

HTML programozás az alapoktól

Írta: Vári Csaba

2007. március

1. Mi az a HTML dokumentum?

A HTML dokumentum egy standard szövegfájl .html vagy .htm kiterjesztéssel, mely speciális formázó utasításokat tartalmaz. (HTML = Hyper Text Markup Language)

Legegyszerűbben a Jegyzetömbbel hozhatjuk létre, szerkeszthetjük és menthetjük el, de ajánlott valamilyen HTML szerkesztő program, például az EditPlus 2 használata. Megjelenítésére leggyakrabban az Internet Explorert, a Netscape-et, vagy a Firefox-ot használjuk de a standard HTML dokumentumot sok más böngésző is értelmezni tudja.

2. Mi az a „TAG”?

A „TAG” (ejtsd: teg, a továbbiakban egyszerűen tag) a HTML dokumentumok formázó utasítása. Minden esetben a „kisebb” és a „nagyobb” relációs jelek közé kell tenni:

```
<TAG>
```

Minden tag-nek van nyitó és záró része. A záró részben az azonosító elé egy „/” jelet kell tenni:

```
</TAG>
```

A HTML 4.01-ben tag-eket írhatjuk kis és nagy betűvel is a böngésző ugyanúgy kezeli majd mindkettőt. A tag nyitó részébe az azonosító után megadhatunk úgynevezett paramétereket is. Ezek a paraméterek a tag-re vonatkozó tulajdonságokat állítják be. A paraméter neve és az egyenlőségjel után a paraméter értékét dupla idézőjelek közé kell tenni. A paramétereket egy szóközzel választjuk el egymástól. A könnyebb áttekinthetőség érdekében a legtöbb HTML szerkesztő oldalhoz hasonló színeket használunk a HTML kódoknál.

```
<TAG paraméter="érték" paraméter="érték">
```

A **style** paraméter egy olyan speciális paraméter, melyben további paramétereket adhatunk meg. A **style** értékében a paraméterek és értékük között csak egy kettőspontot kell rakni és az értéket a következő paramétertől egy pontosvesszővel és szóközzel kell elválasztani. A **style**-al tulajdonképpen a CSS -t használjuk, de erre most nem fogunk kitérni.

```
<TAG paraméter="érték" style="paraméter:érték; paraméter:érték">
```

A tag nyitó és záró része közé kerül az a rész, amelyre a tag vonatkozik. Ha például egy szöveget kék színnel és 5-ös betűnagysággal szeretnénk kiírni:

```
<BETŰ szín="kék" betűnagyság="5">Ide jön s szöveg...</BETŰ>
```

3. A HTML dokumentum alapszerkezete

Minden HTML oldal a következőképpen épül fel:

```
<HTML>
  <HEAD>
    <TITLE> [ ..... ] </TITLE>
  </HEAD>
  <BODY>
    [ ..... ]
    [ ..... ]
    [ ..... ]
  </BODY>
</HTML>
```

Jól látható, hogy minden nyitó tag-nek megvan a záró párja, és hogy mindig azt a tag-et zárjuk le először, amelyiket utoljára megnyitottuk. A HTML oldalak mind ilyen keretes szerkezetűek! Néhány tag esetében a záró tag-et elhagyhatjuk, de erre majd mindig külön felhívom a figyelmet!

Most vegyük sorra ezt a 4 alapvető tag-et:

- <HTML>** : Azt jelzi, hogy a közte lévő szöveget a HTML szabvány szerint kell értelmeznie a böngészőnek. Ennek a tag-nek nincsenek paraméterei, és mindig kötelező lezárni.
- <HEAD>** : Ez a HTML oldal fejrésze. Sok hasznos információt elhelyezhetünk benne, de egyelőre még csak a következő tag-et tesszük bele:
- <TITLE>** : A HTML oldal címe, vagy ha úgy tetszik fejléce, ami a böngésző fejlécében jelenik meg. Paraméterei nincsenek, és mindig kötelező lezárni!
- <BODY>** : A HTML oldal törzse. A nyitó és záró tag közé beírt szöveget a böngésző megpróbálja értelmezni, majd megjeleníteni. Mindig le kell zárni.

4. Megjegyzések beszúrása

A megjegyzés tag-et olyan szövegek elhelyezésére használjuk, melyet böngésző nem veszi figyelembe, nem jelenít meg csak a szerkesztés közben lehet rá szükségünk:

```
<!-- Ide jön a megjegyzés szövege -->
```

5. Az oldalunk specifikációjának megadása

Az oldalunk specifikációja az első dolog amit HTML szerkesztésnél megadunk. Ezzel mondjuk meg a böngészőnek, hogy melyik HTML specifikációt használja a dokumentum. Három fajtája van:

HTML Strict DTD : Szigorú szabványkövetéshez használatos DTD (dokumentumtípus definíció - Document Type Definition). Megjelenítési hibák nem megengedettek. Használható stílus meghatározásokkal (Cascading Style Sheets-CSS).

HTML Transitional DTD : A 'Transitional DTD'-vel meghatározott weboldal tartalmazhat olyan paramétereket és tag-eket, amelyek a W3C szerint stílusmeghatározásokkal helyettesítendők.

HTML Frameset DTD : A Frameset DTD használandó keretes (frames) weboldalak esetén. A 'Frameset DTD' azonos a 'Transitional DTD'-vel, de itt a <body> tag-et a <frameset> tag helyettesíti.

A HTML Strict DTD :

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

A HTML Transitional DTD :

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

A HTML Frameset DTD :

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

6. A háttér és a színek beállítása

6.1 Háttér beállítása

Az oldal háttérét és háttérszínét a <BODY> tag-ben lehet paraméterként megadni. Ha háttérképet is használunk, akkor ajánlott a háttérszínét a képhez hasonlóra beállítani, így biztos nem lesz soha olvashatósági problémánk (például, ha ki van kapcsolva a képek letöltése a böngészőben). A háttér színét meg lehet adni a hexa kódjával (pl. #FF01DC) vagy ha van a rövid angol nevével (pl. crimson):

```
<BODY bgcolor="#FF01DC">
```

```
<BODY bgcolor="crimson">
```

Ajánlott olyan háttérképet megadni, ami nem túlságosan tarka, így a szöveg mindenhol jól olvasható lesz. A háttérképet relatív (pl.: *images/hatter.jpg*) és abszolút (pl.: *http://valami.hu/konyvtar/hatter.jpg*) linkkel is megadhatjuk:

```
<BODY bgcolor="#FF01DC" background="hatter.jpg">
```

A háttérkép tulajdonságait a **style** paraméterrel módosíthatjuk:

```
<BODY background="hatter.jpg" style="background-repeat:no-repeat;
background-attachment:fixed">
```

A **style**-ba a következő paraméterek kerülhetnek:

- background-position** : A háttérkép pozícióját határozhatjuk meg vele.
left = balra, **center** = középre, **right** = jobbra
- background-repeat** : Ha kisebb a kép mint a képernyőnk, akkor mozaikszerűen beilleszti-e a képet. A **no-repeat** értékre csak egyszer illeszti be a képet.
- background-attachment** : A háttérkép mozgatása. Ha értéként **fixed**-et írunk a háttérképünk az oldal görgetésénél sem fog elmozdulni.

6.2 A szövegek színének beállítása

Az oldal színeit is a <BODY> tag-be írt paraméterekkel tudjuk beállítani:

text="XX" : Az oldalon lévő szövegek alapértelmezett színének megadása.

link="XX" : Az linkek alapértelmezett színének megadása.

vlink="XX" : A már felkeresett linkek színének megadása.

alink="XX" : Az aktív hivatkozások színének megadása.

7. Háttérzene beállítása

Ha meg szeretnénk adni az oldalunk háttérzenéjét, azt a <HEAD> részben tehetjük meg a <BGSOUND> tag-el. Paraméterként kell beírni a zene elérési útvonalát:

```
<HEAD>  
  <BGSOUND src="hatterzene.wav" loop="-1">  
</HEAD>
```

Ha a loop értéke -1, akkor a zeneszámot a végtelenségig fogja ismételtetni. Ha valamilyen 0-nál nagyobb számot írunk be, akkor a megadott számszor fogja megismételni a zenét.

8. Margók beállítása

A margókat a háttérhez hasonlóan a <BODY> tagban tudjuk a paraméterekkel beállítani:

```
<BODY topmargin="10" leftmargin="10">
```

A margók beállításánál használható paraméterek:

`topmargin="XX"` : A felső és az alsó margó mérete pixelben.

`leftmargin="XX"` : A jobb és a bal oldali margó mérete pixelben.

`marginheight="XX"` : Ugyanaz, mintha a `topmargin`-t használtuk volna.

`marginwidth="XX"` : Ugyanaz, mintha a `leftmargin`-t használtuk volna.

