

CUPRINS

CAP. 1. NOȚIUNI GENERALE

- 1.1. Ce înseamnă HTML?**
- 1.2. Documentele HTML – structură**
- 1.3. Elemente de bază**

CAP. 2. CREAREA ȘI EDITAREA UNEI PAGINI WEB

- 2.1. Alegerea programului de editare**
- 2.2. Crearea unui document web**
- 2.3. Stabilirea proprietăților documentului**
- 2.4. Utilizarea imaginilor ca fundal**
- 2.5. Aplicație practică**

CAP. 3. FORMATAREA DOCUMENTULUI

- 3.1. Crearea titlurilor în pagină (Headings)**
- 3.2. Crearea paragrafelor**
- 3.3. Ruperea rândurilor**
- 3.4. Despre paragrafe pre-formatate**
- 3.5. Inserarea unei linii orizontale**
- 3.6. Formatarea unei secțiuni a documentului**
- 3.7. Aplicație practică**

CAP. 4. FORMATAREA TEXTULUI

- 4.1. Schimbarea aspectului textului**
- 4.2. Schimbarea stilului textului**
- 4.3. Alte elemente de formatare a textului**
- 4.4. Caractere speciale și simboluri**
- 4.5. Aplicație practică**

CAP. 5. CREAREA LISTELOR

- 5.1. Noțiuni generale**
- 5.2. Listele ordonate**
- 5.3. Listele neordonate**
- 5.4. Listele de definiții**
- 5.5. Listele imbricate**
- 5.6. Aplicație practică**

CAP. 6. HYPERLINK-URI

- 6.1. Noțiuni generale**
- 6.2. URL-uri**
- 6.3. Ancore (Anchors)**
- 6.4. Aplicație practică**

CAP. 7. FOLOSIREA IMAGINILOR

- 7.1. Tipuri de imagini**
- 7.2. Inserarea imaginilor**
- 7.3. Formatarea imaginilor**
- 7.4. Aplicație practică**
- 7.5. Folosirea imaginilor ca link-uri**
- 7.6. Imagini de tip client-side image map**
- 7.7. Aplicație practică**

CAP. 8. TABELE

- 8.1. Ce sunt tabelele?**
- 8.2. Formatarea tabelelor**
- 8.3. Sub-elementele tabelelor**
- 8.4. Aplicație practică**

CAP. 9. CADRE

- 9.1. Noțiuni generale**
- 9.2. Structura unei pagini ce definește cadre**
- 9.3. Atributele cadrelor**
- 9.4. Destinații**
- 9.5. Cadre interioare**
- 9.6. Elementul <NOFRAMES>**
- 9.7. Aplicație practică**

CAP. 10. FORMULARE

- 10.1. Noțiuni generale**
- 10.2. Elementele formularelor**
- 10.3. Crearea unui formular**
- 10.4. Codificarea datelor**
- 10.5. Inserarea controalelor de tip <INPUT> într-un formular**
- 10.6. Alte tipuri de controale**
- 10.7. Alte atribute ale formularelor**
- 10.8. Aplicație practică**

CAP. 11. INTRODUCERE ÎN JAVA ȘI JAVASCRIPT

- 11.1. Noțiuni generale**
- 11.2. Limbajul JAVA**
- 11.3. Limbajul JAVASCRIPT**
- 11.4. Aplicație practică**

CAP. 12. PAGINI WEB MULTIMEDIA

- 12.1. Ce este multimedia?**
- 12.2. Formatul fișierelor multimedia ce conțin sunete**
- 12.3. Formatul fișierelor multimedia ce conțin video**
- 12.4. Adăugarea de informații multimedia unei pagini web**

- 12.5. Inserarea unui film QuickTime într-o pagină web**
- 12.6. Inserarea unui film Real Video într-o pagină web**
- 12.7. Alte metode de inserare a unor sunete într-o pagină web**
- 12.8. Aplicație practică**

CAP. 13. ELEMENTE SPECIALE

- 13.1. Meta-Elementele**
- 13.2. Inserarea comentariilor**
- 13.3. Aplicație practică**

CAP. 14. CONSIDERAȚII GENERALE DESPRE DESIGN ȘI ASPECT

- 14.1. Diferențe dintre un document web și unul tipărit**
- 14.2. Mărimea optimă a unei paginii web și optimizarea ei**
- 14.3. Structura și aspectul unui site**
- 14.4. Combinarea culorilor**
- 14.5. Tipurile de caractere folosite**

ÎNCHEIERE

CAP. 1. NOȚIUNI GENERALE

1.1. Ce înseamnă HTML?

HTML este acronimul de la **HyperText Markup Language** și reprezintă un limbaj pentru crearea și marcarea (formatare, aranjare) unui document astfel încât să poată fi publicat pe World Wide Web și vizualizat cu ajutorul unui browser (Internet Explorer, Netscape, Opera etc.).

Termenul de hypertext desemnează un material sub formă de text și imagine, interconectat într-o manieră complexă, nesecvențială, în care utilizatorul poate naviga, căuta informații referitoare la un obiect. Hypertext-ul trebuie interpretat ca un text care semnalează o legătură la o altă informație web, de obicei un alt document web, și este identificat prin subliniere sau culoare, pentru a-l deosebi de textul simplu.

Hypermedia este un termen aproape sinonim celui de hypertext, singura deosebire fiind faptul că subliniază prezența și a unor elemente care nu sunt de tip text, cum ar fi animații, secvențe sonore sau secvențe video.

HTML se utilizează din 1990, cunoscând câteva versiuni de dezvoltare, fiecare dintre acestea îmbunătățind performanțele limbajului. Ultima variantă (la data elaborării acestui ghid) este HTML 4.01. ce include facilitățile versiunilor anterioare (tag-uri de marcă, tag-uri pentru hiperlegături, antete, paragrafe, liste, elemente de meniu, formatare caractere, imagini in-line și tag-uri pentru schimbul de date dinamic între utilizatori), adăugând facilități și extensii pentru numere, tabele și elemente de control.

1.2. Documentele HTML – structură

Un document HTML este format dintr-o succesiune de blocuri de informație. Aceste blocuri pot fi incluse unul în altul. Un bloc este delimitat de simboluri speciale, numite **tag-uri** (etichete). Modul în care un document este marcat cu elemente și cu atribute ale acestor elemente se realizează în conformitate cu Document Type Definition (DTD – definiția tipului de document). Aceasta conține regulile ce caracterizează fiecare tip de document.

Sursa autorizată pentru furnizarea de informații despre HTML și HTML DTD este World Wide Web Consortium (W3C) având adresa <http://www.w3.org>.

Tag-ul este termenul consacrat pentru a defini elementele cu care sunt marcate textul și grafica într-o pagină web. Fiecare tag este încadrat de semnele “<” (mai mic) și “>” (mai mare).

De exemplu: Acest text se va vedea îngroșat într-un browser

Primul cuvânt sau caracter ce apare în interiorul acestor paranteze poartă numele de **element**.

Majoritatea elementelor au un tag de deschidere și unul de închidere cu aceeași structură, dar cu prezența caracterului “/” în fața denumirii elementului.

De exemplu: este un tag de închidere

Cuvântul sau cuvintele ce urmează după element și despărțite de acesta printr-un spațiu, poartă de numirea de **atribute**, având rolul de descrie proprietățile elementelor.

Atributele sunt urmate de semnul “=” și pot avea diferite **valori**. Valoarea unui atribut este trecută, de obicei, între ghilimele.

De exemplu: Acest text va fi albastru

Elementele HTML pot avea unul sau mai multe atribute:

De exemplu: Acest text va fi albastru și cu o unitate mai mare decât normal

De remarcat că atributele nu apar și în tag-urile de închidere.

Elementele HTML specifică structura logică a unui document web și sugerează prezentarea vizuală a documentului. HTML furnizează două tipuri de elemente:

- tag-urile care permit delimitarea antetelor, paragrafelor, listelor, tabelelor, legăturilor, imaginilor, etc;
- referințele de entitate caracter care permit utilizarea în documentele HTML a simbolurilor declarate drept caractere de control. Acestea se identifică ușor deoarece întotdeauna încep cu **ampersand** “&” . De exemplu, pentru a reprezenta simbolul “<” într-un document HTML, se va utiliza referința de entitate caracter **<** (less then).

Notă: Tag-urile se închid în ordinea inversă deschiderii lor, sintaxa corectă a unei formatări cu mai multe tag-uri fiind reprezentată astfel:

<TAG1><TAG2>...<TAGn> obiect formatat </TAGn>...</TAG2></TAG1>

1.3. Elemente de bază

Orice pagină web (document HTML) are în structură trei elemente obligatorii: un element numit **HTML**, care cuprinde întregul document, și două sub-elemente ale acestuia: **HEAD** (antet, cap) și **BODY** (corp).

- Tag-ul <HTML> este primul tag care trebuie să apară într-un fișier HTML. El va încadra alături de tag-ul său corespunzător de închidere (</HTML>) întreaga pagină web.
- Tag-ul <HEAD> va marca partea de antet a paginii web. Are tag corespunzător de închidere (</HEAD>).
- Tag-ul <BODY> va încadra conținutul paginii web. Are tag corespunzător de închidere (</BODY>).
- Tag-ul <TITLE> marchează titlul unui document HTML, cel care va fi afișat în bara de titlu a browser-ului. Acest tag se folosește numai în interiorul tag-ului HEAD. Dacă acest tag lipsește, atunci în bara de titlu va apărea numele fișierului.

Iată un model de structură a unui document HTML:

<HTML>

<HEAD>

<TITLE>*Titlul documentului***</TITLE>**

</HEAD>

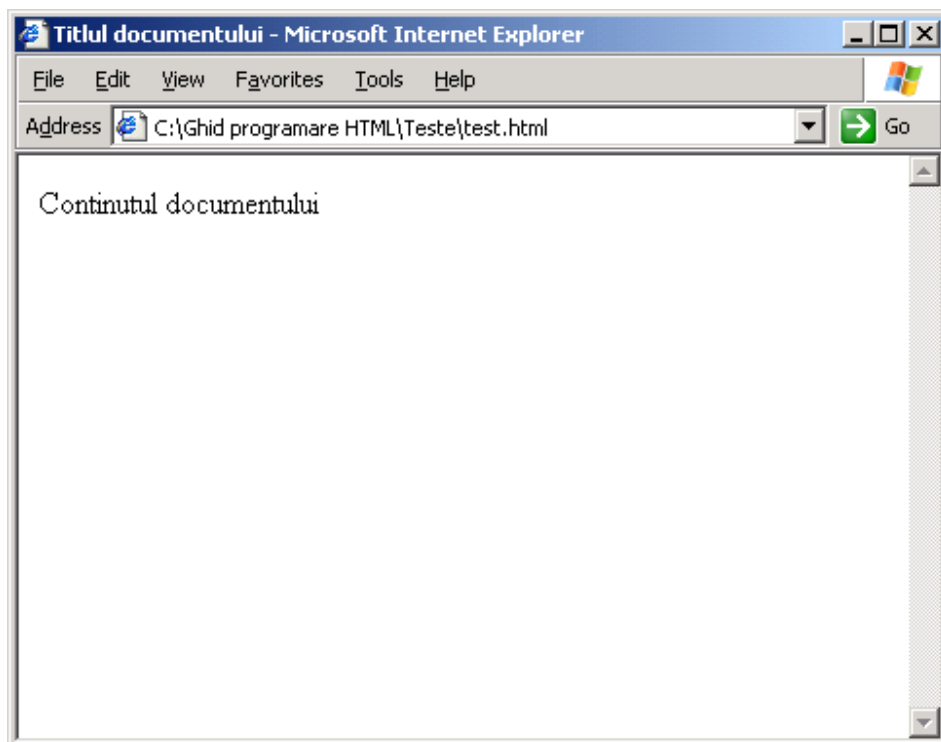
<BODY>

Continutul documentului

</BODY>

</HTML>

Și iată cum arată un document HTML ce utilizează tag-urile din structura anterioară:



Notă: Modul de scriere al denumirii tag-urilor este indiferent de caz, scrierea lor cu caractere majuscule sau cu caractere minuscule nu influențează comportamentul acestora: <TAG> este echivalent cu <tag> și cu <Tag> sau cu <taG> etc. Pentru evidențierea mai pronunțată a acestora, am optat pentru scrierea cu majuscule a denumirii tag-urilor pe parcursul întregii lucrări.

CAP. 2. CREAREA ȘI EDITAREA UNEI PAGINI WEB

2.1. Alegerea programului de editare

Există o multitudine de programe pe care le puteți folosi la crearea paginilor web.

Editoarele de text sunt cele mai simple editoare care permit crearea și salvarea fișierelor fără coduri de formatare ascunse, care pot afecta modul de afișare a unei pagini web în browsere.

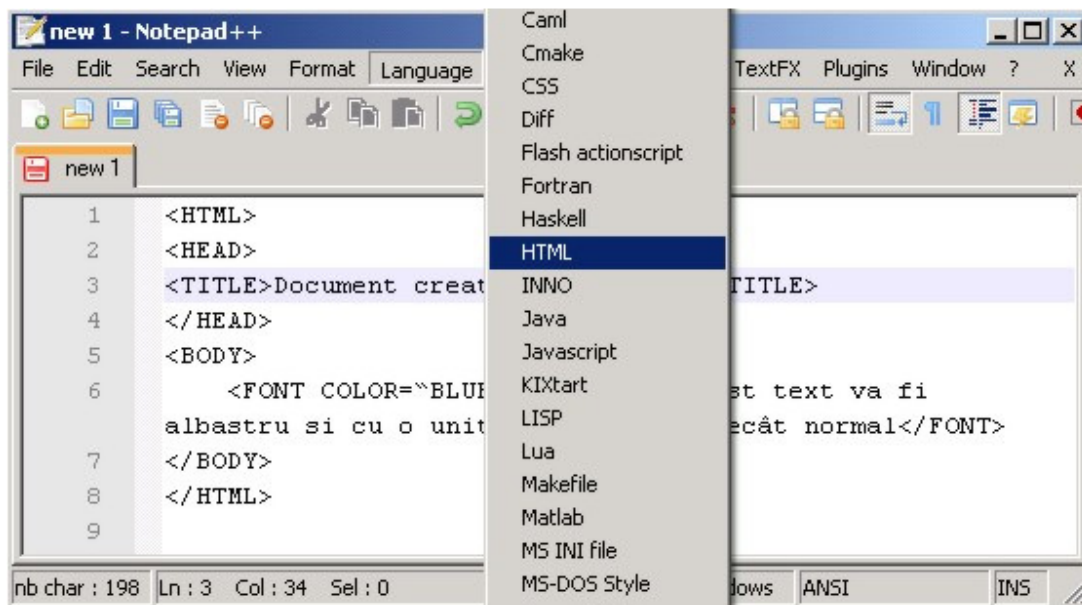
Editoarele specializate HTML le permit utilizatorilor să creeze documente web într-un mod rapid și ușor și, prin urmare, foarte eficient, doar prin apăsarea câtorva butoane sau prin apelarea câtorva funcții predefinite. În loc de a scrie de mână codul sursă HTML, aceste programe generează ele însele acest cod în locul vostru. Acest tip de editoare reprezintă unelte excelente pentru dezvoltatorii web cu experiență. Chiar și în acest caz este necesar să înțelegeți limbajul HTML pentru a putea edita codul și a înlătura eventualele neajunsuri din documentele web.

În prezent, chiar și versiunile Microsoft Word vă permit salvarea documentelor în format pagină web, dar, dacă veți analiza conținutul codului sursă, veți avea surpriza să constatați cât de mult a fost “înzorzonat” față de varianta scrisă de mână.

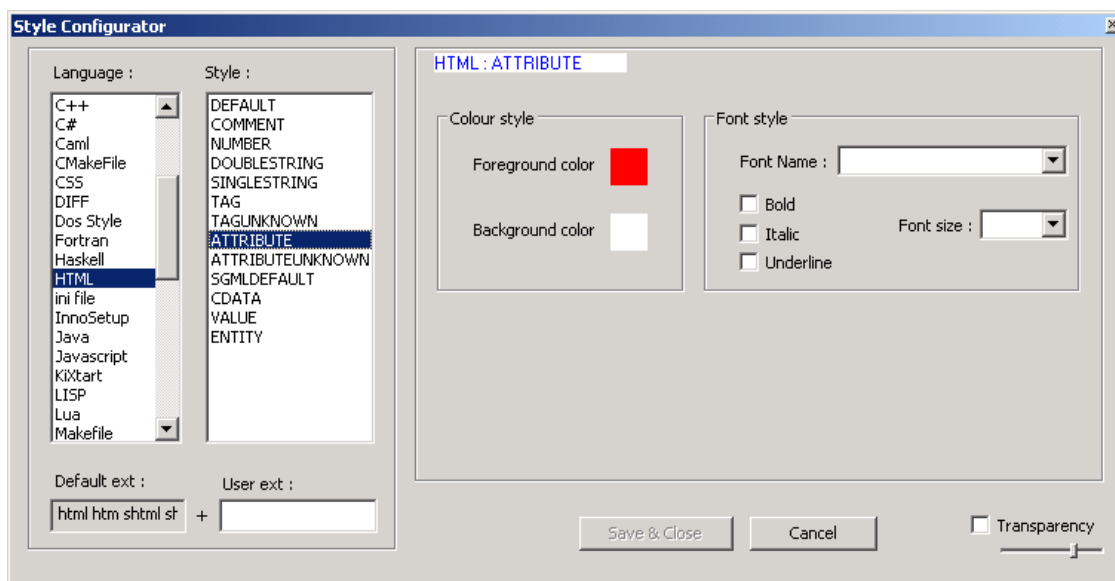
Pentru exemplificarea practică a noțiunilor teoretice prezentate în acest ghid, vă recomand folosirea editorului **Notepad++**. Acesta este un editor gratuit de cod-sursă care suportă mai multe limbaje de programare și rulează sub Windows. Oferă facilități precum Syntax Highlighting, stiluri personalizate, editare simultană a mai multor fișiere, multiview și bookmarks. Acest produs, bazat pe componentele de editare Scintilla (un component de editare foarte puternic) este sub licență GPL (General Public License). Programul poate fi găsit la adresa <http://notepad-plus.sourceforge.net>.

Datorită folosirii unor culori diferite, acest program vă permite să efectuați o distincție clară a elementelor, atributelor și valorilor, contribuind în acest fel la înțelegerea noțiunilor generale de programare.

Dar mai întâi va trebui să selectați tipul de limbaj folosit, în cazul nostru HTML, din meniul **Language**.

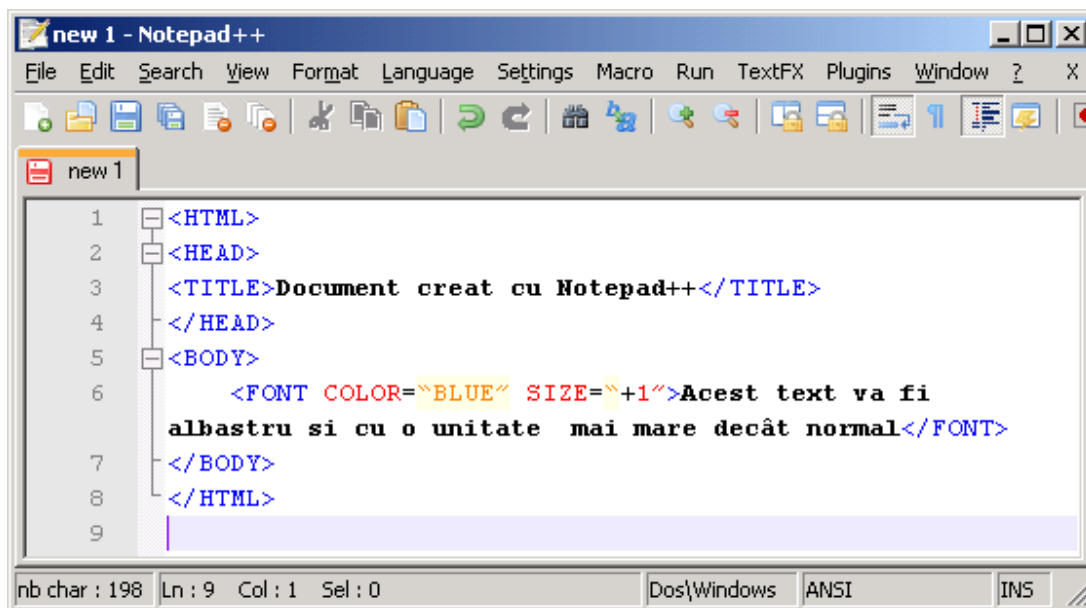


Există o schemă de culori implicită pentru afișarea codurilor dar poate fi definită și de utilizator din meniul **Settings → Styler configurator....**



Pentru exemplificare, am ales culoarea albastră pentru tag-uri, culoarea roșie pentru atribute, culoarea portocalie pentru valori și culoarea neagră pentru textul general.

În Notepad++ codurile unui document HTML se vor vedea astfel:



2.2. Crearea unui document web

Așa cum am menționat și în capitolul precedent, există anumite elemente care sunt strict necesare în structura unui document web. Prin urmare, întotdeauna când creați un document web va trebui să începeți prin scrierea acestor elemente esențiale, după care veți adăuga celelalte tag-uri.

Prin urmare, în Notepad++ veți scrie următoarele:

<HTML>

<HEAD>

<TITLE> </TITLE>

</HEAD>

<BODY>

</BODY>

</HTML>

În acest moment aveți secțiunile HEAD și BODY cuprinse între tag-urile HTML. Mai aveți, de asemenea, și elementul TITLE, în interiorul elementului HEAD. Textul pe care îl veți scrie în interiorul tag-urilor TITLE va fi afișat în browsere, pe rândul cel mai de sus, precum și în motoarele de căutare. În cazul în care nu scrieți nimic, în browsere va fi afișat numele documentului web (de exemplu: index.html).

Dacă un utilizator dorește să pună un semn de carte “Bookmark” la documentul respectiv, similar cu adăugarea acestuia la “Favorites”, textul conținut în TITLE va fi cel adăugat în listă.

Iată un exemplu de titlu:

<TITLE> Ghid de programare HTML </TITLE>

2.3. Stabilirea proprietăților documentului

Întregul conținut al documentului format din text, imagini, culori, elemente grafice, va fi cuprins între tag-urile **BODY**.

Proprietățile documentului sunt controlate de attributele elementului BODY. Aceste attribute sunt:

Atribut	Valoare	Descriere
alink	rgb(x,x,x) #xxxxxx colorname	Precizează culoarea link-urilor active din document.
background	nume fișier	O imagine folosită ca fundal pentru document
bgcolor	rgb(x,x,x) #xxxxxx colorname	Precizează culoarea fundalului documentului.
link	rgb(x,x,x) #xxxxxx colorname	Precizează culoarea link-urilor din document.
text	rgb(x,x,x) #xxxxxx colorname	Precizează culoarea textului utilizat în document.
vlink	rgb(x,x,x) #xxxxxx colorname	Precizează culoarea link-urilor vizitate din document.

Culorile sunt precizate folosind codurile RGB (red-green-blue) și reprezentate prin valorile hexazecimale. Fiecare secțiune formată din două cifre reprezintă valoarea componentei nuanțelor de roșu, verde și albastru ce intră în componența culorii respective.

Vă prezentăm în continuare un tabel cu cele mai uzuale culori, compoziția RGB, precum și codul hexazecimal aferent fiecărei culori:

Denumire culoare	Valoare RGB	Cod hexazecimal
Black (negru)	0;0;0	000000
Blue (albastru)	0;0;255	0000FF

SaddleBrown (maron)	139;69;19	8B4513
Grey (gri)	84;84;84	545454
Green (verde)	0;255;0	00FF00
Orange (portocaliu)	255;165;0	FFA500
Red (roșu)	255;0;0	FF0000
Purple (violet)	128;0;128	800080
White (alb)	255;255;255	FFFFFF
Yellow (galben)	255;255;0	FFFF00

Cei mai mulți designeri web își ale schemele de culori astfel încât să se asorteze cu schema generală de culori a site-ului.

Modul de formatare a acestor attribute este următorul:

<BODY BGCOLOR="#000000" TEXT="#FFFF00" LINK="#0000FF" VLINK="#800080" ALINK="#FF0000">

Rezultatul codurilor precedente va fi un document web cu fundal negru, textul de culoare galbenă, link-urile de culoare albastră, link-urile active de culoare roșie și link-urile vizitate de culoare violetă.

Notă: Codurile hexazecimale vor fi precedate întotdeauna de semnul "#"

2.4. Utilizarea imaginilor ca fundal

Elementul BODY permite, de asemenea, utilizarea unei imagini ca fundal. Această imagine va fi afișată în mod repetat (tiled) pe fundal în browsere, adică va fi multiplicată pe toată suprafața documentului web. În vederea realizării unui aspect profesional, este recomandat ca imaginile folosite ca fundal să poată fi alăturate fără însă a li se observa îmbinările. Imaginile folosite ca fundal trebuie să fie în format .gif sau .jpg.

Vă recomandăm să aveți în vedere faptul că întreg conținutul documentului va fi afișat deasupra imaginii folosite ca fundal. Prin urmare, imaginea va trebui să fie în culori pale, sau în contrast cu culoarea textului afișat, astfel încât textul să poată fi citit cu ușurință.

Sintaxa folosirii acestui atribut într-un document web este următoarea:

<BODY BACKGROUND="fundal.gif">

Rezultatul va fi afișarea ca fundal a imaginii din fișierul fundal.gif.

2.5. Aplicație practică

În baza celor prezentate anterior, vom realiza un document web folosind editorul Notepad++.

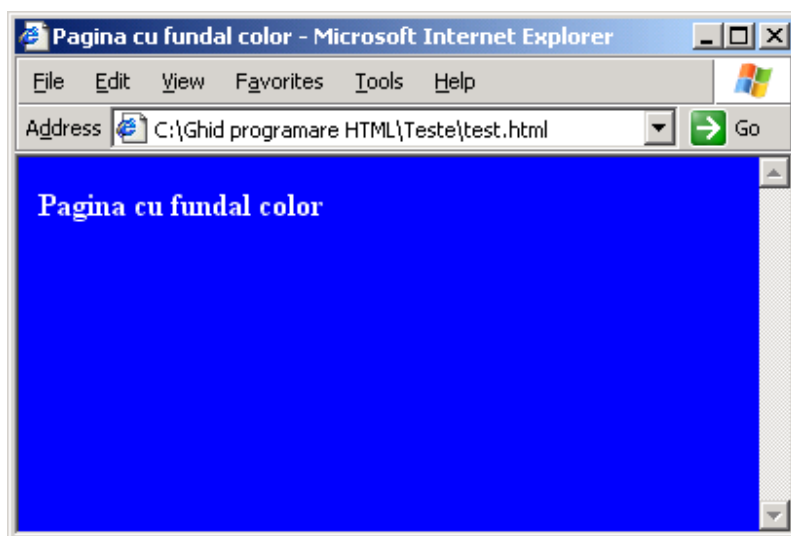
În documentul nou creat (new1) vom scrie următoarele linii de limbaj HTML:

```
<HTML>
<HEAD>
  <TITLE>Pagina cu fundal color</TITLE>
</HEAD>
<BODY BGCOLOR="#0000FF" TEXT="#FFFFFF">
  <B>Pagina cu fundal color</B>
</BODY>
</HTML>
```

Din meniul **File** se alege opțiunea **Save**, se stabilește calea unde va fi salvat fișierul (de exemplu **C:\Ghid programare HTML\Teste**) și vom denumi fișierul **test.html**

Pentru a vedea rezultatul exercițiului în browserul cu care suntem familiarizați (Internet Explorer, Netscape, Opera etc.) vom deschide fișierul respectiv (din meniul **File** se alege opțiunea **Open** sau **Open File**, în funcție de browser).

Rezultatul afișat va fi următorul:



Și acum vom crea un nou document (sau îl putem modifica pe cel creat anterior), pentru a utiliza o imagine ca fundal.

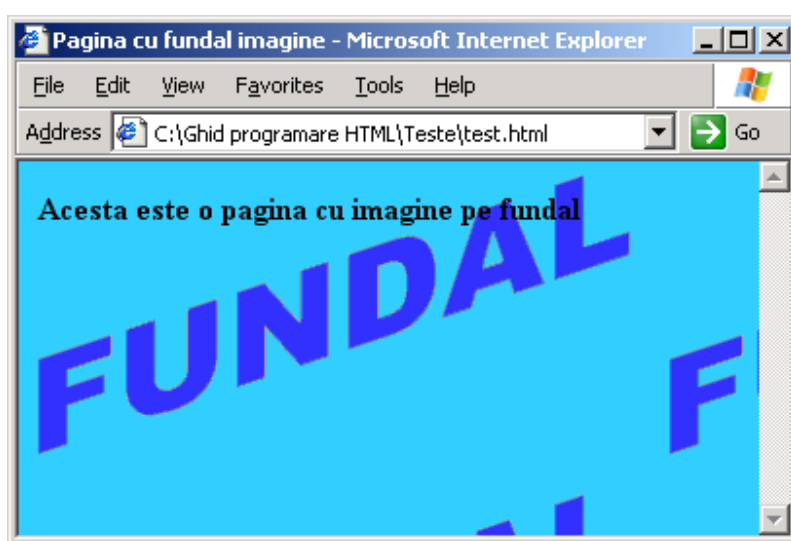
```
<HTML>
<HEAD>
  <TITLE>Pagina cu fundal imagine</TITLE>
</HEAD>
<BODY BACKGROUND="fundal.gif">
```

```
<B>Acesta este o pagina cu imagine pe fundal</B>
</BODY>
</HTML>
```

Dacă nu aveți un fișier de tip imagine (.gif sau .jpg) care se pretează ca fundal de pagină, puteți descărca fișierul nostru *fundal.gif* dând click [aici](#).

Notă: Fiți atenți la calea utilizată la descărcarea fișierului! Fișierul *fundal.gif* trebuie salvat în același director ca și fișierul *test.html*

Rezultatul afișat va fi următorul:



Rezultatul va fi identic și dacă folosiți imaginea existentă pe site-ul nostru, în condițiile în care îi specificați calea exactă și aveți o conexiune internet permanentă, astfel:

```
<HTML>
<HEAD>
  <TITLE>Pagina cu fundal imagine</TITLE>
</HEAD>
<BODY BACKGROUND="http://html.webmarketclub.com/teste/fundal.gif">
  Acesta este o pagină cu imagine pe fundal
</BODY>
</HTML>
```

CAP. 3. FORMATAREA DOCUMENTULUI

3.1. Crearea titlurilor în pagină (Headings)

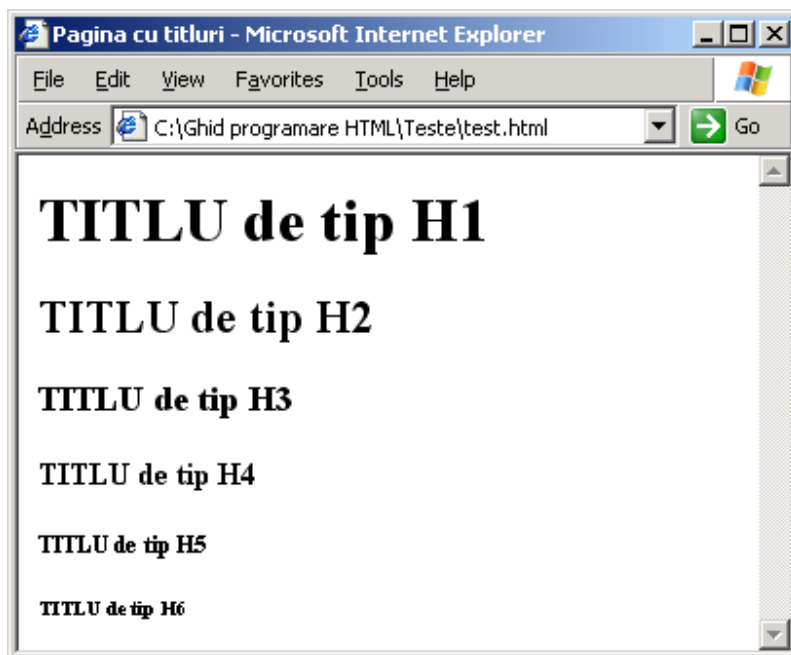
Elementele constituite din titluri și subtitluri se regăsesc în interiorul secțiunii BODY și sunt marcate de tag-urile **<H1>** până la **<H6>**, având dimensiunile corespunzătoare pe o scală de la 1 la 6, unde **<H1>** are dimensiunea cea mai mare, iar **<H6>** are dimensiunea cea mai mică. Este necesară folosirea tag-urilor de închidere **</H1>** ... **</H6>**.

Prezența titlurilor într-un document web este facultativă, ele putând apărea în orice ordine dorită de creatorul documentului, deși, pentru a obține cele mai bune efecte vizuale, este recomandat să le utilizați în ordine crescătoare (de la H1 la H6).

Iată, cum arată concret formatarea acestor titluri într-un document web.

```
<HTML>
<HEAD>
  <TITLE>Pagina cu titluri</TITLE>
</HEAD>
<BODY>
  <H1>TITLU de tip H1</H1>
  <H2>TITLU de tip H2</H2>
  <H3>TITLU de tip H3</H3>
  <H4>TITLU de tip H4</H4>
  <H5>TITLU de tip H5</H5>
  <H6>TITLU de tip H6</H6>
</BODY>
</HTML>
```

Rezultatul afișat va fi următorul:



Atributele specifice acestui element sunt:

Atribut	Valoare	Descriere
align	left center right justify	Precizează modul de aliniere a textului cuprins în titlu.

3.2. Crearea paragrafelor

Paragrafele vă permit să adăugați text în documentul web într-o asemenea manieră încât este ajustat în mod automat la dimensiunea ferestrei browser-ului în care este afișat, adică fiecare rând de text va avea lățimea ferestrei în care este afișat.

Pentru a marca prezența unui paragraf se folosește tag-ul **<P>**. Acesta necesită tag de închidere **</P>**.

Formatarea paragrafelor într-un document web se realizează astfel:

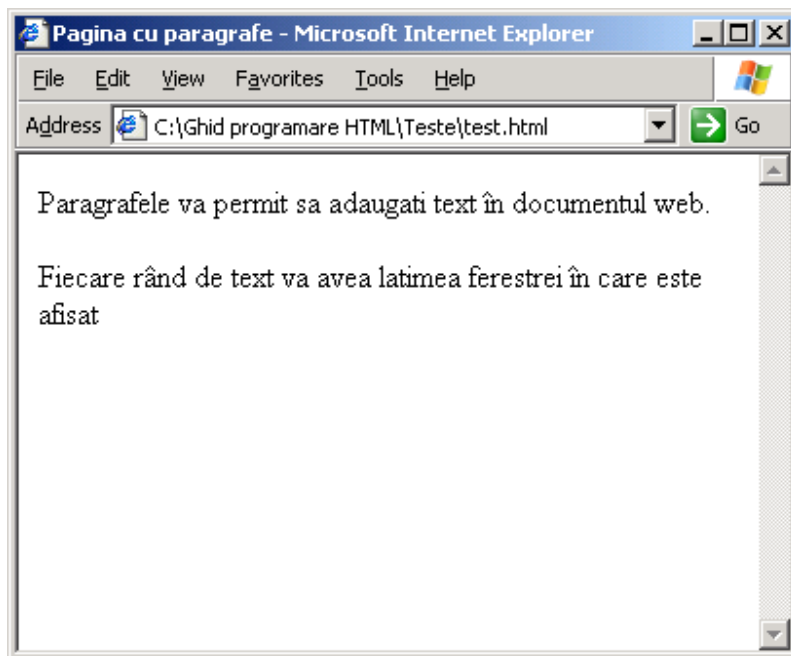
```

<HTML>
<HEAD>
  <TITLE>Pagina cu paragrafe</TITLE>
</HEAD>
<BODY>
  <P>Paragrafele va permit sa adaugati text în documentul web.</P>
  <P>Fiecare rând de text va avea latimea ferestrei în care este afisat </P>
</BODY>

```

</HTML>

Rezultatul afișat va fi următorul:



Notă: Dacă veți dori să introduceți mai multe spații goale între cuvintele din interiorul unui paragraf, va trebui să folosiți codul “ ” (NBSP = Non-Breaking Space) deoarece browser-ele ignoră spațiile multiple dintre cuvinte.

Atributele specifice acestui element sunt:

Atribut	Valoare	Descriere
align	left center right justify	Precizează modul de aliniere a textului cuprins în paragraf.

3.3. Ruperea rândurilor

O modalitate de a împărți o zonă de text pe rânduri este de a semnala browser-ului exact unde vrem să fie efectuată ruperea de rânduri, folosind tag-ul **
**. Acesta impune afișarea textului situat după el pe linia următoare. Nu are tag corespunzător de închidere.

Pentru exemplificare vom folosi următorul cod:

<HTML>

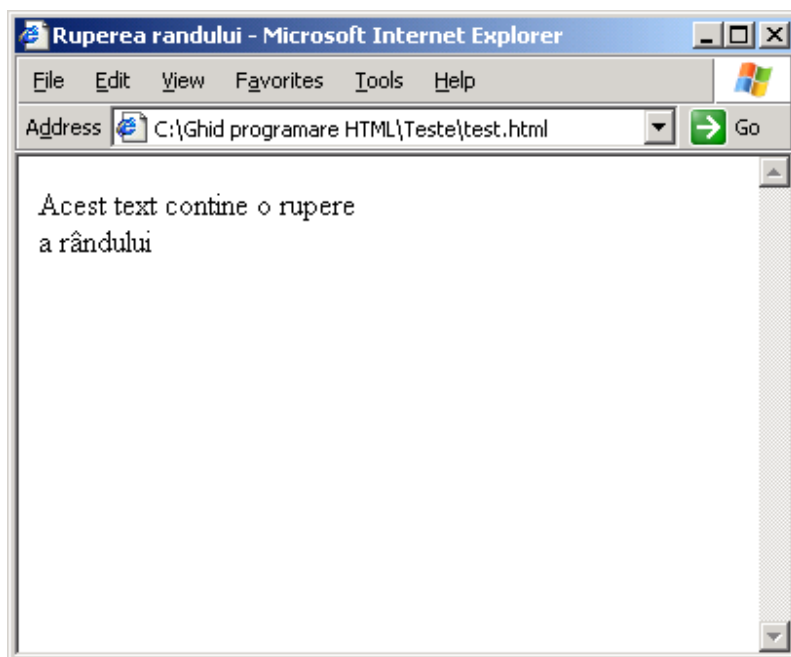
<HEAD>

```

<TITLE>Ruperea randului</TITLE>
</HEAD>
<BODY>
  <P>Acest text contine o rupere <BR> a rândului </P>
</BODY>
</HTML>

```

Rezultatul afișat va fi următorul:



3.4. Despre paragrafe pre-formatate

Paragrafele pre-formatate sunt definite prin tag-ul **<PRE>**. Acesta se folosește pentru a indica browser-ului că textul trebuie să fie afișat exact așa cum este scris în codul HTML, respectând spațiile și ruperea rândurilor. Necesită tag corespunzător de închidere **</PRE>**.

