

SHOPPING MALL MANAGEMENT SYSTEM



**SUBMITTED AS A PART OF C.B.S.E. CURRICULUM
FOR THE YEAR 2008-09**

***SUBMITTED TO :-
Ms. SUNITA***

SUBMITTED BY :-

***XII A
ROLL NO. -***

CONTENTS

- CERTIFICATE
- ACKNOWLEDGEMENT
- PROJECT PREAMBLE
- PROJECT STUDY
- ALGORITHM
- SOURCE CODE
- OUTPUT

CERTIFICATE

This is to certify that

_____, **Roll No. :**

&

_____, **Roll No. :**

of Class XII A, K.V. No. 2, Delhi Cantt,

have worked on the project titled

SHOPING MALL MANAGEMENT SYSTEM

and have completed the project to my satisfaction.

Teacher's Signature

.....

Student's Signature

.....

ACKNOWLEDGEMENT

It gives me immense pleasure in expressing my deep sense of gratitude to my respected teacher,

Ms. Sunita for her inspiring guidance and thought in the preparation of the project.

This project would not have been completed without her valuable guidance and her tremendous effort.

PROJECT PREAMBLE

The project titled “**Shopping Mall Management System**”

has been developed as per the requirement of
CBSE for subject Computer Science (083) for
AISSCE – 2009. The source code has been
developed in C++. Compiler is Turbo C++.

The project deals with creation of records,
inserting data, deleting, searching and
printing of stored records. It also
includes validations.

PROJECT STUDY

CLASSES & FUNCTIONS USED IN THE PROJECT :

```
/**
 * CLASS NAME : menu
 * DETAILS : IT CONTROLS OVER ALL THE FUNCTIONS
 */
```

```
class menu
{
    public :
        void main_menu(void) ;
    private :
        void edit_menu(void) ;
};
```

```
/**
 * CLASS NAME : item
 * DETAILS : IT CONTROLS OVER ALL THE FUNCTIONS
 * RELATED TO ITEMS
 */
```

```
class item
{
    public :
        void add_item(void) ;
        void delete_item(void) ;
        void modify_item(void) ;
        void list_of_item(void) ;
        void purchase(void) ;
    private :
        int last_code(void) ;
        void delete_record(int) ;
        void modify_record(int) ;
        void display_record(int) ;
        int item_found(int) ;
        int recordno(int) ;
        int itemcode ;
        char itemname[30] ;
        float itemprice ;
};
```

```
/**
 * CLASS NAME : account
 * DETAILS : IT CONTROLS OVER ALL THE FUNCTIONS
 * RELATED TO MAKING BILL
 */
```

```
class account
{
    public :
        void bill_list(void) ;
        void prepare_bill(int) ;
        int last_billno(void) ;
};
```

```
void add_bill(int, int t_itemcode, char *t_itemname
              , float t_qty, float t_price) ;
private :
int  code, billno, length ;
int  dd, mm, yy ;
float price, quantity ;
char name[30] ;
};
```

ALGORITHM

FOR INSERTING A RECORD TO A FILE:

- ✓ Open the file in append mode.
- ✓ Input the information for the record from the user.
- ✓ Using 'write' function available in file 'fstream.h' write the record on the file.
- ✓ Close the file.

FOR DELETION OF A RECORD FROM A FILE:

- ✓ Open the master file i.e. the file that stores all the records, in read mode.
- ✓ Open the transaction file i.e. a temporary file created for transaction, in write mode.
- ✓ Input the information for the record to be deleted.
- ✓ Read the records from the master file till you reach the end of file. Compare each record with the record to be deleted. If the two records are found different, write the original record to the transaction file.
- ✓ Close both master as well as transaction file.
- ✓ Delete the master file using function 'unlink'.
- ✓ Rename the transaction file as the master file.

FOR MODIFICATION OF A RECORD IN A FILE:

- ✓ Open the master file i.e. the file that stores all the records, in read mode.
- ✓ Open the transaction file i.e. a temporary file created for transaction, in write mode.
- ✓ Input the record to be modified from the user.
- ✓ Read the records from the master file till you reach the end of file. Compare each record with the record to be modified. If the two records are found different, write the original record to the transaction file & if the records are same write the 'new' (updated) record to the transaction file.
- ✓ Close both master as well as transaction file.
- ✓ Delete the master file using function 'unlink'.
- ✓ Rename the transaction file as the master file.

FOR SEARCHING A RECORD IN A FILE:

- ✓ Open the file in read mode in which record is to be searched.
- ✓ Input the information required for comparing the distinct records.
- ✓ Start reading the records from the file till end of the file is not encountered. Compare each record, if the record matches the display the record, if value is not matched at last display 'not found'.
- ✓ Close the file.

