

In questo articolo, che costituisce un'appendice al tutorial su HTML5 e CSS3 postato nei mesi scorsi, vedremo come modificare la pagina del blog precedentemente analizzata in modo da renderla fruibile e visivamente gradevole sia sotto Internet Explorer 8/9, che sotto Google Chrome, Safari e Mozilla Firefox.

Quello che ci interessa, in particolare, è raggiungere una buona compatibilità del nostro codice con le versioni 8 e 9 del browser di casa Microsoft.

Per chi volesse rivedere le parti già pubblicate può consultare la sezione "Articoli precedenti" sottostante.

---

### Articoli precedenti:

#### HTML5 & CSS3 - Parte 1

In questo tutorial creeremo la pagina di un Blog usando tecniche di ultima generazione basate su HTML5 e CSS3.

Il tutorial mira a mostrare il modo in cui saranno costruiti i siti web lato frontend allorché le software house avranno integrato le nuove specifiche HTML e CSS nei loro programmi di navigazione. Difatti gran parte dei più noti browser web, non supporta appieno le nuove specifiche di questi linguaggi, a cominciare da Internet Explorer 8. Tuttavia, la versione 9 di quest'ultimo browser, giunta ormai in fase di beta test, dovrebbe colmare il gap presente rispetto software di navigazione di altri produttori.

[Leggi articolo](#)

### HTML5 & CSS3 - Parte 2

Nella prima parte di questo tutorial abbiamo introdotto alcuni concetti chiave legati alle nuove specifiche di HTML5, abbiamo introdotto alcuni nuovi tag per la costruzione di pagine web ed abbiamo parlato del fatto che questi nuovi tag sono semanticamente significativi per i motori di ricerca e per gli screen reader.

Successivamente, abbiamo introdotto il progetto d'esempio che desideriamo realizzare che consiste nell'implementazione dell'interfaccia di frontend di un ipotetico Blog, abbiamo presentato una bozza della struttura della pagina e una sua prima conversione in codice HTML5.

[Leggi articolo](#)

### HTML5 & CSS3 - Parte 3

Nelle lezioni precedenti di questo tutorial abbiamo parlato di HTML5, abbiamo presentato alcuni dei nuovi tag introdotti con esso ed abbiamo creato la parte di codice HTML che costituisce il progetto d'esempio che abbiamo deciso di realizzare, ovvero la creazione della pagina di un blog.

A partire da questa lezione, invece, parliamo di CSS3 ed introduciamo il codice necessario ad impaginare e formattare il nostro markup.

[Leggi articolo](#)

### HTML5 & CSS3 - Parte 4

## HTML5 & CSS3 - Appendice Speciale (Parte 1)

Scritto da Enrico Battuello

Sabato 24 Novembre 2012 19:15 - Ultimo aggiornamento Venerdì 07 Dicembre 2012 01:33

---

Nelle prime due lezioni di questo tutorial abbiamo parlato di HTML5, abbiamo presentato alcuni dei nuovi tag introdotti con esso ed abbiamo creato la parte di codice HTML che costituisce il progetto d'esempio che desideriamo realizzare, ovvero la pagina di un blog.

Nella terza lezione, poi, abbiamo iniziato a "stendere" il foglio di stile CSS per impaginare i diversi elementi della pagina. In particolare, abbiamo visto la sezione di reset ed inizializzazione del CSS e la sezione di formattazione della barra di navigazione.

[Leggi articolo](#)

### HTML5 & CSS3 - Parte 5

Nelle prime due lezioni di questo tutorial abbiamo parlato di HTML5, presentando alcuni dei nuovi tag introdotti con esso ed abbiamo creato la parte di codice HTML che costituisce il progetto d'esempio che desideriamo realizzare, ovvero la pagina di un blog.

Nella terza e quarta lezione, poi, abbiamo iniziato a trascrivere il codice del foglio di stile necessario ad impaginare le diverse sezioni del nostro documento HTML. In particolare, siamo partiti dalla definizione di una rapida sezione di reset nel CSS, per poi passare alla stilizzazione della barra di navigazione e della sezione "intro" della nostra pagina, facendo particolare riferimento ad alcuni tag nuovi, introdotti con le specifiche CSS3, necessari per posizionare, formattare e impaginare le immagini di sfondo di una specifica sezione.

