

Egyszerű HTML5 weboldalak készítése (GKxB_INTM049)

Dr. Hatwágner F. Miklós

Széchenyi István Egyetem, Győr

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

2019. október 16.

- Cél: a nyers szöveg egyes részeit strukturálni, jelentésbeli többletet hozzáadni (pl. fejezetcím, bekezdés)
- Történeti előzmény: nyomdai előkészítés, kéziratok szerkesztése, gépi szedőrendszerek
- Példák jelölőnyelvekre: [roff](#), [LaTeX](#), [SGML](#)


```

5  .\" -----
6  .SH SYNOPSIS
7  .\" -----
8  .
9  .SY groff
10 .OP \-abcegi jklpstzCEGNRSUVXZ
11 .OP \-d cs
12 .OP \-D arg
13 .OP \-f fam
14 .OP \-F dir
15 .OP \-I dir

```

◀ ◻ ▶ ◀ ◻ ▶ ◀ ≡ ▶ ◀ ≡ ▶ ≡

- ◀ ◻ ▶ ◀ ◻ ▶ ◀ ≡ ▶ ◀ ≡ ▶ ≡

- Legfontosabb építőelemek
 - Elemek ([element] nyitó és záró címkék [tag] által határolva)
 - A nyitó tagben attribútumok (kulcs-érték párok) adhatók meg
 - Elemek egymásba ágyazhatóak
 - Elemek, attribútumok alkalmazási szabályai → Document Type Definition (DTD)
- Néhány korai, jelentős alkalmazás
 - Electronic Manuscript Project of the Association of American Publishers (AAP, tudományos dokumentumok)
 - Computer-aided Acquisition and Logistic Support (CALS, katonai dokumentumok kezelése)
 - LinuxDoc (Linux csomagok)

SGML példa

```
<!DOCTYPE PEOPLE SYSTEM
  "people.dtd">
<PEOPLE DATE="15 6 2000">
  <NAME TITLE="Mr">
    <FIRST>Wally</FIRST>
    <LAST>Wallpaper</LAST>
  </NAME>
  <NAME>
    <LAST>Jackson</LAST>
  </NAME>
  <NAME TITLE="Dr">
    <FIRST>Susan</FIRST>
    <MIDDLE>Ramsay</MIDDLE>
    <LAST>Sukie</LAST>
  </NAME>
</PEOPLE>
```

people.dtd

```
<!ELEMENT people - - (name+)>
<!ATTLIST people date NUMBERS #REQUIRED>

<!ELEMENT name - - (first?, middle?, last)>
<!ATTLIST name title CDATA #IMPLIED>

<!ELEMENT first - - (#PCDATA)>
<!ELEMENT middle - - (#PCDATA)>
<!ELEMENT last - - (#PCDATA)>
```

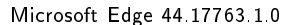
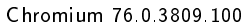
Forrás: [OmniMark dokumentáció](#)

- ◀ ◻ ▶ ◀ ◻ ▶ ◀ ≡ ▶ ◀ ≡ ▶ ≡

- ◀ ◻ ▶ ◀ ◻ ▶ ◀ ≡ ▶ ◀ ≡ ▶ ≡

- Struktúra kialakítása az SGML-hez hasonlóan: egymásba ágyazható elemek, címkék, attribútumok
- Beágyazási szabályok, használható attribútumok → „szabvány” (ajánlás)
- Helytelenül formázott dokumentumok
 - Nincsenek hibaüzenetek
 - A böngésző a tőle telhető legjobb eredményt nyújtja
 - Kompatibilitási okokból az elavult megoldásokat is kénytelen támogatni
 - Ellenőrzés különböző böngészőkben vs. **szintaxis validálás**

```
<h2>Vacak HTML</h1>
<li><a>Ez nem túl</b> jó.
<li><i>Sőt, valójában eléggé <g>silány</g></em>
</ul>
Azért a böngésző mégis megjelenít <a hiv=jajj>valamit.
```



Válasszunk egy szövegszerkesztőt (pl. [Geany](#), [VS Code](#), [Notepad++](#), ...), gépeljük be és mentjük ki az alábbi fájlt `elso.txt` néven, UTF-8 kódolással!