SOURCE CODE

```

//*****
//          SHOPPING MALL MANAGEMENT SYSTEM
//*****

//*****
//   INCLUDED HEADER FILES
//*****

#include <fstream.h>
#include <process.h>
#include <string.h>
#include <stdlib.h>
#include <stdio.h>
#include <ctype.h>
#include <conio.h>
#include <dos.h>

//*****
//   CLASS NAME : menu
//   DETAILS   : IT CONTROLS OVER ALL THE FUNCTIONS
//*****

class menu
{
    public :
        void main_menu(void) ;
    private :
        void edit_menu(void) ;
};

//*****
//   CLASS NAME : item
//   DETAILS   : IT CONTROLS OVER ALL THE FUNCTIONS
//               RELATED TO ITEMS
//*****

class item
{
    public :
        void add_item(void) ;
        void delete_item(void) ;
        void modify_item(void) ;
        void list_of_item(void) ;
        void purchase(void) ;
    private :
        int last_code(void) ;
        void delete_record(int) ;
        void modify_record(int) ;
};

```

```

        void display_record(int) ;
        int item_found(int) ;
        int recordno(int) ;
        void sort(void) ;
        int itemcode ;
        char itemname[30] ;
        float itemprice ;
};

//*****
// CLASS NAME : account
// DETAILS : IT CONTROLS OVER ALL THE FUNCTIONS
// RELATED TO MAKING BILL
//*****

class account
{
    public :
        void bill_list(void) ;
        void prepare_bill(int) ;
        int last_billno(void) ;
        void add_bill(int t_billno, int t_itemcode, char *t_itemname,
                    float t_qty, float t_price) ;
    private :
        int code, billno;
        int dd, mm, yy ;
        float price, quantity ;
        char name[30] ;
};

//*****
// THIS FUNCTION CREATE MAIN MENU AND CALLS OTHER FUNCTIONS
//*****

void menu :: main_menu(void)
{
    clrscr() ;
    char ch ;
    item f;
    while (1)
    {
        clrscr() ;
        gotoxy(32,6) ;
        cout <<"SHOPPING MALL MANAGEMENT SYSTEM" ;
        gotoxy(32,7) ;
        cout <<"~~~~~" ;
        gotoxy(32,9) ;
        cout <<"1: PURCHASE" ;
    }
}

```

```

gotoxy(32,11);
cout <<"2: SEE MENU" ;
gotoxy(32,13);
cout <<"3: EDIT" ;
gotoxy(32,15);
cout <<"4: TOTAL BILL" ;
gotoxy(32,17);
cout <<"0: QUIT" ;
gotoxy(32,20);
cout <<"Enter Choice : " ;
ch = getche();
if (ch == 27)
    return ;
else
if (ch == '1')
    f.purchase();
else
if (ch == '2')
    f.list_of_item();
else
if (ch == '3')
    edit_menu();
else
if (ch == '4')
{
    account a ;
    a.bill_list();
}
else
if (ch == '0')
    break ;
}
}

```

```

//*****
//      THIS FUNCTION CREATE EDIT MENU AND CALLS OTHER FUNCTIONS
//*****

```

```

void menu :: edit_menu(void)
{
    char ch ;
    while (1)
    {
        for (int i=5; i<=21; i++)
        {
            gotoxy(21,i);
            cout <<"          ";
        }
        gotoxy(32,6);
    }
}

```

```

cout <<"E D I T M E N U" ;
gotoxy(32,7) ;
cout <<"~~~~~" ;
gotoxy(32,10) ;
cout <<"1: ADD ITEM" ;
gotoxy(32,12) ;
cout <<"2: MODIFY ITEM" ;
gotoxy(32,14) ;
cout <<"3: DELETE ITEM" ;
gotoxy(32,16) ;
cout <<"0: EXIT" ;
gotoxy(32,19) ;
cout <<"Enter Choice : " ;
ch = getche() ;
if (ch == 27)
    return ;
else
if (ch == '1')
{
    item f ;
    f.add_item() ;
    break ;
}
else
if (ch == '2')
{
    item f ;
    f.modify_item() ;
    break ;
}
else
if (ch == '3')
{
    item f ;
    f.delete_item() ;
    break ;
}
else
if (ch == '0')
    break ;
}
}

```

```

//*****
//      THIS FUNCTION RETURNS THE CODE OF THE LAST RECORD IN THE
//      ITEM FILE (ITEM.DAT).
//*****

```

```

int item :: last_code(void)

```

```

{
    fstream file ;
    file.open("ITEM.DAT", ios::in) ;
    file.seekg(0,ios::beg) ;
    int t=0 ;
    while (file.read((char *) this, sizeof(item)))
        t = itemcode ;
    file.close() ;
    cout<<"t="<<t;
    cout<<"press any key to contd..";
    getch();
    return t ;
}

```

```

//*****
//      THIS FUNCTION DISPLAY THE LIST OF THE ITEMS
//*****

```

```

void item :: list_of_item(void)
{
    clrscr() ;
    fstream file ;
    file.open("ITEM.DAT", ios::in) ;
    file.seekg(0) ;
    int row = 6 , found = 0 , pageno = 1 ;
    gotoxy(30,2) ;
    cout <<"LIST OF ITEMS" ;
    gotoxy(29,3) ;
    cout <<"~~~~~" ;
    gotoxy(3,4) ;
    cout <<"ITEM CODE  ITEM NAME          ITEM PRICE" ;
    gotoxy(2,5) ;
    cout <<"~~~~~" ;
    while (file.read((char *) this, sizeof(item)))
    {
        delay(20) ;
        found = 1 ;
        gotoxy(5,row) ;
        cout<<itemcode ;
        gotoxy(14,row) ;
        cout<<itemname ;
        gotoxy(37,row) ;
        cout<<itemprice ;
        if ( row == 22 )
        {
            row = 5 ;
            gotoxy(66,1) ;
            cout <<"Page no. : " <<pageno ;
            gotoxy(66,2) ;

