

CSS

CASCADING STYLE SHEET

Alberto Ferrari

Cascading Style Sheet

- I fogli di stile a cascata (detti anche semplicemente fogli di stile) vengono usati per definire la rappresentazione di documenti HTML, XHTML e XML.
- L'introduzione dei fogli di stile si è resa necessaria per separare i contenuti dalla formattazione.

Wikipedia

Separazione contenuto/layout

- I tag html erano stati in origine studiati per definire il contenuto di un documento html
- Il layout del documento doveva essere preso in carica dal browser, senza usare tag di formattazione

Sintassi Css

- La più semplice regola css è composta di tre parti: un selettore, una proprietà ed un valore:
`selettore{proprietà: valore}`
- Il selettore può essere l'elemento html che si vuole definire
- La proprietà è l'attributo che si vuole cambiare
- Alla proprietà è associato un valore
- La proprietà e il valore sono separati da due-punti e racchiusi tra parentesi graffe
`body {color: black}`

Come inserire un foglio di stile

- Fogli di stile esterni
- Fogli di stile interni
- Stile inline

Fogli di stile esterni

- Foglio esterno: ideale se l'applicazione ha molte pagine
 - Si può cambiare l'aspetto di un intero sito modificando un solo file
 - Ogni pagina deve essere collegata al foglio di stile
Tag `<link>` all'interno della sezione head
- ```
<head>
<link rel="stylesheet" type="text/css" href="miostile.css" />
</head>
```
- Il browser leggerà le definizioni di stile dal file `miostile.css`, e formatterà il documento di conseguenza

## Scrivere fogli di stile

- Un foglio di stile esterno può essere scritto in qualsiasi editor di testo
- Il file non deve contenere alcun tag html
- Il foglio di stile deve essere salvato con estensione .css
- Un esempio di foglio di stile:

```
body {background-image: url("immagini/miosfondo.gif")}
p {margin-left: 20px}
```

## Fogli di stile interni

- Un foglio di stile interno può essere usato quando un documento html ha uno stile unico
- Si definisce nella sezione head con il tag <style>

```
<head>
<style type="text/css">
hr {color: red}
p {margin-left: 20px}
body {background-image: url("immagini/foto.jpg")}
</style>
</head>
```

## Stile inline

- Uno stile inline fa perdere molti dei vantaggi dei fogli di stile, mischia il contenuto con la presentazione
  - Metodo da usare quando uno stile deve essere applicato ad una sola occorrenza di un elemento in un documento
- Per definire uno stile inline si usa l'attributo style dell'elemento da formattare
  - L'attributo style può contenere qualsiasi proprietà css

```
<p style="color: green; margin-left: 20px">
Testo del paragrafo </p>
```

## Ordine di priorità (cascata)

- Gli stili possono essere specificati in un singolo elemento html, nella sezione <head> di una pagina html, o in un file css esterno
- Quale stile sarà usato quando è specificato più di uno stile per un elemento html?
- Tutti gli stili vengono raccolti a cascata in un nuovo foglio di stile virtuale secondo le seguenti regole
  - Default del browser (priorità più bassa)
  - Foglio di stile esterno
  - Foglio di stile interno (nella sezione <head>)
  - Stile inline (nell'elemento html – priorità più alta)

## Sintassi Css - Regole

- Se il valore è composto da più parole, bisogna racchiuderlo tra virgolette doppie
- Se si vuole specificare più di una proprietà, bisogna separarle con un punto-e-virgola
- L'esempio mostra come definire un paragrafo con allineamento centrato e colore del testo rosso

```
p {
 text-align: center;
 color: red;
 font-family: "sans serif";
}
```

## Sintassi Css - Raggruppare i selettori

- Si possono raggruppare i selettori: bisogna separare i selettori con una virgola
- Nell'esempio gli elementi di titolo sono raggruppati; ogni elemento di titolo sarà verde

```
h1, h2, h3, h4, h5, h6 {
 color: green;
}
```

## Selettore di classe

- Con il selettore di classe si possono definire stili diversi per uno stesso tipo di elemento html
- Es. due tipi di paragrafo: uno allineato a destra e uno allineato al centro

```
p.firma {text-align: right}
p.importante {text-align: center}
```

## Use dei selettori di classe

- Bisogna impostare l'attributo class nel documento html

```
<p class="firma">
 Paragrafo che sarà allineato a destra
</p>
<p class="importante">
 Paragrafo che risulterà centrato.
</p>
```

## Classe

- Si può omettere il nome del tag nel selettore per definire uno stile da applicare a tutti gli elementi html di una certa classe
- Es. Allineare al centro tutti gli elementi di classe importante
- Sia l'elemento h1 che p sono di classe importante
- Entrambi gli elementi seguiranno le regole del selettore .importante

```
<h1 class="importante">
 Titolo centrato
</h1>
<p class="importante">
 Anche questo paragrafo centrato
</p>
```

## Selettore id

- Il selettore id è diverso dal selettore class, mentre un selettore class può applicarsi a molti elementi su una pagina, il selettore id si applica ad un solo elemento
- Un attributo id deve essere unico all'interno del documento
- La regola di stile qui sotto sarà applicata ad un elemento p che ha il valore di id posto a para1
- La regola di stile qui sotto sarà applicata all' elemento avente come id il valore xyz

```
p#para1 {
 text-align: center;
 color: red;
}
#xyz {color: green}
<h1 id="xyz">Esempio testo</h1>
```

