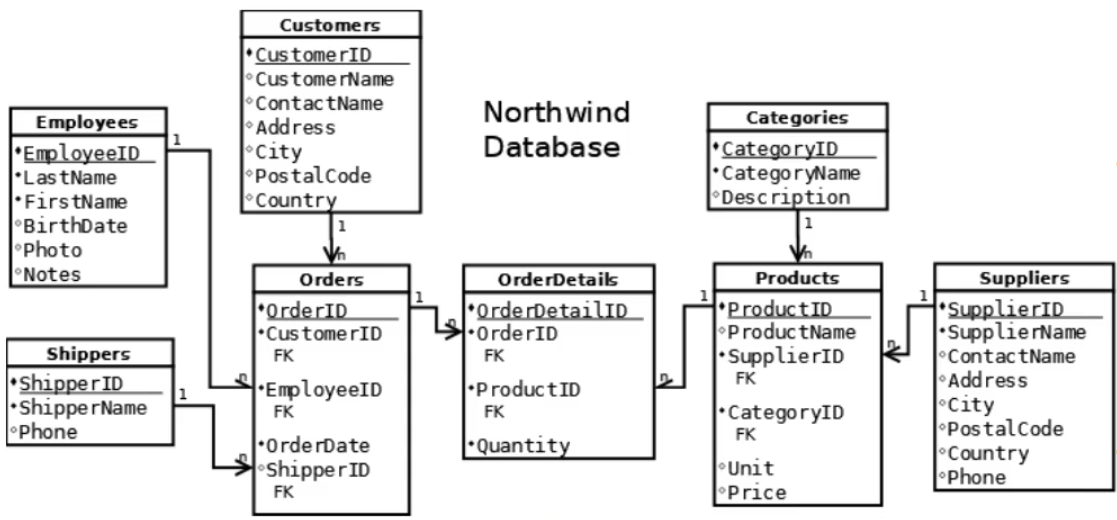
**Objective 1:** Explore the Northwind database.

**Database Overview:**



**Objective 2:** Examine the contents of database tables to identify data tables and fields required to inform a business decision.

**Business Question:**

*“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 we identify those employees?”*

**Proposed Solution:**

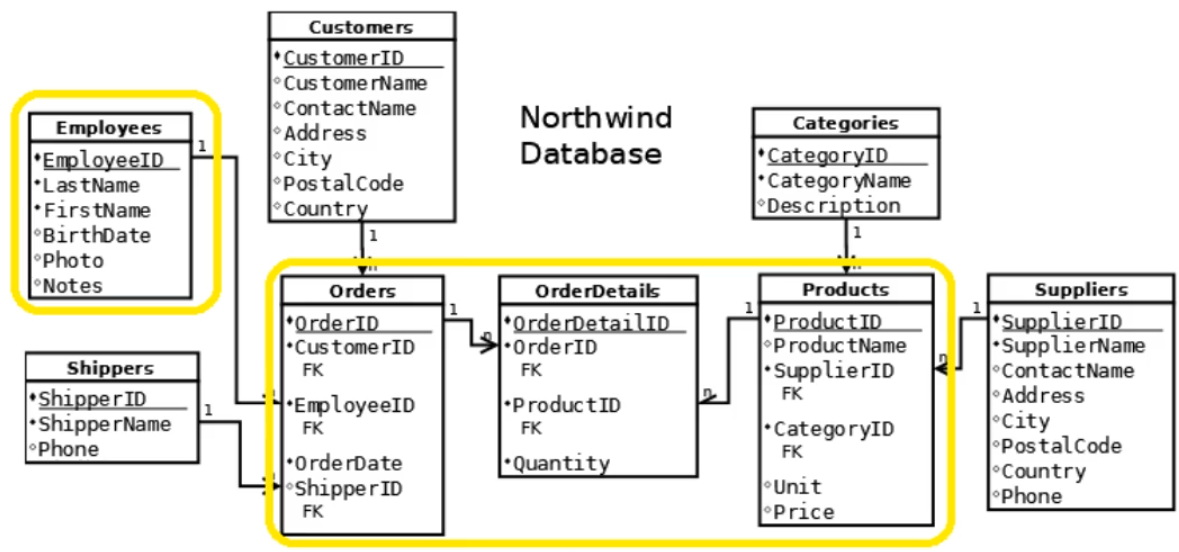
A list of employees who have orders with five highest sales amounts.

