纯s'q'l

# 1、取出map的数据放入表中

有时候需要简单地把一个Map中所有的key和value获取出来，拼到sql语句中。MyBatis提供的一种方法是遍历Map中的entrySet，然后把key扔进index里面，value扔进item中。具体的一个使用的例子如下：

**[html]** [view plain](http://blog.csdn.net/clementad/article/details/55099432) [copy](http://blog.csdn.net/clementad/article/details/55099432)

1. **<insert** id="operationName" parameterType="map"**>**
2. INSERT INTO table\_name(hot\_word, cnt)
3. VALUES
4. **<foreach** item="value" index="key" collection="mapData.entrySet()" open="(" separator="),(" close=")"**>**
5. #{key}, #{value}
6. **</foreach>**
7. ON DUPLICATE KEY UPDATE
8. cnt=VALUES(cnt)
9. **</insert>**

# 2、结果为：list<map<String,Object>>

## 1、mapper

public interface HealerJeanMapper {  
  
  
 List<Map<String,Object>> sqlMap();  
}

## 2、mapper.xml

<select id="sqlMap" resultType="java.util.HashMap">  
 SELECT h.id as id ,h.subject as subject FROM healerjean h;  
</select>

## 3、解释：

### 1、返回类型必须是java.util.HashMap

### 2、map中的value 必须是Objecrt

## 4、controller测试

@RequestMapping("sqlMap")  
@ResponseBody  
public List<Map<String,Object>> sqlMap(){  
 return healerJeanMapper.sqlMap();  
}



# 3、mybatis 自定义参数、返回类型map

## 1、参数

|  |
| --- |
| <select id=*"getUserCount"* parameterMap=*"getUserCountMap"* statementType=*"CALLABLE"*resultType="java.util.HashMap  >  CALL ges\_user\_count(?,?)  </select>  <parameterMap type=*"java.util.Map"* id=*"getUserCountMap"*>  <parameter property=*"sexid"* mode=*"IN"* jdbcType=*"INTEGER"*/>  <parameter property=*"userCount"* mode=*"OUT"* jdbcType=*"INTEGER"*/>  </parameterMap> |

## 2、结果，返回字段一一匹配，想想之前@Results和@Result

|  |
| --- |
| <resultMap id=*"BaseResultMap"* type=*"com.taotao.pojo.TbUser"* >  <id column=*"id"* property=*"id"* jdbcType=*"BIGINT"* />  <result column=*"username"* property=*"username"* jdbcType=*"VARCHAR"* />  <result column=*"password"* property=*"password"* jdbcType=*"VARCHAR"* />  <result column=*"phone"* property=*"phone"* jdbcType=*"VARCHAR"* />  <result column=*"email"* property=*"email"* jdbcType=*"VARCHAR"* />  <result column=*"created"* property=*"created"* jdbcType=*"TIMESTAMP"* />  <result column=*"updated"* property=*"updated"* jdbcType=*"TIMESTAMP"* />  </resultMap> |

### 1、解释：

### 1、property和column分别指定实体类属性和数据表的列名。

# 4、If的使用

## 1、mapper接口

public interface CustomerMapper {  
  
 List<Customer> findCustomerList( @Param("name") String name,  
 @Param("status") Integer status,  
 @Param("offset") Integer offset,  
 @Param("limit") Integer limit);  
  
}

## 2、mapper.xml sql'语句

*<?*xml version="1.0" encoding="UTF-8"*?>*<!DOCTYPE mapper PUBLIC "-//mybatis.org//DTD Mapper 3.0//EN" "http://mybatis.org/dtd/mybatis-3-mapper.dtd" *>*<mapper namespace="com.duodian.admore.dao.mybatis.customer.CustomerMapper">  
  
 <select id="findCustomerList" resultType="com.duodian.admore.entity.db.customer.Customer">  
 SELECT t.\* from crm\_customer t  
 WHERE t.isVisiblisVisiblee = 1  
 <if test="name != null and name != ''">  
 and t.name like CONCAT('%','${name}','%' )  
 </if>  
 <if test="status != null">  
 and t.status = #{status}  
 </if>  
 order by t.id DESC  
 limit #{offset}, #{limit}  
 </select>  
  
 <select id="countCustomerList" resultType="java.lang.Long">  
 select count(\*) from crm\_customer t  
 WHERE t.isVisible = 1  
 <if test="name != null and name != ''">  
 and t.name like CONCAT('%','${name}','%' )  
 </if>  
 <if test="status != null">  
 and t.status = #{status}  
 </if>  
 </select>  
  
</mapper>

# 5、where include的使用（加入模糊查询筛选条件）自带去除第一个and

@Override  
 public Page<SignetApplyRecord> findVerifySignetData(Long admId, Pageable pageable, SignetQuery query) {  
  
 query.setStartDate(DateHelper.getDateFirstTime(query.getStartDate()));  
 query.setEndDate(DateHelper.getDateLastTime(query.getEndDate()));  
 query.setOffset(pageable.getOffset());  
 query.setLimit(pageable.getPageSize());  
  
   
   
 List<SignetApplyRecord> list = signetMapper.findVerifySignetList(query);  
   
