**mysql中常用日期比较与计算函数**

假如有个表product有个字段add\_time,它的数据类型为datetime,有人可能会这样写sql：

|  |  |
| --- | --- |
| 代码如下 | 复制代码 |
| [select](http://www.111cn.net/tags.php/select/) \* from product where add\_time = '2013-01-12' | |

对于这种语句，如果你存储的格式是YY-mm-dd是这样的，那么OK，如果你存储的格式是：2013-01-12 23:23:56这种格式你就悲剧了，这是你就可以使用DATE()函数用来返回日期的部分，所以这条sql应该如下处理：

|  |  |
| --- | --- |
| 代码如下 | 复制代码 |
| select \* from product where Date(add\_time) = '2013-01-12' | |

再来一个，如果你要查询2013年1月份加入的产品呢？

|  |  |
| --- | --- |
| 代码如下 | 复制代码 |
| select \* from product where date(add\_time) between '2013-01-01' and '2013-01-31'  你还可以这样写：  select \* from product where Year(add\_time) = 2013 and Month(add\_time) = 1 | |

这些你该知道mysql日期函数在对你处理日期比较问题的作用了吧？

其date\_col的值是在最后30天以内：

|  |  |
| --- | --- |
| 代码如下 | 复制代码 |
| mysql> SELECT something FROM table  WHERE TO\_DAYS(NOW()) - TO\_DAYS(date\_col) <= 30; | |

DAYOFWEEK(date)   
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。

|  |  |
| --- | --- |
| 代码如下 | 复制代码 |
| mysql> select DAYOFWEEK('1998-02-03');  -> 3 | |

WEEKDAY(date)   
返回date的星期索引(0=星期一，1=星期二, ……6= 星期天)。

|  |  |
| --- | --- |
| 代码如下 | 复制代码 |
| mysql> select WEEKDAY('1997-10-04 22:23:00');  -> 5  mysql> select WEEKDAY('1997-11-05');  -> 2 | |

DAYOFMONTH(date)   
返回date的月份中日期，在1到31范围内。

|  |  |
| --- | --- |
| 代码如下 | 复制代码 |
| mysql> select DAYOFMONTH('1998-02-03');  -> 3 | |

DAYOFYEAR(date)   
返回date在一年中的日数, 在1到366范围内。

|  |  |
| --- | --- |
| 代码如下 | 复制代码 |
| mysql> select DAYOFYEAR('1998-02-03');  -> 34 | |

MONTH(date)   
返回date的月份，范围1到12。

|  |  |
| --- | --- |
| 代码如下 | 复制代码 |
| mysql> select MONTH('1998-02-03');  -> 2 | |

DAYNAME(date)   
返回date的星期名字。

|  |  |
| --- | --- |
| 代码如下 | 复制代码 |
| mysql> select DAYNAME("1998-02-05");  -> 'Thursday' | |

MONTHNAME(date)   
返回date的月份名字。

|  |  |
| --- | --- |
| 代码如下 | 复制代码 |
| mysql> select MONTHNAME("1998-02-05");  -> 'February' | |

QUARTER(date)   
返回date一年中的季度，范围1到4。

|  |  |
| --- | --- |
| 代码如下 | 复制代码 |
| mysql> select QUARTER('98-04-01');  -> 2 | |

原文地址：[点击打开链接http://williambryantliu.blog.hexun.com/85734616\_d.html#commentsListhttp://williambryantliu.blog.hexun.com/85734616\_d.html#commentsList](http://williambryantliu.blog.hexun.com/85734616_d.html#commentsList)

转载  2018-03-02   投稿：wdc   [[http://www.jb51.net/skin/2018/images/text-message.png](http://www.jb51.net/article/135811.htm#comments) 我要评论](http://www.jb51.net/article/135811.htm#comments)

MYSQL数据库中的日期比较与计算是经常用到的，例如比较两个日期大小，计算两个日期相差多少天

MySql中时间比较的实现

**unix\_timestamp()**

unix\_timestamp 函数可以接受一个参数，也可以不使用参数。

它的返回值是一个无符号的整数。不使用参数，它返回自1970年1月1日0时0分0秒到现在所经过的秒数，

如果 使用参数，参数的类型为时间类型或者时间类型的字符串表示，则是从1970-01-01 00:00:00到指定时间所经历的秒数。

有了这个函数，就可以很自然地把时间比较转换为一个无符号整数的比较。

例如，判断一个时间是否在一个区间内

[?](http://www.jb51.net/article/135811.htm)

|  |  |
| --- | --- |
| 1 | unix\_timestamp( time ) between unix\_timestamp( 'start ') and unix\_timestamp( 'end' ) |

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：

[?](http://www.jb51.net/article/135811.htm)

|  |  |
| --- | --- |
| 1 | mysql> SELECT something FROM table WHERE TO\_DAYS(NOW()) - TO\_DAYS(date\_col) <= 30; |

