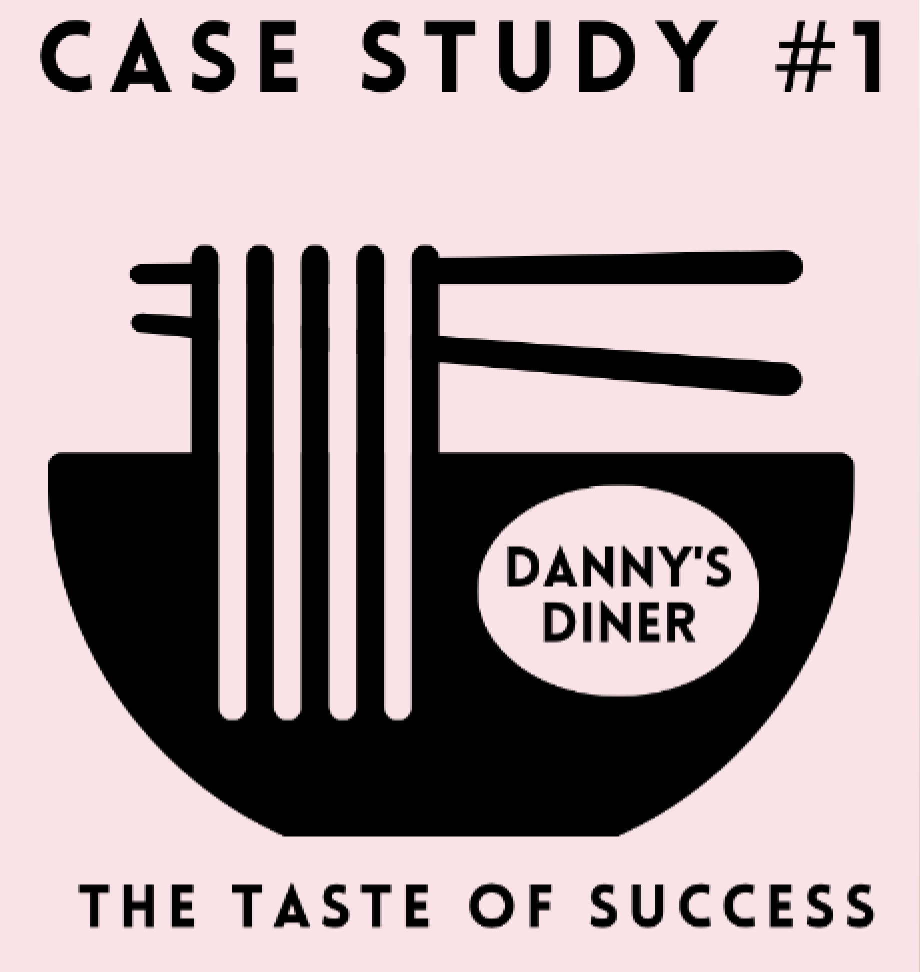
DANNY'S DINER

SQL CASE STUDY

Introduction

Danny seriously loves Japanese food so in the beginning of 2021, he decides to embark upon a risky venture and opens up a cute little restaurant that sells his 3 favourite foods: sushi, curry and ramen.

Danny’s Diner is in need of your assistance to help the restaurant stay afloat - the restaurant has captured some very basic data from their few months of operation but have no idea how to use their data to help them run the business.



