



CSS

Óbudai Egyetem, Java Programozás Mérnök-informatikai kar Labor 6

Bedők Dávid
2012.09.30.
v0.3

CSS

Cascading Style Sheets

- nem programozási, hanem **stílusleíró nyelv**
- nem csupán HTML/XHTML formázására használható, azonban itt domináns
- W3C (World Wide Web Consortium) kezeli specifikációját
- aktuális verziója CSS3, mely még fejlesztés alatt áll
- a **weboldalak kinézetének, stílusának meghatározásáért felelős**

Elvek

A HTML (főleg HTML5 előtt) tartalmaz számos olyan tag-et, mely weboldalak kinézetét definiálja, és nem a tartalmát. Ezen tag-ek használatát megpróbáljuk a lehetőségekhez mérten a legjobban kerülni, és minden stílus beállítást CSS-ben definiálni.

Nem lesz scope az, hogy hogyan lehet ezt máshogy megtenni, hogy miként lehet keverni a kettőt. A cél egy helyesnek vélt út mutatása egy tisztább munka felé.

Hello CSS

```
<!DOCTYPE html>
<html>
<head>
<link rel="stylesheet" type="text/css" href="style/sample.css"
/>
<title>hello css</title>
</head>
<body>
<h1>Hello CSS!</h1>
</body>
</html>
```

hellocss.html

```
body {
  font-family    : Verdana;
  font-size      : 11px;
  color          : RGB(0,0,150);
}
```

style/sample.css

Selectorok

A CSS stílusokat mindig hozzárendeljük a HTML dokumentumban található elemek egy halmazához. A stílusokat csoportosítjuk ({} pár), a csoportokat selectorokkal látjuk el.

- egy csoportnak lehet több selectora (vesszővel elválasztva: body, p, pre {...})
- egy selector több HTML elemet is kiválaszthat (de gyakori az is, hogy csak egy specifikusat)
- a HTML elemek a rájuk érvényes összes stílus beállítás összefésülését megkapják (létezik a deklaráció helyétől függő prioritási sorrend, és ennél fogva előfordulhat az is, hogy egyik stílus kiüti, avagy felülírja a korábban definiáltat)

Element selector

A HTML elem/tag nevét kell megadni.
Lehetőség van hierarchikus megszorításra!

```
table {  
    // <table>-re lesz érvényes  
}  
  
table, div {  
    // <table>-re és <div>-re lesz érvényes  
}  
  
table p {  
    // <table>-en belüli <p>-kre érvényesül (table-re nem!  
}
```

Létezik még `element>element` és `element+element` reláció is, de ezekkel részleteiben nem foglalkozunk.

Element selector

```
<!DOCTYPE html>
<html>
<head>
<link rel="stylesheet" type="text/css"
href="style/elementselector.css" />
<title>element selector</title>
</head>
<body>
<div id="divaid" class="divclass">DIV A</div>
<div id="divbid" class="divclass">DIV B</div>
<div id="divcid">DIV C</div>
</body>
</html>
```

elementselector.html

```
div {
  font-family      : Verdana;
  font-size        : 11px;
  color            : RGB(0,0,150);
}
```

style/elementselector.css

Class selector

```
<!DOCTYPE html>
<html>
<head>
<link rel="stylesheet" type="text/css"
href="style/classselector.css" />
<title>class selector</title>
</head>
<body>
<div id="divaid" class="divclass">DIV A</div>
<div id="divbid" class="divclass">DIV B</div>
<div id="divcid">DIV C</div>
</body>
</html>
```

classselector.html

```
.divclass {
  font-family      : Verdana;
  font-size        : 11px;
  color            : RGB(0,0,150);
}
```

style/classselector.css

ID selector

```
<!DOCTYPE html>
<html>
<head>
<link rel="stylesheet" type="text/css"
href="style/idselector.css" />
<title>id selector</title>
</head>
<body>
<div id="divaid" class="divclass">DIV A</div>
<div id="divbid" class="divclass">DIV B</div>
<div id="divcid">DIV C</div>
</body>
</html>
```

idselector.html

```
#divaid,
#divcid {
    font-family      : Verdana;
    font-size        : 11px;
    color            : RGB(0,0,150);
}
```

style/idselector.css

Attribute selector

```
<!DOCTYPE html>
<html>
<head>
<link rel="stylesheet" type="text/css"
href="style/attributeselector.css" />
<title>attribute selector</title>
</head>
<body>
<div id="divaid" class="divclass">DIV A</div>
<div class="divclass">DIV B</div>
<div id="divcid">DIV C</div>
</body>
</html>
```

attributeselector.html

```
[id] {
  font-family      : Verdana;
  font-size        : 11px;
  color            : RGB(0,0,150);
}
```

style/attributeselector.css

Attribute selector

A HTML attribútum nevét kell megadni, de van lehetőség név-érték pár alapján szűrni is!

```
[id] {  
    // azon elemekre lesz érvényes, aminek az id-ja megadott  
}  
  
[type="text"] {  
    // azon elemekre lesz érvényes, melyeknek type  
    // attribútumának értéke "text"  
}
```

