

1.

```
INSERT
INTO books
VALUES(110, 'Fundamentals of Database Systems', 'Ramez
Elmasri', 2011, 'textbook');
```

	bookid	title	author	year	category
1	101	Introduction to Databases	David Maier	1981	textbook
2	102	Database System Concepts	Abraham Silberschatz	2004	textbook
3	103	Principles of Database and Knowledge-Base Systems	Jeffrey Ullman	1997	textbook
4	104	First Course in Database Systems	Jeffrey Ullman	2008	textbook
5	105	Promises to Keep: On Life and Politics	Joseph Biden	2008	politics
6	106	Change We Can Believe In: Barack Obama's Plan to Renew America's Promise	Barack Obama	2008	politics
7	107	Blindness	Jose Saramago	1999	romance
8	108	The Gospel According to Jesus Christ	Jose Saramago	1994	romance
9	109	Sauces: Classical and Contemporary Sauce Making	James Peterson	2008	cooking
10	110	Fundamentals of Database Systems	Ramez Elmasri	2011	textbook

```
UPDATE books
SET year=2012
WHERE bookid=110;
```

	bookid	title	author	year	category
1	101	Introduction to Databases	David Maier	1981	textbook
2	102	Database System Concepts	Abraham Silberschatz	2004	textbook
3	103	Principles of Database and Knowledge-Base Systems	Jeffrey Ullman	1997	textbook
4	104	First Course in Database Systems	Jeffrey Ullman	2008	textbook
5	105	Promises to Keep: On Life and Politics	Joseph Biden	2008	politics
6	106	Change We Can Believe In: Barack Obama's Plan to Renew America's Promise	Barack Obama	2008	politics
7	107	Blindness	Jose Saramago	1999	romance
8	108	The Gospel According to Jesus Christ	Jose Saramago	1994	romance
9	109	Sauces: Classical and Contemporary Sauce Making	James Peterson	2008	cooking
10	110	Fundamentals of Database Systems	Ramez Elmasri	2012	textbook

```
DELETE
FROM books
WHERE bookid=110;
```

	bookid	title	author	year	category
1	101	Introduction to Databases	David Maier	1981	textbook
2	102	Database System Concepts	Abraham Silberschatz	2004	textbook
3	103	Principles of Database and Knowledge-Base Systems	Jeffrey Ullman	1997	textbook
4	104	First Course in Database Systems	Jeffrey Ullman	2008	textbook
5	105	Promises to Keep: On Life and Politics	Joseph Biden	2008	politics
6	106	Change We Can Believe In: Barack Obama's Plan to Renew America's Promise	Barack Obama	2008	politics
7	107	Blindness	Jose Saramago	1999	romance
8	108	The Gospel According to Jesus Christ	Jose Saramago	1994	romance
9	109	Sauces: Classical and Contemporary Sauce Making	James Peterson	2008	cooking

2.

```
CREATE VIEW cus_book1(customer_id,book_name,cost)
AS
SELECT cid,bookid,pprice
FROM purchases;
```

入过滤条件, 按回车键开始

包含搜索内容 请输入搜索内容

Hzaaaa (59.77.6.4:26000)

数据库 (1)

postgres

系统模式 (483)

用户模式 (12)

public (0)

s4396 (12)

函数/过程 (0)

普通表 (7)

books

列

约束

索引

customers

dept

emp

pricing

purchases

列

约束

索引

salgrade

视图 (1)

cus_book1

列名

外表 (0)

序列 (0)

同义词 (0)

触发器 (4)

表空间 (0)

	customer_id	book_name	cost
1	1	101	20.00
2	1	102	60.00
3	1	103	40.00
4	1	104	120.00
5	1	106	10.00
6	1	107	7.49
7	1	109	32.97
8	2	101	20.00
9	2	102	60.00
10	3	103	40.00
11	3	104	120.00
12	3	109	32.97
13	4	105	20.00
14	5	106	60.00
15	6	107	60.00
16	7	109	30.45

3.

```
CREATE VIEW young_cus
AS
SELECT *
FROM customers
WHERE age BETWEEN 18 AND 35;
SELECT *
FROM young_cus;
```

	cid	cname	age
1	1	john	20
2	2	mary	18
3	3	jane	28
4	5	joyce	33
5	6	terry	25

4.

```
CREATE VIEW cus_book2(customer_id,customer_name,total_cost)
AS
SELECT customer_id,book_name,cost
FROM cus_book1;
```

条件, 按回车键开始

包含搜索内容 请输入搜索内容

	customer_id	customer_n...	total_cost
1	1	101	20.00
2	1	102	60.00
3	1	103	40.00
4	1	104	120.00
5	1	106	10.00
6	1	107	7.49
7	1	109	32.97
8	2	101	20.00
9	2	102	60.00
10	3	103	40.00
11	3	104	120.00
12	3	109	32.97
13	4	105	20.00
14	5	106	60.00
15	6	107	60.00
16	7	109	30.45

