



Manuale HTML

Introduzione

Cos'è HTML?

HTML è l'acronimo di HyperText Markup Language. I documenti HTML sono in formato ASCII e possono essere creati con qualsiasi Editor, anche se esistono programmi specifici per semplificare il lavoro. Sia utilizzando un Editor che un programma apposito è sempre possibile vedere quali sono i risultati utilizzando un qualsiasi Browser (Netscape, Mosaic, ecc.). Il linguaggio HTML usa i "markup tags", cioè delle parole chiave, per indicare al Browser come deve essere mostrato il testo. Non tutti i Browser supportano i tags della versione 3 di HTML, quelli sconosciuti verranno perciò ignorati in fase di visualizzazione.

Una delle caratteristiche principali dell'HTML (e del Web stesso) è l'ipertestualità, cioè la possibilità di muoversi liberamente all'interno dei documenti seguendo dei Links, parole (o lettere o immagini) simbolicamente legate ad altri documenti (sia che questi siano presenti sulla stessa macchina che su server remoti), o a parti interne del documento stesso.

È importante saper usare i tags di base per ottenere gli effetti desiderati in fase di visualizzazione del documento, per saper legare le parole ad altri documenti, conoscendo tag aggiuntivi è sicuramente possibile migliorare la qualità e quindi l'impatto del documento stesso; scrivendo HTML è necessario ricordare che i nonostante i documenti siano scritti in ASCII, non tutte le tabelle ASCII sparse per il mondo riconoscono simboli e lettere particolari alla stessa maniera, bisogna perciò utilizzare dei sostitutivi per quei simboli che potrebbero risultare incomprensibili (lettere accentate, e altri simboli), questi sostitutivi sono detti entities e permettono di simulare caratteri non standard attraverso i caratteri standard.

La versione 3.2 di HTML supporta tra le novità presenti l'utilizzo di texture per gli sfondi, la possibilità di utilizzare i colori e le font, la formattazione del testo e delle immagini, la possibilità di incorporare oggetti (come musica di fondo) all'interno del documento, ha inoltre un ampio supporto per quanto riguarda JAVA e Javascript.

Tags di Base

Il Linguaggio

La creazione di una pagina standard

La prima cosa da fare quando si crea un nuovo documento HTML è definirlo secondo questo schema:

```
<HTML>
<HEAD>
</HEAD>
<BODY>
</BODY>
</HTML>
```

Tutto il documento dovrà essere contenuto fra <HTML> e </HTML>; nella sezione HEAD ci andrà inserito solamente il titolo e nella sezione BODY il documento vero e proprio.

Nel tag <BODY> inseriremo le informazioni riguardanti l'aspetto del documento secondo questo schema:

```
<BODY BGCOLOR="#xxxxxx" TEXT="#xxxxxx" LINK="#xxxxxx" VLINK="#xxxxxx"
ALINK="#xxxxxx">
```



Dove BGCOLOR e' il colore dello sfondo, TEXT il colore del testo del documento, LINK il colore dei caratteri per le parole linkate, VLINK il colore del testo per i link già visitati, ALINK il colore del testo al momento del click sul link; i colori sono espressi in RGB in formato esadecimale (ad es. #000000 corrisponde a R=0 G=0 B=0, quindi nero; #FFFFFF a R=255 G=255 B=255 quindi bianco); se come sfondo vogliamo utilizzare una immagine o una texture al posto di BGCOLOR (oppure associato ad esso) si utilizza BACKGROUND="nome_file".