Iată cum se poate pune în practică folosirea acestui tag:

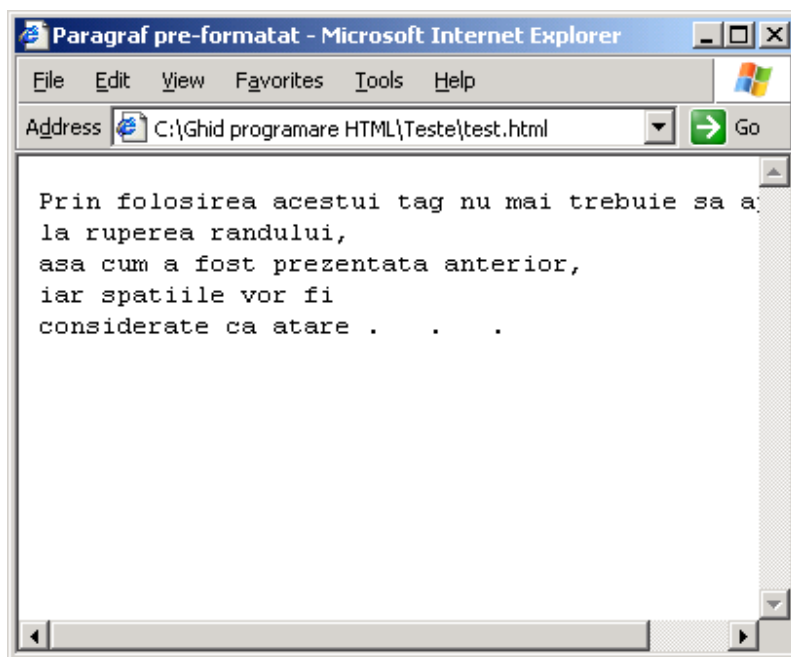
```

<HTML>
<HEAD>
  <TITLE>Paragraf pre-formatat</TITLE>
</HEAD>
<BODY>
  <PRE>Prin folosirea acestui tag nu mai trebuie sa apelam
    la ruperea randului,
    asa cum a fost prezentata anterior,
    iar spatiile vor fi

```

```
considerate ca atare . . .</PRE>  
</BODY>  
</HTML>
```

Rezultatul afișat va fi următorul:



De remarcat că acest tag are o utilizare mai restrânsă, un neajuns major fiind faptul că textul nu este ajustat în mod automat la dimensiunea ferestrei browser-ului în care este afișat (așa cum se poate observa și în imagine), iar fontul folosit este din grupa Courier New.

Ajustarea lățiiimii paragrafului se poate face prin folosirea următorului atribut:

Atribut	Valoare	Descriere
width	număr	Precizează numărul maxim de caractere pe un rând (în mod obișnuit 40, 80 sau 132), în funcție de rezoluția monitotului recomandată utilizatorilor.

3.5. Inserarea unei linii orizontale

Un alt element ce contribuie la redarea unui aspect mai placut unui document web este linia orizontală (Horizontal Line), redată în limbajul HTML prin tag-ul **<HR>**.

Acest element nu are un tag de închidere. Pur și simplu adăugați tag-ul **<HR>** în interiorul secțiunii BODY și veți avea ca rezultat o linie orizontală.

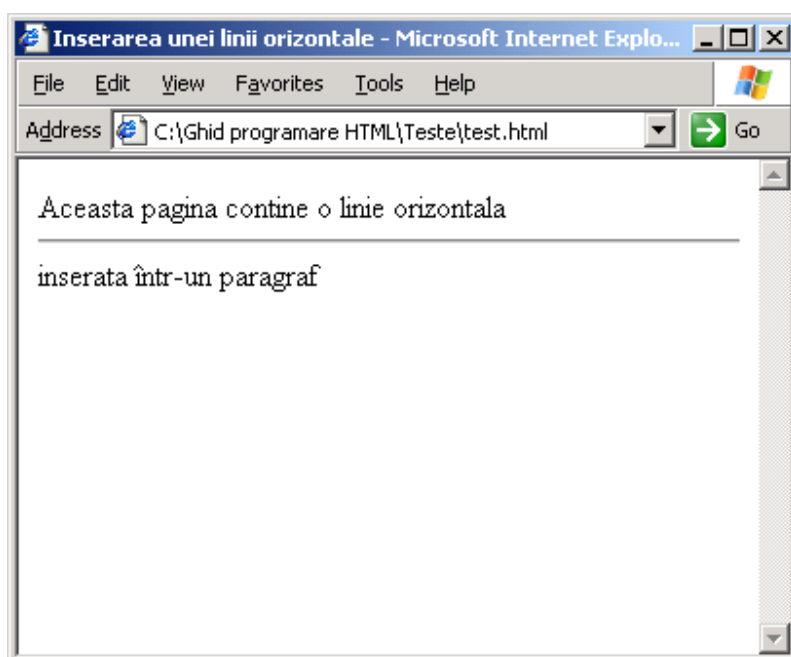
Vom exemplifica cele de mai sus folosind următorul cod:

```

<HTML>
<HEAD>
    <TITLE>Inserarea unei linii orizontale</TITLE>
</HEAD>
<BODY>
    <P>Aceasta pagina contine o linie orizontala <HR> inserata într-un
paragraf</P>
</BODY>
</HTML>

```

Rezultatul afișat va fi următorul:



Atributele specifice acestui element sunt:

Atribut	Valoare	Descriere
align	left center right	Precizează modul de aliniere liniei în document.
noshade	noshade	Când este prezent, acest atribut are ca rezultat înlăturarea efectului de umbră, setat în mod implicit pentru linia orizontală.
size	pixels %	Precizează grosimea liniei orizontale.
width	pixels %	Precizează lățimea liniei orizontale.
color	rgb(x,x,x)	Precizează culoarea liniei orizontale.

	#xxxxxx colorname	
--	----------------------	--

3.6. Formatarea unei secțiuni a documentului

Pentru formatarea unei întregi secțiuni (division) a unui document se folosește tag-ul **<DIV>**. În interiorul unei secțiuni pot fi cuprinse o serie de elemente care vor păstra caracteristicile acestea, stabilite prin atributele sale, cu excepția cazului în care acele elemente nu au propriile atribute.

Folosirea acestui tag la deschidere necesită și folosirea tag-ului **</DIV>** la închidere.

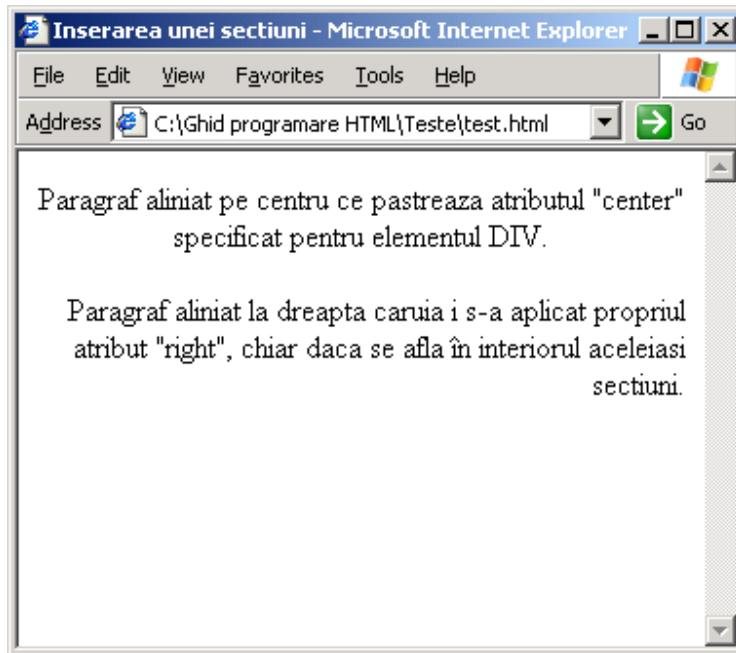
Atributele specifice acestui element sunt:

Atribut	Valoare	Descriere
align	left center right justify	Precizează modul de aliniere a textului cuprins în interiorul secțiunii.

Pentru exemplificare vom folosi următorul cod:

```
<HTML>
<HEAD>
<TITLE>Inserarea unei secțiuni</TITLE>
</HEAD>
<BODY>
  <DIV align="center">
    <P>Paragraf aliniat pe centru ce pastreaza atributul "center" specificat
    pentru elementul DIV.</P>
    <P align="right">Paragraf aliniat la dreapta caruia i s-a aplicat propriul
    atribut "right", chiar daca se afla în interiorul aceleiasi secțiuni.</P>
  </DIV>
</BODY>
</HTML>
```

Rezultatul afișat va fi următorul:



3.7. Aplicație practică

În baza celor prezentate în acest capitol, vom crea un nou fișier numit **test2.html** folosind editorul Notepad++, urmând pașii descriși în capitolul precedent, în care vom scrie următoarele:

<HTML>

<HEAD>

<TITLE>Testarea cunostintelor dobândite în Capitolul 3</TITLE>

</HEAD>

<BODY>

Începem prin inserarea unui text neformatat, care, după cum se poate observa, are caracteristicile unui text de tip paragraf, aliniat la stânga.

Inserăm un Titlu de tip H1

Inserăm si un Titlu de tip H2

<P align="justify">Adăugăm text în document, aliniat atât la stânga cât și la dreapta, text ce va avea lățimea ferestrei în care este afisat.</P>

*<P align="right">Puteti observa rezultatul inserării unei
 rupei de rând în interiorul unui paragraf, precum si folosirea unor spatii multiple între cuvinte, într-un paragraf aliniat la dreapta.</P>*

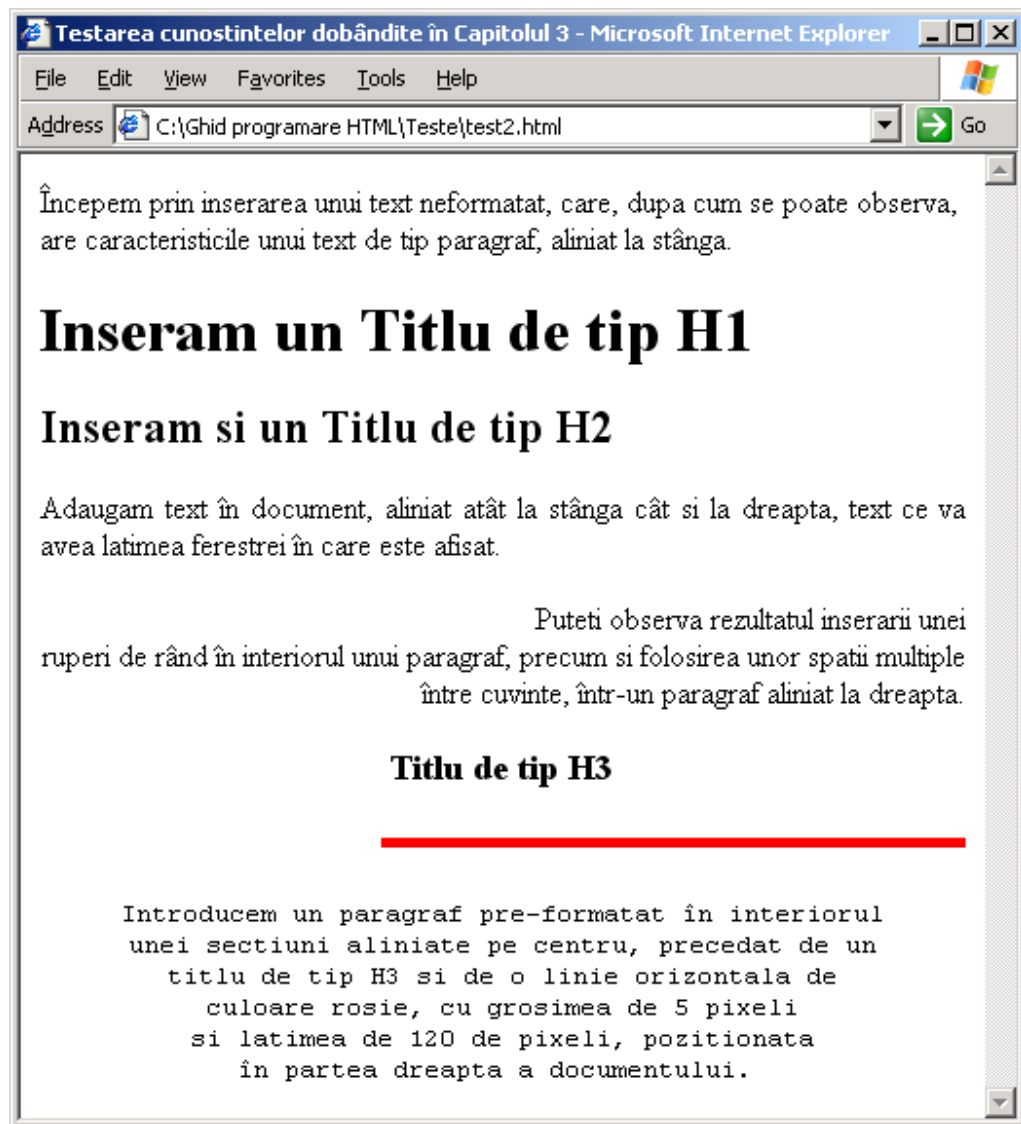
<DIV align="center">

Titlu de tip H3

**<PRE width="40">Introducem un paragraf pre-formatat în interiorul
unei secțiuni aliniate pe centru, precedat de un
titlu de tip H3 și de o linie orizontală de**

*culoare rosie, cu grosimea de 5 pixeli
si lătimea de 120 de pixeli, pozitionată
în partea dreaptă a documentului.</PRE></DIV>
</BODY>
</HTML>*

Rezultatul afişat va fi următorul:



*Note: 1. În Internet Explorer n-a fost recunoscut atributul width="40" al elementului <PRE>, ci doar în Netscape.
2. Aspectul documentului afişat în browser depinde şi de dimensiunea ferestrei în care este vizualizat, prin urmare, poate fi diferit de imaginea prezentată.*

CAP. 4. FORMATAREA TEXTULUI

4.1. Schimbarea aspectului textului

Dacă până acum am vorbit despre formatarea unor întregi secțiuni sau paragrafe, a sosit timpul să ne ocupăm de formatarea textului în detaliu, și anume de formatarea proprie fiecărui cuvânt sau caracter.

Pentru început vom analiza tag-ul **** care stabilește tipul de caracter folosit, mărimea și culoarea acestuia. Necesită tag de închidere ****.

Prin urmare, atributele care definesc acest tag sunt următoarele:

Atribut	Valoare	Descriere
color	rgb(x,x,x) #xxxxxx colorname	Precizează culoarea caracterelor.
face	denumirea tipului de caracter	Precizează tipul de caracter ce va fi folosit în text. Pot fi specificate mai multe tipuri concomitent (pentru cazul în care un utilizator nu are anumit tip de font, de exemplu <i>face="Arial,Tahoma,Verdana,Helvetica"</i>)
size	o valoare numerică de la 1 la 7	Precizează mărimea caracterelor (valoarea implicită este 3).

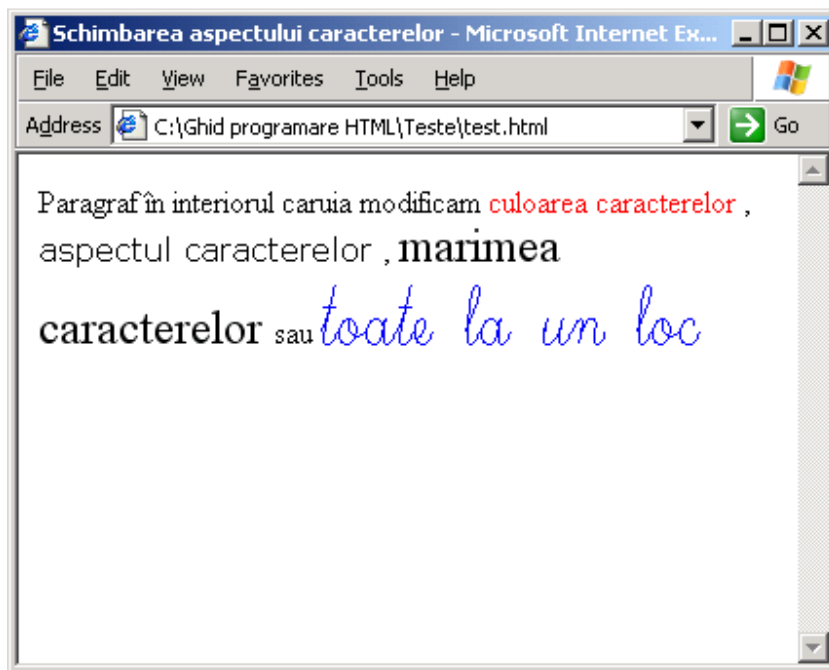
Elementul **** poate fi cuprins în interiorului multor elemente (cum ar fi paragraf, titlu, secțiune etc.), dar la fel de bine aceste elemente se pot afla în interiorul elementului ****, rezultatele obținute fiind similare.

**<H1>Titlu</H1> ** este similar cu
<H1>Titlu </H1>

În exemplul următor vom avea în vedere aspecte legate de elementul **** și ne vom folosi de același fișier **test.html**:

```
<HTML>
<HEAD>
<TITLE> Schimbarea aspectului caracterelor</TITLE>
</HEAD>
<BODY>
  <P>Paragraf în interiorul caruia modificam <FONT color="red">
culoarea caracterelor </FONT> , <FONT face="Verdana"> aspectul caracterelor
</FONT> , <FONT size="+2"> mărimea caracterelor </FONT> sau <FONT
color="blue" face="Script" size="+5"> toate la un loc </FONT> </P>
</BODY>
</HTML>
```

Rezultatul afișat va fi următorul:



Notă: Prezența elementului <P> în exemplul precedent este facultativă.

4.2. Schimbarea stilului textului

Principalele elementele de stil ce caracterizează corpurile de literă și care contribuie la evidențierea acestora, sunt marcate de următoarele tag-uri:

 - are ca efect **îngroșarea** textului

<I> - are ca efect *italicizarea* textului

<U> - are ca efect sublinierea textului

<BIG> - are ca efect **mărirea** textului

<SMALL> - are ca efect **micșorarea** textului

<TT> - are ca efect spațierea în mod egal a textului (litera I și litera M vor ocupa același spațiu în interiorul unui cuvânt)

<S> <STRIKE> - toate au ca efect afișarea ~~textului~~ tăiat cu o linie orizontală

Toate aceste tag-uri necesită tag-uri de închidere.

Pentru exemplificare vom folosi următorul cod:

<HTML>

<HEAD>

<TITLE> *Schimbarea stilului caracterelor* </TITLE>

</HEAD>

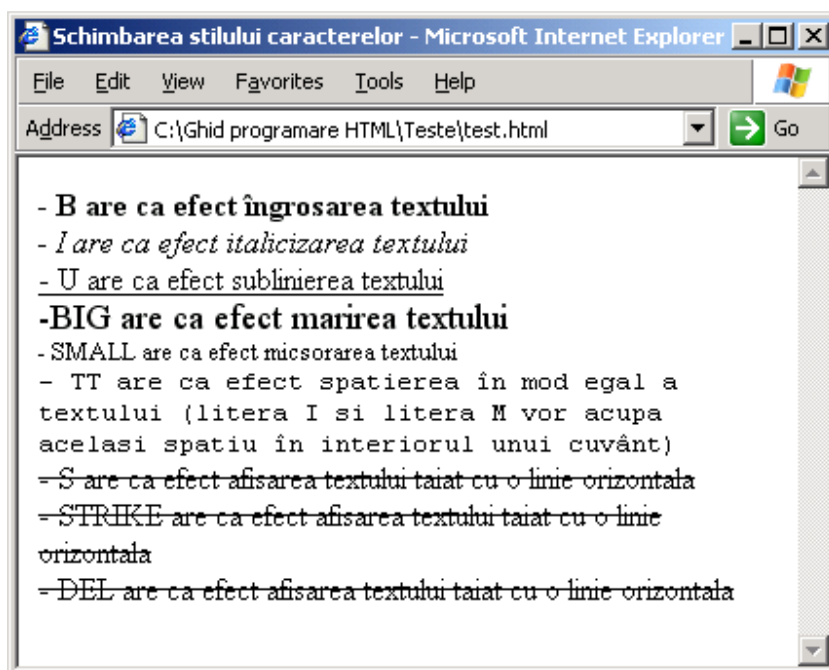
<BODY>

 - **B** are ca efect **îngrosarea textului**

<I> - *I* are ca efect *italicizarea textului* </I>

<U> - *U* are ca efect sublinierea textului **</U>
**
<BIG> - *BIG* are ca efect marirea textului **</BIG>
**
<SMALL> - *SMALL* are ca efect micșorarea textului **</SMALL>
**
<TT> - *TT* are ca efect spațierea în mod egal a textului (litera I și litera M vor ocupa același spațiu în interiorul unui cuvânt) **</TT>
**
<S> - *S* are ca efect afișarea textului tăiat cu o linie orizontală **</S>
**
<STRIKE> - *STRIKE* are ca efect afișarea textului tăiat cu o linie orizontală **</STRIKE>
**
**** - *DEL* are ca efect afișarea textului tăiat cu o linie orizontală ****
</BODY>
</HTML>

Rezultatul afișat va fi următorul:



4.3. Alte elemente de formatare a textului

Pe lângă elementele prezentate mai sus, mai există o serie de elemente cu o utilizare mai restrânsă pe care le vom menționa pe scurt:

<SUP> - Permite afișarea caracterelor exponent (^{superscript})
<SUB> - Permite afișarea caracterelor indice (_{subscript})
**** - Accentuează un *text*
**** - Întărește modul de afișare a unui **text**
<DFN> - Indică faptul că textul reprezintă *definiția unui termen*
<CODE> - Indică faptul că textul reprezintă *un cod de calculator*
<SAMP> - Indică faptul că textul reprezintă *un model de cod de calculator*
<KBD> - Indică faptul că textul reprezintă *un text de tastatură*

<VAR> - Indică faptul că textul reprezintă *o variabilă*

<CITE> - Indică faptul că textul reprezintă *un citat*

Toate aceste tag-uri necesită tag-uri de închidere.

4.4. Caractere speciale și simboluri

Caracterele speciale și simbolurile care nu se găsesc, în mod obișnuit, pe tastatură, pot fi inserate în text prin secvențe de cod speciale. Aceste caractere se regăsesc într-un set internațional de caractere cunoscut și sub denumirea de ISOLatin-1 (ISO-8859-1).

Caracterele speciale sunt recunoscute de limbajul HTML datorită faptului că încep cu semnul "&" (ampersand) și se încheie cu semnul ";" (punct și virgulă).

Cele mai uzuale astfel de caractere au fost cuprinse în tabelul următor:

Caracterul special	Reprezentare	Simbolizare
ampersand	&	&
asterisk	∗	*
cent sign	¢	¢
copyright	©	©
fraction one qtr	¼	¼
fraction one half	½	½
greater-than sign	>	>
less-than sign	<	<
non-breaking space	 	
quotation mark	"	"
registration mark	®	®
trademark sign	™	™

4.5. Aplicație practică

Vom pune în practică cele prezentate anterior și vom crea un nou fișier numit **test3.html** care va conține următoarele linii de cod:

<HTML>

<HEAD>

<TITLE> *Elemente de formatare a textului***</TITLE>**

</HEAD>

<BODY>

Titlu de tip H3

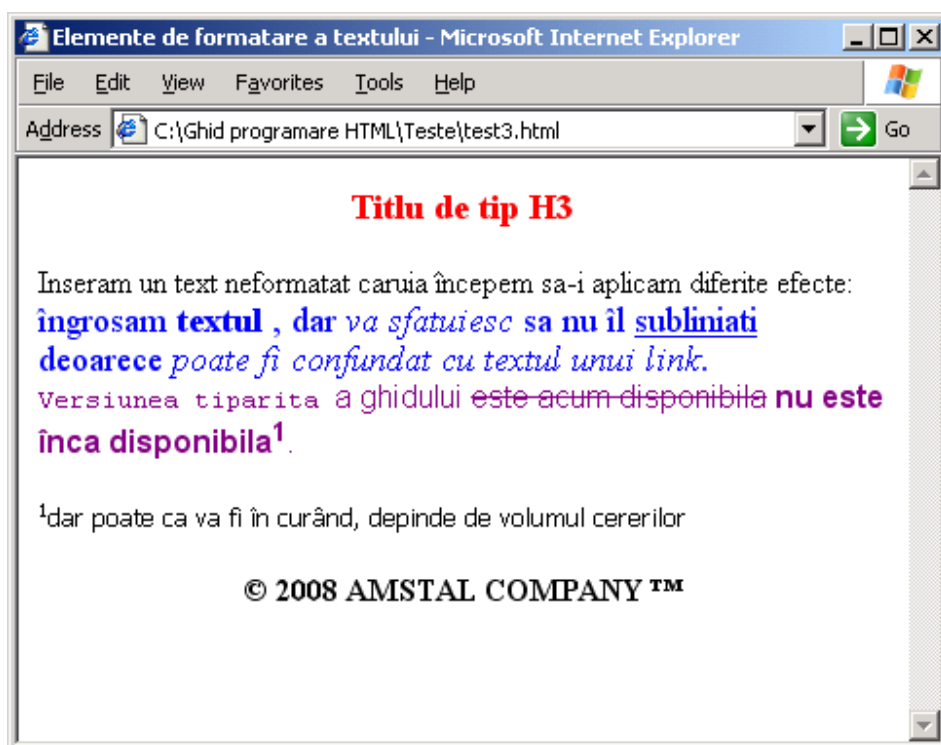
Inseram un text neformatat caruia începem sa-i aplicam diferite efecte: **îngrosam textul**, dar *va sfatuiesc sa nu îl subliniati* deoarece *poate fi confundat cu textul unui link*.

Versiunea tiparita a ghidului ¹ este acum disponibila **nu este încă disponibila**.

¹dar poate ca va fi în curând, depinde de volumul cererilor

© 2008 AMSTAL COMPANY

Rezultatul afișat va fi următorul:



Notă: Puteți observa în codul sursă marcarea tag-urilor atât cu caractere majuscule, cât și cu caractere minuscule, rezultatul nefiind influențat de modul de scriere al acestora, întrucât browser-ele nu țin cont de aceste diferențe.

CAP. 5. CREAREA LISTELOR

5.1. Noțiuni generale

Limbajul HTML le permite utilizatorilor să folosească diverse mecanisme pentru prezentarea informațiilor sub forma listelor.

Listele reprezintă succesiuni de elemente, fiecare element fiind evidențiat printr-un număr sau printr-un marcaj, având rolul de a atrage atenția asupra unor idei din text. Fiecare listă poate conține unul sau mai multe elemente.

În funcție de elementele conținute, listele sunt de trei tipuri:

- liste ordonate
- liste neordonate
- liste de definiții

5.2. Listele ordonate

Într-o listă ordonată elementele sunt numerotate automat de browser. Tag-ul ce marchează începutul unei liste ordonate este ****. Folosirea tag-ului de închidere **** este obligatorie.

Elementele listei sunt marcate de tag-ul ****. Folosirea tag-ului de închidere **** este opțională.

Modul de folosire a acestor tag-uri este redat în exemplul următor:

```
<OL>
<LI>primul element</LI>
<LI>al doilea element</LI>
<LI>al treilea element</LI>
</OL>
```

având ca rezultat afișarea listei în această formă:

1. primul element
2. al doilea element
3. al treilea element

Atributele specifice acestui element sunt:

Atribut	Valoare	Descriere
start	număr sau literă	Precizează valoarea cu care va începe numerotarea.
type	A	Sunt folosite simboluri alfabetice majuscule
	a	Sunt folosite simboluri alfabetice minuscule
	I	Sunt folosite simboluri numerice romane majuscule
	i	Sunt folosite simboluri numerice romane minuscule
	1	Sunt folosite simboluri numerice arabe

5.3. Listele neordonate

Într-o listă neordonată elementele sunt marcate prin simboluri similare cu **Bullets** din Microsoft Word, simbolul implicit fiind • “disc”. Tag-ul ce marchează începutul unei liste ordonate este `<UL>`. Folosirea tag-ului de închidere `</UL>` este obligatorie.

La fel ca și la listele ordonate, elementele listei sunt marcate de tag-ul `<LI>`.

Modul de folosire a acestor tag-uri este redat în exemplul următor:

```
<UL>
<LI>primul element</LI>
<LI>al doilea element</LI>
<LI>al treilea element</LI>
</UL>
```

având ca rezultat afișarea listei în această formă:

- primul element
- al doilea element
- al treilea element

Atributele specifice acestui element sunt:

Atribut	Valoare	Descriere
type	disc	Sunt folosite simboluri de tip disc
	square	Sunt folosite simboluri de tip pătrat
	circle	Sunt folosite simboluri de tip cerc

5.4. Listele de definiții

Listele de definiții sunt puțin diferite de celelalte două tipuri, întrucât fiecare termen al listei se compune din două elemente: **Termenul Definiției** și **Descrierea Definiției**.

Tag-ul ce marchează începutul unei liste de definiții este `<DL>`. Folosirea tag-ului de închidere `</DL>` este obligatorie.

Tag-ul ce marchează termenul definiției este `<DT>`. Folosirea tag-ului de închidere `</DT>` este opțională.

Tag-ul ce marchează descrierea definiției este `<DD>`. Folosirea tag-ului de închidere `</DD>` este opțională.

Modul de folosire a acestui tag este redat în exemplul următor:

```
<DL>
<DT>HTML</DT>
<DD>Hyper Text Markup Language</DD>
<DT>WWW</DT>
<DD>World Wide Web!</DD>
```

</DL>

având ca rezultat afișarea listei în această formă:

HTML
Hyper Text Markup Language
WWW
World Wide Web

După cum se poate observa, descrierea definiției este aliniată mai interior, tocmai pentru a sublinia caracterul relației de subordonare existent față de termenul definiției.

5.5. Listele imbricate

Uneori este necesar să folosim anumite tipuri de liste în interiorul altora. Împreună, acestea formează **listele** imbricate. În astfel de cazuri, lista interioară nu mai trebuie să fie delimitată de tag-ul .

Modul de folosire a listelor imbricate este redat în exemplul următor:

```
<OL type="I">  
  <LI>Introducere  
    <OL TYPE="I">  
      <LI>Cuvânt înainte  
      <LI>Notiuni generale  
    </OL>  
  <LI>Realizarea unei pagini web  
    <OL TYPE="I" START="3">  
      <LI>Alegerea editorului HTML  
      <LI>Machetarea paginii  
      <LI>Elementele HTML  
      <UL TYPE="circle">  
        <LI>Sectiunea HEAD  
        <LI>Sectiunea BODY  
      </UL>  
      <LI>Apleturi JAVA  
    </OL>  
  <LI>Optimizarea paginii web  
  <LI>Publicarea pe internet a paginii web  
</OL>
```

având ca rezultat afișarea listei în această formă:

- I. Introducere
 - 1. Cuvânt înainte
 - 2. Notiuni generale

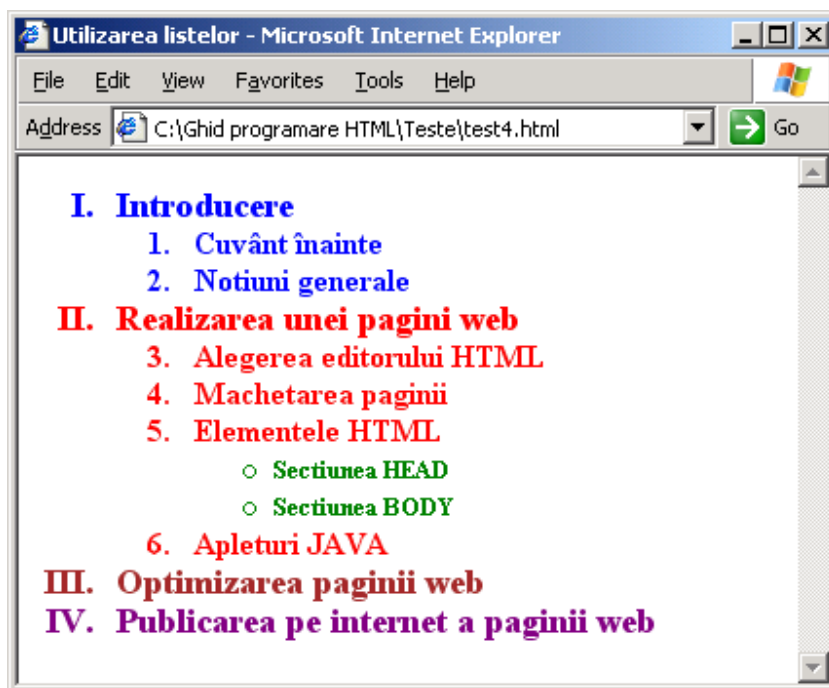
- II. Realizarea unei pagini web
 - 3. Alegerea editorului HTML
 - 4. Machetarea paginii
 - 5. Elementele HTML
 - Sectiunea HEAD
 - Sectiunea BODY
 - 6. Apleturi JAVA
- III. Optimizarea paginii web
- IV. Publicarea pe internet a paginii web

5.6. Aplicație practică

Vom încerca în continuare să punem în practică cele prezentate anterior combinate cu elemente de formatare a textului și vom crea un nou fișier numit **test4.html** care va conține următoarele linii de cod:

```
<HTML>
<HEAD>
<TITLE>Utilizarea listelor</TITLE>
</HEAD>
<BODY>
<B>
<OL type="I"><FONT color="blue" size="4">
  <LI>Introducere
    <OL TYPE="I"><FONT size="3">
      <LI>Cuvânt înainte
      <LI>Noțiuni generale
    </OL><FONT color="red" size="4">
    <LI>Realizarea unei pagini web
    <OL TYPE="I" START="3"><FONT size="3">
      <LI>Alegerea editorului HTML
      <LI>Machetarea paginii
      <LI>Elementele HTML
      <UL TYPE="circle"><FONT color="green" size="2">
        <LI>Sectiunea HEAD
        <LI>Sectiunea BODY</FONT>
      </UL>
      <LI>Apleturi JAVA</FONT>
    </OL>
    <FONT color="brown"><LI>Optimizarea paginii web</FONT>
    <FONT color="purple"><LI>Publicarea pe internet a paginii web
    </FONT>
  </OL></FONT>
</B></BODY>
</HTML>
```

Rezultatul afișat va fi următorul:



CAP. 6. HYPERLINK-URI

6.1. Noțiuni generale

Hypertext-ul reprezintă o metodă de realizare a unui text, ideală pentru utilizarea pe calculator, care permite cititorului să parcurgă materialul în maniera aleasă de el. Pentru a realiza un hypertext, mai întâi „tranșați” informațiile în unități mici, manevrabile, cum ar fi paginile individuale de text. Aceste unități sunt numite **noduri**. Apoi înglobați în text **hyperlink-uri** (numite și **ancore**). Când cititorul execută click pe un hyperlink, programul de hypertext afișează un nod diferit. Procesul de navigare printre nodurile legate în acest fel se numește răsfoire (**browsing**). O colecție de hyperlink-uri este numită web. WWW (World Wide Web) este un sistem de hypertext la scară globală. Aplicațiile de hypertext sunt foarte utile în special atunci când se lucrează cu cantități mari de text, ca în cazul enciclopediilor sau al documentelor juridice.

Pentru simplificare, ca și până acum, când facem referire la hyperlink-uri le vom numi simplu **link-uri**.

Pentru a include un link către un document propriu sau aflat pe un alt server, este absolut necesar să cunoașteți adresa exactă a acelui document și modul de inserare a unei **ancore** în propriul document HTML.

Adresa unui document poate fi inserată la modul relativ sau la modul absolut.

De exemplu:

test.html – reprezintă o referință relativă

în vreme ce

C:\Ghid programare HTML\Teste\test.html – reprezintă o referință absolută

6.2. URL-uri

În Internet, fiecare document are propria sa adresă identificată printr-un URL (Uniform Resource Locator). Acesta constă dintr-o combinație de elemente scrise într-o anumită ordine, astfel:

protocol://numele_de_domeniu/cale

unde:

protocol – reprezintă software-ul sau maniera prin care serverul de domeniu comunică documentul unui browser

numele_de_domeniu – reprezintă numele alocat pe Internet serverului ce găzduiește documentul web

calea – reprezintă numele directorului (și al eventualelor subdirectoare) unde este stocat documentul web

Iată câteva exemple de URL-uri:

<http://html.webmarketclub.com/index.html>

- *tipul protocolului este **http***
- ***html** este, în acest caz, numele unui subdomeniu din domeniul **webmarketclub** situat în zona **com** (ce identifică site-urile comerciale)*
- ***index.html** este numele unui fișier*

<http://html.webmarketclub.com/teste/computer.jpg>

- *tipul protocolului este **http***
- ***html** este, ca și mai sus, numele unui subdomeniu*
- ***teste** reprezintă numele unui subdirector*
- ***computer.jpg** este numele unui fișier*

ftp://ftp.uni-stuttgart.de/pub/download/madyn_v15_160107.zip

- *tipul protocolului este **ftp**. Acest serviciu este utilizat pentru a crea link-uri către fișiere ce pot fi descărcate după servere ftp.*
- *gazda este [ftp.uni-stuttgart.de](ftp://ftp.uni-stuttgart.de)*
- ***pub** și **download** reprezintă numele unor subdirectoare*
- ***madyn_v15_160107.zip** este numele unui fișier*

mailto: html@webmarketclub.com

- *tipul serviciului este identificat ca fiind **mail client program**. Acest link va lansa în execuție clientul de mail al utilizatorului.*
- *destinatarul email-ului este html@webmarketclub.com*

6.3. Ancore

Ancorele (anchors) reprezintă elementele HTML care precizează atât sursa cât și destinația unui link. Sunt marcate prin tag-ul de deschidere **<A>** și necesită tag-ul de închidere ****.

