

Második webtanfolyam

Disclaimer

Akkor mi a cél?

Erre minek kell három alkalom?

Az alkalom tematikája

Az alapoktól a flex-ig

Izgalmasabb részletek

A flexen túl

Hogyan tovább?

Házifeladat

Merülj el egy témában

Izgalmas játékok

Gyere el a következő alkalomra

Disclaimer

Két óra alatt nem lehet megtanulni se designolni, se ügyesen CSS-ezni. Ez itt nem is cél.

A CSS kifejezetten egy olyan része a webfejlesztésnek, amit nagyrészt Google keresések végtelen halmazával jellemezhetünk a legjobban.

Akkor mi a cél?

Úgy lehet valamit megtanulni, ha foglalkozunk vele. Ahhoz, hogy időt szánjunk rá, motivációra van szükségünk. Itt ezt szeretném felkelteni bennetek.

Erre minek kell három alkalom?

A motiváció mellett pár alapismeretet is szeretnénk átadni, amivel már önállóan is képesek lesztek fejlődni.

Rossz érzés, amikor egy új dologba kezdünk és minden ijesztő, bonyolult, távoli. Én ezt az érzést szeretném csökkenteni. Meg persze pár gyakorlati tippet is adni.

TL;DR: Ha idáig eljutottál, ajánlom a Hogyan tovább? rész átnézését, hátha lesz benne valami, ami érdekel.

Az alkalom tematikája

Ha más forrásból is meg szeretnéd ismerni az első lépéseket, akkor ajánlom [ezt az oldalt](#).

A kezdetekben nagyon jó lesz, az advanced témák kapcsán kicsit hiányos.

Az alapoktól a flex-ig

Kétféle nyelvet használtunk, a **HTML**-t és a **CSS**-t.

HTML

Fő részei:

<html>

Ilyet VS Codeban a `html : 5` rövidítés beírásával, majd az `enter` lenyomásával készíthetünk (feltéve, hogy a fájl kiterjesztése `.html`).

<head>

Ide sok mindent írhatunk, egyelőre arra használtuk, hogy a `css` fájlainkat linkeljük a dokumentumunkhoz. Ezt VS Codeban a `link : css` rövidítés majd `enter` lenyomásával tehetjük meg. A keletkezett elem mezőit értelemszerűen töltjük ki.

</head>

<body>

Ide tudjuk írni magát a dokumentumunkat, a következőkben felsorolok pár elemet, amit használtunk.

<main> Akár további elemeket is írhatunk ide. **</main>**

<section> Akár további elemeket is írhatunk ide. **</section>**

<div> Akár további elemeket is írhatunk ide. **</div>**

<p> Ide (nagyjából) csak szöveget írhatunk, más elemet nem ágyazhatunk be. **</p>**

<h1> Ide (nagyjából) csak szöveget írhatunk, más elemet nem ágyazhatunk be. **</h1>**

</body>

</html>

Ezekről részletesebben itt.

Egyszer ezt érdemes átpörgetni.

Ha úgy érzed el vagy veszve a VS Codeben.

Illetve rövid leírás a VS Codeos shortcutokhoz.

CSS

Mi a `.css` fájlokat egy `css` mappába helyeztük, ami a későbbiekben egy jó szokáshoz fog vezetni.

Ez a nyelv a következőkből áll:

selector: Ezzel kijelölhetünk bizonyos elemeket (van, hogy egyet, van hogy az összeset).

property: Kiválaszthatunk vele bizonyos tulajdonságo(ka)t.

érték: A property-knek értéket adhatunk.

Ez mit is jelent?

Izgalmasabb részletek

Az oldalunk bodyja

```
<body>
  <main id="ez-egy-id">
    <section class="card">
      <p> Ez az én szövegem. </p>
    </section>
  </main>
</body>
```

Természetesen a **<head>**-ben linkeltük az alábbi CSS fájlt.

