综合练习1

此套练习供《尚硅谷-MySQL数据库基础篇》使用

数据表：

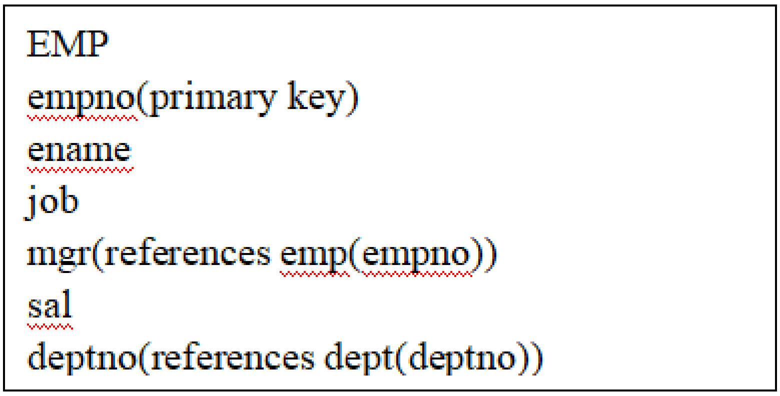
- dept:

deptno(primary key), dname, loc

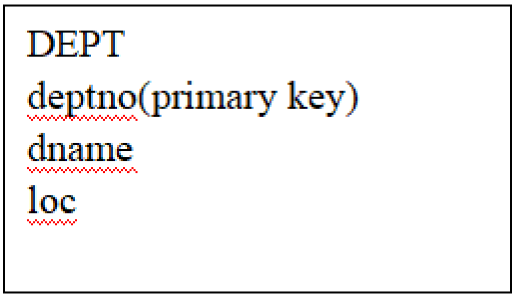
- emp:

empno(primary key), ename, job,mgr(references emp(empno)), sal,

deptno(references dept(deptno))



EMP
empno(primary key)
ename
job
mgr(references emp(empno))
sal
deptno(references dept(deptno))



DEPT
deptno(primary key)
dname
loc

问题

```
1. 列出emp表中各部门的部门号，最高工资，最低工资

2. 列出emp表中各部门job 含'REP'的员工的部门号，最低工资，最高工资

3. 对于emp中最低工资小于7000的部门中job为'SA_REP'的员工的部门号，最低工资，最高工资

4. 写出对上题的另一解决方法
（请补充）

5. 根据部门号由高而低，工资由低而高列出每个员工的姓名，部门号，工资

6. 列出'Abel'所在部门中每个员工的姓名与部门号
```

```
7．列出每个员工的姓名，工作，部门号，部门名

8．列出emp中工作为'SH_CLERK'的员工的姓名，工作，部门号，部门名

9．对于emp中有管理者的员工，列出姓名，管理者姓名（管理者外键为mgr）

10．对于dept表中，列出所有部门名，部门号，同时列出各部门工作为'SH_CLERK'的员工名与工作

11．对于工资高于本部门平均水平的员工，列出部门号，姓名，工资，按部门号排序

12．对于emp，列出各个部门中工资高于本部门平均水平的员工数和部门号，按部门号排序

13．对于emp中工资高于本部门平均水平，人数多于1人的，列出部门号，高于部门平均工资的人数，按部门号排序

14．对于emp中工资高于本部门平均水平，且其人数多于3人的，列出部门号，部门人数，按部门号排序

15．对于emp中低于自己工资至少5人的员工，列出其部门号，姓名，工资，以及工资少于自己的人数
```

答案

1. 列出emp表中各部门的部门号，最高工资，最低工资

```
select max(sal) as 最高工资,min(sal) as 最低工资,deptno from emp group by deptno;
```

2. 列出emp表中各部门job 含’REP’的员工的部门号，最低工资，最高工资

```
select  max(sal) as 最高工资,

        min(sal) as 最低工资,

        deptno as 部门号

from emp

where job like '%REP%'

group by deptno;
```

3. 对于emp中最低工资小于7000的部门中job为'SA_REP'的员工的部门号，最低工资，最高工资

```
select max(sal) as 最高工资,

       min(sal) as 最低工资,

       deptno as 部门号

from emp  b

where job='SA_REP' and 7000> (
    select min(sal)

    from emp  a

    where a.deptno=b.deptno)

    group by b.deptno
```

4. 写出对上题的另一解决方法（请补充）

```
select deptno,min(sal),max(sal)

from emp

where job = 'SA_REP' and deptno in (

    select deptno

    from emp

    --group by deptno

    having min(sal) < 7000

)

group by deptno
```

5. 根据部门号由高而低，工资由低而高列出每个员工的姓名，部门号，工资

```
select deptno as 部门号,ename as 姓名,sal as 工资

from emp

order by deptno desc,sal asc
```

6. 列出'Abel'所在部门中每个员工的姓名与部门号

```
#方法一

select ename,deptno

from emp

where deptno = (select deptno from emp where ename = 'Abel')


#方法二

select ename,deptno

from emp e1
```

```
where exists (

       select 'x'

       from emp e2

       where e1.deptno = e2.deptno

       and e2.ename = 'Abel'

)
```

7. 列出每个员工的姓名，工作，部门号，部门名

```
select ename,job,emp.deptno,dept.dname

from emp,dept

where emp.deptno=dept.deptno
```

8. 列出emp中工作为'SH_CLERK'的员工的姓名，工作，部门号，部门名

```
select ename,job,dept.deptno,dname

from emp,dept

where dept.deptno=emp.deptno and job='SH_CLERK'
```

9. 对于emp中有管理者的员工，列出姓名，管理者姓名（管理者外键为mgr）

```
select a.ename as 姓名,b.ename as 管理者

from emp  a,emp b

where a.mgr is not null and a.mgr=b.empno
```

10. 对于dept表中，列出所有部门名，部门号，同时列出各部门工作为'SH_CLERK'的员工名与工作

```
select dname as 部门名,dept.deptno as 部门号,ename as 员工名,job as 工作

from dept,emp

where dept.deptno = emp.deptno(+) and job = 'SH_CLERK'
```

11. 对于工资高于本部门平均水平的员工，列出部门号，姓名，工资，按部门号排序

```
#方法一

select a.deptno as 部门号,a.ename as 姓名,a.sal as 工资

from emp  a

where a.sal>(

    select avg(sal)

    from emp

    b where a.deptno=b.deptno)

    order by a.deptno


#方法二

select e.deptno,ename,sal

from emp e,(select deptno,avg(sal) avg_sal from emp group by deptno) b

where e.sal > b.avg_sal and e.deptno = b.deptno
```

12. 对于emp，列出各个部门中工资高于本部门平均水平的员工数和部门号，按部门号排序

```
select count(a.sal) as 员工数,a.deptno  部门号

from emp  a

where a.sal>(select avg(sal) from emp b where a.deptno=b.deptno)

group by a.deptno

order by a.deptno
```

13. 对于emp中工资高于本部门平均水平，人数多于1人的，列出部门号，高于部门平均工资的人数，按部门号排序

```
select *

from(

select deptno,count(*) count_num

from emp e

where sal > (

   select avg(sal)
   from emp e1

   where e.deptno = e1.deptno

)

group by deptno

) e1

where e1.count_num > 1

order by e1.deptno
```

