

# TEHNICA CSS FOLOSITĂ ÎN REALIZAREA UNUI PORTAL EDUCAȚIONAL

Dora Coardos

Cornelia Ioana Lepădatu

*Institutul Național de Cercetare - Dezvoltare în Informatică - ICI, București*

*„Educația este întotdeauna un orizont, nu o destinație”*

*(Nicolae Iorga)*

**Rezumat:** Scopul principal al articolului este de a prezenta unele rezultate ale cercetării obținute în cadrul proiectului PN 03 13 07 04, care a avut ca obiectiv principal identificarea unor *noi metode și tehnici* pentru realizarea unui portal pentru instruirea copiilor preșcolari. În acest articol, se face o prezentare succintă despre *foile de stil în cascadă*, adesea cunoscute sub prescurtarea de CSS-uri. Intenția noastră este de a arăta ce sunt, la ce folosesc și cum se folosesc. De asemenea, vom arăta de ce este util să le folosim.

**Cuvinte cheie:** Cascading Style Sheets, Sistem Educațional, World Wide Web (W3C), Internet.

## 1. Introducere

Implicațiile educaționale ale utilizării Internetului sunt încă greu de imaginat întrucât organizarea cunoașterii și a accesului la aceasta devine în principiu, fără limite.

Volumul mare de informații pe care un individ trebuie să și le însușească în timpul școlarizării duce, pe de o parte, la necesitatea structurării mai bune a informației prezentate și, pe de altă parte, la necesitatea creșterii randamentului de învățare [1]. Dacă pentru rezolvarea primei părți a problemei responsabil este sistemul de învățământ, pentru partea a doua, trebuie acționat la nivelul individului prin formarea unor deprinderi încă de la vârsta preșcolară, folosind, în mod atractiv și plăcut, facilitățile oferite de sistemele educaționale existente.

Prezentul articol se referă la folosirea tehnicilor avansate pentru realizarea de software educațional, de instruire a preșcolarilor și de informare a părinților, cu ajutorul calculatorului, prin intermediul Internetului.

De o foarte mare importanță pentru copiii preșcolari sunt primele noțiuni pe care aceștia și le însușesc și, mai ales, calitatea acestora. Astăzi, un nou tip de învățare își face loc în viața noastră. Procesul de învățare devine mai plăcut și mai interesant prin intermediul calculatorului. Lecțiile vor fi însoțite de texte sugestive, care pot fi cu ușurință asimilate de cei mici, și de imagini viu colorate și expresive, care permit dezvoltarea limbajului și a vocabularului.

Aceste lecții se referă la noțiunile de bază, care orice copil trebuie să le cunoască. Pe măsură ce este parcurs, bagajul de cunoștințe generale crește pornind de la noțiuni simple, cum ar fi culorile, și ajungând până la cunoașterea de poezioare, cântecele precum și a unor proverbe și zicători. Noțiunile elementare, precum animale domestice și sălbatice, anotimpuri, litere și cifre, familie și multe altele încep să aibă înțeles de la vârste fragede, dându-le astfel posibilitatea celor mici nu numai să le învețe, ci să se familiarizeze cu calculatorul și, în mod special, cu lucrul cu mouse-ul. Programele sunt astfel structurate încât copilul poate alege orice lecție din cele prezentate cu ajutorul mouse-ului sau poate repeta anumite părți dintr-o lecție până când ajunge să cunoască și să înțeleagă toate noțiunile cuprinse în lecția respectivă.

Legătura dintre joc și învățare este de mult recunoscută și subliniată de pedagogi și psihologi [2]. A învăța jucându-te este idealul copiilor și al pedagogilor deopotrivă. Scenariile pentru aceste lecții sunt create în conformitate cu programa școlară, elaborată și aprobată de Ministerul Educației și Cercetării. Portalul pe care-l vom realiza va conține lecții de aritmetică și lecții de abecedar, create în așa fel încât copilul să se distrează învățând. Programele extrem de simple nu necesită cunoștințe de calculator, dar fac copilul să-și dorească să-l cunoască, iar includerea acestora în procesul de instruire face din învățământul românesc un învățământ competitiv pe plan internațional.

Instruirea asistată de calculator este cel mai obișnuit termen folosit în descrierea utilizării calculatoarelor în scopuri educaționale. În multe țări, utilizarea calculatoarelor și a tehnologiilor informatice a intrat în practica școlară, deschizând noi orizonturi procesului de instruire. Lecțiile, extrem de prietenoase, pot fi folosite de orice copil care știe să folosească mouse-ul și urmărește indicațiile de pe ecran.

Ținând cont de tendința actuală a sistemului educațional național, promovat de Ministerul Educației și Cercetării, de a micșora vârsta la care se preconizează a se începe ciclul primar, se remarcă și necesitatea

abordării într-o nouă manieră a pregătirii preșcolărilor, a conținutului programelor didactice și a metodelor de predare, cât și a scopului urmărit de fiecare lecție în parte.