O ancoră poate fi utilizată în două moduri:

1. Pentru a crea un *semn de carte (bookmark)* în interiorul unui document utilizând atributul **NAME** sau **ID**. Odată creat, acesta devine ținta potențială a unui link.
2. Pentru a crea un link către alt document sau către un semn de carte utilizând atributul **HREF**, valoarea atributului fiind dată de un anumit URL.

*Notă: Vă reamintim că schema de culori ce va fi afișată pentru link-uri este controlată de atributele elementului **BODY**, și anume **LINK**, **VLINK** și **ALINK**, descrise în Capitolul 2 din acest ghid. Nespecificarea acestor valori va determina browser-ul să afișeze schema de culori implicită.*

Pentru exemplificare vom folosi următoarele secvențe de cod:

```
<A NAME="nume_semn_de_carte">Semn de carte</A>
```

are ca rezultat atribuirea unui semn de carte

```
<A HREF="#nume_semn_de_carte ">Du-te la semnul de carte </A>
```

are ca rezultat realizarea unui link în document către semnul de carte

```
<A HREF="http://html.webmarketclub.com/teste/test5.html">Du-te la Testul 5  
pe site-ul Webmarket Club </A>
```

are ca rezultat realizarea unui link în document către un alt document pe Internet, în acest caz **test5.html**

Desigur, poate fi combinat link-ul către o pagină cu link-ul intern existent pe acea pagină, având ca rezultat specificarea unei secțiuni anumite dintr-un document web.

```
<A HREF="http://html.webmarketclub.com/teste/test5.html#semn_de_carte">  
Du-te la semnul de carte din cadrul Testului 5 pe site-ul Webmarket Club </A>
```

Tot cu ajutorul ancorelor se poate trimite un mail printr-un singur click în interiorul unui document, astfel:

```
<A HREF="mailto:emailuser@host">Nume</a>
```

Un alt atribut important al acestui element este reprezentat de **TARGET**, care specifică locul în care se va deschide URL-ul. Valorile pe care le poate lua acest atribut sunt:

- **_blank** – URL-ul țintă se va deschide într-o nouă fereastră (este implicit, deci poate fi și nespecificat)
- **_top** – URL-ul țintă se va deschide în fereastra curentă
- **_self** – URL-ul țintă se va deschide în același cadru din care s-a dat click
- **_parent** – URL-ul țintă se va deschide în cadrul părinte (dar despre cadre vom vorbi ceva mai târziu)

6.4. Aplicație practică

Pentru a pune în practică cele prezentate în acest capitol, vom crea un nou fișier numit **test5.html** care va conține următoarele linii de cod:

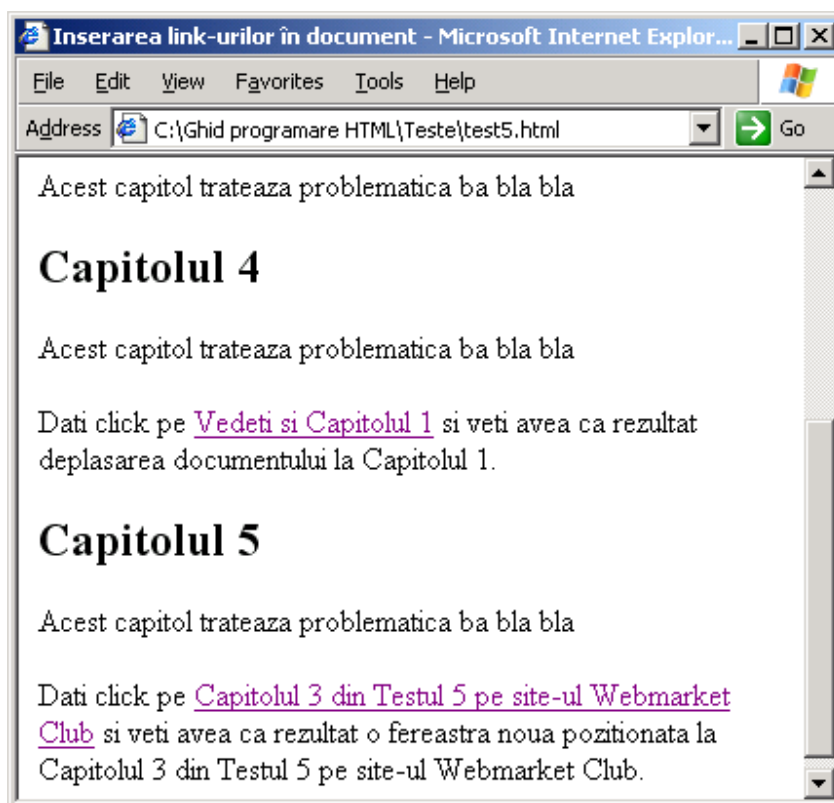
```
<HTML>  
<HEAD>  
  <TITLE>Inserarea link-urilor în document</TITLE>  
</HEAD>  
<BODY>  
  <H2><A name="C1">Capitolul 1</A></H2>  
  <P>Acest capitol trateaza problematica ba bla bla</P>  
  <H2><A name="C2">Capitolul 2</A></H2>  
  <P>Acest capitol trateaza problematica ba bla bla</P>  
  <H2><A name="C3">Capitolul 3</A></H2>  
  <P>Acest capitol trateaza problematica ba bla bla</P>  
  <H2><A name="C4">Capitolul 4</A></H2>
```

```

<P>Acest capitol trateaza problematica ba bla bla</P>
<P>Dati click pe <A href="#C1">Vedeti si Capitolul 1</A> si veti avea ca
rezultat deplasarea documentului la Capitolul 1.</P>
<H2><A name="C5">Capitolul 5</A></H2>
<P>Acest capitol trateaza problematica ba bla bla</P>
<P>Dati click pe <A href="http://html.webmarketclub.com/teste/test5.html#C3"
target="_blank">Capitolul 3 din Testul 5 pe site-ul Webmarket Club</A> si veti avea ca
rezultat o fereastră noua poziționată la Capitolul 3 din Testul 5 pe site-ul Webmarket
Club.</P>
</BODY>
</HTML>

```

Rezultatul afișat va fi următorul:



CAP. 7. FOLOSIREA IMAGINILOR

7.1. Tipuri de imagini

Deoarece nu toate tipurile de imagini sunt suportate de browsere, vom proceda mai întâi la o trecere în revistă a principalelor tipuri ce pot fi utilizate în documentele web. Singurele tipuri care pot fi utilizate fără restricții în browsere sunt următoarele:

GIF: Graphic Interchange Format, sunt imagini care au maximum 256 de culori. Formatul GIF este cel mai bun pentru redarea imaginilor de tipul logo-uri, pictograme și butoane. Este formatul cel mai răspândit pe Internet.

O categorie aparte a acestui format o constituie fișierele animate de tip GIF care contribuie la obținerea unor efecte deosebite.

JPG, JPEG: Joint Photographic Expert Group, sunt imagini care pot avea până la 16,7 milioane de culori. Formatul JPEG este recomandat pentru redarea fotografiilor și a altor imagini care necesită o claritate deosebită.

Mai există câteva formate care pot fi utilizate, dar cu anumite rezerve, întrucât nu vor fi recunoscute în toate browserele, fiind recomandată convertirea lor cu programe specializate (Adobe PhotoShop, Corel PhotoPaint etc.) în formatele prezentate anterior. Aceste formate sunt:

PNG: Portable Network Graphics

BMP: MS Windows BitMaP

TIFF: Tagged Image File Format

PCX: PC Paintbrush Format

La alegerea unei imagini trebuie avut în vedere faptul că cele de dimensiuni mari (ne referim la dimensiunea fișierului, nu la suprafața imaginii) vor încetini în mod considerabil încărcarea unui document web. O pagină web care conține multe imagini se va încărca mult mai greu decât o pagină web care conține mai mult text și mai puține imagini. De aceea trebuie să realizați un echilibru între rapiditatea de încărcare și design-ul site-ului.

7.2. Inserarea imaginilor

Dacă v-ați hotărât să inserați o imagine într-un document web, va trebui să folosiți tag-ul ****. Prezența tag-ului de închidere **** este opțională.

Spre deosebire de alte tag-uri, la folosirea tag-ului **** este **obligatorie** specificarea atributului **<SRC>** care stabilește sursa (source) imaginii ce trebuie încărcată, și este **recomandată** folosirea atributului **<ALT>**, care stabilește textul ce descrie imaginea și funcționează ca un tag (alternative). Este afișat atunci când vizitatorul deplasează cursorul mouse-ului deasupra imaginii sau când sunt probleme cu încărcarea imaginii.

Vom exemplifica cele de mai sus folosind următorul cod:

<HTML>

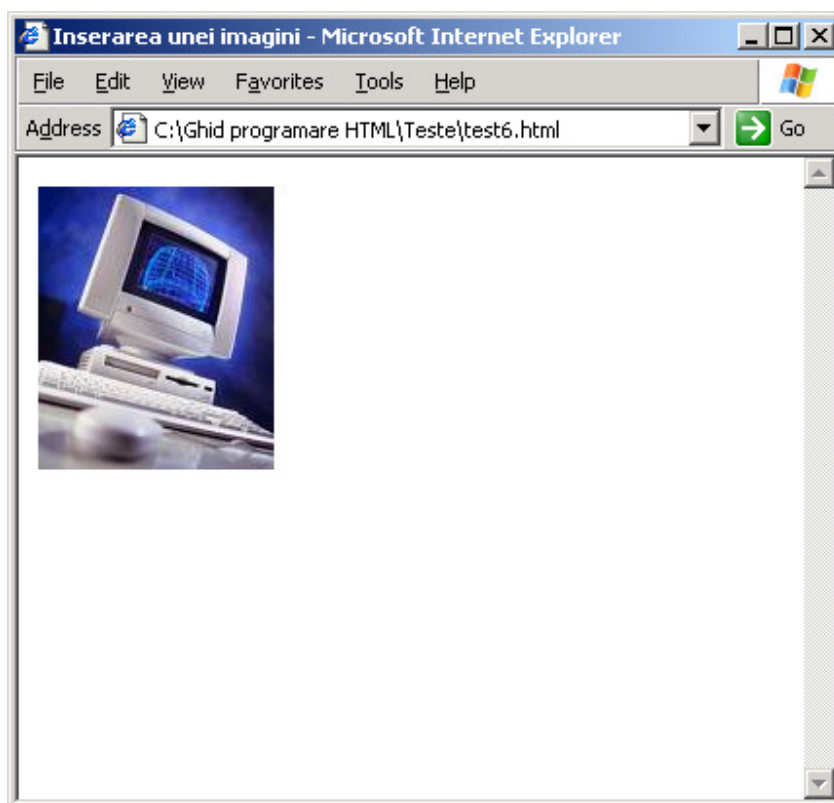
<HEAD>

```
<TITLE>Inserarea unei imagini</TITLE>
</HEAD>
<BODY>
  <IMG SRC="computer.jpg" ALT="Un computer in imagine">
</BODY>
</HTML>
```

Dacă nu aveți un fișier de tip imagine (.gif sau .jpg) pe care să-l folosiți ca exemplu, puteți descărca fișierul nostru dând click [aici](#).

Notă: Fiți atenți la calea utilizată la inserarea fișierului imagine! În acest exemplu fișierul computer.jpg este salvat în același director ca și fișierul test6.html.

Rezultatul afișat va fi următorul:



Rezultatul va fi identic și dacă folosiți imaginea existentă pe site-ul nostru, în condițiile în care îi specificați calea exactă și aveți o conexiune internet permanentă, astfel:

```
<HTML>
<HEAD>
  <TITLE>Inserarea unei imagini</TITLE>
</HEAD>
<BODY>
```

```

<IMG SRC="http://html.webmarketclub.com/teste/computer.jpg" ALT="Un
computer in imagine">
</BODY>
</HTML>

```

7.3. Formatarea imaginilor

Formatarea imaginilor se realizează utilizând celelalte atribute opționale care definesc acest tag, și anume:

Atribut	Valoare	Descriere
align	top bottom middle left right	Specifică cum va fi aliniată imaginea în raport cu textul ce o înconjoară.
border	pixels	Definește o bordură în jurul unei imagini.
height	pixels	Definește înălțimea unei imagini.
hspace	pixels	Definește spațiul liber din părțile laterale ale unei imagini.
ismap	URL	Definește imaginea de tip “server-side image map”.
longdesc	URL	Adresa URL către un document care conține o descriere lungă a imaginii.
usemap	URL	Definește imaginea de tip “client-side image map”. Se folosește în combinație cu tag-urile <map> și <area>.
vspace	pixels	Definește spațiul liber din părțile de sus și de jos ale unei imagini.
width	pixels	Definește lățimea unei imagini.

7.4. Aplicație practică

În continuare vom prezenta modul de inserare și de formatare a unor imagini, astfel că vom crea un nou fișier numit **test6.html** care va conține următoarele linii de cod:

```

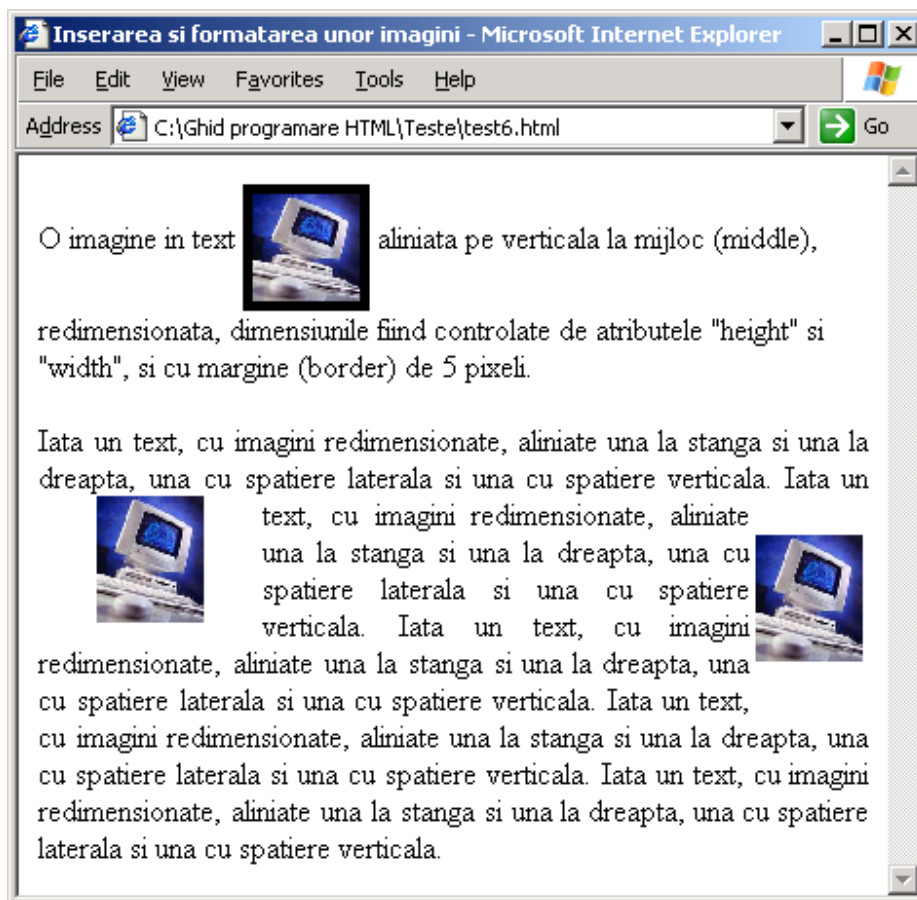
<HTML>
<HEAD>
  <TITLE>Inserarea si formatarea unor imagini</TITLE>
</HEAD>
<BODY>
  <P>O imagine in text

```

*** aliniata pe verticala la mijloc (middle), redimensionata, dimensiunile fiind controlate de attributele "height" si "width", si cu margine (border) de 5 pixeli.</P>***

***<P ALIGN="justify">Iata un text, cu imagini redimensionate, aliniate una la stanga si una la dreapta, una cu spatiere laterala si una cu spatiere verticala. Iata un text, cu imagini redimensionate, aliniate una la stanga si una la dreapta, una cu spatiere laterala si una cu spatiere verticala. Iata un text, cu imagini redimensionate, aliniate una la stanga si una la dreapta, una cu spatiere laterala si una cu spatiere verticala. Iata un text, cu imagini redimensionate, aliniate una la stanga si una la dreapta, una cu spatiere laterala si una cu spatiere verticala. Iata un text, cu imagini redimensionate, aliniate una la stanga si una la dreapta, una cu spatiere laterala si una cu spatiere verticala.</P>
</BODY>
</HTML>***

Rezultatul afişat va fi următorul:



7.5. Folosirea imaginilor ca link-uri

A devenit o modă înlocuirea link-urilor de tip text cu link-uri de tip imagine sau pictogramă (icon). De exemplu, a devenit o practică uzuală înlocuirea textului "next" cu o pictogramă simbolizând o săgeată orientată spre dreapta sau cu o pictogramă reprezentând o mână cu arătătorul indicând spre dreapta.

În mod obișnuit, o imagine plasată în interiorul unei ancore (adică între tag-urile `<A>` și `</A>`) devine o parte a conținutului ancorei, astfel încât un click pe imaginea respectivă va avea același rezultat ca și un click pe textul unei ancore.

Browser-ele pot modifica imaginile într-un mod speciale (de obicei adăugându-le o bordură) pentru a indica faptul că reprezintă un link.

Sintaxa folosirii acestor tag-uri este următoarea:

```
<A HREF="document.html"> <IMG SRC="image.gif" BORDER="pixeli"  
WIDTH="pixeli" HEIGHT="pixeli" ALT="Text alternativ"> </A>
```

Standardul HTML oferă posibilitatea să includeți mai multe link-uri într-o singură imagine, astfel încât dacă dați click în zone diferite ale imaginii, vor fi accesate link-uri diferite. Imaginile de acest tip sunt cunoscute sub numele de *imagini-hartă* (image maps).

Există două posibilități de a crea astfel de imagini-hartă:

1. prin folosirea atributului **ismap** în tag-ul sunt create imaginile de tip *server-side image map*

2. prin folosirea atributului **usemap** în tag-ul sunt create imaginile de tip *client-side image map*

Deoarece pentru imaginile de tip server-side image map este necesar accesul la serverul pe care se află stocată imaginea, precum și faptul că sunt generate întârzieri la încărcarea paginilor, ne vom ocupa numai de imaginile client-side image map care nu prezintă aceste neajunsuri.

7.6. Imagini de tip client-side image map

Aceste imagini sunt create prin folosirea atributului **usemap** în elementul și definite prin elementele speciale marcate de tag-urile <MAP> și <AREA> (se folosesc numai împreună, elementul <AREA> fiind întotdeauna inclus în elementul <MAP>).

Tag-ul <MAP> definește o imagine ca fiind de tip hartă, iar tag-ul <AREA> specifică coordonatele zonelor imaginii-hartă, precum și link-urile corespunzătoare fiecărei zone.

Atributul usemap preia valoarea atributelor **id** sau **name** (în funcție de browser), stabilite în tag-ul <MAP>.

Trebuie amintit faptul că doar tag-ul <MAP> necesită tag-ul de închidere </MAP>.

Atributele opționale care definesc tag-ul <MAP> sunt:

Atribut	Valoare	Descriere
id	nume_hartă	Atribuie o denumire imaginii-hartă
name	nume_hartă	Este identic cu id , se folosește doar pentru a asigura compatibilitatea cu anumite browsere

Atributele opționale care definesc tag-ul <AREA> sunt:

Atribut	Valoare	Descriere
alt	text	Specifică o descriere alternativă a zonei active
shape	rect sau rectangle circ sau circle poly sau polygon	Definește tipul de zonă (dreptunghi, cerc sau poligon)
coords	dacă shape="rect" atunci coords="stânga,sus,dreapta,jos" dacă shape="circ" atunci coords="centru_x,centru_y,rază" dacă shape="poly" atunci coords="x1,y1,x2,y2,...,xn,yn"	Specifică coordonatele zonei active
href	URL	Specifică URL-ul țintă al zonei
nohref	true false	Exclude o zonă din hartă
target	_blank _parent _self _top	• URL-ul țintă se va deschide într-o nouă fereastră (este implicit, deci poate fi și nespecificat) • URL-ul țintă se va deschide în fereastra curentă • URL-ul țintă se va deschide în același cadru din care s-a dat click • URL-ul țintă se va deschide în cadrul părinte

7.7. Aplicație practică

În continuare vom prezenta modul de utilizare a unei imagini-hartă, astfel că vom crea un nou fișier numit **test7.html** care va conține următoarele linii de cod:

```

<HTML>
<HEAD>
  <TITLE>Utilizarea unei imagini-harta</TITLE>
</HEAD>
<BODY>
  <P>Dati click pe una din formele geometrice:</P>
  <IMG SRC="http://html.webmarketclub.com/teste/imagini-harta.gif"
  WIDTH="268" HEIGHT="95" ALT="Imagine de tip harta"
  USEMAP="#harta">
  <MAP ID="harta" NAME="harta">
    <AREA SHAPE="rect" COORDS="0,0,65,95"
    HREF="http://html.webmarketclub.com/teste/dreptunghi.html"
    TARGET="_blank" ALT="Dreptunghi">

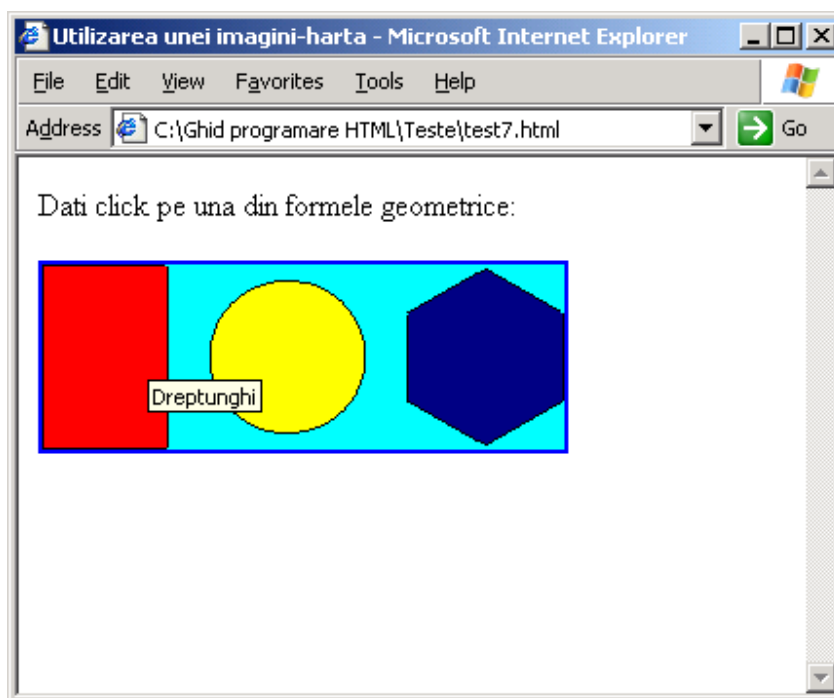
```

```

<AREA SHAPE="circle" COORDS="125,47,40"
HREF="http://html.webmarketclub.com/teste/cerc.html" TARGET="_blank"
ALT="Cerc">
<AREA SHAPE="poly"
COORDS="185,25,224,1,224,1,268,25,268,72,226,93,185,71"
HREF="http://html.webmarketclub.com/teste/hexagon.html"
TARGET="_blank" ALT="Hexagon">
</MAP>
</BODY>
</HTML>

```

Rezultatul afișat va fi următorul:



Notă: Pentru aflarea coordonatelor formelor geometrice am folosit un program specializat "Mapedit" (este un program shareware, poate fi folosit pentru testare numai 30 de zile și poate fi descărcat la adresa <http://www.boutell.com/mapedit>)

CAP. 8. TABELE

8.1. Ce sunt tabelele?

Un tabel reprezintă un ansamblu de date (cifre, simboluri, informații) afișate în celule sistematizate într-o structură formată din rânduri și coloane.

În afară de rolul de afișa date, în HTML au și un rol important în realizarea aspectului general al documentului web.

Celula unui tabel poate conține oricare din elementele HTML acceptate în secțiunea BODY, aceasta incluzând texte, imagini, formulare, antete și chiar alte tabele.

Pentru a marca prezența unui tabel se folosește tag-ul de deschidere **<TABLE>**. Acesta necesită tag-ul de închidere **</TABLE>**.

Elementul **<TABLE>** are următoarele sub-elemente:

- Rândul tabelului (Table Row) marcat de tag-urile **<TR>** **</TR>** (prezența sa este obligatorie)
- Datele tabelului (Table Data) marcate de tag-urile **<TD>** **</TD>** (prezența sa este obligatorie)
- Antetul rândurilor și coloanelor (Table Header) marcat de tag-urile **<TH>** **</TH>**
- Coloana (Column) marcată de tag-ul **<COL>** ce nu necesită tag de închidere
- Descrierea tabelului marcată de tag-urile **<CAPTION>** **</CAPTION>**
- Secțiunea antetului tabelului (Table Header Section) marcată de tag-urile **<THEAD>** **</THEAD>**
- Secțiunea corpului tabelului (Table Body Section) marcată de tag-urile **<TBODY>** **</TBODY>**
- Secțiunea subsolului tabelului (Table Footer Section) marcată de tag-urile **<TFOOT>** **</TFOOT>**

Formatarea tabelului se face începând de sus în jos și de la stânga la dreapta, fiind definite, secvențial, antetul și datele fiecărei celule din fiecare coloană, progresând în jos, rând cu rând.

Un exemplu conținând cod HTML referitor la tabele, este următorul:

```
<TABLE BORDER="1">  
<TR>  
  <TH>Antetul Coloanei 1 </TH>  
  <TH>Antetul Coloanei 2 </TH>  
</TR>  
<TR>  
  <TD>Rândul 1 - Coloana 1</TD>  
  <TD>Rândul 1 - Coloana 2</TD>  
</TR>  
<TR>  
  <TD>Rândul 2 - Coloana 1</TD>  
  <TD>Rândul 2 - Coloana 2</TD>  
</TR>  
<TR>
```

```

<TD>Rândul 3 - Coloana 1</TD>
<TD>Rândul 3 - Coloana 2</TD>
</TR>
</TABLE>

```

Rezultatul va fi următorul:

Antetul Coloanei 1	Antetul Coloanei 2
Rândul 1 - Coloana 1	Rândul 1 - Coloana 2
Rândul 2 - Coloana 1	Rândul 2 - Coloana 2
Rândul 3 - Coloana 1	Rândul 3 - Coloana 2

8.2. Formatarea tabelelor

Așa cum am amintit și anterior, tabelele sunt de mare ajutor în realizarea aspectului general al unui document web. Ele vă permit punerea în evidență a anumitor obiecte din document și sunt foarte utile la crearea formularelor, aspect pe care îl vom aborda în capitolul următor.

Principalele atribute care vă permit formatarea elementului <TABLE> sunt următoarele:

Atribut	Valoare	Descriere
align	left center right	Precizează modul de aliniere al tabelului în document.
background	nume fișier	O imagine folosită ca fundal pentru tabel.
bgcolor	rgb(x,x,x) #xxxxxx colorname	Precizează culoarea fundalului tabelului.
border	pixels	Definește grosimea bordurii din tabel (dacă nu se dorește bordură, aceasta ia valoarea 0 sau nu se specifică)
cellpadding	pixels %	Specifică spațiul dintre marginile fiecărei celule și conținutul său.
cellspacing	pixels %	Specifică spațiul dintre celule.
frame	void above below vsides	Are efect numai când atributul "border"> 0 și specifică care părți ale bordurii sunt vizibile: - nici o parte - numai partea de sus - numai partea de jos - numai părțile de sus și de jos

	hsides lhs rhs box border	- numai părțile laterale - numai partea din stânga - numai partea din dreapta - toate cele patru părți - toate cele patru părți
rules	none groups rows cols all	Specifică asupra căror grupe de celule se aplică regulile referitoare la bordură: - nici o regulă (este implicit) - regulile se aplică grupelor de rânduri sau de coloane - regulile se aplică numai rândurilor - regulile se aplică numai coloanelor - regulile se aplică tuturor celulelor
summary	text	Specifică o descriere mai detaliată a tabelului
width	% pixels	Specifică lățimea ocupată de tabel în cadrul paginii web (în cazul în care nu se specifică, lățimea va depinde de conținutul tabelului)

Să presupunem că dorim să afișăm în documentul nostru orarul desfășurării anumitor cursuri. Tabelul va conține zilele de desfășurare (ca antete de coloană), orele de desfășurare (ca antete de rânduri) și denumirea cursurilor.

Pentru aceasta vom crea **test8.html** care va conține următoarele linii de cod:

```

<HTML>
<HEAD>
  <TITLE>Formatarea tabelelor</TITLE>
</HEAD>
<BODY>
  <TABLE><CAPTION>Orar (fara bordura aliniat la stanga) </CAPTION>
  <TR>
    <TH>&nbsp;</TH>
    <TH>Luni</TH>
    <TH>Miercuri</TH>
    <TH>Vineri</TH>
  </TR>
  <TR>
    <TH>10-12</TH>
    <TD>Curs HTML</TD>
    <TD>Curs PHP</TD>
    <TD>Curs JavaScript</TD>
  </TR>
  <TR>
    <TH>12-14</TH>
    <TD>Curs JavaScript</TD>
    <TD>Curs PHP</TD>
    <TD>Curs HTML</TD>
  </TR>

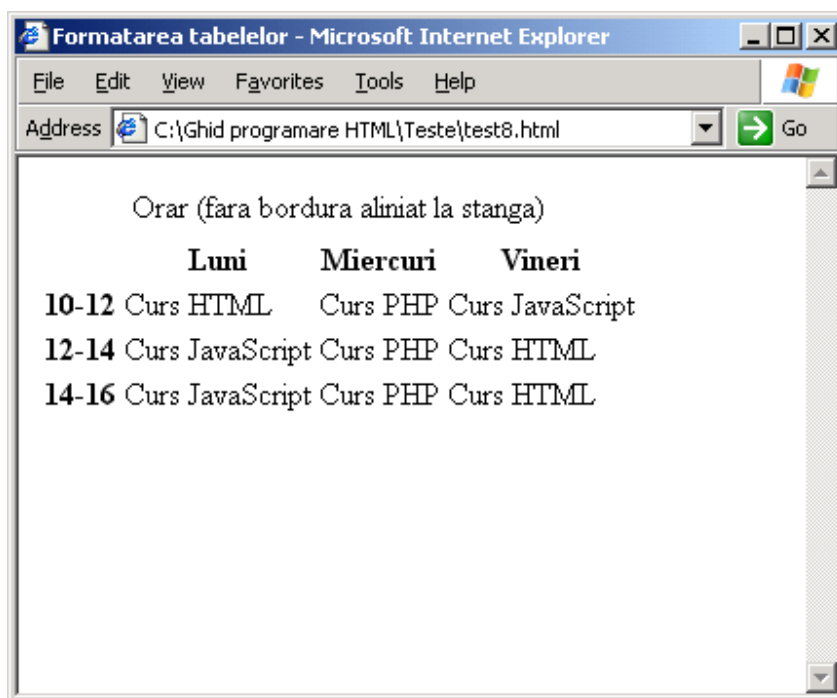
```

```

</TR>
<TR>
  <TH>14-16</TH>
  <TD>Curs JavaScript</TD>
  <TD>Curs PHP</TD>
  <TD>Curs HTML</TD>
</TR>
</TABLE>
</BODY>
</HTML>

```

Rezultatul va fi următorul:

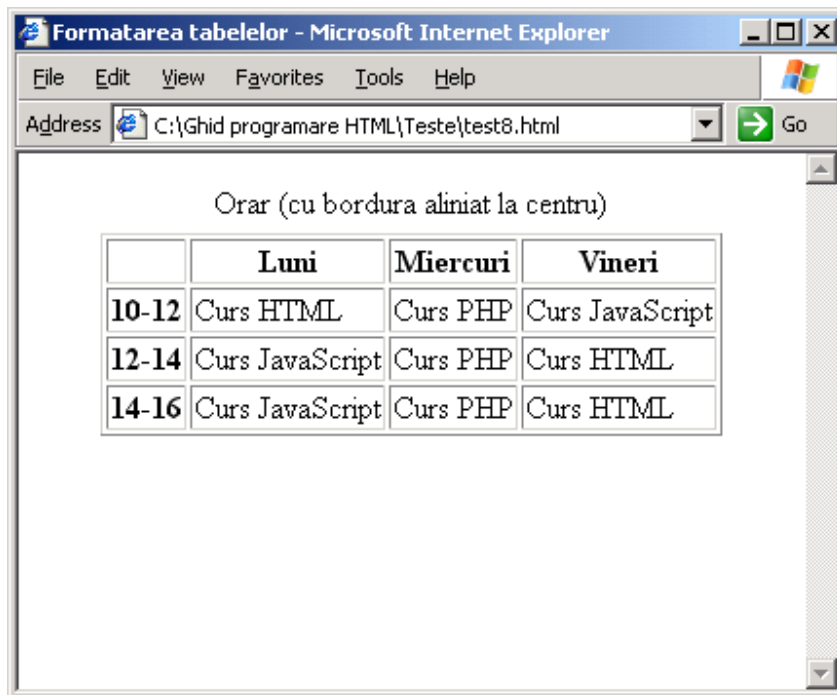


Notă: *HTML nu acceptă celule goale, așa că a trebuit să creăm o celulă care conține caracterul *

Vom adăuga tabelului nostru o bordură și îl vom alinia pe centrul paginii. Pentru aceasta vom adăuga elementului <TABLE> următoarele atribute:

<TABLE border="1" align="center"><CAPTION>Orar (cu bordura aliniat la centru) </CAPTION>

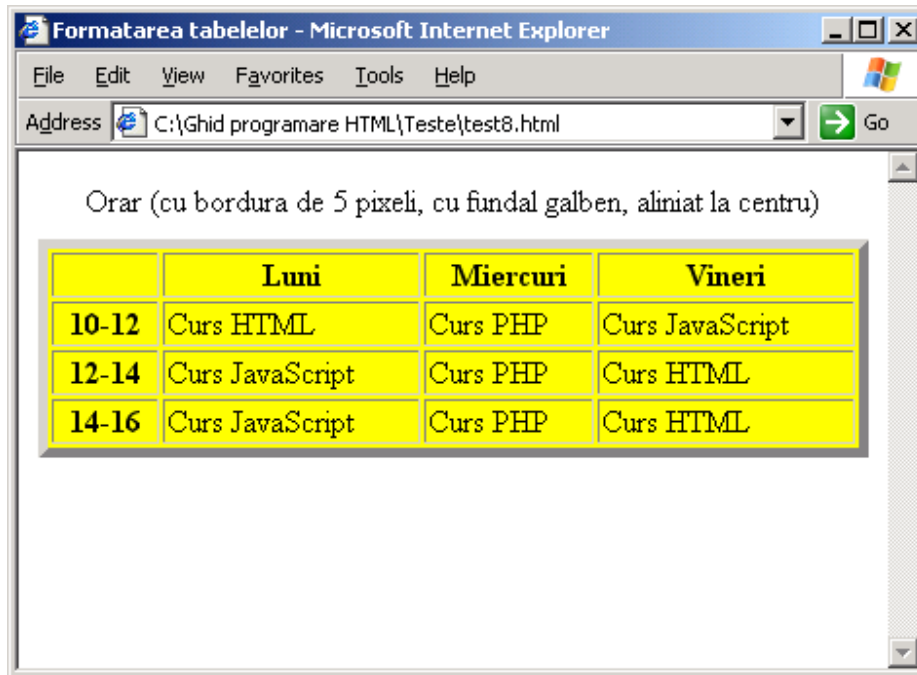
și rezultatul va fi următorul:



În continuare vom îngroșa puțin bordura, vom adăuga un fundal galben tabelului și îi vom mări lățimea, astfel încât să acopere întreaga fereastră. Aceasta se realizează astfel:

```
<TABLE border="5" align="center" bgcolor="yellow" width="100%">  
<CAPTION>Orar (cu bordura de 5 pixeli, cu fundal galben, aliniat la centru)  
</CAPTION>
```

