

# Uitwerkingen opdrachten Javascript

© 2018, 5HART-IT Opleidingen BV

Versie 1.0, Februari 2018



## Opdrachten bij Basisbegrippen

Vraag met `window.prompt` om twee getallen en tel deze bij elkaar op. Toon de som op het scherm.

tip: maak gebruik van `parseFloat(...)` om de invoer naar getallen om te zetten.

```
var a = window.prompt("Eerste getal:");
var b = window.prompt("Tweede getal:");
a = parseFloat(a);
b = parseFloat(b);
console.log(`${a} + ${b} = ${a+b}`);
```

Probeer de volgende oefeningen uit:

Vul een variabele `x` met een getal;

Verhoog de waarde van `x` met 1;

Toon `x` in een boodschap, bijvoorbeeld:

De waarde van `x` is nu: 6

Kunnen het verhogen en tonen ook in één keer?

Wijzig `x`: tel er 1 bij op als het getal kleiner is dan 10, maak 'm anders gelijk aan 10

Maak een array met namen

Toon de eerste naam

```
var x = 8;
x++;
console.log("De waarde van x is nu: " + x);
// of in één keer:
// console.log("De waarde van x is nu: " + ++x);

x = x<10 ? x+1 : 10;
console.log(x); // 9
var namen = ["Kwik", "Kwek", "Kwak"];
console.log(namen[0]);
```

## Opdrachten bij Programmastructuren

Gegeven is de volgende array:

```
var prijzen = [10, 3.5, 2.7, 8];
```

Bereken de totale prijs en toon deze op de console.

Gebruik eerst de for loop.

Doe daarna nog eens met de for ... of loop.

```
var prijzen = [10, 3.5, 2.7, 8];
var som = 0;
// gewone loop:
for (var i=0;i<prijzen.length;i++){
    som+=prijzen[i];
}
console.log("Totaal met for loop: ", som);
// reset
som=0;
// for ... of loop
for (let prijs of prijzen){
    som+=prijs;
}
console.log("Totaal met for ... of loop: ", som);
```

De volgende code geeft een geheel getal tussen 1 en 10:

```
var getal = Math.floor(Math.random()*10)+1;
```

Maak een programma die met

```
var poging = parseInt(window.prompt("geef een getal: "))
```

blijft vragen om het getal te raden, totdat het getal geraden is.

Toon "hoger" als de poging te laag was,

"lager" als de poging te hoog was,

en anders "getal geraden in ... pogingen".

Gebruik een while loop met if statements.

```
var getal = Math.floor(Math.random()*10)+1;
var geraden = false;
var boodschap = 'Welk getal tussen 1 en 10 heb ik in mijn hoofd?';
var pogingen = 0;
while(geraden == false){
    var poging = parseInt(prompt(boodschap));
    pogingen++;
    if (poging < getal) {
        boodschap = 'Probeer hoger: ';
    } else if (poging > getal){
        boodschap = 'Probeer lager: ';
    } else {
        geraden = true;
    }
}
window.alert(`Je hebt ${getal} geraden in ${pogingen} pogingen!`);
```

## Opdrachten bij Functies

Maak een functie `maal(a, b)` die `a` en `b` met elkaar vermenigvuldigt.  
Zorg ervoor dat het ook werkt als `b` niet is opgegeven (welke waarde krijgt `b` als `b` niet wordt opgegeven?)

Optioneel: maak het ook mogelijk om meer waarden op te geven, bijvoorbeeld:

```
var result = maal(3,4,5);  
console.log(result); // 60
```

```
function maal(a, b){  
  a = a||0;  
  b = b||1;  
  var result = a*b;  
  for(var i=2;i<arguments.length;i++){  
    result*=arguments[i];  
  }  
  return result;  
}  
console.log(maal());           // 0  
console.log(maal(2));          // 2  
console.log(maal(2,3));        // 6  
console.log(maal(2,3,4));      // 24
```

```
/*Gegeven is deze array:  
var getallen = [1,3,5,7,9];  
Doorloop deze array met getallen.forEach(...) en toon elk getal  
verdubbeld. Doe dit met (naar keuze):  
een zelf gedefinieerde functie  
een anonieme functie  
een arrow functie*/
```

```
var getallen = [1,3,5,7,9];  
function toonDubbel(getal){  
  console.log(getal*2);  
}  
getallen.forEach(toonDubbel);  
getallen.forEach(function(getal){  
  console.log(getal*2);  
});  
getallen.forEach(x=>console.log(x*2));
```

Maak een functie maal(a, b) die het volgende teruggeeft:

maal(3,4) : 12

maal(3,'a'): aaa

maal('test', 2): testtesttest

maal('a', 'b'): foutmelding: één van de argumenten moet een getal zijn.

Maak hierbij gebruik van typeof en throw. Voor de herhaling van tekst kan repeat() van String worden gebruikt (zie documentatie op MDN).

```
function maal(a, b){
  if(typeof a === 'number' && typeof b === 'number'){
    return a * b;
  }
  if(typeof a === 'string' && typeof b === 'number'){
    return a.repeat(b);
  }
  if(typeof a === 'number' && typeof b === 'string'){
    return b.repeat(a);
  }
  throw 'geef minstens één getal op';
}

try {
  console.log(maal(3,4));
  console.log(maal('a',3));
  console.log(maal('test', 2));
  console.log(maal('a', 'b'));
} catch(fout){
  console.log(fout);
}
```

Maak een IIFE die een functie oplevert die een totaal bijhoudt.

var totaal = (...){};

console.log(totaal(3)); // 3

console.log(totaal(1)); // 4

console.log(totaal(2)); // 6

```
var totaal = (
  function(){
    var som = 0;
    return function(getal){
      return som+=getal;
    };
  })();
console.log(totaal(3)); // 3
console.log(totaal(1)); // 4
console.log(totaal(2)); // 6*
```

## Opdrachten bij Objecten

**Maak een object met willekeurige eigenschappen**, waaronder:

een string  
een array  
een functie

Probeer vervolgens de eigenschappen te gebruiken / tonen / aan te passen.

```
// opdracht 1 objecten
// maak een object met een string, array en functie

let getallen = {
  title: "priemgetallen",
  values: [],
  add: function(...val){
    this.values.splice(this.values.length,0,...val);
  },
  show: function(){
    this.values.forEach(x => console.log(x));
  }
}

getallen.add(2,3,5,7,9,11);
getallen.show()
```

**Maak een array met vier namen.**

Verwissel de eerste twee namen van plek met behulp van array methods.

Vervang de derde naam met behulp van een array method.

Toon alle namen aan elkaar, door van de array een string te maken.

```
const namen = ["jan", "pier", "joris", "corneel"];

// verwissel de eerste twee namen
naam = namen.shift();
console.log(naam);
namen.splice(1,0,naam);
console.log(namen);

// vervang de derde naam
namen.splice(2,1,"henk");
console.log(namen);

// toon alle namen aan elkaar:
console.log(namen.join(""));
```

**Maak een class Boek waarmee boek-objecten kunnen worden aangemaakt.**

Elk boek-object heeft een auteur en een titel.

Geef ook een functie toString(): deze geeft een boek als tekst terug.

Maak vervolgens een array van boeken.

