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

```sql
SELECT s.customer_id,
    SUM(m.price) AS total_amt_spent
FROM sales s LEFT JOIN menu m
  ON s.product_id=m.product_id
GROUP BY customer_id;
```

Answer

customer_id	total_amt_spent
A	76
B	74
C	36

2. How many days has each customer visited the restaurant?
```sql
 SELECT customer_id,
        COUNT(DISTINCT(order_date)) AS total_days_visited
 FROM sales
 GROUP BY customer_id;
```

Answer

customer_id	total_days_visited
A	4
B	6
C	2

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

```sql
WITH first_item
AS
(
SELECT s.customer_id,
       m.product_name,
       ROW_NUMBER() OVER (PARTITION BY customer_id ORDER BY order_date ASC) AS rn
FROM sales s LEFT JOIN menu m
     ON s.product_id=m.product_id
)

SELECT customer_id,
       product_name
FROM first_item
WHERE rn=1;
```

Answer

customer_id	product_name
A	sushi
B	curry
C	ramen

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

```sql
SELECT TOP 1
            s.product_id,
            m.product_name AS most_ordered_item,
            COUNT(s.product_id) AS no_of_times_ordered
FROM sales s LEFT JOIN menu m
     ON s.product_id=m.product_id
GROUP BY s.product_id, product_name
ORDER BY no_of_times_ordered DESC;
```

Answer

product_id	most_ordered_item	no_of_times_ordered
3	ramen	8

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

```sql
WITH most_popular_item
AS
(
SELECT
    s.customer_id,
    m.product_name,
    COUNT(s.product_id) AS no_of_times_ordered,
    RANK() OVER (PARTITION BY s.customer_id ORDER BY
COUNT(s.product_id) DESC) AS rk
  FROM sales s LEFT JOIN menu m
      ON s.product_id=m.product_id
  GROUP BY s.customer_id, s.product_id, m.product_name
  )
  SELECT
    customer_id,
    product_name AS popular_item
  FROM most_popular_item
  WHERE rk=1;
```

customer_id	popular_item
A	ramen
B	sushi
B	curry
B	ramen
C	ramen

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

```sql
WITH cte
AS
(
SELECT
    s.customer_id,
    s.order_date,
    s.product_id,
    me.join_date,
    ROW_NUMBER() OVER (PARTITION BY s.customer_id ORDER BY s.order_date ASC) AS rn
FROM sales s JOIN members me
    ON s.customer_id=me.customer_id
WHERE s.order_date>me.join_date
)
SELECT
    c.customer_id,
    c.product_id,
    m.product_name
FROM cte c LEFT JOIN menu m
  ON c.product_id=m.product_id
WHERE rn=1
```

Answer

customer_id	product_id	product_name
A	3	ramen
B	1	sushi

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

```sql
WITH cte1
AS
(
SELECT
    s.customer_id,
    s.order_date,
    s.product_id,
    me.join_date,
    DENSE_RANK() OVER (PARTITION BY s.customer_id ORDER BY
s.order_date DESC) AS rn
FROM sales s JOIN members me
    ON s.customer_id=me.customer_id
WHERE s.order_date<me.join_date
)
SELECT
    c.customer_id,
    c.product_id,
    m.product_name
FROM cte1 c LEFT JOIN menu m
    ON c.product_id=m.product_id
WHERE rn=1
```

customer_id	product_id	product_name
A	1	sushi
B	1	sushi

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

```sql
SELECT
    s.customer_id,
    COUNT(s.product_id) AS total_items_before_membership,
    SUM(m.price) AS total_amt_before_membership
FROM sales s JOIN members me
    ON s.customer_id=me.customer_id
JOIN menu m
    ON s.product_id=m.product_id
WHERE s.order_date<me.join_date
GROUP BY s.customer_id;
```

Answer

customer_id	total_items_before_membership	total_amt_before_membership
A	2	25
B	3	40

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

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

Answer

customer_id	total_amt_spent	total_points
A	76	860
B	74	940
C	36	360

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?

```sql
SELECT
    s.customer_id,
    SUM(CASE WHEN s.order_date  BETWEEN me.join_date AND dateadd(day,+6,me.join_date)
    then m.price*20
    ELSE m.price*10 END) AS total_points
FROM sales s JOIN members me
    ON s.customer_id=me.customer_id
JOIN menu m
    ON s.product_id=m.product_id
WHERE s.order_date BETWEEN '2021-01-01'  AND '2021-01-31'
GROUP BY s.customer_id;
```

Answer

customer_id	total_points
A	1270
B	720

```
Bonus Questions
1)Join all three tables
 SELECT
     s.customer_id,
     s.order_date,
     m.product_name,
     m.price,
     CASE WHEN s.order_date>=me.join_date then 'Y'
     ELSE 'N' END AS member
 FROM sales s JOIN members me
     ON s.customer_id=me.customer_id
 JOIN menu m
     ON s.product_id=m.product_id
```

```sql
2)Ranking all the things
 SELECT
    A.customer_id,
    A.order_date,
    A.product_name,
    A.price,
    CASE WHEN member='N' THEN 'null'
    ELSE RANK () OVER (PARTITION BY A.customer_id ORDER BY
A.order_date ASC) END AS ranking
 FROM
 (SELECT
    s.customer_id,
    s.order_date,
    m.product_name,
    m.price,
    CASE WHEN s.order_date>=me.join_date then 'Y'
    ELSE 'N' END AS member
 FROM sales s JOIN members me
    ON s.customer_id=me.customer_id
 JOIN menu m
    ON s.product_id=m.product_id) A
```