

Стрингове (низове)

Емилиян Кадийски

<https://digitalrazgrad.org>
<https://digitaltargovishte.org>

Планът за днес

1. Особености
2. Инициализация
3. Конкатенация
4. Методи
5. Задачи
6. GitHub

A solid dark blue circle is centered on the page. Inside the circle, the Russian words "ОСОБЕНОСТ" and "И" are written in white, bold, sans-serif capital letters, stacked vertically.

**ОСОБЕНОСТ
И**

Особености

Символните низове (strings, стрингове) са поредица от символи.

В Java за обработка на низове се използва класът String.

```
String course = "Intro to Java";
```

Символните низове се записват като последователност от символи, оградена в кавички.

Кавичките не са част от стойността на низа!



Особености

String е клас, а не прост тип (каквито например са int или boolean).

- Променливите могат да имат стойност null.
- Сравняват се по различен начин от простите типове.
- Класът String има методи за различни действия с низове (разгледайте ги).



ИНЦИАЛИЗА

-

ЦИЯ

Задаване на стойност

Можем да присвоим стойност на променлива от тип String по няколко начина:

```
String name = "Ivan";
```

```
String job = new String("programmer");
```

```
String two = 2 + "";
```

Можем да преобразуваме друг тип в низ, като го съберем с низ.





**КОНКАТЕНА-
ЦИЯ**

Конкатенация (слепване) на стрингове

```
String firstName = "Ivan";
```

```
String lastName = "Petrov";
```

Използване на знак „+“ за слепване на низове:

```
String fullName = firstName + " " + lastName; // Ivan Petrov
```

Използване на метода „concat()“ за слепване на низове:

```
String fullName2 = lastName.concat(", ").concat(firstName); // Petrov, Ivan
```

StringBuilder

При класа **String** поредицата от символи , записана в паметта, не се изменя (нарича се *immutable*). При промяна на променливата, съдържанието не се променя, а се създава ново място в паметта, в което е записана новата стойност.

Затова не се препоръчва долепяне на низове в цикъл!

За работа в низове, когато имаме изменение на стойността на низа, се препоръчва използването на класа `StringBuilder`.

```
StringBuilder builder = new StringBuilder();  
builder.append("Text");  
String result = builder.toString();
```





МЕТОДИ

charAt

- Можем да достъпваме символите на низа по индекси. Индексирането започва от 0, както при масивите:

```
String country = "Programming";
```

```
System.out.println(country.charAt(2)); // o
```

- Можем да вземем дължината на низа:

```
System.out.println(country.length()); // 8
```



indexOf

Методът `indexOf` му се подава текст и знак или дума, която трябва да намери в този текст.

Резултатът е число - индексът на първото срещане на търсеният символ в текста.

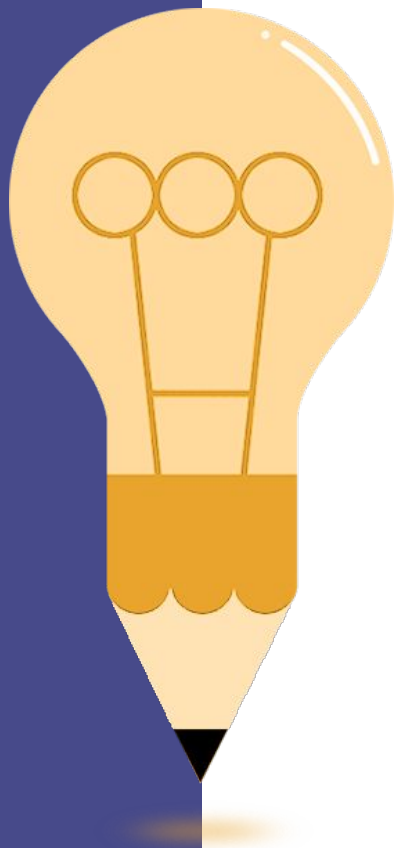
Пример:

```
String text = "This is text. Look at that text. Text text."
```

```
int search = text.indexOf("t"); //search = 8
```



