



**Курсова робота**

з предмету: «Програмування, частина 2 (Об’єктно-орієнтоване програмування)»

на тему:

**«Базові принципи об’єктно-орієнтованого програмування»**

Індивідуальне завдання:

«Банк»

**Виконав:**

ст. гр. КІ-15

Леземезюк Т.Т.

**Прийняв:**

ст. викл., к.т.н.

Козак Н.В.

Львів-2020

Анотація

Курсова робота з об’єктно-орієнтованого програмування на тему «Базові принципи об’єктно-орієнтованого програмування» по завданню «Банк» зроблена заради вивчення базових принципів ООП та закріплення цих знань на практиці. Програмна реалізація написана на мові програмування С++ за допомогою програмного забезпечення Microsoft Visual Studio 2019.

Зміст:

Анотація …………………………………………………………………………. 2

Завдання на курсову роботу……………………………………………………...4

1. Огляд та обґрунтовування вибору технологій об’єктно-орієнтованого програмування…………………………………………………………………….5

2. Аналіз та розробка алгоритму згідно індивідуального завдання………...…7

3. Програмна реалізація алгоритму згідно індивідуального завдання…….....11

4. Вiдлагодження та тестування реалізованої програми згідно індивідуального завдання……………………………………………………….18

5. Аналіз продуктивності (часу роботи) реалізованої програми згідно індивідуального завдання……………………………………………………….21

Завдання на курсову роботу

Індивідуальне завдання: Банк.

Вхідні дані були взяті з банку «ОщадБанк» , а саме: розрахунок та функціонування депозитів і кредитів, процентні ставки на кредити та депозити. Також, метод створення власного рахунку та переказ коштів. Функція банкомату також розроблена на основі даних з ОщадБанку.

1. Огляд та обґрунтовування вибору технологій об'єктно-орієнтованого програмування.

Об'єктно-орієнтоване програмування (ООП) — одна з парадигм програмування, яка розглядає програму як множину «об'єктів», що взаємодіють між собою. Основу ООП складають три основні концепції: інкапсуляція, успадкування та поліморфізм. Одною з переваг ООП є краща модульність програмного забезпечення. Основу ООП складають чотири основні концепції: інкапсуляція, успадкування, поліморфізм та абстракція.

1.1 Успадкування (наслідування).

Клас може мати «підкласи», спеціалізовані, розширені версії надкласу. Можуть навіть утворюватися цілі дерева успадкування. Підкласи успадковують атрибути та поведінку своїх батьківських класів, і можуть вводити свої власні. Успадкування може бути одиничне (один безпосередній батьківський клас) та множинне (кілька батьківських класів).

1.2 Інкапсуляція.

Приховування деталей про роботу класів від об'єктів, що їх використовують або надсилають їм повідомлення. Інкапсуляція досягається шляхом вказування, які класи можуть звертатися до членів об'єкта. Як наслідок, кожен об'єкт представляє кожному іншому класу певний інтерфейс — члени, доступні іншим класам. Інкапсуляція потрібна для

того, аби запобігти використанню користувачами інтерфейсу тих частин реалізації, які, швидше за все, будуть змінюватись. Це дозволить полегшити внесення змін, тобто, без потреби змінювати і користувачів інтерфейсу. Часто, члени класу позначаються як публічні (англ. public), захищені (англ. protected) та приватні (англ. private), визначаючи, чи доступні вони всім класам, підкласам, або лише до класу в якому їх визначено.

1.3 Поліморфізм.

Поліморфізм означає залежність поведінки від класу, в якому ця поведінка викликається, тобто, два або більше класів можуть реагувати по-різному на однакові повідомлення. На практиці - це реалізовується шляхом реалізації ряду підпрограм (функцій, процедур, методів тощо) з однаковими іменами, але з різними параметрами. В залежності від того, що передається і вибирається відповідна підпрограма.

Одною з переваг ООП є краща модульність програмного забезпечення (тисячу функцій процедурної мови, в ООП можна замінити кількома десятками класів із своїми методами). Попри те, що ця парадигма з'явилась в 1960-тих роках, вона не мала широкого застосування до 1990-тих, коли розвиток комп'ютерів та комп'ютерних мереж дозволив писати надзвичайно об'ємне і складне програмне забезпечення, що змусило переглянути підходи до написання програм. Сьогодні багато мов програмування або підтримують ООП, або ж є цілком об'єктно-орієнтованими (зокрема, Java, C#, C++, Python і так далі).

На відміну від традиційних поглядів, коли програму розглядали як набір підпрограм, або як перелік інструкцій комп'ютеру, ООП програми можна вважати сукупністю об'єктів. Відповідно до парадигми об'єктно-орієнтованого програмування, кожен об'єкт здатний отримувати повідомлення, обробляти дані, та надсилати повідомлення іншим об'єктам. Кожен об'єкт — своєрідний незалежний автомат з окремим призначенням та відповідальністю.

2. Аналіз та розробка алгоритму згідно індивідуального завдання

* 1. Опис алгоритму та оцінка його складності

В алгоритмі присутні 5 конструкції if:

1. Перевірка імені(при запуску програми);
2. Вибір операції;
3. Вибір банківської операції;
4. Перевірка коштів та pin-коду карти;

Якщо користувач вже є клієнтом банку буде виведено його дані, якщо користувач не є клієнтом тоді буде запропонована реєстрації, по закінченню реєстрації буде виведена інформація про його новий рахунок в банку. Після виконання будь-якої операції буде виводитись інформація щодо рахунку користувача.

* 1. Аналіз особливостей алгоритму

1. В першому операторі if програма перевіряє чи клієнт вже має рахунок в банку, зчитуючи з файлу “Storage.txt” та запишуться в “Form.txt”. Якщо користувач вже є клієнтом банку відкриється вікно з виводом інформації про його рахунок. Якщо користувач не має рахунку в банку відкриється вікно реєстрації (де ви зможете створити рахунок та карту), вся введена інформація буде записана в перший буферний файл “Form.txt”, після цього виведеться інформація щодо рахунку користувача.
2. В другому операторі if користувач буде мати вибір між банківськими операціями, банкоматом, терміналом та закриттям програми. При виборі четвертого варіанта вся інформація з файла “Storage.txt” буде записана в “Form.txt” та видалена після цього буферний файл “Form.txt” буде записаний в “Storage.txt” та видалений – програма закриється .
3. Якщо користувач вибере один з перших трьох варіантів. Всі маніпуляції з грошима які користувач виконає будуть записані в другий буферний файл “Storage2.txt”. Після виконання операцій буде виводитись меню вибору другого оператора.
   1. Опис алгоритму у псевдокоді

1 Input FirstName;

2 Read Name;

3 If(FirstName ==Name) Than

4 Print FirstName, LastName, Money, CardNumber, CardPin;

5 End;

6 If(FirstName!=Name)Than

7 Input FirstName, LastName, Money, CardNumber, CardPin;

8 Print FirstName, LastName, Money, CardNumber, CardPin;

9 End;

Input Choice;

While Choice!=4 Do

10 Input Choice;

11 If(Choice==1)Than;

12 Print Deposits and Credits;

13 Input Choice 2;

14 If(Choise2==1)Than;

Input Time, Proz;

Input Time, Proc;

Do;

MoneyProc = Money \* Time / 100;

Credit = (moneyProc \* Proc) + Money;

End;

If(Choise2==2)Than;

Input Time, Proc;

Do;

MoneyProc = Money \* Time / 100;

Credit = (moneyProc \* Proc) + Money;

End;

If(Choise==2)Than

Print ATM;

Input Pin;

If(Pin==CardPin)Than

Do;

Input MoneyI;

Money=Money-MoneyI;

Else Goto Input Pin;

End;

If(Choise==3)Than

Print Terminal;

Input Pin;

If(Pin==CardPin)Than

Do;

Input MoneyI;

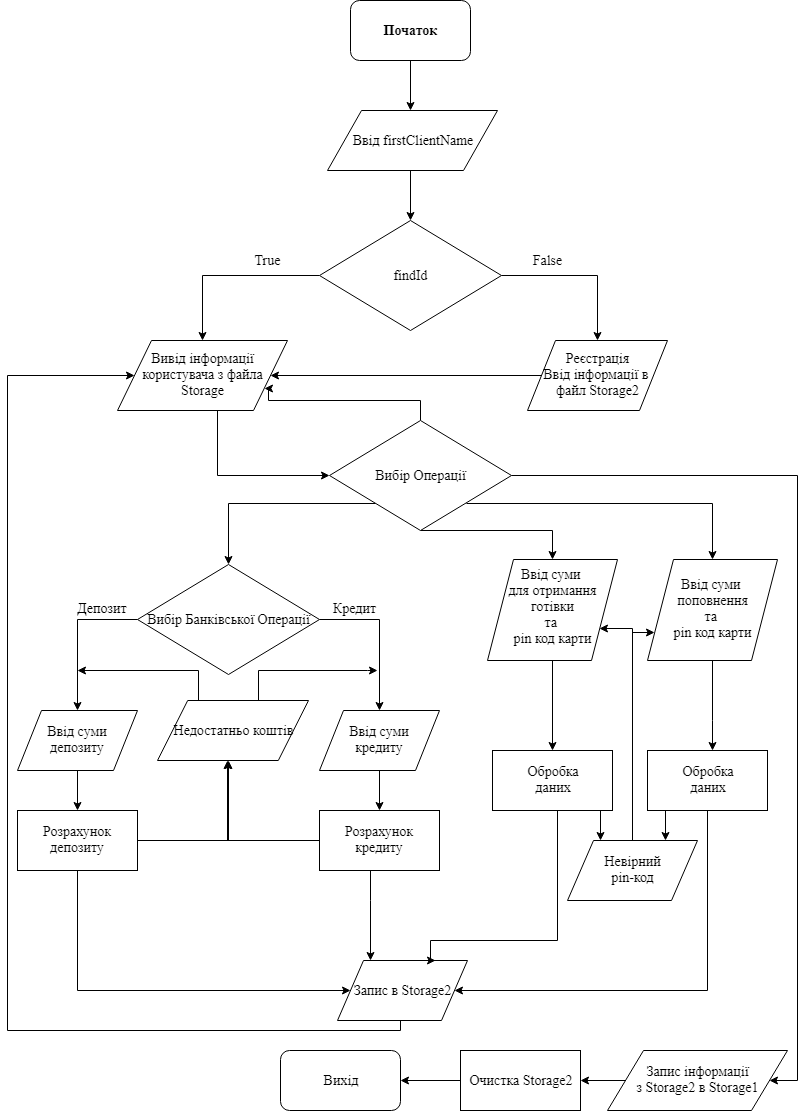
Money=Money-MoneyI;

Else Goto Input Pin;

End;

End;

Print FirstName, LastName, Money, CardNumber, CardPin;

2.4 Розробка граф-схеми алгоритму

3. Програмна реалізація алгоритму згідно індивідуального завдання

* 1. Опис програмної реалізації алгоритму

3.1.1 Опис структур даних програми

В програмі наявні змінні:

* цілого типу для виконання умови If;
* символьний тип для виконання If;
* цілого, символьного та з плаваючою комою типи даних для зчитування з файлу, виконання обчислень та виведення в консоль/файл;
* об’єкти класів для виконання методів.

3.1.2 Схема викликів функцій або діаграма класів

1. Вивід на екран пропозиції по вводу імені користувача;
2. Зчитування з файлу та перевірка імені користувача;
3. Якщо ім’я з файлу співпадає з введеним переходить до пункту 3 і записує в буферний файл;
4. Якщо ім’я з файлу не співпадає з введеним переходить до реєстрації далі до пункту 3 і записує в буферний файл;
5. Вивід інформації рахунку клієнта;
6. Меню вибору операцій;
7. Банківські операції(Депозити та Кредити);
8. Банкомат;
9. Термінал;
10. Баланс;
11. Вихід;
12. Після виконання одної з операцій дані перезаписуються в буферному файлі;
13. Виводиться меню вибору операцій;
14. При виході з програми дані з головного файла записуються в буферний після чого головний файл видаляється і створюється новий далі дані з буферного файлу записуються в головний програма закривається;

3.1.3 Опис реалізації основних функцій/методів

1. Функція перевірки імені користувача

void Client::findId()

{

ifstream f1;

f1.open(path1);

ifstream f2;

string nameid;

string line;

string name;

string lastname;

int num;

int pin;

double money;

bool t1 = 0;

while (!f1.eof()) {

getline(f1, line);

if (line.find(getFirstClientName()) != string::npos) {

t1++;

stringstream ss(line);

ss >> name;

ss >> lastname;

ss >> pin;

ss >> num;

ss >> money;

setFirstClientName(name);

setLastClientName(lastname);

setCardNumClient(num);

setPinClient(pin);

setMoneyClient(money);

ofstream f;

f.open("form.txt");

f << line;

f.close();

//cout << line << endl;

setId(true);

break;

}

}

if(t1==0) {

setId(false);

}

}

1. Функція перезапису файлів(та завершення програми)

void Client::revrite()

{

string line;

string linebuff;

ofstream f2;

remove("Storage2.txt");

f2.open(path2, ios\_base::app);

f2 << getFirstClientName() << " " << getLastClientName() << " " << getPinClient() << " " << getCardClient() << " ";

f2 << getMoneyClient()<<endl;

ifstream f1;

f1.open(path1);

while (!f1.eof()) {

getline(f1, line);

if (line.find(getFirstClientName()) == string::npos) {

f2 << line << endl;;

}

}

f1.close();

f2.close();

ofstream f3;

remove("Storage.txt");

f3.open(path1, ios\_base::app);

ifstream f4;

f4.open(path2);

while (!f4.eof()) {

getline(f4, line);

f3 << line<<endl;

}

f3.close();

f4.close();

}

1. Створення буферного файлу

void Client::start()

{

string line;

string name;

string lastname;

int num;

int pin;

double money;

ifstream form;

form.open("form.txt");

getline(form, line);

stringstream ss(line);

ss >> name;

ss >> lastname;

ss >> pin;

ss >> num;

ss >> money;

setFirstClientName(name);

setLastClientName(lastname);

setCardNumClient(num);

setPinClient(pin);

setMoneyClient(money);

form.close();

}

1. Реєстрація

void Client::registration()

{

int x;

cout << "Ви не маєте рахункiв в нашому банку"<<endl;

cout << "Бажаєте завести?" << endl;

cout << "Так(1) або Ні(2)" << endl;

cin >> x;

if (x == 1)

{

string name;

string lastname;

cout << "Введiть Iмя'я" << endl;

cin >> name;

setFirstClientName(name);

cout << "Введiть Прiзвище" << endl;

cin >> lastname;

setLastClientName(lastname);

cout << "Бажаєте створити карту?" << endl;

cout << "Так(1) або Ні(2)" << endl;

int y;

cin >> y;

if (y == 1)

