SQL
Data Analysis Project

- *Shivam Singh | Data Analyst*

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 favourite 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 its few months of operation but has no idea how to use its data to help them run the business.

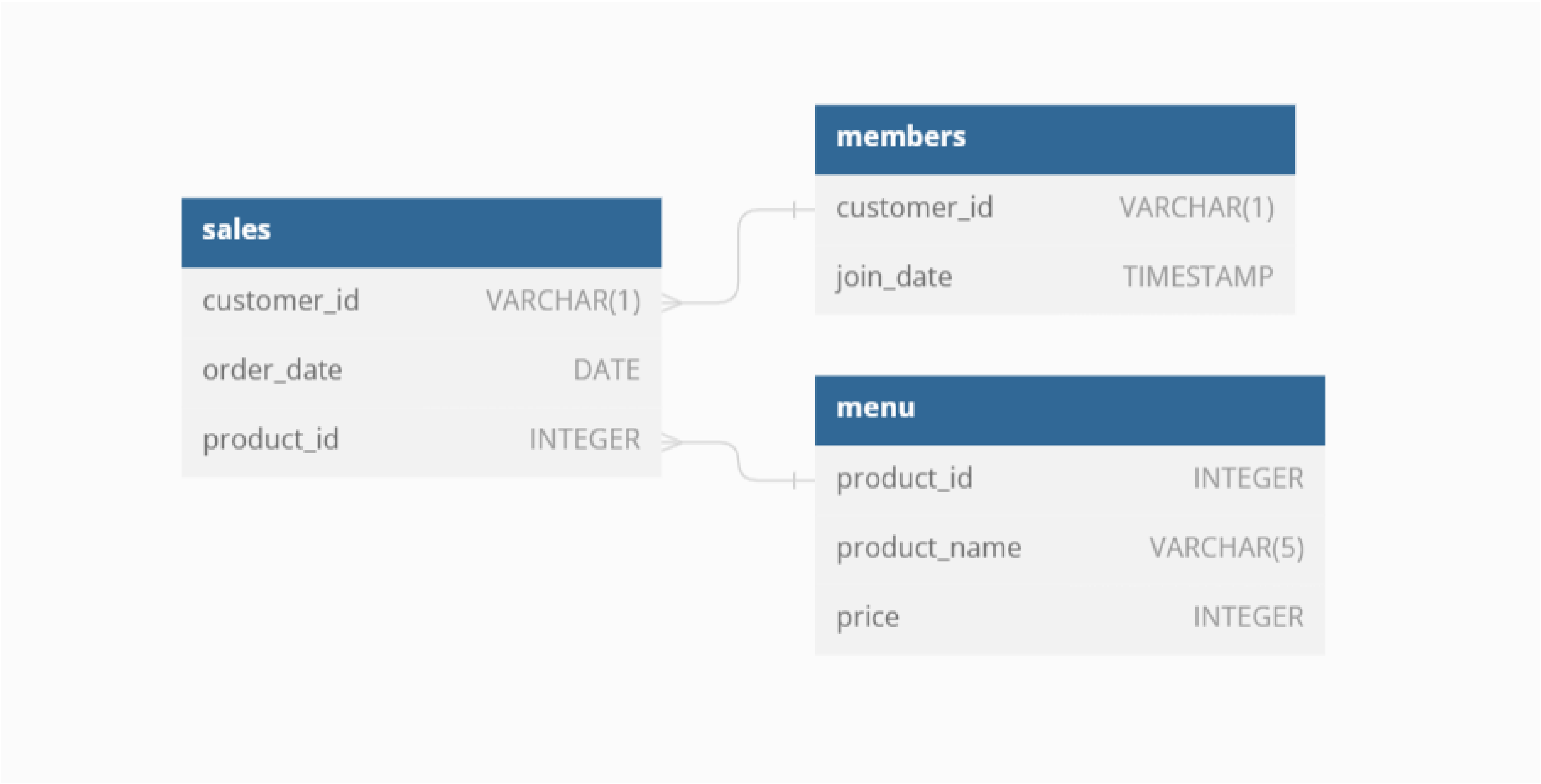
PROBLEM STATEMENT:

Danny wants to use the data to answer a few simple questions about his customers, especially about their visiting patterns, how much money they've spent and also which menu items are their favourite. Having this deeper connection with his customers will help him deliver a better and more personalised experience for his loyal customers.

He plans on using these insights to help him decide whether he should expand the existing customer loyalty program - additionally, he needs help to generate some basic datasets so his team can easily inspect the data without needing to use SQL.

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

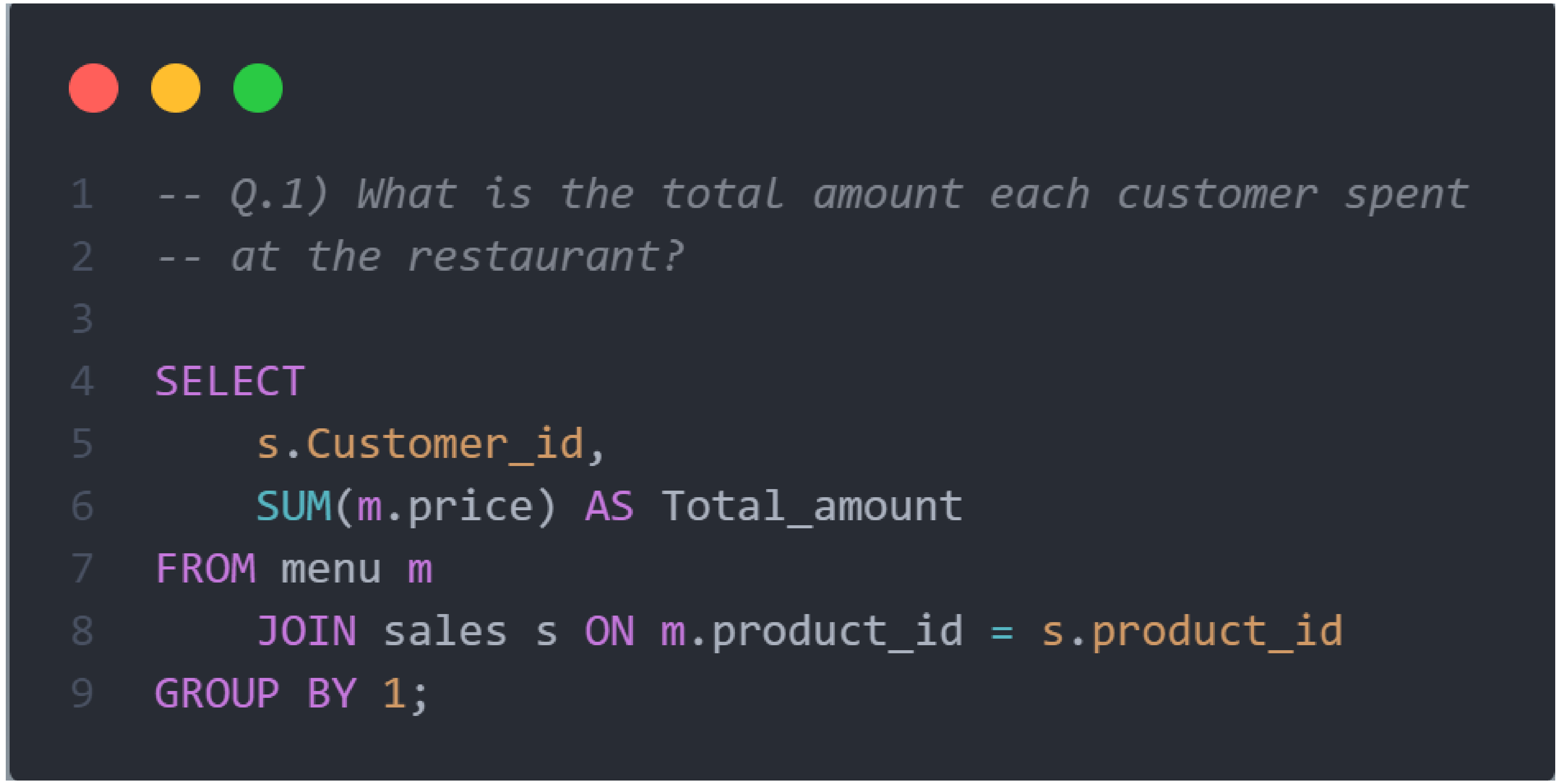
ENTITY RELATIONSHIP DIAGRAM:



sales	
customer_id	VARCHAR(1)
order_date	DATE
product_id	INTEGER

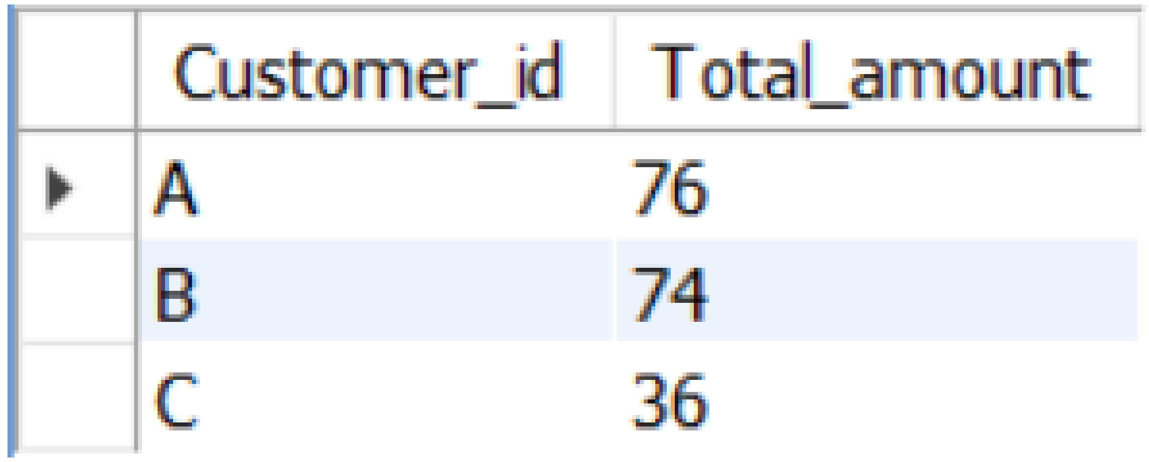
members	
customer_id	VARCHAR(1)
join_date	TIMESTAMP

menu	
product_id	INTEGER
product_name	VARCHAR(5)
price	INTEGER

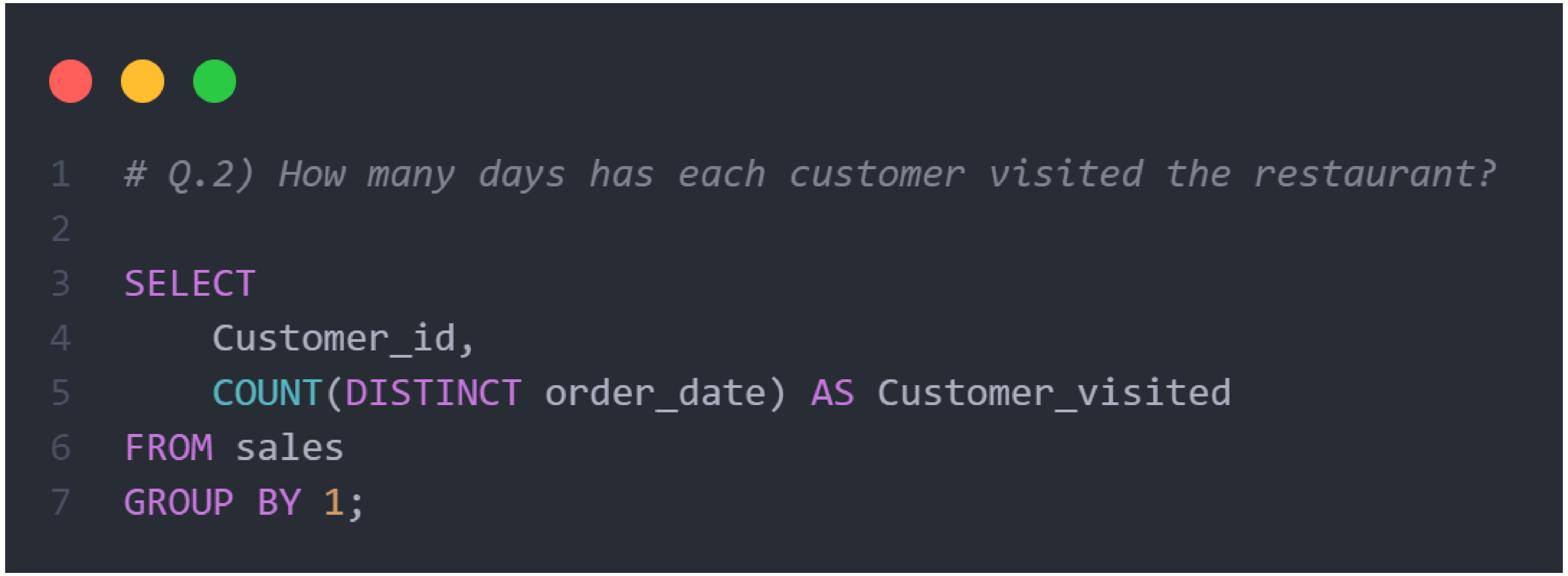


```sql
-- Q.1) What is the total amount each customer spent
-- at the restaurant?

SELECT
    s.Customer_id,
    SUM(m.price) AS Total_amount
FROM menu m
    JOIN sales s ON m.product_id = s.product_id
GROUP BY 1;
```