  
 Long count = signetMapper.countVerifySignet(query);  
  
 return new PageImpl<>(list, pageable, count);  
 }

<select id="findVerifySignetList" resultType="com.duodian.admore.entity.db.signet.SignetApplyRecord">  
 SELECT t.*\**,d.nickName userNickName,ifnull(e.realName,f.realName) userAuthName,b.name applyAdmName,ura.remittanceAccount  
 FROM signet\_apply\_record t  
 LEFT JOIN user d ON d.id = t.userId  
 LEFT JOIN user\_auth\_personal e on e.userId = t.userId AND e.status = 1  
 LEFT JOIN user\_auth\_enterprise f on f.userId = t.userId AND f.status = 1  
 LEFT JOIN user\_email\_account g ON g.userId = t.userId  
 left join sys\_admin\_user b on b.id = t.applyAdmId  
 LEFT JOIN user\_remittance\_account ura ON ura.userId = t.userId  
 <if test="currAdmId4Auth != null">  
 LEFT JOIN sys\_admin\_user\_customer h ON h.admId = #{currAdmId4Auth} AND h.userId = t.userId  
 LEFT JOIN (SELECT wt.pid,count(*\**) cc FROM workflow\_task\_examine\_details wte LEFT JOIN workflow\_task wt ON wt.id = wte.taskId WHERE wt.taskType = 23 AND wte.admId = #{currAdmId4Auth} GROUP BY wt.pid) w ON w.pid = t.id  
 </if>  
 <where>  
 <include refid="findVerifySignetWhere"></include>  
 </where>  
 order by t.id desc  
 <if test="offset != null and limit != null">  
 limit #{offset}, #{limit}  
 </if>  
</select>  
  
  
<sql id="findVerifySignetWhere">  
 <if test="startDate != null">  
 <![CDATA[ and t.cdate >= #{startDate} ]]>  
 </if>  
 <if test="endDate != null">  
 <![CDATA[ and t.cdate <= #{endDate} ]]>  
 </if>  
 <if test="userParam != null and userParam != '' and userParam!= 'undefined'">  
 AND (t.userId = #{userParam} OR d.nickName LIKE CONCAT('%',#{userParam},'%' ) OR e.realName LIKE  
 CONCAT('%',#{userParam},'%' ) OR f.realName LIKE CONCAT('%',#{userParam},'%' ) OR g.email LIKE  
 CONCAT('%',#{userParam},'%' ))  
 </if>  
 <if test="taskType != null">  
 and t.taskType = #{taskType}  
 </if>  
 <if test="taskTypeList != null and taskTypeList.size() > 0">  
 and t.taskType IN  
 <foreach collection="taskTypeList" index="index" item="at" open="(" separator="," close=")">  
 #{at}  
 </foreach>  
 </if>  
 <if test="signetType != null">  
 and t.type = #{signetType}  
 </if>  
 <if test="status != null">  
 and t.status = #{status}  
 </if>  
 <if test="currAdmId4Auth != null ">  
 <![CDATA[ and (h.admId = #{currAdmId4Auth} OR w.cc > 0) ]]>  
 </if>  
</sql>

<select id="findUserAppsSpreadEffectList" resultType="com.duodian.admore.entity.db.apps.AppsSpreadEffectReport">  
 SELECT t.*\**,b.trackName,b.smallIcon FROM apps\_spread\_effect\_report t  
 LEFT JOIN apps\_user\_app a ON a.trackId = t.trackId  
 LEFT JOIN apps\_app b ON b.trackId = t.trackId  
 <where>  
 <if test="userId != null">  
 a.userId = #{userId}  
 </if>  
 <if test="startDate != null ">  
 <![CDATA[ and t.spreadDate >= #{startDate} ]]>  
 </if>  
 <if test="endDate != null ">  
 <![CDATA[ and t.spreadDate <= #{endDate} ]]>  
 </if>  
 </where>  
 order by a.id desc  
</select>

SELECT \* FROM OD\_LINK\_TYPE

<where>

LINK\_URI != BASE\_TYPE\_URI

<if test="linkUri != null">

AND LINK\_URI LIKE '%${linkUri}%'

</if>

<if test="linkLabel != null">

AND LINK\_LABEL LIKE '%${linkLabel}%'

</if>

<if test="baseTypeUri != null">

AND BASE\_TYPE\_URI=#{baseTypeUri}

</if>

</where>

# 5.、foreach list结合作为参数在mapper中的查询（taskTypeList 为List<Integer>）

<sql id="findVerifySignetWhere">  
 <if test="startDate != null">  
 <![CDATA[ and t.cdate >= #{startDate} ]]>  
 </if>  
 <if test="endDate != null">  
 <![CDATA[ and t.cdate <= #{endDate} ]]>  
 </if>  
 <if test="userParam != null and userParam != '' and userParam!= 'undefined'">  
 AND (t.userId = #{userParam} OR d.nickName LIKE CONCAT('%',#{userParam},'%' ) OR e.realName LIKE  
 CONCAT('%',#{userParam},'%' ) OR f.realName LIKE CONCAT('%',#{userParam},'%' ) OR g.email LIKE  
 CONCAT('%',#{userParam},'%' ))  
 </if>  
 <if test="taskType != null">  
 and t.taskType = #{taskType}  
 </if>  
 <if test="taskTypeList != null and taskTypeList.size() > 0"> and t.taskType IN  
 <foreach collection="taskTypeList" index="index" item="at" open="(" separator="," close=")">  
 #{at}  
 </foreach>  
 </if>  
 <if test="signetType != null">  
 and t.type = #{signetType}  
 </if>  
 <if test="status != null">  
 and t.status = #{status}  
 </if>  
 <if test="currAdmId4Auth != null ">  
 <![CDATA[ and (h.admId = #{currAdmId4Auth} OR w.cc > 0) ]]>  
 </if>  
</sql>

