-- Q1

SELECT customer\_id, SUM(price) AS 'Total\_Amount\_Spent'

FROM sales JOIN menu ON sales.product\_id = menu.product\_id

GROUP BY customer\_id;

-- Output:



-- Q2

SELECT customer\_id, COUNT(order\_date) AS 'Total\_Days\_Visited'

FROM sales GROUP BY customer\_id;

-- Output:



-- Q3

SELECT customer\_id, product\_name

FROM sales JOIN menu ON sales.product\_id = menu.product\_id

WHERE order\_date IN

(SELECT MIN(order\_date) FROM sales GROUP BY customer\_id);

-- Output:



-- Q4

SELECT product\_name, COUNT(\*) AS 'Total\_purchases'

FROM sales JOIN menu ON sales.product\_id = menu.product\_id

GROUP BY sales.product\_id ORDER BY Total\_Purchases DESC LIMIT 1;

-- Output:



-- Q5

SELECT customer\_id, product\_name, item\_bought\_count FROM

(SELECT customer\_id, product\_name, count(sales.product\_id) AS 'item\_bought\_count',

dense\_rank() over(partition by customer\_id order by count(sales.product\_id) desc)

as cte from sales join menu on sales.product\_id = menu.product\_id

group by customer\_id, sales.product\_id ) as subquery where cte=1;

-- Output:



-- Q6

select customer\_id, product\_name from (select row\_number()

over(partition by sales.customer\_id order by order\_date) as cte,

sales.customer\_id, order\_date, join\_date, product\_name from sales join menu

on sales.product\_id = menu.product\_id join members on members.customer\_id = sales.customer\_id

where order\_date >= join\_date)as subquery where cte =1 ;

-- Output:



-- Q7

select customer\_id, product\_name from( select row\_number()

over(partition by sales.customer\_id order by order\_date) as cte,

sales.customer\_id, order\_date, join\_date, product\_name from sales

join menu on sales.product\_id = menu.product\_id join members on members.customer\_id = sales.customer\_id

where order\_date <= join\_date) as subquery where cte = 1 ;

-- Output:



-- Q8

SELECT members.customer\_id, COUNT(DISTINCT menu.product\_id) AS 'Items\_ordered',

SUM(price) AS 'Money\_spent' FROM members JOIN sales

ON members.customer\_id = sales.customer\_id JOIN menu

ON menu.product\_id = sales.product\_id WHERE order\_date < join\_date GROUP BY customer\_id;

-- Output:



-- Q9

SELECT customer\_id,SUM(CASE WHEN product\_name = 'sushi' THEN 20 \* price ELSE 10 \* price

END) AS money\_spent FROM sales JOIN menu ON sales.product\_id = menu.product\_id

GROUP BY customer\_id ORDER BY customer\_id;

-- Output:



-- Q10

WITH dates AS( SELECT \*, adddate(join\_date, interval 7 day) as valid\_date, '2021-01-31'

AS last\_date FROM members) Select S.Customer\_id, SUM( Case When m.product\_ID = 1 THEN m.price\*20

When S.order\_date between D.join\_date and D.valid\_date Then m.price\*20 Else m.price\*10 END )

AS Points From Dates D join Sales S On D.customer\_id = S.customer\_id Join Menu M

On M.product\_id = S.product\_id Where S.order\_date < d.last\_date Group by S.customer\_id;

-- Output:



-- Bonus\_Table\_1

SELECT s.customer\_id, Order\_date, product\_name, price, CASE

WHEN order\_Date < join\_date THEN 'N' ELSE 'Y'

END AS member FROM sales s JOIN menu m ON s.product\_id = m.product\_id

JOIN members ms ON ms.customer\_id = s.customer\_id ORDER BY s.customer\_id , order\_date;

-- Output:



-- Bonus\_Table\_2

with cte as (select s.customer\_id, order\_date, product\_name, price, case

when order\_Date < join\_date then 'N' else 'Y' end as member from sales s join menu m

on s.product\_id = m.product\_id join members ms on ms.customer\_id= s.customer\_id

order by s.customer\_id, order\_date) select \*, case when member = 'N' then null else

rank() over(partition by s.customer\_id, member order by order\_date) end as ranking from cte;

-- Output:

