

A Web jelölőnyelvei

Jeszenszky Péter

Debreceni Egyetem, Informatikai Kar

jeszenszky.peter@inf.unideb.hu

Utolsó módosítás: 2022. november 9.

A Web jelölőnyelvei

- HTML
- SVG
- MathML

HTML

- *„A HTML a Web elsődleges leíró nyelve.”*
- *„[...] egy szemantikai szintű leíró nyelv és a kapcsolódó szemantikai szintű alkalmazásprogramozási interfészek a Weben elérhető oldalak készítéséhez, melyek a statikus dokumentumoktól a dinamikus alkalmazásokig terjednek.”*
 - Lásd: *HTML Living Standard*
<https://html.spec.whatwg.org/>

HTML verziók használata

- W3Techs: *Usage statistics and market share of HTML for websites*
<https://w3techs.com/technologies/details/ml-html>
 - *„HTML5 is used by 95.9% of all the websites who use HTML”*

HTML 4.01

- ~~HTML 4.01 Specification~~ (W3C ajánlás, 1999. december 24.; hatálytalanítva: 2018. március 27.) <https://www.w3.org/TR/html401/>
 - Az utolsó SGML-alapú HTML verzió.
- Dokumentumtípus-deklarációk:
 - **Strict:**

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

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

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

XHTML (1)

- Az XML alkalmazásként definiált HTML szigorúbb szabályokat ír elő a dokumentumok számára, így azok feldolgozása egyszerűbb.
- Különösen lényeges ez a hagyományos asztali gépekhez képest korlátozott lehetőségekkel bíró eszközökénél (például mobil eszközöknél).
- Az XHTML illetve annak modularizációja lehetővé teszi az XHTML kombinálását más XML alkalmazásokkal.
 - Például MathML és SVG beágyazás XHTML dokumentumokba – ezek a dokumentumok a továbbiakban azonban már nem XHTML dokumentumok.

XHTML (2)

- ~~XHTML™ 1.0 The Extensible HyperText Markup Language (Second Edition) — A Reformulation of HTML 4 in XML 1.0~~ (W3C ajánlás, 2000. január 26.; hatálytalanítva: 2018. március 27.) <https://www.w3.org/TR/xhtml1/>
 - A HTML 4 újrafogalmazása XML alkalmazásként.
- Dokumentumtípus-deklarációk:
 - **Strict:**

```
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Strict//EN"
  "http://www.w3.org/TR/xhtml1/DTD/xhtml1-strict.dtd">
```
 - **Transitional:**

```
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN"
  "http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd">
```
 - **Frameset:**

```
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Frameset//EN"
  "http://www.w3.org/TR/xhtml1/DTD/xhtml1-frameset.dtd">
```
- Média típus: application/xhtml+xml

XHTML (3)

- ~~XHTMLTM Modularization 1.1 – Second Edition~~ (W3C ajánlás, 2010. július 29.; hatálytalanítva: 2018. március 27.) <https://www.w3.org/TR/xhtml-modularization/>
 - A modularizáció lehetővé teszi az XHTML résznyelveinek definiálását és az XHTML kiterjesztését.
 - Megvalósítható a DTD vagy XML Schema felhasználásával is.
 - Szabványos modulok egy készletét biztosítja.
 - Például: Frames (frame, frameset, noframes elemek), Hypertext (a elem), Text (div, h1, p, ... elemek), ...
 - Több modul kombinálása révén úgynevezett hibrid dokumentumtípusok létrehozását teszi lehetővé.

XHTML (4)

- ~~XHTMLTM Basic 1.1 – Second Edition~~ (W3C ajánlás, 2010. november 23.; hatálytalanítva: 2018. március 27.)
<https://www.w3.org/TR/xhtml-basic/>
 - Az XHTML egy olyan részhalmaza, amely számos különböző eszköz számára alkalmas (mobiltelefonok, PDA-k, elektronikus könyvolvasók, tv-készülékek, ...).
 - A dokumentumtípus-definíció megvalósítása modulok segítségével az *XHTML Modularization* ajánlásban foglaltak szerint.
- Dokumentumtípus-deklaráció:
 - `<!DOCTYPE html PUBLIC`
 `"-//W3C//DTD XHTML Basic 1.1//EN"`
 `"http://www.w3.org/TR/xhtml1-basic/xhtml1-basic11.dtd">`
- Média típus: `application/xhtml+xml`

XHTML (5)

- *XHTML*[™] 2.0 (W3C munkacsoport feljegyzés, 2010. december 16.) <https://www.w3.org/TR/xhtml2/>
 - Egy új nyelv, mely nem volt kompatibilis visszafelé a korábbi HTML és XHTML nyelvekkel.
 - A W3C úgy döntött, hogy nem folytatják tovább a fejlesztést, így nem lesz XHTML 2.0 ajánlás.
 - A HTML5 vette át eredetileg kitűzött szerepét.
 - Lásd: *Frequently Asked Questions (FAQ) about the future of XHTML*
<https://www.w3.org/2009/06/xhtml-faq.html>

HTML5

- A HTML legutóbbi verziója.

