Database-System Experiment-5

21307289 刘森元

1. 实验目的

熟悉SQL语句的数据查询语言，能够使用SQL语句对数据库进行嵌套查询。

2. 实验环境

Macbook Pro 2021 (Apple M1 Pro)

MacOS Ventura 13.5.2

PostgreSQL 15.4 (Homebrew)

Z-shell 5.9

3. 实验内容

- 通过实验验证对子查询的两个限制条件。
- 体会相关子查询和不相关自查询的不同。
- 考察4类谓词的用法，包括：
  - 第1类，IN，NOT IN；
  - 第2类，带有比较运算符的子查询；
  - 第3类，SOME，ANY或ALL谓词的子查询；
  - 第4类，带有EXISTS谓词的子查询。

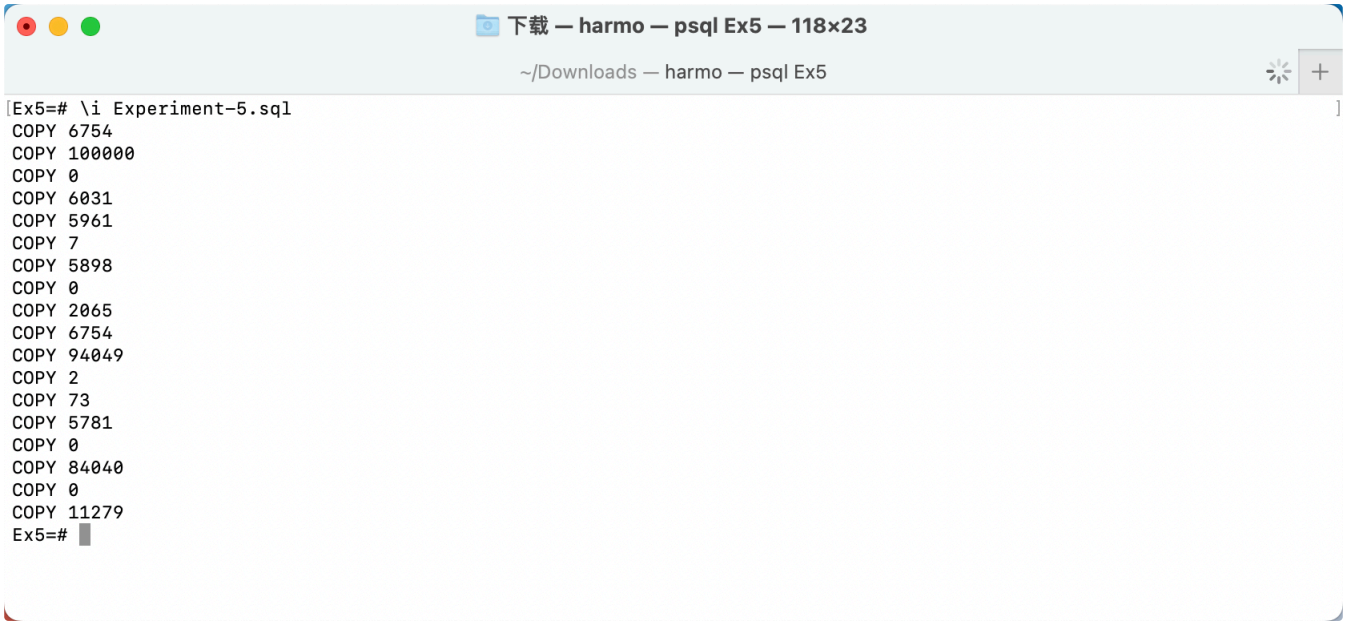
4. 实验步骤

首先使用 `\i` 命令执行 sql 脚本，将数据导入到数据库中

```
Experiment-5=# \i STUDENTS.sql
Experiment-5=# \i TEACHERS.sql
Experiment-5=# \i COURSES.sql
Experiment-5=# \i CHOICES.sql
```

以下完整代码见 Experiment-5.sql，输出结果均使用 `COPY` 导出至 `.csv` 文件

使用 `\i Experiment-5.sql` 命令执行脚本，有结果



Task-1

```sql
-- 查询学号850955252的学生同年级的所有学生资料
COPY(
    SELECT *
    FROM STUDENTS
    WHERE grade = (SELECT grade FROM STUDENTS WHERE sid = '850955252')
)
TO '/Users/qiu_nangong/Documents/Github/Database-System/Experiment-5/Task-1.csv'
WITH (FORMAT 'csv', HEADER TRUE, DELIMITER ',');
```

Task-2

```sql
-- 查询所有的有选课的学生的详细信息
COPY(
    SELECT *
    FROM STUDENTS
    WHERE sid IN (SELECT sid FROM CHOICES)
)
TO '/Users/qiu_nangong/Documents/Github/Database-System/Experiment-5/Task-2.csv'
WITH (FORMAT 'csv', HEADER TRUE, DELIMITER ',');
```

