

Web-technológia

Cascading Style Sheets

Dr. Hatwágner F. Miklós

Széchenyi István Egyetem, Győr

https://github.com/wajzy/GKxB_INTM049.git

2020. február 12.

CSS: Cascading Style Sheets

- \approx lépcsőzetes/sorba kapcsolt stíluslapok
- *formázás, megjelenés* leírásának elválasztása a *tartalomtól* (HTML), előnyei:
 - külön fájlban tárolható, ami több weboldalhoz is használható, így csökken az összesített kódméret,
 - egységessé válik ezen oldalak megjelenése,
 - egymástól függetlenül, egyidejűleg lehet szerkeszteni a formát és a tartalmat,
 - gyorsabban módosítható a megjelenés, mert csak egy helyen kell változtatni,
 - hatékonyabbá válik a gyorstárazás,
- különféle médiára eltérő formázás lehetséges (pl. képernyő, nyomtatás)
- a CSS ereje
- hivatalos W3C oldal

Formázás CSS-sel (cssFormazas.css)

```
3 <head>
4   <title>CSS formázás</title>
5   <link rel="stylesheet" type="text/css" href="cssFormazas.css">
6   <meta charset="utf-8" />
7 </head>
8 <body>
9   <p class="kiemelt">CSS formázás</p>
10 </body>
```

```
1 .kiemelt {
2     font-size: larger;
3     font-style: italic;
4 }
```

Szelektor (selector)

```
.kiemelt {
```

```
font-size: larger;  
font-style: italic;
```

```
font-style: italic;
```

Érték (value)

Tulajdonság (property)

Deklaráció sablonja

```
szelektor {
    tulajdonság1: érték(ek);
    tulajdonság2: érték(ek);
    ...
    tulajdonságN: érték(ek);
}
```

Szelektor

Mit akarunk formázni?

Tulajdonság

Milyen tulajdonságán változtassunk?

Érték

Milyen legyen az új állapot?

- `/* megjegyzes */`
- végleges kódból célszerű elhagyni
- Lehet több soros is

CSS ellenőrző

p { font-style: italic; }

p { font-style: italic; }

```
#lablec { font-size: 10pt; }
```

```
#lablec { font-size: 10pt; }
```

Az id nem kezdődhet számjegy karakterrel!

```
* { font-size: smaller; }
```

```
* { font-size: smaller; }
```

Osztály (class attribútum alapján)

```

.kisbetus { font-size: small; } /* bármilyen HTML elemhez */
.kisbetus { font-size: small; } /* bármilyen HTML elemhez, rövid
alak */

```

```
p.voros { color: red; } /* csak adott (pl. <p>) HTML elemhez */
```

A class értéke nem kezdődhet számjeggyel, de lehet egyszerre több, szóközzel elválasztott értéke:

Apróbetűs piros bekezdés

Elemek csoportosítása

h1, h2, h3 { font-family: Arial; }

14 <p class="bevezeto">A magyarországi informatika kezdete a első
→ elektronikus számítógép elkészültétől eredeztethető. A szobányi m
→ éretű M-3-at szovjet tervek alapján építették meg, a magyar
→ szakembergárda pedig ezen kísérletezte ki, hogy miként lehet egy
→ számítógépet a népgazdaság érdekeit szolgáló mérnöki-, és gazdasá
→ gi számításokra felhasználni – például az Erzsébet-híd statikai
→ számításait ellenőrizték vele.</p>

15 <p>A mintegy kétévnyi építést követően hivatalosan 1959. január 21-én á
→ tadott <a href="https://itf.njszt.hu/termek_hardware/m-3-
→ szamitogep">M-3 számítógépre a <a href="https://ajovomultja.
→ hu/news/60-eves-az-elso-magyar-szamitogep">korabeli sajtóbeszámol
→ óban még számológépként hivatkoztak, mert a számítógép szó ké
→ söbb honosodott meg a magyar nyelvben.</p>

A C A B D +18 A

- 1 Külső fájlban (css kiterjesztés, `<link>` elem)
- 2 A `<head>` elembe ágyazott `<style>` elemben. Csak akkor ajánlott, ha egyetlen HTML fájlt kívánunk formázni ezekkel a stílusokkal.
- 3 Soron belül: a HTML elemek `style` attribútumának értékeként. Ismét **keveredik a tartalom a stílussal**, ezért általában **nem ajánlott** a használata!

```
<head>
  <title>Egyszerű szelektorok</title>
  <meta charset="utf-8" />
  <style>
    p { font-family: Garamond, "Times New Roman", Serif; }
    a { color: navy; }
    .forras {font-style: italic; }
    .bevezeto { font-weight: bold; }
  </style>
</head>

<h1 style="font-family: Arial, Sans-Serif;">Ma van a magyar
  ↪ informatika születésnapja</h1>
```

