



# NORTHWIND TRADERS

## SALES REPORT

### 1994-1996



Report by

Siddhant Chandekar

## Detailed Dataset Schema Report: Northwind Traders

This report presents a comprehensive overview of the Northwind Traders dataset, meticulously detailing each CSV file, which functions as a distinct table within a relational database structure. For each table, its primary purpose, columnar schema, and inferred data types are precisely delineated.

### 1. categories.csv

**Purpose:** This file serves to define and categorize the various product types available within the inventory system.

Column Name	Inferred Data Type	Description
CategoryID	INTEGER	A unique numerical identifier assigned to each product category, serving as the primary key for this table.
CategoryName	TEXT	The designated name of the product category (e.g., Beverages, Confections).
Description	TEXT	A concise textual explanation providing further detail regarding the category.
Picture	TEXT	A reference or file path to an associated image, typically stored as binary data or a text-based reference in database systems.

### 2. customers.csv

**Purpose:** This file is dedicated to storing comprehensive information pertaining to the clientele who initiate and place orders.

Column Name	Inferred Data Type	Description
CustomerID	TEXT	A unique alphanumeric identifier for each customer, functioning as the primary key for this table.
CompanyName	TEXT	The formal name of the

		customer's organization or business entity.
ContactName	TEXT	The full name of the primary individual designated as the contact person within the customer's company.
ContactTitle	TEXT	The professional designation or job title held by the contact person.
Address	TEXT	The complete street address associated with the customer's primary location.
City	TEXT	The municipality in which the customer is situated.
Region	TEXT	The state or province corresponding to the customer's geographical location; this field may contain null values.
PostalCode	TEXT	The postal or zip code relevant to the customer's address.
Country	TEXT	The nation in which the customer is geographically located.
Phone	TEXT	The primary telephone number for the customer.
Fax	TEXT	The facsimile number associated with the customer; this field may contain null values.

### 3. employees.csv

**Purpose:** This file contains detailed records concerning the company's workforce, with a particular emphasis on personnel engaged in sales and order processing functions.

Column Name	Inferred Data Type	Description
EmployeeID	INTEGER	A unique numerical identifier assigned to each employee,

		serving as the primary key for this table.
LastName	TEXT	The surname of the employee.
FirstName	TEXT	The given name of the employee.
Title	TEXT	The official job designation or role held by the employee (e.g., Sales Representative).
TitleOfCourtesy	TEXT	A formal honorific or courtesy title used for the employee (e.g., Mr., Ms., Dr.).
BirthDate	DATE	The employee's date of birth.
HireDate	DATE	The date on which the employee commenced their employment with the company.
Address	TEXT	The residential street address of the employee.
City	TEXT	The municipality where the employee resides.
Region	TEXT	The state or province of the employee's residence; this field may contain null values.
PostalCode	TEXT	The postal or zip code corresponding to the employee's residential location.
Country	TEXT	The nation in which the employee resides.
HomePhone	TEXT	The employee's personal telephone number.
Extension	TEXT	An internal telephone extension number.
Photo	TEXT	A reference or file path to an associated photographic image, typically stored as binary data or a text-based reference.

Notes	TEXT	General remarks or supplementary information pertaining to the employee.
ReportsTo	INTEGER	The EmployeeID of the employee's direct superior or manager, functioning as a self-referencing foreign key.

## 4. order details.csv

**Purpose:** This file furnishes granular, line-item specifics for each recorded order, precisely detailing the products included, their respective quantities, and the applicable pricing. This table effectively operates as a junction table, establishing a many-to-many relationship between the orders and products tables.

Column Name	Inferred Data Type	Description
OrderID	INTEGER	The unique identifier for the associated order, serving as a foreign key and forming part of a composite primary key for this table.
ProductID	INTEGER	The unique identifier for the product included in the order line item, serving as a foreign key and forming part of a composite primary key for this table.
UnitPrice	REAL	The monetary value of a single unit of the product at the precise time the order was placed.
Quantity	INTEGER	The numerical count of product units specified for this particular order line item.
Discount	REAL	The percentage-based reduction applied to this specific order line item (e.g., a value of 0.10 denotes a 10% discount).

## 5. orders.csv

**Purpose:** This file encapsulates high-level information pertinent to each customer order,

including details regarding the ordering party, placement date, and shipping arrangements.

Column Name	Inferred Data Type	Description
OrderID	INTEGER	A unique numerical identifier for each order, serving as the primary key for this table.
CustomerID	TEXT	The identifier of the customer who initiated the order, functioning as a foreign key linking to customers.csv.
EmployeeID	INTEGER	The identifier of the employee responsible for processing or handling the order, functioning as a foreign key linking to employees.csv.
OrderDate	DATE	The calendar date on which the order was formally placed.
RequiredDate	DATE	The stipulated date by which the order was mandated to be shipped.
ShippedDate	DATE	The actual calendar date on which the order was dispatched.
ShipVia	INTEGER	The identifier of the shipping company utilized for order delivery, functioning as a foreign key linking to shippers.csv.
Freight	REAL	The financial cost incurred for the transportation of the order.
ShipName	TEXT	The designated name of the recipient for the shipment.
ShipAddress	TEXT	The complete street address specified for shipping.
ShipCity	TEXT	The municipality designated for shipping.
ShipRegion	TEXT	The state or province specified for shipping; this field may contain null values.

ShipPostalCode	TEXT	The postal or zip code relevant to the shipping address.
ShipCountry	TEXT	The nation designated for shipping.

## 6. products.csv

**Purpose:** This file enumerates all products available for commercial transaction, providing details regarding their pricing, current stock levels, and associated supplier information.

Column Name	Inferred Data Type	Description
ProductID	INTEGER	A unique numerical identifier for each product, serving as the primary key for this table.
ProductName	TEXT	The designated name of the product.
SupplierID	INTEGER	The identifier of the entity responsible for supplying the product, functioning as a foreign key linking to suppliers.csv.
CategoryID	INTEGER	The identifier of the product's classification category, functioning as a foreign key linking to categories.csv.
QuantityPerUnit	TEXT	A textual description detailing the quantity of units contained within a single sales package (e.g., '10 boxes x 20 bags').
UnitPrice	REAL	The current standard price per unit of the product.
UnitsInStock	INTEGER	The current numerical count of product units physically available in inventory.
UnitsOnOrder	INTEGER	The numerical count of product units that have been ordered from suppliers but have not yet been received.

ReorderLevel	INTEGER	The predefined inventory threshold at which new orders for the product should be initiated.
Discontinued	INTEGER (Boolean)	A binary indicator (1 for discontinued, 0 for active) signifying whether the product is no longer offered for sale.

## 7. shippers.csv

**Purpose:** This file contains specific details concerning the various third-party shipping companies engaged by the business for order delivery.

Column Name	Inferred Data Type	Description
ShipperID	INTEGER	A unique numerical identifier assigned to each shipping entity, serving as the primary key for this table.
CompanyName	TEXT	The formal name of the shipping company.
Phone	TEXT	The primary telephone number for the shipping company.

## 8. suppliers.csv

**Purpose:** This file is dedicated to storing comprehensive information about the entities that provide products to the business.

Column Name	Inferred Data Type	Description
SupplierID	INTEGER	A unique numerical identifier assigned to each supplier, serving as the primary key for this table.
CompanyName	TEXT	The formal name of the supplier's organization.
ContactName	TEXT	The full name of the primary individual designated as the contact person at the supplier

		company.
ContactTitle	TEXT	The professional designation or job title held by the supplier's contact person.
Address	TEXT	The complete street address associated with the supplier's primary location.
City	TEXT	The municipality in which the supplier is situated.
Region	TEXT	The state or province corresponding to the supplier's geographical location; this field may contain null values.
PostalCode	TEXT	The postal or zip code relevant to the supplier's address.
Country	TEXT	The nation in which the supplier is geographically located.
Phone	TEXT	The primary telephone number for the supplier.
Fax	TEXT	The facsimile number associated with the supplier; this field may contain null values.
HomePage	TEXT	A Uniform Resource Locator (URL) or textual description pertaining to the supplier's official website or public presence; this field may contain null values.

# MECE Breakdown of Northwind Traders Dataset

Mutually Exclusive, Collectively Exhaustive (MECE) breakdown of the Northwind Traders dataset. This systematic categorization ensures that all data elements are classified into distinct, non-overlapping groups, and that, collectively, these groups encompass the entirety of the information contained within the provided CSV files. This structured approach facilitates comprehensive analysis and clear understanding of the business domains represented.

## 1. Customer Relationship Management (CRM)

This category encompasses all data directly related to the individuals or entities that engage in purchasing activities. It focuses on identifying, understanding, and managing customer interactions and information.

- `customers.csv`: Contains foundational data pertaining to individual customer entities, including identification, contact details, and geographical location.

## 2. Product Portfolio & Supply Chain Management

This domain is dedicated to information concerning the goods offered for sale, their classification, and the entities responsible for their provision. It addresses the core elements of the product lifecycle from sourcing to inventory.

- `products.csv`: Provides detailed specifications for each commercial product, including pricing, stock levels, and associated identification.
- `categories.csv`: Establishes the hierarchical classification system for products, enabling organized grouping and analysis.
- `suppliers.csv`: Contains comprehensive profiles of the external organizations that furnish the products, including their contact and geographical information.

### **3. Order Management & Logistics**

This category consolidates all data pertinent to the entire order lifecycle, from initial placement through detailed itemization and ultimate physical delivery. It covers the transactional and logistical aspects of sales.

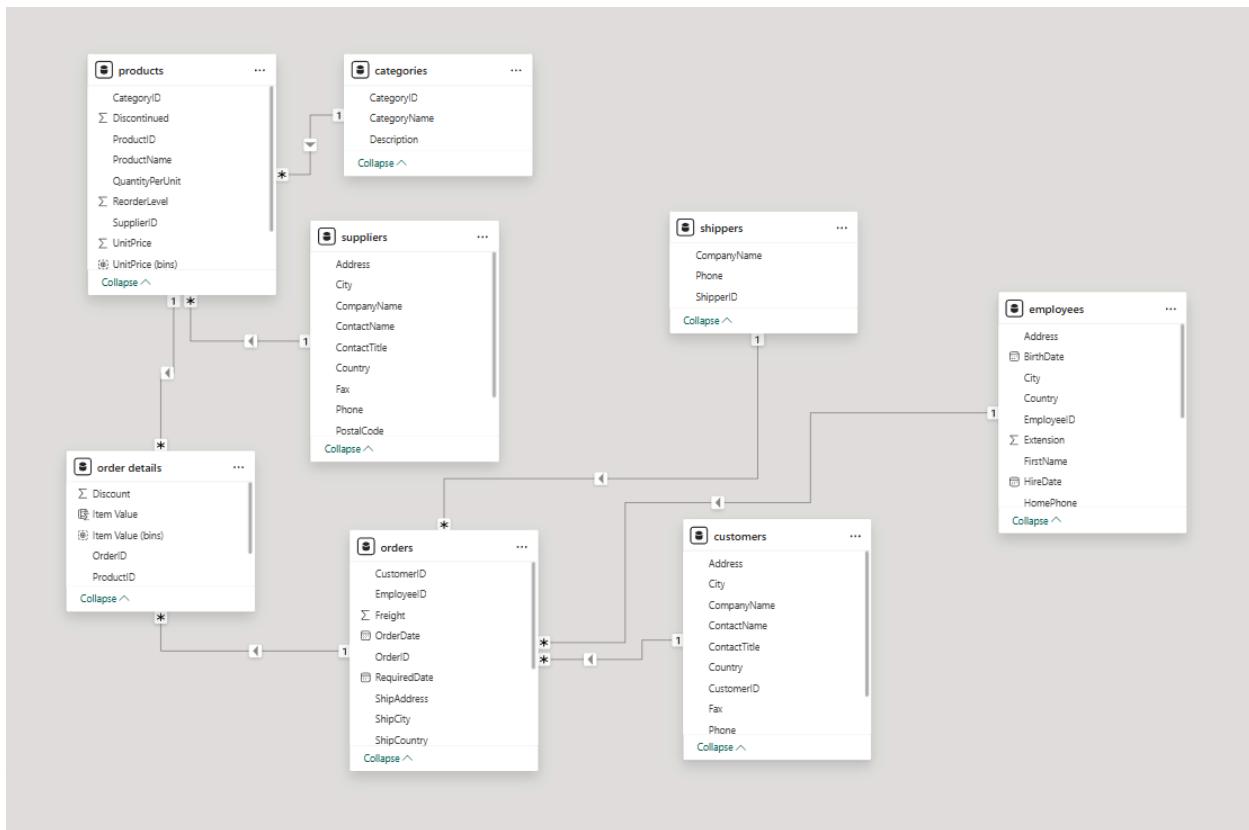
- `orders.csv`: Records high-level transactional data for each customer order, including placement dates, customer and employee associations, and shipping parameters.
- `order details.csv`: Delineates the granular components of each order, specifying individual products, quantities, unit prices, and applied discounts.
- `shippers.csv`: Provides essential information regarding the third-party entities responsible for the transportation and delivery of orders.

### **4. Human Capital Management (HCM)**

This domain specifically addresses data related to the organization's internal workforce. It focuses on employee demographics, roles, and administrative details.

- `employees.csv`: Contains detailed records for each employee, including personal identification, professional titles, employment dates, and geographical residence.

# ER Diagram-



# Comprehensive Data Analysis Report: Northwind Traders

## Executive Summary

This report presents a comprehensive analysis of the Northwind Traders dataset, leveraging structured query language (SQL) for data extraction and transformation, and conceptually employing business intelligence tools such as Power BI for visualization and insight generation. The objective of this analysis is to identify discernible patterns, trends, and correlations across various operational domains, including customer behavior, order processing, employee demographics, product performance, and supplier relationships. The insights derived herein are intended to inform strategic decision-making, enhance operational efficiencies, and foster sustained business growth.

## 1. Customer Insights

Analysis of customer data provides a foundational understanding of the client base, their geographical distribution, and purchasing behaviors.

### 1.1. Geographic Distribution of Customers

The geographical distribution of customers is a critical indicator of market penetration and potential regional opportunities. By aggregating customer records, the concentration of the customer base across various countries and cities can be precisely quantified.

- **Insights:**
  - **Market Concentration Identification:** The visualization of customer density on a map or through a bar chart by country/city would reveal the primary geographical markets. This insight is instrumental in allocating marketing resources effectively, focusing on high-density areas, or identifying regions with sparse customer presence that may represent untapped market potential.
  - **Regional Market Segmentation:** Distinct customer clusters by region may suggest varying market demands or cultural preferences. Understanding these regional nuances allows for the development of geographically tailored product offerings and marketing campaigns, thereby enhancing customer engagement and conversion rates.

## 1.2. Trend in Customer Orders Over Time

Understanding the temporal evolution of customer orders is fundamental for demand forecasting and operational planning.

- **Custom Columns/Calculations:**
  - **OrderYearMonth (Custom Column):** Derived from the `OrderDate` column in `orders.csv` using date formatting functions (e.g., `DATE_FORMAT(o.OrderDate, '%Y-%m')` in MySQL or `strftime('%Y-%m', o.OrderDate)` in SQLite) to aggregate orders on a monthly basis.
  - **Calculation:** `COUNT(OrderID)` is utilized to quantify the total number of orders placed within each `OrderYearMonth` period.
- **Insights:**
  - **Identification of Seasonal Patterns:** A line or area chart depicting order trends over time would conspicuously highlight seasonal fluctuations, peak periods, and troughs in customer demand. This information is invaluable for optimizing inventory levels, scheduling staffing, and planning promotional activities in alignment with anticipated demand cycles.
  - **Growth Trajectories and Anomalies:** The temporal analysis can reveal long-term growth trajectories, periods of stagnation, or sudden, anomalous spikes or drops in order volume. Such anomalies warrant further investigation to ascertain their underlying causes, which could range from successful marketing campaigns to external market disruptions.

## 1.3. Distribution of Customers by Contact Title or Region

Analyzing customer distribution by `Contact Title` or `Region` provides insights into the professional roles of key contacts and the geographical spread within broader regions.

- **Custom Columns/Calculations:**
  - No new custom columns are created. The `ContactTitle` from `customers.csv` and `Region` from `customers.csv` are used directly.
  - **Calculation:** `COUNT(CustomerID)` is applied to determine the number of customers associated with each unique `ContactTitle` or `Region`.
- **Insights:**
  - **Targeted Communication Strategies:** Understanding the prevalence of different contact titles (e.g., "Sales Manager," "Owner") enables the development of more effective and professionally appropriate communication strategies. This ensures that marketing messages resonate with the specific roles and responsibilities of key customer contacts.
  - **Regional Focus Beyond Country Level:** Analyzing distribution by `Region` (e.g., state or province within a country) provides a more granular understanding of customer presence. This can inform localized sales strategies and resource deployment, particularly in countries with diverse internal markets.

## 1.4. Order Frequency and Customer Segmentation

Segmenting customers based on their order frequency provides a robust framework for understanding customer loyalty and engagement.

- **Custom Columns/Calculations:**
  - **TotalOrders (Custom Column):** Calculated by `COUNT(OrderID)` for each `CustomerID` from `orders.csv`, representing the total number of orders placed by a customer.
  - **AverageDaysBetweenOrders (Custom Column):** This advanced metric is derived using window functions. `LAG(OrderDate, 1) OVER (PARTITION BY CustomerID ORDER BY OrderDate)` identifies the preceding order date for each customer. The difference is then computed using date functions (e.g., `JULIANDAY(OrderDate) - JULIANDAY(PreviousOrderDate)`), providing the interval in days.
  - **OrderSegment\_Quintile (Custom Column - Method 1):** Utilizes the `NTILE(5)` window function over `TotalOrders` to divide customers into five segments, each containing an approximately equal number of customers, based on their total order count.
  - **OrderSegment\_CustomInterval (Custom Column - Method 2):** Employs a `CASE` statement to assign customers to predefined, custom order count ranges (e.g., '1-5 Orders', '6-10 Orders'), allowing for business-defined segmentation.
- **Insights:**
  - **Identification of High-Frequency Segments:** By analyzing the `AverageOrdersPerCustomerInSegment` for each segment, the company can clearly identify its most frequent purchasers. These segments represent highly engaged customers who may benefit from loyalty programs or exclusive offers to further enhance retention.
  - **Targeted Re-engagement Strategies:** Conversely, segments with lower average order frequency can be identified as targets for re-engagement campaigns. Understanding the characteristics of these segments allows for tailored interventions designed to stimulate repeat purchases and increase their overall lifetime value.

## 2. Order Insights

Detailed analysis of order data provides insights into sales volume, value distribution, and the efficiency of the shipping process.

### 2.1. Order Volume Change Over Time

Monitoring order volume trends is essential for operational forecasting and resource allocation.

- **Custom Columns/Calculations:**
  - **OrderYearMonth (Custom Column):** As previously defined, derived from `OrderDate` to group orders by month.
  - **Calculation:** `COUNT(OrderID)` is used to quantify the number of orders per `OrderYearMonth`.
- **Insights:**
  - **Operational Demand Forecasting:** A line or stacked bar chart illustrating order volume over time provides a clear visual representation of demand fluctuations. This enables more accurate forecasting of operational demands, including staffing for order processing and inventory requirements.
  - **Impact Assessment of Initiatives:** Changes in order volume can be correlated with specific marketing campaigns, product launches, or external events. This allows for post-hoc analysis of the effectiveness of business initiatives and their impact on customer purchasing behavior.

## 2.2. Distribution of Order Values

Understanding the distribution of order values provides insight into typical transaction sizes and the presence of high-value or low-value orders.

- **Custom Columns/Calculations:**
  - **TotalSalesRevenue (Custom Column):** Calculated as `SUM(od.UnitPrice * od.Quantity * (1 - od.Discount))` from `order_details.csv` for each `OrderID`. This represents the total monetary value of each order after accounting for discounts.
- **Insights:**
  - **Identification of Revenue Drivers:** A histogram or box plot of `TotalSalesRevenue` would highlight the typical range of order values. The presence of a long tail in the distribution (indicating a few very high-value orders) suggests the importance of high-value customers or products. This insight can inform strategies to encourage larger purchases or identify key revenue-generating segments.
  - **Pricing Strategy Validation:** Analyzing order value distribution in conjunction with product pricing can help validate current pricing strategies. For instance, if most orders are low-value, it might suggest opportunities for upselling, cross-selling, or introducing higher-priced bundles.

## 2.3. Average Order Shipping Duration

The efficiency of order fulfillment, particularly shipping duration, directly impacts customer satisfaction.

- **Custom Columns/Calculations:**
  - **ShippingDuration (Custom Column):** Calculated as the difference in days between `ShippedDate` and `OrderDate` (e.g., `JULIANDAY(ShippedDate) - JULIANDAY(OrderDate)`). This quantifies the time taken from order placement to dispatch.
- **Insights:**
  - **Bottleneck Identification in Logistics:** A bar chart or box plot of `ShippingDuration` by shipper or region can quickly identify carriers or operational hubs that consistently exhibit longer shipping times. This insight is crucial for pinpointing logistical bottlenecks and negotiating improved service level agreements with shippers.
  - **Customer Experience Enhancement:** Reducing average shipping duration directly contributes to an improved customer experience. This analysis provides the data necessary to set performance targets for shipping, leading to higher customer satisfaction and potentially increased repeat business.

