

# WEBTERVEZÉS – GYAKORLATI JEGYZET

## 2. gyakorlat

HTML alapok II., a CSS alapjai

**Készítették:**

Cservenák Bence

Farkas Anikó

Savanya Sándor

### FIGYELEM!

A jegyzet folyamatosan készül, így előfordulhatnak benne apróbb hibák, hiányosságok, elírások. Ha valaki esetleg ilyenre találna, kérem, írjon a `Cservenak.Bence@stud.u-szeged.hu` címre, hogy mihamarabb javíthassuk.

## 1. A HTML alapjai (2. rész)

Folytassuk a HTML dokumentum törzsrészában (body) használható tagekkel való ismerkedést!

### A. Néhány fontosabb szövegelem

- `<p>...</p>`: bekezdés
  - előtte és utána térköz található
- `<pre>...</pre>`: előre formázott szöveg
  - alakhű megjelenítést biztosít
  - akkor használjuk, ha a szöveg jellegéből szemantikusan következik a tartalom szóközökkel, térközökkel való formázása
- `<br/>`: sortörés (páratlan tag)
- `<hr/>`: elválasztó vonal (páratlan tag)

```
<h1>Piramis</h1>
<hr/>
<pre>
    0
  0 0
0 0 0
0 0 0 0
</pre>
A piramis egy négyzet alapú <br/> gúla
alakú építmény.
```

### Piramis

```
0
0 0
0 0 0
0 0 0 0
```

A piramis egy négyzet alapú  
gúla alakú építmény.

- `<em>...</em>`: tartalom hangsúlyozása (szemantikus jelentés, általában dőlt)
- `<strong>...</strong>`: erős kiemelés (szemantikus jelentés, általában félkövér)

```
<em>Ricsi</em> ma vizsgázik prog2-ből. <br/>
Ricsi <em>ma</em> vizsgázik prog2-ből. <br/>
Ricsi ma vizsgázik <em>prog2-ből</em>. <br/>
<p><strong>Figyelem!</strong> A ketrecbe benyúlni veszélyes!</p>
```

*Ricsi* ma vizsgázik prog2-ből.  
Ricsi *ma* vizsgázik prog2-ből.  
Ricsi ma vizsgázik *prog2-ből*.

**Figyelem!** A ketrecbe benyúlni veszélyes!

**Megjegyzés:** A HTML tageket **ne** használjuk formázási célokra! Ha egy szövegrészt félkövérré vagy dőltté szeretnénk tenni, használjunk CSS-t!

- `<sub>...</sub>`: alsó index
- `<sup>...</sup>`: felső index
- `<q>...</q>`: idézet (az idézőjeleket automatikusan kiteszi)
- `<blockquote>...</blockquote>`: idézetblokk
- `<code>...</code>`: kódrészlet
- `<mark>...</mark>`: vizuális szövegkiemelés

```
Alsó index: x<sub>2</sub>, felső index: x<sup>2</sup> <br/>
<q>A Kalkulus még könnyű tárgynak számít</q>
<blockquote>Never gonna give you up <br/>
Never gonna let you down <br/>
Never gonna run around <br/>
And desert you</blockquote>
<code>int main(int argc, char **argv)</code> <br>
rizsa rizsa <mark>lényeg</mark> rizsa
```

Alsó index:  $x_2$ , felső index:  $x^2$   
 "A Kalkulus még könnyű tárgynak számít"

Never gonna give you up  
 Never gonna let you down  
 Never gonna run around  
 And desert you

```
int main(int argc, char **argv)
rizsa rizsa lényeg rizsa
```

## B. Az `iframe` tag

- `<iframe>...</iframe>`: inline frame, más weboldalt lehet megnyitni benne
  - `src`: a megnyitni kívánt weboldal URL-je
  - `width`: iframe szélessége (érdemesebb CSS-ben megadni)
  - `height`: iframe magassága (érdemesebb CSS-ben megadni)

```
<iframe src="http://ttik.hu" width="600" height="600"></iframe>
```