[Leggi articolo](#)

---

Ovviamente, per raggiungere il nostro scopo utilizzeremo dei javascript e dei fogli di stile che suppliranno alle mancanze dei browser Microsoft. A tal proposito faccio notare che IE8 non riconosce i tag HTML5, è del tutto conforme alle specifiche CSS2.1 ma a queste aggiunge un

supporto molto modesto e limitato per le nuove specifiche CSS3.

Con Internet Explorer 9 le cose migliorano nettamente. I tag HTML5 sono riconosciuti ed il supporto ai diversi moduli che costituiscono le specifiche CSS3 del World Wide Web Consortium (W3C) risulta notevolmente migliorato. Tuttavia, alcune caratteristiche come l'impaginazione multicolonna del testo risultano ancora non implementate.

Per chi volesse approfondire l'argomento, Microsoft mette a disposizione a [questo link](#) un'interessante tabella di compatibilità fra le diverse versioni del proprio browser (a partire da IE5) e le specifiche CSS2.1 e CSS3.

Ma iniziamo a vedere le parti salienti del nostro codice. Innanzitutto per sopperire alla mancanza di compatibilità di Internet Explorer 8 con le specifiche HTML5, utilizziamo un javascript che permette di interpretare correttamente i nuovi tag consentendo l'utilizzo degli stili su di essi. Fondamentalmente questo script definisce una funzione che, dopo aver individuato i nuovi tag della specifica HTML5 nel nostro codice html, li innesta opportunamente nell'albero del documento HTML che si sta elaborando applicando a ciascuno di essi gli stili definiti nell'attributo style o in un foglio di stile apposito.

La definizione di questa particolare funzione è inserita in un file javascript richiamato in un "conditional comment" (commento condizionale) che viene interpretato come codice solo dai browser Microsoft successivi alla versione 5.

Lo stesso codice javascript utilizza una particolare feature, supportata fin dalla versione 4.0 del browser di casa Microsoft, che si chiama "conditional compilation" (compilazione condizionale) con cui è possibile far eseguire il blocco di codice javascript solo in IE 4.0 o successivi. In questo modo si evita che il codice javascript in questione possa influire sui browser di altri produttori.

Questo javascript è conosciuto con il nome di HTML5Shiv ed è rilasciato sotto licenza MIT, una forma di licenza per il software free del Massachusetts Institute of Technology. Chi lo desiderasse può trovare ulteriori informazioni a [questo indirizzo](#) .

## HTML5 & CSS3 - Appendice Speciale (Parte 1)

Scritto da Enrico Battuello

Sabato 24 Novembre 2012 19:15 - Ultimo aggiornamento Venerdì 07 Dicembre 2012 01:33

---

La versione che presento qui di HTML5Shiv, è quella usata nell'esempio. Essa è piuttosto vecchia ed anche piuttosto criptica dal punto di vista del codice utilizzato. Per chi fosse interessato, è sicuramente disponibile una versione più aggiornata e comprensibile dello script a [questo indirizzo](#) .

Ma andiamo al codice di HTML5Shiv:

```
{codecitation class="brush: js; gutter: true;highlight: [3,5,6,24,25,26];" width="700px"}

// html5shiv MIT @rem remysharp.com/html5-enabling-script // iepp v1.6.2 MIT @jon_neal
iecss.com/print-protector /*@cc_on( function(m,c){ var
z="abbr|article|aside|audio|canvas|details|figcaption|figure|footer|header|" +
"hgroup|mark|meter|nav|output|progress|section|summary|time|video"; function n(d){
for(var a=-1;++a<o;) d.createElement(i[a]) } function p(d,a){ for(var
e=-1,b=d.length,j,q=[];++e<b;){ j=d[e]; if((a=j.media||a)!="screen")
q.push(p(j.imports,a),j.cssText) } return q.join("") } var
g=c.createElement("div"); g.innerHTML="<z>i</z>"; if(g.childNodes.length!==1){
var i=z.split("|"), o=i.length, s=RegExp("(^|\\s)("+z+")","gi"),
t=RegExp("</?("+z+")","gi"), u=RegExp("(^[^\\n]*?\\s)("+z+")([\\n\\w\\W]*?)","gi"),
r=c.createDocumentFragment(), k=c.documentElement; g=k.firstChild;
var h=c.createElement("body"), l=c.createElement("style"),f; n(c); n(r);
g.insertBefore(l,g.firstChild); l.media="print"; m.attachEvent("onbeforeprint",
function(){ var d=-1, a=p(c.styleSheets,"all"), e=[], b;
for(f=f||c.body;(b=u.exec(a))!=null;)
e.push((b[1]+b[2]+b[3]).replace(s,"$1.iepp_$2")+b[4]);
for(l.styleSheet.cssText=e.join("n");++d<o;){ a=c.getElementsByTagName(i[d]);
e=a.length; for(b=-1;++b<e;)
if(a[b].className.indexOf("iepp_")<0) a[b].className+=" iepp_"+i[d] }
r.appendChild(f); k.appendChild(h); h.className=f.className;
h.innerHTML=f.innerHTML.replace(t,"<$1font") } );
m.attachEvent("onafterprint", function(){ h.innerHTML="";
k.removeChild(h); k.appendChild(f); l.styleSheet.cssText="" }
) } } )(this,document);@*/

}/codecitation}
```

