

`select distinct Test from Table :` 可以查询出唯一的数据, 去除重复数据

`select * from people where peopleId in (select peopleId from people group by peopleId having count(peopleId) > 1):` 查询重复数据

`select * from driver_cert2 limit ?,? :` 表示: 从第几条数据开始, 查询多少记录?

```
SELECT Persons.LastName, Persons.FirstName, Orders.OrderNo
FROM Persons
INNER JOIN Orders
ON Persons.Id_P=Orders.Id_P
ORDER BY Persons.LastName
```

 使用inner关键字联合查询两表中相同的数据

mybatis关联查询:

使用collection关键字

```
<resultMap type="com.zyb.pojo.User" id="userResult">
  <id column="id" property="id" />
  <result column="name" property="name" />
  <result column="username" property="username" />
  <result column="password" property="password" />
  <result column="img" property="image" />
  <result column="sex" property="sex" />
  <result column="saying" property="saying" />
  <result column="age" property="age" />
  <collection ofType="com.zyb.pojo.UserImg" property="imgs">
    <id column="i_id" property="id" /><!-- column表示的下面用来标识的别名, property表示对应的pojo的属性 -->
    <result column="i_img" property="img" />
    <result column="i_user_id" property="user_id" />
  </collection>
</resultMap>
```

```
<select id="findUserById" resultMap="userResult"
  parameterType="int">
  select u.id,u.name,u.username,u.password,u.img,u.sex,
  u.saying,u.age, m.id as i_id, m.img as i_img,m.user_id as i_user_id
  from user u LEFT JOIN user_img
  m on u.id = m.user_id WHERE u.id =
  #{id};
</select>
```

此时一定要把对应的别名跟属性写出来, 不可用*代替。不然无法关联出相应信息。

一个用户可以对应多张图片, 在这条语句中, 我们给user命名为u。这是第一种方式, 还可以再写一个select, 利用<collection>中的select标签。

在<collection>里面, 要把相同的数据, 例如id, 要指定别名, 不然相同的id标识会与主表的字段重复, 在标签里面, column属性并不是指数据库对应的列, 二是一个别名, 要不同于主表的名称。

- JOIN: 如果表中有至少一个匹配, 则返回行
- LEFT JOIN: 即使右表中没有匹配, 也从左表返回所有的行
- RIGHT JOIN: 即使左表中没有匹配, 也从右表返回所有的行
- FULL JOIN: 只要其中一个表中存在匹配, 就返回行
- 关于insert等语句传入参数的问题。

传入参数的问题：如果要往mybatis中传入多个不同类型的参数，除了把参数包装成一个对象，使用map集合外，还可以使用@Param注解，绑定对应参数，此时就不使用parameterType引入

代码如下：

```
//这应该是一种留言，允许外来访客进行留言
public void insertRemark(@Param(value = "blog_id") int blog_id,@Param(value = "remark") String remark);

<insert id="insertRemark">
insert into remark(blog_id,remark) values(#{blog_id},#{remark});
</insert>
```

UUID

UUID 含义是通用唯一识别码 (Universally Unique Identifier)，这是一个软件建构的标准。

UUID保证对在同一时空中的所有机器都是唯一的。通常平台会提供生成的API。很适合用来做数据库的主键。

left join、inner join中的on与where的区别

其中，on是进行两表之间的选择判断，where是进行主表的选择判断。如果在on用主表的字段判断，则此选择判断将不起作用。ON影响的时候特定选择的表，而where影响的是整体。

join执行分4个步骤：

- 1、先对两个表执行交叉连接(笛卡尔积)
- 2、应用on筛选器
- 3、添加外部行
- 4、应用where筛选器

mybatis模糊查询

在配置文件中编写sql时，为了防止混淆常用<code>>和<code><来代替sql语句中大于号和小于号。

1. 使用 `${...}` 代替 `#{...}`，例如 `SELECT * FROM tableName WHERE name LIKE '%${text}%'`;

2. 在java程序中拼接

3. concat的使用。例如：

```
SELECT * FROM tableName WHERE name LIKE CONCAT(CONCAT('%', #{text}), '%');
```

```
SELECT * FROM User WHERE name LIKE CONCAT('%', #{name}, '%')。
```

4. bond标签：

例如：<code><select

```
id="selectBykeyWord" parameterType="string" resultType="com.why.mybatis.entity.RoleEntity">
    <bind name="pattern" value="'%' + keyword + '%'" />
    SELECT * FROM t_role WHERE role_name LIKE #{pattern} OR id LIKE #{pattern} OR
role_type like #{pattern}
</select>
```

bind标签的两个属性都是不选项，name为绑定到上下文的变量名，value为OGNL表达式，创建一个bind标签后，就可以在下面直接使用了。使用bind拼接字符串不仅可以避免因更换数据库而修改SQL，也能预防SQL注入。