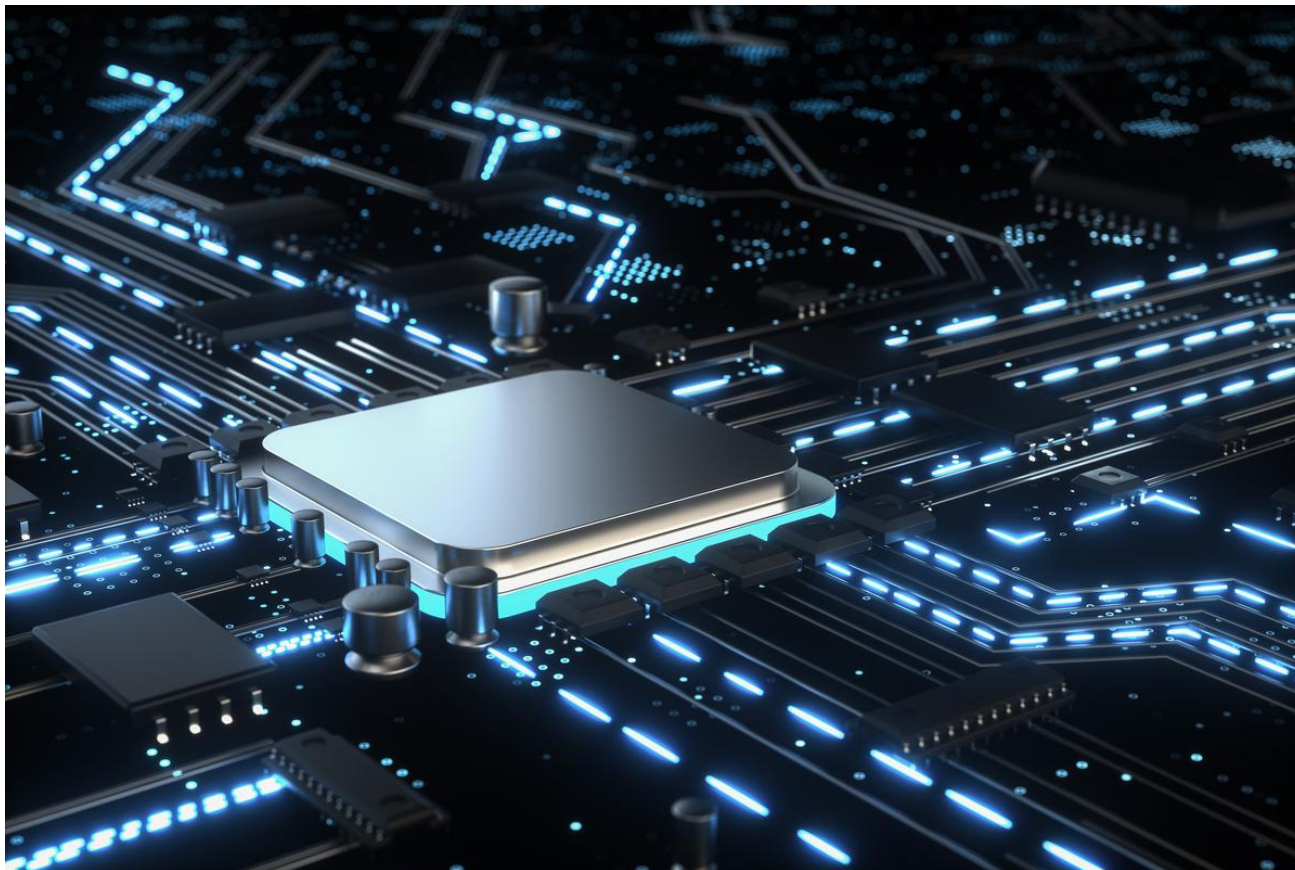




中山大學
SUN YAT-SEN UNIVERSITY



2021级 《数据库原理与应用》 第11周

2024.5.8

用（1条）SQL语句解决八皇后问题

提示（有三种办法，注意**可能需要运行很长时间**）

1 类似多重循环的暴力型SQL

2 带参数with

3 层次查询递归求解

如果有新的创造性方法总评+5

- 用PL/SQL的for, while等循环语句, 建立一张100万行的table (先创建, 里面只需要包含一个日期型列c1), 其中奇数行插入sysdate, 偶数行插入sysdate+1

```
SQL> create table test_table  
2  (record_number int,  
3  current_date date);
```

表已创建。

```
SQL> ■
```

```
declare
max_record constant int:=1000000;
i int:=1;
begin
for i in 1..max_record loop
if mod(i,2)=1 then
insert into test_table values (i,sysdate);
else
insert into test_table values (i,sysdate+1);
end if;
end loop;
end;
```

注意：PL/SQL不会自动提交

- 需要在程序最后补上commit, 或手工提交

```
SQL*Plus: Release 11.2.0.1.0 Production on 星期日 5月 8 12:48:21 2022
Copyright (c) 1982, 2010, Oracle. All rights reserved.

请输入用户名: scott
输入口令:

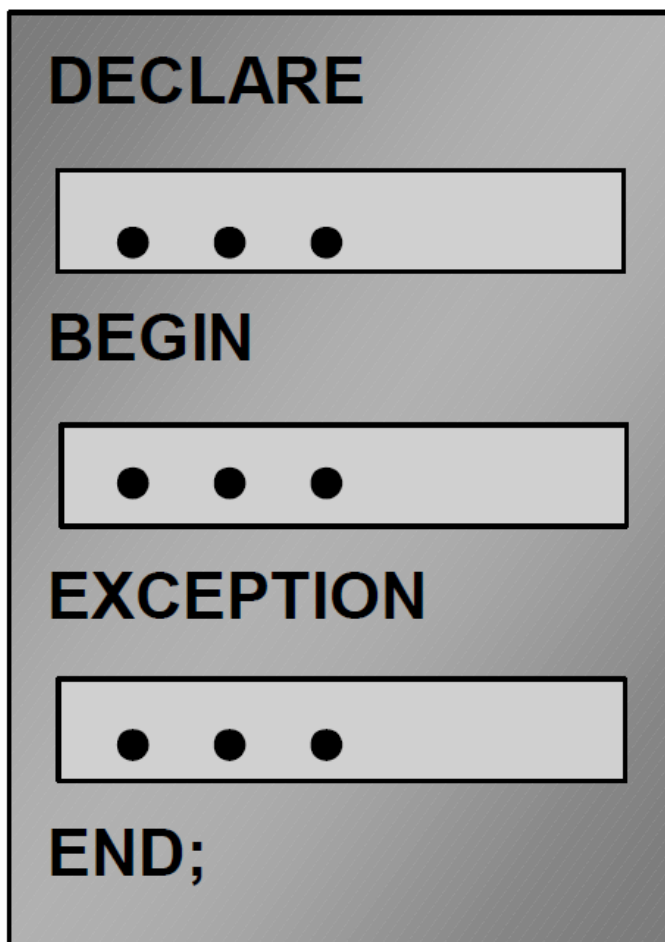
连接到:
Oracle Database 11g Enterprise Edition Release 11.2.0.1.0 - 64bit Production
With the Partitioning, OLAP, Data Mining and Real Application Testing options

SQL> select count(*) from test_table;

COUNT(*)
-----
         0

SQL>
```

知识点：PL/SQL程序基本结构





知识点：PL/SQL怎样输出

```
declare  
max_record constant int:=10000;  
i int:=1;  
a number:=0;  
begin  
for i in 1..max_record loop  
insert into test_table values (i,sysdate);  
end loop;  
commit;  
select count(*) into a from test_table;  
dbms_output.enable;  
dbms_output.put_line(a);  
end;
```



设置sqlplus环境变量serveroutput

```
SQL> declare
  2 max_record constant int:=10000;
  3 i int:=1;
  4 a number:=0;
  5 begin
  6 for i in 1..max_record loop
  7 insert into test_table values (i,sysdate);
  8 end loop;
  9 commit;
10 select count(*) into a from test_table;
11 dbms_output.enable;
12 dbms_output.put_line(a);
13 end;
14 /
```