```

```

        cout <<"=====";
        pageno++;
        gotoxy(1,25);
        cout <<"Press any key to continue...";
        getch();
        clrscr();
        gotoxy(30,2);
        cout <<"LIST OF ITEMS";
        gotoxy(3,4);
        cout <<"ITEM CODE  ITEM NAME          ITEM PRICE";
        gotoxy(2,5);
        cout <<"~::~::~::~::~::~::~::~::~::~::~::~::~::~::~::~::~";
        cout <<"~::~";
    }
    else
        row++;
}
if ( !found )
{
    gotoxy(5,10);
    cout <<"\nRecords not found";
}
gotoxy(66,1);
cout <<"Page no. : " <<pageno;
gotoxy(66,2);
cout <<"=====";
gotoxy(1,25);
cout <<"Press any key to continue...";
getch();
file.close ();
}

```

```

//*****
//          THIS FUNCTION ADD RECORDS IN THE ITEM FILE (ITEM.DAT)
//*****

```

```

void item:: add_item(void)
{
    int tcode, valid ;
    char ch, t_itemprice[10] ;
    tcode = last_code() ;
    tcode++;
    do
    {
        clrscr() ;
        gotoxy(71,2);
        cout <<"<0>=Exit" ;
        gotoxy(27,3);
        cout <<"ADD ITEM TO THE LIST" ;
    }
}

```



```

gotoxy(26,4) ;
cout <<"~~~~~";
gotoxy(5,6) ;
cout <<"Item Code : " <<tcode ;
gotoxy(5,8) ;
cout <<"Item Name : " ;
gotoxy(5,10) ;
cout <<"Item Price : " ;
do
{
    valid = 1 ;
    gotoxy(1,8) ; clrnl() ;
    gotoxy(1,24) ; clrnl() ;
    gotoxy(1,25) ; clrnl() ;
    gotoxy(3,25) ;
    cout <<"ENTER ITEM NAME TO ADD IN THE MENU" ;
    gotoxy(5,8) ;
    cout <<"Item Name : " ;
    gets(itemname) ;
   strupr(itemname) ;
    if (itemname[0] == '0')
        return ;
    if ((strlen(itemname) < 1) || (strlen(itemname) > 20))
    {
        valid = 0 ;
        gotoxy(3,24) ;
        cout <<"\7 Range = 1..20" ;
        getch() ;
    }
} while (!valid) ;
do
{
    valid = 1 ;
    gotoxy(1,12) ; clrnl() ;
    gotoxy(1,24) ; clrnl() ;
    gotoxy(1,25) ; clrnl() ;
    gotoxy(3,25) ;
    cout <<"ENTER ITEM PRICE TO ADD IN THE MENU" ;
    gotoxy(5,10) ;
    cout <<"Item Price : " ;
    gets(t_itemprice) ;
    itemprice = atof(t_itemprice) ;
    if (t_itemprice[0] == '0')
        return ;
    if (itemprice < 200 || itemprice > 2500)
    {
        valid = 0 ;
        gotoxy(3,24) ;
        cout <<"\7 Range = 200..2500" ;
        getch() ;
    }
}

```

```

    }
} while (!valid) ;
do
{
    gotoxy(1,15) ; clrcl() ;
    gotoxy(1,24) ; clrcl() ;
    gotoxy(1,25) ; clrcl() ;
    gotoxy(5,15) ;
    cout <<"Do you want to save this record (y/n) : " ;
    ch = getche() ;
    ch = toupper(ch) ;
    if (ch == '0')
        return ;
} while (ch != 'N' && ch != 'Y') ;
if (ch == 'Y')
{
    itemcode = tcode ;
    fstream file ;
    file.open("ITEM.DAT", ios::out | ios::app ) ;
    file.write((char *) this, sizeof(item)) ;
    file.close() ;
    tcode++ ;
}
do
{
    gotoxy(1,17) ; clrcl() ;
    gotoxy(1,24) ; clrcl() ;
    gotoxy(1,25) ; clrcl() ;
    gotoxy(5,17) ;
    cout <<"Do you want to add more records (y/n) : " ;
    ch = getche() ;
    ch = toupper(ch) ;
    if (ch == '0')
        return ;
} while (ch != 'N' && ch != 'Y') ;
} while (ch == 'Y') ;
}

```

```

//*****
//      THIS FUNCTION DISPLAY THE RECORD OF THE GIVEN CODE FROM
//      THE ITEM FILE (ITEM.DAT)
//*****

```

```

void item :: display_record(int tcode)
{
    fstream file ;
    file.open("ITEM.DAT", ios::in) ;
    file.seekg(0,ios::beg) ;
    while (file.read((char *) this, sizeof(item)))

