

Applicazione Ipermediali

2009/2010

Cascading Style Sheet

Politecnico di Milano



Obiettivi

Capire l'importanza della separazione tra contenuto e layout grafico.

Capire l'importanza di aderire agli standard.

Apprendere i CSS

- Selettori
- Proprietà
- Valori

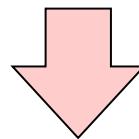
Imparare alcune tecniche di impaginazione

Il Problema

HTML permette di “formattare” il testo.

MA:

- Ha dei limiti, imposti dalla specifica del linguaggio
- Non permette di separare il contenuto dalla grafica

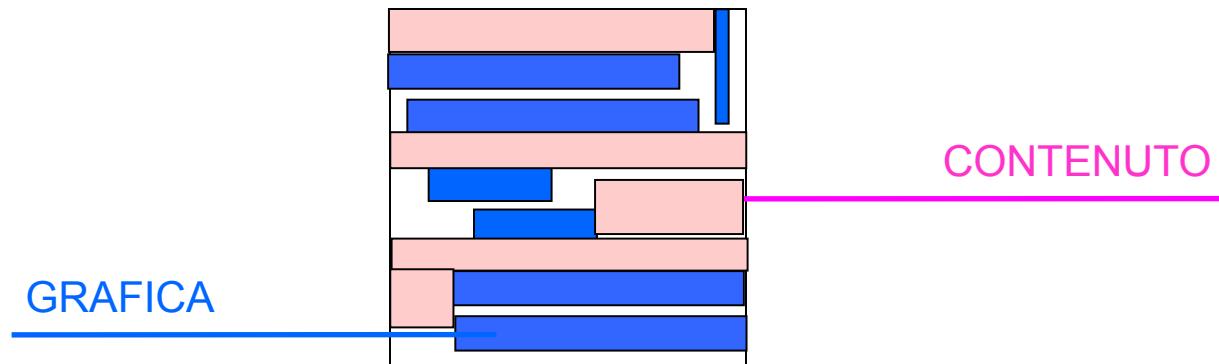


Se voglio modificare il contenuto DEVO modificare anche la grafica.

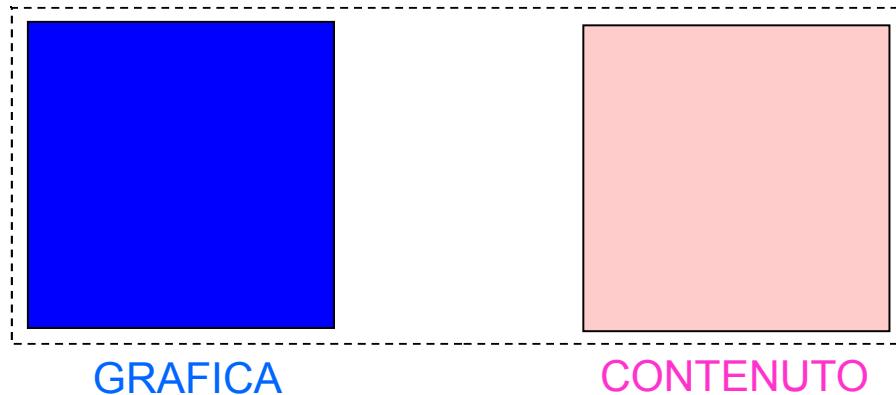
Se voglio modificare la grafica DEVO modificare anche il contenuto.

Il Problema

HTML:



HTML + CSS:



Esempio pratico:

<http://www.csszengarden.com/>

九九
空空
全全

Zen Garden

The Road to Enlightenment

Littering a dark and dreary road lay the past relics of browser-specific tags, incompatible DOMs, and broken CSS support.

Today, we must clear the mind of past practices. Web enlightenment has been achieved thanks to the tireless efforts of folk like the W3C, WaSP and the major browser creators.

So What is This About?

There is clearly a need for CSS to be taken seriously by graphic artists. The Zen Garden aims to excite, inspire, and encourage participation. To begin, view some of the existing designs in the list. Clicking on any one will load the style sheet into this very page. The code remains the same, the only thing that has changed is the external .css file. Yes, really.

Download the sample

[html file](#) and [css file](#)

HTML and CSS

CSS allows complete and total control over the style of a hypertext document. The only way this can be illustrated in a way that gets people excited is by demonstrating what it truly can accomplish visually.

