Axon Car Sales Dashboard Documentation

Introduction:

A small Company Axon, which is retailer selling classic cars, is facing issues in managing and analyzing their sales data. The sales team is struggling to make sense of the data and they do not have a centralized system to manage and analyze the data. The management is unable to get accurate and up-to-date sales report, which is effecting the decision making process. To manage this issue, the company has introduced a Business Intelligence (BI) tool that can help them manage and analyze their sales data effectively.

Goal:

The goal of the capstone project is to design and implement a BI solution using Power BI and SQL that can help the company manage and analyze their sales data effectively.

About Database:

The database schema contains 8 tables and a brief description of these tables is:

1. Customers: stores customer's data.

2. Products: stores a list of scale model cars.

3. ProductLines: stores a list of product line categories.

4. Orders: stores sales orders placed by customers.

5. OrderDetails: stores sales order line items for each sales order.

6. Payments: stores payments made by customers based on their accounts.

7. Employees: stores all employee information as well as the organization structure such as who reports to whom.

8. Offices: stores sales office data.

Optimizing Sales Insights:

The process begins by bringing data from a MySQL database into Power BI. We then clean and prepare the data for a closer look. Using Power BI, we create interactive dashboards and reports to help the sales team and management understand the information better. SQL is applied to perform advanced analysis, revealing insights to enhance sales strategies. This approach ensures that the management can access real-time data for making well-informed decisions.

Created a Calendar Table:

The Calendar table is a helpful addition for time-related analysis. It contains a list of dates with useful columns like year, quarter, month, and day. This table enhances time-based calculations and is valuable for creating meaningful reports and dashboards.

Key Measures:

1. Total sales

DAX FORMULA:

```
Total_Sales = SUMX('classicmodels orderdetails', 'classicmodels
orderdetails'[quantityOrdered] * 'classicmodels orderdetails'[priceEach])
```

- Total_Sales measure calculates the overall sales by iterating through each row in the 'classicmodels orderdetails' table, multiplying the 'quantityOrdered' by 'priceEach' for each row, and then summing up these individual values.
- This results in the total revenue generated across all orders.

2. Total profit

DAX FORMULA:

```
Total_Profit = SUMX('classicmodels orderdetails', 'classicmodels
orderdetails'[quantityOrdered] * ('classicmodels orderdetails'[priceEach] -
RELATED('classicmodels products'[buyPrice])))
```

- The Total_Profit measure calculates the overall profit by iterating through each row in the 'classicmodels orderdetails' table. It multiplies the 'quantityOrdered' by the difference between 'priceEach' and the corresponding 'buyPrice' from the related 'classicmodels products' table for each row.
- The individual profit values are then summed up, providing the total profit across all orders.

3. Profit Percentage

DAX FORMULA:

```
Profit_Percentage = DIVIDE([Total_Profit],[Total_Sales])
```

- Profit_Percentage measure calculates the profit percentage by dividing the Total_Profit by the Total_Sales.
- It gives insight into the profitability of sales, indicating what portion of the revenue is retained as profit.
- The result is expressed as a percentage, providing a key metric for evaluating the financial performance of the sales.

4. Average Price

DAX FORMULA:

```
Avg_Price = AVERAGE('classicmodels orderdetails'[priceEach])])
```

- The Avg_Price measure computes the average unit price ('priceEach') of products in the 'classicmodels orderdetails' table.
- It provides a metric to understand the typical price at which products are sold.

5. Product Lines

DAX FORMULA:

```
Product_Lines = DISTINCTCOUNT('classicmodels products'[productLine])
```

- The Product_Lines measure counts the unique product lines in the 'classicmodels products' table.
- It gives a clear view of the variety offered by Axon.

6. Total Customers

DAX FORMULA:

```
Total_Customers = COUNTROWS('classicmodels customers')
```

The Total_Customers measure counts the total number of rows (customers) in the 'classicmodels customers' table, representing the overall customer count for Axon.

7. Total Orders

DAX FORMULA:

```
TotalOrders = COUNT('classicmodels orderdetails'[orderNumber])
```

- The TotalOrders measure counts the total number of sales orders in the 'classicmodels orderdetails' table by calculating the distinct count of 'orderNumber'.
- This metric provides a comprehensive view of the overall number of sales transactions for Axon.

8. Total Quantity Ordered

DAX FORMULA:

```
TotalQuantityOrdered = SUM('classicmodels orderdetails'[quantityOrdered])
```

- The TotalQuantityOrdered measure is derived by summing the values in the 'quantityOrdered' column within the 'classicmodels orderdetails' table.
- This measure provides a consolidated figure representing the total quantity of products ordered across all transactions.

9. Total Quantity in stock

DAX FORMULA:

```
TotalQuantityInStock = SUM('classicmodels products'[quantityInStock])
```

- TotalQuantityInStock is a measure that sums up the 'quantityInStock' column in the 'classicmodels products' table.
- It provides a consolidated view of the total quantity of products currently available in stock.
- This measure is valuable for efficient inventory management, offering insights into the overall stock availability for Axon's various product lines.

