



BEVEZETÉS A STÍLUSLAPOK HASZNÁLATÁBA



Abonyi-Tóth Andor, ELTE Informatikai Kar

CSS



- Cascading Style Sheets
 - ▣ Lépcsőzetes stíluslapok
- CSS 1. szint: 1996 decemberében jelent meg
- Cél, hogy elkülönítsük a dokumentum megjelenését és tartalmát
 - ▣ Növeli a használhatóságot, rugalmasságot, megjelenés kezelhetőségét és csökkenti a dokumentum komplexitását.
- CSS 2.szint: 1998 május
- CSS 2.1.: butított változat, igazodva a böngészők (gyenge) tudásához
- CSS 3.szint: Moduláris felépítésű. Egyes részei fejlesztés alatt...
 - ▣ Nagy részét a korszerű böngészőprogramok támogatják.

CSS Modulok



□ <https://www.w3.org/Style/CSS/current-work>

TABLE OF SPECIFICATIONS

Ordered from most to least stable:

Completed	Current	Upcoming	Notes	EO
CSS Snapshot 2010	NOTE			EO
CSS Snapshot 2007	NOTE			EO
CSS Color Level 3	REC	REC	See Errata	EO
CSS Namespaces	REC	REC		EO
Selectors Level 3	REC	REC		EO
CSS Level 2 Revision 1	REC	REC	See Errata	EO
CSS Level 1	REC		Unmaintained, see Snapshot	EO
CSS Print Profile	NOTE			EO
Media Queries	REC	REC		EO
CSS Style Attributes	REC	REC		EO
Stable	Current	Upcoming	Notes	EO
CSS Backgrounds and Borders Level 3	CR	PR		EO
CSS Conditional Rules Level 3	CR	CR		EO
CSS Multi-column Layout	CR	CR		EO
CSS Values and Units Level 3	CR	PR		EO
CSS Cascading and Inheritance Level 3	CR	PR		EO
CSS Snapshot 2015	NOTE		Latest stable CSS	EO

Azonos tartalom, különböző arculat



□ <http://www.csszengarden.com/>

This page is an archived list of Zen Garden designs added to date. You can also see the ones that didn't make the cut. Total designs listed: 210

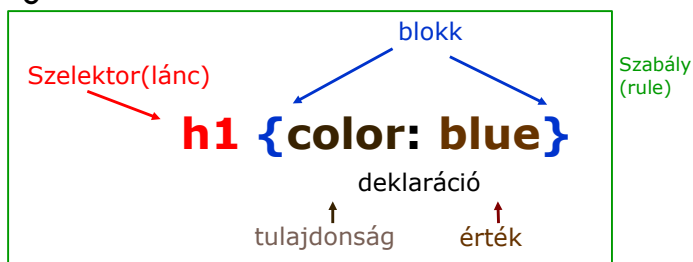
The screenshot shows the CSS Zen Garden website. At the top, there's a navigation bar with 'SELECT A DESIGN' and 'ARCHIVES'. Below this, a list of designs is displayed, each with a thumbnail and the designer's name. The design by El Collar de Tomas is highlighted. To the right, a detailed view of this design is shown, featuring a dark, atmospheric image of a building interior. The text on the page describes the design and provides links to download the HTML and CSS files. The overall layout is clean and professional, with a focus on the visual appeal of the CSS designs.

CSS utasítás



A CSS utasítás két részből áll:

- A szelektor(lánc) tartalmazza a formázandó HTML elem(ek)et
- A deklaráció végzi el a szelektorban meghatározott elemek formázását.



Stíluslap csatolása



```
<!DOCTYPE html>
<html lang="hu">
  <head>
    <meta charset="utf-8">
    <title>Stíluslapok</title>
    <link rel="stylesheet" type="text/css" href="kepek/pelda/stilus1.css" title="sajat">
    <style type="text/css">
      @import url("kepek/pelda/stilus2.css");
    <!--
    h1 {color: blue}
    -->
    </style>
  </head>

  <body>
    <h1>A címsor1 kék</h1>
    <h2>A címsor2 piros</h2>
    <p style="color:green">Az egész bekezdés zöld</p>
  </body>
</html>
```

Lapon belüli
definíció

Stíluslap
importálás

Hivatkozás
külső
stíluslapra

Beágyazott
(in-line)
megadás

Inline (beágyazott) megadás



```
<p style="color: green">Az egész bekezdés zöld</p>
```

- Ehhez a **style** paramétert használjuk.
- Csak az adott elemre (illetve azok gyerekeire) lesz érvényes a szabály.
- Nem szerencsés a használata, nem hatékony, körülményes a változtatások átvezetése.

```
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01//EN"
"http://www.w3.org/TR/html4/strict.dtd">
<html>
<head>
<meta http-equiv="Content-Type" content="text/html; charset=utf-8">
<title>Inline példa</title>
</head>
<body>
<h1 style="color:blue">Címsor 1</h1>
<p>Lorem ipsum dolor sit amet, consectetur adipiscing elit.</p>
<h1 style="color:blue">Címsor 1</h1>
<p>Lorem ipsum dolor sit amet, consectetur adipiscing elit.</p>
<h1 style="color:blue">Címsor 1</h1>
<p>Lorem ipsum dolor sit amet, consectetur adipiscing elit.</p>
</body>
</html>
```

Ha minden 1-es címsort kékre akarunk állítani, akkor NE így csináljuk!

```

<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01//EN"
"http://www.w3.org/TR/html4/strict.dtd">
<html>
<head>
<meta http-equiv="Content-Type" content="text/html; charset=utf-8">
<title> Már nem Inline példa</title>
<style type="text/css">
  h1 {color:blue}
</style>
</head>
<body>
<h1>Címsor 1</h1>
<p>Lorem ipsum dolor sit amet, consectetur adipiscing elit.</p>
<h1>Címsor 1</h1>
<p>Lorem ipsum dolor sit amet, consectetur adipiscing elit.</p>
<h1>Címsor 1</h1>
<p>Lorem ipsum dolor sit amet, consectetur adipiscing elit.</p>
</body>
</html>

```

A lapon belüli definícióval megoldottuk, hogy minden 1-es címsor kék színű legyen.

A lapon belüli definíciót külső állományba vihetjük, amit belinkelhetünk, vagy importálhatunk.

9

Hozzunk létre egy css állományt (pl. pelda.css) a következő tartalommal a HTML állományunk mappájában:

```
h1 {color:blue}
```

A HTML kód:

```

<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01//EN"
"http://www.w3.org/TR/html4/strict.dtd">
<html>
<head>
<meta http-equiv="Content-Type" content="text/html; charset=utf-8">
<title>Inline példa</title>
<link rel="stylesheet" type="text/css" href="pelda.css">
</head>
<body>
<h1>Címsor 1</h1>
<p>Lorem ipsum dolor sit amet, consectetur adipiscing elit.</p>
<h1>Címsor 1</h1>
<p>Lorem ipsum dolor sit amet, consectetur adipiscing elit.</p>
<h1>Címsor 1</h1>
<p>Lorem ipsum dolor sit amet, consectetur adipiscing elit.</p>
</body>
</html>

```

Ezzel a megoldással ugyanazt a stíluslapot több oldalhoz is hozzárendelhetjük. Ha úgy döntünk, hogy a címsor mégsem kék színű, akkor csak egy helyen kell megváltoztatni a tulajdonságot...

10

Tulajdonságok csoportosítása



`h1 {font-family: verdana}`
`h2 {font-family: verdana}`
`h3 {font-family: verdana}` → `h1, h2, h3 {font-family: verdana}`

A szelektorokat vesszővel választjuk el.

`h1 {font-family: helvetica}`
`h1 {font-size: 12pt}`
`h1 {font-style: normal}`



`h1 {`
 `font-family: helvetica;`
 `font-size: 12pt;`
 `font-style: normal;`
`}`

A tulajdonság:érték párok után pontosvessző áll.

`h1 {font-weight: bold}`
`h1 {font-size: 12pt}`
`h1 {font-family: helvetica}`



`h1 {font: bold 12pt helvetica}`

Itt az egyes értékek között szóköz van.
(shorthand megadásnak nevezi az angol szabvány)

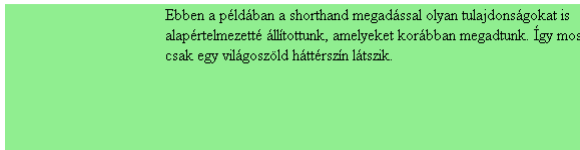
Shorthand megadás specialitása



- A rövidített (shorthand) paramétermegadásnál ügyelni kell arra, hogy ezzel minden olyan altulajdonságot is alapértelmezett állapotba tudunk hozni, amit explicit módon nem adtunk meg.

```
<style type="text/css">
div {
  background-image:url('http://lorempixel.com/150/150/sports/2/');
  background-repeat:no-repeat;
  background-position: top left;
  height:150px;
  padding-left:160px;
}
div {background:lightgreen}
</style>
```

Ebben a példában a shorthand megadással olyan tulajdonságokat is alapértelmezetté állítottunk, amelyeket korábban megadtunk. Így most csak egy világoszöld háttérszín látszik.



Ismeretlen tulajdonságok



- A böngészők figyelmen kívül hagyják az ismeretlen vagy hibás tulajdonságokat.

<code>img { float: left }</code>	<code>/* helyes CSS 2.1 deklaráció */</code>
<code>img { float: left here }</code>	<code>/* a „here” hibás érték */</code>
<code>img { background: "red" }</code>	<code>/* az értéket nem kell idézőjelbe tenni */</code>
<code>img { border-width: 3 }</code>	<code>/* hiányzik a mértékegység */</code>

Amit a fentiekből a CSS értelmező figyelembe vesz:

```
img { float: left }  
img {}  
img {}  
img {}
```

Lezáratlan string



- A megnyitott, de le nem zárt idézőjelek, aposztrófok szintén problémát okoznak.

```
p { color: green;  
    font-family: 'Courier New Times  
    font-style: italic;  
    background-color: yellow; }
```

Amit a fentiekből a CSS értelmező figyelembe vesz:

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

Stíluslapok optimalizálása



□ <http://www.cssoptimiser.com/>



```
h1 {color: #000080}
h2 {color: #000080}
h3 {color: #000080}
```

```
p {font-family: Arial, Verdana, sans-serif}
p.fejlec {color: #000080}
```



```
h1,h2,h3,p.fejlec {
    color:#000080
}
```

```
p {
    font-family:Arial,Verdana,sans-serif
}
```

Stíluslapok validálása



□ <http://jigsaw.w3.org/css-validator/>

 **CSS Validation Service**
Cascading Style Sheets (CSS) és stíluslapokkal rendelkező (X)HTML dokumentumok ellenőrzése

URI segítségével | **File feltöltéssel** | **Közvetlen bevitellel**

Érvényesítés URI segítségével

Adja meg a dokumentum (HTML CSS-sel vagy csak CSS) URI-jét, melyet érvényesíteni szeretne:

Cím:

► További lehetőségek

Mértékegységek



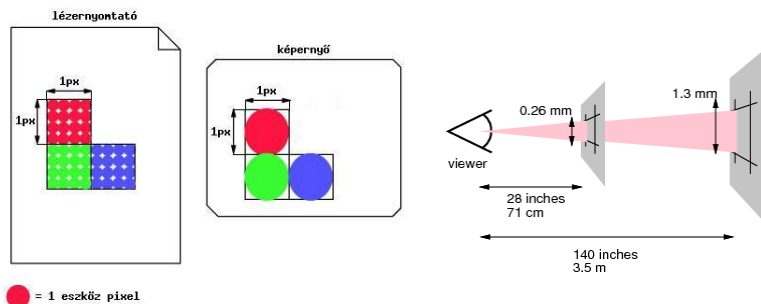
A számokat egybeírjuk a mértékegységekkel!
Vagyis 12px és nem 12 px.

Abszolút	Relatív
in (2.54 cm) cm mm pt (1/72in) pc (12pt)	em (adott környezet betűméretéhez képest történő megadás. pl. 1.2em) ex (x-magasság - többnyire az adott font x betűjének magassága. pl. 2ex) % (pl. 150%) px (képpont) pl. 16px [csak bizonyos értelemben tekinthető relatívnak, körütekintéssel használjuk!]

Mértékegységek (Pixel)



- Pixel: relatív, mert mérete a megjelenítő eszköztől függ.

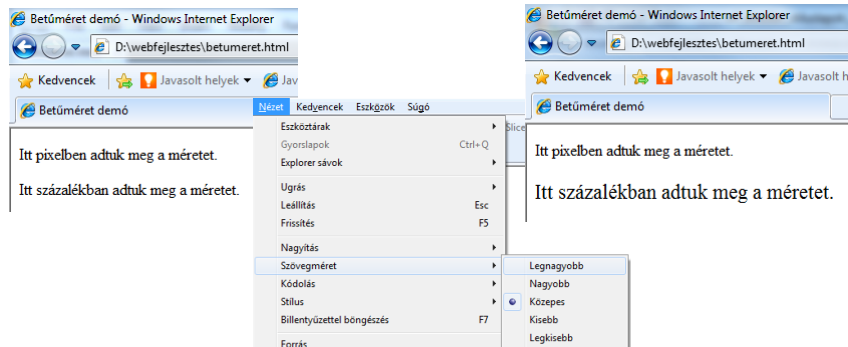


<http://weblabor.hu/cikkek/cssalapjai6>

Mértékegységek (Pixel)



- De betűméret megadásnál lehetőleg ne használjuk, mert nem lesz (minden böngészőben) nagyítható a betűméret.



Százalékos megadás



- A %-os érték relatív értéket jelent, de hogy minek az értékét, az változó. Lehet:
 - Azonos elem más tulajdonsága,
 - Szülő elem egy tulajdonsága,
 - A formázott tartalom egy tulajdonsága (pl. tartalmazó elem szélessége).

Színjelölések



□ Szín nevek

□ `<p style="color:BlueViolet">`

Color Name	HEX	Color
AliceBlue	#F0F8FF	
AntiqueWhite	#FAEBD7	
Aqua	#00FFFF	
Aquamarine	#7FFFD4	
Azure	#F0FFFF	
Beige	#F5F5DC	
Bisque	#FFE4C4	
Black	#000000	
BlanchedAlmond	#FFEBCD	
Blue	#0000FF	
BlueViolet	#8A2BE2	
Brown	#A52A2A	
BurlyWood	#DEB887	
CadetBlue	#5F9EA0	
Chartreuse	#7FFF00	
Chocolate	#D2691E	
Coral	#FF7F50	

http://www.w3schools.com/cssref/css_colornames.asp

Színjelölések



□ Szín megadás decimális RGB kóddal

□ `<p style="color:rgb(51,71,28)">`

□ Szín megadás hexadecimális RGB kóddal

□ `<p style="color:#f66">`

□ `<p style="color:#deb887">`

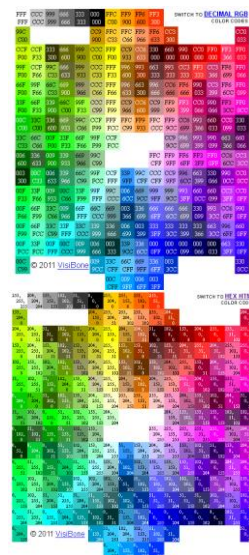
□ Szín megadás decimális HSL kóddal

□ `<p style="color:hsl(300,50,70)">`

□ Szín megadás százalékosan

□ `<p style="color:rgb(50%,70%,30%)">`

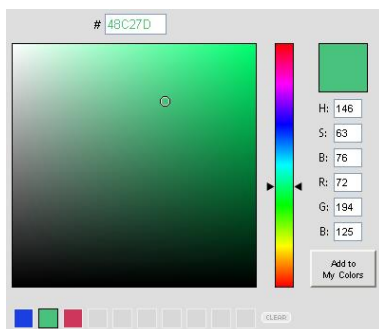
<http://html-color-codes.com/>



Színkódok meghatározása



- Szinte bármelyik grafikus rajzoló, képmanipuláló program alkalmas a színkódok meghatározására.
- Hasznos online eszköz:
 - <http://www.colorpicker.com/>



Átlátszóság



- A korszerű (CSS3 kompatibilis) böngészők az átlátszóságot is ismerik
 - rgb helyet használhatunk rgba, hsl helyett hsla megadást
 - `background-color: rgba(255, 255, 255, 0.5);`
 - Az utolsó paraméter 0 és 1 közti szám.
 - 0: teljesen átlátszó
 - 1: teljesen átlátszatlan

<http://24ways.org/2009/working-with-rgba-colour/>

HEADING 1

```
h1 {  
    color: rgb(0, 0, 0);  
    background-color: rgba(255, 255, 255, 0.5);  
}
```

Rgba, hsla stb. támogatottság



																
	CHROME	FIREFOX	OPERA	SAFARI				CHROME	FIREFOX	OPERA	IE					
	25	20	12	15	1	5.1	6	25	15	12	6	7	8	9	10	
RGBA	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✗	✗	✗	✓	✓	93%
HSLA	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✗	✗	✗	✓	✓	93%
Box Sizing	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✗	✗	✓	✓	✓	54%
Background Size	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✗	✗	✗	✓	✓	92%
Multiple Backgrounds	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✗	✗	✗	✓	✓	92%
Border Image	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✗	✗	✗	✗	✗	87%
Border Radius	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✗	✗	✗	✓	✓	93%
Box Shadow	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✗	✗	✗	✓	✓	92%
Text Shadow	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✗	✗	✗	✗	✓	82%
Opacity	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✗	✗	✗	✓	✓	93%
CSS Animations	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✗	✗	✗	✗	✓	75%
CSS Columns	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✗	✗	✗	✗	✓	88%

<http://www.findmebyip.com/litmus/>

Fallback



- Érdeemes gondoskodnunk arról, hogy az átlátszóságot nem ismerő böngészők is megjelenítsenek egy színt.
- ▣ A lenti példában az rgba-t nem ismerő böngészőkben beállításra kerül egy szín, amelyet felülírunk később, de ezt csak az rgba-t ismerő böngészők veszik figyelembe.

CSS

```
div {
  background: rgb(200, 54, 54); /* The Fallback */
  background: rgba(200, 54, 54, 0.5);
}
```

<http://css-tricks.com/rgba-browser-support/>

URL megadás



- `url('url');` vagy `url("url");`

- *Példa*

- `body { background-image: url('kepek/hatter.jpg'); }`
- `li { list-style: url('http://www.vala.hol/kep.png') disc }`

Dokumentumfa

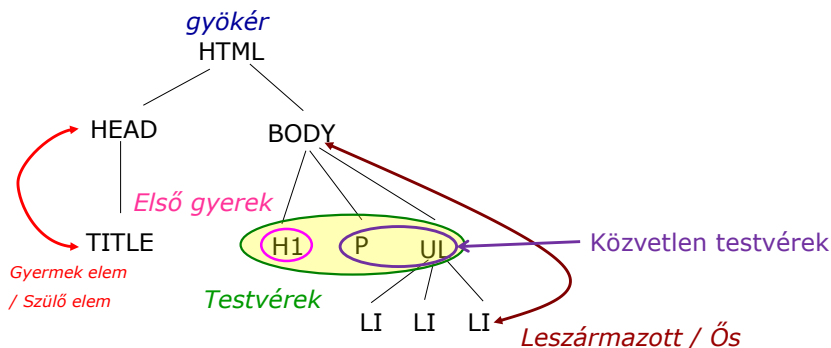


- **Szülőnek** nevezzük azokat az elemeket, amelyek tartalmazznak más tageket, amik a szülő elem **gyermekai**.
- Azokat a gyermekeket, amelyek közös szülővel rendelkeznek, **testvéreknek** nevezzük.
- Azok a testvérek, amelyek közvetlenül egymás után helyezkednek el a dokumentumban, **közvetlen testvéreknek** nevezzük.
- A fában egy elem alatti részfa elemeit az elem **leszármazottainak**, a fa az elem felett elhelyezkedő elemeit, pedig annak **őseinek** nevezzük.

