

MySQL函数大全及用法示例

1、字符串函数

ascii(str)

返回字符串str的第一个字符的ascii值(str是空串时返回0)

```
mysql> select ascii('2');
-> 50
mysql> select ascii(2);
-> 50
mysql> select ascii('dete');
-> 100
```

ord(str)

如果字符串str句首是单字节返回与ascii()函数返回的相同值。

如果是一个多字节字符,以格式返回((first byte ascii code)*256+(second byte ascii code))*256+third byte ascii code...

```
mysql> select ord('2');
-> 50
```

conv(n,from_base,to_base)

对数字n进制转换,并转换为字符串返回(任何参数为null时返回null,进制范围为2-36进制,当to_base是负数时n作为有符号数否则作无符号数,conv以64位点精度工作)

```
mysql> select conv("a",16,2);
-> '1010'
mysql> select conv("6e",18,8);
-> '172'
mysql> select conv(-17,10,-18);
-> '-h'
mysql> select conv(10+"10"+"10"+0xa,10,10);
-> '40'
```

bin(n)

把n转为二进制值并以字符串返回(n是bigint数字,等价于conv(n,10,2))

```
mysql> select bin(12);
-> '1100'
```

oct(n)

把n转为八进制值并以字符串返回(n是bigint数字,等价于conv(n,10,8))

```
mysql> select oct(12);
-> '14'
```

hex(n)

把n转为十六进制并以字符串返回(n是bigint数字,等价于conv(n,10,16))

```
mysql> select hex(255);
-> 'ff'
```

char(n,...)

返回由参数n,...对应的ascii代码字符组成的一个字符串(参数是n,...是数字序列,null值被跳过)

```
mysql> select char(77,121,83,81,'76');
-> 'mysql'
mysql> select char(77,77.3,'77.3');
-> 'mmm'
```

concat(str1,str2,...)

把参数连成一个长字符串并返回(任何参数是null时返回null)

```
mysql> select concat('my', 's', 'ql');
-> 'mysql'
mysql> select concat('my', null, 'ql');
-> null
mysql> select concat(14.3);
-> '14.3'
```

length(str)

octet_length(str)

char_length(str)

character_length(str)

返回字符串str的长度(对于多字节字符char_length仅计算一次)

```
mysql> select length('text');
```

```
-> 4
```

```
mysql> select octet_length('text');
```

```
-> 4
```

locate(substr,str)

position(substr in str)

返回字符串substr在字符串str第一次出现的位置(str不包含substr时返回0)

```
mysql> select locate('bar', 'foobarbar');
```

```
-> 4
```

```
mysql> select locate('xbar', 'foobar');
```

```
-> 0
```

locate(substr,str,pos)

返回字符串substr在字符串str的第pos个位置起第一次出现的位置(str不包含substr时返回0)

```
mysql> select locate('bar', 'foobarbar',5);
```

```
-> 7
```

instr(str,substr)

返回字符串substr在字符串str第一次出现的位置(str不包含substr时返回0)

```
mysql> select instr('foobarbar', 'bar');
```

```
-> 4
```

```
mysql> select instr('xbar', 'foobar');
```

```
-> 0
```

lpad(str,len,padstr)

用字符串padstr填补str左端直到字符串长度为len并返回

```
mysql> select lpad('hi',4,'??');
```

```
-> '??hi'
```

rpadd(str,len,padstr)

用字符串padstr填补str右端直到字符串长度为len并返回

```
mysql> select rpadd('hi',5,'?');
```

```
-> 'hi???'
```

left(str,len)

返回字符串str的左端len个字符

```
mysql> select left('foobarbar', 5);
```

```
-> 'fooba'
```

right(str,len)

返回字符串str的右端len个字符

```
mysql> select right('foobarbar', 4);
```

```
-> 'rbar'
```

substring(str,pos,len)

substring(str from pos for len)

mid(str,pos,len)

