BT query
GROUP BY LÀ CÓ SẴN ORDER BY
NÊN SAU GROUP BY K CẦN ORDER BY
—**1. What is the total amount each customer spent at the restaurant?**

```
SELECT
      s.customer_id,
   SUM(m.price)
FROM dannys_diner.menu as m
JOIN dannys_diner.sales as s on m.product_id = s.product_id
GROUP BY s.customer_id
```

—2 **How many days has each customer visited the restaurant?**

```
SELECT  customer_id,  COUNT(DISTINCT order_date)
FROM dannys_diner.sales
GROUP BY customer_id;
```

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

```
WITH ordered_sales 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 ORDER_RANK
FROM dannys_diner.sales AS S
JOIN dannys_diner.menu as M ON S.product_id = M.product_id
)

SELECT customer_id, product_name
FROM ordered_sales
WHERE ORDER_RANK = 1
GROUP BY customer_id, product_name
```

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

```
WITH PurchasedItemByCus as (
SELECT customer_id, product_id, COUNT(product_id) AS CountProduct
FROM dannys_diner.sales
GROUP BY customer_id, product_id
ORDER BY customer_id
)

 SELECT product_id, SUM(CountProduct)
 FROM PurchasedItemByCus
 GROUP BY product_id
 ORDER BY SUM(CountProduct) DESC
```

```
 LIMIT 1

 SELECT *
 FROM PurchasedItemByCus
```

—5 **Which item was the most popular for each customer?**

```
WITH MostPurchasedItemByCustomer as (
SELECT S.customer_id, S.product_id, M.product_name,
  COUNT(S.product_id) AS CountProduct,
  DENSE_RANK() OVER(
        PARTITION BY S.customer_id
    ORDER BY COUNT(S.product_id) DESC
  ) AS RANK1
FROM dannys_diner.sales as S
JOIN dannys_diner.menu as M ON S.product_id = M.product_id
GROUP BY S.customer_id, S.product_id, M.product_name
ORDER BY S.customer_id
)

SELECT customer_id, product_name, CountProduct
 FROM MostPurchasedItemByCustomer
 WHERE RANK1 = 1
 ORDER BY customer_id

-- 6. Which item was purchased first by the customer after they became a
member (Có gồm cả khách hàng chưa là mem hay không?)
WITH PurchasedFirstByMem as (
        SELECT M.product_name, MEM.customer_id, S.order_date,
        DENSE_RANK() OVER (
        PARTITION BY MEM.customer_id
        ORDER BY S.order_date
    ) as RANK1
        FROM dannys_diner.menu as M
        JOIN dannys_diner.sales AS S ON M.product_id = S.product_id
        JOIN dannys_diner.members AS MEM ON S.customer_id =
MEM.customer_id
        WHERE MEM.join_date < S.order_date
)

SELECT *
FROM PurchasedFirstByMem
WHERE RANK1 = 1

—-- 7. Which item was purchased just before the customer became a member?
(First?)
        SELECT S.customer_id, M.product_name
```

```
    FROM dannys_diner.members as MEM
  RIGHT JOIN dannys_diner.sales as S on S.customer_id = MEM.customer_id
  JOIN dannys_diner.menu as M on S.product_id = M.product_id
  WHERE MEM.join_date IS NULL OR MEM.join_date > S.order_date
  GROUP BY S.customer_id, M.product_name
  ORDER BY S.customer_id
```

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

```
    SELECT    S.customer_id, SUM(M.price), COUNT(S.product_id)
    FROM dannys_diner.members as MEM
    RIGHT JOIN dannys_diner.sales as S on S.customer_id =
MEM.customer_id
    JOIN dannys_diner.menu as M on S.product_id = M.product_id
    WHERE MEM.join_date IS NULL OR MEM.join_date > S.order_date
    GROUP BY S.customer_id
    ORDER BY S.customer_id
```

-- 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 * 10 * 2)
   else (M.price * 10) END) as TotalPoint
    FROM dannys_diner.sales as S
    LEFT JOIN dannys_diner.menu as M on S.product_id = M.product_id
    GROUP BY S.customer_id
    ORDER BY S.customer_id
```