`elso.html`

```
<!DOCTYPE html>
<html lang="hu-HU">
  <head>
    <!-- Ez itt egy megjegyzés -->
    <title>Első weboldalam!</title>
    <meta charset="utf-8" />
  </head>
  <body>
    <h1>Minden kezdet nehéz</h1>
    <p>Még ezt sem volt egyszerű összehozni, pedig olyan rövidke...</p>
  </body>
</html>
```

Dokumentum típusának meghatározása

HTML5 **Nincs DTD!**

```
<!DOCTYPE html>
```

4.01, Szigorú

```
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01//EN"  
"http://www.w3.org/TR/html4/strict.dtd">
```


4.01, Átmeneti

```
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN"  
"http://www.w3.org/TR/html4/loose.dtd">
```

4.01, Keretek

```
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01 Frameset//EN"  
"http://www.w3.org/TR/html4/frameset.dtd">
```

Elemek (element)

- Általában nyitó és záró címkék (tag) között, pl. `<body>...</body>`, `<p>...</p>`
- Néha a böngésző kitalálja, hol kellene lennie az elem (pl. `<p>`, ``) záró címkéjének, így az elhagyható, de **nem javasolt** (XML elemző számára szabálytalanná teszi a fájlt)
- Léteznek üres elemek is; itt nincs mit közbezárni címkékkel, pl. `<meta />`, vízszintes vonal `<hr />` vagy `<hr>`
- Rögzített szabályok szerint egymásba ágyazhatók
- Kis- és nagybetűkre érzéketlen, de **ajánlott** a kisbetűs írásmód
- A szöveg tördelése független a forrásszöveg tördelésétől (pl. az egymás mellé gépelt fehér karaktereket egynek tekinti)
- Címkék mindig `<` és `>` jelek között
- Jelentéssel bíró karakterek bevitele **entitásokkal** (pl. `<` → `<`; vagy `>` → `>`;) 

Elemek

`<html>` gyökérelem, pontosan egynek kell lennie (dokumentum nyelve attribútummal, ld.)

`<head>` metaadatok

`<title>` dokumentum címe (böngészőablak vagy -fül felirata)

`<meta>` általános metaadat

`<body>` megjelenítendő tartalom

`<h1>` „Címsor1”

`<p>` Bekezdés (paragraph)

Megjegyzések

`<!--` és `-->` között

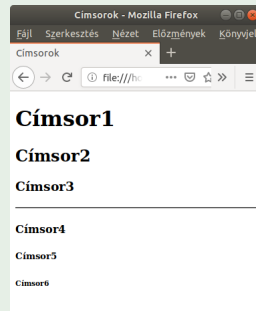
Attribútumok

- Mindig a nyitó címkében (pl. `<html lang="hu-HU">`, [RFC1766](#) szerint)
- Kulcs-érték párok, = jellel elválasztva
- **Ajánlott** a kulcsot kisbetűvel írni
- Az értéket **ajánlott** idézni, lehetőleg " -vel (de az ' is megfelel; szóközt tartalmazó értéknél pedig kötelező)
- Egy címkében lehet több attribútum is
- Vagy egy sem (minimalizált szintaxis); itt az attribútum léte hordoz információt (pl. `<p hidden>`). XML feldolgozók megkövetelik az értéket, pl. `<p hidden="hidden">`

- `<h1>`, `<h2>`, ..., `<h6>`: legmagasabbtól legalacsonyabb szintig
- Például: `<h1>Első fejezet, amelyben bemutatnak bennünket Micimackónak és a méheknek, mellékesen a könyv is elkezdődik</h1>`
- Általában nagyobb betűméretek és a címsor elé és/vagy mögé tett térközök jellemzik
- Keresőmotorok is használhatják a dokumentum struktúrájának feltérképezésére
- Tematikus részek elválasztására gyakran elválasztó vonalat (`<hr />`) használnak

Készítsen weboldalt, ami egymás alá írja a *Címsor1*, *Címsor2*, ..., *Címsor6* szövegeket, a nekik megfelelő HTML elemekkel, a 3. és a 4. szövegsor között pedig húz egy vízszintes vonalat!

cimsorok.html



- Bekezdések `<p>` elemmel jelölhetők
- Például: `<p title="Bek. 1">Első bekezdés.</p><p>Második bekezdés.</p>`
- `title` globális attribútum felugró szövegdoboz, „tooltip” szöveg megadására
- Alapértelmezetten térközt hagy a böngésző a bekezdések között
- Sortörés új bekezdés (és térköz) nélkül: `
`
- Előformázott szöveg (pl. karakteres módú program kimenete), egyenszélességű (monospace) karakterekkel, forrás fehér karaktereinek megőrzésével: `<pre>`

