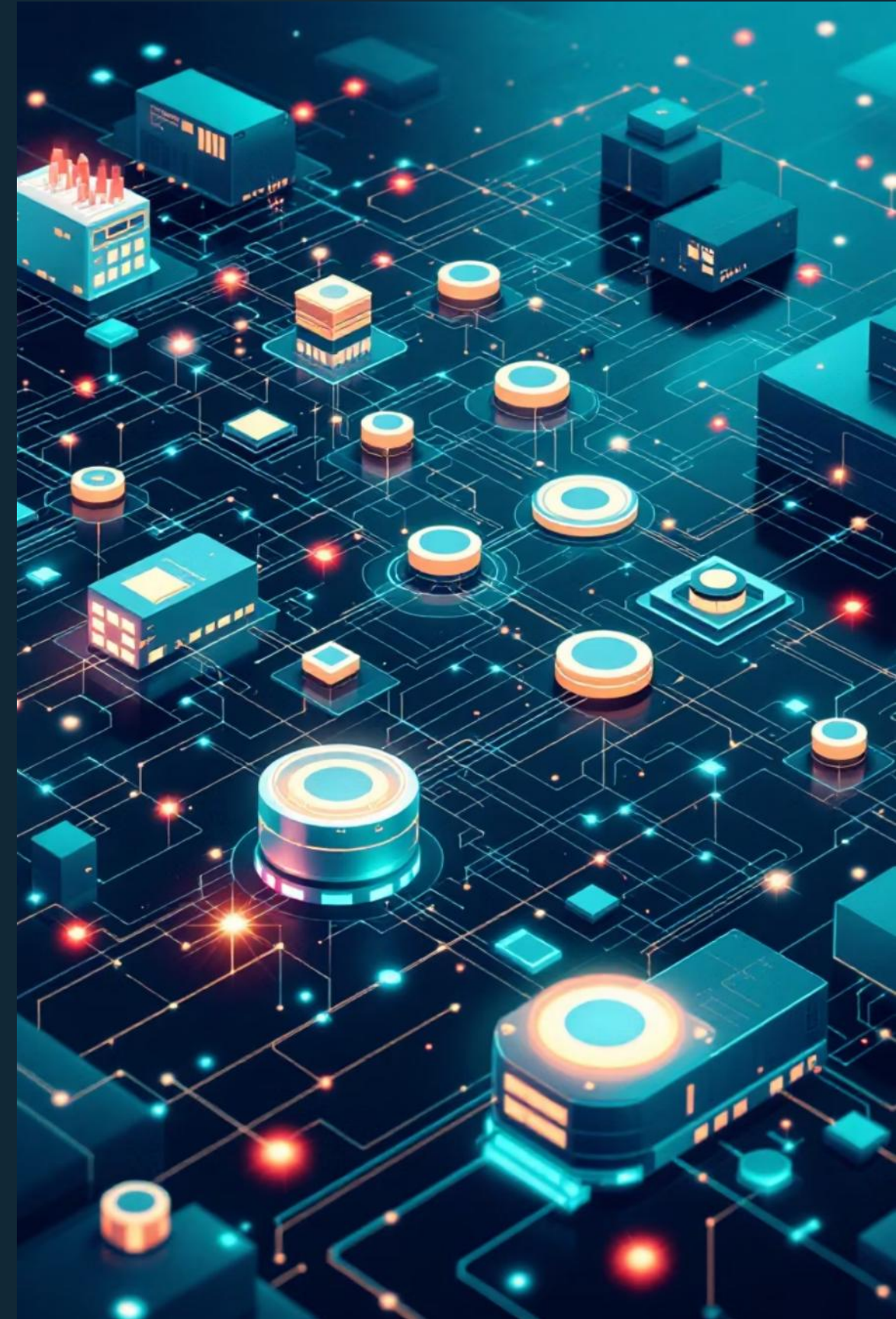


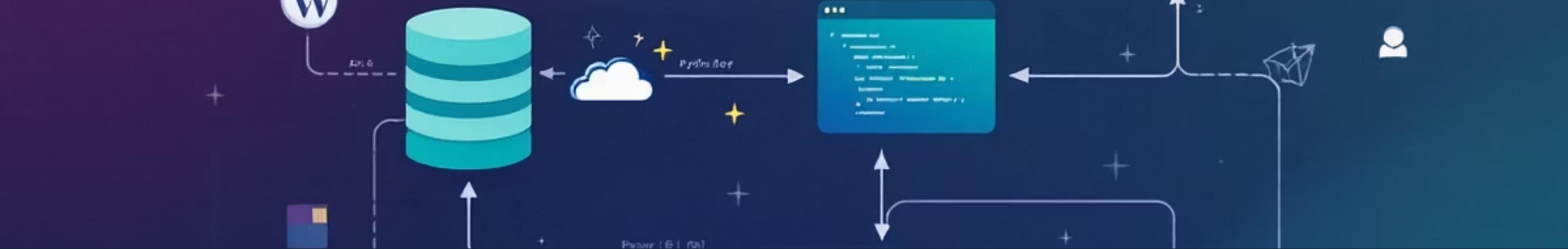
End-to-End Supply Chain Analytics Project

From Raw Data to Insightful Dashboards

Presented by: Nagwan Khaled Sagea

Team members: Nagwan Khaled Sagea, Osama Essam Mohamed, Mina Salama Fayez, Mohamed Osama Ali, Mohamed Mamdouh Mohamed, Ghada Ibrahim





Project Pipeline Overview

This project encompasses a full-spectrum data analytics journey, transforming raw supply chain data into actionable insights.



1. SQL Server

Data loading, initial querying, and structure exploration.



2. Python

Extensive data cleaning, EDA, and statistical analysis.



3. Export Cleaned CSV

A pristine dataset ready for visualization.



4. Power BI Dashboard

Interactive visualizations, KPIs, and strategic insights.

SQL Section Overview

SQL Server serves as the foundational layer, ensuring data integrity and providing initial insights into the raw supply chain data.

Loading Raw Data

Efficiently ingested the raw supply chain dataset into SQL Server, establishing the primary data source.

Data Quality Checks

Performed initial checks on data quality, identifying inconsistencies and potential issues early in the process.

Column Relationships

Explored and understood the intrinsic relationships between various columns, crucial for subsequent analysis.

Initial Insights

Executed queries to extract preliminary insights, guiding further data exploration and cleaning steps.

```
Query3.sql...ot connected X
43 GO
44
45
46 -- [x] Display the primary key information of the table
