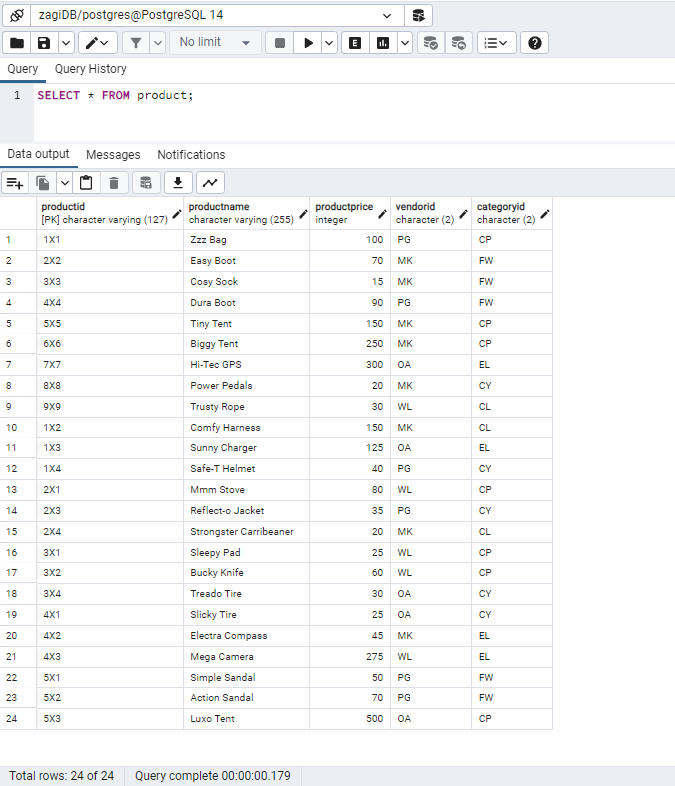
**LAB4**

**SIMRAN- 100377444**

QUERY1. Retrieve the entire contents of the Product table (all columns and all rows of the table).

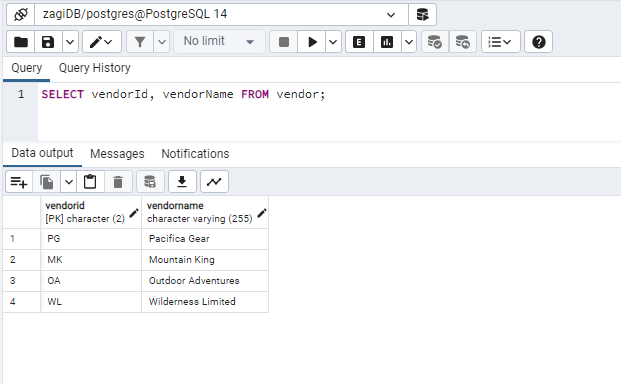
ANS: SELECT \* FROM product;



Query2. Display the VendorID and VendorName for all vendors.

ANS: SELECT vendorId , vendorName

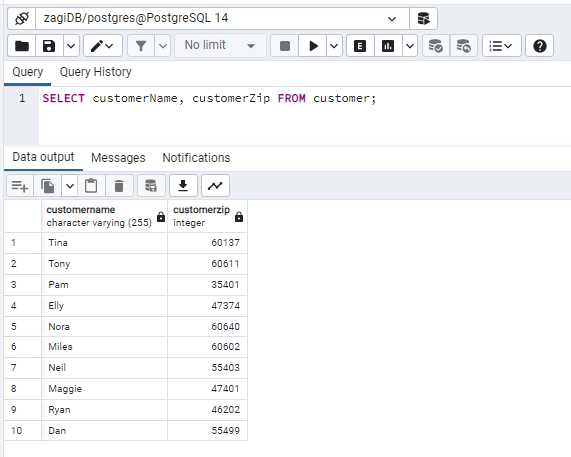
FROM vendor;



Query3. Display the CustomerName and CustomerZip for all customers.