返回字符串str的位置pos起len个字符mysql> select substring('quadratically',5,6);

```
-> 'ratica'
```

substring(str,pos)

substring(str from pos)

返回字符串str的位置pos起的一个子串

```
mysql> select substring('quadratically',5);
```

```
-> 'ratically'
```

```
mysql> select substring('foobarbar' from 4);
```

```
-> 'barbar'
```

substring_index(str,delim,count)

返回从字符串str的第count个出现的分隔符delim之后的子串

(count为正数时返回左端,否则返回右端子串)

```
mysql> select substring_index('www.mysql.com', '.', 2);
```

```
-> 'www.mysql'
```

```
mysql> select substring_index('www.mysql.com', '.', -2);
```

```
-> 'mysql.com'
```

ltrim(str)

返回删除了左空格的字符串str

```
mysql> select ltrim(' barbar');  
-> 'barbar'
```

rtrim(str)

返回删除了右空格的字符串str

```
mysql> select rtrim('barbar ');  
-> 'barbar'
```

trim([[both | leading | trailing] [remstr] from] str)

返回前缀或后缀remstr被删除了的字符串str(位置参数默认both,remstr默认值为空格)

```
mysql> select trim(' bar ');  
-> 'bar'  
mysql> select trim(leading 'x' from 'xxxbarxxx');  
-> 'barxxx'  
mysql> select trim(both 'x' from 'xxxbarxxx');  
-> 'bar'  
mysql> select trim(trailing 'xyz' from 'barxyz');  
-> 'barx'
```

soundex(str)

返回str的一个同音字符串(听起来"大致相同"字符串有相同的同音字符串,非数字字母字符被忽略,在a-z外的字母被当作元音)

```
mysql> select soundex('hello');  
-> 'h400'  
mysql> select soundex('quadratically');  
-> 'q36324'
```

space(n)

返回由n个空格字符组成的一个字符串

```
mysql> select space(6);  
-> '      '
```

replace(str,from_str,to_str)

用字符串to_str替换字符串str中的子串from_str并返回

```
mysql> select replace('www.mysql.com', 'w', 'ww');  
-> 'wwwwww.mysql.com'
```

repeat(str,count)

返回由count个字符串str连成的一个字符串(任何参数为null时

返回null,count<=0时返回一个空字符串)

```
mysql> select repeat('mysql', 3);  
-> 'mysqlmysqlmysql'
```

reverse(str)

颠倒字符串str的字符顺序并返回

```
mysql> select reverse('abc');  
-> 'cba'
```

insert(str,pos,len,newstr)

把字符串str由位置pos起len个字符长的子串替换为字符串newstr并返回

```
mysql> select insert('quadratic', 3, 4, 'what');  
-> 'quwhattic'
```

elt(n,str1,str2,str3,...)

返回第n个字符串(n小于1或大于参数个数返回null)

```
mysql> select elt(1, 'ej', 'heja', 'hej', 'foo');  
-> 'ej'  
mysql> select elt(4, 'ej', 'heja', 'hej', 'foo');  
-> 'foo'
```

field(str,str1,str2,str3,...)

返回str等于其后的第n个字符串的序号(如果str没找到返回0)

```
mysql> select field('ej', 'hej', 'ej', 'heja', 'hej',  
'foo');
```

```
-> 2
mysql> select field('fo', 'hej', 'ej', 'heja', 'hej',
'foo');
-> 0
```

find_in_set(str,strlist)

返回str在字符串集strlist中的序号(任何参数是null则返回null,如果str没找到返回0,参数1包含", "时工作异常)

```
mysql> select find_in_set('b','a,b,c,d');
-> 2
```

make_set(bits,str1,str2,...)

