OBJECT ORIENTED PROGRAMMING

CSC 210

Bahria University
Discovering Knowledge

Term Project

Super Market Management System

Mohammad Abdullah (01-134211-049)
BS(CS)-2B

Super Market Management System

Abstract:

This document outlines the developed code and class diagram of Super Market Management System.

Introduction:

Super Market Management System project is developed in C++ platform. This is simple and basic level small project for learning purpose. You can modify this project as per your requirements and develop a perfect advance level project. It is developed using C++ language.

Platform:

The platform (IDE) used to develop this program is Visual Studio 2019.

Concepts Used:

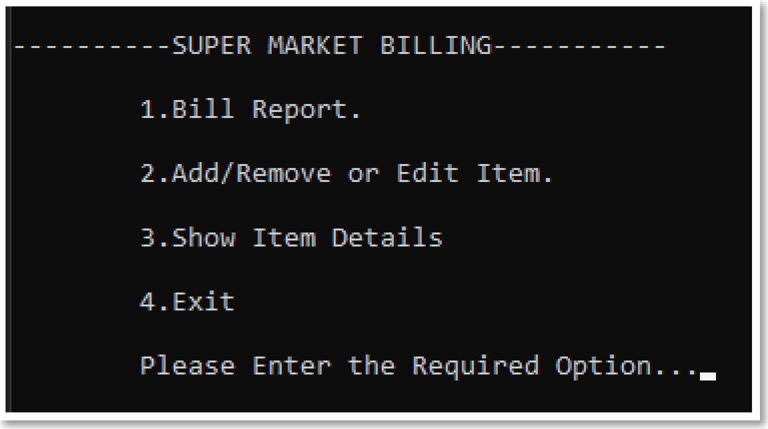
The concepts of Object-Oriented Programming used in developing this project include:

- Class and Objects
- Composition of Classes
- Inheritance of Classes
- File Handling

Features of this Project:

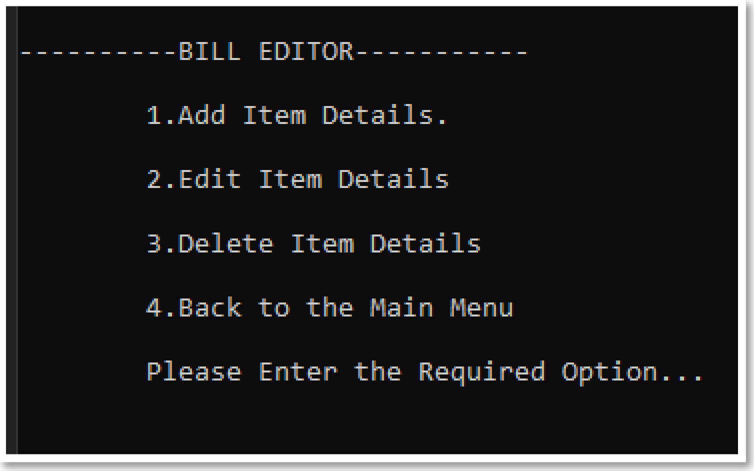
Features of this project in managing products and their bill details are:

1. Managing Bill Report
    i. Displaying Bill Report of All Items.



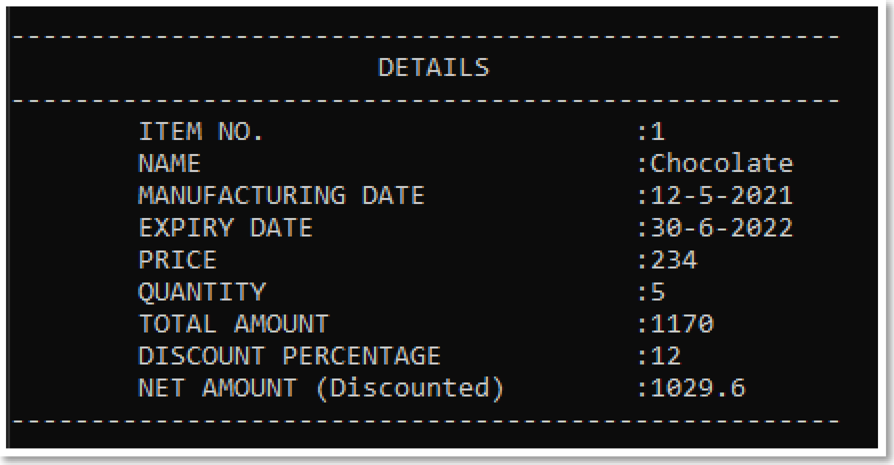
```
----------SUPER MARKET BILLING-----------

        1.Bill Report.

        2.Add/Remove or Edit Item.

        3.Show Item Details

        4.Exit

        Please Enter the Required Option..._
```

2. Adding, Removing and Editing Item Details.



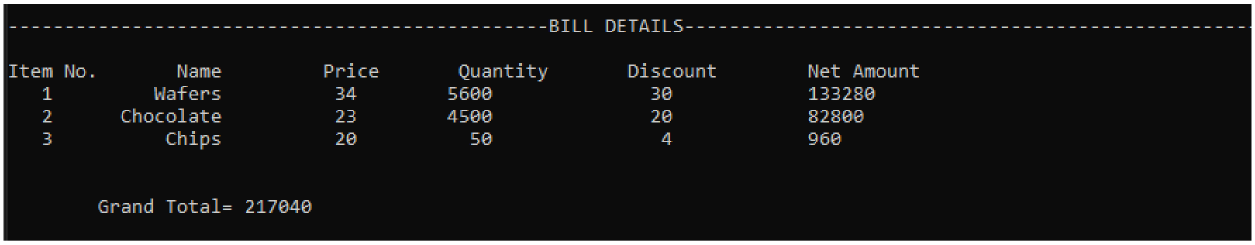
```
----------BILL EDITOR-----------

        1.Add Item Details.

        2.Edit Item Details

        3.Delete Item Details

        4.Back to the Main Menu

        Please Enter the Required Option...
```

