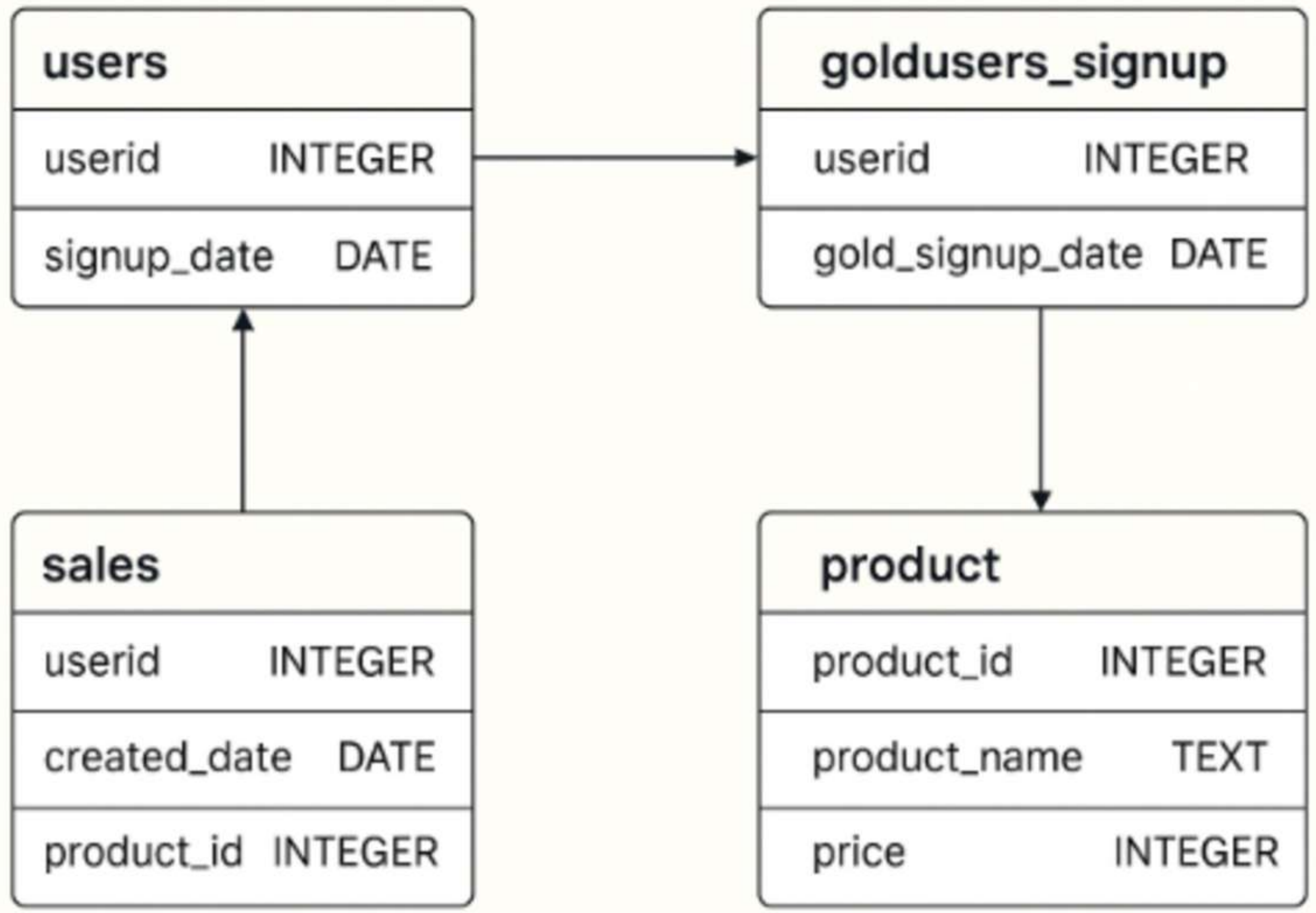
ZOMATO

Presented by Anurag Shaw

ABSTRACT

This project models a simplified database for a food delivery app like Zomato. It tracks user signups, premium (Gold) memberships, orders, and food items. The schema supports analysis of user activity, sales trends, and the impact of loyalty programs on purchases.

SCHEMA



users	
userid	INTEGER
signup_date	DATE

goldusers_signup	
userid	INTEGER
gold_signup_date	DATE

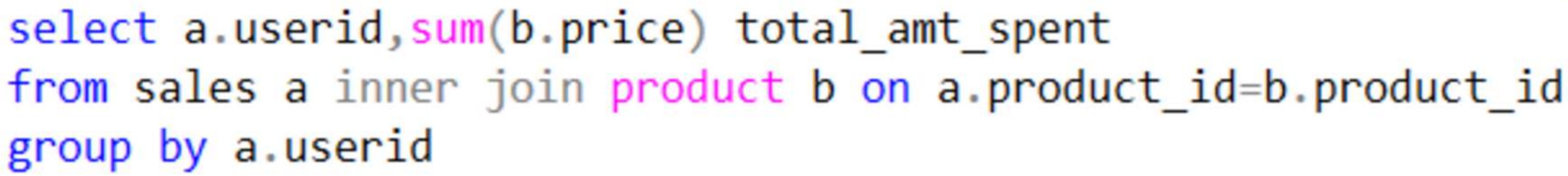
sales	
userid	INTEGER
created_date	DATE
product_id	INTEGER

product	
product_id	INTEGER
product_name	TEXT
price	INTEGER

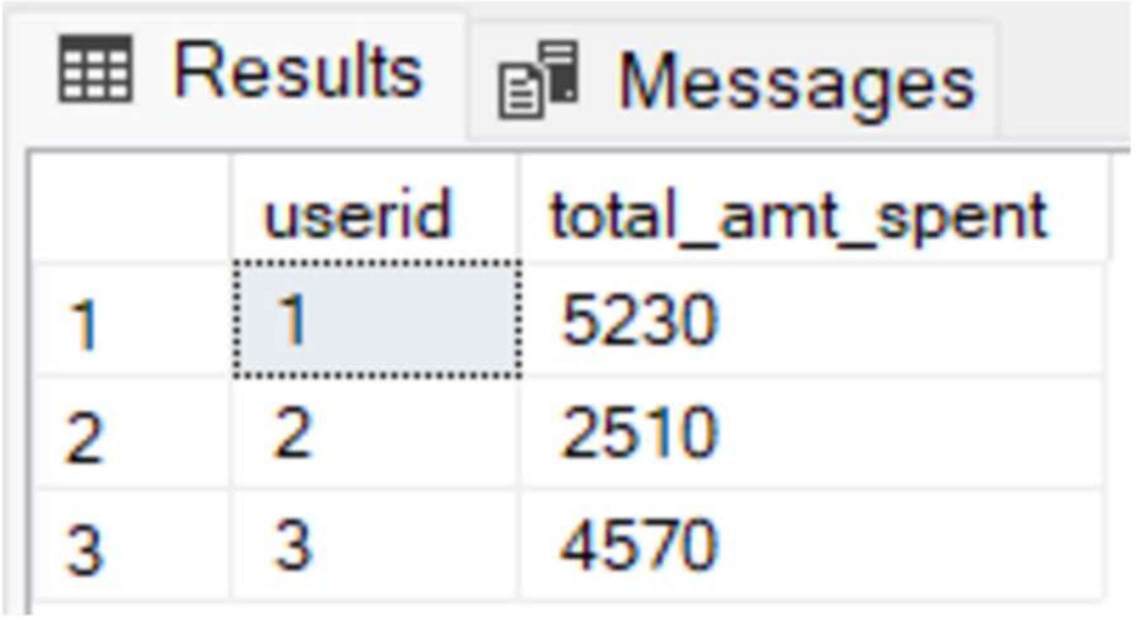
GOALS

To design a relational database for a food delivery app that tracks users, orders, premium memberships, and products to enable effective data analysis and business insights.

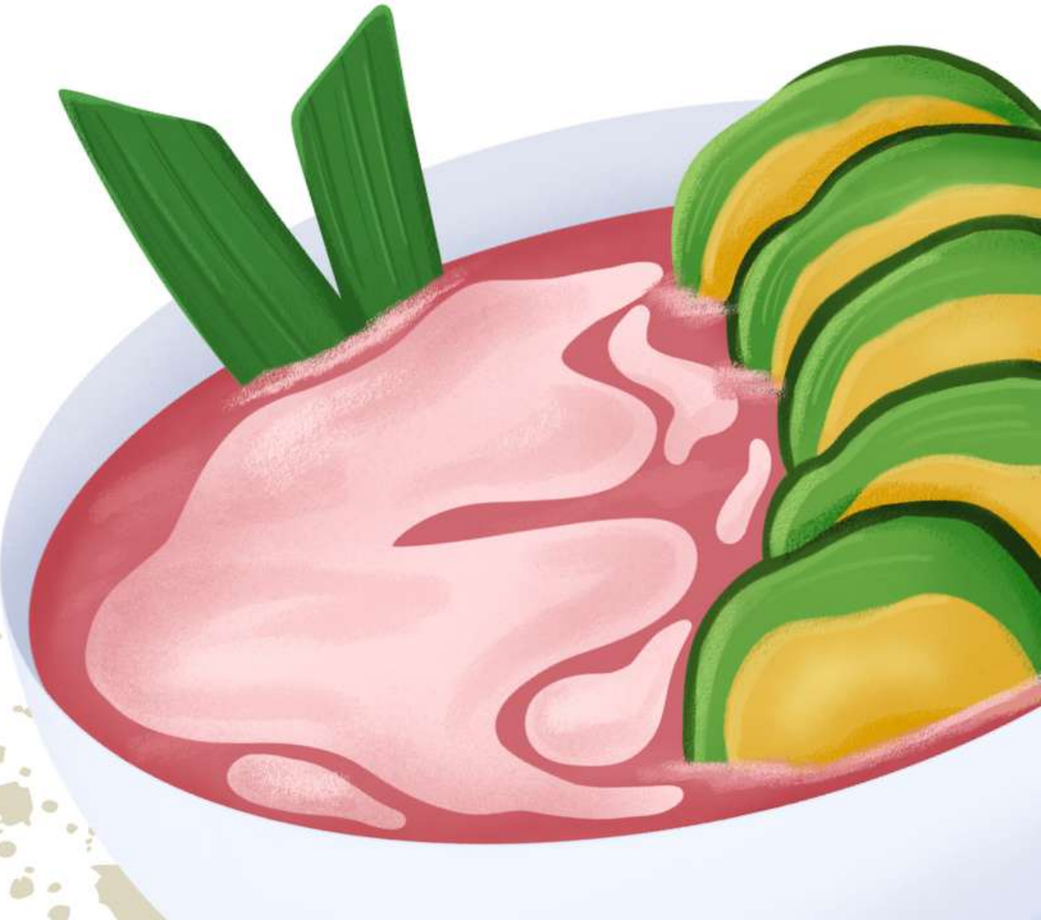
WHAT IS THE TOTAL AMOUNT EACH CUSTOMER SPENT ON ZOMATO?



```
select a.userid,sum(b.price) total_amt_spent
from sales a inner join product b on a.product_id=b.product_id
group by a.userid
```

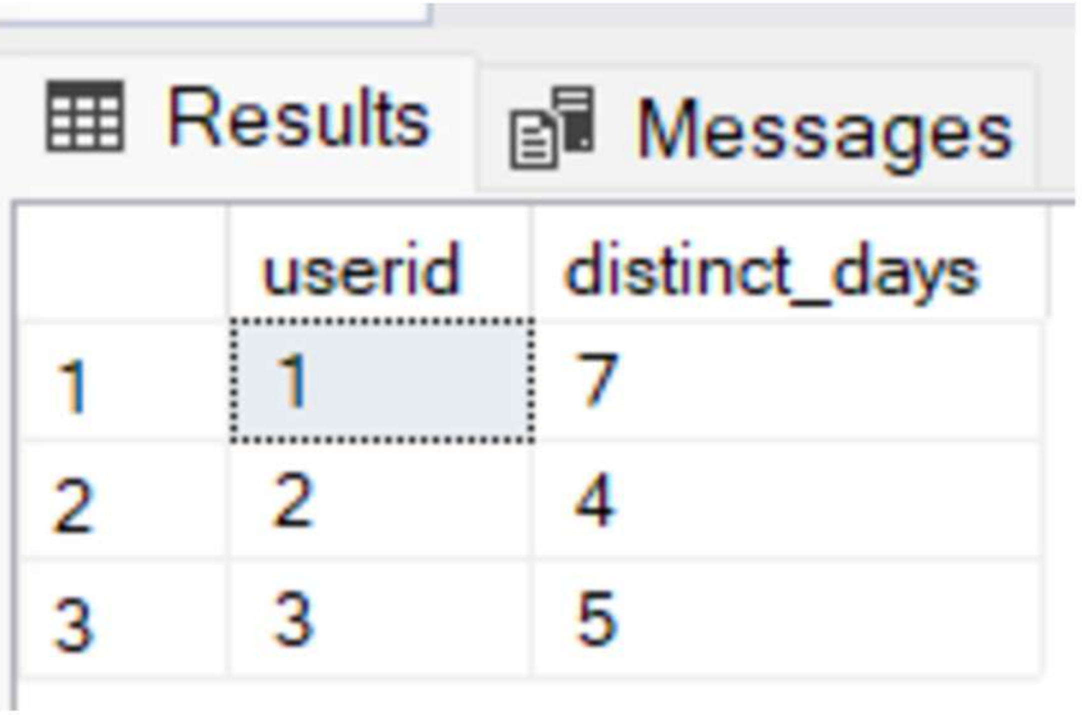


	userid	total_amt_spent
1	1	5230
2	2	2510
3	3	4570

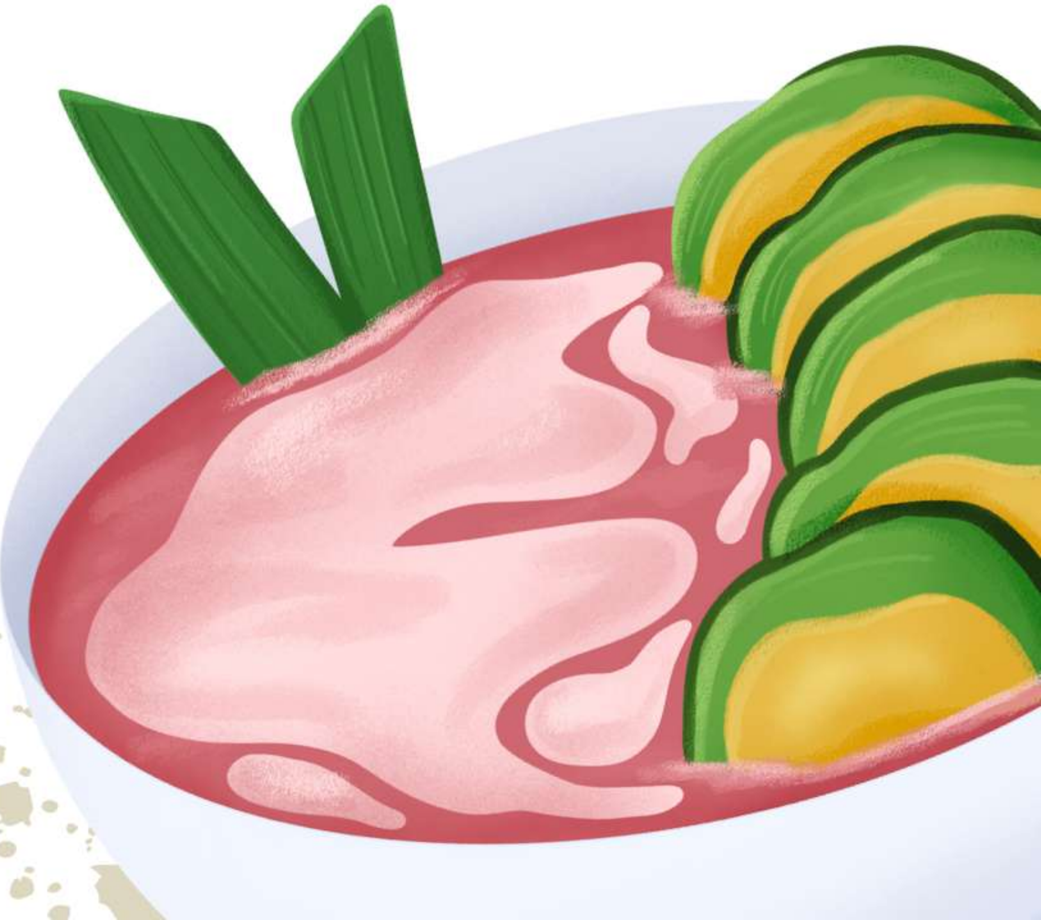


HOW MANY DAYS HAS EACH CUSTOMER VISITED ZOMATO?

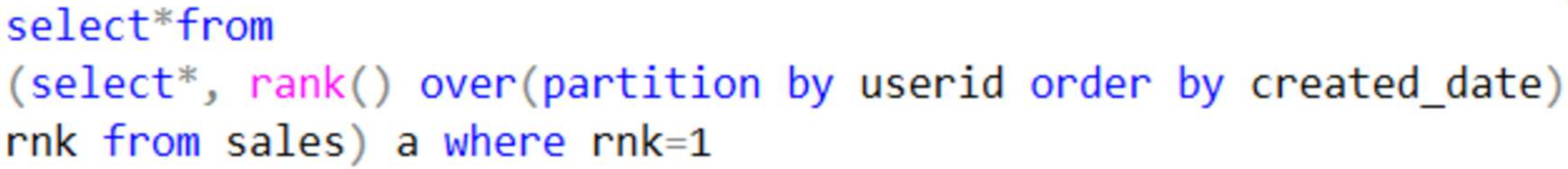
```
select userid, count(distinct created_date)
distinct_days from sales group by userid;
```



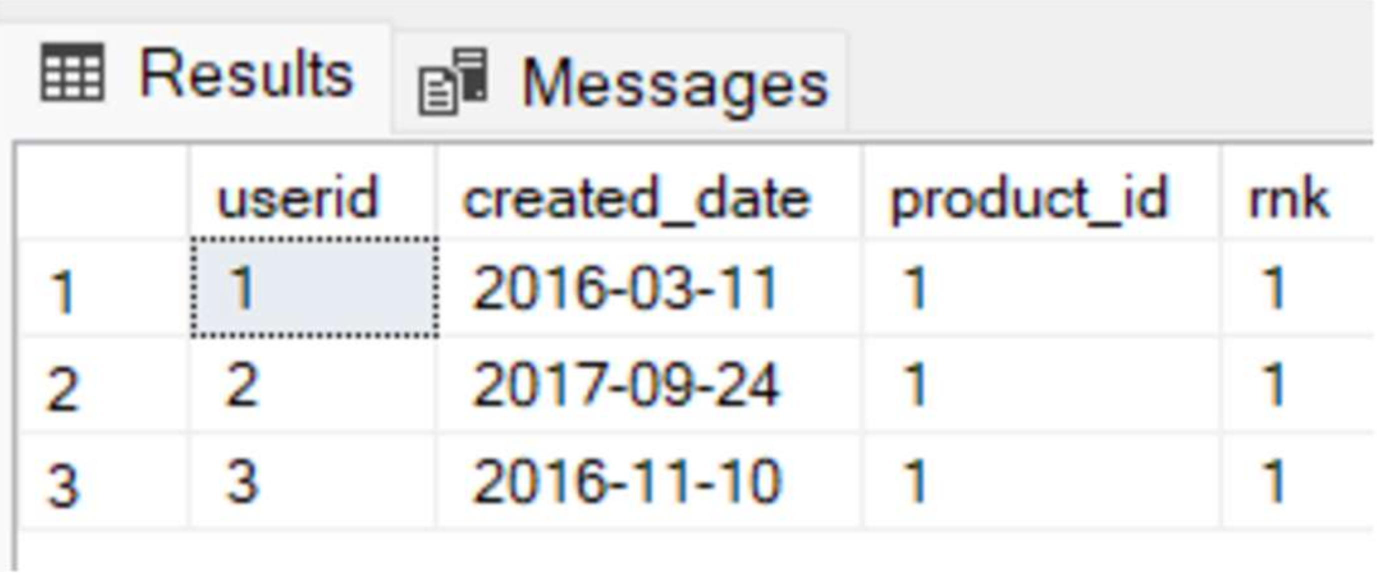
	userid	distinct_days
1	1	7
2	2	4
3	3	5



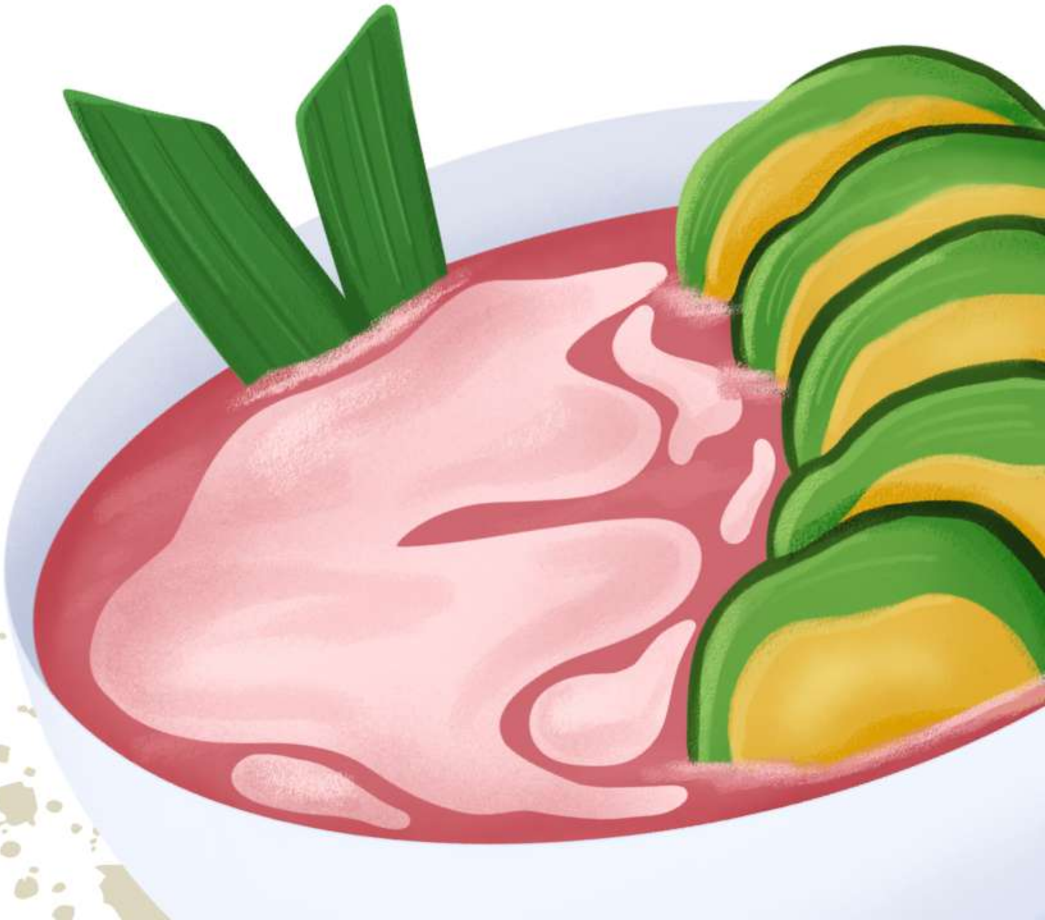
WHAT WAS THE FIRST PRODUCT PURCHASED BY EACH CUSTOMER?



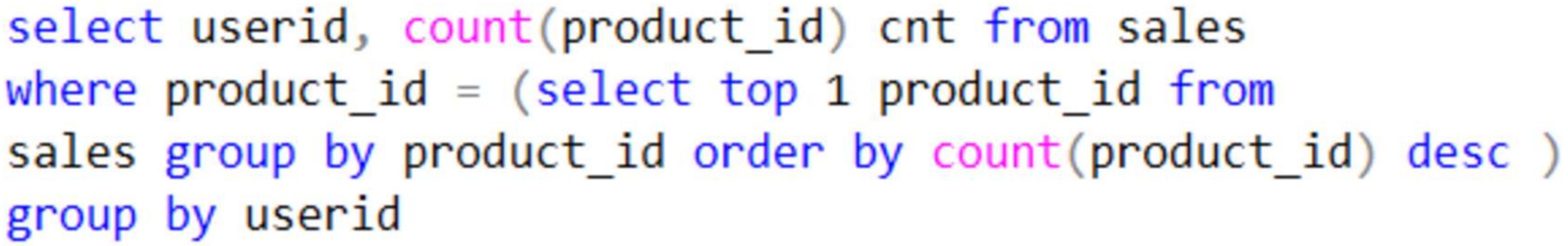
```
select*from
(select*, rank() over(partition by userid order by created_date)
rnk from sales) a where rnk=1
```



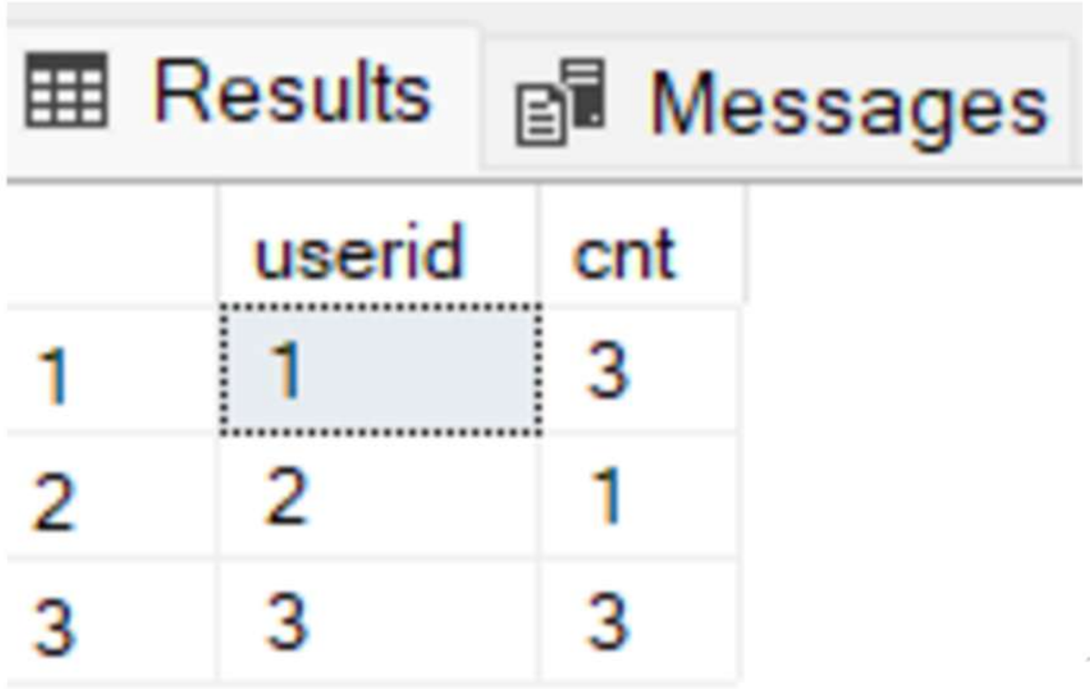
	userid	created_date	product_id	rnk
1	1	2016-03-11	1	1
2	2	2017-09-24	1	1
3	3	2016-11-10	1	1



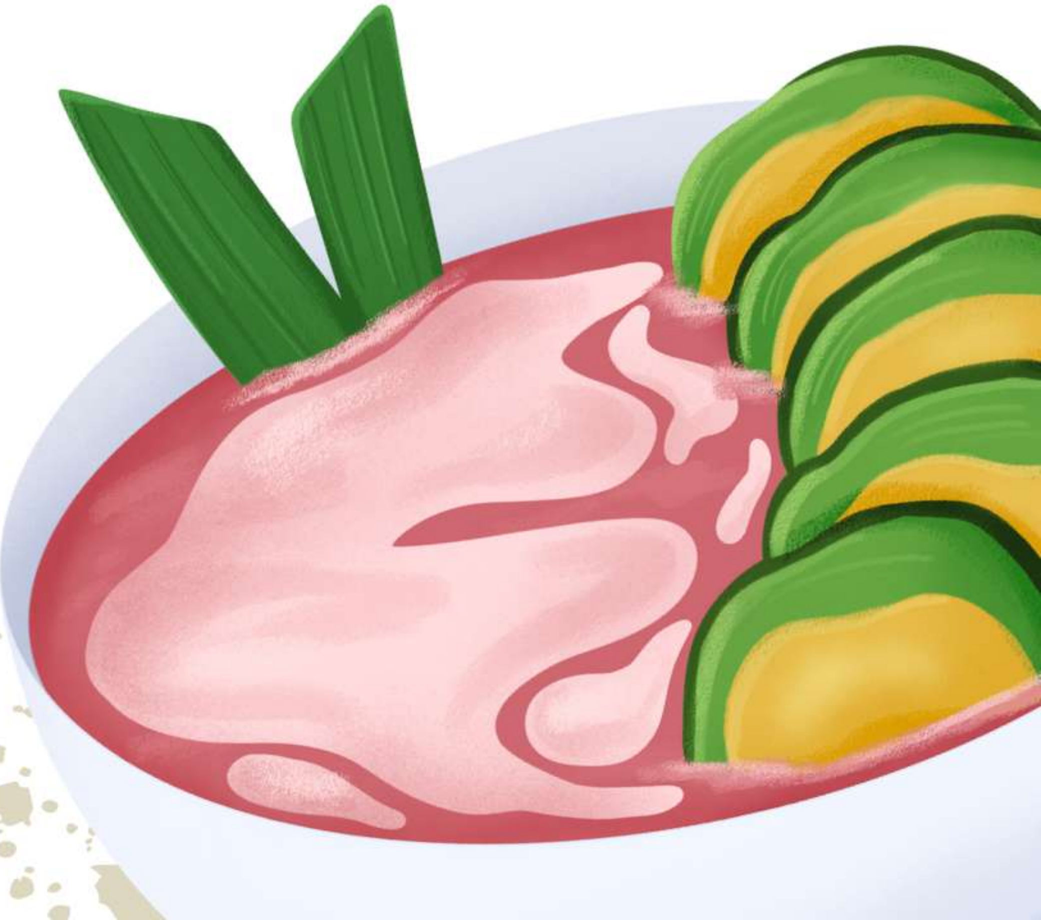
WHAT IS THE MOST PURCHASED ITEM ON THE MENU AND HOW MANY TIMES WAS IT PURCHASED BY ALL CUSTOMERS?



