Danny's Diner Case Study

Created by: Mohd Wasi

Email: wasimohd.2001@gmail.com

```
CREATE DATABASE CASE_STUDY;
USE CASE_STUDY;

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

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');

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

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

```
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');
```

PROBLEMS STATEMENTS

-- Q1 What is the total amount each customer spent at the restaurant?

```
SELECT S.CUSTOMER_ID,SUM(M.PRICE) AS 'TOTAL AMOUNT'
FROM SALES S INNER JOIN MENU M ON S.PRODUCT_ID = M.PRODUCT_ID
GROUP BY S.CUSTOMER_ID;
```

-- Q2 How many days has each customer visited the restaurant?

```
SELECT CUSTOMER_ID,COUNT(DISTINCT(ORDER_DATE)) FROM SALES
GROUP BY CUSTOMER_ID;
```

-- Q3 What was the first item from the menu purchased by each customer?

```
SELECT DISTINCT S.CUSTOMER_ID,M.PRODUCT_NAME FROM SALES S
INNER JOIN MENU M ON S.PRODUCT_ID = M.PRODUCT_ID
WHERE S.ORDER_DATE = ANY(SELECT MIN(ORDER_DATE) FROM SALES GROUP BY CUSTOMER_ID);
SELECT * FROM SALES;
```

-- Q4 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 'PRODUCT NAME' FROM SALES S
INNER JOIN MENU M ON S.PRODUCT_ID = M.PRODUCT_ID GROUP BY M.PRODUCT_NAME
ORDER BY 'PRODUCT NAME' DESC;
```

-- Q5 Which item was the most popular for each customer?

```
WITH R as
(SELECT s.customer_id, m.product_name, count(s.product_id) as [count],
dense_rank() OVER(PARTITION BY s.customer_id ORDER BY COUNT(s.product_id) DESC) as [rank]
FROM sales as s
INNER JOIN menu as m
ON s.product_id = m.product_id
GROUP BY s.customer_id,m.product_name,s.product_id)
SELECT customer_id, product_name, count , rank
FROM R
WHERE rank = 1
```

-- Q6 Which item was purchased first by the customer after they became a member?

```
WITH RankedProducts AS (
    SELECT
        s.customer_id,
        m.product_name,
        DENSE_RANK() OVER (PARTITION BY s.customer_id ORDER BY COUNT(*) DESC) AS r
    FROM Sales s
    JOIN Menu m ON m.product_id = s.product_id
    JOIN Members mem ON mem.customer_id = s.customer_id
    WHERE s.order_date >= mem.join_date
    GROUP BY s.customer_id, m.product_name
)
SELECT
    customer_id,
    product_name
FROM RankedProducts
WHERE r = 1;
```

-- Q7 In the first week after a customer joins the program (including their join date)

```
WITH R as(
SELECT s.customer_id, m.product_name, dense_rank() OVER(PARTITION BY s.customer_id ORDER BY
s.order_date desc) as r
FROM sales as s
INNER JOIN menu as m
ON s.product_id = m.product_id
INNER JOIN  members as mem
ON s.customer_id = mem.customer_id
WHERE s.order_date < mem.join_date
```

```
)
SELECT customer_id, product_name
FROM R
WHERE r = 1;
```

-- Q8 What is the total items and amount spent for each member before they became a member?

```
SELECT s.customer_id,count(s.product_id) as total_items,sum(m.price) as total_amount
From sales as s
INNER JOIN menu as m
ON s.product_id = m.product_id
INNER JOIN members as mem
ON s.customer_id = mem.customer_id
WHERE order_date < join_date
GROUP BY s.customer_id;
```

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

```
WITH R as
(SELECT *,
CASE
WHEN m.product_name = 'sushi' THEN price * 20
WHEN m.product_name != 'sushi' THEN price * 10
END as points
FROM menu m)
SELECT customer_id, SUM(points) as points
FROM sales as s
INNER JOIN R as r
ON s.product_id = r.product_id
GROUP BY s.customer_id;
```

-- Question 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
    customer_id,
    SUM(Earning_Point) AS Total_earning_point
FROM (
    SELECT
        s.customer_id,
        s.order_date,
```