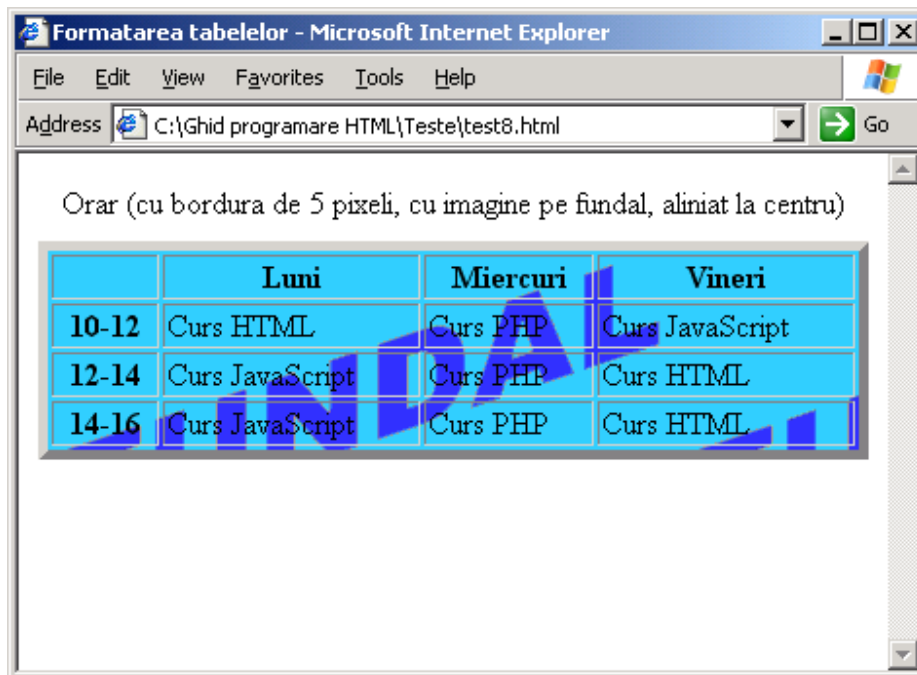
iar rezultatul va fi următorul:



În loc de o singură culoare putem folosi o imagine ca fundal. Pentru aceasta vom înlocui atributul **bgcolor** cu **background** și vom specifica denumirea fișierului de tip imagine:

```
<TABLE border="5" align="center" background="fundal.gif"
width="100%"> <CAPTION>Orar (cu bordura de 5 pixeli, cu imagine pe fundal, aliniat
la centru) </CAPTION>
```

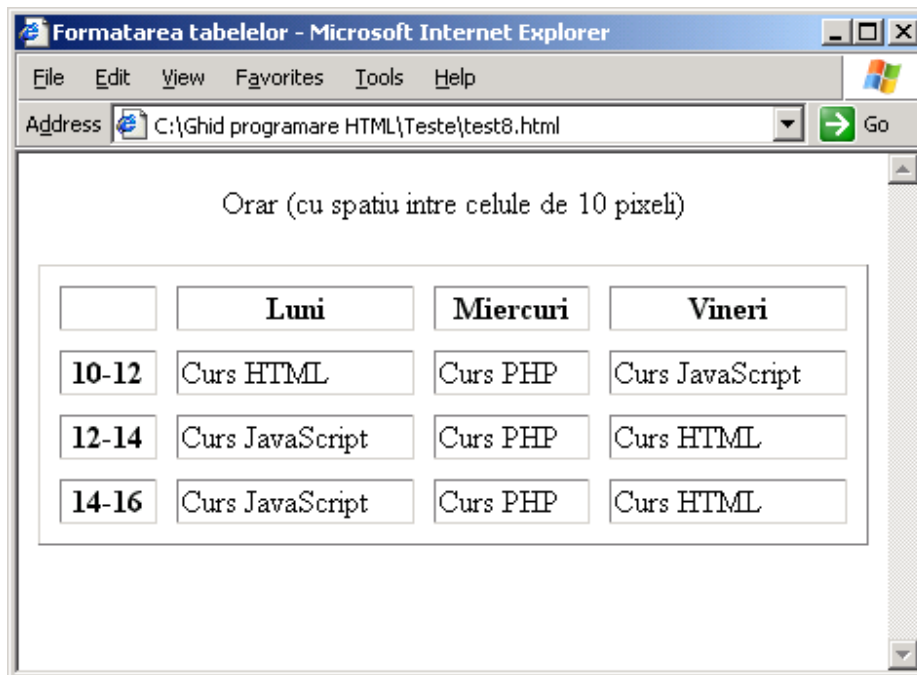
iar vom obține următorul rezultat:



Să ne ocupăm și de spațierea dintre celule dând atributului **cellspacing** valoarea 10 (implicit acesta are valoarea 0).

```
<TABLE border="1" align="center" cellspacing="10" width="100%">
<CAPTION>Orar (cu spatiu intre celule de 10 pixeli) </CAPTION>
```

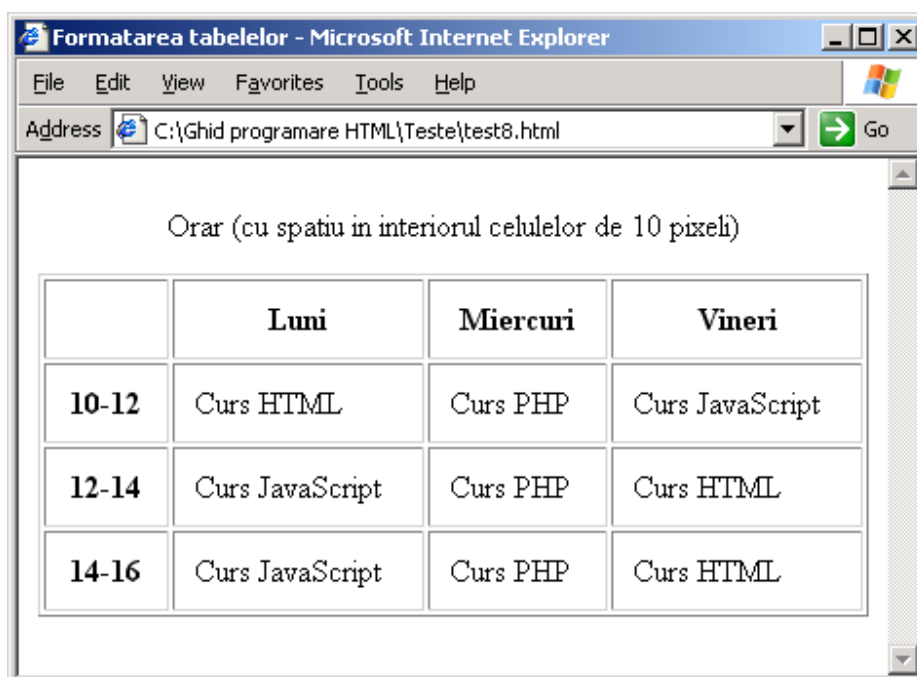
Rezultatul va fi următorul:



Mărirea sau micșorarea spațiului dintre pereții unei celule și conținutul său se realizează prin alocarea unor valori corespunzătoare atributului **cellpadding**, astfel:

<TABLE border="1" align="center" cellpadding="10" width="100%">
<CAPTION>Orar (cu spatiu in interiorul celulelor de 10 pixeli) </CAPTION>

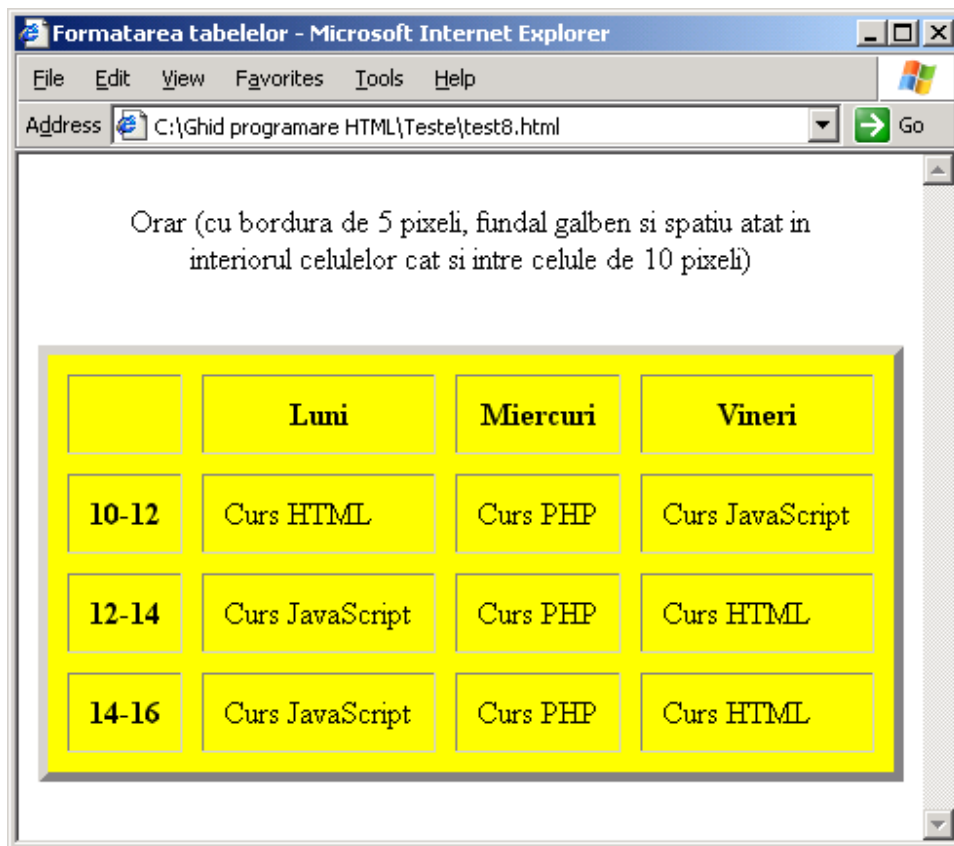
iar rezultatul va fi următorul:



Iată și o combinație a atributelor folosite până acum:

<TABLE border="5" align="center" bgcolor="yellow" cellpadding="10" cellspacing="10" width="100%"> <CAPTION>Orar (cu bordura de 5 pixeli, fundal galben si spatiu atat in interiorul celulelor cat si intre celule de 10 pixeli) </CAPTION>

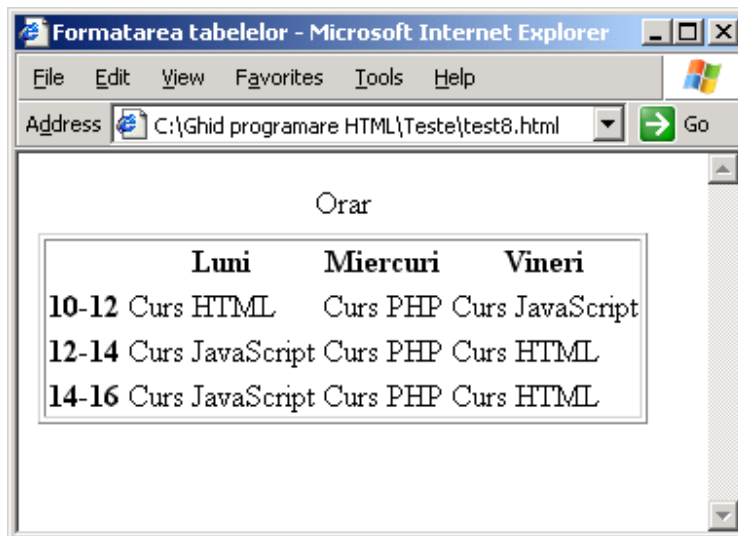
ce va avea următorul rezultat:



În continuare vom vedea ce efecte se obțin prin folosirea atributelor **frame** și **rules**. Aceste două attribute sunt strâns legate de folosirea atributului **border**, prin urmare vor fi utilizate toate 3 concomitent. De exemplu:

***<TABLE border="1" rules="none" frame="box">
<CAPTION>Orar</CAPTION>***

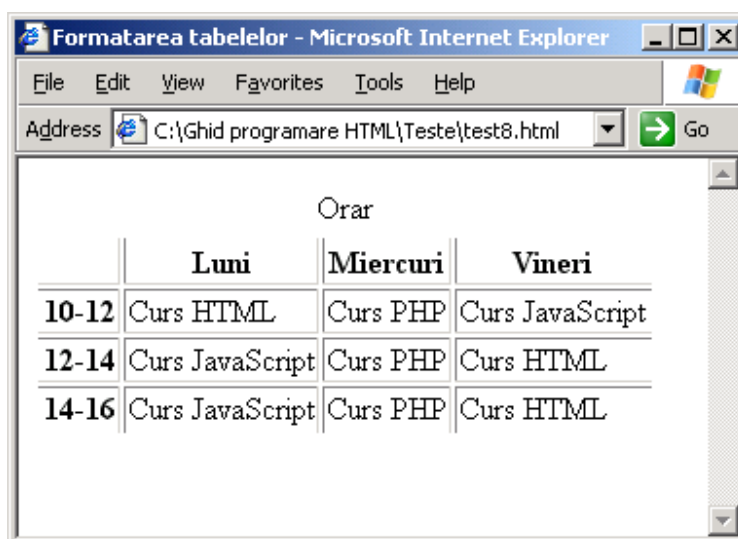
va avea ca efect obținerea unui tabel fără bordură în interior:



Pentru a obține un tabel fără bordură exterioară, ci numai cu cele interioare, vom da următoarele valori:

`<TABLE border="1" rules="all" frame="void">`

și vom rezulta:



În funcție de necesități, puteți combina aceste atribute pentru a obține rezultatele dorite.

8.3. Sub-elementele tabelor

În continuare vom analiza fiecare sub-element al tabelor și atributele sale.

Așa cum ați putut vedea mai sus, fiecare tag **<TR>** desemnează un rând într-un tabel. Fiecare rând poate conține unul sau mai multe elemente **<TD>** sau **<TH>**.

Din punct de vedere tehnic, tag-ul **</TR>** este opțional, dar folosirea lui vă ajută la eliminarea confuziilor.

Principalele atribute care vă permit formatarea acestui element sunt următoarele:

Atribut	Valoare	Descriere
Align	left center right justify char	Precizează modul de aliniere pe orizontală a conținutului.
Valign	top middle bottom baseline	Precizează modul de aliniere pe verticală a conținutului.
bgcolor	rgb(x,x,x) #xxxxxx colorname	Precizează culoarea fundalului unui rând întreg.

Pentru exemplificare, vom folosi fișierul creat anterior, și anume **test8.html**, în care vom include:

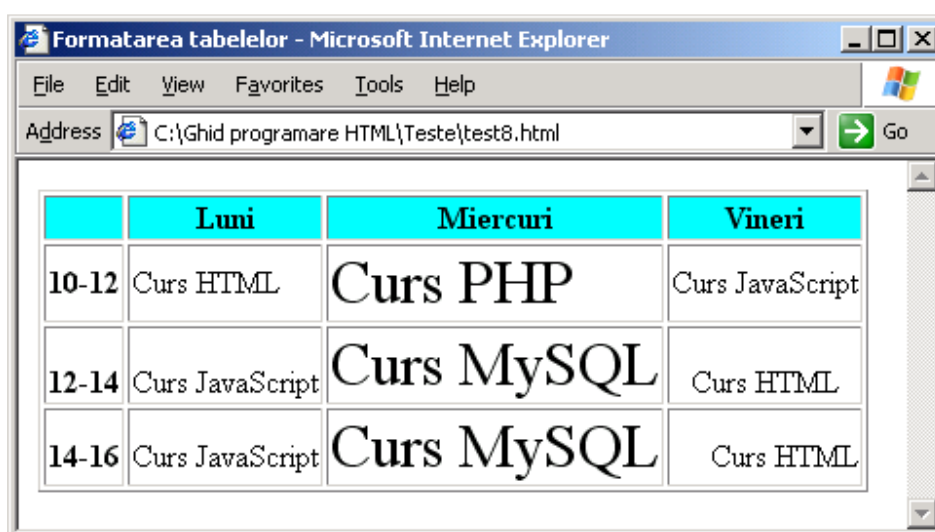
```
<HTML>
<HEAD>
  <TITLE>Formatarea tabelor</TITLE>
</HEAD>
<BODY>
  <TABLE BORDER="1">
    <TR BGCOLOR="cyan">
      <TH>&nbsp;</TH>
      <TH>Luni</TH>
      <TH>Miercuri</TH>
      <TH>Vineri</TH>
    </TR>
    <TR ALIGN="left" VALIGN="middle">
      <TH>10-12</TH>
      <TD>Curs HTML</TD>
      <TD><FONT SIZE="6">Curs PHP</FONT></TD>
      <TD>Curs JavaScript</TD>
    </TR>
    <TR ALIGN="center" VALIGN="bottom">
      <TH>12-14</TH>
      <TD>Curs JavaScript</TD>
      <TD><FONT SIZE="6">Curs MySQL</FONT></TD>
      <TD>Curs HTML</TD>
```

```

</TR>
<TR ALIGN="right" VALIGN="baseline">
  <TH>14-16</TH>
  <TD>Curs JavaScript</TD>
  <TD><FONT SIZE="6">Curs MySQL</FONT></TD>
  <TD>Curs HTML</TD>
</TR>
</TABLE>
</BODY>
</HTML>

```

Rezultatul va fi următorul:



The screenshot shows a web browser window titled "Formatarea tabelelor - Microsoft Internet Explorer". The address bar shows the file path "C:\Ghid programare HTML\Teste\test8.html". The displayed table has the following content:

	Luni	Miercuri	Vineri
10-12	Curs HTML	Curs PHP	Curs JavaScript
12-14	Curs JavaScript	Curs MySQL	Curs HTML
14-16	Curs JavaScript	Curs MySQL	Curs HTML

Dacă celelalte atribute produc efecte vizibile, consider că trebuie oferite câteva explicații suplimentare privind deosebirea între alinierea verticală **BOTTOM** și **BASELINE**.

Alinierea **BOTTOM** se face după cea mai de jos cotă a caracterelor, în vreme ce alinierea **BASELINE** are ca punct de reper baza textului, după cum se poate vedea în imaginea de mai jos:

	Luni	Miercuri	Vineri
10-12	Curs HTML	Curs PHP	Curs JavaScript
12-14	Curs JavaScript	<u>Curs MySQL</u>	aliniere BOTTOM
14-16	Curs JavaScript	<u>Curs MySQL</u>	aliniere BASELINE

Mai departe ne vom ocupa de tag-ul **<TD>** care desemnează o celulă ce conține date într-un tabel.

Principalele atribute care vă permit formatarea acestui elementului sunt următoarele:

Atribut	Valoare	Descriere
align	left center right justify char	Precizează modul de aliniere pe orizontală a conținutului.
valign	top middle bottom baseline	Precizează modul de aliniere pe verticală a conținutului.
bgcolor	rgb(x,x,x) #xxxxxx colorname	Precizează culoarea fundalului celulei.
background	nume fișier	O imagine folosită ca fundal pentru celulă.
height	% pixels	Specifică înălțimea ocupată de celulă în cadrul tabelului (în cazul în care nu se specifică, înălțimea va depinde de conținutul celulei)
width	% pixels	Specifică lățimea ocupată de celulă în cadrul tabelului (în cazul în care nu se specifică, lățimea va depinde de conținutul celulei)
colspan	număr	Indică numărul de celule din tabel care vor fi unite pe o coloană.
rowspan	număr	Indică numărul de celule din tabel care vor fi unite pe un rând.
nowrap	nowrap	Se folosește pentru a dezactiva ajustarea textului la lățimea unei celule (funcționează numai dacă n-a fost definită valoarea atributului "width").

Pentru exemplificare, vom folosi fișierul creat anterior, și anume **test8.html**, în care vom include:

```
<HTML>
<HEAD>
  <TITLE>Formatarea tabelelor</TITLE>
</HEAD>
<BODY>
  <TABLE BORDER="1">
    <TR>
      <TH>&nbsp;</TH>
      <TH>Luni</TH>
      <TH>Miercuri</TH>
      <TH>Vineri</TH>
```

```

</TR>
<TR>
  <TH WIDTH="40">10-12</TH>
  <TD BGCOLOR="cyan">Curs HTML</TD>
  <TD>Curs JavaScript</TD>
  <TD ROWSPAN="3" BACKGROUND="fundal.gif"
NOWRAP="nowrap"><B>Oricât de mare ar fi continutul acestei celule, <BR>textul nu
va trece pe rândul următor fără folosirea<BR>
tag-ului BR care execută ruperea rândului. </TD>
</TR>
<TR>
  <TH>12-14</TH>
  <TD COLSPAN="2" ALIGN="center" HEIGHT="60">Curs
JavaScript</TD>
</TR>
<TR>
  <TH>14-16</TH>
  <TD ALIGN="right">Curs JavaScript</TD>
  <TD BGCOLOR="orange">Curs HTML</TD>
</TR>
</TABLE>
</BODY>
</HTML>

```

Rezultatul va fi următorul:



Notă: Există numeroase probleme legate de modul în care browsere diferite interpretează atributul width, rezultatul afișat putând diferi foarte mult de la caz la caz. Dacă totuși considerați utilă folosirea acestui atribut, recomandarea noastră e să-l

definiți pentru fiecare celulă în parte concomitent cu definirea acestuia pentru întregul tabel, astfel încât suma valorilor alocate celulelor să corespundă cu valoarea alocată tabelului.

Referitor la tag-ul **<TH>** vă vom spune doar că funcționează aproape identic cu tag-ul **<TD>**, având aceleași atribute, cu excepția faptului că **<TH>** specifică unei celule că este antet pentru un rând sau pentru o coloană, conținutul de tip text fiind automat îngroșat.

Sub-elementul Coloană marcat de tag-ul **<COL>** definește valorile atributelor pentru una sau mai multe coloane ale unui tabel. Se poate folosi numai în interiorul tag-ului **<TABLE>**, dar înainte de oricare dintre tag-urile **<TR>**, **<THEAD>** sau **<TBODY>**.

Fiecare tag **<COL>** utilizat definește proprietățile unei singure coloane, exceptând cazul în care nu este utilizat atributul **span** pentru a indica mai multe coloane. Deci, primul tag **<COL>** definește proprietățile primei coloane, al doilea tag **<COL>** definește proprietățile celei de-a doua coloane etc.

Atributele care permit formatarea sub-elementului **<COL>** sunt următoarele:

Atribut	Valoare	Descriere
align	left center right justify	Precizează modul de aliniere al datelor în coloană.
valign	top middle bottom baseline	Precizează modul de aliniere pe verticală a conținutului în coloană.
bgcolor	rgb(x,x,x) #xxxxxx colorname	Precizează culoarea fundalului coloanei.
width	% pixels	Specifică lățimea ocupată de coloană în cadrul tabelului.
span	număr	Precizează numărul de coloane cărora le sunt definite proprietățile.

Pentru exemplificare, vom folosi fișierul **test8.html**, care va conține următoarele:

```

<HTML>
<HEAD>
  <TITLE>Formatarea tabelelor</TITLE>
</HEAD>
<BODY>
  <TABLE border="1">
    <COL>
      <COL valign="top" bgcolor="yellow" span="2">
        <COL align="right" width="120">

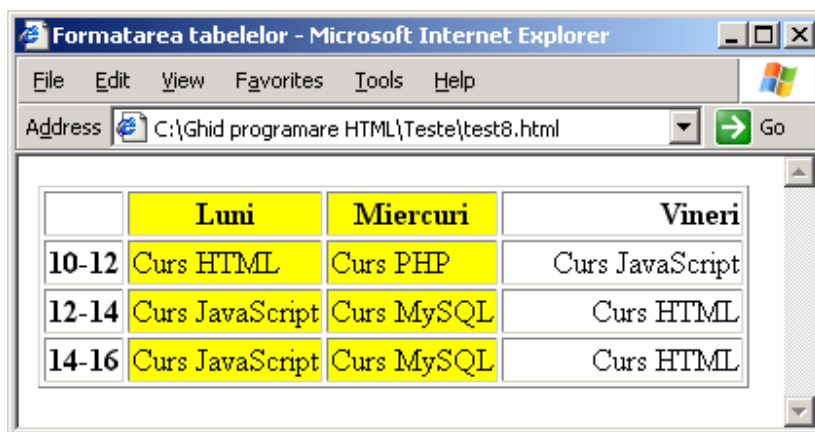
```

```

<TR>
  <TH>&nbsp;</TH>
  <TH>Luni</TH>
  <TH>Miercuri</TH>
  <TH>Vineri</TH>
</TR>
<TR>
  <TH>10-12</TH>
  <TD>Curs HTML</TD>
  <TD>Curs PHP</TD>
  <TD>Curs JavaScript</TD>
</TR>
<TR>
  <TH>12-14</TH>
  <TD>Curs JavaScript</TD>
  <TD>Curs MySQL</TD>
  <TD>Curs HTML</TD>
</TR>
<TR>
  <TH>14-16</TH>
  <TD>Curs JavaScript</TD>
  <TD>Curs MySQL</TD>
  <TD>Curs HTML</TD>
</TR>
</TABLE>
</BODY>
</HTML>

```

Rezultatul va fi următorul:



The screenshot shows a web browser window titled "Formatarea tabelor - Microsoft Internet Explorer". The address bar shows the file path "C:\Ghid programare HTML\Teste\test8.html". The rendered table is as follows:

	Luni	Miercuri	Vineri
10-12	Curs HTML	Curs PHP	Curs JavaScript
12-14	Curs JavaScript	Curs MySQL	Curs HTML
14-16	Curs JavaScript	Curs MySQL	Curs HTML

Notă: Observați folosirea tag-ului <COL> fără nici un atribut pentru prima coloană. În cazul în care dorim să folosim acest tag numai pentru o anumită coloană, atunci trebuie folosit pentru toate coloanele precedente, chiar dacă nu produce nici o modificare a formătărilor conținutului.

Ne vom referi puțin și la tag-ul <CAPTION> pe care l-am folosit chiar de la primul exemplu. Vom preciza faptul că are un singur atribut, și anume **align** care poate lua valorile *left*, *right*, *top* și *bottom* care indică locul în care va fi poziționată descrierea tabelului.

De asemenea, nu vom insista nici asupra elementelor <THEAD>, <TBODY> și <TFOOT> datorită folosirii lor extrem de rar, în principal din cauza incompatibilității cu diverse browsere, dar și a faptului că folosirea lor nu aduce foarte multe avantaje utilizatorului.

Atributele lor sunt următoarele:

Atribut	Valoare	Descriere
align	left center right justify char	Precizează modul de aliniere pe orizontală a conținutului.
valign	top middle bottom baseline	Precizează modul de aliniere pe verticală a conținutului.
bgcolor	rgb(x,x,x) #xxxxxx colorname	Precizează culoarea fundalului unei secțiuni întregi.

Pentru exemplificare am ales următoarele linii de cod:

```
<HTML>
<HEAD>
  <TITLE>Formatarea tabelelor</TITLE>
</HEAD>
<BODY>
  <TABLE border="1">
    <THEAD valign="top" bgcolor="yellow">
      <TR>
        <TH>&nbsp;</TH>
        <TH>Luni</TH>
        <TH>Miercuri</TH>
        <TH>Vineri</TH>
      </TR>
    </THEAD>
    <TBODY align="center" bgcolor="cyan">
      <TR>
        <TH>10-12</TH>
        <TD>Curs HTML</TD>
        <TD>Curs PHP</TD>
```

```

    <TD>Curs JavaScript</TD>
</TR>
<TR>
    <TH>12-14</TH>
    <TD>Curs JavaScript</TD>
    <TD>Curs MySQL</TD>
    <TD>Curs HTML</TD>
</TR>
<TR>
    <TH>14-16</TH>
    <TD>Curs JavaScript</TD>
    <TD>Curs MySQL</TD>
    <TD>Curs HTML</TD>
</TR>
</TBODY>
<TFOOT>
<TR>
    <TH colspan="4">Prof. Ion Popescu</TH>
</TR>
</TFOOT>
</TABLE>

</BODY>
</HTML>

```

Rezultatul va fi următorul:

	Luni	Miercuri	Vineri
10-12	Curs HTML	Curs PHP	Curs JavaScript
12-14	Curs JavaScript	Curs MySQL	Curs HTML
14-16	Curs JavaScript	Curs MySQL	Curs HTML
Prof. Ion Popescu			

8.4. Aplicație practică

În continuare vom pune în practică cele prezentate în acest capitol, combinând (atât cât se poate) elementele prezentate:

```

<HTML>
<HEAD>
<TITLE>Formatarea tabelelor</TITLE>
</HEAD>
<BODY>
<TABLE border="5" cellpadding="5" cellspacing="15"
bgcolor="#F5F5DC"> <CAPTION align="bottom"> Orarul cursurilor pentru
programatori Web</CAPTION>
<COL align="center" bgcolor="cornsilk">
<COL align="center" width="120" span="3" bgcolor="bisque">
<THEAD><TR bgcolor="#FFFFFFEE"><TH colspan="4">Curs programatori
Web </TH></TR></THEAD>
<TBODY><TR bgcolor="cornsilk">
<TH>Ora</TH>
<TH>Luni</TH>
<TH>Miercuri</TH>
<TH>Vineri</TH>
</TR>
<TR>
<TH>10-12</TH>
<TD>Curs HTML</TD>
<TD>Curs PHP</TD>
<TD>Curs JavaScript</TD>
</TR>
<TR>
<TH>12-14</TH>
<TD>Curs JavaScript</TD>
<TD rowspan="2">Curs MySQL</TD>
<TD>Curs PHP</TD>
</TR>
<TR>
<TH height="50">14-16</TH>
<TD>Curs PHP</TD>
<TD>Curs HTML</TD>
</TR></TBODY>
<TFOOT><TR bgcolor="#FFFFFFEE"><TD colspan="4"
align="right"><FONT color="#8B0000"><B>Conferentiar prof. Ion Popescu
</B></FONT></TD></TR></TFOOT>
</TABLE>
</BODY>
</HTML>

```

Rezultatul va fi următorul:

Formatarea tabelelor - Microsoft Internet Explorer

File Edit View Favorites Tools Help

Address  C:\Ghid programare HTML\Teste\test8.html  Go

Curs programatori Web			
Ora	Luni	Miercuri	Vineri
10-12	Curs HTML	Curs PHP	Curs JavaScript
12-14	Curs JavaScript	Curs MySQL	Curs PHP
14-16	Curs PHP		Curs HTML

Conferentiar prof. Ion Popescu

Orarul cursurilor pentru programatori Web

CAP. 9. CADRE

9.1. Noțiuni generale

Cadrele vă permit divizarea ferestrei browser-ului în mai multe zone dreptunghiulare, dispuse pe rânduri și/sau coloane, fiecare dintre acestea conținând un document web diferit. Fiecare astfel de zonă poartă denumirea de **cadru (frame)**. Cadrele sunt foarte populare deoarece reprezintă cea mai simplă modalitate de a realiza un document ce conține părți statice și părți variabile (care se pot schimba).

O pagină ce conține cadre reprezintă de fapt un document cu mai multe pagini, fiecare cadru conținând o pagină web. În afară de acestea, mai există un document web care descrie modul în care va fi împărțită pagina în cadre.

De exemplu, pentru a crea un document cu două cadre (unul în stânga și unul în dreapta) este nevoie de trei documente web. În imaginea de mai jos sunt reprezentate aceste documente așa cum sunt ele afișate de browsere.

În acest exemplu, **doc.html** reprezintă documentul care descrie modul în care fereastra browser-ului va fi divizată în cele două zone dreptunghiulare marcate prin cadrele ce conțin celelalte două documente web.

9.2. Structura unei pagini ce definește cadre

Definirea cadrelor se realizează prin folosirea elementului `<FRAMESET>` ce se plasează într-un document web în locul elementului `<BODY>`.

Elementul <FRAMESET> descrie modul în care fereastra browser-ului va fi împărțită, mai exact stabilește numărul și dimensiunile cadrelor (zonelor dreptunghiulare) dispuse pe rânduri și/sau coloane. Necesită tag de închidere </FRAMESET>

Atributele acestui element sunt legate de existența rândurilor sau coloanelor, precum și de modul de delimitare a acestora unele de altele:

Atribut	Valoare	Descriere
cols	pixels % *	Definește numărul și mărimea coloanelor într-un document.
rows	pixels % *	Definește numărul și mărimea rândurilor într-un document.
frameborder	yes / no 1 / 0	Stabilește dacă există sau nu bordură pentru cadre
framespacing sau border	pixels	Stabilește grosimea bordurii cadrelor (framespacing în Internet Explorer și border în Netscape).
bordercolor	rgb(x,x,x) #xxxxxx colorname	Precizează culoarea bordurii cadrelor.

În interiorul elementului <FRAMESET> este elementul <FRAME>, câte unul pentru fiecare cadru, care are rolul de a stabili documentele corespunzătoare fiecăruia dintre cadre.

Pentru exemplificare vor crea un document cu două cadre dispuse vertical. În acest sens vor trebui create cele trei documente corespunzătoare.

Vom crea fișierul **doc1.html**, corespunzător cadrului 1, cu următoarea structură:

```
<HTML>
<HEAD>
<TITLE>Documentul 1</TITLE>
</HEAD>
<BODY BGCOLOR="cyan">
<H3>Documentul 1 in cadrul 1</H3>
</BODY>
</HTML>
```

Vom crea și fișierul **doc2.html**, corespunzător cadrului 2, cu următoarea structură:

```
<HTML>
<HEAD>
<TITLE>Documentul 2</TITLE>
</HEAD>
<BODY BGCOLOR="yellow">
<H3>Documentul 2 in cadrul 2</H3>
```

```
</BODY>
</HTML>
```

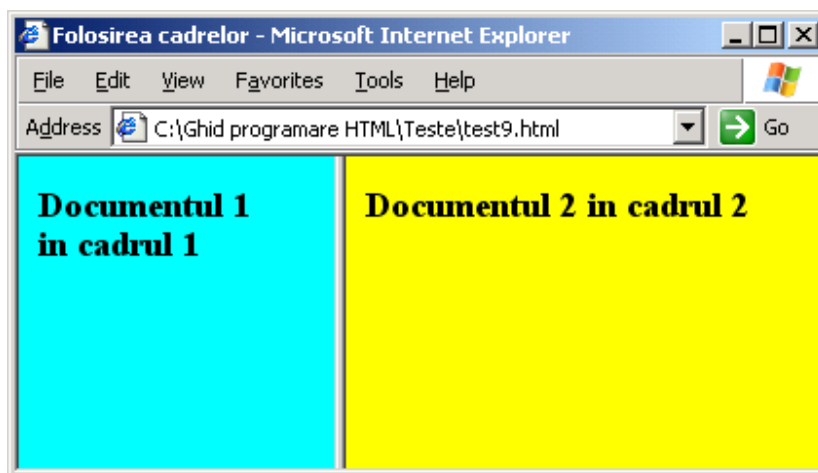
În final vom crea fișierul **test9.html**, care va stabili dispunerea cadrelor, cu următoarea structură:

```
<HTML>
<HEAD>
<TITLE>Folosirea cadrelor</TITLE>
</HEAD>

<FRAMESET COLS="40%,60%">
  <FRAME SRC="doc1.html">
  <FRAME SRC="doc2.html">
</FRAMESET>

</HTML>
```

Rezultatul afișat va fi următorul:



XXXXX

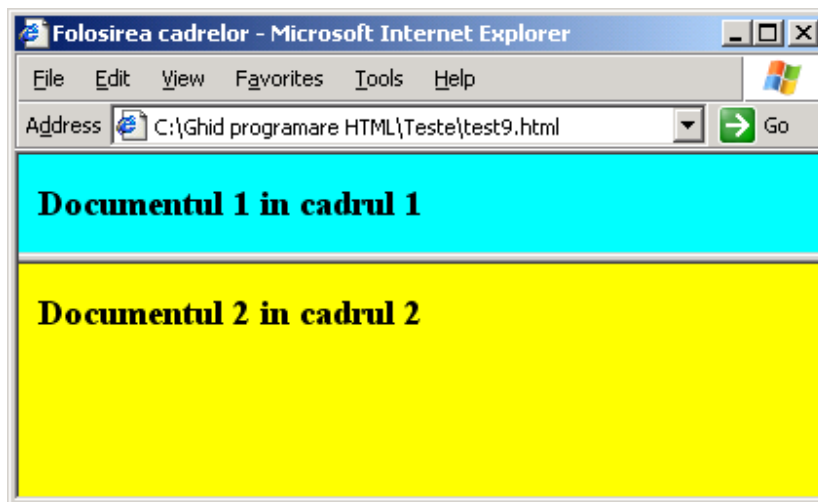
În cazul în care dorim ca cele două cadre să fie dispuse orizontal, fișierul **test9.html** va avea următoarea structură:

```
<HTML>
<HEAD>
<TITLE>Folosirea cadrelor</TITLE>
</HEAD>

<FRAMESET ROWS="30%,70%">
  <FRAME SRC="doc1.html">
  <FRAME SRC="doc2.html">
</FRAMESET>
```

</HTML>

iar rezultatul afișat va fi următorul:



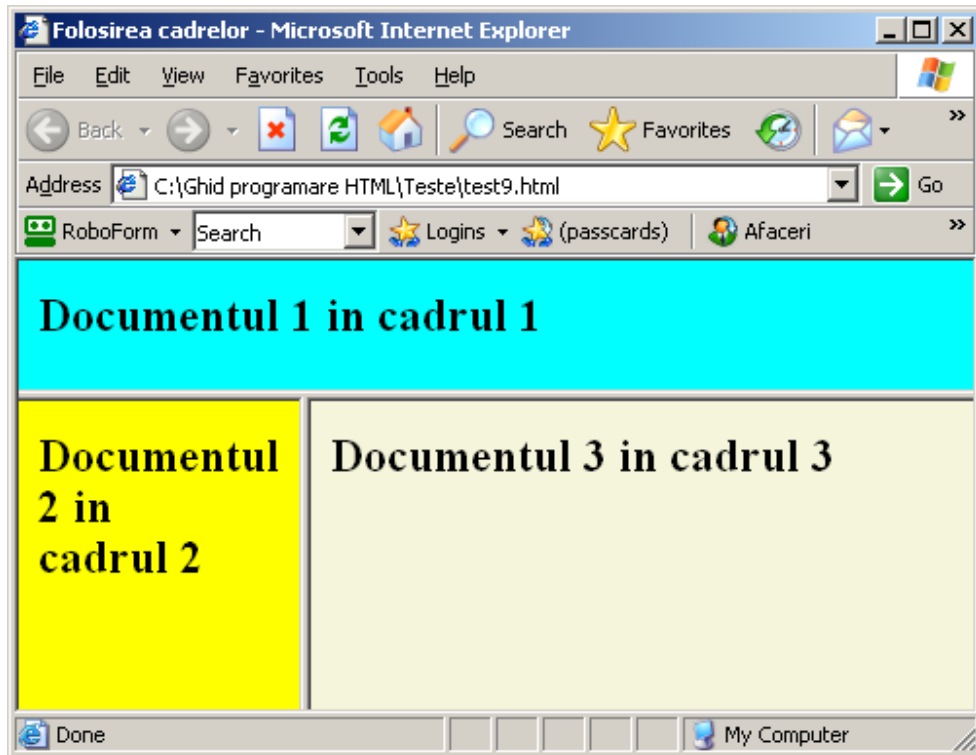
Pentru realizarea unei structuri combinate, cu trei cadre, dintre care unul pe orizontală și două pe verticală, vom crea fișierul **doc3.html**, corespunzător cadrului 3, cu următoarea structură:

```
<HTML>  
<HEAD>  
<TITLE>Documentul 3</TITLE>  
</HEAD>  
<BODY BGCOLOR="beige">  
  <H3>Documentul 3 in cadrul 3</H3>  
</BODY>  
</HTML>
```

iar fișierul **test9.html** va avea următoarea structură:

```
<HTML>  
<HEAD>  
<TITLE>Folosirea cadrelor</TITLE>  
</HEAD>  
<FRAMESET ROWS="30%,70%">  
  <FRAME SRC="doc1.html">  
    <FRAMESET COLS="30%,70%">  
      <FRAME SRC="doc2.html">  
      <FRAME SRC="doc3.html">  
    </FRAMESET>  
  </FRAMESET>  
</HTML>
```

