

- 表结构

班级表: class		学生表: student			
cid	caption	sid	sname	gender	class_id
1	三年二班	1	钢蛋	女	1
2	一年三班	2	铁锤	女	1
3	三年一班	3	山炮	男	2
老师表: teacher		课程表: course			
tid	tname	cid	cname	tearch_id	
1	波多	1	生物	1	
2	苍空	2	体育	1	
3	饭岛	3	物理	2	
成绩表: score					
sid	student_id	corse_id	number		
1	1	1	60		
2	1	2	59		
3	2	2	100		

```

1  /*
2   Navicat Premium Data Transfer
3
4   Source Server          : localhost
5   Source Server Type     : MySQL
6   Source Server Version : 50624
7   Source Host            : localhost
8   Source Database       : sqlexam
9
10  Target Server Type     : MySQL
11  Target Server Version : 50624
12  File Encoding         : utf-8
13
14  Date: 10/21/2016 06:46:46 AM
15  */
16
17  SET NAMES utf8;
18  SET FOREIGN_KEY_CHECKS = 0;
19
20  -- -----
21  -- Table structure for `class`
22  -- -----
23  DROP TABLE IF EXISTS `class`;
24  CREATE TABLE `class` (
25    `cid` int(11) NOT NULL AUTO_INCREMENT,
26    `caption` varchar(32) NOT NULL,
27    PRIMARY KEY (`cid`)
28  ) ENGINE=InnoDB AUTO_INCREMENT=5 DEFAULT CHARSET=utf8;
29

```

```

30  -- -----
31  --  Records of `class`
32  -- -----
33  BEGIN;
34  INSERT INTO `class` VALUES ('1', '三年二班'), ('2', '三年三班'), ('3', '一年二
    班'), ('4', '二年九班');
35  COMMIT;
36
37  -- -----
38  --  Table structure for `course`
39  -- -----
40  DROP TABLE IF EXISTS `course`;
41  CREATE TABLE `course` (
42      `cid` int(11) NOT NULL AUTO_INCREMENT,
43      `cname` varchar(32) NOT NULL,
44      `teacher_id` int(11) NOT NULL,
45      PRIMARY KEY (`cid`),
46      KEY `fk_course_teacher` (`teacher_id`),
47      CONSTRAINT `fk_course_teacher` FOREIGN KEY (`teacher_id`) REFERENCES
    `teacher` (`tid`)
48  ) ENGINE=InnoDB AUTO_INCREMENT=5 DEFAULT CHARSET=utf8;
49
50  -- -----
51  --  Records of `course`
52  -- -----
53  BEGIN;
54  INSERT INTO `course` VALUES ('1', '生物', '1'), ('2', '物理', '2'), ('3',
    '体育', '3'), ('4', '美术', '2');
55  COMMIT;
56
57  -- -----
58  --  Table structure for `score`
59  -- -----
60  DROP TABLE IF EXISTS `score`;
61  CREATE TABLE `score` (
62      `sid` int(11) NOT NULL AUTO_INCREMENT,
63      `student_id` int(11) NOT NULL,
64      `course_id` int(11) NOT NULL,
65      `num` int(11) NOT NULL,
66      PRIMARY KEY (`sid`),
67      KEY `fk_score_student` (`student_id`),
68      KEY `fk_score_course` (`course_id`),
69      CONSTRAINT `fk_score_course` FOREIGN KEY (`course_id`) REFERENCES
    `course` (`cid`),
70      CONSTRAINT `fk_score_student` FOREIGN KEY (`student_id`) REFERENCES
    `student` (`sid`)
71  ) ENGINE=InnoDB AUTO_INCREMENT=53 DEFAULT CHARSET=utf8;
72
73  -- -----
74  --  Records of `score`
75  -- -----
76  BEGIN;

