

Online Shop Management System

KMUTT



Workload

- *Basic Inventory Function:* Together
- *Advance Inventory Function:* Thanakorn Soonjaw ID: 65070503450
Kanchai Lerdsrisakulrat ID: 65070503475
- *Accounting and POS:* Thanapat Ngoennet ID: 65070503449
Parunchai Kochseni ID: 65070503456
Phakalpol Maneesopa ID: 65070503459
- *Debug & Improve:* Together

Main

OUTPUT:

```
CPE100 Module 3 : Online Shop Management System
-----
Group Members
-----
65070503449 Thanaphat Ngoennet
65070503450 Thanakorn Soonjaw
65070503456 Parunchai Kochseni
65070503459 Phakalpol Maneesopa
65070503475 Kanchai Lerdsrisakulrat
-----
Press 1 to enter the program
Press 0 to exit the program
-----
```

SOURCE CODE :

```
○ ○ ○
#include <stdio.h>
#include <stdlib.h>
#include <conio.h>
#include "login.c"
#include "mainmenuselection.c"
#include "inventorysystem.c"
#include "accountingsystem.c"
#include "possystem.c"

int mainmenuselection();
int inventorysystem();
int accountingsystem();
int possystem();
int login();

int main() {
    printf("-----\n");
    printf("CPE100 Module 3 : Online Shop Management System\n");
    printf("-----\n");
    printf("          Group Members\n");
    printf("-----\n");
    printf("          65070503449 Thanaphat Ngoennet\n");
    printf("          65070503450 Thanakorn Soonjaw\n");
    printf("          65070503456 Parunchai Kochseni\n");
    printf("          65070503459 Phakalpol Maneesopa\n");
    printf("          65070503475 Kanchai Lerdsrisakulrat\n");
    printf("-----\n");
    printf("          Press 1 to enter the program\n");
    printf("          Press 0 to exit the program\n");
    printf("-----\n");
    printf("\nChoice --> ");
    int main_menu;
    scanf("%d", &main_menu);
    switch(main_menu) {
        case 1:
            system("cls");
            login();
            break;
        case 0:
            exit(1);
            break;
        default:
            system("cls");
            printf("      Your choice was wrong please try again!!\n\n");
            main();
            break;
    }
    getch();
    return 0;
}
```

2022



Login

OUTPUT :

```
-----  
          LOGIN  
-----  
Press 1 to login as client  
Press 2 to login as administrator  
Press 0 to exit the program  
-----  
Choice -->
```

SOURCE CODE :

```
-----  
int login() {  
    printf("-----\n");  
    printf("          LOGIN\n");  
    printf("-----\n");  
    printf("          Press 1 to login as client\n");  
    printf("          Press 2 to login as administrator\n");  
    printf("          Press 0 to exit the program\n");  
    printf("-----\n");  
    printf("\nChoice --> ");  
    int choice;  
    scanf("%d", &choice);  
    switch(choice) {  
        case 1:  
            system("cls");  
            clientmainmenuselection();  
            break;  
        case 2:  
            system("cls");  
            admin_log();  
            break;  
        case 0:  
            exit(1);  
            break;  
        default:  
            system("cls");  
            printf("      Your choice was wrong please try again!!\n\n");  
            login();  
            break;  
    }  
}
```

2022



Client Menu

OUTPUT:

```
MAIN MENU
-----
Press 1 to enter POS system
Press 2 to enter login menu
Press 0 to exit the program
```

SOURCE CODE :

```
int clientmainmenuselection() {
    printf("-----|\n");
    printf("|           MAIN MENU           |\n");
    printf("-----|\n");
    printf("|       Press 1 to enter POS system |\n");
    printf("|       Press 2 to enter login menu |\n");
    printf("|       Press 0 to exit the program |\n");
    printf("-----|\n");
    printf("\nChoice --> ");
    int button;
    scanf("%d", &button);
    switch(button) {
        case 1:
            system("cls");
            possystem();
            break;
        case 2:
            system("cls");
            login();
            break;
        case 0:
            system("cls");
            exit(1);
            break;
        default:
            system("cls");
            printf("      Your choice was wrong please try again!!\n\n\n");
            clientmainmenuselection();
            break;
    }
}
```

2022

Point Of Sales

OUTPUT :

```
Point of sale
-----
Press 1 to view product
Press 2 to purchase product
Press 3 to view daily summary
Press 4 to view weekly summary
Press 5 to back to main menu
Press 0 to exit the program
-----
Choice -->
```

SOURCE CODE :

```
○○○
#include <stdio.h>
#include <string.h>
#include <stdlib.h>
#include <time.h>
#define true 0
#define false 1

int accountingsystem();
int mainmenuselection();
int inventorysystem();
int IDChecker(int i, int j);
int checkID(int id);
int readFile();
int writeFile();
void purchaseproduct();
void viewproduct();
void daily();
void weekly();
void adminweekly();
void admindaily();

FILE *p;

int writeposFile() {
    int i;
    p = fopen("pos.txt", "w");
    if(p == NULL) {
        return -1;
    }
    fprintf(p, "%d\n", countpos);
    for(i = 0; i < countpos; ++i) {
        fprintf(p, "%d\n", buy[i].id);
        fputs(buy[i].name, p);
        fprintf(p, "\n");
        fprintf(p, "%d\n", buy[i].quantity);
        fprintf(p, "%f\n", buy[i].price);
        fprintf(p, "%f\n", buy[i].principle);
        fputs(buy[i].timestp, p);
        fprintf(p, "\n");
        fprintf(p, "%d\n", buy[i].time);
    }
    fclose(p);
    return 0;
}

int readposFile() {
    int n = 0;
    int i;
    p = fopen("pos.txt", "r");
    if(p == NULL) {
        return -1;
    }
    fscanf(p, "%d\n", &n);
    for(i = 0; i < n; ++i) {
        fscanf(p, "%d\n", &buy[i].id);
        fgets(buy[i].name, 20,p);
        buy[i].name[strlen(buy[i].name) - 1] = 0;
        fscanf(p, "%d\n", &buy[i].quantity);
        fscanf(p, "%f\n", &buy[i].price);
        fscanf(p, "%f\n", &buy[i].principle);
        fgets(buy[i].timestp, 30,p);
        buy[i].timestp[strlen(buy[i].timestp) - 1] = 0;
        fscanf(p, "%d\n", &buy[i].time);
    }
    fclose(p);
    return n;
}

int possystem() {
printf("|-----\n");
printf(" Point of sale\n");
printf("-----\n");
printf(" Press 1 to view product\n");
printf(" Press 2 to purchase product\n");
printf(" Press 3 to view daily summary\n");
printf(" Press 4 to view weekly summary\n");
printf(" Press 5 to back to main menu\n");
printf(" Press 0 to exit the program\n");
printf("-----\n");
printf("\nChoice --> ");
int choice;
scanf("%d", &choice);
switch(choice) {
    case 1:
        system("cls");
        viewproduct();
        previousPOS();
        break;
    case 2:
        system("cls");
        purchaseproduct();
        break;
    case 3:
        system("cls");
        daily();
        previousdaily();
        break;
    case 4:
        system("cls");
        weekly();
        previousweekly();
        break;
    case 5:
        system("cls");
        clientmainmenuselection();
        break;
    case 0:
        exit(1);
        break;
    default:
        system("cls");
        printf("Your choice was wrong please try again!!\n");
        possystem();
        break;
}
}
```