14. 对于emp中工资高于本部门平均水平，且其人数多于3人的，列出部门号，部门人数，按部门号排序

```
#方法一

select count(a.empno) as 员工数,a.deptno as 部门号,avg(sal) as 平均工资

from emp a

where (
    select count(c.empno)

    from emp  c

    where c.deptno=a.deptno and c.sal>(

        select avg(sal)

        from emp b

        where c.deptno=b.deptno)
    )>3

group by a.deptno order by a.deptno


#方法二

select m.deptno,count(ee1.empno)

from(

select e1.deptno,count(empno) count_num

from emp e1

where e1.sal >

(select avg(sal) from emp e2 where e1.deptno = e2.deptno)

group by e1.deptno
) m,emp ee1

where m.count_num > 3 and m.deptno = ee1.deptno

group by m.deptno
```

15. 对于emp中低于自己工资至少5人的员工，列出其部门号，姓名，工资，以及工资少于自己的人数

```
select a.deptno,a.ename,a.sal,(

    select count(b.ename)

    from emp as b

    where b.sal<a.sal) as 人数

from emp as a

where (select count(b.ename) from emp as b where b.sal<a.sal)>5
```

综合练习2：帮MM管理员工档案

此套练习供《尚硅谷-MySQL数据库基础篇》使用

需求

你好，我是一名刚毕业的大学生，现在入职到了一家IT企业做HR。可是我不懂电脑，你能帮我管理公司的人事档案数据吗？

需求1：公司要求，员工档案要包括以下这些信息：**编号**，**姓名**，**工资**，**生日**

需求2：怎么样？数据库表创建好了？那麻烦你帮我把下面这些数据保存起来吧？

姓名	**工资**	**生日**
马云	2025.33	1973-8-12
李彦宏	3209.49	1986-7-14
马化腾	1436.12	1964-8-10

需求3：呀！对不起，我忘记了，表格中还需要保存“**手机号**”！能修改一下表格吗？

需求4：呀！对不起，我又忘记了，公司还需要维护“部门”数据，同时记录每个员工是属于哪个部门的！

需求5：有一位同事辞职了，请帮我把他从系统中删除吧！他的员工编号是：5

需求6：有一位同事涨工资了，涨了200块钱，同时他手机号也改了，新的手机号是：13586705312。请帮我改一下吧，你真是个好人！这位同事的编号是17。

需求7：公司要打印报表，请帮我把全部信息都打印出来吧！

需求8：有同事要补办工牌，请帮我把他的全部信息都调取出来，他的编号是：63

需求9：Linda快过生日了，帮我查一下她生日的具体日期和手机号吧！

需求10：公司要调查薪酬情况，请帮我查询一下工资在2000到5000之间的员工信息，以及工资在3000以上的人数！

需求11：听说有些同事的工资正好是1000、3000或5000，帮我查查他们是谁吧？

需求12：公司开年会，要让名字里有字母o的同事表演节目，帮我查一下吧！

需求13：糟糕，有些同事的手机号是空的，帮我查询一下是哪些人吧！

需求14：市场部的主管想了解他们部门员工的工资，帮我查一下吧！哦，对了，要按顺序显示哦！市场部的部门名称是：Sales

需求15：上述查询返回的记录太多了，查看起来很不方便，怎么样能够实现分页查询呢？

答案

```
#需求1
CREATE TABLE emp(
    emp_id INT AUTO_INCREMENT PRIMARY KEY,
    emp_name VARCHAR(25),
    salary DOUBLE(10, 2),
    birthday DATE
);

SELECT * FROM emp;

#需求2
INSERT INTO emp(emp_name, salary, birthday)
VALUES('马云', 2025.33, '1973-8-12');

INSERT INTO emp(emp_name, salary, birthday)
VALUES('李彦宏', 3000.22, '1987-6-5');

INSERT INTO emp(emp_name, salary, birthday)
VALUES('马化腾', 5555.55, '1956-6-6');

#需求3
ALTER TABLE emp
ADD telephone VARCHAR(30);

#需求4
CREATE TABLE depart(
    depart_id INT AUTO_INCREMENT PRIMARY KEY,
    depart_name VARCHAR(30)
);

SELECT * FROM depart;

ALTER TABLE emp
ADD depart_id_fk INT;

ALTER TABLE emp
ADD CONSTRAINT emp_depart_id_fk FOREIGN KEY(depart_id_fk) REFERENCES
depart(depart_id);

SELECT * FROM emp;
SELECT * FROM depart;

#需求5
DELETE FROM emp
WHERE emp_id = 5;

#需求6
UPDATE emp
SET salary = salary + 200, telephone = '13586705312'
WHERE emp_id = 17;

#需求7：略

#需求8
```

```
SELECT *
FROM emp
WHERE emp_id = 63;

#需求9
SELECT birthday, telephone
FROM emp
WHERE emp_name = 'Linda';

#需求10
SELECT *
FROM emp
WHERE salary BETWEEN 2000 AND 5000;

SELECT COUNT(*)
FROM emp
WHERE salary > 3000;

#需求11
SELECT *
FROM emp
WHERE salary IN(1000, 3000, 5000);

#需求12
SELECT *
FROM emp
WHERE emp_name LIKE '%o%';

#需求13
SELECT *
FROM emp
WHERE telephone IS NULL;

#需求14
SELECT salary, depart_name
FROM emp
JOIN depart
ON emp.`depart_id_fk` = depart.`depart_id`
WHERE depart_name = 'Sales'
ORDER BY salary DESC;

#需求15
SELECT *
FROM emp
LIMIT 0, 10;
```