10. Sales Trend

DAX FORMULA:

```
SalesTrend = CALCULATE([Total_Sales], ALL('Calendar'[Date]))
```

- The SalesTrend measure is calculated using the CALCULATE function, taking the [Total_Sales] measure and applying the ALL function to the 'Calendar'[Date] column.
- This implies that a dedicated calendar table has been created, and the 'Date' column from this table is utilized to analyze sales trends.

11. Sum of the sales

DAX FORMULA:

```
Sum_of_Sales = SUM('classicmodels orderdetails'[priceEach])
```

- The Sum_of_Sales measure calculates the sum of the 'priceEach' column from the 'classicmodels orderdetails' table.
- This measure provides the total of the sales amounts for all individual line items in the orders.

12 .Year to date sales

DAX FORMULA:

```
YTD_Sales = TOTALYTD([Total_Sales],'Calendar'[Date])
```

- The YTD_Sales measure calculates the year-to-date total sales by considering the cumulative sum of sales from the beginning of the year up to the selected date.
- This measure provides insights into the ongoing performance and trends in sales throughout the year.

13. Total number of products

DAX FORMULA:

```
Total_products = COUNTROWS('classicmodels products')
```

- The Total_Products measure counts the total number of unique products available in the 'classicmodels products' table.
- It gives a complete view of the products in Axon's inventory, helping in managing stock and understanding the range of offerings.

14. Total number of payments

DAX FORMULA:

```
TotalPayment = SUM('classicmodels payments'[amount])
```

Total_Payment measures the overall sum of payments recorded in the 'classicmodels payments' table, offering a clear perspective on the total monetary transactions.

15. Total sales for previous month

DAX FORMULA:

```
Total_Sales_PreviousPeriod =
CALCULATE ([Total_Sales],PREVIOUSMONTH('Calendar'[Date]))
```

The Total_Sales_PreviousPeriod measure calculates the total sales for the previous month by using the CALCULATE function and referencing the Total_Sales measure within the context of the previous month obtained from the 'Calendar' table.

16. Sales Growth percentage

DAX FORMULA:

```
SalesGrowthPercentage = (([Total_Sales] - [Total_Sales_PreviousPeriod]) /
[Total_Sales_PreviousPeriod]) * 100
```

This measure calculates the percentage change in sales between the current period and the previous period, providing insights into the sales performance trend.

Key performance metrics :

- The implemented measures in Axon Car Sales Dashboard provide key insights into various aspects of the business.
- The Total_Sales and Total_Profit metrics provide a clear view of revenue generation and profitability.
- The Profit Percentage indicates the profit margin, while Avg_Price gives an average unit price analysis.
- The Product_Lines measure counts unique product lines, assisting in assessing the product variety.
- Total_Customers and TotalOrders help understand the customer base and the overall number of orders.
- TotalQuantityInStock aids in inventory management, and SalesTrend considers the sales trend over time.
- The Total_Products metric provides a quick look at the overall variety of products in Axon's inventory.
- TotalPayment reflects the total amount received, while Total_Sales_PreviousPeriod and SalesGrowthPercentage offer insights into sales growth and performance trends.

- The inclusion of the Calendar table enhances time-based analysis. These metrics collectively empower the Axon team to make data-driven decisions and optimize sales strategies.

Dashboard and Sql Query insights:

Total Sales and Profit:

With a total profit of $3.83 million, the company has earned $9.6 million in total sales, showcasing strong performance.

Customer and Order Metrics:

The company has a customer base of 94 and has processed a total of 2,996 orders.

Top Sales Contributor:

USA emerged as the highest sales contributor, achieving $3.1 million in total sales.

Profitable Product Line:

The 'Classic Cars' product line stands out with the highest profit margin, reaching an impressive 39.89%.

Outstanding Product Line Sales:

Classic Cars achieved an outstanding total sale of $3.8 million.

Sales Performance Over Time:

In 2004, the company achieved its highest total sales compared to other years.

Top Order Quantity product:

The product '1992 Ferrari 360 Spider Red' achieved the highest order quantity, totaling 1808 units.

Stock Quantity:

The available stock quantity stands at 0.56 million units.

Product Inventory:

The total number of distinct products in Axon's inventory is 110.

Top Product Line:

'Classic Cars' Leads with the Highest Order Quantity of 35,582 Units.

Vendor Excellence: Sales vs. Profitability:

- ✓ Classic Metal Creations has emerged as the top-performing vendor, showcasing exceptional sales and contributing significantly to the company's overall success.
- ✓ Additionally, Unimax Art Galleries stands out as the leading vendor when considering total profit, demonstrating its significant contribution to the company's financial success.

Conclusion:

This detailed sales analysis with Power BI and SQL has given valuable insights into how Axon Car Sales is doing. Analyzing vital metrics like sales, profit, customer engagement, and product performance has given us valuable insights into Axon's market standing. The identified top-performing vendors, popular product lines, and historical sales trends equip the management with actionable insights for strategic decision-making. This analysis helps Axon improve how they sell cars and lets them respond better to changes in the market. By using Power BI for visuals and SQL for detailed analysis, Axon Car Sales can make smarter decisions based on data in the competitive automotive retail industry.