47 EXEC sp_pkeys @table_name = 'supply_chain_data';
48
49 -- [x] Check the number of NULL values in each column (for data cleaning purposes)
50 SELECT
51     SUM(CASE WHEN Product_type IS NULL THEN 1 ELSE 0 END) AS Null_Product_type,
52     SUM(CASE WHEN SKU IS NULL THEN 1 ELSE 0 END) AS Null_SKU,
53     SUM(CASE WHEN Price IS NULL THEN 1 ELSE 0 END) AS Null_Price,
54     SUM(CASE WHEN Availability IS NULL THEN 1 ELSE 0 END) AS Null_Availability,
55     SUM(CASE WHEN Number_of_products_sold IS NULL THEN 1 ELSE 0 END) AS Null_Num_Sold,
56     SUM(CASE WHEN Revenue_generated IS NULL THEN 1 ELSE 0 END) AS Null_Revenue,
57     SUM(CASE WHEN Customer_demographics IS NULL THEN 1 ELSE 0 END) AS Null_Demographics,
58     SUM(CASE WHEN Stock_levels IS NULL THEN 1 ELSE 0 END) AS Null_Stock,
59     SUM(CASE WHEN Production_Lead_Time IS NULL THEN 1 ELSE 0 END) AS Null_ProductionLeac
60     SUM(CASE WHEN Order_quantities IS NULL THEN 1 ELSE 0 END) AS Null_OrderQuantities,
61     SUM(CASE WHEN Shipping_times IS NULL THEN 1 ELSE 0 END) AS Null_ShippingTimes,
62     SUM(CASE WHEN Shipping_carriers IS NULL THEN 1 ELSE 0 END) AS Null_Carriers,
63     SUM(CASE WHEN Shipping_costs IS NULL THEN 1 ELSE 0 END) AS Null_ShippingCosts,
64     SUM(CASE WHEN Supplier_name IS NULL THEN 1 ELSE 0 END) AS Null_SupplierName,
65     SUM(CASE WHEN Location IS NULL THEN 1 ELSE 0 END) AS Null_Location,
66     SUM(CASE WHEN Supplier Lead Time IS NULL THEN 1 ELSE 0 END) AS Null_SupplierLeadTime
% No issues found Ln: 1 Ch: 1 MIXED CRLF
```