补充数据：

emp表的数据

```
INSERT INTO emp (emp_name,salary,birthday,telephone,depart_id_fk) VALUES
('Zenobia',1847.74,'1994-10-08','18811769371',1);
INSERT INTO emp (emp_name,salary,birthday,telephone,depart_id_fk) VALUES
('Yvette',2578.90,'1992-06-11','18811769325',2);
INSERT INTO emp (emp_name,salary,birthday,telephone,depart_id_fk) VALUES
('Xaviera',4438.63,'1986-09-21',NULL,3);
INSERT INTO emp (emp_name,salary,birthday,telephone,depart_id_fk) VALUES
('Winni',2545.94,'1975-09-29','18811769305',4);
INSERT INTO emp (emp_name,salary,birthday,telephone,depart_id_fk) VALUES
('Winifred',2509.29,'1983-10-12',NULL,5);
INSERT INTO emp (emp_name,salary,birthday,telephone,depart_id_fk) VALUES
('Vivien',5592.78,'1980-07-19','18811769315',6);
INSERT INTO emp (emp_name,salary,birthday,telephone,depart_id_fk) VALUES
('Violet',1000,'1978-12-01','18811769201',7);
INSERT INTO emp (emp_name,salary,birthday,telephone,depart_id_fk) VALUES
('Veromca',2245.30,'1972-05-24',NULL,8);
INSERT INTO emp (emp_name,salary,birthday,telephone,depart_id_fk) VALUES
('Vanessa',9998.74,'1983-05-23',NULL,9);
INSERT INTO emp (emp_name,salary,birthday,telephone,depart_id_fk) VALUES
('Ursula',6857.09,'1980-12-31','18811769132',10);
INSERT INTO emp (emp_name,salary,birthday,telephone,depart_id_fk) VALUES
('Theresa',8542.15,'1971-08-09','18811769135',11);
INSERT INTO emp (emp_name,salary,birthday,telephone,depart_id_fk) VALUES
('Tammy',3000,'1973-05-02','18811768752',12);
INSERT INTO emp (emp_name,salary,birthday,telephone,depart_id_fk) VALUES
('Stacey',5000,'1985-01-18','18811768753',13);
INSERT INTO emp (emp_name,salary,birthday,telephone,depart_id_fk) VALUES
('Setlla',8421.29,'1994-06-17',NULL,14);
INSERT INTO emp (emp_name,salary,birthday,telephone,depart_id_fk) VALUES
('Shirley',9958.03,'1984-06-20','18811768715',15);
INSERT INTO emp (emp_name,salary,birthday,telephone,depart_id_fk) VALUES
('Sebastiane',4246.59,'1992-05-21','18811768723',16);
INSERT INTO emp (emp_name,salary,birthday,telephone,depart_id_fk) VALUES
('Ruby',3000,'1976-04-20','18811768725',17);
INSERT INTO emp (emp_name,salary,birthday,telephone,depart_id_fk) VALUES
('Roberta',6172.17,'1976-06-07','18811768675',18);
INSERT INTO emp (emp_name,salary,birthday,telephone,depart_id_fk) VALUES
('Renata',8814.89,'1986-05-04','18811768650',19);
INSERT INTO emp (emp_name,salary,birthday,telephone,depart_id_fk) VALUES
('Rachel',6990.94,'1991-02-27','18811768673',20);
INSERT INTO emp (emp_name,salary,birthday,telephone,depart_id_fk) VALUES
('Quintina',9098.04,'1985-07-22','18811768632',21);
INSERT INTO emp (emp_name,salary,birthday,telephone,depart_id_fk) VALUES
('Poppy',1000,'1974-07-20','18811768635',22);
INSERT INTO emp (emp_name,salary,birthday,telephone,depart_id_fk) VALUES
('Phyllis',4293.83,'1975-04-23','18811768631',23);
INSERT INTO emp (emp_name,salary,birthday,telephone,depart_id_fk) VALUES
('Phoenix',9562,'1986-08-29','18811768613',24);
INSERT INTO emp (emp_name,salary,birthday,telephone,depart_id_fk) VALUES
('Phoebe',7860.54,'1994-11-07','18811768621',25);
INSERT INTO emp (emp_name,salary,birthday,telephone,depart_id_fk) VALUES
('Philipppa',9572.94,'1984-10-28','18811768593',1);
INSERT INTO emp (emp_name,salary,birthday,telephone,depart_id_fk) VALUES
('Penny',4594,'1982-11-26','18811768602',2);
```

```
INSERT INTO emp (emp_name,salary,birthday,telephone,depart_id_fk) VALUES
('Penelope',2361.62,'1984-12-20','18811768395',3);
INSERT INTO emp (emp_name,salary,birthday,telephone,depart_id_fk) VALUES
('Pearl',5000,'1991-04-11','18811768397',4);
INSERT INTO emp (emp_name,salary,birthday,telephone,depart_id_fk) VALUES
('Paula',8728.85,'1993-11-18','18811768530',5);
INSERT INTO emp (emp_name,salary,birthday,telephone,depart_id_fk) VALUES
('Patricia',4137.15,'1993-08-11','18811768570',6);
INSERT INTO emp (emp_name,salary,birthday,telephone,depart_id_fk) VALUES
('Pandora',4147.66,'1981-10-25','18811768571',7);
INSERT INTO emp (emp_name,salary,birthday,telephone,depart_id_fk) VALUES
('Pamela',2338.69,'1970-04-23','18811768590',8);
INSERT INTO emp (emp_name,salary,birthday,telephone,depart_id_fk) VALUES
('Ophelia',5049.5,'1984-04-24','18811768591',9);
INSERT INTO emp (emp_name,salary,birthday,telephone,depart_id_fk) VALUES
('Olivia',4103.37,'1994-03-05','18811768392',10);
INSERT INTO emp (emp_name,salary,birthday,telephone,depart_id_fk) VALUES
('Olive',8781.14,'1975-11-20','18701368566',11);
INSERT INTO emp (emp_name,salary,birthday,telephone,depart_id_fk) VALUES
('Olga',2118.31,'1974-11-06','18701368699',12);
INSERT INTO emp (emp_name,salary,birthday,telephone,depart_id_fk) VALUES
('Nydia',3000,'1979-12-23','18701371299',13);
INSERT INTO emp (emp_name,salary,birthday,telephone,depart_id_fk) VALUES
('Novia',5996.61,'1980-12-23','18701373066',1);
INSERT INTO emp (emp_name,salary,birthday,telephone,depart_id_fk) VALUES
('Norma',6582.16,'1990-01-15','18701376399',2);
INSERT INTO emp (emp_name,salary,birthday,telephone,depart_id_fk) VALUES
('Nelly',1400.57,'1978-10-05','18701575123',3);
INSERT INTO emp (emp_name,salary,birthday,telephone,depart_id_fk) VALUES
('Natividad',3168.08,'1972-02-09','18701638388',4);
INSERT INTO emp (emp_name,salary,birthday,telephone,depart_id_fk) VALUES
('Nancy',2333.92,'1973-11-12','18710051588',5);
INSERT INTO emp (emp_name,salary,birthday,telephone,depart_id_fk) VALUES
('Myrna',5000,'1972-05-02','18810659199',6);
INSERT INTO emp (emp_name,salary,birthday,telephone,depart_id_fk) VALUES
('Myra',8507.84,'1990-04-16','13501187739',7);
INSERT INTO emp (emp_name,salary,birthday,telephone,depart_id_fk) VALUES
('Monica',3711.35,'1981-01-07','13501200179',8);
INSERT INTO emp (emp_name,salary,birthday,telephone,depart_id_fk) VALUES
('Mona',7531.16,'1986-04-26','13501263679',9);
INSERT INTO emp (emp_name,salary,birthday,telephone,depart_id_fk) VALUES
('Molly',6376.71,'1971-07-09','13501265069',10);
INSERT INTO emp (emp_name,salary,birthday,telephone,depart_id_fk) VALUES
('Mignon',1252.32,'1993-02-21','13501272559',11);
INSERT INTO emp (emp_name,salary,birthday,telephone,depart_id_fk) VALUES
('Megan',7202.9,'1986-02-24','13501278633',12);
INSERT INTO emp (emp_name,salary,birthday,telephone,depart_id_fk) VALUES
('Marguerite',6313.3,'1981-01-05','13552235345',13);
INSERT INTO emp (emp_name,salary,birthday,telephone,depart_id_fk) VALUES
('Lynn',9777.82,'1974-01-15','13552290788',14);
INSERT INTO emp (emp_name,salary,birthday,telephone,depart_id_fk) VALUES
('Lydia',8568.8,'1988-04-04','13552618388',15);
INSERT INTO emp (emp_name,salary,birthday,telephone,depart_id_fk) VALUES
('Louise',3547.9,'1987-03-09','13552623288',16);
INSERT INTO emp (emp_name,salary,birthday,telephone,depart_id_fk) VALUES
('Linda',7963.88,'1974-01-04','13552625111',17);
INSERT INTO emp (emp_name,salary,birthday,telephone,depart_id_fk) VALUES
('Kristin',6140.55,'1976-04-03','13552795788',18);
```

```
INSERT INTO emp (emp_name,salary,birthday,telephone,depart_id_fk) VALUES
('Kelly',6517.5,'1976-08-31','13552977388',19);
INSERT INTO emp (emp_name,salary,birthday,telephone,depart_id_fk) VALUES
('Katherine',2587.49,'1982-11-27','13601172019',20);
INSERT INTO emp (emp_name,salary,birthday,telephone,depart_id_fk) VALUES
('Juliet',4776.88,'1984-05-08','13651300588',21);
INSERT INTO emp (emp_name,salary,birthday,telephone,depart_id_fk) VALUES
('Judith',3976.65,'1985-12-06','13651300788',22);
INSERT INTO emp (emp_name,salary,birthday,telephone,depart_id_fk) VALUES
('Josephine',7318.35,'1974-10-25','13671365788',23);
INSERT INTO emp (emp_name,salary,birthday,telephone,depart_id_fk) VALUES
('Jessica',4740.74,'1974-03-24','13681279088',24);
INSERT INTO emp (emp_name,salary,birthday,telephone,depart_id_fk) VALUES
('Janice',2185.81,'1971-07-12','13683551077',25);
INSERT INTO emp (emp_name,salary,birthday,telephone,depart_id_fk) VALUES
('Jacqueline',8268.45,'1975-07-15','13683553211',20);
INSERT INTO emp (emp_name,salary,birthday,telephone,depart_id_fk) VALUES
('Irma',2470.36,'1992-04-22','13683553799',21);
INSERT INTO emp (emp_name,salary,birthday,telephone,depart_id_fk) VALUES
('Ida',8519.9,'1989-06-07','13683555722',22);
INSERT INTO emp (emp_name,salary,birthday,telephone,depart_id_fk) VALUES
('Hilary',6218.39,'1978-06-10','13683561077',23);
INSERT INTO emp (emp_name,salary,birthday,telephone,depart_id_fk) VALUES
('Hermosa',1592.84,'1976-12-07','13683563277',24);
INSERT INTO emp (emp_name,salary,birthday,telephone,depart_id_fk) VALUES
('Heloise',7464.97,'1991-07-11','13683565177',25);
INSERT INTO emp (emp_name,salary,birthday,telephone,depart_id_fk) VALUES
('Helen',6004.4,'1972-02-10','13683575977',7);
INSERT INTO emp (emp_name,salary,birthday,telephone,depart_id_fk) VALUES
('Harriet',1161.67,'1971-07-31','13683578699',8);
INSERT INTO emp (emp_name,salary,birthday,telephone,depart_id_fk) VALUES
('Gwendolyn',6138.14,'1974-04-23','13683579077',9);
INSERT INTO emp (emp_name,salary,birthday,telephone,depart_id_fk) VALUES
('Gustave',6119.81,'1982-02-13','13683580766',10);
INSERT INTO emp (emp_name,salary,birthday,telephone,depart_id_fk) VALUES
('Griselda',8268.45,'1994-02-17','13683581977',11);
INSERT INTO emp (emp_name,salary,birthday,telephone,depart_id_fk) VALUES
('Gloria',2470.36,'1992-11-23','13683582766',12);
INSERT INTO emp (emp_name,salary,birthday,telephone,depart_id_fk) VALUES
('Geraldine',8519.9,'1994-07-02','13683591877',13);
INSERT INTO emp (emp_name,salary,birthday,telephone,depart_id_fk) VALUES
('Frederica',6218.39,'1970-07-18','13683592677',14);
INSERT INTO emp (emp_name,salary,birthday,telephone,depart_id_fk) VALUES
('Freda',1592.84,'1985-06-12','13683598177',15);
INSERT INTO emp (emp_name,salary,birthday,telephone,depart_id_fk) VALUES
('Frances',7464.97,'1977-10-12','13683598766',16);
INSERT INTO emp (emp_name,salary,birthday,telephone,depart_id_fk) VALUES
('Florence',6004.4,'1988-08-29','13683601577',17);
INSERT INTO emp (emp_name,salary,birthday,telephone,depart_id_fk) VALUES
('Flora',1161.67,'1977-09-24','13683603177',18);
INSERT INTO emp (emp_name,salary,birthday,telephone,depart_id_fk) VALUES
('Evelyn',6138.14,'1983-08-23','13683608177',19);
INSERT INTO emp (emp_name,salary,birthday,telephone,depart_id_fk) VALUES
('Evangeline',6119.81,'1978-10-28','13683613599',20);
INSERT INTO emp (emp_name,salary,birthday,telephone,depart_id_fk) VALUES
('Eunice',6704.9,'1980-02-27','13683613877',21);
INSERT INTO emp (emp_name,salary,birthday,telephone,depart_id_fk) VALUES
('Eudora',6621.36,'1992-12-14','13683617566',22);
```

```
INSERT INTO emp (emp_name,salary,birthday,telephone,depart_id_fk) VALUES
('Erin',3000,'1974-01-11','13683619566',23);
INSERT INTO emp (emp_name,salary,birthday,telephone,depart_id_fk) VALUES
('Emma',7139.4,'1972-12-03','13683621266',24);
INSERT INTO emp (emp_name,salary,birthday,telephone,depart_id_fk) VALUES
('Emily',6156.79,'1971-03-10','13691056588',25);
INSERT INTO emp (emp_name,salary,birthday,telephone,depart_id_fk) VALUES
('Elizabeth',8520.18,'1981-03-20','13701056332',15);
INSERT INTO emp (emp_name,salary,birthday,telephone,depart_id_fk) VALUES
('Edwina',7512.87,'1972-01-22','13701112057',16);
INSERT INTO emp (emp_name,salary,birthday,telephone,depart_id_fk) VALUES
('Dorothy',4151.84,'1973-09-22','13701117057',17);
INSERT INTO emp (emp_name,salary,birthday,telephone,depart_id_fk) VALUES
('Doreen',8053.19,'1992-01-15','13701117163',18);
INSERT INTO emp (emp_name,salary,birthday,telephone,depart_id_fk) VALUES
('Dolores',3004.19,'1984-03-11','13701117613',19);
INSERT INTO emp (emp_name,salary,birthday,telephone,depart_id_fk) VALUES
('Delia',3367.42,'1987-03-22','13701118317',20);
INSERT INTO emp (emp_name,salary,birthday,telephone,depart_id_fk) VALUES
('Cynthia',3175.62,'1987-10-31','13701133371',21);
INSERT INTO emp (emp_name,salary,birthday,telephone,depart_id_fk) VALUES
('Cornelia',7760.89,'1985-01-19','13701277781',22);
INSERT INTO emp (emp_name,salary,birthday,telephone,depart_id_fk) VALUES
('Constance',5712.07,'1985-02-05','13701322150',23);
INSERT INTO emp (emp_name,salary,birthday,telephone,depart_id_fk) VALUES
('Clementine',3000,'1988-09-13','13716791688',24);
INSERT INTO emp (emp_name,salary,birthday,telephone,depart_id_fk) VALUES
('Clara',8238.39,'1975-12-22','13717827188',25);
INSERT INTO emp (emp_name,salary,birthday,telephone,depart_id_fk) VALUES
('Claire',5000,'1985-07-05','13717935188',4);
INSERT INTO emp (emp_name,salary,birthday,telephone,depart_id_fk) VALUES
('Christine',1411.58,'1988-12-13','13717951688',5);
INSERT INTO emp (emp_name,salary,birthday,telephone,depart_id_fk) VALUES
('Chloe',9354.48,'1981-08-05','13718370588',6);
INSERT INTO emp (emp_name,salary,birthday,telephone,depart_id_fk) VALUES
('Charlotte',6699.46,'1981-12-29','13718449123',7);
INSERT INTO emp (emp_name,salary,birthday,telephone,depart_id_fk) VALUES
('Caroline',891.62,'1974-01-28','13718928388',8);
INSERT INTO emp (emp_name,salary,birthday,telephone,depart_id_fk) VALUES
('Brook',618.19,'1980-12-22','13720010388',9);
INSERT INTO emp (emp_name,salary,birthday,telephone,depart_id_fk) VALUES
('Bridget',3672.85,'1985-07-01','13801066471',10);
INSERT INTO emp (emp_name,salary,birthday,telephone,depart_id_fk) VALUES
('Bblythe',1000,'1975-12-01','13801337803',11);
INSERT INTO emp (emp_name,salary,birthday,telephone,depart_id_fk) VALUES ('Bella
',4870.36,'1984-04-05','13901252053',12);
INSERT INTO emp (emp_name,salary,birthday,telephone,depart_id_fk) VALUES
('Audrey',3000,'1994-08-20','15001010233',13);
INSERT INTO emp (emp_name,salary,birthday,telephone,depart_id_fk) VALUES
('Athena',5000,'1980-09-02','15001010766',14);
INSERT INTO emp (emp_name,salary,birthday,telephone,depart_id_fk) VALUES
('Atalanta',3175.62,'1983-06-07','15001011233',15);
INSERT INTO emp (emp_name,salary,birthday,telephone,depart_id_fk) VALUES
('Astrid',7760.89,'1994-03-20','15001011800',16);
INSERT INTO emp (emp_name,salary,birthday,telephone,depart_id_fk) VALUES
('Antonia',5712.07,'1982-03-22','15001012122',17);
INSERT INTO emp (emp_name,salary,birthday,telephone,depart_id_fk) VALUES
('Annabelle',344.6,'1976-02-13','15001012199',18);
```

```
INSERT INTO emp (emp_name,salary,birthday,telephone,depart_id_fk) VALUES
('Angela',8238.39,'1981-12-17','15001035266',19);
INSERT INTO emp (emp_name,salary,birthday,telephone,depart_id_fk) VALUES
('Amelia',1000,'1982-06-09','15001035366',20);
INSERT INTO emp (emp_name,salary,birthday,telephone,depart_id_fk) VALUES
('Adelaide',3672.85,'1974-05-22','15001036266',21);
```