## C. Listák, felsorolások

- `<ul>...</ul>`: rendezetlen (számozatlan) lista
  - `<li>...</li>`: listaelem
- `<ol>...</ol>`: rendezett (számozott) lista
  - `<li>...</li>`: listaelem
  - fontosabb attribútumok:
    - **reversed**: fordított számozás
    - **start**: számozás kezdősorszáma
    - **type**: felsorolási jel típusa
      - lehetséges értékek: 1 (arab szám), i (“kisbetűs” római szám), I (“nagybetűs” római szám), a (kisbetű), A (nagybetű)

```

<!-- Rendezetlen lista -->
<ul>
  <li>Kenyér</li>
  <li>Tej</li>
  <li>Felvágott</li>
</ul>

<!-- Rendezett lista -->
<ol>
  <li>Nyisd ki a szemed!</li>
  <li>Nyomd ki az ébresztőt!</li>
  <li>Aludj vissza!</li>
</ol>
    
```

- Kenyér
- Tej
- Felvágott

1. Nyisd ki a szemed!
2. Nyomd ki az ébresztőt!
3. Aludj vissza!

- a fentiekből készíthetünk többszintű listát is

```

<ol>
  <li>Rendezett listaelem</li>
  <li>
    Rendezett listaelem
    <ul>
      <li>Rendezetlen listaelem</li>
      <li>Rendezetlen listaelem</li>
    </ul>
  </li>
  <li>Rendezett listaelem</li>
</ol>
    
```

1. Rendezett listaelem
2. Rendezett listaelem
  - Rendezetlen listaelem
  - Rendezetlen listaelem
3. Rendezett listaelem

## D. Táblázatok

- `<table>...</table>`: táblázat beszúrása
  - `<tr>...</tr>`: sor
  - `<th>...</th>`: fejléc cella
  - `<td>...</td>`: oszlopcella
    - `colspan`: oszlopok összevonása
    - `rowspan`: sorok összevonása
    - `headers`: cella társítása adott id-jű fejléc cellá(k)hoz
  - `<caption>...</caption>`: táblázat címe
    - közvetlenül a `<table>` tag után írjuk
  - `<thead>...</thead>`: táblázat fejlécsorainak csoportja (elhagyható)
  - `<tbody>...</tbody>`: táblázat adatsorainak csoportja (elhagyható)
  - `<tfoot>...</tfoot>`: táblázat láblécsorainak csoportja (elhagyható)
  - `<colgroup>...</colgroup>`: oszlopok csoportosítása
    - közvetlenül a `<caption>` után (ha van) és a `<thead>` előtt (ha van)
    - a `<col />` páratlan tag egy oszlopot jelöl a csoportosításban
    - a `<col span="n" />` n darab oszlopot jelöl a csoportosításban

```
<table>
  <caption>Valami klassz cím</caption>
  <colgroup>
    <col style="background-color: lightblue;" />      <!-- világoskék -->
    <col style="background-color: lightgray;" span="2" /> <!-- világosszürke -->
  </colgroup>
  <thead>
    <tr>
      <th id="a">Fejléc</th>
      <th id="b" colspan="2">Oszlopösszevonás</th>
    </tr>
  </thead>
  <tbody>
    <tr>
      <th id="c" rowspan="2">Sorösszevonás</th>
      <td headers="b c">adat</td>
      <td>még több adat</td>
    </tr>
    <tr>
      <td>ez is adat</td>
      <td>meg ez is</td>
    </tr>
  </tbody>
</table>
```

Valami klassz cím

Fejléc	Oszlopösszevonás	
Sorösszevonás	adat	még több adat
	ez is adat	meg ez is

## 2. A CSS alapjai

- A CSS (Cascading Style Sheets) egy stílusleíró nyelv
  - a weboldal tartalmának formázására, elrendezésére használjuk
  - a CSS fájlok kiterjesztése: **.css**
  - utasítások szintaxisa: **kijelölő** (amit formázok) + **meghatározásblokk** (ahogyan formázok)
    - a meghatározásblokk tulajdonság-érték párokat tartalmaz
    - például: `h1 { color: blue; font-size: 24px; font-weight: bold; }`
  - kommentek szintaxisa: `/* ez egy komment a CSS kódban */`

