Project Description

‘Store Management System’ is a program developed with ‘c++’ programming language. It is a program that can be used to maintain a shop, specially shop’s selling system. Both the owner of shop and customer can use this software.

When the program is started to run, a welcome note and two main menus are displayed on screen. Two main menus are:

* Admin Panel
* Customer Panel.

In Admin menu, there are some sub-menus such as:

* View product list
* Entry a new product
* Sale
* Sold Product list
* Return to main menu.

Here an admin can maintain his/her products in the shop and can keep track on sell. He/she can add new products and can see sold product list also.

In Customer menu, there is only the product displaying option. Customer can make order by seeing the product list.

Input / Output Domain

* Input:
* Products Information in the shop
  + Product Code
  + Product Name
  + Product Price
  + Product Colour
  + Product Quantity
  + Product Manufacture Company
* Output:
  + All the information that takes as input according to the user’s choice.

Flow Chart

Class Diagram

**Base class:** product

**Members:**

* + product code (Protected)
  + unit price (Protected)
  + product name
  + stock quantity
  + setting & setting function

**Derived class**: foods

**Members**:

* + - flavour
    - nutritional facts (energy value)
    - setting & setting function

**Derived class**: goods

**Members**:

* + colour
  + manufacture company
  + setting & setting function

Challenges Faced

* Sometimes code was not run properly.
* Problem faced with library & header files.
* Sometimes code was not terminated properly.
* Some features of C++ could not be included because of their complexity.
* We had to work hard on this. Had to think and give much time on this.
* Many errors were made but now it is running well.

Future Improvement

* Adding admin password changing option.
* Adding customer’s order option.
* Adding product editing option.
* More user friendly.

Appendix (All Codes)

#include<iostream>

#include<iomanip> //For using setw() function.

#include<cstdlib> //For using exit() function.

using namespace std;

string pass="12345"; //Defining a default password for entering admin panel.

void admin(); //Declaring Admin menu function.

void customer(); //Declaring Customer menu function.

int security(); //Declaring Admin password checking function.

void menu(); //Declaring Menu function.

void display(); //Declaring Product list showing function.

void entry(); //Declaring product entry function.

void sale(); //Declaring product selling function.

void selling\_list(); //Declaring sold product list showing function.

/\*It is base class.\*/

class product

{

protected: //Here, encapsulation is done.

string code;

double price;

public: //Public members.

string name;

int quantity;

void set\_code(string x){code=x;} //Getting & setting value.

string get\_code(){return code;}

void set\_price(double y){price=y;}

double get\_price(){return price;}

void set\_name(string z){name=z;}

string get\_name(){return name;}

void set\_quantity(int w){quantity=w;}

int get\_quantity(){return quantity;}

};

/\*Derived class as public.\*/

class foods:public product

{

public: //Public members.

string flavour;

double n\_facts;

void set\_flavour(string y){flavour=y;} //Getting & setting value.

string get\_flavour(){return flavour;}

void set\_n\_facts(double x){n\_facts=x;}

double get\_n\_facts(){return n\_facts;}

void show(){cout<<left<<setw(15)<<get\_code()<<right<<setw(17)<<get\_name()<<right<<setw(16)<<get\_price()<<right<<setw(20)<<get\_quantity()<<right<<setw(16)<<get\_flavour()<<right<<setw(28)<<get\_n\_facts()<<"\n";} //Display values.

};

/\*Derived class as public.\*/

class goods:public product

{

public: //Public members.

string colour;

string m\_company;

void set\_colour(string x){colour=x;} //Getting & setting value.

string get\_colour(){return colour;}

void set\_m\_company(string y){m\_company=y;}

string get\_m\_company(){return m\_company;}

void show(){cout<<left<<setw(15)<<get\_code()<<right<<setw(17)<<get\_name()<<right<<setw(16)<<get\_price()<<right<<setw(20)<<get\_quantity()<<right<<setw(16)<<get\_colour()<<right<<setw(28)<<get\_m\_company()<<"\n";} //Display values.

} ;

/\*Main function\*/

int main()

{

cout<<"Welcome To The Shop Management System Software...\n"; //Welcome text.

menu(); //Calling menu function.

return 0; //Ending process of program.

