Министерство науки и высшего образования Российской Федерации

Федеральное государственное бюджетное образовательное учреждение высшего образования

«Рязанский государственный радиотехнический университет

имени В.Ф.Уткина»

Кафедра ЭВМ

Отчет по лабораторной работе №10

на тему

«Работа с файловым потоком»

по дисциплине  
**«Визуальное программирование»**

Выполнили:

Студенты группы №140

Бригада 5

Сафаров Д. А.

Тимохин Е. С.

Проверили:  
ст. преп. Хизриева Н.И.

ст. преп. Бастрычкин А.С.

**Цель работы**

Получить навыки работы с потоковым чтением и записью на языке C#.

**Задание**

Разработать класс FileClass для работы с текстовым файлом.

В классе должны быть:

* Закрытый конструктор, принимающий путь, по которому будет создан новый файл;
* Статический метод Create принимающий путь для нового файла и возвращает объект, созданный конструктором;
* Закрытый конструктор, принимающий путь, по которому будет открыт существующий файл;
* Статический метод Open принимающий путь существующего файла и возвращающий объект, созданный конструктором;
* Метод ReadLine для чтения строки из файла;
* Метод Read для чтения count символов из файла;
* Метод Writeline для записи строки в файл;
* Метод Write для записи count символов в файл.

Для чтения и записи использовать классы StreamReader и StreamWriter (поля reader и writer). Эти классы должны работать с одним и тем же потоком FileStream. Они должны создаваться один раз в закрытом методе CreateStream. Этот метод должен использоваться в конструкторах.

Файл должен открываться только один раз. reader и writer должны создаваться только один раз, а не при каждом обращении к файлу. Поток, reader и writer должны закрываться тоже только один раз.

Не использовать конструкцию using. Разработать метод Close для сохранения изменений и освобождения ресурсов.

Продемонстрировать работу с созданным классом.

**Практическая часть**

Код программы представлен в приложении А. Результат её работы представлен на рисунке 1.



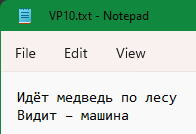


Рисунок 1 – Результаты работы программы

Приложение А. Текст программы

**FileClass.cs**

namespace FileClassLogic

{

public class FileClass

{

private FileStream? \_stream;

private StreamReader? \_reader;

private StreamWriter? \_writer;

public bool isClosed { get; private set; } = true;

private FileClass(string filePath, bool isNewFile)

{

if(CreateStream(filePath, isNewFile))

isClosed = false;

}

~FileClass()

{

Close();

}

public static FileClass Create(string filePath) =>

new FileClass(filePath, true);

public static FileClass Open(string filePath) =>

new FileClass(filePath, false);

public string? ReadLine()

{

if (isClosed) throw new Exception("Error: cannot read from closed file!");

return \_reader.ReadLine();

}

public string Read(ushort count)

{

if (isClosed) throw new Exception("Error: cannot read from closed file!");

var buffer = new char[count];

if (\_reader.Read(buffer, 0, Convert.ToInt32(count)) == 0)

return "";

else

return new string(buffer);

}

public void WriteLine(string str)

{

if (isClosed) throw new Exception("Error: cannot write to closed file!");

\_writer.WriteLine(str);

\_writer.Flush();

}

public void Write(string str, ushort count)

{

if (isClosed) throw new Exception("Error: cannot write to closed file!");

\_writer.Write(str.Substring(0, Convert.ToInt32(count)));

\_writer.Flush();

}

public void Close()

{

if (isClosed) return;

\_writer.Dispose();

\_reader.Dispose();

\_stream.Dispose();

isClosed = true;

}

private bool CreateStream(string filePath, bool isNewFile)

{

if (isNewFile)

\_stream = new FileStream

(

filePath,

FileMode.Create,

FileAccess.ReadWrite,

FileShare.ReadWrite | FileShare.Delete

);

else

try

{

\_stream = new FileStream

(

filePath,

FileMode.Open,

FileAccess.ReadWrite,

FileShare.ReadWrite | FileShare.Delete

);

}

catch (FileNotFoundException)

{

Console.WriteLine($"\n Error: file '{filePath}' not found!");

return false;

}

\_reader = new StreamReader(\_stream);

\_writer = new StreamWriter(\_stream);

return true;

}

}

}

**Program.cs**

using FileClassLogic;

string filePath = @"C:\VP10.txt";

/\* Создание файла и запись\*/

var newFile = FileClass.Create(filePath);

newFile.WriteLine("Идёт медведь по лесу"); // "Идёт медведь по лесу"

newFile.Write("Видит – машина горит", 5+3+6); // "Видит – машина"

newFile.Close();

/\* Открытие файла и чтение\*/

var file = FileClass.Open(filePath);

file.Read(4+1); // "Идёт "

string bear = file.Read(7); // "медведь"

file.ReadLine(); // "по лесу\n"

Console.WriteLine(file.ReadLine().Replace("машина", bear)); // "Видит - медведь"

file.Close();

/\* Удаление файла \*/

Console.WriteLine("\n> Пожалуйста, нажмите Enter...");

Console.Read();

File.Delete(filePath);