

# 数据库第八周作业

19336035 陈梓乐

- [数据库第八周作业](#)
  - [1. 解决第 8 周课程幻灯片第 15 页的 3 个问题](#)
  - [2. 解决第 8 周课程幻灯片第 56 页的 2 个问题](#)
  - [3. 阅读第 8 周课程幻灯片第 57 页关于用 1 条 SQL 解决八皇后问题的代码](#)

## 1. 解决第 8 周课程幻灯片第 15 页的 3 个问题

1. 怎样用一条SQL语句判断两个集合（假设都没有重复元素）是否相等或是子集关系？

以 emp 表和 dept 表为例，我们要找他们的 deptno 列是否为子集关系，为此只需找仅在其中一个集合出现的元素即可。

```
with tmp as (  
    select e.deptno a, d.deptno b  
    from emp e  
    full outer join dept d  
    on e.deptno = d.deptno  
)  
select * from  
    (select b onlyB from tmp where a is null)  
    full outer join  
    (select a onlyA from tmp where b is null)  
    on onlyB = onlyA;
```

TERMINAL   PROBLEMS   OUTPUT   DEBUG CONSOLE

SQL = with tmp as (  
2        select e.deptno a, d.deptno b  
3        from emp e  
4        full outer join dept d  
5        on e.deptno = d.deptno  
6    )  
7 select \* from  
8 (select b onlyB from tmp where a is null)  
9 full outer join  
10 (select a onlyA from tmp where b is null)  
11 on onlyB = onlyA;  
  
     ONLYB       ONLYA  
-----  
     40

## 2. 列出选修课程与某位指定同学完全一样的同学

我们先来查看一下数据：

```
select * from score pivot(  
    sum(score)  
    for cid in ('C1' C1, 'C2' C2, 'C3' C3, 'C4' C4, 'C5' C5)  
);
```

TERMINAL PROBLEMS OUTPUT DEBUG CONSOLE

```
SQL = select * from score pivot(  
2     sum(score)  
3     for cid in ('C1' C1, 'C2' C2, 'C3' C3, 'C4' C4, 'C5' C5)  
4 );
```

SID	C1	C2	C3	C4	C5
S1	100	100	100	100	80
S2	80	60		60	
S3		80	60	80	
S4			80		40
S5		60	80		80
S6				100	100
S7			90		90

已选择7行。

接着我们来查找与 S4 选修课程一模一样的学生：

```
select id, name from student where not exists(  
    select a.sid, a.cid, b.cid from (  
        (select sid, cid from score) a  
        full outer join  
        (select cid from score where sid = 'S4') b  
        on a.cid = b.cid  
    ) where a.sid = id and (a.cid is null or b.cid is null)  
);
```

显然我们应该选出 S4 和 S7 出来：

```
SQL = select id, name from student where not exists(  
2     select a.sid, a.cid, b.cid from (  
3         (select sid, cid from score) a  
4         full outer join  
5         (select cid from score where sid = 'S4') b  
6         on a.cid = b.cid  
7     ) where a.sid = id and (a.cid is null or b.cid is null)  
8 );
```

ID	NAME
S4	D
S7	G

### 3. 列出所有选修课程完全一样的同学名单，以学号对的形式输出结果

我们只要两两判断即可：

```
select s1.id, s2.id from student s1, student s2 where not exists(  
    select id from class where id in  
        (select cid from score where sid = s1.id)  
    and id not in  
        (select cid from score where sid = s2.id)  
    or id not in  
        (select cid from score where sid = s1.id)  
    and id in  
        (select cid from score where sid = s2.id)  
) and s1.id <> s2.id;
```

TERMINAL PROBLEMS OUTPUT DEBUG CONSOLE

```
SQL = select s1.id, s2.id from student s1, student s2 where not exists(  
2     select id from class where id in  
3         (select cid from score where sid = s1.id)  
4     and id not in  
5         (select cid from score where sid = s2.id)  
6     or id not in  
7         (select cid from score where sid = s1.id)  
8     and id in  
9         (select cid from score where sid = s2.id)  
10 ) and s1.id <> s2.id;
```

ID	ID
S4	S7
S7	S4

## 2. 解决第 8 周课程幻灯片第 56 页的 2 个问题

### 1. 假设每个人都只能和直接上司或直接下属交流，求任意两人间交流信息需要经过的最小中间节点数。

设 `layer` 表示经手的最少领导数，则关键是求出两个员工之间特有的领导的数量，记为 `unique_leader`。

```

1  with tmp as (
2      select a.ename i, b.ename j from emp a, emp b
3      where b.empno in (
4          select empno from emp
5          start with empno = a.empno
6          connect by prior mgr = empno
7      ) and a.ename <> b.ename
8  ) select a.ename ename1, b.ename ename2, (
9      case
10     when b.ename = a.ename then 0
11     when b.ename in (select j from tmp where i = a.ename) or
12         a.ename in (select j from tmp where i = b.ename) then (
13         select count(*) - 1 from emp where ename in (
14             select j from tmp where i = b.ename
15         union
16             select j from tmp where i = a.ename
17         minus (
18             select j from tmp where i = a.ename
19         intersect
20             select j from tmp where i = b.ename
21         ))
22     )
23     else (
24         select count(*) + 1 from emp where ename in (
25             select j from tmp where i = b.ename
26         union
27             select j from tmp where i = a.ename
28         minus (
29             select j from tmp where i = a.ename
30         intersect
31             select j from tmp where i = b.ename
32         ))
33     ) end
34 ) layer from emp a, emp b;

