

学号 WA2214014 专业 人工智能 姓名 杨跃浙
实验日期 06.13 教师签字 成绩

实验报告

【实验名称】 实验 4-嵌套查询与集合查询

【实验目的】

1. 熟悉基本的嵌套查询与集合查询的概念和作用；
2. 了解数据库管理系统 DBMS 实现嵌套查询的基本方法；
3. 掌握 SQL 语言嵌套查询与集合查询语句的语法和功能，掌握并熟练运用嵌套查询与集合查询语句实现数据库的查询应用。
4. 了解 SQL 查询结果的处理方式

【实验原理】

嵌套查询

嵌套查询概述

一个 SELECT-FROM-WHERE...语句称为一个查询块

将一个查询块嵌套在另一个查询块的 WHERE 子句或 HAVING 短语的条件中的查询称为嵌套查询

子查询的限制

不能使用 ORDER BY 子句

层层嵌套方式反映了 SQL 语言的结构化

有些嵌套查询可以用连接运算替代

根据嵌套查询的表达方式可以将子查询分为相关子查询和不相关子查询两种, 数据库管理系统对嵌套查询求解方法不同。

不相关子查询

相关子查询

不相关子查询:

子查询的查询条件不依赖于父查询

由里向外 逐层处理。即每个子查询在上一级查询处理之前求解, 子查询的结果用于建立其父查询的查找条件。

相关子查询: 子查询的查询条件依赖于父查询

首先取外层查询中表的第一个元组, 根据它与内层查询相关的属性值处理内层查询, 若 WHERE 子句返回值为真, 则取此元组放入结果表

然后再取外层表的下一个元组

重复这一过程, 直至外层表全部检查完为止

带有 IN 谓词的子查询

带有比较运算符的子查询

带有 ANY (SOME) 或 ALL 谓词的子查询

带有 EXISTS 谓词的子查询

带有 IN 谓词的子查询

[例 2] 查询由“北京新天地”供应商供应 P001 号零件的工程项目代码。

此查询要求可以分步来完成

① 确定“北京新天地”供应商的供应商号

```
SELECT  SNO
FROM    S
WHERE   SNAME= '北京新天地';
```

结果为: S003

② 查找所有 S003 供应商供应 P001 号零件的工程项目代码。

```

SELECT    JNO
          FROM      SPJ
          WHERE     SNO= 'S003' AND PNO='P001';

```

将第①步查询嵌入到第②步查询的条件中

```

SELECT JNO
      FROM SPJ
WHERE SNO IN
      (SELECT SNO
        FROM   S
        WHERE  SNAME= '北京新天地') AND PNO='P001';

```

此查询为不相关子查询。

带有比较运算符的子查询

当能确切知道内层查询返回单值时，可用比较运算符（>，<，=，>=，<=，!= 或<>）。

与 ANY 或 ALL 谓词配合使用

[例6]假设一个供应商(没有重名)只可能在一个城市,则在[例4]中 IN 可以用 = 代替 IN :

```

SELECT Sno,Sname,CITY
      FROM   S
      WHERE CITY =
             (SELECT CITY
              FROM   S
              WHERE Sname= '北京新天地');

```

此例为不相关子查询，嵌套查询内外层虽然是同一个表，但可以加别名，也可以不加别名。

带有 ANY 或 ALL 谓词的子查询

谓词语义

ANY (SOME) : 任意一个值

ALL: 所有值

需要配合使用比较运算符

> ANY 大于子查询结果中的某个值

> ALL 大于子查询结果中的所有值

< ANY 小于子查询结果中的某个值

< ALL 小于子查询结果中的所有值

>= ANY 大于等于子查询结果中的某个值

>= ALL 大于等于子查询结果中的所有值

<= ANY 小于等于子查询结果中的某个值

<= ALL 小于等于子查询结果中的所有值

= ANY 等于子查询结果中的某个值

=ALL 等于子查询结果中的所有值 (通常没有实际意义)

