Министерство образования Республики Беларусь

Министерство высшего образования и науки Российской Федерации

МЕЖГОСУДАРСТВЕННОЕ ОБРАЗОВАТЕЛЬНОЕ УЧРЕЖДЕНИЕ ВЫСШЕГО ОБРАЗОВАНИЯ «БЕЛОРУССКО-РОССИЙСКИЙ УНИВЕРСИТЕТ»

Кафедра «Автоматизированные системы управления»

Лабораторная работа № 1

Работа с web-ресурсами

Выполнила

Студентка гр. АСОИ-181

Cушко Валерия Григорьевна

Проверила

Старший преподаватель

Выговская Наталья Владимировна

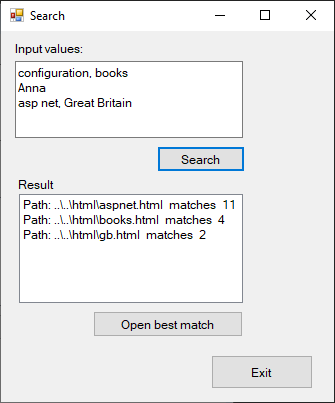
Могилёв 2021

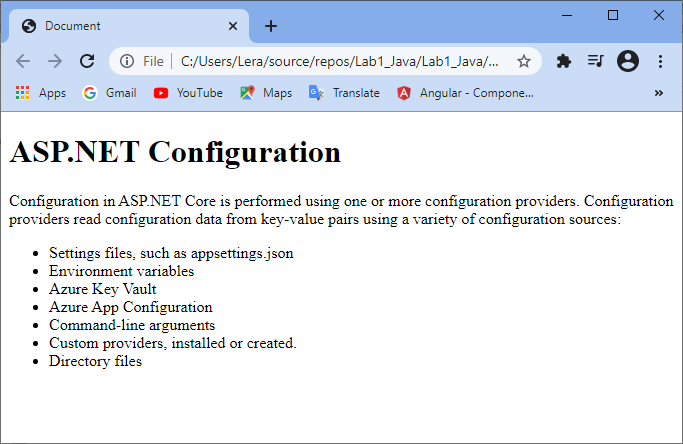
Цель работы: познакомиться с возможностями поиска информации в удаленных URL-ресурсах.

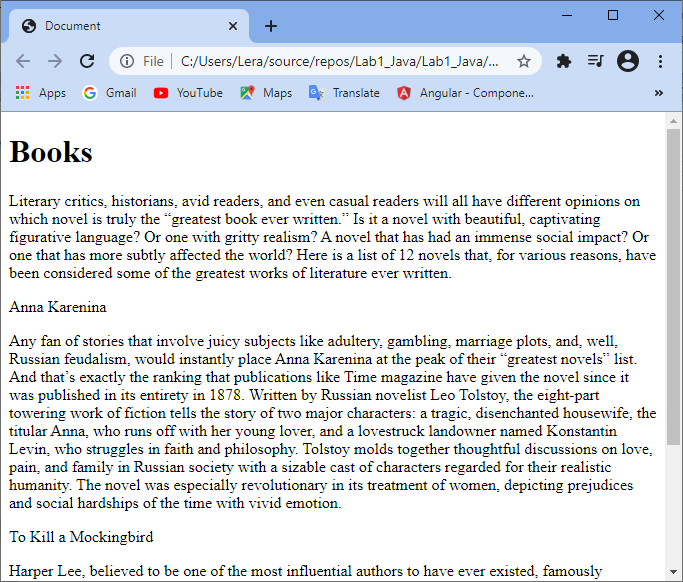
Задание.

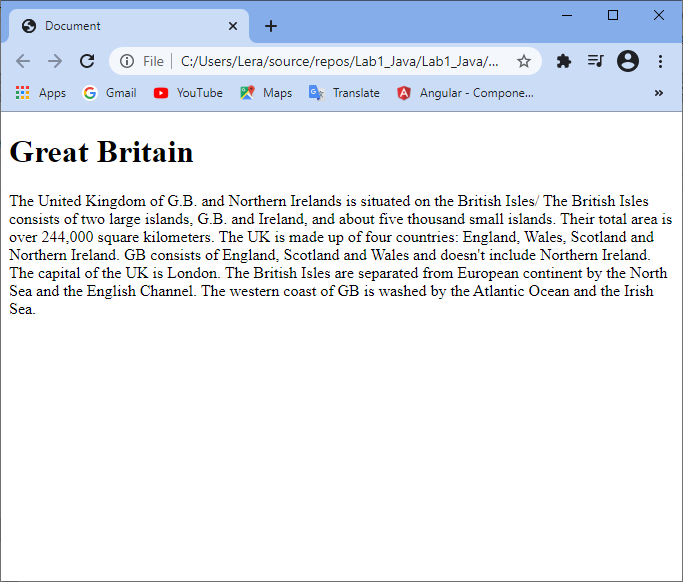
1. Подготовить собственный набор html-документов (согласовать с преподавателем).

2. Завершить заготовку программы так, чтобы файл с наибольшим числом совпадений открывался броузером.









using System;

using System.Collections.Generic;

using System.Diagnostics;

using System.IO;

using System.Linq;

using System.Text.RegularExpressions;

using System.Windows.Forms;

namespace Lab1\_Java

{

public partial class Form1 : Form

{

public static List<string> SearchWords;

public static List<string> HtmlPath = new List<string>();

public static IList<(string, int)> Results;

public Form1()

{

InitializeComponent();

HtmlPath.AddRange(Directory.GetFiles(@"..\..\html", "\*.html"));

this.button2.Enabled = false;

}

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

{

Application.Exit();

}

private void button3\_Click(object sender, EventArgs e)

{

var searchItems = this.textBox1.Text;

if(searchItems == "")

{

MessageBox.Show("Not input value for search", "Warning", MessageBoxButtons.OK, MessageBoxIcon.Warning);

}

else

{

SearchWords = new List<string>();

this.listBox1.Items.Clear();

SearchWords.AddRange(this.textBox1.Text.Split(new char[] { ',', '.', ' ', '\n' }, StringSplitOptions.RemoveEmptyEntries));

Search();

DisplayResults();

this.button2.Enabled = true;

}

}

private void Search()

{

Results = new List<(string, int)>();

foreach (var path in HtmlPath)

{

var doc = new HtmlAgilityPack.HtmlDocument();

doc.Load(path);

var nodes = doc.DocumentNode.SelectNodes("//body");

var matchesCount = 0;

foreach (var node in nodes)

{

foreach(var word in SearchWords)

{

matchesCount += Regex.Matches(node.InnerText, word, RegexOptions.IgnoreCase).Count;

}

}

Results.Add((path, matchesCount));

}

}

private void DisplayResults()

{

foreach(var result in Results)

{

this.listBox1.Items.Add($"Path: {result.Item1} matches {result.Item2}");

}

}

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

{

var path = FindBestMatch();

if(path == null)

{

MessageBox.Show("Not matches", "Information", MessageBoxButtons.OK, MessageBoxIcon.Information);

}

else

{

Process.Start("chrome.exe", path);

}

}

private string FindBestMatch()

{

var max = Results.Max(result => result.Item2);

if(max == 0)

{

return null;

}

var path = Results.FirstOrDefault(result => result.Item2 == max);

return path.Item1;

}

}

}