Dokumentumfa



- Forrásdokumentumban van kódolva.
- Minden elemnek pontosan egy szülő eleme van, kivéve a **gyökérel**et, amelynek nincs szülő eleme.



Öröklődés



- A tulajdonságok (egy része) öröklődik a szülő elem(ek)től

```
<h1 style="color:blue">  
A címsor  
  <em>mindig</em>  
fontos.</h1>
```



A címsor mindig fontos.

Az em elem tartalma is kék lett, mert örökölte a színt a szülőtől...

- A teljes lapra vonatkozó tulajdonságokat a BODY elemhez kell kötni.

```
body  
{ font-size:12pt;  
  font-family: Arial, Verdana, sans-serif  
}
```

Öröklődés



- A gyermek elemek nem a szülőkből megadott relatív értékeket öröklik, hanem azok számított értékét!

```
body
{
  font-size: 12px;
  text-indent: 3em;
}

h1 { font-size: 25px }
```

Bekezdés első sorának behúzása. Itt ez most 36px-nek felel meg.

A h1 öröklíti a text-indent értéket a szülőtől (body), de csak a számított értéket. Vagyis a behúzás nem $3 \times 25\text{px} = 75\text{px}$, hanem 36px lesz.

Ha nem így lenne...



Címsor

Lorem ipsum dolor sit amet, consectetur adipiscing elit. Duis in purus augue. In hac habitasse platea dictumst. Sed fringilla, velit sed commodo feugiat, mi leo porttitor sapien, non rutrum est nisi vitae purus. Donec eget elit non ipsum vulputate imperdiet. Proin scelerisque porta mi, in dignissim sapien interdum vitae. Sed non ante sit amet tellus accumsan placerat non in elit.

Címsor

Lorem ipsum dolor sit amet, consectetur adipiscing elit. Duis in purus augue. In hac habitasse platea dictumst. Sed fringilla, velit sed commodo feugiat, mi leo porttitor sapien, non rutrum est nisi vitae purus. Donec eget elit non ipsum vulputate imperdiet. Proin scelerisque porta mi, in dignissim sapien interdum vitae. Sed non ante sit amet tellus accumsan placerat non in elit.

Ha nem számított értéket örökölné a gyerek, akkor a betűméret hatással lenne az első sor behúzására is, ami nem kívánt eredménnyel járna.

A számított érték öröklése miatt az első sor behúzása független lesz a betűmérettől...

Címsor

Lorem ipsum dolor sit amet, consectetur adipiscing elit. Duis in purus augue. In hac habitasse platea dictumst. Sed fringilla, velit sed commodo feugiat, mi leo porttitor sapien, non rutrum est nisi vitae purus. Donec eget elit non ipsum vulputate imperdiet. Proin scelerisque porta mi, in dignissim sapien interdum vitae. Sed non ante sit amet tellus accumsan placerat non in elit.

Címsor

Lorem ipsum dolor sit amet, consectetur adipiscing elit. Duis in purus augue. In hac habitasse platea dictumst. Sed fringilla, velit sed commodo feugiat, mi leo porttitor sapien, non rutrum est nisi vitae purus. Donec eget elit non ipsum vulputate imperdiet. Proin scelerisque porta mi, in dignissim sapien interdum vitae. Sed non ante sit amet tellus accumsan placerat non in elit.

Öröklődés és az inherit érték



- Nem minden beállítás öröklődik.
 - ▣ Pl. a szegély (border) nem öröklődik, de beállíthatjuk ezt az `inherit` értékkel.

```
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01//EN"
"http://www.w3.org/TR/html4/strict.dtd">
<html>
<head>
<meta http-equiv="Content-Type" content="text/html;
charset=utf-8">
<title></title>
<style type="text/css">
div {border:1px solid blue;padding:5px;}
p {border:inherit;}
</style>
</head>
<body>
<div>
<p>Lorem ipsum dolor sit amet...</p>
<p>Nam id nisl vitae neque posuere interdum...</p>
</div>
</body>
</html>
```

Lorem ipsum dolor sit amet, consectetur adipiscing elit. Aenean in dolor a tellus feugiat molestie sed quis nisi. Vivamus augue nisi, imperdiet id consequat id, luctus ac velit.

Nam id nisl vitae neque posuere interdum. Quisque pulvinar fermentum metus, at imperdiet purus accumsan blandit. Nunc bibendum, nibh vitae lobortis varius, erat eros elementum felis, a tempor nisl lectus at turpis. Praesent sodales mauris ac massa dignissim fermentum.

A bekezdéseken alpból nem lenne szegély.

Lorem ipsum dolor sit amet, consectetur adipiscing elit. Aenean in dolor a tellus feugiat molestie sed quis nisi. Vivamus augue nisi, imperdiet id consequat id, luctus ac velit.

Nam id nisl vitae neque posuere interdum. Quisque pulvinar fermentum metus, at imperdiet purus accumsan blandit. Nunc bibendum, nibh vitae lobortis varius, erat eros elementum felis, a tempor nisl lectus at turpis. Praesent sodales mauris ac massa dignissim fermentum.

Az inherit érték miatt a bekezdés is öröklí a szülő elem szegély beállításait.

A kezdeti (initial) érték



- Minden tulajdonságnak van egy kezdeti (initial) értéke, amely egy tulajdonság definíciós táblában van elhelyezve. Ha tehát vissza akarjuk állítani az alapértéket, de nem tudjuk, hogy az pontosan micsoda, használhatjuk az initial értéket.

```
div {min-width:initial;}
```

Összekapcsolt szelektor (leszármazás)



- Ha más tulajdonságokat akarok adni az elemnek, attól függően, hogy milyen elemben szerepel, összekapcsolhatom a szelektorokat
- Mód: `szelektor1 szelektor2 {definíció}`
- Példa

■ `h1 {color: blue}`

■ `em {color: blue}`

```
<h1>Ez nagyon <em>fontos</em> dolog</h1>  
<p> Ez is <em>fontos</em></p>
```

Ez nagyon fontos dolog
Ez is fontos.

`h1 em {color: red}`

Ez nagyon fontos dolog
Ez is fontos.

Osztályok (class) használata



□ CSS

■ `.osztálynév {definíció}`

- Ez egy általános osztály, amelyet több elemnél is felhasználhatunk.

■ `szelektor.osztálynév {definíció}`

- Ekkor egy adott szelektorhoz lesz kötve az osztály

```
.fontos {color:red;}
```

```
p.szegely {border:1px solid black;}
```

□ HTML

■ class attribútum használata

■ Egy elem több osztályba is tartozhat

```
<p class="fontos">  
<p class="szegely">  
<p class="szegely fontos">
```

Osztályok (class) használata



```
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01//EN"
"http://www.w3.org/TR/html4/strict.dtd">
<html>
<head>
<meta http-equiv="Content-Type"
content="text/html; charset=utf-8">
<title>Osztály példa</title>
<style type="text/css">
p {padding:5px;margin:10px;}
.fontos {color:red;}
p.szegely {border:1px solid black;}
h1.szegely {border:2px dotted blue;}
</style>
</head>
<body>
<h1>Címsor 1</h1>
<p class="fontos">Lorem ipsum dolor sit amet,
consectetur adipiscing elit.</p>
<p class="szegely">Lorem ipsum dolor sit amet,
consectetur adipiscing elit.</p>
<p class="szegely fontos">Lorem ipsum dolor sit
amet, consectetur adipiscing elit.</p>
<h1 class="szegely fontos">Címsor 1</h1>
<p>Lorem ipsum dolor sit amet, consectetur
adipiscing elit.</p>
</body>
</html>
```

Címsor 1

Lorem ipsum dolor sit amet, consectetur adipiscing elit

Lorem ipsum dolor sit amet, consectetur adipiscing elit

Lorem ipsum dolor sit amet, consectetur adipiscing elit

Címsor 1

Lorem ipsum dolor sit amet, consectetur adipiscing elit

← Több osztályba is tartozhat
ugyanaz az elem!

Több osztály használata



```
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01//EN"
"http://www.w3.org/TR/html4/strict.dtd">
<html>
<head>
<meta http-equiv="Content-Type" content="text/html; charset=utf-
8">
<title>Osztály példa</title>
<style type="text/css">
h1.szegely {border:2px dotted blue;}
.fontos {color:red;}
h1.szegely.fontos {border:1px solid red; color:black;}
</style>
</head>
<body>
<h1 class="szegely">Címsor 1</h1>

<h1 class="fontos">Címsor 1</h1>

<h1 class="szegely fontos">Címsor 1</h1>

</body>
</html>
```

Címsor 1

Címsor 1

Címsor 1



Alap esetben kék szegéllyel és piros színnel jelenne meg.
A .szegely.fontos CSS definíció miatt más lesz a megjelenése.

Egyedi azonosítók (ID) használata



□ CSS

```
div#fejléc {margin:10px}
```

▣ #azonosító, vagy szelektor#azonosító

□ HTML

```
<div id="fejléc"></div>
```

▣ id attribútum használata

▣ Egyedinek kell lennie az oldalon belül!

▣ Fontos a kis- és nagybetű közti különbség.

▣ HTML 4.01 szerint:

- Betűvel kezdődik, amelyet számok, betűk és a következő jelek követhetnek: - _ : .

▣ HTML 5: legalább 1 karakter, nem tartalmaz szóközt.

Példa



```
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01//EN" "http://www.w3.org/TR/html4/strict.dtd">
```

```
<html>
```

```
<head>
```

```
<title></title>
```

```
<meta http-equiv="Content-Type" content="text/html; charset=utf-8">
```

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

```
div#fejléc {height:150px; background-color:lightgray;}
```

```
#tartalom {height:auto; background-color:white;}
```

```
div#lablec {height:50px; background-color:lightgray;}
```

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

```
</style>
```

```
</head>
```

```
<body>
```

```
<div id="fejléc" class="kozep">Fejléc</div>
```

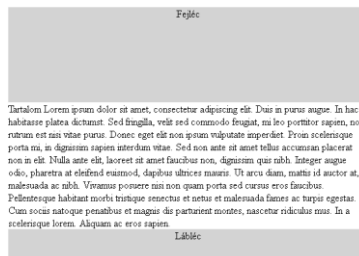
```
<div id="tartalom">Tartalom Lorem ipsum.. </div>
```

```
<div id="lablec" class="kozep">Lábléc</div>
```

```
</body>
```

```
</html>
```

id és class besorolás is lehet egyszerre..





Pszudó elemek és pszudó osztályok



- Pszudó elem (látszólagos elem)
 - ▣ A dokumentumfában nem jelölt elemek elérését teszi lehetővé. Pl. egy szónak az első betűje, vagy egy bekezdésnek az első sora, egy elem elé- illetve után generált elemek
 - :first-letter, :first-line, :before, :after
- Pszudó osztály (látszólagos osztály)
 - ▣ Az elemek egyes állapotaihoz rendelt osztályok

Pszedó elemek és pszedó osztályok



- A szintaxisuk két részből áll, egy szelektorból és annak egy **látszólagos osztályából**, ami egy **kettős ponttal kezdődik**
- Szelektor:látszólagos_osztály {def}

Link állapotok

Normál link

Ezt már meglátogattam

Főlé viszem az egeret

- a:link {} /* link */
- a:visited {} /* látogatott link */
- a:active {} /* aktív link */
- a:hover {} /* ha fölé visszük az egeret */



```
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01//EN"
"http://www.w3.org/TR/html4/strict.dtd">
<html>
  <head>
    <title></title>
    <meta http-equiv="Content-Type" content="text/html; charset=utf-8">
    <style type="text/css">
      body {background-color:black;}
      a:link {color:yellow; }
      a:visited {color:#ff9900; }
      a:hover {color:white;}
    </style>
  </head>
  <body>
    <p><a href="p1.html">Normál link</a></p>
    <p><a href="p2.html">Ezt már meglátogattam</a></p>
    <p><a href="p3.html">Főlé viszem az egeret</a></p>
  </body>
</html>
```

Látszólagos osztályok



- Kombinálhatók normál osztályokkal

- `a.kulso:visited {color: blue}`

A CSS1 -ben a `:link`, `:visited` és `:active` egymást kölcsönösen kizárták; a CSS2 -ben az `:active` előfordulhat együtt a `:link`, vagy `:visited` látszólagos osztállyal.

Példa:

```
a:link:active { color: red }
```

Látszólagos elemek



A HTML **elemek tartalmának kitüntetett részei**, például egy bekezdés első sora, vagy az első betűje. Ilyen elemek a következők:

- | | |
|--|---|
| □ <code>:first-letter</code> | egy szöveg első betűje . |
| □ <code>:first-line</code> | egy szöveg első sora . |
| □ <code>:after</code> | generált tartalom közvetlenül az elem után . |
| □ <code>:before</code> | generált tartalom közvetlenül az elem előtt . |
| CSS3 □ <code>::selection</code> | a kijelölés stílusát határozhatjuk meg. |

A látszólagos elemeket kezdhethetjük **szimpla vagy dupla kettősponttal** is, jelentésük változatlan marad. Kivéve a `::selection`-t ami csak dupla kettősponttal működik.

:first-letter



□ Az első karakter formázása

```
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01//EN"
"http://www.w3.org/TR/html4/strict.dtd">
<html>
<head>
<title></title>
<meta http-equiv="Content-Type" content="text/html;
charset=utf-8">
<style type="text/css">
  p {font-size: 12pt; line-height: 12pt}
  p:first-letter {font-size: 200%;}
  p.iniciales:first-letter {font-size: 300%;
float:left; padding-top:5px; padding-right:5px;}
</style>
</head>
<body>
  <p>Lorem ipsum dolor sit amet...</p>
  <p>Morbi sit amet felis eros, ut blandit ligula.
Vestibulum feugiat facilisis ullamcorper... </p>
  <p class="iniciales">Vivamus neque orci, viverra
vitae posuere at, laoreet varius metus. Mauris blandit
cursus feugiat... </p>
</body>
</html>
```

>Lorem ipsum dolor sit amet, consectetur adipiscing elit. Aenean ut ipsum sit amet mi tristique semper in sodales mi. Integer dictum dapibus arcu eu commodo. Donec nisl lorem, lobortis ac viverra a, tristique id lacus. Proin pharetra risus felis, at malesuada nisl. Donec congue libero in nibh malesuada eu molestie orci porttitor.

Morbi sit amet felis eros, ut blandit ligula. Vestibulum feugiat facilisis ullamcorper. Class aptent tacit sociosqu ad litora torquent per conubia nostra, per inceptos himenaeos. Nulla faucibus volutpat ipsum et semper. Suspendisse quam massa, luctus semper condimentum nec, luctus nec arcu.

Vivamus neque orci, viverra vitae posuere at, laoreet varius metus. Mauris blandit cursus feugiat. Fusce sed purus eu tellus pharetra eusmod. Ut eleifend pretium nisl id placerat. Aenean id condimentum ligula. Maecenas a odio eget neque dictum lacinia. Suspendisse auctor imperdiet bibendum. Fusce a dignissim dolor. Curabitur scelerisque erat accumsan felis placerat varius fermentum ipsum ultrices.

:first-line



□ Első sor formázása

```
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML
4.01//EN"
"http://www.w3.org/TR/html4/strict.dtd">
<html>
<head>
<title></title>
<meta http-equiv="Content-Type"
content="text/html; charset=utf-8">
<style type="text/css">
  p:first-line {font-size: 130%;}
</style>
</head>
<body>
<p>Lorem ipsum dolor sit amet, consectetur
adipiscing elit... </p>
<p>Morbi sit amet felis eros, ut blandit
ligula. Vestibulum feugiat facilisis
ullamcorper ...</p>
</body>
</html>
```

Lorem ipsum dolor sit amet, consectetur adipiscing elit. Aenean ut ipsum sit amet mi tristique semper in sodales mi. Integer dictum dapibus arcu eu commodo. Donec nisl lorem, lobortis ac viverra a, tristique id lacus. Proin pharetra risus felis, at malesuada nisl. Donec congue libero in nibh malesuada eu molestie orci porttitor.

Morbi sit amet felis eros, ut blandit ligula. Vestibulum feugiat facilisis ullamcorper. Class aptent tacit sociosqu ad litora torquent per conubia nostra, per inceptos himenaeos. Nulla faucibus volutpat ipsum et semper. Suspendisse quam massa, luctus semper condimentum nec, luctus nec arcu.

:before, :after, ::selection



```
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01//EN"
"http://www.w3.org/TR/html4/strict.dtd">
<html>
<head>
<meta http-equiv="Content-Type" content="text/html; charset=utf-8">
<title></title>
<style type="text/css">
p {padding:10px; border:1px solid black;
/* szegélyt rakunk a bekezdésekre */}

p.megjegyzes:before
{content: "Ez a szöveg elé kerül. "; font-style:italic; }

p.pelda:after
{content: "Ez a szöveg után kerül."; font-weight:bold; display:block; }

::selection {background-color:yellow;}

</style>
</head>
<body>
<p class="megjegyzes">Lorem ipsum dolor..</p>
<p class="pelda">Proin in massa dapibus neque commodo placerat a a ligula.
</body>
</html>
```

Ez a szöveg ki
lett jelölve a
böngésző-
programban.



Ez a szöveg elé kerül. Lorem ipsum dolor sit amet, consectetur adipiscing elit

Proin in massa dapibus neque commodo placerat a a ligula.
Ez a szöveg után kerül.

Dinamikus látszólagos osztályok



<input type="checkbox"/> :link	olyan link, ami még nem volt kiválasztva .
<input type="checkbox"/> :visited	olyan link, ami már ki volt kiválasztva , vagyis rákattintottunk.
<input type="checkbox"/> :hover	olyan elem, amelyre az egérkurzor mutat .
<input type="checkbox"/> :active	éppen aktív elem . Pl. egy betöltődő link.
<input type="checkbox"/> :focus	olyan elem, amelyiken a fókusz van .
CSS3 <input type="checkbox"/> :target	az anchor (könyvjelző) linkek célpontjainak stílusaiért felel.
CSS3 <input type="checkbox"/> :enabled	annak az elemnek (többnyire űrlapelemnek) a stílusaiért felel, amelyek engedélyezve van .
CSS3 <input type="checkbox"/> :disabled	annak az elemnek a stílusaiért felel, amelyek le van tiltva .
CSS3 <input type="checkbox"/> :checked	annak az checkbox -nak a stílusaiért felel, amely ki van jelölve .

Később ezekre is nézünk példákat!

Strukturális látszólagos osztályok



CSS3	<input type="checkbox"/> :root	elem, amely nem rendelkezik szülővel .
CSS3	<input type="checkbox"/> :empty	az az elem, ami nem rendelkezik egyetlen gyermekkel sem .
CSS3	<input type="checkbox"/> :only-child	az az elem, amely nem rendelkezik testvérrel .
CSS3	<input type="checkbox"/> :only-of-type	az az elem, amelyből nincs hasonló típusú a testvérei között .
	<input type="checkbox"/> :first-child	az az elem, amely az összes testvére közül az első .
CSS3	<input type="checkbox"/> :first-of-type	az az elem, amely a szelektor típusából a testvérei között az első .
CSS3	<input type="checkbox"/> :nth-child(n)	a testvérei között az n. elem .
CSS3	<input type="checkbox"/> :nth-of-type(n)	a testvérei között az n. az adott szelektorra illeszkedő elemek közül .
CSS3	<input type="checkbox"/> :nth-last-of-type(n)	a testvérei között hátulról az n. az adott szelektorra illeszkedő elemek közül .
CSS3	<input type="checkbox"/> :last-child	a testvérei közül az utolsó elem .
CSS3	<input type="checkbox"/> :last-of-type	az az elem, amely a szelektor típusából a testvérei között az utolsó .

