# 第三章

## 第3题

### ⑴

SELECT \* FROM S WHERE A = ‘10’

### ⑵

SELECT A,B FROM S

### ⑶

SELECT S.C, S.D, A, B, E, F

FROM S, T

WHERE S.C = T.C AND

S.D = T.D

### ⑷

SELECT S.C, S.D, A, B, E, F

FROM S, T

WHERE S.C = T.C

### ⑸

SELECT S.\*, T.\*

FROM S, T

WHERE S.A < T.E

### ⑹

SELECT S.C, S.D, T.\*

FROM S, T

## 第4题

### 创建数据库及表

USE master DROP DATABASE IF EXISTS SPJ

CREATE DATABASE SPJ

GO

USE SPJ CREATE TABLE S /\* 供应商表\*/

(

SNO CHAR(4) PRIMARY KEY, /\* 列级完整性约束条件,SNO是主码\*/

SNAME CHAR(20) NOT NULL, /\* SNAME不为空\*/

STATUS SMALLINT,

CITY CHAR(20)

);

CREATE TABLE P /\* 零件表\*/

(

PNO CHAR(4) PRIMARY KEY, /\* 列级完整性约束条件,PNO是主码\*/

PNAME CHAR(20) NOT NULL, /\* PNAME不为空\*/

COLOR CHAR(10),

WEIGHT SMALLINT

);

CREATE TABLE J /\* 工程项目表\*/

(

JNO CHAR(4) PRIMARY KEY, /\* 列级完整性约束条件,JNO是主码\*/

JNAME CHAR(20) NOT NULL, /\* JNAME不为空\*/

CITY CHAR(20)

);

CREATE TABLE SPJ /\* 供应情况表\*/

(

SNO CHAR(4) NOT NULL,

PNO CHAR(4) NOT NULL,

JNO CHAR(4) NOT NULL,

QTY INT

);

ALTER TABLE SPJ

ADD CONSTRAINT FK\_SPJ\_S\_SNO --添加约束 名称 FK\_含外键的表\_被参照表\_含外键的表中的属性

FOREIGN KEY (SNO) REFERENCES S(SNO) --外键约束，外键列名，被引用列名

ALTER TABLE SPJ

ADD CONSTRAINT FK\_SPJ\_P\_PNO --添加约束 名称 FK\_含外键的表\_被参照表\_含外键的表中的属性

FOREIGN KEY (PNO) REFERENCES P(PNO) --外键约束，外键列名，被引用列名

ALTER TABLE SPJ

ADD CONSTRAINT FK\_SPJ\_J\_JNO --添加约束 名称 FK\_含外键的表\_被参照表\_含外键的表中的属性

FOREIGN KEY (JNO) REFERENCES J(JNO) --外键约束，外键列名，被引用列名

ALTER TABLE SPJ

ADD CONSTRAINT PK\_SPJ\_SNOPNOJNO PRIMARY KEY nonclustered

(SNO,

PNO,

JNO)

INSERT INTO S (SNO,SNAME,STATUS,CITY) VALUES ('S1','精益',20,'天津');

INSERT INTO S (SNO,SNAME,STATUS,CITY) VALUES ('S2','盛锡',10,'北京');

INSERT INTO S (SNO,SNAME,STATUS,CITY) VALUES ('S3','东方红',30,'北京');

INSERT INTO S (SNO,SNAME,STATUS,CITY) VALUES ('S4','丰泰盛',20,'天津');

INSERT INTO S (SNO,SNAME,STATUS,CITY) VALUES ('S5','为民',30,'上海');

SELECT \* FROM S

INSERT INTO P (PNO,PNAME,COLOR,WEIGHT) VALUES ('P1','螺母','红',12);

INSERT INTO P (PNO,PNAME,COLOR,WEIGHT) VALUES ('P2','螺栓','绿',17);

INSERT INTO P (PNO,PNAME,COLOR,WEIGHT) VALUES ('P3','螺丝刀','蓝',14);

INSERT INTO P (PNO,PNAME,COLOR,WEIGHT) VALUES ('P4','螺丝刀','红',14);

