# 数据库常用函数

## 数学函数

abs(x) 返回x的绝对值

bin(x) 返回x的二进制（oct返回八进制，hex返回十六进制）

ceiling(x) 返回大于x的最小整数值

exp(x) 返回值e（自然对数的底）的x次方

floor(x) 返回小于x的最大整数值

greatest(x1,x2,...,xn)返回集合中最大的值

least(x1,x2,...,xn) 返回集合中最小的值

ln(x) 返回x的自然对数

log(x,y)返回x的以y为底的对数

mod(x,y) 返回x/y的模（余数）

pi()返回pi的值（圆周率）

rand()返回０到１内的随机值,可以通过提供一个参数(种子)使rand()随机数生成器生成一个指定的值。

round(x,y)返回参数x的四舍五入的有y位小数的值

sign(x) 返回代表数字x的符号的值

sqrt(x) 返回一个数的平方根

truncate(x,y) 返回数字x截短为y位小数的结果

## 聚合函数(常用于group by从句的select查询中)

avg(col)返回指定列的平均值

count(col)返回指定列中非null值的个数

min(col)返回指定列的最小值

max(col)返回指定列的最大值

sum(col)返回指定列的所有值之和

group\_concat(col) 返回由属于一组的列值连接组合而成的结果

## 字符串函数

ascii(char)返回字符的ascii码值

bit\_length(str)返回字符串的比特长度

concat(s1,s2...,sn)将s1,s2...,sn连接成字符串

concat\_ws(sep,s1,s2...,sn)将s1,s2...,sn连接成字符串，并用sep字符间隔

insert(str,x,y,instr) 将字符串str从第x位置开始，y个字符长的子串替换为字符串instr，返回结果

find\_in\_set(str,list)分析逗号分隔的list列表，如果发现str，返回str在list中的位置

lcase(str)或lower(str) 返回将字符串str中所有字符改变为小写后的结果

left(str,x)返回字符串str中最左边的x个字符

length(s)返回字符串str中的字符数

ltrim(str) 从字符串str中切掉开头的空格

position(substr in str) 返回子串substr在字符串str中第一次出现的位置

quote(str) 用反斜杠转义str中的单引号

repeat(str,srchstr,rplcstr)返回字符串str重复x次的结果

reverse(str) 返回颠倒字符串str的结果

right(str,x) 返回字符串str中最右边的x个字符

rtrim(str) 返回字符串str尾部的空格

strcmp(s1,s2)比较字符串s1和s2

trim(str)去除字符串首部和尾部的所有空格

ucase(str)或upper(str) 返回将字符串str中所有字符转变为大写后的结果

## 日期和时间函数

curdate()或current\_date() 返回当前的日期

curtime()或current\_time() 返回当前的时间

date\_add(date,interval int keyword)返回日期date加上间隔时间int的结果(int必须按照关键字进行格式化),如：selectdate\_add(current\_date,interval 6 month);

date\_format(date,fmt) 依照指定的fmt格式格式化日期date值

date\_sub(date,interval int keyword)返回日期date加上间隔时间int的结果(int必须按照关键字进行格式化),如：selectdate\_sub(current\_date,interval 6 month);

dayofweek(date) 返回date所代表的一星期中的第几天(1~7)

dayofmonth(date) 返回date是一个月的第几天(1~31)

dayofyear(date) 返回date是一年的第几天(1~366)

dayname(date) 返回date的星期名，如：select dayname(current\_date);

from\_unixtime(ts,fmt) 根据指定的fmt格式，格式化unix时间戳ts

hour(time) 返回time的小时值(0~23)

minute(time) 返回time的分钟值(0~59)

month(date) 返回date的月份值(1~12)

monthname(date) 返回date的月份名，如：select monthname(current\_date);

now() 返回当前的日期和时间

quarter(date) 返回date在一年中的季度(1~4)，如select quarter(current\_date);

week(date) 返回日期date为一年中第几周(0~53)

year(date) 返回日期date的年份(1000~9999)

