**Customer Sales Analysis Project:** Unlocking Customer Loyalty

An In-Depth Analysis on sales and Customers using RFM Segmentation Methodology.

**Project Overview:**

Ladies and Gentlemen, gather 'round and listen closely! We've got a task at hand that requires some serious sleuthing skills - uncovering the secrets of our customer's shopping habits. And what better way to do that than by diving into a sea of sales data and categorizing our customers based on their Monetary Value, Frequency, and Recency of purchase. Yes, you heard it right, we'll be using the powerful tool of RFM (Recency, Frequency, Monetary) analysis to get to the bottom of this mystery.

Think of it as a modern-day treasure hunt, where instead of searching for buried gold, we'll be uncovering insights about our customers' buying patterns. And we won't be doing this alone! We'll be armed with two trusty tools - SQL Server Management Studio (SSMS) and Power BI. These tools will help us sift through the data, analyze it, and present it in a visually stunning way. So buckle up, and let's go on this exciting journey to discover what makes our customers tick!

**Tools:**

1. Microsoft SQL Server Management Studio
2. Power BI

A tale of two heroes who worked together to conquer the world of customer sales analysis! On one side, we have SSMS, the stoic and dependable workhorse that tirelessly crunched the numbers and brought order to the chaos of customer data. And on the other, we have Power BI, the flashy showman who took those numbers and turned them into stunning, eye-catching visuals that told the story of our customer's buying habits. Together, these two made a formidable team, and with their help, we were able to gain valuable insights into the world of customer sales. So sit back, grab a popcorn, and let's see how SSMS and Power BI teamed up to save the day!

**Data source:**

The data used in this project was extracted from a local sales database containing information on customer purchases.

**Project Methodology:**

**Data Cleaning and Preparation:** The first step is to clean and prepare the sales data to make it usable for analysis. This involves removing missing or duplicate data, correcting data types, modeling and formatting the data.

**Exploratory Data Analysis:** I used SQL to perform exploratory data analysis (EDA). This involved aggregating and transforming the data to uncover patterns, relationships, and anomalies in the data. I utilized SQL functions like SUM, AVG, COUNT, MAX, MIN, DATEDIFF, and NTILE to calculate various metrics such as monetary value, average monetary value, frequency of purchases, last order date, recency, and more. I also employed grouping and grouping using GROUP BY to consolidate the data based on certain attributes such as customer name.

Furthermore, I used conditional statements (CASE WHEN) to categorize customers into different segments based on the values of the calculated metrics. By doing this, I was able to gain insights into the behavior and patterns of different types of customers, which could inform decision making and business strategy. The results of my EDA were stored in a temporary table (#rfm) and later used to generate visualizations to communicate my findings.

**Data Visualization:** Finally, the results of the customer segmentation analysis are visualized using data visualization tools such as bar charts and pie charts. These visualizations help to provide a clear understanding of the customer segments and their distribution.

**Lets dive right in.**

In this session, we're taking a dive into the thrilling world of data-driven questions and how we can use SQL and Exploratory Data Analysis to get the answers we need! Now, we won't be diving into the nitty-gritty of data preparation and cleaning (that's for another day), but instead, we'll jump right into the fun stuff! Get ready to see some code snippets, eye-catching visuals, and maybe even a few laughs along the way. So buckle up, grab your coffee (or your favorite snack), and let's dive in!

**EXPLORATORY DATA ANALYSIS**

Starting off any Exploratory Data Analysis (EDA) project is a critical first step, and my analysis of customer sales data was no different. To kick things off, I carefully selected the columns that were relevant to the questions I was trying to answer. This step is crucial as it lays the foundation for the rest of the analysis and determines the direction in which we move forward. A thorough understanding of the data, its structure, and what it represents is essential before moving on to the next stage of EDA. I made sure I had a good grip on the columns I was working with, as they would inform my next steps in uncovering insights and drawing meaningful conclusions from the data. So, let's get cracking!

SELECT productCode,orderNumber,orderDate,productName,productLine,productVendor,quantityInStock,quantityOrdered,MSRP,

MSRP,buyPrice,priceEach,status,customerName,customerNumber,amount,country

FROM [sales\_database].[dbo].[Sales data]

**QUESTIONS**

I came up with a couple of questions for this project and we shall be answering them all, using SQL and display visuals for some of the questions.

Here are some of the questions we shall be answering.

