问题及描述：

–1.学生表

Student(SID,Sname,Sage,Ssex) --SID 学生编号,Sname 学生姓名,Sage 出生年月,Ssex 学生性别

–2.课程表

Course(CID,Cname,TID) --CID --课程编号,Cname 课程名称,TID 教师编号

–3.教师表

Teacher(TID,Tname) --TID 教师编号,Tname 教师姓名

–4.成绩表

SC(SID,CID,score) --SID 学生编号,CID 课程编号,score 分数

\*/

–创建测试数据

create table Student(SID varchar(10),Sname nvarchar(10),Sage datetime,Ssex nvarchar(10));

insert into Student values(‘01’ , ‘赵雷’ , ‘1990-01-01’ , ‘男’);

insert into Student values(‘02’ , ‘钱电’ , ‘1990-12-21’ , ‘男’);

insert into Student values(‘03’ , ‘孙风’ , ‘1990-05-20’ , ‘男’);

insert into Student values(‘04’ , ‘李云’ , ‘1990-08-06’ , ‘男’);

insert into Student values(‘05’ , ‘周梅’ , ‘1991-12-01’ , ‘女’);

insert into Student values(‘06’ , ‘吴兰’ , ‘1992-03-01’ , ‘女’);

insert into Student values(‘07’ , ‘郑竹’ , ‘1989-07-01’ , ‘女’);

insert into Student values(‘08’ , ‘王菊’ , ‘1990-01-20’ , ‘女’);

create table Course(CID varchar(10),Cname nvarchar(10),TID varchar(10));

insert into Course values(‘01’ , ‘语文’ , ‘02’);

insert into Course values(‘02’ , ‘数学’ , ‘01’);

insert into Course values(‘03’ , ‘英语’ , ‘03’);

create table Teacher(TID varchar(10),Tname nvarchar(10));

insert into Teacher values(‘01’ , ‘张三’);

insert into Teacher values(‘02’ , ‘李四’);

insert into Teacher values(‘03’ , ‘王五’);

create table SC(SID varchar(10),CID varchar(10),score decimal(18,1));

insert into SC values(‘01’ , ‘01’ , 80);

insert into SC values(‘01’ , ‘02’ , 90);

insert into SC values(‘01’ , ‘03’ , 99);

insert into SC values(‘02’ , ‘01’ , 70);

insert into SC values(‘02’ , ‘02’ , 60);

insert into SC values(‘02’ , ‘03’ , 80);

insert into SC values(‘03’ , ‘01’ , 80);

insert into SC values(‘03’ , ‘02’ , 80);

insert into SC values(‘03’ , ‘03’ , 80);

insert into SC values(‘04’ , ‘01’ , 50);

insert into SC values(‘04’ , ‘02’ , 30);

insert into SC values(‘04’ , ‘03’ , 20);

insert into SC values(‘05’ , ‘01’ , 76);

insert into SC values(‘05’ , ‘02’ , 87);

insert into SC values(‘06’ , ‘01’ , 31);

insert into SC values(‘06’ , ‘03’ , 34);

insert into SC values(‘07’ , ‘02’ , 89);

insert into SC values(‘07’ , ‘03’ , 98);

开始练习

–1、查询"01"课程比"02"课程成绩高的学生的信息及课程分数

–1.1、查询同时存在"01"课程和"02"课程的情况

select a.\* , b.score,c.score from Student a , SC b , SC c

where a.SID = b.SID and a.SID = c.SID and b.CID = ‘01’ and c.CID = ‘02’ and b.score > c.score

–1.2、查询同时存在"01"课程和"02"课程的情况和存在"01"课程但可能不存在"02"课程的情况(不存在时显示为null)(以下存在相同内容时不再解释)

select a.\* , b.score,c.score from Student a

left join SC b on a.SID = b.SID and b.CID = ‘01’

left join SC c on a.SID = c.SID and c.CID = ‘02’

where b.score > ifnull(c.score,0)

–2、查询"01"课程比"02"课程成绩低的学生的信息及课程分数

–2.1、查询同时存在"01"课程和"02"课程的情况

select a.\* , b.score ,c.score from Student a , SC b , SC c

where a.SID = b.SID and a.SID = c.SID and b.CID = ‘01’ and c.CID = ‘02’ and b.score < c.score

