# Web-technológia Cascading Style Sheets

Dr. Hatwágner F. Miklós

Széchenyi István Egyetem, Győr

https://github.com/wajzy/GKxB\_INTM049.git 2020. február 2.

## CSS: Cascading Style Sheets

- ≈ 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



## Elavult módszer (htmlFormazas.html)

## Formázás CSS-sel (cssFormazas.ht ml)

```
<head>
      <title>CSS formázás</title>
      <link rel="stylesheet" type="text/css" href="cssFormazas.css">
      <meta charset="utf-8" />
    </head>
    <body>
      CSS formázás
10
    </body>
```

## Formázás CSS-sel (cssFormazas.css)

```
. kiemelt {
 font—size: larger;
 font-style: italic;
```

```
Szelektor (selector)

.kiemelt {
    font-size: larger;
    font-style: italic;
}

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?

### Megjegyzések a CSS-ben:

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

CSS ellenőrző

```
HTML elem neve
   p { font-style: italic; }

Egyedi azonosító (id attribútum) alapján
   #lablec { font-size: 10pt; }
   Az id nem kezdődhet számjegy karakterrel!

Univerzális szelektor, mindenre illeszkedik
   * { 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. ) 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; }

```
egyszeruSzelektor1.html
```

```
<head>
       <title>Egyszerű szelektorok</title>
       <link rel="stylesheet" type="text/css"</pre>
                                               href="egyszeruSzelektor.css">
       <meta charset="utf-8" />
     </head>
8
     <body>
       <article>
10
         <header>
           <h1>Ma van a magyar informatika születésnapja</h1>
11
12
           Forrás: <a href="https://njszt.hu/hu/news"</pre>

→ /2020-01-21/ma-van-magyar-informatika-szuletesnapja">NJSZT</a>,
               \rightarrow <time datetime="2020-01-21,100:00">2020 január 21 </time>
13
         </header>
```

## egyszeruSzelektor1.html

```
14
         A magyarországi informatika kezdete a első
             ← elektronikus számítógép elkészültétől eredeztethető. A szobányi m
             → éretű M—3—at szoviet tervek a|apján építették meg. a magyar
             → szakembergárda pedig ezen kísér|etezte ki, hogy miként lehet egy
             → számítógépet a népgazdaság érdekeit szo|gá|ó mérnöki—. és gazdasá
             → gi számításokra felhasználni — például az Erzsébet—híd statikai

→ számításait ellenőrizték vele 
15
         A mintegy kétévnyi építést követően hivatalosan 1959. január 21-én á

→ tadott <a href="https://itf.njszt.hu/termek_hardware/m-3-"
</p>
             → szamitogep">M-3 számítógépre</a> a <a href="https://ajovomultja.
             → hu/news/60-eves-az-elso-magyar-szamitogep">korabeli sajtóbeszámol
             → óban</a>> még számológépként hivatkoztak, mert a számítógép szó ké

→ sőbb honosodott meg a magyar nyelvben
```

## egyszeruSzelektor1.css

```
h1, h2, h3, h4, h5, h6 {
  font—family: Arial, Sans—Serif;
}

p { font—family: Garamond, "Times_New_Roman", Serif; }

a { color: navy; }

forras {font—style: italic; }

bevezeto { font—weight: bold; }
```

#### Ma van a magyar informatika születésnapja

Forrás: NJSZT, 2020. január 21.

A magyarországi informatika kezdete a első elektronikus számítógép elkészültétől eredeztethető. A szobányi méretű M-3-at szorjet tervek alapján építették meg, a magyar szakembergárda pedig ezen kisérletezte ki, hogy miként lehet egy számítógépet a népgazdaság érdekelt szolgáló mérnöki-, és gazdasági számításokra felhasználni - például az Erzsébet-hid statikai számításalt ellenőrizték vele.

A mintegy kétévnyi építést követően hivatalosan 1959. január 21-én átadott M-3 számítógépre a korabeli sajtóbeszámolóban még számológépként hivatkoztak, mert a számítógép szó később honosodott meg a magyar nyelvben.

Bár ún. nulladik generációs számítógépet már az M-3 előtt néhány évvel is építettek a Budapesti Műszaki Egyetemen, Kozma László MESz-1 nevű gépe még telefonközpontokban használatos elektromágneses relékből épült fel, az M-3-mal ellentétben még nem felelt meg a Neumann-elveknek, amelyek előírják, hogy egy számítógépnek teljes egészében elektronikus módon kell működnie, és egy belső tárban kell tárolnia a programutasításokat.

