


mybatis plus 条件构造器queryWrapper学习

原创bird_tp于 2020-04-17 20:24:32 发布124822收藏622

分类专栏:mybatis plus文章标签:mybatisplus条件查询queryWrapper

mybatis plus 专栏收录该内容

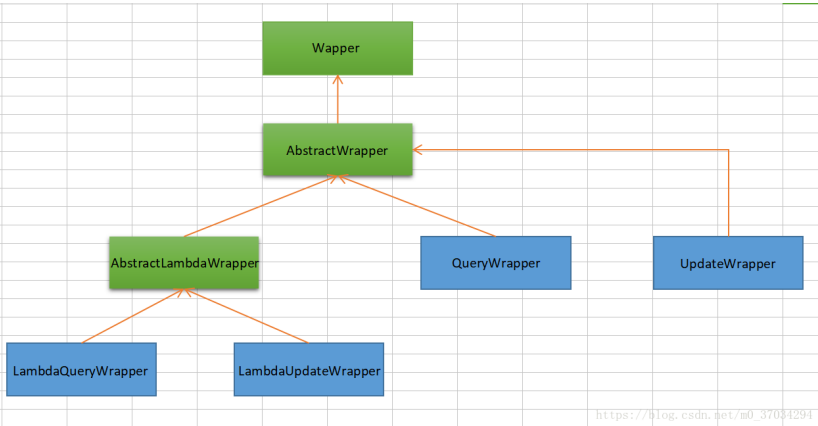
7 订阅5 篇文章

订阅专栏

首先奉上 mybatis plus 的官方文档链接，再看本博客有不清晰的地方，可以查看文档 <https://mp.baomidou.com/guide/>

一、queryWrapper是什么

queryWrapper是mybatis plus中实现查询的对象封装操作类，他的层级关系如下



在上面的图片中，

- Wrapper**：条件构造抽象类，最顶端父类，抽象类中提供4个方法西面贴源码展示
- AbstractWrapper**：用于查询条件封装，生成 sql 的 where 条件
- AbstractLambdaWrapper**：Lambda 语法使用 Wrapper统一处理解析 lambda 获取 column。
- LambdaQueryWrapper**：看名称也能明白就是用于Lambda语法使用的查询Wrapper
- LambdaUpdateWrapper**：Lambda 更新封装Wrapper
- QueryWrapper**：Entity 对象封装操作类，不是用lambda语法
- UpdateWrapper**：Update 条件封装，用于Entity对象更新操作

二、使用demo

```
使用的关键的代码
QueryWrapper<PbListBlack> sectionQueryWrapper = new QueryWrapper<>();
sectionQueryWrapper.eq("OPTYPE", 1);
sectionQueryWrapper.eq("BLTYPE", 1);
List<PbListBlack> pbListBlacks =
ipbListBlackMapper.selectList(sectionQueryWrapper);
```

上面这段代码的意思就是，首先新建一个QueryWrapper对象，类型为PbListBlack对象，也就是你需要查询的实体数据，


```
sectionQueryWrapper.eq("OPTYPE", 1);

sectionQueryWrapper.eq("BLTYPE", 1);
```

这两句的意思是PbListBlack对象对应的数据库表中的OPTYPE，BLTYPE字段值要为1

然后调用ipbListBlackMapper.selectList方法，入参就为前面新建好的查询对象封装类

下面贴上查询实体的代码

bird_tp

关注

```

@Data
@EqualsAndHashCode(callSuper = false)
@Accessors(chain = true)
@TableName("PB_LIST_BLACK")
@ApiModel(value = "PB_LIST_BLACK对象", description = "")
public class PbListBlack extends Model<PbListBlack> {
    @TableField("ID")
    private Long id;

    @TableField("USERID")
    private String userId;
    @TableField("SPGATE")
    private String spgate;
    @TableField("SPNUMBER")
    private String spnumber;
    @TableField("PHONE")
    private Long phone;
    @TableField("OPTYPE")
    private Integer optype;
    @TableField("OPTTIME")
    private Timestamp optTime;
    @TableField("CORPCODE")
    private String corpCode;
    @TableField("SVRTYPE")
    private String svrType;

    @TableField("MSG")
    private String msg = " ";

    @ApiModelProperty(value = "黑名单类型。1: 短信; 2: 彩信")
    @TableField("BLTYPE")
    private Integer blType;

}

```

然后是IPbListBlackMapper的代码，代码很简单

```

@Repository
public interface IPbListBlackMapper extends BaseMapper<PbListBlack> {
}

