

[数据库] Oracle单表查询总数及百分比和数据横向纵向连接

原创 Eastmount 2016-06-14 02:41:20 7683 收藏 1

展开



Python+TensorFlow人工智能

该专栏为人工智能入门专栏，采用Python3和TensorFlow实现人工智能相关算法。前期介绍安装流程、基础语法、



Eastmount

¥9.90

订阅

这是最近项目关于SQL语句的，本文简单记录并总结以下几个知识点：

- 1.如何统计一张表中某个字段的总数，如不同"专业"的学生数及所占百分比；
- 2.如何联系另一张表进行查询某个字段的总数及百分比；
- 3.简单介绍decode防止分母为0和trunc保留小数位数等函数；
- 4.通常复杂的SQL语句会涉及到查询结果横向连接和纵向连接，这里进行介绍。

最近买了本《Oracle查询优化改写技巧与案例-有教无类 落落》，推荐大家也阅读下。后面我也会补充一些相关数据
的知识，希望对大家有所帮助吧！文章还是以基础知识为主，同时主要是解决实际的问题。同时我采用自问自答的新叙述
方法进行介绍~

0. 前言

假如现在存在如下图所示的一张表，这种表在数据库中是很常见，主要包括字段（序号，教师姓名，单位名称，性
别，学历）。该表的信息为：TEST_TEACHER (id, name, dw_name, sex, degree)。

ID	NAME	DW_NAME	SEX	DEGREE
001	王一	软件学院	男	博士
002	王二	软件学院	男	博士
003	李一	软件学院	男	博士
004	李二	软件学院	男	硕士
005	张一	计算机学院	女	硕士
006	张二	计算机学院	男	博士
007	张三	计算机学院	男	学士
008	杨一	数学学院	男	学士
009	杨二	数学学院	女	博士
010	杨三	数学学院	女	硕士

1.问题一：如何在单表中统计总数及比例

现在需要统计各个单位的教师人数，及该单位的不同学历的、性别的人数。这是非常常见的问题，当时项目中统计的
内容很多，包括：年龄、学缘、职称、专业等等。

这很简单并且方法很多，常见的主要使用子查询或group by分组。

group by:

```
select DW_NAME, COUNT(DW_NAME) as 人数 from TEST_TEACHER
group by DW_NAME
order by DW_NAME;
```

输出结果如下图所示：其中order by是按汉语拼音排序输出。

DW_NAME	人数
计算机学院	3
软件学院	4
数学学院	3

如果现在需要统计各个学院的男性、女性教师人数，博士、硕士、学士的教师人数，需要怎么办呢？因为它都是同一张表的信息，一种方法是使用子查询，这里介绍另外一种方法，通过CASE WHEN THEN实现。

CASE WHEN THEN:

```
select DW_NAME, COUNT(DW_NAME) as SUM,
       COUNT(CASE WHEN SEX='男' THEN 1 END) as Man, COUNT(CASE WHEN SEX='女' THEN 1 END) as Women,
       COUNT(CASE WHEN DEGREE='博士' THEN 1 END) as BS, COUNT(CASE WHEN DEGREE='硕士' THEN 1 END) as SS
from TEST_TEACHER
group by DW_NAME order by DW_NAME;
```

输出结果如下图所示：表示如何SEX为"男"，统计加1。

DW_NAME	SUM	MAN	WOMEN	BS	SS
计算机学院	3	2	1	1	1
软件学院	4	4	0	3	1
数学学院	3	1	2	1	1

现在如果需要统计每个学院所占所有人数的比例，怎么计算呢？

传统方法该步骤是分别统计两个数，再通过后台Java或C++进行处理的，或者是使用除法：学院总数/老师表总数，但是已经使用了分组group by，如果再使用一遍统计TEST_TEACHER总数就会报错。

这里使用的是Oracle的函数ratio_to_report() over()实现。

ratio_to_report:

```
select DW_NAME, COUNT(DW_NAME) as SUM,
       ratio_to_report(COUNT(1)) OVER(),
       COUNT(CASE WHEN SEX='男' THEN 1 END) as Man, COUNT(CASE WHEN SEX='女' THEN 1 END) as Women
from TEST_TEACHER
group by DW_NAME order by DW_NAME;
```

输出结果如下图所示：其中计算机3个老师，总数10个老师，比例占0.3。

学院	总数	比例	男	女
计算机学院	3	0.3	2	1
软件学院	4	0.4	4	0
数学学院	3	0.3	1	2

2.问题二：联系另一张表统计总数及比例

在设计数据库过程中，通常会将教师的信息设置为一张表，学院信息会存在另一张表中，同理授课信息、发表论文信息都会存在单独的表，再通过教师姓名（或教师编号）连接查询。

所以这里讲述第二种统计总数及比例的方法，假设还存在一张学院表，如下图所示：该表TEST_DEP结构为(DW_NAME, DW_CODE, YEAR, ADDR)。

DW_NAME	DW_CODE	YEAR	ADDR
软件学院	08111003	2014	软件楼
计算机学院	07111002	2014	计算机中心
数学学院	06111001	2013	数学楼

