Mysql练习题

CREATE TABLE Class(

calss\_id int(10) PRIMARY key not null auto\_increment,

calss\_name VARCHAR(64) not null

)

**Class表的定义**

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **字段名** | **字段描述** | **数据类型** | **主键** | **外键** | **非空** | **唯一** | **自增** |
| class\_id | 编号 | INT(10) | 是 | 否 | 是 | 是 | 是 |
| class\_name | 班级名称 | VARCHAR(64) | 否 | 否 | 是 | 否 | 否 |

INSERT INTO `class` VALUES ('1', '三年二班'), ('2', '三年三班'), ('3', '一年二班'), ('4', '二年九班');

CREATE TABLE subject(

subject\_id int(10) primary key not null auto\_increment,

subject\_name VARCHAR(64) not null,

teacher\_id int(10)

)

**Subject表的定义**

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **字段名** | **字段描述** | **数据类型** | **主键** | **外键** | **非空** | **唯一** | **自增** |
| subject\_id | 编号 | INT(10) | 是 | 否 | 是 | 是 | 是 |
| subject\_name | 班级名称 | VARCHAR(64) | 否 | 否 | 是 | 否 | 否 |
| teacher\_id | 教师id | INT(10) | 否 | 否 | 否 | 否 | 否 |

INSERT INTO `course` VALUES ('1', '生物', '1'), ('2', '物理', '2'), ('3', '体育', '3'), ('4', '美术', '2');

CREATE TABLE score(

score\_id int(10) primary key not null auto\_increment,

subject\_id int(10) not null,

student\_id int(10),

score int(10)

)

**Score表的定义**

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **字段名** | **字段描述** | **数据类型** | **主键** | **外键** | **非空** | **唯一** | **自增** |
| score\_id | 编号 | INT(10) | 是 | 否 | 是 | 是 | 是 |
| subject\_id | 课程id | INT(10) | 否 | 否 | 是 | 否 | 否 |
| student\_id | 学生id | INT(10) | 否 | 否 | 否 | 否 | 否 |
| score | 分数 | INT(10) | 否 | 否 | 否 | 否 | 否 |

INSERT INTO `score` VALUES ('1', '1', '1', '10'), ('2', '1', '2', '9'), ('5', '1', '4', '66'), ('6', '2', '1', '8'), ('8', '2', '3', '68'), ('9', '2', '4', '99'), ('10', '3', '1', '77'), ('11', '3', '2', '66'), ('12', '3', '3', '87'), ('13', '3', '4', '99'), ('14', '4', '1', '79'), ('15', '4', '2', '11'), ('16', '4', '3', '67'), ('17', '4', '4', '100'), ('18', '5', '1', '79'), ('19', '5', '2', '11'), ('20', '5', '3', '67'), ('21', '5', '4', '100'), ('22', '6', '1', '9'), ('23', '6', '2', '100'), ('24', '6', '3', '67'), ('25', '6', '4', '100'), ('26', '7', '1', '9'), ('27', '7', '2', '100'), ('28', '7', '3', '67'), ('29', '7', '4', '88'), ('30', '8', '1', '9'), ('31', '8', '2', '100'), ('32', '8', '3', '67'), ('33', '8', '4', '88'), ('34', '9', '1', '91'), ('35', '9', '2', '88'), ('36', '9', '3', '67'), ('37', '9', '4', '22'), ('38', '10', '1', '90'), ('39', '10', '2', '77'), ('40', '10', '3', '43'), ('41', '10', '4', '87'), ('42', '11', '1', '90'), ('43', '11', '2', '77'), ('44', '11', '3', '43'), ('45', '11', '4', '87'), ('46', '12', '1', '90'), ('47', '12', '2', '77'), ('48', '12', '3', '43'), ('49', '12', '4', '87'), ('52', '13', '3', '87');

CREATE TABLE student(

student\_id int(10) PRIMARY key not null auto\_increment,

sex VARCHAR(64) not null,

class\_id int(10),

student\_name VARCHAR(64)

)

**Student表的定义**

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **字段名** | **字段描述** | **数据类型** | **主键** | **外键** | **非空** | **唯一** | **自增** |
| student\_id | 编号 | INT(10) | 是 | 否 | 是 | 是 | 是 |
| sex | 性别 | VARCHAR(64) | 否 | 否 | 是 | 否 | 否 |
| class\_id | 班级id | INT(10) | 否 | 否 | 否 | 否 | 否 |
| student\_name | 学生姓名 | VARCHAR(64) | 否 | 否 | 否 | 否 | 否 |

INSERT INTO `student` VALUES ('1', '男', '1', '理解'), ('2', '女', '1', '钢蛋'), ('3', '男', '1', '张三'), ('4', '男', '1', '张一'), ('5', '女', '1', '张二'), ('6', '男', '1', '张四'), ('7', '女', '2', '铁锤'), ('8', '男', '2', '李三'), ('9', '男', '2', '李一'), ('10', '女', '2', '李二'), ('11', '男', '2', '李四'), ('12', '女', '3', '如花'), ('13', '男', '3', '刘三'), ('14', '男', '3', '刘一'), ('15', '女', '3', '刘二'), ('16', '男', '3', '刘四');

