控制流函数  
http://www.cnblogs.com/Images/OutliningIndicators/None.gifIFNULL(expr1,expr2)   
http://www.cnblogs.com/Images/OutliningIndicators/None.gif如果expr1不是NULL，IFNULL()返回expr1，否则它返回expr2。IFNULL()返回一个数字或字符串值，取决于它被使用的上下文环境。   
http://www.cnblogs.com/Images/OutliningIndicators/None.gifmysql> select IFNULL(1,0);  
http://www.cnblogs.com/Images/OutliningIndicators/None.gif        -> 1  
http://www.cnblogs.com/Images/OutliningIndicators/None.gifmysql> select IFNULL(0,10);  
http://www.cnblogs.com/Images/OutliningIndicators/None.gif        -> 0  
http://www.cnblogs.com/Images/OutliningIndicators/None.gifmysql> select IFNULL(1/0,10);  
http://www.cnblogs.com/Images/OutliningIndicators/None.gif        -> 10  
http://www.cnblogs.com/Images/OutliningIndicators/None.gifmysql> select IFNULL(1/0,'yes');  
http://www.cnblogs.com/Images/OutliningIndicators/None.gif        -> 'yes'  
http://www.cnblogs.com/Images/OutliningIndicators/None.gif IF(expr1,expr2,expr3)   
http://www.cnblogs.com/Images/OutliningIndicators/None.gif如果expr1是TRUE(expr1<>0且expr1<>NULL)，那么IF()返回expr2，否则它返回expr3。IF()返回一个数字或字符串值，取决于它被使用的上下文。   
http://www.cnblogs.com/Images/OutliningIndicators/None.gifmysql> select IF(1>2,2,3);  
http://www.cnblogs.com/Images/OutliningIndicators/None.gif        -> 3  
http://www.cnblogs.com/Images/OutliningIndicators/None.gifmysql> select IF(1<2,'yes','no');  
http://www.cnblogs.com/Images/OutliningIndicators/None.gif        -> 'yes'  
http://www.cnblogs.com/Images/OutliningIndicators/None.gifmysql> select IF(strcmp('test','test1'),'yes','no');  
http://www.cnblogs.com/Images/OutliningIndicators/None.gif        -> 'no'  
http://www.cnblogs.com/Images/OutliningIndicators/None.gifexpr1作为整数值被计算，它意味着如果你正在测试浮点或字符串值，你应该使用一个比较操作来做。   
http://www.cnblogs.com/Images/OutliningIndicators/None.gif  
http://www.cnblogs.com/Images/OutliningIndicators/None.gifmysql> select IF(0.1,1,0);  
http://www.cnblogs.com/Images/OutliningIndicators/None.gif        -> 0  
http://www.cnblogs.com/Images/OutliningIndicators/None.gifmysql> select IF(0.1<>0,1,0);  
http://www.cnblogs.com/Images/OutliningIndicators/None.gif        -> 1  
http://www.cnblogs.com/Images/OutliningIndicators/None.gif在上面的第一种情况中，IF(0.1)返回0，因为0.1被变换到整数值, 导致测试IF(0)。这可能不是你期望的。在第二种情况中，比较测试原来的浮点值看它是否是非零，比较的结果被用作一个整数。   
http://www.cnblogs.com/Images/OutliningIndicators/None.gif  
http://www.cnblogs.com/Images/OutliningIndicators/None.gifCASE value WHEN [compare-value] THEN result [WHEN [compare-value] THEN result http://www.cnblogs.com/Images/dot.gif] [ELSE result] END   
http://www.cnblogs.com/Images/OutliningIndicators/None.gif　   
http://www.cnblogs.com/Images/OutliningIndicators/None.gifCASE WHEN [condition] THEN result [WHEN [condition] THEN result http://www.cnblogs.com/Images/dot.gif] [ELSE result] END   
http://www.cnblogs.com/Images/OutliningIndicators/None.gif第一个版本返回result，其中value=compare-value。第二个版本中如果第一个条件为真，返回result。如果没有匹配的result值，那么结果在ELSE后的result被返回。如果没有ELSE部分，那么NULL被返回。   
http://www.cnblogs.com/Images/OutliningIndicators/None.gifmysql> SELECT CASE 1 WHEN 1 THEN "one" WHEN 2 THEN "two" ELSE "more" END;  
http://www.cnblogs.com/Images/OutliningIndicators/None.gif       -> "one"  
http://www.cnblogs.com/Images/OutliningIndicators/None.gifmysql> SELECT CASE WHEN 1>0 THEN "true" ELSE "false" END;  
http://www.cnblogs.com/Images/OutliningIndicators/None.gif       -> "true"  
http://www.cnblogs.com/Images/OutliningIndicators/None.gifmysql> SELECT CASE BINARY "B" when "a" then 1 when "b" then 2 END;  
http://www.cnblogs.com/Images/OutliningIndicators/None.gif       -> NULL  
http://www.cnblogs.com/Images/OutliningIndicators/None.gif 数学函数  
http://www.cnblogs.com/Images/OutliningIndicators/None.gif  
http://www.cnblogs.com/Images/OutliningIndicators/None.gif所有的数学函数在一个出错的情况下返回NULL。   
http://www.cnblogs.com/Images/OutliningIndicators/None.gif  
http://www.cnblogs.com/Images/OutliningIndicators/None.gif-   
http://www.cnblogs.com/Images/OutliningIndicators/None.gif单目减。改变参数的符号。   
http://www.cnblogs.com/Images/OutliningIndicators/None.gifmysql> select - 2;  
http://www.cnblogs.com/Images/OutliningIndicators/None.gif注意，如果这个操作符与一个BIGINT使用，返回值是一个BIGINT！这意味着你应该避免在整数上使用-，那可能有值-2^63！ ABS(X)   
http://www.cnblogs.com/Images/OutliningIndicators/None.gif返回X的绝对值。   
http://www.cnblogs.com/Images/OutliningIndicators/None.gifmysql> select ABS(2);  
http://www.cnblogs.com/Images/OutliningIndicators/None.gif        -> 2  
http://www.cnblogs.com/Images/OutliningIndicators/None.gifmysql> select ABS(-32);  
http://www.cnblogs.com/Images/OutliningIndicators/None.gif        -> 32  
http://www.cnblogs.com/Images/OutliningIndicators/None.gif该功能可安全用于BIGINT值。   
http://www.cnblogs.com/Images/OutliningIndicators/None.gif  
http://www.cnblogs.com/Images/OutliningIndicators/None.gifSIGN(X)   
http://www.cnblogs.com/Images/OutliningIndicators/None.gif返回参数的符号，为-1、0或1，取决于X是否是负数、零或正数。   
http://www.cnblogs.com/Images/OutliningIndicators/None.gifmysql> select SIGN(-32);  
http://www.cnblogs.com/Images/OutliningIndicators/None.gif        -> -1  
http://www.cnblogs.com/Images/OutliningIndicators/None.gifmysql> select SIGN(0);  
http://www.cnblogs.com/Images/OutliningIndicators/None.gif        -> 0  
http://www.cnblogs.com/Images/OutliningIndicators/None.gifmysql> select SIGN(234);  
http://www.cnblogs.com/Images/OutliningIndicators/None.gif        -> 1  
http://www.cnblogs.com/Images/OutliningIndicators/None.gif   MOD(N,M)   
http://www.cnblogs.com/Images/OutliningIndicators/None.gif　   
http://www.cnblogs.com/Images/OutliningIndicators/None.gif%   
http://www.cnblogs.com/Images/OutliningIndicators/None.gif模 (类似C中的%操作符)。返回N被M除的余数。   
http://www.cnblogs.com/Images/OutliningIndicators/None.gifmysql> select MOD(234, 10);  
http://www.cnblogs.com/Images/OutliningIndicators/None.gif        -> 4  
http://www.cnblogs.com/Images/OutliningIndicators/None.gifmysql> select 253 % 7;  
http://www.cnblogs.com/Images/OutliningIndicators/None.gif        -> 1  
http://www.cnblogs.com/Images/OutliningIndicators/None.gifmysql> select MOD(29,9);  
http://www.cnblogs.com/Images/OutliningIndicators/None.gif        -> 2  
http://www.cnblogs.com/Images/OutliningIndicators/None.gif这个函数可安全用于BIGINT值。  FLOOR(X)   
http://www.cnblogs.com/Images/OutliningIndicators/None.gif返回不大于X的最大整数值。   
http://www.cnblogs.com/Images/OutliningIndicators/None.gif mysql> select FLOOR(1.23);  
http://www.cnblogs.com/Images/OutliningIndicators/None.gif        -> 1  
http://www.cnblogs.com/Images/OutliningIndicators/None.gifmysql> select FLOOR(-1.23);  
http://www.cnblogs.com/Images/OutliningIndicators/None.gif        -> -2  
http://www.cnblogs.com/Images/OutliningIndicators/None.gif注意返回值被变换为一个BIGINT！  CEILING(X)   
http://www.cnblogs.com/Images/OutliningIndicators/None.gif返回不小于X的最小整数值。   
http://www.cnblogs.com/Images/OutliningIndicators/None.gifmysql> select CEILING(1.23);  
http://www.cnblogs.com/Images/OutliningIndicators/None.gif        -> 2  
http://www.cnblogs.com/Images/OutliningIndicators/None.gifmysql> select CEILING(-1.23);  
http://www.cnblogs.com/Images/OutliningIndicators/None.gif        -> -1  
http://www.cnblogs.com/Images/OutliningIndicators/None.gif注意返回值被变换为一个BIGINT！   
http://www.cnblogs.com/Images/OutliningIndicators/None.gif  
http://www.cnblogs.com/Images/OutliningIndicators/None.gifROUND(X)   
http://www.cnblogs.com/Images/OutliningIndicators/None.gif返回参数X的四舍五入的一个整数。   
http://www.cnblogs.com/Images/OutliningIndicators/None.gifmysql> select ROUND(-1.23);  
http://www.cnblogs.com/Images/OutliningIndicators/None.gif        -> -1  
http://www.cnblogs.com/Images/OutliningIndicators/None.gifmysql> select ROUND(-1.58);  
http://www.cnblogs.com/Images/OutliningIndicators/None.gif        -> -2  
http://www.cnblogs.com/Images/OutliningIndicators/None.gifmysql> select ROUND(1.58);  
http://www.cnblogs.com/Images/OutliningIndicators/None.gif        -> 2  
http://www.cnblogs.com/Images/OutliningIndicators/None.gif 注意返回值被变换为一个BIGINT!   
http://www.cnblogs.com/Images/OutliningIndicators/None.gif  
http://www.cnblogs.com/Images/OutliningIndicators/None.gifROUND(X,D)   
http://www.cnblogs.com/Images/OutliningIndicators/None.gif返回参数X的四舍五入的有D为小数的一个数字。如果D为0，结果将没有小数点或小数部分。   
http://www.cnblogs.com/Images/OutliningIndicators/None.gifmysql> select ROUND(1.298, 1);  
http://www.cnblogs.com/Images/OutliningIndicators/None.gif        -> 1.3  
http://www.cnblogs.com/Images/OutliningIndicators/None.gifmysql> select ROUND(1.298, 0);  
http://www.cnblogs.com/Images/OutliningIndicators/None.gif        -> 1  
http://www.cnblogs.com/Images/OutliningIndicators/None.gif注意返回值被变换为一个BIGINT!   