PL/SQL 过程已成功完成。

```
SQL> set serveroutput on
SQL> declare
  2 max_record constant int:=10000;
  3 i int:=1;
  4 a number:=0;
  5 begin
  6 for i in 1..max_record loop
  7 insert into test_table values (i,sysdate);
  8 end loop;
  9 commit;
10 select count(*) into a from test_table;
11 dbms_output.enable;
12 dbms_output.put_line(a);
13 end;
14 /
1010000
1020000
```

PL/SQL 过程已成功完成。



PL/SQL的基本语法

- IF语句
- LOOP语句
- WHILE-LOOP语句
- FOR-LOOP语句
- 赋值语句
- 注释

Syntax:

```
IF condition THEN  
    statements;  
[ELSIF condition THEN  
    statements;  
[ELSE  
    statements;  
END IF;
```

Syntax:

```
LOOP                                -- delimiter
  statement1;                      -- statements
  . . .                            -- EXIT statement
  EXIT [WHEN condition];          -- EXIT statement
END LOOP;                          -- delimiter
```

condition is a Boolean variable or
expression (TRUE, FALSE,
or NULL);

Syntax:

```
WHILE condition LOOP  
    statement1;  
    statement2;  
    . . .  
END LOOP;
```

← Condition is evaluated at the beginning of each iteration.

Use the **WHILE** loop to repeat statements while a condition is **TRUE**.

Syntax:

```
FOR counter IN [REVERSE]  
    lower_bound..upper_bound LOOP  
    statement1;  
    statement2;  
    . . .  
END LOOP;
```

- Use a FOR loop to shortcut the test for the number of iterations.
- Do not declare the counter; it is declared implicitly.
- '*lower_bound* .. *upper_bound*' is required syntax.



在PL/SQL中使用SQL

- 可直接使用DML语句
- Select ... into ...
- 不能使用DDL语句

样例

- 计算在上一个样例中输出所花的时间，要求不使用sqlplus的timing，用PL/SQL来实现

```
declare  
max_record constant int:=1000000;  
i int:=1;  
begintime date;  
endtime date;  
runtime number;  
Begin  
begintime:=sysdate;  
for i in 1..max_record loop  
if mod(i,2)=1 then /*转下页*/
```




代码 (接上页)

```
insert into test_table values (i,sysdate);  
else  
insert into test_table values (i,sysdate+1);  
end if;  
end loop;  
commit;  
endtime:=sysdate;  
runtime:=(endtime-begintime)*86400;  
dbms_output.enable;  
dbms_output.put_line(runtime);  
end;
```

表被截断。

[illegible]

PL/SQL 过程已成功完成。

PL/SQL程序的几种形式

- 匿名块（与SQL语句类似，在客户端直接做adhoc查询）
- 存储过程（作为数据库对象保存在数据库中的代码，通过参数进行输入输出数据交换）
- 存储函数（作为数据库对象保存在数据库中的代码，需要有返回值作为函数的输出）

- 实现存储函数zh，能把输入的一位阿拉伯数字转成中文大写

create or replace function zh (n number)

return char as

T varchar(20):='零壹贰叁肆伍陆柒捌玖';

begin

return substr(T,n+1,1);

end;

测试结果



```
SQL> create or replace function zh (n number)
2   return char as
3   T varchar(20) := '零壹贰叁肆伍陆柒捌玖';
4   begin
5   return substr(T, n+1, 1);
6   end;
7   /
```

函数已创建。

```
SQL> select zh(8) from dual;
```

ZH(8)

捌

样例

- 实现存储函数dx, 能把阿拉伯数字的金额（最大到十万位）转化为中文大写金额

```
create or replace function dx (n number)
return varchar as
b varchar(16):='拾万仟百拾元角分';
c varchar(8);
g varchar(32);
i int;
begin
c:=replace(lpad(ltrim(to_char(n,'999999.99')),9,'0'),' ','');
for i in 1..8 loop
g:=g||zh(substr(c,i,1))||substr(b,i,1);
end loop;
return g;
end;
```

```
SQL> create or replace function dx (n number)
2  return varchar as
3  b varchar(16):='拾万仟百拾元角分';
4  c varchar(8);
5  g varchar(32);
6  i int;
7  begin
8  c:=replace(lpad(ltrim(to_char(n,'999999.99')),9,'0'),'.','');
9  for i in 1..8 loop
10 g:=g||zh(substr(c,i,1))||substr(b,i,1);
11 end loop;
12 return g;
13 end;
14 /
```