Guardiamo quest'esempio:

```
<HTML>
<HEAD>
<TITLE> HTML di Esempio </TITLE>
</HEAD>
<BODY BGCOLOR="#FFFFFF" LINK="#FF0000" VLINK="#9F0000" ALINK="#0000FF">
<H2>Titolo della Pagina </H2><P>
Questo è solo un esempio<BR>
Che mostra dei semplici tags
</BODY>
</HTML>
```

Verrà mostrato così:

Titolo della Pagina

Questo è solo un esempio
Che mostra dei semplici tags

I tags vengo racchiusi da <...> e da un corrispondente </...> alla fine del testo da formattare; nell'esempio il titolo (che verrà mostrato sopra al menu del browser) è contrassegnato da <TITLE> e dal corrispondente </TITLE> alla fine. Nel minuscolo esempio illustrato vengono utilizzati anche i tags di intestazione -Headers- (in questo caso <H2>) che definiscono la dimensione del carattere dell'header (l'intestazione); il tag di Line Break (
) che forza il testo su una nuova linea ed il tag Paragraph (<P>) che oltre a mandare a capo lascia una linea vuota. I contrassegni <P> e
 sono utilizzati perché il linguaggio HTML non è sensibile alla formattazione con la quale è scritto (tranne che per un caso); inoltre questi due tags non hanno il corrispondente </...> visto che si possono inserire solo a fine linea.

Questi sono i tag fondamentali per creare una pagina standard semplice; vedremo più avanti come creare effetti particolari usando dei tag che vanno ad influenzare il testo stesso.

Abbiamo visto il tag che si utilizza per creare un Header (nel nostro caso di dimensione 2 <H2>), l'header è il tipo di testo che viene utilizzato solitamente come titolo per la pagina all'interno del documento; e viene definito attraverso questo tag <Hn> e </Hn> dove 'n' è un numero compreso fra 6 e 1 (dal più piccolo al più grande), dopo l'header non è necessario inserire il tag di paragrafo o di Line Break. Digitando il documento invece bisogna ricordarsi di utilizzarli ogni qual volta si voglia andare a capo o saltare una riga (come già detto il browser non è sensibile alla formattazione utilizzata in fase di creazione).

La prima cosa che ci verrà voglia di utilizzare nel nostro documento sono i Links; la sintassi per legare un parola (od una frase) ad un indirizzo (HTTP, FTP, GOPHER, MAIL, ecc.) è questa:

```
<A HREF="http://in.un.luogo.lontano">Portami là...</A>
```

...che darà questo risultato:

[Portami là...](http://in.un.luogo.lontano) (non utilizzare questo link!!!);



Il riferimento (HREF) può essere verso qualsiasi tipo di indirizzo raggiungibile col browser o da un visualizzatore esterno (una sessione telnet ad esempio), nel caso che il browser non supporti il tipo di file linkato o non esistano visualizzatori per quel tipo di file il browser chiede come gestirlo, offrendo la possibilità di salvarlo sul disco.

Oltre che verso documenti esterni si possono definire dei documenti interni (utili in caso di pagine di ampie dimensioni contenenti più argomenti); per definire dei link interni bisogna compiere due operazioni, la prima consiste di definire il target (cioè il punto di arrivo del link):

```
<A NAME = "demo_target">questo è il target</A>
```

E quindi definire il link con il solito HREF secondo questa modalità:

```
<A HREF = "#demo_target">link verso un target interno</A>
```

Bisogna ricordarsi di utilizzare il simbolo # nell'HREF verso un target, inoltre i target possono essere sfruttati anche se usiamo un HREF da un documento esterno

Per separare logicamente le pagine si utilizzano i Rulers, linee orizzontali che possono essere inserite con

```
<HR ALIGN=x SIZE=x NOSHADE WIDTH="x%">
```

dove align e' l'allineamento del ruler rispetto alla pagina SIZE la sua altezza espressa in pixel width la sua lunghezza in percentuale della pagina e noshade un flag che permette di ottenere linee piatte (il default è in 3D).