http://www.cnblogs.com/Images/OutliningIndicators/None.gif  
http://www.cnblogs.com/Images/OutliningIndicators/None.gifEXP(X)   
http://www.cnblogs.com/Images/OutliningIndicators/None.gif返回值e（自然对数的底）的X次方。   
http://www.cnblogs.com/Images/OutliningIndicators/None.gifmysql> select EXP(2);  
http://www.cnblogs.com/Images/OutliningIndicators/None.gif        -> 7.389056  
http://www.cnblogs.com/Images/OutliningIndicators/None.gifmysql> select EXP(-2);  
http://www.cnblogs.com/Images/OutliningIndicators/None.gif        -> 0.135335  
http://www.cnblogs.com/Images/OutliningIndicators/None.gif LOG(X)   
http://www.cnblogs.com/Images/OutliningIndicators/None.gif返回X的自然对数。   
http://www.cnblogs.com/Images/OutliningIndicators/None.gifmysql> select LOG(2);  
http://www.cnblogs.com/Images/OutliningIndicators/None.gif        -> 0.693147  
http://www.cnblogs.com/Images/OutliningIndicators/None.gifmysql> select LOG(-2);  
http://www.cnblogs.com/Images/OutliningIndicators/None.gif        -> NULL  
http://www.cnblogs.com/Images/OutliningIndicators/None.gif如果你想要一个数字X的任意底B的对数，使用公式LOG(X)/LOG(B)。   
http://www.cnblogs.com/Images/OutliningIndicators/None.gif  
http://www.cnblogs.com/Images/OutliningIndicators/None.gifLOG10(X)   
http://www.cnblogs.com/Images/OutliningIndicators/None.gif返回X的以10为底的对数。   
http://www.cnblogs.com/Images/OutliningIndicators/None.gifmysql> select LOG10(2);  
http://www.cnblogs.com/Images/OutliningIndicators/None.gif        -> 0.301030  
http://www.cnblogs.com/Images/OutliningIndicators/None.gifmysql> select LOG10(100);  
http://www.cnblogs.com/Images/OutliningIndicators/None.gif        -> 2.000000  
http://www.cnblogs.com/Images/OutliningIndicators/None.gifmysql> select LOG10(-100);  
http://www.cnblogs.com/Images/OutliningIndicators/None.gif        -> NULL  
http://www.cnblogs.com/Images/OutliningIndicators/None.gifPOW(X,Y)   
http://www.cnblogs.com/Images/OutliningIndicators/None.gif　   
http://www.cnblogs.com/Images/OutliningIndicators/None.gifPOWER(X,Y)   
http://www.cnblogs.com/Images/OutliningIndicators/None.gif返回值X的Y次幂。   
http://www.cnblogs.com/Images/OutliningIndicators/None.gifmysql> select POW(2,2);  
http://www.cnblogs.com/Images/OutliningIndicators/None.gif        -> 4.000000  
http://www.cnblogs.com/Images/OutliningIndicators/None.gifmysql> select POW(2,-2);  
http://www.cnblogs.com/Images/OutliningIndicators/None.gif        -> 0.250000SQRT(X)   
http://www.cnblogs.com/Images/OutliningIndicators/None.gif返回非负数X的平方根。   
http://www.cnblogs.com/Images/OutliningIndicators/None.gifmysql> select SQRT(4);  
http://www.cnblogs.com/Images/OutliningIndicators/None.gif        -> 2.000000  
http://www.cnblogs.com/Images/OutliningIndicators/None.gifmysql> select SQRT(20);  
http://www.cnblogs.com/Images/OutliningIndicators/None.gif        -> 4.472136  
http://www.cnblogs.com/Images/OutliningIndicators/None.gif PI()   
http://www.cnblogs.com/Images/OutliningIndicators/None.gif返回PI的值（圆周率）。   
http://www.cnblogs.com/Images/OutliningIndicators/None.gifmysql> select PI();  
http://www.cnblogs.com/Images/OutliningIndicators/None.gif        -> 3.141593  
http://www.cnblogs.com/Images/OutliningIndicators/None.gifCOS(X)   
http://www.cnblogs.com/Images/OutliningIndicators/None.gif返回X的余弦, 在这里X以弧度给出。   
http://www.cnblogs.com/Images/OutliningIndicators/None.gifmysql> select COS(PI());  
http://www.cnblogs.com/Images/OutliningIndicators/None.gif        -> -1.000000  
http://www.cnblogs.com/Images/OutliningIndicators/None.gif SIN(X)   
http://www.cnblogs.com/Images/OutliningIndicators/None.gif返回X的正弦值，在此X以弧度给出。   
http://www.cnblogs.com/Images/OutliningIndicators/None.gifmysql> select SIN(PI());  
http://www.cnblogs.com/Images/OutliningIndicators/None.gif        -> 0.000000  
http://www.cnblogs.com/Images/OutliningIndicators/None.gif TAN(X)   
http://www.cnblogs.com/Images/OutliningIndicators/None.gif返回X的正切值，在此X以弧度给出。   
http://www.cnblogs.com/Images/OutliningIndicators/None.gifmysql> select TAN(PI()+1);  
http://www.cnblogs.com/Images/OutliningIndicators/None.gif        -> 1.557408  
http://www.cnblogs.com/Images/OutliningIndicators/None.gif ACOS(X)   
http://www.cnblogs.com/Images/OutliningIndicators/None.gif返回X反余弦，即其余弦值是X。如果X不在-1到1的范围，返回NULL。   
http://www.cnblogs.com/Images/OutliningIndicators/None.gifmysql> select ACOS(1);  
http://www.cnblogs.com/Images/OutliningIndicators/None.gif        -> 0.000000  
http://www.cnblogs.com/Images/OutliningIndicators/None.gifmysql> select ACOS(1.0001);  
http://www.cnblogs.com/Images/OutliningIndicators/None.gif        -> NULL  
http://www.cnblogs.com/Images/OutliningIndicators/None.gifmysql> select ACOS(0);  
http://www.cnblogs.com/Images/OutliningIndicators/None.gif        -> 1.570796  
http://www.cnblogs.com/Images/OutliningIndicators/None.gif ASIN(X)   
http://www.cnblogs.com/Images/OutliningIndicators/None.gif返回X反正弦值，即其正弦值是X。L如果X不在-1到1的范围，返回NULL。   
http://www.cnblogs.com/Images/OutliningIndicators/None.gifmysql> select ASIN(0.2);  
http://www.cnblogs.com/Images/OutliningIndicators/None.gif        -> 0.201358  
http://www.cnblogs.com/Images/OutliningIndicators/None.gifmysql> select ASIN('foo');  
http://www.cnblogs.com/Images/OutliningIndicators/None.gif        -> 0.000000  
http://www.cnblogs.com/Images/OutliningIndicators/None.gif ATAN(X)   
http://www.cnblogs.com/Images/OutliningIndicators/None.gif返回X的反正切值，即其正切值是X。   
http://www.cnblogs.com/Images/OutliningIndicators/None.gifmysql> select ATAN(2);  
http://www.cnblogs.com/Images/OutliningIndicators/None.gif        -> 1.107149  
http://www.cnblogs.com/Images/OutliningIndicators/None.gifmysql> select ATAN(-2);  
http://www.cnblogs.com/Images/OutliningIndicators/None.gif        -> -1.107149ATAN2(X,Y)   
http://www.cnblogs.com/Images/OutliningIndicators/None.gif返回2个变量X和Y的反正切。它类似于计算Y/X的反正切，除了两个参数的符号被用来决定结果的象限。   
http://www.cnblogs.com/Images/OutliningIndicators/None.gifmysql> select ATAN(-2,2);  
http://www.cnblogs.com/Images/OutliningIndicators/None.gif        -> -0.785398  
http://www.cnblogs.com/Images/OutliningIndicators/None.gifmysql> select ATAN(PI(),0);  
http://www.cnblogs.com/Images/OutliningIndicators/None.gif        -> 1.570796 COT(X)   
http://www.cnblogs.com/Images/OutliningIndicators/None.gif返回X的余切。   
http://www.cnblogs.com/Images/OutliningIndicators/None.gifmysql> select COT(12);  
http://www.cnblogs.com/Images/OutliningIndicators/None.gif        -> -1.57267341  
http://www.cnblogs.com/Images/OutliningIndicators/None.gifmysql> select COT(0);  
http://www.cnblogs.com/Images/OutliningIndicators/None.gif        -> NULL  
http://www.cnblogs.com/Images/OutliningIndicators/None.gifRAND()   
http://www.cnblogs.com/Images/OutliningIndicators/None.gif　   
http://www.cnblogs.com/Images/OutliningIndicators/None.gifRAND(N)   
http://www.cnblogs.com/Images/OutliningIndicators/None.gif返回在范围0到1.0内的随机浮点值。如果一个整数参数N被指定，它被用作种子值。   
http://www.cnblogs.com/Images/OutliningIndicators/None.gifmysql> select RAND();  
http://www.cnblogs.com/Images/OutliningIndicators/None.gif        -> 0.5925  
http://www.cnblogs.com/Images/OutliningIndicators/None.gifmysql> select RAND(20);  
http://www.cnblogs.com/Images/OutliningIndicators/None.gif        -> 0.1811  
http://www.cnblogs.com/Images/OutliningIndicators/None.gifmysql> select RAND(20);  
http://www.cnblogs.com/Images/OutliningIndicators/None.gif        -> 0.1811  
http://www.cnblogs.com/Images/OutliningIndicators/None.gifmysql> select RAND();  
http://www.cnblogs.com/Images/OutliningIndicators/None.gif        -> 0.2079  
http://www.cnblogs.com/Images/OutliningIndicators/None.gifmysql> select RAND();  
http://www.cnblogs.com/Images/OutliningIndicators/None.gif        -> 0.7888  
http://www.cnblogs.com/Images/OutliningIndicators/None.gif你不能在一个ORDER BY子句用RAND()值使用列，因为ORDER BY将重复计算列多次。然而在MySQL3.23中，你可以做： SELECT \* FROM table\_name ORDER BY RAND()，这是有利于得到一个来自SELECT \* FROM table1,table2 WHERE a=b AND c<d ORDER BY RAND() LIMIT 1000的集合的随机样本。注意在一个WHERE子句里的一个RAND()将在每次WHERE被执行时重新评估。   
http://www.cnblogs.com/Images/OutliningIndicators/None.gif  
http://www.cnblogs.com/Images/OutliningIndicators/None.gifLEAST(X,Y,http://www.cnblogs.com/Images/dot.gif)   
http://www.cnblogs.com/Images/OutliningIndicators/None.gif有2和2个以上的参数，返回最小(最小值)的参数。参数使用下列规则进行比较：   
http://www.cnblogs.com/Images/OutliningIndicators/None.gif如果返回值被使用在一个INTEGER上下文，或所有的参数都是整数值，他们作为整数比较。   
http://www.cnblogs.com/Images/OutliningIndicators/None.gif如果返回值被使用在一个REAL上下文，或所有的参数是实数值，他们作为实数比较。   
http://www.cnblogs.com/Images/OutliningIndicators/None.gif如果任何参数是一个大小敏感的字符串，参数作为大小写敏感的字符串被比较。   
http://www.cnblogs.com/Images/OutliningIndicators/None.gif在其他的情况下，参数作为大小写无关的字符串被比较。   
http://www.cnblogs.com/Images/OutliningIndicators/None.gifmysql> select LEAST(2,0);  
http://www.cnblogs.com/Images/OutliningIndicators/None.gif        -> 0  
http://www.cnblogs.com/Images/OutliningIndicators/None.gifmysql> select LEAST(34.0,3.0,5.0,767.0);  
http://www.cnblogs.com/Images/OutliningIndicators/None.gif        -> 3.0  
http://www.cnblogs.com/Images/OutliningIndicators/None.gifmysql> select LEAST("B","A","C");  
http://www.cnblogs.com/Images/OutliningIndicators/None.gif        -> "A"  
http://www.cnblogs.com/Images/OutliningIndicators/None.gif在MySQL 3.22.5以前的版本，你可以使用MIN()而不是LEAST。   
http://www.cnblogs.com/Images/OutliningIndicators/None.gif  
http://www.cnblogs.com/Images/OutliningIndicators/None.gifGREATEST(X,Y,http://www.cnblogs.com/Images/dot.gif)   
http://www.cnblogs.com/Images/OutliningIndicators/None.gif返回最大(最大值)的参数。参数使用与LEAST一样的规则进行比较。   
http://www.cnblogs.com/Images/OutliningIndicators/None.gifmysql> select GREATEST(2,0);  
http://www.cnblogs.com/Images/OutliningIndicators/None.gif        -> 2  
http://www.cnblogs.com/Images/OutliningIndicators/None.gifmysql> select GREATEST(34.0,3.0,5.0,767.0);  
http://www.cnblogs.com/Images/OutliningIndicators/None.gif        -> 767.0  