HTML5: történet

- Lásd:
 - *HTML Living Standard – Introduction – History*
<https://html.spec.whatwg.org/multipage/introduction.html#history-2>

HTML5: fejlesztés (1)

- Eredetileg a WHATWG fejlesztette ki a HTML5 specifikációt.
- A W3C 2007-ben kapcsolódott be a HTML5 fejlesztésébe.
 - Lásd:
 - Ian Hickson. *The WHATWG Blog – W3C restarts HTML effort*. 7 March 2007.
<https://blog.whatwg.org/w3c-restarts-html-effort>
 - *W3C Relaunches HTML Activity*. 7 March 2007.
<https://www.w3.org/2007/03/html-pressrelease>

HTML5: fejlesztés (2)

- 2012 júliusa és 2019 júniusa között a WHATWG és a W3C is külön specifikációt fejlesztett, melynek során eltérő fejlesztési modellt követtek.
 - A W3C a HTML 5.2 specifikáció ajánlásként való kiadása után a következő verzión dolgozott (HTML 5.3).
 - A WHATWG specifikációja soha nem lesz lezárta, folyamatosan fejlesztik („élő szabvány”).

HTML5: fejlesztés (3)

- 2019 júniusáig a W3C-n belül a Web Platform Munkacsoport fejlesztette a HTML nyelvhez kapcsolódó specifikációkat.
- A W3C specifikációi a WHATWG specifikációján alapultak.
 - A W3C bizonyos részeket külön dokumentumokba emelt ki.

HTML5: fejlesztés (4)

- 2019. május 28-án a két szervezet aláírt egy megállapodást arról, hogy együttműködnek a HTML és DOM specifikációk egyetlen verziójának kifejlesztésén.
 - Lásd: Jeff Jaffe. *W3C and WHATWG to work together to advance the open Web platform*. 28 May 2019.
<https://www.w3.org/blog/2019/05/w3c-and-whatwg-to-work-together-to-advance-the-open-web-platform/>
- Együttműködési megállapodás:
 - A HTML-t és a DOM-ot elsősorban a WHATWG fejleszti.
 - A W3C ajánlasként szándékozik jóváhagyni és kiadni a WHATWG specifikációkat.
 - Lásd: *Memorandum of Understanding Between W3C and WHATWG*. May 28, 2019.
<https://www.w3.org/2019/04/WHATWG-W3C-MOU.html>
- A továbbiakban a W3C-n belül a HTML Munkacsoport felelős a HTML fejlesztéséért.
 - Lásd: *HTML Working Group* <https://www.w3.org/groups/wg/htmlwg>

HTML5 szabvány

- **WHATWG:**

- *HTML Living Standard*

- <https://html.spec.whatwg.org/multipage/>

- *HTML: The Living Standard – Edition for Web Developers* <https://html.spec.whatwg.org/dev/>

- Nem tartalmazza a csupán a böngészőgyártók számára szóló információkat.

- **W3C:**

- A <https://www.w3.org/TR/html> URI jelenleg átirányít a WHATWG specifikációra.

HTML elemek (1)

- Az elemeknek, attribútumoknak és attribútumértékeknek meghatározott jelentése (szemantikája) van.
 - Például az `ol` elem egy rendezett listát ábrázol, a `lang` attribútum pedig a tartalom nyelvét ábrázolja.
- A szerzők számára tilos az elemek, attribútumok és attribútumértékek a megfelelő rendeltetésüktől eltérő jelentésbeli céllal történő használata.

HTML elemek (2)

- A HTML előző verziókban rendelkezésre álló prezentációs lehetőségek többsége többé nem megengedett.
- A prezentációs jelölők problémái:
 - A prezentációs elemek használata rontja a hozzáférhetőséget.
 - Magasabb karbantartási költségek.
 - Nagyobb dokumentumméret.

HTML elemek (3)

- Csak a `style` attribútum és a `style` elem maradt meg, mint prezentációs jelölési lehetőség.

HTML elemek (4)

- Média-függetlenként lettek újrafogalmazva az következő, korábban prezentációs elemek:

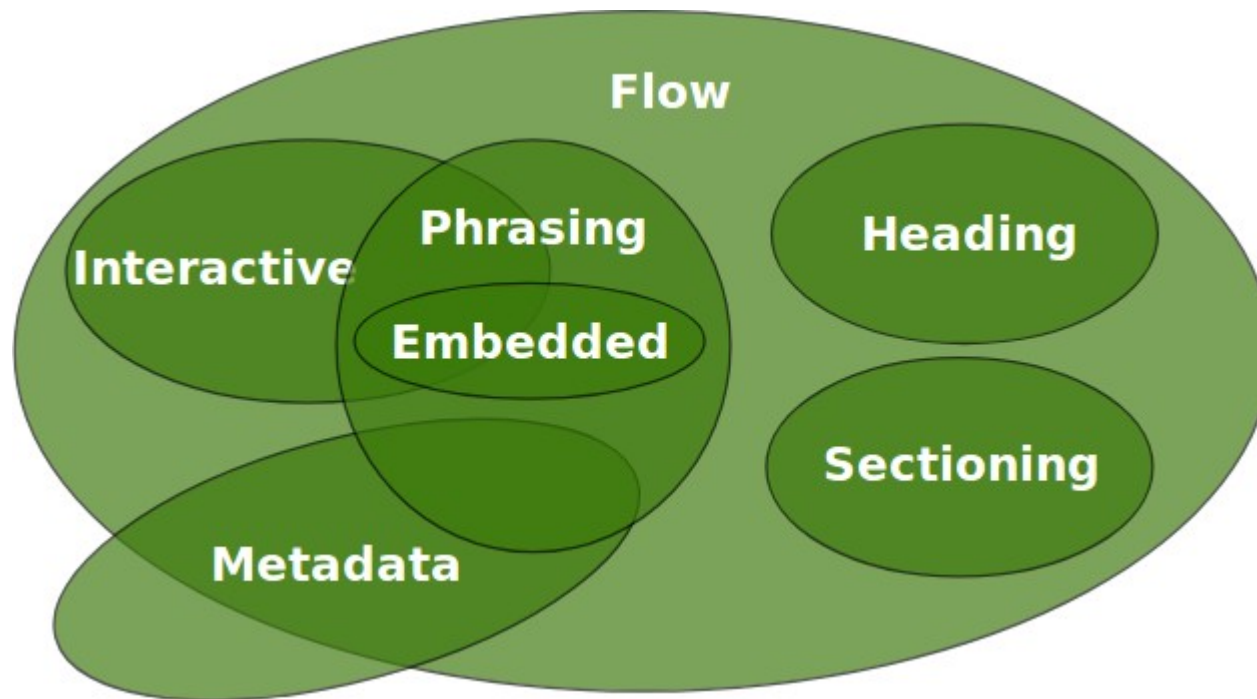
Elem	Leírás	
	HTML 4.01, XHTML 1.0	HTML5
b	Félkövér betű	Kulcsszavak
i	Kurzív betű	Hangnembeli változás
hr	Vízszintes választóvonal	Témaváltás
small	Kisebb betűméret	Lapszéli megjegyzés
s	Áthúzás	Pontatlan szöveg
u	Aláhúzás	Artikulálatlan jelölés (például hibásan írt szöveg)