INSERT INTO P (PNO,PNAME,COLOR,WEIGHT) VALUES ('P5','凸轮','蓝',40);

INSERT INTO P (PNO,PNAME,COLOR,WEIGHT) VALUES ('P6','齿轮','红',30);

SELECT \* FROM P

INSERT INTO J (JNO,JNAME,CITY) VALUES ('J1','三建','北京');

INSERT INTO J (JNO,JNAME,CITY) VALUES ('J2','一汽','长春');

INSERT INTO J (JNO,JNAME,CITY) VALUES ('J3','弹簧厂','天津');

INSERT INTO J (JNO,JNAME,CITY) VALUES ('J4','造船厂','天津');

INSERT INTO J (JNO,JNAME,CITY) VALUES ('J5','机车厂','唐山');

INSERT INTO J (JNO,JNAME,CITY) VALUES ('J6','无线电厂','常州');

INSERT INTO J (JNO,JNAME,CITY) VALUES ('J7','半导体厂','南京');

SELECT \* FROM J

INSERT INTO SPJ (SNO,PNO,JNO,QTY) VALUES ('S1','P1','J1',200);

INSERT INTO SPJ (SNO,PNO,JNO,QTY) VALUES ('S1','P1','J3',100);

INSERT INTO SPJ (SNO,PNO,JNO,QTY) VALUES ('S1','P1','J4',700);

INSERT INTO SPJ (SNO,PNO,JNO,QTY) VALUES ('S1','P2','J2',100);

INSERT INTO SPJ (SNO,PNO,JNO,QTY) VALUES ('S2','P3','J1',400);

INSERT INTO SPJ (SNO,PNO,JNO,QTY) VALUES ('S2','P3','J2',200);

INSERT INTO SPJ (SNO,PNO,JNO,QTY) VALUES ('S2','P3','J4',500);

INSERT INTO SPJ (SNO,PNO,JNO,QTY) VALUES ('S2','P3','J5',400);

INSERT INTO SPJ (SNO,PNO,JNO,QTY) VALUES ('S2','P5','J1',400);

INSERT INTO SPJ (SNO,PNO,JNO,QTY) VALUES ('S2','P5','J2',100);

INSERT INTO SPJ (SNO,PNO,JNO,QTY) VALUES ('S3','P1','J1',200);

INSERT INTO SPJ (SNO,PNO,JNO,QTY) VALUES ('S3','P3','J1',200);

INSERT INTO SPJ (SNO,PNO,JNO,QTY) VALUES ('S4','P5','J1',100);

INSERT INTO SPJ (SNO,PNO,JNO,QTY) VALUES ('S4','P6','J3',300);

INSERT INTO SPJ (SNO,PNO,JNO,QTY) VALUES ('S4','P6','J4',200);

INSERT INTO SPJ (SNO,PNO,JNO,QTY) VALUES ('S5','P2','J4',100);

INSERT INTO SPJ (SNO,PNO,JNO,QTY) VALUES ('S5','P3','J1',200);

INSERT INTO SPJ (SNO,PNO,JNO,QTY) VALUES ('S5','P6','J2',200);

INSERT INTO SPJ (SNO,PNO,JNO,QTY) VALUES ('S5','P6','J4',500);

SELECT \* FROM SPJ

（1）供应工程J1零件的供应商代码

USE SPJ SELECT SNO 供应工程J1零件的供应商代码

FROM SPJ

WHERE JNO = 'J1'

（2）供应工程J1零件P1的供应商代码

USE SPJ SELECT SNO 供应工程J1零件P1的供应商代码

FROM SPJ

WHERE JNO = 'J1' AND PNO = 'P1'

（3）供应工程J1零件为红的供应商号码

USE SPJ SELECT SNO 供应工程J1零件为红的供应商号码

FROM SPJ

WHERE SPJ.JNO = 'J1' AND SPJ.PNO IN ( SELECT PNO

FROM P

WHERE COLOR = '红')