# 7、choose when

<if test="flag != null">  
 <choose>  
 <when test="flag == 1">  
 and t.status = 0  
 </when>  
 <when test="flag == 2">  
 and t.status = 1  
 </when>  
 <when test="flag == 3">  
 and t.expressStatus = 1  
 </when>  
 <when test="flag == 4">  
 and t.status = -2  
 </when>  
 </choose>  
</if>

# 8、ifnull （如果不是空返回第一个，否则返回第二个）

<select id="findRedStartSpread" parameterType="com.duodian.admore.dao.db.redstart.query.RedStartSpreadQuery" resultType="com.duodian.admore.dao.db.redstart.bean.RedStartHistoryBean">  
 SELECT  
 k.trackId,  
 e.smallIcon,  
 e.formattedPrice,  
 e.price,  
 e.fileSizeBytes,  
 e.trackName,  
 f.name admName,  
 a.nickName userName,  
 ifnull(b.realName,c.realName) authName,  
 DATE\_FORMAT(k.spreadDateStart, '%Y-%m-%d') AS ymd,  
 k.userId  
 FROM  
 redstart\_spread k

# 9。制作参数为map值进行传入（opt项目SkinsController）

## 1、controller接收参数

@RequestMapping("data")  
@ResponseBody  
public ResponseBean data(String name,

Integer type,

Integer status  
 ,@RequestParam(value = "page",defaultValue = "0") Integer page){  
   
 int pageSize = 15;  
 Pageable pageable = new PageRequest(page,pageSize);  
 Page<SkinAppInfoData> dataPage = skinsService.findSkinAppInfoList(pageable,"name",name,"type",type,"status",status);  
 return ResponseBean.buildSuccess(dataPage);  
  
}

## 2、service 制作map参数（下面的功能是模糊查询🏠分页）

### 1、service接口

public Page<SkinAppInfoData> findSkinAppInfoList(Pageable pageable, Object... param) throws AppException;

### 2、service开始实现 （pageable 主要是利用里面的参数制作limit参数的）

@Override  
public Page<SkinAppInfoData> findSkinAppInfoList(Pageable pageable, Object... param) throws AppException {  
 Map data = MyBatisHelper.mergeParameterMap(pageable,param);

if(data.get("startDate") != null){  
 Date startDate = (Date) data.get("startDate");  
 data.put("startDate", com.duodian.admore.core.helper.DateHelper.getDateFirstTime(startDate));  
}  
if(data.get("endDate") != null){  
 Date endDate = (Date) data.get("endDate");  
 data.put("endDate", com.duodian.admore.core.helper.DateHelper.getDateLastTime(endDate));  
}

List<SkinAppInfoData> dataList = skinsMapper.findSkinList(data);  
  
 for(SkinAppInfoData skinAppInfoData :dataList){  
 List<ChannelJson> channelJsonList = new ArrayList<>();  
 if(skinAppInfoData.getChannelJson()!=null&&!"".equals(skinAppInfoData.getChannelJson())) {  
 JSONArray jsonArray = JSONArray.fromObject(skinAppInfoData.getChannelJson());  
 for (int i = 0; i < jsonArray.size(); i++) {  
 ObjectMapper objectMapper = new ObjectMapper();  
 try {  
 ChannelJson channelJson = objectMapper.readValue(jsonArray.get(i).toString(), ChannelJson.class);  
 channelJsonList.add(channelJson);  
 } catch (IOException e) {  
 e.printStackTrace();  
 }  
 }  
 }  
 skinAppInfoData.setChannelJsonList(channelJsonList);  
  
 }  
  
 Long count = skinsMapper.countSkinList(data);  
 return new PageImpl<SkinAppInfoData>(dataList,pageable,count);  
}

### 3、MyBatisHelper工具栏

public class MyBatisHelper {  
 public static final String PARAM\_OFFSET = "offset";  
 public static final String PARAM\_LIMIT = "limit";  
  
 public MyBatisHelper() {  
 }  
  
 public static Map<String, Object> mergeParameterMap(Object... parameter) {  
 if (parameter.length % 2 != 0) {  
 throw new IllegalArgumentException("parameter须为key-value对应参数");  
 } else {  
 Map<String, Object> map = new HashMap();  
  
 for(int i = 0; i < parameter.length; i += 2) {  
 map.put(parameter[i].toString(), parameter[i + 1]);  
 }  
  
 return map;  
 }  
 }  
  
 public static Map<String, Object> mergeParameterMap(Pageable pageable, Object... parameter) {  
 if (parameter.length % 2 != 0) {  
 throw new IllegalArgumentException("parameter须为key-value对应参数");  
 } else {  
 Map<String, Object> map = new HashMap();  
 map.put("offset", pageable.getOffset());  
 map.put("limit", pageable.getPageSize());  
  
 for(int i = 0; i < parameter.length; i += 2) {  
 map.put(parameter[i].toString(), parameter[i + 1]);  
 }  
  
 return map;  
 }  
 }  
}

### 4、mapper.java （下面这两个其实就是专门用来做制作page分页的）

public interface SkinsMapper {  
  
  
 public List<SkinAppInfoData> findSkinList(Map param);  
  
 public Long countSkinList(Map param);  
  
