

HTML

Introduzione

Il linguaggio HTML è un tipo di documenti SGML. Questo significa che esiste un DTD di HTML che è espresso secondo le regole menzionate.

HTML viene usato dai browser WWW per visualizzare documenti ipertestuali. I browser sono poco preoccupati della correttezza sintattica o strutturale dei documenti HTML secondo le regole date dal DTD SGML.

Questo significa che tra un documento HTML visualizzabile da un browser WWW ed un documento HTML corretto esistono differenze anche sensibili. A noi invece interessano documenti HTML corretti secondo la definizione del DTD.

È normale associare un significato strutturale agli elementi definiti in un DTD. HTML associa anche significati grafici agli elementi che definisce. Ciò dà istruzioni più o meno precise su come rendere graficamente gli elementi che definisce. Questo porta anche ad abusi della resa grafica che a noi interessano poco. Per noi la resa grafica finale, in assenza di linguaggi di stile appropriato, è **secondaria**.

HTML non forza strutture gerarchiche nei suoi documenti. Inoltre in HTML i vincoli di contenimento tra elementi sono pochi e piuttosto ovvi. I browser WWW sono ancora più lassisti a questo proposito. Tuttavia vanno rigorosamente rispettati.

Gli identificatori di elementi ed attributi sono *case-insensitive*. Questo significa che maiuscolo o minuscolo sono irrilevanti (anche combinazioni di maiuscolo e minuscolo). Tuttavia nelle entità, nei valori degli attributi e ovviamente nel contenuto del testo maiuscole e minuscole possono essere importanti.

Stili di blocco e stili di carattere

```
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 3.2//EN">
```

```
<HTML>
```

```
<HEAD>
```

```
<TITLE>Titolo del documento</TITLE>
```

```
</HEAD>
```

```
<BODY>
```

```
<H1>Intestazione maggiore</H1>
```

```
<P>Questo &acute; un paragrafo completo di un documento. Scrivo e scrivo finch&grave; non riempio varie righe, perch&grave; voglio vedere come va a capo automaticamente. Certo non &acute; un documento molto eccitante.</P>
```

```
<P>Forse che vi aspettavate <B>poesia</B>?</P>
```

```
<HR>
```

```
Una lista di cose importanti da ricordare
```

```
<UL>
```

```
<LI>Gestione di spazi, tabulatori e ritorni a capo</LI>
```

```
<LI>Dichiarazione di tipo del documento</LI>
```

```
<LI>Struttura del documento</LI>
```

```
<LI>Annidamento e chiusura dei tag</LI>
```

```
</UL>
```

```
</BODY>
```

```
</HTML>
```



Commenti

- All'inizio del documento, la dichiarazione `<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 3.2//EN">` serve per identificare in SGML il tipo di documento realizzato. I browser WWW non ne hanno bisogno, ma noi la mettiamo sempre.
- Gli spazi tra un elemento e l'altro hanno un significato speciale: spazi, ritorni a capo e tabulazioni, anche multipli, sono egualmente trasformati in spazi semplici nella visualizzazione dei dati. Quindi è uguale scrivere:

```
<P>Forse che vi aspettavate <B>poesia</B>?</P>
```

e

```
<P>Forse che vi aspettavate  
<B>poesia</B>?</P>
```

e anche

```
<P>Forse      che      vi  
aspettavate  <B>poesia</B>?</P>
```

ma non

```
<P>Forse che vi aspettavate <B>poesia</B>
?
</P>
```

- Tag di struttura del documento:

HTML, HEAD, BODY, TITLE

- Tag di strutturazione di blocchi:

Intestazione: H1, H2, H3, H4, H5, H6

Paragrafo: P, PRE, XMP, LISTING, ADDRESS

Liste: UL, OL, DIR, MENU, DL

- Tag di strutturazione dei caratteri:

Elementi idiomatici: CITE, CODE, EM, KBD, SAMP, STRONG, VAR

Elementi tipografici: B, I, TT





Netscape: Titolo del documento

Back Forward Home Edit Reload Images Print Find Stop

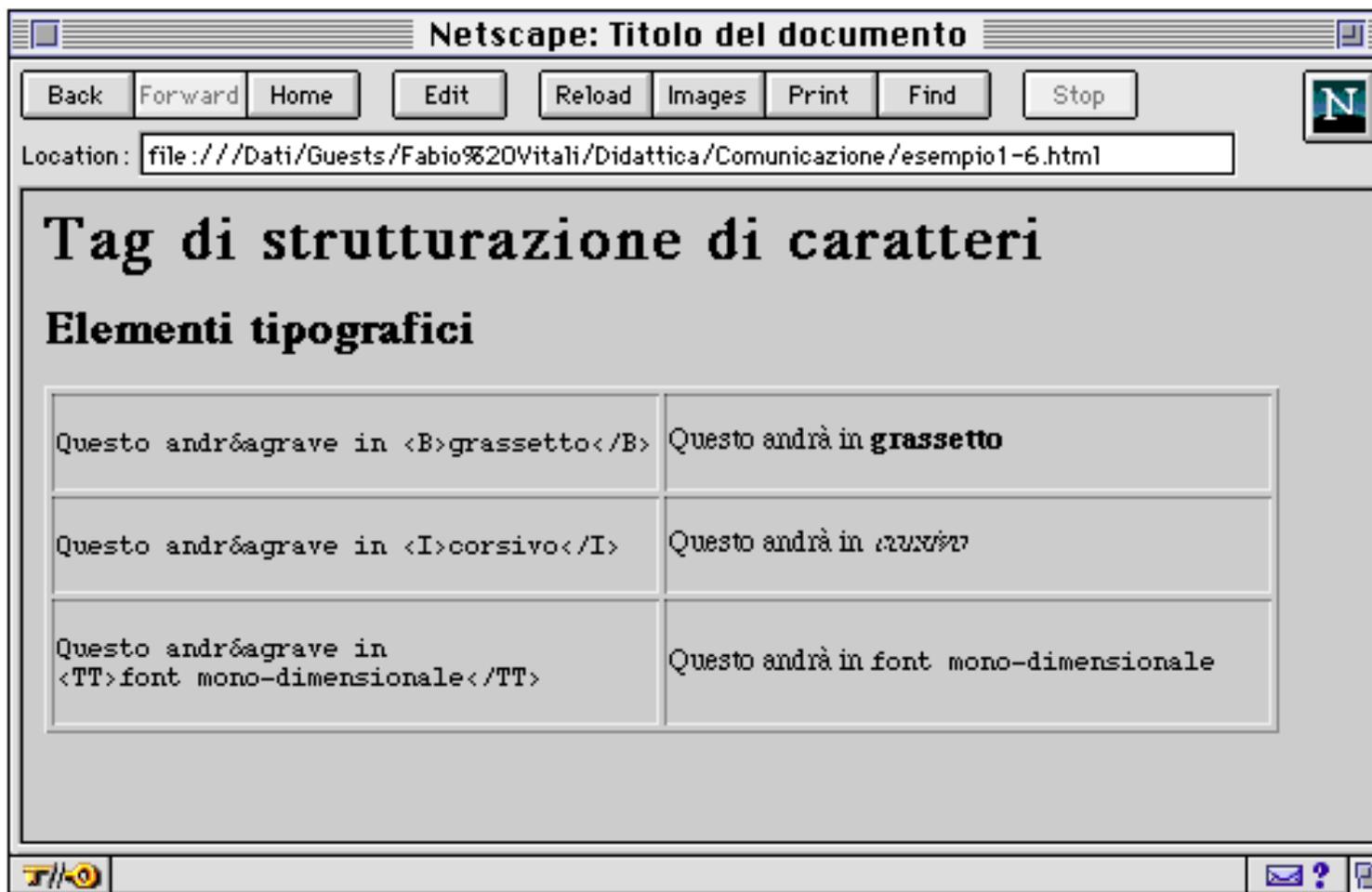
Location: file:///Dati/Comunicazione/esempio1ter.html

