Front End 4 - DOM Manipulation

Inleiding

Neem eerste de theorie door op de deze pagina!

De opdrachten van dit blok sluiten op elkaar aan. Je moet ze dus in de jusite volgorde maken.

> Waarschuwing: als je de opdrachten met ChatGPT maakt en je snapt niet wat je doet dan loop je met de laatste opdrachten vast. Deze opdrachtnen zijn getest en kunnen niet met behulp van ChatGPT 4.0 integraal worden opgelost.

Kom je ergens niet verder, ga dan ***niet*** verder met de volgende opdracht maar vraag hulp!

Opdracht 1, kleur aanpassen (via id)

Je hebt de volgende HTML/JS-code.

```
<!DOCTYPE html>
<html lang="nl">
<head>
    <meta charset="UTF-8">
    <title>Tekstkleur Veranderen</title>
</head>
<body>
    <p id="tekst">Dit is een voorbeeldtekst.</p>
    <button onclick="veranderKleur()">Verander Kleur</button>

    <script>
        function veranderKleur() {
            // maak de code af en plaats hieronder de code die de kleur van de tekst verandert
            ...
        }
    </script>
</body>
</html>
```

De JavaScript-functie wordt aangeroepen vanuit de `onclick` attribuut van de HTML-knop.

Wanneer de knop wordt ingedrukt, moet de kleur van de tekst in het paragraafelement veranderen. Maak hiervoor het script af.

Zorg dat je in het script zoekt naar het element met het id "tekst". Als je het element hebt gevonden dan kan je de daarna de kleur aanpassen.

Inleveren

Zet jouw naam als commentaar in de JS-code en lever het geteste en werkende HTML-pagina in.

Opdracht 2, kleur aanpassen (via class)

Je hebt de volgende HTML/JS-code:

```
<!DOCTYPE html>
<html lang="nl">
<head>
    <meta charset="UTF-8">
    <title>Achtergrondkleur Veranderen</title>
</head>
<body>
    <p class="kleurVerander">Dit is een voorbeeldtekst 1.</p>
    <p class="kleurVerander">Dit is een voorbeeldtekst 2.</p>
    <p class="kleurVeranderNiet">Dit is een voorbeeldtekst 3.</p>
    <p class="kleurVerander">Dit is een voorbeeldtekst 4.</p>

    <button onclick="veranderAchtergrondKleur()">Verander Achtergrondkleur</button>

    <script>
        function veranderAchtergrondKleur() {
            // maak de code af en plaats hieronder de code die de kleur van de tekst met de
class kleurVerander verandert
            ...
        }
    </script>
</body>
```

```
</html>
```

De JavaScript-functie wordt aangeroepen vanuit de `onclick` attribuut van de HTML-knop.

Wanneer de knop wordt ingedrukt, moet de kleur van de tekst in alle paragraafelement met de class *kleurVerander* veranderen. Maak hiervoor het script af.

Maak gebruik van document.getElementsByClassName en gebruik een for-loop.

> Denk erom dat document.getElementsByClassName een array met elementen terug geeft en dat je met de loop door het array moet heenlopen.

Bedek zelf een mooie kleur.

Inleveren

Zet jouw naam als commentaar in de JS-code en lever het geteste en werkende HTML-pagina in.

Opdracht 3, stijl toggle

Je hebt de volgende HTML/JS-code:

```
<!DOCTYPE html>
<html lang="nl">
<head>
    <meta charset="UTF-8">
    <title>Tekststijl Veranderen</title>
</head>
<body>
    <h1>Dit is een koptekst</h1>
    <p>Dit is een gewone paragraaf.</p>
    <p class="speciaal">Dit is een speciale paragraaf.</p>
    <div>Dit is een div element.</div>
    <p class="speciaal">Nog een speciale paragraaf.</p>
    <p id="speciaal">Nog een speciale paragraaf.</p>

    <button onclick="veranderTekstStijl()">Verander Tekststijl</button>
```

```
    <script>
        function veranderTekstStijl() {
            var elementen = document.querySelectorAll(........);
            for (......) {
□□      // Maak de elementen green, bold, and italic (schuingedrukt)
            }
    </script>
</body>
</html>
```

Vul de code aan op de puntje en zorg ervoor dat zodra op de knop wordt gedrukt:

- alle elementen met de class *speciaal* ***groen, vet en schuingedrukt*** worden.

Als er weer op de knop wordt gedrukt dan worden:

- alle elementen met de class *speciaal* worden weer zwarte, niet vet en niet schuingedrukt.

De knop wisselt dus telkens de stijl alle tekst met de class *speciaal*.