Létezik még `[attribute~="value"]` és `[attribute|="value"]` selector is, melyekkel részleges tartalmazásra, illetve elején lévő egyezésre lehet szűrni.

CSS3 megfelelői ezeknek: `[attribute^="value"]`,
`[attribute$="value"]` és `[attribute*="value"]`

Event selector

```
<!DOCTYPE html>
<html>
<head>
<link rel="stylesheet" type="text/css" href="style/ eventselector.css" />
<title>event selector</title>
</head>
<body>
<a href="#a">link A</a>
<a href="#b">link B</a>
</body>
</html>
```

eventselector.html

```
a:link {
    color          : RGB (0,0,150) ;
}
a:visited {
    color          : RGB (0,150,0) ;
}
a:hover {
    color          : RGB (150,0,0) ;
}
```

style/eventselector.css

Létezik továbbá: :active,
:focus, :first-letter,
:first-line, :first-child, illetve
CSS3-ban :enabled,
:disabled, :checked

CSS3 selector lehetőségei I

css3selector.html

```
<!DOCTYPE html>
<html>
<head>
<link rel="stylesheet" type="text/css"
href="style/css3selector.css" />
<title>css3 selector</title>
</head>
<body>
<table>
  <tbody>
    <tr>
      <td>10</td>
      <td>asdfgh</td>
      <td>4343</td>
    </tr>
    ...
  </tbody>
</table>
</body>
```

CSS3 selector lehetőségei II

```
table {  
    border-spacing    : 0px;  
}  
td {  
    border            : 1px solid black;  
    width             : 100px;  
}  
tr:hover {  
    color             : RGB(250,250,250);  
    cursor            : pointer;  
}  
tr:nth-of-type(odd) {  
    background-color  : RGB(200,200,200);  
}  
tr:nth-of-type(even) {  
    background-color  : RGB(150,150,150);  
}
```

<style/css3selector.css>

További ezzel rokon selectorok is bekerültek a CSS3-ba, mint a :first-of-type, :last-of-type, :only-of-type, :only-child, :nth-child(n), :nth-last-child(n), :nth-last-of-type(n), :last-child.

Selectorok kombinációja

A selectorokat szinte tetszés szerint kombinálhatjuk (az észszerűség szab általában határt). Pl. elég gyakori az element és az attribútum szerinti selector kombinációja.

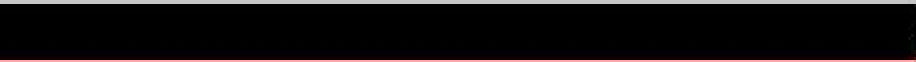







```
input [type="text"] {  
    // azon <input> elemekre lesz érvényes, melyeknek type  
    // attribútumának értéke "text"  
}  
  
ul .demo {  
    // az <ul>-en belüli class="demo" elemekre vonatkozik  
}  
  
ul.demo {  
    // a class="demo"-val rendelkező <ul> tagekre vonatkozik  
}
```

CSS Unitok

Ott, ahol mértéket kell kifejezni, valamilyen egységre van szükség. CSS-ben ezek a következők:

- %: százalék
- in: inch
- cm: centiméter
- mm: milliméter
- em: 1em az aktuális font mérete, 2em a duplája, stb.
- ex: x-magassága a font-nak
- pt: 1/72 inch (point)
- pc: 12pt (pica)
- px: pixels

Színkódok I

Color	Color HEX	Color RGB
	#000000	rgb(0,0,0)
	#FF0000	rgb(255,0,0)
	#00FF00	rgb(0,255,0)
	#0000FF	rgb(0,0,255)
	#FFFF00	rgb(255,255,0)
	#00FFFF	rgb(0,255,255)
	#FF00FF	rgb(255,0,255)
	#C0C0C0	rgb(192,192,192)
	#FFFFFF	rgb(255,255,255)

Színkódok II

000000	000033	000066	000099	0000CC	0000FF
003300	003333	003366	003399	0033CC	0033FF
006600	006633	006666	006699	0066CC	0066FF
009900	009933	009966	009999	0099CC	0099FF
00CC00	00CC33	00CC66	00CC99	00CCCC	00CCFF
00FF00	00FF33	00FF66	00FF99	00FFCC	00FFFF
330000	330033	330066	330099	3300CC	3300FF
333300	333333	333366	333399	3333CC	3333FF
336600	336633	336666	336699	3366CC	3366FF
339900	339933	339966	339999	3399CC	3399FF
33CC00	33CC33	33CC66	33CC99	33CCCC	33CCFF
33FF00	33FF33	33FF66	33FF99	33FFCC	33FFFF
660000	660033	660066	660099	6600CC	6600FF
663300	663333	663366	663399	6633CC	6633FF
666600	666633	666666	666699	6666CC	6666FF
669900	669933	669966	669999	6699CC	6699FF
66CC00	66CC33	66CC66	66CC99	66CCCC	66CCFF
66FF00	66FF33	66FF66	66FF99	66FFCC	66FFFF
990000	990033	990066	990099	9900CC	9900FF
993300	993333	993366	993399	9933CC	9933FF
996600	996633	996666	996699	9966CC	9966FF
999900	999933	999966	999999	9999CC	9999FF
99CC00	99CC33	99CC66	99CC99	99CCCC	99CCFF
99FF00	99FF33	99FF66	99FF99	99FFCC	99FFFF
CC0000	CC0033	CC0066	CC0099	CC00CC	CC00FF
CC3300	CC3333	CC3366	CC3399	CC33CC	CC33FF
CC6600	CC6633	CC6666	CC6699	CC66CC	CC66FF
CC9900	CC9933	CC9966	CC9999	CC99CC	CC99FF
CCCC00	CCCC33	CCCC66	CCCC99	CCCCCC	CCCCFF
CCFF00	CCFF33	CCFF66	CCFF99	CCFFCC	CCFFFF
FF0000	FF0033	FF0066	FF0099	FF00CC	FF00FF
FF3300	FF3333	FF3366	FF3399	FF33CC	FF33FF
FF6600	FF6633	FF6666	FF6699	FF66CC	FF66FF
FF9900	FF9933	FF9966	FF9999	FF99CC	FF99FF
FFCC00	FFCC33	FFCC66	FFCC99	FFCCCC	FFCCFF
FFFF00	FFFF33	FFFF66	FFFF99	FFFFCC	FFFFFF

Az RGB komponensek megadásával 16 millió különböző színt definiálhatunk. A színeket megadhatjuk hex és RGB(R,G,B) alakban is.

A bal oldali táblázat az ún. "web safe color" 256 színe (ma már ez történelem).

További lehetőségek:

- RGBA(R,G,B,A) alak, ahol A az alpha channel (átlátszóság) 0.0 és 1.0 közötti értékekkel
- HSL és HSLA alak (hue (0-360), saturation (%), and lightness (%), alpha channel (0.0-1.0))
- előre definiált szín nevek (17 alap, de összesen 147 darab)

Háttér

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

backgrounds.html

```
..
<div class="bgrColor">Lorem ipsum dolor sit amet, consectetur
adipiscing elit.</div>
<div class="bgrImage">
<p>Lorem ipsum dolor sit amet, consectetur adipiscing elit. In
vehicula euismod scelerisque.</p></div>
```

```
.bgrColor {
  background-color : RGB(200,200,150);
}
.bgrImage {
  background-image      : url('../images/book.png');
  background-repeat     : no-repeat;
  /* [repeat|no-repeat|repeat-x|repeat-y] */
  background-position : right top;
  /* [left|center|right] [top|center|bottom]
  */
}
```

style/backgrounds.css

Szöveg formázás

```
<link rel="stylesheet" type="text/css" href="style/ textformatting.css" />
..
<div class="demo">Lorem ipsum dolor sit amet, <a href="#">consectetur</a>
adipiscing elit.</div>
<p>Maecenas... molestie.</p>
```

textformatting.html

```
.demo {
  color                : RGB(170,170,120);
  text-align           : center; /* [left|center|right|justify] */
}
.demo a {
  text-decoration     : none;
  /* [none|overline|line-through|underline|blink] */
  text-transform      : uppercase;
  /* [uppercase|lowercase|capitalize] */
}
p {
  text-indent         : 50px;
  letter-spacing     : 2px;
  /* can be negative! */
  word-spacing       : 5px;
  line-height        : 150%;
  direction          : ltr;
  /* [ltr|rtl] */
}
```

CSS3-ban van lehetőség
árnyékolásra is:
**text-shadow: 2px 2px
#FF0000;**

style/textformatting.css

Betűtípusokról általában

Hiába definiál egy weboldal egy specifikus betűtípust, ha a gép, melyen az oldalt megtekintik, ezt nem ismeri. A tartalom közvetítése szempontjából szerencsésebb, ha a szöveg mindenképpen megjelenik valahogyan, ezért vannak tipikusan weben használt, elterjedt betűtípusok.