Task-3

```sql
-- 查询没有学生选的课程的编号
COPY(
    SELECT cid
    FROM COURSES
    WHERE cid NOT IN (SELECT DISTINCT cid FROM CHOICES)
)
TO '/Users/qiu_nangong/Documents/Github/Database-System/Experiment-5/Task-3.csv'
WITH (FORMAT 'csv', HEADER TRUE, DELIMITER ',');
```

Task-4

```sql
-- 查询选修了课程名为C++的学生学号和姓名
COPY(
    SELECT STUDENTS.sid, STUDENTS.sname
    FROM STUDENTS
    JOIN CHOICES ON STUDENTS.sid = CHOICES.sid
    JOIN COURSES ON CHOICES.cid = COURSES.cid
    WHERE COURSES.cname = 'c++'
)
TO '/Users/qiu_nangong/Documents/Github/Database-System/Experiment-5/Task-4.csv'
WITH (FORMAT 'csv', HEADER TRUE, DELIMITER ',');
```

Task-5

```sql
-- 找出选修课程成绩最差的选课记录
COPY(
    SELECT *
    FROM CHOICES
    WHERE (cid, score) IN (
        SELECT cid, MIN(score)
        FROM CHOICES
        GROUP BY cid
    )
    ORDER BY CHOICES.cid, CHOICES.no
)
TO '/Users/qiu_nangong/Documents/Github/Database-System/Experiment-5/Task-5.csv'
WITH (FORMAT 'csv', HEADER TRUE, DELIMITER ',');
```

Task-6

```
-- 找出和课程UML或课程C++的课时一样的课程名称
COPY(
    SELECT cname
    FROM COURSES
    WHERE hour = (SELECT hour FROM COURSES WHERE cname = 'uml')
    OR hour = (SELECT hour FROM COURSES WHERE cname = 'c++')
)
TO '/Users/qiu_nangong/Documents/Github/Database-System/Experiment-5/Task-6.csv'
WITH (FORMAT 'csv', HEADER TRUE, DELIMITER ',');
```

Task-7

```
-- 查询所有选修编号10001的课程的学生的姓名
COPY(
    SELECT STUDENTS.sname
    FROM STUDENTS
    JOIN CHOICES ON STUDENTS.sid = CHOICES.sid
    WHERE CHOICES.cid = '10001'
)
TO '/Users/qiu_nangong/Documents/Github/Database-System/Experiment-5/Task-7.csv'
WITH (FORMAT 'csv', HEADER TRUE, DELIMITER ',');
```

Task-8

```
-- 查询选修了所有课程的学生姓名
COPY(
    SELECT sname
    FROM STUDENTS
    WHERE STUDENTS.sid IN (
        SELECT CHOICES.sid
        FROM CHOICES
        GROUP BY CHOICES.sid
        HAVING COUNT(DISTINCT CHOICES.cid) = (SELECT COUNT(DISTINCT COURSES.cid)
FROM COURSES)
    )
)
TO '/Users/qiu_nangong/Documents/Github/Database-System/Experiment-5/Task-8.csv'
WITH (FORMAT 'csv', HEADER TRUE, DELIMITER ',');
```

Prac-1

```
-- 查询选修C++课程的成绩比姓名为znkoo的学生高的所有学生的编号和姓名
COPY (
    SELECT STUDENTS.sid, STUDENTS.sname
    FROM STUDENTS
```

```
    JOIN CHOICES ON STUDENTS.sid = CHOICES.sid
    JOIN COURSES ON CHOICES.cid = COURSES.cid
    WHERE COURSES.cname = 'c++' AND CHOICES.score > (
        SELECT score
        FROM CHOICES
        JOIN STUDENTS ON CHOICES.sid = STUDENTS.sid
        WHERE STUDENTS.sname = 'znkoo' AND CHOICES.cid = (SELECT cid FROM COURSES
WHERE cname = 'c++')
    )
)
TO '/Users/qiu_nangong/Documents/Github/Database-System/Experiment-5/Prac-1.csv'
WITH (FORMAT 'csv', HEADER TRUE, DELIMITER ',');
```

Prac-2

```
-- 找出和学生883794999或学生850955252的年级一样的学生的姓名
COPY (
    SELECT sname
    FROM STUDENTS
    WHERE grade = (SELECT grade FROM STUDENTS WHERE sid = '88379499')
    OR grade = (SELECT grade FROM STUDENTS WHERE sid = '850955252')
)
TO '/Users/qiu_nangong/Documents/Github/Database-System/Experiment-5/Prac-2.csv'
WITH (FORMAT 'csv', HEADER TRUE, DELIMITER ',');
```

Prac-3

```
-- 查询没有选修Java的学生名称
COPY (
    SELECT sname
    FROM STUDENTS
    WHERE sid NOT IN (
        SELECT DISTINCT sid
        FROM CHOICES
        JOIN COURSES ON CHOICES.cid = COURSES.cid
        WHERE COURSES.cname = 'java'
    )
)
TO '/Users/qiu_nangong/Documents/Github/Database-System/Experiment-5/Prac-3.csv'
WITH (FORMAT 'csv', HEADER TRUE, DELIMITER ',');
```