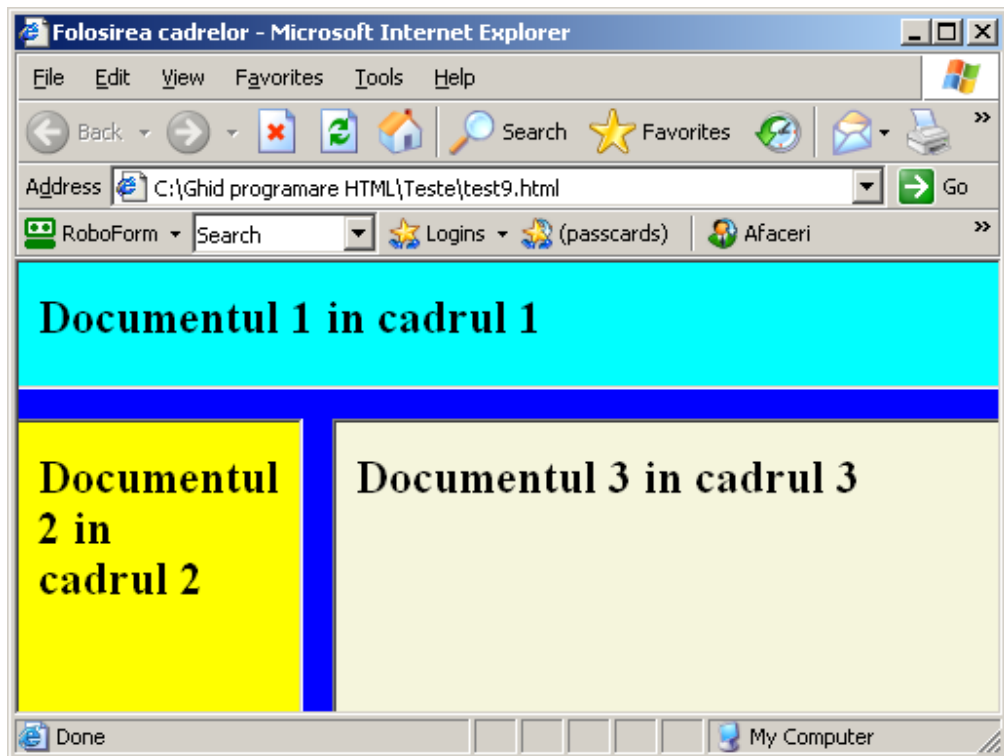
Rezultatul afișat va fi următorul:



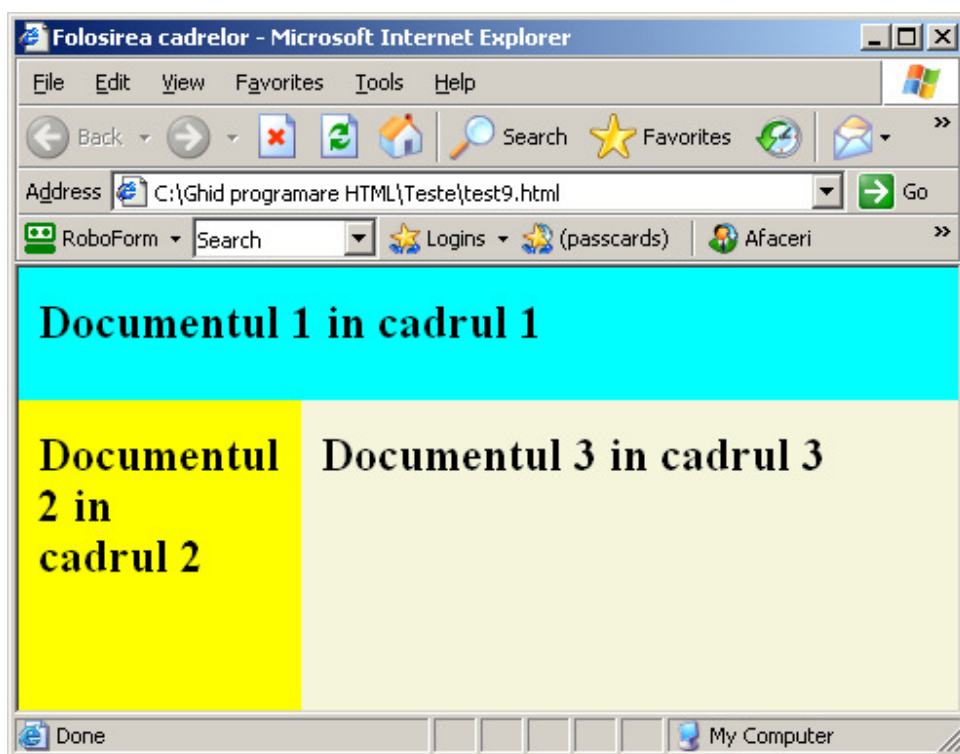
În final vom analiza și atributele ce definesc bordura, astfel că vom folosi următoarele exemple:

```
<HTML>
<HEAD>
<TITLE>Folosirea cadrelor</TITLE>
</HEAD>
<FRAMESET ROWS="30%,70%" FRAMEBORDER="yes"
FRAMESPACING="15" BORDERCOLOR="blue">
  <FRAME SRC="doc1.html">
    <FRAMESET COLS="30%,70%">
      <FRAME SRC="doc2.html">
        <FRAME SRC="doc3.html">
    </FRAMESET>
  </FRAMESET>
</FRAMESET>
</HTML>
```

Rezultatul afișat va fi următorul:



Pentru eliminarea bordurilor, attributele *frameborder* și *framespacing* vor primi valoarea 0, iar rezultatul afișat va fi următorul:



9.3. Atributele cadrelor

Elementul <FRAME> definește fiecare cadru dintr-un document. Este obligatorie prezența acestui element în fiecare secțiune marcată de tag-ul <FRAMESET>. Nu necesită tag de închidere.

Atributele ce caracterizează acest element sunt următoarele:

Atribut	Valoare	Descriere
src	URL	Reprezintă link-ul către fișierul ce se va deschide în cadrul respectiv..
frameborder	yes / no 1 / 0	Stabilește dacă există sau nu bordură pentru cadrul respectiv.
longdesc	URL	Reprezintă link-ul către o descriere lungă folosită pentru browserele ce nu suportă cadre.
marginheight	pixels	Definește marginile de sus și de jos ale cadrului.
marginwidth	pixels	Definește marginile din stânga și din dreapta ale cadrului.
noresize	noresize	Când este setat, acest atribut împiedică ajustarea mărimii cadrului de către utilizatori.
scrolling	yes no auto	Determină prezența sau absența barei de derulare (scrollbar).
name	text	Definește denumirea cadrului.
target	_blank _self _parent _top	Specifică unde va deschis URL-ul țintă: ➤ _blank – într-o fereastră nouă; ➤ _self – în același cadru din care a fost accesat; ➤ _parent – în cadrul părinte; ➤ _top - direct în fereastra browser-ului, luând locul documentului existent.

Pentru a observa efectele acestor atribute, ne vom folosi de fișierul **test9.html** cu următoarea structură:

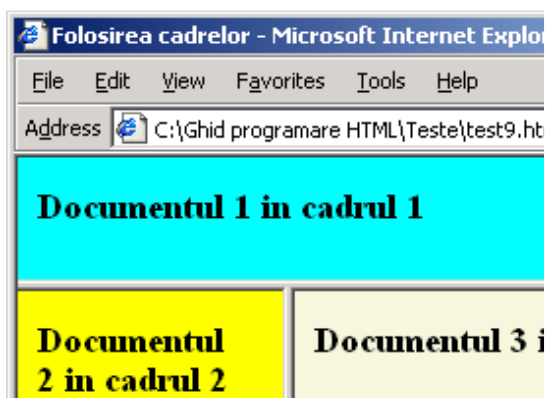
```
<HTML>
<HEAD>
<TITLE>Folosirea cadrelor</TITLE>
</HEAD>
<FRAMESET ROWS="30%,70%">
  <FRAME SRC="doc1.html">
    <FRAMESET COLS="30%,70%">
      <FRAME SRC="doc2.html">
        <FRAME SRC="doc3.html">
    </FRAMESET>
  </FRAME>
</FRAMESET>
```

```
</FRAMESET>
</HTML>
```

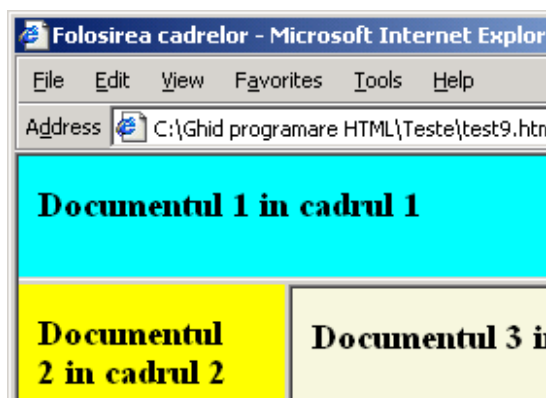
Atributul **src** a fost folosit în exemplele anterioare, prin urmare nu vom insista asupra lui.

Vom adăuga atributul **frameborder="no"** cadrului nr. 2 (`<frame src="doc2.html" frameborder="no">`) și veți observa diferența:

Cadrul 2 cu bordură

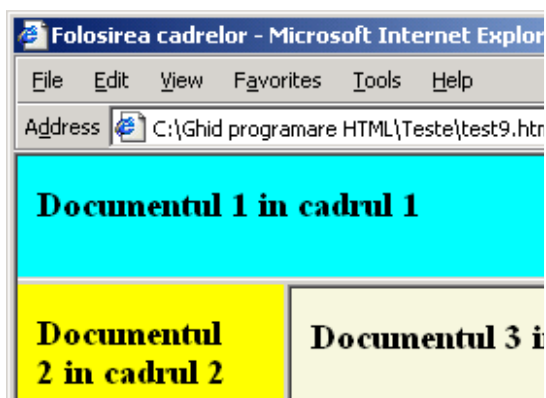


Cadrul 2 fără bordură

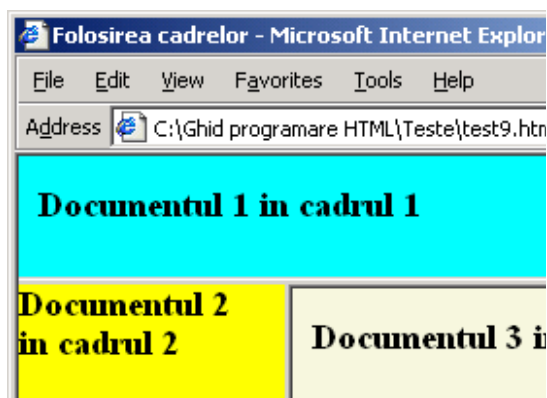


În continuare vom adăuga și atributul **marginheight** cadrului nr. 2 (`<frame src="doc2.html" frameborder="no" marginheight="0">`) pentru a pune în evidență modul în care se modifică marginile de sus și de jos ale cadrului:

Cadrul 2 cu margini implicite



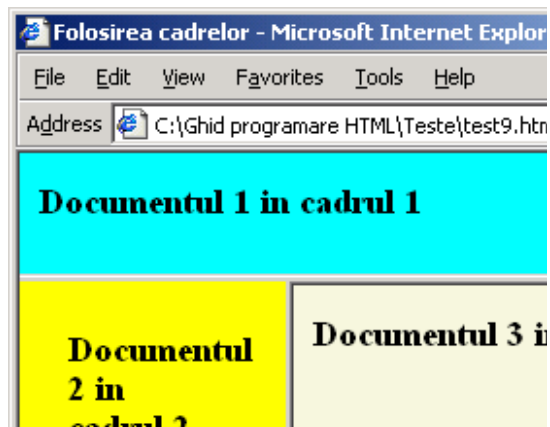
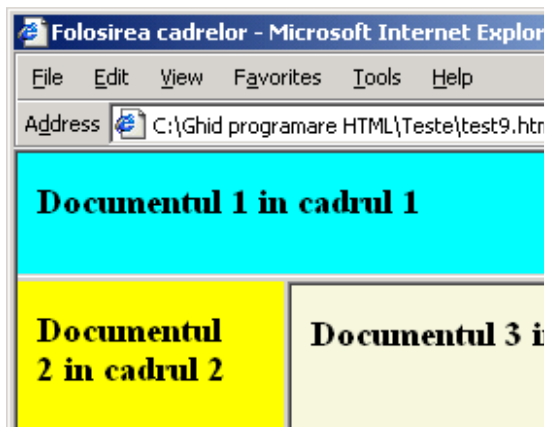
Cadrul 2 fără margini sus și jos



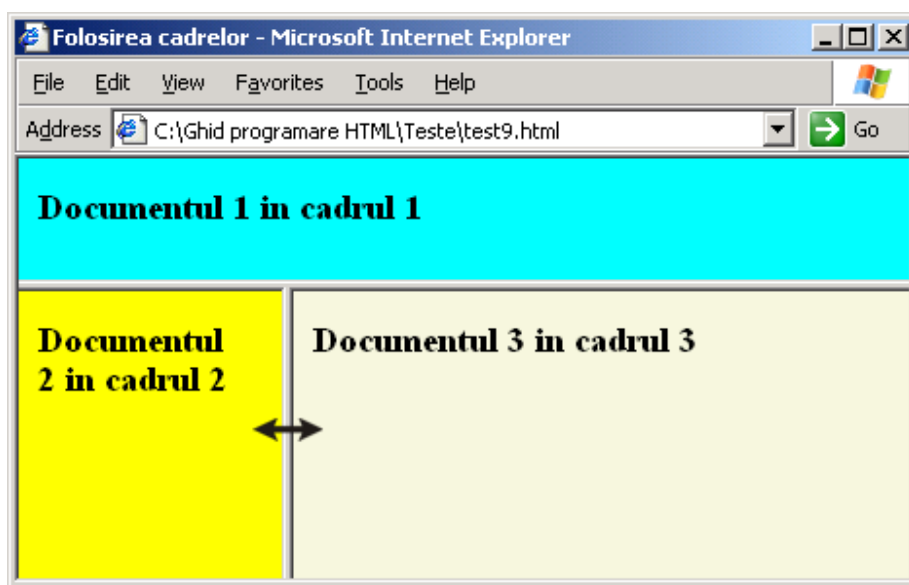
Adăugând și atributul **marginwidth** cadrului nr. 2 (`<frame src="doc2.html" frameborder="no" marginheight="25" marginwidth="25">`) vom pune în evidență modul în care se modifică marginile din stânga și din dreapta ale cadrului:

Cadrul 2 cu margini implicite

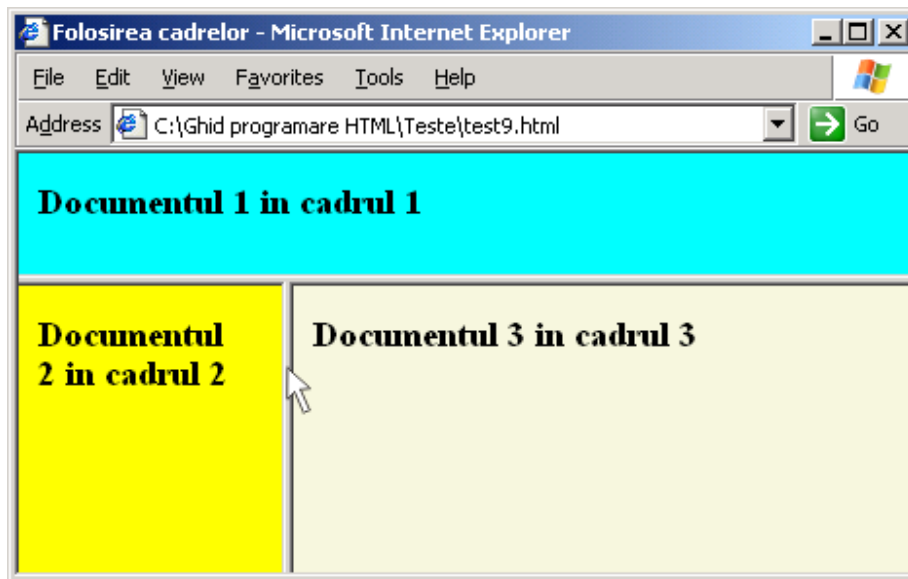
Cadrul 2 cu margini de 25 pixeli



Pentru a pune în evidență atributul **noresize**, mai întâi încercați să poziționați cursorul de la mouse pe bordura dintre cadre, după care, în momentul în care vă apare semnul " ↔ ", ținând apăsat pe butonul din stânga de la mouse, mutați bordura ce separă cadrele:

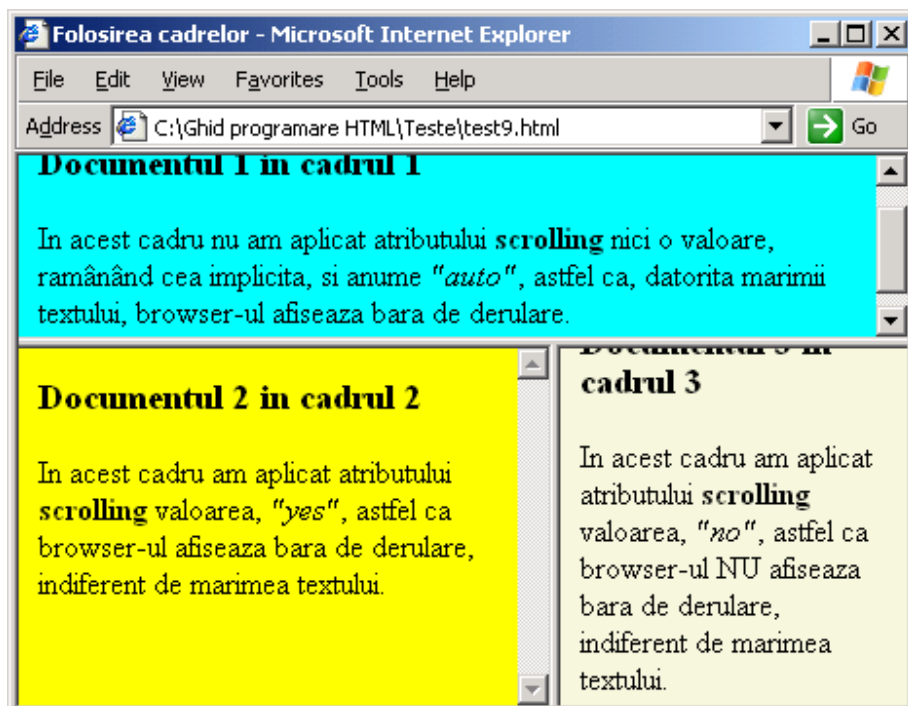


Aplicând atributul **noresize** (<frame src="doc2.html" noresize>), facilitatea de redimensionare a cadrelor va fi dezactivată, prin urmare nu va mai apărea semnul " ↔ ", ci doar cursorul normal al mouse-ului, ca în imaginea de mai jos.



Atributul **scrolling** vă permite afișarea barei de derulare în fiecare cadru. Valoarea implicită este “auto”, caz în care, în funcție de mărimea conținutului, browser-ul va decide dacă este necesară afișarea barei de derulare. În cazul atribuirii valorii “yes”, bara de derulare va fi afișată întotdeauna, iar în cazul atribuirii valorii “no”, bara de derulare nu va fi afișată deloc.

Pentru exemplificare, vom adăuga conținut celor trei documente ce populează cadrele, și anume doc1.html, doc2.html și doc2.html.



Prin folosirea atributului **name**, cadrelor le pot fi atribuite denumiri, ceea ce va poate ajuta la identificarea mai usoara a acestora.

9.4. Destinații

Când creai link-uri într-un document web ce conține cadre, este necesară folosirea unui atribut suplimentar denumit **target**. Acesta este corelat cu atributul **name** prezentat anterior.

Atributul **target** care poate apărea în următoarele elemente <A>, <AREA>, <BASE> și <FORM>.

Exemple:

```
<A HREF="file.htm" TARGET="nume_fereastră">  
<AREA SHAPE="RECT" COORDS="5,6,2,4" HREF="file.html"  
TARGET="nume_fereastră">  
<BASE TARGET="nume_fereastră">  
<FORM ACTION="cgi-bin/script" TARGET="nume_fereastră">
```

Valorile posibile pentru atributul **target** sunt:

- a) numele (valoarea atributului **name**) unei ferestre deschise de browser sau numele (valoarea atributului **name**) unui cadru încărcat de browser
- b) una din următoarele 4 ținte speciale, ce nu pot fi atribuite atributului **name**, fiind denumiri rezervate cu funcții specifice:
 - TARGET="_blank" (documentul va fi încărcat într-o fereastră nouă a browser-ului);
 - TARGET="_top" (documentul va fi încărcat direct în fereastra browser-ului, luând locul documentului existent);
 - TARGET="_parent" (documentul va fi încărcat în fereastra/setul de cadre părinte);
 - TARGET="_self" (documentul va fi încărcat în același cadru);

Pentru exemplificare, vom folosi fișierul **test9.html** cu următoarea structură:

```
<HTML>  
<HEAD>  
<TITLE>Folosirea cadrelor</TITLE>  
</HEAD>  
<FRAMESET ROWS="15%,85%">  
<FRAME SRC="doc1.html" NAME="sus">  
<FRAMESET COLS="70%,30%">  
<FRAME SRC="doc2.html">  
<FRAME SRC="doc3.html" NAME="dreapta">  
</FRAMESET>  
</FRAMESET>  
</HTML>
```

Vom folosi fișierul **doc1.html**, corespunzător cadrului 1, cu următoarea structură:

```
<HTML>
<HEAD>
  <TITLE>Documentul 1</TITLE>
</HEAD>
<BODY BGCOLOR="cyan">
  <H3>Documentul 1 in cadrul 1</H3>
</BODY>
</HTML>
```

În fișierul **doc2.html** vom insera link-uri către **doc4.html** cu destinații diferite, astfel:

```
<HTML>
<HEAD>
  <TITLE>Documentul 2</TITLE>
</HEAD>
<BODY BGCOLOR="yellow">
  <H3>Documentul 2 in cadrul 2</H3>
  <P><B>Exemplificare destinatii</B><BR>
  <A HREF="doc4.html" TARGET="dreapta"> Documentul 4 in cadrul din
dreapta</A><BR>
  <A HREF="doc4.html" TARGET="sus"> Documentul 4 in cadrul de
sus</A><BR>
  <A HREF="doc4.html" TARGET="_self"> Documentul 4 in acelasi cadru din
care este accesat link-ul (si anume in Cadrul 2)</A><BR>
  <A HREF="doc4.html" TARGET="_parent"> Documentul 4 in cadrul
parinte</A><BR>
  <A HREF="doc4.html" TARGET="_top"> Documentul 4 direct in browser (in
acest exemplu rezultatul este identic ca la optiunea precedenta)</A><BR>
  <A HREF="doc4.html" TARGET="_blank"> Documentul 4 intr-o fereastra
noua a browser-ului </A>
</BODY>
</HTML>
```

Fișierul **doc3.html**, corespunzător cadrului 3, va avea următoarea structură:

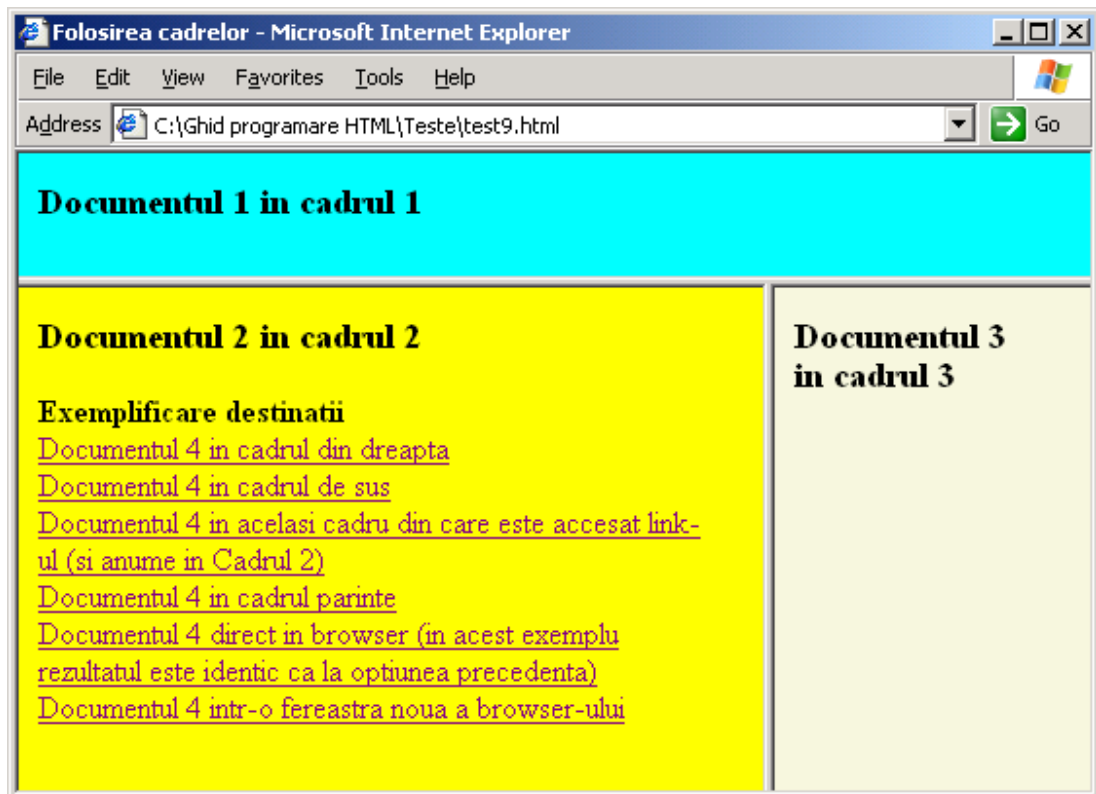
```
<HTML>
<HEAD>
  <TITLE>Documentul 3</TITLE>
</HEAD>
<BODY BGCOLOR="beige">
  <H3>Documentul 3 in cadrul 3</H3>
</BODY>
```

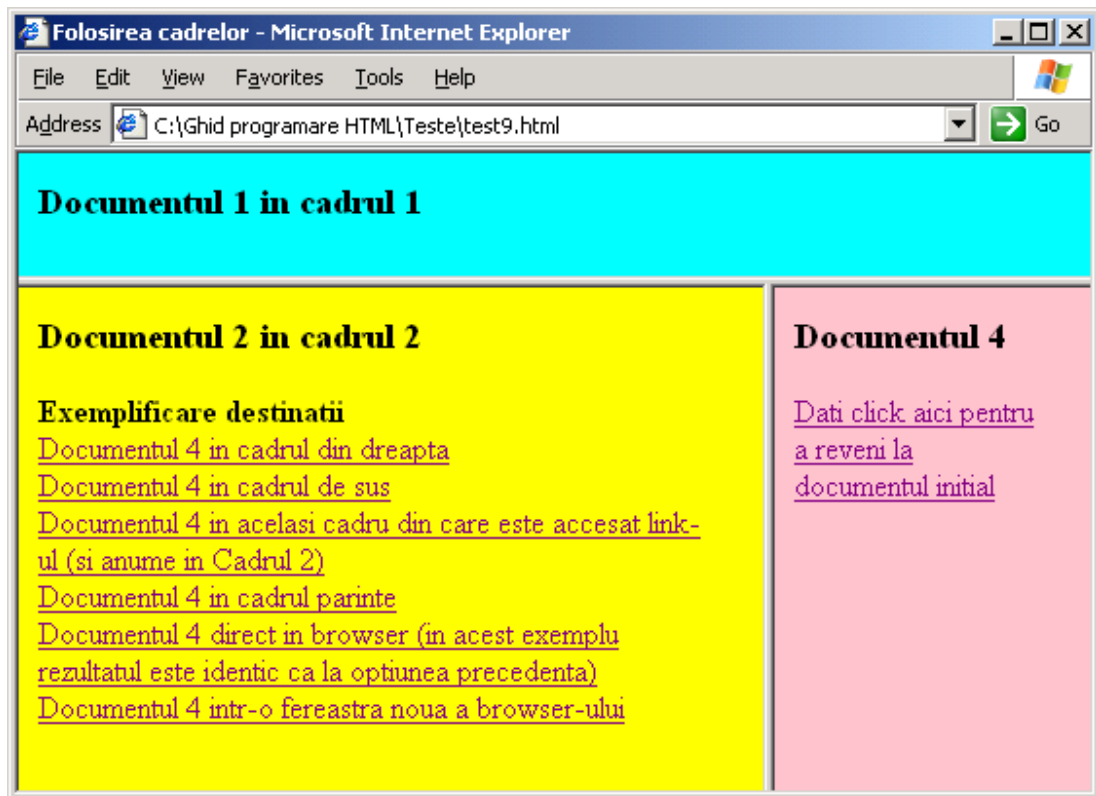
</HTML>

Și în final vom crea fișierul **doc4.html** cu următoarea structură:

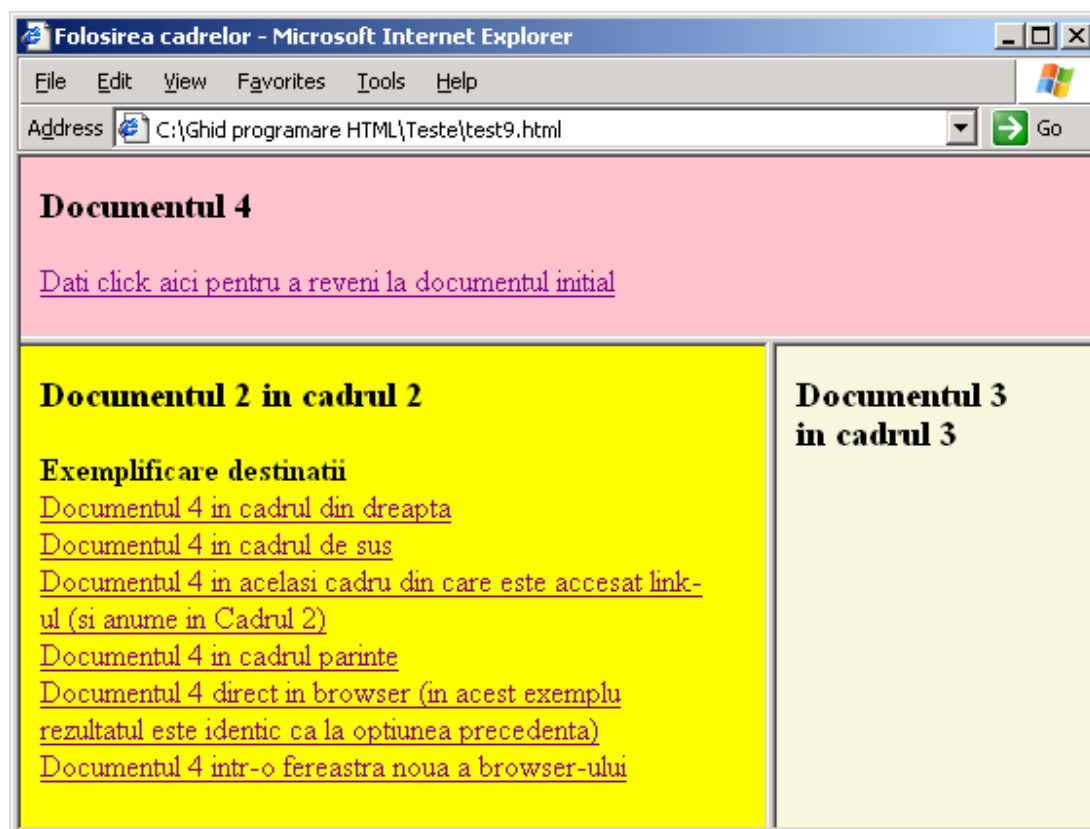
```
<HTML>
<HEAD>
  <TITLE>Documentul 4</TITLE>
</HEAD>
<BODY BGCOLOR="pink">
  <H3>Documentul 4</H3>
  <P><A HREF="test9.html" TARGET="_top"> Dati click aici pentru a reveni
la documentul initial</A></P>
</BODY>
</HTML>
```

Rezultatele obținute vor fi următoarele:

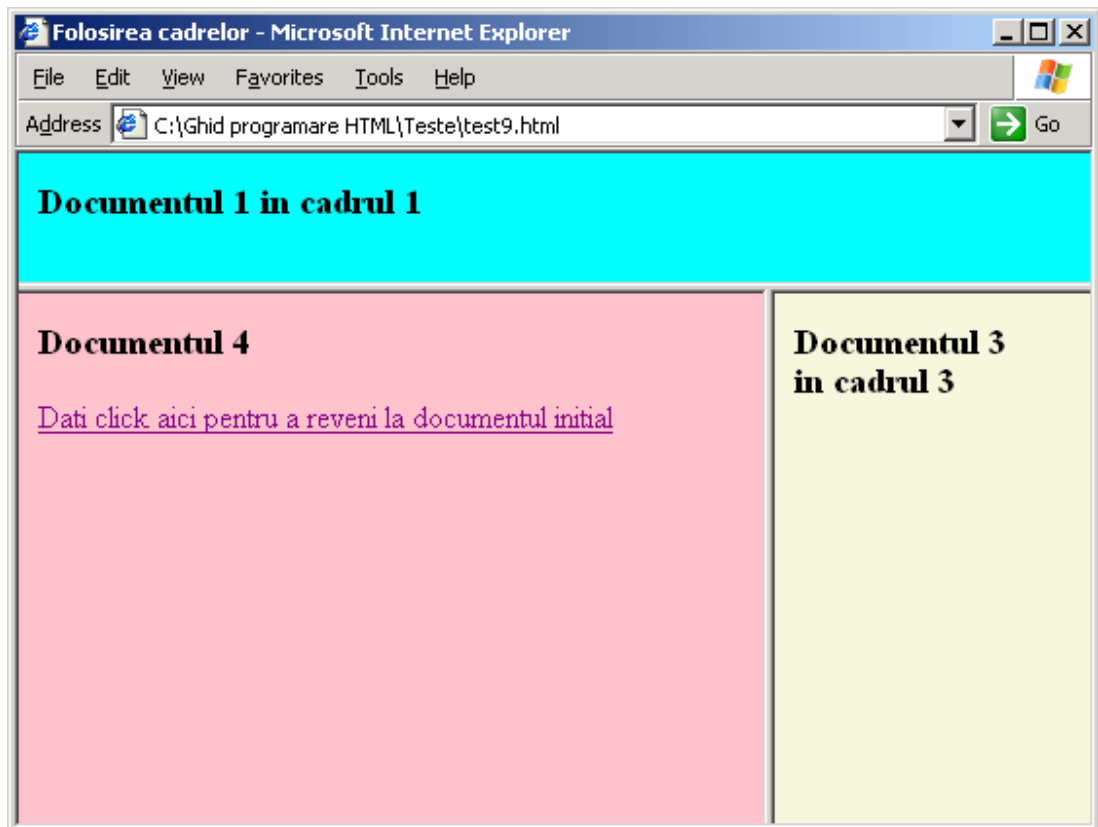




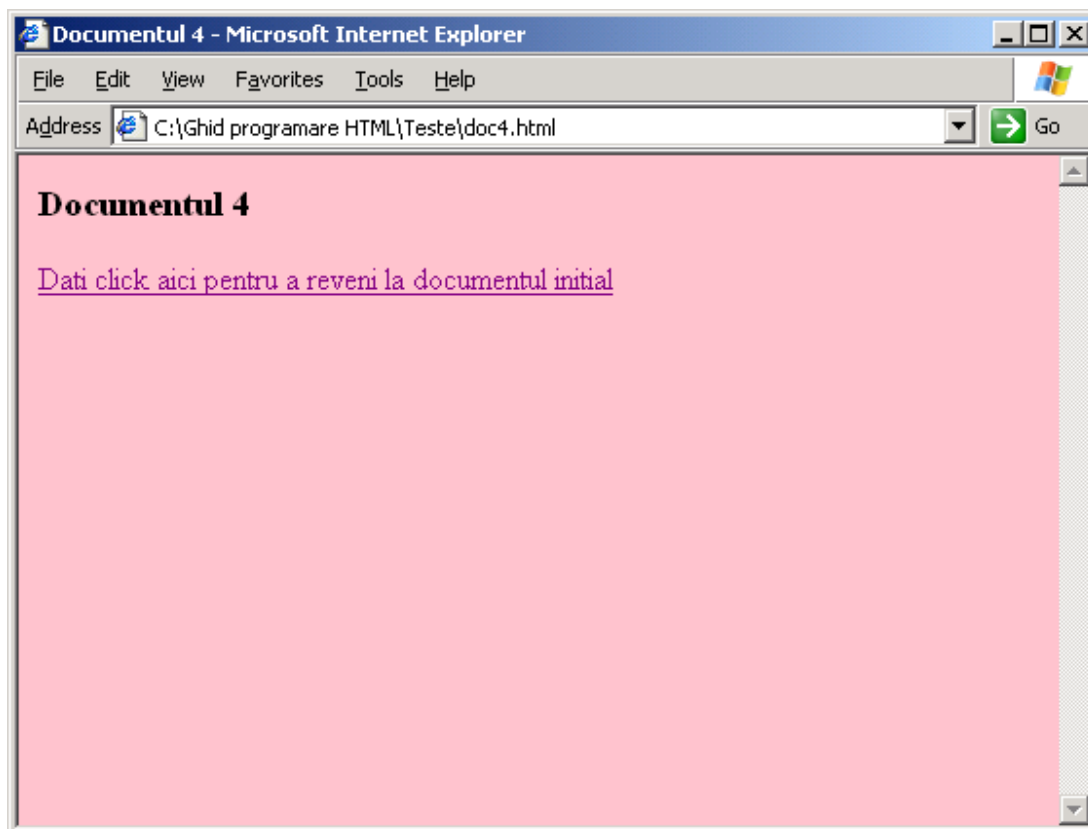
Documentul 4 în cadrul din dreapta



Documentul 4 în cadrul de sus



Documentul 4 în același cadru din care este accesat link-ul (și anume în Cadrul 2)



Imaginea afișată este identică pentru ultimele 3 opțiuni:

- Documentul 4 în cadrul parinte
- Documentul 4 direct în browser
- Documentul 4 într-o fereastră nouă a browser-ului

9.5. Cadre interioare

Cadrele interioare (in-line) sunt cadre ce pot fi inserate oriunde în interiorul unei pagini web (similar elementelor imagine). Pentru a insera un cadru interior, se folosește elementul `<IFRAME>` ce necesită tag-ul de închidere `</IFRAME>`.

