

Tehnici Web CURSUL 2

Semestrul I, 2017-2018
Carmen Chirita

<https://sites.google.com/site/fmitehniciweb/>

Elementele semantice **pre** si **code**

<pre> text preformatat </pre>

Este un element de tip block;
Păstrează spațiile și trecerea
la linie nouă

<code> cod într-un limbaj de programare </code>

Este un element inline;
Nu păstrează spațiile și trecerea la linie nouă
(se poate folosi împreună cu elementul **pre**)

```
<code>
#include "stdio.h"

void main()
{
    printf("\nHello World\n"); // print to screen
}
```

```
</code>
<hr>
<pre>
<code>
#include "stdio.h"
```

```
void main()
{
    printf("\nHello World\n"); // print to screen
}
```

```
</code>
</pre>
```

<hr>

```
<div>
  o  o
  )-(
  (O O)
  \=/
  .-.
  /\ /\
  // /\ \
  _/ {,-} \._
  || || || ||
  \_|| ||\_
  `---" "---'
</div>
```

<hr>

```
<pre>
  o  o
  )-(
  (O O)
  \=/
  .-.
  /\ /\
  // /\ \
  _/ {,-} \._
  || || || ||
  \_|| ||\_
  `---" "---'
</pre>
```

</pre>

```
#include "stdio.h" void main() { printf("\nHello World\n"); // print to screen }
```

```
#include "stdio.h"
```

```
void main()
{
    printf("\nHello World\n"); // print to screen
}
```

```
o o )-( (O O) \=/ .-. /\ /\ _// /\ \ _ =./ {,-} \._ = || || || || _|| ||_ `---" "---'
```

```

  o  o
  )-(
  (O O)
  \=/
  .-.
  /\ /\
  // /\ \
  _/ {,-} \._
  || || || ||
  \_|| ||\_
  `---" "---'
```

METADATA

<head>

<title>Titlu </title>

<meta charset="utf-8">

<meta name="autor" content="nume-autor">

<meta name="description" content="descriere">

<meta name="keywords" content="cuvinte cheie">

<base href="base URL for relative paths" >

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

<script type ="text/javascript"> ...</script>

<script type ="text/javascript" src="fis.js"></script>

<link href="fis.tip" type="image/tip" rel="shortcut icon">

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

<head>

Elementul **base**

Defineste un URL de baza pentru toate URL-urile relative din document;

Poate exista un singur element **base** în document

```
<head>
<title>Pagina mea</title>
<base href="/home/carmen/TEHNICI_WEB_CURSURI/EXEMPLE/"
target="_blank">
</head>
<body>
<a href="exemplul1.html" >Exemplul 1</a>
<a href="exemplul2.html" >Exemplul 2</a>

</body>
```

CSS = CASCADING STYLE SHEETS

- Folosit la definirea aspectului paginii:
 - Stilizarea elementelor: culoare, font, etc.
 - Așezarea elementelor în pagina
- Reguli CSS:

```
selector {  
    prop1: valoare1;  
    prop2: valoare2;  
    ...  
}
```

Exemplu:

```
p {  
    color: red;  
    margin-left: 5px;  
}  
  
section {  
    width: 70%;  
    background-color: blue;  
}
```

CSS

- CSS se poate introduce în trei moduri:
 - **Inline:**

```
<section style="width:80% ; background-color:white;">continut-sectiune</section>
```

- **În file:**

```
<head>  
<style type="text/css">  
section {width:80%; background-color:white;}  
</style>
```

- **În fisier separat (varianta recomandată)**

```
<head>  
<link href="nume-fisier.css" type="text/css" rel="stylesheet">  
</head>
```

