

A menu-driven C program for a mart billing system handles customer invoices, including the process of generating invoices, saving them to a binary file, and retrieving them based on customer names.

Functionalities:

1. A menu screen displaying the operations.
2. Generating new invoices.
3. View all the invoices.
4. View the invoice of a particular customer based on the customer's name.
5. Handling Invalid Inputs

Source Code:

```
#include<stdio.h>

#include<time.h>

#include<string.h>

#include<stdlib.h>


struct items {

    char item[20];

    float price;

    int qty;

};


struct orders {

    char customer[50];

    char date[50];

    int numOfItems;

    struct items itm[50];

};
```

```
// Functions to generate bills
```

```
void generateBillHeader(char name[50], char date[30]) {  
    printf("\n\n");  
    printf("\t FRIENDS MART");  
    printf("\n\t -----");  
    printf("\nDate: %s", date);  
    printf("\nInvoice To: %s", name);  
    printf("\n");  
    printf("-----\n");  
    printf("Items\t\t\t");  
    printf("Qty\t\t\t");  
    printf("Total\t\t\t");  
    printf("\n-----");  
    printf("\n\n");  
}
```

```
void generateBillBody(char item[20], int qty, float price) {  
    printf("%s\t\t\t", item);  
    printf("%d\t\t\t", qty);  
    printf("%.2f\t\t\t", qty * price);  
    printf("\n");  
}
```

```
void generateBillFooter(float total) {  
    printf("\n");  
    float dis = 0.1 * total;  
    float netTotal = total - dis;  
    float cgst = 0.09 * netTotal, grandTotal = netTotal + 2 * cgst; // netTotal + cgst + sgst  
    printf("-----\n");  
    printf("Sub Total\t\t\t%.2f", total);  
    printf("\nDiscount @10%%\t\t\t%.2f", dis);  
    printf("\n\t\t\t-----");  
    printf("\nNet Total\t\t\t%.2f", netTotal);  
    printf("\nCGST @9%%\t\t\t%.2f", cgst);
```

```

printf("\nSGST @9%%\t\t%.2f", cgst);
printf("\n-----");
printf("\nGrand Total\t\t%.2f", grandTotal);
printf("\n-----\n");
}

int main() {
    int opt, n, i;
    struct orders ord;
    struct orders order;
    char saveBill = 'y', contFlag = 'y';
    char name[50];
    FILE *fp;
    // Dashboard
    while (contFlag == 'y') {
        system("cls");
        float total = 0;
        int invoiceFound = 0;
        printf("\t=====FRIENDS MART=====");
        printf("\n\nPlease select your preferred operation");
        printf("\n\n1.Generate Invoice");
        printf("\n\n2.Show all Invoices");
        printf("\n\n3.Search Invoice");
        printf("\n\n4.Exit");
        printf("\n\nYour choice:\t");
        fflush(stdin);
        scanf("%d", &opt);
        fgetc(stdin);
        switch (opt) {
            case 1:
                system("cls");
                printf("\nPlease enter the name of the customer:\t");
                fgets(ord.customer, 50, stdin); //getting name
                ord.customer[strlen(ord.customer) - 1] = '\0';

```

```

time_t t=time(NULL);

struct tm date=*localtime(&t);

int yr=date.tm_year+1900;

int mon=date.tm_mon+1;

int d=date.tm_mday;

char year[10];

char month[10];

char day[10];

char fulldate[20];

snprintf(year, sizeof(year), "%d", yr);

snprintf(month, sizeof(month), "%d/", mon);

snprintf(day, sizeof(day), "%d/", d);

strcpy(fulldate, day);

strcat(fulldate, month);

strcat(fulldate, year);

strcpy(ord.date, fulldate); //getting current date

printf("\nPlease enter the number of items:\t");

fflush(stdin);

scanf("%d", &n);

ord.numOfItems = n;

for (i = 0; i < n; i++) {

//      fgetc(stdin);

      fflush(stdin);

      printf("\n\n");

      printf("Please enter the item %d name:\t", i + 1);

      fgets(ord.itm[i].item, 50, stdin);

      ord.itm[i].item[strlen(ord.itm[i].item) - 1] = '\0';

      printf("Please enter the quantity:\t");

      fflush(stdin);

      scanf("%d", &ord.itm[i].qty);

      printf("Please enter the unit price:\t");

      fflush(stdin);

      scanf("%f", &ord.itm[i].price);

      total += ord.itm[i].qty * ord.itm[i].price;