把参数1的数字转为二进制,假如某个位置的二进制位等于1,对应位置的字符串选入字符串集并返回(null串不添加到结果中)

```
mysql> select make_set(1,'a','b','c');
-> 'a'
mysql> select make_set(1 | 4,'hello','nice','world');
-> 'hello,world'
mysql> select make_set(0,'a','b','c');
-> ''
```

export_set(bits,on,off,[separator,[number_of_bits]])

按bits排列字符串集,只有当位等于1时插入字符串on,否则插入off(separator默认值",",number_of_bits参数使用时长度不足补0而过长截断)

```
mysql> select export_set(5,'y','n',' ',4)
-> y,n,y,n
```

lcase(str)

lower(str)

返回小写的字符串str

```
mysql> select lcase('quadratically');
-> 'quadratically'
```

ucase(str)

upper(str)

返回大写的字符串str

```
mysql> select ucase('quadratically');
-> 'quadratically'
```

load_file(file_name)

读入文件并且作为一个字符串返回文件内容(文件无法找到,路径不完整,没有权限,长度大于max_allowed_packet会返回null)
mysql> update table_name set blob_column=load_file
("/tmp/picture") where id=1;

2、数学函数

abs(n)

返回n的绝对值

```
mysql> select abs(2);
-> 2
mysql> select abs(-32);
-> 32
```

sign(n)

返回参数的符号(为-1、0或1)

```
mysql> select sign(-32);
-> -1
mysql> select sign(0);
-> 0
mysql> select sign(234);
-> 1
```

mod(n,m)

取模运算,返回n被m除的余数(同%操作符)

```
mysql> select mod(234, 10);
-> 4
mysql> select 234 % 10;
```

```
-> 4
mysql> select mod(29,9);
-> 2
```

floor(n)

返回不大于n的最大整数值

```
mysql> select floor(1.23);
-> 1
mysql> select floor(-1.23);
-> -2
```

ceiling(n)

返回不小于n的最小整数值

```
mysql> select ceiling(1.23);
-> 2
mysql> select ceiling(-1.23);
-> -1
```

round(n,d)

返回n的四舍五入值,保留d位小数(d的默认值为0)

```
mysql> select round(-1.23);
-> -1
mysql> select round(-1.58);
-> -2
mysql> select round(1.58);
-> 2
mysql> select round(1.298, 1);
-> 1.3
mysql> select round(1.298, 0);
-> 1
```

exp(n)

返回值e的n次方(自然对数的底)

```
mysql> select exp(2);
-> 7.389056
mysql> select exp(-2);
-> 0.135335
```

log(n)

返回n的自然对数

```
mysql> select log(2);
-> 0.693147
mysql> select log(-2);
-> null
```

log10(n)

返回n以10为底的对数

```
mysql> select log10(2);
-> 0.301030
mysql> select log10(100);
-> 2.000000
mysql> select log10(-100);
-> null
```

pow(x,y)

power(x,y)

返回值x的y次幂

```
mysql> select pow(2,2);
-> 4.000000
mysql> select pow(2,-2);
-> 0.250000
```

sqrt(n)

返回非负数n的平方根

```
mysql> select sqrt(4);
-> 2.000000
mysql> select sqrt(20);
-> 4.472136
```

pi()

返回圆周率

```
mysql> select pi();  
-> 3.141593
```

cos(n)

返回n的余弦值

```
mysql> select cos(pi());  
-> -1.000000
```

sin(n)

返回n的正弦值

```
mysql> select sin(pi());  
-> 0.000000
```

tan(n)

返回n的正切值

```
mysql> select tan(pi()+1);  
-> 1.557408
```

acos(n)

返回n反余弦(n是余弦值,在-1到1的范围,否则返回null)

```
mysql> select acos(1);  
-> 0.000000  
mysql> select acos(1.0001);  
-> null  
mysql> select acos(0);  
-> 1.570796
```

asin(n)

返回n反正弦值

```
mysql> select asin(0.2);  
-> 0.201358  
mysql> select asin('foo');  
-> 0.000000
```

atan(n)

返回n的反正切值

```
mysql> select atan(2);  
-> 1.107149  
mysql> select atan(-2);  
-> -1.107149
```

atan2(x,y)

