

Шакиров Р.Л. гр 4204 дата 05.12.2025

Лабораторная работа №. 3

Конструкторы, поля только для чтения, вызов конструкторов

C#

2 задание: Создать приложение, в котором в цикле создается 10 объектов класса. Поля только для чтения каждого экземпляра равны порядковому номеру, отражающего очередность создания объектов

```
using System;
```

```
public class NumberObject {
    private static int _counter = 0;
    public readonly int Number;

    public NumberObject() {
        Number = ++_counter;
    }
}

class Program {
    static void Main() {
        for (int i = 0; i < 10; i++) {
            var obj = new NumberObject();
            Console.WriteLine($"Создан объект под номером: {obj.Number}");
        }
    }
}
```

Задание 8: Объявить класс, содержащий одномерный целочисленный массив, а также функцию, для печати элементов этого массива. Конструктор класса принимает один параметр - ссылку на массив (внешний массив). Конструктор выделяет память под внутренний массив размерностью совпадающей с размерностью внешнего массива, затем значения элементов внешнего массива копирует во внутренний массив и сортирует внутренний массив по возрастанию. Класс также должен содержать поля `readonly` для минимального и максимального элемента массива. Значения этих полей определяются в результате работы конструктора. Создать объект этого класса. Элементы внешнего массива ввести с клавиатуры. Распечатать элементы массива объекта, минимальное и максимальное значение массива объекта.

```
using System;
```

```
public class ArrayProcessor {
    private int[] _internalArray;
    public readonly int MinValue;
    public readonly int MaxValue;
```

```

//Конструктор принимающий ссылку на внешний массив
public ArrayProcessor(int[] externalArray) {
    if (externalArray == null || externalArray.Length == 0) {
        throw new ArgumentException("Массив не может быть null или
пустым");
    }

    //Выделяем память под внутренний массив
    _internalArray = new int[externalArray.Length];

    //Копируем значения из внешнего массива
    Array.Copy(externalArray, _internalArray, externalArray.Length);

    //Сортируем внутренний массив по возрастанию
    Array.Sort(_internalArray);

    //Определяем мин/макс значения
    MinValue = _internalArray[0];
    MaxValue = _internalArray[_internalArray.Length - 1];
}

//Функция для печати элементов массива
public void PrintArray() {
    Console.WriteLine("Элементы массива: ");
    foreach (var element in _internalArray) {
        Console.WriteLine(element + " ");
    }
    Console.WriteLine();
}

}

class Program {
    static void Main() {
        try {
            Console.WriteLine("Введите элементы массива через пробел:
");
            string input = Console.ReadLine();

            //Преобразуем введенную строку в массив целых
чисел
            string[] inputArray = input.Split(' ');
            int[] externalArray = new int[inputArray.Length];

            for (int i = 0; i < inputArray.Length; i++) {
                externalArray[i] = int.Parse(inputArray[i]);
            }

            //Создаем объект класса ArrayProcessor
            ArrayProcessor processor = new ArrayProcessor(externalArray);

            //Вывод
            processor.PrintArray();
            Console.WriteLine($"Минимальное значение: {processor.MinValue}");
            Console.WriteLine($"Максимальное значение: {processor.MaxValue}");
        }
    }
}

```

```
        catch (Exception ex) {  
            Console.WriteLine($"О ш и б к а : {ex.Message}");  
        }  
    }  
}
```

Вывод: рассмотрели конструкторы, поля только для чтения, вызов конструкторов