一些示例：

获取当前系统时间：select from\_unixtime(unix\_timestamp());

select extract(year\_month from current\_date);

select extract(day\_second from current\_date);

select extract(hour\_minute from current\_date);

返回两个日期值之间的差值(月数)：select period\_diff(200302,199802);

在mysql中计算年龄：

select date\_format(from\_days(to\_days(now())-to\_days(birthday)),'%y')+0 as age from employee;

这样，如果brithday是未来的年月日的话，计算结果为0。

下面的sql语句计算员工的绝对年龄，即当birthday是未来的日期时，将得到负值。

select date\_format(now(), '%y') - date\_format(birthday, '%y') -(date\_format(now(), '00-%m-%d') <date\_format(birthday, '00-%m-%d')) as age from employee

## 加密函数

aes\_encrypt(str,key) 返回用密钥key对字符串str利用高级加密标准算法加密后的结果，调用aes\_encrypt的结果是一个二进制字符串，以blob类型存储

aes\_decrypt(str,key) 返回用密钥key对字符串str利用高级加密标准算法解密后的结果

decode(str,key) 使用key作为密钥解密加密字符串str

encrypt(str,salt) 使用unixcrypt()函数，用关键词salt(一个可以惟一确定口令的字符串，就像钥匙一样)加密字符串str

encode(str,key) 使用key作为密钥加密字符串str，调用encode()的结果是一个二进制字符串，它以blob类型存储

md5() 计算字符串str的md5校验和

password(str) 返回字符串str的加密版本，这个加密过程是不可逆转的，和unix密码加密过程使用不同的算法。

sha() 计算字符串str的安全散列算法(sha)校验和

示例：

select encrypt('root','salt');

select encode('xufeng','key');

select decode(encode('xufeng','key'),'key');#加解密放在一起

select aes\_encrypt('root','key');

select aes\_decrypt(aes\_encrypt('root','key'),'key');

select md5('123456');

select sha('123456');

## 控制流函数

mysql有4个函数是用来进行条件操作的，这些函数可以实现sql的条件逻辑，允许开发者将一些应用程序业务逻辑转换到数据库后台。

mysql控制流函数：

case when[test1] then [result1]...else [default] end如果testn是真，则返回resultn，否则返回default

case [test] when[val1] then [result]...else [default]end 如果test和valn相等，则返回resultn，否则返回default

if(test,t,f) 如果test是真，返回t；否则返回f

ifnull(arg1,arg2) 如果arg1不是空，返回arg1，否则返回arg2

nullif(arg1,arg2) 如果arg1=arg2返回null；否则返回arg1

这些函数的第一个是ifnull()，它有两个参数，并且对第一个参数进行判断。如果第一个参数不是null，函数就会向调用者返回第一个参数；如果是null,将返回第二个参数。

如：select ifnull(1,2), ifnull(null,10),ifnull(4\*null,'false');

nullif()函数将会检验提供的两个参数是否相等，如果相等，则返回null，如果不相等，就返回第一个参数。

如：select nullif(1,1),nullif('a','b'),nullif(2+3,4+1);

和许多脚本语言提供的if()函数一样，mysql的if()函数也可以建立一个简单的条件测试，这个函数有三个参数，第一个是要被判断的表达式，如果表达式为真，if()将会返回第二个参数，如果为假，if()将会返回第三个参数。

如：selectif(1<10,2,3),if(56>100,'true','false');

if()函数在只有两种可能结果时才适合使用。然而，在现实世界中，我们可能发现在条件测试中会需要多个分支。在这种情况下，mysql提供了case函数，它和php及perl语言的switch-case条件例程一样。

case函数的格式有些复杂，通常如下所示：

case [expression to be evaluated]

when [val 1] then [result 1]

when [val 2] then [result 2]

when [val 3] then [result 3]

......

