Topgyal Gurung

[Topgyal.gurung71@myhunter.cuny.edu](mailto:Topgyal.gurung71@myhunter.cuny.edu)

CS 435-02

Students discussed with: Riyadh and Wei

**Consider a schema as follows:**  
Supplier (SID, NAME, SUPPLIER\_ZIP)  
Parts (PID, PNAME, PTYPE,SID)   
Sale (SHOPID, PID, SALES\_DATE, QTY)   
Shop (SHOPID, SHOP\_ZIP)

Use VOLUME = SUM(QTY)

SUPPLIER with SID supplies this part. PTYPE can be Perishable or Imperishable.There can be more than one SHOPID per SHOP\_ZIP and more than one Supplier per Supplier\_ZIP. More than one supplier can supply a PARTS.

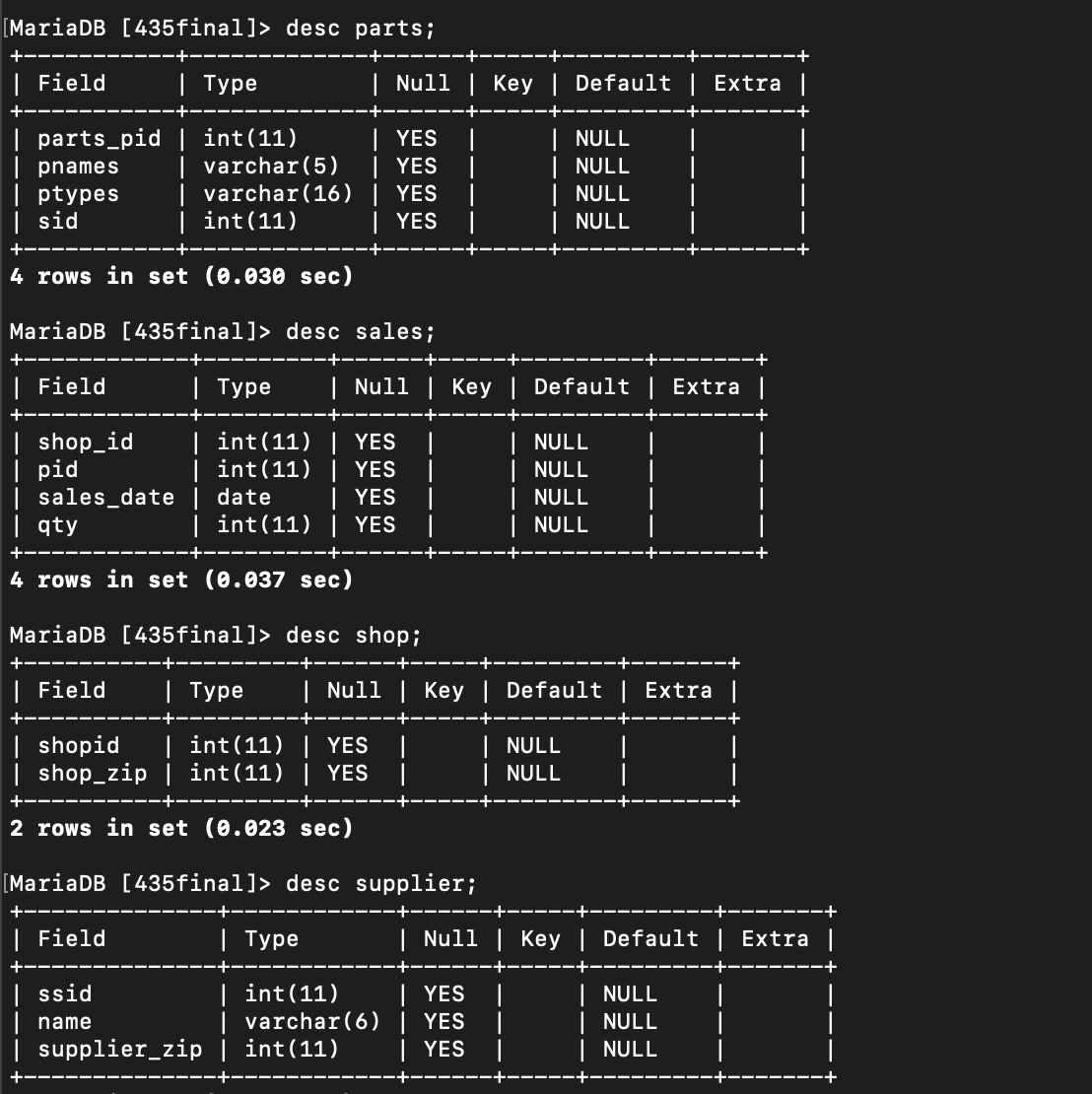
First I used the R console and imported the data and then I export it to my desktop with the folder name finalData in order to do the solve in MySQL MariaDB:

* *write.csv(sales,"/Users/topgyaltsering/Desktop/finalData/sales.csv",row.names=FALSE)*

Then to use MySQL in MariaDB, I created database: *435final* and imported data from the folder:

* *load data local infile '/Users/topgyaltsering/Desktop/435final /finalData/supplier.csv' into table supplier fields terminated by ',' enclosed by '"' lines terminated by '\n' ignore 1 lines;*

*Description of tables:*

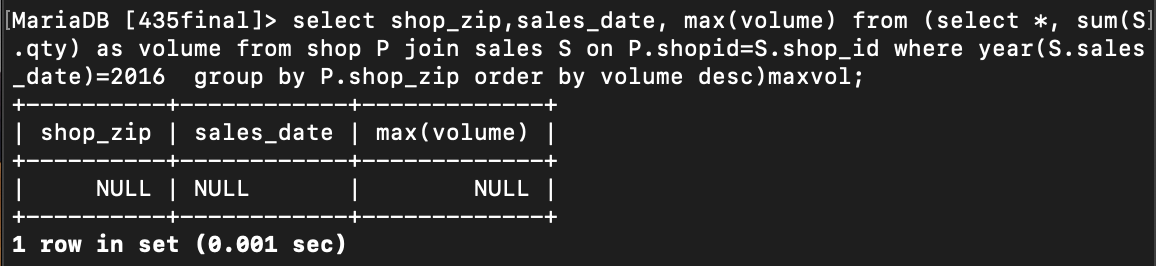
**

1. **Which SHOP\_ZIP sold the most volume in the year 2016?**

Time spent: 10 mins

NOTE: dublicate column attribute (shopid) for shop and sales) so I name shop\_id for sales S.

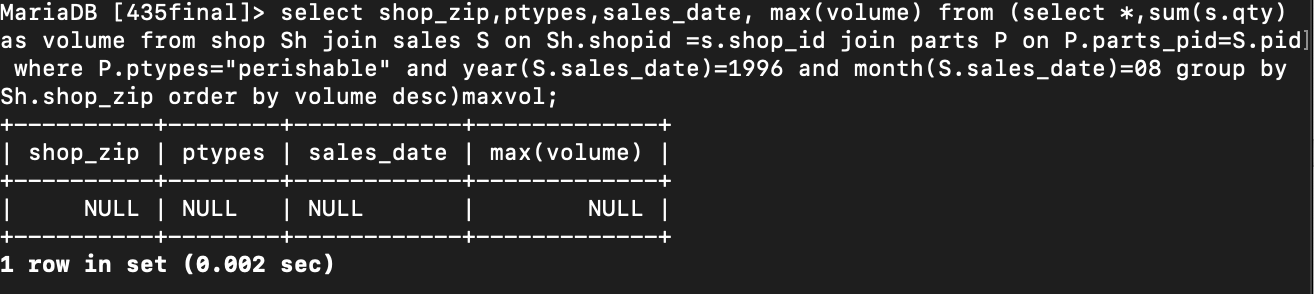
* select shop\_zip,sales\_date, max(volume) from (select \*, sum(S.qty) as volume from shop P join sales S on P.shopid=S.shop\_id where year(S.sales\_date)=2016 group by P.shop\_zip order by volume desc)maxvol;



