

## Introduzione al linguaggio HTML

Con il diffondersi del World Wide Web, una delle più diffuse applicazioni disponibili sulla rete Internet, si è sviluppato anche uno speciale linguaggio denominato Hypertext Markup Language (HTML) utilizzato per costruire i documenti presenti nel Web.

HTML, che tradotto in italiano suona come Linguaggio di Contrassegno Ipertestuale, rientra nella categoria dei linguaggi a marcatori; esso è cioè costituito da un insieme di istruzioni (dette marcatori, o tag), che servono a descrivere la struttura del documento.

In altre parole, un documento HTML è un file di testo composto da una serie di tag (e relativi attributi), attraverso i quali il documento presenta testo, immagini statiche, animazioni, suoni, così da creare un documento multimediale.

Ciò che si richiede quindi per scrivere un documento in HTML è la conoscenza della sintassi con cui i marcatori vengono introdotti all'interno del documento.

Un documento HTML non necessita di un particolare software o hardware per essere sviluppato. Ciò significa che tale documento potrà essere reso disponibile su diverse piattaforme e senza il vincolo del possesso di un particolare programma; è sufficiente un semplice elaboratore di testi, come il Blocco Note di Windows o il SimpleText sul Mac, e il salvataggio del relativo file con un'estensione *htm* o *html*.

Esistono alcuni programmi, i cosiddetti editor HTML, che consentono di realizzare documenti HTML in modo molto meno faticoso e con ottimi risultati. È comunque consigliabile acquisire una conoscenza dettagliata del linguaggio HTML prima di affidarsi completamente ad un editor HTML, che in ogni caso presenterà dei limiti intrinseci.

Fra gli editor HTML troviamo ad esempio Netscape Composer, incluso nel pacchetto di Netscape Communicator, Microsoft FrontPage e Macromedia Dreamweaver; anche con le ultime versioni di Microsoft Word è possibile creare una nuova pagina Web a partire da un documento di Word ed usare numerosi modelli che rappresentano i tipi più comuni di pagine Web.

Due sono le caratteristiche fondamentali del linguaggio HTML, riflesse anche dallo stesso nome:

- HTML è un linguaggio ipertestuale che permette di stabilire collegamenti (link) all'interno di uno stesso documento oppure fra più documenti nello stesso sito Web o in siti diversi.
- HTML è un linguaggio a marcatori che incorpora, oltre all'informazione che si vuole comunicare attraverso la pagina Web, particolari sequenze di caratteri (marcatori, tag) che descrivono la struttura e il comportamento del documento.

I Tag si dividono essenzialmente in due categorie:

- strutturali. Il loro utilizzo consente di determinare la forma del documento; essi, in altre parole, definiscono il documento stesso, i paragrafi di intestazione e di corpo, i titoli delle pagine Web ed altri identificatori del documento.
- di formattazione. Il loro utilizzo consente l'organizzazione e la formattazione dei documenti; essi consentono la scrittura di un testo colorato o in corsivo, di disegnare linee orizzontali o aggiungere tabelle.

I tag, per essere riconosciuti da un browser, sono delimitati dai separatori angolari "<" e ">"; alcuni tag, inoltre, hanno bisogno di un tag gemello, detto tag di fine, il cui compito è quello di delimitare la parte del documento a cui applicare la proprietà del tag stesso.

Per distinguere il tag di inizio da quello di fine, a quest'ultimo viene aggiunto il simbolo "/". Nell'esempio che segue:

**<B>**Questo testo è scritto in **<BR>**grassetto, **</B>** il seguente no....

la parte di testo delimitata dai tag **<B>** e **</B>** sarà visualizzata in grassetto; la parola grassetto, invece, sarà visualizzata alla riga successiva, poiché preceduta dal tag **<BR>**; se fosse stato ommesso il tag **</B>**, anche il resto del documento sarebbe stato visualizzato in grassetto. Si noti, infine, come il tag **<BR>** non sia accompagnato dal relativo tag di chiusura.

La maggior parte dei tag, per essere applicati in maniera ottimale, sono accompagnati da *Attributi* che hanno il compito di fornire informazioni aggiuntive alle proprietà dei tag stessi. Un attributo è definito in genere da un nome e da un valore, separati da un segno di uguale (=), anche se, in alcuni casi, è richiesto il solo nome dell'attributo.

Ad esempio, il tag **<BODY>**, comprensivo dell'attributo per la specificazione del colore giallo dello sfondo, assume la seguente forma:

**<BODY BGCOLOR=#FFFF40>**

La struttura fondamentale di un documento HTML è la seguente:

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

Il tag **<HTML>** indica l'inizio del documento e il corrispondente tag di chiusura **</HTML>** ne indica la fine; questo tag comunica al browser che il file prelevato è stato redatto secondo quel linguaggio di contrassegno che è l'HTML.

Se si omettessero i due tag **<HTML>**, alcuni browser potrebbero non caricare il documento o rappresentarlo in maniera non corretta; la causa di tale anomalia è che HTML non è l'unico linguaggio di contrassegno esistente, per cui esso potrebbe essere confuso, ad esempio, con il linguaggio XML (Extensible Markup Language).

Una volta definiti i limiti del documento si passa alla suddivisione dello stesso in testata e corpo.

tag **<HEAD>** e **</HEAD>** racchiudono l'intestazione del documento dove si possono inserire informazioni che non vengono visualizzate nella pagina, ma che ne descrivono caratteristiche come titolo, autore e parole-chiave. Quando il browser incontra questi simboli capisce che si tratta d'istruzione da eseguire e non di testo da visualizzare sullo schermo.

I tag **<BODY>** e **</BODY>** delimitano l'inizio e la fine del contenuto della pagina Web; fra questi tag è compreso tutto il testo che compare nella pagina Web, inclusi i vari collegamenti.

Il codice base di ogni pagina web può contenere altri elementi aggiuntivi come:

- **<!DOCTYPE attributi>**: questo tag serve a dichiarare la versione HTML su cui il documento è basato e va inserito all'inizio del file prima di **<HTML>**....**</HTML>**.
- **<!-- commento -->**: questo tag consente di commentare alcune parti del codice della pagina e può essere inserito in qualsiasi punto.

Quando nel codice HTML si inseriscono questi tag aggiuntivi bisogna fare sempre attenzione a non accrescere esageratamente ed inutilmente le dimensioni della pagina che incidono sensibilmente sui tempi di trasmissione, caricamento e visualizzazione nel browser.

## I tag di intestazione

L'intestazione di un file HTML è definita attraverso i tag di apertura e chiusura **<HEAD>** e **</HEAD>**. In essa possono essere contenute alcune informazioni fondamentali sul file stesso come il titolo, il formato del file, la data dell'ultima modifica o le parole chiave; tali informazioni possono essere utilizzate dai motori di ricerca senza dover caricare tutto il documento.

I tag più utilizzati nella sezione HEAD sono: **Title, Base, Link, Meta**.

Il tag **<TITLE>**, insieme al suo gemello **</TITLE>**, ha il compito di visualizzare, sulla barra del titolo del browser, il testo inserito fra di essi. Tenendo conto che alcuni browser inseriscono anche il proprio nome sulla barra del titolo, è opportuno che il titolo della pagina Web sia breve in modo da essere visualizzato completamente, ma non troppo per evitare di essere incomprensibili.

In mancanza di un titolo, i browser utilizzano generalmente il nome del file HTML.

I collegamenti ipertestuali all'interno di un documento HTML sono riferiti automaticamente allo stesso server in cui è posto il documento stesso. Attraverso il tag **<BASE>**, è possibile definire, per i collegamenti relativi, una posizione diversa, sia essa un'altra directory o un server completamente diverso.

Per ottenere ciò, occorre associare al tag **<BASE>** l'attributo HREF secondo la sintassi:

```
<BASE HREF = "protocollo/nomedelserver">
```

dove protocollo rappresenta uno standard di comunicazione come HTTP e nomedelserver rappresenta il nome del server predefinito (ad esempio www.microsoft.com) oppure il relativo indirizzo IP (ad esempio 194.94.24.2).

Il tag **<LINK>** è utilizzato per il collegamento a fogli stile esterni al documento:

```
<LINK REL="stylesheet" type="text/css" href="stili.css">
```

I server Web inviano normalmente, oltre alla pagina Web così come appare, una serie di intestazioni per aiutare il browser ad interpretare il documento stesso. Il tag **<META>** è utilizzato per passare informazioni sulla gestione del documento e fornire dati utilizzati dai motori di ricerca, senza dover caricare completamente tutto il documento; esso è accompagnato da tre attributi: **HTTP-EQUIV, NAME, CONTENT**.

Se il tag utilizza l'attributo HTTP-EQUIV, l'informazione ad esso associata è inclusa nell'intestazione di risposta generata dal server, se invece il tag utilizza l'attributo NAME, l'informazione ad esso associata rappresenta un ausilio per gli utenti che consultino il sorgente del documento; l'attributo CONTENT rappresenta il reale valore contenuto nel tag.

Per meglio comprendere quanto scritto finora, è opportuno fornire qualche esempio:

1. **<META NAME=author CONTENT="Ing. Antonio Guadagno">**

dove il valore **author** indica la persona che ha creato la pagina oppure il nome dell'editor HTML utilizzato.

Per inserire nell'intestazione una descrizione sommaria del documento si utilizza:

2. **<META NAME=description CONTENT=" Ing. Antonio Guadagno – Sito di interfaccia con i clienti">**

Per inserire nell'intestazione un elenco di parole-chiave che specificano l'argomento della pagina si utilizza:

3. **<META NAME=keywords CONTENT="software, reti, multimedialità, formazione, web, internet, archiviazione ottica">**

Il valore **keyword** indica al browser quali sono i termini utilizzabili dai motori di ricerca per l'individuazione della pagina. Le parole chiave possono essere separate sia da spazi che da virgole; alcuni motori di ricerca prediligono la prima tecnica, altri la seconda.

4. `<META HTTP-EQUIV=refresh CONTENT=15>`

Il valore **refresh** indica al browser di ricaricare il documento dopo 15 secondi; tale operazione è richiesta quando la pagina Web viene aggiornata costantemente come avviene ad esempio per le quotazioni in borsa.

5. `<META HTTP-EQUIV=reply-to CONTENT=info@seneta.it>`

Il valore **reply-to** indica l'indirizzo di e-mail che gli utenti possono utilizzare.

6. `<META NAME=copyright CONTENT="2004 Ing. Antonio Guadagno">`

Il valore **copyright** indica che la pagina è protetta dalla legge sul copyright.

## Il corpo di una pagina Web

La sezione `<BODY>` è il corpo principale del documento HTML dove vanno inseriti tutti i contenuti che devono apparire nella pagina web. Alcune caratteristiche generali come lo sfondo della pagina, il colore del testo ed il colore dei collegamenti ipertestuali, possono essere specificate mediante i relativi attributi.

Per impostare un colore come sfondo della pagina web si usa l'attributo **BGCOLOR**. Ad esempio:

`<BODY BGCOLOR="#000000">`

dove il codice del colore è in formato esadecimale. Si può indicare il colore mediante il nome in inglese anziché il formato esadecimale:

`<BODY BGCOLOR="black">`

Questa però non è una pratica comune in quanto non tutti i browser sono in grado di interpretare correttamente tale genere di espressione.

È possibile impostare come sfondo della pagina web un'immagine in formato GIF o JPEG utilizzando l'attributo **BACKGROUND** secondo la sintassi:

`<BODY BACKGROUND="percorso e nome del file">`

Se l'immagine si trova nella stessa cartella della pagina HTML e si chiama sfondo.gif occorrerà scrivere:

`<BODY BACKGROUND="sfondo.gif">`

Inserendo un'immagine come sfondo della pagina web, questa viene automaticamente replicata dal browser in modo da coprire l'intero spazio visibile. È allora opportuno scegliere un'immagine che si presti ad essere replicata e affiancata in un numero indefinito di copie, senza che si vedano le giunture o si creino stacchi poco estetici.

Per impostare a priori il colore del testo della pagina si usa l'attributo **TEXT** secondo la sintassi:

`<BODY TEXT="#codice del colore">`

Se non specificato, il colore di default del testo è il nero.

Tramite gli attributi del tag `<BODY>` si possono definire anche i colori dei collegamenti ipertestuali.

I colori da definire sono tre: per i collegamenti ipertestuali in condizioni normali si usa l'attributo **LINK**="codice del colore"; per i collegamenti attivi, nel momento cioè in cui vi si fa clic, si usa l'attributo **ALINK**="codice del colore"; per i collegamenti già visitati si usa l'attributo **VLINK**="codice del colore".

In pratica per personalizzare il colore dei collegamenti la sintassi è:

```
<BODY LINK="colore1" ALINK="colore2" VLINK="colore3">
```

## La gestione dei colori

Per riprodurre nel modo più fedele possibile la gamma dei colori visibili in natura si ricorre normalmente alla miscelazione di alcune tinte di base: dalla loro combinazione si ottengono tutte le sfumature intermedie. Questa miscelazione si basa su due diverse tecniche o metodi:

**La sintesi sottrattiva**, basata sull'assorbimento e dunque sulla sottrazione della luce da parte dei corpi fisici. Questa tecnica si applica per la stampa dei colori su carta o su altri supporti. I colori di base in questo caso sono il ciano, il magenta, il giallo e il nero; la sintesi sottrattiva viene perciò indicata con la sigla **CMYK** (Cyan, Magenta, Yellow, black).

**La sintesi additiva**, basata sull'emissione e dunque sull'addizione o sovrapposizione della luce grazie a dispositivo di proiezione. Questa tecnica si applica per la visualizzazione dei colori su schermi e monitor. I colori di base in questo caso sono il rosso, il verde e il blu; la sintesi additiva viene perciò indicata con la sigla **RGB** (Red, Green, Blue).

Poiché le pagine Web hanno come applicazione primaria la visualizzazione sui monitor, il metodo di riferimento per la gestione dei colori è la sintesi additiva (RGB). Ogni possibile tonalità di colore è definita da una combinazione di valori delle tre componenti di base: rosso, verde e blu.

In pratica, ogni colore è identificato da tre valori (numeri): il primo indica la "quantità" di rosso presente nel colore in questione; il secondo la "quantità" di verde; il terzo la "quantità" di blu.

Ciascuno di questi tre valori può variare da 0 (minimo: componente non presente) a 255 (massimo: componente dominante). Ad esempio il nero corrisponde ai valori RGB 0, 0, 0, il bianco corrisponde ai valori RGB 255, 255, 255.

Nel linguaggio HTML per indicare i valori RGB si utilizza il sistema di codifica esadecimale anziché quello decimale. I numeri esadecimali si basano su un sistema di numerazione in base 16 che comprende le cifre da 0 a 9 e da A a F.

Se paragonato al sistema di numerazione decimale, il sistema esadecimale ha un aspetto un po' strano poiché i numeri possono includere delle lettere. Questa piccola difficoltà non deve però scoraggiare.

I codici esadecimali dei colori sono costituiti da numeri di sei cifre. Le prime due cifre definiscono il livello del rosso, le seconde due cifre definiscono il livello del verde e la terza coppia di cifre definisce il livello del blu. Questi valori si possono usare in vari elementi del codice Html, ma devono essere sempre preceduti dal segno #.

Quando si scrive il codice Html è possibile specificare il nome dei colori al posto del valore esadecimale, ma non tutti i browser supportano questi nomi. In caso di dubbio è preferibile utilizzare i valori esadecimali.

Anche la visualizzazione dei colori può essere diversa a seconda del browser o del sistema operativo utilizzato. La stessa pagina (con gli stessi colori) potrebbe apparire diversa se visualizzata da un Pc o da un Mac, da un browser piuttosto che da un altro. Questo è purtroppo un limite intrinseco delle tecnologie Web e non c'è modo di garantirsi un'assoluta fedeltà di riproduzione di colori.

La tabella seguente mostra i 16 nomi dei colori riconosciuti dalla maggior parte dei browser, che sono stati elencati nelle specifiche Html 4.0 prodotte dal consorzio W3C.

Nome	Valore	Nome	Valore
Aqua	00FFFF	Navy	000080
Black	000000	Olive	808000
Blue	0000FF	Purple	800080
Fucsia	FF00FF	Red	FF0000
Gray	808080	Silver	C0C0C0
Green	008000	Teal	008080
Lime	00FF00	White	FFFFFF
Maroon	800000	Yellow	FFFF00

## Formattazione del carattere

Le prime versioni del linguaggio HTML così come le prime versioni dei browser consentivano unicamente la scelta tra due tipi di caratteri, uno a spaziatura variabile, analogo al Times, che è il tipo di carattere predefinito, ed uno a spaziatura fissa, analogo al Courier.

Nella versione 3.2 di HTML è stato introdotto il tag **<FONT>** che permette di personalizzare l'aspetto del testo tramite gli attributi **FACE** per impostare il tipo di carattere, **COLOR** per impostare il colore del testo e **SIZE** per impostare la dimensione del testo. La sintassi per impostare il tipo di carattere è:

```
<FONT FACE="tipo di carattere"> testo </FONT>
```

Bisogna considerare che è necessario che sul computer di chi visualizza la pagina sia installato il tipo di carattere indicato nel tag **<FONT>**. Infatti, se il browser non trova sul computer di chi naviga il carattere previsto lo sostituisce con quello predefinito, cambiando radicalmente lo stile della pagina.

Per aggirare il problema è possibile indicare una serie di caratteri alternativi secondo la sintassi:

```
<FONT FACE="Verdana, Arial, Helvetica, sans-serif"> testo </FONT>
```

In tal modo se il primo carattere della lista non è presente sarà visualizzato il secondo; se anche il secondo non è presente sarà visualizzato il terzo, e così via. Così facendo si ha un maggiore controllo sul tipo di carattere che deve essere visualizzato nella pagina web.

Per impostare il colore del testo, la sintassi è:

```
<FONT COLOR="#colore"> testo </FONT>
```

dove il colore va impostato nel formato esadecimale. Questo comando prevale sul colore eventualmente definito nel tag **<BODY>**.

Infine per modificare le dimensioni dei caratteri la sintassi è:

```
<FONT SIZE="valore"> testo </FONT>
```

dove il valore si esprime con un numero intero che può essere indicato in senso relativo o assoluto. In senso relativo il valore va da -3 (molto piccolo) a +3 (molto grande). In senso assoluto il valore va da 1 (molto piccolo) a 7 (molto grande).

Evidentemente tutti gli attributi del tag **<FONT>** esaminati possono essere combinati fra loro secondo la sintassi:

```
<FONT FACE="Arial, Helvetica" COLOR=#336699 SIZE=2> testo </FONT>
```

Il tipo di carattere utilizzato è uno degli elementi più importanti di un sito Web ben realizzato. Tutti coloro che visiteranno il sito Web risponderanno in modo istintivo a un determinato tipo di caratteri. Nella progettazione di pagine web la scelta dei caratteri da usare non è molto vasta ma bisogna limitarsi all'uso dei caratteri predefiniti residenti nella maggior parte dei sistemi operativi.

Come detto in precedenza, specificando un font che non è disponibile sul sistema dell'utente, il suo browser lo sostituirà con il font standard, che nella maggior parte dei browser è il "Times New Roman".

I caratteri predefiniti residenti nella maggior parte dei sistemi operativi sono: **Arial, Courier, Georgia, Times New Roman, Trebuchet e Verdana.**

La leggibilità su schermo dipende soprattutto dalla cosiddetta altezza x (l'altezza di una x minuscola) e dalla dimensione generale del carattere. Così un tipo di carattere come Times New Roman ha una leggibilità media sullo schermo del computer perché ha un'altezza x moderata. Questo carattere è consigliato nei documenti con molto testo che probabilmente saranno stampati e non letti sullo schermo.

Invece caratteri quali Georgia e Verdana sono stati progettati per essere letti su schermo, infatti hanno altezze x elevate e sono molto grandi, se paragonati a tipi di caratteri più tradizionali con la stessa dimensione di punti.

Le altezze x esagerate e la forma pesante di questi caratteri conferiscono però un aspetto massiccio e rozzo al documento se questo viene stampato. Lo schema più convenzionale per l'uso dei caratteri impone di utilizzare un carattere con grazie come Times New Roman o Georgia per il testo ed un carattere senza grazie come Verdana o Arial come contrasto per i titoli.

Questo schema può essere considerato come suggerimento o punto di partenza per la scelta dei caratteri delle proprie pagine web; ogni progettista web è libero di scegliere i caratteri da usare testandoli opportunamente sullo schermo per produrre un ragionevole equilibrio tra densità d'informazioni e leggibilità generale.

È consigliabile, comunque, utilizzare pochi font e poche dimensioni, per dare alle pagine un aspetto più chiaro, così come, per dare uniformità, si deve applicare lo stesso tipo di carattere e la stessa combinazione di stili in tutte le pagine di un sito Web evitando cambiamenti casuali.

Se si deve necessariamente usare un font ben preciso, non standard, si può creare il testo all'interno di un'immagine GIF o JPEG. Questa tecnica consente anche di applicare al testo ombreggiature ed altri effetti grafici.

