SQL Project

DATA ANALYZATION OF ONLINE FOOD ORDERING AND DELIVERING PLATFORM "ZOMATO"

```
mysql> Select * from goldusers_signup;
+--------+------------------+
| userid | gold_signup_date |
+--------+------------------+
|      1 | 2017-09-22       |
|      3 | 2017-04-21       |
+--------+------------------+
2 rows in set (0.00 sec)

mysql> Select * from users;
+--------+-------------+
| userid | signup_date |
+--------+-------------+
|      1 | 2014-02-09  |
|      2 | 2015-01-15  |
|      3 | 2014-04-11  |
+--------+-------------+
3 rows in set (0.00 sec)

mysql> Select * from product;
+------------+--------------+-------+
| product_id | product_name | price |
+------------+--------------+-------+
|          1 | p1           |   980 |
|          2 | p2           |   870 |
|          3 | p3           |   330 |
+------------+--------------+-------+
3 rows in set (0.00 sec)
```

```
mysql> Select * from sales;
+--------+--------------+------------+
| userid | created_date | product_id |
+--------+--------------+------------+
|      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 |
+--------+--------------+------------+
16 rows in set (0.00 sec)
```

Q1. What is the total amount, each customer spent on the platform?

A1. Select userid, sum(price) as Amount from sales s inner join product p ON s.product_id= p.Product_id group by userid;

```
+--------+--------+
| userid | Amount |
+--------+--------+
|      1 |   5230 |
|      3 |   4570 |
|      2 |   2510 |
+--------+--------+
3 rows in set (0.00 sec)
```

Q2. For how many days, each customer visited the platform?

A2. Select userid, count(created_date) from sales group by userid;

Select userid, count(distinct created_date) from sales group by userid; (if dates are not distinct)

userid	count(created_date)
1	7
3	5
2	4

3 rows in set (0.00 sec)

Q3. What was the first product purchased by each customer (indicate by product_id)?
A3. Select userid, product_id, dense_rank() over (partition by userid order by created_date) as rnk from sales;

userid	product_id	rank
1	1	1
1	3	2
1	1	3
1	2	4
1	2	5
1	3	6
1	2	7
2	1	1
2	2	2
2	3	3
2	3	4
3	1	1
3	2	2
3	2	3
3	2	4
3	1	5

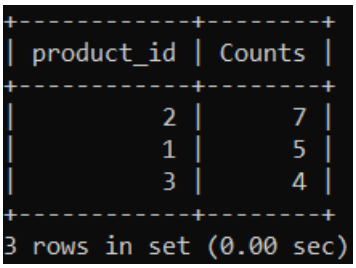
16 rows in set (0.00 sec)

Select b.* from (Select userid, product_id, dense_rank() over (partition by userid order by created_date) as rnk from sales) b where rnk=1;

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

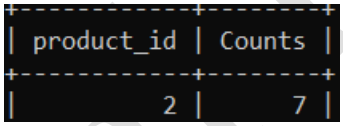
3 rows in set (0.00 sec)

Q4. What is the most purchased item and how many times it's been purchased on the platform?
A4. Select product_id, count(created_date) as Counts from sales group by product_id order by count(product_id) desc;



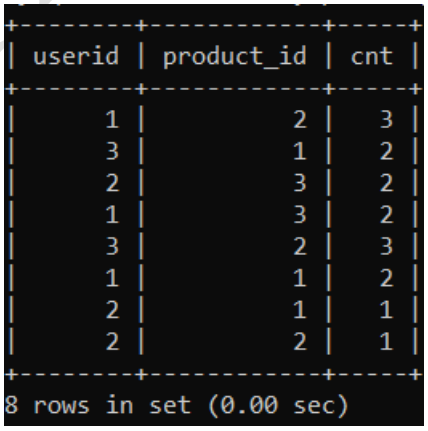
```
+------------+--------+
| product_id | Counts |
+------------+--------+
|          2 |      7 |
|          1 |      5 |
|          3 |      4 |
+------------+--------+
3 rows in set (0.00 sec)
```

Select b.* from (Select product_id, count(created_date) as Counts from sales group by product_id order by count(product_id) desc) b where Counts= max(Counts);



```
+------------+--------+
| product_id | Counts |
+------------+--------+
|          2 |      7 |
+------------+--------+
```

Q5. What is the most purchased item for each customer?
A5. Select userid, product_id, count(product_id) as cnt from sales group by userid, product_id;



```
+--------+------------+-----+
| userid | product_id | cnt |
+--------+------------+-----+
|      1 |          2 |   3 |
|      3 |          1 |   2 |
|      2 |          3 |   2 |
|      1 |          3 |   2 |
|      3 |          2 |   3 |
|      1 |          1 |   2 |
|      2 |          1 |   1 |
|      2 |          2 |   1 |
+--------+------------+-----+
8 rows in set (0.00 sec)
```

Select c.*, dense_rank() over (partition by userid order by counts desc) as rnk from (Select userid, product_id, count(product_id) as counts from sales group by userid, product_id) c;

userid	product_id	counts	rnk
1	2	3	1
1	3	2	2
1	1	2	2
2	3	2	1
2	1	1	2
2	2	1	2
3	2	3	1
3	1	2	2

8 rows in set (0.00 sec)

Select d.* from (Select c.*, dense_rank() over (partition by userid order by counts desc) as rnk from (Select userid, product_id, count(product_id) as counts from sales group by userid, product_id) c) d where rnk=1;

userid	product_id	counts	rnk
1	2	3	1
2	3	2	1
3	2	3	1

3 rows in set (0.00 sec)

Note: By analysing from above query, the platform can throw the offers/discounts to each user on their most purchased items, just to stick that customer to the platform and maintain the sales.

