# SQL面试题目

介绍

主要的表是学生、课程、成绩、教师四张表，本示例的特点是有模拟数据，加深理解和印象，答案主要基于Oracle来实现的。MySQL需要做相应修改。

## 数据库建表

### 学生表

#列含义注释

STUDENT(SNO,SNAME,Sage,Ssex) 学生表

SNO：学号;

SNAME：学生姓名;

Sage：学生年龄;

Ssex：学生性别;

Oracle：创建语句

CREATE TABLE STUDENT (

SNO NUMBER(12,0) PRIMARY KEY,

SNAME VARCHAR2(10) NOT NULL,

SAGE NUMBER(3,0) NOT NULL,

SSEX CHAR(2) NOT NULL );

MySQL：创建语句

CREATE TABLE STUDENT (

SNO INT PRIMARY KEY,

SNAME VARCHAR(10) NOT NULL,

SAGE INT NOT NULL,

SSEX CHAR(2) NOT NULL );

表描述：

COMMENT ON TABLE STUDENT IS '学生信息表';

COMMENT ON COLUMN STUDENT.SNO IS '学号';

COMMENT ON COLUMN STUDENT.SNAME IS '学生姓名';

COMMENT ON COLUMN STUDENT.SAGE IS '学生年龄';

COMMENT ON COLUMN STUDENT.SSEX IS '学生性别';

### 课程表

#列含义注释

COURSE(CNO,Cname,TNO) 课程表

CNO：课程编号；

Cname：课程名字；

TNO：教师编号 ；

Oracle：创建语句

CREATE TABLE COURSE (

CNO NUMBER(12,0) PRIMARY KEY,

CNAME VARCHAR2(10) NOT NULL,

TNO NUMBER(12,0) NOT NULL );

MySQL：创建语句

CREATE TABLE COURSE (

CNO INT PRIMARY KEY,

CNAME VARCHAR(10) NOT NULL,

TNO INT NOT NULL );

表描述

COMMENT ONTABLE COURSE IS '课程表';

COMMENT ON COLUMN COURSE.CNO IS '课程编号';

COMMENT ON COLUMN COURSE.CNAME IS '课程名字';

COMMENT ON COLUMN COURSE.TNO IS '教师编号';

### 成绩表

#列含义注释

SC(SNO,CNO,SCore) 成绩表

SNO：学号；

CNO：课程编号；

SCore：成绩 ；

Oracle：创建语句

CREATE TABLE SC (

SNO NUMBER(12,0),

CNO NUMBER(12,0),

SCORE NUMBER(3,0) NOT NULL ) ;

MySQL：创建语句

CREATE TABLE SC (

SNO INT,

CNO INT,

SCORE INT NOT NULL ) ;

表描述：

COMMENT ON TABLE SC IS '成绩表';

COMMENT ON COLUMN SC.SNO IS '学号';

COMMENT ON COLUMN SC.CNO IS '课程编号';

COMMENT ON COLUMN SC.SCORE IS '成绩';

ALTER TABLE SC ADD CONSTRAINTS PK\_SC PRIMARY KEY (SNO,CNO);

### 教师表

#列含义注释

TEACHER(TNO,TNAME) 教师表

TNO：教师编号；

TNAME：教师名字；

Oracle：创建语句

CREATE TABLE TEACHER (

TNO NUMBER(12,0) PRIMARY KEY,

TNAME VARCHAR2(10) NOT NULL );

MySQL：创建语句

CREATE TABLE TEACHER (

TNO INT PRIMARY KEY,

TNAME VARCHAR(10) NOT NULL );

表描述

COMMENT ON TABLE TEACHER IS '教师表';

COMMENT ON COLUMN TEACHER.TNO IS '教师编号';

COMMENT ON COLUMN TEACHER.TNAME IS '教师名字';

## 模拟数据

### 学生表

INSERT INTO STUDENT(SNO,SNAME,SAGE,SSEX) VALUES(1,'小红',10,'女');

INSERT INTO STUDENT(SNO,SNAME,SAGE,SSEX) VALUES(2,'小明',12,'男');

INSERT INTO STUDENT(SNO,SNAME,SAGE,SSEX) VALUES(3,'小绿',13,'男');

INSERT INTO STUDENT(SNO,SNAME,SAGE,SSEX) VALUES(4,'小蓝',15,'男');

INSERT INTO STUDENT(SNO,SNAME,SAGE,SSEX) VALUES(5,'小青',11,'男');

INSERT INTO STUDENT(SNO,SNAME,SAGE,SSEX) VALUES(6,'小白',10,'女');

INSERT INTO STUDENT(SNO,SNAME,SAGE,SSEX) VALUES(7,'小紫',14,'女');

INSERT INTO STUDENT(SNO,SNAME,SAGE,SSEX) VALUES(8,'小黄',13,'男');

INSERT INTO STUDENT(SNO,SNAME,SAGE,SSEX) VALUES(9,'小粉',12,'女');

INSERT INTO STUDENT(SNO,SNAME,SAGE,SSEX) VALUES(10,'小橙',11,'女');

INSERT INTO STUDENT(SNO,SNAME,SAGE,SSEX) VALUES(11,'小黑',9,'男');

INSERT INTO STUDENT(SNO,SNAME,SAGE,SSEX) VALUES(12,'小粉',14,'男');

\*插入后记得commit提交数据

### 课程表

INSERT INTO COURSE(CNO,CNAME,TNO) VALUES(111,'python1',50);

INSERT INTO COURSE(CNO,CNAME,TNO) VALUES(112,'python2',51);

INSERT INTO COURSE(CNO,CNAME,TNO) VALUES(113,'linux1',52);

INSERT INTO COURSE(CNO,CNAME,TNO) VALUES(114,'linux2',53);