{

createCard();

setPinClient(pinCard);

setCardClient(cardNum);

cout << "Ваш номер карти - " << getCardClient() << " та ваш pin - " << getPinClient() << endl;

cout << "Бажаєте покласти гроші на ваш рахунок?" << endl;

cout << "Так(1) або Ні(2)" << endl;

int a;

cin >> a;

if (a == 1)

{

cout << "Введіть суму - ";

double money;

cin >> money;

setMoneyClient(money);

}

}

}

1. Депозити (Принцип роботи розрахунку кредитів подібний

double Operations::deposit(int time, double firstMoney)

{

float procent;

double moneyProc;

double deposit;

double buf;

if (time == 3)

{

procent = 4;

moneyProc =firstMoney \* 4 / 100;

deposit = (moneyProc \* 3)+firstMoney;

buf = deposit - firstMoney;

setMoneyD(buf);

return deposit;

}

if (time == 9)

{

procent = 11;

moneyProc = firstMoney \* 4 / 100;

deposit = (moneyProc \* 9) + firstMoney;

buf = deposit - firstMoney;

setMoneyD(buf);

return deposit;

}

if (time == 15)

{

procent = 20;

moneyProc = firstMoney \* 4 / 100;

deposit = (moneyProc \* 15) + firstMoney;

buf = deposit - firstMoney;

setMoneyD(buf);

return deposit;

}

}

1. Операції з банкоматом

double ATM::takeMoney(double money)

{

double moneyIN=0;

double moneyOUT=0;

while (true) {

cout << "Введiть суму яку ви хочете зняти - ";

cin >> moneyIN;

cout << endl;

if (moneyIN > money)

{

cout << "У вас недостатньо коштiв" << endl;

}

else

{

moneyOUT = money - moneyIN;

break;

}

}

return moneyOUT;

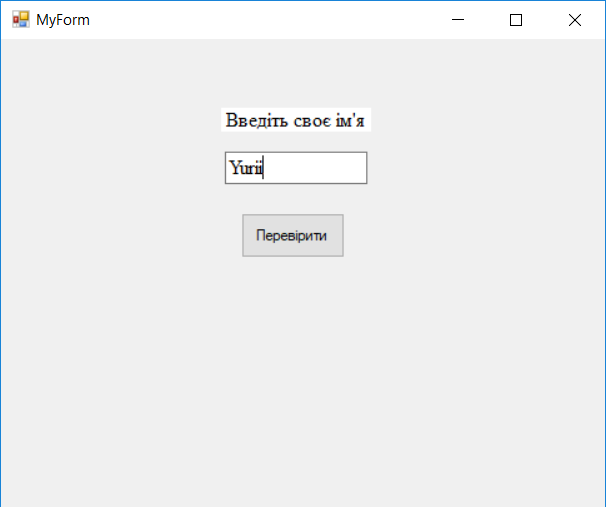
}

Решта методів наведенні у додатку.

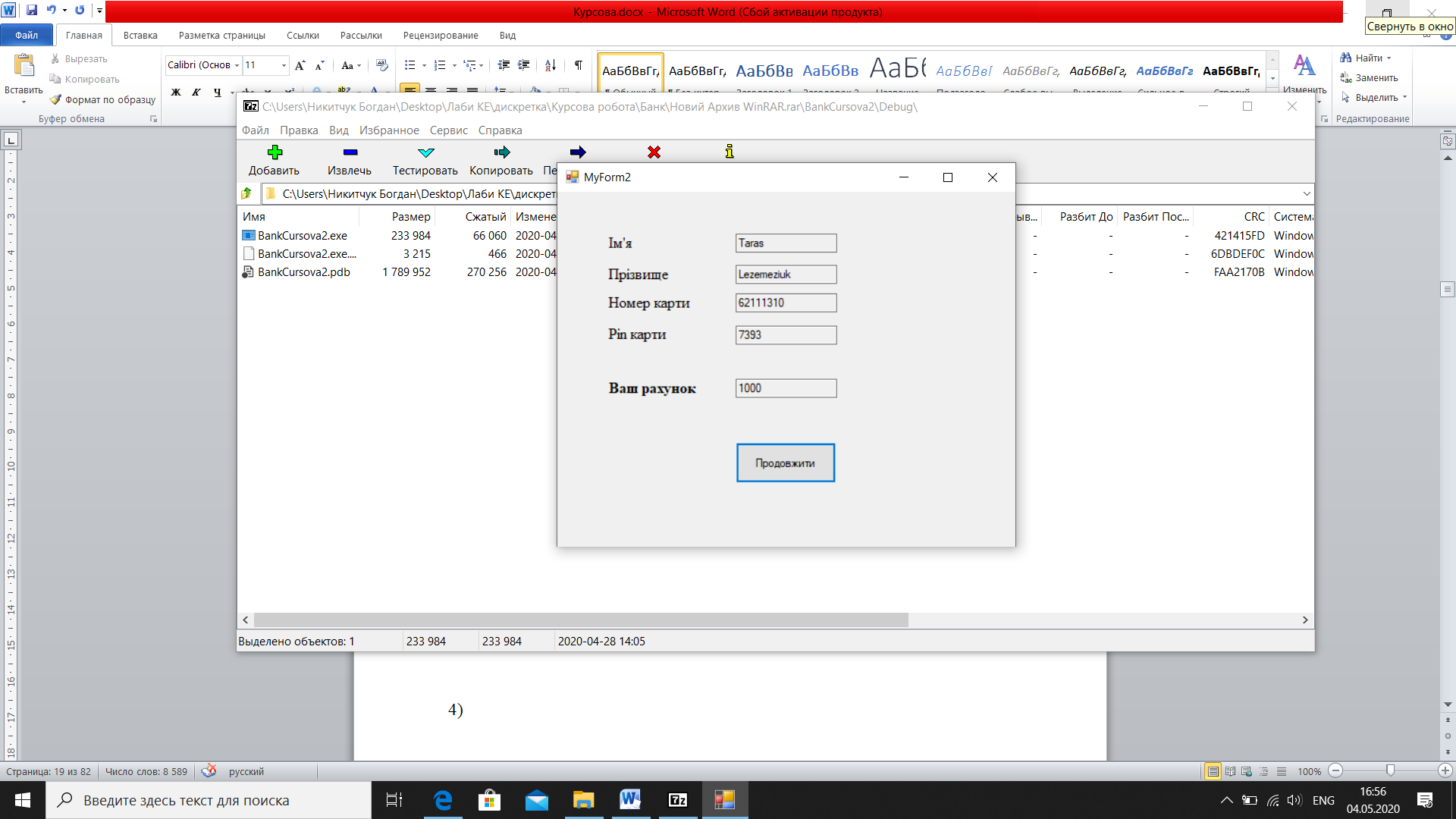
4. Вiдлагодження та тестування реалізованої програми згідно індивідуального завдання

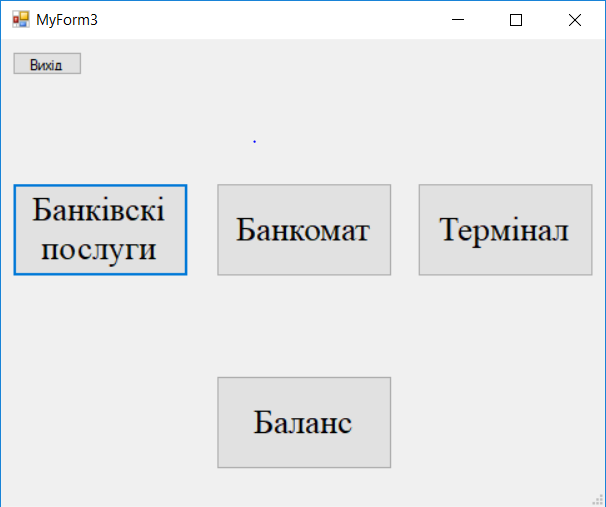
Для зручного користування програми користувачем, було створено графічний інтерфейс. Текст коду графічного інтерфейсу наведено у додатку.

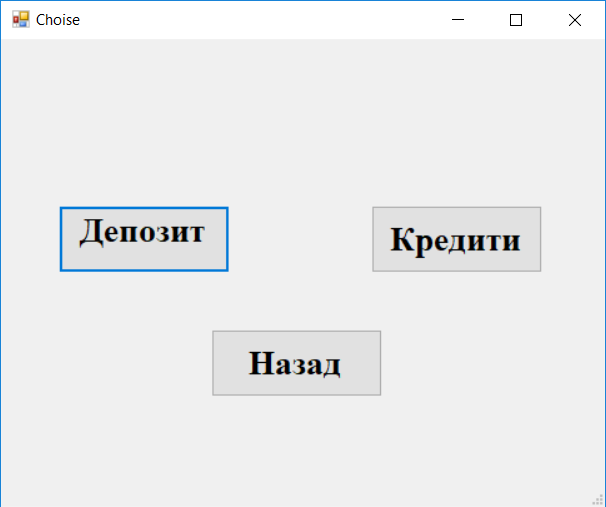
Скріншоти роботи програми:

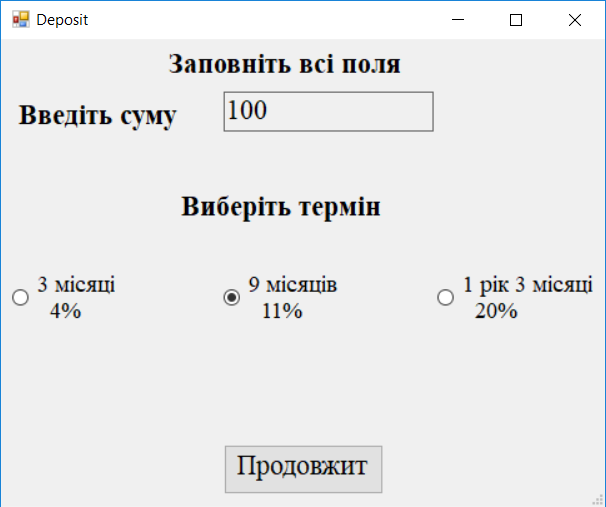


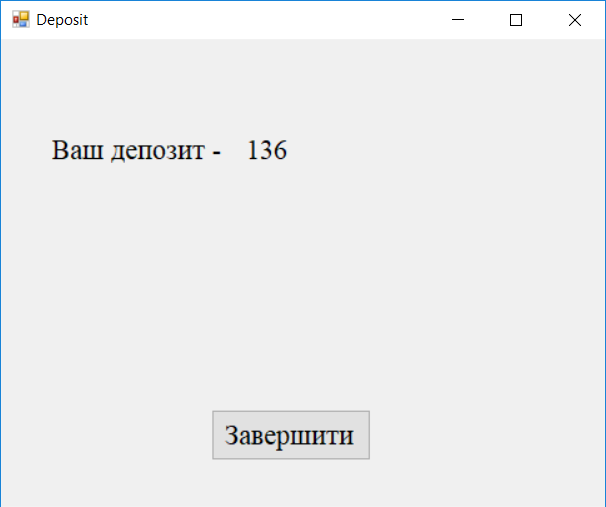
Taras



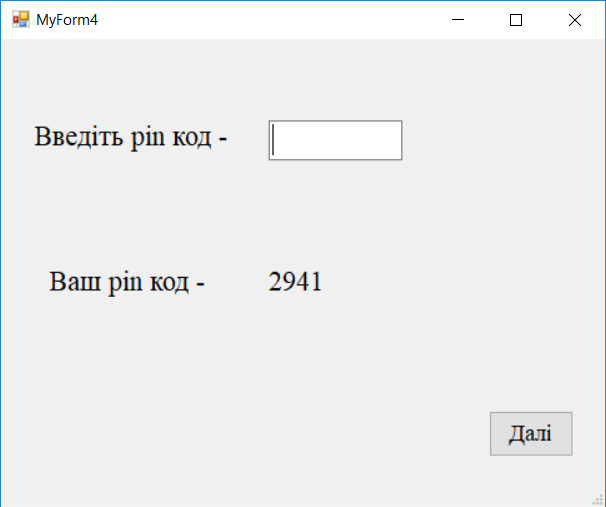


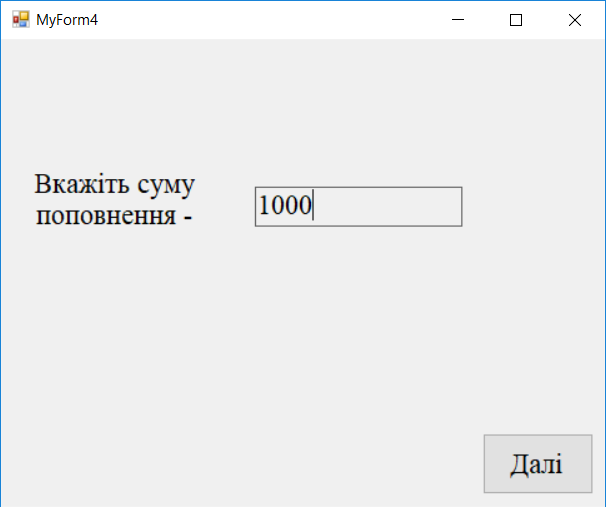


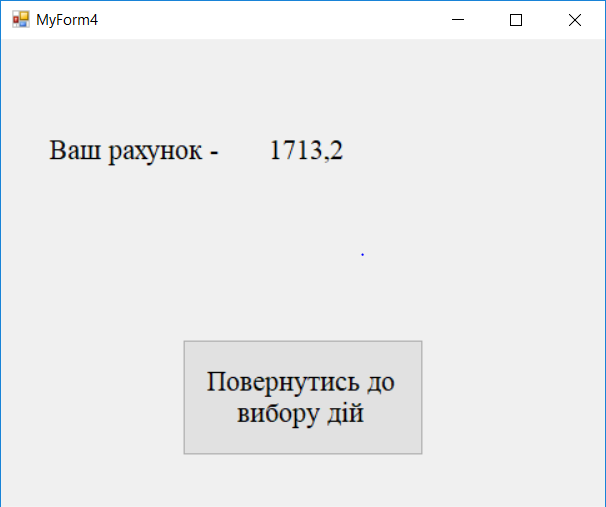


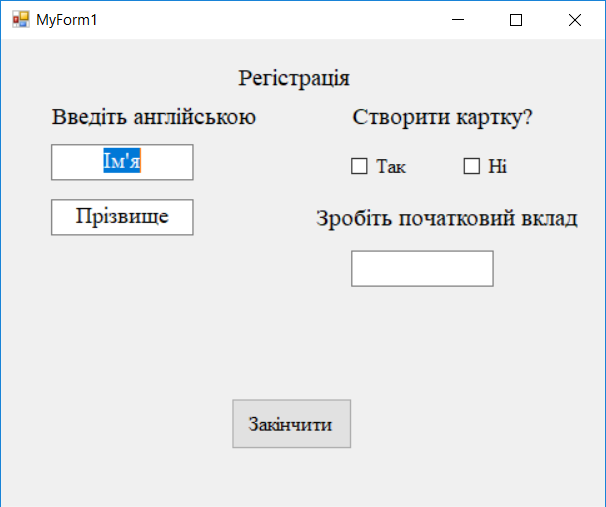


1. Робота термінала









Вікно реєстрації

Все спрацювало як і задумувалось.

Підсумовуючи програма працює справно.

5. Аналіз продуктивності (часу роботи) та структурної складності реалізованої програми згідно індивідуального завдання.

5.1 Аналіз структурної складності виконання програми

У програмі наявні 3 оператори if (та ще 2 в меню реєстрації). Ці конструкції начебто уявно відділяють блок коду від усієї програми, що покращує сприйняття й розуміння виконаних та запланованих цілей (функцій, операцій). Як висновок, можу сказати що, виконання програми на структурному рівні не є зовсім простим.

5.2 Аналіз часу виконання програми в залежності від об'єму вхідних даних.

Всі операції всередині програми виконуються миттєво, немає затримки між переходами по меню та виведенням інформації на консоль. Читання даних з файлу та виконання операцій з числами виконується бездоганно.

Висновок:

В результаті виконання даної курсової роботи, я вивчив базові принципи об’єктно-орієнтованого програмування та закріпив отримані знання на практиці. Я створив програму за допомогою якої можна здійснювати безліч банківських операції , зокрема: поповнення поточного рахунку, зняття готівки та отримання депозиту або кредиту. Роблячи цю курсову роботу я використав основні принципи об’єктно-орієнтованого програмування, а саме наслідування, поліморфізм та інкапсуляція, що дало можливість полегшити користування програмою. Також, я створив графічний інтерфейс даної програми для зручного та ефективного користування.

Список літератури:

1. **Програмування, частина 2 (Об’єктно-орієнтоване програмування):** методичні вказівки до виконання курсової роботи з дисципліни ―Програмування, частина 2 (Об’єктно-орієнтоване програмування)‖ для студентів першого (бакалаврського) рівня вищої освіти спеціальності 123 «Комп’ютерна інженерія»./ Укл.: Ю.В.Морозов, І.І. Пастернак – Львів: Видавництво Національного університету ―Львівська політехніка‖, 2017. – 29 с.

2. Прата С. Язык программирования C++. Лекции и упражнения. 6-е издание / Стивен Прата; пер. с англ. – М.: ООО «И.Д. Вильямс», 2012. – 1248 с.: ил. – Парал. тит. англ.

Додаток

Код програми з графічним інтерфейсом

Bank.h

#pragma once

#include<iostream>

#include<string>

#include<fstream>

class Bank

{

protected:

std::string path1 = "Storage.txt";

std::string firstName;

std::string lastName;

double money;

public:

Bank();

Bank(std::string);

void setFirstName(std::string);

std::string getFirstName();

void setLastName(std::string);

std::string getLastName();

void setMoney(double);

double getMoney();

void menu();

void storage();

void findId();

~Bank();

};

Bank.cpp

#include "Bank.h"

Bank::Bank()

{

}

Bank::Bank(std::string name)

{

this->firstName = name;

}

void Bank::setFirstName(std::string name)

{

this->firstName = name;

}

std::string Bank::getFirstName()

{

return firstName;

}

void Bank::setMoney(double money)

{

this->money = money;

}

double Bank::getMoney()

{

return money;

}

void Bank::menu()

{

std::string name;

//getline(cin, name);

setFirstName(name);

//cout << "Enter your money" << endl;

double money;

//cin >> money;

setMoney(money);

}

void Bank::findId()

{

}

Bank::~Bank()

{

}

ATM.h

#pragma once

#include "Bank.h"

#include<string>

#include<iostream>

using namespace std;

class ATM :

public Bank

{

protected:

string path1 = "Storage.txt";

string path2 = "Storage2.txt";

int numATM;

int pinATM;

double moneyATM;

public:

ATM();

ATM(double);

void setNumATM(int);

int getNumATM();

void setPinATM(int);

int getPinATM();

void setMoneyATM(double);

double getMoneyATM();

double terminal(double, double);

void Balanc();

double takeMoney(double, double);

~ATM();

};

ATM.cpp

#include "ATM.h"

ATM::ATM()

{

}

ATM::ATM(double money)

{

this->moneyATM = money;

}

void ATM::setNumATM(int num)

{

this->numATM = num;

}

int ATM::getNumATM()

{

return numATM;

}

void ATM::setPinATM(int pin)

{

this->pinATM = pin;

}

int ATM::getPinATM()

{

return pinATM;

}

void ATM::setMoneyATM(double money)

{

this->moneyATM = money;

}

double ATM::getMoneyATM()

{

return moneyATM;

}

double ATM::terminal(double moneyIn, double moneyOut)

{

double money;

money = moneyIn + moneyOut;

return money;

}

void ATM::Balanc()

{

cout << "Ваш баланс - ";

cout << moneyATM << endl;

}

double ATM::takeMoney(double moneyIn, double moneyOut)

{

double money;

money = moneyIn - moneyOut;

return money;

}

ATM::~ATM()

{

}

Card.h

#pragma once

#include "Bank.h"

#include<iostream>

#include<string>

#include <ctime>

#include <cstdlib>

class Card :

public Bank

{

protected:

int pinCard;

int cardNum;

double moneyCard;

public:

Card();

void setPinCard(int);

int getPinCard();

void setCardNum(int);

int getCardNum();

void setMoneyCard(double);

double getMoneyCard();

void createCard();

~Card();

};

Card.cpp

#include "Card.h"

Card::Card()

{

}

void Card::setPinCard(int pin)

{

this->pinCard = pin;

}

int Card::getPinCard()

{

return pinCard;

}

void Card::setCardNum(int num)

{

this->cardNum = num;

}

int Card::getCardNum()

{

return cardNum;

}

void Card::setMoneyCard(double money)

{

this->moneyCard = money;

}

double Card::getMoneyCard()

{

return moneyCard;

}

void Card::createCard()

{

int pin=0;

int num=0;

int x;

srand(time(NULL));

for (int i = 0; i < 4; i++) {

if (i != 0) pin \*= 10;

pin += rand() % 10;

}

setPinCard(pin);

std::cout << pin<<std::endl;

for (int i = 0; i < 8; i++) {

if (i != 0) num \*= 10;

num += rand() % 10;

}

setCardNum(num);

std::cout << num;

}

Card::~Card()

{

}

Client.h

#pragma once

#include<iostream>

#include"Card.h"

#include"ATM.h"

#include"Employee.h"

#include"Operations.h"

#include<string>

#include<fstream>

#include<sstream>

#include<cstdio>

using namespace std;

class Client :public Card, public ATM, public Employee

{

protected:

string path1 = "Storage.txt";

string path2 = "Storage2.txt";

string firstNameClient;

string lastNameClient;

double moneyClient;

int cardNumClient;

int pinClient;

bool id;

bool ac;

double moneyClientD;

double moneyClientC;

public:

Client();

Client(string);

Client(string firstNameClient, double);

void setFirstClientName(string);

string getFirstClientName();

void setLastClientName(string);

string getLastClientName();

void setId(bool);

bool getID();

void setAc();

bool getAc();

void setMoneyClient(double);

double getMoneyClient();

void setMoneyClientD(double);

double getMoneyClientD();

void setMoneyClientC(double);

double getMoneyClientC();

void setCardNumClient(int);

int getCardNumClient();

void setPinClient(int);

int getPinClient();

void registration(bool);

void revrite();

void menu();

void storage();

void start();

void storageOut();

void findId();

void terminalClient(double);

void atmClient(double);

bool atmCheck(double);

double operD(int, double);

double operC(int, double);

~Client();

};

Client.cpp

#include "Client.h"

Client::Client()

{

}

Client::Client(string name)

{

this->firstNameClient = name;

}

Client::Client(string firstNameClient, double money)

{

this->moneyClient = money;

}

void Client::setFirstClientName(string name)

{

this->firstNameClient = name;

}

string Client::getFirstClientName()

{

return firstNameClient;

}

void Client::setLastClientName(string name)

{

this->lastNameClient = name;

}

string Client::getLastClientName()

{

return lastNameClient;

}

void Client::setId(bool i)

{

this->id = i;

}

bool Client::getID()

{

return id;

}

void Client::setAc()

{

this->ac = operation();

}

bool Client::getAc()

{

return ac;

}

void Client::setMoneyClient(double money)

{

this->moneyClient = money;

}

double Client::getMoneyClient()

{

return moneyClient;

}

void Client::setMoneyClientD(double money)

{

this->moneyClientD = money;

}

double Client::getMoneyClientD()

{

return moneyClientD;

}

void Client::setMoneyClientC(double money)

{

this->moneyClientC = money;

}

double Client::getMoneyClientC()

{

return moneyClientC;

}

void Client::setCardNumClient(int num)

{

this->cardNumClient = num;

}

int Client::getCardNumClient()

{

return cardNumClient;

}

void Client::setPinClient(int pin)

{

this->pinClient = pin;

}

int Client::getPinClient()

{

return pinClient;

}

void Client::revrite()

{

string line;

string linebuff;

ofstream f2;

start();

remove("Storage2.txt");

f2.open(path2, ios\_base::app);

f2 << getFirstClientName() << " " << getLastClientName() << " " << getPinClient() << " " << getCardNumClient()<< " ";

f2 << getMoneyClient() << endl;

ifstream f1;

f1.open(path1);

while (!f1.eof()) {

getline(f1, line);

if (line.find(getFirstClientName()) == string::npos) {

f2 << line << endl;;

}

}

f1.close();

f2.close();

ofstream f3;

remove("Storage.txt");

f3.open(path1, ios\_base::app);

ifstream f4;

f4.open(path2);

while (!f4.eof()) {

getline(f4, line);

f3 << line << endl;

}

f3.close();

f4.close();

remove("form.txt");

}

void Client::storage()

{

remove("form.txt");

ofstream f2;

f2.open("form.txt", ios\_base::app);

f2 << firstNameClient << " " << lastNameClient << " " << pinClient << " " << cardNumClient << " ";

f2 << moneyClient << endl;

}

void Client::start()

{

string line;

string name;

string lastname;

int num;

int pin;

double money;

ifstream form;

form.open("form.txt");

getline(form, line);

stringstream ss(line);

ss >> name;

ss >> lastname;

ss >> pin;

ss >> num;

ss >> money;

setFirstClientName(name);

setLastClientName(lastname);

setCardNumClient(num);

setPinClient(pin);

setMoneyClient(money);

form.close();

}

void Client::findId()

{

ifstream f1;

f1.open(path1);

ifstream f2;

string nameid;

string line;

string name;

string lastname;

int num;

int pin;

double money;

bool t1 = 0;

while (!f1.eof()) {

getline(f1, line);

if (line.find(getFirstClientName()) != string::npos) {

t1++;

stringstream ss(line);

ss >> name;

ss >> lastname;

ss >> pin;

ss >> num;

ss >> money;

setFirstClientName(name);

setLastClientName(lastname);

setCardNumClient(num);

setPinClient(pin);

setMoneyClient(money);

ofstream f;

f.open("form.txt");

f << line;

f.close();

//cout << line << endl;

setId(true);

break;

}

}

if(t1==0) {

setId(false);

}

}

void Client::terminalClient(double in)

{

double m;

m=terminal(getMoneyClient(),in);

setMoneyClient(m);

storage();

}

void Client::atmClient(double in)

{

double m;

m = takeMoney(getMoneyClient(), in);

setMoneyClient(m);

storage();

}

bool Client::atmCheck(double money)

{

if (money > getMoneyClient())

{

return false;

}

if (money <= getMoneyClient())

{

return true;

}

}

double Client::operD(int time, double money)

{

double m;

m= Dep(time, money);

setMoneyClient(getMoneyClient() + getMoneyD());

storage();

return m;

}

double Client::operC(int time, double money)

{

double m;

double x;

m = Cre(time, money);

x = getMoneyClient()-m;

setMoneyClient(x);

storage();

return m;

}

Client::~Client()

{

//remove("form.txt");

}

Operations.h

#pragma once

#include "Bank.h"

class Operations :

public Bank

{

private:

double moneyD;

double moneyC;

public:

Operations();

double getMoneyD();

void setMoneyD(double);

double getMoneyC();

void setMoneyC(double);

double credit(int, double);

double deposit(int,double);

};

Operations.cpp

#include "Operations.h"

Operations::Operations()

{

}

double Operations::getMoneyD()

{

return moneyD;

}

void Operations::setMoneyD(double money)

{

this->moneyD = money;

}

double Operations::getMoneyC()

{

return 0.0;

}

void Operations::setMoneyC(double)

{

}

double Operations::credit(int time, double firstMoney)

{

float procent;

double moneyProc;

double credit;

double buf;

if (time == 12)

{

procent = 3;

moneyProc = firstMoney \* 3 / 100;

credit = (moneyProc \* 12) + firstMoney;

buf = credit - firstMoney;

setMoneyC(buf);

return credit;

}

if (time == 36)

{

procent = 8;

moneyProc = firstMoney \* 8 / 100;

credit = (moneyProc \* 36) + firstMoney;

buf =credit - firstMoney;

setMoneyC(buf);

return credit;

}

if (time == 72)

{

procent = 12;

moneyProc = firstMoney \* 12 / 100;

credit = (moneyProc \* 72) + firstMoney;

buf = credit - firstMoney;

setMoneyC(buf);

return credit;

}

}

double Operations::deposit(int time, double firstMoney)

{

float procent;

double moneyProc;

double deposit;

double buf;

if (time == 3)

{

procent = 4;

moneyProc =firstMoney \* 4 / 100;

deposit = (moneyProc \* 3)+firstMoney;

buf = deposit - firstMoney;

setMoneyD(buf);

return deposit;

}

if (time == 9)

{

procent = 11;

moneyProc = firstMoney \* 4 / 100;

deposit = (moneyProc \* 9) + firstMoney;

buf = deposit - firstMoney;

setMoneyD(buf);

return deposit;

}

if (time == 15)

{

procent = 20;

moneyProc = firstMoney \* 4 / 100;

deposit = (moneyProc \* 15) + firstMoney;

buf = deposit - firstMoney;

setMoneyD(buf);

return deposit;

}

}

Employee.h

#pragma once

#include "Operations.h"

#include <ctime>

#include <cstdlib>

class Employee :

public Operations

{

private:

double moneyEmplo;

bool acc;

public:

Employee();

void setMoneyEmplo(double);

double getMoneyEmplo();

void setAcc(bool);

bool getAcc();

bool operation();

void transfer();

double Dep(int, double);

double Cre(int, double);

};

Employee.cpp

#include "Employee.h"

Employee::Employee()

{

}

void Employee::setMoneyEmplo(double money)

{

this->moneyEmplo = money;

}

double Employee::getMoneyEmplo()

{

return moneyEmplo;

}

void Employee::setAcc(bool a)

{

this->acc = a;

}

bool Employee::getAcc()

{

return acc;

}

bool Employee::operation()

{

srand(time(NULL));

setAcc(rand() % 2 == 0);

if (getAcc() == true)

{

setAcc(true);

return true;

}

if (getAcc() == false)

{

setAcc(false);

return false;

}

}

void Employee::transfer()

{

}

double Employee::Dep(int time, double money)

{

return deposit(time, money);

}

double Employee::Cre(int time, double money)

{

return credit(time,money);

}

MyForm.h

#pragma once

#include"MyForm1.h"

#include"MyForm2.h"

#include <msclr\marshal\_cppstd.h>

#include"Client.h"

#include<string>

namespace BankCursova2 {

using namespace System;

using namespace System::ComponentModel;

using namespace System::Collections;

using namespace System::Windows::Forms;

using namespace System::Data;

using namespace System::Drawing;

/// <summary>

/// Сводка для MyForm

/// </summary>

public ref class MyForm : public System::Windows::Forms::Form

{

public:

MyForm(void)

{

InitializeComponent();

//

//TODO: добавьте код конструктора

//

}

protected:

/// <summary>

/// Освободить все используемые ресурсы.

