

# Universitatea Tehnică "Gheorghe Asachi" Iași

## Facultatea de Electronica, Telecomunicatii si Tehnologia Informatiei

---



## Atelier

### Creație pagini web



conf. dr. ing. Radu Damian  
sl. dr. ing. Daniel Mătășaru

---



# Introducere

“I.Like.IT@ETTI.TUIASI.RO”

[www.tuiasi.ro](http://www.tuiasi.ro)

1813

1860

1910

1937

1993

## CSS

- Cascading Style Sheets
- Introdus pentru a separa **continutul** documentului de **prezentarea** sa (forma, culori, caractere)
- Permite:
  - Accesibilitate crescuta (acelasi document poate fi afisat diferit pe dispozitive diferite)
  - Tratare unitara a mai multor pagini
  - Reduce complexitatea pozitionarii

## CSS vs. HTML

```
<!DOCTYPE html PUBLIC
<html>
<!-- created 2003-12-12 -->
<head>
  <title>sample</title>
</head>
<body>
  <p>Voluptatem accusantium
    totam rem aperiam.</p>
</body>
</html>
```

**HTML**

```
h1 { color: white;
      background: orange;
      border: 1px solid bla
      padding: 0 0 0 0;
      font-weight: bold;
}
/* begin: seaside-theme */

body {
  background-color:white;
  color:black;
  font-family:Arial,sans-serif;
  margin: 0 4px 0 0;
  border: 12px solid;
}
```

**CSS**

# Cascading Style Sheets



- CSS 1 – 1996 (forma)
- CSS 2 – 1998 (pozitionare)
- CSS 2.1 – 2007 (corectii)
- CSS 3.0 – module independente
  - 2011/2012 ->
- CSS 4.0 – module independente
  - Draft 2018 ->
- Utilitatea dependenta de suportul din partea dispozitivelor de afisare
- Majoritatea browser-elor suporta CSS 2.
  - suport partial CSS 3.0 pentru majoritatea browser-elor moderne
- CSS implica degradare eleganta (“**degrade gracefully**”)

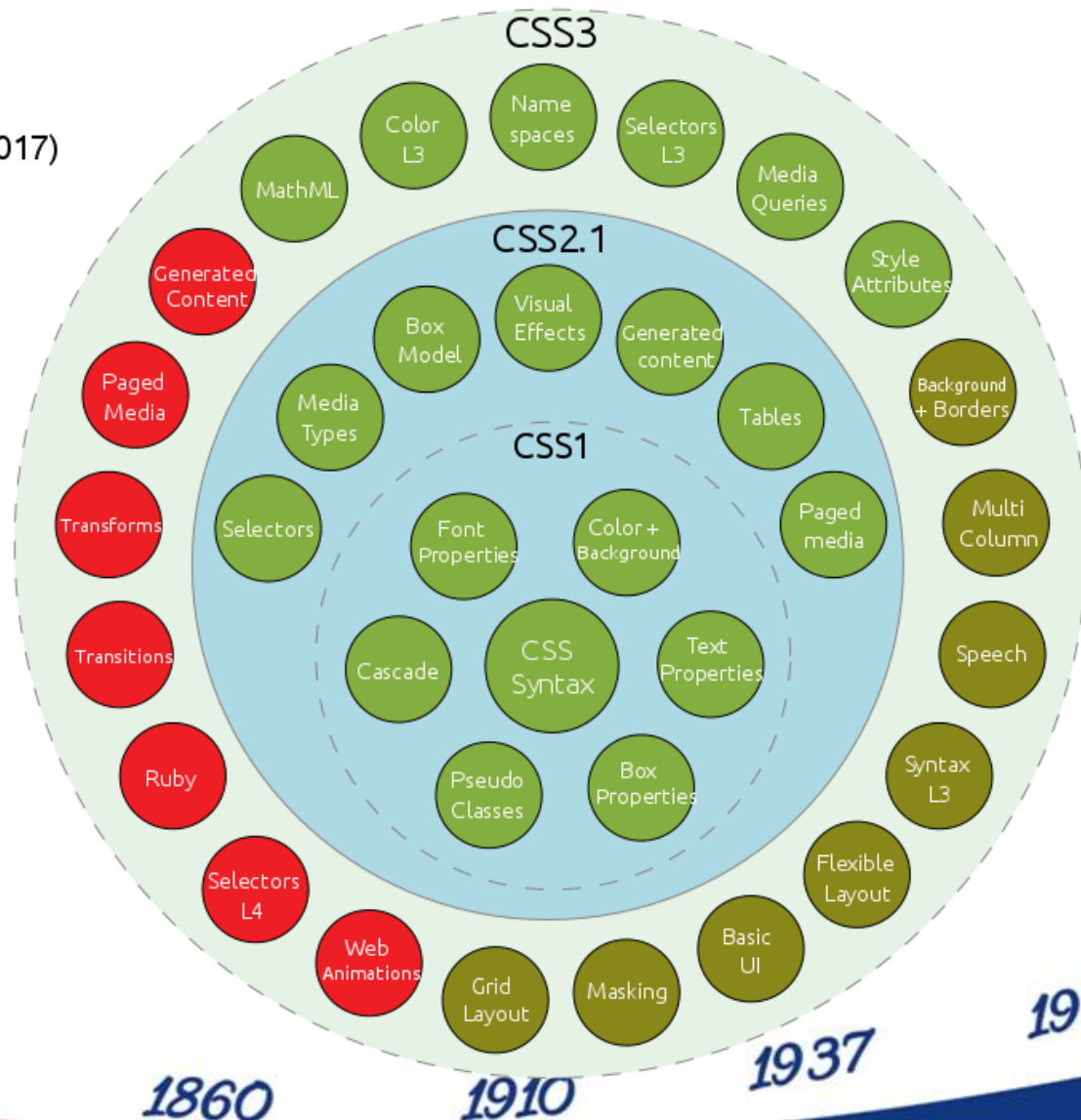


# CSS3

# CSS3

Taxonomy & Status (September 2017)

- W3C Recommendation
- Candidate Recommendation
- Last Call
- Working Draft
- Obsolete or inactive



## HTML vs. CSS

- HTML
  - **structura** logica a documentului,
- CSS
  - **forma** efectiva
- HTML 5
  - Etichete HTML privitoare la **formă** (<font>, color=“red”) au fost introduse in HTML 3.2 si declarate **depreciate** in HTML 4.01
    - dispar in HTML 5
  - Intercalarea intre elementele de structura a celor privitoare la formă in HTML duce la cresterea complexitatii documentului si creste inutil dimensiunea acestuia
    - Exemplu tipic: Microsoft Word -> Save as HTML -> documente de 10 ori mai mari decat un document cu aceeasi forma scris pur in HTML

~1930

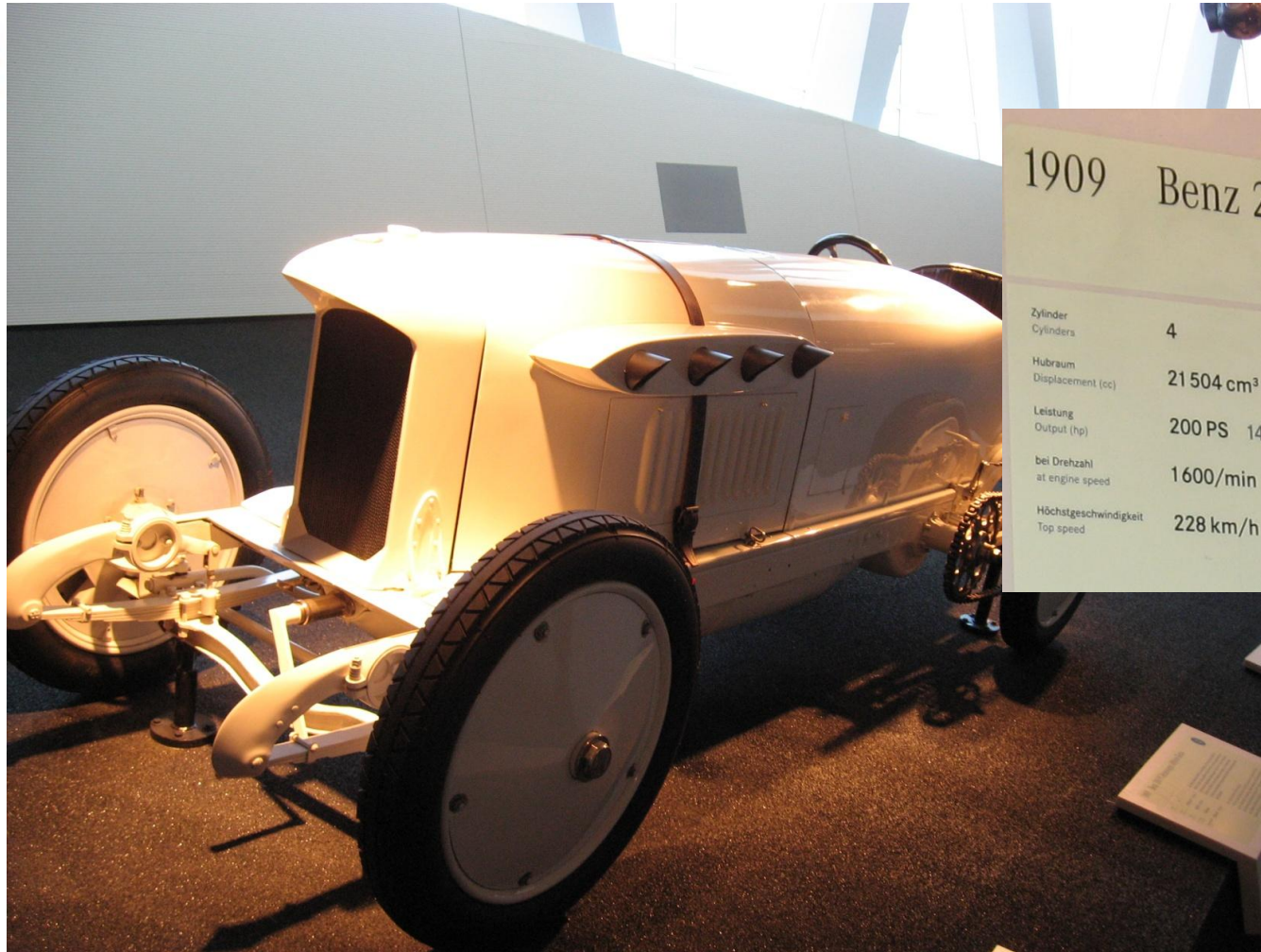




~1930



## 1909



1909 Benz 200 PS Rennwagen »Blitzen«	
Zylinder Cylinders	4
Hubraum Displacement (cc)	21 504 cm <sup>3</sup> 1 312 cu in
Leistung Output (hp)	200 PS 147 kW
bei Drehzahl at engine speed	1 600/min
Höchstgeschwindigkeit Top speed	228 km/h 142 mph

Der »Blitzen-Benz« ist 1909 der erste 200 km/h fährt. Seine größten Erfolge erzielt er mit dem 2000 cm<sup>3</sup> Zylindermotor ausgestatteten Rekordwagen mit 228 km/h über die Sahara. Er ist damit das schnellste Fahrzeug der Welt.

