

СИМВОЛНИ НИЗОВЕ

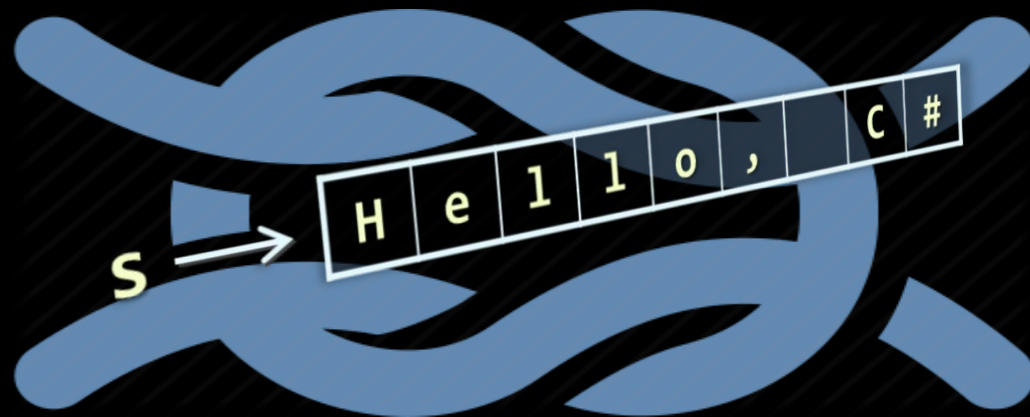
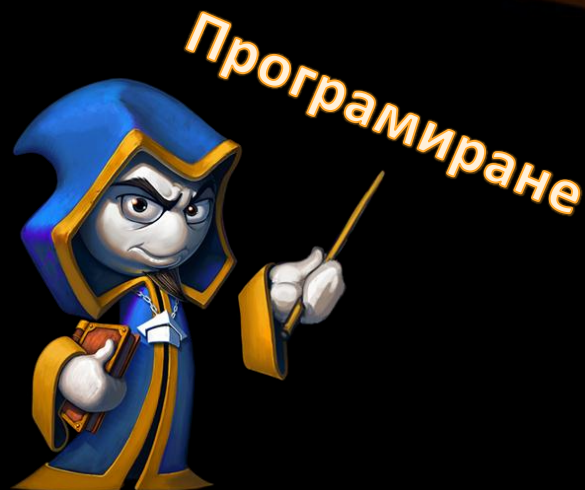
Текстов тип данни. Работа с текст



