1、查询“c001”课程比“c002”课程成绩高的所有学生的学号；

select e.sno

from

(select sc.sno,sc.score from course,sc where course.cno=sc.cno and sc.cno='c001') e,(select sc.sno,sc.score from course,sc where course.cno=sc.cno and sc.cno='c002') d

where e.sno=d.sno

and e.score>d.score;

2、查询平均成绩大于60 分的同学的学号和平均成绩；

select sc.sno,avg(sc.score) from sc group by sc.sno having avg(sc.score)>60;

3、查询所有同学的学号、姓名、选课数、总成绩；

select sc.sno,student.sname,count(course.cno),sum(sc.score) from student,course,sc where student.sno=sc.sno

and course.cno=sc.cno group by student.sname,sc.sno;

4、查询姓“刘”的老师的个数；

select count(teacher.tname) from teacher where teacher.tname like '刘%';

5、查询没学过“谌燕”老师课的同学的学号、姓名；

select student.sno,student.sname

from student

where not(student.sname in

(select distinct student.sname from student,teacher,course,sc where teacher.tno=course.tno

and course.cno=sc.cno and sc.sno=student.sno

and teacher.tname='谌燕'));

6、查询学过“c001”并且也学过编号“c002”课程的同学的学号、姓名；

select sc.sno,student.sname from student,sc where student.sno=sc.sno

and (sc.cno='c001' or sc.cno='c002')

group by sc.sno,student.sname

having count(\*)=2;

7、查询学过“谌燕”老师所教的所有课的同学的学号、姓名；

select sc.sno,student.sname

from sc,student,(select course.cno from teacher,course where teacher.tno=course.tno

and teacher.tname='谌燕') e

where e.cno=sc.cno and student.sno=sc.sno

group by sc.sno,student.sname

having count(\*)=(select count(course.cno) from teacher,course where teacher.tno=course.tno

and teacher.tname='谌燕');

8、查询课程编号“c002”的成绩比课程编号“c001”课程低的所有同学的学号、姓名；

select e.sno

from

(select sc.sno,sc.score from course,sc where course.cno=sc.cno and sc.cno='c001') e,(select sc.sno,sc.score from course,sc where course.cno=sc.cno and sc.cno='c002') d

where e.sno=d.sno

and e.score>d.score;

9、查询所有课程成绩小于60 分的同学的学号、姓名；

select student.sno,student.sname

from student,sc

where student.sno=sc.sno

group by student.sno,student.sname

having max(sc.score)<60;

10、查询没有学全所有课的同学的学号、姓名；

select sc.sno,student.sname

from course,sc,student

where course.cno=sc.cno and student.sno=sc.sno

group by sc.sno,student.sname

having count(sc.cno)!=(select count(course.cno)

from course);

11、查询至少有学一门课跟“s001”的同学所学相同的同学的学号和姓名；

select distinct student.sno,student.sname from student,sc where student.sno=sc.sno and sc.cno in(select cno from sc where sno='s001')

and student.sno!='s001';

12、查询学过学号为“s001”同学所有门课的其他同学学号和姓名；

select s.sno,s.sname from student s where s.sno!='s001' and (not exists(

((select cno from sc where sc.sno=s.sno minus select cno from sc where sc.sno='s001')

union

(select cno from sc where sc.sno='s001' minus select cno from sc where sc.sno=s.sno)))

or

exists

(

select cno from sc where sc.sno=s.sno minus select cno from sc where sc.sno='s001'

))

;

或者：

select sc.sno,student.sname

from sc,student,(select cno from sc where sno='s001') b

where sc.cno=b.cno and student.sno=sc.sno and sc.sno!='s001'

group by sc.sno,student.sname

having count(sc.cno)=(select count(cno) from sc where sno='s001');

1. 把“SC”表中“谌燕”老师教的课的成绩都更改为此课程的平均成绩；

update sc s

set score=(

select t.avg\_score from (select sc.cno,avg(sc.score) avg\_score

from teacher,course,sc where teacher.tno=course.tno and sc.cno=course.cno

and teacher.tname='谌燕' group by sc.cno) t where s.cno=t.cno)

where exists(select 1 from (select sc.cno

from teacher,course,sc where teacher.tno=course.tno and sc.cno=course.cno

and teacher.tname='谌燕') t where s.cno=t.cno);

