上海大学

SHANGHAI UNIVERSITY

数据库原理（一）作业一

组　　号	小组 A
学号姓名	20121034 胡才郁
日　　期	2022 年 12 月 21 日

第 2 题

对于教学数据库的三个基本表。试用 SQL 的查询语句表达下列查询。

```
S(SNO,SNAME,AGE,SEX,SDEPT)
SC(SNO,CNO,GRADE)
C(CNO,CNAME,CDEPT,TNAME)
```

1. 检索 **LIU** 老师所授课程的课程号和课程名

```sql
SELECT CNO, CNAME FROM C
WHERE TNAME = 'LIU';
```

2. 检索年龄大于 **23** 岁的男学生的学号和姓名

```sql
SELECT SNO, SNAME FROM S
WHERE AGE > 23 AND SEX = '男';
```

3. 检索学号为 **S3** 的学生所学课程的课程名和任课教师名

```sql
SELECT CNAME, TNAME FROM SC, C
WHERE SC.CNO = C.CNO AND CNO = 'S3';
```

4. 检索至少选修 **LIU** 老师所授课程中一门课程的女学生姓名

```sql
SELECT SNAME FROM S,SC,C
WHERE S.SNO = SC.SNO AND SC.CNO = C.CNO AND SEX = '女' AND TNAME = 'LIU';
```

5. 检索 **WANG** 同学不学的课程的课程号

```sql
SELECT CNO FROM C
WHERE EXISTS
    (SELECT * FROM S
        WHERE SNAME = 'WANG' AND NOT EXISTS
            (SELECT * FROM SC
             WHERE SC.SNO = S.SNO AND SC.CNO = C.CNO)
```

6. 检索至少选修两门课程的学生学号

```sql
SELECT SNO FROM SC
GROUP BY SNO
HAVING COUNT(*) >= 2;
```

7. 检索全部学生都选修的课程的课程的课程号和课程名

```
SELECT DISTINCT CNO, CNAME FROM C
WHERE NOT EXISTS
    (SELECT * FROM S
     WHERE NOT EXISTS
            (SELECT * FROM SC
             WHERE SC.SNO = S.SNO AND SC.CNO = C.CNO)
```

8. 检索选修课程包含 **LIU** 老师所授课程的学生学号

```
SELECT DISTINCT SNO FROM S, C, SC
WHERE S.SNO = SC.SNO AND SC.CNO = C.CNO AND TNAME = 'LIU';
```

第 6 题

试用 SQL 查询语句表达下列对教学数据库中三个基本表 S、SC、C 的查询。

1. 统计有学生选修的课程门数

```
SELECT COUNT(DISTINCT CNO) FROM SC;
```

2. 求选修 C4 课程的学生的平均年龄

```
SELECT AVG(AGE) AS AVG_AGE FROM S, SC
WHERE S.SNO = SC.SNO AND SC.CNO = 'C4';
```

3. 求 LIU 老师所授课程的每门课程的学生平均成绩

```
SELECT AVG(GRADE) AS AVG_GRADE FROM SC, C
WHERE SC.CNO = C.CNO AND TNAME = 'LIU'
GROUP BY SC.CNO;
```

4. 统计每门课程的学生选修人数（超过 10 人的课程才统计）。要求输出课程号和选修人数，查询结果按人数降序排列，若人数相同，按课程号升序排列

```
SELECT CNO, COUNT(SNO) AS COUNT_SNO FROM SC
GROUP BY CNO HAVING COUNT(*) > 10
ORDER BY COUNT_SNO DESC, CNO ASC;
```

5. 检索学号比 WANG 同学大，而年龄比他小的学生姓名

```
SELECT SNAME FROM S
WHERE SNO > (SELECT SNO FROM S WHERE SNAME = 'WANG') AND
      AGE < (SELECT AGE FROM S WHERE SNAME = 'WANG');
```

6. 检索姓名以 WANG 打头的所有学生的姓名和年龄

```
SELECT SNAME, AGE FROM S
WHERE SNAME LIKE 'WANG%';
```

7. 在 SC 中检索成绩为空值的学生学号和课程号

```
SELECT SNO, CNO FROM SC
WHERE GRADE IS NULL;
```

8. 求年龄大于女同学平均年龄的男学生姓名和年龄

```
SELECT SNAME, AGE FROM S
WHERE SEX = '男' AND
      AGE > (SELECT AVG(AGE) FROM S
                WHERE SEX = '女');
```