2. Which SHOP\_ZIP sold the most volume in Perishable in August 2018?

Time spent: 10 min

* select shop\_zip,ptypes,sales\_date, max(volume) from (select \*,sum(s.qty) as volume from shop Sh join sales S on Sh.shopid =s.shop\_id join parts P on P.parts\_pid=S.pid where P.ptypes="perishable" and year(S.sales\_date)=1996 and month(S.sales\_date)=08 group by Sh.shop\_zip order by volume desc)maxvol;



1. List all the SHOP\_ZIPs that sell more Perishable than Imperishable on the first day of the month?

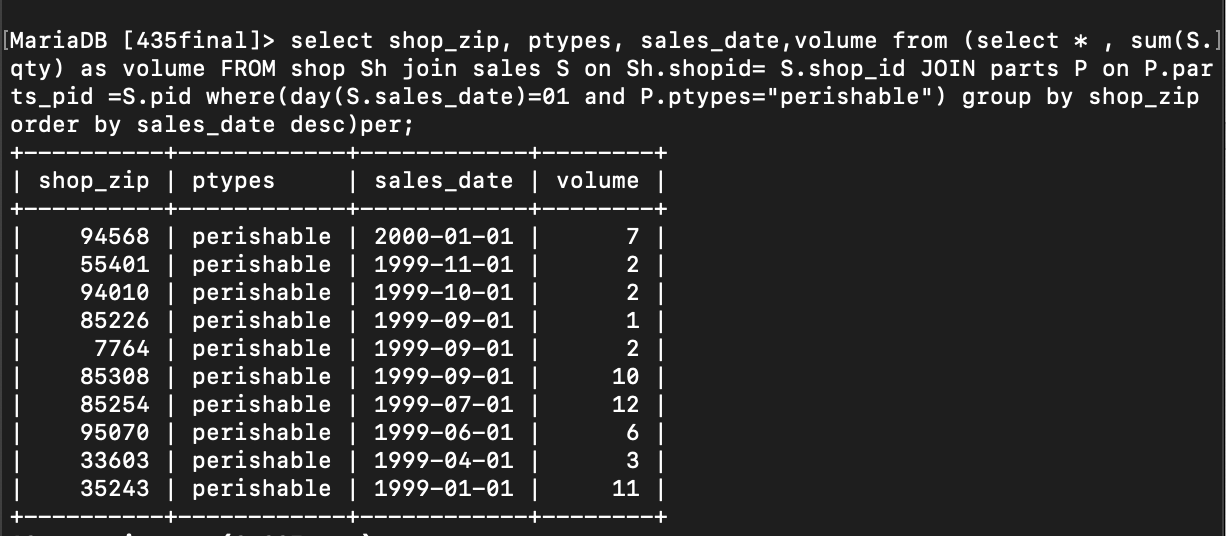
Time spent: 30 mins

NOTE:

* select perishable.\* from (select shop\_id,sales\_date, sum(qty) pqty from sales as S join parts as P on S.pid =P.parts\_pid and P.ptypes="perishable" where day(sales\_date)=1 group by sales\_date) as Perishable join (select shop\_id,sales\_date, sum(qty) pqty from sales as S join parts P on S.pid=P.parts\_pid and ptypes="imperishable" where day(sales\_date)=1 group by sales\_date) as imperishable on Perishable.pqty>imperishable.pqty and Perishable.shop\_id=imperishable.shop\_id;

