Name : Yash Vilas Dhamak

Mail id : yashdhamak2002@gmail.com

Inventory Management System

Code :

```
using System;
using System.Collections.Generic;

class Item
{
    public int ID { get; set; }
    public string Name { get; set; }
    public double Price { get; set; }
    public int Quantity { get; set; }

    public Item(int id, string name, double price, int quantity)
    {
        ID = id;
        Name = name;
        Price = price;
        Quantity = quantity;
```

```
    }

    public override string ToString()
    {
        return $"ID: {ID}, Name: {Name}, Price: {Price}, Quantity: {Quantity}";
    }
}

class Inventory
{
    private List<Item> items;

    public Inventory()
    {
        items = new List<Item>();
    }

    public void AddItem(int id, string name, double price, int quantity)
    {
        Item newItem = new Item(id, name, price, quantity);
        items.Add(newItem);
        Console.WriteLine("Item added successfully!");
    }
```

```
public void DisplayAllItems()
{
   foreach (var item in items)
   {
      Console.WriteLine(item);
   }
}

public Item FindItemByID(int id)
{
   return items.Find(item => item.ID == id);
}

public void UpdateItem(int id)
{
   Item itemToUpdate = FindItemByID(id);
   if (itemToUpdate != null)
   {
      Console.WriteLine("Enter field to update (id, name, price, quantity, all):");
      string fieldToUpdate = Console.ReadLine().ToLower();

      switch (fieldToUpdate)
      {
         case "id":
```

```
    Console.Write("Enter new ID: ");
    int newId = int.Parse(Console.ReadLine());
    itemToUpdate.ID = newId;
    break;
case "name":
    Console.Write("Enter new name: ");
    itemToUpdate.Name = Console.ReadLine();
    break;
case "price":
    Console.Write("Enter new price: ");
    itemToUpdate.Price = double.Parse(Console.ReadLine());
    break;
case "quantity":
    Console.Write("Enter new quantity: ");
    itemToUpdate.Quantity = int.Parse(Console.ReadLine());
    break;
case "all":
    Console.Write("Enter new ID: ");
    int newID = int.Parse(Console.ReadLine());
    Console.Write("Enter new name: ");
    itemToUpdate.Name = Console.ReadLine();
    Console.Write("Enter new price: ");
    itemToUpdate.Price = double.Parse(Console.ReadLine());
    Console.Write("Enter new quantity: ");
```

```
                itemToUpdate.Quantity = int.Parse(Console.ReadLine());
                break;
            default:
                Console.WriteLine("Invalid field!");
                break;
        }
        Console.WriteLine("Item updated successfully!");
    }
    else
    {
        Console.WriteLine("Item not found!");
    }
}

public void DeleteItem(int id)
{
    Item itemToDelete = FindItemByID(id);
    if (itemToDelete != null)
    {
        items.Remove(itemToDelete);
        Console.WriteLine("Item deleted successfully!");
    }
    else
    {
```

```
            Console.WriteLine("Item not found!");
        }
    }
}

class Program
{
    static void Main(string[] args)
    {
        Inventory inventory = new Inventory();

        while (true)
        {
            Console.WriteLine("----------------- Welcome To Inventory Management System -------------------");
            Console.WriteLine("\nMenu:");
            Console.WriteLine("1. Add Item");
            Console.WriteLine("2. Display All Items");
            Console.WriteLine("3. Find Item by ID");
            Console.WriteLine("4. Update Item");
            Console.WriteLine("5. Delete Item");
            Console.WriteLine("6. Exit");

            Console.Write("Enter your choice: ");
            int choice = int.Parse(Console.ReadLine());
```

```
switch (choice)
{
    case 1:
        Console.WriteLine("Enter item details:");
        Console.Write("ID: ");
        int id = int.Parse(Console.ReadLine());
        Console.Write("Name: ");
        string name = Console.ReadLine();
        Console.Write("Price: ");
        double price = double.Parse(Console.ReadLine());
        Console.Write("Quantity: ");
        int quantity = int.Parse(Console.ReadLine());

        inventory.AddItem(id, name, price, quantity);
        break;
    case 2:
        Console.WriteLine("All Items:");
        inventory.DisplayAllItems();
        break;
    case 3:
        Console.Write("Enter item ID to find: ");
        int idToFind = int.Parse(Console.ReadLine());
        Item foundItem = inventory.FindItemByID(idToFind);
```

```
            if (foundItem != null)
            {
                Console.WriteLine("Found Item:");
                Console.WriteLine(foundItem);
            }
            else
            {
                Console.WriteLine("Item not found!");
            }
            break;
        case 4:
            Console.Write("Enter item ID to update: ");
            int idToUpdate = int.Parse(Console.ReadLine());
            inventory.UpdateItem(idToUpdate);
            break;
        case 5:
            Console.Write("Enter item ID to delete: ");
            int idToDelete = int.Parse(Console.ReadLine());
            inventory.DeleteItem(idToDelete);
            break;
        case 6:
            Console.WriteLine("Exiting program...");
            Environment.Exit(0);
            break;
```

```
            default:
                Console.WriteLine("Invalid choice!");
                break;
        }
    }
  }
}
```