!= (或<>) ANY 不等于子查询结果中的某个值

!= (或<>) ALL 不等于子查询结果中的任何一个值

ANY (或 SOME) , ALL 谓词与聚集函数、IN 谓词的等价转换关系

	=	<>或!=	<	<=	>	>=
ANY	IN	--	<MAX	<=MAX	>MIN	>= MIN
ALL	--	NOT IN	<MIN	<= MIN	>MAX	>= MAX

带有 EXISTS 谓词的子查询

1. EXISTS 谓词

存在量词

带有 EXISTS 谓词的子查询不返回任何数据，只产生逻辑真值“true”或逻辑假值“false”。

若内层查询结果非空，则外层的 WHERE 子句返回真值

若内层查询结果为空，则外层的 WHERE 子句返回假值

由 EXISTS 引出的子查询，其目标列表达式通常都用*，因为带 EXISTS 的子查询只返回真值或假值，给出列名无实际意义

2. NOT EXISTS 谓词

若内层查询结果非空，则外层的 WHERE 子句返回假值

若内层查询结果为空，则外层的 WHERE 子句返回真值

集合查询

SELECT 语句的查询结果是一个集合，两个或多个具有相同类型结果集的

SELECT 语句可以进行集合的并、交、差等运算。

并操作 UNION

交操作 INTERSECT

差操作 EXCEPT (MINUS)

参加集合操作的各查询结果的列数必须相同；对应项的数据类型也必须相同

1. 并操作 UNION

UNION 运算符可以将两个或两个以上 SELECT 语句的查询结果合并成一个结果集，称为并查询。格式如下：

```
(SELECT * FROM TABLE A) UNION[ALL] (SELECT * FROM TABLE B)
```

UNION：将多个查询结果合并起来时，系统自动去掉重复元组。

UNION ALL：将多个查询结果合并起来时，保留重复元组，因为不需要消除重复元组的步骤，所以查询速度更快。

2. 交操作 INTERSECT

INTERSECT 运算符用于返回两个或两个以上 SELECT 语句的查询结果集合的交集，称为交查询。格式如下：

```
(SELECT * FROM TABLE A) INTERSECT(SELECT * FROM TABLE B)
```

查询结果的处理

在 DBMS 中，SELECT 语句的查询结果可以在交互式界面中直接显示给用户，前面所举的例子都是这种方式，也可以使用子查询的方式将第一次查询的查询结果直接作为查询的数据源表。

使用子查询的方式将第一次查询的查询结果直接作为查询的数据源表。

还可以使用 INTO 短语将查询结果集永久保存到命名的表中。

【实验内容】

针对供应链管理数据库 SPJ，进行各种嵌套与集合查询：

- (1)查询使用了“S001”供应商供应的零件的工程项目信息；
- (2)查询供应零件“螺丝刀”的供应商信息；
- (3)查询供应商“北京新天地”的所有客户（工程项目）的信息；
- (4)查询供应了“J003”工程“P001”零件的供应商信息；
- (5)查询使用了“天津”供应商供应的“红色”零件的工程项目信息；
- (6)查询没有使用“天津”供应商供应的“红色”零件的工程项目信息；
- (7)查询使用了“S001”供应的“螺母”零件的工程信息；
- (8)查询使用了“螺母”零件的工程信息及相应供应商信息；
- (9)查询给“长春一汽”项目供应零件的供应商信息和供应情况；
- (10)查询给“北京三建”项目供应“螺丝刀”零件最多的供应商信息和供应情况；
- (11)查询使用了“螺母”或“螺丝刀”零件的工程项目信息；
- (12)查询既使用了“螺母”又使用了“螺丝刀”零件的工程项目信息；
- (13)查询使用了“螺母”零件但没有使用“螺丝刀”零件的工程项目信息；
- (14) 查询使用过所有供应商供应的零件的工程项目信息。

1.

The screenshot shows the Data Studio interface with a SQL query in the editor and its results in the Results pane.

SQL Query:

```

1= SELECT *
2= FROM J014
3= WHERE JNO IN (
4=   SELECT JNO
5=   FROM SPJ014
6=   WHERE SNO = 'S001'
7= );
8

```

Results:

jno	jname	city
J001	北京三建	北京
J002	长春一汽	长春
J003	新安弹簧厂	天津
J004	临江造船厂	天津

运行时间: 816 ms
查询提交时间: 2024-06-13 15:03:43.936 GMT+ 获取4行, 共179 ms
可以编辑查询结果。

查询使用了“S001” 供应商供应的零件的工程项目信息；

2.