Come si vede il codice è molto compatto. Mi preme precisare che questa versione di HTML5Shiv nella sua trascrizione originale non era neanche indentata e tutto il codice era scritto su un'unica riga.

Poiché l'analisi di questo script non è tra gli obiettivi dell'articolo, nel codice mi sono limitato ad evidenziare:

- alla riga 3 l'uso dell'istruzione `@cc_on` per l'attivazione della conditional compilation;
- alle righe 5 e 6 la variabile stringa `z` in cui sono inseriti i tag che saranno elaborati da questo script (separati mediante il carattere "|");
- alle righe 24, 25 e 26 l'uso della funzione `RegExp` che viene adoperata in javascript per elaborare le regular expression. Queste ultime sono oggetti che descrivono pattern di caratteri e sono usate per eseguire compiti di pattern-matching e operazioni di ricerca e sostituzione del testo.

Oltre ad `HTML5Shiv`, per rendere la pagina del nostro blog compatibile con IE8 utilizzeremo un altro javascript: [CSS3 multi-column](#).

Come è facilmente intuibile dal nome, questo javascript si occupa di fornire il supporto per i tag `CSS3` multi-colonna. In particolare esso permette di interpretare le properties:

- `column-count`;
- `column-width`;
- `column-gap`;
- `column-rule`;

mentre ignora altre properties, come:

- `column-rule-width` (in sostituzione è possibile usare `column-rule`);
- `column-rule-style` (in sostituzione è possibile usare `column-rule`);
- `column-rule-color` (in sostituzione è possibile usare `column-rule`);
- `column-span`;
- `column-width-policy`;
- `column-space-distribution`;

Come il precedente script, anche `CSS3 multi-column` è molto vecchio ma a differenza di `HTML5Shiv` potrebbe non esserci una versione più aggiornata. Inoltre, è bene precisare che la

versione originale di questo script non funziona con il nostro blog probabilmente a causa della sua età. Difatti, come vedremo più avanti, manca il supporto al tag HTML5 "ARTICLE". Più avanti vedremo il modo in cui è stato aggirato il problema.

Poiché CSS3 multi-column è uno script molto lungo, in questo articolo non posterò integralmente il suo codice ma evidenzierò solo la modifica (banale) che ho apportato affinché lavori correttamente con il nostro esempio. In ogni caso, renderò disponibile la versione modificata dello script nell'archivio zip contenente l'esempio finale funzionante.

Chi volesse ulteriori informazioni su questo javascript e sul suo autore può iniziare a dare un'occhiata a [questo indirizzo](#) . La versione in mio possesso è la 1.02 beta dell'08 gennaio 2008. Credo sia anche l'ultima release ufficiale rilasciata dall'autore, Cdric Savarese. La licenza che lo accompagna è di tipo [CC-GNU LGPL](#) .