La formattazione del testo

I paragrafi oltre ad essere formattati con <P> e
 possono essere centrati ed allineati con i seguenti tag

```
<CENTER>...</CENTER>  
<P ALIGN=right>...</P>
```

E possono essere rientrati di una tabulazione con

```
<BLOCKQUOTE>...</BLOCKQUOTE>
```

E se proprio è necessario usare un tipo di testo preformattato si può utilizzare questo tag che mostrerà il testo come è stato digitato

```
<PRE>...</PRE>
```

Liste e Menu

Per organizzare al meglio le informazioni e renderle più comprensibili all'utente vengono utilizzate le liste (o i menu); HTML 3 ne mette a disposizione cinque: *liste con 'bullet'*, *liste numerate*, *menu*, *Directory list* e *liste di definizione*. Per capirne le differenze e l'utilizzo si possono vedere i seguenti esempi relativi ad ogni tipo di lista:

Lista con bullet:

Primo elemento



Secondo elemento

La sintassi per crearla è questa:

```
<UL TYPE=disc>  
<LI>Primo elemento  
<LI>Secondo elemento  
</UL>
```

Dove TYPE definisce il tipo di bullet da utilizzare (Disc, Circle - default - , Square); ogni elemento della lista dev'essere preceduto dal tag (List Item).

Lista numerata:

1. Primo elemento
2. Secondo elemento

Con questa sintassi:

```
<OL TYPE=1 START=1>  
<LI>Primo elemento  
<LI>Secondo elemento  
</OL>
```

Type è un numero compreso fra 1 e 5 (1 = 1,2,3....; 2 = A,B,C...; 3 = a,b,c...; 4 = I, II, III...; 5 = i, ii, iii...); e START è il numero di partenza.

Menu:

- Primo elemento
- Secondo elemento

La sintassi è questa:

```
<MENU>  
...  
</MENU>
```

Con i List item come al solito.

Directory List:

- Primo elemento
- Secondo elemento

Questa volta i List item sono racchiusi nei contrassegni <DIR> e </DIR>.

La Definition List si presenta così:

Questa è la prima riga della Definition List

Queste sono le successive linee
della Definition List

Provate voi adesso che ne sapete di più!!!!

La sintassi per le definition list è questa:



```
<DL>
<DT>Questa è la prima riga della Definition List
<DD>Queste sono le successive linee
...
</DL>
```

Aggiungete gli items che funzionano da titoletti con <DT> e quelli che devono essere rientrati contassegnandoli con <DD> .

Le liste possono essere combinate fra loro e organizzate su più livelli (ad esempio il primo livello una lista numerata e un secondo livello a menu), anche se solitamente non si trovano liste con più di tre livelli (che risultano poco leggibili).

Font e Colori

Font Colori e caratteri

Il linguaggio HTML 3.2 permette di variare la dimensione dei caratteri anche nel documento oltre che negli header, permette il grassetto, il corsivo ed il sottolineato da la possibilità di utilizzare alcune font predefinite. Gli effetti ottenibili e le sintassi da utilizzare sono contenute in questa tabella:

Effetto	Sintassi
Dimensione dei caratteri	...
Dimensione dei caratteri	...
<i>Corsivo</i>	<I>... </I>
Grassetto	...
<u>Sottolineato</u>	<U>... </U>
Typewriter	<TT>... <TT>
Lampeggiante	<BLINK>... </BLINK>
<i>Enfasi</i>	...
Strong	...
Sample	<SAMP>... <SAMP>
Code	<CODE>... /CODE>
Keyboard	<KBD>... </KBD>
<i>Variabile</i>	<VAR>... </VAR>
<i>Definition</i>	<DFN>... </DFN>
<i>Citazione</i>	<CITE>... </CITE>
<i>Address</i>	<ADDRESS>... </ADDRESS>

I tags sopraelencati possono essere combinati fra loro per ottenere vari effetti.

HTML 3.2 supporta i font true type (sempre che questi siano supportati dal sistema, altrimenti vengono ignorati) per utilizzarli si utilizza

```
<FONT FACE="nome_font">...</FONT>
```

Sempre utilizzando il tag font oltre alla dimensione (con SIZE da -2 a +4) ed il font (con FACE) è possibile cambiare il colore di solo alcune parole del testo

```
<FONT COLOR="#xxxxxx">...</FONT>
```



il colore è espresso in esadecimale secondo lo stesso criterio utilizzato nel tag BODY.

