

Elemente CSS

Cascading Style Sheets (CSS) este practic un —pseudo-limbaj|| de formatare, utilizat pentru a descrie modul de prezentare a documentelor scrise într-un limbaj bazat pe marcate (HTML, XML etc.). Fișierele CSS (foi de stil în cascadă) permit separarea conținutului HTML propriu-zis al unui document, de stilul de afișare/formatare în pagină al acestuia. Codul HTML se utilizează, de obicei, doar pentru aranjarea conținutului în pagină, iar detaliile care țin de afișare (culori, fonturi, fundaluri, margini etc.) se realizează cu ajutorul CSS-urilor, acestea aplicându-se suplimentar peste codului HTML, în cadrul unui site Web. Cu alte cuvinte, CSS-urile realizează separarea prezentării paginii de structura sa efectivă. Aplicarea foilor de stil în cascadă asupra codului HTML se poate face în mai multe moduri, putându-se vorbi de:

- stiluri interne;
- stiluri externe;
- stiluri în linie;
- clase CSS.

Stiluri interne

În cazul utilizării stilurilor interne, codul CSS se plasează în interiorul fiecărei pagini HTML pe care se dorește să se aplice stilurile respective, între tag-urile `<head>` `</head>`, după cum se poate vedea în continuare:

`<head>`

```
<title>Exemplu utilizare stiluri interne</title>
```

```
<style type="text/css">Aici se definesc stilurile CSS</style>
```

```
</head>
```

Pentru exemplificare, se prezintă un script HTML în care se utilizează stiluri interne:

```
<html>
```

```
<head>
```

```
<title> Exemplu de utilizare a stilurilor interne!!! </title>
```

```
<style type="text/css">
```

```
table {
```

```
font-family: "Ionescu Script ITC";
```

```
font-size: 36px;
```

```
color: #FFFFFFF;
```

```
background-color: #0099FF;
```

```
border: 4px double #0033FF;
```

```
text-align: center;
```

```
}
```

```
</style>
```

```
</head>
```

```
<body>
```

```
<br><br>
```

```
<table align="center">
```

<tr>

<td>

Exemplu de utilizare a stilurilor interne

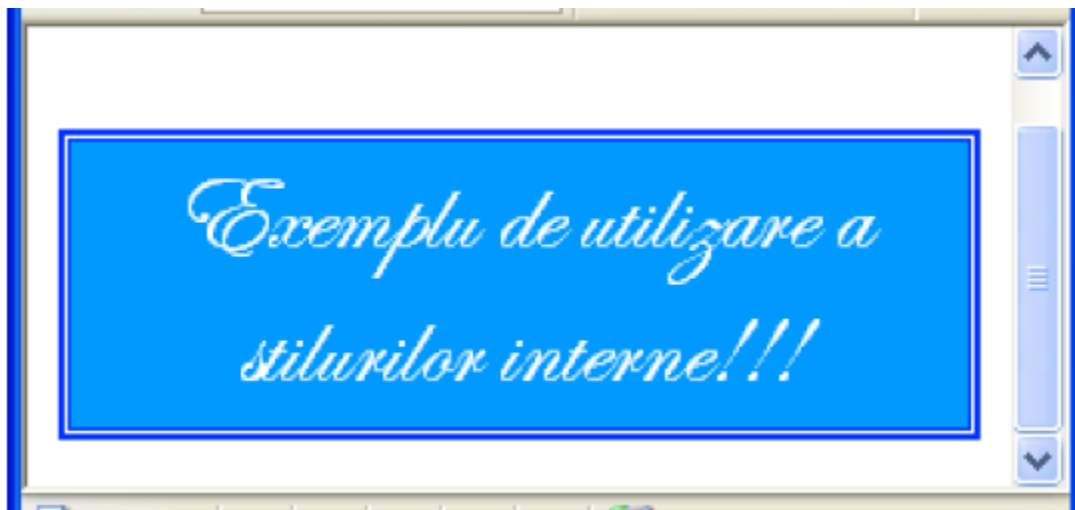
</td>

</tr>

</table>

</body>

</html>



Utilizând aceasta metodă de aplicare a CSS-urilor asupra codului HTML, dacă

se dorește o schimbare a stilului de afișare (mărimea fontului, culoare, etc)