/// </summary>

~MyForm()

{

if (components)

{

delete components;

}

}

private: System::Windows::Forms::Button^ button1;

protected:

private: System::Windows::Forms::TextBox^ textBox1;

private: System::Windows::Forms::Label^ label1;

private:

/// <summary>

/// Обязательная переменная конструктора.

/// </summary>

System::ComponentModel::Container ^components;

#pragma region Windows Form Designer generated code

/// <summary>

/// Требуемый метод для поддержки конструктора — не изменяйте

/// содержимое этого метода с помощью редактора кода.

/// </summary>

void InitializeComponent(void)

{

this->button1 = (gcnew System::Windows::Forms::Button());

this->textBox1 = (gcnew System::Windows::Forms::TextBox());

this->label1 = (gcnew System::Windows::Forms::Label());

this->SuspendLayout();

//

// button1

//

this->button1->Location = System::Drawing::Point(256, 171);

this->button1->Name = L"button1";

this->button1->Size = System::Drawing::Size(111, 44);

this->button1->TabIndex = 6;

this->button1->Text = L"Перевірити";

this->button1->UseVisualStyleBackColor = true;

this->button1->Click += gcnew System::EventHandler(this, &MyForm::Button1\_Click);

//

// textBox1

//

this->textBox1->Font = (gcnew System::Drawing::Font(L"Times New Roman", 12, System::Drawing::FontStyle::Regular, System::Drawing::GraphicsUnit::Point,

static\_cast<System::Byte>(204)));

this->textBox1->Location = System::Drawing::Point(239, 111);

this->textBox1->Name = L"textBox1";

this->textBox1->Size = System::Drawing::Size(150, 30);

this->textBox1->TabIndex = 5;

this->textBox1->Text = L"Ім\'я";

//

// label1

//

this->label1->AutoSize = true;

this->label1->BackColor = System::Drawing::SystemColors::HighlightText;

this->label1->Font = (gcnew System::Drawing::Font(L"Times New Roman", 12, System::Drawing::FontStyle::Regular, System::Drawing::GraphicsUnit::Point,

static\_cast<System::Byte>(204)));

this->label1->ForeColor = System::Drawing::Color::Black;

this->label1->Location = System::Drawing::Point(235, 68);

this->label1->Name = L"label1";

this->label1->Size = System::Drawing::Size(154, 22);

this->label1->TabIndex = 4;

this->label1->Text = L"Введіть своє ім\'я";

this->label1->TextAlign = System::Drawing::ContentAlignment::TopCenter;

//

// MyForm

//

this->AutoScaleDimensions = System::Drawing::SizeF(8, 16);

this->AutoScaleMode = System::Windows::Forms::AutoScaleMode::Font;

this->ClientSize = System::Drawing::Size(644, 460);

this->Controls->Add(this->button1);

this->Controls->Add(this->textBox1);

this->Controls->Add(this->label1);

this->Name = L"MyForm";

this->StartPosition = System::Windows::Forms::FormStartPosition::CenterScreen;

this->Text = L"MyForm";

this->ResumeLayout(false);

this->PerformLayout();

}

#pragma endregion

private: System::Void Button1\_Click(System::Object^ sender, System::EventArgs^ e) {

bool id=true;

std::string name = msclr::interop::marshal\_as<std::string>(textBox1->Text);

Client c(name);

c.findId();

if (false == c.getID()) {

this->Hide();

MyForm1^ f2 = gcnew MyForm1();

f2->Show();

}

if (true ==c.getID())

{

this->Hide();

MyForm2^ f3 = gcnew MyForm2();

f3->Show();

}

}

};

}

MyForm1.h

#pragma once

#include"Client.h"

#include"MyForm2.h"

#include <msclr\marshal\_cppstd.h>

namespace BankCursova2 {

using namespace System;

using namespace System::ComponentModel;

using namespace System::Collections;

using namespace System::Windows::Forms;

using namespace System::Data;

using namespace System::Drawing;

/// <summary>

/// Сводка для MyForm1

/// </summary>

public ref class MyForm1 : public System::Windows::Forms::Form

{

public:

MyForm1(void)

{

InitializeComponent();

//

//TODO: добавьте код конструктора

//

}

protected:

/// <summary>

/// Освободить все используемые ресурсы.

/// </summary>

~MyForm1()

{

if (components)

{

delete components;

}

}

private: System::Windows::Forms::Button^ button1;

protected:

private: System::Windows::Forms::CheckBox^ checkBox2;

private: System::Windows::Forms::CheckBox^ checkBox1;

private: System::Windows::Forms::Label^ label3;

private: System::Windows::Forms::TextBox^ textBox2;

private: System::Windows::Forms::TextBox^ textBox1;

private: System::Windows::Forms::Label^ label2;

private: System::Windows::Forms::Label^ label1;

private: System::Windows::Forms::TextBox^ textBox3;

private: System::Windows::Forms::Label^ label4;

private:

/// <summary>

/// Обязательная переменная конструктора.

/// </summary>

System::ComponentModel::Container ^components;

#pragma region Windows Form Designer generated code

/// <summary>

/// Требуемый метод для поддержки конструктора — не изменяйте

/// содержимое этого метода с помощью редактора кода.

/// </summary>

void InitializeComponent(void)

{

this->button1 = (gcnew System::Windows::Forms::Button());

this->checkBox2 = (gcnew System::Windows::Forms::CheckBox());

this->checkBox1 = (gcnew System::Windows::Forms::CheckBox());

this->label3 = (gcnew System::Windows::Forms::Label());

this->textBox2 = (gcnew System::Windows::Forms::TextBox());

this->textBox1 = (gcnew System::Windows::Forms::TextBox());

this->label2 = (gcnew System::Windows::Forms::Label());

this->label1 = (gcnew System::Windows::Forms::Label());

this->textBox3 = (gcnew System::Windows::Forms::TextBox());

this->label4 = (gcnew System::Windows::Forms::Label());

this->SuspendLayout();

//

// button1

//

this->button1->Font = (gcnew System::Drawing::Font(L"Times New Roman", 12));

this->button1->Location = System::Drawing::Point(246, 353);

this->button1->Name = L"button1";

this->button1->Size = System::Drawing::Size(129, 50);

this->button1->TabIndex = 15;

this->button1->Text = L"Закінчити";

this->button1->UseVisualStyleBackColor = true;

this->button1->Click += gcnew System::EventHandler(this, &MyForm1::Button1\_Click);

//

// checkBox2

//

this->checkBox2->AutoSize = true;

this->checkBox2->Font = (gcnew System::Drawing::Font(L"Times New Roman", 12, System::Drawing::FontStyle::Regular, System::Drawing::GraphicsUnit::Point,

static\_cast<System::Byte>(204)));

this->checkBox2->Location = System::Drawing::Point(493, 112);

this->checkBox2->Name = L"checkBox2";

this->checkBox2->Size = System::Drawing::Size(52, 26);

this->checkBox2->TabIndex = 14;

this->checkBox2->Text = L"Ні";

this->checkBox2->UseVisualStyleBackColor = true;

//

// checkBox1

//

this->checkBox1->AutoSize = true;

this->checkBox1->Font = (gcnew System::Drawing::Font(L"Times New Roman", 12, System::Drawing::FontStyle::Regular, System::Drawing::GraphicsUnit::Point,

static\_cast<System::Byte>(204)));

this->checkBox1->Location = System::Drawing::Point(374, 112);

this->checkBox1->Name = L"checkBox1";

this->checkBox1->Size = System::Drawing::Size(63, 26);

this->checkBox1->TabIndex = 13;

this->checkBox1->Text = L"Так";

this->checkBox1->UseVisualStyleBackColor = true;

this->checkBox1->CheckedChanged += gcnew System::EventHandler(this, &MyForm1::CheckBox1\_CheckedChanged);

//

// label3

//

this->label3->AutoSize = true;

this->label3->Font = (gcnew System::Drawing::Font(L"Times New Roman", 13.8F, System::Drawing::FontStyle::Regular, System::Drawing::GraphicsUnit::Point,

static\_cast<System::Byte>(204)));

this->label3->Location = System::Drawing::Point(369, 64);

this->label3->Name = L"label3";

this->label3->Size = System::Drawing::Size(194, 27);

this->label3->TabIndex = 12;

this->label3->Text = L"Створити картку\?";

//

// textBox2

//

this->textBox2->Font = (gcnew System::Drawing::Font(L"Times New Roman", 13.8F, System::Drawing::FontStyle::Regular, System::Drawing::GraphicsUnit::Point,

static\_cast<System::Byte>(204)));

this->textBox2->Location = System::Drawing::Point(54, 157);

this->textBox2->Name = L"textBox2";

this->textBox2->Size = System::Drawing::Size(150, 34);

this->textBox2->TabIndex = 11;

this->textBox2->Text = L"Прізвище";

this->textBox2->TextAlign = System::Windows::Forms::HorizontalAlignment::Center;

this->textBox2->TextChanged += gcnew System::EventHandler(this, &MyForm1::TextBox2\_TextChanged);

//

// textBox1

//

this->textBox1->Font = (gcnew System::Drawing::Font(L"Times New Roman", 13.8F, System::Drawing::FontStyle::Regular, System::Drawing::GraphicsUnit::Point,

static\_cast<System::Byte>(204)));

this->textBox1->Location = System::Drawing::Point(54, 104);

this->textBox1->Name = L"textBox1";

this->textBox1->Size = System::Drawing::Size(150, 34);

this->textBox1->TabIndex = 10;

this->textBox1->Text = L"Ім\'я";

this->textBox1->TextAlign = System::Windows::Forms::HorizontalAlignment::Center;

this->textBox1->TextChanged += gcnew System::EventHandler(this, &MyForm1::TextBox1\_TextChanged);

//

// label2

//

this->label2->AutoSize = true;

this->label2->Font = (gcnew System::Drawing::Font(L"Times New Roman", 13.8F, System::Drawing::FontStyle::Regular, System::Drawing::GraphicsUnit::Point,

static\_cast<System::Byte>(204)));

this->label2->Location = System::Drawing::Point(49, 64);

this->label2->Name = L"label2";

this->label2->Size = System::Drawing::Size(222, 27);

this->label2->TabIndex = 9;

this->label2->Text = L"Введіть англійською";

//

// label1

//

this->label1->AutoSize = true;

this->label1->Font = (gcnew System::Drawing::Font(L"Times New Roman", 13.8F, System::Drawing::FontStyle::Regular, System::Drawing::GraphicsUnit::Point,

static\_cast<System::Byte>(204)));

this->label1->Location = System::Drawing::Point(248, 26);

this->label1->Name = L"label1";

this->label1->Size = System::Drawing::Size(127, 27);

this->label1->TabIndex = 8;

this->label1->Text = L"Регістрація";

//

// textBox3

//

this->textBox3->Font = (gcnew System::Drawing::Font(L"Times New Roman", 13.8F, System::Drawing::FontStyle::Regular, System::Drawing::GraphicsUnit::Point,

static\_cast<System::Byte>(204)));

this->textBox3->Location = System::Drawing::Point(374, 208);

this->textBox3->Name = L"textBox3";

this->textBox3->Size = System::Drawing::Size(150, 34);

this->textBox3->TabIndex = 16;

this->textBox3->TextAlign = System::Windows::Forms::HorizontalAlignment::Center;

this->textBox3->TextChanged += gcnew System::EventHandler(this, &MyForm1::TextBox3\_TextChanged);

//

// label4

//

this->label4->AutoSize = true;

this->label4->Font = (gcnew System::Drawing::Font(L"Times New Roman", 13.8F, System::Drawing::FontStyle::Regular, System::Drawing::GraphicsUnit::Point,

static\_cast<System::Byte>(204)));

this->label4->Location = System::Drawing::Point(331, 164);

this->label4->Name = L"label4";

this->label4->Size = System::Drawing::Size(273, 27);

this->label4->TabIndex = 17;

this->label4->Text = L"Зробіть початковий вклад";

//

// MyForm1

//

this->AutoScaleDimensions = System::Drawing::SizeF(8, 16);

this->AutoScaleMode = System::Windows::Forms::AutoScaleMode::Font;

this->ClientSize = System::Drawing::Size(644, 460);

this->Controls->Add(this->label4);

this->Controls->Add(this->textBox3);

this->Controls->Add(this->button1);

this->Controls->Add(this->checkBox2);

this->Controls->Add(this->checkBox1);

this->Controls->Add(this->label3);

this->Controls->Add(this->textBox2);

this->Controls->Add(this->textBox1);

this->Controls->Add(this->label2);

this->Controls->Add(this->label1);

this->Name = L"MyForm1";

this->Text = L"MyForm1";

this->Load += gcnew System::EventHandler(this, &MyForm1::MyForm1\_Load);

this->ResumeLayout(false);

this->PerformLayout();

}

#pragma endregion

private: System::Void TextBox2\_TextChanged(System::Object^ sender, System::EventArgs^ e) {

}

private: System::Void TextBox1\_TextChanged(System::Object^ sender, System::EventArgs^ e) {

}

private: System::Void CheckBox1\_CheckedChanged(System::Object^ sender, System::EventArgs^ e) {

}

private: System::Void Button1\_Click(System::Object^ sender, System::EventArgs^ e) {

if (checkBox1->Checked == true)

{

string name=msclr::interop::marshal\_as<std::string>(textBox1->Text);

string last =msclr::interop::marshal\_as<std::string>(textBox2->Text);

double money =System::Convert::ToDouble(textBox3->Text);

remove("form.txt");

int pin = 0;

int num = 0;

int x;

srand(time(NULL));

for (int i = 0; i < 4; i++) {

if (i != 0) pin \*= 10;

pin += rand() % 10;

}

for (int i = 0; i < 8; i++) {

if (i != 0) num \*= 10;

num += rand() % 10;

}

ofstream f2;

f2.open("form.txt", ios\_base::app);

f2 << name << " " << last << " " << pin << " " <<num << " ";

f2 << money << endl;

}

this->Hide();

MyForm2^ f3 = gcnew MyForm2;

f3->Show();

//Application::Exit();

}

private: System::Void TextBox3\_TextChanged(System::Object^ sender, System::EventArgs^ e) {

}

private: System::Void MyForm1\_Load(System::Object^ sender, System::EventArgs^ e) {

}

};

}

MyForm2.h

#pragma once

#include"Client.h"

#include"MyForm3.h"

#include <msclr\marshal\_cppstd.h>

using namespace System;

//using namespace msclr::interop;

namespace BankCursova2 {

using namespace System;

using namespace System::ComponentModel;

using namespace System::Collections;

using namespace System::Windows::Forms;

using namespace System::Data;

using namespace System::Drawing;

/// <summary>

/// Сводка для MyForm2

/// </summary>

public ref class MyForm2 : public System::Windows::Forms::Form

{

public:

MyForm2(void)

{

InitializeComponent();

//

//TODO: добавьте код конструктора

//

}

protected:

/// <summary>

/// Освободить все используемые ресурсы.