3. Showing Details of Particular Item.



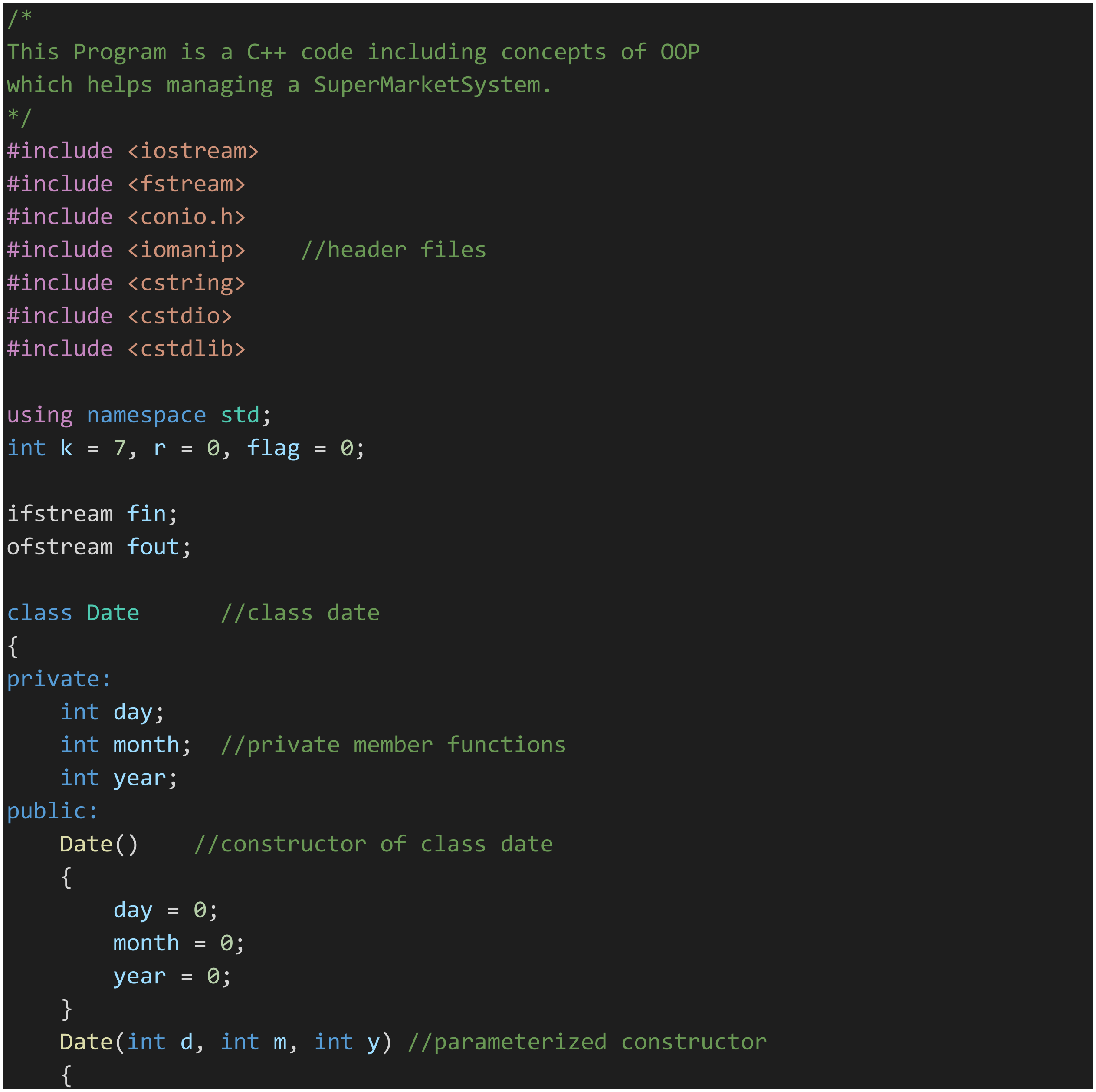
```
----------------------------------------------------
                       DETAILS
----------------------------------------------------
        ITEM NO.                       :1
        NAME                           :Chocolate
        MANUFACTURING DATE             :12-5-2021
        EXPIRY DATE                    :30-6-2022
        PRICE                          :234
        QUANTITY                       :5
        TOTAL AMOUNT                   :1170
        DISCOUNT PERCENTAGE            :12
        NET AMOUNT (Discounted)        :1029.6
----------------------------------------------------
```

4. Showing Bill Report of all the Existing Items.

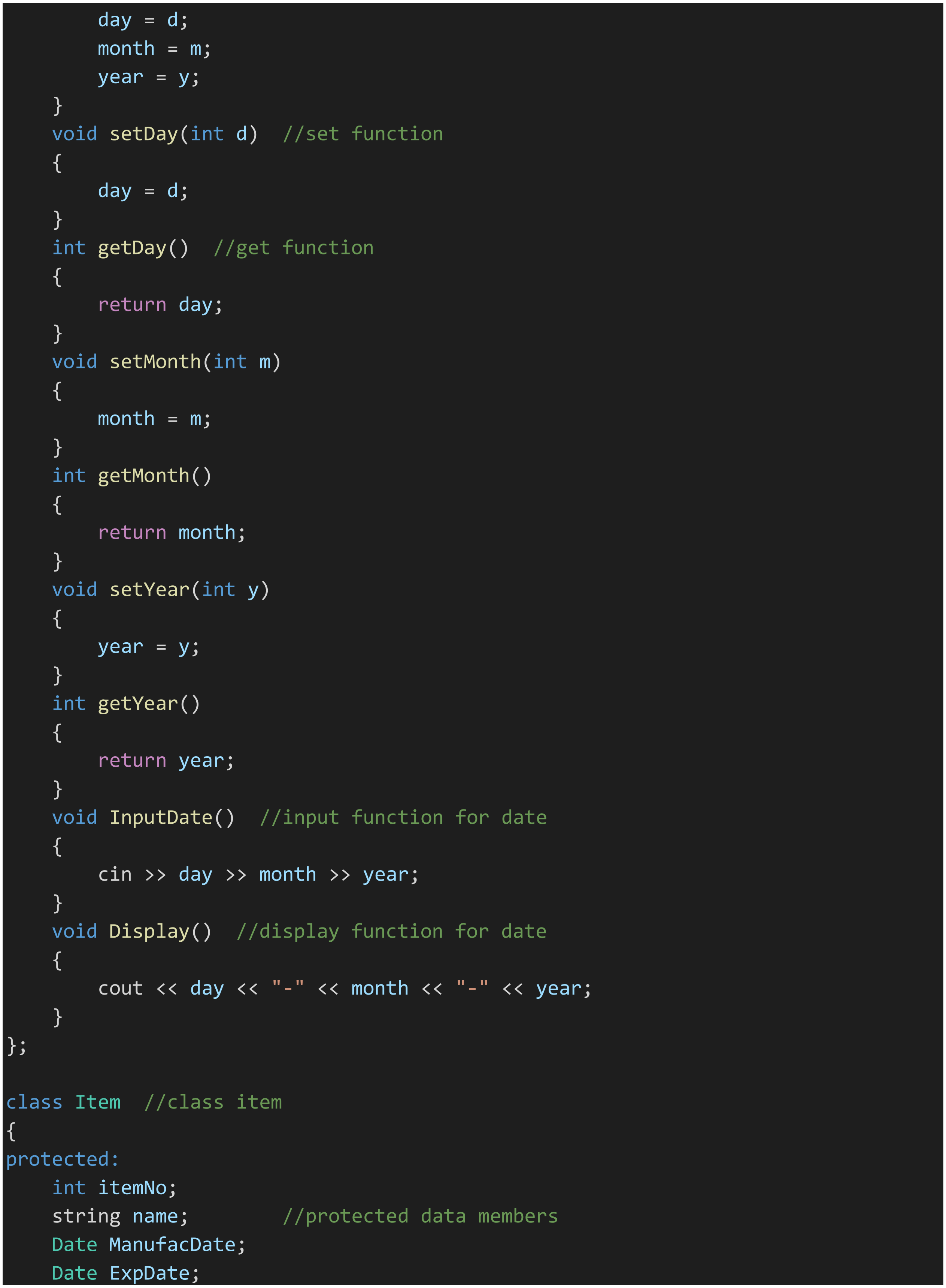


```
-----------------------------------------------BILL DETAILS-----------------------------------------------

Item No.        Name          Price         Quantity        Discount          Net Amount
   1          Wafers           34          5600             30             133280
   2       Chocolate           23          4500             20             82800
   3           Chips           20            50              4             960


        Grand Total= 217040
```

