# 三 查询详解

## 1 排序order by

默认查询的结果的排列顺序是按照最初表中的顺序排列的，如果要实现按照自己的需要排列，使用 order by。

select \* from 表名 order by 列1 asc,列2desc,...

asc 默认，从小到大排列，即升序；

desc 从大到小排序，即降序。

若存在多个字段，列1 列2，则按照列1排序后，值相同的部分按照列2再次排序。

查询男生学生信息，按学号降序：

select \* from students where gender=1 order by id desc;

## 2 五个聚合函数

聚合的目的是为了统计，但是聚合后只有结果没有以前的数据集合了。

count(\*)表示计算总行数，括号中写星与列名，结果是相同的

select count(\*) from students; 查询学生总数

select count(\*) number from students; 学生总数赋值给number

max(列)表示求此列的最大值

select max(id) from students where gender=0; 查询女生的编号最大值

min(列)表示求此列的最小值

select min(id) from students where isdelete=0; 查询未删除的学生最小编号

sum(列)表示求此列的和

select sum(id) from students where gender=1; 查询男生的编号之和

avg(列)表示求此列的平均值

select avg(id) from students where isdelete=0 and gender=0;查询未删女生的编号平均值

注意：如果表中没有数据记录，count返回0，其他函数返回null。

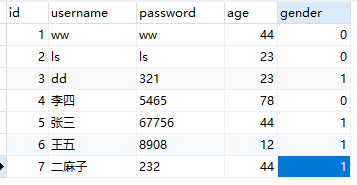
## 3 分组group by

在具体使用统计函数时，都是针对表中所有记录或者指定条件（where）的数据记录进行统计。但是实际应用中，经常会把所有数据进行分组，对分组后的数据进行统计。

按照字段分组，表示此字段相同的数据会被放到一个组中。分组后，只能查询出相同的数据列，一般会对分组后的数据进行统计，做聚合运算。分组仍然是为了聚合，额外提供了聚合后的集合。

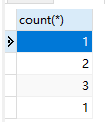
select 列1,列2,聚合... from 表名 group by 列1,列2,列3...

现在有如下一张user表：



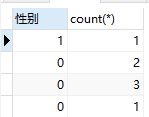
现在我们要按照年龄分组：年龄相同的分为一组

select count(\*) from user group by age;



可以看到统计信息：分成了四组，但没有显示结果集。按照性别查询，新的查询如下：

select gender as 性别,count(\*) from user group by age;查询结果如下：



## 4 分组后筛选：having

select 列1,列2,聚合... from 表名 group by 列1,列2,列3... having 列1,...聚合...

查询男生总人数

方案一

select count(\*)

from students

where gender=1;

-----------------------------------

方案二：

select gender as 性别,count(\*)

from students

group by gender

having gender=1;

对比where与having

where是对from后面指定的表进行数据筛选，属于对原始数据的筛选

having是对group by的结果进行筛选

## 5 获取部分行-分页

从start（第几条数据）开始，获取count（每页展示多少条）条数据，start默认从0开始：

select \* from 表名 limit start,count

示例：

select \* from users limit 2; 只查询2行结果；

SELECT \* FROM users LIMIT 4,2; 从第五行开始取出2行数据；

分页：每页显示m条数据，当前显示第n页

select \* from students

where isdelete=0

limit (n-1)\*m,m

# 四 完整的select语句

select distinct \*

from 表名

where ....

group by ... having ...

order by ...

limit star,count

执行顺序为：

from 表名

where ....

group by ...

select distinct \*

having ...

order by ...

limit star,count

实际使用中，只是语句中某些部分的组合，而不是全部

# 五 SQL与正则

查找username包含 d 的用户：

select username from user where username regexp 'd' order by age;

