Daily Grind Coffee Sales Performance Dashboard

A Power BI Case Study

Prepared By:
Ardonna Cardines
Data Analyst

Date:
September 2025

DAILY
GRIND
COFFEE

Table of Contents

Project Overview

Objective

The objective of this project was to transform raw transactional data from a coffee shop into a **scalable business intelligence solution**. Beyond simply visualizing the flat file, the project involved restructuring the dataset into a star schema, creating new measures with DAX, and designing interactive dashboards that deliver actionable insights for sales, customer behavior, operations, and finance.

Use Case

This project simulates a real-world business intelligence scenario for a fictional coffee shop chain, *Daily Grind Coffee*. Leadership needed a centralized reporting system that could answer questions such as:

- How are sales trending over time, locations, and products?
- What role do loyalty customers play compared to non-loyalty customers?
- Which products and categories are driving revenue growth?
- How do operational factors like time of day and store hours affect performance?

By re-engineering the dataset into a relational schema, the solution was built with scalability in mind, ensuring it could grow with the business and integrate seamlessly into a relational database such as Microsoft SQL Server.

Audience

1. **Executives & Managers**: High-level KPIs and trends to guide strategic decisions.
2. **Store Managers & Operations Teams**: Insights into staffing, peak hours, and store performance.
3. **Marketing Teams**: Customer loyalty and product insights to refine promotions and campaigns.

Scope

The reporting solution consists of three key dashboards that provide a holistic view of the coffee shop's operations:

1. **Sales Overview** – Revenue, transactions, average order value, sales trends.
2. **Product Performance** – Top products, category analysis, size preferences.
3. **Customer Insights** – Loyalty program analysis, repeat vs. new customers, basket size

By taking the dataset beyond its original flat-file structure, this project demonstrates the full analytics lifecycle: **data preparation, schema design, DAX modeling, and dashboard storytelling**.

Methodology

The original dataset was a flat transactional table that I restructured into a relational star schema to improve scalability and analytical flexibility. By creating separate dimension tables for products, customers, employees, dates, and locations, and linking them to a central sales fact table, I enabled easier slicing, filtering, and drill-down analysis in Power BI. This approach mirrors real-world business intelligence practices, where normalized and well-modeled data structures lead to faster queries, simplified DAX calculations, and more maintainable dashboards.

Data Sources

The dataset used in this project represents point-of-sale transaction records from a coffee shop chain. It contains 15 columns and 4,998 transactions across multiple store locations, including detailed information on products, customers, and employees.

File Name: *coffee_shop_sales.csv*

Data Fields

- **Transaction Details**: Transaction ID, Date, Time, Location, Payment Method
- **Product Information**: Product ID, Product Name, Product Category, Size, Unit Price, Quantity, Total Price
- **Customer Information:** Customer ID, Loyalty Points Earned
- **Employee Information**: Barista ID

Preparation Tools

The data cleansing, preparation and schema creation were done in Power Query. Additional measures were created using DAX.

Power Query: For cleaning and transforming data before loading into the data model.

DAX (Data Analysis Expressions): For calculated measures and KPIs.

In the following sections, all DAX formulas and measures are presented in a monospaced font (Consolas) to distinguish them from narrative text.

Data Cleaning and Preparation

The dataset required several preparation steps before analysis:

Missing Values

- *Size* had missing values for non-beverage products (e.g., bakery items).
    - Replaced *NA* to *OneSize*
- *Customer ID* was missing for guest purchases (non-loyalty customers).
    - Customers without IDs were converted to *Guest*.

Transactions without customer IDs were treated as guest purchases. The purpose of this decision kept analysis in mind by helping to distinguish between loyalty and non-loyalty customers.

Duplicates

Verified that each transaction_id was unique.

Data Types

- Converted transaction_date to Date format.
- Converted transaction_time to Time format.
- Ensured numeric fields (unit_price, quantity, total_price) were properly typed.

