

Intro to Java

7 - Strings / Exercises: Basics of String methods

Exercises: Basics of String methods

Looking at the Java reference documentation

A reference is a technical documentation that explains all the capabilities available in a class or in a library. This exists also for every Java class.

- Google for `java string reference` to find the reference documentation of the `String` class or visit [the official Java String Documentation](#).
- Try to find a method that does what you need, if it exists.
- Do not use the `SEARCH` box on the website but the search of the browser.
- Read the explanation and how it is used.
- Use this information to do the following exercises.

`length()`, `isEmpty()`, `toUpperCase()`, `toLowerCase()`

Write a program that prints the length of a given string.

Print if the string is empty

Print the same string with all its letters changed to upper case

Print the same string with all its letters changes to lower case

```
Input: Hello World
Output:
Length: 11
Empty? false
HELLO WORLD
hello world
```

`equals(String anotherString)` and `equalsIgnoreCase(String anotherString)`

Write a program that reads two strings from the input and checks if they are equal or only equal with case ignored.

Input:

Wirsing

Wirsing

Output: Equal

Input:

Wirsing

wirsing

Output: Not equal but equal with case ignored

Input:

Wirsing

Vanillepudding

Output: Not equal

contains(String str)

Write a program that reads two strings from the user and outputs if one is a part of the other.

Input:

Vanillepuddingeis

pudding

Output: pudding is a part of Vanillepuddingeis

startsWith(String prefix)

Write a program that reads two strings from the user and outputs if one word is the prefix (first part) of the other.

Input:

Vanillepudding

Vanille

Output: Vanille is a prefix of Vanillepudding

endsWith(String suffix)

Write a program that reads two strings from the user and outputs if one word is the suffix (last part) of the other.

```
Input:
Vanillepudding
pudding
Output: pudding is a suffix of Vanillepudding
```

charAt(int index)

Write a program that prints the index and the character for a given word.

```
Input: Hello
Output:
Index: 0, Character: H
Index: 1, Character: e
Index: 2, Character: l
Index: 3, Character: l
Index: 4, Character: o
```

indexOf(String substring)

Write a program that takes two strings and prints where in the first string the second string starts

```
Input 1: This is a test
Input 2: is
Output: is can be found at index 2
```

indexOf(String substring, int index)

Write a program that takes an int as position and two strings and prints where in the first string the second string starts

```
Input 1: This is a test
Input 2: is
Start Position: 4
Output: is can be found at index 5
```

substring(int beginIndex, int endIndex)

Write a program that reads a string and a number from the input and splits the given string at the index of the number.

Input:

Vanillepuddingeis

7

Output:

Vanille - puddingeis

replace(String target, String replacement)

Use the replace method to decipher this string `lala#lwve#-rwgra22lalang!#<3`

The code is:

Code	Letter
w	o
#	` ` (space)
2	m
lala	i
-	p

compareTo(String anotherString)

Write a program that reads two strings from the user and outputs them in lexicographical order.

Input:

Banane

Ananas

Output: Ananas, Banane

Input:

Vanillepudding

Wirsing

Output: Vanillepudding, Wirsing

Made with ❤ by teachers at [ReDI School](#).