„Tradicionális” **tördelési módok**

Blokkszintű (block)

Új sorban kezdődik, és a teljes elérhető szélességet elhasználja, pl. `<p>`, `<pre>`, `<h1>-<h6>`, `<hr />`

Soron belüli (inline)

Aktuális sorban kezdődik, csak olyan széles helyet foglal, ahol éppen elfér, pl. `
`

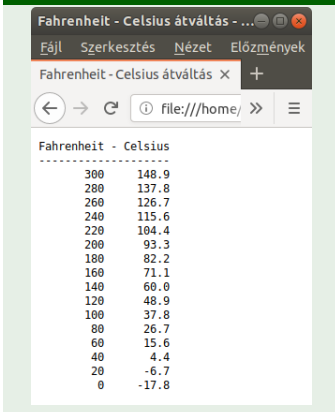
- a költő neve legyen első szintű címsor,
- a mű címe második szintű,
- a szakaszok (arab számokkal jelölve) harmadik szintűek.
- A bekezdéseket és a bekezdésen belüli sortöreseket állítsa be!
- A bekezdések fölé mozgatva az egeret lássuk a bekezdés sorszámát, pl. 1/1, 1/2, ..., 2/3



Formázza meg a klasszikus Fahrenheit-Celsius átváltó **program kimenetét** a mellékelt ábra szerint, azaz őrizze meg a program karakteres kimenetének formázását a `<pre>` elemmel!

A `<pre>` elem megőrzi a HTML forrásban lévő szóközöket, sortöréseket, és monospace betűtípust használ.

fahrcels.html



Fahrenheit - Celsius	
300	148.9
280	137.8
260	126.7
240	115.6
220	104.4
200	93.3
180	82.2
160	71.1
140	60.0
120	48.9
100	37.8
80	26.7
60	15.6
40	4.4
20	-6.7
0	-17.8

A soron belüli formázó elemek közül előnyben részesítjük azok használatát, melyek *szemantikai többletet* adnak a szöveg tartalmának, a pusztán *formázási célú* elemekkel szemben → erre ott a CSS!

Ettől függetlenül, néhány, csak formázási célú elem használata továbbra is szabványos a HTML5-ben.

`<small>`

Kisbetűs szöveg. (A `<big>` elem a HTML5-ben már nem támogatott.)

`<i>` (italics)

Döntötten szedett szöveg, jelentéstöbblet nélkül.

`` (emphasized)

Hangsúlyos, fontos szövegrész, melyet a böngésző alapértelmezetten általában dőlt betűkkel jelenít meg.

`` (bold)

Félkövéren szedett szöveg, jelentéstartalom nélkül.

`` (strong importance)

Kiemelten hangsúlyos, fontos szövegrész, melyet a böngésző alapértelmezetten általában félkövér betűkkel jelenít meg.

`<sup>` (superscript)

Felső index.

`<sub>` (subscript)

Alsó index.

`<ins>` (inserted)

Utólag beszúrt szöveg (ált. aláhúzással jelölve).

`` (deleted)

Kitörölt szöveg (ált. áthúzással jelölve).

`<mark>`

Kijelölt szöveg (ált. sárga háttérrel kiemelve).

Próbálja meg előállítani azt a HTML fájlt, ami a jobb oldalon látható módon jelenik meg a böngészőben! Kiinduláshoz felhasználhatja a dokumentum **nyers szövegét**. Ne feledje, hogy az olyan, önmagukban is jelentéssel bíró karakterek megjelenítése, mint pl. a **<** karakter, **HTML entitásokkal** lehetséges.

soronbelul.html

Sorön belüli formázások - Moz...
Fájl Szerkesztés Nézet Előzmények

Sorön belüli formázások x +

← → ↺ ⓘ file:///home/ » ≡

A `<small>` elemmel kis betűs szöveget lehet írni. Az `<i>` *dőlt betűket* eredményez, de az `` nem csak formáz, hanem a szövegrész **fontosságát** is kiemeli. A `` szintén csak **félkövérre** formáz, de a `` **nagyon fontos** tartalmat sugall.

$$E = mc^2$$

Az **oktató** ma síkhülyenagyon felkészületlen volt.