Techniques have been demonstrated by structuralists and coders. Designers have yet to make their mark. This needs to change.

Participation

Graphic artists only please. You are modifying this page, so strong CSS skills are necessary, but the example files are commented well enough that even CSS novices can use them as starting points.

Please see the [CSS Resource Guide](#) for advanced tutorials and tips on working with CSS.

You may modify the style sheet in any way you wish, but not the HTML. This may seem daunting at first if you've never worked this way before, but follow the listed links to learn more, and use the sample files as a guide.

Download the sample [html file](#) and [css file](#) to work on a copy locally. Once you have completed your masterpiece (and please, don't submit half-finished work) upload your .css file to a web server under your control. Send us a link to the file and if we choose to use it, we will spider the associated images. Final submissions will be placed on our server.

The Beauty of CSS Design



Select a design:
 Retro Theater by Eric Rögård
 Lily Pond by Rose Thorogood
 Icicle Outback by Timo Virtanen
 Zen Army by Carl Desamond
 The Original by Joachim Schüller
 Floral Touch by Jadas Jimmy
 Elegance in Simplicity by Mani Shetlar
 Dazzling Beauty by Deny Sir Suprayono

archives:
[next designs >](#)
[View All Designs](#)

resources:
[View This Design's CSS](#)
[CSS Resources](#)
[FAQ](#)

cssZENGARDEN

The Beauty of CSS Design

A demonstration of what can be accomplished visually through CSS-based design. Select any style sheet from the list to load it into this page.
[Download the sample HTML file](#) and [CSS file](#).

Select a design

RETRO THEATER
 by Eric Rögård
LILY POND
 by Rose Thorogood
ICICLE OUTBACK
 by Timo Virtanen
ZEN ARMY
 by Carl Desamond
THE ORIGINAL
 by Joachim Schüller
FLORAL TOUCH
 by Jadas Jimmy
ELEGANCE IN SIMPLICITY
 by Mani Shetlar
DAZZLING BEAUTY
 by Deny Sir Suprayono

The Road to Enlightenment

Littering a dark and dreary road lay the past relics of browser-specific tags, incompatible DOMs, and broken CSS support.

Today, we must clear the mind of past practices. Web enlightenment has been achieved thanks to the tireless efforts of folk like the W3C, WaSP and the major browser creators.

The css Zen Garden invites you to relax and meditate on the important lessons of the masters. Begin to see with clarity. Learn to use the (yet to be) time-honored techniques in new and invigorating fashion. Become one with the web.

So What Is This About?



cssZENGARDEN

THE BEAUTY OF CSS DESIGN

[Download the sample HTML file and CSS file](#)

SELECT A DESIGN

Retro Theater by Eric Rögård
 Lily Pond by Rose Thorogood
 Icicle Outback by Timo Virtanen
 Zen Army by Carl Desamond
 The Original by Joachim Schüller
 Floral Touch by Jadas Jimmy
 Elegance in Simplicity by Mani Shetlar
 Dazzling Beauty by Deny Sir Suprayono

Littering a dark and dreary road lay the past relics of browser-specific tags, incompatible DOMs, and broken CSS support.

Today, we must clear the mind of past practices. Web enlightenment has been achieved thanks to the tireless efforts of folk like the W3C, WaSP and the major browser creators.

The css Zen Garden invites you to relax and meditate on the important lessons of the masters. Begin to see with clarity. Learn to use the (yet to be) time-honored techniques in new and invigorating fashion. Become one with the web.

THE ROAD TO ENLIGHTENMENT

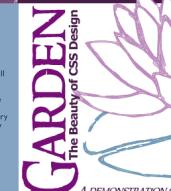
[Download the sample HTML file and CSS file](#)

BENEFITS

Why participate? For recognition, inspiration, and a resource we can all refer back to. We can all benefit from CSS-based design. This is sorely needed, even today. More and more major sites are using CSS, but not enough have. One day this gallery will be a historical curiosity; that day is not yet.

REQUIREMENTS

What can be done with CSS?



GARDEN

THE BEAUTY OF CSS DESIGN

A DEMONSTRATION OF WHAT CAN BE

SO WHAT IS THIS ABOUT?

