create database as\_zomato;

use as\_zomato;

CREATE TABLE goldusers\_signup

(userid int,gold\_signup\_date date);

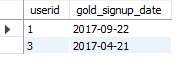
INSERT INTO goldusers\_signup

(userid,gold\_signup\_date)

VALUES (1,'2017-09-22'),

(3,'2017-04-21');

select \* from goldusers\_signup;



CREATE TABLE users

(userid int,signup\_date date);

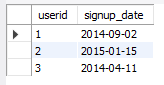
INSERT INTO users(userid,signup\_date)

VALUES (1,'2014-09-02'),

(2,'2015-01-15'),

(3,'2014-04-11');

select \* from users;



CREATE TABLE sales

(userid int,

created\_date date,

product\_id int);

INSERT INTO sales(userid,created\_date,product\_id)

VALUES (1,'2017-04-19',2),

(3,'2019-12-18',1),

(2,'2020-07-20',3),

(1,'2019-10-23',2),

(1,'2018-03-19',3),

(3,'2016-12-20',2),

(1,'2016-11-09',1),

(1,'2016-05-20',3),

(2,'2017-09-24',1),

(1,'2017-03-11',2),

(1,'2016-03-11',1),

(3,'2016-11-10',1),

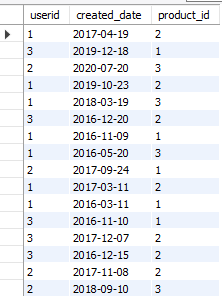
(3,'2017-12-07',2),

(3,'2016-12-15',2),

(2,'2017-11-08',2),

(2,'2018-09-10',3);

select \* from sales;



CREATE TABLE product

(product\_id int,

product\_name text,

price int);

INSERT INTO product

(product\_id,product\_name,price)

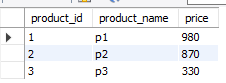
VALUES

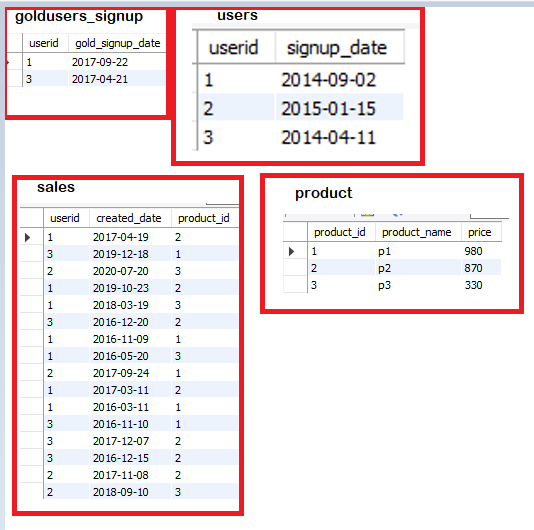
(1,'p1',980),

(2,'p2',870),

(3,'p3',330);

select \* from product;





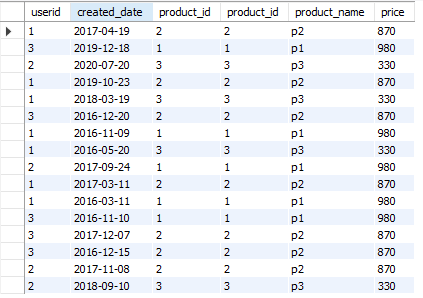
Questions:

1.what is total amount each customer spent on zomato ?

First take inner join of sales and product

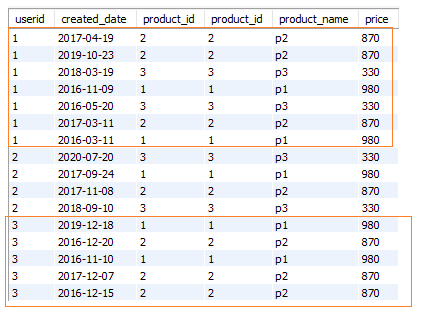
select \* from sales s inner join product p on

s.product\_id = p.product\_id;



select \* from sales s inner join product p on

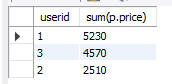
s.product\_id = p.product\_id order by s.userid;



select s.userid, sum(p.price) from sales s inner join product p on

s.product\_id = p.product\_id

group by s.userid;

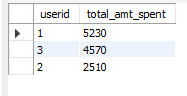


select s.userid, sum(p.price) as total\_amt\_spent

from sales s inner join product p on

s.product\_id = p.product\_id

group by s.userid;

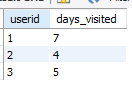


2.How many days has each customer visited zomato?