Probeer daarin diverse functies van een array uit, zoals:

- toevoegen aan begin of eind van de array
- alle boeken tonen met forEach
- sorteren op titel
- zoeken op auteur

Zie documentatie bij Array

```

/* jshint esversion:6 */
class Boek {
  constructor(auteur, titel){
    this.auteur = auteur;
    this.titel = titel;
  }
  toString(){
    return this.auteur + ": " + this.titel;
  }
}
var boeken = [];
boeken.push(new Boek("Jan", "Het leven van Jan"));
boeken.unshift(new Boek("Piet", "Het leven van Piet"));
boeken.splice(1,0,new Boek("Klaas", "Het leven van Klaas"));
boeken.forEach(b=>console.log(b.toString()));
boeken.sort((a,b)=>a.titel<b.titel?-1:a.titel>b.titel?1:0);
console.log(boeken.map(b=>b.auteur).join());
boeken.filter(b=>b.auteur=='Piet').forEach(b=>console.log(b.toString()));

```

### **Maak een class Datum.**

Datum heeft een constructor waaraan dag, maand en jaar in cijfers worden doorgegeven:  
 this.dag = new Date(...,...);

Het prototype van Datum krijgt een Array maanden:  
 ["januari", "februari", "maart", ...];

Geef de class Datum ook de functie toString(): deze geeft de dag als string terug,  
 bijvoorbeeld:  
 4 februari 2018.

Tip: gebruik this.datum.getMonth() en this.maanden om de juiste maand op te halen.

```

/* jshint esversion: 6 */

class Datum {
  constructor(dag, maand, jaar){
    this.dag = new Date(jaar, maand-1, dag);
  }
  toString(){
    return this.dag.getDate() + " " +
    this.maanden[this.dag.getMonth()] + " " + this.dag.getFullYear();
  }
}
Datum.prototype.maanden = ["Januari", "Februari", "Maart", "April",
"Mei", "Juni",
                        "Juli", "Augustus", "September", "October",
"November", "December"];

var dag = new Datum(5,2,2018);
console.log(dag.toString());

```



## Opdrachten DOM en events

Maak een div met tekst, en een knop.

Als op de knop wordt geklikt verdwijnt de tekst.

Met nog een druk op de knop wordt de tekst weer getoond.

Maak gebruik van de CSS eigenschap visibility

HTML:

```
<div id="tekst" class="zichtbaar">
Dit is de tekst.
</div>
<button id="knop">
verberg
</button>
```

CSS:

```
.zichtbaar{visibility:visible}
.onzichtbaar{visibility:hidden}
```

Javascript:

```
var knop = document.getElementById("knop");
knop.onclick = function(){
    var tekst = document.getElementById("tekst");
    if(tekst.classList.contains("zichtbaar")){
        tekst.classList.replace("zichtbaar", "onzichtbaar");
        knop.textContent="toon";
    } else {
        tekst.classList.replace("onzichtbaar", "zichtbaar");
        knop.textContent="verberg";
    }
};
```

Maak een input en een output element in HTML:  
<input id="invoer" type="text" placeholder="invoer">  
<br><output id="uitvoer"></output>

Als het input element is gewijzigd (het "change" event gaat af),  
toon dan de waarde van het element in het uitvoer element.

(De tekstwaarde van het input element zit in het value attribuut: invoer.value.  
De tekstwaarde van het output element zit in de uitvoer.textContent).

```
var invoer = document.getElementById("invoer");
var uitvoer = document.querySelector("#uitvoer");
invoer.addEventListener("change", function(){
    var tekst = invoer.value;
    uitvoer.textContent=tekst.toUpperCase();
    invoer.value="";
});
```

Maak een array met artikelen (alleen de namen).  
Genereer hieruit een lijst op het scherm (li-elementen binnen een leeg ul-element).  
Probeer dit eerst met een String template, zoals eerder voorbeeld. Probeer het ook met  
createElement en appendChild.

Optioneel: geef elk element ook een data attribuut nr, gevuld met de index van elk element  
uit de array.

Optioneel: sorteer de array voordat deze op het scherm wordt gezet.

