

# INVENTORY MANAGEMENT SYSTEM

## Project Description:

This Python-based system is designed to provide seamless inventory management for a large manufacturing company. By utilizing object-oriented programming (OOP) principles, the system manages the core operations of inventory tracking, reordering, material transfers, and forecasting. It ensures that the inventory system can be accessed and updated concurrently from various production units while safeguarding sensitive data via an authentication mechanism.

The system is divided into multiple modules:

- **Inventory Management:** Track the availability and stock levels of materials.
- **Reordering System:** Automatically triggers a reorder when stock levels fall below a predefined threshold.
- **Material Transfer System:** Handles material transfers between warehouses and raises alerts for delays.
- **Forecasting System:** Generates reports based on historical data to forecast future demand and calculate inventory turnover.
- **Alert System:** Sends alerts for shipment delays or stock discrepancies via SMTP-based email notifications.

**Security:** Ensures authentication for secure access to inventory data.

## Required Modules

To run the project, the following Python modules need to be installed:

1. pandas - For data manipulation and analysis.
2. smtplib - For sending email notifications.
3. email.mime - For constructing email messages.
4. Need to Run **main.py** file.

## OPERATIONS

1. **Product Management:** Add Material, Update Stock and Display Stock.
2. **Check Records:** Checks stock levels and triggers an automatic reorder for materials with stock levels below a threshold. If no materials need reordering, the system outputs a message indicating no reorder is required.
3. **Transfer System:** Manages the transfer of materials between warehouses, including delays. Sends email alerts when delays exceed a specified duration using SMTP.
4. **Reporting:** Generate reports on inventory levels, sales, and orders.
5. **User Authentication:** Secure login and access control.

## Sample Input and Output

### Authentication

#### Input:

```
C:\Users\admin\Downloads\Inventory Management System>python main.py
Welcome to the Inventory Management System
Enter your username: admin
Enter your password: Python
User admin authenticated successfully.
Welcome admin! Your role is Admin.
```

#### Output:

```
User admin authenticated successfully.
Welcome to the Inventory Management System
```

### 1. Update Inventory

#### Input:

```
Enter your choice: 1
Enter Item ID to update: 101
Enter quantity to add: 30
```

#### Output:

```
Updated Laptop stock to 80
Updated Item 101 by 30 units.
```

TEAM MEMBERS:  
Vinay Regonda(vregonda)  
Bhavyashree.V(bhavenug)

## 2. Add New Material

### Input:

```
Enter your choice: 2
Enter new Item ID: 105
Enter new Item name: Smartphones
Enter initial stock quantity: 100
Enter cost of goods sold: 12000
Enter historical sales: 560
```

### Output:

```
Added new material: ID 105, Name Smartphones, Stock 100.
```

## 3. Check Reorders:

### Input and Output:

```
Enter your choice: 3
Checking for reorder requirements...
Reordering Item 103.
Item 103 has been reordered.
```

## 4. Manage Transfers

### Input:

```
Enter your choice: 4
Enter Item ID to transfer: 102
Enter source warehouse: warehouse A
Enter target warehouse: arehouse B
Enter quantity to transfer: 10
Enter transfer delay in hours: 6
```

### Output:

```
Delay in shipment for Item 103. Delay: 6 hours.
Alert email sent to bhavyashreev123@gmail.com.
```

## 5. Generate Records

### Input and Output:

```
Enter your choice: 5
Generating reports...
Calculating inventory turnover...

--- Inventory Turnover Report ---
  id      name  turnover_rate
0  101    Laptop         100.0
1  102    Silver         100.0
2  103    Steel          100.0
3  104  T-Shirts          60.0
4  105 Smartphones       120.0
Generating forecast report...

--- Forecast Report ---
  id      name  forecasted_sales
0  101    Laptop         4950.0
1  102    Silver        2750.0
2  103    Steel         1980.0
3  104  T-Shirts        2200.0
4  105 Smartphones         616.0
```

## 6. Display Stock Levels

### Input and Output:

```
Enter your choice: 6

--- Current Stock Levels ---
ID: 101, Name: Laptop, Stock: 50
ID: 102, Name: Silver, Stock: 30
ID: 103, Name: Steel, Stock: 20
ID: 104, Name: T-Shirts, Stock: 50
ID: 105, Name: Smartphones, Stock: 100
```

## 7.Exit

### Input and Output:

```
Enter your choice: 7
Exiting the system.
```

## Conclusion

This inventory management system project has been a comprehensive application of Python programming concepts. It helped me solidify my understanding of OOP, SMTP for sending email alerts, and working with the panda's library for data analysis.