



THE TASTE OF SUCCESS

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 our 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.

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.

Schema

Entity Relationship Diagram

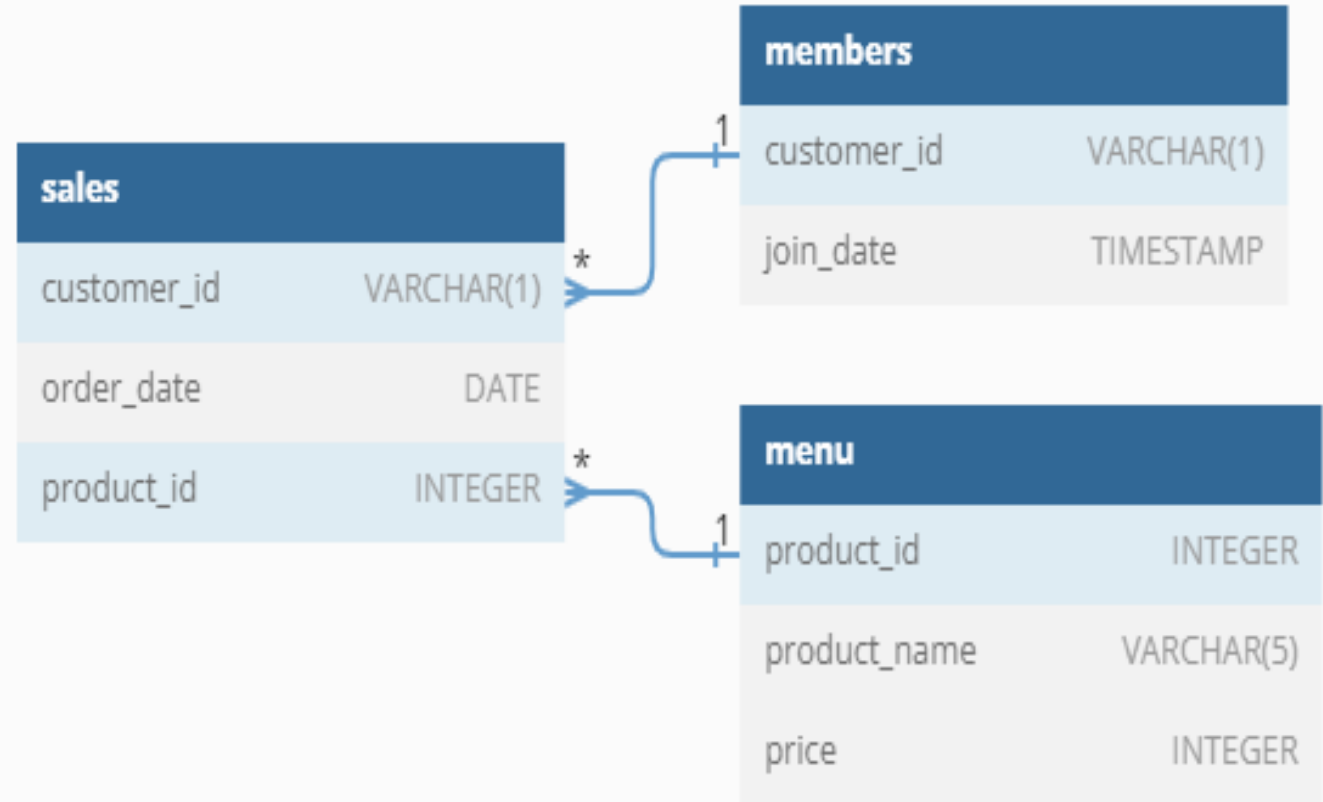


Table 1: Sales

customer_id	order_date	product_id
A	01-01-2021	1
A	01-01-2021	2
A	07-01-2021	2
A	10-01-2021	3
A	11-01-2021	3
A	11-01-2021	3
B	01-01-2021	2
B	02-01-2021	2
B	04-01-2021	1
B	11-01-2021	1
B	16-01-2021	3
B	01-02-2021	3
C	01-01-2021	3
C	01-01-2021	3
C	07-01-2021	3

Table 2: Menu

product_id	product_name	price
1	sushi	10
2	curry	15
3	ramen	12

Table 3: Members

customer_id	join_date
A	07-01-2021
B	09-01-2021
A	07-01-2021
B	09-01-2021

Generating Insights using SQL



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

```
SELECT
    customer_id, SUM(m.price) AS total_spent
FROM
    sales s
    JOIN
    menu m ON s.product_id = m.product_id
GROUP BY customer_id;
```

OUTPUT-

customer_id	total_spent
A	76
B	74
C	36

INSIGHTS-

It shows the total amount spent by each customer at the restaurant indicating spending potential of a customer on food.