Pentru realizarea acestui portal educațional pentru părinții cu copii preșcolari, s-au folosit facilitățile oferite de CSS-uri (Cascading Style Sheets) pentru descrierea fiecărui element HTML. Această tehnică a CSS-urilor permite dezvoltatorilor de pagini Web, cât și utilizatorilor, un control mai mare asupra modului de afișare a paginilor Web.

## 2. Foi de stil în cascadă - CSS (Cascading Style Sheets)

Cu toate că cele două browsere principale – Netscape și Internet Explorer – continuă să adauge noi etichete și atribute (cum sunt etichete pentru <font> și atribute de culoare) la specificațiile originale HTML, este din ce în ce mai greu a crea pagini Web în care conținutul documentului HTML să fie în mod clar separat de forma de prezentare a documentului pe Web.

Pentru a rezolva această problemă, Consorțiul World Wide Web (W3C), responsabil pentru standardizare HTML, creează „*Stiluri*” (Modele) în plus pentru HTML 4.0. Ambele browsere, Netscape 4.0 și Internet Explorer 4.0, recunosc aceste „stiluri”.

Stilurile pun la dispoziția creatorilor de site-uri web noi posibilități de personalizare a paginilor HTML.

Practic, tehnologia CSS facilitează oricărui proiectant de pagini web posibilitatea de a crea un stil pentru fiecare tag HTML întrebuițat la un moment dat [3], putând ulterior să aplice implicit acest stil în toate paginile site-ului pentru elementul de pagină respectiv (adică pentru tagul HTML stilizat).

Standardul CSS permite ca informațiile cu privire la stil să fie specificate și referite în mai multe moduri.

Stilurile pot fi specificate în interiorul fiecărui tag HTML în parte [4], în cadrul secțiunii HEAD a documentelor HTML sau într-un fișier extern cu extensia .css. În interiorul aceluiași document HTML, pot fi referite mai multe fișiere de tip CSS externe.

Aplicarea stilurilor reprezintă o extindere importantă a posibilităților de design, evitând utilizarea de fișiere grafice mari, ce determină încetinirea încărcării paginilor și manipularea lor greoaie.

Folosind stilurile HTML puteți fi sigur că cititorii portalului vor vedea textul din pagină exact așa cum a fost el proiectat. Textul și proprietățile acestuia, care au fost inițial controlate de browsere, se reîntorc la autor, acolo unde le este, de fapt, locul.

### 2.1. Ce sunt foile de stil în cascadă (CSS-urile)?

CSS reprezintă abrevierea pentru “Cascading Style Sheets” (foi de stil în cascadă).

*Un stil* reprezintă o colecție de atribute ale textului și ale modului de aranjare a documentului, care pot fi aplicate în mod selectiv unui document sau doar unei părți din acesta. Aceste atribute pot fi tipul de font, mărimea și grosimea acestuia, marginile, paragrafele și orice element ce poate influența aspectul textului în pagină. Gruparea lor în stiluri permite autorului să aplice aceeași colecție de atribute la diferite părți ale unui document.

Ca și o pagină web, o *foaie de stil în cascadă* constă din instrucțiuni de tip text ASCII care indică browser-ului cum să formateze documentul care se încarcă. Instrucțiunile CSS pot fi plasate chiar în cadrul respectivei pagini web sau se pot afla într-un document separat, care este legat de pagina HTML.

*Foile de stil* permit adăugarea de facilități noi la un fișier de tip HTML, oferind astfel proiectanților și dezvoltatorilor de pagini Web un control mai mare asupra modului de afișare a paginilor Web. Prin intermediul foilor de stil, aceștia pot crea diverse stiluri pentru afișarea unor elemente comune paginilor unui Web site (ex: antete și legături).

Termenul *cascading* derivă din faptul că la aceeași pagină Web pot fi aplicate mai multe stiluri.

CSS-urile reprezintă un mecanism simplu, pentru a controla stilurile într-un document Web, fără a-i compromite structura. Prin separarea elementelor vizuale de proiectare (fonturi, culori, margini etc.) de structura logică a unei pagini Web, CSS dă proiectanților Web controlul mult dorit fără să sacrifice integritatea datelor, astfel asigurându-le posibilitatea de utilizare în mai multe situații.

Stilurile în HTML 4.0 definesc modul cum sunt afișate elemente HTML, la fel ca etichetele de fonturi și atribute de culoare în HTML 3.2. Stilurile sunt, în mod normal, salvate în fișiere externe documentelor

HTML. Foile cu stiluri externe permit schimbarea aspectului în toate paginile din site, prin editarea unui singur document cu extensia .css. Dacă se dorește schimbarea fonturilor sau culorii tuturor titlurilor în toate paginile site-ului, se poate apela un CSS construit pentru această acțiune.

