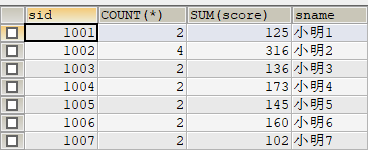
**演示部分**

1 查询所有同学的学号、姓名、选课数、总成绩

第一步：先查询所有同学的学号， 选课数， 总成绩  
​  
select sid, count(\*) , sum(score) from sc group by sid; // X 虚拟表  
​  
第二步：把上面的虚拟表X与学生表，进行联合查询,获取学生姓名  
​  
SELECT \* from student s, X where s.sid = X.sid;     
​  
最后，把X虚拟表以sql语句代替, 挑选只需要查询的字段  
​  
SELECT X.\* , s.sname FROM student s, (SELECT sid, COUNT(\*) , SUM(score) FROM sc GROUP BY sid) x WHERE s.sid = X.sid;     
​



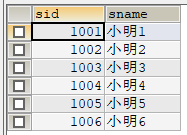
2 查询“html”课程比“css”课程成绩高的所有学生的学号

第一步：先查询html课程，对应的课程号  
​  
select cid from course where cname='html'; //具体的数据X  
​  
SELECT cid FROM course WHERE cname='css'; //具体的数据y   
​  
第二步：查询选取了html课程的学生的学生号和成绩  
​  
select \* from sc where cid = X;   
​  
初步整理：  
​  
select \* from sc where cid = (SELECT cid FROM course WHERE cname='html'); // 虚拟表A  
​  
同理查询选取了css课程的学生的学生号和成绩  
​  
SELECT \* FROM sc WHERE cid = (SELECT cid FROM course WHERE cname='css');   //// 虚拟表B  
​  
第三步：把虚拟表A和虚拟表B，进行联合查询，其中表连接的条件是：同一个人的成绩，即A.sid = B.sid  
​  
select sid from A , B where A.sid = B.sid and A.score >B.score;  
​  
最后整理：  
​  
SELECT A.sid FROM   
(SELECT \* FROM sc WHERE cid = (SELECT cid FROM course WHERE cname='html'))A , (SELECT \* FROM sc WHERE cid = (SELECT cid FROM course WHERE cname='css'))B   
​WHERE A.sid = B.sid AND A.score >B.score;  
​



3 查询学过“叶平”老师课的同学的学号、姓名；

第一步：根据“叶平”老师的名字，查询其老师编号  
​  
select tid from teacher where tname='叶平'; // 数据A  
​  
第二步：根据老师的tid, 查询其教授的课程的编号  
​  
select cid from course where tid = A;    
​  
初步整理:  
​  
SELECT cid FROM course WHERE tid = (SELECT tid FROM teacher WHERE tname='叶平');   // 注意：结果是一列多行，即：一个集合B  
​  
​  
​  
第三步：从成绩表中，根据课程编号，查询选修了课程的学生  
​  
select sid from sc where cid in B;  
​  
再次整理：（并增加去重）  
​  
SELECT sid FROM sc WHERE cid IN (SELECT cid FROM course WHERE tid = (SELECT tid FROM teacher WHERE tname='叶平')); // 结果，还是一个集合C  
​  
第四步：到学生表查找学生学号和姓名，要求其学号，必须在上面的集合C中  
​  
select sid, sname from student where sid in C;  
​  
最后整理：  
​  
SELECT sid, sname FROM student WHERE sid IN (SELECT sid FROM sc WHERE cid IN (SELECT cid FROM course WHERE tid = (SELECT tid FROM teacher WHERE tname='叶平')));  
​



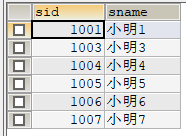
4 查询没学过“叶平”老师课的同学的学号、姓名；

目的：反向思维  
​  
先查询：学过“叶平”老师课的同学的学号，如果其结果是：集合X, 最后的结果，就是不在集合X中学生的学号  
​  
根据上题的过程，直接编写答案：  
​  
SELECT sid, sname FROM student WHERE sid not IN (SELECT sid FROM sc WHERE cid IN (SELECT cid FROM course WHERE tid = (SELECT tid FROM teacher WHERE tname='叶平')));  
​  
​  
​