Prac-4

```
-- 找出课时最少的课程的详细信息
COPY (
    SELECT *
    FROM COURSES
    WHERE hour = (SELECT MIN(hour) FROM COURSES)
)
TO '/Users/qiu_nangong/Documents/Github/Database-System/Experiment-5/Prac-4.csv'
WITH (FORMAT 'csv', HEADER TRUE, DELIMITER ',');
```

Prac-5

```
-- 查询工资最高的教师的编号和开设的课程号
COPY (
    SELECT CHOICES.tid, CHOICES.cid
    FROM CHOICES
    JOIN TEACHERS ON TEACHERS.tid = CHOICES.tid
    WHERE TEACHERS.tid IN (
        SELECT tid
        FROM TEACHERS
        WHERE salary = (SELECT MAX(salary) FROM TEACHERS)
    )
    GROUP BY CHOICES.tid, CHOICES.cid
)
TO '/Users/qiu_nangong/Documents/Github/Database-System/Experiment-5/Prac-5.csv'
WITH (FORMAT 'csv', HEADER TRUE, DELIMITER ',');
```

Prac-6

```
-- 找出选修课程ERP成绩最高的学生编号
COPY (
    SELECT DISTINCT sid
    FROM CHOICES
    WHERE score = (
        SELECT MAX(score)
        FROM CHOICES
        JOIN COURSES ON CHOICES.cid = COURSES.cid
        WHERE COURSES.cname = 'erp'
    )
)
TO '/Users/qiu_nangong/Documents/Github/Database-System/Experiment-5/Prac-6.csv'
WITH (FORMAT 'csv', HEADER TRUE, DELIMITER ',');
```

Prac-7

```
-- 查询没有学生选修的课程的名称
COPY (
    SELECT cname
    FROM COURSES
    WHERE cid NOT IN (
        SELECT DISTINCT cid
        FROM CHOICES
    )
)
TO '/Users/qiu_nangong/Documents/Github/Database-System/Experiment-5/Prac-7.csv'
WITH (FORMAT 'csv', HEADER TRUE, DELIMITER ',');
```

Prac-8

```
-- 找出讲授课程UML的教师讲授的所有课程名称
COPY (
    SELECT DISTINCT tid, cname
    FROM COURSES
    JOIN CHOICES ON COURSES.cid = CHOICES.cid
    WHERE CHOICES.tid IN (
        SELECT tid
        FROM CHOICES
        WHERE cid = (SELECT cid FROM COURSES WHERE cname = 'uml')
    )
    ORDER BY tid, cname
)
TO '/Users/qiu_nangong/Documents/Github/Database-System/Experiment-5/Prac-8.csv'
WITH (FORMAT 'csv', HEADER TRUE, DELIMITER ',');
```

Prac-9

```
-- 查询选修了编号200102901的教师开设的所有课程的学生编号
COPY (
    WITH X AS (
        SELECT DISTINCT cid
        FROM CHOICES
        WHERE tid = '200102901'
    )
    SELECT sid
    FROM CHOICES
    WHERE sid IN (
        SELECT sid
        FROM CHOICES
        WHERE cid IN (SELECT cid FROM X)
```

```
        GROUP BY sid
        HAVING COUNT(DISTINCT cid) = (SELECT COUNT(*) FROM X)
    )
)
TO '/Users/qiu_nangong/Documents/Github/Database-System/Experiment-5/Prac-9.csv'
WITH (FORMAT 'csv', HEADER TRUE, DELIMITER ',');
```

Prac-10

```
-- 查询选修课程Database的学生集合与选修课程UML的学生集合的并集
COPY (
    SELECT DISTINCT sid
    FROM CHOICES
    JOIN COURSES ON CHOICES.cid = COURSES.cid
    WHERE cname = 'database'
    OR cname = 'uml'
    ORDER BY sid
)
TO '/Users/qiu_nangong/Documents/Github/Database-System/Experiment-5/Prac-10.csv'
WITH (FORMAT 'csv', HEADER TRUE, DELIMITER ',');
```

5. 实验心得

通过本次实验，我熟悉了SQL语句中的嵌套查询的使用方法，并了解了子查询的限制条件和不同类型的谓词。在实践过程中，我学会了通过子查询来解决复杂的查询需求，如查询同年级的学生、查询选课信息、找出成绩最差的选课记录等。同时，我也学会了使用各种谓词来进行子查询，如IN、EXISTS、SOME/ANY/ALL等。

在编写查询语句时，我需要仔细考虑子查询的位置、子查询的返回结果和主查询之间的关系，确保查询逻辑的正确性。此外，我还注意到了在处理多个条件时，可以使用逻辑运算符（如AND、OR）来组合条件，以满足更复杂的查询需求。

通过实践，我对数据库查询的灵活性和强大性有了更深入的了解。嵌套查询为我们提供了一种强大的工具，可以在单个查询中完成多个条件的判断和数据的提取，提高了查询的效率和灵活性。

总的来说，本次实验使我对SQL语句中的嵌套查询有了更深入的认识和理解，并通过实践提升了我的查询能力和解决问题的能力。