}

/\*Global declaration of variable & objects\*/

int i,choice;

foods ob1[20]; //Derived class object.

goods ob2[20]; //Derived class object.

product pd[20]; //Base class object.

/\*Defining Menu function\*/

void menu()

{

cout<<"\nMain Menu:\n"; //Showing menu list.

cout<<"1. Admin Panel\n";

cout<<"2. Customer Panel\n";

cout<<"\nPlease select an option:\t";

cin>>choice; //Taking choice from user.

switch(choice)

{

case 1:

if(security()==1) //Calling security function to check password.

cout<<"\nPassword matched.\n";

admin(); //If password is matched, only then admin function will be executed.

break;

case 2:

customer(); //Calling customer function.

break;

default:

cout<<"\nInvalid selection. Try again:\n";

menu();

break;

}

}

/\*Defining admin function\*/

void admin()

{

cout<<"\nAdmin Menu:\n"; //Showing Admin menu.

cout<<"1. View Product List.\n";

cout<<"2. Entry A New Product.\n";

cout<<"3. Sale.\n";

cout<<"4. Sold Products List.\n";

cout<<"5. Return To Main Menu.\n";

cout<<"\nPlease select an option:\t";

cin>>choice; //Taking choice from user.

switch(choice)

{

case 1:

display(); //Calling display function.

admin(); //Calling admin function. After executing display function, admin function will be executed.

break;

case 2:

entry(); //Calling entry function.

admin(); //Calling admin function. After executing display function, admin function will be executed.

break;

case 3:

sale(); //Calling sale function.

admin(); //Calling admin function. After executing display function, admin function will be executed.

break;

case 4:

selling\_list(); //Calling selling\_list function.

admin(); //Calling admin function. After executing display function, admin function will be executed.

break;

case 5:

menu();

break;

default:

cout<<"\nInvalid Selection.\n";

admin();

break;

}

}

/\*Defining customer function\*/

void customer()

{

cout<<"\nHere Is The Available Product List.\n";

display(); //Calling display function.

exit(0);

}

/\*Defining password checking function\*/

int security()

{

cout<<"\nAdmin selected.\tEnter Password to Enter Admin Panel:\t";

string word;

cin>>word;

if(word==pass) //Matching user given password with default password.

return 1;

else

{

cout<<"\nWrong password. Try again:\n";

menu();

}

}

/\*Defining display/product list showing function\*/

void display()

{

cout<<"\n\n ======================= Product List =======================\n\n";

cout<<"\n"<<left<<setw(15)<<"Product Code"<<right<<setw(17)<<"Product Name"<<right<<setw(16)<<"Unit Price"<<right<<setw(20)<<"Stock Quantity"<<right<<setw(16)<<"Flavour"<<right<<setw(28)<<"Energy (Kcal/100gm)\n\n";

for(i=0;i<20;i++)

{ //For foods list.

if(ob1[i].get\_price()!=0) //Price 0 means, no product in that array. So no need to be printed.

ob1[i].show(); //Calling class function with object for printing product list.

}

cout<<"\n\n\n"<<left<<setw(15)<<"Product Code"<<right<<setw(17)<<"Product Name"<<right<<setw(16)<<"Unit Price"<<right<<setw(20)<<"Stock Quantity"<<right<<setw(16)<<"Colour"<<right<<setw(28)<<"Company\n\n";

for(i=0;i<20;i++)

{ //For goods list.

if(ob2[i].get\_price()!=0) //Price 0 means, no product in that array. So no need to be printed.

ob2[i].show(); //Calling class function with object for printing product list.

}

}

/\*Defining product entry function\*/

void entry()

{

int quantity;

double price,n\_facts;

string code,name,flavour,colour,m\_company;

for(i=i;i<20;i++)

{

cout<<"\nEnter the product code (product code starts with f or g):\t";

cin>>code;

if(code[0]=='f') //Checking whether product type is foods or goods.