1. **What was the total quantity of each product sold and the total revenue generated?**

SELECT productName, SUM(quantityOrdered) as TotalQuantity, SUM(amount) as TotalRevenue

FROM [sales\_database].[dbo].[Sales data]

GROUP BY productName

1. **What was the average price of each product and the average revenue per order?**

SELECT AVG(priceEach) as AveragePrice, AVG(amount) as AverageRevenuePerOrder

FROM [sales\_database].[dbo].[Sales data]

1. **Which product line and vendor generated the highest revenue?**

SELECT productLine, productVendor, SUM(amount) as TotalRevenue

FROM [sales\_database].[dbo].[Sales data]

GROUP BY productLine, productVendor

ORDER BY TotalRevenue DESC

LIMIT 1

1. **Who was the most frequent customer and what was their total spend?**

SELECT customerName, SUM(amount) as TotalSpend

FROM [sales\_database].[dbo].[Sales data]

GROUP BY customerName

ORDER BY TotalSpend DESC

LIMIT 1

1. **Which country generated the most revenue?**

SELECT country, SUM(amount) as TotalRevenue

FROM [sales\_database].[dbo].[Sales data]

GROUP BY country

ORDER BY TotalRevenue DESC

LIMIT 1

1. **What was the sales trend over time, was there a specific period with high or low sales?**

SELECT YEAR(orderDate) as Year, SUM(amount) as TotalRevenue

FROM [sales\_database].[dbo].[Sales data]

GROUP BY YEAR(orderDate)

ORDER BY YEAR(OrderDate) ASC

1. **Was there any relationship between quantity ordered and revenue?**

SELECT quantityOrdered, SUM(amount) as TotalRevenue

FROM [sales\_database].[dbo].[Sales data]

GROUP BY quantityOrdered

1. **Total number of orders per customer.**

SELECT customerName, COUNT(orderNumber) AS TotalOrders

FROM [sales\_database].[dbo].[Sales data]

GROUP BY customerName;

1. **Total revenue generated by each product line.**

SELECT productLine, SUM(quantityOrdered \* priceEach) AS TotalRevenue

FROM [sales\_database].[dbo].[Sales data]

GROUP BY productLine;

1. **Most frequently ordered product.**

SELECT productName, COUNT(productName) AS Orders

FROM [sales\_database].[dbo].[Sales data]

GROUP BY productName

ORDER BY Orders DESC

LIMIT 1;

1. **Number of orders shipped vs. cancelled.**

SELECT status, COUNT(status) AS Orders

FROM [sales\_database].[dbo].[Sales data]

GROUP BY status;

1. **The most expensive product sold.**

SELECT productName, MAX(priceEach) AS MostExpensiveProduct

FROM [sales\_database].[dbo].[Sales data];

GROUP BY productName

ORDER BY 2 DESC

1. **The total revenue generated by each vendor.**

SELECT productVendor, SUM(quantityOrdered \* priceEach) AS TotalRevenue

FROM [sales\_database].[dbo].[Sales data]

GROUP BY productVendor;

1. **The difference in revenue between shipped and cancelled orders.**

SELECT status, SUM(amount) AS Revenue

FROM [sales\_database].[dbo].[Sales data]

GROUP BY status;

1. **Who are the best customers?**

This is where RFM Analysis comes in.

RFM (Recency, Frequency, Monetary) analysis is a marketing technique used to analyze customer behavior and identify valuable customers. I performed RFM Analysis on the customer sales data by aggregating the sales transactions for each customer, and calculating the following metrics:

**Recency**: The number of days since the customer's last purchase

**Frequency**: The number of purchases made by the customer

**Monetary**: The total amount spent by the customer

Using SQL, I performed RFM Analysis as follows;

**Recency:**

WITH customer\_max\_order\_date AS (

SELECT

customerNumber,

MAX(orderDate) AS max\_order\_date

FROM

[sales\_database].[dbo].[Sales data]

GROUP BY

customerNumber

),

recency AS (

SELECT

customerNumber,

DATEDIFF(CURDATE(), max\_order\_date) AS recency

FROM

customer\_max\_order\_date

)

SELECT

customerNumber,

recency

FROM

The code starts by creating a subquery named customer\_max\_order\_date that finds the latest order date for each customer by using the MAX() function and grouping by customerNumber.

Then, another subquery named recency is created which uses the previous subquery to calculate the difference between the current date and the latest order date for each customer. This calculation is performed using the DATEDIFF() function.