–2.2、查询同时存在"01"课程和"02"课程的情况和不存在"01"课程但存在"02"课程的情况

select a.\* , b.score , c.score from Student a

left join SC b on a.SID = b.SID and b.CID = ‘01’

left join SC c on a.SID = c.SID and c.CID = ‘02’

where ifnull(b.score,0) < c.score

–3、查询平均成绩大于等于60分的同学的学生编号和学生姓名和平均成绩

select a.SID , a.Sname , cast(avg(b.score) as decimal(18,2)) avg\_score

from Student a , sc b

where a.SID = b.SID

group by a.SID , a.Sname

having cast(avg(b.score) as decimal(18,2)) >= 60

order by a.SID

–4、查询平均成绩小于60分的同学的学生编号和学生姓名和平均成绩

–4.1、查询在sc表存在成绩的学生信息的SQL语句。

select a.SID , a.Sname , cast(avg(b.score) as decimal(18,2)) avg\_score

from Student a , sc b

where a.SID = b.SID

group by a.SID , a.Sname

having cast(avg(b.score) as decimal(18,2)) < 60

order by a.SID

–4.2、查询在sc表中不存在成绩的学生信息的SQL语句。

select a.SID , a.Sname , isnull(cast(avg(b.score) as decimal(18,2)),0) avg\_score

from Student a left join sc b

on a.SID = b.SID

group by a.SID , a.Sname

having isnull(cast(avg(b.score) as decimal(18,2)),0) < 60

order by a.SID

–5、查询所有同学的学生编号、学生姓名、选课总数、所有课程的总成绩

–5.1、查询所有有成绩的SQL。

select a.SID 学生编号 , a.Sname 学生姓名 , count(b.CID) 选课总数, sum(score) 所有课程的总成绩

from Student a , SC b

where a.SID = b.SID

group by a.SID,a.Sname

order by a.SID

–5.2、查询所有(包括有成绩和无成绩)的SQL。

select a.SID 学生编号 , a.Sname 学生姓名 , count(b.CID) 选课总数, sum(score) 所有课程的总成绩

from Student a left join SC b

on a.SID = b.SID

group by a.SID,a.Sname

order by a.SID

–6、查询"李"姓老师的数量

–方法1

select count(Tname) 李姓老师的数量 from Teacher where Tname like ‘李%’

–方法2

select count(Tname) 李姓老师的数量 from Teacher where left(Tname,1) = ‘李’

–7、查询学过"张三"老师授课的同学的信息

select distinct Student.\* from Student , SC , Course , Teacher

where Student.SID = SC.SID and SC.CID = Course.CID and Course.TID = Teacher.TID and Teacher.Tname = ‘张三’

order by Student.SID

–8、查询没学过"张三"老师授课的同学的信息

select m.\* from Student m where SID not in (select distinct SC.SID from SC , Course , Teacher where SC.CID = Course.CID and Course.TID = Teacher.TID and Teacher.Tname = ‘张三’) order by m.SID

–9、查询学过编号为"01"并且也学过编号为"02"的课程的同学的信息

–方法1

select Student.\* from Student , SC where Student.SID = SC.SID and SC.CID = ‘01’ and exists (Select 1 from SC SC\_2 where SC\_2.SID = SC.SID and SC\_2.CID = ‘02’) order by Student.SID

–方法2

select Student.\* from Student , SC where Student.SID = SC.SID and SC.CID = ‘02’ and exists (Select 1 from SC SC\_2 where SC\_2.SID = SC.SID and SC\_2.CID = ‘01’) order by Student.SID

–方法3

select m.\* from Student m where SID in

(

select SID from

(

select distinct SID from SC where CID = ‘01’

union all

select distinct SID from SC where CID = ‘02’

) t group by SID having count(1) = 2

)

order by m.SID

–10、查询学过编号为"01"但是没有学过编号为"02"的课程的同学的信息

–方法1

select Student.\* from Student , SC where Student.SID = SC.SID and SC.CID = ‘01’ and not exists (Select 1 from SC SC\_2 where SC\_2.SID = SC.SID and SC\_2.CID = ‘02’) order by Student.SID

–方法2

select Student.\* from Student , SC where Student.SID = SC.SID and SC.CID = ‘01’ and Student.SID not in (Select SC\_2.SID from SC SC\_2 where SC\_2.SID = SC.SID and SC\_2.CID = ‘02’) order by Student.SID