INSERT INTO COURSE(CNO,CNAME,TNO) VALUES(115,'mysql1',54);

INSERT INTO COURSE(CNO,CNAME,TNO) VALUES(116,'mysql2',55);

INSERT INTO COURSE(CNO,CNAME,TNO) VALUES(117,'java1',56);

INSERT INTO COURSE(CNO,CNAME,TNO) VALUES(118,'java2',57);

INSERT INTO COURSE(CNO,CNAME,TNO) VALUES(119,'大数据1',58);

INSERT INTO COURSE(CNO,CNAME,TNO) VALUES(120,'大数据2',59);

INSERT INTO COURSE(CNO,CNAME,TNO) VALUES(121,'go1',60);

INSERT INTO COURSE(CNO,CNAME,TNO) VALUES(122,'go2',61);

INSERT INTO COURSE(CNO,CNAME,TNO) VALUES(123,'mongodb1',50);

INSERT INTO COURSE(CNO,CNAME,TNO) VALUES(124,'mongodb2',54);

INSERT INTO COURSE(CNO,CNAME,TNO) VALUES(125,'redis1',58);

INSERT INTO COURSE(CNO,CNAME,TNO) VALUES(126,'redis2',53);

INSERT INTO COURSE(CNO,CNAME,TNO) VALUES(127,'docker1',56);

INSERT INTO COURSE(CNO,CNAME,TNO) VALUES(128,'docker2',60);