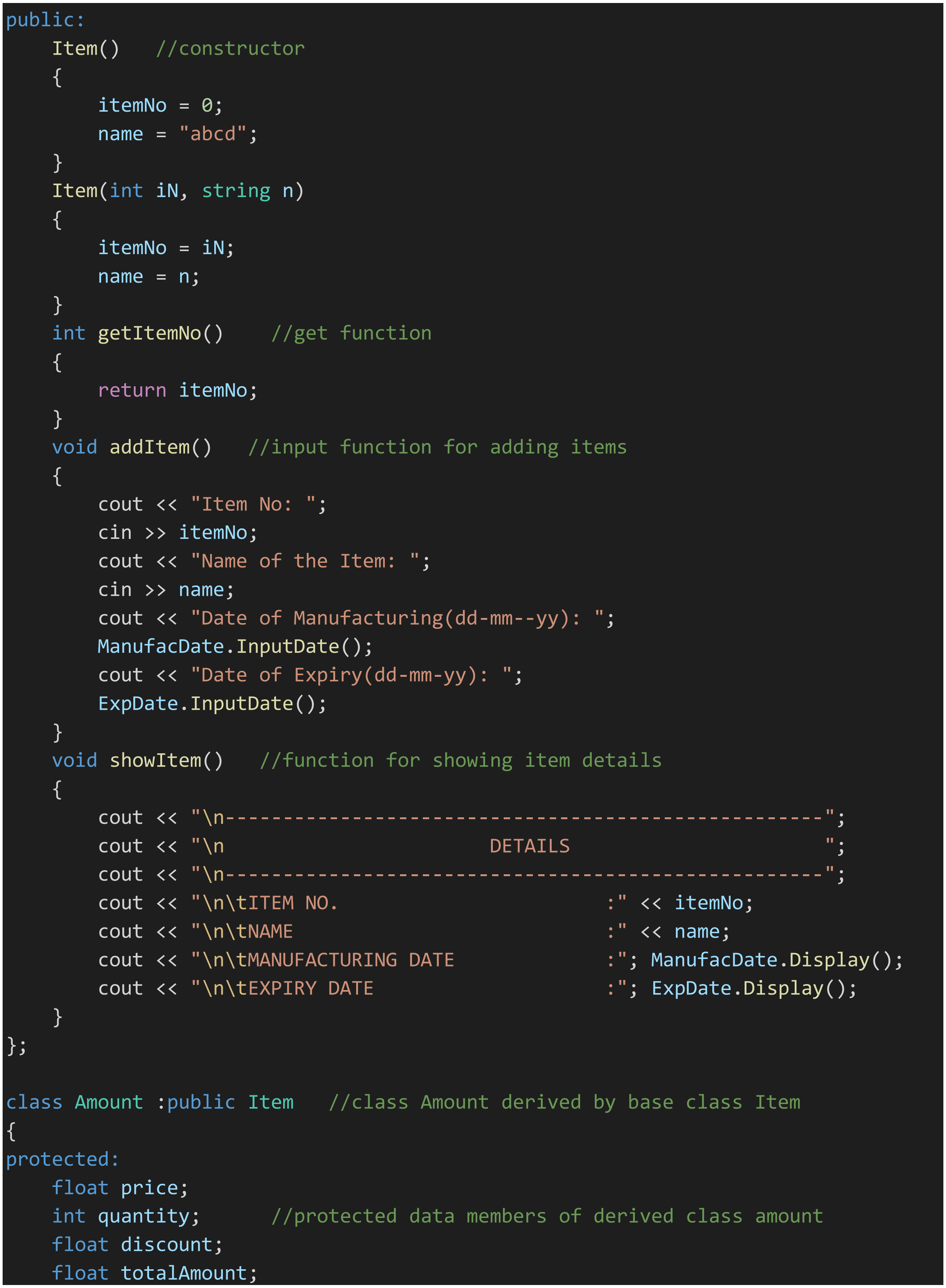
Source Code:



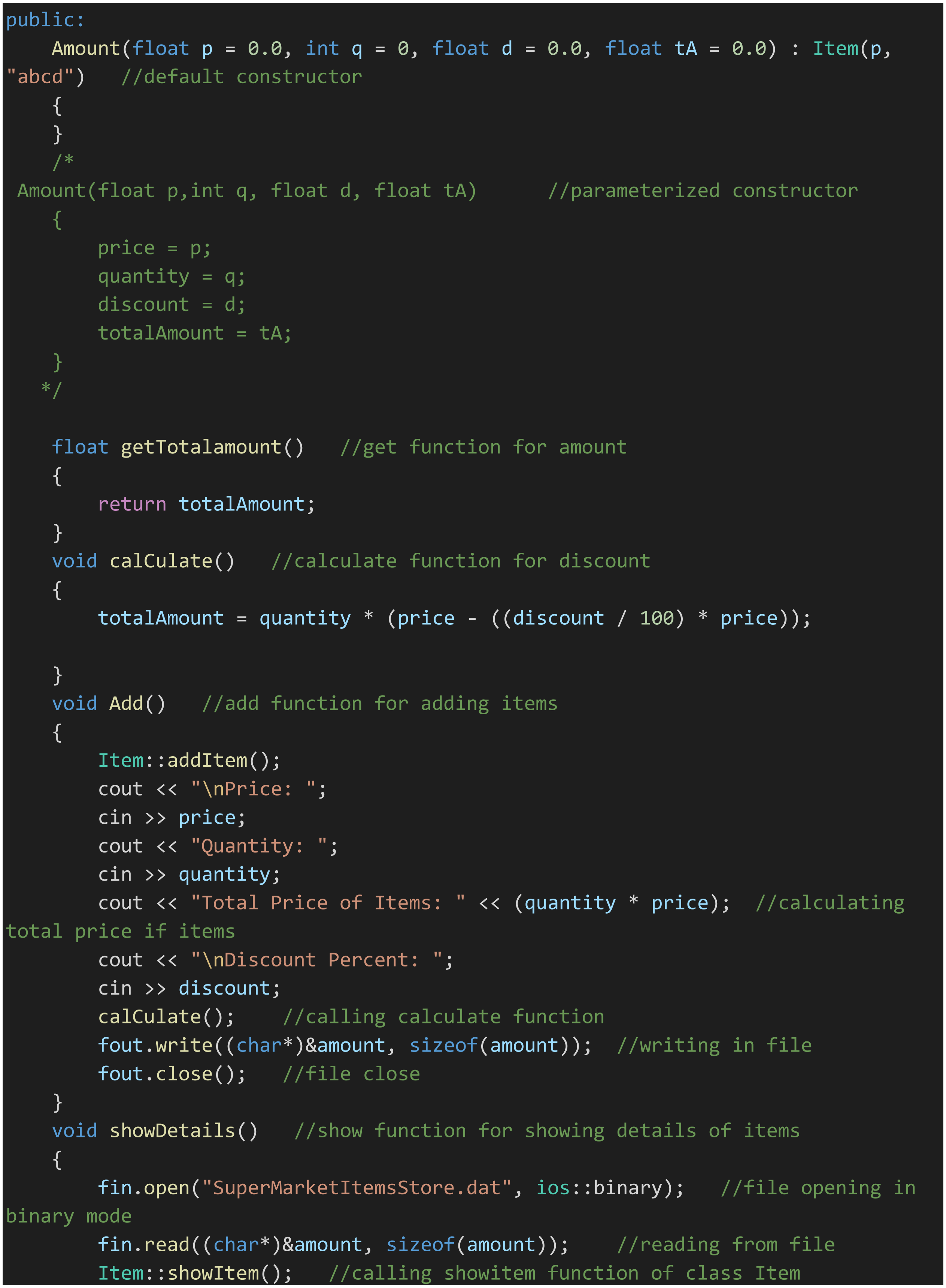
```
/*
This Program is a C++ code including concepts of OOP
which helps managing a SuperMarketSystem.
*/
#include <iostream>
#include <fstream>
#include <conio.h>
#include <iomanip>    //header files
#include <cstring>
#include <cstdio>
#include <cstdlib>

using namespace std;
int k = 7, r = 0, flag = 0;

ifstream fin;
ofstream fout;

class Date      //class date
{
private:
    int day;
    int month;  //private member functions
    int year;
public:
    Date()    //constructor of class date
    {
        day = 0;
        month = 0;
        year = 0;
    }
    Date(int d, int m, int y) //parameterized constructor
    {
```