### A. CSS kód elhelyezése

- **Beágyazott (inline) CSS:** HTML objektumok **style** attribútumaként adjuk meg
  - például: `<p style="color: red;">Ez egy bekezdés.</p>`
  - az adott objektumra lesz érvényes
  - nem túl hatékony – ha lehet, ne használjuk
- **Lapon belüli (internal) CSS:** a **head**-ben adjuk meg `<style>` és `</style>` tagek között
  - például: `<style> p { color: red; } </style>`
  - az aktuális dokumentumra lesz érvényes
- **Külső (external) CSS:** a **head**-ben adjuk meg a `<link/>` páratlan tag segítségével
  - például: `<link rel="stylesheet" type="text/css" href="foo.css"/>`
  - minden csatoló dokumentumra érvényes lesz a `foo.css` fájlban lévő CSS kód
  - ha több HTML oldalt szeretnénk hasonlóan formázni, ezt érdemes használni

### B. Alapvető kijelölők

- **p:** az összes `<p>...</p>` HTML objektumot kijelöli (element selector)
- **.foo:** az összes `class="foo"` attribútummal rendelkező objektumot kijelöli (class selector)
- **#foo:** az `id="foo"` attribútummal rendelkező objektumot jelöli ki (id selector)
- **\***: minden HTML objektumra vonatkozik (universal selector)

### C. Értékek, mértékegységek

- Hosszúság megadása
  - abszolút hosszúság: mm (milliméter), cm (centiméter), in (inch), pt (pont), pc (pica)
  - relatív hosszúság: % (más értékekből számolódik), px (pixel), em (betűmérethez viszonyít), rem (gyökérelem betűméretéhez viszonyít) vw (ablakszélesség 1%-ához viszonyít), vh (ablakmagasság 1%-ához viszonyít)
- Szögek megadása: deg (fok), rad (radián), grad (gradián)

- Színek megadása: névvel (pl. `white`), decimális RGB-kóddal (pl. `rgb(255, 255, 255)`), hexadecimális RGB-kóddal (pl. `#FFFFFF` vagy röviden `#FFF`)

### D. Objektumok méretezése

- `width`: objektum szélessége
  - megadható: `px`, `%`, `auto` (a böngésző számolja)
  - legkisebb szélesség: `min-width`, legnagyobb szélesség: `max-width`
- `height`: objektum magassága
  - megadható: `px`, `%`, `auto` (a böngésző számolja)
  - legkisebb magasság: `min-height`, legnagyobb magasság: `max-height`

### E. Szövegek formázása

- `font-family`: betűtípus
  - a több szóból álló betűtípusok (pl. `Courier New`) nevét `'`-ok között adjuk meg
  - több betűtípus felsorolható (vesszővel elválasztva) – az első telepített fog megjelenni
- `font-size`: betűméret
- `font-weight`: betűvastagság
  - értékek: `normal` (alapértelmezett vastagság), `bold` (vastag), `bolder` (vastagabb), `lighter` (vékonyabb), szám (100 és 900 között)
- `font-style`: `italic`; – dőlt betűssé tétel
- `color`: betűszín
- `text-align`: szöveg vízszintes igazítása
  - értékek: `left` (balra), `right` (jobbra), `center` (középre), `justify` (sorkizárt)

```
<p>Ez itt egy <span>formázandó</span> szöveg.</p>

<!-- CSS utasítások a head-ben -->
<style>
  p {
    font-family: Arial, Verdana, sans-serif;
    font-size: 18px;
    font-weight: bold;
    color: darkblue;
  }

  span {
    font-family: 'Courier New';
    font-style: italic;
    color: rgb(255, 0, 0);    /* vörös */
  }
</style>
```

Ez itt egy **formázandó** szöveg.