There is clearly a need for CSS to be taken seriously by graphic artists. The Zen Garden aims to excite, inspire, and encourage participation. To begin, view some of the existing designs in the list. Clicking on any one will load the style sheet into this very page. The code remains the same, the only thing that has changed is the external .css file. Yes, really.

CSS allows complete and total control over the style of a hypertext document. The only way this can be illustrated in a way that gets people excited is by demonstrating what it truly can accomplish visually.

Techniques have been demonstrated by structuralists and coders. Designers have yet to make their mark. This needs to change.

Graphic artists only please. You are modifying this page, so strong CSS skills are necessary, but the example files are commented well enough that even CSS novices can use them as starting points.

Please see the [CSS Resource Guide](#) for advanced tutorials and tips on working with CSS.



Perché?

Direttiva W3C(World Wide Web Consortium): standard!

Ottimizzazione del codice

Ottimizzazione delle risorse

- Il grafico fa la grafica
- Il content manager fa i contenuti
- Il webmaster
 - Fonde grafica e contenuti
 - Mantiene separazione
 - Farcisce di tecnologia

Separazione di contenuto da aspetto grafico non pone vincoli al canale (MULTICANALITA')

Cascading Style Sheet

1996: definizione dei fogli di stile da parte del W3C (CSS Level 1 o CSS-1)

Dal 1996 al 2000: scarso supporto da parte dei browser

Dal 2000: **CSS-1** ben supportato da:

- Explorer 5 e maggiori
- Opera 5 e maggiori
- Netscape 6 e maggiori
- Safari

Cascading Style Sheet

CSS2 rilasciato da W3C nel 2002.

Problema della compatibilità tra i browser.

<http://www.quirksmode.org/css/contents.html>

Vantaggi degli standard sul web

- Compatibilità (???)
- Indipendenza dal dispositivo
- Accessibilità
- Sviluppo e manutenzione (uno standard “non muore in fretta”)
- Leggerezza del codice
- Motori di ricerca (di nuovo la compatibilità)

L'albero del documento

Codice XHTML composto da oggetti/elementi.

Questi oggetti possono essere vuoti o contenerne di altri.

```
<radice>
  <elemento1>
    <elemento4></elemento4>
    <elemento5></elemento5>
  </elemento1>
  <elemento2></elemento2>
  <elemento3></elemento3>
</radice>
```

L'albero del documento

Si vengono quindi a crearsi diversi rapporti di parentela tra i vari oggetti.

“PROGENITORE”:

- “radice” è progenitore di tutti gli altri elementi

```
<radice>
```

```
<elemento1>
  <elemento4></elemento4>
  <elemento5></elemento5>
</elemento1>
<elemento2></elemento2>
<elemento3></elemento3>
```

```
</radice>
```

L'albero del documento

“DISCENDENTE”

- Ogni elemento (tranne “radice”) è discendente di “radice”

```
<radice>
```

```
  <elemento1>
    <elemento4></elemento4>
    <elemento5></elemento5>
  </elemento1>
  <elemento2></elemento2>
  <elemento3></elemento3>
```

```
</radice>
```

L'albero del documento

“PADRE”

- “radice” è padre di “elemento1”, “elemento2”, “elemento3”
- “elemento1” è padre di “elemento4”, “elemento5”

```
<radice>
  <elemento1>
    <elemento4></elemento4>
    <elemento5></elemento5>
  </elemento1>
  <elemento2></elemento2>
  <elemento3></elemento3>
</radice>
```

L'albero del documento

“FIGLIO”

- “elemento1”, “elemento2”, “elemento3” sono figli di “radice”
- “elemento4”, “elemento5” sono figli di “elemento1”

```
<radice>
  <elemento1>
    <elemento4></elemento4>
    <elemento5></elemento5>
  </elemento1>
  <elemento2></elemento2>
  <elemento3></elemento3>
</radice>
```

L'albero del documento

“FRATELLO”

- “elemento4” è fratello di “elemento5”
 - Ma anche: “elemento5” è fratello di “elemento4”
- “elemento1”, “elemento2”, “elemento3” sono fratelli.

```
<radice>
  <elemento1>
    <elemento4></elemento4>
    <elemento5></elemento5>
  </elemento1>
  <elemento2></elemento2>
  <elemento3></elemento3>
</radice>
```

Tipo di elemento

3 “tipi” di elementi:

- “BLOCK LEVEL”
 - Identifica un’unità di contenuto
 - <p>, <div>, <h1..h6>,
- “INLINE LEVEL”
 - All’interno di un’unità di contenuto, specifica caratteristiche
 - <a>, , , ...
- “REPLACED ELEMENT”
 - Sono oggetti di cui si sa a priori la dimensione predefinita
 - , <textarea>, <select>, ...