when [val n] then [result n]

else [default result]

end

这里，第一个参数是要被判断的值或表达式，接下来的是一系列的when-then块，每一块的第一个参数指定要比较的值，如果为真，就返回结果。所有的when-then块将以else块结束，当end结束了所有外部的case块时，如果前面的每一个块都不匹配就会返回else块指定的默认结果。如果没有指定else块，而且所有的when-then比较都不是真，mysql将会返回null。

case函数还有另外一种句法，有时使用起来非常方便，如下：

case

when [conditional test 1] then [result 1]

when [conditional test 2] then [result 2]

else [default result]

end

这种条件下，返回的结果取决于相应的条件测试是否为真。

示例：

mysql>select case 'green'

when 'red' then 'stop'

when 'green' then 'go' end;

select case 9 when 1 then 'a' when 2 then 'b' else 'n/a' end;

select case when (2+2)=4 then 'ok' when(2+2)<>4 then 'not ok' end asstatus;

select name,if((isactive = 1),'已激活','未激活') as result fromuserlogininfo;

select fname,lname,(math+sci+lit) as total,

case when (math+sci+lit) < 50 then 'd'

when (math+sci+lit) between 50 and 150 then 'c'

when (math+sci+lit) between 151 and 250 then 'b'

else 'a' end

as grade from marks;

select if(encrypt('sue','ts')=upass,'allow','deny') as loginresultfrom users where uname = 'sue';#一个登陆验证

## 格式化函数

date\_format(date,fmt) 依照字符串fmt格式化日期date值

format(x,y) 把x格式化为以逗号隔开的数字序列，y是结果的小数位数

inet\_aton(ip) 返回ip地址的数字表示

inet\_ntoa(num) 返回数字所代表的ip地址

time\_format(time,fmt) 依照字符串fmt格式化时间time值

其中最简单的是format()函数，它可以把大的数值格式化为以逗号间隔的易读的序列。

示例：

select format(34234.34323432,3);

select date\_format(now(),'%w,%d %m %y %r');

select date\_format(now(),'%y-%m-%d');

select date\_format(19990330,'%y-%m-%d');

select date\_format(now(),'%h:%i %p');

select inet\_aton('10.122.89.47');

select inet\_ntoa(175790383);

## 类型转化函数

为了进行数据类型转化，mysql提供了cast()函数，它可以把一个值转化为指定的数据类型。类型有：binary,char,date,time,datetime,signed,unsigned

示例：

select cast(now() as signed integer),curdate()+0;

select 'f'=binary 'f','f'=cast('f' as binary);

## 系统信息函数

database() 返回当前数据库名

benchmark(count,expr) 将表达式expr重复运行count次

connection\_id() 返回当前客户的连接id

found\_rows() 返回最后一个select查询进行检索的总行数

user()或system\_user() 返回当前登陆用户名

version() 返回mysql服务器的版本

示例：

select database(),version(),user();

selectbenchmark(9999999,log(rand()\*pi()));#该例中,mysql计算log(rand()\*pi())表达式9999999次。

### 三级标题

#### 四级标题

#### xxx

# 一级标

## 二级标题

### 三级标题

#### 四级标题

#### Xxx

# 练习

1.显示所有首字母大写的雇员姓名

   SELECT ENAME FROM EMP WHERE INITCAP(SUBSTR(ENAME,1,1))=SUBSTR(ENAME,1,1);

2.显示所有雇员姓名的前三个字符

   SELECT SUBSTR(ENAME,1,3) FROM EMP;

3.显示所有雇员姓名将‘A’ 替换为‘a’

   SELECT REPLACE(ENAME,'A','a') FROM EMP;

4.显示所有员工直接上级的姓名

   SELECT E.ENAME 员工姓名, P.ENAME 员工上级 FROM EMP E,EMP P WHERE E.MGR=P.EMPNO;

5.取得所有员工的工资等级

  SELECT E.ENAME,S.GRADE FROM EMP E,SALGRADE S WHERE E.SAL BETWEEN S.LOSAL AND HISAL;

6.取得薪水最高的部门编号

 SELECT E.SAL 最高工资,D.DEPTNO 部门编号 FROM EMP E,DEPT D WHERE E.SAL=(SELECT MAX(SAL) FROM EMP ) AND E.DEPTNO=D.DEPTNO;