**Tables used for answering the business question:**

* Employees: First and last names of employees
* Orders: Orders with the highest sales value
* OrderDetails: Quantity to compute the sales value
* Products: Unit Price to compute the sales value

**Sales Value per Order**

Sales amount per item = Quantity multiplied by Unit Price

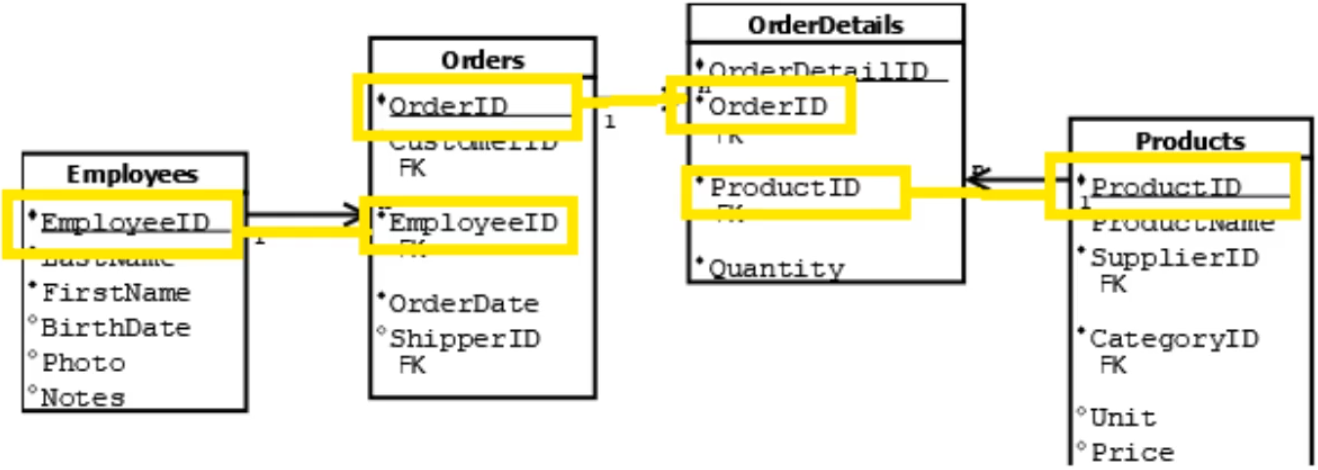


**Objective 3:** Retrieve the data needed to solve a business question by joining multiple tables in SQL.

**Tables and fields used for answering the business question:**

* Employees: LastName and FirstName
* Orders: OrderID
* OrderDetails: ProductID and Quantity
* Products: Price

Common key between the related tables:



Joining the 4 tables with their related keys

*\*Note: W3Schools SQL system is slightly different with other SQL programs, that is why there is a parenthesis in FROM statement in order to execute the query.*

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

FROM ((Employees

INNER JOIN Orders

ON Employees.EmployeeID = Orders.EmployeeID)

