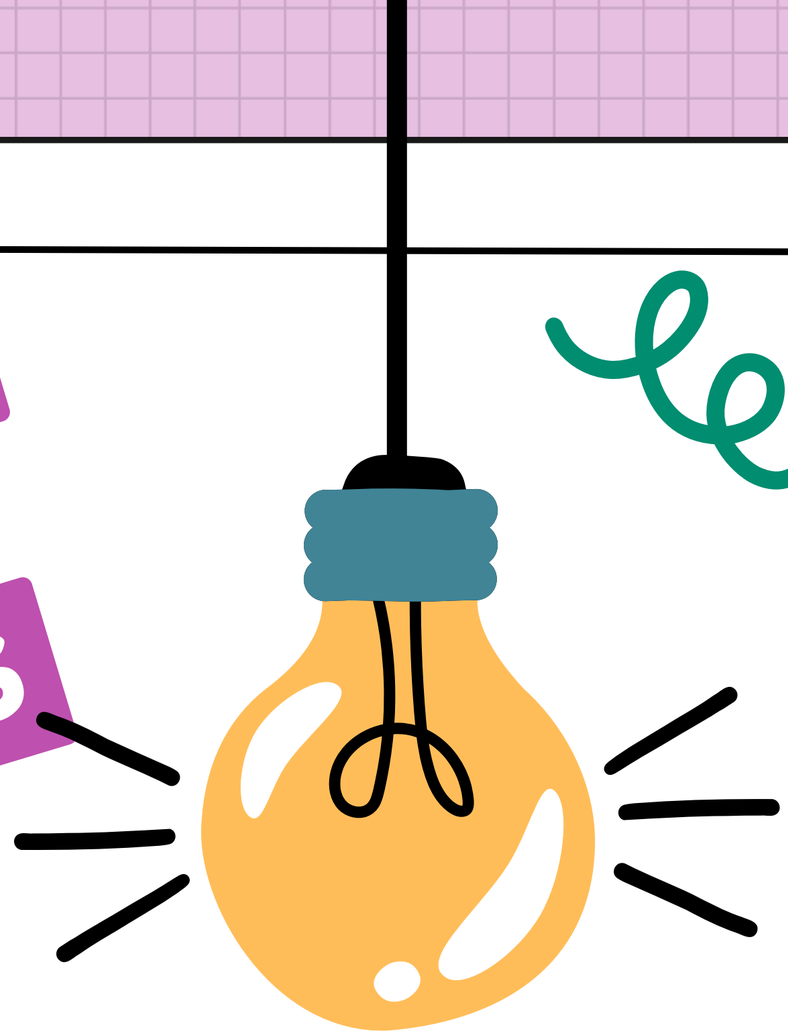
RESTAURANT CUSTOMER INSIGHTS AND LOYALTY PROGRAM ANALYSIS



SQL

DATABASE

- `CREATE DATABASE project;`
- `USE project;`

TABLES

SALES

```sql
CREATE TABLE sales(
    customer_id VARCHAR(1),
    order_date DATE,
    product_id INT
);

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);
```

MENU

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

INSERT INTO menu
    (product_id, product_name, price)
VALUES
    (1, 'sushi', 10),
    (2, 'curry', 15),
    (3, 'ramen', 12);
```

MEMBERS

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

INSERT INTO members
    (customer_id, join_date)
VALUES
    ('A', '2021-01-07'),
    ('B', '2021-01-09');
```

WHAT IS THE TOTAL AMOUNT EACH CUSTOMER SPENT AT THE RESTAURANT?

```sql
-- Total amount each customer spent at the restaurant
SELECT s.customer_id, SUM(m.price) AS total_spent
FROM sales s
JOIN menu m ON s.product_id = m.product_id
GROUP BY s.customer_id;
```

customer_id	total_spent
A	76
B	74
C	36

HOW MANY DAYS HAS EACH CUSTOMER VISITED THE RESTAURANT?

```sql
-- Number of days each customer visited the restaurant
SELECT customer_id, COUNT(DISTINCT order_date) AS visit_days
FROM sales
GROUP BY customer_id;
```

customer_id	visit_days
A	4
B	6
C	2

WHAT WAS THE FIRST ITEM FROM THE MENU PURCHASED BY EACH CUSTOMER?

```sql
-- First item from the menu purchased by each customer
SELECT s.customer_id, m.product_name AS first_item_purchased, s.order_date AS first_purchase_date
FROM sales s
JOIN menu m ON s.product_id = m.product_id
JOIN (
    SELECT customer_id, MIN(order_date) AS first_purchase_date
    FROM sales
    GROUP BY customer_id
) AS first_order ON s.customer_id = first_order.customer_id AND s.order_date = first_order.first_purchase_date;
```

customer_id	first_item_purchased	first_purchase_date
A	sushi	2021-01-01
A	curry	2021-01-01
B	curry	2021-01-01
C	ramen	2021-01-01
C	ramen	2021-01-01

WHAT IS THE MOST PURCHASED ITEM ON THE MENU AND HOW MANY TIMES WAS IT PURCHASED BY ALL CUSTOMERS?

```sql
-- Most purchased item on the menu and its purchase count
SELECT m.product_name, COUNT(*) 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

IF EACH $1 SPENT EQUATES TO 10 POINTS AND SUSHI HAS A 2X POINTS MULTIPLIER - HOW MANY POINTS WOULD EACH CUSTOMER HAVE?

```sql
-- If each $1 spent equates to 10 points and sushi has a 2x points multiplier -
SELECT s.customer_id,
       SUM(CASE
               WHEN m.product_name = 'sushi' THEN m.price * 2 * 10
               ELSE m.price * 10
           END) AS total_points
FROM sales s
JOIN menu m ON s.product_id = m.product_id
GROUP BY s.customer_id;
```

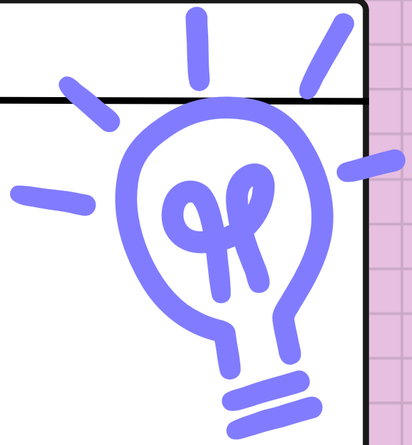
customer_id	total_points
A	860
B	940
C	360

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?

```sql
SELECT s.customer_id,
       SUM(CASE
               WHEN s.order_date <= DATE_ADD(mem.join_date, INTERVAL 7 DAY) THEN m.price * 2 * 10
               ELSE m.price * 10
           END) AS total_points
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 <= '2021-01-31'
GROUP BY s.customer_id;
```

customer_id	total_points
B	1240
A	1520

CONCLUSIONS



THIS PROJECT UTILIZES SQL QUERIES TO EXPLORE CUSTOMER DATA, ANSWER KEY BUSINESS QUESTIONS, AND HELP THE RESTAURANT MAKE DATA-DRIVEN DECISIONS RELATED TO CUSTOMER ENGAGEMENT, PRODUCT OFFERINGS, AND LOYALTY REWARDS PROGRAMS.