Finally, the main query selects the customerNumber and recency columns from the recency subquery, which gives the final result set containing the recency value for each customer.

**Frequency:**

SELECT

customerNumber,

COUNT(orderNumber) AS frequency

FROM

orders

GROUP BY

customerNumber;

**Frequency:**

SELECT

customerNumber,

SUM(amount) AS monetary

FROM

orders

GROUP BY

customerNumber;

By combining these metrics, customers can be segmented into different groups based on their recency, frequency, and monetary value, which can inform targeted marketing strategies.

DROP TABLE IF EXISTS #rfm

;WITH rfm AS

(

SELECT

CUSTOMERNAME,

SUM(amount) MonetaryValue,

AVG(amount) AvgMonetaryValue,

COUNT(orderNumber) Frequency,

MAX(orderDate) last\_order\_date,

(SELECT MAX(orderDate) FROM [sales\_database].[dbo].[Sales data]) max\_order\_date,

DATEDIFF(DD, MAX(orderDate), (SELECT MAX(orderDate) FROM [sales\_database].[dbo].[Sales data])) Recency

FROM [sales\_database].[dbo].[Sales data]

GROUP BY customerName

),

rfm\_calc AS

(

SELECT r.\*,

NTILE(4) OVER (ORDER BY Recency DESC) rfm\_recency,

NTILE(4) OVER (ORDER BY Frequency) rfm\_frequency,

NTILE(4) OVER (ORDER BY MonetaryValue) rfm\_monetary

FROM rfm r

)

SELECT

c.\*, rfm\_recency+ rfm\_frequency+ rfm\_monetary AS rfm\_cell,

CAST(rfm\_recency AS VARCHAR) + CAST(rfm\_frequency AS VARCHAR) + CAST(rfm\_monetary AS VARCHAR)rfm\_cell\_string

INTO #rfm

FROM rfm\_calc c

SELECT customerName , rfm\_recency, rfm\_frequency, rfm\_monetary,

CASE

WHEN rfm\_cell\_string in (111, 112 , 121, 122, 123, 132, 211, 212, 114, 141) THEN 'lost\_customers' --lost customers

WHEN rfm\_cell\_string in (133, 134, 143, 244, 334, 343, 344, 144) THEN 'slipping away, cannot lose' -- (Big spenders who haven�t purchased lately) slipping away

WHEN rfm\_cell\_string in (311, 411, 331) THEN 'new customers'

WHEN rfm\_cell\_string in (222, 223, 233, 322) THEN 'potential churners'

WHEN rfm\_cell\_string in (323, 333,321, 422, 332, 432) THEN 'active' --(Customers who buy often & recently, but at low price points)

WHEN rfm\_cell\_string in (433, 434, 443, 444) THEN 'loyal'

END rfm\_segment

FROM #rfm

The above SQL snippet is creating a temporary table called #rfm, which calculates the Recency, Frequency, and Monetary Value (RFM) of each customer. The first WITH clause calculates the Monetary Value, Average Monetary Value, Frequency, last order date, maximum order date and Recency of each customer by aggregating the sales data. The second WITH clause calculates the RFM score of each customer by dividing the RFM metrics into 4 equal parts (i.e., quartiles) using the NTILE function.

The SELECT statement after that joins the calculated RFM score with the customer data and creates a new table with the rfm\_cell and rfm\_cell\_string columns. The final SELECT statement categorizes the customers into different segments based on the rfm\_cell\_string value.

Overall, this SQL snippet can be used to identify the customer segments based on their RFM metrics and analyze their behavior in order to improve customer retention and increase revenue.

**Insights:**

**Project Results:**

The project results show that most of the customers belong to the "lost customers" and "potential churners" segment, which requires attention to retain themThis result indicates that the company needs to focus on retaining its existing customers while also attracting new ones.

**Conclusions and Recommendations:**

The customer sales analysis project provides valuable insights into customer behavior and allows the company to better understand its customers. Based on the results, it is recommended that the company focus on retaining its existing customers by providing personalized experiences and improving customer engagement. Additionally, the company should also focus on attracting new customers through effective marketing strategies.

In conclusion, this project provides a comprehensive analysis of customer sales data and helps the company to make informed decisions based on customer behavior and trends. By continually analyzing and understanding customer sales data, the company can continuously improve its product and services to better meet customer needs.