Distinct days customer visited zomato.

select userid, count(distinct created\_date) as days\_visited from sales

group by userid;



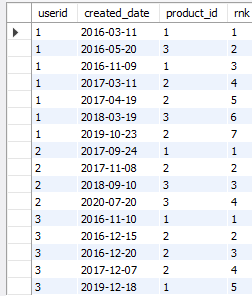
3. Which was the first product purchased by each customer?

Date needs to considered as the oldest. So need to order the date

select \*, rank() over

(partition by userid order by created\_date) as rnk

from sales;

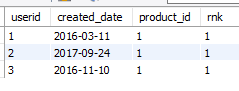


select \* from

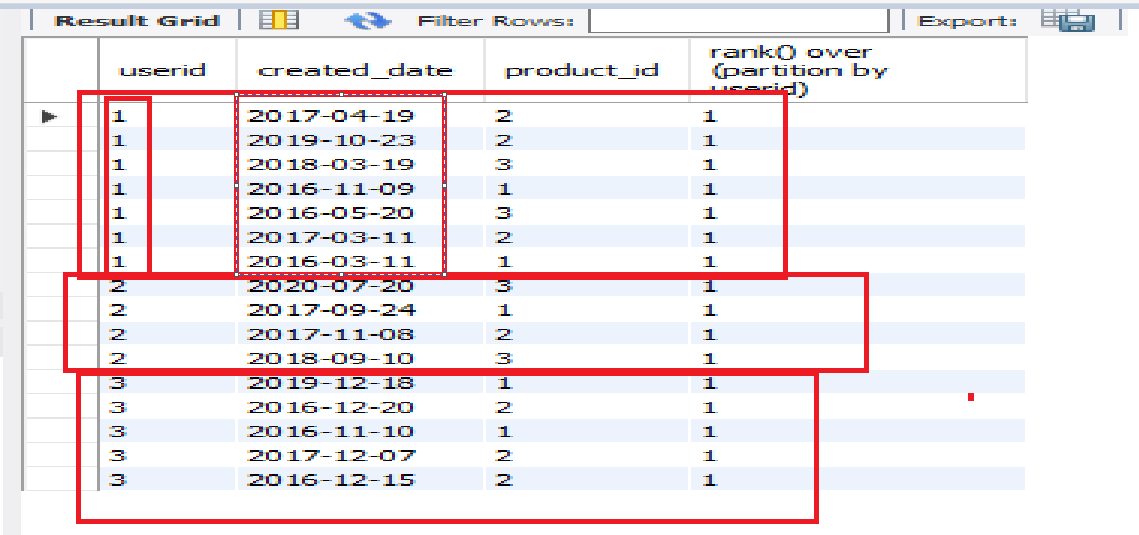
(select \*, rank() over

(partition by userid order by created\_date) as rnk

from sales) a where rnk = 1;



Extra:



select \*,

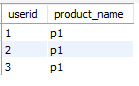
rank() over

(partition by userid)

from sales;

Final Query:

select p1.userid, p2.product\_name from   
 ( select \* from (select \*,   
rank() over   
(partition by userid   
order by created\_date asc) as rnk  
from sales) p  
where rnk = 1) p1 inner join product p2 on p1.product\_id=p2.product\_id;

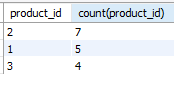


4. Which is the most purchased item on menu & how many times was it purchased by all customers ?

Most purchased products

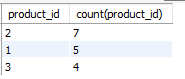
select product\_id, count(product\_id) from sales

group by product\_id;



select product\_id, count(product\_id) from sales

group by product\_id order by count(product\_id) desc;



Product\_id 2 is the highly bought product.

