

JavaScript alapozó 2.

A CSS alapjai



Tartalom

- CSS (Cascading Style Sheets)
- Selectorok
- Property-k
- CSS Precedencia
- CSS helye a kódban
- Doboz modell
- Functionök
- Media query-k
- CSS keretrendszerek
- Superset nyelvek

CSS (Cascading Style Sheets)

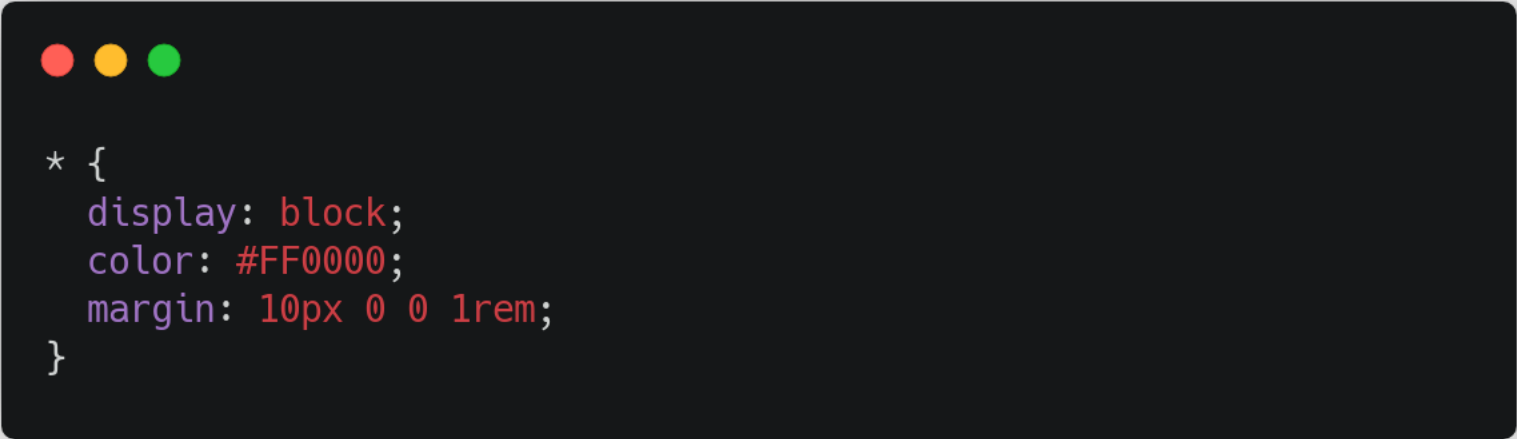
- Leíró nyelv HTML elemek formázására
- Oldal struktúrájának, elemek elrendezésének definiálására
- Animációk létrehozására
- CSS3-al foglalkozunk alapvetően, de ennél többet mond el a <https://caniuse.com>
- Működése:
 - CSS selectorokkal elemeket választunk ki
 - És az ezekre vonatkozó szabályok definiálásával módosítjuk a megjelenést

Selectorok

- Egy vagy több HTML elem kiválasztására
- Főbb típusai:
 - class (.container)
 - id (#container)
 - element (div)
 - pseudo-class (:hover)
 - attribute ([target=_blank])
 - Kombinált selectorok (div.container:hover)

Property-k

- CSS tulajdonságok megadása egy-egy kiválasztott elemhez
- Kulcs érték párok (kulcs: érték) formában



```
* {  
  display: block;  
  color: #FF0000;  
  margin: 10px 0 0 1rem;  
}
```

Selectorok - Class



```
.container {  
  background-color: #9CAABA;  
  width: 100%;  
}
```



```
<div class="container"></div>
```

Selectorok - Id



```
#container {  
  background-color: #9CAABA;  
  width: 100%;  
}
```



```
<div id="container"></div>
```

Selectorok - Element



```
div {  
  background-color: #9CAABA;  
  width: 100%;  
}
```



```
<div>  
  <p>Lorem</p>  
</div>  
<div>  
  <p>Ipsum</p>  
</div>
```


Selectorok – Pseudo-class

```
div {  
  background-color: #9CAABA;  
  width: 100%;  
}  
  
div:hover {  
  background-color: #475D75;  
  color: #FFF;  
}
```

```
<div>  
  <p>Lorem</p>  
</div>  
<div>  
  <p>Ipsum</p>  
</div>
```

Selectorok - Attribute



```
[target=_blank] {  
  background-color: #9CAABA;  
  width: 100%;  
}
```



```
<a target="_blank">  
  Lorem  
</a>  
<div>  
  <p>Ipsum</p>  
</div>
```

Selectorok – Kombinált selector

```
div {  
  background-color: #9CAABA;  
  width: 100%;  
}  
  
div.container:hover {  
  background-color: #475D75;  
  color: #FFF;  
}
```

```
<div class="container">  
  <p>Lorem</p>  
</div>  
<div>  
  <p>Ipsum</p>  
</div>
```

CSS Precedencia

Meghatározza, hogy ha egy elemre egy adott tulajdonságot (property-t) több szabály is beállítana, akkor melyik fog érvényre jutni

1. !important kulcsszóval jelölt tulajdonság
2. Selector specifikusság szerint (ld. következő dia)
3. Deklarálás sorrendje szerint (később deklarált az erősebb)

CSS Precedencia - Specifikusság szerint (csökkenő sorrendben)

1. Style attribútumban definiált értékek
2. Kombinált selector (div.container:hover)
3. id (#container)
4. class (.container)
5. attribute ([target=_blank])
6. element (div)
7. *
8. Örökölt tulajdonságok

Hova írhatunk CSS kódot?