/// </summary>

~MyForm2()

{

if (components)

{

delete components;

}

}

private: System::Windows::Forms::Button^ button1;

private: System::Windows::Forms::Label^ label1;

private: System::Windows::Forms::Label^ label2;

private: System::Windows::Forms::Label^ label3;

private: System::Windows::Forms::Label^ label4;

private: System::Windows::Forms::Label^ label5;

private: System::Windows::Forms::TextBox^ textBox1;

private: System::Windows::Forms::TextBox^ textBox2;

private: System::Windows::Forms::TextBox^ textBox3;

private: System::Windows::Forms::TextBox^ textBox4;

private: System::Windows::Forms::TextBox^ textBox5;

protected:

private:

/// <summary>

/// Обязательная переменная конструктора.

/// </summary>

System::ComponentModel::Container ^components;

#pragma region Windows Form Designer generated code

/// <summary>

/// Требуемый метод для поддержки конструктора — не изменяйте

/// содержимое этого метода с помощью редактора кода.

/// </summary>

void InitializeComponent(void)

{

this->button1 = (gcnew System::Windows::Forms::Button());

this->label1 = (gcnew System::Windows::Forms::Label());

this->label2 = (gcnew System::Windows::Forms::Label());

this->label3 = (gcnew System::Windows::Forms::Label());

this->label4 = (gcnew System::Windows::Forms::Label());

this->label5 = (gcnew System::Windows::Forms::Label());

this->textBox1 = (gcnew System::Windows::Forms::TextBox());

this->textBox2 = (gcnew System::Windows::Forms::TextBox());

this->textBox3 = (gcnew System::Windows::Forms::TextBox());

this->textBox4 = (gcnew System::Windows::Forms::TextBox());

this->textBox5 = (gcnew System::Windows::Forms::TextBox());

this->SuspendLayout();

//

// button1

//

this->button1->Location = System::Drawing::Point(250, 325);

this->button1->Name = L"button1";

this->button1->Size = System::Drawing::Size(141, 53);

this->button1->TabIndex = 0;

this->button1->Text = L"Продовжити";

this->button1->UseVisualStyleBackColor = true;

this->button1->Click += gcnew System::EventHandler(this, &MyForm2::Button1\_Click);

//

// label1

//

this->label1->AutoSize = true;

this->label1->Font = (gcnew System::Drawing::Font(L"Times New Roman", 12, System::Drawing::FontStyle::Regular, System::Drawing::GraphicsUnit::Point,

static\_cast<System::Byte>(204)));

this->label1->Location = System::Drawing::Point(67, 54);

this->label1->Name = L"label1";

this->label1->Size = System::Drawing::Size(42, 22);

this->label1->TabIndex = 1;

this->label1->Text = L"Ім\'я";

this->label1->Click += gcnew System::EventHandler(this, &MyForm2::Label1\_Click);

//

// label2

//

this->label2->AutoSize = true;

this->label2->Font = (gcnew System::Drawing::Font(L"Times New Roman", 12, System::Drawing::FontStyle::Regular, System::Drawing::GraphicsUnit::Point,

static\_cast<System::Byte>(204)));

this->label2->Location = System::Drawing::Point(67, 132);

this->label2->Name = L"label2";

this->label2->Size = System::Drawing::Size(120, 22);

this->label2->TabIndex = 2;

this->label2->Text = L"Номер карти";

//

// label3

//

this->label3->AutoSize = true;

this->label3->Font = (gcnew System::Drawing::Font(L"Times New Roman", 12, System::Drawing::FontStyle::Regular, System::Drawing::GraphicsUnit::Point,

static\_cast<System::Byte>(204)));

this->label3->Location = System::Drawing::Point(67, 95);

this->label3->Name = L"label3";

this->label3->Size = System::Drawing::Size(91, 22);

this->label3->TabIndex = 3;

this->label3->Text = L"Прізвище";

//

// label4

//

this->label4->AutoSize = true;

this->label4->Font = (gcnew System::Drawing::Font(L"Times New Roman", 12, System::Drawing::FontStyle::Regular, System::Drawing::GraphicsUnit::Point,

static\_cast<System::Byte>(204)));

this->label4->Location = System::Drawing::Point(67, 172);

this->label4->Name = L"label4";

this->label4->Size = System::Drawing::Size(90, 22);

this->label4->TabIndex = 4;

this->label4->Text = L"Pin карти";

this->label4->Click += gcnew System::EventHandler(this, &MyForm2::Label4\_Click);

//

// label5

//

this->label5->AutoSize = true;

this->label5->Font = (gcnew System::Drawing::Font(L"Times New Roman", 12, System::Drawing::FontStyle::Bold, System::Drawing::GraphicsUnit::Point,

static\_cast<System::Byte>(204)));

this->label5->Location = System::Drawing::Point(67, 243);

this->label5->Name = L"label5";

this->label5->Size = System::Drawing::Size(128, 23);

this->label5->TabIndex = 5;

this->label5->Text = L"Ваш рахунок";

//

// textBox1

//

this->textBox1->BackColor = System::Drawing::SystemColors::Control;

this->textBox1->Location = System::Drawing::Point(250, 54);

this->textBox1->Name = L"textBox1";

this->textBox1->Size = System::Drawing::Size(141, 22);

this->textBox1->TabIndex = 6;

this->textBox1->TextChanged += gcnew System::EventHandler(this, &MyForm2::TextBox1\_TextChanged);

//

// textBox2

//

this->textBox2->BackColor = System::Drawing::SystemColors::Control;

this->textBox2->Location = System::Drawing::Point(250, 173);

this->textBox2->Name = L"textBox2";

this->textBox2->Size = System::Drawing::Size(141, 22);

this->textBox2->TabIndex = 7;

//

// textBox3

//

this->textBox3->BackColor = System::Drawing::SystemColors::Control;

this->textBox3->Location = System::Drawing::Point(250, 132);

this->textBox3->Name = L"textBox3";

this->textBox3->Size = System::Drawing::Size(141, 22);

this->textBox3->TabIndex = 8;

//

// textBox4

//

this->textBox4->BackColor = System::Drawing::SystemColors::Control;

this->textBox4->Location = System::Drawing::Point(250, 95);

this->textBox4->Name = L"textBox4";

this->textBox4->Size = System::Drawing::Size(141, 22);

this->textBox4->TabIndex = 9;

this->textBox4->TextChanged += gcnew System::EventHandler(this, &MyForm2::TextBox4\_TextChanged);

//

// textBox5

//

this->textBox5->BackColor = System::Drawing::SystemColors::Control;

this->textBox5->Location = System::Drawing::Point(250, 243);

this->textBox5->Name = L"textBox5";

this->textBox5->Size = System::Drawing::Size(141, 22);

this->textBox5->TabIndex = 10;

this->textBox5->TextChanged += gcnew System::EventHandler(this, &MyForm2::TextBox5\_TextChanged);

//

// MyForm2

//

this->AutoScaleDimensions = System::Drawing::SizeF(8, 16);

this->AutoScaleMode = System::Windows::Forms::AutoScaleMode::Font;

this->ClientSize = System::Drawing::Size(644, 460);

this->Controls->Add(this->textBox5);

this->Controls->Add(this->textBox4);

this->Controls->Add(this->textBox3);

this->Controls->Add(this->textBox2);

this->Controls->Add(this->textBox1);

this->Controls->Add(this->label5);

this->Controls->Add(this->label4);

this->Controls->Add(this->label3);

this->Controls->Add(this->label2);

this->Controls->Add(this->label1);

this->Controls->Add(this->button1);

this->Name = L"MyForm2";

this->StartPosition = System::Windows::Forms::FormStartPosition::CenterScreen;

this->Text = L"MyForm2";

this->Load += gcnew System::EventHandler(this, &MyForm2::MyForm2\_Load);

this->ResumeLayout(false);

this->PerformLayout();

}

#pragma endregion

private: System::Void Button1\_Click(System::Object^ sender, System::EventArgs^ e) {

this->Hide();

MyForm3^ f4 = gcnew MyForm3;

f4->ShowDialog();

this->Show();

Client c;

c.start();

textBox4->Text = msclr::interop::marshal\_as<System::String^>(c.getLastClientName());

textBox3->Text = System::Convert::ToString(c.getCardNumClient());

textBox2->Text = System::Convert::ToString(c.getPinClient());

textBox1->Text = msclr::interop::marshal\_as<System::String^>(c.getFirstClientName());

textBox5->Text = System::Convert::ToString(c.getMoneyClient());

}

private: System::Void Label1\_Click(System::Object^ sender, System::EventArgs^ e) {

}

private: System::Void Label4\_Click(System::Object^ sender, System::EventArgs^ e) {

}

private: System::Void TextBox1\_TextChanged(System::Object^ sender, System::EventArgs^ e) {

}

private: System::Void TextBox5\_TextChanged(System::Object^ sender, System::EventArgs^ e) {

}

private: System::Void TextBox4\_TextChanged(System::Object^ sender, System::EventArgs^ e) {

}

private: System::Void MyForm2\_Load(System::Object^ sender, System::EventArgs^ e) {

Client c;

c.start();

textBox4->Text = msclr::interop::marshal\_as<System::String^>(c.getLastClientName());

textBox3->Text = System::Convert::ToString(c.getCardNumClient());

textBox2->Text= System::Convert::ToString(c.getPinClient());

textBox1->Text = msclr::interop::marshal\_as<System::String^>(c.getFirstClientName());

textBox5->Text = System::Convert::ToString(c.getMoneyClient());

}

};

}

MyForm3.h

#pragma once

#include"MyForm4.h"

#include"MyForm5.h"

#include"Choise.h"

namespace BankCursova2 {

using namespace System;

using namespace System::ComponentModel;

using namespace System::Collections;

using namespace System::Windows::Forms;

using namespace System::Data;

using namespace System::Drawing;

/// <summary>

/// Сводка для MyForm3

/// </summary>

public ref class MyForm3 : public System::Windows::Forms::Form

{

public:

MyForm3(void)

{

InitializeComponent();

//

//TODO: добавьте код конструктора

//

}

protected:

/// <summary>

/// Освободить все используемые ресурсы.

/// </summary>

~MyForm3()

{

if (components)

{

delete components;

}

}

private: System::Windows::Forms::Button^ button1;

protected:

private: System::Windows::Forms::Button^ button2;

private: System::Windows::Forms::Button^ button3;

private: System::Windows::Forms::Button^ button4;

private: System::Windows::Forms::Button^ button6;

private:

/// <summary>

/// Обязательная переменная конструктора.

/// </summary>

System::ComponentModel::Container ^components;

#pragma region Windows Form Designer generated code

/// <summary>

/// Требуемый метод для поддержки конструктора — не изменяйте

/// содержимое этого метода с помощью редактора кода.

/// </summary>

void InitializeComponent(void)

{

this->button1 = (gcnew System::Windows::Forms::Button());

this->button2 = (gcnew System::Windows::Forms::Button());

this->button3 = (gcnew System::Windows::Forms::Button());

this->button4 = (gcnew System::Windows::Forms::Button());

this->button6 = (gcnew System::Windows::Forms::Button());

this->SuspendLayout();

//

// button1

//

this->button1->Font = (gcnew System::Drawing::Font(L"Times New Roman", 19.8F, System::Drawing::FontStyle::Regular, System::Drawing::GraphicsUnit::Point,

static\_cast<System::Byte>(204)));

this->button1->Location = System::Drawing::Point(12, 141);

this->button1->Name = L"button1";

this->button1->Size = System::Drawing::Size(188, 92);

this->button1->TabIndex = 0;

this->button1->Text = L"Банківскі послуги";

this->button1->UseVisualStyleBackColor = true;

this->button1->Click += gcnew System::EventHandler(this, &MyForm3::Button1\_Click);

//

// button2

//

this->button2->Font = (gcnew System::Drawing::Font(L"Times New Roman", 19.8F, System::Drawing::FontStyle::Regular, System::Drawing::GraphicsUnit::Point,

static\_cast<System::Byte>(204)));

this->button2->Location = System::Drawing::Point(229, 141);

this->button2->Name = L"button2";

this->button2->Size = System::Drawing::Size(188, 92);

this->button2->TabIndex = 1;

this->button2->Text = L"Банкомат";

this->button2->UseVisualStyleBackColor = true;

this->button2->Click += gcnew System::EventHandler(this, &MyForm3::Button2\_Click);

//

// button3

//

this->button3->Font = (gcnew System::Drawing::Font(L"Times New Roman", 19.8F, System::Drawing::FontStyle::Regular, System::Drawing::GraphicsUnit::Point,

static\_cast<System::Byte>(204)));

this->button3->Location = System::Drawing::Point(444, 141);

this->button3->Name = L"button3";

this->button3->Size = System::Drawing::Size(188, 92);

this->button3->TabIndex = 2;

this->button3->Text = L"Термінал";

this->button3->UseVisualStyleBackColor = true;

this->button3->Click += gcnew System::EventHandler(this, &MyForm3::Button3\_Click);

//

// button4

//

this->button4->Font = (gcnew System::Drawing::Font(L"Times New Roman", 19.8F, System::Drawing::FontStyle::Regular, System::Drawing::GraphicsUnit::Point,

static\_cast<System::Byte>(204)));

this->button4->Location = System::Drawing::Point(229, 331);

this->button4->Name = L"button4";

this->button4->Size = System::Drawing::Size(188, 92);

this->button4->TabIndex = 3;

this->button4->Text = L"Баланс";

this->button4->UseVisualStyleBackColor = true;

this->button4->Click += gcnew System::EventHandler(this, &MyForm3::Button4\_Click);

//

// button6

//

this->button6->Location = System::Drawing::Point(12, 12);

this->button6->Name = L"button6";

this->button6->Size = System::Drawing::Size(75, 23);

this->button6->TabIndex = 5;

this->button6->Text = L"Вихід";

this->button6->UseVisualStyleBackColor = true;

this->button6->Click += gcnew System::EventHandler(this, &MyForm3::Button6\_Click);

//

// MyForm3

//

this->AutoScaleDimensions = System::Drawing::SizeF(8, 16);

this->AutoScaleMode = System::Windows::Forms::AutoScaleMode::Font;

this->ClientSize = System::Drawing::Size(644, 460);

this->Controls->Add(this->button6);

this->Controls->Add(this->button4);

this->Controls->Add(this->button3);

this->Controls->Add(this->button2);

this->Controls->Add(this->button1);

this->Name = L"MyForm3";

this->StartPosition = System::Windows::Forms::FormStartPosition::CenterScreen;

this->Text = L"MyForm3";

this->ResumeLayout(false);

}

#pragma endregion

private: System::Void Button3\_Click(System::Object^ sender, System::EventArgs^ e) {

this->Hide();

MyForm4^ Atm = gcnew MyForm4;

Atm->ShowDialog();

this->Show();

}

private: System::Void Button2\_Click(System::Object^ sender, System::EventArgs^ e) {

this->Hide();

MyForm5^ Atm = gcnew MyForm5;

Atm->ShowDialog();

this->Show();

}

private: System::Void Button1\_Click(System::Object^ sender, System::EventArgs^ e) {

this->Hide();

Choise^ c = gcnew Choise;

c->ShowDialog();

this->Show();

}

private: System::Void Button4\_Click(System::Object^ sender, System::EventArgs^ e) {

this->Close();

}

private: System::Void Button6\_Click(System::Object^ sender, System::EventArgs^ e) {

Client c;

c.revrite();

Application::Exit();

}

};

}

MyForm4.h

#pragma once

#include"Client.h"

//#include"MyForm3.h"

namespace BankCursova2 {

using namespace System;

using namespace System::ComponentModel;

using namespace System::Collections;

using namespace System::Windows::Forms;

using namespace System::Data;

using namespace System::Drawing;

/// <summary>

/// Сводка для MyForm4

/// </summary>

public ref class MyForm4 : public System::Windows::Forms::Form

{

public:

MyForm4(void)

{

InitializeComponent();

//

//TODO: добавьте код конструктора

//

}

protected:

/// <summary>

/// Освободить все используемые ресурсы.