ANS: SELECT CustomerName, CustomerZip

FROM customer;

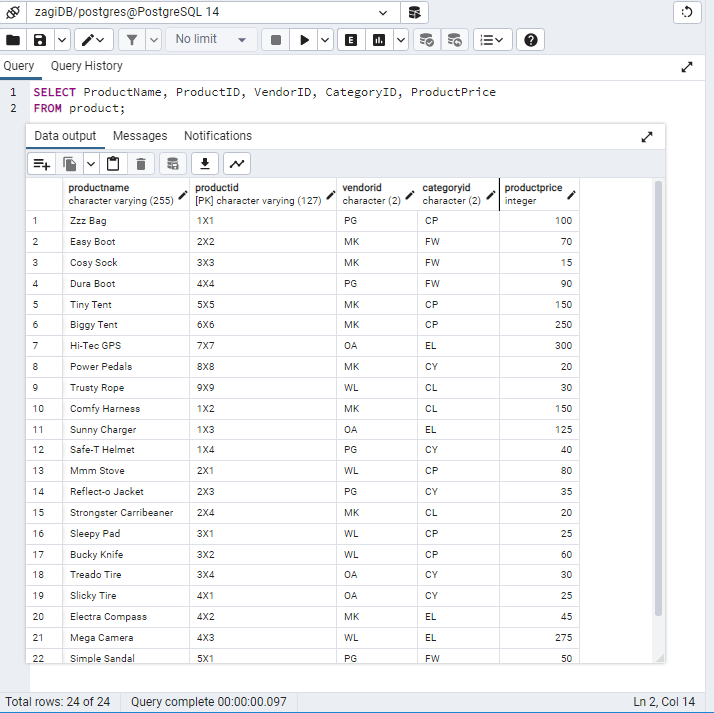


QUERY4. Retrieve the entire contents of the table PRODUCT. The columns must be displayed in the following order: ProductName, ProductID, VendorID, CategoryID, ProductPrice.

ANS:

SELECT ProductName, ProductID, VendorID, CategoryID, ProductPrice

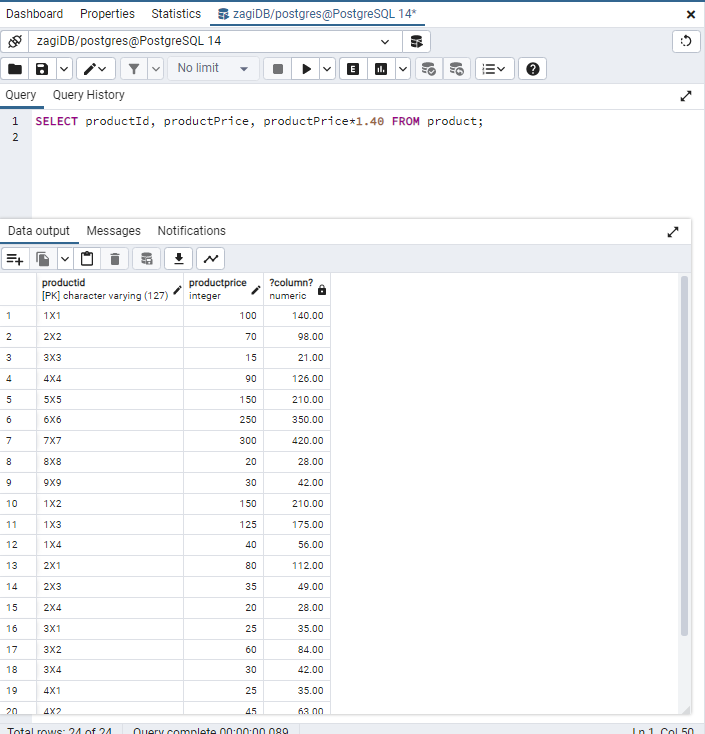
FROM product;



QUERY5. For the table PRODUCT, display 3 columns ProductID, ProductPrice, and a column showing ProductPrice increased by 40%

ANS: SELECT productID, productPrice, productPrice\*1.40

FROM product;

