**Лабораторная работа №7**

**Делегаты, события**

**Задание 1**

**Требования:**

* В каждой задаче необходимо создать пользовательский делегат.
* Должно быть два класса с методами, которые будут обработчиками делегата.
* Необходимо продемонстрировать работу с методами через делегат.
* Уровень сложности: **3**.

**Делегат для обработки сообщений**

Создайте делегат MessageHandler, который принимает строку.  
Создайте два класса ConsoleLogger и FileLogger, методы которых будут обрабатывать сообщения.  
Демонстрируйте работу с делегатом.

**Код**

using System;

using System.Collections.Generic;

using System.Linq;

using System.Text;

using System.Threading.Tasks;

using System.IO;

namespace Task1

{

public delegate void MessageHandler(string message);

internal class Program

{

static void Main(string[] args)

{

ConsoleLogger consoleLogger = new ConsoleLogger();

FileLogger fileLogger = new FileLogger("log.txt");

MessageHandler messageHandler = consoleLogger.Log;

messageHandler += fileLogger.Log;

messageHandler("Сообщение отправилось.");

Console.WriteLine("в папку log.txt:");

Console.WriteLine(File.ReadAllText("log.txt"));

}

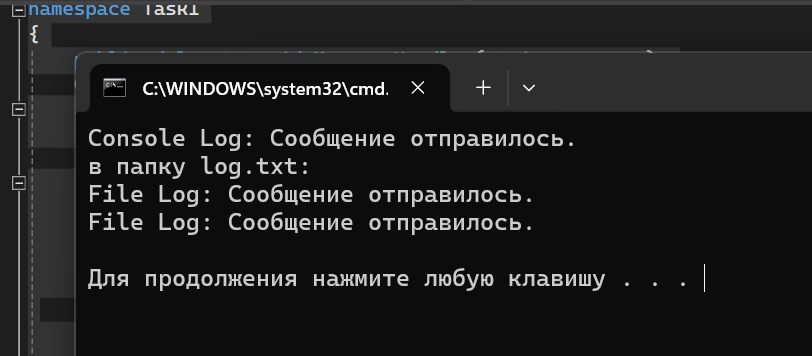
}

}

**Таблица 7.1 – Входные и выходные данные**

|  |  |
| --- | --- |
| **Входные данные** | **Выходные данные** |
| **Сообщение отправлено** | **Сообщение отправлено** |

**Анализ результата**

****

**Задание 2**

**Требования:**

* В каждой задаче организовать передачу делегата в качестве параметра метода.
* Использовать два метода в качестве **callback** (методы обратного вызова).
* Уровень сложности: **3**.

**Делегат для обработки строк**

Создайте делегат StringProcessor, который принимает строку.  
Метод ProcessString должен принимать строку и делегат, который обрабатывает её.  
Создайте два метода: ToUpperCase и ToLowerCase. Передавайте их в ProcessString.

**Код**

using System;

using System.Collections.Generic;

using System.Linq;

using System.Text;

using System.Threading.Tasks;

namespace Task2

{

public delegate string StringProcessor(string input);

internal class Program

{

static void Main(string[] args)

{

string testString = "Привет Мир!";

string upperResult = ProcessString(testString, ToUpperCase);

Console.WriteLine($"Верхний регистор: {upperResult}");

string lowerResult = ProcessString(testString, ToLowerCase);

Console.WriteLine($"Нижний регистор: {lowerResult}");

}

public static string ProcessString(string input, StringProcessor processor)

{

return processor(input);

}

public static string ToUpperCase(string input)

{

return input.ToUpper();

}

public static string ToLowerCase(string input)

{

return input.ToLower();

}

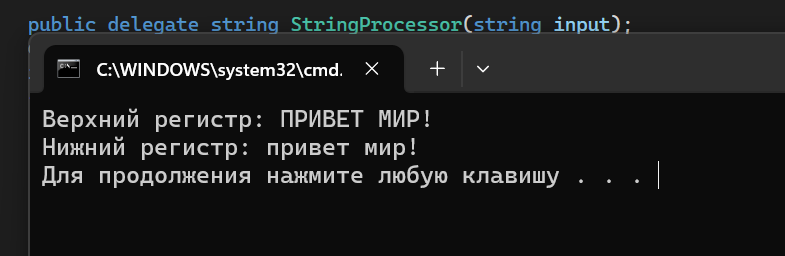
}

}

**Таблица 7.2 – Входные и выходные данные**

|  |  |
| --- | --- |
| **Входные данные** | **Выходные данные** |
| **Привет мир !** | **Нижний регистр: привет мир!**  **Верхний регистр: ПРИВЕТ МИР!** |

**Анализ результата**

****