**In HTML:**

```
<ul id="lijst">

</ul>
```

**In Javascript:**

```
var artikelen = ["pen", "potlood", "gum"];
var lijst = document.getElementById("lijst");
artikelen.sort();

lijst.innerHTML = artikelen.map((naam, index)=><li data-
nr="${index}">${naam}</li>`).join("");

function maakItem(naam, index){
    var li = document.createElement("li");
    li.textContent=naam;
    li.setAttribute("data-nr", index);
    return li;
}
var mijnLijst = document.createElement("ul");
var index=0;
for(var artikel of artikelen){
    var li = maakItem(artikel, index++);
    mijnLijst.appendChild(li);
}
lijst.parentNode.replaceChild(mijnLijst,lijst);
```

Maak een object met:  
een array namen  
een functie add(...) om een naam toe te voegen  
een functie toList() die de array sorteert,  
en daaruit LI-elementen maakt (zie eerdere opdracht).

Test het object eerst met console.log.

```
var object = {  
  data : ["Jan", "Pier", "Joris", "Corneel"],  
  add: function (naam) {  
    this.data.push (naam);  
  },  
  toList: function () {  
    this.data.sort (  
      (a,b)=>a.toUpperCase ()<b.toUpperCase ()?-1:  
      a.toUpperCase ()>b.toUpperCase ()?1:0);  
    var result = this.data.map ( (naam)=>  
      `<li>${naam}</li>` ).join ("");  
    return result;  
  }  
};
```

Maak in HTML een input veld en een leeg UL element aan.  
Als het change event op het input veld afgaat,  
gebruik dan add() om de waarde toe te voegen aan de array,  
en toList() om de innerHTML van het UL element te vervangen.

**In HTML:**

```
<label for="">voeg naam toe</label>  
<input type="text" id="namen"/>  
<ul></ul>
```

**In Javascript:**

```
/*jshint esversion:6 */  
  
var ul = document.getElementsByTagName ("ul") [0];  
ul.innerHTML=object.toList ();  
  
var namen = document.getElementById ("namen");  
namen.addEventListener ("change", function () {  
  object.add (namen.value);  
  namen.value="";  
  ul.innerHTML=object.toList ();  
}, false);
```

Maak een webpagina die vraagt om twee getallen (twee inputvelden) en die daarna de som ervan op het scherm toont.  
Controleer of elk veld numeriek is. Geef anders een duidelijke foutmelding.

```
var getal1 = parseFloat(...);  
if(isNaN(getal1)){  
  ....  
}
```

#### **In HTML:**

```
<input id="getal1"> + <input id="getal2"> = <output  
id="som"></output>
```

#### **In Javascript:**

```
let getal1 = document.getElementById("getal1");  
let getal2 = document.getElementById("getal2");  
let som = document.getElementById("som");  
let x = 0;  
let y = 0;  
  
function verwerkGetal(e) {  
  let id = e.target.id;  
  let getal = parseInt(e.target.value);  
  if(id=="getal1"){  
    x = getal;  
  } else if(id=="getal2") {  
    y = getal;  
  }  
  som.textContent=`${x+y}`;  
}  
  
getal1.addEventListener("change", verwerkGetal);  
getal2.addEventListener("change", verwerkGetal);
```

**Casus:** Maak een applicatie met:

- een class Artikel met een naam, een prijs en een aantal. De naam en de prijs worden aan de constructor meegegeven. Artikel heeft ook een functie totaal(): die geeft de prijs \* aantal terug.
- Een lijst artikelen (array).
- De HTML pagina heeft een select item, waarin alle artikelen als option elementen vermeld staan (via Javascript).
- Als een gebruiker een artikel selecteert, wordt deze toegevoegd aan het "boodschappenmandje": een rij van een tabel, waarin naam, aantal, prijs en prijs \* aantal vermeld staan.
- Maak gebruik van een module object (var module = { ... } ) waarin
- de bestellingen als array worden bijgehouden,
- met een add(artikel) functie die een bestelling toevoegt, of als het artikel al besteld is, het aantal ophooft.
- een remove(index) functie wordt aangeroepen als wordt dubbelgeklikt op een rij. Dan wordt het aantal met één verminderd, of als het aantal op 1 staat wordt de bestelling verwijderd uit de array. (gebruik slice() met index)
- Een sum() functie berekent het totaal van alle bestellingen.
- Een toRows() functie maakt table rows (tr en td elementen) van alle artikelen die in bestelling zijn. Deze wordt na elke wijziging (dubbel klik of gekozen artikel) gebruikt om de innerHTML van de table aan te passen.
- Optioneel: geef ook een toJSON functie aan het object: deze geeft de bestellingen als JSON string terug. Gebruik console.log om deze te testen.