Il testo grafico è indicato per creare loghi, intestazioni di pagina e immagini di navigazione.

---

I tag per la formattazione del testo si suddividono in stili fisici e stili logici.

Gli stili fisici sono tag che definiscono in modo univoco gli attributi grafici dei caratteri. I principali tag fisici sono (tutti hanno il tag gemello di chiusura):

<b>&lt;B&gt;</b>	testo in grassetto
<b>&lt;I&gt;</b>	testo in corsivo
<b>&lt;U&gt;</b>	testo sottolineato
<b>&lt;STRIKE&gt;</b>	testo barrato o depennato
<b>&lt;SUP&gt;</b>	testo apice
<b>&lt;SUB&gt;</b>	testo pedice

Usare il tag **<U>... </U>** è sconsigliabile perché un testo sottolineato può creare confusione con i collegamenti ipertestuali che solitamente appaiono sottolineati.

Gli stili logici sono tag che determinano l'aspetto del testo in base al presunto scopo della formattazione. I principali tag logici sono (tutti hanno il tag gemello di chiusura):

<b>&lt;ADDRESS&gt;</b>	indirizzo per contatti, il testo viene visualizzato in corsivo
<b>&lt;BLOCKQUOTE&gt;</b>	il testo viene rientrato di circa 1 cm dal margine sinistro. Si possono sommare più tag <BLOCKQUOTE> consecutivi per avere rientri maggiori di 1 cm. Si rammenti che il linguaggio HTML non permette di definire un rientro destro
<b>&lt;CITE&gt;</b>	citazione, il testo viene visualizzato in corsivo
<b>&lt;DFN&gt;</b>	definizione, il testo viene visualizzato in corsivo
<b>&lt;KBD&gt;</b>	Digitazione da tastiera, con visualizzazione a spaziatura fissa
<b>&lt;SAMP&gt;</b>	testo esemplificativo, visualizzato in carattere a spaziatura fissa
<b>&lt;STRONG&gt;</b>	rafforzamento, il testo viene visualizzato in grassetto
<b>&lt;VAR&gt;</b>	variabile, il testo viene visualizzato in corsivo

I tag logici rappresentano una soluzione elegante dal punto di vista formale, ma il loro risultato in fase di visualizzazione può variare secondo il browser dell'utente.

## Paragrafi ed allineamenti

Quando si scrive il codice HTML, per mandare a capo il testo che segue non basta premere il tasto Invio. Anche se nel codice una frase sembra andare a capo, ciò non accade una volta che la pagina viene visualizzata nel browser. Per mandare a capo il testo di una pagina web si usano i tag **<BR>**, **<P>** o **<DIV>**.

Il tag **<BR>** equivale ad un'interruzione di riga che non spezza il paragrafo. Infatti, il testo che segue va a capo mantenendo tutte le caratteristiche del testo precedente e senza che si crei spazio vuoto tra le righe. Il tag **<BR>** non richiede un corrispondente tag di chiusura, basta inserirlo nel punto in cui si vuole che la frase vada a capo.

Il tag **<P>** crea invece un'interruzione di paragrafo; il testo successivo va a capo, lasciando una riga di spazio vuoto, e può essere formattato in maniera diversa rispetto al testo precedente. Il tag **<P>** può essere usato singolo oppure (scelta consigliata) con la rispettiva chiusura **</P>**.

Il tag **<DIV>** che si usa con la chiusura **</DIV>** serve a creare un blocco di testo che va a capo e può assumere caratteristiche di formattazione diverse dal testo precedente, senza tuttavia creare spazio vuoto tra il nuovo testo e quello precedente.

Per impostare gli allineamenti di testo che valgono paragrafo per paragrafo si usa l'attributo **ALIGN** del tag **<P>** secondo la sintassi:

```
<P ALIGN="valore">paragrafo da allineare</P>
```

dove il valore di **ALIGN** può essere:

- **left** per l'allineamento a sinistra,
- **center** per l'allineamento al centro,
- **right** per quello a destra,
- **justify** per l'allineamento giustificato.

L'allineamento a sinistra è quello predefinito, quindi specificare l'attributo `ALIGN="left"` è superfluo, mentre il valore `justify` è stato introdotto a partire da Html 4.0 e quindi non tutti i browser sono in grado di interpretarlo correttamente.

Allo stesso modo di `<P ALIGN="valore">...</P>` si può usare il tag `<DIV ALIGN="valore">...</DIV>` per identificare e formattare in modo omogeneo interi blocchi di testo anziché i singoli paragrafi. Quindi usando il tag `<DIV>...</DIV>` si contrassegna una sezione di testo, costituita da uno o più paragrafi.

Esiste anche un tag che permette di definire un paragrafo di testo preformattato. Si tratta del tag `<PRE>...</PRE>` che fa in modo che il testo venga visualizzato dal browser nel modo in cui è stato digitato nel codice HTML usando un carattere a spaziatura fissa.

Una maniera per creare blocchi di testo indipendenti è sicuramente quella di utilizzare il tag `<HR>` (Horizontal Rules).

Esso inserisce una linea orizzontale divisoria e può essere associato ai seguenti attributi di formattazione:

WIDTH	specifica l'ampiezza della riga, in pixel o in percentuale rispetto alla pagina
SIZE	specifica lo spessore della riga in pixel
ALIGN	specifica l'allineamento della riga nella pagina (left, right, center)
COLOR	specifica il colore della riga
NOSHADE	Elimina l'ombreggiatura di default associata alla riga

## Titoli

Mediante l'inserimento di opportuni titoli il testo di una pagina HTML può essere suddivisa in capitoli e sottocapitoli. Per i titoli HTML sono previsti sei livelli gerarchici.

Il tag per definire un titolo è `<Hn>... </Hn>` dove n è un valore che va da 1 a 6. Ciò significa che si possono avere sei livelli di titolazione, da `<H1>` (livello superiore) a `<H6>` (livello inferiore).

In visualizzazione i titoli appaiono come paragrafi a sé stanti con caratteristiche di formattazione legate al livello di titolazione. L'aspetto dei titoli può essere personalizzato per quanto riguarda l'allineamento e la formattazione.

È importante inserire dei titoli nei testi delle pagine web perché questo incrementa notevolmente la leggibilità e perché i motori di ricerca, quando catalogano i siti web, spesso si basano anche sui titoli presenti nella pagina per classificarne gli argomenti.

## Elenchi

Il primo tipo di elenco che vedremo è quello puntato; le voci di questo elenco non hanno un ordine di successione ed ogni voce viene evidenziata da un simbolo. Il tag è `<UL>` (Unordered List) e funziona con il rispettivo `</UL>`. Nel mezzo del tag `<UL>... </UL>` ciascuna voce è identificata dal tag `<LI>`. Esempio:

```
<UL>
  <LI> voce a
  <LI> voce b
```

```
<LI> voce c
</UL>
```

Al tag <UL> si può aggiungere l'attributo **TYPE**="tipo di simbolo" che permette di scegliere un simbolo personalizzato. I valori previsti sono:

- **disc**: punto pieno
- **circle**: punto vuoto
- **square**: quadrato nero.

Un altro tipo di elenco è quello numerato; le voci di questo elenco hanno un preciso ordine di successione ed ogni voce viene evidenziata da un numero oppure opzionalmente, da una lettera. Il tag è <OL> (Ordered List) e funziona con il rispettivo </OL>. Nel mezzo del tag <OL>... </OL> ciascuna voce è identificata dal tag <LI>. Al tag <OL> si può aggiungere l'attributo **TYPE**="tipo di numerazione" che permette di scegliere numeri o lettere. I valori previsti sono:

- **1**: numeri arabi,
- **I**: numeri romani maiuscoli,
- **i**: numeri romani minuscoli,
- **A**: lettere maiuscole,
- **a**: lettere minuscole.

Si può anche usare l'attributo **START**="nr" per iniziare il conteggio da un qualsiasi valore.

## Le tabelle

Oltre ad avere la funzione di rappresentare dati di ogni genere allineati in righe e colonne, le tabelle in HTML si utilizzano per costruire l'intera struttura di una pagina web.

Con le tabelle, Infatti, si può definire il layout di una pagina web in modo da disporre il testo su colonne multiple e di ottenere tipi di allineamenti complessi.

Per realizzare ad esempio una pagina suddivisa in tre colonne è possibile definire una tabella a tre colonne con i bordi invisibili.

Il tag per creare una tabella è <TABLE> con il rispettivo </TABLE>. Nell'area compresa tra <TABLE> e </TABLE> ogni riga si definisce con la coppia di tag <TR>....</TR> (Table row). All'interno di ogni riga si creano le celle desiderate; ogni cella si definisce con la coppia di tag <TD>.....</TD> (Table data). Ad esempio con la sintassi:

```
<TABLE>
  <TR>
    <TD>cella 1-1</TD>
    <TD>cella 1-2</TD>
  </TR>
  <TR>
    <TD>cella 2-1</TD>
    <TD>cella 2-2</TD>
  </TR>
</TABLE>
```

Viene definita una tabella contenente due righe e due colonne.

Le tabelle si possono nidificare, cioè è possibile mettere delle tabelle all'interno di altre tabelle creando così strutture più o meno complesse che si adattano perfettamente a qualsiasi tipo di impaginazione voluta.

Queste potrebbero addirittura contenere parti di immagini che si ricompongono cella per cella come se si trattasse di un puzzle.

Anzi, per immagini di grosse dimensioni vengono adoperate proprio a questo modo per non creare lunghe attese di caricamento.

La larghezza di una tabella si specifica tramite l'attributo **WIDTH** che può essere espresso in pixel con valore assoluto, o in percentuale rispetto alla larghezza della pagina. Con la sintassi:

`<TABLE WIDTH=600>...</TABLE>`

si crea una tabella con una larghezza di 600 pixel; con la sintassi:

`<TABLE WIDTH="90%">...</TABLE>`

si crea una tabella con una larghezza pari all'90% della pagina. Se non si specifica la larghezza, la tabella si estende orizzontalmente in modo da occupare l'intera pagina.

Con l'attributo **ALIGN**="valore" del tag `<TABLE>` si può definire l'allineamento della tabella rispetto alla pagina. Il valore di **ALIGN** può essere left, center o right.

Per definire il bordo di una tabella si usa l'attributo **BORDER** secondo la sintassi:

`<TABLE BORDER="spessore">.....</TABLE>`

dove lo spessore deve essere espresso da un numero intero di pixel. Se non si imposta alcun valore il bordo delle celle resta invisibile.

Per associare un colore al bordo della tabella si usa l'attributo **BORDERCOLOR**:

`<TABLE BORDER="spessore" BORDERCOLOR="#colore">.....</TABLE>`

Per impostare uno sfondo personalizzato di una tabella si usano gli stessi attributi del tag `<BODY>` cioè **BGCOLOR**="#colore" per definire un colore di sfondo omogeneo e **BACKGROUND**="image" per collocare sullo sfondo un'immagine.

Questi due attributi possono essere inseriti all'interno del tag di apertura `<TABLE>` per assegnare la caratteristica all'intera tabella, o all'interno del tag di apertura `<TD>` per assegnare la caratteristica ad una singola cella. Se non viene impostato nessun valore lo sfondo della tabella rimane neutro.

E' possibile, anche se poco utilizzato (in quanto supportato soltanto da IE), strutturare una tabella in intestazione, corpo e piede attraverso i tag

<code>&lt;THEAD&gt;...&lt;/THEAD&gt;</code>	specifica la sezione di intestazione della tabella ed è simile all'elemento <b>HEAD</b> di html.
<code>&lt;TBODY&gt;...&lt;/TBODY&gt;</code>	specifica la sezione del corpo della tabella, è analogo all'elemento <code>&lt;BODY&gt;</code> di html ed influenza la visualizzazione della tabella stessa
<code>&lt;TFOOT&gt;...&lt;/TFOOT&gt;</code>	specificare la sezione del piede della tabella

Per personalizzare la visualizzazione dei bordi interni (in genere quando si fa uso dei tag `<THEAD>` `<TBODY>` e `<TFOOT>`), si utilizza l'attributo **RULES** che assume i seguenti valori:

- **none** = rimuove tutti i bordi interni
- **groups** = visualizza bordi orizzontali fra le sezioni `<THEAD>``<TBODY>``<TFOOT>`

- **rows** = visualizza bordi orizzontali tra ciascuna riga
- **cols** = visualizza bordi orizzontali tra ciascuna colonna
- **all** = visualizza tutti i bordi interni

Si può regolare a piacere la spaziatura tra le celle e all'interno delle celle. Con l'attributo **CELLSPACING** si definisce la spaziatura tra una cella e quelle adiacenti. La sintassi è:

`<TABLE CELLSPACING="numero di pixel">...</TABLE>`

In mancanza di indicazioni il valore predefinito è 2 pixel.

Con l'attributo **CELLPADDING** si definisce la spaziatura interna della cella, ovvero la quantità di spazio vuoto tra il margine della cella ed i suoi contenuti. La sintassi è:

`<TABLE CELLPADDING="numero di pixel">...</TABLE>`

In mancanza di indicazioni il valore predefinito è 1 pixel.

Nel tag `<TD>` si possono specificare gli attributi **WIDTH** e **HEIGHT** che permettono di definire la larghezza e l'altezza di ogni cella. Quindi con:

`<TD WIDTH=120>...</TD>` oppure `<TD WIDTH="20%">...</TD>`

si fissa la larghezza della cella, e con:

`<TD HEIGHT =15>...</TD>` oppure `<TD HEIGHT = "20%">...</TD>`

si fissa l'altezza della cella.

Tutte le celle di una stessa colonna devono avere la stessa larghezza e tutte le celle di una stessa riga devono avere la stessa altezza.

Se vengono indicati valori differenti il browser si basa automaticamente su quello più elevato. Per non incorrere in errori di visualizzazione, quando si fissano valori specifici di altezza e larghezza delle celle, bisogna essere sicuri che il contenuto di ogni cella non ecceda lo spazio disponibile.