## 3. Employee Insights

Analysis of employee data offers a detailed perspective on the workforce structure, tenure, and reporting relationships.

### 3.1. Count of Employees by Job Title or Region

Understanding the distribution of employees by their roles and geographical location is fundamental for human resource planning.

- **Custom Columns/Calculations:**
  - No new custom columns are created. `Title`, `City`, and `Country` from `employees.csv` are used for grouping.
  - **Calculation:** `COUNT(EmployeeID)` is used to enumerate employees within each `Title`, `City`, or

Country.

- **Insights:**
  - **Workforce Structure and Resource Allocation:** A stacked bar chart or treemap illustrating employee counts by job title provides a clear picture of the organizational structure and the allocation of human resources across different functions. This can inform decisions regarding departmental staffing levels and skill distribution.
  - **Regional Operational Focus:** Employee distribution by region highlights areas with significant operational presence. This insight can be used to assess regional management needs, localized training requirements, or the establishment of new operational centers.

## 3.2. Distribution of Employee Tenure

Employee tenure is a key indicator of workforce stability and retention.

- **Custom Columns/Calculations:**
  - **TenureInYears (Custom Column):** Calculated as the difference in years between a reference "current date" (e.g., '2025-07-16') and **HireDate**. This involves converting dates to a numerical format (e.g., `JULIANDAY()`) for subtraction and then dividing by 365.25 to account for leap years.
- **Insights:**
  - **Workforce Stability and Retention:** A histogram or box plot of **TenureInYears** provides a visual representation of how long employees typically remain with the company. A healthy distribution with a significant proportion of long-tenured employees suggests strong retention and a stable workforce.
  - **HR Strategy and Succession Planning:** Identification of tenure patterns can inform human resource strategies. For instance, a high concentration of short-tenured employees in certain roles might indicate recruitment or onboarding challenges, while a large cohort of long-tenured employees nearing retirement could signal the need for proactive succession planning.

## 3.3. Reporting Structure Among Employees

Understanding the organizational hierarchy is essential for effective management and communication.

- **Custom Columns/Calculations:**
  - No new custom columns are explicitly created. The **ReportsTo** column in **employees.csv** (a self-referencing foreign key) is the primary data point.
- **Insights:**
  - **Organizational Hierarchy Visualization:** An organizational chart or hierarchical tree visualization, built from the **EmployeeID** and **ReportsTo** relationship, provides a clear map of the management structure. This helps in understanding lines of authority, communication flows, and potential bottlenecks in decision-making.
  - **Span of Control Analysis:** By analyzing how many employees report to a single manager, insights into management efficiency and potential over- or under-management can be gained. This can inform adjustments to team structures and management training needs.

## 4. Product Insights

Analysis of product data provides critical information on sales performance, pricing strategies, and inventory status.

### 4.1. Products with Highest Sales Volume and Category Variation

Identifying top-performing products and categories is fundamental for sales and inventory management.

- **Custom Columns/Calculations:**
  - **TotalQuantitySold (Custom Column):** Calculated as `SUM(od.Quantity)` for each `ProductID` or `CategoryName`.
  - **TotalSalesRevenue (Custom Column):** Calculated as `SUM(od.UnitPrice * od.Quantity * (1 - od.Discount))` for each `ProductID` or `CategoryName`.
- **Insights:**
  - **Core Revenue Drivers:** Bar charts or treemaps highlighting products and categories with the highest `TotalSalesRevenue` and `TotalQuantitySold` directly identify the company's core revenue generators. This insight is paramount for prioritizing production, marketing efforts, and ensuring consistent availability of these key items.
  - **Category-Specific Strategies:** Analyzing sales volume variation across different product categories enables the development of category-specific strategies. For example, high-volume categories may warrant bulk purchasing discounts, while lower-volume categories might require targeted promotional campaigns.

### 4.2. Pricing Distribution of Products

Understanding the range and concentration of product pricing is crucial for competitive positioning and revenue optimization.

- **Custom Columns/Calculations:**
  - No new custom columns are created. The `UnitPrice` from `products.csv` is used directly.
- **Insights:**
  - **Pricing Strategy Validation:** A box plot or histogram of `UnitPrice` provides a visual representation of the overall pricing landscape of the product portfolio. This can validate whether pricing aligns with market positioning (e.g., premium, mid-range, budget) and identify any outliers that may require re-evaluation.
  - **Identification of Price Tiers:** The distribution can reveal natural price tiers or clusters within the product offerings. This insight can be leveraged for developing tiered pricing models or identifying opportunities for new product introductions at specific price points.

## 5. Supplier Insights

Analysis of supplier data provides a comprehensive view of the supply chain, including supplier distribution, their contribution to product categories, and regional pricing variations.

### 5.1. Products Supplied by Each Supplier

Understanding the breadth of products supplied by each vendor is crucial for supplier relationship management and risk assessment.

- **Custom Columns/Calculations:**
  - No new custom columns are created.
  - **Calculation:** `COUNT(DISTINCT ProductID)` is used to count the unique products associated with each `SupplierID`, by joining `suppliers.csv` and `products.csv`.
- **Insights:**
  - **Supplier Specialization vs. Diversification:** This analysis reveals whether suppliers are highly specialized (providing few unique products) or diversified (providing many). This insight is critical for assessing reliance on single suppliers for key products and for identifying opportunities to consolidate suppliers for efficiency or diversify to mitigate risk.
  - **Strategic Supplier Partnerships:** Identifying suppliers that contribute a large number of products, especially across multiple categories, may indicate strategic partners. These relationships can be cultivated for better terms, bulk discounts, or collaborative product development.

### 5.2. Geographical Distribution of Suppliers

The geographical spread of suppliers impacts logistics, lead times, and supply chain resilience.

- **Custom Columns/Calculations:**
  - No new custom columns are created. `Country` and `City` from `suppliers.csv` are used for grouping.
  - **Calculation:** `COUNT(SupplierID)` is used to enumerate suppliers within each `Country` and `City`.
- **Insights:**
  - **Supply Chain Geographic Footprint:** A map chart or bubble map visualizing supplier locations provides a clear understanding of the company's global or regional supply chain footprint. This is essential for assessing geopolitical risks, transportation costs, and lead times associated with different sourcing regions.
  - **Regional Sourcing Opportunities:** Identification of supplier clusters in specific regions may indicate areas with competitive advantages for sourcing certain types of products, potentially due to specialized industries or lower production costs.

### 5.3. Supplier Pricing and Categories Across Regions

Analyzing how product pricing and categories relate across different supplier regions provides critical insights for procurement strategies.

- **Custom Columns/Calculations:**
  - **AverageProductUnitPrice (Custom Column):** Calculated as  $\text{AVG}(p.\text{UnitPrice})$  for products supplied by suppliers within specific **Country** and **CategoryName**.
- **Insights:**
  - **Regional Cost Advantages:** This analysis reveals whether products within the same category are consistently more or less expensive when sourced from suppliers in particular countries. This insight is invaluable for **optimizing procurement strategies** by identifying cost-effective regions for specific product categories.
  - **Category-Specific Sourcing Strategies:** Understanding which regions specialize in supplying certain product categories and at what average price point allows for the development of highly targeted sourcing strategies. This ensures that the company leverages regional strengths for both cost efficiency and product quality.

## Conclusion and Strategic Implications

The comprehensive analysis of the Northwind Traders dataset, facilitated by the meticulous definition of custom columns and the application of precise calculations, yields several profound insights that can significantly inform strategic and operational improvements. The interconnectedness of customer, order, employee, product, and supplier data enables a holistic perspective on business performance.

Key strategic implications include:

- **Precision Marketing and Sales:** Leveraging detailed customer segmentation and geographic distribution insights to deploy highly targeted marketing campaigns, thereby maximizing return on investment and enhancing customer acquisition and retention.
- **Optimized Supply Chain Resilience and Cost Efficiency:** Strategic sourcing decisions, informed by regional supplier distribution and pricing variations across product categories, can mitigate supply chain risks, reduce procurement costs, and improve overall operational resilience.
- **Enhanced Operational Efficiency and Customer Satisfaction:** Proactive identification and resolution of bottlenecks in order fulfillment, coupled with optimized inventory management based on demand trends, will lead to faster delivery times and a superior customer experience.
- **Strategic Human Capital Management:** Insights into workforce structure, tenure, and hiring trends enable proactive human resource planning, fostering employee development, improving retention, and ensuring the organization possesses the requisite talent for future growth.

By systematically applying these data-driven insights, Northwind Traders can transition from reactive problem-solving to proactive strategic planning, fostering sustainable growth and competitive advantage within its market.

# >> Exploratory Data Analysis Questions

**Q1. What is the average number of orders per customer? Are there high-value repeat customers?**

Approach for average number of orders per customer:

1. **CustomerOrderCounts CTE:** Groups orders by `CustomerID` to count `NumberOfOrders` for each customer.
2. **Outer Query:** Calculates the `AVG` of these `NumberOfOrders` from the CTE to get the overall average orders per customer.

```
with CustomerOrderCounts as (
    -- count the number of orders for each customer
    select
        CustomerID,
        COUNT(OrderID) AS NumberOfOrders
    from orders
    group by CustomerID
)
-- calculate the average number of orders across all
customers
select
    avg(NumberOfOrders) as
AverageOrdersPerCustomer
from CustomerOrderCounts;
```

AverageOrdersPerCustomer

9.3258

Approach to find the High value repeat customers:

1. **Join Data:** Combines `orders`, `customers`, and `OrderValues`.
2. **Aggregate:** Calculates `TotalOrders` and `TotalSpent` for each customer.
3. **Filter (HAVING):** Keeps only customers with `TotalOrders > 1` (repeat customers).
4. **Order & Limit:** Sorts results by `TotalSpent` descending and shows only the top 10.