USE SPJ SELECT SNO 供应工程J1零件为红的供应商号码

FROM ( SELECT PNO

FROM P

WHERE COLOR = '红') AS RED,

( SELECT SNO, PNO

FROM SPJ

WHERE JNO = 'J1') AS J1

WHERE J1.PNO = RED.PNO

USE SPJ SELECT SNO 供应工程J1零件为红的供应商号码

FROM (SELECT SNO, PNO

FROM SPJ

WHERE JNO = 'J1') AS J1

WHERE J1.PNO IN (SELECT PNO

FROM P

WHERE COLOR = '红')

（4）没有使用天津供应商生产的红零件的工程号

USE SPJ

SELECT DISTINCT JNO 没有使用天津供应商生产的红零件的工程号

FROM SPJ

WHERE JNO NOT IN (SELECT JNO

FROM SPJ,P,S

WHERE S.CITY='天津' AND COLOR='红' AND S.SNO=SPJ.SNO AND P.PNO=SPJ.PNO)

USE SPJ

SELECT JNO 没有使用天津供应商生产的红零件的工程号

FROM J

WHERE EXISTS

(SELECT \*

FROM SPJ

WHERE SPJ.JNO = J.JNO

)

AND NOT EXISTS

(SELECT \*

FROM (SELECT PNO,SNO

FROM (SELECT PNO

FROM P

WHERE COLOR = '红')AS RED, /\*先选择，后连接\*/

(SELECT SNO

FROM S

WHERE CITY = '天津')AS TIANJIN

) AS REDTJ

WHERE EXISTS

(SELECT \*

FROM SPJ

WHERE JNO = J.JNO

AND PNO = REDTJ.PNO

AND SNO = REDTJ.SNO

)

)

USE SPJ

SELECT DISTINCT JNO 没有使用天津供应商生产的红零件的工程号

FROM SPJ spj1

WHERE NOT EXISTS

(SELECT \*

FROM (SELECT PNO,SNO

FROM (SELECT PNO

FROM P

WHERE COLOR = '红')AS RED, /\*先选择，后连接\*/

(SELECT SNO

FROM S

WHERE CITY = '天津')AS TIANJIN

) AS REDTJ

WHERE EXISTS

(SELECT \*

FROM SPJ spj2

WHERE spj2.JNO = spj1.JNO

AND PNO = REDTJ.PNO

AND SNO = REDTJ.SNO

)

)

（5）用了S1供应的全部零件的项目号

use SPJ SELECT DISTINCT JNO 用了S1供应的全部零件的项目号

FROM SPJ spj1

WHERE NOT EXISTS(

SELECT \*

FROM (SELECT DISTINCT PNO

FROM SPJ

WHERE SNO = 'S1') AS S1

WHERE NOT EXISTS

(SELECT \*

FROM (SELECT PNO FROM SPJ

WHERE JNO = spj1.JNO) AS spj1\_pno

WHERE S1.PNO = spj1\_pno.PNO))

**相关文件：点击在SQL server中打开**

创建数据库及表：

（1）

（2）

（3）

（4）

（5）

## 第5题

（1）找出所有供应商的姓名和所在城市。

use SPJ

SELECT SNAME 供应商, CITY 所在城市

FROM S

（2）找出所有零件的名称颜色重量。

use SPJ

SELECT PNAME 零件, COLOR 颜色, WEIGHT 重量

FROM P

（3）使用S1所供应的零件的工程号。

use SPJ

SELECT DISTINCT JNO

FROM SPJ

WHERE SNO = 'S1'

（4）找出项目J2所使用的各种零件的名称及其数量。

/\*不考虑零件由不同供应商提供

J2是巧合\*/

use SPJ

SELECT PNAME, QTY

FROM (SELECT PNO, PNAME

FROM P) AS PNOPNAME,(SELECT PNO, QTY

FROM SPJ

WHERE JNO = 'J2') AS PNOQTY

WHERE PNOPNAME.PNO = PNOQTY.PNO

ORDER BY PNAME

/\*考虑零件是由不同供应商提供的\*/

SELECT PNAME 零件名称, SUM(QTY) 零件数量

FROM (SELECT TOP 100 PERCENT PNAME, QTY /\*必须加上 SELECT TOP 100 PERCENT, 不然order by 不能在子查询中\*/

