

# Egyszerű HTML5 weboldalak készítése (GKxB\_INTM049)

Dr. Hatwágner F. Miklós

Széchenyi István Egyetem, Győr

[https://github.com/wajzy/GKxB\\_INTM049.git](https://github.com/wajzy/GKxB_INTM049.git)

2019. szeptember 23.

- 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](#)

## RUNOFF

nyers szövegből és parancsokból (.XX) álló fájlok → tördelt megjelenítés buta terminálok (OS: Compatible Time Sharing System, CTSS, 1963)

### runoff

a RUNOFF bővített képességű portja *IBM Selectric* terminálokhoz (OS: Multiplexed Information and Computing Service, multics, ≈'60-as évek vége)

### roff

a runoff továbbfejlesztése a Bell Telephone Labs-nál (1973) a PDP-11 géphez kapcsolt *Graphic Systems CAT* (grafikus szedőegység) miatt. A roff család:

#### troff

typesetter roff a CAT-hez

#### nroff

terminálokhoz és nyomtatókhoz

#### roff

korlátozott képességű runoff utód, nem fejlesztették tovább

### groff

GNU implementáció, máig fejlesztik → man oldalak

```
/usr/share/man/man1/groff.1.gz/groff.1
```

```

1 .TH GROFF 1 "10 February 2018" "Groff Version
   1.22.3"
2 .SH NAME
3 groff \- front-end for the groff document
   formatting system

75 .\" -----
76 .SH SYNOPSIS
77 .\" -----
78 .
79 .SY groff
80 .OP \-abcgijklpstzCEGNRSUVXZ
81 .OP \-d cs
82 .OP \-D arg
83 .OP \-f fam
84 .OP \-F dir
85 .OP \-I dir

```

```

groff@lesov:~$ cat /usr/share/man/man1/groff.1.gz/groff.1
GROFF(1)                                General Commands Manual                                GROFF(1)

NAME
  groff - front-end for the groff document formatting system

SYNOPSIS
  groff [-abcgijklpstzCEGNRSUVXZ] [-d cs] [-D arg] [-f fam] [-F dir] [-I dir] [-K arg] [-L arg] [-M name] [-M dir]
        [-m num] [-o list] [-P arg] [-r sn] [-T dev] [-W name] [-W name] [file ...]
  groff -h | --help
  groff -v | --version [option ...]

DESCRIPTION
  This document describes the groff program, the main front-end for the groff document formatting system. The groff program and macro suite is the implementation of a roff(?) system within the free software collection GNU (http://www.gnu.org). The groff system has all features of the classical roff, but adds many extensions.

  The groff program allows to control the whole groff system by command line options. This is a great simplification in comparison to the classical case (which uses pipes only).

OPTIONS
  The command line is parsed according to the usual GNU convention. The whitespace between a command line option and its argument is optional. Options can be grouped behind a single '-' (minus character). A filename or - (minus character) denotes the standard input.

  As groff is a wrapper program for troff both programs share a set of options. But the groff program has some additional, native options and gives a new meaning to some troff options. On the other hand, not all troff options can be fed into groff.

  Native groff Options
  The following options either do not exist for troff or are differently interpreted by groff.

  -D arg Set default input encoding used by preconv to arg. Implies -k.

  -e      Preprocess with eqn.

  -g      Preprocess with grn.

  -G      Preprocess with grap. Implies -p.

  -h      --help Print a help message.

Manual page groff(1) line 1 (press h for help or q to quit)

```

T<sub>E</sub>X

Betűszedő rendszer,  
fejlesztője Donald E.  
Knuth, 1978  
(Elégedetlenség  
könyvének szedésével.)

L<sup>A</sup>T<sub>E</sub>X

T<sub>E</sub>X-en alapuló  
szövegformázó  
rendszer, Leslie  
Lamport, 1983

## html.tex

```

44 \title{Web technológiák - HTML}[Egyszerű HTML5 weboldalak készítése]
45 \subtitle{[GKx8_INTM049]}
46 \author{Dr. Hatwagner F. Miklós}
47 \institute{Széchenyi István Egyetem, Győr}
48 \date{\hiv{\href{https://github.com/wajzy/GKx8_INTM049.git}{https://github.com/wajzy/GKx8_INTM049.git}}\ \today}
49
50 \begin{document}
51
52 \begin{frame}[plain]
53 \titlepage
54 \end{frame}
55
56 \section{Jelölőnyelvek}
57
58
59 \begin{frame}
60 \begin{itemize}
61
62 \item Cél: a nyers szöveg egyes részeit strukturálni, jelentésbeli többletet hozzáadni (pl. fejezetcím, bekezdés)
63 \item Történeti előzmény: nyomdai előkészítés, kéziratok szerkesztése, gépi szedőrendszerek
64 \item Példák jelölőnyelvekre:
65 \hiv{\href{https://man7.org/linux/man-pages/man7/roff.7.html}{roff}},
66 \hiv{\href{https://www.latex-project.org/}{LaTeX}},
67 \hiv{\href{https://en.wikipedia.org/wiki/Standard_Generalized_Markup_Language}{SGML}}
68 \end{itemize}
69 \end{frame}
70
71 \subsection{roff}
72
73
74 \begin{frame}
75 \footnotesize
76 \begin{description}[n]

