

实体类字段不一致的问题处理方案

- 实体类里的字段 和 SQL 字段需要一致, 如果不一致的情况, 可以使用 AS 别名的方法处理

```
1 SELECT id, brand_name AS brandName, company_name AS companyName,  
2 ordered, description, `status` FROM tb_brand;
```

- resultMap 映射解决 字段不一致问题

```
1 <resultMap id="brand_resultmap" type="com.lovecoding.pojo.Brand">  
2     <result property="brandName" column="brand_name" />  
3     <result property="companyName" column="company_name" />  
4     <result property="description" column="company_name" />  
5     <result property="id" column="id" />  
6     <result property="ordered" column="ordered" />  
7     <result property="status" column="status" />  
8 </resultMap>
```

- 我们如果又重复使用的 SQL 字段 我们可以把这片字段 提取出来使用 <sql> 代码片段封装, 达到复用的效果

```
1 <sql id="brand_column" >  
2     id, brand_name, company_name, ordered, description, `status`  
3 </sql>
```

```
1 <mapper namespace="com.lovecoding.mapper.BrandMapper">  
2     <resultMap id="brand_resultmap" type="com.lovecoding.pojo.Brand">  
3         <result property="brandName" column="brand_name" />  
4         <result property="companyName" column="company_name" />  
5         <result property="description" column="company_name" />  
6         <result property="id" column="id" />  
7         <result property="ordered" column="ordered" />  
8         <result property="status" column="status" />  
9     </resultMap>  
10    <sql id="brand_column" >  
11        id, brand_name, company_name, ordered, description, `status`  
12    </sql>  
13    <select id="getBrandList" resultMap="brand_resultmap" >  
14        SELECT  
15            <include refid="brand_column"></include>  
16        FROM tb_brand;  
17    </select>
```

批量删除

- 我们主要是 利用 foreach 标签循环处理, foreach 可以用来循环 列表 数组 集合 等等数据
- 方面我们生成 复杂的SQL

```

1 <delete id="deleteByIds">
2     DELETE FROM tb_brand WHERE id IN
3     <foreach collection="array" item="id" separator="," open="(" close=")" >
4         #{id}
5     </foreach>
6 </delete>

```

批量添加

- 批量添加 也是利用 foreach 循环生成一个 insert SQL语句

```

1 <insert id="insertBatch">
2     INSERT INTO `tb_brand` (`brand_name`, `company_name`, `ordered`, `description`,
3     `status`) VALUES
4     <foreach collection="list" separator="," item="brand" >
5         (#{brand.brandName},#{brand.companyName},#{brand.ordered},#{
6         {brand.description},#{brand.status})
7     </foreach>
8 </insert>

```

collection 标签处理子查询 组装数据

对于一些 复杂的业务逻辑, 我们可以在 mybatis 里面 resultMap 定义集合, 然后执行子查询, 优点是 代码量少, 纯SQL 解决问题, 操作容易, 容错率高!!

当然了, 能哟个一条SQL 解决, 尽可能使用一条SQL 解决问题!!!

```

1 <mapper namespace="com.lovecoding.mapper.AuthorBookMapper">
2     <resultMap id="select_author_book" type="com.lovecoding.pojo.Author" >
3         <result column="username" property="username" />
4         <result column="password" property="password" />
5         <result column="email" property="email" />
6         <result column="id" property="id" />
7         <collection property="books" ofType="com.lovecoding.pojo.Book"
8         select="selectBook" column="{authorId=id}" />
9     </resultMap>
10
11     <select id="selectAuthorBook" resultMap="select_author_book">
12         SELECT * FROM tb_author_book WHERE authorId = #{authorId}
13     </select>
14
15     <select id="selectBook" resultMap="select_author_book">
16         SELECT * FROM tb_book WHERE authorId = #{authorId}
17     </select>
18 </mapper>

```

```
8      </resultMap>
9
10     <resultMap id="book" type="com.lovecoding.pojo.Book" >
11         <result column="author_id" property="authorId" />
12         <result column="book_name" property="bookName" />
13         <result column="id" property="id" />
14     </resultMap>
15
16     <select id="selectAuthor" resultMap="select_author_book">
17         SELECT * FROM `tb_author`;
18     </select>
19
20     <select id="selectBook" resultMap="book">
21         SELECT * FROM `tb_book` WHERE author_id = #{authorId}
22     </select>
23
24 </mapper>
```