CSS come “motore di regole a cascata”

C’è un motore di rendering “a regole”

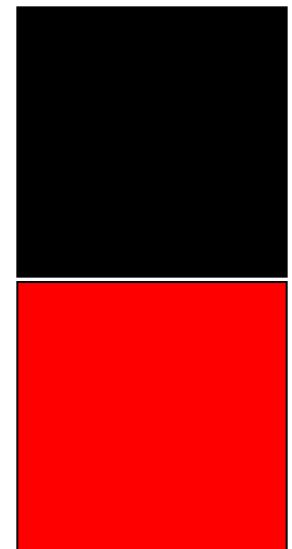
Ogni regola definisce una proprietà di un (gruppo di) elemento/i

Le regole sono eseguite “in cascata”, una dopo l’altra. Si possono naturalmente verificare “conflitti”:

- Esempio: devo dipingere un muro.
- R1: dipingo il muro di rosso
- R2: dipingo il muro di nero
- A seconda dell’ordine di esecuzione sequenziale delle regole ho due risultati diversi:

R1 → R2

R2 → R1



Composizione di una regola

Una regola è composta da:

- **SELETTORE**: ci fa capire a quale/i elemento/i si riferisce la regola
- **PROPRIETA'**: è una caratteristica dell'oggetto che la regola vuole modificare
- **VALORE**: è un valore numerico/simbolico associato ad un'unità di misura. E' lo stato desiderato della proprietà

ES: R1 = Dipingi il muro di bianco

- Selettore: “il muro”
- Proprietà: “colore del muro”
- Valore: “bianco”

Chi esegue la regola? Il Browser!



Esempio (in action!)

Coloriamo lo sfondo!

```
body {  
    background: #FFFFFF;
```

Selettore

Proprietà

Valore

Foglio di stile e commenti

Un foglio di stile è un insieme di regole.

All'interno di un foglio di stile è possibile inserire dei commenti

- /* questo è un commento */
- // questo è un commento, **ma viene interpretato come regola da Internet Explorer 6!!!**
 - Utile per risolvere la compatibilità.

I commenti /* */ possono essere distribuiti su più righe

I selettori

* è selettore universale, seleziona TUTTI gli elementi dell'albero del documento

Il selettore di tipo è costituito dal nome di un qualsiasi tipo di elemento XHTML

- h1 { ... }
- p { ... }
- body { ... }

Tutte le occorrenze dei selettori nel documento XHTML saranno influenzate dalle regole.

I selettori: le classi

Il selettore di classe serve a selezionare oggetti in qualche modo omogenei tra loro.

- .fontBoldBlack { ... }
- .boxRed { ... }
- .titleGreen { ... }

Nel codice XHTML si dovrà esplicitare quali elementi fanno parte delle classi:

```
<p class="titleGreen">Titolo</p>
...
<div class="boxRed">BOX 1</div>
<div class="boxRed">BOX 2</div>
```

I selettori: le classi

Esempi di selettori di classe:

- 1) `.notizia { ... }`
- 2) `h2.notizia { ... }`
- 3) `p.notizia { ... }`

- 1) Viene applicato a TUTTI gli elementi con `class="notizia"`
- 2) Viene applicato SOLO agli elementi `<h2>` con `class="notizia"`
- 3) Viene applicato SOLO agli elementi `<p>` con `class="notizia"`

La regola “di classe” comincia sempre con un “.”

