

- ◆ Rete di reti
- ◆ Architettura client/server
- ◆ Server: Provider, Router, Web Server, Email Server
- ◆ Servizi: World Wide Web, Email, Chat, VideoConf, Telnet
- ◆ Clienti: browser (IE, Netscape, Mozilla, Opera), lettore email (Out. Exp., Eudora), IRC

1

- ⊕ **HTML** acronimo di **Hypertext Markup Language** ("Linguaggio di contrassegno per gli Iper testi")
- ⊕ È il linguaggio con cui le pagine web vengono scritte e progettate
- ⊕ Non è un linguaggio di programmazione
- ⊕ È un **linguaggio di contrassegno** (o 'di marcatura'), che permette di indicare come disporre gli elementi all'interno di una pagina: le indicazioni vengono date attraverso degli appositi marcatori, detti **"tag"**
- ⊕ Un file **HTML** è di tipo testo ("puro testo" o "testo ASCII"), come un file di tipo TXT
- ⊕ Può essere utilizzato un comune editor di testo
- ⊕ Per funzionare come pagina web, il file deve essere rinominato con estensione .HTML o .HTM

2

- ⊕ scarica i vari files .HTML che si trovano su un computer remoto (server web) e che fanno riferimento a un certo indirizzo www.pincopallino.it
- ⊕ legge i documenti scritti in html, e a seconda delle indicazioni ivi contenute, visualizza la pagina in un modo, piuttosto che in un altro; inoltre i vari files associati a quel documento (ad esempio le immagini, o i filmati in flash) vengono disposti secondo le indicazioni del codice html

3

- Per iniziare a scrivere pagine web avete bisogno di:
- ⊕ uno o più **browser** per visualizzare le pagine
 - ⊕ un **editor testuale** per scrivere il codice HTML (blocco note di Windows, o altri editor testuali come 1Page, che è gratuito)
 - ⊕ editor tipo **WYSIWYG** (What You See Is What You Get: quello che vedi è quello che ottieni) come Frontpage (Microsoft) e Dreamweaver (Macromedia)

4

Il file HTML contiene istruzioni che consentono al browser di riconoscerlo e gestirlo correttamente. Le istruzioni sono chiamate **Tag** (marcatori).

Un **tag** è un comando racchiuso tra i segni < (minore) e > (maggiore),

Quando il browser incontra questi simboli capisce che si tratta d'istruzione da eseguire e non di testo da visualizzare sullo schermo.

Nella maggior parte dei casi i **tag** funzionano in coppia, richiedono cioè un'istruzione d'apertura ed una di chiusura secondo lo schema:

```
<TAG>elemento cui si applica l'istruzione</TAG>
```

In molti Tag è possibile specificare uno o più attributi che modificano l'output

```
<TAG attrib1="valore" attrib2="valore">contenuto</TAG>
```

Ecco alcuni esempi

```
<P align="right">testo</P>
```

```
<IMG SRC="foto.jpg">
```

5

I tag possono essere annidati l'uno dentro l'altro. Anzi molto spesso è necessario farlo. Ad esempio:

```
<TAG1 attributi>
  contenuto 1
  <TAG2>
    contenuto 2
  </TAG2>
</TAG1>
```

Potremmo quindi avere ad esempio:

```
<P align="right">
  testo 1
  <P align="left">
    testo 2
  </P>
</P>
```

L'annidamento ci permette quindi di attribuire formattazioni successive al testo che stiamo inserendo.

6

Per rendere il nostro codice più leggibile è bene inserire **"commenti"** nei punti più significativi: si tratta di indicazioni significative per il webmaster, ma invisibili al browser. Inserendo i commenti in punti specifici del documento ci permette di mantenere l'orientamento anche in file molto complessi e lunghi. La sintassi è la seguente:

```
<!-- questo è un commento -->
```

e ci permette di "commentare" i vari punti della pagina. Ad esempio:

```
<!-- menu di sinistra -->
```

```
<!-- barra in alto -->
```

```
<!-- eccetera -->
```

L'HTML è **"case insensitive"**, cioè indipendente dal formato. Questo significa che è del tutto indifferente se scrivere i tag in maiuscolo o in minuscolo. **È PREFERIBILE IL MINUSCOLO**

7

La struttura essenziale di ogni pagina web è costituita da:

Il tag **<!DOCTYPE attributi>** che informa il browser che stiamo utilizzando le specifiche Word Wide Web Consortium .

I tag **<HTML>** e **</HTML>** all'inizio ed alla fine del **documento**; essi indicano al browser che tutto ciò che è compreso nel mezzo è una pagina web.

I 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. Il contenuto non viene visualizzato.

I tag **<BODY>** e **</BODY>** racchiudono il **corpo** del documento ovvero dove si inserisce il contenuto da visualizzare nella pagina.

```
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01
Transitional//IT">
<HTML>
<HEAD>...</HEAD>
<BODY>...</BODY>
</HTML>
```

8

I TAGS DI INTESTAZIONE: <TITLE>

I tag da inserire nell'intestazione sono: <TITLE> e <META>.