–11、查询没有学全所有课程的同学的信息

–11.1、

select Student.\*

from Student , SC

where Student.SID = SC.SID

group by Student.SID , Student.Sname , Student.Sage , Student.Ssex having count(CID) < (select count(CID) from Course)

–11.2

select Student.\*

from Student left join SC

on Student.SID = SC.SID

group by Student.SID , Student.Sname , Student.Sage , Student.Ssex having count(CID) < (select count(CID) from Course)

–12、查询至少有一门课与学号为"01"的同学所学相同的同学的信息

select distinct Student.\* from Student , SC where Student.SID = SC.SID and SC.CID in (select CID from SC where SID = ‘01’) and Student.SID <> ‘01’

–13、查询和"01"号的同学学习的课程完全相同的其他同学的信息

select Student.\* from Student where SID in

(select distinct SC.SID from SC where SID <> ‘01’ and SC.CID in (select distinct CID from SC where SID = ‘01’)

group by SC.SID having count(1) = (select count(1) from SC where SID=‘01’))

–14、查询没学过"张三"老师讲授的任一门课程的学生姓名

select student.\* from student where student.SID not in

(select distinct sc.SID from sc , course , teacher where sc.CID = course.CID and course.TID = teacher.TID and teacher.tname = ‘张三’)

order by student.SID

–15、查询两门及其以上不及格课程的同学的学号，姓名及其平均成绩

select student.SID , student.sname , cast(avg(score) as decimal(18,2)) avg\_score from student , sc

where student.SID = SC.SID and student.SID in (select SID from SC where score < 60 group by SID having count(1) >= 2)

group by student.SID , student.sname

–16、检索"01"课程分数小于60，按分数降序排列的学生信息

select student.\* , sc.CID , sc.score from student , sc

where student.SID = SC.SID and sc.score < 60 and sc.CID = ‘01’

order by sc.score desc

–17、按平均成绩从高到低显示所有学生的所有课程的成绩以及平均成绩

–17.1 SQL 2000 静态

select a.SID 学生编号 , a.Sname 学生姓名 ,

max(case c.Cname when ‘语文’ then b.score else null end) 语文 ,

max(case c.Cname when ‘数学’ then b.score else null end) 数学 ,

max(case c.Cname when ‘英语’ then b.score else null end) 英语 ,

cast(avg(b.score) as decimal(18,2)) 平均分

from Student a

left join SC b on a.SID = b.SID

left join Course c on b.CID = c.CID

group by a.SID , a.Sname

order by 平均分 desc

–17.2 SQL 2000 动态

declare @sql nvarchar(4000)

set @sql = ‘select a.SID ’ + ‘学生编号’ + ’ , a.Sname ’ + ‘学生姓名’

select @sql = @sql + ‘,max(case c.Cname when ‘’’+Cname+’’’ then b.score else null end) ‘+Cname+’ ’

from (select distinct Cname from Course) as t

set @sql = @sql + ’ , cast(avg(b.score) as decimal(18,2)) ’ + ‘平均分’ + ’ from Student a left join SC b on a.SID = b.SID left join Course c on b.CID = c.CID

group by a.SID , a.Sname order by ’ + ‘平均分’ + ’ desc’

exec(@sql)

–18、查询各科成绩最高分、最低分和平均分：以如下形式显示：课程ID，课程name，最高分，最低分，平均分，及格率，中等率，优良率，优秀率

–及格为>=60，中等为：70-80，优良为：80-90，优秀为：>=90

–方法1

select m.CID 课程编号 , m.Cname 课程名称 ,

max(n.score) 最高分 ,

min(n.score) 最低分 ,

cast(avg(n.score) as decimal(18,2)) 平均分 ,

cast((select count(1) from SC where CID = m.CID and score >= 60)\*100.0 / (select count(1) from SC where CID = m.CID) as decimal(18,2)) 及格率 ,

cast((select count(1) from SC where CID = m.CID and score >= 70 and score < 80 )\*100.0 / (select count(1) from SC where CID = m.CID) as decimal(18,2)) 中等率 ,

cast((select count(1) from SC where CID = m.CID and score >= 80 and score < 90 )\*100.0 / (select count(1) from SC where CID = m.CID) as decimal(18,2)) 优良率 ,

