



Oracle SQL常用函数

📅 2016-09-06

文中例子涉及到的表数据：

```
1  SQL> select empno,ename,job, hiredate,sal,deptno,comm from lzp.emp;
2
3      EMPNO  ENAME      JOB          HIREDATE          SAL      DEPTNO      COMM
4  -----
5      7369 SMITH        CLERK        17-12月-80        800        20         null
6      7499 ALLEN        SALESMAN     20-2月 -81       3100        30        300
7      7521 WARD          SALESMAN     22-2月 -81       1250        30        500
8      7566 JONES        MANAGER     02-4月 -81       2975        20         null
9      7654 MARTIN       SALESMAN     28-9月 -81       8000        30       1400
10     7698 BLAKE        MANAGER     01-5月 -81       2850        30         null
11     7782 CLARK        MANAGER     09-6月 -81       2450        10         null
12     7788 SCOTT        ANALYST     19-4月 -87       3000        20         null
13     7839 KING          PRESIDENT  17-11月-81       5000        10         null
14     7844 TURNER       SALESMAN     08-9月 -81       1500        30          0
15     7876 ADAMS        CLERK       23-5月 -87       1100        20         null
16     7900 JAMES        CLERK       03-12月-81        950        30         null
17     7902 FORD          ANALYST     03-12月-81       3000        20         null
18     7934 MILLER       CLERK       23-1月 -82       1300        10         null
19
20  已选择14行。
```

字符函数

📌 字符串截取

```
1  SQL> select substr('mrbird',3,6) from dual;
2
3  SUBS
4  ----
5  bird
```

📌 查找子串位置

```
1  SQL> select instr('mrBird','Bird') from dual;
2
3  INSTR('MRBIRD','BIRD')
4  -----
5                      3
```

📌 字符串连接

```
1  /* 1. || */
2  SQL> select 'mr' || 'Bird' as result from dual;
3
4  RESULT
5  -----
6  mrBird
7  /* 2.concat */
8  SQL> select concat('mr','Bird') as result from dual;
9
10 RESULT
11 -----
12 mrBird
```

📌 去除空格

```
1  SQL >
2  SELECT
3      LTRIM ('  mrBird') l,
4      RTRIM ('mrBird  ') r,
5      TRIM ('  mrBird  ') T
6  FROM
```

```

7      dual;
8
9      L      R      T
10     -----
11  mrBird mrBird mrBird

```

去除前缀和后缀

```

1  SQL >
2  SELECT
3      TRIM (LEADING 'M' FROM 'MMMMMrBird') s1,
4      TRIM (TRAILING 'D' FROM 'mrBirdDDDD') s2,
5      TRIM ('M' FROM 'MMMrBirdMMMM') s3
6  FROM
7      dual;
8
9  S1      S2      S3
10     -----
11  mrBird mrBird mrBird

```

计算字符串长度

```

1  SQL> select length('mrbird') from dual;
2
3      LENGTH('MRBIRD')
4      -----
5                      6

```

initcap (首字母变大写), lower (变小写), upper (变大写)

```

1  SQL> select initcap('mrbird') s1, lower('MRBIRD') s2, upper('mrbird') s3 from dual;
2
3  S1      S2      S3
4  -----
5  Mrbird mrbird MRBIRD

```

替换

```

1  SQL> select replace('mrXiaoniao', 'Xiaoniao', 'Bird') from dual;
2
3  REPLAC
4  -----
5  mrBird

```

decode[实现if..then 逻辑] 注:第一个是表达式,最后一个是不满足任何一个条件的值

```

1  SQL>
2  SELECT
3      DECODE (
4          deptno,
5          10,
6          '人力资源部',
7          20,
8          '软件开发部',
9          30,
10         '市场部',
11         '其他神秘部门'
12     ) deptName,
13     deptno,
14     ename
15  FROM
16      lzp.emp
17  WHERE
18      empno >= 7900;
19
20  DEPTNAME      DEPTNO  ENAME
21  -----
22  市场部          30  JAMES
23  软件开发部      20  FORD
24  人力资源部      10  MILLER