```

- SGML (Standard Generalized Markup Language), ISO 8879:1986
- Szabványos jelölőnyelv dokumentumok szerkezetének leírására, beleértve a címkék definiálását is
- Gépfüggetlen metanyelv
- Előzménye: GML (1969)
  - C. Goldfarb (IBM), E. Mosher, R. Lorie
  - dokumentumtípusonként egyedi kódolási séma definiálható
  - előre definiált elemek egymásba ágyazhatóak
  - először az IBM nyomdarendszere használta
- Tulajdonságai
  - Deklaratív: struktúrát és attribútumokat rögzít, nem a feldolgozás módját (→ időtállóság)
  - Gépi feldolgozás lehetősége

- Legfontosabb építőelemek
  - Elemek ([element] nyitó és záró címkek [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](#)



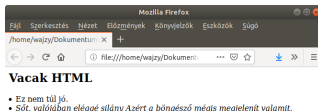
- ENQUIRE: a CERN dokumentumtároló, -megosztó szoftvere. (Tim Berners-Lee, 1980)
- HTML első említése: T.B.L., 1991 (18 elem, melyek a CERN SGMLguid-on, a kutatóintézet SGML alkalmazásán alapultak)
- A HTML egy SGML alkalmazás: 1993-2014
- HTML 4.01: Strict/Transitional/Frameset DTD, 1999
- Aktuális változat: [HTML5](#)
- Néhány újdonság: videó- és hanglejátszás, vektorgrafika, többszálúsítás, helyi adattárolás, bittérképes grafika, stb.
- „Élő szabvány”, meghatározó szervezetek: [W3C](#) (ajánlások), [WHATWG](#) (innovatív technológiák)
- 2019-től a WHATWG tartja karban a HTML szabványát.
- XHTML: XML előírásoknak megfelelő HTML; a HTML5 „feleslegessé” tette

- Egyszerű szövegfájl (jellemzően UTF-8 kódolással)
- Dokumentum *strukturájának* jelölésére, pl.
  - fejlécek
  - listák
  - bekezdések
  - hiperhivatkozások
- Megjelenítést befolyásolja
  - böngésző alapértelmezése
  - felhasználó globális beállításai a böngészőben
  - stíluslapok (CSS)
- Megjelenítés leválasztása
  - helyes megjelenítés többféle böngészőben
  - könnyebben karbantartható oldalak
  - nem vizuális böngészők támogatása

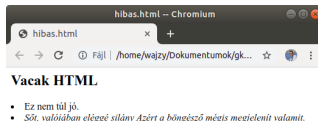
- Struktúra kialakítása az SGML-hez hasonlóan: egymásba ágyazható elemek, címkék, attribútumok
- Beágyazási szabályok, használható attribútumok → „szabvány” (ajánlás)
- Helytelenül formázott dokumentumok
  - Nincsenek hibaüzenetek
  - A böngésző a tőle telhető legjobb eredményt nyújtja
  - Kompatibilitási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

## hibas.html

```
<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.
```



Mozilla Firefox 69.0



Chromium 76.0.3809.100



Microsoft Edge 44.17763.1.0

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>
    <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>`, `<li>`) 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. `<` → `&lt;` vagy `>` → `&gt;`)

## 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)



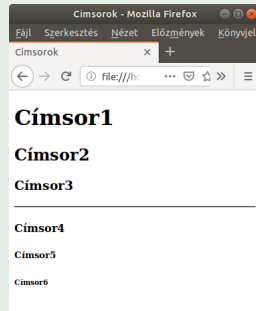
## Attribútumok

- 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
- 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
- `title` attribútum felugró szövegdoboz, „tooltip” szöveg megadására
- Alapértelmezetten térközt hagy a böngésző a bekezdések között
- Sortörés új bekezdés (és térköz) nélkül: `<br />`
- 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. `<br />`

Formázza meg a **verset** a mellékelt ábra szerint, azaz

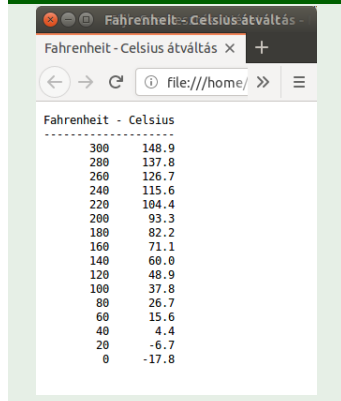
- a költő neve legyen első szintű címsor,
- a mű címe második szintű,
- a szakaszok (arab számokkal jelölve) harmadik szintűek.
- A bekezdéseket és a bekezdésen belüli sortöreseket állítsa be!
- A bekezdések fölé mozgatva az egeret lássuk a bekezdés sorszámát, pl. 1/1, 1/2, ..., 2/3

attila.html



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

## fahrnels.html



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.

`<em>` (emphasized)

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

## <b> (bold)

Félkövéren szedett szöveg, jelentéstöbblet nélkül.

## <strong> (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).

## <del> (deleted)

Kitörölt szöveg (ált. áthúzással jelölve).



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

