

# 狂神说MySQL05：MySQL函数

秦疆 狂神说 2020-05-02

狂神说MySQL系列连载课程，通俗易懂，基于MySQL5.7.19版本，欢迎各位狂粉转发关注学习。禁止随意转载，转载记住贴出B站视频链接及公众号链接！



上课视频同步文档



## MySQL函数

### 常用函数

#### 数据函数

```
SELECT ABS(-8); /*绝对值*/
SELECT CEILING(9.4); /*向上取整*/
SELECT FLOOR(9.4); /*向下取整*/
SELECT RAND(); /*随机数,返回一个0-1之间的随机数*/
SELECT SIGN(0); /*符号函数: 负数返回-1, 正数返回1, 0返回0*/
```

#### 字符串函数

```
SELECT CHAR_LENGTH('狂神说坚持就能成功'); /*返回字符串包含的字符数*/
SELECT CONCAT('我','爱','程序'); /*合并字符串,参数可以有多个*/
SELECT INSERT('我爱编程helloworld',1,2,'超级热爱'); /*替换字符串,从某个位置开始替换某个长度*/
SELECT LOWER('KuangShen'); /*小写*/
SELECT UPPER('KuangShen'); /*大写*/
SELECT LEFT('hello,world',5); /*从左边截取*/
SELECT RIGHT('hello,world',5); /*从右边截取*/
SELECT REPLACE('狂神说坚持就能成功','坚持','努力'); /*替换字符串*/
SELECT SUBSTR('狂神说坚持就能成功',4,6); /*截取字符串,开始和长度*/
SELECT REVERSE('狂神说坚持就能成功'); /*反转*/

-- 查询姓周的同学,改成邹
SELECT REPLACE(studentname,'周','邹') AS 新名字
FROM student WHERE studentname LIKE '周%';
```

#### 日期和时间函数

```
SELECT CURRENT_DATE(); /*获取当前日期*/
SELECT CURDATE(); /*获取当前日期*/
```

```
SELECT NOW(); /*获取当前日期和时间*/
SELECT LOCALTIME(); /*获取当前日期和时间*/
SELECT SYSDATE(); /*获取当前日期和时间*/

-- 获取年月日,时分秒
SELECT YEAR(NOW());
SELECT MONTH(NOW());
SELECT DAY(NOW());
SELECT HOUR(NOW());
SELECT MINUTE(NOW());
SELECT SECOND(NOW());
```

系统信息函数

```
SELECT VERSION(); /*版本*/
SELECT USER(); /*用户*/
```

聚合函数

函数名称	描述
COUNT()	返回满足Select条件的记录总和数，如 select count(*) 【不建议使用 *，效率低】
SUM()	返回数字字段或表达式列作统计，返回一列的总和。
AVG()	通常为数值字段或表达列作统计，返回一列的平均值
MAX()	可以为数值字段，字符字段或表达式列作统计，返回最大的值。
MIN()	可以为数值字段，字符字段或表达式列作统计，返回最小的值。

<>

```
-- 聚合函数
/*COUNT:非空的*/
SELECT COUNT(studentname) FROM student;
SELECT COUNT(*) FROM student;
SELECT COUNT(1) FROM student; /*推荐*/

-- 从含义上讲，count(1) 与 count(*) 都表示对全部数据行的查询。
-- count(字段) 会统计该字段在表中出现的次数，忽略字段为null 的情况。即不统计字段为null 的记录。
-- count(*) 包括了所有的列，相当于行数，在统计结果的时候，包含字段为null 的记录；
-- count(1) 用1代表代码行，在统计结果的时候，包含字段为null 的记录 。
/*
很多人认为count(1)执行的效率会比count(*)高，原因是count(*)会存在全表扫描，而count(1)可以针对一个字段进行查询。其实不然，count(1)和count(*)都会对全表进行扫描，统计所有记录的条数，包括那些为null的记录，因此，它们的效率可以说是相差无几。而count(字段)则与前者不同，它会统计该字段不为null的记录条数。

下面它们之间的一些对比：
```

```

1) 在表没有主键时, count(1)比count(*)快
2) 有主键时, 主键作为计算条件, count(主键)效率最高;
3) 若表格只有一个字段, 则count(*)效率较高。
*/

SELECT SUM(StudentResult) AS 总和 FROM result;
SELECT AVG(StudentResult) AS 平均分 FROM result;
SELECT MAX(StudentResult) AS 最高分 FROM result;
SELECT MIN(StudentResult) AS 最低分 FROM result;