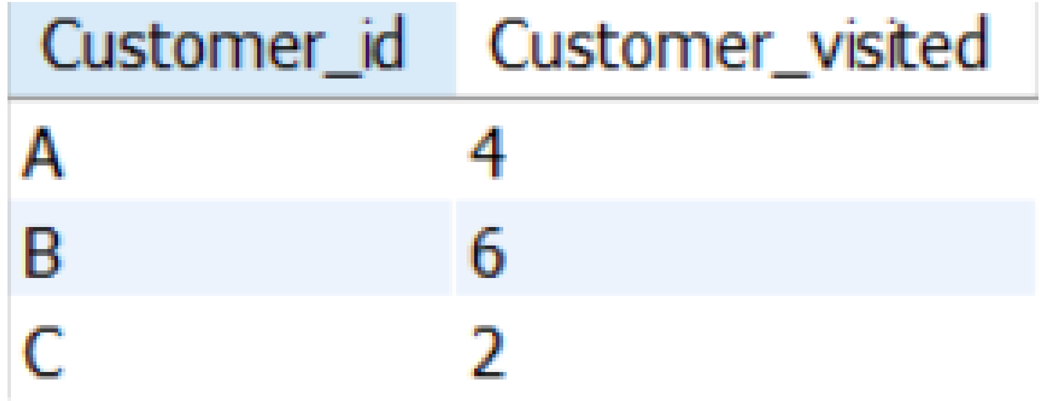


Customer_id	Total_amount
A	76
B	74
C	36

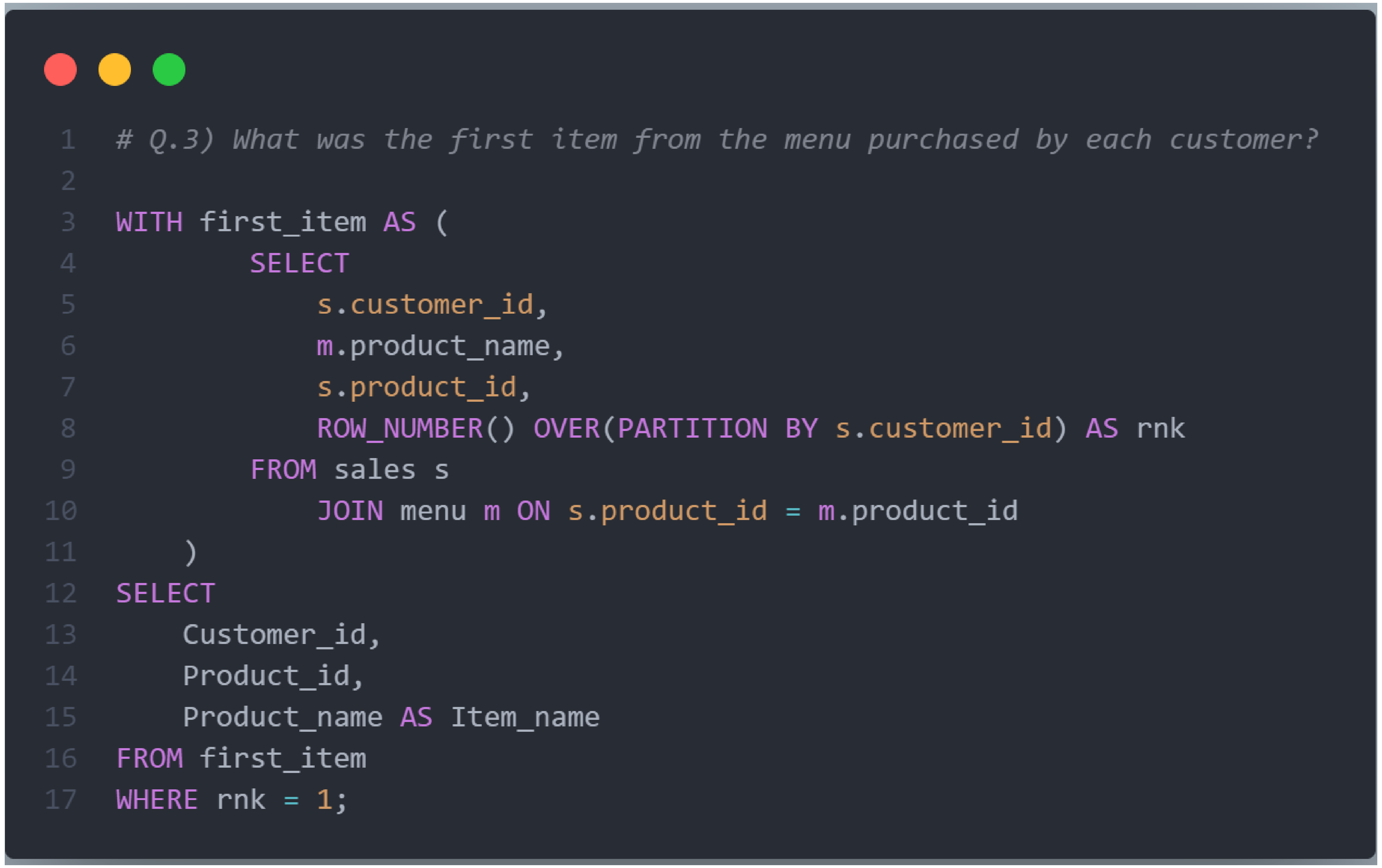


```sql
# Q.2) How many days has each customer visited the restaurant?

SELECT
    Customer_id,
    COUNT(DISTINCT order_date) AS Customer_visited
FROM sales
GROUP BY 1;
```

