Mysql练习题

**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');

**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');

**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');

**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');

**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；

select

class\_id '学号',

COUNT(sex) '人数'

from student

group by sex



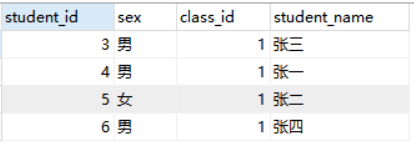
1. 查询姓“张”的学生名单；

select \*

from student

where

student\_name like '张%'



1. 课程平均分从高到低显示

select

subject\_id '课程编号',

AVG(score) '平均分数'

from

score

GROUP BY subject\_id

ORDER BY AVG(score) desc



1. 查询有课程成绩小于60分的同学的学号、姓名；

select

stu.student\_id '学号',

stu.student\_name '姓名',

sc.score '分数'

from

student stu ,score sc

where

sc.score < 60 and

stu.student\_id = sc.student\_id



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；

select

stu.student\_id '学号',

stu.student\_name '学生姓名'

from

(select DISTINCT student\_id from score

where

subject\_id in(select subject\_id from score where student\_id = '1')) a,

student stu

where

stu.student\_id = a.student\_id



1. 查询出只选修了一门课程的全部学生的学号和姓名；

select

stu.student\_name '学生姓名',

stu.student\_id '学号'

from

student stu,(select a.student\_id from (select student\_id,COUNT(subject\_id) 'number' from score GROUP BY student\_id) a where a.number = 1) b

where

stu.student\_id = b.student\_id



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；

select

c.subject\_id '课程',

c.max '最高分',

d.min '最低分'

from

(select subject\_id,MAX(score) 'max' from score GROUP BY subject\_id) c,

(select subject\_id,MIN(score) 'min' from score GROUP BY subject\_id) d

where

c.subject\_id = d.subject\_id



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；

SELECT

student\_id '学号',

student\_name '学生姓名'

FROM

student

WHERE

student\_id IN (

SELECT

sca.student\_id

FROM

(

SELECT

student\_id,

subject\_id,

score

FROM

score

WHERE

subject\_id = '1'

) sca

INNER JOIN (

SELECT

student\_id,

subject\_id,

score

FROM

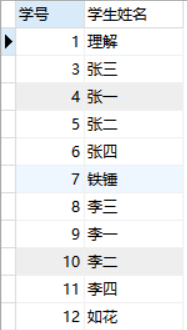
score

WHERE

subject\_id = '2'

) scb ON sca.student\_id = scb.student\_id

)



1. 查询“生物”课程比“物理”课程成绩高的所有学生的学号；

SELECT

sca.student\_id

FROM

(

select

sc.student\_id,

sub.subject\_name,

sc.score

from

score sc,subject sub

where

sc.subject\_id = '1' and

sub.subject\_id = sc.subject\_id

) sca

INNER JOIN (

select

sc.student\_id,

sub.subject\_name,

sc.score

from

score sc,subject sub

where

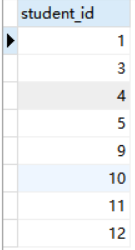
sc.subject\_id = '2' and

sub.subject\_id = sc.subject\_id

) scb ON sca.student\_id = scb.student\_id

where

sca.score > scb.score



1. 查询平均成绩大于60分的同学的学号和平均成绩;

select

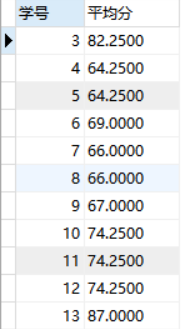
student\_id '学号',

AVG(score) '平均分'

from

score

GROUP BY student\_id HAVING AVG(score)>60



1. 查询所有同学的学号、姓名、选课数、总成绩；

select

stu.student\_id '学号',

stu.student\_name '姓名',

sce.count '选课数',

sce.sum '总成绩'

from

student stu,

(select

student\_id,

sum(score) 'sum',

count(subject\_id) 'count'

from

score

GROUP BY student\_id ) sce

where

sce.student\_id = stu.student\_id



1. 查询姓“李”的老师的个数；

select

COUNT(teacher\_name) '个数'

from

teacher

where

teacher\_name like '李%'



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 '姓名'

from

student stu

RIGHT JOIN (select

student\_id,

subject\_id

from

score

GROUP BY student\_id having subject\_id not in(select

subject\_id

from

`subject`

where

teacher\_id in(SELECT teacher\_id FROM teacher where teacher\_name = '张磊老师'))) suba on stu.student\_id = suba.student\_id



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 sc1.student\_id 'id'

FROM

(SELECT

\*

from

score

where

subject\_id = '1') sc1

INNER JOIN (SELECT

\*

from

score

where

subject\_id = '2') sc2 on sc1.student\_id = sc2.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

stu.student\_id '学号',

stu.student\_name '姓名'

from

student stu , `subject` sub, score sc , teacher tea

where

stu.student\_id = sc.student\_id

and sc.subject\_id = sub.subject\_id

and sub.teacher\_id = tea.teacher\_id

and tea.teacher\_name = '李平老师'

GROUP BY stu.student\_id

having count(\*) = 2



1. 查询没有学全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

(

SELECT

count(subject\_id) c\_subject\_id,

student\_id

FROM

score

GROUP BY

student\_id

) t1

WHERE

t1.c\_subject\_id < (

SELECT

count(subject\_id)

FROM

`subject`

)

)



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；

SELECT

stu.student\_id '学号',

stu.student\_name '姓名'

FROM

Student stu,

Score sc

WHERE

stu.student\_id = sc.student\_id

AND stu.student\_id NOT IN (

SELECT

student\_id

FROM

Score

WHERE

subject\_id NOT IN (

SELECT

subject\_id

FROM

Score

WHERE

student\_id = 2

)

)

AND stu.student\_id != 2

GROUP BY

stu.student\_id

HAVING

COUNT(\*) = (

SELECT

COUNT(\*)

FROM

Score

WHERE

student\_id = 2

)



1. 删除学习“李平”老师课的SC表记录；

DELETE

FROM

Score1

where

score\_id in(select

score\_id

from

score sc,

(select

subject\_id

from

subject

where

teacher\_id in (SELECT

teacher\_id

FROM

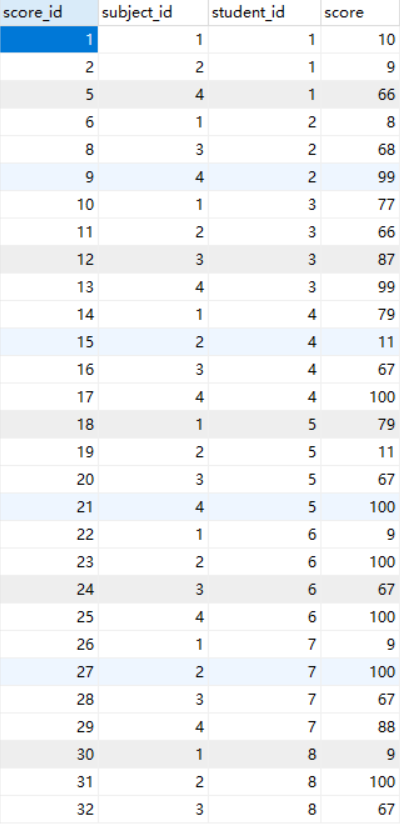
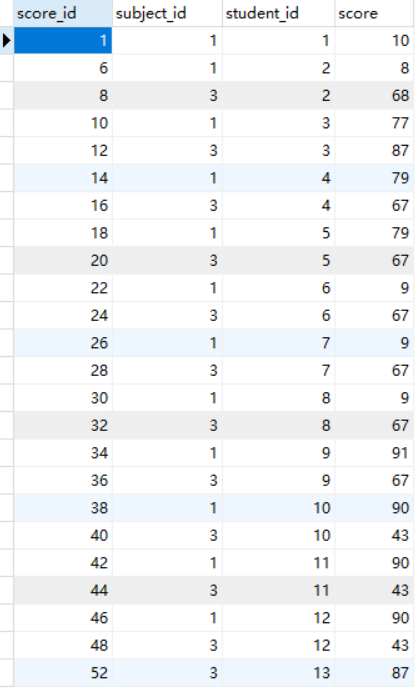
teacher

where

teacher\_name = '李平老师')) a

where

sc.subject\_id = a.subject\_id)

1. 向SC表中插入一些记录，这些记录要求符合以下条件：①没有上过编号“002”课程的同学学号；②插入“002”号课程的平均成绩；
2. 按平均成绩从低到高显示所有学生的“生物”、“物理”、“体育”三门的课程成绩，按如下形式显示： 学生ID,生物,物理,体育,有效课程数,有效平均分；

SELECT

sc.student\_id '学号',

(

SELECT

score

FROM

score

LEFT JOIN SUBJECT ON score.subject\_id = subject.subject\_id

WHERE

subject\_name = '生物'

AND score.student\_id = sc.student\_id

) AS '生物',

(

SELECT

score

FROM

score

LEFT JOIN SUBJECT ON score.subject\_id = subject.subject\_id

WHERE

subject\_name = '物理'

AND score.student\_id = sc.student\_id

) AS '物理',

(

SELECT

score

FROM

score

LEFT JOIN SUBJECT ON score.subject\_id = subject.subject\_id

WHERE

subject\_name = '体育'

