

HTML & CSS

En grundläggande introduktion

Fredrik Hall 2016-08-30

Johan Holmberg 2023-11-10

En webbsida

Fyra nivåer

- Innehåll (text, bilder, video)
- Struktur (**HTML**)
- Presentation (**CSS**)
- Beteende (Javascript)

Separation of concerns!

HTML – *HyperText Markup Language*

Ett av många märkspråk (HTML, XML, SVG m.fl.)

- Ger innehåll betydelse (semantik)
- Språket utgör regler för hur det ska skrivas (syntax)

Grunden för en webbsida

HTML-element

Byggstenen

Definierar innehållets semantiska betydelse

Elementets beståndsdelar



Elementtypen beskriver innehållets typ (semantisk betydelse)

Ytterligare information om elementet anges som attribut

Taggar innesluter innehåll

Taggar

```
<!-- Exempel på element med start- och sluttagg -->  
<h1>HTML och CSS</h1>  
  
<!-- Exempel på en öppen tagg -->  

```

Elementet har antingen:

- En start- och sluttagg:
 - <tagg-typ> innehåll </tagg-typ>
- Öppen tagg (tomma element, ingen sluttagg):
 - <tagg-typ>

Attribut

```
<!-- Elementet img med attributen src och alt samt värde -->  
  
  
<!--Elementet a med attributet href samt värde -->  
<a href="http://google.se">Google</a>  
  
<!-- Elementet p med attributen id och class samt värde -->  
<p id="mittId" class="minKlass"></p>
```

Ger ytterligare information om elementet

- Attribut-namn och -värde (ex. `class="minKlass"`)
- Generella och specifika attribut
 - Vissa attribut är obligatoriska (ex. attributet `src` för elementet `img`)
 - Andra är frivilliga (ex. attributen `class` och `id`)

Elementtyper

Kan generellt kategoriseras som

- Block-element
 - Fyller horisontellt (radbrytning före och efter)
 - Kan innehålla block-, inline-element eller innehåll
- Inline-element
 - Lika bred som innehållet (inga radbrytnigar)
 - Kan innehålla inline-element eller innehåll
- HTML 5 introducerar andra typer

Vanliga HTML-element

- Huvud, fot och navigation
- Sektioner och artiklar
- Generell avdelare (division)
- Rubriker och stycken
- Bilder
- Länkar

Huvud, fot och navigation

(Block-element)

```
<!-- Ett huvud -->  
<header>  
</header>  
  
<!-- Navigation -->  
<nav>  
</nav>  
  
<!-- En fot -->  
<footer>  
</footer>
```

En webbsida har ofta ett sidhuvud, en sidfot och en global meny

Kan vara header och footer för delar av en sida

Det kan finnas olika typer av navigation, t.ex. en global och en lokal

Sektioner och artiklar

(Block-element)

```
<!-- En sektion innehållandes två artiklar -->
<section>
  <!-- Första artikeln -->
  <article>
  </article>
  <!-- Andra artikeln -->
  <article>
  </article>
</section>
```

En sektion är en del av ett dokument (exempelvis ett kapitel)

En artikel är en komplett eller självständig informationspunkt
(exempelvis ett inlägg eller en tidningsartikel)

Dessa kan nästlas – det finns inga regler för hur det ska göras

Generell avdelare (division)

(Block-element)

```
<!-- Ett generellt innehåll -->  
<div>  
</div>
```

I brist på andra semantiska element

Avänd hellre *section*, *article*, *header*, *footer* eller *nav*

Rubriker och stycken

(Block-element)

```
<!-- Exempel på rubriker i tre nivåer -->  
<h1>HTML & CSS</h1>  
<h2>Olika typer av HTML-element</h2>  
<h3>Rubriker och stycken</h3>  
  
<!-- Exempel på ett stycke -->  
<p>Rubriker och stycken används främst för att märka upp text-innehåll</p>
```

Används främst för text-innehåll

Rubriker för olika nivåer (*h1* – *h6*)

Bilder

(Inline-element)

```
<!-- Bild från en lokal mapp -->  
  
  
<!-- Bild från en resurs på internet -->  

```

Värdet för `src`-attributet är bilden som ska visas (sökväg)

Värdet för `alt`-attribut beskriver bilden och ökar därmed användbarheten

Länkar

(Inline-element)

```
<!-- Länk till Google (notera title attributet) -->
<a href="http://google.com" title="Gå till Google">Google</a>

<!-- Länk till en sida i samma lokala mapp (kontakt.html) -->
<a href="kontakt.html" title="Gå till kontaktsidan">Kontakt</a>

<!-- Länk som öppnar en bild (notera att innehållet är ett bild-element) -->
<a href="path/to/images/cat.jpg" title="Visa katt bilden">
  
</a>
```