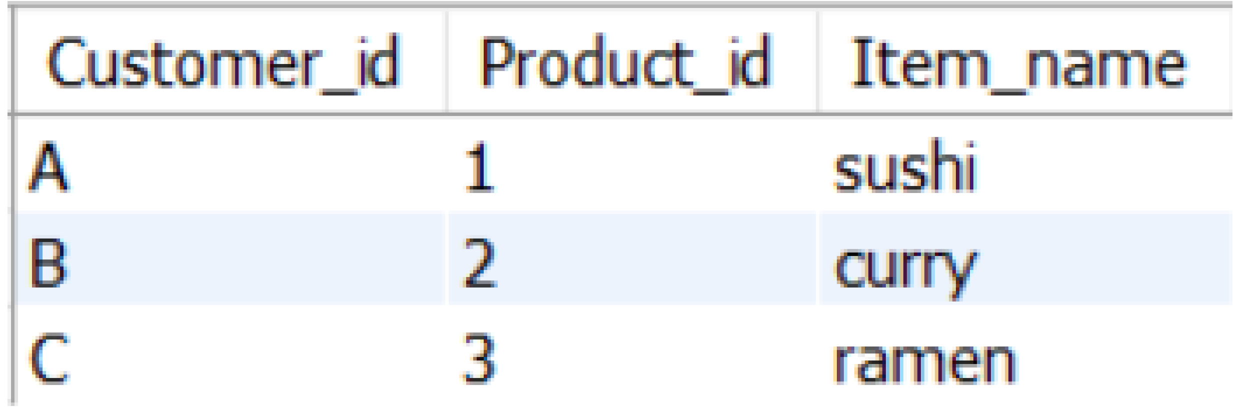


Customer_id	Customer_visited
A	4
B	6
C	2

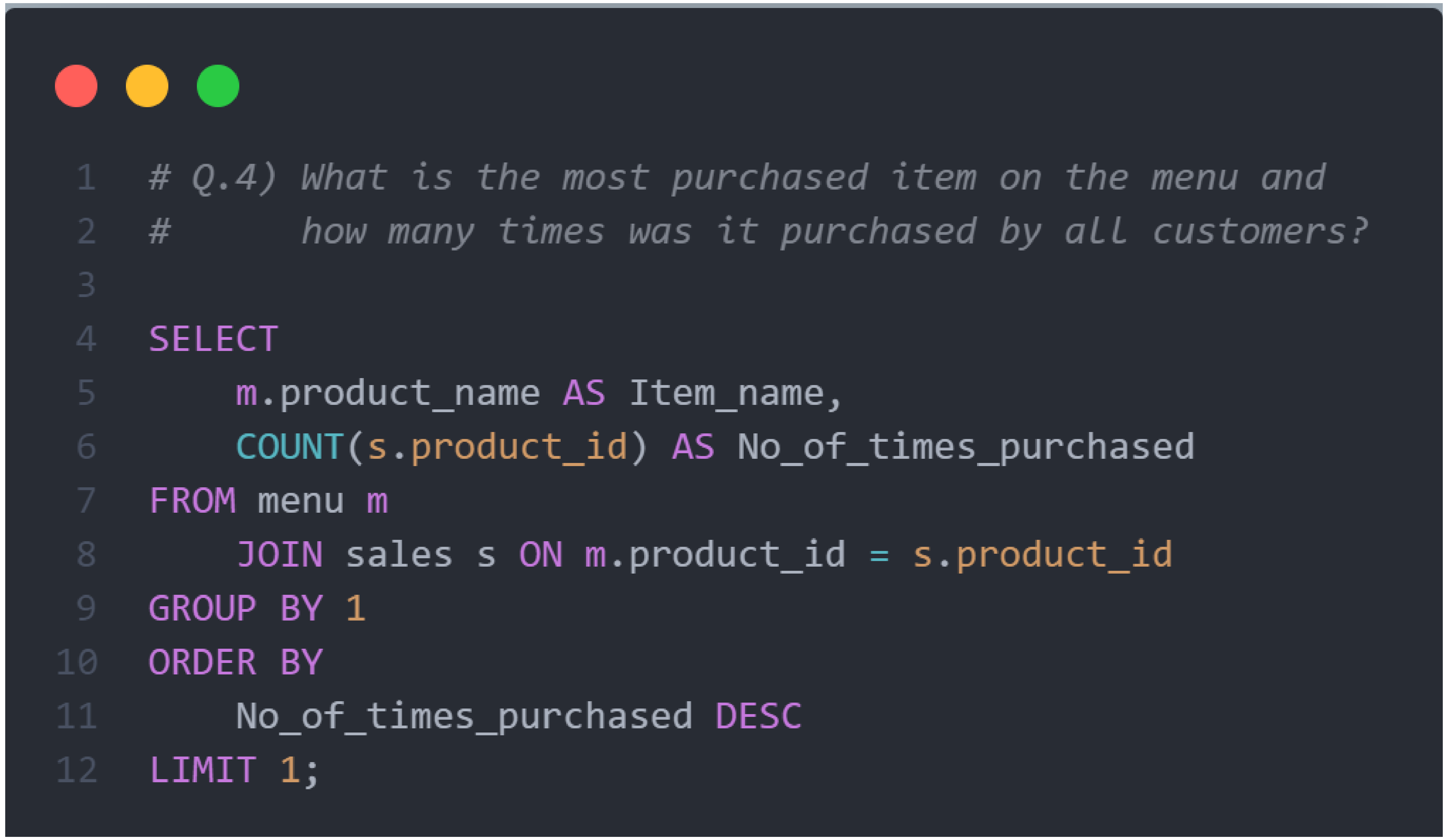


```
# Q.3) What was the first item from the menu purchased by each customer?

WITH first_item AS (
        SELECT
            s.customer_id,
            m.product_name,
            s.product_id,
            ROW_NUMBER() OVER(PARTITION BY s.customer_id) AS rnk
        FROM sales s
            JOIN menu m ON s.product_id = m.product_id
    )
SELECT
    Customer_id,
    Product_id,
    Product_name AS Item_name
FROM first_item
WHERE rnk = 1;
```