Inleveren

Zet jouw naam als commentaar in de JS-code en lever het geteste en werkende script in.

Zet jouw naam als commentaar in de JS-code en lever het geteste en werkende HTML-pagina in.

Opdracht 4, counter

Bekijk het volgende voorbeeld.

```
<div id="nummerDiv">123</div>
<button onclick="toonGetal()">Toon Getal</button>

<script>
    function toonGetal() {
         var divElement = document.getElementById('nummerDiv');
         var getal = divElement.innerText; // Lees de tekst (in dit geval een getal) uit de
div
```

```
        alert("Het getal in de div is: " + getal);
    }
</script>
```

Probeer te begrijpen wat er gebeurt en test de code om te kijken of je snapt wat er gebeurt!

Opdracht

1. maak een extra button
2. als je op de button druk dan roep je een javaScript funcite aan (bedenk zelf een naam voor de functie).
3. met een console.log druk je de waarde af die in de div staat; in dit voorbeeld dus 123.
4. tel 1 op bij het getal en druk dit getal ook af met een console.log. In dit voorbeeld is dat dus 124.

   > Als je in JS een getal bij een getal optelt met + dan 'denkt' JS standaard dat je een string hebt. Dan wordt 1+1 dus 11 en niet 2! Om dit te voorkomen moet je aangeven dat je geen string maar een integer hebt.
   >
   > Dit doe je met parseInt(string).
   >
   > Dus parseInt(getal) + 1 wordt dan 2.
   >
   > Dus, stel getal = 1
   >
   > getal + 1 wordt 11 en
   > parseInt(getal) + 1 wordt 2

5. dan zet je het getal (in dit voorbeeld dus 124) terug in de div.

Als jet goed is, heb je nu een div met een getal en telkens als je op de knop druk wordt het getal opgehoogd.

Inleveren

Zet jouw naam als commentaar in de JS-code en lever het geteste en werkende HTML-pagina in.

Opdracht 5, Ronde teller

Maak een interactieve webpagina met een tabel die schaatsers en hun rondeteller weergeeft.

Schaatser	Aantal Rondes
Mo Mot	0
Ayoub Allemaal	2
Klaas Kaas	0
Goedele Goedgeluk	0
Roos Rommelhuis	0
Lars Leeghoofd	0

De gebruiker kan het aantal ronden voor elke schaatser verhogen door op de teller in de tabel te klikken.

Begin met deze HTML/CSS-code:

```
<!DOCTYPE html>
<html lang="nl">
<head>
    <meta charset="UTF-8">
    <title>Schaatser Ronde Teller</title>
    <style>
        table {
            width: 50%;
            border-collapse: collapse;
            margin: 20px auto;
        }
        th, td {
            border: 1px solid #dddddd;
            text-align: left;
            padding: 8px;
        }
        th {
            background-color: #f2f2f2;
        }
        td.rondes {
            cursor: pointer;
            background-color: #e6ffe6;
        }
        td.rondes:hover {
            background-color: #ccffcc;
        }
    </style>
</head>
```

```
<body>

    <table>
        <tr>
            <th>Schaatser</th>
            <th>Aantal Rondes</th>
        </tr>
        <tr>
            <td>Mo Mot</td>
            <td class="rondes" onclick="verhoogRondes(this)">0</td>
        </tr>
        <tr>
            <td>Ayoub Allemaal</td>
            <td class="rondes" onclick="verhoogRondes(this)">0</td>
        </tr>
        <tr>
            <td>Klaas Kaas</td>
            <td class="rondes" onclick="verhoogRondes(this)">0</td>
        </tr>
        <tr>
            <td>Goedele Goedgeluk</td>
            <td class="rondes" onclick="verhoogRondes(this)">0</td>
        </tr>
        <tr>
            <td>Roos Rommelhuis</td>
            <td class="rondes" onclick="verhoogRondes(this)">0</td>
        </tr>
        <tr>
            <td>Lars Leeghoofd</td>
            <td class="rondes" onclick="verhoogRondes(this)">0</td>
        </tr>
    </table>

</body>
</html>
```

Maak de code af en maak een script waarbij als je op het ronde nummer drukt, de ronde met een wordt verhoogd. Denk aan het gebruik van parseInt (vorige opdracht).

Inleveren

Zet jouw naam als commentaar in de JS-code en lever het geteste en werkende HTML-pagina in.

Opdracht 6, eventListener

In plaats van een functie aan heel veel rijen toe te voegen, kan je ook een zogenaamde **eventListener** maken.