CSS-urile deschid un drum nou în proiectarea paginilor Web deoarece ele permit dezvoltatorilor să controleze stilul și modul de prezentare al informațiilor în mai multe pagini Web odată. Astfel, un dezvoltator de pagini Web poate defini un stil pentru fiecare element HTML și îl poate aplica la mai multe pagini Web. Dacă se dorește o schimbare globală, pur și simplu se schimbă stilul și toate elementele din paginile Web se actualizează automat.

Astfel, putem afirma că:

- CSS-urile reprezintă *foi de stiluri* în cascadă;
- stilurile definesc cum să apară elementele HTML;
- stilurile sunt în mod normal stocate în fișiere de stiluri;
- stilurile se vor adăuga în fișiere de tip HTML pentru a rezolva o problemă;
- fișierele de stiluri externe pot fi salvate în mai multe lucrări;
- fișierele de stiluri externe sunt fișiere cu extensia .css;
- definițiile multiple de stiluri vor fi realizate într-un singur fișier.

## 2.2 Utilizarea foilor de stiluri în cascadă

CSS-urile trec peste setările implicite ale browser-ului pentru interpretarea etichetelor ce vor fi afișate, permițând utilizarea oricărui element HTML indicat printr-o etichetă de deschidere și una de închidere, la care se aplică stiluri de atribute definite fie local, fie într-un fișier de stiluri.

Foile de stiluri permit ca anunțarea stilului să se facă în mai multe moduri. Stilul poate fi specificat:

- în interiorul unui singur element HTML;
- în interiorul head-erului unei pagini HTML;
- într-un fișier extern cu extensia .css.

Foile de stiluri în cascadă permit compunerea unor stiluri separate, cum ar fi proprietățile de culoare și font, seturile de caractere, grafica și prezentarea. Mai mult, controlul formătărilor este divizat între autor și cititor, acesta din urmă putându-și interpune propria sa înălțare de foi de stiluri, deci, contribuind la prezentarea datelor, în funcție de dorințele sale individuale.

Utilizând ordinea în cascadă se pune întrebarea „care stil va fi utilizat când există mai mult decât un stil specificat pentru un element HTML?”

Într-o manieră generală, se poate spune că toate stilurile din cascadă vor intra într-un nou stil „virtual” prin folosirea unor reguli, dintre care numai patru reguli au cea mai mare importanță în stabilirea priorității și anume:

- browser implicit;
- stiluri externe;
- stiluri interne (în interiorul etichetei <head>);
- stil integrat (în interiorul unui element HTML).

Astfel, un stil integrat (în interiorul unui element HTML) are cea mai mare prioritate, ceea ce semnifică că el va depăși toate stilurile declarate în interiorul etichetelor <head>, într-un stil extern și într-un browser (o valoare implicită).

## 2.3. Sintaxa CSS-urilor

Sintaxa unui CSS se compune din trei părți: selector, proprietate, valoare [9]. Astfel, forma generală a unui CSS este:

**selector {proprietate: valoare}**

unde:

- **selector** - este în mod normal un element/tag HTML care se dorește a fi definit;
- **proprietatea** - este un atribut care se dorește a fi schimbat;
- **valoare** - luată de fiecare proprietate.

Proprietatea și valoarea sunt separate prin “:” și sunt incluse între acolade “{ }”. De exemplu:

```
body {color: black}
```

Dacă valoarea este formată din mai multe cuvinte, se pune între ghilimele. De exemplu:

```
P {font-family: "sans serif"}
```

**Notă:** Dacă se dorește specificarea mai multor proprietăți, acestea se vor separa prin “;”. În exemplul următor, se arată cum se definește un paragraf aliniat pe centru care are un text de culoare roșie:

```
P {text-align:center;color:red}
```

Pentru a realiza definirea unui stil cât mai lizibil, se pot scrie proprietățile câte una pe o linie, așa cum se poate vedea, de altfel, și în exemplul de mai jos:

```
P  
{  
text-align: center;  
color: black;  
font-family: arial  
}
```

### Gruparea selectorilor

Selectorii se pot grupa. Separarea lor se face prin virgulă “,”. În exemplu de mai jos se poate vedea cum sunt grupate toate elementele de antet. Fiecare element de antet va fi de culoare verde.

```
h1,h2,h3,h4,h5,h6 {color: green}
```

### Clasa Selector

Prin intermediul clasei selector se pot defini diferite stiluri pentru atributele aceluiași tip de element HTML. Să presupunem că dorim să avem două tipuri de paragrafe în documentul nostru: un paragraf aliniat la dreapta și unul aliniat pe centru. În acest caz, se definesc următoarele clase de stiluri pentru elementul „paragraf”:

```
p.right {text-align: right}  
p.center {text-align: center}
```