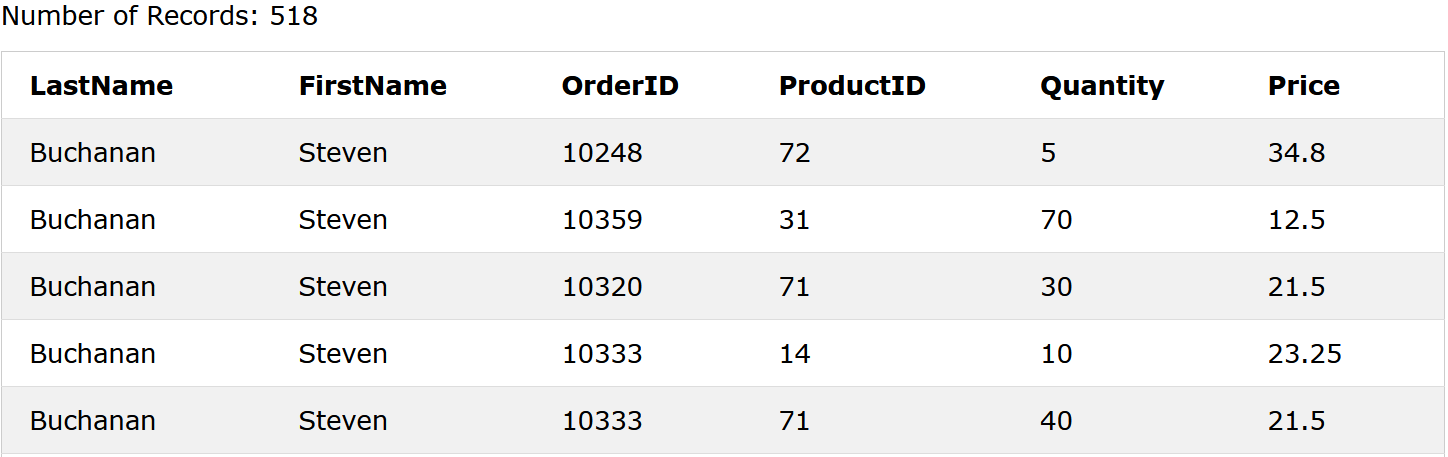
INNER JOIN OrderDetails

ON Orders.OrderID = OrderDetails.OrderID)

INNER JOIN Products

ON OrderDetails.ProductID = Products.ProductID

ORDER BY LastName, FirstName;



Objective 4: Write SQL code to calculate and aggregate data

Adding a new column of sales value per order

SELECT LastName, FirstName, Orders.OrderID, Products.ProductID, Quantity, Price, Quantity \* Price AS SalesAmt

FROM ((Employees

INNER JOIN Orders

ON Employees.EmployeeID = Orders.EmployeeID)

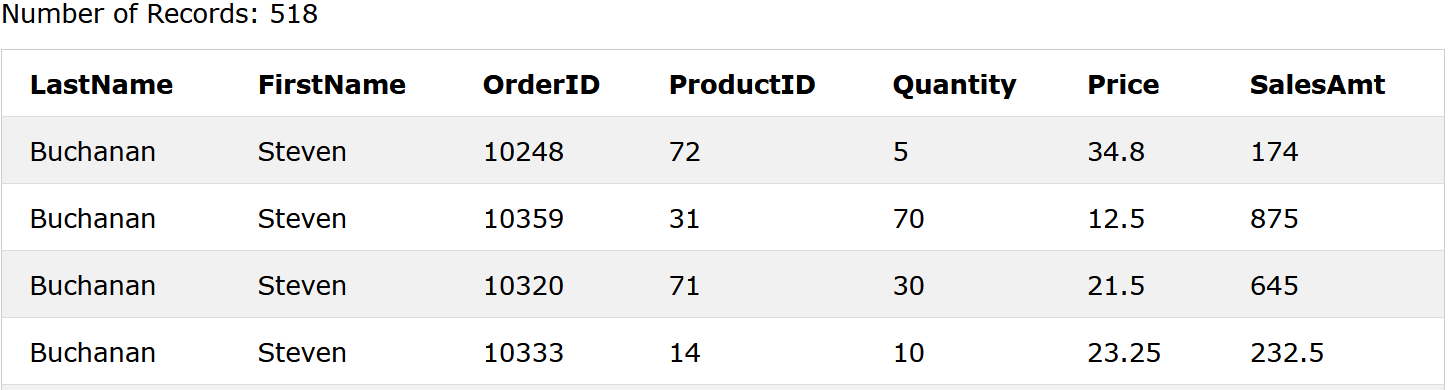
INNER JOIN OrderDetails

ON Orders.OrderID = OrderDetails.OrderID)