Az ehhez tartozó CSS

```

html {
  /*
  Színátmenetet állítunk be háttérként.
  További infók: https://www.w3schools.com/css/css3\_gradients.asp
  */
  background: linear-gradient(90deg, #fff9e0 0, #ffd6d6 100%);
}

/*
A classok neve elé pontot írunk a kijelölésükhöz.
Az id-k neve elé pedig #-et, tehát például #ez-egy-id.
Az id selector erősebb, mint a class selector, az pedig
erősebb a sima tag-es kijelölésnél.
*/
.card {
  background: white;

  /*
  A chrome F12-es módjában tudjuk finomhangolni.
  Amúgy általában két árnyék szokott lenni,
  egy a direkt fénynek egy pedig az indirektnek.
  */
  box-shadow: 0 0 8px 1px rgba(0, 0, 0, 0.15);

  /* Ekvivalens ezzel: padding: 10px 20px 10px 20px; */
  padding: 10px 20px;

  /* Ekvivalens ezzel: margin: 30px 30px 30px 30px; */
  margin: 30px;

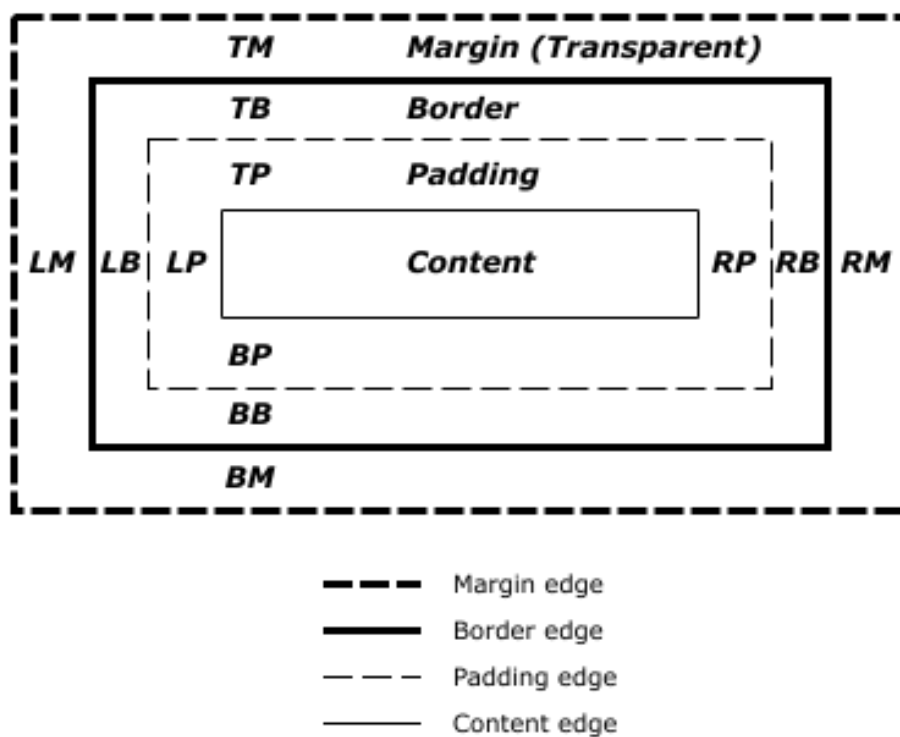
  /*
  Mennyire legyen lekerekített?
  Ha nagy számot adunk, akkor kör alakú lesz.
  */
  border-radius: 15px;

  /* Pont, mint MS Wordben. Lehetne még: left, right és justify. */
  text-align: center;

  /* Ha az első betűtípust nem találja, akkor a következővel próbálkozik. */
  font-family: "Open Sans", Arial, sans-serif;
}

```

Ennek kapcsán hasznos lehet a **box model**:



Ugyanez kifejtve részletesebben.

Még miről volt szó?

```
/*  
Hogyan csináljunk insetes képet.  
Erről a következő alkalmon kicsit részletesebben.  
*/  
.image-container {  
  
  /* Érdeklődőknek: https://css-tricks.com/box-sizing/ */  
  width: 100px;  
  height: 100px;  
  
  border-radius: 1000px;  
  
  /*  
  Nem maga mögé, hanem saját magába veti az árnyékát  
  az inset hatásásra.  
  */  
  box-shadow: inset 0 0 2px 2px rgba(0, 0, 0, 0.33);  
}
```

```

/* Itt az adott css fájlból relatíven adjuk meg a kép forrását. */
background-image: url('../media/picture.jpg');
background-position: center;
background-size: cover;
}

/* Végül a lényeg. */
.flexible {
    display: flex;

    /* Alapértelmezetten row, ezen kívül, ami még érdekes az a column. */
    flex-direction: row;

    /* Törjön-e új sorba? Igen. */
    flex-wrap: wrap;

    /*
    flex-direction: row-nál az x-irányú elhelyezést írja le.
    flex-direction: column-nál az y-irányú elhelyezést írja le.
    */
    justify-content: space-evenly;

    /*
    Pont fordítva, mint az előbb:
    flex-direction: row-nál az y-irányú elhelyezést írja le.
    flex-direction: column-nál az x-irányú elhelyezést írja le.
    */
    align-items: center;
}

```

A flex-ről kicsit részletesebben.