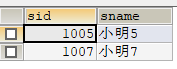
5 查询没有学全所有课的同学的学号、姓名

第一步：统计课程的总数  
​  
select count(\*) from course; // 数据A  
​  
第二步：成绩表中，按照学生分组，统计每个学生的选课数，只有选课数 小于 课程的总数，说明学全了  
​  
select sid from sc group by sid having count(\*) < A;  
​  
初步整理：  
​  
SELECT sid FROM sc GROUP BY sid HAVING COUNT() <(SELECT COUNT() FROM course); // 集合B  
​  
第三步：查询学生信息，其中学号必须在集合B中  
​  
select \* from student where sid in B;  
​  
​  
​  
最后整理：  
​  
SELECT sid, sname FROM student WHERE sid IN (SELECT sid FROM sc GROUP BY sid HAVING COUNT() <(SELECT COUNT() FROM course));  
​



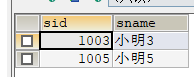
6 查询所有课程成绩小于80分的同学的学号、姓名；

目的：正确理解题意，只要有一门课程的成绩 大于等于80分，就不符合条件  
​  
第一步：把课程成绩大于等于80分的学生的学号，查询出来  
​  
select sid from sc where score >=80;  
​  
优化：去重  
​  
SELECT distinct sid FROM sc WHERE score >=80; // 集合A  
​  
​  
​  
第二步：查询满足条件的学生信息，其学号不能出现在集合A中  
​  
select sid, sname from student where sid not in A;   
​  
最后整理：  
​  
SELECT sid, sname FROM student WHERE sid NOT IN (SELECT DISTINCT sid FROM sc WHERE score >=80);   
​  
​  
​



7 查询和“1001”号的同学学习的课程完全相同的其他同学学号和姓名；

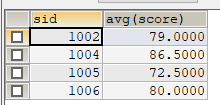
分析：  
​  
某个同学，学习的课程数量和1001的数量相同，并且1001没有学过的课程，他也没有学过，即符合要求。  
​  
​  
​  
第一步：查询1001号同学，选课的数量  
​  
select count(\*) from sc where sid=1001; // 数据A  
​  
第二步：查询选课数量和1001号相同的同学学号  
​  
select sid from sc group by sid having count(\*) = A;   
​  
初步整理：  
​  
SELECT sid FROM sc GROUP BY sid HAVING COUNT() = (SELECT COUNT() FROM sc WHERE sid=1001); // 第一个学生集合X  
​  
​  
​  
第三步：查询1001号学生，学过的课程集合  
​  
select cid from sc where sid =1001; // 集合B  
​  
第四步：查询学生集合，其学习的课程，有不在集合B的  
​  
select sid from sc where cid not in B;  
​  
再次整理：(需要去重)  
​  
SELECT distinct sid FROM sc WHERE cid NOT IN (SELECT cid FROM sc WHERE sid =1001); // 第二个学生集合Y  
​  
第五步：符合条件的学生，应该在集合X中，并且不在集合Y中，并且排除1001号学生本身  
​  
select sid from student where sid in X and sid not in y and sid !=1001;  
​  
最后整理：  
​  
SELECT sid,sname FROM student   
​  
WHERE sid IN (SELECT sid FROM sc GROUP BY sid HAVING COUNT() = (SELECT COUNT() FROM sc WHERE sid=1001))   
​  
AND sid NOT IN (SELECT DISTINCT sid FROM sc WHERE cid NOT IN (SELECT cid FROM sc WHERE sid =1001))  
​  
AND sid !=1001;



**1. 难度1级**

1 查询平均成绩大于70分的同学的学号和平均成绩；

select sid, avg(score) from sc group by sid having avg(score)>70;



2 查询姓“叶”的老师的个数；

\* select count(\*) from teacher where tname like "叶%";



3 查询学过“css”并且也学过编号“3001”课程的同学的学号、姓名；

第一步：先查询css的编程号

\* select cid from course where cname="css"; 数据x

第二步：查询学过css课程的学生学号

select sid, cid from sc where cid = X;

初步整理：

select sid, cid from sc where cid =(select cid from course where cname="css"); // 虚拟表A

同理：查询学过编号“3001”课程的同学的学号

select sid, cid from sc where cid =3001; // 虚拟表B

第三步：虚拟表A和B，联合查询，查询的条件是：sid相同，即符合查询条件的学生学号

select A.sid from A, B where A.sid = B.sid;