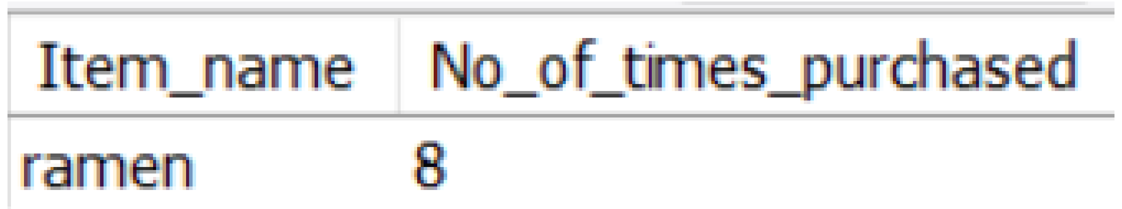


Customer_id	Product_id	Item_name
A	1	sushi
B	2	curry
C	3	ramen

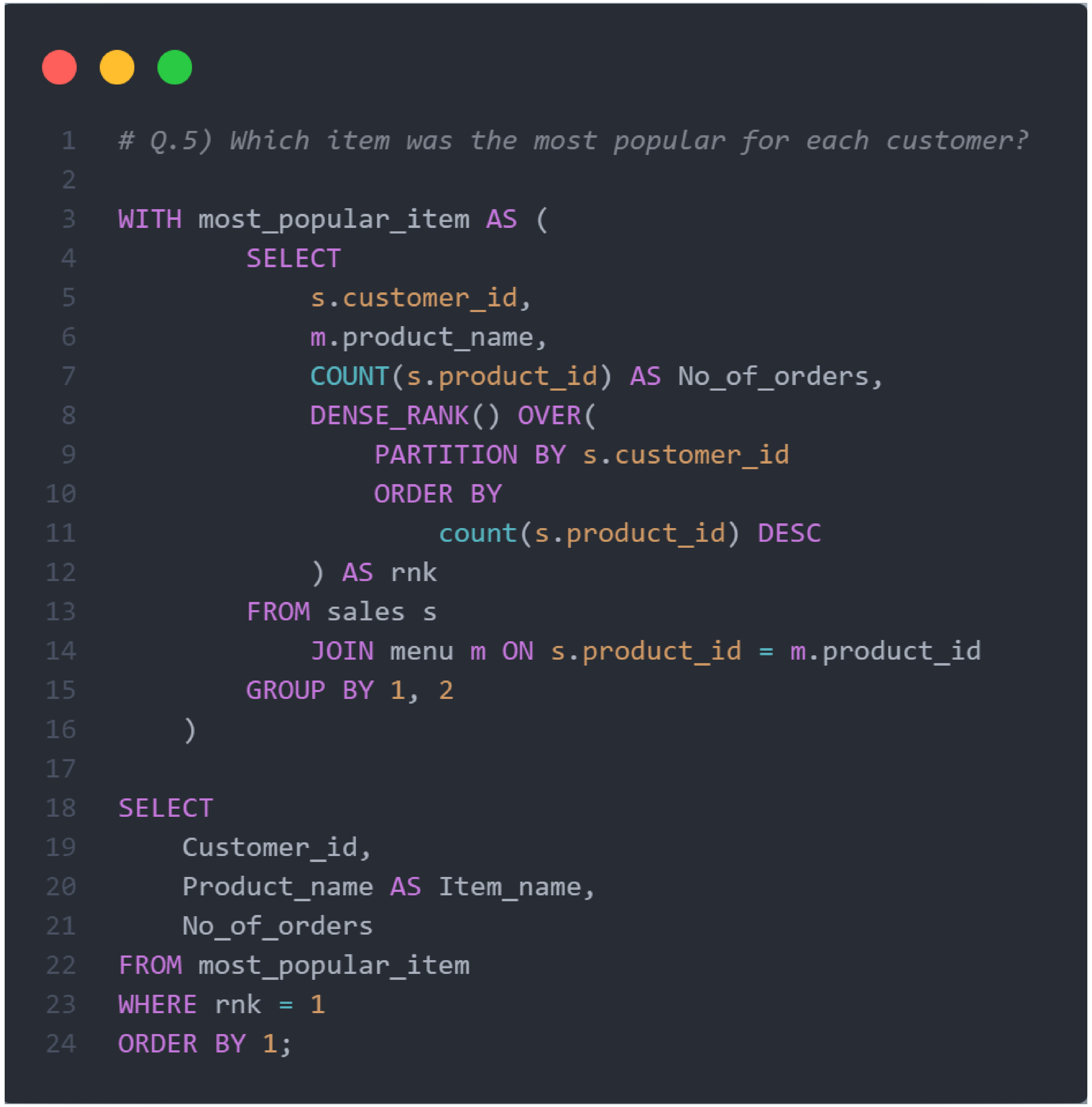


```sql
# Q.4) What is the most purchased item on the menu and
#      how many times was it purchased by all customers?

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



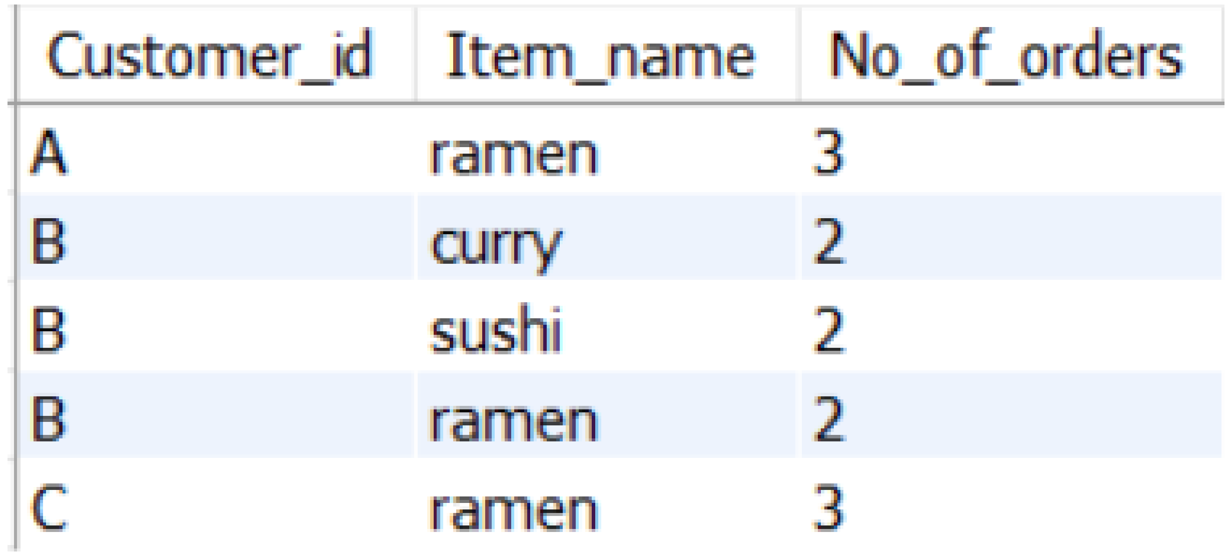
Item_name	No_of_times_purchased
ramen	8



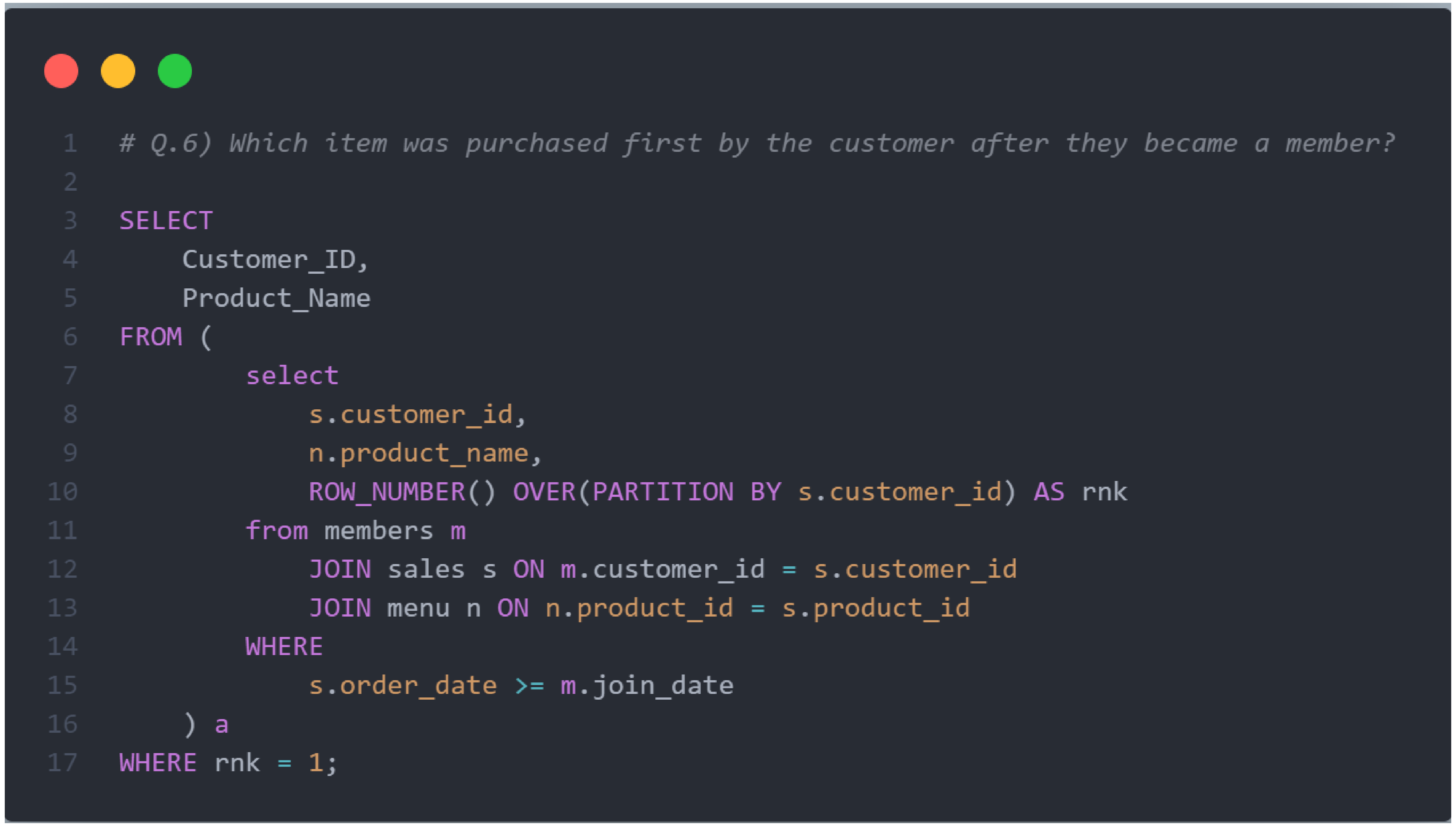
```sql
# Q.5) Which item was the most popular for each customer?

WITH most_popular_item AS (
        SELECT
            s.customer_id,
            m.product_name,
            COUNT(s.product_id) AS No_of_orders,
            DENSE_RANK() OVER(
                PARTITION BY s.customer_id
                ORDER BY
                    count(s.product_id) DESC
            ) AS rnk
        FROM sales s
            JOIN menu m ON s.product_id = m.product_id
        GROUP BY 1, 2
    )