7.取得每个部门平均工资的薪水等级

   SELECT D.SAL 部门平均工资,S.GRADE 工资等级排名 FROM (SELECT AVG(SAL) SAL FROM EMP GROUP BY DEPTNO) D,SALGRADE S WHERE D.SAL BETWEEN LOSAL AND HISAL;

8.取得普通员工（没有在mgr字段上出现的） 的最高薪水

  SELECT MAX(E1.SAL) FROM EMP E1 WHERE E1.EMPNO<>E1.MGR;

9.取得平均薪水最高的部门编号

   SELECT E1.DEPTNO ,E1.SALS FROM (SELECT AVG(SAL),DEPTNO SALS FROM EMP GROUP BY DEPTNO ) E,  (SELECT AVG(SAL),DEPTNO SALS FROM EMP GROUP BY DEPTNO ) E1 WHERE MAX(E.SALS)=E1.SALS;

10.取得薪水最高排名的第六名到第十名的员工信息

    SELECT E.\* FROM (SELECT ROWNUM R,EMP.SAL FROM EMP ORDER BY SAL) E WHERE E.R>6 AND E.R<10;

11.取得每个薪水等级有多少员工

   SELECT GRADE 工资等级,COUNT(S.GRADE) 当前工资等级人数 FROM EMP E,SALGRADE S WHERE E.SAL BETWEEN S.LOSAL AND S.HISAL GROUP BY S.GRADE;

12.将员工的入职日期按照年和月排序，如果月份相同就按照日期排序

    SELECT ENAME, TO\_CHAR(HIREDATE,'YYYY') YEAR,TO\_CHAR(HIREDATE,'MM') MONTH,TO\_CHAR(HIREDATE,'DD') DAY FROM EMP ORDER BY YEAR DESC,MONTH DESC,DAY DESC;

13.查询和scott相同部门的员工姓名和入职日期

     SELECT ENAME,HIREDATE FROM EMP WHERE DEPTNO=(SELECT DEPTNO FROM EMP WHERE ENAME='SCOTT');

14.查询部门在newyork的员工姓名和部门名称

    SELECT E.ENAME,D.DNAME FROM EMP E,DEPT D WHERE D.LOC='NEW YORK' AND E.DEPTNO=D.DEPTNO;

15.查询上司是king的员工姓名和工资

   SELECT  E.\* FROM EMP E,EMP E1 WHERE  E1.ENAME='KING' AND E.MGR=E1.EMPNO;

16.查询员工姓名包含有‘u’的部门的员工信息

   SELECT \* FROM EMP WHERE DEPTNO=(SELECT DEPTNO  FROM EMP WHERE ENAME LIKE'%U%');

17.查询早于员工上级入职日期提前的员工信息

    SELECT E.ENAME 员工姓名,E.HIREDATE 员工入职日期,E1.HIREDATE 上级入职日期 FROM EMP E,EMP E1 WHERE E.HIREDATE<E1.HIREDATE AND E.MGR=E1.EMPNO;

18.列出从事同一个工作但在不同岗位的员工一种组合

  SELECT DISTINCT  E.ENAME,E.JOB,E1.ENAME FROM EMP E,EMP E1 WHERE E.JOB=E1.JOB AND E.DEPTNO!=E1.DEPTNO AND E.ENAME<E1.ENAME;

19.取得每个部门薪水最高的员工姓名

    SELECT ENAME,SAL,D.DNAME FROM EMP,DEPT D WHERE SAL IN(SELECT MAX(SAL) FROM EMP GROUP BY DEPTNO) AND D.DEPTNO=EMP.DEPTNO;

20.列出所有CLERK的姓名，部门名称，及其部门人数

    SELECT E.ENAME,D.DNAME,(SELECT COUNT(1) FROM EMP WHERE DEPTNO=D.DEPTNO) 部门人数 FROM EMP E,DEPT D WHERE E.DEPTNO=D.DEPTNO AND E.JOB='CLERK';

21.查出上级中工资超过3000的姓名

  SELECT P.\* FROM EMP E,EMP P WHERE E.MGR=P.EMPNO AND P.SAL>3000;