Учителски екип
Обучение за ИТ кариера

<https://it-kariera.mon.bg/e-learning/>

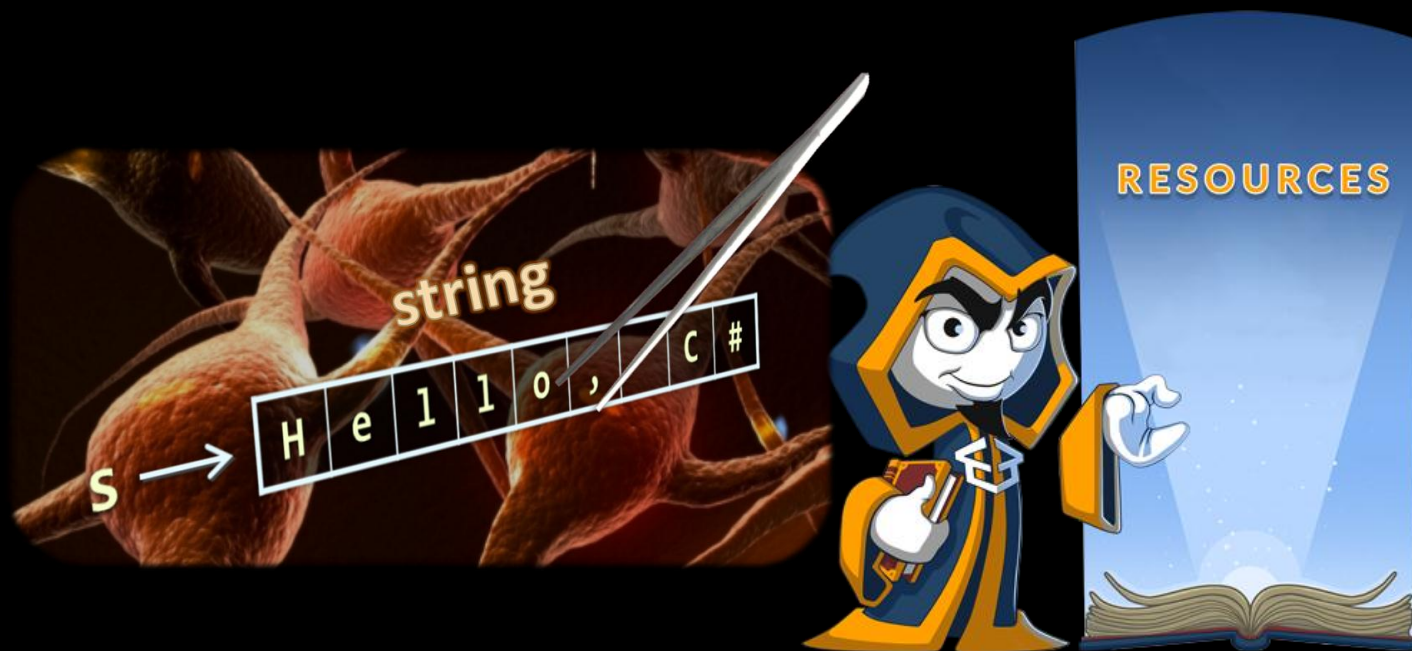
<https://github.com/BG-IT-Edu/School-Programming/tree/main/Courses/Applied-Programmer/Programming-Fundamentals>



Съдържание

1. Какво е **String**?
2. Какво означава **Immutable**?
3. Въведение в обработката на символни низове

