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

sex '性别',

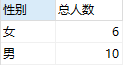
COUNT(sex) '总人数'

FROM

Student

GROUP BY sex

运行结果如图所示：



1. 查询姓“张”的学生名单；

SELECT

student\_id '学号',

sex '性别',

class\_id '班级id',

student\_name '姓名'

FROM

Student

WHERE

student\_name LIKE '张%'

运行结果如图所示：



1. 课程平均分从高到低显示

SELECT

subject\_id '课程',

AVG(score) '平均分'

FROM

Score

GROUP BY subject\_id

ORDER BY AVG(score) DESC

运行结果如图所示：



1. 查询有课程成绩小于60分的同学的学号、姓名；

SELECT DISTINCT

stu.student\_id '学号',

stu.student\_name '姓名'

FROM

Student stu,Score sc

WHERE

stu.student\_id=sc.student\_id

AND sc.score<60

运行结果如图所示：



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 DISTINCT

stu.student\_id '学号',

stu.student\_name '姓名'

FROM

Student stu,Score sc

WHERE

stu.student\_id=sc.student\_id

AND sc.subject\_id in(SELECT subject\_id

FROM Score WHERE student\_id=1)

运行结果如图所示：



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\_id '学号',

stu.student\_name '姓名'

FROM

Student stu,Score sc

WHERE

stu.student\_id=sc.student\_id

GROUP BY sc.student\_id

HAVING COUNT(sc.subject\_id)=1

运行结果如图所示：



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

运行结果如图所示：



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 DISTINCT

a.student\_id '学号',

a.student\_name '姓名'

FROM

(

SELECT

stu.student\_id,

stu.student\_name,

sc.score

FROM

Student stu,

Score sc

WHERE

stu.student\_id = sc.student\_id

AND sc.subject\_id = 1

) a,

(

SELECT

stu.student\_id,

stu.student\_name,

sc.score

FROM

Student stu,

Score sc

WHERE

stu.student\_id = sc.student\_id

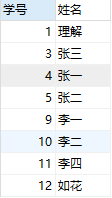
AND sc.subject\_id = 2

) b

WHERE

a.score > b.score

运行结果如图所示：



1. 查询“生物”课程比“物理”课程成绩高的所有学生的学号；

SELECT DISTINCT

a.student\_id '学号'

FROM

(

SELECT

sub.subject\_id,

sub.subject\_name,

sc.score,

sc.student\_id

FROM

SUBJECT sub,

Score sc

WHERE

sub.subject\_id = sc.subject\_id

AND sub.subject\_name = '生物'

) a,

(

SELECT

sub.subject\_id,

sub.subject\_name,

sc.score,

sc.student\_id

FROM

SUBJECT sub,

Score sc

WHERE

sub.subject\_id = sc.subject\_id

AND sub.subject\_name = '物理'

) b

WHERE

a.score > b.score

运行结果如图所示：



1. 查询平均成绩大于60分的同学的学号和平均成绩;

SELECT

stu.student\_id '学号',

AVG(score) '平均分'

FROM

Student stu,Score sc

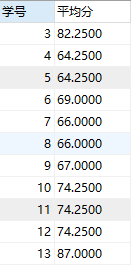
WHERE

stu.student\_id=sc.student\_id

GROUP BY stu.student\_id

HAVING AVG(score)>60

运行结果如图所示：



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',

COUNT(\*) '选课数',

SUM(sc.score) '总成绩'

FROM

Student stu,Score sc

WHERE

stu.student\_id=sc.student\_id

GROUP BY sc.student\_id

运行结果如图所示：



1. 查询姓“李”的老师的个数；

SELECT

COUNT(\*) '总数'

FROM

Teacher

WHERE

teacher\_name LIKE '李%'

运行结果如图所示：



1. 查询没学过“张磊老师”课的同学的学号、姓名；

SELECT

student\_id '学号',

student\_name '姓名'

FROM

Student

WHERE

student\_id NOT IN

(SELECT

stu.student\_id '学号'

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='张磊老师')

运行结果如图所示：



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

a.student\_id '学号',

a.student\_name '姓名'

FROM

(

SELECT

stu.student\_id,

stu.student\_name

FROM

Student stu,

Score sc

WHERE

stu.student\_id = sc.student\_id

AND sc.subject\_id = 1

) a,

(

SELECT

stu.student\_id,

stu.student\_name

FROM

Student stu,

Score sc

WHERE

stu.student\_id = sc.student\_id

AND sc.subject\_id = 2

) b

WHERE

a.student\_id = b.student\_id

运行结果如图所示：



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

运行结果如图所示：



1. 查询没有学全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

LEFT JOIN

Score sc

ON

stu.student\_id=sc.student\_id

GROUP BY stu.student\_id

HAVING COUNT(\*)<(SELECT COUNT(\*) FROM Subject)

