Retail Sales Data Analysis Project – ETL & BI project

Overview

This project uses the **Kaggle API** to download the dataset **"Retail Sales Order Data"**. The data is then processed and cleaned using **Python** and **Pandas**, followed by loading the cleaned data into a **SQL Server** for further data analysis. The analysis helps solve business questions related to sales performance, product trends, and profitability.

Technologies & Tools Used

- **Data Extraction:** Kaggle API (for dataset import)
- **Data Processing & Cleaning:** Python (Pandas)
- **Data Transformation & Loading:** Python (SQLAlchemy for loading data into SQL Server)
- **Data Analysis & Querying:** SQL (SQL Server Management Studio)

---

1. Data Extraction & Loading

Technologies: Kaggle API, Python (zipfile, pandas)

- **Data Retrieval:** Used the **Kaggle API** to download the **"Retail Sales Order Data"**.
- **File Extraction:** Utilized Python's zipfile module to extract the dataset files.
- **Data Loading:** Loaded the CSV file into a **Pandas DataFrame** for further data manipulation.

---

2. Data Cleaning & Preprocessing

Technologies: Python (Pandas)

- **Column Standardization:** Standardized column names by converting them to lowercase and replacing spaces with underscores for consistency.
- **Missing Data Handling:** Used **Pandas** to handle missing values effectively using methods like fillna() and dropna().
- **Data Type Conversion:** Converted the order_date column from string to datetime format using pd.to_datetime() for accurate time-based analysis.
- **Feature Engineering:** Created new columns like discount, sale_price, and profit to provide more insights into sales performance.
- **Data Cleaning:** Removed irrelevant columns such as list_price, cost_price, and discount_percent to focus on essential information.

---

3. Data Transformation & Loading into SQL Server

Technologies: Python (SQLAlchemy, Pandas), SQL Server

- **SQL Server Connection:** Established a connection between Python and SQL Server using the **SQLAlchemy** library.
- **Data Loading:** Loaded the cleaned DataFrame into a **SQL Server** table named df_orders using the to_sql() method in **Pandas**. This allowed for structured storage and further SQL-based analysis.

---

4. Data Analysis (SQL)

Technologies: SQL (SSMS)

- **Top Revenue-Generating Products:** Wrote SQL queries to identify the top 10 products based on total sales revenue using the SUM() function and GROUP BY clause.
- **Top Selling Products by Region:** Used a Common Table Expression (CTE) and the ROW_NUMBER() window function to rank products by sales within each region and select the top 5 products.
- **Sales Growth Comparison:** Compared sales growth year-over-year and month-over-month for different regions and product categories using SQL aggregate functions and CTEs.
- **Profit Growth Analysis:** Analyzed the changes in profitability between 2022 and 2023 by calculating the difference in sales and profits for each product category.

---

5. Insights and Reporting

Technologies: Python (Matplotlib, Seaborn), SQL (SSMS)

SQL Analysis: Performed detailed business analysis by querying the data in **SQL Server Management Studio (SSMS)**, answering business questions like:

  - What are the top-selling products by region?
  - Which months saw the highest sales growth?
  - What is the profit growth across product categories?

Conclusion

This project demonstrates a complete **end-to-end data analytics pipeline**:

1. **Data extraction** via Kaggle API
2. **Data cleaning** and **processing** using **Python** and **Pandas**
3. **Data analysis** and **insight generation** through **SQL** in **SQL Server Management Studio** (SSMS)