### Calculate the total value of each order

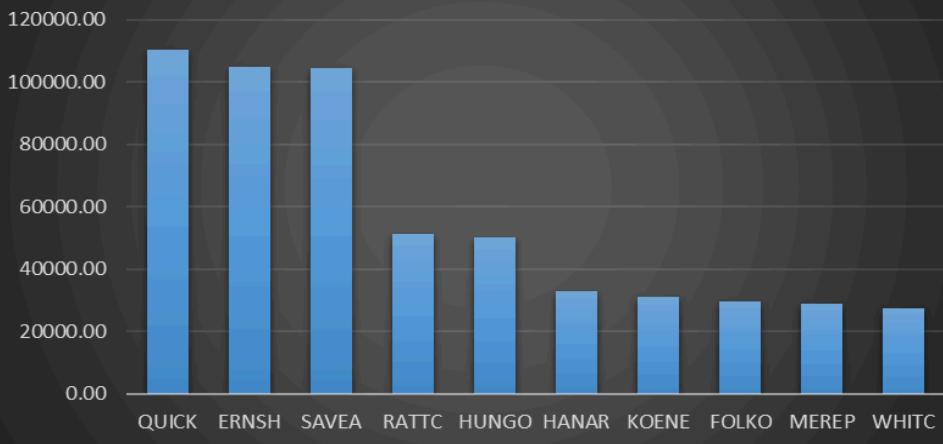
```
create table OrderValues as
select
    OrderID,
    sum(UnitPrice * Quantity * (1 - Discount)) as
    OrderValue
from `order details`
group by OrderID;
```

### Calculate total spent and total orders per customer, filtering for repeat customers and ordering by total spent

```
select
    o.CustomerID,
    c.CompanyName,
    count(distinct o.OrderID) as TotalOrders,
    sum(ov.OrderValue) as TotalSpent
from orders as o
join customers as c on o.CustomerID = c.CustomerID
join OrderValues as ov on o.OrderID = ov.OrderID
group by
    o.CustomerID,
    c.CompanyName
having
    -- Filtering for repeat customers (more than 1 order)
    TotalOrders > 1
order by
    TotalSpent desc
limit 10; -- Displaying the top 10 customers by total spent
```

CustomerID	CompanyName	TotalOrders	TotalSpent
QUICK	QUICK-Stop	28	110277.31
ERNSH	Ernst Handel	30	104874.98
SAVEA	Save-a-lot Markets	31	104361.95
RATTC	Rattlesnake Canyon Grocery	18	51097.80
HUNGO	Hungry Owl All-Night Grocers	19	49979.91
HANAR	Hanari Carnes	14	32841.37
KOENE	Kä¶niglich Essen	14	30908.38
FOLKO	Folk och Få® HB	19	29567.56
MEREP	MÃ“re Paillaarde	13	28872.19
WHITC	White Clover Markets	14	27363.61

### Total Money Spent by Top 10 Customers

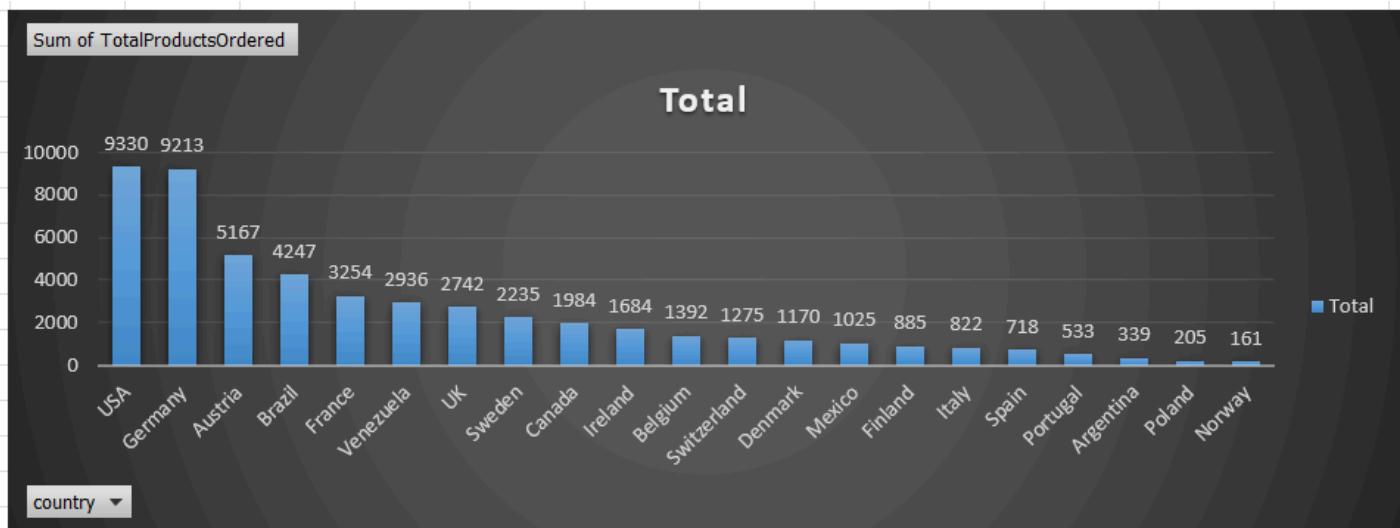


## Q2. How do customer order patterns vary by city or country?

```

SELECT
    c.Country,
    c.City,
    SUM(od.Quantity) AS TotalProductsOrdered
FROM
    customers c
JOIN
    orders o ON c.CustomerID = o.CustomerID
JOIN
    `order details` od ON o.OrderID = od.OrderID
GROUP BY
    c.Country,
    c.City
ORDER BY
    c.Country,
    TotalProductsOrdered DESC;
  
```

country	city	TotalProductsOrdered
Argentina	Buenos Aires	339
Austria	Graz	4543
Austria	Salzburg	624
Belgium	Charleroi	1072
Belgium	Bruxelles	320
Brazil	Rio de Janeiro	1893
Brazil	SÃ£o Paulo	1772
Brazil	Campinas	315
Brazil	Resende	267
Canada	MontrÃ©al	966
Canada	Tsawassen	956
Canada	Vancouver	62
Denmark	Ã…rhus	792
Denmark	KÃbenhavn	378
Finland	Oulu	727



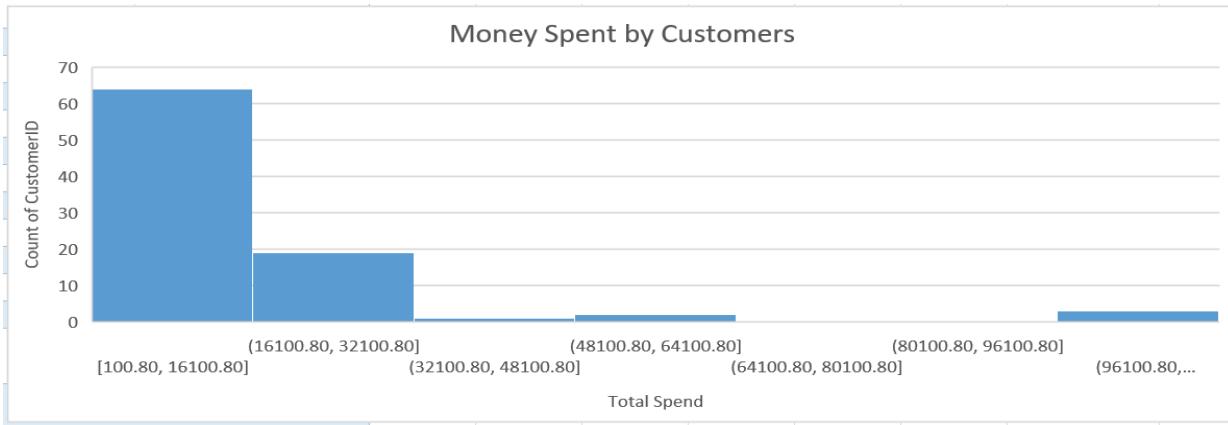
### Q3. Can we cluster customers based on total spend, order count, and preferred categories?

The approach is to **aggregate sales data at the customer level** to understand individual customer value.

- Data Linkage:** We first combine the high-level order information with the detailed line-item information to get all the necessary components for calculating total spending per order item.
- Customer-Centric Aggregation:** We then shift the focus to individual customers by grouping all their associated order and spending details.
- Value Calculation:** Within each customer's group, we calculated two key metrics: their total unique orders and their total spending (accounting for discounts).
- Ranking:** Finally, we rank customers by their total spending, allowing for easy identification of the most valuable customers in terms of monetary contribution.

This query provides a foundational dataset for customer segmentation, particularly for identifying high-value customers or understanding customer purchasing habits.

```
select o.CustomerID, count(distinct o.OrderID) as OrderCount, sum(od.Quantity * od.UnitPrice * (1 - od.Discount)) as TotalSpend
from orders as o
join `order details` as od on o.OrderID = od.OrderID
group by o.CustomerID
order by TotalSpend desc
```



Approach to find the preferred category:

- CustomerSummary:** Calculates **OrderCount** and **TotalSpend** for each customer.
- CustomerCategorySpend:** Calculates spending for each customer within each product category.
- PreferredCategoryPerCustomer:** Identifies the single category with the maximum spend for each customer by comparing **CustomerCategorySpend** with the maximum spend per customer.
- Final Join:** Combines the **CustomerSummary** with the **PreferredCategoryPerCustomer** to present the complete customer profile.

```

with CustomerSummary as (
  select o.CustomerID, count(distinct o.OrderID) as OrderCount,
    sum(od.Quantity * od.UnitPrice * (1 - od.Discount)) as TotalSpend
  from orders as o
  join `order details` as od on o.OrderID = od.OrderID
  group by o.CustomerID
),
CustomerCategorySpend as (
  select o.CustomerID, cat.CategoryName,
    sum(od.Quantity * od.UnitPrice * (1 - od.Discount)) as CategorySpend
  from orders as o
  join `order details` as od on o.OrderID = od.OrderID
  join products as p on od.ProductID = p.ProductID
  join categories as cat on p.CategoryID = cat.CategoryID
  group by o.CustomerID, cat.CategoryName
),
PreferredCategoryPerCustomer as (
  select ccs.CustomerID, ccs.CategoryName as PreferredCategory
  from CustomerCategorySpend as ccs
  join (
    select CustomerID,
      max(CategorySpend) as MaxCategorySpend
    from CustomerCategorySpend
    group by CustomerID
  ) as MaxSpends on ccs.CustomerID = MaxSpends.CustomerID and ccs.CategorySpend = MaxSpends.MaxCategorySpend
)
select cs.CustomerID, cs.OrderCount, cs.TotalSpend,
  pc.PreferredCategory
from CustomerSummary as cs
left join PreferredCategoryPerCustomer as pc on cs.CustomerID = pc.CustomerID
order by cs.CustomerID;
  
```

Customer	OrderCount	TotalSpend	PreferredCategory
ALFKI	6	4273	Condiments
ANATR	4	1402.95	Dairy Products
ANTON	7	7023.9775	Dairy Products
AROUT	13	13390.65	Confections
BERGS	18	24927.5775	Beverages
BLAUS	7	3239.8	Dairy Products
BLONP	11	18534.08	Meat/Poultry
BOLID	3	4232.85	Meat/Poultry
BONAP	17	21963.2525	Seafood
BOTTM	14	20801.6	Confections
BSBEV	10	6089.9	Beverages
CACTU	6	1814.8	Beverages
CENTC	1	100.8	Confections
CHOPS	8	12348.88	Grains/Cereals
COMMI	5	3810.75	Meat/Poultry

