**Классы в .NET. Специальные типы классов**

**Задание 1**

Создать класс А с целочисленными полями а и b и двумя методами согласно варианту.Внутри класса реализовать конструктор для инициализации a и b. Создать объект класса и продемонстрировать работу со всеми элементами класса. Метод вычисления суммы a и b, метод вычисления значения выражения .

**Код**

using System;

using System.Collections.Generic;

using System.Linq;

using System.Text;

using System.Threading.Tasks;

namespace Task1

{

internal class Program

{

static void Main(string[] args)

{

A obj = new A(2, 1);

int sumResult = obj.CalculateSum();

double expressionResult = obj.CalculateExpression();

Console.WriteLine($"Сумма a и b: {sumResult}");

Console.WriteLine($"Значение выражения: {expressionResult}");

}

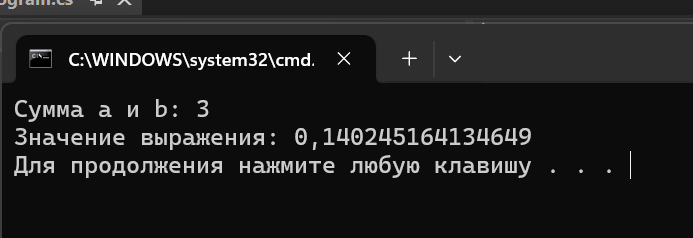
}

}

**Таблица 4.1 – Входные и выходные данные**

|  |  |
| --- | --- |
| **Входные данные** | **Выходные данные** |
| **1, 2** | **3, 0.1402** |

**Анализ результата**

****

**Задание 2**

Создайте static классы, которые выполняют различные операции с массивами объектов, включая сортировку, фильтрацию, вычисление статистики и генерацию данных.

Создать static класс ArrayUtils, содержащий метод GetMaxValue, который находит максимальный возраст среди массива объектов класса Person.

**Код**

using System;

using System.Collections.Generic;

using System.Linq;

using System.Text;

using System.Threading.Tasks;

namespace Task2

{

static class ArrayUtils

{

public static int GetMaxValue(Person[] people)

{

int maxAge = people[0].Age;

foreach (var person in people)

{

if (person.Age > maxAge)

{

maxAge = person.Age;

}

}

return maxAge;

}

}

}

**Таблица 4.2 – Входные и выходные данные**

|  |  |
| --- | --- |
| **Входные данные** | **Выходные данные** |
| **"Alice", 30, "Bob", 25, "Charlie", 35, "Diana", 28** | **35** |