

# Inventory Management System - C++ Code Explanation

## 1. Structure to Hold Product Information

The Product structure is used to define what a product should look like in terms of its id, name, quantity, and price. Each product in the inventory will have these four data points.

```
struct Product {  
    int id;  
    string name;  
    int quantity;  
    double price;  
};
```

## 2. Inventory Class

The Inventory class manages all products and has functions for adding, updating, deleting, restocking, and selling products, as well as displaying all products in the inventory.

### Key Parts of the Inventory Class

#### Private Variables:

```
vector<Product> products;    // Holds all products in the inventory  
int lowStockThreshold;      // Minimum quantity for low stock alert
```

### Functions in the Inventory Class

#### addProduct:

Adds a new product to the inventory and checks if the product ID is unique.

#### updateProduct:

Updates the quantity and price of an existing product and checks for low stock.

#### deleteProduct:

Deletes a product from inventory based on its ID.

**restockProduct:**

**Increases the quantity of a product by a specified amount.**

**sellProduct:**

**Reduces quantity upon sale. Shows an error if there is insufficient quantity.**

**checkLowStock:**

**Checks if a product's quantity is below the threshold and shows a low stock warning.**

**displayProducts:**

**Displays all products in the inventory with their details.**

### **3. Main Function and Menu System**

**The main function provides a user menu to perform various inventory management actions, like adding or deleting products. The program runs until the user chooses to exit.**

```
int main() {  
    Inventory inventory(5); // Low stock threshold set to 5  
    int choice;  
  
    while (true) {  
        cout << "--- Inventory Management System ---" << endl;  
        cout << "1. Add Product" << endl;  
        cout << "2. Update Product" << endl;  
        cout << "3. Delete Product" << endl;  
        cout << "4. Restock Product" << endl;  
        cout << "5. Sell Product" << endl;  
        cout << "6. Display All Products" << endl;  
        cout << "7. Exit" << endl;  
        cout << "Enter your choice: ";  
        cin >> choice;  
  
        if (choice == 7) break;  
  
        int id, quantity;
```

```
double price;
string name;

switch (choice) {
    case 1:
        cout << "Enter product ID: ";
        cin >> id;
        cout << "Enter product name: ";
        cin.ignore();
        getline(cin, name);
        cout << "Enter quantity: ";
        cin >> quantity;
        cout << "Enter price: ";
        cin >> price;
        inventory.addProduct(id, name, quantity, price);
        break;
    // Other cases follow...
}
}
return 0;
}
```