The screenshot shows the Data Studio interface with a SQL query in the editor and its results in the Results pane.

SQL Query:

```

1= SELECT *
2= FROM S014
3= WHERE SNO IN (
4=   SELECT SNO
5=   FROM SPJ014
6=   WHERE PNO IN (
7=     SELECT PNO
8=     FROM P014
9=     WHERE PNAME = '螺丝刀'
10=   )
11= );
12

```

Results:

sno	sname	stat	city
S002	北京启明星	A	北京
S003	北京新天地	C	北京
S005	上海普丰	C	上海

运行时间: 233 ms
查询提交时间: 2024-06-13 15:04:29.571 GMT+ 获取3行, 共169 ms
可以编辑查询结果。

查询供应零件“螺丝刀” 的供应商信息；

3.

The screenshot shows the Data Studio interface with a SQL query executed. The query is as follows:

```

1 SELECT *
2 FROM J014
3 WHERE JNO IN (
4     SELECT JNO
5     FROM SPJ014
6     WHERE SNO IN (
7         SELECT SNO
8         FROM S014
9         WHERE SNAME = '北京新天地'
10    )
11 );
12

```

The execution time is 188 ms. The results table is shown below:

jno	jname	city
1001	北京三建	北京

Query execution time: 2024-06-13 15:05:04.672 GMT+ 获取1行, 共138 ms. 可以编辑查询结果.

自动保存完成: Thu Jun 13 15:03:09 GMT+08:00 2024

查询供应商“北京新天地”的所有客户（工程项目）的信息；

4.

The screenshot shows the Data Studio interface with a SQL query executed. The query is as follows:

```

1 SELECT *
2 FROM S014
3 WHERE SNO IN (
4     SELECT SNO
5     FROM SPJ014
6     WHERE PNO = 'P001' AND JNO = 'J003'
7 );
8

```

The execution time is 179 ms. The results table is shown below:

sno	sname	stat	city
S001	天津安贝儿	B	天津

Query execution time: 2024-06-13 15:05:31.605 GMT+ 获取1行, 共123 ms. 可以编辑查询结果.

自动保存完成: Thu Jun 13 15:03:09 GMT+08:00 2024

查询供应了“J003”工程“P001”零件的供应商信息；

5.

The screenshot shows the Data Studio interface with a SQL query executed. The query is as follows:

```
1= SELECT *
2= FROM J014
3= WHERE JNO IN (
4=   SELECT JNO
5=   FROM SPJ014
6=   WHERE SNO IN (
7=     SELECT SNO
8=     FROM S014
9=     WHERE CITY = '天津'
10=   ) AND PNO IN (
11=     SELECT PNO
12=     FROM P014
13=     WHERE COLOR = '红'
14=   )
15= );
16=
```

The results table shows 3 rows:

jno	jname	city
J001	北京三建	北京
J003	新安染整厂	天津
J004	临江造船厂	天津

运行时间: 226 ms
查询提交时间: 2024-06-13 15:06:48.595 GMT+ 获取3行, 共163 ms
可以编辑查询结果。

查询使用了“天津”供应商供应的“红色”零件的工程项目信息;

6.

The screenshot shows the Data Studio interface with a SQL query executed. The query is as follows:

```
1= SELECT *
2= FROM J014
3= WHERE JNO NOT IN (
4=   SELECT JNO
5=   FROM SPJ014
6=   WHERE SNO IN (
7=     SELECT SNO
8=     FROM S014
9=     WHERE CITY = '天津'
10=   ) AND PNO IN (
11=     SELECT PNO
12=     FROM P014
13=     WHERE COLOR = '红'
14=   )
15= );
16=
```

