

Reader

- JavaScript BOM en DOM
- JavaScript Window-object
- JavaScript Document-object
- JavaScript Events

JavaScripty & jQuery

- Chapter 3: Functions, Methods & Objects, pagina 120 vanaf de kop "What are Built-in Objects?".
- Chapter 5: Document Object Model.
- Chapter 6: Events.

Het boek 'even doorlezen' is niet voldoende! Doe altijd de opdrachten uit het boek.

Opdracht 1

1. Bestudeer de HTML-code van 'webshop-prototype.html'. De bedoeling van de opdracht is deze HTML-code dynamisch te genereren met behulp van JavaScript. Je gebruikt hiervoor DOM-technieken.
2. De HTML-tags moeten worden gevuld met teksten die zijn aangeleverd in het bestand 'webshop-database.js'. Zoals je ziet zijn er drie producten omschreven als objecten, die in een array zijn gestopt met de naam 'producten'.
3. Open 'webshop.html' en 'webshop.js'.
4. Schrijf nu met JavaScript de benodigde code om 'webshop-prototype.html' exact na te maken. Je gebruikt hiervoor o.a. DOM manipulatie om tags dynamisch aan te maken en te vullen met gegevens uit de producten array van 'webshop-database.js'.
5. Voorkom dubbele code: haal productgegevens op middels een for-loop.

Opdracht 2

1. Bekijk in de browser eerst het eindresultaat van 'events-oefening-1-result.html'. Het rode blokje beweegt steeds een X-aantal pixels (die je kunt opgeven in het input-veld), na een druk op de knop.
2. Open 'events-oefening-1.html' en bestudeer de HTML-code.
3. Maak een variabele genaamd 'blokje' en assign het HTML-object #blokje.
4. Maak een variabele genaamd 'knop' en assign het HTML-object #moveKnop.

5. Maak een variabele genaamd 'hoeveelBewegen', zodat je kunt bijhouden wat de huidige CSS left property value is van #blokje. Assign de variabele met het getal 0.
6. Maak een functie genaamd moveBlokje().
7. Schrijf nu de functie-code voor moveBlokje() door de value van #moveHoeveel op te vragen.
8. Update daarna de variabele 'hoeveelBewegen' door de waarde van #moveHoeveel op te tellen bij de huidige waarde van 'hoeveelBewegen'.
9. Update de CSS left property van #blokje met de nieuwe waarde van 'hoeveelBewegen'.
10. Schrijf een DOM event handler voor #moveKnop. Wanneer er op #moveKnop gedrukt wordt, moet de functie 'moveBlokje' worden uitgevoerd.

Opdracht 3

1. Bekijk in de browser eerst het eindresultaat van 'events-oefening-2.html'. Klik op één van de tien rode blokken om zijn kleur te veranderen in blauw.
2. Open 'events-oefening-2.html'
3. Schrijf een onload event voor window, zodat jouw JavaScript-code pas wordt uitgevoerd als de browser klaar is met het opbouwen van de DOM.
4. Maak een variabele aan 'deDivs' (array). Assign het met alle div-tags van het document.
5. Schrijf een for-loop zodat elke div in het document een rode achtergrond kleur krijgt.
6. Maak een functie 'changeColor'. De functie heeft één parameter genaamd 'geklikteDiv'.
7. Schrijf de code voor de functie 'changeColor'. De bedoeling is de achtergrondkleur van 'geklikteDiv' rood te maken als het blauw is, en blauw als het rood is.
8. Schrijf een for-loop zodat elke div in het document een click-event bind krijgt middels de .addEventListener() method.
9. Gebruik het eventObject dat JavaScript automatisch aanmaakt bij addEventListener() om er achter te komen op welk object er geklikt is.
10. Als het click event plaats vindt moet de functie changeColor aangeroepen worden. Vergeet niet de functie één parameter mee te geven: de zojuist aangeklikte div.
11. Extra oefenen: maak de .addEventListener compatible met IE-browsers, volgens de voorbeelden uit je boek.