

## Notiuni de HTML

---

### Ce este HTML

HTML sau HyperText Markup Language este, așa cum o sugerează și definiția, un limbaj de marcare pentru *hypertext*.

Termenul *Hipertext* se referă la modalitatea în care se fac legăturile între diferite documente HTML. În fapt, *hipertext* trebuie înțeles ca un text mai profund decât decât un text normal. De exemplu, o pagină de *hypertext* nu este doar o simplă pagină de text. Ea este o pagină care, pe lângă textul propriu-zis, mai conține și niste legături (*hiperlegături*) către alte texte care abordează același subiect ca și cel tratat în pagina de unde se fac legăturile (hiperlegăturile sunt acele texte subliniate sau nu, de cele mai multe ori scrise cu albastru și, la poziționarea cu mouse-ul asupra lor, cursorul devine activ).

Astfel se poate constata că, pentru un subiect oarecare, se dispune de o documentație impresionantă ce poate fi accesată de la o simplă pagină de hipertext (prin intermediul hiperlegăturilor). Datorită acestor trimiteri către alte texte se poate constata că textul inițial are o profunzime mult mai mare, decât ar avea un text normal care este limitat doar la ceea ce se vede. În esență, hypertextul trebuie înțeles ca un text cu legături către alte texte.

Pentru a vedea cum funcționează acest limbaj de marcare, se presupune că a fost scris un document cu un editor de texte precum Word, Wordpad sau orice alt editor mai performant. Ca orice document, el va conține mai multe elemente cum ar fi: un titlu, un paragraf sau poate chiar un tabel. Poziționarea în pagină a textului se realizează cu ajutorul tastaturii. Un browser de internet (Internet Explorer, Netscape navigator, etc) nu poate citi decât fișiere HTML, adică acele fișiere cu extensia .htm sau .html. Un asemenea browser nu poate afișa corect elementele unui text, adică nu poate afișa un titlu sau un paragraf, decât dacă i se sugerează acest lucru cu ajutorul unor *comenzi*.

În concluzie, un fișier HTML va conține pe lângă textul care se dorește a fi afișat în fereastra browser-ului și niste elemente (tag-uri) care marchează textul respectiv, sugerând în acest fel browser-ului care este titlul documentului, care este paragraful sau care este tabelul.

### Structura paginii HTML

Paginile pe internet sunt, de regula, realizate într-un limbaj numit **HTML** - HyperText Mark-up Language. Editarea acestora se poate face în orice program de editare texte, pornind chiar cu banalul **Notepad** din Windows. Toate notiunile prezentate în acest ghid pot fi puse în practică utilizând Notepad.

Fișierul care se editează se va numi **index.html**. În general fișierul principal al unui site internet (cel care se încarcă atunci când tastăm o adresă de genul www.abc.abc) este numit index.html. Instrucțiunile folosite într-o pagină HTML se numesc **tag-uri** și acestea indică browser-ului (Netscape, Internet Explorer, etc) modul de prezentare a informației din pagină. Tag-urile se scriu între semnele < și >. Tag-urile nu trebuie scrise respectând reguli foarte stricte (nu e nevoie să scrii numai câte un tag pe rând sau să folosești numai litere mari, etc). Un exemplu de utilizare a tag-ului FONT este:

```
<FONT FACE=ARIAL COLOR=#FF0000>EED – Facultatea de Inginerie Electrica</FONT>
```

Instrucțiunea de mai sus va determina afișarea textului "EED – Facultatea de Inginerie Electrica" cu font arial și culoarea roșie. După cum se poate observa, există un tag **înaintea** textului și unul care **încheie** textul respectiv. **Majoritatea tag-urilor funcționează în acest mod: se pun înaintea informației pe care dorim să o prezentăm într-un anumit mod, iar apoi la sfârșitul acesteia**, ca în exemplul de mai sus.



Pe lângă această entitate, mai există și altele. *Entitățile*, constituie o metodă de reprezentare a caracterelor care nu ar putea fi redată corect în cadrul unui document HTML dacă ar fi scrise ca atare, direct de la tastatură, sau care nu sunt disponibile la tastatură. O entitate va fi scrisă sub forma &nume entitate;. Mai jos sunt exemplificate câteva entități precum și caracterele aferente.

- &lt; ( < ) - semnul mai mic decât
- &gt; ( > ) - semnul mai mare decât
- &copy; ( © ) - semnul pentru copyright
- &amp; ( & ) - semnul ampersand
- &beta; ( β ) - litera beta din alfabetul grecesc
- &lt; ( < ) - semnul mai mic decât
- &gt; ( > ) - semnul mai mare decât
- &amp; ( & ) - semnul ampersand

**Trecerea la un nou rând** nu se face apăsând Enter în fișierul HTML, ci utilizând tag-ul **<BR>** (acest tag se utilizează singur; nu se folosește cu **</BR>**). Astfel, pentru a scrie "EED" pe un rând, a lăsa un rând liber și "USV" pe un rând separat, se va folosi:

```
EED<BR><BR>USV
```

S-a utilizat mai sus de două ori tag-ul **<BR>**: o dată pentru a trece la un nou rând și o dată pentru a lăsa un rând liber. Tag-ul utilizat pentru formatarea modului de afișare a textului este **<FONT>...</FONT>**. Tag-ul FONT suportă mai mulți parametri, printre care:

- SIZE=x, unde x este dimensiunea fontului (1 fiind dimensiunea cea mai mică);
- COLOR=#RRGGBB - culoarea cu care se va afișa fontul. Culoarea implicită este neagră;
- FACE=font - fontul cu care va fi scris textul respectiv. Atenție! Vizitatorii paginii trebuie să aibă fontul respectiv instalat pe calculator, deci este indicat să se utilizeze numai fonturi "clasice". Fontul implicit este Times New Roman. Acești parametri pot fi utilizați separat sau simultan în cadrul tag-ului FONT. Astfel, instrucțiunea de mai jos afișează text folosind culoarea albastru:

```
<FONT COLOR=#0000AA>EED</FONT>
```

Instrucțiunea de mai jos afișează același text cu font arial, culoare albastră și dimensiune mai mare:

```
<FONT COLOR=#0000AA SIZE=4 FONT=ARIAL>EED</FONT>
```

După cum se poate vedea, se pot utiliza mai mulți parametri în cadrul aceluiași tag FONT. Aceștia nu se despart prin virgulă, ci printr-un spațiu. De reținut că parametrul FACE poate primi mai multe fonturi. Astfel, un tag de tipul

```
<FONT FACE=VERDANA, ARIAL>...</FONT>
```

va instrui browserul să utilizeze fontul Verdana, iar dacă acesta nu este găsit, să utilizeze Arial. Mai există trei tag-uri utile pentru formatarea textelor: **<B>...</B>** - utilizat pentru scrierea textelor cu **bold**, **<I>...</I>** - utilizat pentru scrierea textelor cu *italics* (litere înclinate) și **<U>...</U>** - pentru a scrie text subliniat. Desigur, pot exista combinații între tag-ul FONT și tag-urile de subliniere, îngroșare sau înclinare, ca de exemplu:

<FONT COLOR=#00AA00><B>Text verde ingrosat</B></FONT>

Trebuie tinut cont ca tag-urile trebuie inchise in ordine inversă față de cum au fost deschise.

In fisierul index.html deschis in Notepad, se va sterge randul cu "...aici se completeaza informatia de prezentat in pagina..." si se inlocuieste cu:

<BR><FONT SIZE=4 COLOR=#0000AA><B>EED</B></FONT><BR>

Ca și în cazul editoarelor de texte, este posibilă introducerea unor titluri (heading-uri). Pentru aceasta HTML dispune de elementul H (heading). Forma generală este:

<Hnr> TITLU </Hnr>

unde **nr** este un număr ce poate avea valori de la 1 la 6 în functie de importanta care o are titlul respectiv în cadrul documentului. H1 va duce la realizarea de titluri foarte evidente, iar H6 va reprezenta forma cea mai puțin evidentă. Mai jos sunt redate modurile de afisare pentru valorile 1 și 2.

<H1>Titlu realizat cu H1</H1>

<H2>Titlu realizat cu H2</H2>

Textul care se dorește a fi introdus în pagina HTML poate fi organizat pe paragrafe (ca și la editoarele de texte). În cazul unui browser acest lucru se specifică prin elementul <P> in cadrul fisierului HTML. Browserul va afisa ca un paragraf tot textul cuprins între tag-urile <P> si </P>. În cazul în care este omisă utilizarea acestor tag-uri întregul text cuprins de tag-urile elementului BODY va fi tratat ca un paragraf unic.

Paragrafele pot fi aliniate în diverse moduri, prin utilizarea atributului ALIGN. De exemplu,

<P ALIGN="RIGHT">Acesta este un paragraf aliniat la dreapta</P>

Paragraful poate avea și ALIGN="JUSTIFY", care va avea drept rezultat alinierea textului la ambele margini. Prin alinierea unui paragraf la ambele margini apare un inconvenient de ordin estetic și anume faptul că, prin această fortare, spațiul dintre cuvinte nu va mai fi același, spațiul variind în functie de modul în care se va umple rândul respectiv.

Câteva observații referitoare la utilizarea paragrafelor:

- Utilizarea repetată a paragrafelor goale (adică o succesiune de elemente sub forma <P><P><P>) în scopul realizării unui spațiu mai mare între două paragrafe nu constituie o soluție corectă. Chiar dacă un browser poate interpreta corect acest lucru există browsere care nu citesc decât un singur paragraf gol, urmînd ca restul paragrafelor goale să fie ignorate.
- Între două paragrafe va rămîne un rând liber. Distanța dintre paragrafe poate fi modificată dar acest lucru nu poate fi realizat prin intermediul elementelor existente în HTML. Dacă dorim să modificăm spațiul dintre paragrafe va trebui să utilizăm CSS. Pînă la apariția CSS, distanța dintre două paragrafe putea fi modificată cu ajutorul elementului BR. Această metodă dă rezultate bune dar permite un control foarte riguros asupra spațiului dintre paragrafe.
- Chiar dacă utilizarea tag-ului de sfîrsit (</P>) nu este obligatorie este indicat să nu fie omis. Lipsa acestui tag va face ca elementul imediat următor să nu mai fie poziționat la un rând distanță față de paragraf. Această regulă este bine să fie aplicată tuturor elementelor HTML. În plus, utilizarea ambelor tag-uri usurează munca atunci cînd este nevoie să se facă modificări în documentul inițial. În acest mod citirea textului sursă se face mai ușor.