AND score.student\_id = sc.student\_id

) AS '体育',

(

SELECT

score

FROM

score

LEFT JOIN SUBJECT ON score.subject\_id = subject.subject\_id

WHERE

subject\_name = '美术'

AND score.student\_id = sc.student\_id

) AS '美术',

count(sc.subject\_id) '有效课程数',

avg(sc.score) '有效平均分'

FROM

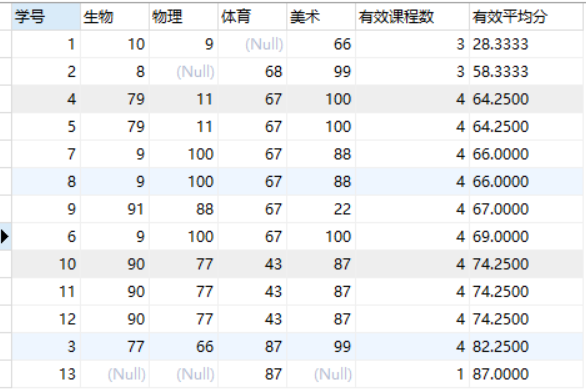
score sc

GROUP BY

student\_id

ORDER BY

avg(sc.score)



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；

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

t\_out1.subject\_id '课程',

t\_out1.avgnum '平均分',

t\_out2.pass\_per '及格率'

FROM

(

SELECT

subject\_id,

avg(score) avgnum

FROM

score

GROUP BY

subject\_id

) t\_out1

LEFT JOIN (

SELECT

t1.subject\_id,

t1.count1 / t2.count2 pass\_per

FROM

(

SELECT

subject\_id,

count(subject\_id) count1

FROM

score

WHERE

score > 60

GROUP BY

subject\_id

) t1

LEFT JOIN (

SELECT

subject\_id,

count(subject\_id) count2

FROM

score

GROUP BY

subject\_id

) t2 ON t1.subject\_id = t2.subject\_id

) t\_out2 ON t\_out1.subject\_id = t\_out2.subject\_id

ORDER BY

avgnum,

pass\_per DESC;



1. 查询各科成绩前三名的记录:(不考虑成绩并列情况)

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,

(

SELECT

score

FROM

score AS s2

WHERE

s2.subject\_id = s1.subject\_id

ORDER BY

score DESC

LIMIT 2,

1

) AS third\_num

FROM

score AS s1

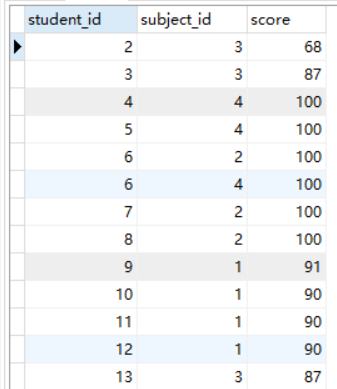
) t2 ON t1.score\_id = t2.score\_id

WHERE

t1.score = t2.first\_num

OR t1.score = t2.second\_num

OR t1.score = t2.third\_num;



1. 查询每门课程被选修的学生数；

SELECT

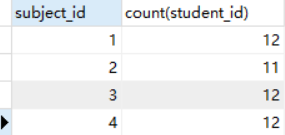
subject\_id,

count(student\_id)

from

score

GROUP BY subject\_id



1. 查询同名同姓学生名单，并统计同名人数；

SELECT

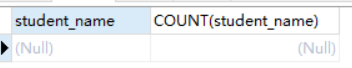
student\_name,

COUNT(student\_name)

FROM

student

GROUP BY student\_name HAVING student\_name >1



1. 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,

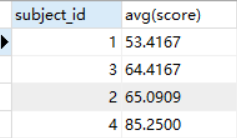
avg(score)

FROM

score

GROUP BY subject\_id

ORDER BY avg(score),subject\_id DESC



1. 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',

AVG(score) '平均分'

FROM

score sc , student stu

WHERE

sc.student\_id = stu.student\_id

GROUP BY sc.student\_id HAVING AVG(score)>85



1. 查询课程名称为“物理”，且分数低于60的学生姓名和分数；

select

stu.student\_id '学号',

stu.student\_name '姓名',

sc.score '成绩'

FROM

subject sub , score sc , student stu

where

sub.subject\_name = '物理'

and sc.score <60

and stu.student\_id = sc.student\_id

and sc.subject\_id = sub.subject\_id



1. 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

stu.student\_id '学号',

stu.student\_name '学生姓名'

from

score sc,student stu

where

subject\_id = 3

and score >80

and sc.student\_id = stu.student\_id



1. 求选了课程的学生人数

SELECT

COUNT(DISTINCT student\_id) '学生个数'

