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');

（插入数据代码位于最下方附件）

1. 查询男生、女生的人数；

SELECT

sex '性别',

COUNT( \* )

FROM

student

GROUP BY

sex



1. 查询姓“张”的学生名单；

SELECT

student\_id '学生编号',

sex '性别',

class\_id '班级',

student\_name '姓名'

FROM

student

WHERE

student\_name LIKE '张%'



1. 课程平均分从高到低显示

SELECT

subject\_name '课程名称',

AVG( score ) '平均分数'

FROM

score sc,

SUBJECT su

WHERE

sc.subject\_id = su.subject\_id

GROUP BY

sc.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

st.student\_id '学号',

st.student\_name '名字'

FROM

student st

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

WHERE

sc.score < 60

GROUP BY

st.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 DISTINCT

st.student\_id '学号',

student\_name '姓名'

FROM

student st

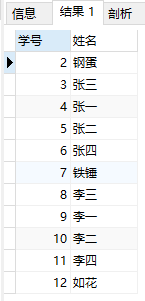
LEFT JOIN score sc ON st.student\_id = sc.student\_id,

( SELECT \* FROM score WHERE student\_id = 1 ) sc1

WHERE

sc.subject\_id IN ( sc1.subject\_id )

AND st.student\_id != 1



1. 查询出只选修了一门课程的全部学生的学号和姓名；

SELECT

st.student\_id '学号',

student\_name '姓名'

FROM

student st

LEFT JOIN ( SELECT student\_id, COUNT( subject\_id ) cs FROM score GROUP BY student\_id ) sc ON st.student\_id = sc.student\_id

WHERE

sc.cs = 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

subject\_id '课程ID',

max( score ) '最高分',

min( score ) '最低分'

FROM

score

GROUP BY

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 DISTINCT

st.student\_id '学号',

student\_name '姓名'

FROM

student st

LEFT JOIN ( SELECT student\_id, score FROM score WHERE subject\_id = 1 ) sc1

ON st.student\_id = sc1.student\_id,

( SELECT score FROM score WHERE subject\_id = 2 ) sc2

WHERE

sc2.score < sc1.score



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

s1.student\_id '学号'

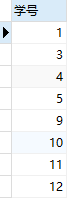
FROM

( SELECT sc1.score, student\_id FROM score sc1 LEFT JOIN subject su1 ON sc1.subject\_id = su1.subject\_id WHERE subject\_name = '生物' ) s1,

( SELECT sc2.score, student\_id FROM score sc2 LEFT JOIN subject su2 ON sc2.subject\_id = su2.subject\_id WHERE subject\_name = '物理' ) s2

WHERE

s1.score > s2.score



1. 查询平均成绩大于60分的同学的学号和平均成绩;

SELECT

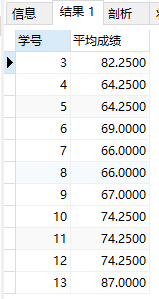
student\_id '学号',

AVG(score) '平均成绩'

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

st.student\_id '学号',

student\_name '姓名',

COUNT( sc.subject\_id ) '选课数',

SUM( sc.score ) '总成绩'

FROM

student st

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

GROUP BY

st.student\_id



1. 查询姓“李”的老师的个数；

SELECT

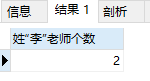
COUNT(\*) '姓“李”老师个数'

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

st.student\_id '学号',

student\_name '姓名'

FROM

student st

WHERE

st.student\_id NOT IN (

SELECT

sc.student\_id

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 = '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

st.student\_id '学号',

student\_name '姓名'

FROM

student st,

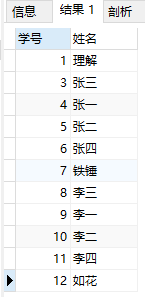
( SELECT student\_id FROM score WHERE subject\_id = 1 ) sc1,

( SELECT student\_id FROM score WHERE subject\_id = 2 ) sc2

WHERE

st.student\_id IN ( sc1.student\_id )

AND st.student\_id IN ( 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 DISTINCT

st.student\_id '学号',

student\_name '姓名'

FROM

student st,

score sc,

SUBJECT sub,

teacher tea

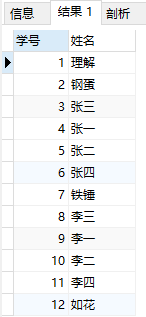
WHERE

st.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'



1. 查询没有学全所有课的同学的学号、姓名；

SELECT

st.student\_id '学号',

student\_name '姓名'

