--QUESTIONS

-- 1. What is the total amount each customer spent at the restaurant?

SELECT

s.customer\_id,

SUM(m.price) AS total\_spent

FROM sales s

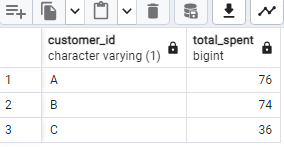
JOIN menu m

ON s.product\_id = m.product\_id

GROUP BY s.customer\_id

ORDER BY s.customer\_id;

**RESULT:**



-- 2. How many days has each customer visited the restaurant?

SELECT

customer\_id,

COUNT(Distinct order\_date) AS visit\_days

FROM sales

GROUP BY customer\_id;

**RESULT:**



-- 3. What was the first item from the menu purchased by each customer?

WITH orders\_rank As

(

SELECT customer\_id,

order\_date,

product\_id,

DENSE\_RANK()OVER(PARTITION BY customer\_id ORDER BY order\_date ASC ) AS orders\_rank

FROM sales

)

SELECT

o.customer\_id,

m.product\_name

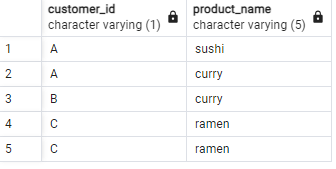
FROM orders\_rank o JOIN menu m

ON o.product\_id=m.product\_id

WHERE o.orders\_rank=1

ORDER BY o.customer\_id;

**RESULT:**



--CAN USE DENSE\_RANK/ROW\_NUMBER SINCE THERE IS NO TIMESTAMP IS GIVEN

-- 4. What is the most purchased item on the menu and how many times was it purchased by all customers?

SELECT m.product\_name,

COUNT(m.product\_name) AS most\_count

FROM sales s JOIN menu m

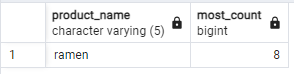
ON s.product\_id=m.product\_id

GROUP BY m.product\_name

ORDER BY most\_count DESC

LIMIT 1;

**RESULT:**



-- 5. Which item was the most popular for each customer?

WITH Fev\_item AS(

SELECT

s.customer\_id,

m.product\_name,

COUNT(m.product\_id) AS order\_count,

DENSE\_RANK() OVER(PARTITION BY s.customer\_id ORDER BY COUNT(s.customer\_id)DESC) AS rnk

FROM menu AS m

JOIN sales As s

ON m.product\_id=s.product\_id

GROUP BY 1,2

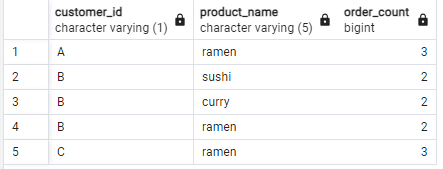
)

SELECT customer\_id,product\_name,order\_count

FROM fev\_item

WHERE rnk=1;

**RESULT:**



--Cust A likes ramen most,

B likes all the items,

C likes ramen most.

-- 6. Which item was purchased first by the customer after they became a member?

WITH member\_cte AS

(

SELECT

s.customer\_id,mb.join\_date,s.order\_date,s.product\_id,

DENSE\_RANK() OVER(PARTITION BY s.customer\_id

ORDER BY s.order\_date) AS rnk

FROM sales s

JOIN members mb

ON s.customer\_id=mb.customer\_id

WHERE s.order\_date>=mb.join\_date

)

SELECT c.customer\_id,c.order\_date,m.product\_name

FROM member\_cte c

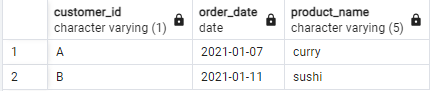
JOIN menu m

ON c.product\_id=m.product\_id

WHERE rnk=1

ORDER BY c.customer\_id;

**RESULT:**



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

WITH before\_member\_cte AS

(

SELECT

s.customer\_id,mb.join\_date,s.order\_date,s.product\_id,

DENSE\_RANK() OVER(PARTITION BY s.customer\_id

ORDER BY s.order\_date DESC) AS rnk

FROM sales s

JOIN members mb

ON s.customer\_id=mb.customer\_id

WHERE s.order\_date<mb.join\_date

)

SELECT bc.customer\_id,bc.order\_date,m.product\_name

FROM before\_member\_cte bc

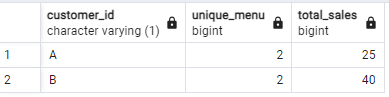
JOIN menu m

ON bc.product\_id=m.product\_id

WHERE rnk=1

ORDER BY bc.customer\_id;

**RESULT:**



-- 8. What is the total items and amount spent for each member before they became a member?

SELECT s.customer\_id,

COUNT(DISTINCT s.product\_id) AS unique\_menu,

SUM(m.price) AS total\_sales

FROM sales AS s

JOIN members AS mb

ON s.customer\_id = mb.customer\_id

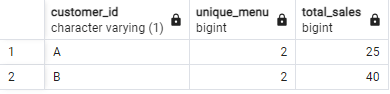
JOIN menu AS m

ON s.product\_id = m.product\_id

WHERE s.order\_date < mb.join\_date

GROUP BY s.customer\_id;

**RESULT:**



--Before becoming member

A ordered 2 items and spent total 25

And B ordered 2 items and spent 40.

-- 9. If each $1 spent equates to 10 points and sushi has a 2x points multiplier -

-- how many points would each customer have?

WITH cte

AS

(

SELECT

product\_id,product\_name,price,

CASE

WHEN product\_name='sushi' THEN price\*20

ELSE price\*10

END AS points

FROM menu

)

SELECT

s.customer\_id,

SUM(c.points) AS totalPoints

FROM cte c

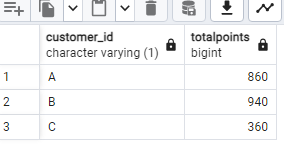
JOIN sales s

ON c.product\_id=s.product\_id

GROUP BY s.customer\_id

ORDER BY s.customer\_id;

**RESULT:**

****

-- 10. In the first week after a customer joins the program (including their join date)

-- they earn 2x points on all items, not just sushi - how many points do customer A and B have at the end of January?

WITH date\_cte

AS

(

SELECT

customer\_id,

join\_date,

join\_date + INTERVAL '6 DAYS' AS first\_week,

(DATE\_TRUNC('MONTH','2021-01-31'::DATE)+ INTERVAL '1 MONTH'-INTERVAL '1 DAY') AS last\_day

FROM members

)

SELECT s.customer\_id,

SUM(

CASE WHEN m.product\_name='sushi' THEN 2\*10\*m.price

WHEN s.order\_date BETWEEN dc.join\_date AND dc.first\_week THEN 2\*10\*m.price

ELSE 10\*m.price

END) AS points

FROM sales s

JOIN date\_cte dc

ON s.customer\_id=dc.customer\_id

JOIN menu m

ON s.product\_id=m.product\_id

WHERE

dc.join\_date<=s.order\_date

AND s.order\_date<=dc.last\_day

GROUP BY s.customer\_id

ORDER BY s.customer\_id;

**RESULT:**



--BOnus questions

--JOIN ALL THINGS

-- Recreate the table with: customer\_id, order\_date, product\_name, price, member (Y/N)

SELECT

s.customer\_id,

s.order\_date,

m.product\_name,

m.price,

CASE

WHEN mb.join\_date>s.order\_date THEN 'N'

WHEN mb.join\_date<=s.order\_date THEN 'Y'

ELSE 'N'

END

AS member

FROM sales s

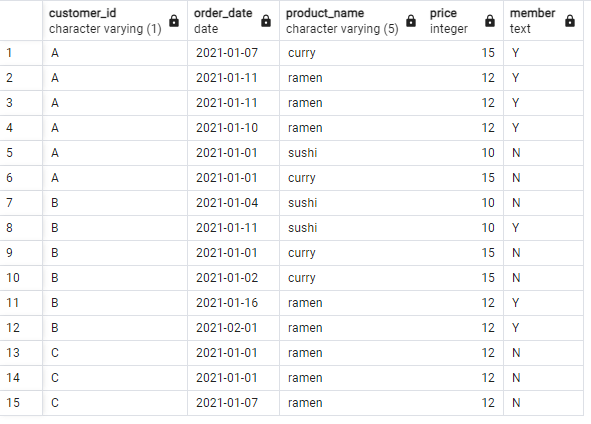
LEFT JOIN menu m

ON s.product\_id=m.product\_id

LEFT JOIN members mb

ON s.customer\_id=mb.customer\_id;

**RESULT:**



--Rank All The Things

WITH cust\_cte AS

(

SELECT

s.customer\_id,

s.order\_date,

m.product\_name,

m.price,

CASE

WHEN mb.join\_date>s.order\_date THEN 'N'

WHEN mb.join\_date<=s.order\_date THEN 'Y'

ELSE 'N'

END

AS member

FROM sales s

LEFT JOIN menu m

ON s.product\_id=m.product\_id

LEFT JOIN members mb

ON s.customer\_id=mb.customer\_id

)

SELECT customer\_id,order\_date,product\_name,price,member,

CASE

WHEN member='N' THEN NULL

ELSE RANK() OVER(PARTITION BY customer\_id,member ORDER BY order\_date)

END AS ranking

FROM cust\_cte;

**RESULT:**