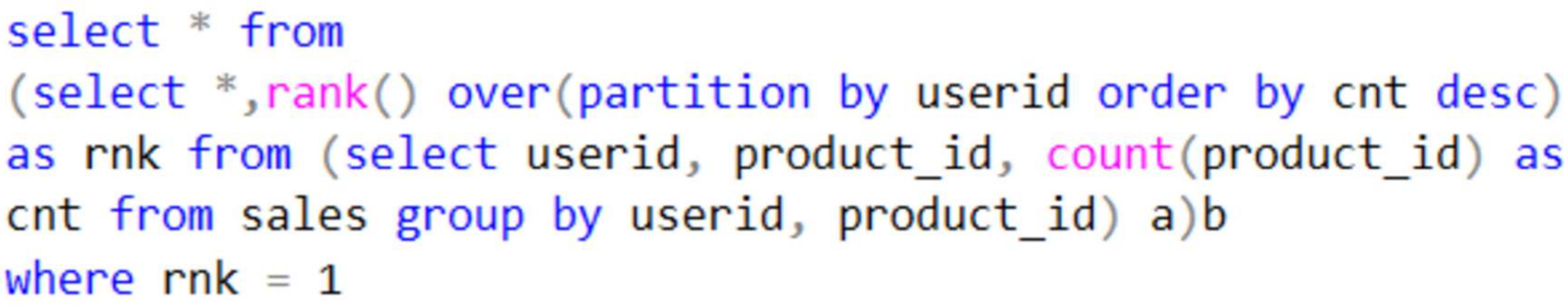
```
select userid, count(product_id) cnt from sales
where product_id = (select top 1 product_id from
sales group by product_id order by count(product_id) desc )
group by userid
```



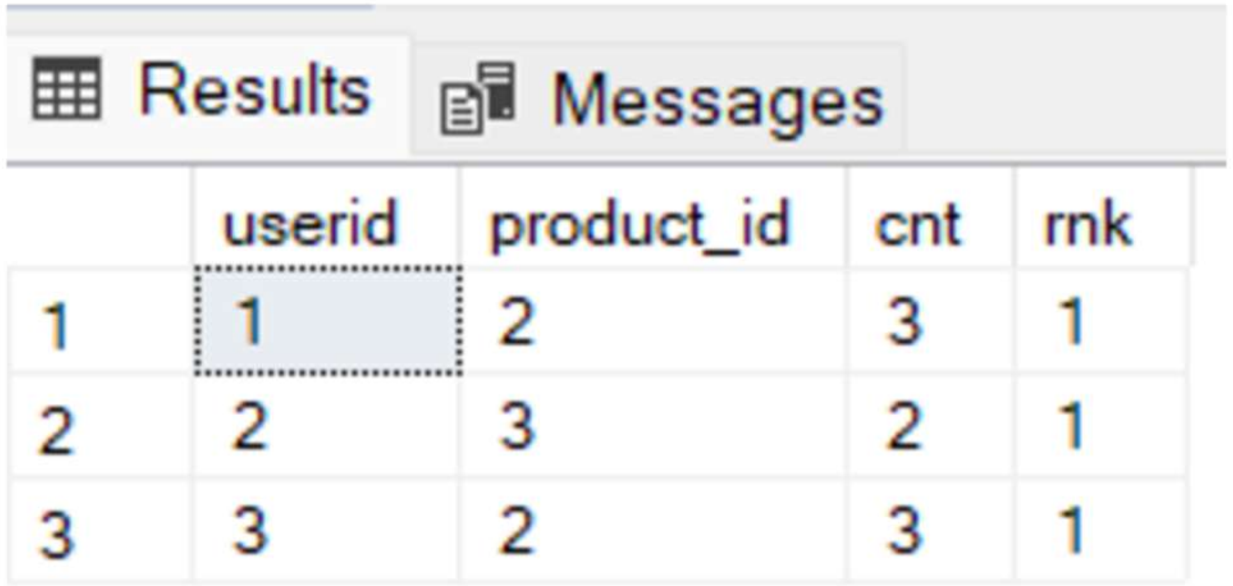
	userid	cnt
1	1	3
2	2	1
3	3	3



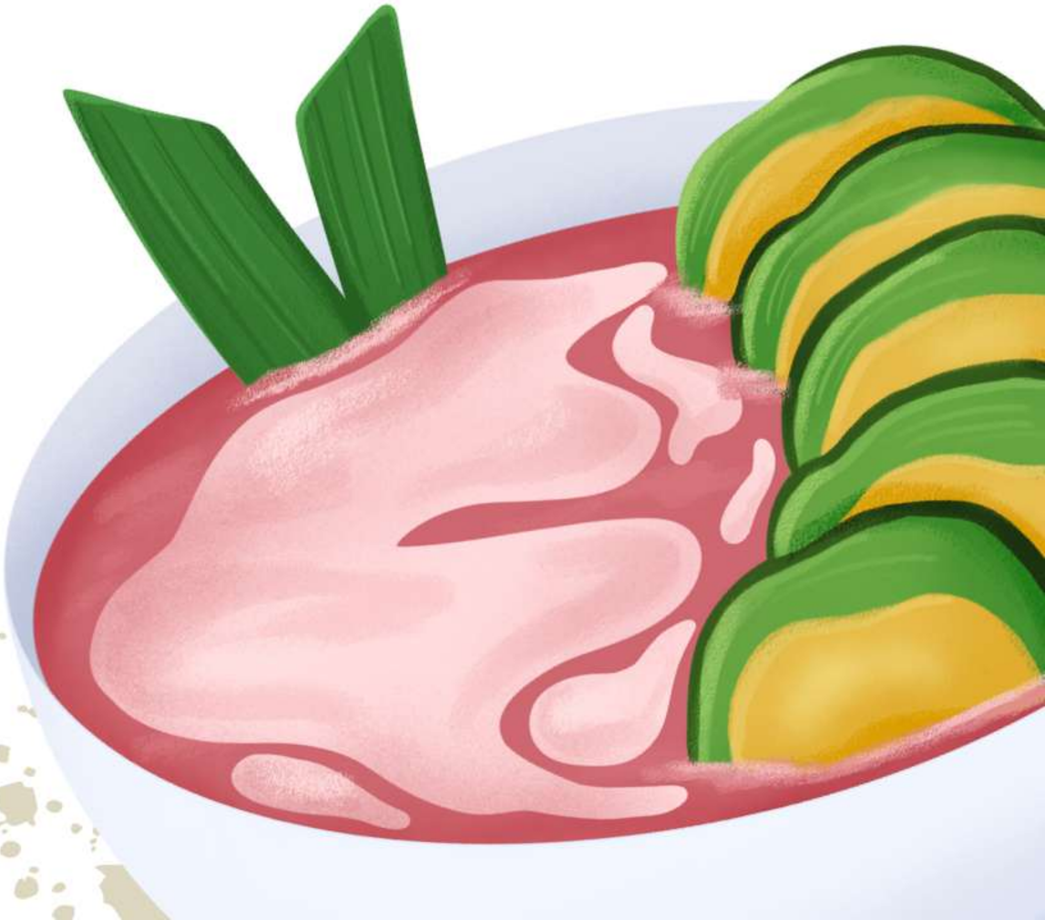
WHICH ITEM WAS THE MOST POPULAR FOR EACH CUSTOMER?



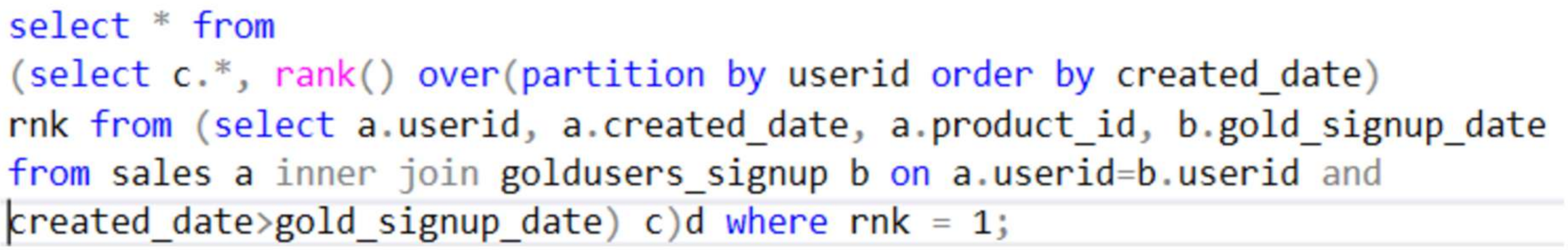
```
select * from
(select *,rank() over(partition by userid order by cnt desc)
as rnk from (select userid, product_id, count(product_id) as
cnt from sales group by userid, product_id) a)b
where rnk = 1
```



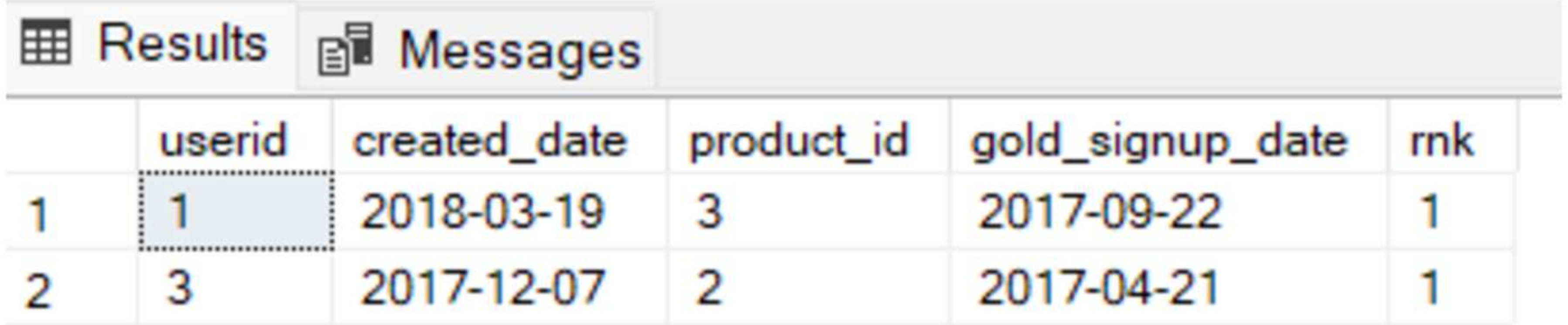
	userid	product_id	cnt	rnk
1	1	2	3	1
2	2	3	2	1
3	3	2	3	1



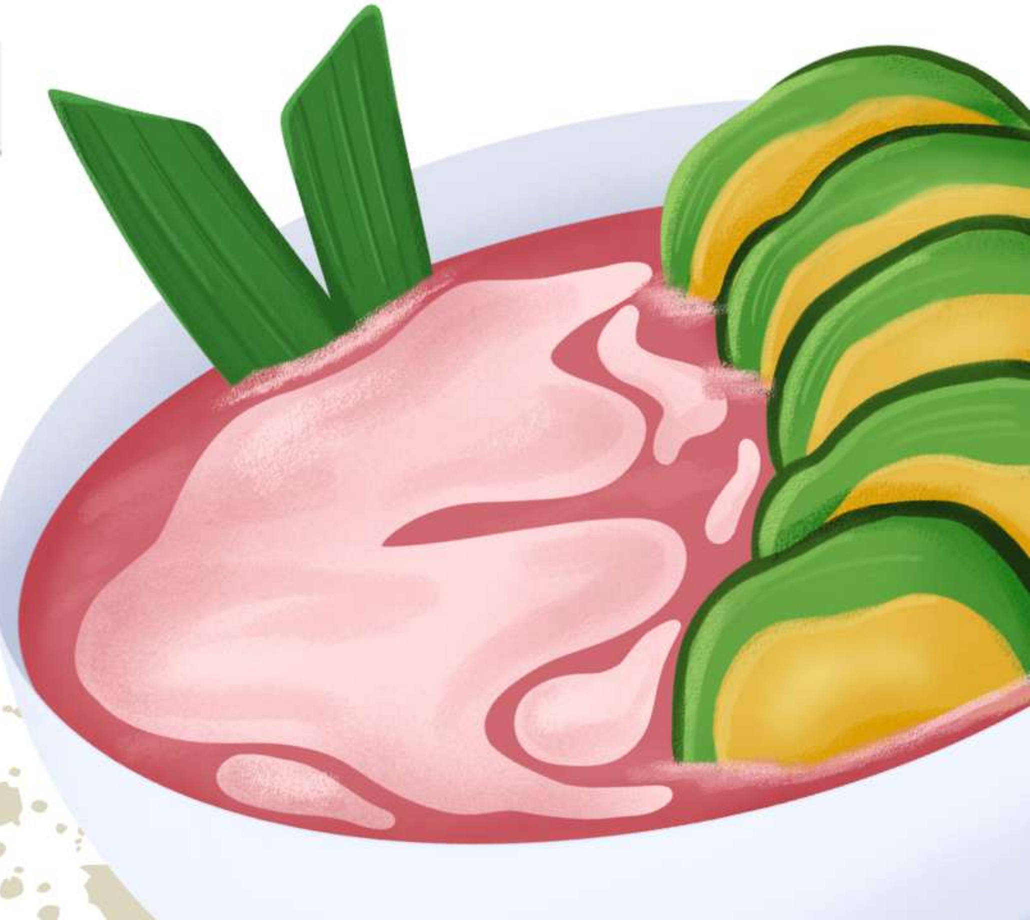
WHICH ITEM WAS PURCHASED FIRST BY THE CUSTOMER AFTER THEY BECAME A MEMBER?