或者：

update sc set score =(select t.ag from(select course.cno, avg(sc.score) ag

from teacher,course,sc where teacher.tno=course.tno and course.cno=sc.cno and teacher.tname='谌燕' group by course.cno)t where sc.cno=t.cno)

where cno in(select course.cno

from teacher,course,sc where teacher.tno=course.tno and course.cno=sc.cno and teacher.tname='谌燕' group by course.cno)

14、查询和“s001”号的同学学习的课程完全相同的其他同学学号和姓名；

select s.sno,s.sname from student s where s.sno!='s001' and not exists(

((select cno from sc where sc.sno=s.sno minus select cno from sc where sc.sno='s001')

union

(select cno from sc where sc.sno='s001' minus select cno from sc where sc.sno=s.sno)));

15、删除学习“谌燕”老师课的SC 表记录；

delete from sc where cno in (select course.cno from teacher,course where teacher.tno=course.tno and teacher.tname='谌燕');

16、向SC 表中插入一些记录，这些记录要求符合以下条件：没有上过编号“c002”课程的同学学号、“c002”号课的平均成绩；

insert into sc(sno,cno,score) select sno,'c002',(select avg(score) from sc where cno='c002') from student where sno not in (select sno from sc where cno='c002');

17、查询各科成绩最高和最低的分：以如下形式显示：课程ID，最高分，最低分

select sc.cno,max(sc.score),min(sc.score) from sc group by sc.cno;

18、按各科平均成绩从低到高和及格率的百分数从高到低顺序

select t1.cno,t1.avg\_score,(t2.total2/t1.total1) as rate from (select cno,avg(score) as avg\_score, count(\*) as total1 from sc group by cno) t1,

(select cno,count(score) as total2 from sc where score>=60 group by cno) t2 where t1.cno=t2.cno order by t1.avg\_score,rate desc;

19、查询不同老师所教不同课程平均分从高到低显示

select c.tno,t.tname,sc.cno,avg(score) as avg\_score from sc, course c,teacher t where sc.cno=c.cno and c.tno=t.tno group by c.tno,sc.cno,t.tname order by avg\_score desc;

20、统计列印各科成绩,各分数段人数:课程ID,课程名称,[100-85],[85-70],[70-60],[ <60]

select c.cno, c.cname,

sum(case when sc.score between 86 and 100 then 1 else 0 end) as "[100-85]",

sum(case when sc.score between 70 and 85 then 1 else 0 end) as "[85-70]",

sum(case when sc.score between 60 and 69 then 1 else 0 end) as "[70-60]",

sum(case when sc.score <60 then 1 else 0 end) as "[<60]"

from sc,course c where sc.cno=c.cno group by c.cno, c.cname;

1. 查询各科成绩前三名的记录:(考虑成绩并列情况)

select b.\* from

(Select student.sname,course.cname,sc.score, dense\_rank()over(partition by course.cname order by sc.score desc)t

from student,course,sc where student.sno=sc.sno and course.cno=sc.cno) b

where b.t<=3

(不考虑成绩并列情况)

select b.\* from

(Select student.sname,course.cname,sc.score, row\_number() over (partition by course.cname order by sc.score desc) t

from student,course,sc where student.sno=sc.sno and course.cno=sc.cno) b

where b.t<=3;

22、查询每门课程被选修的学生数

select course.cname,count(sc.sno) from course,sc where course.cno=sc.cno group by course.cname;

23、查询出只选修了一门课程的全部学生的学号和姓名

select sc.sno,student.sname from student,sc where student.sno=sc.sno group by sc.sno,student.sname having count(sc.cno)=1;

24、查询男生、女生人数

select student.ssex,count(student.sno) from student group by student.ssex;

25、查询姓“张”的学生名单

select \* from student where sname like '张%';

26、查询同名同性学生名单，并统计同名人数

select s.\*, count(\*) over(partition by s.sname) as "同名" from (select s1.sno,s1.sname from student s1, student s2 where s1.sno!=s2.sno and s1.sname=s2.sname and s1.ssex=s2.ssex) s

;

27、1981年出生的学生名单(注：Student 表中Sage 列的类型是number

select sname from student where sage=to\_char(sysdate,'yyyy')-1981

28、查询每门课程的平均成绩，结果按平均成绩升序排列，平均成绩相同时，按课程号降序排列

select sc.cno,avg(sc.score) d from course,sc where course.cno=sc.cno group by sc.cno

order by d,sc.cno desc;