Transformations

Using Power Query and DAX in Power BI, the following transformations were applied

Calculated Columns

```
customer_type =  Table.AddColumn(#"Replaced Value", "customer_type", each if
[loyalty_points_earned] = 0 then "Non-Loyalty" else "Loyalty")

hour = Table.AddColumn(#"Changed Type1", "hour", each
Time.Hour([transaction_time]))

minute = Table.AddColumn(#"Added Custom1", "minute", each
Time.Minute([transaction_time]))

second = Table.AddColumn(#"Added Custom2", "second", each
Time.Second([transaction_time]))

weekday_name = Table.AddColumn(#"Reordered Columns", "weekday_name", each
Date.DayOfWeekName([transaction_date]))

is_weekend_flag = Table.AddColumn(#"Changed Type2", "is_weekend_flag", each
if [weekday_name] = "Saturday" or [weekday_name]="Sunday" then "Weekend" else
"Weekday")

am_pm_flag = Table.AddColumn(#"Changed Type3", "am_pm_flag", each if[hour] =
"13" or [hour] = "14" or [hour] = "15" or [hour] = "16"or [hour] = "17"or
```

```
[hour] = "18"or [hour] = "19"or [hour] = "20"or [hour] = "21" then "PM" else "AM")
```

time_of_day = `Table.AddColumn(#"Reordered Columns1", "time_of_day", each if[hour] = "8" then "Breakfast" else if [hour] = "9" then "Breakfast" else if [hour] = "10" then "Breakfast" else if [hour] = "11" then "Lunch" else if [hour] = "12" then "Lunch" else if [hour] = "13" then "Lunch" else if [hour] = "14" then "Lunch" else if [hour] = "15" then "Late Lunch" else if [hour] = "16" then "Late Lunch" else "Dinner")`

The original flat file consisted of 15 columns and the ending Fact Sales table consisted of 23 columns.

Measures

total_sales = `total_sales = SUM(FactSales[total_price])`

total_sales_ytd `=TOTALYTD(SUM('FactSales'[total_price]), 'DimDate'[Date])`

total_sales_py = `CALCULATE(SUM('FactSales'[total_price]), DimDate[Date], SAMEPERIODLASTYEAR(DimDate[Date]))`

total_sales_YoY_% `=DIVIDE([ytd-py], [total_sales_ytd])`

total_sales_loyalty = `CALCULATE([total_sales], 'FactSales'[customer_type] IN { "Loyalty" })`

total_sales_nonloyalty = `CALCULATE([total_sales], 'FactSales'[customer_type] IN { "Non-Loyalty" })`

count_transactions = `CALCULATE(DISTINCTCOUNT(FactSales[transaction_id]))`

count_transactions_loyalty = `count_transactions_loyalty = CALCULATE(COUNTA('FactSales'[transaction_id]), 'FactSales'[customer_type] IN { "Loyalty" })`

count_transactions_nonloyalty = `CALCULATE(COUNTA('FactSales'[transaction_id]), 'FactSales'[customer_type] IN { "Non-Loyalty" })`

average_basket_order_size = `DIVIDE([total_sales], COUNTA('FactSales'[transaction_id]))`
`average_basked_order_size_loyalty = DIVIDE([total_sales_loyalty], [count_transactions_loyalty])`

```
average_basket_order_size_nonloyalty = DIVIDE([total_sales_nonloyalty],
[count_transactions_nonloyalty])

count_customers = CALCULATE(DISTINCTCOUNT(FactSales[customer_id]))

count_customers_loyalty =  CALCULATE([count_customers],
'FactSales'[customer_type] IN { "Loyalty" })

count_customers_nonloyalty =
CALCULATE(
    [count_customers],
    'FactSales'[customer_type] IN { "Non-Loyalty" }
)

active_customers = DISTINCTCOUNT(FactSales[customer_id])

retained_customers = VAR current_month = MAX('DimDate'[Date]) VAR
previous_month = EOMONTH(current_month, -1)VAR customers_previous_month =
CALCULATETABLE(VALUES(FactSales[customer_id]), DATESINPERIOD('DimDate'[Date],
previous_month, 1, MONTH)) VAR customers_current_month =
CALCULATETABLE(VALUES(FactSales[customer_id]), DATESINPERIOD('DimDate'[Date],
current_month, 1, MONTH)) RETURN
COUNTROWS(INTERSECT(customers_previous_month, customers_current_month))

retention_rate = DIVIDE([retained_customers], CALCULATE([Active Customers],
DATEADD('DimDate'[Date], -1, MONTH)))
```

Data Modeling

The original dataset was provided as a flat transactional file containing details of every purchase. While this format was suitable for raw storage, it was not ideal for analysis and reporting. To optimize performance and enable flexible analysis in Power BI, I restructured the dataset into a star schema.

Star Schema Design

At the center of the schema is the **FactSales** table, which records all transactional details. Supporting this fact table are several dimension tables that describe key entities such as products, customers, employees, dates, and store locations.

