Bank management

Description: This C++ programs on BANKING SYSTEM has account class with data members like account number, name, deposit, withdraw amount and type of account. Customer data is stored in a binary file. A customer can deposit and withdraw amount in his account. User can create, modify and delete account.

In this banking system project, we have not used graphics to keep program simple. Click on download project button to download zip folder which contains C++ source code file, sample data file (.dat file), output screen (.doc file).

Source code:

#include<iostream>

#include<fstream>

#include<cctype>

#include<iomanip>

#include <cstdlib>

using namespace std;

class account

{

int acno;

char name[50];

int deposit;

char type;

public:

void create\_account(); //function to get data from user

void show\_account() const; //function to show data on screen

void modify(); //function to add new data

void dep(int); //function to accept amount and add to balance amount

void draw(int); //function to accept amount and subtract from balance amount

void report() const; //function to show data in tabular format

int retacno() const; //function to return account number

int retdeposit() const; //function to return balance amount

char rettype() const; //function to return type of account

}; //class ends here

void account::create\_account()

{

cout<<"\nEnter The account No. : ";

cin>>acno;

cout<<"\n\nEnter The Name of The account Holder : ";

cin.ignore();

cin.getline(name,50);

cout<<"\nEnter Type of The account (C/S) : ";

cin>>type;

type=toupper(type);

cout<<"\nEnter The Initial amount(>=500 for Saving and >=1000 for current ) : ";

cin>>deposit;

cout<<"\n\n\nAccount Created..";

}

void account::show\_account() const

{

cout<<"\nAccount No. : "<<acno;

cout<<"\nAccount Holder Name : ";

cout<<name;

cout<<"\nType of Account : "<<type;

cout<<"\nBalance amount : "<<deposit;

}

void account::modify()

{

cout<<"\nAccount No. : "<<acno;

cout<<"\nModify Account Holder Name : ";

cin.ignore();

cin.getline(name,50);

cout<<"\nModify Type of Account : ";

cin>>type;

type=toupper(type);

cout<<"\nModify Balance amount : ";

cin>>deposit;

}

void account::dep(int x)

{

deposit+=x;

}

void account::draw(int x)

{

deposit-=x;

}

void account::report() const

{

cout<<acno<<setw(10)<<" "<<name<<setw(10)<<" "<<type<<setw(6)<<deposit<<endl;

}

int account::retacno() const

{

return acno;

}

int account::retdeposit() const

{

return deposit;

}

char account::rettype() const

{

return type;

}

void write\_account(); //function to write record in binary file

void display\_sp(int); //function to display account details given by user

void modify\_account(int); //function to modify record of file

void delete\_account(int); //function to delete record of file

void display\_all(); //function to display all account details

void deposit\_withdraw(int, int); // function to deposit/withdraw amount for given account

void intro(); //introductory screen function

int main()

{

char ch;

int num;

intro();

do

{

system("cls");

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

cout<<"\n\n\t01. NEW ACCOUNT ";

cout<<"\n\n\t02. DEPOSIT AMOUNT ";

cout<<"\n\n\t03. WITHDRAW AMOUNT ";

cout<<"\n\n\t04. BALANCE ENQUIRY ";

cout<<"\n\n\t05. ALL ACCOUNT HOLDER LIST ";

cout<<"\n\n\t06. CLOSE AN ACCOUNT ";

cout<<"\n\n\t07. MODIFY AN ACCOUNT ";

cout<<"\n\n\t08. EXIT";

cout<<"\n\n\tSelect Your Option (1-8) ";

cin>>ch;

system("cls");

switch(ch)

{

case '1':

write\_account();

break;

case '2':

cout<<"\n\n\tEnter the account No. : "; cin>>num;

deposit\_withdraw(num, 1);

break;

case '3':

cout<<"\n\n\tEnter the account No. : "; cin>>num;

deposit\_withdraw(num, 2);

break;

case '4':

cout<<"\n\n\tEnter the account No. : "; cin>>num;

display\_sp(num);

break;

case '5':

display\_all();

break;

case '6':

cout<<"\n\n\tEnter The account No. : "; cin>>num;

delete\_account(num);

break;

case '7':

cout<<"\n\n\tEnter The account No. : "; cin>>num;

modify\_account(num);

break;

case '8':

cout<<"\n\n\tThanks for using the bank";

break;

default :cout<<"\a";

}

cin.ignore();

cin.get();

}while(ch!='8');

return 0;

}

void write\_account()