dept表数据

```
INSERT INTO `depart`(`depart_name`) VALUES ('Administration');
INSERT INTO `depart`(`depart_name`) VALUES ('Marketing');
INSERT INTO `depart`(`depart_name`) VALUES ('Purchasing');
INSERT INTO `depart`(`depart_name`) VALUES ('Human Resources');
INSERT INTO `depart`(`depart_name`) VALUES ('Shipping');
INSERT INTO `depart`(`depart_name`) VALUES ('IT');
INSERT INTO `depart`(`depart_name`) VALUES ('Public Relations');
INSERT INTO `depart`(`depart_name`) VALUES ('Sales');
INSERT INTO `depart`(`depart_name`) VALUES ('Executive');
INSERT INTO `depart`(`depart_name`) VALUES ('Finance');
INSERT INTO `depart`(`depart_name`) VALUES ('Accounting');
INSERT INTO `depart`(`depart_name`) VALUES ('Treasury');
INSERT INTO `depart`(`depart_name`) VALUES ('Corporate Tax');
INSERT INTO `depart`(`depart_name`) VALUES ('Control And Credit');
INSERT INTO `depart`(`depart_name`) VALUES ('Shareholder Services');
INSERT INTO `depart`(`depart_name`) VALUES ('Benefits');
INSERT INTO `depart`(`depart_name`) VALUES ('Manufacturing');
INSERT INTO `depart`(`depart_name`) VALUES ('Construction');
INSERT INTO `depart`(`depart_name`) VALUES ('Contracting');
INSERT INTO `depart`(`depart_name`) VALUES ('Operations');
INSERT INTO `depart`(`depart_name`) VALUES ('IT Support');
INSERT INTO `depart`(`depart_name`) VALUES ('Government Sales');
INSERT INTO `depart`(`depart_name`) VALUES ('Retail Sales');
INSERT INTO `depart`(`depart_name`) VALUES ('Recruiting');
INSERT INTO `depart`(`depart_name`) VALUES ('Payroll');
```

