Retail Order Data Analysis - Project Documentation

# Objective

To analyze and optimize sales performance by identifying trends, top-performing products, and growth opportunities using sales transaction data.

# Process Overview

## 1. Data Source Acquisition

Platform: Kaggle  
Details: The dataset containing retail order data was sourced from Kaggle. It includes information on product details, sales transactions, and shipment information.

## 2. Data Transformation

Performed data cleaning and transformations to ensure the dataset is analysis-ready.  
Splitted the cleaned data into two tables:  
 - product: Includes product-specific details (e.g., category, price, discount, profit).  
 - shipment: Contains shipment-related details (e.g., city, region, order date).

## 3. Data Storage

Database: PostgreSQL  
Setup: Established a connection to PostgreSQL.  
Action: Uploaded the product and shipment tables into the database for efficient querying.

## 4. Query Writing

Requirements Addressed:  
 - 10 Given Requirements: Queries to calculate KPIs such as top-performing products, revenue by category, etc.  
 - 10 Additional Requirements: Custom queries created to explore deeper insights like regional performance, discount impact on sales, etc.

### Queries Example

SELECT category, SUM(profit) AS total\_profit  
FROM product  
GROUP BY category  
ORDER BY total\_profit DESC;

## 5. Data Visualization

Tool Used: Streamlit  
Purpose: To create interactive dashboards and visualizations for insights derived from the SQL queries.

### Key Visualizations

- Bar charts for product and category performance.  
- Line charts for year-over-year comparisons.  
- Scatter plots for the relationship between discounts and sales.

## 6. Publishing

Platform: GitHub  
Details: The entire project, including source files, queries, and Streamlit app code, was documented and published for transparency and evaluation.

# Project Features

End-to-End Workflow: Covers data acquisition, transformation, analysis, and visualization.  
Database Integration: Effective use of PostgreSQL for query execution.  
Interactive Visuals: Insights are easily accessible through interactive Streamlit dashboards.  
Custom Insights: Enhanced the project scope with self-defined analytical queries.

# Short Presentation Summary

## Introduction

Briefly introduce the project objective.

## Methodology

Explain each step:  
- Sourced data from Kaggle.  
- Transformed and uploaded the data into PostgreSQL.  
- Wrote SQL queries for analysis.  
- Visualized results using Streamlit.

## Highlights

Showcase interactive dashboards and how the analysis supports business decisions.

## Conclusion

Discuss findings and potential future enhancements.