\*插入后记得commit提交数据

### 成绩表

INSERT INTO SC(SNO,CNO,SCORE) VALUES(1,111,69); INSERT INTO SC(SNO,CNO,SCORE) VALUES(1,112,80); INSERT INTO SC(SNO,CNO,SCORE) VALUES(1,113,75); INSERT INTO SC(SNO,CNO,SCORE) VALUES(1,114,79); INSERT INTO SC(SNO,CNO,SCORE) VALUES(1,115,63); INSERT INTO SC(SNO,CNO,SCORE) VALUES(1,116,90); INSERT INTO SC(SNO,CNO,SCORE) VALUES(1,117,95); INSERT INTO SC(SNO,CNO,SCORE) VALUES(1,118,77); INSERT INTO SC(SNO,CNO,SCORE) VALUES(1,119,60); INSERT INTO SC(SNO,CNO,SCORE) VALUES(1,120,83); INSERT INTO SC(SNO,CNO,SCORE) VALUES(1,121,88); INSERT INTO SC(SNO,CNO,SCORE) VALUES(1,122,75); INSERT INTO SC(SNO,CNO,SCORE) VALUES(1,123,50); INSERT INTO SC(SNO,CNO,SCORE) VALUES(1,124,44); INSERT INTO SC(SNO,CNO,SCORE) VALUES(1,125,77); INSERT INTO SC(SNO,CNO,SCORE) VALUES(1,127,79); INSERT INTO SC(SNO,CNO,SCORE) VALUES(1,128,77); INSERT INTO SC(SNO,CNO,SCORE) VALUES(2,111,77); INSERT INTO SC(SNO,CNO,SCORE) VALUES(2,112,68); INSERT INTO SC(SNO,CNO,SCORE) VALUES(2,113,76); INSERT INTO SC(SNO,CNO,SCORE) VALUES(2,114,95); INSERT INTO SC(SNO,CNO,SCORE) VALUES(2,115,90); INSERT INTO SC(SNO,CNO,SCORE) VALUES(2,116,57); INSERT INTO SC(SNO,CNO,SCORE) VALUES(2,117,99); INSERT INTO SC(SNO,CNO,SCORE) VALUES(2,118,86); INSERT INTO SC(SNO,CNO,SCORE) VALUES(2,120,73); INSERT INTO SC(SNO,CNO,SCORE) VALUES(2,121,73); INSERT INTO SC(SNO,CNO,SCORE) VALUES(2,122,82); INSERT INTO SC(SNO,CNO,SCORE) VALUES(2,123,67); INSERT INTO SC(SNO,CNO,SCORE) VALUES(2,124,69); INSERT INTO SC(SNO,CNO,SCORE) VALUES(2,125,53); INSERT INTO SC(SNO,CNO,SCORE) VALUES(2,126,64); INSERT INTO SC(SNO,CNO,SCORE) VALUES(2,127,75); INSERT INTO SC(SNO,CNO,SCORE) VALUES(2,128,71); INSERT INTO SC(SNO,CNO,SCORE) VALUES(3,111,82); INSERT INTO SC(SNO,CNO,SCORE) VALUES(3,112,67); INSERT INTO SC(SNO,CNO,SCORE) VALUES(3,113,69); INSERT INTO SC(SNO,CNO,SCORE) VALUES(3,114,44); INSERT INTO SC(SNO,CNO,SCORE) VALUES(3,115,77); INSERT INTO SC(SNO,CNO,SCORE) VALUES(3,116,98); INSERT INTO SC(SNO,CNO,SCORE) VALUES(3,117,79); INSERT INTO SC(SNO,CNO,SCORE) VALUES(3,118,77); INSERT INTO SC(SNO,CNO,SCORE) VALUES(3,119,68); INSERT INTO SC(SNO,CNO,SCORE) VALUES(3,120,76); INSERT INTO SC(SNO,CNO,SCORE) VALUES(3,121,95); INSERT INTO SC(SNO,CNO,SCORE) VALUES(3,122,73); INSERT INTO SC(SNO,CNO,SCORE) VALUES(3,123,82); INSERT INTO SC(SNO,CNO,SCORE) VALUES(3,124,63); INSERT INTO SC(SNO,CNO,SCORE) VALUES(3,125,90); INSERT INTO SC(SNO,CNO,SCORE) VALUES(3,126,95); INSERT INTO SC(SNO,CNO,SCORE) VALUES(3,127,77); INSERT INTO SC(SNO,CNO,SCORE) VALUES(3,128,60); INSERT INTO SC(SNO,CNO,SCORE) VALUES(4,111,77); INSERT INTO SC(SNO,CNO,SCORE) VALUES(4,112,68); INSERT INTO SC(SNO,CNO,SCORE) VALUES(4,113,76); INSERT INTO SC(SNO,CNO,SCORE) VALUES(4,114,95); INSERT INTO SC(SNO,CNO,SCORE) VALUES(4,115,73); INSERT INTO SC(SNO,CNO,SCORE) VALUES(4,116,82); INSERT INTO SC(SNO,CNO,SCORE) VALUES(4,117,76); INSERT INTO SC(SNO,CNO,SCORE) VALUES(4,118,95); INSERT INTO SC(SNO,CNO,SCORE) VALUES(4,119,90); INSERT INTO SC(SNO,CNO,SCORE) VALUES(4,120,57); INSERT INTO SC(SNO,CNO,SCORE) VALUES(4,121,69); INSERT INTO SC(SNO,CNO,SCORE) VALUES(4,122,80); INSERT INTO SC(SNO,CNO,SCORE) VALUES(4,123,75); INSERT INTO SC(SNO,CNO,SCORE) VALUES(4,124,79); INSERT INTO SC(SNO,CNO,SCORE) VALUES(4,125,68); INSERT INTO SC(SNO,CNO,SCORE) VALUES(4,126,54); INSERT INTO SC(SNO,CNO,SCORE) VALUES(4,127,77); INSERT INTO SC(SNO,CNO,SCORE) VALUES(4,128,90); INSERT INTO SC(SNO,CNO,SCORE) VALUES(5,111,77); INSERT INTO SC(SNO,CNO,SCORE) VALUES(5,112,98); INSERT INTO SC(SNO,CNO,SCORE) VALUES(5,113,79); INSERT INTO SC(SNO,CNO,SCORE) VALUES(5,115,68); INSERT INTO SC(SNO,CNO,SCORE) VALUES(5,116,85); INSERT INTO SC(SNO,CNO,SCORE) VALUES(5,117,83); INSERT INTO SC(SNO,CNO,SCORE) VALUES(5,118,81); INSERT INTO SC(SNO,CNO,SCORE) VALUES(5,119,95); INSERT INTO SC(SNO,CNO,SCORE) VALUES(5,120,90); INSERT INTO SC(SNO,CNO,SCORE) VALUES(5,121,57); INSERT INTO SC(SNO,CNO,SCORE) VALUES(5,122,69); INSERT INTO SC(SNO,CNO,SCORE) VALUES(5,124,75); INSERT INTO SC(SNO,CNO,SCORE) VALUES(5,125,75); INSERT INTO SC(SNO,CNO,SCORE) VALUES(5,126,70); INSERT INTO SC(SNO,CNO,SCORE) VALUES(5,127,53); INSERT INTO SC(SNO,CNO,SCORE) VALUES(5,128,50); INSERT INTO SC(SNO,CNO,SCORE) VALUES(6,113,90); INSERT INTO SC(SNO,CNO,SCORE) VALUES(6,114,57); INSERT INTO SC(SNO,CNO,SCORE) VALUES(6,115,69); INSERT INTO SC(SNO,CNO,SCORE) VALUES(6,116,80); INSERT INTO SC(SNO,CNO,SCORE) VALUES(6,117,60); INSERT INTO SC(SNO,CNO,SCORE) VALUES(6,118,73); INSERT INTO SC(SNO,CNO,SCORE) VALUES(6,120,73); INSERT INTO SC(SNO,CNO,SCORE) VALUES(6,121,76); INSERT INTO SC(SNO,CNO,SCORE) VALUES(6,122,95); INSERT INTO SC(SNO,CNO,SCORE) VALUES(6,123,90); INSERT INTO SC(SNO,CNO,SCORE) VALUES(6,124,57); INSERT INTO SC(SNO,CNO,SCORE) VALUES(6,125,69); INSERT INTO SC(SNO,CNO,SCORE) VALUES(6,126,80); INSERT INTO SC(SNO,CNO,SCORE) VALUES(6,128,74); INSERT INTO SC(SNO,CNO,SCORE) VALUES(7,111,57); INSERT INTO SC(SNO,CNO,SCORE) VALUES(7,112,69); INSERT INTO SC(SNO,CNO,SCORE) VALUES(7,114,60); INSERT INTO SC(SNO,CNO,SCORE) VALUES(7,115,73); INSERT INTO SC(SNO,CNO,SCORE) VALUES(7,116,90); INSERT INTO SC(SNO,CNO,SCORE) VALUES(7,124,57); INSERT INTO SC(SNO,CNO,SCORE) VALUES(7,125,43); INSERT INTO SC(SNO,CNO,SCORE) VALUES(7,126,65); INSERT INTO SC(SNO,CNO,SCORE) VALUES(7,127,73); INSERT INTO SC(SNO,CNO,SCORE) VALUES(7,128,80); INSERT INTO SC(SNO,CNO,SCORE) VALUES(8,111,71); INSERT INTO SC(SNO,CNO,SCORE) VALUES(8,112,69); INSERT INTO SC(SNO,CNO,SCORE) VALUES(8,113,77); INSERT INTO SC(SNO,CNO,SCORE) VALUES(8,114,81); INSERT INTO SC(SNO,CNO,SCORE) VALUES(8,115,95); INSERT INTO SC(SNO,CNO,SCORE) VALUES(8,116,90); INSERT INTO SC(SNO,CNO,SCORE) VALUES(8,117,57); INSERT INTO SC(SNO,CNO,SCORE) VALUES(8,119,65); INSERT INTO SC(SNO,CNO,SCORE) VALUES(8,120,73); INSERT INTO SC(SNO,CNO,SCORE) VALUES(8,121,80); INSERT INTO SC(SNO,CNO,SCORE) VALUES(8,122,66); INSERT INTO SC(SNO,CNO,SCORE) VALUES(8,123,80); INSERT INTO SC(SNO,CNO,SCORE) VALUES(8,124,60); INSERT INTO SC(SNO,CNO,SCORE) VALUES(8,125,73); INSERT INTO SC(SNO,CNO,SCORE) VALUES(8,126,90); INSERT INTO SC(SNO,CNO,SCORE) VALUES(8,127,73); INSERT INTO SC(SNO,CNO,SCORE) VALUES(8,128,76); INSERT INTO SC(SNO,CNO,SCORE) VALUES(9,111,68); INSERT INTO SC(SNO,CNO,SCORE) VALUES(9,112,72); INSERT INTO SC(SNO,CNO,SCORE) VALUES(9,114,45); INSERT INTO SC(SNO,CNO,SCORE) VALUES(9,115,73); INSERT INTO SC(SNO,CNO,SCORE) VALUES(9,116,80); INSERT INTO SC(SNO,CNO,SCORE) VALUES(9,117,66); INSERT INTO SC(SNO,CNO,SCORE) VALUES(9,118,80); INSERT INTO SC(SNO,CNO,SCORE) VALUES(9,121,92); INSERT INTO SC(SNO,CNO,SCORE) VALUES(9,122,57); INSERT INTO SC(SNO,CNO,SCORE) VALUES(9,123,69); INSERT INTO SC(SNO,CNO,SCORE) VALUES(9,124,80); INSERT INTO SC(SNO,CNO,SCORE) VALUES(9,125,60); INSERT INTO SC(SNO,CNO,SCORE) VALUES(9,126,73); INSERT INTO SC(SNO,CNO,SCORE) VALUES(9,127,79); INSERT INTO SC(SNO,CNO,SCORE) VALUES(10,111,71); INSERT INTO SC(SNO,CNO,SCORE) VALUES(10,112,69); INSERT INTO SC(SNO,CNO,SCORE) VALUES(10,113,77); INSERT INTO SC(SNO,CNO,SCORE) VALUES(10,114,81);

INSERT INTO SC(SNO,CNO,SCORE) VALUES(10,116,96);

INSERT INTO SC(SNO,CNO,SCORE) VALUES(10,117,50); INSERT INTO SC(SNO,CNO,SCORE) VALUES(10,118,40); INSERT INTO SC(SNO,CNO,SCORE) VALUES(10,119,63); INSERT INTO SC(SNO,CNO,SCORE) VALUES(10,120,92);

INSERT INTO SC(SNO,CNO,SCORE) VALUES(10,121,57);

INSERT INTO SC(SNO,CNO,SCORE) VALUES(10,122,69); INSERT INTO SC(SNO,CNO,SCORE) VALUES(10,123,80); INSERT INTO SC(SNO,CNO,SCORE) VALUES(10,124,62); INSERT INTO SC(SNO,CNO,SCORE) VALUES(10,125,73);

INSERT INTO SC(SNO,CNO,SCORE) VALUES(10,127,92);

INSERT INTO SC(SNO,CNO,SCORE) VALUES(10,128,73); INSERT INTO SC(SNO,CNO,SCORE) VALUES(11,124,99); INSERT INTO SC(SNO,CNO,SCORE) VALUES(12,122,100); INSERT INTO SC(SNO,CNO,SCORE) VALUES(12,126,99);

INSERT INTO SC(SNO,CNO,SCORE) VALUES(12,124,97);

\*插入后记得commit提交数据

### 教师表

INSERT INTO TEACHER(TNO,TNAME) VALUES(50,'李泳谊');

INSERT INTO TEACHER(TNO,TNAME) VALUES(51,'oldboy');

INSERT INTO TEACHER(TNO,TNAME) VALUES(52,'何清');

INSERT INTO TEACHER(TNO,TNAME) VALUES(53,'张耀');

INSERT INTO TEACHER(TNO,TNAME) VALUES(54,'周玉强');

INSERT INTO TEACHER(TNO,TNAME) VALUES(55,'夏东志');

INSERT INTO TEACHER(TNO,TNAME) VALUES(56,'赵志超');

INSERT INTO TEACHER(TNO,TNAME) VALUES(57,'谭建鑫');

INSERT INTO TEACHER(TNO,TNAME) VALUES(58,'Alex');

INSERT INTO TEACHER(TNO,TNAME) VALUES(59,'刘老师');

INSERT INTO TEACHER(TNO,TNAME) VALUES(60,'郭加磊');

INSERT INTO TEACHER(TNO,TNAME) VALUES(61,'武老师');

\*插入后记得commit提交数据

# 题目要求

## 查询“111”课程比“112”课程成绩高的所有学生的学号

select a.SNO,a.CNO,a.SCore,b.CNO,b.SCore from (

select SNO,CNO,SCore

from SC

where CNO=111) a

join (select SNO,CNO,SCore

from SC

where CNO=112) b

on a.SNO=b.SNO and a.SCore>b.SCore;

+------+------+-------+------+-------+

| SNO | CNO | SCore | CNO | SCore |

+------+------+-------+------+-------+

| 2 | 111 | 77 | 112 | 68 |

| 3 | 111 | 82 | 112 | 67 |

| 4 | 111 | 77 | 112 | 68 |

| 8 | 111 | 71 | 112 | 69 |

| 10 | 111 | 71 | 112 | 69 |

+------+------+-------+------+-------+

## 查询平均成绩大于60分的同学的学号和平均成绩；

select SNO,AVG(SCORE) as b from SC

group by SNO

HAVING b>60;