A CSS ezen kívül egy másik módszert is alkalmaz: sokféle betűtípus létezik, melyek között vannak hasonlóak, ezért a CSS-ben ezeket csoportosították, és nekünk elég csak a csoportot definiálnunk, a kliens gépe majd kiválasztja az ebbe a csoportba tartozó betűtípus egyikét. Ezek a "generic family" nevek:

- serif (Times New Roman, Georgia) "talpas betűk", vezetik az olvasást
- sans-serif (Arial, Verdana) "nem talpas betűk"
- monospace (Courier New) azonos szélességű betűk

A betűtípusokat sorba rendezhetjük (az első ismertet fogja a kliens gép használni), ezért a sor utolsó elem mindig egy "generic family" típus legyen!

Font-family példák

Georgia, serif

"Palatino Linotype", "Book Antiqua", Palatino, serif

"Times New Roman", Times, serif

Arial, Helvetica, sans-serif

"Arial Black", Gadget, sans-serif

"Comic Sans MS", cursive, sans-serif

Impact, Charcoal, sans-serif

"Lucida Sans Unicode", "Lucida Grande", sans-serif

Tahoma, Geneva, sans-serif

"Trebuchet MS", Helvetica, sans-serif

Verdana, Geneva, sans-serif

"Courier New", Courier, monospace

"Lucida Console", Monaco, monospace

CSS3-ban megjelent a lehetőség saját font-ok betöltésére és használatára:

```
@font-face {  
    font-family : customFont;  
    src        : url('font.ttf');  
}
```

Betűtípusok

```
<link rel="stylesheet" type="text/css" href="style/ fonts.css" />
..
<body>
<p>Maecenas in fringilla ante...</p>
</body>
```

fonts.html

```
body {
  font-size          : 100%;
  /* this set the default font-size (require for IE) */
}
p {
  font-family       : Arial, Verdana, sans-serif;
  font-style        : italic;
  /* [normal|italic|oblique] */
  font-size         : 0.75em; /* 16px * 0.75 = 12px */
  /* font-size: 13px; */
  font-weight       : bold;
  /* [lighter|normal|bold] or 100..900 normal:400 bold:700 */
  font-variant      : small-caps;
  /* [normal|small-caps] */
```

style/fonts.css

Listák formázása I

```
<link rel="stylesheet" type="text/css" href="style/lists.css"
/>
..
<ul class="demo">
  <li>Turbo Pascal</li>
  <li>PHP</li>
  <li>C#</li>
  <li>Java</li>
</ul>
<ol>
  <li>Turbo Pascal</li>
  <li>PHP</li>
  <li>C#</li>
  <li>Java</li>
</ol>
<ul class="demoimg">
  <li>Turbo Pascal</li>
  <li>PHP</li>
  <li>C#</li>
```

lists.html

Listák formázása II

style/lists.css

```
ul.demo {
  list-style-type      : circle;
  /* def: disc */
  /* [circle|disc|square|none] */
}
ol {
  list-style-type      : upper-roman;
  /* def: decimal */
  /* [armenian|cjk-ideographic|decimal|decimal-leading-zero|georgi
an|hebrew|hiragana|hiragana-iroha|katakana|katakana-iroha|lower
-alpha|lower-greek|lower-latin|lower-roman|upper-alpha|upper-la
tin|upper-roman|none] */
  list-style-position : inside;
  /* [inside|outside] */
}
ul.demoimg {
  list-style-image : url('../images/sample.gif');
  /* this is not crossbrowser solution */
}
```

Keretek

```
<link rel="stylesheet" type="text/css" href="style/fonts.css"
/>
..
<div id="demoA">demo</div>
<br/>
<div id="demoB">demo</div>
```

borders.html

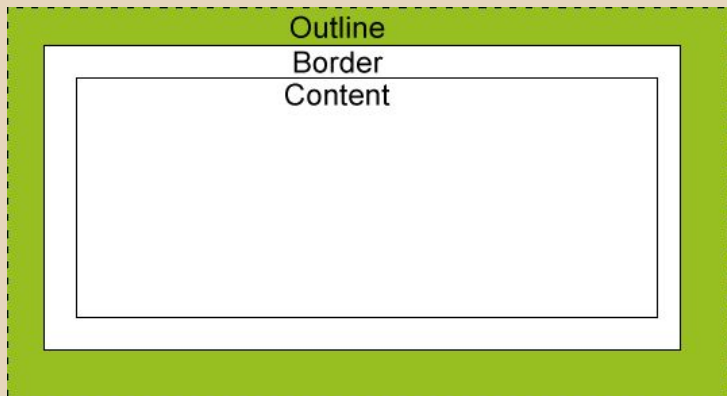
```
div#demoA {
  border-style           : solid;
  /* [none|hidden|dotted|dashed|solid|double|groove|ridge|inset|ou
tset]*/
  border-width          : 2px;
  border-color          : RGB(150,150,200);
}
div#demoB {
  border-top-style      :dotted;
  border-right-style   :solid;
  border-bottom-style  :dotted;
  border-left-style    :solid;
```

További érdekességek:
border-spacing:10px;
border-bottom-left-radius:2em;
border-radius:25px;

style/borders.css

Körvonal

A körvonal a kerethez (border) nagyon hasonló, azonban míg a keret méretei beleszámítanak az elem méret tulajdonságaiba, a körvonal (outline) nem!



```
div#demoA {  
  outline-style : solid;  
  outline-width : 2px;  
  outline-color : RGB(150,150,200);  
}
```

Ugyanazon style értékek megengedettek, mint a border esetén.

Margó

A margó egy traszparent (átlátszó) elem, mely az outline-on kívül helyezkedik el. A négy oldalát külön-külön lehet állítani.

```
div {  
    margin-top      : 100px;  
    margin-bottom   : 100px;  
    margin-right    : 50px;  
    margin-left     : 50px;  
}
```

Párnázás

A párnázás szintén egy traszparent (átlátszó) elem, hasonlóan a margóhoz. A különbség abban van, hogy itt a kereten (border) belüli terület párnázását állítjuk be.

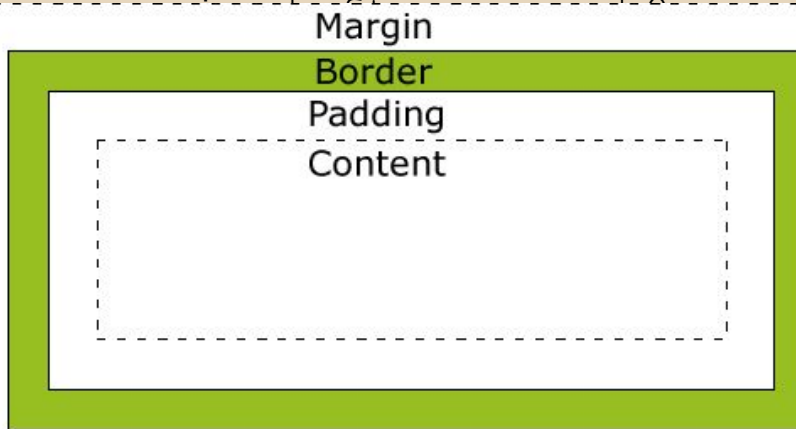
```
div {  
    padding-top      : 100px;  
    padding-bottom   : 100px;  
    padding-right    : 50px;  
    padding-left     : 50px;  
}
```

margin, outline, border, padding

```
<link rel="stylesheet" type="text/css" href="style/mobp.css" />  
..  
<br/>  
<div>demo</div>
```

mobp.html

```
div {  
  margin-top      :  
100px;  
  margin-bottom  : 100px;  
  margin-right   : 50px;
```



```
  outline-style   : solid;  
  outline-width   : 10px;  
  outline-color   :  
RGB (200, 50, 200);  
  border-style    : dashed;  
  border-width    : 5px;  
  border-color    :  
RGB (150, 150, 200);  
  padding-top     : 40px;  
  padding-bottom  : 40px;  
  padding-right   : 20px;  
  padding-left    : 20px;  
}
```

style/mobp.css

Méreték

IE8 és korábbi verziói nem képesek ezen összeadást elvégezni, ezért ott kicsit máshogy működik mindez.

