

In查询

```
1 List<Predicate> predicates = new ArrayList<>();
2
3 if(StringUtils.isNotBlank(keyword)){
4     Predicate condition = cb.like(root.get("title"), "%" + keyword + "%");
5     predicates.add(condition);
6 }
7 if(nodeId != null){
8     Predicate condition = cb.equal(root.get("node").get("id"), nodeId);
9     predicates.add(condition);
10 }
11 if(status != null && status.size() > 0){
12     Path<Object> path = root.get("status");
13     CriteriaBuilder.In<Object> in = cb.in(path);
14     for(InfoStatusEnum infoStatusEnum : status){
15         in.value(infoStatusEnum);
16     }
17     predicates.add(cb.and(in));
18 }
19
20 if(predicates.size() == 0) {
21     return cb.and();
22 }
23
24 Object[] list = predicates.toArray();
25 Predicate[] t = new Predicate[predicates.size()];
26 Predicate[] result = predicates.toArray(t);
27 return cb.and(result);
```

组合And、Or查询

```
1 Page<Charge> result = chargeRepository.findAll(new Specification<Charge>() {
2     @Override
3     public Predicate toPredicate(Root<Charge> root, CriteriaQuery<?> query, CriteriaBuilder cb) {
4         List<Predicate> predicates = new ArrayList<>();
5
6         Predicate deptPredicate = cb.equal(root.get("contract").get("dept").get("id"), OptUserIdThre
7
8         if(StringUtils.isNotBlank(searchKey)){
9             Predicate condition = cb.and(
10                 cb.or(
11                     cb.like(root.get("contract").get("tenantShopName"), "%"+searchKey+"%"),
12                     cb.like(root.get("contract").get("tenantName"), "%"+searchKey+"%"),
13                     cb.like(root.get("contract").get("code"), "%"+searchKey+"%")
14                 ),
15                 deptPredicate);
16             predicates.add(condition);
17         }
18
19         if(StringUtils.isNotBlank(shopMall)){
```

```

20     Predicate condition = cb.and(cb.like(root.get("contract").get("shopMallName"), "%"+shop
21     predicates.add(condition);
22 }
23
24 if(StringUtils.isNotBlank(receiveDate)){
25     SimpleDateFormat sdf = new SimpleDateFormat("yyyy-MM-dd");
26     Date d = null;
27     try {
28         d = sdf.parse(receiveDate);
29         Date s = new Date(DateUtil.getDayBegin(d) - 1);
30         Date e = new Date(DateUtil.getDayEnd(d));
31
32         Predicate condition1 = cb.and(cb.greaterThan(root.get("chargeTime"), s), deptPredic
33         predicates.add(condition1);
34
35         Predicate condition2 = cb.and(cb.lessThan(root.get("chargeTime"), e), deptPredicate
36         predicates.add(condition2);
37
38     } catch (ParseException e) {
39         e.printStackTrace();
40     }
41 }
42
43 if(receivablesTypeEnum != null){
44     Predicate condition = cb.and(cb.equal(root.get("typeEnum"), receivablesTypeEnum), deptF
45     predicates.add(condition);
46 }
47
48 if(predicates.size() == 0) {
49     return cb.and(deptPredicate);
50 }
51
52 Object[] list = predicates.toArray();
53 Predicate[] t = new Predicate[predicates.size()];
54 Predicate[] result = predicates.toArray(t);
55 return cb.and(result);
56 }
57 }, PageConver.my2Spring(page,new MySort()));

```