Acceasi pagina poate fi stilizata diferit prin modificarea fisierului de stil.

```
<head>
<link href="style1.css" type="text/css" rel="stylesheet" >

<!-- <link href="style2.css" type="text/css" rel="stylesheet" > -->
</head>
<body>
<header> PRIMA PAGINA

<nav>
  <a href="#">ACASA</a>
  <a href="#">CONTACT</a>
</nav>
</header>
<div id="container">

<section id="baralat">
  <p>AAAAAA</p>
  <p>BBBBBB</p>
</section>

<section id="main">
<h2> Titlu</h2>
<p>Duis adipiscing nibh sit amet egestas dapibus. Fusce id
quam purus. Quisque fringilla ante at risus tempus, in
tristique nisi accumsan. Morbi vel sollicitudin sem, non
placemat ligula. Ut imperdiet, libero in iaculis egestas, eros
ante pharetra lacus, a cursus nibh lacus et elit. In eget orci
id metus fermentum lacinia. Sed in blandit est, at vehicula
tellus.

Integer vulputate elit in ligula gravida bibendum. Curabitur
venenatis metus in vulputate iaculis. Quisque augue nisi,
bibendum vitae nulla eget, egestas facilisis. Sed in blandit
est, at vehicula tellus.

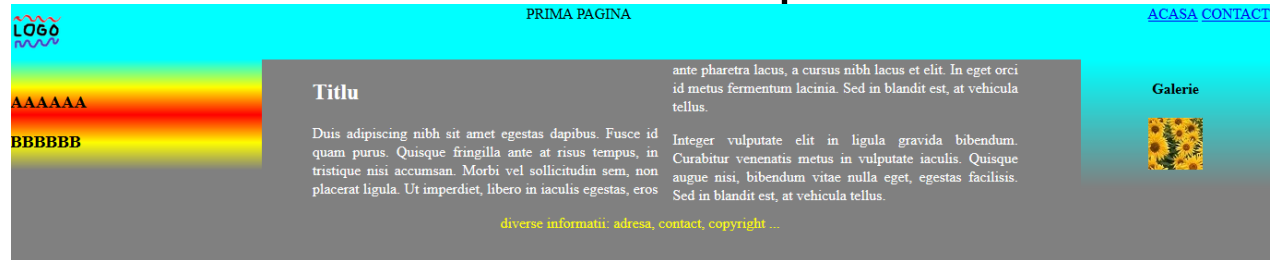
diverse informatii: adresa, contact, copyright ...

</p>
<p>Integer vulputate elit ...</p>
</section>
</div>
<aside>
<p>
<h4> Galerie </h4>

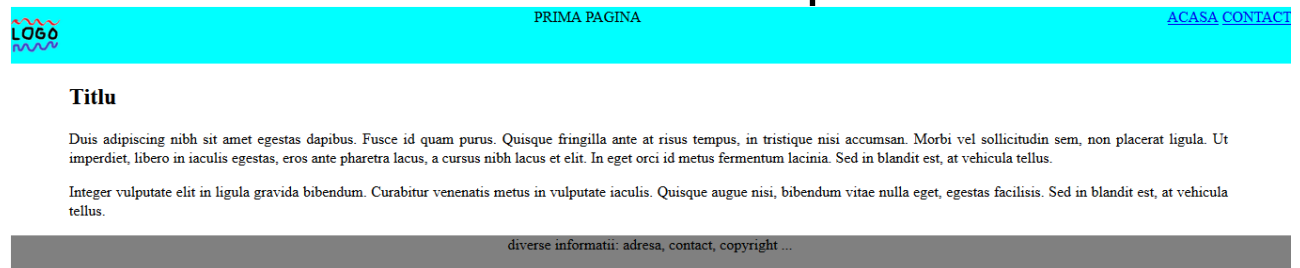
</p>
</aside>

</body>
```

style1.css



style2.css



Precedenta stilurilor

precedenta
cea mai
mica



precedenta
cea mai
MARE

Stilul implicit al browser-ului

Fisierul css extern (<link> in header)
si stiluri “in file” (<style> in header)
(se aplica stilul care apare scris ultimul în <head>)

Stiluri inline

Selectorii CSS

<http://www.w3.org/TR/css3-selectors/>

```
selector1, selector2 { proprietate1:valoare1 }
```

```
selector1 { proprietate2:valoare2 }
```

Aceluiasi element i se pot aplica mai multe reguli;
este bine sa se grupeze elementele care au un stil comun, iar stilurile
specifice sa fie scrise in reguli separate.

Unele stiluri sunt mostenite de la elementele parinte (in arborele asociat
documentului), dar pot fi schimbate folosind reguli specifice.

Proprietăți CSS

color : pentru culoarea textului

background-color: pentru culoarea fondului

Culorile se pot specifica prin:

- Numele culorii: **red, green, blue, ...**

<https://codepo8.github.io/10kb-CSS-colour-game/>

Culori CSS

- Valoarea `rgb(red,green, blue)`,
`red`, `green`, `blue` au valori
intre 0-255
- Valoarea hexazecimala `#rrggbb`,
`rr`,`gg`,`bb` au valori intre 00-ff

`rgb(255, 0, 0)`

`rgb(0, 0, 255)`

`rgb(60, 179, 113)`

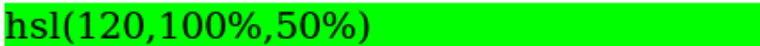
`rgb(238, 130, 238)`

CC6600	CC6633	CC6666
CC9900	CC9933	CC9966
CCCC00	CCCC33	CCCC66
CCFF00	CCFF33	CCFF66
FF0000	FF0033	FF0066

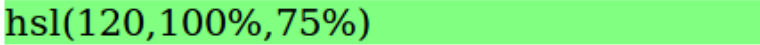
Culori CSS (noua în CSS3)

