

Final Report: Sales and Customer Behavior Analysis Using the Northwind Database

By Mariam Ayman Helmy

Introduction

The Northwind database provides a comprehensive dataset for analyzing sales patterns, customer behavior, and operational performance.

The primary goal of this project is to utilize SQL queries to extract meaningful insights that will help Northwind improve sales strategies and strengthen customer relationships. By categorizing customers, evaluating product performance, and assessing employee contributions, this analysis aims to identify actionable trends and opportunities for growth.

This report documents the results of SQL queries written to address the project goals, along with the findings and their business implications. Visualizations derived from the results are also included to facilitate a clearer understanding of the data trends.

Contents:

1. SQL Queries, Analysis, and Insights

- **Customer Segmentation**
 - RFM Analysis
 - Order Value Segments
 - Customer Segmentation Key Insights
- **Product Analysis**
 - Top 10 Revenue Generators
 - Top 10 Most Frequently Ordered Products
 - Bottom 5 Products by Sales Volume
 - Product Analysis Key Insights
- **Order Analysis**
 - Seasonality
 - Day-of-the-Week Analysis
 - Order Size Distribution
 - Order Analysis Insights and Visualizations
- **Employee Performance**
 - Employee Performance Evaluation
 - Employee Performance Key Insights

2. Conclusion

- **Summary of Insights**
- **Recommendations for Northwind**

1. SQL Queries, Analysis, and Insights

Customer Segmentation

Analysis of customer segmentation using two methods: **RFM Analysis** and **Order Value Segmentation**.

▪ RFM Segmentation

RFM analyzes customers based on their purchase behavior to identify actionable customer groups.

SQL Query

```
SELECT
    customerid,
    JULIANDAY('2024-12-20') - JULIANDAY(MAX(OrderDate)) AS Recency, --Fixed Reference date so that I get the same result everytime
    COUNT(O.orderid) AS TotalOrders,
    SUM(unitprice * quantity * (1 - discount)) AS AmountSpent
FROM
    Orders O
INNER JOIN
    "Order Details" D
ON
    O.OrderID = D.OrderID
GROUP BY
    customerid
ORDER BY
    Recency, TotalOrders DESC, AmountSpent DESC
```

For the first query, the recency, frequency and monetary values are calculated as:

- JULIANDAY('2024-12-20') - JULIANDAY(MAX(OrderDate)) AS Recency, to calculate the number of days since the customer's most recent order using a set date to get the same results every time the query is run.
- COUNT(O.orderid) AS TotalOrders to count the total orders placed by each customer
- SUM(unitprice * quantity * (1 - discount)) AS AmountSpent to compute the total revenue generated by the customer, accounting for discounts

```
SELECT
    customerid,
    JULIANDAY('2024-12-20') - JULIANDAY(MAX(OrderDate)) AS Recency,
    COUNT(O.orderid) AS TotalOrders,
    SUM(unitprice * quantity * (1 - discount)) AS AmountSpent,
    CASE
        WHEN JULIANDAY('now') - JULIANDAY(MAX(OrderDate)) <= 450
            AND COUNT(O.orderid) >= 7000
            AND SUM(unitprice * quantity * (1 - discount)) >= 5250000 THEN 'Champion'
        WHEN COUNT(O.orderid) >= 6500
            OR SUM(unitprice * quantity * (1 - discount)) >= 5000000 THEN 'Potential Loyalist'
        ELSE 'At Risk'
    END AS CustomerSegment
FROM
    Orders O
INNER JOIN
    "Order Details" D
ON
    O.OrderID = D.OrderID
GROUP BY
    customerid
ORDER BY
    Recency, TotalOrders DESC, AmountSpent DESC
```

After reviewing the result values of the first query, a threshold or criteria for the customer segmentation categories was created in the case statement for the final query.

