

2 : JavaScript voor developers in opleiding

Alles op Gitlab of Github!

Zorg er voor dat alle opdrachten in gitlab of github worden gezet. Dit mag gewoon een eigen publieke repository zijn, of vraag aan je collega's om een repository aan te maken waarmee je kan werken.

Sources for Javascript

In programming, the search engine is your best friend for filling knowledge gaps and learning from other programmers. Best practices are always important. Note that not all information is a best practice. Lots of code snippets are experimental and not all self titled experts are real experts. Always verify information and look for multiple examples if you need inspiration. Furthermore, there are always interesting websites for different programming languages. Make sure that searching with Google, for example, is well controlled.

Google search explanations:

<https://support.google.com/websearch/answer/134479?hl=en>

<https://google.com>

Javascript manuals

<https://developer.mozilla.org/en-US/docs/Web/JavaScript/Guide>

Books for junior developers

<https://pepa.holla.cz/wp-content/uploads/2015/11/JavaScript-For-Dummies-4th-Edition.pdf>

Opdracht 2 : De som van A & B

Opdracht 2.1

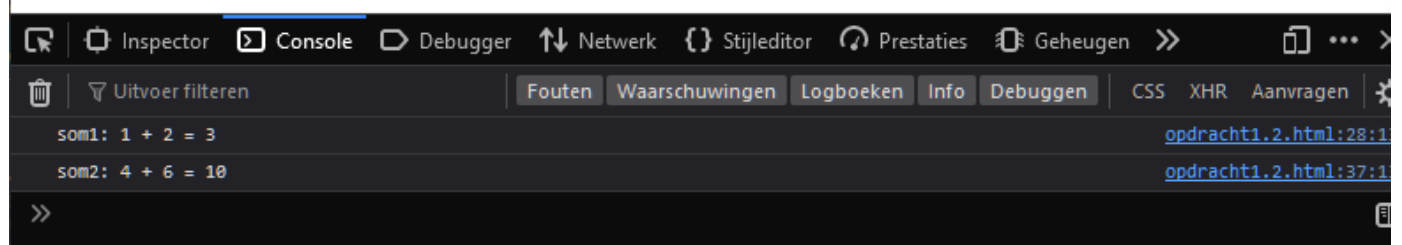
Bekijk onderstaande code, uitgaande van het feit dat de functie som is geïmplementeerd, en voorspel de werking en voorzie de code van duidelijk en commentaar die de functionaliteit beschrijft (dus niet een vertaling is van Javascript -> nederlands/engels).

```
//.....

var a = 1;
var b = 2;
□
var som1 = new som(a, b);
som1.calculate(function(antwoord)
{
□console.log("som1: " + som1.a + " + " + som1.b + " = " + antwoord);
});
□
var a = '4';
var b = "6";
□
var som2 = new som(a, b);
som2.calculate(function(antwoord)
{
□console.log("som2: " + som2.a + " + " + som2.b + " = " + antwoord);
});
```

Opdracht 2.2

Implementeer het object som, zodat deze op identieke manier als hierboven te zien is, gebruikt kan worden. Zorg voor een geldig HTML5 document en voorzie de code van duidelijk commentaar die de functionaliteit beschrijft (dus niet een vertaling is van JavaScript -> nederlands/engels). Het resultaat moet er als volgt uit zien:



Opdracht 2.3

Vul nu een array met 10 sommen en zorg er voor dat middels de ontwikkelde functie "antwoord" per som brekend wordt en dat het resultaat van de array getoond wordt middels `console.log(opgaves)`. Voer de opdracht een keer uit met een for loop, while loop en met een foreach voor de iteratie van variabele opgaves. Zorg voor een geldig HTML5 document en voorzie de code van duidelijk commentaar die de functionaliteit beschrijft (dus niet een vertaling is van php -> nederlands/engels).

```
//.....

var opgaves = [];
for(var i=0; i<3; i++)
{
  opgaves.push({
    a: Math.round(Math.random() * 10),
    b: Math.round(Math.random() * 10),
    antwoord: null
  });
}

// schrijf hieronder de code die nodig is om de in de array
// staande berekeningen uit te voeren en middels console.log()
// te presenteren.
//
..... oplossing hier .....
```

verwachte output:



```
Array(3) [ {}, {}, {} ]
  0: Object { a: 6, b: 7, antwoord: 13 }
    a: 6
    antwoord: 13
    b: 7
    <prototype>: Object { _ }
  1: Object { a: 7, b: 3, antwoord: 10 }
    a: 7
    antwoord: 10
    b: 3
    <prototype>: Object { _ }
  2: Object { a: 4, b: 5, antwoord: 9 }
    a: 4
    antwoord: 9
    b: 5
    <prototype>: Object { _ }
  length: 3
  <prototype>: Array []
```

Opdracht 2.4

Bespreek de opdracht met je collega's.

