



#2.en - String, StringBuilder, equality, and regular expressions

Exercise 1 - Assignment, equality, and references

1.

```
var s = "toto";
System.out.println(s.length());
```

What is the type of 's'? How does the compiler know that there is a `length()` method in 's'?

2. What does the following code display?

```
var s1 = "toto";
var s2 = s1;
var s3 = new String(s1);

System.out.println(s1 == s2);
System.out.println(s1 == s3);
```

Explain.

3. What is the method to use if you want to test if the content of the strings is the same?

```
var s4 = "toto";
var s5 = new String(s4);

System.out.println(/* compare the content of s4 and s5 */);
```

4. What does the following code display?

```
var s6 = "toto";
var s7 = "toto";

System.out.println(s6 == s7);
```

Explain.

5. Explain why is it important that `java.lang.String` is not mutable.
6. What does the following code display?

```
var s8 = "hello";
s8.toUpperCase();
System.out.println(s8);
```

Explain.

Exercise 2 - In Morse code. Stop.

Write a class `Morse` that displays the character strings taken in argument separated by "Stop. " .

```
$ java Morse this is funny
this Stop. is Stop. funny Stop.
```

1. First use the operator `+` which allows the concatenation of strings.

2. What the class `java.lang.StringBuilder` stands for?
Why does its `append (String)` method return an object of type `StringBuilder` ?
3. Rewrite the `Morse` class using a `StringBuilder` .
What is the advantage compared to the previous solution?
4. Copy the following code into a `Test` class:

```
public static void main(String[] args) {  
    var first = args[0];  
    var second = args[1];  
    var last = args[2];  
    System.out.println(first + ' ' + second + ' ' + last);  
}
```

why can we use a `' '` instead of `" "` ?

Compile this code then use the command `javap` to display *the Java bytecode* (which is not in assembler) generated.

```
javap -c Test
```

What can you deduce from it?

5. Compile the code of question 1, then use the command `javap` to display *the Java bytecode* generated.
What can you deduce from it?
When should you use `StringBuilder.append ()` rather than `+` ?
And why is the practical work teacher going to grumble if I write a `+` in a call to the `append` method?

Exercise 3 - Pattern Matching

The purpose of this exercise is to manipulate regular expressions in java. We will use the classes from the `java.util.regex` package for this.

1. What is the use of the class `java.util.regex.Pattern` and its method `compile` ?
What is the class `java.util.regex.Matcher` for?
2. Write a program that reads strings on the command line and displays strings that correspond to numbers (namely strings with all characters between '0' and '9').
3. Modify the program so that, if a character string begins with characters which are not numbers, these numbers are also displayed (for example, with the strings "abc", "ab3", "4de", "f6h" and "789", we display: "3 789").
4. Write a method which takes as a parameter a character string containing an IPv4 address and returns an array of 4 bytes. You must test that it is indeed a valid address.
You will use the notion of group for this.