Sample Results

CustomerID	Recency	TotalOrders	AmountSpent	CustomerSegment
MAISD	418.9931944445707	6391	4815829.325	At Risk
BOLID	419.23775462945923	7154	5398064.44	Champion
MORGK	419.72310185199603	7252	5349089.59	Champion
ROMEY	420.72994212945923	6598	4782770.28	Potential Loyalist
WHITC	421.4579976852983	6514	4662707.935	Potential Loyalist
HUNGC	421.87706018518656	7808	5698023.67	Champion
WILMK	422.4779050927609	6230	4635588.03	At Risk
OLDWO	423.19878472201526	6559	4809258.5125	Potential Loyalist
ANATR	423.5168865742162	7497	5534356.65	Champion
PRINI	423.7376388888806	7477	5436770.55	Champion
VALON	423.8451504628174	6727	4925094.61	Potential Loyalist
FRANR	423.8609837964177	7047	5180799.4	Potential Loyalist

▪ Order Value Segmentation

This segmentation categorizes customers based on their average order value.

SQL Query

```
SELECT
    customerid,
    Avg(unitprice * quantity * (1 - discount)) AS AvgOrderValue
FROM
    Orders O
INNER JOIN
    "Order Details" D
ON
    O.OrderID = D.OrderID
GROUP BY
    customerid
ORDER BY
    AvgOrderValue DESC
```

The first query computes the average value of each customer's orders as:

- Avg(unitprice * quantity * (1 - discount)) AS AvgOrderValue

On which the criteria for the customer value segmentation were set for the case statement in the final query.

```
SELECT
  customerid,
  Avg(unitprice * quantity * (1 - discount)) AS AvgOrderValue,
  CASE
    WHEN Avg(unitprice * quantity * (1 - discount)) > 741 THEN 'High Value'
    WHEN Avg(unitprice * quantity * (1 - discount)) BETWEEN 727 AND 741 THEN 'Medium Value'
    ELSE 'Low Value'
  END AS CustomerValue
FROM
  Orders O
INNER JOIN
  "Order Details" D
ON
  O.OrderID = D.OrderID
GROUP BY
  customerid
ORDER BY
  AvgOrderValue DESC
```

Sample Results

CustomerID	AvgOrderValue	CustomerValue
HANAR	755.2506014814815	High Value
BOLID	754.5519206038581	High Value
MAISD	753.5329877953372	High Value
VINET	752.9048105665624	High Value
FAMIA	750.8865674846626	High Value
RANCH	750.6224790710235	High Value
GROSR	749.8450467741935	High Value
EASTC	749.3826149942706	High Value
PERIC	749.2720534790959	High Value

Customer Segmentation Key Insights

Analyzing the dataset through the lens of RFM and Order Value segmentation, several key business insights emerge:

1. Customer Segmentation and Prioritization

- **Champions:** These are your most valuable customers, with high frequency (TotalOrders) and high monetary value (AmountSpent). For example:
 - **BOLID:** With 7,154 orders and over \$5.39M spent, this customer demonstrates loyalty and consistent engagement.

- HUNGC: Tops the chart with 7,808 orders and \$5.69M in spending.
- BSBEV: A standout with the highest orders (8,287) and over \$6.15M spent.
- Potential Loyalists: Customers with high recency but moderate to high monetary values and frequency. They are potential long-term loyalists if targeted correctly.

2. At-Risk Customers

- These customers show low recency, indicating disengagement, despite having contributed significant orders and revenue in the past.
 - Example: MAISD and ALFKI both contributed high monetary values but have declining recency values. Targeted re-engagement campaigns could help win them back.

3. High-Value Customers Based on Average Order Value

- Customers with an AvgOrderValue exceeding \$740 represent a significant revenue base, making them crucial for premium product offers or upselling strategies.
- Actionable Insight: Focus premium services, early access to new products, or exclusive promotions for this segment.

4. Medium and Low Value Segments

- Many customers fall into the Medium Value segment with average orders in the range of \$730–740.
- This group represents an opportunity for targeted upselling, or rewards to increase loyalty and transition them to High-Value customers.

5. Strategic Recommendations

- Loyalty Programs: Reward top segments (Champions, Potential Loyalists) with benefits based on spending and orders.
- Reactivation Campaigns: Develop win-back strategies for At-Risk customers, leveraging past purchase data to offer personalized deals or incentives.
- Customer Retention: Regularly engage Potential Loyalists with offers to encourage transition to Champion status.
- Focus on Medium Value customers with tailored promotions for complementary products to increase their monetary contributions.

Product Analysis

Summarizing the product analysis results based on three key performance indicators: **High Revenue Value Products**, **High Sales Volume Products** and **Slow Movers**.

▪ Top 10 Revenue-Generating Products

SQL Query

```
SELECT
    P.ProductID,
    ProductName,
    SUM(D.UnitPrice * D.Quantity * (1 - D.Discount)) AS Revenue
FROM
    "Order Details" AS D
INNER JOIN
    Products AS P ON D.ProductID = P.ProductID
GROUP BY
    P.ProductID, ProductName
ORDER BY
    Revenue DESC
LIMIT 10
```

This query identifies the top 10 products that generate the highest revenue.

- The revenue for each product is calculated by multiplying the unit price, quantity sold, and applying any applicable discount.
- The result is ordered by revenue in descending order, and the top 10 products are selected.

Results

ProductID	ProductName	Revenue
38	Côte de Blaye	53265895.235
29	Thüringer Rostbratwurst	24623469.232
9	Mishi Kobe Niku	19423037.5
20	Sir Rodney's Marmalade	16653807.36
18	Camaron Tigers	12604671.875
59	Raclette Courdavault	11216410.7
51	Manjimup Dried Apples	10664768.65
62	Tarte au sucre	9952936.07
43	Ipoh Coffee	9333374.7
28	Rössle Sauerkraut	9252765.44

- **High Sales Volume: Top 10 Most Frequently Ordered Products**

SQL Query

```
CREATE VIEW Sales_Volume_Products AS -- View will be used for high and low sales volume
SELECT
    P.ProductID,
    ProductName,
    SUM(D.Quantity) AS TotalQuantitySold
FROM
    "Order Details" AS D
INNER JOIN
    Products AS P ON D.ProductID = P.ProductID
GROUP BY
    P.ProductID, ProductName

SELECT * FROM Sales_Volume_Products ORDER BY TotalQuantitySold DESC LIMIT 10
```

CREATE VIEW Sales_Volume_Products: This creates a reusable view named `Sales_Volume_Products` that aggregates sales data for products. The use of this view simplifies repeated queries for sales volume analysis. The resulting view contains the following columns:

- **ProductID:** Unique identifier for each product.
- **ProductName:** Name of the product.
- **TotalQuantitySold:** Total quantity sold for each product.

The second query based on the created view is:

- **ORDER BY TotalQuantitySold DESC:** sorts the products in descending order of `TotalQuantitySold` to prioritize high sales volumes
- **LIMIT 10:** limits the output to the top 10 products.

Results

ProductID	ProductName	TotalQuantitySold
66	Louisiana Hot Spiced Okra	206213
20	Sir Rodney's Marmalade	205637
19	Teatime Chocolate Biscuits	205487
61	Sirop d'érable	205005
26	Gumbär Gummibärchen	204761
70	Outback Lager	204403
57	Ravioli Angelo	204251
59	Raclette Courdavault	204137
7	Uncle Bob's Organic Dried Pears	203970
34	Sasquatch Ale	203667

- **Slow Movers: Bottom 5 Products by Sales Volume**

SQL Query

```
SELECT * FROM Sales_Volume_Products ORDER BY TotalQuantitySold LIMIT 5
```

This query selects the bottom 5 products from the view that was previously created to summarize the sales volume for each product:

- ORDER BY TotalQuantitySold: sorts the products in ascending order of TotalQuantitySold to prioritize low sales volumes.
- LIMIT 5: Limits the output to the 5 products with the lowest sales volumes.

Results

ProductID	ProductName	TotalQuantitySold
58	Escargots de Bourgogne	197673
47	Zaanse koeken	197889
4	Chef Anton's Cajun Seasoning	198726
29	Thüringer Rostbratwurst	199010
73	Röd Kaviar	199042

- **Product Analysis Key Insights**

1. High Revenue Products:

- The top 10 revenue-generating products represent high-performing assets for the business. By identifying these products, Northwind can:
 - Focus marketing efforts on promoting these items.
 - Consider offering bundled deals or cross-sell opportunities with these high-selling products.

2. High Sales Volume Products:

- The top 10 most frequently ordered products are highly popular among customers. Northwind can:
 - Leverage these products in customer loyalty programs or promotions.