通常是遍历该单位表，然后子查询连接教师表，统计不同单位的人数信息。同样统计专业教师信息、教师发表论文、

学院发表论文都是这种方法。

ID	NAME	DW_NAME	SEX	DEGREE
001	王一	软件学院	男	博士
002	王二	软件学院	男	博士
003	李一	软件学院	男	博士
004	李二	软件学院	男	硕士
005	张一	计算机学院	女	硕士
006	张二	计算机学院	男	博士
007	张三	计算机学院	男	学士
008	杨一	数学学院	男	学士
009	杨二	数学学院	女	博士
010	杨三	数学学院	女	硕士

两表连接子查询：统计总数

```
select t2.DW_NAME,
      (select COUNT(*) from TEST_TEACHER t1 where t1.DW_NAME=t2.DW_NAME) as 总数,
      (select COUNT(*) from TEST_TEACHER t1 where t1.DW_NAME=t2.DW_NAME and SEX='男') as 男,
      (select COUNT(*) from TEST_TEACHER t1 where t1.DW_NAME=t2.DW_NAME and SEX='女') as 女
from TEST_DEP t2;
```

输出结果如下图所示：这种方法是非常常见的一种统计方法，而不是仅仅通过单表，因为数据库往往都会通过外键联系其他表。

DW_NAME	总数	男	女
软件学院	4	4	0
计算机学院	3	2	1
数学学院	3	1	2

这种两表联系的另一个优势是更方便计算百分比、比例等，因为同一张表使用两次时很麻烦的，如 "select (select * from t1) from t1;"这里使用子查询计算比例代码如下。

两表连接子查询：统计比例

```
select t2.DW_NAME,
      (select COUNT(*) from TEST_TEACHER t1 where t1.DW_NAME=t2.DW_NAME) as 总数,
      ((select COUNT(*) from TEST_TEACHER t1 where t1.DW_NAME=t2.DW_NAME)
      / (select COUNT(*) from TEST_TEACHER)) as 总数比例,
      (select COUNT(*) from TEST_TEACHER t1 where t1.DW_NAME=t2.DW_NAME and SEX='男') as 男,
      (select COUNT(*) from TEST_TEACHER t1 where t1.DW_NAME=t2.DW_NAME and SEX='女') as 女,
      ((select COUNT(*) from TEST_TEACHER t1 where t1.DW_NAME=t2.DW_NAME and SEX='男')
      / (select COUNT(*) from TEST_TEACHER t1 where t1.DW_NAME=t2.DW_NAME)) as 男教师比例
from TEST_DEP t2;
```

输出结果如下图所示：

DW_NAME	总数	总数比例	男	女	男教师比例
软件学院	4	0.4	4	0	1
计算机学院	3	0.3	2	1	0.6666666666666667
数学学院	3	0.3	1	2	0.3333333333333333

其中总共10个教师，计算机学院3人，所占比例=3/10=0.4，同时男教师比例为计算机学院男教师2人，女教师1人，故比例=2/3=0.666，这里使用的方法是子查询的除法。其实更常见的方法是获取总数，然后后台业务逻辑进行计算。

3.问题三： 除法防止分母为0及保留有效位数

如上图输出为0.6666667，此时需要保留有效位数，这里的方法可以是使用trunc()函数，整数可用floor()函数。
TRUNC(x[,y])功能: 计算截尾到y位小数的x值，y缺省为0，结果变为一个整数值。
trunc()是截断操作，floor(x)是小于或等于x的最大整数。

TRUNC() FLOOR():

```
select t2.DW_NAME,
       (select COUNT(*) from TEST_TEACHER t1 where t1.DW_NAME=t2.DW_NAME) as 总数,
       (select COUNT(*) from TEST_TEACHER t1 where t1.DW_NAME=t2.DW_NAME and SEX='男') as 男,
       (select COUNT(*) from TEST_TEACHER t1 where t1.DW_NAME=t2.DW_NAME and SEX='女') as 女,
       trunc(((select COUNT(*) from TEST_TEACHER t1 where t1.DW_NAME=t2.DW_NAME and SEX='男')
              / (select COUNT(*) from TEST_TEACHER t1 where t1.DW_NAME=t2.DW_NAME))*100,2) as 百分比
from TEST_DEP t2;
```

输出结果如下图所示:

DW_NAME	总数	男	女	百分比
软件学院	4	4	0	100
计算机学院	3	2	1	66.66
数学学院	3	1	2	33.33

其中输出为66.66，表示计算机学院男教师所占百分比，其中保留2位有效数字的方法为：trunc(统计男教师子查询 / 统计教师子查询*100, 2)。

Oracle中通常需要统计如男生占全班总人数比例等用法，此时如果分母为0，它会报错"*[Err] ORA-01476: divisor is equal to zero*"。那怎么办呢?