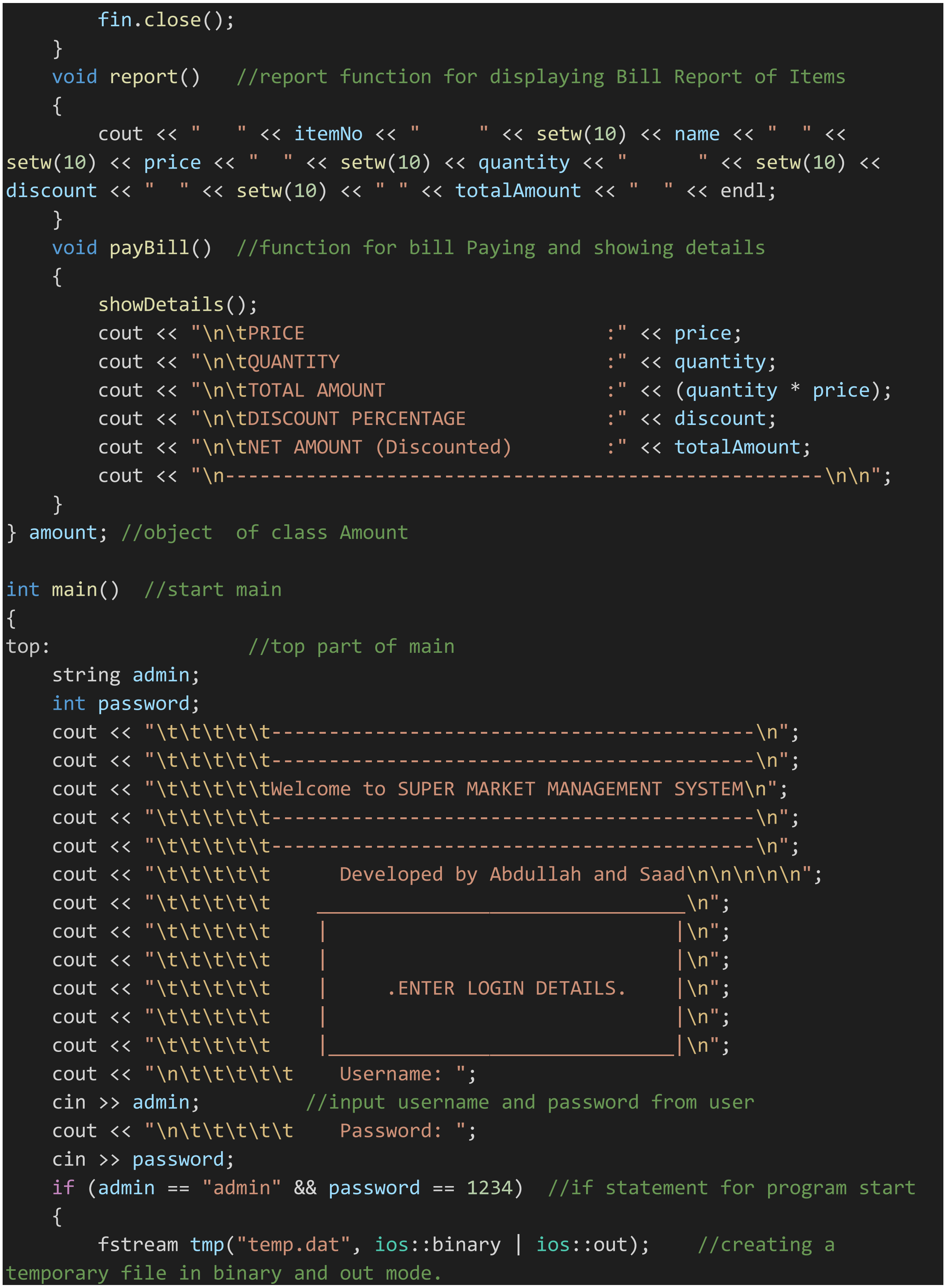
```
        day = d;
        month = m;
        year = y;
    }
    void setDay(int d)  //set function
    {
        day = d;
    }
    int getDay()  //get function
    {
        return day;
    }
    void setMonth(int m)
    {
        month = m;
    }
    int getMonth()
    {
        return month;
    }
    void setYear(int y)
    {
        year = y;
    }
    int getYear()
    {
        return year;
    }
    void InputDate()  //input function for date
    {
        cin >> day >> month >> year;
    }
    void Display()  //display function for date
    {
        cout << day << "-" << month << "-" << year;
    }
};

class Item  //class item
{
protected:
    int itemNo;
    string name;        //protected data members
    Date ManufacDate;
    Date ExpDate;
```



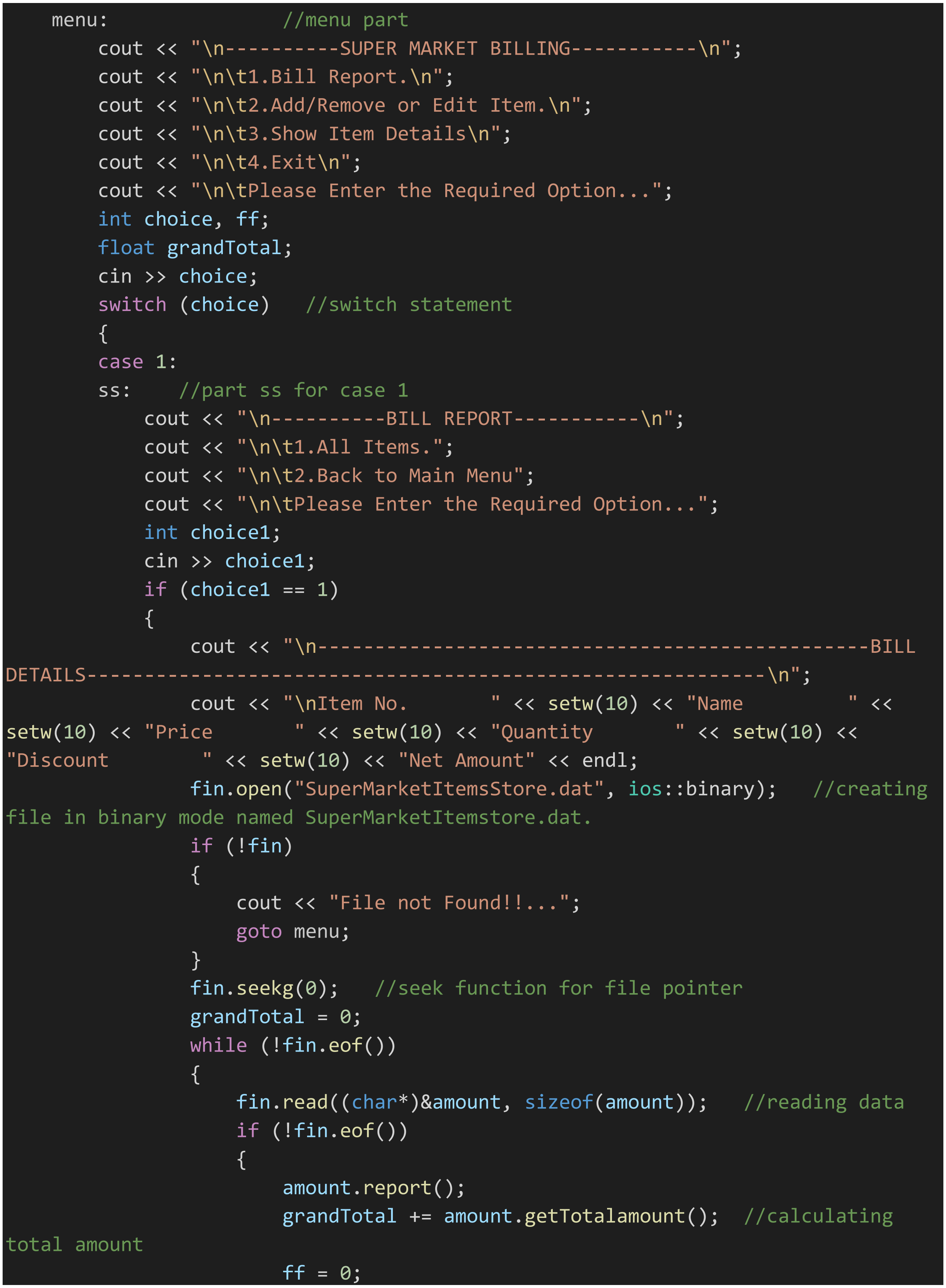
```
public:
    Item()   //constructor
    {
        itemNo = 0;
        name = "abcd";
    }
    Item(int iN, string n)
    {
        itemNo = iN;
        name = n;
    }
    int getItemNo()    //get function
    {
        return itemNo;
    }
    void addItem()   //input function for adding items
    {
        cout << "Item No: ";
        cin >> itemNo;
        cout << "Name of the Item: ";
        cin >> name;
        cout << "Date of Manufacturing(dd-mm--yy): ";
        ManufacDate.InputDate();
        cout << "Date of Expiry(dd-mm-yy): ";
        ExpDate.InputDate();
    }
    void showItem()   //function for showing item details
    {
        cout << "\n----------------------------------------------------";
        cout << "\n                       DETAILS                      ";
        cout << "\n----------------------------------------------------";
        cout << "\n\tITEM NO.                       :" << itemNo;
        cout << "\n\tNAME                           :" << name;
        cout << "\n\tMANUFACTURING DATE             :"; ManufacDate.Display();
        cout << "\n\tEXPIRY DATE                    :"; ExpDate.Display();
    }
};

class Amount :public Item   //class Amount derived by base class Item
{
protected:
    float price;
    int quantity;      //protected data members of derived class amount
    float discount;
    float totalAmount;
```