- Valoarea `hsl(hue, saturation, lightness)`,
`hue` reprezintă gradul culorii (intre 0-360)
`saturation` reprezintă intensitatea (intre 0%-100%)
`lightness` reprezintă luminozitatea (intre 0%-100%)

`hsl(120,100%,50%)`




`hsl(120,100%,75%)`




`hsl(120,100%,25%)`




`hsl(120,60%,70%)`



`hsl(290,100%,50%)`



`hsl(290,60%,70%)`



Culori CSS (noi în CSS3)

- Valoarea `rgba(red,green,blue, alpha)`, este o extensie a valorii `rgb` cu valoarea `alpha` care reprezinta opacitatea (intre 0.0-1.0)
- Valoarea `hsla(hue, saturation, lightness, alpha)` este o extensie a valorii `hsl` cu valoarea `alpha` care reprezinta opacitatea (intre 0.0-1.0)

`rgba(255, 99, 71, 0)`

`rgba(255, 99, 71, 0.2)`

`rgba(255, 99, 71, 0.4)`

`rgba(255, 99, 71, 0.6)`

`rgba(255, 99, 71, 0.8)`

`rgba(255, 99, 71, 1)`

Distanța/marimea în CSS

pot fi măsurate absolut sau relativ.

Unități absolute: mm, cm, px, in, pt

Unități relative: **em** = dimensiunea fontului curent
% = procente din dimensiunea
parintelui / fontului curent

```
<div id="vara">  
  
</div>
```

```
<style>  
div #vara {width: 30px;  
           height: 20px;}  
</style>
```

HTML (unitatea de măsură este px)

```

```

Selectori CSS

- Selectorul universal *

Exemplu: `* {border: 1px solid blue;}`

Titlu cu h1
Paragraful1 in div
Paragraful2 in div
Paragraful3 in div
Paragraful4

- Selectorul de tip (type selector) : este numele unui element din documentul HTML: **h1, section, p, div, ..**

Exemplu:

```
p {color: green;  
margin-left: 25px;  
}
```

HTML

```
<body>  
<h1>Titlu cu h1</h1>  
<div>  
<p>Paragraful1 in div</p>  
<p>Paragraful2 in div</p>  
<p>Paragraful3 in div</p>  
<div>  
<p>Paragraful4 </p>  
</body>
```

Titlu cu h1

Paragraful1 in div

Paragraful2 in div

Paragraful3 in div

Paragraful4

Selectori CSS

Attribute selector

`e[attr]` selecteaza elem. `e` care au `attr` ca atribut

`e[attr="val"]` selecteaza elem. `e` cu `attr="val"`

`e[attr~="val"]` selecteaza elem. `e` cu proprietatea ca valoarea lui `attr` contine cuvantul `val`

Exemplu: `a [href="http://www.w3.org/"]`

Selectorul prin id

Element HTML : <tag id="myid"> continut </tag>

(un document HTML poate contine un singur atribut id cu valoarea "myid")

#myid selecteaza elementul cu id="myid"

Exemplu:

<div id="rosu"> continut </div>

poate fi selectat folosind

#rosu sau div#rosu

Selectorul prin clasa (**class selector**)

`<tag1 class="newcl"> continut </tag1>`

`<tag2 class="newcl"> continut </tag2>`

Cu atributul **class** grupam mai multe elemente, care vor avea un stil comun; poate apare de mai multe ori in acelasi document HTML.

`.newcl` selecteaza elementele cu `class="newcl"`

Exemplu: ` continut `
este selectat folosind `span.rosu`

Selectarea prin **id** are specificitate mai mare decat selectarea prin **class**

```
#id {color:green;}  
.class {color: blue;}
```

Exemplu

```
<p class="class" id="id"> Exemplu </p>
```

Cand doua stiluri au aceeasi specificitate, castiga cel care este scris ultimul in fisierul CSS.

```
.class1 {color: blue;}  
.class2 {color: green;}
```

Exemplu

```
<p class="class2 class1">Exemplu</p>
```

Observatii:

Proprietatile marcate **!important** sunt aplicate indiferent de specificitatea lor

```
p { color: #ff0000 !important; }
```

Unele proprietati pot lua valoarea **inherit**, aceasta insemnand ca proprietatea este mostenita de la elementul parinte; este folosita pentru a suprascrie un alt stil si a forta comportamentul implicit.