HTML elemek (5)

- A specifikációban definiált minden egyes elemnek van egy tartalommodellje (az elem szükséges tartalmának egy leírása).
 - Egy HTML elem tartalma meg kell, hogy feleljen a tartalommodelljében leírt követelményeknek.

HTML elemek (6)

- Minden egyes HTML elem nulla vagy több kategóriába sorolható az alábbiak közül:



Forrás: <https://html.spec.whatwg.org/images/content-venn.svg>

HTML elemek (7)

- Idegen elemek:
 - A MathML és SVG névterekbe tartozó elemek.

HTML elemek (8)

- A HTML5-ben a jobb tagoláshoz bevezetett elemek:
 - `article`
 - `aside`
 - `figure`
 - `footer`
 - `header`
 - `nav`
 - `section`

HTML elemek (9)

- A HTML5-ben bevezetett további új elemek:
 - audio
 - canvas
 - dialog
 - meter
 - progress
 - time
 - video
 - ...

HTML elemek (10)

- Az összes HTML elem:
 - *HTML Standard – Index – Elements*
<https://html.spec.whatwg.org/multipage/indices.html#elements-3>
 - *MDN Web Docs – HTML Elements Reference*
<https://developer.mozilla.org/en-US/docs/Web/HTML/Element>

HTML elemek (11)

- Példa:

https://html-basics.surge.sh/html-basics_hu

Szemantikus HTML (1)

- Példa:

- Forrás: *Semantics in HTML*

- https://developer.mozilla.org/en-US/docs/Glossary/Semantics#semantics_in_html

```
<h1>This is a top-level heading</h1>
```

```
<span style="font-size: 2em; margin: 0.67em 0;">  
  This is not a top-level heading  
</span>
```

Szemantikus HTML (2)

- Előnyök:
 - Jó a keresőoptimalizálás (*search engine optimization* – SEO)
 - Javítja a karbantarthatóságot
 - Akadálymentesség
- Lásd: *HTML: A good basis for accessibility*
<https://developer.mozilla.org/en-US/docs/Learn/Accessibility/HTML>

Szemantikus HTML (3)

- Kapcsolódó fogalom: **akadálymentesség** (***web accessibility***)
 - Az akadálymentesség azt jelenti, hogy a webhelyeket, eszközöket és technológiákat úgy tervezik és fejlesztik, hogy azokat fogyatékkal élő embertársaink is használhassák.
 - See:
<https://www.w3.org/standards/webdesign/accessibility>

Globális HTML attribútumok (1)

- Valamennyi HTML elemhez megadható attribútumok:
 - `class`
 - `dir`
 - `id`
 - `lang`
 - `style`
 - `title`
 - `xml:lang` (csak az XML szintaxisban hatásos)
 - Egyéni adat attribútumok
 - ...
 - Lásd: https://developer.mozilla.org/en-US/docs/Web/HTML/Global_attributes

Globális HTML attribútumok (2)

- `class` attribútum:
 - A szerzők az elemek kiterjesztéséhez használhatják a `class` attribútumot, gyakorlatilag saját elemeket létrehozva úgy, hogy közben a legalkalmasabb létező HTML elemet használják.
 - A HTML elemeken megadott `class` attribútum értéke olyan tokenek egy szóközökkel elválasztott listája, melyek azokat a különféle osztályokat ábrázolják, melyekhez az elem tartozik.
 - Példák:

```
<p class="author">...</p>  
<p class="note">...</p>  
<p class="warning">...</p>
```

Globális HTML attribútumok (3)

- Egyéni adat attribútumok:
 - Egy egyéni adat attribútum egy olyan attribútum, melynek neve a `data` - karakterlánccal kezdődik, melyet legalább egy karakter követ a kötőjel után.
 - Olyan egyéni adatok, állapot, annotációk és más hasonlóak az oldal vagy alkalmazás számára történő privát tárolására szolgálnak, melyekhez nincsenek megfelelőbb elemek vagy attribútumok.
 - Minden HTML elemhez tetszőleges számú egyéni adat attribútum adható meg tetszőleges értékkel.

HTML szintaxisok (1)

- A HTML egy dokumentumok leírására szolgáló absztrakt nyelvet határoz meg, az ezt az absztrakt nyelvet használó erőforrások átviteléhez pedig két konkrét szintaxist.