Szövegrészek idézése. Forrás jelölése: cite attribútummal.

`<q>` (quote)

Rövid szövegrészlet idézése, ált. automatikusan körbeveszi a böngésző idézőjelekkel.
Soron belüli elem.

`<blockquote>`

Hosszú szövegrészek, bekezdések idézése. Jellemzően behúzással formázva.
Blokkszintű elem.

Rövidítések. Kifejtés megadható: title globális attribútummal.

`<abbr>`

Rövidítések és betűszavak jelöléséhez. Soron belüli elem.

`<acronym>`

Betűszavak jelöléséhez használták, **ELAVULT**.

Szöveg írásirányának jelölése: `dir` globális attribútummal, alapértelmezést a nyelvvel együtt a `html` elemben kell állítani. Értékek:

`ltr` (left to right)
balról jobbra

`rtl` (right to left)
jobbról balra

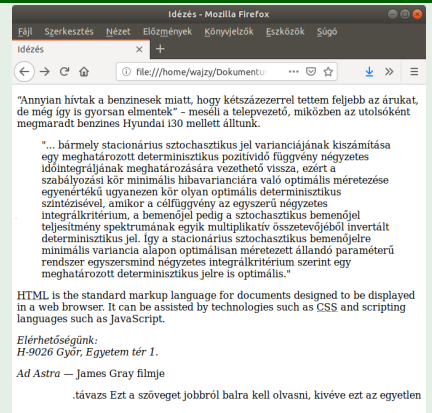
Írásirány helyi módosítása: `<bdo>` (Bi-Directional Override) elemmel, `dir` attribútummal

Mű címének jelölése: `<cite>` soron belüli elemmel.

Postacím megadása: `<address>` blokkszintű elemmel.

Próbálja meg előállítani azt a HTML fájlt, ami a jobb oldalon látható módon jelenik meg a böngészőben! Kiinduláshoz felhasználhatja a dokumentum **nyers szövegét**. Az idézőjelek többféle szélességben elérhetők: `–`, `—`.

idezes.html



Képeket az `` üres, soron belüli elemmel adunk meg. Attribútumok:

`alt` (alternate text)

Kép leírása (gyengénlátók, szöveges böngészők, stb. számára), kötelező.

`src` (source)

Teljes/relatív elérési útvonal, URL

`width`

Szélesség képpontokban

`height`

Magasság képpontokban

Bár a méretet megadó attribútumok nem elavultak, *ajánlott* a méretet CSS-sel megadni (CSS szabályok felülbírálják az attribútumok tartalmát). Az oldalak felesleges újratördelése elkerülhető a méretek megadásával.

Széles körben támogatott formátumok: jpeg, gif, png.

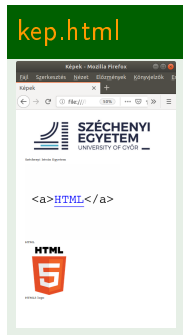
Képek (elemek) beágyazhatók a <figure> blokk szintű elembe.
Célszerű beágyazni a kép feliratát is <figcaption> elembe.

Például:

```
<figure>
  
  <figcaption>Az egyetem logoja</figcaption>
</figure>
```

Készítse el az alábbi oldalt! Képek adatai:

- 1 Forrás: [SZElogo.png](#), méret: 7088x2363 (kicsinyítse 10%-ra!)
- 2 Forrás: [html.gif](#), méret: 500x400
- 3 Forrás:
https://www.w3.org/html/logo/downloads/HTML5_Logo_256.png,
méret: 256x256



Hiperhivatkozások készíthetők `<a>` (anchor) soron belüli elemmel. A nyitó és záró címkék közötti szöveg/kép az érzékeny terület. Attribútumok:

`href` (hypertext reference)

Abszolút/relatív útvonal/URL, az ugrás célja. Könyvtár megadás esetén célszerű `/` jellel zárni.