View Product (POS)

OUTPUT :

INVENTORY				
NAME	PROD ID	QUANTITY	MINNIMUM QUANTITY	PRICE
aa	1	6	1	20.00
b	2	25	1	6.00
c	3	10	1	15.00
dd	4	57	1	3.00

Press 0 to back to POS system:

SOURCE CODE :

```
○○○
void viewproduct() {
    int i;
    count = readFile();
    if(count < 0) {
        puts("cannot open file");
    }
    printf("|-----|-----|-----|-----|-----|\n");
    printf("|-----|-----|-----|-----|-----|\n");
    printf("|-----|-----|-----|-----|-----|\n");
    printf("|-----|-----|-----|-----|-----|\n");
    printf("|-----|-----|-----|-----|-----|\n");
    for(i = 0; i < count; i++) {
        printf("| %-13s %-11d %-10d %-8d %-8.2f |\n", product[i].name,
product[i].id, product[i].quantity, product[i].min_quan, product[i].price);
    }
    printf("|-----|-----|-----|-----|-----|\n");
    printf("\n\n\n");
```

```
○○○
void previousPOS() {
    int choice;
    printf("Press 0 to back to POS system: ");
    scanf("%d", &choice);
    switch(choice) {
        case 0:
            system("cls");
            possystem();
            break;
        default:
            system("cls");
            viewproduct();
            break;
    }
}
```



Purchase Product

OUTPUT :

INVENTORY				
NAME	PROD ID	QUANTITY	MINNIMUM QUANTITY	PRICE
aa	1	5	1	20.00
b	2	25	1	6.00
c	3	10	1	15.00
dd	4	57	1	3.00

Choose product's ID

Product ID: 1

Are you sure to buy this product?

Sure press 1.

Not Sure press 0 and choose product.

Choice --> ■

SOURCE CODE:

Purchase Part 1:

```
○ ○ ○
void purchaseproduct() {
    viewproduct();
    time_t t = time(NULL);
    char *str_time = ctime(&t);
    int quant, i, id, choice, z = false, temp1 = false, temp2 = false, temp3 = false, c = false;
    count = readFile();
    countpos = readposFile();
    printf("Choose product's ID\n");
    printf("Product ID: ");
    fflush(stdin);
    scanf("%d", &id);
    printf("Are you sure to buy this product?\nSure press 1. \nNot Sure press 0 and choose
product.\nChoice --> ");
    scanf("%d", &choice);
    switch (choice){
        case 1:
            break;
        case 0:
            purchaseproduct();
            break;
        default:
            purchaseproduct();
            break;
    }
}
```

Purchase Product

OUTPUT :

```

INVENTORY
NAME | PROD ID | QUANTITY | MINNIMUM QUANTITY | PRICE
aa      1          5              1            20.00
b       2         25             1            6.00
c       3         10             1            15.00
dd      4         57             1            3.00

Choose product's ID
Product ID: 1
Are you sure to buy this product?
Sure press 1.
Not Sure press 0 and choose product.
Choice --> -

```

```

INVENTORY
NAME | PROD ID | QUANTITY | MINNIMUM QUANTITY | PRICE
aa      1          5              1            20.00

Enter the quantity you want to buy: 1
Successful purchase!
Press 1 to purchase another product / Press 2 to back to POS menu:

```

Purchase Part 2 :

SOURCE CODE :

```

OOO
for(i = 0; i < count; i++) {
    if(id == product[i].id) {
        z = true;
        printf("\nItem available!\n");
        cartviewproductforpurchase(id);
        printf("Enter the quantity you want to buy: ");
        fflush(stdin);
        scanf("%d", &quant);
        if(quant <= product[i].quantity && quant >= product[i].min_quan) {
            printf("\nSuccessful purchase!\n");
            temp1 = true;
        }
        else if(quant > product[i].quantity) {
            printf("\nInsufficient quantity please restock!\n");
            temp2 = true;
            break;
        }
        else if(quant < product[i].min_quan) {
            printf("\nSorry, but there are minimum quantities for each item!\n");
            temp3 = true;
            break;
        }
        product[i].quantity -= quant;
        if(countpos > 0) {
            countpos = readposFile();
        }
        buy[countpos].id = product[i].id;
        strcpy(buy[countpos].name, product[i].name);
        buy[countpos].quantity = quant;
        buy[countpos].price = product[i].price;
        buy[countpos].principle = product[i].principle;
        strcpy(buy[countpos].timestp, str_time);
        buy[countpos].time = t;
        ++countpos;
    }
}
if(z == false) {
    printf("Cant find the product id: %d\n\n", id);
    IDPurchaseCorrection();
}
writeFile();
writeposFile();
if(temp1 == true) {
    successfulpurchase();
}
else if(temp2 == true) {
    insufficientQuant();
}
else if(temp3 == true) {
    wrongminimumQuant();
}
previousPOS();
}

void previousPOS() {
    int choice;
    printf("Press 0 to back to POS system: ");
    scanf("%d", &choice);
    switch(choice) {
        case 0:
            system("cls");
            possystem();
            break;
        default:
            system("cls");
            viewproduct();
            break;
    }
}

```

Daily summary

OUTPUT :

DAY SUMMARY (COUNT BY FIRST PRODUCT)						
DAY	NAME	PROD ID	QUANTITY	PRICE	TIME	
1	aa	1	1	20.00	Thu Dec 15	11:50:20 2022
1	dd	4	1	3.00	Thu Dec 15	11:50:28 2022
1	aa	1	1	20.00	Thu Dec 15	12:53:08 2022
1	aa	1	1	20.00	Thu Dec 15	12:53:46 2022
TOTAL OF DAY 1 = 63.00						
DAY SUMMARY (COUNT BY FIRST PRODUCT)						
DAY	NAME	PROD ID	QUANTITY	PRICE	TIME	
2	bb	2	3	45.00	Fri Dec 16	15:33:32 2022
2	cc	3	1	56.00	Fri Dec 16	15:33:43 2022
TOTAL OF DAY 2 = 101.00						
DAY SUMMARY (COUNT BY FIRST PRODUCT)						
DAY	NAME	PROD ID	QUANTITY	PRICE	TIME	
3	cc	3	1	56.00	Sat Dec 17	15:34:54 2022
TOTAL OF DAY 3 = 56.00						