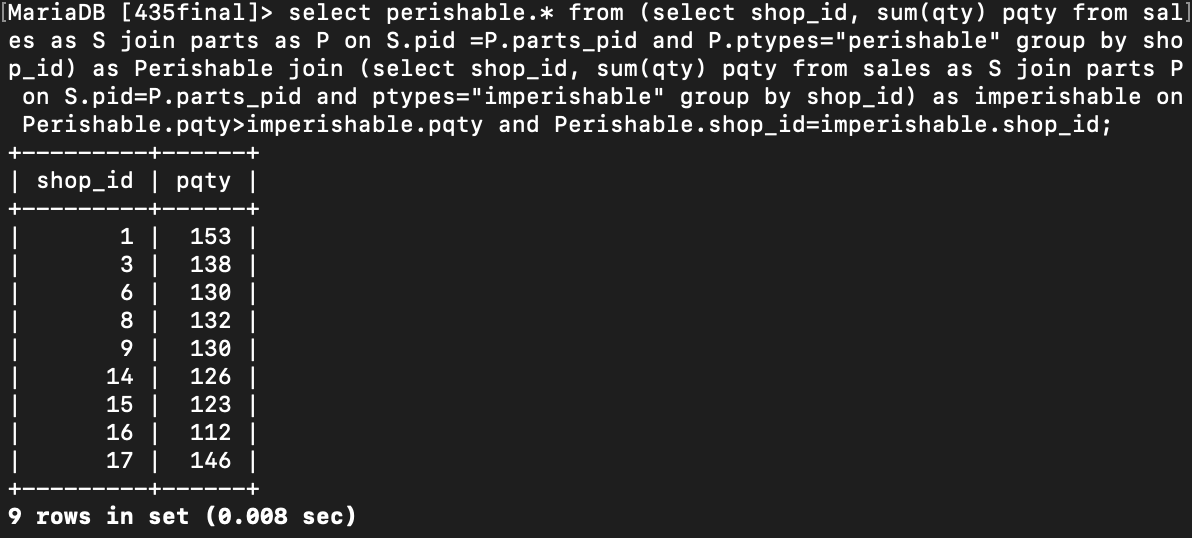
SOLUTION:

* select shop\_zip, ptypes, sales\_date,volume from (select \* , sum(S.qty) as volume FROM shop Sh join sales S on Sh.shopid= S.shop\_id JOIN parts P on P.parts\_pid =S.pid where(day(S.sales\_date)=01 and P.ptypes="perishable") group by shop\_zip order by sales\_date desc)per;

  
  
4. List all Suppliers that supplied more Perishable than Imperishable.

Time: 15 mins

* select perishable.\* from (select shop\_id, sum(qty) pqty from sales as S join parts as P on S.pid =P.parts\_pid and P.ptypes="perishable" group by shop\_id) as Perishable join (select shop\_id, sum(qty) pqty from sales as S join parts P on S.pid=P.parts\_pid and ptypes="imperishable" group by shop\_id) as imperishable on Perishable.pqty>imperishable.pqty and Perishable.shop\_id=imperishable.shop\_id;



NOTES :

From table parts, Total sales by each: select sid,count(\*) as total from parts group by sid;

Based on ptypes: select sid,ptypes,count(\*) as total from parts group by sid,ptypes;

*//only printing all perishable*

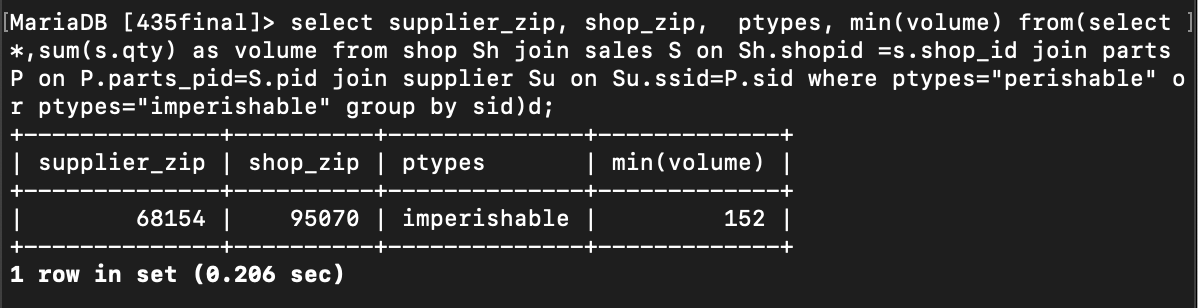
* select name,supplier\_zip,total,ptypes from (select \*,count(\*) as total from parts P join supplier S on P.sid=S.ssid where P.ptypes="perishable" group by sid,ptypes order by name)sup;

5. For each supplier\_zip list the SHOP\_ZIP that sold the minimum volume

perishable/imperishable combined

Time:15 mins

* select supplier\_zip, shop\_zip, ptypes, min(volume) from(select \*,sum(s.qty) as volume from shop Sh join sales S on Sh.shopid =s.shop\_id join parts P on P.parts\_pid=S.pid join supplier Su on Su.ssid=P.sid where ptypes="perishable" or ptypes="imperishable" group by sid)d;



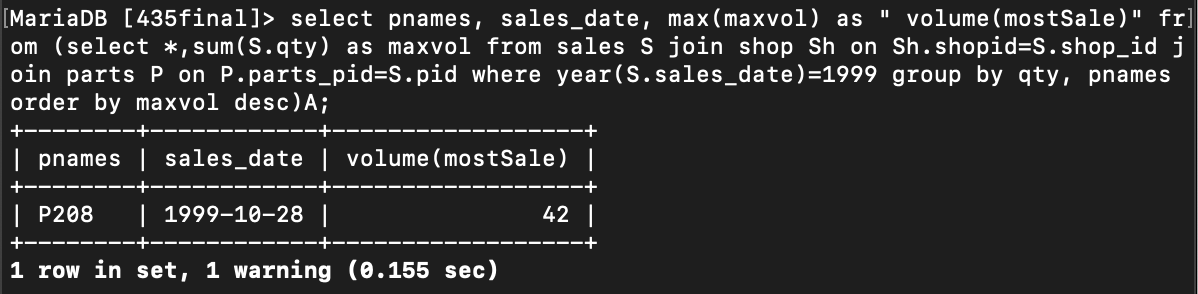
NOTE: count(\*) as total from parts group by sid

6. Find the Part(NAME) that sold the most in 2017

Time: 10mins

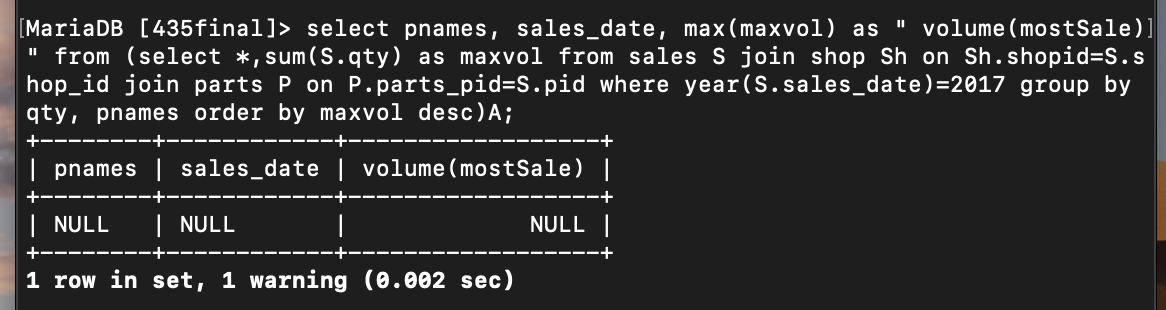
Examined with 1999

* select pnames, sales\_date, max(maxvol) as " volume(mostSale)" from (select \*,sum(S.qty) as maxvol from sales S join shop Sh on Sh.shopid=S.shop\_id join parts P on P.parts\_pid=S.pid where year(S.sales\_date)=1999 group by qty, pnames order by maxvol desc)A;



SOLUTION: For **year 2017**,

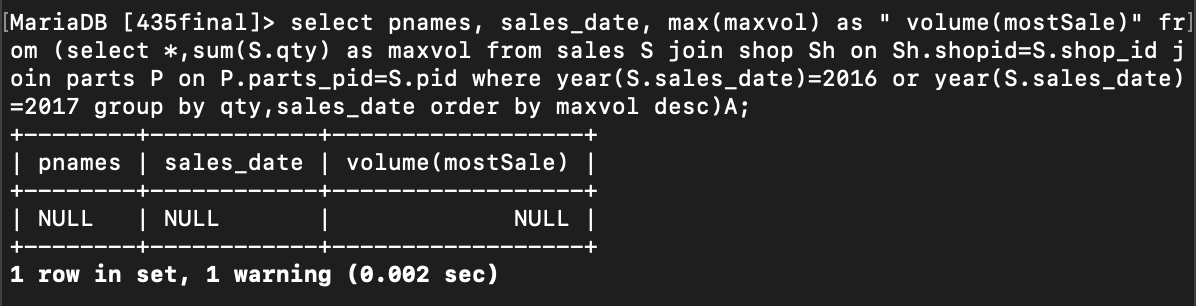
* select pnames, sales\_date, max(maxvol) as " volume(mostSale)" from (select \*,sum(S.qty) as maxvol from sales S join shop Sh on Sh.shopid=S.shop\_id join parts P on P.parts\_pid=S.pid where year(S.sales\_date)=2017 group by qty, pnames order by maxvol desc)A;