数据库 (1)
postgres
系统模式 (483)
用户模式 (22)
public (0)
s4396 (22)
函数/过程 (0)
普通表 (7)
books
customers
dept
emp
pricing
purchases
salgrade
视图 (3)
cus_book1
列名
cus_book2
young_cus
外表 (0)
序列 (0)
同义词 (0)
触发器 (12)

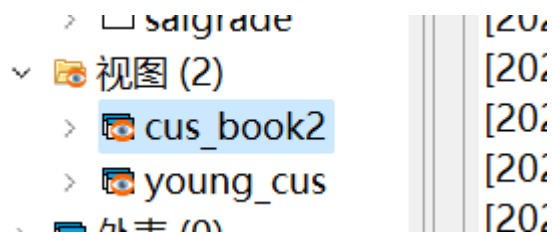
4.(1)

```
SELECT *
FROM cus_book2
ORDER BY total_cost DESC;
```

	customer_id	customer_n...	total_cost
1	1	104	120.00
2	3	104	120.00
3	6	107	60.00
4	5	106	60.00
5	1	102	60.00
6	2	102	60.00
7	3	103	40.00
8	1	103	40.00
9	1	109	32.97
10	3	109	32.97
11	7	109	30.45
12	2	101	20.00
13	4	105	20.00
14	1	101	20.00
15	1	106	10.00
16	1	107	7.49

4.(2)

```
DROP VIEW cus_book1 CASCADE;
```



4.(3)

```
SELECT *  
FROM cus_book2  
ORDER BY total_cost DESC;
```

```
[2023-04-09 15:38:34.248 CST]: [NOTICE] drop cascades to view cus_book2  
[2023-04-09 15:39:32.100 CST]: [ERROR] 执行失败  
错误代码: [0]SQL错误码: = 42P01  
[172.18.69.153:62357/59.77.6.4:26000] ERROR: relation "cus_book2" does not exist on dn_6001  
位置:63  
Line Number: 48
```

查询视图cus_book2时出现报错，使用cascade删除cus_book1时基于此视图的cus_book2视图失效，但是cus_book2的定义并没有被删除

5.

```
ALTER TABLE customers ADD YOB INTEGER;
```

	cid	cname	age	yob
1	1	john	20	[NULL]
2	2	mary	18	[NULL]
3	3	jane	28	[NULL]
4	4	ann	40	[NULL]
5	5	joyce	33	[NULL]
6	6	terry	25	[NULL]
7	7	claire	80	[NULL]
8	8	john	60	[NULL]

6.

```
UPDATE customers  
SET YOB=date_part('year',now())-age;
```

	cid	cname	age	yob
1	1	john	20	2003
2	2	mary	18	2005
3	3	jane	28	1995
4	4	ann	40	1983
5	5	joyce	33	1990
6	6	terry	25	1998
7	7	claire	80	1943
8	8	john	60	1963

7.

```
SELECT *
FROM young_cus;
```

	cid	cname	age
1	1	john	20
2	2	mary	18
3	3	jane	28
4	5	joyce	33
5	6	terry	25

查询结果和之前没有区别，因为根据young_cus的定义该视图不包括新增加的列

8.

4.

--创建表

```
CREATE TABLE S
(SNO CHAR(4) PRIMARY KEY,
SNAME VARCHAR(20),
STATUS INTEGER,
CITY VARCHAR(8)
);
INSERT INTO S VALUES('S1', '精益', 20, '天津');
INSERT INTO S VALUES('S2', '盛锡', 10, '北京');
INSERT INTO S VALUES('S3', '东方红', 30, '北京');
INSERT INTO S VALUES('S4', '丰泰盛', 20, '天津');
INSERT INTO S VALUES('S5', '为民', 30, '上海');
```

```
CREATE TABLE P
(PNO CHAR(4) PRIMARY KEY,
PNAME VARCHAR(20),
COLOR CHAR(4),
```

```

WEIGHT INTEGER
);
INSERT INTO P VALUES('P1','红',12);
INSERT INTO P VALUES('P2','绿',17);
INSERT INTO P VALUES('P3','蓝',14);
INSERT INTO P VALUES('P4','红',14);
INSERT INTO P VALUES('P5','蓝',40);
INSERT INTO P VALUES('P6','红',30);

CREATE TABLE J
(JNO CHAR(4) PRIMARY KEY,
JNAME VARCHAR(20),
CITY VARCHAR(8)
);
INSERT INTO J VALUES('J1','三建','北京');
INSERT INTO J VALUES('J2','一汽','长春');
INSERT INTO J VALUES('J3','弹簧厂','天津');
INSERT INTO J VALUES('J4','造船厂','天津');
INSERT INTO J VALUES('J5','机车厂','唐山');
INSERT INTO J VALUES('J6','无线电厂','常州');
INSERT INTO J VALUES('J7','半导体厂','南京');

CREATE TABLE SPJ
(SNO CHAR(4),
PNO CHAR(4),
JNO CHAR(4),
QTY INTEGER,
PRIMARY KEY (SNO,PNO,JNO),
FOREIGN KEY (SNO) REFERENCES S(SNO),
FOREIGN KEY (PNO) REFERENCES P(PNO),
FOREIGN KEY (JNO) REFERENCES J(JNO)
);
INSERT INTO SPJ VALUES('S1','P1','J1',200);
INSERT INTO SPJ VALUES('S1','P1','J3',100);
INSERT INTO SPJ VALUES('S1','P1','J4',700);
INSERT INTO SPJ VALUES('S1','P2','J2',100);
INSERT INTO SPJ VALUES('S2','P3','J1',400);
INSERT INTO SPJ VALUES('S2','P3','J2',200);
INSERT INTO SPJ VALUES('S2','P3','J4',500);
INSERT INTO SPJ VALUES('S2','P3','J5',400);
INSERT INTO SPJ VALUES('S2','P5','J1',400);
INSERT INTO SPJ VALUES('S2','P5','J2',100);

```

```

INSERT INTO SPJ VALUES('S3','P1','J1',200);
INSERT INTO SPJ VALUES('S3','P3','J1',200);
INSERT INTO SPJ VALUES('S4','P5','J1',100);
INSERT INTO SPJ VALUES('S4','P6','J3',300);
INSERT INTO SPJ VALUES('S4','P6','J4',200);
INSERT INTO SPJ VALUES('S5','P2','J4',100);
INSERT INTO SPJ VALUES('S5','P3','J1',200);
INSERT INTO SPJ VALUES('S5','P6','J2',200);
INSERT INTO SPJ VALUES('S5','P6','J4',500);

```

	sno	sname	status	city
1	S1	精益	20	天津
2	S2	盛锡	10	北京
3	S3	东方红	30	北京
4	S4	丰泰盛	20	天津
5	S5	为民	30	上海

	pno	pname	color	weight
1	P1	螺母	红	12
2	P2	螺栓	绿	17
3	P3	螺丝刀	蓝	14
4	P4	螺丝刀	红	14
5	P5	凸轮	蓝	40
6	P6	齿轮	红	30

	jno	jname	city
1	J1	三建	北京
2	J2	一汽	长春
3	J3	弹簧厂	天津
4	J4	造船厂	天津
5	J5	机车厂	唐山
6	J6	无线电厂	常州
7	J7	半导体厂	南京

	sno	pno	jno	qty
1	S1	P1	J1	200
2	S1	P1	J3	100
3	S1	P1	J4	700
4	S1	P2	J2	100
5	S2	P3	J1	400
6	S2	P3	J2	200
7	S2	P3	J4	500
8	S2	P3	J5	400
9	S2	P5	J1	400
10	S2	P5	J2	100
11	S3	P1	J1	200
12	S3	P3	J1	200
13	S4	P5	J1	100
14	S4	P6	J3	300
15	S4	P6	J4	200
16	S5	P2	J4	100
17	S5	P3	J1	200
18	S5	P6	J2	200
19	S5	P6	J4	500

(1)

```
SELECT SNO
FROM spj
WHERE JNO='J1';
```

	sno
1	S1
2	S2
3	S2
4	S3
5	S3
6	S4
7	S5

(2)

```
SELECT SNO
FROM spj
WHERE JNO='J1' AND PNO='P1';
```


	sno
1	S1
2	S3

(3)

```
SELECT SNO
FROM spj,p
WHERE spj.JNO='J1' AND spj.PNO=p.PNO AND p.COLOR='红';
```

	sno
1	S1
2	S3

(4)

```
SELECT jno
FROM j
WHERE NOT EXISTS
(
    SELECT *
    FROM spj,s,p
    WHERE spj.sno=s.sno AND j.jno=spj.jno AND
    spj.pno=p.pno AND s.city='天津' AND p.color='红'
);
```

	jno
1	J7
2	J5
3	J6
4	J2

(5)

```

SELECT jno
FROM spj spj1
WHERE NOT EXISTS
(
  SELECT *
  FROM spj spj2
  WHERE spj2.sno='s1' AND
  NOT EXISTS
  (
    SELECT *
    FROM spj spj3
    WHERE spj3.sno=spj1.sno AND spj3.pno=spj2.pno
  )
);

```

	jno
1	J1
2	J3
3	J4
4	J2

5.

