

# mysql 常见的时间查询

作者: [c3gen](#)

原文链接: <https://ld246.com/article/1510042991525>

来源网站: [链滴](#)

许可协议: [署名-相同方式共享 4.0 国际 \(CC BY-SA 4.0\)](#)

最近使用的一些 时间查询统计

测试表结构如下:

Create Table |

```
CREATE TABLE test_t (  
id int(10) NOT NULL AUTO_INCREMENT,  
create_time datetime DEFAULT NULL,  
PRIMARY KEY (id)  
) ENGINE=InnoDB AUTO_INCREMENT=10 DEFAULT CHARSET=utf8
```

两个字段 方便操作

测试数据如下:

id	create_time
1	2017-11-03 14:22:50
2	2017-11-04 14:22:54
3	2017-11-03 14:22:57
4	2017-11-01 14:23:00
5	2017-11-05 14:23:03
6	2017-11-01 14:23:11
7	2017-11-07 14:23:15
8	2017-11-02 14:23:27
9	2017-10-31 14:23:40
10	2017-11-06 14:38:37
11	2017-07-21 15:38:25
12	2017-08-18 15:38:35
13	2017-11-07 15:53:48
14	2016-11-07 16:20:25

```
insert into test_t (id, create_time) values('1','2017-11-03 14:22:50');  
insert into test_t (id, create_time) values('2','2017-11-04 14:22:54');  
insert into test_t (id, create_time) values('3','2017-11-03 14:22:57');  
insert into test_t (id, create_time) values('4','2017-11-01 14:23:00');  
insert into test_t (id, create_time) values('5','2017-11-05 14:23:03');  
insert into test_t (id, create_time) values('6','2017-11-01 14:23:11');  
insert into test_t (id, create_time) values('7','2017-11-07 14:23:15');  
insert into test_t (id, create_time) values('8','2017-11-02 14:23:27');  
insert into test_t (id, create_time) values('9','2017-10-31 14:23:40');  
insert into test_t (id, create_time) values('10','2017-11-06 14:38:37');  
insert into test_t (id, create_time) values('11','2017-07-21 15:38:25');  
insert into test_t (id, create_time) values('12','2017-08-18 15:38:35');  
insert into test_t (id, create_time) values('13','2017-11-07 15:53:48');  
insert into test_t (id, create_time) values('14','2016-11-07 16:20:25');
```

为了方便查看， 都叫上了order by 排序

1.要统计每日的新增数据， 就要把时间戳格式的时间FORMAT成年-月-日的格式， 并按照format过  
字段做分组 (group by)， 计算count

- SELECT DATE\_FORMAT(create\_time,'%Y-%m-%d') AS TIME , COUNT(\*) AS COUNT FROM test\_t GROUP BY TIME

time	count
2017-07-21	1
2017-08-18	1
2017-10-31	1
2017-11-01	2
2017-11-02	1
2017-11-03	2
2017-11-04	1
2017-11-05	1
2017-11-06	1
2017-11-07	1

2.要统计最近7天每天的新增数据，按时间倒序排序

- SELECT DATE\_FORMAT(create\_time,'%Y-%m-%d') AS TIME, COUNT(\*) AS COUNT FROM test\_t WHERE create\_time >= DATE(NOW()) - INTERVAL 6 DAY GROUP BY DAY(create\_time) ORDER BY create\_time DESC;

TIME	COUNT
2017-11-07	1
2017-11-06	1
2017-11-05	1
2017-11-04	1
2017-11-03	2
2017-11-02	1
2017-11-01	2

3.要统计最近7天的新增数据，总计数据

- SELECT \* FROM test\_t WHERE DATE\_SUB(CURDATE(), INTERVAL 6 DAY) <= DATE(create\_time) ORDER BY create\_time DESC

id	create_time
7	2017-11-07 14:23:15
10	2017-11-06 14:38:37
5	2017-11-05 14:23:03
2	2017-11-04 14:22:54
3	2017-11-03 14:22:57
1	2017-11-03 14:22:50
8	2017-11-02 14:23:27
6	2017-11-01 14:23:11
4	2017-11-01 14:23:00