Ma vediamo subito il blocco di codice modificato che ci interessa:

```
{codecitation class="brush:js;gutter:false;highlight:[5,14];" width="700px"}

// Versione originale function CSS3MultiColumn() { //alert('Development Version'); var
cssCache = new Object(); var splittableTags = new Array('P','DIV', 'SPAN',
'BLOCKQUOTE','ADDRESS','PRE', 'A', 'EM', 'I', 'STRONG', 'B', 'CITE', 'OL', 'UL', 'LI' ); var
pseudoCSSRules = new Object(); var ut = new CSS3Utility(); /* continuazione del blocco di
codice */ // Versione modificata function CSS3MultiColumn() { //alert('Development
Version'); var cssCache = new Object(); var splittableTags = new Array('ARTICLE','P','DIV',
'SPAN', 'BLOCKQUOTE','ADDRESS','PRE', 'A', 'EM', 'I', 'STRONG', 'B', 'CITE', 'OL', 'UL', 'LI' );
var pseudoCSSRules = new Object(); var ut = new CSS3Utility(); /* continuazione del blocco
di codice */

{/codecitation}
```

Come evidenziato, la modifica si trova all'inizio della funzione javascript CSS3MultiColumn e in pratica consiste nell'aggiunta del tag "ARTICLE" all'array "splittableTags" dei tags splittabili, ovvero nell'elenco degli elementi che, quando contengono testo, possono essere suddivisi su più colonne. Pertanto, per quanto appena detto, se un domani volesse splittare un tag "SECTION", inseritelo in questo array.

Una piccola nota riguardante gli elementi splittabili. Non è possibile suddividere su più colonne i tag "IMG", "VIDEO", "OBJECT" o equivalenti. Quindi non è un caso che essi non siano presenti nella dichiarazione ed inizializzazione dell'array splittableTags del blocco di codice precedente.

Ci sono poi una serie di limitazioni/avvertenze riguardanti questo script:

- I valori per gli attributi column-width e column-gap devono essere espressi in pixels;
- Il parser elabora anche le regole CSS commentate;
- Il parser elabora solo stylesheets linkati ovvero fogli di stile caricati con il tag link mentre ignora gli style elements, le direttive di importazione e gli stili inline.
- Poiché il markup della pagina web viene comunque modificato dallo script, alcune regole CSS potrebbero non funzionare dopo l'applicazione del layout multi-colonna.
- Poiché l'implementazione di questa utility si basa su un parser javascript degli stili molto "leggero", conviene evitare l'uso di strutture CSS complesse, come l'uso di regole in cascata.

Nonostante la presenza di questi "limiti" vedremo comunque che CSS3 Multi Column si adatta perfettamente alle nostre esigenze.

Concludiamo l'elenco degli script utilizzati per rendere il nostro layout compatibile con Internet Explorer 8 introducendo il terzo ed ultimo script. Si tratta di [border-radius.htc](#).

Border Radius è un HTML Component (HTC) script usato per creare angoli arrotondati in Internet Explorer 6, 7 e 8. In generale un file HTML Component è un meccanismo non standard per implementare negli script un componente che modifica il comportamento standard di un oggetto dell'HTML Dinamico (DHTML). Pertanto, un file HTC è un file HTML che contiene degli script (JScript/VBScript) e un insieme di elementi specifici HTC che definiscono un componente che modifica/ridefinisce il comportamento (behavior) di un elemento/oggetto dell'HTML.

Di solito un file di un componente HTML ha estensione .htc e tipo MIME "text/x-component".

Chi volesse ulteriori dettagli può dare un'occhiata ai seguenti link:

- [HTC Reference](#) ;
- [Using HTML Components to Implement DHTML Behaviors in Script](#) ;
- [HTML Components](#) ;

Io non posterò il codice di questo file .htc nell'articolo tuttavia lo renderò disponibile nell'archivio zip contenente l'esempio completo funzionante. E' comunque possibile scaricarlo una copia dal link presentato poco sopra.

Vediamo, quindi, la versione definitiva ed univoca del codice che compone la pagina del nostro blog. Ovviamente essa presenta delle differenze rispetto a quella che avevamo visto nella parte 5 di questo tutorial ma la struttura risulta essere praticamente immutata.

Tengo a precisare fin da subito che, mentre per la parte HTML ho adottato un unico file per tutti i browser invece, per la parte CSS, ho adoperato un foglio di stile per Internet Explorer ed un foglio di stile per Google Chrome, Safari e Mozilla Firefox, linkando dinamicamente da HTML il file corretto, in funzione del browser adoperato, mediante l'utilizzo dei conditional comments.