cast((select count(1) from SC where CID = m.CID and score >= 90)\*100.0 / (select count(1) from SC where CID = m.CID) as decimal(18,2)) 优秀率

from Course m , SC n

where m.CID = n.CID

group by m.CID , m.Cname

order by m.CID

–方法2

select m.CID 课程编号 , m.Cname 课程名称 ,

(select max(score) from SC where CID = m.CID) 最高分 ,

(select min(score) from SC where CID = m.CID) 最低分 ,

(select cast(avg(score) as decimal(18,2)) from SC where CID = m.CID) 平均分 ,

cast((select count(1) from SC where CID = m.CID and score >= 60)\*100.0 / (select count(1) from SC where CID = m.CID) as decimal(18,2)) 及格率,

cast((select count(1) from SC where CID = m.CID and score >= 70 and score < 80 )\*100.0 / (select count(1) from SC where CID = m.CID) as decimal(18,2)) 中等率 ,

cast((select count(1) from SC where CID = m.CID and score >= 80 and score < 90 )\*100.0 / (select count(1) from SC where CID = m.CID) as decimal(18,2)) 优良率 ,

cast((select count(1) from SC where CID = m.CID and score >= 90)\*100.0 / (select count(1) from SC where CID = m.CID) as decimal(18,2)) 优秀率

from Course m

order by m.CID

–19、按各科成绩进行排序，并显示排名

–19.1 sql 2000用子查询完成

–Score重复时保留名次空缺

select t.\* , px = (select count(1) from SC where CID = t.CID and score > t.score) + 1 from sc t order by t.cid , px

–Score重复时合并名次

select t.\* , px = (select count(distinct score) from SC where CID = t.CID and score >= t.score) from sc t order by t.cid , px

–19.2 sql 2005用rank,DENSE\_RANK完成

–Score重复时保留名次空缺(rank完成)

select t.\* , px = rank() over(partition by cid order by score desc) from sc t order by t.CID , px

–Score重复时合并名次(DENSE\_RANK完成)

select t.\* , px = DENSE\_RANK() over(partition by cid order by score desc) from sc t order by t.CID , px

