Project:-

1. Design a simple inventory management system for a small store using structs and file handling. The program should allow users to add, update, and delete items in the inventory and display the current stock.

Code:-

#include <iostream>

#include <fstream>

#include <vector>

#include <string>

using namespace std;

struct Item {

int id;

string name;

int quantity;

float price;

};

vector<Item> inventory;

void loadInventoryFromFile() {

ifstream file("inventory.txt");

if (file.is\_open()) {

inventory.clear();

while (!file.eof()) {

Item item;

file >> item.id >> item.name >> item.quantity >> item.price;

inventory.push\_back(item);

}

file.close();

}

}

void saveInventoryToFile() {

ofstream file("inventory.txt");

if (file.is\_open()) {

for (const auto& item : inventory) {

file << item.id << " " << item.name << " " << item.quantity << " " << item.price << endl; }

file.close();

}

}

void displayInventory() {

cout << "Current Inventory:\n";

cout << "ID | Name | Quantity | Price\n";

for (const auto& item : inventory) {

cout << item.id << " | " << item.name << " | " << item.quantity << " | " << item.price << endl; }

}

void addItem() {

Item newItem;

cout << "Enter ID: ";

cin >> newItem.id;

cout << "Enter Name: ";

cin >> newItem.name;

cout << "Enter Quantity: ";

cin >> newItem.quantity;

cout << "Enter Price: ";

cin >> newItem.price;

inventory.push\_back(newItem);

saveInventoryToFile();

cout << "Item added to inventory.\n";

}

void updateItem() {

int id;

cout << "Enter the ID of the item to update: "; cin >> id;

for (auto& item : inventory) {

if (item.id == id) {

cout << "Enter new quantity: ";

cin >> item.quantity;

cout << "Enter new price: ";

cin >> item.price;

saveInventoryToFile();

cout << "Item updated.\n";

return;

}

}

cout << "Item with ID " << id << " not found.\n"; }

void deleteItem() {

int id;

cout << "Enter the ID of the item to delete: "; cin >> id;

for (auto it = inventory.begin(); it != inventory.end(); ++it) { if (it->id == id) {

it = inventory.erase(it);

saveInventoryToFile();

cout << "Item deleted.\n";

return;

}

}

cout << "Item with ID " << id << " not found.\n"; }

int main() {

loadInventoryFromFile();

int choice;

do {

cout << "\nInventory Management System\n"; cout << "1. Display Inventory\n";

cout << "2. Add Item\n";

cout << "3. Update Item\n";

cout << "4. Delete Item\n";

cout << "0. Exit\n";

cout << "Enter your choice: ";

cin >> choice;

switch (choice) {

case 1:

displayInventory();

break;

case 2:

addItem();

break;

case 3:

updateItem();

break;

case 4:

deleteItem();

break;

case 0:

cout << "Exiting...\n";

break;

default:

cout << "Invalid choice. Please enter a valid option.\n"; break;

}

} while (choice != 0);

return 0;

}

Output:-

Inventory Management System

1. Display Inventory

2. Add Item

3. Update Item

4. Delete Item

0. Exit

Enter your choice:

1

Current Inventory:

ID | Name | Quantity | Price

Inventory Management System

1. Display Inventory

2. Add Item

3. Update Item

4. Delete Item

0. Exit

Enter your choice:

2

Enter ID:

01

Enter Name:

Pencils

Enter Quantity:

23

Enter Price:

10

Item added to inventory.

Inventory Management System 1. Display Inventory

2. Add Item

3. Update Item

4. Delete Item

0. Exit

Enter your choice:

1

Current Inventory:

ID | Name | Quantity | Price 1 | Pencils | 3 | 10

Inventory Management System 1. Display Inventory

2. Add Item

3. Update Item

4. Delete Item

0. Exit

Enter your choice:

2

Enter ID:

02

Enter Name:

Erasers

Enter Quantity:

25

Enter Price:

55

Item added to inventory.

Inventory Management System 1. Display Inventory

2. Add Item

3. Update Item

4. Delete Item

0. Exit

Enter your choice:

1

Current Inventory:

ID | Name | Quantity | Price 1 | Pencils | 3 | 10

2 | Erasers | 25 | 55

Inventory Management System 1. Display Inventory

2. Add Item

3. Update Item

4. Delete Item

0. Exit

Enter your choice:

3

Enter the ID of the item to update: 1

Enter new quantity:

10

Enter new price:

85

Item updated.

Inventory Management System 1. Display Inventory

2. Add Item

3. Update Item

4. Delete Item

0. Exit

Enter your choice:

1

Current Inventory:

ID | Name | Quantity | Price 1 | Pencils | 10 | 85

2 | Erasers | 25 | 55

Inventory Management System 1. Display Inventory

2. Add Item

3. Update Item

4. Delete Item

0. Exit

Enter your choice:

4

Enter the ID of the item to delete: 02

Item deleted.

Inventory Management System 1. Display Inventory

2. Add Item

3. Update Item

4. Delete Item

0. Exit

Enter your choice:

1

Current Inventory:

ID | Name | Quantity | Price 1 | Pencils | 10 | 85

Inventory Management System 1. Display Inventory

2. Add Item

3. Update Item

4. Delete Item

5. Exit

Enter your choice:

5

Exiting...