- Сравняване
- Съединяване
- Търсене
- Извличане на подниз
- Разделяне



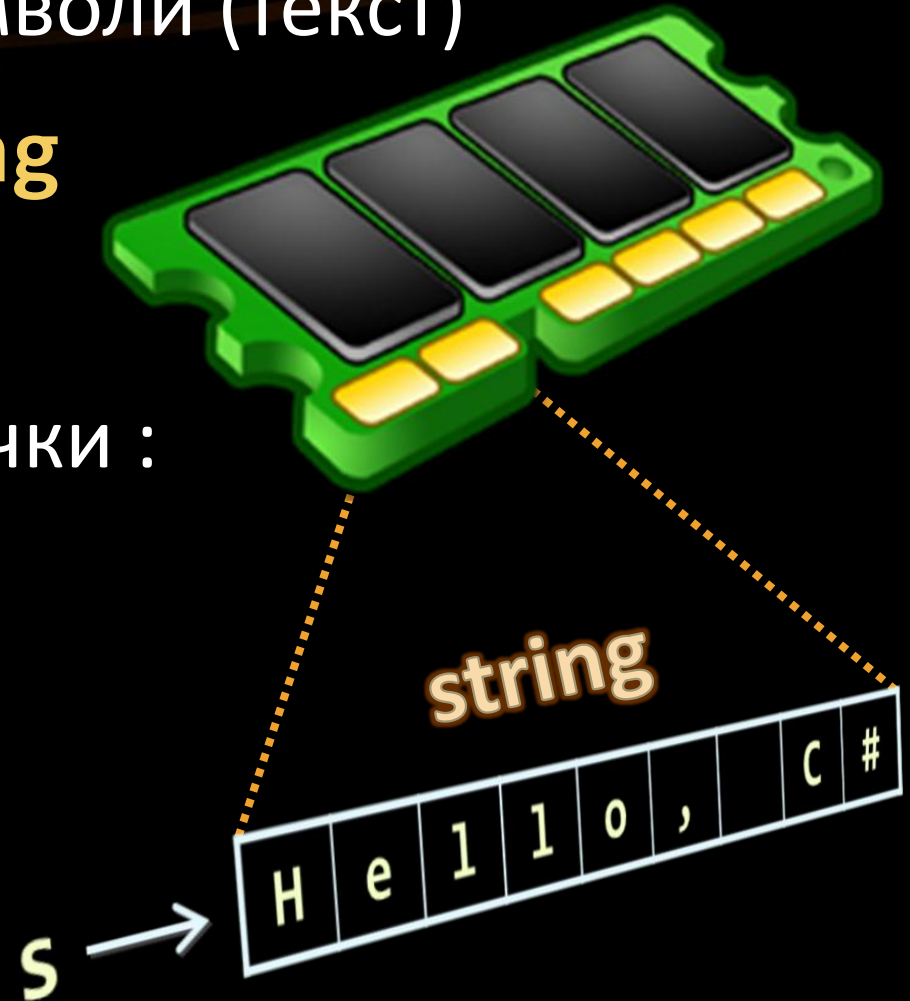
СИМВОЛНИ НИЗОВЕ

- Символните низове са поредица от символи (текст)
- Декларират се с ключовата дума **string**
 - Синоним е на **System.String** в .NET
- Символните низове се ограждат с кавички :

```
string s = "Hello, C#";
```

- Операторът "+" слепва низовете:

```
string s = "Hello" + " " + "C#";
```



В C# символните низове са **Immutable**, използват **Unicode**

- Низовете са **immutable** (read-only) поредици от символи
 - Достъпни са по индекс (read-only)

```
string str =  
    "Hello, C#";  
let ch = str[2]; // ОК  
str[2] = 'a'; // Грешка!
```

index =	0	1	2	3	4	5	6	7	8
str[index] =	Н	е	l	l	о	,		С	#

- Символните низове **Unicode** (може да се ползват много и най-различни азбуки, например Арабски):

```
string greeting = "السَّلَامُ عَلَيْكُمْ"; // As-salamu alaykum
```

Инициализация на символни низове

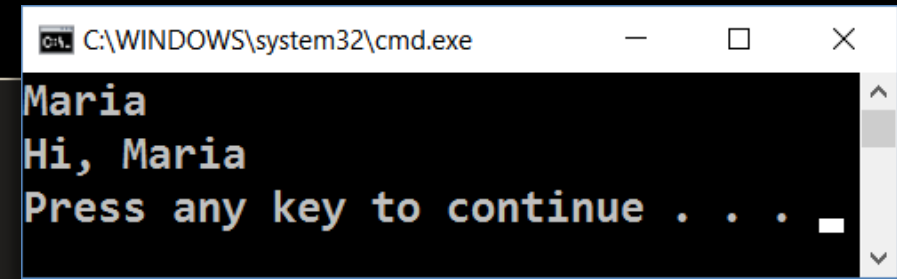
- Инициализация от **string literal**:

```
string str = "Hello, C#";
```

0	1	2	3	4	5	6	7	8
H	e	l	l	o	,		C	#

- Въвеждане на **string** от конзолата:

```
string name = Console.ReadLine();  
Console.WriteLine("Hi, " + name);
```



- Преобразуване на **string** от и до **char array**:

```
string str = new String(new char[] {'s', 't', 'r'});  
char[] charArr = str.ToCharArray(); // ['s', 't', 'r']
```

Сравняване на символни низове

- **Ordinal** (двоично) сравняване на символни низове

```
int eq = (str1 == str2); // ползва String.Equals(...)
```

- **Case-insensitive** сравняване на символни низове

```
int result = string.Compare(str1, str2, true);  
// result == 0 ако str1 съвпада с str2  
// result < 0 ако str1 е преди str2  
// result > 0 ако str1 е след str2
```



