

HTML

Aprirete il vostro editor preferito, qualunque esso sia (Blocco notes), anche uno di quelli visuali se ha la possibilità di andare in modalità testo o html (Word).

I **TAGS**, o elementi, che compongono HTML devono essere racchiusi fra parentesi angolari <queste>

Si tratta di parole chiave, termini in lingua inglese. L'elemento (tag) smette di produrre il suo effetto nel momento in cui lo stesso viene ripetuto facendolo precedere da una barra laterale / contenuta sempre all'interno delle parentesi angolari.

L'inizio di un documento HTML avrà come elemento (tag) iniziale: <HTML> e come elemento (tag) finale </html> poco importa se siano scritti in maiuscolo o minuscolo, tutto ciò che sarà contenuto all'interno di questi due elementi (tags) sarà interpretato dal browser come codice HTML.

Difficile ?

Direi proprio di no :-)

Un documento HTML è composto da due sezioni ben distinte: una testa <HEAD> ed un corpo <BODY>

Nella testa (head) vanno messi quei comandi che inviano informazioni al server o quel codice che deve essere letto con una certa priorità: script, fogli di style, ecc. ecc.

Nel corpo (body) tutto il resto. Anche questi elementi, necessitano del relativo elemento di chiusura: </head> e </body> Riassumendo, una struttura HTML tipo potrebbe essere questa:

```
<html>
<head>
codice per il server, scritps, fogli di style.
</head>
<body>
codice da visualizzare come documento testo ed immagini
</body>
</html>
```

Per il momento all'interno di <Head> non metteremo nulla, si vedrà più avanti cosa inserire, adopereremo quindi soltanto la zona <body>.

Col vostro editor create la struttura sopra e portatevi col cursore all'interno degli elementi (tags) <body> ... </body> Inseriremo del testo, per cui lo faremo precedere dall'elemento <p> che significa: paragrafo, andremo a scrivere un qualsiasi testo: "questa è la mia prima pagina web" per esempio, facciamo in modo di chiudere anche l'elemento <p> con il relativo </p>. Se tutto è stato fatto correttamente dovremo avere una struttura di questo genere:

```
<html>
<head>
</head>
<body>
<p> Questa è la mia prima pagina web </p>
</body>
</html>
```

Non importa come avete disposto le frasi del testo, potete lasciare spazi o ritornare a capo quante volte volete, poco importa; il browser ignorerà il vostro modo di scrivere limitandosi a leggere e interpretare soltanto gli elementi validi di html. (più avanti si vedrà come lasciare spazi e come andare a capo)

A questo punto non resta che salvare il tutto.

Se non avete un editor che salva direttamente in html con molta probabilità il vostro file avrà

un'estensione di tipo .txt, ovvio direte voi, visto che è stato scritto con un normale editor per testi. Rinominate questo file in modo che abbia come estensione htm o html altrimenti il browser non riuscirà ad interpretarlo in modo corretto.

A proposito di questa operazione, dovrei fare alcune precisazioni: l'operazione di rinomina di per se è molto semplice ma chi adopera Windows potrebbe non vedere l'estensione finale del file appena salvato, in questo caso rinominarlo servirebbe a poco dal momento che windows aggiungerebbe html al nome appena assegnato, lasciando però di fatto il file in formato txt

Mi rendo conto che non sia un'operazione molto semplice, specialmente per un principiante in senso generale, per chi avesse problemi in questo tipo di operazione vi rimando a supporti futuri.

Consapevole che se avessimo adoperato un editor visuale quale: "composer" per chi adopera Netscape o "Front Page Express" per chi adopera Internet explorer, non ci sarebbe stato bisogno di niente altro che non fosse stato scrivere: [questa è la mia prima pagina web](#) e salvare...

Preferivate una cosa fatta in questa maniera ?

Se si, avete sbagliato rubrica, questa è infatti destinata a coloro i quali vogliono sapere come fare le pagine web e non come farle fare a vari editors visuali :-))

IL TESTO

Digitate uno spazio ed inserite l'attributo `align="xxx"` dove xxx è uno dei seguenti validi attributi: center, left, right, justify. Questi servono rispettivamente per allineare il testo al centro, al margine sinistro, al margine destro, giustificato per entrambi i margini.