SOURCE CODE :

```

void daily() {
    int i = 0, flag = 0, j, choice;
    float total, totalarry[1000];
    countpos = readposFile();
    int temp = countpos;
    if(countpos < 0) {
        puts("Cannot open file");
    }
    printf("|-----|\n");
    printf("|\n");
    printf("|\n");
    while(i >= 0) {
        total = 0;
        printf("|\n");
        printf("|\n");
        printf("|\n");
        for(j = 0; j < countpos; j++) {
            if((buy[j].time - buy[0].time) >= i * 86400 && (buy[j].time - buy[0].time) < (i + 1) *
86400) {
                total = total + (buy[j].price * buy[j].quantity);
                printf(" | %4d %11s %11d %5d %9.2f %12s |\n", i + 1,
buy[j].name, buy[j].id, buy[j].quantity, buy[j].price * buy[j].quantity, buy[j].timestp);
                temp--;
                flag++;
            }
        }
        totalarry[i] = total;
        printf("|\n");
        printf("|\n");
        printf("|\n");
        if(flag == 0) {
            printf("\n");
        }
        if(temp == 0) {
            break;
        }
        i++;
    }
}

```



Weekly summary

OUTPUT :

WEEK SUMMARY (COUNT BY FIRST PRODUCT)						
Week	NAME	PROD ID	QUANTITY	PRICE	TIME	
1	aa	1	1	20.00	Thu Dec 15 11:50:20 2022	
1	dd	4	1	3.00	Thu Dec 15 11:50:28 2022	
1	aa	1	1	20.00	Thu Dec 15 12:53:08 2022	
1	aa	1	1	20.00	Thu Dec 15 12:53:46 2022	
1	bb	2	3	45.00	Fri Dec 16 15:33:32 2022	
1	cc	3	1	56.00	Fri Dec 16 15:33:43 2022	
1	cc	3	1	56.00	Sat Dec 17 15:34:54 2022	
1	dd	4	1	3.00	Sun Dec 18 15:35:53 2022	
1	aa	1	1	20.00	Sun Dec 18 15:36:11 2022	
1	bb	2	3	45.00	Sun Dec 18 15:36:21 2022	
TOTAL OF WEEK 1 = 288.00						
Week	NAME	PROD ID	QUANTITY	PRICE	TIME	
2	dd	4	1	3.00	Thu Dec 22 15:24:34 2022	
2	dd	4	2	6.00	Thu Dec 22 15:27:31 2022	
2	aa	1	1	20.00	Fri Dec 23 15:29:55 2022	
2	dd	4	1	3.00	Fri Dec 23 15:30:07 2022	
2	dd	4	1	3.00	Sat Dec 24 15:30:36 2022	
2	aa	1	1	20.00	Sat Dec 24 15:30:48 2022	
TOTAL OF WEEK 2 = 55.00						

SOURCE CODE :

```
○ ○ ○

void weekly() {
    int i = 0, flag = 0, j, choice;
    float total, totalarry[1000];
    countpos = readposFile();
    int temp = countpos;
    if(countpos < 0) {
        puts("Cannot open file");
    }
    printf("|-----\n");
    printf("|   \n");
    while(i >= 0) {
        total = 0;
        printf("|-----\n");
        printf("| Week |      NAME      | PROD ID | QUANTITY | PRICE | TIME\n");
        printf("|   \n");
        printf("|-----\n");
        for(j = 0; j < countpos; j++) {
            if((buy[j].time - buy[0].time) >= i * 604800 && (buy[j].time - buy[0].time) < (i + 1) *
604800) {
                total = total + (buy[j].price * buy[j].quantity);
                printf("| %4d %11s      %11d      %5d      %-9.2f      %-12s |\n", i + 1,
buy[j].name, buy[j].id, buy[j].quantity, buy[j].price * buy[j].quantity, buy[j].timestp);
                temp--;
                flag++;
            }
        }
        totalarry[i] = total;
        printf("|-----\n");
        printf("|   \n");
        printf("TOTAL OF WEEK %d = %-8.2f\n", i + 1, totalarry[i]);
        printf("|-----\n");
        printf("|\n\n");
        if(flag == 0) {
            printf("\n");
        }
        if(temp == 0) {
            break;
        }
        i++;
    }
}
```

Login as Admin

OUTPUT :

```
|-----|  
|      Login as administrator|  
|-----|  
  
Enter username: admin  
  
Enter password: ■
```

SOURCE CODE :

```
○ ○ ○  
  
void admin_log() {  
    char adminusername[100] = "admin";  
    char adminpassword[100] = "cpe";  
    char inputuser[100];  
    char inputpassword[100];  
    printf("|\n-----|\n");  
    printf("|\n      Login as administrator\n      |\n-----|\n");  
    printf("\nEnter username: ");  
    scanf("%s", &inputuser);  
    printf("\nEnter password: ");  
    scanf("%s", &inputpassword);  
    if(strcmp(adminusername, inputuser) == 0 && (strcmp(adminpassword, inputpassword)) == 0) {  
        system("cls");  
        mainmenuselection();  
    } else {  
        printf("\nInvalid username or password\n");  
        previouslogin();  
    }  
}
```

```
○ ○ ○  
  
#include <string.h>  
  
int mainmenuselection();  
int inventorysystem();  
int accountingsystem();  
int possystem();  
int login();  
void admin_log();  
int clientmainmenuselection();  
  
void previouslogin() {  
    int choice;  
    printf("Press 1 to try again / Press 2 to back to login menu: ");  
    scanf(" %d", &choice);  
    switch(choice) {  
    case 1:  
        system("cls");  
        admin_log();  
        break;  
    case 2:  
        system("cls");  
        login();  
        break;  
    default:  
        system("cls");  
        printf("Your choice was wrong please try again!!\n");  
        previouslogin();  
        break;  
    }  
}
```

2022



Admin Menu

OUTPUT:

```
MAIN MENU

Press 1 to enter inventory system
Press 2 to enter accounting system
Press 3 to enter daily summary
Press 4 to enter weekly summary
Press 5 to back to Login menu
Press 0 to exit the program
```

SOURCE CODE :

2022

```
int mainmenuselection() {
    printf("-----|\\n");
    printf("          MAIN MENU           |\\n");
    printf("-----|\\n");
    printf("          Press 1 to enter inventory system |\\n");
    printf("          Press 2 to enter accounting system |\\n");
    printf("          Press 3 to enter daily summary     |\\n");
    printf("          Press 4 to enter weekly summary     |\\n");
    printf("          Press 5 to back to Login menu       |\\n");
    printf("          Press 0 to exit the program         |\\n");
    printf("-----|\\n");
    printf("\nChoice --> ");
    int button;
    scanf("%d", &button);
    switch(button) {
        case 1:
            system("cls");
            inventoriesystem();
            break;
        case 2:
            system("cls");
            accountingsystem();
            break;
        case 3:
            system("cls");
            admindaily();
            previousdailyadmin();
            break;
        case 4:
            system("cls");
            adminweekly();
            previousweeklyadmin();
            break;
        case 5:
            system("cls");
            login();
            break;
        case 0:
            exit(1);
            break;
        default:
            system("cls");
            printf("      Your choice was wrong please try again!!\n\n\n");
            mainmenuselection();
            break;
    }
}
```



