



szit

&lt; HTML

# HTML5 nyelv

- **Szerző:** Sallai András
- Copyright © Sallai András, 2009-2021
- Licenc: GNU Free Documentation License 1.3
- Web: <https://szit.hu> [<https://szit.hu>]

## Bevezetés

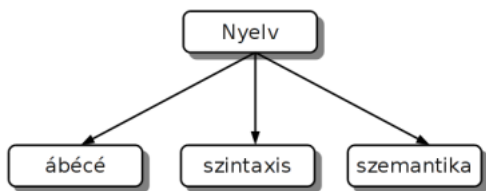
A HTML, a **HyperText Makeup Language** szavakból alkotott betűszó. Jelentése Hiperszöveget leíró nyelv. Tulajdonképpen weboldalak készítésére lett létrehozva. A HTML5-ig hosszú „kanyargós út vezetett. A HTML, oldalak megjelenítése sosem volt egy egzakt, minden részében pontosan meghatározott művelet. Ez természetesen a böngészőkön múlik. Azokon belül is mindig a legnépszerűbb böngészők alakítják a szabványokat, amelyeket más böngészőgyártók, kisebb-nagyobb sikerrel szintén megvalósítanak. A böngészők mindig más képességekkel rendelkeztek, holott mindig ugyanazt a weboldalt kellett megjeleníteni. Ha egy weboldal például Firefoxra lett optimalizálva, elkerülhetetlenül lesz aki a weboldalt Chrome-mal, Operával vagy más böngészővel nézi majd meg. A böngészők a nem ismert dokumentum elemeket egyszerűen figyelmen kívül hagyják, de mellette az ismerteket megjelenítik, saját tudásuk szerint. A webmestereknek így nehéz dolguk van, amikor azt szeretnék elérni, hogy minden böngészőben egységes legyen a weboldal megjelenése.

A jelenlegi legújabb HTML verzió a HTML5. A böngészők a HTML5 nyelvet nem ismerik százszázalékosan, de folyamatosan megvalósítják az újabb verziókban. A böngészőnk HTML5 képességét ellenőrizzük itt: <http://html5test.com> [<http://html5test.com>]

## A nyelvekről

### Nyelvek szerkezete

Mivel nyelvről beszélünk, ezért nézzük meg hogyan épül fel egy nyelv. Egy bármilyen nyelv szerkezetileg három részből áll: ábécé, szintaktika, szemantika.



Az ábécé a nyelv jelkészlete. A HTML esetén ez azok a karakterek, amelyekkel elkészíthetünk egy weboldalt.

A szintaktika a helyesírási szabályok. A HTML esetén a weblap leírásának szabályai.

A szemantika a nyelv jelentéstana. Ha szintaktikailag helyes mondatokat hozunk létre, de a jelentés értelmetlen, vagy nem az amit vártunk, akkor szemantikai hibáról beszélünk. A HTML esetén, nem azt a kinézetet, vagy tartalmat kapjuk amit vártunk, de szintaktikailag helyes.

### Nyelvek osztályozása

A nyelveket feloszthatjuk természetes és mesterséges nyelvekre.

Természetes	Mesterséges	
	Kommunikációra szánt	Formális
magyar	Esperanto	C
angol	Eurolatin	C++
német	Egyszerű angol	Python
francia	Gestuno	Perl
olasz	Volapük	<u>HTML</u>
spanyol	Intal	XHTML
görög	Ido	HTML5

A formális nyelvek osztályozása

Formális nyelvek				
lekérdező	dokumentumleíró	programozási	matematikai	logikai
SQL	<u>HTML</u>	C		
xBase [ <a href="https://en.wikipedia.org/wiki/xBase">https://en.wikipedia.org/wiki/xBase</a> ]	XHTML	C++		
HPCC ECL [ <a href="https://en.wikipedia.org/wiki/ECL_(data-centric_programming_language)">https://en.wikipedia.org/wiki/ECL_(data-centric_programming_language)</a> ]	SGML	Pascal		
	XML	Java		

## Eredet (SGML, XML)

A HTML5 nyelvet az XML alapján alkották meg, amely egy általános célú dokumentum leíró nyelv. A HTML5 „laza” szabályzata miatt ugyanakkor nem teljesíti százszázalékosan az XML követelményeit.

Az XML az SGML (Standard Generalized Markup Language) alapján készült.

Az SGML nyelven egy dokumentumot több részre tagolunk általunk meghatározott elemekkel. Ezen elemek neveit a "<„ és ">„ jelek között helyezzük el, hogy elkülönüljön a dokumentum tartalmától. Az elemek neveit mi magunk találhatjuk ki. Például tárolni szeretnénk egy gépjármű adatait akkor az egész dokumentum tartalmát közrefoghatjuk egy gépjármű nevű elemmel:

```
<gepjarmu>

</gepjarmu>
```

Egy gépjárműről szeretnénk tárolni a rendszámát, úrtartalmát és színét.

```
<gepjarmu>
  <rendszam>ASG-843</rendszam>
  <urtartalom>1800</urtartalom>
  <szin>piros</szin>
</gepjarmu>
```

A dokumentumot a gépjármű elem kezdi és zárja, ezért ezeket gyökér elemeknek nevezzük. Vegyük észre, hogy a záró elem nevét mindig ’/’, karakter előzi meg.

XML

```
<?xml version="1.0"?>
<gepjarmu>
  <rendszam>ASG-843</rendszam>
  <urtartalom>1800</urtartalom>
  <szin>piros</szin>
</gepjarmu>
```

Gyakorlat 01

- 1. Mi az SGML?
- 2. Mi az XML?
- 3. Hogyan osztályozzuk a nyelveket?
- 4. Hogyan épül fel egy nyelv?
- 5. Milyen dokumentumleíró nyelv alapján készült a HTML5?
- 6. Készítsen egy SGML dokumentumot, amely dolgozók adatait tárolja.

HTML szabályok (helyesírás)

DTD

Úgynevezett DTD-ékben definiáljuk az SGML és az XML alapján alkalmazni kívánt szintaxist.

A DTD a Document Type Definition szavakból alkotott betűszó, magyarul Dokumentum Típus Definíció néven fordítható.

Egy DTD egyetlen sora általánosan így írható le:

```
<!DOCTYPE root-elem [elem-deklarációk]>
```

A felkiáltó jel és root elem után megadhatunk egyéb deklarációt is, de az nem kötelező.

DTD példák

Járművekről általános szöveges leírást szeretnénk tárolni. Az eltárolandó szöveg például:

A járművek a közlekedés eszközei.

A DTD így nézhet ki:

```
jarmu.dtd:

<!DOCTYPE jarmu [
<!ELEMENT jarmu (#PCDATA)>
]>
```

A példában egyetlen elemet deklaráltunk. Ez a gyökérelem, ami bármilyen szöveget tartalmazhat.

A dokumentum ekkor így nézhet ki:

```
<!DOCTYPE jarmu SYSTEM "jarmu.dtd">
<jarmu>
  A járművek a közlekedés eszközei.
</jarmu>
```

Járművekről szeretnénk információt tárolni egy dokumentumban.

A PCDATA a „parsed character data” szavakból alkotott betűszó. Az SGML vagy XML parszer átvizsgálja az ilyen adatot, és markup-okat és entity-ket keres benne. Ha ilyet talál, azokat visszakódolja (kifejti). Ilyen kód lehet az &lt; vagy &g;

A CDATA a „character data” szavakból alkotott betűszó. Ilyen tartalomban az XML parszer nem végez entitáskeresést, ha van benne egy &lt; akkor azt négy önálló karakterként értelmezi. A CDATA tartalmazta a <![CDATA[ nyitó és ]]> bezáró jelek közé tesszük.

XML Schema

Az XML Schema-át a W3C a DTD utódként definiálta. Dokumentációja és használata igen bonyolult. A DTD pedig nem teljesen fedi le az XML lehetőségeit.

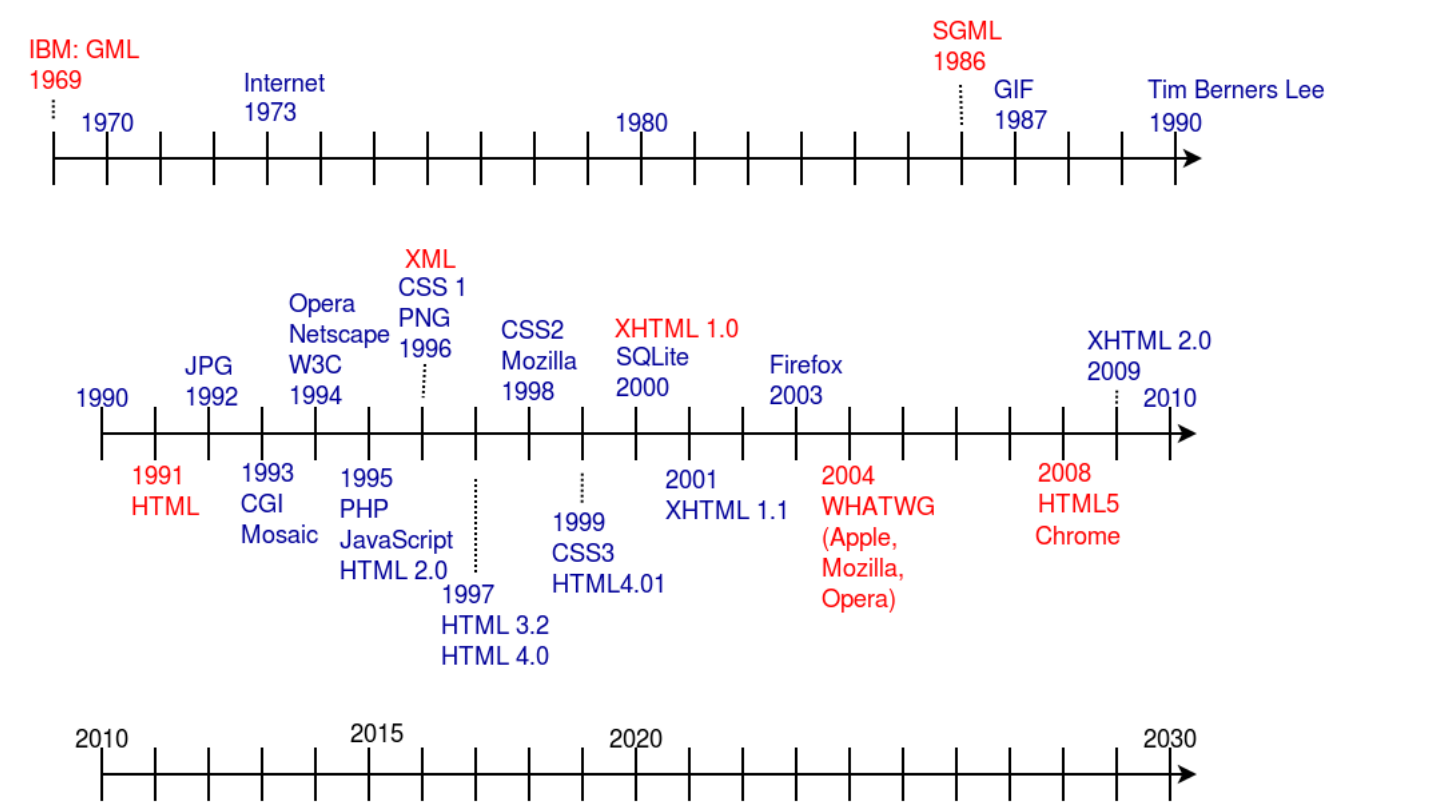
A HTML nyelvről

Az Internet az **ARPANET**-el kezdődött **1969**-ben. Ekkor és még ez után több mint 20 évig webről szó sem volt. Ezek szerint az Internet a korai időszakban kb. 20 évig weboldalak nélkül létezett. Meg kell jegyezzük, az Internet ma sem egyenlő a Webbel (nem úgy, mint ahogy azt sokan gondolják). Az Interneten sok más szolgáltatás is van, de kétségtelen, hogy legnépszerűbb a web, ha nem is a legfontosabb.