再次整理：

select A.sid from

(select sid, cid from sc where cid =(select cid from course where cname="css"))A, (select sid, cid from sc where cid =3001)B

where A.sid = B.sid; // 集合Z

第四步：查询学生表，符合条件的学生学号，必须在集合Z中

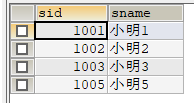
select sid ,sname from student where sid in Z;

最后整理：

select sid ,sname from student where sid in (select A.sid from

(select sid, cid from sc where cid =(select cid from course where cname="css"))A, (select sid, cid from sc where cid =3001)B

where A.sid = B.sid);



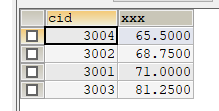
4 查询两门及以上不及格课程的同学的学号及其平均成绩

--先查询2门及以上不及格课程的同学的学号  
\* select sid from sc where score <60 group by sid having count(\*)>=2; //x  
--在查询所有学生的学号，及其平均成绩  
select sid, avg(score) from sc group by sid;  
​  
--综合在一起：查询两门及以上不及格课程的同学的学号及其平均成绩  
​  
select sid, avg(score) from sc where sid in (x) group by sid ;  
最后答案  
select sid, avg(score) from sc where sid in (select sid from sc where score <60 group by sid having count(\*)>=2) group by sid ;



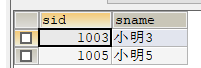
5 查询每门课程的平均成绩，结果按平均成绩升序排列，平均成绩相同时，按课程号降序排列

// 查询每门课的平均成绩  
 select avg(score) ,cid from sc group by cid;  
​  
 // 排序  
 select cid ,avg(score) as xxx   from sc group by cid order by xxx , cid desc;



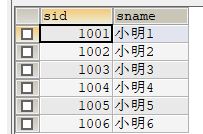
6 查询课程编号“3001”的成绩比课程编号“3002”课程低的所有同学的学号、姓名；

// 查询课程编号是：3002的成绩  
 select sid , score from sc where cid=3001; x  
​  
 // 查询课程编号是：3002的成绩  
 select sid , score from sc where cid=3002; y  
​  
 // 查询课程编号3001比3002低的学生的sid  
 select x.sid from x, y where x.sid = y.sid and x.score <y.score;  
​  
 整理：  
 select x.sid from (select sid , score from sc where cid=3001)x, (select sid , score from sc where cid=3002)y where x.sid = y.sid and x.score <y.score;  
   
 结果：  
 select sid, sname from student where sid in (select x.sid from (select sid , score from sc where cid=3001)x, (select sid , score from sc where cid=3002)y where x.sid = y.sid and x.score <y.score);



7 查询至少有一门课与学号为“1001”的同学所学相同的同学的学号和姓名；

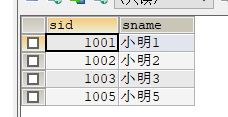
// 先查询1001号同学所需的所有课程的cid  
 select cid from sc where sid =1001; x  
​  
 // 查询至少有一门课程与其相同(去重)  
 select distinct sid from sc where cid in (x); y  
​  
 // 查询学生的学号和姓名  
 select sid ,sname from student where sid in (y);  
​  
 // 整理  
 select sid ,sname from student where sid in (select distinct sid from sc where cid in (select cid from sc where sid =1001));



**2. 难度2级**

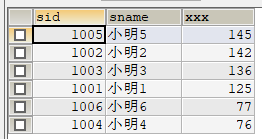
1 查询学过“叶平”老师所教的所有课的同学的学号、姓名；

​  
 // 先查询叶平老师教授的课程数量  
 select count(\*) from course where tid =(select tid from teacher where tname='叶平'); x   
​  
 // 查询叶平老师教授课程的cid(多个)  
 select cid from course where tid =(select tid from teacher where tname="叶平"); y  
   
 // 成绩表中，只查询，所有选课叶平老师的记录  
 select \* from sc where cid in (y); z  
   
 // 对于上面的z， 通过学号分组统计，当统计的次数= x 的时候，符合条件的sid  
 select sid from ()z group by sid having count(\*) = (x);  
​  
     整理：  
​  
 select sid from (select \* from sc where cid in (select cid from course where tid =(select tid from teacher where tname="叶平")) ) z group by z.sid having count(\*) = ( select count(\*) from course where tid =(select tid from teacher where tname='叶平') );  
   
 最终：查询学号和姓名  
 select sid, sname from student where sid in (select sid from (select \* from sc where cid in (select cid from course where tid =(select tid from teacher where tname="叶平")) ) z group by z.sid having count(\*) = ( select count(\*) from course where tid =(select tid from teacher where tname='叶平') ));



