МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ

ТЕХНІЧНИЙ КОЛЕДЖ ТНТУ ІМЕНІ ІВАНА ПУЛЮЯ

ВІДДІЛЕННЯ ЕЛЕКТРОННИХ АПАРАТІВ

Циклова комісія програмних систем і комплексів

**ЗВІТ**

про виконання лабораторних робітз дисципліни:

**«ОБ’ЄКТНО-ОРІЄНТОВАНЕ ПРОГРАМУВАННЯ»**

Студента  3  курсу групи КН-321 спеціальності 122 «Комп’ютерні науки»

Слюсарчин О.В

(прізвище та ініціали)

Перевірив: Р.О. Слободян

(підпис)

Тернопіль – 2020

**Лабораторна робота №4**

**Тема:**  Успадкування класів

**Мета:** Ознайомитись зі способами та механізмами успадкування класів та навчитись використовувати їх для побудови об’єктно-орієнтованих програм.

**Завдання 1** Уявіть собі видавничу компанію, яка торгує книгами і аудіо-записами цих книг. Створіть клас publication, в якому зберігаються назва (рядок) і ціна (тип float) книги. Від цього класу успадковуються ще два класи: book, який містить інформацію про кількість сторінок у книзі (типу int), і type, який містить час запису книги у хвилинах (тип float). У кожному з цих трьох класів повинен бути метод getdata(), через який можна отримувати дані від користувача з клавіатури, і putdata(), призначений для виведення цих даних. Напишіть функцію main() програми для перевірки класів book і type. Створіть їх об'єкти в програмі і запросіть користувача ввести і вивести дані з використанням методів getdataQ і putdata().

**КОД ПРОГРАМИ**

#include <iostream>

#include <string>

using namespace std;

class **Publication** {

string name;

float price;

public:

**Publication**() {}

int **GetData**()

{

cout << "enter name of the book:" << endl;

cin >> name;

cout << "enter price of the book:" << endl;

cin >> price;

}

void **PutData**()

{

cout << "name of the book" << endl << name << endl;

cout << "price of the book" <<endl << price << endl;

}

};

class **Book**: Publication

{

int QuantityOfPages;

public:

int **GetData**()

{

Publication::GetData();

cout << "Enter quantity of pages:" << endl;

cin >> QuantityOfPages;

}

void **PutData**()

{

Publication::PutData();

cout << "Quantity of pages is:" << QuantityOfPages << endl;

}

};

class **Type**: Publication

{

public:

float TimeOfWritingBook;

int **GetData**()

{

Publication::GetData();

cout << "enter time of writing book" << endl;

cin >> TimeOfWritingBook;

}

void **PutData**()

{

Publication::PutData();

cout << "Time of writing book is:" << endl;

}

};

int **main**()

{

setlocale (0,"ukr");

Book NumberOne;

Type NumberTwo;

NumberOne.GetData();

NumberOne.PutData();

NumberTwo.GetData();

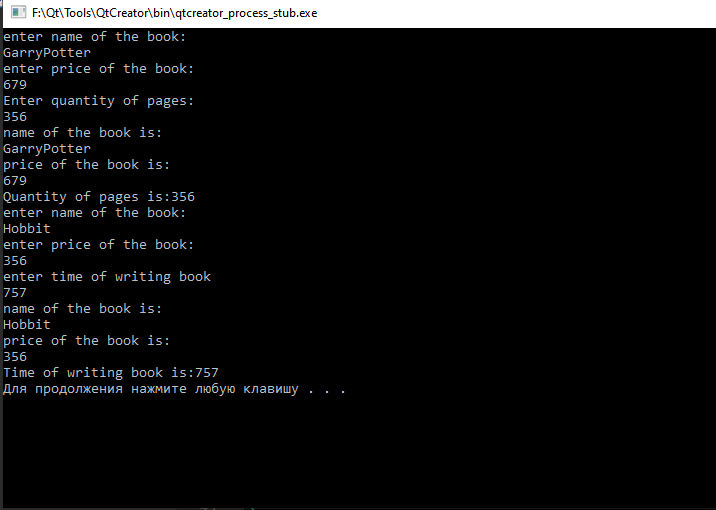
NumberTwo.PutData();

system ("pause");

return 0;

