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 |
| student\_id | 学生id | INT(10) | 否 | 否 | 是 | 否 | 否 |
| subject\_id | 课程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 |

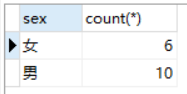
INSERT INTO `student` VALUES ('1', '男', '1', '理解'), ('2', '女', '1', '钢蛋'), ('3', '男', '1', '张三'), ('4', '男', '1', '张一'), ('5', '女', '1', '张二'), ('6', '男', '1', '张四'), ('7', '女', '2', '铁锤'), ('8', '男', '2', '李三'), ('9', '男', '2', '李一'), ('10', '女', '2', '李二'), ('11', '男', '2', '李四'), ('12', '女', '3', '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

sex,

count(\*)

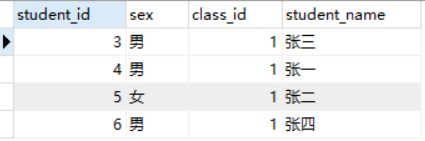
FROM

student

GROUP BY

sex;

1. 查询姓“张”的学生名单；



SELECT

\*

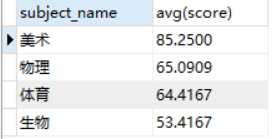
FROM

student

WHERE

student\_name LIKE '张%';

3.课程平均分从高到低显示



SELECT

le.subject\_name,

avg(score)

FROM

score as sc

LEFT JOIN

lesson as le

on

le.subject\_id=sc.subject\_id

GROUP BY

sc.subject\_id

ORDER BY

avg(score) desc

4.查询有课程成绩小于60分的同学的学号、姓名；



SELECT

stu.student\_id,

stu.student\_name,

MIN(sco.score)

from

student as stu

RIGHT JOIN

score as sco

on stu.student\_id=sco.student\_id

WHERE

sco.score < 60

GROUP BY

sco.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\_name,

stu.student\_id

FROM

score

RIGHT JOIN

(

SELECT

subject\_id

FROM

score

WHERE

student\_id=1) as sj

on score.subject\_id = sj.subject\_id

Left JOIN

student as stu

on score.student\_id = stu.student\_id

WHERE

score.student\_id <> 1

GROUP BY

student\_name

1. 查询出只选修了一门课程的全部学生的学号和姓名；



SELECT

score.student\_id,

student\_name,

count(\*)

FROM

score

Left JOIN

student

on score.student\_id=student.student\_id

GROUP BY

score.student\_id

HAVING

count(\*)=1

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

score.subject\_id '课程ID',

max(score) '最高分',

min(score) '最低分'

FROM

score

LEFT JOIN

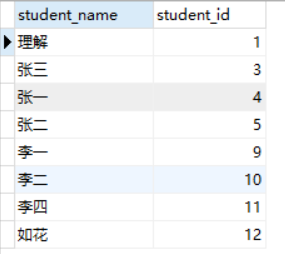
lesson

ON score.subject\_id=lesson.subject\_id

GROUP BY

lesson.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\_name,

s1.student\_id

FROM

(

SELECT

subject\_id,

student\_id,

score as sc1

FROM

score

WHERE

subject\_id=1

) as s1

inner JOIN

(

SELECT

subject\_id,

student\_id,

score as sc2

FROM

score

WHERE

subject\_id=2

)as s2

ON s1.student\_id=s2.student\_id

LEFT JOIN

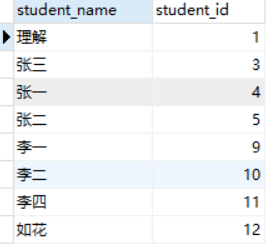
student

ON student.student\_id=s1.student\_id

WHERE

sc1>sc2

1. 查询“生物”课程比“物理”课程成绩高的所有学生的学号；



SELECT

student\_name,

s1.student\_id

FROM

(

SELECT

score.subject\_id,

student\_id,

score as sc1

FROM

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

WHERE

subject\_name='生物'

