

一棵树的房间

主页

博客

相册

个人档案

好友

查看文章

oracle over() 函数

2008/04/09 09:37
Oracle 语法之 OVER (PARTITION BY ..)

select * from test

数据:

A B C
1 1 1
1 2 2
1 3 3
2 2 5
3 4 6

---将B列值相同的对应的C 列值加总
select a,b,c, SUM(C) OVER (PARTITION BY B) C_Sum
from test

A B C C_SUM
1 1 1 1
1 2 2 7
2 2 5 7
1 3 3 3
3 4 6 6

PARTITION BY B : 把B列进行分割(本人觉得分割有点类似于分组group by,但不等于分组)

我们先来分析一下 C_SUM 的结果是怎么出来的.

B列总共有四个值: 1 2 3 4. 第一行的C_SUM值为1, 因为所对应的C列值为1; 第二行的C_SUM值为7. 为什么呢?因为B列中值为2的行总共有两列,所以 SUM(C) 的值为 7 (等于两个C列的相加 2+5 = 7). B列下面的值3 4原理同第一列的值1.

---如果不需要已某个列的值分割,那就要用 null

eg: 就是将C的列值summary 放在每行后面

select a,b,c, SUM(C) OVER (PARTITION BY null) C_Sum
from test

A B C C_SUM
1 1 1 17
1 2 2 17
1 3 3 17
2 2 5 17
3 4 6 17

因为没有分割列,所以C_SUM的值为所有C列的值的和. 17=1+2+3+5+6.

SQL> select deptno,ename,sal
2 from emp
3 order by deptno;

DEPTNO	ENAME	SAL
10	CLARK	2450
	KING	5000

相关文章

- [Oracle row_number\(\) over\(\)解析...](#)
- [oracle的分析函数over 及开窗函数...](#)
- [oracle中over\(\)开窗函数的理解](#)
- [oracle 的over函数](#)
- [oracle分析函数row_number\(\) o ve...](#)
- [利用ORACLE的MINUS函数和O VER函...](#)
- [oracle中rank\(\),dense_rank,part...](#)
- [oracle中使用rank over partitio...](#)
- [Oracle Syntax - SUM and Over](#)
- [oracle over\(\)&union 查询](#)

[更多>>](#)

MILLER	1300
20 SMITH	800
ADAMS	1100
FORD	3000
SCOTT	3000
JONES	2975
30 ALLEN	1600
BLAKE	2850
MARTIN	1250
JAMES	950
TURNER	1500
WARD	1250

已选择14行。

2.先来一个简单的，注意over(...)条件的不同，
使用 sum(sal) over (order by ename)... 查询员工的薪水"连续"求和，
注意over (order by ename)如果没有order by 子句，求和就不是"连续"的，
放在一起，体会一下不同之处：

```
SQL> select deptno,ename,sal,
2 sum(sal) over (order by ename) 连续求和,
3 sum(sal) over () 总和,      -- 此处sum(sal) over () 等同于sum(sal)
4 100*round(sal/sum(sal) over (),4) "份额(%)"
5 from emp
6 /
```

DEPTNO	ENAME	SAL	连续求和	总和	份额(%)
20	ADAMS	1100	1100	29025	3.79
30	ALLEN	1600	2700	29025	5.51
30	BLAKE	2850	5550	29025	9.82
10	CLARK	2450	8000	29025	8.44
20	FORD	3000	11000	29025	10.34
30	JAMES	950	11950	29025	3.27
20	JONES	2975	14925	29025	10.25
10	KING	5000	19925	29025	17.23
30	MARTIN	1250	21175	29025	4.31
10	MILLER	1300	22475	29025	4.48
20	SCOTT	3000	25475	29025	10.34
20	SMITH	800	26275	29025	2.76
30	TURNER	1500	27775	29025	5.17
30	WARD	1250	29025	29025	4.31

已选择14行。

3.使用子分区查出各部门薪水连续的总和。注意按部门分区。注意over(...)条件的不同，
sum(sal) over (partition by deptno order by ename) 按部门"连续"求总和
sum(sal) over (partition by deptno) 按部门求总和
sum(sal) over (order by deptno, ename) 不按部门"连续"求总和
sum(sal) over () 不按部门，求所有员工总和，效果等同于sum(sal)。

```
SQL> select deptno,ename,sal,
2 sum(sal) over (partition by deptno order by ename) 部门连续求和,--各部门的薪水"连续"求和
3 sum(sal) over (partition by deptno) 部门总和, -- 部门统计的总和,同一部门总和不变
4 100*round(sal/sum(sal) over (partition by deptno),4) "部门份额(%)" ,
5 sum(sal) over (order by deptno,ename) 连续求和, --所有部门的薪水"连续"求和
6 sum(sal) over () 总和, -- 此处sum(sal) over () 等同于sum(sal)，所有员工的薪水总和
7 100*round(sal/sum(sal) over (),4) "总份额(%)"
8 from emp
9 /
```