}

**РЕЗУЛЬТАТ ВИКОНАННЯ ПРОГРАМИ**

****

**Завдання 2** До класів з попереднього завдання (попередньо зберігши окремо код) додайте базовий клас sales, в якому міститься масив, що складається з трьох значень типу float, куди можна записати загальну вартість проданих книг за останні три місяці. Включіть в клас методи getdata() для отримання значень вартості від користувача і putdata() для виведення цих цифр. Змініть класи book і type так, щоб вони стали похідними обох класів: publication і sales. Об'єкти класів book і type повинні вводити і виводити дані про продажі разом з іншими своїми даними. Напишіть функцію main() для створення об'єктів класів book і type, щоб протестувати можливості введення/виведення даних.

**КОД ПРОГРАМИ**

#include <iostream>

#include <string>

using namespace std;

class **Publication**

{

string name;

float price;

public:

**Publication**() {}

int **GetData**()

{

cout << "enter name of the book:" << endl;

cin >> name;

cout << "enter price of the book:" << endl;

cin >> price;

}

void **PutData**()

{

cout << "name of the book" << endl << name << endl;

cout << "price of the book" <<endl << price << endl;

}

};

class **Sales**

{

float array [3];

float price2;

public:

int **GetData**()

{

cout << "enter price from user" << endl;

cin >> price2;

}

void **PutData**()

{

cout << "price from user is:" << endl << price2 << endl;

}

};

class **Book**: Publication,Sales

{

int QuantityOfPages;

public:

int **GetData**()

{

Publication::GetData();

Sales::GetData();

cout << "Enter quantity of pages:" << endl;

cin >> QuantityOfPages;

}

void **PutData**()

{

Publication::PutData();

Sales::PutData();

cout << "Quantity of pages is:" << QuantityOfPages << endl;

}

};

class **Type**: Publication,Sales

{

public:

float TimeOfWritingBook;

int **GetData**()

{

Publication::GetData();

Sales::GetData();

cout << "enter time of writing book" << endl;

cin >> TimeOfWritingBook;

}

void **PutData**()

{

Publication::PutData();

Sales::PutData();

cout << "Time of writing book is:" << TimeOfWritingBook << endl;

}

};

int **main**()

{

setlocale (0,"ukr");

Book NumberOne;

Type NumberTwo;

Sales NumberThree;

NumberOne.GetData();

NumberOne.PutData();

NumberTwo.GetData();

NumberTwo.PutData();

NumberThree.GetData();

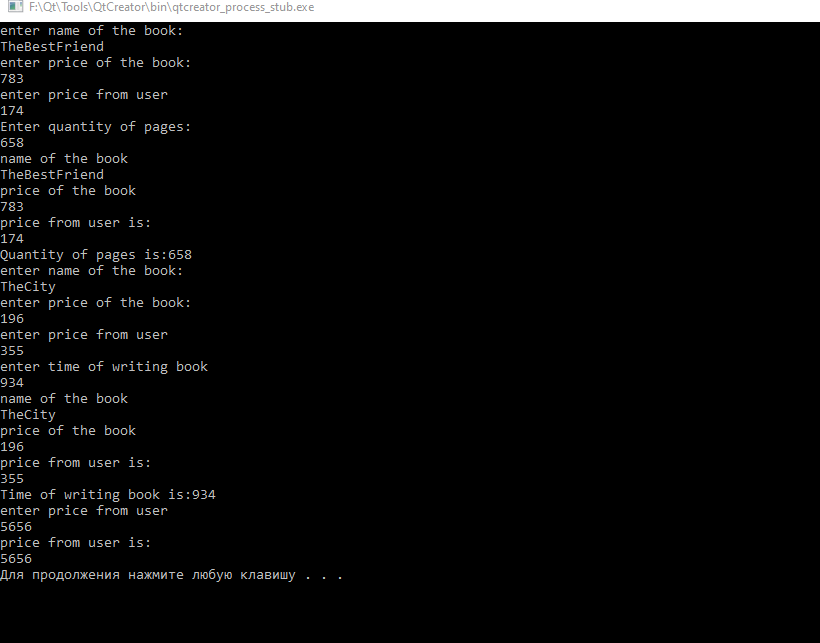
NumberThree.PutData();

system ("pause");

return 0;