- Evaluate if there is room to increase prices slightly without affecting customer demand, especially for fast-moving products.
- Ensure adequate inventory levels to avoid stock-outs, which could lead to missed sales.

3. Slow Movers:

- The slow-moving products require attention. Northwind can:
 - Analyze the reasons behind low sales, whether due to poor marketing, outdated features, or customer disinterest.
 - Consider discounting or bundling these products with higher-selling items to improve turnover.
 - Phase out products that are not performing well and focus on improving the product mix.

Order Analysis

Provides insights into order trends based on **Seasonality**, **Day-of-the-Week Analysis**, and **Distribution of Order Quantities**.

- **Seasonality: Identify Seasonal Fluctuations in Order Volume**

SQL Query

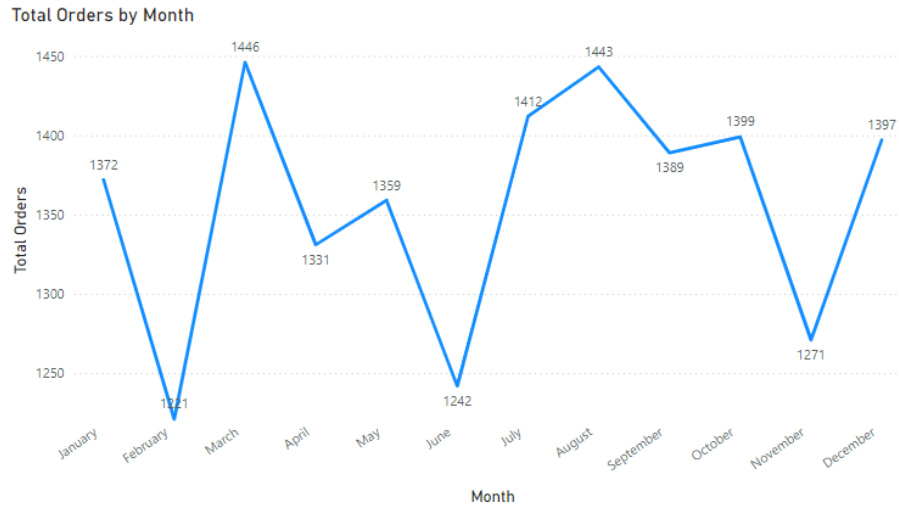
```
SELECT
    strftime('%m', OrderDate) AS OrderMonth,
    COUNT(OrderID) AS TotalOrders
FROM
    Orders
GROUP BY
    OrderMonth
ORDER BY
    OrderMonth
```

This query uses:

- `strftime('%m', OrderDate)` to extract the month from the `OrderDate` column.
- It counts the total number of orders (`COUNT(OrderID)`) for each month (`GROUP BY OrderMonth`).
- The result is ordered by the month (`ORDER BY OrderMonth`), giving insights into how orders are distributed throughout the year.

Results & Visualization

OrderMonth	TotalOrders
01	1372
02	1221
03	1446
04	1331
05	1359
06	1242
07	1412
08	1443
09	1389
10	1399
11	1271
12	1397



This line graph is used to visualize the number of orders across the months, highlighting peak and off-peak months and helping to understand any seasonal patterns.

▪ Day-of-the-Week Analysis: Most Popular Order Days

SQL Query

```
SELECT
    CASE strftime('%w', OrderDate)
        WHEN '0' THEN 'Sunday' -- Default index for weekdays starts at 0 for Sunday
        WHEN '1' THEN 'Monday'
        WHEN '2' THEN 'Tuesday'
        WHEN '3' THEN 'Wednesday'
        WHEN '4' THEN 'Thursday'
        WHEN '5' THEN 'Friday'
        WHEN '6' THEN 'Saturday'
    END AS WeekDay,
    COUNT(OrderID) AS TotalOrders
FROM
    Orders
GROUP BY
    WeekDay
ORDER BY
    TotalOrders DESC
```

This query summarizes orders by:

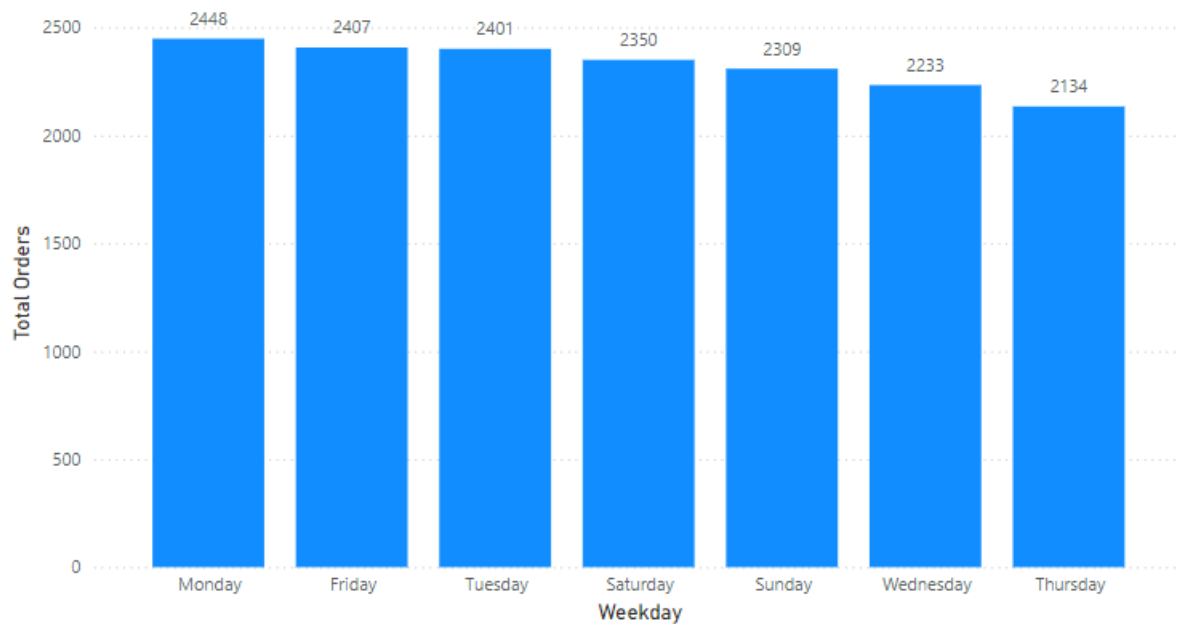
- The `strftime('%w', OrderDate)` function extracts the day of the week as a number starting by default 0 as Sunday.
- A CASE statement assigns meaningful names to these numbers.

- Orders are grouped by the day of the week and sorted by the total number of orders in descending order.

Results & Visualization

WeekDay	TotalOrders
Monday	2448
Friday	2407
Tuesday	2401
Saturday	2350
Sunday	2309
Wednesday	2233
Thursday	2134

Total Orders by Weekday



This chart is used to display the number of orders for each day of the week, helping to identify which days of the week are most popular for placing orders.

- **Order Size Analysis: Distribution of Order Quantities**

SQL Query

```
SELECT CASE
    WHEN TotalQuantity BETWEEN 1 AND 10 THEN '1-10'
    WHEN TotalQuantity BETWEEN 11 AND 20 THEN '11-20'
    WHEN TotalQuantity BETWEEN 21 AND 50 THEN '21-50'
    WHEN TotalQuantity BETWEEN 51 AND 100 THEN '51-100'
    WHEN TotalQuantity BETWEEN 101 AND 500 THEN '101-500'
    WHEN TotalQuantity BETWEEN 501 AND 1000 THEN '501-1000'
    WHEN TotalQuantity BETWEEN 1001 AND 2000 THEN '1001-2000'
    WHEN TotalQuantity BETWEEN 2001 AND 3000 THEN '2001-3000'
    ELSE '3000+'
END AS OrderSizeRange,
COUNT(OrderID) AS OrderCount
FROM ( SELECT OrderID,
    SUM(Quantity) AS TotalQuantity
    FROM "Order Details"
    GROUP BY OrderID
) AS OrderDetails
GROUP BY
    OrderSizeRange
ORDER BY -- To ensure the order is correct
    CASE
        WHEN TotalQuantity BETWEEN 1 AND 10 THEN 1
        WHEN TotalQuantity BETWEEN 11 AND 20 THEN 2
        WHEN TotalQuantity BETWEEN 21 AND 50 THEN 3
        WHEN TotalQuantity BETWEEN 51 AND 100 THEN 4
        WHEN TotalQuantity BETWEEN 101 AND 500 THEN 5
        WHEN TotalQuantity BETWEEN 501 AND 1000 THEN 6
        WHEN TotalQuantity BETWEEN 1001 AND 2000 THEN 7
        WHEN TotalQuantity BETWEEN 2001 AND 3000 THEN 8
        ELSE 9
```