```

```

    {
        if (itemcode == tcode)
        {
            gotoxy(5,3);
            cout <<"Item Code : "<<itemcode ;
            gotoxy(5,4);
            cout <<"Item Name : "<<itemname ;
            gotoxy(5,5);
            cout <<"Item Price : "<<itemprice ;
            break ;
        }
    }
    file.close() ;
}

```

```

//*****
//      THIS FUNCTION RETURN THE VALUE 1 IF THE RECORD IS FOUND
//      FOR THE GIVEN CODE IN THE ITEM FILE (ITEM.DAT)
//*****

```

```

int item :: item_found(int tcode)
{
    fstream file ;
    file.open("ITEM.DAT", ios::in) ;
    file.seekg(0,ios::beg) ;
    int found=0 ;
    while (file.read((char *) this, sizeof(item)))
    {
        if (itemcode == tcode)
        {
            found++ ;
            break ;
        }
    }
    file.close() ;
    return found ;
}

```

```

//*****
//      THIS FUNCTION RETURN THE RECORD NO. OF THE GIVEN CODE IN
//      THE ITEM FILE (ITEM.DAT)
//*****

```

```

int item :: recordno(int tcode)
{
    fstream file ;
    file.open("ITEM.DAT", ios::in) ;
    file.seekg(0,ios::beg) ;

```

```

int found=0 ;
while (file.read((char *) this, sizeof(item)))
{
    found++ ;
    if (itemcode == tcode)
        break ;
}
file.close() ;
return found ;
}

```

```

//*****
//      THIS FUNCTION DELETES THE RECORD FOR THE GIVEN CODE FROM
//      THE ITEM FILE (ITEM.DAT)
//*****

```

```

void item :: delete_record(int tcode)
{
    fstream file ;
    file.open("ITEM.DAT", ios::in) ;
    fstream temp ;
    temp.open("temp.dat", ios::out) ;
    file.seekg(0,ios::beg) ;
    while ( !file.eof() )
    {
        file.read((char *) this, sizeof(item)) ;
        if ( file.eof() )
            break ;
        if ( itemcode != tcode )
            temp.write((char *) this, sizeof(item)) ;
    }
    file.close() ;
    temp.close() ;
    file.open("ITEM.DAT", ios::out) ;
    temp.open("temp.dat", ios::in) ;
    temp.seekg(0,ios::beg) ;
    while ( !temp.eof() )
    {
        temp.read((char *) this, sizeof(item)) ;
        if ( temp.eof() )
            break ;
        file.write((char *) this, sizeof(item)) ;
    }
    file.close() ;
    temp.close() ;
}

```

```

/*****
//      THIS FUNCTION GIVES THE CODE NUMBER TO DELETE RECORD FROM
//      THE ITEM FILE (ITEM.DAT)
*****/

void item :: delete_item(void)
{
    clrscr() ;
    char t_code[5], ch ;
    int t, tcode ;
    gotoxy(3,25) ;
    cout <<"Press <ENTER> to see the list" ;
    gotoxy(5,3) ;
    cout <<"Enter Item Code of the item to be deleted : " ;
    gets(t_code) ;
    t = atoi(t_code) ;
    tcode = t ;
    if (t_code[0] == '0')
        return ;
    if (tcode == 0)
    {
        list_of_item() ;
        gotoxy(1,25) ; clrscr() ;
        gotoxy(3,25) ;
        cout <<"Press <ENTER> to Exit" ;
        gotoxy(5,24) ;
        cout <<"Enter Item Code of the item to be deleted : " ;
        gets(t_code) ;
        t = atoi(t_code) ;
        tcode = t ;
        if (tcode == 0)
            return ;
    }
    clrscr() ;
    if (!item_found(tcode))
    {
        gotoxy(5,5) ;
        cout <<"\7Record not found" ;
        getch() ;
        return ;
    }
    display_record(tcode) ;
    do
    {
        gotoxy(1,8) ; clrscr() ;
        gotoxy(5,8) ;
        cout <<"Do you want to delete this record (y/n) : " ;
        ch = getche() ;
        ch = toupper(ch) ;
    } while (ch != 'N' && ch != 'Y') ;
}

```

```

    if (ch == 'N')
        return ;
    delete_record(tcode) ;
    gotoxy(5,15) ;
    cout <<"\7Record Deleted" ;
    getch() ;
}

```

```

//*****
//      THIS FUNCTION MODIFY THE RECORD FOR THE GIVEN CODE FROM
//      THE ITEM FILE (ITEM.DAT)
//*****