**DAYOFWEEK(date)**

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.jb51.net/article/135811.htm)

|  |  |
| --- | --- |
| 1  2 | mysql> select DAYOFWEEK('1998-02-03');  -> 3 |

**WEEKDAY(date)**

返回date的星期索引(0=星期一，1=星期二, ……6= 星期天)。

[?](http://www.jb51.net/article/135811.htm)

|  |  |
| --- | --- |
| 1  2  3  4 | mysql> select WEEKDAY('1997-10-04 22:23:00');  -> 5  mysql> select WEEKDAY('1997-11-05');  -> 2 |

**DAYOFMONTH(date)**

返回date的月份中日期，在1到31范围内。

[?](http://www.jb51.net/article/135811.htm)

|  |  |
| --- | --- |
| 1  2 | mysql> select DAYOFMONTH('1998-02-03');  -> 3 |

**DAYOFYEAR(date)**

返回date在一年中的日数, 在1到366范围内。

[?](http://www.jb51.net/article/135811.htm)

|  |  |
| --- | --- |
| 1  2 | mysql> select DAYOFYEAR('1998-02-03');  -> 34 |

**MONTH(date)**

返回date的月份，范围1到12。

[?](http://www.jb51.net/article/135811.htm)

|  |  |
| --- | --- |
| 1  2 | mysql> select MONTH('1998-02-03');  -> 2 |

**DAYNAME(date)**

返回date的星期名字。

[?](http://www.jb51.net/article/135811.htm)

|  |  |
| --- | --- |
| 1  2 | mysql> select DAYNAME("1998-02-05");  -> 'Thursday' |

**MONTHNAME(date)**

返回date的月份名字。

[?](http://www.jb51.net/article/135811.htm)

|  |  |
| --- | --- |
| 1  2 | mysql> select MONTHNAME("1998-02-05");  -> 'February' |

**QUARTER(date)**

返回date一年中的季度，范围1到4。

[?](http://www.jb51.net/article/135811.htm)

|  |  |
| --- | --- |
| 1  2 | mysql> select QUARTER('98-04-01');  -> 2 |

**WEEK(date)**

**WEEK(date,first)**

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.jb51.net/article/135811.htm)

|  |  |
| --- | --- |
| 1  2  3  4  5  6 | 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。

[?](http://www.jb51.net/article/135811.htm)

|  |  |
| --- | --- |
| 1  2 | mysql> select YEAR('98-02-03');  -> 1998 |

**HOUR(time)**

返回time的小时，范围是0到23。

[?](http://www.jb51.net/article/135811.htm)

|  |  |
| --- | --- |
| 1  2 | mysql> select HOUR('10:05:03');  -> 10 |

**MINUTE(time)**

返回time的分钟，范围是0到59。

[?](http://www.jb51.net/article/135811.htm)

|  |  |
| --- | --- |
| 1  2 | mysql> select MINUTE('98-02-03 10:05:03');  -> 5 |

**SECOND(time)**

回来time的秒数，范围是0到59。

[?](http://www.jb51.net/article/135811.htm)

|  |  |
| --- | --- |
| 1  2 | mysql> select SECOND('10:05:03');  -> 3 |

**PERIOD\_ADD(P,N)**

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.jb51.net/article/135811.htm)

|  |  |
| --- | --- |
| 1  2 | 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应该以格式YYMM或YYYYMM。注意，时期参数P1和P2不是日期值。

[?](http://www.jb51.net/article/135811.htm)

|  |  |
| --- | --- |
| 1  2 | 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)**

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

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"

MySQL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.jb51.net/article/135811.htm)

|  |  |
| --- | --- |
| 1  2  3  4  5  6  7  8  9  10  11  12  13  14  15  16  17  18  19  20  21  22  23  24  25  26  27  28  29 | 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 |

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.jb51.net/article/135811.htm)

|  |  |
| --- | --- |
| 1  2 | mysql> select DATE\_ADD('1998-01-30', Interval 1 month);  -> 1998-02-28 |

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)**

给出一个日期date，返回一个天数(从0年的天数)。

[?](http://www.jb51.net/article/135811.htm)

|  |  |
| --- | --- |
| 1  2  3  4 | mysql> select TO\_DAYS(950501);  -> 728779  mysql> select TO\_DAYS('1997-10-07');  -> 729669 |

**FROM\_DAYS(N)**

给出一个天数N，返回一个DATE值。

[?](http://www.jb51.net/article/135811.htm)

|  |  |
| --- | --- |
| 1  2 | mysql> select FROM\_DAYS(729669);  -> '1997-10-07' |

**DATE\_FORMAT(date,format)**

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)

%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=Sunday ……6=Saturday ）  
%U 星期(0……52), 这里星期天是星期的第一天  
%u 星期(0……52), 这里星期一是星期的第一天  
%% 一个文字“%"