CREATE TABLE teacher(

teacher\_id int(10) PRIMARY key not null auto\_increment,

teacher\_name VARCHAR(64) not null

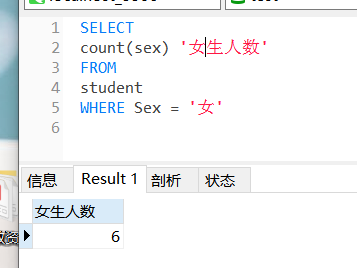
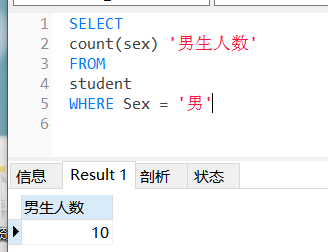
)

**Teacher表的定义**

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **字段名** | **字段描述** | **数据类型** | **主键** | **外键** | **非空** | **唯一** | **自增** |
| teacher\_id | 编号 | INT(10) | 是 | 否 | 是 | 是 | 是 |
| teacher\_name | 姓名 | VARCHAR(64) | 否 | 否 | 是 | 否 | 否 |

INSERT INTO `teacher` VALUES ('1', '张磊老师'), ('2', '李平老师'), ('3', '刘海燕老师'), ('4', '朱云海老师'), ('5', '李杰老师');