Inventory System

OUTPUT:

```
INVENTORY PROGRAM

Press 1 to add product
Press 2 to modify product
Press 3 to delete product
Press 4 to display all exiting product
Press 5 to back to main menu
Press 0 to exit the program
```

```
#include <stdio.h>
#include <string.h>
#include <stdlib.h>
#define true 0
#define false 1

int mainmenuselection();
int accountingsystem();
int possystem();
int IDChecker(int i, int j);
int checkID(int id);
void deleteproduct();
void addproduct();
int modifyproduct();
void displayproduct();
void previous();
int checkIDmodfify();

struct product {
    int id;
    char name[20];
    int quantity;
    float price;
    float principle;
    int min_quan;
    float total;
};

struct product product[40];
struct bought {
    int id;
    char name[20];
    int quantity;
    float price;
    float principle;
    char timestamp[30];
    int day;
    int time;
};

struct bought buy[30];
struct day {
    int id[100];
    int quantity[100];
    float price[100];
    float total;
};

struct day d[30];
int count = 0;
int countpos = 0;
int dcount = 0;
```

SOURCE CODE :

```
int inventorysystem() {
    int choice;
    count = readfile();
    if(count < 0)
        printf("Cannot locate file\n");
    do {
        printf("-----|\\n");
        printf("-----| INVENTORY PROGRAM |\\n");
        printf("-----|\\n");
        printf("-----| Press 1 to add product |\\n");
        printf("-----| Press 2 to modify product |\\n");
        printf("-----| Press 3 to delete product |\\n");
        printf("-----| Press 4 to display all exiting product |\\n");
        printf("-----| Press 5 to back to main menu |\\n");
        printf("-----| Press 0 to exit the program |\\n");
        printf("-----|\\n");
        printf("\nChoice --> ");
        scanf("%d", &choice);
        switch(choice){
            case 1:
                system("cls");
                displayaddproduct();
                break;
            case 2:
                system("cls");
                inventoryformodifyprod();
                break;
            case 3:
                system("cls");
                showinventoryfordelfun();
                break;
            case 4:
                system("cls");
                displayproduct();
                break;
            case 5:
                system("cls");
                mainmenuselection();
                break;
            case 0:
                system("cls");
                exit(1);
                break;
            default:
                printf("Your choice was wrong please try again");
                break;
        }
    }while(choice != 0);
}
```

2022

Add Product

OUTPUT :

INVENTORY					
NAME	PROD ID	QUANTITY	MINIMUM QUANTITIES	PRICE	PRINCIPLE
aa	1	1	1	20.00	5.00
dd	4	51	1	3.00	1.00
bb	2	24	2	15.00	3.00
cc	3	17	1	56.00	25.00
ll	6	4	1	6.00	3.00

```
ENTER NEW PRODUCTS
Product ID: 5
Product name: ee
Quantity of the product: 14
Price of the product: 2
Principle of the product: 9
Minimum Order Quantity of the product:
```

Added successfully!!

INVENTORY PROGRAM

- Press 1 to add product
- Press 2 to modify product
- Press 3 to delete product
- Press 4 to display all existing product
- Press 5 to back to main menu
- Press 0 to exit the program

Choice --> -

SOURCE CODE :

```
void displayaddproduct() {
    int i;
    count = readFile();
    if(count < 0) {
        puts("cannot open file");
    }
    printf("|-----|-----|-----|-----|-----|-----|-----|\n");
    printf(" |-----|-----|-----|-----|-----|-----|-----|\n");
    printf(" |-----|-----|-----|-----|-----|-----|-----|\n");
    printf(" |-----|-----|-----|-----|-----|-----|-----|\n");
    for(i = 0; i < count; i++) {
        printf(" | %-11s %--12d %--12d %--10d %--13.2f %--11.2f |\n", product[i].name, product[i].id,
product[i].quantity, product[i].min_quan, product[i].price, product[i].principle);
    }
    printf(" |-----|-----|-----|-----|-----|-----|-----|\n");
    addproduct();
}

void addproduct() {
    printf("ENTER NEW PRODUCTS\n");
    readFile();
    if(count > 0) {
        count = readFile();
        IDChecker(0, count);
    }else {
        printf("\nProduct ID Number: ");
        fflush(stdin);
        scanf("%d", &product[count].id);
    }
    printf("Product name: ");scanf("%s",product[count].name);
    printf("Quantity of the product: ");scanf("%d", &product[count].quantity);
    printf("Price of the product: ");scanf("%f", &product[count].price);
    printf("Principle of the product: ");scanf("%f", &product[count].principle);
    printf("Minimum Order Quantity of the product: ");scanf("%d", &product[count].min_quan);
    ++count;
    system("cls");
    printf("Added successfully!!\n");
    writeFile();
}
```

```
int IDChecker(int i, int j) {
    count = readFile();
    printf("Product ID: ");
    fflush(stdin);
    scanf("%d", &product[count].id);
    if(product[i].id == product[j].id) {
        printf("Product ID number is already taken!!\n");
        return IDChecker(i++, j--);
    }
}
```

2022

Modify Product

OUTPUT :

INVENTORY						
NAME	PROD ID	QUANTITY	MINIMUM QUANTITIES	PRICE	PRINCIPLE	
aa	1	1	1	20.00	5.00	
dd	4	51	1	3.00	1.00	
bb	2	24	2	15.00	3.00	
cc	3	17	1	56.00	25.00	
ll	6	4	1	6.00	3.00	

INVENTORY						
NAME	PROD ID	QUANTITY	MINIMUM QUANTITIES	PRICE	PRINCIPLE	
ll	6	4	1	6.00	3.00	

Press 1 to update product ID number
 Press 2 to update product's name
 Press 3 to update product's quantity
 Press 4 to update product's price
 Press 5 to update product's principle
 Press 6 to Update product's minimum quantity
 Press 0 to Back to inventory menu

