****

**Final Semester project**

**Object oriented Prpgramming**

**Submitted by:**

Humza Fazal Abbasi (01-134212-061)

Abu bakr Siddique (01-134212-011)

BS (CS) - 2C

**Dated: 22 June, 2022**

**Online Store Management System**

\****A program that allows users to shop products and retrieve information about various products of a IT company***.

ABOUT THE PROJECT:

The user will be able make an account, research and learn about various products. These products will be split into many categories. A shopping system will also be implemented and the user will be presented with a bill. There will also be a section of the program dedicated to staff that can login with their exclusive accounts and be able to add, edit or delete information regarding various products.

CONTENT OF THE PROJECT:

The project will include almost all the concepts discussed so far in OOP lectures such as classes, constructors, encapsulation, inheritance, composition etc. the program will be made with ADT and UML diagram in mind. By the end of the project, we believe that all our concepts will be cleared and no doubts/confusion will remain.

Header files:

<iostream>

<fstream>

<string>

<conio.h>

<iomanip>

<windows.h>Loading Animation.

ADT (.h file):

#include <string>

#pragma once

int color\_code =1;

void system\_color(int color);

class animation

{

public :

void greeting();

void load\_animation();

void loading();

};

class error

{

    private :

    int e;

    public:

    error();

    error(int e);

    int display();

};

class Specs

{

    private:

    protected:

        std::string IDFile;

        std::string ProcessorFile;

        std::string MemoryFile;

        std::string StorageFile;

        int PriceFile;

        int StockFile;

    public:

    void setSpecs(std::string typeFile,std::string cameraFile, std::string simFile, std::string graphicsFile, std::string colorFile, std::string weightFile, std::string sizeFile);

    void getProducts();

    void searchproducts(std::string ID);

    void deleteproducts(std::string ID);

    void buyproducts(std::string ID);

};

class Desktop : public Specs

{

    protected:

    std::string weightFile;

    std::string sizeFile;

    std::string empty = "N/A";

    std::string type = "Desktop";

    public:

    void setDesktop();

};

class Desktop\_Gaming : public Desktop

{

    private:

    public:

    void setDesktop\_Gaming();

};

class ProArt : public Desktop

{

    private:

    public:

    void setProArt();

};

class Mobile : public Specs

{

    protected:

    std::string camera;

    std::string sim;

    std::string empty= "N/A";

    std::string type= "Mobile";

    public:

    void setMobile();

};

class ROG\_phone : public Mobile

{

    private:

    public:

    void setROG\_phone();

};

class Zen\_phone : public Mobile

{

    private:

    public:

    void setZen();

};

class Laptop : public Specs

{

    protected:

    std::string graphics;

    std::string color;

    std::string empty= "N/A";

    std::string type = "Laptop";

    public:

    void setLaptop();

};

class For\_Student : public Laptop

{

    protected:

    public:

    void setFor\_Student();

};

class Vivobook : public For\_Student

{

    private:

    public:

    void setVivobook();

};

class Chromebook : public For\_Student

{

    private:

    public:

    void setChromebook();

};

class For\_Gaming : public Laptop

{

    protected:

    public:

    void setFor\_Gaming();

};

class ROG : public For\_Gaming

{

    private:

    public:

    void setROG();

};

class TUF : public For\_Gaming

{

    private:

    public:

    void setTUF();

};

class Webpage

{

    private:

    Specs S;

    Desktop\_Gaming Dg;

    ProArt P;

    ROG\_phone Rp;

    Zen\_phone Zp;

    Vivobook V;

    Chromebook C;

    ROG R;

    TUF T;

    std::string str;

    int c;

    animation a;

    public:

    void display();

};

class cwebpage

{

private:

    int c;

    Specs s;

    std::string str;

    animation a;

public:

    void cdisplay();

};

class Asus\_Account

{

    protected:

    std::string username;

    std::string password;

    std::string username\_from\_file;

    std::string password\_from\_file;

    char choice;

    public:

    void ForgotPassword();

};

class RegisterAccount : public Asus\_Account

{

    private:

    public:

    void register\_account();

};

class Specs;

class LoginAccount : public Asus\_Account

{

    private:

    cwebpage c;

    std::string staff= "staff";

    int stafflen=5;

    int temp;

    animation a;

    public:

    void login\_account();