1. 查询男生、女生的人数；



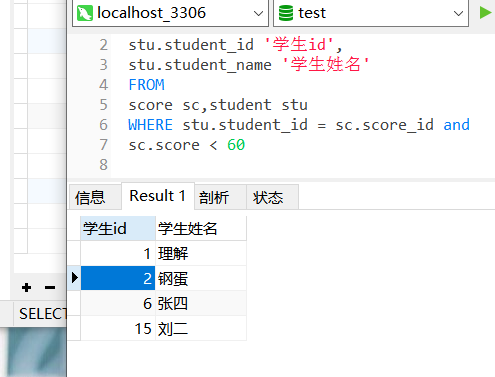
1. 查询姓“张”的学生名单；



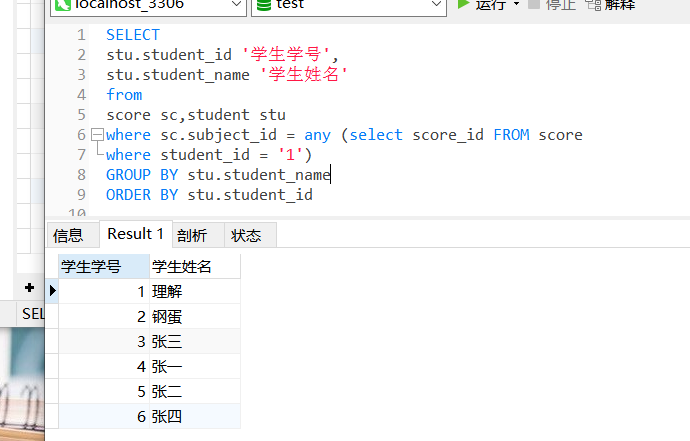
1. 课程平均分从高到低显示



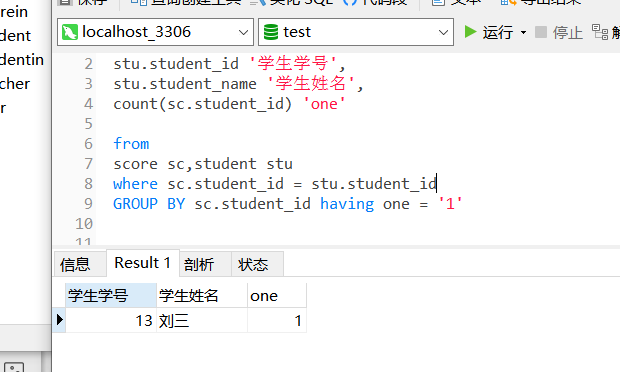
1. 查询有课程成绩小于60分的同学的学号、姓名；



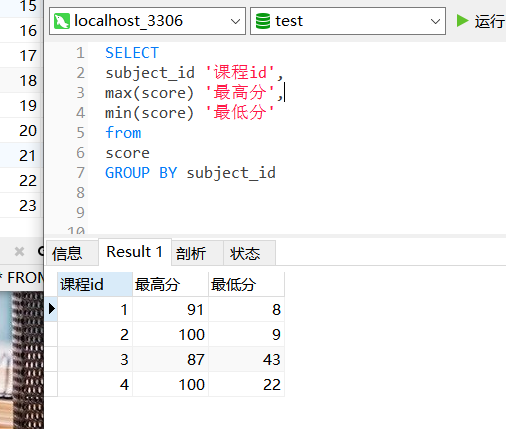
1. 查询至少有一门课与学号为1的同学所学课程相同的同学的学号和姓名；



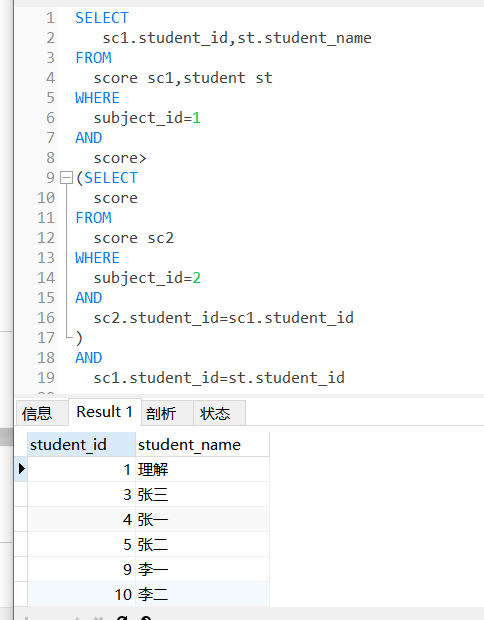
1. 查询出只选修了一门课程的全部学生的学号和姓名；



1. 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；



1. 查询课程编号“2”的成绩比课程编号“1”课程低的所有同学的学号、姓名；



1. 查询“生物”课程比“物理”课程成绩高的所有学生的学号；

SELECT

sc1.student\_id,stu.student\_name

FROM

score sc1,student stu

WHERE

subject\_id=1

AND

score>

(SELECT

score

FROM

score sc2

WHERE

subject\_id=2

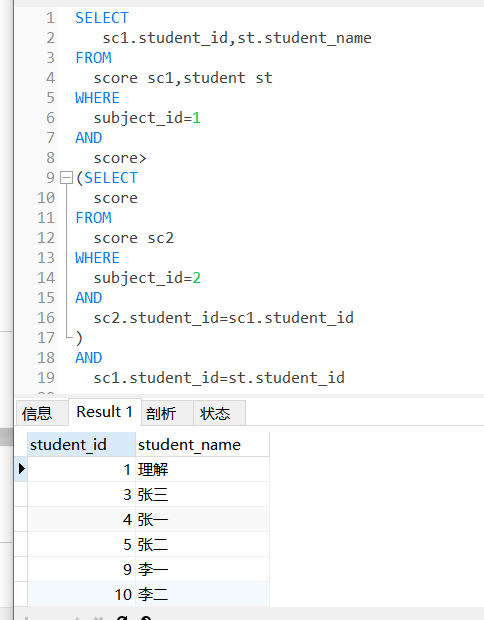
AND

sc2.student\_id=sc1.student\_id

)

AND

sc1.student\_id=stu.student\_id



1. 查询平均成绩大于60分的同学的学号和平均成绩;

select

sc.student\_id '学号',

round(avg(sc.score)) '平均成绩'

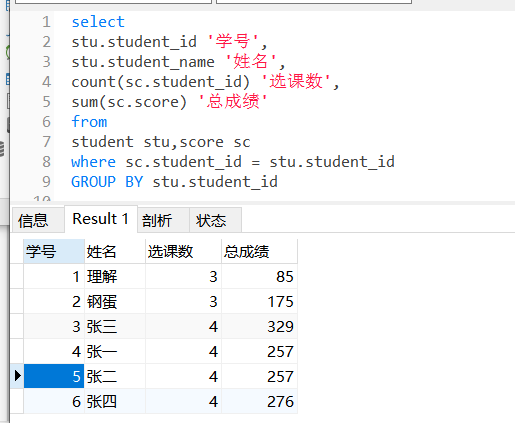
from

student stu,score sc

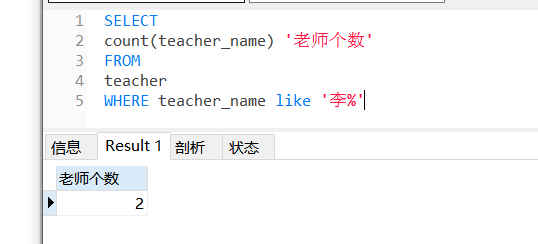
GROUP BY sc.student\_id having avg(sc.score) > 60



1. 查询所有同学的学号、姓名、选课数、总成绩；



1. 查询姓“李”的老师的个数；



1. 查询没学过“张磊老师”课的同学的学号、姓名；

select

stu.student\_id '学号',

stu.student\_name '学生姓名'

from

score sc,student stu

where stu.student\_id = sc.student\_id and stu.student\_id not in

(

select

student\_id

FROM score

where

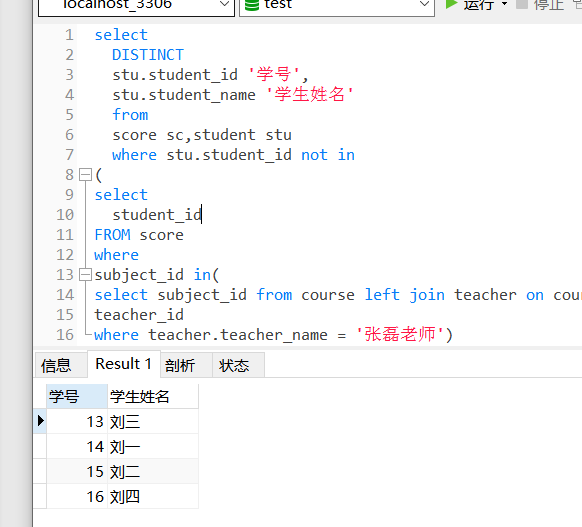
subject\_id in(

select subject\_id from course left join teacher on course.teacher\_id = teacher.

teacher\_id

where teacher.teacher\_name = '张磊老师')

)



1. 查询学过“1”并且也学过编号“2”课程的同学的学号、姓名；



select

student\_id '学号',

student\_name '姓名'

from

student

where student\_id in(

select

t1.student\_id

FROM

(select sc.student\_id from score sc where sc.subject\_id = '1') t1,

(select sc1.student\_id from score sc1 where sc1.subject\_id = '2') t2

where

t1.student\_id = t2.student\_id

)