```

题目:

```

-- 查询不同课程的平均分,最高分,最低分
-- 前提:根据不同的课程进行分组

SELECT subjectname,AVG(studentresult) AS 平均分,MAX(StudentResult) AS 最高
分,MIN(StudentResult) AS 最低分
FROM result AS r
INNER JOIN `subject` AS s
ON r.subjectno = s.subjectno
GROUP BY r.subjectno
HAVING 平均分>80;

/*
where写在group by前面.
要是放在分组后面的筛选
要使用HAVING..
因为having是从前面筛选的字段再筛选, 而where是从数据表中的>字段直接进行的筛选的
*/

```

## MD5 加密

### 一、MD5简介

MD5即Message-Digest Algorithm 5（信息-摘要算法5），用于确保信息传输完整一致。是计算机广泛使用的杂凑算法之一（又译摘要算法、哈希算法），主流编程语言普遍已有MD5实现。将数据（如汉字）运算为另一固定长度值，是杂凑算法的基础原理，MD5的前身有MD2、MD3和MD4。

### 二、实现数据加密

新建一个表 testmd5

```

CREATE TABLE `testmd5` (
  `id` INT(4) NOT NULL,
  `name` VARCHAR(20) NOT NULL,
  `pwd` VARCHAR(50) NOT NULL,
  PRIMARY KEY (`id`)
) ENGINE=INNODB DEFAULT CHARSET=utf8

```

插入一些数据

```
INSERT INTO testmd5 VALUES (1,'kuangshen','123456'), (2,'qinjiang','456789')
```

如果我们要对pwd这一列数据进行加密，语法是：

```
update testmd5 set pwd = md5(pwd);
```

如果单独对某个用户(如kuangshen)的密码加密：

```
INSERT INTO testmd5 VALUES (3,'kuangshen2','123456')
update testmd5 set pwd = md5(pwd) where name = 'kuangshen2';
```

插入新的数据自动加密

```
INSERT INTO testmd5 VALUES (4,'kuangshen3',md5('123456'));
```

查询登录用户信息（md5对比使用，查看用户输入加密后的密码进行比对）

```
SELECT * FROM testmd5 WHERE `name`='kuangshen' AND pwd=MD5('123456');
```

## 小结

-- ===== 内置函数 =====

-- 数值函数

```
abs(x)          -- 绝对值 abs(-10.9) = 10
format(x, d)    -- 格式化千分位数值 format(1234567.456, 2) = 1,234,567.46
ceil(x)         -- 向上取整 ceil(10.1) = 11
floor(x)        -- 向下取整 floor(10.1) = 10
round(x)        -- 四舍五入去整
mod(m, n)       -- m%n m mod n 求余 10%3=1
pi()            -- 获得圆周率
pow(m, n)       -- m^n
sqrt(x)         -- 算术平方根
rand()          -- 随机数
truncate(x, d)  -- 截取d位小数
```

-- 时间日期函数

```
now(), current_timestamp(); -- 当前日期时间
current_date();            -- 当前日期
current_time();            -- 当前时间
date('yyyy-mm-dd hh:ii:ss'); -- 获取日期部分
time('yyyy-mm-dd hh:ii:ss'); -- 获取时间部分
date_format('yyyy-mm-dd hh:ii:ss', '%d %y %a %d %m %b %j'); -- 格式化时间
unix_timestamp();          -- 获得unix时间戳
from_unixtime();           -- 从时间戳获得时间
```

-- 字符串函数

```
length(string)          -- string长度，字节
char_length(string)     -- string的字符个数
substring(str, position [,length]) -- 从str的position开始，取length个字符
replace(str, search_str, replace_str) -- 在str中用replace_str替换search_str
```

```
instr(string ,substring)      -- 返回substring首次在string中出现的位置
concat(string [,...])        -- 连接字符串
charset(str)                 -- 返回字符串字符集
lcase(string)                -- 转换成小写
left(string, length)         -- 从string2中的左边起取length个字符
load_file(file_name)         -- 从文件读取内容
locate(substring, string [,start_position]) -- 同instr,但可指定开始位置
lpad(string, length, pad)     -- 重复用pad加在string开头,直到字符串长度为length
ltrim(string)                -- 去除前端空格
repeat(string, count)        -- 重复count次
rpad(string, length, pad)     -- 在str后用pad补充,直到长度为length
rtrim(string)                -- 去除后端空格
strcmp(string1 ,string2)     -- 逐字符比较两字符串大小

-- 聚合函数
count()
sum();
max();
min();
avg();
group_concat()

-- 其他常用函数
md5();
default();
```



视频同步更新

如果觉得帮助到了您，不妨赞赏支持一下吧！



“赠人玫瑰，手有余香”

狂神说 的赞赏码

 狂神说



 狂神说