}

*<?*xml version="1.0" encoding="UTF-8"*?>*<!DOCTYPE mapper PUBLIC "-//mybatis.org//DTD Mapper 3.0//EN" "http://mybatis.org/dtd/mybatis-3-mapper.dtd" *>*<mapper namespace="com.duodian.admore.dao.db.skins.SkinsMapper">  
  
 <select id="findSkinList" resultType="com.duodian.admore.data.skins.SkinAppInfoData">  
 SELECT  
 A1.`appid`,  
 A1.`appSecret`,  
 A1.`icon`,  
 A1.`name`,  
 A1.`type`,  
 A1.`cdate`,  
 A1.`downloadUrl`,  
 A1.`remark`,  
 A1.`status`,  
 A1.`trackId`,  
 A1.`bundleId`,  
 A1.`scheme`,  
 A1.`apnscer`,  
 A1.`password`,  
 A1.`udate`,  
 A1.`makerMemo`,  
 A1.`haveBackstage`,  
 A1.`channelJson`,  
 A1.`filePath`  
 FROM `skin\_app\_info\_check` a1  
 where A1.status not in (9)  
 <if test="name != null and name != ''">  
 AND (A1.trackId = #{name}  
 OR A1.name LIKE CONCAT('%',#{name},'%' )  
 OR A1.appid LIKE CONCAT('%',#{name},'%' )  
 OR A1.appSecret LIKE CONCAT('%',#{name},'%' )  
 )  
 </if>  
 <if test="type != null and type != '' ">  
 and A1.type = #{type}  
 </if>  
 <if test="status != null and status != '' ">  
 and A1.status = #{status}  
 </if>  
 order by A1.cdate desc  
 <if test="offset != null and limit != null">  
 limit #{offset}, #{limit}  
 </if>  
 </select>  
  
 <select id="countSkinList" resultType="java.lang.Long">  
 SELECT  
 count(*\**)  
 FROM `skin\_app\_info\_check` A1  
 where A1.status != 9  
 <if test="name != null and name != ''">  
 AND (A1.trackId = #{trackId} OR A1.name LIKE CONCAT('%',#{name},'%' ))  
 </if>  
 <if test="type != null">  
 and A1.type = #{type}  
 </if>  
 <if test="status != null">  
 and A1.status = #{status}  
 </if>  
 </select>  
</mapper>

## 5、有分组的

<select id="findCouponTaoKeDataByParam" resultType="com.duodian.youhui.entity.db.coupon.CouponTaoKeData">  
 SELECT c.itemTitle,COUNT(c.itemId),sum(c.estimateAmount),c.adzoneName,c.adzonePid FROM coupon\_taoke\_data c WHERE 1=1  
 <where>  
 <include refid="findCouponTaoKeDataByParamSQL"></include>  
 </where>  
 GROUP by c.itemId  
 <if test="offset != null and limit != null">  
 limit #{offset}, #{limit}  
 </if>  
</select>  
  
<select id="countFindCouponTaoKeDataByParam">  
 select count(*\**) from (  
 SELECT c.itemTitle FROM coupon\_taoke\_data c WHERE 1=1  
 <where>  
 <include refid="findCouponTaoKeDataByParamSQL"></include>  
 </where>  
 GROUP by c.itemId  
 ) as a  
</select>  
  
<sql id="findCouponTaoKeDataByParamSQL">  
 <if test="startDate != null ">  
 <![CDATA[ and c.createTime >= #{endDate} ]]>  
 </if>  
 <if test="endDate != null ">  
 <![CDATA[ and c.createTime <= #{endDate} ]]>  
 </if>  
 <if test="adzonePid != null and adzonePid != ''">  
 <![CDATA[ and c.adzonePid <= #{adzonePid} ]]>  
 </if>  
</sql>

# 10、模糊查询，对象作为参数传入，类似于9中的map

## 1、query对象 (查询参数)

public class SysUserQuery implements Serializable {  
  
 private static final long ***serialVersionUID*** = -4564423981924197001L;  
  
 private Long id;  
 private Integer offset;  
 private Integer limit;  
 private Date startDate;  
 private Date endDate ;  
 private String userid;  
 private String userParam;  
 private Integer status;  
  
  
  
}

## 2、controller层

@RequestMapping("data")  
@ResponseBody  
public ResponseBean data(@RequestParam(defaultValue = "0")Integer page, @RequestParam(defaultValue = "15")Integer pageSize, SysUserQuery query){  
 try {  
 Pageable pageable = new PageRequest(page,pageSize);  
 return ResponseBean.buildSuccess(sysDingUserService.getDingUserData(pageable,query));  
 } catch (AppException e) {  
 return ResponseBean.buildFailure(e.getMessage());  
 } catch (Exception e) {  
 logger.error(e.getMessage(),e);  
 return ResponseBean.buildFailure(ErrorCodeEnum.***系统错误***);  
 }  
}

## 3、service层，将pageable分页对象放入

@Override  
 public Page<SysDingUser> getDingUserData(Pageable pageable, SysUserQuery query) {  
  
 query.setOffset(pageable.getOffset());  
 query.setLimit(pageable.getPageSize());  
  
 List<SysDingUser> list = sysMapper.findSysDingUserList(query);  
  
 Long count = sysMapper.countSysDingUser(query);  
 return new PageImpl<>(list, pageable, count);  
 }  
}