```

```

77 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');
78 COMMIT;
79
80 -- -----
81 -- Table structure for `student`
82 -- -----
83 DROP TABLE IF EXISTS `student`;
84 CREATE TABLE `student` (
85     `sid` int(11) NOT NULL AUTO_INCREMENT,
86     `gender` char(1) NOT NULL,
87     `class_id` int(11) NOT NULL,
88     `sname` varchar(32) NOT NULL,
89     PRIMARY KEY (`sid`),
90     KEY `fk_class` (`class_id`),
91     CONSTRAINT `fk_class` FOREIGN KEY (`class_id`) REFERENCES `class`
    (`cid`)
92 ) ENGINE=InnoDB AUTO_INCREMENT=17 DEFAULT CHARSET=utf8;
93
94 -- -----
95 -- Records of `student`
96 -- -----
97 BEGIN;
98 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');
99 COMMIT;
100
101 -- -----
102 -- Table structure for `teacher`
103 -- -----
104 DROP TABLE IF EXISTS `teacher`;
105 CREATE TABLE `teacher` (
106     `tid` int(11) NOT NULL AUTO_INCREMENT,
107     `tname` varchar(32) NOT NULL,
108     PRIMARY KEY (`tid`)
109 ) ENGINE=InnoDB AUTO_INCREMENT=6 DEFAULT CHARSET=utf8;
110
111 -- -----
112 -- Records of `teacher`
113 -- -----

```

```

114 BEGIN;
115 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');
116 COMMIT;
117
118 SET FOREIGN_KEY_CHECKS = 1;
119

```

- 1、自行创建测试数据
- 2、查询“生物”课程比“物理”课程成绩高的所有学生的学号；
- 3、查询平均成绩大于60分的同学的学号和平均成绩；
- 4、查询所有同学的学号、姓名、选课数、总成绩；
- 5、查询姓“李”的老师的个数；
- 6、查询没学过“叶平”老师课的同学的学号、姓名；
- 7、查询学过“001”并且也学过编号“002”课程的同学的学号、姓名；
- 8、查询学过“叶平”老师所教的所有课的同学的学号、姓名；
- 9、查询课程编号“002”的成绩比课程编号“001”课程低的所有同学的学号、姓名；
- 10、查询有课程成绩小于60分的同学的学号、姓名；
- 11、查询没有学全所有课的同学的学号、姓名；
- 12、查询至少有一门课与学号为“001”的同学所学相同的同学的学号和姓名；
- 13、查询至少学过学号为“001”同学所选课程中任意一门课的其他同学学号和姓名；
- 14、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；
- 15、删除学习“叶平”老师课的SC表记录；
- 16、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；
- 17、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；
- 18、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；
- 19、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；
- 20、课程平均分从高到低显示（现实任课老师）；
- 21、查询各科成绩前三名的记录:(不考虑成绩并列情况)
- 22、查询每门课程被选修的学生数；
- 23、查询出只选修了一门课程的全部学生的学号和姓名；
- 24、查询男生、女生的人数；
- 25、查询姓“张”的学生名单；
- 26、查询同名同姓学生名单，并统计同名人数；
- 27、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；

- 28、查询平均成绩大于85的所有学生的学号、姓名和平均成绩；
- 29、查询课程名称为“数学”，且分数低于60的学生姓名和分数；
- 30、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；
- 31、求选了课程的学生人数
- 32、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；
- 33、查询各个课程及相应的选修人数；
- 34、查询不同课程但成绩相同的学生的学号、课程号、学生成绩；
- 35、查询每门课程成绩最好的前两名；
- 36、检索至少选修两门课程的学生学号；
- 37、查询全部学生都选修的课程的课程号和课程名；
- 38、查询没学过“叶平”老师讲授的任一门课程的学生姓名；
- 39、查询两门以上不及格课程的同学的学号及其平均成绩；
- 40、检索“004”课程分数小于60，按分数降序排列的同学学号；
- 41、删除“002”同学的“001”课程的成绩；