

## 数据库 SQL 实验

081152039 赵建荣

### 实验平台

Arch Linux-3.0, mysql 5.5.16.

### 实验内容

```
#####
#
#           Section 1
#
#####

##### 1.1 CREATE #####
CREATE TABLE Student (
    SNO INT PRIMARY KEY,
    SNAME CHAR(9) NOT NULL,
    SEX CHAR(2),
    DEPTNO INT
);

CREATE TABLE Course (
    CNO INT,
    CNAME CHAR(20) NOT NULL,
    TNO INT,
    CREDIT INT,
    PRIMARY KEY(CNO, TNO)
);

CREATE TABLE SC (
    SNO INT,
    CNO INT,
    GRADE INT,
    PRIMARY KEY(SNO, CNO)
);

CREATE TABLE Teacher (
    TNO INT PRIMARY KEY,
    TNAME CHAR(8) NOT NULL,
    DEPTNO INT
);

CREATE TABLE Dept (
    DEPTNO INT PRIMARY KEY,
    DNAME CHAR(20) NOT NULL
);
```

```

##### 1.2 MODIFY #####
ALTER TABLE Student ADD AGE INT;
ALTER TABLE Student MODIFY AGE SMALLINT;

##### 1.3 DELETE #####
DROP TABLE Student;
DROP TABLE Course;
DROP TABLE SC;
DROP TABLE Teacher;
DROP TABLE Dept;

#####
#
#                               Section 2
#
#####

##### 2.1 INSERT #####
INSERT INTO Student VALUES
(1001, '张天', 'm', 10, 20),
(1002, '李兰', 'f', 10, 21),
(1003, '陈铭', 'm', 10, 21),
(1004, '刘茜', 'f', 20, 21),
(1005, '马阳', 'm', 20, 22);

INSERT INTO Course VALUES
(1, '数据结构', 101, 4),
(2, '数据库', 102, 4),
(3, '离散数学', 103, 4),
(4, 'C 语言程序设计', 101, 2);

INSERT INTO SC VALUES
(1001, 1, 80), (1001, 2, 85), (1001, 3, 78),
(1002, 1, 78), (1002, 2, 82), (1002, 3, 86),
(1003, 1, 92), (1003, 3, 90),
(1004, 1, 87), (1004, 4, 90),
(1005, 1, 85), (1005, 4, 92);

INSERT INTO Teacher VALUES
(101, '张星', 10), (102, '李珊', 10),
(103, '赵应', 10), (104, '刘田', 20);

INSERT INTO Dept VALUES
(10, '计算机'), (20, '信息');

##### 2.2 SINGLE TABLE QUERY #####
SELECT * FROM Student;
SELECT SNAME FROM Student WHERE SEX = 'f';
SELECT * FROM SC WHERE GRADE BETWEEN 80 AND 89 ORDER BY GRADE DESC;

```

```
SELECT DEPTNO, COUNT(*) FROM Student GROUP BY DEPTNO;
```

```
##### 2.3 LINK QUERY #####
```

```
SELECT SNAME, AGE
FROM Student, Dept
WHERE SEX = 'f'
AND AGE <= 21
AND Student.DEPTNO = Dept.DEPTNO
AND Dept.DNAME = '信息';
```

```
##### 2.4 EMBEDDED QUERY #####
```

```
## 1 ##
```

```
SELECT SNAME
FROM Student
WHERE SNO IN (
    SELECT SNO
    FROM SC, Course
    WHERE SC.CNO = Course.CNO
    GROUP BY SNO
    HAVING SUM(Course.CREDIT) < 10
);
```

```
## 2 ##
```

```
#####
SELECT CNAME, SNAME, temp.GRADE
FROM Student,
    Course,
    (SELECT SNO, CNO, GRADE
     FROM SC
     WHERE (CNO, GRADE) IN (
         SELECT CNO, MAX(GRADE)
         FROM SC
         GROUP BY CNO
     )
    ) AS temp
WHERE Student.SNO = temp.SNO
AND Course.CNO = temp.CNO;
```

```
## 3 ##
```

```
#####
SELECT DISTINCT SNO
FROM SC
WHERE NOT EXISTS (
    SELECT *
    FROM (
        SELECT CNO
        FROM SC
        WHERE SNO = 1001
    ) AS a
    WHERE NOT EXISTS (
        SELECT *
        FROM SC as b
```

```

        WHERE b.SNO = SC.SNO
        AND b.CNO = a.CNO
    )
);

## 4 ##
#####
SELECT SNAME
FROM Student
WHERE NOT EXISTS (
    SELECT *
    FROM (
        SELECT CNO
        FROM Course, Teacher
        WHERE Course.TNO = Teacher.TNO
        AND Teacher.TNAME = '张星'
    ) AS c
    WHERE NOT EXISTS (
        SELECT *
        FROM SC
        WHERE SNO = Student.SNO
        AND CNO = c.CNO
    )
);

##### 2.5 MODIFY #####
UPDATE SC
SET GRADE = GRADE + 2
WHERE CNO IN (
    SELECT CNO
    FROM Course, Teacher
    WHERE Course.CNAME = '数据结构'
    AND Course.TNO = Teacher.TNO
    AND Teacher.TNAME = '张星');

##### 2.6 DELETE #####
DELETE FROM SC
WHERE SNO IN (
    SELECT SNO
    FROM Student
    WHERE SNAME = '马阳');

#####
#
#                               Section 3
#
#####

##### 3.1 CREATE #####
CREATE VIEW CS_STUDENT
AS SELECT *

```

```
FROM Student
WHERE DEPTNO IN (
    SELECT DEPTNO
    FROM Dept
    WHERE DNAME = '计算机'
);
```

```
##### 3.2 DELETE #####
DROP VIEW CS_STUDENT;
```