Tag di strutturazione di blocchi

Paragrafi

<pre><P>One nice feature of XML is that it is easily processable by the Desperate C/C++/Java/Perl hacker: the syntax is simple enough that you can do useful things with XML without a full XML parser. </P></pre>	<p>One nice feature of XML is that it is easily processable by the Desperate C/C++/Java/Perl hacker: the syntax is simple enough that you can do useful things with XML without a full XML parser.</p>
<pre><PRE>One nice feature of XML is that it is easily processable by the Desperate C/C++/Java/Perl hacker: the syntax is simple enough that you can do useful things with XML without a full XML parser. </PRE></pre>	<p>One nice feature of XML is that it is easily processable by the Desperate C/C++/Java/Perl hacker: the syntax is simple enough that you can do useful things with XML without a full XML parser.</p>
<pre><XMP>One nice feature of XML is that it is easily processable by the Desperate C/C++/Java/Perl hacker: the syntax is simple enough that you can do useful things with XML without a full XML parser. </XMP></pre>	<p>One nice feature of XML is that it is easily processable by the Desperate C/C++/Java/Perl hacker: the syntax is simple enough that you can do useful things with XML without a full XML parser.</p>
<pre><ADDRESS>Questo documento fu scritto da Fabio Vitali ad uso del corso di Informatica Generale per Scienze della Comunicazione </ADDRESS></pre>	<p><i>Questo documento fu scritto da Fabio Vitali ad uso del corso di Informatica Generale per Scienze della Comunicazione</i></p>

1/10 ?



Tag di strutturazione di caratteri

Elementi Idiomatici

Come dice Vitali, "<CITE>si usa CITE per fare citazioni</CITE>"	Come dice Vitali, " <i>si usa CITE per fare citazioni</i> "
Questo ´ codice: <CODE>void main() </CODE> ` il punto di partenza dei programmi C	Questo ´ codice: void main() ´ il punto di partenza dei programmi C
E allora non farlo!	E allora <i>non farlo!</i>
Premere <KBD>F1</KBD> per continuare	Premere F1 per continuare
La sequenza <SAMP>ABCDCBA</SAMP> ´ palindroma	La sequenza ABCDCBA ´ palindroma
Ti dico di non farlo!	Ti dico di non farlo!
La variabile <VAR>i</VAR> ´ la guardia del ciclo	La variabile <i>i</i> ´ la guardia del ciclo

