

A PROJECT REPORT ON
"INVENTORY MANAGEMENT SYSTEM"
SUBMITTED BY:

Mr. Samarth Abhaykumar Tanpure
(2124UCEM1038)

SUBJECT:

PROGRAMING IN PROBLEM SOLVING
USING C++

Under the guidance of
Miss. ISHWARI TIRSE



Department of Computer Science and Engineering

Sanjivani Rural Education Society's

SANJIVANIUNIVERSITY

KOPARGAON – 423603, DIST : AHMEDNAGAR

2024-2025

INDEX

SR NO	CONTENT	PAGE NO.
1.	INTRODUCTION	3
2.	CODE	4
3.	OUTPUT	6
4.	CONCLUSION	9

INTRODUCTION

Inventory Management System (IMS) is a technology solution that helps businesses monitor, manage, and optimize their stock levels, track inventory in real time, and streamline the process of ordering, storing, and using inventory. This system can be implemented through software and hardware, combining tools such as barcode scanners, RFID systems, and IoT devices to provide accurate data and control over products, parts, or materials.

CODE

```
#include <iostream>
#include <vector>
#include <string>
#include <iomanip>
using namespace std;

class Item {
    int id, quantity;
    double price;
    string name;
public:
    Item(int id, const string &name, int quantity, double price)
        : id(id), name(name), quantity(quantity), price(price) {}
    int getId() const { return id; }
    const string& getName() const { return name; }
    void setQuantity(int q) { quantity = q; }
    void setPrice(double p) { price = p; }
    void display() const {
        cout << setw(5) << id << setw(20) << name << setw(10) << quantity << setw(10) << price << '\n';
    }
};

class Inventory {
    vector<Item> items;
public:
    void addItem(int id, const string &name, int quantity, double price) {
        items.emplace_back(id, name, quantity, price);
        cout << "Item added.\n";
    }
    void removeItem(int id) {
        auto it = find_if(items.begin(), items.end(), [=](const Item &item) { return item.getId() == id; });
        if (it != items.end()) { items.erase(it); cout << "Item removed.\n"; }
        else cout << "Item not found.\n";
    }
    void updateItem(int id, int quantity, double price) {
        for (auto &item : items)
            if (item.getId() == id) { item.setQuantity(quantity); item.setPrice(price); cout << "Item updated.\n"; return; }
        cout << "Item not found.\n";
    }
}
```

```

        void viewInventory() const {
cout << setw(5) << "ID" << setw(20) << "Name" << setw(10) << "Qty" << setw(10) << "Price\n";
        cout << "-----\n";
        for (const auto &item : items) item.display();
        }
        void searchItem(int id) const {
for (const auto &item : items) if (item.getId() == id) { item.display(); return; }
        cout << "Item not found.\n";
        }
        void searchItem(const string &name) const {
for (const auto &item : items) if (item.getName() == name) { item.display(); return; }
        cout << "Item not found.\n";
        }
};

        int main() {
Inventory inventory;
        int choice;
        while (true) {
cout << "\n1. Add 2. Remove 3. Update 4. View 5. Search by ID 6. Search by Name 7.
Exit\nChoice: ";
        cin >> choice;
        if (choice == 1) {
int id, quantity; double price; string name;
        cout << "ID: "; cin >> id; cin.ignore();
        cout << "Name: "; getline(cin, name);
        cout << "Quantity: "; cin >> quantity;
        cout << "Price: "; cin >> price;
inventory.addItem(id, name, quantity, price);
        } else if (choice == 2) {
int id; cout << "ID to remove: "; cin >> id;
        inventory.removeItem(id);
        } else if (choice == 3) {
int id, quantity; double price;
        cout << "ID to update: "; cin >> id;
        cout << "New Quantity: "; cin >> quantity;
        cout << "New Price: "; cin >> price;
inventory.updateItem(id, quantity, price);
        } else if (choice == 4) {
inventory.viewInventory();
        } else if (choice == 5) {
int id; cout << "ID to search: "; cin >> id;

```

```
        inventory.searchItem(id);
        } else if (choice == 6) {
            string name; cin.ignore();
            cout << "Name to search: "; getline(cin, name);
            inventory.searchItem(name);
        } else if (choice == 7) break;
        else cout << "Invalid choice.\n";
    }
    return 0;
}
```

OUTPUT

1. Add 2. Remove 3. Update 4. View 5. Search by ID 6. Search by Name 7. Exit

Choice: 1

ID: 101

Name: Widget A

Quantity: 50

Price: 15.75

Item added.

1. Add 2. Remove 3. Update 4. View 5. Search by ID 6. Search by Name 7. Exit

Choice: 4

ID	Name	Qty	Price
101	Widget A	50	15.75

1. Add 2. Remove 3. Update 4. View 5. Search by ID 6. Search by Name 7. Exit

Choice: 1

ID: 102

Name: Widget B

Quantity: 30

Price: 12.50

Item added.

Add 2. Remove 3. Update 4. View 5. Search by ID 6. Search by Name 7. Exit

Choice: 4

ID	Name	Qty	Price

101	Widget A	50	15.75
102	Widget B	30	12.50

1. Add 2. Remove 3. Update 4. View 5. Search by ID 6. Search by Name 7. Exit

Choice: 3

ID to update: 101

New Quantity: 45

New Price: 14.50

Item updated.

1. Add 2. Remove 3. Update 4. View 5. Search by ID 6. Search by Name 7. Exit

Choice: 5

ID to search: 101

101	Widget A	45	14.50
-----	----------	----	-------

1. Add 2. Remove 3. Update 4. View 5. Search by ID 6. Search by Name 7. Exit

Choice: 7

CONCLUSION

Inventory Management System project serves as a basic but effective solution for managing stock in small to medium-sized businesses. It provides essential functionalities such as adding, updating, viewing, and searching for items within the inventory, making it easier for users to keep track of stock levels, prices, and item details.

Through this project, we implemented key programming concepts, including object-oriented principles with the use of classes (Item and Inventory), data encapsulation, and user input handling