



## Introduksjon

HTML, HyperText Markup Language, er språket som beskriver innholdet på alle nettsider. HTML er strengt tatt ikke et programmeringsspråk, men brukes sammen med språk som for eksempel JavaScript, PHP og Python for å lage fullstendige nettløsninger.

## Steg 1: Verdens enkleste nettside

La oss komme i gang med å lage verdens enkleste nettside.

### ✓ Sjekkliste

- ☐ Åpne **Notepad** eller et tilsvarende program for å skrive ren tekst (ikke en tekstbehandler som for eksempel **Word**).
- ☐ Skriv følgende linje øverst i dokumentet:

Hei verden!
- ☐ Lagre dokumentet som **heiverden.html**. Pass på at etternavnet til filen blir **.html** og ikke **.txt** eller **.html.txt**.

### 🚩 Se resultatet

- ☐ Åpne en nettleser og velg **Fil -> Åpne fil** eller tilsvarende fra menyen (du kan ofte bruke tastatursnarveien **Ctrl - O**). Velg filen **heiverden.html** som vi nettopp lagde.
- ☐ Du vil nå se en nesten tom skjerm med teksten **Hei verden!** øverst i venstre hjørne. Vi har laget verdens enkleste nettside.

Strengt tatt har vi **ikke** skrevet et HTML-dokument enda. HTML (i motsetning til de fleste programmeringsspråk) blir tolket i *beste mening*, og nettleseren tilgir oss at vi har gjort en del feil. La oss derfor utvide eksempelet vårt.

### ✓ Sjekkliste

- ☐ Gå tilbake til **Notepad** og endre **heiverden.html** slik at den ser ut som følger:

```
<html>
<head>
<title>Hei HTML</title>
</head>

<body>
<h1>Hei verden!</h1>
<p>
Her er en liten nettside.
</p>
</body>
</html>
```

- ☐ Lagre filen, gå tilbake til nettleseren og frisk opp siden ( **F5** ). Finner du igjen tekstene **Hei HTML** , **Hei verden!** og **Her er en liten nettside** på siden?

HTML er et såkalt *Markup Language*. Det betyr at vi markerer hva forskjellige deler av innholdet er. Du ser en del ord mellom **<** og **>**, for eksempel **<html>** , **<body>** og **<h1>** . Disse er markørene.

Legg også merke til at markørene kommer i par, hvor de avsluttes med en ekstra **/** . For eksempel **<title>Hei HTML</title>** , **<h1>Hei verden!</h1>** og til og med **<html> ... andre markører og tekst ... </html>** .

- ☐ Se på HTML-koden du skrev tidligere. Klarer du å gjette hva de forskjellige HTML-markørene betyr?  
Et HTML-dokument skal være avgrenset av **<html> ... </html>** , og delt i to deler: En **head** som inneholder informasjon om dokumentet som ikke vises på selve siden, og en **body** som er selve innholdet på siden.  
Markøren **<title>** viser tittelen til nettsiden. Denne brukes ofte av nettleseren på faner og i historikken. For å markere overskrifter kan vi bruke forskjellige **<h ...>** -arkører ( **h** for *heading* ). **<h1>** er den største overskriften, mens også **<h2>** til **<h6>** kan brukes. Vanlig tekst legges ofte i **<p>** -markører ( **p** for *paragraph* ).
- ☐ Prøv å endre litt på nettsiden din. Endre størrelsen på overskriften. Legg til flere avsnitt og flere overskrifter.

## Steg 2: Lenker

En av de tingene som gjorde nettet så revolusjonerende var muligheten til å lenke sammen informasjon (det er dette som ble kalt *Hypertext*). Vi skal nå se hvor lett det er å lage lenker på nettsiden vår.

### ✓ Sjekkliste

- ☐ Legg til følgende avsnitt i **body** en i dokumentet ditt:

```
<p>
Vi er på <a href="http://www.kartverket.no/">Kartverket</a> i dag.
</p>
```

- ☐ Se på siden i nettleseren. Dukk det opp en lenke til Kartverkets hjemmeside?

For å lage lenker bruker vi markøren **<a>** ( **a** for *anchor* ). Men i tillegg til navnet på markøren ser vi at vi også gir litt ekstra informasjon. Vi må fortelle hvor lenken skal peke. Dette gjør vi ved å bruke nøkkelordet **href** (*hyper-reference*).

- ☐ Prøv selv å endre lenken slik at den peker et annet sted.
- ☐ Legg til flere lenker på siden din.

## Steg 3: Stiler

Nå har vi laget innhold på siden vår. Men den ser ikke veldig fin ut. For det meste bryr ikke HTML seg om utseendet. I stedet bruker vi noe som kalles CSS (Cascading Style Sheets) for å fortelle hvordan siden skal se ut.

### ✓ Sjekkliste

- ☐ Legg til følgende i **head** -delen av dokumentet ditt.

```
<style type="text/css">
  body { background: yellow }
</style>
```

Vi kjenner igjen `<style>` som en markør som sier at den markerte teksten er av typen CSS. Selve teksten `body { background: yellow }` inneholder et par rare tegn, men virker ellers ganske selvforklarende. Det ser ut som om vi ønsker en gul bakgrunn!

- ☐ Frisk opp siden din, og se at den nå har fått en nydelig gulfarge!
- ☐ Vi kan legge på flere stilegenskaper ved å skille dem med semikolon. Endre stilen din som følger:

```
body { background: yellow; color: blue }
```

Nå skrives teksten i en blå farge.

Legg merke til at vi ikke trengte å spesifisere at overskriftene (`<h1>`) eller avsnittene (`<p>`) skulle skrives med blått. I stedet har vi sagt at all tekst innenfor `<body>` skal være blå, dette er hva *cascading* betyr.

- ☐ Vi kan likevel overstyre stiler på lavere nivå. Endre stilen din ved å legge til noen egenskaper for `p`:

```
body { background: yellow; color: blue }  
p { color: green; font-family: Arial, sans-serif }
```

## Steg 4: Bokser

Av og til vil vi ha enda mer kontroll over utseendet enn vi får gjennom vanlige HTML-markører. Til dette kan vi bruke noe som kalles bokser og klasser.

### ✓ Sjekkliste

- ☐ Legg til følgende linje i CSS-definisjonen din:

```
.utrop { color: red }
```

Her er ikke `.utrop` en vanlig markør. I stedet forteller vi om utseendet til *klassen* `utrop`. Dette gjør det enkelt å legge litt spesiell funksjonalitet på deler av nettsiden.

- ☐ Legg til et avsnitt på formen:

```
<p>  
Her er en <span class="utrop">liten</span> nettside.  
</p>
```

Markøren `span` bruker vi her bare til å lage en liten (usynlig) boks som vi så kan anvende klassen `utrop` på.

- ☐ Med `<div>` kan man lage større bokser. Den skal vi se på etter pausen!