Mål-adressen är värdet för `href`-attributet

Representationen av länken (det som visas för användaren) klassas som innehåll och kan vara exempelvis en text eller bild

Tillgängliga element – några typer

- Sektioner (ex. sektioner och artiklar)
- Innehållshantering (ex. stycken och listor)
- Textnivå-semantik (ex.länkar, betoning)
- Inbäddad media (ex. bilder, video, ljud)
- Tabeller
- Formulär

Se W3C för element grupperade efter funktion

HTML-dokument

Samling av HTML-element

Element bildar ett träd av noder

DOM (Document Object Model) i webbläsaren

Kan sparas i en fil (oftast .html eller .htm)

Dokument-typ

```
<!-- Exempel: HTML4.01 -->  
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01 Frameset//EN"  
    "http://www.w3.org/TR/html4/frameset.dtd">  
  
<!-- Exempel: HTML5 -->  
<!DOCTYPE html>
```

Deklarerar vilket (märk-)språk dokumentet är skrivet i

Måste vara första raden i HTML-dokumentet

Rekommenderad standard är HTML5

Ett komplett HTML5-dokument

Måste innehålla nedanstående kod för att kunna valideras

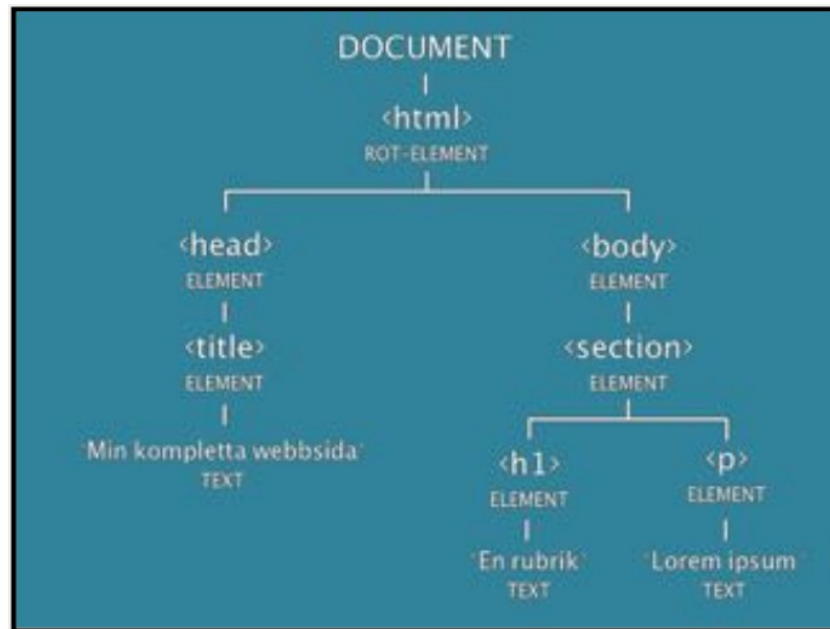
```
<!DOCTYPE html>
<html>
  <!-- head (metadata) -->
  <head>
    <title>Min kompletta webbsida</title>
  </head>
  <!-- body (sidans innehåll) -->
  <body>
  </body>
</html>
```

<head> innehåller *meta-data* (data om dokumentet)

<body> innehåller dokumentets synliga innehåll

Elementen bildar ett nod-träd

Dokument-trädet



Representeras av DOM (Document Object Model) i webbläsaren

Element i dokumentet

```
<!-- Ett p-element -->  
<p>  
  <!-- Ett barn till p-elementet -->  
  <em></em>  
</p>
```

Element som öppnas i andra element är barn till
det omgivande elementet

```
<!-- Felaktig stängning av element -->  
<!-- em-elementet öppnas i p-elementet men stängs är utanför -->  
<p><em></p></em>  
  
<!-- Korrekt stängning av element -->  
<!-- em-elementet öppnas och stängs i p-elementet -->  
<p><em></em></p>
```

Elementet måste stängas på samma nivå och gren som de öppnas

Kommentarer skrivs <!-- så här -->

Validera HTML



W3C Markup Validator

Säkerställer att syntaxen är korrekt och att reglerna efterlevs

HTML i korthet

- Märkspråk
 - Semantik (betydelse) och syntax (regler)
- HTML-element (taggar, attribut och innehåll)
 - Block- och inline-dokument
- HTML-dokument kan sparas i filer (.html, .htm)

Mer om HTML

- Referenser och fortsatt läsning
 - [W3C](#)
 - [WHATWG](#)
 - [Mozilla](#)
- Bra länkar
 - HTML-element: [W3C](#) och [Mozilla](#)
 - HTML-attribut: [W3C](#) och [Mozilla](#)
 - [Validera HTML \(W3C\)](#)