HTML:

```
<select id="artikelList"></select>
<br><br>
<table id="bestelling"></table>
<button id="export">
export
</button>
```

CSS:

```
table{
  border-collapse:collapse;
}

tr#totaal {
  border-top:solid;
  border-width:1px 0;
}
```

Javascript:

```
*jshint esversion:6 */

console.clear();

class Artikel {
  constructor(naam, prijs){
    this.naam=naam;
    this.prijs=prijs;
    this.aantal=1;
  }
  totaal(){
    return this.prijs * this.aantal;
  }
}
```

```

var artikelen = [new Artikel("pen", 1.5),
new Artikel("potlood",0.5), new Artikel("gum", 0.8)];

var artikelList = document.getElementById("artikelList");
var options = artikelen
    .map((a,i)=>`<option value="${i}">${a.naam} (${a.prijs})</option>`)
    .join("");
artikelList.innerHTML = "<option value='-1'>maak keuze</option>" +
options;

var module = {
    data : [],
    add: function(artikel){
        var index = this.data.indexOf(artikel);
        if(index>=0) this.data[index].aantal++;
        else this.data.push(artikel);
    },
    remove: function(i){
        if(this.data[i].aantal>1) this.data[i].aantal--;
        else this.data.splice(i,1);
    },
    toRows: function(){
        return this.data
            .map((a,i)=>
`<tr data-nr='${i}'><td>${a.naam}</td><td>${a.aantal} *</td>`
+`<td>${a.prijs} =</td><td>${a.totaal()}</td></tr>`)
            .join("");
    },
    sum: function(){
        return this.data
            .map(a=>a.totaal())
            .reduce((a,b)=>a+b,0);
    },
    toJSON: function(){
        return JSON.stringify(this.data);
    }
};

var table = document.getElementsByTagName("table")[0];

function addRemoveListeners(){
var tr = document.getElementsByTagName("tr");
tr = Array.from(tr);
tr.forEach(
    row=>row.addEventListener("dblclick",function(){
        module.remove(row.dataset.nr);
        reset();
    }));
}

function resetOptions(){
    document.querySelector("#artikelList option[value='-1']").selected="selected";
}

function reset(){
    table.innerHTML=module.toRows();
    if(module.data.length>0) table.innerHTML +=
`<tr id="totaal"><td>Totaal:</td><td> </td><td>
</td><td>${module.sum()}</td></tr>`;
}

```

```
    addRemoveListeners();  
    resetOptions();  
}  
  
reset();  
artikelList.addEventListener("input", function(e) {  
    module.add(artikelen[e.target.value]);  
    reset();  
}, false);  
  
document.getElementById("export").onclick=()=>  
console.log(module.toJSON());
```