```

```

void item :: modify_record(int tcode)
{
    int recno ;
    recno = recordno(tcode) ;
    int valid, t_code ;
    char ch, t_itemprice[10], t_itemcode[5] ;
    gotoxy(71,2) ;
    cout <<"<0>=Exit" ;
    gotoxy(5,12) ;
    cout <<"Item Code : " ;
    gotoxy(5,14) ;
    cout <<"Item Name : " ;
    gotoxy(5,16) ;
    cout <<"Item Price : " ;
    do
    {
        gotoxy(20,12) ; clrnl() ;
        cout <<"Change (y/n) : " ;
        ch = getche() ;
        ch = toupper(ch) ;
        if (ch == '0')
            return ;
    } while (ch != 'N' && ch != 'Y') ;
    valid = 0 ;
    while (ch == 'Y' && !valid)
    {
        valid = 1 ;
        gotoxy(1,12) ; clrnl() ;
        gotoxy(1,24) ; clrnl() ;
        gotoxy(1,25) ; clrnl() ;
        gotoxy(3,25) ;
        cout <<"ENTER ITEM CODE TO ADD IN THE MENU" ;
        gotoxy(5,12) ;
        cout <<"Item Code : " ;
        gets(t_itemcode) ;
        t_code = atoi(t_itemcode) ;
    }
}

```

```

if (t_code == 0)
    return ;
if (item_found(t_code) && t_code != tcode)
{
    valid = 0 ;
    gotoxy(3,24) ;
    cout <<"\7 CODE ALREADY GIVEN" ;
    getch() ;
}
}
if(ch=='N')
{
    t_code=tcode;
}
do
{
    gotoxy(20,14) ; clrhol() ;
    cout <<"Change (y/n) : " ;
    ch = getche() ;
    ch = toupper(ch) ;
    if (ch == '0')
        return ;
} while (ch != 'N' && ch != 'Y') ;
valid = 0 ;
while (ch == 'Y' && !valid)
{
    valid = 1 ;
    gotoxy(1,14) ; clrhol() ;
    gotoxy(1,24) ; clrhol() ;
    gotoxy(1,25) ; clrhol() ;
    gotoxy(3,25) ;
    cout <<"ENTER ITEM NAME TO ADD IN THE MENU" ;
    gotoxy(5,14) ;
    cout <<"Item Name : " ;
    gets(itemname) ;
   strupr(itemname) ;
    if (itemname[0] == '0')
        return ;
    if ((strlen(itemname) < 1) || (strlen(itemname) > 20))
    {
        valid = 0 ;
        gotoxy(3,24) ;
        cout <<"\7 Range = 1..20" ;
        getch() ;
    }
}
do
{
    gotoxy(20,16) ; clrhol() ;
    cout <<"Change (y/n) : " ;

```

```

    ch = getche() ;
    ch = toupper(ch) ;
    if (ch == '0')
        return ;
} while (ch != 'N' && ch != 'Y') ;
/* valid = 0 ;
while (ch == 'Y' && !valid)
{
    valid = 1 ;
    gotoxy(1,16) ; clrnl() ;
    gotoxy(1,24) ; clrnl() ;
    gotoxy(1,25) ; clrnl() ;
    gotoxy(3,25) ;
    cout <<"ENTER ITEM COST TO ADD IN THE MENU" ;
    gotoxy(5,16) ;
    cout <<"Item Cost : " ;
    gets(t_itemcost) ;
    itemcost = atof(t_itemcost) ;
    if (t_itemcost[0] == '0')
        return ;
    if (itemcost < 1 || itemcost > 800)
    {
        valid = 0 ;
        gotoxy(3,24) ;
        cout <<"\7 Range = 1..800" ;
        getch() ;
    }
}
do
{
    gotoxy(20,18) ; clrnl() ;
    cout <<"Change (y/n) : " ;
    ch = getche() ;
    ch = toupper(ch) ;
    if (ch == '0')
        return ;
} while (ch != 'N' && ch != 'Y') ;
*/
valid = 0 ;
while (ch == 'Y' && !valid)
{
    valid = 1 ;
    gotoxy(1,18) ; clrnl() ;
    gotoxy(1,24) ; clrnl() ;
    gotoxy(1,25) ; clrnl() ;
    gotoxy(3,25) ;
    cout <<"ENTER ITEM PRICE TO ADD IN THE MENU" ;
    gotoxy(5,16) ;
    cout <<"Item Price : " ;
    gets(t_itemprice) ;

```



```

    itemprice = atof(t_itemprice) ;
    if (t_itemprice[0] == '0')
        return ;
    if (itemprice <200 || itemprice > 2500)
    {
        valid = 0 ;
        gotoxy(3,24) ;
        cout <<"\7 Range = 200..2500" ;
        getch() ;
    }
}
do
{
    gotoxy(1,21) ; clrnl() ;
    gotoxy(1,24) ; clrnl() ;
    gotoxy(1,25) ; clrnl() ;
    gotoxy(5,21) ;
    cout <<"Do you want to save this record (y/n) : " ;
    ch = getche() ;
    ch = toupper(ch) ;
    if (ch == '0')
        return ;
} while (ch != 'N' && ch != 'Y') ;
if (ch == 'N')
    return ;
itemcode = t_code ;
getch() ;
fstream file ;
file.open("ITEM.DAT", ios::out | ios::ate) ;
int location ;
location = (recno-1) * sizeof(item) ;
file.seekp(location) ;
file.write((char *) this, sizeof(item)) ;
file.close() ;
sort() ;
clrscr() ;
gotoxy(5,15) ;
cout <<"\7Record Modified" ;
getch() ;
}

//*****
//      THIS FUNCTION GIVES THE CODE NO. TO MODIFY RECORD FROM
//      THE ITEM FILE (ITEM.DAT)
//*****

void item :: modify_item(void)
{
    clrscr() ;
    char t_code[5], ch ;

