

LEZIONE 07 – CSS BASE

I fogli di stile

I fogli di stile rispondono all'esigenza del web designer di controllare maggiormente l'aspetto finale delle pagine web e soprattutto di separare il contenuto dalla forma, per poter più facilmente riadattare un contenuto a media diversi. Nell'intento di chi ha sviluppato questo standard, i CSS dovrebbero permettere di separare la descrizione della formattazione di una pagina dalla descrizione della struttura e del contenuto del. In questo modo, la struttura del documento rimane più pulita ed è più facile modificare l'aspetto della pagina semplicemente modificando lo stile che le è stato applicato, soprattutto se le indicazioni dello stile CSS sono salvate in un file a parte poi richiamato nel file HTML. Inoltre, applicando lo stesso file con le indicazioni di stile a tutte le pagine di un sito, otterrete facilmente un'uniformità di stile in tutto il sito il che non guasta mai. Lo standard è arrivato alla versione 3: CSS3.

La sintassi CSS (Cascading Style Sheets, Fogli di stile in sequenza)

Le indicazioni dei fogli di stile seguono una sintassi diversa rispetto a HTML ed è opportuno anticiparla qui, in modo che gli esempi seguenti siano più chiari. Ogni indicazione di stile si chiama regola e segue questa struttura:

Selettore {proprietà:valore; proprietà:valore}

Il **selettore** è l'elemento da modificare, che può essere un tag HTML o una classe (ne parleremo fra un po').

La **proprietà** è l'aspetto del selettore che si vuole modificare (per esempio il colore) e **valore** è il valore che questa proprietà deve assumere.

È possibile modificare più di una proprietà in un'unica regola, separando le varie proprietà con un punto e virgola (;).

Fra la proprietà e il suo valore occorre un simbolo di due punti (:). Le coppie proprietà-valore vanno poste fra parentesi graffe.

ESEMPI

h1 {font-color:green; font-size:9pt}

Questa riga definisce l'aspetto dei tag `<h1>` stabilendo che essi saranno di colore verde e la dimensione del carattere sarà di nove punti. Quindi, abbiamo usato due coppie proprietà-valore. Da questo primo esempio, notate anche che, quando il nome della proprietà è composto da due parole (font-size, font-color) queste devono essere separate da un trattino (-).

body {font-family: Verdana;}

per indicare che una pagina deve essere scritta usando il font Verdana, significa che tutti gli elementi in `<body>` saranno scritti con questo font a meno che non sia specificato diversamente per il singolo elemento.

Quindi, se alla regola precedente aggiungo questa:

h1 {font-family: Arial;}

significa che tutto il contenuto di `<body>` sarà scritto in Verdana, tranne i titoli `<h1>` che useranno l'Arial. In pratica, possiamo dire che l'indicazione più specifica prevale su quella eventualmente ereditata.

Per esempio, la regola:

```
p {font-family: arial; font-size: 12pt; margin-right:0.80in; margin-left: 0.80in}
```

che imposta per i paragrafi il carattere **Arial 12 punti** e una distanza dal margine destro e sinistro di **0,80 inch**, può essere abbreviata come segue:

```
p {font-family: arial; size: 12pt; margin: 0.80in; 0.80in}
```

In pratica, abbiamo raggruppato le indicazioni sul carattere (**family** e **size**) e quelle relative ai margini (per quanto riguarda i margini, se non si danno indicazioni particolari, i valori sono relativi, nell'ordine, ai margini superiore, destro, inferiore e sinistro).

SINTASSI ABBREVIATA DEI SELETTORI

Oltre che per gli attributi, è possibile adottare una sorta di sintassi abbreviata anche per i selettori. Immaginate che i titoli `<h1>`, `<h2>` e `<h3>` debbano essere tutti rossi e in grassetto. Ognuno avrà poi una dimensione diversa. È possibile creare una regola che raggruppa le formattazioni comuni, poi delle regole per le formattazioni specifiche, in questo modo:

```
h1, h2, h3 {font-weight: bold; color: #FF0000; }
```

```
h1 {font-size: 24pt;}
```

```
h2 {font-size: 20pt;}
```

```
h3 {font-size: 16pt;}
```

Come fornire le indicazioni di stile

Finora abbiamo accennato alla sintassi CSS, ma non abbiamo ancora visto come associare le regole CSS a una pagina HTML.

Fogli di stile esterni

La soluzione che spesso risulta più pratica è quella di creare un file esterno che contenga tutte le indicazioni di stile e che abbia estensione **.CSS**.

Questo file potrà essere richiamato da tutte le pagine in cui occorre senza la necessità di riscrivere le regole di stile ogni volta. I vantaggi sono molti, ci limitiamo a segnalarne alcuni:

- uniformità formale di tutte le pagine in cui questo file è richiamato;
- facilità di “restyling” delle pagine. Basta modificare il file con le indicazioni di stile per rinnovare completamente l'aspetto di tutte le pagine a cui tale file è applicato.

Ma quale dev'essere la struttura del file **.CSS**? È molto semplice. Dovete solo inserire le regole di stile, una dopo l'altra, per esempio così:

```
body {font-size: 10pt;}
```

```
a:link {text-decoration: none; color: #006600;}
```

```
a:hover {text-decoration: underline; color: #006600;}
```

```
a:visited {color: #009900; text-decoration: none;}
```

```
table {font-size: 9pt; }
```

```
h1 {font-size: 12pt; color: #006600;}
```

Una volta salvato, il foglio di stile va importato nella pagina HTML che deve essere formattata. Basterà aggiungere alla sezione **<head>** della pagina la riga che segue:

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

Dove il valore dell'attributo href è il nome del foglio di stile che vogliamo incorporare (in questo caso stile.css). È possibile richiamare in una pagina più fogli di stile esterni. In caso di conflitto, prevalgono le regole del foglio importato per primo.

Fogli di stile interni

Se preferite inserire le indicazioni di stile direttamente all'interno della pagina HTML da modificare, potete ricorrere al tag **<style>**, da inserire nella sezione **<head>** della pagina.

Tutte le regole di stile vanno inserite fra la coppia di tag **<style>** **</style>**. Le indicazioni di stile espresse in questo modo saranno valide per tutta la pagina. Facciamo un esempio:

```
<head> ...
```

```
<style type="text/css">
```

```
body { font-family: Verdana, Arial, Helvetica, sans-serif; font-size: 10pt; }
```

```
h1 { font-size: 12pt; color: #FF0000; }
```

```
</style>
```

```
</head>
```

Stili in linea – attributo style

CSS e HTML ci permettono anche di specificare le indicazioni di stile “in linea”, cioè all'interno dei normali tag HTML, utilizzando l'attributo **style**.

Il valore dell'attributo style sarà proprio la regola di stile, priva, ovviamente, del selettore. Per esempio, per indicare che uno specifico paragrafo deve avere un rientro a sinistra e a destra di **0,50 inch** possiamo scrivere:

```
<p style="margin-left:0.50in; margin-right:0.50"> contenuto del paragrafo </p>
```

Classi e selettori

Negli esempi che vi abbiamo proposto in precedenza abbiamo sempre usato come selettore un tag HTML, ma i CSS ci permettono di creare regole più “raffinate” che si riferiscono a selettori più specifici. Per esempio, è possibile usare l'attributo **id** o l'attributo **class** per identificare uno o più elementi della pagina e applicare lo stile solo a quello.

Facciamo un esempio molto semplice. Immaginiamo di voler creare uno stile che allinea a destra i paragrafi e uno che li allinea al centro. Se usassimo come selettore il tag **<p>** tutti i paragrafi di una pagina avrebbero una determinata formattazione. Noi, invece, abbiamo bisogno di applicare ai vari paragrafi stili diversi. Vediamo come farlo.

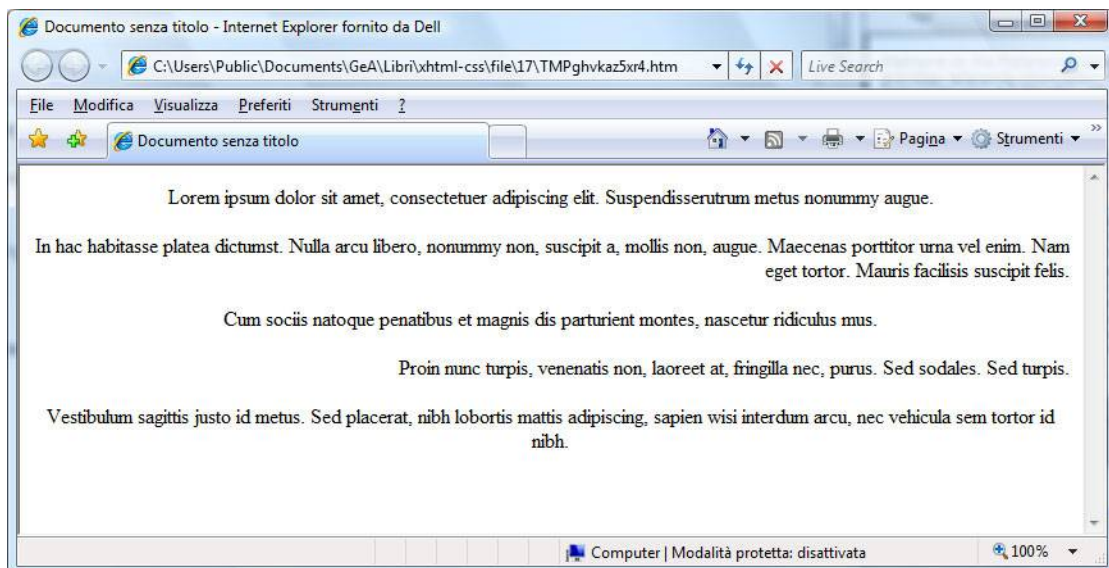
La sintassi è diversa a seconda che scegliate di identificare i paragrafi con l'attributo **id** o con l'attributo **class**.