Извод



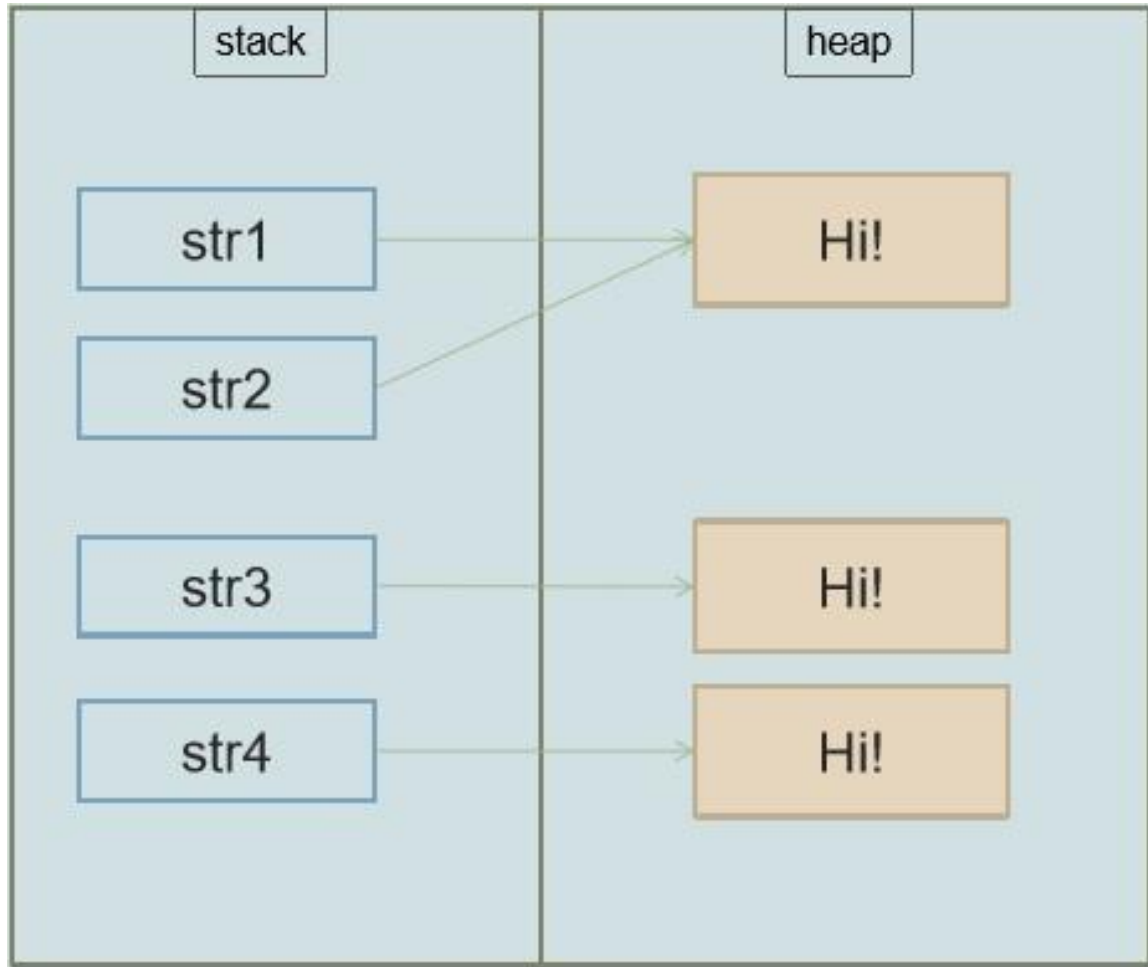
Винаги, когато работите с променливи от тип String, имайте предвид, че може да имат стойност null и винаги правете проверки!

equals

Тъй като String е клас, символните низове в Java не се сравняват с оператора „==“. При използване на „==“, се сравняват техните адреси, а не стойности.

```
String str1 = "Hi!";  
String str2 = "Hi!";  
boolean equal = str1 == str2;  
System.out.println(equal); // true  
  
String str3 = new String("Hi!");  
String str4 = new String("Hi!");  
equal = str3 == str4;  
System.out.println(equal); // false
```

equals



Променливата не запазва в себе си стойността на низа, а сочи към място в паметта, където се пази тази стойност.

