МГТУ им. Баумана  
Факультет «Информатика и системы управления»  
Кафедра «Автоматизированные системы обработки информации и управления»  
Дисциплина «Базовые компоненты интернет-технологий»

Отчёт по лабораторной работе №4

Выполнил:  
Студент группы ИУ5-33Б  
Кривцов Н.А.  
Преподаватель:  
Гапанюк Ю.Е.

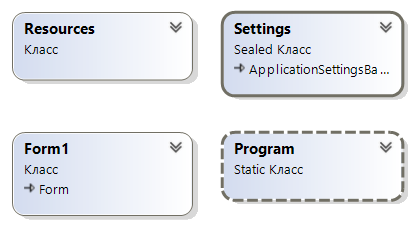
Описание задания:

Разработать программу, реализующую работу с файлами.

1. Программа должна быть разработана в виде приложения Windows Forms на языке C#. По желанию вместо Windows Forms возможно использование WPF.
2. Добавить кнопку, реализующую функцию чтения файла в список слов List<string>.
3. Для выбора имени файла используется класс OpenFileDialog, который открывает диалоговое окно с выбором файла. Ограничить выбор только файлами с расширением «.txt».
4. Для чтения из файла рекомендуется использовать статический метод ReadAllText() класса File (пространство имен System.IO). Содержимое файла считывается методом ReadAllText() в виде одной строки, далее делится на слова с использованием метода Split() класса string. Слова сохраняются в список List<string>.
5. При сохранении слов в список List<string> дубликаты слов не записываются. Для проверки наличия слова в списке используется метод Contains().
6. Вычислить время загрузки и сохранения в список с использованием класса Stopwatch (пространство имен System.Diagnostics). Вычисленное время вывести на форму в поле ввода (TextBox) или надпись (Label).
7. Добавить на форму поле ввода для поиска слова и кнопку поиска. При нажатии на кнопку поиска осуществлять поиск введенного слова в списке. Слово считается найденным, если оно входит в элемент списка как подстрока (метод Contains() класса string).
8. Добавить на форму список (ListBox). Найденные слова выводить в список с использованием метода «название\_списка.Items.Add()». Вызовы метода «название\_списка.Items.Add()» должны находится между вызовами методов «название\_списка.BeginUpdate()» и «название\_списка. EndUpdate()».

Вычислить время поиска с использованием класса Stopwatch. Вычисленное время вывести на форму в поле ввода (TextBox) или надпись (Label).

Диаграмма классов:



Текст программы:

Класс Form1:

using System;

using System.Collections.Generic;

using System.ComponentModel;

using System.Data;

using System.IO;

using System.Linq;

using System.Text;

using System.Diagnostics;

using System.Windows.Forms;

namespace Files {

public partial class Form1 : Form {

public Form1() {

InitializeComponent();

}

/// <summary>

/// Список слов WordList, содержащихся в файле.

/// </summary>

public List<string> WordList { get; set; }

/// <summary>

/// Создание списка слов WordList, содержащихся в файле. Активация остальных элементов управления формы.

/// </summary>

/// <param name="sender"></param>

/// <param name="e"></param>

private void FileSelectionDialog\_FileOk(object sender, CancelEventArgs e) {

//Сокращение полного имени файла для менее громоздкого отображения на экране

FileNameLabelShown.Text = "..." + FileSelectionDialog.FileName.Remove(0, FileSelectionDialog.FileName.LastIndexOf('\\')+1);

Stopwatch sw = new Stopwatch();

sw.Start();

string FileContents = File.ReadAllText(FileSelectionDialog.FileName);

string[] FileWords = FileContents.Split(' ', '\n', '\r', '!', ',', '?', '—', '.', '-', ':', ';');

List<string> list = new List<string>();

foreach (string s in FileWords) {

if (!list.Contains(s) && s != "") {

list.Add(s);

}

}

WordList = list;

sw.Stop();

ElapsedListCreationTime.Text = sw.ElapsedTicks.ToString() + " тиков";

WordSearchButton.Enabled = true;

WordSearchBox.Enabled = true;

WordListBox.Enabled = true;

WordListBoxClearButton.Enabled = true;

}

/// <summary>

/// Вызов диалога выбора текстового файла.

/// </summary>

/// <param name="sender"></param>

/// <param name="e"></param>

private void FileSelectionButton\_Click(object sender, EventArgs e) {

FileSelectionDialog.ShowDialog();

}

/// <summary>

/// Поиск слова в списке WordList и вывод найденных слов в WordListBox.

/// </summary>

/// <param name="sender"></param>

/// <param name="e"></param>

private void WordSearchButton\_Click(object sender, EventArgs e) {

Stopwatch sw = new Stopwatch();

sw.Start();

if (WordList.Contains(WordSearchBox.Text)) {

label1.Text = "Слово найдено";

if (!WordListBox.Items.Contains(WordSearchBox.Text)) {

WordListBox.BeginUpdate();

WordListBox.Items.Add(WordSearchBox.Text);

WordListBox.EndUpdate();

}

}

else {

label1.Text = "Слово не найдено";

}

sw.Stop();

ElapsedWordSearchTime.Text = sw.ElapsedTicks.ToString() + " тиков";

}

/// <summary>

/// Реализация функционала клавиши Enter при вводе слова для поиска в окне WordSearchBox

/// </summary>

/// <param name="sender"></param>

/// <param name="e"></param>

private void WordSearchBox\_Enter(object sender, EventArgs e) {

AcceptButton = WordSearchButton;

}

/// <summary>

/// Предотвращение некорректной работы клавиши Enter при неактивном окне WordSearchBox

/// </summary>

/// <param name="sender"></param>

/// <param name="e"></param>

private void WordSearchBox\_Leave(object sender, EventArgs e) {

AcceptButton = null;

}

/// <summary>

/// Возврат состояния программы к исходному.

/// </summary>

/// <param name="sender"></param>

/// <param name="e"></param>

private void WordListBoxClearButton\_Click(object sender, EventArgs e) {

WordListBox.Items.Clear();

WordList.Clear();

FileNameLabelShown.Text = "Нет";

WordSearchButton.Enabled = false;

WordSearchBox.Enabled = false;

WordListBox.Enabled = false;

WordListBoxClearButton.Enabled = false;

}

}

}

Класс Program:

using System;

using System.Collections.Generic;

using System.Linq;

using System.Threading.Tasks;

using System.Windows.Forms;

namespace Files {

static class Program {

/// <summary>

/// Главная точка входа для приложения.

/// </summary>

[STAThread]

static void Main() {

Application.EnableVisualStyles();

Application.SetCompatibleTextRenderingDefault(false);

Application.Run(new Form1());

}

}

}

Экранные формы:

