

# HTML Reference

Marco Bonvini

10 ottobre 2009

## Indice

<b>1</b>	<b>Struttura del codice</b>	<b>2</b>
1.1	I Tags	2
1.1.1	Utilizzo dei Tags	2
1.1.2	Le opzioni di un Tag	2
1.1.3	Tags multipli	3
1.2	I commenti	3
<b>2</b>	<b>Descrizione dei vari Tags</b>	<b>5</b>
2.1	Tags senza chiusura	5
2.1.1	Tag per le Immagini	5
2.2	Formattazione del testo	6
2.2.1	Tags semplici (senza opzioni)	6
2.2.2	Paragrafo	6
2.2.3	Carattere	6
2.3	Links	7
<b>3</b>	<b>Entities</b>	<b>8</b>

# 1 Struttura del codice

Il codice HTML è concettualmente molto semplice: tutto ciò che viene scritto viene visualizzato, a meno che non sia racchiuso tra 2 caratteri molto speciali: < e >.

Questi due caratteri sono ciò che *fa la differenza*. All'interno dei caratteri < e > dovrà essere inserita la parte di codice vero e proprio.

## 1.1 I Tags

Tra ogni coppia di < e > dovrà essere inserito un Tag: un Tag è un comando che permette di fare un'azione singola. La difficoltà dell'HTML sta giusto nel conoscere tutti i numerosi Tags che esistono e le loro opzioni.

### 1.1.1 Utilizzo dei Tags

Quasi tutti i Tags vanno *aperti e chiusi*. Ciò significa che esiste un modo per *chiudere* un Tag: occorre riscrivere il Tag ma preceduto da uno slash /. In pratica

```
<TAG>Testo a cui sarà applicato il tag</TAG>
```

Ad esempio, per scrivere del testo in grassetto, si usa il Tag <B> (dall'inglese *bold*)

Questo è il modo di scrivere in <B>grassetto</B>

(la parola "grassetto" sarà visualizzata in **grassetto**, il resto della frase no)

### 1.1.2 Le opzioni di un Tag

La maggior parte dei Tags accettano anche delle opzioni; esse semplicemente vanno inserite subito dopo il Tag, separate da esso e tra di loro da uno spazio:

```
<TAG OPTION_1 OPTION_2>Testo a cui sarà applicato il tag</TAG>
```

Come si vede dall'esempio alla chiusura del Tag le opzioni non devono essere ripetute.

Ad esempio, per scrivere del testo in rosso e in carattere "Arial", si può procedere così:

```
<FONT COLOR="red" FACE="Arial">Questo è scritto in rosso e in Arial</FONT>
```

Come si vede le opzioni COLOR e FACE sono seguite da un uguale = (simbolo di assegnazione) e dal valore racchiuso tra virgolette ": sono poche le opzioni che non richiedono valori (io non ho intenzione di mostrarne alcuna di queste poche), quindi è importante puntualizzare due punti importanti:

- è sempre meglio racchiudere il valore tra virgolette "; se il valore è costituito da una singola parola, senza spazi, in teoria si può fare a meno delle virgolette; ma se nel valore sono presenti spazi o altri caratteri particolari le virgolette sono necessarie;
- le virgolette " possono essere sostituite da apici ' ; questo è utile se il valore deve contenere uno di quei due caratteri: in tal caso si usa l'altro;

Per un esempio relativo al secondo punto mi basta dire che nel Tag per l'inserimento di immagini esiste un'opzione, ALT, che permette di visualizzare del testo alla sosta del mouse sull'immagine; se voglio visualizzare un apostrofo (cioè un apice) devo usare le virgolette per impostare il valore:

```
ALT="Questa è un'immagine"
```

Per maggiori dettagli si veda il §2.1.1, relativo all'inserimento di immagini.

### 1.1.3 Tags multipli

È possibile (di fatto si fa sempre) aprire diversi Tags in cascata, per sovrapporre le opzioni. Ad esempio se voglio scrivere una frase in grassetto e una parola anche in corsivo (il Tag è I, dall'inglese *italic*) posso fare così:

```
<B>Solo grassetto, <I>grassetto e corsivo,</I> di nuovo solo grassetto.</B>
```

il risultato sarà una cosa del genere:

**Solo grassetto, *grassetto e corsivo*, di nuovo solo grassetto.**

ATTENZIONE: è fondamentale che l'ordine della apertura e chiusura dei Tags sia coerente: cioè che l'ultimo ad essere stato aperto sia il primo ad essere chiuso.

In questo senso i Tags si *aprono* e si *chiudono* come le parentesi in matematica.

