

# Java Script Challenge

## *DOM Challenge – Bouw jouw eigen mini-app*

[datasource](#)

### ? Leerdoelen

- Je past alle basisvaardigheden toe rondom DOM-manipulatie.
- Je maakt een interactieve webapp met HTML, CSS en JavaScript.
- Je leert werken met gebruikersinvoer, events en eventueel localStorage.
- Je leert een eenvoudige planning te maken voor je project.
- Je leert AI bewust inzetten en daarover verantwoording afleggen.

### ? Uitleg

In deze les ontwerp je zelf een interactieve DOM-app. Je past de technieken toe uit de vorige lessen. Kies één van de volgende apps, of verzin een eigen variant.

### Mogelijke projecten:

- **Todo-lijst:** gebruiker voert een taak in, kan deze toevoegen, afvinken (class toggle), en verwijderen (event.target.remove()).
- **Quiz:** gebruiker kiest een antwoord en krijgt direct feedback.
- **Poll/stemming:** klik op een optie, zie het aantal stemmen stijgen.
- **Chatbox:** gebruikersberichtjes invoeren die onder elkaar verschijnen.
- **Boekenkast:** boeken toevoegen, afvinken als gelezen, verwijderen, zoeken én opslaan in localStorage.

### ? Eisen

**Let op:** Bij elk project gebruik je minimaal deze onderdelen:

- Een of meer **invoervelden + knoppen**
- Een **lijst** of reeks elementen die via JavaScript groeit
- Een invoerveld met een **search** functie waarbij als je die gebruikt alleen de gezochte items worden getoond.
- Na een search moet je ook weer terug gaan en **alle** elementen kunnen laten zien.
- **Event-handling** (bijv. click, submit)
- Bij een delete van een item, vraag je om eerst om **bevestiging**: "Weet je zeker dat je item 'voorbeeld' wilt verwijderen?"
- Je maakt tenminste gebruik van `querySelector`, `addEventListener`, en `innerHTML`, `classList.toggle()` en `remove()`

Bij het project **Boekenkast** moet je daarnaast ook werken met `localStorage`.

## ? Optioneel

- Als je alles in één pagina maakt, dan kan je er eventueel vrij eenvoudig een app voor je telefoon van maken.

## ?? Opdracht – Kies en bouw jouw mini-app

1. Maak een bestand met een duidelijke naam, bijv. `dom-project.html`.
2. Kies welk DOM-project je maakt (zie hierboven).
3. Maak een **planning** van je project in een tabel (zie voorbeeld hieronder).
4. Werk stap voor stap: begin met HTML, voeg daarna JavaScript toe.
5. Gebruik technieken uit eerdere lessen (DOM)
6. Stijl je pagina met CSS zodat deze overzichtelijk en prettig in gebruik is.

### ☐ Voorbeeld planning

Onderdeel	Tijd	Status
HTML structuur	20m	

JS: items toevoegen	30m		
JS: verwijderen/afvinken	20m		
Styling	15m		
Reflectie schrijven	15m		
.....			
....			

## ? Reflectie

- Wat vond je het makkelijkst en het moeilijkst?
- Welke technieken heb je toegepast? Noem er minstens drie.
- Waar heb je AI voor gebruikt?
- Wat zou je in de toekomst nog willen verbeteren of leren?

## ? Inleveren

- Lever je `dom-project.html` in, samen met je CSS-bestand (indien apart).
- Lever je **planning** in als .txt of .pdf.
- Lever je **reflectie** in als .txt of .pdf.
- Lever je **AI-logboek** in: geef aan welke prompts je gebruikte, wat je codeerde met hulp van AI, en wat je zelf schreef of aanpaste.
- Eventueel: voeg screenshots toe van je werkende webapp als je trots bent op het resultaat!
- **Je laat het resultaat aan de docent zien en je kan de code uitleggen.**

---

Revision #8

Created 2025-06-06 20:40:38 UTC by Max

Updated 2025-11-12 12:37:30 UTC by Max