Benz »Lightning Benz« 200 hp racing car  
In 1909 the Lightning Benz was the fastest car in the world.

[www.tuiasi.ro](http://www.tuiasi.ro)

1813

1860

1910

1937

1993



1930-1950



[www.tuiasi.ro](http://www.tuiasi.ro)

1813

1860

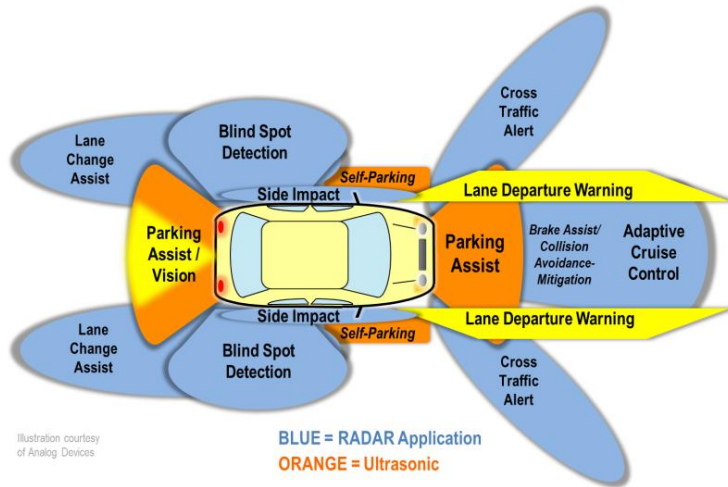
1910

1937

1993

# Tehnologie

> 2010



< 1950



www.tuiasi.ro

1813

1860

1910

1937

1993



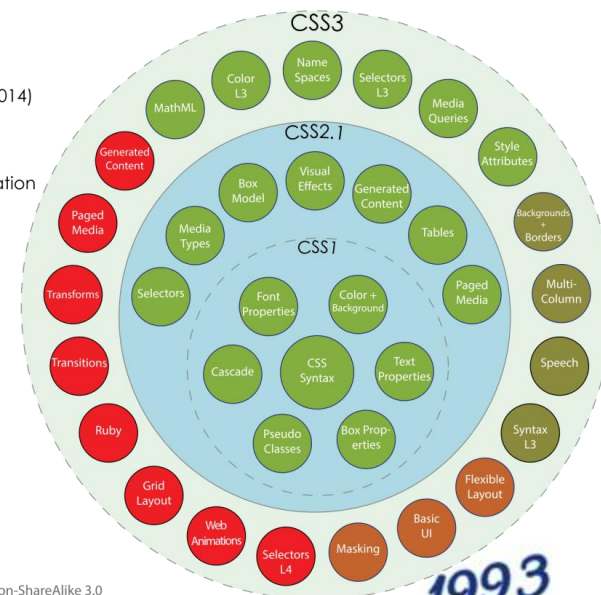
## HTML/XHTML

- HTML 4.01 – 24.12.1999
- XHTML 1.1 – Mai 2001
- CSS 2.1 – 2004-2007

## CSS3

Taxonomy & Status (October 2014)

- W3C Recommendation
- Candidate Recommendation
- Last Call
- Working Draft
- Obsolete or inactive



By Sergey Mavrody 2011-14 | CC Attribution-ShareAlike 3.0

## HTML/CSS vs. C++

```
int i,a;
for (i=1;i<=10;i++) {
    a=2*i;
    printf(“%d\n”,a);
}
```

- C++/Pascal/etc.
  - actiuni
  - genereaza date

```
<p>Text Text Text <font color=“red”>Text
</font>Text Text Text</p>
<p>Text <strong>Text </strong>Text Text
Text Text Text
</p>
```

- HTML/CSS/etc.
  - forma
  - a datelor existente

**Javascript, PHP**



# Concepte CSS

**“I.Like.IT@ETTI.TUIASI.RO”**

[www.tuiasi.ro](http://www.tuiasi.ro)

1813

1860

1910

1937

1993

## CSS Concepte

- Un document CSS e un document **ASCII** (Notepad) care contine **specificatii CSS**
- **Specificatiile** CSS sunt de forma:
  - Selector { “regula CSS”; [“regula CSS”; ] }
- Reguli CSS de forma:
  - atribut:valoare;
- Comentarii
  - Similar cu C/C++ , PHP, etc.
  - Cuprinse intre “/\*” si “\*/”



## Selector CSS

- Selector **HTML**
  - conceput pentru a modifica modalitatea de afisare standard a **etichetelor HTML**
- Selector **clasa**
  - conceput pentru a lucra cu atributul HTML **class**="..."
  - util pentru aplicare **uniforma** a formei mai multor elemente din document
- Selector **identificator**
  - conceput pentru a lucra cu atributul HTML **id**="..."
  - util pentru aplicarea unei forme **specifice** unui anume element din document

## Selector CSS

- Redefinirea etichetelor HTML standard
  - exemplu: afisarea implicita a tuturor celulelor de tabel din document este schimbata
- Redefinirea se poate face **si** doar in interiorul unei clase
  - exemplu: in interiorul claselor “footer” link-urile vor fi afisate altfel

```
td {  
  font-family:Tahoma, Verdana;  
  font-size:14px;  
  color:#D5A787;  
  vertical-align: top;  
  text-align: center;  
}
```

```
.footer a {  
  color:#FFD2B3;  
  text-decoration:none;  
}  
.footer a:hover {  
  color:#ffffff;  
  text-decoration:none;  
}
```

## Selector CSS

- **clase – “.”**

- Semnul “.” inaintea numelui
- permit reutilizarea unei anumite forme grafice
  - `<p class="menu">...</p>`
  - `<h1 class="menu">...</h1>`
  - `<tr class="menu">...</tr>`

- **identificator element – “#”**

- Semnul “#” inaintea numelui
  - `<tr class="menu">...<td id="menu_curent">...</td>...</tr>`

```
.menu {  
  font-family:Tahoma, Verdana;  
  font-size:14px;  
  color:#6F3A15;  
  text-align:center;  
  font-weight:bold;  
}
```

```
#menu_curent {  
  font-family:Tahoma, Verdana;  
  font-size:15px;  
  color:#FFFFFF;  
  text-align:center;  
  font-weight:bold;  
}
```

## Selector CSS

- selector clasa si identificator concepute pentru a fi utilizate cu elementele de grupare HTML
  - `<div>...</div>` : grupare de tip bloc (“block”)
  - `<span>...</span>` : grupare inserata (“in-line”)
- anumite reguli au efect numai daca sunt atribuite unor elemente corespunzatoare (block sau in-line)



## Specificari complexe ale selectorilor

- **.clasa1** {} – pentru orice element caruia i se atribuie atributul HTML class=“clasa1”
- **p.clasa1** {} – numai paragrafelor care sunt afectate clasei “clasa1” <p class=“clasa1”>...
- **.clasa1 p** {} – oricarui paragraf aflat in interiorul unui bloc afectat clasei “clasa1”
  - <div class=“clasa1”><p>...</p><p>...</p></div>
- in anumite conditii speciale (mai ales pentru eticheta HTML <a>)
  - **a:hover** {} , **a:visited** {}

## Specificitati CSS (ponderi)

Selectors	Specificity
H1 {color: white;}	1
P EM {color: green;}	2
.grape {color: red;}	10
P.bright {color: blue;}	11
P.bright EM.dark {color: yellow;}	22
#id218 {color: brown;}	100
style=" "	1000

## Exemplu specificitati

- style="color: blue"
- suprascrie
- #xyz { color: red; }

```
< <html>
<head>
<meta charset="utf-8" />
<style type="text/css">
#xyz { color: red; }
</style>
</head>
<body>
<p id="xyz" style="color: blue"> Demonstratie </p>
</body>
</html>
```

## CSS concepte

- Trei modalitati de a specifica stiluri
  - stiluri externe (**External style sheet**)
  - stiluri interne (**Internal style sheet**)
  - stiluri inserate (**Inline style**)
- Trei entitati care impun stiluri:
  - **dispozitivul** de afisare (browser)
  - **autorul** documentului
  - **cititorul** documentului



## CSS concepte

- In orice moment un element din document poate fi sub controlul mai multor reguli care se contrazic
- **Ordinea prioritatilor** (in ordinea cresterii acesteia)
  - **dispozitivul** de afisare (browser)
  - **autorul** documentului – in ordinea definirii. De obicei:
    - External style sheet
    - Internal style sheet
    - Inline style
  - **cititorul** documentului (in diverse forme, “text size”, “zoom” de ex)
  - Prioritatea maxima data cititorului documentului este in dezbatere si inconsistenta (nu la toate regulile)

## Ordinea prioritatilor CSS

High Priority	CSS Source Type	Description
1	Importance	The ‘!important’ value overwrites the previous priority types
2	Inline	A style applied to an HTML element via HTML ‘style’ property
3	Media Type	A property definition applies to all media types, unless a media specific CSS defined
4	User defined	Most browsers have the accessibility feature: a user defined CSS
5	Selector specificity	A specific contextual selector (#heading p) overwrites generic definition
6	Rule order	Last rule declaration has a higher priority
7	Parent inheritance	If a property is not specified, it is inherited from a parent element
8	CSS property definition in HTML document	CSS rule or CSS inline style overwrites a default browser value
9	Browser default	The lowest priority: browser default value is determined by W3C initial value specifications

## External style sheet

- regulile CSS se gasesc intr-un fisier extern, de obicei cu extensia “.css”
- fisierul respectiv este incarcat in document in sectiunea antet
  - `<head><link rel="stylesheet" type="text/css" href="stiluri.css" /></head>`
- utilizare obligatorie cand o suita de documente au o prezentare comuna