By:
Navjoth Singh
M.B.A

singhnavjoth1138@gmail.com
+91-63951-80098

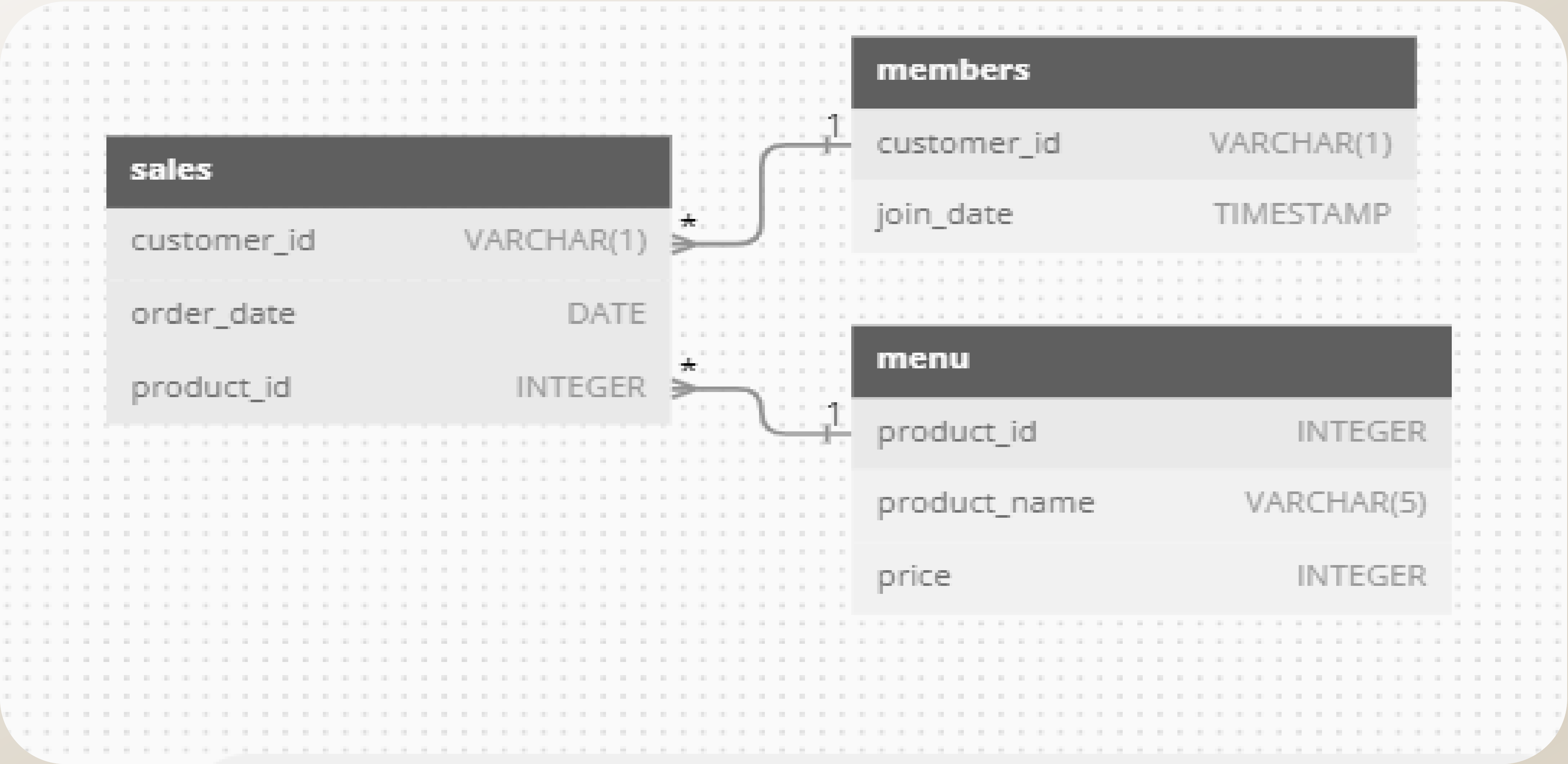
PROBLEM STATEMENT

- Danny wants to use the data to answer a few simple questions about his customers, especially about their visiting patterns, how much money they've spent and also which menu items are their favourite. Having this deeper connection with his customers will help him deliver a better and more personalised experience for his loyal customers.
- He plans on using these insights to help him decide whether he should expand the existing customer loyalty program - additionally he needs help to generate some basic datasets so his team can easily inspect the data without needing to use SQL.
- Danny has provided you with a sample of his overall customer data due to privacy issues - but he hopes that these examples are enough for you to write fully functioning SQL queries to help him answer his questions!

Danny has shared with you 3 key datasets for this case study:

- sales
- menu
- members

SCHEMA DIAGRAM



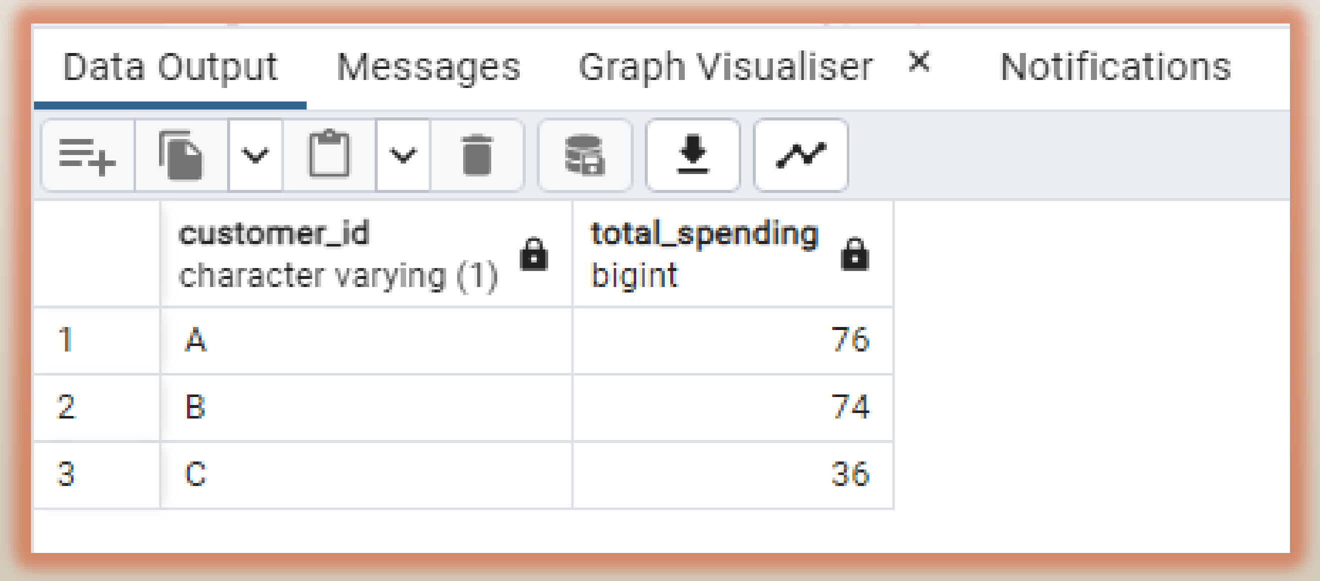
sales	
customer_id	VARCHAR(1)
order_date	DATE
product_id	INTEGER

members	
customer_id	VARCHAR(1)
join_date	TIMESTAMP

menu	
product_id	INTEGER
product_name	VARCHAR(5)
price	INTEGER

Q1. WHAT IS THE TOTAL AMOUNT EACH CUSTOMER SPENT AT THE RESTAURANT?