I selettori: le classi

E' possibile associare più classi ad un unico elemento, separandole con lo spazio

```
<p class="notizia notiziaImportante">  
    Corpo della notizia  
</p>
```

I selettori: gli identificatori

Gli identificatori sono i selettori complementari rispetto alle classi, ovvero servono a distinguere oggetti unici e irripetibili.

```
<p id="titolo">Questo è il titolo</p>
<p id="sottotitolo">Sottotitolo</p>
<p class="paragrafo">...</p>
<p class="paragrafo">...</p>
```

La regola CSS corrispondente ad un identificatore comincerà con il simbolo **#**

```
#titolo {  
    ...  
}
```

I selettori: pseudo-classi

Esistono alcuni selettori “di comodo”, universali, detti PSEUDO-CLASSI

:first-child

- Individua il primo figlio di un “padre”

Pseudo-classi dei collegamenti

- :link → si applica a tutti i collegamenti non visitati
- :visited → si applica a tutti i collegamenti visitati

Pseudo-classi dinamiche

- :hover → si applica quando il mouse è sopra l’oggetto
- :focus → si applica quando il mouse ha selezionato (click) un elemento
- :active → si applica quando un oggetto diventa attivo (esempio: link cliccato)

I selettori: pseudo-elementi

Oltre alle pseudo-classi esistono anche gli pseudo-elementi, una sorta di “estensione” dell’albero del documento

:first-line

:first-letter

:before e :after ***

```
p:first-line {  
    font-weight: bold; /*testo in grassetto*/  
}
```

*** non supportato da IE6

I selettori: discendenti, fratelli, figli

```
<h1>Lorem <strong>Ipsum mea sale</strong> <p>Decore  
<strong>aperiam blandit</strong> quo ei.</p> iisquae laoreet  
ei </h1>  
<p>Decore <strong>aperiam blandit</strong> quo ei.</p>  
<p>Stet invenire nam no.</p>  
<p>Sint oblique... </p>
```

Discendenti (a qualunque livello)

```
h1 strong { ... }
```

I selettori: discendenti, fratelli, figli

```
<h1>Lorem <strong>Ipsum mea sale</strong> <p>Decore  
<strong>aperiam blandit</strong> quo ei.</p> iisquae laoreet  
ei </h1>  
<p>Decore <strong>aperiam blandit</strong> quo ei.</p>  
<p>Stet invenire nam no.</p>  
<p>Sint oblique... </p>
```

Discendenti (a qualunque livello)

```
h1 strong { ... }
```

Figli (subito sotto di un livello)

```
h1 > strong { ... }
```

I selettori: discendenti, fratelli, figli

```
<h1>Lorem <strong>Ipsum mea sale</strong> <p>Decore  
<strong>aperiam blandit</strong> quo ei.</p> iisquae laoreet  
ei </h1>  
<p>Decore <strong>aperiam blandit</strong> quo ei.</p>  
<p>Stet invenire nam no.</p>  
<p>Sint oblique... </p>
```

Discendenti (a qualunque livello)

```
h1 strong { ... }
```

Figli (subito sotto di un livello)

```
h1 > strong { ... }
```

Fratelli (immediatamente precedenti)

```
h1 + p { ... }
```

I selettori: attributi e raggruppamenti

Come le classi e gli identificatori, ogni attributo può essere selezionato:

```
<a href=www.google.it title="google.com">Google!</a>
```

```
a[title=google.com] {  
    ...  
}
```

Posso associare più comportamenti a diversi selettori, creando regole composte:

```
h1, #id, p.titolo {  
    ...  
}
```

La cascata

Due o più regole, anche con selettori diversi, possono riferirsi allo stesso oggetto.

Quando due regole si applicano allo stesso oggetto e cercano di modificare la medesima proprietà, si applica la regola della cascata.



1) Si calcola la **specificità** (ABC) del selettore

A = n. di selettori di tipo ID

B = n. di selettori di classe/pseudo-classe/attributo

C = n. di selettori di tipo

2) Vince la regola con maggior specificità

3) In caso di pareggio, vince la regola scritta DOPO.

La specificità: esempi

Selettore	A	B	C	ABC	S
*	0	0	0	000	0
a	0	0	1	001	1
p a	0	0	2	002	2
p a:link	0	1	2	012	12
body p.classe a:link	0	2	3	023	23
p#id	1	0	1	101	101

Direttiva !important

Indipendentemente dal selettore, singole dichiarazioni possono assumere massima priorità se seguite dalla direttiva **!important**

```
p {  
    background: #FFFFFF;          /* sfondo bianco */  
    color: #000000 !important;    /* colore del testo nero */  
}  
  
p#titolo {  
    background: #EEAA44;          /* sfondo arancio */  
    color: #0000FF;               /* colore del testo blu */  
}
```