Display one 1 product highly bought

select product\_id, count(product\_id) as cnt from sales

group by product\_id order by count(product\_id) desc limit 1;



Only display the product\_id

select product\_id from sales

group by product\_id order by count(product\_id) desc limit 1;

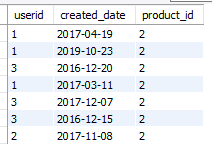


Now, how many times product\_id 2 was purchased by all customers

select \* from sales

where product\_id = (select product\_id from sales

group by product\_id order by count(product\_id) desc limit 1);



select userid, count(product\_id) as cnt from sales

where product\_id = (select product\_id from sales

group by product\_id order by count(product\_id) desc limit 1)

group by userid;



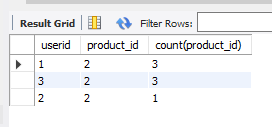
FINAL QUERY

select userid, product\_id, count(product\_id)

from sales where product\_id= (select product\_id from sales

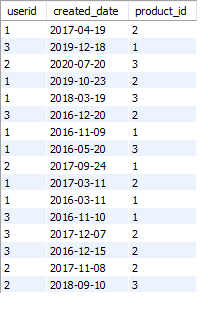
group by product\_id order by count(product\_id) desc limit 1)

group by userid, product\_id;



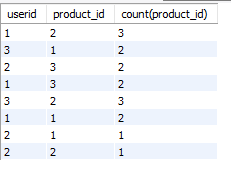
5.which item was most popular for each customer?

