



Увод в програмирането с Java



СИМВОЛНИ НИЗОВЕ

Съдържание

- Особености
- Сравняване
- Конкатенация
- StringBuilder
- Форматиране
- Задачи
- Домашно

Особенности

Символни низове в Java

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).

Някои методи

- **charAt(...);**

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

```
String country = "Bulgaria";
```

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

- **length();**

Можем да вземем дължината на низа:

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

Някои методи

- `indexOf(...);`

Пример:

```
String str = "Java Intro Course";  
int index = str.indexOf("Java"); // index = 0
```

- `toLowerCase();` -> всички букви стават малки
- `toUpperCase();` -> всички букви стават главни

Някои методи

- `equals(...);`
- `equalsIgnoreCase(...);`

Пример:

```
String langName = "Java";  
String upperLangName = "JAVA";  
System.out.println(langName .equals(upperLangName )); // false  
System.out.println(langName .equalsIgnoreCase(upperLangName )); // true
```

Някои методи

- **substring(...);**

Пример:

```
String courseTitle = "Java Intro Course";  
String result = courseTitle.substring(5, 10); // result = "Intro"
```

- **replace(...);**

```
String courseTitle = "Java Intro Course";  
String result = courseTitle.replace("Intro Course", "OOP"); // result = "Java OOP"
```

Някои методи

- **split(...);**

Пример:

```
String courses = "Java, PHP, C#";  
String[] coursesArray = courses.split(", "); // coursesArray = {"Java", "PHP", " C#"};
```

- **trim();**

Пример:

```
String courseTitle = " Java Intro Course ";  
String trimTitle = courseTitle.trim(); // trimTitle = "Java Intro Course";
```

Пример

Прочетете името на потребителя от конзолата. Отпечатайте в конзолата първата и последната буква и дължината му.

За четене на символен низ от конзолата:

```
Scanner input = new Scanner(System.in);  
String s = input.nextLine();
```

Задача

а) Напишете метод, който приема като параметри 2 символни низа и връща първата позиция, на която вторият низ се съдържа в първия.

Пример:

`firstPosition("I like Java.", "Java") = 7`

Задача

- b) Подайте null като стойност на първия параметър и вижте какъв е резултатът.
- c) Нека методът да връща -5, ако някоя от стойностите на подадените низове е null.
- d) Нека методът да връща -3, ако първият низ е празен (има дължина 0).

Извод

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

Задаване на стойност

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.

Сравняване на стрингове

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

Примери за сравнение

```
String str1 = "Hi!";
```

```
String str2 = "Hi!";
```

```
boolean equal = str1 == str2;
```

```
System.out.println(equal); // true
```

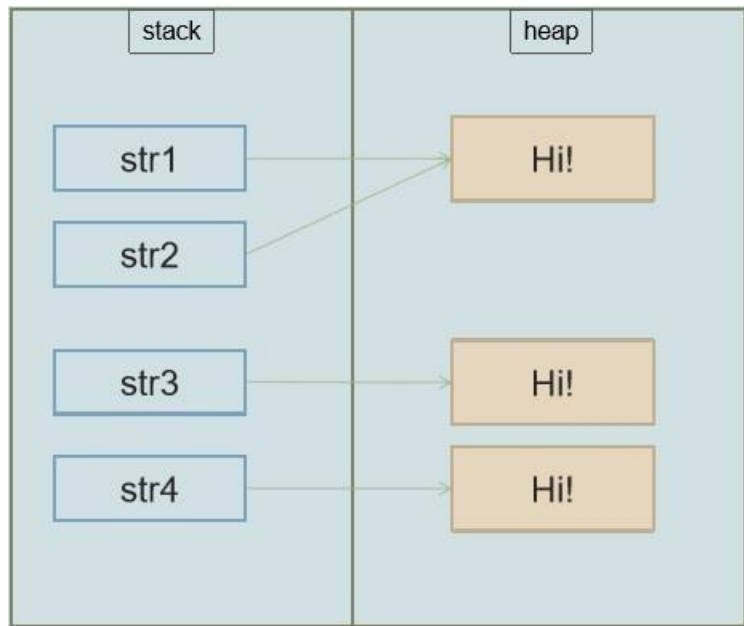
```
String str3 = new String("Hi!");
```

```
String str4 = new String("Hi!");
```

```
boolean equal = str3 == str4;
```

```
System.out.println(equal); // false
```

Представяне в паметта



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.

Правилно сравняване на низове

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
```

Задача

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

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

Слепване (конкатенация) на низове

Слепване (конкатенация) на низове

Пример:

```
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
```


Задача

Да се напише метод, който връща символен низ, съдържащ поредица от числата от 1 до n (n е параметър на метода).

Пример: `formattedNumbers(3)` = "123"

Какво става, когато въведем голямо число, напр. 5000, 20000?

Важно

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!

StringBuilder

StringBuilder

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

Пример

1. Създаваме нова променлива от клас `StringBuilder`:

```
StringBuilder sb = new StringBuilder();
```

```
    sb.append("Numbers: ");
```

```
    for(int i = 1; i <= n; i++) {
```

```
        sb.append(i);
```

```
    }
```

Използваме метод `append()`, за да добавяме

в края на низа:

```
    sb.append(i);
```

```
}
```

3. Използваме метод `toString()`, за да преобразуваме резултата в `String`:

```
return sb.toString();
```

Форматиране на низове

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
```

Задача

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

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

Задачи

Задача

1. Даден е масив, който съдържа адрес за изпращане на писмо. На всяка от позициите има съответно: [име на получател, държава, град, адрес, пощенски код]. Напишете метод, който приема като параметър такъв масив и връща като резултат низ, който съдържа всеки елемент от адреса на отделен ред.

Примерен резултат:

„Иван Иванов
България
София
Ул. Незабравка 12
1000“

Задача

2. Напишете метод, който приема като вход символен низ и заменя всички срещания на "Java" с „PHP“.

Напр: `replace("I study Java. Java is nice.")` -> `"I study PHP. PHP is nice."`

Задача

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

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

Домашно

Задача

1. Даден е масив, който съдържа адрес за изпращане на писмо. На всяка от позициите има съответно: [име на получател, държава, град, адрес, пощенски код]. Напишете метод, който приема като параметър такъв масив и връща низ, съдържащ адреса, форматиран по следния начин:

Име, Държава

Град ПК, Адрес

Пример - резултат:

„Иван Иванов, България

Враца 3000, Ул. Кокиче 14“

Задача

2. Напишете метод, който приема като вход низ и връща низа, изписан наобратно.

Пример: backwards("Chocolate") -> "etaloCohC"

Задача

3. Напишете метод, който приема за вход два низа и извежда колко пъти вторият се среща в първия.

Напр: `wordCount("I live in BG. I am now in Vratsa", "in")` -> 2

Задача

4. Даден е символен низ, съставен от няколко "забранени" думи, разделени със запетая. Даден е и текст, съдържащ тези думи. Да се напише програма, която замества забранените думи в текста със звездички.

Пример:

Microsoft announced its next generation Java compiler today. It uses advanced parser and special optimizer for the Microsoft JVM.

Низ от забранените думи: "Java,JVM,Microsoft".

Резултат:

***** announced its next generation **** compiler today. It uses advanced parser and special optimizer for the ***** **.

Задача

Прочетете главата „Символни низове“ от книгата „Въведение в програмирането с Java“

<https://www.introprogramming.info/intro-java-book/read-online/glava13-simvolni-nizove/>