Combinarea selectorilor

Selectori pentru descendenți

$E F$ selectează toate elementele F care sunt descendenți ai lui E

Selectori pentru copii

$E > F$ selectează elementele F care sunt copii ai lui E

Selectori pentru frați

$E + F$ selectează toate elementele F care sunt frați cu E și care urmează imediat după E

$E \sim F$ selectează toate elementele F care sunt frați cu E și care urmează după E

Exemple

```
div p {  
  background-color: yellow;  
}
```

HTML

```
body>  
<div>  
  <p>Paragraph 1 in the div.</p>  
  <p>Paragraph 2 in the div.</p>  
  <span><p>Paragraph 3 in the div.</p></span>  
</div>  
<p>Paragraph 4. Not in a div.</p>  
<p>Paragraph 5. Not in a div.</p>  
</body>
```

```
div > p {  
  background-color: yellow;  
}
```

Descendent

Paragraph 1 in the div.

Paragraph 2 in the div.

Paragraph 3 in the div.

Paragraph 4. Not in a div.

Paragraph 5. Not in a div.

Copil

Paragraph 1 in the div.

Paragraph 2 in the div.

Paragraph 3 in the div.

Paragraph 4. Not in a div.

Paragraph 5. Not in a div.

Adjacent sibling

```
<body>
<div>
  <p>Paragraph 1 in the div.</p>
  <p>Paragraph 2 in the div.</p>
</div>

<p>Paragraph 3. Not in a div.</p>
<p>Paragraph 4. Not in a div.</p>

</body>
```

```
div + p {
  background-color: yellow;
}
```

Paragraph 1 in the div.

Paragraph 2 in the div.

Paragraph 3. Not in a div.

Paragraph 4. Not in a div.

General sibling

```
<body>

<p>Paragraph 1.</p>

<div>
  <code>Some code.</code>
  <p>Paragraph 2.</p>
</div>

<p>Paragraph 3.</p>
<code>Some code.</code>
<p>Paragraph 4.</p>

</body>
```

```
div ~ p {
  background-color: yellow;
}
```

Paragraph 1.

Some code.

Paragraph 2.

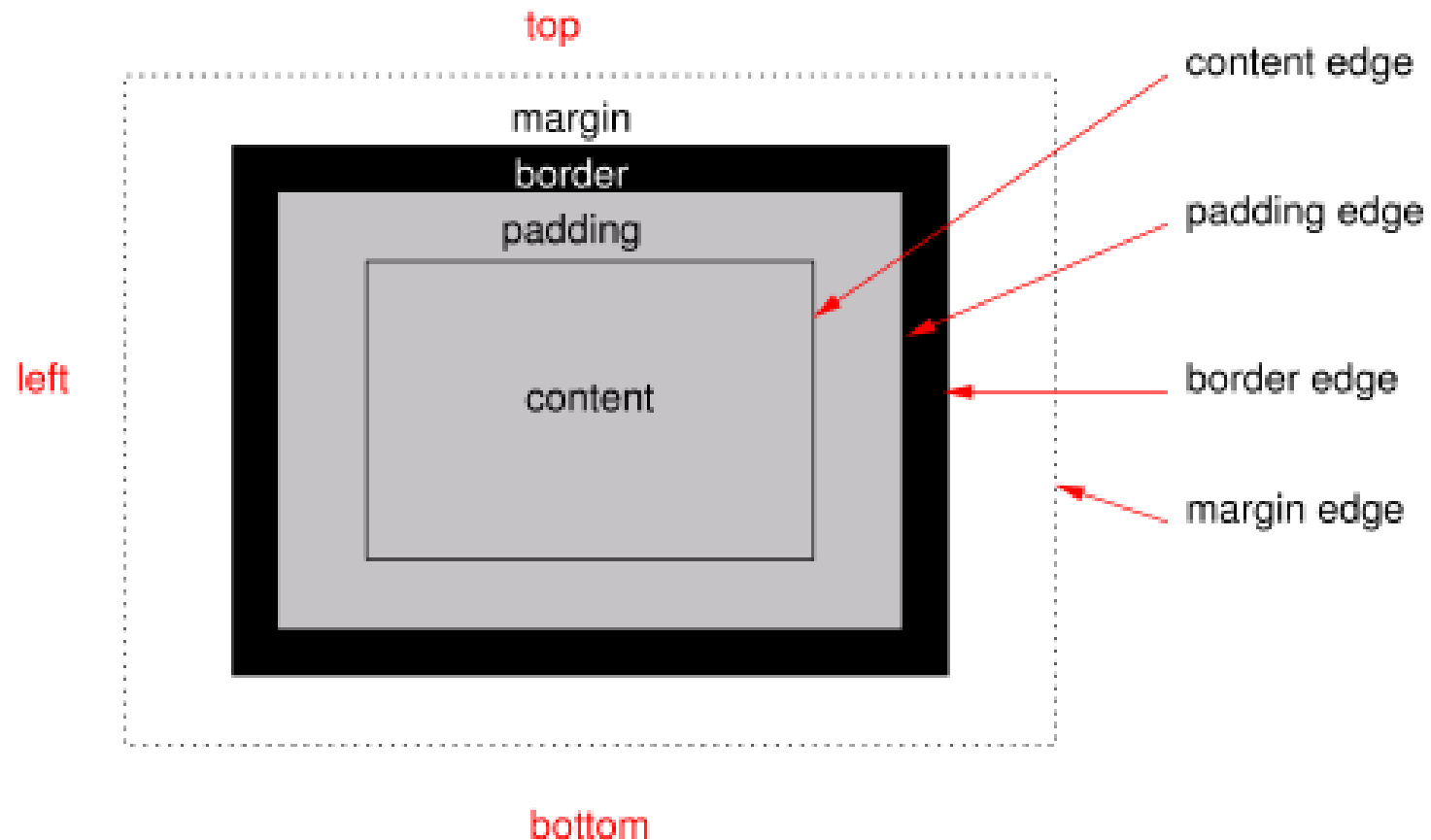
Paragraph 3.