Enter your choice -->

SOURCE CODE :

```
void inventoryformodifyprod(){
    system("cls");
    int i;
    count = readfile();
    if(count < 0) {
        puts("cannot open file");
    }
    printf("-----");
    printf("-----");
    printf("-----");
    printf("-----");
    printf("-----");
    printf("-----");
    for(i = 0; i < count; i++) {
        printf("%-11s %-12d %-12d %-10d %-13.2f %-11.2f |\n", product[i].name, product[i].id,
product[i].quantity,product[i].min_quan, product[i].price, product[i].principle);
    }
    printf("-----");
    printf("-----");
    modifyproduct();
}
```

```
int modifyproduct() {
    int id;
    int test;
    int i;
    int choice,choice2,choice3;
    printf("EDIT A PRODUCT!");
    printf("\nEnter the id of the product that you want to edit: ");
    fflush(stdin);
    scanf("%d",&id);
    test = checkID(id);
    // printf("\ntest = %d\n",test);
    if(test == 0) {
        system("cls");
        printf("The id num %d is not found.\n", id);
        printf("Press 0 return to inventorymenu or press 1 to try again --> ");
        scanf("%d",&choice2);
        switch (choice2)
        {
        case 0:
            system("cls");
            inventoriesystem();
            break;
        case 1:
            system("cls");
            inventoryformodifyprod();
        default:
            inventoriesystem();
            break;
        }
    }else {
        readfile();
    }
}

int checkID(int idz) {
    int i;
    count = readfile();
    readfile();
    for(i = 0; i < count; i++) {
        if(idz == product[i].id) {
            fclose(f);
            return 1;
            break;
        }else{
            continue;
        }
    }
    fclose(f);
    return 0;
}
```



Modify Product

OUTPUT :

```
INVENTORY
NAME | PROD ID | QUANTITY | MINIMUM QUANTITIES | PRICE | PRINCIPLE
11      6          4           1        6.00       3.00

Press 1 to update product ID number
Press 2 to update product's name
Press 3 to update product's quantity
Press 4 to update product's price
Press 5 to update product's principle
Press 6 to Update product's minimum quantity
Press 0 to Back to inventory menu

Enter your choice -->
```

```
fclose(f);
f = fopen("inventory.txt", "r");
readFile();
writeFile();
fclose(f);
system("cls");
showInveninmodify(id);
printf("Product was updated!!\n");
printf("Press 1 to edit product again or press 0 to return to inventorymenu --> ");
scanf("%d",&choice3);
switch (choice3){
case 1:
    system("cls");
    inventoryformodifyprod();
    break;
case 0:
    system("cls");
    inventorysystem();
    break;
default:
    system("cls");
    inventorysystem();
    break;
}
```

SOURCE CODE :

```
for(i = 0; i < count; i++){
    if(id != product[i].id){
        writeFile();
    }else {
        system("cls");
        showInveninmodify(id);
        printf("\nPress 1 to update product ID number");
        printf("\nPress 2 to update product's name");
        printf("\nPress 3 to update product's quantity");
        printf("\nPress 4 to update product's price");
        printf("\nPress 5 to update product's principle");
        printf("\nPress 6 to Update product's minimum quantity");
        printf("\nEnter your choice --> ");
        fflush(stdin);
        scanf("%d", &choice);
        switch(choice) {
        case 1:
            system("cls");
            showInveninmodify(id);
            printf("Enter new ID: ");
            fflush(stdin);
            scanf("%d",&product[i].id);
            break;
        case 2:
            system("cls");
            showInveninmodify(id);
            printf("Enter new name: ");
            fflush(stdin);
            gets(product[i].name);
            break;
        case 3:
            system("cls");
            showInveninmodify(id);
            printf("Enter new quantity: ");
            scanf("%d", &product[i].quantity);
            break;
        case 4:
            system("cls");
            showInveninmodify(id);
            printf("Enter new price: ");
            scanf("%f", &product[i].price);
            break;
        case 5:
            system("cls");
            showInveninmodify(id);
            printf("Enter new principle: ");
            scanf("%f", &product[i].principle);
            break;
        case 6:
            system("cls");
            showInveninmodify(id);
            printf("Enter the new minimum order quantity: ");
            scanf("%d", &product[i].min_quan);
            break;
        case 0:
            system("cls");
            inventorysystem();
            break;
        default:
            printf("Invalid selection");
            break;
        }
        writeFile();
    }
}
```

2022



CPE100 Project

Update Product ID

OUTPUT :

```
fclose(f);
f = fopen("inventory.txt", "r");
readFile();
writeFile();
fclose(f);
system("cls");
showInveninmodif(id);
printf("Product was updated!!\n");
printf("Press 1 to edit product again or press 0 to return to inventorymenu --> ");
scanf("%d",&choice3);
switch (choice3){
case 1:
    system("cls");
    inventoryformodifyprod();
    break;
case 0:
    system("cls");
    inventorysystem();
    break;
default:
    system("cls");
    inventorysystem();
    break;
}
}
```

SOURCE CODE :

```

for(i = 0; i < count; i++){
    if(id != product[i].id){
        writeFile();
    }else {
        system("cls");
        showInveninmodify(id);
        printf("\nPress 1 to update product ID number");
        printf("\nPress 2 to update product's name");
        printf("\nPress 3 to update product's quantity");
        printf("\nPress 4 to update product's price");
        printf("\nPress 5 to update product's principle");
        printf("\nPress 6 to Update product's minimum quantity");
        printf("\nEnter your choice --> ");
        fflush(stdin);
        scanf("%d", &choice);
        switch(choice) {
            case 1:
                system("cls");
                showInveninmodify(id);
                printf("Enter new ID: ");
                fflush(stdin);
                scanf("%d", &product[i].id);
            break;
            case 2:
                system("cls");
                showInveninmodify(id);
                printf("Enter new name: ");
                fflush(stdin);
                gets(product[i].name);
            break;
            case 3:
                system("cls");
                showInveninmodify(id);
                printf("Enter new quantity: ");
                scanf("%d", &product[i].quantity);
            break;
            case 4:
                system("cls");
                showInveninmodify(id);
                printf("Enter new price: ");
                scanf("%f", &product[i].price);
            break;
            case 5:
                system("cls");
                showInveninmodify(id);
                printf("Enter new principle: ");
                scanf("%f", &product[i].principle);
            break;
            case 6:
                system("cls");
                showInveninmodify(id);
                printf("Enter the new minimum order quantity: ");
                scanf("%d", &product[i].min_quan);
            break;
            case 0:
                system("cls");
                inventorysystem();
            break;
            default:
                printf("Invalid selection");
            break;
        }
        writeFile();
    }
}

```

2022



CPE100 Project

Update Product Name

OUTPUT :

```
fclose(f);
    f = fopen("inventory.txt", "r");
    readFile();
    writeFile();
    fclose(f);
    system("cls");
    showInveninmodify(id);
    printf("Product was updated!!\n");
    printf("Press 1 to edit product again or press 0 to return to inventorymenu --> ");
    scanf("%d",&choice3);
    switch (choice3){
        case 1:
            system("cls");
            inventoryformodifyprod( );
            break;
        case 0:
            system("cls");
            inventorysystem( );
            break;
        default:
            system("cls");
            inventorysystem( );
            break;
    }
}
```

SOURCE CODE :