<TITLE> definisce il titolo della pagina web. Il titolo appare sulla barra superiore della finestra del browser quando la pagina viene visualizzata durante la navigazione. Questo tag richiede chiusura. La sintassi corretta è:

```
<HEAD>
<TITLE>Titolo della pagina</TITLE>
</HEAD>
```

È buona norma specificare il titolo di ogni pagina web perché una pagina senza titolo è anonima e poco professionale. Non è possibile inserire nella stessa intestazione più di un tag <TITLE>.

9

I TAGS DI INTESTAZIONE: <META>

<META> serve per inserire nell'intestazione dati che non vengono visualizzati nella pagina ma servono al browser e ai motori di ricerca per gestirne i contenuti in modo più efficace. *Nome dell'autore della pagina*, breve descrizione dei *contenuti della pagina*, elenco schematico di *parole-chiave legate ai contenuti*. Non richiede chiusura ma l'aggiunta di alcuni attributi/valori secondo lo schema

```
<META NAME="VALORE" CONTENT="VALORE">
```

dove NAME indica il tipo di informazione e CONTENT il contenuto concreto della meta-informazione. Ad esempio:

```
<META NAME=author CONTENT="nome e cognome">
```

Per inserire una descrizione del documento si utilizza:

```
<META NAME=description CONTENT="breve descrizione della pagina">
```

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

```
<META NAME=keywords CONTENT="parola1, parola2, parola3,...">
```

10

IL CORPO: <BODY>

La sezione <BODY> è il corpo principale del documento HTML dove vanno inseriti tutti i contenuti che devono apparire nella pagina web. Presenta numerosi attributi/proprietà

× Attributo **bgcolor**: serve per impostare un colore di sfondo

```
<BODY bgcolor="blue">
```

Non è consigliabile inserire la notazione del colore facendo riferimento al nome. È preferibile utilizzare la corrispondente codifica esadecimale del colore, che ci permette anche di scegliere anche tonalità di colore non standard.

```
<BODY bgcolor="#0000FF">
```

11

IL CORPO: <BODY>

Ecco una tabella con la notazione di alcuni colori (molti di essi sono disponibili anche nelle varianti "dark" e "light", ad esempio: "darkblue", "lightblue"):

| colore | parola chiave | notazione esadecimale |
|-----------|---------------|-----------------------|
| arancione | orange | #FFA500 |
| blu | blue | #0000FF |
| bianco | white | #FFFFFF |
| giallo | yellow | #FFFF00 |
| grigio | gray | #808080 |
| marrone | brown | #A52A2A |
| nero | black | #000000 |
| rosso | red | #FF0000 |
| verde | green | #008000 |
| viola | violet | #8B2323 |

12

IL CORPO: <BODY>

× Attributo **background**: serve per inserire una immagine come sfondo della pagina

```
<BODY background="immagine.gif">
```

Se l'immagine di sfondo NON si trova nella stessa cartella della nostra pagina HTML, è necessario anteporre al nome del file il relativo percorso.

L'immagine di sfondo verrà ripetuta in orizzontale e in verticale.

È anche possibile combinare bgcolor con background, in modo che mentre l'immagine di sfondo viene caricata, venga comunque visualizzata una colorazione della pagina.

È importante **assegnare sempre un colore alla pagina** anche quando lo sfondo della pagina è bianco (bgcolor="#FFFFFF"). In caso contrario il browser assegna alla pagina il colore di sfondo che l'utente ha impostato nella finestra del sistema operativo.

13

IL CORPO: <BODY>

× Attributi **leftmargin**, **topmargin**: servono per creare un po' di bordo tra la pagina e i limiti della finestra

Ad esempio si può impostare un bordo nullo

```
<BODY leftmargin="0" topmargin="0">
```

× Attributo **lang**: serve per impostare la lingua utilizzata nella pagina.

```
<BODY lang="it">
```

Questo attributo non è solo una proprietà del tag body, ma può essere riferito alla maggior parte dei tag HTML

14

IL CORPO: <BODY>

× Attributo **text**: serve per impostare il colore di base per il testo visualizzato nella pagina

```
<BODY text="green">
```

× Attributi **link**, **alink**, **vlink**: servono per impostare il colore utilizzato nei collegamenti.

Tre stati canonici dei link:

- **Link a riposo**, colorato per default con il blu (#0000FF).
- **Link visitato**, colorato per default con il viola (#800080).
- **Link attivo** quando è stato cliccato e sta avvenendo il passaggio da una pagina all'altra (caratteristica poco utile)

Esempio:

```
<BODY link="red" alink="yellow" vlink="green" text="blue">
```

15

IL CORPO: <BODY>

Esempio: clicca su [esempio1.html](#)

Conclusione

In un approccio di impaginazione più professionale, l'aspetto che riguarda la visualizzazione deve essere separato dal contenuto.

Le regole che indicano come visualizzare lo sfondo, il testo o i collegamenti saranno precisate nei Fogli di Stile o CSS

16

I TITOLI: <H>

Il tag <H> imposta uno stile predefinito per il testo. È utile per i titoli di pagina

| tag | Visualizzazione codice | Descrizione |
|--------------------|------------------------|--|
| <H1>titolo 1 </H1> | titolo 1 | "H" sta per "heading", cioè titolo: le grandezze previste sono sei. |
| <H2>titolo 2 </H2> | titolo 2 | Dall'<h1>, che è il più importante, si va via via degradando fino all'<h6>. |
| <H3>titolo 3 </H3> | titolo 3 | Il tag <hx> (sia esso h1 o h6) risulta formattato in grassetto e lascia una riga vuota prima e dopo di sé. |
| <H4>titolo 4 </H4> | titolo 4 | Clicca su esempio2.html |
| <H5>titolo 5 </H5> | titolo 5 | |
| <H6>titolo 6 </H6> | titolo 6 | |

17

I PARAGRAFI: <P>

Il tag <P> crea nuovi paragrafi ciascuno con il suo allineamento. Il **paragrafo** è l'unità di base entro cui suddividere un testo. Il tag <P> lascia una riga vuota prima della sua apertura e dopo la sua chiusura.

× Attributo **align**: serve per impostare l'allineamento

```
<P> testo </P>
<P align="left"> testo </P>
<P align="right"> testo </P>
<P align="center"> testo </P>
<P align="justify"> testo </P>
```

Esempio: clicca su [esempio3.html](#)

Anche in questo caso l'attributo "align" è **disapprovato dal W3C**. Meglio ricorrere ai fogli di stile.

18

I PARAGRAFI:

Il tag
 consente il ritorno a capo (interruzione di riga). Non crea un nuovo paragrafo.

Non ha attributi e non necessita di chiusura

19

LINEE ORIZZONTALI: <HR>

Il tag <HR> "horizontal Rule" serve per tracciare una riga orizzontale. Utile per separare il testo. Ecco in azione

Presenta numerosi attributi

- × "noshade" evita di sfumare la linea
- × "size" indica l'altezza in pixel
- × "width" è la larghezza in pixel o in percentuale
- × "align" l'allineamento
- × "color" per impostare il colore

```
<HR noshade size="5" width="50%" align="center" color="red">
```

20

STILI FISICI

Vengono definiti come **fisici** quei tag che definiscono graficamente lo stile del carattere, indipendentemente dalla funzione del contenuto del tag

| Codice HTML | Descrizione |
|------------------------|--|
| testo | Grassetto |
| <I>testo</I> | Corsivo |
| <PRE>testo</PRE> | Visualizza il testo così come è stato inserito nel file html |
| <U>testo</U> | Sottolineato |
| <STRIKE>testo</STRIKE> | Testo barrato. |
| ^{testo} | "Superscript" per la posizione apice |
| _{testo} | "Subscript": per la posizione pedice |

Esempio: clicca su [esempio3.html](#)

21

STILI LOGICI

Vengono definiti come **logici** quei tag che forniscono anche informazioni sul contenuto e la loro formattazione è spesso lasciata al browser con risultati a volte deludenti. Proprio per questo gli stili logici sono entrati in disuso e sono poco usati. Alcuni di questi non producono effetto visivo

| Codice HTML | Descrizione |
|------------------------------------|--------------------------------|
| <ABBR>abbreviazione</ABBR> | Indica una abbreviazione |
| <ADDRESS>indirizzo</ADDRESS> | Per indirizzi. Effetto corsivo |
| <BLOCKQUOTE>citazione</BLOCKQUOTE> | Per citazioni. Effetto rientro |
| <CITE>citazione</CITE> | Citazioni brevi. Corsivo |
| enfasi | Per enfaticizzare. Corsivo |
| rafforzamento | Testo rafforzato. Grassetto |

22

FORMATO TESTO:

Il tag permette la scelta del colore, delle dimensioni e del tipo di carattere del testo. È un **tag obsoleto**, perché la formattazione del testo in tutti i siti moderni viene attribuita attraverso i fogli di stile. In ogni caso si tratta di un argomento che un buon webmaster non può ignorare. Presenta alcuni attributi fondamentali.

× "face" per impostare il carattere

```
<FONT face="Arial">testo</FONT>
```

È bene tener conto di due accorgimenti:

- **scegliere caratteri "sicuri"**, che siano cioè senz'altro presenti sul pc dell'utente
- **non indicare un solo carattere**, ma una serie di caratteri che gradualmente si allontanano dal risultato che vorremmo ottenere, ma non di molto. In questo modo il browser dell'utente cercherà di trovare nella propria cartella dei fonts il primo carattere indicato, se non lo trova passerà al secondo, e solo come ultima spiaggia sceglierà di utilizzare il carattere predefinito (il famigerato "Times")

23

FORMATO TESTO:

Esempi:

```
<FONT face="Verdana, Arial, Helvetica, sans-serif">testo</FONT>
<FONT face="Arial, Helvetica, sans-serif"> testo </FONT>
<FONT face="Times New Roman, Times, serif"> testo </FONT>
<FONT face="Courier New, Courier, mono"> testo </FONT>
<FONT face="Georgia, Times New Roman, Times, serif"> testo </FONT>
<FONT face="Geneva, Arial, Helvetica, sans-serif"> testo </FONT>
```

24

FORMATO TESTO:

× "color" per impostare il colore del carattere

```
<FONT color="red">testo</FONT>
```

Come sempre è bene inserire la combinazione esadecimale al posto del nome

```
<FONT face="Verdana, Arial, Helvetica, sans-serif" color="#f3d422">testo</FONT>
```

25

FORMATO TESTO:

× "size" per impostare la dimensione del carattere

```
<FONT size="3">testo</font>
```

Ci sono due modi per impostare la dimensione:

- valori interi da 1 a 7
- valori relativi alla dimensione di base del tag font (di default "3")

```
<FONT size="1">testo</FONT> testo
<FONT size="2">testo</FONT> testo
<FONT size="3">testo</FONT> testo
<FONT size="4">testo</FONT> testo
<FONT size="5">testo</FONT> testo
<FONT size="6">testo</FONT> testo
<FONT size="7">testo</FONT> testo
```

26

FORMATO TESTO:

Nel caso dei valori relativi alla dimensione di base è possibile "spostarsi" nella scala di grandezza del utilizzando "+" e "-". Nell'esempio seguente la misura sarà un +2 rispetto a quella base

```
<FONT size="+2">testo</FONT>
```

La grandezza base può essere modificata con il tag <basefont> impostato prima di un blocco di testo

```
<BASEFONT size="1">
Ho impostato la misura base
<FONT size="+2"> Testo di 2 grandezze superiore </FONT>
<BR>
<FONT size="3"> Testo di grandezza 3. </FONT>
<BR><BR>
<BASEFONT size="2">
<FONT size="+2"> Testo di 2 grandezze superiore </FONT>
<BR>
<FONT size="3"> Testo di grandezza 3.
</FONT>
```

Esempio: clicca su [esempio4.html](#)

27

ELENCHI: e

Se abbiamo la necessità di inserire un elenco di termini, possiamo utilizzare:

- Elenchi ordinati
- Elenchi non ordinati

Sintassi generale elenco ordinato

```
<OL>
  <LI>primo elemento in lista
  <LI>secondo elemento in lista
</OL>
```

Sintassi generale elenco NON ordinato

```
<UL>
  <LI>primo elemento in lista
  <LI>secondo elemento in lista
</UL>
```

28

ELENCHI: e

Lo stile dell'elenco visualizzato di default è la numerazione (1,2,3,...) per quello numerato e il pallino pieno per quello non numerato. È possibile indicare uno stile differente per mezzo dell'attributo type. Gli elenchi possono essere annidati.

Opzioni possibili per l'elenco numerato:

- type="1" numeri arabi
- type="a" alfabeto minuscolo
- type="A" alfabeto maiuscolo
- type="i" numeri romani minuscoli
- type="I" numeri romani maiuscoli

Opzioni possibili per l'elenco NON numerato:

- type="disc" pallino pieno
- type="circle" cerchio vuoto
- type="square" quadrato pieno

Esempio: clicca su [esempio5.html](#)

29

I LINK: <A>

Una delle caratteristiche che ha fatto la fortuna del web è l'essere costituito non da testi ma da ipertesti. I link sono "il ponte" che consente di passare da un testo all'altro. In quanto tali, i link sono formati da due componenti:

- il collegamento (non importa se si tratta di testo o di immagine)
- la risorsa verso cui il collegamento punta (destinatario)

Sono possibili quattro tipologie di collegamento:

1. collegamento a un altro sito web
2. collegamento a un'altra pagina (file html) del medesimo sito
3. collegamento a un punto interno alla pagina (link interno)
4. collegamento a indirizzo e-mail

Il tag utilizzato è <A> unitamente all'attributo href e la sintassi è la seguente:

```
<A href="risorsa collegata">testo</A>
```

30

I LINK: <A>

1. collegamento a un altro sito web

Ecco la sintassi per creare un link con riferimento a un sito web

```
Vuoi diventare un esperto di web? Allora visita il sito <A href="http://www.html.it/">HTML</A> e imparerai i trucchi del mestiere
```

Ecco il risultato

Vuoi diventare un esperto di web? Allora visita il sito [HTML](#) e imparerai i trucchi del mestiere

È indifferente che la destinazione sia una pagina HTML di un sito, un'immagine, un file pdf, un file zip, o un file exe: il meccanismo del link funziona allo stesso modo indipendentemente dal tipo di risorsa; poi il browser si comporterà in modo differente a seconda della risorsa.

31

I LINK: <A>

2. collegamento a un'altra pagina (file html) del medesimo sito

Ecco la sintassi per creare un link a una pagina

```
Per avere ulteriori informazioni visita la sezione <A href="prodotti.html">PRODOTTI</A>
```

Ecco il risultato

Per avere ulteriori informazioni visita la sezione [PRODOTTI](#)

32

ATTENZIONE AI PERCORSI

Quando le pagine di un sito web sono numerose, è buona norma disporre i file in più cartelle seguendo una suddivisione logica. Si viene a creare così una struttura ad albero (**albero del sito**).

In conseguenza a ciò è importante prestare massima attenzione ai percorsi.

