

Objekter i HTML-dokumentet

Foreløpig har vi stort sett jobbet med JavaScript og HTML/CSS hver for seg, men det er først når vi kombinerer disse tre språkene, at vi kan lage fine og interaktive nettsider. I dette læringsløpet skal vi se på hvordan vi kan hente ut og endre HTML-innhold med JavaScript.

Når en nettleser skal vise fram en nettside, lager den en modell av nettsidens innhold. Denne modellen kalles *Document Object Model* (DOM). Den har fått dette navnet fordi modellen består av objekter som hører til HTML-dokumentet. Objektene er blant annet HTML-elementer. Det er denne modellen som lager grunnlaget for hva nettleseren viser fram, og det er innholdet i denne modellen vi kan redigere ved å bruke JavaScript. Det er altså Document Object Model vi redigerer med JavaScript, ikke selve HTML-koden.

La oss se på et eksempel før vi går inn i detaljene. I programmet nedenfor kan du se et enkelt eksempel på et HTML-dokument med litt CSS-kode. Vi skal nå se på DOM-en til denne nettsiden.



```
1  <!doctype html>
2  <html>
3    <head>
4      <title>DOM</title>
5      <meta charset="utf-8">
6      <style>
7        body {
8          background-color: #f5f5f5;
9          font-family: Georgia, serif;
10       }
11       #hoved {
12         width: 60%;
13         margin: 100px auto;
14         padding: 40px;
15         background-color: white;
16         border: 1px solid #cccccc;
17       }
18       h1, h2 {
19         font-family: Arial, sans-serif;
20       }
21       .blaa {
22         padding: 20px;
23         background-color: #01739F;
24       }
25     </style>
26   </head>
27   <body>
28     <div id="hoved">
29       <h1>Velkommen til siden min</h1>
30       <p>På denne siden kan du lære mer om meg.</p>
31       <p class="blaa">Min favorittfarge er blå.</p>
32     </div>
33   </body>
```

DOM-oversikt

HTML-elementene i denne koden kan vi illustrere som i figuren nedenfor. Vi har `<html>`-elementet, som inneholder alle de andre elementene, og vi har `<head>`-elementet, som inneholder `<title>` og `<style>`, og `<body>`-elementet, som inneholder et `<div>`-element med en overskrift (`<h1>`) og to avsnitt (`<p>`).

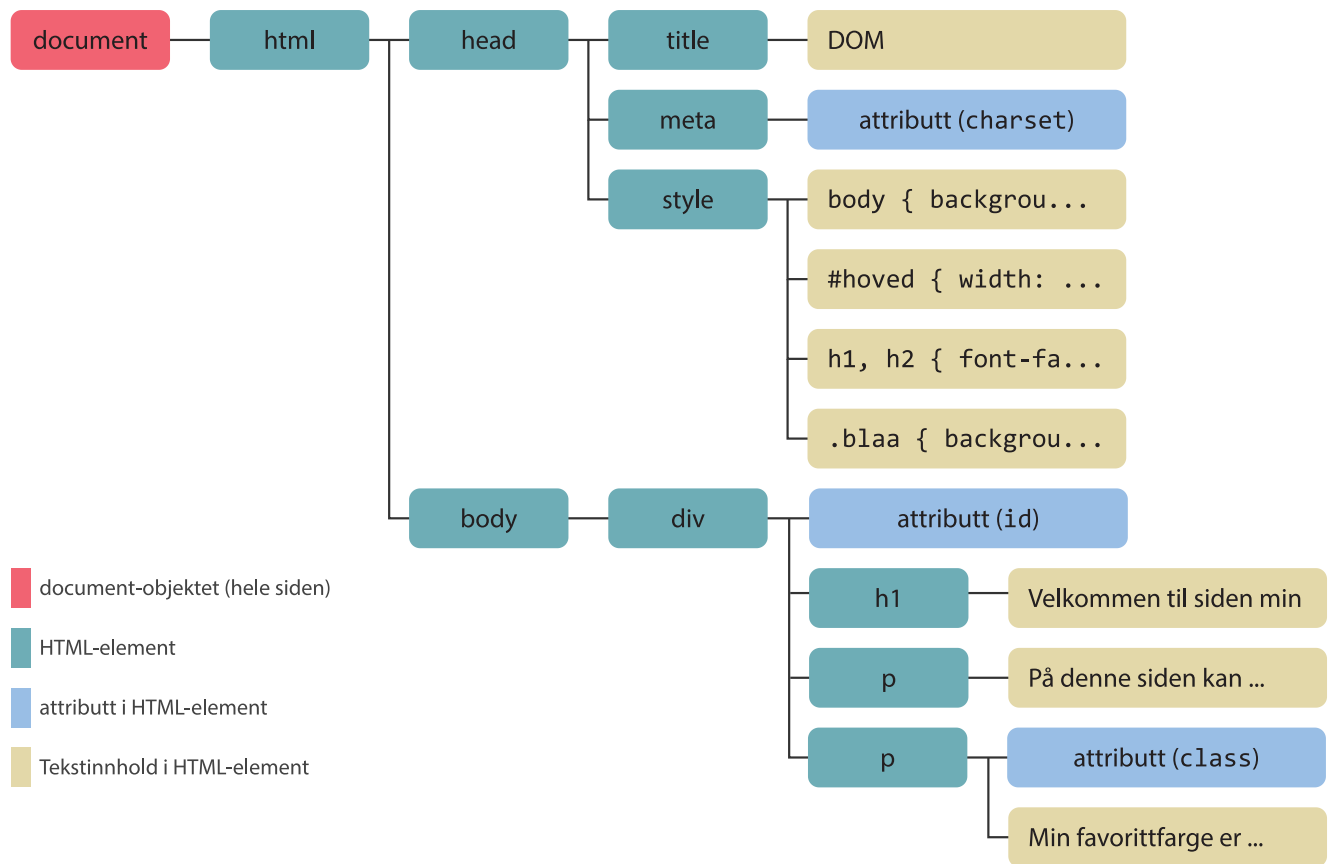
Disse elementene er alle eksempler på objektene som til sammen utgjør innholdet i «Document Object Model» til denne nettsiden.



DOM-treet

I tillegg til HTML-elementer finner vi også HTML-attributter og tekstinnhold i «Document Object Model». I eksemplet ovenfor er altså `class="blaa"` et eksempel på en attributtegenskap, mens teksten `Velkommen til siden min` er et eksempel på en tekstegenskap. Til sammen utgjør HTML-elementer, HTML-

attributter og tekstinnhold et *DOM-tre*. DOM-treet for vårt eksempel kan du se nedenfor. I «starten» av treet finner vi også «document-objektet», som representerer hele dokumentet vårt.



Snakk!

DOM-treet

Sammenlign DOM-treet med koden lenger opp på siden. Finner du igjen i alt innholdet?

DOM-treet viser alt innholdet på nettsiden, og vi kan gå inn og redigere alle elementene i treet ved å bruke JavaScript. Vi kan også legge til og fjerne elementer, slik at vi i teorien kan bygge opp fullstendige nettsider utelukkende ved hjelp av JavaScript. I resten av dette læringsløpet skal vi se på hvordan vi kan opprette kontakt mellom JavaScript og DOM-treet, og hvordan vi kan bruke JavaScript til å gjøre endringer på det.

Oppgaver

- 1 Hva er «Document Object Model» (DOM)?
- 2 Ta utgangspunkt i koden og figurene ovenfor og gi eksempler på hva slags innhold vi finner i «Document Object Model».
- 3 Finn et HTML-dokument du har laget tidligere. Lag et DOM-tre for dokumentet.