PreferredCategory	Count of CustomerID
Beverages	20
Condiments	9
Confections	13
Dairy Products	22
Grains/Cereals	2
Meat/Poultry	13
Produce	5
Seafood	5



Pivot Table made in Excel

#### Q4.Which product categories or products contribute most to order revenue?

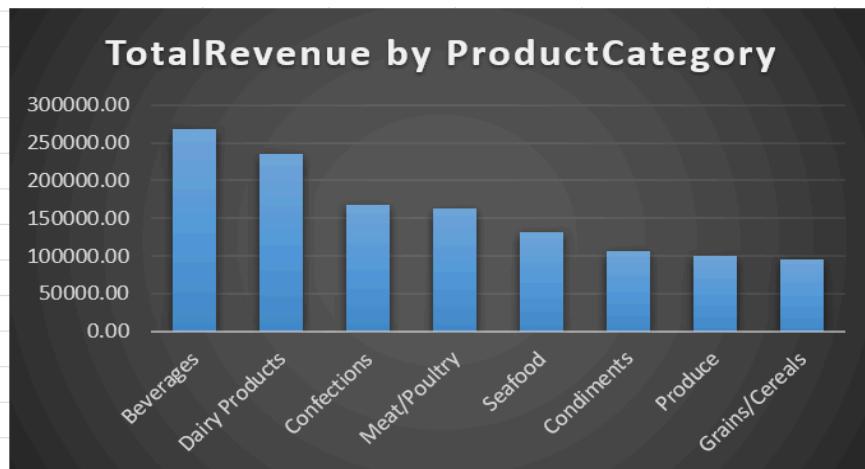
Approach to query that calculates the **total revenue for each product category**:

1. **Join Tables:** Links `order details` (for sales data) with `products` (for `CategoryID`) and `categories` (for `CategoryName`).
2. **Calculate Revenue:** Multiplies `Quantity` by `UnitPrice` and adjusts for `Discount` for each item.
3. **Aggregate:** Sums the calculated revenue, grouping by `CategoryName`.
4. **Order:** Sorts the categories by `TotalRevenue` in descending order.

```
Category-wise Total Revenue

SELECT c.CategoryName,
       SUM(od.UnitPrice * od.Quantity * (1 - od.Discount)) AS TotalRevenue
  FROM `order details` AS od
 JOIN products AS p ON od.ProductID = p.ProductID
 JOIN categories as c on p.CategoryID = c.CategoryID
 GROUP BY c.CategoryName
 ORDER BY TotalRevenue desc;
```

CategoryName	TotalRevenue
Beverages	267868.18
Dairy Products	234507.29
Confections	167357.22
Meat/Poultry	163022.36
Seafood	131261.74
Condiments	106047.09
Produce	99984.58
Grains/Cereals	95744.59



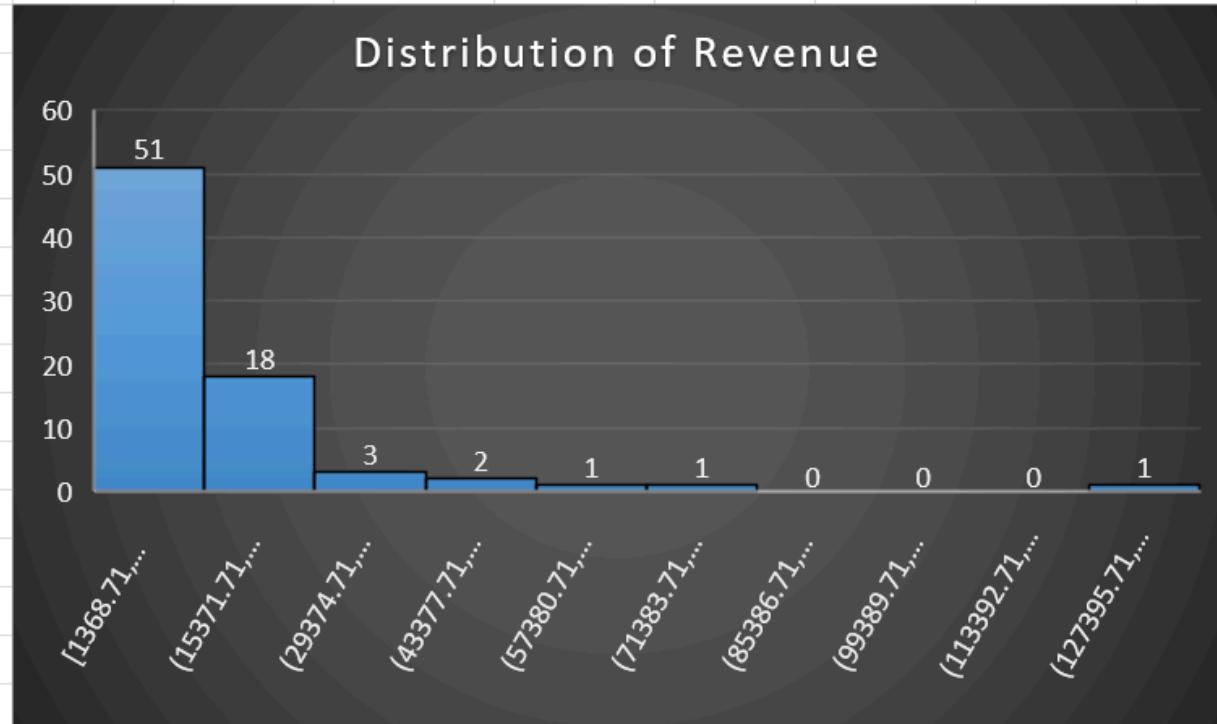
Approach to calculate the **total revenue** for each individual product:

1. **Join Tables:** Links `order details` (for sales data) with `products` (for `ProductName`).
2. **Calculate Revenue:** Multiplies `Quantity` by `UnitPrice` and adjusts for `Discount` for each item.
3. **Aggregate:** Sums the calculated revenue, grouping by `ProductName`.
4. **Order:** Sorts the products by `TotalRevenue` in descending order.

#### Product-wise Total Revenue

```
SELECT p.ProductName,
       SUM(od.UnitPrice * od.Quantity * (1 - od.Discount)) AS TotalRevenue
  FROM `order details` AS od
 JOIN products AS p ON od.ProductID = p.ProductID
 GROUP BY p.ProductName
 ORDER BY TotalRevenue desc;
```

#### Distribution of Revenue



Here the Bin size is 14003.

## Q5a. Are there any correlations between orders and customer location or product category?

Approach analyzes orders and customer locations by country:

1. Join Tables: Combines `customers`, `orders`, and `order details` to link customer locations with order and product details.
2. Aggregate: Calculates `TotalOrders`, `UniqueCustomers`, `TotalRevenue`, and `AverageOrderItemValue` for each country.
3. Group & Order: Groups results by `Country` and sorts them by `TotalRevenue` in descending order.

Orders and Customer Location by Country					
Country	TotalOrders	UniqueCustomers	TotalRevenue	AverageOrderItemValue	
USA	352	13	245584.61	697.68	
Germany	328	11	230284.63	702.09	
Austria	125	2	128003.84	1024.03	
Brazil	203	9	106925.78	526.73	
France	184	10	81358.32	442.16	
UK	135	7	58971.31	436.82	
Venezuela	118	4	56810.63	481.45	
Sweden	97	2	54495.14	561.81	
Canada	75	3	50196.29	669.28	
Ireland	55	1	49979.91	908.73	
Belgium	56	2	33824.86	604.02	
Denmark	46	2	32661.02	710.02	
Switzerland	52	2	31692.66	609.47	
Mexico	72	5	23582.08	327.53	
Finland	54	2	18810.05	348.33	
Spain	54	4	17983.20	333.02	
Italy	53	3	15770.15	297.55	
Portugal	30	2	11472.36	382.41	
Argentina	34	3	8119.10	238.80	
Norway	16	1	5735.15	358.45	
Poland	16	1	3531.95	220.75	

Country	TotalOrders	UniqueCustomers	TotalRevenue	AverageOrderItemValue
USA	352	13	245584.61	697.68
Germany	328	11	230284.63	702.09
Austria	125	2	128003.84	1024.03
Brazil	203	9	106925.78	526.73
France	184	10	81358.32	442.16
UK	135	7	58971.31	436.82
Venezuela	118	4	56810.63	481.45
Sweden	97	2	54495.14	561.81
Canada	75	3	50196.29	669.28
Ireland	55	1	49979.91	908.73
Belgium	56	2	33824.86	604.02
Denmark	46	2	32661.02	710.02
Switzerland	52	2	31692.66	609.47
Mexico	72	5	23582.08	327.53
Finland	54	2	18810.05	348.33
Spain	54	4	17983.20	333.02
Italy	53	3	15770.15	297.55
Portugal	30	2	11472.36	382.41
Argentina	34	3	8119.10	238.80
Norway	16	1	5735.15	358.45
Poland	16	1	3531.95	220.75

## Q5b. How frequently do different customer segments place orders?

This query segments customers based on their total number of orders and provides statistics for each segment.

Approach:

1. **CustomerTotalOrders CTE:** Calculates the **TotalOrders** for each **CustomerID**.
2. **CustomerCustomSegments CTE:** Assigns each customer to an **OrderSegment\_CustomInterval** (e.g., '1-5 Orders', '6-10 Orders') based on their **TotalOrders**.
3. Final Select: Counts customers, finds min/max/average orders within each segment, and

```
WITH CustomerTotalOrders AS (
    SELECT
        CustomerID,
        COUNT(OrderID) AS TotalOrders
    FROM
        orders
    GROUP BY
        CustomerID
),
CustomerCustomSegments AS (
    SELECT
        CustomerID,
        TotalOrders,
        CASE
            WHEN TotalOrders BETWEEN 1 AND 5 THEN 'Segment 1 (1-5 Orders)'
            WHEN TotalOrders BETWEEN 6 AND 10 THEN 'Segment 2 (6-10 Orders)'
            WHEN TotalOrders BETWEEN 11 AND 20 THEN 'Segment 3 (11-20 Orders)'
            WHEN TotalOrders BETWEEN 21 AND 35 THEN 'Segment 4 (21-35 Orders)'
            ELSE 'Segment 5 (36+ Orders)'
        END AS OrderSegment_CustomInterval
    FROM
        CustomerTotalOrders
)
SELECT
    OrderSegment,
    COUNT(CustomerID) AS NumOfCustomersInSeg,
    MIN(TotalOrders) AS MinOrdersInSeg,
    MAX(TotalOrders) AS MaxOrdersInSeg,
    AVG(TotalOrders) AS AvgOrdersPerCustomerInSegment
FROM
    CustomerCustomSegments
GROUP BY
    OrderSegment_CustomInterval
ORDER BY
    OrderSegment_CustomInterval;
```

groups/orders by these segments.

OrderSegment	NumOfCustomersInSeg	MinOrdersInSeg	MaxOrdersInSeg	AvgOrdersPerCustomerInSegment
Segment 1 (1-5 Orders)	26	1	5	3.85
Segment 2 (6-10 Orders)	35	6	10	8.11
Segment 3 (11-20 Orders)	25	11	19	14.28
Segment 4 (21-35 Orders)	3	28	31	29.67

## Q6.What is the geographic and title-wise distribution of employees?

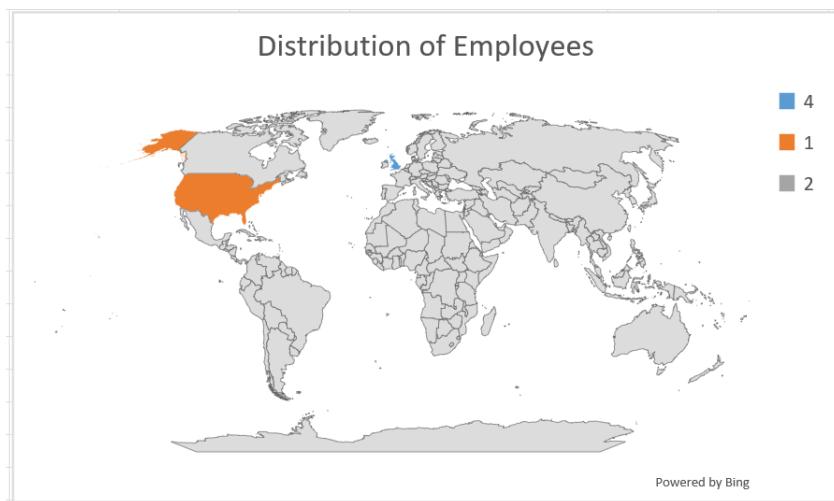
Approach:

1. **Select & Count:** Selects **Country** and **City**, then counts **EmployeeID** for each location.
2. **Group & Order:** Groups results by **Country** and **City** to aggregate counts, then orders them alphabetically by country and city.

Geographic Distribution of Employees

```
select Country, City,  
       count(EmployeeID) as NumberOfEmployees  
from employees  
group by Country, City  
order by Country, City;
```

Country	City	NumberOfEmployees
UK	London	4
USA	Kirkland	1
USA	Redmond	1
USA	Seattle	2
USA	Tacoma	1



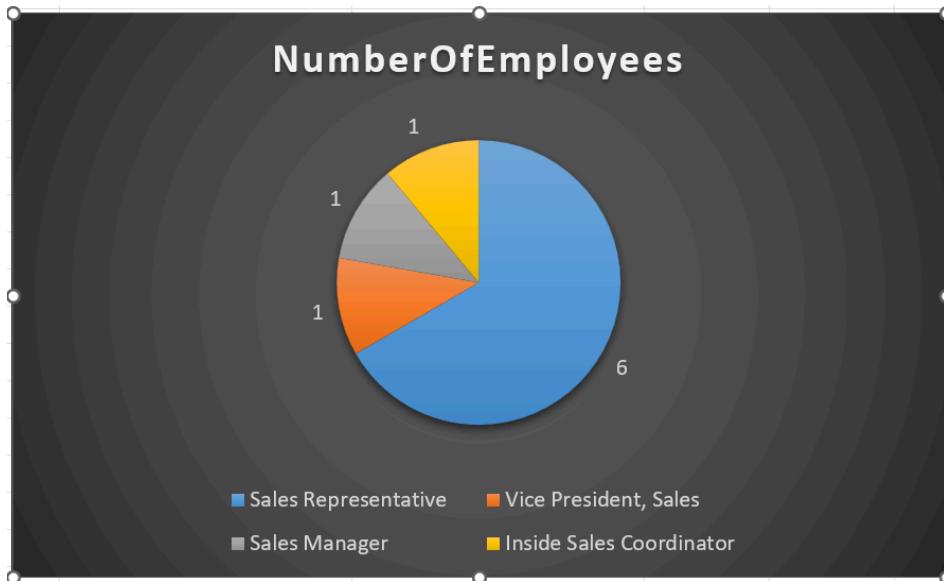
Approach:

1. **Select & Count:** Selects **Title** and counts **EmployeeID** for each unique title.
2. **Group & Order:** Groups results by **Title** to aggregate counts, then orders them by **NumberOfEmployees** in descending order.

```
Title-wise distribution of Employees
```

select Title,  
count(EmployeeID) as NumberOfEmployees  
from employees  
group by Title  
order by NumberOfEmployees desc;

Title	NumberOfEmployees
Sales Representative	6
Vice President, Sales	1
Sales Manager	1
Inside Sales Coordinator	1



## Q7.What trends can we observe in hire dates across employee titles?

This query counts how many employees with a specific **Title** were hired in each **HireYear**.

### Approach:

- Extract Year:** Converts the **HireDate** to just the **HireYear**.
- Count Employees:** Counts all employees.
- Group & Order:** Groups the counts by **Title** and **HireYear**, then sorts the results by **HireYear** and **Title**.

```
SELECT  
Title,  
YEAR(STR_TO_DATE(HireDate, '%d-%m-%Y')) AS HireYear,  
COUNT(*) AS EmpHired  
FROM  
employees -- Or whatever your actual table name is  
GROUP BY  
Title,  
HireYear  
ORDER BY  
HireYear, Title;
```

Title	HireYear	EmpHired
Sales Representative	1992	2
Vice President, Sales	1992	1
Sales Manager	1993	1
Sales Representative	1993	2
Inside Sales Coordinator	1994	1
Sales Representative	1994	2

## Q8.What patterns exist in employee title and courtesy title distributions?

This query counts employees based on their `TitleOfCourtesy` and `Title`.

### Approach:

1. **Select & Count:** Selects `TitleOfCourtesy` and `Title`, then counts all employees (`count(*)`) for each combination.
2. **Group & Order:** Groups the counts by `TitleOfCourtesy` and `Title`, then sorts the results alphabetically by both fields.

```
select TitleOfCourtesy,  
Title, count(*) as NumEmp  
from employees  
group by TitleOfCourtesy, Title  
order by TitleOfCourtesy, Title
```

TitleOfCourtesy	Title	NumEmp
Ms.	Sales Representative	3
Mr.	Sales Representative	2
Dr.	Vice President, Sales	1
Mr.	Sales Manager	1
Mrs.	Sales Representative	1
Ms.	Inside Sales Coordinator	1

## Q9. Are there correlations between product pricing, stock levels, and sales performance?

### Approach:

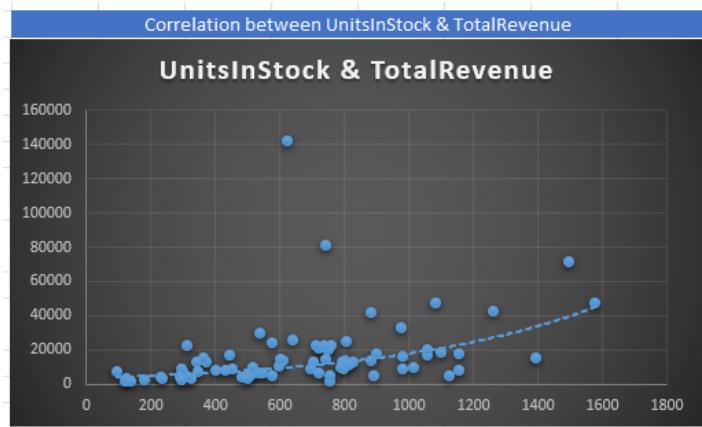
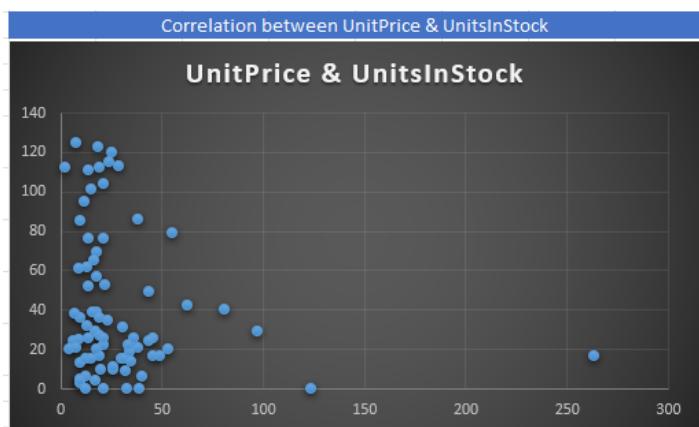
1. **Join Tables:** Uses a `LEFT JOIN` to combine `products` (for product details like ID, name, price, stock) with `order details` (for sales quantities and discounts). The `LEFT JOIN` ensures all products are included, even if they haven't been sold.
2. **Calculate Aggregates:** For each product, it `SUMs` `Quantity` to get `TotalUnitsSold` and `SUMs` `(UnitPrice * Quantity * (1 - Discount))` to get `TotalRevenue`.
3. **Group & Order:** Groups the results by all product-specific columns to ensure unique product rows, then orders them by `ProductID`.

