**Project Title: Inventory Management System**

**Project Overview**We propose to design a lightweight yet functional solution for tracking and managing stock levels, purchases, sales, and generating reports.

Given Colab’s cloud-based environment and Python’s flexibility, the system employs pandas for data manipulation and supports basic file-based storage using CSV files or Google Sheets for data persistence. Optional features such as dashboards or automated alerts may be integrated to enhance usability.

## **1. Project Objectives**

* Develop a user-friendly Inventory Management System to efficiently track stock levels, orders, sales, and deliveries.
* Enable real-time updating and management of inventory data to maintain accuracy and consistency.  
  Provide comprehensive analytical insights, including stock shortages, reorder alerts, and sales trends.
* Ensure robust data persistence with seamless data import and export capabilities.
* Implement a modular and scalable codebase to facilitate future enhancements and system integration.
* Incorporate a secure login feature to control user access and maintain data confidentiality.
* Offer optional email notifications to designated users, enabling proactive inventory management and timely responses to critical events.

## **2. Features**

* **Inventory Item Management:** Add, update, and delete inventory items, including detailed product information such as name, SKU, quantity, price, and supplier data.
* **Stock Tracking:** Continuously monitor stock levels with automated alerts to notify users of low inventory conditions.
* **Order Management:** Record and manage incoming orders while automatically updating inventory quantities.
* **Sales Tracking:** Track sales transactions, adjust stock quantities accordingly, and generate detailed sales reports.
* **Search and Filter:** Facilitate quick and efficient location of items using various search criteria and filtering options.
* **Data Visualization:** Present inventory trends and summaries through clear and informative charts.
* **Data Import/Export:** Support loading data from CSV files and exporting reports for external use and backup.
* **User Input Validation:** Implement rigorous validation to minimize data entry errors and maintain data integrity.
* **User Authentication:** A secure login feature enables user authentication. Each user must log in using a unique set of credentials. Access rights can be role-based (e.g., administrator, staff, or viewer), ensuring data protection and controlled access.
* **Email Notifications (Optional):** Send automated email alerts to designated users to proactively manage inventory levels and respond promptly to reorder needs or other critical events.

## **3. Libraries**

Considering the development environment is Google Colab, the following Python libraries will be utilized:

* pandas: for efficient data manipulation and storage.
* numpy” to support numerical computations.
* matplotlib and seaborn” for generating graphical representations of data.
* ipywidgets: to create interactive UI components within the Colab notebook.
* sheets: for handling Google sheet file input and output operations.
* hashlib / getpass: to implement secure password handling for the login feature.
* smtplib and email: to enable email notification functionality.

## **4. Language**

* The project will be developed entirely in **Python**, within the Google Colab environment.

## **5. Project Overview**

The Inventory Management System will be implemented as a Python notebook within Google Colab, utilizing Python’s data libraries.Inventory data will be managed using Pandas DataFrames, facilitating straightforward manipulation, querying, and exporting of data. User interaction will be supported through dynamic forms and controls created with ipywidgets, allowing real-time updates and data visualization without requiring users to exit the notebook environment.

The project will commence with defining the inventory data schema and developing fundamental CRUD (Create, Read, Update, Delete) operations. Subsequently, additional modules for stock tracking, sales monitoring, and order management will be incorporated. The system will also include visual analytics components to provide meaningful insights into inventory status and trends. Data import and export features will implemented using Google Sheets.

A secure login feature will be integrated at the outset to control access and protect sensitive inventory information. Furthermore, the system will offer optional email notification capabilities, allowing designated users to receive timely alerts for critical inventory events, thereby supporting proactive and efficient inventory management.

The modular architecture of the system will enable straightforward expansion, such as integration with external databases, web interfaces, or automated restocking functionalities in future iterations.