The core of our analysis begins with the `SupplyChain` table, encompassing a wide array of operational metrics.

The core of our analysis begins with the `SupplyChain` table, encompassing a wide array of operational metrics.

- Product_type
- SKU
- Price
- Revenue_generated
- Customer_demographics
- Stock_levels
- Lead_time
- Manufacturing_lead_time
- Order_quantities
- Shipping_times
- Transportation_modes
- Shipping_costs
- Supplier_name
- Location

[illegible]

SQL Queries Explained

Strategic SQL queries were crafted to quickly assess data and extract initial performance indicators.

Preview Data

```
SELECT TOP 10 * FROM SupplyChain;
```

Purpose: Quick data inspection and schema verification.



Total Orders

```
SELECT Location, COUNT(*) FROM SupplyChain GROUP BY Location;
```

Purpose: Identify high-volume order locations.



Avg Shipping Time

```
SELECT Transportation_modes, AVG(Shipping_times) FROM SupplyChain GROUP BY Transportation_modes;
```

Purpose: Evaluate efficiency of transport modes.



Supplier Cost

```
SELECT Supplier_name, SUM(Manufacturing_costs) AS TotalManCost, SUM(Shipping_costs) AS TotalShipCost FROM SupplyChain GROUP BY Supplier_name;
```

Purpose: Analyze supplier cost-efficiency.

```
283 -- ☒ Identify potentially risky suppliers with both high manufacturing cost and defect rate
284 v SELECT
285     Supplier_name,
286     ROUND(AVG(CAST(Manufacturing_costs AS FLOAT)), 2) AS Avg_Manufacturing_Cost,
287     ROUND(AVG(CAST(Defect_rates AS FLOAT)), 3) AS Avg_Defect_Rate
288 FROM dbo.supply_chain_data
289 GROUP BY Supplier_name
290 HAVING AVG(CAST(Manufacturing_costs AS FLOAT)) > 2 AND AVG(CAST(Defect_rates AS FLOAT)) > 0.02
291 ORDER BY Avg_Manufacturing_Cost DESC, Avg_Defect_Rate DESC;
292
293 v -- ☒ Identify the most cost-efficient supplier per product type
294 -- This helps understand which supplier offers the lowest manufacturing cost for each product type.
295 -- ☒ Find the most cost-efficient supplier per product type
296 v WITH AvgCostPerSupplier AS (
297     SELECT
298         Product_type,
299         Supplier_name,
300         ROUND(AVG(CAST(Manufacturing_costs AS FLOAT)), 2) AS Avg_Manufacturing_Cost
```

Results			
Messages			
	Supplier_name	Avg_Manufacturing_Cost	Avg_Defect_Rate
1	Supplier 4	62.71	2.337
2	Supplier 1	45.25	1.804
3	Supplier 5	44.77	2.665
4	Supplier 3	43.63	2.466
5	Supplier 2	41.62	2.363



SQL Business Questions Answered

The SQL analysis directly addressed critical business questions, laying the groundwork for the Power BI dashboard's KPIs.

1

Average Lead Time?

Calculated to understand production and delivery cycles.

2

Highest Revenue Supplier?

Identified top-performing suppliers based on revenue contribution.