DEPTNO	ENAME	SAL	部门连续求和	部门总和	部门份额(%)	连续求和	总和	总份额(%)
10	CLARK	2450	2450	8750	28	2450	29025	8.44
	KING	5000	7450	8750	57.14	7450	29025	17.23
	MILLER	1300	8750	8750	14.86	8750	29025	4.48
20	ADAMS	1100	1100	10875	10.11	9850	29025	3.79
	FORD	3000	4100	10875	27.59	12850	29025	10.34
	JONES	2975	7075	10875	27.36	15825	29025	10.25
	SCOTT	3000	10075	10875	27.59	18825	29025	10.34
	SMITH	800	10875	10875	7.36	19625	29025	2.76
30	ALLEN	1600	1600	9400	17.02	21225	29025	5.51
	BLAKE	2850	4450	9400	30.32	24075	29025	9.82
	JAMES	950	5400	9400	10.11	25025	29025	3.27
	MARTIN	1250	6650	9400	13.3	26275	29025	4.31
	TURNER	1500	8150	9400	15.96	27775	29025	5.17
	WARD	1250	9400	9400	13.3	29025	29025	4.31

已选择14行。

4.来一个综合的例子，求和规则有按部门分区的，有不分区的例子
SQL> select deptno,ename,sal,sum(sal) over (partition by deptno order by sal) dept_sum,
2 sum(sal) over (order by deptno,sal) sum
3 from emp;

DEPTNO	ENAME	SAL	DEPT_SUM	SUM

10	MILLER	1300	1300	1300
	CLARK	2450	3750	3750
	KING	5000	8750	8750
20	SMITH	800	800	9550
	ADAMS	1100	1900	10650
	JONES	2975	4875	13625
	SCOTT	3000	10875	19625
	FORD	3000	10875	19625
30	JAMES	950	950	20575
	WARD	1250	3450	23075
	MARTIN	1250	3450	23075
	TURNER	1500	4950	24575
	ALLEN	1600	6550	26175
	BLAKE	2850	9400	29025

已选择14行。

5.来一个逆序的，即部门从大到小排列，部门里各员工的薪水从高到低排列，累计和的规则不变。

SQL> select deptno,ename,sal,
2 sum(sal) over (partition by deptno order by deptno desc,sal desc) dept_sum,
3 sum(sal) over (order by deptno desc,sal desc) sum
4 from emp;

DEPTNO	ENAME	SAL	DEPT_SUM	SUM

30	BLAKE	2850	2850	2850
	ALLEN	1600	4450	4450
	TURNER	1500	5950	5950
	WARD	1250	8450	8450
	MARTIN	1250	8450	8450
	JAMES	950	9400	9400
20	SCOTT	3000	6000	15400
	FORD	3000	6000	15400
	JONES	2975	8975	18375
	ADAMS	1100	10075	19475
	SMITH	800	10875	20275
10	KING	5000	5000	25275
	CLARK	2450	7450	27725
	MILLER	1300	8750	29025

已选择14行。

6.体会：在"... from emp;"后面不要加order by 子句，使用的分析函数的(partition by deptno order by sal) 里已经有排序的语句了，如果再在句尾添加排序子句，一致倒罢了，不一致，结果就令人费劲了。如：

SQL> select deptno,ename,sal,sum(sal) over (partition by deptno order by sal) dept_sum,
2 sum(sal) over (order by deptno,sal) sum
3 from emp
4 order by deptno desc;

DEPTNO	ENAME	SAL	DEPT_SUM	SUM

30	JAMES	950	950	20575
	WARD	1250	3450	23075
	MARTIN	1250	3450	23075
	TURNER	1500	4950	24575
	ALLEN	1600	6550	26175
	BLAKE	2850	9400	29025
20	SMITH	800	800	9550
	ADAMS	1100	1900	10650
	JONES	2975	4875	13625
	SCOTT	3000	10875	19625
	FORD	3000	10875	19625

10 MILLER	1300	1300	1300
CLARK	2450	3750	3750
KING	5000	8750	8750

已选择14行

类别: [Oracle](#) | [添加到收藏](#) | [浏览\(511\)](#) | [评论 \(1\)](#)

上一篇: [一公司的oracle sql面试题](#) 下一篇: [一个简单的题目](#)

最近读者:



登录后, 您就出现在这里。



[蟹蟹悠悠](#)



[superxzl](#)



[dengtujian](#)



[wgqweb](#)



[maincoolbo](#)



[dapao9830](#)



[尼是唯一](#)



[htl258](#)

1

网友评论:

- 1
- 网友:seecle
arly

2009/07/29 12:49 | [回复](#)
非常感谢!

发表评论:

姓 名:

[注册](#) | [登录](#)

网址或邮箱:

(选填)

:

内 容:

插入表情

验证码:

请点击后输入四位验证码, 字母不区分大小写