Description :

This inventory management system features functionalities like adding, displaying, finding, updating, and deleting items, all accessible through a console interface. It efficiently manages inventory tasks, offering a user-friendly experience with flexible options for updating items. Its error handling ensures data integrity and provides informative feedback. Overall, it streamlines inventory processes, enhances control, and simplifies management tasks for users.

Key Features:

1. Add Item: Users can add new items to the inventory by providing details such as ID, name, price, and quantity through the console interface.

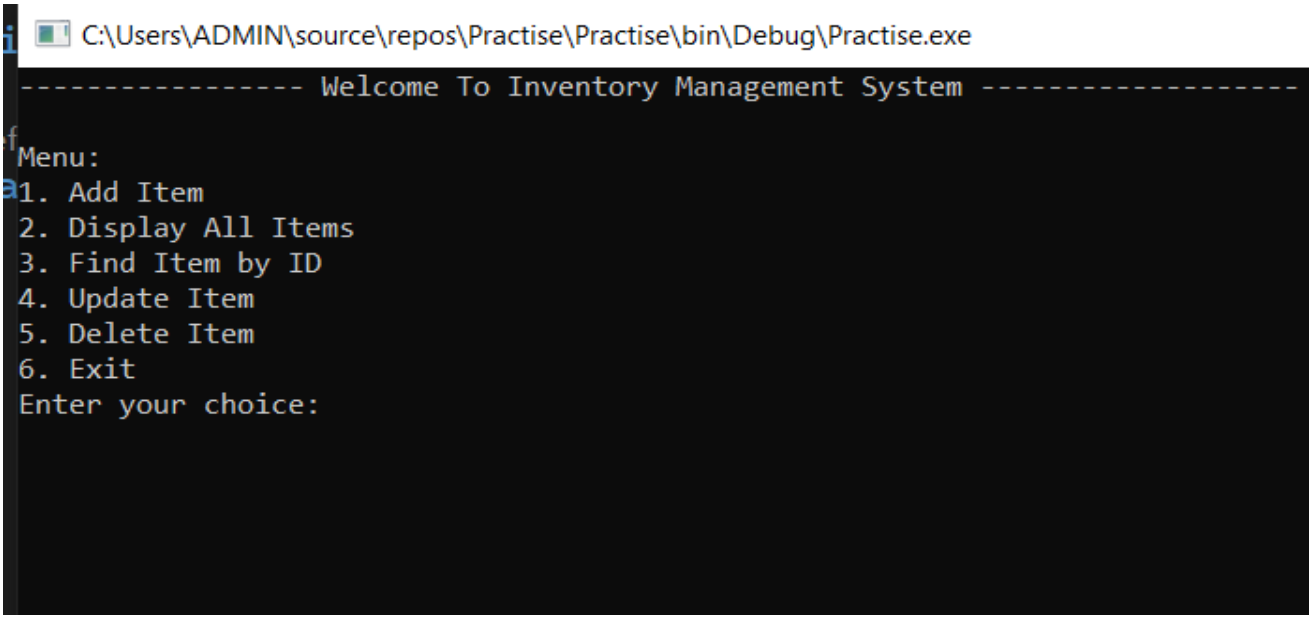
2. Display All Items: Users can view the complete list of items in the inventory, including their ID, name, price, and quantity.