};

class Intro

{

private:

RegisterAccount R;

LoginAccount L;

Asus\_Account FP;

int choice;

char ch;

animation a;

public:

Intro();

};

int main();

Source code (.cpp):

#include <iostream>

#include <fstream>

#include <string>

#include <vector>

#include <conio.h>

#include <iomanip>

#include <windows.h>

using namespace std;

#include "code10.h"

void system\_color(int color)

{

    HANDLE hConsole = GetStdHandle(STD\_OUTPUT\_HANDLE);

    SetConsoleTextAttribute(hConsole, color);

}

void animation:: greeting(){

    char x = 178  ;

    for(int i = 0 ; i < 50 ; i++)

    {

        cout << x  << x <<  x  ;

        cout << x   << x <<  x  ;

        Sleep(5);

    }

    cout << " \t\t\t Welcome to ASUS " << endl ;

    cout << " \t\t\t Welcome to ASUS " << endl ;

    cout << " \t\t\t Welcome to ASUS " << endl ;

    for(int i = 0 ; i < 50 ; i++)

    {

        cout << x  << x <<  x  ;

        cout << x   << x <<  x  ;

        Sleep(5);

    }

}

void animation:: load\_animation(){

    system\_color(color\_code);

    color\_code++;

    greeting(  );

    system\_color(color\_code);

    color\_code++;

    greeting(  );

    system\_color(color\_code);

    color\_code++;

    greeting(  );

    system\_color(color\_code);

    system\_color(color\_code);

    color\_code++;

    greeting(  );

    system\_color(color\_code);

    color\_code++;

    greeting(  );

}

void animation :: loading(){

    system\_color(color\_code);

    cout << " \n\n\n\n\n\n\n\n\n\n\n\n\n\n\n ";

    cout << "\t\t\t\t\t\t\tLOADING Please wait ..... : \n\n";

char x = 178;

    for(int i = 0 ; i < 138 ; i++)

    {

        cout << x ;

        Sleep(2);

    }

}

void Specs :: setSpecs(string typeFile,string cameraFile, string simFile, string graphicsFile, string colorFile, string weightFile, string sizeFile)

{

    cout<<"\nEnter ID (3 digits): ";

    cin>>IDFile;

    cout<<"\nEnter processor: ";

    cin>>ProcessorFile;

    cout<<"\nEnter memory: ";

    cin>>MemoryFile;

    cout<<"\nEnter storage: ";

    cin>>StorageFile;

    cout<<"\nEnter Price: ";

    cin>>PriceFile;

    cout << "\nEnter Stock: ";

    cin>>StockFile;

    ofstream write(IDFile+".txt");

    write <<"Product ID: "<<IDFile <<endl;

    write <<"Product Type: "<<typeFile<<endl;

    write <<"Processor: "<<ProcessorFile <<endl;

    write <<"Memory: "<<MemoryFile <<endl;

    write <<"Storage: "<<StorageFile <<endl;

    write <<"Camera: "<<cameraFile<<endl;

    write <<"SIM: "<<simFile<<endl;

    write <<"Graphics: "<<graphicsFile<<endl;

    write <<"Color: "<<colorFile<<endl;

    write <<"Weight: "<<weightFile<<endl;

    write <<"Size: "<<sizeFile<<endl;

    write <<"Price: "<<PriceFile <<endl;

    write <<"Stock: "<<StockFile <<endl;

    write.close();

    ofstream allwrite("listofallproducts.txt", ios :: app);

    allwrite <<IDFile <<"\t"<<typeFile<<"\t"<<ProcessorFile<<"\t"<<"    "<<MemoryFile<<"\t"<<"  "<<StorageFile<<"\t"<<cameraFile<<"\t"<<simFile<<"\t"<<graphicsFile<<"\t"<<colorFile<<"\t"<<weightFile<<"\t"<<sizeFile<<"\t"<<StockFile<<"\t"<<PriceFile<<endl;

    allwrite <<endl;

    allwrite.close();

    fstream price(IDFile+"price.txt",ios::app);

    price << PriceFile;

    price.close();

}

void Specs :: getProducts()