1. 查询学过“李平老师”所教的所有课的同学的学号、姓名；

select

student\_id '学号',

student\_name '姓名'

from

student

where student\_id in (

select

student\_id

from

score

where subject\_id in (

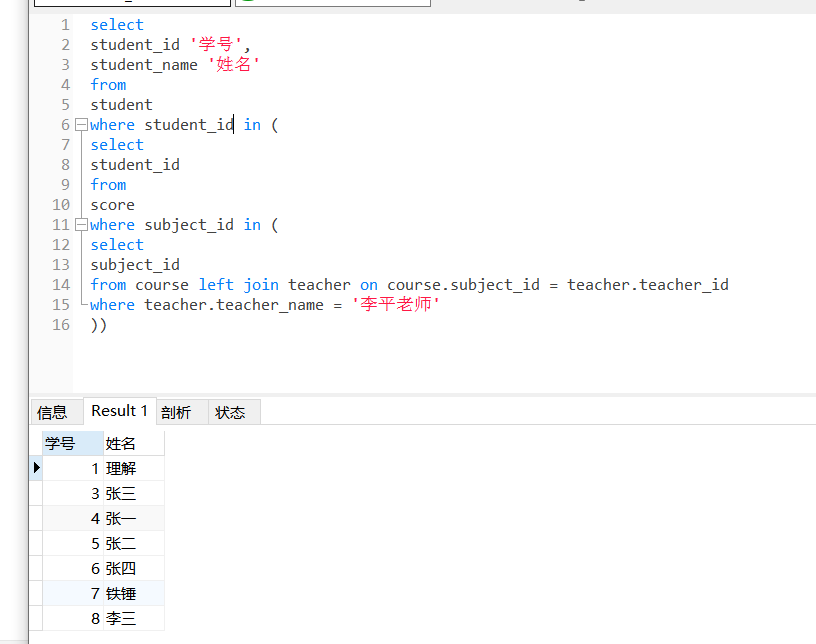
select

subject\_id

from course left join teacher on course.subject\_id = teacher.teacher\_id

where teacher.teacher\_name = '李平老师'

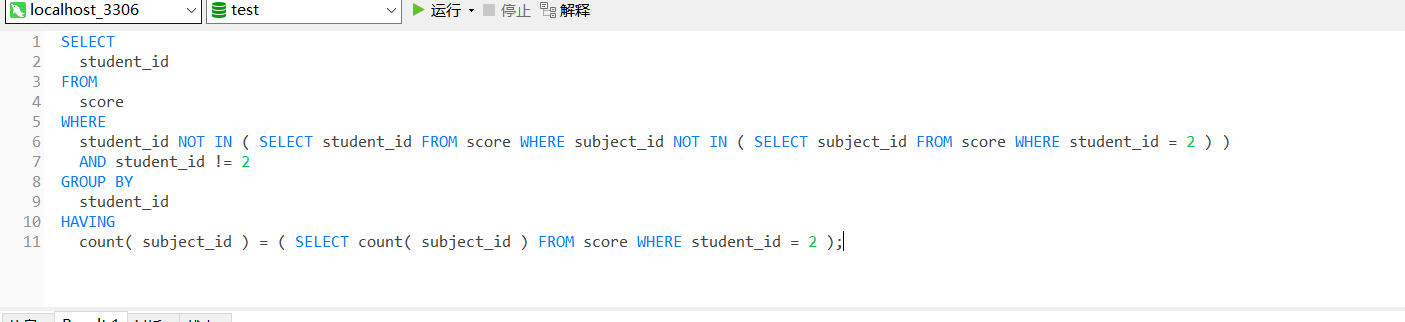
))



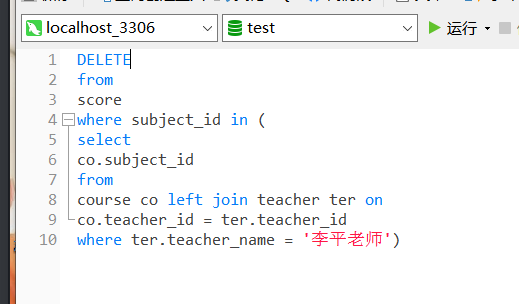
1. 查询没有学全所有课的同学的学号、姓名；



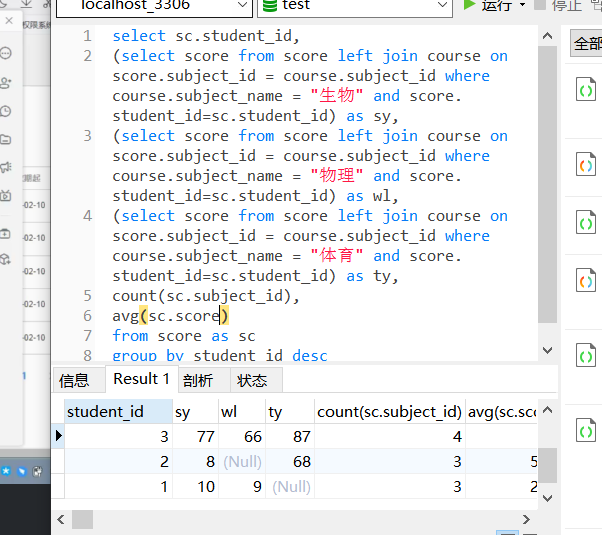
1. 查询和“002”号的同学学习的课程完全相同的其他同学学号和姓名；



1. 删除学习“叶平”老师课的SC表记录；



1. 按平均成绩从低到高显示所有学生的“语文”、“数学”、“英语”三门的课程成绩，按如下形式显示： 学生ID,语文,数学,英语,有效课程数,有效平均分；



select sc.student\_id,

(select score from score left join course on score.subject\_id = course.subject\_id where course.subject\_name = "生物" and score.student\_id=sc.student\_id) as sy,

