

Шакиров Р.Л. гр.4204 дата 05.12.2025
Лабораторная работа №. 2
Классы, член данные и член функции класса
C#

Задание 2: Объявить класс, содержащий два поля целого типа, свойства для работы с этими полями, а также статическую функцию. Функция принимает два параметра целого типа и осуществляет обмен значений этих параметров. Создать объект этого класса. Ввести с клавиатуры два числа и присвоить значения этих чисел полям объекта. Распечатать поля объекта. Осуществить обмен значений полей объекта и вновь распечатать поля объекта.

```
using System;

class Program
{
    // Объявляем класс Numbers
    class Numbers
    {
        // Приватные поля целого типа
        private int number1;
        private int number2;

        // Свойство для работы с полем number1
        public int Number1
        {
            get { return number1; }
            set { number1 = value; }
        }

        // Свойство для работы с полем number2
        public int Number2
        {
            get { return number2; }
            set { number2 = value; }
        }

        // Статическая функция для обмена значений двух
        // параметров
        // ВОЗВРАЩАЕМ кортеж вместо использования ref
        public static (int, int) Swap(int a, int b)
        {
            // Возвращаем значения в обратном порядке
            return (b, a);
        }

        // Альтернативный метод: обмен значений полей
        // самого объекта
        public void SwapFields()
        {
            int temp = number1;
            number1 = number2;
            number2 = temp;
        }
    }
}
```

```

        }

    }

    static void Main(string[] args)
    {
        // Создаем объект класса Numbers
        Numbers numbers = new Numbers();

        // Ввод первого числа с клавиатуры
        Console.Write("Введите первое число: ");
        numbers.Number1 = Convert.ToInt32(Console.ReadLine());

        // Ввод второго числа с клавиатуры
        Console.Write("Введите второе число: ");
        numbers.Number2 = Convert.ToInt32(Console.ReadLine());

        // Распечатываем поля объекта до обмена
        Console.WriteLine("\nДо обмена значений:");
        Console.WriteLine($"Number1 = {numbers.Number1}");
        Console.WriteLine($"Number2 = {numbers.Number2}");

        // СПОСОБ 1: Используем статический метод с
        // возвращаемым кортежем
        (numbers.Number1, numbers.Number2) = Numbers.Swap(numbers.Number1, numbers.Number2);

        // СПОСОБ 2: Используем метод объекта для обмена
        // его полей
        // numbers.SwapFields();

        // СПОСОБ 3: Прямой обмен через кортежи (самый
        // простой)
        // (numbers.Number1, numbers.Number2) = (numbers.Number2, numbers.Number1);

        // Распечатываем поля объекта после обмена
        Console.WriteLine("\nПосле обмена значений:");
        Console.WriteLine($"Number1 = {numbers.Number1}");
        Console.WriteLine($"Number2 = {numbers.Number2}");

        // Ждем нажатия клавиши перед закрытием консоли
        Console.WriteLine("\nНажмите любую клавишу для выхода...");
        Console.ReadKey();
    }
}

```

Задание 9: Объявить класс, содержащий одномерный целочисленный массив размерностью 10, индексатор для работы с этим массивом. Создать объект данного класса. Ввести с клавиатуры элементы массива объекта. Найти элемент массива наиболее близкий по своему значению среднему арифметическому значению элементов массива. Распечатать значение этого элемента и его индекс.

```

using System;

class IntArray

```

```
{  
    private int[] array; // Приватное поле для хранения массива  
  
    public IntArray()  
    {  
        array = new int[10]; // Инициализация массива размером 10  
    }  
  
    // Индексатор для доступа к элементам массива  
    public int this[int index]  
    {  
        get  
        {  
            // Проверка корректности индекса  
            if (index < 0 || index >= array.Length)  
                throw new IndexOutOfRangeException("Индекс выходит за  
границы массива");  
            return array[index];  
        }  
        set  
        {  
            if (index < 0 || index >= array.Length)  
                throw new IndexOutOfRangeException("Индекс выходит за  
границы массива");  
            array[index] = value;  
        }  
    }  
  
    // Свойство для получения длины массива  
    public int Length => array.Length;  
}  
  
class Program  
{  
    static void Main()  
    {  
        IntArray arr = new IntArray(); // Создание объекта класса IntArray  
  
        // Ввод элементов массива с клавиатуры  
        Console.WriteLine("Введите 10 целых чисел:");  
        for (int i = 0; i < arr.Length; i++)  
        {  
            Console.Write($"Элемент {i}: ");  
            arr[i] = int.Parse(Console.ReadLine());  
        }  
  
        // Вычисление среднего арифметического  
        double sum = 0;  
        for (int i = 0; i < arr.Length; i++)  
        {  
            sum += arr[i];  
        }  
        double average = sum / arr.Length;  
  
        // Поиск ближайшего элемента к среднему  
        // арифметическому
```

```
double minDifference = double.MaxValue; // Минимальная разница
int closestValue = arr[0]; // Ближайшее значение
int closestIndex = 0; // Индекс ближайшего значения

for (int i = 0; i < arr.Length; i++)
{
    // Вычисление абсолютной разницы между
    // текущим элементом и средним
    double difference = Math.Abs(arr[i] - average);

    // Если найдена меньшая разница, обновляем
    // результаты
    if (difference < minDifference)
    {
        minDifference = difference;
        closestValue = arr[i];
        closestIndex = i;
    }
}

// Вывод результата
Console.WriteLine($"Среднее арифметическое: {average:F2}");
Console.WriteLine($"Ближайший элемент: {closestValue}");
Console.WriteLine($"Индекс элемента: {closestIndex}");
}
```

Вывод: рассмотрены классы, члены и функции класса