**原文地址：**[sql语句多表查询(学生表/课程表/教师表/成绩表 )](http://blog.sina.com.cn/s/blog_78092e280100pk9d.html" \o "sql语句多表查询(学生表/课程表/教师表/成绩表 <wbr>)" \t "_blank)**作者：**[海豚湾孬蛋](http://blog.sina.com.cn/u/2013867560" \o "海豚湾孬蛋" \t "_blank)

问题及描述：  
--1.学生表  
Student(S#,Sname,Sage,Ssex) --S# 学生编号,Sname 学生姓名,Sage 出生年月,Ssex 学生性别  
--2.课程表  
Course(C#,Cname,T#) --C# --课程编号,Cname 课程名称,T# 教师编号  
--3.教师表  
Teacher(T#,Tname) --T# 教师编号,Tname 教师姓名  
--4.成绩表  
SC(S#,C#,score) --S# 学生编号,C# 课程编号,score 分数  
\*/  
--创建测试数据  
create table Student(S# varchar(10),Sname nvarchar(10),Sage datetime,Ssex nvarchar(10))  
insert into Student values('01' , N'赵雷' , '1990-01-01' , N'男')  
insert into Student values('02' , N'钱电' , '1990-12-21' , N'男')  
insert into Student values('03' , N'孙风' , '1990-05-20' , N'男')  
insert into Student values('04' , N'李云' , '1990-08-06' , N'男')  
insert into Student values('05' , N'周梅' , '1991-12-01' , N'女')  
insert into Student values('06' , N'吴兰' , '1992-03-01' , N'女')  
insert into Student values('07' , N'郑竹' , '1989-07-01' , N'女')  
insert into Student values('08' , N'王菊' , '1990-01-20' , N'女')  
create table Course(C# varchar(10),Cname nvarchar(10),T# varchar(10))  
insert into Course values('01' , N'语文' , '02')  
insert into Course values('02' , N'数学' , '01')  
insert into Course values('03' , N'英语' , '03')  
create table Teacher(T# varchar(10),Tname nvarchar(10))  
insert into Teacher values('01' , N'张三')  
insert into Teacher values('02' , N'李四')  
insert into Teacher values('03' , N'王五')  
create table SC(S# varchar(10),C# 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)  
go

--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 [课程'01'的分数],c.score [课程'02'的分数] from Student a , SC b , SC c  
where a.S# = b.S# and a.S# = c.S# and b.C# = '01' and c.C# = '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 [课程"01"的分数],c.score [课程"02"的分数] from Student a  
left join SC b on a.S# = b.S# and b.C# = '01'  
left join SC c on a.S# = c.S# and c.C# = '02'  
where b.score > isnull(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 [课程'01'的分数],c.score [课程'02'的分数] from Student a , SC b , SC c  
where a.S# = b.S# and a.S# = c.S# and b.C# = '01' and c.C# = '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 [课程"01"的分数],c.score [课程"02"的分数] from Student a  
left join SC b on a.S# = b.S# and b.C# = '01'  
left join SC c on a.S# = c.S# and c.C# = '02'  
where isnull(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.S# , a.Sname , cast(avg(b.score) as decimal(18,2)) avg\_score  
from Student a , sc b  
where a.S# = b.S#  
group by a.S# , a.Sname  
having cast(avg(b.score) as decimal(18,2)) >= 60  
order by a.S#

--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.S# , a.Sname , cast(avg(b.score) as decimal(18,2)) avg\_score  
from Student a , sc b  
where a.S# = b.S#  
group by a.S# , a.Sname  
having cast(avg(b.score) as decimal(18,2)) < 60  
order by a.S#  
--4.2、查询在sc表中不存在成绩的学生信息的SQL语句。  
select a.S# , a.Sname , isnull(cast(avg(b.score) as decimal(18,2)),0) avg\_score  
from Student a left join sc b  
on a.S# = b.S#  
group by a.S# , a.Sname  
having isnull(cast(avg(b.score) as decimal(18,2)),0) < 60  
order by a.S#

--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.S# [学生编号], a.Sname [学生姓名], count(b.C#) 选课总数, sum(score) [所有课程的总成绩]  
from Student a , SC b  
where a.S# = b.S#  
group by a.S#,a.Sname  
order by a.S#  
--5.2、查询所有(包括有成绩和无成绩)的SQL。  
select a.S# [学生编号], a.Sname [学生姓名], count(b.C#) 选课总数, sum(score) [所有课程的总成绩]  
from Student a left join SC b  
on a.S# = b.S#  
group by a.S#,a.Sname  
order by a.S#