A HTML, a HyperText Makeup Language szavakból alkotott betűszó. Jelentése **Hiperszöveget** leíró nyelv.

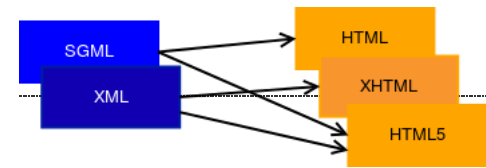
A HTML-t **Tim Berners Lee** találta ki 1990-ben, majd **1991**-ben közzé tette. Eredeti célja szerint a HTML, csak az oldal szerkezetét határozta volna meg, és egy másik leírónyelv kell meghatározza az oldal kinézetét, stílusát. Az SGML általános célú dokumentumleíró nyelvet választotta a nyelv alapjául.

A HTML nem kis és nagybetűérzékeny. A kezdetekben a HTML elemeket nagybetűvel írtuk, hogy azok jól elkülönüljenek a tartalomtól. Idővel a HTML nyelv átalakítása XML alapokra szülte az XHTML nyelvet. Mivel ez XML-en alapszik az elemeket csak kisbetűvel lehetett írni.



Most a HTML5 nyelvben a kis és nagybetű újra lényegtelen. A HTML5 elemeket azonban ajánlott mégis kisbetűvel írni, mert a böngésző úgyis átalakítja.

**Tim Berners Lee** egy szövetséget hozott létre a webes szabványok kidolgozására, amelynek neve W3C lett. Az első HTML verzió után további verziók jelentek meg.



HTML verziók	
első	1991
2.0	1995
3.2	1997
4.0	1997
4.01	1999

XHTML verziók	
1.0	2000
1.1	2001
2.0	2009

HTML5	
első	2008

A HTML5-öt nem a W3C konzorcium hozza létre. A böngészőgyártók nem voltak hajlandóak igazodni a szigorú XHTML szabványhoz. Úgy döntöttek meg kell jeleníteni a szemetet is. 2007-ben a **Mozilla Foundation**, **Apple**, és **Opera Software** létrehozta egy munkacsoportot **WHATWG** néven, azaz **Web Hypertext Application Technology Working Group**. A csoport létrehozta a HTML5 nyelvet. A W3C konzorciummal való együttműködést azonban nem zárták ki, maga a konzorcium kénytelen hozzájuk igazodni, mivel az alternatív böngészők egyre nagyobb részesedést szereznek a böngészőpiacon. A W3C konzorciumra vár végül a feladat, hogy kidolgozza a HTML5 szabványt, miközben a böngésző gyártók megvalósítják azt.

A böngészők HTML5 képességét pontozással mérjük 1 – 555-ös skálán, amelyen 2013 áprilisában a következő pontszámok érvényesek:

2013 április (ekkor a max. pont: 500)		
Böngésző	Pont	Bónusz
Chrome 26	468	13
Opera 12.10	419	9
Firefox 20	394	10
Safari 6.0	378	8

2013 április (ekkor a max. pont: 500)		
Böngésző	Pont	Bónusz
Internet Explorer 10	320	6

- <http://html5test.com/> [<http://html5test.com/>]

A web talán a leggyorsabban változó technológia. Amióta van – a gyors változásoknak köszönhetően – mindig más és más volt a népszerű a webmesterek között. Az első időszakban ha valakinek volt saját honlapja, az menő volt. Később képekkel tarkították a hiperszöveget. Ez után jött a CGI, amely által a weboldalak interaktívak lettek.

A következő ábrán megpróbáltam összeszedni a teljesség igénye nélkül a fontosabb mérföldköveket. Az ábra a teljesség igénye nélkül készült és szubjektív ítéleteken alapszik.



## Gyakorlat 02

- Milyen nyelvből ered a HTML nyelv?
- Mi az SGML?
- Mi a HTML?
- Mi a DTD?
- Ki találta ki a HTML nyelvet?
- Ki készítette el az XML nyelvet?
- Szerkezetileg hogyan épül fel egy nyelv?
- Sorolja fel a HTML fontosabb verzióit.
- A HTML5 verziót kik találták ki?
- Mi a W3C konzorcium?
- Mondjon egy magyar intézményt, amely tagja a W3C konzorciumnak.
- Mi az ARPANET?
- Mikortól létezik az ARPANET?
- Mikor alkották meg az első weboldalt?
- Az XHTML-nek melyik az utolsó verziója?
- Mikor definiálták az XML 1.0 verziót?

## Helló Világ

### Kezdés

A következő példa egy végtelenül egyszerű weblapot jelenít meg, amelynek van egy cím és egy bekezdése.

index.html

```
<!DOCTYPE html>
<html lang="hu">
<head>
<title>Cím</title>
<meta charset="utf-8">
</head>
<body>

<h1>Weblap címe</h1>
<p>Egy bekezdés</p>

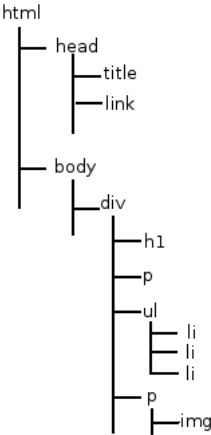
</body>
</html>
```

A HTML oldalt egy doctype sorral kezdük, majd a html elemek jönnek. A html elemek közé egy head és egy body elem jön. A head részbe a weblaphoz hozzáadott járulékos információkat adjuk meg, mint az oldal címe ami a böngészőfüleken jelenik meg a karakter kódolása, stb. A body elemek között adjuk meg magát a weboldalt, azt aminek a böngészőben meg kell jelennie.

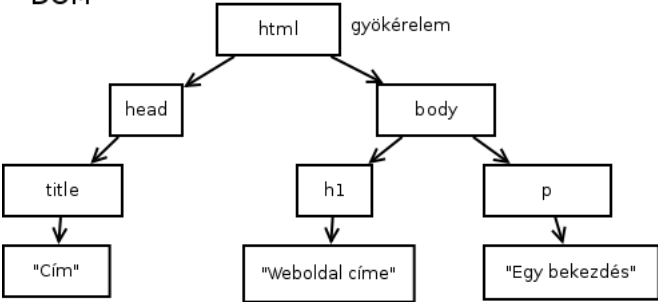
### DOM

A HTML oldal leírható egy dokumentum objektum modellel (DOM):

Dokumentumfa



DOM



Dokumentum típus definíció

Egy HTML oldal első sora mindig a dokumentum típusát és verzióját leíró sor. Ezek mindig egy ”!DOCTYPE„ karaktersorozattal kezdődnek.

Elsőként lássuk a HTML5 előtt használatos DOCTYPE sorokat:

```
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01//EN" "http://www.w3.org/TR/html4/strict.dtd">
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN" "http://www.w3.org/TR/html4/loose.dtd">
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01 Frameset//EN" "http://www.w3.org/TR/html4/frameset.dtd">
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Strict//EN" "http://www.w3.org/TR/xhtml1/DTD/xhtml1-strict.dtd">
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.1//EN" "http://www.w3.org/TR/xhtml11/DTD/xhtml11.dtd">
```

A HTML5 definíciója:

```
<!DOCTYPE html>
```

Dokumentum

A HTML dokumentum két részre osztható egy fejrészre és egy törzsre.

Fejrész
Törzs

A fejrész a „head” elemek között van. A törzs pedig a „body” elemek között. A fej és a törzsrészt a „html” gyökérelem fogja közre:

```
<html>
<head>
Fejrész
</head>
<body>
Törzs
</body>
</html>
```

A gyökérelem számára megadhatunk egy tulajdonságot, amely az oldal nyelvét fogja tartalmazni:

```
<html lang="hu">
```

Ez a „lang” tulajdonság.

Értéke a nemzetköz nyelvi szabványban megadott egyik nyelv lehet (iso-639).

Fejrész

A fejrész a „head” eleme között helyezkedik el és tartalmazhat újabb elemeket. A fejrész tartalma nem jelenik meg a böngészőben, viszont hasznos információkat tartalmaznak a weboldalról.

Kötelező elem a „title”. A „title” elemmel mondjuk meg a böngésző számára, mit írjon ki a címsorban, vagy ha fülével dolgozunk a böngészőfülön.

Nem kötelező elem a „meta”. Ilyen elemet egyet vagy többet is tartalmazhat egy weboldal. Mindenképpen ajánlott megadni a karakterkódolást.

Gyakorlat 03

1. Mire való a DOCTYPE sor?
2. Mik a gyökérelemek?
3. Milyen részekre osztható egy HTML forrás?
4. Mi a szerepe a title elemnek?
5. Írjon példát egy HTML5 előtti DOCTYPE sorra.

Meta elemek

A meta szót a magyarban összetett szavakban szoktuk használni és valaminek a módosulását jelöli.

A HTML-be is hasonló a szerepe. A weboldalnak vannak olyan tulajdonságai amik közvetlenül nem jelennek meg a weblapon, de hatással vannak arra, módosítják. Ezeket a tulajdonságokat a meta elemmel adjuk meg. A meta elemet mindig a head elemek között adjuk meg. Több dologra is használjuk, ezért a meta elemnél attribútumokkal adjuk meg éppen mire szeretnénk használni.

A meta elemek a „meta” névvel kezdődnek:

```
<meta ... >
<meta name=“érték” content=“érték” />
```

Általában két tulajdonságot adunk meg, egy „name” vagy egy „http-equiv” és egy „content” tulajdonságot. Ez alól kivétel lehet a karakterkódolás, amelyet a HTML5 megenged rövidebb formában.

```
<meta charset="utf-8" />
```

Átírányítás ennyi idő múlva:

```
<meta http-equiv="Refresh" content="0; URL="masikweblap.html">
```

Még több meta elem:

```
<meta name="generator" content="Bluefish">
<meta name="author" content="Sallai András">
<meta name="date" content="2009-01-01">
<meta name="copyright" content="© 2009, Sallai András">
<meta name="keywords" content="web, html, css">
<meta name="description" content="Weboldal készítése">
<meta name="ROBOTS" content="NOINDEX, NOFOLLOW">
<meta http-equiv="content-type" content="text/html; charset=UTF-8">
<meta http-equiv="expires" content="0">
```

- A weboldalt készítő program
- A weboldal írója, szerzője – Lehet bő felsorolás is.
- A készítés dátuma
- Terjesztési feltételek
- Kulcsszavak
- Leírás
- A keresőrobotok indexeljék-e a weboldalt
- Weboldal kódolása
- A weblap lejár

Még több meta tag:

```
<meta name="contact" content="nagyontitkospostafiok@gmail.com" />
<meta http-equiv="last-modified" content="YYYY-MM-DD@hh:mm:ss TMZ" />
<meta http-equiv="cache-control" content="no-cache" />
<meta http-equiv="content-language" content="hu,en" />
<meta http-equiv="pragma" content="no-cache" />
```

- A weboldal tulajdonosának elérhetősége (A spammerek botjai figyelik!)
- Utolsó módosítás
- Az oldal legyen tárolva a böngésző tárolójában (cache).
- A content-language a tartalma a [ISO 639-1 \[https://en.wikipedia.org/wiki/List\\_of\\_ISO\\_639-1\\_codes\]](https://en.wikipedia.org/wiki/List_of_ISO_639-1_codes) alapján meghatározott két betű.
- A pragma csak egy másik lehetőség a weblap tárolásának megakadályozására.

