要元素的使用
- 02 - 理解关联关系的基本概念
- 03 - 掌握基于XML的一对一关联映射方式
- 04 - 掌握基于XML的一对多关联映射方式
- 05 - 掌握基于XML的多对多关联映射方式
- 06 - 掌握基于注解的一对一关联映射方式
- 07 - 掌握基于注解的一对多关联映射方式
- 08 - 掌握基于注解的多对多关联映射方式

目录

01

项目需求

02

背景知识

03

项目实施

04

经典问题强化

05

本章小结

01

项目需求

项目经理老王：小王，之前我们已经学习了MyBatis的基础以及核心组件，这些都是运用Mybatis进行开发的基础。

程序员小王：现在使用Mybatis完成一个单表的增删改查功能我已经掌握了，但是在实际的业务中，表与表之间的关联关系可不是只有单表这种简单情况，还有多表之间的关联查询以及一些复杂的sql查询，这些使用Mybatis可以实现么？

项目经理老王：可以的，Mybatis可不是只能做单表的查询，它还可以使用动态SQL完成复杂的sql查询，比如比较查询、模糊查询、批量查询，以及可以实现多表之间的关联查询。现在客户要求系统能针对不同的用户可以操作的模块不同，需要实现一个权限管理功能。

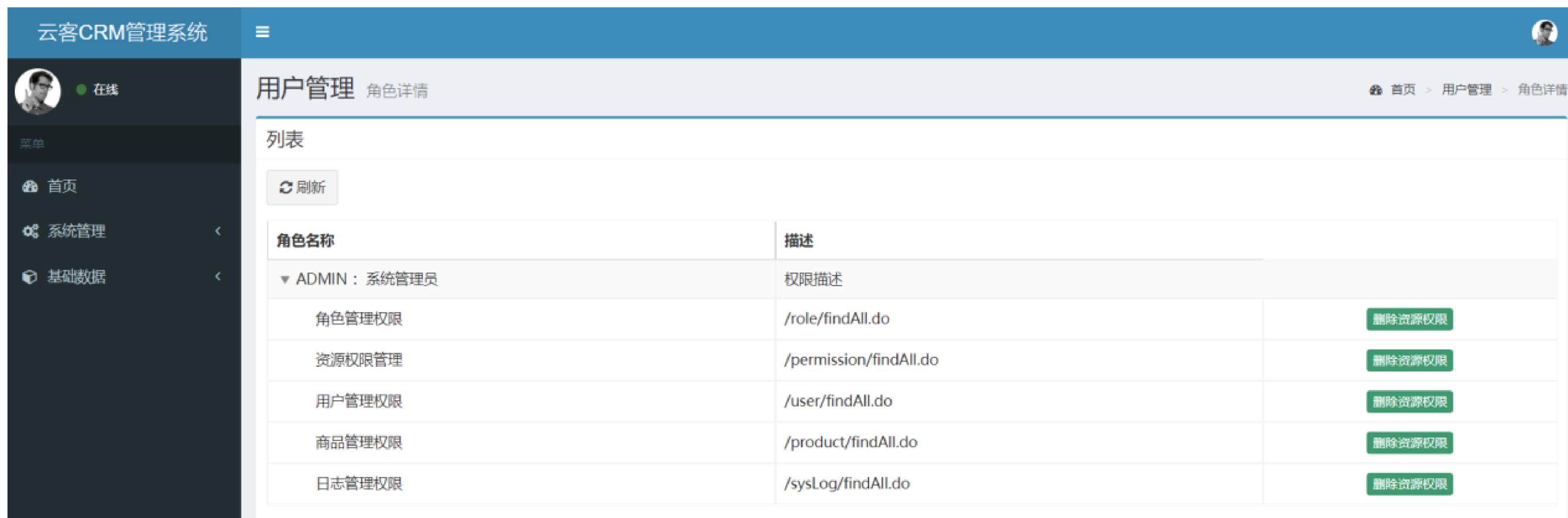
程序员小王：对于权限管理功能会涉及到用户、角色、资源几张表，我在学习Mybatis的关联映射之后就可以实现这个功能模块。

角色管理：权限列表页面展示了当前所有角色的列表，每个角色都有对应的描述和操作。用户可以进行增加、删除、修改等操作。为了方便用户同时操作多个角色，提供了全选和反选功能。

The screenshot shows the 'Role Management' interface. The top navigation bar includes the system name '云客CRM管理系统', a menu icon, and a user profile. The left sidebar contains navigation options: '菜单', '首页', '系统管理', and '基础数据'. The main content area is titled '角色管理 全部角色' and features a breadcrumb trail '首页 > 角色管理 > 全部角色'. Below the title, there are buttons for '新建角色' and '刷新'. The main part of the interface is a table with columns for selection, ID, role name, description, and actions. Two roles are listed: 'ADMIN' (System Administrator) and 'King' (Total Authority). Each role has three action buttons: '权限详情', '删除角色', and '添加权限'.