函数已创建。

```
SQL> select dx(543.90) from dual;
```

DX(543.90)

零拾零万零仟伍百肆拾叁元玖角零分



使用存储过程与存储函数的好处

- 集中修改，方便维护
- 速度更快（省去网络传输与编译时间）
- 可以受到数据库权限机制保护



怎样调用存储过程和存储函数

- 在sqlplus中使用存储过程，打入 “exec <过程名>”
- 在程序中使用存储过程，直接使用<过程名>即可
- 使用存储函数与使用标准SQL函数没有区别

包 (package) 与包体 (package body)

- 类似C语言中库的概念，可以把存储函数，存储过程组装成包
- 包存放定义部分，包体存放具体代码
- 调用时在存储过程和存储函数前加 “<包名>.”
- Oracle有大量的内置包，例如之前接触过的dbms_output和dbms_rowid，用法可以参考Oracle官方文档《PL/SQL packages》（很厚），熟悉使用Oracle的内置包是学习PL/SQL的重要内容

代码：自己建立一个包

create or replace package compute

is

function jc (x in number) return number;

function c (n in number,m in number) return number;

function p (n in number,m in number) return number;

end compute;



代码 (接上页)

```
create or replace package body compute
is
function jc (x in number) return number
is
i int:=1;
pr int:=1;
begin
for i in 1..x loop
pr:=pr*i;
end loop;
return (pr);
end jc;
```



代码 (接上页)

```
function c (n in number,m in number) return number  
is  
begin  
return jc(n)/(jc(m)*jc(n-m));  
end c;  
  
function p (n in number,m in number) return number  
is  
begin  
return jc(n)/jc(n-m);  
end p;  
  
end compute;
```

```
SQL> create or replace package compute
2  is
3  function jc (x in number) return number;
4  function c (n in number,m in number) return number;
5  function p (n in number,m in number) return number;
6  end compute;
7  /
```

程序包已创建。

```
SQL> create or replace package body compute
2  is
3  function jc (x in number) return number
4  is
5  i int:=1;
6  pr int:=1;
7  begin
8  for i in 1..x loop
9  pr:=pr*i;
10 end loop;
11 return (pr);
12 end jc;
13 function c (n in number,m in number) return number
14 is
15 begin
16 return jc(n)/(jc(m)*jc(n-m));
17 end c;
18 function p (n in number,m in number) return number
19 is
20 begin
21 return jc(n)/jc(n-m);
22 end p;
23 end compute;
24 /
```

程序包体已创建。

测试结果



```
SQL> select compute.jc(10) from dual;
```

```
COMPUTE. JC(10)
```

```
-----  
3628800
```

```
SQL> select compute.p(10,5) from dual;
```

```
COMPUTE. P(10,5)
```

```
-----  
30240
```

```
SQL> select compute.c(10,3) from dual;
```

```
COMPUTE. C(10,3)
```

```
-----  
120
```

- 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？
- 以之前的ta和tb表作为实验

C	C2
A	2400
B	3400
X	100
Y	1400
Z	30
W	500
U	160
V	1300
P	800
Q	1500

C	C2
A	2400
B	3400
V	1300
Y	1400
Q	1500

- **思路1**：借助完成关系除法的思路， A 是 B 的子集，当且仅当不存在 A 中的元素，它在 B 是找不到的（可以类似的利用两个not exists构造查询）
- **思路2**：通过比较 A ， B ， $A \cap B$ ， $A \cup B$ ， $A - B$ ， $B - A$ 等的元素个数来判断，把元素逐个逐对匹配的难度简化为count。例如如果 $A \cap B$ 与 $A \cup B$ 的元素个数相同，则 $A = B$ 。如果 $A \cap B$ 与 A 的元素个数相同，则 A 是 B 的子集，等等

作业答案



```
select 'tb是ta的子集' from dual
where (select count(*) from tb)=
(select count(*) from (
(select * from ta) intersect (select * from tb)
));
```

```
SQL> select 'tb是ta的子集' from dual
2  where (select count(*) from tb)=
3  (select count(*) from (
4  (select * from ta) intersect (select * from tb)
5  ));
```