modificarea va trebui realizată în toate paginile care conțin acel stil. Ținând cont de aceste aspect, această metoda este indicat a fi folosită doar în situația în care se dorește —stilizarea|| unui număr mic de pagini, fiind destul de neproductivă o realizare a acestor modificări pe zeci sau chiar sute de pagini ale unui site WEB.

Stiluri externe

Un fișier CSS extern poate fi scris cu orice editor simplu de text (Notepad, Wordpad, etc) sau cu editoare specializate (gen *Dreamweaver*). Fișierul CSS nu conține cod HTML, ci doar cod CSS și trebuie salvat cu extensia *.css*.

Referirea fișierului extern CSS în paginile HTML se face prin plasarea unui tag *link* (legătură), în secțiunea `<head>` `</head>` a fiecărei pagini în cadrul căreia se dorește aplicarea stilul respectiv, având forma următoare:

```
<link rel="stylesheet" type="text/css" href="Calea catre fisierul.css" />
```

În continuare, se prezintă un exemplu de referire a unui fișier extern *.css* într-o pagină HTML:

```
<html>
```

```
<title> Exemplu de utilizare a stilurilor externe!!! </title><head>
```

```
<link href="cssExt.css" rel="stylesheet" type="text/css">
```

```
</head>
```

```
<body>
```

Exemplu de utilizare a stilurilor externe in body!!!

```
<table>
```

```
<tr>
```

```
<td>
```

Exemplu de utilizare a stilurilor externe in tabel!!!

```
</td>
```

```
</tr>
```

```
</table>
```

```
</body>
```

```
</html>
```

Fișierul CSS referit și utilizat în pagina HTML anterioară (*cssExt.css*), se

consideră a avea următorul conținut exemplificativ (efectul utilizării lui putându-se observa în imagine):

```
body
```

```
{
```

```
font-family: "Courier";
```

```
font-size: 26px;
```

```
color: #000000;
```

```
background-color: #B3B3B3;
```

```
text-align: center;
```

```
}
```

```
table {
```

```
font-family: "Edwardian Script ITC";
```

```
font-size: 46px;
```

```
color: #FFFFFF;
```

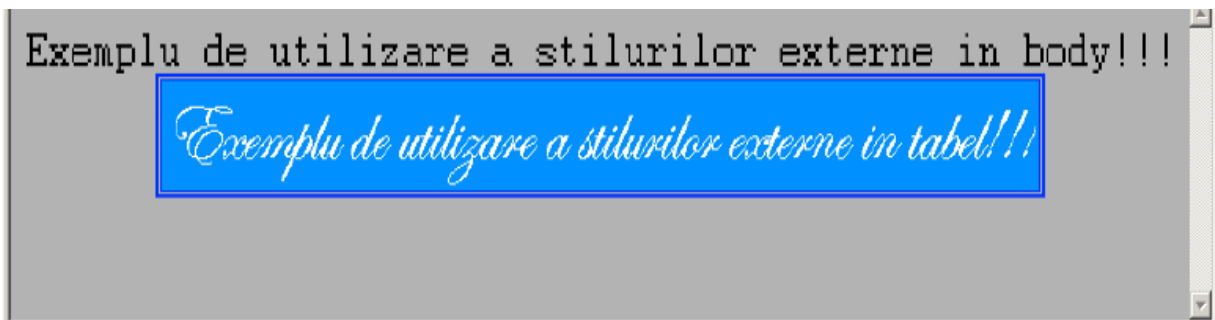
```
background-color: #0090FF;
```

```
border: 4px double #0033FF;
```

```
text-align: center;
```

```
}
```

Aceasta metodă. de utilizare a unor fișiere de stil externe, este preferată în momentul în care un site WEB conține un număr mare de pagini utilizând aceleași reguli de stil, motivul fiind evident: atunci când se dorește modificare aspectului întregului site, este suficientă doar o modificare într-o singură locație, și anume - fișierului CSS (efectul resimțindu-se asupra tuturor paginilor din site care folosesc foaia de stil respectivă). Astfel, printr-o singură operație, se poate schimba rapid aspectul întregului site, indiferent de dimensiunea lui (număr de pagini).