  
  
7. Find the part(NAME) whose sales volume in 2016 was higher than the sales volume in 2017?

Time spent: 10mins

This show the max from the two years

* select pnames, sales\_date, max(maxvol) as " volume(mostSale)" from (select \*,sum(S.qty) as maxvol from sales S join shop Sh on Sh.shopid=S.shop\_id join parts P on P.parts\_pid=S.pid where year(S.sales\_date)=2016 or year(S.sales\_date)=2017 group by qty,sales\_date order by maxvol desc)A;



8.1) Are these tables efficiently designed?

Time spent: 5 mins

No, the tables are not efficiently designed as

For supplier, supplier name use the same sid number except for S word in front of sid.

Likewise for parts, parts\_pid and pnames are same except pnames has P before the parts\_pid number. It takes space and waste of memory. For ptypes: we just have perishable and imperishable but there is repetition in the table and taking lots of space.

Sales table has listed all day, month and year but all data are for year 1999. And there is no primary key.

8.2) Is there anything you can do to make it more efficient?

Time: 5 mins

We can remove name for supplier table and use sid as identifying the suppliers.

We can have parts\_pid to identify parts for parts table. Pnames is not needed.

We can split parts into one table for perishable and another for imperishable

We can just have day and month for table sales. Year 1999 is not needed. we can name year with the sales\_date. It can be something like sales\_date1999.

8.3) what level of Normalization is each table in?

Time : 5 mins

NORMALIZATION: purpose to eliminate redundant data and ensure data dependence

1NF: all attributes are single valued.

2NF: has primary key (direct dependencies)

3NF: no transitive dependence

Table:

Shop is in 1NF, 2NF and 3NF as well or simply we can say in 3NF. Primary key: shopid

Parts is in 2NF not in 3NF because ptype and pnames has transitive dependence

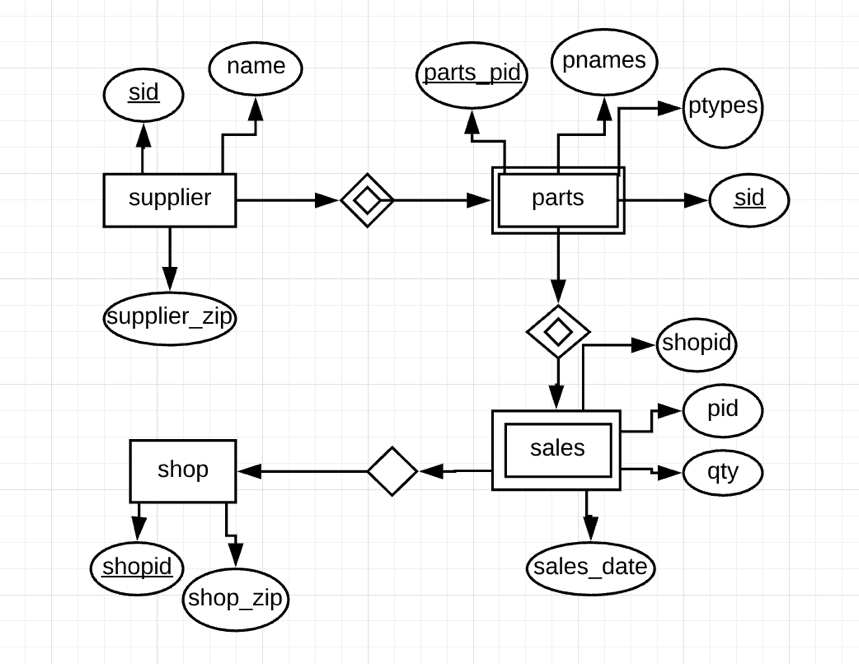
Table sales is in 3NF . primary key can be combination of shopid and pid

Table supplier is in 3NF. Primary key:SID

8.4) Draw an E-R diagram using your favorite notation, identifying Weak entities,

relationships, cardinality

time: 5 mins



NOTE: Sales table is weak entity as it can not exist on its own. For it to exist there needs to be parts to be sold by supplier.