/// </summary>

~MyForm4()

{

if (components)

{

delete components;

}

}

private: System::Windows::Forms::Button^ button1;

private: System::Windows::Forms::TextBox^ textBox1;

private: System::Windows::Forms::TextBox^ textBox2;

private: System::Windows::Forms::Panel^ panel1;

private: System::Windows::Forms::Button^ button2;

private: System::Windows::Forms::TextBox^ textBox4;

private: System::Windows::Forms::Label^ label4;

private: System::Windows::Forms::Panel^ panel2;

private: System::Windows::Forms::TextBox^ textBox3;

private: System::Windows::Forms::Label^ label3;

private: System::Windows::Forms::Button^ button3;

protected:

private:

/// <summary>

/// Обязательная переменная конструктора.

/// </summary>

System::ComponentModel::Container ^components;

#pragma region Windows Form Designer generated code

/// <summary>

/// Требуемый метод для поддержки конструктора — не изменяйте

/// содержимое этого метода с помощью редактора кода.

/// </summary>

void InitializeComponent(void)

{

System::Windows::Forms::Label^ label1;

System::Windows::Forms::Label^ label2;

this->button1 = (gcnew System::Windows::Forms::Button());

this->textBox1 = (gcnew System::Windows::Forms::TextBox());

this->textBox2 = (gcnew System::Windows::Forms::TextBox());

this->panel1 = (gcnew System::Windows::Forms::Panel());

this->textBox4 = (gcnew System::Windows::Forms::TextBox());

this->label4 = (gcnew System::Windows::Forms::Label());

this->button2 = (gcnew System::Windows::Forms::Button());

this->panel2 = (gcnew System::Windows::Forms::Panel());

this->button3 = (gcnew System::Windows::Forms::Button());

this->textBox3 = (gcnew System::Windows::Forms::TextBox());

this->label3 = (gcnew System::Windows::Forms::Label());

label1 = (gcnew System::Windows::Forms::Label());

label2 = (gcnew System::Windows::Forms::Label());

this->panel1->SuspendLayout();

this->panel2->SuspendLayout();

this->SuspendLayout();

//

// label1

//

label1->AutoSize = true;

label1->Font = (gcnew System::Drawing::Font(L"Times New Roman", 16.2F, System::Drawing::FontStyle::Regular, System::Drawing::GraphicsUnit::Point,

static\_cast<System::Byte>(204)));

label1->Location = System::Drawing::Point(30, 80);

label1->Name = L"label1";

label1->Size = System::Drawing::Size(219, 33);

label1->TabIndex = 1;

label1->Text = L"Введіть pin код - ";

//

// label2

//

label2->AutoSize = true;

label2->Font = (gcnew System::Drawing::Font(L"Times New Roman", 16.2F, System::Drawing::FontStyle::Regular, System::Drawing::GraphicsUnit::Point,

static\_cast<System::Byte>(204)));

label2->Location = System::Drawing::Point(46, 223);

label2->Name = L"label2";

label2->Size = System::Drawing::Size(181, 33);

label2->TabIndex = 3;

label2->Text = L"Ваш pin код - ";

//

// button1

//

this->button1->Font = (gcnew System::Drawing::Font(L"Times New Roman", 13.8F, System::Drawing::FontStyle::Regular, System::Drawing::GraphicsUnit::Point,

static\_cast<System::Byte>(204)));

this->button1->Location = System::Drawing::Point(520, 366);

this->button1->Name = L"button1";

this->button1->Size = System::Drawing::Size(90, 45);

this->button1->TabIndex = 0;

this->button1->Text = L"Далі";

this->button1->UseVisualStyleBackColor = true;

this->button1->Click += gcnew System::EventHandler(this, &MyForm4::Button1\_Click);

//

// textBox1

//

this->textBox1->Font = (gcnew System::Drawing::Font(L"Times New Roman", 16.2F, System::Drawing::FontStyle::Regular, System::Drawing::GraphicsUnit::Point,

static\_cast<System::Byte>(204)));

this->textBox1->Location = System::Drawing::Point(285, 80);

this->textBox1->Name = L"textBox1";

this->textBox1->Size = System::Drawing::Size(141, 39);

this->textBox1->TabIndex = 2;

//

// textBox2

//

this->textBox2->BackColor = System::Drawing::SystemColors::Control;

this->textBox2->BorderStyle = System::Windows::Forms::BorderStyle::None;

this->textBox2->Font = (gcnew System::Drawing::Font(L"Times New Roman", 16.2F, System::Drawing::FontStyle::Regular, System::Drawing::GraphicsUnit::Point,

static\_cast<System::Byte>(204)));

this->textBox2->Location = System::Drawing::Point(285, 223);

this->textBox2->Name = L"textBox2";

this->textBox2->Size = System::Drawing::Size(141, 32);

this->textBox2->TabIndex = 4;

this->textBox2->TextChanged += gcnew System::EventHandler(this, &MyForm4::TextBox2\_TextChanged);

//

// panel1

//

this->panel1->Controls->Add(this->panel2);

this->panel1->Controls->Add(this->textBox4);

this->panel1->Controls->Add(this->label4);

this->panel1->Controls->Add(this->button2);

this->panel1->Dock = System::Windows::Forms::DockStyle::Fill;

this->panel1->Location = System::Drawing::Point(0, 0);

this->panel1->Name = L"panel1";

this->panel1->Size = System::Drawing::Size(644, 460);

this->panel1->TabIndex = 5;

this->panel1->Visible = false;

this->panel1->Paint += gcnew System::Windows::Forms::PaintEventHandler(this, &MyForm4::Panel1\_Paint\_1);

//

// textBox4

//

this->textBox4->BackColor = System::Drawing::SystemColors::Control;

this->textBox4->BorderStyle = System::Windows::Forms::BorderStyle::FixedSingle;

this->textBox4->Font = (gcnew System::Drawing::Font(L"Times New Roman", 16.2F, System::Drawing::FontStyle::Regular, System::Drawing::GraphicsUnit::Point,

static\_cast<System::Byte>(204)));

this->textBox4->Location = System::Drawing::Point(271, 145);

this->textBox4->Name = L"textBox4";

this->textBox4->Size = System::Drawing::Size(220, 39);

this->textBox4->TabIndex = 4;

//

// label4

//

this->label4->AutoSize = true;

this->label4->Font = (gcnew System::Drawing::Font(L"Times New Roman", 16.2F, System::Drawing::FontStyle::Regular, System::Drawing::GraphicsUnit::Point,

static\_cast<System::Byte>(204)));

this->label4->Location = System::Drawing::Point(30, 127);

this->label4->Name = L"label4";

this->label4->Size = System::Drawing::Size(182, 66);

this->label4->TabIndex = 3;

this->label4->Text = L"Вкажіть суму \r\nпоповнення - ";

this->label4->TextAlign = System::Drawing::ContentAlignment::MiddleCenter;

//

// button2

//

this->button2->Font = (gcnew System::Drawing::Font(L"Times New Roman", 16.2F, System::Drawing::FontStyle::Regular, System::Drawing::GraphicsUnit::Point,

static\_cast<System::Byte>(204)));

this->button2->Location = System::Drawing::Point(513, 388);

this->button2->Name = L"button2";

this->button2->Size = System::Drawing::Size(119, 60);

this->button2->TabIndex = 0;

this->button2->Text = L"Далі";

this->button2->UseVisualStyleBackColor = true;

this->button2->Click += gcnew System::EventHandler(this, &MyForm4::Button2\_Click);

//

// panel2

//

this->panel2->Controls->Add(this->button3);

this->panel2->Controls->Add(this->label3);

this->panel2->Controls->Add(this->textBox3);

this->panel2->Dock = System::Windows::Forms::DockStyle::Fill;

this->panel2->Location = System::Drawing::Point(0, 0);

this->panel2->Name = L"panel2";

this->panel2->Size = System::Drawing::Size(644, 460);

this->panel2->TabIndex = 5;

this->panel2->Visible = false;

//

// button3

//

this->button3->Font = (gcnew System::Drawing::Font(L"Times New Roman", 16.2F, System::Drawing::FontStyle::Regular, System::Drawing::GraphicsUnit::Point,

static\_cast<System::Byte>(204)));

this->button3->Location = System::Drawing::Point(193, 296);

this->button3->Name = L"button3";

this->button3->Size = System::Drawing::Size(257, 115);

this->button3->TabIndex = 2;

this->button3->Text = L"Повернутись до вибору дій";

this->button3->UseVisualStyleBackColor = true;

this->button3->Click += gcnew System::EventHandler(this, &MyForm4::Button3\_Click\_1);

//

// textBox3

//

this->textBox3->BackColor = System::Drawing::SystemColors::Control;

this->textBox3->BorderStyle = System::Windows::Forms::BorderStyle::None;

this->textBox3->Font = (gcnew System::Drawing::Font(L"Times New Roman", 16.2F, System::Drawing::FontStyle::Regular, System::Drawing::GraphicsUnit::Point,

static\_cast<System::Byte>(204)));

this->textBox3->Location = System::Drawing::Point(285, 94);

this->textBox3->Name = L"textBox3";

this->textBox3->Size = System::Drawing::Size(185, 32);

this->textBox3->TabIndex = 1;

this->textBox3->TextChanged += gcnew System::EventHandler(this, &MyForm4::TextBox3\_TextChanged);

//

// label3

//

this->label3->AutoSize = true;

this->label3->Font = (gcnew System::Drawing::Font(L"Times New Roman", 16.2F, System::Drawing::FontStyle::Regular, System::Drawing::GraphicsUnit::Point,

static\_cast<System::Byte>(204)));

this->label3->Location = System::Drawing::Point(46, 94);

this->label3->Name = L"label3";

this->label3->Size = System::Drawing::Size(193, 33);

this->label3->TabIndex = 0;

this->label3->Text = L"Ваш рахунок - ";

//

// MyForm4

//

this->AutoScaleDimensions = System::Drawing::SizeF(8, 16);

this->AutoScaleMode = System::Windows::Forms::AutoScaleMode::Font;

this->ClientSize = System::Drawing::Size(644, 460);

this->Controls->Add(this->panel1);

this->Controls->Add(this->textBox2);

this->Controls->Add(label2);

this->Controls->Add(this->textBox1);

this->Controls->Add(label1);

this->Controls->Add(this->button1);

this->Name = L"MyForm4";

this->StartPosition = System::Windows::Forms::FormStartPosition::CenterScreen;

this->Text = L"MyForm4";

this->Load += gcnew System::EventHandler(this, &MyForm4::MyForm4\_Load);

this->panel1->ResumeLayout(false);

this->panel1->PerformLayout();

this->panel2->ResumeLayout(false);

this->panel2->PerformLayout();

this->ResumeLayout(false);

this->PerformLayout();

}

#pragma endregion

private: System::Void Button1\_Click(System::Object^ sender, System::EventArgs^ e) {

panel1->Visible = true;

}

private: System::Void Panel1\_Paint(System::Object^ sender, System::Windows::Forms::PaintEventArgs^ e) {

}

private: System::Void TextBox2\_TextChanged(System::Object^ sender, System::EventArgs^ e) {

}

private: System::Void MyForm4\_Load(System::Object^ sender, System::EventArgs^ e) {

Client c;

c.start();

textBox2->Text = System::Convert::ToString(c.getPinClient());

textBox3->Text= System::Convert::ToString(c.getMoneyClient());

}

private: System::Void Button2\_Click(System::Object^ sender, System::EventArgs^ e) {

double money;

Client c;

c.start();

money=System::Convert::ToDouble(textBox4->Text);

c.terminalClient(money);

panel2->Visible = true;

textBox3->Text = System::Convert::ToString(c.getMoneyClient());

}

private: System::Void Panel1\_Paint\_1(System::Object^ sender, System::Windows::Forms::PaintEventArgs^ e) {

}

private: System::Void Label5\_Click(System::Object^ sender, System::EventArgs^ e) {

}

private: System::Void Button3\_Click(System::Object^ sender, System::EventArgs^ e) {

//this->Close();

//MyForm3^ f = gcnew MyForm3;

//f->Show();

}

private: System::Void TextBox5\_TextChanged(System::Object^ sender, System::EventArgs^ e) {

}

private: System::Void TextBox3\_TextChanged(System::Object^ sender, System::EventArgs^ e) {

}

private: System::Void Button3\_Click\_1(System::Object^ sender, System::EventArgs^ e) {

this->Close();

}

private: System::Void TextBox5\_TextChanged\_1(System::Object^ sender, System::EventArgs^ e) {

}

private: System::Void Label5\_Click\_1(System::Object^ sender, System::EventArgs^ e) {

}

};

}

MyForm5.h

#pragma once

#include"Client.h"

namespace BankCursova2 {

using namespace System;

using namespace System::ComponentModel;

using namespace System::Collections;

using namespace System::Windows::Forms;

using namespace System::Data;

using namespace System::Drawing;

/// <summary>

/// Сводка для MyForm5

/// </summary>

public ref class MyForm5 : public System::Windows::Forms::Form

{

public:

MyForm5(void)

{

InitializeComponent();

//

//TODO: добавьте код конструктора

//

}

protected:

/// <summary>

/// Освободить все используемые ресурсы.