FROM

student st

LEFT JOIN ( SELECT student\_id, COUNT( student\_id ) co FROM score GROUP BY student\_id ) sc

ON st.student\_id = sc.student\_id

WHERE

sc.co < 4 OR sc.co IS NULL



17.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

st.student\_id '学号',

student\_name '姓名'

FROM

student st

RIGHT JOIN (

SELECT

student\_id

FROM

score

WHERE

student\_id NOT IN ( SELECT student\_id FROM score WHERE subject\_id NOT IN ( SELECT subject\_id FROM score WHERE student\_id = 2 ) )

AND student\_id != 2

GROUP BY

student\_id

) sc ON st.student\_id = sc.student\_id



18.删除学习“李平”老师课的SC表记录；

DELETE

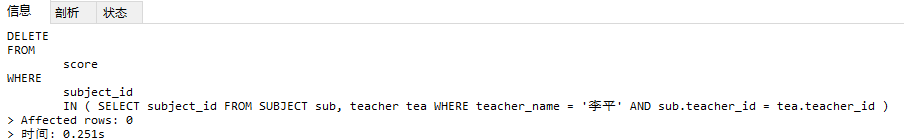
FROM

score

WHERE

subject\_id

IN ( SELECT subject\_id FROM SUBJECT sub, teacher tea WHERE teacher\_name = '李平' AND sub.teacher\_id = tea.teacher\_id )



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

sc.student\_id '学生ID',

s1.score '生物',

s2.score '物理',

s3.score '体育',

s4.score '美术',

COUNT( sc.subject\_id ) '有效课程数',

AVG( sc.score ) '有效平均分'

FROM

score sc

LEFT JOIN ( SELECT student\_id, score FROM score sc LEFT JOIN SUBJECT sub ON sc.subject\_id = sub.subject\_id WHERE subject\_name = '生物' ) s1 ON sc.student\_id = s1.student\_id

LEFT JOIN ( SELECT student\_id, score FROM score sc LEFT JOIN SUBJECT sub ON sc.subject\_id = sub.subject\_id WHERE subject\_name = '物理' ) s2 ON sc.student\_id = s2.student\_id

LEFT JOIN ( SELECT student\_id, score FROM score sc LEFT JOIN SUBJECT sub ON sc.subject\_id = sub.subject\_id WHERE subject\_name = '体育' ) s3 ON sc.student\_id = s3.student\_id

LEFT JOIN ( SELECT student\_id, score FROM score sc LEFT JOIN SUBJECT sub ON sc.subject\_id = sub.subject\_id WHERE subject\_name = '美术' ) s4 ON sc.student\_id = s4.student\_id

GROUP BY

sc.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

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

sc.subject\_id '学科ID',

AVG( sc.score ) '平均成绩',

s1.co / COUNT( sc.subject\_id ) \* 100 '及格率'

FROM

score sc

LEFT JOIN ( SELECT subject\_id, COUNT( \* ) co FROM score WHERE score >= 60 GROUP BY subject\_id ) s1

ON sc.subject\_id = s1.subject\_id

GROUP BY

sc.subject\_id

ORDER BY

平均成绩 ASC



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

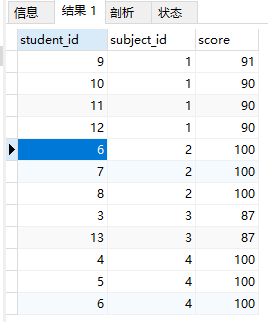
from score as s1

) t2

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

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

order by t1.subject\_id asc, t1.score desc



1. 查询每门课程被选修的学生数；

SELECT

subject\_id '科目',

COUNT( student\_id ) '选修学生数'

FROM

score

GROUP BY

subject\_id



25.查询同名同姓学生名单，并统计同名人数；

SELECT

student\_name '姓名',

COUNT( 1 ) '同名学生数'

FROM

student st

GROUP BY

student\_name



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ID',

AVG( score ) '平均成绩'

FROM

score

GROUP BY

subject\_id

ORDER BY

平均成绩 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

st.student\_id '学号',

st.student\_name '姓名',

sc.avg '平均成绩'

FROM

student st

LEFT JOIN ( SELECT student\_id, AVG( score ) AS avg FROM score GROUP BY student\_id ) sc

ON st.student\_id = sc.student\_id

WHERE

sc.avg > 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 '姓名',

cu.score '分数'

FROM

student st

LEFT JOIN

( SELECT student\_id, score FROM score sc LEFT JOIN SUBJECT su

ON sc.subject\_id = su.subject\_id WHERE su.subject\_name = '生物' ) cu

ON st.student\_id = cu.student\_id

WHERE

cu.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 '姓名',

sc.score '分数'