http://www.cnblogs.com/Images/OutliningIndicators/None.gifmysql> select GREATEST("B","A","C");  
http://www.cnblogs.com/Images/OutliningIndicators/None.gif        -> "C"  
http://www.cnblogs.com/Images/OutliningIndicators/None.gif在MySQL在 3.22.5 以前的版本, 你能使用MAX()而不是GREATEST.  DEGREES(X)   
http://www.cnblogs.com/Images/OutliningIndicators/None.gif返回参数X，从弧度变换为角度。   
http://www.cnblogs.com/Images/OutliningIndicators/None.gifmysql> select DEGREES(PI());  
http://www.cnblogs.com/Images/OutliningIndicators/None.gif        -> 180.000000RADIANS(X)   
http://www.cnblogs.com/Images/OutliningIndicators/None.gif返回参数X，从角度变换为弧度。   
http://www.cnblogs.com/Images/OutliningIndicators/None.gifmysql> select RADIANS(90);  
http://www.cnblogs.com/Images/OutliningIndicators/None.gif        -> 1.570796  
http://www.cnblogs.com/Images/OutliningIndicators/None.gifTRUNCATE(X,D)   
http://www.cnblogs.com/Images/OutliningIndicators/None.gif返回数字X，截断为D位小数。如果D为0，结果将没有小数点或小数部分。   
http://www.cnblogs.com/Images/OutliningIndicators/None.gifmysql> select TRUNCATE(1.223,1);  
http://www.cnblogs.com/Images/OutliningIndicators/None.gif        -> 1.2  
http://www.cnblogs.com/Images/OutliningIndicators/None.gifmysql> select TRUNCATE(1.999,1);  
http://www.cnblogs.com/Images/OutliningIndicators/None.gif        -> 1.9  
http://www.cnblogs.com/Images/OutliningIndicators/None.gifmysql> select TRUNCATE(1.999,0);  
http://www.cnblogs.com/Images/OutliningIndicators/None.gif        -> 1  
http://www.cnblogs.com/Images/OutliningIndicators/None.gif字符串函数对于针对字符串位置的操作，第一个位置被标记为1。 ASCII(str)   
http://www.cnblogs.com/Images/OutliningIndicators/None.gif返回字符串str的最左面字符的ASCII代码值。如果str是空字符串，返回0。如果str是NULL，返回NULL。   
http://www.cnblogs.com/Images/OutliningIndicators/None.gifmysql> select ASCII('2');  
http://www.cnblogs.com/Images/OutliningIndicators/None.gif        -> 50  
http://www.cnblogs.com/Images/OutliningIndicators/None.gifmysql> select ASCII(2);  
http://www.cnblogs.com/Images/OutliningIndicators/None.gif        -> 50  
http://www.cnblogs.com/Images/OutliningIndicators/None.gifmysql> select ASCII('dx');  
http://www.cnblogs.com/Images/OutliningIndicators/None.gif        -> 100也可参见ORD()函数。   
http://www.cnblogs.com/Images/OutliningIndicators/None.gif  
http://www.cnblogs.com/Images/OutliningIndicators/None.gifORD(str)   
http://www.cnblogs.com/Images/OutliningIndicators/None.gif如果字符串str最左面字符是一个多字节字符，通过以格式((first byte ASCII code)\*256+(second byte ASCII code))[\*256+third byte ASCII codehttp://www.cnblogs.com/Images/dot.gif]返回字符的ASCII代码值来返回多字节字符代码。如果最左面的字符不是一个多字节字符。返回与ASCII()函数返回的相同值。   
http://www.cnblogs.com/Images/OutliningIndicators/None.gifmysql> select ORD('2');  
http://www.cnblogs.com/Images/OutliningIndicators/None.gif        -> 50  
http://www.cnblogs.com/Images/OutliningIndicators/None.gif CONV(N,from\_base,to\_base)   
http://www.cnblogs.com/Images/OutliningIndicators/None.gif在不同的数字基之间变换数字。返回数字N的字符串数字，从from\_base基变换为to\_base基，如果任何参数是NULL，返回NULL。参数N解释为一个整数，但是可以指定为一个整数或一个字符串。最小基是2且最大的基是36。如果to\_base是一个负数，N被认为是一个有符号数，否则，N被当作无符号数。 CONV以64位点精度工作。   
http://www.cnblogs.com/Images/OutliningIndicators/None.gifmysql> select CONV("a",16,2);  
http://www.cnblogs.com/Images/OutliningIndicators/None.gif        -> '1010'  
http://www.cnblogs.com/Images/OutliningIndicators/None.gifmysql> select CONV("6E",18,8);  
http://www.cnblogs.com/Images/OutliningIndicators/None.gif        -> '172'  
http://www.cnblogs.com/Images/OutliningIndicators/None.gifmysql> select CONV(-17,10,-18);  
http://www.cnblogs.com/Images/OutliningIndicators/None.gif        -> '-H'  
http://www.cnblogs.com/Images/OutliningIndicators/None.gifmysql> select CONV(10+"10"+'10'+0xa,10,10);  
http://www.cnblogs.com/Images/OutliningIndicators/None.gif        -> '40'  
http://www.cnblogs.com/Images/OutliningIndicators/None.gif BIN(N)   
http://www.cnblogs.com/Images/OutliningIndicators/None.gif返回二进制值N的一个字符串表示，在此N是一个长整数(BIGINT)数字，这等价于CONV(N,10,2)。如果N是NULL，返回NULL。   
http://www.cnblogs.com/Images/OutliningIndicators/None.gifmysql> select BIN(12);  
http://www.cnblogs.com/Images/OutliningIndicators/None.gif        -> '1100'  
http://www.cnblogs.com/Images/OutliningIndicators/None.gifOCT(N)   
http://www.cnblogs.com/Images/OutliningIndicators/None.gif返回八进制值N的一个字符串的表示，在此N是一个长整型数字，这等价于CONV(N,10,8)。如果N是NULL，返回NULL。   
http://www.cnblogs.com/Images/OutliningIndicators/None.gifmysql> select OCT(12);  
http://www.cnblogs.com/Images/OutliningIndicators/None.gif        -> '14'  
http://www.cnblogs.com/Images/OutliningIndicators/None.gif HEX(N)   
http://www.cnblogs.com/Images/OutliningIndicators/None.gif返回十六进制值N一个字符串的表示，在此N是一个长整型(BIGINT)数字，这等价于CONV(N,10,16)。如果N是NULL，返回NULL。   
http://www.cnblogs.com/Images/OutliningIndicators/None.gifmysql> select HEX(255);  
http://www.cnblogs.com/Images/OutliningIndicators/None.gif        -> 'FF'  
http://www.cnblogs.com/Images/OutliningIndicators/None.gif CHAR(N,http://www.cnblogs.com/Images/dot.gif)   
http://www.cnblogs.com/Images/OutliningIndicators/None.gifCHAR()将参数解释为整数并且返回由这些整数的ASCII代码字符组成的一个字符串。NULL值被跳过。   
http://www.cnblogs.com/Images/OutliningIndicators/None.gifmysql> select CHAR(77,121,83,81,'76');  
http://www.cnblogs.com/Images/OutliningIndicators/None.gif        -> 'MySQL'  
http://www.cnblogs.com/Images/OutliningIndicators/None.gifmysql> select CHAR(77,77.3,'77.3');  
http://www.cnblogs.com/Images/OutliningIndicators/None.gif        -> 'MMM'  
http://www.cnblogs.com/Images/OutliningIndicators/None.gif CONCAT(str1,str2,http://www.cnblogs.com/Images/dot.gif)   
http://www.cnblogs.com/Images/OutliningIndicators/None.gif返回来自于参数连结的字符串。如果任何参数是NULL，返回NULL。可以有超过2个的参数。一个数字参数被变换为等价的字符串形式。   
http://www.cnblogs.com/Images/OutliningIndicators/None.gifmysql> select CONCAT('My', 'S', 'QL');  
http://www.cnblogs.com/Images/OutliningIndicators/None.gif        -> 'MySQL'  
http://www.cnblogs.com/Images/OutliningIndicators/None.gifmysql> select CONCAT('My', NULL, 'QL');  
http://www.cnblogs.com/Images/OutliningIndicators/None.gif        -> NULL  
http://www.cnblogs.com/Images/OutliningIndicators/None.gifmysql> select CONCAT(14.3);  
http://www.cnblogs.com/Images/OutliningIndicators/None.gif        -> '14.3'  
http://www.cnblogs.com/Images/OutliningIndicators/None.gifLENGTH(str)   
http://www.cnblogs.com/Images/OutliningIndicators/None.gif　   
http://www.cnblogs.com/Images/OutliningIndicators/None.gifOCTET\_LENGTH(str)   
http://www.cnblogs.com/Images/OutliningIndicators/None.gif　   
http://www.cnblogs.com/Images/OutliningIndicators/None.gifCHAR\_LENGTH(str)   
http://www.cnblogs.com/Images/OutliningIndicators/None.gif　   
http://www.cnblogs.com/Images/OutliningIndicators/None.gifCHARACTER\_LENGTH(str)   
http://www.cnblogs.com/Images/OutliningIndicators/None.gif返回字符串str的长度。   
http://www.cnblogs.com/Images/OutliningIndicators/None.gifmysql> select LENGTH('text');  
http://www.cnblogs.com/Images/OutliningIndicators/None.gif        -> 4  
http://www.cnblogs.com/Images/OutliningIndicators/None.gifmysql> select OCTET\_LENGTH('text');  
http://www.cnblogs.com/Images/OutliningIndicators/None.gif        -> 4  
http://www.cnblogs.com/Images/OutliningIndicators/None.gif注意，对于多字节字符，其CHAR\_LENGTH()仅计算一次。   
http://www.cnblogs.com/Images/OutliningIndicators/None.gif  
http://www.cnblogs.com/Images/OutliningIndicators/None.gifLOCATE(substr,str)   
http://www.cnblogs.com/Images/OutliningIndicators/None.gif　   
http://www.cnblogs.com/Images/OutliningIndicators/None.gifPOSITION(substr IN str)   
http://www.cnblogs.com/Images/OutliningIndicators/None.gif返回子串substr在字符串str第一个出现的位置，如果substr不是在str里面，返回0.   
http://www.cnblogs.com/Images/OutliningIndicators/None.gifmysql> select LOCATE('bar', 'foobarbar');  
http://www.cnblogs.com/Images/OutliningIndicators/None.gif        -> 4  
http://www.cnblogs.com/Images/OutliningIndicators/None.gifmysql> select LOCATE('xbar', 'foobar');  
http://www.cnblogs.com/Images/OutliningIndicators/None.gif        -> 0  
http://www.cnblogs.com/Images/OutliningIndicators/None.gif该函数是多字节可靠的。  LOCATE(substr,str,pos)   
http://www.cnblogs.com/Images/OutliningIndicators/None.gif返回子串substr在字符串str第一个出现的位置，从位置pos开始。如果substr不是在str里面，返回0。   
http://www.cnblogs.com/Images/OutliningIndicators/None.gifmysql> select LOCATE('bar', 'foobarbar',5);  
http://www.cnblogs.com/Images/OutliningIndicators/None.gif        -> 7  
http://www.cnblogs.com/Images/OutliningIndicators/None.gif这函数是多字节可靠的。   
http://www.cnblogs.com/Images/OutliningIndicators/None.gif  
http://www.cnblogs.com/Images/OutliningIndicators/None.gifINSTR(str,substr)   
http://www.cnblogs.com/Images/OutliningIndicators/None.gif返回子串substr在字符串str中的第一个出现的位置。这与有2个参数形式的LOCATE()相同，除了参数被颠倒。   
http://www.cnblogs.com/Images/OutliningIndicators/None.gifmysql> select INSTR('foobarbar', 'bar');  
http://www.cnblogs.com/Images/OutliningIndicators/None.gif        -> 4  
http://www.cnblogs.com/Images/OutliningIndicators/None.gifmysql> select INSTR('xbar', 'foobar');  
http://www.cnblogs.com/Images/OutliningIndicators/None.gif        -> 0  
http://www.cnblogs.com/Images/OutliningIndicators/None.gif这函数是多字节可靠的。   
http://www.cnblogs.com/Images/OutliningIndicators/None.gif  
http://www.cnblogs.com/Images/OutliningIndicators/None.gifLPAD(str,len,padstr)   
http://www.cnblogs.com/Images/OutliningIndicators/None.gif返回字符串str，左面用字符串padstr填补直到str是len个字符长。   
http://www.cnblogs.com/Images/OutliningIndicators/None.gifmysql> select LPAD('hi',4,'??');  
http://www.cnblogs.com/Images/OutliningIndicators/None.gif        -> '??hi'  
http://www.cnblogs.com/Images/OutliningIndicators/None.gif RPAD(str,len,padstr)   
http://www.cnblogs.com/Images/OutliningIndicators/None.gif返回字符串str，右面用字符串padstr填补直到str是len个字符长。     
http://www.cnblogs.com/Images/OutliningIndicators/None.gifmysql> select RPAD('hi',5,'?');  
http://www.cnblogs.com/Images/OutliningIndicators/None.gif        -> 'hi???'  