+------+---------+

| SNO | b |

+------+---------+

| 1 | 74.1765 |

| 2 | 75.0000 |

| 3 | 76.2222 |

| 4 | 76.7222 |

| 5 | 75.3125 |

| 6 | 74.5000 |

| 7 | 66.7000 |

| 8 | 75.0588 |

| 9 | 71.0000 |

| 10 | 71.5625 |

| 11 | 99.0000 |

| 12 | 98.6667 |

+------+---------+

## 查询所有同学的学号、姓名、选课数、总成绩；

select SC.SNO,STUDENT.SNAME,COUNT(SC.CNO),SUM(SC.SCORE)

from SC

join STUDENT

on SC.SNO=STUDENT.SNO

group by SC.SNO;

+------+--------+---------------+---------------+

| SNO | SNAME | COUNT(SC.CNO) | SUM(SC.SCORE) |

+------+--------+---------------+---------------+

| 1 | 小红 | 17 | 1261 |

| 2 | 小明 | 17 | 1275 |

| 3 | 小绿 | 18 | 1372 |

| 4 | 小蓝 | 18 | 1381 |

| 5 | 小青 | 16 | 1205 |

| 6 | 小白 | 14 | 1043 |

| 7 | 小紫 | 10 | 667 |

| 8 | 小黄 | 17 | 1276 |

| 9 | 小粉 | 14 | 994 |

| 10 | 小橙 | 16 | 1145 |

| 11 | 小黑 | 1 | 99 |

| 12 | 小粉 | 3 | 296 |

+------+--------+---------------+---------------+

## 查询姓“李”的老师的个数；

select TNAME,COUNT(TNAME) from TEACHER group by TNAME having TNAME like '李%';

+-----------+--------------+

| TNAME | COUNT(TNAME) |

+-----------+--------------+

| 李泳谊 | 1 |

+-----------+--------------+

mysql> select count(TNAME) from TEACHER where TNAME like '李%';

+--------------+

| count(TNAME) |

+--------------+

| 1 |

+--------------+

## 查询没学过'郭加磊'老师课的同学的学号、姓名；

select STUDENT.SNO,STUDENT.SNAME from STUDENT

where NOT EXISTS (

select t1.SNO,t1.SNAME from (

(select distinct STUDENT.SNO,STUDENT.SNAME,TEACHER.TNAME

from STUDENT

join SC

on STUDENT.SNO=SC.SNO

join COURSE

on SC.CNO=COURSE.CNO

join TEACHER

on COURSE.TNO=TEACHER.TNO

where TEACHER.TNAME like '郭加磊') as t1 )

where t1.SNO=STUDENT.SNO);

+-----+--------+

| SNO | SNAME |

+-----+--------+

| 11 | 小黑 |

| 12 | 小粉 |

+-----+--------+

## 查询学过“111”并且也学过编号“112”课程的同学的学号、姓名；

select distinct STUDENT.SNO,STUDENT.SNAME from STUDENT join SC on STUDENT.SNO=SC.SNO where SC.CNO==111

+-----+--------+

| SNO | SNAME |

+-----+--------+

| 1 | 小红 |

| 2 | 小明 |

| 3 | 小绿 |

| 4 | 小蓝 |

| 5 | 小青 |

| 7 | 小紫 |

| 8 | 小黄 |

| 9 | 小粉 |

| 10 | 小橙 |

+-----+--------+

## 查询学过“郭加磊”老师所教的所有课的同学的学号、姓名；

select distinct STUDENT.SNO,STUDENT.SNAME

from STUDENT

join SC

on STUDENT.SNO=SC.SNO

join COURSE

on SC.CNO=COURSE.CNO

join TEACHER

on COURSE.TNO=TEACHER.TNO

where TEACHER.TNAME like '郭加磊';

## 查询课程编号“112”的成绩比课程编号“111”课程低的所有同学的学号、姓名

select t1.SNO,STUDENT.SNAME from

(select t1.SNO,t1.CNO,t1.SCORE from SC as t1

where t1.CNO=112) as t1

join (select t2.SNO,t2.CNO,t2.SCORE from SC as t2

where t2.CNO=111) as t2

on t1.SNO=t2.SNO

join STUDENT

on STUDENT.SNO=t1.SNO

where t1.SCORE>t2.SCORE;

+------+--------+

| SNO | SNAME |

+------+--------+

| 1 | 小红 |

| 5 | 小青 |

| 7 | 小紫 |

| 9 | 小粉 |

+------+--------+

## 查询所有课程成绩小于60分的同学的学号、姓名

select distinct SC.SNO,STUDENT.SNAME from SC

join STUDENT

on SC.SNO=STUDENT.SNO

where SCORE<60;

+------+--------+

| SNO | SNAME |

+------+--------+

| 1 | 小红 |

| 2 | 小明 |

| 3 | 小绿 |

| 4 | 小蓝 |

| 5 | 小青 |

| 6 | 小白 |

| 7 | 小紫 |

| 8 | 小黄 |

| 9 | 小粉 |

| 10 | 小橙 |

+------+--------+

## 查询没有学全所有课的同学的学号、姓名

select t2.SNO,STUDENT.SNAME from (select \* from (

select SNO,count(sno) as sum from SC

group by SNO) as t1

where t1.sum!=(select count(CNO) as sum from COURSE)) as t2

join STUDENT

on STUDENT.SNO=t2.SNO;

+------+--------+

| SNO | SNAME |

+------+--------+

| 1 | 小红 |

| 2 | 小明 |

| 5 | 小青 |

| 6 | 小白 |

| 7 | 小紫 |

| 8 | 小黄 |

| 9 | 小粉 |

| 10 | 小橙 |

| 11 | 小黑 |

| 12 | 小粉 |

+------+--------+

## 查询至少有一门课与学号为“1”的同学所学相同的同学的学号和姓名

select distinct t1.SNO,t1.SNAME from STUDENT as t1

join SC as t2

on t1.SNO=t2.SNO

where t2.CNO in (select CNO from SC where SNO=1);