<input type="checkbox"/>	ID	角色名称	描述	操作
<input type="checkbox"/>	1	ADMIN	系统管理员	权限详情 删除角色 添加权限
<input type="checkbox"/>	3	King	总统权限(无视一切防御)	权限详情 删除角色 添加权限

权限详情：展示某个角色所拥有的权限列表，一个角色可以拥有多个权限，例如ADMIN角色是管理员，它拥有角色管理、资源管理、用户管理、商品管理、日志管理等多个权限。



云客CRM管理系统

用户管理 角色详情

列表

刷新

角色名称	描述	
ADMIN : 系统管理员	权限描述	
角色管理权限	/role/findAll.do	删除资源权限
资源权限管理	/permission/findAll.do	删除资源权限
用户管理权限	/user/findAll.do	删除资源权限
商品管理权限	/product/findAll.do	删除资源权限
日志管理权限	/sysLog/findAll.do	删除资源权限

资源权限列表：该页面展示了所有资源权限列表，每个资源权限都对应着对某一特定资源的管理，例如用户管理权限就可以对用户进行管理。

The screenshot shows a web interface for '云客CRM管理系统' (Yunke CRM Management System). The main content area is titled '资源权限管理 全部权限' (Resource Permission Management All Permissions). Below the title, there are two buttons: '添加资源权限' (Add Resource Permission) and '刷新' (Refresh). A table lists the permissions with columns for selection, ID, permission name, URL, and actions. The table contains six rows of data.

<input type="checkbox"/>	ID	权限名称	URL	操作
<input type="checkbox"/>	107	角色管理权限	/role/findAll.do	详情 删除资源权限
<input type="checkbox"/>	1943	资源权限管理	/permission/findAll.do	详情 删除资源权限
<input type="checkbox"/>	23527	用户管理权限	/user/findAll.do	详情 删除资源权限
<input type="checkbox"/>	23528	商品管理权限	/product/findAll.do	详情 删除资源权限
<input type="checkbox"/>	23529	日志管理权限	/sysLog/findAll.do	详情 删除资源权限

资源权限详情页面可以查看某一资源所对应的所有角色及其权限，这些角色具有对该资源的管理能力。不同的角色所拥有的资源权限是不同的，比如开发人员可以对日志进行管理，而销售人员可以对商品进行管理。

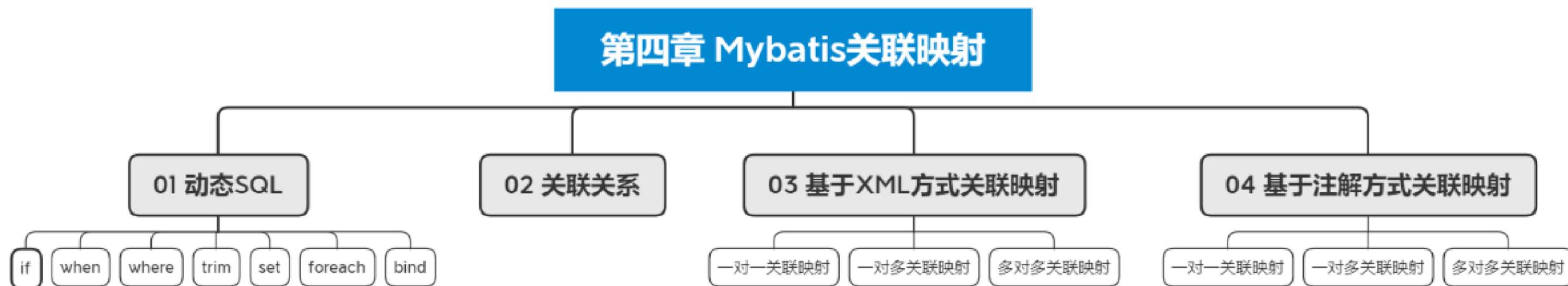
The screenshot shows the 'Resource Permission Management' page in the 'Yunke CRM Management System'. The page title is '资源权限管理 权限详情'. The breadcrumb navigation is '首页 > 资源权限管理 > 资源权限详情'. The page content includes a '列表' (List) section with a '刷新' (Refresh) button. Below the button is a table with the following data:

资源权限名称	详情描述	
▼ 角色管理权限: (/role/findAll.do)		
ADMIN	系统管理员	删除角色
King	总统权限(无视一切防御)	删除角色

At the bottom of the table, there is a '添加角色' (Add Role) button.

