Case Study:

Danny’s Diner

Leveraging Customer Data for Business Insights and Loyalty Program Enhancement

Case Study: Danny's Diner

Introduction:

Danny Ma, a passionate lover of Japanese food, opened a small restaurant called Danny's Diner at the beginning of 2021. The restaurant specializes in three favourite Japanese dishes: sushi, curry, and ramen. As the business has been operating for a few months, Danny wants to leverage the captured data to gain insights about his customers. He aims to understand their visiting patterns, expenditure, and preferred menu items. By establishing a deeper connection with his customers, Danny plans to enhance their experience through personalized offerings. Additionally, he seeks assistance in expanding the existing customer loyalty program and generating basic datasets for easy data inspection.

Problem Statement:

Danny needs to utilize the available data to answer several fundamental questions about his customers. These questions include analysing visiting patterns, customer expenditure, favourite menu items, and evaluating the potential expansion of the loyalty program. Furthermore, Danny requires assistance in creating basic datasets to simplify data analysis for his team, without relying on complex SQL queries.

Methodology:

To achieve our objective, we will perform data analysis on the Danny's Diner database using SQL queries. We will start by answering some easy questions and gradually move towards more complex questions that require advanced SQL skills.

Datasets:

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

- sales
- menu
- members

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

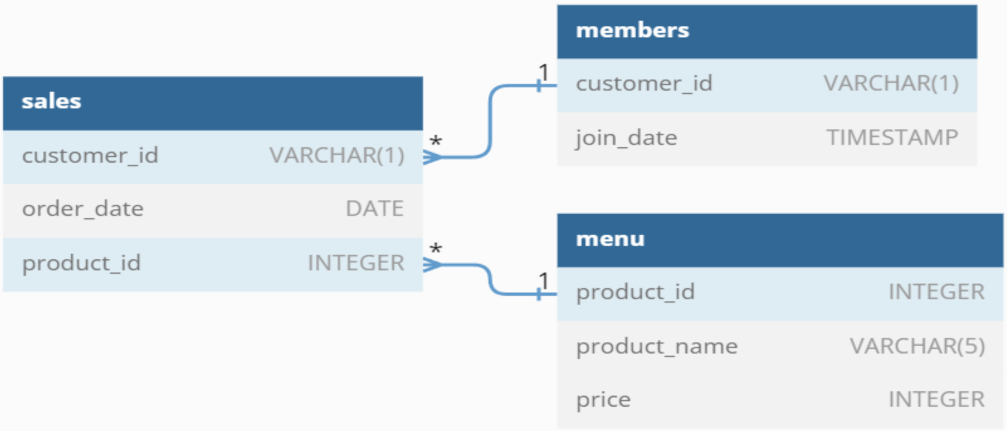


Table 1: sales

The sales table captures all customer_id level purchases with an corresponding order_date and product_id information for when and what menu items were ordered.

Table 2: menu

The menu table maps the product_id to the actual product_name and price of each menu item.

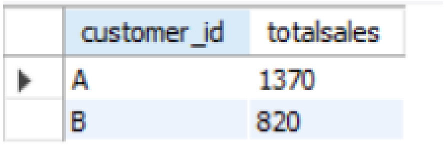
Table 3: members

The final members table captures the join_date when a customer_id joined the beta version of the Danny's Diner loyalty program.

Case Study Questions

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

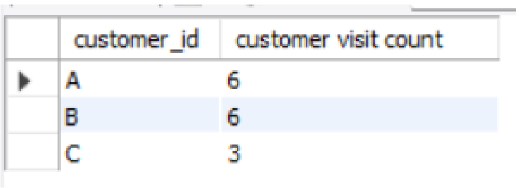
```
select customer_id, sum(price) as total_amount_spent
from sales
inner join menu
on sales.product_id=menu.product_id
group by customer_id;
```



customer_id	totalsales
A	1370
B	820

2. How many days has each customer visited the restaurant?

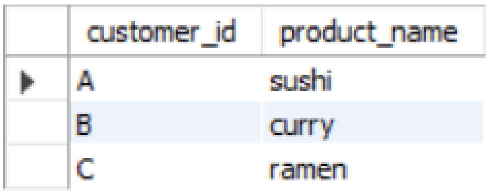
```
select customer_id, count(*) as 'customer visit count'
from sales
group by customer_id;
```



customer_id	customer visit count
A	6
B	6
C	3

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

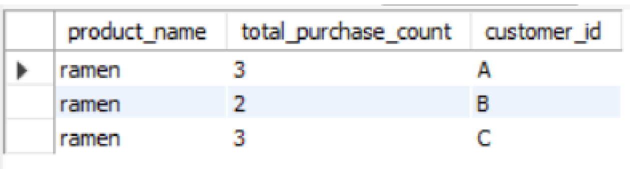
```
select customer_id,product_name from
(select customer_id,product_name,
row_number() over (partition by sales.customer_id order by order_date)
as row_no
from sales
inner join menu
on sales.product_id=menu.product_id) as x
where row_no<2;
```



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?

```
select m.product_name, count(*) AS total_purchase_count,
s.customer_id
from sales s
Join menu m on s.product_id = m.product_id
where s.product_id IN (
                                select product_id
                                from sales
                                group by product_id
 having count(*) = (
                           select max(purchase_count)
                           from (
                                   select count(*) AS purchase_count
                                   from sales
                                   Group by product_id
                                    ) t
                                  )
                                 )
group by m.product_id, m.product_name, s.customer_id;
```