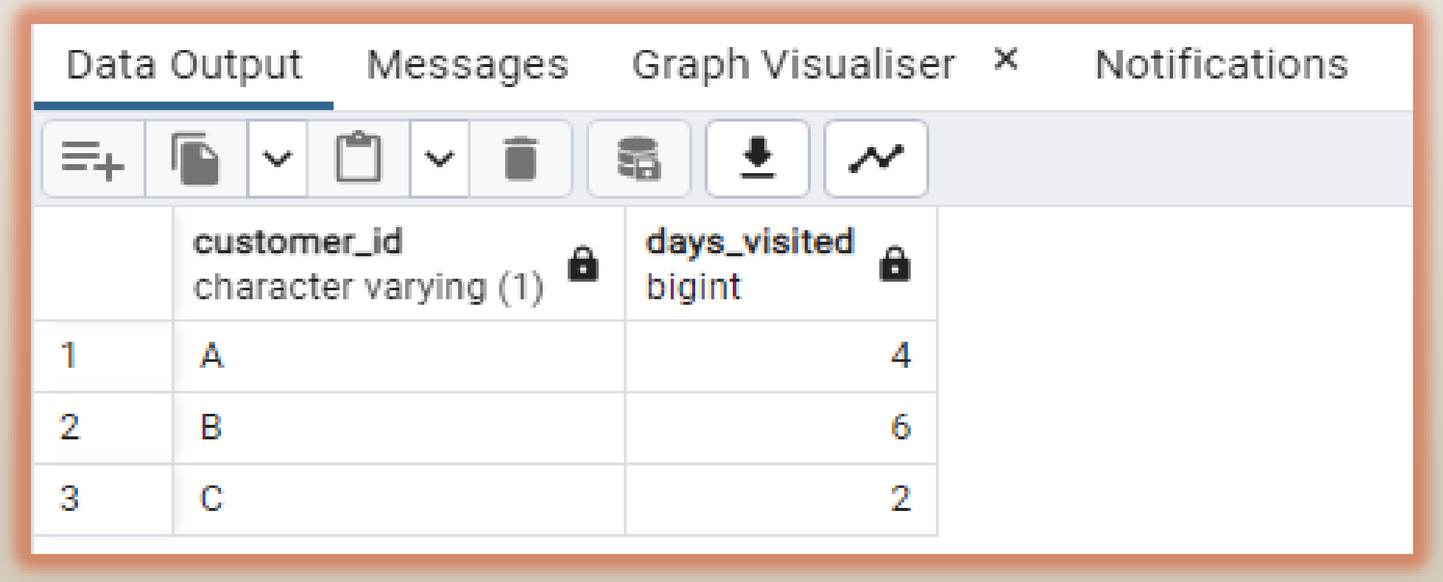
```
select s.customer_id,sum(mu.price) as total_spending from sales as s
join menu as mu on mu.product_id = s.product_id
group by 1
order by 1
```



	customer_id character varying (1)	total_spending bigint
1	A	76
2	B	74
3	C	36

Q2. HOW MANY DAYS HAS EACH CUSTOMER VISITED THE RESTAURANT?

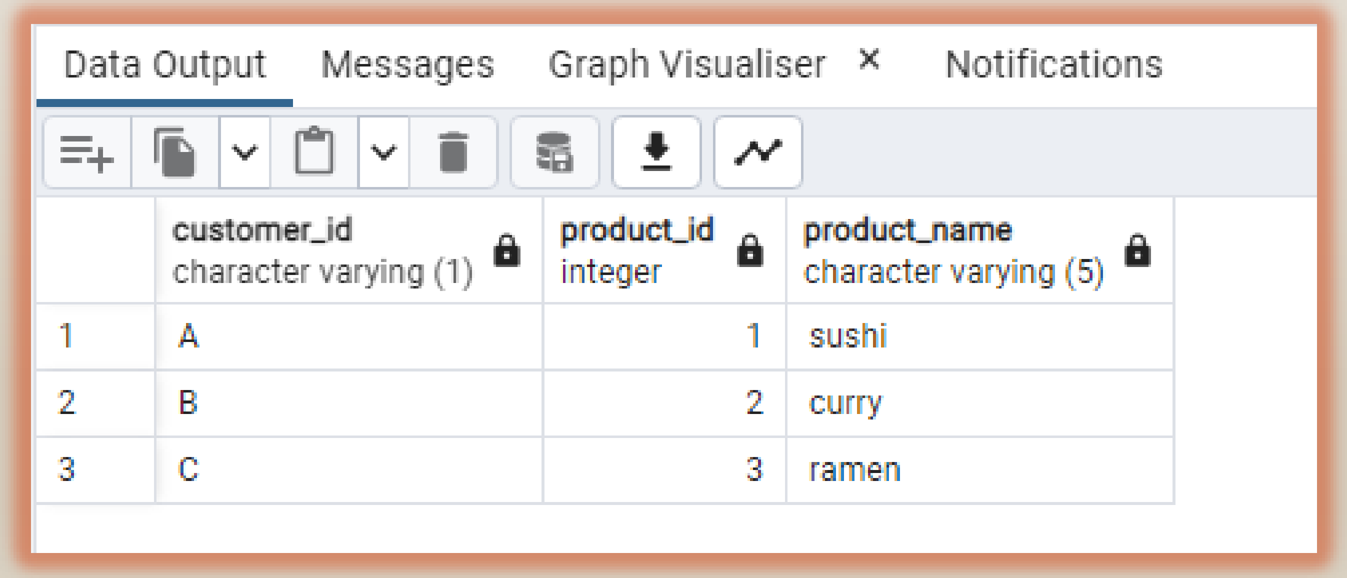
```
select customer_id, count(distinct(order_date)) as days_visited from sales
group by 1
order by 1
```