SELECT
    Customer_id,
    Product_name AS Item_name,
    No_of_orders
FROM most_popular_item
WHERE rnk = 1
ORDER BY 1;
```

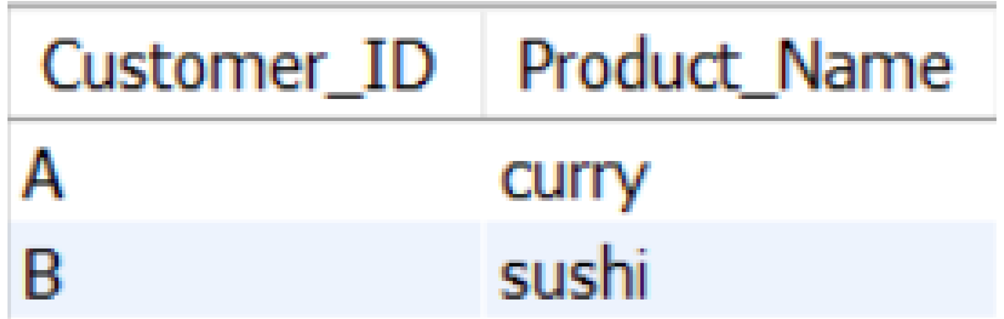


Customer_id	Item_name	No_of_orders
A	ramen	3
B	curry	2
B	sushi	2
B	ramen	2
C	ramen	3

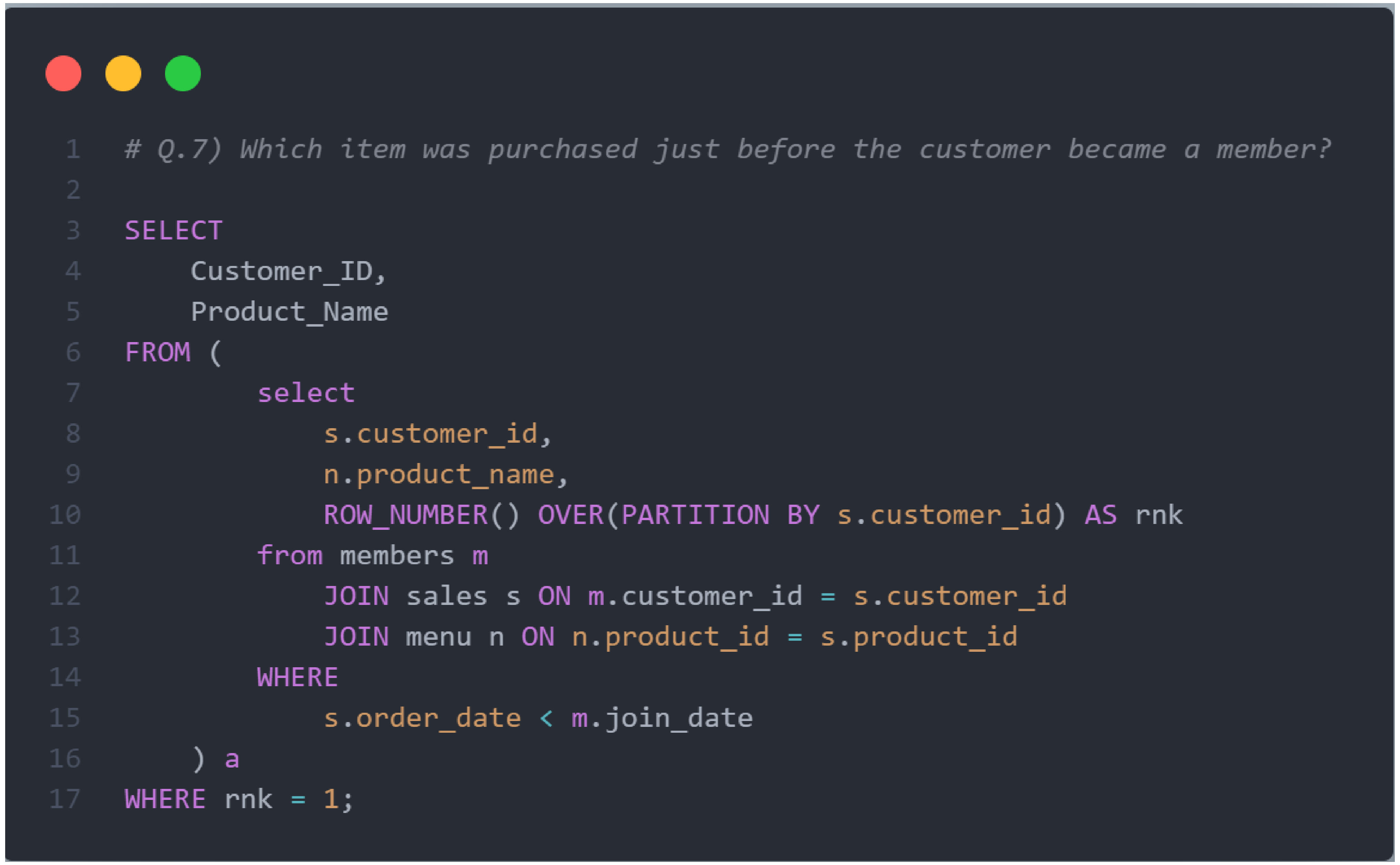


```
# Q.6) Which item was purchased first by the customer after they became a member?

SELECT
    Customer_ID,
    Product_Name
FROM (
        select
            s.customer_id,
            n.product_name,
            ROW_NUMBER() OVER(PARTITION BY s.customer_id) AS rnk
        from members m
            JOIN sales s ON m.customer_id = s.customer_id
            JOIN menu n ON n.product_id = s.product_id
        WHERE
            s.order_date >= m.join_date
    ) a
WHERE rnk = 1;
```

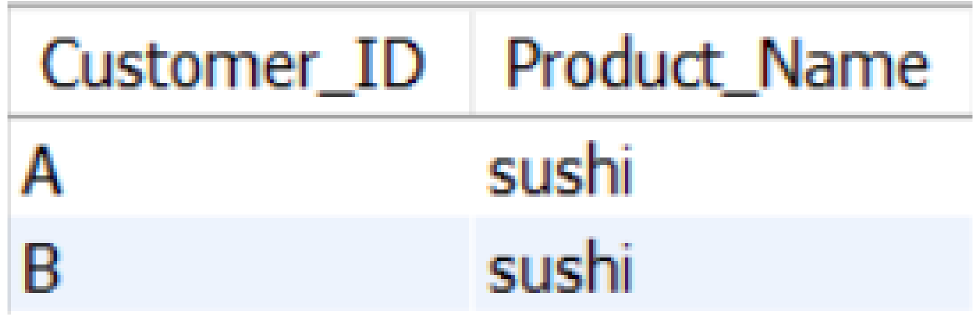


Customer_ID	Product_Name
A	curry
B	sushi

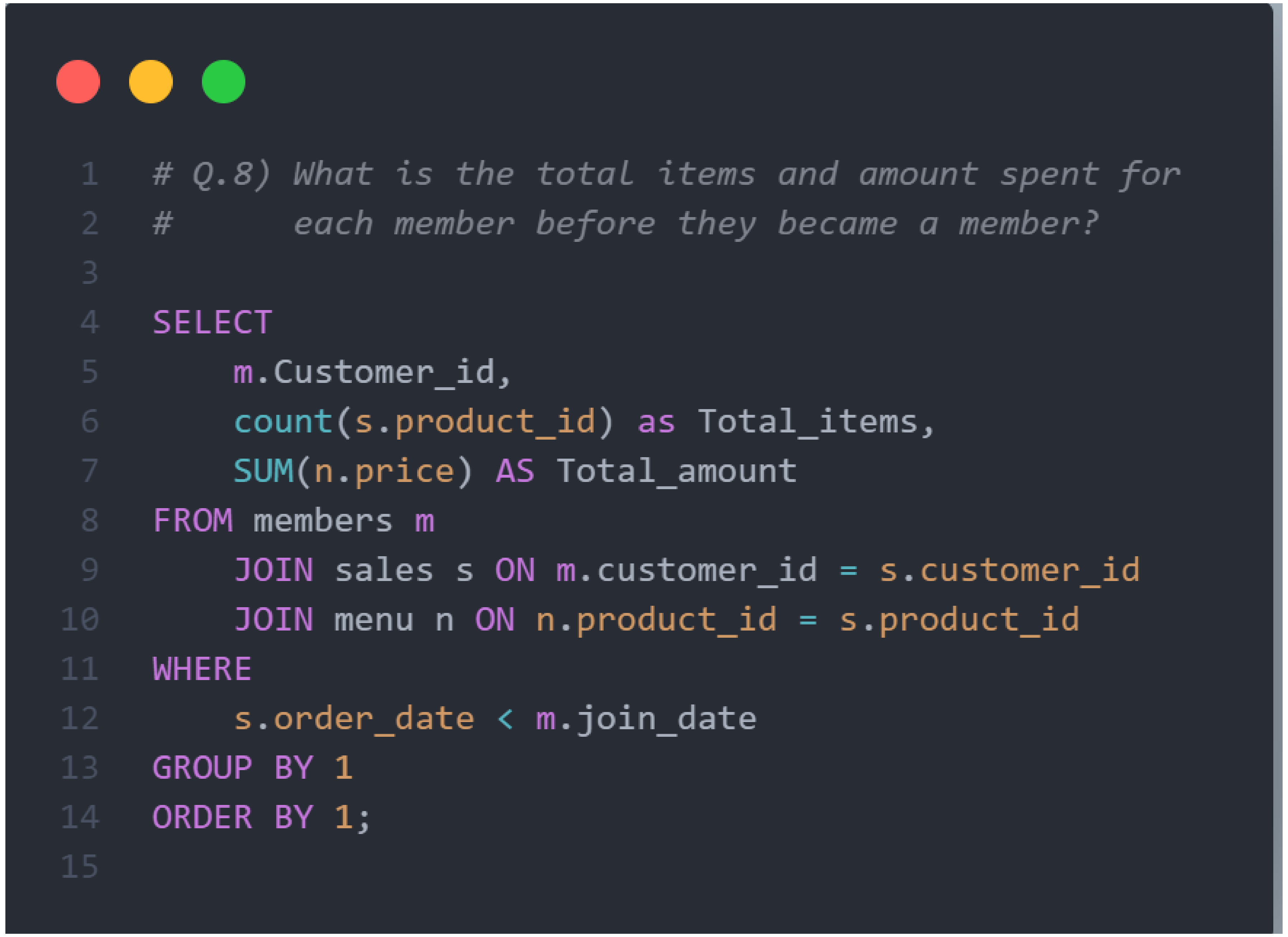