Később ezekre is nézünk példákat!

Egyéb látszólagos osztályok



Nyelvfüggőség

- ☐ :lang
 - segítségével **nyelvfüggővé tehetjük** stílusszabályainkat. A nyelvet a lang HTML attribútum segítségével állíthatjuk be.

Tagadás

- CSS3** ☐ :not()
 - **tagadást fogalmazhatunk meg**. Erre az összes olyan szelektor illeszkedik, amelyekre nem igaz a paraméterként adott tulajdonság.

Később ezekre is nézünk példákat!

Teljes dokumentáció

<https://www.w3.org/TR/css-cascade-3/>



Probléma felvetés

Az alábbi kód milyen színű 1-es címsort eredményez?

```
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01//EN"
"http://www.w3.org/TR/html4/strict.dtd">
<html>
  <head>
    <title></title>
    <meta http-equiv="Content-Type" content="text/html; charset=utf-8">
    <style type="text/css">
      div.szegely h1 {color:green}
      h1 {color:blue;}
    </style>
  </head>
  <body>
    <div class="szegely">
      <h1 style="color:red">Címsor</h1>
    </div>
  </body>
</html>
```

Címsor



Rangsor



- Ellentmondó definíciók esetén melyik fog érvényesülni? => **súly** hozzárendelése
- **Rangsor felállítás szabályai**
 - ▣ Kérdéses elem/tulajdonság párosra alkalmazott összes deklaráció előkeresése
 - A kérdéses elem szelektorként szerepel?
 - Igen: OK
 - Nem: Van örökölt érték?
 - Igen: OK
 - Nem: Alaptulajdonságok jut(hat)nak érvényre

Rangsor



- **Meghatározás sorrendje szerint**
 - ▣ Későbbi deklaráció > korábbi deklaráció

```
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01
Transitional//EN"
"http://www.w3.org/TR/html4/loose.dtd">
<html>
  <head>
    <title></title>
    <style type="text/css">
      h1 {color: blue}
      h1 {color: green}
    </style>
  </head>

  <body>
    <h1>Ez egy 1-es címsor</h1>
  </body>
</html>
```

Ez egy 1-es címsor

Rangsor



□ explicit súly szerinti rendezés

- ▣ ! Important > normál

```
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN"
"http://www.w3.org/TR/html4/loose.dtd">
<html>
<head>
<title></title>
<style type="text/css">
body {background-color: #c0c0c0}
h1 {color: black ! important; background: white ! important }
h1 {color: blue}
</style>
</head>
<body>
<h1>Ez egy 1-es cím sor</h1>
</body>
</html>
```

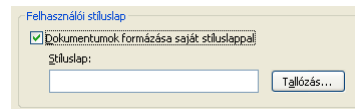
Ez egy 1-es cím sor

Rangsor



□ Eredet szerinti rendezés

- ▣ Szerző definíciói > Olvasó definíciói (felhasználói stíluslap) > Kliens program beállításai
- ▣ Mi van, ha olvasóként felül akarom bíráltni a szerző definícióit => **! Important** attribútum használata



- ▣ Példa
 - a:link {color: green ! important}
- ▣ Mi van ha a szerző is használta?
 - Szerző (important) > olvasó (important) [CSS1]
 - Szerző (important) < olvasó (important) [CSS2+]

Rangsor



□ Szelektor egyedisége szerint

- ▣ a: Ha a style attribútum lett használva, akkor 1-es az értéke.
- ▣ b: ID attribútumok száma
- ▣ c: Egyéb attribútumok száma (osztály, pszeudó osztály)
- ▣ d: Elemek és pszeudó elemek száma
 - A abcd számokat össze kell fűzni, így megkapjuk az egyediséget

Rangsor



*	/* a=0 b=0 c=0 d=0 => egyediség = 0 */
li {...}	/* a=0 b=0 c=0 d=1 => egyediség = 1 */
li:first-letter {...}	/* a=0 b=0 c=0 d=2 => egyediség = 2 */
ul li {...}	/* a=0 b=0 c=0 d=2 => egyediség = 2 */
ol ul li {...}	/* a=0 b=0 c=0 d=3 => egyediség = 3 */
ol ul+li {...}	/* a=0 b=0 c=0 d=3 => egyediség = 3 */
li.red {...}	/* a=0 b=0 c=1 d=1 => egyediség = 11 */
li.red.level {...}	/* a=0 b=0 c=2 d=1 => egyediség = 21 */
ol ul li.red {...}	/* a=0 b=0 c=1 d=3 => egyediség = 13 */
#x34y {...}	/* a=0 b=1 c=0 d=0 => egyediség = 100 */
style=""	/* a=1 b=0 c=0 d=0 => egyediség = 1000 */

Kiértékelési sorrend



1. Összes deklaráció megkeresése a cél média-típus esetén. A deklaráció akkor jut érvényre, ha az adott szelektor illeszkedik a kérdéses elemre, és a cél média-típus illeszkedik a megadott média listára (@media szabályok, és hivatkozott stíluslapok)
2. Rangsorolás a fontosság (normál, fontos) és eredet (szerző, felhasználó, kliens program) szerint. Ennek sorrendje:
 1. Kliens program deklarációi
 2. Felhasználó normál deklarációi
 3. Szerző normál deklarációi
 4. Szerző fontos deklarációi
 5. Felhasználó fontos deklarációi
3. Az azonos fontosságú és eredetű szabályok rangsorolása, a specifikusabb szelektorok felülírják a kevésbé specifikusakat. A pszeudó-elemek és pszeudó-osztályok normál elemként és osztályként vannak figyelembe véve.
4. Meghatározás sorrendje: ha két deklaráció azonos súlyú, eredetű, specifikusságú, akkor a későbbi deklarált nyer. Az importált stíluslapokban való deklarációk megelőzik a stíluslap többi deklarációját.

62

Médiatípusok



Használható médiatípusok



- All: összes eszköz
- Aural/speech: beszédszintetizátor
- Braille: braille kijelzők számára
- Embossed: braille nyomtató
- Handheld: kézi eszközök (kis képernyő)
- Print: nyomtatás
- Projection: kivetítő
- Screen: színes képernyők
- Tty: fix szélességű karakteres megjelenítők
- Tv: televízió típusú eszközök

■ Alacsonyabb felbontás, limitált gördíthetőség, hang rendelkezésre áll...

Médiafüggő stíluslap megadása



- 1. módszer
 - Cél média megadása `@media` vagy `@import` szabályokkal.
 - `@import url("fancyfonts.css") screen;`
 - `@media print { /* nyomtatásra vonatkozó def. */ }`
- 2. módszer
 - A cél médium megadása a dokumentumleíró nyelvben
 - `<LINK rel="stylesheet" type="text/css" media="print, handheld" href="nyomt.css">`

A @media szabály - példa



```
<!DOCTYPE html>
<html>
<head>
<meta charset="utf-8">
<title></title>
<style type="text/css">
@media screen
{
  body {
    background-color:darkblue;
    color:white;
    font-family:sans-serif;
  }
}

@media print
{
  body {
    background-color:white;
    color:black;
    font-family:serif;
  }
  div#menu {display:none} /* ne látszódjon */
}
</style>
</head>
<body>
<div id="menu">Menüpont1  Menüpont2 </div>
<h1>Címsor 1</h1>
<p>Szöveg</p>
</body>
</html>
```

Menüpont1 Menüpont2

Címsor 1

Szöveg

Megjelenés a képernyőn

Címsor 1

Szöveg

Megjelenés nyomtatásban
(nem látszik a menü,
talpas betűtípus, más színek)

Média csoportok (CSS 2.1)



Média típusok	Média csoportok			
	continuous/ paged	visual/aural/ tactile	grid/ bitmap	interactive/ static
braille	continuous	tactile	grid	both
embossed	paged	tactile	grid	static
handheld	both	visual , audio, speech	both	both
print	paged	visual	bitmap	static
projection	paged	visual	bitmap	interactive
speech	continuous	aural	N/A	both
screen	continuous	visual, audio	bitmap	both
tty	continuous	visual	grid	both
tv	both	visual, audio	bitmap	both

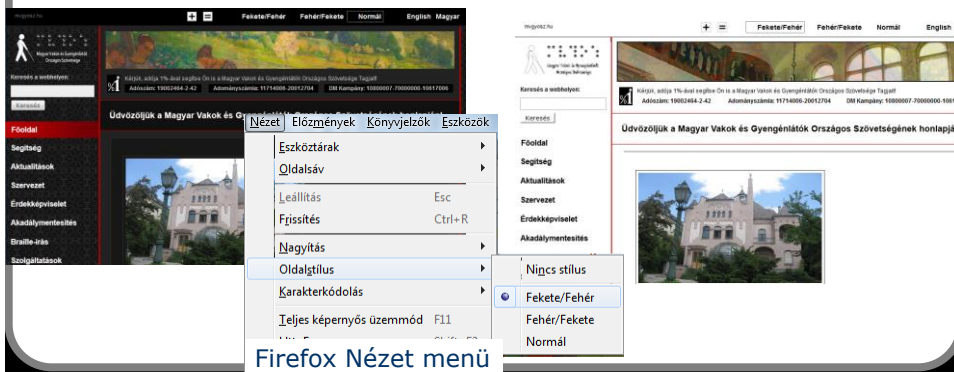
Vagyis ez is használható: `@media paged {}`
`@media interactive {}`

<http://www.w3.org/TR/CSS21/media.html>

Alternatív stíluslapok megadása



- `<link rel="stylesheet" type="text/css" href="uri" title="Normál" media="screen" >`
- `<link rel="alternate stylesheet" type="text/css" href="uri" title="Fekete/fehér" media="screen" >`



Firefox Nézet menü

Váltás a stílusok között



- Javascript segítségével is válthatunk stílust
- Az alábbi megoldásban cookie tárolja el a választást.
- <http://alistapart.com/article/alternate>





CSS 1 és CSS 2 szelektorok



Minta	Jelentés	
*	Megfelel bármelyik elemnek.	CSS 2
E	Egyezik bármely E elemmel (bármely E típusú elemmel).	
E F	Egyezik bármely F elemmel, ami egy E elem leszármazottja.	
E > F	Egyezik bármely F elemmel, ami egy E elem gyermek-eleme	CSS 2
E:first-child	Megegyezik az E elemmel, ha az E a szülő elemének első gyermeke.	CSS 2
E:link E:visited	Megegyezik az E elemmel, ha az E elem egy olyan link, amelynek célját még nem látogatták (:link), vagy már látogatták (:visited).	
E:active E:hover E:focus	Megegyezik az E elemmel, bizonyos felhasználói beavatkozások alatt.	CSS 2
E:lang(c)	Megegyezik az E típusú elemmel, ha az a (beszélt) c nyelven íródott.	CSS 2

CSS 1 és CSS 2 szelektorok



Minta	Jelentés	CSS
E + F	Megegyezik bármely F elemmel, amelyet közvetlenül megelőz egy E elem.	2.
E[valami]	Megegyezik bármely E elemmel, amelynek "valami" attribútuma be van állítva (tekintet nélkül annak értékére).	2.
E[valami="proba"]	Megegyezik azzal az E elemmel, amelynek "valami" attribútumának értéke pontosan "proba".	2.
E[valami~="proba"]	Megegyezik azzal az E elemmel, amelynek "valami" attribútumának értékei szóközzel elválasztott értékek, és egyik közülük pontosan "proba".	2.
E[lang="en"]	Megegyezik azzal az E elemmel, amelynek "lang" attribútuma pontosan az „en”, vagy „en”-nel kezdődik és kötőjel követi.	2.
DIV.warning	Csak HTML. Ugyanaz, mint a DIV[class~="warning"].	
E#azonosito	Megegyezik azzal az E elemmel, amelynek ID-je "azonosito".	

* szelektor



```
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML
4.01 Transitional//EN"
"http://www.w3.org/TR/html4/loose.dtd">
<html>
<head>
<title></title>
<style type="text/css">
* { margin:0; /* margó lenullázása */
padding:0; /* kitöltés lenullázása */
}
p {margin:10px; padding:10px;}
</style>
</head>
<body>
<p>Lorem ipsum dolor ... </p>
<p>Nullam a posuere turpis...</p>
</body>
</html>
```

A * szelektort leggyakrabban arra használjuk, hogy minden elem margóját és kitöltését lenullázzuk. Ezen értékeket aztán minden általunk használt elem esetén újra beállítjuk, így egységesek lesznek a margók, nem az adott böngésző alapbeállításán múlnak.

Lorem ipsum dolor sit amet, consectetur adipiscing elit. Quisque turpis lacus, viverra vitae sodales vel, iaculis at nunc. Ut lobortis dapibus leo sed semper. Vestibulum pellentesque eros at risus volutpat at interdum nunc aliquet. Praesent lacus leo, vulputate sed sagittis et, vulputate sit amet nibh. Praesent quis neque id arcu accumsan imperdiet. Duis placerat mauris id quam eleifend elementum. Pellentesque tincidunt auctor mattis. Maecenas sed ornare risus.

Nullam a posuere turpis. Ut placerat viverra eros at condimentum. Sed nisl ipsum, commodo id hendrerit ac, vestibulum sed erat. Aliquam feugiat fermentum justo, dapibus porta odio interdum ac. Nullam facilisis pretium leo eget fringilla. In hac habitasse platea dictumst. Nulla facilisi. In id leo nec diam sodales egestas. Vestibulum facilisis sem at mi consequat gravida. Maecenas accumsan, magna in tincidunt bibendum, nisi odio molestie velit, sit amet tincidunt ligula magna porttitor arcu. Curabitur hendrerit libero id nisi dictum sed pretium elit tempor. Sed ac risus arcu. Integer condimentum erat sit amet est scelerisque cursus.

Amíg nem volt * szelektor



- Kellett egy külön stíluslap, amelyben az összes elem margó,kitöltés, stb. értéke be volt állítva...

- <http://meyerweb.com/eric/tools/css/reset/>

```
/* http://meyerweb.com/eric/tools/css/reset/
v2.0 | 20110126
License: none (public domain)
*/

html, body, div, span, applet, object, iframe,
h1, h2, h3, h4, h5, h6, p, blockquote, pre,
a, abbr, acronym, address, big, cite, code,
del, dfn, em, img, ins, kbd, q, s, samp,
small, strike, strong, sub, sup, tt, var,
b, u, i, center,
dl, dt, dd, ol, ul, li,
fieldset, form, label, legend,
table, caption, tbody, tfoot, thead, tr, th, td,
article, aside, canvas, details, embed,
figure, figcaption, footer, header, hgroup,
menu, nav, output, ruby, section, summary,
time, mark, audio, video {
    margin: 0;
    padding: 0;
    border: 0;
    font-size: 100%;
    font: inherit;
    vertical-align: baseline;
}
/* HTML5 display-role reset for older browsers */
article, aside, details, figcaption, figure,
footer, header, hgroup, menu, nav, section {
    display: block;
}
body {
    line-height: 1;
```

Leszármazás (E F)



```
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01
Transitional//EN"
"http://www.w3.org/TR/html4/loose.dtd">
<html>
<head>
<title></title>
<style type="text/css">
em {color:blue}
h1 {color:blue}
/* ha az em elem a h1 leszármazottja
Legyen zöld színnel megjelenítve*/
h1 em {color:green;}
</style>
</head>
<body>
<h1>CSS, avagy <em>lépcsőzetes
stíluslapok</em></h1>
<p>A CSS (angolul Cascading Style Sheets) a
számítástechnikában egy <em>stílusleíró
nyelv</em>, mely a HTML vagy XHTML típusú
strukturált dokumentumok megjelenését írja le.
Ezen kívül használható bármilyen XML alapú
dokumentum stílusának leírására is, mint
például az SVG, XUL stb.</p>
</body>
</html>
```

CSS, avagy lépcsőzetes stíluslapok

A CSS (angolul Cascading Style Sheets) a számítástechnikában egy *stílusleíró nyelv*, mely a HTML vagy XHTML típusú strukturált dokumentumok megjelenését írja le. Ezen kívül használható bármilyen XML alapú dokumentum stílusának leírására is, mint például az SVG, XUL stb.

Leszármazás (E F)



```
<ul id="pelda">
  <li>1. elem
  <ul>
    <li>1. elem 1. alelem</li>
    <li>1. elem 2. alelem
      <ul>
        <li>1. elem 2. alelem 1. al-alelem</li>
        <li>1. elem 2. alelem 2. al-alelem</li>
      </ul>
    </li>
  </ul>
</li>
</li>
<li>2. elem
  <ul>
    <li>2. elem 1. alelem</li>
    <li>2. elem 2. alelem
      <ul>
        <li>2. elem 2. alelem 1. al-alelem</li>
        <li>2. elem 2. alelem 2. al-alelem</li>
      </ul>
    </li>
    <li>2. elem 3. alelem</li>
  </ul>
</li>
<li>3. elem</li>
</ul>
```

CSS:

ul#pelda li {border:1px solid red;}

- 1. elem
 - 1. elem 1. alelem
 - 1. elem 2. alelem
 - 1. elem 2. alelem 1. al-alelem
 - 1. elem 2. alelem 2. al-alelem
- 2. elem
 - 2. elem 1. alelem
 - 2. elem 2. alelem
 - 2. elem 2. alelem 1. al-alelem
 - 2. elem 2. alelem 2. al-alelem
 - 2. elem 3. alelem
- 3. elem

Minden elemben lévő elem szegélyt kap.

Gyerek elem (E>F)



```
<ul id="pelda">
  <li>1. elem
  <ul>
    <li>1. elem 1. alelem</li>
    <li>1. elem 2. alelem
      <ul>
        <li>1. elem 2. alelem 1. al-alelem</li>
        <li>1. elem 2. alelem 2. al-alelem</li>
      </ul>
    </li>
  </ul>
</li>
</li>
<li>2. elem
  <ul>
    <li>2. elem 1. alelem</li>
    <li>2. elem 2. alelem
      <ul>
        <li>2. elem 2. alelem 1. al-alelem</li>
        <li>2. elem 2. alelem 2. al-alelem</li>
      </ul>
    </li>
    <li>2. elem 3. alelem</li>
  </ul>
</li>
<li>3. elem</li>
</ul>
```

CSS:

ul#pelda > li {border:1px solid red;}

- 1. elem
 - 1. elem 1. alelem
 - 1. elem 2. alelem
 - 1. elem 2. alelem 1. al-alelem
 - 1. elem 2. alelem 2. al-alelem
- 2. elem
 - 2. elem 1. alelem
 - 2. elem 2. alelem
 - 2. elem 2. alelem 1. al-alelem
 - 2. elem 2. alelem 2. al-alelem
 - 2. elem 3. alelem
- 3. elem

Csak azon listaelem kap szegélyt, ami gyerekeleme a külső elemnek.