(select score from score left join course on score.subject\_id = course.subject\_id where course.subject\_name = "物理" and score.student\_id=sc.student\_id) as wl,

(select score from score left join course on score.subject\_id = course.subject\_id where course.subject\_name = "体育" and score.student\_id=sc.student\_id) as ty,

count(sc.subject\_id),

avg(sc.score)

from score as sc

group by student\_id desc

21.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

subject\_id '课程ID',

max(score) '最高分',

min(score) '最低分'

from

score

GROUP BY subject\_id

1. 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

subject\_id '课程ID',

round(avg(score)) '平均成绩',

sum(case when sc.score>=60 then 1 else 0 end)/count(1)\*100 as 及格率

from

score sc

GROUP BY subject\_id

ORDER BY round(avg(score)) asc,sum(case when sc.score>=60 then 1 else 0 end)/count(1) desc

Case when then

1. 查询各科成绩前三名的记录:(不考虑成绩并列情况)

SELECT

sc\_1.subject\_id,

sc\_1.score,

count(DISTINCT sc\_2.score) AS ser

FROM

score AS sc\_1

INNER JOIN score AS sc\_2 ON sc\_1.subject\_id = sc\_2.subject\_id

AND sc\_1.score <= sc\_2.score

GROUP BY

sc\_1.subject\_id,

sc\_1.score

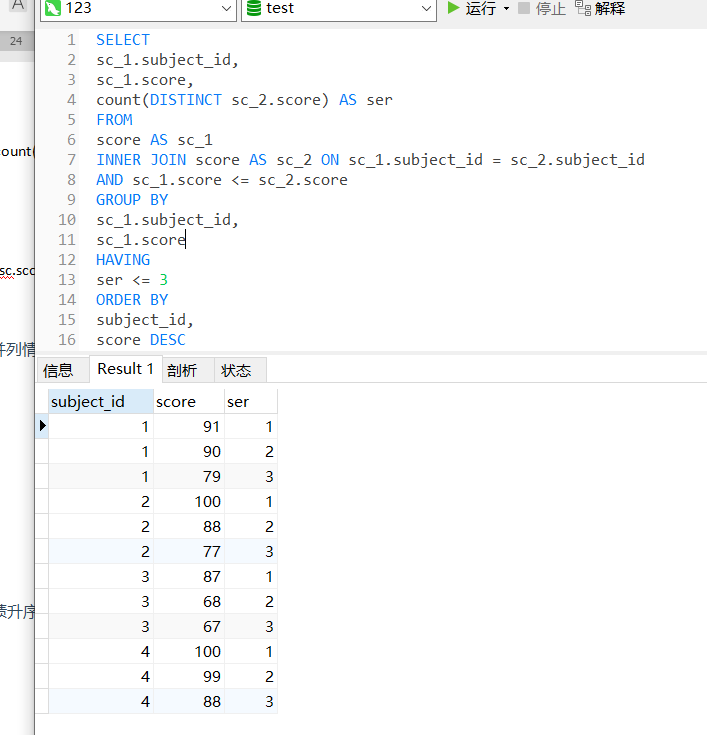
HAVING

ser <= 3

ORDER BY

subject\_id,

score DESC



24.查询每门课程被选修的学生数；

select

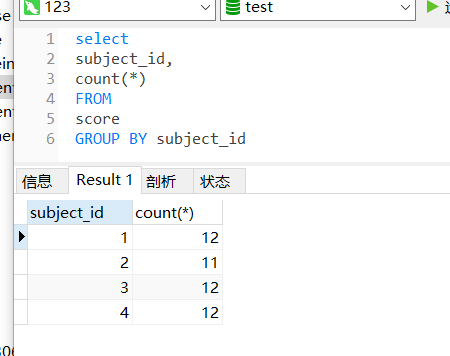
subject\_id,

count(\*)

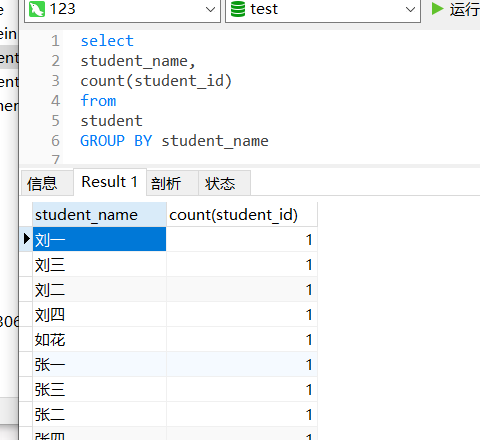
FROM

score

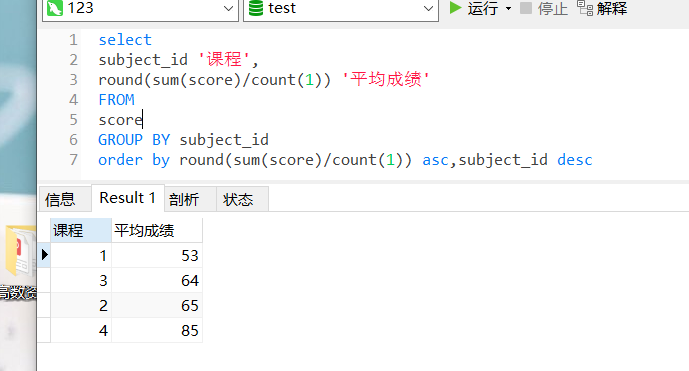
GROUP BY subject\_id



25.查询同名同姓学生名单，并统计同名人数；



26.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

subject\_id '课程',

round(sum(score)/count(1)) '平均成绩'