I valori

Alle varie proprietà CSS possono essere assegnati diversi tipi di valori

- “inherit”
- “auto”
- Numeri
- Lunghezze
- Valori percentuali
- URI
- Colori
- Misure assolute e relative
- Famiglie di caratteri
- Parole chiave contestuali alla proprietà

I valori: inherit

La parola chiave `inherit` indica che il valore da assegnare alla proprietà deve essere *ereditato* da un elemento progenitore per il quale la proprietà è stata specificata.

```
.foo {  
    background-color: white;  
    color: black;  
}  
  
.bar {  
    background-color: inherit;  
    color: inherit;  
    font-weight: normal;  
}
```

```
<div class="foo">  
    <p class="bar">  
        Hello, world.  
        This is a very short  
        paragraph!  
    </p>  
</div>
```

I valori: auto

Il valore auto indica che il motore di rendering dovrà utilizzare il valore di default della proprietà a cui si riferisce.

Il valore predefinito di una proprietà DIPENDE dall'oggetto a cui si applica la proprietà.

CENTRARE OGGETTI

- P.blocktext { margin-left: auto; margin-right: auto; width: 6em } ... <P class="blocktext">This rather...

```
P.blocktext {  
    margin-left: auto;  
    margin-right: auto;  
    width: 6em  
}
```

```
<P class="blocktext">This rather...
```

I valori: numeri e lunghezze

Alcune proprietà ammettono come valori semplicemente dei numeri

Le lunghezze sono costituite da un numero intero/reale a cui deve essere associata **ESPLICITAMENTE** un'unità di misura

- Unità di Misura Relative

- em: dimensione del carattere (dipende dal browser)
- ex: altezza del carattere 'x'
- px: pixel dello schermo

- Unità di Misura Assolute

- in: pollici (1 inch = 2,54 cm)
- cm: centimetri
- mm: millimetri
- pt: punti (nei CSS, 1 pt = 1/72 in)
- pc: picas (1 pc = 12pt)

I valori: le percentuali

Le percentuali sono un numero decimale ($1 < \text{numero} < 100$) seguito dal simbolo %.

Le percentuali hanno bisogno di un riferimento, che dipende dalla proprietà in questione.

```
#titolo {  
    width: 10 %;  
    color: black;  
}
```



In questo caso, se ci si riferisce ad un BLOCK element, esso assumerà larghezza pari al 10% del suo CONTENITORE.

I valori: URI

Alcune proprietà necessitano di avere a disposizione risorse disponibili da qualche parte nel web.

Esempio: lo sfondo

```
p {  
background-image: url(smallPic.jpg);  
background-repeat: repeat;  
}
```

This <p> has a background image repeating in both directions (default repeat). If you use this option, make sure that your image was designed to be repeated.

Esempi di URI:

- url("http://website/file");
- url("/file"); url("../file"); → relative path!

I valori: i colori

000000	000033	000066	000099	0000CC	0000FF
003300	003333	003366	003399	0033CC	0033FF
006600	006633	006666	006699	0066CC	0066FF
009900	009933	009966	009999	0099CC	0099FF
00CC00	00CC33	00CC66	00CC99	00CCCC	00CCFF
00FF00	00FF33	00FF66	00FF99	00FFCC	00FFFF
330000	330033	330066	330099	3300CC	3300FF
333300	333333	333366	333399	3333CC	3333FF
336600	336633	336666	336699	3366CC	3366FF
339900	339933	339966	339999	3399CC	3399FF
33CC00	33CC33	33CC66	33CC99	33CCCC	33CCFF
33FF00	33FF33	33FF66	33FF99	33FFCC	33FFFF
660000	660033	660066	660099	6600CC	6600FF
663300	663333	663366	663399	6633CC	6633FF
666600	666633	666666	666699	6666CC	6666FF
669900	669933	669966	669999	6699CC	6699FF
66CC00	66CC33	66CC66	66CC99	66CCCC	66CCFF
66FF00	66FF33	66FF66	66FF99	66FFCC	66FFFF
990000	990033	990066	990099	9900CC	9900FF
993300	993333	993366	993399	9933CC	9933FF
996600	996633	996666	996699	9966CC	9966FF
999900	999933	999966	999999	9999CC	9999FF
99CC00	99CC33	99CC66	99CC99	99CCCC	99CCFF
99FF00	99FF33	99FF66	99FF99	99FFCC	99FFFF
CC0000	CC0033	CC0066	CC0099	CC00CC	CC00FF
CC3300	CC3333	CC3366	CC3399	CC33CC	CC33FF
CC6600	CC6633	CC6666	CC6699	CC66CC	CC66FF
CC9900	CC9933	CC9966	CC9999	CC99CC	CC99FF
CCCC00	CCCC33	CCCC66	CCCC99	CCCCCC	CCCCFF
CCFF00	CCFF33	CCFF66	CCFF99	CCFFCC	CCFFFF
FF0000	FF0033	FF0066	FF0099	FF00CC	FF00FF
FF3300	FF3333	FF3366	FF3399	FF33CC	FF33FF
FF6600	FF6633	FF6666	FF6699	FF66CC	FF66FF
FF9900	FF9933	FF9966	FF9999	FF99CC	FF99FF
FFCC00	FFCC33	FFCC66	FFCC99	FFCCCC	FFCCFF
FFFF00	FFFF33	FFFF66	FFFF99	FFFFCC	FFFFFF