9.1) Write a query to generate a report of the following format:  
 NAME,PNAME,QTY

Time: 5 mins

select name,pnames,qty from (select \* from supplier S join parts P on S.ssid=P.sid join sales Sa on Sa.pid=P.parts\_pid)query;

Detail: select Su.name,P.pnames, S.qty from supplier Su, Parts P ,Sales S;

9.2) Write an equivalent Relational Algebra expression.

Relational Algebra:

σ =select

∏ =project

⊗ =join

ρ=rename

U = union

Time: 5 mins

∏supplier.name.parts.pname,sale,qty Supplier ⊗ supplier.sid=parts.sid Parts THETA JOIN parts.pid=sales.pid Sales

10.1) Given A SQL Statement of the form "select a, b, c from t" and its output, can you

Determine the number of columns in table t?

Time:5 mins

I can use count(\*) to count no of columns

* select count(\*) from information\_schema.columns where table\_schema='databaase\_name' and table\_name=t;

10.2) Write the above select statement in Relational Algebra

Time: 2 mins

= ∏a,b,c (t)

10.3) write select \* from t in Relational Algebra

Time: 2 mins

= t

Or σ (t)

11) consider a table as shown persons (pid, name, gender, relation1, rid)  relation can be son-of,

daughter-of and rid   
  
**some example tuples:**  
(1,'vr','M',null,null)  
(2,'ra','F',null,null)  
(19,'rk','M','son-of',1)  
(19,'rk','M','son-of',2)

11.1) Please determine a primary key for this relation.

Time: 3 mins

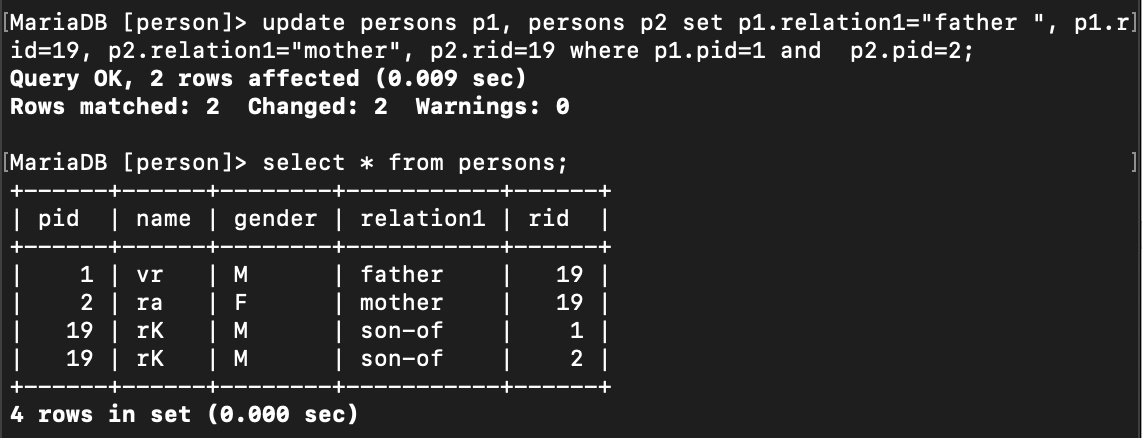
Primary key: (name, rid)

11.2) write a query to generate a report as follows:

('rk','father','vr','mother','ra')

Time: 15mins

* update persons p1, persons p2 set p1.relation1="father ", p1.rid=19, p2.relation1="mother", p2.rid=19 where p1.pid=1 and p2.pid=2;
* select \* from persons;



11.3) Write the above query in relational algebra

Time: 10 mins:

Update:

ρ p1.relation1= “father” p1.pid=19 p2.relation1= “mother” p2.pid=19 σ p1.pid=1 and p2.pid=2

Select \* from persons:

* persons