Some code.

Paragraph 4.

“Box Model” pentru elemente HTML

<http://www.w3.org/TR/css3-box/>



https://www.w3schools.com/cssref/css_default_values.asp

Proprietati CSS: border

`border: grosime stil culoare;`
`border-left: grosime stil culoare;`
`border-right: grosime stil culoare;`
`border-top: grosime stil culoare;`
`border-bottom: grosime stil culoare;`

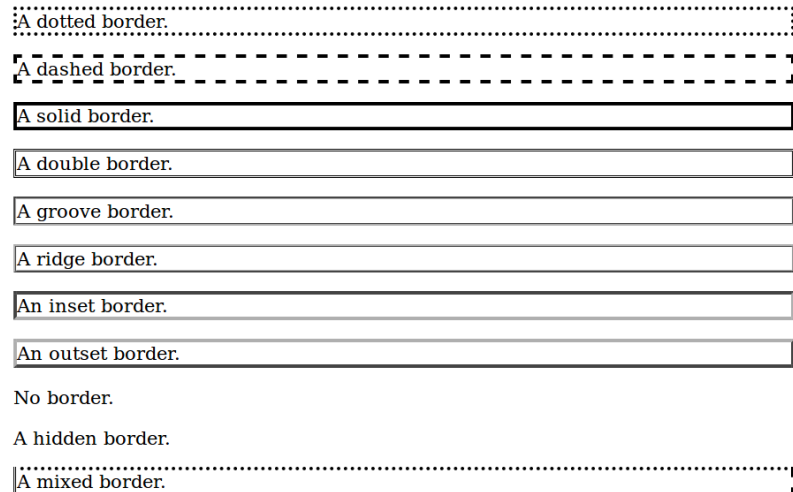
`border-color:`
`border-style:`
`border-width:`
`border-left-color:`
`border bottom-width:`
...

grosime: exprimata in `px`, `pt`, `em`, sau
`thin`, `medium`, `thick`

stil: `none`, `solid`, `dotted`, `dashed`, `inset`,
`outset`, `double`, `groove`, `ridge`,
`hidden`

`border-radius: 12px;`

margini cu colturi rotunde



Proprietatea `border-color` nu funcționează singura, trebuie obligatoriu și `border-style`

CSS

```
<style>  
body{ border: thick ridge red; width:30%;}  
</style>
```

Doua exemple

[Sectioni cu articole](#)

[Articol cu sectiuni](#)

