现有关系模式如下：

学生（学号，姓名，性别，出生年月）；课程（课程号，课程名，教师姓名）；

选课表（课程号，学号，成绩）

1．检索年龄大于 20 岁的男生的学号和姓名。

```
select   stu_id,stu_anme
from student as s
        where 2020-year(birthday) >20   and   stu_sex=’M’;
```

2．检索选修了姓刘的老师所教授的课程的女学生的姓名。

```
select stu_name
     from student as s, stu_course as sc, course as c
where   s.stu_id=sc.stu_id   and   sc.course_id=c.course_id   and   sex=’F’   and   teacher like ’刘%’;
```

3．检索李想同学不学的课程的课程号和课程名。

```
select   course_id , course_name
   from   course as c
        where   not exists
               (select * from student as s, stu_course as sc
Where s.stu_id=sc.stu_id and sc.course_id=c.course_id and s.stu_name=’李想’);
```

4．检索至少选修了两门课程的学生的学号。

```
select   distinct   x.stu_id   ,x.stu_name
     from   stu_course as x, stu_course as   y
           where   x.stu_id=y.stu_id   and   x.course_id <> y.course_id;
```

5．求刘老师所教授课程的每门课的平均成绩。

```
 select c.course_id,avg(sc.grade)
    from stu_coure sc,course c
      Where sc.course_id = c.course_id and c.course_teacher like ‘刘%’
Group by c.course_id;
```

6．假设不存在重修的情况，请统计每门课的选修人数(选课人数超过两人的课程才统计)。要求显示课程号和人数，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 sc.course_id,count(sc.stu_id)
     From stu_course as sc
        Group by sc.course_id
          Having count(*) >2
Order by 2 desc,1;
```

7．求年龄大于所有女生年龄的男生的姓名和年龄。

```
Select s.stu_name,s.stu_age
```

```
         From student as s
           Where s.stu_sex=’M’ and
              s.stu_age >all                   //或者  >(select max(stu_age)…)
                (select b.stu_age
                    From student as b
Where b.stu_sex = ‘F’);
```

8．假定不存在重修的情况，求选修了所有课程的学生的学号姓名。(可以不用相关子查询做)

除了用 exists 查询，还有一种做法：

```
select studentnumber from sc group by studentnumber having count(*)=(select count(*) from
course)
```

9．　查询重修次数在 2 次以上的学生学号，课程号，重修次数

```
Select  学号,课程号, count(*) from sc group by  学号,课程号  having count(*)>=3
```

10．　查询被单个学生重修次数最多的课程号，课程名，教师姓名

```
    select  课程  from  选课  group by  学生号,课程 having count(*)>=all(
select count(*) from  选课  group by  学生号,课程  having count(*)>1)
```

学生（学号，姓名，年龄，性别，班级）
课程（课程号，课程名，先修课程号，学分）注意：此表的主键是(课程号)
选课（学号，课程号，教师号，成绩）
教师（教师号，教师名称）

1. 查找李力的所有不及格的课程名称和成绩，按成绩降序排列
2. 列出每门课的**学分**，选修的学生人数，及学生成绩的平均分
   ```
   select count(stuid), avg(成绩), avg(学分) from sc, course c where
       Sc.课程号=c.课程号  group by c.课程号
   ```
3. 选出所修课程总学分在 10 分以下的学生（注：不及格的课程没有学分）。
   ```
   Select stuid from c, sc where sc.courseid=c.courseid and 成绩>60 group by stuid
   having sum(学分)<10
   ```
4. 选出选课门数最多的学生学号及选课数量
   ```
   Select studentid,count(course) from sc group by studentid having count(course)>all (select
   count(course) from course group by student)
   ```
5. 列出每门课的最高分及获得该分数的学生
   ```
       SELECT StuID, courseid, score
       FROM    sc AS a
       WHERE EXISTS    (SELECT courseid    FROM    sc
                               Where courseid=a.courseid
                               GROUP BY courseid
                           HAVING (MAX(score) = a.sc_score))
   ```
6. 选出物理课得分比所有男学生的物理课平均分高的学生姓名
7. 选出修习过物理课的直接先修课的学生
   ```
   Select * from student where exists( select * from sc where sc.学号=student.学号  and  课程号
   in (select  先修课 from course where course.name='物理'))
   ```
8. 选出有两门以上先修课的课程（包括直接先修课、间接先修课）(用课程表)
   ```
       Select * from course c where exists(
       Select * from course c1, course c2 where c.先修=c1.课程号  and c1.先修=c2.
   课程号）
   ```