HTML szintaxisok (2)

- **HTML szintaxis:**

- Bár nagyon hasonlít az SGML-hez és az XML-hez, egy külön nyelv saját feldolgozási szabályokkal.
- Kompatibilis a legtöbb ősi böngészővel.
- Fájlkiterjesztés: `.html`, `.htm`
- Média típus: `text/html`

- **XML szintaxis:**

- Az *XML 1.0* és a *Namespaces in XML 1.0* szabványokon alapuló szintaxis.
- Nem határoz meg az további szintaktikai követelményeket az XML-hez előírtakon túl.
- XHTML szintaxisnak is nevezik.
- Fájlkiterjesztés: `.xhtml`, `.xht`
- Média típus: `application/xhtml+xml`

HTML: a HTML szintaxis (1)

- Példa:

```
<!DOCTYPE html>
<html lang="en">
  <head>
    <meta charset="utf-8">
    <title>Sample Page</title>
    <link rel="stylesheet" href="style.css">
  </head>
  <body>
    <p>Hello, World!</p>
  </body>
</html>
```

HTML: a HTML szintaxis (2)

- Kötelező a dokumentumtípus-deklaráció.

HTML: a HTML szintaxis (3)

- Speciális karakterek:
 - Elem szövege nem tartalmazhat ' < ' karaktert vagy félreérthető ' & ' karaktert.
 - Attribútumérték nem tartalmazhat félreérthető ' & ' karaktert.
- Félreérthető ' & ' karakter:
 - Egy olyan ' & ' karakter, melyet egy vagy több ASCII alfanumerikus karakter és egy ' ; ' karakter követ, melyek nem felelnek meg a szabvány által definiált nevesített karakterhivatkozások egyikének sem (például &nosuchchar ;).
 - Lásd: *Named character references*
<https://html.spec.whatwg.org/multipage/named-characters.html#named-character-references>

HTML: a HTML szintaxis (4)

- Az elem- és attribútumnevek kisbetű-nagybetű érzéketlenek.
 - Az elemek és attribútumok neveinek megadásakor (még az idegen elemeknél is) tetszőlegesen keverhetők a kis- és nagybetűk.
 - Ugyanabban a nyitó címkében soha nem fordulhat elő két vagy több olyan attribútum, melyek nevei kisbetű-nagybetű érzéketlen hasonlítás esetén megegyeznek.

HTML: a HTML szintaxis (5)

- Nem idézett attribútumérték szintaxis:
 - Ha egy, az üres karakterlánctól különböző attribútumérték nem tartalmaz egyetlen literális *whitespace* karaktert sem, akkor megadható az attribútumérték-határolók elhagyásával.
 - Ekvivalensek például az alábbiak:

```
<input value="yes">  
<input value=yes>
```

HTML: a HTML szintaxis (6)

- Logikai attribútumok:
 - Sok attribútum logikai.
 - Egy logikai attribútum jelenléte egy elemen az igaz értéket ábrázolja, hiánya pedig a hamis értéket.
 - Ha megjelenik az attribútum, akkor értéke az üres karakterlánc kell, hogy legyen, vagy egy olyan érték, mely kisbetű-nagybetű érzéketlen hasonlítás esetén megegyezik az attribútum nevével.
 - Ekvivalensek például az alábbiak:

```
<input type=checkbox checked name=agree disabled>  
<input type=checkbox checked=checked name=agree disabled=disabled>  
<input type='checkbox' checked=' ' name="agree" disabled="">
```

HTML: a HTML szintaxis (7)

- Üres (*void*) elemek:
 - Csak nyitó címkéjük van, tilos hozzájuk záró címke megadása.
 - Például: `br`, `img`, `input`, `link`, `meta`, ...

HTML: a HTML szintaxis (8)

- Az idegen elemeknek vagy egy nyitó és egy záró címkéjük van, vagy egy önlezáróként jelölt nyitó címkéjük, mely esetben nem lehet záró címkéjük.
 - Ekvivalensek például az alábbi SVG elemek:

```
<circle cx="50" cy="50" r="50"></circle>  
<circle cx="50" cy="50" r="50"/>
```

HTML: a HTML szintaxis (9)

- Opcionális címkék:
 - Bizonyos elemek nyitó és záró címkéi elhagyhatók.
 - Egy elem nyitó címkéjének elhagyása az itt tárgyalt esetekben nem azt jelenti, hogy az elem nem nincs ott (feltételezetten, de ott van)!
 - Például egy HTML dokumentumnak mindig van egy gyökéreleme, még akkor is, ha a `<html>` karakterlánc egyáltalán nem jelenik meg benne.
 - Lásd: *Optional tags*
<https://html.spec.whatwg.org/#optional-tags>

HTML: a HTML szintaxis (10)

- Opcionális címkék: (folytatás)
 - Elhagyható egy `li` elem záró címkéje, ha az elemet közvetlenül egy másik `li` elem követi vagy nincs több tartalom a szülő elemében.
 - Ekvivalensek például az alábbiak:

```
<ul>  
  <li>Apple</li>  
  <li>Banana</li>  
  <li>Cherry</li>  
</ul>
```

```
<ul>  
  <li>Apple  
  <li>Banana  
  <li>Cherry  
</ul>
```

HTML: a HTML szintaxis (11)

- Opcionális címkék: (folytatás)
 - Elhagyható a `html` elem nyitó címkéje, ha elsőként nem egy megjegyzést tartalmaz.
 - Elhagyható a `html` elem záró címkéje, ha nem egy megjegyzés követi közvetlenül.
 - ...