3. Find Item by ID: Users can search for a specific item in the inventory by entering its unique ID.

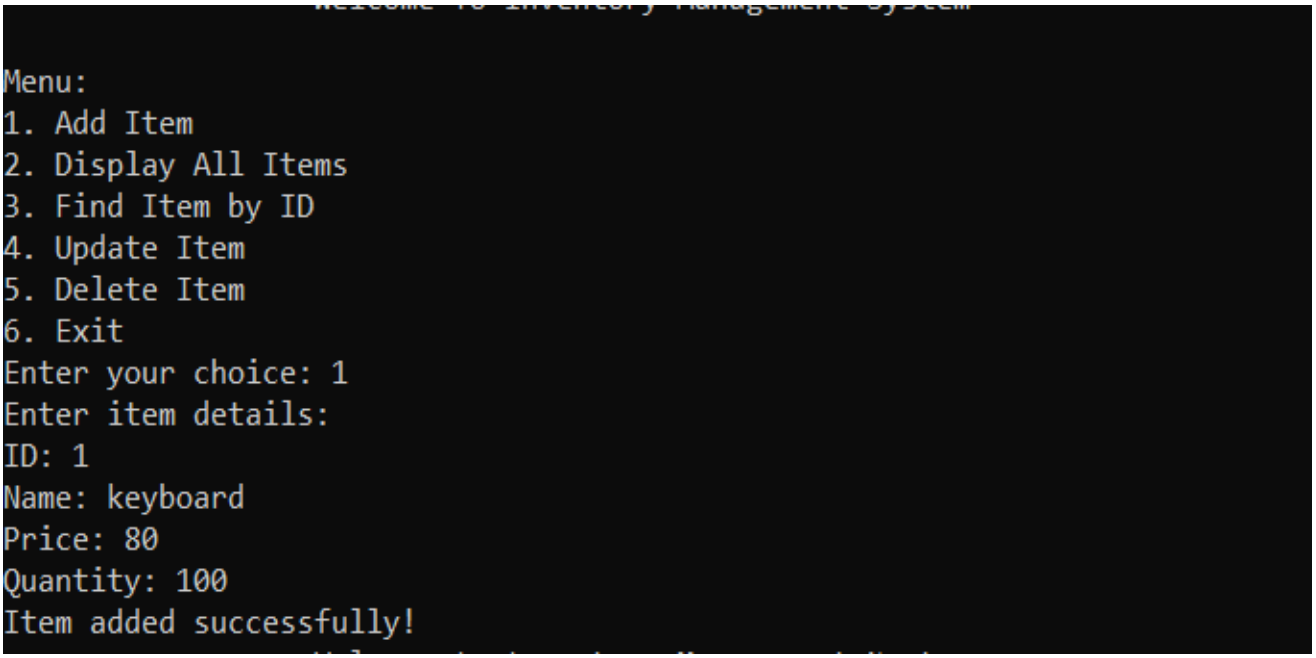
4. Update Item: Users have the flexibility to update item details individually (name, price, quantity) or all fields at once, including the item's ID.