Immagini

Font Colori e caratteri

Il linguaggio HTML 3.2 permette di variare la dimensione dei caratteri anche nel documento oltre che negli header, permette il grassetto, il corsivo ed il sottolineato da la possibilità di utilizzare alcune font predefinite. Gli effetti ottenibili e le sintassi da utilizzare sono contenute in questa tabella:

Effetto	Sintassi
Dimensione dei caratteri	...
Dimensione dei caratteri	...
<i>Corsivo</i>	<I>... </I>
Grassetto	...
<u>Sottolineato</u>	<U>... </U>
Typewriter	<TT>... <TT>
Lampeggiante	<BLINK>... </BLINK>
<i>Enfasi</i>	...
Strong	...
Sample	<SAMP>... <SAMP>
Code	<CODE>... /CODE>
Keyboard	<KBD>... </KBD>
<i>Variabile</i>	<VAR>... </VAR>
<i>Definition</i>	<DFN>... </DFN>
<i>Citazione</i>	<CITE>... </CITE>
<i>Address</i>	<ADDRESS>... </ADDRESS>

I tags sopraelencati possono essere combinati fra loro per ottenere vari effetti.

HTML 3.2 supporta i font true type (sempre che questi siano supportati dal sistema, altrimenti vengono ignorati) per utilizzarli si utilizza

...

Sempre utilizzando il tag font oltre alla dimensione (con SIZE da -2 a +4) ed il font (con FACE) è possibile cambiare il colore di solo alcune parole del testo

...

il colore è espresso in esadecimale secondo lo stesso criterio utilizzato nel tag BODY.

Immagini

Immagini

La maggior parte dei browser presenti supportano le immagini in formato GIF e JPEG è possibile quindi migliorare le proprie pagine inserendo della grafica, il tag di base per inserire un file grafico è questo



``

Con lo svilupparsi dell'HTML anche il tag IMG si è evoluto arricchendosi di flags che permettono un ampio controllo sull'immagine inserita, il tag completo risulta

```
<IMG SRC="nomefile" LOW SRC="nomefile" HEIGHT="x" WIDTH="x"
ALT="nome_alternativo" ALIGN="x" ISMAP USEMAP="#nome_mappa" VSPACE="x"
HSPACE="x" BORDER="x">
```

i vari flags (oltre al già citato IMG SRC) definiscono i seguenti attributi:

- LOW SRC un file grafico alternativo in bassa risoluzione che verrà caricato prima dell'immagine vera e propria;
- HEIGHT e WIDTH le dimensioni in pixel se si desidera riscalare il file;
- ALT il nome alternativo che verrà visualizzato dai browser non grafici;
- ALIGN l'allineamento dell'immagine rispetto al testo che la circonda (RIGHT, LEFT, ABSBOTTOM, ABSMIDDLE, TEXTTOP, MIDDLE, BOTTOM);
- ISMAP nel caso che l'immagine sia linkata ad un file di mappa che rende l'immagine clickabile;
- USEMAP il nome della mappa nel caso che si utilizzi una mappa client-sided (direttamente inserita nell'HTML);
- VSPACE la distanza verticale dal testo (espressa in pixels);
- HSPACE la distanza orizzontale dal testo (in pixels);
- BORDER la dimensione del bordo dell'immagine (sempre in pixels).

GIF e JPEG

La scelta fra i formati grafici non è molto ampia, dobbiamo perciò valutare i pro ed i contro dei due formati a disposizione.

Il formato JPEG è molto compresso ed i file risultano più piccoli (a favore della maggiore velocità di caricamento e minore ingombro sul web), ma la compressione JPEG è di tipo loseless, questo significa che la qualità delle immagini JPEG risulta inferiore rispetto alle GIF; le immagini in formato GIF hanno inoltre (nel nuovo formato 89a) due caratteristiche molto importanti per il web, la progressione del caricamento e la trasparenza. La progressione del caricamento fa sì che l'immagini si carichi dapprima in bassa risoluzione ed acquisti definizione man mano che si carica (le JPG al contrario caricano direttamente la loro versione completa dell'immagine), questo permette di avere velocemente un preview dell'immagine che visualizzeremo e quindi decidere se bloccare il caricamento; la trasparenza permette a certe zone dell'immagine (definite durante la sua creazione) di risultare trasparenti sullo sfondo della pagina, con meno disturbo ed un effetto migliore.