返回2个变量x和y的反正切(类似y/x的反正切,符号决定象限)

```
mysql> select atan(-2,2);  
-> -0.785398  
mysql> select atan(pi(),0);  
-> 1.570796
```

cot(n)

返回x的余切

```
mysql> select cot(12);  
-> -1.57267341  
mysql> select cot(0);  
-> null
```

rand()

rand(n)

返回在范围0到1.0内的随机浮点值(可以使用数字n作为初始值)

```
mysql> select rand();  
-> 0.5925  
mysql> select rand(20);  
-> 0.1811  
mysql> select rand(20);  
-> 0.1811  
mysql> select rand();
```

```
-> 0.2079
mysql> select rand();
-> 0.7888
```

degrees(n)

把n从弧度变换为角度并返回

```
mysql> select degrees(pi());
-> 180.000000
```

radians(n)

把n从角度变换为弧度并返回

```
mysql> select radians(90);
-> 1.570796
```

truncate(n,d)

保留数字n的d位小数并返回

```
mysql> select truncate(1.223,1);
-> 1.2
mysql> select truncate(1.999,1);
-> 1.9
mysql> select truncate(1.999,0);
-> 1
```

least(x,y,...)

返回最小值(如果返回值被用在整数(实数或大小敏感字符串)上下文或所有参数都是整数(实数或大小敏感字符串)则他们作为整数(实数或大小敏感字符串)比较,否则按忽略大小写的字符串被比较)

```
mysql> select least(2,0);
-> 0
mysql> select least(34.0,3.0,5.0,767.0);
-> 3.0
mysql> select least("b","a","c");
-> "a"
```

greatest(x,y,...)

返回最大值(其余同least())

```
mysql> select greatest(2,0);
-> 2
mysql> select greatest(34.0,3.0,5.0,767.0);
-> 767.0
mysql> select greatest("b","a","c");
-> "c"
```

3、时期时间函数

dayofweek(date)

返回日期date是星期几(1=星期天,2=星期一,.....7=星期六,odbc标准)

```
mysql> select dayofweek('1998-02-03');
-> 3
```

weekday(date)

返回日期date是星期几(0=星期一,1=星期二,.....6= 星期天)。

```
mysql> select weekday('1997-10-04 22:23:00');
-> 5
mysql> select weekday('1997-11-05');
-> 2
```

dayofmonth(date)

返回date是一月中的第几日(在1到31范围内)

```
mysql> select dayofmonth('1998-02-03');
-> 3
```

dayofyear(date)

返回date是一年中的第几日(在1到366范围内)

```
mysql> select dayofyear('1998-02-03');
-> 34
```

month(date)

返回date中的月份数值

```
mysql> select month('1998-02-03');
-> 2
```

dayname(date)

返回date是星期几(按英文名返回)

```
mysql> select dayname("1998-02-05");
-> 'thursday'
```

monthname(date)

返回date是几月(按英文名返回)

```
mysql> select monthname("1998-02-05");
-> 'february'
```

quarter(date)

返回date是一年的第几个季度

```
mysql> select quarter('98-04-01');
-> 2
```

week(date,first)

返回date是一年的第几周(first默认值0,first取值1表示周一是一年的开始,0从周日开始)

```
mysql> select week('1998-02-20');
-> 7
mysql> select week('1998-02-20',0);
-> 7
mysql> select week('1998-02-20',1);
-> 8
```

year(date)

返回date的年份(范围在1000到9999)

```
mysql> select year('98-02-03');
-> 1998
```

hour(time)

返回time的小时数(范围是0到23)

```
mysql> select hour('10:05:03');
-> 10
```

minute(time)

返回time的分钟数(范围是0到59)

```
mysql> select minute('98-02-03 10:05:03');
-> 5
```

second(time)

返回time的秒数(范围是0到59)

```
mysql> select second('10:05:03');
-> 3
```

period_add(p,n)