FactSales: transaction_id, transaction_date, transaction_time, product_id, customer_id, barista_id, location, payment_method, quantity, unit_price, total_price, loyalty_points_earned, customer_type, hour, minute, second, weekday_name, is_weekend_flag, am_pm_flag, time_of_day

DimProducts: product_id, product_name, product_category, size

DimCustomers: customer_id, lifetime_spend (calculated), lifetime_points (calculated), is_repeat_customer_flag (calculated)

DimEmployees: barista_id

DimDate: date, year, month, quarter, weekday (generated in Power BI)

DimLocation: location, store_type (e.g., Airport, University, Mall, Downtown)

Purpose of Restructuring

The purpose of moving from a flat file to a star schema was to follow business intelligence best practices and improve the analytical value of the data:

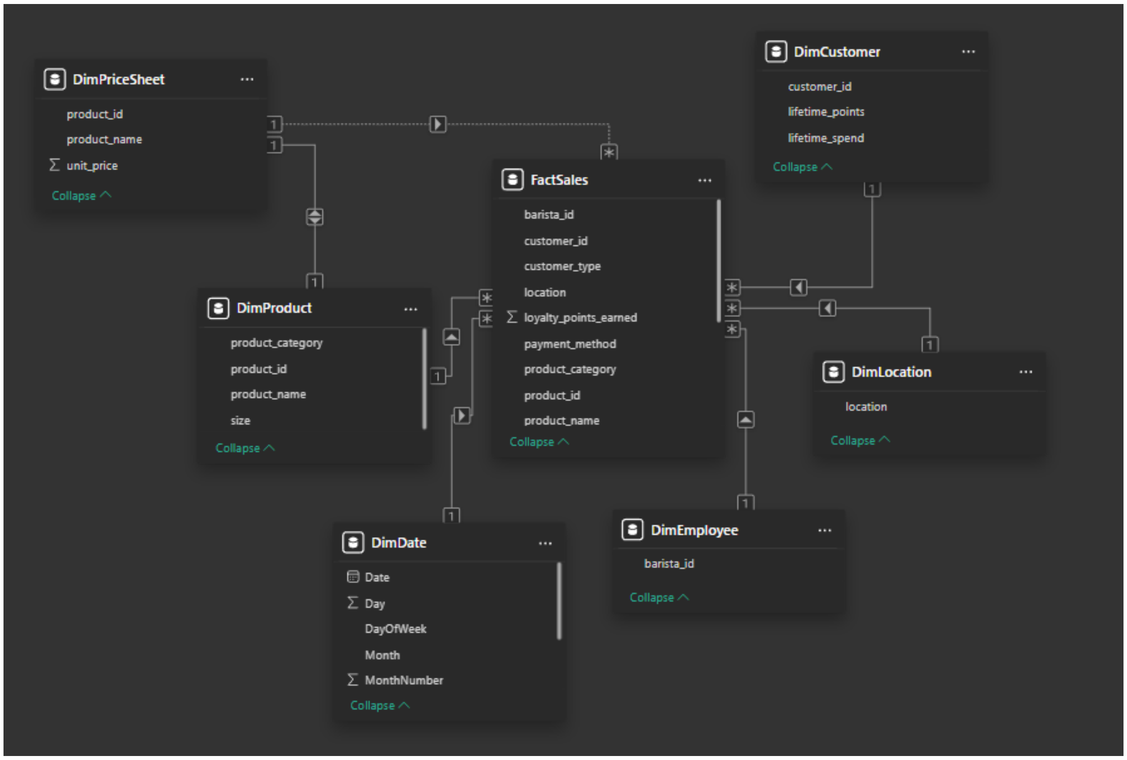
> ***Flexibility***: Enables slicing and filtering by product, customer, employee, location, and time dimensions.
>
> ***Performance***: Aggregations and DAX calculations run faster on well-modeled tables.
>
> ***Clarity***: Dashboards become easier to interpret, since metrics are organized around business entities (e.g., "Products" vs. raw transaction fields).
>
> ***Scalability***: New datasets (e.g., supplier data, marketing campaigns) can be integrated without redesigning the entire model.

Implementation in Power BI

Using Power Query, the dataset was transformed into multiple tables by:

- Duplicating the original dataset and extracting unique values to form dimension tables (Products, Customers, Employees, Locations).
- Generating a Date table in Power BI to support time intelligence functions.
- Establishing relationships between the FactSales table and dimension tables using primary/foreign keys (e.g., product_id, customer_id, barista_id).

This structured approach mirrors how data is modeled in professional BI environments and ensures that the dashboards are not only visually appealing but also analytically robust.




```
FactSales[product_id] → DimProducts[product_id]

FactSales[customer_id] → DimCustomers[customer_id]

FactSales[barista_id] → DimEmployees[barista_id]

FactSales[location] → DimLocation[location]

FactSales[transaction_date] → DimDate[Date]
```

Dashboard Design

Wireframes & Layout Planning

Before building the dashboards, I sketched wireframes to define the structure and flow of information. Each page was designed to follow a top-down narrative: starting with high-level KPIs, followed by trend analysis, and ending with detailed breakdowns for deeper insights.

The report consists of three dashboards across separate pages:

1. **Sales Overview**

   Filters: Product Category, Date, Location

   KPIs: Total Sales, YoY Growth %, Average Order Value, Total Transactions

   Charts and Visuals: Insights (summarizes key data points in dashboard), Total Sales by Month (line chart), Payment Method Share (donut chart), Total Sales by Customer Type (donut chart), Total Sales by Barista (stacked bar chart)

2. **Product Performance**

   Filters: Location and Date

   KPIs: Insights (summarizes key data points in dashboard), Total Sales, Average Basket Size, Total Transactions

   Charts and Visuals: Top Products by Sales, Total Sales by Product Category, Product Drill-Through, Coffee Size Preference, Tea Size Preference

3. **Customer Insights**

   Filters: Date, Location and Date

   KPIs: Total Customers, Total Repeat Customers, Total New Customers, Customer Retention Rate

   Charts and Visuals: Customer Favorite (card), Customer Least Favorite (card), Loyalty vs. Non-Loyalty Sales (line chart), Customer Retention Rate (line chart), Top Customers by Spend (stacked bar chart), Customer Type Sales by Location (stacked column chart), Most Popular Time of Day (stacked chart)

Interactivity

To maximize usability, the dashboards included interactive features:

- Slicers: Filter by Date, Location, and Product Category

- Drill-through: Ability to click on a product or customer to see detailed transaction-level data.

Design Rationale

The dashboards were designed to answer key business questions in a logical, user-friendly flow:

- Executives can quickly scan overall KPIs and growth trends.
- Product managers can dive into bestsellers and category breakdowns.
- Marketing can explore loyalty behavior and customer segmentation.
- Operations managers can track staffing efficiency and peak times.
- Finance can monitor revenue drivers and customer profitability.

Key Metrics & Visuals

KPIs

Total Sales
Definition: The sum of all transaction values.
Formula (DAX): `SUM(FactSales[total_price])`

Total Sales Year-to-Date
Definition: The sum of all transaction values year-to-date.
Formula (DAX): `TOTALYTD(SUM('FactSales'[total_price]), 'DimDate'[Date]`

Total Sales Prior Year
Definition: The sum of all transaction values for the prior year.
Formula (DAX): `CALCULATE(SUM('FactSales'[total_price]), DimDate[Date], SAMEPERIODLASTYEAR(DimDate[Date]))`

YoY Growth %
Definition: The percentage increase or decrease in total sales year-over-year.
Formula (DAX): `SUM(FactSales[total_price])`

Average Order Value
Definition: The average of total sales.
Formula (DAX): `DIVIDE([total_sales], COUNTA('FactSales'[transaction_id]))`

Total Transactions

Definition: The count of total orders.

Formula (DAX): `CALCULATE(DISTINCTCOUNT(FactSales[transaction_id]))`

Active Customers

Definition: The distinct count of all customers.

Formula (DAX): `DISTINCTCOUNT(FactSales[customer_id])`

Retained Customers

Definition: The count of repeat customers.

Formula (DAX):
```
VAR current_month = MAX('DimDate'[Date]) VAR
previous_month = EOMONTH(current_month, -1)VAR
customers_previous_month =
CALCULATETABLE(VALUES(FactSales[customer_id]),
DATESINPERIOD('DimDate'[Date], previous_month, 1, MONTH)) VAR
customers_current_month =
CALCULATETABLE(VALUES(FactSales[customer_id]),
DATESINPERIOD('DimDate'[Date], current_month, 1, MONTH)) RETURN
COUNTROWS(INTERSECT(customers_previous_month,
customers_current_month))
```

Customer Retention Rate

Definition: The percentage of customers who make another purchase in a time period.

