**mysql常用函数**

**一、数学函数**  
ABS(x)   返回x的绝对值

BIN(x)   返回x的二进制（OCT返回八进制，HEX返回十六进制）  
CEILING(x)   返回大于x的最小整数值  
EXP(x)   返回值e（自然对数的底）的x次方  
FLOOR(x)   返回小于x的最大整数值  
GREATEST(x1,x2,...,xn)返回集合中最大的值  
LEAST(x1,x2,...,xn)      返回集合中最小的值  
LN(x)                    返回x的自然对数  
LOG(x,y)返回x的以y为底的对数  
MOD(x,y)                 返回x/y的模（余数）  
PI()返回pi的值（圆周率）  
RAND()返回０到１内的随机值,可以通过提供一个参数(种子)使RAND()随机数生成器生成一个指定的值。  
ROUND(x,y)返回参数x的四舍五入的有y位小数的值  
SIGN(x) 返回代表数字x的符号的值  
SQRT(x) 返回一个数的平方根  
TRUNCATE(x,y)            返回数字x截短为y位小数的结果  
  
**二、聚合函数(常用于GROUP BY从句的SELECT查询中)**  
AVG(col)返回指定列的平均值  
COUNT(col)返回指定列中非NULL值的个数  
MIN(col)返回指定列的最小值  
MAX(col)返回指定列的最大值  
SUM(col)返回指定列的所有值之和  
GROUP\_CONCAT(col) 返回由属于一组的列值连接组合而成的结果  
  
**三、字符串函数**  
ASCII(char)返回字符的ASCII码值  
BIT\_LENGTH(str)返回字符串的比特长度  
CONCAT(s1,s2...,sn)将s1,s2...,sn连接成字符串  
CONCAT\_WS(sep,s1,s2...,sn)将s1,s2...,sn连接成字符串，并用sep字符间隔  
INSERT(str,x,y,instr) 将字符串str从第x位置开始，y个字符长的子串替换为字符串instr，返回结果  
FIND\_IN\_SET(str,list)分析逗号分隔的list列表，如果发现str，返回str在list中的位置  
LCASE(str)或LOWER(str) 返回将字符串str中所有字符改变为小写后的结果  
LEFT(str,x)返回字符串str中最左边的x个字符  
LENGTH(s)返回字符串str中的字符数  
LTRIM(str) 从字符串str中切掉开头的空格  
POSITION(substr,str) 返回子串substr在字符串str中第一次出现的位置  
QUOTE(str) 用反斜杠转义str中的单引号  
REPEAT(str,srchstr,rplcstr)返回字符串str重复x次的结果  
REVERSE(str) 返回颠倒字符串str的结果  
RIGHT(str,x) 返回字符串str中最右边的x个字符  
RTRIM(str) 返回字符串str尾部的空格  
STRCMP(s1,s2)比较字符串s1和s2  
TRIM(str)去除字符串首部和尾部的所有空格  
UCASE(str)或UPPER(str) 返回将字符串str中所有字符转变为大写后的结果  
  
**四、日期和时间函数**  
CURDATE()或CURRENT\_DATE() 返回当前的日期  
CURTIME()或CURRENT\_TIME() 返回当前的时间  
DATE\_ADD(date,INTERVAL int keyword)返回日期date加上间隔时间int的结果(int必须按照关键字进行格式化),如：SELECTDATE\_ADD(CURRENT\_DATE,INTERVAL 6 MONTH);  
DATE\_FORMAT(date,fmt)  依照指定的fmt格式格式化日期date值  
DATE\_SUB(date,INTERVAL int keyword)返回日期date加上间隔时间int的结果(int必须按照关键字进行格式化),如：SELECTDATE\_SUB(CURRENT\_DATE,INTERVAL 6 MONTH);  
DAYOFWEEK(date)   返回date所代表的一星期中的第几天(1~7)  
DAYOFMONTH(date)  返回date是一个月的第几天(1~31)  
DAYOFYEAR(date)   返回date是一年的第几天(1~366)  
DAYNAME(date)   返回date的星期名，如：SELECT DAYNAME(CURRENT\_DATE);  
FROM\_UNIXTIME(ts,fmt)  根据指定的fmt格式，格式化UNIX时间戳ts  
HOUR(time)   返回time的小时值(0~23)  
MINUTE(time)   返回time的分钟值(0~59)  
MONTH(date)   返回date的月份值(1~12)  
MONTHNAME(date)   返回date的月份名，如：SELECT MONTHNAME(CURRENT\_DATE);  
NOW()    返回当前的日期和时间  
QUARTER(date)   返回date在一年中的季度(1~4)，如SELECT QUARTER(CURRENT\_DATE);  
WEEK(date)   返回日期date为一年中第几周(0~53)  
YEAR(date)   返回日期date的年份(1000~9999)  
一些示例：  
获取当前系统时间：SELECT FROM\_UNIXTIME(UNIX\_TIMESTAMP());  
SELECT EXTRACT(YEAR\_MONTH FROM CURRENT\_DATE);  
SELECT EXTRACT(DAY\_SECOND FROM CURRENT\_DATE);  
SELECT EXTRACT(HOUR\_MINUTE FROM CURRENT\_DATE);  
返回两个日期值之间的差值(月数)：SELECT PERIOD\_DIFF(200302,199802);  
在Mysql中计算年龄：  
SELECT DATE\_FORMAT(FROM\_DAYS(TO\_DAYS(NOW())-TO\_DAYS(birthday)),'%Y')+0 AS age FROM employee;  
这样，如果Brithday是未来的年月日的话，计算结果为0。  
下面的SQL语句计算员工的绝对年龄，即当Birthday是未来的日期时，将得到负值。  
SELECT DATE\_FORMAT(NOW(), '%Y') - DATE\_FORMAT(birthday, '%Y') -(DATE\_FORMAT(NOW(), '00-%m-%d') <DATE\_FORMAT(birthday, '00-%m-%d')) AS age from employee  
  