	customer_id character varying (1)	days_visited bigint
1	A	4
2	B	6
3	C	2

Q3. WHAT WAS THE FIRST ITEM FROM THE MENU PURCHASED BY EACH CUSTOMER?

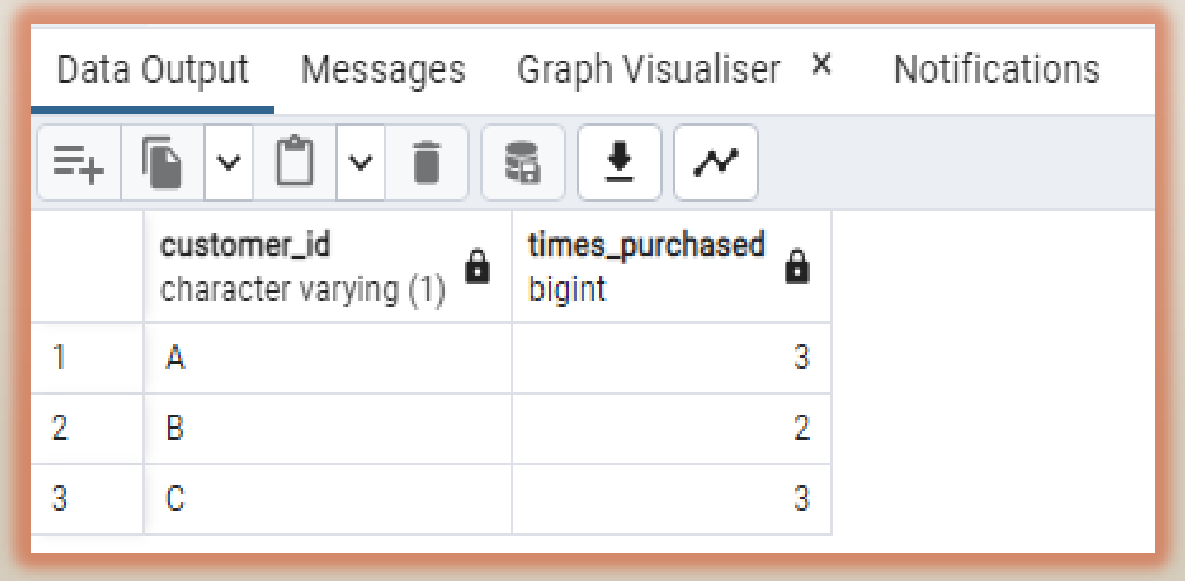
```
with cte as(
SELECT customer_id, product_id, ROW_NUMBER() OVER (PARTITION BY customer_id)
AS dishes_ordered
FROM sales)
select cte.customer_id, cte.product_id,mu.product_name from cte
join menu as mu on mu.product_id=cte.product_id
where cte.dishes_ordered=1
```



	customer_id character varying (1)	product_id integer	product_name character varying (5)
1	A	1	sushi
2	B	2	curry
3	C	3	ramen

Q4. WHAT IS THE MOST PURCHASED ITEM ON THE MENU AND HOW MANY TIMES WAS IT PURCHASED BY ALL CUSTOMERS?

```
with cte as(
select product_id,count(product_id) as times_purchased from sales
group by 1
limit 1)
select s.customer_id, count(cte.times_purchased) as times_purchased from sales as s
join cte on cte.product_id=s.product_id
group by 1
```