Pentru a utiliza aceste clase în documentul HTML, se procedează astfel:

```
<p class="right"> Acesta este un paragraf aliniat la dreapta</p>  
<p class="center"> Acesta este un paragraf aliniat pe centru</p>
```

**Notă:** În cadrul unui element HTML, poate fi specificată numai o singură clasă de atribute. Astfel, exemplul următor este greșit deoarece, în cadrul elementului <p>, sunt specificate două clase de atribute pentru acest element:

```
<p class="right" class="center"> Acesta este un paragraf</p>
```

Dacă se dorește definirea unei clase de atribute care să fie utilizată pentru toate elementele HTML care au atributul respectiv, se poate omite numele etichetei în selectorul ce va defini clasa respectivă. În exemplul următor, se definește clasa class="center" prin intermediul căreia toate elementele HTML ce au atributul "align" vor fi aliniate centrat.

```
.center {text-align: center}
```

Într-un document HTML, această clasă se poate utiliza astfel:

```
<h1 class="center"> Acest antet va fi aliniat pe centru</h1>
<p class="center"> Acest paragraf va fi aliniat pe centru</p>
```

## ID Selector

Identificatorul selectorului (ID) este diferit de clasa Selector. În timp ce clasa Selector se poate aplica la mai multe elemente dintr-o pagină, un identificator de selector se poate aplica numai unui element. Un identificator de selector trebuie să fie unic într-un document.

Definițiile de stil de mai jos asociază elementului "paragraf" identificatorul cu valoarea "para 1" sau "para 2":

```
p#para1 {text-align: center; color: red;}
p#para2 {text-align: right; color: magenta;}
```

Următoarea definiție de stil va putea să fie asociată oricărui element HTML, care are atributul "ID".

```
*#wer345 {color: green;}
```

Stilurile definite mai sus pot fi apelate în cadrul unor elemente HTML prin intermediul atributului "ID":

```
<h1 id="wer345">IN INTERNET EXPLORER</h1>
<p id="para1">Identificator ID</p>
<h3 id="wer345">Stil "wer345" in H3</h3>
<p id="wer345">Stil "wer345" in P</p>
<p id="para2">Stil "para2" in P</p>
<h2 id="para2">Stil "para2" in H2</h2>
```

## Comentarii CSS

Se pot introduce comentarii în CSS pentru a explica codul [7]. Comentariul este ignorat de browser. Un comentariu în CSS începe cu "/\*" și se termină cu "\*/". De exemplu:

```
/* Acesta este un comentariu */
p
{
text-align: center;
/* Acesta este un alt comentariu */
color: black;
font-family: arial
}
```

## 2.4. Inserare CSS-uri

Pentru formatarea unui document în conformitate cu diverse foi de stil (CSS), există în HTML câteva tag-uri specifice, care permit inserarea CSS-urilor: <style>, <link>, <div>, <span>[6].

<b>&lt;style&gt;</b>	Definește un stil într-un document. <ul style="list-style-type: none"><li>• Tag-ul style este plasat în secțiunea head</li><li>• Dacă se dorește includerea unei foi de stil într-o pagină, trebuie să se definească extern foaia de stil și să se facă link la ea prin intermediul tag-ului</li></ul>
<b>Sintaxă</b>	<b>&lt;style type="text/css"&gt; ... &lt;/style&gt;</b>
<b>&lt;link&gt;</b>	Definește relația dintre două documente. <ul style="list-style-type: none"><li>• Tag-ul link este un element vid, conținând numai atribute</li></ul>

	<ul style="list-style-type: none"> <li>• Tag-ul link este plasat în secțiunea head și poate apărea de mai multe ori</li> </ul>
Sintaxă	<code>&lt;link rel="stylesheet" type="text/css" href="theme.css" /&gt;</code>
<code>&lt;div&gt;</code>	Definește o secțiune într-un document, permițând formatarea acesteia printr-o foaie de stil (CSS). <ul style="list-style-type: none"> <li>• Browser-ele inserează o linie liberă înainte și după secțiune</li> </ul>
Sintaxă	<code>&lt;div&gt; ... &lt;/div&gt;</code>
Exemplu	<p>Acesta este un text</p> <pre>&lt;div style="color:#FF0000; background: 'yellow';"&gt; &lt;h4&gt;Acesta este un header într-o secțiune div&lt;/h4&gt; &lt;p&gt; Acesta este un paragraf într-o secțiune div&lt;/p&gt; &lt;/div&gt;</pre>
<code>&lt;span&gt;</code>	Permite gruparea textului în interiorul unui element în vederea formatării acestuia printr-o foaie de stil (CSS).
Sintaxă	<code>&lt;span&gt; ... &lt;/span&gt;</code>
Exemplu	<pre>&lt;p&gt;Cascading Style Sheets (CSS) &lt;span style="color:#0000FF;"&gt; - Foi de Stil - &lt;/span&gt;</pre> <p>permit adăugarea de facilități noi limbajului HTML, oferind astfel proiectanților și dezvoltatorilor de pagini Web un control mai mare asupra modului de afișare al paginilor Web.</p> <pre>&lt;p&gt; &lt;p&gt; &lt;span style="color:#00DD45;"&gt;</pre> <p>Prin intermediul foilor de stil (CSS) aceștia pot crea diverse stiluri pentru afișarea unor elemente comune paginilor dintr-un Web site (ex: antete și legături).</p> <pre>&lt;/span&gt; &lt;/p&gt;</pre>

Când un browser citește un stil, el va formata documentul în conformitate cu acesta. Există trei moduri de inserare a unui stil corespunzătoare tipurilor de stiluri definite:

### Stiluri externe

Foaia de stiluri externă este ideală când stilul este aplicat la mai multe pagini. Cu o foaie de stil externă se poate schimba aspectul întregului site prin modificarea unui singur fișier. Fiecare pagină poate să aibă legătură la foaia de stiluri folosind eticheta `<link>`. Eticheta `<link>` trebuie plasată în interiorul secțiunii principale (head).

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

Browser-ul va citi definițiile stilului din fișierul defstil.css și va formata documentul în conformitate cu el.

O foaie cu stiluri externă poate fi scrisă cu orice editor de text. Fișierul nu trebuie să conțină etichete HTML. Foaia de stiluri va fi salvată cu extensia .css. Un exemplu de fișier pentru o foaie de stiluri este prezentat în continuare:

```
hr {color: sienna}
p {margin-left: 20px}
body {background-image: url("images/back40.gif")}
```

**Observație.** Nu se lasă spațiu între valoarea proprietății și unitatea de măsură! Dacă se pune `margin-left: px 20` în loc de `margin-left: 20px` aceasta va funcționa numai în IE6, dar nu va funcționa în Mozilla sau Netscape.

### Stiluri interne

O foaie de stiluri internă trebuie folosită când un singur document are un stil propriu. Se definește stilul intern în secțiunea principală încadrat între etichetele `<style>` după cum se vede și în exemplul de mai jos:

```
<head>
  <style type="text/css">
    hr {color: sienna}
    p {margin-left: 20px}
    body {background-image: url("images/back40.gif")}
  </style>
</head>
```

Browser-ul va citi definițiile stilului și va formata documentul în conformitate cu acesta.

**Notă:** Un browser ignoră în mod normal etichetele necunoscute. Aceasta înseamnă că vechile browsere a căror acțiune nu suportă stilurile, vor ignora etichetele `<style>`, dar conținutul din cadrul etichetelor `<style>` va fi afișat pe pagină. Se poate preveni acest inconvenient pentru un browser vechi prin includerea definiției de stil în elemente de comentariu HTML.

```
<head>
<style type="text/css">
  <!--
    hr {color: sienna}
    p {margin-left: 20px}
    body {background-image: url("images/back40.gif")}
  -->
</style>
</head>
```

### Stiluri integrate

Un stil integrat pierde multe din avantajele foilor de stiluri prin îmbinarea conținutului cu prezentarea. Se recomandă utilizarea acestui mod numai când un stil se aplică o singură dată pentru un element. Pentru a utiliza stilurile integrate, se vor pune componentele stilului ca valori ale atributului "style". Componentele stilului pot conține orice proprietăți CSS. Exemplul de mai jos arată cum se poate schimba culoarea textului și marginea din stânga în cadrul unui paragraf.

```
<p style="color: sienna; margin-left: 20px">Acesta este un paragraf</p>
```

Dacă în același selector au fost grupate câteva proprietăți, în diferite foi de stiluri, valorile vor fi moștenite din stilul specificat cel mai curând. De exemplu, un stil extern are proprietățile următoare pentru selectorul h3:

```
h3 {color: red;text-align: left;font-size: 8pt}
```

Și un stil intern are proprietățile următoare pentru selectorul h3:

```
h3 {text-align: right; font-size: 20pt}
```

Dacă o pagină HTML conține cele două foi de stiluri, internă și externă, atunci proprietățile pentru selectorul h3 vor fi:

```
color: red; text-align: right; font-size: 20pt
```

Culoarea este moștenită din stilul extern, iar alinierea textului și dimensiunea textului din stilul intern.

## 2.5. Proprietăți definite prin CSS-uri

*Foile de stil* permit definirea unor proprietăți pentru anumite tipuri de elemente HTML. Elementele de tip HTML, care pot fi definite cu ajutorul CSS-urilor sunt: *background-ul*, *textul*, *fonturile*, *border-ul*, *marginile*, *padding-ul*, *listele* [5].

