Министерство образования и молодежной политики Свердловской области



ГАПОУ СО «Екатеринбургский колледж транспортного строительства»

# Отчёт по программе

# «Практическое занятие №16 (4 задания)»

Выполнил: Идрисов Руслан Русланович

Группа: ПР-22

Преподаватель: Мирошниченко Г.В

2023

**1. Задание:**

Напишите программу для отображения отсортированного списка стран в

соответствии с длиной строки (названия страны), затем по имени в порядке

возрастания, у которых численность населения больше n=104 000 000

человек (численность вводится с клавиатуры пользователем).

Тип данных – коллекция.

В текстовом файле записана информация о странах (название страны,

численность населения).

Россия 146 023 195

Сингапур 6 058 739

США 335 028 178

Франция 65 790 152

Индия 1 408 044 253

Бразилия 215 681 045

Египет 105 838 455

**Входные и выходные данные:**

**Входные:**

textBox2 (nameF) string – наименование файла с расширением из которого надо считать информацию и передать в listBox1

textBox1 (populationLimit) int – кол-во численности по которому нужно сортировать

**Выходные:**

listBox1 – отображает список стран, а также список отсортированных стран

**Листинг программы**

public partial class Form1 : Form

{

private List<Country> countries;

public Form1()

{

InitializeComponent();

}

private void button1\_Click(object sender, EventArgs e)

{

if (countries == null || countries.Count == 0)

{

MessageBox.Show("Сначала выгрузите информацию");

return;

}

int populationLimit;

if (int.TryParse(textBox1.Text, out populationLimit))

{

countries = countries.Where(c => c.Population > populationLimit).ToList();

countries = countries.OrderBy(c => c.Name.Length).ThenBy(c => c.Name).ToList();

listBox1.Items.Clear();

foreach (Country country in countries)

{

listBox1.Items.Add(country.Name + " " + country.Population);

}

}

else

{

MessageBox.Show("Введите корректное число");

}

}

private void button2\_Click(object sender, EventArgs e)

{

string nameF = textBox2.Text;

countries = new List<Country>();

if (File.Exists(nameF))

{

string[] lines = File.ReadAllLines(nameF);

foreach (string line in lines)

{

string[] parts = line.Split(' ');

string name = parts[0];

string chislStr = parts[1].Replace(" ", " ").Replace(" ", "");

int chisl;

if (int.TryParse(chislStr, out chisl))

{

countries.Add(new Country(name, chisl));

}

else

{

MessageBox.Show($"Численность населения: {chislStr}, для страны '{name}' не удалось преобразовать в число (возможно есть лишние пробелы)");

}

}

listBox1.Items.Clear();

foreach (Country country in countries)

{

listBox1.Items.Add(country.Name + " " + country.Population);

}

}

else

{

MessageBox.Show("Не удалось найти файл");

}

}

}

public class Country

{

public string Name { get; set; }

public int Population { get; set; }

public Country(string name, int population)

{

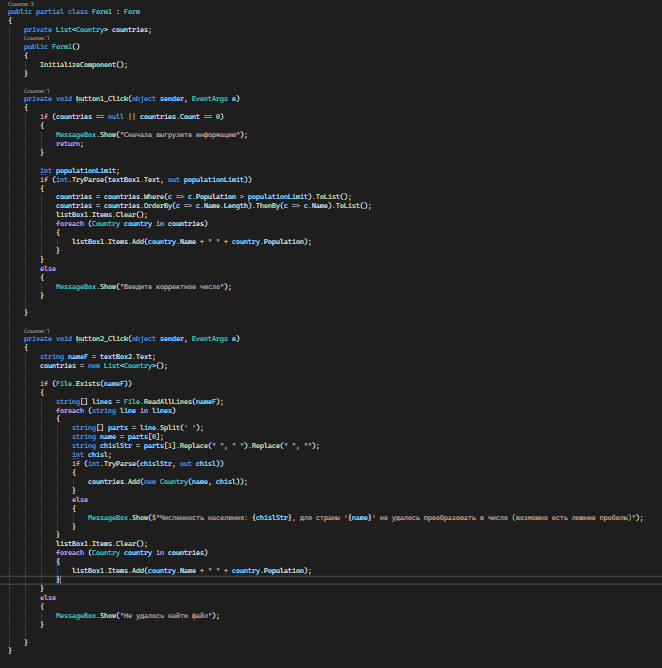
Name = name;

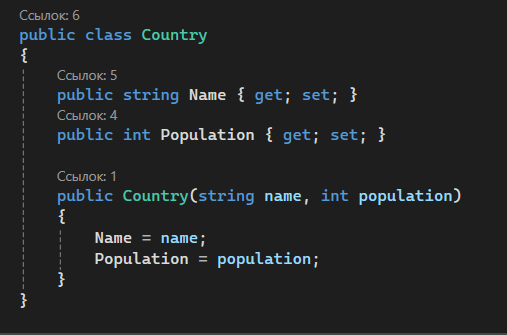
Population = population;

}

}

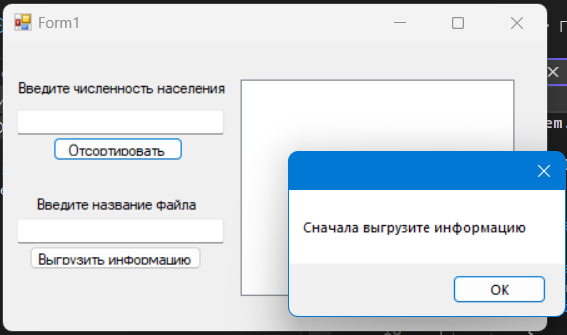
**Скрин-шот экранов (если есть)**

****

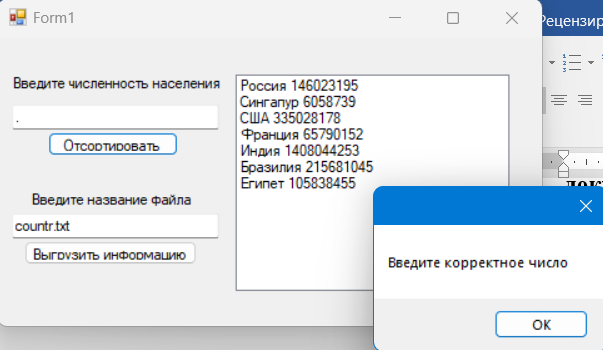
****

**Тестовые ситуации**

**1.Нельзя отсортировать список пока не выгрузили информацию из документа.**

****

**2.Численность должна быть обязательно числом**

****

**3. Если документ не найден выводится соответствующее сообщение**