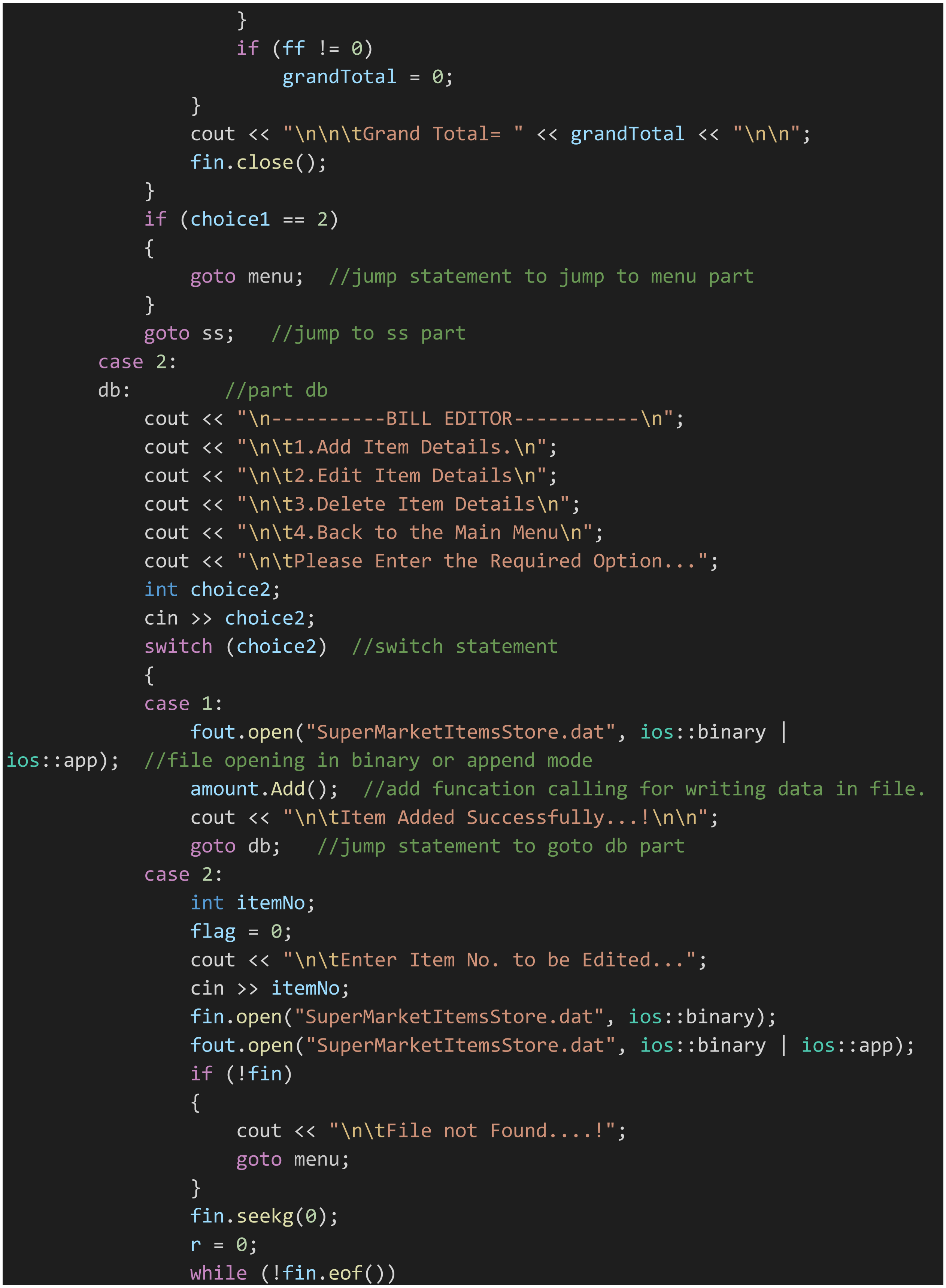
```
public:
    Amount(float p = 0.0, int q = 0, float d = 0.0, float tA = 0.0) : Item(p,
"abcd")   //default constructor
    {
    }
    /*
 Amount(float p,int q, float d, float tA)      //parameterized constructor
    {
        price = p;
        quantity = q;
        discount = d;
        totalAmount = tA;
    }
   */

    float getTotalamount()   //get function for amount
    {
        return totalAmount;
    }
    void calCulate()   //calculate function for discount
    {
        totalAmount = quantity * (price - ((discount / 100) * price));

    }
    void Add()   //add function for adding items
    {
        Item::addItem();
        cout << "\nPrice: ";
        cin >> price;
        cout << "Quantity: ";
        cin >> quantity;
        cout << "Total Price of Items: " << (quantity * price);  //calculating
total price if items
        cout << "\nDiscount Percent: ";
        cin >> discount;
        calCulate();    //calling calculate function
        fout.write((char*)&amount, sizeof(amount));  //writing in file
        fout.close();   //file close
    }
    void showDetails()   //show function for showing details of items
    {
        fin.open("SuperMarketItemsStore.dat", ios::binary);   //file opening in
binary mode
        fin.read((char*)&amount, sizeof(amount));    //reading from file
        Item::showItem();   //calling showitem function of class Item
```



