



Denne guide er oprindeligt udgivet på Eksperten.dk

## CSS - en gennemgang

**Denne artikel er til dem der ønsker at vide mere om hvad CSS er og hvad CSS kan bruges til hvad angår WWW.**

**Der er lidt CSS historie med samt små eksempler på de forskellige type af CSS.**

**Artiklen henvender sig ikke til dig der allerede kender CSS.**

Skrevet den **04. Feb 2009** af **foxmulder58** | kategorien **Design / Generelt** | ★★☆☆☆☆

CSS - Cascading Style Sheets hvad kan de?, og måske endnu vigtigere hvad kan de gøre for din webside. Det forsøger jeg i denne artikel at forklare.

### HTML 4.0 og Cascading Style Sheets

Det nok mest interessante ved HTML 4.0. er nok af det understøtter Cascading Style Sheets (CSS) fuldstændigt.

I HTML 3.2 eksisterede der en masse "indbyggede" formaterende elementer og attributter som skulle formatere designet. Men disse elementer og attributter gjorde det svært for webside udviklere at udvikle og vedligeholde store websider med hundredvis af dokumenter.

Med HTML 4.0. kom muligheden heldigvis at kunne anvende et Cascading Style Sheet, der således giver webside udvikleren muligheden for at rette i et sted i style Sheets hvilket så formaterer alle de sider der er berørt af denne.

HTML 4.0 giver Cascading Style Sheets muligheden for at vise browseren hvorledes et dokument skal fremvises. Til dette anvender man de 2 attributter ID , CLASS og STYLE.

## 2 Versioner af Cascading Style Sheets

Cascading stylesheets er kommet i to versioner, nemlig version 1 og version 2. Disse er oftest refereret til som CSS1 og CSS2.

CSS1 giver muligheden for at vælge fonte, farver og understøttes i mere eller mindre grad af Internet Explorer 4 og opefter, Netscape 4 og opefter samt Opera browseren.

CSS2 giver mulighed for at positionere forskellige elementer, at lave print sheets - dvs. et Style Sheet der kan anvendes til print. Endvidere kan CSS2 lave layout på tabeller (<tables>). CSS2 understøttes udmærket af browseren IE 5 og opefter.

Cascading Stylesheet kan altså ændre udseendet på alle elementer der findes på din webside. Hvert enkelt element på din side kan altså formateres efter lige dine ønsker med hensyn til placering, farve, skrift type osv.

## **Fordelen ved at anvende et Cascading Style Sheet**

### **Adskil indhold fra formatering**

Først og fremmest er fordelen at man kan skille formatering fra indholdet. Dette medvirker til at når man designer en webside har bedre kontrol, samt mere overskuelighed, eftersom tekstindhold og formatering ligger i hvert sit dokument. (medmindre man anvender et internt Cascading Style Sheet, men det kommer vi til.).

### **Nemmere redigering**

Har du en webside der består af rigtig mange HTML-sider med hver deres specifikke indhold (skriftstørrelse, skrifttype, farver osv.), skal du hvis du ikke anvender Cascading Style Sheets åbne hver enkelt dokument i din WYSIWYG editor og redigere hver enkelt side.

### **Hurtigere svartider på din side**

HTML er ikke designet til at fylde meget, men dine sider bliver jo tungere og tungere jo mere koder der kommer på den. Ved at anvende Cascading Style Sheets skal oplysningerne omkring formateringen kun hentes én enkelt gang, og nedsætter dermed overførselshastigheden på brugerens computer.

Efter at have set lidt på hvad Cascading Style Sheet er det godt at komme med et par eksempler på hvordan Cascading Style Sheet kan anvendes.

#### De 2 forskellige typer af stylesheets

Der findes 2 type af Cascading Style Sheets, det eksterne stylesheet, og de interne Cascading Style Sheet.

Lad os se på hvordan disse benyttes:

#### Det eksterne Cascading Style Sheet

Det eksterne style sheet er et helt separat dokument hvor der angives de formateringer som HTML-dokumenterne så henter og anvender.