Fiecare proprietate are o acțiune, o valoare și poate fi recunoscută de diferite versiuni ale browser-elor.

*Proprietățile background-ului*, care se pot defini prin intermediul unei foi de stil, permit controlul culorii background-ului unui element, setarea unei imagini ca background, repetarea unei imagini a background-ului vertical sau orizontal și poziționarea unei imagini pe o pagină.

Aceste proprietăți sunt: *background*, *background-attachment*, *background-color*, *background-image*, *background-repeat*.

*Proprietățile CSS pentru text* permit controlul asupra aspectului textului. Există posibilitatea de a schimba culoarea textului, de a mări sau micșora spațiul dintre caracterele dintr-un text, alinierea textului, decorarea textului, alinierea primei linii a paragrafului și multe altele. Aceste proprietăți sunt: *color*, *direction*, *letter-spacing*, *text-align*, *text-decoration*, *text-indent*, *text-shadow*, *text-transform*, *unicode-bidi*, *white-space*, *word-spacing*.

*Proprietățile CSS pentru fonturi* permit schimbarea familiei fonturilor, îngroșarea, mărirea și stilul pentru un text. În CSS1, fonturile pot fi identificate prin numele lor. Dacă un browser nu recunoaște această specificare pentru fonturi, atunci va utiliza fontul implicit. Aceste proprietăți sunt: *font*, *font-family*, *font-size*, *font-size-adjust*, *font-stretch*, *font-style*, *font-variant*, *font-weight*.

*Proprietățile CSS pentru border* permit definirea stilului, culorii și dimensiunii acestora. Aceste proprietăți sunt: *border*, *border-bottom*, *border-bottom-color*, *border-bottom-style*, *border-bottom-width*, *border-color*, *border-left*, *border-left-color*, *border-left-style*, *border-left-width*, *border-right*, *border-right-color*, *border-right-style*, *border-right-width*, *border-style*, *border-top*, *border-top-color*, *border-top-style*, *border-top-width*, *border-width*.

*Proprietățile CSS pentru margini* definesc spațiile din jurul elementelor. Este posibil să fie folosite valori negative pentru a suprapune marginea peste conținut. Marginile de sus, din dreapta, de jos și din stânga, pot fi schimbate independent folosind proprietăți separate. De asemenea, poate fi utilizată o proprietate în forma scurtă pentru a schimba toate marginile o dată. Aceste proprietăți sunt: *margin*, *margin-bottom*, *margin-left*, *margin-right*, *margin-top*.

*Proprietățile CSS pentru padding* definesc spațiul dintre marginea elementului și conținutul acestuia. O valoare negativă nu este permisă. Spațiul de umplere de sus, din dreapta, de jos și din stânga poate fi schimbat independent, folosind proprietăți diferite. Există o proprietate de umplere în formă scurtă pentru a controla mai multe margini o dată. Aceste proprietăți sunt: *padding*, *padding-bottom*, *padding-left*, *padding-right*, *padding-top*.

*Proprietățile CSS pentru liste* permit schimbarea între diferiți marcatori de liste, setarea unei imagini ca marcator de listă și setarea locului unde să fie plasat marcatorul. Aceste proprietăți sunt: *list-style*, *list-style-image*, *list-style-position*, *list-style-type*.

## 2.6. Exemple de utilizare a foilor de stil în pagini HTML

Pentru a arăta modul de lucru cu foile de stil în cascadă (CSS-uri), vom da câteva exemple care sunt mai reprezentative.

Exemplul 1: Constă dintr-un fișier HTML care utilizează eticheta <link> pentru a face legătura la o foaie de stiluri externă.

```
<html>
<head>
  <link rel="stylesheet" type="text/css" href="ex1.css" />
</head>
<body>
  <h1>Acesta este un antet de 36 pt</h1>
  <h2>Acest antet este albastru</h2>
```



```
<p>Acesta este un paragraf cu marginea stângă la 50 pixels</p>
</body>
</html>
```

Fișierul de stiluri apelat mai sus și având denumirea ex1.css este de forma:

```
body {background-color: yellow}
h1 {font-size: 36pt}
h2 {color: blue}
p {margin-left: 50px}
```

Rezultatul aplicării acestui stil în fișierul HTML este un frame care conține:

**Acesta este un antet de 36 pt**  
**Acest antet este albastru**  
Acesta este un paragraf cu marginea stângă la 50 pixeli