{ //Entry foods information.

ob1[i].set\_code(code); //Passing parameters in objects.

cout<<"\nEnter the product name:\t";

cin>>name;

ob1[i].set\_name(name);

cout<<"\nEnter unit price:\t";

cin>>price;

ob1[i].set\_price(price);

cout<<"\nEnter available stock quantity:\t";

cin>>quantity;

ob1[i].set\_quantity(quantity);

cout<<"\nEnter food flavour:\t";

cin>>flavour;

ob1[i].set\_flavour(flavour);

cout<<"\nEnter nutritional facts (Energy in Kcal/100gm):\t";

cin>>n\_facts;

ob1[i].set\_n\_facts(n\_facts);

}

else

{ //Entry goods information.

ob2[i].set\_code(code); //Passing parameters in objects.

cout<<"\nEnter the product name:\t";

cin>>name;

ob2[i].set\_name(name);

cout<<"\nEnter unit price:\t";

cin>>price;

ob2[i].set\_price(price);

cout<<"\nEnter available stock quantity:\t";

cin>>quantity;

ob2[i].set\_quantity(quantity);

cout<<"\nEnter colour:\t";

cin>>colour;

ob2[i].set\_colour(colour);

cout<<"\nEnter manufacture company name:\t";

cin>>m\_company;

ob2[i].set\_m\_company(m\_company);

}

cout<<"\nProduct entry successful.\n\nPress 1 to entry another product or\nPress 0 to return to the admin menu:\t";

cin>>choice;

if(choice==1)

continue; //For continuing product entry.

else if(choice==0)

admin(); //Else going back to Admin menu.

break;

}

}

/\*Defining product selling function\*/

void sale()

{

cout<<"\nEnter the product code to sell:\t";

string sell;

cin>>sell;

for(i=0;i<20;i++)

{

if(ob1[i].get\_price()!=0 || ob2[i].get\_price()!=0) //Price 0 means, no product in that array. So no need to be checked.

{

if(sell==ob1[i].get\_code() || sell==ob2[i].get\_code()) //Matching user given code with existing product code in product list.

{

if(sell[0]=='f') //For foods item sell.

{

cout<<"\nEnter the quantity:\t";

int quantity;

cin>>quantity;

ob1[i].set\_quantity(ob1[i].get\_quantity()-quantity);

cout<<"\nDone.\tTotal price: "<<ob1[i].get\_price()\*quantity<<"Taka.\n";

pd[i].set\_code(ob1[i].get\_code()); //Passing selling item info to the base class with objects for showing sold item list.

pd[i].set\_name(ob1[i].get\_name());

pd[i].set\_quantity(quantity);

pd[i].set\_price(ob1[i].get\_price()\*quantity);

cout<<"\n\nPress 1 to sell another product or\nPress 0 to return to the admin menu:\t";

cin>>choice;

if(choice==1)

continue; //For continuing product sell.

else if(choice==0)

admin(); //Else going back to Admin menu.

break;

}

else //For goods item sell.

{

cout<<"\nEnter the quantity:\t";

int quantity;

cin>>quantity;

ob2[i].set\_quantity(ob2[i].get\_quantity()-quantity);

cout<<"\nDone.\tTotal price: "<<ob2[i].get\_price()\*quantity<<"Taka.\n";

pd[i].set\_code(ob2[i].get\_code()); //Passing selling item info to the base class with objects for showing sold item list.

pd[i].set\_name(ob2[i].get\_name());

pd[i].set\_quantity(quantity);

pd[i].set\_price(ob2[i].get\_price()\*quantity);

cout<<"\n\nPress 1 to sell another product or\nPress 0 to return to the admin menu:\t";

cin>>choice;

if(choice==1)

continue; //For continuing product sell.

else if(choice==0)

admin(); //Else going back to Admin menu.

break;

}

}

}

}

cout<<"\nNo products found with this product code. Try again.\n";

sale(); //If no product matched in the list then again return to the product selling function.

}

/\*Defining sold product list showing function\*/

void selling\_list()

{

cout<<"\n\n\ ======= Here is the sold product list =======\n\n";

cout<<"\n"<<left<<setw(15)<<"Product Code"<<right<<setw(17)<<"Product Name"<<right<<setw(16)<<"Quantity"<<right<<setw(20)<<"Total Taka\n\n";

for(i=0;i<20;i++)

{

if(pd[i].get\_quantity()!=0) //Price 0 means, no product in that array. So no need to be printed.

cout<<left<<setw(15)<<pd[i].get\_code()<<right<<setw(17)<<pd[i].get\_name()<<right<<setw(16)<<pd[i].get\_quantity()<<right<<setw(20)<<pd[i].get\_price()<<"\n";

}

}

The End