Stiluri în linie

Stilurile în linie se definesc chiar în codul etichetei HTML aferente elementului care se dorește a fi stilizat, după cum se poate vedea în exemplul următor:

```
<body>
```

```
<p style="color: #00ddff; font-size: 20;">Titlu</p>
```

```
<h2 style="font-size: 16;font-weight: bold; color: #ff3300;">Exemplu
```

```
utilizare stiluri in linie!!! </h2>
```

```
</body>
```



Stilurile în linie sunt mai puțin utilizate, deoarece ele nu permit schimbări rapide de stil pe mai multe fișiere în același timp, modificările trebuind realizate pe fiecare element în parte, și în fiecare pagină în parte.

Anterior au fost prezentate trei metode de aplicare a CSS-urilor asupra codului HTML. În situația în care, se folosesc două sau chiar trei metode în același timp, se pune întrebarea: care este ordinea/prioritatea folosirii lor pentru o interpretare corectă de către browser?

Răspunsul este: metodele se vor executa în cascadă, în ordinea următoare: prima oară -stilurile în linie, urmate apoi de stilurile interne, iar în final - stilurile externe, aceasta fiind și ordinea lor de prioritizare.

Evident, un element deja stilizat, spre exemplu, cu un stil linie, nu este restilizat de o regulă de stil existentă într-un fișier CSS extern, acționând imediat ulterior conform regulii de prioritizare anterior enunțate.

Clase CSS

Clasele CSS se utilizează pentru stilizarea în mod diferențiat a unor mulțimi de tag-uri HTML (distribuite în una sau mai multe pagini WEB). Acest mod de lucru este similar cu utilizarea stilurilor în linie, avantajul major fiind acela că atunci când se dorește efectuarea unei modificări de stil pe mai multe elemente/pagini, aceasta nu trebuie efectuată individual la nivelul fiecărui element. Astfel, este

suficientă o modificare în cadrul clasei CSS care definește stilurile respective, efectul acesteia răsfrângându-se asupra tuturor elementelor pe care acționează clasa respectivă.

Definirea unei clase CSS începe cu semnul punct (.), după care urmează numele clasei. Se recomandă folosirea unor denumiri sugestive pentru numele clasei, care să indice ce anume face stilul respectiv. O clasă CSS poate fi folosită în cadrul unui fișier HTML ori de câte ori este nevoie. În continuare prezentăm un exemplu de clasă care stabilește dimensiunea și culoarea unui text:

```
.text20albastru  
{  
font-size: 20px;  
color: 00ddff;  
}
```

Pornind de la exemplificarea din la paragraful —Stiluri în linie, se prezintă modul de definire și utilizare a unor clase CSS într-un document HTML, clasele fiind stocate într-un fișier de stil extern:

- Fișierul HTML utilizând stiluri externe bazate pe clase:

```
<head>  
  
<title> Exemplu de utilizare a stilurilor in linie!!! </title>  
  
<link href="claseCSS.css" rel="stylesheet" type="text/css">  
  
</head>  
  
<body>  
  
<p class="text20albastru">Titlu</p>
```



```
<h2 class="text16rosu">Exemplu utilizare stiluri in linie!!! </h2>
```

```
</body>
```

- Fișierul CSS (*claseCSS.css*) în care sunt definite cele două clase (ambele stilizând texte, dar în mod diferit):

```
.text20albastru
```

```
{
```

```
font-size: 20px;
```

```
color: 00ddff;
```

```
}
```

```
.text16rosu
```

```
{
```

```
font-size: 16px;
```

```
font-weight: bold;
```

```
color: ff3300;
```

```
}
```

Stilizarea obținută în cadrul fișierului HTML anterior prezentat este evident identică cu cea obținută prin utilizarea stilurilor în linie.

Ca o concluzie, în contextul dezvoltării unor aplicații Web tot mai complexe, conținând un număr tot mai mare de pagini, și implicit de fișiere script, stilurile CSS constituie la ora actuală strategia consacrată, de maximă eficiență, pentru formatarea și designul primar al acestora.

Meniuri create cu CSS

Una dintre aplicabilitățile cele mai uzuale ale CSS-urilor constă în crearea de meniuri necesare navigării într-o aplicație, atât foarte simple, cât și cu o

complexitate deosebită. Scheletul HTML (ca fundament al unui meniu) pe care sunt aplicate stilurile CSS, constă în structuri de tip listă (folosindu-se etichete pentru crearea unor liste neordonate , împreună cu elementele lor constitutive), încapsulate eventual într-un DIV, respectiv hyperlink-urile aferente (ancore <a>).

Din păcate dependența de browser se face resimțită și în cazul utilizării de stiluri CSS, astfel încât (mai ales pentru meniuri mai complexe), este foarte posibil ca un meniu care funcționează pe o familie de browsere, să nu fie complet funcțional pe o alta. Într-un astfel de caz, soluția de rezolvare constă în apelarea suplimentară a unor scripturi JavaScript.

Exemplul următor, utilizând doar CSS (fără elemente JavaScript), implementează un meniu simplu (fără submeniuri), fiind funcțional atât pe Mozilla, cât și pe Internet Explorer. Pentru crearea meniului s-a utilizat un fișier extern CSS, în care sunt definite mai multe stiluri.

Fișierul HTML (aferent unui meniu cu trei opțiuni) are următorul cod:

```
<link href="menu-html.css" rel="stylesheet" type="text/css">
```

```
<div id="nav-menu">
```

```
<ul>
```

```
<li><a href="#">Optiune1</a></li>
```

```
<li><a href="#">Optiune 2</a></li>
```

```
<li><a href="#">Optiune 3</a></li>
```


</div>



Fișierul implementând stilurile externe CSS (*menu-html.css*) conține următoarele elemente de stilizare:

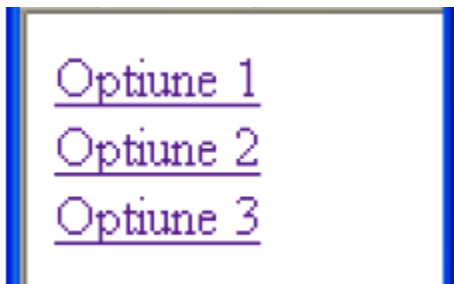
```
#nav-menu ul
```

```
{
```

```
padding: 0;
```

```
margin: 0;
```

```
}
```



```
#nav-menu li
```

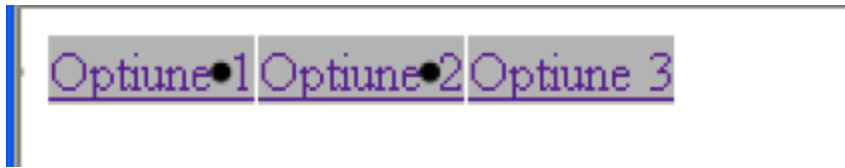
```
{
```

```
float: left;
```

```
margin: 0 0.4;
```

```
background:#B3B3B3;
```

```
}
```



#nav-menu li a

{

background: url(background.jpg) #fff bottom left repeat-x;

height: 2em;

line-height: 2em;

float: left;

width: 9em;

display: block;

border: 0.1em solid #dcdce9;

color: #FFFFFF;

text-decoration: none;

text-align: center;

}

Figurile anterioare prezintă efectul succesiv cumulat al fiecărui nou element de stil aplicat (ultimul conducând chiar la meniul final).