HTML: a HTML szintaxis (12)

- Opcionális címkék: (folytatás)
 - Ha nem lényegesek az elemek közötti whitespace karakterek, ekvivalensek például az alábbiak:

```
<!DOCTYPE html>
<html>
  <head>
    <title>Sample Page</title>
  </head>
  <body>
    <p>Hello, World!</p>
  </body>
</html>
```

```
<!DOCTYPE html>
<title>Sample Page</title>
<p>Hello, World!</p>
```


HTML: a HTML szintaxis (13)

- Nem támogatottak a névtér-deklarációk, még idegen elemekhez sem.
- CDATA-szakaszok csak idegen tartalomban (MathML vagy SVG) használhatók.

HTML: az XML szintaxis

- Példa:

```
<?xml version="1.0" encoding="UTF-8"?>
<!DOCTYPE html>
<html xmlns="http://www.w3.org/1999/xhtml" xml:lang="en">
  <head>
    <title>Sample Page</title>
    <link rel="stylesheet" href="style.css"/>
  </head>
  <body>
    <p>Hello, World!</p>
  </body>
</html>
```

Nincs HTML DTD

- DTD-k és XML sémák nem képesek kifejezni a HTML által támasztott valamennyi megfelelési követelményt.
 - Lásd például az egyéni data-* attribútumokat.
 - *Embedding custom non-visible data with the data-* attributes*
<https://html.spec.whatwg.org/multipage/dom.html#custom-data-attribute>

HTML dokumentumtípus-deklaráció

- A HTML szintaxisban a `<!DOCTYPE html>` dokumentumtípus-deklaráció szükséges, melynek célja mindössze annak biztosítása, hogy a dokumentum megjelenítése a szabványos módban történjen.
 - A fenti rövid dokumentumtípus-deklaráció előállítására nem képes szoftverek használhatják helyette a `<!DOCTYPE html SYSTEM "about:legacy-compat">` dokumentumtípus-deklarációt.
 - Lásd: *The DOCTYPE* <https://html.spec.whatwg.org/multipage/syntax.html#the-doctype>
- Az XML szintaxisban tetszőleges dokumentumtípus-deklaráció használható, megadása nem is kötelező.
 - Az `application/xhtml+xml` média típussal továbbított dokumentumok megjelenítése mindig a szabványos módban történik.
 - Lásd: *Writing documents in the XML syntax* <https://html.spec.whatwg.org/multipage/xhtml.html#writing-xhtml-documents>
- Lásd még:
 - <https://www.w3.org/TR/html5-diff/#doctype>

DOM (1)

- Egy DOM fa egy dokumentum memóriabeli ábrázolása.
- Vonatkozó szabvány:
 - *DOM Living Standard* <https://dom.spec.whatwg.org/>

DOM (2)

- A DOM egy alkalmazásprogramozási interfész (API) dokumentumok (főleg HTML és XML dokumentumok) eléréséhez és manipulálásához.
 - A DOM a **Dokumentum Objektum Modell** (***Document Object Model***) jelenti.
- Minden ilyen dokumentumot egy fa ábrázol, mely az alábbi fajta csomópontokból áll:
 - Document, DocumentType, DocumentFragment, Element, Text, ProcessingInstruction és Comment.

DOM (3)

- Példa:

```
<!DOCTYPE html>
<html lang="en">
  <head>
    <title>Sample Page</title>
  </head>
  <body>
    <p>Hello, World!</p>
    <!-- This is a comment -->
  </body>
</html>
```

```
{ DOCTYPE: html
  { html lang="en"
    { head
      { #text: 
      { title
        { #text: Sample Page
      { #text: 
    { #text: 
    { body
      { #text: 
      { p
        { #text: Hello, World!
      { #text: 
      { #comment: This is a comment
      { #text: 
```

DOM (4)

- Minden egyes csomópontot egy API-val rendelkező objektum ábrázol, így tehát manipulálható.
- A DOM interfészek Web IDL-ben kerülnek leírásra.

DOM (5)

- A Web IDL egy interfészleíró nyelv, mely böngészőkben implementálható interfészek leírására szolgál.
 - Az aktuális szabvány:
 - *Web IDL Living Standard* <https://webidl.spec.whatwg.org/>
 - Példa a használatra:
 - Node interfész
<https://dom.spec.whatwg.org/#interface-node>

DOM (6)

- A HTML specifikáció a HTML elemek ábrázolásához a DOM interfészeket kiterjesztő további interfészeket határoz meg.
 - Példa:
 - meta elem
<https://html.spec.whatwg.org/multipage/semantics.html#the-meta-element>

DOM (7)

- A DOM nem csupán egy API, a HTML implementációk megfelelési kritériumai is DOM műveletekkel vannak meghatározva.
 - Példa: *The Navigator object*
<https://html.spec.whatwg.org/multipage/system-state.html#the-navigator-object>
- A specifikációk jórészt a DOM fák segítségével vannak megfogalmazva, nem pedig a jelölőkével.

DOM (8)

- Egy DOM fa szkriptekből manipulálható az oldalon.
 - Példa:

```
<!DOCTYPE html>
<html lang="en">
  <head>
    <title>DOM Example</title>
  </head>
  <body>
    <p>User agent: <span id="ua"></span></p>
    <script>
      document.getElementById("ua").innerHTML =
        navigator.userAgent;
    </script>
  </body>
</html>
```

DOM (9)

- Eszközök:
 - *Live DOM Viewer*
<https://software.hixie.ch/utilities/js/live-dom-viewer/>

HTML: DOM, HTML és XML szintaxis

- A DOM, a HTML szintaxis és az XML szintaxis közül nem mind képes ugyanazt a tartalmat ábrázolni.
 - Lásd például:

	HTML	XML	DOM
Névterek	nem	igen	igen
noscript	igen	nem	nem
- -> karakterlánc megjegyzésben	nem	nem	igen