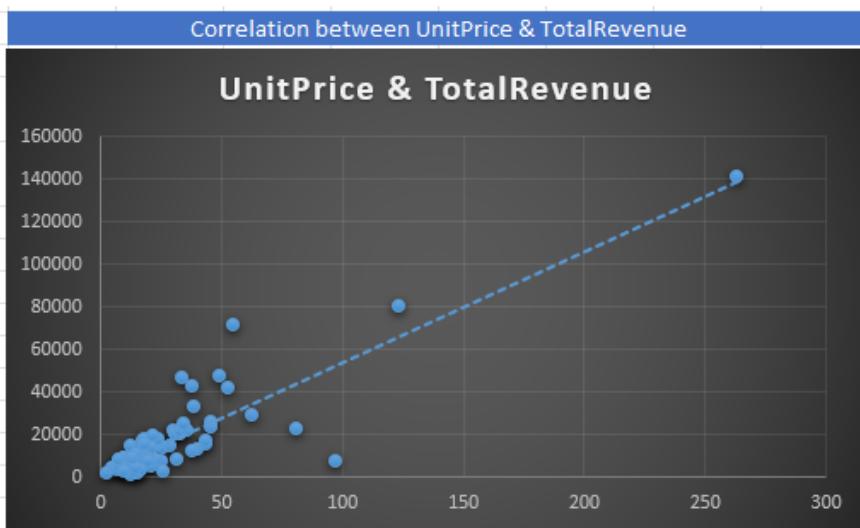
```

SELECT
    p.ProductID,
    p.ProductName,
    p.UnitPrice,
    p.UnitsInStock,
    p.UnitsOnOrder,
    SUM(od.Quantity) AS TotalUnitsSold,
    SUM(od.UnitPrice * od.Quantity * (1 - od.Discount)) AS TotalRevenue
FROM
    products AS p
LEFT JOIN
    `order details` AS od ON p.ProductID = od.ProductID
GROUP BY
    p.ProductID,
    p.ProductName,
    p.UnitPrice,
    p.UnitsInStock,
    p.UnitsOnOrder
ORDER BY
    ProductID ;

```

ProductID	ProductName	UnitPrice	UnitsInStock	UnitsOnOrder	TotalUnitsSold	TotalRevenue
1	Chai	18	39	0	828	12788.1
2	Chang	19	17	40	1057	16355.96
3	Aniseed Syrup	10	13	70	328	3044
4	Chef Anton's Cajun Seasoning	22	53	0	453	8567.9
5	Chef Anton's Gumbo Mix	21.35	0	0	298	5347.2
6	Grandma's Boysenberry Spread	25	120	0	301	7137
7	Uncle Bob's Organic Dried Pears	30	15	0	763	22044.3
8	Northwoods Cranberry Sauce	40	6	0	372	12772
9	Mishi Kobe Niku	97	29	0	95	7226.5
10	Ikura	31	31	0	742	20867.34
11	Queso Cabrales	21	22	30	706	12901.77
12	Queso Manchego La Pastora	38	86	0	344	12257.66
13	Konbu	6	24	0	891	4960.44
14	Tofu	23.25	35	0	404	7991.49
15	Genen Shouyu	15.5	39	0	122	1784.825
16	Pavlova	17.45	29	0	1158	17215.7755
17	Alice Mutton	39	0	0	978	32698.38
18	Carnarvon Tigers	62.5	42	0	539	29171.875
19	Teatime Chocolate Biscuits	9.2	25	0	723	5862.62
20	Sir Rodney's Marmalade	81	40	0	313	22563.36
Grand Total						
			10	10	10	10





## Q10. How does product demand change over months or seasons?

- Data Source: It joins the `orders` and `order details` tables to link order dates with sales quantities.
- Grouping: It groups the data by the year and month (`YYYY-MM`) derived from the `OrderDate`.
- Aggregation: It sums the `Quantity` for each group.
- Sorting: The final results are sorted chronologically by year and month.

```

SELECT
    DATE_FORMAT(o.OrderDate, '%Y-%m') AS OrderYearMonth,
    SUM(od.Quantity) AS QuantitySold
FROM
    orders AS o
JOIN
    `order details` AS od ON o.OrderID = od.OrderID
GROUP BY
    OrderYearMonth
ORDER BY
    OrderYearMonth;
  
```



## Q11. Can we identify anomalies in product sales or revenue performance?

**MonthlySales CTE:** This first part calculates the total units sold and total revenue for each product in each month. It joins the `order details`, `orders`, and `products` tables to get all the necessary data.

**SalesWithChange CTE:** This second part builds on the first by using window functions (`LAG`). For each product, it retrieves the units sold and revenue from the *previous* month, storing them in `PrevUnits` and `PrevRevenue`.

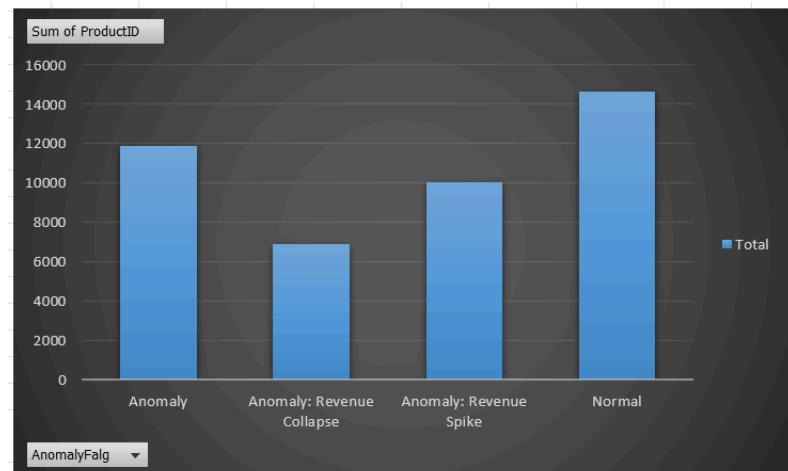
**Final SELECT:** This main query uses the data from the CTEs to:

- Calculate month-over-month growth percentages for units sold and revenue.
- Use a `CASE` statement to create an `AnomalyFlag` that identifies potential issues like a "Revenue Collapse," "Revenue Spike," or a general "Anomaly" based on changes in units and revenue.
- Filter out the very first month's data for each product, as there is no previous month to compare to.
- Order the final results to show anomalies first, followed by products with the highest revenue growth.

```
WITH MonthlySales AS (
    SELECT
        p.ProductID,
        p.ProductName,
        DATE_FORMAT(o.OrderDate, '%Y-%m') AS OrderMonth,
        SUM(od.Quantity) AS UnitsSold,
        SUM(od.UnitPrice * od.Quantity * (1 - od.Discount)) AS Revenue
    FROM `order details` od
    JOIN Orders o ON od.OrderID = o.OrderID
    JOIN Products p ON od.ProductID = p.ProductID
    GROUP BY p.ProductID, p.ProductName, DATE_FORMAT(o.OrderDate, '%Y-%m')
),
SalesWithChange AS (
    SELECT
        ProductID,
        ProductName,
        OrderMonth,
        UnitsSold,
        Revenue,
        LAG(UnitsSold) OVER (PARTITION BY ProductID ORDER BY OrderMonth) AS PrevUnits,
        LAG(Revenue) OVER (PARTITION BY ProductID ORDER BY OrderMonth) AS PrevRevenue
    FROM MonthlySales
)
SELECT
    ProductID,
    ProductName,
    OrderMonth,
    UnitsSold,
    PrevUnits,
    Revenue,
    PrevRevenue,
    ROUND(100.0 * (UnitsSold - PrevUnits) / NULLIF(PrevUnits, 0), 2) AS UnitsGrowthPercent,
    ROUND(100.0 * (Revenue - PrevRevenue) / NULLIF(PrevRevenue, 0), 2) AS RevenueGrowthPercent,
    CASE
        WHEN ABS(UnitsSold - PrevUnits) > 50 THEN 'Anomaly'
        WHEN Revenue < PrevRevenue * 0.5 THEN 'Anomaly: Revenue Collapse'
        WHEN Revenue > PrevRevenue * 1.5 THEN 'Anomaly: Revenue Spike'
        ELSE 'Normal'
    END AS AnomalyFlag
FROM SalesWithChange
WHERE PrevUnits IS NOT NULL
ORDER BY AnomalyFlag DESC, RevenueGrowthPercent DESC;
```

ProductID	ProductName	OrderMonth	UnitsSold	PrevUnits	Revenue	PrevRevenue	UnitGrowthPercen	RevenueGrowthPercen	AnomalyFlag
11	Queso Cabrales	1995-11	15	10	315	210	50	50	Normal
8	Northwoods Cranberry Sauce	1996-03	30	20	1200	800	50	50	Normal
29	Thüringer Rostbratwurst	1994-09	15	10	1485	990	50	50	Normal
29	Thüringer Rostbratwurst	1994-10	25	15	2227.5	1485	66.67	50	Normal
23	Tunnbröd	1995-02	60	40	432	288	50	50	Normal
41	Jack's New England Clam Chowder	1995-03	30	25	231	154	20	50	Normal
30	Nord-Ost Matjeshering	1995-11	3	2	58.2525	38.835	50	50	Normal
58	Escargots de Bourgogne	1995-08	30	20	397.5	265	50	50	Normal
65	Louisiana Fiery Hot Pepper Sauce	1996-01	82	50	1557.7	1041.975	64	49.49	Normal
46	Spegesild	1995-03	35	22	307.2	206.4	59.09	48.84	Normal
65	Louisiana Fiery Hot Pepper Sauce	1994-11	40	30	672	453.6	33.33	48.15	Normal
56	Gnocchi di nonna Alice	1995-12	97	68	3275.6	2223	42.65	47.35	Normal
1	Chai	1995-07	10	8	180	122.4	25	47.06	Normal
1	Chai	1996-05	104	81	1741.5	1188	28.4	46.59	Normal
56	Gnocchi di nonna Alice	1995-03	113	75	3313.6	2280	50.67	45.33	Normal
44	Stonewall Tomato	1995-04	24	16	261.77	240.05	8.33	45.24	Normal

AnomalyFlag	Sum of ProductID
Anomaly	11878
Anomaly: Revenue Collapse	6896
Anomaly: Revenue Spike	10032
Normal	14663



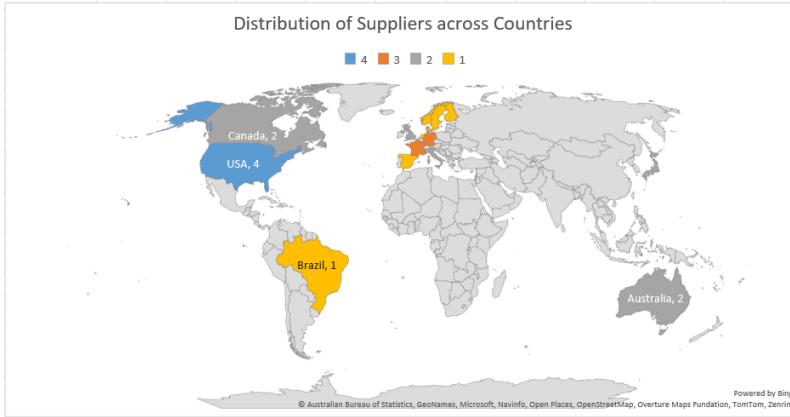
## Q12. Are there any regional trends in supplier distribution and pricing?

Purpose: To find the number of suppliers per country.

- Grouping: It groups all the records from the `suppliers` table by `Country`.
- Aggregation: It counts the number of suppliers within each country group, aliasing the count as `NoOfSuppliers`.
- Ordering: It sorts the final results to show the countries with the most suppliers first.

```

SELECT
    Country,
    COUNT(SupplierID) AS NoOfSuppliers
FROM
    suppliers
GROUP BY
    Country
ORDER BY
    NumberofSuppliers DESC;
  
```



Country	NoOfSupplier
USA	4
Germany	3
France	3
UK	2
Japan	2
Australia	2
Italy	2
Canada	2
Spain	1
Sweden	1
Brazil	1
Norway	1
Sweden	1
Singapore	1
Denmark	1
Netherland	1
Finland	1

Purpose: To determine the average product price based on the supplier's location.