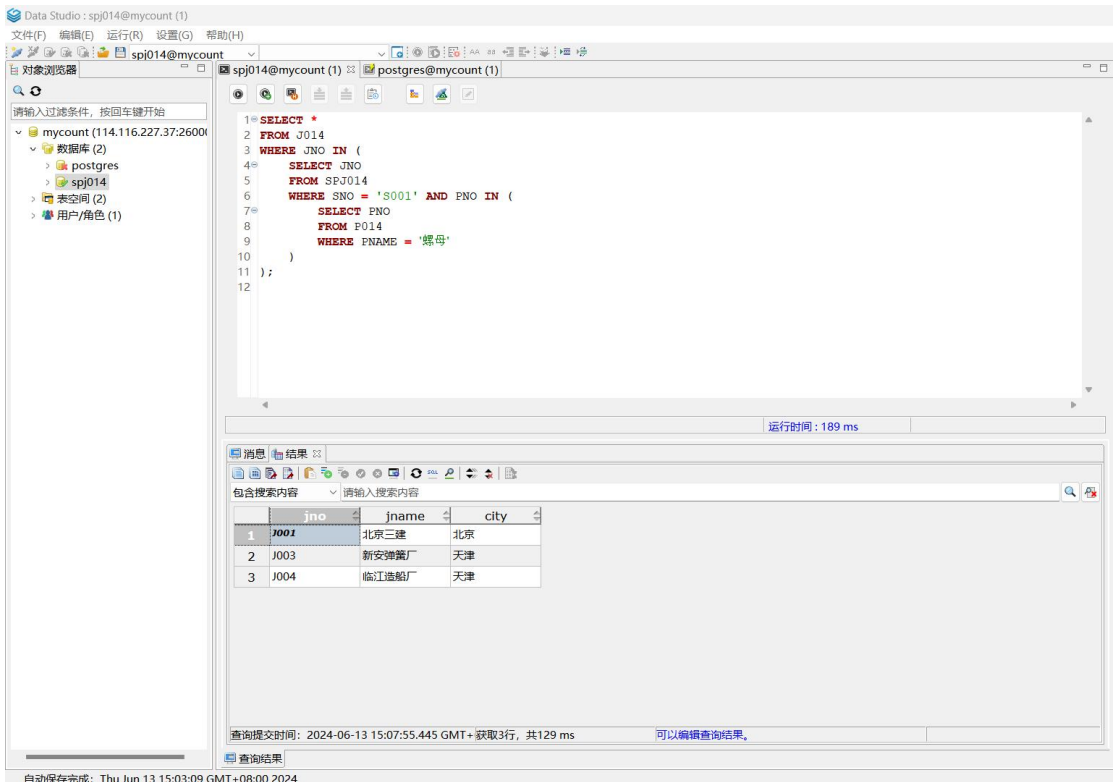
The results table shows 4 rows:

jno	jname	city
J002	长春一汽	长春
J005	唐山机车厂	唐山
J006	新新无线电厂	常州
J007	裕泰半导体厂	南京

运行时间: 196 ms
查询提交时间: 2024-06-13 15:07:28.335 GMT+ 获取4行, 共130 ms
可以编辑查询结果。

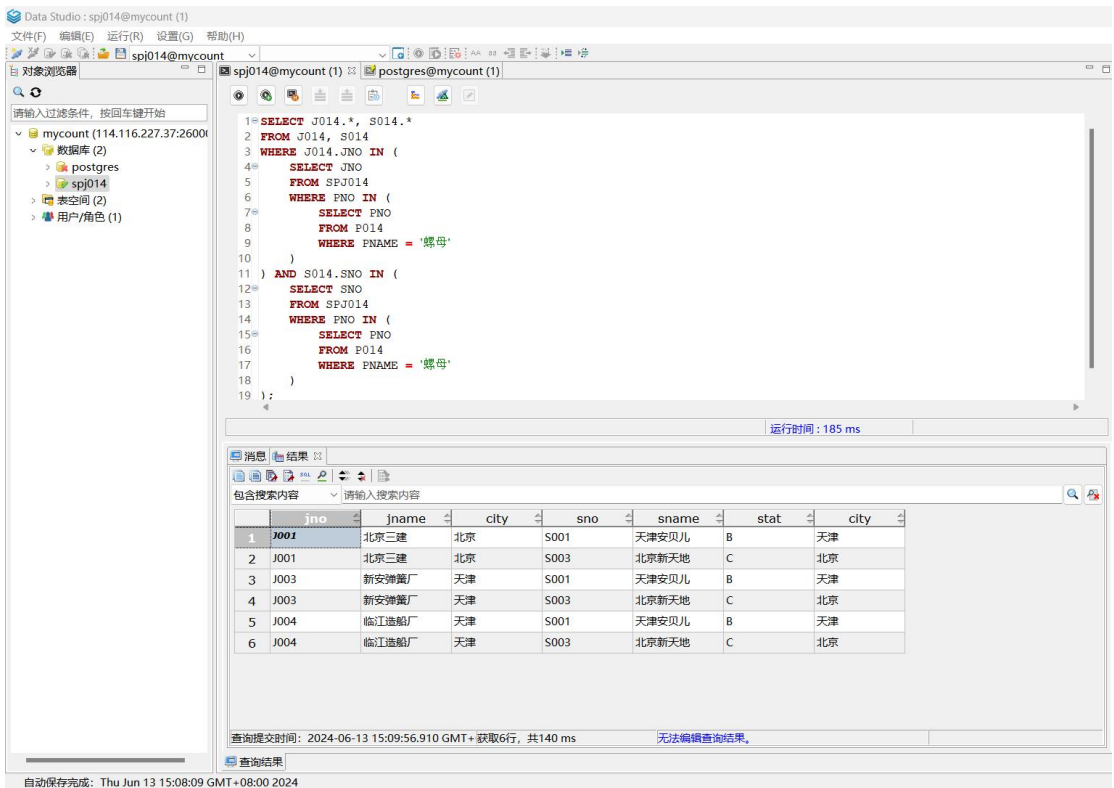
查询没有使用“天津”供应商供应的“红色”零件的工程项目信息;