## 4、mybatis查询语句

<select id="findSysDingUserList" resultType="com.duodian.admore.entity.db.admin.SysDingUser">  
 SELECT t.*\**, a.admId,b.name admName FROM sys\_ding\_user t  
 LEFT JOIN sys\_admin\_user\_ding a ON a.userid = t.userid  
 LEFT JOIN sys\_admin\_user b ON b.id = a.admId  
 <where>  
 <if test="userParam != null and userParam != ''">  
 and (t.userid = #{userParam} OR t.name LIKE CONCAT('%',#{userParam},'%' ) OR t.email LIKE CONCAT('%',#{userParam},'%' )  
 OR t.orgEmail LIKE CONCAT('%',#{userParam},'%' ) OR t.mobile LIKE CONCAT('%',#{userParam},'%' ) OR t.position LIKE CONCAT('%',#{userParam},'%' ))  
 </if>  
 </where>  
 ORDER BY t.id ASC  
 limit #{offset}, #{limit}  
</select>

# 11、取出制作DTO对象

## 1、对于数据库字段匹配的，可以直接选择

## 2、对于不匹配的使用 as 转化

<select id="findRedStartSpread" parameterType="com.duodian.admore.dao.db.redstart.query.RedStartSpreadQuery" resultType="com.duodian.admore.dao.db.redstart.bean.RedStartHistoryBean">  
 SELECT  
 k.trackId,  
 e.smallIcon,  
 e.formattedPrice,  
 e.price,  
 e.fileSizeBytes,  
 e.trackName,  
 f.name admName,  
 a.nickName userName,  
  
 DATE\_FORMAT(k.spreadDateStart, '%Y-%m-%d') AS ymd,  
 k.userId  
 FROM  
 redstart\_spread k

### 1、RedStartHistoryBean

public class RedStartHistoryBean implements Serializable {  
  
 private Long userId;  
 private String ymd;  
 private String trackId;  
 private String smallIcon;  
 private String bundleId;  
 private String formattedPrice;  
 private BigDecimal price;  
 private BigInteger fileSizeBytes;  
 private String trackName;  
  
 private String fileSizeDesc; //文件大小描述  
 private String authName;  
 private String admName;  
 private String userName;

### 2、RedStartSpreadQuery

public class RedStartSpreadQuery implements Serializable {  
  
 private Long userId;  
 private String trackId;  
 private String trackName;  
 private Date startDate;  
 private Date endDate;  
 private String name;  
 private Integer status;  
  
 private Integer offset;  
 private Integer limit;  
 private Date point;  
  
 private String userParam;  
 private String appParam;

# 12、只有一个参数传入的时候，不能直接写参数名字了 而是使用下面的\_parameter

## 1、mapepr.java

List<CustomerChance> getCustomerList(Long adminId);

## 2、mapper.xml

<select id="getCustomerList" resultType="com.duodian.admore.entity.db.customer.CustomerChance">  
 SELECT *\** FROM `crm\_customer\_chance` c  
 where c.isVisible = 1  
 <if test="\_parameter != null">  
 and c.adminId = #{\_parameter}  
 </if>  
</select>

# 13、加入原生符号

## 1、AND t.cdate &gt;= #{startDate}  </if>

[<![CDATA[]]>和转义字符](http://www.cnblogs.com/catgatp/p/6403382.html)

被<![CDATA[]]>这个标记所包含的内容将表示为**纯文本**，比如<![CDATA[<]]>表示文本内容“<”。   
　　此标记用于xml文档中，我们先来看看使用转义符的情况。我们知道，在xml中，”<”、”>”、”&”等字符是不能直接存入的，否则xml语法检查时会**报错**，如果想在xml中使用这些符号，必须将其转义为实体，如”&lt;”、”&gt;”、”&amp;”，这样才能保存进xml文档。   
　　在使用程序读取的时候，解析器会自动将这些实体转换回”<”、”>”、”&”。举个例子：   
　　<age> age < 30 </age>   
　　上面这种写法会报错，**应该这样写**：   
　　<age> age &lt; 30 </age>   
　　值得注意的是：   
　　(1)转义序列字符之间不能有空格；   
　　(2) 转义序列必须以”;”结束；   
　　(3) 单独出现的”&”不会被认为是转义的开始；   
　　(4) 区分大小写。   
　　   
　　在XML中，需要转义的字符有：   
　　(1)&　　　&amp;   
　　(2)<　　　&lt;   
　　(3)>　　　&gt;   
　　(4)＂　　　&quot;   
　　(5)＇　　　&apos;   
　　但是严格来说，在XML中只有”<”和”&”是非法的，其它三个都是可以合法存在的，但是，把它们都进行转义是一个好的习惯。   
　　不管怎么样，转义前的字符也好，转义后的字符也好，都会被xml解析器解析，**为了方便起见**，使用<![CDATA[]]>来包含不被xml解析器解析的内容。但要注意的是：   
　　(1) 此部分不能再包含”]]>”；   
　　(2) 不允许嵌套使用；   
　　(3)”]]>”这部分不能包含空格或者换行。   
　　最后，说说<![CDATA[]]>和xml转移字符的关系，它们两个看起来是不是感觉功能重复了？   
　　是的，它们的功能就是一样的，只是应用场景和需求有些不同：   
　　(1)<![CDATA[]]>不能适用所有情况，转义字符可以；   
　　(2) 对于短字符串<![CDATA[]]>写起来啰嗦，对于长字符串转义字符写起来可读性差；   
　　(3) <![CDATA[]]>表示xml解析器忽略解析，所以更快。