Cominciamo con un esempio che usa l'attributo **class**:

```

<head>
<style type="text/css">
.destra {text-align: right;}
.centro {text-align: center;} </style>
</head>
<body>
<p class="centro">
Lorem ipsum dolor sit amet, consectetur adipiscing elit. Suspendisse utrum metus nonummy augue.
</p>
<p class="destra">
In hac habitasse platea dictumst. Nulla arcu libero, nonummy non, suscipit a, mollis non, augue. Maecenas porttitor urna vel enim. Nam eget tortor. Mauris facilisis suscipit felis.
</p>
<p class="centro">
Cum sociis natoque penatibus et magnis dis parturient montes, nascetur ridiculus mus.
</p>
<p class="destra">
Proin nunc turpis, venenatis non, laoreet at, fringilla nec, purus. Sed sodales. Sed turpis.
</p>
<p class="centro">
Vestibulum sagittis justo id metus. Sed placerat, nibh lobortis mattis adipiscing, sapien wisi interdum arcu, nec vehicula sem tortor id nibh.
</p>
</body>
</html>

```



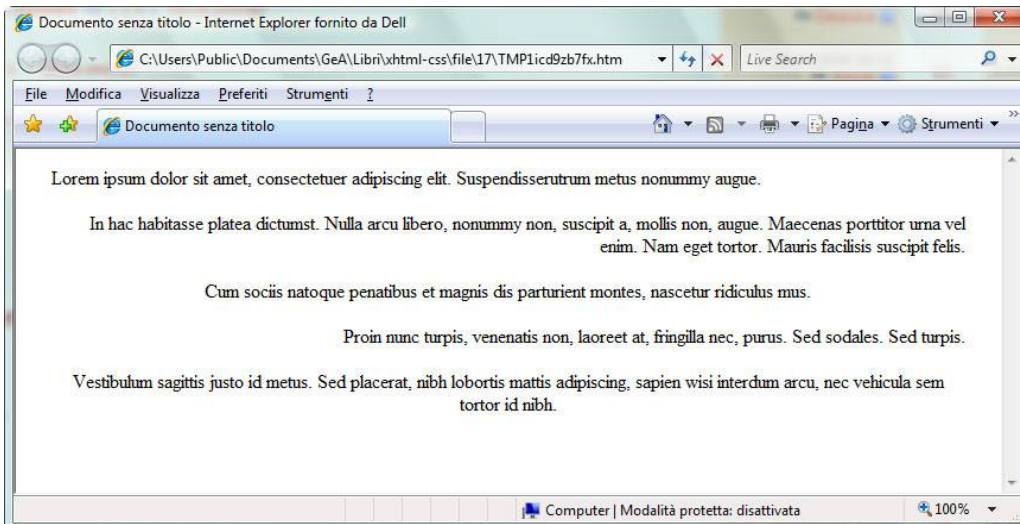
Al posto di **class** si potrebbe usare l'attributo **id** il risultato non cambia.

In pratica, quando volete identificare un elemento (non necessariamente un tag `<div>`) con class dovete definire il nome dello stile usando, prima del nome, un punto (.). Se invece usate id, prima del nome dello stile occorre utilizzare il simbolo cancelletto (#).

Se usate la sintassi di class, avete anche la possibilità di specificare il tag per cui volete usarlo (altrimenti può essere utilizzato per tutti i tipi di tag).

Correggeremo l'esempio che abbiamo fatto in precedenza aggiungendo anche la definizione di un **margin** di 20 px a sinistra e a destra di ogni paragrafo, a qualsiasi classe esso appartenga.

```
<head>
<style type="text/css">
p.destra { text-align: right; }
p.centro { text-align: center; }
p { margin-right: 20px; margin-left: 20px; }
</style>
</head>
<body>
<p>
Lorem ipsum dolor sit amet, consectetur adipiscing elit. Suspendisse utrum metus nonummy augue.
</p>
<p class="destra">
In hac habitasse platea dictumst. Nulla arcu libero, nonummy non, suscipi a, mollis non, augue. Maecenas porttitor urna vel enim. Nam eget tortor. Mauris facilisis suscipit felis.
</p>
<p class="centro">
Cum sociis natoque penatibus et magnis dis parturient montes, nascetur ridiculus mus.
</p>
<p class="destra">
Proin nunc turpis, venenatis non, laoreet at, fringilla nec, purus. Sed sodales. Sed turpis.
</p>
<p class="centro">
Vestibulum sagittis justo id metus. Sed placerat, nibh lobortis mattis adipiscing, sapien wisi interdum arcu, nec vehicula sem tortor id nibh.
</p>
</body>
</html>
```