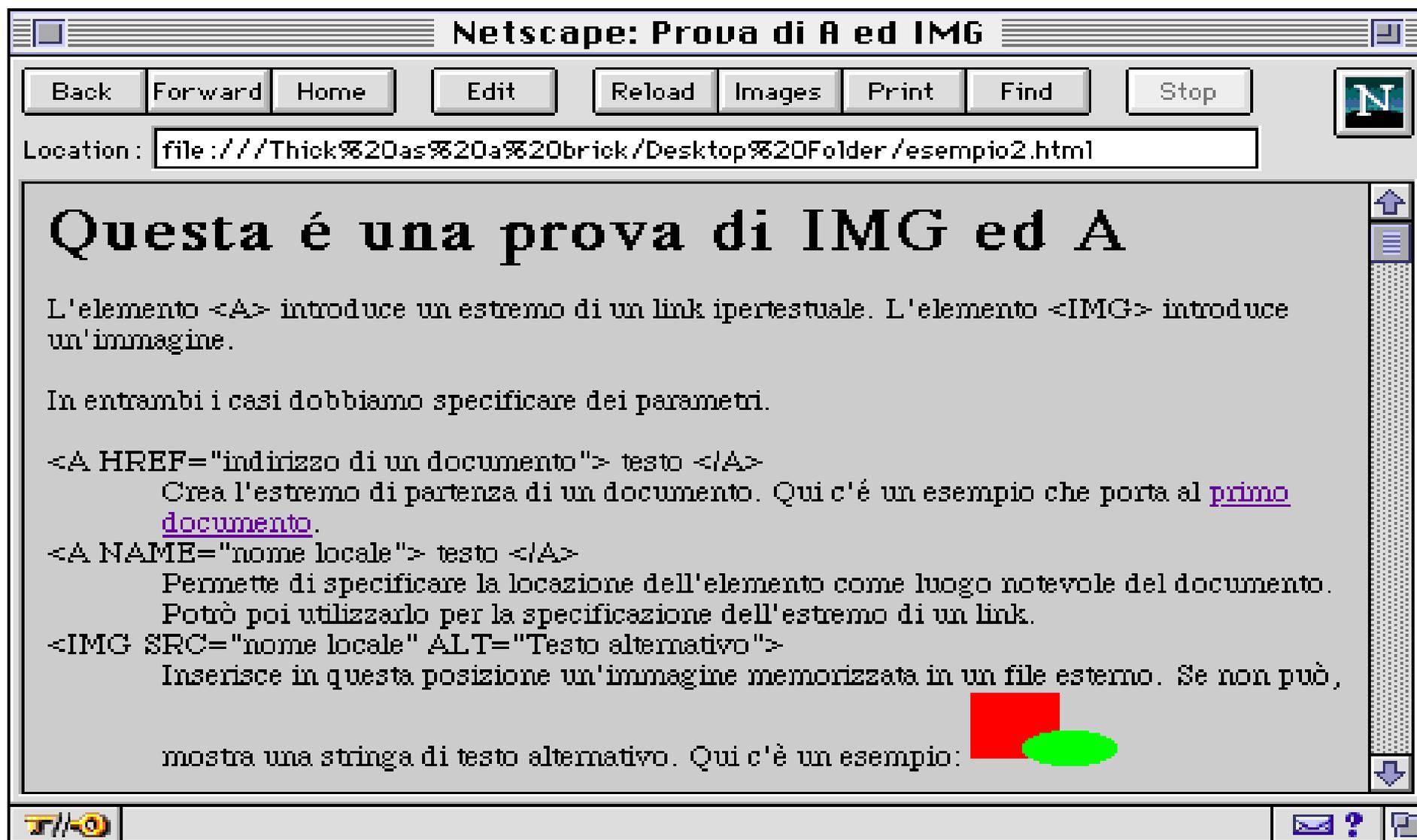
Link ed immagini

```
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 3.2//EN">
<HTML>
<HEAD>
<TITLE>Prova di A ed IMG</TITLE>
</HEAD>

<BODY>
<H1>Questa &acute; una prova di IMG ed A</H1>

<P>L'elemento &lt;A&gt; introduce un estremo di un link
ipertestuale. L'elemento &lt;IMG&gt; introduce
un'immagine. </P>
<P>In entrambi i casi dobbiamo specificare dei
parametri.</P>
```

```
<DL>
  <DT> &lt;A HREF="indirizzo di un documento"&gt; testo
&lt;/A&gt;
  <DD> Crea l'estremo di partenza di un documento. Qui
c'è un esempio che porta al <A
HREF="esempio1.html">primo documento</A>.
  <DT> &lt;A NAME="nome locale"&gt; testo &lt;/A&gt;
  <DD> Permette di specificare la locazione
dell'elemento come luogo notevole del documento.
Potrà poi utilizzarlo per la specificazione <A
NAME="prova">dell'estremo di un link</A>.
  <DT> &lt;IMG SRC="nome locale" ALT="Testo
alternativo"&gt;
  <DD> Inserisce in questa posizione un'immagine
memorizzata in un file esterno. Se non può,
mostra una stringa di testo alternativo. Qui c'è
un esempio: <IMG SRC="esempio.gif" ALT="Un rettangolo
ed un'ellisse">
</DL>
</BODY>
</HTML>
```



Commenti

- Un anchor (elemento A) **deve** avere o un attributo HREF o un attributo NAME. Può averli entrambi. Un'immagine **deve** avere l'attributo SRC. E' molto opportuno che abbia anche l'attributo ALT.
- L'attributo HREF specifica una destinazione. Quindi `` è un estremo di partenza. La destinazione è specificata come un URL (*Universal Resource Locator*).
- Gli URL sono o assoluti ("`http://www.cs.unibo.it/~fabio/index.html`") o relativi ("`esempio2.html`"). Distinzione tra i due.
- L'attributo NAME specifica una stringa come *nome* dell'anchor. E' poi possibile utilizzarlo in altri contesti. L'uso più comune è come destinazione puntuale di un link. Es: `click `

- L'attributo SRC specifica l'URL del file contenente l'immagine. L'URL, di nuovo, può essere o assoluto o relativo. Due sono i formati accettati comunemente: GIF e JPEG. Ci sarà una lezione sui formati grafici in seguito.
- A permette altri attributi:

TITLE: indica il titolo da dare ad un documento senza titolo

REL: specifica il tipo di relazione esistente tra il documento di partenza e quello di arriva. Importantissimo, ma solitamente inutilizzato.

REV: specifica il tipo inverso di relazione esistente tra partenza ed arrivo.

- IMG permette altri attributi:

ALIGN: permette di specificare come allineare l'immagine al testo circostante. Può avere i valori TOP, MIDDLE e BOTTOM.

ISMAP (senza valori): indica che l'immagine è una mappa server-side. La vedremo in seguito.

WIDTH: Può forzare una larghezza dell'immagine. L'immagine può venire espansa o ristretta per forzare la larghezza indicata, o semplicemente tagliata. L'unità di misura è il pixel.

HEIGHT: forza una altezza dell'immagine. Vale il discorso che vale per WIDTH.