Versie geheel met class:

```
/*jshint esversion:6 */
```

```
console.clear();
```

```
let Artikel = (function(tableId, selectId){
  let bestellingen = [];
  let artikelen = [];
  return class {
    constructor(naam, prijs){
      this.naam = naam;
      this.prijs = prijs;
      this.aantal = 1;
    }
    add(){
      let index = bestellingen.indexOf(this);
      if(index>=0){
        bestellingen[index].aantal++;
      } else {
        bestellingen.push(this);
      }
    }
    remove(){
      let index = bestellingen.indexOf(this);
      if(index>=0){
        if(bestellingen[index].aantal>1){
          bestellingen[index].aantal--;
        } else {
          bestellingen.splice(index,1);
        }
      }
    }
    totaal(){
      return this.prijs * this.aantal;
    }
    static get bestellingen(){
      return JSON.stringify(bestellingen);
    }
    toRow(){
      var that = this;
      var row = document.createElement("tr");
      var td1 = document.createElement("td");
      var td2 = td1.cloneNode();
      var td3 = td1.cloneNode();
      var td4 = td1.cloneNode();
      var naam = document.createTextNode(this.naam + ": ");
      var aantal = document.createTextNode(this.aantal + " * ");
      var prijs = document.createTextNode(this.prijs + " = ");
      var totaal = document.createTextNode(this.totaal());
      td1.appendChild(naam);
      td2.appendChild(aantal);
      td3.appendChild(prijs);
      td4.appendChild(totaal);
      row.appendChild(td1);
      row.appendChild(td2);
      row.appendChild(td3);
      row.appendChild(td4);
      row.addEventListener("dblclick", function(){
        that.remove();
        Artikel.resetTable(tableId);
      });
    }
  };
})(tableId, selectId);
```



```

    });
    return row;
}
static toRows(){
    var rows = bestellingen.map(a => a.toRow());
    var row = document.createElement("tr");
    row.setAttribute("id", "totaal");
    var td1 = document.createElement("td");
    var td2 = td1.cloneNode();
    var td3 = td1.cloneNode();
    var td4 = td1.cloneNode();
    var naam = document.createTextNode("Totaal:");
    var aantal = document.createTextNode(" ");
    var prijs = document.createTextNode(" ");
    var totaal = document.createTextNode(Artikel.sum());
    td1.appendChild(naam);
    td2.appendChild(aantal);
    td3.appendChild(prijs);
    td4.appendChild(totaal);
    row.appendChild(td1);
    row.appendChild(td2);
    row.appendChild(td3);
    row.appendChild(td4);
    rows.push(row);
    return rows;
}
static resetTable(){
    var table = document.getElementById(tableId);
    table.innerHTML = "";
    Artikel.toRows().forEach(row => table.appendChild(row));
}
static resetOptions(){
    var artikelList = document.getElementById(selectId);
    artikelList.innerHTML="";
    Artikel.toOptions().forEach(o=>artikelList.appendChild(o));
    document.querySelector(`#${selectId}
option[value='0']`).selected="selected";
    artikelList.addEventListener("input", function(event){
        var artikel = artikelen.find(x => x.naam ==
event.target.value);
        artikel.add();
        Artikel.resetOptions2();
        Artikel.resetTable();
    });
}

static resetOptions2(){
    document.querySelector(`#${selectId}
option[value='0']`).selected="selected";
}
static sum(){
    return bestellingen.map(a => a.totaal())
        .reduce((a,b)=> a + b,0);
}
static addArtikel(a){
    artikelen.push(a);
}
toOption(){
    //var that = this;
    var option = document.createElement("option");
    option.setAttribute("value", this.naam);

```

```

        var content = document.createTextNode(`${this.naam}
        (${this.prijs})`);
        option.appendChild(content);
        return option;
    }
    static toOptions() {
        var options = artikelen.map(a => a.toOption());
        var option = document.createElement("option");
        option.setAttribute("value", 0);
        var content = document.createTextNode(`Maak een keuze`);
        option.appendChild(content);
        options.unshift(option);
        return options;
    }
};

})( "bestelling", "artikelList");

Artikel.addArtikel(new Artikel("pen", 1.5));
Artikel.addArtikel(new Artikel("potlood",0.5));
Artikel.addArtikel(new Artikel("gum", 0.8));
Artikel.resetOptions();
document.getElementById("export").addEventListener("click",
function() {
    console.log(Artikel.bestellingen);
});

```