2020.11.06

<if test="keyword != null and keyword != ''">

<bind name="name" value="'%'+ keyword +'%'"/>

and name like #{name}

</if>

Example example = new Example(TutorApprovalOpinion.class);  
example.orderBy("crtTime").asc();  
Example.Criteria criteria = example.createCriteria();  
criteria.andEqualTo("tenantId", tenantId);  
criteria.andEqualTo("processId", instId);

<select id="getNeedList" resultType="java.util.Map">  
 SELECT DISTINCT  
 RES.ID\_ as id,  
 RES.PROC\_INST\_ID\_ as instId,  
 DATE\_FORMAT(TAY.crt\_time,"%Y-%m-%d") as createTime,  
 TAY. NO as no,  
 TAY. NAME AS name,  
 CONCAT(TAY.degree\_type,",",TAY.about) AS identificationType,  
 TAY.application\_type as applicationType,  
 TUB.sex as sex,  
 DEF.ID\_ as defId,  
 RES.TASK\_DEF\_KEY\_ as taskDefKey,  
 RES.NAME\_ as taskName  
 FROM  
 lark\_flow.ACT\_RU\_TASK RES  
 LEFT JOIN lark\_flow.ACT\_RU\_IDENTITYLINK I ON I.TASK\_ID\_ = RES.ID\_  
 LEFT JOIN lark\_flow.ACT\_RE\_PROCDEF DEF ON DEF.ID\_ = RES.PROC\_DEF\_ID\_  
 LEFT JOIN mht\_lark\_tutor.tutor\_application\_type TAY ON TAY.process\_id = RES.PROC\_INST\_ID\_  
 LEFT JOIN mht\_lark\_tutor.teaching\_user\_base TUB ON TUB.no = TAY.no  
 WHERE  
 RES.SUSPENSION\_STATE\_ = 1  
 AND RES.TENANT\_ID\_ = #{tenantId}  
 AND (  
 RES.ASSIGNEE\_ = #{userId}  
 OR (  
 RES.ASSIGNEE\_ IS NULL  
 AND I.TYPE\_ = 'candidate'  
 AND (  
 I.USER\_ID\_ = #{userId}  
 OR (  
 I.GROUP\_ID\_ IN  
 (  
 SELECT CONCAT('dept\_',depart\_id) as depart\_id  
 FROM lark\_admin.base\_user  
 WHERE id = #{userId}  
 )  
 AND if(RES.TASK\_DEF\_KEY\_=#{company},I.GROUP\_ID\_=(SELECT CONCAT('dept\_',id) as id FROM lark\_admin.base\_depart WHERE code=TAY.company),true)  
 And if(RES.TASK\_DEF\_KEY\_=#{trainingCompany} ,I.GROUP\_ID\_=(SELECT CONCAT('dept\_',id) as id FROM lark\_admin.base\_depart WHERE code=TAY.training\_company),true)  
 )  
 )  
 )  
 )  
 <if test="no != null and no != ''">  
 AND TAY. no = #{no}  
 </if>  
 <if test="applicationType != null and applicationType != ''">  
 AND TAY.application\_type = #{applicationType}  
 </if>  
 <if test="year != null and year != ''">  
 AND TAY.year = #{year}  
 </if>  
 <if test="taskDefKey != null and taskDefKey != ''">  
 AND RES.TASK\_DEF\_KEY\_ = #{taskDefKey}  
 </if>  
 and DEF.CATEGORY\_=#{category}  
 ORDER BY  
 RES.CREATE\_TIME\_ DESC  
</select>

Stream

passList=passList.stream().sorted((m1,m2)->{  
 Object s1 = m1.get(DynamicFieldEnum.*SORT*.getCode());  
 Object s2 = m2.get(DynamicFieldEnum.*SORT*.getCode());  
 if (s1==null || s2==null) return 0;  
 return ((Integer)s1).compareTo((Integer)s2);  
 }).collect(Collectors.*toList*());

Map<String, List<AwardsResult>> ballotId2ResultMap = awardsResults  
 .stream()  
 .collect(Collectors.*groupingBy*(AwardsResult::getSlimId));

