Intro to Java

7 - Strings / Exercises: Strings

Exercises: Strings

Looking at the Java reference documentation

A reference is a technical documentation that explain 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.

Counting characters

Write a program that counts how many ‘e’s and how many ‘u’s are in the following text:

```
Als Gregor Samsa eines Morgens aus unruhigen Träumen erwachte,

fand er sich in seinem Bett zu einem ungeheueren Ungeziefer verwandelt.
```

To check your solution:

```
There are 23 'e's and 7 'u's.
```

NOTE: uppercase and lowercase characters are different characters.

As a bonus, modify the program to print how many times each letter appears in the text.

For example given:

```
mississippi
```

The output would be:

```
m occurring 1 times

i occurring 4 times

s occurring 4 times

p occurring 2 times
```

Think: How to show the occurrence of each letter only once?

Reverse

Write a method that reverses a string.

Examples:

- `java` -> `avaj`
- `Hello world` —> `dlrow olleH`
- `ReDI is a great school!` -> `!loohcs taerg a si IDeR`

HINT: create an empty string, fill it and return it.

Switch Parts

Write a method that switches the parts of a given string in the middle.

Examples:

- `java` -> `vaja`
- `Hello world` —> `world Hello`
- `ReDI is a great school!` -> `reat school!ReDI is a g`

Palindrome

Write a method which checks if a word is a palindrome.

A palindrome is a word which you can read from both sides (from left to right or from right to left) and you will get the same word.

Examples are: `Ebbe`, `Anna`, `Kajak` or `Gag`.

NOTE: the uppercase and lowercase characters should be considered as the same character, otherwise the method doesn’t work with words starting uppercase.

Secret messages

We need a program to decode secret messages. Our messages are kept secret by a simple substitution cipher. A substitution cipher is when we change certain letters with other letters. The known recorded use of a substitution cipher was as early as the Roman Empire.

For example if our substitution cipher replaces each letter of the alphabet with the letter that follows that letter in the alphabet a message like:

```
HELLOWORLD
```

Would be encoded in our secret code as:

```
IFMMPXPSME
```

Because `I` follows `H`, `F` follows `E` and so on. In a special case `Z` is followed by `A`.

Our program will deal with messages written in capital letters of the English alphabets (A through Z) and words without spaces and punctuation marks. So if the user provides `SFEJKBWB` as input, the output should be `REDIJAVA`.

HINT:

- Remember for loops
- Remember methods on String like length and charAt
- Remember that we can concatenate String with String and how will String win(?get?) when we try to concatenate String with non-String (alternatively:
- Remember that we can concatenate strings with strings, how will be the output when we try to concatenate a string with a non-string

As a bonus:

- Research ASCII table
- Try to figure out why `'I'- 1` gives us `'H'`

Made with ❤ by teachers at ReDI School.