## 1、举例说明

<sql id=*"pageSuffix"*>

) a where rownum <![CDATA[<=]]> #{end,jdbcType=INTEGER} ) b where b.rn <![CDATA[>]]> #{start,jdbcType=INTEGER}

</sql>

# 14、一个条件匹配多个字段

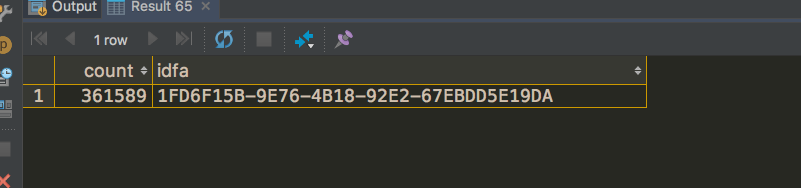


<if test="userParam != null and userParam != ''">  
 AND (t.userId = #{userParam}  
 OR a.nickName LIKE CONCAT('%',#{userParam},'%' )  
 OR b.realName LIKE CONCAT('%',#{userParam},'%' )  
 OR c.realName LIKE CONCAT('%',#{userParam},'%' )  
 OR t.customerId LIKE CONCAT('%',#{userParam},'%' )  
 OR t.customerName LIKE CONCAT('%',#{userParam},'%'))  
</if>

# 15、count详解

## 1、这样输出结果只有一行，因为count（\*）本来代表的就是一个数字，本身就是一行

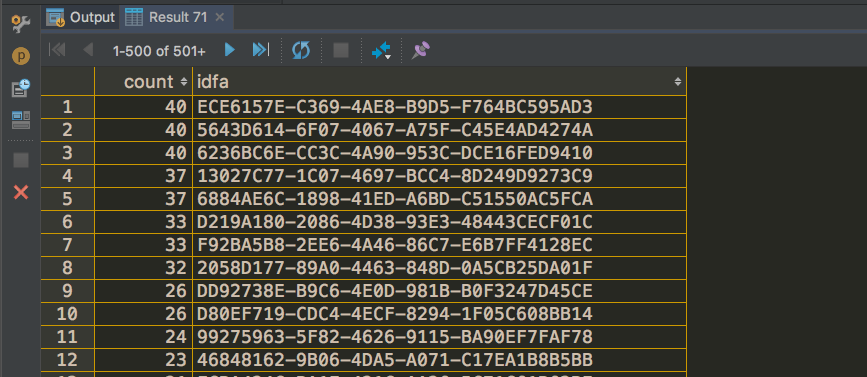
SELECT count(*\**) as "count",idfa from apps\_click\_record a; # WHERE a.keywordId = '169995' GROUP by idfa ORDER BY count(\*) DESC ;



## 2、使用group分组 （下面二者是一样的） 其实这里就表示分组之后每组的个数

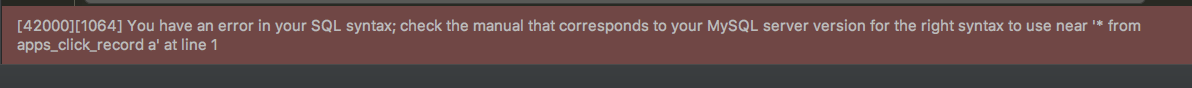
SELECT count(*\**) as "count",idfa from apps\_click\_record a WHERE a.keywordId = '169995' GROUP by idfa ORDER BY count(*\**) DESC ;

SELECT count(idfa) as "count",idfa from apps\_click\_record a WHERE a.keywordId = '169995' GROUP by idfa ORDER BY count(*\**) DESC ;



### 3、count（\*） 和 \* 的查询 是错误的

SELECT count(*\**) as "count",\* from apps\_click\_record a;



# 16、多个条件分组，通过传入的日期），以及自身的tpe，activityId进行分组

SELECT date\_format(passTime,'%Y-%m-%d'),type ,activityId, sum(ifnull(settleMoney,0)) as settleMoney ,sum(ifnull(award,0)) award FROM cpl\_step\_passed\_record  
 where date\_format(passTime,'%Y-%m-%d') = '2018-02-27'  
GROUP BY type,activityId,date\_format(passTime,'%Y-%m-%d');

# 17、日期时间段

/\*某个时间段激活的idfa \*/  
select count(idfa) from apps\_passed\_record where keywordId in (175515) and  
 date\_format(passTime,'%Y-%m-%d %H:%i') > '2018-04-22 10:30:00' and  
 date\_format(passTime,'%Y-%m-%d %H:%i') < '2018-04-22 24:00:00' ;

%H:%i:%S")

*/\*\*  
 \* 首页新闻展示查询  
 \** ***@param s*** *\** ***@return*** *\*/*@Query(value = **"select new cn.edu.dlut.career.dto.school.NewsDTO(n.id,n.title,n.publishDate) from News n where n.newsColumn = ?1 AND (n.isTop = FALSE OR (n.isTop = TRUE AND n.topEndDate <= to\_char(now(), 'YYYY-MM-DD')))\n"** +  
 **"AND n.endDate>= to\_char(now(), 'YYYY-MM-DD') order by n.publishDate DESC"**)  
LinkedList<NewsDTO> findByNewsColumn(String s, Pageable pageable);

 date类型是Oracle常用的日期型变量，他的时间间隔是秒。两个日期型相减得到是两个时间的间隔，注意单位是“天”。例如：查看一下当前距离伦敦奥运会开幕还有多长时间：