--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 N'李%'  
--方法2  
select count(Tname) ["李"姓老师的数量] from Teacher where left(Tname,1) = N'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.S# = SC.S# and SC.C# = Course.C# and Course.T# = Teacher.T# and Teacher.Tname = N'张三'  
order by Student.S#

--8、查询没学过"张三"老师授课的同学的信息  
select m.\* from Student m where S# not in (select distinct SC.S# from SC , Course , Teacher where SC.C# = Course.C# and Course.T# = Teacher.T# and Teacher.Tname = N'张三') order by m.S#

--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.S# = SC.S# and SC.C# = '01' and exists (Select 1 from SC SC\_2 where SC\_2.S# = SC.S# and SC\_2.C# = '02') order by Student.S#  
--方法2  
select Student.\* from Student , SC where Student.S# = SC.S# and SC.C# = '02' and exists (Select 1 from SC SC\_2 where SC\_2.S# = SC.S# and SC\_2.C# = '01') order by Student.S#  
--方法3  
select m.\* from Student m where S# in  
(  
  select S# from  
  (  
    select distinct S# from SC where C# = '01'  
    union all  
    select distinct S# from SC where C# = '02'  
  ) t group by S# having count(1) = 2  
)  
order by m.S#

--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.S# = SC.S# and SC.C# = '01' and not exists (Select 1 from SC SC\_2 where SC\_2.S# = SC.S# and SC\_2.C# = '02') order by Student.S#  
--方法2  
select Student.\* from Student , SC where Student.S# = SC.S# and SC.C# = '01' and Student.S# not in (Select SC\_2.S# from SC SC\_2 where SC\_2.S# = SC.S# and SC\_2.C# = '02') order by Student.S#

--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.S# = SC.S#  
group by Student.S# , Student.Sname , Student.Sage , Student.Ssex having count(C#) < (select count(C#) from Course)  
--11.2  
select Student.\*  
from Student left join SC  
on Student.S# = SC.S#  
group by Student.S# , Student.Sname , Student.Sage , Student.Ssex having count(C#) < (select count(C#) 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.S# = SC.S# and SC.C# in (select C# from SC where S# = '01') and Student.S# <> '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 S# in  
(select distinct SC.S# from SC where S# <> '01' and SC.C# in (select distinct C# from SC where S# = '01')  
group by SC.S# having count(1) = (select count(1) from SC where S#='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.S# not in  
(select distinct sc.S# from sc , course , teacher where sc.C# = course.C# and course.T# = teacher.T# and teacher.tname = N'张三')  
order by student.S#

--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.S# , student.sname , cast(avg(score) as decimal(18,2)) avg\_score from student , sc  
where student.S# = SC.S# and student.S# in (select S# from SC where score < 60 group by S# having count(1) >= 2)  
group by student.S# , 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.C# , sc.score from student , sc  
where student.S# = SC.S# and sc.score < 60 and sc.C# = '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.S# 学生编号 , a.Sname 学生姓名 ,  
       max(case c.Cname when N'语文' then b.score else null end) [语文],  
       max(case c.Cname when N'数学' then b.score else null end) [数学],  
       max(case c.Cname when N'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.S# = b.S#  
left join Course c on b.C# = c.C#  
group by a.S# , a.Sname  
order by 平均分 desc  
--17.2 SQL 2000 动态  
declare @sql nvarchar(4000)  
set @sql = 'select a.S# ' + N'学生编号' + ' , a.Sname ' + N'学生姓名'  
select @sql = @sql + ',max(case c.Cname when N'''+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)) ' + N'平均分' + ' from Student a left join SC b on a.S# = b.S# left join Course c on b.C# = c.C#  
group by a.S# , a.Sname order by ' + N'平均分' + ' desc'  
exec(@sql)