**Exemplul 2.** Constă dintr-un fișier HTML, care utilizează eticheta <link> pentru a face legătura la o foaie de stiluri externă.

```
<html>
<head>
  <link rel="stylesheet" type="text/css" href="ex2.css" />
</head>
<body>
  <h1>Acesta este un antet de nivel 1</h1>
  <hr>
  <p>Puteți vedea cum stilurile formatează texte</p>
  <p><a href="http://www.microsoft.com"
    target="_blank">Acesta este un link</a></p>
</body>
</html>
```

Fișierul de stiluri apelat mai sus și având denumirea ex2.css este de forma:

```
body {background-color: tan}
h1 {color:maroon; font-size:20pt}
hr {color:navy}
p {font-size:11pt; margin-left: 15px}
a:link {color:green}
a:visited {color:yellow}
a:hover {color:black}
a:active {color:blue}
```

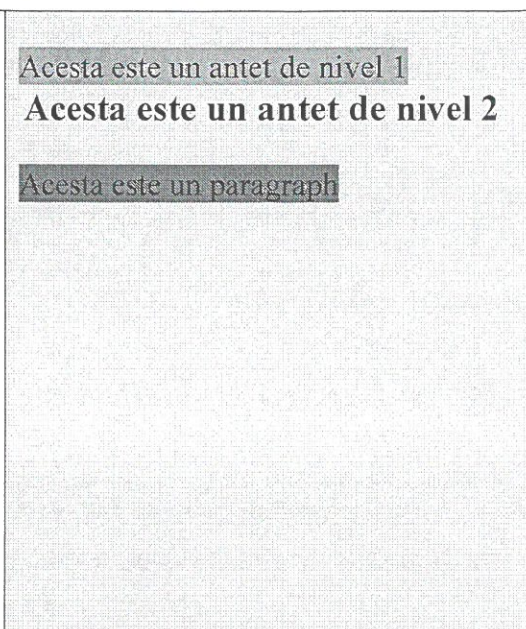
Rezultatul aplicării acestui stil în fișierul HTML este un frame care conține:

Acesta este un antet de nivel 1  

---

**Puteți vedea cum stilurile formatează texte**  
**Acesta este un link**

**Exemplul 3.** Cum se poate seta culoarea unui background și culoarea diferitelor elemente cum ar fi antetele din pagină. Stilul este definit în interiorul fișierului HTML.

<pre>&lt;html&gt; &lt;head&gt; &lt;style type="text/css"&gt;   body {background-color: yellow}   h1 {background-color: #00ff00}   h2 {background-color: transparent}   p {background-color: rgb(250,0,255)} &lt;/style&gt; &lt;/head&gt;  &lt;body&gt; &lt;h1&gt;Acesta este un antet de nivel 1&lt;/h1&gt; &lt;h2&gt;Acesta este un antet de nivel 2&lt;/h2&gt; &lt;p&gt;Acesta este un paragraph&lt;/p&gt; &lt;/body&gt; &lt;/html&gt;</pre>	
--	--

## 2.7. Pseudo - clase și clase în CSS-uri

Pseudo-clasele se utilizează în foile de stil (CSS) pentru a adăuga anumite efecte unor selectoare sau unei părți a acestora [8]. Sintaxa unei pseudo-clase este:

*selector:pseudo-class {proprietate: valoare}*

Clasele în CSS-uri pot fi, de asemenea, utilizate cu pseudo-clase:

*selector.class:pseudo-class { proprietate: valoare}*

### 2.7.1. Pseudo-clasa anchor (a)

Un link care poate fi activ, vizitat, nevizitat, sau în momentul în care se trece cu mouse-ul peste el poate fi afișat în diferite moduri în browsere care suportă CSS-uri:

*a:link {color: #FF0000} /\* link nevizitat\*/*

*a:visited {color: #00FF00} /\* link vizitat \*/*

*a:hover {color: #FF00FF} /\* mouse over link \*/*

*a:active {color: #0000FF} /\* link activ \*/*

#### Notă:

- *a:hover* trebuie să fie după *a:link* și *a:visited* într-o definiție CSS pentru a fi efectiv;
- *a:active* trebuie să fie după *a:hover* într-o definiție CSS pentru a fi efectiv;
- numele pseudo-claselor nu sunt case-sensitive;
- versiunile IE 4 și următoarele suportă anchor pseudo-class. NN 4.5 și Netscape 6 suportă numai parțial pseudo-clasa anchor.

Pseudo-clasele pot fi combinate cu clase CSS. De exemplu:

*a.red:visited {color: #FF0000}*

*<a class="red" href="css\_syntax.asp">CSS Syntax</a>*

Dacă link-ul din exemplul de mai sus a fost vizitat, el va fi afișat cu roșu.

