

# HO GENT

OOSDII

*Strings en reguliere expressies*

# Table of Contents

|                                       |   |
|---------------------------------------|---|
| 1. Oefening Strings .....             | 1 |
| 2. Oefening reguliere expressies..... | 3 |

# 1. Oefening Strings

- Bepaal voor elke oefening de uitvoer.

Maak gebruik van de cursus en de API-documentatie om de correcte uitvoer te bepalen.

```
// ---
System.out.println("\n0ef1");
char[] helloArray = { 'h', 'e', 'l', 'l', 'o', '.' };
String helloString = new String(helloArray);
System.out.println("helloString = " + helloString);

// ---
System.out.println("\n0ef2");
String greeting = "Hello!";
int l = greeting.length();
System.out.println("Lengte is : " + l);

// --
System.out.println("\n0ef3");
if (greeting.equals("Hi"))
    System.out.println("Informal Greeting");
else
    System.out.println("Greeting");

String s1 = "mary!";
System.out.println(s1.equalsIgnoreCase("Mary!"));

s1 = "mary!";
System.out.println(s1.equals("Mary!"));

// ---
System.out.println("\n0ef4");
String greeting2 = "Hi Mary!";
greeting2 = greeting2.toUpperCase();
System.out.println(greeting2);

// ---
System.out.println("\n0ef5");
String str = "Strings are immutable";
String s = str.concat(" all the time");
System.out.println(str);
System.out.println(s);

// ---
System.out.println("\n0ef6");
String pause = "    Hmm    ";
String newS = pause.trim();
System.out.println(newS);
System.out.println(pause);
```

```

// ---
System.out.println("\n0ef7");
String greeting3 = "Hello!";
char k = greeting.charAt(0);
char j = greeting.charAt(1);
System.out.println(k);
System.out.println(j);

// ---
System.out.println("\n0ef8");
String str1 = new String("Welcome to Java");
char[] str2 = new char[7];
str1.getChars(2, 9, str2, 0);
System.out.println("Gekopieerd: " + Arrays.toString(str2));

// ---
System.out.println("\n0ef9");
String sample = "AbcdefG";
String sub = sample.substring(2);
System.out.println(sub);

String sub2 = sample.substring(2, 5);
System.out.println(sub2);

// ---
System.out.println("\n0ef10");
String greeting4 = "Hi Mary!";
l = greeting4.indexOf("Mary");
int l2 = greeting4.indexOf("Sally");
System.out.println("l = " + l);
System.out.println("l2 = " + l2);

String name = "Mary, Mary quite contrary";
System.out.println(name.indexOf("Mary", 1));
System.out.println(name.indexOf("Mary", 0));
System.out.println(name.indexOf("Mary", 8));

name = "Mary, Mary, Mary quite so";
System.out.println(name.lastIndexOf("Mary"));

// ---
System.out.println("\n0ef11");
String entry = "adventure";
int r = entry.compareTo("zoo");
System.out.println(r);
r = entry.compareTo("adventure");
System.out.println(r);
r = entry.compareTo("above");
System.out.println(r);

// ---

```

```

System.out.println("\n0ef12");
System.out.println(name.regionMatches(1, entry, 0, 2));

// ---
System.out.println("\n0ef13");
System.out.println(entry.startsWith("ad"));
System.out.println(entry.startsWith("ven", 2));

// ---
System.out.println("\n0ef14");
char[] ar = entry.toCharArray();
System.out.println(ar[2]);

// ---
System.out.println("\n0ef15");
s1 = "Hello";
System.out.println(s1 == "Hello");

s1 = new String("Hello");
System.out.println(s1 == "Hello");

s1 = "Hello";
System.out.println(s1.equals("Hello"));

s1 = new String("Hello");
System.out.println(s1.equals("Hello"));

// ---
System.out.println("\n0ef16");
double d = 102939939.939;
boolean b = true;
long lo = 1232874;
char[] arr = { 'a', 'b', 'c', 'd', 'e', 'f', 'g' };

System.out.println( String.valueOf(d));
System.out.println(String.valueOf(b));
System.out.println(String.valueOf(lo));
System.out.println( String.valueOf(arr));

```

## 2. Oefening reguliere expressies

Gegeven:

Een aantal Strings en een reguliere expressie

Gevraagd:

Welke Strings matchen met de reguliere expressie?

Vb String tekst = "abababa";

boolean match = tekst.matches("a(ab)\*a");

1 **a(ab)\*a**

- 1) abababa
- 2) aaba
- 3) aabbaa
- 4) aba
- 5) aabababa

2 **ab+c?**

- 1) abc
- 2) ac
- 3) abbb
- 4) bbc

3 **a.[bc]+**

- 1) abc
- 2) abbbbbbbb
- 3) azc
- 4) abcbcbcbc
- 5) ac
- 6) asccbbbbcbbccc

4 **abc|xyz**

- 1) abc
- 2) xyz
- 3) abc|xyz

5 **[a-z]+[.!]**

- 1) battle!
- 2) Hot
- 3) green
- 4) swamping.
- 5) jump up.
- 6) undulate?
- 7) is.?

6 **[a-zA-Z]\*[^\s]=**

- 1) Butt=
- 2) BotHEr,=
- 3) Ample
- 4) FIIdIE 7h=
- 5) Brittle =
- 6) Other.=

7 **[a-z][.!]\s+[A-Z]**

- 1) A. B
- 2) c! d
- 3) e f
- 4) g. H
- 5) i? J
- 6) k L

8 **(very )+(fat )?(tall|ugly) man**

- 1) very fat man
- 2) fat tall man
- 3) very very fat ugly man
- 4) very very very tall man

9 **<[^>]+>**

- 1) <an xml tag>
- 2) <opentag> <closetag>
- 3) </closetag>
- 4) <>
- 5) <with attribute="77">