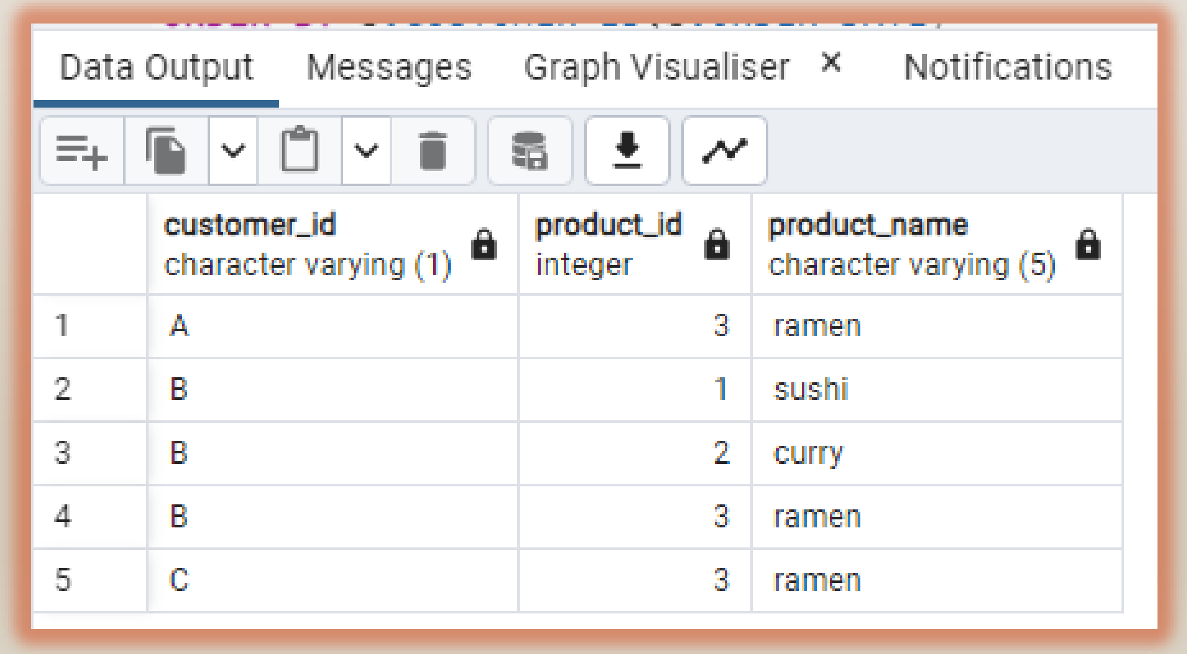


	customer_id character varying (1)	times_purchased bigint
1	A	3
2	B	2
3	C	3

Q5. WHICH ITEM WAS THE MOST POPULAR FOR EACH CUSTOMER?

```
WITH cte AS (SELECT customer_id, product_id ,
COUNT(product_id) OVER (PARTITION BY customer_id, product_id) as count FROM sales)

SELECT cte.customer_id, cte.product_id, mu.product_name
FROM cte
JOIN menu AS mu ON mu.product_id = cte.product_id
WHERE cte.count = (SELECT MAX(count) FROM cte AS cte2 WHERE cte2.customer_id =
cte.customer_id)
GROUP by 1,2,3
ORDER BY cte.customer_id
```

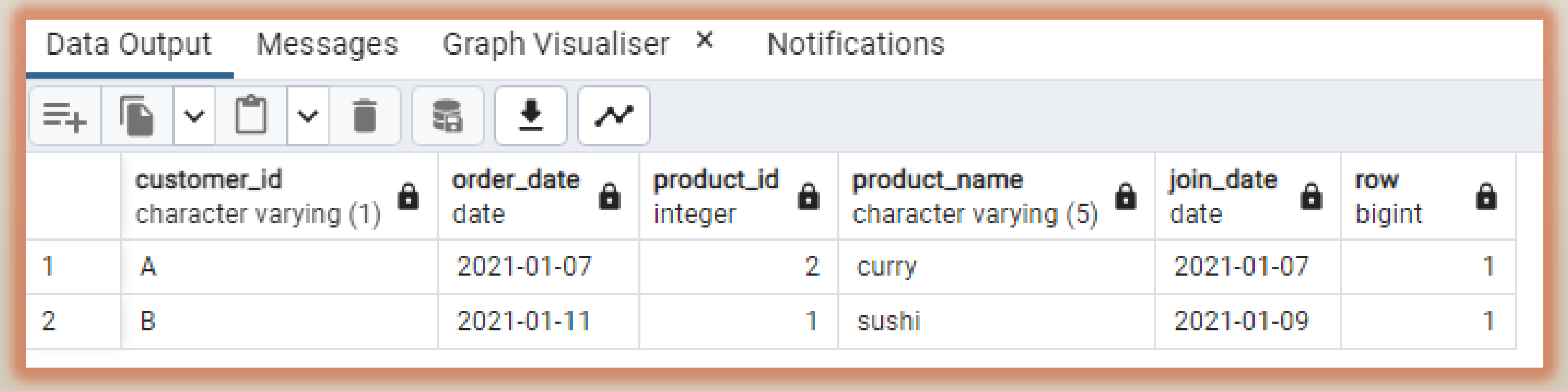


	customer_id character varying (1)	product_id integer	product_name character varying (5)
1	A	3	ramen
2	B	1	sushi
3	B	2	curry
4	B	3	ramen
5	C	3	ramen

Q6. WHICH ITEM WAS PURCHASED FIRST BY THE CUSTOMER AFTER THEY BECAME A MEMBER?

```
WITH CTE AS
(SELECT S.*,MU.PRODUCT_NAME,ME.JOIN_DATE, ROW_NUMBER() OVER(PARTITION BY
S.CUSTOMER_ID) AS ROW FROM MENU AS MU
JOIN SALES AS S ON S.PRODUCT_ID=MU.PRODUCT_ID
JOIN MEMBERS AS ME ON ME.CUSTOMER_ID=S.CUSTOMER_ID
WHERE ME.JOIN_DATE<=S.ORDER_DATE
ORDER BY S.CUSTOMER_ID,S.ORDER_DATE)

SELECT * FROM CTE
WHERE ROW=1
```