Se avete eseguito correttamente le istruzioni, dovrete avere una cosa di questo tipo:

```
<p align="left">Questa è la mia prima pagina web</p>
```

Vediamo i vari risultati a video:

```
<p align="left">Questa è la mia prima pagina web allineata a sinistra</p>
```

Questa è la mia prima pagina web allineata a **sinistra**

```
<p align="right">Questa è la mia prima pagina web allineata a destra</p>
```

Questa è la mia prima pagina web allineata a **destra**

```
<p align="center">Questa è la mia prima pagina web allineata centrale</p>
```

Questa è la mia prima pagina web allineata **centrale**

```
<p align="justify">Questa è la mia prima pagina web giustificata</p>
```

Questa è la mia prima pagina web **giustificata**, la sola frase non contiene una quantità di testo sufficiente per poterne vedere l'effetto giustificato, così scrivo qualcosa in modo da aggiungerne e rendere meglio l'idea di come avviene la giustificazione, adoperata spesso per

testi lunghi e ben formattati. Ecco fatto, dovrebbe essere sufficiente, guardate i due margini, sono entrambi allineati.

Digitando del testo vi accorgete presto che a differenza degli editor normali, ogni volta che volete tornare a capo non basta la semplice pressione del tasto invio o enter. Nelle pagine web per tornare a capo si fa uso dell'elemento `
` oppure aprendo e chiudendo un nuovo paragrafo `<p></p>` Vi accorgete anche che inserendo più spazi fra una parola e l'altra questi non saranno visualizzati, ignorati tutti quanti lasceranno il posto ad uno solo.

LE IMMAGINI

Apriete il vostro editor, qualunque esso sia, e richiamate la pagina che avevate creato, quella della lezione precedente per capirci. Posizionate il cursore nel punto in cui dovrà essere inserita l'immagine, questo il TAG o elemento, da adoperare:

```

```

Dove xxxxxxxxxxx è il nome della vostra immagine e yyy la sua estensione. Le immagini consentite dal linguaggio html sono .gif .jpg e .png

Si adoperano quelle di tipo gif quando servono immagini animate o che abbiano delle trasparenze (quelle che lasciano vedere lo sfondo), mentre per tutti gli altri casi è preferibile adoperare quelle di tipo jpg essendo questo un formato compresso, risulta molto più veloce da caricare; un aspetto quest'ultimo da non trascurare mai quando si adoperano immagini. Ricordatevi che dimensioni esagerate significa anche peso esagerato; sulla rete un'immagine da 100 Kb spesso induce il visitatore ad abbandonare il sito prima ancora che questa sia stata completamente visualizzata, fate dunque in modo di evitarlo nel modo più assoluto, ed anzi, al contrario, preoccupatevi che queste siano le più "leggere possibili". Il formato .png nato ultimamente più per problemi legali che non per reali esigenze tecniche, è adoperato molto raramente in quanto non risulta poi così "ottimale", ma questo è un mio modesto parere per altro molto personale.:-)

Detto questo, passiamo ad inserire la nostra immagine. Supponiamo che questa si chiami weblink.gif, sarà inserita nel listato in questa maniera:

```
<html>
<head>
</head>
<body>
<p>Questa è la mia prima pagina web </p>



</body>
</html>
```

Notate la sintassi del comando img che è leggermente diversa da quella adoperata per inserire il testo con <p>, infatti il tag per le immagini non ha il relativo tag di chiusura, quindi non serve mettere . Noterete anche che dopo la parola chiave: img, che indica appunto "immagine", c'è un: "src" che significa: source, sorgente, cioè quale immagine deve essere visualizzata. Questo nome va inserito all'interno dei doppi apici, ed è praticamente il nome dell'immagine stessa. Tutto chiaro?

Perfetto, se adesso salvate il vostro file sovrascrivendo quello che avevate richiamato, e lo andate poi a visualizzare, noterete che l'immagine non c'è; o meglio c'è qualcosa di nuovo ma non si vede. Come mai? Semplice, vi serve un'immagine che si chiami "web-link.gif" e che si

trovi nella stessa cartella dove si trova il file html, cioè questo che avete appena modificato e salvato.

A questo punto vi consiglio di lasciare perdere l'immagine web-link.gif ; procuratevi voi una qualsiasi immagine, quella che preferite a patto che sia in uno dei due formati [gif](#) o [jpg](#), mettetela nella stessa cartella dove avete il file html e cambiate il nome all'interno di SRC=" " inserendo il nome della vostra immagine, fatto ? Adesso portatevi una riga sotto il tag servito per inserire l'immagine e scrivete: La mia prima pagina web ha finalmente anche la sua prima immagine.

Se avete fatto tutto quanto correttamente, la vostra pagina comincia a prendere consistenza, con la stessa procedura potete inserire tutti i testi e le immagini che vorrete, ricordandovi sempre di non superare mai certe dimensioni. Tante piccole immagini vengono sommate e potrebbero creare lo stesso problema di una sola immagine da molti kb. Per evitare questo conviene suddividere in più pagine i documenti lunghi o ricchi di fotografie.

Prima di proseguire vediamo cosa potremo aggiungere al tag: ``

Intanto se la nostra fotografia non si trovasse nella stessa cartella dove c'è il file che la richiama, dovremo necessariamente farla precedere dal "path" cioè il percorso che specifica appunto la sua posizione, questo potrebbe essere anche un URL valido, cioè un indirizzo di un sito Internet che la contiene, l'immagine in questo caso sarà visibile solo nel caso in cui siate collegati alla rete.

Questi alcuni probabili validi percorsi:

```



```

Se specifichiamo le dimensioni dell'immagine, aiuteremo il browser a visualizzare tutta la pagina con molta più rapidità, per cui sarebbe opportuno inserire sempre queste informazioni. Per farlo si adoperano i due termini in lingua inglese per larghezza e altezza:

```

```

I numeri 64 e 128 sono pixel, l'unità di misura dell'immagine stessa, rispettivamente: width per la larghezza, mentre height per l'altezza. E' anche possibile specificare se l'immagine deve avere un bordo e di quale spessore; per ottenere questo si adopera l'attributo: [border](#) dove è ammesso anche il valore zero nel caso in cui non si volesse alcun bordo.

```

```

Avrete certamente notato che tutti i vari attributi restano all'interno delle parentesi angolari, `<queste>` e potrebbero essere messi anche in ordine diverso da quello che ho adoperato io purché ne sia rispettata la corretta sintassi, e per sintassi intendo il segno di = uguale e le virgolette ", anzi i doppi apici". Ci sono poi molti altri attributi che potrebbero essere inseriti allo stesso modo, vi elenco quelli che a mio avviso si adoperano di più, rimandandovi alla guida HTML nel caso si volesse conoscere l'elenco completo

[alt](#) = nome dell'immagine. Questo nome viene visualizzato passando sull'immagine stessa con il puntatore del mouse.

[align](#) = Allineamento dell'immagine: destro, sinistro, nel mezzo, sopra, sotto. Tutti questi termini vanno scritti con l'equivalente termine in lingua inglese: right, left, middle, top, bottom.

[hspace](#) = Spazio laterale destro e sinistro dall'eventuale testo, da un'altra immagine o dal bordo di una tabella se inserita al suo interno. Questo valore è espresso in pixel.

`vspace` = Spazio laterale sopra sotto dall'eventuale testo, da un'altra immagine o dal bordo si una tabella se inserita al suo interno. Questo valore è espresso in pixel.

Gli attributi inseribili all'interno del tag IMG sono molti di più, ma visto il loro utilizzo e considerato che queste lezioni sono destinate ai principianti, credo che quelli citati siano più che sufficienti per coprire la maggior parte delle necessità.

Questo un esempio di un richiamo ad un'immagine adoperando tutti gli attributi finora descritti.

```

```

Da notare che fra un attributo e l'altro si lascia uno spazio vuoto.

Nota: il nome dell'immagine deve essere scritto esattamente come è, rispettando cioè eventuali maiuscole/minuscole/spazi, questo perché a differenza di windows che non fa distinzione alcuna, una volta trasferito il lavoro sul server l'immagine di nome Web-link.gif è considerata un'immagine diversa dalla stessa immagine che si chiamasse invece WEB-link.gif (differenza dovuta alle maiuscole)

I LINK O ANCORE

Dopo aver visto come inserire [testo](#) e [immagini](#), non resta che affrontare l'argomento che da vita alla grande rete: i links o ancore, cioè quelle scritte interattive che se cliccate col mouse danno modo di saltare da una pagina all'altra o come spesso succede, da un intero sito all'altro. Sono proprio queste Ancore, o links ipertestuali, che hanno decretato il successo del web.

Apriete il vostro editor, qualunque esso sia, e richiamate la pagina che avevate creato, quella della lezione precedente per capirci. Posizionate il cursore nel punto esatto in cui dovrà essere inserita un'ancora, questo il [TAG](#) da adoperare:

```
<A> </A>
```

A come ancora... così da solo è inutilizzabile, ma vediamo combinato con l'attributo: [HREF](#) per esempio.

[HREF](#) significa hypertext reference, indica che ciò che si trova fra le ancore di apertura e chiusura diventerà un collegamento ipertestuale, ovvero sarà richiamata un'altra pagina o un punto preciso della stessa pagina, quello specificato nell'attributo stesso.

Vediamo un esempio pratico, supponiamo che nella pagina web ci sia un link al mio sito: [web-link.it](#) (uno sito a caso tanto per cambiare :-))
Sarà sufficiente specificarlo in questo modo:

```
<html>  
<head>  
</head>  
<body>
```

```
<A HREF="http://web-link.it">Web-Link</A>
```

```
</body>  
</html>
```

Notate la sintassi del comando, la parola web-link è l'unica cosa che si vedrà sulla pagina, e si trova fra i due tags Ancora di cui il primo <A> ha l'attributo Href che indica quale URL valido richiamare nel momento in cui viene fatto clic sulla scritta, l'ultimo è la chiusura, da non dimenticare mai!

Potrei sostituire la scritta Web-Link con una qualsiasi altra frase, oppure adoperare delle immagini, il risultato sarebbe comunque lo stesso, tutto ciò che si trova fra i due tags ancora, se cliccato, porterebbe all'apertura del sito web-link.it perché è quello l'indirizzo specificato dopo HREF

Se io introducessi una immagine al posto della scritta, darei vita ad un processo chiamato nidificazione degli elementi (tags), cioè si possono inserire elementi html all'interno di altri elementi html, questo perché ogni elemento ha i suoi attributi e trovandosi all'interno di un altro elemento eredita eventuali attributi dell'elemento che lo incorpora.

```
<html>
<head>
</head>
<body>

<A HREF="http://web-link.it"></A>

</body>
</html>
```

Fra le virgolette dell'attributo HREF oltre agli indirizzi URL validi come quello dell'esempio , è possibile specificare anche :

```
<A HREF="ftp://...">
<A HREF="mailto:...">
<A HREF="news:...">
<A HREF="nomefile.zip">
<A HREF="nomepagina.html">
```

Che servono rispettivamente per:

```
<A HREF="ftp://...">
Collegamento ad un sito FTP.
```

```
<A HREF="mailto: nome@server">
Indirizzo di posta elettronica per la spedizione di una e-mail all'indirizzo specificato, facendo uso del proprio programma di posta elettronica.
```

```
<A HREF="news:...">
Collegamento ad un gruppo di discussione (newsgroup Usenet).
```

```
<A HREF="nomefile.zip">
Scaricamento (download) di un file in formato compresso.
```

```
<A HREF="nomepagina.html">
Collegamento ad altra pagina all'interno della stessa cartella.
```

Ci sarebbero anche altre possibili forme di links quali: gopher, nntp, telnet, wais, newsrc; ma visto il loro raro utilizzo, ritengo che possano servire soltanto per creare confusione ad un principiante.

Attenzione, la pagina richiamata dal link andrà a sostituire la pagina richiamante, se non è questo l'effetto desiderato, è possibile aggiungere un attributo che specifichi dove aprire

questa pagina, o meglio su quale finestra del browser di navigazione dovrà essere aperta. Vediamo come:

```
target="nome della finestra"
```

Se avete un sito strutturato con frames, cioè diversi riquadri finestre all'interno di una finestra principale, potete specificare il nome della finestra nella quale vedere il risultato del link, oppure fare in modo che se ne possa aprire una nuova adoperando la parola riservata: `_blank`. Vediamo un esempio completo:

```
<html>
<head>
</head>
<body>
<A HREF="http://web-link.it" target="_blank">Web-Link</A>
</body>
</html>
```

Una nuova finestra sarà aperta in modo da mantenere anche la vostra, quella dalla quale è stato possibile fare il clic sul link, per cui ci saranno due browser aperti uno con la vostra pagina e l'altro con la pagina associata al link.

