**Pharmacy Management System**

**Source Code:**

//Data Structure Laboratory Mini-Project

// -------------------------------

// Header Files

// -------------------------------

#include <iostream>

#include <stdlib.h>

#include <string>

#include <cctype>

#include <cmath>

#include <cstdio>

#include <fstream>

#include<conio.h>

#include <iomanip>

#define max 10

using namespace std;

//the header file

// -------------------------------

// Class Declaration

// -------------------------------

class medicineType //base class

{

public:

void take\_order();//to take\_order

void delete\_order(); //to delete the order

void modify(); //to modify the order

void order\_list(); //to display the order\_list

void daily\_summary(); //to display daily\_summary

void exit(); //function to exit system

medicineType();//constuctor

};

medicineType::medicineType ()

{

} //constructor for class CarType

struct node //constract node

{

int reciept\_number;

string customerName;

string date;

int quantity[10];

string type = {"OTC"};

int x, menu2[10];

double amount[10];

string medicineName[10]={"Probiotics","Vitamin C(500mg)","Acid Free C(500mg)","Women'S Multivate","Marino Tablet","Maxi Cal Tablet",

"Amino Zinc Tablet","Burnex","Fabuloss 5","Royal Propollen"};

double Medicine[10] = {2.00,3.00,1.00,4.00,1.00,5.00,7.00,4.00,3.00,5.00};

double total;

node \*prev;

node \*next;

node \*link;

}\*q, \*temp; //pointer declaration

node \*start\_ptr = NULL;

node \*head = NULL;

node \*last = NULL;

int main() // Main function

{

system("cls");

char r1=201,r2=200,r3=188,r4=187,r5=205,r6=186,r7=220,r8=202,r9=204,r10=203,r11=206,r12=185;

cout<<"\n";

for(int i=0;i<89;i++)

cout<<r7;

cout<<endl;

cout<<"\n "<<r1<<r5<<r5<<r5<<r5<<r5<<r5<<r5<<r5<<r5<<r5<<r4<<" INTERNATIONAL INSTITUTE OF INFORMATION TECHNOLOGY"<<endl;

cout<<" "<<r6<<" I2IT "<<r6<<" Affilated to SPPU\n";

cout<<" "<<r2<<r5<<r5<<r5<<r5<<r5<<r5<<r5<<r5<<r5<<r5<<r3<<" Approved to AICTE, RECOGNIZED BY DTE, Govt.\n";

cout<<" Accredited with Grade B++ by NAAC.\n";

for(int i=0;i<89;i++)

cout<<r7;

cout<<"\n\n\n\n\t\t\t\tCreated By:"<<endl;

cout<<"\t\t\tMahesh Gaikwad (SC27)"<<endl;

cout<<"\t\t\tPrathamesh Bhise (SC44)"<<endl;

cout<<"\t\t\tPranav Gaikwad (SC41)"<<endl;

getch();

// system("COLOR 79"); //Color to change background

medicineType medicine;

int menu;

do

{

system("cls");

cout<<"\t\t\tPharmacy Management System - UNIMAS\n";

cout<<"\t\t==================================================\n\n";

cout<<"\t\t--------------------------------------------------\n";

cout<<"\t\t||\t1. Take new Medicine order \t\t ||\n";

cout<<"\t\t||\t2. delete latest Medicine order\t\t ||\n";

cout<<"\t\t||\t3. Modify Order List \t\t\t ||\n";

cout<<"\t\t||\t4. Print the Reciept and Make Payment \t ||\n";

cout<<"\t\t||\t5. Daily Summary of total Sale \t\t ||\n";

cout<<"\t\t||\t6. Exit\t\t\t\t\t ||\n";

cout<<"\t\t--------------------------------------------------\n";

cout<<"Enter choice: ";

cin>>menu;

switch (menu)

{

case 1:

{

medicine.take\_order(); //function add

break;

} //end case 1

case 2:

{

medicine.delete\_order(); //function delete

system("PAUSE");

break;

} //end case 2

case 3:

{

medicine.modify(); //function modify

system("PAUSE");

break;

} //end case 3

case 4:

{

medicine.order\_list(); //function order

system("PAUSE");

break;

} //end case 4

case 5:

{

medicine.daily\_summary(); //function daily\_summary

system("PAUSE");

break;

} //end case 5

case 6:

{

medicine.exit(); //function exit

goto a;

break;

} //end case 6

default:

{

cout<<"You enter invalid input\nre-enter the input\n"<<endl;

break;

}//end defeault

}//end Switch

}while(menu!=6);//end do

a://goto

cout<<"thank you"<<endl;

system ("PAUSE");

return 0;

}//end main function

void medicineType::take\_order() //function to take\_order

{

system("cls");

int i;

int choice, quantity, price,None;

cout <<"\nAdd Order Details\n";

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

node \*temp;

temp=new node;

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

cout<<"DRUGS ID"<<"\tDRUGS TYPE"<<" \t\tDRUGS NAME"<<" DRUGS PRICE(INR)"<<endl;