- 1 soron belüli formázások
- 2 külső és belső (<link>, <style> elemek) formázások
- 3 böngésző alapértelmezése

Azonos prioritás (pl. két külső stíluslap) esetén a később betöltött szabály felülírja a korábbi.

```

6      <style>
7          p { color: red; }
8      </style>
9      <link rel="stylesheet" type="text/css" href="utkozes1.css" />
10 </head>
11 <body>
12     <p>Első bekezdés.</p>
13     <p style="color: green;">Második bekezdés</p>

```

```
1 p { color: blue; }
```

Első bekezdés.

Második bekezdés

utkoztes2.html

```
6      <link rel="stylesheet" type="text/css" href="utkozes1.css" />
7      <style>
8          p { color: red; }
9          p { color: orange; }
10     </style>
11 </head>
12 <body>
13     <p>Első bekezdés.</p>
```

utkozes1.css

```
1 p { color: blue; }
```

Első bekezdés.

color

background-color

Háttérszín

kulcsszavakkal

Pl. red (vörös), green (zöld), blue (kék), white (fehér), black (fekete), ...

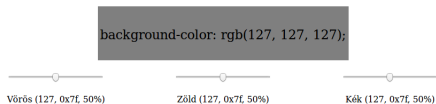
140 szabványos színekód

Hexadecimálisan, RGB összetevőkkel

Pl. narancsszín: #ff7f00, ahol # jelzi a 16-os számrendszerbeli alakot, ff a vörös (Red), 7f a zöld (Green) és 00 a kék (Blue) összetevő intenzitása 8 biten előjel nélkül, fixpontosan. Additív színkeverés.

rgb(red, green, blue), ahol mindhárom összetevő lehet 0-255 közötti decimális egész, vagy 0-100%. Pl. rgb(255,0,0) vagy rgb(100%, 0%, 0%) vörös színt eredményez.

`rgb(red, green, blue, alpha)`, ahol a színösszetevőket egy átlátszóság érték követi $[0, 1]$).



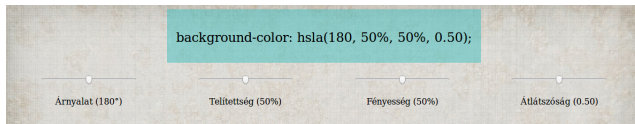
szinek1.html

hsl() függvény

hsl(hue, saturation, lightness), ahol hue az árnyalat, $[0, 360]$ fok közötti elfordulás a színceréken. Pl. 0° a vöröshöz, 120° a zöldhöz, 240° a kékhez tartozik. saturation a telítettség, százalékban. A 0% a színinformáció hiányát (szürkeség) jelzi, 100% a teljes színezettséget. lightness a fényesség, szintén százalékban. A 0% mindig fekete, a 100% mindig fehér színt ad.

hsla() függvényrel

A fentiek kiegészülnek átlátszósággal.

[szinek2.html](#)

Ennek meg a háttere legyen citromsárga!

HTML elemek háttérével kapcsolatos tulajdonságok:

`background-color`

A háttér színe. Alapértelmezetten transparent, azaz átlátszó.

`background-image`

Háttérkép, amivel alapértelmezés szerint kicsempézi az elem teljes területét (margókat nem). Alapértéke none, nincs háttérkép. Az `url()` függvény paramétereként adható meg a képfájl, pl.

```
background-color: url("hatter.png");
```

Megadhatók [színátmenetek](#) is.

A szöveg maradjon **olvasható** a háttéren!

background-repeat

Háttérkép csempézési iránya

- repeat mindkét irányban, túlnyúló részek levágásával, alapértelmezés
- repeat-x csak vízszintesen
- repeat-y csak függőlegesen
- no-repeat csak egyszer, alapértelmezetten a bal felső sarokban
- round torzítja a képet a vágás elkerülésére
- space csak annyiszor ismételi, ami vágás nélkül elfér, közöttük helyet hagy

Két érték megadásakor az első a vízszintes, második a függőleges irányra vonatkozik.



background-repeat:

- ☒ Mindkét irányban csempéz (repeat)
- ☐ Vízszintesen csempéz (repeat-x)
- ☐ Függőlegesen csempéz (repeat-y)
- ☐ Csak egyszer jelenik meg (no-repeat)
- ☐ Elosztja térközökkel (space)
- ☐ Torzít (round)

hatter.html

Igazítás, a vízszintes és a függőleges pozíciót várja. Ha egyet kap, a másik center lesz.

