Project Objective

The goal of this project is to use **SQL** (Structured Query Language) to analyze and optimize the inventory management system for **TechElectro Inc.**, a company dealing with product sales. The project focuses on making decisions based on data related to product sales, economic factors, and stock levels. It involves cleaning data, combining it into useful views, and creating methods to monitor inventory and sales trends.

Key Components of the Project

1. **Data Exploration**:
   - Before making decisions, we must first explore the data.
   - The first step was loading and inspecting the data from the company's different sources, such as:
     - **Product sales information**
     - **Product category details**
     - **External economic factors like GDP and inflation rates**
   - We check the structure of the data to understand what kind of information is stored in each column and ensure it's accurate.
   - *Example*: Using SQL, we examine the first few rows of each table (like `SELECT * FROM external_factors LIMIT 5`) to confirm that the data looks correct.
2. **Data Cleaning**:
   - This step ensures that all data types (such as dates, numbers) are accurate and consistent across the tables.
   - Missing and duplicate data is checked and fixed. For example, we make sure every sale has a corresponding date, and every product has valid pricing.
   - Duplicates (repeated data) are removed to prevent errors in analysis.
   - *Example*: Changing sales dates from text to actual date formats and deleting duplicate records.
3. **Data Integration**:
   - After cleaning, the data from various sources is combined. This allows us to analyze everything together, such as sales trends with external economic factors.
   - Views are created to combine tables (like `sales_data`, `product_data`, and `external_factors`) into a single, more meaningful dataset. This makes it easier to track which products are selling under certain conditions.
   - *Example*: Using SQL `JOIN` operations to merge product information with sales data.
4. **Descriptive Analysis**:

- This involves calculating basic statistics like average sales, median stock levels, and product performance.
- By doing so, we can identify:
  - Which products sell the most
  - How often products run out of stock
  - Patterns in stock levels over time.
- *Example*: We calculate the total sales for each product to determine which products are high-demand and need more stock.

5. **External Factors Analysis**:
- This section investigates how external economic factors like GDP and inflation affect product sales.
- *Example*: If the GDP is high, sales may increase because people have more money to spend. We analyze how sales differ during periods of economic growth and decline.

6. **Inventory Optimization**:
- A key part of this project is optimizing the company's stock levels, ensuring that the right amount of product is available to meet customer demand while minimizing excess stock.
- Using calculations like **Lead Time Demand** and **Safety Stock**, we set "reorder points," which indicate when the company should order more stock.
- *Example*: A system is set up to automatically calculate when stock levels drop below a certain point, triggering a reorder to avoid stockouts.

7. **Overstocking and Understocking**:
- This section identifies products that are either overstocked (too much inventory) or understocked (frequent stockouts).
- This helps in reducing costs related to excess inventory and preventing lost sales due to stockouts.
- *Example*: SQL queries were run to track how often products go out of stock and the potential lost sales during those times.

8. **Monitor and Adjust**:
- We create SQL procedures to routinely monitor stock levels, sales trends, and product stockouts, ensuring the company can react quickly to changes in demand.
- These automated systems help ensure continuous adjustment to the inventory system, leading to better customer satisfaction and fewer lost sales.
- *Example*: SQL procedures are created to track stockout frequencies and report the highest stockout days for different products.

9. **Feedback Loop**:
- A feedback system is developed to gather insights from stakeholders, helping refine and improve the inventory system based on real-world needs.
- This ensures continuous improvement and alignment with business goals.

- *Example*: Setting up periodic review meetings and a platform for stakeholders to provide feedback.

Conclusion

This SQL-based project provides a detailed analysis and optimization framework for **TechElectro Inc.**. By cleaning the data, integrating it, and running various analyses, the company can make smarter inventory decisions. The process includes tracking sales trends, analyzing external economic factors, and ensuring products are restocked optimally, all while minimizing excess inventory and avoiding stockouts.