http://www.cnblogs.com/Images/OutliningIndicators/None.gifLEFT(str,len)   
http://www.cnblogs.com/Images/OutliningIndicators/None.gif返回字符串str的最左面len个字符。   
http://www.cnblogs.com/Images/OutliningIndicators/None.gifmysql> select LEFT('foobarbar', 5);  
http://www.cnblogs.com/Images/OutliningIndicators/None.gif        -> 'fooba'  
http://www.cnblogs.com/Images/OutliningIndicators/None.gif该函数是多字节可靠的。   
http://www.cnblogs.com/Images/OutliningIndicators/None.gif  
http://www.cnblogs.com/Images/OutliningIndicators/None.gifRIGHT(str,len)   
http://www.cnblogs.com/Images/OutliningIndicators/None.gif返回字符串str的最右面len个字符。   
http://www.cnblogs.com/Images/OutliningIndicators/None.gifmysql> select RIGHT('foobarbar', 4);  
http://www.cnblogs.com/Images/OutliningIndicators/None.gif        -> 'rbar'  
http://www.cnblogs.com/Images/OutliningIndicators/None.gif该函数是多字节可靠的。   
http://www.cnblogs.com/Images/OutliningIndicators/None.gif  
http://www.cnblogs.com/Images/OutliningIndicators/None.gifSUBSTRING(str,pos,len)   
http://www.cnblogs.com/Images/OutliningIndicators/None.gif　   
http://www.cnblogs.com/Images/OutliningIndicators/None.gifSUBSTRING(str FROM pos FOR len)   
http://www.cnblogs.com/Images/OutliningIndicators/None.gif　   
http://www.cnblogs.com/Images/OutliningIndicators/None.gifMID(str,pos,len)   
http://www.cnblogs.com/Images/OutliningIndicators/None.gif从字符串str返回一个len个字符的子串，从位置pos开始。使用FROM的变种形式是ANSI SQL92语法。   
http://www.cnblogs.com/Images/OutliningIndicators/None.gifmysql> select SUBSTRING('Quadratically',5,6);  
http://www.cnblogs.com/Images/OutliningIndicators/None.gif        -> 'ratica'  
http://www.cnblogs.com/Images/OutliningIndicators/None.gif该函数是多字节可靠的。   
http://www.cnblogs.com/Images/OutliningIndicators/None.gif  
http://www.cnblogs.com/Images/OutliningIndicators/None.gifSUBSTRING(str,pos)   
http://www.cnblogs.com/Images/OutliningIndicators/None.gif　   
http://www.cnblogs.com/Images/OutliningIndicators/None.gifSUBSTRING(str FROM pos)   
http://www.cnblogs.com/Images/OutliningIndicators/None.gif从字符串str的起始位置pos返回一个子串。   
http://www.cnblogs.com/Images/OutliningIndicators/None.gifmysql> select SUBSTRING('Quadratically',5);  
http://www.cnblogs.com/Images/OutliningIndicators/None.gif        -> 'ratically'  
http://www.cnblogs.com/Images/OutliningIndicators/None.gifmysql> select SUBSTRING('foobarbar' FROM 4);  
http://www.cnblogs.com/Images/OutliningIndicators/None.gif        -> 'barbar'  
http://www.cnblogs.com/Images/OutliningIndicators/None.gif该函数是多字节可靠的。   
http://www.cnblogs.com/Images/OutliningIndicators/None.gif  
http://www.cnblogs.com/Images/OutliningIndicators/None.gifSUBSTRING\_INDEX(str,delim,count)   
http://www.cnblogs.com/Images/OutliningIndicators/None.gif返回从字符串str的第count个出现的分隔符delim之后的子串。如果count是正数，返回最后的分隔符到左边(从左边数) 的所有字符。如果count是负数，返回最后的分隔符到右边的所有字符(从右边数)。   
http://www.cnblogs.com/Images/OutliningIndicators/None.gifmysql> select SUBSTRING\_INDEX('www.mysql.com', '.', 2);  
http://www.cnblogs.com/Images/OutliningIndicators/None.gif        -> 'www.mysql'  
http://www.cnblogs.com/Images/OutliningIndicators/None.gifmysql> select SUBSTRING\_INDEX('www.mysql.com', '.', -2);  
http://www.cnblogs.com/Images/OutliningIndicators/None.gif        -> 'mysql.com'  
http://www.cnblogs.com/Images/OutliningIndicators/None.gif该函数对多字节是可靠的。   
http://www.cnblogs.com/Images/OutliningIndicators/None.gif  
http://www.cnblogs.com/Images/OutliningIndicators/None.gifLTRIM(str)   
http://www.cnblogs.com/Images/OutliningIndicators/None.gif返回删除了其前置空格字符的字符串str。   
http://www.cnblogs.com/Images/OutliningIndicators/None.gifmysql> select LTRIM('  barbar');  
http://www.cnblogs.com/Images/OutliningIndicators/None.gif        -> 'barbar'  
http://www.cnblogs.com/Images/OutliningIndicators/None.gifRTRIM(str)   
http://www.cnblogs.com/Images/OutliningIndicators/None.gif返回删除了其拖后空格字符的字符串str。   
http://www.cnblogs.com/Images/OutliningIndicators/None.gifmysql> select RTRIM('barbar   ');  
http://www.cnblogs.com/Images/OutliningIndicators/None.gif        -> 'barbar'  
http://www.cnblogs.com/Images/OutliningIndicators/None.gif该函数对多字节是可靠的。  TRIM([[BOTH | LEADING | TRAILING] [remstr] FROM] str)   
http://www.cnblogs.com/Images/OutliningIndicators/None.gif返回字符串str，其所有remstr前缀或后缀被删除了。如果没有修饰符BOTH、LEADING或TRAILING给出，BOTH被假定。如果remstr没被指定，空格被删除。   
http://www.cnblogs.com/Images/OutliningIndicators/None.gifmysql> select TRIM('  bar   ');  
http://www.cnblogs.com/Images/OutliningIndicators/None.gif        -> 'bar'  
http://www.cnblogs.com/Images/OutliningIndicators/None.gifmysql> select TRIM(LEADING 'x' FROM 'xxxbarxxx');  
http://www.cnblogs.com/Images/OutliningIndicators/None.gif        -> 'barxxx'  
http://www.cnblogs.com/Images/OutliningIndicators/None.gifmysql> select TRIM(BOTH 'x' FROM 'xxxbarxxx');  
http://www.cnblogs.com/Images/OutliningIndicators/None.gif        -> 'bar'  
http://www.cnblogs.com/Images/OutliningIndicators/None.gifmysql> select TRIM(TRAILING 'xyz' FROM 'barxxyz');  
http://www.cnblogs.com/Images/OutliningIndicators/None.gif        -> 'barx'  
http://www.cnblogs.com/Images/OutliningIndicators/None.gif该函数对多字节是可靠的。   
http://www.cnblogs.com/Images/OutliningIndicators/None.gif  
http://www.cnblogs.com/Images/OutliningIndicators/None.gifSOUNDEX(str)   
http://www.cnblogs.com/Images/OutliningIndicators/None.gif返回str的一个同音字符串。听起来“大致相同”的2个字符串应该有相同的同音字符串。一个“标准”的同音字符串长是4个字符，但是SOUNDEX()函数返回一个任意长的字符串。你可以在结果上使用SUBSTRING()得到一个“标准”的 同音串。所有非数字字母字符在给定的字符串中被忽略。所有在A-Z之外的字符国际字母被当作元音。   
http://www.cnblogs.com/Images/OutliningIndicators/None.gifmysql> select SOUNDEX('Hello');  
http://www.cnblogs.com/Images/OutliningIndicators/None.gif        -> 'H400'  
http://www.cnblogs.com/Images/OutliningIndicators/None.gifmysql> select SOUNDEX('Quadratically');  
http://www.cnblogs.com/Images/OutliningIndicators/None.gif        -> 'Q36324'  
http://www.cnblogs.com/Images/OutliningIndicators/None.gif SPACE(N)   
http://www.cnblogs.com/Images/OutliningIndicators/None.gif返回由N个空格字符组成的一个字符串。   
http://www.cnblogs.com/Images/OutliningIndicators/None.gifmysql> select SPACE(6);  
http://www.cnblogs.com/Images/OutliningIndicators/None.gif        -> '      '  
http://www.cnblogs.com/Images/OutliningIndicators/None.gif REPLACE(str,from\_str,to\_str)   
http://www.cnblogs.com/Images/OutliningIndicators/None.gif返回字符串str，其字符串from\_str的所有出现由字符串to\_str代替。   
http://www.cnblogs.com/Images/OutliningIndicators/None.gifmysql> select REPLACE('www.mysql.com', 'w', 'Ww');  
http://www.cnblogs.com/Images/OutliningIndicators/None.gif        -> 'WwWwWw.mysql.com'  
http://www.cnblogs.com/Images/OutliningIndicators/None.gif该函数对多字节是可靠的。   
http://www.cnblogs.com/Images/OutliningIndicators/None.gif  
http://www.cnblogs.com/Images/OutliningIndicators/None.gifREPEAT(str,count)   
http://www.cnblogs.com/Images/OutliningIndicators/None.gif返回由重复countTimes次的字符串str组成的一个字符串。如果count <= 0，返回一个空字符串。如果str或count是NULL，返回NULL。   
http://www.cnblogs.com/Images/OutliningIndicators/None.gifmysql> select REPEAT('MySQL', 3);  
http://www.cnblogs.com/Images/OutliningIndicators/None.gif        -> 'MySQLMySQLMySQL'  
http://www.cnblogs.com/Images/OutliningIndicators/None.gif REVERSE(str)   
http://www.cnblogs.com/Images/OutliningIndicators/None.gif返回颠倒字符顺序的字符串str。   
http://www.cnblogs.com/Images/OutliningIndicators/None.gifmysql> select REVERSE('abc');  
http://www.cnblogs.com/Images/OutliningIndicators/None.gif        -> 'cba'  
http://www.cnblogs.com/Images/OutliningIndicators/None.gif该函数对多字节可靠的。   
http://www.cnblogs.com/Images/OutliningIndicators/None.gif  
http://www.cnblogs.com/Images/OutliningIndicators/None.gifINSERT(str,pos,len,newstr)   
http://www.cnblogs.com/Images/OutliningIndicators/None.gif返回字符串str，在位置pos起始的子串且len个字符长得子串由字符串newstr代替。   
http://www.cnblogs.com/Images/OutliningIndicators/None.gifmysql> select INSERT('Quadratic', 3, 4, 'What');  
http://www.cnblogs.com/Images/OutliningIndicators/None.gif        -> 'QuWhattic'  
http://www.cnblogs.com/Images/OutliningIndicators/None.gif该函数对多字节是可靠的。   
http://www.cnblogs.com/Images/OutliningIndicators/None.gif  
http://www.cnblogs.com/Images/OutliningIndicators/None.gifELT(N,str1,str2,str3,http://www.cnblogs.com/Images/dot.gif)   
http://www.cnblogs.com/Images/OutliningIndicators/None.gif如果N= 1，返回str1，如果N= 2，返回str2，等等。如果N小于1或大于参数个数，返回NULL。ELT()是FIELD()反运算。   
http://www.cnblogs.com/Images/OutliningIndicators/None.gifmysql> select ELT(1, 'ej', 'Heja', 'hej', 'foo');  
http://www.cnblogs.com/Images/OutliningIndicators/None.gif        -> 'ej'  
http://www.cnblogs.com/Images/OutliningIndicators/None.gifmysql> select ELT(4, 'ej', 'Heja', 'hej', 'foo');  
http://www.cnblogs.com/Images/OutliningIndicators/None.gif        -> 'foo'  
http://www.cnblogs.com/Images/OutliningIndicators/None.gifFIELD(str,str1,str2,str3,http://www.cnblogs.com/Images/dot.gif)   
http://www.cnblogs.com/Images/OutliningIndicators/None.gif返回str在str1, str2, str3, http://www.cnblogs.com/Images/dot.gif清单的索引。如果str没找到，返回0。FIELD()是ELT()反运算。   
http://www.cnblogs.com/Images/OutliningIndicators/None.gifmysql> select FIELD('ej', 'Hej', 'ej', 'Heja', 'hej', 'foo');  
http://www.cnblogs.com/Images/OutliningIndicators/None.gif        -> 2  
http://www.cnblogs.com/Images/OutliningIndicators/None.gifmysql> select FIELD('fo', 'Hej', 'ej', 'Heja', 'hej', 'foo');  
http://www.cnblogs.com/Images/OutliningIndicators/None.gif        -> 0  
http://www.cnblogs.com/Images/OutliningIndicators/None.gifFIND\_IN\_SET(str,strlist)   
http://www.cnblogs.com/Images/OutliningIndicators/None.gif如果字符串str在由N子串组成的表strlist之中，返回一个1到N的值。一个字符串表是被“,”分隔的子串组成的一个字符串。如果第一个参数是一个常数字符串并且第二个参数是一种类型为SET的列，FIND\_IN\_SET()函数被优化而使用位运算！如果str不是在strlist里面或如果strlist是空字符串，返回0。如果任何一个参数是NULL，返回NULL。如果第一个参数包含一个“,”，该函数将工作不正常。   
http://www.cnblogs.com/Images/OutliningIndicators/None.gifmysql> SELECT FIND\_IN\_SET('b','a,b,c,d');  
http://www.cnblogs.com/Images/OutliningIndicators/None.gif        -> 2  
http://www.cnblogs.com/Images/OutliningIndicators/None.gif MAKE\_SET(bits,str1,str2,http://www.cnblogs.com/Images/dot.gif)   
http://www.cnblogs.com/Images/OutliningIndicators/None.gif返回一个集合 (包含由“,”字符分隔的子串组成的一个字符串)，由相应的位在bits集合中的的字符串组成。str1对应于位0，str2对应位1，等等。在str1, str2, http://www.cnblogs.com/Images/dot.gif中的NULL串不添加到结果中。   
http://www.cnblogs.com/Images/OutliningIndicators/None.gifmysql> SELECT MAKE\_SET(1,'a','b','c');  
http://www.cnblogs.com/Images/OutliningIndicators/None.gif        -> 'a'  
http://www.cnblogs.com/Images/OutliningIndicators/None.gifmysql> SELECT MAKE\_SET(1 | 4,'hello','nice','world');  