Per intenderci: se volessi scrivere prima in grassetto, poi in grassetto-corsivo e poi solo in corsivo **non posso** fare così:

```
<B>Solo grassetto, <I>grassetto e corsivo,</B> solo corsivo</I>
```

Uno dei modi *corretti* per ottenere il risultato desiderato è questo:

```
<B>Solo grassetto, <I>grassetto e corsivo,</I></B><I> solo corsivo</I>
```

che restituisce il valore desiderato

**Solo grassetto, *grassetto e corsivo*, solo corsivo.**

Una cosa interessante è che si può aprire in cascata lo stesso Tag: questo ha un'utilità solo se il Tag accetta opzioni, e se alle diverse aperture le opzioni date sono diverse. Ad esempio per scrivere una frase in rosso a dimensione 3 e una sola parola a dimensione 4 si può fare così:

```
<FONT COLOR="red" SIZE="3">Questa è una <FONT SIZE="4">grande</FONT> frase.</FONT>
```

che verrà visualizzata più o meno così:

Questa è una **grande** frase.

## 1.2 I commenti

I commenti per definizione sono parti di testo che non devono essere visibili, nel risultato, ma solo nel codice. Per commentare parti di testo basterà utilizzare un apposito Tag; ma la potenzialità dei commenti non viene dalla possibilità di commentare testo (cosa che può interessare solo il creatore della pagina (in realtà ad uno stadio più avanzato permette anche dell'altro)) ma dalla possibilità di commentare codice, cioè gli altri Tags. Un Tag commentato non verrà interpretato dal Browser (ciò ha particolare utilità in fase di creazione della pagina per effettuare modifiche provvisorie).

Per commentare si fa così:

```
<!-- Testo commentato -->  
<!-- testo e <B>tag</B> commentati -->  
<!-- <B> -->Il testo qui in origine era grassetto<!-- </B> -->
```

Nell'ultimo esempio si vede una parte di testo che in origine era in grassetto e che ora risulta normale perché il Tag B è stato commentato.

ATTENZIONE: è necessario che sia commentata sia l'apertura di un Tag che la sua chiusura, per evitare incoerenze e danni al codice che segue (si veda il terzo esempio).

Quello che non si può fare è commentare solo un'opzione dentro un Tag.

## 2 Descrizione dei vari Tags

### 2.1 Tags senza chiusura

Ci sono pochi Tags che ci interessano che non possono essere chiusi, perché rappresentano un'azione che eseguono lì dove sono inseriti e non un attributo da applicare al testo che segue.

- `<BR>` Break Line: va a capo (utilissimo);
- `<HR>` Horizontal Rule: inserisce una riga orizzontale, che va da parte a parte della pagina.

I Tags sopra elencati non hanno opzioni.

#### 2.1.1 Tag per le Immagini

Per inserire un'immagine occorre usare il Tag `IMG`, che non ha chiusura:

```
<IMG SRC="path">
```

L'opzione `SRC` è indispensabile. Il valore, `path`, deve essere il percorso dell'immagine (relativo o assoluto). Gli utenti Windows non hanno molta dimestichezza con la scrittura del `path`, visto che questo va scritto in formato Unix. A scopo di completezza do alcune direttive sulla scrittura del `path` in formato Unix:

- `./` indica la cartella corrente;
- `../` indica la cartella immediatamente prima della cartella corrente;
- per salire di più di una cartella occorre ripetere più volte `../`: ad esempio per salire di tre cartelle occorre scrivere `../../../`;
- una volta posizionati si scriva il `path` semplicemente usando i nomi delle cartelle separati da slash `/`;
- gli utenti Windows spesso sono invogliati a chiamare files e cartelle con nomi contenenti spazi: per ovviare al problema precedere lo spazio con un back slash `\`.

Ci sono poi una serie di opzioni facoltative; le più utili sono:

- `ALT="fraseimmagine"` visualizza fraseimmagine alla sosta del mouse sull'immagine;
- `ALIGN="left"` indica la posizione dell'immagine: può essere `left`, `right` o `center`;
- `WIDTH="larghezza"` e `HEIGHT="altezza"` specificano le dimensioni dell'immagine; si deve usare solo se si vuole una dimensione diversa dall'originale;
- `BORDER="0"` indica la dimensione del bordo; se è `0` non ci saranno bordi, altrimenti il bordo avrà la dimensione specificata (in pixels): `1`, `2`, `3`, `4`, ... (`4` è già enorme).

## 2.2 Formattazione del testo

### 2.2.1 Tags semplici (senza opzioni)

Precisando che in effetti si possono avere alcune opzioni avanzate in tutti i Tags, elencherò adesso una serie di Tags che normalmente si usano senza specificare alcuna opzione:

- <B> Bold: grassetto;
- <I> Italic: corsivo;
- <U> Underlined: sottolineato;
- <BIG> Big: carattere di grossa dimensione;
- <SMALL> Small: carattere di piccola dimensione;
- <H1> Level-1 Heading: Titolo molto molto grande;
- <H2> Level-2 Heading: Titolo molto grande;
- <H3> Level-3 Heading: Titolo grande;
- <H4> Level-4 Heading: Titolo dimensione normale;
- <H5> Level-5 Heading: Titolo piccolo;
- <H6> Level-6 Heading: Titolo molto piccolo;
- <CENTER> Center: centrato: vale per testo, immagini e tutto il resto.