1 select to\_date('2012-7-28 03:12:00','yyyy-mm-dd hh24:mi:ss')-sysdate from dual

结果是：92.2472685185185天，然后你根据相应的时间换算你想要的间隔就行！这个结果可能对程序员有用，对于想直接看到结果的人，这个数字还不是很直观，所以，就引出了timestamp类型

timestamp是DATE类型的扩展，可以精确到小数秒（fractional\_seconds\_precision），可以是0 to9，缺省是６。两个timestamp相减的话，不能直接的得到天数书，而是得到，  
多少天，多少小时，多少秒等，例如：同样查看一下当前距离伦敦奥运会开幕还有多长时间.

1 select to\_timestamp('2012-7-28 03:12:00','yyyy-mm-dd hh24:mi:ss')-systimestamp from dual

     结果是：+000000092 05:51:24.032000000，稍加截取，就可以得到92天5小时，51分钟，24秒，这样用户看起来比较直观一些！但是这个数字对程序员来说不是很直观了，如果想要具体的时间长度的话，并且精度不要求到毫秒的话，

### 选择最近48小时候内的用户

<select id="find48HoursDistinctOpenId" resultType="com.duodian.youhui.entity.db.callperson.OnlineChatPerson">  
 <![CDATA[ SELECT *\** from call\_online\_chat\_person c WHERE TIMESTAMPDIFF(HOUR,c.cdate,now()) < 48 ]]>  
</select>  
  
<!--  
 //前面减去后面，前面为过去事件  
 select TIMESTAMPDIFF(HOUR,'2018-05-30 00:00:00',now()) as '时间差' ;  
 SECOND 秒 SECONDS  
 MINUTE 分钟 MINUTES  
 HOUR 时间 HOURS  
 DAY 天 DAYS  
 MONTH 月 MONTHS  
 YEAR 年 YEARS  
 -->

# 18、delete删除表中数据

平时delete其实用到情况不太多，所以我不小心晕菜了

1,delete from user as u where u.userid=6;  
2,delete from user u where u.userid=6;  
3,delete from user where userid=6;  
4,delete u.\* from user u where u.userid=6;  
5,delete u from user u where u.userid=6;  
  
通过测试可以发现，1和2，都是错误的MySQL语句，而3和4和5是可行的。所以，如果我们想在删除操作中使用别名的话，就要采用第4种方法或者第五种方法

# 19、请求分页参数

@GetMapping("getOriginData")  
public Wrapper<?> getOriginData(String phone, Integer bankId, PageQuery pageQuery){

@Setter  
@ApiModel("分页对象")  
@Accessors(chain = true)  
public class PageQuery {  
  
 @ApiModelProperty(value = "开始页数，从1开始",example = "1", required = true,dataType = "java.lang.Integer")  
 private Integer pageNum = 1;  
 @ApiModelProperty(value = "每页数量",example = "20", required = true,dataType = "java.lang.Integer")  
 private Integer pageSize = 20;  
 @ApiModelProperty(hidden = true)  
 @Getter  
 private int navigatePages = 8;  
  
 public Integer getPageSize() {  
 return pageSize == null ? 20 : pageSize;  
 }  
  
 public Integer getPageNum() {  
 return pageNum == null ? 1 : pageNum;  
 }  
}

# 20、timestamp 多个日期，如果可能为空，则建议使用datetime

普通字段不要设置为timestamp，timestamp列必须有默认值，默认值可以为“0000-00-00 00:00:00”，但不能为null。如果我们在save实体的时候，没有给相关timestamp设置值，那么他就会自动由mysql将当前时间设置进去， cdate和udate，都是在java基础上控制的

createTime datetime default null ,  
clickTime datetime default null,

@Temporal(TemporalType.***TIMESTAMP***)  
@ApiModelProperty(value = "创建时间")  
private Date createTime;  
  
@Temporal(TemporalType.***TIMESTAMP***)  
@ApiModelProperty(value = "点击时间")  
private Date clickTime;

cdate timestamp NOT NULL DEFAULT CURRENT\_TIMESTAMP,  
udate timestamp NOT NULL DEFAULT CURRENT\_TIMESTAMP ON UPDATE CURRENT\_TIMESTAMP,

@Temporal(TemporalType.***TIMESTAMP***)  
@Column(columnDefinition="TIMESTAMP DEFAULT CURRENT\_TIMESTAMP",insertable = true,updatable = false)  
@ApiModelProperty(hidden = true)  
private Date cdate;  
  
@UpdateTimestamp  
@Temporal(TemporalType.***TIMESTAMP***)  
@ApiModelProperty(hidden = true)  
private Date udate;

## 重点：2、@Temporal(TemporalType.DATE)插入数据库中的日期会自动变成 00.00.00，应该使用TIMESTAMP，sql中是datetime

# 21、多条件排序

举例：订单降序 1，订单升序 2 ，成交额降序 3，成交额升序 4，

<select id="findCouponTaoKeDataByParam" resultType="com.duodian.youhui.data.coupon.CouponTaoKeItemGoodSummaryData">  
 SELECT c.itemTitle,COUNT(c.itemId) as orderSize,sum(c.estimateAmount) AS sumEstimateAmount ,c.adzoneName,c.adzonePid,c.createTime,c.itemId FROM coupon\_taoke\_data c  
 <where>  
 c.dataType = 1 and c.status = 1  
 <include refid="findCouponTaoKeDataByParamSQL"></include>  
 </where>  
 GROUP by c.itemId,c.adzonePid  
  