3

Cost Distribution by Mode?

Analyzed shipping costs across different transportation methods.

4

Avg. Defect Rate?

Determined product quality metrics per type.

Python Section Overview

Python was instrumental for advanced data manipulation, ensuring the dataset's quality and readiness for in-depth analysis.



Advanced Cleaning

Utilized Pandas for sophisticated data cleansing routines.



Handle Missing Values

Strategically addressed and imputed or removed null values.



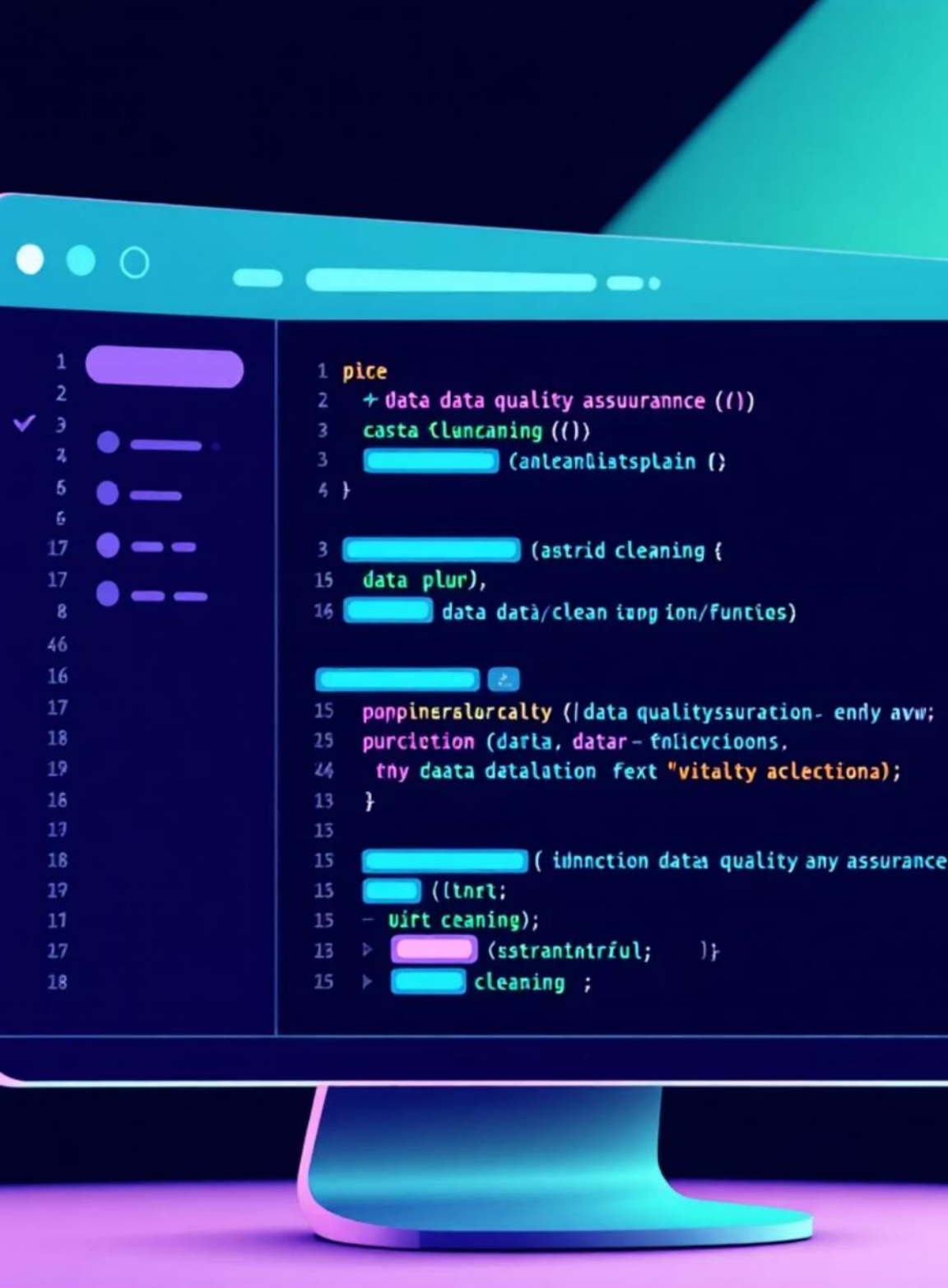
Outlier Detection

Implemented robust methods to identify and manage data outliers.