FROM (SELECT PNO, PNAME /\*因为对于一个表的SELECT,返回的并不是一个表格，而是一个游标\*/

FROM P) AS PNOPNAME,(SELECT PNO, QTY

FROM SPJ

WHERE JNO = 'J2') AS PNOQTY

WHERE PNOPNAME.PNO = PNOQTY.PNO

ORDER BY PNAME) AS RESULT

GROUP BY RESULT.PNAME

（5）上海厂商提供的所有零件号码。

use SPJ

SELECT DISTINCT PNO

FROM SPJ

WHERE SNO IN (SELECT SNO

FROM S

WHERE CITY = '上海')

（6）使用上海产的零件的工程名称。

use SPJ

SELECT JNAME

FROM (SELECT JNO, JNAME

FROM J) AS JNOJNAME, (SELECT DISTINCT JNO

FROM SPJ

WHERE SNO IN (SELECT SNO

FROM S

WHERE CITY = '上海')) AS JNOTEMP

WHERE JNOJNAME.JNO = JNOTEMP.JNO

（7）没有使用天津产的零件的工程号。

use SPJ

SELECT DISTINCT JNO

FROM SPJ

WHERE SNO NOT IN (SELECT SNO

FROM S

WHERE CITY = '天津')

（8）把全部红色零件的颜色改成蓝色。

use SPJ

UPDATE P

SET COLOR = '蓝'

WHERE COLOR = '红'

（9）由S5供给J4的零件P6改为由S3供应。

use SPJ

UPDATE SPJ

SET SNO = 'S3'

WHERE SNO = 'S5' AND JNO = 'J4' AND PNO = 'P6'

（10）从供应商关系中删除供应商号是S2的记录，并从供应情况关系中删除相应的记录。

use SPJ

/\*1\*/

GO

BEGIN TRAN

SELECT \* FROM S /\*删除前的表格\*/

SELECT \* FROM SPJ

DELETE FROM SPJ WHERE SNO = 'S2'

DELETE FROM S WHERE SNO = 'S2'

SELECT \* FROM S /\*删除后的表格\*/

SELECT \* FROM SPJ

ROLLBACK

/\*2\*/

GO

BEGIN TRAN

SELECT \* FROM S /\*删除前的表格\*/

SELECT \* FROM SPJ

DECLARE @fun VARCHAR(200)

SET @fun = 'ALTER TABLE SPJ DROP CONSTRAINT FK\_SPJ\_S\_SNO' /\*删除外键约束\*/

EXEC(@fun)

DELETE FROM S WHERE SNO = 'S2'

DELETE FROM SPJ WHERE SNO = 'S2'

SELECT \* FROM S /\*删除后的表格\*/

SELECT \* FROM SPJ

SET @fun = 'ALTER TABLE SPJ ADD CONSTRAINT FK\_SPJ\_S\_SNO FOREIGN KEY (SNO) REFERENCES S(SNO)' /\*重新添加外键约束\*/

EXEC(@fun)

ROLLBACK

（11）请将(S2，J6，P4，200)插入供应情况关系。

INSERT INTO SPJ VALUES('S2', 'J6', 'P4', 200)

## 第9题

创建视图

use SPJ

GO

CREATE VIEW SANJIAN (SNO, PNO, QTY)

AS

SELECT SNO, PNO, QTY

FROM SPJ, (SELECT JNO

FROM J

WHERE JNAME = '三建') AS SANJIANJNO

WHERE SPJ.JNO = SANJIANJNO.JNO

（1）找出三建工程项目使用的各种零件代码及其数量。

USE SPJ

SELECT PNO 零件代码, SUM(QTY) 数量

FROM SANJIAN

GROUP BY PNO

（2）找出供应商S1的供应情况

use SPJ

SELECT PNO S1供应零件号, SUM(QTY) 零件数量

FROM SANJIAN

GROUP BY PNO

# 第四章

## 第6题

创建用户U1，U2

（1）授予用户U1对两个表的所有权限，并可给其他用户授权。

use STU\_CLASS

GRANT ALL

ON STUDENT

TO U1

WITH GRANT OPTION

GRANT ALL

ON CLASS

TO U1

WITH GRANT OPTION

（2）授予学生U2对学生表具有查看权限，对家庭住址具有更新权限。

use STU\_CLASS

GRANT SELECT

ON STUDENT

TO U2

GRANT UPDATE

ON STUDENT(SADDRESS)