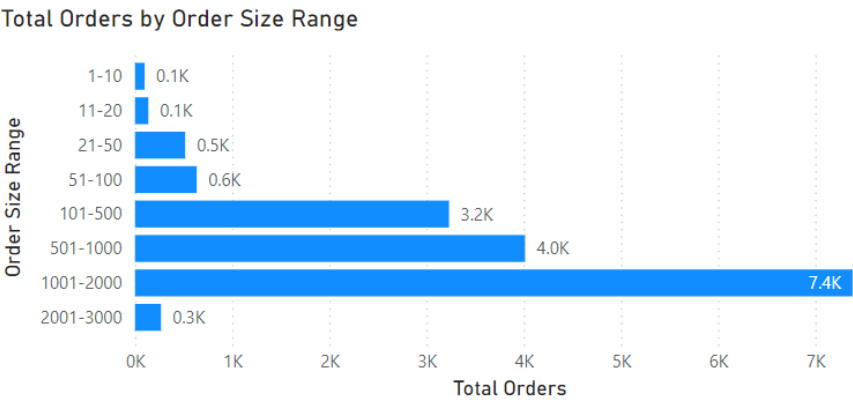
The inner query (SUM(Quantity) AS TotalQuantity) calculates the total quantity of items ordered for each OrderID.

The outer query categorizes these totals into size ranges (e.g., '1-10', '11-20') using a CASE statement and groups them.

The final output includes the number of orders in each size range, sorted logically.

Results & Visualization

OrderSizeRange	OrderCount
1-10	99
11-20	137
21-50	516
51-100	635
101-500	3231
501-1000	4013
1001-2000	7384
2001-3000	267



The bar chart represents the count of orders in each size range.

▪ Order Analysis Key Insights

The analysis of orders provides critical insights into customer behavior, demand patterns, and operational planning:

- Seasonality:
 - Orders fluctuate across the year, with peaks in March (1,446 orders), August (1,443 orders), and July (1,412 orders), which suggested increased activity during these months.
 - Lower activity is observed in February (1,221 orders) and June (1,242 orders). This seasonality can guide inventory preparation and promotional campaigns during peak months.

2. Day-of-the-Week Patterns:

- The most popular day for placing orders is Monday (2,448 orders), followed closely by Friday (2,407 orders) and Tuesday (2,401 orders).
- Thursday (2,134 orders) is the least active day. Understanding these patterns allows for optimized staffing, marketing strategies, and resource allocation.

3. Order Size Distribution:

- Large orders dominate, with 7,384 orders in the 1001-2000 size range and 4,013 orders in the 501-1000 range.
- Smaller orders (1-10 and 1-20 ranges) are less common, with only 99 and 137 orders, respectively.
- This distribution highlights the importance of accommodating large-scale orders while exploring opportunities to increase smaller order volumes through targeted campaigns.

Employee Performance Analysis

Evaluate employee performance based on: **Total sales volume generated**, **Number of orders processed** and **Average order value**.

SQL Query

```
SELECT
    E.EmployeeID,
    LastName || ', ' || FirstName AS EmployeeName,
    COUNT(O.OrderID) AS TotalOrders,
    SUM(unitprice * quantity * (1 - discount)) AS TotalRevenue,
    AVG(unitprice * quantity * (1 - discount)) AS AverageOrderValue
FROM
    Employees E
INNER JOIN
    Orders O ON E.EmployeeID = O.EmployeeID
INNER JOIN
    "Order Details" OD ON O.OrderID = OD.OrderID
GROUP BY
    E.EmployeeID, EmployeeName
ORDER BY
    E.EmployeeID
```

This query joins the Employees, Orders, and Order Details tables to evaluate each employee's performance by:

- **Total Orders:** COUNT(O.OrderID) - The number of orders each employee processed.
- **Total Revenue:** SUM(unitprice * quantity * (1 - discount)) - Total revenue generated from the orders.
- **Average Order Value:** AVG(unitprice * quantity * (1 - discount)) - Average revenue per order.