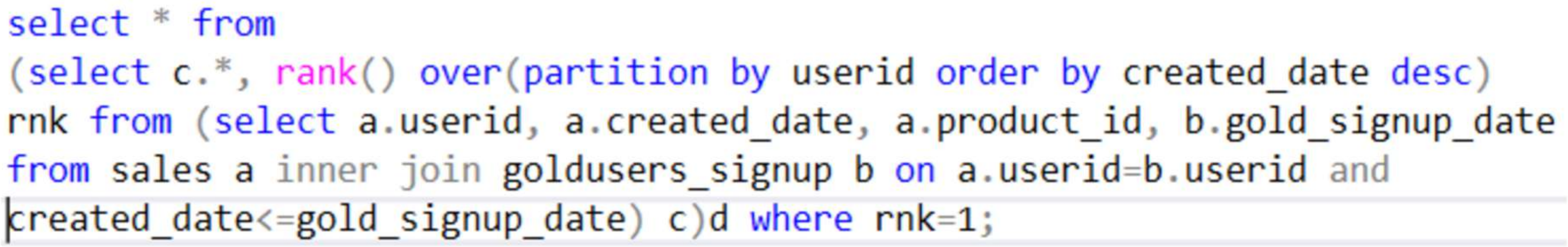
```
select * from
(select c.*, rank() over(partition by userid order by created_date)
rnk from (select a.userid, a.created_date, a.product_id, b.gold_signup_date
from sales a inner join goldusers_signup b on a.userid=b.userid and
created_date>gold_signup_date) c)d where rnk = 1;
```



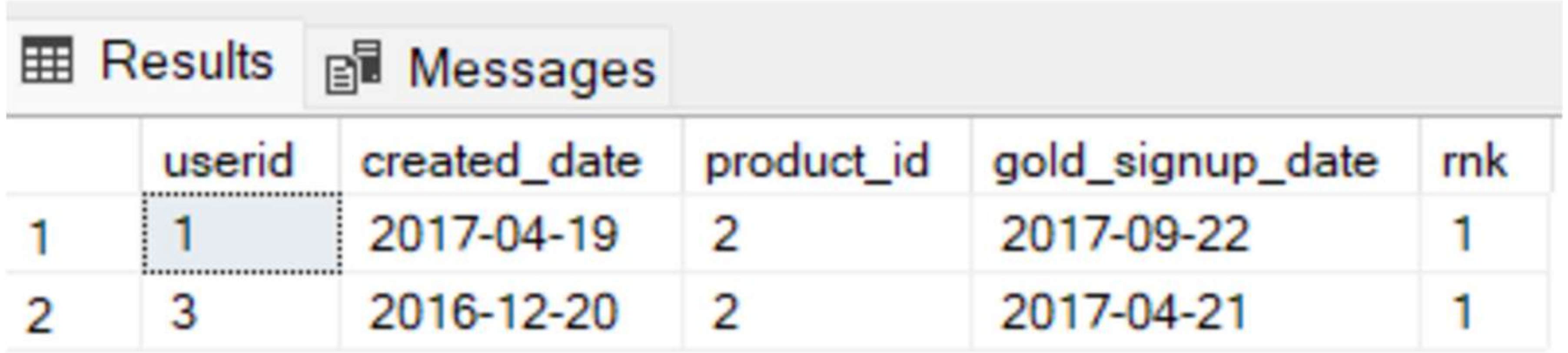
	userid	created_date	product_id	gold_signup_date	rnk
1	1	2018-03-19	3	2017-09-22	1
2	3	2017-12-07	2	2017-04-21	1



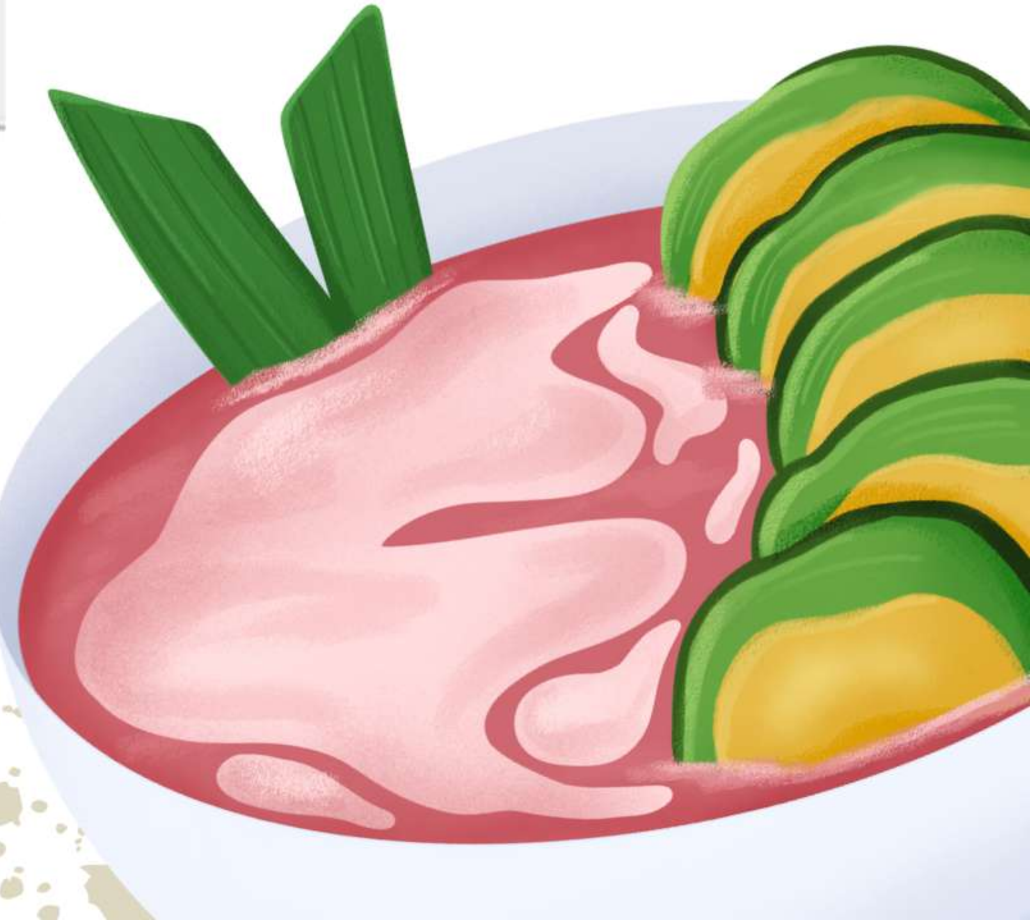
WHICH ITEM WAS PURCHASED JUST BEFORE THE CUSTOMER BECAME A MEMBER?



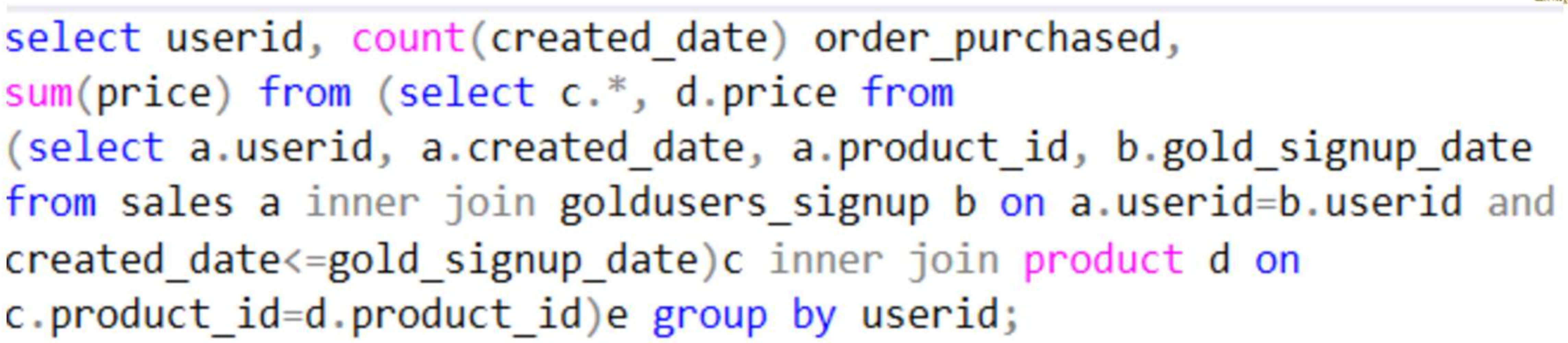
```
select * from
(select c.*, rank() over(partition by userid order by created_date desc)
rnk from (select a.userid, a.created_date, a.product_id, b.gold_signup_date
from sales a inner join goldusers_signup b on a.userid=b.userid and
created_date<=gold_signup_date) c)d where rnk=1;
```