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

cout<<"0001"<<"\t"<<"\tOTC"<<"\t\t"<<" Probiotics"<<" INR 2.00"<<endl;

cout<<"0002"<<"\t"<<"\tOTC"<<"\t\t"<<" Vitamin C(500mg)"<<" INR 3.00"<<endl;

cout<<"0003"<<"\t"<<"\tOTC"<<"\t\t"<<" Acid Free C(500mg)"<<" INR 1.00"<<endl;

cout<<"0004"<<"\t"<<"\tOTC"<<"\t\t"<<" Women'S Multivate"<<" INR 4.00"<<endl;

cout<<"0005"<<"\t"<<"\tOTC"<<"\t\t"<<" Marino Tablet"<<" INR 1.00"<<endl;

cout<<"0006"<<"\t"<<"\tOTC"<<"\t\t"<<" Maxi Cal Tablet"<<" INR 5.00"<<endl;

cout<<"0007"<<"\t"<<"\tOTC"<<"\t\t"<<" Amino Zinc Tablet"<<" INR 7.00"<<endl;

cout<<"0008"<<"\t"<<"\tOTC"<<"\t\t"<<" Burnex"<<" INR 4.00"<<endl;

cout<<"0009"<<"\t"<<"\tOTC"<<"\t\t"<<" Fabuloss 5"<<" INR 3.00"<<endl;

cout<<"0010"<<"\t"<<"\tOTC"<<"\t\t"<<" Royal Propollen"<<" INR 5.00"<<endl;

cout<<" "<<endl;

temp = new node;

cout << "Type Order no: ";

cin >> temp->reciept\_number;

cout<< "Enter Customer Name: ";

cin>> temp->customerName;

cout<<"Enter Date : ";

cin>>temp->date;

cout << "How many Medicine would you like to order:"<< endl;

cout<<"( Maximum is 10 order for each transaction ) \n";

cout << " " ;

cin >> temp->x;

if (temp->x >10)

{

cout << "The Medicine you order is exceed the maximum amount of order !";

system("pause");

}

else{

for (i=0; i<temp->x; i++)

{

cout << "Please enter your selection : "<<endl;

cin>> temp->menu2[i];

cout<< "Medicine Name: " <<temp->medicineName[temp->menu2[i]-1]<<endl;

cout << "How many medicine do you want: ";

cin >> temp->quantity[i];

temp->amount[i] = temp->quantity[i] \* temp->Medicine[temp->menu2[i]-1];

cout << "The amount You need to pay is: " << temp->amount[i]<<" INR"<<endl;

system("PAUSE");

}

cout<<"==========================================================================="<<endl;

cout << "Order Taken Successfully"<<endl;

cout<<"==========================================================================="<<endl;

cout << "Go to Reciept Menu to Pay The Bill"<<endl;

cout<<"==========================================================================="<<endl;

system ("PAUSE");

temp->next=NULL;

if(start\_ptr!=NULL)

{

temp->next=start\_ptr;

}

start\_ptr=temp;

system("cls");

}

}//End function take\_order

void medicineType::order\_list() //Function to display receipt

{

int i, num, num2;

bool found; //variable to search

system("cls");

node \*temp;

temp=start\_ptr;

found = false;

cout<<" Enter the Reciept Number To Print The Reciept\n";

cin>>num2;

cout<<"\n";

cout<<"==========================================================================="<<endl;

cout <<"\t\tHere is the Order list\n";

cout<<"==========================================================================="<<endl;

if(temp == NULL) //Invalid receipt code

{

cout << "\tThere is no Order to show\n\t\t\tSo The List is Empty\n\n\n";

}

while(temp !=NULL && !found)

{

if (temp->reciept\_number==num2)

{

found = true;

}

else

{

temp = temp -> next;

}

if (found) //print the receipt

{

cout <<"Reciept Number : "<<temp->reciept\_number;

cout <<"\n";

cout<<"Customer Name: "<<temp->customerName<<endl;

cout<<"Order Date : "<<temp->date<<endl;

cout<<"\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_"<<endl;

cout << "===============================================================================" << endl;

cout << "| Medicine Type | Medicine Name | Quantity | Total Price |" << endl;

cout << "=======++==================++================++===============++===============" << endl;

for (i=0;i<temp->x;i++)

{

cout << temp->type <<" \t\t";

cout<<temp->medicineName[temp->menu2[i]-1]<<"\t\t\t ";

cout<<temp->quantity[i] <<"\t\t";

cout<< temp->amount[i]<<" INR"<<endl;

cout<<"\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_"<<endl;

}

int k=0;

int sum = 0;

while(k<i){

sum += temp->amount[k];

k++;

}

temp->total = sum;

// +temp->amount[1]+temp->amount[2]+temp->amount[3]+temp->amount[4]+temp->amount[5]+temp->amount[6]+temp->amount[7]

// +temp->amount[8]+temp->amount[9];

cout<<"Total Bill is : "<<temp->total;

cout<<"\n";

cout << "Type the exact amount You need to pay: ";

cin >> num;

do{

cout<<"\n";

if(num!=temp->total){

cout<<"\nPlease Enter The Full Bill Payment...";

cout << "\nType the exact amount You need to pay: ";

cin >> num;