Q6. What is the first item purchased by the customer right after they became a gold user?
A6. Select c.* from (Select a.userid, a.gold_signup_date, b.product_id, b.created_date, dense_rank() over (partition by a.userid order by created_date) as rnk from goldusers_signup a inner join sales b ON a.userid=b.userid and created_date>=gold_signup_date) c where rnk=1;

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

2 rows in set (0.00 sec)

Q7. What is the first item purchased by the customer right before they became a gold user?
A7. Select c.* from (Select a.userid, a.gold_signup_date, b.product_id, b.created_date, dense_rank() over (partition by a.userid order by created_date desc) as 'rnk' from goldusers_signup a inner join sales b ON a.userid=b.userid and created_date<= a.gold_signup_date) c where rnk=1;

```
+--------+------------------+------------+--------------+-----+
| userid | gold_signup_date | product_id | created_date | rnk |
+--------+------------------+------------+--------------+-----+
|      1 | 2017-09-22       |          2 | 2017-04-19   |   1 |
|      3 | 2017-04-21       |          2 | 2016-12-20   |   1 |
+--------+------------------+------------+--------------+-----+
2 rows in set (0.04 sec)
```

Q8. What are the total orders and amount spent by the customer right before they became a gold user?
A8. Select userid, count(created_date) as order_purchased, sum(price) as total_amt_spent from (Select c.*, price from (Select a.userid, a.gold_signup_date, b.product_id, b.created_date from goldusers_signup a inner join sales b ON a.userid=b.userid and created_date<= gold_signup_date) c inner join product p ON c.product_id=p.product_id) d group by userid;

```
+--------+-----------------+-----------------+
| userid | order_purchased | total_amt_spent |
+--------+-----------------+-----------------+
|      3 |               3 |            2720 |
|      1 |               5 |            4030 |
+--------+-----------------+-----------------+
2 rows in set (0.00 sec)
```

Q9. In the first year when the customer joins the super membership, they earn 5 points for every 10 rupees spent (irrespective of whatever they purchase). So, what is their points earning in 1st year?
A10. Select a.userid, a.gold_signup_date, b.product_id, b.created_date from goldusers_signup a inner join sales b ON a.userid=b.userid and created_date>=gold_signup_date and created_date<=dateadd(year, 1, gold_signup_date)

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

Select c.*, d.price*(0.5) as total_points_earned from (Select a.userid, a.gold_signup_date, b.product_id, b.created_date from goldusers_signup a inner join sales 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;

(Since 0.5 zomato points = 1 rupees)

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

Q10. Rank all the transaction of the customers after they become gold member otherwise put their rank as 0?

A11. Select s.userid, s.created_date, g.Gold_signup_date from sales s Left join goldusers_signup g ON s.userid=g.userid and created_date>=gold_signup_date;

userid	created_date	Gold_signup_date
1	2017-04-19	NULL
3	2019-12-18	2017-04-21
2	2020-07-20	NULL
1	2019-10-23	2017-09-22
1	2018-03-19	2017-09-22
3	2016-12-20	NULL
1	2016-11-09	NULL
1	2016-05-20	NULL
2	2017-09-24	NULL
1	2017-03-11	NULL
1	2016-03-11	NULL
3	2016-11-10	NULL
3	2017-12-07	2017-04-21
3	2016-12-15	NULL
2	2017-11-08	NULL
2	2018-09-10	NULL

Select c.*, case when gold_signup_date is null then 0 else dense_rank() over (partition by userid order by created_date desc) end as rnk from (Select s.userid, s.created_date, g.Gold_signup_date from sales s Left join goldusers_signup g ON s.userid=g.userid and created_date>=gold_signup_date) c;

userid	created_date	Gold_signup_date	rnk
1	2019-10-23	2017-09-22	1
1	2018-03-19	2017-09-22	2
1	2017-04-19	NULL	0
1	2017-03-11	NULL	0
1	2016-11-09	NULL	0
1	2016-05-20	NULL	0
1	2016-03-11	NULL	0
2	2020-07-20	NULL	0
2	2018-09-10	NULL	0
2	2017-11-08	NULL	0
2	2017-09-24	NULL	0
3	2019-12-18	2017-04-21	1
3	2017-12-07	2017-04-21	2
3	2016-12-20	NULL	0
3	2016-12-15	NULL	0
3	2016-11-10	NULL	0