综合练习3

此套练习供《尚硅谷-MySQL数据库基础篇》使用

测 试

1．创建表

表名	member						
列名	MEMBER_ID	LAST_NAME	LAST_NAME	ADDRESS	CITY	PHONE	JOIN_DATE
主键	yes						
非空	yes	yes					yes
唯一	yes						
默认值							sysdate
数据类型	number	varchar	varchar	varchar	varchar	varchar	date
长度	10	25	25	100	30	15	

2．创建表

表名	title					
列名	TITLE_ID	TITLE	DESCRIPTION	RATING	CATEGORY	RELEASE_DATE
主键	yes					
非空	yes	yes	yes			
唯一	yes					
检查				G, PG, R,NC17, NR	DRAMA,COMEDY,ACTION, CHILD,SCIFI,DOCUMENTARY	
数据类型	number	varchar2	varchar2	varchar2	varchar2	date
长度	10	60	400	4	20	

3．创建表

表名	TITLE_COPY		
列名	COPY_ID	TITLE_id	status
主键	yes	yes	
外键		yes(title.title_id)	
非空	yes	yes	yes
唯一	yes	yes	
检查			
数据类型	number	number	varchar2
长度	10	10	15

4．创建表

表名	rental					
列名	book_date	Member_id	Copy_id	Ac_ret_date	Exp_ret_dte	Title_id
主键	yes	yes	yes			yes
外键		yes(member.member_id)	yes(title_copy.copy_id)			yes(title_copy.title_id)
默认值	sysdate				sysdate + 2	
数据类型	date	number	number	date	date	date
长度		10	10			

5．创建表

表名	RESERVATION		
列名	res_date	member_id	Title_id
主键	yes	yes	yes
外键		yes(member.member_id)	yes(title.title_id)
非空	yes	yes	yes
唯一	yes	yes	
检查			
数据类型	date	number	number
长度		10	10

6. 查询数据字典视图user_tables和user_constraints以确认所创建的对象

7．创建序列MEMBER_ID_SEQ.，由101开始，每次增长1，无最大值限制，不放入内存

8．创建序列title_ID_SEQ.，由101开始，每次增长1，无最大值限制，不放入内存

9．向表title中添加数据

Title	*Description*	*Rating*	*Category*	*Release_date*
Willie andChristmasToo	All of Willie’s friendsmake a Christmas list forSanta, but Willie has yet to add his own wish list.	G	CHILD	05-OCT-1995
Alien Again	Yet another installation ofscience fiction history. Canthe heroine save the planetfrom the alien life form?	R	SCIFI	19-MAY-1995
The Glob	A meteor crashes near asmall American town andunleashes carnivorous gooin this classic.	NR	SCIFI	12-AUG-1995
My Day Off	With a little luck and a lotof ingenuity, a teenagerskips school for a day inNew York	PG	COMEDY	12-JUL-1995
Miracles onIce	A six-year-old has doubtsabout Santa Claus, but shediscovers that miraclesreally do exist.	PG	DRAMA	12-SEP-1995
Soda Gang	After discovering a cacheof drugs, a young couplefind themselves pittedagainst a vicious gang.	NR	ACTION	01-JUN-1995

10．向表member中添加数据

First_Name	*Last_Name*	*Address*	*City*	*Phone*	*Join_Date*
Carmen	Velasquez	283 King Street	Seattle	206-899-6666	08-MAR-1990
LaDoris	Ngao	5 Modrany	Bratislava	586-355-8882	08-MAR-1990
Midori	Nagayama	68 Via Centrale	Sao Paolo	254-852-5764	17-JUN-1991
Mark	Quick-to-See	6921 KingWay	Lagos	63-559-7777	07-APR-1990
Audry	Ropeburn	86 Chu Street	Hong Kong	41-559-87	18-JAN-1991
Molly	Urguhart	3035 Laurier	Quebec	418-542-9988	18-JAN-1991