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

## External style sheet

- Dezavantaje – minore in conditiile actuale
  - necesitatea unei conexiuni suplimentare la server
  - cantitatea de date mai mare (doar la prima pagina incarcata – se aduc si stilurile care nu sunt folosite)
- Avantaje
  - separa total **continutul** de **forma**
  - tratare unitara a paginilor, usureaza
    - navigarea, dpdv. al utilizatorului (**Don't make me think!**)
    - intretinerea si modificarea intregului site, dpdv. al designer-ului
  - dimensiune minima a documentelor (o regula se aplica de multe ori dar se scrie o singura data)
  - incepand de la a doua pagina din suita, browser-ul este deja in posesia stilurilor, viteza e sporita

## Internal style sheet

- Regulile CSS sunt indicate de asemenea in sectiunea antet (<head>... </head>)
- Se foloseste **eticheta** HTML **<style>...</style>**

- Se foloseste cand un document are o forma unica in site

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



## Internal style sheet

- Avantaj
  - se elimina conexiunea suplimentara la server
  - separa **continutul** de **forma**
- **Dezavantaj/Avantaj**
  - Daca in sectiunea antet apare dupa fisierul extern de stiluri, va suprascrie stilurile corespunzatoare
- Dezavantaj
  - nu permite tratarea uniforma a documentelor

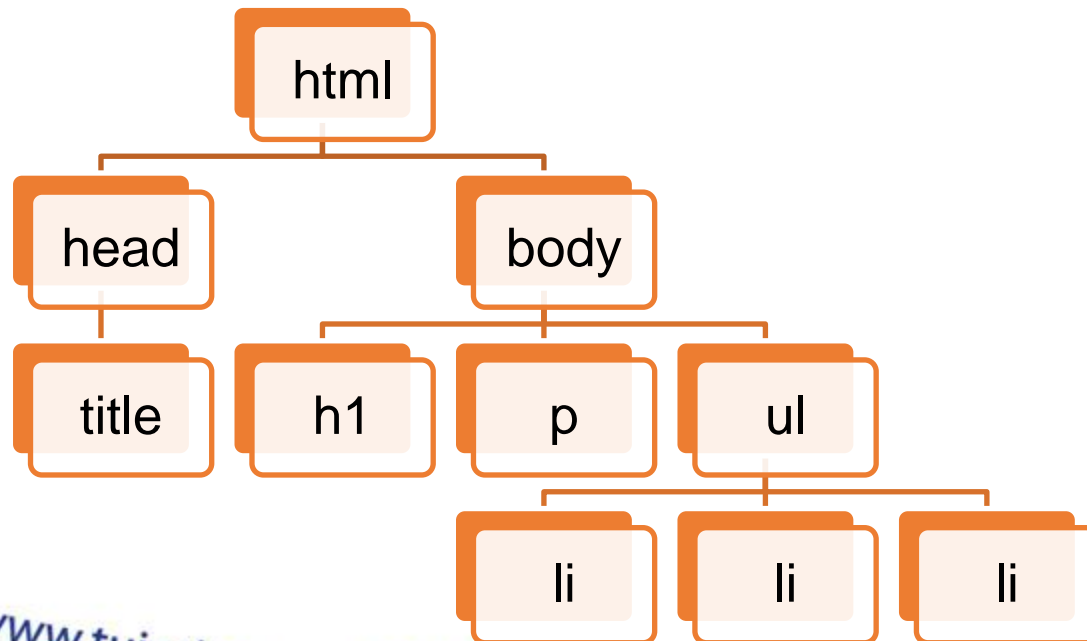
## Inline style sheet

- Se utilizeaza **atributul HTML style=“...”**
- Modifica o singura eticheta
  - `<p style="color:sienna;margin-left:20px">Un paragraf</p>`
- De obicei utilizat punctual pentru a introduce efecte ce nu pot fi obtinute cu HTML
- De evitat:
  - **nu** separa forma de continut
  - creste dimensiunea documentului (vezi un fisier HTML salvat cu Microsoft Word)

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

## Relatia parinte copil si mostenire

- Orice document are o structura de tip graf in care elementele sunt introduse in interiorul altor elemente (DOM)



```
<html>
<head>
<title>pagina mea</title>
</head>
<body>
  <h1>Compozitori:</h1>
  <p>
  <ul>
    <li> elvis costello
    <li> johannes brahms
    <li> georges brassens
  </ul>
</body>
</html>
```

## Relatia parinte copil si mostenire

- in CSS valoarea implicita a multor reguli este **inherit** mostenire de la parinte
- Exemplu:
  - daca pentru body se fixeaza regula: `body {color:#FF0000;}` textul va fi implicit rosu pentru toate elementele incluse (p,li,ul) daca:
    - nu apar reguli suplimentare (p {color:black;} )
    - nu exista reguli predefinite (h1 – regula browser-ului – h1 mai mare decat p, dar nu afecteaza culoarea)

```
<html>
<head>
<title>pagina mea</title>
</head>
<body>
  <h1>Compozitori:</h1>
  <p>
    <ul>
      <li> elvis costello
      <li> johannes brahms
      <li> georges brassens
    </ul>
  </p>
</body>
</html>
```



{CSS}



**Elemente frecvente**  
“I.Like.IT@ETTI.TUIASI.RO”

[www.tuiasi.ro](http://www.tuiasi.ro)

1813

1860

1910

1937

1993



## Controlul fundalului: culoare

- Pentru elementele cu structura bloc
- **background-color**=culoare
  - nume:
    - background-color= black;
  - Hexa - # Rosu (0-255=00-FF), Verde, Albastru
    - background-color =#FF0000;
  - RGB – rgb(rosu = 0-255,verde,albastru)
    - background-color:rgb(15,150,47);

## Controlul fundalului: imagine

- **background-image**
  - `body {background-image:url('bgdesert.jpg');}`
- **background-repeat**
  - `repeat-x; repeat-y; repeat; no-repeat; inherit;`
- **background-attachment**
  - `fixed; scroll;`
- **background-position**: “oriz.” “vert.”;
  - “oriz.”: `left; right; center; valoare[um]`
  - “vert.”: `top; bottom; center; valoare[um]`

## Controlul fundalului - scurt

- **background**
- Exemplu: `body {background:#ffffff url('img_bkg.png') no-repeat top right;}`
- Ordinea:
  - background-color
  - background-image
  - background-repeat
  - background-attachment
  - background-position

## Modificarea textului

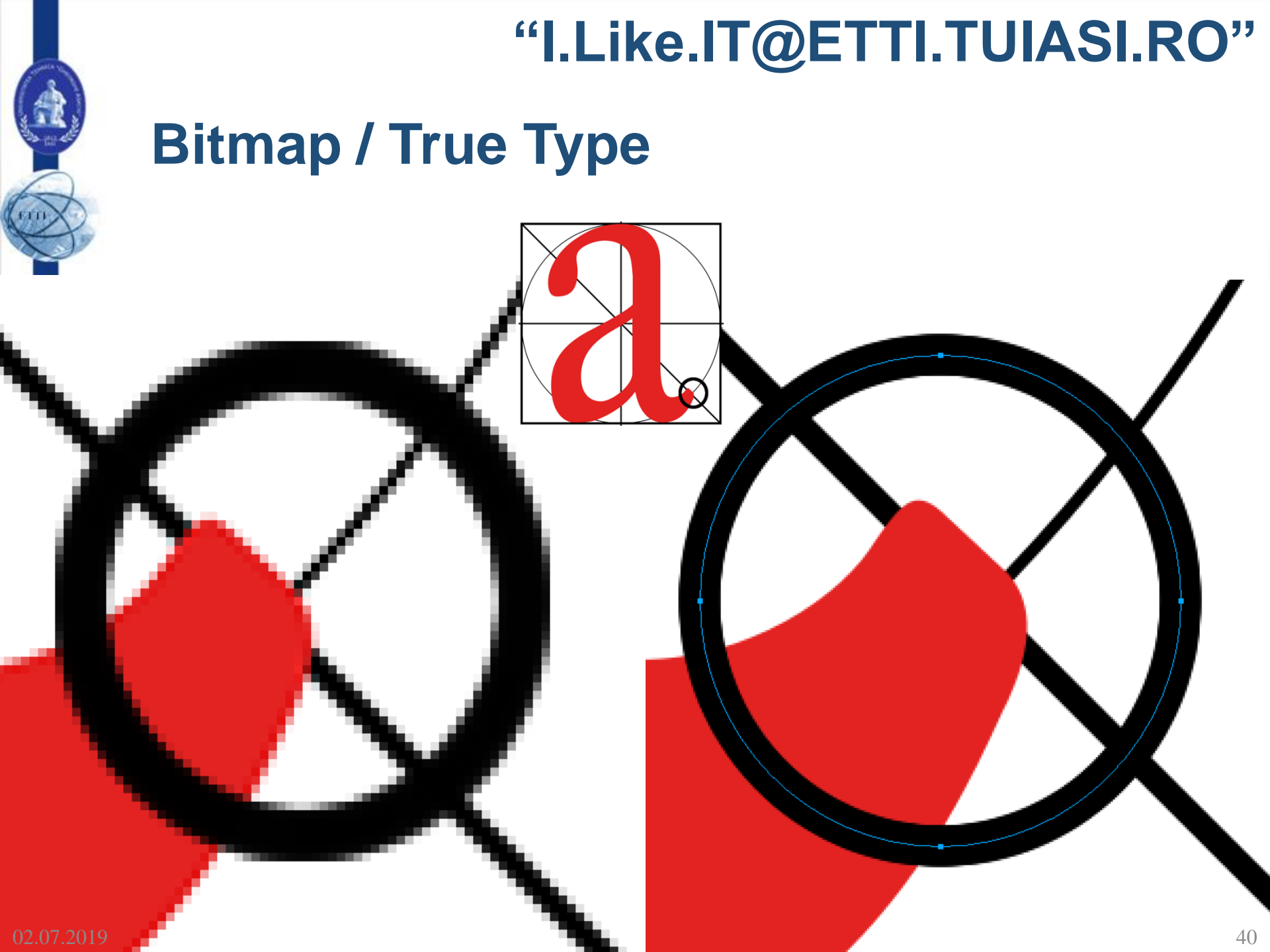
- culoare – **color**:[valoare];
- aliniere – **text-align**:left | center | right | justify;
- decorare – **text-decoration**:overline | line-through | underline | blink | none;
  - de obicei utilizat pentru eliminarea sublinierii la link-uri `<a style="text-decoration: none;">`
- transformare – **text-transform**:uppercase | lowercase | capitalize;
- indent – alinierea primului rand din paragraf  
**text-indent**:50px;