Een event is een gebeurtenis en een **eventListener** luistert dus of er een bepaalde gebeurtenis plaatsvindt.

We gaan onze code uit de vorige opdracht (5) aanpassen.

1. haal alle onclick attributen uit de <td>'s weg.
2. geef de tabel een id '*rondeTellerTabel*'

Bekijk de volgende code:

```
document.addEventListener('DOMContentLoaded', function() {
            var rondesCellen = document.querySelectorAll('#rondetellerTabel .rondes');
            rondesCellen.forEach(function(cel) {
                cel.addEventListener('click', verhoogRondes);
            });
        });
```

Uitleg

Regel	Uitleg
1	Zodra de DOM (HTML-document) is geladen voer de volgende code uit.
2	Laad alle cellen uit de tabel met het id *rondeTellerTabel* en de class *rondes.*
3	Itereer door alle gevonden cellen (*iteren* betekent 'doorlopen').
4	Laat alle gevonden cellen luisteren naar een click en als er een click plaatsvindt, roep dan de functie *verhoogRondes* aan.

De functie *verhoogRondes* moeten we iets aanpassen.

```
function verhoogRondes() {
            var huidigeRondes = parseInt(this.innerText);
            this.innerText = huidigeRondes + 1;
        }
```

Hier gebruiken we *this*, *this* is het huidige element waarop is gelicked.

Zorg ervoor dat de ronde teller weer werkt maar nu met de eventListener zoals hierboven uitgelegd.

Lees goed en aandachtig en als het moet lees het nog eens stap voor stap. Als je alle stappen volgt en begrijpt dan zul je de opdracht kunnen maken.

Inleveren

De aangepaste code waarin je ronde kan tellen van de vorige opdracht, maar nu maak je gebruik van de eventlistener zoals uitgelegd.

Opdracht 7, shift key

We nemen de code van onze vorige opdracht en gaan functionaliteit uitbreiden.

Soms wordt er namelijk per ongeluk op de verkeerde kolom geclicked. Je wilt dan het rondenummer verlagen.

Als de shif-key wordt ingedrukt terwijl we clicken moet de het rondenummer worden verlaagd.

Je kun in JS testen of de shift-key wordt ingedrukt:

```
if (event.shiftKey) {
```

Gebruik dit en bereid de JS code uit zodat:

- als je clicked, wordt het rondenummer één hoger (dat hadden we al).
- als je shift-clicked, wordt het rondenummer één lager (nieuw).

We hebben nog één extra eis:

- het rondenummer mag mag niet negatief worden.

Inleveren

De aangepaste code (volledig werkende HTML/CSS/JS) van de vorige opdracht aangevuld met de functionaliteit zoals hierboven beschreven.

Plaats commentaar in de code met uitleg én plaats commentaar met jouw naam.

Opdracht 8, shift key en kleur

We maken nog een laatste aanpassing aan de ronde teller. Zorg dat je de vorige opdracht helemaal af hebt.

Als je op de rondeteller klikt wordt de rondeteller verhoogd, doe je dat met shift dan wordt de ronde teller verlaagd.

De achtergrondkleur van de ronde teller is groen. Om aan te geven dat de rondeteller is verlaagd wordt de achtergrondkleur rood. Als daarna de teller weer wrodt verhoogd, wordt de achtergrond kleur weer groen.

Schaatser	Aantal Rondes
Mo Mot	0
Ayoub Allemaal	2
Klaas Kaas	4
Goedele Goedgeluk	0

De eisen:

- standaard staan alle ronde tellers op 0 en zijn ze groen.
- als je op een ronde teller clicked dan, wordt die verhoogd en is/wordt de achtergrondkleur groen.
- als je op een rondeteller shift-clicked, dan wordt die verlaagd en/is wordt de achtergrondkleur rood.
- de achtergrondkleur mag neit negatief worden (lager dan 0), dus als je op een 0 shift-clicked dan gebeurt er niets; de achtergrondkleur blijft hetzelfde en de rondeteller wordt niet verlaagd.

Inleveren

De aangepaste code (volledig werkende HTML/CSS/JS) van de vorige opdracht aangevuld met de functionaliteit zoals hierboven beschreven.

Plaats commentaar in de code met uitleg én plaats commentaar met jouw naam.