http://www.cnblogs.com/Images/OutliningIndicators/None.gif        -> 'hello,world'  
http://www.cnblogs.com/Images/OutliningIndicators/None.gifmysql> SELECT MAKE\_SET(0,'a','b','c');  
http://www.cnblogs.com/Images/OutliningIndicators/None.gif        -> ''  
http://www.cnblogs.com/Images/OutliningIndicators/None.gifEXPORT\_SET(bits,on,off,[separator,[number\_of\_bits]])   
http://www.cnblogs.com/Images/OutliningIndicators/None.gif返回一个字符串，在这里对于在“bits”中设定每一位，你得到一个“on”字符串，并且对于每个复位(reset)的位，你得到一个“off”字符串。每个字符串用“separator”分隔(缺省“,”)，并且只有“bits”的“number\_of\_bits” (缺省64)位被使用。   
http://www.cnblogs.com/Images/OutliningIndicators/None.gifmysql> select EXPORT\_SET(5,'Y','N',',',4)  
http://www.cnblogs.com/Images/OutliningIndicators/None.gif        -> Y,N,Y,N   
http://www.cnblogs.com/Images/OutliningIndicators/None.gifLCASE(str)   
http://www.cnblogs.com/Images/OutliningIndicators/None.gif　   
http://www.cnblogs.com/Images/OutliningIndicators/None.gifLOWER(str)   
http://www.cnblogs.com/Images/OutliningIndicators/None.gif返回字符串str，根据当前字符集映射(缺省是ISO-8859-1 Latin1)把所有的字符改变成小写。该函数对多字节是可靠的。   
http://www.cnblogs.com/Images/OutliningIndicators/None.gifmysql> select LCASE('QUADRATICALLY');  
http://www.cnblogs.com/Images/OutliningIndicators/None.gif        -> 'quadratically'  
http://www.cnblogs.com/Images/OutliningIndicators/None.gif UCASE(str)   
http://www.cnblogs.com/Images/OutliningIndicators/None.gif　   
http://www.cnblogs.com/Images/OutliningIndicators/None.gifUPPER(str)   
http://www.cnblogs.com/Images/OutliningIndicators/None.gif返回字符串str，根据当前字符集映射(缺省是ISO-8859-1 Latin1)把所有的字符改变成大写。该函数对多字节是可靠的。   
http://www.cnblogs.com/Images/OutliningIndicators/None.gifmysql> select UCASE('Hej');  
http://www.cnblogs.com/Images/OutliningIndicators/None.gif        -> 'HEJ'  
http://www.cnblogs.com/Images/OutliningIndicators/None.gif该函数对多字节是可靠的。   
http://www.cnblogs.com/Images/OutliningIndicators/None.gif  
http://www.cnblogs.com/Images/OutliningIndicators/None.gifLOAD\_FILE(file\_name)   
http://www.cnblogs.com/Images/OutliningIndicators/None.gif读入文件并且作为一个字符串返回文件内容。文件必须在服务器上，你必须指定到文件的完整路径名，而且你必须有file权限。文件必须所有内容都是可读的并且小于max\_allowed\_packet。如果文件不存在或由于上面原因之一不能被读出，函数返回NULL。   
http://www.cnblogs.com/Images/OutliningIndicators/None.gifmysql> UPDATE table\_name  
http://www.cnblogs.com/Images/OutliningIndicators/None.gif           SET blob\_column=LOAD\_FILE("/tmp/picture")  
http://www.cnblogs.com/Images/OutliningIndicators/None.gif           WHERE id=1;  
http://www.cnblogs.com/Images/OutliningIndicators/None.gif  
http://www.cnblogs.com/Images/OutliningIndicators/None.gifMySQL必要时自动变换数字为字符串，并且反过来也如此： mysql> SELECT 1+"1";  
http://www.cnblogs.com/Images/OutliningIndicators/None.gif        -> 2  
http://www.cnblogs.com/Images/OutliningIndicators/None.gifmysql> SELECT CONCAT(2,' test');  
http://www.cnblogs.com/Images/OutliningIndicators/None.gif        -> '2 test'  
http://www.cnblogs.com/Images/OutliningIndicators/None.gif如果你想要明确地变换一个数字到一个字符串，把它作为参数传递到CONCAT()。 如果字符串函数提供一个二进制字符串作为参数，结果字符串也是一个二进制字符串。被变换到一个字符串的数字被当作是一个二进制字符串。这仅影响比较。   日期和时间函数   对于每个类型拥有的值范围以及并且指定日期何时间值的有效格式的描述见7.3.6 日期和时间类型。 这里是一个使用日期函数的例子。下面的查询选择了所有记录，其date\_col的值是在最后30天以内： mysql> SELECT something FROM table  
http://www.cnblogs.com/Images/OutliningIndicators/None.gif           WHERE TO\_DAYS(NOW()) - TO\_DAYS(date\_col) <= 30;  
http://www.cnblogs.com/Images/OutliningIndicators/None.gifDAYOFWEEK(date)   
http://www.cnblogs.com/Images/OutliningIndicators/None.gif返回日期date的星期索引(1=星期天，2=星期一, ……7=星期六)。这些索引值对应于ODBC标准。   
http://www.cnblogs.com/Images/OutliningIndicators/None.gifmysql> select DAYOFWEEK('1998-02-03');  
http://www.cnblogs.com/Images/OutliningIndicators/None.gif        -> 3  
http://www.cnblogs.com/Images/OutliningIndicators/None.gif WEEKDAY(date)   
http://www.cnblogs.com/Images/OutliningIndicators/None.gif返回date的星期索引(0=星期一，1=星期二, ……6= 星期天)。   
http://www.cnblogs.com/Images/OutliningIndicators/None.gifmysql> select WEEKDAY('1997-10-04 22:23:00');  
http://www.cnblogs.com/Images/OutliningIndicators/None.gif        -> 5  
http://www.cnblogs.com/Images/OutliningIndicators/None.gifmysql> select WEEKDAY('1997-11-05');  
http://www.cnblogs.com/Images/OutliningIndicators/None.gif        -> 2  
http://www.cnblogs.com/Images/OutliningIndicators/None.gif DAYOFMONTH(date)   
http://www.cnblogs.com/Images/OutliningIndicators/None.gif返回date的月份中日期，在1到31范围内。   
http://www.cnblogs.com/Images/OutliningIndicators/None.gifmysql> select DAYOFMONTH('1998-02-03');  
http://www.cnblogs.com/Images/OutliningIndicators/None.gif        -> 3  
http://www.cnblogs.com/Images/OutliningIndicators/None.gifDAYOFYEAR(date)   
http://www.cnblogs.com/Images/OutliningIndicators/None.gif返回date在一年中的日数, 在1到366范围内。   
http://www.cnblogs.com/Images/OutliningIndicators/None.gifmysql> select DAYOFYEAR('1998-02-03');  
http://www.cnblogs.com/Images/OutliningIndicators/None.gif        -> 34  
http://www.cnblogs.com/Images/OutliningIndicators/None.gifMONTH(date)   
http://www.cnblogs.com/Images/OutliningIndicators/None.gif返回date的月份，范围1到12。   
http://www.cnblogs.com/Images/OutliningIndicators/None.gifmysql> select MONTH('1998-02-03');  
http://www.cnblogs.com/Images/OutliningIndicators/None.gif        -> 2  
http://www.cnblogs.com/Images/OutliningIndicators/None.gif DAYNAME(date)   
http://www.cnblogs.com/Images/OutliningIndicators/None.gif返回date的星期名字。   
http://www.cnblogs.com/Images/OutliningIndicators/None.gifmysql> select DAYNAME("1998-02-05");  
http://www.cnblogs.com/Images/OutliningIndicators/None.gif        -> 'Thursday'  
http://www.cnblogs.com/Images/OutliningIndicators/None.gifMONTHNAME(date)   
http://www.cnblogs.com/Images/OutliningIndicators/None.gif返回date的月份名字。   
http://www.cnblogs.com/Images/OutliningIndicators/None.gifmysql> select MONTHNAME("1998-02-05");  
http://www.cnblogs.com/Images/OutliningIndicators/None.gif        -> 'February'  
http://www.cnblogs.com/Images/OutliningIndicators/None.gif QUARTER(date)   
http://www.cnblogs.com/Images/OutliningIndicators/None.gif返回date一年中的季度，范围1到4。   
http://www.cnblogs.com/Images/OutliningIndicators/None.gifmysql> select QUARTER('98-04-01');  
http://www.cnblogs.com/Images/OutliningIndicators/None.gif        -> 2  
http://www.cnblogs.com/Images/OutliningIndicators/None.gif WEEK(date)   
http://www.cnblogs.com/Images/OutliningIndicators/None.gif　   
http://www.cnblogs.com/Images/OutliningIndicators/None.gifWEEK(date,first)   
http://www.cnblogs.com/Images/OutliningIndicators/None.gif对于星期天是一周的第一天的地方，有一个单个参数，返回date的周数，范围在0到52。2个参数形式WEEK()允许你指定星期是否开始于星期天或星期一。如果第二个参数是0，星期从星期天开始，如果第二个参数是1，从星期一开始。   
http://www.cnblogs.com/Images/OutliningIndicators/None.gifmysql> select WEEK('1998-02-20');  
http://www.cnblogs.com/Images/OutliningIndicators/None.gif        -> 7  
http://www.cnblogs.com/Images/OutliningIndicators/None.gifmysql> select WEEK('1998-02-20',0);  
http://www.cnblogs.com/Images/OutliningIndicators/None.gif        -> 7  
http://www.cnblogs.com/Images/OutliningIndicators/None.gifmysql> select WEEK('1998-02-20',1);  
http://www.cnblogs.com/Images/OutliningIndicators/None.gif        -> 8  
http://www.cnblogs.com/Images/OutliningIndicators/None.gifYEAR(date)   
http://www.cnblogs.com/Images/OutliningIndicators/None.gif返回date的年份，范围在1000到9999。   
http://www.cnblogs.com/Images/OutliningIndicators/None.gifmysql> select YEAR('98-02-03');  
http://www.cnblogs.com/Images/OutliningIndicators/None.gif        -> 1998  
http://www.cnblogs.com/Images/OutliningIndicators/None.gifHOUR(time)   
http://www.cnblogs.com/Images/OutliningIndicators/None.gif返回time的小时，范围是0到23。   
http://www.cnblogs.com/Images/OutliningIndicators/None.gifmysql> select HOUR('10:05:03');  
http://www.cnblogs.com/Images/OutliningIndicators/None.gif        -> 10  
http://www.cnblogs.com/Images/OutliningIndicators/None.gifMINUTE(time)   
http://www.cnblogs.com/Images/OutliningIndicators/None.gif返回time的分钟，范围是0到59。   
http://www.cnblogs.com/Images/OutliningIndicators/None.gifmysql> select MINUTE('98-02-03 10:05:03');  
http://www.cnblogs.com/Images/OutliningIndicators/None.gif        -> 5  
http://www.cnblogs.com/Images/OutliningIndicators/None.gifSECOND(time)   
http://www.cnblogs.com/Images/OutliningIndicators/None.gif回来time的秒数，范围是0到59。   
http://www.cnblogs.com/Images/OutliningIndicators/None.gifmysql> select SECOND('10:05:03');  
http://www.cnblogs.com/Images/OutliningIndicators/None.gif        -> 3  
http://www.cnblogs.com/Images/OutliningIndicators/None.gifPERIOD\_ADD(P,N)   
http://www.cnblogs.com/Images/OutliningIndicators/None.gif增加N个月到阶段P（以格式YYMM或YYYYMM)。以格式YYYYMM返回值。注意阶段参数P不是日期值。   
http://www.cnblogs.com/Images/OutliningIndicators/None.gifmysql> select PERIOD\_ADD(9801,2);  
http://www.cnblogs.com/Images/OutliningIndicators/None.gif        -> 199803  
http://www.cnblogs.com/Images/OutliningIndicators/None.gifPERIOD\_DIFF(P1,P2)   
http://www.cnblogs.com/Images/OutliningIndicators/None.gif返回在时期P1和P2之间月数，P1和P2应该以格式YYMM或YYYYMM。注意，时期参数P1和P2不是日期值。   
http://www.cnblogs.com/Images/OutliningIndicators/None.gifmysql> select PERIOD\_DIFF(9802,199703);  
http://www.cnblogs.com/Images/OutliningIndicators/None.gif        -> 11  
http://www.cnblogs.com/Images/OutliningIndicators/None.gif    DATE\_ADD(date,INTERVAL expr type)   
http://www.cnblogs.com/Images/OutliningIndicators/None.gif　   
http://www.cnblogs.com/Images/OutliningIndicators/None.gifDATE\_SUB(date,INTERVAL expr type)   
http://www.cnblogs.com/Images/OutliningIndicators/None.gif　   
http://www.cnblogs.com/Images/OutliningIndicators/None.gifADDDATE(date,INTERVAL expr type)   
http://www.cnblogs.com/Images/OutliningIndicators/None.gif　   
http://www.cnblogs.com/Images/OutliningIndicators/None.gifSUBDATE(date,INTERVAL expr type)   
http://www.cnblogs.com/Images/OutliningIndicators/None.gif这些功能执行日期运算。对于MySQL 3.22，他们是新的。ADDDATE()和SUBDATE()是DATE\_ADD()和DATE\_SUB()的同义词。在MySQL 3.23中，你可以使用+和-而不是DATE\_ADD()和DATE\_SUB()。（见例子）date是一个指定开始日期的DATETIME或DATE值，expr是指定加到开始日期或从开始日期减去的间隔值一个表达式，expr是一个字符串；它可以以一个“-”开始表示负间隔。type是一个关键词，指明表达式应该如何被解释。EXTRACT(type FROM date)函数从日期中返回“type”间隔。下表显示了type和expr参数怎样被关联： type值  含义  期望的expr格式    
http://www.cnblogs.com/Images/OutliningIndicators/None.gifSECOND  秒 SECONDS    
http://www.cnblogs.com/Images/OutliningIndicators/None.gifMINUTE  分钟 MINUTES    
http://www.cnblogs.com/Images/OutliningIndicators/None.gifHOUR  时间 HOURS    
http://www.cnblogs.com/Images/OutliningIndicators/None.gifDAY  天 DAYS    
http://www.cnblogs.com/Images/OutliningIndicators/None.gifMONTH  月 MONTHS    
http://www.cnblogs.com/Images/OutliningIndicators/None.gifYEAR  年 YEARS    
http://www.cnblogs.com/Images/OutliningIndicators/None.gifMINUTE\_SECOND  分钟和秒 "MINUTES:SECONDS"    
http://www.cnblogs.com/Images/OutliningIndicators/None.gifHOUR\_MINUTE  小时和分钟 "HOURS:MINUTES"    
http://www.cnblogs.com/Images/OutliningIndicators/None.gifDAY\_HOUR  天和小时 "DAYS HOURS"    
http://www.cnblogs.com/Images/OutliningIndicators/None.gifYEAR\_MONTH  年和月 "YEARS-MONTHS"    
http://www.cnblogs.com/Images/OutliningIndicators/None.gifHOUR\_SECOND  小时, 分钟， "HOURS:MINUTES:SECONDS"    
http://www.cnblogs.com/Images/OutliningIndicators/None.gifDAY\_MINUTE  天, 小时, 分钟 "DAYS HOURS:MINUTES"    
http://www.cnblogs.com/Images/OutliningIndicators/None.gifDAY\_SECOND  天, 小时, 分钟, 秒 "DAYS HOURS:MINUTES:SECONDS"    
http://www.cnblogs.com/Images/OutliningIndicators/None.gif  
http://www.cnblogs.com/Images/OutliningIndicators/None.gifMySQL在expr格式中允许任何标点分隔符。表示显示的是建议的分隔符。如果date参数是一个DATE值并且你的计算仅仅包含YEAR、MONTH和DAY部分(即，没有时间部分)，结果是一个DATE值。否则结果是一个DATETIME值。   
http://www.cnblogs.com/Images/OutliningIndicators/None.gif  
http://www.cnblogs.com/Images/OutliningIndicators/None.gifmysql> SELECT "1997-12-31 23:59:59" + INTERVAL 1 SECOND;  
http://www.cnblogs.com/Images/OutliningIndicators/None.gif        -> 1998-01-01 00:00:00  
http://www.cnblogs.com/Images/OutliningIndicators/None.gifmysql> SELECT INTERVAL 1 DAY + "1997-12-31";  
http://www.cnblogs.com/Images/OutliningIndicators/None.gif        -> 1998-01-01  
http://www.cnblogs.com/Images/OutliningIndicators/None.gifmysql> SELECT "1998-01-01" - INTERVAL 1 SECOND;  
http://www.cnblogs.com/Images/OutliningIndicators/None.gif       -> 1997-12-31 23:59:59   
http://www.cnblogs.com/Images/OutliningIndicators/None.gifmysql> SELECT DATE\_ADD("1997-12-31 23:59:59",  
http://www.cnblogs.com/Images/OutliningIndicators/None.gif                       INTERVAL 1 SECOND);  
http://www.cnblogs.com/Images/OutliningIndicators/None.gif        -> 1998-01-01 00:00:00  
http://www.cnblogs.com/Images/OutliningIndicators/None.gifmysql> SELECT DATE\_ADD("1997-12-31 23:59:59",  
http://www.cnblogs.com/Images/OutliningIndicators/None.gif                       INTERVAL 1 DAY);  
http://www.cnblogs.com/Images/OutliningIndicators/None.gif        -> 1998-01-01 23:59:59  
http://www.cnblogs.com/Images/OutliningIndicators/None.gifmysql> SELECT DATE\_ADD("1997-12-31 23:59:59",  
http://www.cnblogs.com/Images/OutliningIndicators/None.gif                       INTERVAL "1:1" MINUTE\_SECOND);  
http://www.cnblogs.com/Images/OutliningIndicators/None.gif        -> 1998-01-01 00:01:00  
http://www.cnblogs.com/Images/OutliningIndicators/None.gifmysql> SELECT DATE\_SUB("1998-01-01 00:00:00",  
http://www.cnblogs.com/Images/OutliningIndicators/None.gif                       INTERVAL "1 1:1:1" DAY\_SECOND);  
http://www.cnblogs.com/Images/OutliningIndicators/None.gif        -> 1997-12-30 22:58:59  
http://www.cnblogs.com/Images/OutliningIndicators/None.gifmysql> SELECT DATE\_ADD("1998-01-01 00:00:00",  
http://www.cnblogs.com/Images/OutliningIndicators/None.gif                       INTERVAL "-1 10" DAY\_HOUR);  
http://www.cnblogs.com/Images/OutliningIndicators/None.gif        -> 1997-12-30 14:00:00  
http://www.cnblogs.com/Images/OutliningIndicators/None.gifmysql> SELECT DATE\_SUB("1998-01-02", INTERVAL 31 DAY);  
http://www.cnblogs.com/Images/OutliningIndicators/None.gif        -> 1997-12-02  
http://www.cnblogs.com/Images/OutliningIndicators/None.gifmysql> SELECT EXTRACT(YEAR FROM "1999-07-02");  
http://www.cnblogs.com/Images/OutliningIndicators/None.gif       -> 1999  
http://www.cnblogs.com/Images/OutliningIndicators/None.gifmysql> SELECT EXTRACT(YEAR\_MONTH FROM "1999-07-02 01:02:03");  
http://www.cnblogs.com/Images/OutliningIndicators/None.gif       -> 199907  
http://www.cnblogs.com/Images/OutliningIndicators/None.gifmysql> SELECT EXTRACT(DAY\_MINUTE FROM "1999-07-02 01:02:03");  
http://www.cnblogs.com/Images/OutliningIndicators/None.gif       -> 20102  
http://www.cnblogs.com/Images/OutliningIndicators/None.gif如果你指定太短的间隔值(不包括type关键词期望的间隔部分)，MySQL假设你省掉了间隔值的最左面部分。例如，如果你指定一个type是DAY\_SECOND，值expr被希望有天、小时、分钟和秒部分。如果你象"1:10"这样指定值，MySQL假设日子和小时部分是丢失的并且值代表分钟和秒。换句话说，"1:10" DAY\_SECOND以它等价于"1:10" MINUTE\_SECOND的方式解释，这对那MySQL解释TIME值表示经过的时间而非作为一天的时间的方式有二义性。如果你使用确实不正确的日期，结果是NULL。如果你增加MONTH、YEAR\_MONTH或YEAR并且结果日期大于新月份的最大值天数，日子在新月用最大的天调整。   
http://www.cnblogs.com/Images/OutliningIndicators/None.gif  
http://www.cnblogs.com/Images/OutliningIndicators/None.gifmysql> select DATE\_ADD('1998-01-30', Interval 1 month);  
http://www.cnblogs.com/Images/OutliningIndicators/None.gif        -> 1998-02-28  
http://www.cnblogs.com/Images/OutliningIndicators/None.gif注意，从前面的例子中词INTERVAL和type关键词不是区分大小写的。  TO\_DAYS(date)   
http://www.cnblogs.com/Images/OutliningIndicators/None.gif给出一个日期date，返回一个天数(从0年的天数)。   
http://www.cnblogs.com/Images/OutliningIndicators/None.gifmysql> select TO\_DAYS(950501);  
http://www.cnblogs.com/Images/OutliningIndicators/None.gif        -> 728779  
http://www.cnblogs.com/Images/OutliningIndicators/None.gifmysql> select TO\_DAYS('1997-10-07');  
http://www.cnblogs.com/Images/OutliningIndicators/None.gif        -> 729669  
http://www.cnblogs.com/Images/OutliningIndicators/None.gifTO\_DAYS()不打算用于使用格列高里历(1582)出现前的值。   
http://www.cnblogs.com/Images/OutliningIndicators/None.gif  
http://www.cnblogs.com/Images/OutliningIndicators/None.gifFROM\_DAYS(N)   
http://www.cnblogs.com/Images/OutliningIndicators/None.gif给出一个天数N，返回一个DATE值。   
http://www.cnblogs.com/Images/OutliningIndicators/None.gifmysql> select FROM\_DAYS(729669);  
http://www.cnblogs.com/Images/OutliningIndicators/None.gif        -> '1997-10-07'  
http://www.cnblogs.com/Images/OutliningIndicators/None.gifTO\_DAYS()不打算用于使用格列高里历(1582)出现前的值。   
http://www.cnblogs.com/Images/OutliningIndicators/None.gif  
http://www.cnblogs.com/Images/OutliningIndicators/None.gifDATE\_FORMAT(date,format)   
http://www.cnblogs.com/Images/OutliningIndicators/None.gif根据format字符串格式化date值。下列修饰符可以被用在format字符串中： %M  月名字(January……December)    
http://www.cnblogs.com/Images/OutliningIndicators/None.gif%W  星期名字(Sunday……Saturday)    
http://www.cnblogs.com/Images/OutliningIndicators/None.gif%D  有英语前缀的月份的日期(1st, 2nd, 3rd, 等等。）   
http://www.cnblogs.com/Images/OutliningIndicators/None.gif%Y  年, 数字, 4 位   
http://www.cnblogs.com/Images/OutliningIndicators/None.gif%y  年, 数字, 2 位   
http://www.cnblogs.com/Images/OutliningIndicators/None.gif%a  缩写的星期名字(Sun……Sat)    
http://www.cnblogs.com/Images/OutliningIndicators/None.gif%d  月份中的天数, 数字(00……31)    
http://www.cnblogs.com/Images/OutliningIndicators/None.gif%e  月份中的天数, 数字(0……31)    
http://www.cnblogs.com/Images/OutliningIndicators/None.gif%m  月, 数字(01……12)    
http://www.cnblogs.com/Images/OutliningIndicators/None.gif%c  月, 数字(1……12)    
http://www.cnblogs.com/Images/OutliningIndicators/None.gif%b  缩写的月份名字(Jan……Dec)    
http://www.cnblogs.com/Images/OutliningIndicators/None.gif%j  一年中的天数(001……366)    
http://www.cnblogs.com/Images/OutliningIndicators/None.gif%H  小时(00……23)    
http://www.cnblogs.com/Images/OutliningIndicators/None.gif%k  小时(0……23)    
http://www.cnblogs.com/Images/OutliningIndicators/None.gif%h  小时(01……12)    
http://www.cnblogs.com/Images/OutliningIndicators/None.gif%I  小时(01……12)    
http://www.cnblogs.com/Images/OutliningIndicators/None.gif%l  小时(1……12)    
http://www.cnblogs.com/Images/OutliningIndicators/None.gif%i  分钟, 数字(00……59)    
http://www.cnblogs.com/Images/OutliningIndicators/None.gif%r  时间,12 小时(hh:mm:ss [AP]M)    
http://www.cnblogs.com/Images/OutliningIndicators/None.gif%T  时间,24 小时(hh:mm:ss)    
http://www.cnblogs.com/Images/OutliningIndicators/None.gif%S  秒(00……59)    
http://www.cnblogs.com/Images/OutliningIndicators/None.gif%s  秒(00……59)    
http://www.cnblogs.com/Images/OutliningIndicators/None.gif%p  AM或PM    
http://www.cnblogs.com/Images/OutliningIndicators/None.gif%w  一个星期中的天数(0=Sunday ……6=Saturday ）   
http://www.cnblogs.com/Images/OutliningIndicators/None.gif%U  星期(0……52), 这里星期天是星期的第一天   
http://www.cnblogs.com/Images/OutliningIndicators/None.gif%u  星期(0……52), 这里星期一是星期的第一天   
http://www.cnblogs.com/Images/OutliningIndicators/None.gif%%  一个文字“%”。    
http://www.cnblogs.com/Images/OutliningIndicators/None.gif  
http://www.cnblogs.com/Images/OutliningIndicators/None.gif所有的其他字符不做解释被复制到结果中。   
http://www.cnblogs.com/Images/OutliningIndicators/None.gif  
http://www.cnblogs.com/Images/OutliningIndicators/None.gifmysql> select DATE\_FORMAT('1997-10-04 22:23:00', '%W %M %Y');  
http://www.cnblogs.com/Images/OutliningIndicators/None.gif        -> 'Saturday October 1997'  
http://www.cnblogs.com/Images/OutliningIndicators/None.gifmysql> select DATE\_FORMAT('1997-10-04 22:23:00', '%H:%i:%s');  
http://www.cnblogs.com/Images/OutliningIndicators/None.gif        -> '22:23:00'  
http://www.cnblogs.com/Images/OutliningIndicators/None.gifmysql> select DATE\_FORMAT('1997-10-04 22:23:00',  
http://www.cnblogs.com/Images/OutliningIndicators/None.gif                          '%D %y %a %d %m %b %j');  
http://www.cnblogs.com/Images/OutliningIndicators/None.gif        -> '4th 97 Sat 04 10 Oct 277'  
http://www.cnblogs.com/Images/OutliningIndicators/None.gifmysql> select DATE\_FORMAT('1997-10-04 22:23:00',  
http://www.cnblogs.com/Images/OutliningIndicators/None.gif                          '%H %k %I %r %T %S %w');  
http://www.cnblogs.com/Images/OutliningIndicators/None.gif        -> '22 22 10 10:23:00 PM 22:23:00 00 6'MySQL3.23中，在格式修饰符字符前需要%。在MySQL更早的版本中，%是可选的。   
http://www.cnblogs.com/Images/OutliningIndicators/None.gif  
http://www.cnblogs.com/Images/OutliningIndicators/None.gifTIME\_FORMAT(time,format)   
http://www.cnblogs.com/Images/OutliningIndicators/None.gif这象上面的DATE\_FORMAT()函数一样使用，但是format字符串只能包含处理小时、分钟和秒的那些格式修饰符。其他修饰符产生一个NULL值或0。   
http://www.cnblogs.com/Images/OutliningIndicators/None.gifCURDATE()   
http://www.cnblogs.com/Images/OutliningIndicators/None.gif　   
http://www.cnblogs.com/Images/OutliningIndicators/None.gifCURRENT\_DATE   
http://www.cnblogs.com/Images/OutliningIndicators/None.gif以'YYYY-MM-DD'或YYYYMMDD格式返回今天日期值，取决于函数是在一个字符串还是数字上下文被使用。   
http://www.cnblogs.com/Images/OutliningIndicators/None.gifmysql> select CURDATE();  
http://www.cnblogs.com/Images/OutliningIndicators/None.gif        -> '1997-12-15'  
http://www.cnblogs.com/Images/OutliningIndicators/None.gifmysql> select CURDATE() + 0;  
http://www.cnblogs.com/Images/OutliningIndicators/None.gif        -> 19971215  
http://www.cnblogs.com/Images/OutliningIndicators/None.gifCURTIME()   
http://www.cnblogs.com/Images/OutliningIndicators/None.gif　   
http://www.cnblogs.com/Images/OutliningIndicators/None.gifCURRENT\_TIME   
http://www.cnblogs.com/Images/OutliningIndicators/None.gif以'HH:MM:SS'或HHMMSS格式返回当前时间值，取决于函数是在一个字符串还是在数字的上下文被使用。   
http://www.cnblogs.com/Images/OutliningIndicators/None.gifmysql> select CURTIME();  
http://www.cnblogs.com/Images/OutliningIndicators/None.gif        -> '23:50:26'  
http://www.cnblogs.com/Images/OutliningIndicators/None.gifmysql> select CURTIME() + 0;  
http://www.cnblogs.com/Images/OutliningIndicators/None.gif        -> 235026  
http://www.cnblogs.com/Images/OutliningIndicators/None.gifNOW()   
http://www.cnblogs.com/Images/OutliningIndicators/None.gif　   
http://www.cnblogs.com/Images/OutliningIndicators/None.gifSYSDATE()   
http://www.cnblogs.com/Images/OutliningIndicators/None.gif　   
http://www.cnblogs.com/Images/OutliningIndicators/None.gifCURRENT\_TIMESTAMP   
http://www.cnblogs.com/Images/OutliningIndicators/None.gif以'YYYY-MM-DD HH:MM:SS'或YYYYMMDDHHMMSS格式返回当前的日期和时间，取决于函数是在一个字符串还是在数字的上下文被使用。   
http://www.cnblogs.com/Images/OutliningIndicators/None.gifmysql> select NOW();  
http://www.cnblogs.com/Images/OutliningIndicators/None.gif        -> '1997-12-15 23:50:26'  
http://www.cnblogs.com/Images/OutliningIndicators/None.gifmysql> select NOW() + 0;  
http://www.cnblogs.com/Images/OutliningIndicators/None.gif        -> 19971215235026  
http://www.cnblogs.com/Images/OutliningIndicators/None.gifUNIX\_TIMESTAMP()   
http://www.cnblogs.com/Images/OutliningIndicators/None.gif　   
http://www.cnblogs.com/Images/OutliningIndicators/None.gifUNIX\_TIMESTAMP(date)   
http://www.cnblogs.com/Images/OutliningIndicators/None.gif如果没有参数调用，返回一个Unix时间戳记(从'1970-01-01 00:00:00'GMT开始的秒数)。如果UNIX\_TIMESTAMP()用一个date参数被调用，它返回从'1970-01-01 00:00:00' GMT开始的秒数值。date可以是一个DATE字符串、一个DATETIME字符串、一个TIMESTAMP或以YYMMDD或YYYYMMDD格式的本地时间的一个数字。   
http://www.cnblogs.com/Images/OutliningIndicators/None.gifmysql> select UNIX\_TIMESTAMP();  
http://www.cnblogs.com/Images/OutliningIndicators/None.gif        -> 882226357  
http://www.cnblogs.com/Images/OutliningIndicators/None.gifmysql> select UNIX\_TIMESTAMP('1997-10-04 22:23:00');  
http://www.cnblogs.com/Images/OutliningIndicators/None.gif        -> 875996580  
http://www.cnblogs.com/Images/OutliningIndicators/None.gif当UNIX\_TIMESTAMP被用于一个TIMESTAMP列，函数将直接接受值，没有隐含的“string-to-unix-timestamp”变换。   
http://www.cnblogs.com/Images/OutliningIndicators/None.gif  
http://www.cnblogs.com/Images/OutliningIndicators/None.gifFROM\_UNIXTIME(unix\_timestamp)   
http://www.cnblogs.com/Images/OutliningIndicators/None.gif以'YYYY-MM-DD HH:MM:SS'或YYYYMMDDHHMMSS格式返回unix\_timestamp参数所表示的值，取决于函数是在一个字符串还是或数字上下文中被使用。   
http://www.cnblogs.com/Images/OutliningIndicators/None.gifmysql> select FROM\_UNIXTIME(875996580);  
http://www.cnblogs.com/Images/OutliningIndicators/None.gif        -> '1997-10-04 22:23:00'  
http://www.cnblogs.com/Images/OutliningIndicators/None.gifmysql> select FROM\_UNIXTIME(875996580) + 0;  
http://www.cnblogs.com/Images/OutliningIndicators/None.gif        -> 19971004222300  
http://www.cnblogs.com/Images/OutliningIndicators/None.gifFROM\_UNIXTIME(unix\_timestamp,format)   
http://www.cnblogs.com/Images/OutliningIndicators/None.gif返回表示 Unix 时间标记的一个字符串，根据format字符串格式化。format可以包含与DATE\_FORMAT()函数列出的条目同样的修饰符。   
http://www.cnblogs.com/Images/OutliningIndicators/None.gifmysql> select FROM\_UNIXTIME(UNIX\_TIMESTAMP(),  
http://www.cnblogs.com/Images/OutliningIndicators/None.gif                            '%Y %D %M %h:%i:%s %x');  
http://www.cnblogs.com/Images/OutliningIndicators/None.gif        -> '1997 23rd December 03:43:30 x'  
http://www.cnblogs.com/Images/OutliningIndicators/None.gifSEC\_TO\_TIME(seconds)   
http://www.cnblogs.com/Images/OutliningIndicators/None.gif返回seconds参数，变换成小时、分钟和秒，值以'HH:MM:SS'或HHMMSS格式化，取决于函数是在一个字符串还是在数字上下文中被使用。   
http://www.cnblogs.com/Images/OutliningIndicators/None.gifmysql> select SEC\_TO\_TIME(2378);  
http://www.cnblogs.com/Images/OutliningIndicators/None.gif        -> '00:39:38'  
http://www.cnblogs.com/Images/OutliningIndicators/None.gifmysql> select SEC\_TO\_TIME(2378) + 0;  
http://www.cnblogs.com/Images/OutliningIndicators/None.gif        -> 3938  
http://www.cnblogs.com/Images/OutliningIndicators/None.gifTIME\_TO\_SEC(time)   
http://www.cnblogs.com/Images/OutliningIndicators/None.gif返回time参数，转换成秒。   
http://www.cnblogs.com/Images/OutliningIndicators/None.gifmysql> select TIME\_TO\_SEC('22:23:00');  
http://www.cnblogs.com/Images/OutliningIndicators/None.gif        -> 80580  
http://www.cnblogs.com/Images/OutliningIndicators/None.gifmysql> select TIME\_TO\_SEC('00:39:38');  
http://www.cnblogs.com/Images/OutliningIndicators/None.gif        -> 2378  
http://www.cnblogs.com/Images/OutliningIndicators/None.gif   
http://www.cnblogs.com/Images/OutliningIndicators/None.gif   
http://www.cnblogs.com/Images/OutliningIndicators/None.gif其他函数  
http://www.cnblogs.com/Images/OutliningIndicators/None.gifDATABASE() 返回当前的数据库名字。 mysql> select DATABASE();  
http://www.cnblogs.com/Images/OutliningIndicators/None.gif        -> 'test'  
http://www.cnblogs.com/Images/OutliningIndicators/None.gif  
http://www.cnblogs.com/Images/OutliningIndicators/None.gif如果没有当前的数据库，DATABASE()返回空字符串。   
http://www.cnblogs.com/Images/OutliningIndicators/None.gifUSER() 　 SYSTEM\_USER() 　 SESSION\_USER() 返回当前MySQL用户名。  
http://www.cnblogs.com/Images/OutliningIndicators/None.gifmysql> select USER();  
http://www.cnblogs.com/Images/OutliningIndicators/None.gif        -> 'davida@localhost'  
http://www.cnblogs.com/Images/OutliningIndicators/None.gif  
http://www.cnblogs.com/Images/OutliningIndicators/None.gif在MySQL 3.22.11或以后版本中，这包括用户名和客户主机名。你可以象这样只提取用户名部分(值是否包括主机名部分均可工作)：   
http://www.cnblogs.com/Images/OutliningIndicators/None.gifmysql> select substring\_index(USER(),"@",1);  
http://www.cnblogs.com/Images/OutliningIndicators/None.gif        -> 'davida'  
http://www.cnblogs.com/Images/OutliningIndicators/None.gif  
http://www.cnblogs.com/Images/OutliningIndicators/None.gifPASSWORD(str) 从纯文本口令str计算一个口令字符串。该函数被用于为了在user授权表的Password列中存储口令而加密MySQL口令。  
http://www.cnblogs.com/Images/OutliningIndicators/None.gifmysql> select PASSWORD('badpwd');  
http://www.cnblogs.com/Images/OutliningIndicators/None.gif        -> '7f84554057dd964b'  
http://www.cnblogs.com/Images/OutliningIndicators/None.gif  
http://www.cnblogs.com/Images/OutliningIndicators/None.gifPASSWORD()加密是非可逆的。PASSWORD()不以与Unix口令加密的相同的方法执行口令加密。你不应该假定如果你的Unix 口令和你的MySQL口令是一样的，PASSWORD()将导致与在Unix口令文件存储的相同的加密值。见ENCRYPT()。   
http://www.cnblogs.com/Images/OutliningIndicators/None.gifENCRYPT(str[,salt]) 使用Unix crypt()系统调用加密str。salt参数应该是一个有2个字符的字符串。（MySQL 3.22.16中，salt可以长于2个字符。）  
http://www.cnblogs.com/Images/OutliningIndicators/None.gifmysql> select ENCRYPT("hello");  
http://www.cnblogs.com/Images/OutliningIndicators/None.gif        -> 'VxuFAJXVARROc'  
http://www.cnblogs.com/Images/OutliningIndicators/None.gif  
http://www.cnblogs.com/Images/OutliningIndicators/None.gif如果crypt()在你的系统上不可用，ENCRYPT()总是返回NULL。ENCRYPT()只保留str起始8个字符而忽略所有其他，至少在某些系统上是这样。这将由底层的crypt()系统调用的行为决定。   
http://www.cnblogs.com/Images/OutliningIndicators/None.gifENCODE(str,pass\_str) 使用pass\_str作为口令加密str。为了解密结果，使用DECODE()。结果是一个二进制字符串，如果你想要在列中保存它，使用一个BLOB列类型。 DECODE(crypt\_str,pass\_str) 使用pass\_str作为口令解密加密的字符串crypt\_str。crypt\_str应该是一个由ENCODE()返回的字符串。 MD5(string) 对字符串计算MD5校验和。值作为一个32长的十六进制数字被返回可以，例如用作哈希(hash)键。  
http://www.cnblogs.com/Images/OutliningIndicators/None.gifmysql> select MD5("testing")  
http://www.cnblogs.com/Images/OutliningIndicators/None.gif        -> 'ae2b1fca515949e5d54fb22b8ed95575'  
http://www.cnblogs.com/Images/OutliningIndicators/None.gif  
http://www.cnblogs.com/Images/OutliningIndicators/None.gif这是一个“RSA数据安全公司的MD5消息摘要算法”。   
http://www.cnblogs.com/Images/OutliningIndicators/None.gifLAST\_INSERT\_ID([expr]) 返回被插入一个AUTO\_INCREMENT列的最后一个自动产生的值。见20.4.29 mysql\_insert\_id()。  
http://www.cnblogs.com/Images/OutliningIndicators/None.gifmysql> select LAST\_INSERT\_ID();  
http://www.cnblogs.com/Images/OutliningIndicators/None.gif        -> 195  
http://www.cnblogs.com/Images/OutliningIndicators/None.gif  
http://www.cnblogs.com/Images/OutliningIndicators/None.gif产生的最后ID以每个连接为基础在服务器被维护，它不会被其他客户改变。如果你更新另外一个有非魔术值(即，不是NULL和不是0的一个值)的AUTO\_INCREMENT列，它甚至不会被改变。如果expr作为一个参数在一个UPDATE子句的LAST\_INSERT\_ID()里面给出，那么参数值作为一个LAST\_INSERT\_ID()值被返回。这可以用来模仿顺序：首先创建表：   
http://www.cnblogs.com/Images/OutliningIndicators/None.gifmysql> create table sequence (id int not null);  
http://www.cnblogs.com/Images/OutliningIndicators/None.gifmysql> insert into sequence values (0);  
http://www.cnblogs.com/Images/OutliningIndicators/None.gif  
http://www.cnblogs.com/Images/OutliningIndicators/None.gif然后表能被用来产生顺序号，象这样：   
http://www.cnblogs.com/Images/OutliningIndicators/None.gifmysql> update sequence set id=LAST\_INSERT\_ID(id+1);  
http://www.cnblogs.com/Images/OutliningIndicators/None.gif  
http://www.cnblogs.com/Images/OutliningIndicators/None.gif你可以不调用LAST\_INSERT\_ID()而产生顺序，但是这样使用函数的实用程序在服务器上自动维护ID值作为最后自动产生的值。你可以检索新的ID值，就像你能读入正常MySQL中的任何正常的AUTO\_INCREMENT值一样。例如，LAST\_INSERT\_ID()（没有一个参数 )将返回新ID。C API函数mysql\_insert\_id()也可被用来得到值。   
http://www.cnblogs.com/Images/OutliningIndicators/None.gifFORMAT(X,D) 格式化数字X为类似于格式'#,###,###.##'，四舍五入到D为小数。如果D为0，结果将没有小数点和小数部分。  
http://www.cnblogs.com/Images/OutliningIndicators/None.gifmysql> select FORMAT(12332.123456, 4);  
http://www.cnblogs.com/Images/OutliningIndicators/None.gif        -> '12,332.1235'  
http://www.cnblogs.com/Images/OutliningIndicators/None.gifmysql> select FORMAT(12332.1,4);  
http://www.cnblogs.com/Images/OutliningIndicators/None.gif        -> '12,332.1000'  
http://www.cnblogs.com/Images/OutliningIndicators/None.gifmysql> select FORMAT(12332.2,0);  
http://www.cnblogs.com/Images/OutliningIndicators/None.gif        -> '12,332'  
http://www.cnblogs.com/Images/OutliningIndicators/None.gif  
http://www.cnblogs.com/Images/OutliningIndicators/None.gifVERSION() 返回表明MySQL服务器版本的一个字符串。  
http://www.cnblogs.com/Images/OutliningIndicators/None.gifmysql> select VERSION();  
http://www.cnblogs.com/Images/OutliningIndicators/None.gif        -> '3.22.19b-log'  
http://www.cnblogs.com/Images/OutliningIndicators/None.gif  
http://www.cnblogs.com/Images/OutliningIndicators/None.gifGET\_LOCK(str,timeout) 试图获得由字符串str给定的一个名字的锁定，第二个timeout为超时。如果锁定成功获得，返回1，如果尝试超时了，返回0，或如果发生一个错误，返回NULL(例如从存储器溢出或线程用mysqladmin kill被杀死)。当你执行RELEASE\_LOCK()时、执行一个新的GET\_LOCK()或线程终止时，一个锁定被释放。该函数可以用来实现应用锁或模拟记录锁，它阻止其他客户用同样名字的锁定请求；赞成一个给定的锁定字符串名字的客户可以使用字符串执行子协作建议的锁定。  