Formula (DAX):
```
DIVIDE([retained_customers], CALCULATE([Active Customers],
DATEADD('DimDate'[Date], -1, MONTH)))
```

Supporting Metrics

Total Sales by Month

Definition: The total sales by month to show trends and patterns over time.

Chart or Visual Type: Line Chart

Payment Method Share

Definition: The payment types that customers use for transactions.

Chart or Visual Type: Donut Chart

Total Sales by Customer Type

Definition: The count of new customers.

Chart or Visual Type: Donut Chart

Total Sales by Barista
Definition: The total sales by barista.
Chart or Visual Type: Bar Chart
Top Products by Sales
Definition: The top ten products based on sales.
Chart or Visual Type: Stacked Bar Chart

Total Sales by Product Category
Definition: The count of new customers.
Chart or Visual Type: Line Chart

Product Drill-Through
Definition: The product details that can be drilled down to granular level.
Chart or Visual Type: Matrix Table

Coffee Size Preference
Definition: The preferences of customers on drink size for coffee.
Chart or Visual Type: Pie Chart

Tea Size Preference
Definition: The preferences of customers on drink size for tea.
Chart or Visual Type: Pie Chart

Customer Favorite
Definition: The best-selling product among all customers (both loyalty and non-loyalty).
Chart or Visual Type: Card

Customer Least Favorite
Definition: The least-selling product among all customers (both loyalty and non-loyalty).
Chart or Visual Type: Card

Loyalty vs. Non-Loyalty Sales
Definition: The total sales of Loyalty and Non-Loyalty customers over a period of time.
Chart or Visual Type: Line Chart

Customer Retention Rate
Definition: The customer retention rate over a period of time.
Chart or Visual Type: Line Chart

Top Customers by Spend
Definition: The top ten customers who have spent the most.
Chart or Visual Type: Stacked Bar Chart

Customer Type Sales by Location
Definition: The count of new customers.
Chart or Visual Type: Stacked Column Chart

Most Popular Time of Day
Definition: The count of new customers.
Chart or Visual Type: Stacked Chart

Insights & Findings

Sales Performance

Overall revenue showed consistent growth over the period, with noticeable spikes during weekends and holiday seasons.

Average Order Value (AOV) remained stable, but transactions increased significantly in the University and Airport locations, suggesting higher customer traffic rather than upselling.

Product Trends

Coffee and Bakery products were the highest revenue drivers, with Croissants and Cappuccinos consistently ranking in the top 5 products.

Beverage sizes showed a strong preference for *Medium*, indicating an opportunity to optimize pricing strategies for upsizing to Large.

Seasonal items (e.g., limited-edition drinks) demonstrated short bursts of high sales, highlighting the effectiveness of promotions.

Customer Insights

Loyalty customers accounted for nearly half of sales revenue, with higher average basket sizes compared to guest customers.

Guest purchases were frequent but lower in value, indicating an opportunity to promote loyalty enrollment to increase retention and spending.

Repeat customers visited more frequently at the Downtown and University locations, suggesting strong brand loyalty in those areas.

Summary of Business Impact

The dashboards reveal clear opportunities to:

Optimize staffing and inventory around peak hours and high-performing locations.

Refine product pricing and promotions, especially for beverage sizes and seasonal items.

Strengthen the loyalty program to convert frequent guest shoppers into long-term customers.

Target underperforming locations (Mall branch) with marketing campaigns or operational changes.

Limitations and Next Steps

Limitations

The dataset was originally provided as a single flat file, which was sufficient for this project but not reflective of how data is typically stored in enterprise environments.

Customer and employee details were limited (only IDs were available), restricting deeper demographic or workforce analysis.

External factors such as supplier costs, marketing spend, and seasonal data were not included, which limits the ability to measure profitability or campaign effectiveness.

Next Steps

Database Integration: Restructure the dataset into a relational star schema and load it into a SQL Server database. This would enable more efficient queries, support larger datasets, and prepare the model for enterprise-scale reporting.

Data Enrichment: Incorporate additional datasets such as customer demographics, supplier pricing, and marketing campaigns to broaden analysis.

Automation: Implement scheduled refreshes in Power BI to keep the dashboards updated in near real-time.

