SQL CASE STUDY

Danny’s Diner

#8weekSQLChallenge

Krishna Sameera Kota

8WEEKSQLCHALLENGE.COM

CASE STUDY #1

THE TASTE OF SUCCESS

DATAWITHDANNY.COM

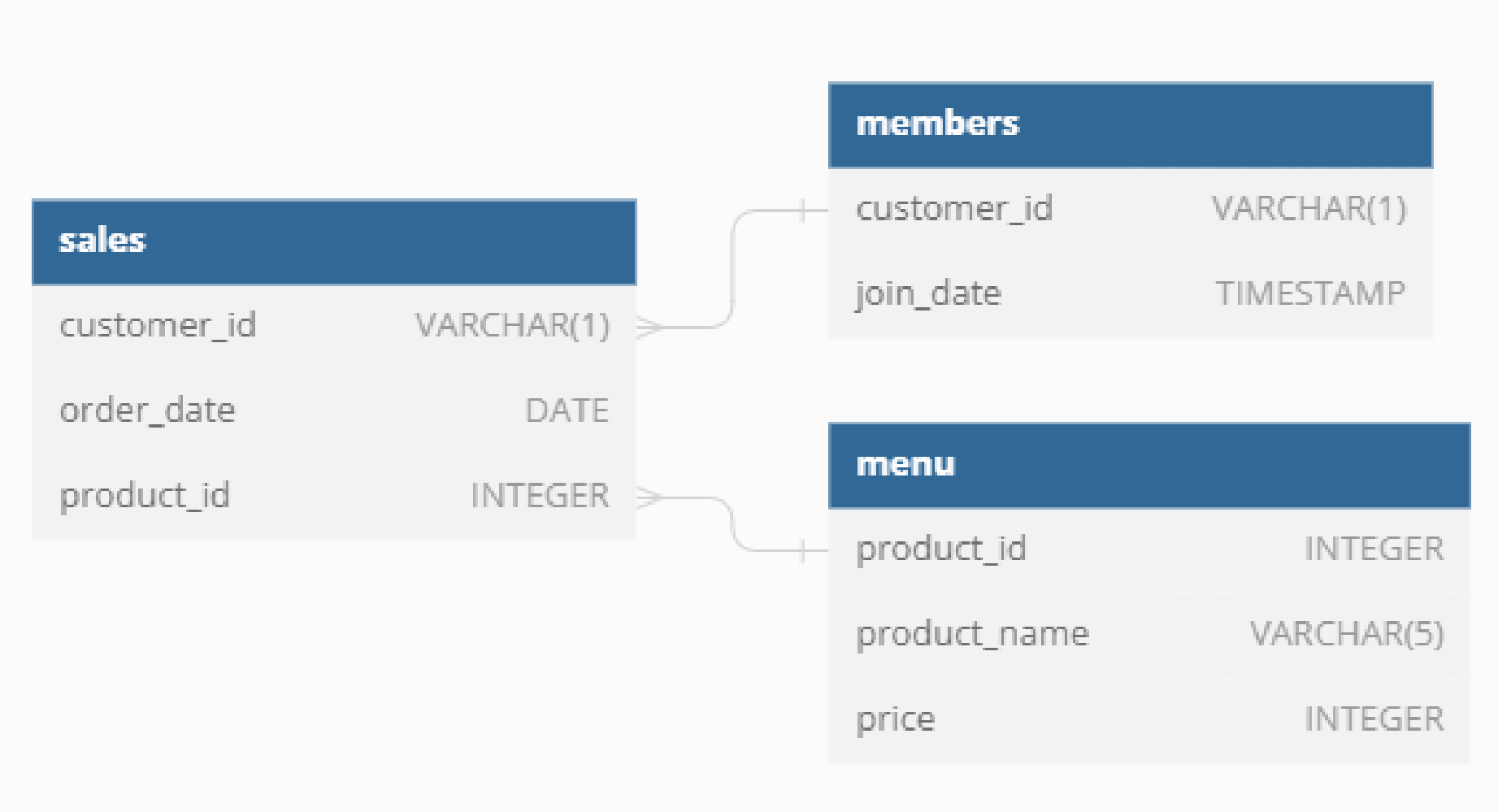
INTRODUCTION:

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 favorite 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.

---------- Krishna Sameera Kota

DATA MODEL


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

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

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

customer_id	total_amount
A	76
B	74
C	36

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

```sql
SELECT
customer_id,
COUNT(order_date) AS total_days
FROM sales
GROUP BY customer_id;
```

customer_id	total_days
A	6
B	6
C	3

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

```
WITH sale_rankings AS (
    SELECT customer_id, order_date, product_name,
    RANK() OVER (PARTITION BY s.customer_id ORDER BY s.order_date) AS order_rank
    FROM sales s
    JOIN menu m ON s.product_id = m.product_id
    )
    SELECT customer_id,
    max(product_name) as first_purchased_item
    FROM sale_rankings
    WHERE order_rank = 1
    GROUP BY customer_id;
```

customer_id	total_days
A	6
B	6
C	3

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

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

product_name	purchase_count
ramen	8

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

```sql
WITH Popularity as (
SELECT
s.customer_id,
m.product_name,
count(m.product_name) as Purchase_count,
row_number() over (partition by s.customer_id order by count(m.product_name) desc) as Popular_item
from sales s
join menu m on s.product_id = m.product_id
group by s.customer_id, m.product_name)
SELECT
    customer_id,
    product_name,
    purchase_count
FROM popularity
WHERE popular_item = 1;
```

customer_id	product_name	purchase_count
A	ramen	3
B	curry	2
C	ramen	3

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

```
WITH first_purchase AS (
SELECT
s.customer_id,
m.product_name,
s.order_date,
DENSE_RANK() OVER (PARTITION BY s.customer_id ORDER BY s.order_date) AS purchase_rank
FROM sales s
JOIN menu m ON s.product_id = m.product_id
JOIN members mem ON s.customer_id = mem.customer_id
WHERE s.order_date >= mem.join_date )
SELECT
customer_id, product_name, order_date
FROM first_purchase
WHERE purchase_rank = 1;
```

customer_id	product_name	order_date
A	curry	2021-01-07
A	curry	2021-01-07
B	sushi	2021-01-11
B	sushi	2021-01-11

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

```sql
WITH purchase_before_membership AS (
SELECT
s.customer_id,
m.product_name,
s.order_date,
DENSE_RANK() OVER (PARTITION BY s.customer_id ORDER BY s.order_date DESC) AS purchase_rank
FROM sales s
JOIN menu m ON s.product_id = m.product_id
JOIN members mem ON s.customer_id = mem.customer_id
WHERE s.order_date <= mem.join_date )

SELECT
customer_id, product_name, order_date
FROM purchase_before_membership
WHERE purchase_rank = 1;
```

customer_id	product_name	order_date
A	curry	2021-01-07
A	curry	2021-01-07
B	sushi	2021-01-04
B	sushi	2021-01-04

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

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

customer_id	total_products	amount_spent
A	2	50
B	2	80

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

```sql
SELECT
s.customer_id,
SUM(CASE WHEN m.product_name = 'sushi' THEN 2 * 10
    ELSE 10
    END * m.price) AS total_points
FROM sales s
JOIN menu m ON s.product_id = m.product_id
GROUP BY s.customer_id;
```

customer_id	total_points
A	860
B	940
C	360

Thankyou