

ЛАБОРАТОРНА РОБОТА №9

Тема: Обробка символьної інформації.

Мета роботи

Навчитися працювати зі строковими та символьними змінними мови C#.

Постановка завдання

Розробити проект з інтерфейсом користувача, використовуючи необхідні компоненти та оброблювач подій мовою C#.

Теоретичні відомості

1. Тип даних `string` та `char`

Для опису рядка (строки) використовують тип `string`:

```
string str = "Деякий текст";
```

Тип `char` представляє собою один символ, який можна описати так:

```
char ch = 'A';
```

Деякі символи у строковому літералі (так називається конструкція у лапках) потрібно записувати у коді по особливому, у escape послідовності:

Послідовність	Справжнє значення
<code>\0</code>	символ з кодом 0
<code>\n</code>	перехід на нову строку
<code>\t</code>	символ табуляції
<code>\\</code>	<code>\</code>
<code>\'</code>	<code>'</code>
<code>\"</code>	<code>"</code>
<code>\udddd</code>	Unicode символ з кодом 4-цифрового числа dddd

Це дозволяє у коді описати строку, яка містить службовий символ, який використовується для закінчення літералу:

```
string motto = "\"Відповідність часу\"";  
string path = "D:\\Programming\\Projects";  
char apostrophe = '\\';
```

Якщо у строковому літералі використовується багато службових символів, то можна його описати дослівно за допомогою символу `@`:

```
string path = @"D:\Programming\Projects\WFA\bin\Debug";
```

Тоді для символу `"` треба використовувати послідовність `""`:

```
string motto = @""""Відповідність часу"""";
```

2. Операції та функції

Для об'єднання символьних даних використовують оператор +:

```
string firstLine = "Мудрість хай довічно",  
        secondLine = "З юністю цвіте!",  
        end = "наш гірничий університет!";  
char quotMark = '"', newline = '\n';  
string result = firstLine + newline;  
result += secondLine + newline;  
result += quotMark + "Vivat" + quotMark + ' ' + end;
```

Функції класу System.Char:

Функція	Результат
Char.IsDigit	true якщо символ є цифрою
Char.IsLetter	true якщо символ є літерою
Char.IsLetterOrDigit	true якщо символ є літерою або цифрою
Char.IsLower	true якщо символ є літерою у нижньому регістрі
Char.IsUpper	true якщо символ є літерою у верхньому регістрі
Char.IsPunctuation	true якщо символ є розділовим знаком
Char.ToLower	повертає символ у нижньому регістрі
Char.ToUpper	повертає символ у верхньому регістрі

Функції класу System.String:

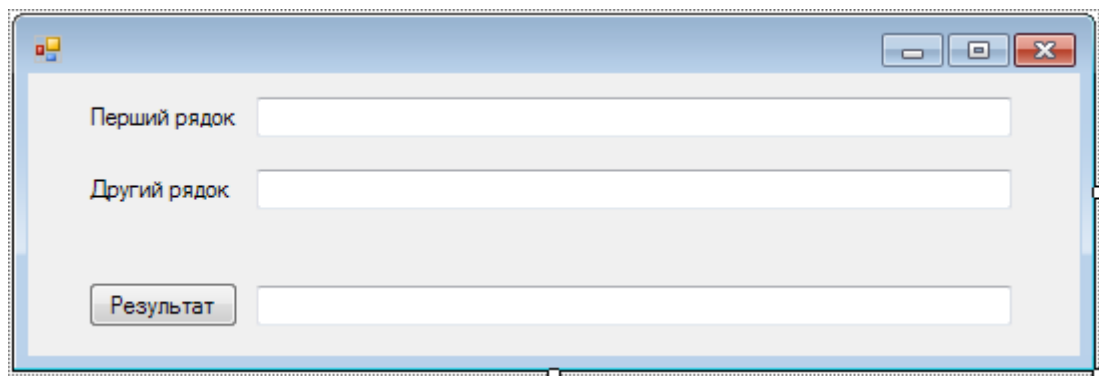
Функція	Результат
змінна[n]	n-ний символ рядка
змінна.Length	кількість символів у рядку
String.Compare(a,b)	виконується лексичне порівняння рядків: повертає значення більше нуля, якщо a>b; повертає значення менше нуля, якщо a<b
String.Concat	об'єднані елементи у вказаній послідовності
String.Copy	копія рядка як нового об'єкту
змінна.CopyTo	копіює задану кількість символів у масив
змінна.IndexOf	індекс першого елемента, рівного заданому
змінна.LastIndexOf	індекс останнього елемента, рівного заданому
змінна.Insert	вставляє заданий рядок у задану позицію
String.Join	об'єднує рядки, розділяючи їх заданим символом
змінна.Remove	рядок, з якого видалена задана кількість символів

Функція	Результат
змінна.Replace	рядок, з заміненними елементами
змінна.Substring	задана частина рядка
змінна.ToLower	рядок у нижньому регістрі
змінна.ToUpper	рядок у верхньому регістрі
змінна.Trim	рядок без символів пробілу на початку та кінці
змінна.Split	масив з рядків, які у початковому рядку були розділені одним із заданих символів

Приклад виконання завдання

Видалити з першого рядка усі символи, які є у другому рядку.

Розв'язання



```
private void button1_Click(object sender, EventArgs e)
{
    if (textBox1.Text != "" && textBox2.Text != "")
    {
        string t = textBox1.Text,
               r = textBox2.Text;
        for (byte i = 0; i < r.Length; i++)
            t = t.Replace(r[i], '\u200B'); // 200B - пробіл нульової ширини
        textBox3.Text = t;
    }
}
```