Opdracht 9, uitklappen

Bekijk en test de volgende code.

```
<!DOCTYPE html>
<html lang="nl">
<head>
    <meta charset="UTF-8">
    <title>Elektronica Producten</title>
    <style>
        .detailRow {
            display: none;
            background-color: #f9f9f9;
        }
        .clickableRow:hover {
            background-color: #f1f1f1;
            cursor: pointer;
        }
        table {
            width: 100%;
            border-collapse: collapse;
        }
        th, td {
            border: 1px solid #dddddd;
            text-align: left;
            padding: 8px;
        }
        th {
            background-color: #f2f2f2;
        }
    </style>
</head>
<body>
```

```
<table id="infoTabel">
    <tr>
        <th>Product</th>
        <th>Categorie</th>
    </tr>
    <tr class="clickableRow">
        <td>iPhone 13</td>
        <td>Smartphone</td>
    </tr>
    <tr class="detailRow">
        <td colspan="2">Merk: Apple | Opslag: 128GB | Kleur: Zwart</td>
    </tr>
    <tr class="clickableRow">
        <td>Samsung Galaxy S21</td>
        <td>Smartphone</td>
    </tr>
    <tr class="detailRow">
        <td colspan="2">Merk: Samsung | Opslag: 256GB | Kleur: Phantom Gray</td>
    </tr>
    <tr class="clickableRow">
        <td>Dell XPS 13</td>
        <td>Laptop</td>
    </tr>
    <tr class="detailRow">
        <td colspan="2">Merk: Dell | Scherm: 13.3 inch | CPU: Intel i7</td>
    </tr>
    <tr class="clickableRow">
        <td>MacBook Air</td>
        <td>Laptop</td>
    </tr>
    <tr class="detailRow">
        <td colspan="2">Merk: Apple | Scherm: 13.3 inch | CPU: M1</td>
    </tr>
    <tr class="clickableRow">
        <td>Sony WH-1000XM4</td>
        <td>Hoofdtelefoon</td>
    </tr>
    <tr class="detailRow">
        <td colspan="2">Merk: Sony | Type: Over-ear | Noise Cancelling: Ja</td>
    </tr>
```

```
    <tr class="clickableRow">
        <td>Google Nest Hub</td>
        <td>Smart Home</td>
    </tr>
    <tr class="detailRow">
        <td colspan="2">Merk: Google | Scherm: 7 inch | Assistent: Google Assistent</td>
    </tr>
</table>

<script>
    document.querySelectorAll('.clickableRow').forEach(row => {
        row.addEventListener('click', function() {
            var detailRow = this.nextElementSibling;
            detailRow.style.display = 'table-row';
        });
    });
</script>

</body>
```

Deze code laat een tabel zien en als je op de regel clicked dan worden details getoond.

clickableRow

De hoofdregels waar je op kan clicken hebben de class *clickableRow.*

detailRow

De detail regels die zichtbaar/onzichtbaar worden door op de hoofdregels te clicken heben de class *detailRow*.

Opdracht

Stap 1

Pas de code aan zodat als de detailregel wordt getoond en er weer op de *clickableRow* wordt gedrukt de detailregel weer wordt 'weggehaald' (`style.display = 'none'`).

Het klikken op de *clickableRow* wordt dus een toggle: het laat de details zien en als er weer wordt geclicked worden de details weer weggehaald.

Stap 2a

We willen een andere vormgeving; de detail-regels moeten inpsringen. Jou lead developer gevraagd om de code op te bouwen en om geen table te gebruiken maar div. Je kirjgt dus een div met een class *clickabelDIv* en daaronder een div met een class *detailRow*

Zorge ervoor dat de twee kolommen in de *clickableDIv* apart in de div staan. Je hebt telkens de naam van het product (bijvoorbeeld iPhone 13) en de catergorie (bijvoorbeeld Smartphone), Maak hier twee divs van die naast elkaar staan (tip: gebruik flexbox). Je hebt dus twee divs die in een div staan:

```
<div class="clickableRow">
    <div class="product">iPhone 13</td>
    <div class="category">Smartphone</td>
</div>
<div class="detailRow">
  ......
```

Stap 2b

Als de table is omgebouwd in divs, zorg er dan voor dat de clickableRows en *detailRow*beter te ondersheiden zijjn:

- De grootte van het lettertype van de *clickableRows* is meer dan die van de *detailsRows*.
- De detailsRows krijgen een margin waardoor het lijkt dat deze inspringen (iets naar rechts gaan)

Stap 3

Pas de kleuren van de tabel aan zodat de tabel mooi is en goed leesbaar is

Stap 4

Plaats commentaar in de code waarin je uitlegt hoe de code werkt. Zet ook je naam in ergens in het commentaar.

Voorbeeld

Als je klaar bent dan zou het er zo kunnen uitzien (kleuren zelf kiezen).

Product	Categorie
iPhone 13	Smartphone
Merk: Apple | Opslag: 128GB | Kleur: Zwart	
Samsung Galaxy S21	Smartphone
Dell XPS 13	Laptop
Merk: Dell | Scherm: 13.3 inch | CPU: Intel i7	
MacBook Air	Laptop
Sony WH-1000XM4	Hoofdtelefoon
Google Nest Hub	Smart Home

Inleveren

De aangepaste code (volledig werkende HTML/CSS/JS) van de vorige opdracht aangevuld met de functionaliteit zoals hierboven beschreven.

Opdracht 10, Winkelmandje

Je hebn nu een soort productpagina gemaakt bij opdracht 9.

We gaan een begin maken met een winkelmandje. Hiervoor gebruiken we de code van de vorige opdracht en breiden deze uit.

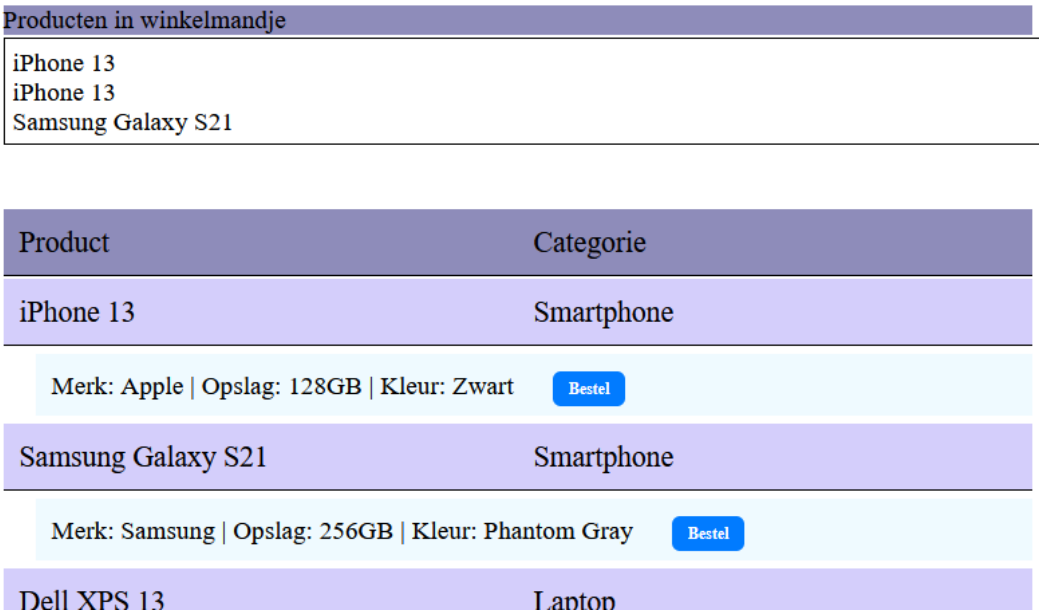
1. Op elke detailRow komt een knopje *Bestellen* - maak deze knop en geef deze knop een mooi CSS stijl.

   Voorbeeld:

iPhone 13	Smartphone
Merk: Apple | Opslag: 128GB | Kleur: Zwart Bestel	

2. Bovenin de pagina maak je een nieuwe div, dit is het winkelmandje.
3. Als de knop *Bestel* wordt ingedrukt dan wordt het product toegevoegd als een regel in het winkelmandje. Al sje bijvoorbeeld op iPhone 13 clicked dan verschijnt er een regel in het winkelmandje met iPhone 13.
4. Als je dezelfde knop weer indrukt dan komt er een tweede regel iPhone13 in het winkelmandje.
5. Maak vervolgens een knop in de het winkelmandje waarmee je het winkelmandje leeg kunt maken.

Voorbeeld



Inleveren

De aangepaste code (volledig werkende HTML/CSS/JS) van de vorige opdracht aangevuld met de functionaliteit zoals hierboven beschreven.

Optie, Teller in winkelmandje

Zet in het winkelmandje achter elk product een teller waarmee je klan aangeven hoeveel je wilt bestellen. Gebruik een (deel van de) code van de rondeteller.

- Zorg ervoor dat het aantal niet lager dan 0 en niet hoger dan 10 kan worden.
- (optioneel) zorg ervoor dat er geen dubelle regels in het winkelmandje kunnen ontstaan.

Inleveren

De aangepaste code (volledig werkende HTML/CSS/JS) van de vorige opdracht aangevuld met de functionaliteit zoals hierboven beschreven.

Revision #4
Created 2024-06-08 15:32:51 UTC by Max
Updated 2025-06-06 14:05:54 UTC by Max