```
# Q.7) Which item was purchased just before the customer became a member?

SELECT
    Customer_ID,
    Product_Name
FROM (
        select
            s.customer_id,
            n.product_name,
            ROW_NUMBER() OVER(PARTITION BY s.customer_id) AS rnk
        from members m
            JOIN sales s ON m.customer_id = s.customer_id
            JOIN menu n ON n.product_id = s.product_id
        WHERE
            s.order_date < m.join_date
    ) a
WHERE rnk = 1;
```

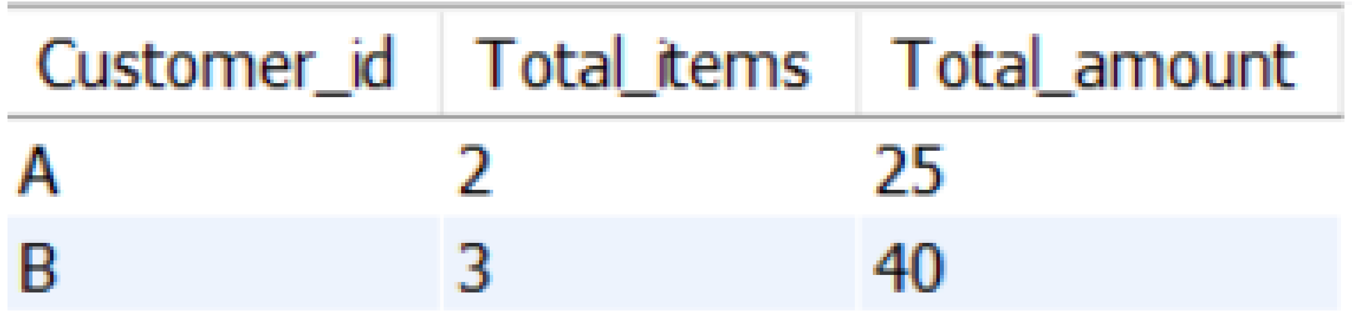


Customer_ID	Product_Name
A	sushi
B	sushi

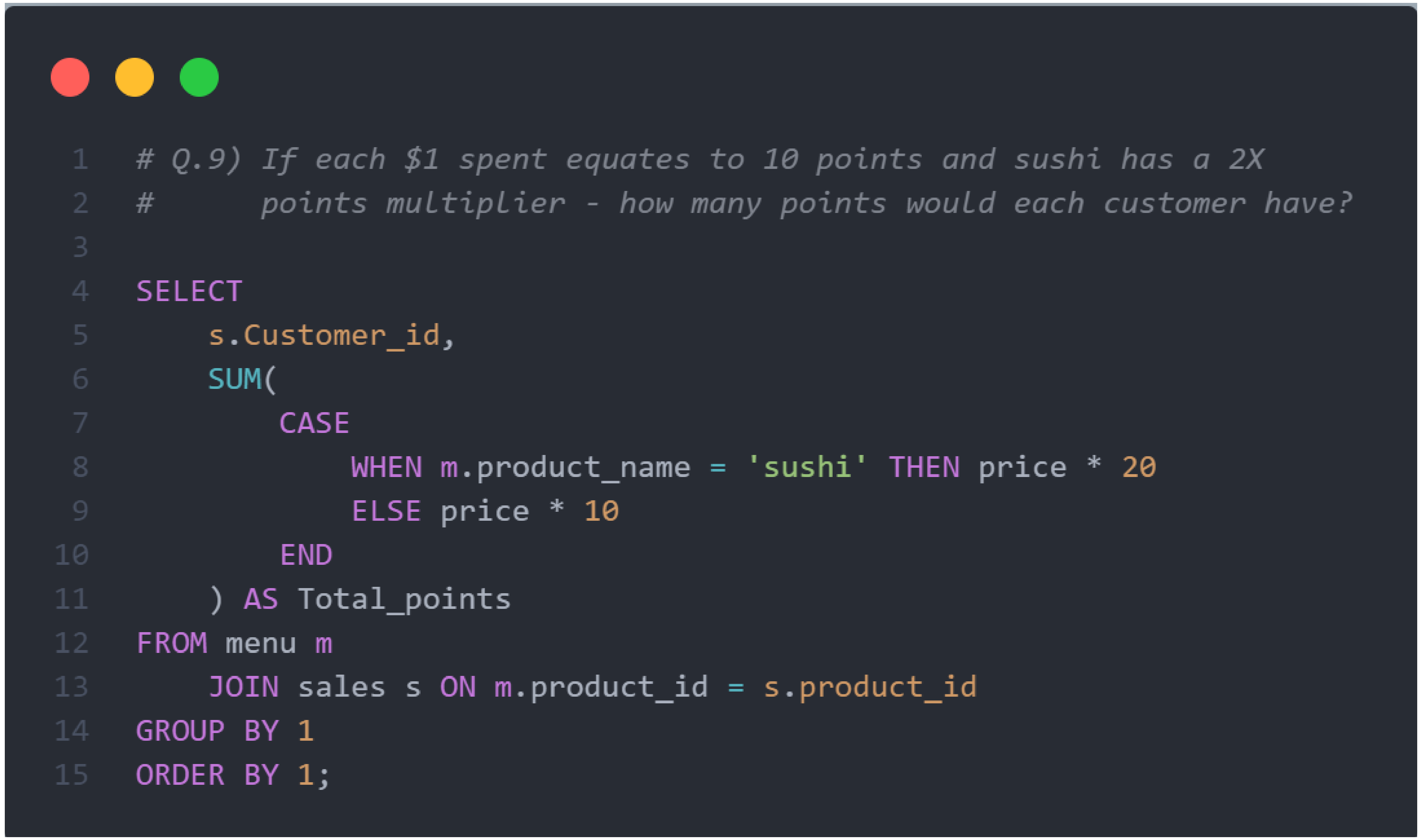


```sql
# Q.8) What is the total items and amount spent for
#      each member before they became a member?

SELECT
    m.Customer_id,
    count(s.product_id) as Total_items,
    SUM(n.price) AS Total_amount
FROM members m
    JOIN sales s ON m.customer_id = s.customer_id
    JOIN menu n ON n.product_id = s.product_id
WHERE
    s.order_date < m.join_date
GROUP BY 1
ORDER BY 1;
```

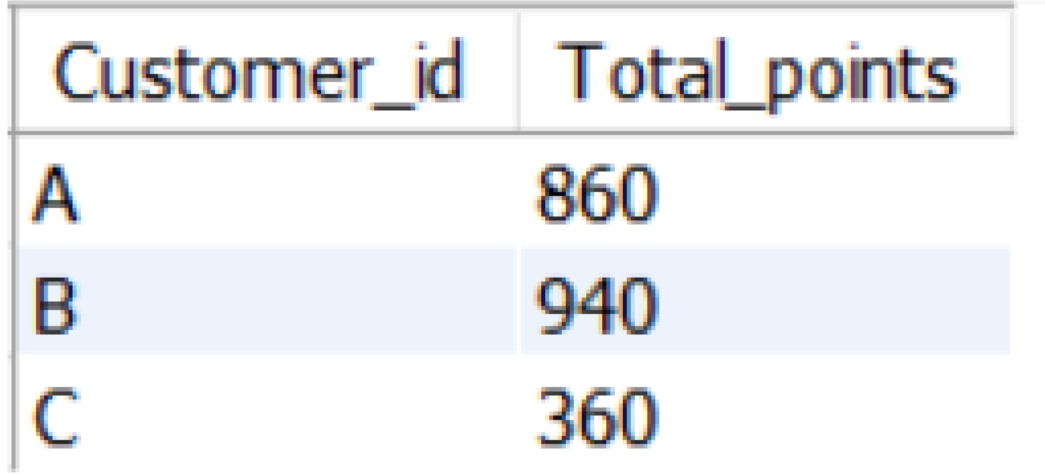


Customer_id	Total_items	Total_amount
A	2	25
B	3	40

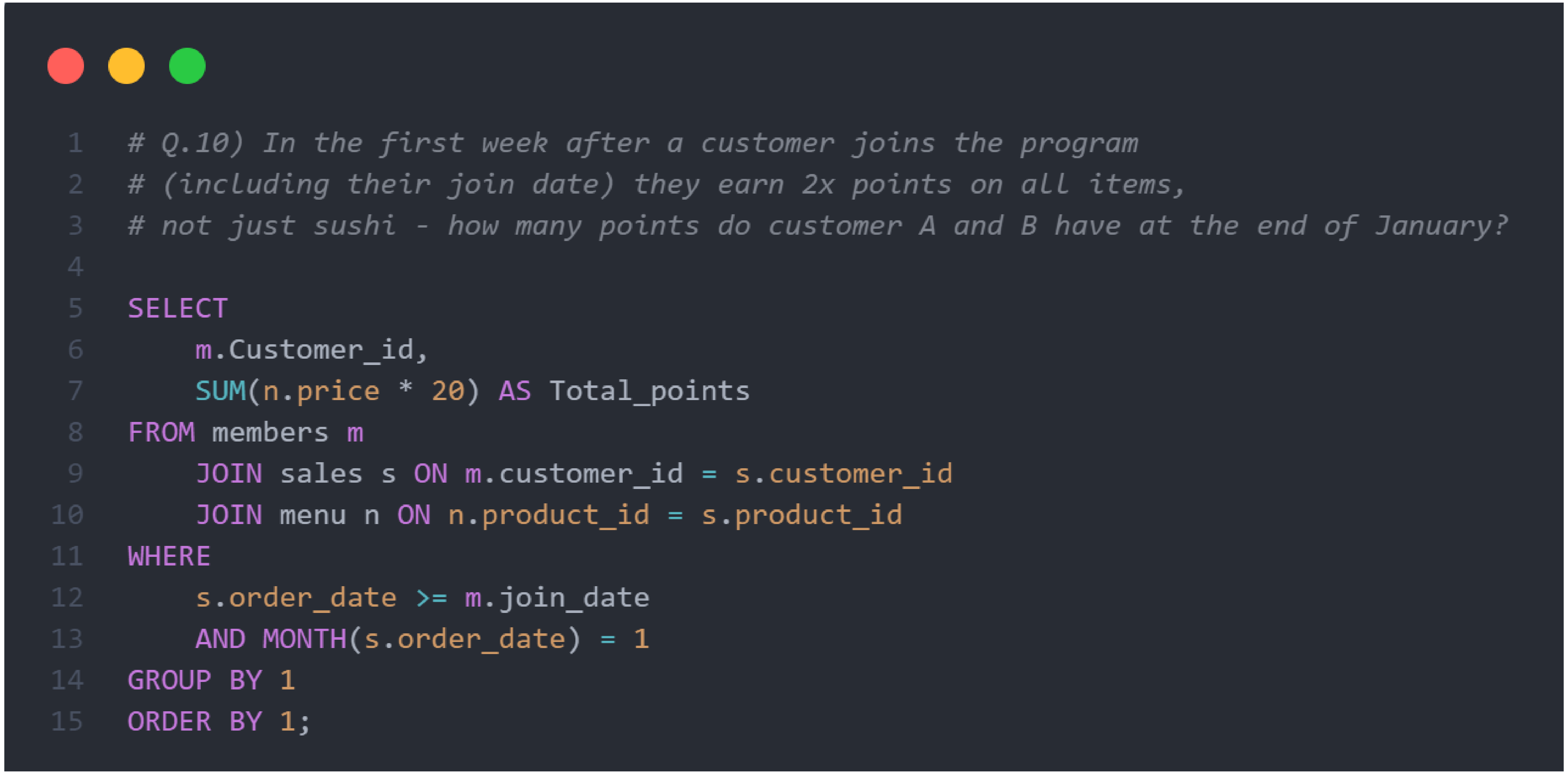


```sql
# Q.9) If each $1 spent equates to 10 points and sushi has a 2X
#      points multiplier - how many points would each customer have?

SELECT
    s.Customer_id,
    SUM(
        CASE
            WHEN m.product_name = 'sushi' THEN price * 20
            ELSE price * 10
        END
    ) AS Total_points
FROM menu m
    JOIN sales s ON m.product_id = s.product_id
GROUP BY 1
ORDER BY 1;
```

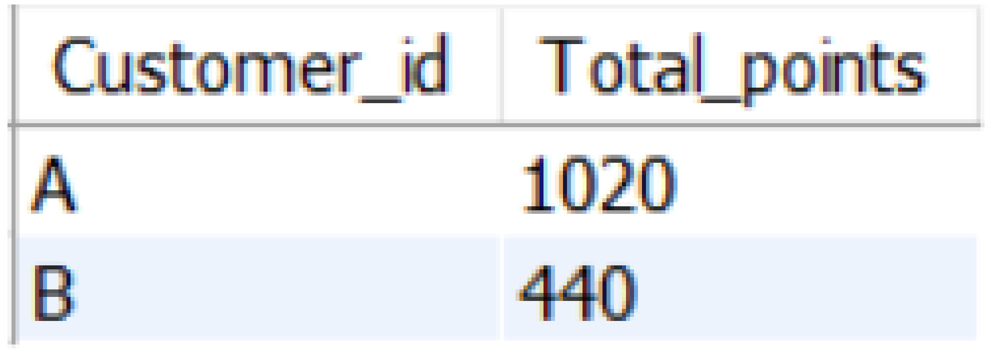


Customer_id	Total_points
A	860
B	940
C	360



```sql
# Q.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
    m.Customer_id,
    SUM(n.price * 20) AS Total_points
FROM members m
    JOIN sales s ON m.customer_id = s.customer_id
    JOIN menu n ON n.product_id = s.product_id
WHERE
    s.order_date >= m.join_date
    AND MONTH(s.order_date) = 1
GROUP BY 1
ORDER BY 1;
```



Customer_id	Total_points
A	1020
B	440

THANK YOU!