(1)

```

SELECT sname,city
FROM s;

```

	sname	city
1	精益	天津
2	盛锡	北京
3	东方红	北京
4	丰泰盛	天津
5	为民	上海

(2)

```
SELECT pname,color,weight
FROM p;
```

	pname	color	weight
1	螺母	红	12
2	螺栓	绿	17
3	螺丝刀	蓝	14
4	螺丝刀	红	14
5	凸轮	蓝	40
6	齿轮	红	30

(3)

```
SELECT jno
FROM spj
WHERE sno='S1';
```

	jno
1	J1
2	J3
3	J4
4	J2

(4)

```
SELECT pname,qty
FROM spj,p
WHERE spj.pno=p.pno AND jno='J2';
```

	pname	qty
1	螺栓	100
2	螺丝刀	200
3	凸轮	100
4	齿轮	200

(5)

```
SELECT DISTINCT p.pno
FROM p,spj,s
WHERE spj.sno=s.sno AND spj.pno=p.pno AND s.city='上海';
```

	pno
1	P3
2	P6
3	P2

(6)

```
SELECT jname
FROM j,spj,s
WHERE spj.sno=s.sno AND spj.jno=j.jno AND s.city='上海';
```

	jname
1	造船厂
2	三建
3	一汽
4	造船厂

(7)

```
SELECT jno
FROM j
WHERE NOT EXISTS
(
    SELECT *
    FROM spj
    WHERE spj.jno=j.jno
    AND sno IN
    (
        SELECT sno
        FROM s
    )
)
```

```
WHERE city='天津'
)
);
```

	jno
1	J7
2	J5
3	J6

(8)

```
UPDATE p
SET color='蓝'
WHERE color='红';
```

	pno	pname	color	weight
1	P2	螺栓	绿	17
2	P3	螺丝刀	蓝	14
3	P5	凸轮	蓝	40
4	P1	螺母	蓝	12
5	P4	螺丝刀	蓝	14
6	P6	齿轮	蓝	30

(9)

```
UPDATE spj
SET sno='S3'
WHERE sno='S5' AND jno='J4' AND pno='P6';
```

	sno	pno	jno	qty
1	S1	P1	J1	200
2	S1	P1	J3	100
3	S1	P1	J4	700
4	S1	P2	J2	100
5	S2	P3	J1	400
6	S2	P3	J2	200
7	S2	P3	J4	500
8	S2	P3	J5	400
9	S2	P5	J1	400
10	S2	P5	J2	100
11	S3	P1	J1	200
12	S3	P3	J1	200
13	S4	P5	J1	100
14	S4	P6	J3	300
15	S4	P6	J4	200
16	S5	P2	J4	100
17	S5	P3	J1	200
18	S5	P6	J2	200
19	S3	P6	J4	500

(10)

```
DELETE
FROM spj
WHERE sno='S2';
```

```
DELETE
FROM s
WHERE sno='S2';
```

	sno	pno	jno	qty
1	S1	P1	J1	200
2	S1	P1	J3	100
3	S1	P1	J4	700
4	S1	P2	J2	100
5	S3	P1	J1	200
6	S3	P3	J1	200
7	S4	P5	J1	100
8	S4	P6	J3	300
9	S4	P6	J4	200
10	S5	P2	J4	100
11	S5	P3	J1	200
12	S5	P6	J2	200
13	S3	P6	J4	500

	sno	sname	status	city
1	S1	精益	20	天津
2	S3	东方红	30	北京
3	S4	丰泰盛	20	天津
4	S5	为民	30	上海

(11)

```
INSERT INTO s(sno) VALUES('S2');  
INSERT INTO spj(sno,jno,pno,qty) VALUES('S2','J6','P4',200);
```

	sno	pno	jno	qty
1	S1	P1	J1	200
2	S1	P1	J3	100
3	S1	P1	J4	700
4	S1	P2	J2	100
5	S3	P1	J1	200
6	S3	P3	J1	200
7	S4	P5	J1	100
8	S4	P6	J3	300
9	S4	P6	J4	200
10	S5	P2	J4	100
11	S5	P3	J1	200
12	S5	P6	J2	200
13	S3	P6	J4	500
14	S2	P4	J6	200

9.

```
--创建  
CREATE VIEW spj_view(sno,pno,qty)  
AS  
SELECT sno,pno,qty  
FROM spj  
WHERE jno=  
(  
    SELECT jno  
    FROM J  
    WHERE JNAME='三建'  
);
```

(1)

```
SELECT pno,qty  
FROM spj_view;
```

	pno	qty
1	P1	200
2	P1	200
3	P3	200
4	P5	100
5	P3	200

(2)

```
SELECT *  
FROM spj_view  
WHERE sno='S1';
```

	sno	pno	qty
1	S1	P1	200