5. Delete Item: Users can remove an item from the inventory by specifying its ID.

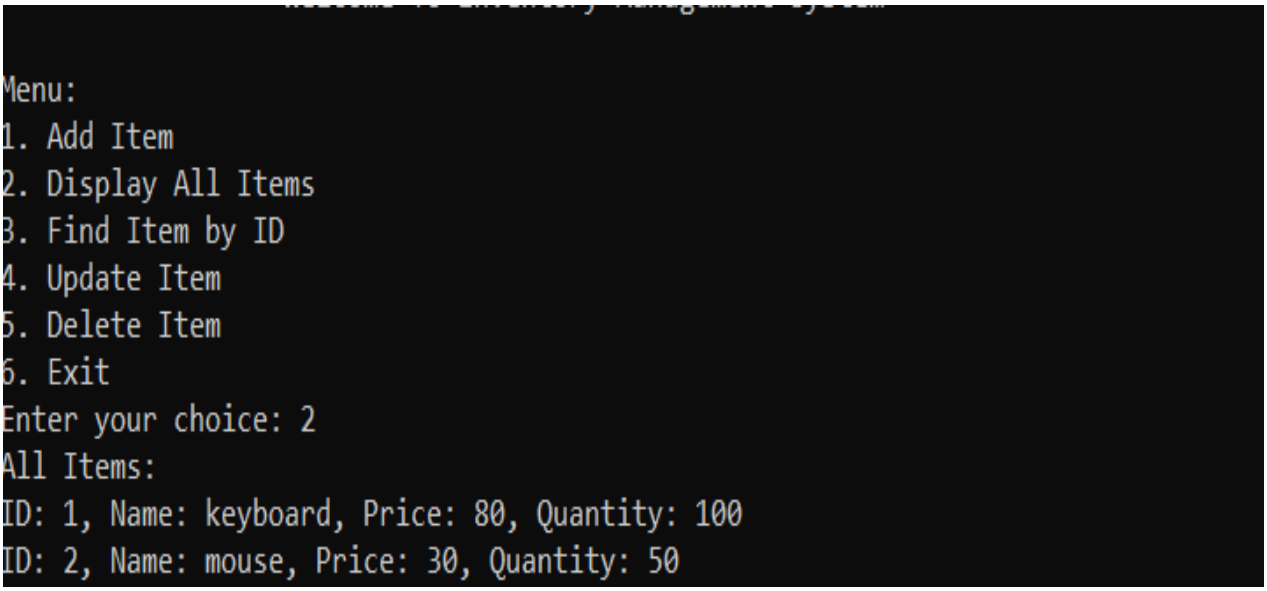
1> Console Login page :



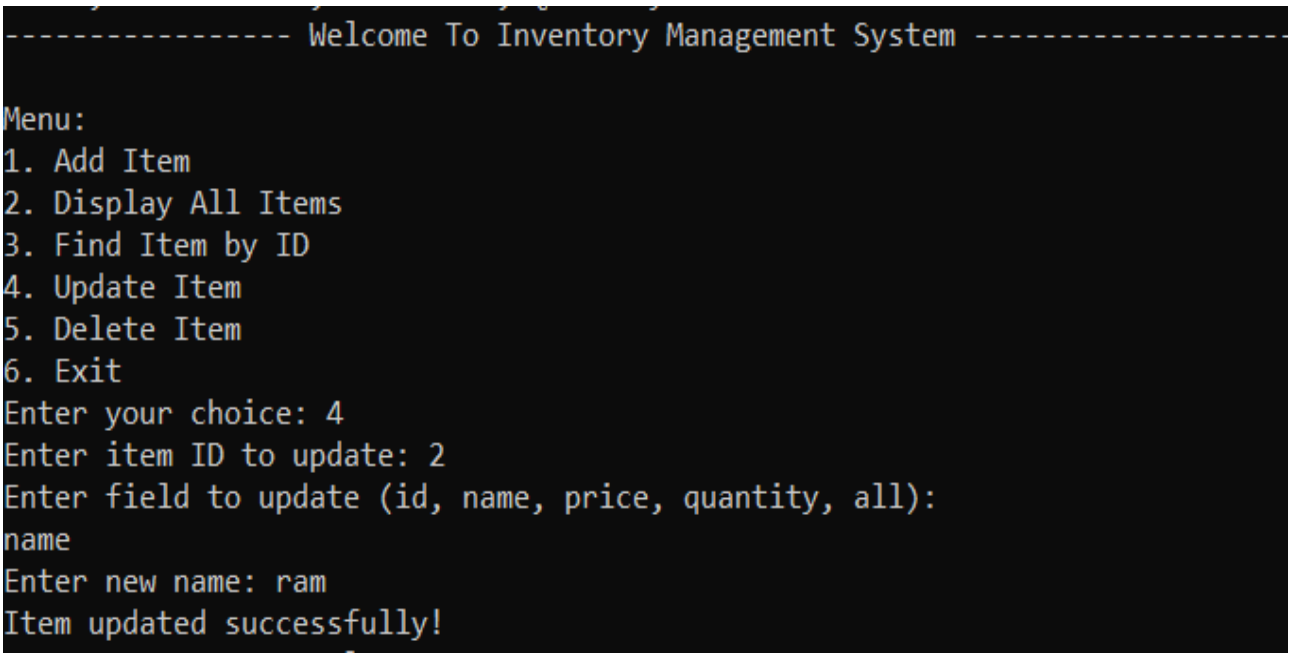
2>Add item in inventory system



3>display item details by giving its id



4>update element by providing its ID(you can update name , id, quantity , price separate)



5>console overview