CSS – *Cascading Style Sheets*

- Beskriver presentation av dokument
 - För märkspråk (exempelvis HTML)
 - Design och layout
- Separerar presentation från innehåll

CSS-regler

Definierar hur element ska presenteras

CSS-regelns delar

```
/* Generell uppbyggnad av en CSS-regel */  
selektor { egenskap: värde; }  
  
/* Alla stycken (p-element) ska ha röd textfärg */  
p { color: red; }
```

Selektor anger vilka element som ska påverkas

Egenskap/värde-par beskriver hur elementet ska presenteras

Selektor

Selektorn avgör vilka element som påverkas.

Kan bland annat peka på ett elements:

- Typ (ex. *p*, *a*, *img*)
- Attribut (ex. *href*, *src*, *alt*)
 - Klass (attributet *class*)
 - ID (attributet *id*)
- Användarhandling/tillstånd (ex. *hover*, *active*, *visited*)

Selektorns syntax och notation

```
/* Selektor för element av typen section */  
section { egenskap: värde; }  
  
/* Selektor för alla h1-element med attributet class="minKlass" */  
/* Brukar benämnas som att elementet har en klass */  
h1[class~=minKlass] { egenskap: värde; }  
h1.minKlass { egenskap: värde; }  
  
/* Selektor för element med id mittId (id="mittId") */  
#mittId { egenskap: värde; }  
  
/* Selektor för alla p-element med klassen braText (class="braText") */  
p.braText { egenskap: värde; }
```

På elementnivå anges typen (ex. *h1*)

På klassnivå anges klassen med en punkt (ex. *.minKlass*)

ID anges med bräddgård/hashtag (ex. *#mittID*)

Selektor-syntax och relationer

```
/* Två selektorer där samma egenskap får samma värde */  
selektor1 { egenskap: värde; }  
selektor2 { egenskap: värde; }  
  
/* Samma som ovan fast kortare syntax */  
selektor1, selektor2 { egenskap: värde; }
```

```
/* BARN */  
  
/* Alla selektor2 två som finns under selektor1 i nodträdet */  
selektor1 selektor2 { egenskap: värde; }  
  
/* Alla selektor2 som är direkt barn till selektor1 */  
selektor1 > selektor2 { egenskap: värde; }
```

```
/* SYSKON */  
  
/* Alla selektor2 som är efterkommande syskon till selektor1 */  
selektor1 ~ selektor2 { egenskap: värde; }  
  
/* Alla selektor2 som är direkt efterkommande syskon till selektor1 */  
selektor1 + selektor2 { egenskap: värde; }
```

Egenskaper

```
/* Ändra textstorlek och -färg för alla stycken (p-element) */  
p {  
    font-size: 14px;  
    color: #0088cc; /* Textfärg som hexadecimal */  
}  
  
/* Gör en ram runt alla stycken (p-element) */  
p {  
    border: 1px solid black; /* 1 pixel tjock, ifylld, svart linje runt elementet */  
}
```

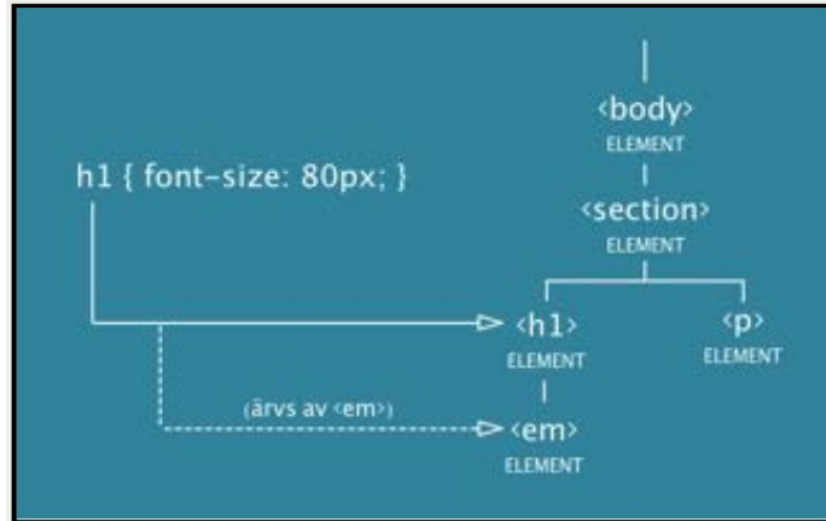
En selektor kan påverka flera egenskaper samtidigt