–20、查询学生的总成绩并进行排名

–20.1 查询学生的总成绩

select m.SID 学生编号 ,

m.Sname 学生姓名 ,

isnull(sum(score),0) 总成绩

from Student m left join SC n on m.SID = n.SID

group by m.SID , m.Sname

order by 总成绩 desc

–20.2 查询学生的总成绩并进行排名，sql 2000用子查询完成，分总分重复时保留名次空缺和不保留名次空缺两种。

select t1.\* , px = (select count(1) from

(

select m.SID 学生编号 ,

m.Sname 学生姓名 ,

isnull(sum(score),0) 总成绩

from Student m left join SC n on m.SID = n.SID

group by m.SID , m.Sname

) t2 where 总成绩 > t1.总成绩) + 1 from

(

select m.SID 学生编号 ,

m.Sname 学生姓名 ,

isnull(sum(score),0) 总成绩

from Student m left join SC n on m.SID = n.SID

group by m.SID , m.Sname

) t1

order by px

select t1.\* , px = (select count(distinct 总成绩) from

(

select m.SID 学生编号 ,

m.Sname 学生姓名 ,

isnull(sum(score),0) 总成绩

from Student m left join SC n on m.SID = n.SID

group by m.SID , m.Sname

) t2 where 总成绩 >= t1.总成绩) from

(

select m.SID 学生编号 ,

m.Sname 学生姓名 ,

isnull(sum(score),0) 总成绩

from Student m left join SC n on m.SID = n.SID

group by m.SID , m.Sname

) t1

order by px

–20.3 查询学生的总成绩并进行排名，sql 2005用rank,DENSE\_RANK完成，分总分重复时保留名次空缺和不保留名次空缺两种。

select t.\* , px = rank() over(order by 总成绩 desc) from

(

select m.SID 学生编号 ,

m.Sname 学生姓名 ,

isnull(sum(score),0) 总成绩

from Student m left join SC n on m.SID = n.SID

group by m.SID , m.Sname

) t

order by px

select t.\* , px = DENSE\_RANK() over(order by 总成绩 desc) from

(

select m.SID 学生编号 ,

m.Sname 学生姓名 ,

isnull(sum(score),0) 总成绩

from Student m left join SC n on m.SID = n.SID

group by m.SID , m.Sname

) t

order by px

–21、查询不同老师所教不同课程平均分从高到低显示

select m.TID , m.Tname , cast(avg(o.score) as decimal(18,2)) avg\_score

from Teacher m , Course n , SC o

where m.TID = n.TID and n.CID = o.CID

group by m.TID , m.Tname

order by avg\_score desc

–22、查询所有课程的成绩第2名到第3名的学生信息及该课程成绩

–22.1 sql 2000用子查询完成

–Score重复时保留名次空缺

select \* from (select t.\* , px = (select count(1) from SC where CID = t.CID and score > t.score) + 1 from sc t) m where px between 2 and 3 order by m.cid , m.px

–Score重复时合并名次

select \* from (select t.\* , px = (select count(distinct score) from SC where CID = t.CID and score >= t.score) from sc t) m where px between 2 and 3 order by m.cid , m.px

–22.2 sql 2005用rank,DENSE\_RANK完成

–Score重复时保留名次空缺(rank完成)

select \* from (select t.\* , px = rank() over(partition by cid order by score desc) from sc t) m where px between 2 and 3 order by m.CID , m.px

–Score重复时合并名次(DENSE\_RANK完成)

select \* from (select t.\* , px = DENSE\_RANK() over(partition by cid order by score desc) from sc t) m where px between 2 and 3 order by m.CID , m.px

–23、统计各科成绩各分数段人数：课程编号,课程名称, 100-85 , 85-70 , 70-60 , 0-60 及所占百分比

–23.1 统计各科成绩各分数段人数：课程编号,课程名称, 100-85 , 85-70 , 70-60 , 0-60

–横向显示

select Course.CID 课程编号 , Cname as 课程名称 ,

sum(case when score >= 85 then 1 else 0 end) 85-100 ,

sum(case when score >= 70 and score < 85 then 1 else 0 end) 70-85 ,

sum(case when score >= 60 and score < 70 then 1 else 0 end) 60-70 ,

sum(case when score < 60 then 1 else 0 end) 0-60

from sc , Course

where SC.CID = Course.CID

group by Course.CID , Course.Cname

order by Course.CID

–纵向显示1(显示存在的分数段)

select m.CID 课程编号 , m.Cname 课程名称 , 分数段 = (

case when n.score >= 85 then ‘85-100’

when n.score >= 70 and n.score < 85 then ‘70-85’

when n.score >= 60 and n.score < 70 then ‘60-70’

else ‘0-60’

end) ,

count(1) 数量

from Course m , sc n

where m.CID = n.CID

group by m.CID , m.Cname , (

case when n.score >= 85 then ‘85-100’

when n.score >= 70 and n.score < 85 then ‘70-85’

when n.score >= 60 and n.score < 70 then ‘60-70’

else ‘0-60’

end)

order by m.CID , m.Cname , 分数段

–纵向显示2(显示存在的分数段，不存在的分数段用0显示)

select m.CID 课程编号 , m.Cname 课程名称 , 分数段 = (

case when n.score >= 85 then ‘85-100’

when n.score >= 70 and n.score < 85 then ‘70-85’

when n.score >= 60 and n.score < 70 then ‘60-70’

else ‘0-60’

end) ,

count(1) 数量

from Course m , sc n

where m.CID = n.CID

group by all m.CID , m.Cname , (

case when n.score >= 85 then ‘85-100’

when n.score >= 70 and n.score < 85 then ‘70-85’

when n.score >= 60 and n.score < 70 then ‘60-70’

else ‘0-60’

end)

order by m.CID , m.Cname , 分数段

–23.2 统计各科成绩各分数段人数：课程编号,课程名称, 100-85 , 85-70 , 70-60 , <60 及所占百分比

–横向显示

select m.CID 课程编号, m.Cname 课程名称,

(select count(1) from SC where CID = m.CID and score < 60) 0-60 ,

cast((select count(1) from SC where CID = m.CID and score < 60)\*100.0 / (select count(1) from SC where CID = m.CID) as decimal(18,2)) 百分比 ,