1. Per far riferimento a un file che si trovi all'interno della stessa cartella basta linkare il nome del file
2. Per far riferimento a un file contenuto in una cartella di livello inferiore alla posizione corrente, basta nominare la cartella seguita dallo "slash", e poi il nome del file. Secondo la formula **Cartella/nomeFile.html**
3. Per tornare su di un livello, è sufficiente utilizzare la notazione: **../nomeFile.html**

ATTENZIONE ANCHE AI NOMI DEI FILE

33

3. collegamento interno

Per effettuare un collegamento interno bisogna ricorrere alle "ancore" ovvero definire i punti di "arrivo" nella pagina. Ciascuna ancora ha un nome. Le ancore si definiscono con la sintassi

```
<A name="primo">testo...
```

E successivamente si imposta il collegamento nel seguente modo

```
<A href="#primo">primo paragrafo</A>
```

34

4. collegamento a una mail

La sintassi non cambia di molto basta anteporre la parola **mailto**:

```
<A href="mailto:tuaMail@TuoSito.it">Scrivimi</A>.
```

Esempio: clicca su [esempio6.html](#)

35

Alcune proprietà del tag <A>:

- × **"title"** per visualizzare un commento mentre il mouse è sopra il link
- × **"target"** per indicare dove aprire la nuova pagina: nella stessa finestra o in una nuova

```
<A href="prodotti.html" target="_new"
  title="Sezione prodotti">Visita i prodotti</A>.
```

36

Nelle pagine web possiamo inserire immagini: diagrammi e grafici, fotografie.

I formati ammessi nel Web sono sostanzialmente tre:

- **GIF (Graphic Interchange Format)**. Le GIF sono immagini con non più di 256 colori (dunque con colori piatti e senza sfumature), come grafici o icone
- **JPG**: è l'acronimo del gruppo di ricerca che ha ideato questo formato (il **Joint Photographic Experts Group**), idoneo per le immagini di qualità fotografica
- **PNG (Portable Network Graphic)**. Il PNG è un tipo di immagine introdotto abbastanza di recente, elaborato dal **W3C** per risolvere i problemi di copyright del formato GIF (che è appunto proprietario); tuttavia oggi il PNG è letto oramai da tutti i browser e offre alcune caratteristiche che gli altri formati non hanno (come il supporto al **canale alfa**, caratteristica questa non ancora perfettamente supportata da ogni browser).

37

Il codice HTML fornisce delle indicazioni al browser su come visualizzare il testo e le immagini all'interno della pagina.

Le immagini sono caricate insieme alla pagina. **Attenzione dunque a non inserire immagini troppo pesanti** (ricordatevi di ottimizzare sempre i file).

Evitare inoltre di sovraccaricare la pagina con troppe immagini (modem analogico da 56 Kbs SIGH!).

Per ottenere un sito web dalla grafica accattivante, spesso è sufficiente giocare con i colori dello sfondo e delle scritte.

La sintassi per inserire un'immagine è:

```
<IMG src="MiaImmagine.gif">
```

Dove "src" significa "source". Il tag non va chiuso

38

Anche in questo caso occorre prestare attenzione ai percorsi
Ecco alcuni esempi in cui l'immagine si trova nella cartella **"immagini"**:

```
<IMG src="immagini/logo.gif">
<IMG src="../immagini/logo.gif">
```



39

Attributi di

- × **"width"** e **"height"** per impostare larghezza e altezza dell'immagine
- × **"alt"** per specificare un testo alternativo (utile per gli ipovedenti)
- × **"title"** per un commento quando il mouse è sopra l'immagine
- × **"longdesc"** per specificare un file con una spiegazione estesa dell'immagine
- × **"border"** per inserire un bordo attorno all'immagine

```
<IMG src="logo.gif" alt="HTML.it" title="HTML.it"
  longdesc="descriz.html" width="224" height="69"
  Border="3">
```

40

LE IMMAGINI:

Anche le immagini possono diventare collegamenti ipertestuali

```
<A href="http://www.html.it">
<IMG src="logo.gif" alt="HTML.it" title="HTML.it"
  longdesc="descriz.html" width="224" height="69"
  border="3">
</A>
```

Si noti che i link lasciano sempre un bordo attorno all'immagine. Se non lo si vuole è necessario impostare a "0" la proprietà **border**

41

LE IMMAGINI:

× **"align"** per posizionare l'immagine e il testo che segue

Con **align** si possono utilizzare i seguenti valori:

- **"left"** o **"right"** per spostare l'immagine a sinistra o a destra
- **"bottom"** per allineare la prima riga di testo sulla sinistra nella parte bassa dell'immagine (è così di default)
- **"middle"** per allineare la prima riga di testo sulla sinistra al centro dell'immagine
- **"top"** per allineare la prima riga di testo sulla sinistra nel lato superiore dell'immagine

× **"hspace"** e **"vspace"** per impostare lo spazio (in pixel) che deve essere lasciato tra l'immagine e ciò che la circonda

Esempio: clicca su [esempio7.html](#)

42

LE TABELLE: <TABLE>

Le tabelle sono una delle parti più importanti di tutto il codice HTML: nate sin dagli inizi del Web per impaginare dati aggregati, si sono poi trasformate in uno strumento indispensabile per gestire i layout grafici della pagina.