A questo punto conoscete tutto ciò che serve per creare un sito, si ho detto sito e non più pagina web, infatti con l'inserimento del tag ANCORA è possibile passare da una pagina all'altra dando vita ad un vero e proprio sito: il vostro.

Sono ancora molte le cose da sapere per fare buoni siti, ma già con questa terza lezione potete cominciare a fare qualcosa, dalla prossima torneremo sulla struttura, sul testo e sulle immagini aggiungendo alcuni particolari e precisazioni che vi serviranno soltanto se nel frattempo avrete cominciato a fare qualcosa di vostro.

Caratteri speciali, fonts e cosmesi del testo

Dovreste essere ora in grado di scrivere la vostra pagina web completa di testo, immagini e links vari. Forse vi sarete già accorti che scrivendo in HTML non sempre è possibile ottenere lo stesso risultato che si ottiene con la normale scrittura di un testo. Certi caratteri infatti hanno regole ben precise, si tratta semplicemente di conoscerle ed applicarle.

Una delle prime cose che si nota è la difficoltà di poter lasciare più di uno spazio vuoto fra una parola e l'altra. Html infatti ignora tutti gli spazi inseriti con il metodo normale e ne lascia di fatto uno soltanto, provate dunque a scrivere una frase mettendo più spazi vuoti fra le varie parole, provate poi a guardarne il risultato sul vostro browser. Noterete con sorpresa che tutti gli spazi sono stati ignorati e sostituiti da uno soltanto, questa è una caratteristica di HTML, come fare allora per averne più di uno ?

Con appositi codici che servono esclusivamente ad introdurre spazi:

` `; Questo simbolo significa non-breaking space, (da notare che non va inserito fra parentesi angolari non essendo un elemento o tag), se ne possono mettere tanti quanti sono gli spazi desiderati, così facendo il browser eseguirà la richiesta.

Anche tornare a capo diventa difficile all'interno di uno stesso paragrafo; come già detto nella prima lezione: la semplice pressione del tasto invio (o return a seconda del tipo di tastiera), non è sufficiente per forzare un ritorno a capo, per fare questa operazione si adopera il tag `
` maiuscolo o minuscolo che sia non importa, questo tag forza un ritorno a capo, più `
` consecutivi equivalgono a più ritorni a capo quindi saltare tante righe quanti sono i BR inseriti.

I due metodi sopra sono di gran lunga i più adoperati durante la scrittura di un testo html, da qui la logica considerazione: per ottenere un semplice spazio vuoto c'è bisogno di un'operazione macchinosa come digitare quella sequenza di 5 caratteri? Se adoperate un editor di testi sì, se invece vi siete procurati un editor html, probabilmente si tratta di fare click su un pulsantino che introdurrà quella sequenza di caratteri, quasi come premere il tasto spazio sulla tastiera.

Ci sono anche altri caratteri che conviene introdurre con le corrette regole che HTML impone, si tratta di caratteri particolari ma che spesso si adoperano nella comune scrittura; per esempio noi Italiani abbiamo le vocali accentate, se scritte come da tastiera queste potrebbero mettere in crisi il browser che le visualizza, è perciò preferibile adoperare il relativo codice per ottenerle :

à = à
è = è
ecc.....

Anche le doppie virgolette, le parentesi angolari e tutto quello che potrebbe essere male interpretato conviene scriverlo con il suo codice; ecco qui di seguito l'elenco dei caratteri di uso più comune:

" = "
& = &
< = <
> = >
€ = €

Per concludere, alcuni caratteri speciali che potrebbero risultare utili:

© = ©
® = ®
£ = £
¼ = ¼
½ = ½
¾ = ¾
¹ = ¹
º = º

Per rendere una pagina più leggibile si fa spesso ricorso ad una specie di cosmesi del testo, si usa infatti sottolinearlo, enfatizzarlo, inclinarlo, ridimensionarlo, ecc. ecc. Vediamo quali sono gli elementi (tags) che producono questi effetti.