- Függőlegesen: left, center, right
- Vízszintesen: top, center, bottom
- Mindkettőnél lehet százalékot, vagy egyéb CSS mértékegységet (pl. képpont) használni.

100

```

7      p {
8          background-image: url("css3.svg");
9          background-size: 10%;
10         background-repeat: no-repeat;
11     }

15     <h1>Balra fent</h1>
16     <p>Lorem ipsum dolor sit amet, consectetur adipiscing elit.

24     <h1>Jobbra lent</h1>
25     <p style="background-position: right bottom">Aenean ut suscipit

37     <h1>Középen</h1>
38     <p style="background-position: center">Ut semper, ante pretium

53     <h1>Jobb alsó saroktól 10%-ra</h1>
54     <p style="background-position: 90% 90%">Lorem ipsum dolor sit

```

background-attachment

- `scroll` a háttér együtt gördül az oldallal, alapértelmezés
- `fixed` rögzített háttér
- `local` az elem tartalmával együtt gördül a háttér

A logo mindig a jobb alsó sarokban: [rogzites1.html](#)

Két bekezdés között kilátszik a háttérben rögzített logo: [rogzites2.html](#)

background

Rövidítés: egy összetett tulajdonsággal sok egyszerű tulajdonság értéke állítható be.

Értékek sorrendje rögzített, de tetszőleges számú érték elhagyható!

```
background: background-color background-image background-repeat
           background-attachment background-position
```

pozicio1.html


```
p {
  background-image: url("css3.svg");
  background-size: 10%;
  background-repeat: no-repeat;
}
```

pozicio2.html

```
p {
  background: url("css3.svg") no-repeat;
  background-size: 10%;
}
```

- `auto`: Alapértelmezés, eredeti méret.
- `szélesség`, `magasság`: utóbbi elhagyásával `auto`-t feltételez. Használhatók CSS mértékegységek és százalékok (a **szülő elem mérete a 100%**, nem a sajátja!).
- `cover` Addig nyújt és vág, amíg le nem fedi a szülő elem teljes területét.
- `contain` Addig nyújt, amíg egyszer bele nem fér a háttér a szülő elembe.



☐ Eredeti méret (auto)
☐ Teljes lefedés (cover)
☒ Maximális méret vágás nélkül (contain)
☐  100%

meret.html

Induljon ki a [rogzites2.html](#) fájlból, és alakítsa át a jobb oldali ábrának megfelelően!

- Az írásszín legyen világos szürke!
- A teljes oldal háttere legyen kék (RGB-összetevők: 0, 145 és 190)!
- A `<div>` elem háttereként állítsa be a [HTML5sticker.png](#) fájlt!
- Ennek helyzete ne függjön a görgetéstől!
- Helyezze el azt a képernyő közepén!
- A képet méretezze aránytartó módon úgy, hogy éppen kitöltse a rendelkezésre álló helyet!
- Próbálja mindezt a lehető legkevesebb CSS tulajdonság felhasználásával elérni!

rogzites3-mo.html, HTML5sticker.png

volutpat. Praesent congue, augue ultricies placerat auctor, neque libero blandit dolor, eu suscipit ante quam ac eros. Vivamus id massa mattis, fermentum tortor in, dignissim ex. Morbi faucibus, sapien non maximus hendrerit, enim lectus suscipit ex, sed volutpat erat odio eget tortor. Curabitur eu ullamcorper dolor. Phasellus vehicula sit amet urna finibus tempor. Nulla in sodales magna.

I've seen the
FUTURE
It's in my
BROWSER



Ut semper, ante pretium aliquam commodo, libero turpis gravida metus, eu semper justo quam at risus. Maecenas vitae libero massa. Proin sodales erat ex, sit amet consequat massa venenatis sit amet. Sed et luctus tortor, eu vestibulum libero. Curabitur euismod nisi sem, vel elementum nibh posuere id. Pellentesque tincidunt sed dui non viverra. Sed suscipit mollis lobortis. Vestibulum facilisis volutpat justo in feugiat. Pellentesque sit amet