    {

        system("cls");

        cout << "Current Products Available"<<endl;

        cout << "ID\t|Type\t|Processor\t|Memory\t|Storage\t|Price\t|Stock\t|Camera\t|SIM\t|Graphics\t|Color\t|Weight\t|Size "<<endl;

        cout << endl;

        string line;

        ifstream read("listofallproducts.txt");

        while(getline(read, line)){

            cout << line<<endl;

        }

        read.close();

    }

void Specs :: searchproducts(string ID)

    {

        string line;

        cout << endl;

        ifstream read(ID+".txt");

        while(getline(read,line)){

        cout << line<<endl;

        }

        read.close();

        system("pause");

    }

void Specs :: deleteproducts(string ID)

    {

        string temp = ID;

        temp += ".txt";

        const char\* c = temp.c\_str();

        string line;

        ofstream write;

        write.open("tempfile.txt");

        ifstream read("listofallproducts.txt");

        while(getline(read,line)){

            if(ID!=line.substr(0,3))

            {

                write<<line<<endl;

            }

        }

        read.close();

        write.close();

        remove("listofallproducts.txt");

        rename("tempfile.txt","listofallproducts.txt");

        remove(c);

        system("pause");

    }

void Specs :: buyproducts(string ID)

{

    string line,p;

        char a;

        cout <<":Product You want too buy:" <<endl;

        ifstream read(ID + ".txt");

        while (getline(read, line)) {

            cout << line << endl;

        }

        read.close();

        ifstream red(ID + "price.txt");

        while (getline(red, p)) {

            cout << "Your bill is:"<< p << endl;

        }

        red.close();

        cout << "Are you sure you want to buy this product(y/n) " << endl;

        cin >> a;

        if (a == 'y')

        {

            cout << "Purchase successfull!!" << endl;

        }

        else

        {

            exit(0);

        }

        system("pause");

}

void Desktop :: setDesktop()

{

                cout << "Enter weight: ";

                cin>>weightFile;

                cout<< "\nEnter size: ";

                cin>>sizeFile;

                Specs : setSpecs(type,empty,empty,empty,empty,weightFile,sizeFile);

}

void Desktop\_Gaming :: setDesktop\_Gaming()

{

        Desktop : setDesktop();

}

void ProArt :: setProArt()

{

        Desktop : setDesktop();

}

void Mobile :: setMobile()

{

                    cout << "Enter Camera detail: ";

                    cin>>camera;

                    cout << "\nEnter SIM detail: ";

                    cin>>sim;

                    Specs : setSpecs(type,camera,sim,empty,empty,empty,empty);

}

void ROG\_phone :: setROG\_phone()

{

        Mobile : setMobile();

}

void Zen\_phone :: setZen(){

        Mobile : setMobile();

}

void Laptop :: setLaptop()

{

                cout << "Enter Graphics detail:";

                cin>>graphics;

                cout << "\nEnter Color: ";

                cin>>color;

                Specs: setSpecs(type,empty,empty,graphics,color,empty,empty);

}

void For\_Student :: setFor\_Student(){

        Laptop: setLaptop();

}

void Vivobook :: setVivobook(){

        For\_Student : setFor\_Student();

}

void Chromebook :: setChromebook(){

        For\_Student : setFor\_Student();

}

void For\_Gaming :: setFor\_Gaming(){

        Laptop: setLaptop();

}

void ROG :: setROG(){

        For\_Gaming : setFor\_Gaming();

}

void TUF:: setTUF(){

        For\_Gaming : setFor\_Gaming();

}

void Webpage :: display()

    {

        system("cls");

        cout<<"Welcome, "<<endl;

        cout<<"1. view products"<<endl;

        cout<<"2. add products"<<endl;

        cout<<"3. delete products"<<endl;

        cout<<"4. search products"<<endl;

        cout<<"5. exit"<<endl;

        cin>>c;

        switch(c)

        {

            case 1:

            system("cls");

            S.getProducts();

            break;

            case 2:

            system("cls");

            cout << "choose product type to add"<<endl;

            cout << "1. Laptop"<<endl;

            cout << "2. Desktop"<<endl;

            cout << "3. Mobile"<<endl;

            cin>>c;

            switch(c)