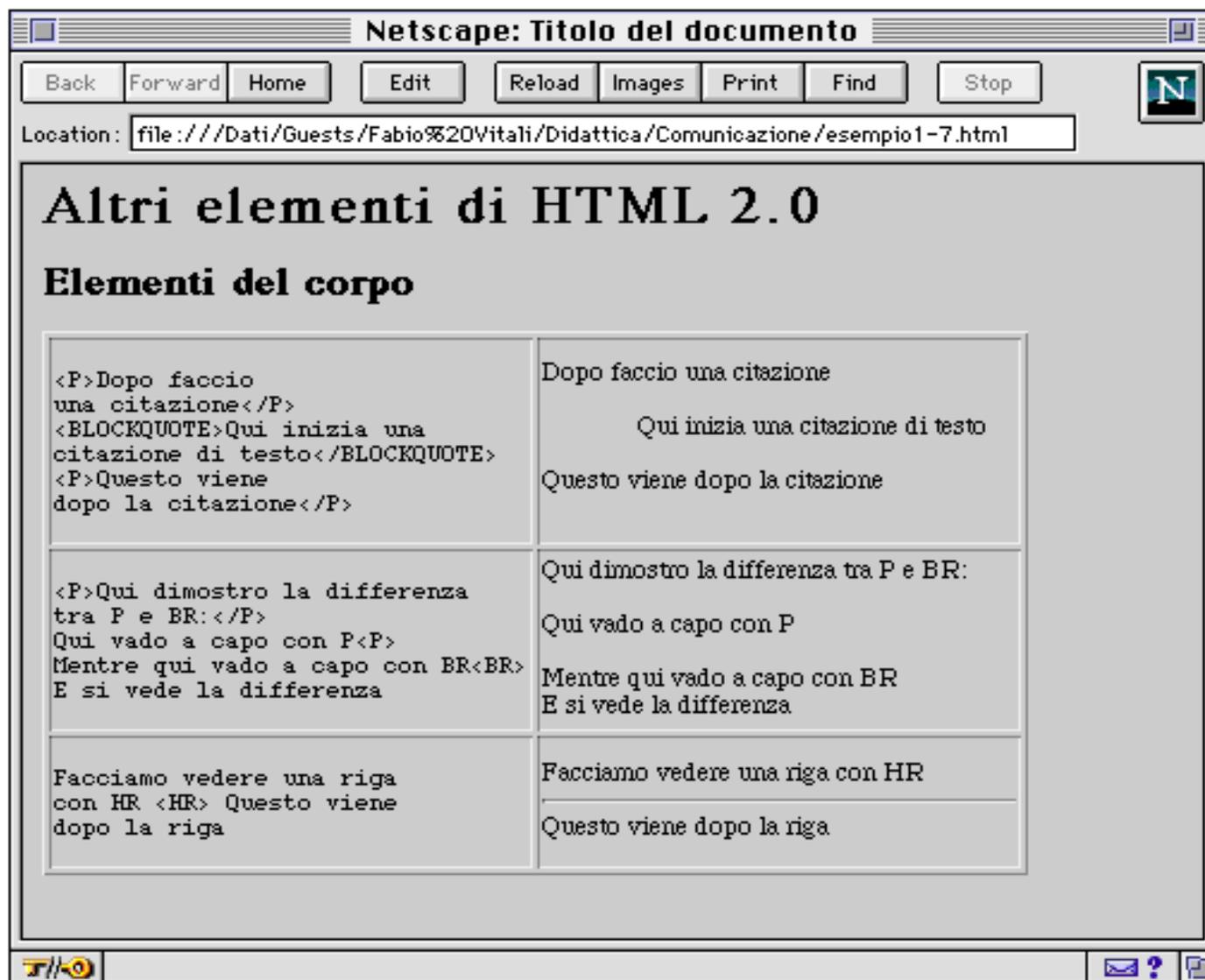
Altri elementi di HTML 2.0

Elementi del corpo

BLOCKQUOTE: Contiene paragrafi di testo presi da un'altra fonte. Tipicamente vengono resi con un margine diverso (più stretto) e forse in corsivo.

BR: Forza un ritorno a capo nel testo. Tipicamente viene reso senza spazio tra le righe

HR: crea un elemento divisorio (una riga, tipicamente) tra sezioni diverse del testo.



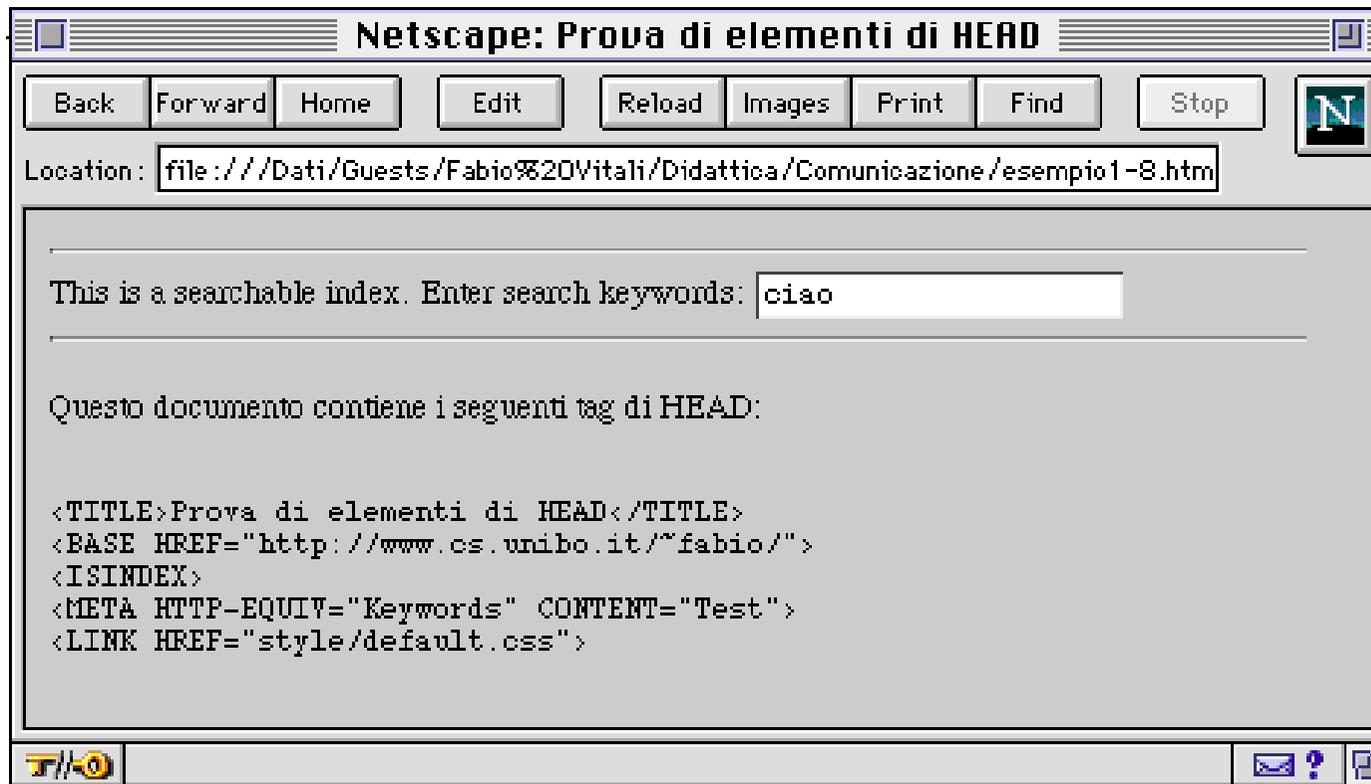
Elementi di HEAD

BASE: Specifica l'indirizzo di base rispetto al quale considerare gli URL relativi di link ed immagini del documento. Deve essere un URL assoluto.