Per creare una tabella con un numero non costante di celle per ogni riga/colonna si usano gli attributi **ROWSPAN** e **COLSPAN** del tag `<TD>`.

`<TD ROWSPAN="numero">...</TD>` estende la cella attraverso il numero di righe specificato.

`<TD COLSPAN="numero">...</TD>` estende la cella attraverso il numero di colonne specificato.

Considerando ad esempio la tabella a due righe e due colonne:

```
<TABLE border=1>
  <TR>
    <TD>cella 1-1</TD>
    <TD>cella 1-2</TD>
  </TR>
  <TR>
    <TD>cella 2-1</TD>
    <TD>cella 2-2</TD>
  </TR>
</TABLE>
```

cella 1-1	cella 1-2
cella 2-1	cella 2-2

utilizzando **ROWSPAN** possiamo in pratica unire la cella 1-1 e la cella 2-1:

```
<TABLE border=1>
  <TR>
    <TD colspan="2">cella 1-1 / 2-1</TD>
    <TD>Cella 1-2</TD>
  </TR>
```

```

<TR>
  <TD ROWSPAN=2>cella 1-1 / 2-1</TD>
  <TD>cella 1-2</TD>
</TR>
<TR>
  <TD>cella 2-2</TD>
</TR>
</TABLE>

```

	Cella 2-2
--	-----------

utilizzando ROWSPAN possiamo in pratica unire la cella 1-1 e la cella 1-2:

```

<TABLE border=1>
  <TR>
    <TD COLSPAN =2>cella 1-1 / 1-2</TD>
  </TR>
  <TR>
    <TD>cella 2-1</TD>
    <TD>cella 2-2</TD>
  </TR>
</TABLE>

```

cella 1-1 / 1-2	
cella 2-1	cella 2-2

Per formattare i contenuti di una cella si utilizzano appositi attributi del tag <TD>. Per l'allineamento orizzontale dei contenuti di una cella si utilizza:

```
<TD ALIGN="valore">.....</TD>
```

dove ALIGN può assumere i valori left (predefinito), center, right o justify.

Per l'allineamento verticale dei contenuti di una cella si utilizza:

```
<TD VALIGN="valore">.....</TD>
```

dove VALIGN può assumere quattro differenti valori:

- **top**: allineamento al margine superiore della cella;
- **middle** (predefinito): allineamento al centro dell'altezza della cella;
- **bottom**: allineamento al margine inferiore della cella;
- **baseline** il contenuto della cella si allinea alla base della prima riga di testo delle celle adiacenti.

Generalmente il testo contenuto in una cella scorre a capo automaticamente (wrap) secondo le dimensioni della cella stessa. È possibile forzare il testo di una cella affinché non vada mai a capo aggiungendo al tag <TD> l'attributo NOWRAP:

```
<TD NOWRAP>.....</TD>
```

Bisogna prestare attenzione al fatto che usando NOWRAP le celle adiacenti variano automaticamente, in base all'allargamento della cella con l'attributo.

## *I collegamenti ipertestuali*

Una pagina senza collegamenti ipertestuali non è una pagina web.

La forza di una pagina web è proprio quella di agganciare ed essere a sua volta agganciata alle altre pagine rendendo tutta la rete un unico documento. In altre parole, cliccare su un link può comportare un

cambio pagina, l'inizio dello scaricamento di un file, l'apertura del vostro programma di posta elettronica ed altro ancora.

I collegamenti ipertestuali si creano associando l'indirizzo (percorso/nome file) del nodo di destinazione ad un elemento visualizzato nella pagina corrente che può essere una stringa di testo o un'immagine.

Per creare un collegamento verso un'altra pagina dello stesso sito la sintassi è:

```
<A HREF="profilo.html"> Profilo Aziendale </A>
```

dove A sta per ancoraggio ed HREF è l'abbreviazione di Hypertext-Reference (riferimento ipertestuale).

Affinché il collegamento funzioni è fondamentale scrivere il nome del file di destinazione correttamente, rispettando maiuscole e minuscole. Nel caso in cui il file di destinazione non si trovi nella stessa cartella del file di posizione bisogna specificare anche il percorso completo:

```
<A HREF="prodotti/prof.html"> Bravo Professionisti </A>
```

dove prodotti è una sottocartella contenente il file prof.html.

Il file a cui si punta può essere indifferentemente una pagina web, un'immagine, un file compresso (.ZIP), un documento di Word, un file PDF, ecc..

Per creare un collegamento verso un'altro sito web è sufficiente inserire nel tag <A HREF> l'indirizzo completo del sito in questione:

```
<A HREF=http://www.seneta.it> Ing. Antonio Guadagno – Sito Personale </A>
```

Oltre i collegamenti ad altri file, è possibile definire i cosiddetti collegamenti interni che permettono al lettore di saltare facilmente da un punto all'altro della pagina visualizzata.

Per creare i collegamenti interni occorre innanzitutto definire le destinazioni (ancore o segnalibri) nei vari punti prescelti mediante il tag <A NAME="segnalibro">:

```
<A NAME="punto1">Primo Capitolo </A>
```

```
<A NAME="punto2"> Secondo Capitolo </A>
```

e poi creare i collegamenti che portano alle due destinazioni secondo la sintassi:

```
<A HREF="#punto1"> Vai Primo Capitolo </A>
```

```
<A HREF="#punto2"> Vai al Secondo Capitolo </A>
```

Si possono anche combinare collegamenti esterni ed interni, cioè definire un collegamento ipertestuale (chiamato collegamento esterno con ponte) che punta ad un'ancora specifica all'interno di un'altra pagina web:

```
<A HREF="prodotti/prof.html#req"> Requisiti per installare Bravo Prof </A>
```

Naturalmente occorre definire l'ancora di destinazione *req* nella pagina *prof.html*.

Per inserire un collegamento ad un indirizzo di posta elettronica, in modo che l'invio del messaggio sia quasi automatizzato, la sintassi è:

```
<A HREF="mailto:info@seneta.it"> Per informazioni </A>
```

Quando si clicca su un collegamento in una pagina web, viene caricata la nuova destinazione (immagine, pagina o sito) nella stessa finestra del browser. Seguendo i vari collegamenti ipertestuali, il lettore potrebbe non ricordarsi più il punto di destinazione.

Per prevenire questo inconveniente si può fare in modo che la destinazione del collegamento venga visualizzata in una nuova finestra del browser che si sovrappone a quella di partenza. Per ottenere questo risultato basta aggiungere al tag <A HREF> di partenza l'attributo **TARGET**:

```
<A HREF="profilo.html" TARGET="_blank"> Profilo Aziendale </A>
```

## Inserimento di immagini

Per inserire un'immagine in una pagina HTML basta inserire il tag: <**IMG SRC**="nome immagine">; questo tag non ha bisogno di chiusura.

Affinché l'immagine venga visualizzata nella pagina web bisogna specificarne il nome, l'estensione e l'eventuale percorso. Usando il tag in questo modo l'immagine appare allineata alla base della riga di testo corrispondente, senza spazi aggiuntivi e mantiene le sue dimensioni assolute.

