**Operations on Strings in Java (String API[[1]](#footnote-1))**

In Java all Strings are immutable, but we can do some operations on String which „create” a new Strings. There a couple ways to operate on Strings in Java.

Task 1. You have given some String for example „Peter” , please do some operations that produces on output „Peterson”.

|  |  |
| --- | --- |
| **Solution name** | **Code** |
| Strings concatenation | String name=”Peter”;  name+=”son”; |
| Contact method from String API | String name=”Peter”;  name = name.concat("son"); |
| StringBuilder class | String name = "Peter";  StringBuilder strb = new StringBuilder(name);  strb.append("son");  String result = strb.toString(); |
| StringBuffer class | String name = "Peter";  StringBuffer stb = new StringBuffer(name);  stb.append("son");  String result = stb.toString(); |

**StringBuffer and StringBuilder are exact the same class with the same API, but StringBuffer is thread safe class. Remember that.**

Some java.lang.String API operations and some descriptions.

**char charAt(int index)**

Returns the char value at the specified index. An index ranges from 0 to length() - 1. The first char value of the sequence is at index 0, the next at index 1, and so on, as for array indexing.

**boolean endsWith(String suffix)**

Tests if this string ends with the specified suffix.

**boolean startsWith(String suffix)**

Tests if this string ends with the specified suffix.

**boolean** **equals(String string)** - Compares this string to the specified object. The result is true if and only if the argument is not null and is a String object that represents the same sequence of characters as this object.

**boolean** **equalsIgnoreCase(String string)** - Compares this String to another String, ignoring case considerations. Two strings are considered equal ignoring case if they are of the same

length and corresponding characters in the two strings are equal ignoring case.

**int length() -** Returns the length of this string. The length is equal to the number of Unicode code units in the string.

There are more methods in String API, like compareTo, codePointAt, substring, toLoweCase, toUpperCase, trim and more.   
  
Try to see in documentation what they do and how works. You can check it using your IDE or in the Internet on Oracle docs (enter in the google sentence „Java Sring API”).

**Excercises**:

In every excercise you will be given a String **„InfoGeolog at Lodz University”**, to check the output (result) just use System.out.println(); to display the result in console.  
**Ex. 1**

Write the code which displays a char at index 4.

**Ex. 2**

Write the code which will produce whole string written in uppercase.

**Ex. 3**

Try to replace all `o` letters in caption on `a`, you may use method replaceAll, or replace.  
Please notice, that method replace is operating on chars (single sign) and you must use ‘ ‘ (single apostroph) to declare a char.

**Ex. 4**

If you know the loops (for example for, do….while, while….do) try to write the code which will append 3 times (using StringBuffer/StringBuilder) letter „T” at the end of your String.  
  
On the input we will given string „**InfoGeolog at Lodz University”** at the output we should be like „**InfoGeolog at Lodz UniversityTTT”.**

**Ex. 5**

Given a String for example „ABCD” try to find a way how to reverse this string, the output should be like „DCBA”.

**Solutions**

**Exercise 1**

String name = "InfoGeolog at Lodz University";

char charSign = name.charAt(4);

System.out.println(charSign);

**Exercise 2**

String name = "InfoGeolog at Lodz University";

name = name.toUpperCase();

System.out.println(name);

**Exercise 2 (second manner)**

String name = "InfoGeolog at Lodz University".toUpperCase();

System.out.println(name);

**Exercise 3**

String name = "InfoGeolog at Lodz University";

name = name.replaceAll("o", "a");

System.out.println(name);

**Exercise 4**

String name = "InfoGeolog at Lodz University";

StringBuffer sbf = new StringBuffer(name);

for (int i = 0; i < 3; i++) {

sbf.append("T");

}

String result = sbf.toString();

System.out.println(result);

**Exercise 5**

String name = "ABCD";

StringBuffer stb = new StringBuffer(name);

stb.reverse();

String result = stb.toString();

System.out.println(result);

1. API – its shortcut of Application Programming Interface – some ready functionality like methods and operations in library, which are ready to use without more coding. [↑](#footnote-ref-1)