|  |
| --- |
| , RD Dep. |
| Практическое задание  NET.C#.01. Введение в .NET Framework 4 |

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| REVISION HISTORY | | | | | |
| Ver. | Description of Change | Author | Date | Approved | |
| Name | Effective Date |
| 1.0 | Initial version | Анжелика Кравчук |  |  |  |
| 1.1 | Review and corrections. | Владимир Тихон |  |  |  |
| 1.2 | Added important requiments and links | Oleg |  |  |  |

**Задание.**

* Разработать консольное приложение для чтения небольшого набора входных данных, форматирования этих данных удобными для чтения образом и отображения отформатированных результатов. Данные представляют собой текст, содержащий пары чисел, представляющие х и у- координаты местоположения объекта. Каждая строка текста содержит один набор координат. Следующий пример демонстрирует типичный набор данных.

23.8976,12.3218  
25.76,11.9463  
24.8293,12.2

В отформатированном виде данные могут выглядеть следующи образом:

X: 23,8976 Y: 12,3218  
X: 25,76 Y: 11,9463  
X: 24,8293 Y: 12,2

* Перенаправить ввод для запуска приложения с помощью текстового файла, содержащего данные, которые нужно отформатировать.
* Реализовать простое консольное (или WPF) приложение, обеспечивающее описанные выше функциональные возможности. Предусмотреть возможность введения данных пользователем путем предоставления поля.
* Перенаправить ввод для запуска приложения с помощью данных из файла. Создать дополнительные тестовые данные и использовать их в качестве входных данных для тестирования приложения.
* Использовать Visual Studio 2010 (или 2012) отладчик для пошагового выполнения кода и изучения работы приложения.
* Добавить XML комментарии к разработанному приложению и создать для него документацию, используя инструмент Sandcastle.

**Важные моменты по реализации (!):**

* Для чисел применить тип Decimal.
  + Для распознавания чисел применить стандартные методе типа.
* Для вывода применить «форматированный вывод чисел» и «форматированный вывод строки с параметрами-объектами».
* Название проекта, название классов и название файлов (с кодом, тестовыми данными, командного файла) должны соответствовать терминам в тексте задания и быть простыми для чтения и понимания.

**Важная теория для выполнения задания (!):**

* Как перенаправить результат выполнения команд в файл

<http://www.windowsfaq.ru/content/view/11/45/>

* Using command redirection operators

<http://technet.microsoft.com/en-us/library/bb490982.aspx>

* Start, Break, Step, Run through Code, and Stop Debugging in Visual Studio

<http://msdn.microsoft.com/en-us/library/vstudio/y740d9d3.aspx>

<http://msdn.microsoft.com/ru-ru/library/y740d9d3.aspx>

* Breakpoints: Use Hit Counts, Call Stack Functions, and Conditions to Break When and Where You Want in the Visual Studio Debugger

<http://msdn.microsoft.com/en-us/library/vstudio/5557y8b4.aspx>

* Viewing Data in the Debugger

<http://msdn.microsoft.com/en-us/library/vstudio/esta7c62.aspx>

* How to: Use the QuickWatch Dialog Box

<http://msdn.microsoft.com/en-us/library/vstudio/cyzbs7s2.aspx>

* How to: Use the Call Stack Window

<http://msdn.microsoft.com/en-us/library/vstudio/a3694ts5.aspx>

**Требования к предоставлению результатов (!):**

* Прислать документацию по приложению, созданную в Sandcastle.
* Прислать скриншот с декомпилированным il-кодом созданного приложения, открытом в Reflector 6.8.2.5.
* Прислать скриншоты с декомпилированным c#- кодом созданного приложения (откомпилированным в Debug-режиме), открытом в JetBrains DotPeek.
  + Прислать скриншоты с декомпилированным c#- кодом созданного приложения (откомпилированным в Realese-режиме), открытом в JetBrains DotPeek.
    - Описать отличия с Debug-режимом.
* Прислать скриншоты с декомпилированным c#- кодом метода ToCharArray(int startIndex, int length) класса string.
* Во время отладки установить переменной, содержащей строку данных, значение null.
  + Сделать скриншоты до и после изменения знания.
  + Указать все способы, которыми можно изменить значение переменной во время отладки.

Дополнительные задания:

##### **Задание 1. Компиляция и запуск приложения из командной строки**

##### Дан код:

*Logic.cs*

using System;

namespace ConsoleApp

{

**public** class Logic

    {

**public** string Method()

        {

            return "I am string in Logic.";

        }

    }

}

*Program.cs*

using System;

namespace ConsoleApp

{

**public** class Program

    {

**static** void Main(string[] args)

        {

**Logic** logic = new **Logic**();

            Console.WriteLine(logic.Method());

        }

    }

}

Необходимо скомпилировать и запустить данный код из консоли - CSC.

**Задание 2. Классы и объекты**

Спроектировать и разработать классы *Матрица*. Реализовать методы: *Заполнить матрицу случайными значениями*, *Сложить две матрицы, Умножить две матрицы, Вывод элементов матрицы*. Написать для данного кода xml-документацию.