**SQL查询强化**

1. **操作符**
2. **比较操作符**

相等 =

不相等 ！= or <>

小于和大于 < or >

比较操作符的组合 <= or >=

例如：

select \* from students where s\_sex='2 and s\_age>18;

select \* from students where s\_sex='2' and s\_age<>18;

1. **逻辑操作符**
2. IS NULL用于NULL值进行比较

例如：select \* from students where s\_name is null;

1. IS NOT NULL 用于非空比较

例如：select \* from students where s\_name is not null;

1. BETWEEN定位最大值与最小值之间

语法：列名 BETWEEN 最小值 and 最大值

例如：select \* from students where s\_age between 18 and 23;

与此类似：select \* from students where s\_age>= 18 and s\_age<= 23;

注意：between包括最大值和最小值。

4．IN用于把一个值与一个指定列表进行比较，当被比较的值至少与列表中的一个值相匹配时返回true。

例如：select \* from students where s\_age in(19,22,23);

5．LIKE 用于通配符把一个值与类似的值进行比较， 百分号：（%）没有字数限制 下划线 (\_)，一个下划线表示一个字符。

例如：

select \* from students where s\_name like '%小\_';

select \* from students where s\_name like '小\_';

select \* from students where s\_name like '小';

select \* from students where s\_name not like '\_黝\_';

6．EXISTS用于搜索指定表里是否存在满足特定条件的记录。

例如：

select \* from students where exists(select \* from students t where s\_sex=2);//此条查询与下面这条数据查询出来的数据是不一样的

select \* from students s where exists(select \* from students t where s.s\_sex=2);

注意：如果不给查询的表起别名限制，则用exists查询出来的数据是没有逻辑可言的，而使用了表的别名进行限制exists括号中结果集的条件字段就会有逻辑可言。

**3、ALL、SOME、和ANY操作符**

**All**: 关键字必须接在一个比较操作符的后面，表示与子查询返回的所有值比较为 TRUE ，则返回 TRUE 。

例如：select s\_name,s\_age from students where s\_age>=all(select s\_age from students); (所有查询结果必须大于等于**all**指定结果集的最大值)

**ANY**：关键字必须接在一个比较操作符的后面，表示与子查询返回的任何值一个比较为TRUE，则返回 TRUE 。

例如：

select s\_name,s\_age from students where s\_age>=any(select s\_age from students); (大于等于**any**指定结果集的最小值)

**SOME**: ANY的别名，较少使用。

例如：

select s\_name,s\_age from students where s\_age<=some(select s\_age from students); (小于等于**some**指定结果集的最大值)

**4、链接操作符**

AND左右的条件都为真是返回真(与)

例如：select \* from students where s\_age<19 and s\_age>22;

OR 一个为真就返回真(或)

例如：select \* from students where s\_age<19 or s\_age>22;

**5、求反操作符**

不相等

NOT BETWEEN

例如：select \* from students where age not between 19 and 22;

NOT IN

例如：select \* from students where s\_age not in(19,22,23);

NOT LIKE

例如：select \* from students where s\_name not like '小\_';

IS NOT NULL

例如：select \* from students where s\_name is not null;

NOT EXISTS

例如：

select \* from students s where not exists(select \* from students t where s.s\_sex=1);

**6、算术操作符 a+b a-b a\*b a/b a%b**

加法、减法、乘法、除法、算术操作操作符的组合

例如：select 5-10-5+22%2+10/2;

**7、关键from,where**

from子句：指定查询数据的表

Where（条件查询）：用于给查询添加条件，从而去除用户不需要的数据。

Where后边跟的**条件值**只有**true或false**，从而限制我们查询中获取的数据。

例如：

select \* from students where false=false;//此两条查询语句查询的结果是不一样的

select \* from students where false;

**8、关键字-order by**

order by 排序，默认为升序（ASC） 降序(DESC)。

语法：

select \* from 表名 [where 条件] order by [asc | desc]

例如：

select \* from students s where 1=1 ORDER BY s.s\_age;

select \* from students s ORDER BY s.s\_age;

select \* from students s where s.s\_age>18 ORDER BY s.s\_age DESC;

注意：SQL排序是基于字符的ASCII码进行排序，数字0-9会按其字符值进行排序，并且位于A-Z之前。

**9、关键字-limit**

limit 限制结果条数

**分页的做法：每页的起始数据位置=（页数page-1）\*每页的显示数目pageSize。**

如：（2-1）\*10=0就是表示第二页的第一条数据的位置

语法：select \* from 表名 [where 条件] limit 起始值,要取的条数

例如：

select \* from students s limit 1,2;//从第二条开始取两条数据。

select \* from students order by s\_age desc limit 6,2;//从第七条数据开始取两条数据

注意：数据的起始位置是从0开始的。

**10、关键字-group by**

group by 分组

与select语句配合使用，把相同的数据划分为组。在select语句里，group by子句在where条件之后，在order by子句之前。

语法：

Select 列名1,列名2 from 表名 where 条件 group by 列名1,列名2 order by 列名1,列名2;

注意：被选中的字段（查询中select之后的字段列表）必须与group by里的字段一样 (不一样虽在mysql中不报错，但是查询的数据是有误的，在oracle、sqlserver中报错)，

avg，max，min，sum和count 函数不包括在内。

例如：

不正确：

select s\_name,s\_sex,count(s\_sex) from students group by s\_sex;//按照性别分组进行统计每组的人数，但是此查询的数据不正确

正确：

select s\_name,s\_sex,count(s\_sex) from students group by s\_sex,s\_name;

select s\_age,count(s\_age) from students group by s\_age order by s\_age;

select s\_name,s\_age,s\_sex,count(s\_age)from students where s\_height='185' group by s\_age,s\_name,s\_sex order by s\_age;

**11、关键字-having**

having 筛选

该子句在select 语句里与group by子句联合使用时，用于告诉group by 子句在输出里包含那几个分组。having 对于group by 的作用相当于where对于select的作用。

语法：

select 列名1,列名2 from 表名

where 条件

group by 列名1,列名2

Having 条件

order by 列名1,列名2

**Having 子句必须跟在group by子句之后，在order by 子句之前。**

例如：

select s\_name,s\_sex,s\_age,count(s\_sex) from students group by s\_sex,s\_name,s\_age having s\_age>=19;

select s\_name,s\_sex,s\_age from students group by s\_sex,s\_name,s\_age having s\_age>18 order by s\_age desc;