

(一) 数据定义

1. 建立基本表

[1] 创建学生表 Student ,由以下属性组成 :学号 SNO(INT 型 ,主键),姓名 SNAME(CHAR 型 , 长度为 8 , 非空), 性别 SEX (CHAR 型 , 长度为 2), 所在系 DEPTNO (INT 型)。

命令 :

```
CREATE TABLE Student (  
  
    SNO INT NOT NULL PRIMARY KEY,  
  
    SNAME CHAR(8) NOT NULL,  
  
    SEX CHAR(2),  
  
    DEPTNO INT  
  
);
```

[2] 创建课程表 Course ,由以下属性组成 :课程号 CNO(INT 型),课程名 CNAME(CHAR 型 , 长度为 20 , 非空),授课教师编号 TNO(INT 型),学分 CREDIT(INT 型)。其中(CNO , TNO) 为主键。

命令 :

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

[3] 创建学生选课表 SC ,由以下属性组成 :学号 SNO ,课程号 CNO ,成绩 GRADE。所有属性均为 INT 型 ,其中 (SNO , CNO) 为主键。

命令 :

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

[4] 创建教师表 Teacher ,由以下属性组成 教师编号 TNO(INT 型 ,主键) 教师姓名 TNAME (CHAR 型 , 长度为 8 , 非空), 所在系 DEPTNO (INT 型)。

命令 :

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

[5] 创建系表 Dept ,由以下属性组成 :系号 DEPTNO(INT 型 ,主键),系名 DNAME(CHAR 型 , 长度为 20 , 非空)。

命令 :

```
CREATE TABLE Dept (  
  
    DEPTNO INT PRIMARY KEY,
```

DNAME CHAR(20) NOT NULL

);

2. 修改基本表

[1] 在 Student 表中加入属性 AGE (INT 型)。

命令：

ALTER TABLE Student ADD AGE INT;

[2] 将 Student 表中的属性 AGE 类型改为 SMALLINT 型。

命令：

ALTER TABLE Student MODIFY COLUMN AGE SMALLINT;

3. 删除基本表

[1] 在所有操作结束后删除 Student 表。

[2] 在所有操作结束后删除 Course 表。

[3] 在所有操作结束后删除 SC 表。

[4] 在所有操作结束后删除 Teacher 表。

[5] 在所有操作结束后删除 Dept 表。

命令：

DROP TABLE Student;

DROP TABLE Course;

DROP TABLE SC;

DROP TABLE Teacher;

DROP TABLE Dept;

(二) 数据操作

1. 插入数据

[1] 向 Student 表插入下列数据：

(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)

命令：

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);**

[2] 向 Course 表插入下列数据：

(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)

命令：

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);**

[3] 向 SC 表插入下列数据：

(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)。

命令：

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);
```

[4] 向 Teacher 表插入下列数据：

(101 , 张星 , 10) (102 , 李珊 , 10)
(103 , 赵应 , 10) (104 , 刘田 , 20)。

命令：

INSERT INTO Teacher

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

[5] 向 Dept 表插入下列数据：

(10 , 计算机) (20 , 信息)。

命令：

INSERT INTO Dept

```
VALUES (10,'计算机'),(20,'信息');
```

2. 单表查询

[1] 查询所有学生的信息。

命令：

```
SELECT * FROM Student;
```

[2] 查询所有女生的姓名。

命令：

```
SELECT SNAME  
  
FROM Student  
  
WHERE SEX = 'f';
```

[3] 查询成绩在 80 到 89 之间的所有学生的选课记录，查询结果按成绩的降序排列。

命令：

```
SELECT *  
  
FROM SC  
  
WHERE GRADE BETWEEN 80 AND 89  
  
ORDER BY GRADE DESC;
```

[4] 查询各个系的学生人数。

命令：

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

3. 连接查询

查询信息系年龄在 21 岁以下（含 21 岁）的女生姓名及其年龄。

命令：

```
SELECT SNAME,AGE
```

from Student S,Dept D

WHERE S.AGE <= 21 AND S.SEX = 'f' AND S.DEPTNO = D.DEPTNO AND

D.DNAME = '信息';

4. 嵌套查询

[1] 查询修课总学分在 10 分以下的学生姓名。

命令：

SELECT SNAME

FROM Student

WHERE SNO IN (SELECT SNO

FROM Course C,SC

WHERE C.CNO = SC.CNO

GROUP BY SNO

HAVING SUM(C.CREDIT) < 10);

[2] 查询各门课程取得最高成绩的学生姓名及其成绩。

命令：

SELECT SNAME,GRADE

FROM (SELECT SNO,GRADE

FROM SC

WHERE GRADE >= ALL (SELECT GRADE

FROM SC C

WHERE SC.CNO = C.CNO))

D ,Student

WHERE D.SNO = Student.SNO;

[3] 查询选修了 1001 学生选课的全部课程的学生学号。

逻辑：查出选修了 1001 号学生选修课的同学，并且满足这些选修课的门数同张星同样多，
则们组逻辑。

命令：

SELECT DISTINCT SC1.SNO

FROM SC SC1

WHERE SC1.CNO = ANY (SELECT CNO

FROM SC

WHERE SNO = 1001)

GROUP BY SNO

HAVING COUNT(CNO) = (SELECT COUNT(CNO) FROM SC WHERE SNO =

1001);

[4] 查询选修了张星老师开设的全部课程的学生姓名。

逻辑：首先查到所有选修了张老师开设的课程的同学，并统计他们选修的门数，如果选修张
星老师的课程门数等于张星老师教授的总门数，则满足逻辑。

命令：

SELECT SNAME

FROM Student

WHERE SNO IN (SELECT SNO

FROM SC

WHERE CNO = ANY(SELECT CNO


```

FROM Course

WHERE TNO IN (SELECT TNO

FROM Teacher

WHERE TNAME = '张星

'))

GROUP BY SNO

HAVING COUNT(CNO) = (SELECT COUNT(CNO)

FROM Course

WHERE TNO IN (SELECT TNO

FROM Teacher

WHERE TNAME = '

张星')));

```

5. 修改数据

将张星老师数据结构课的学生成绩全部加 2 分。

命令：

```

UPDATE SC

SET GRADE = GRADE + 2

WHERE CNO IN ( SELECT C.CNO

FROM Course C,Teacher T

WHERE C.CNAME = '数据结构' AND C.TNO = T.TNO

AND T.TNAME = '张星');

```

6. 删除数据

删除马阳同学的所有选课记录。

命令：

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

(三) 视图操作

1. 建立视图

在插入数据的 Student 基本表上为计算机科学与技术系的学生记录建立一个视图 CS_STUDENT。

命令：

```
CREATE VIEW CS_STUDENT  
  
AS SELECT *  
  
FROM Student WHERE DEPTNO IN (SELECT DEPTNO  
  
FROM Dept  
  
WHERE DNAME = '计算机');
```

2. 删除视图

在操作结束后删除视图 CS_STUDENT。

命令：

```
DROP VIEW CS_STUDENT;
```