National University of Computer and Emerging Sciences



Lab Exercise 9

For

Object Oriented Programming Lab

|  |  |
| --- | --- |
| Name | Muhammad Zain |
| Roll no | 19F-0228 |
| Course Instructor(s) | Dr. Danish |
| Lab Instructor(s) | Mr. Mughees Ismail |
| Semester | Spring 2020 |

**FAST School of Computing**

|  |
| --- |
| **Question#1** |

Task 1

# Source Code:

#include<iostream>

#include<string>

#include<iomanip>

using namespace std;

class PersonData {

protected:

string First\_Name;

string Last\_Name;

string Address;

string City;

string State;

int Zip;

string Contact;

public:

PersonData()//default constructor

{

First\_Name = "-";

Last\_Name = "-";

Address = "-";

City = "-";

State = "-";

Zip = 0;

Contact = "-";

}

void setName()

{

cout << "Enter First Name: ";

cin >> First\_Name;

while (First\_Name == "")

{

cin >> First\_Name;

}

cout << "Enter Last Name: ";

cin >> Last\_Name;

while (Last\_Name == "")

{

cin >> Last\_Name;

}

}

void getName()

{

cout << First\_Name << " " << Last\_Name;

}

void setContact()

{

cout << "Enter Contact Number: ";

getline(cin, Contact);

while (Contact == ""&& Contact.length() != 12)

{

getline(cin, Contact);

}

}

string getContact()

{

return Contact;

}

void setCity()

{

cout << "Enter your City ";

getline(cin, City);

while (City == "")

{

getline(cin, City);

}

}

string getCity()

{

return City;

}

void setAddress()

{

cout << "Enter your Address ";

getline(cin, Address);

while (Address == "")

{

getline(cin, Address);

}

}

string getAddress()

{

return Address;

}

void setZip() {

cout << "Enter the Zip " << endl;

cin >> Zip;

while (Zip == 0)

{

cin >> Zip;

}

}

int getZip()

{

return Zip;

}

void setState()

{

cout << "Enter your State ";

getline(cin, State);

while (State == "")

{

getline(cin, State);

}

}

string getState()

{

return State;

}

void InputCustomerData() {

setName();

setAddress();

setCity();

setContact();

setState();

setZip();

}

void DisplayCustomerData() {

getName();

cout << setfill(' ') << setw(20) << getAddress() << setfill(' ') << setw(20) << getCity();

cout << setfill(' ') << setw(20) << getContact();

cout << setfill(' ') << setw(20) << getState() << setfill(' ') << setw(18) << getZip();

}

};

class CustomerData :public PersonData {

protected:

int CustomerNumber;

bool MailingList;

public:

char choice;

void setCustomerNumber()

{

cout << "Enter the Coustomer Number" << endl;

cin >> CustomerNumber;

while (CustomerNumber == 0)

{

cin >> CustomerNumber;

}

}

int getCustomerNumber()

{

return CustomerNumber;

}

void InputCustomerData() {

PersonData::InputCustomerData();

setCustomerNumber();

cout << "If you want mailing list :Press Y " << endl;

cout << "If you dont want mailing list :Press N " << endl;

cin >> choice;

{

if ((choice = 'y') || (choice = 'Y'))

{

MailingList = false;

}

else if ((choice = 'N') || (choice = 'n'))

{

MailingList = true;

}}

}

void DisplayCustomerData()

{

PersonData::DisplayCustomerData();

cout << getCustomerNumber();

cout << getCustomerNumber() << setfill(' ') << setw(20) << MailingList; cout << endl << endl;

}

};

int main()

{

int number;

cout << "For how many Customers you want to enter the data" << endl;

cin >> number;

CustomerData \*Customer\_ptr = new CustomerData[number];

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

{

Customer\_ptr[i].InputCustomerData();

}

cout << endl << endl;

cout << "Name" << setw(23) << "Address" << setw(20) << "City" << setw(20) << "Contact No";

cout << setw(20) << "State" << setw(20) << "Zip-Code ";

cout << "Coustomer\_No" << setw(20) << "Mail list " << endl;