- Atributul ALIGN sugerează alinierea pe orizontală. Acest atribut precum și valorile sale sunt identice pentru mai multe elemente din cadrul HTML.

Alte formătări de text:

<B>	Acest text este <b>Bold</b>
<I>	Acest text este <i>Italic (cursiv)</i>
<TT>	Acest text este TeLeType
<U>	Acest text este <u>Underline</u>
<STRIKE>	Acest text este <del>STRIKE</del>
<BIG>	Acest text este Big
<SMALL>	Acest text este Small
<SUB>	Acest text este <sup>SUBscript</sup>
<SUP>	Acest text este <sup>SUPrascript</sup>

Textul mai poate fi prezentat și sub forma unor liste. Acestea pot fi de două feluri: ordonate (numerotate) sau neordonate. Tag-urile folosite pentru acestea sunt <OL></OL>, respectiv <UL></UL>. Elementele listei sunt introduse prin intermediul tag/ului <LI>. Exemple:

<UL>		<OL>
<LI>Elementul 1		<LI>Text 1
<LI>Elementul 2		<LI>Text 2
<LI>Elementul 3		<LI>Text 3
</UL>		</OL>

Se poate observa că folosirea tag-ului </LI> nu este obligatorie.

### Exemplu

```
<html>
<head><title>Exemplu liste 2</title></head>
<body><h1 align=center>O lista neordonata</h1><hr>
Glosar de termeni de World Wide Web
<ul>Culori uzuale disponibile prin nume
<li>Black <li>White <li>Red <li>Green <li>Blue <li>Yellow <li>Purple <li>Aqua
</ul>
</body>
</html>
```

Listele neordonate pot fi imbricate pe mai multe niveluri

```
<html>
<head><title>Exemplu liste 3</title></head>
<body><h1 align=center>O lista neordonata de liste neordonate</h1><hr>
Glosar de termeni de World Wide Web
<ul>Elemente și atribute a unei pagini HTML
<li>Frameset <ul>Atribute: <li>cols <li>rows <li>border           </ul>
<li>Frame <ul>Atribute: <li>src <li>name <li>scrolling       </ul>
```

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

O lista ordonata de elemente este un bloc de text delimitat de etichetele corespondente `<ol>...</ol>` ("ol" vine de la "ordered list" = lista ordonata). Fiecare element al listei este initiat de eticheta `<li>` (list item). Lista va fi indentata fata de restul paginii Web si fiecare element al listei incepe pe un rand noua.

```
<html>
<head><title>Exemplu liste 4</title></head>
<body><h1 align=center>O lista ordonata</h1><hr>
<ol>Culori uzuale disponibile prin nume
<li>Black <li>White <li>Red <li>Green <li>Blue <li>Yellow <li>Purple <li>Aqua
</ol>
</body>
</html>
```

Tag-urile `<ol>` si `<li>` pot avea un atribut `type` care stabileste tipul de caractere utilizate pentru ordonarea listei. Valorile posibile sunt:

- " A " pentru ordonare de tipul A , B , C , D etc. ( litere mari );
- " a " pentru ordonare de tipul a , b , c , d etc. ( litere mici );
- " I " pentru ordonare de tipul I , II , III , IV etc. ( cifre romane mari );
- " i " pentru ordonare de tipul i , ii , iii , iv etc. ( cifre romane mici );
- " 1 " pentru ordonare de tipul 1 , 2 , 3 , 4 etc. ( cifre arabe - optiune prestabilita );

Urmatorul exemplu este o lista ordonata cu cifre romane

```
<html>
<head><title>Exemplu liste 5</title></head>
<body><h1 align=center>O lista ordonata cu cifre romane</h1><hr>
<ol type="I">Culori uzuale disponibile prin nume
<li>Black <li>White <li>Red <li>Green <li>Blue <li>Yellow <li>Purple <li>Aqua
</ol>
</body>
</html>
```

Tag-ul `<ol>` poate avea un atribut `start` care stabileste valoarea initiala a secventei de ordonare. Valoarea acestui atribut trebuie sa fie un numar intreg pozitiv.

Urmatorul exemplu este o lista ordonata cu litere mari, incepand de la valoarea C.

```
<html>
<head><title>Exemplu liste 6</title></head>
<body><h1 align=center>O lista ordonata cu litere mari, incepand de la valoarea C</h1><hr>
<ol type="A" start="3">Culori uzuale disponibile prin nume
<li>Red <li>Green <li>Blue <li>Yellow <li>Purple <li>Aqua
</ol>
</body>
</html>
```

Tag-ul `<li>` poate avea un atribut `value` care satileste valoare pentru elementul respectiv al listei. Valoarea acestui atribut trebuie sa fie un numar intreg pozitiv (vezi urmatorul exemplu).

```
<html>
<head><title>Exemplu liste 7</title></head>
<body><h1 align=center>O lista ordonata avand itemi setati individual</h1><hr>
<ol start="3">Repetati urmatorii pasi ai algoritmului
<li>salvati fisierul;
<li value="6">incarcati fisierul in browser;
<li>schimbati browserul utilizat
<li>incarcati din nou fisierul
</ol>
</body>
</html>
```

Listele ordonate pot fi imbricate intre ele sau cu liste neordonate ,ca in exemplul urmator.

```
<html>
<head><title> Exemplu liste 8</title></head>
<body><h1 align=center>O lista ordonata de liste ordonate si neordonate</h1><hr>
<ol>Un sistem informatic include:
  <li>Hardware:
    <ol>
      <li>placa de baza
      <li>procesor
      <li>memorie
      <li>harddisk
    </ol>
  <li>Software:
    <ul>
      <li>Linux
      <li>Windows
      <li>OS/2
      <li>Unix
    </ul>
  <li>Software de aplicatie:
    <ul type="disc">
      <li>VisualC++
      <li>Java
      <li>SQL
      <li>CorelDraw
    </ul>
</ol>
</body>
</html>
```

### Utilizari speciale ale listelor

Daca intr-o lista, in loc de elementele acesteia introduse prin `<li>`, se inserează un bloc de text, aceasta va fi indentat ( întocmai ca elementele unei liste).

```

<html>
<head><title> Exemplu liste 9</title></head>
<body><h1>Un bloc de text indentat</h1><hr>
<ol> Un bloc de text indentat. Un bloc de text indentat. Un bloc de text indentat. Un bloc de
text indentat. Un bloc de text indentat. Un bloc de text indentat. Un bloc de text indentat. Un
bloc de text indentat. Un bloc de text indentat.
</ol>
</body>
</html>

```

In exemplul următor lista de definiții are itemii **<dt>** și **<dd>** multipli.

```

<html>
<head><title> Exemplu liste 10</title></head>
<body><h1 align=center>O lista de definitii speciala</h1><hr>
<dl>Program
<dt><b>Luni</b>
<dt><b>Marti</b>
<dt><b>Miercuri</b>
<dd><i>Ora 9.00.</i>Inscriere
<dd><i>Ora 11.00.</i>Audieri
<dd><i>Ora 13.30.</i>Raspunsuri
</dl>
</body>
</html>

```

Ultimul exemplu este o lista neordonată personalizată care utilizează pe post de elemente imagini și texte.

```

<html>
<head><title> Exemplu liste 11</title></head>
<body><h1>O lista personalizata</h1><hr>
<ul>Tipuri de masini Volkswagen sunt:<br>
Golf<br>
Jetta<br>
Passat<br>
Bora<br>
Corrado<br>
Transporter<br>
</body>
</html>

```

## Culori

S-a văzut în capitolul anterior că textele din paginile HTML pot avea diverse culori cu un tag de tipul

**<FONT COLOR=#00FF00>...</FONT>**.

Se va prezenta in continuare **sistemul de codificare a culorilor** pentru paginile web. Fiecare culoare este interpretata ca fiind compusa din diverse procente ale culorilor **rosu, verde si albastru** (RGB). Acestea sunt exprimate prin numere in baza 16. Astfel, numarul 00 reprezinta nivelul minim dintr-o anumita culoare iar numarul FF reprezinta nivelul maxim. Acordand ponderi exprimate prin aceste numere in baza 16 culorilor rosu, verde si albastru (Red, Green, Blue - RGB) se poate obtine orice culoare:

black = "#000000"	maroon = "#800000"
green = "#008000"	navy = "#000080"
silver = "#C0C0C0"	red = "#FF0000"
lime = "#00FF00"	blue = "#0000FF"
gray = "#808080"	purple = "#800080"
olive = "#808000"	teal = "#008080"
white = "#FFFFFF"	fuchsia = "#FF00FF"
yellow = "#FFFF00"	aqua = "#00FFFF"

Culorile de mai sus vor fi mai putin "aprinse" daca in loc de FF se foloseste AA, spre exemplu: **#00AA00**. Folosind acelasi sistem se va putea specifica culoarea de fundal (**background**) a unei pagini web, in cadrul tag-ului BODY despre care s-a vorbit la inceput:

```
<BODY BGCOLOR=#DDFFFF>
```

Tag-ul BODY mai poate contine si alte informatii interesante, ca de exemplu culoarea cu care vor fi afisate link-urile catre alte pagini (utilizand atributul LINK) si link-urile catre alte pagini care au fost vizitate deja (folosind atributul VLINK-visited link):

```
<BODY BGCOLOR=#DDFFFF LINK=#0000AA VLINK=#0000AA>
```

In fisierul index.html din Notepad, se va inlocui randul ce contine tag-ul **<BODY>** cu cel de mai sus si se va salva fisierul (se inlocuieste tag-ul BODY de la inceputul fisierului, nu cel de la sfarsit care este **</BODY>**). Fundalul paginii ar trebui sa devina albastru deschis.

## Exemple

1.

```
<html>
<head>
<title>culoare textului </title>
</head>
<body text=red>
Un text de culoare rosie.
</body>
</html>
```

2.

```
<html>
<head>
<title>Attribute multiple </title>
</head>
<body bgcolor=blue text=yellow>
Fond de culoare albastra si text de culoare galbena.
</body>
</html>
```

## Tabele

Tabelele se utilizeaza folosind tag-ul `<TABLE>...</TABLE>`. Tabelele sunt formate din căsuțe, care sunt organizate pe **randuri** (rows - linii orizontale de casute) și **coloane** (columns).

1				

Tabelul de mai sus contine 3 randuri, fiecare rand continand 5 elemente. In interiorul unui tag de tip TABLE, randurile sunt declarate folosind tag-ul `<TR>...</TR>`, iar in cadrul fiecarui tag TR elementele sunt declarate cu tag-ul `<TD>...</TD>`. Fiecare element (casuta) poate contine text, imagini, link-uri, etc. Un tabel de tipul celui de mai sus ar fi codificat astfel:

```
<TABLE>
  <TR><TD>1</TD><TD></TD><TD></TD><TD></TD><TD></TD></TR>
  <TR><TD></TD><TD></TD><TD></TD><TD></TD><TD></TD></TR>
  <TR><TD></TD><TD></TD><TD></TD><TD></TD><TD></TD></TR>
</TABLE>
```

In exemplul de mai sus exista un tag `<TABLE>...</TABLE>` care delimiteaza tabelul si trei tag-uri `<TR>...</TR>` corespunzand celor trei randuri ale tabelului. Fiecare tag TR contine cate 5 tag-uri `<TD>...</TD>`. Intre `<TD>` si `</TD>` se introduce continutul efectiv al fiecărei casute. In exemplul de mai sus numai casuta din stanga-sus are continut - cifra 1 (căsuțele fără continut nu sunt afisate).

Tag-urile de definire a tabelelor pot avea diverse atribute, pentru formatarea tabelului. Astfel, tag-ul TABLE poate avea:

- WIDTH = x sau WIDTH = x% - acesta specifica lungimea intregului tabel in pixeli (de exemplu WIDTH = 500) sau in procente fata de lungimea intregii ferestre (de exemplu WIDTH = 50%);
- HEIGHT = x sau HEIGHT = x% - similar cu WIDTH, dar pentru inaltimea intregului tabel (exemple: HEIGHT = 300 sau HEIGHT = 100%);
- BORDER= x - daca x este 1 atunci tabelul va avea si "cadru" (linii de delimitare a elementelor. Daca x este 0, atunci acestea vor lipsi (deci BORDER=0 sau BORDER=1);
- CELLSPACING = x - specifica spatiul dintre casute, in pixeli;
- CELLPADDING = x - specifica spatiul dintre marginea unei casute si continutul acesteia, in pixeli. Si in acest caz pot fi folosite toate aceste atribute impreuna, numai o parte a lor, sau nici unul. Un tag TABLE ce le foloseste pe toate ar arata astfel:

```
<TABLE WIDTH=80% HEIGHT=200 BORDER=1 CELLPADDING=0 CELLSPACING=5>
.....
</TABLE>
```

Si tag-ul TD poate avea tot felul de atribute interesante:

- WIDTH = x sau WIDTH = x% - lungimea casutei in pixeli sau ca procent din lungimea intregului tabel;
- HEIGHT = x sau HEIGHT = x% - inaltimea casutei in pixeli sau ca procent din inaltimea intregului tabel;

- BGCOLOR = #RRGGBB - culoarea fundalului casutei, ca de exemplu BGCOLOR = #EEEEDD.
- ALIGN = RIGHT, ALIGN = LEFT sau ALIGN = CENTER - modul cum va fi pozitionat continutul casutei in casuta: la dreapta, la stanga sau central;
- VALIGN = TOP, VALIGN = BOTTOM sau VALIGN = CENTER - similar, dar pentru pozitionarea pe verticala: sus, jos sau central. Tine cont de faptul ca textul din interiorul unei casute trebuie formatat si el la randul sau. O casuta ce ar contine un text rosu pe fond verde deschis s-ar codifica astfel:

```
<TD BGCOLOR = #DDFFDD><FONT COLOR = #AA0000>text</FONT></TD>
```

Există posibilitatea de a se insera într-o pagină HTML și tabele atipice, ca de exemplu:

CELULA 1	CELULA 2	CELULA 3
CELULA 4		

Un asemenea tabel s-ar codifica astfel:

```
<TABLE>
  <TR><TD>CELULA 1</TD><TD>CELULA 2</TD><TD>CELULA 3</TD></TR>
  <TR><TD COLSPAN=3>CELULA 4</TD></TR>
</TABLE>
```

Se poate remarca utilizarea atributului **COLSPAN = 3** in prima celula a celui de-al doilea rand. Acesta spune ca celula respectiva se "intinde" pe lungimea a trei celule din structura normala a tabelului. Desigur, acest al doilea rand va contine numai o singura celula, nu trei. Similar se poate proiecta un tabel cu o celula care se intinde pe verticala pe spatiul a doua celule:

CELULA 1	CELULA 2	CELULA 3
CELULA 4	CELULA 5	

Codul acestuia este:

```
<TABLE>
  <TR><TD>CELULA 1</TD><TD>CELULA 2</TD>
  <TD ROWSPAN=2>CELULA 3</TD></TR>
  <TR><TD>CELULA 4</TD><TD>CELULA 5</TD></TR>
</TABLE>
```

Se poate observa utilizarea atributului **ROWSPAN=2** in a treia celula a primului rand al tabelului. Acesta spune ca aceasta celula se intinde pe verticala pe "inaltimea" a doua randuri. Desigur, cel de-al doilea rand contine numai doua celule (nu trei), intrucat a treia vine din randul de sus.

Tabelele pot reprezenta un instrument foarte puternic in cadrul unui fisier HTML. O tehnica interesanta de pozitionare cat mai exacta a unor informatii consta in utilizarea unor tabele in tabele (tag-uri TD care au drept continut un tabel, de exemplu). In acest caz este foarte importanta succesiunea corecta a tag-urilor TD, TR si TABLE pentru o afisare corecta. O alta tehnica interesanta

este utilizarea tabelelor cu BORDER=0 (fara cadru) pentru gruparea unor date fara a da impresia existentei unui tabel propriu-zis.

## Exemple de tabele

In continuare se dau doua exemple de tabele care pot fi incarcate si in fisierul index.html deschis in Notepad:

```
<TABLE WIDTH=700 BORDER=0 CELLPADDING=5 CELLSPACING=5>
  <TR>
    <TD WIDTH=30% BGCOLOR=#006666 ALIGN=LEFT>
      <FONT FACE=VERDANA,ARIAL SIZE=2 COLOR=#FFFFFF>
      <B> Adresa: </B></FONT></TD>
    <TD BGCOLOR="#CCEEEE" ALIGN=LEFT>
      <FONT FACE=VERDANA,ARIAL SIZE=2>
      <I>Strada Universitatii, Nr. 13</I></FONT></TD>
  </TR>
  <TR>
    <TD WIDTH=30% BGCOLOR=#006666 ALIGN=LEFT>
      <FONT FACE=VERDANA,ARIAL SIZE=2 COLOR=#FFFFFF>
      <B> Descriere: </B></FONT></TD>
    <TD BGCOLOR="#CCEEEE" ALIGN=LEFT>
      <FONT FACE=VERDANA,ARIAL SIZE=2>
      <I>Aici se scrie textul de prezentare</I></FONT></TD>
  </TR>
</TABLE>
```

Exemplul de mai sus nu contine decat notiuni explicate deja in capitolele anterioare: este vorba de un tabel "clasic" cu lungime de 700 pixeli (WIDTH=700), fara cadru (BORDER=0) si cu spatiu de 5 pixeli atat intre casute (CELLSPACING=5) cat si intre cadru si informatia din casuta (CELLPADDING=5); tabelul are doua randuri (continute intre cele doua perechi de tag-uri <TR>...</TR>); fiecare rand contine cate doua casute:

- prima de lungime 30% din total (WIDTH=30%) cu culoarea fondului bleumarin (BGCOLOR=#006666) si textul aliniat la stanga. Textul din aceasta casuta este scris cu font Verdana de dimensiune 2 si culoare alba (FONT FACE=VERDANA,ARIAL SIZE=2 COLOR=#FFFFFF) si este ingrosat (B).
- a doua casuta cu fond bleu (BGCOLOR=#CCEEEE) si text aliniat de asemenea la stanga (ALIGN=LEFT). Contine text cu acelasi font Verdana de dimensiune 2, dar scris cu negru (implicit) si cu litere inclinate (I).

Un tabel, ceva mai simplu:

```
<TABLE WIDTH=700 BORDER=1 CELLPADDING=5>
  <TR>
    <TD BGCOLOR=#CCEEEE><FONT FACE=VERDANA,ARIAL SIZE=2><B> NOU
    </B>
    .....
  </TD>
  </TR>
</TABLE>
```

Acest tabel are aceeași lungime de 700 de pixeli (WIDTH=700), are și cadru (BORDER=1) și un spațiu de la cadru la textul din casute de 5 pixeli (CELLPADDING=5). Tabelul conține un singur rând și o singură casută în acest rând. Casuta are fondul bleu (BGCOLOR=#CCEEEE). În interiorul casutei este un text scris cu font Verdana de dimensiune 2. Unele cuvinte din tabel sunt scrise cu bold (B).

## Alte prelucrări

Un alt tag care poate apărea în fișierele HTML este <HR>. Acesta are un rol foarte simplu și anume de a trasa o linie orizontală pe toată lățimea ferestrei. Acest tag funcționează la fel ca și BR (adică, nu se folosește </HR>). Tag-ul <HR> primește și atributul **WIDTH** sub forma WIDTH = x sau WIDTH = x% care specifică lungimea liniei în pixeli sau ca procent din lățimea totală a ferestrei. Exemplu:

```
<HR WIDTH=600>
```

În fișierul index.html deschis în Notepad se vor introduce următoarele instrucțiuni între ultimul tag </TABLE> și tag-ul </CENTER> de la sfârșitul fișierului:

```
<BR><BR><BR><BR><BR><BR><HR>  
<FONT FACE=ARIAL SIZE=1>Toate drepturile rezervate. Pagina realizată în anul 2004</FONT>
```

Se va folosi această ocazie pentru a prezenta și modalitatea de includere a unor comentarii într-o pagină HTML. Acestea trebuie incluse între <!-- și --> ca în exemplul următor:

```
<!-- acest text nu va fi afișat în pagina -->
```

## Lucrul cu imagini

Pentru inserarea unor imagini în fișierele HTML se va utiliza tag-ul IMG. Atributele acestuia sunt următoarele:

- Atribut Semnificație Valoare
- SRC - reprezintă adresa la care se află imaginea precum și numele acesteia
- ALT - text alternativ pentru descrierea imaginii text
- WIDTH - lățimea imaginii pixeli
- HEIGHT - înălțimea imaginii pixeli
- ALIGN - poziționarea pe verticală sau orizontală a imaginii top | middle | bottom | left | right
- HSPACE - spațiul orizontal dintre imagine și textul care o înconjoară pixeli
- VSPACE - spațiul orizontal dintre imagine și text pixeli
- USEMAP - atribut care specifică adresa unde se găsește definiția unei imagini de tip hartă aflată pe partea de client
- ISMAP - specifică utilizarea unei imagini de tip hartă al cărei fișier de definiție se află pe server
- BORDER - stabilește dimensiunea marginilor care înconjoară o imagine ce este folosită drept legătură pixeli

Atributul SRC indică adresa sau locul unde se află fișierul imagine ce urmează a fi introdus în pagină precum și numele acestuia. În termeni tehnici se spune că acest atribut are drept valoare un URL (Uniform Resource Locators). Un URL trebuie înțeles ca fiind o simplă adresă de Internet. Exemple:

```
SRC="numeimagine.gif"  
SRC="imagini/numeimagine.gif"  
SRC="http://www.numesite.com/imagini/numeimagine.gif"
```

Primele două URL-uri (adrese) se mai numesc și adrese relative. Aceste adrese au drept punct de referință locul de unde se face adresarea. Mai concret, este vorba de locul unde se află pagina HTML în care se dorește a fi inserată imaginea aflată în fișierul "numeimagine.gif". În primul exemplu ea se află în același director cu pagina html. În al 2-lea exemplu, imaginea se află într-un subdirector intitulat "imagini".

Ultimul exemplu reprezintă o adresă absolută, adică o adresă de internet completă unde se regăsește un anumit fișier. Spre deosebire de adresele relative care se modifică odată cu modificarea punctului de referință (în sensul că fișiere HTML se pot afla și în alte directoare), adresele absolute rămân neschimbate. Exemple:

- SRC="imagine.gif" imaginea se află în același director cu documentul HTML în care urmează să fie plasată
- SRC="imagini/imagine.gif" imaginea se află în subdirectorul imagini (la un nivel imediat inferior) în timp ce documentul exemplu.html se află în directorul probe
- SRC="../imagine.gif" imaginea se află în directorul imediat superior (pentru a face această referire în sens superior se utilizează "../")
- SRC="../../imagine.gif" imaginea se află la două nivele superioare față de fișierul exemplu.html
- SRC="../imagini/imagine.gif" calea ce trebuie parcursă este un director în sus (pentru a ajunge în directorul probe) și un director în jos pentru a ajunge în directorul imagini
- SRC="../../media/imagini/imagine.gif" 3 nivele în sus (sunt numerotate) pentru a ajunge în directorul principal și de 2 nivele în jos pentru a ajunge în subdirectorul imagini. În practică ar trebui evitate asemenea structuri alambicate.

În ceea ce privește adresele absolute, acestea nu se modifică în funcție de poziția fizică a fișierului de unde sunt apelate. Revenind la ultimul exemplu, o adresă absolută ar arăta sub forma:

```
SRC="http://www.numesite.com/principal/media/imagini/imagine.gif"
```

Adresele relative sunt de preferat celor absolute. Avantajul principal este dat de faptul că fișierele sunt accesate ceva mai rapid. La acesta se adaugă și economia de efort care se face prin omiterea scrierii unei adrese complete. Utilizând adresele relative, accesarea legăturilor dintr-un site se poate face atât în faza de proiectare (atunci când directoarele se află pe HDD propriu) cât și în cazul în care există o variantă "offline" care va fi oferită vizitatorilor. Adresele relative pot fi utilizate numai în cazul în care atât fișierele care sunt referite, cât și fișierele din care se face referirea se regăsesc pe același site.

Alinierea imaginilor se face utilizând tag-ul ALIGN. Exemple:

```
<IMG SRC="pics/calc.gif" ALIGN="top">calculator  
<IMG SRC="pics/ calc.gif" ALIGN="middle">calculator  
<IMG SRC="pics/ calc.gif" ALIGN="bottom">calculator
```

Pentru a se furniza o descriere a imaginii, se utilizează atributul ALT. Acest lucru este util în cazurile în care browser-ele nu au activată opțiunea de redare a imaginilor sau în cazul în care, datorită unei conexiuni mai lente la rețea sau un server prea lent, nu se poate încărca partea grafică a unei pagini (sau se încarcă incomplet). În această situație, în locul imaginii va apărea un chenar în care va fi scris textul alternativ. Exemplu:

Acesta este un calculator.

```
<IMG SRC="calc.gif" ALIGN="left" ALT="Descriere imagine">
```

Acesta este un calculator.