A HTML elemek dimenzióit is megadhatjuk, azonban fontos tudni, hogy a tényleges kitöltést mi határozza meg!

szélesség

- margin-left
- border-left-width
- padding-left
- **width**
- padding-right
- border-right-width
- margin-right

magasság

- margin-top
- border-top-width
- padding-top
- **height**
- padding-bottom
- border-bottom-width
- margin-bottom

Dimenziók

A dimenziók tipikusan px-ben vagy %-ban adottak, néhol speciális konstansok is megjelennek.

```
div {  
    height      : 100px; /* %, auto */  
    width       : 100px; /* %, auto */  
    max-height  : 150px; /* %, none */  
    max-width   : 150px; /* %, none */  
    min-height  : 50px; /* % */  
    min-width   : 50px; /* % */  
}
```


Láthatatlanság

```
div {  
    visibility      :hidden;  
}
```

Eltünteti az elemet, de a hely, amit elfoglal ott marad!

```
div {  
    display       : none;  
}
```

Eltünteti az elemet, és az elem semmilyen helyet nem foglal az oldalból!

inline vs. block element

A HTML bemutatásakor az inline és a block elementek bemutatásra kerültek.

A CSS-ben van lehetőség az elementek ezen jellegzetességét megváltoztatni!

```
div { display : inline; }  
span { display : block; }
```

A fenti példa esetén a <div>-ek előtt (és után) nem lesz sorugrás, míg a -ek esetén igen.

Pozícionálás (nem scope)

Érdemes törekedni arra, hogy a HTML elemek pozícionálás nélkül is helyesen jelenjenek meg. Azonban indokolt esetben van lehetőség pozícionálni is. Pozícionálás fajtái:

- **statikus** pozíció (nincs pozícionálás)
- **fix** pozíció (a böngésző bal felső sarkához képest fog elhelyezkedni, sosem kerül át máshova (pl. scroll hatására sem) [position:fixed;top:1px;right:5px;left:5px;bottom:1px])
- **relatív** pozíció (az elem eredeti helyéhez képest mozog el) [position:relative;left:-25px;]
- **abszolút** pozíció (az első szülő eleméhez képest relatív, amennyiben az nem statikus, különben a <html> elemhez viszonyodik) [position:absolute;]

A pozícionálás "mellékhatása" az ún. overlapping, vagyis hogy az elemek elfedik egymást. Az elfedés sorrendje z-index alapján történik.

Lebegtetés

```
<link rel="stylesheet" type="text/css" href="style/float.css"
/>
..
<p>
  Lorem ipsum dolor sit amet..<br/>
  
  Lorem ipsum dolor sit amet..<br/>
  Lorem ipsum dolor sit amet..<br/>
</p>
<p>
  Maecenas in fringilla ante..<br/>
  
  Maecenas in fringilla ante..<br/>
</p>
```

float.html

```
.toright { float      : right; }
.toleft  { float      : left;  }
```

style/float.css

Árnyék

```
<link rel="stylesheet" type="text/css" href="style/shadow.css"
/>
..
<div>
  <p>Lorem ipsum dolor sit amet
mus.</p>
</div>
```

shadow.html

```
div {
  border      : 1px solid #AAAAAA;
  padding     : 5px;
  width       : 400px;
  box-shadow  : 10px 10px 10px 2px #101010;
  /* box-shadow: h-shadow v-shadow blur spread color; */
}
```

style/shadow.css

"Shorthand" tulajdonság

A legtöbb CSS kategória egyes stílus beállítása megadható rövid alakban is (így csökken a CSS mérete, ~talán könnyebben olvasható az állomány). Pl.:

```
div {  
    border          : 5px solid RGB(0,0,200);  
    /* border-width border-style border-color */  
}
```

nem scope ennek részletezése

Egyéb CSS tulajdonságok

```
overflow      : scroll; /* [scroll|hidden|auto] */
cursor       : wait; /* [crosshair|default|help|move|..] */
clear        : both; /* [left|right|both|none] */
animation    : ...
appearance   : normal;
              /* [normal|icon|window|button|menu|field] */
clip         : rect(0px,60px,200px,0px);
columns      : ...
content      : ...
counter-increment      : ...
counter-reset : ...
empty-cells  : hide; /* [show|hide] */
nav-down    : ...
opacity     : 0.5; /* 0.0 .. 1.0 */
perspective : 150;
resize      : both; /* [none|both|horizontal|vertical] */
transform   : rotate(7deg);
transition  : width 2s;
```

Kérdések

?