```

📌 case[实现switch ..case 逻辑]

```
1  SQL>
2  SELECT
3      CASE
4  WHEN sal < 2000 THEN
5      '被剥削的人'
6  WHEN sal < 3000 THEN
7      '被压榨的人'
8  WHEN sal < 5000 THEN
9      '普通的人'
10 ELSE
11     '努力的人'
12 END meno,
13 sal,
14 ename
15 FROM
16     lzp.emp
17 WHERE
18     deptno = 30
19 ORDER BY
20     sal ASC;
21
22  MENO                SAL  ENAME
23  -----
24  被剥削的人          950  JAMES
25  被剥削的人          1250 WARD
26  被剥削的人          1500 TURNER
27  被压榨的人          2850 BLAKE
28  普通的人            3100 ALLEN
29  努力的人            8000 MARTIN
30
31  已选择6行。
```

📌 十进制和ASCII互相转换

```
1  SQL> select ascii('A'),chr(65) from dual;
2
3  ASCII('A') C
4  -----
5           65 A
```

日期函数

TO_DATE格式(以时间: 2007-11-02 13:45:25 为例)

Year			
yy	two digits	两位年	显示值:07
yyy	three digits	三位年	显示值:007
yyyy	four digits	四位年	显示值:2007
Month			
mm	number	两位月	显示值:11
mon	abbreviated	字符集表示	显示值:11月,若是英文版,显示nov
month	spelled out	字符集表示	显示值:11月,若是英文版,显示november
Day			
dd	number	当月第几天	显示值:02
ddd	number	当年第几天	显示值:306
dy	abbreviated	当周第几天简写	显示值:星期五,若是英文版,显示fri
day	spelled out	当周第几天全写	显示值:星期五,若是英文版,显示friday

2018/4/4

Oracle SQL常用函数 | MrBird

ddspth	spelled out	当月第几天英文	显示值:second
Hour			
hh	two digits	12小时进制	显示值:01
hh24	two digits	24小时进制	显示值:13
Minute			
mi/mm	two digits	60进制	显示值:45
Second			
ss	two digits	60进制	显示值:25
Others			
Q	digit	季度	显示值:4
WW	digit	当年第几周	显示值:44
W	digit	当月第几周	显示值:1

24小时格式下时间范围为： 0:00:00 - 23:59:59....

12小时格式下时间范围为： 1:00:00 - 12:59:59

日期字符串相互转换

时间转字符串:

Year :

```
1  SQL>
2  SELECT
3      TO_CHAR (SYSDATE, 'yy') yy,
4      TO_CHAR (SYSDATE, 'yyy') yyy,
5      TO_CHAR (SYSDATE, 'yyyy') yyyy
6  FROM
7      dual;
8
9  YY YYY YYYY
10 -- --- ----
11 16 016 2016
```

Month :

```
1  SQL>
2  SELECT
3      TO_CHAR (SYSDATE, 'mm') mm,
4      TO_CHAR (SYSDATE, 'mon') mon,
5      TO_CHAR (SYSDATE, 'month') month
6  FROM
7      dual;
8
9  MM MON      MONTH
10 -- - - - - - - - -
11 09 9月      9月
```

Day :

```
1  SQL>
2  SELECT
3      TO_CHAR (SYSDATE, 'dd') dd,
4      TO_CHAR (SYSDATE, 'ddd') ddd,
5      TO_CHAR (SYSDATE, 'dy') dy,
6      TO_CHAR (SYSDATE, 'day') DAY,
7      TO_CHAR (SYSDATE, 'ddspth') ddspth
8  FROM
9      dual;
10
11 DD DDD DY      DAY      DDSPTH
```

```

12  -- --- -----
13  06 250 tue      tuesday      sixth