Tabelle

Le Tables

Le tabella sono una delle più utili novità presenti nell'HTML 3. Permettono di riassumere dati come semplici tabelle e possono essere usate anche come sommari o mappe (quella presente all'inizio di questo documento).

Per creare una tabella utilizzare il contrassegno `<TABLES>... </TABLES>` al cui interno definirete i dati. Vediamo un esempio:

Prima colonna	Seconda colonna
Dati della prima colonna	Dati della seconda colonna



La sintassi per creare una semplice tabella come questa è:

```
<TABLE BORDER=3>
<TH ALIGN=center>Prima colonna</TH><TH ALIGN=center>Seconda
colonna</TH><TR></TR>
<TD ALIGN=center>Prima colonna</TD><TH ALIGN=center>Seconda
colonna</TD><TR></TD>
</TABLE>
```

Abbiamo così definito una tabella con bordo (3) con degli Header (TH) e dei dati (TD), organizzata su due colonne (il tag TR permette di decidere quando andare a capo). I dati sono centrati nelle colonne (ALIGN=center).

E' inoltre possibile scegliere la profondità della tables (Nella dichiarazione TABLE aggiungere WIDTH= seguito dalla percentuale di profondità), il Cell spacing, cioè la distanza fra testo e celle, (aggiungendo CELLSPACING=); il Cell padding, lo spessore dei divisori delle caselle, (aggiungendo CELLPADDING=); per i dati possiamo decidere l'allineamento orizzontale e verticale (ALIGN e VALIGN) il colspan ed il rowspan, cioè se la cella deve essere larga più di una cella standard (COLSPAN e ROWSPAN); inoltre se non vogliamo che i dati vengano disposti su più righe aggiungiamo NOWRAP;

per modificare il colore della tabella si utilizza BGCOLOR (con i valori esadecimali come al solito).

Per gli header possiamo scegliere l'ALIGN il COLSPAN ed il ROWSPAN. Vediamo un esempio più complesso:

Prima colonna	Seconda colonna	Terza colonna
dati prima/seconda colonna		dati seconda colonna
Ancora la prima	... la seconda	ed ecco la terza!!

Questa tabella è stata creata così:

```
<TABLE BORDER=1 CELLSPACING=5 CELLPADDING=5>
<TH ALIGN=right COLSPAN=1 ROWSPAN=1>Prima colonna</TH><TH ALIGN=left COLSPAN=1
ROWSPAN=1>Seconda colonna</TH><TH>Terza colonna</TH><TR></TR>
<TD ALIGN=center VALIGN=top COLSPAN=2 ROWSPAN=1>dati prima/seconda
colonna</TD><TD ALIGN=left VALIGN=bottom COLSPAN=1 ROWSPAN=1>dati seconda
colonna</TD><TR></TR> <TD ALIGN=center>Ancora la prima</TD><TD>... la
seconda</TD><TD>ed ecco la terza!!</TD> </TABLE>
```

I valori di default per CELLSPACING, CELLPADDING, COLSPAN e ROWSPAN son uguali ad 1; quello di WIDTH è uguale a 80%; BORDER è 0; l'ALIGN dell'Header è center; l'ALIGN dei dati left e il VALIGN è middle.

E' possibile agire con tutti questi comandi su ogni singola cella della tabella inserendo i vari flag (ALIGN, WIDTH, BGCOLOR...) nel tag <TD> invece che direttamente nel <TABLE>, ottenendo così una varietà di effetti molto ampia.