) as s1

inner JOIN

(

SELECT

score.subject\_id,

student\_id,

score as sc2

FROM

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

WHERE

subject\_name='物理'

)as s2

ON s1.student\_id=s2.student\_id

LEFT JOIN

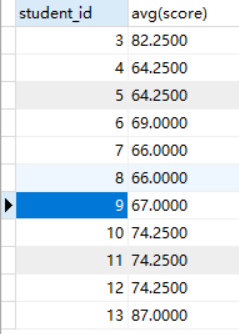
student

ON student.student\_id=s1.student\_id

WHERE

sc1>sc2

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,

avg(score)

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

s1.student\_id as '学号',

student\_name as '姓名',

cou as '选课数',

ss as '总成绩'

FROM

student LEFT JOIN

(

SELECT

student\_id,

count(\*) as cou,

sum(score) as ss

FROM

score

GROUP BY

student\_id

) as s1

ON student.student\_id=s1.student\_id

1. 查询姓“李”的老师的个数；



SELECT

count(\*)

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

score.student\_id,

student\_name,

teacher\_name

FROM

score

LEFT JOIN

lesson

ON

score.subject\_id=lesson.subject\_id

LEFT JOIN

teacher

ON

lesson.teacher\_id=teacher.teacher\_id

LEFT JOIN

student

ON

student.student\_id=score.student\_id

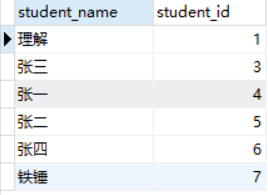
GROUP BY

score.student\_id

HAVING

teacher\_name not in('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\_name,

s1.student\_id

FROM

(

SELECT

subject\_id,

student\_id,

score as sc1

FROM

score

WHERE

subject\_id=1

) as s1

inner JOIN

(

SELECT

subject\_id,

student\_id,

score as sc2

FROM

score

WHERE

subject\_id=2

)as s2

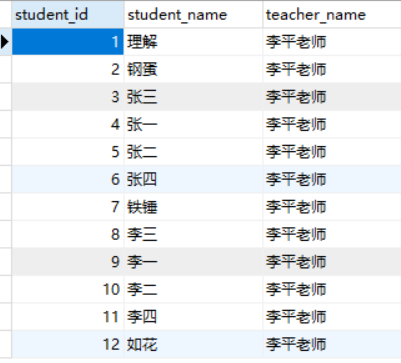
ON s1.student\_id=s2.student\_id

LEFT JOIN

student

ON student.student\_id=s1.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

score.student\_id,

student\_name,

teacher\_name

FROM

score

LEFT JOIN

lesson

ON

score.subject\_id=lesson.subject\_id

LEFT JOIN

teacher

ON

lesson.teacher\_id=teacher.teacher\_id

LEFT JOIN

student

ON

student.student\_id=score.student\_id

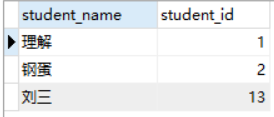
WHERE

teacher\_name ='李平老师'

GROUP BY

score.student\_id

1. 查询没有学全所有课的同学的学号、姓名；



SELECT

s1.student\_name,

s1.student\_id

FROM

(

SELECT

student\_name,

score.student\_id,

count(\*) AS cou

FROM

score

LEFT JOIN student ON score.student\_id = student.student\_id

GROUP BY

score.student\_id

) AS s1

WHERE

cou < (

SELECT

max(s2.cou1)

FROM

(

SELECT

count(\*) AS cou1

FROM

score

LEFT JOIN student ON score.student\_id = student.student\_id

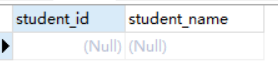
GROUP BY

score.student\_id

) AS s2

)