## F. Háttér formázása

- **background-color**: háttérszín
  - megadható szín vagy **transparent** (átlátszó, alapértelmezett érték)
- **background-image**: háttérkép
  - **url('foo.png')**: kép elérési útvonala
- **background-repeat**: háttérkép ismétlődése
  - **repeat** (vízszintes és függőleges ismétlődés, alapértelmezett), **repeat-x** (vízszintes ismétlődés), **repeat-y** (függőleges ismétlődés), **no-repeat** (nem ismétli)
- **background-position**: háttérkép pozíciója (x, y)
  - x: vízszintes pozíció – **left** (bal), **right** (jobb), **center** (középre)
  - y: függőleges pozíció – **top** (fent), **bottom** (lent), **center** (középre)
  - nem kötelező mindkét értéket megadni (pl. **background-position: center;**)
- **background-size**: háttérkép mérete (x, y)
  - x: vízszintes méret, y: függőleges méret
  - nem kötelező mindkét értéket megadni
  - **auto**: automatikus (alapértelmezett)
  - **cover**: kitölti a hátteret az eredeti méretarányok megtartásával
  - **contain**: a legnagyobb méretre skáláz, az eredeti méretarányok megtartásával
- a háttér tulajdonságai összevonva is megadhatók a **background** CSS tulajdonsággal
  - pl. **background: lightblue url('bg.jpg') no-repeat center;**
  - színátmenetes kitöltés: **linear-gradient**(irány, szín<sub>1</sub>, szín<sub>2</sub>);

```
<h1>Üdvözöllek a weboldalamon!</h1>

<!-- CSS utasítások a head-ben -->
<style>
  html, body {
    height: 100%;
  }

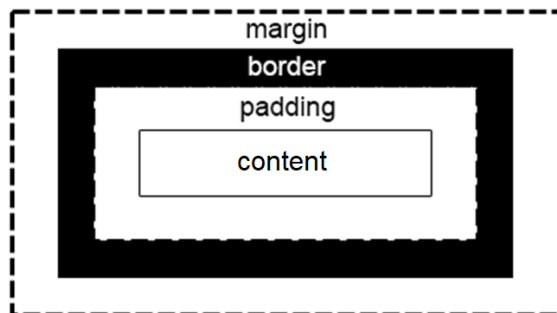
  body {
    background: url('bg.jpg') no-repeat center;
  }

  h1 {
    background-color: darkblue; /* régebbi böngészők */
    background: linear-gradient(90deg, darkblue, blue);
    color: #fff;
    text-align: center;
  }
</style>
```





## G. A dobozmodell



Minden HTML objektum felfogható egy téglalap alakú dobozként. Ezt nevezzük **dobozmodell**nek.

A dobozon belül szerepel a tartalom, ami körül térköz (padding) található. Ezen kívül az objektumok rendelkezhetnek szegéllyel (border) is. A szegély körül található a külső margó (margin).

## a. Térközök megadása

- `padding: [padding-top] [padding-right] [padding-bottom] [padding-left];`
  - a paraméterek rendre: felső, jobb, alsó, bal térköz
  - ezek a paraméterek külön-külön is megadhatók
  - nem megadott paraméter esetén az értéket a szemközti oldaltól örökli

## b. Margók megadása

- `margin: [margin-top] [margin-right] [margin-bottom] [margin-left];`
  - a paraméterek rendre: felső, jobb, alsó, bal margó
  - ezek a paraméterek külön-külön is megadhatók
  - nem megadott paraméter esetén az értéket a szemközti oldaltól örökli

```
/* elem vízszintesen középre igazítása */
div {
    margin-left: auto;
    margin-right: auto;
    text-align: center;
}
```

## c. Szegély megadása

- `border: [border-width] [border-style] [border-color];`
  - a paraméterek külön-külön is megadhatók
  - `border-width`: a szegély vastagsága
    - `px`, `thin` (vékony), `medium` (közepes), `thick` (vastag)
  - `border-style`: a szegély stílusa
    - `solid` (folytonos), `dashed` (szaggatott), `dotted` (pontosított), `double` (dupla), `groove` (faragott), `ridge` (kidomborodó), `none` (nincs), `hidden` (rejtett), ...
  - `border-color`: a szegély színe

```
<div><h2>Gratulálok! Nyertél egy iPhone-t</h2></div>

<!-- CSS utasítások a head-ben -->
<style>
  div {
    border: 5px double red;
    background-color: blue;
    color: yellow;
    text-align: center;
  }
</style>
```

**Gratulálok! Nyertél egy iPhone-t**

**Példa:** Szegély megadása CSS-ben

### H. Elem megjelenítése

- **display:** elem megjelenítése
  - **inline:** sorszintű – vízszintesen ismétlődik, amíg kifér, utána új sorban ugyanígy
  - **block:** blokkszintű – előtte és utána térköz található
  - **inline-block:** inline, de megadhatók a méretei (szélesség, magasság)
  - **none:** elem eltüntetése
  - nagy népszerűségnek örvend a [flexbox](#) és a [grid](#)

```

<p class="magical">Sitty-sutty, eltűntem</p>
<p>Én pedig itt maradtam</p>

<!-- CSS utasítások a head-ben -->
<style>
  img { /* középre igazítás */
    display: block;
    margin: auto;
  }

  .magical { display: none; }
</style>
```



Én pedig itt maradtam