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

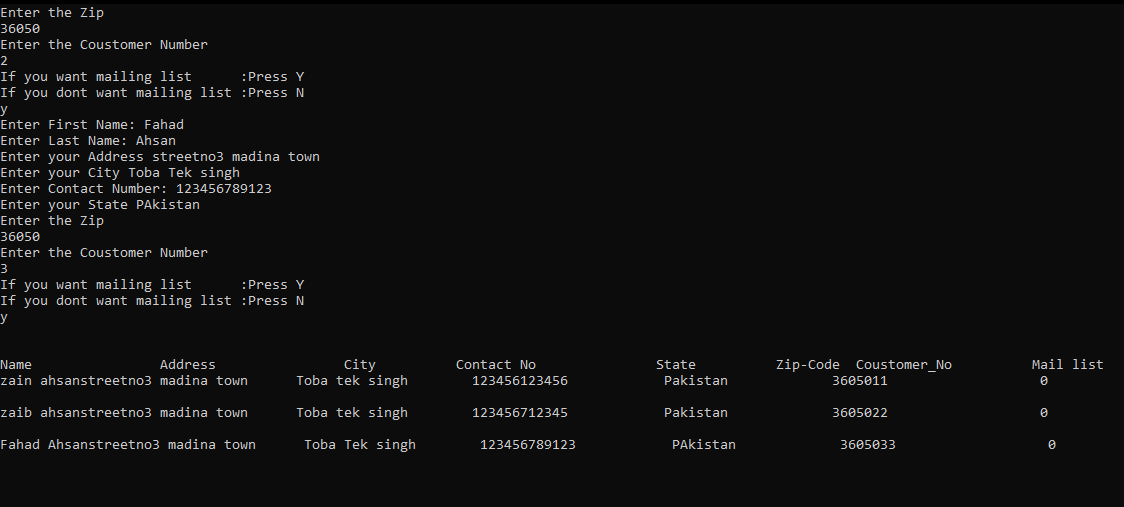
{

Customer\_ptr[i].DisplayCustomerData();

}

system("pause>0");

# }Snip:



|  |
| --- |
| **Question#2** |

#include<iostream>

#include<string>

#include<iomanip>

using namespace std;

class CustomerData {

protected:

int CustomerNumber;

bool MailingList;

public:

char choice;

void setCustomerNumber()

{

cout << "Enter the Coustomer Number" << endl;

cin >> CustomerNumber;

while (CustomerNumber == 0)

{

cin >> CustomerNumber;

}

}

int getCustomerNumber()

{

return CustomerNumber;

}

void InputCustomerData() {

setCustomerNumber();

cout << "If you want mailing list :Press Y " << endl;

cout << "If you dont want mailing list :Press N " << endl;

cin >> choice;

{

if ((choice = 'y') || (choice = 'Y'))

{

MailingList = false;

}

else if ((choice = 'N') || (choice = 'n'))

{

MailingList = true;

}}

}

void DisplayCustomerData()

{

cout << "Coustomer\_No" << setw(20) << "Mail list " << endl;

cout << getCustomerNumber() << setfill(' ') << setw(20) << MailingList << endl;

}

};

class PreferredCustomer :public CustomerData

{

protected:

double purchaseAmount;

double discountLevel;

public:

void set\_purchasesAmount()

{

cout << "Enter the Amount of purchasing" << endl;

cin >> purchaseAmount;

}

double get\_purchasesAmount()

{

return purchaseAmount;

}

void set\_discountLevel()

{

if (purchaseAmount >= 500 && purchaseAmount <= 1000)

{

cout << "Congratulation! You got 5% disscount on all future purchases :-) " << endl;

}

else if (purchaseAmount >= 1000 && purchaseAmount <= 1500)

{

cout << "Congratulation! You got 6% disscount on all future purchases :-) " << endl;

}

else if (purchaseAmount >= 1500 && purchaseAmount <= 2000)

{

cout << "Congratulation! You got 7% disscount on all future purchases :-) " << endl;

}

else if (purchaseAmount >= 2000)

{

cout << "Congratulation! You got 10% disscount on all future purchases :-) " << endl;