TO U2

（3）将对班级表查看权限授予所有用户。

use STU\_CLASS

GRANT SELECT

ON CLASS

TO PUBLIC

（4）将对学生表的查询、更新权限授予角色R1。

use STU\_CLASS

GO

CREATE ROLE R1

GRANT SELECT,UPDATE

ON STUDENT

TO R1

（5）将角色R1授予用户U1，并且U1可继续授权给其他角色。

use STU\_CLASS

GRANT R1

TO U1

WITH ADMIN OPTION

## 第7题

（1）用户王明对两个表有SELECT权限。

GRANT SELECT

ON 职工,部门

TO 王明

（2）用户李勇对两个表有INSERT和DELETE权限。

GRANT INSERT,DELETE

ON 职工,部门

TO 李勇

（3）每个职工对自己的记录有SELECT权限。

GRANT SELECT

ON 职工

WHEN USER()=NAME

TO ALL

（4）用户刘星对职工表有SELECT 权力，对工资字段具有更新权力。

GRANT SELECT,UPDATE(工资)

ON 职工

TO 刘星

（5）用户张新具有修改这两个表结构的权限。

GRANT ALTER

ON 职工,部门

TO 张新

（6）用户周平具有对两个表所有权力（读，插，改，删数据），并具有给其他用户授权的权力。

GRANT ALL

ON 职工,部门

TO 周平

WITH GRANT OPTION

（7）用户杨兰具有从每个部门职工中 SELECT 最高工资、最低工资、平均工资的权力,他 不能查看每个人的工资。

CREATE VIEW 工资

AS

SELECT 部门号,名称,MAX(工资 ),MIN(工资 ),AVG(工资 )

FROM 职工,部门

WHERE 职工.部门号 =部门.部门号

GROUP BY 职工 . 部门号

GRANT SELECT ON 部门工资

TO 杨兰

## 第8题

（1）

REVOKE SELECT

ON 职工,部门

FROM 王明

（2）

REVOKE INSERT,DELETE

ON 职工,部门

FROM 李勇

（3）

REOVKE SELECT ON 职工

WHEN USER ( ) =NAME

FROM ALI

（4）

REVOKE SELECT,UPDATE

ON 职工

FROM 刘星

（5）

REVOKE ALTER

ON 职工,部门

FROM 张新

（6）

REVOKE ALL PRIVILIGES

ON 职工,部门

FROM 周平

（7）

REVOKE SELECT

ON 部门工资

FROM 杨兰;

DROP VIEW 部门工资

# 第五章

## 第6题

CREATE TABLE DEPT

(Deptno CHAR(2),

Deptname VARCHAR(10),

Manager VARCHAR(10),

PhoneNumber Char(12)

CONSTRAINT PK\_SC PRIMARY KEY(Deptno));

CREATE TABLE EMP

(Empno CHAR(4),

Ename VARCHAR(10),

Age SMALLINT(2),

CONSTRAINT C1 CHECK ( Aage<=60),

Job VARCHAR(9)

Sal INT(7,2),

Deptno CHAR(2),

CONSTRAINT FK\_DEPTNO

FOREIGN KEY(Deptno)

REFERENCES DEPT(Deptno));

## 第8题

CREATE TABLE FEMALE

(FID INT(5) PRIMARY KEY,

FNAME CHAR(10) NOT NULL,

FAGE NUMERIC(3) NOT NULL);

CREATE TABLE MALE

(MID INT(5) PRIMARY KEY,

MNAME CHAR(10) NOT NULL,

MAGE NUMERIC(3) NOT NULL);

CREATE ASSERTION COUNT\_PEOPLE

CHECK(50>=(SELECT COUNT(\*) FROM FEMALE) +

(SELECT COUNT(\*) FROM MALE))