

Denne oppgaven teller ikke mot sluttkarakter.

Mandatory Assignment 2

Forfall 4. okt. 2019 av 23.59 **Poeng** 0 **Må leveres** en nettsteds-URL

Introduction

Please read the page called [Mandatory Assignments - Information](#) before continuing this assignment.

The goal of the assignment is to test whether you have advanced your skills so that we can proceed with more complex Javascript programming assignments. This includes:

- use of operators
- use of flow control statements
- using arrays
- defining functions

You will see that we have included bonus sub-tasks and a bonus task. Although these are not mandatory, they are recommended.

Task 1 - User-defined List

In this task, you will create a website that allows users to define and manage their own lists, for example a shopping list.

Minimum recommended HTML elements: A button, an input field, and two paragraph elements. The input field should be used so that users can add elements to their list. The first paragraph should display all current list elements and the second paragraph information about the array and some fun stuff, if you would like to.

The script should implement the following functionality:

- a) When the button is pressed the text in the input field should be parsed and appended to an Array
- b) The content of the first paragraph will then be updated according to the content of the Array
- c) The content of the second paragraph should also be updated to show information about the list: the first element, the last element, and the number of elements in the Array. You can also add other information if you would like, e.g. the middle element.

The display of array elements in the paragraph could be done as a simple comma separated String.

Bonus tasks:

- d) Separate printed elements by something different than ";" for example; a new line, or " * "*
 - e) Print each array element on a new line next to a number that corresponds to the index of the element in the array*
 - f) Make an undo button, that deletes the last element in your array.*
 - g) Create an input field with type="number" and a new button. When this button is pressed you delete the Array element at the index of the number entered in the input field and warn the user if no such element exists.*
-

Task 2 - Password Checker

In this task, you will create a website that allows users to enter their username and password. You should save the user-entered login and password in variables (subtask a) and then you should perform necessary checks whose complexity is increasing (subtask b, c, ...). The resulting website should perform all checks and display which check fails, if any!

Recommended HTML elements: A button, two input fields, and possibly a paragraph. You can use the paragraph or console.log or an alert box to display which checks fail and welcome the user if all checks are passed.

- a)** When the button is pressed the value of the two input fields is saved as two different variables.
- b)** Username and password should be different.
- c)** Password must be 8 characters or more.
- d)** Password must be 32 characters or less.
- e)** If all checks succeed you should welcome the user.

Bonus tasks:

- f) Password must contain at least one upper and one lowercase letter.*
 - g) Password must contain at least one special character.*
 - h) Password must contain a number*
-

Task 3 - Mathematics

In this task, you will write a function that calculates and returns the area of a circle with two decimal points precision. The functions should have a circle radius parameter and return the circle area. The function should check if the function argument is a positive integer. If not, the function should return undefined.

Circle area is calculated using $A = \pi \cdot r^2$, where r is the circle radius.

The function should then called on the values demonstrated below:

```
console.log( calcCircleAreaByRadius(1) ); // 3.14
console.log( calcCircleAreaByRadius(8) ); // 201.06
console.log( calcCircleAreaByRadius("fifteen") ); // undefined
console.log( calcCircleAreaByRadius(-4) ); // undefined
```

(Hint: `Math.PI`, `Math.pow()`)

Task 4 - Logical Gate

In this task, you have to write at least two functions that implement the boolean XOR operation. The truth table of the XOR operation is shown below:

Exclusive-OR gate



A	B	Output
0	0	0
0	1	1
1	0	1
1	1	0

There are many ways to solve this task and your should implement at least two in two different functions. Both should take two boolean input parameters and return one boolean value. One of the functions should implement the boolean expression below:

$a \oplus b = (a \vee b) \wedge \neg(a \wedge b)$, where \oplus is the *XOR* operator, \wedge is the *AND* operator (`&&` in Javascript), \vee is the *OR* operator (`||` in Javascript), and \neg the *Not* operator (`!` in Javascript)

You are free to decide how to implement the other (use ifs, the ternary operator, or whatever will work)

Task 5 - Console Drawing

In this task, you will use different programming techniques to perform console drawing. Console drawing is the process of creating a drawing in the console by using one (or more) characters (see ASCII art). For example, the following is a drawing of a chessboard:

```

#       #       #       #
  #     #     #     #     #
#       #       #       #
  #     #     #     #     #
#       #       #       #
  #     #     #     #     #
#       #       #       #
  #     #     #     #     #
#       #       #       #
  #     #     #     #     #

```

For those who enjoy having a specific goal your task will be to draw a Christmas tree.

For the rest of you: Draw whatever pattern you would like.

Your script or function should allow you to change which character you will use to draw. You may want to structure your work using the following sub-tasks:

- a) Decide how tall (number of rows) and how wide (number of characters per row) the tree should be.
- b) Implement an outer loop so that you can draw each row
- c) Create the content of each row. Use conditional statements or an inner loop as you see fit.

This task is demanding to solve in a way where it looks "good". Do not let that demotivate you! The important thing about this task is to try different things in loops, so you can better understand how a loop functions in runtime. (This task is also super practice for nested loops). All kinds of trees/drawing that are drawn computationally will be accepted.

WARNING: Simply printing (`Console.log`'ing) fixed hard-coded template literals will not be accepted. If you do this you are gambling that the rest of the assignment is sufficient enough for an approved grade.

```

\  *
 ***
*****

```

*****`

BONUS: Task 6 - Prime Numbers

NB, this task is not mandatory and is just meant for extra training. Everyone is however strongly encouraged to do this task as well.

Create a function that takes an integer as a parameter and returns true or false based on if the number is a prime number. the functions should return false if the input is anything other than a positive integer.

The function should be called using the values like demonstrated below:

```
console.log(isPrime(-13)); // false
console.log(isPrime(1)); // false
console.log(isPrime(2)); // true
console.log(isPrime("sixtyfour")); // false
console.log(isPrime(2473)); // true
console.log(isPrime(2475)); // false
console.log(isPrime(3301)); // true
```

(Hint: DuckDuckGo.com)