**五、加密函数**  
AES\_ENCRYPT(str,key)  返回用密钥key对字符串str利用高级加密标准算法加密后的结果，调用AES\_ENCRYPT的结果是一个二进制字符串，以BLOB类型存储  
AES\_DECRYPT(str,key)  返回用密钥key对字符串str利用高级加密标准算法解密后的结果  
DECODE(str,key)   使用key作为密钥解密加密字符串str  
ENCRYPT(str,salt)   使用UNIXcrypt()函数，用关键词salt(一个可以惟一确定口令的字符串，就像钥匙一样)加密字符串str  
ENCODE(str,key)   使用key作为密钥加密字符串str，调用ENCODE()的结果是一个二进制字符串，它以BLOB类型存储  
MD5()    计算字符串str的MD5校验和  
PASSWORD(str)   返回字符串str的加密版本，这个加密过程是不可逆转的，和UNIX密码加密过程使用不同的算法。  
SHA()    计算字符串str的安全散列算法(SHA)校验和  
示例：  
SELECT ENCRYPT('root','salt');  
SELECT ENCODE('xufeng','key');  
SELECT DECODE(ENCODE('xufeng','key'),'key');#加解密放在一起  
SELECT AES\_ENCRYPT('root','key');  
SELECT AES\_DECRYPT(AES\_ENCRYPT('root','key'),'key');  
SELECT MD5('123456');  
SELECT SHA('123456');  
  
