#查询student表中重名的学生，结果包含id和name，按name,id升序

#需要查询某一列重复的行，一般通过group by(有重复的列)然后取count>1的值。

SELECT id, `name` FROM student WHERE `name` IN(

SELECT `name` FROM student GROUP BY(`name`) HAVING(COUNT(\*) > 1)

) ORDER BY `name`;

#在student\_course表中查询平均分不及格的学生，列出学生id和平均分

#where子句中不能用聚集函数作为条件表达式，但是having短语可以，where和having的区别在于对用对象不同，where作用于记录，having作用于组。

SELECT sid, AVG(score) AS avg\_score FROM student\_course GROUP BY sid HAVING(avg\_score > 60);

#在student\_course表中查询每门课成绩都不低于80的学生id

SELECT DISTINCT sid FROM student\_course WHERE sid not in(

SELECT sid FROM student\_course WHERE score < 80);

#查询每个学生的总成绩，结果列出学生姓名和总成绩 如果使用下面的sql会过滤掉没有成绩的人

SELECT `name`, SUM(score) total FROM student, student\_course WHERE

student.id = student\_course.sid GROUP BY sid;

#更保险的做法应该是使用 左外连接

SELECT id, `name`, SUM(score) total FROM student LEFT JOIN student\_course ON

student.id = student\_course.sid GROUP BY sid;

#总成绩最高的学生，结果列出学生id和总成绩

SELECT sid, SUM(score) AS sum\_score FROM student\_course GROUP BY sid ORDER BY sum\_score DESC LIMIT 1;

#在student\_course表查询课程1成绩第2高的学生，如果第2高的不止一个则列出所有的学生

#这是个查询 第N大数 的问题。 我们先查出第2高的成绩：

select \* from student\_course

where cid=1 and score = (

select score from student\_course where cid = 1 group by score order by score desc limit 1,1

);

#在student\_course表查询各科成绩最高的学生，结果列出学生id、课程id和对应的成绩

select \* from student\_course as x where score>=

(select max(score) from student\_course as y where cid=x.cid);

#在student\_course表中查询每门课的前2名，结果按课程id升序，同一课程按成绩降序 这个问题也就是取每组的前N条纪录

select \* from student\_course x where

2>(select count(\*) from student\_course y where y.cid=x.cid and y.score>x.score)

order by cid,score desc;

#一个叫team的表，里面只有一个字段name,一共有4条纪录，分别是a,b,c,d,对应四个球队，两两进行比赛，用一条sql语句显示所有可能的比赛组合

select a.name, b.name

from team a, team b

where a.name < b.name