## Font

- Definitie: Font = desenul (forma grafica) a caracterului
- Fonturi
  - Outline font (True type) = desen vectorial, curbe Bezier - scalarea pastreaza calitatea maxima
  - Bitmap font = harta de pixeli – scalarea duce la aparitia artifactelor

## Bitmap / True Type



## Font

- desen de caractere

F

Sans-serif

F

Serif

Generic	Familie	Descriere
Serif	Times New Roman Georgia	Apar mici linii la terminatia anumitor caractere
Sans-serif	Arial Verdana	Fara liniute la sfarsit de caracter
Monospace	Courier New Lucida Console	Latime constanta a caracterelor

## Web safe font

- Fonturi definite de CSS
  - sans-serif
  - serif
  - monospace
  - cursive
  - fantasy
- Fonturi safe Microsoft
  - Arial
  - Courier New
  - Georgia
  - Times New Roman
  - Verdana
  - Trebuchet MS
  - Lucida Sans

sans-serif

serif

monospace

**cursive**

*fantasy*

Arial

Courier New

Georgia

Times New Roman

Verdana

Trebuchet MS

Lucida Sans



## Font

- desen de caracter – **font-family**: [lista];
  - font-family: Arial , "Times New Roman", sans-serif;
    - se utilizeaza Arial
    - daca Arial nu exista se utilizeaza Times New Roman
    - daca nici Times New Roman nu exista se utilizeaza sans-serif
    - daca nici sans-serif nu exista se utilizeaza fontul implicit in browser
- **font-style**: normal | italic | oblique;
- **font-size**: [dimensiune];
- **font-weight**: normal | bold | bolder | lighter | 100 – 900;
- **inherit** implicit pentru toate attributele;

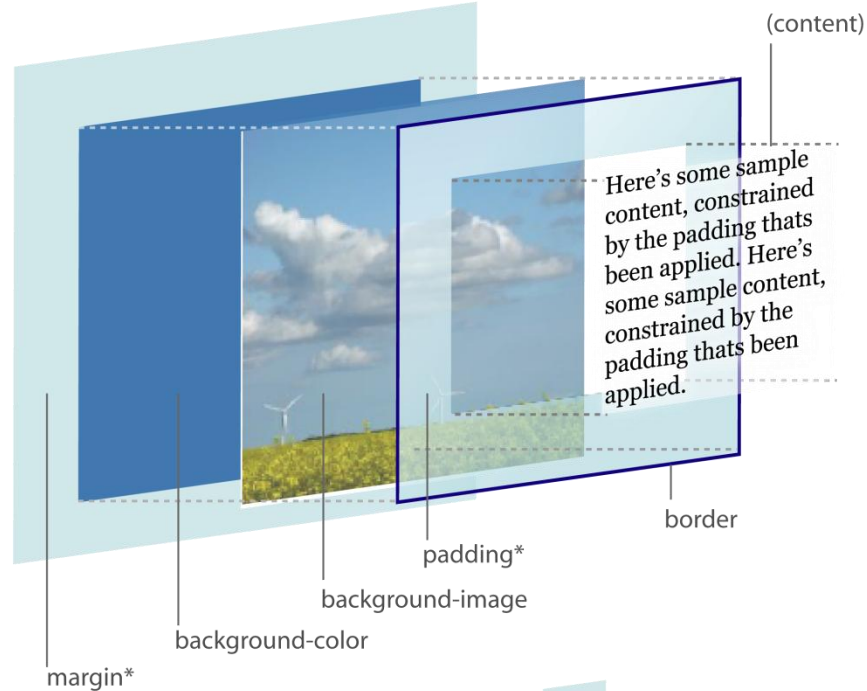
## Dimensiunile in CSS (text)

- relativ la dimensiunea implicita in browser
  - **xx-small, x-small, small, medium, large, x-large, xx-large** (implicit medium)
- relativ la dimensiunea din blocul parinte
  - **smaller, larger**
- absolute
  - in, cm, mm, pt (1 point = 1/72 in), pc (1 pica = 12 pt)
- relative
  - **em, ex – relative la dimensiunea in blocul parinte**
    - un cuvânt cu dimensiunea 2em = dimensiunea de 2 ori mai mare decat a textului din acel paragraf
    - precizie mai buna, valori fractionare permise (1.25em, 0.85em etc.)
  - px (pixeli) dependent de dispozitivul de afisare

## CSS Box Model

- Orientat in jurul conceptului de “cutie” – Box model

THE CSS BOX MODEL HIERARCHY



## CSS Box Model

- parametrii “box model” pot fi aplicati tuturor elementelor cu structura bloc, controland intregul bloc (margin, chenar, distanta intre chenar si continut).



## Margine

- spatiu liber lasat in exteriorul blocului
- **margin-top**: [valoare] | auto;
- **margin-right**: [valoare] | auto;
- **margin-bottom**: [valoare] | auto;
- **margin-left**: [valoare] | auto;
- scurt
  - **margin**: [top] [right] [bottom] [left];
  - **margin**: [top] [right= left] [bottom];
  - **margin**: [top=bottom] [right= left];
  - **margin**: [top=bottom=right= left];



## Chenar

- **border-style**: none | hidden | dotted | dashed | solid | double | groove | ridge | inset | outset | inherit; - deblocheaza celelalte proprietati
- **border-color**: culoare;
- **border-width**: thin | medium | thick | [valoare] | inherit;
- scurt: **border**: [border-width] [border-style] [border-color];
  - border: 5px solid red;
- detaliat:
  - **border-left-color**: ...
  - **border-top-width**: ...
  - etc.

## Distanța chenar-continut

- spațiu liber lasat în interiorul blocului
- **padding-top**: [valoare] | auto;
- **padding-right**: [valoare] | auto;
- **padding-bottom**: [valoare] | auto;
- **padding-left**: [valoare] | auto;
- scurt
  - **padding**: [top] [right] [bottom] [left];
  - **padding**: [top] [right= left] [bottom];
  - **padding**: [top=bottom] [right= left];
  - **padding**: [top=bottom=right= left];

## Dimensiuni, vizibilitate, afisare

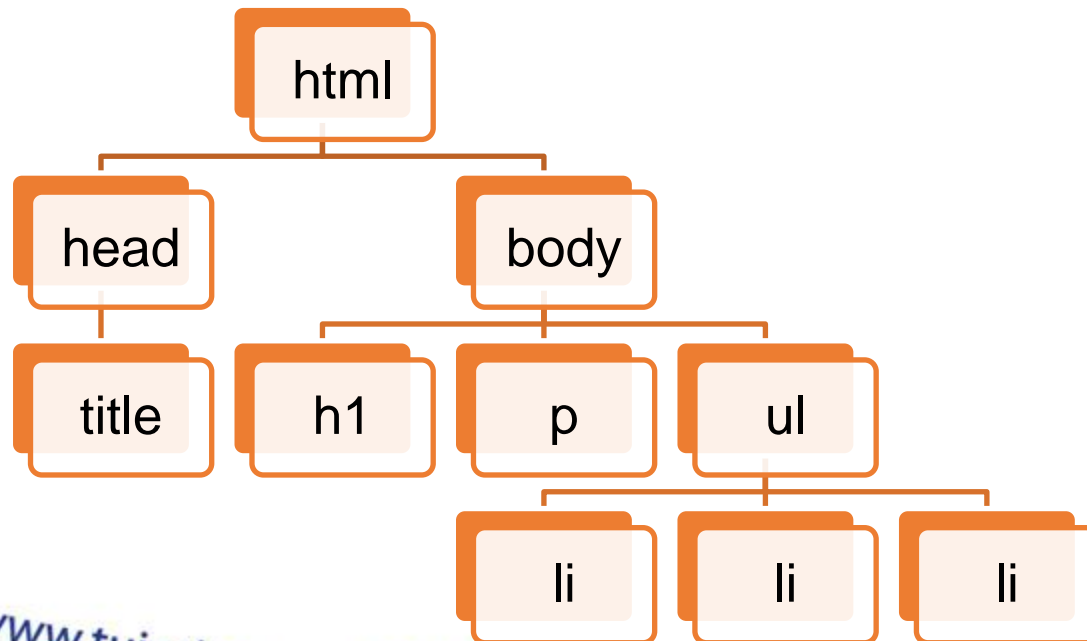
- **height**: auto | [valoare] | [%] | inherit
- **width**: auto | [valoare] | [%] | inherit
- CSS 2: **min-height**, **max-height**, **min-width**, **max-width**:none(max) | [valoare] | [%] | inherit
- **visibility**:visible | hidden;
- **display**:none | inline | block;

## Pozitionare

- **position**:absolute | fixed | relative | static | inherit;
- **top, right, bottom, left**:auto | [valoare] | [%] | inherit;
- **z-index**: [valoare intreaga]; - mai mare = deasupra

# DOM

- DOM – Document Object Model: structura de tip graf



```
<html>
<head>
<title>pagina mea</title>
</head>
<body>
  <h1>Compozitori:</h1>
  <p>
  <ul>
    <li> elvis costello
    <li> johannes brahms
    <li> georges brassens
  </ul>
</body>
</html>
```



## Relatie cu Javascript in aplicatii

- Javascript poate accesa prin intermediul obiectului DOM atasat documentului HTML elementele din structura arbore DOM si modifica proprietatile corespunzatoare
- se deschide astfel calea spre aplicatii dinamice



# Exercitii

“I.Like.IT@ETTI.TUIASI.RO”

[www.tuiasi.ro](http://www.tuiasi.ro)

1813

1860

1910

1937

1993

## Exercitiu 1

- Utilizati **Notepad ++** pentru a crea un document HTML cu text, link-uri imagini, sau reutilizati unul din fisierele din lectia precedenta
- Pentru un paragraf din document modificati forma (culoare sau dimensiune) prin **toate** cele trei modalitati de a specifica stiluri
  - stiluri externe (**External style sheet**)
    - va fi necesar sa creati un al doilea fisier in **Notepad ++** salvat cu extensia “\*.css”
  - stiluri interne (**Internal style sheet**)
  - stiluri inserate (**Inline style**)
- Vizualizati pagina creata in browser (Edge/Internet Explorer/ Chrome)