```

```

int t, tcode ;
gotoxy(3,25) ;
cout <<"Press <ENTER> to see the list" ;
gotoxy(5,3) ;
cout <<"Enter Item Code of the item to be Modify : " ;
gets(t_code) ;
t = atoi(t_code) ;
tcode = t ;
if (t_code[0] == '0')
    return ;
if (tcode == 0)
{
    list_of_item() ;
    gotoxy(1,25) ; clrscr() ;
    gotoxy(3,25) ;
    cout <<"Press <ENTER> to Exit" ;
    gotoxy(5,24) ;
    cout <<"Enter Item Code of the item to be modify : " ;
    gets(t_code) ;
    t = atoi(t_code) ;
    tcode = t ;
    if (tcode == 0)
        return ;
}
clrscr() ;
if (!item_found(tcode))
{
    gotoxy(5,5) ;
    cout <<"\7Record not found" ;
    getch() ;
    return ;
}
display_record(tcode) ;
do
{
    gotoxy(5,8) ;
    cout <<"Do you want to Modify this record (y/n) : " ;
    ch = getche() ;
    ch = toupper(ch) ;
} while (ch != 'N' && ch != 'Y') ;
if (ch == 'N')
    return ;
modify_record(tcode) ;
}

//*****
//    THIS FUNCTION SORT THE RECORD IN THE ITEM FILE (ITEM.DAT)
//    ACCORDING TO THE CODE NUMBERS.
//*****

```

```

void item :: sort(void)
{
    int i=0,j ;
    item arr[100] ;
    item temp ;
    fstream file ;
    file.open("ITEM.DAT", ios::in) ;
    file.seekg(0,ios::beg) ;
    while (file.read((char *) &arr[i], sizeof(item)))
        i++ ;
    int size ;
    size = i ;
    file.close() ;
    for (i=1; i<size; i++)
        for (j=0; j<size-i; j++)
            {
                if (arr[j].itemcode > arr[j+1].itemcode)
                    {
                        temp=arr[j];
                        arr[j]=arr[j+1];
                        arr[j+1]=temp;
                    }
            }
    file.open("ITEM.DAT", ios::out) ;
    for (i=0; i<size; i++)
        file.write((char *) &arr[i], sizeof(item)) ;
    file.close() ;
}

```

```

//*****
//      THIS FUNCTION PURCHASES THE FOOD ITEM IN THE LIST
//*****

```

```

void item :: purchase(void)
{
    clrscr() ;
    account a ;
    int t_billno, purchased=0 ;
    t_billno = a.last_billno() ;
    t_billno++ ;
    char t_code[5], ch, t_quantity[5] ;
    int t, tcode, i=0, valid ;
    float qty ;
    int t_itemcode ;
    float t_qty, t_price ;
    char t_itemname[30] ;
    struct date d;
    int d1, m1, y1 ;
    getdate(&d);
    d1 = d.da_day ;

```

```

m1 = d.da_mon ;
y1 = d.da_year ;
do
{
    clrscr() ;
    gotoxy(3,25) ;
    cout <<"Press <ENTER> to see the list" ;
    gotoxy(5,3) ;
    cout <<"Enter Item Code of the item to be Purchase : " ;
    gets(t_code) ;
    t = atoi(t_code) ;
    tcode = t ;
    if (t_code[0] == '0')
    {
        if (purchased)
            a.prepare_bill(t_billno) ;
        return ;
    }
    if (tcode == 0)
    {
        list_of_item() ;
        gotoxy(1,25) ; clrscr() ;
        gotoxy(3,25) ;
        cout <<"Press <ENTER> to Exit" ;
        gotoxy(5,24) ;
        cout <<"Enter Item Code of the item to be Purchased : " ;
        gets(t_code) ;
        t = atoi(t_code) ;
        tcode = t ;
        if (tcode == 0)
        {
            if (purchased)
                a.prepare_bill(t_billno) ;
            return ;
        }
    }
    clrscr() ;
    if (!item_found(tcode))
    {
        gotoxy(5,5) ;
        cout <<"\7Item Code not found" ;
        getch() ;
        if (purchased)
            a.prepare_bill(t_billno) ;
        return ;
    }
    gotoxy(60,2) ;
    cout <<"Date:" <<d1 <<"/" <<m1 <<"/" <<y1 ;
    display_record(tcode) ;
}
do