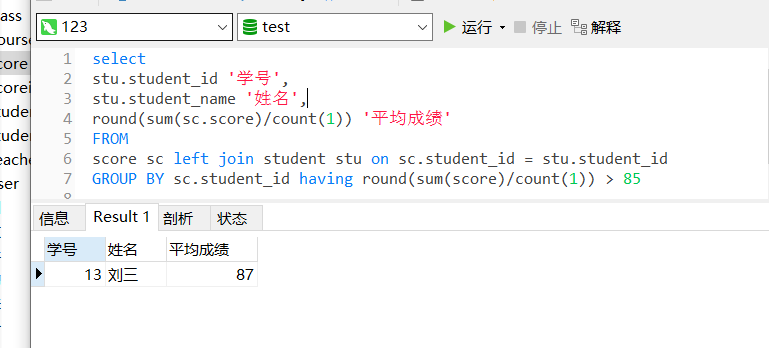
FROM

score

GROUP BY subject\_id

order by round(sum(score)/count(1)) asc,subject\_id desc

27.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

stu.student\_id '学号',

stu.student\_name '姓名',

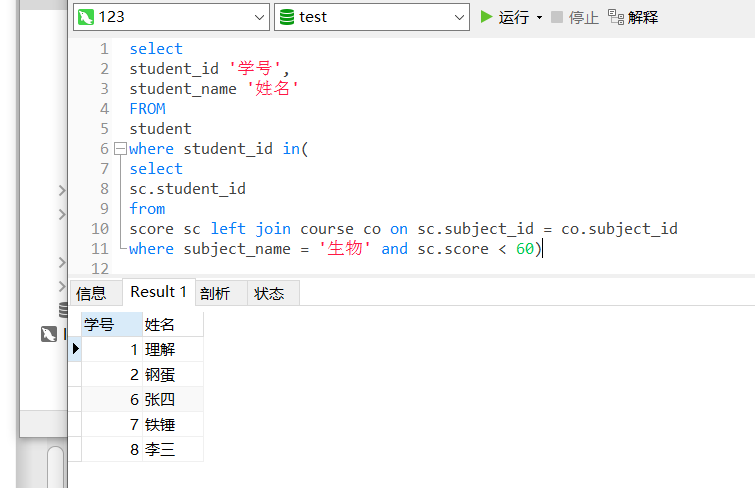
round(sum(sc.score)/count(1)) '平均成绩'

FROM

score sc left join student stu on sc.student\_id = stu.student\_id

GROUP BY sc.student\_id having round(sum(score)/count(1)) > 85

28.查询课程名称为“生物”，且分数低于60的学生姓名和分数；



select

student\_id '学号',

student\_name '姓名'

FROM

student

where student\_id in(

select

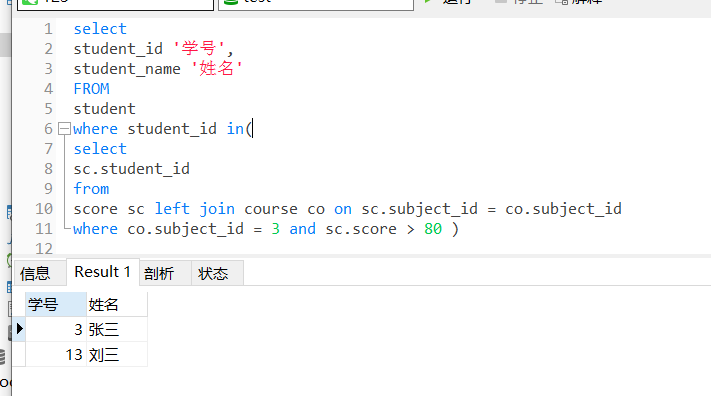
sc.student\_id

from

score sc left join course co on sc.subject\_id = co.subject\_id

where subject\_name = '生物' and sc.score < 60)

29.查询课程编号为003且课程成绩在80分以上的学生的学号和姓名；



select

student\_id '学号',

student\_name '姓名'

FROM

student

where student\_id in(

select

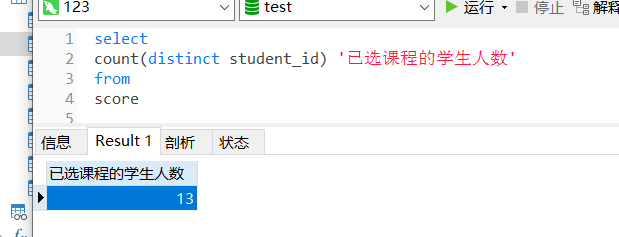
sc.student\_id

from

score sc left join course co on sc.subject\_id = co.subject\_id

where co.subject\_id = 3 and sc.score > 80 )

