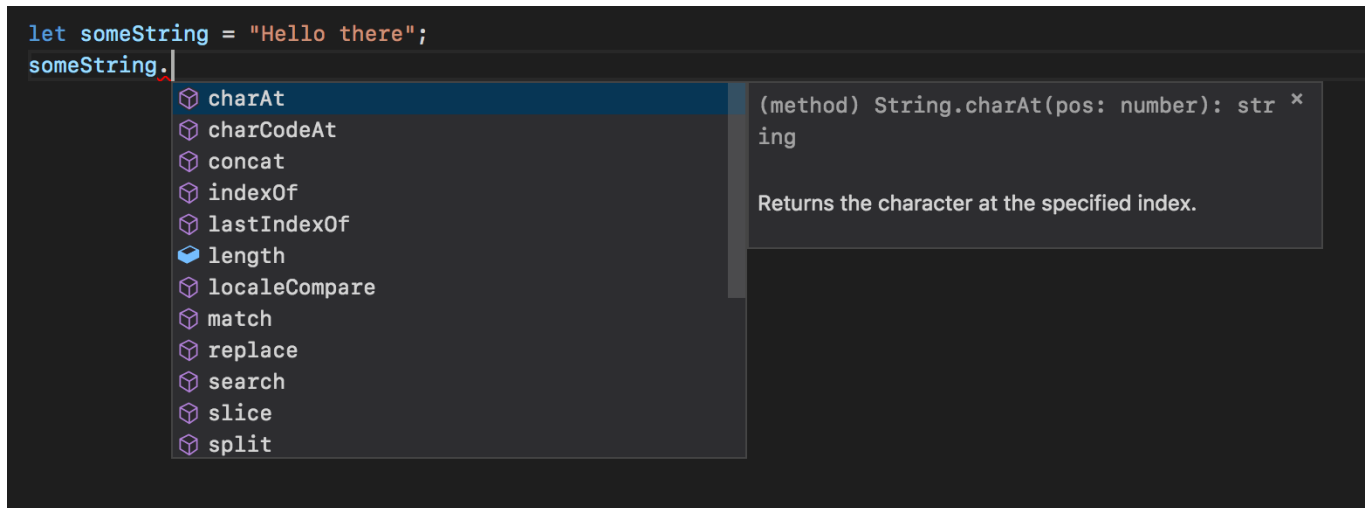


UniLecs #136. Custom String.IndexOf()

UniLecs • October 29, 2018



string functions in Typescript

Задача: в исходной строке необходимо найти первое и последнее вхождение заданного символа. Если символа в строке нет, выведите -1.

Входные данные: `str` - входная строка в кодировке Unicode, кол-во символов в строке не больше 10^3 . `Symbol` - символ в кодировке Unicode.

Вывод: первое и последнее вхождение заданного символа в исходной строке.

Пример:

1. `str = "AbcAde"; symbol = 'A'; Answer = (0, 3)`
2. `str = "bbcAde"; symbol = 'A'; Answer = (3, 3)`
3. `str = "bbcdef"; symbol = 'A'; Answer = (-1, -1)`

Идея: посимвольно обрабатываем исходную строку. В отдельной переменной храним кол-во вхождений заданного символа и запоминаем индекс его первого и последнего вхождения.

Реализация: в качестве реализации данного метода используем метод расширения в C# для строкового типа. В качестве выходного типа для метода используем кортеж вида `Tuple<int, int>`, где первый параметр - индекс первого вхождения, а второй - последнего вхождения заданного символа в исходную строку.

Подробнее про кортежи в C# вы можете почитать тут:

- C# tuple types
- Кортежи

```
1  using System;
2
3  // Метод расширения для типа String
4  public static class StringExtension
5  {
6      // Кастомная реализация метода String.IndexOf()
7      // Возвращаем кортеж Tuple<int, int>,
8      // где первый параметр - 1е вхождение заданного символа, второй параметр - последнее вхождение
9      public static Tuple<int, int> CustomIndexOf(this string str, char symbol)
10     {
11         int firstIndex = -1, lastIndex = -1, count = 0;
12         for (int i = 0; i < str.Length; i++)
13         {
14             if (str[i] == symbol)
15             {
16                 if (count == 0)
17                 {
18                     firstIndex = i;
19                 } else {
20                     lastIndex = i;
21                 }
22                 count++;
23             }
24         }
25         lastIndex = lastIndex == -1 ? firstIndex : lastIndex;
26         return new Tuple<int, int>(firstIndex, lastIndex);
27     }
28 }
29
30 public class Program
31 {
32     public static void Main()
33     {
34         Console.WriteLine("UniLecs");
35         Console.WriteLine(string.Format("Answer = {0}", "AbcAde".CustomIndexOf('A'))); // (0, 3)
36         Console.WriteLine(string.Format("Answer = {0}", "bbcAde".CustomIndexOf('A'))); // (3, 3)
37         Console.WriteLine(string.Format("Answer = {0}", "bbcdef".CustomIndexOf('A'))); // (-1, -1)
38     }
39 }
```

C#

<https://gist.github.com/unilecs/465c9dco4b4f84e335c8f2a548f36ae1>

Test:

<https://dotnetfiddle.net/AmvTeq>