INNER JOIN Products

ON OrderDetails.ProductID = Products.ProductID

ORDER BY LastName, FirstName;



Taking the sum of the SalesAmt column

SELECT LastName, FirstName, Orders.OrderID, SUM(Quantity \* Price) AS SalesAmt

FROM ((Employees

INNER JOIN Orders

ON Employees.EmployeeID = Orders.EmployeeID)

INNER JOIN OrderDetails

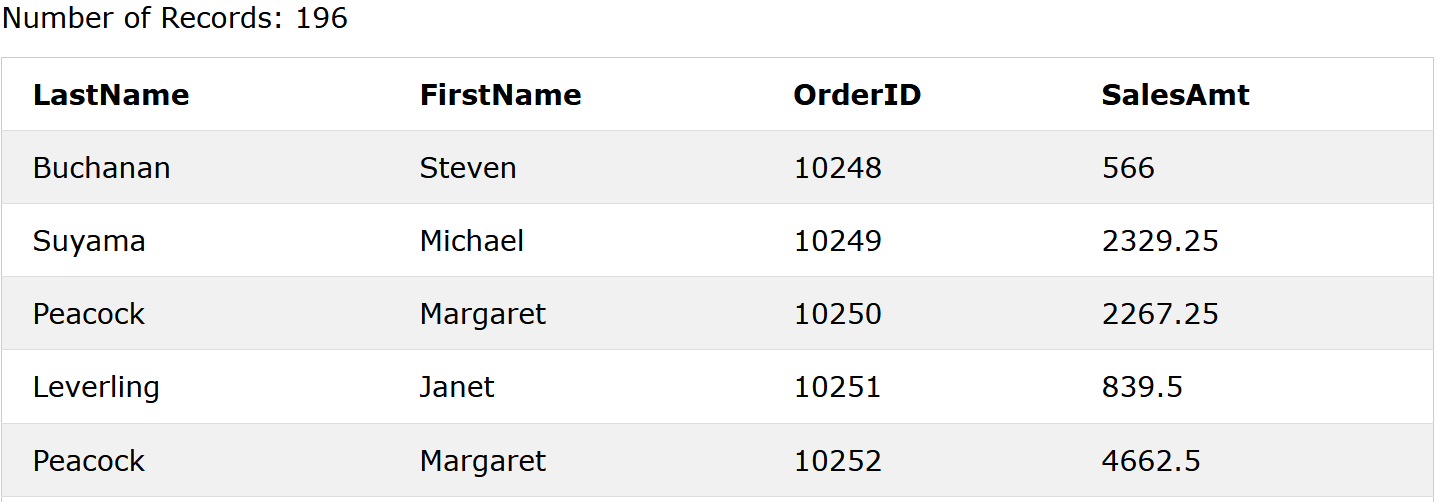
ON Orders.OrderID = OrderDetails.OrderID)

INNER JOIN Products

ON OrderDetails.ProductID = Products.ProductID

GROUP BY LastName, FirstName, Orders.OrderID

ORDER BY Orders.OrderID;



Objective 5: Display data in a format that can be used to inform a business decision

Displaying by SalesAmt by descending to determine the top 5 employees with most sales per order

SELECT TOP 7 LastName, FirstName, Orders.OrderID, SUM(Quantity \* Price) AS SalesAmt

FROM ((Employees

INNER JOIN Orders

ON Employees.EmployeeID = Orders.EmployeeID)