Első gyerek (:first-child)



```
<ul id="pelda">
  <li>1. elem
  <ul>
    <li>1. elem 1. alelem</li>
    <li>1. elem 2. alelem
      <ul>
        <li>1. elem 2. alelem 1. al-alelem</li>
        <li>1. elem 2. alelem 2. al-alelem</li>
      </ul>
    </li>
  </ul>
</li>
<li>2. elem
  <ul>
    <li>2. elem 1. alelem</li>
    <li>2. elem 2. alelem
      <ul>
        <li>2. elem 2. alelem 1. al-alelem</li>
        <li>2. elem 2. alelem 2. al-alelem</li>
      </ul>
    </li>
    <li>2. elem 3. alelem</li>
  </ul>
</li>
<li>3. elem</li>
</ul>
```

CSS:

ul#pelda > li:first-child {border:1px solid red;}

- 1. elem
 - 1. elem 1. alelem
 - 1. elem 2. alelem
 - 1. elem 2. alelem 1. al-alelem
 - 1. elem 2. alelem 2. al-alelem
- 2. elem
 - 2. elem 1. alelem
 - 2. elem 2. alelem
 - 2. elem 2. alelem 1. al-alelem
 - 2. elem 2. alelem 2. al-alelem
 - 2. elem 3. alelem
- 3. elem

Csak azon listaelem kap szegélyt, ami gyerekeleme a külső elemnek, és maga első gyermeknek számít.

Első gyerek (:first-child)



```
<ul id="pelda">
  <li>1. elem
  <ul>
    <li>1. elem 1. alelem</li>
    <li>1. elem 2. alelem
      <ul>
        <li>1. elem 2. alelem 1. al-alelem</li>
        <li>1. elem 2. alelem 2. al-alelem</li>
      </ul>
    </li>
  </ul>
</li>
<li>2. elem
  <ul>
    <li>2. elem 1. alelem</li>
    <li>2. elem 2. alelem
      <ul>
        <li>2. elem 2. alelem 1. al-alelem</li>
        <li>2. elem 2. alelem 2. al-alelem</li>
      </ul>
    </li>
    <li>2. elem 3. alelem</li>
  </ul>
</li>
<li>3. elem</li>
</ul>
```

CSS:

ul#pelda li:first-child {border:1px solid red;}

- 1. elem
 - 1. elem 1. alelem
 - 1. elem 2. alelem
 - 1. elem 2. alelem 1. al-alelem
 - 1. elem 2. alelem 2. al-alelem
- 2. elem
 - 2. elem 1. alelem
 - 2. elem 2. alelem
 - 2. elem 2. alelem 1. al-alelem
 - 2. elem 2. alelem 2. al-alelem
 - 2. elem 3. alelem
- 3. elem

Itt minden olyan listaelem szegélyt kap, ami első gyerek.

Nyelvfüggő stílus (:lang)



```
<!DOCTYPE html>
<html>
<head>
<title></title>
<style type="text/css">
  p:lang(en) {color:#0000FF;}
  p:lang(hu) {color:#FF0000; margin-left:20px;}
</style>
</head>
<body>
<p lang="en">Friend: a person who can turn a horrible day
into an absolutely perfect one.</p>
<p lang="hu">Barát az, aki egy rettenetes napból is
tökéleteset tud varázsolni.</p>
<p lang="en">If you can't laugh at yourself, friend, i'd
be glad to do it for you.</p>
<p lang="hu">Ha nem tudsz magadon nevetni, barátom, én
örömmel megteszem helyetted. </p>
</body>
</html>
```

Friend: a person who can turn a horrible day into an absolutely perfect one.

Barát az, aki egy rettenetes napból is tökéleteset tud varázsolni.

If you can't laugh at yourself, friend, i'd be glad to do it for you.

Ha nem tudsz magadon nevetni, barátom, én örömmel megteszem helyetted.

Közvetlen megelőzés (E+F)



```
<!DOCTYPE html>
<html>
<head>
<title></title>
<style type="text/css">
  p {text-indent:20px;}
  h1+p {text-indent:0;}
</style>
</head>
<body>
<h1>Címsor</h1>
<p>Lorem ipsum dolor sit amet...</p>
<p>Lorem ipsum dolor sit amet...</p>
<p>Lorem ipsum dolor sit amet...</p>
<p>Lorem ipsum dolor sit amet...</p>
</body>
</html>
```

Címsor

Lorem ipsum dolor sit amet, consectetur adipiscing elit. Mauris viverra sem ut purus vehicula eget lobortis mi luctus. Aliquam at luctus lectus.

Lorem ipsum dolor sit amet, consectetur adipiscing elit. Mauris viverra sem ut purus vehicula eget lobortis mi luctus. Aliquam at luctus lectus.

Lorem ipsum dolor sit amet, consectetur adipiscing elit. Mauris viverra sem ut purus vehicula eget lobortis mi luctus. Aliquam at luctus lectus.

Lorem ipsum dolor sit amet, consectetur adipiscing elit. Mauris viverra sem ut purus vehicula eget lobortis mi luctus. Aliquam at luctus lectus.

Azon bekezdések első sora nincs behúzva,
amelyek közvetlenül 1-es címsor után következnek.

Attribútum szelektor (paraméter)



```
<!DOCTYPE html>
<html>
<head>
<title></title>
<style type="text/css">
  img[usemap] {border:2px solid red;}
</style>
</head>
<body>
<div>



</div>
</body>
</html>
```



Ahol a usemap paraméter szerepelt, ott piros szegélyt kapott a kép.

Attribútum szelektor (paraméter=„érték”)



```
<!DOCTYPE html>
<html>
<head>
<title></title>
<style type="text/css">
  input[type="password"]
  {
    background-color: yellow;
  }
</style>
</head>
<body>
<form action="#">
  <label>Név</label>
  <input type="text" >
  <label>Jelszó</label>
  <input type="password" >
</form>
</body>
</html>
```

Név Jelszó

A jelszó (password) típusú input mező háttere sárgára állított.
A szöveg (text) típusú mező háttere nem változott.

Attribútum szelektor (paraméter~="érték")



```
<!DOCTYPE html>
<html>
<head>
<title></title>
<style type="text/css">
  a[title~="ELTE"] {background-color : yellow ;}
</style>
</head>
<body>
<ul>
<li><a href="http://www.inf.elte.hu" title="ELTE Informatikai Kar">ELTE IK</a></li>
<li><a href="http://www.w3c.hu" title="W3C konzorcium">W3C</a></li>
<li><a href="http://www.elte.hu" title="Az ELTE weblapja">ELTE</a></li>
</ul>
</body>
</html>
```

- ELTE IK
- W3C
- ELTE

Csak azon linkek háttere lesz sárga, amelyek title attribútumában szerepel az ELTE szó.

E[lang | ="en"] típusú szelektor



```
<!DOCTYPE html>
<html>
<head>
<title></title>
<style type="text/css">
  p[lang|"en"] {color:blue;}
</style>
</head>
<body>
  <p lang="hu">A Clash egy közismert együttes.</p>
  <p lang="en-gb">The Clash are a well-known band.</p>
  <p lang="en-us">The Clash is a well-known band.</p>
</body>
</html>
```

A Clash egy közismert együttes.

The Clash are a well-known band.

The Clash is a well-known band.

Mindegy, hogy amerikai, vagy britt angol nyelvjárás szerint írtunk, kék lesz a szöveg színe.

CSS 2.1 szelektorok böngésző támogatottsága



Selector or declaration	IE 5-5	IE 6	IE 7	IE 8	IE 9	IE 10	FF 1.0 Win	FF 1.0.2 Mac	Saf 5.1 Win	Saf 5.1.2 Mac	Chrome 10 Win	Chrome 17 Mac	Opera 11.61 Win	Opera 11.61 Mac
* selector Selects all elements		yes					yes		yes		yes		yes	
> selector Selects an element that is a child of another element	no		yes				yes		yes		yes		yes	
+ selector Selects an element that is a next sibling of another element	no	static	almost	yes			yes		yes		yes		yes	
Static The browser doesn't correctly update the styles of the original element when another element is placed before it. <ul style="list-style-type: none"> IE8 updates correctly, but only after you remove the focus from the test link. 														
[attr] selector Selects an element with a certain attribute	no		yes				yes		yes		yes		yes	
Multiple classes Whether an element can support two or more class names	buggy		yes				yes		yes		yes		yes	
<ul style="list-style-type: none"> IE6 and lower make a mistake; when confronted with <code>p.test.small</code> they act as if the selector is <code>p.small</code>. 														

<http://www.quirksmode.org/css/contents.html>

CSS 2.1 szelektorok böngésző támogatottsága



Selector or declaration	IE 5-5	IE 6	IE 7	IE 8	IE 9	IE 10	FF 1.0 Win	FF 1.0.2 Mac	Saf 5.1 Win	Saf 5.1.2 Mac	Chrome 10 Win	Chrome 17 Mac	Opera 11.61 Win	Opera 11.61 Mac
:before and :after To generate content before and after an element	no		yes				yes		yes		yes		yes	
:hover and :active An element in a hovered (mouseover) or active (mousedown) state.	minimal	incomplete	almost				yes		yes		yes		yes	
<ul style="list-style-type: none"> IE 5/6 supports both only on links. IE 7 supports :hover, but not :active, on all elements. IE 8-10 (and maybe older ones, too) have a slight bug: clicking the mouse down on a nested element does not trigger :active. Try it in the test page by depressing the mouse button on one of the code examples. The :active styles do not kick in. 														
:first-child An element that is the first child of its parent	no	static	almost	yes			yes		yes		yes		yes	
Static Browser doesn't update the styles when elements are added dynamically. <ul style="list-style-type: none"> IE8 does the right thing only after you remove the focus from the test link. 														
:first-line and :first-letter The first line or first letter of an element		yes					yes		almost		almost		almost	
<ul style="list-style-type: none"> Safari, Chrome and Konqueror add an extra T when you do the dynamic test. Opera leaves out the T of the second and subsequent sentences when you do the dynamic test. 														
:focus Form fields that receive user focus	no		yes				yes		yes		yes		yes	

<http://www.quirksmode.org/css/contents.html>



CSS3 szelektorok

Minta	Jelentés
E ~ F	Olyan F elem, amelyet korábban megelőzött egy E elem. (hasonló mint E + F, de nem kell, hogy közvetlenül egymás után következzenek az elemek)
E[attr]	<p>p[testAttr^=foo] bármely elem, amely „testAttr” paraméterének van értéke, és az a „foo” karakterlánccal kezdődik.</p> <p>p[testAttr\$=foo] bármely elem, amely „testAttr” paraméterének van értéke, és az a „foo” karakterlánccal végződik.</p> <p>p[testAttr*=foo] bármely elem, amely „testAttr” paraméterének van értéke, és abban szerepel „foo” karakterlánc.</p>
:empty	Tartalom nélküli elemre illeszkedik
:enabled :disabled :checked	Enabled, disabled, checked állapotú (ürlap)elemekre illeszkedik
:first-of-type :last-of-type :only-of-type	<p>X:first-of-type olyan X elem, amely a szülőjének első gyereke</p> <p>X:last-of-type olyan X elem, amely a szülőjének utolsó gyereke</p> <p>X:only-of-type olyan X elem, amely a szülőjének egyetlen gyereke</p>
:last-child	Utolsó gyermekelem
:not	<p>Tagadás</p> <p>p#test *:not(em) {text-decoration: underline;} A test azonosítójú bekezdés leszármazottai, kivéve az em elemeket.</p>

89

Néhány példa



Néhány példa



Kódrészlet

```
<div id="content">
  <p>Lorem ipsum dolor sit amet, consectetur adipiscing elit.</p>
  <div id="text1">
    <p>Lorem ipsum dolor sit amet, consectetur adipiscing elit. </p>
  </div>
  <div id="text2">
    <p>Lorem ipsum dolor sit amet, consectetur adipiscing elit. </p>
    <div>
      <p>Lorem ipsum dolor sit amet, consectetur adipiscing elit. </p>
    </div>
  </div>
</div>
```

CSS

```
div#content >:first-child {color:red;}
```

Lorem ipsum dolor sit amet,
consectetur adipiscing elit.

Lorem ipsum dolor sit amet,
consectetur adipiscing elit.

Lorem ipsum dolor sit amet,
consectetur adipiscing elit.

Lorem ipsum dolor sit amet,
consectetur adipiscing elit.

:first-child



Kódrészlet

```
<div id="content">
  <p>Lorem ipsum dolor sit amet, consectetur adipiscing elit.</p>
  <div id="text1">
    <p>Lorem ipsum dolor sit amet, consectetur adipiscing elit. </p>
  </div>
  <div id="text2">
    <p>Lorem ipsum dolor sit amet, consectetur adipiscing elit. </p>
    <div>
      <p>Lorem ipsum dolor sit amet, consectetur adipiscing elit. </p>
    </div>
  </div>
</div>
</div>
```

CSS

```
div#content >div:first-child {color:red;}
```

Lorem ipsum dolor sit amet,
consectetur adipiscing elit.

Lorem ipsum dolor sit amet,
consectetur adipiscing elit.

Lorem ipsum dolor sit amet,
consectetur adipiscing elit.

Lorem ipsum dolor sit amet,
consectetur adipiscing elit.

:first-of-type



Kódrészlet

```
<div id="content">
  <p>Lorem ipsum dolor sit amet, consectetur adipiscing elit.</p>
  <div id="text1">
    <p>Lorem ipsum dolor sit amet, consectetur adipiscing elit. </p>
  </div>
  <div id="text2">
    <p>Lorem ipsum dolor sit amet, consectetur adipiscing elit. </p>
    <div>
      <p>Lorem ipsum dolor sit amet, consectetur adipiscing elit. </p>
    </div>
  </div>
</div>
</div>
```

CSS

```
div#content >div:first-of-type {color:red;}
```

Lorem ipsum dolor sit amet,
consectetur adipiscing elit.

Lorem ipsum dolor sit amet,
consectetur adipiscing elit.

Lorem ipsum dolor sit amet,
consectetur adipiscing elit.

Lorem ipsum dolor sit amet,
consectetur adipiscing elit.

:only-of-type



Kódrészlet

```
<div id="content">
  <p>Lorem ipsum dolor sit amet, consectetur adipiscing elit.</p>
  <div id="text1">
    <p>Lorem ipsum dolor sit amet, consectetur adipiscing elit. </p>
  </div>
  <div id="text2">
    <p>Lorem ipsum dolor sit amet, consectetur adipiscing elit. </p>
    <div>
      <p>Lorem ipsum dolor sit amet, consectetur adipiscing elit. </p>
    </div>
  </div>
</div>
```

Lorem ipsum dolor sit amet,
consectetur adipiscing elit.

Lorem ipsum dolor sit amet,
consectetur adipiscing elit.

Lorem ipsum dolor sit amet,
consectetur adipiscing elit.

Lorem ipsum dolor sit amet,
consectetur adipiscing elit.

CSS

div#content > :only-of-type {color:red;}

Selector or declaration	IE 5.5	IE 6	IE 7	IE 8	IE 9	IE 10 pr2	FF 11.0 Win	FF 10.0.3 Mac	Saf 5.1 Win	Saf 5.1.3 Mac	Chrome 18 Win	Chrome 17 Mac	Opera 11.61 Win	Opera 11.61 Mac
~ selector	no		yes				yes	yes	yes	yes	yes	yes	yes	
Selects an element that is a general next sibling of another element														
Advanced [attr] selectors	no		yes				yes	yes	yes	yes	yes	yes	yes	
Selects an element with an attribute whose value starts with, ends with, or contains a certain string.														
:empty	no		yes				yes	yes	yes	yes	yes	yes	yes	
Empty elements														
:enabled, :disabled and :checked	no		yes				yes	yes	yes	yes	yes	yes	yes	
For enabled, disabled, or checked form fields.														
:first-of-type	no		yes				yes	yes	yes	yes	yes	yes	yes	
The first element of its type.														
:last-child	no		yes				yes	yes	yes	yes	yes	yes	yes	
An element that is the last child of its parent														
:last-of-type	no		yes				yes	yes	yes	yes	yes	yes	yes	
The last element of its type.														
:not	no		yes				yes	yes	yes	yes	yes	yes	yes	
Negation of a selector														
Selector or declaration	IE 5.5	IE 6	IE 7	IE 8	IE 9	IE 10 pr2	FF 11.0 Win	FF 10.0.3 Mac	Saf 5.1 Win	Saf 5.1.3 Mac	Chrome 18 Win	Chrome 17 Mac	Opera 11.61 Win	Opera 11.61 Mac

<http://www.quirksmode.org/css/contents.html>



Minta	Jelentés
:nth-child()	Egy formulával meghatározható sorsszámú gyerek <code>div#test > *:nth-child(3n+1) { color: red; }</code>
:nth-of-type()	Egy formulával meghatározható sorsszámú, adott típusú gyerek <code>div#test div:nth-of-type(5n+2) { background-color: #9999ff; }</code>
:nth-last-child() :nth-last-of-type()	Hasonló a fentiekhez, de az utolsó elemre vonatkozik. (A sorsszám az utolsótól számítódik)
:only-child	Olyan elem, amely a szülő elem egyetlen gyermeke
:root	Minden blokk gyökéreleme (HTML-ben ez a <html> elem lesz.)
::selection	A felhasználó által kijelölt szöveg
:target	Azon elem, ami link célként van beállítva (egyedi neve van és mutat rá link a #egyedinév formátumban)

:nth-child() – valahányadik gyerek



```

...
<style type="text/css">
td {
    padding:10px;
    border:1px solid blue;
}
table {
    border:1px solid green;
    width:250px;
    border-collapse:collapse;
}
tr:nth-child(3n+1) {
    background-color:lightgreen;
}
tr:nth-child(3n+2) {
    background-color:lightblue;
}
</style>
...
<body>
<table>
<tr><td>1</td><td>2</td></tr>
<tr><td>3</td><td>4</td></tr>
<tr><td>5</td><td>6</td></tr>
<tr><td>7</td><td>8</td></tr>
<tr><td>9</td><td>10</td></tr>
</table>
</body>
</html>

```

1	2
3	4
5	6
7	8
9	10

A táblázatsorok közül a 3-mal oszthatók az alap (fehér) háttérrel kapják, a következő világoszöld, a rákövetkező világoskék lesz.

Selector or declaration	IE 5.4	IE 6	IE 7	IE 8	IE 9	IE 10	FF 11.0 Win	FF 10.0.2 Mac	Saf 5.1 Win	Saf 5.1.2 Mac	Chrome 18 Win	Chrome 17 Mac	show page-contents Opera 11.61 Win	Opera 11.61 Mac
:nth-child() Select elements according to a formula	no	yes	yes	yes	yes	yes	yes	yes	yes	yes	yes	yes	incorrect	
• Opera gets totally confused in the dynamic test.														
:nth-last-child()	no	yes	yes	yes	yes	yes	yes	yes	yes	yes	yes	yes	yes	
:nth-last-of-type()	no	yes	yes	yes	yes	yes	yes	yes	yes	yes	yes	yes	yes	
:nth-of-type() Select elements according to a formula	no	yes	yes	yes	yes	yes	yes	yes	yes	yes	yes	yes	incorrect	
• Opera gets totally confused in the dynamic test.														
:only-child An element that is the only child of its parent	no	yes	yes	yes	yes	yes	yes	yes	yes	yes	yes	yes	yes	
:only-of-type The only element of its type.	no	yes	yes	yes	yes	yes	yes	yes	yes	yes	yes	yes	yes	
:root The root element, or initial containing block.	no	yes	yes	yes	yes	yes	yes	yes	yes	yes	yes	yes	yes	
::selection Text selected by the user.	no	yes					-moz-		yes	yes	yes	yes	yes	
:target	no	incomplete					yes		yes	yes	yes	yes	yes	

<http://www.quirksmode.org/css/contents.html>

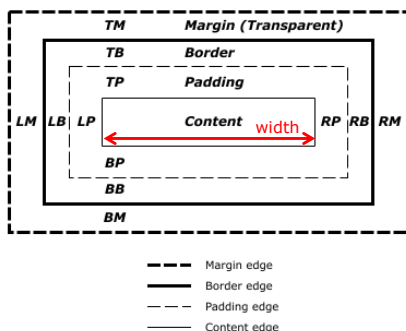
97

98

Formázásmodell (dobozmodell)



Formázásmodell



Margin: margó
Border: szegély
Padding: kitöltés
Content: tartalom

T,R,B,L: felső, jobb, alsó, bal

A tartalom (content) mérete a megjelenített tartalomtól függ, de beállítható a width és height (szélesség és magasság) paraméterekkel.

Szegélyek



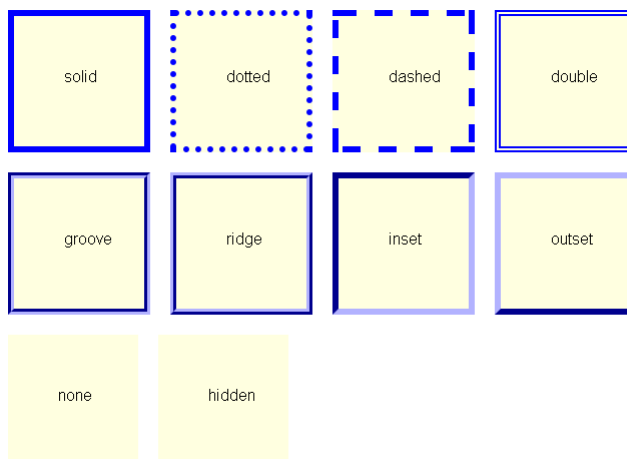
- Mind a négy oldali szegélyre vonatkozik az alábbi:
 - ▣ Border-style: stílus [none, hidden, dotted, dashed, solid, double, groove, ridge, inset, outset]
 - ▣ Border-width: vastagság [thin, medium, thick, *méret*]
 - ▣ Border-color: szín vagy transparent (átlátszó)
- Kompakt megadás mind a négy oldalra szimmetrikusan:
 - ▣ border: 1px solid green;

Szegélyek



- Külön is megadható a négy oldalnak
 - ▣ `border-top-color`; `border-right-color`; `border-bottom-color`; `border-left-color`;
 - ▣ `border-top-style`; `border-right-style`; `border-bottom-style`; `border-left-style`;
 - ▣ `border-top-width`; `border-right-width`; `border-bottom-width`; `border-left-width`;

Szegély stílusok



Mi a különbség a hidden és none között? Speciális esetekben (pl. táblázatok formázásánál) szükséges.

Táblázatok szegélye



HTML kód:

```
<table>
  <tr>
    <td>1</td>
    <td>2</td>
  </tr>
  <tr>
    <td>3</td>
    <td>4</td>
  </tr>
</table>
```

A border-collapse paraméterrel
összevonható a szegély.
Értékei: collapse | separate | inherit

CSS:

```
table {border:1px solid blue;}
td {padding:10px;}
```

1	2
3	4

CSS:

```
table {border:1px solid blue;}
td {padding:10px;
  border:1px solid blue;}
```

1	2
3	4

CSS:

```
table {border:1px solid blue;
  border-spacing:0;}
td {padding:10px;
  border:1px solid blue;}
```

1	2
3	4

CSS:

```
table {border:1px solid blue;
  border-collapse:collapse;}
td {padding:10px;
  border:1px solid blue;}
```

1	2
3	4

Hidden vs None érték...



```
<table style="border-collapse:collapse; border:solid;">
  <tr>
    <td style="border-right:hidden; border-bottom:hidden">1</td>
    <td style="border:solid">2</td>
  </tr>
  <tr>
    <td style="border:none">3</td>
    <td style="border:solid">4</td>
  </tr>
</table>
```

1	2
3	4

```
<table style="border-collapse:collapse; border:solid;">
  <tr>
    <td style="border-right:none; border-bottom:none">1</td>
    <td style="border:solid">2</td>
  </tr>
  <tr>
    <td style="border:none">3</td>
    <td style="border:solid">4</td>
  </tr>
</table>
```

1	2
3	4

Üres cellák és a szegély



```
<table>
  <tr>
    <td>1</td>
    <td>2</td>
  </tr>
  <tr>
    <td>3</td>
    <td>4</td>
  </tr>
  <tr>
    <td></td>
    <td></td>
  </tr>
</table>
```

CSS:
td {padding:10px;
border:1px solid blue;}

1	2
3	4

CSS:
td {padding:10px;
border:1px solid blue;
empty-cells:hide;}

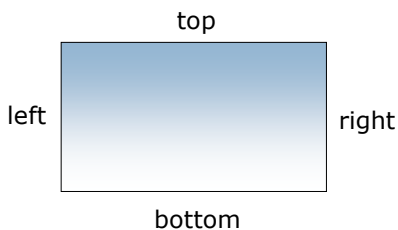
1	2
3	4

CSS:
td {padding:10px;
border:1px solid blue;
empty-cells:hide;}

1	2
3	4

table {border:1px solid green;}

Margó, kitöltés megadása



margin:
margin-top:
margin-right:
margin-bottom:
margin-left:

padding:
padding-top:
padding-right:
padding-bottom:
padding-left:

margin-top: 5px;
margin-right: 10px;
margin-bottom: 15px;
margin-left: 3px;



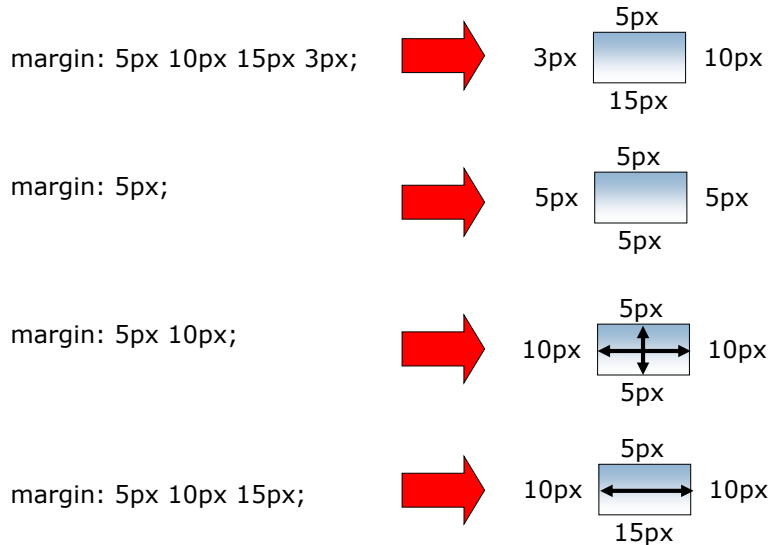
margin: 5px 10px 15px 3px;

padding-top: 5px;
padding-right: 10px;
padding-bottom: 15px;
padding-left: 3px;



padding: 5px 10px 15px 3px;

Margó megadása



Margó összevonás

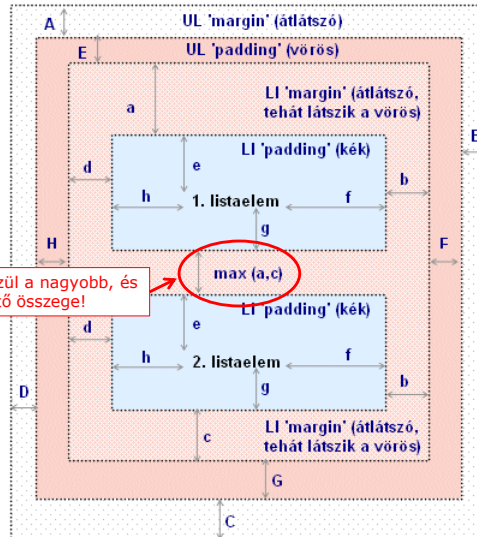


- A szomszédos vertikális margók összevonásra kerülhetnek bizonyos feltételek fennállása esetén.
- Bővebb info:
 - <http://www.w3.org/TR/CSS21/box.html>

Listaelemek margó összevonása



```
<style type="text/css">
ul {
  background: red;
  margin: A B C D;
  padding: E F G H;
}
li {
  color: white;
  background: blue;
  margin: a b c d;
  padding: e f g h;
}
</style>
...
<ul>
  <li>Első elem</li>
  <li>Második elem</li>
</ul>
```



Példa margin, padding értékre



p {border:1px solid blue;}

Lorem ipsum dolor sit amet, consectetur adipiscing elit.
Aenean in dolor a tellus feugiat molestie sed quis nisi.
Vivamus augue nisi, imperdiet id consequat id, luctus ac velit.

↑ Alapértelmezett margó

Nam id nisl vitae neque posuere interdum. Quisque pulvinar fermentum metus, at imperdiet purus accumsan blandit. Nunc bibendum, nibh vitae lobortis varius, erat eros elementum felis, a tempor nisi lectus at turpis. Praesent sodales mauris ac massa dignissim fermentum.

p {border:1px solid blue;
margin:0;padding:10px;}

Lorem ipsum dolor sit amet, consectetur adipiscing elit.
Aenean in dolor a tellus feugiat molestie sed quis nisi.
Vivamus augue nisi, imperdiet id consequat id, luctus ac velit.

Nam id nisl vitae neque posuere interdum. Quisque pulvinar fermentum metus, at imperdiet purus accumsan blandit. Nunc bibendum, nibh vitae lobortis varius, erat eros elementum felis, a tempor nisi lectus at turpis. Praesent sodales mauris ac massa dignissim fermentum.

p {border:1px solid blue; margin:0;}

Lorem ipsum dolor sit amet, consectetur adipiscing elit.
Aenean in dolor a tellus feugiat molestie sed quis nisi.
Vivamus augue nisi, imperdiet id consequat id, luctus ac velit.

Nam id nisl vitae neque posuere interdum. Quisque pulvinar fermentum metus, at imperdiet purus accumsan blandit. Nunc bibendum, nibh vitae lobortis varius, erat eros elementum felis, a tempor nisi lectus at turpis. Praesent sodales mauris ac massa dignissim fermentum.

p {border:1px solid blue;
padding:10px;margin:5px;}

Lorem ipsum dolor sit amet, consectetur adipiscing elit. Aenean in dolor a tellus feugiat molestie sed quis nisi. Vivamus augue nisi, imperdiet id consequat id, luctus ac velit.

Nam id nisl vitae neque posuere interdum. Quisque pulvinar fermentum metus, at imperdiet purus accumsan blandit. Nunc bibendum, nibh vitae lobortis varius, erat eros elementum felis, a tempor nisi lectus at turpis. Praesent sodales mauris ac massa dignissim fermentum.

Dobozok mérete



- Méret megadásnál
 - ▣ width: szélesség
 - ▣ height: magasság
 - ▣ Minimális és maximális méretek megadása
 - min-width: minimális szélesség
 - max-width: maximális szélesség
 - min-height: minimális magasság
 - max-height: maximális magasság

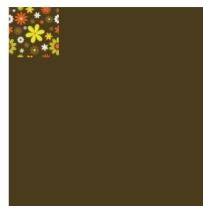
min/max-width/height	no	mini	mal	yes	yes	yes	yes	yes						
<ul style="list-style-type: none">IE 6 supports <code>min-height</code> on TD's only.														
Selector or declaration	IE 5.5	IE 6	IE 7	IE 8	IE 9	IE 10 prz	FF 11.0 Win	FF 10.0.2 Mac	Saf 5.1 Win	Saf 5.1.2 Mac	Chrome 18 Win	Chrome 17 Mac	Opera 11.61 Win	Opera 11.61 Mac

Háttérkép megadása



- CSS-ben nem csak mozaikszerű háttér elrendezésre van mód
- A háttérképet igazíthatjuk
 - ▣ [background-position](#)
 - ▣ vízszintesen (left, center, right, %, hossz) függőlegesen (top, center, bottom, %, hossz)
- Beállíthatjuk az ismétlődés módját
 - ▣ [background-repeat](#)
 - ▣ Nincs (no-repeat), mozaikszerű (repeat), vízszintesen (repeat-x), függőlegesen (repeat-y)
- Csatolás módja
 - ▣ [background-attachment](#)
 - ▣ Fix (fixed) vagy tartalommal gördíthető (scroll)

Példák



Egy **div** elem, beállított háttérszínnel. Ennél próbáljuk ki a különböző háttér elhelyezéseket.

```
background-image:url('hatter.jpg');
```

```
background-image:url('hatter.jpg');  
background-repeat:no-repeat;
```



```
background-image:url('hatter.jpg');  
background-repeat:repeat-x;
```

```
background-image:url('hatter.jpg');  
background-repeat:repeat-y;
```

```
background-image:url('hatter.jpg');  
background-repeat:no-repeat;  
background-position:center right;
```

Példák

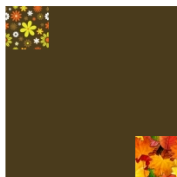


```
background-image:url('hatter.jpg');  
background-repeat:no-repeat;  
background-position:20px 80px;
```

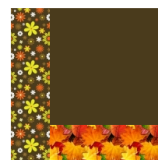
```
background-image:url('hatter.jpg');  
background-repeat:no-repeat;  
background-position:50% 50%;
```

```
background-image:url('hatter.jpg');  
background-repeat:repeat-x;  
background-position:50% 50%;
```

Több háttérkép megadása (CSS3)



```
background-image:url('hatter.jpg'), url('hatter2.jpg');
background-repeat:no-repeat, no-repeat;
background-position: left top, right bottom;
```



```
background-image:url('hatter.jpg'), url('hatter2.jpg');
background-repeat:repeat-y, repeat-x;
background-position: left top, right bottom;
```



```
background-image:url('hatter.jpg'), url('hatter2.jpg');
background-repeat:repeat-x, repeat;
background-position: left top, right bottom;
```

A többi gyakorlato megnézzük!

Kompatibilitás



Selector or declaration	IE 5.5	IE 6	IE 7	IE 8	IE 9	FF 3.0 Win	FF 3.0 Mac	Saf 5.1 Win	Saf 5.1 Mac	Chrome 10 Win	Chrome 10 Mac	Opera 10.5 Win	Opera 10.5 Mac
Multiple background images	no	no	yes	yes	yes	yes	yes	yes	yes	yes	yes	yes	yes
background-attachment: fixed	no	yes	yes	yes	yes	yes	yes	yes	yes	yes	yes	yes	yes
background-attachment: local	no	yes	no	yes	yes	yes	yes	yes	yes	yes	yes	yes	yes
background-repeat: repeat, repeat-x, repeat-y, no-repeat	yes	yes	yes	yes	yes	yes	yes	yes	yes	yes	yes	yes	yes
The traditional values													

<http://www.quirksmode.org/css/background.html>



Alapfogalmak



- A dokumentumfa elemei a doboz modellnek megfelelően jelennek meg, ahol fontos
 - a doboz mérete és a doboz típusa (blokk, inline),
 - a pozícionálási séma,
 - az elemek viszonya a dokumentumfában,
 - külső információk (viewport [képmező] mérete)

Blokkszintű elemek



- Blokkszintű elemek
 - ▣ Vizuálisan blokként jelennek meg (pl. bekezdés, lista, div)
 - ▣ A display tulajdonság következő értékei tartoznak ide: block, list-item, table
- A szabvány beszél még blokkszintű dobozokról, illetve blokk tartalmazó dobozról is...
- Bővebben:
<http://www.w3.org/TR/CSS2/visuren.html#block-boxes>

Soron belüli (inline) elemek



- Soron beüli elemek
 - ▣ Vizuálisan nem alkotnak blokkot, soron belül rendeződnek (pl. img, em, strong, span)
 - ▣ A display tulajdonság következő értékei tartoznak ide: inline, inline-table, inline-block
- A szabvány beszél még soron belüli dobozokról is.
- Bővebben:
<http://www.w3.org/TR/CSS2/visuren.html#block-boxes>

Div és span tagek használata a HTML nyelvben



- `<div></div>`
 - ▣ Ezzel az elemmel több más elemet körbekeríthetünk. Így egy adott div elemre beállított tulajdonságok öröklődnek a leszármazott elemekre.
 - ▣ Gyakran használjuk a taget arra, hogy pozícionált blokkokat hozzunk létre.
 - HTML5-ben sok olyan strukturális tag jelenik meg, ami miatt a div-et ritkábban kell használnunk...
- ``
 - ▣ Ez az elem soron belüli elem, vagyis főként karakterszintű formázások elvégzéséhez szoktuk használni.

Display tulajdonság



- Display: Egy elem megjelenítési módját adja meg.
 - ▣ Inline, block, list-item, run-in, compact, marker, table, inline-table, table-row-group, table-header-group, table-footer-group, table-row, table-column-group, table-column, table-cell, table-caption, none

Display tulajdonság 2.



- **display: none**
 - Nem jelenik meg. Fókuszot nem kap (javascript segítségével igen).
- **display: inline**
 - Az elem soron belüli elem lesz (előtte és utána nem lesz sortörés)
- **display: block**
 - Blokkszintű elemként viselkedik az elem
- **display: run-in**
 - Befutó megjelenés. Az elemünk az utána következő blokk elem első soron belüli eleme lesz

Display tulajdonság 3.



- **display: compact**
 - A megjelenítő egy egy soros soron belüli blokként próbálja beilleszteni a következő blokk elembe - amennyiben a következő egy blokk elem és "befér" oda - a kompakt megjelenítés tulajdonságú elemet.
- **display: list-item**
 - A lista elemek két egymás mellett álló téglalapként jelennek meg, a bal oldali behúzásként is felfogható, a jobb oldaliban pedig a lista tartalma jelenik meg.