```

为表示效果，本代码不附上截图，请参照第二小问的截图。

## 2. 能否用宽表展示结果？

参照以下代码：

```

1  with tmp as (
2      select a.ename i, b.ename j from emp a, emp b
3      where b.empno in (
4          select empno from emp
5          start with empno = a.empno
6          connect by prior mgr = empno
7      ) and a.ename <> b.ename
8  ) select * from (
9      select a.ename ename1, b.ename ename2, (
10         case
11         when b.ename = a.ename then 0
12         when b.ename in (select j from tmp where i = a.ename) or
13             a.ename in (select j from tmp where i = b.ename) then (
14             select count(*) - 1 from emp where ename in (
15                 select j from tmp where i = b.ename
16             union
17                 select j from tmp where i = a.ename
18             minus (
19                 select j from tmp where i = a.ename
20             intersect
21                 select j from tmp where i = b.ename
22             ))
23         )
24         else (
25             select count(*) + 1 from emp where ename in (
26                 select j from tmp where i = b.ename
27             union
28                 select j from tmp where i = a.ename
29             minus (
30                 select j from tmp where i = a.ename
31             intersect
32                 select j from tmp where i = b.ename
33             ))
34         ) end
35     ) layer from emp a, emp b
36 ) pivot (
37     sum(layer)
38     for ename2 in (
39         'SMITH','ALLEN','WARD','JONES','MARTIN',
40         'BLAKE','CLARK','SCOTT','KING','TURNER','ADAMS',
41         'JAMES','FORD','MILLER'
42     )
43 );

```

效果如下，经检验结果正确。

ENAME1	'SMITH'	'ALLEN'	'WARD'	'JONES'	'MARTIN'	'BLAKE'	'CLARK'	'SCOTT'	'KING'	'TURNER'	'ADAMS'	'JAMES'	'FORD'	'MILLER'
ALLEN	4	0	1	2	1	0	2	3	1	1	4	1	3	3
JONES	1	2	2	0	2	1	1	0	0	2	1	2	0	2
FORD	0	3	3	0	3	2	2	1	1	3	2	3	0	3
CLARK	3	2	2	1	2	1	0	2	0	2	3	2	2	0
MILLER	4	3	3	2	3	2	0	3	1	3	4	3	3	0
SMITH	0	4	4	1	4	3	3	2	2	4	3	4	0	4
WARD	4	1	0	2	1	0	2	3	1	1	4	1	3	3
MARTIN	4	1	1	2	0	0	2	3	1	1	4	1	3	3
SCOTT	2	3	3	0	3	2	2	0	1	3	0	3	1	3
TURNER	4	1	1	2	1	0	2	3	1	0	4	1	3	3
ADAMS	3	4	4	1	4	3	3	0	2	4	0	4	2	4
BLAKE	3	0	0	1	0	0	1	2	0	0	3	0	2	2
KING	2	1	1	0	1	0	0	1	0	1	2	1	1	1
JAMES	4	1	1	2	1	0	2	3	1	1	4	0	3	3

已选择14行。

### 3. 阅读第 8 周课程幻灯片第 57 页关于用 1 条 SQL 解决八皇后问题的代码

- 1) 给代码加上适当的注释, 使我们能看明白解决问题的思路
- 2) 修改代码解决十皇后问题
- 3) (可选) 用 SQL 解决 N 皇后问题, 截图结果 (包括部分解和解的总数) 直接发给老师, N 最大者 (可并列) 可以获得总评加分 2 分奖励, 加油!

将注释放在修改后的代码中, 并附上十四皇后代码的运行结果, 十皇后只需修改对应参数。事实上, 我们只需用两位数字表示一个皇后的位置即可。

```
1 with sou as (  
2     select level n,1 k from dual connect by level<=14      /* Select a position*/  
3 ), ntt(n,k) as (  
4     select sou.n ,sou.k from sou where k=1  
5     union all  
6     select ntt.n*100+a.n, ntt.k+1                          /* Record the position*/  
7     from ntt,sou a  
8     /* If there is no conflict between the new position and the recorded position*/  
9     where not exists(  
10        select 1  
11        from (select level b1 from dual connect by level<=13) t      /* a range */  
12        where t.b1<=ntt.k and (  
13            /* If the first position is 0 ~ 9, we convert it in format 01 ~ 09 */  
14            ntt.n < power(10, 2*ntt.k-1) and (  
15                a.n=to_number(substr(0||to_char(ntt.n),b1 * 2 - 1,2)) or  
16                a.n=to_number(substr(0||to_char(ntt.n),b1 * 2 - 1,2))+(ntt.k+1-t.b1) or  
17                a.n=to_number(substr(0||to_char(ntt.n),b1 * 2 - 1,2))-(ntt.k+1-t.b1)  
18            ) or ntt.n >= power(10, 2*ntt.k-1) and (  
19                a.n=to_number(substr(to_char(ntt.n),b1 * 2 - 1,2)) or  
20                a.n=to_number(substr(to_char(ntt.n),b1 * 2 - 1,2))+(ntt.k+1-t.b1) or  
21                a.n=to_number(substr(to_char(ntt.n),b1 * 2 - 1,2))-(ntt.k+1-t.b1)  
22            )  
23        )  
24        ) and ntt.k<=13  
25 )  
26 select to_char(n) from ntt where ntt.k=14;
```

TERMINAL PROBLEMS OUTPUT DEBUG CONSOLE

```
1412100702040901031106080513  
1412100702040901031106081305  
1412100702060301110805130409  
1412100702060301110905130804  
1412100703060213010905080411  
1412100704010502091113080603  
1412100802050301071113060409  
1412100802050301091113070406  
1412100803050211010613090704
```

已选择365596行。

已用时间: 01: 37: 30.43

SQL =