- HTML style tag-be
- HTML elem style attribútumába
- CSS fájlba —> Ebben az esetben ezt a fájlt be kell hivatkozni a HTML fájlba (bonyolultabb esetben akár a JavaScript fájlba való hivatkozással is megoldható)

HTML style tag

```
<html>
  <head>
    <style>
      div {
        background-color: #9CAABA;
        width: 100%
      }

      div.container:hover {
        background-color: #475D75;
        color: #FFF;
      }
    </style>
  </head>

  <body>
    <div class="container">
      <p>Lorem</p>
    </div>
    <div>
      <p>Ipsum</p>
    </div>
  </body>
</html>
```

HTML elem style attribútum

```
<html>
  <head>
    <style>
      div.container:hover {
        background-color: #475D75 !important;
        color: #FFF;
      }
    </style>
  </head>

  <body>
    <div class="container" style="background-color: #9CAABA; width: 100%">
      <p>Lorem</p>
    </div>
    <div>
      <p>Ipsum</p>
    </div>
  </body>
</html>
```


CSS fájl hivatkozás

styles.css

```
:root {
  --primary-color: #9CAABA;
}

div {
  --container-width: 100%;

  background-color: var(--primary-color);
  width: var(--container-width);
}

div.container:hover {
  background-color: #475D75 !important;
  color: #FFF;
}
```

index.html

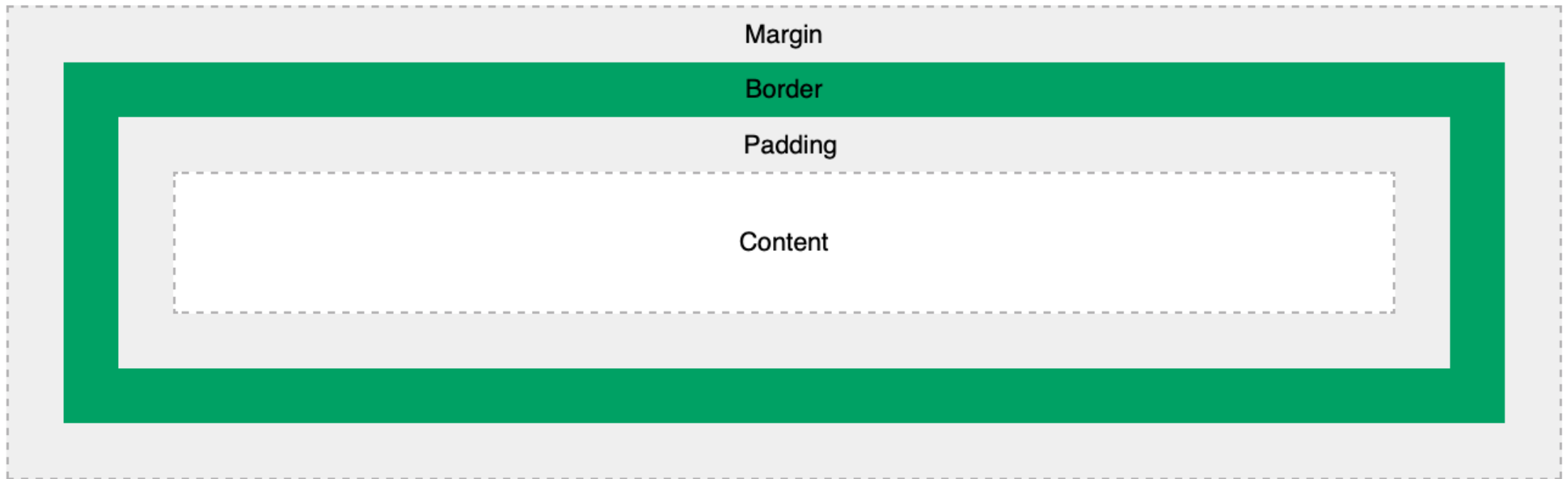
```
<html>
  <head>
    <link rel="stylesheet" href="./styles.css">
  </head>

  <body>
    <div class="container">
      <p>Lorem</p>
    </div>
    <div>
      <p>Ipsum</p>
    </div>
  </body>
</html>
```

Doboz modell (Box model)

