

```

1.select name, birthday from student;
2.select * from student where name like '王%';
3.select * from score where degree between 60 and 80;
4.select t.name, c.cname from teacher t
course where c.tno=t.no and t.sex= '男'
5.select name from student where sex=(select sex from
student where name= '李军' ) and class=(select class from
student where name = '李军' )
6.insert into student values( '999' , '程功' , '
男' ,10/01/80,' 95035' );
7.update student set class=' 95031' where no= '999' ;
8.delete from student where no= '999' ;

```

设有下列关系模式

STUDENT (NO, NAME, SEX, BIRTHDAY, CLASS)

TEACHER (NO, NAME, SEX, BIRTHDAY, PROF, DEPART)

COURSE (CNO, CNAME, TNO)

SCORE (NO, CNO, DEGR)

其中, 学生信息表 STUDENT 由学号 (NO)、学生姓名 (NAME)、性别 (SEX)、出生年月日 (BIRTHDAY)、班级 (CLASS) 组成, 记录学生的情况: 教师信息表 TEACHER 由教师号 (NO)、教师姓名 (NAME)、性别 (SEX)、出生年月日 (BIRTHDAY)、职称 (PROF)、教师所在系 (DEPART) 组成, 记录教师的情况: 课程表 COURSE 由课程号 (CNO)、课程名称 (CNAME)、任课教师号 (TNO) 组成, 记录所开课程及任课教师情况; 成绩表 SCORE 由学生学号 (NO)、课程号 (CNO)、成绩 (DEGREE) 组成, 记录学生选课情况及相应的成绩。用 SQL 语句实现以下操作

- (1) 显示 STUDENT 表中每个学生的姓名和出生日期。
- (2) 显示 STUDENT 表中所有姓“王”的学生记录。
- (3) 显示成绩表中成绩在 60 分到 80 分之间的所有记录
- (4) 显示“男”教师及其所上的课程。(显示该教师的姓名和所上的课程名)
- (5) 选出和“李军”同学同性别并同班的学生姓名。
- (6) 向 STUDENT 表中插入一条学生记录。(学号:999, 姓名:程功, 性别:男, 出生年月日:10/01/80, 班级 95035)
- (7) 在 STUDENT 表中将学号为“999”的学生的班号改为“95031”
- (8) 在 STUDENT 表中删除学号为“999”的学生记录。

设数据库中有 3 个关系

职工表 EMP (E#, ENAME, AGE, SEX, ECITY), 其属性分别表示职工工号、姓名、年龄、性别和籍贯

工作表 WORKS (E#, C#, SALARY), 其属性分别表示职工工号、工作的公司编号和工资

公司表 COMP (C#, CNAME, CITY), 其属性分别表示公司编号、公司名称和公司所在城市。

试用 SQL 语言写出下列操作

- 1) 检索超过 50 岁的男职工的工号和姓名
- 2) 假设每个职工只能在一个公司工作, 检索工资超过 1000 元的男性职工工号、姓名和工资。
- 3) 假设每个职工可在多个公司工作, 检索在编号为 C4 和 C8 公司兼职的职工工号和姓名。
- 4) 检索在“联华公司”工作、工资超过 1000 元的男性职工的工号和姓名。
- 5) 假设每个职工可在多个公司工作, 检索每个职工的兼职公司数目和工资总数, 显示 (E#, NUM, SUM SALARY), 分别表示工号、公司数目和工资总数。
- 6) 检索年龄和“小雨”一样的职工的工号和姓名。
- 7) 检索和工号为 E6 的职工工资一样的职工姓名和工号。
- 8) 检索联华公司中低于本公司平均工资的职工工号和姓名。
- 9) 在每个公司中为 50 岁以上职工加薪 100 元 (若职工为多个公司工作, 可重复加)。