**SQL Case Study**

**Introduction**

Danny has a severe love for Japanese cuisine, so in the beginning of 2021 he makes the dangerous decision to build a cute little restaurant serving his three favourite dishes, sushi, curry, and ramen.

Danny's Diner is in need of your aid to stay in business. Although they have collected some very basic data over the course of their short time in company, they are unsure of how to use it.

**Problem Statement**

Danny wants to use the information to find simple answers about his customers, particularly about their spending habits, frequency of visits, and favourite menu items. He will be able to provide his devoted customers a better and more individualised experience thanks to this deeper connection with them.

He intends to use these insights to inform his decision about whether to expand the current customer loyalty programme. He also needs assistance in creating some simple datasets so his team can quickly examine the data without the need for SQL.

Due to privacy concerns, Danny has only given you a sample of his entire customer database, but he hopes that these examples will be sufficient to inspire you to create fully functional SQL queries that will assist him in finding the answers to his queries.

Danny has shared with you 3 key datasets for this case study:

* sales
* menu
* members

You can inspect the entity relationship diagram and example data below.

Graphical user interface, application, website

Description automatically generated

## Example Datasets

All datasets exist within the dannys\_diner database schema - be sure to include this reference within your SQL scripts as you start exploring the data and answering the case study questions.

### Table 1: sales

The sales table captures all customer\_id level purchases with a corresponding order\_date and product\_id information for when and what menu items were ordered.

| **customer\_id** | **order\_date** | **product\_id** |
| --- | --- | --- |
| 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 |

### Table 2: menu

The menu table maps the product\_id to the actual product\_name and price of each menu item.

| **product\_id** | **product\_name** | **price** |
| --- | --- | --- |
| 1 | sushi | 10 |
| 2 | curry | 15 |
| 3 | ramen | 12 |

### Table 3: members

The final members table captures the join\_date when a customer\_id joined the beta version of the Danny’s Diner loyalty program.

| **customer\_id** | **join\_date** |
| --- | --- |
| A | 2021-01-07 |
| B | 2021-01-09 |

**Case Study Questions**

Each of the following case study questions can be answered using a single SQL statement:

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?
8. What are the total items and amount spent for each member before they became a member?
9. If each $1 spent equates to 10 points and sushi has a 2x points multiplier - how many points would each customer have?
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?