2022



Update Product Quantity

OUTPUT :

```
fclose(f);
f = fopen("inventory.txt", "r");
readFile();
writeFile();
fclose(f);
system("cls");
showInveninmodify(id);
printf("Product was updated!!\n");
printf("Press 1 to edit product again or press 0 to return to inventorymenu --> ");
scanf("%d",&choice3);
switch (choice3){
case 1:
    system("cls");
    inventoryformodifyprod();
    break;
case 0:
    system("cls");
    inventorysystem();
    break;
default:
    system("cls");
    inventorysystem();
    break;
}
}
```

SOURCE CODE :

```

for(i = 0; i < count; i++){
    if(id != product[i].id){
        writeFile();
    }else {
        system("cls");
        showInveninmodify(id);
        printf("\nPress 1 to update product ID number");
        printf("\nPress 2 to update product's name");
        printf("\nPress 3 to update product's quantity");
        printf("\nPress 4 to update product's price");
        printf("\nPress 5 to update product's principle");
        printf("\nPress 6 to Update product's minimum quantity");
        printf("\nEnter your choice --> ");
        fflush(stdin);
        scanf("%d", &choice);
        switch(choice) {
            case 1:
                system("cls");
                showInveninmodify(id);
                printf("Enter new ID: ");
                fflush(stdin);
                scanf("%d", &product[i].id);
            break;
            case 2:
                system("cls");
                showInveninmodify(id);
                printf("Enter new name: ");
                fflush(stdin);
                gets(product[i].name);
            break;
            case 3:
                system("cls");
                showInveninmodify(id);
                printf("Enter new quantity: ");
                scanf("%d", &product[i].quantity);
            break;
            case 4:
                system("cls");
                showInveninmodify(id);
                printf("Enter new price: ");
                scanf("%f", &product[i].price);
            break;
            case 5:
                system("cls");
                showInveninmodify(id);
                printf("Enter new principle: ");
                scanf("%f", &product[i].principle);
            break;
            case 6:
                system("cls");
                showInveninmodify(id);
                printf("Enter the new minimum order quantity: ");
                scanf("%d", &product[i].min_quan);
            break;
            case 0:
                system("cls");
                inventorysystem();
            break;
            default:
                printf("Invalid selection");
            break;
        }
        writeFile();
    }
}

```

2022



Update Product Price

OUTPUT :

```
INVENTORY
NAME | PROD ID | QUANTITY | MINIMUM QUANTITIES | PRICE | PRINCIPLE
kk      6          20           1        20.00       5.00

Enter new price: 25_
INVENTORY
NAME | PROD ID | QUANTITY | MINIMUM QUANTITIES | PRICE | PRINCIPLE
kk      6          20           1        25.00       5.00

Product was updated!
Press 1 to edit product again or press 0 to return to inventorymenu --> _
```

```
fclose(f);
f = fopen("inventory.txt", "r");
readFile();
writeFile();
fclose(f);
system("cls");
showInveninmodify(id);
printf("Product was updated!!\n");
printf("Press 1 to edit product again or press 0 to return to inventorymenu --> ");
scanf("%d",&choice3);
switch (choice3){
case 1:
    system("cls");
    inventoryformodifyprod();
    break;
case 0:
    system("cls");
    inventorysystem();
    break;
default:
    system("cls");
    inventorysystem();
    break;
}
```

SOURCE CODE :

```
for(i = 0; i < count; i++){
    if(id != product[i].id){
        writeFile();
    }else {
        system("cls");
        showInveninmodify(id);
        printf("\nPress 1 to update product ID number");
        printf("\nPress 2 to update product's name");
        printf("\nPress 3 to update product's quantity");
        printf("\nPress 4 to update product's price");
        printf("\nPress 5 to update product's principle");
        printf("\nPress 6 to Update product's minimum quantity");
        printf("\nEnter your choice --> ");
        fflush(stdin);
        scanf("%d", &choice);
        switch(choice) {
        case 1:
            system("cls");
            showInveninmodify(id);
            printf("Enter new ID: ");
            fflush(stdin);
            scanf("%d",&product[i].id);
            break;
        case 2:
            system("cls");
            showInveninmodify(id);
            printf("Enter new name: ");
            fflush(stdin);
            gets(product[i].name);
            break;
        case 3:
            system("cls");
            showInveninmodify(id);
            printf("Enter new quantity: ");
            scanf("%d", &product[i].quantity);
            break;
        case 4:
            system("cls");
            showInveninmodify(id);
            printf("Enter new price: ");
            scanf("%f", &product[i].price);
            break;
        case 5:
            system("cls");
            showInveninmodify(id);
            printf("Enter new principle: ");
            scanf("%f", &product[i].principle);
            break;
        case 6:
            system("cls");
            showInveninmodify(id);
            printf("Enter the new minimum order quantity: ");
            scanf("%d", &product[i].min_quan);
            break;
        case 0:
            system("cls");
            inventorysystem();
            break;
        default:
            printf("Invalid selection");
            break;
        }
        writeFile();
    }
}
```

2022



Update Product Principle

OUTPUT :

```
fclose(f);
    f = fopen("inventory.txt", "r");
    readFile();
    writeFile();
    fclose(f);
    system("cls");
    showInveninmodif(id);
    printf("Product was updated!!\n");
    printf("Press 1 to edit product again or press 0 to return to inventorymenu --> ");
    scanf("%d",&choice3);
    switch (choice3){
        case 1:
            system("cls");
            inventoryformodifyprod();
            break;
        case 0:
            system("cls");
            inventorysystem();
            break;
        default:
            system("cls");
            inventorysystem();
            break;
    }
}
```

SOURCE CODE :

2022



Update Product Minimum Quantity

OUTPUT :

```
INVENTORY
+-----+
| NAME | PROD ID | QUANTITY | MINIMUM QUANTITIES | PRICE | PRINCIPLE |
+-----+
| kk   |       6  |      20    |           1        | 25.00 |     6.00  |
+-----+
Enter the new minimum order quantity: 2
INVENTORY
+-----+
| NAME | PROD ID | QUANTITY | MINIMUM QUANTITIES | PRICE | PRINCIPLE |
+-----+
| kk   |       6  |      20    |           2        | 25.00 |     6.00  |
+-----+
Product was updated!!
Press 1 to edit product again or press 0 to return to inventorymenu -->
```

```
fclose(f);
f = fopen("inventory.txt", "r");
readFile();
writeFile();
fclose(f);
system("cls");
showInveninmodify(id);
printf("Product was updated!!\n");
printf("Press 1 to edit product again or press 0 to return to inventorymenu --> ");
scanf("%d",&choice3);
switch (choice3){
case 1:
    system("cls");
    inventoryformodifyprod();
    break;
case 0:
    system("cls");
    inventorysystem();
    break;
default:
    system("cls");
    inventorysystem();
    break;
}
```

SOURCE CODE :

```
for(i = 0; i < count; i++){
    if(id != product[i].id){
        writeFile();
    }else {
        system("cls");
        showInveninmodify(id);
        printf("\nPress 1 to update product ID number");
        printf("\nPress 2 to update product's name");
        printf("\nPress 3 to update product's quantity");
        printf("\nPress 4 to update product's price");
        printf("\nPress 5 to update product's principle");
        printf("\nPress 6 to Update product's minimum quantity");
        printf("\nEnter your choice --> ");
        fflush(stdin);
        scanf("%d", &choice);
        switch(choice) {
        case 1:
            system("cls");
            showInveninmodify(id);
            printf("Enter new ID: ");
            fflush(stdin);
            scanf("%d",&product[i].id);
            break;
        case 2:
            system("cls");
            showInveninmodify(id);
            printf("Enter new name: ");
            fflush(stdin);
            gets(product[i].name);
            break;
        case 3:
            system("cls");
            showInveninmodify(id);
            printf("Enter new quantity: ");
            scanf("%d", &product[i].quantity);
            break;
        case 4:
            system("cls");
            showInveninmodify(id);
            printf("Enter new price: ");
            scanf("%f", &product[i].price);
            break;
        case 5:
            system("cls");
            showInveninmodify(id);
            printf("Enter new principle: ");
            scanf("%f", &product[i].principle);
            break;
        case 6:
            system("cls");
            showInveninmodify(id);
            printf("Enter the new minimum order quantity: ");
            scanf("%d", &product[i].min_quan);
            break;
        case 0:
            system("cls");
            inventorysystem();
            break;
        default:
            printf("Invalid selection");
            break;
        }
        writeFile();
    }
```

2022

Delete Product

OUTPUT :

```

INVENTORY
+-----+-----+-----+-----+-----+-----+
| NAME | PROD ID | QUANTITY | MINIMUM QUANTITIES | PRICE | PRINCIPLE |
+-----+-----+-----+-----+-----+-----+
| aa   |      1   |      1   |      1   | 20.00 |      5.00 |
| dd   |      4   |     51   |      1   |  3.00 |      1.00 |
| bb   |      2   |     24   |      2   | 15.00 |      3.00 |
| cc   |      3   |     17   |      1   | 56.00 |     25.00 |
| ll   |      7   |      5   |      1   |  6.00 |      3.00 |
+-----+-----+-----+-----+-----+-----+
Delete successfully
+-----+-----+-----+-----+-----+-----+
|           INVENTORY PROGRAM           |
+-----+-----+-----+-----+-----+-----+
| Press 1 to add product              |
| Press 2 to modify product           |
| Press 3 to delete product           |
| Press 4 to display all existing product |
| Press 5 to back to main menu        |
| Press 0 to exit the program         |
+-----+-----+-----+-----+-----+-----+
Choice -->

```

SOURCE CODE :

```

void deleteproduct() {
    count = readFile();
    int id;
    int i, j, choice;
    int z = false;
    printf("Enter the product ID that you want to delete --> ");
    fflush(stdin);
    scanf("%d", &id);
    for(i = 0; i < count; i++) {
        if(product[i].id == id){
            z = true;
            for(j = i; j < (count - 1); j++) {
                product[j] = product[j + 1];
            }
            count--;
            printf("Delete successfully\n");
        }
    }
    if(z == false) {
        printf("Cant find product id: %d.\n", id);
        printf("Press 1 to try again / Press 0 to go back to inventory page --> ");
        scanf("%d", &choice);
        switch(choice){
            case 1:
                system("cls");
                showinventoryfordelfun();
                break;
            case 0:
                system("cls");
                inventorysystem();
                break;
            default:
                system("cls");
                inventorysystem();
                break;
        }
    }
    writeFile();
}

void showinventoryfordelfun(){
    system("cls");
    int i;
    count = readFile();
    if(count < 0) {
        puts("cannot open file");
    }
    printf("-----|-----|-----|-----|-----|-----|\n");
    printf("-----|-----|-----|-----|-----|-----|\n");
    printf("-----|-----|-----|-----|-----|-----|\n");
    printf("-----|-----|-----|-----|-----|-----|\n");
    for(i = 0; i < count; i++) {
        printf("| %11s %12d %12d %10d %13.2f %11.2f |\n", product[i].name, product[i].id,
        product[i].quantity, product[i].min_quan, product[i].price, product[i].principle);
    }
    printf("-----|-----|-----|-----|-----|-----|\n");
    deleteproduct();
}

```

2022

Display Product

OUTPUT :

INVENTORY					
NAME	PROD ID	QUANTITY	MINIMUM QUANTITIES	PRICE	PRINCIPLE
aa	1	1	1	20.00	5.00
dd	4	51	1	3.00	1.00
bb	2	24	2	15.00	3.00
cc	3	17	1	56.00	25.00

Press 0 to back to inventory --> .

SOURCE CODE :

```
○○○
void displayproduct() {
    int i;
    count = readfile();
    if(count < 0) {
        puts("cannot open file");
    }
    printf("|-----|\n");
    printf("|-----|\n");
    printf("|-----|\n");
    printf(" | NAME | PROD ID | QUANTITY | MINIMUM QUANTITIES | PRICE |\n");
    printf(" |-----|\n");
    for(i = 0; i < count; i++) {
        printf(" | %-11s %--12d %--12d %--10d %--13.2f %--11.2f |\n",
product[i].name, product[i].id, product[i].quantity, product[i].min_quan, product[i].price,
product[i].principle);
    }
    printf(" |-----|\n");
    printf(" |-----| \n\n\n");
    previous();
}
```

2022



Accounting System

OUTPUT:

Bought List						
NAME	PROD ID	QUANTITY	PRICE	PRINCIPLE	TIME	
aa	1	1	20.00	5.00	Thu Dec 15 11:50:20 2022	
dd	4	1	3.00	1.00	Thu Dec 15 11:50:28 2022	
aa	1	1	20.00	5.00	Thu Dec 15 12:53:08 2022	
aa	1	1	20.00	5.00	Thu Dec 15 12:53:46 2022	
dd	4	1	3.00	1.00	Thu Dec 22 15:24:34 2022	
dd	4	2	6.00	2.00	Thu Dec 22 15:27:31 2022	
aa	1	1	20.00	5.00	Fri Dec 23 15:29:55 2022	
dd	4	1	3.00	1.00	Fri Dec 23 15:30:07 2022	
dd	4	1	3.00	1.00	Sat Dec 24 15:30:36 2022	
aa	1	1	20.00	5.00	Sat Dec 24 15:30:48 2022	
bb	2	3	45.00	9.00	Fri Dec 16 15:33:32 2022	
cc	3	1	56.00	25.00	Fri Dec 16 15:33:43 2022	
cc	3	1	56.00	25.00	Sat Dec 17 15:34:54 2022	
dd	4	1	3.00	1.00	Sun Dec 18 15:35:53 2022	
aa	1	1	20.00	5.00	Sun Dec 18 15:36:11 2022	
bb	2	3	45.00	9.00	Sun Dec 18 15:36:21 2022	
TOTAL : 343.00				PROFIT : 238.00		