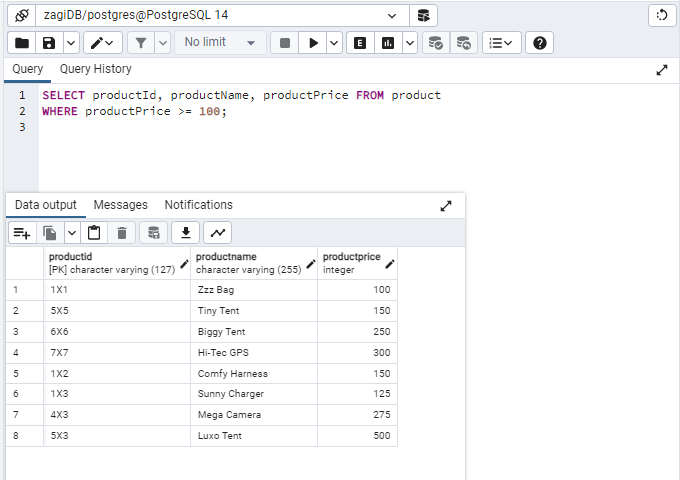


QUERY6. Display the ProductID, ProductName, and ProductPrice for products with a ProductPrice of $100 or higher.

ANS: SELECT productID, productName, productPrice

FROM product

WHERE productPrice >= 100;

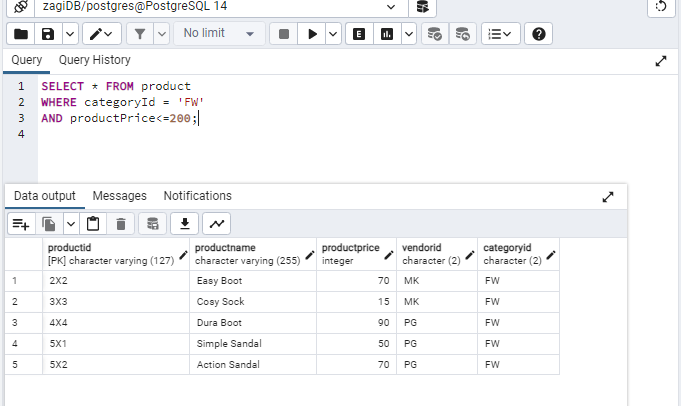


QUERY7. Retrieve the ProductID, ProductName, VendorID, CategoryID, and ProductPrice of products in the FW category whose price is equal to or below $200 (Hint: two conditions in WHERE clause)

ANS: SELECT \* FROM product

WHERE categoryId = ‘FW’

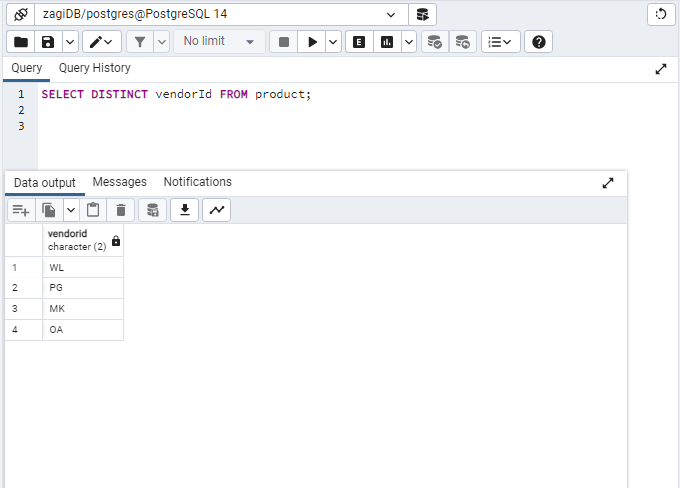
AND productPrice <=200;



QUERY8. Display the VendorID of all vendors that we have a product from them. In the result, we must not see duplicate vendorIDs.

ANS: SELECT DISTINCT vendorId

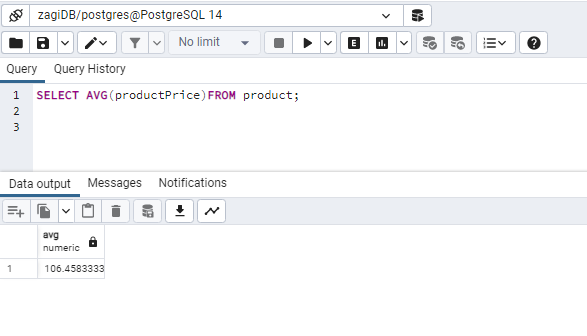
FROM Product;



QUERY 9. Retrieve the average price of all products. (Hint: Use AVG)

ANS: SELECT AVG(productPrice)