Statistical Profiling

Generated comprehensive summary statistics and visual EDA.





Python Cleaning Steps

A systematic approach to data cleaning in Python ensured a high-quality dataset, enhancing the reliability of subsequent analysis.

01

Remove Duplicates

```
df = df.drop_duplicates()
```

Eliminated redundant entries to maintain data integrity.

03

Outlier Removal (IQR)

```
IQR = Q3 - Q1; lower = Q1 - 1.5 * IQR; upper = Q3 + 1.5 * IQR  
df = df[(df[col] >= lower) & (df[col] <= upper)]
```

Used Interquartile Range to prevent skewed analyses from extreme values.

02

Remove Missing Values

```
df = df.dropna()
```

Addressed gaps in the dataset for complete analysis.

04

Standardize Text Fields

Cleaned text data by removing extra spaces and normalizing casing (e.g., title case) for consistency.

📌 This ensures accuracy, prevents category duplication, and enhances dashboard clarity.



Python EDA (Before Cleaning)

Exploratory Data Analysis before cleaning revealed crucial patterns and issues within the raw data, guiding our remediation efforts.

Visuals Generated:

- Boxplots (for outlier detection)
- Missing values summary
- Duplicate count reports
- Histograms (for distribution analysis)
- Correlation matrices (for relationships)

Key Findings:

- Numerous outliers in **Price**, **Revenue**, and **Shipping Time**.
- Presence of duplicate rows affecting data integrity.
- Significant missing values across several key columns.
- Strong positive correlation identified between **Revenue** and **Price**.
- **Manufacturing Lead Time** emerged as a bottleneck for specific SKUs.

Python EDA (After Cleaning)

The post-cleaning EDA confirmed the successful transformation of the raw data into a reliable and robust dataset, primed for Power BI.



Cleaned Dataset

Ready for advanced analytics.



Outliers Removed

Ensuring accurate insights.



Missing Values Resolved

Complete and comprehensive data.



Text Values Standardized

Improved consistency and clarity.

Final Output: `cleaned_data.csv`

This polished dataset is now the cornerstone for generating dynamic visualizations and strategic insights in Power BI.

```
17]: """
# 1) IMPORTS
# =====
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
import seaborn as sns
from sqlalchemy import create_engine
from IPython.display import display

# =====
# 2) LOAD FULL DATA FROM SQL
# =====

engine = create_engine(
    "mssql+pyodbc://LAPTOP-ENGNAWA/SupplyChain"
    "?driver=ODBC+Driver+18+for+SQL+Server"
    "&trusted_connection=yes"
    "&Encrypt=no"
)

# ✅ TOP 10 هنا جينا كل الجدول من غير
df_original = pd.read_sql("SELECT * FROM SupplyChain")

print(f"📊 Original dataset shape:", df_original.shape)
print(f"\n📋 Preview of dataset (first 5 rows):")
print(df_original.head())

# =====
# 3) BASIC INFO BEFORE CLEANING
# =====

print("\n♦ Data Overview BEFORE cleaning:")
print(df_original.info())

print("\n♦ Missing Values per Column (BEFORE):")
print(df_original.isnull().sum())
```


Power BI Dashboard Overview

The Power BI dashboard consolidates our analysis into an intuitive and interactive experience, providing clear strategic insights across three key analytical pages.

Analytical Pages:

- Overview
- Product Insights
- Supplier Insights

Key Dashboard Features:

- Dynamic KPIs for quick performance assessment
- Advanced DAX measures for complex calculations
- Interactive slicers for granular data exploration
- Customized UI design for enhanced user experience
- Business story-driven visuals for actionable insights





KPIs & DAX Measures Used

Key Performance Indicators and their DAX measures were meticulously crafted in Power BI to provide actionable insights into the supply chain's performance.