Flera selektorer kan peka på samma element

Egenskaper med värde beskrivs som `egenskap: värde`

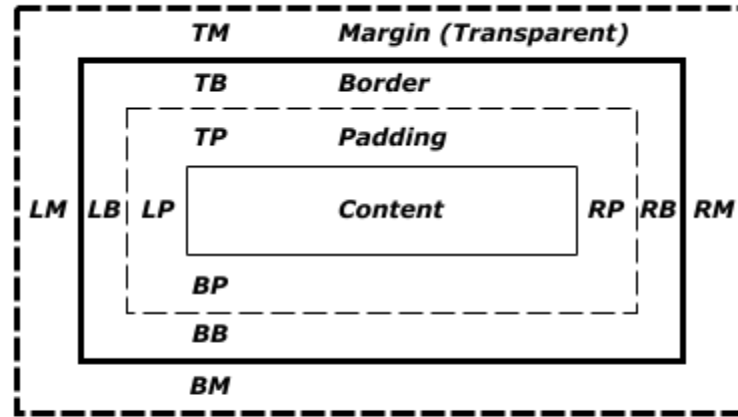
Vissa egenskaper kan ha flera värden, separerade med mellanslag

Arv



Vissa värden ärvs av barnen i nod-trädet

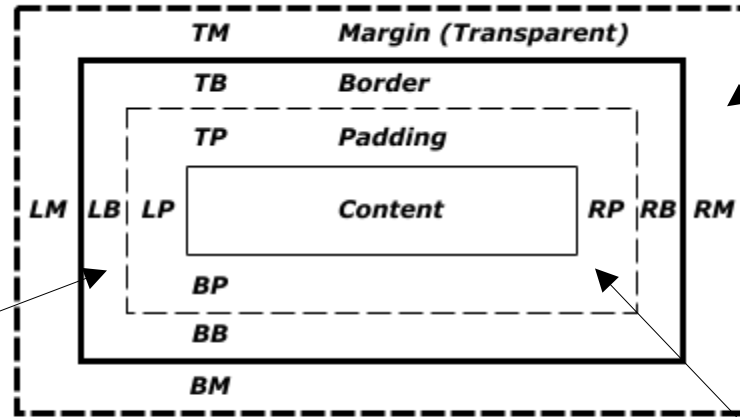
Box-modellen



- Margin edge
- Border edge
- - - Padding edge
- Content edge

Avgör elementets storlek

Boxens storlek



Margin: Avstånd till nästa element

Border: Eventuell ram runt elementet

Padding: Avstånd mellan innehållsarean och elementets ytterkant

- Margin edge
- Border edge
- - - Padding edge
- Content edge

Storlek = innehållsarea + padding + border + margin

Stilmall

Samling av CSS-regler

Kan sparas i separat fil (.css)

Placering av stilregler

CSS-regler kan skrivas på tre ställen:

- I elementens *style*-attribut
- Direkt i dokumentets huvud (*style*-elementet)
- I en extern fil som länkas in i dokumentets huvud

Placering av stilregler

```
<!-- Värde för style-attributet -->  
<p style="color: red;">Mitt stycke</p>
```

```
<head>  
  <!-- Direkt i dokumentets huvud -->  
  <style>  
    p.fetStil { font-weight: bold; }  
  </style>  
</head>
```

```
<head>  
  <!-- I en extern stilmall -->  
  <link rel="stylesheet" href="path/to/my-stylesheet.css">  
</head>
```

Prioritering av regler

```
/* Två regler för p som påverkar samma egenskap */  
/* Den första gäller för den är mer precis (3) */  
body p { font-size: 14px; }  
p { font-size: 24px; }  
  
/* Två regler för p som påverkar samma egenskap */  
/* Den andra gäller för den kommer sist (4) */  
p { color: red; }  
p { color: green; }
```

1. Var kommer regeln från (utvecklaren, användaren, webbläsaren)
2. Var finns regeln (elementet, head, extern fil, webbläsaren)
3. Vilken noggrannhet har regeln
4. Vilken ordning kommer reglerna i

CSS i korthet

- Regler för presentation av märkspråk (HTML)
- Selektorer pekar på element
 - Exempelvis element-typ, klass och id
- Egenskap/värde-par påverkar presentationen
- Vissa egenskaper ärvs

Mer om CSS

- Referenser och fortsatt läsning
 - [Mozilla](#)
 - [W3C](#)
- Bra länkar
 - [Getting started with CSS](#) (Mozilla)
 - [Selektorer](#) (Mozilla)
 - [Egenskaper](#) (Mozilla)
 - [Validera CSS](#) (W3C)

Distanskurser

Webbutveckling 1 på MAU (HTML, CSS & JS)

[Codecademy.com](https://www.codecademy.com) (HTML & CSS)

[Khanacademy.org](https://www.khanacademy.org) (HTML & CSS)