## Exercitiu 2

- Se lucreaza cu stiluri externe (**External style sheet**)
  - e necesar sa creati un al doilea fisier in **Notepad ++** salvat cu extensia “\*.css” sau sa refolositi fisierul realizat anterior
- Utilizati **Notepad ++** pentru a modifica documentele **HTML si CSS** pentru a exersa cu diversele tipuri de specificare a selectorilor
  - eticheta HTML
  - clasa
  - identificator
  - eticheta HTML in interiorul unei clase
  - eticheta HTML cu o anumita clasa
  - etc
- Vizualizati pagina creata in browser (Edge/Internet Explorer/ Chrome)

## Exercitiu 3

- Se lucreaza cu stiluri externe (**External style sheet**)
  - e necesar sa creati un al doilea fisier in **Notepad ++** salvat cu extensia “\*.css” sau sa refolositi fisierul realizat anterior
- Utilizati **Notepad ++** pentru a modifica documentele **HTML si CSS** pentru a exersa cu modificarea textului
  - dimensiune
  - culoare
  - font
  - forma (caractere mari/mici, subliniere, grosime)
  - etc
- Vizualizati pagina creata in browser (Edge/Internet Explorer/ Chrome)



## Exercitiu 4

- Se lucreaza cu stiluri externe (**External style sheet**)
  - e necesar sa creati un al doilea fisier in **Notepad ++** salvat cu extensia “\*.css” sau sa refolositi fisierul realizat anterior
- Utilizati **Notepad ++** pentru a modifica documentele **HTML si CSS** pentru a exersa controlul unui element bloc (CSS Box Model)
  - margine
  - background culoare/pozitie
  - padding
  - border
  - etc
- Vizualizati pagina creata in browser (Edge/Internet Explorer/ Chrome)

## Exercitiu 5

- Se lucreaza cu stiluri externe (**External style sheet**)
  - e necesar sa creati un al doilea fisier in **Notepad ++** salvat cu extensia “\*.css” sau sa refolositi fisierul realizat anterior
- Utilizati **Notepad ++** pentru a modifica documentele **HTML si CSS** pentru a exersa pozitionarea unui element bloc (CSS Box Model) fata de alte elemente de tip bloc (se utilizeaza, introdus fortat daca e nevoie **<div>...</div>**)
  - margine
  - pozitionare
  - dimensiune
  - suprapuneri
  - etc
- Vizualizati pagina creata in browser (Edge/Internet Explorer/ Chrome)



# Exemple CSS

## "I.Like.IT@ETTI.TUIASI.RO"

## CSS Zen Garden

- <http://www.csszengarden.com/>
- un fisier html **comun**
- schimbarea formei permisa numai prin intermediul CSS

## CSS Zen Garden – HTML original

### css Zen Garden

#### The Beauty of CSS Design

A demonstration of what can be accomplished visually through CSS-based design. Select any style sheet from the list to load it into this page.

Download the sample [html file](#) and [css file](#)

#### The Road to Enlightenment

Littering a dark and dreary road lay the past relics of browser-specific tags, incompatible DOMs, and broken CSS support.

Today, we must clear the mind of past practices. Web enlightenment has been achieved thanks to the tireless efforts of folk like the W3C, WaSP and the major browser creators.

The css Zen Garden invites you to relax and meditate on the important lessons of the masters. Begin to see with clarity. Learn to use the (yet to be) time-honored techniques in new and invigorating fashion. Become one with the web.

#### So What is This About?

There is clearly a need for CSS to be taken seriously by graphic artists. The Zen Garden aims to excite, inspire, and encourage participation. To begin, view some of the existing designs in the list. Clicking on any one will load the style sheet into this very page. The code remains the same, the only thing that has changed is the external .css file. Yes, really.

CSS allows complete and total control over the style of a hypertext document. The only way this can be illustrated in a way that gets people excited is by demonstrating what it can truly be, once the reins are placed in the hands of those able to create beauty from structure. To date, most examples of neat tricks and hacks have been demonstrated by structurists and coders. Designers have yet to make their mark. This needs to change.

#### Participation

Graphic artists only please. You are modifying this page, so strong CSS skills are necessary, but the example files are commented well enough that even CSS novices can use them as starting points. Please see the [CSS Resource Guide](#) for advanced tutorials and tips on working with CSS.

You may modify the style sheet in any way you wish, but not the HTML. This may seem daunting at first if you've never worked this way before, but follow the listed links to learn more, and use the sample files as a guide.

Download the sample [html file](#) and [css file](#) to work on a copy locally. Once you have completed your masterpiece (and please, don't submit half-finished work) upload your .css file to a web server under your control. [Send us a link](#) to the file and if we choose to use it, we will spider the associated images. Final submissions will be placed on our server.

#### Benefits

Why participate? For recognition, inspiration, and a resource we can all refer to when making the case for CSS-based design. This is sorely needed, even today. More and more major sites are taking the leap, but not enough have. One day this gallery will be a historical curiosity; that day is not today.

#### Requirements

We would like to see as much CSS1 as possible. CSS2 should be limited to widely-supported elements only. The css Zen Garden is about functional, practical CSS and not the latest bleeding-edge tricks viewable by 2% of the browsing public. The only real requirement we have is that your CSS validates.



# CSS Zen Garden – HTML original

## css Zen Garden

### The Beauty of CSS Design

A demonstration of what can be accomplished visually through CSS-based design. Select any style sheet from the

Download the sample [html file](#) and [css file](#)

### The Road to Enlightenment

Littering a dark and dreary road lay the past relics of browser-specific tags, incompatible DOMs, and broken CSS

Today, we must clear the mind of past practices. Web enlightenment has been achieved thanks to the tireless effort

The css Zen Garden invites you to relax and meditate on the important lessons of the masters. Begin to see with the eye. Become one with the web.

### So What is This About?

There is clearly a need for CSS to be taken seriously by graphic artists. The Zen Garden aims to excite, inspire, and educate. When you load the style sheet into this very page. The code remains the same, the only thing that has changed is the design. **1993**

CSS allows complete and total control over the style of a hypertext document. The only way this can be illustrated is by placing it in the hands of those able to create beauty from structure. To date, most examples of neat tricks and hacks



## CSS Zen Garden



*Css Zen Garden*  
the beauty of css design.

A demonstration of what can be accomplished visually through CSS-based design. Select any style sheet from the list to load it into this page.

[DOWNLOAD THE SAMPLE HTML FILE AND CSS FILE](#)

### *The Road To Enlightenment*

Littering a dark and dreary road lay the past relics of browser-specific tags, incompatible DOMs, and broken CSS support.

Today, we must clear the mind of past practices. Web enlightenment has been achieved thanks to the tireless efforts of folk like the W3C, WaSP and the major browser creators.

The css Zen Garden invites you to relax and meditate on the important lessons of the masters. Begin to see with clarity. Learn to use the (yet to be) time-honored techniques in new and invigorating fashion. Become one with the web.

### *So What is This About*

There is clearly a need for CSS to be taken seriously by graphic artists. The Zen Garden aims to excite, inspire, and encourage participation. To begin, view some of the existing designs in the list. Clicking on any one will load the style sheet into this very page. The code remains the same, the only thing that has changed is the external .css file. Yes, really.

CSS allows complete and total control over the style of a hypertext document. The only way this can be illustrated in a way that gets people excited is by demonstrating what it can truly be, once the reins are placed in the hands of those able to create beauty from structure. To date, most examples of neat tricks and hacks have been

#### SELECT A DESIGN:

- ☐ [Under the Seal](#)  
by Eric Stoltz
- ☐ [Make 'em Proud](#)  
by Michael McAdghon and  
Scotty Rief snyder
- ☐ [Orchid Beauty](#)  
by Kevin Addison
- ☐ [Oceanscape](#)  
by Justin Gray
- ☐ [CSS Co., Ltd.](#)  
by Benjamin Kramm
- ☐ [Sakura](#)  
by Tatsuya Uchida
- ☐ [Kyoto Forest](#)  
by John Polkowski
- ☐ [A Walk in the Garden](#)  
by Simon Van  
Hauwermeiren

#### ARCHIVES:

[next designs »](#)

[View All Designs](#)

#### RESOURCES:

## CSS Zen Garden



Download the sample [html file](#) and [css file](#)

### CSS Zen Garden

A demonstration of what can be accomplished visually through CSS-based design. Select any stylesheet from the list to load it into this page.



#### Select a Design

- [Under the Seal](#)  
by [Eric Stoltz](#)
- [Make 'em Proud](#)  
by [Michael McAnthon](#) and [Scotty Reifsnnyder](#)
- [Orchid Beauty](#)  
by [Kevin Addison](#)
- [Oceanscape](#)  
by [Justin Gray](#)
- [CSS Co., Ltd.](#)  
by [Benjamin Klemm](#)
- [Sakura](#)  
by [Tatsuya Uchida](#)
- [Kyoto Forest](#)  
by [John Polittowski](#)
- [A Walk in the Garden](#)  
by [Simon Van Hauwermeiren](#)

#### Archives

- [next designs »](#)
- [View All Designs](#)

#### Resources

#### The Road to Enlightenment

Littering a dark and dreary road lay the past relics of browser-specific tags, incompatible [DOMs](#), and broken [CSS](#) support. Today, we must clear the mind of past practices. Web enlightenment has been achieved thanks to the tireless efforts of folk like the [W3C](#), [WaSP](#) and the major browser creators. The [css Zen Garden](#) invites you to relax and meditate on the important lessons of the masters. Begin to see with clarity. Learn to use the (yet to be) time-honored techniques in new and invigorating fashion. Become one with the web.

#### So What is This About



```
h2 { color: #0000FF; font-weight: bold; font-size: 1.2em; }
h3 { color: #0000FF; font-weight: bold; font-size: 1.1em; }
h4 { color: #0000FF; font-weight: bold; font-size: 1.0em; }
```

There is clearly a need for [CSS](#) to be taken seriously by graphic artists. The Zen Garden aims to excite, inspire, and encourage participation. To begin, view some of the existing designs in the list. Clicking on any one will load the style sheet into this very page. The code remains the same, the only thing that has changed is the external [css](#) file. Yes, really. [CSS](#) allows complete and total control over the style of a hypertext document.