7.



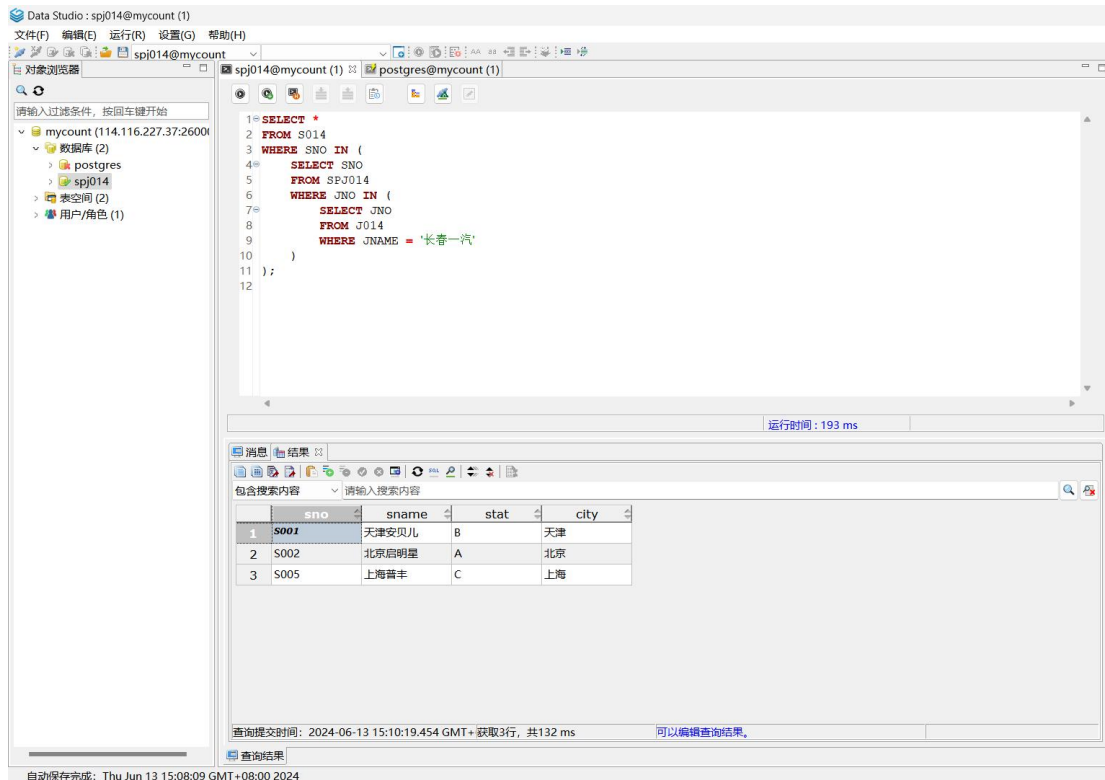
查询使用了“S001”供应的“螺母”零件的工程信息；

8.