INNER JOIN OrderDetails

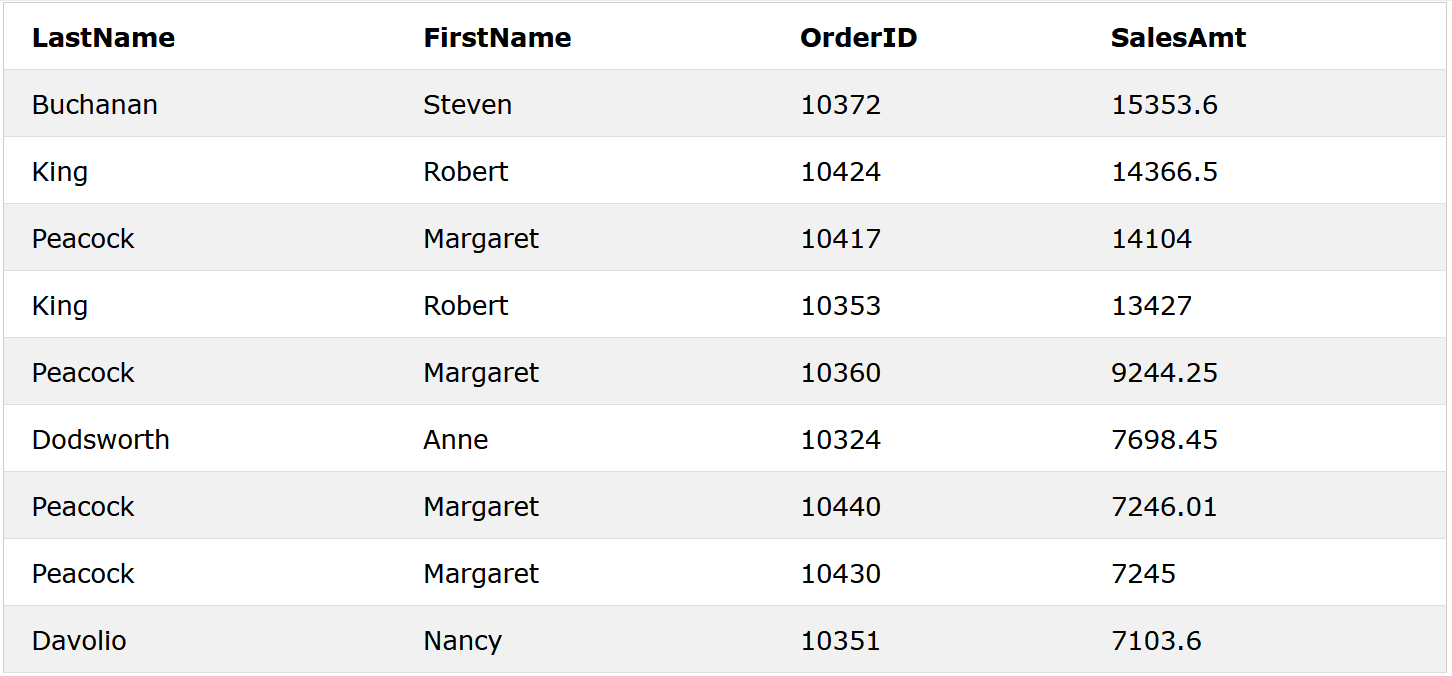
ON Orders.OrderID = OrderDetails.OrderID)

INNER JOIN Products

ON OrderDetails.ProductID = Products.ProductID

GROUP BY LastName, FirstName, Orders.OrderID

ORDER BY SUM(Quantity \* Price) DESC;



If we check their order numbers, the unique top 5 customers have OrderID = 10372, 10424, 10417, 10324 and 10351

Filtering the OrderIDs

SELECT TOP 7 LastName, FirstName, Orders.OrderID, SUM(Quantity \* Price) AS SalesAmt

FROM ((Employees

INNER JOIN Orders

ON Employees.EmployeeID = Orders.EmployeeID)

INNER JOIN OrderDetails

ON Orders.OrderID = OrderDetails.OrderID)