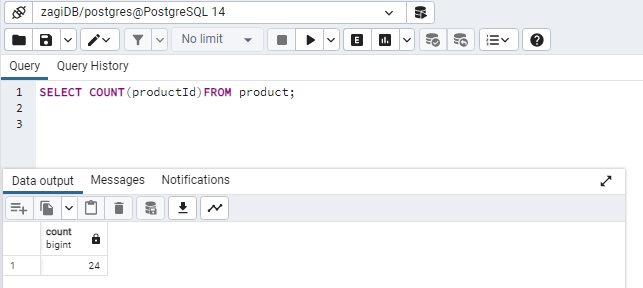
FROM product;



QUERY10. Show how many products are there for sale. (Hint: COUNT)

ANS: SELECT COUNT(productId)

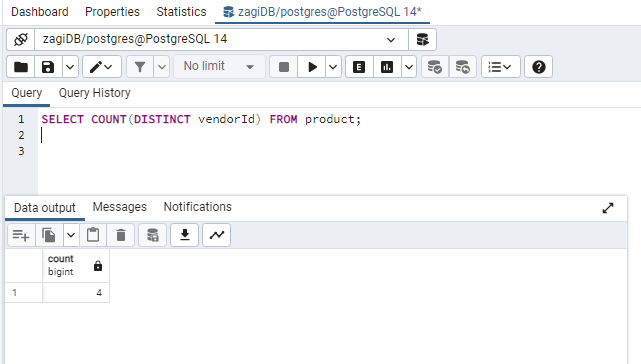
FROM product;



QUERY 11. Count how many distinct vendors are there in the product table. (Hint: the answer is 4)

ANS: SELECT COUNT (DISTINCT vendorId)

FROM product;



QUERY12. Retrieve the number of products, average product price, lowest product price, and highest product price in the CP product category.

ANS:

SELECT COUNT(productId) AS noOfProducts,

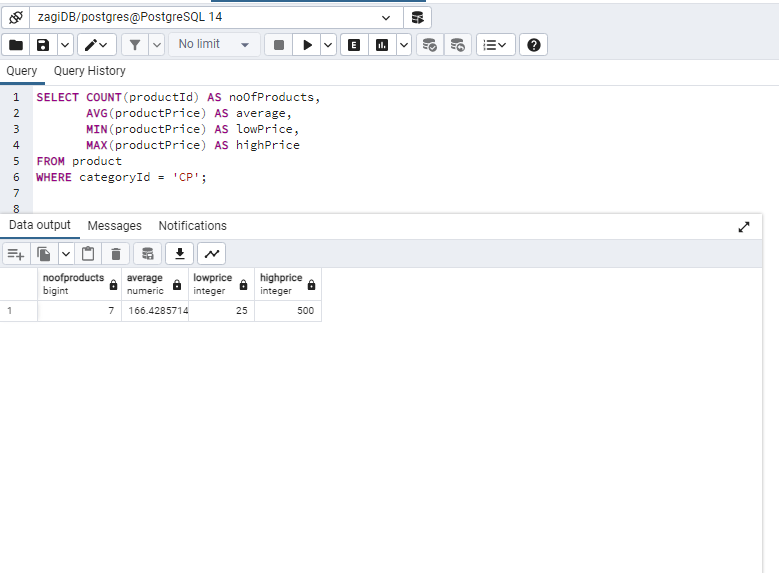
AVG (productPrice) AS average,

MIN(productPrice ) AS lowPrice,

MAX(productPrice) AS highPrice

FROM product

WHERE category = ’CP’;



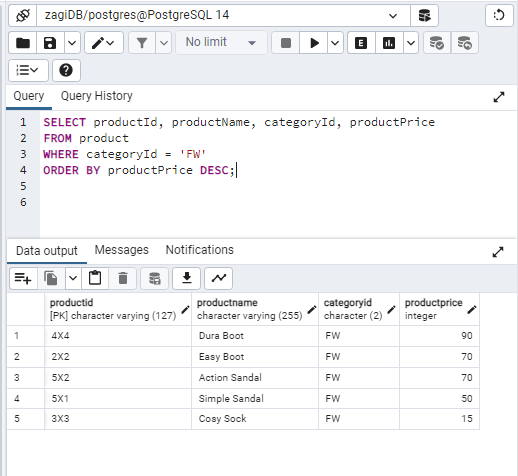
QUERY13. Retrieve the product ID, product name, category ID, and product price for each product in the FW product category, sorted by product price in descending order.