SOURCE CODE:

Daily Summary (Admin)

OUTPUT:

DAY SUMMARY (COUNT BY FIRST PRODUCT)						
DAY	NAME	PROD ID	QUANTITY	PRICE	PRINCIPLE	TIME
1	aa	1	1	20.00	5.00	Thu Dec 15 11:50:20 2022
1	dd	4	1	3.00	1.00	Thu Dec 15 11:50:28 2022
1	aa	1	1	20.00	5.00	Thu Dec 15 12:53:08 2022
1	aa	1	1	20.00	5.00	Thu Dec 15 12:53:46 2022

TOTAL OF DAY 1 = 63.00	PROFIT OF DAY 1 = 47.00
------------------------	-------------------------

DAY	NAME	PROD ID	QUANTITY	PRICE	PRINCIPLE	TIME
2	bb	2	3	45.00	9.00	Fri Dec 16 15:33:32 2022
2	cc	3	1	56.00	25.00	Fri Dec 16 15:33:43 2022

TOTAL OF DAY 2 = 101.00	PROFIT OF DAY 2 = 67.00
-------------------------	-------------------------

SOURCE CODE :

```

○ ○ ○

void admindaily() {
    int i = 0, flag = 0, j, choice;
    float total, totalarry[1000];
    float princsum, profit[1000];
    countpos = readposFile();
    int temp = countpos;
    if(countpos < 0) {
        puts("Cannot open file");
    }
    printf("|\n-----|\n");
    printf("|\n-----|\n");
    while(i >= 0) {
        total = 0;
        princsum = 0;
        printf("|\n-----|\n");
        printf("|\n-----|\n");
        printf("|\n-----|\n");
        printf("|\n-----|\n");
        printf("|\n-----|\n");
        for(j = 0; j < countpos; j++) {
            if((buy[j].time - buy[0].time) >= i * 86400 && (buy[j].time - buy[0].time) < (i + 1) *
86400) {
                total = total + (buy[j].price * buy[j].quantity);
                princsum = princsum + (buy[j].principle * buy[j].quantity);
                printf("|\n%-4d %-14s %-11d %-4d %-11.2f %-11.2f %-12s
|\n", i + 1, buy[j].name, buy[j].id, buy[j].quantity, buy[j].price * buy[j].quantity, buy[j].principle*
buy[j].quantity, buy[j].timestamp);
                temp--;
                flag++;
            }
        }
        profit[i] = total - princsum;
        totalarry[i] = total;
        printf("|\n-----|\n");
        printf("|\n-----|\n");
        printf("|\n-----|\n");
        printf("|\n-----|\n");
        printf("|\n-----|\n");
        if(flag == 0) {
            printf("|\n-----|\n");
            printf("|\n-----|\n");
            printf("|\n-----|\n");
            printf("|\n-----|\n");
            printf("|\n-----|\n");
            printf("Data not found!!\n");
        }
        if(temp == 0) {
            break;
        }
        i++;
    }
}

```

2022

Weekly Summary (Admin)

OUTPUT:

WEEK SUMMARY (COUNT BY FIRST PRODUCT)						
WEEK	NAME	PROD ID	QUANTITY	PRICE	PRINCIPLE	TIME
1	aa	1	1	20.00	5.00	Thu Dec 15 11:50:20 2022
1	dd	4	1	3.00	1.00	Thu Dec 15 11:50:28 2022
1	aa	1	1	20.00	5.00	Thu Dec 15 12:53:08 2022
1	aa	1	1	20.00	5.00	Thu Dec 15 12:53:46 2022
1	bb	2	3	45.00	9.00	Fri Dec 16 15:33:32 2022
1	cc	3	1	56.00	25.00	Fri Dec 16 15:33:43 2022
1	cc	3	1	56.00	25.00	Sat Dec 17 15:34:54 2022
1	dd	4	1	3.00	1.00	Sun Dec 18 15:35:53 2022
1	aa	1	1	20.00	5.00	Sun Dec 18 15:36:11 2022
1	bb	2	3	45.00	9.00	Sun Dec 18 15:36:21 2022
TOTAL OF WEEK 1 = 288.00			PROFIT OF WEEK 1 = 198.00			
WEEK SUMMARY (COUNT BY FIRST PRODUCT)						
WEEK	NAME	PROD ID	QUANTITY	PRICE	PRINCIPLE	TIME
2	dd	4	1	3.00	1.00	Thu Dec 22 15:24:34 2022
2	dd	4	2	6.00	2.00	Thu Dec 22 15:27:31 2022
2	aa	1	1	20.00	5.00	Fri Dec 23 15:29:55 2022
2	dd	4	1	3.00	1.00	Fri Dec 23 15:30:07 2022
2	dd	4	1	3.00	1.00	Sat Dec 24 15:30:36 2022
2	aa	1	1	20.00	5.00	Sat Dec 24 15:30:48 2022
TOTAL OF WEEK 2 = 55.00			PROFIT OF WEEK 2 = 40.00			

SOURCE CODE :

```

void adminweekly() {
    int i = 0, flag = 0, choice;
    float total, totalarry[1000];
    float princsum, profit[1000];
    countpos = readposFile();
    int temp = countpos;
    if(countpos < 0) {
        puts("Cannot open file");
    }
    printf("|-----|\n");
    printf("|-----|\n");
    while(i >= 0) {
        total = 0;
        princsum = 0;
        printf("|-----|\n");
        printf(" | WEEK | NAME | PROD ID | QUANTITY | PRICE | PRINCIPLE |\n");
        printf(" |-----|\n");
        for(j = 0; j < countpos; j++) {
            if((buy[j].time - buy[0].time) >= i * 604800 && (buy[j].time - buy[0].time) < (i + 1) *
604800) {
                total = total + (buy[j].price * buy[j].quantity);
                princsum = princsum + (buy[j].principle * buy[j].quantity);
                printf(" | %4d %-14s %-11d %-4d %-11.2f %-11.2f %-12s
| \n", i + 1, buy[j].name, buy[j].id, buy[j].quantity, buy[j].price * buy[j].quantity, buy[j].principle*
buy[j].quantity, buy[j].timestamp);
                temp--;
                flag++;
            }
        }
        profit[i] = total - princsum;
        totalarry[i] = total;
        printf(" |-----|\n");
        printf(" |-----|\n");
        printf(" |-----| TOTAL OF WEEK %d = %-8.2f | PROFIT OF WEEK
%d = %-8.2f | \n", i + 1, totalarry[i], i+1, profit[i]);
        printf(" |-----|\n");
        if(flag == 0) {
            printf(" |-----|\n");
            Data not found!!
        }
        if(temp == 0) {
            break;
        }
        i++;
    }
}

```

2022

THANK YOU!!!

