/\* --------------------

Case study dataset - [**https://8weeksqlchallenge.com/case-study-1/**](https://8weeksqlchallenge.com/case-study-1/)

Case Study Questions <https://www.db-fiddle.com/f/2rM8RAnq7h5LLDTzZiRWcd/138>

--------------------\*/

--------------------------------------------

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

/\*

SELECT

customer\_id,

sum(price) AS Total\_Amount

FROM dannys\_diner.sales s

JOIN dannys\_diner.menu m ON

s.product\_id = m.product\_id

GROUP BY customer\_id

\*/

----------------------------------------------

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

/\*

SELECT customer\_id,

count(order\_date) as No\_days\_visited\_rest

FROM dannys\_diner.sales s

GROUP BY customer\_id

\*/

---------------------------------------------------------

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

/\*

WITH cte3 as

(SELECT

customer\_id,

s.product\_id,

product\_name,

row\_number() over(PARTITION BY customer\_id ORDER BY order\_date) AS First\_Order

FROM dannys\_diner.sales s

JOIN dannys\_diner.menu m ON

s.product\_id = m.product\_id)

SELECT customer\_id, product\_name--,max(product\_id)

FROM cte3

where First\_Order = 1

\*/

--------------------------------------------------------------

-- 4. What is the most purchased item on the menu

--and how many times was it purchased by all customers?

/\*

WITH cte1 as

(SELECT

product\_id,

COUNT(s.product\_id) as TimesOrdered

FROM dannys\_diner.sales s

GROUP BY s.product\_id

limit 1),

cte2 as

(SELECT

m.product\_id,

TimesOrdered,

m.product\_name FROM cte1

JOIN dannys\_diner.menu m

ON cte1.product\_id = m.product\_id)

SELECT \* FROM cte2

\*/

--------------------------------------------------

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

/\*

WITH cte2 as

(

SELECT

s.customer\_id,

m.product\_id,

COUNT(s.product\_id) as PCount

FROM dannys\_diner.sales s

JOIN dannys\_diner.menu m

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

GROUP BY s.customer\_id,m.product\_id ),

cte3 as

(

SELECT

customer\_id,

product\_name,

row\_number() over(PARTITION BY customer\_id ORDER BY PCOUNT desc) as Max\_Ordered

FROM cte2

JOIN dannys\_diner.menu me

ON cte2.product\_id = me.product\_id

)

SELECT

customer\_id,

product\_name

FROM cte3

WHERE Max\_Ordered = 1

\*/

-----------------------------------------------

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

/\*

WITH cte\_3 as (SELECT s.customer\_id,

s.product\_id,

order\_date,

mu.product\_id,

mu.product\_name,

--MAX(order\_date),

row\_number()

OVER(PARTITION BY s.customer\_id ORDER BY order\_date )

AS First\_Order

FROM dannys\_diner.sales s

JOIN dannys\_diner.members mem

ON s.customer\_id=mem.customer\_id

JOIN dannys\_diner.menu mu

ON s.product\_id = mu.product\_id

WHERE order\_date > mem.join\_date)

SELECT

customer\_id,

product\_name as First\_Order\_Aftr\_Joining

FROM cte\_3

where First\_Order = 1

\*/

---------------------------------------

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

/\*

WITH cte\_4 as (SELECT s.customer\_id,

s.product\_id,

order\_date,

mu.product\_id,

mu.product\_name,

--MAX(order\_date),

row\_number()

OVER(PARTITION BY s.customer\_id ORDER BY order\_date desc)

AS First\_Order

FROM dannys\_diner.sales s

JOIN dannys\_diner.members mem

ON s.customer\_id=mem.customer\_id

JOIN dannys\_diner.menu mu

ON s.product\_id = mu.product\_id

WHERE order\_date <= mem.join\_date)

SELECT

customer\_id,

product\_name as Last\_Order\_B4\_Joining

FROM cte\_4 where First\_Order = 1

\*/

----------------------------------------------------

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

/\*

With cte5 as

(SELECT

s.customer\_id,

s.order\_date,

s.product\_id,

sum(price) over (Partition by s.customer\_id) as Total\_Amt\_Cust

FROM dannys\_diner.sales s

JOIN dannys\_diner.members mem

ON s.customer\_id=mem.customer\_id

JOIN dannys\_diner.menu mu

ON s.product\_id = mu.product\_id

WHERE s.Order\_date < mem.join\_date)

SELECT

customer\_id,

total\_amt\_cust,

COUNT(\*) as Count\_Items

from cte5

GROUP BY customer\_id, total\_amt\_cust

\*/

---------------------------------------------------

-- 9. If each $1 spent equates to 10 points and sushi has a 2x points multiplier - how many points would each customer have?

/\*

WITH cte6 as

(SELECT s.customer\_id, s.product\_id,

CASE

WHEN s.product\_id = 1 THEN mu.price\*2

ELSE mu.price

END cust\_points,

mu.price

FROM dannys\_diner.sales s

JOIN dannys\_diner.menu mu

ON s.product\_id = mu.product\_id)

SELECT

customer\_id,

SUM(cust\_points) as Total\_Cust\_Points

FROM cte6

GROUP BY customer\_id

\*/

-----------------------------------------------------

-- 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 cte7 as

(

SELECT

s.customer\_id,

s.order\_date,

s.product\_id,

mu.price,

mem.join\_date

FROM dannys\_diner.sales s

JOIN dannys\_diner.members mem

ON s.customer\_id=mem.customer\_id

JOIN dannys\_diner.menu mu

ON s.product\_id = mu.product\_id

WHERE EXTRACT(month from s.Order\_date) = 1

),

cte8 as

(

select \*,(date(order\_date)-date(join\_date) ),

CASE WHEN ((date(order\_date)-date(join\_date)) between 0 AND 7) THEN price\*2

ELSE price

END As cust\_points

FROM cte7)

SELECT

customer\_id,

SUM(cust\_points)

FROM cte8

WHERE ((date(order\_date)-date(join\_date)) between 0 AND 7)

GROUP BY customer\_id

\*/