#### Participation



Graphic artists only please. You are modifying this page, so strong [CSS](#) skills are necessary, but the example files are commented well enough that even [CSS](#) novices can use them as starting points. Please see the [CSS Resource Guide](#) for advanced tutorials and tips on working with [CSS](#). You may modify the style sheet in any way you wish, but not the [HTML](#). This may seem daunting at first if



## CSS Zen Garden



*“ A demonstration of what can be accomplished visually through CSS-based design. Select any style sheet from the list to load it into this page. ”*

### The Road to Enlightenment

Littering a dark and dreary road lay the past relics of browser-specific tags, incompatible DOMs, and broken CSS support.

Today, we must clear the mind of past practices. Web enlightenment has been achieved thanks to the tireless efforts of folk like the W3C, WaSP and the major browser creators.

The css Zen Garden invites you to relax and meditate on the important lessons of the masters. Begin to see with clarity. Learn to use the (yet to be) time-honored techniques in new and invigorating fashion. Become one with the web.

### So What is This About?

There is clearly a need for CSS to be taken seriously by graphic artists. The Zen Garden aims to excite, inspire, and encourage participation. To begin, view some of the existing designs in the list. Clicking on any one will load the style sheet into this very page. The code remains the same, the only thing that has changed is the external .css file. Yes, really.

CSS allows complete and total control over the style of a hypertext document. The only way this can be illustrated in a way that gets people excited is by demonstrating what it can truly be, once the reins are placed in the hands of those able to create beauty from structure. To date, most examples of neat tricks and hacks have been demonstrated by structurists and coders. Designers have yet to make their mark. This needs to change.

### Participation

Graphic artists only please. You are modifying this page, so strong CSS skills are necessary, but the example files are commented well enough that even CSS novices can use them as starting points. Please see the [CSS Resource Guide](#) for advanced tutorials and tips on working with CSS.

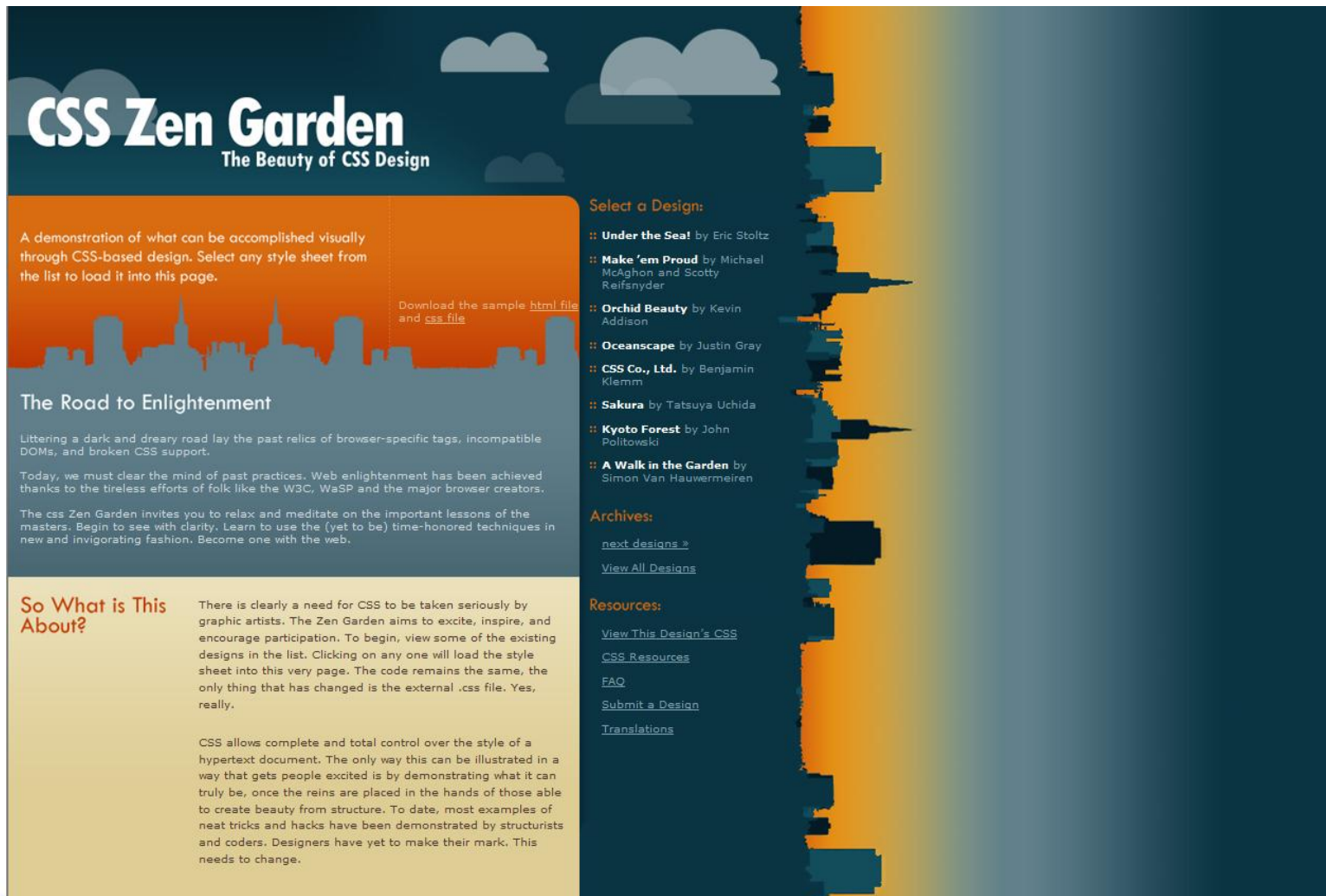
You may modify the style sheet in any way you wish, but not the HTML. This may seem daunting at first if you've never worked this way before, but follow the listed links to learn more, and use the sample files as a

**Select a design**

- ✦ Under the Seal by Eric Stoltz
- ✦ Make 'em Proud by Michael McAghon and Scotty Reifsnyder
- ✦ Orchid Beauty by Kevin Addison
- ✦ Oceanscape by Justin Gray
- ✦ CSS Co., Ltd. by Benjamin Klemm
- ✦ Sakura by Tatsuya Uchida
- ✦ Kyoto Forest by John Poltowski
- ✦ A Walk in the Garden by Simon Van Hauwermeiren

**Archives**

## CSS Zen Garden



**CSS Zen Garden**  
The Beauty of CSS Design

A demonstration of what can be accomplished visually through CSS-based design. Select any style sheet from the list to load it into this page.

Download the sample [html file](#) and [css file](#)

### The Road to Enlightenment

Littering a dark and dreary road lay the past relics of browser-specific tags, incompatible DOMs, and broken CSS support.

Today, we must clear the mind of past practices. Web enlightenment has been achieved thanks to the tireless efforts of folk like the W3C, WaSP and the major browser creators.

The css Zen Garden invites you to relax and meditate on the important lessons of the masters. Begin to see with clarity. Learn to use the (yet to be) time-honored techniques in new and invigorating fashion. Become one with the web.

### So What is This About?

There is clearly a need for CSS to be taken seriously by graphic artists. The Zen Garden aims to excite, inspire, and encourage participation. To begin, view some of the existing designs in the list. Clicking on any one will load the style sheet into this very page. The code remains the same, the only thing that has changed is the external .css file. Yes, really.

CSS allows complete and total control over the style of a hypertext document. The only way this can be illustrated in a way that gets people excited is by demonstrating what it can truly be, once the reins are placed in the hands of those able to create beauty from structure. To date, most examples of neat tricks and hacks have been demonstrated by structurists and coders. Designers have yet to make their mark. This needs to change.

### Select a Design:

- Under the Sea!** by Eric Stoltz
- Make 'em Proud** by Michael McAgnon and Scotty Reifsnnyder
- Orchid Beauty** by Kevin Addison
- Oceanscape** by Justin Gray
- CSS Co., Ltd.** by Benjamin Klemm
- Sakura** by Tatsuya Uchida
- Kyoto Forest** by John Politowski
- A Walk in the Garden** by Simon Van Hauwermeiren

### Archives:

- [next designs >](#)
- [View All Designs](#)

### Resources:

- [View This Design's CSS](#)
- [CSS Resources](#)
- [FAQ](#)
- [Submit a Design](#)
- [Translations](#)



## CSS Zen Garden

### The Beauty of CSS Design

A demonstration of what can be accomplished through CSS-based design. Select any style sheet from the list to load it into this page.

Download the example [html file](#) and [css file](#)



### The Road to Enlightenment

#### So What is This About?

There is a continuing need to show the power of CSS. The Zen Garden aims to excite, inspire, and encourage participation. To begin, view some of the existing designs in the list. Clicking on any one will load the style sheet into this very page. The HTML remains the same, the only thing that has changed is the external CSS file. Yes, really.

CSS allows complete and total control over the style of a hypertext document. The only way this can be illustrated in a way that gets people excited is by demonstrating what it can truly be, once the reins are placed in the hands of those able to create beauty from structure. Designers and coders alike have contributed to the beauty of the web; we can always push it further.

HTML  
CSS  
CC  
A11y  
GH

#### Participation

Strong visual design has always been our focus. You are modifying this page, so strong CSS skills are necessary too, but the example files are commented well enough that even CSS novices can use them as starting points. Please see the [CSS Resource Guide](#) for advanced tutorials and tips on working with CSS.

You may modify the style sheet in any way you wish, but not the HTML. This may seem daunting at first if

[www.tuiasi.ro](http://www.tuiasi.ro)

1813



## CSS Zen Garden



### SELECT A DESIGN:

Mid Century Modern by **Andrew**  
Garments by **Dan I**  
Steel by **Steffen Kr**  
Apothecary by **Tren**  
Screen Filler by **Elliot Jay Stock**  
Fountain Kiss by **Jc**  
A Robot Named Jimmy by **melt**  
Verde Moderna by **Dave Si**

### ARCHIVES:

Next Designs >  
View All Designs

### RESOURCES:

View This Design's CSS

CSS Resources

FAQ

Submit a Design

Translations

### WHAT IS THIS ABOUT?

A demonstration of what can be accomplished through CSS-based design. Select any style sheet from the list to load it into this page.

Download the example [html file](#) and [css file](#)

to have total control over the document. The only way this is possible is by using CSS, which is what gets people excited about it. It can truly be, once the hands of those able to create it are put to the beauty of the web; it is fun here.