(select count(1) from SC where CID = m.CID and score >= 60 and score < 70) 60-70 ,

cast((select count(1) from SC where CID = m.CID and score >= 60 and score < 70)\*100.0 / (select count(1) from SC where CID = m.CID) as decimal(18,2)) 百分比 ,

(select count(1) from SC where CID = m.CID and score >= 70 and score < 85) 70-85 ,

cast((select count(1) from SC where CID = m.CID and score >= 70 and score < 85)\*100.0 / (select count(1) from SC where CID = m.CID) as decimal(18,2)) 百分比 ,

(select count(1) from SC where CID = m.CID and score >= 85) 85-100 ,

cast((select count(1) from SC where CID = m.CID and score >= 85)\*100.0 / (select count(1) from SC where CID = m.CID) as decimal(18,2)) 百分比

from Course m

order by m.CID

–纵向显示1(显示存在的分数段)

select m.CID 课程编号 , m.Cname 课程名称 , 分数段 = (

case when n.score >= 85 then ‘85-100’

when n.score >= 70 and n.score < 85 then ‘70-85’

when n.score >= 60 and n.score < 70 then ‘60-70’

else ‘0-60’

end) ,

count(1) 数量 ,

cast(count(1) \* 100.0 / (select count(1) from sc where CID = m.CID) as decimal(18,2)) 百分比

from Course m , sc n

where m.CID = n.CID

group by m.CID , m.Cname , (

case when n.score >= 85 then ‘85-100’

when n.score >= 70 and n.score < 85 then ‘70-85’

when n.score >= 60 and n.score < 70 then ‘60-70’

else ‘0-60’

end)

order by m.CID , m.Cname , 分数段

–纵向显示2(显示存在的分数段，不存在的分数段用0显示)

select m.CID 课程编号 , m.Cname 课程名称 , 分数段 = (

case when n.score >= 85 then ‘85-100’

when n.score >= 70 and n.score < 85 then ‘70-85’

when n.score >= 60 and n.score < 70 then ‘60-70’

else ‘0-60’

end) ,

count(1) 数量 ,

cast(count(1) \* 100.0 / (select count(1) from sc where CID = m.CID) as decimal(18,2)) 百分比

from Course m , sc n

where m.CID = n.CID

group by all m.CID , m.Cname , (

case when n.score >= 85 then ‘85-100’

when n.score >= 70 and n.score < 85 then ‘70-85’

when n.score >= 60 and n.score < 70 then ‘60-70’

else ‘0-60’

end)

order by m.CID , m.Cname , 分数段

–24、查询学生平均成绩及其名次

–24.1 查询学生的平均成绩并进行排名，sql 2000用子查询完成，分平均成绩重复时保留名次空缺和不保留名次空缺两种。

select t1.\* , px = (select count(1) from

(

select m.SID 学生编号 ,

m.Sname 学生姓名 ,

isnull(cast(avg(score) as decimal(18,2)),0) 平均成绩

from Student m left join SC n on m.SID = n.SID

group by m.SID , m.Sname

) t2 where 平均成绩 > t1.平均成绩) + 1 from

(

select m.SID 学生编号 ,

m.Sname 学生姓名 ,

isnull(cast(avg(score) as decimal(18,2)),0) 平均成绩

from Student m left join SC n on m.SID = n.SID

group by m.SID , m.Sname

) t1

order by px

select t1.\* , px = (select count(distinct 平均成绩) from

(

select m.SID 学生编号 ,

m.Sname 学生姓名 ,

isnull(cast(avg(score) as decimal(18,2)),0) 平均成绩

from Student m left join SC n on m.SID = n.SID

group by m.SID , m.Sname

) t2 where 平均成绩 >= t1.平均成绩) from

(

select m.SID 学生编号 ,

m.Sname 学生姓名 ,

isnull(cast(avg(score) as decimal(18,2)),0) 平均成绩

from Student m left join SC n on m.SID = n.SID

group by m.SID , m.Sname

) t1

order by px

–24.2 查询学生的平均成绩并进行排名，sql 2005用rank,DENSE\_RANK完成，分平均成绩重复时保留名次空缺和不保留名次空缺两种。

select t.\* , px = rank() over(order by 平均成绩 desc) from

(

select m.SID 学生编号 ,

m.Sname 学生姓名 ,

isnull(cast(avg(score) as decimal(18,2)),0) 平均成绩

from Student m left join SC n on m.SID = n.SID

group by m.SID , m.Sname

) t

order by px

select t.\* , px = DENSE\_RANK() over(order by 平均成绩 desc) from

(

select m.SID 学生编号 ,

m.Sname 学生姓名 ,

isnull(cast(avg(score) as decimal(18,2)),0) 平均成绩

from Student m left join SC n on m.SID = n.SID

group by m.SID , m.Sname

) t

order by px

–25、查询各科成绩前三名的记录

–25.1 分数重复时保留名次空缺

select m.\* , n.CID , n.score from Student m, SC n where m.SID = n.SID and n.score in

(select top 3 score from sc where CID = n.CID order by score desc) order by n.CID , n.score desc

–25.2 分数重复时不保留名次空缺，合并名次

–sql 2000用子查询实现

select \* from (select t.\* , px = (select count(distinct score) from SC where CID = t.CID and score >= t.score) from sc t) m where px between 1 and 3 order by m.Cid , m.px

–sql 2005用DENSE\_RANK实现

select \* from (select t.\* , px = DENSE\_RANK() over(partition by Cid order by score desc) from sc t) m where px between 1 and 3 order by m.CID , m.px

–26、查询每门课程被选修的学生数

select Cid , count(SID) 学生数 from sc group by CID

–27、查询出只有两门课程的全部学生的学号和姓名

select Student.SID , Student.Sname

from Student , SC

where Student.SID = SC.SID

group by Student.SID , Student.Sname

having count(SC.CID) = 2

order by Student.SID

–28、查询男生、女生人数

select count(Ssex) as 男生人数 from Student where Ssex = N’男’

select count(Ssex) as 女生人数 from Student where Ssex = N’女’

select sum(case when Ssex = N’男’ then 1 else 0 end) 男生人数 ,sum(case when Ssex = N’女’ then 1 else 0 end) 女生人数 from student

select case when Ssex = N’男’ then N’男生人数’ else N’女生人数’ end 男女情况 , count(1) 人数 from student group by case when Ssex = N’男’ then N’男生人数’ else N’女生人数’ end

–29、查询名字中含有"风"字的学生信息

select \* from student where sname like N’%风%’

select \* from student where charindex(N’风’ , sname) > 0

–30、查询同名同性学生名单，并统计同名人数

select Sname 学生姓名 , count() 人数 from Student group by Sname having count() > 1

–31、查询1990年出生的学生名单(注：Student表中Sage列的类型是datetime)

select \* from Student where year(sage) = 1990

select \* from Student where datediff(yy,sage,‘1990-01-01’) = 0

select \* from Student where datepart(yy,sage) = 1990

select \* from Student where convert(varchar(4),sage,120) = ‘1990’

–32、查询每门课程的平均成绩，结果按平均成绩降序排列，平均成绩相同时，按课程编号升序排列

select m.CID , m.Cname , cast(avg(n.score) as decimal(18,2)) avg\_score

from Course m, SC n

where m.CID = n.CID

group by m.CID , m.Cname

order by avg\_score desc, m.CID asc

–33、查询平均成绩大于等于85的所有学生的学号、姓名和平均成绩

select a.SID , a.Sname , cast(avg(b.score) as decimal(18,2)) avg\_score

from Student a , sc b

where a.SID = b.SID

group by a.SID , a.Sname

having cast(avg(b.score) as decimal(18,2)) >= 85

order by a.SID

–34、查询课程名称为"数学"，且分数低于60的学生姓名和分数

select sname , score

from Student , SC , Course

where SC.SID = Student.SID and SC.CID = Course.CID and Course.Cname = N’数学’ and score < 60

–35、查询所有学生的课程及分数情况；

select Student.\* , Course.Cname , SC.CID , SC.score

from Student, SC , Course

where Student.SID = SC.SID and SC.CID = Course.CID

order by Student.SID , SC.CID

–36、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.\* , Course.Cname , SC.CID , SC.score

from Student, SC , Course

where Student.SID = SC.SID and SC.CID = Course.CID and SC.score >= 70

order by Student.SID , SC.CID

–37、查询不及格的课程

select Student.\* , Course.Cname , SC.CID , SC.score

from Student, SC , Course

where Student.SID = SC.SID and SC.CID = Course.CID and SC.score < 60

order by Student.SID , SC.CID

–38、查询课程编号为01且课程成绩在80分以上的学生的学号和姓名；

select Student.\* , Course.Cname , SC.CID , SC.score

from Student, SC , Course

where Student.SID = SC.SID and SC.CID = Course.CID and SC.CID = ‘01’ and SC.score >= 80