```

Hour :

```

1  SQL>
2  SELECT
3      TO_CHAR (SYSDATE, 'hh') hh,
4      TO_CHAR (SYSDATE, 'hh24') hh24
5  FROM
6      dual;
7
8  HH HH24
9  -- --
10 05 17

```

Minute,Second略

季度, 周:

```

1  SQL>
2  SELECT
3      TO_CHAR (SYSDATE, 'Q') Q,
4      TO_CHAR (SYSDATE, 'WW') WW,
5      TO_CHAR (SYSDATE, 'W') W
6  FROM
7      dual;
8
9  Q WW W
10 - - -
11 3 36 1

```

字符串转时间:

```

1  SQL> select to_date('2016-09-06','yyyy-MM-dd') time from dual;
2
3  TIME
4  -----
5  06-9月 -16

```

📌 next_day

返回下个星期的日期,day为1-7或星期日-星期六,1表示星期日,也可以用英文表示星期。

```

1  SQL> select next_day(sysdate,'monday') time from dual;
2
3  TIME
4  -----
5  12-9月 -16

```

📌 两个日期相差天数

```

1  SQL> select floor(sysdate - to_date('20020405','yyyymmdd')) time from dual;
2
3      TIME
4  -----
5      5268

```

📌 months_between

```

1  SQL>
2  SELECT
3      MONTHS_BETWEEN (
4          TO_DATE ('2016-09-01', 'yyyy-MM-dd'),
5          TO_DATE ('2016-01-01', 'yyyy-MM-dd')
6      ) months
7  FROM
8      dual;
9
10     MONTHS
11  -----
12           8

```

📌 round [舍入到最接近的日期] (day:舍入到最接近的星期日)

```

1  SQL>
2  SELECT
3      SYSDATE S1,
4      ROUND (SYSDATE) S2,
5      ROUND (SYSDATE, 'year') YEAR,
6      ROUND (SYSDATE, 'month') MONTH,
7      ROUND (SYSDATE, 'day') DAY
8  FROM
9      dual;
10
11  S1                S2                YEAR                MONTH                DAY
12  -----
13  07-9月 -16        07-9月 -16        01-1月 -17        01-9月 -16        04-9月 -16

```

📌 计算时间差

注:oracle时间差是以天数为单位,所以换算成年月,日

```

1  SQL>
2  SELECT
3      FLOOR (
4          TO_NUMBER (
5              SYSDATE - TO_DATE (
6                  '2007-11-02 15:55:03',
7                  'yyyy-mm-dd hh24:mi:ss'
8              )
9          ) / 365
10 ) AS spanYears
11 FROM
12     dual;
13
14  SPANYEARS
15  -----
16          8
17
18  SQL>
19  SELECT
20      CEIL (
21          MONTHS_BETWEEN (
22              SYSDATE,
23              TO_DATE (
24                  '2007-11-02 15:55:03',
25                  'yyyy-mm-dd hh24:mi:ss'
26              )
27          ) AS spanMonths
28 FROM
29     dual;
30
31  SPANMONTHS
32  -----
33          107
34
35  SQL>
36  SELECT
37      FLOOR (
38          TO_NUMBER (
39              SYSDATE - TO_DATE (
40                  '2007-11-02 15:55:03',
41                  'yyyy-mm-dd hh24:mi:ss'
42              )
43          ) AS spanDays
44 FROM
45     dual;
46
47  SPANDAYS
48  -----
49          3231
50
51  SQL>
52  SELECT
53      FLOOR (
54          TO_NUMBER (
55              SYSDATE - TO_DATE (

```

```

55         '2007-11-02 15:55:03',
56         'yyyy-mm-dd hh24:mi:ss'
57     )
58     ) * 24
59 ) AS spanHours
60 FROM
61     dual;
62
63 SPANHOURS
64 -----
65         77562
66 /* 分秒略，以此类推 */

```

📌 查找月的最后一天

```

1  SQL> select last_day(sysdate) from dual;
2
3  LAST_DAY(SYSDA
4  -----
5  30-9月 -16

```

📌 ADD_MONTHS(date, i)