- Minden HTML elemet képzeljünk el úgy mint amit körbevesz egy láthatatlan doboz
- Minden ilyen doboz az alábbi részekből áll
 - Content (Tartalom) — A doboz tartalma (kép, szöveg, stb.)
 - Padding — Térköz a doboz tartalma és a széle között
 - Border (Szegély) — A doboz szegélye
 - Margin — Térköz a doboz és más elemek között

Doboz modell (Box model)



CSS változók

- Egy érték tárolására szolgálnak
- JS kódból is elérhetőek és módosíthatók
- Scope-al rendelkeznek

CSS változók

```
<html>
  <head>
    <style>
      :root {
        --primary-color: #9CAABA;
      }

      div {
        --container-width: 100%;

        background-color: var(--primary-color);
        width: var(--container-width);
      }

      div.container:hover {
        background-color: #475D75 !important;
        color: #FFF;
      }
    </style>
  </head>

  <body>
    <div class="container">
      <p>Lorem</p>
    </div>
    <div>
      <p>Ipsum</p>
    </div>
  </body>
</html>
```

Functionök

- Néhány speciális előre definiált CSS függvény létezik (https://www.w3schools.com/cssref/css_functions.asp)
- Bonyolultabb értékek definiálására
- Például:
 - width: calc(100% - 15px)
 - color: rgba(255, 0, 0, 0.5)
 - margin: var(--margin-top) 0 0 1rem

Media query-k

- Reszponzivitás alapja
- Eszköz tulajdonságoktól függő CSS definíciók megadása
- Előre definiált tulajdonságok kombinációjával lehet eszköz típusokra szűrni
- Például:
 - @media screen and (max-width: 992px)
 - @media only screen and (orientation: portrait)
 - @media screen and (max-width: 992px) and (orientation: portrait)
 - @media screen and (max-width: 992px) and (orientation: portrait), (min-width: 1200px)

Media query-k

```
● ● ●  
  
@media screen and (max-width: 992px) and (orientation: portrait) {  
    .container {  
        font-size: 12px;  
    }  
}  
  
@media screen and (min-width: 1200px) {  
    .container {  
        font-size: 32px;  
    }  
}
```


Kódolás



CSS keretrendszerek

- Sokrétű CSS definíciókat adnak, egységes megjelenés
- Jellemzően:
 - lefedik a form elemeket
 - reszponzív megoldásokat adnak
 - különböző extra megoldásokat tartalmaznak (pl.: modal)
- Például:
 - [Bootstrap](#)
 - [Materialize](#)
 - [Tailwind](#)

Superset nyelvek

- CSS kényelmetlenségeit javítják ki jellemzően
- Jobb megoldást adnak a strukturálásra
- Például:
 - [SCSS](#) / SASS
 - [LESS](#)

Házi feladat

- Csináljunk az oldal aljára egy div-et ami:
 - 1200px felett 80% széles, az alatt 100% széles
 - A magassága legyen a tartalom által meghatározva automatikusan
 - Legyen neki border-je és legyenek lekerekítve a sarkai
 - Rakjunk bele egy pár soros szöveget (<https://www.lipsum.com/>)
 - Rakjunk bele két képet (<https://unsplash.com/>)
 - A szöveget igazítsuk vízszintesen középre
 - A két kép egymás mellett jelenjen meg, és ne érjenek össze (Ez szorgalmi)
 - A tartalom sehol ne érjen hozzá a div széléhez, és a div se érjen hozzá a közelében lévő más elemekhez

Házi feladat

[Privacy notice](#)

Send



Lorem ipsum dolor sit amet, consectetur adipiscing elit. Proin finibus leo non sagittis volutpat. Nullam mauris lectus, euismod a dui quis, placerat aliquam erat. Donec scelerisque justo nec egestas ornare. Cras commodo ornare facilisis. In in posuere elit, sit amet consectetur metus. Nulla egestas imperdiet lorem nec facilisis. Sed ac pulvinar leo. Nam rhoncus augue sed libero sagittis ultricies. Etiam vehicula commodo est, nec ornare risus auctor sit amet. In hac habitasse platea dictumst. Etiam at sapien ut sem dictum tempus. Cras rutrum sapien mi, ac aliquam nibh lobortis vel. Suspendisse libero leo, bibendum maximus dignissim non, rhoncus nec dolor. Mauris venenatis, sapien quis malesuada vestibulum, purus velit porttitor magna, eget commodo elit ligula non lectus. Phasellus laoreet velit in ante ultricies cursus. Maecenas eget turpis rhoncus, tincidunt purus eget, imperdiet erat.



Extra feladat

- Definiáljunk egy listát HTML-ben (ul, li) és vegyünk fel bele legalább 4 elemet
- Oldjuk meg, hogy minden páros számú elem a listában kapjon zöld hátteret, a páratlanok pirosat, az első elem felső 2px-es bordert, az utolsó elem pedig 2px-es alsó bordert. Minden elem között legyen 1px-es border.