 <if test="order != null">  
 <choose>  
 <when test="order == 1">  
 order by orderSize DESC  
 </when>  
 <when test="order == 2">  
 order by orderSize asc  
 </when>  
 <when test="order == 3">  
 order by sumEstimateAmount DESC  
 </when>  
 <when test="order == 4">  
 order by sumEstimateAmount asc  
 </when>  
 </choose>  
 </if>  
  
 <if test="offset != null and limit != ''">  
 limit #{offset}, #{limit}  
 </if>  
  
</select>

## 2、给排序添加非空条件

1、使用order byorderid desc实现降序时，orderid 为null数据的会排在数据的最后面；

但是，order by orderid升序时，orderid 为null的数据则会排在最前面，如果想要将orderid

为null的数据排在最后，就需要加上is null。

select \* from b\_programme u order by u.orderid is null, u.orderid

# 23、参数传入为0，判断null的时候

## id传值为0时(前提是id对应的类型为long 或者 Integer，String型无此问题)，发现并没有执行if里的sql，因为在mybatis中会自动把0当成‘’空字符串，所以建议以后传入这种类型的数据，最好还是不要传入0，可以将0改变为其他的数字，比如5等

List<CouponItemGood> dataByParam(@Param("status") Integer status,

<if test="status !=null and status !=''">  
 <if test="status == 5">  
 and c.connectStatus = 0 and c.createAdminId is not NULL  
 </if>  
 <if test="status == 1">  
 and c.connectStatus = 1 and c.status = 1 and c.createAdminId is not NULL  
 </if>  
 <if test="status == 2">  
 and c.connectStatus = 1 and c.status = 0 and c.createAdminId is not NULL  
 </if>  
 <if test="status == 3">  
 and c.status = 2 and c.createAdminId is not NULL  
 </if>  
 <if test="status == 4">  
 and c.createAdminId is NULL  
 </if>  
 <if test="status == 6 ">  
 and c.status not in (0) and c.zhiboStatus=1  
 </if>  
 <if test="status == 7 ">  
 and c.scheduleDealStatus = 0 and c.createAdminId is not NULL  
 </if>  
</if>

## 第二种解决方法

传递Integer类型的值，如果传递的是0，到mybatis的mapper的xml文件中是把值当作空字符串，mybatis源码对其进行了强制定义。

<if test="status != null and status !=  '' or status == 0">

使用时增加多一个or status == 0判断

# 23、查询随机数优化

## 24.1、性能比较差的一个

#### RAND()  函数返回的是一个小于1的随机数

<!--失效之后，从选品库中随便找一个 -->  
<select id="wechatSuiJiItemGoodImageUrl" resultType="java.lang.String">  
 <![CDATA[  
 SELECT c.erWeiMaInfoUrl FROM coupon\_item\_good c WHERE  
 TIMESTAMPDIFF(DAY ,cdate,now()) < 5 ORDER BY RAND() LIMIT 1  
 ]]>  
</select>

## 24.2、优化

**SQL ROUND() 语法**

SELECT ROUND(column\_name,decimals) FROM table\_name

|  |  |
| --- | --- |
| **参数** | **描述** |
| column\_name | 必需。要舍入的字段。 |
| decimals | 必需。规定要返回的小数位数。 |

<select id="findUserCouponAdzone" resultType="com.duodian.youhui.entity.db.coupon.CouponAdzone">  
 <![CDATA[  
 SELECT *\**  
 FROM `coupon\_adzone` AS t1  
 JOIN (SELECT ROUND(RAND() \* (SELECT MAX(id)  
 FROM `coupon\_adzone`)) AS id) AS t2  
 WHERE t1.id >= t2.id

and t1.status = 1 AND t1.adzoneType = 3 and (TIMESTAMPDIFF(HOUR, t1.userUseTime, now()) > #{timeDiff} OR t1.userInfoId IS NULL )  
 ORDER BY t1.id ASC  
 LIMIT 1;  
 ]]>  
</select>

# 24、exist 和in的性能

**如果查询的两个表大小相当，那么用in和exists差别不大**。

如果两个表中一个较小，一个是大表，则子查询表大的用exists，子查询表小的用in：

例如：表A（小表），表B（大表）

1：

select \* from A where cc in (select cc from B) 效率低，用到了A表上cc列的索引；

select \* from A where exists(select cc from B where cc=A.cc) 效率高，用到了B表上cc列的索引。

相反的

2：

select \* from B where cc in (select cc from A) 效率高，用到了B表上cc列的索引；

select \* from B where exists(select cc from A where cc=B.cc) 效率低，用到了A表上cc列的索引。

not in 和not exists如果查询语句使用了not in 那么内外表都进行全表扫描，没有用到索引；而not extsts 的子查询依然能用到表上的索引。**所以无论那个表大，用not exists都比not in要快**。

in 与 =的区别

select name from student where name in ('zhang','wang','li','zhao');

与

select name from student where name='zhang' or name='li' or name='wang' or name='zhao'

的结果是相同的。