/// </summary>

~MyForm5()

{

if (components)

{

delete components;

}

}

private: System::Windows::Forms::Label^ label1;

protected:

private: System::Windows::Forms::TextBox^ textBox1;

private: System::Windows::Forms::Label^ label2;

private: System::Windows::Forms::TextBox^ textBox2;

private: System::Windows::Forms::Button^ button1;

private: System::Windows::Forms::Panel^ panel1;

private: System::Windows::Forms::TextBox^ textBox3;

private: System::Windows::Forms::Label^ label3;

private: System::Windows::Forms::Button^ button2;

private: System::Windows::Forms::TextBox^ textBox4;

private: System::Windows::Forms::Label^ label4;

private: System::Windows::Forms::Panel^ panel2;

private: System::Windows::Forms::Label^ label5;

private: System::Windows::Forms::Button^ button3;

private: System::Windows::Forms::TextBox^ textBox5;

private: System::Windows::Forms::Label^ label6;

private:

/// <summary>

/// Обязательная переменная конструктора.

/// </summary>

System::ComponentModel::Container ^components;

#pragma region Windows Form Designer generated code

/// <summary>

/// Требуемый метод для поддержки конструктора — не изменяйте

/// содержимое этого метода с помощью редактора кода.

/// </summary>

void InitializeComponent(void)

{

this->label1 = (gcnew System::Windows::Forms::Label());

this->textBox1 = (gcnew System::Windows::Forms::TextBox());

this->label2 = (gcnew System::Windows::Forms::Label());

this->textBox2 = (gcnew System::Windows::Forms::TextBox());

this->button1 = (gcnew System::Windows::Forms::Button());

this->panel1 = (gcnew System::Windows::Forms::Panel());

this->panel2 = (gcnew System::Windows::Forms::Panel());

this->button3 = (gcnew System::Windows::Forms::Button());

this->textBox5 = (gcnew System::Windows::Forms::TextBox());

this->label6 = (gcnew System::Windows::Forms::Label());

this->label5 = (gcnew System::Windows::Forms::Label());

this->button2 = (gcnew System::Windows::Forms::Button());

this->textBox4 = (gcnew System::Windows::Forms::TextBox());

this->label4 = (gcnew System::Windows::Forms::Label());

this->textBox3 = (gcnew System::Windows::Forms::TextBox());

this->label3 = (gcnew System::Windows::Forms::Label());

this->panel1->SuspendLayout();

this->panel2->SuspendLayout();

this->SuspendLayout();

//

// label1

//

this->label1->AutoSize = true;

this->label1->Font = (gcnew System::Drawing::Font(L"Times New Roman", 16.2F, System::Drawing::FontStyle::Regular, System::Drawing::GraphicsUnit::Point,

static\_cast<System::Byte>(204)));

this->label1->Location = System::Drawing::Point(31, 77);

this->label1->Name = L"label1";

this->label1->Size = System::Drawing::Size(275, 33);

this->label1->TabIndex = 0;

this->label1->Text = L"Введіть ваш pin-код - ";

//

// textBox1

//

this->textBox1->BackColor = System::Drawing::SystemColors::Control;

this->textBox1->Font = (gcnew System::Drawing::Font(L"Times New Roman", 16.2F));

this->textBox1->Location = System::Drawing::Point(312, 71);

this->textBox1->Name = L"textBox1";

this->textBox1->Size = System::Drawing::Size(184, 39);

this->textBox1->TabIndex = 1;

//

// label2

//

this->label2->AutoSize = true;

this->label2->Font = (gcnew System::Drawing::Font(L"Times New Roman", 16.2F, System::Drawing::FontStyle::Regular, System::Drawing::GraphicsUnit::Point,

static\_cast<System::Byte>(204)));

this->label2->Location = System::Drawing::Point(31, 188);

this->label2->Name = L"label2";

this->label2->Size = System::Drawing::Size(183, 33);

this->label2->TabIndex = 2;

this->label2->Text = L"Ваш pin-код - ";

//

// textBox2

//

this->textBox2->BackColor = System::Drawing::SystemColors::Control;

this->textBox2->BorderStyle = System::Windows::Forms::BorderStyle::None;

this->textBox2->Font = (gcnew System::Drawing::Font(L"Times New Roman", 16.2F));

this->textBox2->Location = System::Drawing::Point(312, 182);

this->textBox2->Name = L"textBox2";

this->textBox2->Size = System::Drawing::Size(184, 32);

this->textBox2->TabIndex = 3;

//

// button1

//

this->button1->Font = (gcnew System::Drawing::Font(L"Times New Roman", 16.2F, System::Drawing::FontStyle::Regular, System::Drawing::GraphicsUnit::Point,

static\_cast<System::Byte>(204)));

this->button1->Location = System::Drawing::Point(517, 396);

this->button1->Name = L"button1";

this->button1->Size = System::Drawing::Size(115, 52);

this->button1->TabIndex = 4;

this->button1->Text = L"Далі";

this->button1->UseVisualStyleBackColor = true;

this->button1->Click += gcnew System::EventHandler(this, &MyForm5::Button1\_Click);

//

// panel1

//

this->panel1->Controls->Add(this->panel2);

this->panel1->Controls->Add(this->label5);

this->panel1->Controls->Add(this->button2);

this->panel1->Controls->Add(this->textBox4);

this->panel1->Controls->Add(this->label4);

this->panel1->Controls->Add(this->textBox3);

this->panel1->Controls->Add(this->label3);

this->panel1->Dock = System::Windows::Forms::DockStyle::Fill;

this->panel1->Location = System::Drawing::Point(0, 0);

this->panel1->Name = L"panel1";

this->panel1->Size = System::Drawing::Size(644, 460);

this->panel1->TabIndex = 5;

this->panel1->Visible = false;

//

// panel2

//

this->panel2->Controls->Add(this->button3);

this->panel2->Controls->Add(this->textBox5);

this->panel2->Controls->Add(this->label6);

this->panel2->Dock = System::Windows::Forms::DockStyle::Fill;

this->panel2->Location = System::Drawing::Point(0, 0);

this->panel2->Name = L"panel2";

this->panel2->Size = System::Drawing::Size(644, 460);

this->panel2->TabIndex = 6;

this->panel2->Visible = false;

//

// button3

//

this->button3->Font = (gcnew System::Drawing::Font(L"Times New Roman", 16.2F, System::Drawing::FontStyle::Regular, System::Drawing::GraphicsUnit::Point,

static\_cast<System::Byte>(204)));

this->button3->Location = System::Drawing::Point(194, 351);

this->button3->Name = L"button3";

this->button3->Size = System::Drawing::Size(259, 77);

this->button3->TabIndex = 6;

this->button3->Text = L"Перейти до інших операцій";

this->button3->UseVisualStyleBackColor = true;

this->button3->Click += gcnew System::EventHandler(this, &MyForm5::Button3\_Click\_1);

//

// textBox5

//

this->textBox5->BackColor = System::Drawing::SystemColors::Control;

this->textBox5->BorderStyle = System::Windows::Forms::BorderStyle::None;

this->textBox5->Font = (gcnew System::Drawing::Font(L"Times New Roman", 16.2F));

this->textBox5->Location = System::Drawing::Point(296, 99);

this->textBox5->Name = L"textBox5";

this->textBox5->Size = System::Drawing::Size(157, 32);

this->textBox5->TabIndex = 5;

//

// label6

//

this->label6->AutoSize = true;

this->label6->Font = (gcnew System::Drawing::Font(L"Times New Roman", 16.2F, System::Drawing::FontStyle::Regular, System::Drawing::GraphicsUnit::Point,

static\_cast<System::Byte>(204)));

this->label6->Location = System::Drawing::Point(70, 98);

this->label6->Name = L"label6";

this->label6->Size = System::Drawing::Size(154, 33);

this->label6->TabIndex = 4;

this->label6->Text = L"Ваш баланс";

//

// label5

//

this->label5->AutoSize = true;

this->label5->Font = (gcnew System::Drawing::Font(L"Times New Roman", 19.8F, System::Drawing::FontStyle::Bold, System::Drawing::GraphicsUnit::Point,

static\_cast<System::Byte>(204)));

this->label5->Location = System::Drawing::Point(118, 289);

this->label5->Name = L"label5";

this->label5->Size = System::Drawing::Size(406, 37);

this->label5->TabIndex = 5;

this->label5->Text = L"У вас недостатньо коштів";

this->label5->Visible = false;

//

// button2

//

this->button2->Font = (gcnew System::Drawing::Font(L"Times New Roman", 16.2F));

this->button2->Location = System::Drawing::Point(530, 397);

this->button2->Name = L"button2";

this->button2->Size = System::Drawing::Size(102, 51);

this->button2->TabIndex = 4;

this->button2->Text = L"Далі";

this->button2->UseVisualStyleBackColor = true;

this->button2->Click += gcnew System::EventHandler(this, &MyForm5::Button2\_Click);

//

// textBox4

//

this->textBox4->BackColor = System::Drawing::SystemColors::Control;

this->textBox4->BorderStyle = System::Windows::Forms::BorderStyle::FixedSingle;

this->textBox4->Font = (gcnew System::Drawing::Font(L"Times New Roman", 16.2F));

this->textBox4->Location = System::Drawing::Point(296, 137);

this->textBox4->Name = L"textBox4";

this->textBox4->Size = System::Drawing::Size(157, 39);

this->textBox4->TabIndex = 3;

//

// label4

//

this->label4->AutoSize = true;

this->label4->Font = (gcnew System::Drawing::Font(L"Times New Roman", 16.2F, System::Drawing::FontStyle::Regular, System::Drawing::GraphicsUnit::Point,

static\_cast<System::Byte>(204)));

this->label4->Location = System::Drawing::Point(37, 122);

this->label4->Name = L"label4";

this->label4->Size = System::Drawing::Size(216, 66);

this->label4->TabIndex = 2;

this->label4->Text = L"Введіть суму яку\r\nви хочете зняти";

//

// textBox3

//

this->textBox3->BackColor = System::Drawing::SystemColors::Control;

this->textBox3->BorderStyle = System::Windows::Forms::BorderStyle::None;

this->textBox3->Font = (gcnew System::Drawing::Font(L"Times New Roman", 16.2F));

this->textBox3->Location = System::Drawing::Point(296, 45);

this->textBox3->Name = L"textBox3";

this->textBox3->Size = System::Drawing::Size(157, 32);

this->textBox3->TabIndex = 1;

//

// label3

//

this->label3->AutoSize = true;

this->label3->Font = (gcnew System::Drawing::Font(L"Times New Roman", 16.2F, System::Drawing::FontStyle::Regular, System::Drawing::GraphicsUnit::Point,

static\_cast<System::Byte>(204)));

this->label3->Location = System::Drawing::Point(37, 44);

this->label3->Name = L"label3";

this->label3->Size = System::Drawing::Size(177, 33);

this->label3->TabIndex = 0;

this->label3->Text = L"Ваш баланс - ";

//

// MyForm5

//

this->AutoScaleDimensions = System::Drawing::SizeF(8, 16);

this->AutoScaleMode = System::Windows::Forms::AutoScaleMode::Font;

this->ClientSize = System::Drawing::Size(644, 460);

this->Controls->Add(this->panel1);

this->Controls->Add(this->button1);

this->Controls->Add(this->textBox2);

this->Controls->Add(this->label2);

this->Controls->Add(this->textBox1);

this->Controls->Add(this->label1);

this->Name = L"MyForm5";

this->StartPosition = System::Windows::Forms::FormStartPosition::CenterScreen;

this->Text = L"MyForm5";

this->Load += gcnew System::EventHandler(this, &MyForm5::MyForm5\_Load);

this->panel1->ResumeLayout(false);

this->panel1->PerformLayout();

this->panel2->ResumeLayout(false);

this->panel2->PerformLayout();

this->ResumeLayout(false);

this->PerformLayout();

}

#pragma endregion

private: System::Void MyForm5\_Load(System::Object^ sender, System::EventArgs^ e) {

Client c;

c.start();

textBox2->Text = System::Convert::ToString(c.getPinClient());

}

private: System::Void Button1\_Click(System::Object^ sender, System::EventArgs^ e) {

panel1->Visible = true;

Client c;

c.start();

textBox3->Text = System::Convert::ToString(c.getMoneyClient());

}

private: System::Void Button2\_Click(System::Object^ sender, System::EventArgs^ e) {

double money;

bool check;

Client c;

c.start();

check = c.getMoneyClient();

money = System::Convert::ToDouble(textBox4->Text);

check = c.atmCheck(money);

if (check==false)

{

label5->Visible=true;

}

if(check==true)

{

panel2->Visible = true;

c.atmClient(money);

textBox5->Text = System::Convert::ToString(c.getMoneyClient());

}

textBox4->Text = System::Convert::ToString(0.0);

}

private: System::Void Button3\_Click(System::Object^ sender, System::EventArgs^ e) {

panel2->Visible = false;

}

private: System::Void Button4\_Click(System::Object^ sender, System::EventArgs^ e) {

this->Close();

}

private: System::Void Panel2\_Paint(System::Object^ sender, System::Windows::Forms::PaintEventArgs^ e) {

}

private: System::Void Button3\_Click\_1(System::Object^ sender, System::EventArgs^ e) {

this->Close();

}

};

}

MyForm6.h

#pragma once

#include"Client.h"

namespace BankCursova2 {

using namespace System;

using namespace System::ComponentModel;

using namespace System::Collections;

using namespace System::Windows::Forms;

using namespace System::Data;

using namespace System::Drawing;

/// <summary>

/// Сводка для MyForm6

/// </summary>

public ref class MyForm6 : public System::Windows::Forms::Form

{

public:

MyForm6(void)

{

InitializeComponent();

//

//TODO: добавьте код конструктора

//

}

protected:

/// <summary>

/// Освободить все используемые ресурсы.

/// </summary>

~MyForm6()

{

if (components)

{

delete components;

}

}

private: System::Windows::Forms::TextBox^ textBox1;

protected:

private: System::Windows::Forms::Label^ label2;

private: System::Windows::Forms::Label^ label4;

private: System::Windows::Forms::RadioButton^ radioButton1;

private: System::Windows::Forms::Label^ label3;

private: System::Windows::Forms::RadioButton^ radioButton2;

private: System::Windows::Forms::RadioButton^ radioButton3;

private: System::Windows::Forms::Button^ button2;

private: System::Windows::Forms::Label^ label1;

private: System::Windows::Forms::Panel^ panel1;

private: System::Windows::Forms::TextBox^ textBox2;

private: System::Windows::Forms::Label^ label5;

private: System::Windows::Forms::Button^ button1;

protected:

private:

/// <summary>

/// Обязательная переменная конструктора.