{

account ac;

ofstream outFile;

outFile.open("account.dat",ios::binary|ios::app);

ac.create\_account();

outFile.write(reinterpret\_cast<char \*> (&ac), sizeof(account));

outFile.close();

}

void display\_sp(int n)

{

account ac;

bool flag=false;

ifstream inFile;

inFile.open("account.dat",ios::binary);

if(!inFile)

{

cout<<"File could not be open !! Press any Key...";

return;

}

cout<<"\nBALANCE DETAILS\n";

while(inFile.read(reinterpret\_cast<char \*> (&ac), sizeof(account)))

{

if(ac.retacno()==n)

{

ac.show\_account();

flag=true;

}

}

inFile.close();

if(flag==false)

cout<<"\n\nAccount number does not exist";

}

void modify\_account(int n)

{

bool found=false;

account ac;

fstream File;

File.open("account.dat",ios::binary|ios::in|ios::out);

if(!File)

{

cout<<"File could not be open !! Press any Key...";

return;

}

while(!File.eof() && found==false)

{

File.read(reinterpret\_cast<char \*> (&ac), sizeof(account));

if(ac.retacno()==n)

{

ac.show\_account();

cout<<"\n\nEnter The New Details of account "<<endl;

ac.modify();

int pos=(-1)\*static\_cast<int>(sizeof(account));

File.seekp(pos,ios::cur);

File.write(reinterpret\_cast<char \*> (&ac), sizeof(account));

cout<<"\n\n\t Record Updated ";

found=true;

}

}

File.close();

if(found==false)

cout<<"\n\n Record Not Found ";

}

void delete\_account(int n)

{

account ac;

ifstream inFile;

ofstream outFile;

inFile.open("account.dat",ios::binary);

if(!inFile)

{

cout<<"File could not be open !! Press any Key... ";

return;

}

outFile.open("Temp.dat",ios::binary);

inFile.seekg(0,ios::beg);

while(inFile.read(reinterpret\_cast<char \*> (&ac), sizeof(account)))

{

if(ac.retacno()!=n)

{

outFile.write(reinterpret\_cast<char \*> (&ac), sizeof(account));

}

}

inFile.close();

outFile.close();

remove("account.dat");

rename("Temp.dat","account.dat");

cout<<"\n\n\tRecord Deleted .. ";

}

void display\_all()

{

account ac;

ifstream inFile;

inFile.open("account.dat",ios::binary);

if(!inFile)

{

cout<<"File could not be open !! Press any Key... ";

return;

}

cout<<"\n\n\t\tACCOUNT HOLDER LIST\n\n";

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

cout<<"A/c no. NAME Type Balance\n";

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

while(inFile.read(reinterpret\_cast<char \*> (&ac), sizeof(account)))

{

ac.report();

}

inFile.close();

}

void deposit\_withdraw(int n, int option)

{

int amt;

bool found=false;

account ac;

fstream File;

File.open("account.dat", ios::binary|ios::in|ios::out);

if(!File)

{

cout<<"File could not be open !! Press any Key... ";

return;

}

while(!File.eof() && found==false)

{

File.read(reinterpret\_cast<char \*> (&ac), sizeof(account));

if(ac.retacno()==n)

{

ac.show\_account();

if(option==1)

{

cout<<"\n\n\tTO DEPOSIT AMOUNT ";

cout<<"\n\nEnter The amount to be deposited : ";

cin>>amt;

ac.dep(amt);

}

if(option==2)

{

cout<<"\n\n\tTO WITHDRAW AMOUNT ";

cout<<"\n\nEnter The amount to be withdraw : ";

cin>>amt;

int bal=ac.retdeposit()-amt;

if((bal<500 && ac.rettype()=='S') || (bal<1000 && ac.rettype()=='C'))

cout<<"Insufficient balance";

else

ac.draw(amt);

}

int pos=(-1)\*static\_cast<int>(sizeof(ac));

File.seekp(pos,ios::cur);

File.write(reinterpret\_cast<char \*> (&ac), sizeof(account));

cout<<"\n\n\t Record Updated ";

found=true;

}

}

File.close();

if(found==false)

cout<<"\n\n Record Not Found ";

}

void intro()

{

cout<<"\n\n\n\t\t\t\t\t\*\*\*RAMAIAH BANKING SYSTEM\*\*\*";

cout<<"\n\n\n\n\t\t\t\t MADE BY : Ramendra , Raga & Monisha ";

cin.get();

}

Submitted by:

MONISHA D 1MS18MLO24

RAGA SATARASI 1MS18ML035

RAMENDRA BHARADWAJ 1MS18ML036