** = bold o grassetto.**

Tutto quello che sarà scritto all'interno dei tag **** e **** sarà visualizzato in grassetto, facciamo un esempio: dovendo scrivere

" www.web-link.it tutto per la tua pagina web"

e volendo la scritta web-link.it in grassetto, si dovrà racchiudere questa fra i due tag **** e ****, in questo modo:

www.web-link.it tutto per la tua pagina web.

Questo il risultato:

www.**web-link.it** tutto per la tua pagina web

Avremo potuto ottenere lo stesso risultato adoperando **** e ****

<U> </U> = Underline, sottolineato.

Tutto quello che sarà scritto all'interno dei tag **<U>** e **</U>** sarà visualizzato in sottolineato, facciamo lo stesso esempio: dovendo scrivere

" www.web-link.it tutto per la tua pagina web"

e volendo la scritta web-link.it in sottolineato, si dovrà racchiudere questa fra i due tag **<U>** e **</U>**, in questo modo:

www.<U>web-link.it</U> tutto per la tua pagina web.

Questo il risultato:
www.[web-link.it](http://www.web-link.it) tutto per la tua pagina web

`<I> </I>` = **Italico, inclinato.**

Tutto quello che sarà scritto all'interno dei tag `<I>` e `</I>` sarà visualizzato in italico, facciamo ancora lo stesso esempio: dovendo scrivere

" www.web-link.it tutto per la tua pagina web"

e volendo la scritta web-link.it in italico, si dovrà racchiudere questa fra i due tag `<I>` e `</I>`, in questo modo:

www.`<I>`web-link.it`</I>` tutto per la tua pagina web.

Questo il risultato:

www.*web-link.it* tutto per la tua pagina web

Avremo potuto ottenere lo stesso risultato adoperando `` e ``

Viene da se che si possono combinare i vari tag fra di loro nidificandoli e di conseguenza sommandone i vari effetti: sottolineato più grassetto più inclinato: `<u><i>` web-link.it`</i></u>` così da ottenere:

www.***web-link.it*** tutto per la tua pagina web

E' anche possibile cambiare il colore, le dimensioni, il tipo di carattere adoperato per la scrittura. I metodi sono due: Adoperare il tag `` con i relativi parametri o fare uso di un foglio di `style`, per quest'ultimo aspetterei di aver acquisito una certa familiarità con il linguaggio html.

Per il tag font invece l'operazione è molto semplice, si tratta di racchiudere il testo nei tags `` `` con la consueta procedura adoperata per tutti gli altri elementi (tags). Fra i parametri adoperabili :

color, face, size. Rispettivamente per colore, tipo di carattere e dimensioni dello stesso.

Vediamo come adoperare questi parametri:

`color=" "` Fra le doppie virgolette sono ammessi i nomi dei colori in lingua inglese, oppure il numero del colore in formato esadecimale (un numero formato da tre coppie esadecimali precedute dal simbolo # cancelletto). Su [questo link](#) alcuni esempi.

`` o anche

`` produrranno lo stesso effetto e cioè questo: **testo scritto in verde**

`` cambia il tipo di carattere adoperando quello specificato:

``

Sto scrivendo con il font **Comic Sans MS**

`` cambia le dimensioni del carattere, dove numero può avere un valore compreso fra -6 e +7 :

``

Sto scrivendo con un font dimensione +3

``

Sto scrivendo con un font dimensione -2

Anche in questo caso si possono avere più attributi e combinarli fra loro:

``

Questo il risultato: **Sto scrivendo con un font dimensione 3 di colore verde e carattere Comic Sans MS**

ESERCIZIO

Utilizzare il Blocco note per scrivere il seguente programma:

```
<html>
<head>
</head>
<body>
<B>
<p align="center"> Questa è la mia prima pagina web </p>
</B>
<br>
<br>

<A HREF="http://web.tiscali.it/dagoproff">Sito prof. D'Agostino</A>
</body>
</html>
```

Salvare il file sul dischetto (o vostra cartella in Documenti) con il nome “vostro cognome. html”.

Verificare che l'icona del file sia quella di Internet Explorer (o altro Browser) e quindi ESEGUIRE il programma dando doppio click sull'icona.

Il vostro programma in HTML verrà interpretato da Internet Explorer (o altro Browser) che lo eseguirà, visualizzando la vostra pagina web.