29、查询平均成绩大于85 的所有学生的学号、姓名和平均成绩

select sc.sno,student.sname,avg(sc.score) from student, sc where student.sno=sc.sno

group by sc.sno,student.sname having avg(sc.score)>85;

30、查询课程名称为“数据库”，且分数低于60 的学生姓名和分数

select s.sname,sc.score from student s, sc, course c where s.sno=sc.sno and c.cno=sc.cno and c.cname='Java Web' and sc.score<60;

31、查询所有学生的选课情况；

select student.sname,course.cname from course,sc,student where course.cno=sc.cno and student.sno=sc.sno order by student.sname;

32、查询任何一门课程成绩在70 分以上的姓名、课程名称和分数；

select student.sname,course.cname,sc.score from student,course,sc where student.sno=sc.sno

and sc.score>=70;

33、查询不及格的课程，并按课程号从大到小排列

select course.cno,course.cname from course,sc where course.cno=sc.cno and sc.score<60 order by course.cno desc;

34、查询课程编号为c001 且课程成绩在80 分以上的学生的学号和姓名；

select student.sno,student.sname,sc.score from student,course,sc where student.sno=sc.sno

and course.cno=sc.cno and course.cno='c001' and sc.score>80;

35、求选了课程的学生人数

select count(distinct sno) from sc;

36、查询选修“谌燕”老师所授课程的学生中，成绩最高的学生姓名及其成绩

select student.sname,sc.score

from student,sc,teacher,course

where student.sno=sc.sno and teacher.tno=course.tno and

course.cno=sc.cno

and sc.score=(select max(sc.score) from teacher,course,sc where teacher.tno=course.tno and

course.cno=sc.cno and teacher.tname='谌燕')

and teacher.tname='谌燕'

37、查询各个课程及相应的选修人数

select course.cname,count(sc.sno) from sc,course where

course.cno=sc.cno group by course.cname;

1. 查询不同课程成绩相同的学生的学号、课程号、学生成绩

只要有成绩相同就符合

select distinct b.sno,b.cno,b.score from sc b,sc d where b.score=d.score and b.cno!=d.cno

全部课程的成绩都相同

select \* from sc where sno in

(select t.sno

from

(select sc.\*,rank()over(partition by sc.sno order by sc.score) d from sc) t

group by t.sno

having count(t.cno)=sum(t.d) and count(t.cno)>1)

或者：

select s.sno,s.cno,s.score from sc s where (select count(cno) from sc where sc.sno=s.sno)>1 and s.sno

not in(select distinct s1.sno from sc s1, sc s2 where s1.sno=s2.sno and s1.cno!=s2.cno and s1.score!=s2.score);

1. 查询每门功课成绩最好的前两名

select b.\* from

(Select student.sname,course.cname,sc.score, dense\_rank() over(partition by course.cname order by sc.score desc) t

from student,course,sc where student.sno=sc.sno and course.cno=sc.cno) b

where b.t<=2

40、统计每门课程的学生选修人数（超过10 人的课程才统计）。要求输出课程号和选修人数，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.cno,count(sc.sno) from sc group by sc.cno having count(sc.sno)>10 order by count(sc.sno) desc,sc.cno;

41、检索至少选修两门课程的学生学号

select sno from sc group by sno having count(cno)>=2

42、查询全部学生都选修的课程的课程号和课程名

select course.cname,course.cno from course,sc,student where course.cno=sc.cno and student.sno=sc.sno group by course.cname,course.cno

having count(sc.sno)=(select count(student.sno) from student);

43、查询没学过“谌燕”老师讲授的任一门课程的学生姓名

select student.sname

from student

where not(student.sname=any

(select distinct student.sname from student,teacher,course,sc where teacher.tno=course.tno

and course.cno=sc.cno and sc.sno=student.sno

and teacher.tname='谌燕'));

44、查询两门以上不及格课程的同学的学号及其平均成绩

select sno,avg(score)

from sc

where sno=(select sno from sc where score<60 group by sno having count(cno)>=2)

group by sno;

45、检索“c004”课程分数小于60，按分数降序排列的同学学号

select sno from sc where cno='c004' and score<60 order by score desc;

46、删除“s002”同学的“c001”课程的成绩

delete from sc where sno='s002' and cno='c001';