1

Total Revenue

`Total Revenue = SUM(cleaned_data[Revenue_generated])`

Measures overall sales performance.

2

Total Orders

`Total Orders = SUM(cleaned_data[Order_quantities])`

Quantifies the total volume of customer orders.

3

Average Delivery Time

`Average Delivery Time = AVERAGE(cleaned_data[Shipping_times])`

Assesses the efficiency of logistics and delivery processes.

4

Average Profit Margin

`Profit Margin = DIVIDE(SUM(cleaned_data[Revenue_generated]) - SUM(cleaned_data[Shipping_costs]), SUM(cleaned_data[Revenue_generated]))`

Evaluates profitability after accounting for shipping costs, converted to percentage for clarity.

5

Orders by Demographics

`Orders by Demo = SUM(cleaned_data[Order_quantities])`

Analyzes order distribution across different customer segments.

Overview Page Insights

The Power BI Overview Page presents a high-level summary of the supply chain's performance, highlighting critical metrics and strategic insights derived from our cleaned and analyzed data.

577.6K

Total Revenue

Overall financial performance, demonstrating top-line growth.

5.75

Avg. Delivery Time (Days)

Efficiency of logistics, showing swift fulfillment processes.

5000

Total Orders

Volume of transactions, indicating market demand and operational scale.

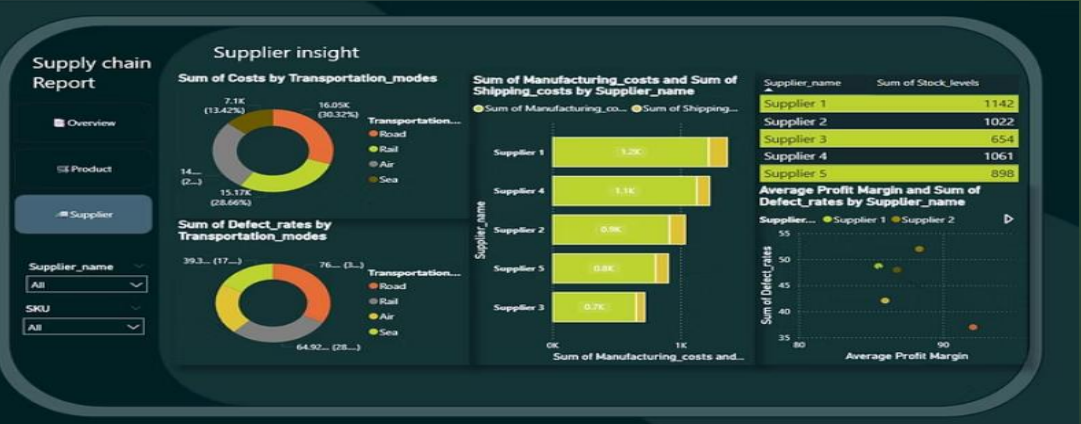
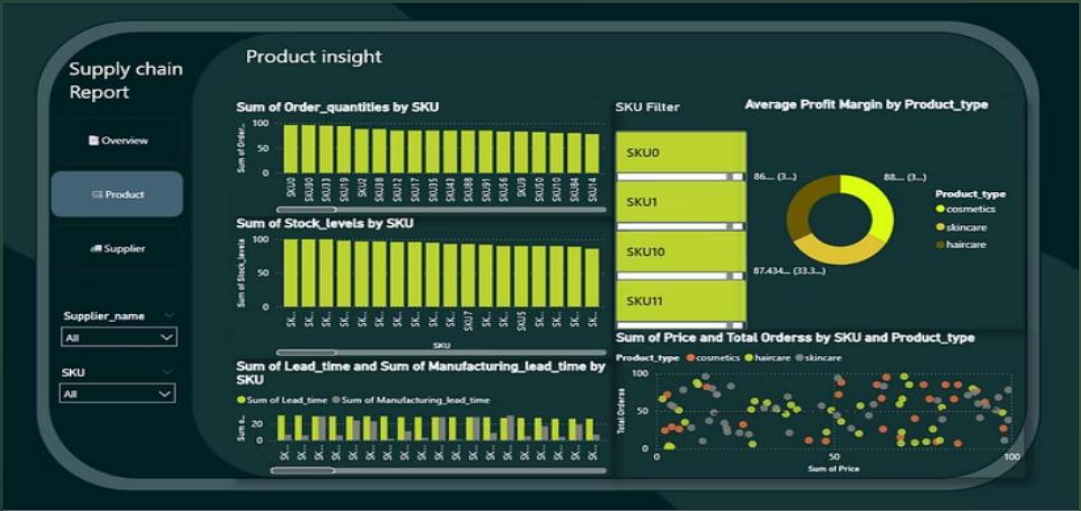
87.44%