Et eksternt Cascading Style Sheet er et eksternt dokument, og har altid filnavnet `.css` . Den eksterne fil kan indeholde ganske få formateringer og lange liste med forskellige formateringer de såkaldte `STYLES`.

Kort fortalt er der overhovedet ikke nogen begrænsninger for hvor mange eksterne Style Sheets et Html-dokument må linke til.

For at linke til et eksternt Style Sheet og anvende de formateringsindstillinger det nu måtte indeholde er du nødt ti at linke til dette. For at linke til et eksternt Style Sheet skal der placeres en kode i HTML-

dokumentets HEAD-kode.

Dette link ser således ud:

```
<link rel="stylesheet" type="text/css" href="mitStyleSheet.css">
```

Man angiver således med denne kode stien ned til der hvor man har sin .css liggende. I ovenstående eksempel hedder min .css fil *mitStyleSheet.css*. Koden *href* angiver således hvilken fil der er HTML-dokumentet skal formateres efter.

I eksempelet lagde jeg min .css fil i rodmappen for websiden. Det er oftest en god idé at placere dine style sheets i en mappe du så eksempelvis kan kalde **css**. Antager vi nu at du har lavet mappen **css** og vil linke dertil skal du selvfølgelig angive det i stien dertil. Derfor kommer koden i stedet for den overstående nu til at se således ud:

```
<link rel="stylesheet" type="text/css" href="css/mitStyleSheet.css">
```

Bemærk nu at vi har angivet at style sheet nu er angivet til at være i mappen *css*.

Her kan du så se hvordan hele HTML-dokumentet ser ud i sin helhed:

<NB!>Bemærk at jeg ikke har sat nogen form for DOCTYPE på min HTML eksempler!, dette skal du dog ikke undlade når du koder dine HTML-dokumenter i al almindelighed[/b]

### HTML-dokument der anvender et eksternt cascading Style Sheet

```
<html>
<head>
<link rel="stylesheet" type="text/css" href="css/mitStyleSheet.css">
</head>
<body>
Her kommer alt det indhold der skal formateres
</body>
</html>
```

[div]Det interne Cascading Style Sheet

Et internt Cascading Style Sheet er en række af formateringsoplysninger der gemmes inde i HTML-dokumentets **HEAD** kode. Da dette Style Sheet er angivet i ét dokument gælder formateringsreglerne således **kun** for dette HTML-dokument.

Lad os se hvordan et sådant internt Style Sheet ser ud.

```
<html>
<head>
<style type="text/css">
<! - -
body{
color:white;
```

```
background:blue;

}

- ->

</style>
</head>
<body>
Sidens indhold
</body>
</html>
```

I ovenstående eksempel kan du jo eventuelt prøve at copy/paste denne kode ind i din WYSIWYG-editor og se resultatet.

Vi har ovenstående HTML-eksempel angivet at al tekst skal være skrevet med farven hvid via reglen `color:white;` og at baggrundsfarven skal være blå via v.h.a reglen `[i] background:blue;`

Den sidste form for Style Sheet jeg vil vise dig kaldes et **inline style sheet**. Inline style sheets adskiller sig fra de 2 øvrige det eksterne og det interne, stylesheets, ved at alle reglerne for formateringen ikke er samlet ét sted men derimod er dette placeret i selv koden.

### Eksempelvis:

**<p style="color:blue"> Den tekst der står her er blå</p>**

I ovenstående eksempelvis har vi via et inline Style Sheet angivet at tagget `<p>` skal vise alt indhold af tekst med farven blå. Når du anvender et inline Style Sheet vælger du specifikt den kode som du ønsker at formatere. Det illustrerede Style inline Style Sheet gælder altså **ikke** for eventuelle andre `<p>` tags.

Her til sidst kunne jeg godt tænke mig at vise dig hvordan du kan anvende Cascading Style Sheet til at formatere en tom containerkode.

Til eksemplet har jeg anvendt tagget `DIV` der er en tom container koder idet `DIV` ikke kan anvendes uden at dette har en eller anden form for formateringsregel (i form ad et Cascading Style Sheet) eller en attribut.