30.求选了课程的学生人数



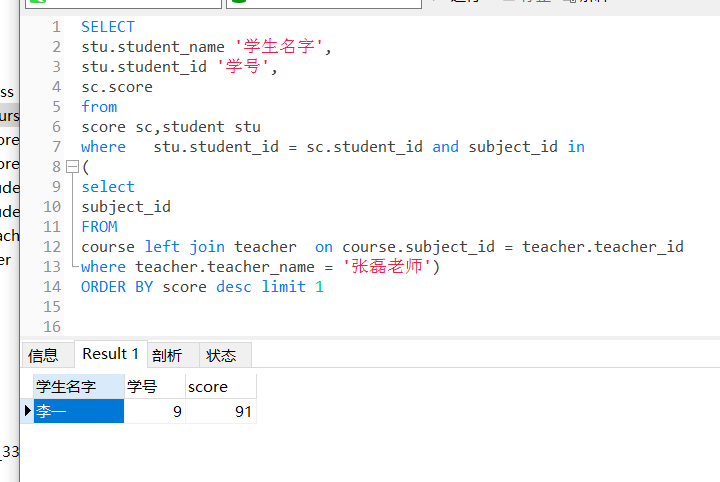
select

count(distinct student\_id) '已选课程的学生人数'

from

score

31.查询选修“张磊”老师所授课程的学生中，成绩最高的学生姓名及其成绩；



SELECT

stu.student\_name '学生名字',

stu.student\_id '学号',

sc.score

from

score sc,student stu

where stu.student\_id = sc.student\_id and subject\_id in

(

select

subject\_id

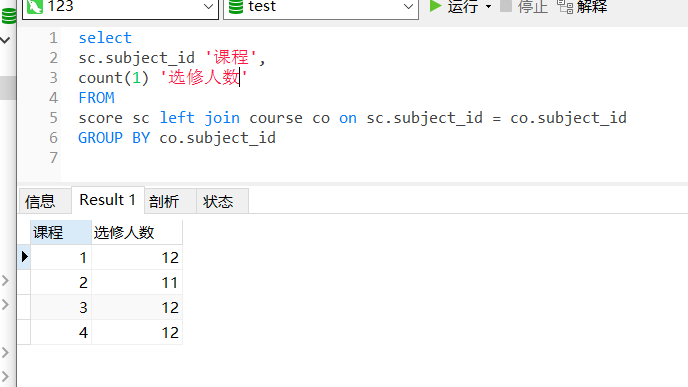
FROM

course left join teacher on course.subject\_id = teacher.teacher\_id

where teacher.teacher\_name = '张磊老师')

ORDER BY score desc limit 1

32.查询各个课程及相应的选修人数；



select

sc.subject\_id '课程',

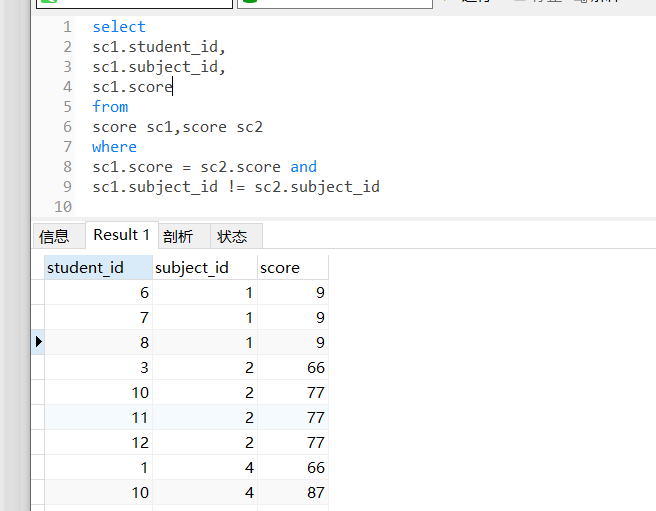
count(1) '选修人数'

FROM

score sc left join course co on sc.subject\_id = co.subject\_id

GROUP BY co.subject\_id

33.查询不同课程但成绩相同的学生的学号、课程号、学生成绩；



select

sc1.student\_id,

sc1.subject\_id,

sc1.score

from

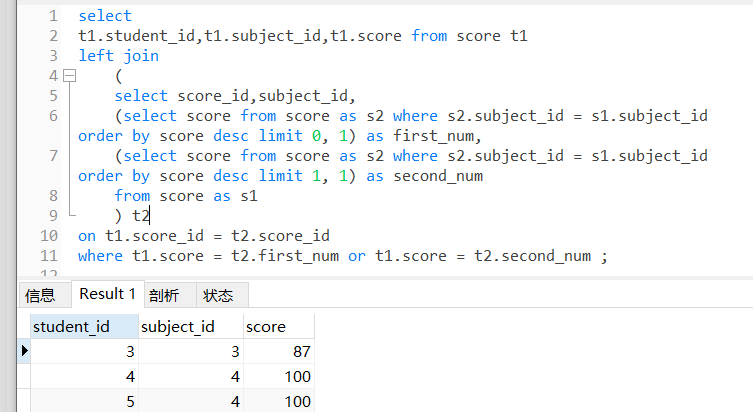
score sc1,score sc2

where

sc1.score = sc2.score and

sc1.subject\_id != sc2.subject\_id

34.查询每门课程成绩最好的前两名；



select

t1.student\_id,t1.subject\_id,t1.score from score t1

left join

(

select score\_id,subject\_id,

(select score from score as s2 where s2.subject\_id = s1.subject\_id order by score desc limit 0, 1) as first\_num,

