
EPAM Systems, RD Dep.

Homework Task

04 – Basics of C#

REVISION HISTORY					
Ver.	Description of Change	Author	Date	Approved	
				Name	Effective Date
1.0	Initial version	Dmitrii_Turlychkin	24.02.2015		
2.0		Nikolay Piskarev	29.05.2017		

Задание 1

Написать программу, которая определяет среднюю длину слова во введенной текстовой строке. Учесть, что символы пунктуации на длину слов влиять не должны. Регулярные выражения не использовать. И не пытаться прописать все ручками. Используйте стандартные методы класса String.

Задание 2

Написать программу, которая удваивает в первой введенной строки все символы, принадлежащие второй введенной строке.

Пример:

Введите первую строку: написать программу, которая

Введите вторую строку: описание

Резльтирующая строка: ннаапппииссааь ппроограамму, коотоораая

Задание 3

Проведите сравнительный анализ скорости работы классов String и StringBuilder для операции сложения строк:

```
string str = "";
StringBuilder sb = new StringBuilder();
int N = 100;
for (int i = 0; i < N; i++)
{
    str += "*";
}
for (int i = 0; i < N; i++)
{
    sb.Append("*");
}
```

Требования к оформлению

1. Для каждой домашней работы необходимо создать собственную папку в корне репозитория.
2. Каждое задание должно представлять собой отдельный проект (project) в рамках общего решения (solution).
3. Правила именования:
Все названия и имена должны быть переведены на английский язык. Именованье транслитерацией не допускается.
 - a. Маска именования репозитория: EPAM_Ext_Lab_<Квартал старта группы>_<Год>_<Фамилия_Имя>.
Например, EPAM_Ext_Lab_Q2_2017_Ivanov_Ivan
 - b. Маска именования решения: HMT_<Порядковый номер>.
Например, HMT_01, HWT_11.
 - c. Маска именования проектов: <Задача><Номер задачи>.
Например, Task01, Task12.
 - d. Каждое задание должно быть оформлено следующим образом:
 1. Формулировка задания в виде комментария (просто копировать-вставить из этого файла).
 2. Код, решающий задачу.
4. Написать письмо преподавателю, информирующее о том, что домашняя работа выполнена. Тема письма должна выглядеть так: [STYYYY_N]Фамилия_Имя_TaskNN. Здесь ST – сокращённое наименование программы (Students Training), YYYY – номер года, N – номер группы, NN – порядковый номер задания в виде двух цифр, напр. 01, 05, 12 и т.д.
Например, ST2017_1 Иванов Иван Task01.

Рекомендации к оформлению кода

1. При добавлении нового файла желательно вверху файла добавить блок с комментарием, для чего предназначен этот файл.
2. При добавлении метода в блоке `<summary></summary>` описать предназначение метода, его входных/выходных параметров и возвращаемых значений.
3. Использовать утилиту StyleCop для проверки правил оформления кода. Перед отправкой задания на проверку запускать StyleCop и исправлять все предупреждения, возникающие в секции Warning Visual Studio. Предупреждения об отсутствии документации можно игнорировать (либо настроить StyleCop таким образом, чтобы он не проверял документирование кода).