order by Student.SID , SC.CID

–39、求每门课程的学生人数

select Course.CID , Course.Cname , count(\*) 学生人数

from Course , SC

where Course.CID = SC.CID

group by Course.CID , Course.Cname

order by Course.CID , Course.Cname

–40、查询选修"张三"老师所授课程的学生中，成绩最高的学生信息及其成绩

–40.1 当最高分只有一个时

select top 1 Student.\* , Course.Cname , SC.CID , SC.score

from Student, SC , Course , Teacher

where Student.SID = SC.SID and SC.CID = Course.CID and Course.TID = Teacher.TID and Teacher.Tname = N’张三’

order by SC.score desc

–40.2 当最高分出现多个时

select Student.\* , Course.Cname , SC.CID , SC.score

from Student, SC , Course , Teacher

where Student.SID = SC.SID and SC.CID = Course.CID and Course.TID = Teacher.TID and Teacher.Tname = N’张三’ and

SC.score = (select max(SC.score) from SC , Course , Teacher where SC.CID = Course.CID and Course.TID = Teacher.TID and Teacher.Tname = N’张三’)

–41、查询不同课程成绩相同的学生的学生编号、课程编号、学生成绩

–方法1

select m.\* from SC m ,(select CID , score from SC group by CID , score having count(1) > 1) n

where m.CID= n.CID and m.score = n.score order by m.CID , m.score , m.SID

–方法2

select m.\* from SC m where exists (select 1 from (select CID , score from SC group by CID , score having count(1) > 1) n

where m.CID= n.CID and m.score = n.score) order by m.CID , m.score , m.SID

–42、查询每门功成绩最好的前两名

select t.\* from sc t where score in (select top 2 score from sc where CID = T.CID order by score desc) order by t.CID , t.score desc

–43、统计每门课程的学生选修人数（超过5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 Course.CID , Course.Cname , count() 学生人数

from Course , SC

where Course.CID = SC.CID

group by Course.CID , Course.Cname

having count() >= 5

order by 学生人数 desc , Course.CID

–44、检索至少选修两门课程的学生学号

select student.SID , student.Sname

from student , SC

where student.SID = SC.SID

group by student.SID , student.Sname

having count(1) >= 2

order by student.SID

–45、查询选修了全部课程的学生信息

–方法1 根据数量来完成

select student.\* from student where SID in

(select SID from sc group by SID having count(1) = (select count(1) from course))

–方法2 使用双重否定来完成

select t.\* from student t where t.SID not in

(

select distinct m.SID from

(

select SID , CID from student , course

) m where not exists (select 1 from sc n where n.SID = m.SID and n.CID = m.CID)

)

–方法3 使用双重否定来完成

select t.\* from student t where not exists(select 1 from

(

select distinct m.SID from

(

select SID , CID from student , course

) m where not exists (select 1 from sc n where n.SID = m.SID and n.CID = m.CID)

) k where k.SID = t.SID

)

–46、查询各学生的年龄

–46.1 只按照年份来算

select \* , datediff(yy , sage , getdate()) 年龄 from student

–46.2 按照出生日期来算，当前月日 < 出生年月的月日则，年龄减一

select \* , case when right(convert(varchar(10),getdate(),120),5) < right(convert(varchar(10),sage,120),5) then datediff(yy , sage , getdate()) - 1 else datediff(yy , sage , getdate()) end 年龄 from student

–47、查询本周过生日的学生

select \* from student where datediff(week,datename(yy,getdate()) + right(convert(varchar(10),sage,120),6),getdate()) = 0

–48、查询下周过生日的学生

select \* from student where datediff(week,datename(yy,getdate()) + right(convert(varchar(10),sage,120),6),getdate()) = -1

–49、查询本月过生日的学生

select \* from student where datediff(mm,datename(yy,getdate()) + right(convert(varchar(10),sage,120),6),getdate()) = 0

–50、查询下月过生日的学生

select \* from student where datediff(mm,datename(yy,getdate()) + right(convert(varchar(10),sage,120),6),getdate()) = -1