+-----+--------+

| SNO | SNAME |

+-----+--------+

| 1 | 小红 |

| 2 | 小明 |

| 3 | 小绿 |

| 4 | 小蓝 |

| 5 | 小青 |

| 6 | 小白 |

| 7 | 小紫 |

| 8 | 小黄 |

| 9 | 小粉 |

| 10 | 小橙 |

| 11 | 小黑 |

| 12 | 小粉 |

+-----+--------+

## 查询至少学过学号为“001”同学所有一门课的其他同学学号和姓名

select distinct t1.SNO,t1.SNAME from STUDENT as t1

join SC as t2

on t1.SNO=t2.SNO

and t1.SNO!=1

where t2.CNO in (select CNO from SC where SNO=1);

+-----+--------+

| SNO | SNAME |

+-----+--------+

| 2 | 小明 |

| 3 | 小绿 |

| 4 | 小蓝 |

| 5 | 小青 |

| 6 | 小白 |

| 7 | 小紫 |

| 8 | 小黄 |

| 9 | 小粉 |

| 10 | 小橙 |

| 11 | 小黑 |

| 12 | 小粉 |

+-----+--------+

## 把“SC”表中“郭加磊”老师教的课的成绩都更改为此课程的平均成绩

select t1.CNO,avg(t1.SCORE),t3.TNAME from SC as t1

join COURSE as t2

on t1.CNO=t2.CNO

join TEACHER as t3

on t2.TNO=t3.TNO

where t3.TNAME="郭加磊"

group by t1.CNO;

+------+---------------+-----------+

| CNO | avg(t1.SCORE) | TNAME |

+------+---------------+-----------+

| 121 | 76.3333 | 郭加磊 |

| 128 | 72.3333 | 郭加磊 |

+------+---------------+-----------+

UPDATE SC t4 SET t4.SCore = (

WITH tmp AS (

SELECT t1.CNO,

AVG(t1.SCore) AS avg\_SCore

FROM SC t1

INNER JOIN COURSE t2

ON t1.CNO = t2.CNO

INNER JOIN TEACHER t3

ON t2.TNO = t3.TNO

GROUP BY t1.CNO )

SELECT t5.avg\_SCore

FROM tmp t5

WHERE t4.CNO = t5.CNO )

WHERE EXISTS (

SELECT t6.CNO

FROM SC t6

INNER JOIN COURSE t7

ON t6.CNO = t7.CNO

INNER JOIN TEACHER t8

ON t7.TNO = t8.TNO

AND t8.TNAME = '郭加磊' )

## 查询和“2”号的同学学习的课程完全相同的其他同学学号和姓名

## 删除学习'郭加磊'老师课的SC表记录

delete from SC where CNO in(select a.cno from (  
select distinct t1.CNO from SC t1  
join COURSE t2  
on t1.CNO=t2.CNO  
join TEACHER t3  
on t2.TNO=t3.TNO  
where t3.TNAME="郭加磊") as a);

## 向SC表中插入一些记录，这些记录要求符合以下条件：没有上过编号“003”课程的同学学号、2号课的平均成绩

## 按平均成绩从高到低显示“python1”、“java2”、“go2”三门的课程成绩

select t2.CNAME, avg(t1.SCORE) SCORE from SC t1

join COURSE t2

on t1.CNO=t2.CNO

where t2.CNAME in("python1","go2","java2")

group by t2.CNAME

order by SCORE desc;

+---------+---------+

| CNAME | SCORE |

+---------+---------+

| go2 | 78.7273 |

| java2 | 76.1250 |

| python1 | 72.1111 |

+---------+---------+

## 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 CNO,max(SCORE),min(SCORE) from SC

group by CNO;

+------+------------+------------+

| CNO | max(SCORE) | min(SCORE) |

+------+------------+------------+

| 111 | 82 | 57 |

| 112 | 98 | 67 |

| 113 | 90 | 69 |

| 114 | 95 | 44 |

| 115 | 95 | 63 |

| 116 | 98 | 57 |

| 117 | 99 | 50 |

| 118 | 95 | 40 |

| 119 | 95 | 60 |

| 120 | 92 | 57 |

| 122 | 100 | 57 |

| 123 | 90 | 50 |

| 124 | 99 | 44 |

| 125 | 90 | 43 |

| 126 | 99 | 54 |

| 127 | 92 | 53 |

+------+------------+------------+

## 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

## 查询如下课程平均成绩和及格率的百分数(用"1行"显示)

python1（111），python2（112），linux1（113），linux2（114）

## 查询不同老师所教不同课程平均分从高到低显示

select t1.TNAME,t2.CNAME,avg(t3.SCORE) as cj from TEACHER t1

join COURSE t2

on t1.TNO=t2.TNO

join SC t3

on t2.CNO=t3.CNO

group by t1.TNAME,t2.CNAME

order by cj desc;

