Кам’янець-Подільський національний університет

 імені Івана Огієнка

КАФЕДРА КОМП’ЮТЕРНИХ НАУК

Навчальна дисципліна «Об’єктно-орієнтоване програмування»

ЛАБОРАТОРНА РОБОТА #0102

Тема: Статичні члени класу

Варіант №3

Виконав:

студент 1-го курсу

групи KNms1-B21

Волошин В.Ю.

Прийняла:

старший викладач,

Мястковська М.О.

Кам’янець-Подільський – 2023

1. **Хід роботи.**

Статичні члени класу в мові програмування C# є ключовим елементом об'єктно-орієнтованого програмування, принцип якого полягає в організації програмного коду через класи та об'єкти.

Основна ідея статичних членів полягає в тому, щоб вони належали самому класу, а не його екземплярам. Це означає, що вони доступні через ім'я класу, а не через конкретний об'єкт класу. Статичні члени можуть включати в себе статичні змінні, статичні методи та статичні властивості.

Однією з ключових переваг використання статичних членів є здатність до спільного використання даних між різними об'єктами класу. Наприклад, якщо у нас є статична змінна в класі, то всі об'єкти цього класу будуть використовувати одну й ту ж змінну. Це дозволяє ефективно обмінюватися даними без необхідності створення багатьох екземплярів класу.

Ще однією важливою перевагою є можливість використовувати статичні члени без створення екземпляра класу. Це особливо корисно в ситуаціях, коли функціональність класу є глобальною і не пов'язана з конкретним екземпляром.

З іншого боку, важливо розуміти обмеження використання статичних членів. Наприклад, статичні члени не можуть бути перевизначені в похідних класах, що може викликати обмеження при спробі створення ієрархії класів. Також важливо пам'ятати, що статичні члени не можуть отримувати прямого доступу до нестатичних членів класу, і вони можуть використовуватися тільки через екземпляри класу.

Отже, статичні члени класу в C# є потужним інструментом, який дозволяє ефективно використовувати об'єктно-орієнтовані можливості мови програмування, проте їхнє використання вимагає уважності та розуміння їхніх особливостей та обмежень.

1. **Завдання до виконання.**

**Завдання 1.**

Створити метод розширення для масиву цілих чисел, який шукає мінімум і максимум в масиві.

**Завдання 2.**

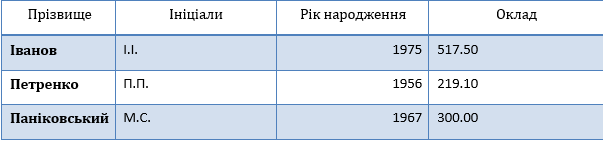
2.1. Створити клас відповідно до варіанту. Клас також повинен містить статичне поле, для підрахунку кількості об’єктів.

2.2. Створити масив об’єктів відповідного типу. Можна даний масив оформити у вигляді окремого класу.

2.3. Створити декілька об’єктів відповідного класу (не менше 10), і записати їх в масив.

2.4. Знайти об’єкти, які відповідають певному критерію. Критерій підібрати самостійно відповідно до варіанту. Наприклад, для першого варіанту: 1) види антилоп, чисельність яких менше заданого значення; 2) вид антилоп кількість яких є мінімальною.

2.5. Перевірити кількість існуючих об’єктів. Якщо це значення перевищує задане граничне значення1 – вивести повідомлення1, якщо значення менше заданого граничного значення2 – вивести повідомлення2.



1. **Лістинг програми.**

public static class ArrayExtensions

    {

        public static void FindMinMax(this int[] array, out int min, out int max)

        {

            if (array == null || array.Length == 0)

            {

                throw new ArgumentException("Array is empty or null.");

            }

            min = array[0];

            max = array[0];

            foreach (var num in array)

            {

                if (num < min)

                {

                    min = num;

                }

                if (num > max)

                {

                    max = num;

                }

            }

        }

    }

public class Employee

    {

        public string LastName { get; set; }

        public string Initials { get; set; }

        public int BirthYear { get; set; }

        public double Salary { get; set; }

        public static int ObjectCount { get; private set; }

        public Employee(string lastName, string initials, int birthYear, double salary)

        {

            LastName = lastName;

            Initials = initials;

            BirthYear = birthYear;

            Salary = salary;

            ObjectCount++;

        }

    }

class Program

{

    static void Main()

    {

        // task 1

        int[] numbers = { 4, 2, 7, 1, 9, 5 };

        int min, max;

        numbers.FindMinMax(out min, out max);

        Console.WriteLine($"Min: {min}, Max: {max}");

        // task 2

        Employee[] employees = new Employee[10];

        Random random = new Random();

        // Створюємо 10 об'єктів Employee та додаємо їх в масив

        for (int i = 0; i < 10; i++)

        {

            employees[i] = new Employee($"Employee{i + 1}", $"{(char)('А' + i)}.{(char)('А' + i)}", 1980 + i, random.Next(2000, 5000));

        }

        // Виводимо інформацію про всіх співробітників

        Console.WriteLine("Список співробітників:");

        foreach (var employee in employees)

        {

            Console.WriteLine($"{employee.LastName}, {employee.Initials}, {employee.BirthYear}, {employee.Salary:C}");

        }

        // Знаходимо співробітників, оклад яких менший за певне значення

        double salaryThreshold = 3000;

        Console.WriteLine($"\nСпівробітники з окладом менше {salaryThreshold:C}:");

        foreach (var employee in employees)

        {

            if (employee.Salary < salaryThreshold)

            {

                Console.WriteLine($"{employee.LastName}, {employee.Initials}, {employee.BirthYear}, {employee.Salary:C}");

            }

        }

        // Перевіряємо кількість існуючих об'єктів

        int objectLimit1 = 15;

        int objectLimit2 = 5;

        if (Employee.ObjectCount > objectLimit1)

        {

            Console.WriteLine($"\nКількість об'єктів перевищує {objectLimit1}. Деякі об'єкти можуть бути не ініціалізовані.");

        }

        else if (Employee.ObjectCount < objectLimit2)

        {

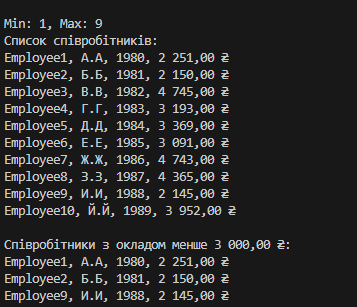
            Console.WriteLine($"\nКількість об'єктів менше {objectLimit2}. Збільште кількість створених об'єктів.");

        }

    }

}

1. **Результат роботи.**



1. **Посилання на проект.**