## 2.8. Pseudo-elemente în CSS-uri

Pseudo-elementele se utilizează în foile de stil (CSS) pentru a adăuga anumite efecte unor selectoare sau unei părți a acestora. Sintaxa unui pseudo-element este:

```
selector:pseudo-element {proprietate: valoare}
```

Clasele în CSS pot fi, de asemenea, utilizate cu pseudo-elemente:

```
selector.class:pseudo-element {proprietate: valoare}
```

### 2.8.1. Pseudo-elementul :first-line

Pseudo-elementul :first-line se utilizează pentru a adăuga un stil special primei linii de text într-un element. De exemplu:

```
p {font-size: 12pt}
```

```
p:first-line {color: #0000FF; font-variant: small-caps}
```

```
<p>Un text care se continua pe una sau mai multe linii</p>
```

În exemplul de mai sus, browser-ul afișează prima linie a textului în concordanță cu pseudo-elementul :first-line. Unde se termină prima linie a textului, depinde de dimensiunea ferestrei browser-ului în Windows.

**Notă:** Pseudo-elementul :first-line se poate utiliza numai cu elemente de tipul block-level.

Pseudo-elementului :first-line i se pot aplica următoarele proprietăți: *font, color, background, word-spacing, letter-spacing, text-decoration, vertical-align, text-transform, line-height, clear*.

### 2.8.2. Pseudo-elementul :first-letter

Pseudo-elementul :first-letter se utilizează pentru a adăuga un stil special primei litere de text într-un element.

De exemplu:

```
p {font-size: 12pt}
```

```
p:first-letter {font-size: 200%; float: left}
```

```
<p>Prima litera dintr-un cuvint</p>
```

**Notă:** Pseudo-elementul :first-letter se poate utiliza numai cu elemente de tipul block-level.

Pseudo-elementului :first-letter i se pot aplica următoarele proprietăți: *font, color, background, margin, padding, border, text-decoration, vertical-align (numai dacă 'float' este 'none'), text-transform, line-height, float, clear*.

### 2.8.3. Pseudo-elemente și clase CSS

Pseudo-elementele pot fi combinate în clase CSS așa cum se poate observa în următorul exemplu:

```
p.article:first-letter {font-size: 200%; color: #FF0000}
```

```
<p class="article">Primul cuvint dintr-un articol.</p>
```

Prin exemplul de mai sus, se realizează marcarea primei litere din textul cuprins în paragraf cu ajutorul clasei "article".

### 2.8.4. Pseudo-elemente multiple

Se pot combina mai multe pseudo-elemente, așa cum se observă și în exemplul de mai jos:

```
p {font-size: 12pt}
```

```
p:first-letter {color: #FF0000; font-size: 200%}
```

```
p:first-line {color: #0000FF}
```

```
<p>Primul cuvint intr-un articol.</p>
```

Prin exemplul de mai sus, se realizează afișarea primei litere din textul unui paragraf cu roșu și dimensiunea fontului 24pt. Restul primei linii va fi de culoare albastră, iar următoarele linii vor avea culoarea implicită.

### 3. Concluzii. Avantajele folosirii „Foilor de stil”

Aplicarea foilor de stil unui document HTML de către creatorii de pagini Web are numeroase avantaje esențiale pentru creatorii de pagini Web:

- folosirea lor reduce considerabil efortul depus atunci când se dorește modificarea aspectului și aranjării elementelor din paginile dumneavoastră; în loc să parcurgeți fiecare document în parte și să faceți modificări asupra fiecărui element, nu mai este necesar să operați modificări decât asupra foi de stiluri, care controlează aceste elemente; modificările se vor reflecta asupra oricărui segment de text care a fost formatat cu stilul respectiv;
- oferă control crescut asupra aspectului și plasării textului în pagină;
- reduc „încurcătura” produsă de multitudinea de deschideri și închideri ale etichetelor care descriu elementele individuale ale textului;
- procesul de modificare a diferitelor elemente din pagină se simplifică întrucât modificarea ulterioară a unui stil se reflectă automat în toate zonele în care a fost folosit;
- pot fi create și aplicate nu doar pentru controlul aspectului fonturilor, ci și a multor altor elemente de formatare ale paginii.

### Bibliografie

1. ENGELMANN, S., T. ENGELMANN: Dați copilului vostru o inteligență superioară, Centrul pentru educație și Dezvoltarea Creativității, 1992.
2. \* \* \*: Technology in Early Childhood Programs, National Association for the Education of Young Children,
3. <http://www.kidsource.com/kidsource/content3/technology.early.p.k12.2.html>.
4. <http://hotwired.lycos.com>
5. <http://www.w3schools.com/html/default.asp>
6. [http://www.w3schools.com/css/css\\_reference.asp](http://www.w3schools.com/css/css_reference.asp)
7. <http://webdesign.about.com/cs/htmltags/a/aa011000a.htm>
8. <http://www.w3.org/TR/REC-CSS2/>
9. <http://www.meyerweb.com/eric/books/css-progref/>
10. <http://www.htmlhelp.com/reference/css/>