4.统计今日新增数据

- SELECT \* FROM test\_t WHERE TO\_DAYS(create\_time) = TO\_DAYS(NOW());

id	create_time
7	2017-11-07 14:23:15

5.统计昨日新增数据

方法1:

- SELECT \* FROM test\_t WHERE TO\_DAYS(NOW()) = 1 + TO\_DAYS(create\_time);

id	create_time
10	2017-11-06 14:38:37

方法2: 取出昨天的和今天的记录过滤掉今天的记录

6.统计最近一周的数据

note: 最近一周不等于最近7天, 一周是从周日开始算的, 比如测试的时间是2017-11-7, 那么往前的周就是11-5开始

- SELECT \* FROM test\_t WHERE YEARWEEK(DATE\_FORMAT(create\_time,'%Y-%m-%d')) = YEARWEEK(NOW()) ORDER BY create\_time DESC

13	2017-11-07 15:53:48
7	2017-11-07 14:23:15
10	2017-11-06 14:38:37
5	2017-11-05 14:23:03

## 7.统计最近一个月的数据

note: 最近一个月不等于最近30天

- SELECT \* FROM test t WHERE create\_time BETWEEN DATE\_SUB(NOW(),INTERVAL 1 MONTH AND NOW() ORDER BY create\_time DESC

13	2017-11-07 15:53:48
7	2017-11-07 14:23:15
10	2017-11-06 14:38:37
5	2017-11-05 14:23:03
2	2017-11-04 14:22:54
3	2017-11-03 14:22:57
1	2017-11-03 14:22:50
8	2017-11-02 14:23:27
6	2017-11-01 14:23:11
4	2017-11-01 14:23:00
9	2017-10-31 14:23:40

## 8.统计上个月的数据

- SELECT \* FROM test\_t WHERE PERIOD\_DIFF(DATE\_FORMAT(NOW() , '%Y%m') , DATE\_FORMAT(create\_time, '%Y%m')) = 1 ORDER BY create\_time DESC

id	create_time
9	2017-10-31 14:23:40

## 9.统计本季度的数据

- SELECT \* FROM test\_t WHERE QUARTER(create\_time)=QUARTER(NOW()) ORDER BY create\_time DESC

13	2017-11-07 15:53:48
7	2017-11-07 14:23:15
10	2017-11-06 14:38:37
5	2017-11-05 14:23:03
2	2017-11-04 14:22:54
3	2017-11-03 14:22:57
1	2017-11-03 14:22:50
8	2017-11-02 14:23:27
6	2017-11-01 14:23:11
4	2017-11-01 14:23:00
9	2017-10-31 14:23:40

## 10.统计上个季度的数据

- SELECT \* FROM test t WHERE QUARTER(create\_time)=QUARTER(DATE\_SUB(NOW(),INTERVAL 1 QUARTER)) ORDER BY create\_time DESC

id	create_time
12	2017-08-18 15:38:35
11	2017-07-21 15:38:25

## 11.统计今年的数据

- `SELECT * FROM test_t WHERE YEAR(create_time)=YEAR(NOW()) ORDER BY create_time DESC`

id	create_time
13	2017-11-07 15:53:48
7	2017-11-07 14:23:15
10	2017-11-06 14:38:37
5	2017-11-05 14:23:03
2	2017-11-04 14:22:54
3	2017-11-03 14:22:57
1	2017-11-03 14:22:50
8	2017-11-02 14:23:27
6	2017-11-01 14:23:11
4	2017-11-01 14:23:00
9	2017-10-31 14:23:40
12	2017-08-18 15:38:35
11	2017-07-21 15:38:25

## 12.统计去年的数据

- `SELECT * FROM test_t WHERE YEAR(create_time)=YEAR(DATE_SUB(NOW(),INTERVAL 1 YEAR)) ORDER BY create_time DESC`

id	create_time
14	2016-11-07 16:20:25