+-----------+------------+---------+

| TNAME | CNAME | cj |

+-----------+------------+---------+

| 夏东志 | mysql2 | 84.8000 |

| 张耀 | redis2 | 78.9000 |

| 武老师 | go2 | 78.7273 |

| 何清 | linux1 | 77.3750 |

| 刘老师 | 大数据2 | 77.1250 |

| 谭建鑫 | java2 | 76.1250 |

| 周玉强 | mysql1 | 75.6667 |

| 赵志超 | docker1 | 75.3333 |

| 李泳谊 | mongodb1 | 74.1250 |

| 赵志超 | java1 | 73.8889 |

| Alex | 大数据1 | 73.5000 |

| oldboy | python2 | 73.3333 |

| 周玉强 | mongodb2 | 72.2308 |

| 李泳谊 | python1 | 72.1111 |

| 张耀 | linux2 | 70.7778 |

| Alex | redis1 | 68.1000 |

+-----------+------------+---------+

## 查询如下课程成绩第 3 名到第 6 名的学生成绩单：python1（111），python2（112），linux1 （113），linux2（114）

## 统计列印各科成绩,各分数段人数:课程ID,课程名称,[100-85] 优秀人数,[85-70] 良好人数,[70-60] 一般人数,[ <60] 刚及格人数

## 查询学生平均成绩及其名次

select STUDENT.SNAME,SC.SNO,avg(SCORE) as CJ from SC

join STUDENT

on SC.SNO=STUDENT.SNO

group by SNO

order by CJ desc;

+--------+------+---------+

| SNAME | SNO | CJ |

+--------+------+---------+

| 小黑 | 11 | 99.0000 |

| 小粉 | 12 | 98.6667 |

| 小青 | 5 | 78.4286 |

| 小蓝 | 4 | 76.3750 |

| 小绿 | 3 | 76.0625 |

| 小明 | 2 | 75.4000 |

| 小黄 | 8 | 74.6667 |

| 小白 | 6 | 74.4167 |

| 小红 | 1 | 73.0667 |

| 小橙 | 10 | 72.5000 |

| 小粉 | 9 | 69.3846 |

| 小紫 | 7 | 65.2222 |

+--------+------+---------+

## 查询各科成绩前三名的记录:(不考虑成绩并列情况)

## 查询每门课程被选修的学生数

select t1.CNAME,count(t2.SCORE) sum from COURSE t1

join SC t2

on t1.CNO=t2.CNO

group by t1.CNAME;

+------------+-----+

| CNAME | sum |

+------------+-----+

| docker1 | 9 |

| go2 | 11 |

| java1 | 9 |

| java2 | 8 |

| linux1 | 8 |

| linux2 | 9 |

| mongodb1 | 8 |

| mongodb2 | 13 |

| mysql1 | 9 |

| mysql2 | 10 |

| python1 | 9 |

| python2 | 9 |

| redis1 | 10 |

| redis2 | 10 |

| 大数据1 | 6 |

| 大数据2 | 8 |

+------------+-----+

## 查询出只选修了一门课程的全部学生的学号和姓名

select t1.SNO,t1.SNAME from STUDENT t1

join SC t2

on t1.SNO=t2.SNO

group by t1.SNO,t1.SNAME

having count(SCore)=1;

+-----+--------+

| SNO | SNAME |

+-----+--------+

| 11 | 小黑 |

+-----+--------+

## 查询男生、女生人数

select SSEX,count(SSEX) from STUDENT

group by SSEX;

+------+-------------+

| SSEX | count(SSEX) |

+------+-------------+

| 女 | 5 |

| 男 | 7 |

+------+-------------+

## 查询名字中有'黑'的学生名单

select \* from STUDENT

where SNAME like '%黑%';

+-----+--------+------+------+

| SNO | SNAME | SAGE | SSEX |

+-----+--------+------+------+

| 11 | 小黑 | 9 | 男 |

+-----+--------+------+------+

## 查询同名同性学生名单，并统计同名人数

select t1.SNAME,count(t1.SNAME) from STUDENT t1

group by t1.SNAME

having count(t1.SNAME)>1;

+--------+-----------------+

| SNAME | count(t1.SNAME) |

+--------+-----------------+

| 小粉 | 2 |

+--------+-----------------+

## 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 t1.CNAME,t1.CNO,avg(t2.SCORE) sum from COURSE t1

join SC t2

on t1.CNO=t2.CNO

group by t1.CNAME,t1.CNO

order by sum asc,t1.CNO desc;

+------------+-----+---------+

| CNAME | CNO | sum |

+------------+-----+---------+

| redis1 | 125 | 68.1000 |

| linux2 | 114 | 70.7778 |

| python1 | 111 | 72.1111 |

| mongodb2 | 124 | 72.2308 |

| python2 | 112 | 73.3333 |

| 大数据1 | 119 | 73.5000 |

| java1 | 117 | 73.8889 |

| mongodb1 | 123 | 74.1250 |

| docker1 | 127 | 75.3333 |

| mysql1 | 115 | 75.6667 |

| java2 | 118 | 76.1250 |

| 大数据2 | 120 | 77.1250 |

| linux1 | 113 | 77.3750 |

| go2 | 122 | 78.7273 |

| redis2 | 126 | 78.9000 |

| mysql2 | 116 | 84.8000 |

+------------+-----+---------+

## 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 t1.SNO,t1.SNAME,avg(t2.SCORE) CJ from STUDENT t1

join SC t2

on t1.SNO=t2.SNO

group by t1.SNO,t1.SNAME

having avg(t2.SCORE)>85;

+-----+--------+---------+

| SNO | SNAME | CJ |

+-----+--------+---------+

| 11 | 小黑 | 99.0000 |

| 12 | 小粉 | 98.6667 |

+-----+--------+---------+

## 查询课程名称为'python1'，且分数低于60的学生姓名和分数

select t1.SNAME,t2.SCORE from STUDENT t1

join SC t2

on t1.SNO=t2.SNO

join COURSE t3

on t2.CNO=t3.CNO

where t3.CNAME='python1'

and

t2.SCORE<60;

+--------+-------+

| SNAME | SCORE |

+--------+-------+

| 小紫 | 57 |

+--------+-------+

## 查询所有学生的选课情况

select t1.SNO,t1.SNAME,t3.CNAME from STUDENT t1

join SC t2

on t1.SNO=t2.SNO

join COURSE t3

on t2.CNO=t3.CNO;