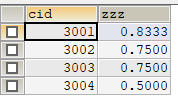
2 查询选修“叶平”老师所授课程的学生中，成绩最高的5名学生姓名及其成绩

// 查询叶平老师，教的所有课程的编号  
 select cid from course where tid = (select tid from teacher where tname="叶平"); x  
​  
 // 如何理解成绩最高的5名学生。 理解成：总成绩最高，即统计所有课程的总成绩，作为标准。  
 select sid , sum(score) xxx from sc where cid in (x) group by sid order by xxx desc;  
​  
 // 整理  
 select sid , sum(score) xxx from sc where cid in (select cid from course where tid = (select tid from teacher where tname="叶平")) group by sid order by xxx desc; y  
   
 // 学生姓名和成绩  
 select sid , sname, y.xxx from student st, ()y where st.sid= y.sid ;  
​  
 // 整理：  
 select st.sid , st.sname, y.xxx from student st, (select sid , sum(score) xxx from sc where cid in (select cid from course where tid = (select tid from teacher where tname="叶平")) group by sid order by xxx desc) y where st.sid= y.sid ;



3 查询所有的课程，并按照及格率的百分数从高到低顺序排列

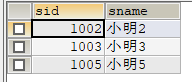
​  
 // 计算每门课程的及格率：人数大于等于60分的 / 该门课的总人数  
 // 先查询，每门课人数及格的人数  
 select cid, count(\*) xxx from sc where score >=60 group by cid; x  
​  
 // 每门课的总人数  
 select cid, count(\*) yyy from sc group by cid; y  
   
 // 获得每门课的及格率  
 select x.cid , x.xxx/y.yyy as zzz from ()x, ()y where x.cid =y.cid;  
​  
 // 整理  
 select x.cid , x.xxx/y.yyy as zzz from (select cid, count(\*) xxx from sc where score >=60 group by cid)x, (select cid, count(\*) yyy from sc group by cid)y where x.cid =y.cid ;  
​  
 // 从高到低排序  
 select x.cid , x.xxx/y.yyy as zzz from (select cid, count(\*) xxx from sc where score >=60 group by cid)x, (select cid, count(\*) yyy from sc group by cid)y where x.cid =y.cid order by zzz desc ;



**3. 难度3级**

查询至少学过学号为“1001”同学所有课的其他同学学号和姓名

// 某同学提供的解题思路，使用左外联接，非常漂亮  
​  
 // 先查询1001号学生学过的所有课程编号  
 select cid from sc where sid =1001;   x  
​  
 // 统计1001号学生选修课程的数量  
 select count(cid) from sc where sid =1001; y  
​  
 // 使用左外联接  
 select \* from sc left join   ()x on sc.cid = x.cid;   
​  
 // 整理  
 select sc.cid, sc.sid, x.cid as xxx from sc left join (select cid from sc where sid =1001)x on sc.cid = x.cid; z  
​  
 // 在y根据学号分组，统计x.cid的数量，如果，统计的次数等于1001号学生选择的课程数，即符合条件。  
​  
 select z.sid from () z group by z.sid having count(z.xxx) = (y);  
​  
 // 整理  
 select z.sid from (select sc.cid, sc.sid, x.cid as xxx from sc left join (select cid from sc where sid =1001)x on sc.cid = x.cid) z group by z.sid having count(z.xxx) = (select count(cid) from sc where sid =1001) ;   q  
   
 // 查询学号和姓名  
 select sid, sname from student where sid !=1001 and sid in (q);  
​  
 // 答案：  
 select sid, sname from student where sid !=1001 and sid in (select z.sid from (select sc.cid, sc.sid, x.cid as xxx from sc left join (select cid from sc where sid =1001)x on sc.cid = x.cid) z group by z.sid having count(z.xxx) = (select count(cid) from sc where sid =1001));



[附：有更好的答案：欢迎分享：邮箱：k116shixian@163.com](mailto:%E9%99%84%EF%BC%9A%E6%9C%89%E6%9B%B4%E5%A5%BD%E7%9A%84%E7%AD%94%E6%A1%88%EF%BC%9A%E6%AC%A2%E8%BF%8E%E5%88%86%E4%BA%AB%EF%BC%9A%E9%82%AE%E7%AE%B1%EF%BC%9Ak116shixian@163.com)