Keresőrobotok

Fentebb már említettük a robots kulcsot, nem néztük meg milyen értékeket vehet fel.

```
<meta name="robots" content="NOINDEX, NOFOLLOW">
```

- All
- Index, follow - ugyanaz, mint az all
- NoIndex
- NoFollow
- NoIndex, NoFollow

Megjegyzés

```
<!-- megjegyzés -->
```

A megjegyzéseket a böngészők figyelmen kívül hagyják, így hasznos megjegyzéseket helyezhetünk el magunk számára a forráskódban.

A megjegyzéske tesztelés esetén is jól jönnek. A kód működő, vagy a teszt szempontjából lényegtelen kódokat megjegyzésbe téve tesztelhetjük a weblapunkat.

A megjegyzések nem egymásba ágyazhatók, és nem tartalmazhatnak ”--„ és ”->„ részeket.

Fejezetcímek meghatározása

Fejezet címeket h1, h2, h3, h4, h5 és h6-os elemekkel határozzuk meg.

A példa kedvéért a tigrisről szeretnénk weblapot csinálni és szeretnénk írni az elterjedéséről, az alfajairól, a megjelenéséről, az életmódjáról és a szaporodásáról. Ekkor a következő módon határozzuk meg fejezetcímeknek:

```
<h2>Elterjedése</h2>

<h2>Alfajok</h2>

<h2>Megjelenése</h2>

<h2>Életmód</h2>
```

A különböző méretek

```
<h1>H1 fejezetcím</h1>
<h2>H2 fejezetcím</h2>
<h3>H3 fejezetcím</h3>
<h4>H4 fejezetcím</h4>
<h5>H5 fejezetcím</h5>
<h6>H6 fejezetcím</h6>
```

H1 fejezetcím

H2 fejezetcím

H3 fejezetcím

H4 fejezetcím

H5 fejezetcím

H6 fejezetcím

Bekezdések

A bekezdés

A bekezdéseket a p elemek közé tesszük.

```
<p>
Vinum regum rex vinorum. Vinum regum rex vinorum.
Vinum regum rex vinorum. Vinum regum rex vinorum.
Vinum regum rex vinorum. Vinum regum rex vinorum.
Vinum regum rex vinorum. Vinum regum rex vinorum.
Vinum regum rex vinorum. Vinum regum rex vinorum.
Vinum regum rex vinorum. Vinum regum rex vinorum.
Vinum regum rex vinorum. Vinum regum rex vinorum.
Vinum regum rex vinorum. Vinum regum rex vinorum.
Vinum regum rex vinorum.
</p>
```

Szerkeszthető bekezdés

A tartalmat szerkeszthetővé tehetem a contenteditable tulajdonsággal:

```
<p contenteditable="true">
Lõrem ipsum
</p>
```

A contenteditable bármely elemmel használható, nem csak a p elemmel.

A következő szöveg szerkeszthető:

Lorem ipsum

Blokkszintű és inline elemek

A blokk elemek a böngésző egyik szélétől a böngésző másik széléig tartanak. Ha blokkszintű elemből kettőt helyezünk el a kódban, a weboldalon a második elem tartalma már újabb sorban jelenik meg.

Az inline elemek egy sorban (soron belül) jelennek meg, ha az kifer.

Blokkszintű elemek

- div
- p
- h1
- h2
- h3
- h4
- h5
- h6
- stb.

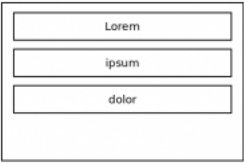
Inline szintű

- span
- b
- i
- a
- stb.

Jelöljük a „Lorem”, „ipsum” és „dolor” szavakat, először blokk elemmel. Például p elem:

```
<p>Lorem</p>
<p>ipsum</p>
<p>dolor</p>
```

Az ábrán látható, hogy egy blokk elem önmagában, kitölti a rendelkezésre álló helyet vízszintesen. Hogy ez látható legyen, a dobozt alkotó blokk elemet egy szegéllyel ábrázolok.



Most jelöljük ugyanezeket a szavakat inline elemmel:

```
<span>Lorem</span>
<span>ipsum</span>
<span>dolor</span>
```

Az ábrán láthatjuk, hogy az inline elemek, a tartalomhoz igazodnak, ha vízszintesen elférnek, akkor egymás mellett jelennek meg.



A CSS segítségével blokk elemekből inline elemet, az inline elemekből blokk elemet készíthetünk a „display” tulajdonsággal. A két típus között CSS segítségével, állítható inline-blokk elem is. Egy inline-blokk elem nem töltik ki vízszintesen a rendelkezésre álló helyet, ahogy az inline elem sem, viszont állítható a szélessége.

Sortörés, elválasztás

Sortörés:

```
<br />
```

Nem törhető szóköz:

```
&nbsp;
```

Feltételes elválasztójel ( soft hyphen):

```
&shy;
```

Ha egy szóban elhelyezzük és az ablak mérete megkívánja a sorok tördelését, a sortörés az elválasztó jel mentén is megtörténhet.

Az a har&shy;cos aki csak az ö&shy;lés&shy;hez ért na&shy;gyon se&shy;bez&shy;het&shy;ő. Az i&shy;gá&shy;zán nagy sza&shy;mu&shy;rá&shy;jok na&shy;gyon szé ha&shy;i&shy;ku&shy;kat í&shy;rtak.

A teszteléshez méretezzük kisebbre az ablakot:

Az a harcos aki csak az öléshez ért nagyon sebezhető. Az igazán nagy samurájok nagyon szép haikukat írtak.

Szöveg kiemelése

HTML kiemelő tagek

Tag	Leírás
<b>	félkövér szöveg
<em>	kiemelt szöveg
<i>	dőlt szöveg
<small>	kis szöveg
<strong>	erős szöveg
<sub>	alsó index
<sup>	felső index
<ins>	beszúrt szöveg
<del>	törölt szöveg
<s>	elavult, pontatlan, nem fontos szöveg

**félkövér** szöveg

*kiemelt* szöveg

*dőlt* szöveg

kis szöveg

**erős** szöveg

alsó index és <sup>felső</sup> index

beszúrt szöveg

~~törölt~~ szöveg

HTML Számítógép kimeneti tagek



Tag	Leírás
<code>	számítógépes kód
<kbd>	billentyűzet
<samp>	mintakód
<var>	változó
<pre>	előreformázott szöveg

Kódszöveg: `printf("alma\n");`

Billentyűkombináció: `Ctrl + D`

Minta: `^a.*`

Másik minta:

Ha mentés során hiba keletkezett akkor kiírjuk a `<samp>Hiba a mentés során!</samp>` szöveget a képernyőre.

Ha mentés során hiba keletkezett akkor kiírjuk a `Hiba a mentés során!` szöveget a képernyőre.

Változó: `$PATH`

Előreformázott szöveg:

```
printf("Behúzással\n");
```

A következő példa egy szerkeszthető pre elemet mutat be:

```
<pre contenteditable="true">
#include <stdio>
main()
{
    printf("Kezdés\n");
}
</pre>
```

```
#include <stdio>
main()
{
    printf("Kezdés\n");
}
```

HTML idézet és definíciós tagek

Tag	Leírás
<abbr>	egy rövidítés
<address>	cím egy dokumentum tulajdonosának elérhetősége
<bdo>	szöveg iránya
<blockquote>	hosszú idézet
<q>	rövid idézet
<cite>	cím idézése
<dfn>	egy kifejezés meghatározása

A [HTML 4.01](#) verzióban létezett még a `<acronym>` elem, amely mozaikszavak megadására lett kitalálva, de a HTML5 már nem támogatja.

Rövidítések:

```
<abbr>stb.</abbr>
<abbr>u.</abbr>
<abbr>dec.</abbr>
<abbr title="február">feb.</abbr>
```

stb. u. dec. feb.

Nagy József  
06(30) 123-4567  
nagy.jozsef@valahol.tld

A szöveg irányának megváltoztatása:

Normál szöveg

```
<bdo dir="ltr">Valami szöveg</bdo>
```

Fordított irány:

```
<bdo dir="rtl">Valami szöveg</bdo>
```

Normál szöveg:

Valami szöveg

A fordított szöveg megjelenítve:

gevözs imalaV

Gyakorlat 04

- Milyen elemmel határozunk meg fejezetcímeket?
- Hány fejezetcímet szintet tudunk meghatározni?
- Milyen elemmel adunk meg címeket?

Lista

Számozatlan lista

```
<ul>
  <li>Kávé</li>
  <li>Tea</li>
  <li>Tej</li>
</ul>
```

Számozott lista

```
<ol>
  <li>Kávé</li>
  <li>Tea</li>
  <li>Tej</li>
</ol>

<ol start="50">
  <li>Kávé</li>
  <li>Tea</li>
  <li>Tej</li>
</ol>

<ol reversed="reversed">
  <li>Kávé</li>
  <li>Tea</li>
  <li>Tej</li>
</ol>
```

Egymásbaágyazás:

```
<ol>
  <li>Kávé</li>
  <li>Tea
    <ol>
      <li>víz</li>
      <li>fű</li>
      <li>édesítő</li>
    </ol>
  </li>
  <li>Tej</li>
</ol>
```

1. Kávé
2. Tea
  - a. víz
  - b. fű
  - c. édesítő
3. Tej

Böngésző támogatottság

A legtöbb jelentős böngésző támogatja az <ol> taget.

Definíció és használat

Az <ol> tag egy számozott listát definiál.

Használj az <li> taget a lista elemeinek meghatározásához.

Különbség a HTML 4.01 és a HTML5 között

A „start” attribútum elavult a HTML 4.01-ben de újra támogatott a HTML5-ben.

A „reversed” új a HTML5-ben.

A „compact” és a „type” attribútum elavult a HTML 4.01-ben és nincs támogatva a HTML5-ben.

Tippek és megjegyzések

Tipp: Számozatlan listához használj az <ul> taget.

Tipp: Használj CSS a stílus meghatározásához.

Attribútumok

(Új) : Új a HTML5-ben.

