Informing a Business Decision Using SQL

For this project, I will be analyzing a database from a fictional company called Northwind Traders that buys products and resells them to customers.

**Database:**

A diagram of a flowchart

Description automatically generated with medium confidence

**Scenario:**

As a method of increasing future sales, the company has decided to give employee bonuses for exemplary performance in sales.

Bonuses will be awarded to those employees who are responsible for the five highest order amounts.

How can I identify these employees?

**Solution:**

I need to obtain a list of employees who have orders with the five highest sales amounts.

To get the sales amount for each employee, I will need to calculate the quantity multiplied by the price for each order.

Tables/Fields needed:

|  |  |
| --- | --- |
| **Tables** | **Fields** |
| Employees | Firstname, Lastname |
| Orders | OrderID |
| OrderDetails | ProductID, Quantity |
| Products | Price |

I will need to join the four tables listed above with the following query:

SELECT LastName, FirstName, Orders.OrderID, Products.ProductID,

Quantity, Price

FROM employees

join orders

on employees.employeeID = orders.employeeid

join orderDetails

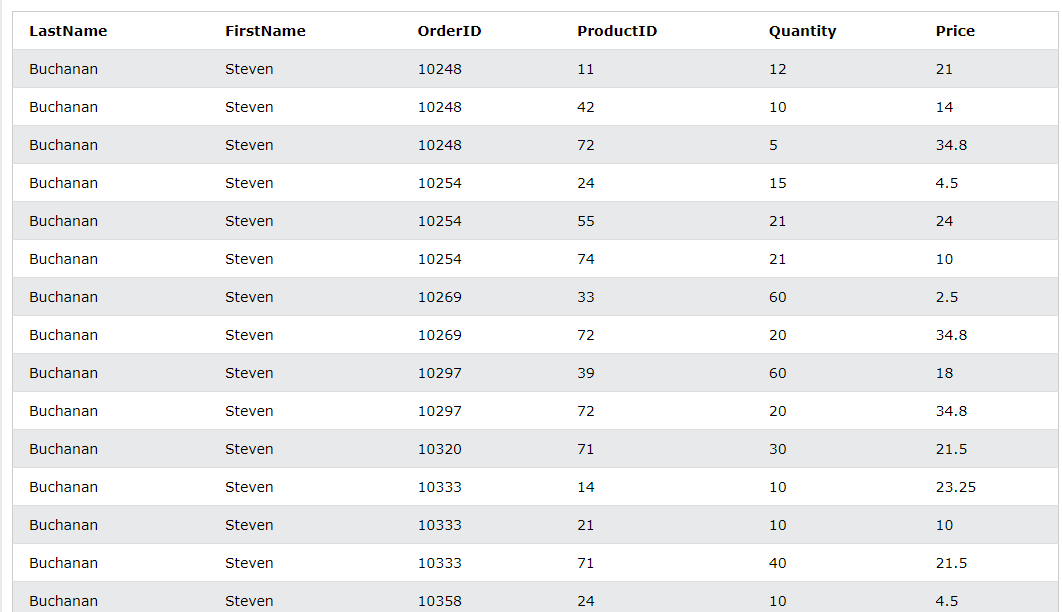
on orders.orderid = orderdetails.orderid

join products

on orderdetails.productid = products.productid

ORDER BY lastname, firstname

Giving the following result:



I will now need to calculate the sales amount per order. To do this I will multiply the quantity by price for each product and then aggregate the data to get one sales amount per order.

SELECT LastName, FirstName, Orders.OrderID, Products.ProductID,

Quantity, Price, SUM(quantity\*price) as SalesAmount

FROM employees

join orders

on employees.employeeID = orders.employeeid

join orderDetails

on orders.orderid = orderdetails.orderid

join products

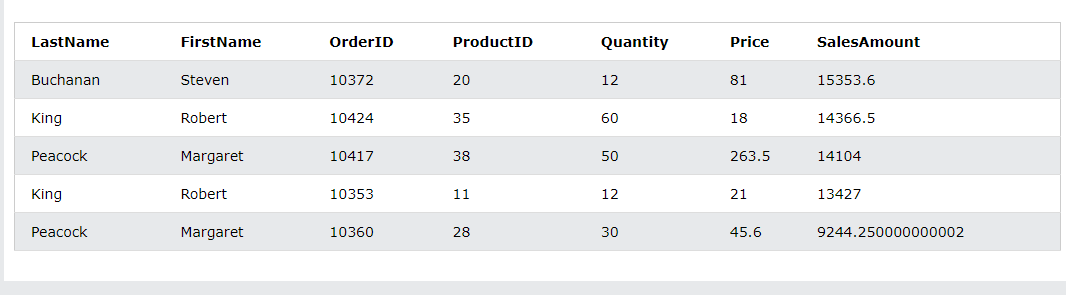
on orderdetails.productid = products.productid

Group by Orders.OrderID

Order by SalesAmount Desc

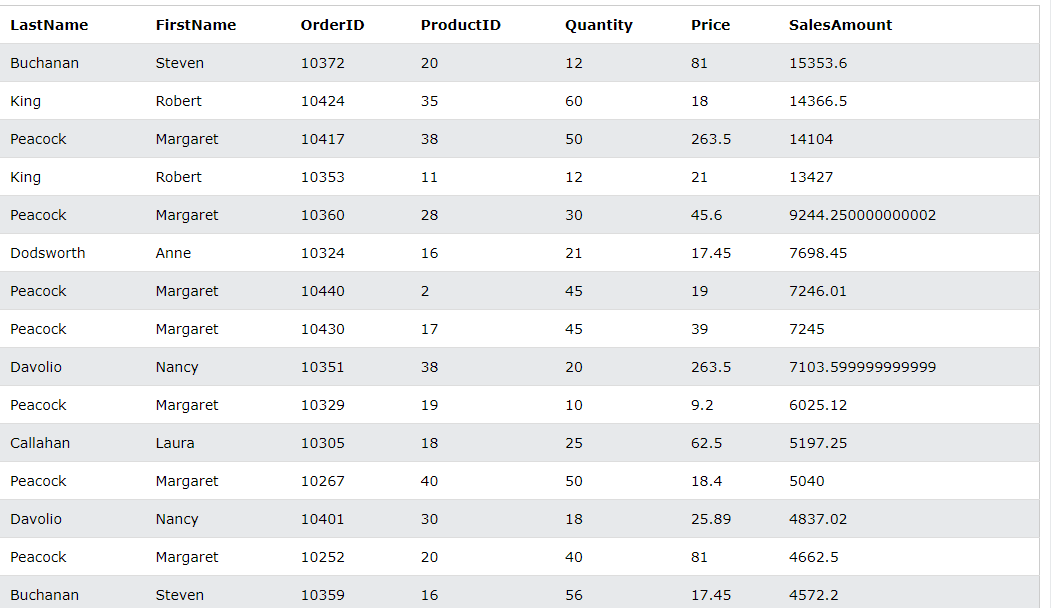
Limit 5

Giving the following result:



The data shows that three employees are responsible for the top five sales amounts, as Robert King and Margaret Peacock have two orders in the top five sales amounts. However, if the sales manager planned to give bonuses to five employees, I can provide a second report giving the sales manager more data in order to decide how to award the bonuses.

Looking at the results beyond the top five:



The next two employees with the highest sales are Anne Dodsworth and Nancy Davolio.

I can use the ‘Having’ clause in my query to filter the order ID’s with the highest sales for the top 5 employees:

SELECT LastName, FirstName, Orders.OrderID, Products.ProductID,

Quantity, Price, SUM(quantity\*price) as SalesAmount

FROM employees

join orders

on employees.employeeID = orders.employeeid

join orderDetails

on orders.orderid = orderdetails.orderid

join products

on orderdetails.productid = products.productid

Group by Orders.OrderID

Having orders.orderid in (10372, 10424, 10417, 10324, 10351)

Order by SalesAmount Desc

Which results in the following:

