SmartMart Retail Sales Analysis

Introduction

The SmartMart project focuses on analyzing retail sales data to extract meaningful business insights. It integrates data wrangling, SQL analysis, Python analytics, and Power BI visualization to provide a complete business intelligence solution. The project aims to track revenue and profits, understand customer behavior, identify best-selling products and categories, optimize stock management, and support better business decision-making.

Revenue Tracker (Power BI Dashboard)

Overview: A complete sales overview with key metrics and interactive reports.

Key Features:
- Total Revenue, Profit, Customers, Transactions
- Sales trends over time
- Top products and categories
- Customer demographics (age, gender, loyalty, city)
- Store and city-wise performance

Purpose: Helps stakeholders monitor sales, identify top products, understand customers, and compare store performance.

Data Wrangling

Overview: Cleaning and preparing raw datasets before SQL and visualization.

Key Features:
- Removed duplicates and inconsistencies
- Handled missing values (gender, city)
- Standardized column names and formats
- Ensured consistency of IDs across datasets
- Exported clean datasets for SQL and Power BI

Purpose: Ensures accurate, reliable, and consistent data for analysis.

SQL Analysis (PostgreSQL)

Overview: Built and managed SmartMart database for structured analysis.

Key Features:
- Created schema with primary & foreign keys
- Imported cleaned datasets
- Removed duplicates, handled missing values

- Analyzed sales by product, customer, and store
- Identified low-stock products
- Found product pairs frequently bought together

Purpose: Provides structured data and SQL-based insights for sales, customer behavior, and inventory management.

Python Analysis (Jupyter Notebook)

Overview: Applied data analysis and data mining techniques in Python.

Key Features:
- Created combined worksheets using joins
- Imported cleaned datasets
- Removed duplicates, handled missing values
- Converted into proper data types
- Applied Apriori Algorithm & Association Rules
- Identified frequently bought product pairs
- Analyzed products and categories for optimization

Purpose: Enables advanced analytics, customer insights, and product optimization using Python.

Tools & Technologies Used

Power BI – Dashboard & visualization
Python (Jupyter) – Data analysis & Apriori algorithm
PostgreSQL (SQL) – Database & queries
Excel/CSV – Initial data storage
Pandas, NumPy, Mlxtend – Data wrangling & association rules

Results & Insights

- Dashboard showed top products and cities driving sales
- Customer demographics provided insight into target audience
- SQL analysis identified low-stock products
- Python analysis revealed frequently bought product pairs useful for cross-selling strategies

Conclusion

The SmartMart project successfully combines data cleaning, SQL database management, Python analytics, and Power BI visualization into a single integrated system. It helps track revenue, monitor inventory, understand customer behavior, and support decision-making for business growth.