}

**РЕЗУЛЬТАТ ВИКОНАННЯ ПРОГРАМИ**



**Завдання 3 (варіант 1)** Створити клас ПАРА ЦІЛИХ ЧИСЕЛ. Визначити конструктори, деструктор, функції доступу до полів, введення-виведення та порівняння пар (пара p1 більша за пару p2, якщо перше число p1 більше за перше число p2 або перші числа рівні і друге число p1 більше за друге число p2). Створити похідний клас ДРОБОВЕ ЧИСЛО з полями: ціла частина, дробова частина. Визначити конструктори за замовчуванням і з різним числом параметрів, деструктор, функції доступу до полів, введення-виведення та порівняння дробових чисел (на рівність/нерівність, більшість/меншість).

**КОД ПРОГРАМИ**

#include <iostream>

#include <string>

using namespace std;

class **PairOfIntegers**

{

int p1;

int p2;

public:

**PairOfIntegers**() {}

int **GetP**()

{

cout << "enter value of p1:" << endl;

cin >> p1;

cout << "enter value of p2:" << endl;

cin >> p2;

if(p1 > p2)

{

cout << "p1 is bigger than p2" << endl;

}

else if (p1 < p2)

{

cout << "p2 is bigger than p1" << endl;

}

else cout << "p1 = p2" << endl;

}

void **SetP**()

{

cout << "p1=" << p1 << endl;

cout << "p2=" << p2 << endl;

}

};

class **Float**: PairOfIntegers

{

int value1;

float value2;

public:

**Float**() {}

int **GetValue**()

{

cout << "enter int value:" << endl;

cin >> value1;

cout << "enter float value:" << endl;

cin >> value2;

if(value1 > value2)

{

cout << "value1 is bigger than value2" << endl;

}

else if (value1 < value2)

{

cout << "value2 is bigger than value1" << endl;

}

else cout << "value1 = value2" << endl;

}

void **SetValue**()

{

cout << "value1=" << value1 << endl;

cout << "value2=" << value2 << endl;

}

};

int **main**()

{

setlocale(0,"ukr");

PairOfIntegers NumberOne,NumbeTwo;

Float NumberThree,NumberFour;

NumberOne.GetP();

NumberOne.SetP();

NumberThree.GetValue();

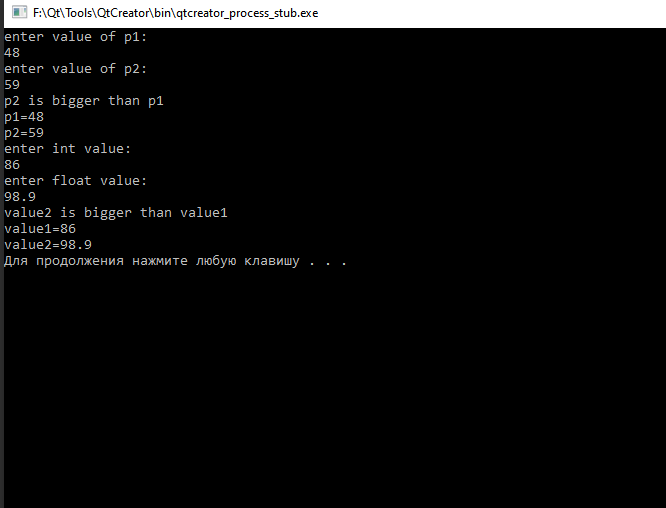
NumberThree.SetValue();

system("pause");

return 0;

}

**РЕЗУЛЬТАТ ВИКОНАННЯ ПРОГРАМИ**



Висновок: Ознайомився зі способами та механізмами успадкування класів та навчитись використовувати їх для побудови об’єктно-орієнтованих програм.