- Data Source: It joins the **suppliers** table with the **products** table on the **SupplierID** to connect products to their origin.
- Aggregation: It calculates the average (**AVG**) of the **UnitPrice** for all products.
- Grouping: It groups the results by **SupplierCountry** and **SupplierCity**, so the average price is calculated for each unique city.

We can quantify trend with the help of Unit Price across Regions

```

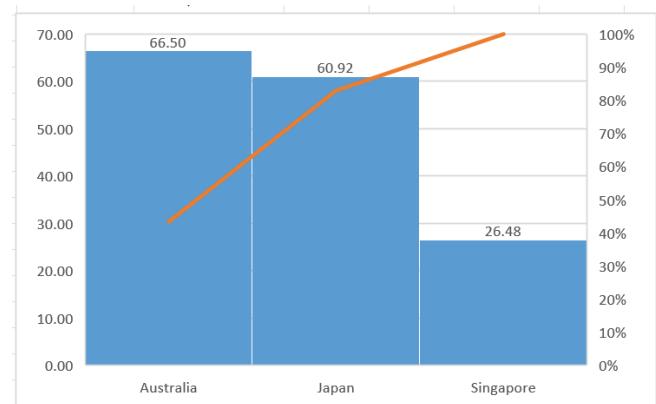
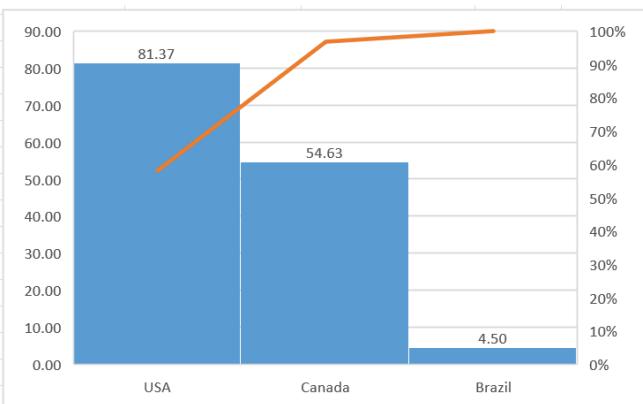
SELECT
    s.Country AS SupplierCountry,
    s.City AS SupplierCity,
    AVG(p.UnitPrice) AS AvgProdUnitPrice
FROM
    suppliers AS s
JOIN
    products AS p ON s.SupplierID = p.SupplierID
GROUP BY
    SupplierCountry,
    SupplierCity
ORDER BY
    SupplierCountry, SupplierCity;
  
```

Country	City	AvgProdUnitPric
Australia	Melbourne	35.57
Australia	Sydney	30.93
Brazil	SÃ£o Paulo	4.50
Canada	MontrÃ©al	15.73
Canada	Ste-Hyacinthe	38.90
Denmark	Lyngby	10.75
Finland	Lappeenranta	18.08
France	Annecy	44.50
France	Montceau	13.25
France	Paris	140.75
Germany	Berlin	29.71
Germany	Cuxhaven	25.89
Germany	Frankfurt	44.68
Italy	Ravenna	26.43
Italy	Salerno	28.75
Japan	Osaka	14.92
Japan	Tokyo	46.00
Netherlands	Zaandam	11.13
Norway	Sandvika	20.00
Singapore	Singapore	26.48
Spain	Oviedo	29.50
Sweden	Stockholm	20.00
Sweden	GÃ¶teborg	15.00
UK	London	15.67
UK	Manchester	28.18
USA	Ann Arbor	31.67
USA	Bend	15.33
USA	Boston	14.02
USA	New Orleans	20.35

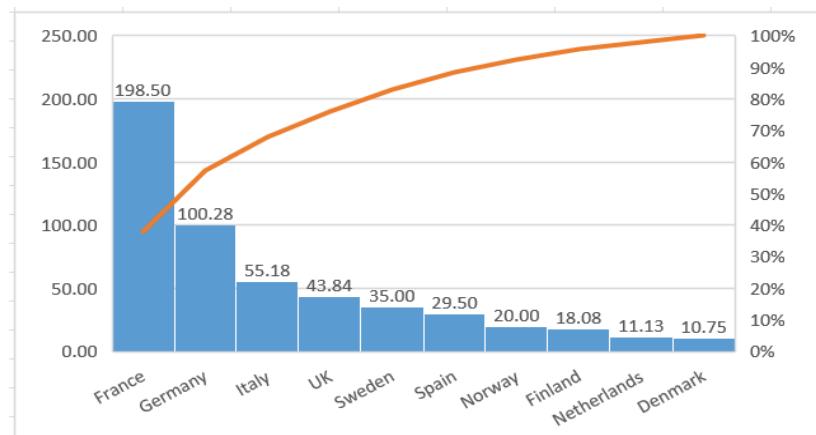
## We have distributed based on the regions:-

Americas Region	
Country	AvgProdUnitPrice
Brazil	4.50
Canada	15.73
Canada	38.90
USA	31.67
USA	15.33
USA	14.02
USA	20.35

Asia-Pacific Region	
Country	AvgProdUnitPrice
Australia	35.57
Australia	30.93
Japan	14.92
Japan	46.00
Singapore	26.48



Europe Region	
Country	AvgProdUnitPrice
Denmark	10.75
Finland	18.08
France	44.50
France	13.25
France	140.75
Germany	29.71
Germany	25.89
Germany	44.68
Italy	26.43
Italy	28.75
Netherlands	11.13
Norway	20.00
Spain	29.50
Sweden	20.00
Sweden	15.00
UK	15.67
UK	28.18



In the above charts we have successfully observed regional trends in Supplier distribution and pricing by dividing the regions into Americas, Asia-Pacific and Europe.

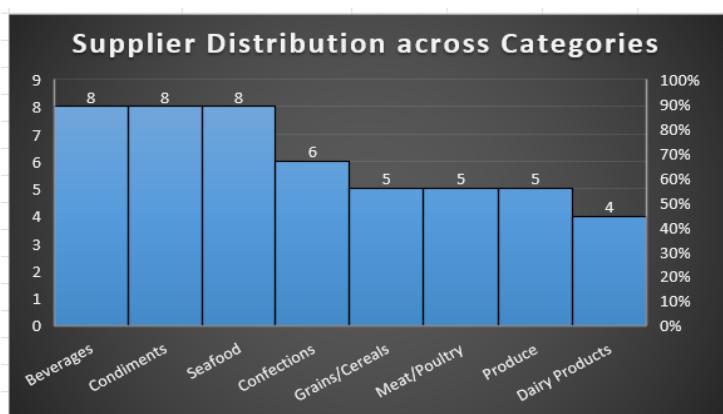
The United States has the highest number of suppliers with 4, followed by France and Germany with 3 each. Most other countries, including the UK, Italy, Australia, Canada, and Japan, have fewer than 3 suppliers. This suggests a concentration of suppliers in the US and parts of Europe.

Products from French suppliers have the highest average unit price at approximately \$76.75. Germany and Australia have the next highest average prices, both over \$30. Brazil has the lowest average unit price at \$4.50.

### Q13. How are suppliers distributed across different product categories?

Approach:

- Joins Tables:** The query connects the `categories` table with the `products` table using the `CategoryID` to associate each product with its category name.
- Groups Data:** It groups the products by their `CategoryName`. This is the basis for counting suppliers within each specific category.
- Counts Suppliers:** For each category group, it counts the number of *distinct* `SupplierIDs`. This ensures that a single supplier who provides multiple products in the same category is only counted once for that category.
- Sorts Results:** The final output is sorted alphabetically by the `CategoryName` for easy readability.



This bar chart shows the number of suppliers for each product category. The height of each bar represents the supplier count, allowing us to immediately see which categories have the most suppliers (tallest bars) and which have the fewest (shortest bars). This visually highlights market concentration and competition across the different product types.

## Q14. How do supplier pricing and categories relate across different regions?

Approach:

- **Joins Tables:** The query links the **suppliers** table with the **products** table on the **SupplierID** to connect each product to its supplier's information.
- **Selects Data:** It retrieves the **CompanyName** of the supplier, the **ProductName**, and the **UnitPrice**.
- **Filters Data:** It applies a filter (**WHERE**) to include only those products where the **UnitPrice** is between 20 and 30.
- **Sorts Results:** The final output is organized by sorting the results alphabetically first by the supplier's **CompanyName** and then by the **ProductName**.

```

SELECT
    s.Country AS SupplierCountry,
    s.City AS SupplierCity,
    c.CategoryName,
    AVG(p.UnitPrice) AS AvgProductUnitPrice
FROM
    suppliers AS s
JOIN
    products AS p ON s.SupplierID = p.SupplierID
JOIN
    categories AS c ON p.CategoryID = c.CategoryID
GROUP BY
    SupplierCountry,
    SupplierCity,
    c.CategoryName
ORDER BY
    SupplierCountry,
    SupplierCity,
    AverageProductUnitPrice DESC;
  
```

SupplierCountry	SupplierCity	CategoryName	AvgProductUnitPrice
Australia	Melbourne	Seafood	62.50
Australia	Melbourne	Condiments	43.90
Australia	Melbourne	Meat/Poultry	39.00
Australia	Melbourne	Confections	17.45
Australia	Melbourne	Beverages	15.00
Australia	Sydney	Produce	53.00
Australia	Sydney	Meat/Poultry	32.80
Australia	Sydney	Grains/Cereals	7.00
Brazil	SÃ£o Paulo	Beverages	4.50
Canada	MontrÃ©al	Meat/Poultry	15.73
Canada	Ste-Hyacinthe	Confections	49.30
Canada	Ste-Hyacinthe	Condiments	28.50
Denmark	Lyngby	Seafood	10.75
Finland	Lappeenranta	Confections	18.13
Finland	Lappeenranta	Beverages	18.00
France	Annecy	Dairy Products	44.50
France	Montceau	Seafood	13.25
France	Paris	Beverages	140.75
Germany	Berlin	Confections	29.71
Germany	Cuxhaven	Seafood	25.89
Germany	Frankfurt	Meat/Poultry	123.79
Germany	Frankfurt	Produce	45.60
Germany	Frankfurt	Grains/Cereals	33.25
Germany	Frankfurt	Condiments	13.00
Germany	Frankfurt	Beverages	7.75

This table is a curated list of products that meet a specific pricing condition. It tells you which products fall into a particular price bracket and identifies the company that supplies each one. The list is organized in a logical way, grouping all products from the same supplier together for easy reference.