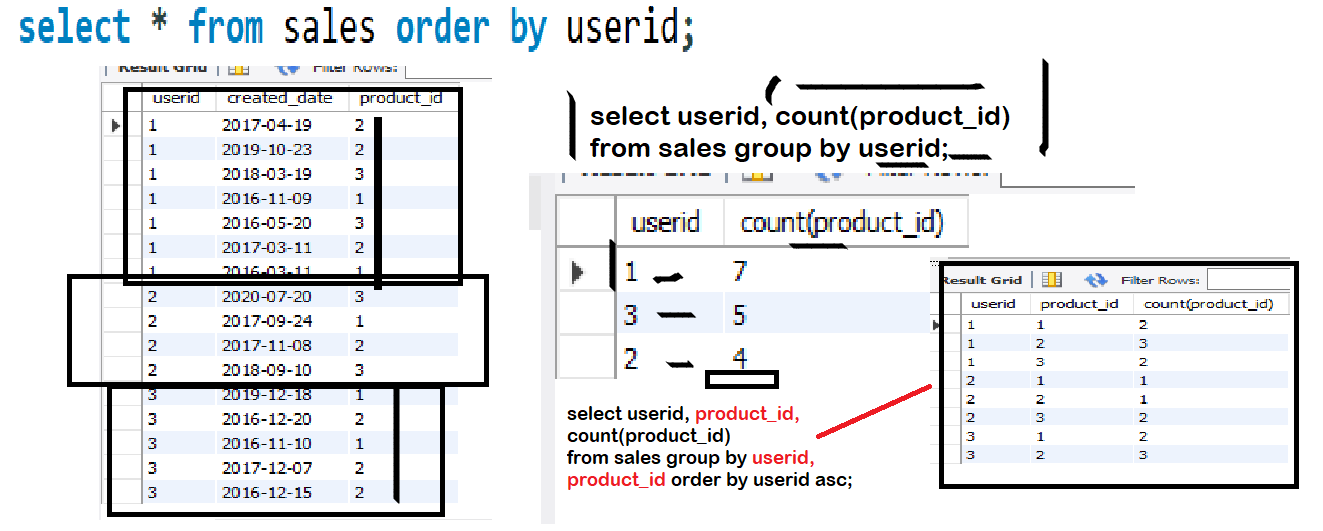
Sales



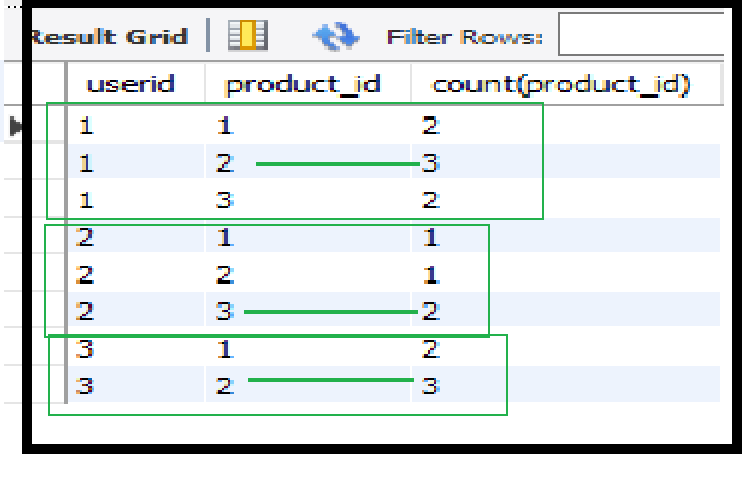
select userid, product\_id,

count(product\_id) from sales group by userid, product\_id;





Use partition



select \*, rank() over

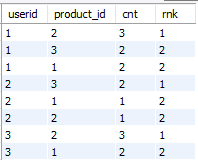
(partition by userid order by cnt desc) as rnk

from

(select userid, product\_id,

count(product\_id) as cnt from sales group by userid,product\_id) a;

Here, a is name of the derived table

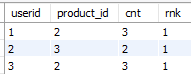


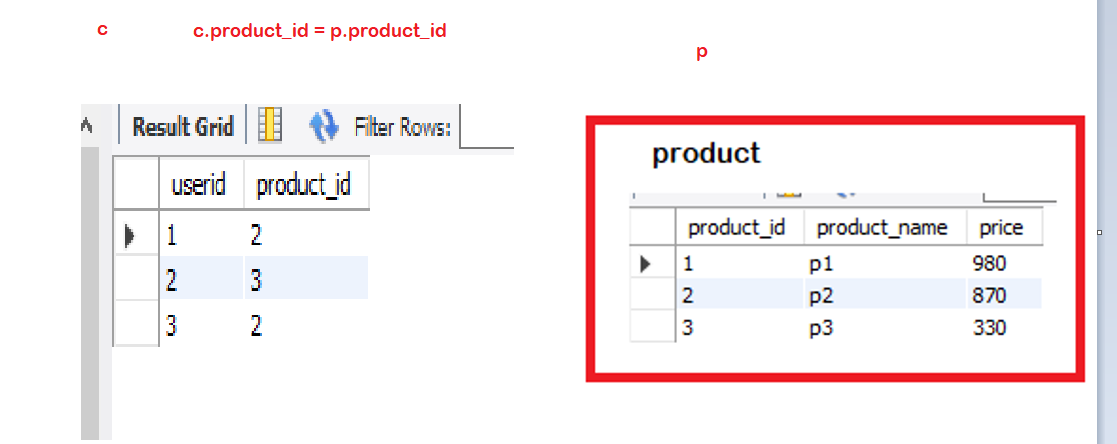
select \* from

(select \*, rank() over (partition by userid order by cnt desc) as rnk

from (select userid, product\_id, count(product\_id) as cnt from sales group by userid,product\_id) a)b

where rnk=1;





select c.userid, c.product\_id, p.product\_name from

(select userid, product\_id from (select \*, rank() over

(partition by userid order by cnt desc) as rnk

from

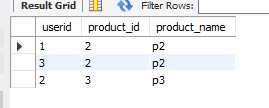
(select userid, product\_id,

count(product\_id) as cnt

from sales group by userid,product\_id) a) b where rnk= 1) c

inner join product p

on c.product\_id = p. product\_id;



6. Which item was purchased first by customer after they become a member ?

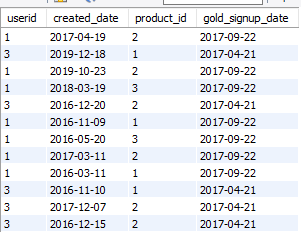
First only retrieve the members having gold membership

select a.userid, a.created\_date, a.product\_id,

b.gold\_signup\_date from

sales a inner join goldusers\_signup b

on a.userid = b.userid;



select \* from

(select a.userid, a.created\_date, a.product\_id,

b.gold\_signup\_date from

sales a inner join goldusers\_signup b

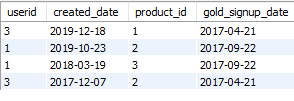
on a.userid = b.userid) p where created\_date > gold\_signup\_date;

select a.userid, a.created\_date, a.product\_id,

b.gold\_signup\_date from

sales a inner join goldusers\_signup b

on a.userid = b.userid and created\_date >= gold\_signup\_date;



