**Danny Ma’s Challenge**

**Tables:**

CREATE TABLE sales (

customer\_id VARCHAR(1),

order\_date DATE,

product\_id INTEGER);

INSERT INTO sales

(customer\_id, order\_date, product\_id)

VALUES

('A', '2021-01-01', 1),

('A', '2021-01-01', 2),

('A', '2021-01-07', 2),

('A', '2021-01-10', 3),

('A', '2021-01-11', 3),

('A', '2021-01-11', 3),

('B', '2021-01-01', 2),

('B', '2021-01-02', 2),

('B', '2021-01-04', 1),

('B', '2021-01-11', 1),

('B', '2021-01-16', 3),

('B', '2021-02-01', 3),

('C', '2021-01-01', 3),

('C', '2021-01-01', 3),

('C', '2021-01-07', 3);

CREATE TABLE menu (

"product\_id" INTEGER,

"product\_name" VARCHAR(5),

"price" INTEGER);

INSERT INTO menu

(product\_id, product\_name, price)

VALUES (1, 'sushi', 10), (2, 'curry', 15), (3, 'ramen', 12);

CREATE TABLE members (

customer\_id VARCHAR,

join\_date DATE);

INSERT INTO members

(customer\_id, join\_date) VALUES ('A', '2021-01-07'), ('B', '2021-01-09');

**Entity Relationship Diagram**

A picture containing text, screenshot, font, number

Description automatically generated

**Queries:**

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

select s.customer\_id, SUM(m.price) as total\_amount\_spent from menu m join sales s

on m.product\_id = s.product\_id group by s.customer\_id

|  |  |
| --- | --- |
| **customer\_id** | **total\_amount\_spent** |
| A | 76 |
| B | 74 |
| C | 36 |

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

select customer\_id, COUNT(distinct order\_date) as visitors\_count from sales

group by customer\_id

|  |  |
| --- | --- |
| **customer\_id** | **visitors\_count** |
| A | 4 |
| B | 6 |
| C | 2 |

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

with first as

(

select s.customer\_id, s.order\_date,s.product\_id,m.product\_name,

row\_number() over (partition by s.customer\_id order by s.product\_id) rn

from sales s join menu m on s.product\_id = m.product\_id

)

select \* from first where rn = 1

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **customer\_id** | **order\_date** | **product\_id** | **product\_name** | **rn** |
| A | 01-01-2021 | 1 | sushi | 1 |
| B | 04-01-2021 | 1 | sushi | 1 |
| C | 01-01-2021 | 3 | ramen | 1 |

4. What is the most purchased item on the menu and how many times was it purchased

by all customers?

select s.product\_id, m.product\_name, count(s.product\_id) as purchase\_count

from sales s join menu m on s.product\_id = m.product\_id

group by s.product\_id,m.product\_name order by count(s.product\_id) desc

|  |  |  |
| --- | --- | --- |
| **product\_id** | **product\_name** | **purchase\_count** |
| 3 | ramen | 8 |
| 2 | curry | 4 |
| 1 | sushi | 3 |

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

with popular as

(

select s.customer\_id, s.product\_id, m.product\_name,count(m.product\_name) as popular\_item,

DENSE\_RANK() over (partition by s.customer\_id order by count(s.product\_id) desc) as rn

from sales s join menu m on m.product\_id = s.product\_id

group by s.customer\_id,s.product\_id,m.product\_name

)

select \* from popular where rn = 1

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **customer\_id** | **product\_id** | **product\_name** | **popular\_item** | **rn** |
| A | 3 | ramen | 3 | 1 |
| B | 1 | sushi | 2 | 1 |
| B | 2 | curry | 2 | 1 |
| B | 3 | ramen | 2 | 1 |
| C | 3 | ramen | 3 | 1 |

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

with first\_purchase as

(

select s.customer\_id, s.order\_date,s.product\_id,m.product\_name, me.join\_date,

row\_number() over (partition by s.customer\_id order by s.product\_id) rn

from sales s join menu m on s.product\_id = m.product\_id join members me

on me.customer\_id = s.customer\_id

where s.order\_date > me.join\_date

group by s.customer\_id, s.order\_date,s.product\_id,m.product\_name,me.join\_date

)

select \* from first\_purchase where rn = 1

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **customer\_id** | **order\_date** | **product\_id** | **product\_name** | **join\_date** | **rn** |
| A | 10-01-2021 | 3 | ramen | 07-01-2021 | 1 |
| B | 11-01-2021 | 1 | sushi | 09-01-2021 | 1 |

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

with purchase as

(

select s.customer\_id, s.order\_date,s.product\_id,m.product\_name, me.join\_date,

row\_number() over (partition by s.customer\_id order by s.order\_date desc) rn

from sales s join menu m on s.product\_id = m.product\_id join members me

on me.customer\_id = s.customer\_id

where s.order\_date < me.join\_date

group by s.customer\_id, s.order\_date,s.product\_id,m.product\_name,me.join\_date