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

student.student\_id,

student\_name

FROM

student

LEFT JOIN (

SELECT

student\_id,

count(\*) AS cou

FROM

score

GROUP BY

student\_id

) AS s2 ON student.student\_id = s2.student\_id

WHERE

student.student\_id IN (

SELECT DISTINCT

student\_id

FROM

score

)

AND student.student\_id NOT IN (

SELECT DISTINCT

student\_id

FROM

(

SELECT

\*

FROM

score

WHERE

subject\_id NOT IN (

SELECT

subject\_id

FROM

score

WHERE

student\_id = 2

)

) AS s1

)

AND cou = (

SELECT

count(\*)

FROM

score

WHERE

student\_id = 2

)

AND student.student\_id != 2

1. 删除学习“叶平”老师课的SC表记录；

19.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；

20.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 DISTINCT

score.student\_id,

sc1 AS '语文',

sc2 AS '数学',

sc3 AS '英语',

av AS '有效平均分',

cou AS '有效科目个数'

FROM

score

LEFT JOIN (

SELECT

student\_id,

score AS sc1

FROM

score

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

WHERE

subject\_name IN ('语文')

) AS s2 ON score.student\_id = s2.student\_id

LEFT JOIN (

SELECT

student\_id,

score AS sc2

FROM

score

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

WHERE

subject\_name IN ('数学')

) AS s3 ON score.student\_id = s3.student\_id

LEFT JOIN (

SELECT

student\_id,

score AS sc3

FROM

score

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

WHERE

subject\_name IN ('英语')

) AS s4 ON score.student\_id = s4.student\_id

LEFT JOIN (

SELECT

student\_id,

count(\*) AS cou,

AVG(score) AS av

FROM

score

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

GROUP BY

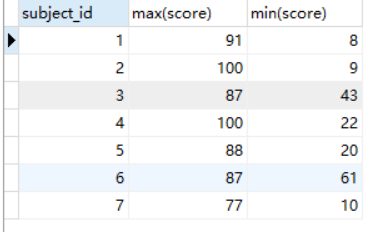
student\_id

) AS s1 ON s1.student\_id = score.student\_id

ORDER BY

av ASC

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,

max(score),

min(score)

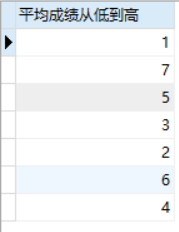
FROM

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 as '平均成绩从低到高'

FROM

score

GROUP BY

SUBJECT\_id

ORDER BY

avg(score)

SELECT

subject\_id as '及格率的百分数从高到低'

FROM

score

GROUP BY

subject\_id

ORDER BY

count(score>60)/count(\*)

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

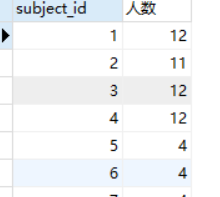
from score as s1

) t2

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

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

1. 查询每门课程被选修的学生数；



SELECT

subject\_id,

count(\*) as 人数

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(\*)

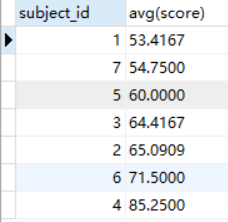
FROM

student

GROUP BY

student\_name

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

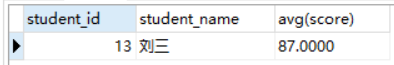
GROUP BY

subject\_id

ORDER BY

avg(score) asc,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

score.student\_id,

student\_name,

avg(score)

FROM

score

LEFT JOIN

student

ON

score.student\_id=student.student\_id

GROUP BY

score.student\_id

HAVING

avg(score) >85

1. 查询课程名称为“数学”，且分数低于60的学生姓名和分数；



SELECT

student\_name,

score

FROM

score

LEFT JOIN

lesson

ON

score.subject\_id=lesson.subject\_id

LEFT JOIN

student

on score.student\_id=student.student\_id

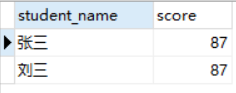
where

lesson.subject\_name='数学'

and

score.score<60

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

student\_name,

score

FROM

score

LEFT JOIN

lesson

ON

score.subject\_id=lesson.subject\_id

LEFT JOIN

student

on score.student\_id=student.student\_id

where

lesson.subject\_id=3

and

score.score>80

1. 求选了课程的学生人数



SELECT

count(\*)