Acest element folosește toate atributele elementului `<FRAME>`, cu aceleași efecte (și anume **src**, **frameborder**, **marginheight**, **marginwidth**, **scrolling** și **name**).

Existând similaritatea cu elementele imagine, o serie de atribute, specifice elementelor imagine, pot fi utilizate cu elementul `<IFRAME>` obținându-se aceleași rezultate (și anume **width**, **height**, **align**, **hspace** și **vspace**).

Într-o pagină web pot fi adăugate oricâte cadre interioare sunt necesare.

Pentru exemplificare vom modifica **doc3.html** astfel:

```
<HTML>  
<HEAD>
```

```

<TITLE>Documentul 3</TITLE>
</HEAD>
<BODY BGCOLOR="beige">
  <H3>Documentul 3 in cadrul 3</H3>
  <IFRAME SRC="doc4.html" ALIGN="center" WIDTH="200"
HEIGHT="150" </IFRAME>
</BODY>
</HTML>

```

Vom obține următorul rezultat:



9.6. Elementul <NOFRAMES>

Înainte de a încheia și acest capitol, trebui să aveți în vedere că încă mai există oameni care folosesc anumite browsere ce nu suportă cadre. În aceste cazuri browserele vor afișa pagini goale.

Problema aceasta poate fi rezolvată prin inserarea elementului <NOFRAMES> în interiorul elementului <FRAMESET>.

În interiorul elementului <NOFRAMES> va trebui să inserați cod HTML obișnuit prin care puteți redirectiona vizitatorii paginilor dvs. către o variantă a site-ului fără cadre

(dacă există), sau puteţi să-i atenţionaţi asupra faptului ca documentul dvs. conţine cadre iar browser-ul lor nu suportă aşa ceva.

De exemplu:

```
<HTML>
<HEAD>
<TITLE> </TITLE>
</HEAD>
<FRAMESET ROWS="40%,60%" COLS="70%,30%">
  <FRAME SRC="doc1.html">
  <FRAME SRC="doc2.html">
  <FRAME SRC="doc3.html">
  <FRAME SRC="doc4.html">
<NOFRAMES>
  <BODY>Aceast site contine cadre pe care browser-ul dvs. nu le poate
  interpreta. Puteti vizita o versiune fara cadre a site-ului la adresa ....
</BODY>
</NOFRAMES>
</FRAMESET>
</HTML>
```

9.7. Aplicaţie practică

Pentru a pune în practică cele prezentate în acest capitol, dar si in cele anterioare, vom realiza o structură combinată, cu trei cadre, dintre care unul pe orizontală şi două pe verticală. Astfel, cadrul orizontal va fi alocat titlului (antetului), cel din stânga va fi alocat unui meniu, iar cel din dreapta va prezenta conţinutul corespunzător meniului.

Structura cadrelor va fi definită în fişierul **test_cadre.html** ce va avea următorul conţinut:

```
<HTML>
<HEAD>
<TITLE>Retete culinare</TITLE>
</HEAD>
<FRAMESET ROWS="15%,85%">
  <FRAME SRC="antet_cadre.html" NAME="antet" SCROLLING="no">
  <FRAMESET COLS="30%,70%">
    <FRAME SRC="meniu_cadre.html" NAME="meniu">
    <FRAME SRC="continut_cadre.html" NAME="retete">
  </FRAMESET>
</NOFRAMES>
  <H1>Retete culinare</H1> Browser-ul dvs. nu suporta cadre? Nicio problema!
  Puteti vedea continutul site-ului nostru in versiunea
  <A HREF="continut_cadre.html">fara cadre</A>.
</NOFRAMES>
```

```
</FRAMESET>
</HTML>
```

Pentru antet vom crea fișierul **antet_cadre.html** cu următorul conținut:

```
<HTML>
<HEAD>
  <TITLE>Titlu</TITLE>
</HEAD>
<BODY BGCOLOR="saddlebrown">
  <H1><FONT COLOR="white">Retete culinare</FONT></H1>
</BODY>
</HTML>
```

Pentru meniu vom crea fișierul **meniu_cadre.html** cu următorul conținut:

```
<HTML>
<HEAD>
  <TITLE>Meniu</TITLE>
</HEAD>
<BODY BGCOLOR="wheat">
  <H2>Meniu</H2>
  <A HREF="continut_cadre.html#Aperitiv din branza de burduf"
  TARGET="retete">Aperitiv din branza de burduf</A><BR><BR>
  <A HREF="continut_cadre.html#Ciorba de caprioara cu smantana"
  TARGET="retete">Ciorba de caprioara cu smantana</A><BR><BR>
  <A HREF="continut_cadre.html#Friptura de iepure"
  TARGET="retete">Friptura de iepure</A><BR><BR>
  <A HREF="continut_cadre.html#Coltunasi dulci cu branza de vaci"
  TARGET="retete">Coltunasi dulci cu branza de vaci</A>
</BODY>
</HTML>
```

Continutul va fi cuprins in fișierul **continut_cadre.html** astfel:

```
<HTML>
<HEAD>
  <TITLE>Continut</TITLE>
</HEAD>
<BODY BGCOLOR="beige">
  <A NAME="Aperitiv din branza de burduf"><H2>Aperitiv din branza de
  burduf</H2></A>
  <B>Ingrediente:</B>
  <UL>
    <LI>1/2 kg branza de oaie (se poate inlocui cu cas proaspat)
    <LI>100 ml smantana
```

- 5 sardele
- ceapa
- chimen
- 50-100 ml bere

Mod de preparare:

<P>Se piseaza branza cu o furculita (daca e cas atunci se da prin masina de tocat). Peste se toarna berea, smantana. Chimenul se macina si se adauga o lingurita fara varf, iar la final sardelele si ceapa rasa. Se amesteca bine bine si se sareaza dupa gust. De preferat sa se serveasca in aceeasi zi.</P>

Recomandari:

<P>Sa se orneze cu bucatele taiate de gogosari sau masline</P>

<H2>Ciorba de caprioara cu smantana</H2>

Ingrediente:

500 g carne de caprioara

100 g ceapa

100 g morcovi

50 g patrunjel

50 g ulei

75 g orez

250 g rosii

40 ml otet

200 g smantana

2 oua

verdeata

sare

Mod de preparare:

<P>Carnea de caprioara se portioneaza si se pune la fiert in apa cu sare. In timpul fierberii se spumeaza cat este nevoie. Ceapa se toaca marunt, se caleste in ulei, se adauga zarzavatul de supa taiat julien si se mai calesc impreuna cateva minute. Se sting cu apa si se pun in ciorba. Cand carnea este aproape fiarta, se pune orezul fiert separat si rosiile curatate de coaja si seminte si taiate cuburi. Smantana se amesteca cu galbenusurile de ou si doua linguri de otet si se adauga la ciorba. Se mai potriveste o data gustul de sare, iar la servit se presara cu verdeata tocata marunt.</P>

<H2>Friptura de iepure</H2>

Ingrediente:

1 kg carne de iepure (pulpa)

50 g slanina afumata

50 g unt

sare

Mod de preparare:

<P>Carnea de iepure se sterge cu un stergar, se impaneaza cu slanina afumata, se freaca cu sare, se unge cu unt si se asaza intr-o tava cu restul de unt si putina apa, se da la cuptor si se frige la un foc potrivit. In timpul frigerii se stropeste cu sosul format in tava. Se mananca cu garnituri de legume si salate de sezon.</P>

<H2>Coltunasi dulci cu branza de vaci</H2>

Ingrediente:

500 g faina

4 oua

500 g branza de vaci

40 g coaja rasa de lamaie

175 g zahar

sare

Mod de preparare:

<P>Din faina, 2 oua, putina sare si putina apa se prepara o coca mai deasa, din care se intinde o foaie si se taie patrate cu latura de 5-7 cm. Branza de vaci se amesteca cu 75 g zahar, 2 oua si coaja de lamaie rasa, se repartizeaza, in mod egal, pe patratele de coca, se indoaie in forma de triunghi si se lipesc bine. Se fierb in apa cu sare, se scurg, se trec printr-un jet de apa rece, se toarna deasupra untul fierbinte si se presara cu restul de zahar. </P>

</BODY>

</HTML>

Rezultatul afişat va fi următorul:



CAP. 10. FORMULARE

10.1. Noțiuni generale

Formulare: iată o noțiune de care ne lovim tot timpul, pentru că suntem nevoiți să completăm formulare din momentul în care ne-am născut și până murim.

Și tot formularele sunt cele care fac cu adevărat din Internet o tehnologie interactivă. Fără aceste formulare, utilizatorul nu ar avea acces decât la anumite resurse statice, predefinite, pe când, cu ajutorul formularelor, pot fi introduși anumiți parametri care dau un caracter dinamic Internet-ului.

Dar pentru a analiza parametrii introduși în formulare sunt necesare anumite aplicații care decid cum va acționa serverul ca răspuns la datele primite. Codul acestor aplicații poate fi scris în limbaje de programare web precum ASP, JAVA, PHP, JSP etc. care nu fac însă subiectul acestui ghid.

Un formular poate conține unul sau mai multe controale de tipul: casete text (text boxes), casete de parole (password boxes), ascunse (hidden), casete de validare (checkboxes), butoane radio (radio buttons), buton trimitere formular (submit), buton resetare formular (reset), buton imagine (image), liste etc., toate plasate în interiorul elementului marcat de tag-ul <FORM>.

Atributele specifice acestui element sunt:

Atribut	Valoare	Descriere
action	URL	Un URL ce definește adresa aplicației web care prelucrează datele sau o adresă de e-mail validă unde vor fi trimise datele după apăsarea butonului Submit (Trimite).
accept	listă de conținut	O listă cu elemente separate de virgulă a tipurilor de conținut pe care serverul ce procesează formularul le poate interpreta corect.
accept-charset	listă de caractere	O listă cu elemente separate de virgulă a caracterelor posibile ce constituie datele formularului. Valoarea implicită este unknown .
enctype	mimetype	Determină mecanismul folosit la codificarea conținutului formularului.
method	get post	Get (implicit) sau post specifică ce metodă HTTP va fi folosită la transmiterea datelor din formulat către serverul web.
name	numele formularului	Definește un nume unic pentru formular, util în cazul folosirii unor scripturi Java sau VB.
target	_blank _self _parent _top	Specifică unde va deschis URL-ul țintă: ➤ _blank – într-o fereastră nouă; ➤ _self – în același cadru din care a fost accesat; ➤ _parent – în cadrul părinte; ➤ _top - direct în fereastra browser-ului, luând locul documentului existent.

10.2. Proiectarea unui formular

În procesul de proiectare de proiectare a unui formular se pornește de la datele pe care le doriți să le obțineți. Pentru aceasta vă recomandăm următoarele:

- alegeți controale care sunt adecvate pentru categoria de date pe care doriți să le obțineți și pentru interacțiunile pe care le furnizează;

- tag-ulți cu claritate fiecare control, astfel încât utilizatorul să înțeleagă funcția acestuia;

- în măsura posibilităților, aliniați tag-urile și marginile din stânga ale controalelor. Structura unui formular bine proiectat este asemănătoare cu aceea a unei pagini de ziar, fiind compusă dintr-o serie de coloane în cadrul cărora fiecare element vizual este aliniat cu elementele plasate deasupra, respectiv dedesubtul său. Puteți obține acest tip de structura folosind tabelele sau tag-ul `<BR>`;

- grupați controalele corelate și separați fiecare asemenea grup de celelalte grupuri prin încorporarea de spații albe în structura dvs.;

- secvența de controale din formular trebuie să fie asemănătoare cu ordinea în care le va manipula utilizatorul. În caz contrar, utilizatorul va fi obligat să piardă timp navigând de la un control la altul.

Este important să rețineți că acestea sunt doar îndrumări, nu reguli. Nefiind reguli, puteți opta, din când în când, pentru încălcarea lor. Pentru a determina dacă o variantă de proiectare este superioară alteia, este util să cereți părerea unui grup de utilizatori. Dacă nici aceștia nu vă pot fi de ajutor, încercați să obțineți asistență din partea unui alt programator. În cel mai rău caz, străduiți-vă să vă puneți dvs. înșivă în rolul unui utilizator al formularului. Întrebați-vă dacă formularul este clar, ușor de utilizat și eficient. În caz afirmativ, este momentul să treceți mai departe și să creați efectiv formularul HTML.

10.3. Crearea unui formular

Dacă procesul de proiectare este bazat pe oportunism, procesul de creare a unui formular, pornind de la structura proiectată, este mai direct.

Mai întâi, creați schița HTML elementară, care va conține controalele din formular, apoi încorporați controalele în structura respectivă. În particular, un formular HTML trebuie să includă un buton de expediere pe care utilizatorul execută click pentru a trimite serverului datele din formular. Deoarece o singură pagină HTML poate conține mai multe formulare, puteți repeta de mai multe ori etapele intermediare ale acestui proces.

Pentru a include un formular în interiorul corpului unei pagini HTML se folosește tag-ul `<FORM>`, care are următoarea sintaxă:

```
<FORM METHOD="metoda" ACTION="url">
```

Atributul *METHOD* al tag-ului *FORM* poate lua una din valorile *GET* sau *POST*. Pentru moment, specificați întotdeauna valoarea *POST*.

Atributul *ACTION* specifică adresa URL a scriptului ce prelucrează datele colectate prin intermediul formularului.

Adresa URL poate fi înscrisă ca adresă absolută (de exemplu `http://www.domeniul_meu.com/prelucrare_formular.php`), respectiv ca adresă relativă, care specifică o locație relativă la locația paginii curente (de exemplu `prelucrare_formular.php`).

Între tag-ul <FORM> și tag-ul de închidere </FORM> corespunzătoare, plasați controalele formularului.

Ca începător, este mai bine să folosiți în mod consecvent metoda POST, deoarece alegerea între metodele GET și POST este destul de complicată. Ca regulă empirică, mulți programatori folosesc GET pentru formulare care execută o căutare sau o interogare, respectiv POST pentru formulare care actualizează o bază de date sau un fișier. Două dezavantaje ale metodei GET sunt acelea că impune o limită asupra cantității de date care pot fi trimise scriptului de prelucrare și că transferă datele prin atașarea șirurilor codificate la adresa URL a scriptului de prelucrare. Astfel, datele trimise prin metoda GET pot fi vizualizate de către utilizator. Un avantaj al metodei GET este acela ca utilizatorii pot insera semne de carte în rezultatele unei interogari sau al unei căutări care folosește metoda GET, dar nu pot executa aceeași operație în cazul unei interogări sau al unei căutări care folosește metoda POST.

10.4. Codificarea datelor

Înainte de a le trimite la destinatar, browser-ul codifică datele din formular. Tehnica de codificare depinde de valoarea atributului **enctype**. Valorile posibile pentru acest atribut sunt:

a) application/x-www-form-urlencoded – utilizată împreună cu metoda **GET** (parametri prestabiliți). În acest caz:

- datele sunt trimise sub formă de listă de perechi **nume=valoare**, separate prin simbolul "&;
- spațiile albe sunt înlocuite de simbolul "+";
- caracterele non-alfanumerice sunt înlocuite cu simbolul "%" urmat de codul ASCII corespunzător caracterului respectiv;
- întreprinderile de rând sunt înlocuite de secvența "%0D%0A%" (CR/LF) (de exemplu: nume=Popescu&prenume=Ion&adresa=Str.Mare+nr.1 %0D%0A%Romania)

b) text/plain – utilizată împreună cu tipul de acțiune **action="mailto"** pentru trimiterea datelor prin e-mail. Acest caz este similar cazului precedent, cu excepția faptului ca spațiile nu mai sunt înlocuite de simbolul "+".

c) multipart/form-data – utilizată împreună cu metoda **POST** pentru trimiterea imaginilor sau a altor fișiere către server.

10.5. Inserarea controalelor de tip <INPUT> într-un formular

Utilizatorii interacționează cu formularele prin ceea ce poartă denumirea de "controale".

Cele mai utilizate tipuri de controale sunt cele definite de elementul <INPUT> prin intermediul atributului *TYPE*. Acestea sunt:

a) Câmpuri de tip TEXT

Se folosesc la introducerea de caractere alfanumerice într-un formular. Sintaxa uzuală este de forma:

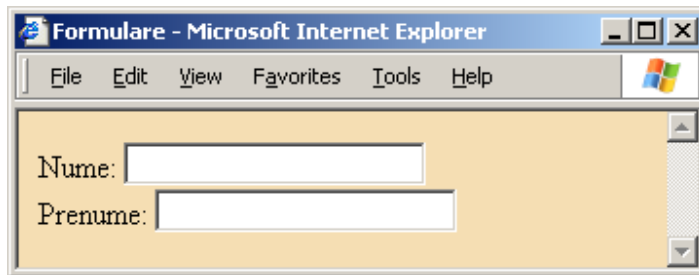
<FORM>

Num: **<INPUT TYPE="text" NAME="nume">
**

Prenume: **<INPUT TYPE="text" NAME="prenume">**

</FORM>

Rezultatul afișat în browser va fi următorul:

A screenshot of a Microsoft Internet Explorer browser window. The title bar says 'Formulare - Microsoft Internet Explorer'. The menu bar includes 'File', 'Edit', 'View', 'Favorites', 'Tools', and 'Help'. The main content area has a light yellow background and contains two text input fields. The first field is labeled 'Nume:' and the second is labeled 'Prenume:'. Both fields are empty.

Câmpurilor de tip text le pot fi precizate următoarele atribute:

TYPE:	text
SIZE:	determină mărimea câmpurilor text ca număr de caractere. Implicit are valoarea de 20 de caractere
MAXLENGTH:	determină numărul maxim de caractere ce pot fi intruse în acel câmp
NAME:	reprezintă numele variabilei ce va fi trimisă aplicației ce prelucrează formularul
VALUE:	afișează o valoare implicită; în general rămâne necompletată

b) Câmpuri de tip parolă (PASSWORD)

Se folosesc la introducerea de parole într-un formular. Sintaxa uzuală este de forma:

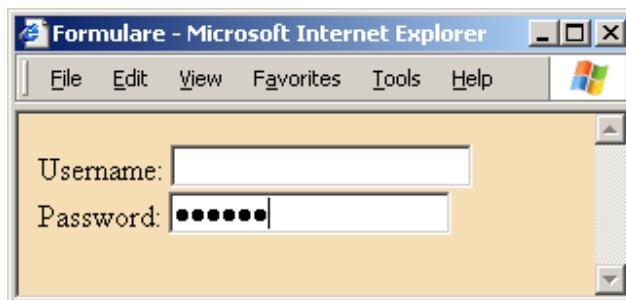
<FORM>

Username: **<INPUT TYPE="text" NAME="user">
**

Password: **<INPUT TYPE="password" NAME="password">**

</FORM>

Rezultatul afișat în browser va fi următorul:

A screenshot of a Microsoft Internet Explorer browser window. The title bar says 'Formulare - Microsoft Internet Explorer'. The menu bar includes 'File', 'Edit', 'View', 'Favorites', 'Tools', and 'Help'. The main content area has a light yellow background and contains two input fields. The first field is labeled 'Username:' and is a text input. The second field is labeled 'Password:' and is a password input, shown with a series of black dots. Both fields are empty.

Câmpurilor de tip parolă le pot fi precizate următoarele atribute:

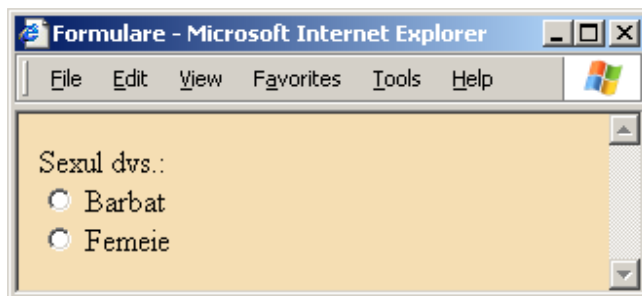
TYPE:	password
SIZE:	determină mărimea câmpurilor parolă ca număr de caractere. Implicit are valoarea de 20 de caractere
MAXLENGTH:	determină numărul maxim de caractere ce pot fi intruse în acel câmp
NAME:	reprezintă numele variabilei ce va fi trimisă aplicației ce prelucrează formularul
VALUE:	afișează o valoare implicită; în general rămâne necompletată

c) Butoane de tip RADIO

Se folosesc pentru a determina efectuarea unei singure alegeri dintr-o serie limitată într-un formular. Sintaxa uzuală este de forma:

```
<FORM>  
Sexul dvs.:<BR>  
<INPUT TYPE="radio" NAME="sex" VALUE="Barbat"> Barbat <BR>  
<INPUT TYPE="radio" NAME="sex" VALUE="Femeie"> Femeie  
</FORM>
```

Rezultatul afișat în browser va fi următorul:



Butoanelor de tip radio le pot fi precizate următoarele atribute:

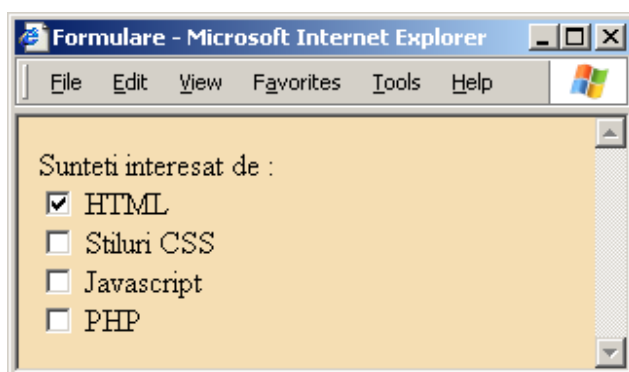
TYPE:	radio
CHECKED:	când se specifică, determină ca una din opțiuni să fie bifată inițial. Nu se poate folosi decât pentru un singur buton radio, în caz contrar, această specificație va fi luată în considerare numai la ultimul buton radio.
NAME:	reprezintă numele variabilei ce va fi trimisă aplicației ce prelucrează formularul
VALUE:	reprezintă valoarea ce va fi trimisă aplicației ce prelucrează formularul

d) Casete de validare (CHECKBOX)

Se folosesc pentru a determina efectuarea mai multor alegeri dintr-o serie limitată într-un formular. Sintaxa uzuală este de forma:

```
<FORM>
Sunteti interesat de :<BR>
<INPUT TYPE="checkbox" NAME="programare" VALUE="html"
CHECKED> HTML <BR>
<INPUT TYPE="checkbox" NAME="programare" VALUE="css"> Stiluri
CSS<BR>
<INPUT TYPE="checkbox" NAME="programare" VALUE="javascript">
Javascript <BR>
<INPUT TYPE="checkbox" NAME="programare" VALUE="php"> PHP
</FORM>
```

Rezultatul afișat în browser va fi următorul:



Casetelor de validare le pot fi precizate următoarele atribute:

TYPE:	checkbox
CHECKED:	când se specifică, determină ca una din opțiuni să fie bifată inițial
NAME:	reprezintă numele variabilei ce va fi trimisă aplicației ce prelucrează formularul
VALUE:	reprezintă valoarea ce va fi trimisă aplicației ce prelucrează formularul

De reținut este faptul că doar casetele de validare selectate sunt incluse în datele transmise către server.

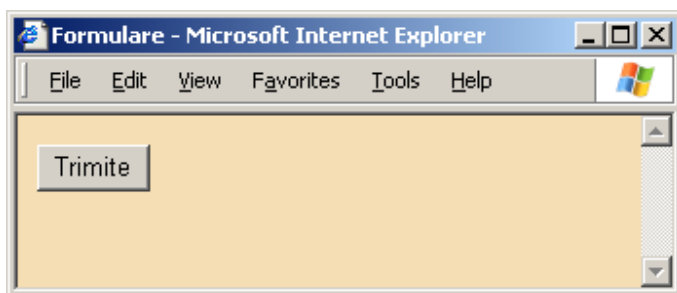
e) Buton de transmitere (SUBMIT)

Fiecare formular necesită un buton de transmitere. Acesta este elementul care determină browser-ul să trimită numele și valorile elementelor formularului către server pentru a fi prelucrate.

Sintaxa uzuală este de forma:

```
<FORM>  
<INPUT TYPE="submit" VALUE="Trimite">  
</FORM>
```

Rezultatul afișat în browser va fi următorul:



Butonul de transmitere poate avea următoarele atribute:

TYPE:	submit
NAME:	reprezintă numele variabilei ce va fi trimisă aplicației ce prelucrează formularul
VALUE:	reprezintă textul (tag-ul) ce va fi afișat pe buton

Un formular îi pot fi definite unul sau mai multe butoane de transmitere.

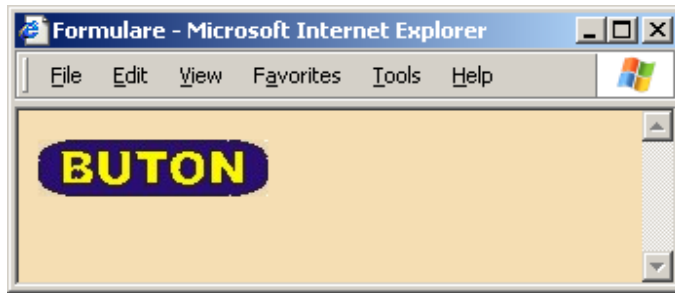
f) Buton de transmitere tip imagine (IMAGE)

Se folosește în cazul în care doriți să adăugați puțină "personalitate" butonului de transmitere.

Sintaxa uzuală este de forma:

```
<FORM>  
<INPUT TYPE="image" SRC="URL">  
</FORM>
```

Folosind ca imagine fișierul **buton.jpg**, existent pe site-ul nostru la adresa <http://html.webmarketclub.com/teste/buton.jpg>, precum și pe CD-ul atașat prezentului ghid, rezultatul afișat în browser va fi următorul:



Acest tip de buton de transmitere poate avea următoarele atribute:

TYPE:	image
NAME:	reprezintă numele butonului ce poate fi utilizat în cazul folosirii unor scripturi
SRC:	indică URL-ul fișierului ce conține imaginea respectivă

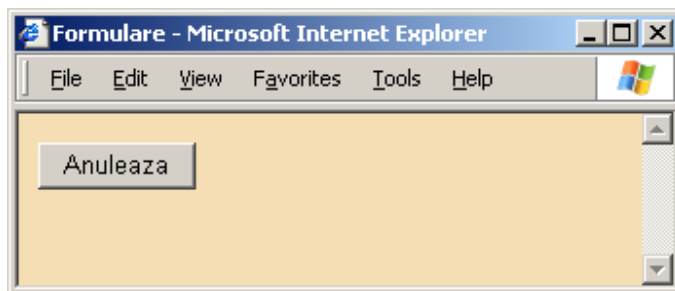
g) Buton de anulare (RESET)

Este o idee bună să includeți în formular și un buton de anulare. Acesta permite utilizatorului să golească întregul conținut al formularului.

Sintaxa uzuală este de forma:

```
<FORM>  
<INPUT TYPE="reset" VALUE="Anuleaza">  
</FORM>
```

Rezultatul afișat în browser va fi următorul:



Butonul de anulare poate avea următoarele atribute:

TYPE:	reset
VALUE	reprezintă textul (tag-ul) ce va fi afișat pe buton

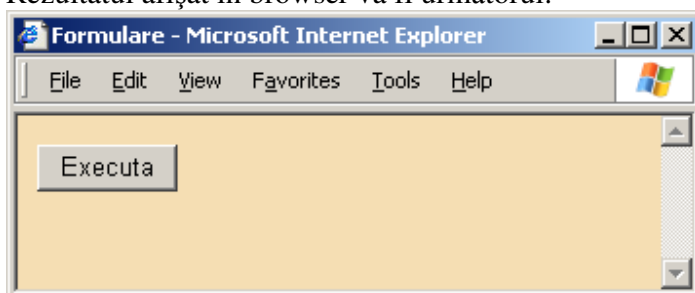
h) Buton generic (BUTTON)

Se folosește pentru a atașa unui formular un buton generic, care nu are un comportament prestabilit, prin apăsarea lui lansându-se în execuție un anumit script (folosind JavaScript sau alt limbaj de programare).

Sintaxa uzuală este de forma:

```
<FORM>  
<INPUT TYPE="button" VALUE="Executa">  
</FORM>
```

Rezultatul afișat în browser va fi următorul:



Butonul generic poate avea următoarele atribute:

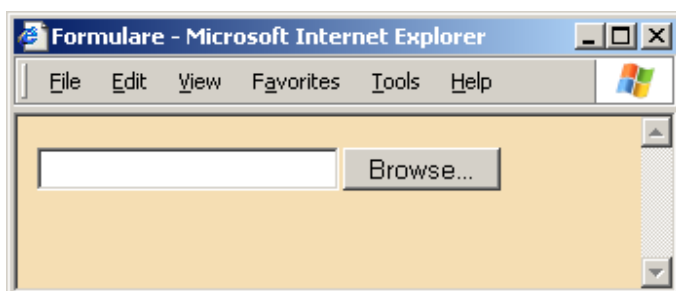
TYPE:	button
NAME:	reprezintă numele butonului ce poate fi utilizat în cazul folosirii unor scripturi
VALUE	reprezintă textul (tag-ul) ce va fi afișat pe buton

i) Câmpuri de tip fișier (FILE)

Sunt folosite pentru a permite utilizatorilor să facă upload fișierelor pe server.
Sintaxa uzuală este de forma:

```
<FORM>  
Username: <INPUT TYPE="file">  
</FORM>
```

Rezultatul afișat în browser va fi următorul:



Câmpurilor de tip fișier le pot fi precizate următoarele atribute:

TYPE:	file
SIZE:	determină mărimea câmpurilor fișier ca număr de caractere. Implicit are valoarea de 20 de caractere
NAME:	reprezintă numele variabilei ce va fi trimisă aplicației ce prelucrează formularul
VALUE:	specifică un nume de fișier prestabilit
ACCEPT:	lista tipurilor MIME separate de virgulă care indică tipul de transfer

j) Câmpuri ascunse (HIDDEN)

Sunt folosite pentru a trimite date fără însă a permite utilizatorilor să le poată vizualiza sau manipula, fiind totuși necesare pentru prelucrarea corectă a formularului. Sintaxa uzuală este de forma:

```
<FORM>  
<INPUT TYPE="hidden">  
</FORM>
```

În acest caz, browser-ul nu va afișa nimic. Utilitatea acestui tip de câmpuri o veți remarca în exemplul din finalul acestui capitol.

Câmpurilor ascunse le pot fi precizate următoarele atribute:

TYPE:	hidden
NAME:	reprezintă numele variabilei ce va fi trimisă aplicației ce prelucrează formularul
VALUE:	reprezintă valoarea ce va fi trimisă aplicației ce prelucrează formularul

10.6. Alte tipuri de controale

Pe lângă controalele prezentate anterior mai există o serie identificate fiecare prin propriul său element, și anume:

a) Liste de selecție

Există două tipuri de astfel de liste: liste cortină (drop-down list) și liste derulante (scrolling list). Ambele se obțin folosind elementul <SELECT>, diferențierea făcându-se cu ajutorul atributului *SIZE*, în cazul listelor cortină nefiind precizat sau având valoarea 1, iar în cazul listelor derulante putând avea orice valoare diferită de 1.

Atributele elementului <SELECT> sunt următoarele:

NAME:	reprezintă numele variabilei ce va fi trimisă aplicației ce
-------	---

prelucrează formularul
SIZE: reprezintă numărul de elemente din listă vizibile
MULTIPLE: dacă este specificat, acest atribut permite realizarea de selecții multiple, a mai multor elemente concomitent prin folosirea combinațiilor *SHIFT+click* pentru elemente consecutive, respectiv *CTRL+click* pentru elemente neconsecutive

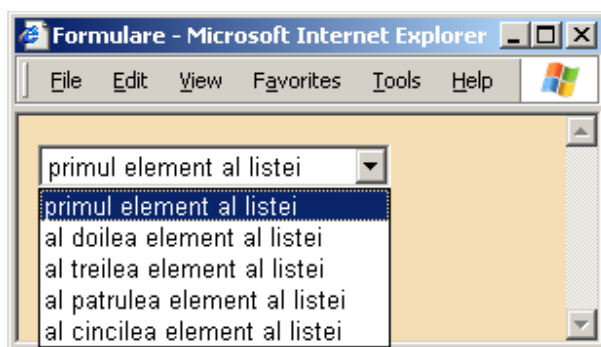
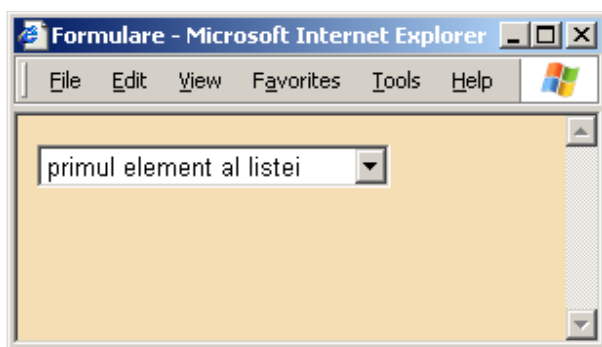
Elementele unor astfel de liste sunt marcate de tag-ul `<OPTION>`.
Atributele elementului `<OPTION>` sunt următoarele:

SELECTED: când acest atribut este prezent, opțiunea respectivă este preselectată
VALUE: reprezintă valoarea ce va fi trimisă aplicației ce prelucrează formularul

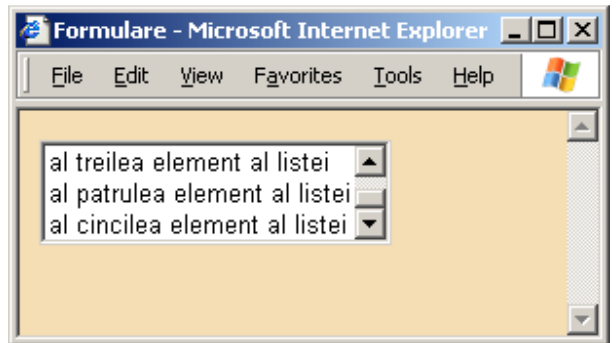
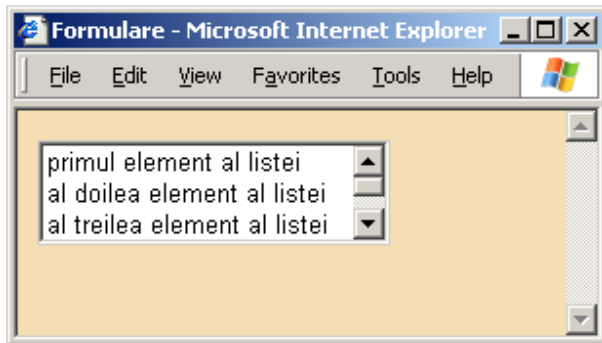
Sintaxa uzuală este de forma:

```
<FORM>  
<SELECT NAME="lista de selectie">  
<OPTION VALUE="1">primul element al listei</OPTION>  
<OPTION VALUE="2">al doilea element al listei</OPTION>  
<OPTION VALUE="3">al treilea element al listei</OPTION>  
<OPTION VALUE="4">al patrulea element al listei</OPTION>  
<OPTION VALUE="5">al cincilea element al listei</OPTION>  
</SELECT>  
</FORM>
```

Întrucât atributul *SIZE* nu a fost precizat, valoarea sa implicită fiind "1", browser-ul va afișa o listă tip cortină:



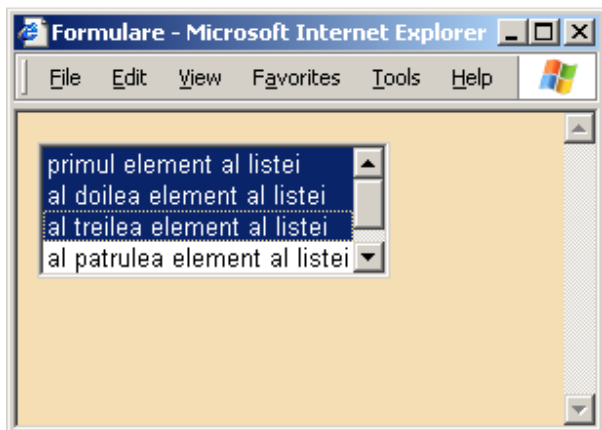
Aplicând atributului *SIZE* valoarea 3 (`<select name="lista de selectie" size="3">`), vom obține o listă derulantă, astfel:



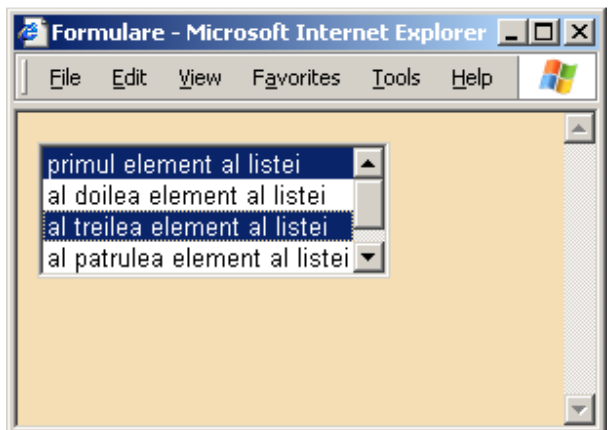
Pentru realizarea de selecții multiple se folosește atributul *MULTIPLE*. În acest caz vă recomandăm să specificați o valoare diferită de 1 atributului *SIZE* întrucât listele cortină nu acceptă selecții multiple. Sintaxa folosită este următoarea:

<SELECT NAME="lista de selectie" SIZE="4" MULTIPLE>

Rezultatul afișat în browser va fi următorul:



Selecție realizată cu SHIFT+click

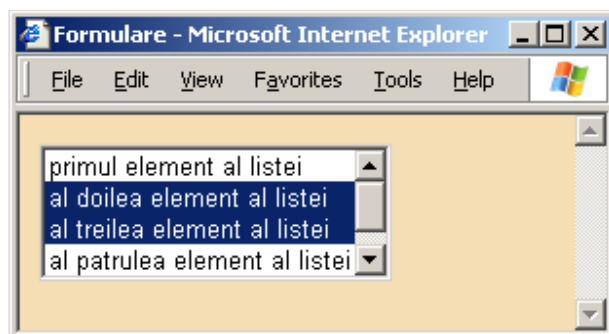


Selecție realizată cu CTRL+click

În cazul în care doriți să introduceți într-o astfel de listă unul sau mai multe elemente preselectate, atunci trebuie să adăugați atributul *SELECTED* fiecăruia din elementele respective, astfel:

```
<FORM>
<SELECT NAME="lista de selectie" SIZE="4" MULTIPLE>
<OPTION VALUE="1">primul element al listei</OPTION>
<OPTION VALUE="2" SELECTED>al doilea element al listei</OPTION>
<OPTION VALUE="3">al treilea element al listei</OPTION>
<OPTION VALUE="4">al patrulea element al listei</OPTION>
<OPTION VALUE="5">al cincilea element al listei</OPTION>
</SELECT>
</FORM>
```

Rezultatul afișat în browser va fi următorul:



Al doilea și al treilea element preselectate

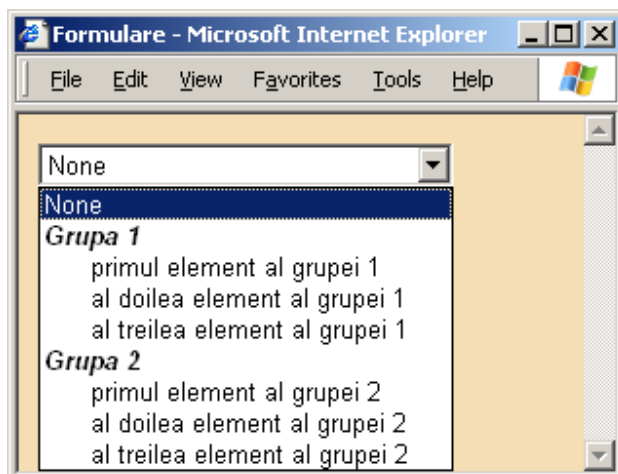
Pentru gruparea elementelor, se folosește sub-elementul `<OPTGROUP>` în interiorul elementului `<SELECT>`.