'TB是TA的子

tb是ta的子集

```
SQL> delete from ta where c1 not in (select c1 from tb);
```

已删除5行。

```
SQL> select * from ta;
```

C	C2
A	2400
B	3400
Y	1400
V	1300
Q	1500

```
select 'ta=tb' from dual
where
(
select count(*) from
(
(select * from ta) intersect (select * from tb)
)
)
=
(
select count(*) from
(
(select * from ta) union (select * from tb)
)
);
```

```
SQL> select 'ta=tb' from dual
      2  where
      3  (
      4  select count(*) from
      5  (
      6  (select * from ta) intersect (select * from tb)
      7  )
      8  )
      9  =
     10  (
     11  select count(*) from
     12  (
     13  (select * from ta) union (select * from tb)
     14  )
     15  );

' TA=T
-----
ta=tb
```



夹带知识点: `wm_concat`函数

- Oracle的内测函数
- 另一种方式的列转行
- Mysql中类似的函数是`group_concat`
- 输出是clob类型


```
SQL> select * from emp;
```

EMPNO	ENAME	JOB	MGR	HIREDATE	SAL	COMM	DEPTNO	LOC
7369	SMITH	CLERK	7902	17-12月-80	1010		20	
7499	ALLEN	SALESMAN	7698	20-2月-81	1980	300	30	
7521	WARD	SALESMAN	7698	22-2月-81	1630	500	30	
7566	JONES	MANAGER	7839	02-4月-81	3185		20	
7654	MARTIN	SALESMAN	7698	28-9月-81	1630	1400	30	
7698	BLAKE	MANAGER	7839	01-5月-81	3230		30	
7782	CLARK	MANAGER	7839	09-6月-81	2950		10	
7839	KING	PRESIDENT		17-11月-81	5500		10	
7844	TURNER	SALESMAN	7698	08-9月-81	1880	0	30	
7900	JAMES	CLERK	7698	03-12月-81	1330		30	
7902	FORD	ANALYST	7566	03-12月-81	3210		20	
7934	MILLER	CLERK	7782	23-1月-82	1800		10	

已选择12行。

```
SQL> select wm_concat(ename) from emp;
```

```
WM_CONCAT(ENAME)
```

```
-----  
SMITH, ALLEN, WARD, JONES, MARTIN, BLAKE, CLARK, KING, TURNER, JAMES, FORD, MILLER
```

- 本身不支持排序，但使用distinct关键字后变相排序（因Oracle实现distinct会先排序再去除重复行）

```
SQL> select wm_concat(distinct ename) from emp;
```

```
WM_CONCAT(DISTINCTENAME)
```

```
-----  
ALLEN, BLAKE, CLARK, FORD, JAMES, JONES, KING, MARTIN, MILLER, SMITH, TURNER, WARD
```

- 通过wm_concat函数判断集合（某列）相等

```
SQL> select 'ta=tb' from dual
      2  where
      3  (select wm_concat(distinct c1) from ta) =
      4  (select wm_concat(distinct c1) from tb);

' TA=T
-----
ta=tb
```

夹带知识点: listagg函数

- Wm_concat的升级, 出现在11gr2之后

```
SQL> select deptno, listagg(ename,',') within group (order by ename) as names  
2  from emp  
3  group by deptno;
```

DEPTNO

NAMES

10

CLARK, KING, MILLER

20

FORD, JONES, SMITH

30

ALLEN, BLAKE, JAMES, MARTIN, TURNER, WARD

用column命令规整输出



```
SQL> column names for A50
SQL> select deptno, listagg(ename,',') within group (order by ename) as names
       2  from emp
       3  group by deptno;
```

DEPTNO NAMES

```
-----
      10 CLARK, KING, MILLER
      20 FORD, JONES, SMITH
      30 ALLEN, BLAKE, JAMES, MARTIN, TURNER, WARD
```

```
SQL>
```

对整个公司进行归并



```
SQL>  
SQL> select listagg(ename,',') within group (order by  ename) as names  
2   from emp  
3   group by 1;
```