Iniziamo, quindi, dal codice HTML:

```
{codecitation class="brush: html; gutter: false;highlight: [5,6,7,10,11,25];" width="700px"}
```

```
<!doctype html> <html> <head> <title>The Car Blog</title> <!--[if lt IE 9]><link
rel="stylesheet" type="text/css" href="css/mainstyle_ie.css" media="screen" /><![endif]--> <!--[if
gte IE 9]><link rel="stylesheet" type="text/css" href="css/mainstyle_no_ie8.css" media="screen"
/><![endif]--> <!--[if !IE]><!--><link rel="stylesheet" type="text/css"
href="css/mainstyle_no_ie8.css" media="screen" /><!--<![endif]--> </head> <body> <!--[if lt IE
9]><script src="js/html5.js"></script><![endif]--> <!--[if lt IE 10]><script
src="js/css3-multi-column.js"></script><![endif]--> <header> <h1>The Car Blog</h1>
</header> <nav> <ul> <li class="selected"><a href="#">Blog</a></li> <li><a
href="#">Informazioni</a></li> <li><a href="#">Archivi</a></li> <li><a
href="#">Contatti</a></li> <li class="subscribe"><a href="#">Sottoscrivi via RSS</a></li>
</ul> </nav> <section id="intro"> <section id="imgIntro"></section> <header> <h2>Ti
piacciono le auto? Sei nel posto giusto!</h2> </header> <p>Sei un appassionato di auto? In
```

questo sito puoi trovare opinioni e recensioni sulle case automobilistiche, test drive e schede sulle prove su strada e tutte le anteprime del 2010. Inoltre, se cerchi il massimo dalla tua auto, puoi consultare la nuovissima sezione "Tuning".

Start Main content area

Quattro anelli e una spina. L'Audi R8 elettrica

Postato il November 15th 2010 da Enrico Battuello - 3 commenti

Quando il sogno dell'auto elettrica diventa realtà il suo nome è Audi R8 e-tron. Il lavoro dei tecnici di Ingolstadt,...



Perché non è sufficiente accoppiare un motore elettrico a una batteria per fare un'automobile elettrica. Tutto è gestito...



Non ci sono le marce e il rapporto unico scaglia "la belva" a 100 km/h in neppure 4.8 secondi, il tempo di un respiro, lasciandoci incollati a i sedili per la gran coppia...

Comments

Federico Nava on Novembre 16th 2010 at 05:55

io la Audi e-Tron la trovo carinissima e al passo coi tempi, il frontale é la parte migliore secondo me. E' sportivo e grintoso e farà strage di cuori!

Roberto Loja on Novembre 16th 2010 at 14:06

tra tutte le sportiva che ci sono in commercio devo dire che la Audi e-Tron non passa inosservata! A me questa auto fa impazzire, altro che miscuglio di altri marchi!

Giacomo Bianco on Novembre 18th 2010 at 18:12

E' davvero una bella auto. Fino ad ora ne ho vista una sola per strada, nera, e sono rimasto molto colpito. Cmq complimenti vivi!!!!

Post a comment

Nome

E-mail

Sito web

Commento

End Main content area

Sidebar

Categorie

- [Berlina \(tre volumi\)](#)
- [Cabriolet](#)
- [Coupé](#)
- [Due volumi](#)
- [Familiare \(Station Wagon\)](#)
- [Fuoristrada \(4x4\)](#)
- [Monovolume](#)
- [Multiuso \(MPV\)](#)
- [SUV](#)

Archivi

- [Maggio 2010](#)
- [Giugno 2010](#)
- [Luglio 2010](#)
- [Agosto 2010](#)
- [Settembre 2010](#)
- [Ottobre 2010](#)
- [Novembre 2010](#)