Proprietati CSS: margin, padding

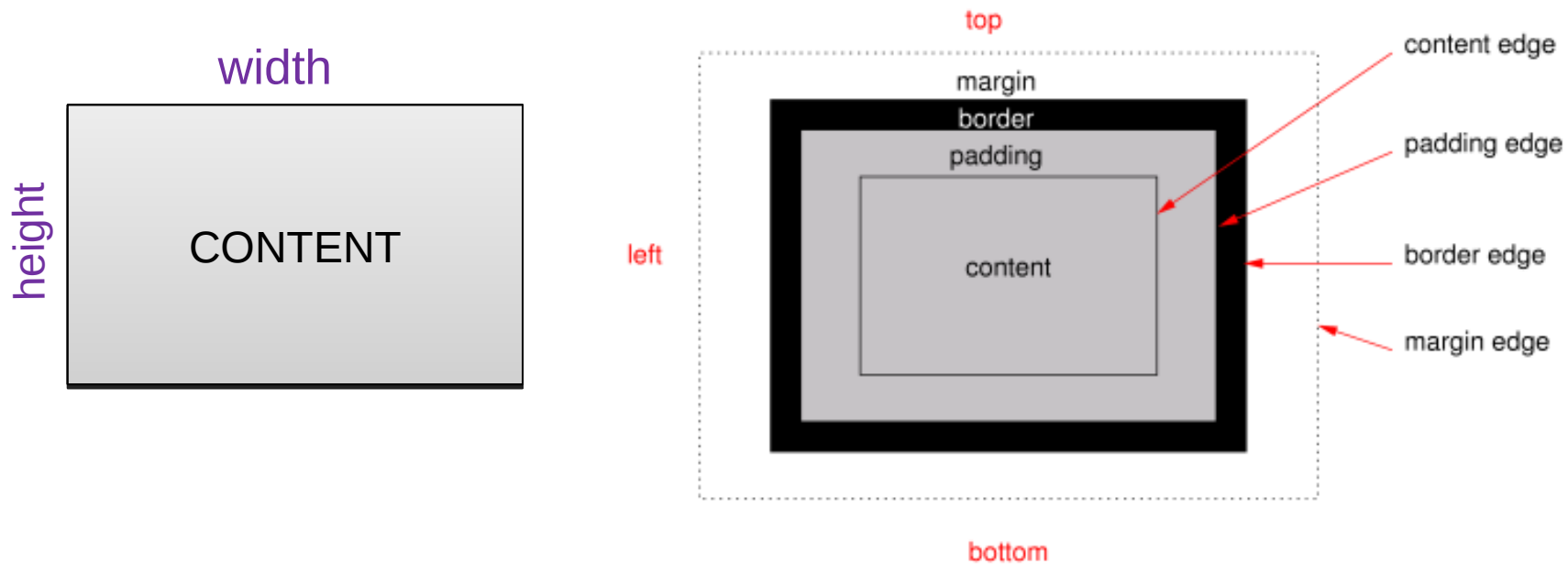
```
padding: 15px;  
padding-left: 10%;  
padding-right: 20%;  
padding-top: 15px;  
padding-bottom: 15px;
```

```
margin: 15px;  
margin-left: 10%;  
margin-right: 20%;  
margin-top: 15px;  
margin-bottom: 15px;
```

Padding și margin au background transparent

```
margin: 15px 20% 15px 10%;  
padding: 15px 20% 15px 10%;  
           ↓   ↓   ↓   ↓  
         top right bottom left
```

Proprietati CSS: width, height



```
width: 500px /* 50% */  
/* height, max-width, min-width,  
   max-height, min-height */
```

Exemplu

```
<head>
<style>
body{background-color: pink;}
div {
    background-color: lightgrey;
    width: 100px;
    border: 25px solid green;
    margin: 30px;
    padding: 15px;
}
</style>
</head>

<body>
<div>Un div stilizat folosing
proprietary background,
width, border, margin,
padding</div>
</body>
```



Proprietati CSS: fonturi

<http://www.w3.org/Style/Examples/007/fonts>

```
font-family: Helvetica, "Times New Roman", serif;  
font-size: 16px /* 16pt, 1.25em, 125%, x-large */  
font-weight: normal /* bold /* lighter / * 100, ..., 900 */  
font-style: normal /* italic /* oblique */  
font-variant: normal /* small-caps */  
font-stretch: normal /* expanded / * condensed /*  
                semi- /* ultra- /* extra- */
```

font-family:

Fonturi generice: serif, sans-serif, monospace,...

Fonturi specifice: Times New Roman, Arial, Georgia, Verdana

font-size:

Unitati: px, pt, em (m-size: 1em = dimesiunea fontului curent)

Valori: xx-small, x-small, small, medium, large, x-large, xx-large,...

Procente: procent din fontul parintelui

at-rules @font-face

<http://www.w3.org/TR/css-syntax-3/>

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

```
@font-face{ font-family: nume-font;  
            src:url(fis-font.ext);  
            }
```

```
selector { font-family: nume-font;}  
</style>
```

at-rules sunt reguli pentru parserul CSS
al browserului


```
<style type="text/css">
@font-face{font-family:fontul-meu;
            src:url(Atlantic%20Bentley.ttf)}

h1 {font-family:fontul-meu;}

</style>
```

Titlu scris cu font Atlantic Bentley

www.dafont.com/

Proprietati CSS: proprietati ale textului

<http://www.w3.org/TR/CSS2/text.html>

```
color: nume-culoare /* valori */;
text-align: center /*left, right, justify */
text-decoration: underline /* overline, line-through */
text-shadow: -2px 5px 2px gray /*none/*initial;
             /* x-offset, y-offset, blur-radius, culoare */
text-indent: 3em /* aliniere prima linie */
text-transform: uppercase /* lowercase /* capitalize
line-height: 150% /* 1.5em , 30px */
word-spacing: 1em /* +15mm */
letter-spacing: 0.1em /* -1px */
```

The default line height in most browsers is about 110% to 120%.