运行结果如图所示：



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

)

运行结果如图所示：



1. 删除学习“李平”老师课的SC表记录；

DELETE

FROM

Score

WHERE

subject\_id IN (

SELECT

sub.subject\_id

FROM

Subject sub,Teacher tea

WHERE

sub.teacher\_id = tea.teacher\_id

AND tea.teacher\_name = '叶平老师'

)

运行结果如图所示：



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”、“美术”三门的课程成绩，按如下形式显示： 学生ID,生物,物理,体育,美术,有效课程数,有效平均分；

SELECT

student\_id '学生ID',

(SELECT sc.score FROM Score sc,Subject sub WHERE sc.subject\_id=sub.subject\_id AND sc.student\_id=sco.student\_id AND sub.subject\_name='生物') '生物',

(SELECT sc.score FROM Score sc,Subject sub WHERE sc.subject\_id=sub.subject\_id AND sc.student\_id=sco.student\_id AND sub.subject\_name='物理') '物理',

(SELECT sc.score FROM Score sc,Subject sub WHERE sc.subject\_id=sub.subject\_id AND sc.student\_id=sco.student\_id AND sub.subject\_name='体育') '体育',

(SELECT sc.score FROM Score sc,Subject sub WHERE sc.subject\_id=sub.subject\_id AND sc.student\_id=sco.student\_id AND sub.subject\_name='美术') '美术',

Count(\*) '有效课程数',

AVG(score) '有效平均分'

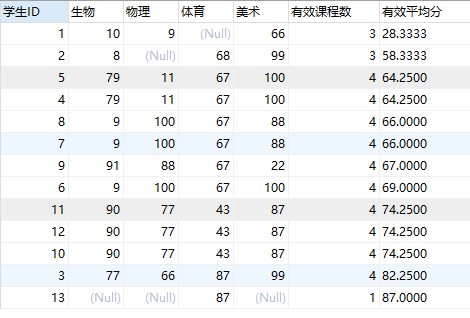
FROM

Score sco

GROUP BY student\_id

ORDER BY AVG(score) ASC

运行结果如图所示：



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

运行结果如图所示：



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

sc.subject\_id '课程ID',

subject\_name '课程',

AVG(score) '平均分',

100 \*(SELECT Count(subject\_id) FROM Score WHERE subject\_id=sc.subject\_id AND score>=60)/

(SELECT COUNT(subject\_id) FROM Score WHERE subject\_id=sc.subject\_id GROUP BY subject\_id) '及格率%'

FROM

Score sc,Subject sub

WHERE

sc.subject\_id=sub.subject\_id

GROUP BY sc.subject\_id

ORDER BY AVG(score) ASC,'及格率%' DESC

运行结果如图所示：



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 '学号',

t1.subject\_id '课程号',

t1.score '成绩'

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

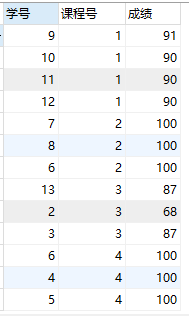
t1.score = t2.first\_num

OR t1.score = t2.second\_num

OR t1.score = t2.third\_num

ORDER BY t1.subject\_id

运行结果如图所示：



1. 查询每门课程被选修的学生数；

SELECT

sc.subject\_id '课程ID',

sub.subject\_name '课程名',

a.num '学生数'

FROM

Score sc,

Subject sub,

(

SELECT

subject\_id,

COUNT(student\_id) 'num'

FROM

Score

GROUP BY

subject\_id

) a

WHERE

sc.subject\_id = sub.subject\_id

AND sc.subject\_id = a.subject\_id

GROUP BY sc.subject\_id

运行结果如图所示：



1. 查询同名同姓学生名单，并统计同名人数；

SELECT

student\_name '姓名',

COUNT(\*)

FROM

Student

GROUP BY student\_name

HAVING COUNT(\*)>1

运行结果如图所示：



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

sub.subject\_id '课程ID',

sub.subject\_name'课程',

AVG(score) '平均分'