Avg. Profit Margin

Core profitability, reflecting effective cost management.

Key Strategic Insights

- Female customers drive the highest order volume, indicating a key demographic focus.
- Mumbai & Chennai stand out as the top-performing locations for sales and distribution.
- Air transportation mode is demonstrably the fastest, crucial for urgent deliveries.
- Road transport is most utilized but shows longer average shipping times.
- Haircare products exhibit comparatively higher defect rates, requiring further quality control investigation.



Product Page Insights

The Product Insights page in Power BI provides granular details on individual product performance, stock management, and demand patterns, enabling data-driven decisions for product optimization.

Key Visualizations:

- Orders by SKU
- Stock levels by SKU
- Lead time vs manufacturing lead time
- Scatter plot: Price vs Orders
- Profit margin by product type

Strategic Insights:

- Overstock Risk:** Some SKUs show high stock levels despite low demand, indicating potential overstocking issues.
- Profit Drivers:** Cosmetics products consistently generate the highest profit margins across the portfolio.
- Demand Sweet Spot:** Mid-priced products demonstrate the highest overall demand, suggesting optimal market positioning.
- Bottleneck Alert:** Specific SKUs are experiencing manufacturing bottlenecks, impacting lead times and product availability.

Supplier Page Insights

The Power BI Supplier Page offers a deep dive into vendor performance, evaluating key metrics to optimize procurement and foster strong supply chain partnerships.

Key Visualizations:

- Manufacturing & shipping costs per supplier
- Defect rates by transportation mode
- Supplier stock comparison
- Profit margin vs defect rate

Strategic Insights:

Supplier 3: Top Performer

- **High Margin:** Consistently delivers products with excellent profit margins.
- **Low Defects:** Maintains superior quality control, resulting in minimal defect rates.
- **Cost Efficient:** Offers competitive pricing and efficient manufacturing processes.

Supplier 5: Underperforming

- **Lower Margins:** Products from this supplier yield reduced profitability.
- **Higher Defect Rates:** Experiences more frequent quality issues, impacting customer satisfaction.
- **Needs Evaluation:** Requires immediate review for potential corrective actions or replacement.

Final Recommendations

Based on our comprehensive analysis, these strategic recommendations are designed to optimize supply chain performance, reduce costs, and enhance overall profitability.



Allocate More Orders to Supplier 3

Leverage **Supplier 3's high margins and low defect rates** to maximize profitability and product quality.



Reduce Sea Transportation

Minimize the use of **Sea transportation due to its high defect rates**, impacting product quality and delivery timelines.



Optimize Inventory for Low-Demand SKUs

Implement stricter inventory controls for **SKUs identified with potential overstocking issues** to free up capital and storage space.



Improve Quality Control for Haircare Products

Address the **comparatively higher defect rates in haircare products** through targeted quality assurance initiatives.