Frames

Le Frames

Un'altra delle grosse innovazioni delle ultime versione dell'HTML sono le frames (supportate da Netscape -v. 2 e successive- e Internet Explorer -v. 3 e successive-);



In pratica si tratta di documenti divisi al loro interno in sotto documenti totalmente indipendenti fra loro, le frames dividono infatti la finestra del documento in più aree logiche, contenenti files HTML, aggiornabili separatamente, permettendo perciò la creazione di documenti molto complessi e spesso molto funzionali (sempre che l'utente disponga di un browser adatto alla loro visualizzazione). Per rendere il tutto ancora più versatile, si possono includere dei FRAMESET al FRAMESET principale

I documenti contenenti frames son di un tipo un po' particolare in quanto all'interno del loro sorgente non è più presente il tag `<BODY>...</BODY>`, ma un sostitutivo `<FRAMESET>...</FRAMESET>`, il documento base si presenterà quindi così:

```
<HTML>
<HEAD>
</HEAD>
<FRAMESET>
...
</FRAMESET>
</HTML>
```

All'interno del tag frameset potranno essere presenti solo questi tag: `<FRAME>`, `<FRAMESET>`, `<NOFRAMES>`.

Sintassi

`<FRAMESET>` viene personalizzato seguendo questo schema:

```
<FRAMESET ROWS='value' COLS='value' BORDER='value' BORDERCOLOR='#xxxxxx'
FRAMEBORDER='value'>
```

i cui flags corrispondono a:

ROWS è una lista di valori separati da virgole, che definiscono la dimensione delle divisioni verticali della pagina, i valori possono essere espressi in percentuale, in pixel, ed un * (detto frame relativa), che occuperà il restante spazio se la somma dei pixel non è sufficiente a coprire l'ampiezza della pagina o la somma percentuale è inferiore a 100%;

COLS funziona come rows, ma divide la pagina in colonne;

BORDER definisce lo spessore del bordo in pixel (Solo Netscape)

BORDERCOLOR definisce il colore del bordo (Solo Netscape)

FRAMEBORDER definisce lo spessore globale per i bordi che separano le frames (Solo Netscape)

Una volta definito il frameset si inseriscono all'interno vi si definiscono le varie frame e le loro caratteristiche con il tag `<FRAME>`

```
<FRAMESET>
<FRAME...>
</FRAMESET>
```

L'esatta sintassi per il tag `<FRAME>` è:



```
<FRAME SRC="url" NAME="nome" MARGINWIDTH="value" MARGINHEIGHT="value"
SCROLLING="yes-no-auto" NORESIZE FRAMEBORDER="value" FRAMESPACING="value"
BORDERCOLOR="#xxxxxx">
```

SRC l'url del file che verrà mostrato nella frame;

NAME il nome a cui riferirsi come target se si dovesse usare un link, per aggiornare una unica frame;

MARGINWIDTH e MARGINHEIGHT è la distanza che il documento HTML contenuto nella frame avrà dai bordi della frame stessa

SCROLLING identifica la presenza di barre di scorrimento (si, no oppure automatiche);

NORESIZE permette alla frame di non essere ridimensionabile (attraverso il trascinarsi dei bordi);

FRAMEBORDER è supportato da Explorer in base al valore (0 non mostra bordi), mentre Netscape lo interpreta come yes|no, Netscape inoltre lo accetta anche nel <FRAMESET>;

FRAMESPACING è accettato solo da Explorer, e definisce lo spazio in pixel che si deve trovare attorno alla frame;

BORDECOLOR è accettato solo da netscape (anche nel <FRAMESET>) e definisci il colore del brodo della frame in base al codice esadecimale.

Dopo aver inserito le varie frames è possibile utilizzare il tag <NOFRAMES>...</NOFRAMES>, al cui interno andrà digitato il documento alternativo per quei browser che ancora non supportano le frames.

```
<HTML>
<HEAD>...</HEAD>
<FRAMESET...>
  <FRAME...>
  <FRAME...>
  ...
<NOFRAMES>
  ...
</NOFRAMES>
</FRAMESET>
</HTML>
```