1. MyBatis实现批量更新的四种方式

### Mybatis中实现批量更新的几种姿势，总有一款适合你

### 一、概述

mybatis中实现批量插入是很简单的，相比大家都知道，这里就不赘述，本文主要讲述如何实现批量更新。

下面介绍本文要讲的几种方式主要是在xml中实现，**不包含需要改动代码逻辑的方法**，这里，除了网上说的普通情况，还有适合mysql和oracle的批量更新方式： 1. case when 2. foreach成多条sql 3. ON DUPLICATE KEY UPDATE （mysql） 4. replace into （mysql）

**这次，我要讲的就是这四种方式。**

**如果大家正在寻找一个java的学习环境，或者在开发中遇到困难，可以[加入我们的java学习圈，点击即可加入](https://link.zhihu.com/?target=https://jq.qq.com/?_wv=1027&k=JVIRLUBf" \t "https://zhuanlan.zhihu.com/p/_blank)，共同学习，节约学习时间，减少很多在学习中遇到的难题。**

### 二、case when

这种方式实现的批量更新操作效率很低，而且，当更新的字段很多时，SQL语句会特别长。

|  |
| --- |
| <update id="updateBatch">  update t\_calendar\_extend  <trim prefix="set" suffixOverrides=",">  <trim prefix="modify\_time = case index" suffix="end,">  <foreach collection="list" item="item">  when #{item.index} then #{item.modifyTime}  </foreach>  </trim>  <trim prefix="user\_type = case index" suffix="end">  <foreach collection="list" item="item">  when #{item.index} then #{item.type}  </foreach>  </trim>  </trim>  <where>  index in (  <foreach collection="list" separator="," item="item">  #{item.index}  </foreach>  )  </where>  </update> |

这里，根据index值来更新modify\_time 和user\_type的值，有几个字段，就要遍历几遍，效率很低。

### 三、foreach成多条sql

这种方式最简单，就是用foreach组装成多条update语句，但Mybatis映射文件中的sql语句默认是不支持以" ; " 结尾的，也就是不支持多条sql语句的执行。所以需要在连接mysql的url上加 &allowMultiQueries=true 这个才可以执行。

<update id="updateBatch" parameterType="java.util.List">

<foreach collection="list" item="item" index="index" open="" close="" separator=";">

update tableName

<set>

name=${item.name},

name2=${item.name2}

</set>

where id = ${item.id}

</foreach> </update>

### 四、ON DUPLICATE KEY UPDATE

MYSQL中的ON DUPLICATE KEY UPDATE，是基于主键（PRIMARY KEY）或唯一索引（UNIQUE INDEX）使用的。

**如果已存在该唯一标示或主键就更新，如果不存在该唯一标示或主键则作为新行插入。**

<update id="updateBatch">

insert into t\_output\_calendar (index,

cal\_date, user\_type, create\_time,

modify\_time, delete\_flag

)

values

<foreach collection="list" item="item" index="index"

separator=",">

(

#{item.index,jdbcType=INTEGER},

#{item.calDate,jdbcType=TIMESTAMP},

#{item.type,jdbcType=TINYINT},

#{item.createTime,jdbcType=TIMESTAMP},

#{item.modifyTime,jdbcType=TIMESTAMP},

#{item.deleteFlag,jdbcType=TINYINT}

)

</foreach>

ON DUPLICATE KEY UPDATE modify\_time = VALUES(modify\_time), user\_type = VALUES(user\_type);

</update>

### 五、replace into

msql的replace into跟insert into的用法完全一样，但是它带有更新功能：

**如果发现表中已经有此行数据（根据主键或者唯一索引判断）则先删除此行数据，然后插入新的数据。否则，直接插入新数据。**

注意，它是先删除数据，然后再插入，这是和ON DUPLICATE KEY UPDATE不同的地方,如果当前的数据库用户没有删除权限，是不能使用replace into的。

示例：

<insert id="updateBatch" parameterType="java.util.List">

replace into t\_output\_calendar (index, cal\_date, user\_type, create\_time,

modify\_time, delete\_flag

)

values

<foreach collection="list" item="item" index="index"

separator=",">

(

#{item.index,jdbcType=INTEGER},

#{item.calDate,jdbcType=TIMESTAMP},

#{item.type,jdbcType=TINYINT},

#{item.createTime,jdbcType=TIMESTAMP},

#{item.modifyTime,jdbcType=TIMESTAMP},

#{item.deleteFlag,jdbcType=TINYINT}

)

</foreach>

</insert>