02

背景知识



动态SQL是MyBatis的强大特性之一。如果使用过JDBC 或其它类似的框架，应该能理解根据不同条件拼接 SQL 语句的麻烦，例如拼接时要确保不能忘记添加必要的空格，还要注意去掉列表最后一个列名的逗号等。

表 4-1 Mybatis 的动态 SQL 包含的主要元素

元素名称	描述
<if>	如果满足条件，就包含其中的内容。
<choose>	包含一组 <when> 和一个 <otherwise> 元素，只会包含第一个满足条件的 <when> 元素或者 <otherwise> 元素。
<when>	用于 <choose> 元素中，表示如果条件满足就包含其中的内容。
<otherwise>	用于 <choose> 元素中，表示如果没有 <when> 元素的条件满足，就包含其中的内容。
<trim>	可以用于在 SQL 语句中去除多余的逗号和空格等。
<where>	用于在 SQL 语句中添加 WHERE 子句，并自动去除多余的 AND 或 OR 连接符。
<set>	用于在 UPDATE 语句中设置要更新的 <u>列及其值</u> 。
<foreach>	用于在 SQL 语句中执行循环操作。
<bind>	用于将一个 OGNL 表达式的值赋给一个变量，然后在后面的 SQL 语句中引用该变量。
<sql>	可以用于定义一个可重用的 SQL 片段。
<include>	可以用于在 SQL 语句中引用另一个 SQL 片段。

通过之前的学习，我们已经能够使用MyBatis以面向对象的方式对数据库进行操作，但这些操作只是针对单表。在实际的开发中，对数据库的操作常常会涉及多张表，这在面向对象中就会映射成对象与对象之间的三种关联关系。

下面对这些关系进行具体说明：

一对一关系：是指对于实体集A与实体集B，A中的每一个实体最多与B中一个实体有关系。反之，在实体集B中的每个实体最多与实体集A中的一个实体有关系。例如每个公民只能拥有一个身份证号，而每个身份证号只能属于一个公民。

一对多关系：是指实体集A与实体集B中至少有 $N(N>0)$ 个实体有关系，并且实体集B中每一个实体至多与实体集A中一个实体有关系。例如一个班级里可以有 multiple 名学生，而每名学生只能属于一个班级。

多对多关系是指实体集A中的每一个实体与实体集B中至少有 $M(M>0)$ 个实体有关系，并且实体集B中的每一个实体与实体集A中的至少 $N(N>0)$ 个实体有关系。例如一名学生可以选修多门课程，一门课程也可以被多名学生选修。

1. 基于XML方式一对一关联映射

在MyBatis中可以使用<association> 子元素来处理这种一对一关系。在<association>元素中，通常可以配置以下属性：

- property：指定映射到的实体类对象属性，与表字段一一对应。
- column：指定表中对应的字段。
- javaType：指定映射到实体对象属性的类型。
- select：指定引入嵌套查询的子SQL 语句，该属性用于关联映射中的嵌套查询。
- fetchType：指定在关联查询时是否启用延迟加载。fetchType 属性有lazy和eager 两个属性值，默认值为lazy（即默认关联映射延迟加载）。

2. 基于XML方式一对多关联映射

与一对一的关联关系相比，实际应用更多的是一对多(或多对一)关系。例如一个用户可以有多个订单，同时多个订单归一个用户所有。

那么使用MyBatis 怎么处理这种一对多关联关系呢？我们可以使用<collection>元素。

<collection>元素的大部分属性与<association>元素相同，在应用时通常使用select属性以配置接口名和方法的方式指向SQL，同时将结果返回给对应的POJO对象。此外，<collection>元素还包含一个特殊属性ofType，它与javaType 属性对应，用于指定实体对象中集合类属性所包含的元素类型。

3. 基于XML方式多对多关联映射

在实际的项目开发中，多对多的关联关系非常常见。以用户和角色为例，一个用户可以拥有多种角色，而一个角色也可以包含多个用户，因此用户和角色之间存在多对多的关联关系。

如果需要查询用户的信息，同时也需要查询该用户拥有的所有角色，可以使用多对多查询的方式来实现。这样就能够方便地获取用户与其所拥有角色之间的关联信息。

以上内容仅为本文档的试下载部分，为可阅读页数的一半内容。如要下载或阅读全文，请访问：<https://d.book118.com/798125042001007005>