11.向表title_copy中插入数据

```
INSERT INTO title_copy(copy_id, title_id, status)

VALUES (1, 92, 'AVAILABLE');

INSERT INTO title_copy(copy_id, title_id, status)

VALUES (1, 93, 'AVAILABLE');

INSERT INTO title_copy(copy_id, title_id, status)

VALUES (2, 93, 'RENTED');

INSERT INTO title_copy(copy_id, title_id, status)

VALUES (1, 94, 'AVAILABLE');

INSERT INTO title_copy(copy_id, title_id, status)

VALUES (1, 95, 'AVAILABLE');

INSERT INTO title_copy(copy_id, title_id,status)

VALUES (2, 95, 'AVAILABLE');

INSERT INTO title_copy(copy_id, title_id,status)

VALUES (3, 95, 'RENTED');

INSERT INTO title_copy(copy_id, title_id,status)

VALUES (1, 96, 'AVAILABLE');

INSERT INTO title_copy(copy_id, title_id,status)

VALUES (1, 97, 'AVAILABLE');
```

12.向表rental中插入数据

```
INSERT INTO rental(title_id, copy_id, member_id,

book_date, exp_ret_date, act_ret_date)

VALUES (92, 1, 101, sysdate-3, sysdate-1, sysdate-2);

INSERT INTO rental(title_id, copy_id, member_id,

book_date, exp_ret_date, act_ret_date)

VALUES (93, 2, 101, sysdate-1, sysdate-1, NULL);

INSERT INTO rental(title_id, copy_id, member_id,

book_date, exp_ret_date, act_ret_date)

VALUES (95, 3, 102, sysdate-2, sysdate, NULL);
```

```
INSERT INTO rental(title_id, copy_id, member_id,

book_date, exp_ret_date,act_ret_date)

VALUES (97, 1, 106, sysdate-4, sysdate-2, sysdate-2);

COMMIT;
```

13. 创建视图并查询视图中的数据

```
CREATE VIEW title_avail AS

SELECT t.title, c.copy_id, c.status, r.exp_ret_date

FROM title t, title_copy c, rental r

WHERE t.title_id = c.title_id

AND c.copy_id = r.copy_id(+)

AND c.title_id = r.title_id(+);
```

14. 插入下列数据

```
INSERT INTO title(title_id, title, description, rating,

category, release_date)

VALUES (title_id_seq.NEXTVAL, 'Interstellar Wars',

'Futuristic interstellar action movie. Can the

rebels save the humans from the evil Empire?',

'PG', 'SCIFI', '07-JUL-77');

INSERT INTO title_copy (copy_id, title_id, status)

VALUES (1, 98, 'AVAILABLE');

INSERT INTO title_copy (copy_id, title_id, status)

VALUES (2, 98, 'AVAILABLE');

INSERT INTO reservation (res_date, member_id, title_id)

VALUES (SYSDATE, 101, 98);

INSERT INTO reservation (res_date, member_id, title_id)

VALUES (SYSDATE, 104, 97);

INSERT INTO rental(title_id, copy_id, member_id)

VALUES (98,1,101);
```

15．更新下列数据

```
UPDATE title_copy

SET status= 'RENTED'

WHERE title_id = 98

AND copy_id = 1;

DELETE

FROM reservation

WHERE member_id = 101;
```

16．查询视图title_avail中的所有数据

17．向表title中加入新的列price， 属性为NUMBER(8,2)

18．向price列中插入数据（使用变量）

Title	*Price*
Willie and Christmas Too	25
Alien Again	35
The Glob	35
My Day Off	35
Miracles on Ice	30
Soda Gang	35
Interstellar Wars	29

19．在新列price中加入非空约束

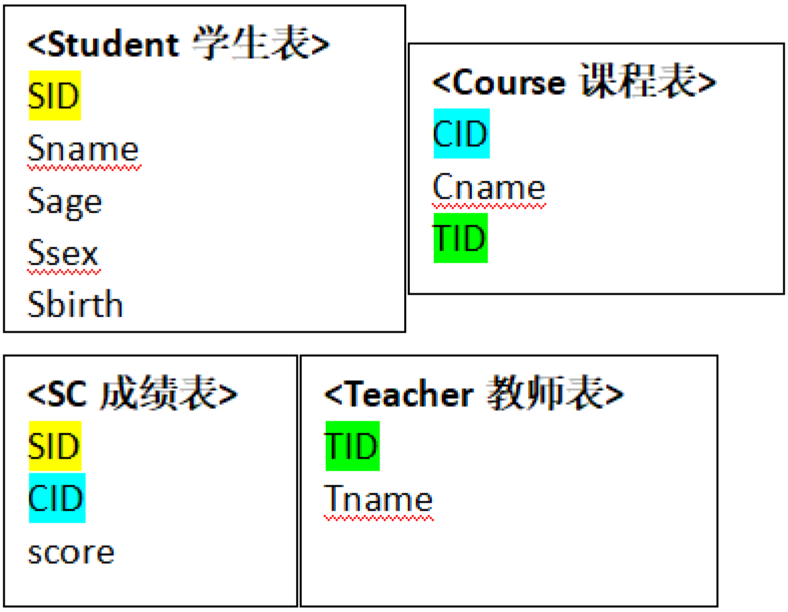
20．检验刚才所作的修正

综合练习4

此套练习供《尚硅谷-MySQL数据库基础篇》使用

数据表：

- **Student(SID, Sname, Sage, Ssex,Sbirth) 学生表**
- **Course(CID, Cname, TID) 课程表**
- **SC(SID, CID, score) 成绩表**
- **Teacher(TID, Tname) 教师表**



<Student 学生表>
SID
Sname
Sage
Ssex
Sbirth

<Course 课程表>
CID
Cname
TID

<SC 成绩表>
SID
CID
score

<Teacher 教师表>
TID
Tname

问题：

1、查询“201”课程比“202”课程成绩高的所有学生的学号;

```
select a.SID

from (select Sid,score from SC where CID=201) a,

     (select Sid,score from SC where CID=202) b

where a.score>b.score and a.Sid=b.Sid;
```

2、查询平均成绩大于“60”分的同学的学号和平均成绩;

```
select SID,avg(score)

from sc

group by SID having avg(score) >60;
```

3、查询“所有”同学的学号、姓名、选课数、总成绩;

```
select Student.SID,Student.Sname,count(SC.CID),sum(score)

from Student left Outer join SC on Student.SID=SC.SID

group by Student.SID,Sname
```

4、查询姓“李”的老师的个数;

```
select count(distinct(Tname))

from Teacher

where Tname like '李%';
```

5、查询没学过“叶平”老师课的同学的学号、姓名;

```
#方法一

select Student.SID,Student.Sname

from Student

where SID not in (

    select distinct(SC.SID)

    from SC,Course,Teacher

    where SC.CID=Course.CID and Teacher.TID=Course.TID and Teacher.Tname='叶平'

);


#方法二

select student.sid,student.sname

from student

where sid not in (

   select sid

   from sc

   where cid in (

       select cid

       from course

       where tid = (

            select tid
            from teacher

            where tname = '叶平'
       )

   )

)
```

6、查询学过“201”并且也学过编号“202”课程的同学的学号、姓名;

```
select Student.SID,Student.Sname

from Student,SC

where Student.SID=SC.SID and SC.CID='001'and exists(

    select * from SC as SC_2

    where SC_2.SID=SC.SID and SC_2.CID='002'

);
```

7、查询学过“叶平”老师所教的“所有课”的同学的学号、姓名;

```
select SID,Sname

from Student

where SID in (

    select SID from SC ,Course ,Teacher

    where SC.CID=Course.CID and Teacher.TID=Course.TID and Teacher.Tname='叶平'

    group by SID having count(SC.CID)=(

            select count(CID) from Course,Teacher

            where Teacher.TID=Course.TID and Tname='叶平'

    )

);
```