FROM

Score sc,Subject sub

WHERE

sc.subject\_id=sub.subject\_id

GROUP BY sub.subject\_id DESC

ORDER BY AVG(score),sub.subject\_id DESC

运行结果如图所示：



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

Student stu,Score sc

WHERE

stu.student\_id=sc.student\_id

GROUP BY stu.student\_id

HAVING AVG(score)>85

运行结果如图所示：



1. 查询课程名称为“生物”，且分数低于60的学生姓名和分数；

SELECT

stu.student\_name '姓名',

sc.score '分数'

FROM

Student stu,Score sc,Subject sub

WHERE

stu.student\_id=sc.student\_id

AND

sc.subject\_id=sub.subject\_id

AND

sub.subject\_name='生物'

AND

sc.score<60

运行结果如图所示：



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 '姓名'

FROM

Student stu,Score sc

WHERE

stu.student\_id=sc.student\_id

AND

sc.subject\_id=3

AND

sc.score>80

运行结果如图所示：



1. 求选了课程的学生人数

SELECT

COUNT(DISTINCT student\_id) '人数'

FROM

Score

运行结果如图所示：



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

stu.student\_name '姓名',

sc.subject\_id '课程号',

score '成绩'

FROM

Student stu,

Score sc

WHERE

stu.student\_id = sc.student\_id

AND score IN (

SELECT

MAX(score)

FROM

Score sc,

SUBJECT sub,

Teacher tea

WHERE

sc.subject\_id = sub.subject\_id

AND sub.teacher\_id = tea.teacher\_id

AND tea.teacher\_name = '李平老师'

)

ORDER BY sc.subject\_id

运行结果如图所示：



1. 查询各个课程及相应的选修人数；

SELECT

sub.subject\_id '课程ID',

sub.subject\_name '课程名',

COUNT(\*) '人数'

FROM

Score sc,Subject sub

WHERE

sc.subject\_id=sub.subject\_id

GROUP BY sc.subject\_id

运行结果如图所示：



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

stu.student\_id '学号',

sc.subject\_id '课程号',

sc.score '成绩'

FROM

Student stu,Score sc,Score sco

WHERE

stu.student\_id=sc.student\_id

AND

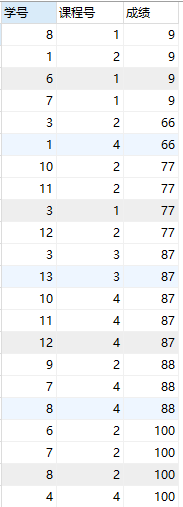
sc.subject\_id<>sco.subject\_id

AND

sc.score=sco.score

ORDER BY sc.score ASC

运行结果如图所示：



1. 查询每门课程成绩最好的前两名；

SELECT

t1.student\_id '学号',

t1.subject\_id '课程号',

t1.score '成绩'

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

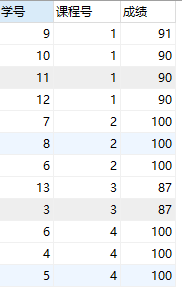
WHERE

t1.score = t2.first\_num

OR t1.score = t2.second\_num

ORDER BY t1.subject\_id

运行结果如图所示：



1. 检索至少选修两门课程的学生学号；

SELECT

stu.student\_id '学号'

FROM

Student stu,Score sc

WHERE

stu.student\_id=sc.student\_id

GROUP BY stu.student\_id

HAVING COUNT(\*)>=2

运行结果如图所示：



1. 查询全部学生都选修的课程的课程号和课程名；

SELECT

sub.subject\_id '课程号',

sub.subject\_name '课程名'

FROM

Student stu,

Score sc,

SUBJECT sub

WHERE

stu.student\_id = sc.student\_id

AND sc.subject\_id = sub.subject\_id

AND (

SELECT

COUNT(student\_id)

FROM

Score sco

WHERE

sco.subject\_id = sc.subject\_id

GROUP BY

subject\_id

) = (SELECT COUNT(\*) FROM Student)

运行结果如图所示：



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 '姓名'

FROM

Student

WHERE

student\_id NOT IN(

SELECT

stu.student\_id

FROM

Student stu,Score sc,Subject sub,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'

)

运行结果如图所示：



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 sc

WHERE

score<60

GROUP BY

student\_id

HAVING COUNT(\*) >= 2

运行结果如图所示：



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 '学号'

FROM

score

WHERE

subject\_id=4

AND

score<60

ORDER BY score DESC

运行结果如图所示：



40.删除“002”同学的“001”课程的成绩；

DELETE FROM Score WHERE student\_id=2 AND subject\_id=1

运行结果如图所示：