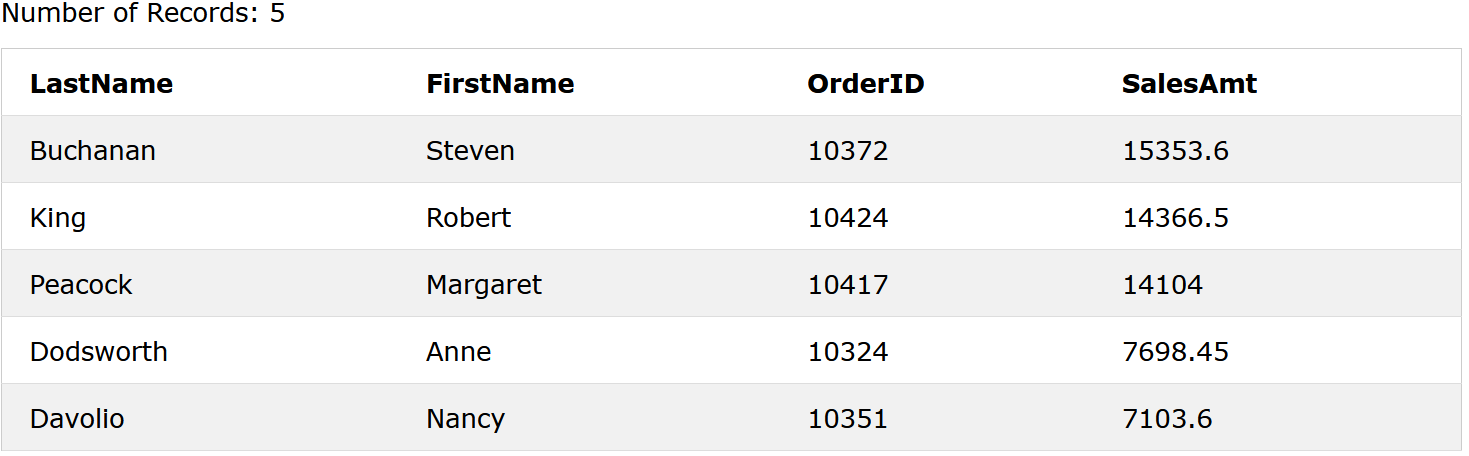
INNER JOIN Products

ON OrderDetails.ProductID = Products.ProductID

GROUP BY LastName, FirstName, Orders.OrderID

HAVING Orders.OrderID IN (10372, 10424, 10417, 10324, 10351)

ORDER BY SUM(Quantity \* Price) DESC;



Conclusion:

“The top 5 unique customers with the most sales per order ID are the following:

1. Steven Buchanan
2. Robert King
3. Margaret Peacock
4. Anne Dodsworth
5. Nancy Davolio

Key Takeaways

Task 1

Title: Project and Database Introduction

* A database is a collection of related tables.
* Data is stored in tables and is organized into rows and columns.
* The W3Schools SQL Tryit Editor provides a place to write and run SQL code.

Task 2

Title: A Look at the Question and the Suggested Solution

* The data analyst strives to use data to answer business questions.
* A business problem is best solved by examining the data available and visualizing which pieces of data should make up the solution.
* Data from multiple tables may be required to solve a business problem.

Task 3

Title: Joining Tables Together in SQL to Obtain Data for Analysis

* To join two tables, they must share a common column.
* The SQL INNER JOIN command returns only rows that match between two tables.
* An SQL query can be keyed into the SQL Tryit editor without regard to case; however, correct spelling and punctuation is critical

Task 4

Title: Calculate and Summarize Sales for each Order

* New, temporary fields can be created as a result of a calculation in SQL.
* Aggregating or grouping data can make it more useful for decision making.
* In SQL code, the SUM() function, together with the GROUP BY clause, can be used to aggregate data.

Task 5

Title: Display the Solution

* Limiting the number of rows that display as the result of an SQL query can be accomplished using the LIMIT command.
* In SQL code, the HAVING command applies a filter after aggregation.
* A data analyst often strives to anticipate alternative types of data that may address a business problem.

Sample SQL Code:

SELECT LastName, FirstName, Orders.OrderID, sum(Quantity \* Price) as SalesAmt

FROM employees

inner join orders

on employees.employeeID = orders.employeeid

inner join orderDetails

on orders.orderid = orderdetails.orderid

inner join products

on orderdetails.productid = products.productid

group by orders.orderid

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

order by salesamt desc