HTML API-k

- Lásd:
 - *MDN Web Docs – Web APIs*
<https://developer.mozilla.org/en-US/docs/Web/API>

Böngésző támogatás

- A modern böngészőmotorok támogatják.
 - Lásd:
 - *The HTML5 test – How well does your browser support HTML5?* (Niels Leenheer) <https://html5test.com/>
<https://github.com/WebPlatformTest/HTML5test>
 - *Can I use... Support tables for HTML5, CSS3, etc.*
<https://caniuse.com/>

Böngésző fejlesztőeszközök

- **Chromium, Google Chrome, Opera:**
 - *Chrome DevTools* <https://developer.chrome.com/docs/devtools/>
- **Firefox:**
 - *Firefox DevTools User Docs*
<https://firefox-source-docs.mozilla.org/devtools-user/>
- **Safari:**
 - *Safari Developer Tools* <https://developer.apple.com/safari/tools/>
<https://support.apple.com/guide/safari-developer>
- **Chromium-based Edge:**
 - *Microsoft Edge DevTools documentation*
<https://docs.microsoft.com/en-us/microsoft-edge/devtools-guide-chromium/landing/>

Böngészők fejlesztői kiadásai

- **Firefox:**

- *Firefox Browser Developer Edition*

- <https://www.mozilla.org/en-US/firefox/developer/>

HTML szerkesztők (1)

- Szabad és nyílt forrású szoftverek:
 - *Visual Studio Code* (platform: Linux, macOS, Windows; licenc: MIT License)
<https://code.visualstudio.com/>
<https://github.com/Microsoft/vscode>
 - Lásd: <https://code.visualstudio.com/docs/languages/html>
 - Ajánlott kiterjesztések:
 - *Live Preview*
<https://marketplace.visualstudio.com/items?itemName=ms-vscode.live-server>
<https://github.com/microsoft/vscode-livepreview>

HTML szerkesztők (2)

- Emmet (programozási nyelv: JavaScript; licenc: *MIT License*)
<https://emmet.io/> <https://github.com/emmetio/emmet>
 - Szövegszerkesztő bővítmények HTML és CSS kód írásának gyorsításához.
 - Számos szövegszerkesztőhöz rendelkezésre áll, mint például: Atom, Brackets, Notepad++, Visual Studio Code, IntelliJ IDEA, ...
 - Lásd: <https://emmet.io/download/>
 - Dokumentáció: <https://docs.emmet.io/>
 - Testreszabás: <https://docs.emmet.io/customization/>
 - Lásd még:
 - *Emmet in Visual Studio Code*
<https://code.visualstudio.com/docs/editor/emmet>

HTML szerkesztők (3)

- Példák Emmet rövidítésekre:

```
ul>li*3
```

kifejtése:

```
<ul>
  <li></li>
  <li></li>
  <li></li>
</ul>
```

```
section.chapter>h1{Introduction}+p
```

kifejtése:

```
<section class="chapter">
  <h1>Introduction</h1>
  <p></p>
</section>
```

HTML szerkesztők (4)

- Példák Emmet rövidítésekre: (folytatás)

```
ul>li*5>a[href=#chapter$]{Chapter $}
```

kifejtése:

```
<ul>
  <li><a href="#chapter1">Chapter 1</a></li>
  <li><a href="#chapter2">Chapter 2</a></li>
  <li><a href="#chapter3">Chapter 3</a></li>
  <li><a href="#chapter4">Chapter 4</a></li>
  <li><a href="#chapter5">Chapter 5</a></li>
</ul>
```

HTML szerkesztők (5)

- Példák Emmet rövidítésekre: (folytatás)

Rövidítés	Kifejtés
@i	@import url();
bd	border: 1px solid #000;
bdt	border-top: 1px solid #000;
bgc	background-color: #fff;
ffm	font-family: monospace;
p10	padding: 10px;
p1e	padding: 1em;
p1.5	padding: 1.5em;
pt10	padding-top: 10px;
pt1e	padding-top: 1em;
tac	text-align: center;
tar	text-align: right;

HTML: Egyéb szoftverek (1)

- Szabad és nyílt forrású szoftverek:
 - *HTML5 Boilerplate* (platform: böngésző; licenc: *MIT License*)
<https://html5boilerplate.com/> <https://github.com/h5bp/html5-boilerplate>
 - *http-server* (platform: Node.js; licenc: *MIT License*)
<https://www.npmjs.com/package/http-server>
<https://github.com/http-party/http-server>
 - *Nu Html Checker* (platform: Java; licenc: *MIT License*)
<https://validator.github.io/validator/>
<https://github.com/validator/validator>
 - Webes interfész: <https://validator.w3.org/nul/>
 - *Tidy* (platform: Linux, macOS, Windows; licenc: *Tidy License*)
<https://www.html-tidy.org/> <https://github.com/htacg/tidy-html5>

HTML: Egyéb szoftverek (2)

- *http-server: a simple static HTTP server*
 - Használat:

```
$ npx http-server -o index.html
```

```
$ npm install -g http-server
```

```
$ http-server -o index.html
```

HTML: online szolgáltatások

- Surge: ingyenes statikus HTML publikálás a parancssorból
 - Webhely: <https://surge.sh/>
 - Dokumentáció: <https://surge.sh/help/>
 - Telepítés és használat:

```
$ npm install --global surge  
$ surge --domain my-domain.surge.sh
```

SVG (1)

- Nyelv két-dimenziós vektorgrafika XML-ben történő leírásához.
 - Az interaktív grafikát és animációt is támogatja.
- Fejlesztő: *W3C SVG Working Group*
<https://www.w3.org/Graphics/SVG/>

SVG (2)

- Az aktuális szabvány:
 - *Scalable Vector Graphics (SVG) 1.1 (Second Edition)* (W3C ajánlás, 2011. augusztus 16.) <https://www.w3.org/TR/SVG11/>
- A szabvány következő verziója:
 - *Scalable Vector Graphics (SVG) 2* (előzetes W3C javaslat, 2018. október 4.) <https://www.w3.org/TR/SVG2/>
- Az SVG egy profilja mobil eszközökre:
 - *Scalable Vector Graphics (SVG) Tiny 1.2 Specification* (W3C ajánlás, 2008. december 22.) <https://www.w3.org/TR/SVGTiny12/>
 - Az MMS (*Multimedia Messaging Service*) vektorgrafikus formátuma.

SVG (3)

- SVG tartalom beágyazható más dokumentumokba.
 - A HTML támogatja a közvetlen beágyazást (lásd az `svg` elemet).
- Séma (DTD):
<https://www.w3.org/Graphics/SVG/1.1/DTD/svg11.dtd>
- Fájlkiterjesztés: `.svg`
- Média típus: `image/svg+xml`

SVG (4)

- Böngésző támogatás:
 - Az összes modern böngésző natív módon támogatja.
 - Lásd: <https://caniuse.com/svg>

SVG (5)

- Szabad és nyílt forrású szoftverek:
 - Szerkesztők:
 - *Inkscape* (platform: Linux, macOS, Windows; licenc: GPLv2) <https://inkscape.org/>
<https://gitlab.com/inkscape/inkscape>
 - *macSVG* (platform: macOS; licenc: *MIT License*)
<https://macsvg.org/> <https://github.com/dsward2/macSVG>
 - *SVG-Edit* (programozási nyelv: JavaScript; licenc: *MIT License*)<https://github.com/SVG-Edit/svgedit>

SVG (6)

- Szabad és nyílt forrású szoftverek:
 - Könyvtárak:
 - *Apache Batik* (programozási nyelv: Java; licenc: *Apache License 2.0*)
<https://xmlgraphics.apache.org/batik/>
 - *Frappe Charts* (programozási nyelv: JavaScript; licenc: *MIT License*)
<https://frappe.io/charts> <https://github.com/frappe/charts>
 - *Rough.js* (programozási nyelv: TypeScript; licenc: *MIT License*)
<https://roughjs.com/> <https://github.com/rough-stuff/rough>
 - *Snap.svg* (programozási nyelv: JavaScript; licenc: *Apache License 2.0*)
<http://snapsvg.io/> <https://github.com/adobe-webplatform/Snap.svg/>
 - *SVG.js* (programozási nyelv: JavaScript; licenc: *MIT License*)
<https://svgjs.dev/> <https://github.com/svgdotjs/svg.js>

SVG (7)

- Mintapéldák:
 - *Bootstrap Icons* (licenc: *MIT License*)
<https://icons.getbootstrap.com/> <https://github.com/twbs/icons/>
 - *Feather – Simply beautiful open source icons* (licenc: *MIT License*) <https://feathericons.com/>
<https://github.com/feathericons/feather>
 - *Inkscape Gallery* <https://inkscape.org/gallery/>
 - *Super Tiny Icons* (licenc: *MIT License*)
<https://github.com/edent/SuperTinyIcons>
 - *Tabler Icons* (licenc: *MIT License*) <https://tabler-icons.io/>
<https://github.com/tabler/tabler-icons>

SVG (8)

- További hasznos címek:
 - *MDN Web Docs – SVG*
<https://developer.mozilla.org/en-US/docs/Web/SVG>
 - *SVG Tutorial*
<https://developer.mozilla.org/en-US/docs/Web/SVG/Tutorial>

MathML (1)

- Nyelv matematikai jelölések XML-ben történő leírásához.
 - Célja lehetővé tenni a Weben matematika „felszolgálatát”, fogadását és feldolgozását, miként a HTML lehetővé tette ezt a funkcionalitást szöveghez.
 - Számos célra felhasználható.
 - Például matematikai tartalom megjelenítése a Weben, matematikai formulák exportálása komputer algebra rendszerekből, ...
- Fejlesztő: *W3C Math Working Group*
<https://www.w3.org/Math/>

MathML (2)

- Az aktuális szabvány:
 - *Mathematical Markup Language (MathML) Version 3.0 2nd Edition* (W3C ajánlás, 2014. április 10.)
<https://www.w3.org/TR/MathML/>
 - ISO szabványként is ratifikálták:
 - *ISO/IEC 40314:2016: Information technology – Mathematical Markup Language (MathML) Version 3.0 2nd Edition*
<https://www.iso.org/standard/58439.html>
- A szabvány következő, jelenleg fejlesztés alatt álló verziója a MathML 4: <https://w3c.github.io/mathml/>

MathML (3)

- Leírható segítségével matematikai kifejezések megjelenítése és jelentése is.
 - Külön elemkészletek szolgálnak a két eltérő célra (prezentációs elemek, tartalmi elemek).
- MathML tartalom beágyazható más dokumentumokba.
 - A HTML támogatja a közvetlen beágyazást (lásd a `math` elemet).

