

## **CSS3: gli effetti più cool, ma che non puoi ancora implementare... o forse si?**

I **CSS** (Cascading Style Sheet) hanno rivoluzionato il modo di approcciare alla programmazione web: non solo permettono di poter applicare al nostro sito una grafica accattivante, ma, tramite i suoi selettori, può essere oggetto di modifiche tramite javascript.

Dal 1996 (anno in cui il W3C diede vita al CSS1) ai giorni nostri, il linguaggio CSS ha fatto passi da gigante, fino alla versione **CSS3** attualmente in fase di realizzazione.

Se da un lato la maggior parte dei browser (Mozilla Firefox, Safari, Chrome) supporta buona parte dei neonati moduli del **CSS3**, dall'altro il più diffuso browser al mondo, Internet Explorer, non ne supporta nativamente quasi nessuno costringendoci, di fatto, a dover ricorrere a numerosi hacks al fine di ottenere risultati vicini alla sufficienza.

Oggi vedremo insieme, infatti, alcuni **effetti che potrebbero rivoluzionare il web**, ma che non possiamo ancora implementare nei nostri siti proprio a causa delle mancanze di Internet Explorer.

Prima di cominciare un avvertimento: nel caso tu stessi visualizzando questa pagina con Internet Explorer, chiudilo e leggi questo articolo con qualsiasi altro browser!

### **CSS3: -moz-column, -webkit-column**

In moltissimi siti web è forte la presenza di elementi tipografici: dai grandi titoli che ricordano le più famose testate giornalistiche, all'utilizzo del Times New Roman; dall'indentatura del testo, fino addirittura all'utilizzo di uno stile giornalistico nella generazione dei contenuti.

C'è solo un elemento tipico dei quotidiani che non può essere riprodotto all'interno di una pagina web: il testo non può essere diviso in colonne (se non con grande fatica) così come invece è possibile all'interno delle riviste e dei giornali.

O meglio, per essere precisi: non **era** praticamente possibile creare colonne di testo perchè il CSS3 ha colmato questa lacuna: stiamo parlando dei moduli **-moz-column** e **-webkit-column**.

**A**liquam erat volutpat. Pellentesque suscipit ornare bibendum. Class aptent taciti sociosqu ad litora torquent per conubia nostra.

Vivamus iaculis lorem eget mauris accumsan convallis. In mauris mi, faucibus quis cursus a, vulputate ornare justo. Morbi venenatis congue ultrices.

Integer non dapibus mi. Nullam tristique tellus in lacus

Cum sociis natoque penatibus et magnis dis parturient montes, nascetur ridiculus mus. Proin vel metus quis turpis euismod tincidunt.

Vivamus commodo lorem quis enim facilisis vehicula. Duis ultricies varius risus vitae gravida. Pellentesque sit amet euismod massa.

Donec consectetur dui non neque consectetur molestie.

Lorem ipsum dolor sit amet, consectetur adipiscing elit. Aliquam at nulla sed est rhoncus ultrices.

Suspendisse a felis id sapien mollis viverra. Donec vehicula sagittis venenatis.

Pellentesque habitant morbi tristique senectus et netus et malesuada fames ac turpis egestas. Praesent rhoncus enim at risus faucibus accumsan nec

Facciamo prima una piccola digressione: come avrai potuto notare i due moduli sono pressoché identici, fatta eccezione per il loro prefisso.

### Perché si utilizza il prefisso **-moz**? Perché **-webkit**?

**-moz** è il prefisso che si riferisce al motore grafico sul quale si basa Mozilla Firefox: soprattutto con il css3, infatti, l'interessamento delle funzioni core dei browsers si è fatto sempre più pesante, tanto da dover differenziare i moduli in base al browser utilizzato.

**-webkit** si riferisce, invece, al motore di Safari e Chrome, i due browsers che, per motivi diversi, stanno conoscendo un momento di crescita e di apprezzamento costante: Safari deve il suo successo al florido momento di tutti i prodotti Apple, Chrome viene da Google e come ogni cosa che viene da Google è un prodotto apprezzato dai più.

Risolto il dilemma amletico riguardante questi due prefissi, vediamo come impiegarli al meglio!

Torniamo alle nostre colonne, quindi, e supponiamo di voler ottenere [questo risultato](#):

Credi sia difficile? Beh, resterai stupefatto dalla sua semplicità!

Vediamo di cosa abbiamo bisogno:

- Una pagina html (*index.html*);
- Un foglio di stile (*style.css*).

Creiamo una pagina html (*index.html*):

Adesso è la volta del foglio di stile da associare al nostro documento (*style.css*):

### Cosa abbiamo fatto?

Partiamo dalla pagina html: diamo un titolo alla nostra pagina, un sottotitolo e dopodiché creiamo un lungo paragrafo, contenente in questo caso del testo di prova.

Completata la pagina html, passiamo al css: optiamo per un design ridotto all'osso nel quale spicca sicuramente questa porzione di codice:

Qui entrano in gioco i prefissi che abbiamo appena visto ossia **-moz** e **-webkit**: mediante **-webkit-column-count** e **-moz-column-count** comunichiamo al foglio di stile il numero di colonne in cui suddividere il nostro testo; attraverso **-moz-column-gap** e **-webkit-column-gap** passiamo al browser la distanza da mantenere tra una colonna e l'altra.

Qui potrai trovare l'[esempio](#): ricordati di visualizzare questa pagina con un browser che non sia Internet Explorer!

Purtroppo non c'è ancora modo di poter implementare questo effetto su Internet Explorer, che nonostante le sue innumerevoli pecche resta comunque il browser più diffuso.

Ci sono però determinati effetti figli del nuovo CSS3, che possono essere ottenuti anche in Internet Explorer tramite degli hacks, come detto all'inizio dell'articolo: è il caso di **-webkit-box-shadow** e **-moz-box-shadow**.

## CSS3: -webkit-box-shadow e -moz-border-shadow

### Diamo profondità agli elementi

Le ombre sono cool: danno un senso di profondità agli elementi presenti all'interno della nostra pagina web, trasmettono interattività, spesso sono quel dettaglio in più che arricchisce il nostro sito.

Il css3 permette di poter applicare un'ombreggiatura realistica, precisa e fortemente modificabile ai nostri elementi.

Ricordi com'è stato facile dividere il nostro testo in colonne? Beh dare profondità ai box tramite le ombre sarà ancora più semplice!

Torniamo sulla pagina (*index.html*) che abbiamo creato in precedenza e supponiamo di inserire il

titolo all'interno di un div (*titleContainer*):

al quale associamo uno stile all'interno nel nostro file "*style.css*":

Diamo uno stile nuovo al nostro titolo che avrà quindi queste sembianze nel css:

Come puoi vedere bene [qui](#) la testata del nostro sito avrà un'ombra realistica e perfettamente controllabile, infatti se esaminiamo i moduli del css3 utilizzati:

possiamo riconoscere in ordine di comparsa:

- **Offset orizzontale** dell'ombreggiatura (2px);
- **Offset verticale** dell'ombreggiatura (5px);
- **Raggio di sfocatura** dell'ombreggiatura (2px);
- **Colore** dell'ombreggiatura (#777).

## E per Internet Explorer?

Come sempre, Internet Explorer è un caso da affrontare separatamente: infatti il browser di casa Microsoft supporta un certo tipo di ombreggiatura sugli elementi delle pagine html, ma con una sintassi proprietaria:

Le voci presenti sono abbastanza intuitibili:

- **color**: il colore in esadecimale;
- **direction**: la direzione dell'ombreggiatura;
- **strenght**: l'intensità dell'effetto ombra.

Uno dei vari inconvenienti di questo hack è che il nostro foglio di stile non passerà più la validazione.

Il risultato? Ecco [qui](#), nella pagina frutto dei nostri esperimenti!

Meno elegante e potente dei due esempi precedenti, ma come si dice: a mali estremi estremi rimedi!

## Conclusioni

In questo articolo hai avuto modo di vedere un effetto veramente bello, ma che non puoi ancora implementare (*testo in colonna*) ed un altro molto utile che con qualche hack puoi visualizzare su tutti i browsers (*ombreggiature*).

Nel prossimo articolo vedremo altri effetti da applicare ai nostri elementi, ma stavolta chi utilizza Internet Explorer non sarà del tutto tagliato fuori!

Il CSS3 è alle porte: non lasciarti trovare impreparato!

Prima di chiudere una piccola curiosità, stai già utilizzando qualcuno dei nuovi moduli del CSS3 nei tuoi lavori?