```

```

}

generateBillHeader(ord.customer, ord.date);
for (i = 0; i < ord.numOfItems; i++) {
    generateBillBody(ord.itm[i].item, ord.itm[i].qty, ord.itm[i].price);
}
generateBillFooter(total);

printf("\nDo you want to save the invoice [y/n]:\t");
fflush(stdin);

scanf("%s", &saveBill);
while(saveBill)
{
    if (saveBill == 'y' || saveBill=='Y') {
        fp = fopen("MartBill.bin", "ab");
        if(fp==NULL){
            printf("Error Accessing File:\n");
            break;
        }
        else{
            int flag=0;
            flag=fwrite(&ord, sizeof(struct orders), 1, fp);
            if (flag)
                printf("\nSuccessfully saved");
            else
                printf("\nError saving");
            fclose(fp);
            break;
        }
    }
    else if(saveBill == 'n' || saveBill=='N'){
        break;
    }
    else{

```

```

printf("\nINVALID INPUT(please enter again)");

printf("\nDo you want to save the invoice [y/n]:\t");

fflush(stdin);

scanf("%s", &saveBill);

    }

}

```

case 2:

```

system("cls");

fp = fopen("MartBill.bin", "rb");

printf("\n *****Your Previous Invoices*****\n");

while (fread(&order, sizeof(struct orders), 1, fp)) {

    float tot = 0;

    generateBillHeader(order.customer, order.date);

    for (i = 0; i < order.numOfItems; i++) {

        generateBillBody(order.itm[i].item, order.itm[i].qty, order.itm[i].price);

        tot += order.itm[i].qty * order.itm[i].price;

    }

    generateBillFooter(tot);

}

fclose(fp);

break;

```

case 3:

```

printf("Enter the name of the customer:\t");

fgets(name, 50, stdin);

name[strlen(name) - 1] = 0;

system("cls");

fp = fopen("MartBill.bin", "rb");

printf("\t*****Invoice of %s*****", name);

while (fread(&order, sizeof(struct orders), 1, fp)) {

    float tot = 0;

    if (!strcmp(order.customer, name)) {

```

```

        generateBillHeader(order.customer, order.date);

        for (i = 0; i < order.numOfItems; i++) {
            generateBillBody(order.itm[i].item, order.itm[i].qty, order.itm[i].price);
            tot += order.itm[i].qty * order.itm[i].price;
        }

        generateBillFooter(tot);

        invoiceFound = 1;
    }
}

if (!invoiceFound) {
    printf("\n\nSorry, the invoice for %s does not exist", name);
}

fclose(fp);

break;

case 4:

    printf("\n\t\t Bye Bye :)\n\n");
    exit(0);
    break;

default:

    printf("Sorry invalid option");
    break;
}

printf("\nDo you want to perform another operation?[y/n]:\t");
fflush(stdin);

scanf("%s", &contFlag);
}

printf("\n\t\t Bye Bye :)\n\n");

return 0;
}

```

OUTPUTS:

```
C:\Users\91939\Desktop\1st-MCA\c\mart.exe
=====FRIENDS MART=====

Please select your preferred operation

1.Generate Invoice
2.Show all Invoices
3.Search Invoice
4.Exit

Your choice: 1
```

1. Menu Page

```
C:\Users\91939\Desktop\1st-MCA\c\mart.exe

Please enter the name of the customer: Vinay
Please enter the number of items: 2

Please enter the item 1 name: Paste
Please enter the quantity: 1
Please enter the unit price: 50

Please enter the item 2 name: Brush
Please enter the quantity: 2
Please enter the unit price: 30

      FRIENDS MART
      -----
Date: 30/9/2023
Invoice To: Vinay
-----
Items                Qty                Total
-----
Paste                 1                 50.00
Brush                 2                 60.00
-----
Sub Total             110.00
Discount @10%         11.00
```

2. Invoice Generation Page


```
C:\Users\91939\Desktop\1st-MCA\c\mart.exe

FRIENDS MART
-----
Date: 30/9/2023
Invoice To: Vinay
-----
Items                Qty                Total
-----
Paste                1                50.00
Brush                2                60.00
-----
Sub Total            110.00
Discount @10%        11.00
-----
Net Total            99.00
CGST @9%              8.91
SGST @9%              8.91
-----
Grand Total          116.82
-----
Do you want to save the invoice [y/n]: y_
```

3. Invoice is generated and waiting for confirmation to save

```
C:\Users\91939\Desktop\1st-MCA\c\mart.exe

CGST @9%              4.05
SGST @9%              4.05
-----
Grand Total          53.10
-----

FRIENDS MART
-----
Date: 30/9/2023
Invoice To: Vinay
-----
Items                Qty                Total
-----
Paste                1                50.00
Brush                2                60.00
-----
Sub Total            110.00
Discount @10%        11.00
-----
Net Total            99.00
CGST @9%              8.91
SGST @9%              8.91
-----
Grand Total          116.82
-----
Do you want to perform another operation?[y/n]: y_
```

4. Invoice is saved and waiting for the response to perform any other operation

```
C:\Users\91939\Desktop\1st-MCA\c\mart.exe

****Your Previous Invoices****

      FRIENDS MART
-----
Date: 30/9/2023
Invoice To: Vinay
-----
Items                Qty                Total
-----
Paste                1                50.00
Brush                2                60.00
-----
Sub Total            110.00
Discount @10%        11.00
-----
Net Total            99.00
CGST @9%              8.91
SGST @9%              8.91
-----
Grand Total          116.82
-----

      FRIENDS MART
-----
Date: 30/9/2023
```

```
C:\Users\91939\Desktop\1st-MCA\c\mart.exe

      FRIENDS MART
-----
Date: 30/9/2023
Invoice To: Pratham
-----
Items                Qty                Total
-----
Paste                1                50.00
-----
Sub Total            50.00
Discount @10%        5.00
-----
Net Total            45.00
CGST @9%              4.05
SGST @9%              4.05
-----
Grand Total          53.10
-----

      FRIENDS MART
-----
Date: 30/9/2023
Invoice To: Alankritha
-----
Items                Qty                Total
-----
```

```
C:\Users\91939\Desktop\1st-MCA\c\mart.exe

      FRIENDS MART
-----
Date: 30/9/2023
Invoice To: Vinay
-----
Items                Qty                Total
-----
Paste                 1                50.00
Brush                 2                60.00
-----
Sub Total              110.00
Discount @10%          11.00
-----
Net Total              99.00
CGST @9%                8.91
SGST @9%                8.91
-----
Grand Total            116.82
-----
Do you want to perform another operation?[y/n]: _
```

5. Viewing all invoices

```
C:\Users\91939\Desktop\1st-MCA\c\mart.exe

=====FRIENDS MART=====

Please select your preferred operation

1.Generate Invoice
2.Show all Invoices
3.Search Invoice
4.Exit

Your choice: 3
Enter the name of the customer: _
```

6. Searching invoices of customer

```
C:\Users\91939\Desktop\1st-MCA\c\mart.exe

*****Invoice of Vinay*****

      FRIENDS MART
-----
Date: 30/9/2023
Invoice To: Vinay
-----
Items                Qty                Total
-----
PAste                1                50.00
Brush                2                60.00
-----
Sub Total            110.00
Discount @10%        11.00
-----
Net Total            99.00
CGST @9%             8.91
SGST @9%             8.91
-----
Grand Total          116.82
-----

      FRIENDS MART
-----
Date: 30/9/2023
Invoice To: Vinay
-----
```

```
C:\Users\91939\Desktop\1st-MCA\c\mart.exe

      FRIENDS MART
-----
Date: 30/9/2023
Invoice To: Vinay
-----
Items                Qty                Total
-----
Paste                1                50.00
Brush                2                60.00
-----
Sub Total            110.00
Discount @10%        11.00
-----
Net Total            99.00
CGST @9%             8.91
SGST @9%             8.91
-----
Grand Total          116.82
-----

Do you want to perform another operation?[y/n]: _
```

7. All invoices of the customer

```
C:\Users\91939\Desktop\1st-MCA\c\mart.exe
=====FRIENDS MART=====

Please select your preferred operation

1.Generate Invoice
2.Show all Invoices
3.Search Invoice
4.Exit

Your choice: 5
Sorry invalid option
Do you want to perform another operation?[y/n]:
```

8. Entering an invalid input

```
C:\Users\91939\Desktop\1st-MCA\c\mart.exe
=====FRIENDS MART=====

Please select your preferred operation

1.Generate Invoice
2.Show all Invoices
3.Search Invoice
4.Exit

Your choice: 4

Bye Bye :)

-----
Process exited after 213.7 seconds with return value 0
Press any key to continue . . .
```

9. Exiting the program