**#Q. What is the total amount each customer has paid for gold membership**

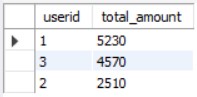
SELECT s.userid, SUM(p.price) AS total\_amount

FROM sales s

JOIN product p

ON s.product\_id=p.product\_id

GROUP BY s.userid;



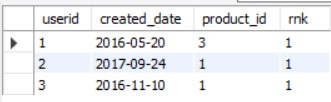
**#Q. what is the first product bought by each customer**

SELECT \* FROM

(SELECT \*, rank() OVER (partition by userid order by created\_date) AS rnk

FROM sales) a

WHERE rnk=1;



**# Q. how many times the most popular product was bought by each customers?**

SELECT userid, count(product\_id) AS no\_of\_times

FROM sales

WHERE product\_id=

(SELECT s.product\_id

FROM sales s

JOIN product p

ON s.product\_id=p.product\_id

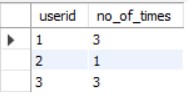
GROUP BY s.product\_id

ORDER BY COUNT(s.product\_id) desc

LIMIT 1)

group by userid

order by userid;



**# Q which items are users buying first after becoming gold member ?**

SELECT userid, created\_date, product\_id, rnk

FROM

(

SELECT a.userid, a.created\_date , a.product\_id , RANK() OVER (PARTITION BY a.userid order by a.created\_date ) AS rnk

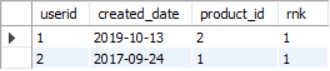
FROM sales a

JOIN goldusers\_signup b

ON a.userid=b.userid

WHERE a.created\_date>=b.gold\_signup\_date ) AS d

WHERE rnk=1;



**#Q . Which item was purchases just before the customer became the member ?**

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

FROM sales a

JOIN goldusers\_signup b

ON a.userid=b.userid

WHERE a.created\_date<=b.gold\_signup\_date

ORDER BY created\_date DESC

LIMIT 1;



**# Q . What is the total order and amount spent for each member before they became a member ?**

SELECT a.userid,created\_date,c.product\_id,price,COUNT(c.product\_id) AS total\_order, SUM(price) AS total\_price

FROM sales a

JOIN goldusers\_signup b

ON a.userid=b.userid

JOIN product c

ON a.product\_id=c.product\_id

WHERE created\_date<=gold\_signup\_date

GROUP BY userid;



**#Q . Rank all the transaction of customers**

SELECT c.\* , rank() over (partition by total\_price order by user ) AS rnk

FROM

(SELECT a.userid , SUM(price) AS total\_price

FROM sales a

JOIN product b

ON a.product\_id=b.product\_id

GROUP BY userid) AS c;

SELECT \*, rank() OVER ( partition by userid order by created\_date) AS rnk

FROM sales;



**# Q . rank all the transactions of a gold member and for every non gold member mark it as N.A**.

SELECT c.\* ,

CASE

WHEN gold\_signup\_date is null then 0 else rank() OVER (partition by userid ORDER BY created\_date DESC)end 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) AS c;