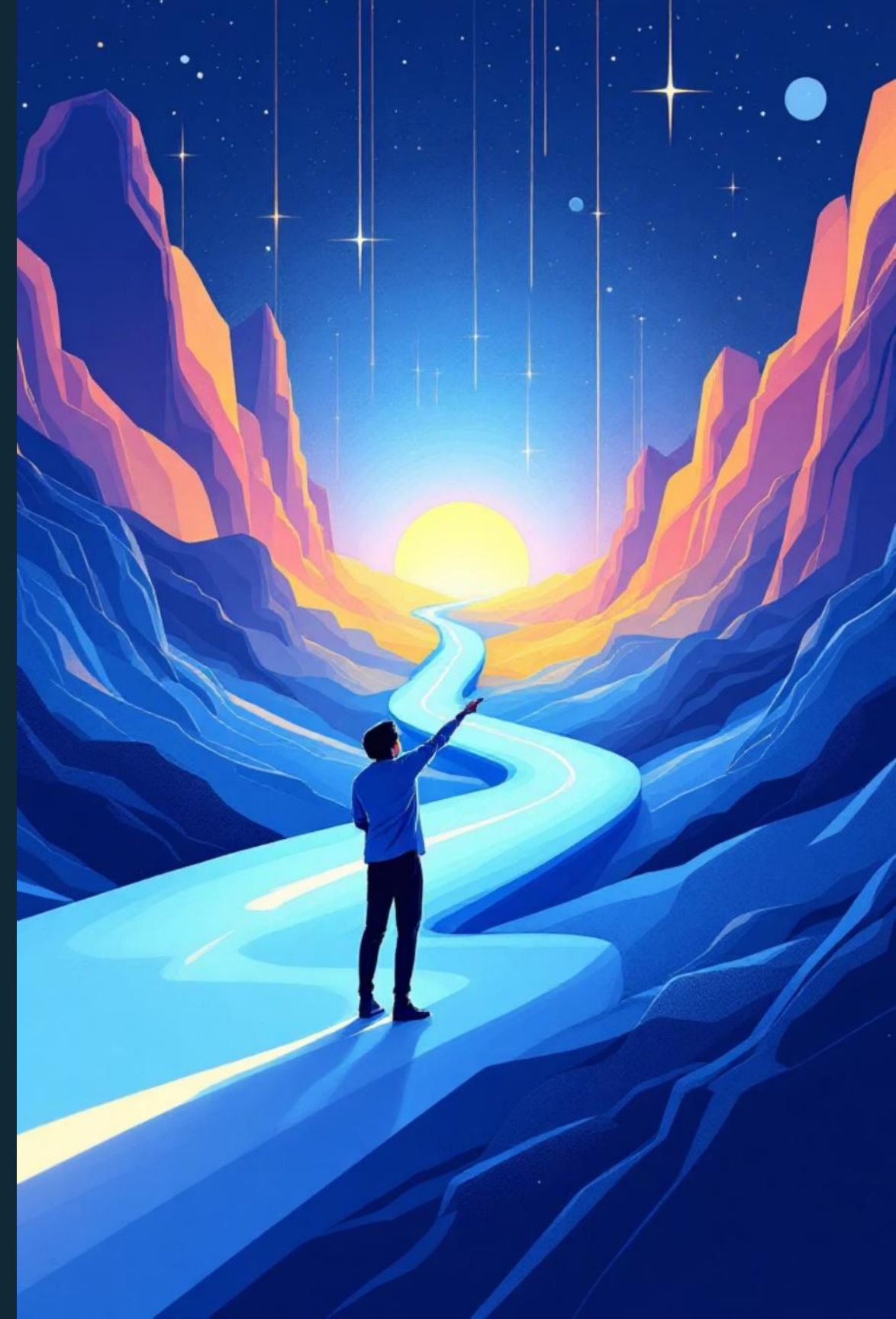
Increase Targeted Marketing for Female Customers

Capitalize on the fact that **female customers drive the highest order volume** with tailored marketing campaigns.



Use Air Transport for Urgent Shipments

Utilize **Air transportation for critical or urgent deliveries**, benefiting from its demonstrably faster shipping times.



Thank You!

Further analytics enhancements will continue to improve supply chain efficiency and drive sustained business growth.