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

```
SELECT
    customer_id, COUNT(DISTINCT order_date) AS
    days_visited
FROM
    sales
GROUP BY customer_id;
```

OUTPUT-

customer_id	days_visited
A	4
B	6
C	2

INSIGHTS-

1. It shows Visiting patterns of customer, how many times a customer visits the restaurants showing some visits often and some visits less.

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

```
WITH cte
  AS (SELECT s.customer_id,
            order_date,
            s.product_id,
            m.product_name,
            Rank()
              OVER(
                partition BY s.customer_id
                ORDER BY order_date) AS rn
  FROM   sales s
        JOIN menu m
          ON s.product_id = m.product_id)
SELECT customer_id,
       product_name
FROM   cte
WHERE  rn = 1;
```

OUTPUT-

customer_id	product_name
A	sushi
A	curry
B	curry
C	ramen
C	ramen

INSIGHTS-

1.First item purchase by the customers shows their first choices which will help danny in knowing entry items.

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(s.product_id) AS
total_purchase
FROM
    menu m
    JOIN
    sales s ON s.product_id = m.product_id
GROUP BY m.product_name
ORDER BY COUNT(s.product_id) DESC
LIMIT 1;
```

OUTPUT-

product_name	total_purchase
ramen	8

INSIGHTS-

1.It shows most popular item among the customer which will help danny to manage the inventory of that particular item.

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

```
WITH cte
  AS (SELECT product_name,
            customer_id,
            Count(product_name) AS order_count,
            Rank()
              OVER(
                partition BY customer_id
                ORDER BY Count(m.product_name)DESC
              ) AS rn
  FROM   sales s
        JOIN menu m
          ON s.product_id = m.product_id
        GROUP BY customer_id,
                product_name)
SELECT DISTINCT ( customer_id ),
               product_name,
               order_count
FROM   cte
WHERE  rn = 1;
```

OUTPUT-

Customer_id	Product_name	Order_count
A	ramen	3
B	curry	2
B	sushi	2
B	ramen	2
C	ramen	3

INSIGHTS-

1.By knowing the popular item purchased by a particular customer danny can provide particular menu suggestions.

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

```
with cte as (SELECT
    menu.product_name,sales.customer_id,dense_rank()
over(partition by sales.customer_id order by order_date
) as rn
FROM
    menu
    JOIN
    sales ON sales.product_id = menu.product_id
    JOIN
    members ON sales.customer_id = members.customer_id
WHERE
    sales.order_date > members.join_date
)
select distinct customer_id,product_name from cte
where rn=1 ;
```

OUTPUT-

customer_id	product_name
A	ramen
B	sushi

INSIGHTS-

1.By knowing the first purchase after membership helps danny to know the engagement of a customer after loyalty program and how much effective it is and does it increase the sale of a particular menu.

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

```
with cte as (SELECT
    menu.product_name,sales.customer_id,dense_rank()
over(partition by sales.customer_id order by order_date
desc ) as rn
FROM
    menu
    JOIN
    sales ON sales.product_id = menu.product_id
    JOIN
    members ON sales.customer_id = members.customer_id
WHERE
    sales.order_date < members.join_date
)
select distinct customer_id,product_name from cte
where rn=1 ;
```

OUTPUT-

customer_id	product_name
A	curry
A	sushi
B	sushi

INSIGHTS-

1. It shows the last purchased dish by a customer.

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

```
SELECT
    s.customer_id,
    COUNT(product_name) AS total_count,
    SUM(price) AS amt_spent
FROM
    menu m
    INNER JOIN
    sales s ON m.product_id = s.product_id
    INNER JOIN
    members mem ON mem.customer_id = s.customer_id
WHERE
    order_date < join_date
GROUP BY s.customer_id;
```

OUTPUT-

customer_ id	total_items_ purchased	amt_ spent
A	2	25
B	3	40

INSIGHTS-

1. It shows the spending of a customer before loyalty program showing the impact of program by differentiating their spending before and after membership.

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(CASE
        WHEN product_name = 'sushi' THEN price * 20
        ELSE price * 10
    END) AS total_points
FROM
    sales
    JOIN
    menu ON sales.product_id = menu.product_id
GROUP BY sales.customer_id;
```

OUTPUT-

customer_id	total_points
A	860
B	940
C	360

INSIGHTS-

1. Giving points to each customer will increase customer engagement in loyalty program.

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
    sales.customer_id,
    SUM(CASE
        WHEN
            sales.order_date BETWEEN members.join_date
            AND DATE_ADD(members.join_date,
                INTERVAL 6 DAY)
        THEN
            menu.price * 20
        END) AS points
FROM
    sales
    JOIN
    menu ON sales.product_id = menu.product_id
    JOIN
    members ON sales.customer_id = members.customer_id
WHERE
    order_date <= '2021-01-31'
GROUP BY sales.customer_id;
```

OUTPUT-

customer_id	points
A	2040
B	400

INSIGHTS-

1. Giving double points to the customer in their first week of membership will increase their interest in loyalty program.

THANK YOU !!

-By Khushi Singhal