In deze paragraaf ga je meer van de basis van JS leren. JS kan namelijk veel meer dan alleen met de DOM werken, al gaan we dit in deze paragraaf wel verder behandelen.

Er zijn namelijk nog een paar handige manieren om de DOM te gebruiken in JS. Je kunt namelijk vanaf een element verder door de DOM heen ‘reizen’. Je kunt daarvoor gebruikmaken van parents en children. Parents zijn elementen die ‘hoger’ liggen. Zo kan de parent van een <p> een <body> zijn. Children zijn elementen die ‘lager’ liggen, dus eigenlijk het omgedraaide van parents.

Stel je wilt de parent van een element met als id ‘getParent’ krijgen. Dit kan met

#CODE

document.getElementById(‘getParent’).parentElement

Een voorbeeld hiervan is:

HTML

#CODE

<html>

<head>

</head>

<body>

<header>

</header>

<div>

<p id='element'>

Test

</p>

</div>

<footer>

</footer>

<script src="js.js"></script>

</body>

</html>

JS

x = document.getElementById('element').parentElement;

x.style.backgroundColor = 'green';

x.style.width = '500px';

x.style.height = '500px';

De JS code zal de <div> met daarin een <p> met als id ‘element’ vierkant maken met een groene kleur.

Om de children van een element te krijgen gebruik je:

document.getElementById(‘getParent’).children

Om het aantal children te krijgen gebruik je #CODE .length Dit werkt hetzelfde voor alle arrays, NAAM.length zal je het aantal elementen terug geven. Variabelen komen later in deze paragraaf nog aan bod.

Bijvoorbeeld document.getElementById(‘getParent’).children.length

Om een bepaald child te krijgen bij meerdere children kun je .children[INDEX] gebruiken en door de children ‘heen bladeren’.

JS is erg veelzijdig en werkt in veel opzichten hetzelfde als andere programmeertalen zoals Python. Het heeft dus ook variabelen, functies, loops en overige.

Variabelen in JS bestaan in verschillende soort, zoals: objecten, integers, arrays en strings. Een variabele dat een object opslaat is bijvoorbeeld de ‘x’ in het voorbeeld en verwijst naar een element van de DOM. Integers zijn gehele getallen en werken net zoals in Python, er zijn ook floats oftewel kommagetallen. Arrays zijn eigenlijk de lists die je kent van Python, oftewel variabelen die meerdere waardes kunnen vasthouden. In JS kan een array ook meerdere type variabelen bevatten. Strings bestaan ook en houden tekst vast.

Om een variabele ergens aan gelijk te stellen gebruik je #CODE var VARIABELE\_NAAM = WAARDE; Je kunt ook #CODE VARIABELE\_NAAM = WAARDE; gebruiken, maar dit is minder duidelijk.

Een paar voorbeelden:

#CODE

var X = document.getElementById('element').parentElement;

var INT = 5;

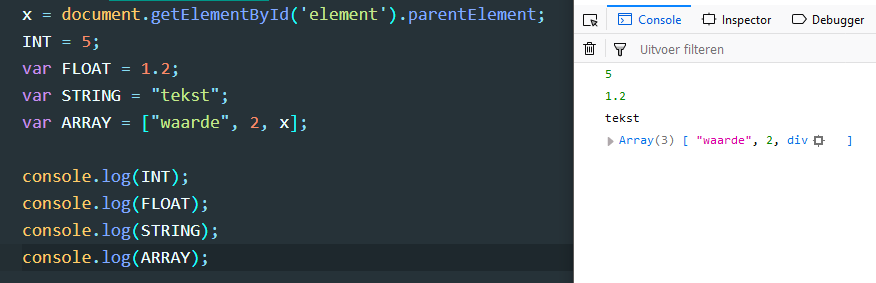
var FLOAT = 1.2;

var STRING = “tekst”

var ARRAY = [“waarde”, 2, X];

Om een variabele te bekijken kun je gebruik maken van alert() of console.log(), bij console.log() kun je meer soorten data zien en is de data zichtbaar in de console. Je moet dus eerst de developer modus starten om hiermee data te kunnen zien. Bij alert() zie je de data meteen als pop-up.

Als je de boven beschreven variabelen via console.log() zichtbaar zou maken krijg je:



Je kunt natuurlijk ook rekenen met de getallen. Dit gaat zoals je gewend bent Python. Voorbeelden zijn:

#CODE

A = 5

B = 10

C = A+B

C = A-B

C = A/B

C = A\*B

Het is ook mogelijk om tekst ‘op te tellen’ zoals je bij getallen zou doen.

#VRAGEN

1. Hoe krijg je het eerste child van een element met meerdere children?
2. Hoe krijg je het aantal children van een element?
3. Kun je in een JS array meerdere type variabelen onderbrengen?
4. Kun je tekst bij elkaar ‘optellen’ in JS?

#ANTWOORDEN

1. .children[0]
2. .children.length
3. Ja, zo kan er bijvoorbeeld een int, string en element in dezelfde array voorkomen.
4. Ja