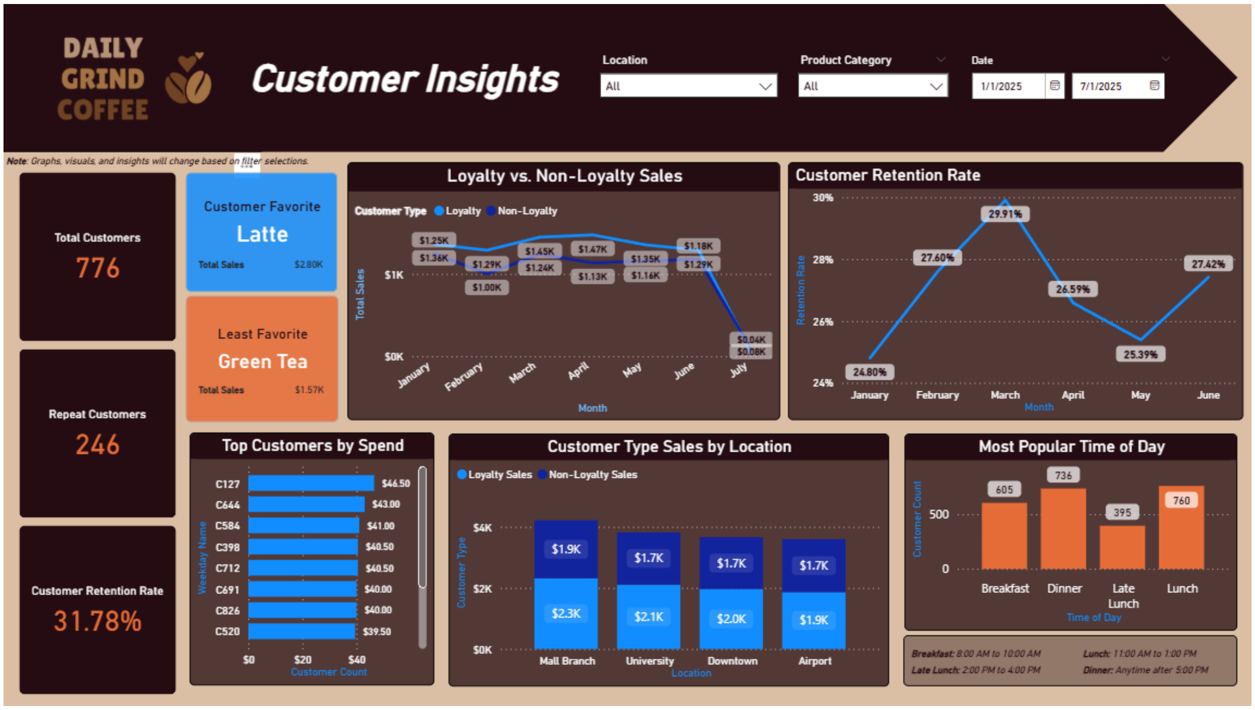
Predictive Analytics: Extend the project by adding forecasting models (e.g., time series analysis for sales trends) to provide forward-looking insights.

Final Dashboard Screenshots

Sales Overview

Product Performance

Customer Insights



Reflection

Completing this project was both a technical and professional growth experience. At the outset, I expected to simply create a Power BI dashboard from a flat CSV file. However, as I worked through the data, I recognized the opportunity to take the project further by applying **best practices in data modeling, schema design, and measure creation**. Converting the dataset into a star schema was a turning point, because it not only improved performance and flexibility but also mirrored the processes used in enterprise BI environments.

The most valuable lesson I gained was the importance of **designing for scalability and maintainability**. Instead of focusing only on visualization, I learned how structuring the underlying data model properly simplifies calculations, improves performance, and enables richer insights. Creating DAX measures for customer behavior, sales growth, and retention pushed me to think beyond static metrics and develop KPIs that could adapt dynamically to filters and business questions.

This project also reinforced the need to think like both a data analyst and a business stakeholder. Building dashboards was not just about displaying numbers — it was about crafting a **narrative flow** that tells the story of sales, products, and customers in a way that different audiences (executives, operations, marketing) could act on. I found myself constantly asking: *"If I were a manager, what decision would this chart help me make?"*

Finally, the experience highlighted areas for future growth. While the dashboards are effective for the current dataset, integrating additional data sources (e.g., marketing campaigns, supplier costs) and scaling to a SQL Server backend would elevate the solution to an enterprise level. Exploring predictive analytics such as forecasting sales or modeling churn would also add forward-looking value.

Overall, this project not only strengthened my technical skills in Power BI, DAX, and Power Query, but also helped me develop the mindset of a **decision science analyst** — one who can bridge raw data and actionable business insights.