

INVENTORY MANAGEMENT FOR A SMALL SHOP

```
#include <stdio.h>
#include <stdlib.h>
#include <string.h>

#define DB_FILE "inventory.dat"

typedef struct {
    int id;
    char name[30];
    int quantity;
    float price;
} Item;

void clearInputBuffer(void) {
    int c;
    while ((c = getchar()) != '\n' && c != EOF) {}
}

int readLine(char *buf, size_t size) {
    if (fgets(buf, (int)size, stdin) == NULL) return 0;
    buf[strcspn(buf, "\n")] = '\0';
    return 1;
}
```

```
void addItem(void) {  
    FILE *fp = fopen(DB_FILE, "ab");  
    if (!fp) {  
        printf("Error: Cannot open database file.\n");  
        return;  
    }  
  
    Item item;  
    printf("Enter Item ID: ");  
    if (scanf("%d", &item.id) != 1) { clearInputBuffer(); fclose(fp); return; }  
    clearInputBuffer();  
  
    printf("Enter Item Name: ");  
    if (!readLine(item.name, sizeof(item.name))) { fclose(fp); return; }  
  
    printf("Enter Quantity: ");  
    if (scanf("%d", &item.quantity) != 1) { clearInputBuffer(); fclose(fp); return; }  
  
    printf("Enter Price: ");  
    if (scanf("%f", &item.price) != 1) { clearInputBuffer(); fclose(fp); return; }  
  
    fwrite(&item, sizeof(item), 1, fp);  
    fclose(fp);  
  
    printf("Item added successfully.\n");  
}
```

```
void displayItems(void){  
    FILE *fp = fopen(DB_FILE, "rb");  
  
    if (!fp) {  
        printf("No inventory file found. Add items first.\n");  
  
        return;  
    }  
  
    Item item;  
    int any = 0;  
  
    printf("\n%-6s %-30s %-10s %-10s\n", "ID", "NAME", "QTY", "PRICE");  
    printf("-----\n");  
  
    while (fread(&item, sizeof(Item), 1, fp) == 1) {  
        any = 1;  
  
        printf("%-6d %-30s %-10d %-10.2f\n", item.id, item.name, item.quantity, item.price);  
    }  
  
    if (!any) printf("(No items found)\n");  
  
    fclose(fp);  
}  
  
void searchItem(void) {  
    FILE *fp = fopen(DB_FILE, "rb");  
  
    if (!fp) {
```

```
printf("No inventory file found.\n");

return;
}

int id, found = 0;

printf("Enter Item ID to search: ");

if (scanf("%d", &id) != 1) { clearInputBuffer(); fclose(fp); return; }

Item item;

while (fread(&item, sizeof(Item), 1, fp) == 1) {

    if (item.id == id) {

        found = 1;

        printf("\nFound:\nID: %d\nName: %s\nQuantity: %d\nPrice: %.2f\n",
               item.id, item.name, item.quantity, item.price);

        break;
    }
}

if (!found) printf("Item not found.\n");

fclose(fp);
}

void updateItem(void) {

FILE *fp = fopen(DB_FILE, "rb+");

if (!fp) {

printf("No inventory file found.\n");

return;
}
```

```
}
```

```
int id, found = 0;

printf("Enter Item ID to update: ");

if (scanf("%d", &id) != 1) { clearInputBuffer(); fclose(fp); return; }

clearInputBuffer();

Item item;

while (fread(&item, sizeof(Item), 1, fp) == 1) {

    if (item.id == id) {

        found = 1;

        printf("Current Name: %s\n", item.name);

        printf("Enter new name (press Enter to keep same): ");

        char newName[30];

        if (!readLine(newName, sizeof(newName))) { fclose(fp); return; }

        if (strlen(newName) > 0) strncpy(item.name, newName, sizeof(item.name) - 1);

        printf("Current Quantity: %d\n", item.quantity);

        printf("Enter new quantity: ");

        if (scanf("%d", &item.quantity) != 1) { clearInputBuffer(); fclose(fp); return; }

        printf("Current Price: %.2f\n", item.price);

        printf("Enter new price: ");

        if (scanf("%f", &item.price) != 1) { clearInputBuffer(); fclose(fp); return; }

        fseek(fp, -(long)sizeof(item), SEEK_CUR);
```

```
    fwrite(&item, sizeof(item), 1, fp);

    printf("Item updated successfully.\n");
    break;
}

}

if (!found) printf("Item not found.\n");

fclose(fp);
}

void deleteItem(void) {
    FILE *fp = fopen(DB_FILE, "rb");
    if (!fp) {
        printf("No inventory file found.\n");
        return;
    }

    FILE *temp = fopen("temp.dat", "wb");
    if (!temp) {
        printf("Error: Cannot create temp file.\n");
        fclose(fp);
        return;
    }

    int id, found = 0;
    printf("Enter Item ID to delete: ");
```

```
if (scanf("%d", &id) != 1) { clearInputBuffer(); fclose(fp); fclose(temp); return; }

Item item;

while (fread(&item, sizeof(item), 1, fp) == 1) {

    if (item.id == id) {

        found = 1; // skip writing this record

    } else {

        fwrite(&item, sizeof(item), 1, temp);

    }

}

fclose(fp);

fclose(temp);

remove(DB_FILE);

rename("temp.dat", DB_FILE);

if (found) printf("Item deleted successfully.\n");

else printf("Item not found.\n");

}

int main(void){

    int choice;

    while (1){

        printf("\n===== SMALL SHOP INVENTORY (C) =====\n");

        printf("1. Add Item\n");
    }
}
```

```
printf("2. View All Items\n");
printf("3. Search Item (by ID)\n");
printf("4. Update Item\n");
printf("5. Delete Item\n");
printf("0. Exit\n");

printf("Enter choice: ");

if (scanf("%d", &choice) != 1) {
    clearInputBuffer();
    printf("Invalid input.\n");
    continue;
}

switch (choice) {
    case 1: addItem(); break;
    case 2: displayItems(); break;
    case 3: searchItem(); break;
    case 4: updateItem(); break;
    case 5: deleteItem(); break;
    case 0: printf("Exiting...\n"); return 0;
    default: printf("Invalid choice.\n");
}
}
```