

设有一个教务管理数据库，其关系模式如下：

- 学生 Student ( sno, sname, dept )
  - 教师 Teacher ( tno, tname, dept)
  - 课程 Course ( cno, cname, dept, opt )      // 课程类型 opt 分为 ‘核心’ 和 ‘其他’
  - 选课 Register(sno, cno, tno, grade, regyear)      // regyear 为选修年份
- // 一个学生一门课只能有一条选课记录；成绩采用百分制，60 分为及格线。

请分别用关系代数和 SQL 语言表示下述查询（第 3、第 7 和第 8 小题仅使用 SQL）

说明：

- 1) 在关系代数中，仅使用表名的首字母表示各关系表
- 2) 分上下午两个部分，上午仅做 1~4 题，下午仅做 5~8 题，多做不限
- 3) 正确答案不唯一，不必要求与参考答案完全一致

---

### 上午部分

---

1. 查询 ‘软院’ 课程的外系选修情况，结果返回外系选修学生的学号、系别及所选的 ‘软院’ 课程号

代数：

```
((C where dept=' 软院' )[cno] join L) join
(S where dept<>' 软院' ))[S.sno, S.dept, C.cno]
```

SQL:

```
Select S.sno, S.dept, L.cno
From Student S, Register R
Where S.sno=R.sno and S.dept<>' 软院' and
      L.cno in (select cno from Course where dept=' 软院' );
```

2. 查询下述学生的学号和姓名：选修过 ‘数学’ 系的所有 ‘核心’ 课程，且这些核心课程的成绩都大于或等于 80

代数：

```
数学系的核心课程 T1 := (C where dept='数学' and opt='核心')[cno]
选修过这些课且成绩都大于或等于 80: T2 := (R where grade>=80)[sno, cno] ÷ T1
满足条件的学生: (T2 join S) [sno, sname]
```

SQL:

```
Select S.sno, S.sname From Student S
Where not exists (
  Select * from Course C
  Where C.dept=' 数学' and C.opt=' 核心' and not exists (
    Select * from Register R
    Where R.sno=S.sno and R.cno=C.cno and grade>=80));
```

其中：最里面的那个子查询，也可以表示如下：

```
..... And C.cno not in
      (Select R.cno from Register R where R.sno=S.sno and grade>=80)
```

3. 查询下述课程的编号及其在 2016 年度 ‘选修人数, 最高分, 最低分’: 在 2016 年度的平均成绩低于 75 分

```
Select cno, count(*), max(grade), min(grade)
From Register
Where regyear=2016
Group by cno
Having avg(grade)<75;
```

4. 查询下述学生的学号和系别: 选修过所在院系的所有 ‘核心’ 课程

关系代数:

Step1: 查询每个学生与其所在院系的核心课程所构成的关系 M1(sno, cno)

```
M1 := ((S join C) where opt=' 核心' ) [sno, cno] // 学生和课程通过 dept 实现自然连接
```

Step2: 查询至少有一门本系的核心课程没有选修的学生 M2[sno]

```
M2 := (M1 - R[sno, cno]) [sno]
```

Step 3: 查出选修过所在院系的所有核心课程的同学及其所在院系

```
((R[sno] - M2) join S) [sno, dept]
```

SQL:

```
Select S.sno, S.dept    From Student S
Where not exists (
    Select * from Course C
    Where C.dept=S.dept and C.opt=' 核心' and not exists (
        Select * from Register R
        Where R.sno=S.sno and R.cno=C.cno));
```

5. 查询教师担任外系授课任务的情况，结果返回教师的编号及系别、所讲授的外系课程的课程号及系别。

关系代数：

$((T \times R \times C) \text{ where } T.tno=R.tno \text{ and } C.cno=R.cno \text{ and } T.dept \neq C.dept) [T.tno, T.dept, C.cno, C.dept]$

SQL:

```
Select T.tno, T.dept, C.cno, C.dept
From Teacher T, Course C, Register R
Where T.tno=R.tno and C.cno=R.cno and T.dept<>C.dept;
```

6. 查询下述学生的学号和姓名：在 2016 年选修了‘孔明’老师所讲授的所有课程且成绩都及格

关系代数：

‘孔明’老师所讲授的课程为： $K := ((R \text{ join } T) \text{ where } T.tname='孔明')[cno]$   
满足条件的学生为： $M := (R \text{ where } regyear=2016 \text{ and } grade \geq 60)[sno, cno] \div K$   
最终的结果关系为： $(M \text{ join } S)[sno, sname]$

SQL:

```
Select S.sno, S.sname From Student S
Where not exists (
    Select * from Teacher T, Register R1
    Where T.tno=R1.tno and T.tname='孔明' and not exists (
        Select * from Register R2
        Where R2.sno=S.sno and R2.cno=R1.cno and R2.regyear=2016));
```

7. 查询下述学生的学号和姓名：选修的外系课程门数低于 5（包括没有选修过外系课程的学生）

参考答案 1:

```
(Select S.sno, S.sname
From Register R, Student S, Course C
Where R.sno=S.sno and R.cno=C.cno and S.dept<>C.dept
Group by S.sno, S.sname
Having count(*) < 5 )
union
(select S.sno, S.sname from Student S
where not exists ( Select * from Register R, Course C
                    Where R.sno=S.sno and R.cno=C.cno and S.dept<>C.dept )
)
```

参考答案 2:

```
select S.sno, S.sname
from Student S
where 5 > some ( Select count(*) from Register R, Course C
                  Where R.sno=S.sno and R.cno=C.cno and S.dept<>C.dept ) ;
```

//注：也可以用量词 any 或者 all 代替 some

8. 查询 2016 年度各门课程的成绩分布情况，结果返回课程编号，该课程的缺考人数（成绩为 NULL）、不及格人数、成绩优秀人数（成绩 $\geq 85$ ）。

Step 1: 首先创建三个视图，分别统计上述人数

缺考人数统计视图 V1:

```
Create view V1(sno, num1) as
select cno, count(*) from Register where regyear=2016 and grade=NULL group by cno ;
```

不及格人数统计视图 V2:

```
Create view V2(sno, num2) as
select cno, count(*) from Register where regyear=2016 and grade<60 group by cno ;
```

优秀人数统计视图 V3:

```
Create view V3(sno, num3) as
select cno, count(*) from Register where regyear=2016 and grade $\geq$ 85 group by cno ;
```

Step 2: 将上述三个视图联接起来

```
Select V1.cno, V1.num1 as 缺考人数, V2.num2 as 不及格人数, V3.num3 as 优秀人数
From V1, V2, V3
Where V1.cno=V2.cno and V2.cno=V3.cno ;
```

//说明:

①也可以不用创建视图，通过在 From 子句中嵌入三个子查询来实现；

②由于有些课程可能不存在缺考、不及格或成绩优秀的情况，所以，可考虑用‘外联接’将三个视图联接起来，但在这里不做要求。