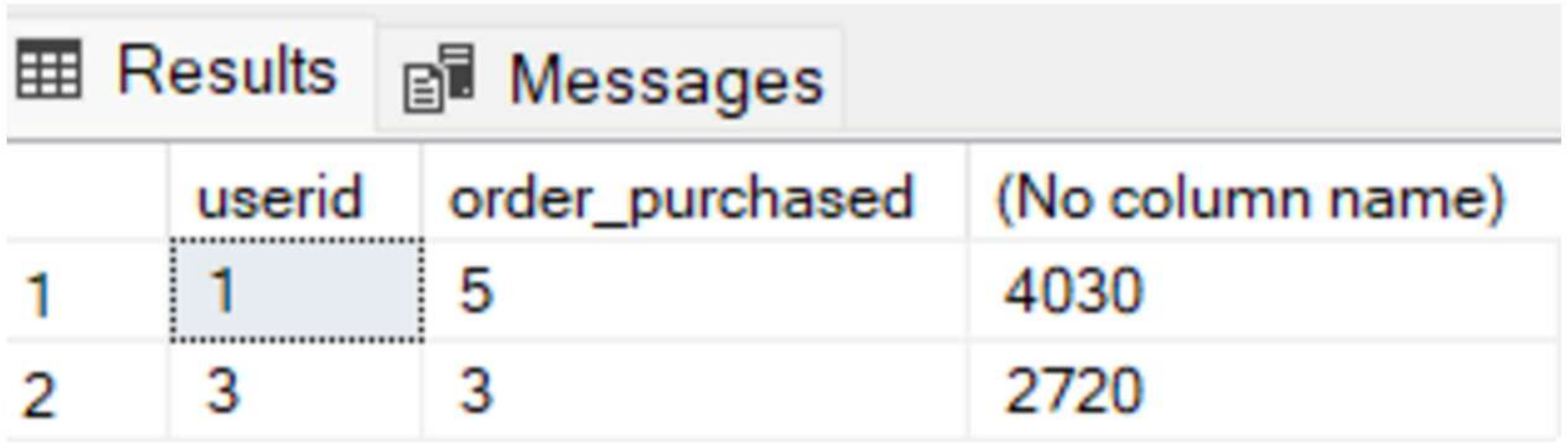
	userid	created_date	product_id	gold_signup_date	rnk
1	1	2017-04-19	2	2017-09-22	1
2	3	2016-12-20	2	2017-04-21	1



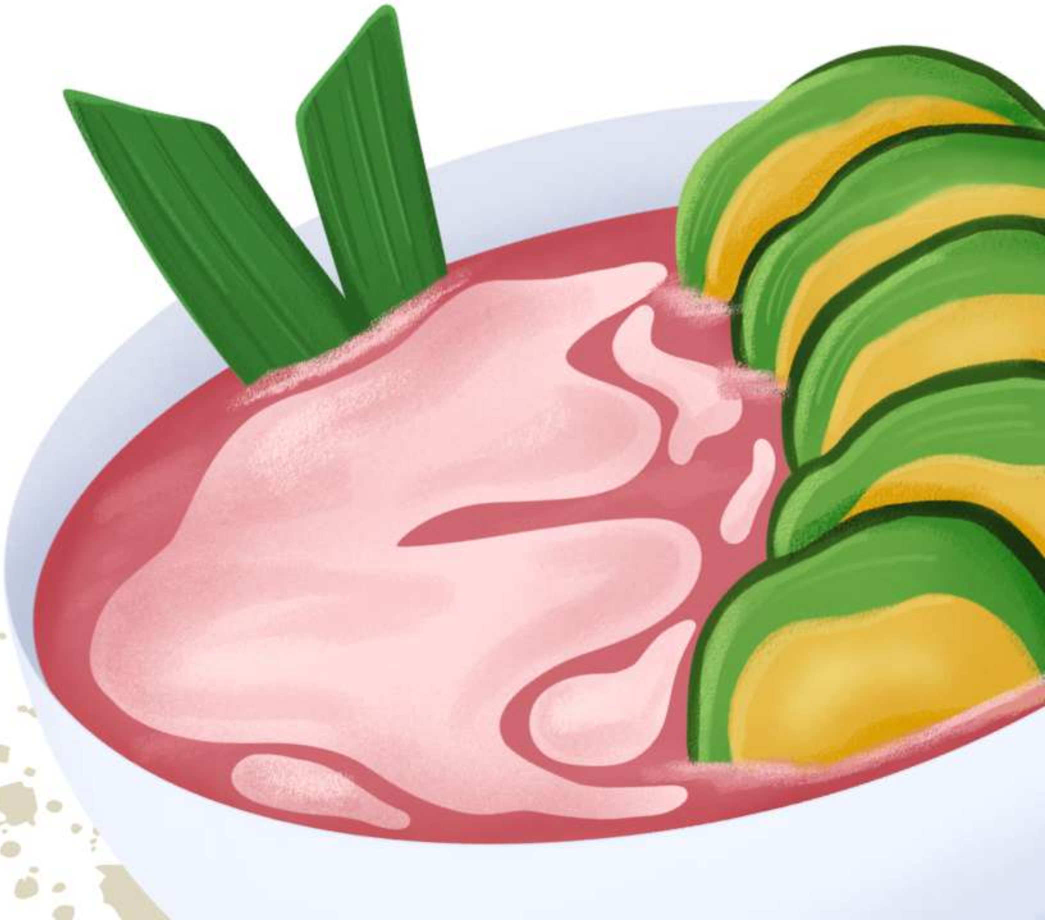
WHAT IS THE TOTAL ORDERS AND AMOUNT SPENT FOR EACH MEMBER BEFORE THEY BECAME A MEMBER?



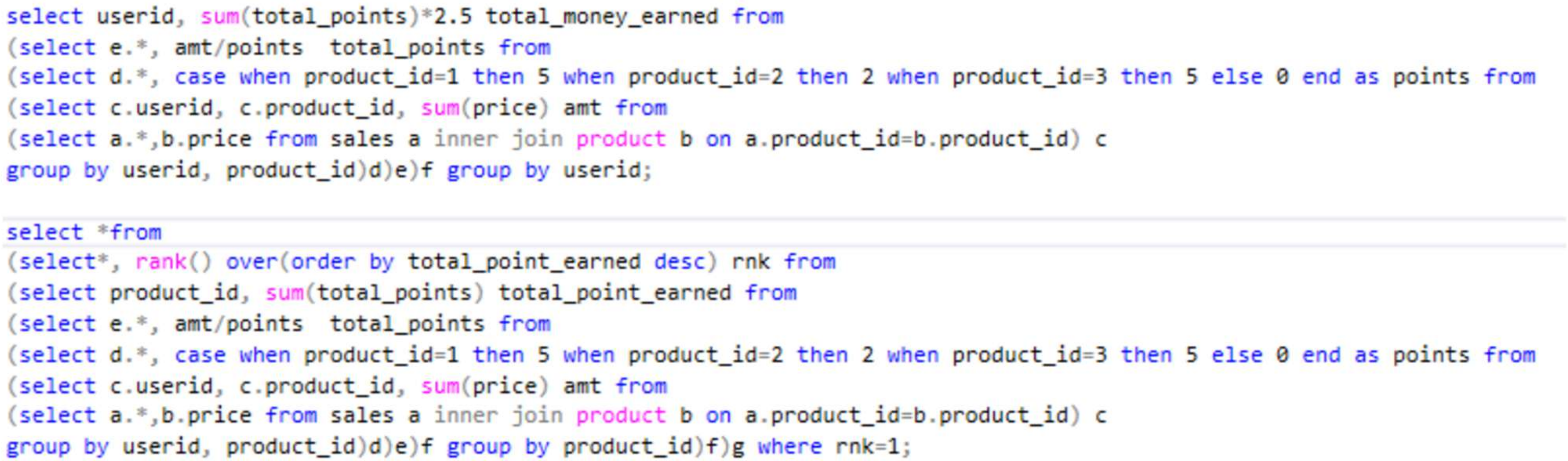
```
select userid, count(created_date) order_purchased,
sum(price) from (select c.*, d.price from
(select a.userid, a.created_date, a.product_id, b.gold_signup_date
from sales a inner join goldusers_signup b on a.userid=b.userid and
created_date<=gold_signup_date)c inner join product d on
c.product_id=d.product_id)e group by userid;
```



	userid	order_purchased	(No column name)
1	1	5	4030
2	3	3	2720

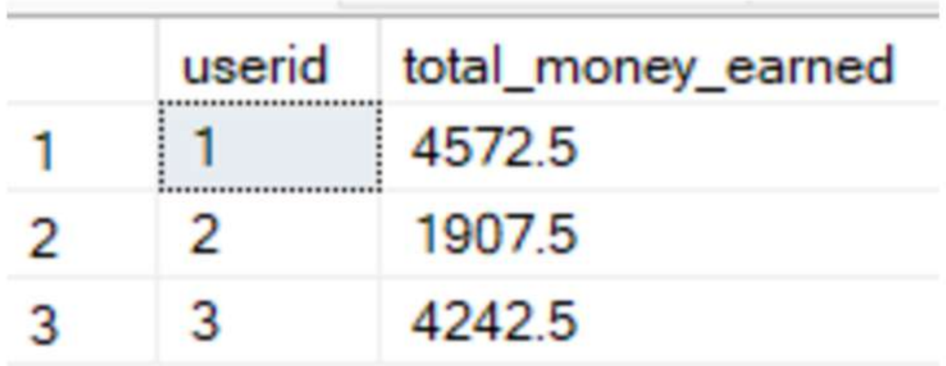


IF BUYING EACH PRODUCT GENERATES POINTS FOR EG 5RS=2 ZOMATO POINT AND EACH PRODUCT HAS DIFFERENT PURCHASING POINTS FOR EG FOR 5RS=1 ZOMATO POINT, FOR P2 10RS=5ZOMATO POINT AND P3 5RS=1 ZOMATO POINT, CALCULATE POINTS COLLECTED BY EACH CUSTOMERS AND FOR WHICH PRODUCT MOST POINTS HAVE BEEN GIVEN TILL NOW

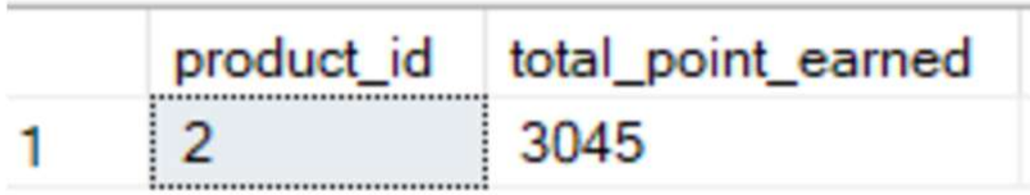


```sql
select userid, sum(total_points)*2.5 total_money_earned from
(select e.*, amt/points  total_points from
(select d.*, case when product_id=1 then 5 when product_id=2 then 2 when product_id=3 then 5 else 0 end as points from
(select c.userid, c.product_id, sum(price) amt from
(select a.*,b.price from sales a inner join product b on a.product_id=b.product_id) c
group by userid, product_id)d)e)f group by userid;

select *from
(select*, rank() over(order by total_point_earned desc) rnk from
(select product_id, sum(total_points) total_point_earned from
(select e.*, amt/points  total_points from
(select d.*, case when product_id=1 then 5 when product_id=2 then 2 when product_id=3 then 5 else 0 end as points from
(select c.userid, c.product_id, sum(price) amt from
(select a.*,b.price from sales a inner join product b on a.product_id=b.product_id) c
group by userid, product_id)d)e)f group by product_id)f)g where rnk=1;
```