Results

!	EmployeeID	EmployeeName	TotalOrders	TotalRevenue	AverageOrderValue
1		Davolio, Nancy	67619	49659423.2345	734.4004382569988
2		Fuller, Andrew	66365	48314100.765	728.005737436902
3		Leverling, Janet	68246	50445573.763	739.1726073762565
4		Peacock, Margaret	69871	51488395.196	736.9065162370655
5		Buchanan, Steven	69868	51386459.1025	735.4791764827961
6		Suyama, Michael	66190	49139966.5595	742.4077135443421
7		King, Robert	67361	49651899.305	737.101576654147
8		Callahan, Laura	67401	49281136.8075	731.163288489785
9		Dodsworth, Anne	66362	49019678.4365	738.6709025722552

▪ **Employee Performance Key Insights**

The evaluation of employee performance offers valuable insights into operational efficiency and individual contributions:

1. Total Orders:

- Margaret Peacock (69,871 orders) and Steven Buchanan (69,868 orders) processed the highest number of orders, indicating strong productivity.
- Other employees, such as Michael Suyama (66,190 orders) and Andrew Fuller (66,365 orders), processed fewer orders but still contributed significantly.

2. Revenue Generation:

- Employees like Margaret Peacock (\$51,488,395.20) and Steven Buchanan (\$51,386,459.10) generated the highest revenue, aligning with their order volume.
- Revenue generation closely correlates with the number of orders handled, indicating consistent performance across the team.

3. Average Order Value:

- Michael Suyama achieved the highest average order value (\$742.41), suggesting an ability to manage high-value transactions effectively.
- Average order values for other employees range from \$728.01 (Andrew Fuller) to \$739.17 (Janet Leverling), reflecting minimal variation and consistent performance across the team.

2. Conclusion

▪ Summary of Insights

The analysis of the Northwind database provided a comprehensive understanding of customer behavior, product performance, order trends, and employee contributions. Key insights include:

1. Customer Segmentation:

- Champions represent the most valuable customers, contributing significantly to sales.
- Potential Loyalists and At-Risk Customers present opportunities for targeted engagement strategies.
- High-value customers based on average order value are ideal for premium product offers and upselling initiatives.

2. Product Analysis:

- High-revenue products and frequently ordered items can drive marketing and inventory strategies.
- Addressing slow-moving products through discounts, bundling, or phasing out can optimize the product mix.

3. Order Trends:

- Seasonality indicates peaks in March, August, and July, guiding inventory and promotional planning.
- Day-of-the-week analysis identifies Monday and Friday as the most active days for orders.
- Order size distribution highlights a dominance of large-scale orders, with potential for increasing smaller orders through tailored campaigns.

4. Employee Performance:

- Consistent productivity and revenue generation across employees, with notable contributions from top performers like Margaret Peacock and Steven Buchanan.
- High average order values demonstrate the team's efficiency in managing high-value transactions.

- **Recommendations for Northwind**

Based on the findings, the following actions are recommended:

1. Customer Engagement:

- Launch loyalty programs for Champions and Potential Loyalists to enhance retention.
- Develop reactivation campaigns targeting At-Risk Customers with personalized incentives.
- Focus on Medium Value customers with promotions to increase spending and loyalty.

2. Product Strategy:

- Prioritize high-revenue and frequently ordered products in marketing efforts.
- Address slow-moving products through discounts, bundling, or product mix adjustments.
- Ensure adequate stock levels for popular products to meet customer demand.

3. Order Management:

- Align inventory and staffing with seasonal demand patterns and peak order days.
- Explore initiatives to boost smaller order volumes through promotions or bundles.

4. Employee Optimization:

- Recognize and reward top-performing employees for their contributions.
- Provide training to further enhance team productivity and efficiency in handling high-value orders.

By implementing these recommendations, Northwind can strengthen customer relationships, optimize product performance, and enhance overall operational efficiency, paving the way for sustained growth and success.