```

```

{
    valid = 1 ;
    gotoxy(1,8) ; clr() ;
    gotoxy(1,24) ; clr() ;
    gotoxy(1,25) ; clr() ;
    gotoxy(3,25) ;
    cout <<"ENTER QUANTITY TO BE PURCHASED IN Numbers." ;
    gotoxy(5,8) ;
    cout <<"Quantity : " ;
    gets(t_quantity) ;
    qty = atoi(t_quantity) ;
    if (t_quantity[0] == '0')
    {
        if (purchased)
            a.prepare_bill(t_billno) ;
        return ;
    }
    if (qty < 1 || qty > 100)
    {
        valid = 0 ;
        gotoxy(3,24) ;
        cout <<"\7 Range = 1..100" ;
        getch() ;
    }
} while (!valid) ;
do
{
    gotoxy(5,10) ; clr() ;
    gotoxy(5,10) ;
    cout <<"Do you want to cancel this purchase (y/n) : " ;
    ch = getch() ;
    ch = toupper(ch) ;
} while (ch != 'N' && ch != 'Y') ;
if (ch == 'N')
{
    purchased = 1 ;
    fstream file ;
    file.open("ITEM.DAT", ios::in) ;
    file.seekg(0,ios::beg) ;
    while (file.read((char *) this, sizeof(item)))
    {
        if (itemcode == tcode)
        {
            t_itemcode = itemcode ;
            strcpy(t_itemname,itemname) ;
            // t_cost = itemcost ;
            t_price = itemprice ;
            t_qty = qty ;
            a.add_bill(t_billno,t_itemcode
                ,t_itemname,t_qty,t_price) ;
        }
    }
}

```

```

                i++;
                break ;
            }
        }
        file.close() ;
    }
do
{
    gotoxy(5,12) ; clrscr() ;
    gotoxy(5,12) ;
    cout <<"Do you want to purchase more (y/n) : " ;
    ch = getch() ;
    ch = toupper(ch) ;
    } while (ch != 'N' && ch != 'Y') ;
} while (ch == 'Y') ;
a.prepare_bill(t_billno) ;
}

```

```

//*****
//      THIS FUNCTION RETURNS THE BILL NO. OF THE LAST
//      RECORD IN THE BILL FILE (BILLING.DAT)
//*****

```

```

int account :: last_billno(void)
{
    fstream file ;
    file.open("BILLING.DAT", ios::in) ;
    file.seekg(0,ios::beg) ;
    int t=0 ;
    while (file.read((char *) this, sizeof(account)))
        t = billno ;
    file.close() ;
    return t ;
}

```

```

//*****
//      THIS FUNCTION ADDS THE RECORD IN THE BILL FILE (BILLING.DAT)
//*****

```

```

void account :: add_bill(int t_billno, int t_itemcode, char t_itemname[30]
, float t_qty, float t_price)
{
    struct date d;
    int d1, m1, y1 ;
    getdate(&d);
    d1 = d.da_day ;

    m1 = d.da_mon ;

```

```

y1 = d.da_year ;
dd = d1 ;
mm = m1 ;
yy = y1 ;
code = t_itemcode ;
strcpy(name,t_itemname) ;
price = t_price ;
quantity = t_qty ;
billno = t_billno ;
fstream file ;
file.open("BILLING.DAT", ios::out | ios::    app ) ;
file.write((char *) this, sizeof(account));
cout<<this->quantity;
cout<<"press any key to contd..";
file.close() ;
}

```

```

//*****
// THIS FUNCTION PREPARES AND DISPLAYS THE BILL FOR THE
// GIVEN BILL NO. ACCORDING TO PURCHASES MADE.
//*****

```

```

void account :: prepare_bill(int t_billno)
{
    clrscr() ;
    struct date d;
    int d1, m1, y1 ;
    getdate(&d);
    d1 = d.da_day ;
    m1 = d.da_mon ;
    y1 = d.da_year ;
    float total=0.0, total_bill=0.0 ;
    gotoxy(33,3) ;
    cout <<"CUSTOMER BILL" ;
    gotoxy(32,4) ;
    cout <<"~~~~~" ;
    gotoxy(55,5) ;
    cout <<"Date:" <<d1 <<"/" <<m1 <<"/" <<y1 ;
    gotoxy(8,7) ;
    cout <<"ITEMS PURCHASED" ;
    gotoxy(8,8) ;
    cout <<"~~~~~" ;
    gotoxy(8,9) ;
    cout <<"Item code Item name          Price Qty Total" ;
    gotoxy(8,10) ;
    cout <<"~~~~~" ;
    int row=11 ;
    fstream file ;
    file.open("BILLING.DAT", ios::in) ;
}

```

```

file.seekg(0) ;
while (file.read((char *) this, sizeof(account)))
{
    if (billno == t_billno)
    {
        gotoxy(8,5) ;
        cout <<"BILL NO. # " <<billno ;
        gotoxy(8,6) ;
        cout <<"~~~~~" ;
        gotoxy(10,row) ;
        cout <<code ;
        gotoxy(18,row) ;
        cout <<name ;
        gotoxy(39,row) ;
        cout <<price ;
        gotoxy(56,row) ;
        cout <<quantity ;
        total = quantity * price ;
        gotoxy(63,row) ;
        cout <<total ;
        total_bill = total_bill + total ;
        row++ ;
    }
}
file.close() ;
gotoxy(39,row+1) ;
cout <<"TOTAL BILL: Rs." <<total_bill <<" /=" ;
getch() ;
}

```

```

//*****
// THIS FUNCTION DISPLAYS THE LIST OF THE BILLS
//*****

```

```

void account :: bill_list(void)
{
    clrscr() ;
    fstream file ;
    file.open("BILLING.DAT", ios::in) ;
    file.seekg(0) ;
    int row=5, found=0, pageno=1, prev_billno=0, flag=0 ;
    float total=0.0, total_bill=0.0 ;
    gotoxy(30,2) ;
    cout <<"LIST OF BILLS" ;
    gotoxy(29,3) ;
    cout <<"~~~~~" ;
    gotoxy(3,4) ;
    cout <<"Billno. Date      Item Code  Item name      Price";
    cout <<" Qty  Total" ;

```



```

gotoxy(3,5);
cout <<"~~~~~";
cout <<"~~~~~" ;
while (file.read((char *) this, sizeof(account)))
{
    row++;
    delay(20);
    found = 1;
    if (prev_billno != billno)
    {
        if (flag)
        {
            gotoxy(52,row);
            // cout <<"TOTAL BILL: Rs." <<total_bill <<"/-" ;
            // total_bill = 0.0 ;
            row++;
        }
        gotoxy(4,row);
        cout <<billno ;
    }
    flag = 1 ;
    gotoxy(11,row);
    cout <<dd <<"/" <<mm <<"/" <<yy ;
    gotoxy(24,row);
    cout <<code ;
    gotoxy(32,row);
    cout <<name ;
    gotoxy(52,row);
    cout <<price ;
    gotoxy(67,row);
    cout <<quantity ;
    total = quantity * price ;
    gotoxy(73,row);
    cout <<total ;
    total_bill = total_bill + total ;
    if ( row >= 23 )
    {
        row = 5 ;
        gotoxy(66,1);
        cout <<"Page no. : " <<pageno ;
        pageno++;
        gotoxy(1,25);
        cout <<"Press any key to continue..." ;
        getche();
        clrscr();
        gotoxy(30,2);
        cout <<"LIST OF BILLS" ;
        gotoxy(3,4);
        cout <<"Billno. Date    Item Code Item name";
        cout <<" Price Qty Total" ;
    }
}

```

```

        gotoxy(3,5);
        cout <<"~~~~~";
        cout <<"~~~~~" ;
    }
    prev_billno = billno ;
}
row++;
// gotoxy(52,row) ;
// cout <<"TOTAL BILL: Rs." <<total_bill<<"/-" ;
if ( !found )
{
    gotoxy(5,10) ;
    cout <<"\7Records not found" ;
}
gotoxy(66,1) ;
cout <<"Page no. : " <<pageno ;
gotoxy(40,23);
cout <<"TOTAL BILL: Rs." <<total_bill<<"/-" ;
gotoxy(1,25) ;
cout <<"Press any key to continue..." ;
getche() ;
file.close () ;
}

//*****
// THIS FUNCTION IS THE MAIN FUNCTION CALLING THE MAIN MENU AND OPENING
// FUNCTION
//*****

void main(void)
{
    clrscr() ;
    menu m ;
    m.main_menu() ;
}

```

OUTPUTS

SHOPPING MALL MANAGEMENT SYSTEM

~~~~~  
1: PURCHASE  
2: SEE MENU  
3: ITEM  
4: TOTAL BILL  
0: QUIT  
Enter Choice : 3

<0>=Exit

## ADD ITEM IN LIST

~~~~~  
Item Code : 1
Item Name : Children Top
Item Price : 250
Do you want to save this record (y/n) : Y
Do you want to enter more records (y/n) : N

<0>=Exit

Item Code : 2
Item Name : Children Lower
Item Price : 200
Do you want to Modify this record (y/n) : Y

Item Code : Change (y/n) : N
Item Name : Change (y/n) : N
Item Price : 225
Do you want to save this record (y/n) : Y
Record Modified.....

Enter the item code to be deleted 3
Item Code : 3
Item Name : Children Shorts
Item Price : 200
Do you want to delete this record (y/n) : n

Enter the itemcode of the item to be purchased 1

Date:20/10/2008
Item Code : 1
Item Name : Children Top
Item Price : 250
Quantity : 2

Date:20/10/2008 I

Item Code : 1

Item Name : Children Top

Item Price : 250

Quantity : 2

Do you want to cancel this purchase (y/n) : n

CUSTOMER BILL

BILL NO. # 1

Date:20/10/2008

ITEMS PURCHASED

Item code	Item name	Price	Qty	Total
-----------	-----------	-------	-----	-------

1	Children Top	250	2	500
---	--------------	-----	---	-----

TOTAL BILL: Rs.500 /=

LIST OF ITEMS

ITEM CODE	ITEM NAME	ITEM PRICE
-----------	-----------	------------

1	Children Top	250
2	Children Lower	225
3	Children Shorts	200

LIST OF BILLS

Billno.	Date	Item Code	Item name	Price	Qty	Total
---------	------	-----------	-----------	-------	-----	-------

1	17/10/2008	1	Children Top	250	2	500
---	------------	---	--------------	-----	---	-----

2	17/10/2008	2	Children Lower	225	1	225
---	------------	---	----------------	-----	---	-----

TOTAL BILL: Rs.725/-

Press any key to continue...