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

class Product {
private:
    string name;
    int id;
    int quantity;
    double price;

public:
    Product(string name, int id, int quantity, double price)
        : name(name), id(id), quantity(quantity), price(price) {}

    int getId() const { return id; }
    int getQuantity() const { return quantity; }
    double getPrice() const { return price; }
    string getName() const { return name; }

    void displayProductInfo() const {
        cout << "Product ID: " << id << "\n"
             << "Product Name: " << name << "\n"
             << "Quantity: " << quantity << "\n"
             << "Price: $" << fixed << setprecision(2) << price << "\n";
    }

    void updateQuantity(int amount) {
        if (quantity + amount >= 0) {
            quantity += amount;
        } else {
            cout << "Insufficient quantity for this update.\n";
        }
    }

    double getTotalValue() const { return price * quantity; }

    void displayProduct() const {
        cout << setw(10) << id << setw(15) << name << setw(10) << quantity << setw(10) << price << "\n";
    }
};

class Inventory {
private:
    vector<Product> products;

public:
    void addProduct() {
        if (products.size() >= 100) {
            cout << "Inventory limit reached. Cannot add more products.\n";
            return;
        }

        string name;
        int id, quantity;
        double price;

        cout << "Enter product name: ";
        cin.ignore(); // Clear any leftover newline characters
        getline(cin, name); // Use getline for product name to allow spaces

        cout << "Enter product ID: ";
        while (!(cin >> id) || id <= 0) {
            cout << "Invalid input. Please enter a positive integer for ID: ";
```

```
            cin.clear();
            cin.ignore(numeric_limits<streamsize>::max(), '\n');
        }

        cout << "Enter quantity: ";
        while (!(cin >> quantity) || quantity < 0) {
            cout << "Invalid input. Please enter a non-negative integer for quantity: ";
            cin.clear();
            cin.ignore(numeric_limits<streamsize>::max(), '\n');
        }

        cout << "Enter price: ";
        while (!(cin >> price) || price < 0) {
            cout << "Invalid input. Please enter a non-negative value for price: ";
            cin.clear();
            cin.ignore(numeric_limits<streamsize>::max(), '\n');
        }

        products.emplace_back(name, id, quantity, price);
        cout << "Product added successfully!\n";

        /* Example output:
           Enter product name: Laptop
           Enter product ID: 101
           Enter quantity: 10
           Enter price: 799.99
           Product added successfully!
        */
    }

    void displayAllProducts() const {
        if (products.empty()) {
            cout << "No products in inventory.\n";
            return;
        }

        cout << setw(10) << "ID" << setw(15) << "Name" << setw(10) << "Quantity" << setw(10) << "Price\n";
        cout << "-----------------------------------------\n";
        for (const auto& product : products) {
            product.displayProduct();
        }

        /* Example output:
           ID        Name       Quantity  Price
           -----------------------------------------
           101       Laptop        10      799.99
           102       Mouse         50      19.99
        */
    }

    void updateProductQuantity() {
        int id, changeAmount;
        cout << "Enter product ID to update: ";
        while (!(cin >> id) || id <= 0) {
            cout << "Invalid input. Please enter a positive integer for ID: ";
            cin.clear();
            cin.ignore(numeric_limits<streamsize>::max(), '\n');
        }

        for (auto& product : products) {
            if (product.getId() == id) {
                cout << "Enter quantity to add/subtract (use negative for reduction): ";
                while (!(cin >> changeAmount)) {
                    cout << "Invalid input. Please enter an integer for quantity change: ";
                    cin.clear();
                    cin.ignore(numeric_limits<streamsize>::max(), '\n');
```

```
                }
                product.updateQuantity(changeAmount);
                cout << "Quantity updated successfully!\n";
                return;
            }
        }
        cout << "Product not found.\n";

        /* Example output:
           Enter product ID to update: 101
           Enter quantity to add/subtract (use negative for reduction): 5
           Quantity updated successfully!
        */
    }

    void deleteProduct() {
        int id;
        cout << "Enter product ID to delete: ";
        while (!(cin >> id) || id <= 0) {
            cout << "Invalid input. Please enter a positive integer for ID: ";
            cin.clear();
            cin.ignore(numeric_limits<streamsize>::max(), '\n');
        }

        for (auto it = products.begin(); it != products.end(); ++it) {
            if (it->getId() == id) {
                products.erase(it);
                cout << "Product deleted successfully!\n";
                return;
            }
        }
        cout << "Product not found.\n";

        /* Example output:
           Enter product ID to delete: 102
           Product deleted successfully!
        */
    }

    void calculateTotalInventoryValue() const {
        double totalValue = 0;
        for (const auto& product : products) {
            totalValue += product.getTotalValue();
        }
        cout << "Total inventory value: $" << fixed << setprecision(2) << totalValue << "\n";

        /* Example output:
           Total inventory value: $11999.85
        */
    }
};

int main() {
    Inventory inventory;
    int choice;

    while (true) {
        cout << "\nInventory Management System\n";
        cout << "1. Add New Product\n";
        cout << "2. Display All Products\n";
        cout << "3. Update Product Quantity\n";
        cout << "4. Delete a Product\n";
        cout << "5. Calculate Total Inventory Value\n";
        cout << "6. Exit\n";
        cout << "Enter your choice: ";
        while (!(cin >> choice) || choice < 1 || choice > 6) {
```

```
            cout << "Invalid choice. Please enter a number between 1 and 6: ";
            cin.clear();
            cin.ignore(numeric_limits<streamsize>::max(), '\n');
        }

        switch (choice) {
            case 1:
                inventory.addProduct();
                break;
            case 2:
                inventory.displayAllProducts();
                break;
            case 3:
                inventory.updateProductQuantity();
                break;
            case 4:
                inventory.deleteProduct();
                break;
            case 5:
                inventory.calculateTotalInventoryValue();
                break;
            case 6:
                cout << "Exiting the program.\n";
                return 0;
        }
    }
}
```