请假

<select id="getNeedList" resultType="java.util.Map">  
 SELECT DISTINCT  
 RES.ID\_ as id,  
 RES.PROC\_INST\_ID\_ as instId,  
 DATE\_FORMAT(LBI.crt\_time,"%Y-%m-%d %h:%i:%s") as createTime,  
 LBI. NO as no,  
 LBI. NAME AS name,  
 LBI.application\_type as applicationType,  
 DATE\_FORMAT(LBI.start\_date,"%Y-%m-%d %h:%i:%s") as startDate,  
 DATE\_FORMAT(LBI.end\_date,"%Y-%m-%d %h:%i:%s") as endDate,  
 LBI.process\_state as processState,  
 LBI.total\_days as totalDays,  
 LBI.explain\_info as explainInfo,  
 DEF.ID\_ as defId,  
 RES.TASK\_DEF\_KEY\_ as taskDefKey,  
 RES.NAME\_ as taskName  
 FROM  
 lark\_flow.ACT\_RU\_TASK RES  
 LEFT JOIN lark\_flow.ACT\_RU\_IDENTITYLINK I ON I.TASK\_ID\_ = RES.ID\_  
 LEFT JOIN lark\_flow.ACT\_RE\_PROCDEF DEF ON DEF.ID\_ = RES.PROC\_DEF\_ID\_  
 LEFT JOIN mht\_lark\_leave.leave\_base\_info LBI ON LBI.process\_id = RES.PROC\_INST\_ID\_  
 LEFT JOIN lark\_admin.base\_user BU ON BU.id = LBI.user\_id  
 WHERE  
 RES.SUSPENSION\_STATE\_ = 1  
 AND RES.TENANT\_ID\_ = #{tenantId}  
 AND (  
 RES.ASSIGNEE\_ = #{userId}  
 OR (  
 RES.ASSIGNEE\_ IS NULL  
 AND I.TYPE\_ = 'candidate'  
 AND (  
 I.USER\_ID\_ = #{userId}  
 OR (  
 I.GROUP\_ID\_ IN  
 (  
 select group\_id from lark\_admin.base\_group\_member where user\_id=#{userId}  
 )  
 AND if(RES.TASK\_DEF\_KEY\_=#{adviser},(CONCAT(#{prefixCollege},  
 <include refid="getCollegeCode" />  
 )=(SELECT t1.code FROM lark\_admin.base\_depart t1,lark\_admin.base\_user t2 WHERE t2.depart\_id=t1.id and t2.id=#{userId})),true)  
 And if(RES.TASK\_DEF\_KEY\_=#{charge} ,(CONCAT(#{prefixCollege},  
 <include refid="getCollegeCode" />  
 )=(SELECT t1.code FROM lark\_admin.base\_depart t1,lark\_admin.base\_user t2 WHERE t2.depart\_id=t1.id and t2.id=#{userId})),true)  
 And if(RES.TASK\_DEF\_KEY\_=#{tutor},(select `zgh` from lark\_admin.yjs\_xh\_ds where xh=LBI.no)=(select `no` from lark\_admin.base\_user where id=#{userId}),true)  
 )  
 )  
 )  
 )  
 <if test="no != null and no != ''">  
 AND LBI.no = #{no}  
 </if>  
 <if test="applicationType != null and applicationType != ''">  
 AND LBI.application\_type = #{applicationType}  
 </if>  
 <if test="taskDefKey != null and taskDefKey != ''">  
 AND RES.TASK\_DEF\_KEY\_ = #{taskDefKey}  
 </if>  
 <if test="startDate != null and startDate != ''">  
 AND LBI.start\_Date > #{startDate}  
 </if>  
 <if test="endDate != null and endDate != ''">  
 AND #{endDate}> LBI.end\_Date  
 </if>  
 and DEF.CATEGORY\_=#{category}  
 ORDER BY  
 RES.CREATE\_TIME\_ DESC  
</select>

<sql id="getCollegeCode">  
 (SELECT  
 code  
 FROM  
 lark\_admin.base\_depart  
 WHERE  
 FIND\_IN\_SET(  
 id,  
 (  
 SELECT  
 CONCAT(departs.id,',',GROUP\_CONCAT(departs.parent\_ids))  
 FROM  
 lark\_admin.base\_depart\_user departUser  
 LEFT JOIN lark\_admin.base\_depart departs ON departUser.depart\_id = departs.id  
 WHERE  
 departUser.user\_id = LBI.user\_id  
 AND departUser.tenant\_id = #{tenantId}  
 )  
 ) and `type`=#{college} limit 1)  
</sql>

SELECT  
 substring\_index( d.`code`, '\_', - 1 )  
FROM  
 lark\_admin.base\_depart d  
 LEFT JOIN lark\_admin.`base\_depart\_user` du ON d.id = du.depart\_id  
 LEFT JOIN lark\_admin.base\_user u ON du.user\_id = u.id  
WHERE  
 u.id = #{userId}  
)

SELECT  
 `code`  
FROM  
 leave\_role\_code\_rt  
WHERE  
 role\_code IN (  
 <foreach collection="roldCodes" separator="," item="item">  
 #{item}  
 </foreach>  
 )