A Magyar Tudományos Akadémia Kibernetikai Kutatócsoportjában megépített, első generációs M-3-ast tudományos számítások embernél gyorsabb végrehajtására tervezték. Noha a gép csupán másodpercenként 30 számítási műveletet tudott elvégezüt (mig a mai gépek processzorai esetében már több tizezer millió tusaitássa Iszámohatunk másodpercenként), az M-3-rai tri porajmank révén így is számos gyakortai feladatra használták fel. E gépen készelt az épülőt szászebet hát sáttalás zámítássavá laspozatosp rogram, de pédalú szállítási köldsége opimalzálásáhar, részesekt visztesébet sáttalásán számátssavá ta spozatosp rogram, de pédalú szállítási köldsége opimalzálásáhar, részesekt visztesébet sáttalásán számátssavá ta spozatosp rogram, de pédalú szállítási köldsége opimalzálásáhar, részesekt visztesébet ástatását számátssavá ta számát számát számát számát számát számát számát számát számátasák számát számát

## Háromféle helyen lehet stílusokat megadni:

- I 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!

## egyszeruSzelektor2.html

```
3
     <head>
       <title>Egyszerű szelektorok</title>
       <meta charset="utf-8"/>
       <style>
         p { font-family: Garamond, "Times_New_Roman", Serif; }
         a { color: navy; }
9
          .forras {font—style: italic; }
10
          .bevezeto { font-weight: bold; }
11
       </style>
12
     </head>
           <h1 style="font-family: Arial, Sans-Serif;">Ma van a magyar
16
               → informatika születésnapja</h1>
```

Ha több előírás is vonatkozik ugyanannak az objektumnak a formázására, elsőként a forrás prioritása dönt (csökkenő sorrendben):

- 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 korábbit.

### utkozes1.html

### utkozes1.css

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

Első bekezdés.

Második bekezdés

### utkozes2.html

### utkozes1.css

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

Első bekezdés.

◆□▶ ◆□▶ ◆□▶ ◆□▶ □ り<0</p>

Számtalan dolognak beállítható a színe CSS tulajdonságokkal, pl.:

#### color

Szöveg írásszíne

### background-color

Háttérszín

Szín, mint a tulajdonság értéke megadható:

#### kulcsszavakkal

Pl. red (vörös), green (zöld), blue (kék), white (fehér), black (fekete), ... 140 szabványos színkód

### 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() függvénnyel

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.

### rgba() függvénnyel

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



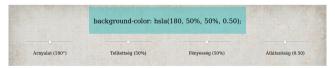
szinek1.html

### hsl() függvénnyel

hsl(hue, saturation, lightness), ahol hue az árnyalat, [0, 360] fok közötti elfordulás a színkeréken. 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() függvénnyel

A fentiek kiegészülnek átlátszósággal.



szinek2.html

Induljon ki a szinezes.html fájlból!
Kapcsolja ezt össze egy külső stíluslappal,
majd érje el, hogy a jobb oldali ábrának
megfelelő színekben pompázzon! Próbáljon
minél több féle szín megadási módszert
alkalmazni! Törekedjen a lehető
legtömörebb CSS szabályok megalkotására!

### szinezes-mo.html, szinezes-mo.css

### Az első szintű címsorok legyenek barnák!

Ez a bekezdés legyen vörös!

Ez pedig zöld, de az utolsó szava legyen viágoszöld!

### Az első szintű címsorok legyenek barnák!

A második és harmadik szintű címsorok legyenek kékek!

Ez egy szürke stílusú bekezdés.

Ez itt egy harmadik szintű címsor.

Ez is szürke.

Ennek meg a háttere legyen citromsárga!

HTML elemek hátterével 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 ismétel, 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

### background-position

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

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

### pozicio1.html

```
7
8
          background—image: url("css3.svg");
          background-size: 10%;
9
          background-repeat: no-repeat;
10
11
      <h1>Balra fent</h1>
15
      Lorem ipsum dolor sit amet, consectetur adipiscing elit.
16
24
      <h1>Jobbra |ent</h1>
25
      Aenean ut suscipit
37
      \langle h1\rangleKözépen\langle h1\rangle
38
      Ut semper. ante pretium
53
      \langle h1 \rangle Jobb alsó saroktól 10\% ra\langle h1 \rangle
      Lorem ipsum dolor sit
54
```

### 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.



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 fájl!
- 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!