Attribútum	Érték	Leírás
reversed (Új)	reversed	Csökkenő sorrend meghatározása (9,8,7...)
start	szám	Kiindulási érték meghatározása a listában
type	1 A a I i	Listajelző megadása

Global Attributes

Az <ol> tag támogatja a globális attribútumokat a HTML5-ben.

Eseményattribútum

Az <ol> tag támogatja az esemény attribútumokat a HTML5-ben.

Definíciós lista

A definíciókat általánosan így használjuk:

```
<dl>
  <dt>fogalom</dt>
  <dd>leírás</dd>
  <dt>fogalom2</dt>
  <dd>leírás</dd>
  <dd>alternatív leírás</dd>
  <dd>másik alternatív leírás</dd>
</dl>

<dl>
  <dt>Kávé</dt>
  <dd>feket forró ital</dd>
  <dt>Tej</dt>
  <dd>Fehér hideg ital</dd>
</dl>
```

HTML elemek:

<dl>	Egy definíciós lista definiálása Definition List
<dt>	Egy definíciós lista egy elemét definiálja Definition Term
<dd>	Egy definíciós lista egy elemének leírása Definition Description

Képek beillesztése

Az img elem

Képek beillesztésénél ügyeljünk arra, hogy megmutassuk a szöveg és a beillesztett kép közötti kapcsolatot.

A weblapra illeszthető képtípusok:

- .jpg (Fényképek beillesztése)
- .gif (Rajzok beillesztése)
- .png (Vegyes felhasználásra)

```

```

A képnek és a html állománynak egy könyvtárban kell lenniük.

Még több attribútum:

```

```

Attribútumok:

- alt - Ha képek nem jeleníthetők meg, ez a szöveg jelenik meg
- width - szélesség
- height - magasság
- title - Ha kép felé visszük az egeret, ez a szöveg jelenik meg.

A title nem az img elem saját attribútuma. A title egy globális attribútum, bármely HTML elemmel használható.

A figure elem

A figure elem, egy új elem a HTML5-ben. Képek és ábrákat vehetünk vele közre. A figure elem mellett rendelkezésünkre áll a figcaption, amellyel egy kép számára feliratot határozhatunk meg.

```
<figure>
  
  <figcaption>Felirat</figcaption>
</figure>

<figure style="
  border-style: solid;
```

```
border-width: 1px;
display: inline-block;
padding: 10px;
">

<figcaption>
    A kép felirata
</figcaption>
</figure>
```

A figure elemmel, természetesen több kép is leírható:

```
<figure>



<figcaption>A képek leírása</figcaption>
</figure>
```

Táblázat

Alap

```
<table border="1">
<tr>
<th>Hónap</th>
<th>Spórolás</th>
</tr>
<tr>
<td>Január</td>
<td>1000 Ft</td>
</tr>
</table>
```

Hónap	Spórolás
Január	1000 Ft

Felirat

```
<table border="1">
<caption>Költségvetés</caption>
<tr>
<th>Hónap</th>
<th>Spórolás</th>
</tr>
<tr>
<td>Január</td>
<td>1000 Ft</td>
</tr>
</table>
```

Költségvetés	
Hónap	Spórolás
Január	1000 Ft

Összefoglaló

A böngészők nem jelenítik meg, de felolvasóprogram esetén hasznos.

```
<table border="1" summary="Ez a táblázat bemutatja a család spórolásának eseményeit">
<caption>Költségvetés</caption>
<tr>
<th>Hónap</th>
<th>Spórolás</th>
</tr>
<tr>
<td>Január</td>
<td>1000 Ft</td>
</tr>
</table>
```

Költségvetés	
Hónap	Spórolás
Január	1000 Ft

Cella nyúlás

```
<table border="1">
<tr>
<th colspan="2">Költségek</th>
</tr>
<tr>
<th>Hónap</th>
<th>Spórolás</th>
</tr>
<tr>
<td>Január</td>
<td>1000 Ft</td>
</tr>
</table>
```

Költségek	
Hónap	Spórolás
Január	1000 Ft

Sor nyúlás

```
<table border="1">
  <tr>
    <th rowspan="2">Adatok</th>
    <th>Hónap</th>
    <th>Spórolás</th>
  </tr>
  <tr>
    <td>Január</td>
    <td>1000 Ft</td>
  </tr>
</table>
```

Adatok	Hónap	Spórolás
	Január	1000 Ft

Sor és cellanyúlás

A rowspan tulajdonsággal sorokat vonhatunk össze.

```
<table border="1">
  <tr>
    <th rowspan="3">Adatok</th>
    <th colspan="2">Költségek</th>
  </tr>
  <tr>
    <th>Hónap</th>
    <th>Spórolás</th>
  </tr>
  <tr>
    <td>Január</td>
    <td>1000 Ft</td>
  </tr>
</table>
```

Adatok	Költségek	
	Hónap	Spórolás
	Január	1000 Ft

# Táblázat

Dolgozók			
Szolnoki dolgozók			
	No.	Név	Fizetés
Alap	1	Penge Béla	8500000
	2	Rendi Lajos	8600000
Összes			17100000

Oszlopok külön stílusban

A colgroup és a col elem segítségével beállíthatjuk az oszlopok háttérszínét.

Az első két oszlop sárga, az utána következő egy oszlop kék.

```
<table>
  <colgroup>
    <col span="2" style="background-color:yellow">
    <col style="background-color:blue">
  </colgroup>
  <tr>
    <th>No.</th>
    <th>Név</th>
    <th>Település</th>
  </tr>
  <tr>
    <td>1</td>
    <td>Nagy József</td>
    <td>Szolnok</td>
  </tr>
  <tr>
    <td>2</td>
    <td>Peres Ferenc</td>
    <td>Szeged</td>
  </tr>
</table>
```

A <colgroup> tag, egy <table> elem gyermek kell legyen. A <caption> elemek után kell állnia, és <thead>, <tbody>, <tfoot> és <tr> elemek előtt.

No.	Név	Település
1	Nagy József	Szolnok
2	Peres Ferenc	Szeged

A táblázat részeinek csoportosítása

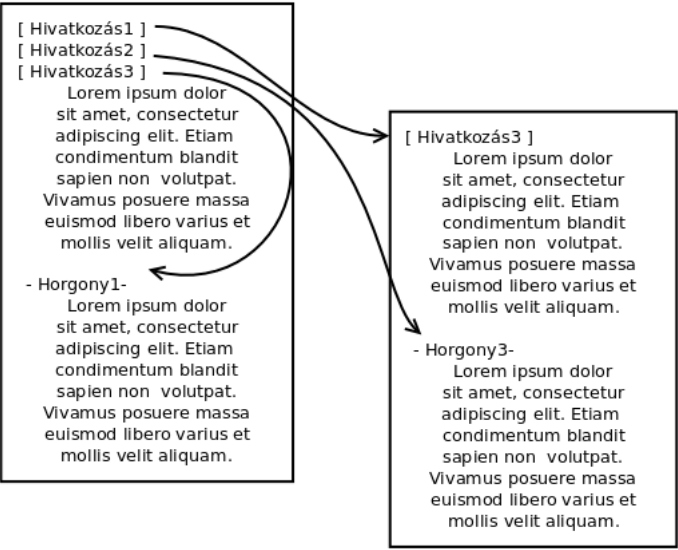
```
<table>
<thead>
<tr>
<th>Név</th>
<th>Fizetés</th>
</tr>
</thead>
<tbody>
<tr>
<td>Nagy János</td>
<td>3850000</td>
</tr>
<tr>
<td>Kemény Ferenc</td>
<td>2870000</td>
</tr>
</tbody>
<tfoot>
<tr>
<td>Összesen</td>
<td>6720000</td>
</tr>
</tfoot>
</table>
```

Név	Fizetés
Nagy János	3850000
Kemény Ferenc	2870000
Randi Lajos	1975000
Tény Imre	3840500
Zila Péter	3510500
Összesen	16046000

Hiperhivatkozás

A HTML hivatkozások miatt lett Hyper Text Markup Language. Egy szöveg azért hypertext, mert vannak benne hivatkozások. A hivatkozásokra kattintva, valamilyen másik szövegbe jutunk, egy pillanat alatt.

A hivatkozás célja lehet egy másik weboldal. Lehet a helyi weboldalon egy adott pont, vagy a másik weboldalon egy adott pont. Utóbbihoz meg kell valahogy jelölni, hol van az ugrás célja. A weblapon belüli célokat úgynevezett horgonnyal jelöljük.



Az alábbiakban megnézzük, hogyan kell horgonyokat és hivatkozásokat elhelyezni.

Horgony:

```
<a name="alul">
```

Hivatkozás:

```
<a href="masik.html">Másik</a>
```

Kép hivatkozásként:

```
<a href="masik.html"></a>
```

Adott az alábbi HTML oldal, négy bekezdéssel. Horgony helyezünk el a harmadik és negyedik bekezdéshez. Ehhez valamilyen nevet kell adnunk a bekezdés számára. Egy hivatkozással, az adott horgonyhoz ugorhat a felhasználó, böngészőből, egyetlen kattintással.

negybekezdes.html

```
<!DOCTYPE html>
<html>
<head>
```

```

</head>
<body>

<h2>1</h2>
<p>
Lorem ipsum dolor sit amet, consectetur adipisicing elit, sed do eiusmod tempor incididunt ut labore et dolore magna aliqua. Ut enim ad minim veniam, qu
</p>

<h2>2</h2>
<p>
Lorem ipsum dolor sit amet, consectetur adipisicing elit, sed do eiusmod tempor incididunt ut labore et dolore magna aliqua. Ut enim ad minim veniam, qu
</p>

<a name="harmadik"></a>
<h2>3</h2>
<p>
Lorem ipsum dolor sit amet, consectetur adipisicing elit, sed do eiusmod tempor incididunt ut labore et dolore magna aliqua. Ut enim ad minim veniam, qu
</p>

<a name="negyedek"></a>
<h2>4</h2>
<p>
Lorem ipsum dolor sit amet, consectetur adipisicing elit, sed do eiusmod tempor incididunt ut labore et dolore magna aliqua. Ut enim ad minim veniam, qu
</p>

</body>
</html>

```

A weblap tetejére, helyezzünk el két hivatkozást az horgonyokra.

```
<a href="#harmadik">Harmadik bekezdés</a>
```

```
<a href="#negyedek">Negyedek bekezdés</a>
```

A teljes kód:

#### negybekezdés.html

```

<!DOCTYPE html>
<html>
<head>

</head>
<body>

<a href="#harmadik">Harmadik bekezdés</a>
<a href="#negyedek">Negyedek bekezdés</a>

<h2>1</h2>
<p>
Lorem ipsum dolor sit amet, consectetur adipisicing elit, sed do eiusmod tempor incididunt ut labore et dolore magna aliqua. Ut enim ad minim veniam, qu
</p>

<h2>2</h2>
<p>
Lorem ipsum dolor sit amet, consectetur adipisicing elit, sed do eiusmod tempor incididunt ut labore et dolore magna aliqua. Ut enim ad minim veniam, qu
</p>

<a name="Harmadik"></a>
<h2>3</h2>
<p>
Lorem ipsum dolor sit amet, consectetur adipisicing elit, sed do eiusmod tempor incididunt ut labore et dolore magna aliqua. Ut enim ad minim veniam, qu
</p>

<a name="Negyedek"></a>
<h2>4</h2>
<p>
Lorem ipsum dolor sit amet, consectetur adipisicing elit, sed do eiusmod tempor incididunt ut labore et dolore magna aliqua. Ut enim ad minim veniam, qu
</p>

</body>
</html>

```

Nézzük meg, hogyan jelenik meg a böngészőben a hiperhivatkozás, a következő ábrán.