(select score from score as s2 where s2.subject\_id = s1.subject\_id order by score desc limit 1, 1) as second\_num

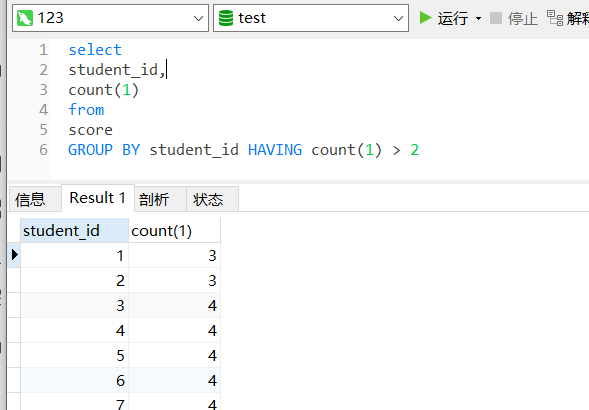
from score as s1

) t2

on t1.score\_id = t2.score\_id

where t1.score = t2.first\_num or t1.score = t2.second\_num ;

35.检索至少选修两门课程的学生学号；



select

student\_id,

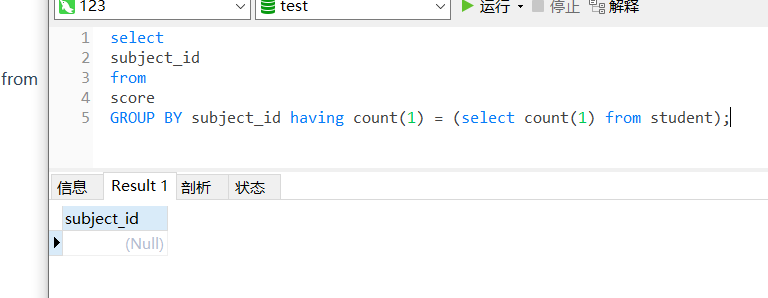
count(1)

from

score

GROUP BY student\_id HAVING count(1) > 2

36.查询全部学生都选修的课程的课程号和课程名；



select

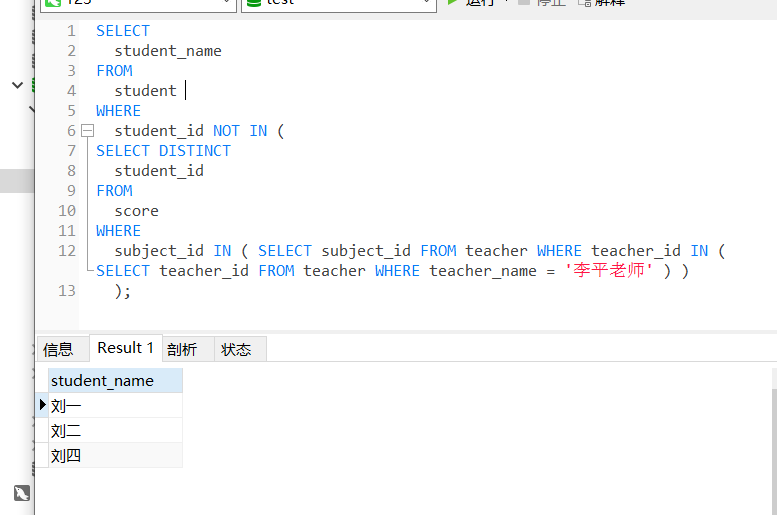
subject\_id

from

score

GROUP BY subject\_id having count(1) = (select count(1) from student);

1. 查询没学过“李平”老师讲授的任一门课程的学生姓名；



SELECT

student\_name

FROM

student

WHERE

student\_id NOT IN (

SELECT DISTINCT

student\_id

FROM

score

WHERE

subject\_id IN ( SELECT subject\_id FROM teacher WHERE teacher\_id IN ( SELECT teacher\_id FROM teacher WHERE teacher\_name = '李平老师' ) )

);

38.查询两门以上不及格课程的同学的学号及其平均成绩；



select

student\_id '学号',

round(avg(score)) '平均成绩'

from score

where student\_id in (

select

student\_id

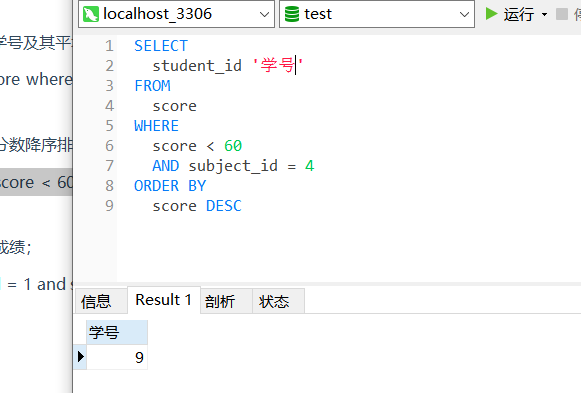
from

score

where score < 60

GROUP BY student\_id having count(1) > 1)

39.检索“004”课程分数小于60，按分数降序排列的同学学号；



40.删除“002”同学的“001”课程的成绩；