正则与like的区别：

like匹配整个列，即使要查找的数据在列值中出现，也无法找到；

正则匹配的是列值，只要值内包含要查找的对象就会返回结果集；

正则使用定位符 ^ 与 $ 可以实现与 like 一样的特性。

匹配模式：

’.000’ 特殊符号 . 表示匹配任意字符

‘1000|2000’ 匹配到1000 or 2000

‘[123] ton’ 匹配到1或者2或者3，比如1ton，2ton都会返回结果集，

写为 1|2| ton 也是正确的。

‘[^123] ton’ 匹配除了1或者2或者3之外的数据

‘[a-z]’ 匹配任意字母

‘[1-3]’ 匹配1 2 3

‘\\-’ 匹配带 - 的数据，由于 - 是特殊字符，需要 \\ 转义

mysql中常见预定义的匹配：

[:alnum:] 任意字母和数字，同：[a-zA-Z0-9]

[:alpha:] 任意字符，同：[a-zA-Z]

[:digit:] 任意数字，同：[0-9]

[:lower:] 任意小写字母，同：[a-z]

[:uper:] 任意大写字母

元字符：

\* 0个或多个匹配

+ 1个或多个匹配，同 1,{}

？ 0或1个匹配，同{0,1}

{n} 指定数目的匹配

{n,} 不少于指定数目的匹配

{n,m} 匹配数目的范围，m不能超过255

^ 文本的开始

$ 文本的结尾

[[:<:]] 词的开始

[[:>:]] 词的结束

案例：

‘\\([0-9] sticks?)’ 匹配？前面的s有或者没有

注意：可以在不使用数据库的情况下测试正则，如：

select ‘hello’ regexp ‘[0-9]’;

# 六 数据处理函数

注意：虽然函数的出现让sql的书写变得简便，且有助于提升性能，但是却造成了数据在不同数据库管理软件之间迁移的困难，因为不同的数据库软件支持的函数不尽一致，所以在书写函数时，尽量给出注释。

## 1 拼接函数 concat()

select concat(username,'(', age, ')') from user;



注意：大多数数据库软件使用的是 + 或者 || 来实现拼接，mysql使用的是 concat函数！

## 2 去除空格函数 trim()

select concat(trim(username)) from user;

此外还有，去除左边空格：ltrim()，去除右边空格rtrim()。

## 3 常见函数

upper() 转换为大写

lower() 转换为小写

left() 返回串左边的字符

rigt()

length() 返回串长度

locate() 找出串的第一个子串

subString() 返回串的子串

## 4 日期和时间处理函数

日期和时间采用响应的数据类型和特殊的格式存储，以便能快速的排序、过滤，并且节省物理存储空间。

一般应用程序不使用用来存储日期和时间的格式，因此日期和时间处理函数通常用来读取、统计、处理这些值。

常见日期和时间处理函数：

NOW() 返回当前的日期和时间

CURDATE() 返回当前的日期

CURTIME() 返回当前的时间

DATE() 提取日期或日期/时间表达式的日期部分

EXTRACT() 返回日期/时间按的单独部分

DATE\_ADD() 给日期添加指定的时间间隔

DATE\_SUB() 从日期减去指定的时间间隔

DATEDIFF() 返回两个日期之间的天数

DATE\_FORMAT() 用不同的格式显示日期/时间

注意：mysql推荐日期写为：yyyy-mm-dd

案例：

select cust\_id,order\_num from orders where order\_date=’2018-01-01’;

这种写法看似没有什么问题，也能查出日期为2018-01-01的订单，但是order\_date的存储格式为datatime，这些值全部具有00:00:00。如果下单的时间具有具体的时间，那么上述的数据就无法查出。

解决办法是仅仅将给出的日期与列中的日期部分进行比较，下列为更可靠的sql语句：

select cust\_id,order\_num from orders where Date(order\_date) = ‘2018-01-01’;

即使存储的只是日期，也推荐使用上述格式查询，当然也有相应的事件函数time()，用法一致。

如何检索出某个时间断下的所有订单？

select cust\_id,order\_num from orders where Date(order\_date) between ‘2005-09-01’ and ‘2005-09-30’;

或者：

select cust\_id,order\_num from orders where Year(order\_date) = 2005 and month(order\_date) = 9

## 5 数值处理函数

abs() 返回绝对值

cos() 返回一个角度的余弦，相应的还有正弦sin() 正切tan()

exp() 返回一个数的指数值

mod() 返回除操作的余数

pi() 返回圆周率

rand() 返回一个随机数

sqrt() 返回一个数的平方根