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

Задание 1. Создайте static классы, которые выполняют различные операции с массивами объектов, включая сортировку, фильтрацию, вычисление статистики и генерацию данных. Добавить в ArrayUtils метод SortByName, который сортирует массив Person по имени в алфавитном порядке.

Листинг программы:

class Person

{ public string Name { get; set; }

public int Age { get; set; }

public Person(string name, int age)

{ Name = name;

Age = age; }

public void Display()

{ Console.WriteLine($"Имя: {Name},");

Console.WriteLine($"Возраст: {Age}."); }}

static class ArrayUtils

{ public static void SortByName(Person[] pe)

{ Array.Sort(pe, (p1, p2) => p1.Name.CompareTo(p2.Name)); }

public static Person[] FilterByAge(Person[] pe, int age)

{ return Array.FindAll(pe, pr => pr.Age > age); }

public static double CalculateAverageAge(Person[] pe)

{ double T = 0;

foreach (var pr in pe)

{ T += pr.Age; }

return T / pe.Length; }}

class Example

{ static void Main()

{ Person[] pe = new Person[]

{ new Person("Ангелина", 19),

… };

Console.WriteLine(" Информация ");

foreach (var pr in pe)

{ pr.Display(); }

ArrayUtils.SortByName(pe);

Console.WriteLine("\nПосле сортировки по имени...");

foreach (var pr in pe)

{ pr.Display(); }

var F = ArrayUtils.FilterByAge(pe, 17);

Console.WriteLine("\nСтарше 17 лет...");

foreach (var pr in F)

{ pr.Display(); }

double A = ArrayUtils.CalculateAverageAge(pe);

Console.WriteLine($"\nСредний возраст равен {A:F2} лет");

}

}

Таблица 1.1 – Входные и выходные данные

|  |  |
| --- | --- |
| Входные данные | Выходные данные |
| Ангелина 19, Дарья 18, Елизавета 17, Екатерина 8. | Информация, После сортировки по имени, Старше 17 лет, Средний возраст равен. |

Анализ результатов:

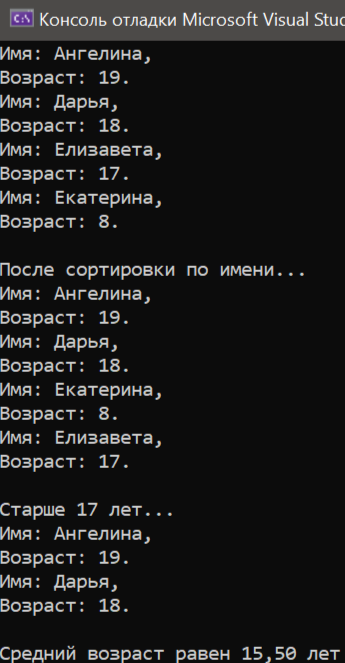


Рисунок 1.1 – Результат работы программы