MathML (4)

- Séma (RELAX NG):
 - <https://www.w3.org/Math/RelaxNG/mathml3/mathml3.rng>
 - <http://www.w3.org/Math/RelaxNG/mathml3/mathml3.rnc>
- Fájlkiterjesztés: .mml
- Média típus: application/mathml+xml,
application/mathml-
presentation+xml,
application/mathml-content+xml

MathML (5)

- Böngésző támogatás:
 - **Blink** (Chromium, Google Chrome, Opera, Chromium-alpú Edge): nem támogatott (fejlesztés alatt)
 - <https://www.chromestatus.com/features/5240822173794304>
 - <https://developer.microsoft.com/en-us/microsoft-edge/status/mathml/?q=mathml>
 - **Gecko** (Firefox): támogatott
 - Lásd: <https://platform-status.mozilla.org/#mathml>
 - **WebKit** (Safari): támogatott
 - <https://trac.webkit.org/wiki/MathML>
 - <https://webkit.org/blog/6803/improvements-in-mathml-rendering/>
 - <https://webkit.org/demos/mathml/MathMLDemo.xhtml>
- Lásd: *Can I use... Support tables for HTML5, CSS3, etc*
<https://caniuse.com/mathml>

MathML (6)

- Sok további alkalmazás támogatja, például a Maple, Wolfram Mathematica, LibreOffice (MathML importálás), ...
 - Ezek egy gyűjteményét lásd itt:
https://www.w3.org/wiki/Math_Tools

MathML (7)

- Szabad és nyílt forrású szoftverek:
 - *MathJax* (programozási nyelv: JavaScript; licenc: *Apache License 2.0*) <https://www.mathjax.org/>
<https://github.com/mathjax/MathJax>
 - JavaScript-ben írt megjelenítő motor.

MathML (8)

- Mintapéldák:
 - <https://developer.mozilla.org/en-US/docs/Web/MathML/Examples>
 - *W3C MathML Test Suite*
<https://www.w3.org/Math/testsuite/>

MathML (9)

- További hasznos címek:
 - *MDN Web Docs – MathML*
<https://developer.mozilla.org/en-US/docs/Web/MathML>
 - *Planet MathML* <https://www.w3.org/Math/planet/>
 - *MathML in Web Browsers* <https://mathml.igalia.com/>

Böngészőmotorok megjelenítési módjai (1)

- A böngészőmotorok az alábbi módokban jelenítenek meg HTML dokumentumokat:
 - Kompatibilitási mód (*quirks mode*)
 - Szabványos mód (*standards mode*)
 - Majdnem szabványos mód (*almost standards mode*)

Böngészőmotorok megjelenítési módjai (2)

- A megjelenítési módok létezésének történeti oka, hogy a korai webszabványok nem voltak kompatibilisek az akkoriban létező böngészők viselkedésével.
 - A böngészők egy új megjelenítési módot vezettek be, hogy megfeleljenek a webszabványoknak, miközben továbbra is képesek a létező tartalmak helyes megjelenítésére.
 - Így eltérő módon történik az aktuális szabványoknak megfelelő modern és a régmódi weboldalak megjelenítése.

Böngészőmotorok megjelenítési módjai (3)

- **Kompatibilitási mód (*quirks mode*):** régi böngészők viselkedésének utánzása (emulálása) az aktuális webszabványokat sértő módon régi weboldalak megjelenítéséhez.
 - *Quirk*: jelentése „furcsa viselkedés”.
- **Szabványos mód (*standards mode*):** a weboldalak megjelenítése az aktuális webszabványoknak megfelelően.
- **Majdnem szabványos mód (*almost standards mode*):** néhány böngészőmotor rendelkezik egy harmadik üzemmóddal is, mely bizonyos magasságok meghatározásában tér el a szabványos módtól, mely például képek táblázatcellákban való elhelyezését érinti.

Böngészőmotorok megjelenítési módjai (4)

- Korábban a böngészőmotorok némileg eltérő kompatibilitási módokat használtak, azonban a WHATWG az alábbi specifikációban szabványosította a kompatibilitási módot:
 - *Quirks Mode – Living Standard*
<https://quirks.spec.whatwg.org/>
 - A specifikáció nem sorol fel a szabványos viselkedéstől való minden eltérést, melyek jelenleg a böngészőkben léteznek, számos ilyen más WHATWG specifikációk részleteznek.

Böngészőmotorok megjelenítési módjai (5)

- Az alábbi WHATWG specifikáció átnevezi a szabványos és a majdnem szabványos módot:
 - *DOM – Living Standard*
<https://dom.spec.whatwg.org/>
 - *Standards mode* helyett *no-quirks mode*
 - *Almost standards mode* helyett *limited-quirks mode*

Böngészőmotorok megjelenítési módjai (6)

- Megjelenítési mód választása egy HTML dokumentumhoz:
 - A `text/html` média típussal továbbított dokumentumokhoz a dokumentumtípus-deklaráció határozza meg a megjelenítési módot.
 - Az `application/xhtml+xml` média típussal továbbított dokumentumok megjelenítése mindig a szabványos módban történik.
 - Ilyenkor a dokumentum elemzése egy olyan XML elemzővel történik, mely a dokumentum jóformáltságát is ellenőrzi.

Böngészőmotorok megjelenítési módjai (7)

- A használt megjelenítési mód meghatározása:
 - A Document DOM interfész `compatMode` attribútuma a "BackCompat" karakterláncot adja vissza, ha a dokumentum megjelenítési módja a kompatibilitási mód, egyébként pedig a "CSS1Compat" karakterláncot.
 - Lásd: *DOM – Living Standard – Interface Document*
<https://dom.spec.whatwg.org/#dom-document-compatmode>

Böngészőmotorok megjelenítési módjai (8)

- További hasznos címek:
 - *MDN Web Docs – Quirks Mode and Standards Mode*
https://developer.mozilla.org/en-US/docs/Web/HTML/Quirks_Mode_and_Standards_Mode
 - Henri Sivonen, *Activating Browser Modes with Doctype*. <https://hsivonen.fi/doctype/>