ANS: SELECT productID, productName, categoryId, productPrice

FROM product

WHERE category = ‘FW’

ORDER BY productPrice DESC;



QUERY14. For each product, retrieve the ProductID, and the total number of product items sold within all transactions. (Hint: transactions can be found in SoldVia table. NoOfItems attribute holds how many items sold in one transaction, but we need to find the total sold of a product in all transactions. See the table below for your reference).

We need the heading of the columns be exactly ‘PRODUCTID’ and ‘Total Sold’ as you can see below.

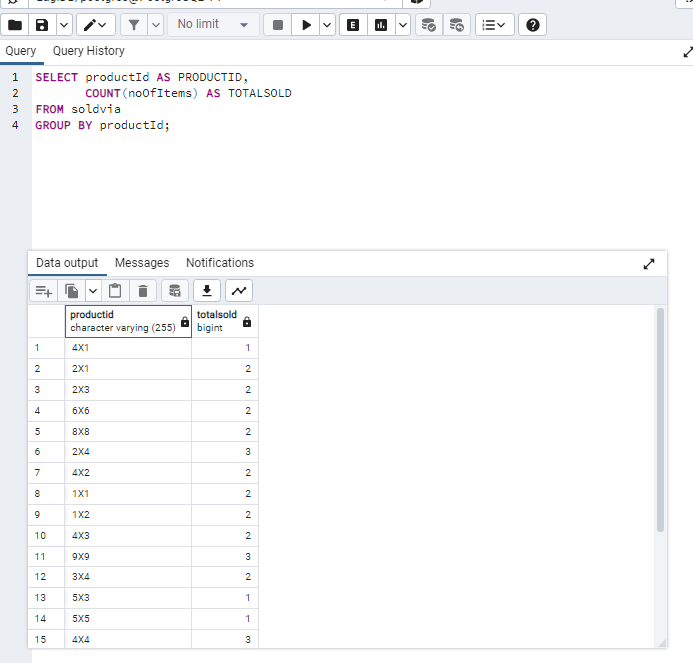
ANS:

SELECT productId AS PRODUCTID,

COUNT(noOfItems) AS TOTALSOLD

FROM soldvia

GROUP BY productId;



Query 15. For each vendor, retrieve the VendorID, number of products supplied by the vendor, and average price of the products supplied by the vendor.

ANS:

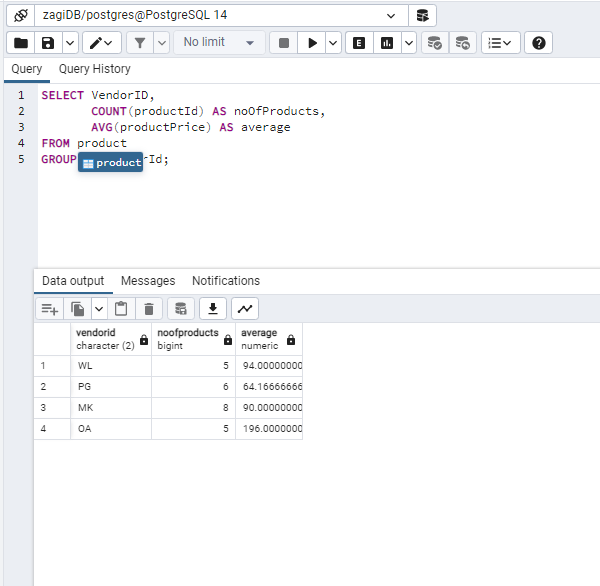
SELECT VendorID,

COUNT(productId) AS noOfProducts,

AVG(productPrice) AS average

FROM product

GROUP BY vendorId;