8、查询课程编号“202”的成绩比课程编号“201”课程低的所有同学的学号、姓名;

```
select SID,Sname from (

    select Student.SID,Student.Sname,score , (

        select score
        from SC SC_2

        where SC_2.SID=Student.SID and SC_2.CID='002'

    ) score2

    from Student,SC

    where Student.SID=SC.SID and CID='001'
) S_2

where score2 < score;
```

9、查询“所有课程成绩”小于60分的同学的学号、姓名；

（取反操作处理）

```
select SID,Sname

from Student

where SID not in (

    select Student.SID

    from Student,SC

    where S.SID=SC.SID and score>60

);
```

10、查询没有学全所有课的同学的学号、姓名;

（count(CID)得到课程的数目）

```
  select Student.SID,Student.Sname

  from Student,SC

  where Student.SID=SC.SID group by Student.SID,Student.Sname having

  count(CID) <(select count(CID) from Course);
```

11、查询至少有一门课与学号为“1001”的同学所学相同的同学的学号和姓名;

```
  select SID,Sname

  from Student,SC

  where Student.SID=SC.SID and CID in (

      select CID

      from SC

      where SID='1001'
  );
```

12、查询至少学过学号为“1001”同学所有一门课的其他同学学号和姓名;

```
 select distinct SC.SID,Sname

 from Student,SC

 where Student.SID=SC.SID and CID in (

     select CID

     from SC

     where SID='001'
 )

and Student.SID <> 1001;
```

13、把“SC”表中“叶平”老师教的课的成绩都更改为此课程的平均成绩;

```
update SC

set score=(

    select avg(SC_2.score)

    from SC SC_2

    where SC_2.CID=SC.CID
)

where cid = (

     select cid

     from Course,Teacher

     where Course.CID=SC.CID and Course.TID=Teacher.TID and Teacher.Tname='叶平'

)
```

14、查询和“1002”号的同学学习的课程完全相同的其他同学学号和姓名;

```
select SID

from SC

where CID in (select CID from SC where SID='1002')

group by SID having count(*)=(select count(*) from SC where SID='1002');
```

15、删除学习“叶平”老师课的SC表记录;

```
  Delete from sc

  where cid = (

        select cid

        from course ,Teacher

        where Course.CID=SC.CID and Course.TID= Teacher.TID and Tname='叶平'

  )
```

16、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、“002”号课的平均成绩;

```
Insert  into SC

as select SID,'002',(

    Select avg(score)

    from SC where CID='002'
)

from Student

where SID not in (Select SID from SC where CID='002');
```

17、按学生平均成绩从高到低显示所有学生的“数据库”、“企业管理”、“英语”三门的课程成绩，按如下形式显示：学生ID,数据库,企业管理,英语,有效课程数,有效平均分

（默认数据库是004，企业管理是001，英语是006)

```
  SELECT SID as 学生ID

    ,(SELECT score FROM SC WHERE SC.SID=t.SID AND CID='004') AS 数据库

    ,(SELECT score FROM SC WHERE SC.SID=t.SID AND CID='001') AS 企业管理

    ,(SELECT score FROM SC WHERE SC.SID=t.SID AND CID='006') AS 英语
    ,COUNT(*) AS 有效课程数, AVG(t.score) AS 平均成绩

  FROM SC AS t

  GROUP BY SID

  ORDER BY avg(t.score)
```

18、查询各科成绩最高和最低的分，以及对应的学号：以如下形式显示：课程ID，最高分，学号，最低分，学号

```
SELECT L.CID courseID,L.score 最高分,L.sid 学号,R.score 最低分,R.sid 学号

FROM SC L ,SC R

WHERE L.CID = R.CID and

   L.score = (SELECT MAX(IL.score)

         FROM SC IL,Student IM

         WHERE L.CID = IL.CID and IM.SID=IL.SID

         GROUP BY IL.CID)

AND

   R.Score = (SELECT MIN(IR.score)

         FROM SC IR

         WHERE R.CID = IR.CID

         GROUP BY IR.CID)
```

19、查询课程号，课程名称，平均成绩和及格率，并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 t.CID AS 课程号,

        max(course.Cname)AS 课程名,

        isnull(AVG(score),0) AS 平均成绩,

        100 * SUM(CASE WHEN isnull(score,0)>=60 THEN 1 ELSE 0 END)/COUNT(*) AS 及格百分数

 FROM SC T,Course

 where t.CID=course.CID

 GROUP BY t.CID

 ORDER BY 100 * SUM(CASE WHEN isnull(score,0)>=60 THEN 1 ELSE 0 END)/COUNT(*) DESC
```

20、查询如下课程平均成绩和及格率的百分数(用"1行"显示): 企业管理（001），马克思（002），OO&UML （003），数据库（004）

```
 SELECT SUM(CASE WHEN CID ='001' THEN score ELSE 0 END)/SUM(CASE CID WHEN '001' THEN 1 ELSE 0 END) AS 企业管理平均分

    ,100 * SUM(CASE WHEN CID = '001' AND score >= 60 THEN 1 ELSE 0 END)/SUM(CASE WHEN CID = '001' THEN 1 ELSE 0 END) AS 企业管理及格百分数

    ,SUM(CASE WHEN CID = '002' THEN score ELSE 0 END)/SUM(CASE CID WHEN '002' THEN 1 ELSE 0 END) AS 马克思平均分

    ,100 * SUM(CASE WHEN CID = '002' AND score >= 60 THEN 1 ELSE 0 END)/SUM(CASE WHEN CID = '002' THEN 1 ELSE 0 END) AS 马克思及格百分数

    ,SUM(CASE WHEN CID = '003' THEN score ELSE 0 END)/SUM(CASE CID WHEN '003' THEN 1 ELSE 0 END) AS UML平均分

    ,100 * SUM(CASE WHEN CID = '003' AND score >= 60 THEN 1 ELSE 0 END)/SUM(CASE WHEN CID = '003' THEN 1 ELSE 0 END) AS UML及格百分数

    ,SUM(CASE WHEN CID = '004' THEN score ELSE 0 END)/SUM(CASE CID WHEN '004' THEN 1 ELSE 0 END) AS 数据库平均分

    ,100 * SUM(CASE WHEN CID = '004' AND score >= 60 THEN 1 ELSE 0 END)/SUM(CASE WHEN CID = '004' THEN 1 ELSE 0 END) AS 数据库及格百分数

    FROM SC
```

21、查询不同老师所教不同课程平均分从高到低显示

```
SELECT max(Z.TID) AS 教师ID,

       MAX(Z.Tname) AS 教师姓名,

       C.CID AS 课程ＩＤ,

       MAX(C.Cname) AS 课程名称,

       AVG(Score) AS 平均成绩

FROM SC AS T,Course AS C ,Teacher AS Z

WHERE T.CID=C.CID and C.TID=Z.TID

GROUP BY C.CID

ORDER BY AVG(Score) DESC
```

22、查询如下课程成绩第 3 名到第 6 名的学生成绩单：企业管理（001），马克思（002），UML （003），数据库（004）

```
  [学生ID],[学生姓名],企业管理,马克思,UML,数据库,平均成绩

  SELECT DISTINCT top 3

   SC.SID As 学生学号,

   Student.Sname AS 学生姓名 ,

   T1.score AS 企业管理,

   T2.score AS 马克思,

   T3.score AS UML,

   T4.score AS 数据库,

   ISNULL(T1.score,0) + ISNULL(T2.score,0) + ISNULL(T3.score,0) + ISNULL(T4.score,0) as 总分

   FROM Student,SC LEFT JOIN SC AS T1

           ON SC.SID = T1.SID AND T1.CID = '001'

      LEFT JOIN SC AS T2

           ON SC.SID = T2.SID AND T2.CID = '002'

      LEFT JOIN SC AS T3

           ON SC.SID = T3.SID AND T3.CID = '003'

      LEFT JOIN SC AS T4

           ON SC.SID = T4.SID AND T4.CID = '004'

   WHERE student.SID=SC.SID and

   ISNULL(T1.score,0) + ISNULL(T2.score,0) + ISNULL(T3.score,0) + ISNULL(T4.score,0)

   NOT IN

   (SELECT

      DISTINCT

      TOP 15 WITH TIES

      ISNULL(T1.score,0) + ISNULL(T2.score,0) + ISNULL(T3.score,0) + ISNULL(T4.score,0)

   FROM sc

      LEFT JOIN sc AS T1
```

```
           ON sc.SID = T1.SID AND T1.CID = 'k1'

      LEFT JOIN sc AS T2

           ON sc.SID = T2.SID AND T2.CID = 'k2'

      LEFT JOIN sc AS T3

           ON sc.SID = T3.SID AND T3.CID = 'k3'

      LEFT JOIN sc AS T4

           ON sc.SID = T4.SID AND T4.CID = 'k4'

   ORDER BY ISNULL(T1.score,0) + ISNULL(T2.score,0) + ISNULL(T3.score,0) +
ISNULL(T4.score,0) DESC);
```

23、统计各科成绩,各分数段人数:课程ID,课程名称,[100-85],[85-70],[70-60],[<60]

```
  SELECT SC.CID as 课程ID, Cname as 课程名称

    ,SUM(CASE WHEN score BETWEEN 85 AND 100 THEN 1 ELSE 0 END) AS [100 - 85]

    ,SUM(CASE WHEN score BETWEEN 70 AND 85 THEN 1 ELSE 0 END) AS [85 - 70]

    ,SUM(CASE WHEN score BETWEEN 60 AND 70 THEN 1 ELSE 0 END) AS [70 - 60]

    ,SUM(CASE WHEN score < 60 THEN 1 ELSE 0 END) AS [60 -]

  FROM SC,Course

  where SC.CID=Course.CID

  GROUP BY SC.CID,Cname;
```

24、查询学生平均成绩及其名次

```
SELECT 1+(SELECT COUNT( distinct 平均成绩)

       FROM (SELECT SID,AVG(score) AS 平均成绩

           FROM SC

         GROUP BY SID

         ) AS T1

      WHERE 平均成绩> T2.平均成绩) as 名次,

   SID as 学生学号,平均成绩

  FROM (SELECT SID,AVG(score) 平均成绩

      FROM SC
```

```
    GROUP BY SID

    ) AS T2

  ORDER BY 平均成绩desc;
```

25、查询各科成绩前三名的记录:(不考虑成绩并列情况)

```
  SELECT t1.SID as 学生ID,t1.CID as 课程ID,Score as 分数

   FROM SC t1

   WHERE score IN (SELECT TOP 3 score

       FROM SC

       WHERE t1.CID= CID

      ORDER BY score DESC

       )

   ORDER BY t1.CID;
```

26、查询每门课程被选修的学生数

```
select Cid,count(SID) from sc group by CID;
```

27、查询出只选修了一门课程的全部学生的学号和姓名

```
select SC.SID,Student.Sname,count(CID) AS 选课数

from SC ,Student

where SC.SID=Student.SID group by SC.SID ,Student.Sname having count(CID)=1;
```

28、查询男生、女生人数

```
 Select count(Ssex) as 男生人数 from Student group by Ssex having Ssex='男';

 Select count(Ssex) as 女生人数 from Student group by Ssex having Ssex='女';
```

29、查询姓“张”的学生名单

```
  SELECT Sname FROM Student WHERE Sname like '张%';
```

30、查询同名学生名单，并统计同名人数

```
  select Sname,count(*) from Student group by Sname having count(*)>1;
```

31、1981年出生的学生名单(注：Student表中Sage列的类型是datetime)

```
select Sname, CONVERT(char (11),DATEPART(year,Sage)) as age

from student

where CONVERT(char(11),DATEPART(year,Sage))='1981';
```

32、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 CID,Avg(score) from SC group by CID order by Avg(score),CID DESC ;
```

33、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 Sname,SC.SID ,avg(score)

 from Student,SC

 where Student.SID=SC.SID group by SC.SID,Sname having   avg(score)>85;
```

34、查询课程名称为“数据库”，且分数低于60的学生姓名和分数

```
select Sname,isnull(score,0)

from Student,SC,Course

where SC.SID=Student.SID and SC.CID=Course.CID and Course.Cname='数据库'and score <60;
```

35、查询所有学生的选课情况；（学号，姓名，课程编号，课程名字）

```
SELECT SC.SID,SC.CID,Sname,Cname

FROM SC,Student,Course

where SC.SID=Student.SID and SC.CID=Course.CID ;
```

36、查询任何一门课程成绩在70分以上的学号、姓名、课程编号和分数;

```
SELECT distinct student.SID,student.Sname,SC.CID,SC.score

FROM student,Sc

WHERE SC.score>=70 AND SC.SID=student.SID;
```

37、查询学生学号，以及其不及格的课程，并按课程号从大到小排列

```
select sid,Cid from sc where score <60 order by CID ;
```

38、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 SC.SID,Student.Sname

from SC,Student

where SC.SID=Student.SID and Score>80 and CID='003';
```

39、求选了课程的学生人数

```
  select count(*) from sc;
```

40、查询选修“叶平”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.Sname,score

 from Student,SC,CourseC,Teacher

 where Student.SID=SC.SID and SC.CID=C.CID and C.TID=Teacher.TID and Teacher.Tname='叶平' and SC.score=(select max(score)from SC where CID=C.CID );
```

41、查询各个课程及相应的选修人数

```
 select count(*) from sc group by CID;
```

42、查询不同课程成绩相同的学生的学号、课程号、学生成绩

```
 select distinct A.SID,B.score

 from SC A ,SC B

 where A.Score=B.Score and A.CID <>B.CID ;
```

43、查询每门功成绩最好的前两名

```
  SELECT t1.SID as 学生ID,t1.CID as 课程ID,Score as 分数

  FROM SC t1

  WHERE score IN (SELECT TOP 2 score

       FROM SC

       WHERE t1.CID= CID

       ORDER BY score DESC

  )

  ORDER BY t1.CID;
```

44、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，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 CID as 课程号,count(*) as 人数

from sc

group by CID

order by count(*) desc,Cid
```

45、检索至少选修两门课程的学生学号

```
select SID

from sc

group by Sid

having count(*) > = 2
```

46、查询全部学生都选修的课程的课程号和课程名

```
select CID,Cname

from Course

where CID in (select Cid from sc group by Cid)
```

47、查询没学过“叶平”老师讲授的任一门课程的学生姓名

```
 select Sname

 from Student

 where SID not in (

     select SID

     from Course,Teacher,SC

     where Course.TID=Teacher.TID and SC.CID=course.CID and Tname='叶平'
 );
```

48、查询两门以上不及格课程的同学的学号及其平均成绩

```
select SID,avg(isnull(score,0))

from SC where SID in (

    select SID
    from SC

    where score <60

    group by SID having count(*)>2

)
group by SID;
```