#### [Harmadik bekezdés](#) [Negyedek bekezdés](#)

Lorem ipsum dolor sit amet, consectetur adipisicing elit, sed do eiusmod tempor incididunt ut labore et dolore magna aliqua. Ut enim ad minim veniam, quis nostrud exercitation ullamco laboris nisi ut aliquip ex ea commodo consequat. Duis aute irure dolor in reprehenderit in voluptate velit esse cillum dolore eu fugiat nulla pariatur. Excepteur sint occaecat cupidatat non proident, sunt in culpa qui officia deserunt mollit anim id est laborum. Lorem ipsum dolor sit

Figure 1: Hiperhivatkozás

A még meg nem látogatott hiperhivatkozásokat, a legtöbb böngésző kék színnel jelöli, aláhúzva.

A hiperhivatkozásokat az alábbi beépített weblapon kipróbálhatja.

[Harmadik bekezdés](#) [Negyedik bekezdés](#)

## 1

Lorem ipsum dolor sit amet, consectetur adipisicing elit, sed do eiusmod tempor incididunt ut labore et dolore magna aliqua. Ut enim ad minim veniam, quis nostrud exercitation ullamco laboris nisi ut aliquip ex ea commodo consequat. Duis aute irure dolor in reprehenderit in voluptate velit esse cillum dolore eu fugiat nulla pariatur. Excepteur sint occaecat cupidatat non proident, sunt in culpa qui officia deserunt mollit anim id est laborum. Lorem ipsum dolor sit amet, consectetur adipisicing elit, sed do eiusmod tempor incididunt ut labore et dolore magna aliqua. Ut enim ad minim veniam, quis nostrud exercitation ullamco laboris nisi ut aliquip ex ea commodo consequat. Duis aute irure dolor in reprehenderit in voluptate velit esse cillum dolore eu fugiat nulla pariatur. Excepteur sint occaecat cupidatat non proident, sunt in culpa qui

A hipertexthivatkozás mutathat másik weblapra is, illetve azon a weblapon belüli is hivatkozhatunk egy horgonyra.

Készítsünk két weblapot, elso.html és masodik.html néven, a tartalmuk, lehet a fenti mintának megfelelően a négy bekezdés. Az weboldalak tetejét a következőre változtassuk:

Az elso.html tetején:

```
<a href="masodik.html">Második welap</a><br>
<a href="masodik.html#negyedik">Második weblap, negyedik bekezdése</a>

<h1>Első weblap</h1>
```

Az masodik.html tetején:

```
<a href="elso.html">Vissza az első weblapra</a>

<h1>Második weblap</h1>
```

Az elso.html weblapon két link (hivatkozás) mutat a masodik.html weblapra. Az első hivatkozás a masodik.html tetejére mutat. A második hivatkozás a masodik.html negyedik bekezdésére mutat. Próbáljuk ki a működést az alábbi beépített weblapon:

[Második welap](#)  
[Második weblap, negyedik bekezdése](#)

# Első weblap

## 1

Lorem ipsum dolor sit amet, consectetur adipisicing elit, sed do eiusmod tempor incididunt ut labore et dolore magna aliqua. Ut enim ad minim veniam, quis nostrud exercitation ullamco laboris nisi ut aliquip ex ea commodo consequat. Duis aute irure dolor in reprehenderit in voluptate velit esse cillum dolore eu fugiat nulla pariatur. Excepteur sint occaecat cupidatat non proident, sunt in culpa qui officia deserunt mollit anim id est laborum. Lorem ipsum dolor sit amet, consectetur adipisicing elit, sed do eiusmod tempor incididunt ut labore et dolore magna aliqua. Ut enim ad minim veniam, quis nostrud exercitation ullamco laboris nisi

## Videó

Videókat a „video” elemmel építhetünk weboldalunkba. A video elem a következő attribútumokkal rendelkezik:

- src
- crossorigin
- poster
- preload
- autoplay
- mediagroup
- loop
- muted
- controls
- width
- height

```
<video src="sample3.ogg" width="300" height="300"></video>
```

- Működik Firefox, Opera, Chrome alatt
- IE8 nem jelenít meg semmit

Többféle formátumban felrakva



```
<video width="300" height="300">
  <source src="sample3.ogg">
  <source src="sample1.mp4">
</video>
```

Típusok megadása

```
<video width="..." height="..." controls>
<source src="...mp4" type='video/mp4; codecs="avc1.42E01E, mp4a.40.2"'>
<source src="...webm" type='video/webm; codecs="vp8, vorbis"'>
<source src="...ogv" type='video/ogg; codecs="theora, vorbis"'>
</video>
```

A következő kódokkal kell számolnunk:

- MPEG-4
- H.264
- Ogg Theora/Vorbis/Opus
- WebM - VP8
- WebM - VP9

A konvertáláshoz ajánlott program:

- [ffmpeg](https://www.ffmpeg.org/) [<https://www.ffmpeg.org/>]
  - <http://www.wikihow.com/Install-FFmpeg-on-Windows> [<http://www.wikihow.com/Install-FFmpeg-on-Windows>]
  - <http://ffmpeg.zeranoe.com/builds/> [<http://ffmpeg.zeranoe.com/builds/>]

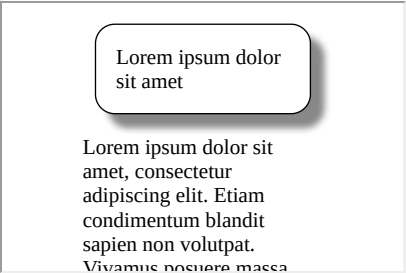
Hang

```
<audio controls="controls">
  <source src="hangfajl.ogg" type="audio/ogg">
  <source src="hangfajl.mp3" type="audio/mpeg">
A böngésződ nem támogatja az audio elemet.
</audio>
```

Beépített weblap

Egy weblap beépítve

```
<iframe src="https://szit.hu/download/doboz.html" width="300" height="200"></iframe>
```

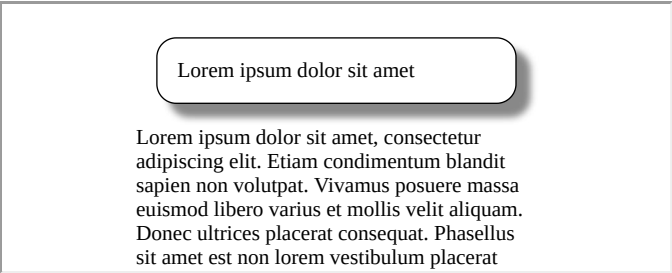


Szegély:

```
frameborder="0"
```

Egy link céljaként

```
<iframe src="https://szit.hu/download/doboz.html" name="iframe_a" width="500" height="200"></iframe>
<p><a href="https://szit.hu/doku.php?id=oktatas" target="iframe_a">http://szit.hu/doku.php?id=oktatas</a></p>
```



<http://szit.hu/doku.php?id=oktatas>

Sanbox

Ha van egy index.html weboldalunk, amelybe beágyazzuk a belso.html weblapot, a belso weblap műveletei automatikusan érvényesek a index.html weblapra is. Ha például van belso.html egy „a” elemmel létrehozott horgony, amihez kattintással ugrunk, az index.html weblap is görgetésre kerül. Az index.html head elemei között megadott beállítások is érvényesek a belso.html oldalon. Ha ezeket szeretnénk elkerülni, használjunk sandbox-ot.

```
<iframe src="belso.html" width="500" sandbox></iframe>
```

Űrlap

Az űrlapokkal valósítjuk meg az interaktív weblapokat. Egy weblap nem attól interaktív, hogy különböző dolgok mozognak rajta. Attól lesz interaktív, hogy a felhasználó adatokat küld a szerver felé. Ezt pedig űrlapok segítségével lehet megtenni. Az űrlap mindig adatokat kér tőlünk, amit egy „Elküldés” gombbal el kell küldenünk a szervernek.

Űrlap létrehozása

Űrlapot a form elemmel tudunk beszúrni. Van két kötelező tulajdonsága, az action és a method. Az action értékeként adjuk meg, hogy milyen szerveroldali script fogja feldolgozni az űrlapot. A method értékeként adjuk meg, hogy milyen metódussal küldjük az űrlapelemet:

urlap.html

```
<form action="akcio.php" method="POST">
<fieldset>
  <legend>Teljesnév</legend>
  <label for="vnev">Vezetéknév:</label>
  <input type="text" name="vnev" value="" id="vnev" /><br />
  <label for="knev">Keresztnév:</label>
  <input type="text" name="knev" value="" id="knev" /><br />
  <input type="submit" value="Elküld" />
</fieldset>
</form>
```

Teljesnév

Vezetéknév:

Keresztnév:

Elküld

Kétféle metódussal küldhetjük el űrlapjainkat:

- GET
- POST

A GET metódus

Az adatokat a meghívó URL paramétereként adjuk át.

A POST metódus

Az adatok egy adatként kerülnek átadásra a szerver számára.

Text típus - szöveges mező

- size - méret
- maxlength - hány karakter hosszú

```
<input maxlength="10" size="10">
```

Űrlap elemek

Beviteli mező

```
<input type="text">
```

Az input tag-el többféle űrlapelem is meghatározható. Hogy melyik jelenik meg az mindig a type tulajdonságtól függ. A text értékkel egy beviteli mezőt jeleníthetünk meg. Kötelező tulajdonság még a name. Ezzel adjuk meg, milyen néven hivatkozhat a beviteli mező tartalmára a feldolgozó program.

Nem kötelező tulajdonság a value. Ezzel kezdőértéket írhatunk a bevitelimezőbe.

```
<input type="text" name="nev" value="Névtelen">
```

Névtelen

A for tulajdonság

A for tulajdonsággal egy label elemet összerendelhetünk egy input elemmel. A for a label attribútumaként kell megadnunk:

```
<label for="nevaz"></label><input type="text" name="nev" value="Névtelen" id="nevaz">
```

Teljes név:

Ennek hatására, ha kattintok a label elem tartalmára, aktiválódik az input elemünk. A for használatához az input elemben lenni kell egy id tulajdonságnak, amelynek értékére hivatkozunk a forban.

Elküldés gomb

```
<input type="submit">
```

Küldés

Törlés gomb

```
<input type="reset">
```

Visszaállítás

Még egy gomb

```
<button type="button">Klikkelj ide!</button>
```

Klikkelj ide!