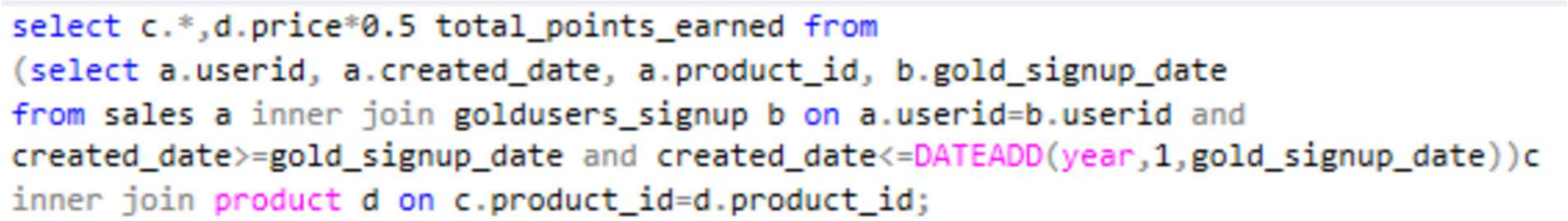


	userid	total_money_earned
1	1	4572.5
2	2	1907.5
3	3	4242.5

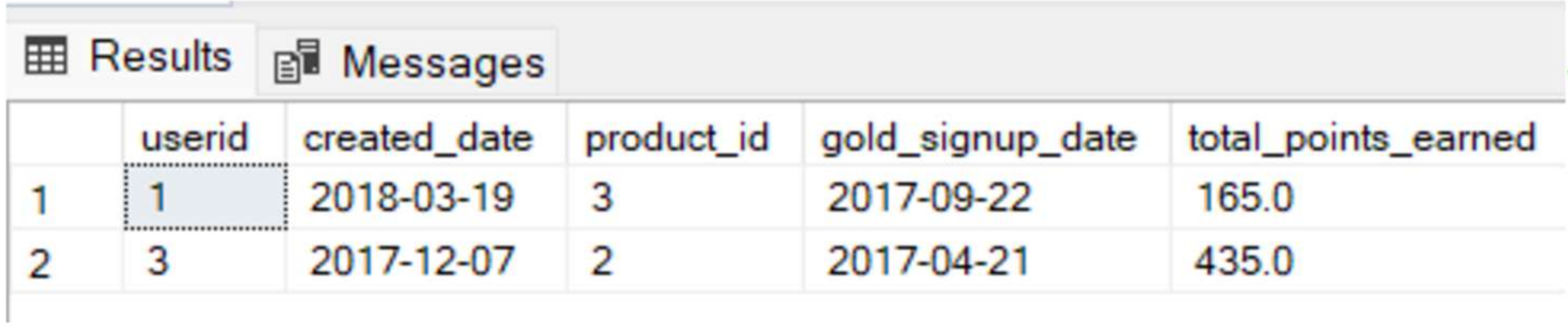


	product_id	total_point_earned
1	2	3045

IN THE FIRST ONE YEAR AFTER A CUSTOMER JOINS THE GOLD PROGRAM(INCLUDING THEIR JOIN DATE) IRRESPECTIVE OF WHAT THE CUSTOMER HAS PURCHASED THEY EARN 5 ZOMATO POINTS FOR EVERY 10RS SPENT WHO EARNED MORE 1 OR 3 AND WHAT WAS THEIR POINTS EARNING IN THEIR FIRST YEAR?
1 ZOMATO PT=2RS => 0.5 ZOMATO PT = 1RS

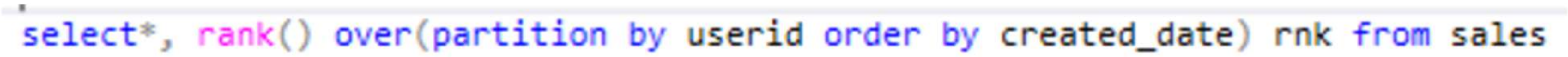


```
select c.*,d.price*0.5 total_points_earned from
(select a.userid, a.created_date, a.product_id, b.gold_signup_date
from sales a inner join goldusers_signup b on a.userid=b.userid and
created_date>=gold_signup_date and created_date<=DATEADD(year,1,gold_signup_date))c
inner join product d on c.product_id=d.product_id;
```

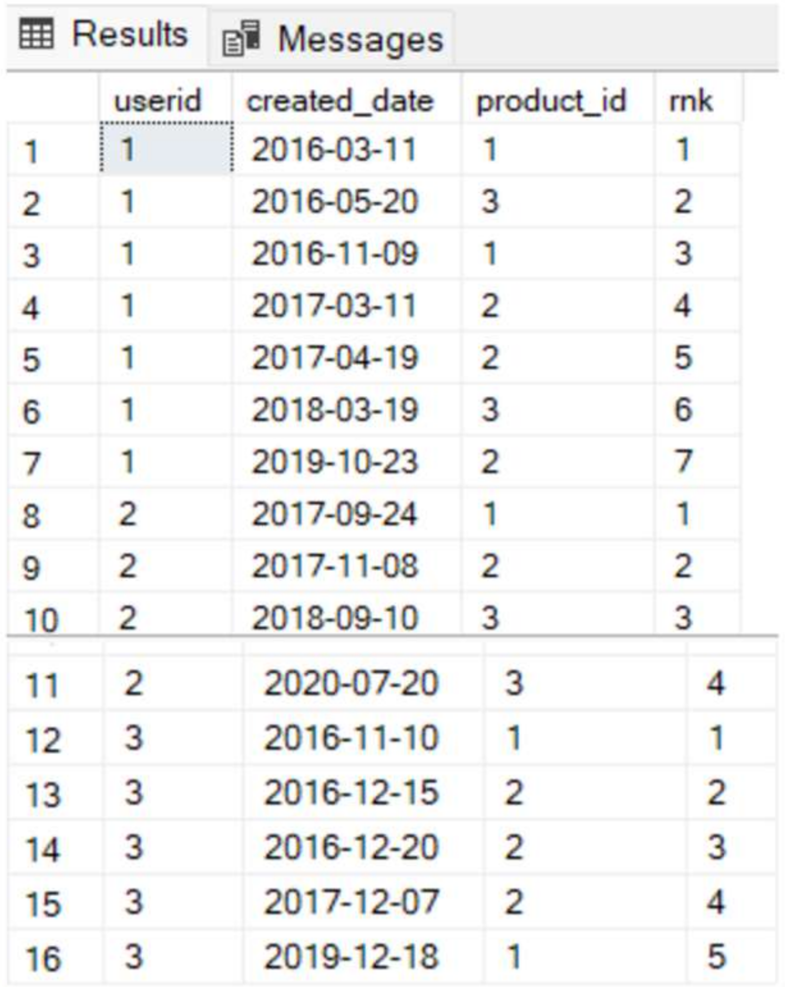


	userid	created_date	product_id	gold_signup_date	total_points_earned
1	1	2018-03-19	3	2017-09-22	165.0
2	3	2017-12-07	2	2017-04-21	435.0

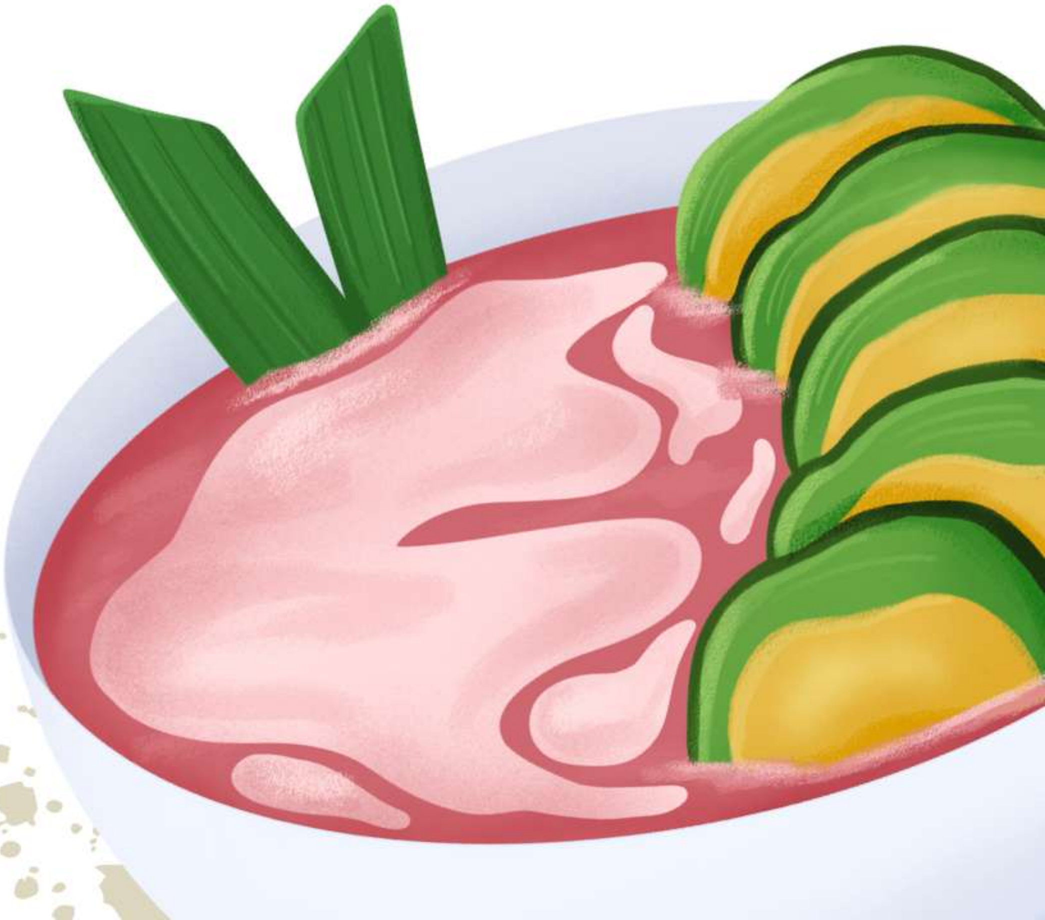
RANK ALL THE TRANSACTION OF THE CUSTOMERS