## Commenti

- I commenti vengono ignorati dal browser
- Un commento comincia con /\* e termina con \*/
- /\* commento\*/

```
p {
 text-align: center;
 /* altro commento */
 color: black;
 font-family: arial;
}
```

## Sfondo

- background-color – Imposta il colore di sfondo di un elemento
- background-image – Imposta una immagine come sfondo – url(...)
- background-repeat – Imposta la ripetizione di una immagine di sfondo – repeat, repeat-x, repeat-y, no-repeat
- background-attachment – Immagine di sfondo fissa o che scorre con il testo – scroll, fixed
- background-position – Posizione di partenza di una immagine di sfondo – top left, top center, top right, center left, center center, center right, bottom left, bottom center, bottom right, x-% y-%, x-pos y-pos
- background – Imposta tutte assieme le proprietà dello sfondo

## Testo

- **color** – Colore del testo
- **direction** – Direzione del testo – ltr, rtl
- **text-align** – Allineamento – left, right, center, justify
- **text-decoration** – Decorazione – none, underline, overline, line-through, blink
- **text-indent** – Indentazione della prima riga – length, %
- **text-shadow** – Ombra
- **text-transform** – Controlla le lettere – none, capitalize, uppercase, lowercase

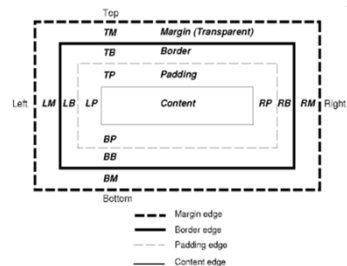
## Font

- **font-style** – Stile, corsivo – normal, italic
- **font-variant** – Maiuscoletto o normale – normal, small-caps
- **font-weight** – Peso, grassetto – normal, bold, bolder, lighter, 100, 200, 300, 400, 500, 600, 700, 800, 900
- **font-size** – Dimensione – xx-small, x-small, small, medium, large, x-large, xx-large, smaller, larger, length, %
- **font-family** – Lista con priorità di nomi di famiglie di font e/o nomi di famiglie generiche – family-name, generic-family (serif, sans-serif, cursive, fantasy, monospace)
- **font** – Imposta tutte le proprietà del font

## Font - dimensioni

- **px**: misura in pixel (fissa rispetto a preferenze utente e risoluzione monitor)
- **pt**: punti tipografici, 1/72 pollice (fissa rispetto a preferenze utente)
- **em**: rispetto alla dimensione di font attuale per l'elemento
- **%**: rispetto alla dimensione di font dell'elemento genitore

## Box model



## Bordi

- **border-width** – Ampiezza dei quattro bordi – 1-4 valori ( )
- **border-style** – Quattro bordi – 1-4 valori – none, hidden, dotted, dashed, solid, double, groove, ridge, inset, outset
- **border-color** – Colore – 1-4 valori – color
- **border** – Imposta tutte le proprietà (uguali) per i quattro bordi
  - `.important { border: 2px dashed red; }`
- **border-top, border-right, ...** – Imposta tutte le proprietà per il bordo superiore, destro, ...
  - `blockquote { border-left: 2px solid black; }`
- I bordi *top, right, bottom, left* possono essere cambiati indipendentemente usando proprietà separate: es. *border-top-width*

## Margini esterni e interni

- **margin** – Spazio attorno agli elementi – 1-4 valori
  - È possibile usare valori negativi per sovrapporre il contenuto
  - I margini *top, right, bottom, left* possono essere cambiati indipendentemente usando proprietà separate, es. *margin-top*
  - Possibile usare scorciatoia *margin* per impostare tutti i margini
- **padding** – Spazio tra il bordo e il contenuto degli elementi – 1-4 valori
  - Non sono ammessi valori negativi
  - I padding *top, right, bottom, left* possono essere cambiati indipendentemente usando proprietà separate, es. *padding-top*
  - Possibile usare la scorciatoia *padding* per impostare tutti i padding

## Dimensioni

- **width** – Ampiezza – auto, length, %
- **height** – Altezza – auto, length, %

## Posizione

- **bottom** – Distanza del limite inferiore di un elemento rispetto all'elemento che lo contiene – *auto, %, length*
- **left** – Sinistra ... – *auto, %, length*
- **right** – Destra ... – *auto, %, length*
- **top** – Superiore – *auto, %, length*
- **vertical-align** – Allineamento verticale – *top, middle, bottom, baseline, sub, super, text-top, text-bottom, length, %*
- **position** – Modalità di posizionamento – *static, relative, absolute, fixed, inherit*
- **float** – Elementi posizionati a lato, circondati da testo – *left, right, none, inherit*

## Visibilità

- **z-index** – Ordine nello stack (rilievo, valori più alti sono in primo piano) – auto, number
- **overflow** – Impostazione per contenuto più ampio dell'area disponibile – *visible, hidden, scroll, auto*
- **visibility** – Visibilità – *visible, hidden* (occupa spazio)
- **display** – Visualizzazione – *none* (non occupa spazio), *block, inline, ...*