SUPPLIER DATABASE.

```
create table suppliers (sid int primary key,
sname varchar(20), address varchar(20));

create table parts (pid int primary key, pnam
varchar(20), color varchar(40));

create table catalog (sid int , pid int ,
foreign Key (sid) references SUPPLIERS (sid),
foreign Key (pid) references PARTS (pid), cos
float(6), primary key (sid, pid));

insert in supplier values
(10001, 'Acme widget', 'Bangalore');
(10002, 'Johns', 'Kolkato');
(10003, 'Vimal', 'Mumbai');
(10004, 'Reliance', 'Delhi');
(10005, 'Mahindra', 'Mumbai');
select * from SUPPLIERS;
commit;
insert into parts values
(20001, 'Book', 'Red');
(20002, 'Pen', 'Red'),
(2000 3, 'Pencil', 'Green'),
(2000 4, 'Mobile', 'Green'))
(20005, 'Charger', 'Black'),
select from parts;
Commit;
```

insert into catalog values (10001, 20001, 10);
(10001, 20002, 10);
(10001, 20003, 30),
(10001, 20004, 10),
(10001, 20005, 10),
(10002, 20001, 10);
(10002, 20002, 20),
(10003, 20003, 30),
(10004, 20003, 40);

```
select * from catalog;

Select DISTINCT p.pname
From Parts P, catalog C
where p.pid = c.pid;
```

Output:

Pname
Book
Pen
Pencil
Mobile
charger

```
Select S.sname
from suppliers
where ((Select count (P.Pid)
        from Part P) = (select count (C.pid)
                        from catalog C
                        where C.sid = S.sid));
```

Output: Sname
Acme Widge

```
Select s.sname
from Suppliers
where ((select count(p.pid)
        from Parts P where color='Red')=
                (select count(c.pid)
                from catalog C, Parts P
                where C.sid=S.sid
                                AND
                C.pid=p.pid AND
                p.color='red'));
```

Output :

Sname
Acme Widget
Johns

```
Select P.Pname
from parts P. Catalog C, Suppliers S
where p.pid = C.pid AND C.sid = S.Sid
AND S.sname = 'Acme Widget'
AND NOT EXISTS (Select *
        From Catalog C1, Suppliers S1
        where P.Pid = C1.pid AND C1.Sid=
                                S1.Sid AND
        S1.Sname <> 'Acme Widget');
```

Pname
Mobile
Charger

```
Select DISTINCT C.sid From Catalog C
where C.Colt 0.7( Select AVG(C1.Colt)
from catalog C1
where C1.pid = C.pid);
```

Output:

Sid

10001

10002

```
Select  p.pid , s.sname
from    Parts P, Supplier S, catalog C
where   C.pid = P.pid
AND     C.Sid = S.Sid
AND     C.Cost = (Select Max (C1.cost)
 From   Catalog C1
 where  C1.pid = P.Pid);
```

Output

Pid	Sname
20001	Acme widget.
20002	Johns.
20002	Johns .
20003	Reliance .
20004	Acme Widget.
20005	Acme Widget.

```
Select  s.sid , s.sname , p.pid , p.pname , c.cost
From    catalog_new C
Join    suppliers_new  ON c.sid = s.sid
Join    parts_new p  ON  c.pid = p.pid
where   c.cost = (Select MAX(cost) from catalog_new);
```

Output:

Sid	sname	pid	pname	cost
10004	Reliance	20003	pencil	40

```
Select sid , s.sname
from suppliers_new s
where s.sid NOT IN(
      Select c.sid
      From catalog_new c
      Join parts_new p ON c.pid = p.pid
      where p.colour = 'Red');
```

Output:

Sid	sname
10003	Vimal
10004	Reliance
10005	Mahindra
-	-

```
Select s.sid , s.sname , SUM(c.cost) AS total_value
From suppliers_new s
Left Join catalog_new c ON s.sid = c.cid
Group BY s.sid, s.sname;
```

Output:

Sid	sname	total_value
10001	Acme widget	70
10002	Johns	30
10003	Vimal	30
10004	Reliance	40
10005	Mahindra	-

```
Select c.sid
from catalog_newC
where c.cost < 20
Group By c.sid
Having count(*) >= 2;
```

Output:

Sid
10001

```
Select c.sid, c.pid, c.cost
From catalog_new C
Where c.cost = (
        Select MIN(c2.cost)
        From catalog_new c2
        Where c2.pid = c.pid
    );
```

Output:

sid	pid	cost
10001	20001	10
10001	20002	10
10001	20003	30
10001	20004	10
10001	20005	10
10002	20001	10
10003	20003	30
-	-	-

```
Create view supplier_part_count AS
Select s.sid, s.sname, count(c.pid) AS To
From suppliers_new s
Left Join catalog_new c ON s.sid = c.sid
Group BY s.sid, s.sname;

create view .most_expensive_supplier_per_part AS
Select c.sid, c.pid, c.cost
From catalog_new C
where c.cost = (
        Select MAX (c2.cost)
        From catalog_new c2
        where c2.pid = c.pid

CREATE TRIGGER prevent_low_cost
Before Insert ON catalog_new
FOR EACH ROW
BEGIN
    IF NEW.cost < 1 THEN
        SIGNAL SQLSTATE '45000'
        SET MESSAGE_TEXT = 'Cost cann
                           ot be less than 1';
    END IF;
END $$
```

```
create trigger set_default_cost_new
Before Insert ON Catalg_new
FOR EACH ROW
  Begin
     IF New.cost IS NULL THEN
          SET NEW.cost = 10;
     END IF;
     END $$
```