FROM

student st

LEFT JOIN

score sc ON st.student\_id = sc.student\_id

WHERE

sc.subject\_id = 3 AND sc.score > 80



1. 求选了课程的学生人数

SELECT

( SELECT COUNT( \* ) FROM student ) - COUNT( \* ) '选了课程的学生人数'

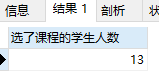
FROM

student st

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

WHERE

sc.subject\_id IS NULL



1. 查询选修“李平”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 '姓名',

MAX( score ) '成绩'

FROM

student st,

score sc,

SUBJECT sub,

teacher tea

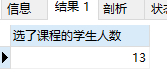
WHERE

st.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'



1. 查询各个课程及相应的选修人数；

SELECT

subject\_id '课程ID',

COUNT( subject\_id ) '选修人数'

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

s1.student\_id '学生1',

s1.subject\_id '课程1',

s1.score '成绩1',

s2.student\_id '学生2',

s2.subject\_id '课程2',

s2.score '成绩2'

FROM

score AS s1,

score AS s2

WHERE

s1.score = s2.score

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



1. 查询每门课程成绩最好的前两名；

SELECT DISTINCT

subject\_id '课程',

( SELECT student\_id FROM score sc WHERE sc.subject\_id = s.subject\_id GROUP BY score ORDER BY score DESC LIMIT 0 ,1 ) 'first\_student',

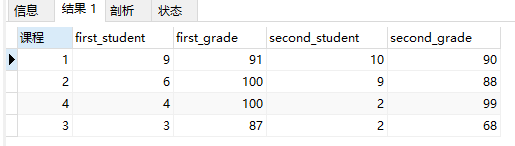
( SELECT score FROM score sc WHERE sc.subject\_id = s.subject\_id GROUP BY score ORDER BY score DESC LIMIT 0 ,1 ) 'first\_grade',

( SELECT student\_id FROM score sc WHERE sc.subject\_id = s.subject\_id GROUP BY score ORDER BY score DESC LIMIT 1 ,1 ) 'second\_student',

( SELECT score FROM score sc WHERE sc.subject\_id = s.subject\_id GROUP BY score ORDER BY score DESC LIMIT 1 ,1 ) 'second\_grade'

FROM

score s



1. 检索至少选修两门课程的学生学号；

SELECT

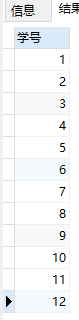
s.student\_id '学号'

FROM

( SELECT student\_id, count( \* ) AS co FROM score GROUP BY student\_id ) s

WHERE

s.co >= 2



1. 查询全部学生都选修的课程的课程号和课程名；

SELECT

sc.subject\_id '课程ID',

subject\_name '课程名称'

FROM

score sc

LEFT JOIN SUBJECT su ON sc.subject\_id = su.subject\_id

GROUP BY

sc.subject\_id

HAVING

COUNT( sc.subject\_id ) = ( SELECT COUNT( \* ) 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 '姓名'

FROM

student st

WHERE

st.student\_id NOT IN (

SELECT

student\_id

FROM

score sc,

SUBJECT su,

teacher tea

WHERE

sc.subject\_id = su.subject\_id

AND su.teacher\_id = tea.teacher\_id

AND tea.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

st.student\_id '学号',

sc.avg '平均成绩'

FROM

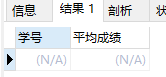
student st

LEFT JOIN ( SELECT student\_id, COUNT( \* ) co, AVG( score ) avg FROM score WHERE score < 60 GROUP BY student\_id ) sc

ON st.student\_id = sc.student\_id

WHERE

sc.co > 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 '学号'

FROM

score

WHERE

subject\_id = 4

AND score < 60

ORDER BY

score DESC



40.删除“002”同学的“001”课程的成绩；

DELETE

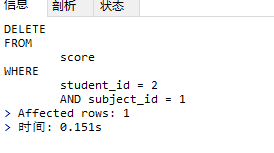
FROM

score

WHERE

student\_id = 2

AND subject\_id = 1



附件：

INSERT INTO `class`

VALUES

( '1', '三年二班' ),

( '2', '三年三班' ),

( '3', '一年二班' ),

( '4', '二年九班' );

INSERT INTO `subject`

VALUES

( '1', '生物', '1' ),

( '2', '物理', '2' ),

( '3', '体育', '3' ),

( '4', '美术', '2' );

INSERT INTO `score` ( score\_id, student\_id, subject\_id, 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' );

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' );

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' );