È possibile definire le dimensioni di visualizzazione di un'immagine specificando gli attributi **WIDTH** (larghezza in pixel) e **HEIGHT** (altezza in pixel). Ad esempio:

```
<IMG SRC="images/logo.jpg" WIDTH="250" HEIGHT="64">
```

Quando si specificano le dimensioni di un'immagine è necessario conoscerne le dimensioni effettive per mantenere inalterato il rapporto larghezza/altezza evitando così di deformare l'immagine. Anche se si vuole visualizzare l'immagine a grandezza naturale è sempre preferibile specificare i valori di **WIDTH** e **HEIGHT** perché così il browser ha maggiori informazioni per collocare l'immagine ed il caricamento della pagina risulta molto più rapido.

Oltre alle dimensioni di visualizzazione si può definire anche l'allineamento dell'immagine rispetto al testo circostante mediante l'attributo **ALIGN**. Esempio:

```
<IMG SRC="images/logo.jpg" WIDTH=250 HEIGHT=64 ALIGN="bottom">
```

I valori di **ALIGN** ammessi sono:

Bottom	il lato inferiore dell'immagine appare allineato alla base della riga di testo (valore predefinito)
Middle	il punto mediano dell'altezza dell'immagine si allinea alla base della riga di testo; il testo si spezza e prosegue sotto l'immagine
Left	l'immagine si posiziona sul lato sinistro della pagina e il testo scorre intorno a lei sul lato destro
Right	l'immagine si posiziona sul lato destro della pagina e il testo scorre intorno a lei sul lato sinistro

Nel caso in cui si voglia allontanare l'immagine dal testo si può aumentare lo spazio vuoto che circonda l'immagine attraverso gli attributi **VSPACE** (spazio verticale, in pixel) e **HSPACE** (spazio orizzontale, in pixel). Ad esempio scrivendo:

```
<IMG SRC="images/logo.jpg" VSPACE=10 HSPACE=20>
```

oltre ad inserire nella pagina web logo.jpg, si specifica che sopra e sotto l'immagine devono rimanere 10 pixel vuoti, e a destra e sinistra devono rimanere 20 pixel vuoti.

Si può anche aggiungere un bordo intorno all'immagine tramite l'attributo **BORDER** che deve essere espresso in pixel secondo la sintassi:

```
<IMG SRC="images/logo.jpg" BORDER=5>
```

Un attributo fondamentale del tag <IMG> è **ALT** (testo alternativo). Grazie a tale attributo si può definire una didascalia associata all'immagine. La sintassi corretta è:

```
<IMG SRC="images/logo.jpg" ALT="Logo descrittivo">
```

Questa didascalia appare durante il caricamento della pagina o quando si passa con il mouse sopra l'immagine. I vantaggi principali dell'usare l'attributo ALT sono:

- i visitatori possono capire prima del caricamento completo il contenuto delle immagini;
- i browser speciali per non-vendenti sono in grado di interpretare correttamente la pagina;
- diversi motori di ricerca tengono conto del valore dell'attributo ALT delle immagini di una pagina per catalogarla con precisione.

Ad ogni immagine si può associare un collegamento ipertestuale includendo il tag <IMG> all'interno della coppia di tag <A HREF> e </A> secondo la sintassi:

```
<A HREF="index.html"><IMG SRC="images/logo.jpg"></A>
```

Associare un collegamento ipertestuale ad un'immagine è molto utile quando si vuole creare ad esempio un menù grafico di navigazione per un sito web. Creando immagini ad hoc come icone o pulsanti il visitatore né intuirà la funzione a colpo d'occhio.

Però, quando un'immagine è cliccabile, appare generalmente circondata da un bordo di colore blu che risulta anti-estetico. In questo caso basta indicare all'interno del tag <IMG> l'attributo **BORDER="0"** per fare in modo che il bordo resti invisibile.

## I caratteri speciali

Alcuni caratteri devono essere scritti, nel file html, mediante una speciale codifica. Per esempio, com'è facile intuire, il segno < viene normalmente interpretato dal browser come inizio di un tag e, pertanto, non visualizzato a schermo. Per visualizzare proprio questo simbolo occorre scriverlo in questo modo: **&#60;**; oppure **&lt;**;

Ci sono poi altri caratteri che devono essere scritti in un modo particolare, affinché tutti i browser siano in grado di visualizzarli: per esempio le lettere accentate, come la voce del verbo essere è che si scrive **&egrave;**;

La seguente tabella contiene una lista delle codifiche dei principali caratteri speciali:

carattere	codice	carattere	codice	carattere	codice	carattere	codice
<	&lt;	Ç	&Ccedil;	Ô	&Ocirc;	æ	&aelig;
>	&gt;	È	&Eacute;	Õ	&Otilde;	é	&eacute;
&	&amp;	È	&Egrave;	Ö	&Ouml;	è	&egrave;
"	&quot;	Ê	&Ecirc;	Ø	&Oslash;	ê	&ecirc;
Á	&Aacute;	Ë	&Euml;	Ó	&Oacute;	ë	&euml;
À	&Agrave;	Í	&Iacute;	Ú	&Uacute;	Ñ	&Ntilde;
Â	&Acirc;	Ì	&Igrave;	Û	&Ugrave;	®	&reg;
Ã	&Atilde;	Î	&Icirc;	Û	&Ucirc;	©	&copy;
Å	&Aring;	Ï	&Iuml;	Ü	&Uuml;	spazio	&nbsp;
Ä	&Auml;	Ð	&ETH;	Ý	&Yacute;		

Æ	&AElig;	Ò	&Ograve;	Þ	&THORN;
---	---------	---	----------	---	---------

## I fogli stile

Nonostante il numero elevato di tag a disposizione, una pagina Web non raggiunge mai un livello estetico paragonabile al risultato ottenibile con un qualsiasi elaboratore di testi.

Un foglio di stile è un insieme di modelli, o stili, che si applicano a varie parti del documento e descrivono il modo in cui viene reso.

I CSS ovvero Cascading Style Sheets (fogli di stile sovrapposti) sono ormai divenuti molto comuni sul web, servono per migliorare l'aspetto estetico e al tempo stesso facilitare la creazione di interi siti; combinati con un linguaggio di scripting, quale ad esempio JavaScript, danno vita all' HTML Dinamico conosciuto anche come DHTML, consentendo di superare quelli che erano considerati i limiti di html standard.

E' persino possibile creare delle vere e proprie animazioni sfruttando l'elevata versatilità offerta dal posizionamento degli oggetti sullo schermo.

La specifica CSS è la forma di fogli di stile consigliata dal World Wide Web Consortium come forma appropriata per l'uso con l'Html ed è quindi anche quella che è stata adottata dai due principali produttori di browser, Microsoft e Netscape.

Purtroppo però, anche i CSS non sono supportati allo stesso modo da tutti i browser; quindi, se si decide di usare i fogli di stile, il consiglio sempre valido è quello di collaudare il proprio lavoro con il maggior numero possibile di browser e loro versioni per far sì che le pagine web progettate vengano visualizzate correttamente nella maggior parte dei casi.

I fogli possono essere utilizzati secondo tre metodologie:

- in linea,
- incorporati o con pagina singola,
- esterni o collegati.