A flexen túl

Ezen kívül természetesen egy csomó lehetőség van még a CSS-ben. Ezeket nem gond, ha nem tudjátok mind elsőre. Ha foglalkoztok vele, szép lassan a nagy részét meg fogjátok ismerni.

Ha elakadtok:

CSS referencia

[Google](#)

Bátran keressetek engem is, szívesen segítek:

[Facebook](#)

[email](#)

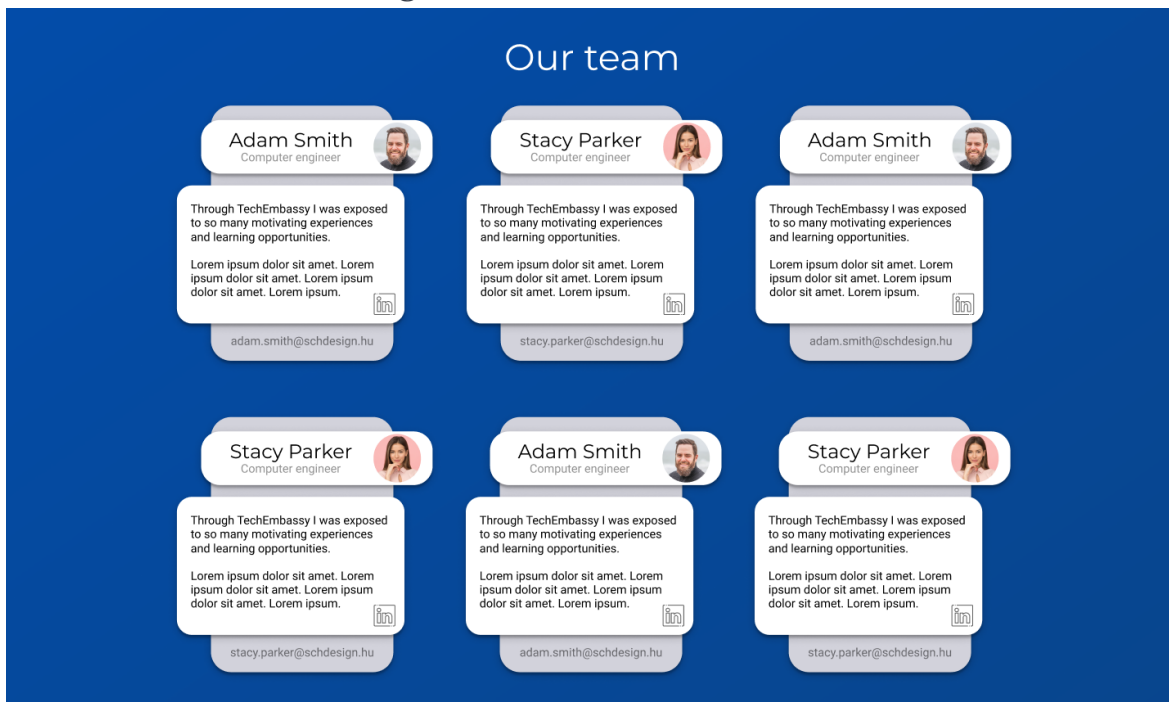
Hogyan tovább?

Házifeladat

Csináljatok valamit CSS-sel.

Bármit, ami érdekel, amit szépnek tartotok, vagy ami lehetőséget biztosít a gyakorlásra.

Ha nincs más ötletetek, ajánlom ennek a képnek (vagy csak egy részének) a HTML + CSS-ben való megvalósítását.



Merülj el egy témában

Ezt úgy teheted meg, hogy a dokumentum bármelyik linkjére klikkelve kicsit utánaolvasol egy-két témának (és természetesen utána ezt ki is próbálsd a gyakorlatban).

Izgalmas játékok

Hogyan működnek a CSS selectorok játék

Ez egész jó.

Hogy működik a flexbox játék

Ez is tetszett.

Bátraknak ízelítő a CSS 3D transzformációkba

Gyere el a következő alkalomra

Ott a webfejlesztés egy másik oldalába fogunk belenézni a JavaScripten keresztül.

Időpont

Hely

2019.02.21. 17:00-19:00

E-mailben küldjük.