/// </summary>

System::ComponentModel::Container ^components;

#pragma region Windows Form Designer generated code

/// <summary>

/// Требуемый метод для поддержки конструктора — не изменяйте

/// содержимое этого метода с помощью редактора кода.

/// </summary>

void InitializeComponent(void)

{

this->textBox1 = (gcnew System::Windows::Forms::TextBox());

this->label2 = (gcnew System::Windows::Forms::Label());

this->label4 = (gcnew System::Windows::Forms::Label());

this->radioButton1 = (gcnew System::Windows::Forms::RadioButton());

this->label3 = (gcnew System::Windows::Forms::Label());

this->radioButton2 = (gcnew System::Windows::Forms::RadioButton());

this->radioButton3 = (gcnew System::Windows::Forms::RadioButton());

this->button2 = (gcnew System::Windows::Forms::Button());

this->label1 = (gcnew System::Windows::Forms::Label());

this->panel1 = (gcnew System::Windows::Forms::Panel());

this->textBox2 = (gcnew System::Windows::Forms::TextBox());

this->label5 = (gcnew System::Windows::Forms::Label());

this->button1 = (gcnew System::Windows::Forms::Button());

this->panel1->SuspendLayout();

this->SuspendLayout();

//

// textBox1

//

this->textBox1->BackColor = System::Drawing::SystemColors::Control;

this->textBox1->BorderStyle = System::Windows::Forms::BorderStyle::FixedSingle;

this->textBox1->Font = (gcnew System::Drawing::Font(L"Times New Roman", 16.2F, System::Drawing::FontStyle::Regular, System::Drawing::GraphicsUnit::Point,

static\_cast<System::Byte>(204)));

this->textBox1->ForeColor = System::Drawing::SystemColors::WindowText;

this->textBox1->Location = System::Drawing::Point(247, 82);

this->textBox1->Name = L"textBox1";

this->textBox1->Size = System::Drawing::Size(223, 39);

this->textBox1->TabIndex = 18;

//

// label2

//

this->label2->AutoSize = true;

this->label2->Font = (gcnew System::Drawing::Font(L"Times New Roman", 16.2F, System::Drawing::FontStyle::Bold, System::Drawing::GraphicsUnit::Point,

static\_cast<System::Byte>(204)));

this->label2->Location = System::Drawing::Point(22, 89);

this->label2->Name = L"label2";

this->label2->Size = System::Drawing::Size(179, 32);

this->label2->TabIndex = 17;

this->label2->Text = L"Введіть суму";

//

// label4

//

this->label4->AutoSize = true;

this->label4->Font = (gcnew System::Drawing::Font(L"Times New Roman", 16.2F, System::Drawing::FontStyle::Bold, System::Drawing::GraphicsUnit::Point,

static\_cast<System::Byte>(204)));

this->label4->Location = System::Drawing::Point(195, 15);

this->label4->Name = L"label4";

this->label4->Size = System::Drawing::Size(251, 32);

this->label4->TabIndex = 15;

this->label4->Text = L"Заповніть всі поля";

//

// radioButton1

//

this->radioButton1->AutoSize = true;

this->radioButton1->Font = (gcnew System::Drawing::Font(L"Times New Roman", 13.8F, System::Drawing::FontStyle::Regular, System::Drawing::GraphicsUnit::Point,

static\_cast<System::Byte>(204)));

this->radioButton1->Location = System::Drawing::Point(28, 233);

this->radioButton1->Name = L"radioButton1";

this->radioButton1->Size = System::Drawing::Size(81, 58);

this->radioButton1->TabIndex = 7;

this->radioButton1->Text = L"1 рік\r\n 3%\r\n";

this->radioButton1->UseVisualStyleBackColor = true;

//

// label3

//

this->label3->AutoSize = true;

this->label3->Font = (gcnew System::Drawing::Font(L"Times New Roman", 16.2F, System::Drawing::FontStyle::Bold, System::Drawing::GraphicsUnit::Point,

static\_cast<System::Byte>(204)));

this->label3->Location = System::Drawing::Point(195, 182);

this->label3->Name = L"label3";

this->label3->Size = System::Drawing::Size(222, 32);

this->label3->TabIndex = 8;

this->label3->Tag = L"";

this->label3->Text = L"Виберіть термін";

//

// radioButton2

//

this->radioButton2->AutoSize = true;

this->radioButton2->Font = (gcnew System::Drawing::Font(L"Times New Roman", 13.8F, System::Drawing::FontStyle::Regular, System::Drawing::GraphicsUnit::Point,

static\_cast<System::Byte>(204)));

this->radioButton2->Location = System::Drawing::Point(263, 233);

this->radioButton2->Name = L"radioButton2";

this->radioButton2->Size = System::Drawing::Size(100, 58);

this->radioButton2->TabIndex = 9;

this->radioButton2->Text = L"3 роки\r\n 8%\r\n";

this->radioButton2->UseVisualStyleBackColor = true;

//

// radioButton3

//

this->radioButton3->AutoSize = true;

this->radioButton3->Font = (gcnew System::Drawing::Font(L"Times New Roman", 13.8F, System::Drawing::FontStyle::Regular, System::Drawing::GraphicsUnit::Point,

static\_cast<System::Byte>(204)));

this->radioButton3->Location = System::Drawing::Point(510, 233);

this->radioButton3->Name = L"radioButton3";

this->radioButton3->Size = System::Drawing::Size(104, 58);

this->radioButton3->TabIndex = 10;

this->radioButton3->Text = L"6 років\r\n 12%";

this->radioButton3->UseVisualStyleBackColor = true;

//

// button2

//

this->button2->Font = (gcnew System::Drawing::Font(L"Times New Roman", 16.2F));

this->button2->Location = System::Drawing::Point(247, 397);

this->button2->Name = L"button2";

this->button2->Size = System::Drawing::Size(170, 51);

this->button2->TabIndex = 19;

this->button2->Text = L"Далі";

this->button2->UseVisualStyleBackColor = true;

this->button2->Click += gcnew System::EventHandler(this, &MyForm6::Button2\_Click);

//

// label1

//

this->label1->AutoSize = true;

this->label1->Font = (gcnew System::Drawing::Font(L"Times New Roman", 16.2F, System::Drawing::FontStyle::Bold, System::Drawing::GraphicsUnit::Point,

static\_cast<System::Byte>(204)));

this->label1->Location = System::Drawing::Point(95, 323);

this->label1->Name = L"label1";

this->label1->Size = System::Drawing::Size(454, 32);

this->label1->TabIndex = 20;

this->label1->Tag = L"";

this->label1->Text = L"Вам відмовлено в виданні кредиту";

this->label1->Visible = false;

//

// panel1

//

this->panel1->Controls->Add(this->button1);

this->panel1->Controls->Add(this->textBox2);

this->panel1->Controls->Add(this->label5);

this->panel1->Dock = System::Windows::Forms::DockStyle::Fill;

this->panel1->Location = System::Drawing::Point(0, 0);

this->panel1->Name = L"panel1";

this->panel1->Size = System::Drawing::Size(644, 460);

this->panel1->TabIndex = 21;

this->panel1->Visible = false;

//

// textBox2

//

this->textBox2->BackColor = System::Drawing::SystemColors::Control;

this->textBox2->BorderStyle = System::Windows::Forms::BorderStyle::None;

this->textBox2->Font = (gcnew System::Drawing::Font(L"Times New Roman", 16.2F));

this->textBox2->Location = System::Drawing::Point(308, 72);

this->textBox2->Name = L"textBox2";

this->textBox2->Size = System::Drawing::Size(162, 32);

this->textBox2->TabIndex = 4;

//

// label5

//

this->label5->AutoSize = true;

this->label5->Font = (gcnew System::Drawing::Font(L"Times New Roman", 16.2F, System::Drawing::FontStyle::Regular, System::Drawing::GraphicsUnit::Point,

static\_cast<System::Byte>(204)));

this->label5->Location = System::Drawing::Point(95, 72);

this->label5->Name = L"label5";

this->label5->Size = System::Drawing::Size(179, 33);

this->label5->TabIndex = 3;

this->label5->Text = L"Ваш кредит - ";

//

// button1

//

this->button1->Font = (gcnew System::Drawing::Font(L"Times New Roman", 16.2F));

this->button1->Location = System::Drawing::Point(228, 358);

this->button1->Name = L"button1";

this->button1->Size = System::Drawing::Size(170, 51);

this->button1->TabIndex = 5;

this->button1->Text = L"Завершити";

this->button1->UseVisualStyleBackColor = true;

this->button1->Click += gcnew System::EventHandler(this, &MyForm6::Button1\_Click);

//

// MyForm6

//

this->AutoScaleDimensions = System::Drawing::SizeF(8, 16);

this->AutoScaleMode = System::Windows::Forms::AutoScaleMode::Font;

this->ClientSize = System::Drawing::Size(644, 460);

this->Controls->Add(this->panel1);

this->Controls->Add(this->label1);

this->Controls->Add(this->button2);

this->Controls->Add(this->textBox1);

this->Controls->Add(this->label2);

this->Controls->Add(this->label4);

this->Controls->Add(this->radioButton3);

this->Controls->Add(this->radioButton2);

this->Controls->Add(this->label3);

this->Controls->Add(this->radioButton1);

this->Name = L"MyForm6";

this->Text = L"MyForm6";

this->panel1->ResumeLayout(false);

this->panel1->PerformLayout();

this->ResumeLayout(false);

this->PerformLayout();

}

#pragma endregion

private: System::Void Button2\_Click(System::Object^ sender, System::EventArgs^ e) {

double money;

int time;

bool check;

double cre;

Client c;

c.start();

check = c.getMoneyClient();

money = System::Convert::ToDouble(textBox1->Text);

if (radioButton1->Checked == true)

{

time = 12;

}

if (radioButton2->Checked == true)

{

time = 36;

}

if (radioButton3->Checked == true)

{

time = 72;

}

c.setAc();

if (c.getAc() == false)

{

label1->Visible = true;

c.setAc();

}

if (c.getAc() == true)

{

label1->Visible = false;

cre = c.operC(time, money);

panel1->Visible = true;

textBox2->Text = System::Convert::ToString(cre);

//panel2->Visible = true;

//c.atmClient(money);

//textBox5->Text = System::Convert::ToString(c.getMoneyClient());

}

}

private: System::Void Button1\_Click(System::Object^ sender, System::EventArgs^ e) {

this->Close();

}

private: System::Void Button3\_Click(System::Object^ sender, System::EventArgs^ e) {

}

};

}

Choise.h

#pragma once

#include"Deposit.h"

#include"MyForm6.h"

namespace BankCursova2 {

using namespace System;

using namespace System::ComponentModel;

using namespace System::Collections;

using namespace System::Windows::Forms;

using namespace System::Data;

using namespace System::Drawing;

/// <summary>

/// Сводка для Choise

/// </summary>

public ref class Choise : public System::Windows::Forms::Form

{

public:

Choise(void)

{

InitializeComponent();

//

//TODO: добавьте код конструктора

//

}

protected:

/// <summary>

/// Освободить все используемые ресурсы.

/// </summary>

~Choise()

{

if (components)

{

delete components;

}

}

private: System::Windows::Forms::Button^ button1;

private: System::Windows::Forms::Button^ button2;

private: System::Windows::Forms::Button^ button3;

protected:

private:

/// <summary>

/// Обязательная переменная конструктора.

/// </summary>

System::ComponentModel::Container ^components;

#pragma region Windows Form Designer generated code

/// <summary>

/// Требуемый метод для поддержки конструктора — не изменяйте

/// содержимое этого метода с помощью редактора кода.

/// </summary>

void InitializeComponent(void)

{

this->button1 = (gcnew System::Windows::Forms::Button());

this->button2 = (gcnew System::Windows::Forms::Button());

this->button3 = (gcnew System::Windows::Forms::Button());

this->SuspendLayout();

//

// button1

//

this->button1->Font = (gcnew System::Drawing::Font(L"Times New Roman", 19.8F, System::Drawing::FontStyle::Bold, System::Drawing::GraphicsUnit::Point,

static\_cast<System::Byte>(204)));

this->button1->Location = System::Drawing::Point(61, 164);

this->button1->Name = L"button1";

this->button1->Size = System::Drawing::Size(183, 66);

this->button1->TabIndex = 0;

this->button1->Text = L"Депозити";

this->button1->UseVisualStyleBackColor = true;

this->button1->Click += gcnew System::EventHandler(this, &Choise::Button1\_Click);

//

// button2

//

this->button2->Font = (gcnew System::Drawing::Font(L"Times New Roman", 19.8F, System::Drawing::FontStyle::Bold, System::Drawing::GraphicsUnit::Point,

static\_cast<System::Byte>(204)));

this->button2->Location = System::Drawing::Point(395, 164);

this->button2->Name = L"button2";

this->button2->Size = System::Drawing::Size(183, 66);

this->button2->TabIndex = 1;

this->button2->Text = L"Кредити";

this->button2->UseVisualStyleBackColor = true;

this->button2->Click += gcnew System::EventHandler(this, &Choise::Button2\_Click);

//

// button3

//

this->button3->Font = (gcnew System::Drawing::Font(L"Times New Roman", 19.8F, System::Drawing::FontStyle::Bold, System::Drawing::GraphicsUnit::Point,

static\_cast<System::Byte>(204)));

this->button3->Location = System::Drawing::Point(224, 286);

this->button3->Name = L"button3";

this->button3->Size = System::Drawing::Size(183, 66);

this->button3->TabIndex = 2;

this->button3->Text = L"Назад";

this->button3->UseVisualStyleBackColor = true;

this->button3->Click += gcnew System::EventHandler(this, &Choise::Button3\_Click);

//

// Choise

//

this->AutoScaleDimensions = System::Drawing::SizeF(8, 16);

this->AutoScaleMode = System::Windows::Forms::AutoScaleMode::Font;

this->ClientSize = System::Drawing::Size(644, 460);

this->Controls->Add(this->button3);

this->Controls->Add(this->button2);

this->Controls->Add(this->button1);

this->Name = L"Choise";

this->StartPosition = System::Windows::Forms::FormStartPosition::CenterScreen;

this->Text = L"Choise";

this->ResumeLayout(false);

}

#pragma endregion

private: System::Void Button1\_Click(System::Object^ sender, System::EventArgs^ e) {

this->Hide();

Deposit^ d = gcnew Deposit;

d->ShowDialog();

}

private: System::Void Button2\_Click(System::Object^ sender, System::EventArgs^ e) {

this->Hide();

MyForm6^ d = gcnew MyForm6;

d->ShowDialog();

}

private: System::Void Button3\_Click(System::Object^ sender, System::EventArgs^ e) {

this->Close();

}

};

}