所有的其他字符不做解释被复制到结果中。

[?](http://www.jb51.net/article/135811.htm)

|  |  |
| --- | --- |
| 1  2  3  4  5  6  7  8  9  10 | 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' |

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。

**TIME\_FORMAT(time,format)**

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。

**CURDATE()**

CURRENT\_DATE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.jb51.net/article/135811.htm)

|  |  |
| --- | --- |
| 1  2  3  4 | 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，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.jb51.net/article/135811.htm)

|  |  |
| --- | --- |
| 1  2  3  4 | 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.jb51.net/article/135811.htm)

|  |  |
| --- | --- |
| 1  2  3  4 | mysql> select NOW();  -> '1997-12-15 23:50:26'  mysql> select NOW() + 0;  -> 19971215235026 |

**UNIX\_TIMESTAMP()  
UNIX\_TIMESTAMP(date)**  
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.jb51.net/article/135811.htm)

|  |  |
| --- | --- |
| 1  2  3  4 | mysql> select UNIX\_TIMESTAMP();  -> 882226357  mysql> select UNIX\_TIMESTAMP('1997-10-04 22:23:00');  -> 875996580 |

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。

**FROM\_UNIXTIME(unix\_timestamp)**

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.jb51.net/article/135811.htm)

|  |  |
| --- | --- |
| 1  2  3  4 | mysql> select FROM\_UNIXTIME(875996580);  -> '1997-10-04 22:23:00'  mysql> select FROM\_UNIXTIME(875996580) + 0;  -> 19971004222300 |

**FROM\_UNIXTIME(unix\_timestamp,format)**

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.jb51.net/article/135811.htm)

|  |  |
| --- | --- |
| 1  2  3 | mysql> select FROM\_UNIXTIME(UNIX\_TIMESTAMP(),  '%Y %D %M %h:%i:%s %x');  -> '1997 23rd December 03:43:30 x' |

**SEC\_TO\_TIME(seconds)**

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.jb51.net/article/135811.htm)

|  |  |
| --- | --- |
| 1  2  3  4 | mysql> select SEC\_TO\_TIME(2378);  -> '00:39:38'  mysql> select SEC\_TO\_TIME(2378) + 0;  -> 3938 |

**TIME\_TO\_SEC(time)**

返回time参数，转换成秒。

[?](http://www.jb51.net/article/135811.htm)

|  |  |
| --- | --- |
| 1  2  3  4 | mysql> select TIME\_TO\_SEC('22:23:00');  -> 80580  mysql> select TIME\_TO\_SEC('00:39:38');  -> 2378 |

Mysql的日期函数很多，用作日期比较的函数有时并不一定是一个函数有可能是多个函数组合在一起使用，所以大家要发挥自己的想像力

**您可能感兴趣的文章:**

1. [MySQL中日期比较时遇到的编码问题解决办法](http://www.jb51.net/article/52510.htm)
2. [PHP以及MYSQL日期比较方法](http://www.jb51.net/article/32279.htm)
3. [mysql 获取当前日期函数及时间格式化参数详解](http://www.jb51.net/article/54095.htm)
4. [MySql用DATE\_FORMAT截取DateTime字段的日期值](http://www.jb51.net/article/53293.htm)
5. [mysql unix准换时间格式查找指定日期数据代码](http://www.jb51.net/article/48334.htm)
6. [MySql日期查询语句详解](http://www.jb51.net/article/42774.htm)
7. [深入mysql YEAR() MONTH() DAYOFMONTH()日期函数的详解](http://www.jb51.net/article/38996.htm)
8. [mysql 查询指定日期时间内sql语句实现原理与代码](http://www.jb51.net/article/32642.htm)
9. [PHP+Mysql日期时间如何转换(UNIX时间戳和格式化日期)](http://www.jb51.net/article/30787.htm)
10. [MySQL 日期时间函数常用总结](http://www.jb51.net/article/30559.htm)
11. [php&mysql 日期操作小记](http://www.jb51.net/article/29768.htm)
12. [MySQL 获得当前日期时间的函数小结](http://www.jb51.net/article/29125.htm)
13. [mysql中取系统当前时间，当前日期方便查询判定的代码](http://www.jb51.net/article/29124.htm)
14. [Mysql 日期时间 DATE\_FORMAT(date,format)](http://www.jb51.net/article/25655.htm)
15. [Mysql中日期和时间函数应用不用求人](http://www.jb51.net/article/25359.htm)
16. [mysql的日期和时间函数](http://www.jb51.net/article/25358.htm)
17. [MySQL日期数据类型、时间类型使用总结](http://www.jb51.net/article/23966.htm)
18. [MySQL的Data\_ADD函数与日期格式化函数说明](http://www.jb51.net/article/23827.htm)