Dimensiunile exacte ale unei imagini se declară prin intermediul atributelor WIDTH și HEIGHT. De exemplu:

```
<IMG SRC="calc.gif" ALIGN="left" width="80" height="60" ALIGN="left"ALT="Descriere imagine">
```

```
<CENTER><IMG SRC="cerc.gif" WIDTH="400" HEIGHT="4"></CENTER>
```

Controlul spațiului pe orizontală și pe verticală se realizează prin atributele HSPACE și VSPACE. De exemplu,

```
<IMG SRC="calc.gif" width="80" height="60"HSPACE="20" VSPACE="20" align="left">
```

Imaginile pot fi folosite și în crearea unor liste, cum ar fi în exemplul următor:

```
<UL>
```

```
  <IMG SRC="a.gif" ALIGN="TOP">Imaginea A<BR>
```

```
  <IMG SRC="b.gif" ALIGN="TOP">Imaginea B <BR>
```

```
  <IMG SRC="ci.gif" ALIGN="TOP">Imaginea C <BR>
```

```
</UL>
```

## Link-uri si pozitionare

Un element important al oricarei pagini web îl constituie **link-urile** către alte pagini. Acestea pot fi link-uri externe, care fac referire către un alt site, sau link-uri interne care fac referire către alte pagini ale aceluiași site. Ambele tipuri se codifică asemănător prin folosirea unui tag de tipul

```
<A HREF=link.html>...</A>
```

(pentru un link intern) sau

```
<A HREF=http://www.abc.abc>...</A>
```

pentru un link extern. Astfel codul de mai jos va afișa textul "Apasati aici pentru pagina 2" care prin apăsare va încărca fișierul pag2.html:

```
<A HREF=pag2.html>Apasati aici pentru pagina 2</A>
```

```
<A HREF="http://www.google.com/">Motor de căutare</a>
```

Codul de mai jos va afisa "Apasati aici pentru pagina XYZ SRL", text care prin apasare va incarca pagina [www.xyz.xyz](http://www.xyz.xyz):

```
<A HREF=http://www.xyz/xyz>Apasati aici pentru pagina XYZ SRL</A>
```

In exemplul de mai sus se poate specifica in ce mod se va incarca pagina respectiva folosind atributul **TARGET**:

- TARGET="nume", unde nume este numele pe care vrei sa il dai unei noi ferestre unde se va incarca pagina. Poti folosi si TARGET="\_blank" pentru o fereastră noua sau TARGET="\_self" pentru incarcarea in aceeasi fereastră.

O alta varianta este posibilitatea transmiterii unui e-mail. Astfel, codul de mai jos afiseaza "Scrie-ne!", text care prin apasare va deschide o fereastră pentru transmiterea unui mesaj la adresa [office@abc.abc](mailto:office@abc.abc):

```
<A HREF="mailto:office@abc.abc">Scrie-ne!</A>
```

In acelasi mod o pagina HTML poate fi impartita in mai multe "capitole" (sectiuni), in caz ca este foarte lunga. Astfel, fiecărei sectiuni i se va asigna un nume, nume ce va putea fi utilizat in incarcarea paginii respective, astfel incat sa se afiseze exact sectiunea respectiva. Fie fisierul lucrare.html care se dorește a fi impartit in doua sectiuni. Codul care face acest lucru ar fi:

```
<A NAME=unu>Sectiunea 1</A>
```

...text pentru sectiunea 1...

```
<A NAME=doi>Sectiunea 2</A>
```

..text pentru sectiunea 2...

Valoarea atributului NAME este unică pentru o pagină HTML. Acest atribut poate avea aceeași valoare, în două pagini HTML diferite.

Daca se dorește să se incarce fisierul lucrare.html din alta pagina, dar astfel incat sa se incarce direct sectiunea 2, se va utiliza:

```
<A HREF=lucrare.html#doi>Apasa aici</A>
```

In mod similar se pot afisa fisiere text (cu extensia **txt**) sau se poate da posibilitatea vizitatorului sa salveze pe disc un alt tip de fisier (**zip** de exemplu):

```
<A HREF=raport.txt>Raport</A>
```

sau

```
<A HREF=program.zip>Salveaza programul</A>
```

Un alt element util in formatarea paginii web este tag-ul **<CENTER >...</CENTER>**. Acesta este utilizat pentru alinierea pe centru a unor elemente (texte, tabele imagini). Tot ce este scris intre tag-ul **<CENTER>** si **</CENTER>** va apare centrat.

```
<B><A HREF="mailto:info@abc.abc.ro">e-mail</A></B>
```

```
<A HREF=http://www.xyz.xyz TARGET="_blank">aici</A>
```

Acum toate elementele tabelului (titlul si cele doua tabele) sunt pozitionate pe centru si exista link-uri pentru e-mail si pagina XYZ.

Exemple:

```
<OL>
  <LI><A NAME="1" HREF="#raspuns1">întrebarea 1</A>
  <LI><A NAME="2" HREF="#raspuns2">întrebarea 2</A>
  <LI> .....
</OL>
<OL>
  <LI><A HREF="raspuns1">text raspuns 1...</A><A HREF="#1">inapoi</A>
  <LI><A HREF="raspuns2">text raspuns 2...</A><A HREF="#2">inapoi</A>
  <LI> .....
</OL>
```

Utilizarea imaginilor ca legături:

```
<A HREF="http://www.test.com/"><IMG SRC="foto/logo1.gif" ALT="Teste HTML"></A>
```

Realizarea legăturilor către elemente non-HTML:

```
<A HREF="tutorial.pdf">Tutorial HTML în format PDF</A>
```