Qualunque sia il metodo utilizzato, i valori associati sono preceduti da ":" (due punti) e non da "=" (uguale) come in HTML; inoltre attributi consecutivi sono separati da ";" (punto e virgola) invece che dallo spazio.

### I fogli stile in linea

I fogli **stile in linea** sono creati aggiungendo un attributo di stile ad un tag esistente nel documento; è possibile ad esempio applicare ad un paragrafo un rientro di un centimetro e una dimensione del carattere pari a 30 pt.

Il codice HTML è il seguente:

```
<HTML>
  <HEAD>
    <TITLE>Stile in linea</TITLE>
    <META NAME="Data" CONTENT="05/09/98">
  </HEAD>
  <BODY TEXT="#000000" BGCOLOR="#ffffff">
    <P style="text-indent:1cm; font:30pt Arial">
      <u><font color = "gray">Applichiamo uno stile in linea </font></u></P>
    </body>
```

```
</HTML>
```

Si possono combinare fra di loro più stili, la sintassi corretta prevede di racchiudere il tutto fra doppi apici separando i vari attributi di stile con il punto e virgola.

Tutto ciò che sarà scritto tra l'elemento di apertura `<p style= ..... >` e il suo elemento di chiusura `</p>` avrà come stile una carattere "Arial" con dimensione pari a 30pt ed un rientro di prima riga di 1 cm.

Con la tecnica in linea è teoricamente possibile assegnare stili diversi ogni volta che venga inserito lo stesso tag, ad esempio è possibile creare tanti stili quanti sono i tag `<p>` utilizzati in una pagina.

### I fogli stile incorporati

I **fogli stile incorporati** sono utilizzati per formattare un intero documento HTML; ciò avviene aggiungendo un tag `<STYLE>` (con il suo gemello di chiusura) tra la testata e il corpo.

Definendo una tantum lo stile, ad inizio pagina, tutti gli elementi interessati seguiranno quello stesso stile.

Per definire le proprietà di un tag, è sufficiente scrivere tra parentesi graffe gli attributi da applicare con i relativi valori; tali proprietà saranno valide in tutto il documento ogni volta che sarà applicato quel tag.

Nel prossimo esempio sono applicati degli attributi di formattazione ai tag `<BODY>` e `<H3>`:

```
<HTML>
  <HEAD>
    <TITLE>Stile incorporato</TITLE>
    <META NAME="Data" CONTENT="05/09/98">
    <STYLE>
      BODY {margin-left:2cm; margin-right:1cm}
      H3 {font: 24pt; color: "aqua"; background: "gray"}
    </STYLE>
  </HEAD>
  <BODY>
    <h3>Applichiamo uno stile incorporato</h3>
  </body>
</HTML>
```

### I fogli stile collegati

Se gli stili da definire sono molti, e molte sono le pagine web che ne faranno uso, la cosa migliore è quella di creare una struttura alla quale tutte le pagine possano fare riferimento.

Tale struttura è foglio di stile, ovvero un documento di testo con estensione.css che al suo interno contiene le dichiarazioni di stile, adoperando le stesse regole sintattiche viste per i fogli di stile incorporati.

È chiaro che, modificando un valore all'interno del file .css, tutte le pagine web ad esso associate, si adegueranno automaticamente

Per associare una pagina web ad un foglio di stile .css, occorre aggiungere nell'intestazione la riga:

```
<LINK Rel=stylesheet href="NomeFile.css">
```

Consideriamo il seguente esempio in cui è mostrato sia il contenuto del documento HTML che il contenuto del documento CSS:

```

<HTML>
  <HEAD>
    <TITLE>Stile collegato</TITLE>
    <LINK rel = stylesheet href = "stile.css">
  </HEAD>
  <BODY>
    <P>Applichiamo uno stile collegato</P>
    <P>Il file contenente lo stile applicato è "stile.css"</P>
  </body>
</HTML>

```

```

<STYLE>
  BODY {margin-left: 2cm; margin-right: 1cm}
  P {font-family:Arial; font-size:20pt; font-weight:bold; text-decoration:underline; text-align:center; line-height:60pt}
</STYLE>

```

Gli attributi che possono essere impiegati nei fogli stile sono in genere classificabili in tre distinte categorie:

- attributi di testo
- attributi di margini, rientri ed allineamento
- attributi di colori e sfondo.

Le successive tabelle elencano i principali attributi suddivisi per categorie.

<b>ATTRIBUTI DI TESTO</b>	
<b>font-family</b>	Determina il tipo di carattere: Arial, Times New Roman, Courier New, etc.
<b>font-size</b>	Determina la dimensione del carattere secondo le seguenti unità di misura: punti (pt), pollici (in), centimetri (cm), pixel (px)
<b>font-style</b>	Determina lo stile del testo: italic o normal
<b>font-weight</b>	Determina lo spessore (peso) del carattere: extra-light, demi-light, light, medium, extra-bold, demi- bold, bold
<b>text-decoration</b>	Determina eventuali decorazioni del testo: none, underline, italic, line-through
<b>line-height</b>	Determina l'interlinea secondo le seguenti unità di misura: punti (pt), pollici (in), centimetri (cm), pixel (px)

<b>ATTRIBUTI DI MARGINI, RIENTRI ED ALLINEAMENTO</b>	
<b>margin-left</b>	Imposta il margine sinistro secondo le seguenti unità di misura: punti (pt), pollici (in), centimetri (cm), pixel (px)
<b>margin-right</b>	Imposta il margine destro secondo le seguenti unità di misura: punti (pt), pollici (in), centimetri (cm), pixel (px)

<b>margin-top</b>	Imposta il margine superiore secondo le seguenti unità di misura: punti (pt), pollici (in), centimetri (cm), pixel (px)
<b>margin-bottom</b>	Imposta il margine inferiore secondo le seguenti unità di misura: punti (pt), pollici (in), centimetri (cm), pixel (px)
<b>text-indent</b>	Imposta il rientro di un testo secondo le seguenti unità di misura: punti (pt), pollici (in), centimetri (cm), pixel (px)
<b>text-align</b>	Imposta il tipo di giustificazione al testo: left, center, right, justify

### ATTRIBUTI DI COLORI E SFONDO

<b>background</b>	Imposta il colore di sfondo ad un'area
<b>Color</b>	Imposta il colore di un testo
<b>background-image</b>	Imposta un'immagine di sfondo secondo la sintassi: url( <i>immagine</i> )

Può accadere a volte di dover definire formattazioni diverse a più paragrafi o a più titoli; per ottenere tale flessibilità è necessario assegnare delle classi ai singoli tag aggiungendovi un'estensione costituita da un nome significativo.

Ad esempio, volendo assegnare due stili al tag <p> si ha il seguente sorgente HTML:

```
<html>
  <head>
    <title>
      Assegnamento di classi ai fogli stile
    </title>
  </head>
  <style>
    p.sinistra {font-family:Arial; font-size:12pt; font-weight:bold; text-align:left}
    p.destra {font-family:Garamond; font-size:14pt; text-align:right}
  </style>
  <body>
    <p class=sinistra> ... </p>
    <p class=destra> ... </p>
  </body>
</html>
```