ISINDEX: Indica che il documento è indicizzato presso il server. Permette di specificare stringhe di ricerca sul documento. Equivalente ad un metodo GET di un form. Non ha contenuto.

LINK: Specifica l'esistenza di un documento collegato globalmente al documento visto.

META: Specifica l'esistenza di una metainformazione associata al documento. In particolare l'esistenza di intestazione MIME del documento.

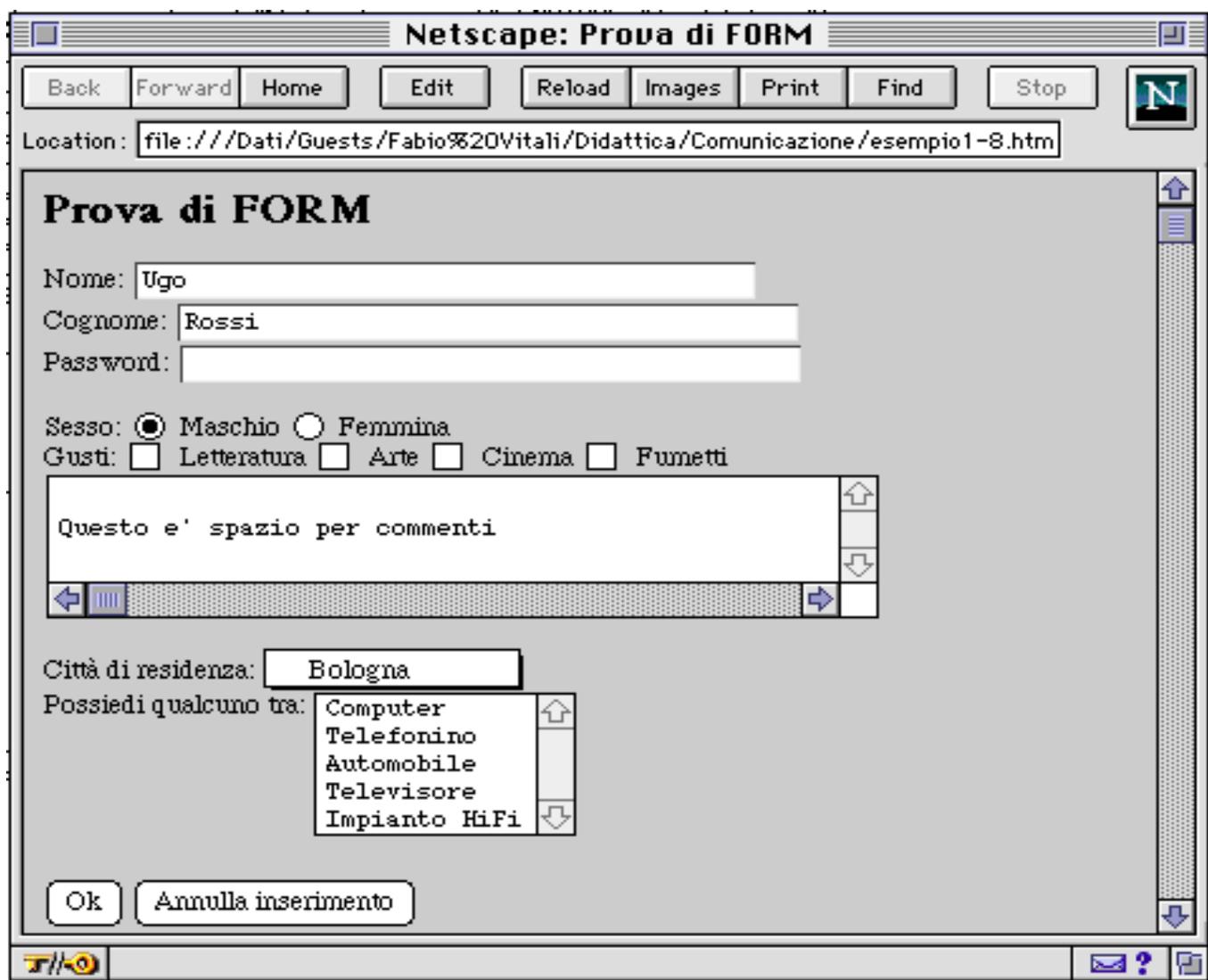


FORM

Con i FORM si utilizzano le pagine HTML per inserire valori che vengono poi elaborati sul server. I FORM sono inestricabilmente legati al CGI (Common Gateway Interface):

Il browser raccoglie dati con un form dall'utente. Crea una connessione HTTP con il server, specificando una ACTION (cioè un destinatario) a cui fare arrivare i dati così raccolti. Il destinatario riceve i dati, li elabora e genera un documento di risposta, che viene spedito, tramite il server HTTP, al browser.

I form permettono di specificare l'inserimento di dati in modulistica: campi di inserimento dati, pulsanti, bottoni radio, checkbox, liste a scomparsa.



```
<P> Questo documento contiene una prova di FORM:</P>
<FORM METHOD="GET" ACTION="http://www.cs.unibo.it/~fabio/pippo.pl">
<BR>Nome: <INPUT TYPE=text NAME="Nome" VALUE="Ugo" SIZE=40>
<BR>Cognome: <INPUT TYPE=text NAME="Cognome" VALUE="Rossi" SIZE=40>
<BR>Password: <INPUT TYPE=password NAME="Password" SIZE=40>
<P>Sesso: <INPUT TYPE=radio NAME="Sesso" VALUE="Maschio" CHECKED>Maschio
        <INPUT TYPE=radio NAME="Sesso" VALUE="Femmina"> Femmina
<BR>Gusti: <INPUT TYPE=checkbox NAME="Gusti" VALUE="Lett"> Letteratura
        <INPUT TYPE=checkbox NAME="Gusti" VALUE="Arte"> Arte
        <INPUT TYPE=checkbox NAME="Gusti" VALUE="Cinema"> Cinema
        <INPUT TYPE=checkbox NAME="Gusti" VALUE="Fumetti"> Fumetti
<BR><TEXTAREA NAME="Commenti" ROWS=3 COLS=50>Questo e' spazio per commenti</TEXTAREA>
<INPUT TYPE=hidden NAME="Lingua" VALUE="Italiana">
<P>Città di residenza: <SELECT NAME="Citta'">
  <OPTION SELECTED>Bologna <OPTION>Modena
  <OPTION>Forlì <OPTION>Ravenna
</SELECT>
<BR>Possiedi qualcuno tra: <SELECT MULTIPLE NAME="Possesso" SIZE=5>
  <OPTION>Computer <OPTION>Telefonino <OPTION>Automobile <OPTION>Televisore
<OPTION>Impianto HiFi
</SELECT>
<P><INPUT TYPE=submit NAME="Submit" VALUE="Ok">
  <INPUT TYPE=reset NAME="Cancel" VALUE="Annulla inserimento">
</FORM>
```