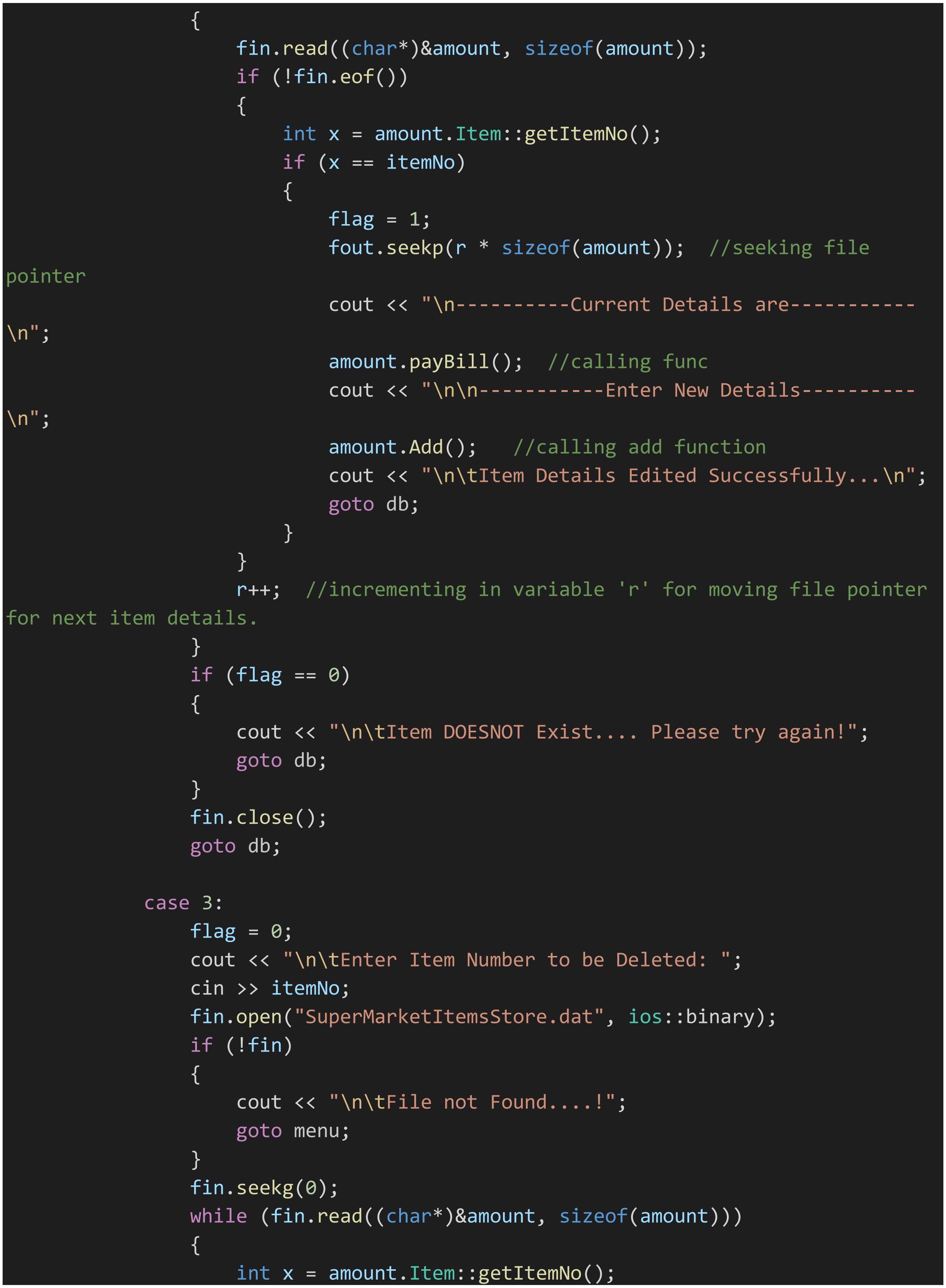
```
        fin.close();
    }
    void report()   //report function for displaying Bill Report of Items
    {
        cout << "   " << itemNo << "     " << setw(10) << name << "  " << 
setw(10) << price << "  " << setw(10) << quantity << "      " << setw(10) << 
discount << "  " << setw(10) << " " << totalAmount << "  " << endl;
    }
    void payBill()  //function for bill Paying and showing details
    {
        showDetails();
        cout << "\n\tPRICE                          :" << price;
        cout << "\n\tQUANTITY                       :" << quantity;
        cout << "\n\tTOTAL AMOUNT                   :" << (quantity * price);
        cout << "\n\tDISCOUNT PERCENTAGE            :" << discount;
        cout << "\n\tNET AMOUNT (Discounted)        :" << totalAmount;
        cout << "\n----------------------------------------------------\n\n";
    }
} amount; //object  of class Amount

int main()  //start main
{
top:                 //top part of main
    string admin;
    int password;
    cout << "\t\t\t\t\t\t------------------------------------------\n";
    cout << "\t\t\t\t\t\t------------------------------------------\n";
    cout << "\t\t\t\t\t\tWelcome to SUPER MARKET MANAGEMENT SYSTEM\n";
    cout << "\t\t\t\t\t\t------------------------------------------\n";
    cout << "\t\t\t\t\t\t------------------------------------------\n";
    cout << "\t\t\t\t\t\t      Developed by Abdullah and Saad\n\n\n\n\n";
    cout << "\t\t\t\t\t\t    ________________________________\n";
    cout << "\t\t\t\t\t\t    |                              |\n";
    cout << "\t\t\t\t\t\t    |                              |\n";
    cout << "\t\t\t\t\t\t    |     .ENTER LOGIN DETAILS.    |\n";
    cout << "\t\t\t\t\t\t    |                              |\n";
    cout << "\t\t\t\t\t\t    |______________________________|\n";
    cout << "\n\t\t\t\t\t\t    Username: ";
    cin >> admin;         //input username and password from user
    cout << "\n\t\t\t\t\t\t    Password: ";
    cin >> password;
    if (admin == "admin" && password == 1234)  //if statement for program start
    {
        fstream tmp("temp.dat", ios::binary | ios::out);    //creating a 
temporary file in binary and out mode.
```



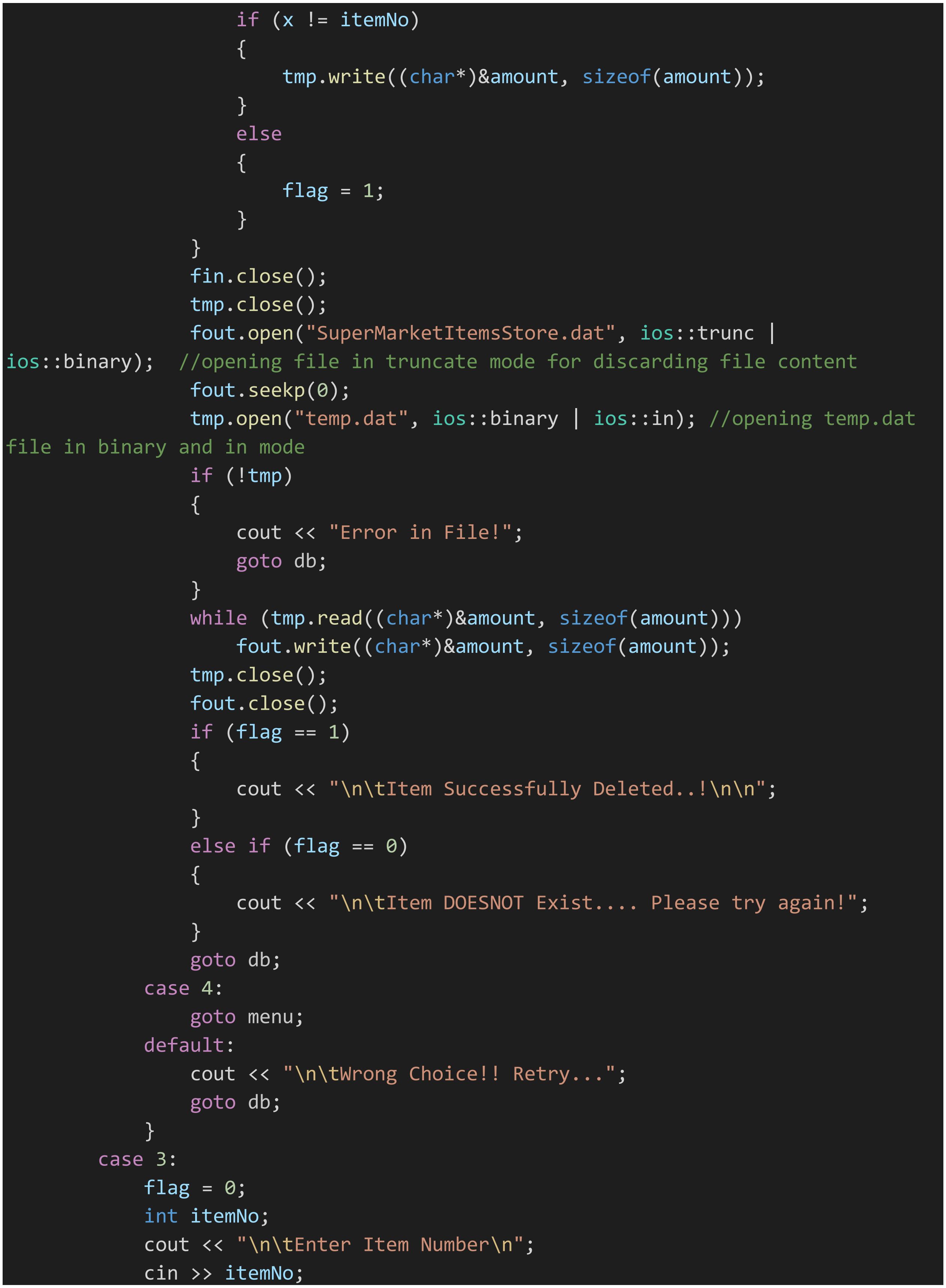
```
    menu:               //menu part
        cout << "\n----------SUPER MARKET BILLING-----------\n";
        cout << "\n\t1.Bill Report.\n";
        cout << "\n\t2.Add/Remove or Edit Item.\n";
        cout << "\n\t3.Show Item Details\n";
        cout << "\n\t4.Exit\n";
        cout << "\n\tPlease Enter the Required Option...";
        int choice, ff;
        float grandTotal;
        cin >> choice;
        switch (choice)   //switch statement
        {
        case 1:
        ss:    //part ss for case 1
            cout << "\n----------BILL REPORT-----------\n";
            cout << "\n\t1.All Items.";
            cout << "\n\t2.Back to Main Menu";
            cout << "\n\tPlease Enter the Required Option...";
            int choice1;
            cin >> choice1;
            if (choice1 == 1)
            {
                cout << "\n-----------------------------------------------BILL DETAILS----------------------------------------------------------\n";
                cout << "\nItem No.       " << setw(10) << "Name         " << setw(10) << "Price       " << setw(10) << "Quantity       " << setw(10) << "Discount        " << setw(10) << "Net Amount" << endl;
                fin.open("SuperMarketItemsStore.dat", ios::binary);   //creating file in binary mode named SuperMarketItemstore.dat.
                if (!fin)
                {
                    cout << "File not Found!!...";
                    goto menu;
                }
                fin.seekg(0);   //seek function for file pointer
                grandTotal = 0;
                while (!fin.eof())
                {
                    fin.read((char*)&amount, sizeof(amount));   //reading data
                    if (!fin.eof())
                    {
                        amount.report();
                        grandTotal += amount.getTotalamount();  //calculating total amount
                        ff = 0;
```