Сравнението с „==“ сравнява адресите, а не реалните стойности.

equals

За сравнение на стойностите на низове се използва методът equals()

```
String str3 = new String("Hi!");  
String str4 = new String("Hi!");  
  
equal = str3.equals(str4);  
  
System.out.println(equal); // true
```

equals

Напишете програма, която очаква от потребителя да въведе името на любимия си език за програмиране в конзолата.

Ако въведенният език е Java, извежда „Good choice!“ и „Try again!“ в противен случай.



format

Метод `String.format()` :

Задава се шаблон с места, в които да се попълват данните от различни типове.

Задават се като следващи параметри на метода данните в последователността, в която трябва да бъдат попълнени.

```
String.format("Name: %s, Age: %d years", "Ivan", 25);
```

```
//Name: Ivan, Age: 25 years
```



format

Използвайки `String.format()`, напишете следното (подчертаните думи да се зададат като места за попълване в шаблона).

//Name: Ivan, Age: 25 years, Lives in Vratsa



split

Методът `split` разделя един Стринг на по-малки стрингове, спрямо знак, който му е подаден.

Резултатът се записва в масив от стрингове.

Пример:

```
String text = "This is text. Look at that text. Text text."
```

```
String[] result = text.split("t");
```

```
// result = {"This is ", "ex", ". Look a", " ", "ha", " ", "ex", " ", "ex", "."}
```



replace

Методът `replace` приема три параметъра - текстът върху който работи, думата която ще заменя и думата с която ще бъде заменена.



substring

Методът `substring` реже един стринг по подаден индекс и създава нов, по-къс стринг.



A solid dark blue circle is centered on the page. Inside the circle, the text 'УПРАЖНЕНИ' is written in white, bold, uppercase letters. Below it, the letter 'Е' is also written in white, bold, uppercase letters.

УПРАЖНЕНИ
Е

Примерна задача

Напишете метод, който приема за параметър даден текст и връща масив, съдържащ думите на този текст.

Напр: `wordify("Java is cool") -> ["Java", "is", "cool"]`

Самостоятелни задачи

задачи за упражнение

Проекти - Ниво 1

Задания за проект

The GitHub logo, which consists of a dark blue circle with the word "GitHub" in white text in the center.

GitHub

Git

- Разработена като страничен проект на създателя на Линукс - Линус Торвалдс през 2005-та.
- В нея са решени проблеми, които той и екипът му са срещали при използването на други подобни системи
- Бърза, лесна за използване, напълно дистрибутирана, поддържа огромни проекти
- Позволява не-линейна разработка

<https://git-scm.com/downloads>



Важни команди

- Команди за първоначален setup:

```
git config --global user.email "you@example.com"
```

```
git config --global user.name "Your Name"
```

- Първо качване в GitHub:

```
git init
```

```
git add -A
```

```
git commit -m "commit name"
```

```
git remote add origin <url>(взима се от GitHub)
```

```
git push origin main
```



GitHub, GitLab, BitBucket, други...

- Съществуват множество услуги, които предоставят съхранение на кода посредством Гит
- В същността си са еднакви и няма значение коя точно ползвате



Въпроси?



ДИГИТАЛНО
ОБЩЕСТВО
ТЪРГОВИЩЕ



ДИГИТАЛНО
ОБЩЕСТВО
РАЗГРАД

ДА
ОТВОРИМ
КРЪГА



КЛУБ НА НСО

Trainings @ Digital Razgrad & Digital Targovishte

- Digital Razgrad
 - <https://digitalrazgrad.org>
 - <https://facebook.com/digitalrazgrad.org>
 - digitalrazgrad.slack.com
- Digital Targovishte
 - <https://digitaltargovishte.org>
 - <https://facebook.com/digitaltargovishte.org>
 - digitaltargovishte.slack.com