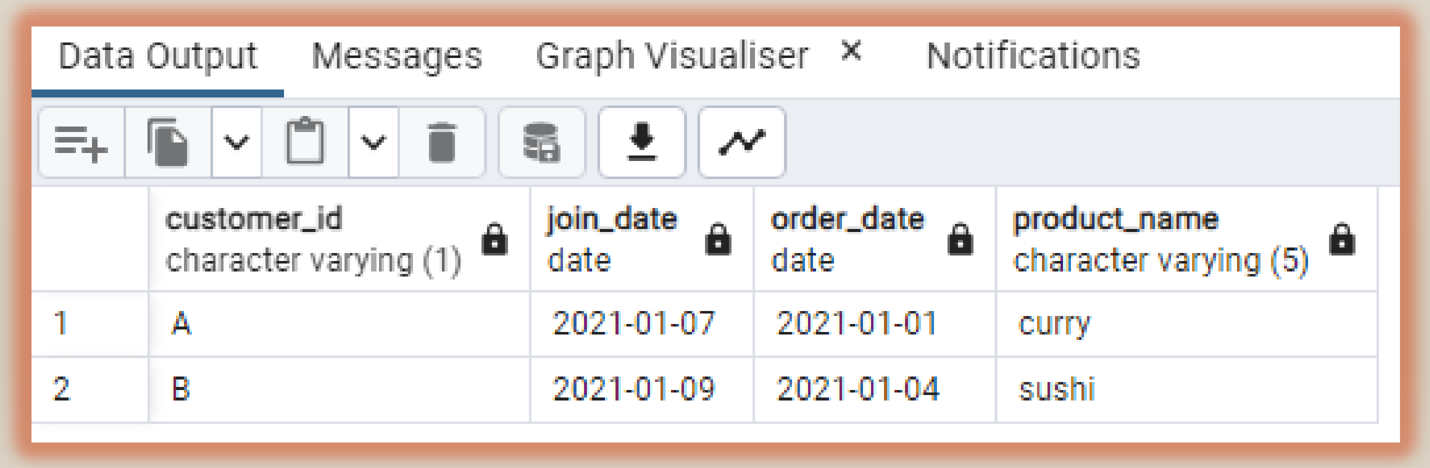


	customer_id character varying (1)	order_date date	product_id integer	product_name character varying (5)	join_date date	row bigint
1	A	2021-01-07	2	curry	2021-01-07	1
2	B	2021-01-11	1	sushi	2021-01-09	1

Q7. WHICH ITEM WAS PURCHASED JUST BEFORE THE CUSTOMER BECAME A MEMBER?

```
WITH CTE AS
(SELECT S.*,MU.PRODUCT_NAME,ME.JOIN_DATE, ROW_NUMBER() OVER(PARTITION BY
S.CUSTOMER_ID ORDER BY S.ORDER_DATE) AS ROW FROM MENU AS MU
JOIN SALES AS S ON S.PRODUCT_ID=MU.PRODUCT_ID
JOIN MEMBERS AS ME ON ME.CUSTOMER_ID=S.CUSTOMER_ID
WHERE ME.JOIN_DATE>S.ORDER_DATE
ORDER BY S.CUSTOMER_ID,S.ORDER_DATE)

SELECT CUSTOMER_ID,JOIN_DATE,ORDER_DATE,PRODUCT_NAME FROM CTE
WHERE CTE.ROW= (SELECT MAX(ROW) FROM CTE AS CTE2 WHERE CTE2.CUSTOMER_ID =
CTE.CUSTOMER_ID)
```