NAMES

ALLEN, BLAKE, CLARK, FORD, JAMES, JONES, KING, MARTIN, MIL
LER, SMITH, TURNER, WARD

■ 列出选修课程与某位指定同学完全一样的同学

```
SQL> select * from sc;
```

S#	C#	G
s1	c1	90
s1	c2	99
s2	c1	70
s2	c2	90
s2	c3	92
s3	c3	60

已选择6行。

```
SQL> insert into sc values('s4','c1',80);
```

已创建 1 行。

```
SQL> insert into sc values('s4','c2',70);
```

已创建 1 行。

套用除法的思路是不对的



```
with aaa as (select c# from sc where s#='s1')  
select sn from s  
where not exists ( select * from aaa where  
not exists (select * from sc  
where s#=s.s# and c#=aaa.c#)  
);
```



```
SQL> select * from s;
```

S#	SN	SD	SA
s1	X	MA	21
s2	Y	PH	20
s3	Z	CS	21
s4	W	MA	25

```
SQL> select * from sc;
```

S#	C#	G
s1	c1	90
s1	c2	99
s2	c1	70
s2	c2	90
s2	c3	92
s3	c3	60
s4	c1	80
s4	c2	70

已选择8行。

```
SQL> with aaa as (select c# from sc where s#='s1')
2  select sn from s
3  where not exists ( select * from aaa where
4  not exists (select * from sc
5  where s#=s.s# and c#=aaa.c#)
6  );
```

SN

X

Y

W

with

aaa as (select c# from sc where s#='s1'),

bbb as (select s#,count(*) xx from (select s#,aaa.c# from aaa,sc
where aaa.c#=sc.c#) group by s#),

ccc as (select s#,count(*) xx from sc group by s#)

select ccc.s# from ccc,bbb where ccc.s#=bbb.s# and

ccc.xx=bbb.xx

and ccc.s#<>'s1';

```
SQL> with aaa as (select c# from sc where s#='s1')  
      2  select s#,aaa.c# from aaa,sc where aaa.c#=sc.c#;
```

S#	C#
s1	c1
s1	c2
s2	c1
s2	c2
s4	c1
s4	c2

已选择6行。

```
SQL> select s#, count(*) from sc group by s#;
```

S#	COUNT (*)
s2	3
s4	2
s1	2
s3	1

```
SQL> with
 2  aaa as (select c# from sc where s#='s1'),
 3  bbb as (select s#,count(*) xx from (select s#,aaa.c# from aaa,sc where aaa.c#=sc.c#) group by s#),
 4  ccc as (select s#,count(*) xx from sc group by s#)
 5  select ccc.s# from ccc,bbb where ccc.s#=bbb.s# and   ccc.xx=bbb.xx
 6  and ccc.xx=(select count(*) from aaa)
 7  and ccc.s#<>'s1';
```

S#

s4

利用listagg函数实现的方法

with

aaaa as

(select s#,listagg(c#','') within group (order by c#) as ccc from sc group by s#)

select * from aaaa where ccc=(select ccc from aaaa where s#='s1') and s#<>'s1';

```
SQL> column ccc for A50
```

```
SQL> with
```

```
2 aaaa as
```

```
3 (select s#,listagg(c#','') within group (order by c#) as ccc from sc group by s#)
```

```
4 select * from aaaa where ccc=(select ccc from aaaa where s#='s1') and s#<>'s1';
```

```
S#      CCC
```

```
-----  
s4      c1, c2
```

- 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

利用listagg函数实现的方法

with

aaaa as

(select s#,listagg(c#','') within group (order by c#) as ccc from sc group by s#)

select A.s#,B.s#,A.ccc

from aaaa A,aaaa B

where A.s#>B.s# and A.ccc=B.ccc;

```
SQL>
SQL> with
2  aaaa as
3  (select s#,listagg(c#','') within group (order by c#) as ccc from sc group by s#)
4  select A.s#,B.s#,A.ccc
5  from aaaa A,aaaa B
6  where A.s#>B.s# and A.ccc=B.ccc;
```

S#	S#	CCC
s4	s1	c1, c2



中山大學
SUN YAT-SEN UNIVERSITY

Thanks

FAQ时间