解决方法：使用函数decode，当分母为0时直接返回0，否则进行除法运算。

select a/b from c; 修改成如下即可：select decode(b, 0, 0, a/b) from c;

例如：decode(XF_ALL, 0, 0, trunc(XF_MATH / XF_ALL * 100, 1)) as BL

上面子查询除法使用decode的并结合trunc保留2位有效数字的SQL如下。

decode():

```
select t2.DW_NAME,
       (select COUNT(*) from TEST_TEACHER t1 where t1.DW_NAME=t2.DW_NAME) as 总数,
       (select COUNT(*) from TEST_TEACHER t1 where t1.DW_NAME=t2.DW_NAME and SEX='男') as 男,
       (select COUNT(*) from TEST_TEACHER t1 where t1.DW_NAME=t2.DW_NAME and SEX='女') as 女,
       trunc(decode((select COUNT(*) from TEST_TEACHER t1 where t1.DW_NAME=t2.DW_NAME),
                    0, 0,
                    ((select COUNT(*) from TEST_TEACHER t1 where t1.DW_NAME=t2.DW_NAME and SEX='男')
                     / (select COUNT(*) from TEST_TEACHER t1 where t1.DW_NAME=t2.DW_NAME))*100,2) as 百分比
from TEST_DEP t2;
```

输出结果如下图所示，核心方法：trunc(decode(b,0,0,a/b)*100, 2)。

DW_NAME	总数	男	女	百分比
软件学院	4	4	0	100
计算机学院	3	2	1	66.66
数学学院	3	1	2	33.33

4.问题四： 查询横向连接和纵向连接

复杂的SQL语句通常包括横向连接和纵向连接。

横向连接：使用自定义子查询

这里使用的方法是子查询自定义命名的方法，该方法只返回一行数据，通常通过Json格式给后台，后台只需要显示

即可。它的优势是不论表多复杂或表之间没有关系，而是只需要把值统计在一行，都能联系返回结果。

如下图所示，这就是横向连接返回的结果，但是需要知道具体的学院名称。

DW_NAME	ZS	BS	DW_NAME1	ZS1	BS1
软件学院	4	3	计算机学院	3	1

在统计不同表的总数信息时，使用该方法比较好。

```
select t1.DW_NAME, t2.ZS, t3.BS, t4.DW_NAME, t5.ZS, t6.BS
from
(select DW_NAME as DW_NAME from TEST_TEACHER where DW_NAME='软件学院' group by DW_NAME) t1,
(select COUNT(*) as ZS from TEST_TEACHER where DW_NAME='软件学院') t2,
(select COUNT(*) as BS from TEST_TEACHER where DW_NAME='软件学院' and DEGREE='博士') t3,
(select DW_NAME as DW_NAME from TEST_TEACHER where DW_NAME='计算机学院' group by DW_NAME) t4,
(select COUNT(*) as ZS from TEST_TEACHER where DW_NAME='计算机学院') t5,
(select COUNT(*) as BS from TEST_TEACHER where DW_NAME='计算机学院' and DEGREE='博士') t6;
```

纵向连接：使用UNION ALL连接

假设现在只需要统计"软件学院"、"计算机学院"的信息，这里需要纵向连接，则使用UNION ALL自然连接。

Union：对两个结果集进行并集操作，不包括重复行，同时进行默认规则的排序；

Union All：对两个结果集进行并集操作，包括重复行，不进行排序。

注意：COUNT需要使用GROUP BY，这种方法的优势是如果表中存在很多NULL的单位信息或只需要统计几个固定的学院信息，此时使用该方法就比较适合。

总之，特定的场合需要特定的处理方法。

```
select DW_NAME, COUNT(DW_NAME) as SUM,
       ratio_to_report(COUNT(1)) OVER() as BL,
       COUNT(CASE WHEN SEX='男' THEN 1 END) as Man, COUNT(CASE WHEN SEX='女' THEN 1 END) as Women
from TEST_TEACHER WHERE DW_NAME='软件学院' GROUP BY DW_NAME
UNION ALL
select DW_NAME, COUNT(DW_NAME) as SUM,
       ratio_to_report(COUNT(1)) OVER() as BL,
       COUNT(CASE WHEN SEX='男' THEN 1 END) as Man, COUNT(CASE WHEN SEX='女' THEN 1 END) as Women
from TEST_TEACHER WHERE DW_NAME='计算机学院' GROUP BY DW_NAME;
```

输出结果如下图所示：

DW_NAME	SUM	RATIO_TO_REPORT(COUNT(1)) OVER()	MAN	WOMEN
软件学院	4	1	4	0
计算机学院	3	1	2	1

相关资料：

[数据库] SQL语句select简单记录总结

[数据库] SQL查询语句表行列转换及一行数据转换成两列

[数据库] Navicat for Oracle基本用法图文介绍

[数据库] Navicat for Oracle设置唯一性和递增序列实验

最后希望文章对你有所帮助，这是一篇我的在线笔记，同时后面结合自己实际项目和SQL性能优化，将分享一些更为专业的文章~还有20多天就要毕业当老师了，哈哈！

(By:Eastmount 2016-06-14 深夜3点 <http://blog.csdn.net//eastmount/>)



Eastmount 博客专家

原创文章 462 获赞 6724 访问量 525万+

关注

他的留言板