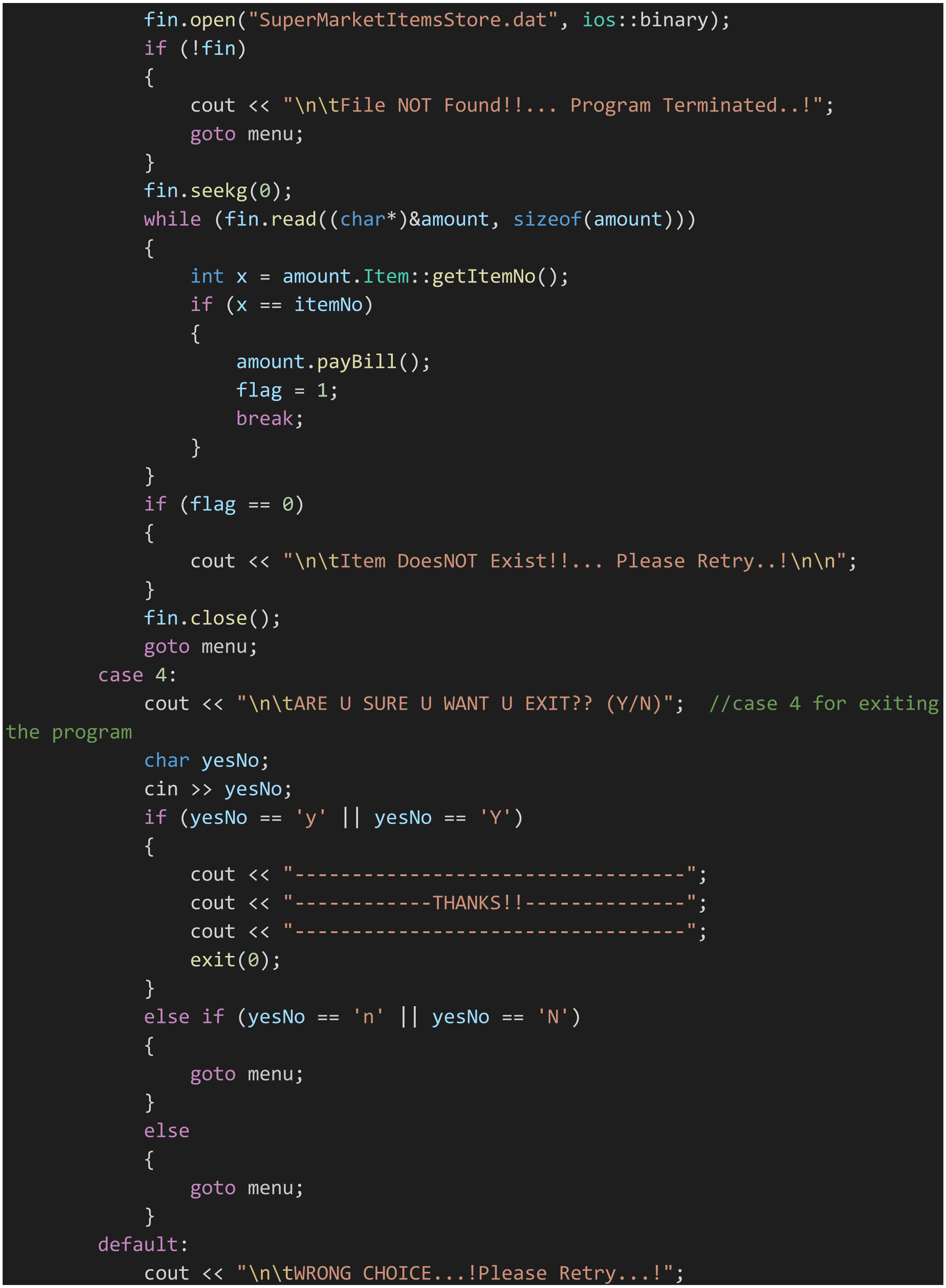
```
                    }
                    if (ff != 0)
                        grandTotal = 0;
                }
                cout << "\n\n\tGrand Total= " << grandTotal << "\n\n";
                fin.close();
            }
            if (choice1 == 2)
            {
                goto menu;  //jump statement to jump to menu part
            }
            goto ss;   //jump to ss part
        case 2:
        db:        //part db
            cout << "\n----------BILL EDITOR-----------\n";
            cout << "\n\t1.Add Item Details.\n";
            cout << "\n\t2.Edit Item Details\n";
            cout << "\n\t3.Delete Item Details\n";
            cout << "\n\t4.Back to the Main Menu\n";
            cout << "\n\tPlease Enter the Required Option...";
            int choice2;
            cin >> choice2;
            switch (choice2)  //switch statement
            {
            case 1:
                fout.open("SuperMarketItemsStore.dat", ios::binary |
ios::app);  //file opening in binary or append mode
                amount.Add();  //add funcation calling for writing data in file.
                cout << "\n\tItem Added Successfully...!\n\n";
                goto db;   //jump statement to goto db part
            case 2:
                int itemNo;
                flag = 0;
                cout << "\n\tEnter Item No. to be Edited...";
                cin >> itemNo;
                fin.open("SuperMarketItemsStore.dat", ios::binary);
                fout.open("SuperMarketItemsStore.dat", ios::binary | ios::app);
                if (!fin)
                {
                    cout << "\n\tFile not Found....!";
                    goto menu;
                }
                fin.seekg(0);
                r = 0;
                while (!fin.eof())
```