http://www.cnblogs.com/Images/OutliningIndicators/None.gifmysql> select GET\_LOCK("lock1",10);  
http://www.cnblogs.com/Images/OutliningIndicators/None.gif        -> 1  
http://www.cnblogs.com/Images/OutliningIndicators/None.gifmysql> select GET\_LOCK("lock2",10);  
http://www.cnblogs.com/Images/OutliningIndicators/None.gif        -> 1  
http://www.cnblogs.com/Images/OutliningIndicators/None.gifmysql> select RELEASE\_LOCK("lock2");  
http://www.cnblogs.com/Images/OutliningIndicators/None.gif        -> 1  
http://www.cnblogs.com/Images/OutliningIndicators/None.gifmysql> select RELEASE\_LOCK("lock1");  
http://www.cnblogs.com/Images/OutliningIndicators/None.gif        -> NULL  
http://www.cnblogs.com/Images/OutliningIndicators/None.gif  
http://www.cnblogs.com/Images/OutliningIndicators/None.gif注意，第二个RELEASE\_LOCK()调用返回NULL，因为锁"lock1"自动地被第二个GET\_LOCK()调用释放。   
http://www.cnblogs.com/Images/OutliningIndicators/None.gifRELEASE\_LOCK(str) 释放字符串str命名的通过GET\_LOCK()获得的锁。如果锁被释放，返回1，如果锁没被这个线程锁定(在此情况下锁没被释放)返回0，并且如果命名的锁不存在，返回NULL。如果锁从来没有通过调用GET\_LOCK()获得或如果它已经被释放了，锁将不存在。 BENCHMARK(count,expr) BENCHMARK()函数重复countTimes次执行表达式expr，它可以用于计时MySQL处理表达式有多快。结果值总是0。意欲用于mysql客户，它报告查询的执行时间。  
http://www.cnblogs.com/Images/OutliningIndicators/None.gifmysql> select BENCHMARK(1000000,encode("hello","goodbye"));  
http://www.cnblogs.com/Images/OutliningIndicators/None.gif+----------------------------------------------+  
http://www.cnblogs.com/Images/OutliningIndicators/None.gif| BENCHMARK(1000000,encode("hello","goodbye")) |  
http://www.cnblogs.com/Images/OutliningIndicators/None.gif+----------------------------------------------+  
http://www.cnblogs.com/Images/OutliningIndicators/None.gif|                                            0 |  
http://www.cnblogs.com/Images/OutliningIndicators/None.gif+----------------------------------------------+  
http://www.cnblogs.com/Images/OutliningIndicators/None.gif1 row in set (4.74 sec)  
http://www.cnblogs.com/Images/OutliningIndicators/None.gif  
http://www.cnblogs.com/Images/OutliningIndicators/None.gif报告的时间是客户端的经过时间，不是在服务器端的CPU时间。执行BENCHMARK()若干次可能是明智的，并且注意服务器机器的负载有多重来解释结果。   
http://www.cnblogs.com/Images/OutliningIndicators/None.gif  
http://www.cnblogs.com/Images/OutliningIndicators/None.gif  
http://www.cnblogs.com/Images/OutliningIndicators/None.gif与GROUP BY子句一起使用的函数  
http://www.cnblogs.com/Images/OutliningIndicators/None.gif如果你在不包含GROUP BY子句的一个语句中使用聚合函数，它等价于聚合所有行。   
http://www.cnblogs.com/Images/OutliningIndicators/None.gifCOUNT(expr) 返回由一个SELECT语句检索出来的行的非NULL值的数目。  
http://www.cnblogs.com/Images/OutliningIndicators/None.gifmysql> select student.student\_name,COUNT(\*)  
http://www.cnblogs.com/Images/OutliningIndicators/None.gif           from student,course  
http://www.cnblogs.com/Images/OutliningIndicators/None.gif           where student.student\_id=course.student\_id  
http://www.cnblogs.com/Images/OutliningIndicators/None.gif           GROUP BY student\_name;  
http://www.cnblogs.com/Images/OutliningIndicators/None.gif  
http://www.cnblogs.com/Images/OutliningIndicators/None.gifCOUNT(\*)在它返回的检索出来的行数目上有些不同，不管他们是否包含NULL值。如果SELECT从一个表检索，或没有检索出其他列并且没有WHERE子句，COUNT(\*)被优化以便快速地返回。例如：   
http://www.cnblogs.com/Images/OutliningIndicators/None.gifmysql> select COUNT(\*) from student;  
http://www.cnblogs.com/Images/OutliningIndicators/None.gif    
http://www.cnblogs.com/Images/OutliningIndicators/None.gifCOUNT(DISTINCT expr,[exprhttp://www.cnblogs.com/Images/dot.gif]) 返回一个不同值的数目。  
http://www.cnblogs.com/Images/OutliningIndicators/None.gifmysql> select COUNT(DISTINCT results) from student;  
http://www.cnblogs.com/Images/OutliningIndicators/None.gif  
http://www.cnblogs.com/Images/OutliningIndicators/None.gif在MySQL中，你可以通过给出一个表达式列表以得到不同的表达式组合的数目。在 ANSI SQL中，你可能必须在CODE(DISTINCT ..)内进行所有表达式的连接。   
http://www.cnblogs.com/Images/OutliningIndicators/None.gifAVG(expr) 返回expr的平均值。  
http://www.cnblogs.com/Images/OutliningIndicators/None.gifmysql> select student\_name, AVG(test\_score)  
http://www.cnblogs.com/Images/OutliningIndicators/None.gif           from student  
http://www.cnblogs.com/Images/OutliningIndicators/None.gif           GROUP BY student\_name;  
http://www.cnblogs.com/Images/OutliningIndicators/None.gif    
http://www.cnblogs.com/Images/OutliningIndicators/None.gifMIN(expr) 　 MAX(expr) 返回expr的最小或最大值。MIN()和MAX()可以有一个字符串参数；在这种的情况下，他们返回最小或最大的字符串值。  
http://www.cnblogs.com/Images/OutliningIndicators/None.gifmysql> select student\_name, MIN(test\_score), MAX(test\_score)  
http://www.cnblogs.com/Images/OutliningIndicators/None.gif           from student  
http://www.cnblogs.com/Images/OutliningIndicators/None.gif           GROUP BY student\_name;  
http://www.cnblogs.com/Images/OutliningIndicators/None.gif   
http://www.cnblogs.com/Images/OutliningIndicators/None.gifSUM(expr) 返回expr的和。注意，如果返回的集合没有行，它返回NULL！ STD(expr) 　 STDDEV(expr) 返回expr标准差(deviation)。这是对 ANSI SQL 的扩展。该函数的形式STDDEV()是提供与Oracle的兼容性。 BIT\_OR(expr) 返回expr里所有位的位或。计算用 64 位(BIGINT)精度进行。 BIT\_AND(expr) 返回expr里所有位的位与。计算用 64 位(BIGINT)精度进行。   
http://www.cnblogs.com/Images/OutliningIndicators/None.gifMySQL扩展了GROUP BY的用法。你可以不出现在的GROUP BY部分的SELECT表达式中使用列或计算，这表示这个组的任何可能值。你可以使用它是性能更好，避免在不必要的项目上排序和分组。例如，你在下列查询中不需要在customer.name上聚合：   
http://www.cnblogs.com/Images/OutliningIndicators/None.gifmysql> select order.custid,customer.name,max(payments)  
http://www.cnblogs.com/Images/OutliningIndicators/None.gif       from order,customer  
http://www.cnblogs.com/Images/OutliningIndicators/None.gif       where order.custid = customer.custid  
http://www.cnblogs.com/Images/OutliningIndicators/None.gif       GROUP BY order.custid;  
http://www.cnblogs.com/Images/OutliningIndicators/None.gif  
http://www.cnblogs.com/Images/OutliningIndicators/None.gif在 ANSI SQL中，你将必须将customer.name加到GROUP BY子句。在MySQL中，名字是冗余的。   
http://www.cnblogs.com/Images/OutliningIndicators/None.gif如果你从GROUP BY部分省略的列在组中不是唯一的，不要使用这个功能。  
http://www.cnblogs.com/Images/OutliningIndicators/None.gif在某些情况下，你可以使用MIN()和MAX()获得一个特定的列值，即使它不是唯一的。下例给出从包含sort列中最小值的行的column值：   
http://www.cnblogs.com/Images/OutliningIndicators/None.gifsubstr(MIN(concat(sort,space(6-length(sort)),column),7,length(column)))  
http://www.cnblogs.com/Images/OutliningIndicators/None.gif  
http://www.cnblogs.com/Images/OutliningIndicators/None.gif注意，如果你正在使用MySQL 3.22(或更早)或如果你正在试图遵从ANSI SQL，你不能在GROUP BY或ORDER BY子句中使用表达式。你可以通过使用表达式的一个别名解决此限制：   
http://www.cnblogs.com/Images/OutliningIndicators/None.gifmysql> SELECT id,FLOOR(value/100) AS val FROM tbl\_name  
http://www.cnblogs.com/Images/OutliningIndicators/None.gif           GROUP BY id,val ORDER BY val;  
http://www.cnblogs.com/Images/OutliningIndicators/None.gif  
http://www.cnblogs.com/Images/OutliningIndicators/None.gif在MySQL3.23中，你可以这样做：   
http://www.cnblogs.com/Images/OutliningIndicators/None.gifmysql> SELECT id,FLOOR(value/100) FROM tbl\_name ORDER BY RAND();  
http://www.cnblogs.com/Images/OutliningIndicators/None.gif  
http://www.cnblogs.com/Images/OutliningIndicators/None.gif

[复制代码](javascript:void(0);)

学习的语句

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*字符串函数\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/  
select insert('ring',2,0,'ok'); /\*从2的地方插入ok字符，插入时替换0个字符\*/  
select repeat('ring',2); /\*重复输出字符串2次\*/  
select REVERSE('ring'); /\*反写\*/  
select mid('ring',2,2);/\*从某个位置取某个长度的字符\*/  
select instr('ring','i');/\*查找字符\*/  
select replace('very good','o','O'); /\*替换所有的字符\*/  
select LENGTH('very good'); /\*返回字符串长度\*/  
select ascii('abc');/\*返回最左边的字符ascii码\*/  
select space(3);/\*产生3次空格\*/  
select CONCAT('ri','n','g',' wang'); /\*组合成字符串\*/  
select lcase('RING'); /\*小写\*/  
select ucase('ring'); /\*转换成大写\*/  
select substring\_index(user(),'@',1); /\*提取当前的用户名,表示提取@左边第一出现的字符，如果后面的为负数则提取右边的字符\*/

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*数学函数\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/  
select conv('ee',16,10); /\*将'ee'从16转换成10，16和10可以是任意进制\*/  
select bin(10); /\*2进制\*/  
select hex(10); /\*16进制\*/  
select oct(10); /\*10进制\*/  
select round(12.5678,2); /\*将小数保留两位，四舍五入\*/  
select round(123.8767); /\*转换为整形，因为没有小数位\*/  
select char(65,66,99); /\*将ascill码组合成字符串\*/  
select least(3,6,5,1); /\*最小数字\*/  
select greatest(6,3,2,8,3,1);  /\*最大数字\*/  
select power(2,3); /\*2的3次幂\*/  
select sqrt(9); /\*开方\*/  
select pi(); /\*圆周率 \*/  
select RAND(11); /\*返回0-1的随机数，指定种子，每次的数都一样\*/  
select rand();  /\*没有指定种子，默认为时间为种子\*/

/\*\*\*\*\*\*\*\*日期时间函数\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/  
select now();  
select to\_days('2008-07-6'); /\*转换成天数\*/  
select to\_days('2008-8-8')-to\_days(now());/\*相差的天数\*/  
select day(now());/\*该日期中的时，分，秒，天，月，年 hour(),minute(),second(),month(),year()\*/  
select dayofyear('2008-2-2'); /\*返回该年已过的天数\*/  
select monthname(now()); /\*返回月的英文名\*/  
select dayname(now());/\*星期几的英文名\*/  
select week(now(),0); /\*表示当前的日期是第几周，后面的数字0表示从周日开始算，1表示从周一开始算\*/  
select from\_days(734215); /\*返回该天数表示的日期\*/  
select DATEDIFF(now(),'2008-8-8'); /\*前面的日期减去后面的日期所得的天数\*/  
select date\_add(now(),interval 2 day); /\*表示给当前的日期加2天\*/  
select date\_add(now(),interval 2 month); /\*表示给当前的日期加2个月\*/  
select CURRENT\_DATE(); /\*当前日期，短格式\*/  
select CURRENT\_TIME();/\*当前的时间\*/  
select current\_user();/\*当前的用户\*/  
select date\_format(now(),'%Y-%m-%d %H:%i:%s');/\*日期的格式，Y表示4位，H表示24小时制\*/  
select date\_format(now(),'%T'); /\*只显示时间，没有日期，T表示24小时制,r表示12小时制\*/  
select date\_format('2008-08-08 14:08:08','%Y-%m-%d'); /\*显示日期部分，时间不要\*/  
select date\_format(now(),'%Y-%m-%d %H:%i:%s %W'); /\*带日期的\*/  
select date\_format('2008-8-8','%W');/\*该日期的星期几\*/

/\*\*\*\*\*\*\*\*\*\*\*其他函数\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/  
select database();/\*返回当前的数据库名字\*/  
select substring\_index(CURRENT\_USER(),'@',1);/\*提取当前的用户名\*/  
select password('123'); /\*mysql的加密方式\*/  
select MD5('123');/\*32位的md5加密\*/  
select LAST\_INSERT\_ID();/\*返回最后一次自动产生的ID值\*/  
select format(123456789.565378,4);/\*输出123,456,789.5654有四舍五入\*/

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*控制流函数\*\*\*\*\*\*\*\*\*\*\*\*\*/  
select if(1,2,3);/\*如果第一参数为空则输入第三参数，否则输入第二个参数,后面的参数可以是字符,第一为整形\*/  
select ifnull(null,2); /\*如果第一参数为空则输出第二个参数，否则输出第一参数\*/

select   
       name,  
       case   
       when age<=20 then '年轻'  
       when age<=40 then '壮年'  
       when age<=50 then '中年'  
       else '老年'  
       end as 'agestr',  
       date\_format(adddate,'%Y-%m-%d') as 'adddate'  
 from users; /\*case 语句的用法\*/