```

仅仅只需要基层BaseMapper接口即可，这样mybatis plus底层封装的方法即可实现帮你查询你设置查询条件查询到的数据

当你的Mapper 继承BaseMapper接口后，无需编写 mapper.xml 文件，即可获得CRUD功能



bird_tp

关注



151



23



622



(

三、QueryWrapper的方法

函数名	说明	说明/例子
eq	等于=	例: eq("name", "老王")-->name = '老王'
ne	不等于<>	例: ne("name", "老王")-->name <> '老王'
gt	大于>	例: gt("age", 18)-->age > 18
ge	大于等于>=	例: ge("age", 18)-->age >= 18
lt	小于<	例: lt("age", 18)-->age < 18
le	小于等于<=	例: le("age", 18)-->age <= 18
between	BETWEEN 值1 AND 值2	例: between("age", 18, 30)-->age between 18 and 30
notBetween	NOT BETWEEN 值1 AND 值2	例: notBetween("age", 18, 30)-->age not between 18 and 30
like	LIKE '值'	例: like("name", "王")-->name like '王%'
notLike	NOT LIKE '值'	例: notLike("name", "王")-->name not like '王%'
likeLeft	LIKE '值'	例: likeLeft("name", "王")-->name like '王'
likeRight	LIKE '值'	例: likeRight("name", "王")-->name like '王'
isNull	字段 IS NULL	例: isNull("name")-->name is null
isNotNull	字段 IS NOT NULL	例: isNotNull("name")-->name is not null
in	字段 IN (v0, v1, ...)	例: in("age", {1,2,3})-->age in (1,2,3)
notIn	字段 NOT IN (v0, v1, ...)	例: notIn("age", 1, 2, 3)-->age not in (1,2,3)
inSql	字段 IN (sql语句)	inSql("id", "select id from table where id < 3") -->id in (select id from table where id < 3)
notInSql	字段 NOT IN (sql语句)	notInSql("id", "select id from table where id < 3") -->age not in (select id from table where id < 3)
groupBy	分组: GROUP BY 字段, ...	例: groupBy("id", "name")-->group by id,name
orderByAsc	排序: ORDER BY 字段, ... ASC	例: orderByAsc("id", "name")-->order by id ASC,name ASC
orderByDesc	排序: ORDER BY 字段, ... DESC	例: orderByDesc("id", "name")-->order by id DESC,name DESC
orderBy	排序: ORDER BY 字段, ...	例: orderBy(true, true, "id", "name") -->order by id ASC,name ASC
having	HAVING (sql语句)	having("sum(age) > {0}", 11)-->having sum(age) > 11
or	拼接 OR	注意事项: 主动调用or表示紧接着下一个方法不是用and连接! (不调用or则默认为使用and连接) 例: eq("id", 1).or().eq("name", "老王")-->id = 1 or name = '老王' 例: and(i -> i.eq("name", "李白").ne("status", "活着")) -->and (name = '李白' and status <> '活着')
apply	拼接 sql	注意事项: 该方法可用于数据库函数 动态入参的params对应前面sqlHaving内部的{index}部分. 这样是不会有sql注入风险的, 反之会有! 例: apply("date_format(dateColumn, '%Y-%m-%d') = {0}", "2008-08-08")-->date_format(dateColumn, '%Y-%m-%d') = '2008-08-08')
last	无视优化规则直接拼接到 sql 的最后	注意事项: 只能调用一次, 多次调用以最后一次为准 有sql注入的风险, 请谨慎使用 例: last("limit 1") 例: exists("select id from table where age = 1") -->exists (select id from table where age = 1)
exists	拼接 EXISTS (sql语句)	例: notExists("select id from table where age = 1") -->not exists (select id from table where age = 1)
notExists	拼接 NOT EXISTS (sql语句)	正常嵌套 不带 AND 或者 OR 例: nested(i -> i.eq("name", "李白").ne("status", "活着")) -->(name = '李白' and status <> '活着')
nested	正常嵌套 不带 AND 或者 OR	

四、总结

上面的博客内容虽然简单，也只展示了一个查询的功能。在你实际使用的时候，可以按照 条件构造器上面包含的的方法，设置你查询数据的条件，调用BaseMapper里相应的方法即可

mybatis-plus QueryWrapper自定义查询条件的实现08-24

主要介绍了mybatis-plus QueryWrapper自定义查询条件的实现，文中通过示例代码介绍的非常详...

mybatis plus条件构造器queryWrapper、updateWrapper09-07

主要介绍了mybatis plus条件构造器queryWrapper、updateWrapper，文中通过示例代码介绍的非...

评论 23

- 请发表有价值的评论， 博客评论不欢迎灌水，良好的社区氛围需大家一起维护。 评论
- @2206 2022.03.27
很清晰
- nicedeshan 2022.03.23
写的很不错,就喜欢这样简单明了的
- 这个名没用过 2022.03.11
感谢博主，对新人很友好！
- <

1

2

3

4

>

MybatisPlus的UpdateWrapper和QueryWrapper的区别_柯柯...3-22

@TestpublicvoidtestWrapper(){LambdaQueryWrapper<Picture>lambdQueryWrapper=newLamb...

mybatis-plus 自带QueryWrapper自定义sql实现复杂查询...3-21

常用写法 xxxDO=实体类, 相当于: select name,

QueryWrapper高级使用示例

bird_tp

关注

151

23

622