Footer

Informazioni

Il Car Blog si prefigge di divulgare

```
e condividere contenuti riguardanti il mondo delle auto, delle moto e dei motori in genere,...</p>
</section> <section id="blogroll"> <header> <h3>Blogroll</h3> </header>
<ul> <li><a href="http://www.autoblog.it/">Auto Blog.it</a></li> <li><a href="#">Altro
Blog 2</a></li> <li><a href="#">Altro Blog 3</a></li> <li><a href="#">Altro Blog
4</a></li> <li><a href="#">Altro Blog 5</a></li> </ul> </section> <section
id="popular"> <header> <h3>Link utili</h3> </header> <ul> <li><a
href="http://www.autoscout24.it">Auto Scout 24</a></li> <li><a
href="http://www.quattroruote.it">Quattroruote</a></li> <li><a
href="http://car4you.quattroruote.it">Car4You Quattroruote</a></li> <li><a
href="http://www.omniauto.it">Omniauto.it</a></li> <li><a
href="http://www1.agenziaentrate.it/servizi/bollo/calcolo/kw_cv_ins.htm">Calcolo bollo auto
(agenzia entrate)</a></li> <li><a href="http://www.gentemotori.it/">Gente Motori</a></li>
<li><a href="http://www.ozracing.com/">OZ Racing</a></li> <li><a href="#">Altro link
2</a></li> </ul> </section> </section> </footer> </body> </html>
```

{/codecitation}

Come si vede il codice è molto simile a quello visto nelle parti precedenti di questo tutorial. E' per questo motivo che mi sono limitato ad evidenziare esclusivamente l'uso dei commenti condizionali per il caricamento dei javascript e dei fogli di stile e la modifica apportata alla sezione "Intro".

Quest'ultima si è resa necessaria per migliorare il rendering finale della pagina in Internet Explorer 8 a seguito dei suoi limiti "tecnologici". In particolare, nella sezione "Intro" è stato rimosso l'uso dei background multipli ed è stata inserita al suo interno una sotto-sezione "imgIntro" usata per visualizzare la seconda immagine.

Questo tipo di rendering risulta efficiente anche sotto gli altri browser.

Osservando i commenti condizionali possiamo vedere che se la versione di Internet Explorer è inferiore alla 9 allora viene caricato il foglio di stile "mainstyle\_ie.css". In caso contrario (versione di IE superiore alla 8 oppure browser differente da IE) viene caricato il foglio di stile "mainstyle\_no\_ie8.css".

Si osservi che, per caricare il foglio di stile corretto, è stato necessario testare separatamente i casi:

- Browser IE9 o superiore ([if gte IE 9]);
- Browser non appartenente alla famiglia IE ([if !IE]);

anche se entrambe le condizioni, in caso di esito positivo, determinano il caricamento dello stesso file "mainstyle\_no\_ie8.css".

Il doppio test è necessario in quanto, mentre un browser IE9 o superiore interpreta correttamente l'istruzione di caricamento che segue il test del commento condizionale "[if gte IE 9]", invece un browser "NON IE" interpreta questa riga come un semplice commento e, viceversa, mentre un browser "NON IE" interpreta correttamente l'istruzione di link del CSS che segue il test del commento condizionale "[if !IE]", invece un browser Microsoft considera questa istruzione un semplice commento.

Per quanto riguarda il caricamento dei javascript, invece, carichiamo HTML5Shiv (html5.js) solo se il browser è IE 9 o inferiore, mentre carichiamo il supporto per l'impaginazione multi-colonna (css3-multi-column.js) per tutte le versioni di Internet Explorer inferiori alla 10.

Si conclude qui la prima parte di questa appendice al tutorial su HTML5 e CSS3.

Nella seconda parte vedremo il codice dei due fogli di stile applicati all'HTML appena visto ed evidenzieremo le principali differenze tra essi.

---

### Articoli successivi:

[HTML5 & CSS3 - Appendice Speciale \(Parte 2\)](#)

## HTML5 & CSS3 - Appendice Speciale (Parte 1)

Scritto da Enrico Battuello

Sabato 24 Novembre 2012 19:15 - Ultimo aggiornamento Venerdì 07 Dicembre 2012 01:33

---

Nella prima parte di questa appendice al tutorial su HTML5 e CSS3 abbiamo introdotto gli script che utilizziamo per rendere la nostra pagina di blog compatibile con tutti i principali browser web in circolazione (Internet Explore 8 o superiore, Google Chrome, Mozilla Firefox, Safari).

Inoltre, abbiamo visto il codice HTML univoco della nostra pagina web.

Ora ci apprestiamo ad esaminare il codice sorgente dei due fogli di stile utilizzati.

[Leggi articolo](#)

---