În cadrul fișierului CSS se poate remarca referirea #nav-menu specificând *id*-ul elementului DIV (înglobând întreaga construcție a meniului). Cele trei stiluri #navmenu ul, #nav-menu li, #nav-menu li a, se referă la formatarea listelor (ul) având ca părinte DIV-ul (#nav-menu), a elementelor listelor (li), respectiv a conținutului ancorelor (a) , având ca părinte elemente ale listei (li), care la rândul

lor au ca părinte DIV-ul (#nav-menu).

Deși rulând codul anterior pe Mozilla, unele stiluri ar putea părea inutile, o simplă rulare pe Internet Explorer este relevantă pentru a dovedi necesitatea lor! Spre exemplificare, eliminarea stilului aferent elementelor din lista (#nav-menu li) conduce pe Internet Explorer la un meniu de genul celui din fig. următoare.



Un exemplu de meniu mai complex, având și sub-meniuri (*drop-down menu*), este prezentat în cele ce urmează. Întregul cod, atât lista HTML, cât și stilurile CSS (ca stiluri interne), sunt integrate într-un același fișier HTML. Efectul fiecărui nou element CSS succesiv adăugat este prezentat în figurile următoare. (fiecare figură fiind alăturată secvenței de cod CSS nou adăugată).

```
<ul id="menu">  
<li><a href="#">Unu</a></li>  
<li><a href="#">Doi</a>  
<ul>  
<li><a href="#">Doi-1</a></li>  
<li><a href="#">Doi-2</a></li>  
<li><a href="#">Doi-3</a></li>  
</ul>  
</li>
```

```
<li><a href="#">Trei</a>
```

```
<ul>
```

```
<li><a href="#">Trei-1</a></li>
```

```
<li><a href="#">Trei-2</a></li>
```

```
</ul>
```

```
</li>
```

```
</ul>
```

```
<style type="text/css">
```

```
ul
```

```
{
```

```
font-family: Arial, Verdana;
```

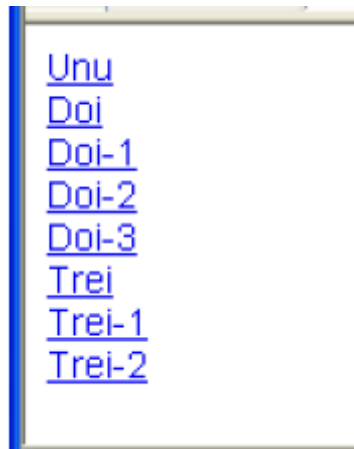
```
font-size: 14px;
```

```
margin: 0;
```

```
padding: 0;
```

```
list-style: none;
```

```
}
```



Stilizare liste

ul li

{

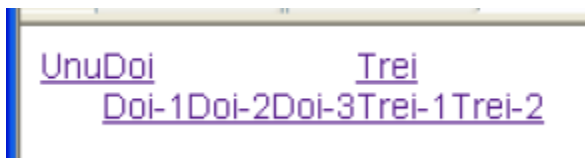
display: block;

position: relative;

float: left;

}

Stilizare elemente liste (având ca părinte liste)



li ul

{

display: none;

}

Stilizare liste (doar cele având ca părinte elemente) – în cazul de față ascunderea lor



ul li a

```
{  
display: block;  
text-decoration: none;  
color: #ffffff;  
border-top: 1px solid #ffffff;  
padding: 5px 15px 5px 15px;  
background: #1e7c9a;  
margin-left: 1px;  
white-space: nowrap;  
}
```

Stilizare ancore <a> având ca părinte elemente listă , care la rândul lor au ca părinte liste (practice toate ancorele meniului, inclusiv cele ascunse)



ul li a:hover

```
{  
background: #3b3b3b;  
}
```


Stilizare elemente (cele vizibile) având ca părinte liste - (setare culoare background/ fundal la

deplasare mouse deasupra ancorelor <a>)



// hover – element selector atașat unei etichete HTML, generând, în momentul în care mouse-ul se deplasează deasupra acesteia, o stilizare fie a acesteia, fie a altei etichete.

// ul li a:hover - când mouse-ul se deplasează deasupra unei ancore <a> (etichetă selectată), modifică fundalul ancorei având ca părinte un element , care la rândul face parte dintr-o listă .

li:hover ul

{

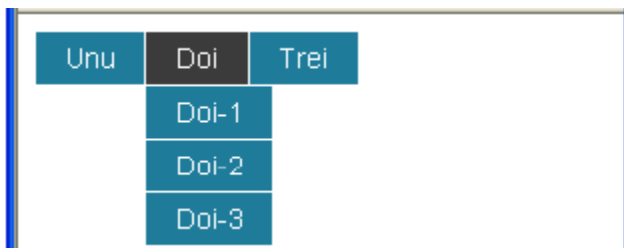
display: block;

position: absolute;

}

// li:hover ul - când mouse-ul se deplasează deasupra unui element , afișează elementul (având ca părinte acel)

Afișare submeniu la o deplasare mouse deasupra unui element având ca părinte o listă (practic anulează ascunderea)



```
li:hover li {  
float: none;  
font-size: 20px;  
}
```

// li:hover li - când mouse-ul se deplasează deasupra unui element ,

stilizează acel (având ca părinte alt)

//aici s-ar putea opri dezvoltarea meniului

La o deplasare deasupra unui element listă - setare scris mai mare

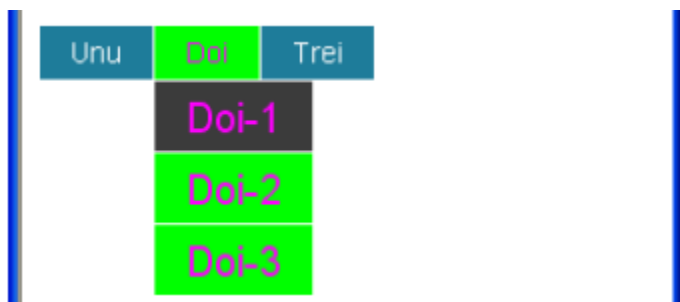


```
li:hover a {  
background: #00ff00;  
color: #ff00ff;  
}
```

//simple schimbări de culori (nu sunt necesare)

</style>

La o deplasare deasupra unui - setare culoare scris și fundal ancoră <a>



Observație

Din păcate, meniul anterior prezentat funcționează integral doar pe Mozilla, și doar parțial pe Internet Explorer (sub-meniurile nefiind active).

Rezolvarea constă în utilizarea de cod suplimentar JavaScript.

Primul exemplu făcea referire la o etichetă DIV, definind o diviziune (secțiune) a unui document HTML, des utilizată împreună cu elemente CSS. Oarecum echivalent cu DIV, eticheta SPAN permite și ea o încapsulare a unor elemente într-o secțiune a unei pagini HTML. Deosebirile majore între cele două

etichete sunt următoarele:

- DIV - permite formatarea întregii secțiuni delimitate (folosind stiluri CSS);

- SPAN - nu permite o formatare a secțiunii delimitate de el, ci doar a

elementelor (de regulă, a textului) incluse în el. În plus, un SPAN este precedat implicit de un paragraf nou, începând pe o linie nouă (mai puțin cazul în care este inclus într-un DIV).

Codul următor, este relevant pentru evidențierea caracteristicilor celor două etichete (DIV și SPAN):

```
<div id="mydiv" style="color: blue; background: yellow">Ceva text in Div 1
```

```
<span style="color: red; background: white"> Ceva text in Span 1 (integrat intr-un  
DIV -fara paragraf nou) </span>
```

Tot text in Div 1.

</div>

Span extern 2 (se observa ca incepe cu paragraf nou)

<h3>Inainte de Span 3 Continut Span 3 Dupa Span 3</h3>

Continut Span 4 - stilizare doar continut efectiv

Exemplificare DIV și SPAN