            {

                case 1:

                system("cls");

                cout << "choose laptop series"<<endl;

                cout << "1. For Gaming"<<endl;

                cout << "2. For Student"<<endl;

                cin>>c;

                switch(c)

                {

                    case 1:

                    system("cls");

                    cout << "choose gaming model"<<endl;

                    cout << "1. ROG"<<endl;

                    cout << "2. TUF"<<endl;

                    cin>>c;

                    switch(c)

                    {

                        case 1:

                        system("cls");

                        a.loading();

                        system("cls");

                        R.setROG();

                        break;

                        case 2:

                        system("cls");

                        a.loading();

                        system("cls");

                        T.setTUF();

                        break;

                        default:

                        exit(0);

                    }

                    break;

                    case 2:

                    system("cls");

                    cout << "choose student model"<<endl;

                    cout << "1. Vivobook"<<endl;

                    cout << "2. Chrome book"<<endl;

                    cin>>c;

                    switch(c)

                    {

                        case 1:

                        system("cls");

                        a.loading();

                        system("cls");

                        V.setVivobook();

                        break;

                        case 2:

                        system("cls");

                        a.loading();

                        system ("cls");

                        C.setChromebook();

                        break;

                        default:

                        exit(0);

                    }

                }

                break;

                case 2:

                system("cls");

                cout << "choose desktop series"<<endl;

                cout << "1. Desktop Gaming"<<endl;

                cout << "2. ProArt"<<endl;

                cin>>c;

                switch(c)

                {

                    case 1:

                    system("cls");

                    a.loading();

                    system("cls");

                    Dg.setDesktop\_Gaming();

                    break;

                    case 2:

                    system("cls");

                    a.loading();

                    system("cls");

                    P.setProArt();

                    break;

                    default:

                    exit(0);

                }

                break;

                case 3:

                system("cls");

                cout << "choose Mobile model"<<endl;

                cout << "1. ROG phone"<<endl;

                cout << "2. Zen phone"<<endl;

                cin>>c;

                switch(c)

                {

                    case 1:

                    system("cls");

                    a.loading();

                    system("cls");

                    Rp.setROG\_phone();

                    break;

                    case 2:

                    system("cls");

                    a.loading();

                    system("cls");

                    Zp.setZen();

                    break;

                    default:

                    exit(0);

                }

            }

            break;

            case 3:

                system("cls");

                cin.ignore();

                cout << "Enter the ID: "<<endl;

                getline(cin,str);

                system("cls");

                a.loading();

                system("cls");

                S.deleteproducts(str);

            break;

            case 4:

                system("cls");

                cin.ignore();

                cout << "Enter the ID: "<<endl;

                getline(cin,str);

                system("cls");

                a.loading();

                system("cls");

                S.searchproducts(str);

                break;

            case 5:

                exit(0);

                break;

            default:

            exit(0);

        }

        cout<<"Try again?(1)";

        cin>>c;

        if(c == 1){

            a.loading();

            display();

        }

        else

            exit(0);

}

void cwebpage :: cdisplay()

    {

        system("cls");

        cout<<"Welcome, "<<endl;

        cout<<"1. view products"<<endl;

        cout<<"2. search products"<<endl;

        cout<<"3. buy products"<<endl;

        cout<<"4. exit"<<endl;

        cin>>c;

        switch(c)

        {

            case 1:

                system("cls");

                a.loading();

                system("cls");

                s.getProducts();

            break;

            case 2:

                system("cls");

                a.loading();

                system("cls");

                cin.ignore();

                cout << "Enter the ID: "<<endl;

                getline(cin,str);

                s.searchproducts(str);

                break;

            case 3:

                system("cls");

                a.loading();

                system("cls");

                cin.ignore();

                cout << "Enter the ID: "<<endl;

                getline(cin,str);

                s.buyproducts(str);

                break;

            case 4:

                exit(0);

                break;

            default:

            exit(0);

        }

    }

void Asus\_Account :: ForgotPassword()

    {

        system("cls");

        cout << "FORGOT PASSWORD PAGE" << endl;

        cout << "Enter username: "<<endl;

        cin >> username;

        ifstream read(username+".txt");

        getline(read,username\_from\_file);

        getline(read,password\_from\_file);

        if(username\_from\_file==username)