FROM

(

SELECT

DISTINCT

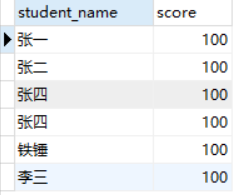
student\_id

FROM

score

) as s1

1. 查询选修“杨艳”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\_name,

score

FROM

score

LEFT JOIN

lesson

ON

score.subject\_id=lesson.subject\_id

LEFT JOIN

student

on score.student\_id=student.student\_id

LEFT JOIN

teacher

on

teacher.teacher\_id=lesson.teacher\_id

where

teacher.teacher\_name='李平老师'

AND score =(

SELECT

max(score)

FROM

score

LEFT JOIN

lesson

ON

score.subject\_id=lesson.subject\_id

LEFT JOIN

student

on score.student\_id=student.student\_id

LEFT JOIN

teacher

on

teacher.teacher\_id=lesson.teacher\_id

where

teacher.teacher\_name='李平老师'

)

1. 查询各个课程及相应的选修人数；



SELECT

subject\_name,

count(\*)

FROM

score

LEFT JOIN

lesson

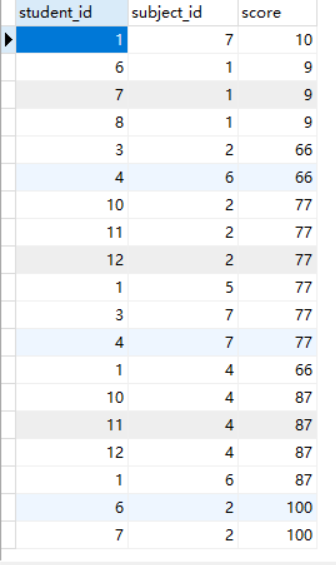
ON

score.subject\_id=lesson.subject\_id

GROUP BY

subject\_name

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

s1.student\_id,

s1.subject\_id,

s1.score

FROM

score s1

left JOIN

score s2

ON

s1.score=s2.score

WHERE

s1.subject\_id!=s2.subject\_id

1. 查询每门课程成绩最好的前两名；

SELECT

subject\_id,score

FROM

score sc1

WHERE

(SELECT

COUNT(\*)

FROM

score sc2

WHERE

sc1.score<sc2.score

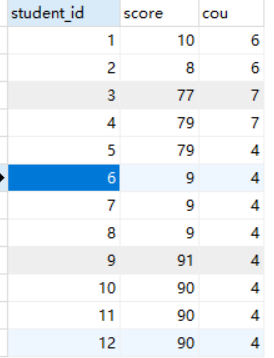
AND

sc1.subject\_id=sc2.subject\_id

)<2

ORDER BY subject\_id,score DESC

1. 检索至少选修两门课程的学生学号；



SELECT

student\_id,

score,

COUNT(\*) as cou

FROM

score

GROUP BY

score.student\_id

having

cou>=2

1. 查询全部学生都选修的课程的课程号和课程名；



SELECT

\*

FROM

lesson s1

WHERE

(

SELECT

count(\*)

FROM

score s2

WHERE

s1.subject\_id=s2.subject\_id

)=(SELECT

count(\*)

FROM

lesson

)

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

lesson

WHERE

teacher\_id IN (

SELECT

teacher\_id

FROM

teacher

WHERE

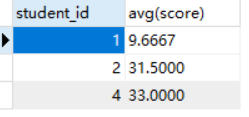
teacher\_name = '李平老师'

)

)

);

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,

avg(score)

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

subject\_id = 1

AND student\_id = 2