Rádiógombok

```
<form action="feldolgoz.php" method="POST">
  <input type="radio" name="nem" value="ferfi" id="ferfi"/>
  <label for="ferfi">Férfi</label><br />
```

```
<input type="radio" name="nem" value="no" id="no"/>
<label for="no">Nő</label><br />
<input type="submit" value="Mehet" />
</form>
```

☐ Férfi  
☐ Nő

A rádiógomboknál nagyon fontos, hogy name tulajdonság azonos értékkel legyen kitöltve az egy csoportba tartozó rádiógomboknál. A böngésző ebből tudja, hogy mely rádiógombok tartoznak egy csoportba. Ha kattintunk egyik rádiógombra, az előzőleg aktív rádiógomb jelöltségének meg kell szűnnie, ha az egy csoportban van az újonnan kattintottal.

Jelölőnégyzet

```
<form>
<input type="checkbox" name="jarmu" value="kerepar" id="kerepar" />
<label for="kerepar">Van kerékpárom</label><br />
<input type="checkbox" name="jarmu" value="kocsi" id="kocsi" />
<label for="kocsi">Van kocsim</label><br />
<input type="submit" value="Mehet" />
</form>
```

☐ Van kerékpárom  
☐ Van kocsim

Szöveghely

```
<textarea rows="3" cols="20">
Lorem ipsum dolor sit amet.
</textarea>
```

Lorem ipsum dolor sit amet.

Legördülő listadoboz

Egyszerű kombóbox

```
<select>
<option>Opel</option>
<option>Ford</option>
<option>Volvo</option>
<option>Saab</option>
<option>Mercedes</option>
<option>Audi</option>
</select>
```

Opel

Alcsoportok

```
<select>
<optgroup label="Svéd kocsik">
  <option value="volvo">Volvo</option>
  <option value="saab">Saab</option>
</optgroup>
<optgroup label="Német kocsik">
  <option value="mercedes">Mercedes</option>
  <option value="audi">Audi</option>
</optgroup>
</select>
```

Volvo

Attribútumok

```
<select name="tipus">
<option value="opel">Opel</option>
<option value="ford">Ford</option>
<option value="volvo" selected="selected">Volvo</option>
<option value="saab">Saab</option>
<option value="mercedes">Mercedes</option>
<option value="audi">Audi</option>
</select>
```

Listadoboz

```
<select name="tipus" multiple size="3">
<option value="opel">Opel</option>
<option value="ford">Ford</option>
<option value="volvo" selected="selected">Volvo</option>
<option value="saab">Saab</option>
<option value="mercedes">Mercedes</option>
<option value="audi">Audi</option>
</select>
```

Opel
 Ford
 Volvo

Azonosítók:

```
<select name='jarmuvek'>
<option value='1'>Opel</option>
<option value='2'>Ford</option>
```

```
<option value='3'>Volvo</option>
<option value='4'>Saab</option>
<option value='5'>Mercedes</option>
<option value='6'>Audi</option>
</select>
```

Vizuális elemek

A form számára létrehozhatunk keretet, keretfeliratot, csak a jobb vizuális megjelenés érdekében.

```
<form>
<fieldset>
<legend>Személyes:</legend>
Név: <input type="text" size="30" /><br />
Email: <input type="text" size="30" /><br />
Születés dátuma: <input type="text" size="10" />
</fieldset>
</form>
```

Személyes:

Név:

Email:

Születés dátuma:

Új űrlap elemek

Színválasztó

- Chrome, Opera

```
<form action="feldolgoz.php">
Válassz színt: <input type="color" name="favcolor"><br>
<input type="submit">
</form>
```

Válassz színt:

Küldés

Naptár

- Chrome, Safari, Opera

```
<form action="feldolgoz.php">
Születésnap: <input type="date" name="bday">
<input type="submit">
</form>
```

Születésnap: éééé . hh . nn .

Küldés

E-mail ellenőrző

- Firefox, Chrome, Opera

```
<form action="feldolgoz.php">
E-mail: <input type="email" name="usremail">
<br>
<input type="submit">
</form>
```

E-mail:

Küldés

Léptetődoboz

- Chrome, Safari, Opera

```
<form action="feldolgoz.php">
Terjedelem (1 és 5 között): <input type="number" name="quantity" min="1" max="5">
<input type="submit">
</form>
```

Terjedelem (1 és 5 között):

Küldés

Nagyság beállítása

- Chrome, Safari, Opera

```
<form action="feldolgoz.php" method="get">
Pontok: <input type="range" name="points" min="1" max="10"><input type="submit">
</form>
```

Pontok:

Küldés

URL

- Firefox, Chrome, Opera

```
<form action="feldolgoz.php">
Add meg a weblapod címét: <input type="url" name="homepage"><br>
<input type="submit">
</form>
```

Add meg a weblapod címét:

Küldés

Titkosítás

- Firefox, Chrome, Safari, Opera

```
<form action="feldolgoz.php" method="get">
Félhasználónév: <input type="text" name="usr_name">
Titkosítás: <keygen name="security">
<input type="submit">
</form>
```

Felhasználónév:  Titkosítás:

Kimenet

- Firefox, Chrome, Safari, Opera

```
<form oninput="x.value=parseInt(a.value)+parseInt(b.value)">0
<input type="range" name="a" value="50">100 +
<input type="number" name="b" value="50">=
<output name="x" for="a b"></output>
</form>
```

0  100 +  =

Automatikus fókus

- Firefox, Chrome, Safari, Opera

```
<form action="feldolgoz.php">
Vezetéknév:<input type="text" name="vnev" autofocus="autofocus"><br>
Keresztnév: <input type="text" name="knev"><br>
<input type="submit">
</form>
```

Vezetéknév:   
Keresztnév:

Új form és input attribútumok

A HTML5 form és input elemei néhány új attribútumot tartalmaz.

<form> / <input> autocomplete attribútum

Az automatikus kiegészítés ki vagy bekapcsolása. Néhány böngészőben aktiválni kell ezt a lehetőséget.

Lehetséges értékek:

- on
- off

```
<form action="feldolgoz.php" autocomplete="on">
Vezetéknév:<input type="text" name="vnev"><br>
Keresztnév: <input type="text" name="knev"><br>
E-mail: <input type="email" name="email" autocomplete="off"><br>
<input type="submit">
</form>
```

Vezetéknév:

Keresztnév:

E-mail:

Az opera nem támogatja.

A <form> novalidate Attribútuma

A novalidate attribútum egy logikai típus.

Ha be van állítva a form elküldésekor nem kér megerősítést.

A Safari nem támogatja.

```
<form action="feldolgoz.php" novalidate>
E-mail: <input type="email" name="email">
<input type="button" value="Mehet">
</form>
```

Az <input> autofocus attribútuma

Az autófókusz attribútum egy logikai típus.

Ha be van állítva, az oldal betöltődése után ez az elem kapja meg a fókuszt.

```
Név:<input type="text" name="nev" autofocus>
```

Az <input> form attribútum

Egy input elem melyik formhoz tartozik.

Az IE nem támogatja.

Akkor hasznos, ha egy input mező azon a formon kívül van, amelyhez tartozik.

```
<form action="feldolgoz.php" id="form1">
  Vezetéknév: <input type="text" name="vnev"><br>
  <input type="submit" value="Mehet">
</form>
```

Keresztnév: <input type="text" name="knev" form="form1">

Újabb példa:

```
<form action="feldolgoz.php" id="form1">
Vezetéknév: <input type="text" name="vnev"><br>
<input type="submit" value="Submit"></form>
```

The "Keresztnév" mező a formon kívül van, de mégis része a formnak.

Keresztnév: <input type="text" name="knev" form="form1">

Vezetéknév:

The "Keresztnév" mező a formon kívül van, de mégis része a formnak.

Keresztnév:

Az <input> formaction attribútum

A formaction segítségével alternatív feldolgozó script adható meg.

A submit típusú elküldőgomboknál használjuk.

```
<form action="reg.php">
  Név: <input type="text" name="nev"><br>
  <input type="submit" value="Regisztrál"><br>
  <input type="submit" formaction="check.php" value="Ellenőriz">
</form>
```

Újabb példa:

```
<form action="feldolgoz.php">
Vezetéknév: <input type="text" name="vnev"><br>
Keresztnév: <input type="text" name="knev"><br>
<input type="submit" value="Küldés"><br>
<input type="submit" formaction="feldolgozAdmin.php" value="Küldés adminként">
</form>
```

Vezetéknév:

Keresztnév:

formenctype attribútum az input elem számára

- Firefox, Chrome, Safari, Opera

Az küldött adatok multipart (fájlfeltöltésnél használjuk) formában kódolása.

```
<form action="feldolgoz.php" method="post">
  First name: <input type="text" name="fname"><br>
  <input type="submit" value="Submit">
  <input type="submit" formenctype="multipart/form-data"
    value="Küldés Multipart/form-data adatként">
</form>
```

Vezetéknév:

Az <input> formmethod attribútuma

A submit gomb számára megadhatjuk milyen metódussal küldjük az adatokat. A formmethod felülírja a form elem method attribútumát.

Használható submit és image típus esetén.

```
<form action="demo_form.asp" method="get">
Vezetéknév: <input type="text" name="vne"><br>
Keresztnév: <input type="text" name="knev"><br>
<input type="submit" value="Küldés">
<input type="submit" formmethod="post" formaction="feldolgoz.php"
  value="Küldés POST-al">
</form>
```

Az input elem formnovalidate attribútuma

A novalidate attribútuma egy logikai típus.

Ha be van állítva, az input elem érvényessége nem lesz ellenőrizve küldéskor.

A formnovalidate attribútum felülírja a form elem novalidate attribútumát.

A formnovalidate a submit típus esetén használható.

```
<form action="feldolgoz.php">
  E-mail: <input type="email" name="email"><br>
  <input type="submit" value="Mehet"><br>
  <input type="submit" formnovalidate value="Küldés ellenőrzés nélkül">
</form>
```

Az input elem formtarget attribútuma

A formtarget attribútummal kulcsszavakat adhatunk meg, szabályozzák a küldés után hol jelenjen meg a válasz.

A formtarget attribútum felülírja a form elem target attribútumát.

A formtarget attribútum a használható a következő attribútumokkal:

- submit
- image

Egy másik ablak megadása:

```
<form action="feldolgoz.php">
  Teljesnév: <input type="text" name="nev"><br>
  <input type="submit" value="Küldés">
  <input type="submit" formtarget="_blank" value="Küldés másik ablakba">
</form>
```

### Az input elem height és width attribútuma

Egy input elem szélességének és magasságának megadása.

Csak a következő input típus esetén használható:

```
image
<input type="image" src="kuldokep.gif" alt="Mehet" width="48" height="48">
```

### Az input elem list attribútuma

A list attribútum kapcsolódik a <datalist> elemhez. Ha beírunk egy karaktert, felkínálja a listából az azonos kezdőbetűjelűeket.

A Safari nem támogatja.

Egy input elem datalistával:

```
<input list="gyumolcsok">

<datalist id="gyumolcsok">
  <option value="alma">
  <option value="körte">
  <option value="barack">
  <option value="szilva">
  <option value="málna">
</datalist>
```

Gyümölcs:

### Az input elem min és max attribútumai

Egy input elem minimum és maximum értékét adhatjuk meg vele.