)

select \* from purchase where rn = 1

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **customer\_id** | **order\_date** | **product\_id** | **product\_name** | **join\_date** | **rn** |
| A | 01-01-2021 | 1 | sushi | 07-01-2021 | 1 |
| B | 04-01-2021 | 1 | sushi | 09-01-2021 | 1 |

--8. What is the total items and amount spent for each member

before they became a member?

select s.customer\_id,

count(s.product\_id) as total\_items, SUM(m.price) as amount\_spent

from sales s join menu m on s.product\_id = m.product\_id join members me

on me.customer\_id = s.customer\_id

where s.order\_date < me.join\_date

group by s.customer\_id

|  |  |  |
| --- | --- | --- |
| **customer\_id** | **total\_items** | **amount\_spent** |
| A | 2 | 25 |
| B | 3 | 40 |

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

how many points would each customer have?

select customer\_id, sum(points) as total\_points from

(

select s.customer\_id, m.product\_name,count(product\_name) as quantity,

case when product\_name = 'sushi' then count(product\_name) \* 20

else count(product\_name) \* 10

end as points

from

sales s join menu m on s.product\_id = m.product\_id

group by s.customer\_id, m.product\_name

)x group by customer\_id

|  |  |
| --- | --- |
| **customer\_id** | **total\_points** |
| A | 70 |
| B | 80 |
| C | 30 |

--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?

select customer\_id, sum(points) as total\_points from

(

select s.customer\_id, s.order\_date,count(product\_name) as quantity,

count(product\_name) \* 20 as points

from sales s join menu m on s.product\_id = m.product\_id

join members me on me.customer\_id = s.customer\_id

where s.order\_date > = me.join\_date and MONTH(s.order\_date)=1

group by s.customer\_id, s.order\_date

)x group by customer\_id

|  |  |
| --- | --- |
| **customer\_id** | **total\_points** |
| A | 80 |
| B | 40 |

--Join All The 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 s.order\_date> me.join\_date then 'Y'

else 'N'

end as Member

from menu m join sales s on m.product\_id = s.product\_id

join members me on me.customer\_id = s.customer\_id

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **customer\_id** | **order\_date** | **product\_name** | **price** | **Member** |
| A | 01-01-2021 | sushi | 10 | N |
| A | 01-01-2021 | curry | 15 | N |
| A | 07-01-2021 | curry | 15 | N |
| A | 10-01-2021 | ramen | 12 | Y |
| A | 11-01-2021 | ramen | 12 | Y |
| A | 11-01-2021 | ramen | 12 | Y |
| B | 01-01-2021 | curry | 15 | N |
| B | 02-01-2021 | curry | 15 | N |
| B | 04-01-2021 | sushi | 10 | N |
| B | 11-01-2021 | sushi | 10 | Y |
| B | 16-01-2021 | ramen | 12 | Y |
| B | 01-02-2021 | ramen | 12 | Y |

Rank All The Things

Danny also requires further information about the ranking of customer products, but he purposely does not need the ranking for non-member purchases so he expects null ranking values for the records when customers are not yet part of the loyalty program

With cte as

(

select s.customer\_id, s.order\_date, m.product\_name,m.price,

case when s.order\_date> me.join\_date then 'Y'

else 'N'

end as Member

from menu m join sales s on m.product\_id = s.product\_id

join members me on me.customer\_id = s.customer\_id)

select \*,

case when cte.Member = 'N' then NULL

else

rank() over (partition by cte.customer\_id,cte.member order by cte.order\_date)

end as ranking from cte

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **customer\_id** | **order\_date** | **product\_name** | **price** | **Member** | **ranking** |
| A | 01-01-2021 | sushi | 10 | N | NULL |
| A | 01-01-2021 | curry | 15 | N | NULL |
| A | 07-01-2021 | curry | 15 | N | NULL |
| A | 10-01-2021 | ramen | 12 | Y | 1 |
| A | 11-01-2021 | ramen | 12 | Y | 2 |
| A | 11-01-2021 | ramen | 12 | Y | 2 |
| B | 01-01-2021 | curry | 15 | N | NULL |
| B | 02-01-2021 | curry | 15 | N | NULL |
| B | 04-01-2021 | sushi | 10 | N | NULL |
| B | 11-01-2021 | sushi | 10 | Y | 1 |
| B | 16-01-2021 | ramen | 12 | Y | 2 |
| B | 01-02-2021 | ramen | 12 | Y | 3 |

**Insights:**

* They were 3 customers A, B, C. A being the highest spent customer at the restaurant.
* Customer B visited the restaurant more number of times among the three (with a total of 6). This can happen before becoming a member too.
* Sushi & Ramen were the most preferred and are firstly ordered by the customers even before becoming the member and after too.
* Before becoming a member, itself, B ordered the highest number of items and highest amount spent.