

resultMap和resultType

resultMap

一、基本介绍

resultMap元素是MyBatis 中最重要最强大的元素。它可以让你从 90% 的 JDBC ResultSets数据提取代码中解放出来，并在一些情形下允许你进行一些 JDBC 不支持的操作。实际上，在为一些比如连接的复杂语句编写映射代码的时候，一份resultMap 能够代替实现同等功能的长达数千行的代码。

ResultMap的设计思想是：对于简单的语句根本不需要配置显式的结果映射，而对于复杂一点的语句只需要描述它们的关系就行了。

二、使用场景

mybatis使用的类的属性名应该保持与sql里的列名（属性名）一样；
在mysql里不区分大小写；
一旦更改，便会出错

三、常用属性

- id属性， resultMap标签的标识。
- type属性， 返回值的全限定类名， 或类型别名。
- autoMapping属性， 值范围true（默认值）|false, 设置是否启动自动映射功能， 自动映射功能就是自动查找与字段名小写同名的属性名， 并调用setter方法。而设置为false后， 则需要**在resultMap内明确注明映射关系才会调用对应的setter方法。**

- **association**

- 1、作用：完成子对象的映射

- 2、属性

- property:映射数据库列的字段或属性。

- colum:数据库的列名或者列标签别名。

- javaTyp:完整java类名或别名。

jdbcType支持的JDBC类型列表列出的JDBC类型。这个属性只在insert, update或删除的时候针对允许空的列有用。

resultMap: 一个可以映射联合嵌套结果集到一个适合的对象视图上的ResultMap。这是一个替代的方式去调用另一个select语句。

property	需要时
column	数
javaType	一个完整的类名, 或者是MyBatis 通常会自行检测需要时
jdbcType	数据表支持的类型列表。这允许空的列有用。JDBC 需JDBC 编码,
typeHandler	使用这个属性可以覆写类型

四、实例

配置resultMap

```
<!-- 配置 查询结果的列名和实体类的属性名的对应关系 -->
<resultMap id="userMap" type="uSeR">
  <!-- 主键字段的对应 -->
  <id property="userId" column="id"></id>
  <!--非主键字段的对应-->
  <result property="userName" column="username"></result>
  <result property="userAddress" column="address"></result>
  <result property="userSex" column="sex"></result>
  <result property="userBirthday" column="birthday"></result>
</resultMap>
```

使用resultMap

```
!-- 配置查询所有操作 -->
<select id="findAll" resultMap="userMap">
select * from user
</select>
```

resultType

MyBatis中在查询进行select映射的时候，返回类型可以用resultType，也可以用resultMap，resultType是直接表示返回类型的，而resultMap则是对外部ResultMap的引用，但是resultType跟resultMap不能同时存在。

在MyBatis进行查询映射时，其实查询出来的每一个属性都是放在一个对应的Map里面的，其中键是属性名，值则是其对应的值。

①当提供的返回类型属性是resultType时，MyBatis会将Map里面的键值对取出赋给resultType所指定的对象对应的属性。所以其实MyBatis的每一个查询映射的返回类型都是ResultMap，只是当提供的返回类型属性是resultType的时候，MyBatis自动的给把对应的值赋给resultType所指定对象的属性。

②当提供的返回类型是resultMap时，因为Map不能很好表示领域模型，就需要自己再进一步的把它转化为对应的对象，这常常在复杂查询中很有作用。