I eksemplet anvender jeg 2 `DIV` tags til at formatere teksten udseende i form af farver, skrifttype og ikke mindst placering.

I mit første eksempel ønsker jeg at teksten skal have en blå farve, have fonten Verdana og placeres til venstre i browser vindue.

For at opnå dette anvender jeg en række regler der så skal formatere den tekst der er i `DIV`.

Jeg laver min første regel som jeg kalder for **afsnit1**

#### **CSS regel for afsnit 1**

```
#afsnit1{
font-family:verdana;
float:left
color:blue;
}
```

<br/>

Ved at indsætte tegnet "#" kan jeg angive at der er tale om et ID. ID`et er her altså afsnit 1 og alle de regler der er for #afsnit er altså :

[div]

font-family = Den font der skal anvendes, du kan selv angive flere fonte på én gang hvis du ønsker.

float:left = Placerer DIV tagget til venstre i browser vinduet

color:blue = Angiver at al teksten i DIV tagget skal vises i farven blå.

Jeg har lavet et mere, der næsten har de samme regler bortset fra at DIV her er placeret til højre og har en tekst farve i DIV der er rød. Du kan se koden for denne her:

### Regel for ID afsnit2

```
#afsnit2{
font-family:arial;
float:right;
color:red;
}
```

<br/>

[/div]

**Hele dokumentet med de 2 DIV formateringer kommer således til at se sådan ud i et internt style sheet.**

```
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01
Transitional//EN" "http://www.w3.org/TR/html4/loose.dtd">

<html>
<head>
<title>Untitled</title>
<style type="text/css">
<!--
#afsnit1{
font-family:verdana;
float:left;
color:blue;
}

#afsnit2{
font-family:arial;
float:right;
color:red;
}
-->
</style>
```

```
</head>
<body>
<div id="afsnit1">Her teksten for afsnit1
```

```
<div id="afsnit2">Her teksten for afsnit2[/div]
</body>
</html>
[/div]
```

For mere information om Cascading Style Sheets kan du se nedenstående URL:

```
http://www.w3.org/Style/CSS/
```

For en reference til CSS1 og CSS2 se nedestående URL`er:

```
http://www.zvon.org/xxl/css1Reference/Output/
```

```
http://www.w3schools.com/css/css\_reference.asp
```

Håber du har fået en bedre forståelse for hvad CSS er for en størrelse og hvad det kan gøre for dig.

Mvh  
Mads

#### **Kommentar af fjappe d. 12. Aug 2005 | 1**

Ok - artikel, men som milo, mener jeg også den er lidt overfladisk.

#### **Kommentar af milo d. 02. Aug 2005 | 2**

En smule overladisk synes jeg, men ikke dårlig.

dj-meisner: # er ikke "til en div". Du bør måske læse igen :)

#### **Kommentar af barbarbo d. 02. Aug 2005 | 3**

Udemærket gennemgang for den der ønsker at starte med at bruge CSS. Giver en god lille indføring i

emnet og peger på mere information

#### **Kommentar af htmlkongen d. 23. Dec 2006 | 4**

Udemærket artikel /Htmlkongen

#### **Kommentar af dj-meisner d. 31. Jul 2005 | 5**

god. lærte at # var til en div, som jeg ikke vidste for 10 min, siden da jeg selv sad og legede.

#### **Kommentar af rtfm\_now d. 01. Aug 2005 | 6**

Fint, men du 1. glemmer CSS3 | 2. "CSS2 understøttes udmærket af browseren IE 5 og opefter." er 90% lodret løgn, IE understøtter hat af CSS2, og ca. intet af CSS3. | 3.  
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN">  
er elendigt til CSS, da IE fucker op, og bruger sin egen standard for CSS (derved dør boxmodellen) hvis der ikke er angivet DTD link, find en flamewar med olebole, og i kan se rigtigt doctype.

#### **Kommentar af fasioide d. 26. Oct 2006 | 7**

Finder denne artikkel både ubrugelig og overfladisk

#### **Kommentar af just\_p d. 01. Mar 2006 | 8**

Helt ok, men overfladisk og gennemgangen på html.dk kan med fordel bruges som supplement for en nybegynder!