Display tulajdonság - példa



```
<div id="menu">
<ul>
<li><a href="#">Kezdőlap</a></li>
<li><a href="#">Magamról</a></li>
<li><a href="#">Hobby</a></li>
<li><a href="#">Linkek</a></li>
</ul>
</div>
```

- [Kezdőlap](#)
- [Magamról](#)
- [Hobby](#)
- [Linkek](#)

CSS nélkül

```
<style type="text/css">
<|—
div#menu ul {padding:0; margin:0;}
div#menu li {
list-style-type:none;
display:inline;
padding: 5px 20px;
border-right:1px dotted black;
margin:0;
}
-->
</style>
```

CSS alkalmazásával

[Kezdőlap](#)

[Magamról](#)

[Hobby](#)

[Linkek](#)

Display böngészőtámogatottság



Selector or declaration	IE 5.5	IE 6	IE 7	IE8	IE9	FF 11.0 Win	FF 10.0.2 Mac	Saf 5.1 Win	Saf 5.1.2 Mac	Chrome 18 Win	Chrome 17 Mac	Opera 11.61 Win	Opera 11.61 Mac
block		yes				yes	yes	yes		yes		yes	
inline		yes				yes		yes		yes		yes	
none		yes				yes		yes		yes		yes	
inline-block	no	incomplete	yes			yes	yes	yes		yes		yes	
• IE 6/7 accepts the value only on elements with a natural display: inline.													
list-item	no	yes				yes		yes		yes		yes	
run-in	no	yes				no		yes		yes		yes	
table	no	yes				yes		yes		yes		yes	
Selector or declaration	IE 5.5	IE 6	IE 7	IE8	IE9	FF 11.0 Win	FF 10.0.2 Mac	Saf 5.1 Win	Saf 5.1.2 Mac	Chrome 18 Win	Chrome 17 Mac	Opera 11.61 Win	Opera 11.61 Mac

[http://www.quirksmode.org/css/display.html](http://www quirksmode.org/css/display.html)

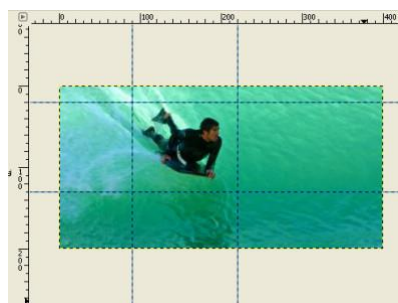
Elemek látható része



- A clip tulajdonság segítségével beállíthatjuk, hogy egy elemnek csak egy meghatározott **részt jelenítsük meg**.
- A rect() érték segítségével megadhatjuk azt a **téglalapot, amit látni szeretnénk az elemből**.
 - A zárójelek között **négy, vesszővel elválasztott hosszúságértéket** adunk meg, melyek sorrendben a fenti, a jobb, az alsó és a bal értékeket határozzák meg. Ezek az értékek az elem bal felső sarkától számolódnak.

```
<!DOCTYPE html>
<html lang="hu">
<head>
<title>Clip és rect() szemléltetése</title>
<meta charset="utf-8">
<style type="text/css">
div {
    width:400px;height:200px;
    border:1px solid black;
    padding:0;position: absolute;
}
div#doboz1 {
    left: 10px;
    top: 10px;
}

div#doboz2 {
    left: 10px;
    top: 250px;
    clip: rect(20px, 220px, 130px, 90px);
}
</style>
</head>
<body>
<div id="doboz1"></div>
<div id="doboz2"></div>
</body>
</html>
```



Átlátszóság (opacity)



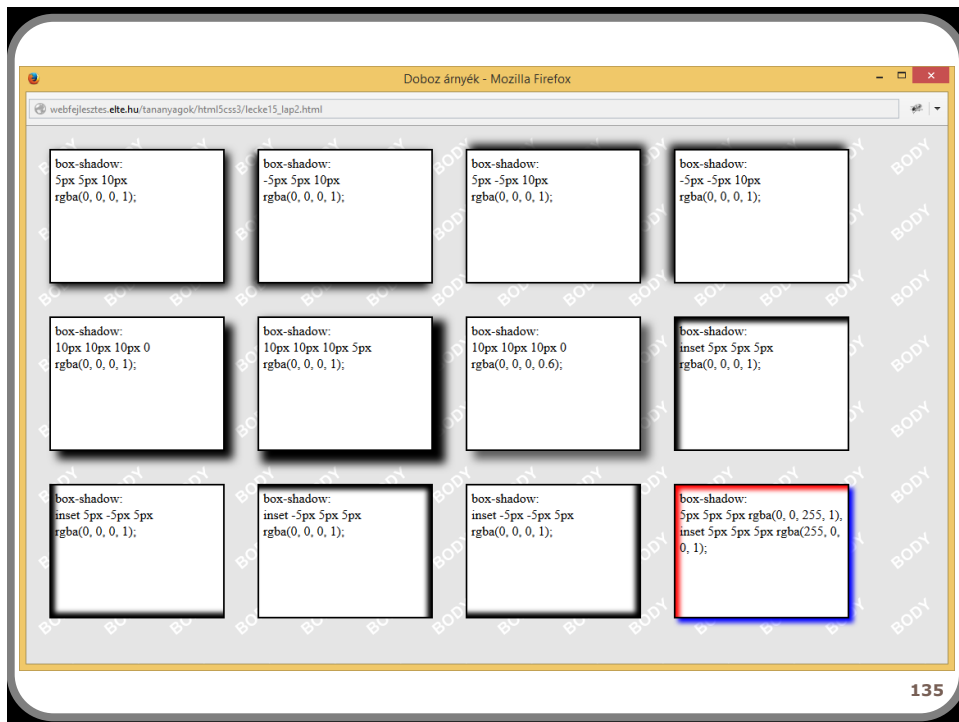
- Megadhatunk egy tetszőleges 0.0 (teljesen átlátszó) és 1.0 (teljesen átlátszatlan) közötti **valós számot**.



Doboz árnyék (box-shadow)



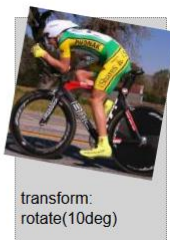
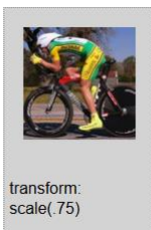
- Először két tetszőleges hosszúságértékkel adjuk meg a vízszintes és függőleges eltolás mértékét.
- Ezután egy pozitív hosszúságérték következik, ami meghatározza, hogy mennyire legyen elmosódott az árnyék. 0 érték esetén nincs elmosódás.
- Majd opcionális módon egy pozitív hosszúságértéket megadva meghatározhatjuk az árnyék (megadott színnel egyező) részének méretét. Ha nem adjuk meg, akkor 0-nak tekinti a böngésző az értékét.
- Ezután egy tetszőleges színértéket megadva beállítjuk az árnyék színét (akár átlátszóság megadásával is, lásd rgba).
- Opcionálisan megadva az inset hatására az árnyék az elemen belülrre kerül.



Transzformációk és átmenetek



- A rotate() érték **elforgatja** az elemet a képernyő síkján, mégpedig a paraméterül kapott szöggel.
- A rotateX() és a rotateY() értékek a paraméterül kapott szöggel **forgatják el** az elemet a képernyő **vízszintes**, és **függőleges** tengelye körül.
- A scale() érték a paraméterül kapott egy vagy (vesszővel elválasztva) két számérték szorosára **méretezi át** az elemet. Az első paraméter a vízszintes, az opcionális második a függőleges átméretezésért felel.
- A scaleX()/scaleY() értékek a paraméterül kapott számérték szorosára **méretezi át** az elemet **vízszintesen/függőlegesen**.



Transzformációk és átmenetek



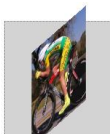
- A `skew()` érték a paraméterül kapott egy vagy (vesszővel elválasztva) két szög mértékével **ferdíti** az elemet. Az első paraméter a vízszintes, az opcionális második a függőleges ferdtítésért felel.
- A `skewX()/skewY()` értékek a paraméterül kapott szöggel **ferditik el** az elemet **vízszintesen/függőlegesen**.
- A `translate()` érték a paraméterül kapott egy vagy (vesszővel elválasztva) két hosszúságérték mértékével **tolja el** az elemet. Az első paraméter a vízszintes, az opcionális második a függőleges eltolásért felel. A pozitív értékek jobbra/le a negatív értékek balra/fel történő eltolást eredményeznek.
- A `translateX()/translateY()` értékek a paraméterül kapott hosszúságértékkel **tolják el** az elemet **vízszintesen/függőlegesen**.



`transform:
translate(10px,
10px);`



`transform:
skew(5deg, 5deg);`

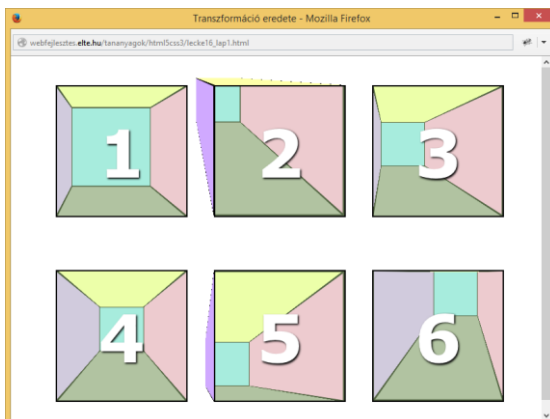


`transform:
rotateX(30deg)
rotateY(-60deg)`

3D transzformációk



- Részletesen a web-es tananyagban térünk ki...





CSS 2.1 pozícionálási sémák



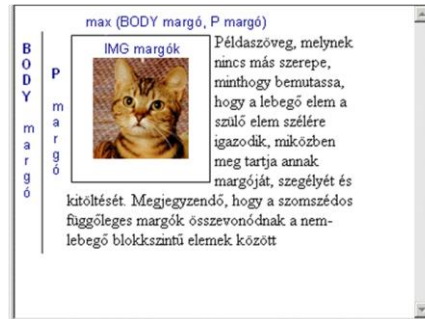
- 3 pozícionálási séma
 - ▣ Normál elrendezés: blokk formázás, soron belüli formázás, relatív pozícionálás
 - ▣ Lebegés (float): Balra, illetve jobbra történhet a lebegtetés. A lebegtetett elemet a többi tartalom körülveszi/körülveheti.
 - ▣ Abszolút pozícionálás: A normál elrendezésből kikerül, a tartalmazó blokk pozíciójához képest történik az elrendezés.

Lebegő elemek (float: left | right | none)



```
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML
4.01 Transitional//EN"
"http://www.w3.org/TR/html4/loose.dtd">
<html>
<head>
<title>stíluslap példa</title>
<style type="text/css">
img {float: left}
body, p, img {margin: 25px}
</style>
</head>
<body>
<p>

Példaszöveg, melynek nincs más
szerepe, minthogy bemutassa, hogy a
lebegő elem a szülő elem szélére
igazodik, miközben megtartja annak
margóját, szegélyét és kitöltését.
megjegyzendő, hogy a szomszédos
függőleges margók összevonódnak a nem-
lebegő blokkszintű elemek között
</p>
</body>
</html>
```



Overflow tulajdonság



- Mi történik, ha a tartalom mérete nagyobb, mint a tartalmazó elemé?

overflow: visible

Ez az alapbeállítás, ekkor a tartalom kilóghat a tartalmazó elemből.

duplaklikk.blog.hu



overflow: hidden

A túlnyúlás el lesz rejtve.



overflow: auto

A túlnyúlás el lesz rejtve, de megjelenik egy gördítősáv, ha szükséges.



overflow: scroll

A túlnyúlás el lesz rejtve, és mindenképpen megjelenik a gördítősáv..



Overflow tulajdonság



- Ha nem szeretnénk, hogy egy lebegtetett elem kilógjon a dobozból, akkor is sikerrel használhatjuk az `overflow:hidden`, vagy `overflow:auto`; tulajdonságot.



Lorem ipsum dolor sit amet, consectetur adipiscing elit. Donec sed ipsum ornare quam rutrum cursus eget quis metus. Sed condimentum, sem at laoreet tincidunt, augue est condimentum sem, tristique rhoncus nulla nulla et arcu. Donec quis lorem fermentum mi eleifend laoreet.

```
div.doboz {  
  background-color:lightyellow;  
  border:1px solid green;  
  padding:5px;  
  overflow:hidden;  
}
```

Lebegő elemek és a clear tulajdonság



- Clear: **none** | **left** | **right** | **both** | **inherit**
 - ▣ az elem melyik oldalán engedélyezi lebegő elemek megjelenését.
- Ha a **clear** értéke **leftre** van beállítva, az elem, amelyre alkalmaztuk, a bal oldalán levő bármely lebegő elem alá kerül.

Lebegő elemek (clear: left | right | both | inherit)

```
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01
Transitional//EN"
"http://www.w3.org/TR/html4/loose.dtd">
<html>
<head>
<title></title>
<style type="text/css">
    img {margin:5px;}
    img.bal {float:left;}
    img.jobb {float:right;}
    p.tores {clear:both}
</style>
</head>
<body>

<p>
Lorem ipsum dolor sit amet, consectetur
adipiscing elit...</p>
<p class="tores">Cum sociis natoque
penatibus et magnis dis parturient
montes...</p>

<p>Phasellus tortor nisi, venenatis sit amet
tincidunt ut,...</p>
</body>
</html>
```

Hiába van még hely a kép mellett, a clear tulajdonság miatt a bekezdés a kép alá kerül...



Lorem ipsum dolor sit amet, consectetur adipiscing elit. Donec elit libero, feugiat ac euismod sit amet, accumsan a mi. Curabitur mi urna, consectetur vel elementum eget, vestibulum id ligula.

Cum sociis natoque penatibus et magnis dis parturient montes, nascetur ridiculus mus. Mauris vestibulum gravida ligula, eget semper ligula tincidunt in. In vitae leo ac metus consectetur feugiat. Nullam pellentesque massa ut ipsum fringilla tristique. In tellus lectus, posuere sed rutrum sit amet, auctor et nunc. Sed a sapien nulla, ut ullamcorper ipsum. Praesent eu nibh a mi convallis consequat vitae sit amet odio.

Phasellus tortor nisi, venenatis sit amet tincidunt ut, fringilla nec diam. Vestibulum commodo, ante non dictum mattis, leo velit blandit nisi, convallis pellentesque arcu justo vel libero. Ut erat augue, eleifend et lobortis eget, dignissim quis lectus. Cras felis magna, condimentum ut porta a, pretium nec nulla. Maecenas faucibus leo in ante venenatis eu molestie orci dapibus. Sed pharetra laoreet euismod. Nulla facilisi. Aliquam ultrices mollis elementum. Curabitur urna metus, eleifend in malesuada eget, hendrerit vel urna. Aenean blandit lectus ac justo dignissim tempus. Quisque faucibus mollis pharetra. Vivamus justo eros, condimentum a semper sed, ornare sed justo. Fusce id dui tortor, accumsan sollicitudin urna. Curabitur ac augue libero, nec mattis lorem



Pozícionálás

- **Position:** static | relative | absolute | fixed | inherit
- **Static**
 - ▣ Nem pozícionált, a kódban elfoglalt helye határozza meg a vásznon való elhelyezkedését
- **Relative**
 - ▣ A statikushoz hasonló helyre kerül az elem, de a bal és felső pozíció megadásával eltolhatjuk. (pl. kerüljön lejjebb és jobbra)
- **Absolute**
 - ▣ Ez az elem kikerül a megjelenítési folyamatból. Az abszolút pozícionálású elemek a (dokumentumfaban) legközelebb eső olyan szülő elemhez képest lesznek pozícionálva, amelyekre szintén meg van adva a position tulajdonság MÁS értékkel, mint a static.
- **Fixed**
 - ▣ A fix pozícionálással a görgetés ellenére is adott pozíción marad az elem. (képernyő-koordinátához rögzül)

Statikus pozícionálás



```
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01
Transitional//EN"
"http://www.w3.org/TR/html4/loose.dtd">
<html>
<head>
<title></title>
<link href="dobozok.css" rel="stylesheet"
type="text/css">
<style type="text/css">
div#doboz1 {
    position: static;
}

div#doboz2 {
    position: static;
}
</style>
</head>
<body>
<div id="doboz1">Lorem ipsum dolor sit amet...</div>
<div id="doboz2">Lorem ipsum dolor sit amet...</div>
</body>
</html>
```

doboz1: Lorem ipsum dolor sit amet, consectetur adipiscing elit. Phasellus orci urna, placerat vel, rutrum vitae, tincidunt quis, eros. Morbi elementum semper velit. Aliquam erat volutpat. Praesent varius ultrices neque. Fusce lorem turpis, eleifend vel, placerat vitae, elementum ac, augue. Pellentesque erat tellus, facilisis vitae, tempor mattis, bibendum ut, ligula.

doboz2: Lorem ipsum dolor sit amet, consectetur adipiscing elit. Phasellus orci urna, placerat vel, rutrum vitae, tincidunt quis, eros. Morbi elementum semper velit. Aliquam erat volutpat. Praesent varius ultrices neque. Fusce lorem turpis, eleifend vel, placerat vitae, elementum ac, augue. Pellentesque erat tellus, facilisis vitae, tempor mattis, bibendum ut, ligula.

A dobozok pozíciója a kódban elfoglalt sorrendtől függ.

Relatív pozícionálás



```
<!DOCTYPE HTML>
<html>
<head>
<link href="dobozok.css" rel="stylesheet" type="text/css">
<style type="text/css">
div#doboz1 {position: static;}
div#doboz2 {
    position: relative;
    width: 90%;
    left: 50px;
    top: 10px;
}
div#doboz3 {
    position: relative;
    width: 400px;
    left: 20px;
    top: 20px;
}
</style>
</head>
<body>
<div id="doboz1">Lorem ipsum dolor sit amet...</div>
<div id="doboz2">Lorem ipsum dolor sit amet...</div>
<div id="doboz3">Lorem ipsum dolor sit amet...</div>
</body>
</html>
```

doboz1: Lorem ipsum dolor sit amet, consectetur adipiscing elit. Phasellus orci urna, placerat vel, rutrum vitae, tincidunt quis, eros. Morbi elementum semper velit. Aliquam erat volutpat. Praesent varius ultrices neque. Fusce lorem turpis, eleifend vel, placerat vitae, elementum ac, augue. Pellentesque erat tellus, facilisis vitae, tempor mattis, bibendum ut, ligula.

doboz2: Lorem ipsum dolor sit amet, consectetur adipiscing elit. Phasellus orci urna, placerat vel, rutrum vitae, tincidunt quis, eros. Morbi elementum semper velit. Aliquam erat volutpat. Praesent varius ultrices neque. Fusce lorem turpis, eleifend vel, placerat vitae, elementum ac, augue. Pellentesque erat tellus, facilisis vitae, tempor mattis, bibendum ut, ligula.

doboz3: Lorem ipsum dolor sit amet, consectetur adipiscing elit. Phasellus orci urna, placerat vel, rutrum vitae, tincidunt quis, eros. Morbi elementum semper velit. Aliquam erat volutpat. Praesent varius ultrices neque. Fusce lorem turpis, eleifend vel, placerat vitae, elementum ac, augue. Pellentesque erat tellus, facilisis vitae, tempor mattis, bibendum ut, ligula.

A dobozt eltolhatjuk ahhoz a pozícióhoz képest, ahova amúgy (statikus poz. esetén) kerülne.

Abszolút pozícionálás (1.)



```
<!DOCTYPE HTML>
<html>
<head>
<title></title>
<link href="dobozok.css" rel="stylesheet"
type="text/css">
<style type="text/css">
div#doboz1 {
    position: absolute;
    width: 150px;
    left: 10px;
    top: 10px;
}

div#doboz2 {
    position: absolute;
    width: 50%;
    left: 170px;
    top: 30px;
}
</style>
</head>
<body>
<div id="doboz1">Lorem ipsum dolor sit amet...</div>
<div id="doboz2">Lorem ipsum dolor sit amet...</div>
</body>
</html>
```

doboz1: Lorem ipsum dolor sit amet, consectetur adipiscing elit. Phasellus orci urna, placerat vel, rutrum vitae, tincidunt quis, eros. Morbi elementum semper velit. Aliquam erat volutpat. Praesent varius ultrices neque. Fusce lorem turpis, eleifend vel, placerat vitae, elementum ac, augue. Pellentesque erat tellus, facilisis vitae, tempor mattis, bibendum ut, ligula.

doboz2: Lorem ipsum dolor sit amet, consectetur adipiscing elit. Phasellus orci urna, placerat vel, rutrum vitae, tincidunt quis, eros. Morbi elementum semper velit. Aliquam erat volutpat. Praesent varius ultrices neque. Fusce lorem turpis, eleifend vel, placerat vitae, elementum ac, augue. Pellentesque erat tellus, facilisis vitae, tempor mattis, bibendum ut, ligula.

Az 1.doboz fix, a 2. doboz változó szélességű.

Abszolút pozícionálás (2.)



```
<!DOCTYPE HTML>
<html>
<head>
<title></title>
<link href="dobozok.css" rel="stylesheet"
type="text/css">
<style type="text/css">
div#doboz1 {
    position: absolute;
    width: 250px;
    left: 10px;
    top: 10px;
}

div#doboz2 {
    position: absolute;
    width: 250px;
    left: 200px;
    top: 40px;
}
</style>
</head>
<body>
<div id="doboz1">Lorem ipsum dolor sit amet...</div>
<div id="doboz2">Lorem ipsum dolor sit amet...</div>
</body>
</html>
```

doboz1: Lorem ipsum dolor sit amet, consectetur adipiscing elit. Phasellus orci urna, placerat vel, rutrum vitae, tincidunt quis, eros. Morbi elementum semper velit. Aliquam erat volutpat. Praesent varius ultrices neque. Fusce lorem turpis, eleifend vel, placerat vitae, elementum ac, augue. Pellentesque erat tellus, facilisis vitae, tempor mattis, bibendum ut, ligula.