Opdracht 2.5

Pas je implementatie van de methode *som* zo aan dat het aantal getallen dat opgeteld kan worden flexibel is. Zie onderstaande implementatie voorbeeld. Zorg voor een geldig HTML5 document en voorzie de code altijd van duidelijk commentaar die de functionaliteit beschrijft (dus niet een vertaling is van JavaScript -> nederlands/engels).

```
//.....
var som1 = new som(1, 3, 6, 10);
som1.calculate(function(antwoord)
{
    console.log("som1 = " + antwoord);
});

var som2 = new som(3,5,4,4,6,2,3,34,54,2,23,54,5);
som2.calculate(function(antwoord)
{
    console.log("som2 = " + antwoord);
});
3
var som3 = new som(3,5);
som3.calculate(function(antwoord)
{
    console.log("som3 = " + antwoord);
});
```

resultaat:



Opdracht 2.6 (Bonus):

Gebruik dezelfde code als uit opdracht 1.3 (*het maakt hierbij niet uit of er gebruik wordt gemaakt van de for loop, while loop of de foreach*). Zorg er voor dat het antwoord van de som pas in de console wordt getoond, nadat het antwoord in seconden verstreken is. Bij bijv. $a = 3$ en $b = 5$, dient het antwoord pas na 8 seconden getoond te worden.

De antwoorden dienen in volgorde in de console terecht te komen, dus het antwoord van bijv. $a = 2$ en $b = 1$ dient na 3 seconden te worden getoond, 5 seconden daarna het antwoord van $a = 3$ en $b = 5$ etc.

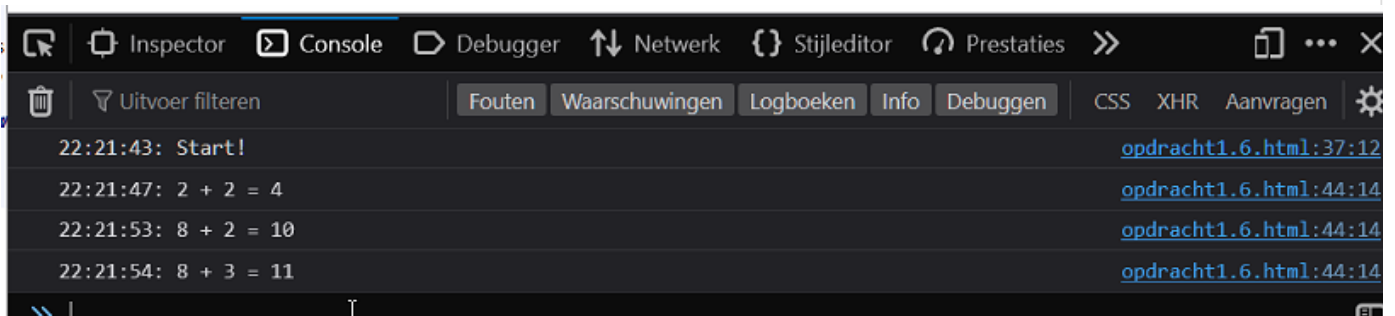
```
//.....

var opgaves = [];
for(var i=0; i<3; i++)
{
  opgaves.push({
    a: Math.round(Math.random() * 10),
    b: Math.round(Math.random() * 10),
    antwoord: null
  });
}

console.log(new Date().toLocaleTimeString() + ": Start!");

// schrijf hieronder de code die nodig is om de in de array
// staande berekeningen uit te voeren en na het verlopen aantal
// seconden (op basis van het antwoord) deze pas in de console
// te tonen.
//
..... oplossing hier .....
```

resultaat:



```
22:21:43: Start!
22:21:47: 2 + 2 = 4
22:21:53: 8 + 2 = 10
22:21:54: 8 + 3 = 11
```

Revision #18

Created 2024-02-14 17:15:59 UTC by Stefan

Updated 2024-06-24 14:08:26 UTC by ADMIN Concera