Acest sub-element are un singur atribut (a cărui prezență este obligatorie), și anume *LABEL*, ce definește tag-ul unui grup.

Sintaxa folosită este următoarea:

```
<FORM>  
<SELECT NAME="lista de selectie">  
<OPTION SELECTED VALUE="none">none</OPTION>  
<OPTGROUP LABEL="Grupa 1">  
<OPTION VALUE="1.1">primul element al grupei 1</OPTION>  
<OPTION VALUE="1.2">al doilea element al grupei 1</OPTION>  
<OPTION VALUE="1.3">al treilea element al grupei 1</OPTION>  
</OPTGROUP>  
<OPTGROUP LABEL="Grupa 2">  
<OPTION VALUE="2.1">primul element al grupei 2</OPTION>  
<OPTION VALUE="2.2">al doilea element al grupei 2</OPTION>  
<OPTION VALUE="2.3">al treilea element al grupei 2</OPTION>  
</OPTGROUP>  
</SELECT>  
</FORM>
```

Rezultatul afișat în browser va fi următorul:



b) Zone cu text (TEXTAREA)

La fel ca și o casetă cu text, o zonă cu text permite unui utilizator să introducă un anumit text. Diferența constă în faptul că o zonă cu text poate permite utilizatorului să introducă mai multe linii de text, în timp ce o casetă cu text permite utilizatorului să introducă o singură linie.

Atributele elementului <TEXTAREA> sunt următoarele:

NAME:	reprezintă numele zonei cu text, respectiv al variabilei ce va fi trimisă aplicației ce prelucrează formularul
ROWS:	specifică numărul de rânduri vizibile în zona cu text
COLS:	specifică numărul de coloane vizibile în zona cu text
READONLY:	când este specificat, indică faptul că utilizatorii formularului nu pot modifica conținutul zonei cu text

Iată sintaxa pentru crearea unei zone cu text:

<FORM>

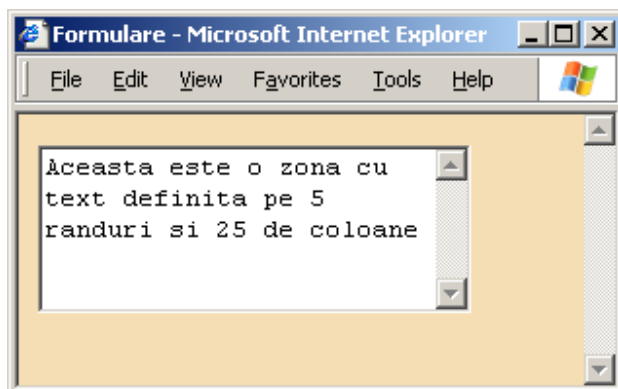
<TEXTAREA NAME="zona cu text" ROWS="5" COLS="25" READONLY>

Aceasta este o zona cu text definita pe 5 randuri si 25 de coloane

</TEXTAREA>

</FORM>

Rezultatul afișat în browser va fi următorul:



10.7. Alte attribute ale formularelor

În afară de cele prezentate anterior, mai există două attribute ce pot fi adăugate elementelor unui formular. Acestea sunt:

- a) **accesskey** – ce definește o scurtătură pentru executarea unei comenzi specifice respectivului element și/sau focalizează acel element. Ca valori posibile se poate folosi orice caracter (de exemplu "f"). Pentru a selecta sau executa comanda, utilizatorul trebuie să țină apăsată tasta ALT și apoi să apese tasta literei corespunzătoare (de exemplu ALT+f).

*Notă: Nu toate browser-ele suportă acest atribut. De exemplu, deși în Microsoft Internet Explorer funcționează corect, în Mozilla Firefox combinația ALT+f va avea ca efect apelarea meniului **File**.*

- b) **tabindex** – ce definește o altă ordine de parcurgere a elementelor formularului în cazul apăsării tastei TAB (ordinea prestabilită fiind aceea a ordinii în care elementele au fost adăugate în formular). Valorile posibile ale acestui atribut sunt reprezentate de orice număr întreg pozitiv.

Sintaxa uzuală privind folosirea acestor attribute este:

<FORM>

Nume: **<INPUT TYPE="text" NAME="nume" ACCESSKEY="n" TABINDEX="1">**Acest control se poate accesa și cu ALT+n**
**

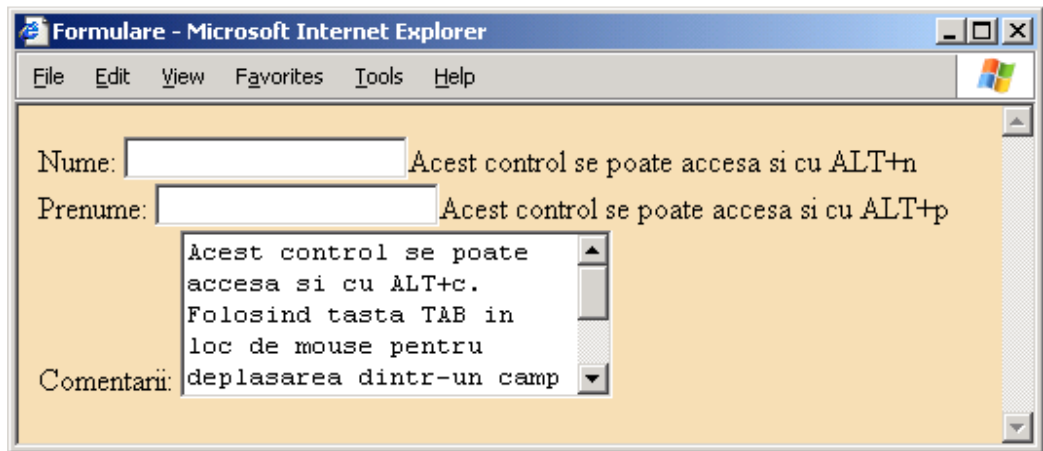
Prenume: **<INPUT TYPE="text" NAME="prenume" ACCESSKEY="p" TABINDEX="3">**Acest control se poate accesa și cu ALT+p**
**

Comentarii: **<TEXTAREA NAME="comentarii" ROWS="5" COLS="25" ACCESSKEY="c" TABINDEX="2">**Acest control se poate accesa și cu ALT+c. Folosind tasta TAB în loc de mouse pentru deplasarea dintr-un camp în altul, vei observa efectul atributului TABINDEX.

</TEXTAREA>

</FORM>

Rezultatul afișat în browser va fi următorul:



10.8. Aplicație practică

Pentru a pune în practică cele prezentate anterior, vom crea un fișier **test10.html** ce va cuprinde un model de formular complex, ale cărui rezultate vor fi expediate pe adresa voastră de e-mail.

Pentru obținerea unui aspect cât mai plăcut, în sensul alinierii corespunzătoare a elementelor formularului, vom crea un tabel în interiorul acestuia.

```
<HTML>
<HEAD>
<TITLE>Formulare</TITLE>
</HEAD>
<BODY BGCOLOR="wheat">
<FORM NAME="formular_F1" METHOD="POST" ENCTYPE="text/plain"
ACTION="mailto:adresa_de_mail@domeniu.com">
<INPUT TYPE="hidden" NAME="Subiect" VALUE="Rezultatul exercitiului
cu formule">
<TABLE BORDER="0" CELLSPACING="10">
<TR><TD>Nume: </TD>
<TD><INPUT TYPE="text" SIZE="25" MAXLENGTH="50" NAME="Nume"
ACCESSKEY="n" TABINDEX="1"></TD></TR>
<TR><TD>Prenume: </TD>
<TD><INPUT TYPE="text" SIZE="25" MAXLENGTH="50"
NAME="Prenume" ACCESSKEY="p" TABINDEX="2"></TD></TR>
<TR><TD>Parola: </TD>
<TD><INPUT TYPE="password" SIZE="25" MAXLENGTH="50"
NAME="Parola" ACCESSKEY="q" TABINDEX="3"></TD></TR>
<TR><TD>Sexul dvs.:</TD>
<TD><INPUT TYPE="radio" NAME="Sex" VALUE="B"> Barbat
<INPUT TYPE="radio" NAME="Sex" VALUE="F"> Femeie</TD></TR>
<TR><TD VALIGN="top">Sunteti interesat/a de:</TD>
```

```

        <TD><INPUT TYPE="checkbox" NAME="Programare" VALUE="html"
CHECKED> HTML <BR>
        <INPUT TYPE="checkbox" NAME="Programare" VALUE="css"> Stiluri
CSS<BR>
        <INPUT TYPE="checkbox" NAME="Programare" VALUE="javascript">
Javascript <BR>
        <INPUT TYPE="checkbox" NAME="Programare" VALUE="php">
PHP</TD></TR>
        <TR><TD VALIGN="top">Ati ajuns la capitolul:</TD>
        <TD><SELECT NAME="Capitolul curent">
        <OPTION VALUE="1">CAP. 1. NOTIUNI GENERALE</OPTION>
        <OPTION VALUE="2">CAP 2. CREAREA SI EDITAREA UNEI PAGINI
WEB</OPTION>
        <OPTION VALUE="3">CAP. 3. FORMATAREA
DOCUMENTULUI</OPTION>
        <OPTION VALUE="4">CAP. 4. FORMATAREA TEXTULUI</OPTION>
        <OPTION VALUE="5">CAP. 5. CREAREA LISTELOR</OPTION>
        <OPTION VALUE="6">CAP. 6. HYPERLINK-URI</OPTION>
        <OPTION VALUE="7">CAP. 7. FOLOSIREA IMAGINILOR</OPTION>
        <OPTION VALUE="8">CAP. 8. TABELE</OPTION>
        <OPTION VALUE="9">CAP. 9. CADRE</OPTION>
        <OPTGROUP LABEL="CAP. 10. FORMULARE">
        <OPTION VALUE="10.1.">Notiuni generale</OPTION>
        <OPTION VALUE="10.2.">Elementele formularelor</OPTION>
        <OPTION VALUE="10.3.">Crearea unui formular </OPTION>
        <OPTION VALUE="10.4.">Codificarea datelor</OPTION>
        <OPTION VALUE="10.5.">Inserarea controalelor de tip "INPUT"</OPTION>
        <OPTION VALUE="10.6.">Alte tipuri de controale </OPTION>
        <OPTION VALUE="10.7.">Alte attribute ale formularelor</OPTION>
        <OPTION VALUE="10.8.">SELECTED>Aplicatie
practica</OPTION></OPTGROUP>
        </SELECT></TD></TR>
        <TR><TD VALIGN="top">Comentarii:</TD>
        <TD><TEXTAREA NAME="Comentarii" ROWS="5" COLS="25"
ACCESSKEY="c" TABINDEX="4">Scrieti aici comentariile
voastre</TEXTAREA></TD></TR>
        <TR><TD ALIGN="center"><INPUT TYPE="submit" VALUE=" Trimite
"></TD>
        <TD ALIGN="center"><INPUT TYPE="reset" VALUE="
Anulează"></TD></TR>
    </TABLE>
</FORM>
</BODY>
</HTML>

```

Rezultatul afișat în browser va fi următorul:

Formulare - Microsoft Internet Explorer

File Edit View Favorites Tools Help

Address C:\Ghid programare HTML\Teste\test10.html Go

Nume:

Prenume:

Parola:

Sexul dvs.: ☐ Barbat ☐ Femeie

Sunteti interesat/a de: ☒ HTML
☐ Stiluri CSS
☐ Javascript
☐ PHP

Ati ajuns la capitolul:

Comentarii:

Scrieti aici
comentariile voastre

iar rezultatul primit (în cazul în care nu completați nimic) va fi:

Subiect=Rezultatul exercitiului cu formulare

Nume=

Prenume=

Parola=

Programare=html

Capitolul curent=10.8.

Comentarii=Scrieti aici comentariile voastre

CAP. 11. JAVA ȘI JAVASCRIPT

11.1. Noțiuni generale

Vom începe acest capitol prin a face distincția Java și JavaScript.

Java este un limbaj de programare orientat-obiect, ce se adresează programatorilor avansați putând fi rulat fără modificări pe orice computer (PC, PDA etc) pe care există instalată o "mașină virtuală" Java (JVM=Java Virtual Machine), fiind extrem de potrivit, deși nu exclusiv, pentru Internet.

Pe de altă parte, JavaScript este un limbaj de programare orientat-obiect bazat pe conceptul prototipurilor. Limbajul este binecunoscut pentru folosirea în construirea siturilor web, dar de asemenea este folosit pentru accesul la obiecte încastate (embedded objects) în alte aplicații.

În ciuda numelui și a unor oarecare similarități în sintaxă, între JavaScript și limbajul Java nu există nici o legătură. Ca și Java, JavaScript are o sintaxă asemănătoare cu cea a limbajului C, dar are mai multe în comun cu Limbajul de Autoprogramare decât cu Java.

11.2. Limbajul JAVA

Java este un limbaj de programare conceput de Sun Microsystems la începutul anilor 90.

A fost conceput ca un limbaj de programare orientat-obiect foarte puternic, "cross-platform" și "device-independent", ceea ce înseamnă că programul compilat trebuie să ruleze atât pe PC-uri, MAC-uri sau calculatoare cu platformă UNIX.

Browser-ele creează un mediu pentru rularea **applet**-urilor Java, numit Java Virtual Machine. Acesta oferă independența platformei pentru applet-urile Java.

Denumirea de **applet** se referă la orice mini-aplicație creată cu ajutorul limbajului Java. Crearea unui applet nu face obiectul acestui ghid. Applet-urile sunt programe compilate separate ce au extensia **".class"**. Acestea sunt descărcate împreună cu pagina web prin folosirea tag-ului <APPLET>.

Sintaxa HTML pentru folosirea tag-ului <APPLET> și a tag-ului său asociat <PARAM>, este următoarea:

```
<APPLET  
  [CODEBASE = directorApplet]  
  [CODE = clasaApplet]  
  [ALT = textAlternativ]  
  [NAME = numeInstanțăApplet]  
  WIDTH = lățimeInPixeli  
  HEIGHT = înălțimeInPixeli  
  [ALIGN = aliniere]  
  [VSPACE = spațiuVertical]  
  [HSPACE = spațiuOrizontal] >  
[< PARAM NAME = numeParametru_1 VALUE = valoare_1 >]  
...
```

[< *PARAM NAME = numeParametru_n VALUE = valoare_n* >]
[text HTML alternativ]
 </APPLET>

Atributele puse între paranteze pătrate sunt opționale. Semnificația atributelor este următoarea:

Atribut	Valoare	Descriere
align	left right top bottom middle baseline texttop absmiddle absbottom	Semnifică modalitatea de aliniere a appletul-ui în pagina Web, semnificațiile lor fiind aceleași ca și la tag-ul .
alt	text	Specifică textul ce trebuie afișat dacă browser-ul înțelege tag-ul APPLET dar nu poate rula applet-uri Java.
code	URL	Specifică numele fișierului ce conține clasa principală a applet-ului. Acesta va fi căutat în directorul specificat de CODEBASE.
codebase	URL	Specifică URL-ul în care se găsește clasa applet-ului. Uzual se exprimă relativ la directorul documentului HTML. În cazul în care lipsește, se consideră implicit URL-ul documentului.
height	pixeli	Specifică înălțimea suprafeței în care va fi afișat applet-ul. <i>Este un atribut obligatoriu.</i>
hspace	pixeli	Specifică numărul de pixeli dintre applet și marginile orizontale ale suprafeței de afișare.
name	text	Oferă posibilitatea de a da un nume respectivei instanțe a applet-ului, astfel încât mai multe applet-uri aflate pe aceeași pagină să comunice între ele folosindu-se de numele lor.
vspace	pixeli	Specifică numărul de pixeli dintre applet și marginile verticale ale suprafeței de afișare.
width	pixeli	Specifică lățimea suprafeței în care va fi afișat applet-ul. <i>Este un atribut obligatoriu.</i>

< *PARAM NAME = numeParametru1 VALUE = valoare1* >

Tag-urile <PARAM> sunt folosite pentru specificarea parametrilor unui applet. Acești parametri permit utilizatorului să personalizeze aspectul sau comportarea unui applet fara a-i schimba codul si recompila clasele.

text HTML alternativ

Este textul ce va fi afișat în cazul în care browser-ul nu înțelege tag-ul <APPLET>. Browser-ele care înțeleg Java vor ignora acest text.

11.3. Limbajul JAVASCRIPT

JavaScript vă permite să creați o interfață activă cu utilizatorul, să dea o impresie vizuală plăcută în timp ce navigați prin paginile site-ului. Puteți verifica validitatea informațiilor introduse într-un formular înainte ca datele să fie trimise către server. Puteți face calcule matematice, crea pagini HTML dinamice, personalizate în funcție de opțiunile utilizatorului sau al browser-ului.

JavaScript controlează browser-ul, și prin intermediul scripturilor puteți deschide ferestre noi, puteți afișa mesaje de avertizare și puteți pune mesaje în bara de stare a ferestrei browserului. Puteți genera ceasuri, calendare și documente cu timpul înscris. Puteți chiar verifica prezența plug-in-urilor în browser, și puteți redirecționa utilizatorul spre o pagină diferită în cazul în care browserul nu are un anumit plug-in.

O mențiune importantă: programele care rulează pe calculatorul utilizatorului sunt numite aplicații client-side (aflate pe partea de client), și programele care rulează pe server (inclusiv CGI-urile) sunt numite aplicații server-side (aflate pe partea de server)

Deoarece JavaScript rulează pe calculatorul dumneavoastră, i-au fost impuse o serie de limitări, majoritatea din rațiuni de securitate, și anume:

- nu permite citirea sau scrierea de fișiere pe calculatorul utilizatorului. Singura excepție este permisiunea de a scrie în directorul de cookie-uri al browserului, dar și aici sunt anumite limite.

- nu permite citirea/scrierea de fișiere pe server. Alternativa este să rulați pe server un program care să preia datele trimise de script și să le scrie pe server (un CGI, scris într-un limbaj precum Perl sau PHP, sau un program Java.

- un script JavaScript nu poate închide o fereastră de browser care nu a fost deschisă de el. Astfel se evită situația când intrați pe un site și vi se închid celelalte ferestre active care conțin alte site-uri...

- nu poate citi informații dintr-un browser care conține o pagina web de pe un alt server, ca să nu puteți afla pe ce site-uri navighează utilizatorul.

Un script JavaScript este un program inclus într-o pagina HTML. Deoarece este încadrat de tag-ul **<script>**, textul scriptului nu apare pe ecran, dar este rulat și interpretat de către browser. Tag-ul **<script>** este prezent cel mai frecvent în secțiunea **<head>** a paginii HTML, deși se pot pune și în secțiunea **<body>**. De obicei, scripturile care urmează să afișeze mesaje pe ecran sunt scrise în **<body>**.

Vă prezentăm în continuare o listă a atributelor ce pot fi adăugate tag-urilor HTML pentru a defini diferite acțiuni:

- **Acțiuni legate de ferestre**

Valabile numai în elementele <BODY> și <FRAMESET>

Atribut	Valoare	Descriere
onload	script	Scriptul ce va fi rulat la încărcarea documentului

onunload	script	Scriptul ce va fi rulat la închiderea documentului
----------	--------	--

- **Acțiuni legate de formulare**

Valabile numai în elementul <FORM>

Atribut	Valoare	Descriere
onchange	script	Scriptul ce va fi rulat când se elementul se schimbă
onsubmit	script	Scriptul ce va fi rulat la trimiterea formularului
onreset	script	Scriptul ce va fi rulat la resetarea formularului
onselect	script	Scriptul ce va fi rulat la selectarea elementului
onblur	script	Scriptul ce va fi rulat la pierderea focalizării asupra elementului
onfocus	script	Scriptul ce va fi rulat la focalizarea asupra elementului

- **Acțiuni legate de tastatură**

Nu sunt valabile în elementele <BASE>, <BDO>,
, <FRAME>, <FRAMESET>, <HEAD>, <HTML>, <IFRAME>, <META>, <PARAM>, <SCRIPT>, <STYLE> și <TITLE>.

Atribut	Valoare	Descriere
onkeydown	script	Scriptul ce va fi rulat când este apăsată o tastă
onkeypress	script	Scriptul ce va fi rulat când este apăsată și eliberată o tastă
onkeyup	script	Scriptul ce va fi rulat când este eliberată o tastă

- **Acțiuni legate de mouse**

Nu sunt valabile în elementele <BASE>, <BDO>,
, <FRAME>, <FRAMESET>, <HEAD>, <HTML>, <IFRAME>, <META>, <PARAM>, <SCRIPT>, <STYLE> și <TITLE>.

Atribut	Valoare	Descriere
onclick	script	Scriptul ce va fi rulat când se face click cu mouse-ul

ondblclick	script	Scriptul ce va fi rulat când se face dublu-click cu mouse-ul
onmousedown	script	Scriptul ce va fi rulat când este apăsat butonul mouse-ului
onmousemove	script	Scriptul ce va fi rulat când este mișcat cursorul de la mouse
onmouseout	script	Scriptul ce va fi rulat când cursorul de la mouse este mutat în afara unui element
onmouseover	script	Scriptul ce va fi rulat când cursorul de la mouse este mutat asupra unui element
onmouseup	script	Scriptul ce va fi rulat când este eliberat butonul mouse-ului

11.4. Aplicație practică

Pentru a exemplifica cele arătate mai sus, vom crea un fișier **test11.html** ce va conține două applet-uri Java și un script Javascript.

```

<HTML>
<HEAD>
<TITLE>Java si Javascript</TITLE>
</HEAD>
<BODY>
<P ALIGN="center"> Iata un applet simplu ce reproduce functiile unui
calculator.<BR><BR>
<APPLET CODE="AdderApplet.class" WIDTH="175" HEIGHT="175">
</APPLET><BR><BR>
Iata un applet ce reproduce un calendar si foloseste diversi parametri.<BR><BR>
<APPLET CODEBASE="C:\Ghid programare HTML\Teste"
CODE="tcaldate.class" WIDTH="200" HEIGHT="170">
  <PARAM NAME="font" VALUE="Arial">
  <PARAM NAME="fontsize" VALUE="12">
  <PARAM NAME="fontstyle" VALUE="1">
  <PARAM NAME="bgcolor" VALUE="13684944">
  <PARAM NAME="bgcolor2" VALUE="14737663">
  <PARAM NAME="bgcolor3" VALUE="400">
  <PARAM NAME="_min" VALUE="01 Ian 2008">
  <PARAM NAME="current" VALUE="1z5 zzIan 08">
  <PARAM NAME="_max" VALUE="31 Ian 2008">
  <PARAM NAME="t_col1" VALUE="400">
  <PARAM NAME="t_col2" VALUE="8421520">
  <PARAM NAME="t_col3" VALUE="14737663">
</APPLET><BR><BR>
Iata un script ce schimba doua imagini intre ele in functie de pozitia mouse-
ului.<BR><BR>

```

```

<SCRIPT LANGUAGE="JavaScript">
<!-- Acest text nu va fi afisat de browserele ce nu suporta JavaScript
image_1aon = new Image(150, 50);
image_1aon.src = "picture1b.gif";
image_1aoff = new Image(150, 50);
image_1aoff.src = "picture1.gif";
// Functii
function
img_act(imgName) {
imgon = eval(imgName + "on.src");
document [imgName].src = imgon;
}
function
img_inact(imgName) {
imgoff = eval(imgName + "off.src");
document [imgName].src = imgoff;
}
--></SCRIPT>

<A HREF="test11.html" ONMOUSEOVER="img_act('image_1a')"
ONMOUSEOUT="img_inact('image_1a')"><IMG
SRC="picture1b.gif" BORDER="0"
WIDTH="80" HEIGHT="50" NAME="image_1a"></A>
</P>
</BODY>
</HTML>

```

Rezultatul afișat în browser-ul IE va fi următorul:

Java si Javascript - Microsoft Internet Explorer

File Edit View Favorites Tools Help

Address C:\Ghid programare HTML\Teste\test11.html Go

Iata un applet simplu ce reproduce functiile unui calculator.

7	8	9	/
4	5	6	*
1	2	3	-
0	.	+	=

CLEAR

Iata un applet ce reproduce un calendar si foloseste diversi parametri.

January 2008

	M	T	W	T	F	S	S
1	31	01	02	03	04	05	06
2	07	08	09	10	11	12	13
3	14	15	16	17	18	19	20
4	21	22	23	24	25	26	27
5	28	29	30	31	01	02	03
6	04	05	06	07	08	09	10

?

Iata un script ce schimba doua imagini intre ele in functie de pozitia mouse-ului.

CLICK

CAP. 12. PAGINI WEB MULTIMEDIA

12.1. Ce este multimedia?

Multimedia reprezintă tot ce poți auzi sau vedea: texte, cărți, imagini, muzică, sunete, CD-uri, DVD-uri, filme, tot felul de înregistrări video sau audio etc.

Multimedia se prezintă într-o multitudine de formate, iar pe Internet veți găsi multe dintre aceste elemente înglobate în pagini web. Browser-ele actuale oferă suport pentru multe dintre aceste formate, dar nu pentru toate.

Suportul pentru sunete, animații și video este tratat în mod diferit de browsere diferite. Unele elemente pot fi manevrate intern, altele necesită anumite plug-in-uri, în vreme ce altele pot solicita activarea anumitor controale ActiveX.

În acest capitol veți afla despre diferitele formate multimedia și cum să le folosiți într-o pagină web.

Elementele multimedia (precum sunetele și video) sunt stocate în fișiere media. Modalitatea cea mai simplă de a determina tipul unui fișier media este de a analiza extensia fișierului respectiv (de exemplu, cele mai utilizate tipuri de fișiere ce conțin imagini au extensia .jpg sau .gif).

12.2. Formatul fișierelor multimedia ce conțin sunete

Sunetele pot fi stocate în diferite formate de fișiere. Astfel deosebim:

a) Formatul MIDI

MIDI (Musical Instrument Digital Interface) este un format ce permite transmiterea de informații muzicale între dispozitive electronice muzicale (precum sintetizatoare) și plăcile de sunet ale calculatoarelor.

Formatul MIDI a fost implementat în 1982 de către industria muzicală. Acest format este foarte flexibil și poate fi folosit pentru (aproape) orice, de la crearea unor sunete foarte simple, până la crearea unora cu adevărat profesionale.

Fișierele MIDI nu conțin sunete, ci un set de instrucțiuni digitale muzicale (note muzicale) care pot fi interpretate de plăcile de sunet ale calculatoarelor.

Dezavantajul la acest tip de fișiere constă în faptul că nu pot stoca muzică (în toată complexitatea ei), ci doar note muzicale.

Avantajul îl constituie mărimea mică a acestor fișiere, prin urmare se pot încărca ușor într-o pagină web, precum și faptul că acest format este suportat de majoritatea sistemelor de operare și de cele mai populare browsere de Internet.

Fișierele de tip MIDI au extensia .mid sau .midi.

b) Formatul RealAudio

Formatul RealAudio a fost dezvoltat pentru Internet de către Real Media. Acest format suportă chiar și video.

Formatul permite difuzarea de informații audio (muzică on-line, Internet radio) cu lățime de bandă joasă. Din cauza acestei lățimi de bandă, calitatea este, de cele mai multe ori, slabă.

Fișierele de tip RealAudio au extensia .rm sau .ram.

c) Formatul AU

Acest formatul a fost dezvoltat de către Sun Microsystems și este suportat de majoritatea sistemelor de operare.

Aceste fișierele au extensia .au.

d) Formatul AIFF

AIFF (Audio Interchange File Format) a fost dezvoltat de către Apple. Acest tip de fișiere nu sunt suportate de toate browser-ele.

Aceste fișierele au extensia .aif sau .aiff.

e) Formatul SND

SND (Sound) a fost dezvoltat, de asemenea, de către Apple și au același dezavantaj, și anume că nu sunt suportate de toate browser-ele.

Aceste fișierele au extensia .snd.

f) Formatul WAVE

WAVE (waveform) a fost dezvoltat de către IBM și Microsoft. Este un format suportat de toate computerele pe care rulează Windows și este recunoscut de majoritatea browsere-lor.

Fișierele de tip WAVE au extensia .wav.

g) Formatul MP3

Fișierele MP3 sunt de fapt fișiere MPEG, dezvoltate inițial pentru video de către Moving Pictures Experts Group. Se poate afirma că fișierele MP3 reprezintă partea de sunet dintr-un fișiere video în format MPEG.

MP3 este în prezent unul dintre cele mai populare formate folosite la înregistrarea muzicii. Sistemul de codificare MP3 combină o bună compresie (ce duce la fișiere mici) cu calitate înaltă. Este de așteptat ca pe viitor toate sistemele de operare să ofere suport pentru acest tip de format.

Fișierele de tip MP3 au extensia .mp3 sau .mpga.

În funcție de scopul pentru care creați o pagină web multimedia va trebui să vă orientați și asupra unui anumit format de fișier. Astfel, dacă doriți ca sunetele înregistrate (muzică sau de alt tip) să poată fi ascultat de toți vizitatorii paginii web, atunci trebuie să folosiți fișiere în format WAV, întrucât este cel mai popular format pe Internet și este suportat de majoritatea browsere-lor.

Dacă pagina web este despre înregistrări muzicale, atunci formatul cel mai potrivit este MP3.

12.3. Formatul fișierelor multimedia ce conțin video

Și imaginile video pot fi stocate în diferite formate de fișiere. Astfel deosebim:

a) Formatul AVI

Formatul AVI (Audio Video Interleave) a fost dezvoltat de către Microsoft. Este un format suportat de toate computerele pe care rulează Windows și este recunoscut de majoritatea browsere-lor. Deși este un format des întâlnit pe Internet, nu este întotdeauna suportat de calculatoarele cu sisteme de operare non-Windows.

Fișierele de tip AVI au extensia .avi.

b) Formatul Windows Media

Acest formatul a fost dezvoltat de către Microsoft. Și acest format este unul suportat de toate computerele pe care rulează Windows, dar nu este suportat de calculatoarele cu sisteme de operare non-Windows decât în urma instalării unor extra componente (playere specializate).

Fișierele de tip Windows Media au extensia .wmv.

c) Formatul MPEG

Formatul MPEG (Moving Pictures Expert Group) este în prezent unul dintre cele mai populare formate folosite pe Internet. Acest format este suportat de majoritatea sistemelor de operare și de cele mai populare browsere de Internet.

Fișierele de tip MPEG au extensia .mpg sau .mpeg.

d) Formatul QuickTime

Formatul QuickTime a fost dezvoltat de către Apple și nu poate rula pe platformele Windows fără instalarea unui player specializat.

Fișierele de tip QuickTime au extensia .mov sau .qtm.

e) Formatul RealVideo

Formatul RealVideo a fost dezvoltat pentru Internet de către Real Media.

Formatul permite difuzarea de informații video (video on-line, Internet video) cu lățime de bandă joasă. Din cauza acestei lățimi de bandă, calitatea este, de cele mai multe ori, slabă.

Fișierele de tip RealVideo au extensia .rm sau .ram.

f) Formatul Shockwave (Flash)

Formatul Shockwave a fost dezvoltat de către Macromedia. Acest format necesită instalarea unui player specializat pentru a putea rula. În prezent, această componentă este oferită preinstalată de ultimele versiuni ale browsere-lor Netscape și Internet Explorer.

Fișierele de tip Shockwave au extensia .swf.

12.4. Adăugarea de informații multimedia unei pagini web

Informațiile multimedia pot fi redate direct de browser sau prin intermediul unui program ajutător (un player specializat), în funcție de elementul HTML și de formatul fișierului multimedia folosit.

Cea mai simplă cale de a realiza acest lucru este prin folosirea unei ancore (link) care să ducă la sursa elementului multimedia. Este cea mai “prietenosă” metodă întrucât

oferă posibilitatea vizitatorilor de a alege între a vedea o pagină web cu elemente multimedia sau fără acestea.

În acest caz pagina se încarcă mult mai repede nemaifiind nevoie să se încarce toate informațiile multimedia.

Sintaxa folosită pentru realizarea celor prezentate mai sus este similară cu cea prezentată în subcapitolul 6.3. (Ancore):

Click here to play multimedia

O metodă specifică, mai complexă, ce are ca rezultat includerea în pagina web (ca parte integrantă din aceasta) a unor informații multimedia, se realizează prin intermediul elementului **<OBJECT>**.

Notă: Deși, în funcție de browser-ul folosit, mai pot fi folosite și alte elemente, cum ar fi <BGSOUND>, recunoscut de Internet Explorer, și <EMBED>, recunoscut de Internet Explorer și Netscape, consorțiul W3C (World Wide Web Consortium) recomandă utilizarea elementului <OBJECT>.

Atributele utilizate cu acest element sunt:

Atribut	Descriere
archive	Precizează locația (URL-ul) fișierelor arhivă.
classid	Precizează locația (URL-ul) obiectului. Valorile posibile depind de proiectantul elementului de control. În Windows, aceste valori sunt înscrise în regiștri și pot fi accesate cu aplicația Registry Editor .
codebase	Precizează calea (URI-ul) în care se găsește clasa obiectului specificat prin atributele <i>classid</i> , <i>data</i> și <i>archive</i> .
codetype	Specifică tipul fișierului specificat prin atributul <i>classid</i> .
data	Precizează numele și locația (URL-ul) obiectului. Se poate folosi împreună sau ca o alternativă a atributului <i>classid</i> , în funcție de tipul obiectului.
declare	Cînd este prezent, acet atribut face ca definiția curentă introdusă de elementul <OBJECT> să fie tratată ca o simplă declarație.
height	Specifică înălțimea suprafeței în care va fi afișat obiectul.
name	Precizează denumirea obiectului (când acesta este cuprins într-un formular).
standby	Precizează textul ce va fi afișat până când este încărcat obiectul.
tabindex	Precizează poziția elementului curent în document (ordinea în care va fi focalizat de utilizator când se folosește tastatura).
type	Determină tipul MIME asociat cu fișierul respectiv. Este un atribut opțional dar recomandat când se folosește atributul <i>data</i> .
width	Specifică lățimea suprafeței în care va fi afișat obiectul.