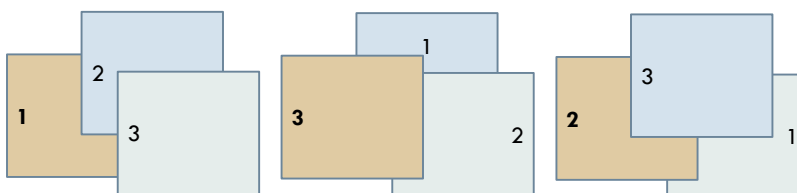
doboz2: Lorem ipsum dolor sit amet, consectetur adipiscing elit. Phasellus orci urna, placerat vel, rutrum vitae, tincidunt quis, eros. Morbi elementum semper velit. Aliquam erat volutpat. Praesent varius ultrices neque. Fusce lorem turpis, eleifend vel, placerat vitae, elementum ac, augue. Pellentesque erat tellus, facilisis vitae, tempor mattis, bibendum ut, ligula.

A két doboz akár egymásra is kerülhet. A 2. doboz takarja az első, mert később szerepel a kódban. Ez felülírható a z-index paraméterrel. Lásd köv. dia.

Rétegek



- A CSS alapú megjelenítéskor minden abszolútként pozícionált elemnek van egy pozíciója a harmadik dimenzióban, a "z" tengely mentén. Minél nagyobb a szám, annál feljebb kerül.
- Pl. `z-index:3;`



Abszolút pozícionálás (3.)



```
<!DOCTYPE HTML>
<html>
<head>
<link href="dobozok.css" rel="stylesheet"
type="text/css">
<style type="text/css">
div#doboz1 {
    position: absolute;
    width: 250px;
    left: 10px;
    top: 10px;
    z-index: 2;
}

div#doboz2 {
    position: absolute;
    width: 250px;
    left: 200px;
    top: 40px;
    z-index: 1;
}
</style>
</head>
<body>
<div id="doboz1">Lorem ipsum dolor sit amet.</div>
<div id="doboz2">Lorem ipsum dolor sit amet.</div>
</body>
</html>
```

doboz1: Lorem ipsum dolor sit amet, consectetur adipiscing elit. Phasellus orci urna, placerat vel, rutrum vitae, tincidunt quis, eros. Morbi elementum semper velit. Aliquam erat volutpat. Praesent varius ultrices neque. Fusce lorem turpis, eleifend vel, placerat vitae, elementum ac, augue. Pellentesque erat tellus, facilisis vitae, tempor mattis, bibendum ut, ligula.

orem ipsum dolor sit amet, per adipiscing elit. Phasellus placerat vel, rutrum vitae, quis, eros. Morbi elementum lit. Aliquam erat volutpat. rarius ultrices neque. Fusce pis, eleifend vel, placerat entum ac, augue. ue erat tellus, facilisis vitae, tempor mattis, bibendum ut, ligula.

A z-index paraméterrel megadhatjuk a mélységi sorrendet. A nagyobb szám feljebbi elhelyezést eredményez a rétegeket tekintve.

Abszolút pozícionálás (4.)



```
<!DOCTYPE HTML>
<html>
<head>
<link href="dobozok.css" rel="stylesheet"
type="text/css">
<style type="text/css">
body {
    margin-left: 180px;
    margin-right: 10px;
    margin-top: 10px;
    margin-bottom: 10px;
}

div#doboz1 {
    position: absolute;
    width: 150px;
    left: 10px;
    top: 10px;
}
</style>
</head>
<body>
<div id="doboz1">Lorem ipsum dolor sit amet.</div>
<div id="doboz2">Lorem ipsum dolor sit amet.</div>
</body>
</html>
```

doboz1: Lorem ipsum dolor sit amet, consectetur adipiscing elit. Phasellus orci urna, placerat vel, rutrum vitae, tincidunt quis, eros. Morbi elementum semper velit. Aliquam erat volutpat. Praesent varius ultrices neque. Fusce lorem turpis, eleifend vel, placerat vitae, elementum ac, augue. Pellentesque erat tellus, facilisis vitae, tempor mattis, bibendum ut, ligula.

doboz2: Lorem ipsum dolor sit amet, consectetur adipiscing elit. Phasellus orci urna, placerat vel, rutrum vitae, tincidunt quis, eros. Morbi elementum semper velit. Aliquam erat volutpat. Praesent varius ultrices neque. Fusce lorem turpis, eleifend vel, placerat vitae, elementum ac, augue. Pellentesque erat tellus, facilisis vitae, tempor mattis, bibendum ut, ligula. Lorem ipsum dolor sit amet, consectetur adipiscing elit. Phasellus orci urna, placerat vel, rutrum vitae, tincidunt quis, eros. Morbi elementum semper velit. Aliquam erat volutpat. Praesent varius ultrices neque. Fusce lorem turpis, eleifend vel, placerat vitae, elementum ac, augue. Pellentesque erat tellus, facilisis vitae, tempor mattis, bibendum ut, ligula. Lorem ipsum dolor sit amet, consectetur adipiscing elit. Phasellus orci urna, placerat vel, rutrum vitae, tincidunt

Itt a body margója miatt kerül a 2. doboz jobbra, és az első doboz a margó miatt előálló üres helyre van bepozícionálva abszolút módon.

Abszolút pozícionálás (5.)



```
<style type="text/css">
div#doboz1 {
    position: absolute;
    left: 1%;
    width: 18%;
    top: 10px;
}

div#doboz2 {
    position: absolute;
    left: 22%;
    width: 55%;
    top: 10px;
}

div#doboz3 {
    position: absolute;
    left: 80%;
    width: 18%;
    top: 10px;
}
</style>
<body>
<div id="doboz1">Lorem ipsum dolor sit amet.</div>
<div id="doboz2">Lorem ipsum dolor sit amet.</div>
<div id="doboz3">Lorem ipsum dolor sit amet.</div>
</body>
</html>
```

doboz1: Lorem ipsum dolor sit amet, consectetur adipiscing elit. Phasellus orci urna, placerat vel, rutrum vitae, tincidunt quis, eros. Morbi elementum semper velit. Aliquam erat volutpat. Praesent varius ultrices neque. Fusce lorem turpis, eleifend vel, placerat vitae, elementum ac, augue. Pellentesque erat tellus, facilisis vitae, tempor mattis, bibendum ut, ligula.

doboz2: Lorem ipsum dolor sit amet, consectetur adipiscing elit. Phasellus orci urna, placerat vel, rutrum vitae, tincidunt quis, eros. Morbi elementum semper velit. Aliquam erat volutpat. Praesent varius ultrices neque. Fusce lorem turpis, eleifend vel, placerat vitae, elementum ac, augue. Pellentesque erat tellus, facilisis vitae, tempor mattis, bibendum ut, ligula. Lorem ipsum dolor sit amet, consectetur adipiscing elit. Phasellus orci urna, placerat vel, rutrum vitae, tincidunt quis, eros. Morbi elementum semper velit. Aliquam erat volutpat. Praesent varius ultrices neque. Fusce lorem turpis, eleifend vel, placerat vitae, elementum ac, augue. Pellentesque erat tellus, facilisis vitae, tempor mattis, bibendum ut, ligula. Lorem ipsum dolor sit amet, consectetur adipiscing elit. Phasellus orci urna, placerat vel, rutrum vitae, tincidunt quis, eros. Morbi elementum semper velit. Aliquam erat volutpat. Praesent varius ultrices neque. Fusce lorem turpis, eleifend vel, placerat vitae, elementum ac, augue. Pellentesque erat tellus, facilisis vitae, tempor mattis, bibendum ut, ligula.

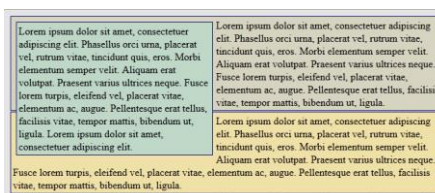
doboz3: Lorem ipsum dolor sit amet, consectetur adipiscing elit. Phasellus orci urna, placerat vel, rutrum vitae, tincidunt quis, eros. Morbi elementum semper velit. Aliquam erat volutpat. Praesent varius ultrices neque. Fusce lorem turpis, eleifend vel, placerat vitae, elementum ac, augue. Pellentesque erat tellus, facilisis vitae, tempor mattis, bibendum ut, ligula.

Három oszlopos, folyékony arculat, melyben a pozícionálás abszolút módon, %-os értékekkel történik.

Lebegtetés



```
...
<style type="text/css">
div#doboz2 {
    background-color: #C0D8C8;
    float: left;
    width: 300px;
    margin: 5px;
}
</style>
...
<body>
<div id="doboz1">Lorem ipsum dolor sit amet
  <div id="doboz2">Lorem ipsum dolor sit amet...</div>
  Lorem ipsum dolor sit amet...
</div>
<div id="doboz3">Lorem ipsum dolor sit amet</div>
</body>
</html>
```



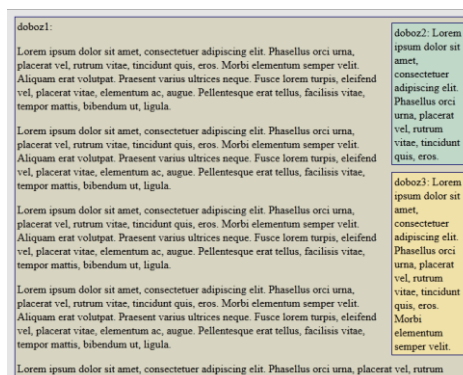
A lebegtetés miatt a többi elem körbefolyja az adott elemet.

Lebegtetés (2.)



```
...
<style type="text/css">
div#doboz2 {
    float: right;
    width: 100px;
    margin: 5px;
}

div#doboz3 {
    float: right;
    width: 100px;
    margin: 5px;
    clear: right;
}
</style>
...
<body>
<div id="doboz1">Lorem ipsum dolor sit
amet
  <div id="doboz2">Lorem ipsum dolor
sit amet...</div>
  <div id="doboz3">Lorem ipsum dolor
sit amet</div>
  Lorem ipsum dolor sit amet
</div>
</body>
</html>
```



A clear tulajdonság miatt a 3. doboz nem kerülhet a 2. doboz mellé.

Hely kiszámítása (calc)



- Gondot okoz, ha a méretezésnél többféle mértékegységet (px,em,%) használtunk, és tudnunk kellene, mekkora a rendelkezésre álló hely.
- Erre jó a **calc()**
- Példa

```
#blokk { width: calc(50% - 100px); }
```
- A blokk 100 képponttal keskenyebb, mint a rendelkezésre álló hely 50%-a

Hely kiszámítása (calc)



- Ahhoz, hogy régebbi böngészőkben is működjön böngészőre vonatkozó prefixeket is használni kell..
 - -webkit: safari/chrome
 - -moz: mozilla firefox
 - -o: opera
 - -ms: microsoft
- ```
#blokk {
 width: -webkit-calc(50% - 100px);
 width: -moz-calc(50% - 100px);
 width: calc(50% - 100px);
}
```

## Calc böngészőtámogatottsága



| Current aligned |        | Usage relative |        | Show all |       |              |              |                   |                    |  |  |
|-----------------|--------|----------------|--------|----------|-------|--------------|--------------|-------------------|--------------------|--|--|
| IE              | Edge * | Firefox        | Chrome | Safari   | Opera | iOS Safari * | Opera Mini * | Android Browser * | Chrome for Android |  |  |
| 8               |        |                | 45     |          |       |              |              | 4.3               |                    |  |  |
| 9               |        |                | 46     |          |       |              |              | 4.4               |                    |  |  |
| 10              |        | 43             | 47     |          |       | 8.4          |              | 4.4.4             |                    |  |  |
| 11              | 13     | 44             | 48     | 9        | 34    | 9.2          | 8            | 47                | 47                 |  |  |
|                 | 14     | 45             | 49     | 9.1      | 35    | 9.3          |              |                   |                    |  |  |
|                 |        | 46             | 50     |          | 36    |              |              |                   |                    |  |  |
|                 |        | 47             | 51     |          |       |              |              |                   |                    |  |  |

<http://caniuse.com/#feat=calc>

## Láthatóság



- Visibility: visible | hidden | collapse
  - ▣ Visible: látható
  - ▣ Hidden: rejtett
  - ▣ Collapse: táblázat elemeknél használatos, egy oszlop vagy sor elrejtésére használható
- Attól, hogy az elem nem jelenik meg, attól még a helyet fenntartja a böngésző. Ha nem ezt akarjuk, akkor a display: none; tulajdonságot használjuk!

## Láthatóság példa



```
<!DOCTYPE html>
<html>
<head>
 <title>Visibility</title>
 <style type="text/css">
 .rejtett
 {visibility:hidden; float:left; }
 </style>
</head>
<body>

<p>Lorem ipsum ...</p>

</body>
</html>
```

Lorem ipsum dolor sit amet, consectetur adipiscing elit. Ut tempus vitae massa a tincidunt. Proin accumsan aliquam quam sit amet tristique. Proin convallis ex felis, quis viverra turpis commodo eu. Phasellus at lacus nisi. Quisque vel sem sodales, varius felis a, egestas tellus. Nullam scelerisque venenatis maximus. Phasellus blandit vulputate tempus.

Cras sit amet tempor metus, in pharetra velit. Maecenas non ipsum id tortor vestibulum rutrum. Integer massa lacus, gravida in condimentum fringilla, placerat ut metus. Donec porttitor gravida libero, vel luctus eros gravida at.

162

## Layoutok (elrendezések)



## Learn CSS layout



□ <http://learnlayout.com/>



### Learn CSS Layout

This site teaches the CSS fundamentals that are used in any website's layout.

I assume you already know what selectors, properties, and values are. And you probably know a thing or two about layout, though it may still be a rage-provoking activity for you. If you want to learn HTML and CSS from the beginning, you should check out [this tutorial](#). Otherwise, let's see if we can save you some fury on your next project.

Get Started

## CSS layout generator



□ <http://csslayoutgenerator.com/>

The screenshot shows the CSSLayoutGenerator.com interface. It features several sections for configuring a CSS layout:

- DOCTYPE:** A dropdown menu set to 'HTML 5'.
- CSS Reset:** A dropdown menu set to 'CSS reset from Eric ...'.
- Layout width:** Radio buttons for 'Fixed' (selected) and 'Liquid'. The fixed width is set to '1000 px'.
- Header:** A checkbox for 'Height'.
- Footer:** A checkbox for 'Height'.
- Sidebars:** Radio buttons for '1 sidebar', '2 sidebars' (selected), and 'No sidebars'. Under '2 sidebars', there are options for '1 left, 1 right' (selected), 'both left', and 'both right'. Widths are set to '250 px' for both left and right sidebars.
- Additional options:** Checkboxes for 'Keep the footer at the bottom of browser window' and 'Emulate equal height columns'.
- Preview:** A visual representation of the generated layout with colored blocks for header, sidebar, main content, and footer.
- Generator About:** A button in the top right corner.
- GENERATECSS:** A large yellow button with the text '30+ CSS Generators now all in one place. Click here! NEW'.



# Free CSS Layouts



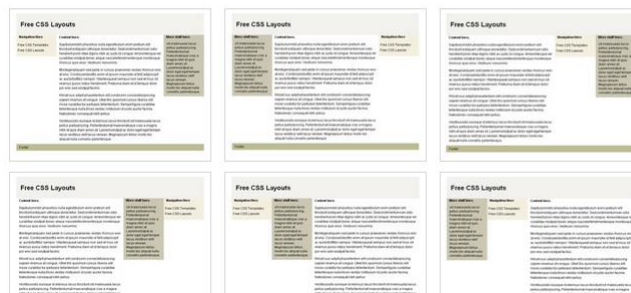
□ <http://www.free-css.com/free-css-layouts/page1>

FREE CSS LAYOUTS PAGE 1

« Previous

| 1 | 2 | 3 | ... | 20 | 21 |

Next »



166

## CSS tulajdonságok



## CSS 2 tulajdonság lista



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

W3C Recommendation

### Appendix F. Full property table

*This appendix is informative, not normative.*

Name	Values	Initial value	Applies to (Default: all)	Inherited?	Percentages (Default: N/A)	Media groups
'azimuth'	<angle>   [[ left-side   far-left   left   center-left   center   center-right   right   far-right   right-side ]    behind ]   leftwards   rightwards   inherit	center		yes		aural
'background-attachment'	scroll   fixed   inherit	scroll		no		visual
'background-color'	<color>   transparent   inherit	transparent		no		visual
'background-image'	<uri>   none   inherit	none		no		visual
'background-position'	{ [ <percentage>   <length>   left   center   right ]   [ <percentage>   <length>   top   center   bottom ]? }   [ [ left   center   right ]    [ top   center   bottom ] ]   inherit	0% 0%		no	refer to the size of the box itself	visual
'background-repeat'	repeat   repeat-x   repeat-y   no-repeat   inherit	repeat		no		visual

## CSS 1,2,3 tulajdonság lista



□ <http://meiert.com/en/indices/css-properties/>

	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z
Property	CSS 1	CSS 2	CSS 2.1	CSS 3	Initial Value																					
alignment-adjust	X	X	X	✓	auto																					
alignment-baseline	X	X	X	✓	baseline																					
animation	X	X	X	✓	depending on individual properties																					
animation-delay	X	X	X	✓	0																					
animation-direction	X	X	X	✓	normal																					
animation-duration	X	X	X	✓	0																					
animation-iteration-count	X	X	X	✓	1																					
animation-name	X	X	X	✓	none																					
animation-play-state	X	X	X	✓	running																					
animation-timing-function	X	X	X	✓	ease																					
appearance	X	X	X	✓	normal																					

## CSS tulajdonságok (W3C)



□ <https://www.w3.org/Style/CSS/all-properties.en.html>

✓	border-collapse	CSS 2.1	REC
	border-color	CSS Backgrounds and Borders Module Level 3	ED
✓	border-color	CSS 2.1	REC
✓	border-color	CSS Backgrounds and Borders Module Level 3	CR
	border-image	CSS Backgrounds and Borders Module Level 3	ED
✓	border-image	CSS Backgrounds and Borders Module Level 3	CR
	border-image-outset	CSS Backgrounds and Borders Module Level 3	ED
✓	border-image-outset	CSS Backgrounds and Borders Module Level 3	CR
	border-image-repeat	CSS Backgrounds and Borders Module Level 3	ED
✓	border-image-repeat	CSS Backgrounds and Borders Module Level 3	CR
	border-image-slice	CSS Backgrounds and Borders Module Level 3	ED
✓	border-image-slice	CSS Backgrounds and Borders Module Level 3	CR

## Vicces CSS példák



```
.rich-people {
 top: 1%;
}
.working-class {
 bottom: 99%;
}
```

```
.ninja {
 color: black;
 visibility: hidden;
 animation-duration: 0.00001s;
}
```

```
.ghost {
 color: white;
 opacity: 0.1;
}
```

```
.autobots {
 transform: translate3d();
}
```

<http://www.hongkiat.com/blog/funny-css-puns/>

## Vicces CSS példák



```
#chucknorris {
 color: #BADA55;
}
```

```
#eminem {
 word-spacing: 0;
}
```

```
#lego {
 display: block;
}
```

```
.hobbit {
 height: 50%;
}
.hobbit #foot {
 width: 200%;
}
```

```
#ikea {
 display: table;
}
```

<http://www.hongkiat.com/blog/funny-css-puns/>

## Vicces CSS példák



```
#tower-of-pisa {
 font-style: italic;
}
```

```
#usa + #mexico {
 border: 1px dashed;
}
```

```
#berlin-wall-1989 {
 border-collapse: collapse;
 height: 0;
}
```

```
#titanic {
 float: none;
 bottom: 0;
}
```

Írjatok Ti is hasonlókat! <http://padlet.com/abonyita/8wglh5stnteo>

<http://www.hongkiat.com/blog/funny-css-puns/>

173

## CSS előfeldolgozók (preprocessors)

<http://www.sitepoint.com/6-current-options-css-preprocessors/>



## CSS előfeldolgozók



Segítségükkel jobban karbantartható kódot készíthetünk

Leggyakrabban használt előfeldolgozók

1. SaSS
2. Less.js
3. Stylus
4. CSS-Crush
5. Myth
6. Rework

## Sass



□ <http://sass-lang.com/>

```
$serif-font-stack: "Georgia", "Times New Roman", serif
$monospace-font-stack: "Cousin", "Courier"

body
 font: normal 18px/22px $serif-font-stack

pre, code
 font: 600 bold 18px/22px $monospace-font-stack
```

## Less



□ <http://lesscss.org/>

```
<head>
 <!-- Reference Less JavaScript file -->
 <script src="path/to/your/less.js"></script>
 <link rel="stylesheet/less" type="text/css" href="your/styles.less" />
</head>
```

```
.button {
 display: inline-block;
 width: 80%;
 max-width: 200px;
 border-radius: 5px;
 background-color: black;
 color: white;
 font-size: 14px;
 margin: 5px;
 padding: 8px;
}
```

```
.button-checkout- process {
 .button(); /* Mixin */
 background-color: silver;
}
```

A button szabályt tartalmát itt is felhasználjuk.

## Stylus



- <http://stylus-lang.com/>
- Több, mint 60 egyedi funkció

```
.button
 display inline-block
 border-color saturation(#000, 50%)
 border-radius 5px
```



```
.button {
 display: inline-block;
 border-color: #959595;
 -webkit-border-radius: 5px;
 -moz-border-radius: 5px;
 border-radius: 5px;
}
```

178

## Moduláris, újrafelhasználható CSS

<http://www.creativebloq.com/css3/create-modular-and-scalable-css-9134351>

<http://blog.rind.hu/ujrafelhasznalhato-css/>

<http://adamlaki.com/bem-modszertan/>



## Újrafelhasználható CSS



A statikus CSS újragondolásával, különböző konvenciók bevezetésével elérhetjük, hogy a kódunk újrafelhasználható, átlátható, karbantartható legyen.

Ilyen konvenciók:

- ❑ **Object Oriented CSS (OOCSS)**
- ❑ **Scalable and Modular Architecture for CSS (SMACSS)**
- ❑ **Don't Repeat Yourself CSS (DRY CSS)**
- ❑ **Block, Element, Modifier (BEM)**

## Don't Repeat Yourself CSS (DRY CSS)



- ❑ Csoportosítsuk az újrafelhasználható CSS deklarációkat
- ❑ Nevezzük el ezeket a csoportokat logikusan
- ❑ Adjunk szelektorokat a különböző csoportokhoz

**DRY CSS REVIEW**

Group re-usable properties	Name the groups logically	Add selectors to various groups
<pre>{   background-color: #fff;   border-color: #ccc; } {   background-color: #fff;   border-color: #888; }</pre>	<pre>#LIGHT-WHITE-BACKGROUND, .light-white-background {   background-color: #fff;   border-color: #ccc; } #MEDIUM-WHITE-BACKGROUND, .medium-white-background {   background-color: #fff;   border-color: #888; }  #LIGHT-RED-TEXT, #MEDIUM-RED-TEXT, #PALE-RED-BACKGROUND, #LIGHT-RED-BACKGROUND, BUTTON, SCREEN-BUTTON, SCREEN-BUTTON-HOVER, SCREEN-BUTTON-CHECK</pre>	<pre>#LIGHT-WHITE-BACKGROUND, .translation, .entry .no-caption, #full-article .entry img, .recent-comment .comment-text, .roundup h3, .post-header-sharing, #post-categories td label, #post-archives .roundup h3, .subscription-manager ol, .light-white-background {   background-color: #fff;   border-color: #ccc; } #MEDIUM-WHITE-BACKGROUND, .textarea:focus, input:focus, #author-email-form input:focus, #respond p input:focus, #wp77 input:focus, .medium-white-background {   background-color: #fff;   border-color: #888; }</pre>



## Object Oriented CSS (OOCSS)



- A struktúrát válasszuk el a designtól
  - ▣ A prezentációra vonatkozó deklarációkat válasszuk külön a struktúrára, pozícionálásra szolgáló deklarációktól!
- A konténert válasszuk el tartalomtól
  - ▣ A komponensek megjelenésének tartalmazó elemtől való függését meg kell szüntetni. Bármelyik objektumot tudni kell áttenni másik tartalmazó elembe úgy, hogy ne változzon meg a megjelenése!

## Scalable and Modular Architecture for CSS (SMACSS)



- **Base:** az alapdefiníciók, a css reset
- **Layout:** az oldalon fellelhető sorok és oszlopok beállításai. Grid css.
- **Module:** az oldalunk kis részegységei. Cikk lista, cikk részletes oldal, tartalmi dobozok.
- **State:** Az állapotok pl. aktív, inaktív, kinyitott, becsukott.
- **Theme:** az oldal azon stílusjegyei, amiktől igazán egyedi lesz az oldalunk.

## Block, Element, Modifier (BEM)



**Block:** A *blokk* a befoglaló elemünket, az oldal egy-egy nagyobb építőegységét jelenti, mint a fejléc, tartalom, oldalsáv, vagy a lábléc! Ez a rész lesz a kijelölőnk alapja!

**Element:** Az *elem* a *block* egy alkotó része. Az elemet a blokk után írjuk két darab alul vonással összekötve.

**Modifiers:** A módosítók segítségével a meglévő blokkok, elemek megjelenésén módosíthatunk. Például, ha van egy üzenet blokkunk, aminek több állapota van, egy semleges, egy figyelmeztetés/hiba visszajelzés, valamint egy sikeres visszaigazolás. Ebben az esetben ez 3 db külön osztály kijelölőt jelent úgy, hogy az utóbbi két esetben módosítanunk kell az eredeti elem megjelenésén. A módosítókat két kötőjellel jelöljük!

## Block, Element, Modifier (BEM)



```
.message {
 background: #eee;
 border-radius: 5px;
 color: #333;
 font-size: 16px;
 padding: 10px;
}

.message--alert {
 @extend .message;
 background: #da4531;
 color: #fff;
}

.message--success {
 @extend .message;
 background: #490a31;
 color: #fff;
}
```

<http://adamlaki.com/bem-modszertan/>

185

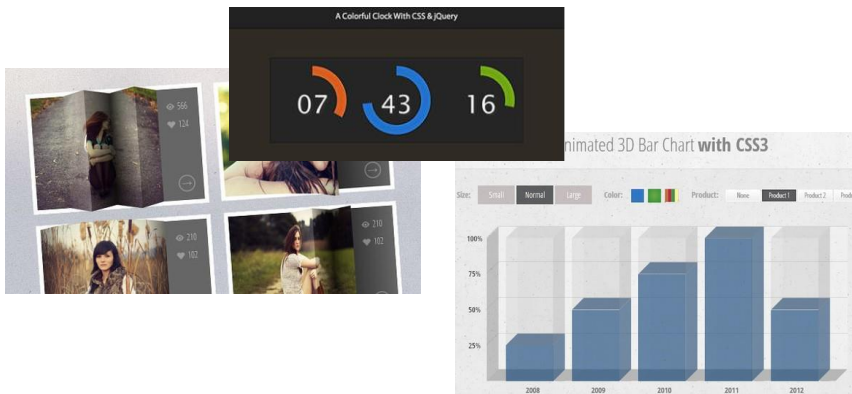
## Menő CSS példák



## 150 Cool CSS Examples And Tutorials



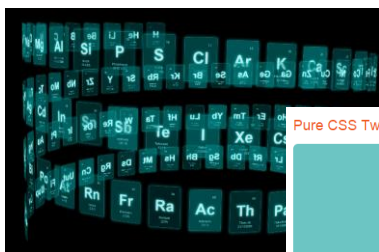
□ <http://1stwebdesigner.com/css-effects/>



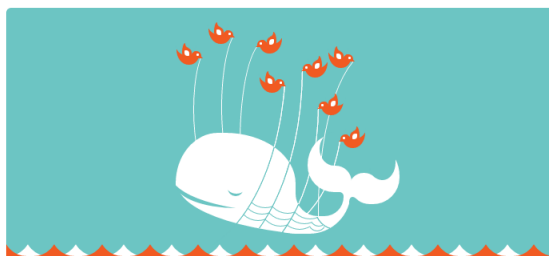
## 22 stunning examples of CSS3 animation



- <http://www.creativebloq.com/css3/animation-with-css3-712437>



Pure CSS Twitter 'Fall Whale'



188

Böngészők viselkedése és a CSS

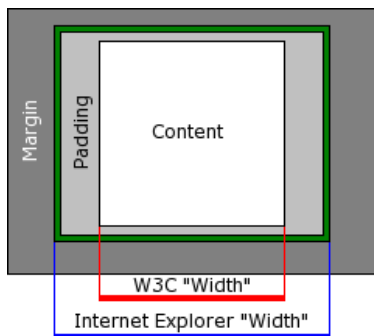


## Böngészők viselkedése és a CSS



### □ Internet Explorer dobozmodell hiba

- A szélesség rosszul számíródik, beleveszi a kitöltés és szegély méretet is.



### Megoldás?

Tudatni kell a böngésző-programmal, hogy szabványos CSS megjelenítést kérünk!

Vagyis használjuk a `<!DOCTYPE...>` megadást!

## Strict (standards), illetve Quirks mód



### □ A böngészők különböző üzemmódokban képesek a CSS oldalakat megjeleníteni

- Quirks: régi szabályok szerinti megjelenítés
- Strict: szabványkövető üzemmód

### □ Mi alapján váltunk?

- A doctype határozza meg
  - Nincs DTD megadás => Quirks mód
  - Legtöbb DTD megadás => Strict (standards) mód
  - Új, vagy ismeretlen DTD => Strict (standards) mód

### □ Lehet Quirks módba lépni DTD megadással?

- Igen. A böngészőprogramoknak van saját listája azon DTD-kről, amelyeknél Quirks módba kell kerülni
- <http://hsivonen.iki.fi/doctype/>

## Mi történik Quirks módban?



- A dobozmodell hibás. A szélesség és magasság a doboz szélességét és magasságát jelenti, nem pedig a tartalomét
- Függőleges margók bizonyos esetben összenyomódnak. (pl. p tag a td elemben)
- Az űrlapoknak Mozilla böngészőben 1 em-es alsó margó van beállítva
- Az elemek közti white-space karaktereknek jelentése lehet. (pl. <li> elemek új sorba törése nem kívánt megjelenést eredményezhet (megnövekedett függőleges hely)

<http://www.cs.tut.fi/~jkorpela/quirks-mode.html>

## Mi történik Quirks módban?



- Tag soup parsing.  
<p>text<table>...</table> kód esetén Firefox Quirks módban a táblázat a <p> elem része, szabványos üzemmódban a <table> lezárja a megnyitott <p> taget.
- CSS bizonyos részei nem működnek (pl. max-width az IE7-ben, vagy white-space: pre.
- Fix pozícionálás nem működik, a fixed pozícionálás helyett static kerül beállításra (IE7)

<http://www.cs.tut.fi/~jkorpela/quirks-mode.html>

## Mi történik Quirks módban?



- A nem szabvány szerinti osztály és azonosítók is elfogadásra kerülnek (amelyek ponttal vagy számmal kezdődnek)
- A kis- és nagybetű nem számít az osztály és id azonosítók megadásánál.
- A mértékegység nélküli paraméterekhez valamilyen rendszer szerint mértékegység társul. (10=>10px; f0f0f0=>#f0f0f0. Ez ellentmond a CSS szabályoknak, mert ezeket figyelmen kívül kellene hagynia a böngészőknek.

<http://www.cs.tut.fi/~jkorpela/quirks-mode.html>

## Mi történik Quirks módban?



- A relatív betűméret beállítások hibásak. A medium nagyobb mint az alap betűméret, small jelenti az alap betűméretet. Így az egész skála (xx-small,x-small, small, large, x-large, xx-large) eggyel nagyobb méretet eredményez, mint kellene.
- A táblázatban a betűméret beállítás az alap betűméretre vonatkozik, nem pedig a táblázatra megadott betűméretre.
- A tartalmazó elemtől nem öröklődik a font beállítás a táblázatok esetén.
- Blokk középre igazításánál a margin: 0 auto; beállítás nem vezet eredményre.

<http://www.cs.tut.fi/~jkorpela/quirks-mode.html>

## Mi történik Quirks módban?



- A kép függőleges igazításánál bizonyos esetekben a doboz alja nem a szöveg alapvonalához igazodik.
- A lebegtetett képeknél van egy alap vízszintes margó beállítás (3px a 0px helyett).
- IE-ben a gyökér elem a body, nem pedig a html. A body tagnél megadott margónak nincs hatása Quirks módban.
- A táblázatok százalékos méretmegadásánál szabványos módban a tartalmazó elem mérete a meghatározó. Quirks módban a rendelkezésre álló helyet veszi a böngésző.

<http://www.cs.tut.fi/~jkorpela/quirks-mode.html>

## Hasznos információk



- Különböző böngészőkben különböző Quirks mód van implementálva
  - ▣ Mozilla a Netscape 4.x-et imitálja
  - ▣ IE6+ az IE5-t imitálja
- Némelyik böngészőben van harmadik üzemmód is (Almost standards mode)
  - ▣ Firefox, Safari, Opera 7.5+
  - ▣ Különbség:
    - Táblázatcellák függőleges méretezése (line-height)
    - Képek blokkszintű (nem beágyazott) elemként történő felfogása

<http://www.quirksmode.org/css/quirksmode.html>



## DTD megadás (HTML) és a különböző módok



Doctype	NS6	Old Moz	Moz & Safari & Opera 10 & HTML5	Opera 9.0	IE 8 & Opera 9.5	IE 7 & Opera 7.10	IE 6 & Opera 7.0	Mac IE 5	Konq 3.2
<a href="#">None</a>	Q	Q	Q	Q	Q	Q	Q	Q	Q
<a href="#">&lt;!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 3.2 Final//EN"&gt;</a>	Q	Q	Q	Q	Q	Q	Q	Q	Q
<a href="#">&lt;!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.0//EN"&gt;</a>	S	S	S	S	S	A	A	A	A
<a href="#">&lt;!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01//EN"&gt;</a>	S	S	S	S	S	A	A	Q	A
<a href="#">&lt;!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.0//EN" "http://www.w3.org/TR/html4/strict.dtd"&gt;</a>	S	S	S	S	S	A	A	A	A
<a href="#">&lt;!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01//EN" "http://www.w3.org/TR/html4/strict.dtd"&gt;</a>	S	S	S	S	S	A	A	A	A
<a href="#">&lt;!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.0 Transitional//EN"&gt;</a>	Q	Q	Q	Q	Q	Q	Q	Q	Q
<a href="#">&lt;!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN"&gt;</a>	Q	Q	Q	Q	Q	Q	Q	Q	Q

<http://hsivonen.iki.fi/doctype/>

## DTD megadás (HTML) és a különböző módok



Doctype	NS6	Old Moz	Moz & Safari & Opera 10 & HTML5	Opera 9.0	IE 8 & Opera 9.5	IE 7 & Opera 7.10	IE 6 & Opera 7.0	Mac IE 5	Konq 3.2
<a href="#">&lt;!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN" "http://www.w3.org/TR/html4/loose.dtd"&gt;</a>	S	S	A	A	A	A	A	A	Q
<a href="#">&lt;!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN" "http://www.w3.org/TR/1999/REC-html401-19991224/loose.dtd"&gt;</a>	Q	S	A	A	A	A	A	A	Q
<a href="#">&lt;!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.0 Transitional//EN" "http://www.w3.org/TR/html4/loose.dtd"&gt;</a>	Q	Q	Q	Q	A	A	A	A	Q

<http://hsivonen.iki.fi/doctype/>

Doctype	NS6	Old Moz	Moz & Safari & Opera 10 & HTML5	Opera 9.0	IE 8 & Opera 9.5	IE 7 & Opera 7.10	IE 6 & Opera 7.0	Mac IE 5	Konq 3.2
<a href="#">&lt;!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.1//EN" "http://www.w3.org/TR/xhtml11/DTD/xhtml11.dtd"&gt;</a>	S	S	S	S	S	A	A	A	A
<a href="#">&lt;!DOCTYPE html PUBLIC "-//W3C//DTD XHTML Basic 1.0//EN" "http://www.w3.org/TR/xhtml1-basic/xhtml1-basic10.dtd"&gt;</a>	S	S	S	S	S	A	A	A	A
<a href="#">&lt;!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Strict//EN" "http://www.w3.org/TR/xhtml1/DTD/xhtml1-strict.dtd"&gt;</a>	S	S	S	S	S	A	A	A	A
<a href="#">&lt;!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN" "http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd"&gt;</a>	S	S	A	A	A	A	A	A	Q
<a href="#">&lt;?xml version="1.0" encoding="UTF-8"?&gt;&lt;!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.1//EN" "http://www.w3.org/TR/xhtml11/DTD/xhtml11.dtd"&gt;</a>	S	S	S	S	S	A	Q	A	Q
<a href="#">&lt;?xml version="1.0" encoding="UTF-8"?&gt;&lt;!DOCTYPE html PUBLIC "-//W3C//DTD XHTML Basic 1.0//EN" "http://www.w3.org/TR/xhtml1-basic/xhtml1-basic10.dtd"&gt;</a>	S	S	S	S	S	A	Q	A	Q

<http://hsivonen.iki.fi/doctype/>

Doctype	NS6	Old Moz	Moz & Safari & Opera 10 & HTML5	Opera 9.0	IE 8 & Opera 9.5	IE 7 & Opera 7.10	IE 6 & Opera 7.0	Mac IE 5	Konq 3.2
<a href="#">&lt;?xml version="1.0" encoding="UTF-8"?&gt;&lt;!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Strict//EN" "http://www.w3.org/TR/xhtml1/DTD/xhtml1-strict.dtd"&gt;</a>	S	S	S	S	S	A	Q	A	Q
<a href="#">&lt;?xml version="1.0" encoding="UTF-8"?&gt;&lt;!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN" "http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd"&gt;</a>	S	S	A	A	A	A	Q	A	Q
<a href="#">&lt;!DOCTYPE HTML PUBLIC "ISO/IEC 15445:2000//DTD HTML//EN"&gt;</a>	Q	S	S	Q	Q	Q	Q	Q	Q
<a href="#">&lt;!DOCTYPE HTML PUBLIC "ISO/IEC 15445:2000//DTD HyperText Markup Language//EN"&gt;</a>	Q	S	S	S	S	A	A	A	Q
<a href="#">&lt;!DOCTYPE HTML PUBLIC "ISO/IEC 15445:1999//DTD HTML//EN"&gt;</a>	S	S	S	Q	Q	Q	Q	Q	Q
<a href="#">&lt;!DOCTYPE HTML PUBLIC "ISO/IEC 15445:1999//DTD HyperText Markup Language//EN"&gt;</a>	S	S	S	S	S	A	A	A	Q
<a href="#">&lt;!DOCTYPE html&gt;</a>	Q	S	S	S	S	A	A	A	

<http://hsivonen.iki.fi/doctype/>

Vége