Commenti su FORM

ACTION è obbligatoria, e specifica l'URL dell'applicazione che riceverà i dati del form. Tipicamente è un'applicazione CGI ed è un URL con protocollo HTTP. Netscape supporta anche il protocollo mailto.

METHOD: indica il metodo HTTP utilizzato. Può essere o GET o POST. GET è opportuno per piccoli form. È equivalente ad una query da URL. POST serve per form più lunghi.

ENCTYPE: permette di specificare il formato di dati con cui vengono trasmessi i dati al server. Per default è il tipo "application/x-www-form-urlencoded". Funziona così:

```
Nome=Ugo&Cognome=Rossi&Password=&Sesso=Maschio&Lingua=Italiana&Commenti=%0D%0AQuesto+e'+spazio+per+commenti%0D%0A&Citta%=Bologna&Submit=Ok
```

I campi sono messi nell'ordine del form, nella forma campo=valore, e separati da &. Tutti i caratteri proibiti nei form (ASCII>127, ritorni a capo, ecc) sono codificati nella forma %xx dove xx è il codice esadecimale del carattere. Gli spazi sono sostituiti da +

Commenti su INPUT

L'attributo TYPE specifica il tipo di oggetto da utilizzare. Valori possibili sono: text, password, radio, checkbox, submit e reset.

L'attributo SIZE specifica la dimensione in caratteri degli INPUT di tipo testo e password.

L'attributo VALUE è il valore che viene passato all'applicazione CGI. Nel caso di testo e password viene visualizzato sullo schermo. Nel caso di reset e submit viene visualizzato all'interno del campo.

L'attributo CHECKED è utilizzabile solo per radio e checkbox, ed indica lo stato di default per il pulsante: in questo caso è attivato.

Il tipo PASSWORD mostra dei pallini al posto dei caratteri, ma il valore della password viene trasmesso in chiaro (cioè non crittografato) nella trasmissione all'applicazione CGI.

Il tipo RADIO permette di scegliere uno tra molti possibili valori. L'unicità è data dal fatto che il nome dell'oggetto di input è lo stesso.

Il tipo SUBMIT crea un pulsante attivando il quale viene trasmesso il form. Il tipo RESET invece svuota i campi e ripristina la situazione di default.

Commenti su TEXTAREA e SELECT

TEXTAREA prevede come attributi solo COLS e ROWS, che indicano la larghezza e l'altezza rispettivamente della zona di testo. Il contenuto del tag è il valore di default dell'oggetto.

I SELECT sono di due tipi: se è previsto l'attributo MULTIPLE visualizza una lista da cui è possibile scegliere più di un valore. Altrimenti visualizza una menu pop-up da cui è possibile scegliere un solo valore.

In caso di SELECT MULTIPLE, l'attributo SIZE specifica quanti elementi vengono visualizzati contemporaneamente.

OPTION introduce le singole scelte di un SELECT. Se si vuole che la scelta sia automaticamente selezionata, bisogna usare l'attributo SELECTED.

Entità in HTML

HTML definisce un certo numero di entità per quei caratteri che sono:

- proibiti perché usati in HTML (<, >, &, “, ecc.)
- proibiti perché non presenti nell'ASCII a 7 bit.

amp	&	quot	“
lt (<i>less than</i>)	<	gt (<i>greater than</i>)	>
reg	®	nbsp (<i>non-breaking space</i>)	
Aelig	Æ	Aacute	Á
Agrave	À	Auml	Ä
aelig	æ	aacute	á
agrave	à	auml	ä
ccedil	ç	ntilde	ñ

ecc.

HTML 3.2

Novità in HEAD

STYLE: HTML 3.2 non definisce criteri di stile, ma prevede uno spazio dove inserire delle specifiche, che dipendono dal browser utilizzato.

SCRIPT: HTML 3.2 non definisce script da eseguire sul client, ma prevede uno spazio dove inserire le specifiche, che dipendono dal browser utilizzato.

ISINDEX: E' possibile specificare la stringa di richiesta:

```
<INSINDEX prompt="Inserisci una parola">
```

Novità in BODY

BODY permette degli attributi:

- BGCOLOR: colore del background
- TEXT: colore del testo
- LINK: colore del link non visitato
- VLINK: colore del link visitato
- ALINK: colore del link attivato

I colori sono introdotti dal carattere # e composti da tre numeri esadecimali, che rappresentano la quota di componente di rosso, verde e blu del colore.

Es:

#000000 = Nero

#FF0000 = Rosso

#00FF00 = Verde scuro

#0000FF = Blu scuro

#FFFFFF = Bianco

#FFFF00 = Giallo

Allineamenti e liste

Tutti gli heading (**H1**, **H2**, ..., **H6**), **P** ed il nuovo elemento **DIV** consentono l'attributo **align** che permette di specificare che allineamento dare al paragrafo:

```
<P ALIGN=RIGHT>  
<H1 ALIGN=CENTER>  
<H3 ALIGN=LEFT>
```

DIV è un nuovo tipo di elemento di blocco. Simile al paragrafo, ma non impone uno spazio tra paragrafi.

```
<DIV ALIGN=RIGHT> kdjbdkas </DIV>
```

UL e **OL** hanno l'attributo **type** per stabilire il tipo di numerazione o di pallini:

```
<UL type=disc> (oppure square, circle)  
<OL type=1> (o a, A, i, I)
```

Nuovi elementi di testo

All'interno dei paragrafi, sono definiti nuovi elementi di testo:

STRIKE: causa la scrittura di testo barrato

BIG: sceglie il font di una dimensione maggiore del normale

SMALL: sceglie il font di una dimensione minore del normale

SUB: mette il testo in posizione di pedice

SUP: mette il testo in posizione di apice

FONT: permette di scegliere caratteristiche assolute come dimensioni e colore. La dimensione (**size**) va da 1 a 7, con 3 valore normale. Posso usare sia valori assoluti che relativi (+1, -2, ecc.). Il colore (**color**) segue le regole dei colori visti in precedenza.

```
<FONT size=3 color=#FF00FF> sldsdlj </FONT>
```

Alcuni browser ammettono l'attributo **face** per scegliere il nome del font, con una lista separata da virgole di suggerimenti. Non fa parte di HTML 3.2.