`target`

Hol nyíljon meg a betöltött tartalom? Értéke lehet:

- `_blank` Új ablakban/fülön
- `_self` Ugyanott, ahol a link is található (alapértelmezés)
- `_parent` Szülő keretben. (A keretek **ELAVULTAK.**)
- `_top` A teljes ablakban, a keretből „kitörve”. (A keretek **ELAVULTAK.**)
- `keretnév` Adott nevű keretben. (A keretek **ELAVULTAK.**)

`title`

Tooltip szöveg

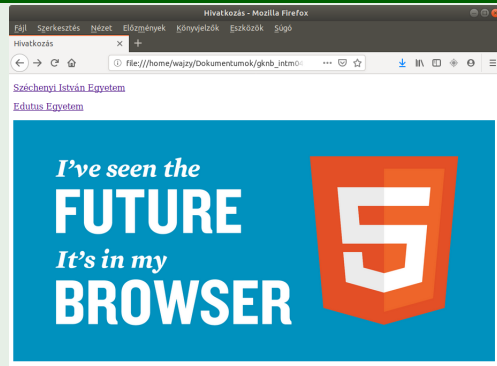
Például: `Ugrás a kereső oldalára`

Az ábrának megfelelően készítsen három hivatkozást egymás alatti bekezdésekben!

- A Széchenyi Egyetem oldalát az aktuális oldal helyére kell betölteni,
- az Edutusét új ablakba/fülre!
- Végül a **HTML5sticker.png** képet használva lehessen eljutni a W3C oldalára!

Készítsen feliratokat (SZE, Edutus, HTML5), melyek megjelennek az egeret a link fölé mozgatva!

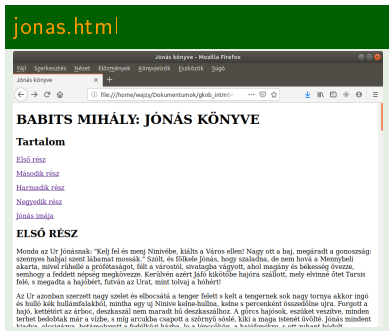
hivatkozas.html



Oldalon belülre mutató hivatkozások is készíthetők, főleg hosszú dokumentumokhoz.

- 1 Egyedi azonosító készítése az ugrás céljához `id` (globális) attribútummal, pl.: `<h1 id="egy">Első fejezet</h1>`
- 2 Hivatkozás (`<a>` elem) készítése, a `href` attribútum értékét `#`-tel kell kezdeni, majd az azonosítóval folytatni, pl. `Ugrás az 1. fejezetre`

A `jonas.txt` fájlból kiindulva hozza létre az ábrán látható oldalt! Az első sor első szintű címsor, a tartalomjegyzék felirat és a vers egyes részei második szintűek. A vers bekezdései bekezdésként jelöltek. A tartalomjegyzék soraira kattintva lehessen elugrani a megfelelő versszakig!



Kép egyes részei kijelölhetők, és más-más oldalakra hivatkozhatnak.

- 1 Megjeleníteni a képet `` elemmel, elhelyezni benne a `usemap` attribútumot, melynek értéke `#`-tel kezdődik, és a *térkép* nevével folytatódik.
- 2 Létrehozni valahol egy `<map>` elemet, melynek `name` attribútuma tartalmazza a térkép nevét.
- 3 Ebbe beágyazni `<area>` üres elemeket, melynek `shape` attribútuma egy alakzatot, `coords` attribútuma pedig ennek koordinátáit definiálja:
 - `rect` téglalap, két átlellenes sarok `X`, `Y` koordinátájával
 - `circle` kör, középpont `X`, `Y` koordinátája, sugár
 - `poly` poligon, egymást követő pontok `x`, `Y` koordinátái; az utolsót összeköti az elsővel
 - `default` a teljes kép külön nem jelölt része
- 4 Az `<area>` `href` attribútuma definiálja a célt.

A [cyclist.jpg](#) felhasználásával hozza létre a [terkepek.html](#) weboldalt! A képen kijelölendő területek alakja, koordinátái és célja az alábbi:

Mit jelöl ki?	Alakzat	Koordináták	Cél
Első kereket	Kör	217, 813, 145	https://ebike.hu/termek/kerek/felni/
Kosarat	Téglalap	102, 420, 256, 643	https://ebike.hu/termek/kiegeszitok/csomagtarto/elore/
Gyerekülést	Poligon	895, 346, 859, 409, 841, 480, 774, 507, 744, 579, 771, 690, 742, 732, 832, 724, 813, 606, 873, 589, 873, 466, 915, 358	https://ebike.hu/termek/kiegeszitok/gyermekules/hatra-vazra/
Maradék részt			https://ebike.hu/