Danny‘s Diner Sql Case Study

By Ashish Jha

Project Overview

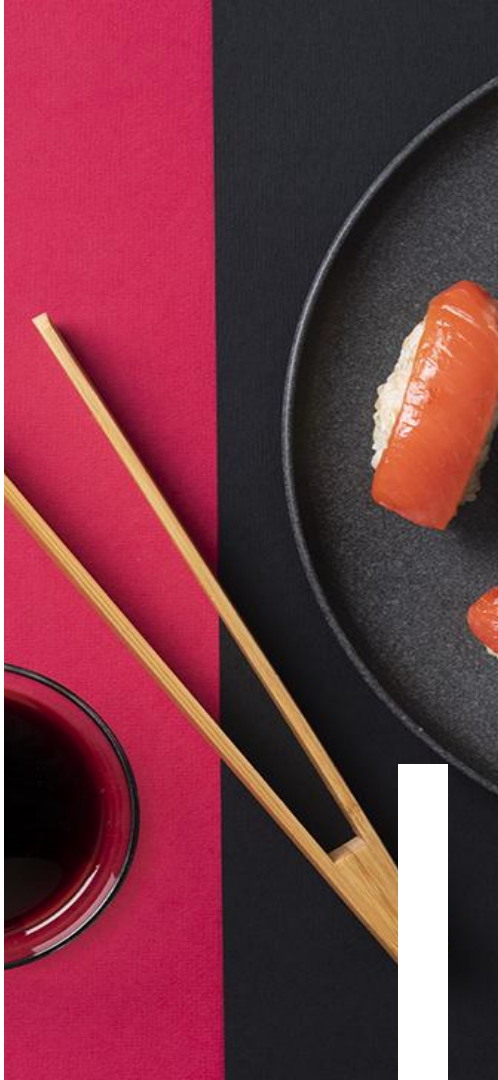
Danny, the owner of a sushi and ramen restaurant, wants to gain **deeper insights into customer behavior** to improve personalization and make data-driven business decisions.

He is particularly interested in:

- Understanding **customer visiting patterns**
- Identifying **total spending** by each customer
- Discovering **favourite menu items**

These insights will help him evaluate the **potential expansion of his customer loyalty program** and improve overall customer experience.

Due to privacy concerns, Danny has shared a **sample of his actual customer data**, hoping that the provided examples are sufficient to build real SQL queries and extract meaningful business insights



Database Schema

```
CREATE SCHEMA dannys_diner;
SET search_path = dannys_diner;

CREATE TABLE sales (
  "customer_id" VARCHAR(1),
  "order_date" DATE,
  "product_id" INTEGER
);
```

```
CREATE TABLE menu (
  "product_id" INTEGER,
  "product_name" VARCHAR(5),
  "price" INTEGER);
```

```
CREATE TABLE members (
  "customer_id" VARCHAR(1),
  "join_date" DATE
);
```

Solved 7 real life Business Problem

1. What is the total amount each customer spent at the restaurant?
2. How many days has each customer visited the restaurant?
3. What was the first item from the menu purchased by each customer?
4. What is the most purchased item on the menu and how many times was it purchased by all customers?
5. Which item was the most popular for each customer?
6. Which item was purchased first by the customer after they became a member?
7. Which item was purchased just before the customer became a member?

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

```
select c.customer_id,sum(m.price) as total_amount_spent

from sales c join menu m on c.product_id = m.product_id

group by c.customer_id

order by total_amount_spent desc;
```

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

```
with first_order as (
    select customer_id,
        MIN(order_date) AS first_order_date
  from sales
  group by customer_id )
select c.customer_id,m.product_name
from sales c join first_order f on c.customer_id = f.customer_id
 and c.order_date = f.first_order_date
join menu m on c.product_id = m.product_id
order by c.customer_id;
```

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

```
select m.product_name , count(*) as totol_purchase

from sales c

join menu m on c.product_id = m.product_id

group by m.product_name

order by totol_purchase desc

limit 1;
```

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

```
with c1 as (
  select c.customer_id,m.product_name,count(*) as purchase_count,
  row_number () over(partition by c.customer_id order by count(*) desc) as rn
  from sales c join menu m on c.product_id=m.product_id
  group by c.customer_id,m.product_name
)
SELECT
    customer_id,
    product_name,
    purchase_count
from c1
 where rn = 1
 order by customer_id;
```

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

```
select m.product_name , count(*) as totol_purchase

from sales c

join menu m on c.product_id = m.product_id

group by m.product_name

order by totol_purchase desc

limit 1;
```

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

```
select customer_id, count(DISTINCT order_date) as visit_day

from sales

group by customer_id

order by visit_day desc;
```

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

```
with alll as (
select c.customer_id,c.order_date,c.product_id ,p.join_date
from sales c join members p on c.customer_id = p.customer_id
where c.order_date<p.join_date),
   c1 as (
 select cs.customer_id,cs.order_date,m.product_name,
 row_number()over (partition by cs.customer_id order by cs.order_date desc) rn
 from alll cs join menu m on cs.product_id= m.product_id
)
select customer_id,product_name,order_date from c1
where rn =1;
```

SUMMARY OF INSIGHTS

🍜 Ramen is the **most purchased** item overall

→ Suggests it could be used in promotions or **loyalty rewards.**

💰 **Customer A** spent the most across all orders

→ **High-value customer** — good for personalized targeting.

📆 Most customers visited on 4–6 distinct days

→ Indicates decent retention; supports weekly campaign timing.

💳 Purchases **after joining membership** were higher-value

→ **Loyalty program seems to increase engagement and spending.**

🔁 Customers ordered different items before & after joining

→ Shows **membership influences purchasing behavior**.

THANK YOU