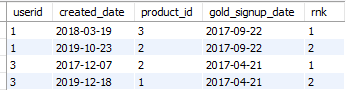
select c.\*, rank() over (partition by userid order by created\_date) rnk from

(select a.userid, a.created\_date, a.product\_id,

b.gold\_signup\_date from

sales a inner join goldusers\_signup b

on a.userid = b.userid and created\_date >= gold\_signup\_date) c;



select \* from

(select c.\*, rank() over (partition by userid order by created\_date) rnk from

(select a.userid, a.created\_date, a.product\_id,

b.gold\_signup\_date from

sales a inner join goldusers\_signup b

on a.userid = b.userid and created\_date >= gold\_signup\_date) c) d where rnk =1;



FINAL QUERY

select e.userid, e.created\_date, e.product\_id,

p.product\_name from

(select \* from (select c.\*, rank() over

(partition by userid

order by created\_date asc) rnk from

(select s.userid, s.created\_date,

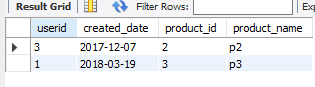
s.product\_id, p.gold\_signup\_date from

sales s inner join goldusers\_signup p

on s.userid = p.userid and

s.created\_date >= p.gold\_signup\_date) c) d where rnk = 1) e inner join product p

on e.product\_id = p.product\_id;



7. which item was purchased just before the customer became a member?

select \* from

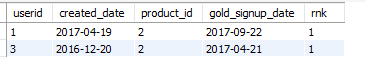
(select c.\*, rank() over (partition by userid order by created\_date desc) rnk from

(select a.userid, a.created\_date, a.product\_id,

b.gold\_signup\_date from

sales a inner join goldusers\_signup b

on a.userid = b.userid and created\_date <= gold\_signup\_date) c) d where rnk =1;



8. what is total orders and amount spent by each member before they become a member?

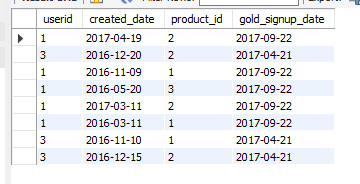
Details of members before they get a gold membership

select a.userid, a.created\_date, a.product\_id,

b.gold\_signup\_date from

sales a inner join goldusers\_signup b

on a.userid = b.userid and created\_date <= gold\_signup\_date;



Now assume above table as c. We need to get the price associated with above products. So take an inner join of above table c with product table.

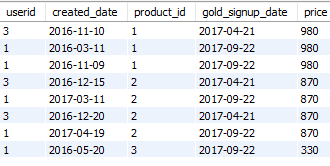
select c.\*, d.price from

(select a.userid, a.created\_date, a.product\_id,

b.gold\_signup\_date from sales a inner join goldusers\_signup b

on a.userid = b.userid and created\_date <= gold\_signup\_date) c inner join product d

on c.product\_id = d.product\_id;



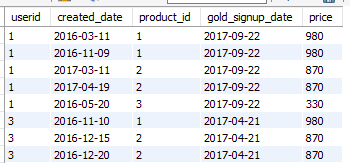
select c.\*, d.price from

(select a.userid, a.created\_date, a.product\_id,

b.gold\_signup\_date from sales a inner join goldusers\_signup b

on a.userid = b.userid and created\_date <= gold\_signup\_date) c inner join product d

on c.product\_id = d.product\_id order by userid;



Now from above table, count the number of orders purchased by each customer and also give total price by grouping the userid.

select userid, count(created\_date) as count\_of\_purchased\_orders,

sum(price) as total\_amt\_spent from

(select c.\*, d.price from

(select a.userid, a.created\_date, a.product\_id,

b.gold\_signup\_date from sales a inner join goldusers\_signup b

on a.userid = b.userid and created\_date <= gold\_signup\_date) c inner join product d

on c.product\_id = d.product\_id)e

group by userid;