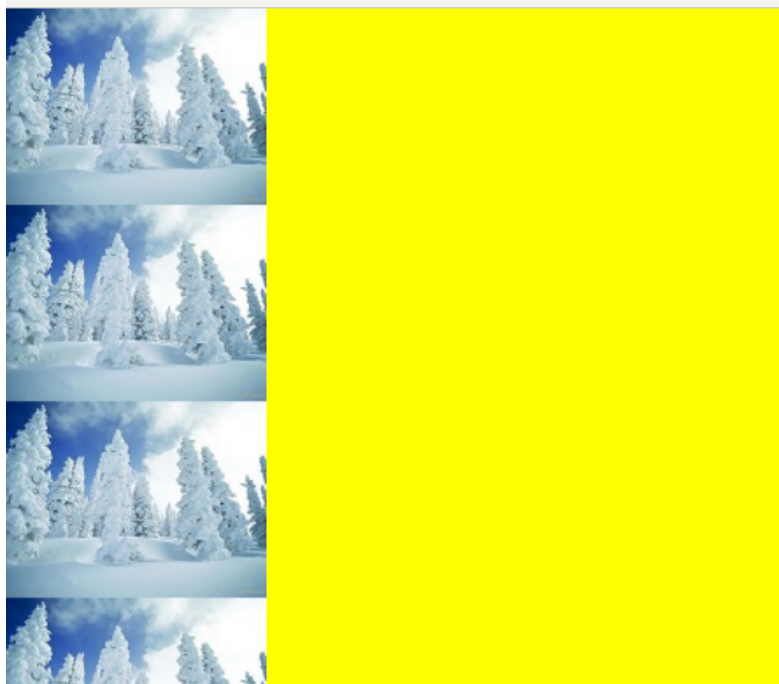
Proprietati CSS: background

```
background-color: culoare;  
background-image: url(adresa-fisier);  
background-repeat: repeat (implicit)  
                        /*repeat-x, repeat-y, or no-repeat*/  
background-position: 370px 20px;  
                        /* top, left, right, bottom, center, procente */  
background-size: 300px 100px(dimensiunea);  
                        /*contain, cover, 100% 100%, auto*/  
background-attachment: fixed /* scroll  
(implicit)*/
```

Proprietati CSS: multiple backgrounds

```
background-image: url(img3), url(img2), url(img1);  
background-position: center center, 20% 80%, top left;  
background-repeat: no-repeat;  
/* no-repeat, no-repeat*/
```

```
<style type="text/css">
body{ background-color: yellow;
background-image: url(larna.jpg);
background-repeat: repeat-y;
}
</style>
```



```
<style type="text/css">
body{ background-color: yellow;
      background-image: url(Primavara.jpg),
url(Vara.jpg), url(larna.jpg);
      background-repeat: repeat-y, repeat-x, repeat;
}
</style>
```



```
<style type="text/css">
body{ background-color: yellow;
      background-image: url(Primavara.jpg), url(larna.jpg), url(Vara.jpg);
      background-repeat: no-repeat;
      background-position: 20px 20px, 100px 100px, 200px 200px;
    }

</style>
```



Pseudo-clase CSS

- Sunt folosite pentru a defini starea unui element:
 - Stilizarea unui element la trecerea cu mouse-ul
 - Stilizarea diferita a linkurilor vizitate și nevizitate
 - Stilizarea unui element când are focus (ex. pentru elemente de tip input)

Pseudo-clase

- Sintaxa: `selector:pseudo-class {`
`proprietate:valoare;}`

`:link` legaturi care nu au fost vizitate

`:visited` legaturi care au fost vizitate

`:hover` legaturi peste care se trece
cu mouse-ul

`:active` legaturi care sunt active

```
a:link {text-decoration:none; color: blue;}  
a:visited {text-decoration:none; color:green;}  
a:hover {text-decoration:underline; color:red;}
```


Pseudo-clase

:first-child selecteaza elementul specificat care este primul copil al altui element