        {

            cout << "Here is your password, "<<username << endl;

            cout << password\_from\_file << endl;

        }

        else

        {

            cout << "Account does not exist";

        }

    }

void RegisterAccount :: register\_account()

    {

        system("cls");

        cout << "REGISTER PAGE"<<endl;

        cout << "Enter username: ";

        cin >> username;

        cout << "\nEnter password: ";

        cin >> password;

        ofstream write(username+".txt");// "abubakr"+".txt"= abubakr.txt

        write<<username<<endl;

        write<<password<<endl;

        write.close();

        cout << "You have been successfully registered" << endl;

        main();

    }

class Specs;

void LoginAccount :: login\_account()

    {

        system("cls");

        cout << "LOGIN PAGE"<< endl;

        cout << "Enter username: "<< endl;

        cin >> username;

        cout << "Enter password: "<< endl;

        cin >> password;

        ifstream read(username+".txt");

        getline(read,username\_from\_file);

        getline(read,password\_from\_file);

        if (username\_from\_file==username && password\_from\_file==password)

        {

            temp = username.size()-stafflen;

            if (username.substr(temp)==staff)

            {

                a.loading();

                system("cls");

                Webpage w;

                w.display();

            }

            else

            {

                a.loading();

                system("cls");

                c.cdisplay();

            }

        }

        else

        {

            cout << "Wrong username/password, forgot password? (y/n): "<< endl;

            cin >> choice;

            if (choice=='y')

            {

                Asus\_Account:ForgotPassword();

            }

            else

                login\_account();

        }

    }

Intro::Intro()

{

    system("cls");

    cout<<"Welcome to ASUS Website, please enter in your credentials to continue browsing through the website"<<endl;

    cout << "\t\t\t\t\t\t\t1. Login \n\n";

    cout << "\t\t\t\t\t\t\t2. Register \n\n";

    cout << "\t\t\t\t\t\t\t3. Exit \n\n";

    cout << "\t\t\t\t\t\t\tEnter your choice : ";

    cin>>choice;

    system("cls");

    a.loading();

    system("cls");

    switch (choice)

    {

    case 1:

        L.login\_account();

        break;

    case 2:

        R.register\_account();

        break;

    case 3:

        exit(0);

        break;

    default:

        cout << "Invalid, would you like to try again? (y/n)" << endl;

        cin>>ch;

        if (ch=='y')

        {

            Intro();

        }

        else

            exit(0);

        break;

    }

}

int main()

{

    try

    {

        system("cls");

        animation object;

        object.load\_animation();

        system("cls");

        Intro i;

    }

    catch (error e)

    {

        system("cls");

        cout << "Invalid choice:Try Again " <<endl<<endl;

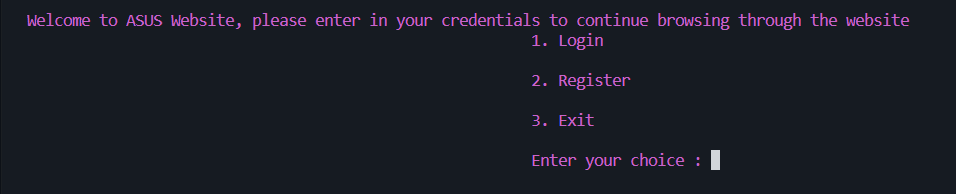
    }

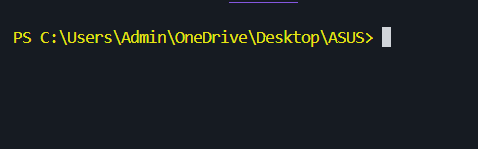
    \_getch();

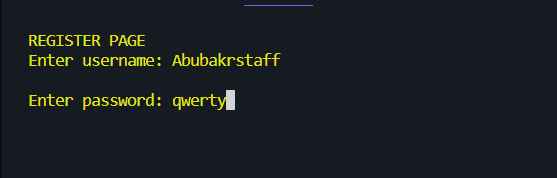
    return 0;