Che cosa vuoi cercare?

HTML 3.2

<pre><P ALIGN=LEFT>Allineato a sinistra</P> <P ALIGN=CENTER>Allineato al centro</P> <P ALIGN=RIGHT>Allineato a destra</P></pre>	<p>Allineato a sinistra</p> <p>Allineato al centro</p> <p>Allineato a destra</p>
<pre> <LI type=disc>Disco <LI type=square>Quadrato <li type=circle>Cerchio </pre>	<ul style="list-style-type: none">● Disco□ Quadrato○ Cerchio
<pre><OL type=A> Primo Secondo Terzo </pre>	<ul style="list-style-type: none">A. PrimoB. SecondoC. Terzo
<pre>Qui inizia del <STRIKE>testo barrato <SMALL>più grave; piccolo</SMALL> </STRIKE> poi normale <BIG>poi grande</BIG><SMALL><SUP> poi piccolo sopra</SUP></SMALL>
H<SUB><SMALL>2</SMALL></SUB>O</pre>	<p>Qui inizia del testo barrato più piccolo poi normale poi grande poi piccolo sopra H₂O</p>

Tabelle

La tabella viene specificata riga per riga. Di ogni riga si possono precisare gli elementi, che sono o intestazioni o celle normali. Una tabella può anche avere una didascalia.

TABLE introduce una tabella. Le tabelle possono annidarsi.

TR introduce una riga di una tabella

TH introduce una cella intestazione. TD una cella normale.

CAPTION introduce una didascalia.

Netscape: Tabelle










Location:

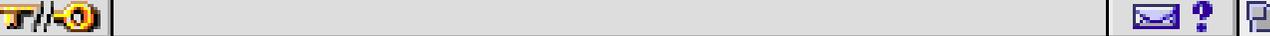
Tabelle

```

<TABLE BORDER=1 CELLPADDING=5>
  <TR><TH>Mese      <TH> Entrate <TH> Uscite </TR>
  <TR><TH>Gennaio   <TD> 12.000 <TD> 10.000 </TR>
  <TR><TH>Febbraio  <TD> 11.500 <TD>  9.000 </TR>
  <TR><TH>Marzo     <TD> 14.000 <TD> 11.000 </TR>
</TABLE>

```

Mese	Entrate	Uscite
Gennaio	12.000	10.000
Febbraio	11.500	9.000
Marzo	14.000	11.000



Netscape: Tabelle

Location: file:///Dati/Guests/Fabio%20Vitali/Didattica/Comunicazi

Tabelle

```

<TABLE BORDER=1 CELLPADDING=10>
  <CAPTION>Una tabella con celle multiple</CAPTION>
  <TR><TH ROWSPAN=2>&nbsp;</TH>
    <TH COLSPAN=2>Media
    <TH ROWSPAN=2>Popolazione
    <TH>Altro</TH></TR>
  <TR><TH>Altezza
    <TH>Peso</TR>
  <TR><TH ALIGN=LEFT>Uomini
    <TD>1.8
    <TD>75
    <TD>48%</TR>
  <TR><TH ALIGN=RIGHT>Donne
    <TD>1.68
    <TD>58
    <TD>52%</TR>
</TABLE>

```

Una tabella con celle multiple

	Media		Popolazione	Altro
	Altezza	Peso		
Uomini	1.8	75	48%	
Donne	1.68	58	52%	

Attributi di TABLE

ALIGN: allineamento della tabella rispetto al documento

WIDTH: larghezza (in assoluto o in percentuale) rispetto al documento

BORDER: larghezza della riga dei bordi. Con 0 non c'è bordo.

CELLSPACING: Spaziatura tra i bordi delle celle.

CELLPADDING: Spaziatura tra il bordo e il testo della cella.

Attributi di TR

HALIGN (left, center, right): allineamento orizzontale della riga

VALIGN (top, bottom): allineamento verticale della riga

Attributi di TR e TD

NOWRAP (sopprime l'andata a capo)

ROWSPAN: numero di righe della cella

COLSPAN: numero di colonne della cella

HALIGN (left, center, right): allineamento orizzontale della riga

VALIGN (top, bottom): allineamento verticale della riga

WIDTH: larghezza della cella

HEIGHT: altezza della cella

Mappe: Mappe server-side

L'attributo "ismap" del tag IMG serve per indicare che l'immagine in questione è una mappa sensibile al click di tipo server-side. Cioè:

```
<A HREF="/cgi-bin/imagemap/mappa">  
  <IMG SRC="immagine.gif" ISMAP>  
</A>
```

viene interpretato come segue: ogni volta che l'utente fa un click col mouse sull'immagine 'immagine.gif', la posizione del click viene spedita come argomento al link ipertestuale relativo, che attiva il programma "/cgi-bin/imagemap/" con "mappa" come argomento di specifica. "Mappa" è un file che indica le forme delle zone sensibili al click dell'immagine, e associa ad ogni zona una destinazione diversa. "/cgi-bin/imagemap/", allora, confronta la posizione del click con la forma delle zone dell'immagine, e attiva la destinazione corrispondente. Questo è chiamato un *imagemap server-side*, perché la corrispondenza tra click e zona vien fatta sul server.

Mappe client-side

HTML 3.2 permette imagemap client-side, cioè tali per cui la corrispondenza viene fatta sul browser, poiché l'elenco delle zone viene spedito insieme al documento. Esso ha questa forma:

```
<IMG SRC=immagine.gif USEMAP="#pippo">  
<MAP NAME="pippo">  
  <AREA shape=rect coord="0,0,100,100" href="doc1.html">  
</MAP>
```

Questo viene interpretato come segue: l'immagine *usa la mappa* con URL "#pippo". L'URL è locale, e quindi viene cercato all'interno del documento quella MAP che ha nome "pippo".