- **Case-sensitive** сравняване на символни низове

```
int result = string.Compare(str1, str2, false);
```

Съединяване (комбиниране) на символни низове

- Използване на метода **Concat()**

```
string str = string.Concat(str1, str2);
```

- Използване на **+** или **+=** оператори

```
string str = str1 + str2 + str3;  
string str += str1;
```



- Всеки обект може да бъде добавен към символен низ

```
string name = "Peter"; int age = 22;  
string s = name + " " + age; // → "Peter 22"
```

Търсене в символен низ

- Намиране на подниз в даден символен низ
 - **str.IndexOf(string term)** – връща индекса на първия СИМВОЛ или **-1**

```
string email = "vasko@gmail.org";  
int firstIndex = email.IndexOf("@"); // 5  
int secondIndex = email.IndexOf("a", 2); // 8  
int notFound = email.IndexOf("/"); // -1
```



Започва от индекс 2

- **str.LastIndexOf(string term)** – намира последно срещане на търсения подниз в низа

```
string verse = "To be or not to be...";  
int lastIndex = verse.LastIndexOf("be"); // 16
```


Задача: Брой появявания на подниз в низ

- Даден ви е **текст** и **шаблон**
 - Намерете колко пъти шаблона се среща в текста
 - Припокриване е позволено

ababa caba
aba



3

aaaaaaaa
aa



5

Welcome to SoftUni
Java



0

Решение: Брой появявания на подниз в низ

```
string input = Console.ReadLine().ToLower();  
string pattern = Console.ReadLine().ToLower();  
  
int counter = 0;  
int index = input.IndexOf(pattern);  
while (index != -1)  
{  
    counter++;  
    index = input.IndexOf(pattern, index + 1);  
}  
  
Console.WriteLine(counter);
```

Извличане на подниз

■ `str.Substring(int startIndex, int length)`

```
string filename = @"C:\Pics\Rila2017.jpg";  
string name = filename.Substring(8, 8);  
// name == "Rila2017"
```



■ `str.Substring(int startIndex)`

```
string filename = @"C:\Pics\Rila2017.jpg";  
string nameAndExtension = filename.Substring(8);  
// nameAndExtension == "Rila2017.jpg"
```

0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19
C	:	\	P	i	c	s	\	R	i	l	a	2	0	1	7	.	j	p	g

Разделяне на символни низове

- Разделяне на символен низ по дадени разделители

```
string[] Split(params char[] separator)
```

- Пример:

```
string listOfFruits = "Apple, Banana, Cherry, Lemon.";
string[] fruits = listOfFruits.Split(' ', ',', '.');
Console.WriteLine("Available fruits are:");
foreach (string fruit in fruits)
    Console.WriteLine(fruit);
```



Поставя всеки елемент от fruits
във fruit и изпълнява кода

Какво научихме този час?