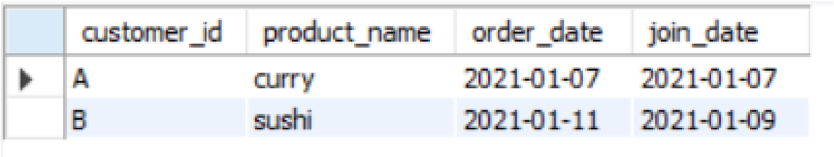
product_name	total_purchase_count	customer_id
ramen	3	A
ramen	2	B
ramen	3	C

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

```
select customer_id,product_name, total_orders
from
        (select customer_id,product_name,count(*) as total_orders,
         row_number() over(partition by customer_id order by count(*)
        desc ) as row_no
        from sales
        inner join menu
        on sales.product_id=menu.product_id
        group by customer_id,product_name) as x
where row_no=1;
```

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

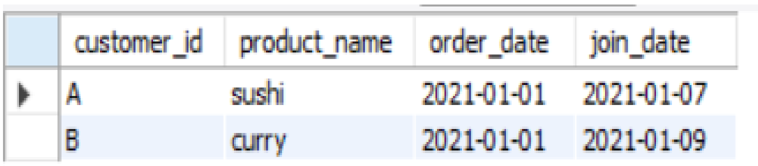
```
select customer_id,product_name,order_date,join_date
from
      (select mem.customer_id,m.product_name,order_date,join_date,
       row_number() over(partition by mem.customer_id) as row_no
       from sales as s
       inner join members as mem on mem.customer_id=s.customer_id
       inner join menu as m on m.product_id=s.product_id
       where join_date<=order_date) as x
 where row_no=1;
```



	customer_id	product_name	order_date	join_date
▶	A	curry	2021-01-07	2021-01-07
	B	sushi	2021-01-11	2021-01-09

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

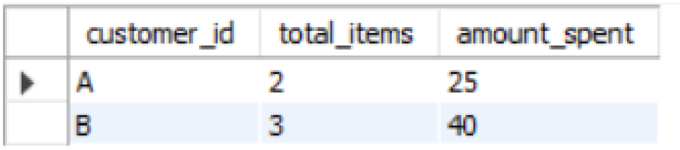
```
select customer_id,product_name,order_date,join_date
from
(select mem.customer_id,m.product_name,order_date,join_date,
row_number() over(partition by mem.customer_id order by order_date)
as row_no
from sales as s
inner join members as mem on mem.customer_id=s.customer_id
inner join menu as m on m.product_id=s.product_id
where order_date<join_date) as x
where row_no=1;
```



	customer_id	product_name	order_date	join_date
▶	A	sushi	2021-01-01	2021-01-07
	B	curry	2021-01-01	2021-01-09

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

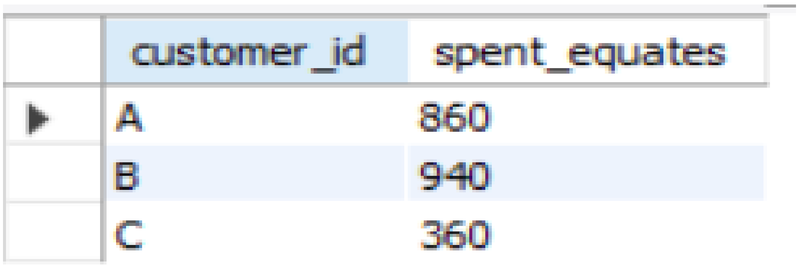
```
select members.customer_id, count(*) as total_items, sum(menu.price)
as amount_spent
from sales
inner join menu on menu.product_id=sales.product_id
inner join members on members.customer_id=sales.customer_id
where join_date>order_date
group by members.customer_id
order by members.customer_id;
```



	customer_id	total_items	amount_spent
▶	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?

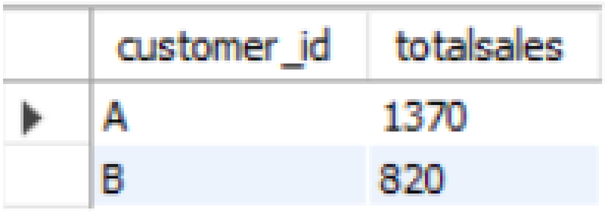
```
select s.customer_id,
sum(case
        when m.product_name='sushi' then m.price*20
        else m.price*10
         end) as spent_equates
from sales s
inner join menu m
on s.product_id=m.product_id
group by s.customer_id;
```



	customer_id	spent_equates
▶	A	860
	B	940
	C	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?

```
select sales.customer_id,
sum (case
        when order_date between join_date and adddate (join_date, interval
S6 day) then menu.price*20
        when menu.product_name='sushi' then menu.price*20
        else menu.price*10
        end) as totalsales
from  sales
inner join menu on menu.product_id=sales.product_id
inner join members on members.customer_id=sales.customer_id
where monthname(sales.order_date)='January'
group by sales.customer_id
order by sales.customer_id;
```



	customer_id	totalsales
▶	A	1370
	B	820

Conclusion:

In this case study, we looked at Danny's Diner, a restaurant that serves Japanese food. We wanted to find ways to help Danny make better decisions for his business using the information he has about his customers.

By analysing the data, we discovered important things about the customers, such as how much money they spent at the restaurant, how often they visited, and their favourite menu items. We also learned about their behaviour before and after joining the loyalty program.

These insights can help Danny improve the customer experience by personalizing their visits and deciding whether to expand the loyalty program. By using data to understand his customers better, Danny can build stronger relationships and make his restaurant more successful.

This case study shows how important it is for businesses to use data to learn about their customers and make smart choices. By understanding what customers want, businesses can make them happier and more likely to come back.

Thank you for watching!!!