增加n个月到时期p并返回(p的格式yymm或yyyymm)

```
mysql> select period_add(9801,2);
-> 199803
```

period_diff(p1,p2)

返回在时期p1和p2之间月数(p1和p2的格式yymm或yyyymm)

```
mysql> select period_diff(9802,199703);
-> 11
```

date_add(date,interval expr type)

date_sub(date,interval expr type)

adddate(date,interval expr type)

subdate(date,interval expr type)

对日期时间进行加减法运算

(adddate())和subdate()是date_add()和date_sub()的同义词,也

可以用运算符+和-而不是函数

date是一个datetime或date值,expr对date进行加减法的一个表

达式字符串type指明表达式expr应该如何被解释

[type值 含义 期望的expr格式]:

second 秒 seconds
minute 分钟 minutes
hour 时间 hours
day 天 days
month 月 months
year 年 years
minute_second 分钟和秒 "minutes:seconds"
hour_minute 小时和分钟 "hours:minutes"
day_hour 天和小时 "days hours"
year_month 年和月 "years-months"
hour_second 小时, 分钟, "hours:minutes:seconds"
day_minute 天, 小时, 分钟 "days hours:minutes"
day_second 天, 小时, 分钟, 秒 "days
hours:minutes:seconds"
expr中允许任何标点做分隔符,如果所有是date值时结果是一个date值,否则结果是一个datetime值)
如果type关键词不完整,则mysql从右端取值,day_second因为缺少小时分钟等于minute_second)
如果增加month、year_month或year,天数大于结果月份的最大天数则使用最大天数)
mysql> select "1997-12-31 23:59:59" + interval 1 second;

```
-> 1998-01-01 00:00:00
mysql> select interval 1 day + "1997-12-31";
-> 1998-01-01
mysql> select "1998-01-01" - interval 1 second;
-> 1997-12-31 23:59:59
mysql> select date_add("1997-12-31 23:59:59",interval 1 second);
-> 1998-01-01 00:00:00
mysql> select date_add("1997-12-31 23:59:59",interval 1 day);
-> 1998-01-01 23:59:59
mysql> select date_add("1997-12-31 23:59:59",interval "1:1" minute_second);
-> 1998-01-01 00:01:00
mysql> select date_sub("1998-01-01 00:00:00",interval "1 1:1:1" day_second);
-> 1997-12-30 22:58:59
mysql> select date_add("1998-01-01 00:00:00", interval "- 1 10" day_hour);
-> 1997-12-30 14:00:00
mysql> select date_sub("1998-01-02", interval 31 day);
-> 1997-12-02
mysql> select extract(year from "1999-07-02");
-> 1999
mysql> select extract(year_month from "1999-07-02 01:02:03");
-> 199907
mysql> select extract(day_minute from "1999-07-02 01:02:03");
-> 20102
```

to_days(date)

返回日期date是西元0年至今多少天(不计算1582年以前)

```
mysql> select to_days(950501);
-> 728779
mysql> select to_days('1997-10-07');
-> 729669
```

from_days(n)

给出西元0年至今多少天返回date值(不计算1582年以前)

```
mysql> select from_days(729669);
-> '1997-10-07'
```

date_format(date,format)

根据format字符串格式化date值

(在format字符串中可用标志符:

%m 月名字(january.....december)
 %w 星期名字(sunday.....saturday)
 %d 有英语前缀的月份的日期(1st, 2nd, 3rd, 等等。)
 %y 年, 数字, 4 位
 %Y 年, 数字, 2 位
 %a 缩写的星期名字(sun.....sat)
 %d 月份中的天数, 数字(00.....31)
 %e 月份中的天数, 数字(0.....31)
 %m 月, 数字(01.....12)
 %c 月, 数字(1.....12)
 %b 缩写的月份名字(jan.....dec)
 %j 一年中的天数(001.....366)
 %h 小时(00.....23)
 %k 小时(0.....23)
 %H 小时(01.....12)
 %I 小时(01.....12)
 %l 小时(1.....12)
 %i 分钟, 数字(00.....59)
 %r 时间,12 小时(hh:mm:ss [ap]m)
 %t 时间,24 小时(hh:mm:ss)
 %s 秒(00.....59)
 %S 秒(00.....59)
 %p am或pm
 %w 一个星期中的天数(0=sunday6=saturday)
 %u 星期(0.....52), 这里星期一是星期的第一天
 %U 星期(0.....52), 这里星期一是星期的第一天
 %% 字符%)

```

mysql> select date_format('1997-10-04 22:23:00','%w %m %
y');
    -> 'saturday october 1997'
mysql> select date_format('1997-10-04 22:23:00','%h:%i:%
s');
    -> '22:23:00'
mysql> select date_format('1997-10-04 22:23:00','%d %y %a
%d %m %b %j');
    -> '4th 97 sat 04 10 oct 277'
mysql> select date_format('1997-10-04 22:23:00','%h %k %i
%r %t %s %w');
    -> '22 22 10 10:23:00 pm 22:23:00 00 6'
  
```

time_format(time,format)

和date_format()类似,但time_format只处理小时、分钟和秒(其余符号产生一个null值或0)

curdate()

current_date()

以'yyyy-mm-dd'或'yyyymmdd'格式返回当前日期值(根据返回值所处上下文是字符串或数字)

```

mysql> select curdate();
    -> '1997-12-15'
mysql> select curdate() + 0;
    -> 19971215
  
```

curtime()

current_time()

以'hh:mm:ss'或'hhmmss'格式返回当前时间值(根据返回值所处上下文是字符串或数字)

```

mysql> select curtime();
    -> '23:50:26'
mysql> select curtime() + 0;
    -> 235026
  
```

now()

sysdate()

current_timestamp()

以'yyyy-mm-dd hh:mm:ss'或'yyyymmddhhmmss'格式返回当前日期时间(根据返回值所处上下文是字符串或数字)

```

mysql> select now();
  
```

```
-> '1997-12-15 23:50:26'
mysql> select now() + 0;
-> 19971215235026
```

unix_timestamp() **unix_timestamp(date)**

返回一个unix时间戳(从'1970-01-01 00:00:00'gmt开始的秒数,date默认值为当前时间)

```
mysql> select unix_timestamp();
-> 882226357
mysql> select unix_timestamp('1997-10-04 22:23:00');
-> 875996580
```

from_unixtime(unix_timestamp)

以'yyyy-mm-dd hh:mm:ss'或'yyyymmddhhmmss'格式返回时间戳的值(根据返回值所处上下文是字符串或数字)

```
mysql> select from_unixtime(875996580);
-> '1997-10-04 22:23:00'
mysql> select from_unixtime(875996580) + 0;
-> 19971004222300
```

from_unixtime(unix_timestamp,format)

以format字符串格式返回时间戳的值

```
mysql> select from_unixtime(unix_timestamp()),'%y %d %m %h:%i:%s %x');
-> '1997 23rd december 03:43:30 x'
```

sec_to_time(seconds)

以'hh:mm:ss'或'hhmmss'格式返回秒数转成的time值(根据返回值所处上下文是字符串或数字)

```
mysql> select sec_to_time(2378);
-> '00:39:38'
mysql> select sec_to_time(2378) + 0;
-> 3938
```

time_to_sec(time)

返回time值有多少秒

```
mysql> select time_to_sec('22:23:00');
-> 80580
mysql> select time_to_sec('00:39:38');
-> 2378
```

转换函数

cast

用法: cast(字段 as 数据类型) [当然是否可以成功转换,还要看数据类型强制转化时注意的问题]

实例: select cast(a as unsigned) as b from cardserver where order by b desc;

convert:

用法: convert(字段,数据类型)

实例: select convert(a ,unsigned) as b from cardserver where order by b desc;