Pharmacy Management System in C++ Documentation

Team Members:

Sami AbuTouq, Shaima Ahmad, Raghad Alqammaz, Amr Saeed

Phase One:

Documentation done by Raghad Alqammaz

Project Requirements

A) Medication Class:

This class was done by Sami AbuTouq,

It has:

1- Setters for each class member to check if the values entered by the user are valid to store in the class member or not.
2- Constant Getters for each class member to return the values of each member.
3- One parameterized constructor to store values entered by the user in each class member after validation from the setters.
4- One constant print function to print medication information.

B) Customer Class:

This class was done by Shaima Ahmad and Amr Saeed,

It has:

1- One default constructor to put default values for each of the class members.
2- One parameterized constructor to store values entered by the user in each class member after validation from the setters.

3- One copy constructor to copy the same values of one object to another object.
4- Setters for each class member to check if the values entered by the user are valid to store in the class member or not.
5- Constant Getters for each class member to return the values of each member.
6- Print Function to print the costumer information.

C) In the Main:

The main was done by Sami AbuTouq,

Before the main there is:

1- A printOption function that prints the user interface and asking the user to enter an option for performing operations like adding medications, adding customers, making purchases, displaying available medications, and calculating total sales and profit.
2- An afterCase function that asks the user after every case if there is another option he would like to choose, if yes the function calls the PrintOption function again.

In the main there is:

1- Started by calling the printOption function to print the user interface.
2- A while loop, with a switch inside that consists out of 7 cases, every case deals with a certain condition according to input from the user.

Phase Two:

Documentation done by Raghad Alqammaz.

Project Requirements

A) Modification on the Medication Class:

This was done by Shaima Ahamad,

It included:

1- Adding the (MedicationID) class member of type integer.
2- Adding the (exp) class member of type Date.

B) Date Class:

This class was done by Raghad Alqammaz,

This class has:

1) Private class members (day_, month_, year_) of type integer.
2) Setters for each class member to check if the values entered by the user are valid to store in the class member or not.
3) Constant getters to return each class member.
4) One defaulted constructor and one copy constructor to copy the same values of one object to another object.
5) A print function to display the class members.

C) Modification on the Costumer Class:

This was done by Shaima Ahamad,

It included:

1-Adding (address) class member of type Address.

3- A constant (`PrintCustomerInfo()`) function to display costumer's information.

D)Address Class:

This class was done by Raghad Alqammaz,

This class has:

1- Private class members (email_, city_ ,streetName_ ,mobileNo_) of type string and (mobileNo_) of type long integer.
2- Setters to put the input of the user in the class members depending on certain conditions.
3- Getters to return the class members.
4- One defaulted constructor and one copy constructor to copy the values of the class members of one object to the class members of another object.
5- A constant (`printDate()`) to display the display the details of the address.

E)Pharmacy Class:

This class was done by Amr Saeed,

This class has:

1- Private class members: (`pharmacyIDCounter_, pharmacyID_, medicationCount_, customerCount_) of type integer, and (Pharmacyname_) of type string, and one array of objects (medications_[50]) of size 50 and type Medication, and one array of objects (customers_[50]) of size 50 and type Customer.`
2- Setter to set the pharmacy name.
3- Constant getters to get the pharmacy name and ID.
4- Adding functions:
   a) `Adding Medication Function`
   b) `Removing Medication By Medication ID Function`
   c) `Removing Medication By Medication Name Function`
   d) `Adding Customer Function`
   e) `Print Medications for a Pharmacy Function`
   f) `Print Customers for a Pharmacy Function`
   g) `Compare 2 Pharmacies By Medication Count Function`
   h) `Compare 2 Pharmacies By Customer Count Function`
   i) `Compute Total Revenue for a Pharmacy Function`
   j) `Compare 2 Pharmacies By Total Revenue Function`

k) Check if the Pharmacy Inventory Full
l) Return the Number of Medications in the Pharmacy
m) Return the Number of Customers in the Pharmacy
n) Print Pharmacy Information
o) Buy a Medicine from a Pharmacy (Make Purchase)
p) Sort Medications by Name
q) Search for Medication by ID
r) Search for Customer by ID :
s) Display Expired Medications for a Pharmacy
t) Remove Expired Medications for a Pharmacy
u) Display Medications Running Low in a Pharmacy
v) Update Customer Information (the function will update the name and the address of the customer and it's ID will stay the same)
w) Update Medication Information

F)Modification on the main:

This was done by Sami AbuTouq,

This included:

1- Adding cases on the switch (8-20), every case deals with a certain condition according to input from the user.
2- Adding functions :
    a) validatePharmacyID()
    b) afterCase()
    c) MakePurchase()
    d) AddCustomer()
    e) AddMedications()
    f) printPharmacyInformation()