Workshop 6 - opdrachten functions

In deze week leer je werken met functions.

Extra lessen (tip)

Ging het college te snel voor je volg dan de volgende les op Udacity.

Introduction to JavaScript - <https://classroom.udacity.com/courses/ud803>

Lesson 5 - functions.

# Opdrachten

Opdrachten om zelf uit te voeren. (weten hoe / tonen)

* [~~https://learn.freecodecamp.org/javascript-algorithms-and-data-structures/basic-javascript/write-reusable-javascript-with-functions~~](https://learn.freecodecamp.org/javascript-algorithms-and-data-structures/basic-javascript/write-reusable-javascript-with-functions)
* [~~https://learn.freecodecamp.org/javascript-algorithms-and-data-structures/basic-javascript/passing-values-to-functions-with-arguments~~](https://learn.freecodecamp.org/javascript-algorithms-and-data-structures/basic-javascript/passing-values-to-functions-with-arguments)
* [~~https://learn.freecodecamp.org/javascript-algorithms-and-data-structures/basic-javascript/global-scope-and-functions~~](https://learn.freecodecamp.org/javascript-algorithms-and-data-structures/basic-javascript/global-scope-and-functions)
* [~~https://learn.freecodecamp.org/javascript-algorithms-and-data-structures/basic-javascript/local-scope-and-functions~~](https://learn.freecodecamp.org/javascript-algorithms-and-data-structures/basic-javascript/local-scope-and-functions)
* [~~https://learn.freecodecamp.org/javascript-algorithms-and-data-structures/basic-javascript/global-vs--local-scope-in-functions~~](https://learn.freecodecamp.org/javascript-algorithms-and-data-structures/basic-javascript/global-vs--local-scope-in-functions)
* [~~https://learn.freecodecamp.org/javascript-algorithms-and-data-structures/basic-javascript/return-a-value-from-a-function-with-return~~](https://learn.freecodecamp.org/javascript-algorithms-and-data-structures/basic-javascript/return-a-value-from-a-function-with-return)
* [~~https://learn.freecodecamp.org/javascript-algorithms-and-data-structures/basic-javascript/understanding-undefined-value-returned-from-a-function~~](https://learn.freecodecamp.org/javascript-algorithms-and-data-structures/basic-javascript/understanding-undefined-value-returned-from-a-function)
* ~~<https://learn.freecodecamp.org/javascript-algorithms-and-data-structures/basic-javascript/assignment-with-a-returned-value>~~

Mocht je nu nog meer willen oefenen dan kan dat op: <https://learn.freecodecamp.org/>

# Eindopdrachten.

Wanneer je de workshop hebt gevolgd en genoeg opdrachten hebt gemaakt moet je de volgende opdrachten kunnen afronden.

1. ~~Zet de uitwerking van de Fibonacci getallenreeks uit de vorige les om in een functie.~~
2. ~~Schrijf een functie countdown die van 10 - 1 aftelt (dit wordt ook in de console getoond). Geef aan de functie een jaartal parameter mee die als hij bij ‘0’ is zegt ‘Happy {jaartal}’.~~
3. ~~Schrijf een functie expressie en declaratie waarmee je aantoont dat een expressie en een declaratie op een andere manier met hoisting omgaan. (als je deze vraag niet snapt dan moet je de videoreeks uit udacity kijken. Daar wordt hoisting goed uitgelegd).~~
4. [facultatief; alleen voor studenten die extra verdieping zoeken] Installeer TypeScript (https://www.typescriptlang.org/) en maak opdracht 1, 2 en 3 met behulp van Typescript. Let er op dat je nu ook voor alle variabelen een type moet declareren.

Inleveren

Deadline is donderdag 09.00 uur. Inleveren via learn.hz.nl

Belangrijk is dat de code van je opdracht op github staat.

Daarnaast lever je een **pdf**  in met daarin ingevuld beoordelingsformulier (zie bijlage 1).

# Bijlage 1 - Beoordelingsformulier.

Naam: Roosmarijn Pluijgers

Github repository url:[**https://github.com/Roos-Skywalker/Programming-basics/tree/master/24-9-2019**](https://github.com/Roos-Skywalker/Programming-basics/tree/master/24-9-2019)

Dit is een subset van de beoordelingscriteria waarop je aan het eind van de cursus wordt beoordeeld. Mocht je nu een onderdeel niet snappen dan mag je dat overslaan.

## 1. Eigen feedback na evaluatie van je code.

|  |  |  |
| --- | --- | --- |
| **Nr** | **Leeropbrengsten.** | **Aanwezig** |
| 1 | Gebruikt consistente naamgeving | Ja |
| 2 | Gebruikt const, let, var in de gevraagde situatie | Ja |
| 3 | Gebruikt array's om data in op te slaan. | Nee |
| 4 | Gebruikt een conditional (if else) | Nee |
| 5 | Gebruikt een loop om herhaling van stappen te voorkomen. | Ja |
| 6 | Schrijft een functie om code te structureren | Ja |
| 7 | Schrijft een functie om herhaling te voorkomen | Ja |
| 8 | Schrijft commentaar volgens de JS docs richtlijnen | Ja |
| 9 | Gebruikt de juiste operatoren bij vergelijkingen | Ja |
| 10 | Gebruikt console.log om te debuggen. | Ja |

## 2. Feedback een van een klasgenoot na evaluatie van je code.

Naam feedbackgever: Neo McNeese

|  |  |  |
| --- | --- | --- |
| **Nr** | **Leeropbrengsten.** | **Aanwezig** |
| 1 | Gebruikt consistente naamgeving | Ja |
| 2 | Gebruikt const, let, var in de gevraagde situatie | Ja |
| 3 | Gebruikt array's om data in op te slaan. | Nee |
| 4 | Gebruikt een conditional (if else) | Nee |
| 5 | Gebruikt een loop om herhaling van stappen te voorkomen. | Ja |
| 6 | Schrijft een functie om code te structureren | Ja |
| 7 | Schrijft een functie om herhaling te voorkomen | Ja |
| 8 | Schrijft commentaar volgens de JS docs richtlijnen | Ja |
| 9 | Gebruikt de juiste operatoren bij vergelijkingen | Ja |
| 10 | Gebruikt console.log om te debuggen. | Ja |