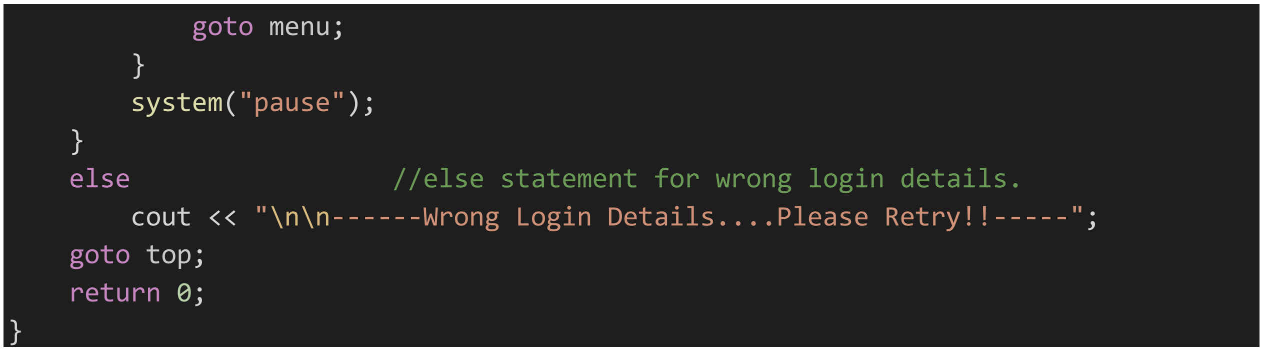
```
                {
                    fin.read((char*)&amount, sizeof(amount));
                    if (!fin.eof())
                    {
                        int x = amount.Item::getItemNo();
                        if (x == itemNo)
                        {
                            flag = 1;
                            fout.seekp(r * sizeof(amount));  //seeking file 
pointer
                            cout << "\n----------Current Details are-----------
\n";
                            amount.payBill();  //calling func
                            cout << "\n\n-----------Enter New Details----------
\n";
                            amount.Add();   //calling add function
                            cout << "\n\tItem Details Edited Successfully...\n";
                            goto db;
                        }
                    }
                    r++;  //incrementing in variable 'r' for moving file pointer 
for next item details.
                }
                if (flag == 0)
                {
                    cout << "\n\tItem DOESNOT Exist.... Please try again!";
                    goto db;
                }
                fin.close();
                goto db;

            case 3:
                flag = 0;
                cout << "\n\tEnter Item Number to be Deleted: ";
                cin >> itemNo;
                fin.open("SuperMarketItemsStore.dat", ios::binary);
                if (!fin)
                {
                    cout << "\n\tFile not Found....!";
                    goto menu;
                }
                fin.seekg(0);
                while (fin.read((char*)&amount, sizeof(amount)))
                {
                    int x = amount.Item::getItemNo();
```



```
                    if (x != itemNo)
                    {
                        tmp.write((char*)&amount, sizeof(amount));
                    }
                    else
                    {
                        flag = 1;
                    }
                }
                fin.close();
                tmp.close();
                fout.open("SuperMarketItemsStore.dat", ios::trunc |
ios::binary);  //opening file in truncate mode for discarding file content
                fout.seekp(0);
                tmp.open("temp.dat", ios::binary | ios::in); //opening temp.dat
file in binary and in mode
                if (!tmp)
                {
                    cout << "Error in File!";
                    goto db;
                }
                while (tmp.read((char*)&amount, sizeof(amount)))
                    fout.write((char*)&amount, sizeof(amount));
                tmp.close();
                fout.close();
                if (flag == 1)
                {
                    cout << "\n\tItem Successfully Deleted..!\n\n";
                }
                else if (flag == 0)
                {
                    cout << "\n\tItem DOESNOT Exist.... Please try again!";
                }
                goto db;
            case 4:
                goto menu;
            default:
                cout << "\n\tWrong Choice!! Retry...";
                goto db;
            }
        case 3:
            flag = 0;
            int itemNo;
            cout << "\n\tEnter Item Number\n";
            cin >> itemNo;
```

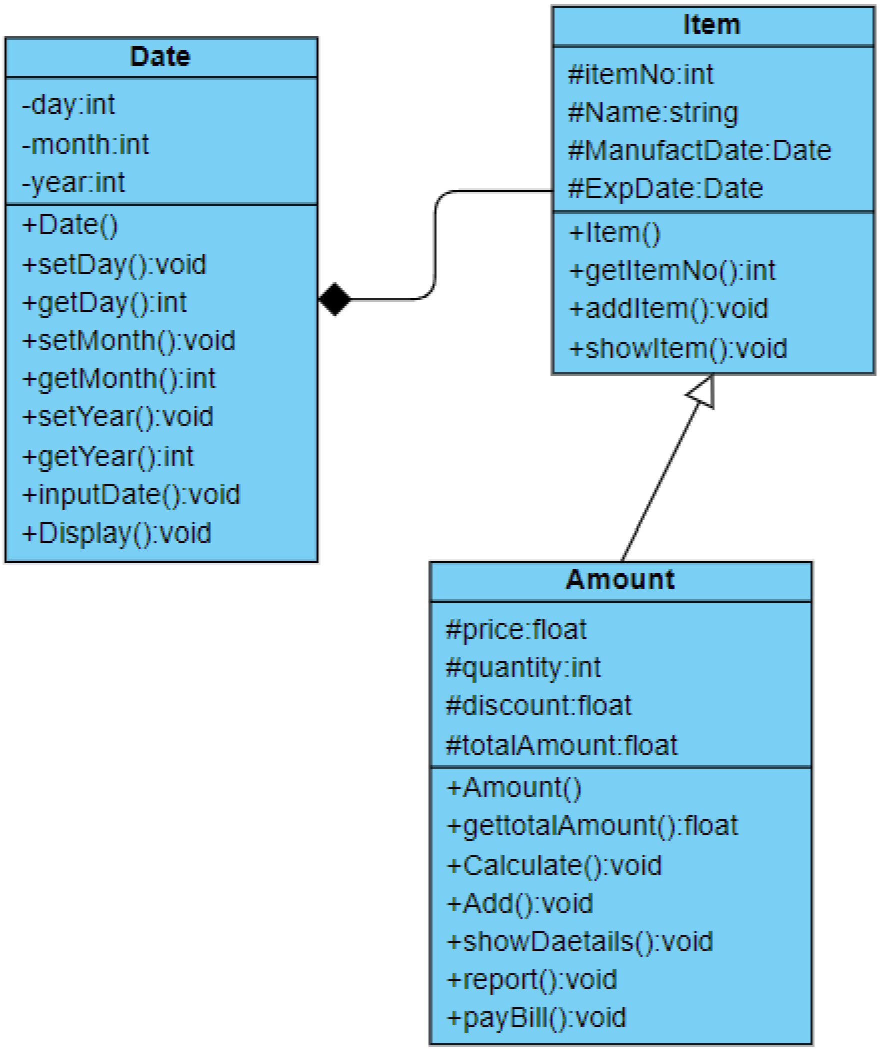


```
            fin.open("SuperMarketItemsStore.dat", ios::binary);
            if (!fin)
            {
                cout << "\n\tFile NOT Found!!... Program Terminated..!";
                goto menu;
            }
            fin.seekg(0);
            while (fin.read((char*)&amount, sizeof(amount)))
            {
                int x = amount.Item::getItemNo();
                if (x == itemNo)
                {
                    amount.payBill();
                    flag = 1;
                    break;
                }
            }
            if (flag == 0)
            {
                cout << "\n\tItem DoesNOT Exist!!... Please Retry..!\n\n";
            }
            fin.close();
            goto menu;
        case 4:
            cout << "\n\tARE U SURE U WANT U EXIT?? (Y/N)";  //case 4 for exiting
the program
            char yesNo;
            cin >> yesNo;
            if (yesNo == 'y' || yesNo == 'Y')
            {
                cout << "----------------------------------";
                cout << "------------THANKS!!--------------";
                cout << "----------------------------------";
                exit(0);
            }
            else if (yesNo == 'n' || yesNo == 'N')
            {
                goto menu;
            }
            else
            {
                goto menu;
            }
        default:
            cout << "\n\tWRONG CHOICE...!Please Retry...!";
```



```
            goto menu;
        }
        system("pause");
    }
    else                     //else statement for wrong login details.
        cout << "\n\n------Wrong Login Details....Please Retry!!-----";
    goto top;
    return 0;
}
```

Class Diagram:



Thank You...