:last-child selecteaza elementul specificat care este ultimul copil al altui element

```
<style type="text/css">
p:first-child{color:red;}
p:last-child{color: blue;}
</style>
</head>
<body>
<p>Paragraful 1</p>
<p>Paragraful 2</p>
<p><b>Acest paragraf</b> contine text ingrosat</p>
<div>
<p>Paragraful 3</p>
<p>Paragraful 4</p>
</div>
</body>
```

Paragraful 1

Paragraful 2

Acest paragraf contine text ingrosat

Paragraful 3

Paragraful 4

Pseudo-clase

:first-of-type selectează elementul care este primul copil de tipul specificat al altui element

:nth-of-type(n) selectează elementul care este al n-lea copil de tipul specificat al altui element

:nth-child(n) selectează elementul de tipul specificat care este al n-lea copil al altui element

n poate fi: numar natural

 Cuvânt cheie (odd, even)

 Formula: $an+b$ (n pornește de la 0)

```
<style type="text/css">
p:nth-of-type(2){color:red;}
p:nth-of-type(odd){color: blue;}
p:nth-of-type(4n+0){color: green;}
</style>
</head>
<body>
<p>Paragraful 1</p>
<p>Paragraful 2</p>
<p>Paragraful 3</p>
<p>Paragraful 4</p>
<p>Paragraful 5</p>
<p>Paragraful 6</p>
<p>Paragraful 7</p>
<p>Paragraful 8</p>
<p>Paragraful 9</p>
<p>Paragraful 10</p>
<p>Paragraful 11</p>
<p>Paragraful 12</p>

</body>
```

Paragraful 1

Paragraful 2

Paragraful 3

Paragraful 4

Paragraful 5

Paragraful 6

Paragraful 7

Paragraful 8

Paragraful 9

Paragraful 10

Paragraful 11

Paragraful 12

```
<style type="text/css">
p:nth-child(3) {
    background: red;
}
p:nth-of-type(3) {
    background: green;
}

</style>
</head>
<body>
<div>Urmeaza 12 paragrafe</div>
<p>Paragraful 1</p>
<p>Paragraful 2</p>
<p>Paragraful 3</p>
<p>Paragraful 4</p>
<p>Paragraful 5</p>
<p>Paragraful 6</p>
<p>Paragraful 7</p>
<p>Paragraful 8</p>
<p>Paragraful 9</p>
<p>Paragraful 10</p>
<p>Paragraful 11</p>
<p>Paragraful 12</p>

</body>
```

Urmeaza 12 paragrafe

Paragraful 1

Paragraful 2

Paragraful 3

Paragraful 4

Paragraful 5

Paragraful 6

Paragraful 7

Paragraful 8

Paragraful 9

Paragraful 10

Paragraful 11

Paragraful 12

```

<style>
table {
    font-family: arial, sans-serif;
    border-collapse: collapse;
    width: 30%;
}

td, th {
    border: 2px solid black;
    text-align: left;
    padding: 8px;
}

tr:nth-child(even) {
    background-color: #dddddd;
}
</style>

```

Nume	Prenume	Varsta
Popescu	Maria	30
Ionescu	Bogdan	26
Sandu	Mihai	14
Badea	Ana	24
Preda	Dan	20
Iancu	Mara	40

Pseudo-elemente CSS (: sau ::)

elemente virtuale, create cu CSS,
care nu apar in DOM

:first-letter prima litera a unui text

:first-line prima linie a unui element bloc

:before sunt folosite pentru a adauga

:after continut inainte/dupa un element
HTML; se folosesc cu proprietatea
content

Pseudo-elemente CSS (: sau ::)

```
<style>
```

```
.email-address:before { content: "Email address: " ;}
```

```
</style>
```

```
</head>
```

```
<body>
```

```
<ul>
```

```
<li class="email-address">myemail@gmail.com</li>
```

```
</ul>
```

Proprietati CSS: visibility, overflow

visibility: visible /* hidden */

seteaza vizibilitatea unui anume element;
chiar daca vizibilitatea este "hidden"
elementul ocupa loc in pagina

overflow: hidden; /* visible, scroll */

descrie modul in care este redat
textul care iese dintr-un cadru

Meniu drop-down

```
ul ul {visibility:hidden;}  
ul li:hover ul {color:blue; visibility:visible;}
```

```
<ul>  
<li>MENIU 1  
    <ul><li> submeniu 1.1</li>  
        <li> submeniu 1.2 </li>  
    </ul> </li>  
<li>MENIU 2  
    <ul><li> submeniu 2.1</li>  
        <li> submeniu 2.2</li>  
    </ul> </li>  
</ul>
```

Exemplu : overflow

```
<style>  
  div {width:2cm; height:2cm; overflow: scroll;}  
</style>
```

```
<div>  
  Acesta este un text foarte lung si important.  
</div>
```