Selezione dei collegamenti sulla base del loro stato

Fra i selettori CSS che sono stati più usati ci sono, senza dubbio, quelli che permettono di modificare l'aspetto di un collegamento a seconda del suo stato. I selettori che avete a disposizione sono:

- **a:link:** modifica l'aspetto di un link allo stato di quiete (il puntatore del mouse non si trova sul link), che non è ancora stato visitato (ossia, il browser non ha ancora visualizzato la pagina a cui quel link punta);
- **a:visited:** modifica l'aspetto di un collegamento già visitato. È molto importante aiutare il visitatore a orientarsi in un sito distinguendo visivamente i collegamenti già visitati da quelli ancora da esplorare;
- **a:focus:** modifica l'aspetto di un link selezionato da tastiera;
- **a:hover:** modifica l'aspetto di un link su cui passa il puntatore del mouse;
- **a:active:** modifica l'aspetto di un link mentre l'utente sta facendo clic su di esso.

Al loro primo apparire questi selettori (chiamati più propriamente pseudoselettori) sono stati usati per eliminare la sottolineatura standard dei collegamenti, salvo rifarla ricomparire quando il mouse passa su di essi. Se volete ottenere un effetto di questo tipo, ecco le regole di stile che vi occorrono:

```
<style type="text/css">
```

```
a:link { text-decoration: none; }
```

```
a:hover { text-decoration: underline; }
```

```
</style>
```

Quando vi accingete a modificare l'aspetto dei collegamenti fate sempre attenzione a non disorientare l'utente. Nei primi anni di vita del web, l'aspetto dei collegamenti, testo blu sottolineato per i link da visitare e testo viola sottolineato per i link già visitati, era una certezza.

A colpo d'occhio si poteva distinguere un link nella pagina. Oggi, però, dopo l'avvento dei fogli di stile, questa certezza è venuta meno. Riflettete sempre se vale la pena di modificare l'aspetto dei collegamenti, magari a scapito della fruibilità della pagina.

La semplice trasformazione dell'aspetto del mouse su un collegamento potrebbe non essere sufficiente per permettere ai vostri lettori di capire che un certo elemento è un collegamento.

Unità di misura più comuni

I fogli di stile hanno un vantaggio notevole, vi permettono di poter specificare in modo preciso la dimensione degli elementi. Sono accettate diverse unità di misura, come i punti tipografici per la dimensione del testo. Per quanto riguarda le dimensioni potete usare le seguenti unità di misura:

- **in (inches – pollici):** poco utile sui monitor, perché poco gestibile in rapporto alla risoluzione e alle dimensioni degli schermi;

- **cm (centimetri) -mm (millimetri):** presentano la stessa difficoltà dei pollici;
- **pt (points -punti):** comoda per definire la dimensione del carattere, è la stessa unità di misura utilizzata dai programmi di videoscrittura;
- **-px (pixels):** corrisponde a un punto sul monitor;
- **-%:** la dimensione è espressa in percentuale rispetto a un altro valore (la misura della pagina, la misura del font base...).

La scelta dell'unità di misura è soggettiva e dettata dalle proprie esigenze. Considerate che ai fini dell'accessibilità, le unità di misura relative sono preferibili, perché favoriscono un più facile ridimensionamento della pagina. Perché la definizione della unità di misura funzioni correttamente, nella definizione dello stile l'unità di misura va scritta attaccata alla misura a cui si riferisce. Per esempio, la regola:

OSSERVAZIONE

body {font-size: 12pt;}

fa in modo che il font di tutta la pagina (o meglio di tutti gli elementi inseriti nel tag `<body>`) abbiano una dimensione di **12 punti tipografici**.

Se invece scrivessimo questo (notate lo spazio vuoto fra 12 e pt):

body {font-size: 12 pt;}

lo stile non funzionerebbe.

Colori

Parlando di HTML abbiamo detto che è possibile specificare un colore con il suo nome inglese (se si tratta di uno dei colori principali) o il suo codice esadecimale. Queste indicazioni sono valide anche nella sintassi CSS, che prevede però anche un altro sistema di indicazione del colore.

È possibile, infatti, costruire un colore specificando, nell'ordine, la percentuale di rosso, verde e blu che deve contenere. Facciamo un esempio:

```
<style type="text/css">
```

```
strong { color:rgb(70%,0%,39%); }
```

```
</style>
```

Questo codice rende tutti i tag `` di una pagina con una sorta di color vinaccia.