}

}while(num!=temp->total);

if (num == temp->total)

{cout <<"\n";

cout <<"\n";

cout<<"Payment Done\nThank You\n";

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

}else{

}

}

}

} //End function order\_list

void medicineType::delete\_order() //function to delete\_order

{

system("cls");

int i, num, count;

cout<<"Enter the data you want to delete \n";

cin>>num;

node \*q;

node \*temp;

bool found;

if(start\_ptr == NULL)

cerr<<"Can not delete from an empty list.\n";

else

{

if(start\_ptr->reciept\_number == num)

{

q = start\_ptr;

start\_ptr = start\_ptr->next;

count--;

if(start\_ptr == NULL)

last = NULL;

delete q;

cout<<"The Reciept is Deleted Successfully"<<endl;

}

else

{

found = false;

temp = start\_ptr;

q = start\_ptr->next;

while((!found) && (q != NULL))

{

if(q->reciept\_number != num)

{

temp = q;

q = q-> next;

}

else

found = true;

}

if(found)

{

temp->next = q->next;

count--;

if(last == q)

last = temp;

delete q;

cout<<"The Reciept is Deleted Successfully"<<endl;

}

else

cout<<"Item to be deleted is not in the list."<<endl;

}

}

} //End function delete\_order

void medicineType::modify() //function to modify order

{

system("cls");

int i, ch, sid;

bool found;

found = false;

temp = start\_ptr;

cout<<"Enter Receipt Number To Modify: ";

cin>>sid;

if (temp==NULL && sid==0)

{

cout<<"NO RECORD TO MODIFY..!"<<endl;

}

else

{

while(temp !=NULL && !found)

{

if (temp->reciept\_number==sid)

{

found = true;

}

else

{

temp = temp -> next;

}

if (found)

{

cout << "Change Order Number: ";

cin >> temp->reciept\_number;

cout<< "Change Customer Name: ";

cin>> temp->customerName;

cout<<"Change Date : ";

cin>>temp->date;

cout << "How many New Medicine would you like to Change:"<< endl;

cout<<"( Maximum is 10 order for each transaction ) \n";

cout << " " ;

cin >> temp->x;

if (temp->x >10)

{

cout << "The Medicine you order is exceed the maximum amount of order !";

system("pause");

}

else{

for (i=0; i<temp->x; i++)

{

cout << "Please enter your selection to Change: "<<endl;

cin>> temp->menu2[i];

cout<< "Change Medicine Name: " <<temp->medicineName[temp->menu2[i]-1]<<endl;

cout << "How many New medicine do you want: ";

cin >> temp->quantity[i];

temp->amount[i] = temp->quantity[i] \* temp->Medicine[temp->menu2[i]-1];

cout << "The amount You need to pay After Modify is: " << temp->amount[i]<<" INR"<<endl;

system("PAUSE");

}

temp = temp->next;

system("cls");

}

cout<<"RECORD MODIFIED....!"<<endl;

}

else

{

if(temp != NULL && temp->reciept\_number != sid)

{

cout<<"Invalid Reciept Number...!"<<endl;

}

}

}

}

} //End modify function

void medicineType::daily\_summary() //Function to display Daily Summary

{

int i,num;

system("cls");

node \*temp ;

temp=start\_ptr;

if(temp == NULL) //Invalid receipt code

{

cout << "\t\t\tThere is no Order to show\n\t\t\tSo The List is Empty\n\n\n";

}

else

{

cout<<"\n";

cout<<"==========================================================================="<<endl;

cout <<" \t\tHere is the Daily Summary of All Orders \n"; //print all receipt

cout<<"==========================================================================="<<endl;

while(temp!=NULL)

{

cout <<"Reciept Number : "<<temp->reciept\_number;

cout <<"\n";

cout<<"Customer Name: "<<temp->customerName<<endl;

cout<<"Order Date : "<<temp->date<<endl;

cout<<"\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_"<<endl;

cout << "==========================================================================" << endl;

cout << "| Medicine Type | Medicine Name | Quantity | Total Price |" << endl;

cout << "=======++==================++================++===============++==========" << endl;

for (i=0;i<temp->x;i++)

{

cout << temp->type <<" \t\t";

cout<<temp->medicineName[temp->menu2[i]-1]<<"\t\t";

cout<<temp->quantity[i] <<"\t\t";

cout<< temp->amount[i]<<" INR"<<endl;

cout<<"\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_"<<endl;

}

int k=0;

int sum = 0;

while(k<i){

sum += temp->amount[k];

k++;

}

temp->total = sum;

// temp->total = temp->amount[0]+temp->amount[1]+temp->amount[2]+temp->amount[3]+temp->amount[4]+temp->amount[5]+temp->amount[6]+temp->amount[7]

// +temp->amount[8]+temp->amount[9];

cout<<"Total Bill is : "<<temp->total;

cout <<"\n";

cout <<"\n";

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

temp=temp->next;

}

}

} //End daily summary

void medicineType::exit() //Function to exit

{

cout<<"\nYou choose to exit.\n"<<endl;

} //end function exit

////////////////////////////THE END OF PROGRAM//////////////////////////////////////////