**六、控制流函数**  
MySQL有4个函数是用来进行条件操作的，这些函数可以实现SQL的条件逻辑，允许开发者将一些应用程序业务逻辑转换到数据库后台。  
MySQL控制流函数：  
CASE WHEN[test1] THEN [result1]...ELSE [default] END如果testN是真，则返回resultN，否则返回default  
CASE [test] WHEN[val1] THEN [result]...ELSE [default]END  如果test和valN相等，则返回resultN，否则返回default  
IF(test,t,f)   如果test是真，返回t；否则返回f  
IFNULL(arg1,arg2) 如果arg1不是空，返回arg1，否则返回arg2  
NULLIF(arg1,arg2) 如果arg1=arg2返回NULL；否则返回arg1  
这些函数的第一个是IFNULL()，它有两个参数，并且对第一个参数进行判断。如果第一个参数不是NULL，函数就会向调用者返回第一个参数；如果是NULL,将返回第二个参数。  
如：SELECT IFNULL(1,2), IFNULL(NULL,10),IFNULL(4\*NULL,'false');  
NULLIF()函数将会检验提供的两个参数是否相等，如果相等，则返回NULL，如果不相等，就返回第一个参数。  
如：SELECT NULLIF(1,1),NULLIF('A','B'),NULLIF(2+3,4+1);  
和许多脚本语言提供的IF()函数一样，MySQL的IF()函数也可以建立一个简单的条件测试，这个函数有三个参数，第一个是要被判断的表达式，如果表达式为真，IF()将会返回第二个参数，如果为假，IF()将会返回第三个参数。  
如：SELECTIF(1<10,2,3),IF(56>100,'true','false');  
IF()函数在只有两种可能结果时才适合使用。然而，在现实世界中，我们可能发现在条件测试中会需要多个分支。在这种情况下，MySQL提供了CASE函数，它和PHP及Perl语言的switch-case条件例程一样。  
CASE函数的格式有些复杂，通常如下所示：  
CASE [expression to be evaluated]  
WHEN [val 1] THEN [result 1]  
WHEN [val 2] THEN [result 2]  
WHEN [val 3] THEN [result 3]  
......  
WHEN [val n] THEN [result n]  
ELSE [default result]  
END  
这里，第一个参数是要被判断的值或表达式，接下来的是一系列的WHEN-THEN块，每一块的第一个参数指定要比较的值，如果为真，就返回结果。所有的WHEN-THEN块将以ELSE块结束，当END结束了所有外部的CASE块时，如果前面的每一个块都不匹配就会返回ELSE块指定的默认结果。如果没有指定ELSE块，而且所有的WHEN-THEN比较都不是真，MySQL将会返回NULL。  
CASE函数还有另外一种句法，有时使用起来非常方便，如下：  
CASE  
WHEN [conditional test 1] THEN [result 1]  
WHEN [conditional test 2] THEN [result 2]  
ELSE [default result]  
END  
这种条件下，返回的结果取决于相应的条件测试是否为真。  
示例：  
mysql>SELECT CASE 'green'  
     WHEN 'red' THEN 'stop'  
     WHEN 'green' THEN 'go' END;  
SELECT CASE 9 WHEN 1 THEN 'a' WHEN 2 THEN 'b' ELSE 'N/A' END;  
SELECT CASE WHEN (2+2)=4 THEN 'OK' WHEN(2+2)<>4 THEN 'not OK' END ASSTATUS;  
SELECT Name,IF((IsActive = 1),'已激活','未激活') AS RESULT FROMUserLoginInfo;  
SELECT fname,lname,(math+sci+lit) AS total,  
CASE WHEN (math+sci+lit) < 50 THEN 'D'  
WHEN (math+sci+lit) BETWEEN 50 AND 150 THEN 'C'  
WHEN (math+sci+lit) BETWEEN 151 AND 250 THEN 'B'  
ELSE 'A' END  
AS grade FROM marks;  
SELECT IF(ENCRYPT('sue','ts')=upass,'allow','deny') AS LoginResultFROM users WHERE uname = 'sue';#一个登陆验证  
  
**七、格式化函数**  
DATE\_FORMAT(date,fmt)  依照字符串fmt格式化日期date值  
FORMAT(x,y)   把x格式化为以逗号隔开的数字序列，y是结果的小数位数  
INET\_ATON(ip)   返回IP地址的数字表示  
INET\_NTOA(num)   返回数字所代表的IP地址  
TIME\_FORMAT(time,fmt)  依照字符串fmt格式化时间time值  
其中最简单的是FORMAT()函数，它可以把大的数值格式化为以逗号间隔的易读的序列。  
示例：  
SELECT FORMAT(34234.34323432,3);  
SELECT DATE\_FORMAT(NOW(),'%W,%D %M %Y %r');  
SELECT DATE\_FORMAT(NOW(),'%Y-%m-%d');  
SELECT DATE\_FORMAT(19990330,'%Y-%m-%d');  
SELECT DATE\_FORMAT(NOW(),'%h:%i %p');  
SELECT INET\_ATON('10.122.89.47');  
SELECT INET\_NTOA(175790383);  
  
**八、类型转化函数**  
为了进行数据类型转化，MySQL提供了CAST()函数，它可以把一个值转化为指定的数据类型。类型有：BINARY,CHAR,DATE,TIME,DATETIME,SIGNED,UNSIGNED  
示例：  
SELECT CAST(NOW() AS SIGNED INTEGER),CURDATE()+0;  
SELECT 'f'=BINARY 'F','f'=CAST('F' AS BINARY);  
  
**九、系统信息函数**  
DATABASE()   返回当前数据库名  
BENCHMARK(count,expr)  将表达式expr重复运行count次  
CONNECTION\_ID()   返回当前客户的连接ID  
FOUND\_ROWS()   返回最后一个SELECT查询进行检索的总行数  
USER()或SYSTEM\_USER()  返回当前登陆用户名  
VERSION()   返回MySQL服务器的版本  
示例：  
SELECT DATABASE(),VERSION(),USER();  
SELECTBENCHMARK(9999999,LOG(RAND()\*PI()));#该例中,MySQL计算LOG(RAND()\*PI())表达式9999999次。