# CSS Zen Garden

CSS ZEN GARDEN

## The Beauty of CSS Design

A DEMONSTRATION OF WHAT CAN BE ACCOMPLISHED THROUGH CSS-BASED DESIGN. SELECT ANY STYLE SHEET FROM THE LIST TO LOAD IT INTO THIS PAGE.

DOWNLOAD THE EXAMPLE [HTML FILE](#) AND [CSS FILE](#)

### Select a Design:

[Mid Century Modern](#) by [Andrew Lohman](#)

[Garments](#) by [Dan Mall](#)

[Steel](#) by [Steffen Knoeller](#)

[Apothecary](#) by [Trent Walton](#)

[Screen Filler](#) by [Elliot Jay Stocks](#)

[Fountain Kiss](#) by [Jeremy Carlson](#)

[A Robot Named Jimmy](#) by [meltmedia](#)

[Verde Moderna](#) by [Dave Shea](#)

### Archives:

[Next Designs](#) ›

[View All Designs](#)



# Exercitii

“I.Like.IT@ETTI.TUIASI.RO”

[www.tuiasi.ro](http://www.tuiasi.ro)

1813

1860

1910

1937

1993

## Exercitiu 6

- Obtineti (download) de la adresa [CSS Zen Garden](#) fisierele **HTML si CSS**
- Utilizati **Notepad ++** pentru a modifica **numai** documentul **CSS** si exersati cu diverse proprietati CSS
  - exercitiu artistic !
- Vizualizati pagina creata in browser (Edge/Internet Explorer/ Chrome)



# Bibliografie CSS

`"I.Like.IT@ETTI.TUIASI.RO"`

[www.tuiasi.ro](http://www.tuiasi.ro)

1813

1860

1910

1937

1993

## Bibliografie CSS

- World Wide Web Consortium (W3C), CSS Standards
  - <https://www.w3.org/Style/CSS/>
- W3Schools CSS Reference
  - <https://www.w3schools.com/cssref/>





# Proprietati CSS

`"I.Like.IT@ETTI.TUIASI.RO"`

[www.tuiasi.ro](http://www.tuiasi.ro)

1813

1860

1910

1937

1993

## Reguli CSS - A

<b>align-content</b>	Specifies the alignment between the lines inside a flexible container when the items do not use all available space
<b>align-items</b>	Specifies the alignment for items inside a flexible container
<b>align-self</b>	Specifies the alignment for selected items inside a flexible container
<b>all</b>	Resets all properties (except unicode-bidi and direction)
<b>animation</b>	A shorthand property for all the <i>animation</i> -* properties
<b>animation-delay</b>	Specifies a delay for the start of an animation
<b>animation-direction</b>	Specifies whether an animation should be played forwards, backwards or in alternate cycles
<b>animation-duration</b>	Specifies how long an animation should take to complete one cycle
<b>animation-fill-mode</b>	Specifies a style for the element when the animation is not playing (before it starts, after it ends, or both)
<b>animation-iteration-count</b>	Specifies the number of times an animation should be played
<b>animation-name</b>	Specifies a name for the @keyframes animation
<b>animation-play-state</b>	Specifies whether the animation is running or paused
<b>animation-timing-function</b>	Specifies the speed curve of an animation

## Reguli CSS - B

<b>backface-visibility</b>	Defines whether or not the back face of an element should be visible when facing the user
<b>background</b>	A shorthand property for all the <i>background</i> -* properties
<b>background-attachment</b>	Sets whether a background image scrolls with the rest of the page, or is fixed
<b>background-blend-mode</b>	Specifies the blending mode of each background layer (color/image)
<b>background-clip</b>	Defines how far the background (color or image) should extend within an element
<b>background-color</b>	Specifies the background color of an element
<b>background-image</b>	Specifies one or more background images for an element
<b>background-origin</b>	Specifies the origin position of a background image
<b>background-position</b>	Specifies the position of a background image
<b>background-repeat</b>	Sets if/how a background image will be repeated
<b>background-size</b>	Specifies the size of the background images
<b>border</b>	A shorthand property for <i>border-width</i> , <i>border-style</i> and <i>border-color</i>
<b>border-bottom</b>	A shorthand property for <i>border-bottom-width</i> , <i>border-bottom-style</i> and <i>border-bottom-color</i>
<b>border-bottom-color</b>	Sets the color of the bottom border
<b>border-bottom-left-radius</b>	Defines the radius of the border of the bottom-left corner
<b>border-bottom-right-radius</b>	Defines the radius of the border of the bottom-right corner
<b>border-bottom-style</b>	Sets the style of the bottom border
<b>border-bottom-width</b>	Sets the width of the bottom border

## Reguli CSS - B

<b>border-collapse</b>	Sets whether table borders should collapse into a single border or be separated
<b>border-color</b>	Sets the color of the four borders
<b>border-image</b>	A shorthand property for all the <i>border-image</i> -* properties
<b>border-image-outset</b>	Specifies the amount by which the border image area extends beyond the border box
<b>border-image-repeat</b>	Specifies whether the border image should be repeated, rounded or stretched
<b>border-image-slice</b>	Specifies how to slice the border image
<b>border-image-source</b>	Specifies the path to the image to be used as a border
<b>border-image-width</b>	Specifies the width of the border image
<b>border-left</b>	A shorthand property for all the <i>border-left</i> -* properties
<b>border-left-color</b>	Sets the color of the left border
<b>border-left-style</b>	Sets the style of the left border
<b>border-left-width</b>	Sets the width of the left border
<b>border-radius</b>	A shorthand property for the four <i>border</i> -*- <i>radius</i> properties
<b>border-right</b>	A shorthand property for all the <i>border-right</i> -* properties
<b>border-right-color</b>	Sets the color of the right border
<b>border-right-style</b>	Sets the style of the right border
<b>border-right-width</b>	Sets the width of the right border
<b>border-spacing</b>	Sets the distance between the borders of adjacent cells

## Reguli CSS - B

<b>border-style</b>	Sets the style of the four borders
<b>border-top</b>	A shorthand property for <i>border-top-width</i> , <i>border-top-style</i> and <i>border-top-color</i>
<b>border-top-color</b>	Sets the color of the top border
<b>border-top-left-radius</b>	Defines the radius of the border of the top-left corner
<b>border-top-right-radius</b>	Defines the radius of the border of the top-right corner
<b>border-top-style</b>	Sets the style of the top border
<b>border-top-width</b>	Sets the width of the top border
<b>border-width</b>	Sets the width of the four borders
<b>bottom</b>	Sets the elements position, from the bottom of its parent element
<b>box-decoration-break</b>	Sets the behavior of the background and border of an element at page-break, or, for in-line elements, at line-break.
<b>box-shadow</b>	Attaches one or more shadows to an element
<b>box-sizing</b>	Defines how the width and height of an element are calculated: should they include padding and borders, or not
<b>break-after</b>	Specifies the page-, column-, or region-break behavior after the generated box
<b>break-before</b>	Specifies the page-, column-, or region-break behavior before the generated box
<b>break-inside</b>	Specifies the page-, column-, or region-break behavior inside the generated box

## Reguli CSS - C

<b>caption-side</b>	Specifies the placement of a table caption
<b>caret-color</b>	Specifies the color of the cursor (caret) in inputs, textareas, or any element that is editable
<b>@charset</b>	Specifies the character encoding used in the style sheet
<b>clear</b>	Specifies on which sides of an element floating elements are not allowed to float
<b>clip</b>	Clips an absolutely positioned element
<b>color</b>	Sets the color of text
<b>column-count</b>	Specifies the number of columns an element should be divided into
<b>column-fill</b>	Specifies how to fill columns, balanced or not
<b>column-gap</b>	Specifies the gap between the columns
<b>column-rule</b>	A shorthand property for all the <i>column-rule</i> -* properties
<b>column-rule-color</b>	Specifies the color of the rule between columns
<b>column-rule-style</b>	Specifies the style of the rule between columns
<b>column-rule-width</b>	Specifies the width of the rule between columns
<b>column-span</b>	Specifies how many columns an element should span across
<b>column-width</b>	Specifies the column width
<b>columns</b>	A shorthand property for <i>column-width</i> and <i>column-count</i>
<b>content</b>	Used with the :before and :after pseudo-elements, to insert generated content
<b>counter-increment</b>	Increases or decreases the value of one or more CSS counters
<b>counter-reset</b>	Creates or resets one or more CSS counters
<b>cursor</b>	Specifies the mouse cursor to be displayed when pointing over an element



## Reguli CSS – D, E

<b>direction</b>	Specifies the text direction/writing direction
<b>display</b>	Specifies how a certain HTML element should be displayed
<b>empty-cells</b>	Specifies whether or not to display borders and background on empty cells in a table

## Reguli CSS – F

<b>filter</b>	Defines effects (e.g. blurring or color shifting) on an element before the element is displayed
<b>flex</b>	A shorthand property for the <i>flex-grow</i> , <i>flex-shrink</i> , and the <i>flex-basis</i> properties
<b>flex-basis</b>	Specifies the initial length of a flexible item
<b>flex-direction</b>	Specifies the direction of the flexible items
<b>flex-flow</b>	A shorthand property for the <i>flex-direction</i> and the <i>flex-wrap</i> properties
<b>flex-grow</b>	Specifies how much the item will grow relative to the rest
<b>flex-shrink</b>	Specifies how the item will shrink relative to the rest
<b>flex-wrap</b>	Specifies whether the flexible items should wrap or not
<b>float</b>	Specifies whether or not a box should float
<b>font</b>	A shorthand property for the <i>font-style</i> , <i>font-variant</i> , <i>font-weight</i> , <i>font-size/line-height</i> , and the <i>font-family</i> properties
<b>@font-face</b>	A rule that allows websites to download and use fonts other than the "web-safe" fonts
<b>font-family</b>	Specifies the font family for text
<b>font-feature-settings</b>	Allows control over advanced typographic features in OpenType fonts
<b>@font-feature-values</b>	Allows authors to use a common name in font-variant-alternate for feature activated differently in OpenType
<b>font-kerning</b>	Controls the usage of the kerning information (how letters are spaced)
<b>font-language-override</b>	Controls the usage of language-specific glyphs in a typeface