Tutti i Tags sopra elencati **vanno chiusi**.

### 2.2.2 Paragrafo

Se si sta scrivendo un testo può essere utile usare come “contenitore” del testo il Tag P, dall’inglese *paragraph*. Esso lascia un po’ di spazio sopra e sotto di sé, e inoltre permette di scegliere l’allineamento del testo (la sua unica opzione); scelte possibili sono left, right e center e justify.

```
<P ALIGN="justify">Testo di un paragrafo giustificato.</P>
```

### 2.2.3 Carattere

Per modificare il carattere e gli attributi dello stesso si usa il Tag FONT.

```
<FONT COLOR="red">Testo in rosso.</FONT>
```

Esso accetta le seguenti opzioni:

- COLOR="colore" specifica il colore del testo; il colore deve essere scritto in inglese (se è un colore standard) o in formato RGB <sup>1</sup> preceduto da # (ad esempio il rosso è COLOR="#FF0000");

---

<sup>1</sup>Nei pc ordinari sono visualizzabili  $2^{24} = 16777216$  colori, ottenibili come combinazioni di 3 colori, Red, Green, Blue (RGB). Nella specificazione del colore si immagina di aggiungere al nero (assenza di colore) delle parti di ciascun colore. Ogni colore può dare  $256 = 2^8 = 16^2$  contributi differenti, cosicché nel complesso si hanno infatti  $256^3 = 16^6 = 2^{24}$  combinazioni possibili.

Che 256 sia proprio  $16^2$  non è un caso: infatti in tal modo, in esadecimale il contributo di ciascun colore può variare da 0 a FF, cioè da 0 a 255.

Però così si può scrivere la combinazione dei 3 colori come se fosse un unico numero, di 6 cifre esadecimali, e non 3 distinti numeri per un totale di 9 cifre decimali.

Ad esempio, il viola, corrisponde a FF00FF che significa FF, 00, FF, cioè tutto rosso, niente verde, tutto blu.

- `SIZE="dimensione"` specifica la dimensione del testo; le dimensioni possibili sono tutti gli interi tra 1 e 7 (1 è piccolo e 7 è grande);
- `FACE="carattere 1, carattere 2, carattere 3, ..."` specifica il carattere da usare; se il browser conosce carattere 1 lo usa, altrimenti prova con carattere 2 e così via. Se non conosce nessuno dei caratteri ne usa uno di default.  
 CONSIGLIO: Se si usa un carattere poco famoso, di quelli molto strani che si scaricano da internet o si installano con i programmi, è molto utile aggiungere alternative a proprio piacimento. Se invece si usa un carattere conosciutissimo, come ad esempio il Courier, basta solo quel valore.

## 2.3 Links

Per rendere una parte di testo o un'immagine o altri tipi di oggetti (di cui non parlerò) dei collegamenti a pagine o ad altro (per esempio per scrivere ad un indirizzo e-mail) si usa il Tag **A**, dall'inglese *anchor*.

```
<A HREF="www.ctan.com">Link a CTAN</A>
```

Esso accetta le seguenti opzioni:

- `HREF="indirizzo"` è l'opzione necessaria per abilitare il link; indirizzo può essere un indirizzo internet o un path relativo o assoluto in formato Unix (vedi §2.1.1); per inviare un'e-mail occorre scrivere `HREF="mailto:indirizzo@domain.com"` dove `indirizzo@domain.com` è l'indirizzo e-mail del destinatario;
- `TARGET="_blank"` apre il link in una nuova pagina (è molto utile); sono possibili altri valori oltre a `_blank` ma non sono indispensabili per la attuale trattazione.

### 3 Entities

Una particolare attenzione va posta in come HTML riesce a visualizzare caratteri speciali. Per esempio, se voglio visualizzare `< o >` come faccio?

Se li scrivo il browser penserà che sto dando dei Tags, quindi ci deve essere un altro modo.

HTML prevede una particolare codifica per la visualizzazione di tutti i caratteri: ad esempio per visualizzare `<` dovrò scrivere `&lt;`; e per `>` `&gt;`; e, come si può intuire, ogni carattere si scrive come `&entity;` dove `entity` è il particolare nome del carattere.

Questa tecnica pone il nuovo problema di come visualizzare il carattere `&`, che si risolve scrivendo `&amp;`; e null'altro.

Particolare interesse si ha per le vocali accentate: queste, scritte nel codice, non piacciono ad alcuni browser. Per questo esistono apposite entities molto semplici per ottenerle; basta sapere che:

- l'accento grave si ottiene aggiungendo `grave` e quello acuto aggiungendo `acute` alla vocale;
- la vocale accentata risulterà maiuscola o minuscola a seconda che sia maiuscola o minuscola nell'entity.

Quindi ad esempio la lettera “perché” si scrive `perch&acute;`; e un “È bello così” a inizio di frase si scrive `&Egrave; bello cos&igrave;`.