对给定的日期加上给定的月，查看每个员工入职20周年

```

1  SQL> select ename,add_months(hiredate,12*20) from lzp.emp;
2
3  ENAME          ADD_MONTHS(HIR
4  -----
5  SMITH          17-12月-00
6  ALLEN          20-2月 -01
7  WARD           22-2月 -01
8  JONES          02-4月 -01
9  MARTIN         28-9月 -01
10 BLAKE          01-5月 -01
11 CLARK          09-6月 -01
12 SCOTT          19-4月 -07
13 KING           17-11月-01
14 TURNER         08-9月 -01
15 ADAMS          23-5月 -07
16 JAMES          03-12月-01
17 FORD           03-12月-01
18 MILLER         23-1月 -02
19
20  已选择14行。

```

📌 EXTRACT

单独获取指定时间的年或月或日

```

1  SQL> select extract(year from sysdate) from dual;
2
3  EXTRACT(YEARFROMSYSDATE)
4  -----
5  2016

```

数字函数

📌 取整函数 (ceil 向上取整,floor 向下取整)

```

1  SQL> select ceil(66.6) N1,floor(66.6) N2 from dual;
2
3  N1          N2
4  -----
5  67          66

```

📌 取幂(power) 和 求平方根(sqrt)

```

1  SQL> select power(3,2) N1,sqrt(9) N2 from dual;
2
3  N1          N2
4  -----

```

```
5          9          3
```

求余

```
1  SQL> select mod(9,5) from dual;
2
3      MOD(9,5)
4  -----
5              4
```

返回固定小数位数 (round:四舍五入, trunc:直接截断)

```
1  SQL> select round(66.667,2) N1,trunc(66.667,2) N2 from dual;
2
3          N1          N2
4  -----
5      66.67      66.66
```

返回值的符号 (正数返回为1,负数为-1)

```
1  SQL> select sign(-32),sign(293) from dual;
2
3      SIGN(-32)  SIGN(293)
4  -----
5              -1          1
```

LEAST、GREATEST

两个函数都可以有多个参数值，但参数类型必须一致，返回结果是参数列表中最大或最小的值。 在比较之前，在参数列表中第二个以后的参数会被隐含的转换为第一个参数的数据类型，所以如果可以转换，则继续比较，如果不能转换将会报错。

```
1  SQL> select least(1,2,3),greatest(sysdate,'01-10月-16') from dual;
2
3      LEAST(1,2,3)  GREATEST(SYSDA
4  -----
5              1 01-10月-16
```

其他函数

vsiz: 返回表达式所需的字节数

```
1  SQL> select vsiz('mrBird123') from dual;
2
3      VSIZE('MRBIRD123')
4  -----
5              9
```

nvl(ex1,ex2):

ex1值为空则返回ex2,否则返回该值本身ex1 (常用) 例: 如果雇员没有佣金, 将显示0, 否则显示佣金

```
1  SQL> select comm,nvl(comm,0) from lzp.emp;
2
3      COMM  NVL(COMM,0)
4  -----
5              0
6      300      300
7      500      500
8              0
9      1400     1400
10              0
11              0
12              0
13              0
14              0
15              0
16              0
17              0
18              0
19
20  已选择14行。
```


📌 nullif(ex1,ex2):

```
1  SQL> select nullif(88,88) from dual;
2
3  NULLIF(88,88)
4  -----
5
6  SQL> select nullif(88,123) from dual;
7
8  NULLIF(88,123)
9  -----
10         88
11
12 SQL> select nullif('mrBird','mrBird') from dual;
13
14  NULLIF
15  -----
```

📌 nvl2(ex1,ex2,ex3) :

如果ex1不为空，显示ex2,否则显示ex3

注：部分内容来自脚本之家

DataBase # Oracle # Oracle 11g

◀ Java Encryption - Decryption

Java 反射机制 ▶

昵称 (必填)

邮箱 (必填)

网址(http://)

MarkDown is support, powered by Valine

回复