## Reguli CSS – F

<b>font-size</b>	Specifies the font size of text
<b>font-size-adjust</b>	Preserves the readability of text when font fallback occurs
<b>font-stretch</b>	Selects a normal, condensed, or expanded face from a font family
<b>font-style</b>	Specifies the font style for text
<b>font-synthesis</b>	Controls which missing typefaces (bold or italic) may be synthesized by the browser
<b>font-variant</b>	Specifies whether or not a text should be displayed in a small-caps font
<b>font-variant-alternates</b>	Controls the usage of alternate glyphs associated to alternative names defined in @font-feature-values
<b>font-variant-caps</b>	Controls the usage of alternate glyphs for capital letters
<b>font-variant-east-asian</b>	Controls the usage of alternate glyphs for East Asian scripts (e.g Japanese and Chinese)
<b>font-variant-ligatures</b>	Controls which ligatures and contextual forms are used in textual content of the elements it applies to
<b>font-variant-numeric</b>	Controls the usage of alternate glyphs for numbers, fractions, and ordinal markers
<b>font-variant-position</b>	Controls the usage of alternate glyphs of smaller size positioned as superscript or subscript regarding the baseline of the font
<b>font-weight</b>	Specifies the weight of a font

## Reguli CSS – G

<b>grid</b>	A shorthand property for the <i>grid-template-rows</i> , <i>grid-template-columns</i> , <i>grid-template-areas</i> , <i>grid-auto-rows</i> , <i>grid-auto-columns</i> , and the <i>grid-auto-flow</i> properties
<b>grid-area</b>	Either specifies a name for the grid item, or this property is a shorthand property for the <i>grid-row-start</i> , <i>grid-column-start</i> , <i>grid-row-end</i> , and <i>grid-column-end</i> properties
<b>grid-auto-columns</b>	Specifies a default column size
<b>grid-auto-flow</b>	Specifies how auto-placed items are inserted in the grid
<b>grid-auto-rows</b>	Specifies a default row size
<b>grid-column</b>	A shorthand property for the <i>grid-column-start</i> and the <i>grid-column-end</i> properties
<b>grid-column-end</b>	Specifies where to end the grid item
<b>grid-column-gap</b>	Specifies the size of the gap between columns
<b>grid-column-start</b>	Specifies where to start the grid item
<b>grid-gap</b>	A shorthand property for the <i>grid-row-gap</i> and <i>grid-column-gap</i> properties
<b>grid-row</b>	A shorthand property for the <i>grid-row-start</i> and the <i>grid-row-end</i> properties
<b>grid-row-end</b>	Specifies where to end the grid item
<b>grid-row-gap</b>	Specifies the size of the gap between rows
<b>grid-row-start</b>	Specifies where to start the grid item
<b>grid-template</b>	A shorthand property for the <i>grid-template-rows</i> , <i>grid-template-columns</i> and <i>grid-areas</i> properties
<b>grid-template-areas</b>	Specifies how to display columns and rows, using named grid items
<b>grid-template-columns</b>	Specifies the size of the columns, and how many columns in a grid layout
<b>grid-template-rows</b>	Specifies the size of the rows in a grid layout

## Reguli CSS – H, I, J, K, L

<b>hanging-punctuation</b>	Specifies whether a punctuation character may be placed outside the line box
<b>height</b>	Sets the height of an element
<b>hyphens</b>	Sets how to split words to improve the layout of paragraphs
<b>image-rendering</b>	Gives a hint to the browser about what aspects of an image are most important to preserve when the image is scaled
<b>@import</b>	Allows you to import a style sheet into another style sheet
<b>isolation</b>	Defines whether an element must create a new stacking content
<b>justify-content</b>	Specifies the alignment between the items inside a flexible container when the items do not use all available space
<b>@keyframes</b>	Specifies the animation code
<b>left</b>	Specifies the left position of a positioned element
<b>letter-spacing</b>	Increases or decreases the space between characters in a text
<b>line-break</b>	Specifies how/if to break lines
<b>line-height</b>	Sets the line height
<b>list-style</b>	Sets all the properties for a list in one declaration
<b>list-style-image</b>	Specifies an image as the list-item marker
<b>list-style-position</b>	Specifies the position of the list-item markers (bullet points)
<b>list-style-type</b>	Specifies the type of list-item marker

## Reguli CSS – M

<b>margin</b>	Sets all the margin properties in one declaration
<b>margin-bottom</b>	Sets the bottom margin of an element
<b>margin-left</b>	Sets the left margin of an element
<b>margin-right</b>	Sets the right margin of an element
<b>margin-top</b>	Sets the top margin of an element
<b>max-height</b>	Sets the maximum height of an element
<b>max-width</b>	Sets the maximum width of an element
<b>@media</b>	Sets the style rules for different media types/devices/sizes
<b>min-height</b>	Sets the minimum height of an element
<b>min-width</b>	Sets the minimum width of an element
<b>mix-blend-mode</b>	Specifies how an element's content should blend with its direct parent background



## Reguli CSS – O

<b>object-fit</b>	Specifies how the contents of a replaced element should be fitted to the box established by its used height and width
<b>object-position</b>	Specifies the alignment of the replaced element inside its box
<b>opacity</b>	Sets the opacity level for an element
<b>order</b>	Sets the order of the flexible item, relative to the rest
<b>orphans</b>	Sets the minimum number of lines that must be left at the bottom of a page when a page break occurs inside an element
<b>outline</b>	A shorthand property for the <i>outline-width</i> , <i>outline-style</i> , and the <i>outline-color</i> properties
<b>outline-color</b>	Sets the color of an outline
<b>outline-offset</b>	Offsets an outline, and draws it beyond the border edge
<b>outline-style</b>	Sets the style of an outline
<b>outline-width</b>	Sets the width of an outline
<b>overflow</b>	Specifies what happens if content overflows an element's box
<b>overflow-wrap</b>	Specifies whether or not the browser may break lines within words in order to prevent overflow (when a string is too long to fit its containing box)
<b>overflow-x</b>	Specifies whether or not to clip the left/right edges of the content, if it overflows the element's content area
<b>overflow-y</b>	Specifies whether or not to clip the top/bottom edges of the content, if it overflows the element's content area

## Reguli CSS – P

<b>padding</b>	A shorthand property for all the <i>padding</i> -* properties
<b>padding-bottom</b>	Sets the bottom padding of an element
<b>padding-left</b>	Sets the left padding of an element
<b>padding-right</b>	Sets the right padding of an element
<b>padding-top</b>	Sets the top padding of an element
<b>page-break-after</b>	Sets the page-break behavior after an element
<b>page-break-before</b>	Sets the page-break behavior before an element
<b>page-break-inside</b>	Sets the page-break behavior inside an element
<b>perspective</b>	Gives a 3D-positioned element some perspective
<b>perspective-origin</b>	Defines at which position the user is looking at the 3D-positioned element
<b>pointer-events</b>	Defines whether or not an element reacts to pointer events
<b>position</b>	Specifies the type of positioning method used for an element (static, relative, absolute or fixed)

## Reguli CSS – T

<b>tab-size</b>	Specifies the width of a tab character
<b>table-layout</b>	Defines the algorithm used to lay out table cells, rows, and columns
<b>text-align</b>	Specifies the horizontal alignment of text
<b>text-align-last</b>	Describes how the last line of a block or a line right before a forced line break is aligned when text-align is "justify"
<b>text-combine-upright</b>	Specifies the combination of multiple characters into the space of a single character
<b>text-decoration</b>	Specifies the decoration added to text
<b>text-decoration-color</b>	Specifies the color of the text-decoration
<b>text-decoration-line</b>	Specifies the type of line in a text-decoration
<b>text-decoration-style</b>	Specifies the style of the line in a text decoration
<b>text-indent</b>	Specifies the indentation of the first line in a text-block
<b>text-justify</b>	Specifies the justification method used when text-align is "justify"
<b>text-orientation</b>	Defines the orientation of the text in a line
<b>text-overflow</b>	Specifies what should happen when text overflows the containing element

## Reguli CSS – T

<b>text-shadow</b>	Adds shadow to text
<b>text-transform</b>	Controls the capitalization of text
<b>text-underline-position</b>	Specifies the position of the underline which is set using the text-decoration property
<b>top</b>	Specifies the top position of a positioned element
<b>transform</b>	Applies a 2D or 3D transformation to an element
<b>transform-origin</b>	Allows you to change the position on transformed elements
<b>transform-style</b>	Specifies how nested elements are rendered in 3D space
<b>transition</b>	A shorthand property for all the <i>transition</i> -* properties
<b>transition-delay</b>	Specifies when the transition effect will start
<b>transition-duration</b>	Specifies how many seconds or milliseconds a transition effect takes to complete
<b>transition-property</b>	Specifies the name of the CSS property the transition effect is for
<b>transition-timing-function</b>	Specifies the speed curve of the transition effect

## Reguli CSS – Q, R, S, U, V

<b>quotes</b>	Sets the type of quotation marks for embedded quotations
<b>resize</b>	Defines if (and how) an element is resizable by the user
<b>right</b>	Specifies the right position of a positioned element
<b>scroll-behavior</b>	Specifies whether to smoothly animate the scroll position in a scrollable box, instead of a straight jump
<b>unicode-bidi</b>	Used together with the direction property to set or return whether the text should be overridden to support multiple languages in the same document
<b>user-select</b>	Specifies whether the text of an element can be selected
<b>vertical-align</b>	Sets the vertical alignment of an element
<b>visibility</b>	Specifies whether or not an element is visible



## Reguli CSS – W, Z

<b>white-space</b>	Specifies how white-space inside an element is handled
<b>widows</b>	Sets the minimum number of lines that must be left at the top of a page when a page break occurs inside an element
<b>width</b>	Sets the width of an element
<b>word-break</b>	Specifies how words should break when reaching the end of a line
<b>word-spacing</b>	Increases or decreases the space between words in a text
<b>word-wrap</b>	Allows long, unbreakable words to be broken and wrap to the next line
<b>writing-mode</b>	Specifies whether lines of text are laid out horizontally or vertically
<b>z-index</b>	Sets the stack order of a positioned element



# Intrebari

“I.Like.IT@ETTI.TUIASI.RO”

[www.tuiasi.ro](http://www.tuiasi.ro)

1813

1860

1910

1937

1993

## Contact

- Laboratorul de microunde si optoelectronica
- <http://rf-opto.etti.tuiasi.ro>
- [rdamian@etti.tuiasi.ro](mailto:rdamian@etti.tuiasi.ro)