FROM

score



1. 查询选修“刘海燕”老师所授课程的学生中，成绩最高的学生姓名及其成绩；

SELECT

student.student\_name '姓名',

score.score '成绩'

from

score , teacher , subject,student

where

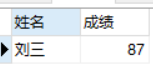
teacher\_name = '刘海燕老师'

and score.subject\_id = `subject`.subject\_id

and teacher.teacher\_id = `subject`.teacher\_id

and student.student\_id = score.student\_id

ORDER BY score DESC LIMIT 1



1. 查询各个课程及相应的选修人数；

SELECT

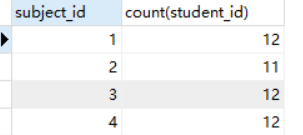
subject\_id,

count(student\_id)

FROM

score

GROUP BY subject\_id



1. 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 DISTINCT

sca.student\_id,

sca.subject\_id,

sca.score,

scb.student\_id,

scb.subject\_id,

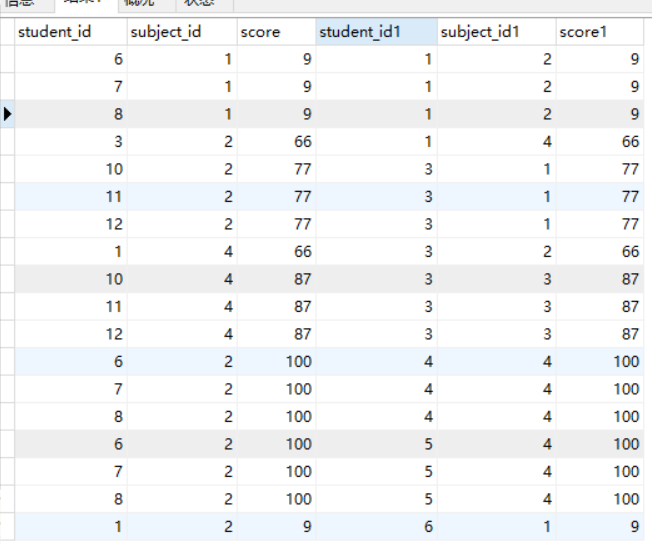
scb.score

FROM

score sca , score scb

WHERE

sca.score = scb.score and sca.subject\_id != scb.subject\_id



1. 查询每门课程成绩最好的前两名；

SELECT

\*

FROM

score

LEFT JOIN (

SELECT

score\_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

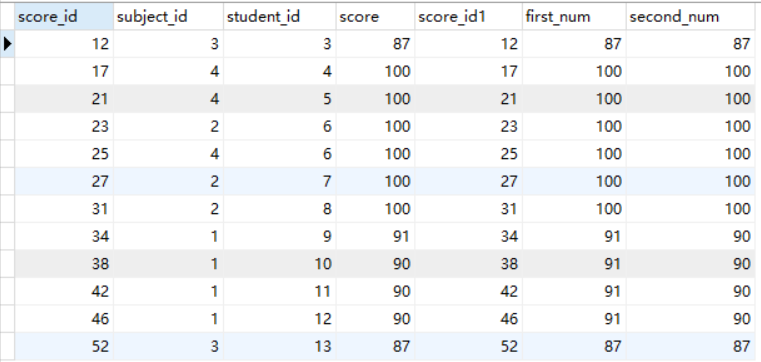
score AS s1

) AS T ON score.score\_id= T.score\_id

WHERE

score.score <= T.first\_num

AND score.score >= T.second\_num



1. 检索至少选修两门课程的学生学号；

SELECT

student\_id

FROM

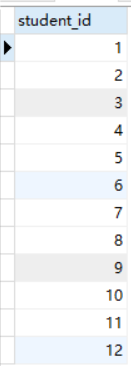
score

GROUP BY

student\_id

HAVING

COUNT(subject\_id)>=2



1. 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(student\_id) = (

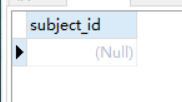
SELECT

count(student\_id)

FROM

student

);



1. 查询没学过“叶平”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

`subject`

WHERE

teacher\_id IN (

SELECT

teacher\_id

FROM

teacher

WHERE

teacher\_name = '李平老师'

)

)

)



1. 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',

avg(score) '平均分'

FROM

score

WHERE

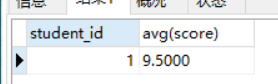
score < 60

GROUP BY

student\_id

HAVING

count(score) >= 2;



1. 检索“004”课程分数小于60，按分数降序排列的同学学号；

SELECT

student\_id

FROM

score

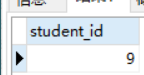
WHERE

score < 60

AND subject\_id = 4

ORDER BY

score DESC



1. 删除“002”同学的“001”课程的成绩；

DELETE

FROM score

where

student\_id = 2

and subject\_id = 1;