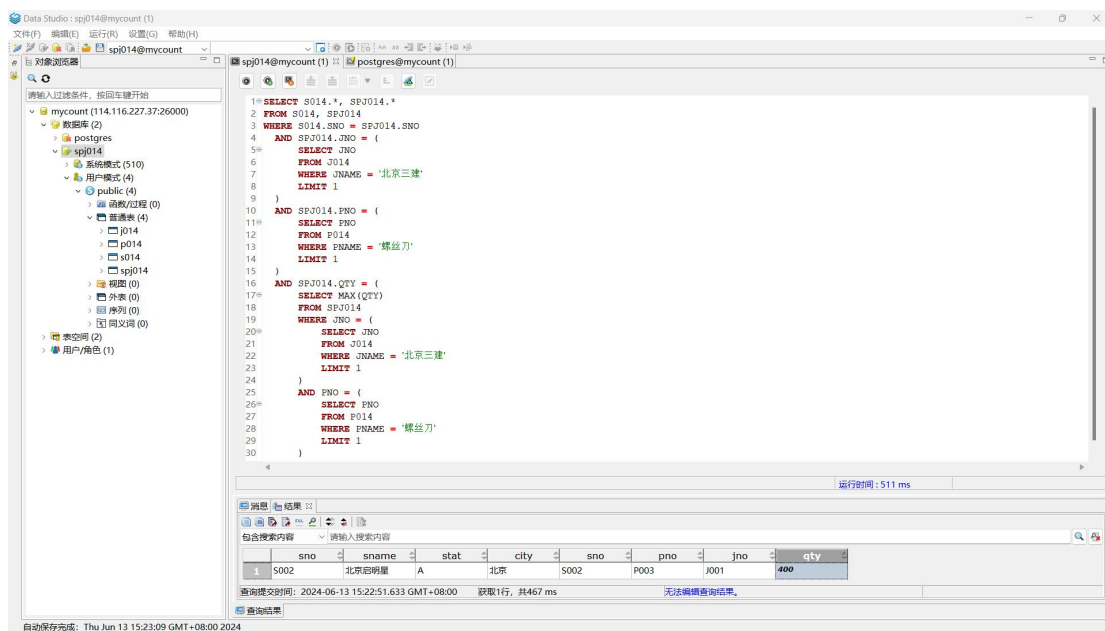
查询使用了“螺母”零件的工程信息及相应供应商信息；

9.



查询给“长春一汽”项目供应零件的供应商信息和供应情况;

10.



查询给“北京三建”项目供应“螺丝刀”零件最多的供应商信息和供应情况;

11.

SQL Query:

```

1 SELECT *
2 FROM J014
3 WHERE JNO IN (
4     SELECT JNO
5     FROM SPJ014
6     WHERE PNO IN (
7         SELECT PNO
8         FROM P014
9         WHERE PNAME IN ('螺母', '螺丝刀')
10    )
11 );
12

```

运行时间: 179 ms

jno	jname	city
J001	北京二建	北京
J002	长春一汽	长春
J003	新安铸钢厂	天津
J004	临江造船厂	天津
J005	唐山机车厂	唐山

查询提交时间: 2024-06-13 15:14:26.914 GMT+ 获取5行, 共118 ms [可以编辑查询结果。](#)

自动保存完成: Thu Jun 13 15:13:09 GMT+08:00 2024

查询使用了“螺母”或“螺丝刀”零件的工程项目信息;

12.

SQL Query:

```

1 SELECT *
2 FROM J014
3 WHERE JNO IN (
4     SELECT JNO
5     FROM SPJ014
6     WHERE PNO IN (
7         SELECT PNO
8         FROM P014
9         WHERE PNAME = '螺母'
10    )
11 ) AND JNO IN (
12     SELECT JNO
13     FROM SPJ014
14     WHERE PNO IN (
15         SELECT PNO
16         FROM P014
17         WHERE PNAME = '螺丝刀'
18    )
19 );
20

```

运行时间: 171 ms

jno	jname	city
J001	北京二建	北京
J004	临江造船厂	天津

查询提交时间: 2024-06-13 15:14:49.167 GMT+ 获取2行, 共128 ms [可以编辑查询结果。](#)

自动保存完成: Thu Jun 13 15:13:09 GMT+08:00 2024

查询既使用了“螺母”又使用了“螺丝刀”零件的工程项目信息;

13.

The screenshot shows the Data Studio interface with a SQL query editor and a results pane. The query is as follows:

```

2 FROM J014
3 WHERE JNO IN (
4   SELECT JNO
5   FROM SPJ014
6   WHERE PNO IN (
7     SELECT PNO
8     FROM P014
9     WHERE PNAME = '螺母'
10  )
11 ) AND JNO NOT IN (
12   SELECT JNO
13   FROM SPJ014
14   WHERE PNO IN (
15     SELECT PNO
16     FROM P014
17     WHERE PNAME = '螺丝刀'
18   )
19 );
20

```

The results pane shows a table with the following data:

jno	jname	city
1	1003	新安钢铁厂 天津

运行时间: 173 ms
查询提交时间: 2024-06-13 15:15:10.560 GMT+ 获取1行, 共146 ms
自动保存完成: Thu Jun 13 15:13:09 GMT+08:00 2024

查询使用了“螺母”零件但没有使用“螺丝刀”零件的工程项目信息;

14.

The screenshot shows the Data Studio interface with a SQL query editor and a results pane. The query is as follows:

```

1= SELECT *
2 FROM J014
3 WHERE JNO IN (
4   SELECT JNO
5   FROM SPJ014
6   GROUP BY JNO
7   HAVING COUNT(DISTINCT SNO) = (
8     SELECT COUNT(DISTINCT SNO)
9     FROM S014
10  )
11 );
12

```

The results pane shows a table with the following data:

jno	jname	city
-----	-------	------

运行时间: 523 ms
查询提交时间: 2024-06-13 15:20:14.420 GMT+ 获取0行, 共486 ms
自动保存完成: Thu Jun 13 15:18:09 GMT+08:00 2024

查询使用过所有供应商供应的零件的工程项目信息。

【小结或讨论】

在本次实验中，我通过对供应链管理数据库 SPJ 的各种嵌套查询和集合查询的实际操作，深入了解了嵌套查询与集合查询的基本概念、语法和功能。这些操作帮助我熟悉了 SQL 语言中嵌套查询与集合查询的使用，掌握了相关语句的编写，并在实验中成功实现了各种复杂的数据库查询应用。

首先，我掌握了嵌套查询的基本概念，理解了查询块的层层嵌套结构，并区分了不相关子查询和相关子查询的不同应用场景。不相关子查询不依赖于父查询，可以独立求解，而相关子查询则依赖于父查询中的条件，需要逐层处理。通过实际编写查询语句，我逐步理解了如何在不同的场景下应用这些查询，以提高查询效率和简化查询过程。

在具体的实验操作中，我针对多种查询需求编写了相应的 SQL 语句。例如，查询使用了特定供应商供应的零件的工程项目信息，查询供应特定零件的供应商信息，查询特定供应商的所有客户信息，以及查询供应特定工程和零件的供应商信息等。这些查询涵盖了 IN 谓词、比较运算符、ANY/ALL 谓词以及 EXISTS 谓词的使用，使我更加全面地掌握了各种嵌套查询技术。

在实验过程中，特别是查询给“北京三建”项目供应“螺丝刀”零件最多的供应商信息和供应情况时，我遇到了子查询返回多行的问题。通过分析和调整查询逻辑，最终解决了这个问题，确保子查询只返回唯一结果。这让我深刻认识到在编写嵌套查询时，子查询必须返回单一结果的必要性，并学会了通过 LIMIT 关键字来限制结果行数，避免类似错误的发生。

此外，我还学习了集合查询的操作方法，通过 UNION、INTERSECT 等运算符实现了查询结果的并集和交集操作，掌握了不同集合操作的应用场景和语法规则。在这些操作中，我意识到参加集合操作的查询结果集必须具有相同的列数和数据类型，这也是在实际操作中需要特别注意的地方。

通过此次实验，我不仅加深了对 SQL 语言嵌套查询与集合查询的理解，还提升了实际编写和调试复杂查询语句的能力。在未来的工作中，这些技能将帮助我更高效地处理数据库查询任务，提高数据分析和处理的效率。在本次实验中遇到的问题和解决方法也为我今后处理类似问题提供了宝贵的经验和参考。总的来说，这次实验使我在理论和实践两个方面都得到了显著提升。