Color	Color HEX	Color RGB
	#000000	rgb(0,0,0)
	#FF0000	rgb(255,0,0)
	#00FF00	rgb(0,255,0)
	#0000FF	rgb(0,0,255)
	#FFFF00	rgb(255,255,0)
	#00FFFF	rgb(0,255,255)
	#FF00FF	rgb(255,0,255)
	#C0C0C0	rgb(192,192,192)
	#FFFFFF	rgb(255,255,255)

Codici di colore

216 “Web Safe Palette”

150 Cross Browser

Esadecimale

- #FFFFFF (RRGGBB)
- #456 (RGB)
- $\text{rgb}(0, 4, 5); \text{rgb}(40\%, 60\%, 0\%);$

I valori: misure assolute e relative (per i caratteri)

Le misure assolute indicano una dimensione di carattere predefinita:

- xx-small;
- x-small;
- small;
- medium;
- large;
- x-large;
- xx-large;

Le misure relative invece hanno solo due valori e si riferiscono alla misura dell'elemento padre

- larger
- smaller

Le proprietà

Dipendono dall'oggetto di riferimento

Devono essere associate ad un valore

Possono avere valori multipli

Problemi di compatibilità

<http://www.htmldog.com/reference/cssproperties/>

<http://www.w3.org/TR/CSS21/propidx.html>

(elenco completo e dettagliato)