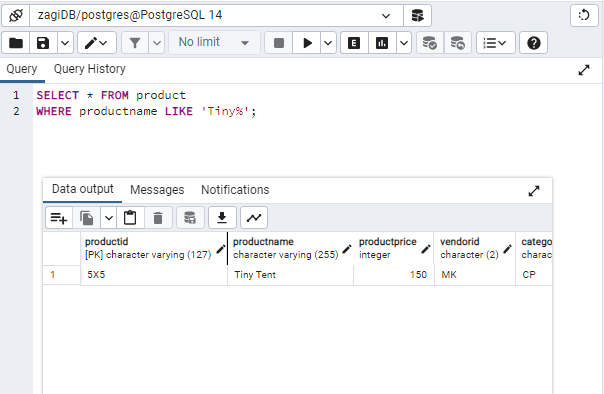


Query 16. Retrieve all attributes of products whose name starts with “Tiny”, for example, ‘Tiny Tent’

ANS:

SELECT \* FROM product

WHERE productname LIKE 'Tiny%';



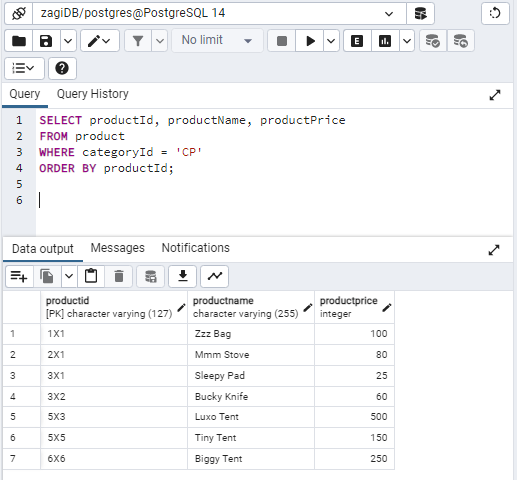
Query 17. Display the ProductID, ProductName, and ProductPrice for productsin the category whose CategoryID value is‘CP’. Sort the results by ProductID.

ANS: SELECT productID, productName, productPrice

FROM product

WHERE categoryId =’CP’

ORDER BY productId;



QUERY18. Display the transaction id (TID) and the total number of items sold in that transaction (of all products) that the total number of items (of all products) sold in that transaction is greater than five. In other words, we want to get the sample result as follows.

We need the heading of the columns be exactly ‘TID’ and ‘Total Items Sold’ as you can see below.

ANS:

SELECT tId AS TID,

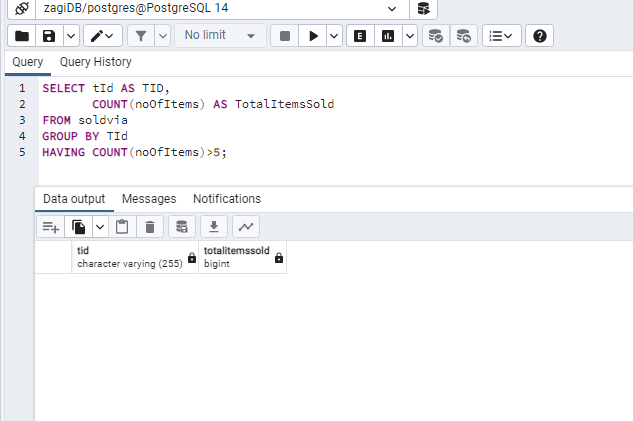
COUNT(noOfItems) AS TotalItemsSold

FROM soldvia

GROUP BY TId

HAVING COUNT(noOfItems)>5;

# There is no transaction in which the number of items sold are more than 5.



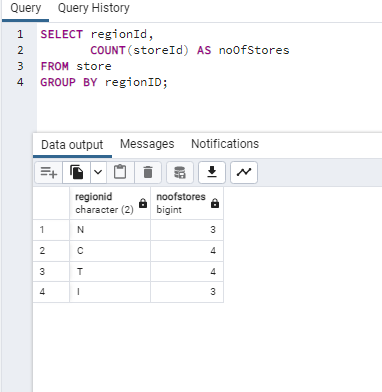
Query 19. Display all RegionIDs and number of stores in their region.

ANS: SELECT regionId,

COUNT(storeId) AS noOfStores

FROM store

GROUP BY regionID;



Query 20. Display RegionID and number of Stores in regions that number of stores in their is 4 or more.

ANS: SELECT regionId,

COUNT(storeId) AS noOfStores

FROM store

GROUP BY regionID

HAVING COUNT(storeId)>=4;