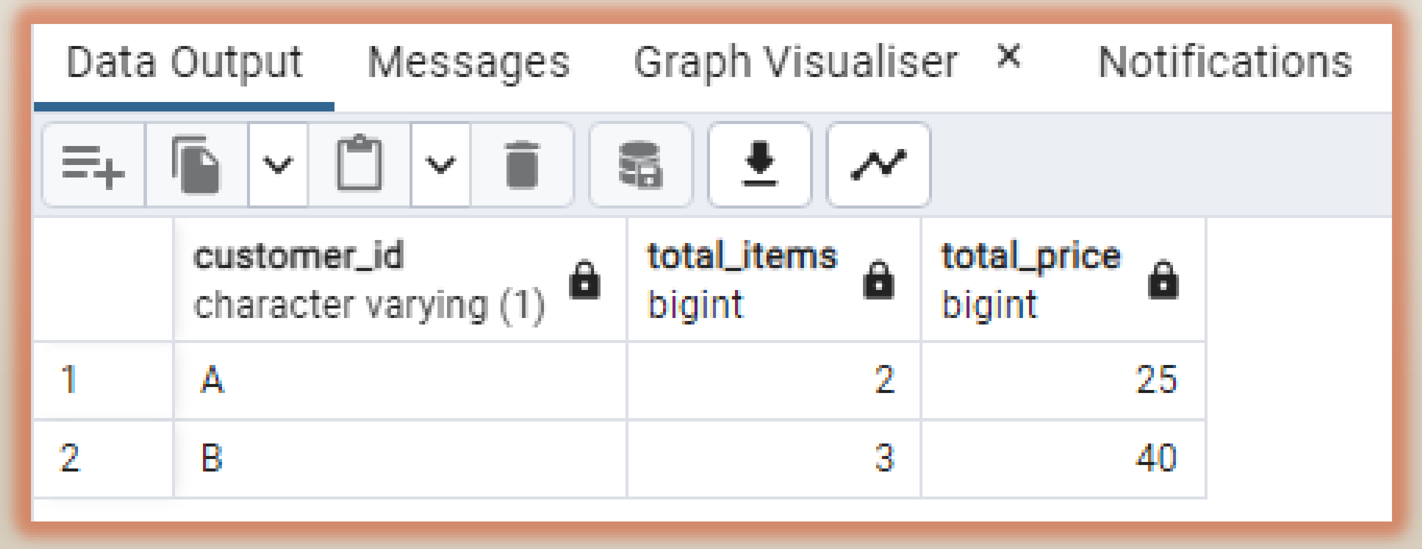


	customer_id character varying (1)	join_date date	order_date date	product_name character varying (5)
1	A	2021-01-07	2021-01-01	curry
2	B	2021-01-09	2021-01-04	sushi

Q8. WHAT IS THE TOTAL ITEMS AND AMOUNT SPENT FOR EACH MEMBER BEFORE THEY BECAME A MEMBER?

```
WITH CTE AS(
SELECT S.*,MU.PRODUCT_NAME,MU.PRICE,ME.JOIN_DATE FROM MENU AS MU
JOIN SALES AS S ON S.PRODUCT_ID=MU.PRODUCT_ID
JOIN MEMBERS AS ME ON ME.CUSTOMER_ID=S.CUSTOMER_ID
WHERE ME.JOIN_DATE>S.ORDER_DATE
ORDER BY S.CUSTOMER_ID,S.ORDER_DATE)

SELECT CUSTOMER_ID,COUNT(PRODUCT_ID) as total_items,SUM(PRICE) as total_price FROM CTE
GROUP BY 1
ORDER BY 1
```

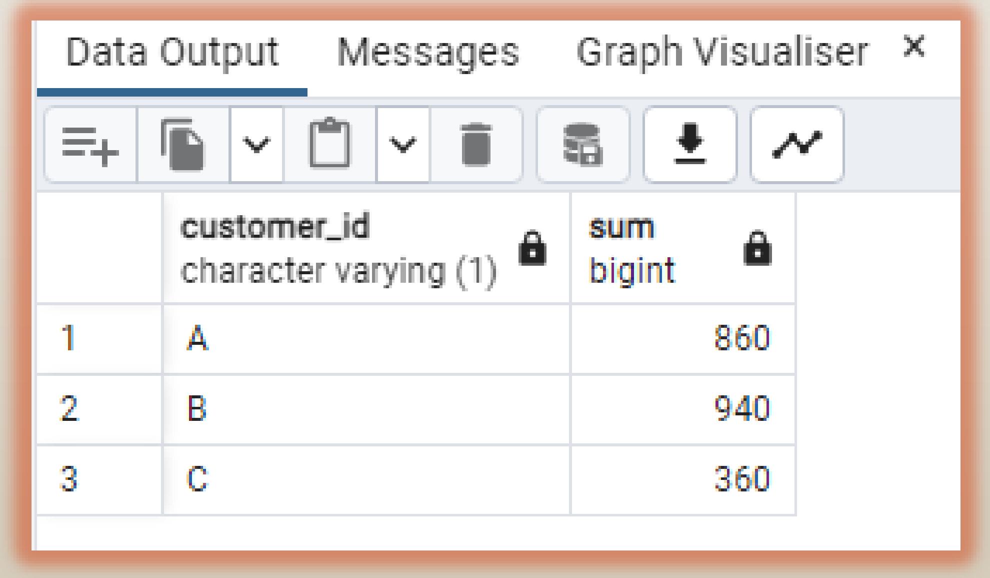


	customer_id character varying (1)	total_items bigint	total_price bigint
1	A	2	25
2	B	3	40

Q9. 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 s.customer_id,s.product_id,mu.product_name,mu.price,
(case mu.product_name when 'sushi' then mu.price*20
else mu.price*10 end) as points from sales as s
join menu as mu on mu.product_id=s.product_id
order by 1,2)

select customer_id,sum(points) from cte
group by 1
```

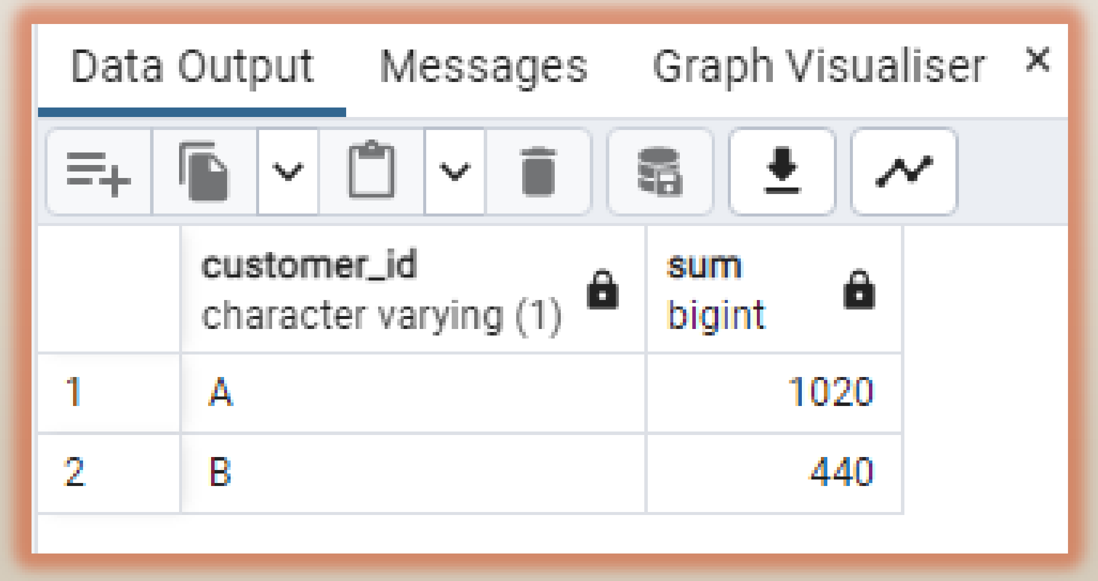


	customer_id character varying (1)	sum bigint
1	A	860
2	B	940
3	C	360

Q10. 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 CTE AS
(SELECT S.*,MU.PRODUCT_NAME,ME.JOIN_DATE, MU.PRICE*20 AS POINTS FROM MENU AS MU
JOIN SALES AS S ON S.PRODUCT_ID=MU.PRODUCT_ID
JOIN MEMBERS AS ME ON ME.CUSTOMER_ID=S.CUSTOMER_ID
WHERE ME.JOIN_DATE<=S.ORDER_DATE
ORDER BY S.CUSTOMER_ID,S.ORDER_DATE)

SELECT CUSTOMER_ID,SUM(POINTS) FROM CTE
WHERE EXTRACT(MONTH FROM ORDER_DATE) = 01
GROUP BY 1
```

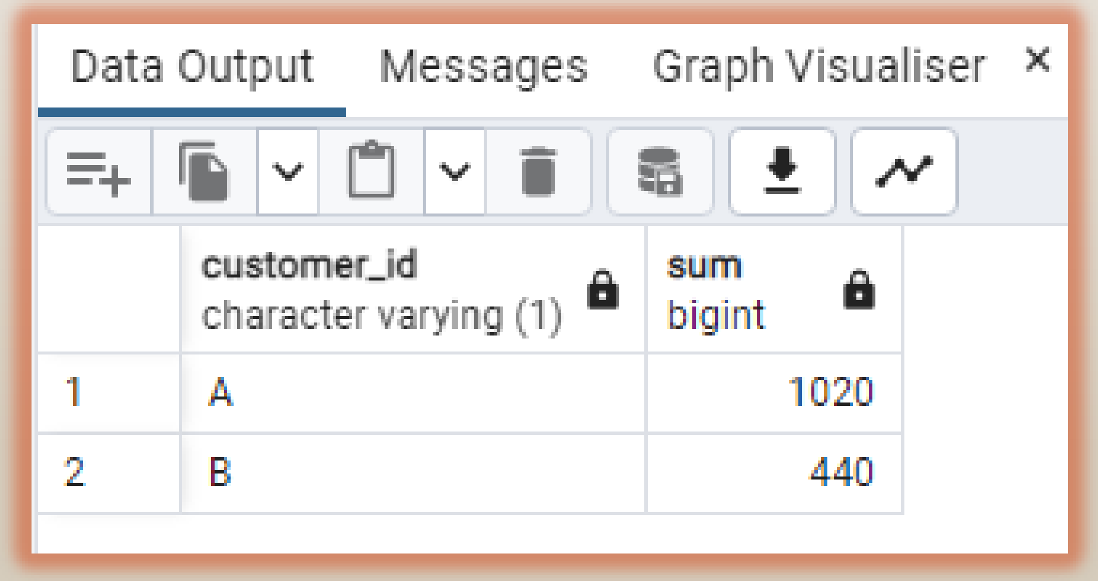


	customer_id character varying (1)	sum bigint
1	A	1020
2	B	440

Q10. 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 CTE AS
(SELECT S.*,MU.PRODUCT_NAME,ME.JOIN_DATE, MU.PRICE*20 AS POINTS FROM MENU AS MU
JOIN SALES AS S ON S.PRODUCT_ID=MU.PRODUCT_ID
JOIN MEMBERS AS ME ON ME.CUSTOMER_ID=S.CUSTOMER_ID
WHERE ME.JOIN_DATE<=S.ORDER_DATE
ORDER BY S.CUSTOMER_ID,S.ORDER_DATE)

SELECT CUSTOMER_ID,SUM(POINTS) FROM CTE
WHERE EXTRACT(MONTH FROM ORDER_DATE) = 01
GROUP BY 1
```



	customer_id character varying (1)	sum bigint
1	A	1020
2	B	440

THANK YOU