Una tabella è formata da RIGHE e COLONNE. I tag necessari per creare una tabella sono:

- **<TABLE>** che "apre" una tabella
- **<TR>** "Table Row" che indica l'apertura di una riga
- **<TD>** "Table Data" che indica una cella della riga

43

LE TABELLE: <TABLE>

Ecco la sintassi per creare una tabella con 2 righe e 3 colonne:

```
<TABLE>
  <TR>
    <TD>prima cella</TD>
    <TD>seconda cella</TD>
    <TD>terza cella</TD>
  </TR>
  <TR>
    <TD>quarta cella</TD>
    <TD>quinta cella</TD>
    <TD>sesta cella</TD>
  </TR>
</TABLE>
```

Con il seguente risultato

| | | |
|--------------|---------------|-------------|
| prima cella | seconda cella | terza cella |
| quarta cella | quinta cella | sesta cella |

44

LE TABELLE: <TABLE>

Alcuni attributi riferibili a tutti e tre i tag **<TABLE>** **<TR>** **<TD>**

- × **"border"** per specificare la dimensione del bordo
- × **"width"** per indicare la larghezza in pixel o in percentuale
- × **"height"** per indicare l'altezza in pixel o in percentuale
- × **"align"** può essere riferito sia a **<TR>**, sia a **<TD>** e definisce l'allineamento dei contenuti a sinistra, a destra o al centro (**left**, **right**, **center**)
- × **"bgcolor"** può essere riferito tanto a **<TABLE>**, **<TR>**, o **<TD>** e consente di impostare un colore di sfondo

Di solito la larghezza e l'altezza globali della tabella sono espresse nel tag **<table>**, mentre la larghezza delle varie celle viene espressa nei **<td>** della prima riga. L'altezza in percentuale non sempre è visualizzata correttamente da tutti i browser.

45

LE TABELLE: <TABLE>

Ad esempio:

```
<TABLE width="75%" border="1">
  <TR>
    <TD width="25%">prima cella</TD>
    <TD width="75%" align="right">seconda cella</TD>
  </TR>
  <TR>
    <TD width="25%">terza cella</TD>
    <TD width="75%" align="right">quarta cella</TD>
  </TR>
</TABLE>
```

Produce il seguente risultato

| | |
|-------------|---------------|
| prima cella | seconda cella |
| terza cella | quarta cella |

75% della pagina web

25% della tabella 75% della tabella

46

LE TABELLE: <TABLE>

Le tabelle vanno immaginate come delle griglie, tutto sommato abbastanza rigide dunque:

- l'eventuale larghezza specificata nelle celle della prima riga avrà effetto anche sulle celle delle righe sottostanti
- non è possibile variare arbitrariamente le dimensioni delle celle, le misure specificate nelle righe sottostanti non avranno infatti effetto
- le dimensioni espresse non devono essere in contraddizione ma mano che si procede verso l'interno della tabella
- la visualizzazione dei layout con indicazioni non corrette è a discrezione del browser, quindi si rischia di ottenere risultati diversi da quelli voluti.

47

LE TABELLE: <TABLE>

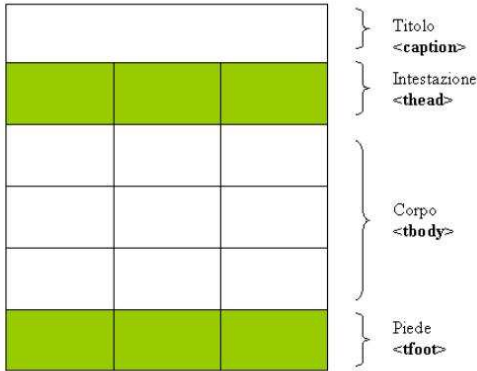
La struttura delle tabelle ha letteralmente invaso le pagine HTML, che si sono riempite di **<TR>** e di **<TD>**. Per portare un po' di ordine nelle specifiche sono state introdotti dei **tag (opzionali)** che consentono di capire facilmente quali siano le diverse parti della tabella.

| | |
|------------------------|---|
| <CAPTION> | è l'intestazione, il titolo con un commento esplicativo sulla tabella |
| <THEAD> | è la testa, parte iniziale della tabella, contiene ad esempio indicazioni sul contenuto delle celle |
| <TFOOT> | è il piede, la conclusione della tabella, consente ad esempio di tirare le somme |
| <TBODY> | è il corpo, la parte centrale con il contenuto vero e proprio della tabella |

<THEAD>, **<TFOOT>**, **<TBODY>** sono tag che consentono di individuare gruppi di righe ("row group").

48

Schema di una tabella

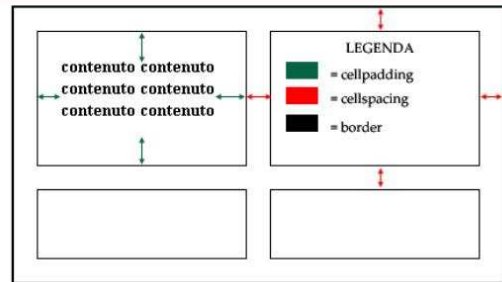


Come è possibile suddividere le righe di una tabella in gruppi ordinati, allo stesso modo è possibile raggruppare gli stili delle colonne. Il tag da usare è il <COLGROUP> e serve per fornire indicazioni su come le colonne debbano essere visualizzate. Purtroppo questo tag non funziona ancora correttamente con tutti i browser.

Il tag <COLGROUP> - la cui chiusura è facoltativa - va posto subito dopo il tag <CAPTION> e prima di <THEAD>, e consente di impostare un layout unico per le colonne senza avere la necessità di specificare allineamento del testo, o il colore di sfondo per ogni singola cella. Con l'attributo "span" possiamo impostare il numero di colonne che fanno parte del gruppo.