--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.S# [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.S# = n.S#  
  group by m.S# , m.Sname  
) t2 where 平均成绩 > t1.平均成绩) + 1 from  
(  
  select m.S# [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.S# = n.S#  
  group by m.S# , m.Sname  
) t1  
order by px

select t1.\* , px = (select count(distinct 平均成绩) from  
(  
  select m.S# [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.S# = n.S#  
  group by m.S# , m.Sname  
) t2 where 平均成绩 >= t1.平均成绩) from  
(  
  select m.S# [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.S# = n.S#  
  group by m.S# , 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.S# [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.S# = n.S#  
  group by m.S# , m.Sname  
) t  
order by px

select t.\* , px = DENSE\_RANK() over(order by [平均成绩] desc) from  
(  
  select m.S# [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.S# = n.S#  
  group by m.S# , 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.C# , n.score from Student m, SC n where m.S# = n.S# and n.score in  
(select top 3 score from sc where C# = n.C# order by score desc) order by n.C# , 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 C# = t.C# and score >= t.score) from sc t) m where px between 1 and 3 order by m.c# , m.px  
--sql 2005用DENSE\_RANK实现  
select \* from (select t.\* , px = DENSE\_RANK() over(partition by c# order by score desc) from sc t) m where px between 1 and 3 order by m.C# , m.px

--26、查询每门课程被选修的学生数  
select c# , count(S#)[学生数] from sc group by C#

--27、查询出只有两门课程的全部学生的学号和姓名  
select Student.S# , Student.Sname  
from Student , SC  
where Student.S# = SC.S#  
group by Student.S# , Student.Sname  
having count(SC.C#) = 2  
order by Student.S#

--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.C# , m.Cname , cast(avg(n.score) as decimal(18,2)) avg\_score  
from Course m, SC n  
where m.C# = n.C#     
group by m.C# , m.Cname  
order by avg\_score desc, m.C# 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.S# , a.Sname , cast(avg(b.score) as decimal(18,2)) avg\_score  
from Student a , sc b  
where a.S# = b.S#  
group by a.S# , a.Sname  
having cast(avg(b.score) as decimal(18,2)) >= 85  
order by a.S#

--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.S# = Student.S# and SC.C# = Course.C# 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.C# , SC.score   
from Student, SC , Course  
where Student.S# = SC.S# and SC.C# = Course.C#  
order by Student.S# , SC.C#

--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.C# , SC.score   
from Student, SC , Course  
where Student.S# = SC.S# and SC.C# = Course.C# and SC.score >= 70  
order by Student.S# , SC.C#

--37、查询不及格的课程  
select Student.\* , Course.Cname , SC.C# , SC.score   
from Student, SC , Course  
where Student.S# = SC.S# and SC.C# = Course.C# and SC.score < 60  
order by Student.S# , SC.C#

--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.C# , SC.score   
from Student, SC , Course  
where Student.S# = SC.S# and SC.C# = Course.C# and SC.C# = '01' and SC.score >= 80  
order by Student.S# , SC.C#

--39、求每门课程的学生人数  
select Course.C# , Course.Cname , count(\*) [学生人数]  
from Course , SC  
where Course.C# = SC.C#  
group by  Course.C# , Course.Cname  
order by Course.C# , 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.C# , SC.score   
from Student, SC , Course , Teacher  
where Student.S# = SC.S# and SC.C# = Course.C# and Course.T# = Teacher.T# 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.C# , SC.score   
from Student, SC , Course , Teacher  
where Student.S# = SC.S# and SC.C# = Course.C# and Course.T# = Teacher.T# and Teacher.Tname = N'张三' and  
SC.score = (select max(SC.score) from SC , Course , Teacher where SC.C# = Course.C# and Course.T# = Teacher.T# 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 C# , score from SC group by C# , score having count(1) > 1) n  
where m.C#= n.C# and m.score = n.score order by m.C# , m.score , m.S#  
--方法2  
select m.\* from SC m where exists (select 1 from (select C# , score from SC group by C# , score having count(1) > 1) n  
where m.C#= n.C# and m.score = n.score) order by m.C# , m.score , m.S#

--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 C# = T.C# order by score desc) order by t.C# , 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.C# , Course.Cname , count(\*) [学生人数]  
from Course , SC  
where Course.C# = SC.C#  
group by  Course.C# , Course.Cname  
having count(\*) >= 5  
order by [学生人数] desc , Course.C#

--44、检索至少选修两门课程的学生学号  
select student.S# , student.Sname  
from student , SC  
where student.S# = SC.S#  
group by student.S# , student.Sname  
having count(1) >= 2  
order by student.S#

--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 S# in  
(select S# from sc group by S# 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.S# not in  
(  
  select distinct m.S# from  
  (  
    select S# , C# from student , course  
  ) m where not exists (select 1 from sc n where n.S# = m.S# and n.C# = m.C#)  
)  
--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.S# from  
  (  
    select S# , C# from student , course  
  ) m where not exists (select 1 from sc n where n.S# = m.S# and n.C# = m.C#)  
) k where k.S# = t.S#  
)

--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

drop table  Student,Course,Teacher,SC