## 查询任何一门课程成绩在70分以上的姓名、课程名称和分数

select t1.SNAME,t3.CNAME,t2.SCORE from STUDENT t1

join SC t2

on t1.SNO=t2.SNO

join COURSE t3

on t2.CNO=t3.CNO

where t2.SCORE>70;

## 查询不及格的课程，并按课程号从大到小排列

select t1.CNO,t1.Cname,t2.SCore from COURSE t1

join SC t2

on t1.CNO=t2.CNO

where t2.SCORE<60

order by t1.CNO desc;

## 查询课程编号为113且课程成绩在80分以上的学生的学号和姓名

select t1.SNO,t1.SNAME from STUDENT t1

join SC t2

on t1.SNO=t2.SNO

where t2.CNO=113

and

t2.SCORE>80;

+-----+--------+-------+

| SNO | SNAME | SCORE |

+-----+--------+-------+

| 6 | 小白 | 90 |

+-----+--------+-------+

## 求选了课程的学生人数

select count(t1.zh) from (select SNO zh from SC group by SNO) as t1;

+--------------+

| count(t1.zh) |

+--------------+

| 13 |

+--------------+

## 查询选修“郭加磊”老师所授课程的学生中，成绩最高的学生姓名及其成绩

select t1.SNAME,t2.SCORE,t3.CNAME from STUDENT t1

join SC t2

on t1.SNO=t2.SNO

join COURSE t3

on t2.CNO=t3.CNO

join TEACHER t4

on t3.TNO=t4.TNO

where (t2.SCORE,t3.CNAME)

in (select max(t5.SCORE),t6.CNAME from SC t5

join COURSE t6

ON t5.CNO=t6.CNO

join TEACHER t7

on t6.TNO=t7.TNO

where t7.TNAME="郭加磊"

group by t6.CNAME);

+--------+-------+---------+

| SNAME | SCORE | CNAME |

+--------+-------+---------+

| 小绿 | 95 | go1 |

| 小蓝 | 90 | docker2 |

+--------+-------+---------+

## 查询各个课程及相应的选修人数

select t1.CNAME,count(t2.SNO) from COURSE t1

join SC t2

on t1.CNO=t2.CNO

group by t1.CNAME;

+------------+---------------+

| CNAME | count(t2.SNO) |

+------------+---------------+

| docker1 | 9 |

| go2 | 11 |

| java1 | 9 |

| java2 | 8 |

| linux1 | 8 |

| linux2 | 9 |

| mongodb1 | 8 |

| mongodb2 | 13 |

| mysql1 | 9 |

| mysql2 | 10 |

| python1 | 9 |

| python2 | 9 |

| redis1 | 10 |

| redis2 | 10 |

| 大数据1 | 6 |

| 大数据2 | 8 |

+------------+---------------+

## 查询不同课程成绩相同的学生的学号、课程号、学生成绩

SELECT t1.SNO,t1.CNO,t1.SCore

FROM SC t1

INNER JOIN SC t2

ON t1.SNO = t2.SNO

AND t1.CNO <> t2.CNO

AND t1.SCore = t2.SCore

order by t1.SNO ,t1.SCore;

## 查询每门功成绩最好的前两名

## 统计每门课程的学生选修人数（超过10人的课程才统计）。要求输出课程号和选修人数，查询结果按人数降序排列，若人数相同，按课程号升序排列

select CNO,count(SNO) sum from SC

group by CNO

having count(SNO)>10

order by sum desc,CNO asc;

+------+-----+

| CNO | sum |

+------+-----+

| 124 | 13 |

| 122 | 11 |

+------+-----+

## 检索至少选修两门课程的学生学号

select SNO from SC

group by SNO

having count(SCore)>2;

+------+

| SNO |

+------+

| 1 |

| 2 |

| 3 |

| 4 |

| 5 |

| 6 |

| 7 |

| 8 |

| 9 |

| 10 |

| 12 |

+------+

## 查询全部学生都选修的课程的课程号和课程名

select t1.CNO,t2.CNAME,count(t1.SNO) from SC t1

join COURSE t2

on t1.CNO=t2.CNO

group by t1.CNO,t2.CNAME

having count(t1.SNO)=(

select count(a.SNO) from (select SNO from SC group by SNO) a);

+------+----------+---------------+

| CNO | CNAME | count(t1.SNO) |

+------+----------+---------------+

| 124 | mongodb2 | 13 |

+------+----------+---------------+

## 查询没学过'郭加磊'老师讲授的任一门课程的学生姓名

select distinct SNAME from STUDENT

where SNO not in (

select distinct t1.SNO from SC t1

join COURSE t2

on t1.CNO=t2.CNO

join TEACHER t3

on t2.TNO=t3.TNO

where t3.TNAME="郭加磊");

+--------+

| SNAME |

+--------+

| 小黑 |

| 小粉 |

+--------+

## 查询两门以上不及格课程的同学的学号及其平均成绩

select t1.SNO,avg(SCORE) from SC t1

where t1.SNO in (

select SNO from SC

where SCORE<60

group by SNO

having count(SCORE)>2)

group by t1.SNO;

+------+------------+

| SNO | avg(SCORE) |

+------+------------+

| 5 | 75.3125 |

| 7 | 66.7000 |

| 10 | 71.5625 |

+------+------------+

## 检索'114'课程分数小于60，按分数降序排列的同学学号

select SNO,SCORE from SC

where CNO=114

and

SCORE<60

order by SCORE desc;

+------+-------+

| SNO | SCORE |

+------+-------+

| 6 | 57 |

| 9 | 45 |

| 3 | 44 |

+------+-------+

## 删除'2'同学的'111'课程的成绩

delete from SC

where SNO=2S

and CNO=111;