Attributi del tag <TABLE>:

- * "border" per impostare il bordo
- * "cellspacing" specifica la distanza (in pixel) tra una cella e l'altra, oppure tra una cella e il bordo. Di default è un pixel, dunque occorrerà sempre azzerarlo esplicitamente, quando non lo si desidera
- * "cellpadding" indica la distanza tra il contenuto della cella e il bordo, il valore viene indicato in pixel o in percentuale. Di default la distanza è nulla
- * "bgcolor" può essere riferito tanto a <TABLE>, <TR>, o <TD> e consente di impostare un colore di sfondo
- * "background" per impostare una immagine di sfondo
- * "valign" per impostare l'allineamento verticale che può essere "bottom" o "middle" o "baseline"



Con questa sintassi ad esempio si imposta una tabella con bordo di 1 pixel, senza spazio tra le celle e con il contenuto che è distanziato dai bordi della cella di 10 pixel:

```
<TABLE width="75%" border="1" cellpadding="10" cellspacing="0">
```

È possibile raggruppare le celle all'interno delle colonne in modo da avere ad esempio una riga da 2 colonne e un'altra da 3. Per ottenere questo risultato è necessario specificare che una cella deve occupare il posto di 2 (o più) colonne. In questo caso si utilizza l'attributo colspan sul <TD>, specificando come valore il numero di celle che devono essere occupate. Ad esempio:

```
<TABLE width="430" border="1" bordercolor="#000000">
  <TR>
    <TD width="30%"> <BR> <BR> <BR> </TD>
    <TD width="30%">&nbsp;</td>
    <TD width="30%">&nbsp;</td>
  </TR>
  <TR>
    <TD><BR> <BR> <BR> </TD>
    <TD colspan="2">&nbsp;</TD>
  </TR>
</TABLE>
```

Produce il seguente risultato:

| | | |
|--|------------------|--|
| | | |
| | <TD colspan="2"> | |

Tramite l'attributo rowspan (da riferirsi sempre a <TD>) è invece possibile creare delle celle che occupino più di una riga. Ad esempio

```
<TABLE width="430" border="1" bordercolor="#000000">
  <TR>
    <td width="30%">&nbsp;</TD>
    <td width="30%" rowspan="2">&nbsp;</TD>
    <td width="30%">&nbsp;</TD>
  </TR>
  <TR>
    <td>&nbsp;</TD>
    <td>&nbsp;</TD>
  </TR>
</TABLE>
```

| | | |
|--|------------------|--|
| | <td rowspan="2"> | |
| | | |

Le tabelle, grazie alle loro molteplici e multiformi caratteristiche, si sono rivelate uno strumento indispensabile non solo per impaginare i dati ma soprattutto per visualizzare i layout grafici: grazie alle tabelle è infatti possibile costruire delle griglie in cui inserire i vari contenuti di un sito e per mezzo degli sfondi, dei margini è possibile riprodurre un'impostazione accattivante.

Clicca su [esempio8.html](#) per visualizzare alcuni esempi di tabelle

Clicca su [esempio9.html](#) per visualizzare il layout di una pagina realizzato con tabelle

I FRAME: <FRAMESET>

I frame ("riquadri") comparvero per la prima volta con Netscape Navigator 2: si tratta della possibilità di suddividere una medesima finestra del browser in vari riquadri indipendenti.

Vantaggi:

- non costringono a ricaricare tutta quanta la pagina, accelerando così la navigazione dell'utente all'interno di un sito web. D'altro canto il fatto che solo una parte del contenuto sia ricaricata fa risparmiare banda anche dal punto di vista del server che deve erogare le pagine
- consente di non ripetere le parti comuni nelle varie pagine di un sito
- Il fatto di poter mantenere fisso su un lato del monitor il menu di navigazione e far scorrere sull'altro lato il contenuto piace a molti utenti, soprattutto quando la risoluzione del monitor è bassa.

57

I FRAME: <FRAMESET>

Per utilizzare i frame, è necessario creare una pagina che contenga la dichiarazione della struttura che vogliamo utilizzare. Vediamo subito il codice (si noti la mancanza di <body>):

```
<HTML>
<HEAD>
...
</HEAD>
<FRAMESET rows="50%,50%" cols="50%, 50%">
  <FRAME src="prima.html">
  <FRAME src="seconda.html">
  <FRAME src="terza.html">
  <FRAME src="quarta.html">
</FRAMESET>
</HTML>
```

Clicca su [esempio10.html](#) per visualizzare il layout di una pagina realizzato con i frame

58

I FRAME: <FRAMESET>

Scompare il tag <body> e al suo posto troviamo il tag <frameset> ("set di riquadri"), che ci permette di indicare come devono essere indicati i frames all'interno della pagina.