9. 求年龄大于所有女同学年龄的男学生姓名和年龄

```
SELECT SNAME, AGE FROM S
WHERE SEX = '男' AND
      AGE > (SELECT MAX(AGE) FROM S
                WHERE SEX = '女');
```

第 7 题

试用 SQL 更新语句表达对教学数据库中三个基本表 S、SC、C 的各个更新操作。

1. 向基本表 S 中插入一个学生元组 **('S9','WU',18)**

```
INSERT INTO S(SNO, SNAME, AGE) VALUES('S9','WU',18);
```

2. 在基本表 S 中检索每一门课程都大于或等于 **80** 分的学生学号、姓名和性别，并将检索结果保存到另一个已存在的基本表 **STUDENT(SNO,SNAME,SEX)** 中

```
INSERT INTO S(SNO, SNAME, AGE)
SELECT DISTINCT S.SNO, SNAME, SEX FROM S, SC
WHERE S.SNO = SC.SNO
GROUP BY S.SNO, SNAME, SEX
HAVING MIN(GRADE) >= 80;
```

3. 在基本表 **SC** 中删除尚无成绩的选课元组

```
DELETE FROM SC WHERE GRADE IS NULL;
```

4. 把 **WANG** 同学的学习选课和成绩全部删去

```
DELETE FROM SC
WHERE SNO IN (SELECT SNO FROM S
              WHERE SNAME = 'WANG');
```

5. 把选修 **MATHS** 课不及格的成绩全部改为空值

```
UPDATE SC SET GRADE = NULL
WHERE GRADE < 60 AND
      CNO IN (SELECT CNO FROM C
              WHERE CNAME = 'MATHS')
```

6. 把低于总平均成绩的女同学成绩提高 **5%**

```
UPDATE SC
SET GRADE = GRADE * 1.05
WHERE SNO IN (SELECT SNO FROM S WHERE SEX = '女') AND
      GRADE < (SELECT AVG(GRADE) FROM SC);
```

7. 在基本表 SC 中修改 C4 课程的成绩，若成绩小于或等于 75 分时提高 5%，若成绩大于 75 分时提高 4%（用两个 UPDATE 语句实现）

```
UPDATE SC
SET GRADE = GRADE * 1.05
WHERE CNO = 'C4' AND GRADE <= 75;
UPDATE SC
SET GRADE = GRADE * 1.04
WHERE CNO = 'C4' AND GRADE > 75;
```

第 9 题

对于教学数据库中基本表 **SC**，已建立下列视图：

```
CREATE VIEW S_GRADE(SNO,C_NUM,AVG_GRADE)
  AS SELECT SNO,COUNT(C_NUM),AVG(GRADE)
    FROM SC
    GROUP BY SNO,C_NUM;
```

试判断下列查询和更新是否允许执行。若允许，写出转换到基本表 **SC** 上的相应操作。

```
SELECT * FROM S_GRADE;
```

允许查询，转换为：

```
SELECT SNO, COUNT(CNO) AS C_NUM, AVG(GRADE) AS AVG_GRADE
FROM SC
GROUP BY SNO;
```

```
SELECT SNO, C_NUM
FROM S_GRADE
WHERE AVG_GRADE>80;
```

允许查询，转换为：

```
SELECT SNO, COUNT(CNO) AS C_NUM
FROM SC
GROUP BY SNO
HAVING AVG(GRADE)>80;
```

```
SELECT SNO, AVG_GRADE
FROM S_GRADE
WHERE C_NUM>(SELECT C_NUM
             FROM S_GRADE
             SNO='S4');
```

允许查询，转换为：

```
SELECT SNO, AVG(CNO) AS AVG_GRADE
FROM SC
GROUP BY SNO
HAVING COUNT(CNO) > (SELECT COUNT(*)
                     FROM SC
                     WHERE SNO='S4');
```

```sql
UPDATE S_GRADE
SET C_NUM = C_NUM + 1
WHERE SNO = 'S4';
```

不允许更新。要在视图 **S_GRADE** 中更改门数之前需要先修改 **SC** 表。许多 **DBMS** 设置试图为只读的，因为这样做不会破坏视图的完整性

```sql
DELETE FROM S_GRADE
WHERE C_NUM > 4;
```

不允许删除。要在视图 **S_GRADE** 中删除数据需要先修改 **SC** 表。许多 **DBMS** 设置试图为只读的，因为这样做不会破坏视图的完整性