Exercises Module 9 - Strings

Task 1 - Palindrome

Create a method isPalindrome that returns true if a string is a palindrome, otherwise false. You can solve it with the previous implementation of reverse for an array, or with built-in methods in Java.

For more information about palindrom: **https://en.wikipedia.org/wiki/Palindrome (https://en.wikipedia.org/wiki/Palindrome)**

Task 2 - Divide the string

We will use Java's built-in split method: **https://www.javatpoint.com/java-string-split (https://www.javatpoint.com/java-string-split)**

(follow the link but skip "Internal implementation" and just read the code examples on how it is used)

See the following string:

```
String str = "Hello and welcome. Nice to meet you tomorrow.";
```

a. Pick out each word on a separate line.

```
Result:

Hello
and
welcome.
Nice
to
meet
you
tomorrow.
```

b. Use the same string as before, but pick out every sentence on a separate line:

Result:

```
Hello and welcome.
Nice to meet you tomorrow.
```

Tip!

Tip!

- To split on "." you have to tell the split method that "." should not be interpreted as a regex expression, so you must protect "." by typing "\\."
- To remove space in the beginning and end of a string, use the trim method.

c.

See the following string:

```
String mixed = "It will be 127 computers and 15 screens";
```

Write a program that first only prints the numbers on one line, and then only prints the words on one line.

Result:

```
127 15
It will be computers and screens
```

Solve this task by using the method *matches*. The *matches* method uses regex patterns to validate if a string matches a desired pattern. This technique is useful for small string matches, but it can get complicated to match on multiple patterns at the same time!

Read about regex here: **http://www.ocpsoft.org/opensource/guide-to-regular-expressions-in-java-part-1/ (http://www.ocpsoft.org/opensource/guide-to-regular-expressions-in-java-part-1/)**

Task 3 - Numbers triangle

Create a program that prints a number triangle with a certain number of lines like this (ask the user to enter the number of lines):

```
Please enter the number of lines of the triangle: 6
1
22
333
4444
55555
666666
```

- Solve it with a nested for loop, ie a for loop inside another for loop.

Task 4 - Validate a registration plate

In this task you will create a program that checks if a registration plate is in the correct format

In this task you will create a program that checks if a registration plate is in the correct format.

A Swedish registration plate contains three letters at the beginning and three digits at the end, for example ABC-123, in that form (this is changing in 2019, but in this exercise we will stick to the traditional format).

- *Create a class called NumberPlateValidation*
- Create a Scanner object so that the user can enter a registration plate
- Create the method *isRegistrationNumberValid* that returns true or false.
- The format of the registration plate is XXX-YYY, where "X" is a capital letter from A to Z and "Y" is a number from 0 to 9. **(https://sv.wikipedia.org/wiki/Registreringsskyltar_i_Sverige#Ol%C3%A4mpliga_bokstavskombinationer)**

Tip!

- Use the built-in method *split* for strings to split the stringt. What character should you use to split the string into the letter part and the numerical part?
- Structure the code into smaller help methods that validate specific rules and returns true or false.

Task 5 - Generate HTML with StringBuilder

Create a program that generates the following HTML code:

```
<table>
    <tr style='background-color:green;color:white;'>
        <td>Row 1: Donald</td>
    </tr>
    <tr style='background-color:yellow;color:black;'>
        <td>Row 2: Daisy</td>
    </tr>
    <tr style='background-color:green;color:white;'>
        <td>Row 3: Huey</td>
    </tr>
    <tr style='background-color:yellow;color:black;'>
        <td>Row 4: Dewey</td>
    </tr>
    <tr style='background-color:green;color:white;'>
        <td>Row 5: Louie</td>
    </tr>
    <tr style='background-color:yellow;color:black;'>
        <td>Row 6: Scrooge</td>
    </tr>
    <tr style='background-color:green;color:white;'>
        <td>Row 7: Ludvig</td>
    </tr>
</table>
```

The names used for the list are:

```
"Donald", "Daisy", "Huey", "Dewey", "Louie", "Scrooge", "Ludvig"
```

To make it easier to read the lines in the web page, inline css has been added so that the font color and background color changes for each row.

Use the same indentation as in the example.

Use StringBuilder appropriately to print the same HTML code on the screen.

Just print the raw html on the screen, no need to create a web browser that parses the html! ;)