Le proprietà: Texts e Fonts

font

font-family

font-size

font-weight

font-style

font-variant

line-height

letter-spacing

word-spacing

text-align

text-indent



Le proprietà: Texts e Fonts

color

background-color

background

background-image

background-repeat

background-position

background-attachment

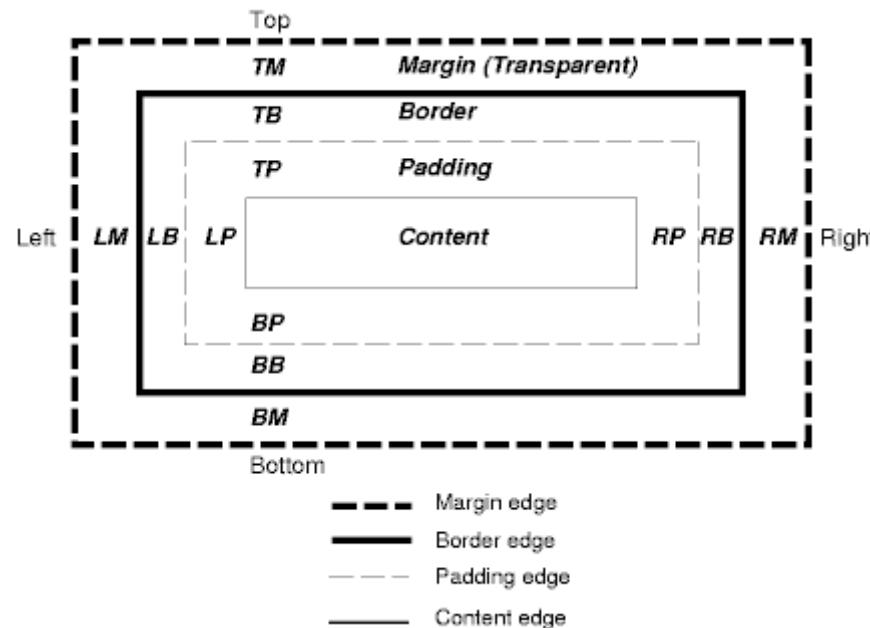
text-transform

vertical-align

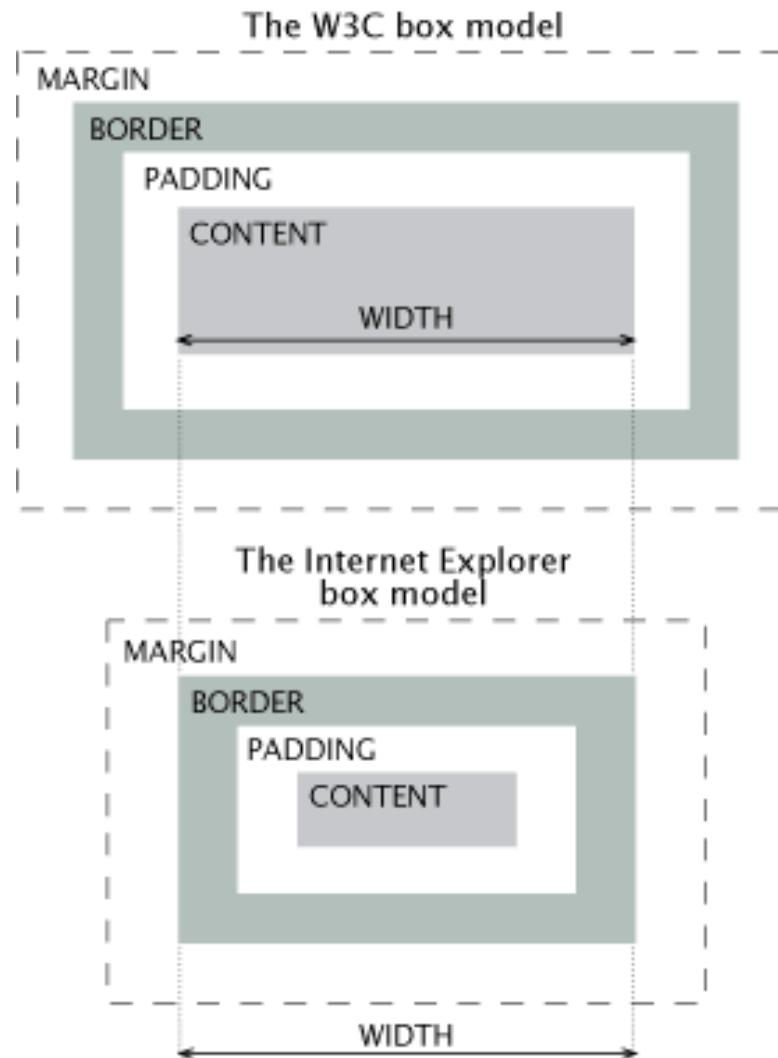
white-space

text-decoration

Le proprietà: il Box Model



Il Box Model secondo IE6



Le proprietà: The Box Model

padding,

- padding-top, padding-right, padding-bottom, padding-left

border,

- border-top, border-right, border-bottom, border-left

border-style,

- border-top-style, border-right-style, border-bottom-style, border-left-style

border-color,

- border-top-color, border-right-color, border-bottom-color, border-left-color

border-width,

- border-top-width, border-right-width, border-bottom-width, border-left-width

Le proprietà: The Box Model

margin, margin-top, margin-right, margin-bottom, margin-left

width

height

min-width

max-width

min-height

max-height

outline

outline-style

outline-color

outline-width

Le proprietà: Posizione e Visualizzazione

position

top

right

bottom

left

clip

overflow

z-index

float

clear

display

visibility

Le proprietà: Liste

list-style

list-style-type

list-style-image

list-style-position

Le proprietà: Tabelle

table-layout

border-collapse

border-spacing

empty-cells

caption-side

Le proprietà: Contenuto generato

content

counter-increment

counter-reset

quotes

Le proprietà: Paginazione e altre..

page-break-before

page-break-after

page-break-inside

orphans

widows

cursor

direction

unicode-bidi