Acest element necesită în mod obligatoriu tag-ul de închidere corespunzător `</OBJECT>`.

Împreună cu elementul `<OBJECT>` (în interiorul acestuia) se folosește și elementul asociat `<PARAM>` care definește parametrii elementului `<OBJECT>`.

Atributele utilizate cu acest element sunt:

Atribut	Descriere
id	Precizează un identificator unic pentru element.
name	Precizează o denumire pentru element.
type	Determină tipul conținutului.
value	Specifică valoarea parametrului.
valuetype	Specifică tipul valorii parametrului.

Elementul `<OBJECT>` poate suporta diferite tipuri de informații multimedia, cum ar fi imagini, sunete, video, pagini web etc.

În funcție de tipul acestor informații, diferă și sintaxa folosită în limbajul HTML.

Astfel, pentru a insera o imagine (în format .jpg), prin intermediul elementului `<OBJECT>`, se folosește următoarea sintaxă:

```
<OBJECT HEIGHT="..." WIDTH="..." TYPE="image/jpeg"  
DATA="nume_fisier.jpg">  
</OBJECT>
```

Pentru a insera un sunet (în format .wav) se folosește următoarea sintaxă:

```
<OBJECT CLASSID="clsid:22D6F312-B0F6-11D0-94AB-0080C74C7E95">  
<PARAM NAME="FileName" VALUE="nume_fisier.wmv">  
</OBJECT>
```

Pentru a insera un film (în format .wmv) se folosește următoarea sintaxă:

```
<OBJECT CLASSID="clsid:22D6F312-B0F6-11D0-94AB-0080C74C7E95">  
<PARAM NAME="FileName" VALUE="nume_fisier.wmv">  
</OBJECT>
```

Pentru a insera o pagină web se folosește următoarea sintaxă:

```
<OBJECT TYPE="text/html" HEIGHT="..." WIDTH="..."  
DATA="http://www.pagina_web.com">  
</OBJECT>
```

Pentru a insera o animație creată în Flash (în format .swf) se folosește următoarea sintaxă:

```
<OBJECT WIDTH="..." HEIGHT="..."  
CLASSID="clsid:D27CDB6E-AE6D-11cf-96B8-444553540000"  
CODEBASE="http://download.macromedia.com/pub/shockwave/cabs/flash  
/swflash.cab#4,0,0,0">  
<PARAM NAME="SRC" VALUE=" nume_fisier.swf">  
</OBJECT>
```

12.5. Inserarea unui film QuickTime într-o pagină web

Formatul QuickTime a fost dezvoltat de către Apple și, deși este un format des întâlnit pe Internet, nu poate rula pe platformele Windows fără instalarea unui player specializat.

Fișierele de tip QuickTime au extensia .mov sau .qtm.

Cu ajutorul elementului <OBJECT> poate fi ușor adăugat cod într-o pagină web astfel încât un film QuickTime să poată fi inserat într-o pagină web.

Astfel, se poate seta instalarea automată a unui player QuickTime în cazul în care nu este deja instalat.

Sintaxa folosită în mod uzual este de forma următoare:

```
<OBJECT WIDTH="240" HEIGHT="300"  
CLASSID="clsid:02BF25D5-8C17-4B23-BC80-D3488ABDDC6B"  
CODEBASE="http://www.apple.com/qtactivex/qtplugin.cab">  
<PARAM NAME="src" VALUE="sample.mov">  
<PARAM NAME="autoplay" VALUE="true">  
<PARAM NAME="controller" VALUE="true">  
</OBJECT>
```

Atributele **width** și **height** trebuie setate astfel încât să corespundă dimensiunii filmului QuickTime (în pixeli).

Atributul **classid** identifică în mod unic software-ul ce va rula filmul. Trebuie setat "clsid:02BF25D5-8C17-4B23-BC80-D3488ABDDC6B" astfel încât să identifice controlul ActiveX ce trebuie instalat pe computer înainte ca filmul să poată fi rulat. Dacă utilizatorul nu are instalat respectivul control ActiveX, browser-ul poate în mod automat să-l descarce și să-l instaleze.

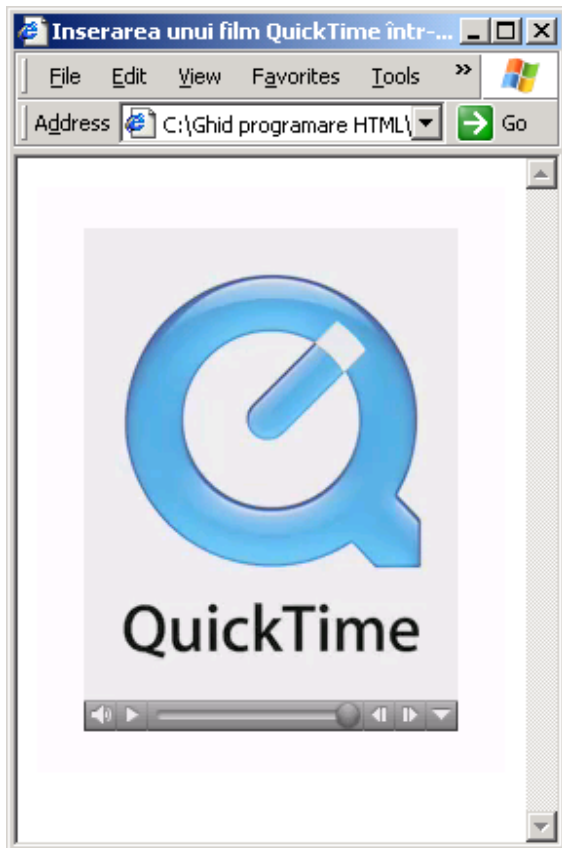
Atributul **codebase** specifică calea (URI-ul) în care se găsește clasa obiectului specificat prin atributul *classid*. Internet Explorer folosește acest atribut pentru a specifica locația de unde poate fi descărcat player-ul corespunzător. În acest caz, trebuie setată "http://www.apple.com/qtactivex/qtplugin.cab", aceasta fiind locația unde se găsește ultima variantă a player-ului QuickTime.

Parametrul **src** trebuie să se orienteze asupra fișierului ce conține filmul.

Parametrul **autoplay** se setează "true" în cazul în care se dorește ca filmul să înceapă să ruleze automat.

La fel și parametrul **controller** se setează "true" în cazul în care se dorește să fie afișată bara de control a player-ului.

În urma inserării codului prezentat mai sus, rezultatul va fi următorul:



12.6. Inserarea unui film Real Video într-o pagină web

Formatul RealVideo a fost dezvoltat pentru Internet de către Real Media.

Formatul permite difuzarea de informații video (video on-line, Internet video) cu lățime de bandă joasă. Din cauza acestei lățimi de bandă, calitatea este, de cele mai multe ori, slabă.

Fișierele de tip RealVideo au extensia .rm sau .ram.

Cu ajutorul elementului <OBJECT> se introduce codul în pagina web astfel încât filmele Real Video să poată fi inserate într-o pagină web.

Spre deosebire de formatul QuickTime, nu se poate seta instalarea automată a player-ului corespunzător în cazul în care nu este deja instalat. Prin urmare, înainte de a putea rula un film în format Real Video, este necesară instalarea programului RealPlayer. Versiunea pentru Windows poate fi descărcată de la adresa <http://uk.real.com/player/win>.

Sintaxa folosită în mod uzual este de forma următoare:

```
<OBJECT WIDTH="320" HEIGHT="240"  
CLASSID="clsid:CFCDA03-8BE4-11cf-B84B-0020AFBBCCFA">
```

```
<PARAM NAME="controls" VALUE="All">  
<PARAM NAME="autostart" VALUE="true">  
<PARAM NAME="src" VALUE="sample.rm">  
</OBJECT>
```

Atributele **width** și **height** trebuie setate astfel încât să corespundă dimensiunii filmului Real Video (în pixeli).

Atributul **classid** identifică în mod unic software-ul ce va rula filmul. Trebuie setat "clsid: CFCDA03-8BE4-11cf-B84B-0020AFB8CCFA" astfel încât să identifice controlul ActiveX ce trebuie instalat pe computer înainte ca filmul să poată fi rulat.

Parametrul **src** trebuie să se orienteze asupra fișierului ce conține filmul.

Parametrul **autostart** se setează "true" în cazul în care se dorește ca filmul să înceapă să ruleze automat.

Parametrul **controls** se setează "All" în cazul în care se dorește să fie afișată bara de control a player-ului sau "ImageWindow " în cazul în care NU se dorește să fie afișată bara de control a player-ului.

În urma inserării codului prezentat mai sus, rezultatul va fi următorul:



12.7. Alte metode de inserare a unor sunete într-o pagină web

Așa cum am atras atenția anterior, pentru introducerea de sunete într-o pagină web mai pot fi folosite și alte elemente, cum ar fi <BGSOUND>, recunoscut de Internet Explorer, și <EMBED>, recunoscut de Internet Explorer și Netscape.

Elementul **<BGSOUND>** este folosit pentru a insera un sunet pe fundalul unei pagini web. Fișierele suportate de acest element pot fi în format WAV, AU sau MID.

Atributele caracteristice acestui element sunt:

Atribut	Descriere
balance	Precizează balansul. Poate lua valori între -10000 (100% stânga) și +10000 (100% dreapta).
delay	Definește timpul de pauză dintre repetiții.
id	Definește un identificator atribuit sunetului.
loop	Precizează de câte ori va fi repetat sunetul (-1 = infinit).
src	Precizează locația (URL-ul) fișierului ce conține sunete.
title	Precizează un titlu atribuit sunetului.
volume	Precizează volumul. Poate lua valori între 0 (max.) și -10000 (min.).

Sintaxa folosirii acestui element este foarte simplă:

```
<HTML>
<BODY>
<H2>Muzica in fundal </H2>
<BGSOUND SRC="town.mid">
</BODY>
</HTML>
```

Folosind codul de mai sus veți obține o pagină web pe fundalul cărei se va auzi o piesă instrumentală.

Elementul **<EMBED>** este folosit pentru a insera elemente multimedia într-o pagină web în cazul browsere-lor care nu suportă elementul **<OBJECT>**.

Elementul **<OBJECT>** poate fi folosit cu browserele mai noi, care suportă controale ActiveX (Internet Explorer ≥ 5), în vreme ce elementul **<EMBED>** este recomandat să fie folosit pentru browserele mai vechi.

Pentru compatibilitate, se recomandă folosirea simultană a celor două elemente. În această situație, un browser care recunoaște elementul **<OBJECT>** va ignora elementul **<EMBED>**.

Atributele caracteristice acestui element sunt:

Atribut	Descriere
autostart	Stabilește dacă obiectul multimedia pornește în mod automat la încărcarea paginii web. Poate lua valorile <i>true</i> sau <i>false</i> .
height	Specifică înălțimea suprafeței în care va fi afișat obiectul.
hidden	Stabilește dacă obiectul multimedia este vizibil pentru vizitatorul paginii

	web. Poate lua valorile <i>true</i> sau <i>false</i> .
src	Precizează locația (URL-ul) fișierului ce conține sunete.
width	Specifică lățimea suprafeței în care va fi afișat obiectul.
volume	Precizează volumul. Poate lua valori între 0 (max.) și -10000 (min.).

Împreună cu elementul <EMBED> se recomandă folosirea elementului <NOEMBED> care ar trebui să conțină informații ce vor fi afișate în cazul în care browser-ul nu suportă elementul <EMBED>.

Din nefericire, un browser care suportă elementul <EMBED> va ignora elementul <NOEMBED>, *chiar dacă este incapabil să afișeze obiectul introdus de elementul <EMBED>*.

Sintaxa recomandată pentru folosirea acestui element este următoarea:

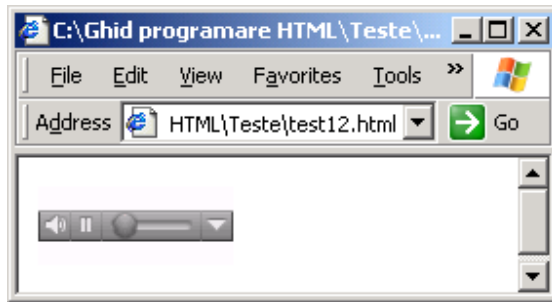
```
<EMBED SRC="cale/fișier_multimedia" WIDTH="..." HEIGHT="..." >
<NOEMBED>
<P>Din păcate nu aveți instalat plugin-ul corespunzător pentru a vedea acest
tip de fișier multimedia. Puteți vedea în continuare o imagine edificatoare.
<IMG SRC="cale/fișier_imagine" WIDTH="..." HEIGHT="..." ALT="o
imagine edificatoare">
</NOEMBED>
```

Se poate folosi chiar și o combinație între elementele <EMBED> și <BGSOUND>, astfel încât, dacă unul din elemente nu este suportat de browser, să fie recunoscut celălalt.

Folosirea codului următor

```
<HTML>
<BODY>
<EMBED SRC="town.mid" WIDTH="100" HEIGHT="40" >
<NOEMBED>
<H2>Muzica în fundal </H2>
<BGSOUND SRC="town.mid">
</NOEMBED>
</BODY>
</HTML>
```

va avea ca rezultat afișarea următorului rezultat:



Notă: Bara de control afișată aparține player-ului QuickTime deoarece a fost setat ca "Default Player" pentru fișierele de tip .MID.

12.8. Aplicație practică

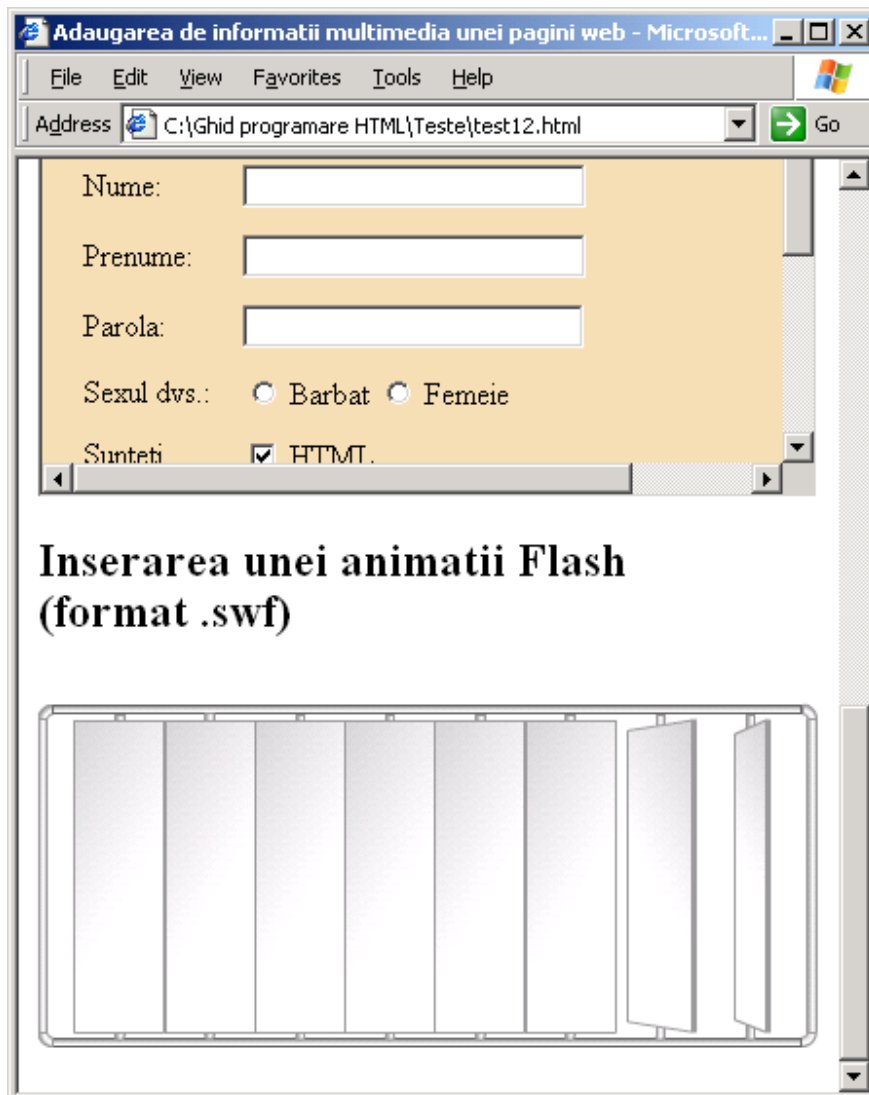
Pentru exemplificare, vom crea un fișier **test12.html** ce va cuprinde o parte dintre elementele prezentate anterior:

```
<HTML>
<HEAD>
<TITLE>Adaugarea de informatii multimedia unei pagini web</TITLE>
</HEAD>
<BODY>
<H2>Inserarea unei imagini</H2>
<OBJECT HEIGHT="200" WIDTH="200" TYPE="image/jpeg"
DATA="computer.jpg">
</OBJECT>
<BR>
<H2>Inserarea unui sunet (format .wav)</H2>
<OBJECT CLASSID="clsid:22D6F312-B0F6-11D0-94AB-0080C74C7E95">
<PARAM NAME="FileName" VALUE="horse.wav">
  <EMBED SRC="horse.wav" WIDTH="100" HEIGHT="40" >
  <NOEMBED>
    Browser-ul dvs. nu suportă elementele OBJECT și nici EMBED
  </NOEMBED>
</OBJECT>
<BR>
<H2>Inserarea unui film (format .wmv)</H2>
<OBJECT CLASSID="clsid:22D6F312-B0F6-11D0-94AB-0080C74C7E95">
<PARAM NAME="FileName" VALUE="movie.wmv">
</OBJECT>
<BR>
<H2>Inserarea unei pagini web</H2>
<OBJECT TYPE="text/html" HEIGHT="200" WIDTH="400"
DATA="test10.html">
</OBJECT>
```

```
<BR>
<H2>Inserarea unei animatii Flash (format .swf)</H2>
<OBJECT WIDTH="400" HEIGHT="200"
CLASSID="clsid:D27CDB6E-AE6D-11cf-96B8-444553540000"
CODEBASE="http://download.macromedia.com/pub/shockwave/cabs/flash/swfla
sh.cab#4,0,0,0">
<PARAM NAME="SRC" VALUE="flash.swf">
</OBJECT>
</BODY>
</HTML>
```

Rezultatul afișat în browser-ul IE va fi următorul:





CAP. 13. ELEMENTE SPECIALE

13.1. Meta-Elementele

Scopul elementelor **meta** este de a oferi informații despre conținutul documentului web. Mai poartă numele de **metadata**, adică sunt date care descriu alte date.

Sunt marcate prin tag-ul <META>, nu necesită tag de închidere, iar folosirea lor este **opțională**. Într-un document web pot fi incluse un număr nelimitat de astfel de elemente.

Aceste elemente pot fi incluse **numai** în interiorul secțiunii <HEAD> a unui document web.

Deși oferă informații relevante pentru browser și pentru motoarele de căutare, descriind conținutul documentului web, acestea nu sunt afișate de către browsere.

Cele mai importante atribute ale acestui element sunt:

Atribut	Valoare	Descriere
name	author description keywords generator revised others	- autorul documentului - descrierea documentului - cuvinte cheie - programul cu care a fost generat codul - data ultimei actualizări (eventual și autorul) - etc.
http-equiv	content-type expires refresh set-cookie	- tipul și setul de caractere - data și ora la care expiră documentul - specifică timpul și un alt URL ce va fi încărcat - setează numele și valoarea pentru cookie-uri

Caracteristic acestui element este faptul că fiecare cuprinde o pereche de atribute. Astfel, descrierea unui document web se poate realiza folosind următoarea sintaxă:

<META NAME="description" CONTENT="conținut">

Definirea cuvintelor cheie, utilă pentru unele motoare de căutare, se poate realiza folosind următoarea sintaxă:

<META NAME="keywords" CONTENT="aici înșiruiți cuvinte cheie relevante despre conținutul documentului web separate prin virgulă">

Datorită abuzurilor unor webmasteri, prin repetarea în mod exagerat a unor cuvinte cheie, multe din motoarele de căutare existente astăzi au încetat să mai considere relevantă informația cuprinsă în această secțiune și se axează pe analiza conținutului documentului.

În cazul în care doriți să redirecționați vizitatorii unei pagini către o altă pagină în mod automat, se folosește atributul **HTTP-EQUIV** cu valoarea **refresh**, în următoarea sintaxă:

**<META HTTP-EQUIV="refresh" CONTENT="timp;
URL=pagina_noua.html">**

Variabila *timp* specifică timpul (în secunde) după care va fi încărcat noul URL specificat în variabila *pagina_noua.html*.

Atenție la folosirea ghilimelelor care cuprind întregul conținut al atributului CONTENT și la punctul și virgula care separă cele două variabile.

Este recomandat ca pentru utilizatorii cu browsere din vechile generații care nu fac refresh în mod automat să adăugați un link în secțiunea <BODY> a documentului, astfel încât vizitatorii să poată fi redirecționați manual către o altă pagină web.

Pentru a insera informații despre tipul de document web (valoarea prestabilită a acestuia este *text/html*) și despre setul de caractere utilizat în document, se atribuie valoarea *content-type* atributului *HTTP-EQUIV*, iar variabilele asociate atributului *CONTENT* vor conține respectivele informații în următoare sintaxă:

**<META HTTP-EQUIV="content-type" CONTENT="text/html;
charset=windows-1250">**

Setul de caractere windows-1250 conține setul de caractere utilizat în Europa Centrală și de Est (Belarus, Croația, Cehia, Ungaria, Polonia, România, Serbia, Slovacia, Slovenia).

13.2. Inserarea comentariilor

În multe situații este necesară inserarea de comentarii încodul sursă al unui document web. Astfel de comentarii sunt ignorate de către browser. Puteți folosi comentariile pentru a explica codul scris la un moment dat, ceea ce v-ar putea ajuta ulterior, în cazul în care doriți să modificați codul.

De asemenea, în interiorul comentariilor pot fi stocate informații specifice pentru anumite programe, care nu vor fi vizibile pentru utilizatori dar vor fi totuși interpretate de programele respective. O practică des întâlnită este de a insera scripturile Javascript și elementele de stil în interiorul comentariilor, astfel încât browserele care nu recunosc respectivele elemente să nu le afișeze.

Tag-urile care definesc un comentariu sunt **<!--** și respectiv **-->**.

Comentariile pot fi plasate oriunde în interiorul unui document web.

Sintaxa uzuală este de forma:

<!-- În această secțiune este cuprins un comentariu -->.

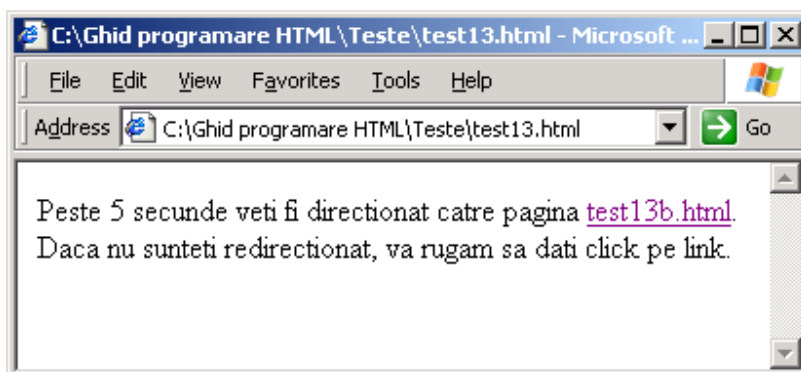
Notă: *Vă recomand să folosiți totuși cu măsură comentariile în interiorul unui document web deoarece, ca orice alt element, mărește dimensiunea fișierului HTML ceea ce duce, implicit, la un timp mai mare de încărcare în browser.*

13.3. Aplicație practică

Pentru a exemplifica cele prezentate anterior, vom crea un fișier **test13.html** ce va avea următorul conținut:

```
<HTML>
<HEAD>
<META HTTP-EQUIV="refresh" CONTENT="5;URL=test13b.html">
<!-- Dupa cum observati, am inlocuit variabila timp cu 5 (ceea ce semnifica 5
secunde) si variabila pagina_noua.html cu test13b.html -->
<META HTTP-EQUIV="content-type" CONTENT="text/html;
charset=windows-1250">
<!-- Am specificat tipul de document HTML si setul de caractere folosit utilizat
pentru Europa Centrala si de Est -->
<META NAME="description" CONTENT="Pagina web ce prezinta exemple de
folosire a unor elemente speciale HTML">
<!-- Am inserat o descriere pe scurt a continutului documentului -->
<META NAME="keywords" CONTENT="HTML, web, refresh, meta,
comentariu, exemplu">
<!-- Am adaugat cateva cuvinte cheie despartite prin virgula -->
</HEAD>
<BODY>
<P>Peste 5 secunde veti fi directionat catre pagina <A
HREF="test13b.html">test13b.html</A>.
<!-- Pentru browserele care nu suporta REFRESH, am adaugat link-ul respectiv
-->
<BR>Daca nu sunteti redirectionat, va rugam sa dati click pe link.</P>
</BODY>
</HTML>
```

Rezultatul afișat în browser-ul IE va fi următorul:

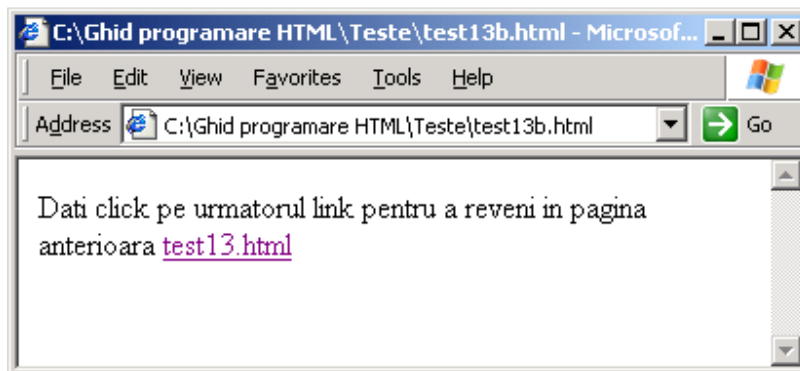


Fișierul **test13b.html** are următorul conținut:

```
<HTML>
```

```
<BODY>  
<P>Dati click pe urmatorul link pentru a reveni in pagina anterioara <A  
HREF="test13.html">test13.html</A></P>  
</BODY>  
</HTML>
```

Rezultatul afișat în browser-ul IE va fi următorul:



CAP. 14. CONSIDERAȚII GENERALE DESPRE DESIGN ȘI ASPECT

Este simplu să faci o pagină web, dar este important să faci o pagină web **bună**. De ce afirm că este simplu să faci o pagină web? Pentru că orice are un editor HTML de tipul WYSIWYG (What You See Is What You Get) poate realiza o pagină web fără a fi nevoit să înțeleagă complexitatea unei pagini web privită ca un întreg.

În acest capitol vom face o scurtă trecere în revistă a aspectelor ce determină realizarea unei pagini web bune și a problemelor cu care v-ați putea confrunța.

14.1. Diferențe dintre un document web și unul tipărit

În contrast cu un document tipărit, un document web este ceva complet diferit. În acest caz, designerul are prea puțin control asupra felului în care este afișată o pagină, întrucât acest lucru depinde în mare măsură de browserul web folosit. Deși frustrant, a face o pagină web să fie afișată la fel în diferite browsere este un lucru extraordinar, constituind o provocare pentru majoritatea webdesignerilor.

Când creează un site, un webdesigner trebuie să țină cont de următoarele:

- Varietatea browserelor
- Varietatea versiunilor pentru același tip de browser
- Varietatea sistemelor de operare folosite
- Varietatea adâncimii culorii pe monitoarele vizitatorilor
- Varietatea rezoluțiilor alese de diverși utilizatori
- Etc.

Cu un număr atât de mare de variabile de care depinde afișarea unei pagini web, nu este posibil întotdeauna ca o pagină web să fie afișată în mod identic pe două calculatoare diferite. Și, așa cum am precizat și anterior, acest lucru constituie o adevărată provocare pentru webdesigneri.

În orice caz, trebuie să țineți cont ca un site nu este făcut pentru voi sau pentru șeful vostru. Numai pentru că există o introducere făcută în Flash pe un site nu înseamnă că trebuie să aveți și voi o introducere similară.

Un site trebuie creat în funcție de publică țintă căruia i se adresează. Aceștia pot fi clienți sau potențiali clienți. Un bun webdesigner își privește rezultatul muncii prin ochii unui potențial client.

De exemplu, dacă concepeți un site adresat persoanelor mai în vârstă, trebuie să aveți grijă să oferiți site-ului un aspect simplu, ușor de urmărit, cu fonturi puțin mai mari decât de obicei.

Trebuie să analizați cu regularitate statisticile referitoare la site pentru a vedea ce browsere sunt folosite de majoritatea vizitatorilor, ce sisteme de operare folosesc, dacă folosesc platforma Javascript de la Sun, sau dacă au activate anumite controale ActiveX. Pentru a obține astfel de informații puteți folosi fie scripturile proprii, fie înscrieți propriul site într-unul dintre site-urile de analiză existente pe Internet (de exemplu www.traffic.ro sau www.statistics.ro).

14.2. Mărimea optimă a unei paginii web și optimizarea ei

Optimizarea paginilor web este unul din aspectele cele importante ale webdesignului. O pagină web optimizată se încarcă mult mai repede pe calculatorul vizitatorului și astfel informația ajunge mult mai repede la țintă. Paginile care se încarcă greu, și aici facem referire la paginile care se încarcă în mai mult de 10 secunde, sau care au o introducere făcută în Flash, pot duce la pierderea unei părți dintre vizitatori, în special a celor care nu au suficientă răbdare. Nimic nu-i împiedică să apese unul din butoanele **Back** sau **Close Window**. Cu toate acestea, introducerile create în Flash oferă și avantaje prin faptul că pot fi foarte atractive din punct de vedere vizual și oferă posibilitatea folosirii unor meniuri interactive.

Pentru a stabili mărimea optimă a unei pagini web trebuie să aveți în vedere cei patru factori care influențează încărcarea într-un browser:

- mărimea fișierului care conține codul HTML
- lățimea de bandă a conexiunii vizitatorilor
- viteza procesorului serverului care găzduiește pagina web
- complexitatea paginii web

Din moment ce nu puteți interveni asupra conexiunii vizitatorilor, trebuie să vă ocupați de ceilalți factori. Astfel, încercați să găzduiți pagina web un server foarte rapid, optimizați pagina web prin “curățarea” codului HTML de elementele redundante, de grafica excesivă, de elementele multimedia care nu sunt strict necesare etc. ceea ce va avea ca rezultat micșorarea dimensiunii fișierului sursă.

Dacă este vorba despre un site care se adresează tuturor categoriilor de utilizatori, atunci mărimea fiecărei pagini web nu trebuie să depășească **75 kb**. Astfel, veți avea garanția că fiecare pagină se încarcă în doar câteva secunde.

La calcularea mărimii unei pagini web se iau în calcul nu numai fișierul HTML, ci și toate elementele grafice și multimedia folosite, inclusiv imaginile de fundal, inclusiv scripturile Javascript externe (fișiere cu extensia .js), inclusiv foile de stil (fișiere cu extensia .css), scripturi PHP etc.

14.3. Structura și aspectul unui site

Structura directoarelor trebuie să fie logică și să reflecte aspectul propriului site.

Un site poate începe ca o simplă pagină web, se poate dezvolta într-un set de câteva pagini, iar în timp, în funcție de cât de importantă doriți să fie prezența voastră pe Internet, poate ajunge chiar și la sute de pagini. Coordonarea unui număr mare de pagini poate fi un lucru extrem de dificil mai ales dacă nu se lucrează într-un mod coerent de la început.

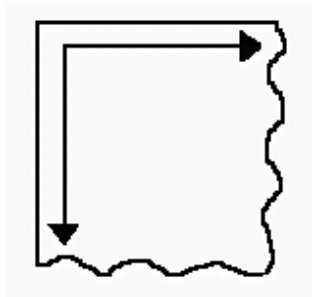
Astfel, trebuie evitat amestecul tuturor fișierelor într-un singur director, fiind necesară gruparea lor în funcție de caracteristicile similare.

De exemplu, toate imaginile incluse într-un site trebuie grupate într-un director separat, cum ar fi *imagini*. Acesta, la rândul său, se poate împărți în subdirectoare cu denumiri sugestive: *pictograme*, *butoane* sau *fundaluri*.

În mod similar pot fi separate și fișierele de alt tip, cum ar fi cele Javascript, PHP sau ASP, care pot fi incluse în directoare cu nume sugestive.

Denumirea dată unui director sau subdirector depinde numai de voi și, chiar dacă este consumatoare de timp, cu siguranță va ajuta la înțelegerea structurii site-ului de către oricine va dori să-i aducă modificări.

Referitor la aspect, trebuie ținut cont de faptul că o pagină web începe în colțul din stânga sus și se poate întinde oricât spre dreapta sau în jos. Acesta face să poată fi infinit de lată și de lungă.



Este recomandabil ca o pagină să nu necesite derulare spre dreapta, de aceea, înainte de a publica o pagină web pe Internet, trebuie testat modul în care se comportă în funcție de diferite rezoluții ale monitorului.

Dintre acestea, cele mai uzuale sunt:

- 640 x 480 (valabilă pentru sistemele foarte vechi)
- 800 x 600
- 1024 x 768
- 1280 x 1024
- 1600 x 1280
- Altele (valabile pentru Mac și Web TV)

14.4. Combinarea culorilor

Alegerea combinației de culori potrivite este esențială pentru orice site. Aceasta trebuie să reflecte, pe cât de mult posibil, conținutul site-ului. O bună combinație de culori poate deschide drumul spre succes, în vreme ce una aleasă în mod nefericit poate duce la eșecul unui site.

Asigurați-vă că alegeți într-un mod chibzuit combinația de culori care să se potrivească temei site-ului. Aceste culori trebuie să fie complementare culorilor folosite pentru logo-ul ales sau în contrast cu acesta, tocmai pentru a-l evidenția.

Rețineți că efectul unei singure culori se schimbă, uneori chiar dramatic, când este folosită în combinație cu alte culori.

De exemplu, culoarea **roșie** este evitată de obicei de site-urile ce oferă informații financiare, în vreme ce combinațiile de **albastru** și **gri** au fost întotdeauna o bună alegere pentru site-urile profesionale.

14.5. Tipurile de caractere folosite

Dacă simțiți un impuls irezistibil să folosiți un tip de caracter (font) incredibil de aspectuos sau unul care reproduce scrisul de mână, sfatul meu este să vă rețineți un astfel de impuls.

Una dintre multitudinea de variabile din webdesign o constituie și disponibilitatea unor anumite fonturi pe computerele vizitatorilor.

De exemplu, dacă folosiți pentru afișarea textului un font ce se găsește în computerul vostru, cum ar fi *Palatino*, dar acesta nu se găsește și în computerul unui

vizitator, browserul va înlocui fontul respectiv cu altul implicit, cum ar fi *Times New Roman* sau *Arial*. Acest lucru va avea drept consecință schimbarea drastică a aspectului paginii afișate și toată străduința voastră pentru realizarea unui site atractiv va rămâne fără rezultat.

Pentru o maximă compatibilitate a tipurilor de caractere între diverși utilizatori și diverse browsere, sunt recomandate trei familii de fonturi:

- **Familia fonturilor fără piciorușe (Sans-serif)**
Arial, Verdana, Geneva, Helvetica, Sans-serif
- **Familia fonturilor cu piciorușe (Serif)**
Times, Times New Roman, Georgia, Serif
- **Familia fonturilor fără monospațiate (Monospace)**
Courier, Courier New, Monospace

Evident, există și metode de a folosi fonturi deosebite într-o pagină web. Cea mai simplă dintre acestea este de a transforma textul respectiv într-o imagine, folosind o serie de programe adecvate. Dar, cum totul are un preț, acesta va contribui la creșterea dimensiunii paginii web și, implicit, la o încărcare mai lentă.

Am tot amintit de diversitatea browserelor existente. Nu vă oprește nimeni să descărcați o copie a fiecăruia dintre ele, mai ales că sunt gratuite, și să le instalați în propriul computer. Astfel veți putea testa propriul site pe câteva dintre cele mai folosite browsere (Internet Explore, Netscape Navigator, Mozilla Firefox, Opera).

ÎNCHEIERE

În cazul în care acest ghid și-a atins scopul și ați reușit să deprindeți cele mai importante noțiuni legate de limbajul HTML, înseamnă că deja puteți crea un site.

Tot ce vă mai rămâne de făcut este să găsiți un server gazdă, unul gratuit pentru început, unde să postați site-ul.

Dintre acestea vă recomandăm:

www.3x.ro

www.home.ro sau www.go.ro

www.idilis.ro

www.rol.ro

Dupa ce v-ați găsit o gazdă pentru a vă plasa site-ul sau pagina web, trebuie să începeți crearea paginilor. Este indicat să aveți două copii ale documentului. O copie trebuie să fie pe serverul gazdei iar a doua pe computerul personal, astfel încât problemele cauzate de căderile de curent, pierderea conexiunii la internet să nu vă afecteze munca.

În continuare, urmând recomandările din capitolul precedent, va trebui să creați un director pe hard disk pentru a stoca documentele HTML, imaginile, sunetele, precum și celelalte elemente pe care doriți să le includeți în paginile web.

Pagina principală trebuie denumită "index.html", majoritatea serverelor fiind setate să recunoască pagina principală după această denumire. În cazul în care "index.html" nu funcționează ca pagină principală, contactați persoana care administrează server-ul și întrebați-l ce denumire implicită are setată server-ul pentru a recunoaște pagina principală.

Cea mai bună experiență în webdesign o veți căpăta navigând pe Internet. Puteți salva o copie completă a site-urilor care vă rețin atenția și le puteți studia codul sursă pentru a analiza (**nu copia**) modul în care au fost făcute.

Vă doresc succes!