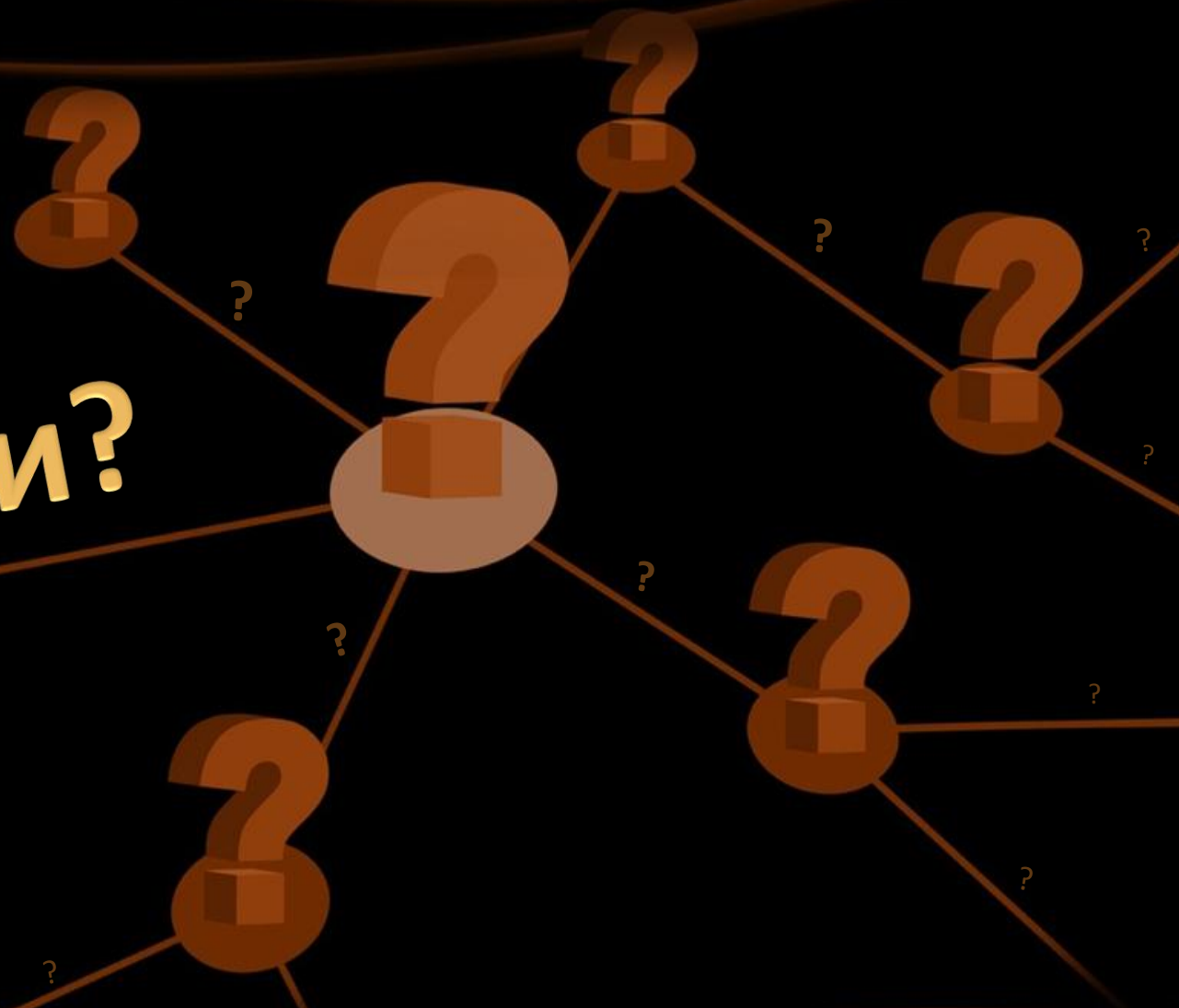
- **Символните низове** са поредица от символи
 - Декларират се с ключовата дума **string**
 - Ограждат се в кавички
 - **Immutable** (read-only) са
- Можем да ги сравняваме, съединяваме
 - Съединяването с “+” е **бавно**, ще разгледаме друг начин в следващите теми
- Търсим подниз в низ с метода **IndexOf**
- Извличаме подниз от низ със **Substring**
- Разделяме низ на поднизове по дадени разделителни символи с метода **Split**



СИМВОЛНИ НИЗОВЕ



Въпроси?



Министерство на образованието и науката (МОН)

- Настоящият курс (презентации, примери, задачи, упражнения и др.) е разработен за нуждите на Национална програма "**Обучение за ИТ кариера**" на МОН за подготовка по професия "Приложен програмист"



Министерство
на образованието
и науката



Национална
програма
„Обучение за
ИТ кариера“

- Курсът е базиран на учебно съдържание и методика, предоставени от **фондация "Софтуерен университет"** и се разпространява под свободен лиценз **CC-BY-NC-SA**



SoftUni
Foundation