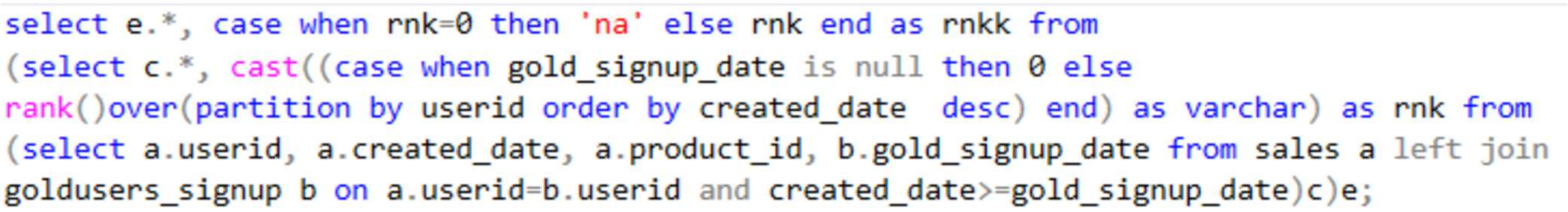
```
select*, rank() over(partition by userid order by created_date) rnk from sales
```



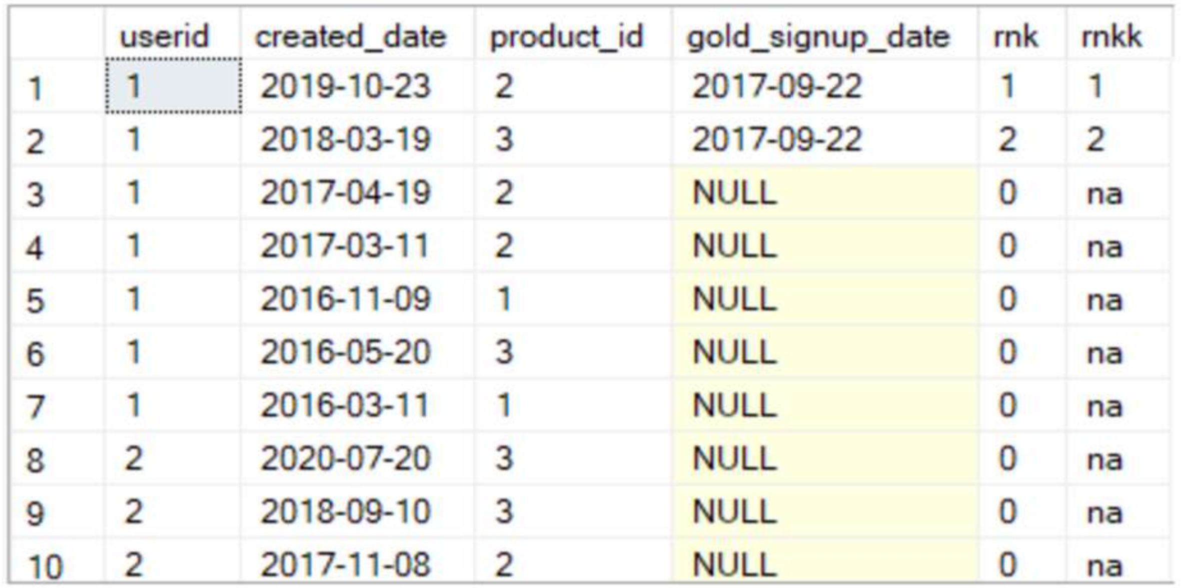
	userid	created_date	product_id	rnk
1	1	2016-03-11	1	1
2	1	2016-05-20	3	2
3	1	2016-11-09	1	3
4	1	2017-03-11	2	4
5	1	2017-04-19	2	5
6	1	2018-03-19	3	6
7	1	2019-10-23	2	7
8	2	2017-09-24	1	1
9	2	2017-11-08	2	2
10	2	2018-09-10	3	3
11	2	2020-07-20	3	4
12	3	2016-11-10	1	1
13	3	2016-12-15	2	2
14	3	2016-12-20	2	3
15	3	2017-12-07	2	4
16	3	2019-12-18	1	5

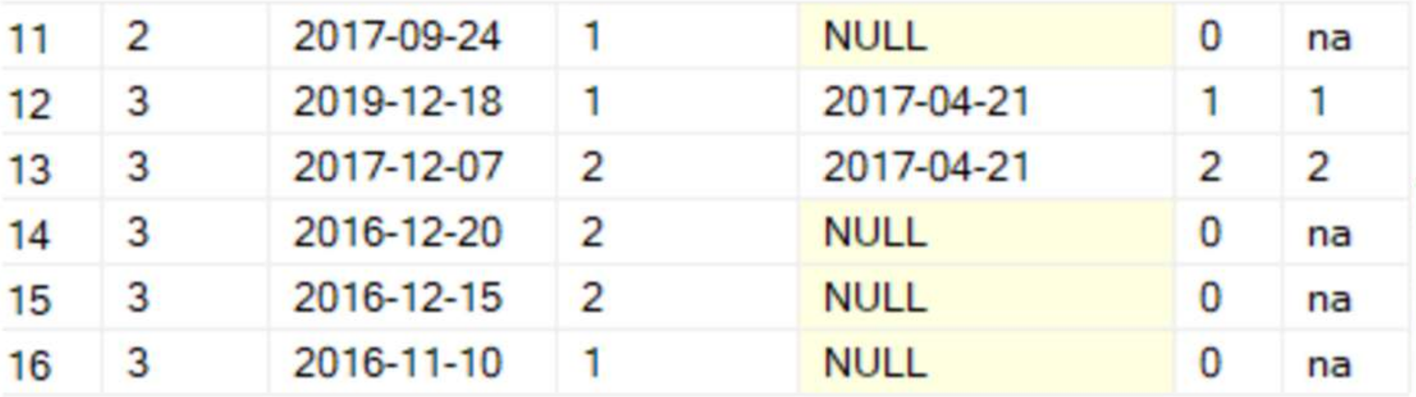


RANK ALL THE TRANSACTIONS FOR EACH MEMBER WHENEVER THEY ARE A ZOMATO GOLD MEMBER FOR EVERY NON GOLD MEMBER TRANSACTION MARK AS NA



```
select e.*, case when rnk=0 then 'na' else rnk end as rnkk from
(select c.*, cast((case when gold_signup_date is null then 0 else
rank()over(partition by userid order by created_date  desc) end) as varchar) as rnk from
(select a.userid, a.created_date, a.product_id, b.gold_signup_date from sales a left join
goldusers_signup b on a.userid=b.userid and created_date>=gold_signup_date)c)e;
```

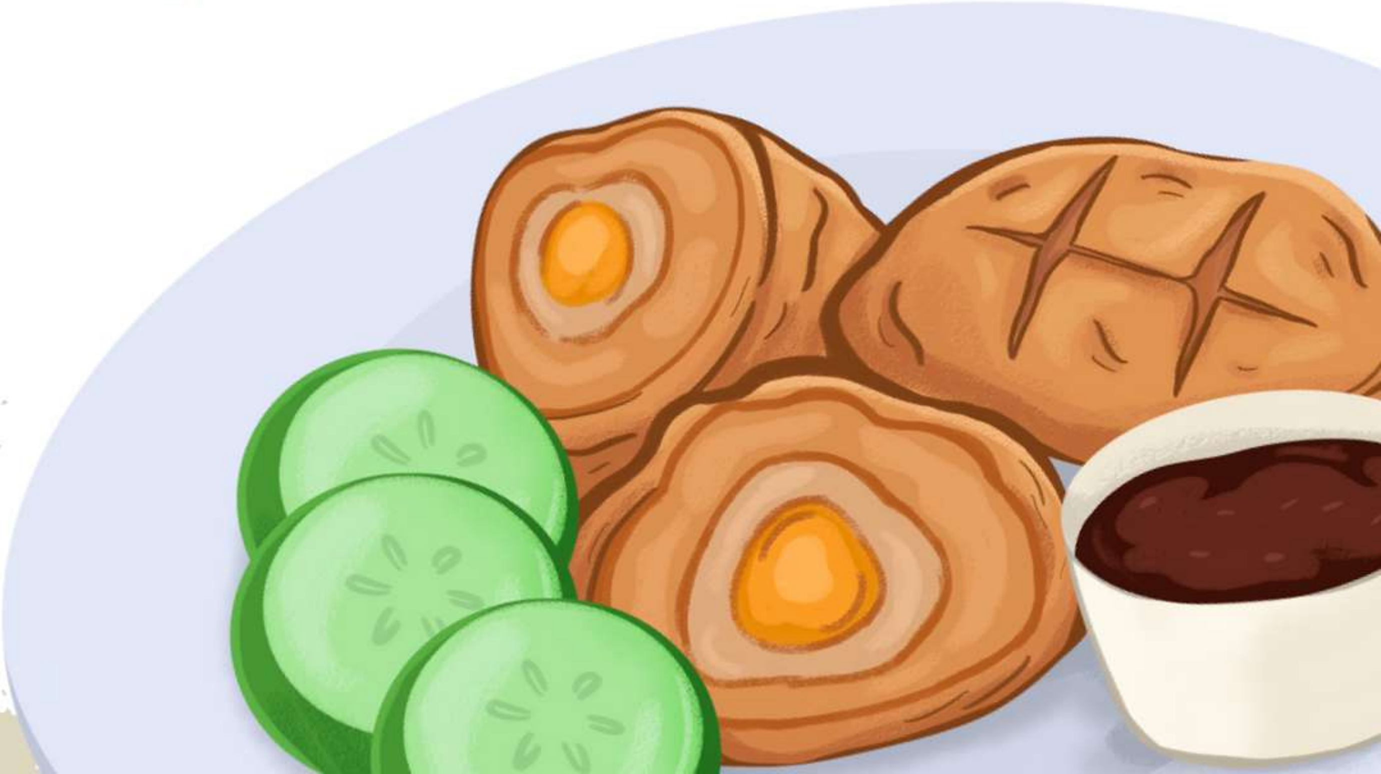




	userid	created_date	product_id	gold_signup_date	rnk	rnkk
1	1	2019-10-23	2	2017-09-22	1	1
2	1	2018-03-19	3	2017-09-22	2	2
3	1	2017-04-19	2	NULL	0	na
4	1	2017-03-11	2	NULL	0	na
5	1	2016-11-09	1	NULL	0	na
6	1	2016-05-20	3	NULL	0	na
7	1	2016-03-11	1	NULL	0	na
8	2	2020-07-20	3	NULL	0	na
9	2	2018-09-10	3	NULL	0	na
10	2	2017-11-08	2	NULL	0	na
11	2	2017-09-24	1	NULL	0	na
12	3	2019-12-18	1	2017-04-21	1	1
13	3	2017-12-07	2	2017-04-21	2	2
14	3	2016-12-20	2	NULL	0	na
15	3	2016-12-15	2	NULL	0	na
16	3	2016-11-10	1	NULL	0	na

CONCLUSION

The designed database effectively models core operations of a food delivery app, enabling efficient tracking of user behavior, premium subscriptions, and sales—laying the foundation for data-driven decision-making.



THANK YOU !

THANK YOU