Il tag <frameset> ha sostanzialmente due importanti attributi:

- * "rows" per specificare il numero e la grandezza delle righe, se omesso, la struttura è a colonne
- * "cols" per specificare il numero e la grandezza delle colonne e, se omesso, la struttura è a righe

La dimensione dei frame può essere "fissa", "percentuale" o "proporzionale". È possibile lasciare che una o più righe si auto-dimensionino, occupando tutto lo spazio che rimane, in questo caso utilizzeremo l'asterisco ("wild card");

59

I FRAME: <FRAMESET>

Esempi di dimensionamento per un frameset di 2 righe e 3 colonne

Dimensione **fissa**

```
<FRAMESET rows="150,*" cols="100,200,*">
```

Dimensione **percentuale**

```
<FRAMESET rows="20%,80%" cols="25%,25%,50%">
```

Dimensione **proporzionale**

```
<FRAMESET rows="1*,3*" cols="3*,2*,1*">
```

Nell'ultimo esempio: l'altezza delle righe viene suddivisa in 4 parti (1+3); la prima riga ne occupa 1 parte e la seconda 3; la larghezza delle colonne viene suddivisa in 6 parti (3+2+1); la prima colonna occupa 3 parti, la seconda 2 e la terza 1. Vedi [esempio11.html](#)

60

I FRAME: <FRAMESET>

Una volta creata la nostra griglia con il tag <frameset>, dobbiamo specificare dove si trova il file di origine di ciascun frame mediante il tag <frame> e l'attributo src.

```
<FRAME src="prima.html">
```

L'origine di ciascun frame è un documento HTML standard. Se le dimensioni del riquadro non sono sufficienti a mostrare il documento nella sua interezza, il frame avrà delle barre di scorrimento, a meno che non sia stato esplicitamente specificato il contrario negli attributi.

È importante assegnare un nome al frame:

```
<FRAME id="frame1" name="frame1" src="prima.html">
```

La sintassi corretta è id="frame1" tuttavia per questioni di compatibilità (con Netscape 4) è oramai entrato nell'uso utilizzare anche name="frame1".

61

I FRAME: <FRAMESET>

È possibile costruire frames annidati. Vedi [esempio12.html](#)

```
<FRAMESET rows="15%,70%,15%">
  <FRAME src="prima.html">
  <FRAMESET cols="25%,50%,25%">
    <FRAME src="seconda.html">
    <FRAME src="terza.html">
    <FRAME src="quarta.html">
  </FRAMESET>
  <FRAME src="quinta.html">
</FRAMESET>
```

62

I FRAME: <FRAMESET>

In una struttura a frames può apparire difficile caricare il contenuto di un link nella sezione appropriata. Grazie all'attributo **target** del tag **<a>** possiamo specificare la destinazione del link e caricare il contenuto di un collegamento nel riquadro che riteniamo più opportuno:

```
<A href="nomeFile.html" target="nomeFrame">...
```

Altri valori di **target**:

- **target="_blank"** apre il link in una nuova finestra
- **target="_self"** apre il link nel frame stesso
- **target="_parent"** il documento viene caricato nel frameset precedente a quello corrente (più esattamente nel frame genitore), utile in strutture annidate
- **target="_top"** il documento viene caricato nella finestra originale, cancellando ogni struttura a frame

63

I FRAME: <FRAMESET>

Il tag **<noframes>**

I vecchi browser possono non essere compatibili con i frames. In previsione di situazioni di questo genere è possibile inserire nel tag **<noframes>** un contenuto appositamente studiato per questa eventualità, ad esempio l'avvertimento che il sito è strutturato a frame, o un contenuto alternativo, oppure una descrizione per i motori di ricerca.

```
<FRAMESET rows="50%,50%">
  <FRAME src="prima.html">
  <FRAME src="seconda.html">
  <NOFRAMES>
    <P>Qui può essere indicato un link a
    <A href="senzaFrame.html">una versione</A>
    della pagina che non usa i frame</P>
  <NOFRAMES>
</FRAMESET>
```

64

I FRAME: <IFRAME>

"IFRAME" significa "inline frame": in qualsiasi momento in un documento che non utilizzi una struttura a frame è possibile creare un frame al volo grazie a questo tag. Possiamo specificare la larghezza e l'altezza del tag, mentre gli attributi di visualizzazione sono gli stessi del tag **<frame>**: si tratta di una vera e propria finestra verso l'esterno all'interno di un documento ordinario. Questo tag è correttamente supportato da tutti i browser moderni (Netscape 4 non lo supporta, ma questo browser oramai sta scomparendo). Vedi esempio13.html. La sintassi è:

```
<IFRAME src="esempio1.html" width="400"
height="300">
Segue testo alternativo per i browser che non
supportano iframe
</IFRAME>
```

65

I FRAME: <IFRAME>

Considerazioni finali

Molti dei motivi che rendevano vantaggioso l'utilizzo dei frames sono venuti meno: la banda a disposizione si è ampliata, i CSS alleggeriscono la struttura dei siti e rendono possibile alcune soluzioni che prima erano difficili (come quella di mantenere un menu di navigazione sempre a portata di mano), e **la gestione dei contenuti può essere semplificata utilizzando le inclusioni lato server**.

Viceversa una struttura a frame risulta molto vantaggiosa nel caso in cui si utilizzino delle vere e proprie applicazioni che utilizzano internet (come le piattaforme di e-learning, la webmail, eventuali aree riservate del sito con accesso tramite login e password). In questo caso la suddivisione dei contenuti evita di sovraccaricare il server (dal momento che così vengono ri-caricati solo le parti strettamente necessari), semplifica la gestione, e quindi si rivela estremamente utile

66