}

}

};

int main()

{

PreferredCustomer Object;

Object.InputCustomerData();

Object.set\_purchasesAmount();

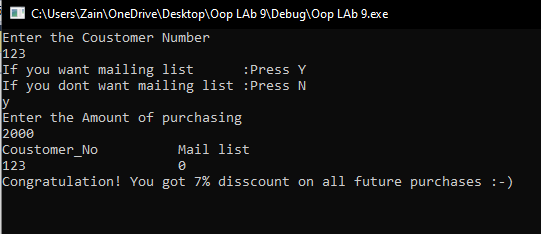
Object.DisplayCustomerData();

Object.set\_discountLevel();

system("pause>0");

}

# Snip:



|  |
| --- |
| **Question#3** |

# Source Code:

#include<iostream>

using namespace std;

class Account {

protected:

double Account\_Balance;

public:

Account()//constructor

{

cout << "Input the Initial Balance " << endl;

cin >> Account\_Balance;

{

if (Account\_Balance<0.0)

{

Account\_Balance = 0.0;

cout << "The initial balance was invalid :-(" << endl;

}

}

}

double setCredit()

{

int deposit = 0;

cout << "Enter the Money to deposit in the Account" << endl;

cin >> deposit;

Account\_Balance += deposit;

return Account\_Balance;

}

double setDebit()

{

int withdraw;

cout << "Enter the Money to with draw from the Account" << endl;

cin >> withdraw;

if (withdraw>Account\_Balance)

{

cout << "Debit amount exceeded Account Balance" << endl;

}

else if (withdraw <= Account\_Balance)

{

Account\_Balance -= withdraw;

}

return Account\_Balance;

}

void getBalance()

{

cout << " Your Current account balance is = " << Account\_Balance << endl;

}

};

class SavingsAccount :public Account {

protected:

double Interest\_Rate;

double Interest\_Amount;

public:

SavingsAccount()

{

cout << "The interest Rate on your Account is set to 5 %" << endl;

Interest\_Rate = 5;

}

double setInterest()

{

Interest\_Amount = (Account\_Balance / 100)\*Interest\_Rate;

cout << "Intereset Amount = " << Interest\_Amount;

}

};

class CheckingAccount :public Account {

protected:

double Fee;

public:

CheckingAccount()

{

cout << "\nAmount per transaction is set to 5 Rs " << endl;

Fee = 5;

} void setCredit()

{

int choice;

double storage;

int deposit = 0;

cout << "Press 1 If you want to Deposit amount" << endl;

cout << "Press 2 If you want to check the balance " << endl;

cin >> choice;

if (choice == 1)

{

storage = Account::setCredit();

storage -= 5;

cout << "Your New Balance is " << storage << endl;

}

else if (choice == 2)

{

Account::getBalance();

}

}

void setDebit()

{

int choice;

double storage;

cout << "Press 1 If you want to Withdraw amount" << endl;

cout << "Press 2 If you want to check the balance " << endl;

cin >> choice;

if (choice == 1)

{

storage = Account::setDebit() - Fee;

cout << "Your Remaining Balance is " << storage << endl;

}

else if (choice == 2)

{

Account::getBalance();

}

}

};

int main()

{

int type = 0;

cout << "If you want to test Account Class Press 1" << endl;

cout << "If you want to test Savings Account Class Press 2" << endl;

cout << "If you want to test Checking Classs Press 3" << endl;

cin >> type;

if (type == 1)

{

cout << "\t\t Account Class (The Base Class)" << endl << endl;

Account obj;

obj.setCredit();

obj.setDebit();

obj.getBalance();

}

else if (type == 2)

{

cout << "\t\t Saving Account Class (Derrived Class 1)" << endl << endl;

SavingsAccount Object;

Object.setCredit();

Object.setDebit();

Object.setInterest();

Object.getBalance();

}

else if (type == 3)

{

cout << "\t\t Checking Class (Derrived Class 2)" << endl << endl;

CheckingAccount object;

object.setCredit();

object.setDebit();

object.getBalance();

}

System(“pause>0”);

}

# Snip:

