Special

Danny's Diner

SQL

Case Study #1

Presentation by
Picnic Rautaray

VISIT PROFILE

Intro

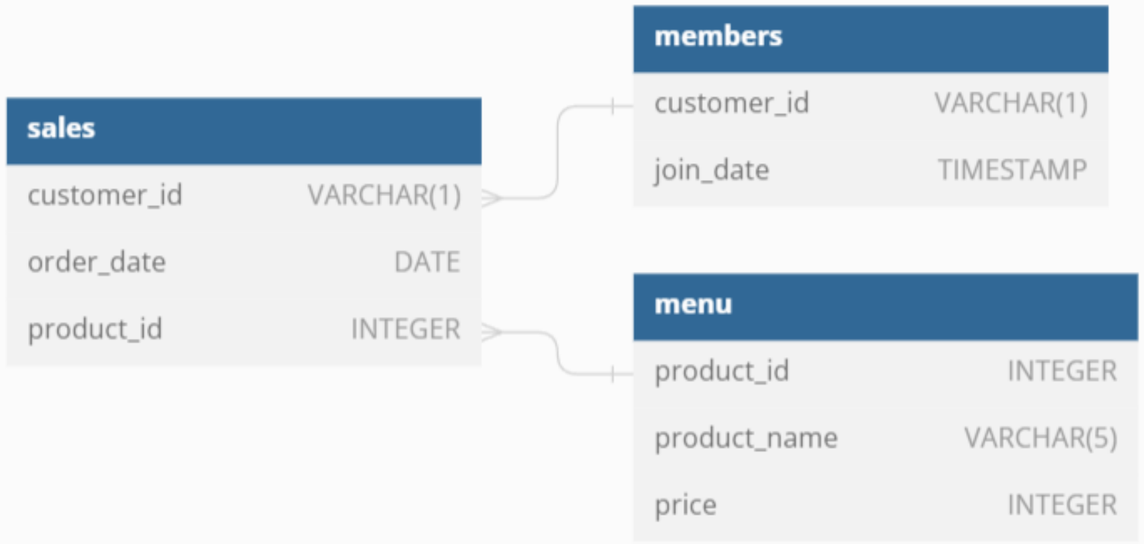
Danny seriously loves Japanese food so in the beginning of 2021, he decides to embark upon a risky venture and opens up a cute little restaurant that sells his 3 favourite foods: sushi, curry and ramen. Danny’s Diner is in need of your assistance to help the restaurant stay afloat – the restaurant has captured some very basic data from their few months of operation but have no idea how to use their data to help them run the business.

SQL PROJECT

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.

SCHEMA


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

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

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

	customer_id character varying (1)	total_amlount bigint
1	A	76
2	B	74
3	C	36

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

```
select customer_id,COUNT(DISTINCT order_date)
from sales
group by customer_id
order by customer_id;
```

	customer_id character varying (1)	count bigint
1	A	4
2	B	6
3	C	2

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

```sql
with ordered_date as(
    select s.customer_id
        , s.order_date
        , m.product_name
        , DENSE_RANK() over (partition by s.customer_id
          order by s.order_date) as rank_no
    from sales as s
    join menu as m
    on  m.product_id = s.product_id
    )
select customer_id , product_name
from ordered_date
where rank_no = 1
group by customer_id , product_name;
```

	customer_id character varying (1)	product_name character varying (5)
1	A	curry
2	A	sushi
3	B	curry
4	C	ramen

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

```sql
select product_name,count(order_date) as orders
from sales
inner join menu on sales.product_id = menu.product_id
group by product_name
order by orders desc
limit 1;
```

	product_name character varying (5)	orders bigint
1	ramen	8

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

```sql
with purchased_item as
(
    select s.customer_id
        , m.product_name
        ,COUNT(m.product_id) AS order_count
        ,RANK() over(partition by s.customer_id order by count(s.customer_id) desc) AS RANK
    from sales as s
    join menu as m
    on s.product_id = m.product_id
    group by s.customer_id , m.product_name
)

select customer_id , product_name,
order_count
from purchased_item
where RANK = 1;
```

	customer_id character varying (1)	product_name character varying (5)	order_count bigint
1	A	ramen	3
2	B	sushi	2
3	B	curry	2
4	B	ramen	2
5	C	ramen	3

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

```sql
with joined_member as (
    select s.customer_id
        , m.product_name
        , DENSE_RANK() over ( partition by s.customer_id
                              order by s.order_date) as rank_id
    from menu as m
    join sales as s
    on s.product_id = m.product_id
    join members as me
    on me.customer_id = s.customer_id
    where s.order_date > me.join_date
)
select  customer_id , product_name
from joined_member
where rank_id=1
group by customer_id , product_name;
```

	customer_id character varying (1)	product_name character varying (5)
1	A	ramen
2	B	sushi

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

```sql
with joined_member as (
    select s.customer_id
        , m.product_name
        , dense_rank() over ( partition by s.customer_id
                              order by s.order_date desc) as rank_id
    from menu as m
    join sales as s
    on s.product_id = m.product_id
    join members as me
    on me.customer_id = s.customer_id
    where s.order_date < me.join_date
)
select  customer_id , product_name
from joined_member
where rank_id=1
group by customer_id , product_name;
```

	customer_id character varying (1)	product_name character varying (5)
1	A	curry
2	A	sushi
3	B	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_item
     , SUM(m.price) as total_sales
from sales as s
join menu as m
    on s.product_id = m.product_id
join members as me
    on s.customer_id = me.customer_id
where s.order_date < me.join_date
group by s.customer_id;
```

	customer_id character varying (1)	total_item bigint	total_sales bigint
1	B	3	40
2	A	2	25

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

```sql
With Points as
(
Select *, Case When product_id = 1 THEN price*20
               Else price*10
               End as Points
From Menu
)
Select S.customer_id, Sum(P.points) as Points
From Sales S
Join Points p
On p.product_id = S.product_id
Group by S.customer_id;
```

	customer_id character varying (1)	points bigint
1	B	940
2	C	360
3	A	860

Let's

stay in touch

Picnic Rautaray

picnicrautaray9@gmail.com