**Q1)---1. Rank employees by their total sales**

**----(Total sales = Total no of orders handled, JOIN employees and orders table)**

**Query:**

SELECT

O.EMPLOYEE\_ID,

CONCAT(E.LAST\_NAME, ' ', E.FIRST\_NAME) AS FULL\_NAME,

DENSE\_RANK() OVER (

ORDER BY

COUNT(O.ORDER\_ID) DESC

)

FROM

ORDERS O

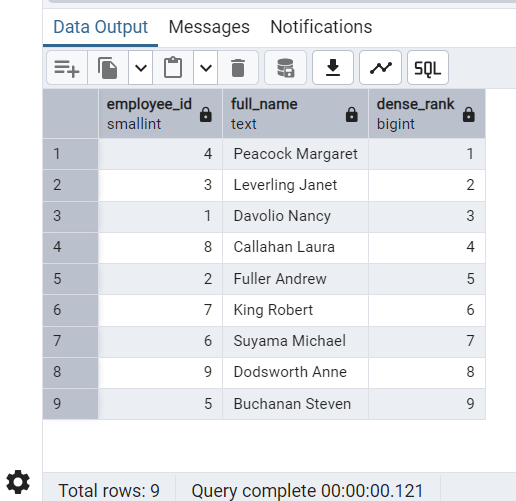
JOIN EMPLOYEES E ON O.EMPLOYEE\_ID = E.EMPLOYEE\_ID

GROUP BY

O.EMPLOYEE\_ID,

FULL\_NAME;

Output:



**Q2) Compare current order's freight with previous and next order for each customer.--(Display order\_id, customer\_id, order\_date, freight,Use lead(freight) and lag(freight).**

**Query:**

SELECT ORDER\_ID,CUSTOMER\_ID,ORDER\_DATE,FREIGHT,

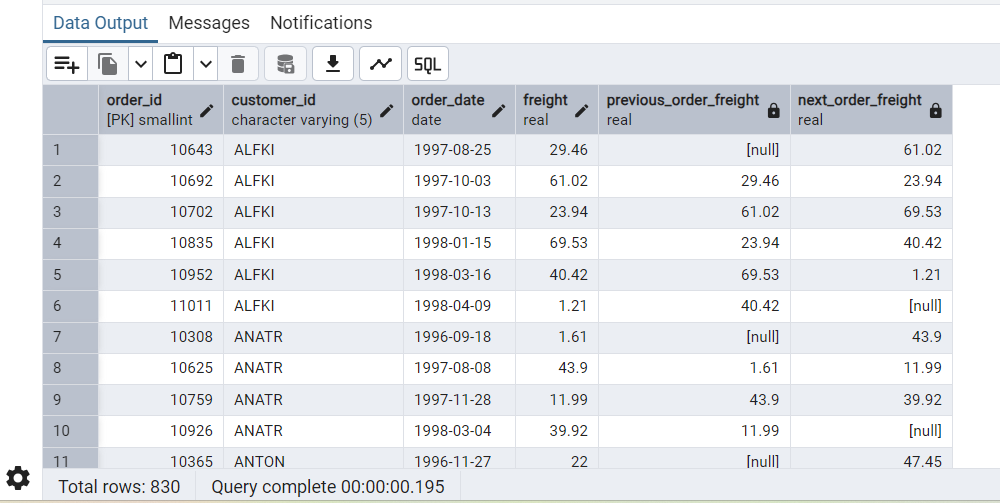
LAG(FREIGHT) OVER (PARTITION BY CUSTOMER\_ID ORDER BY ORDER\_ID) AS PREVIOUS\_ORDER\_FREIGHT,

LEAD(FREIGHT) OVER (PARTITION BY CUSTOMER\_ID ORDER BY ORDER\_ID) AS NEXT\_ORDER\_FREIGHT

FROM ORDERS

ORDER BY CUSTOMER\_ID,ORDER\_ID;

Output:



**Q3) Show products and their price categories, product count in each category, avg price:**

**--(HINT:**

**--· Create a CTE which should have price\_category definition:**

**-- WHEN unit\_price < 20 THEN 'Low Price'**

**-- WHEN unit\_price < 50 THEN 'Medium Price'**

**-- ELSE 'High Price'**

**--· In the main query display: price\_category, product\_count in each price\_category, ROUND(AVG(unit\_price)::numeric, 2) as avg\_price)**

**Query:**

WITH

CATEGORY AS (

SELECT

PRODUCT\_ID,

UNIT\_PRICE,

CASE

WHEN UNIT\_PRICE < 20 THEN 'Low Price'

WHEN UNIT\_PRICE < 50 THEN 'Medium Price'

ELSE 'high price'

END AS PRICE\_CATEGORY

FROM

PRODUCTS

)

SELECT

PRODUCT\_ID,

UNIT\_PRICE,

PRICE\_CATEGORY,

COUNT(PRODUCT\_ID) OVER (

PARTITION BY

PRICE\_CATEGORY

) AS PRODUCT\_COUNT,

ROUND(

AVG(UNIT\_PRICE) OVER (

PARTITION BY

PRICE\_CATEGORY

)::NUMERIC,2

) AS AVERAGE\_UNIT\_PRICE

FROM CATEGORY;

Output:

