**Практическая работа № 12**

**StringBuilder**

1. ***Цель работы.*** Получение навыков составления программ с классом StringBuilder в среде Visual Studio.Net

***2. Литература.***  Т.А.Павловская “С#. Программирование на языке высокого уровня”, И.Г.Семакин, А.П.Шестаков «Основы программирования», Т.А.Павловская, Ю.А.Щупак «С/С++. Структурное программирование. Практикум», Е.Л.Шиманович «С/С++ в примерах и задачах», Эндрю Троелсен «Язык программирования С#2005 и платформа .Net2.0»

***3. Подготовка к работе.***  Изучить конспект лекций по темам «Строки и символы».

***4. Перечень оборудования.*** Интегрированная среда Visual Studio.Net.

***5. Задание.***

1. Научиться составлять программы в интегрированной среде Visual Studio.Net.

***6. Порядок выполнения работы***.

**1. Конкатенация строк** Создай StringBuilder, в который последовательно добавь строки "С#", " " и "Programming". Выведи получившуюся строку.

**2. Инвертирование строки** Сделай её инверсию и выведи результат.

**3. Удаление символов** Добавь текст "C# is cool and fun!". Удали слова "and fun" и выведи изменённую строку.

**4. Вставка текста** Добавь к ней в середину слово "С#". Используй метод insert.

**5. Поиск и замена**. Найди слово "cool" и замени его на "best" с помощью метода replace.

**8. Извлечение подстроки** Напиши программу, которая принимает строку и извлекает из неё подстроку с помощью substring. Сначала выведи строку, а затем её часть с 3-го по 7-й символ.

9. **Зашифровать текст**



Привет, меня зовут ваше ФИО.

9. **Расшифровать текст**

Ею фцтзу

Методы класса StringBuilder:

|  |  |
| --- | --- |
| Append() | Добавляет указанное количество подстрок |
| AppendLine() | Добавляет строку к текущей строке |
| AppendFormat() | Добавляет строку, сформированную в соответствии со спецификатором формата |
| Insert() | Вставляет подстроку в строку |
| Remove() | Удаляет символ из текущей строки |
| Replace() | Заменяет все вхождения символа другим символом или вхождения подстроки другой подстрокой |
| ToString() | Возвращает текущую строку в виде объекта System.String (переопределение метода класса System.Object) |