* example用来存放一些去重，排序，分类，分页等信息；
* criteria用来传字段参数。

mapper接口中的函数及方法：

|  |  |
| --- | --- |
| 方法 | 功能说明 |
| int countByExample(UserExample example) thorws SQLException | 按条件计数 |
| int deleteByPrimaryKey(Integer id) thorws SQLException | 按主键删除 |
| int deleteByExample(UserExample example) thorws SQLException | 按条件查询 |
| String/Integer insert(User record) thorws SQLException | 插入数据（返回值为ID） |
| User selectByPrimaryKey(Integer id) thorws SQLException | 按主键查询 |
| ListselectByExample(UserExample example) thorws SQLException | 按条件查询 |
| ListselectByExampleWithBLOGs(UserExample example) thorws SQLException | 按条件查询（包括BLOB字段）。只有当数据表中的字段类型有为二进制的才会产生 |
| int updateByPrimaryKey(User record) thorws SQLException | 按主键更新 |
| int updateByPrimaryKeySelective(User record) thorws SQLException | 按主键更新值不为null的字段 |
| int updateByExample(User record, UserExample example) thorws SQLException | 按条件更新 |
| int updateByExampleSelective(User record, UserExample example) thorws SQLException | 按条件更新值不为null的字段 |

### 1、什么是Example类

xxxExample example = new xxxExample();

Criteria criteria= new Example().createCriteria();



mybatis-generator会为每个字段产生如上的Criterion，如果表的字段比较多，产生的Example类会十分庞大。理论上通过example类可以构造你想到的任何的筛选条件。在mybatis-generator中加以配置，配置数据表的生成操作就可以自动生层example了。

### 2、了解example成员变量

// 升序还是降序，参数格式：字段+空格+asc(desc)

protected String orderByClause;

// 取出重复，true是选择不重复记录

protected boolean distinct;

// 自定义查询条件，Criteria的集合，集合中对象是由or连接

protected List<Criteria> oredCriteria;

// 内部类Criteria包含一个Criterion的集合（Criteria是Criterion的复数）

// 每一个Criteria对象内包含的Criterion之间是由AND连接的。

public static class Criteria extends GeneratedCriteria {

protected Criteria() {  
 super();  
 }  
}

// 是mybatis逆向工程中的代码模型



// 是最基本的，最底层的Where条件，用于字段级的筛选



### 3、example使用前的准备

比如我的example是根据user表生成的，UserMapper属于dao层，UserMapper.xml是对应的映射文件。

UserMapper接口：

long countByExample(CompetingStoreExample example);

List<CompetingStore> selectByExample(CompetingStoreExample example);

UserServiceImpl实现类：

UserExample example = new UserExample();

UserExample.Criteria criteria = example.createCriteria();

### 4、查询用户数量

long count = UserMapper.countByExample(example);

类似于：select count(\*) from user

### 5、where条件查询或多条件查询

example.setOrderByClause("age asc");//升序

     example.setDistinct(false);//不去重

     if(!StringUtils.isNotBlank(user.getName())){

          criteria.andNameEqualTo(user.getName());

     }

     if(!StringUtils.isNotBlank(user.getSex())){

          criteria.andSexEqualTo(user.getSex());

     }

     List<User> userList=userMapper.selectByExample(example);

类似于：select \* from user where name={#user.name} and sex={#user.sex} order by age asc;

UserExample.Criteria criteria1 = example.createCriteria();

UserExample.Criteria criteria2 = example.createCriteria();

     if(!StringUtils.isNotBlank(user.getName())){

          Criteria1.andNameEqualTo(user.getName());

     }

     if(!StringUtils.isNotBlank(user.getSex())){

          Criteria2.andSexEqualTo(user.getSex());

     }

     Example.or(criteria2);

     List<User> userList=userMapper.selectByExample(example);

类似于：select \* from user where name={#user.name} or sex={#user.sex} ;

### 6、模糊查询

      if(!StringUtils.isNotBlank(user.getName())){

           criteria.andNameLIke(‘%’+name+’%’);

      }

      List<User>  userList=userMapper.selectByExample(example);

类似于：select \* from user where name like %{#user.name}%

### 7、分页查询

        int start = (currentPage - 1) \* rows;

        //分页查询中的一页数量

        example.setPageSize(rows);

        //开始查询的位置

        example.setStartRow(start);

        List<User> userList=userMapper.selectByExample(example);

类似于：select \* from user limit start to rows