}

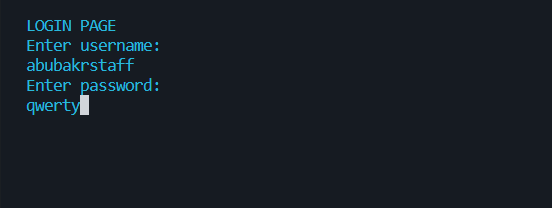
Output screen:



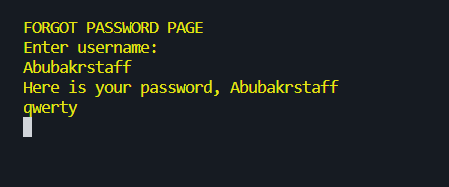
(3. Exit):  


(2.Register):  


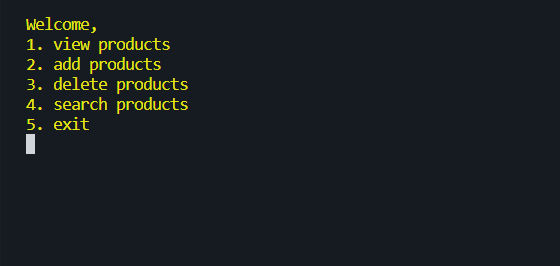
(1. Login):

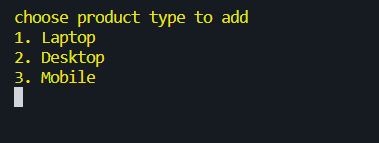


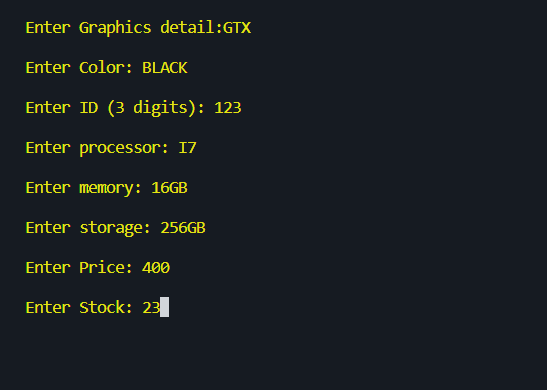
(IF user forgot their password, the program will prompt them to enter their username as verification to get their password)



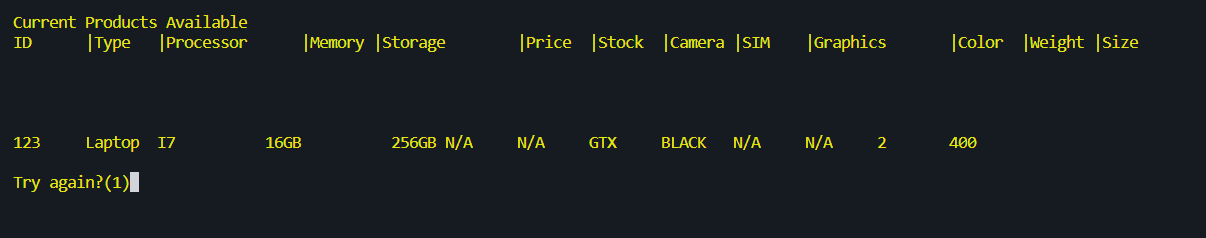
(IF logged in as STAFF member (you are a staff member if your username ends with the keyword “staff”. For example: “abubakrstaff” in the above picture) you will be prompted with this screen).

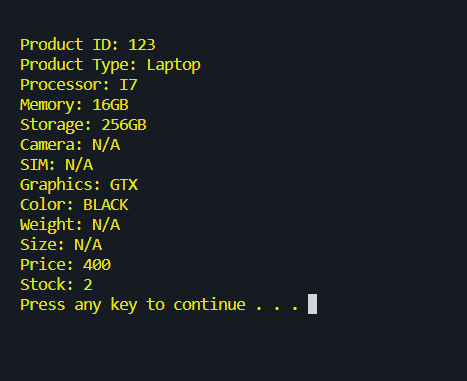


(2. ADD products):  




(1.View Products):



(4.Search products (it will prompt you to enter ID of product)):  


(3. Delete Products): will delete all info of that products when user enters its ID.

(IF user Logged in as a customer (i.e. they don’t have “staff” at the end of user name) they will only be able to buy, view and search products)