SHAPE stabilisce la forma della zona in questione. Può essere **rect** (rettangolo), **circle** (cerchio) o **poly** (poligonale).

COORD specifica le coordinate dell'oggetto. Consiste in una stringa composta da una lista di numeri separati da virgole.

```
shape=rect coord="left-x, top-y, right-x, bottom-y"
```

```
shape=circle coord="center-x, center-y, radius"
```

```
shape=poly coord="x1, y1, x2, y2, . . . . , xn, yn"
```

HREF indica la destinazione associata con la zona specificata. Se non si vuole associare una destinazione ad una zona, si usa l'attributo (senza valore) NOHREF.

ALT permette di precisare un testo mostrato nella zona di status o elementi per creare un menù di scelte nel caso di un browser non-grafico.

Estensioni proprietarie

Netscape Navigator e Communicator

BLINK: Lampeggia il testo

```
<BLINK> Questo lampeggia! </BLINK>
```

EMBED: permette di inserire oggetti esterni, tramite plug-in. Se hanno una visualizzazione, è necessario prenotare spazio del documento.

```
<EMBED SRC=URL WIDTH=n HEIGHT=n>Testo alternativo </EMBED>
```

MULTICOL: divide automaticamente il testo tra più colonne. E' possibile precisare il numero di colonne e la distanza (*gutter*) tra una colonna e l'altra.

```
<MULTICOL COLS=n GUTTER=pixel WIDTH=perc> Testo  
</MULTICOL>
```

LAYER: Permette di precisare livelli sovrapposti di testo. Funziona con Javascript.

Microsoft Internet Explorer

BGSOUND: Un suono o musica da emettere durante la visualizzazione della pagina. Non può essere né inibito, né interrotto.

```
<BGSOUND SRC="URL" LOOP=n>
```

Tabelle: è possibile associare caratteristiche globali alle colonne di tabelle. Ha senso con gli stili CSS (di cui parleremo). E' inoltre possibile specificare altri elementi di tabella come la testa e la coda.

<COL>	Table Column
<COLGROUP>	Table Column Group
<TBODY>	Table Body
<TFOOT>	Table Footer
<THEAD>	Table Head

MARQUEE: permette di specificare un testo che scrolla sullo schermo come un'insegna luminosa. Molto fastidioso.

```
<MARQUEE BEHAVIOR="scroll|slide|alternate"  
DIRECTION="left|right" LOOP=n>testo</MARQUEE>
```

Frame

Un documento HTML di Netscape ed Internet Explorer (ma NON W3C!!!) può contenere o un BODY o un FRAMESET.

I frame servono per dividere la finestra in più zone indipendenti ed associare a ciascuna di loro un documento diverso. Ogni zona (o *frame*) ha un nome, che viene usata per specificare in quale zona viene visualizzato la destinazione di un link.

FRAMESET introduce una serie di frame o verticali o orizzontali. FRAMESET può contenere annidati altri FRAMESET (per cambiare l'orientamento).

La definizione dei link A cambia per rendere possibile precisare il nome di una zona o di una finestra come destinazione del documento.

La parte NOFRAME contiene il documento che deve essere visualizzato da parte dei browser che non supportano i frame.

```
<HTML>
  <HEAD>
    <TITLE> Prova frame </FRAME>
  </HEAD>
  <FRAMESET COLS="50%,*">
    <FRAME SRC="pippo.html" NAME="sinistra">
    <FRAME SRC="pluto.html" NAME="destra">
  </FRAMESET>
  <NOFRAMES>
  <BODY>
    etc...
  </BODY>
</HTML>
```

FRAMESET specifica il numero di colonne o di righe della serie di frame. Vengono precisati con una sequenza di valori in pixel o proporzionali rispetto alla finestra. Il termine ‘*’ significa ‘tutto lo spazio rimanente’.

```
<FRAMESET ROWS="100,57,*"  
<FRAMESET COLS="20%,*,70%">  
<FRAMESET ROWS="*,*,*">
```

FRAME specifica le caratteristiche di un frame, indicandone l’URL di default ed il nome da usare nei link. L’attributo SCROLLING (valori: yes, no, auto) precisa se debbano essere visualizzate o meno le barre di scrolling. L’attributo NORESIZE (senza valore) specifica che non è possibile ridefinire per l’utente la dimensione del frame.

```
<FRAME SRC=URL NAME="sinistra" SCROLLING="yes"  
NORESIZ>
```

NOFRAMES contiene il documento sostitutivo da mostrare da parte dei browser che non conoscono i frame. Esso contiene una parte BODY regolare. E’ importante ricordarsi di metterla e testarla.

Usare i frame nei link: È definito un nuovo attributo, TARGET, per specificare in quale frame o finestra debba comparire il documento. Viene specificato il nome del frame oppure quattro nomi speciali (non utilizzabili come nomi di frame):

_blank: forza la destinazione in una nuova finestra

_self: forza la destinazione nella stessa finestra

_parent: forza la destinazione al posto del frameset di cui fa parte

_top: forza la destinazione nella finestra intera di cui fa parte.

```
<A HREF=URL TARGET="sinistra">
```

IFRAME: estensione di Explorer per creare un frame volante (una finestra autonoma sempre in primo piano).

```
<IFRAME SRC=URL NAME="winname">
```

Spazi

NOBR crea una zona di testo il cui contenuto non va mai a capo, a meno di specificare **esplicitamente** la rottura con il tag


```
<NOBR>testo che non va a capo tranne <BR> qui.</NOBR>
```

SPACER crea uno spazio bianco orizzontale o verticale o interna ad un paragrafo (attributo TYPE con valori **horizontal**, **vertical** o **block**) e di dimensione definita in pixel. Non ha contenuto.

```
<SPACER TYPE="horizontal" SIZE=50>
```

WBR specifica dove, all'interno di una parola, è possibile spezzarla. Permette cioè di specificare la divisione sillabica. E' utile per parole MOLTO lunghe.

```
Precipitevo<WBR>lissimevol<WBR>mente
```