A következő input típusokkal használható:

- number
- range
- date
- datetime
- datetime-local
- month
- time
- week

Legkésőbbi dátum 2005-05-01:

```
<input type="date" name="bday" max="2005-05-01">
```

Csak ez utáni dátum 2008-01-01:

```
<input type="date" name="bday" min="2008-01-01">
```

Terjedelem 1 és 10 között:

```
<input type="number" name="quantity" min="1" max="10">
```

### Az input multiple attribútuma

A multiple attribútum logikai típusú adatot tartalmazhat.

Ha be van állítva, akkor több értéket is megadhat egy input elemben.

A következő input típusoknál használható:

- email
- file

A file upload field that accepts multiple values:

Fájlok kiválasztása:

### Az <input> pattern attribútuma

A pattern attribútummal megadható egy szabályos kifejezés, amellyel ellenőrizhető az elem tartalma.

A pattern a következő típusú input attribútumokkal használható: text, password, url, tel, email, és search.

A mindenhol használható title attribútumban érdemes leírni a felhasználó számára egy segítséget.

A Safari nem támogatja.

A példa kedvéért készítsünk egy olyan szabályos kifejezést, amely 4 számot fogad el irányítószámként:

Írányítószám: `<input type="text" name="iranyitoszam" pattern="[0-9]{4}" title="Négy szám adható meg.">`

Írányítószám:

A `<input>` **placeholder** attribútum

A placeholder attribútum egy rövid leírás arról, mit írhatunk az inputmezőbe.

A következő típusú input elemekkel működik: text, password, url, tel, email, és search.

`<input type="text" name="nev" placeholder="Teljesnév">`

Teljesnév

A `<input>` **required** attribútum

A required attribútum egy logikai típust tartalmazhat.

Ha be van állítva, kötelező kitölteni az elem tartalmát elküldés előtt.

A required attribútum a következő típusokkal működik: text, password, url, tel, email, search, date pickers, number, checkbox, radio, és file.

A safari nem támogatja.

Teljesnév: `<input type="text" name="nev" required>`

Teljesnév:

Az `<input>` **step** attribútum

A step attribútummal megadható a legális számok intervalluma egy input elem esetén.

Ha például `step=„4”`, akkor elfogadható a -8, -4, 0, 4, 8 stb.

A max és min attribútumokkal a leghatékonyabb.

A következő típusú input elemekkel használható: number, range, date, datetime, datetime-local, month, time és a week.

A firefox nem támogatja!

`<input type="number" name="pontok" step="4">`

Step 4-es értékkel:

Min 2, max 22:

Részletek

A details a HTML5 egy új eleme. Segítségével egy szöveghez tartozó részletes leírás nyitható le, vagy zárható be, igény szerint.

`<details>  
 <summary>Lorem ipsum</summary>  
 <p>Lorem ipsum dolor est amet</p>  
</details>`

A böngészők támogatása:

Goog Chrome	IE/Edge	Firefox	Safari	Opera
12.0	79.0	49.0	6.0	15.0

Jelölő elemek

A jelölő elemekkel a többi elemet vagy tartalmat csoportosítjuk, megjelöljük, hogy azokat önálló egységként tudjuk formázni CSS-ben.

div

A div elem a division, ejtsd: [diˈviʒən] szóból van, amelynek több jelentése is van: csoport, elválasztás, közfal, megoszlás, szakasz, tanács stb.

A div elemmel blokk szintű dobozokat hozhatunk létre.

span

A span angol szó, ejtsd: [spæn] jelentései: arasz, átfog, átível, fesztáv, ív stb.

A span elemmel inline szintű elemet hozunk létre.

Például egy szövegrészletet külön szeretnék formázni.

Lorem ipsum `<span>dolores</span>` sit amet.

```
span {  
  text-decoration: undrline;  
}
```

Vagy

Lorem ipsum `<span class="dol">dolores</span>` sit amet.

```
span.dol {  
  text-decoration: undrline;  
}
```



A data attribútum

A HTML5 bevezette az egyéni attribútumokat, különböző adatok tárolására. Az egyéni attribútumokat a data- előtaggal kezdjük, amit egy leírónév követ. Egy elem tetszőleges számú egyéni attribútumot tartalmazhat.

```
<div data-magas="35" data-kor="37">János</div>
<button
  data-az="1"
  data-nev="János"
  data-telepules="Szolnok"
  id="torlesgomb"
  >Törlés</button>
```

Az adatattribútumokban csak szöveget lehet tárolni, objektumokat nem tárolhatunk. Az objektumok esetleg szerializálással tárolhatók.

Olvasás és írás JavaScriptben

```
var button = document.getElementById('torlesgomb');

var az = button.getAttribute('data-az');
var nev = button.getAttribute('data-nev');

button.setAttribute('data-az', '2');
button.setAttribute('data-nev', 'Pál');
```

Olvasás írás jQuery-ben

```
var az = $('#torlesgomb').attr('az');
var id = $('#torlesgomb').attr('nev');

$('#torlesgomb')
  .attr('data-az', '2')
  .attr('data-nev', 'Pál');
```

Elérés dataset API-n keresztül

```
var button = document.getElementById('torlesgomb');

var az = button.dataset.az;
var nev = button.dataset.nev;

button.dataset.az = '2';
button.dataset.nev = 'Pál';
```

Hozzáférés CSS-ből

```
[data-telepules='Szolnok'] {
  background-color: red;
}

li[data-telepules='Szolnok'] {
  background-color: red;
}
```

Tároló elemek

A tárolóelemekről

A tároló elemeket a HTML5-ben hozták létre, az oldal tartalmainak felosztására. Korábban erre csak a div elemek áltak rendelkezésre, amelyeket mindig egy azonosítóval vagy osztállyal jelöltünk. A HTML5-től a következő elemek is rendelkezésre állnak.

<header >	A weboldal felső része
<nav>	Navigációs része
<article>	Független, önálló tartalom
<section>	A weboldal egy része. Nem használjuk burkoló elemként!
<aside>	Oldaljegyzet
<footer>	A weboldal alsó része

Az Internet Exploler nem ismeri ezeket az elemeket, ezért JavaScript és CSS segítségével szokás az oldalt kompatibilissé tenni.

IE-hez javítás JavaScriptben és CSS-ben:

```
document.createElement("header");
document.createElement("footer");
document.createElement("hgroup");
document.createElement("section");
document.createElement("article");
document.createElement("nav");

header, footer, hgroup, section, article, nav {
  display: block;
}
```

Tárolóelem példa

```
<article>
  <header>
    <h1>A macskák élete</h1>
  </header>
  <p>Első bekezdés</p>
  <aside>
    <q>A nagymacskák, azok másféle állatok</q>
  </aside>
  <p>A macskák élete</p>
</article>
```

A macskák élete

Első bekezdés

„A nagymacskák azok másféle állatok”

A macskák élete

Fejrészcsoport

```
<header>
  <h1>Programozás</h1>
  <h2>Objektum orientált megvalósítások</h2>
</header>
```

Programozás

Objektum orientált megvalósítások

Átírányítás

```
<meta http-equiv="refresh" content="0; url=http://masikdomain.valahol">
```

Esetleg 10 másodpercre állítjuk, így 10 másodperc múlva átírányítunk:

```
index.html
<html>
<head>
<!--A felhasználó átírányítása másik webhelyre -->
<title>redirect</title>
<meta http-equiv="refresh" content="10; URL=http://www.masikdomain.valahol">
</head>
<body>
Ez az oldal el lett távolítva. 10 másodpercen belül új oldalra írányítunk át.
Ha ez nem történik meg, kattints a következő linkre:
<a href="http://www.masikdomain.valahol">Másik tartománynév</a>.
</body>
</html>
```

A favicon

A favicon (favorites icon, vagyis kedvenc ikon) egy weboldalhoz vagy weblaphoz tartozó ikon.

A böngészők különböző helyeken jelenítenek meg. Ilyen például a böngésző címsora, a böngészőfül eleje, a könyvjelzők előtt.

Ha nincs semmi beállítva, akkor a webszerver gyökérkönyvtárában elhelyezett favicon.ico nevű fájl kerül megjelenítésre.

A böngészők ma már az ICO mellett már GIF és PNG fájlokat is tudnak kezelni (akár animáltakat is).

A fájlnak nem kell az adott szerveren lenni.

Mérete 16×16.

A HTML5-ben elég ennyi:

```
<link rel="icon" href="ikon.ico" />
```

Korábbi HTML verziók:

```
<link rel="shortcut icon" type="image/png" href="/utvonal/image.png" />
<link rel="shortcut icon" href="/utvonal/images/favicon.ico" />
<link rel="icon" type="images/gif" href="http://domain.com/image.gif" />
<link rel="icon" type="images/vnd.microsoft.icon" href="http://zold.and/ikon.ico" />
```

Speciális karakterek

Vannak olyan karakterek, amelyeknek speciális szerepük van HTML, vagy egyszerűen csak nem szerepelt a korábbi kódtáblákban. Az ilyen karakterek megjelenítéséhez úgynevezett entitásokat használhatunk. Az entitásnak lehet neve és mindig van egy kódja is. A következő táblázat az entítások listáját tartalmazza az teljesség igénye nélkül.

Karakter	Kódnév	Számkód	
<	&lt;	&#60;	
>	&gt;	&#62;	
&	&amp;	&#38;	
”	&quot;	&#34;	
	&nbsp;	&#160;	nem törhető szóköz
	&shy;	&#173;	elválasztó jel
§	&sect;	&#167;	
©	&copy;	&#169;	

Karakter	Kódnév	Szám kód	
®	&reg;	&#174;	
™	&trade;	&#8482;	
€	&euro;	&#8364;	
¢	&cent;	&#162;	
£	&pound;	&#163;	
¥	&yen;	&#165;	
°	&deg;	&#176;	
×	&times;	&#215;	
¼	&frac14;	&#188;	
½	&frac12;	&#189;	
¾	&frac34;	&#190;	
∞	&infin;	&#8734;	
≠	&ne;	&#8800;	
≤	&le;	&#8804;	
≥	&ge;	&#8805;	
±	&plusmn;	&#177;	
÷	&divide;	&#247;	
μ	&micro;	&#181;	
f	&fnof;	&#402;	
∏	&prod;	&#8719;	
Σ	&sum;	&#8721;	
✱	&lowast;	&#8727;	
√	&radic;	&#8730;	
∫	&int;	&#8747;	
≡	&equiv;	&#8801;	
¹	&sup1;	&#185;	
²	&sup2;	&#178;	
³	&sup3;	&#179;	
¡	&brvbar;	&#166;	
«	&laquo;	&#171;	
´	&acute;	&#180;	
·	&middot;	&#183;	
¨	&uml;	&#168;	
»	&raquo;	&#187;	
¶	&para;	&#182;	
...	&hellip;	&#8230;	
←	&larr;	&#8592;	
↑	&uarr;	&#8593;	
→	&rarr;	&#8594;	
↓	&darr;	&#8595;	
↔	&harr;	&#8596;	
⇐	&lArr;	&#8656;	
⇑	&uArr;	&#8657;	
⇒	&rArr;	&#8658;	
⇓	&dArr;	&#8659;	
⇔	&hArr;	&#8660;	
#	&num;	&#35;	

Még több:

- <http://dev.w3.org/html5/html-author/charref> [http://dev.w3.org/html5/html-author/charref]
- <http://www-archive.mozilla.org/newlayout/testcases/layout/entities.html> [http://www-archive.mozilla.org/newlayout/testcases/layout/entities.html]

## Rajzvászon

### Vászon

```
<canvas id="vaszon1" width="200" height="100" style="border:1px solid #000000;">
A böngésződ nem támogatja a HTML5 rajzvászont.
</canvas>
```



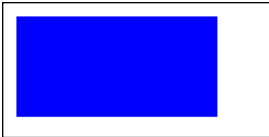
### Rajzolás JavaScripttel

```
<canvas id="vaszon2" width="200" height="100" style="border:1px solid #000000;">
A böngésződ nem támogatja a HTML5 rajzvásznont.
</canvas>
```

```
<script>

var c=document.getElementById("vaszon2");
var ctx=c.getContext("2d");
ctx.fillStyle="#0000ff";
ctx.fillRect(10,10,150,75);

</script>
```



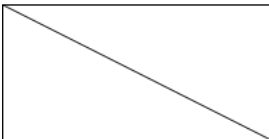
### Vonalhúzás

```
<canvas id="vaszon3" width="200" height="100" style="border:1px solid #000000;">
A böngésződ nem támogatja a HTML5 rajzvásznont.
</canvas>
```

```
<script>

var c=document.getElementById("vaszon3");
var ctx=c.getContext("2d");
ctx.moveTo(0,0);
ctx.lineTo(300,150);
ctx.stroke();

</script>
```

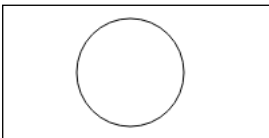


### Kör

```
<canvas id="vaszon4" width="200" height="100" style="border:1px solid #000000;">
A böngésződ nem támogatja a HTML5 rajzvásznont.
</canvas>
```

```
<script>
var c=document.getElementById("vaszon4");
var ctx=c.getContext("2d");
ctx.beginPath();
ctx.arc(95,50,40,0,2*Math.PI);
ctx.stroke();

</script>
```



### Több rajz

```
<canvas id="vaszon" width="600" height="220"></canvas>
```

```
<script>
var v = document.getElementById("vaszon").getContext("2d");

v.fillStyle="rgba(0,0,123,1.0)";
v.fillRect(50, 10, 150, 50);

v.strokeStyle="rgba(0,0,123,1.0)";
v.strokeRect(50, 80, 240, 120);

v.beginPath();
v.arc(450, 110, 100, Math.PI * 2/2, Math.PI * 4/2);
v.lineWidth = 10;
v.lineCap = 'round';
v.strokeStyle = 'rgba(255, 127, 0, 1.0)';
v.stroke();

</script>
```

### Szöveg

```
<canvas id="vaszon5" width="200" height="100" style="border:1px solid #000000;">
A böngésződ nem támogatja a HTML5 rajzvásznont.
</canvas>
```

```
<script>
var c=document.getElementById("vaszon5");
var ctx=c.getContext("2d");
ctx.font="30px Arial";
ctx.fillText("Helló Világ",10,50);

</script>
```

A rectangular canvas with a black border containing the text "Helló Világ" in a large, black, sans-serif font, centered horizontally and vertically.

### Szöveg 2

```
<canvas id="vaszon6" width="200" height="100" style="border:1px solid #000000;">
A böngésződ nem támogatja a HTML5 rajzvásznont.
</canvas>
```

```
<script>
var c=document.getElementById("vaszon6");
var ctx=c.getContext("2d");
ctx.font="30px Arial";
ctx.strokeText("Hello World",10,50);

</script>
```

A rectangular canvas with a black border containing the text "Hello World" in a large, black, sans-serif font, centered horizontally and vertically.

### Színátmenet

```
<canvas id="vaszon7" width="200" height="100" style="border:1px solid #000000;">
A böngésződ nem támogatja a HTML5 rajzvásznont.
</canvas>
```

```
<script>
var c=document.getElementById("vaszon7");
var ctx=c.getContext("2d");

// Átmenet
var grd=ctx.createLinearGradient(0,0,200,0);
grd.addColorStop(0,"blue");
grd.addColorStop(1,"white");

// Kitöltés
ctx.fillStyle=grd;
ctx.fillRect(10,10,150,80);

</script>
```



### Színátmenet 2

```
<canvas id="vaszon8" width="200" height="100" style="border:1px solid #000000;">
A böngésződ nem támogatja a HTML5 rajzvásznont.
</canvas>
```

```
<script>

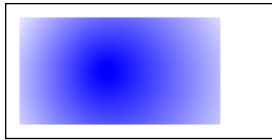
var c=document.getElementById("vaszon8");
var ctx=c.getContext("2d");

// Átmenet
```

```
var grd=ctx.createRadialGradient(75,50,5,90,60,100);
grd.addColorStop(0,"blue");
grd.addColorStop(1,"white");
```

```
// Kitöltés
ctx.fillStyle=grd;
ctx.fillRect(10,10,150,80);
```

```
</script>
```



## Kép

```
<p>A kép használata:</p>
```

```

```

```
<p>Vászon:</p>
```

```
<canvas id="vaszon9" width="250" height="300" style="border:1px solid #d3d3d3;">
A böngésződ nem támogatja a HTML5 rajzvásznont.
</canvas>
```

```
<script>
```

```
var c=document.getElementById("vaszon9");
var ctx=c.getContext("2d");
var img=document.getElementById("sikoly");
ctx.drawImage(img,10,10);
```

```
</script>
```

A kép használata:



Vászon:



## Más példa

```
<!DOCTYPE html>
<html>
<head>
<meta charset="utf-8" />
<title></title>
</head>
<body>
```

```
<canvas id="canvas1" width="200" height="100" style="border:1px solid #000000;"></canvas>
```

```
<script>
var can=document.getElementById("canvas1");
var ctx=can.getContext("2d");
ctx.fillStyle="#00FF00";
ctx.fillRect(30,30,50,50);
</script>
</body>
</html>
```

```
<canvas id="canvas1" width="200" height="100" style="border:1px solid #000000;"></canvas>
```

```
<script>
var can=document.getElementById("canvas1");
var ctx=can.getContext("2d");

var grd=ctx.createLinearGradient(0,0,200,0);
grd.addColorStop(0,"blue");
grd.addColorStop(1,"white");

ctx.fillStyle=grd;
ctx.fillRect(30,30,100,50);
</script>
```



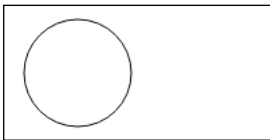
```
var can=document.getElementById("canvas1");
var ctx=can.getContext("2d");

ctx.moveTo(0,0);
ctx.lineTo(200,100);
ctx.stroke();
```

A moveTo() -től a.lineTo()-ig húzunk vonalat.

```
//arc(x, y, r, start, stop)
var can=document.getElementById("canvas1");
var ctx=can.getContext("2d");

ctx.beginPath();
ctx.arc(55, 50, 40, 0, 2 * Math.PI);
ctx.stroke();
```



## Viewport

Ahhoz, hogy telefonon is megfelelő méretben jelenjen meg a weblapunk, állítsuk be a viewportot.

```
<meta name="viewport" content="width=device-width, initial-scale=1.0">
```

## Gyakorlatok

- Keressen olyan HTML elemeket, az alábbi oldalon, amik nem szerepelnek ezen az oldalon:
  - <https://www.w3schools.com/tags/> [<https://www.w3schools.com/tags/>] (2020)

## Szerkesztők

WYSIWYG szerkesztők

- Kompozer [<https://en.wikipedia.org/wiki/Kompozer>]
- FrontPage [[https://en.wikipedia.org/wiki/Microsoft\\_FrontPage](https://en.wikipedia.org/wiki/Microsoft_FrontPage)]

Szerkesztők

- Bluefish
- Notepad2
- Scite
- Screem [<http://www.screem.org>]

## Függelék

### HTML5 sablon

```
<!DOCTYPE html>
<html lang="hu">
<head>

<meta charset="utf-8">
<title>Névtelen</title>

<!--[if lt IE 9]>
<script src="http://html5shim.googlecode.com/svn/trunk/html5.js"></script>
<![endif]-->

</head>
<body>

</body>
</html>
```

### PDF beépítése

```
<object data="azenallomanyom.pdf" type="application/pdf" width="100%" height="100%">

<p>A böngésző nem képes megjeleníteni a PDF dokumentumot.</p>

</object>
```

## Szemantikus elemek

A szemantikus elemek egyértelműen megmondják a böngésző és a fejlesztő számára is, hogy mit tartalmaznak.

Szemantikus elemek:

<form>, <table>, <article>.

A nem szemantikus elemek:

<div>, <span>

Tartalmuk akármi lehet.

Néhány szemantikus elem:

- <table>
- <form>
- <section>
- <article>
- <nav>
- <header>
- <footer>
- <figure>
- <figcaption>
- <details>
- <main>
- <mark>
- <summary>
- <time>

## Linkek és Irodalom

- Richard Clark, Oli Studholme, Divya Manian, Christopher Murphy: **Beginning HTML5 and CSS3**
  - <http://thewebevolved.com> [<http://thewebevolved.com>]
- Bruce Lawson and Remy Sharp: **INTRODUCING HTML5**
  - <http://introducinghtml5.com/> [<http://introducinghtml5.com/>]
- Mark Pilgrim: **DIVE INTO HTML5**
  - <http://diveintohtml5.info> [<http://diveintohtml5.info>]
- Bruce Lawson, Remy Sharp: **Bemutakozik a HTML 5**
  - [http://www.libri.hu/konyv/bruce\\_lawson.bemutakozik-a-html-5.html](http://www.libri.hu/konyv/bruce_lawson.bemutakozik-a-html-5.html) [[http://www.libri.hu/konyv/bruce\\_lawson.bemutakozik-a-html-5.html](http://www.libri.hu/konyv/bruce_lawson.bemutakozik-a-html-5.html)]
  - <http://www.kiskapu.hu/index.php?BODY=BookInfo&OP=details&ID=120344> [<http://www.kiskapu.hu/index.php?BODY=BookInfo&OP=details&ID=120344>]
- <http://www.whatwg.org/specs/web-apps/current-work/multipage/> [<http://www.whatwg.org/specs/web-apps/current-work/multipage/>] (A HTML5)
- <http://www.w3.org/TR/2012/CR-html5-20121217/> [<http://www.w3.org/TR/2012/CR-html5-20121217/>] (HTML5)
- <http://www.w3.org/TR/2012/CR-2dcontext-20121217/> [<http://www.w3.org/TR/2012/CR-2dcontext-20121217/>] (Canvas 2D)
- <http://www.whatwg.org/specs/web-apps/current-work/multipage/the-canvas-element.html> [<http://www.whatwg.org/specs/web-apps/current-work/multipage/the-canvas-element.html>] (Canvas)
- <http://www.sgmlsource.com/> [<http://www.sgmlsource.com/>] (SGML történet)
- <http://en.wikipedia.org/wiki/Favicon> [<http://en.wikipedia.org/wiki/Favicon>] (A favikon)

Hasznos:

- <http://modernizr.com/> [<http://modernizr.com/>] (JavaScript könyvtárak)