**SQL - END OF MODULE ASSESSMENT**

**SQL Tasks**

**Task 1**. Retrieve all orders for products that cost more than $25, displaying the OrderID, ProductID, Quantity, and Total.

SELECT Orders.OrderID, Products.ProductID, Orders.Quantity, Orders.Total

FROM Products

JOIN Orders

ON Products.ProductID = Orders.ProductID

WHERE Products.price > 25

ORDER BY Orders.ProductID ASC , Orders.Quantity ASC

A screenshot of a computer

AI-generated content may be incorrect.

**Task 2.** List each customer and the total quantity of products they ordered, sorted by the highest total quantity.

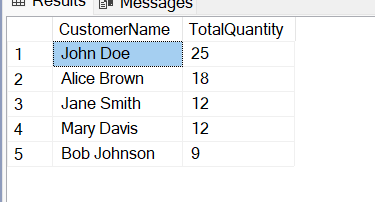
SELECT Customers.CustomerName, Sum(Orders.Quantity) AS TotalQuantity

FROM Customers

JOIN Orders ON Customers.CustomerID = Orders.CustomerID

GROUP BY Customers.CustomerName

ORDER BY TotalQuantity DESC, Customers.CustomerName



**Task 3.** Display each product’s name, and its total sales, and classify it as "High Revenue" if total sales exceed $300, otherwise "Low Revenue".

SELECT Products.ProductName, SUM(Orders.Total) AS TotalSales,

CASE

WHEN SUM(Orders.Total) > 300 THEN 'High Revenue'

ELSE 'Low Revenue'

END AS RevenueCategory

FROM Products

JOIN Orders

ON Products.ProductID = Orders.ProductID

GROUP BY Products.ProductName

A screenshot of a computer

AI-generated content may be incorrect.

**Task 4.** Find customers who placed at least two orders in December 2023, with each order totaling over $100. Display their names and the count of such qualifying orders.

SELECT Customers.CustomerName, COUNT(Orders.OrderID) AS OrderCount

FROM Customers

JOIN Orders

ON Customers.CustomerID = Orders.CustomerID

WHERE

YEAR(Orders.OrderDate) = 2023

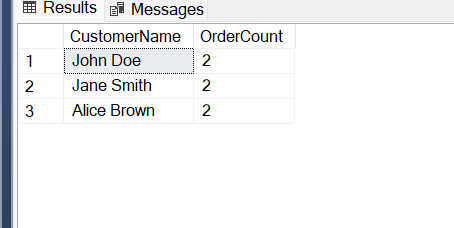
AND MONTH(Orders.OrderDate) = 12

AND Orders.Total > 100

GROUP BY Customers.CustomerName

HAVING COUNT(Orders.OrderID) >= 2

ORDER BY Customers.CustomerName DESC



**Task 5.** List each order with the first three characters of the customer name, the product name, and the total order amount.

SELECT Orders.OrderID, LEFT(Customers.CustomerName, 3) AS ShortCustomerName, Products.ProductName, Orders.Total

FROM Customers

JOIN Orders

ON Customers.CustomerID = Orders.CustomerID

JOIN Products

ON Products.ProductID = Orders.ProductID

ORDER BY Products.ProductName ASC

A screenshot of a computer

AI-generated content may be incorrect.

**Task 6.** Group the orders by product name, convert the product names to lowercase, and show the total quantity ordered for each product, sorted by product name.

SELECT LOWER(Products.ProductName) AS ProductNameLower, SUM(Orders.Quantity) AS TotalQuantityOrdered

FROM Products

JOIN Orders

ON Products.ProductID = Orders.ProductID

GROUP BY

LOWER(Products.ProductName)

ORDER BY

ProductNameLower

A screenshot of a computer

AI-generated content may be incorrect.