9. Bekezdés, sortörés és szököz

9.1 Bekezdések létrehozása

Bekezdéseket a **<P>** (paragraph) tag-el tudunk létrehozni. Mindig le kell zárni! Paraméterként azt szoktuk megadni, hogy a bekezdésen belül balra, középre vagy jobbra legyen igazítva a szöveg (**left** = balra, **center** = középre, **right** = jobbra, **justify** = sorkizárt).

```
<P align="center">A szöveg helye</P>
```

A bekezdéseinket a már korábban megismert **style** paraméterrel is formázhatjuk:

```
<P align="center" style="margin-left:10; line-height:150%">  
A szöveg helye</P>
```

A **style**-ban a következő paramétereket adhatjuk meg:

- text-align:XX** : A bekezdés vízszintes igazítását adhatjuk meg. (**left**, **right**, **center**, **justify**)
- margin-left:XX** : Behúzás a szöveg előtt. Pixelben kell megadni.
- margin-right:XX** : Behúzás a szöveg után. Pixelben kell megadni.
- margin-top:XX** : Térköz a szöveg fölött. Pixelben kell megadni.
- margin-bottom:XX** : Térköz a szöveg alatt. Pixelben kell megadni.
- text-indent:XX** : Az első sor behúzása. Pixelben kell megadni.
- line-height:XX%** : Sortávolság. Százalékban kell megadni, az egyszeres sortávolság a 100%.

9.2 Sortörés és mesterséges szököz

A HTML dokumentumban lévő, egymást követő szöközöket a böngésző egyetlen szököznek fogja fel. Ugyanez a helyzet a sortöréssel is: hiába kezdünk új sort a szövegszerkesztőben, a böngésző csupán egy szóközt fog beilleszteni a sortörés helyére.

Ahhoz tehát, hogy igazi sortörést illetve sok egymást követő szóközt helyezünk el a dokumentumban, speciális vezérlő elemekre van szükségünk:

**
** : Sortörés beszúrása. Nem kell lezárni!

** ** : Egy mesterséges szököz beszúrása, melyet már megjelenít a böngésző.

10. A betűtípusok módosítása

A szövegek betűtípusát úgy tudjuk módosítani, hogy a szöveget a **** tag közé rakjuk. Le kell zárni.

```
<FONT size="5" color="navy" face="arial">A szöveg helye</FONT>
```

A **** tag-be írható paraméterek:

size="X" : A szöveg szabványméretét adhatjuk meg 1-7 -ig. 1 = 8 pt, 2 = 10pt, 3 = 12 pt, 4 = 14 pt, 5 = 18 pt, 6 = 24 pt, 7 = 36 pt

color="XX" : A szöveg színét adhatjuk meg a hexa kódjával (pl: #FF01DC) vagy rövid angol nevével (pl: crimson).

face="XX" : A szöveg betűtípusát adhatjuk meg (pl. arial, vagy courier new). Csak alap betűtípusokat adjunk meg, ami minden számítógépen megtalálható.

A betűtípust a **** tag és a **style** paraméter segítségével is formázhatjuk:

```
<SPAN style="font-variant:small-caps">A szöveg helye</SPAN>
```

A **style**-ban megadható paraméterek:

text-decoration:XX : Értékként megadhatjuk: **overline** = fölhúzott
blink = villogó

font-variant:small-caps : Kiskapitális.

font-transform:XX : Értékként megadhatjuk: **uppercase** = nagybetűs
capitalize = nagy kezdőbetűs

visibility:hidden : Rejtett.

letter-spacing:XX : A szöveg betűközét adhatjuk meg pt-ban.
Pl. a 2 pt az kétszeresre sűrített, a -2 pt a kétszeresre ritkített.

Léteznek úgynevezett szövegformázó tag-ek, melyek közé kell írni a szöveget. Mindig le kell zárni őket!

Szövegformázó tag-ek:

	Félkövér	<VAR>	Változó
<I>	Dőlt	<KBD>	Billentyűzet
<U>	Aláhúzott		Áthúzott
	Kiemelt	<SAMP>	Minta
<SUB>	Alsó index	<CODE>	Kód (Courier New font)
<SUP>	Felső index	<CITE>	Idézet, aláírás
<DFN>	Definíció	<STRIKE>	Ugyanaz mint a

11. Fejlécek

Az előző fejezetekben már megtudtuk, hogy lehet nagy és vastag betűkkel fejléceket készíteni a WEB-oldalaink számára. Létezik azonban egy egyszerűbb megoldás is, a szabványos fejlécek használata. Ehhez a `<H1>`. . . `<H6>` tag-eket kell használni, melyek közül a H1 a legnagyobb, és a H6 a legkisebb:

```
<H2>Üdvözöllek az oldalamon!</H2>
```

A fejléceket az **align** paraméterrel formázhatjuk (**left** = bal, **center** = középre, **right** = jobbra, **justify** = sorkizárt):

```
<H4 align="center">Ez egy középre igazított fejléc</H4>
```

12. Listák

A HTML-ben háromféle listát hozhatunk létre. A rendezett listát, a rendezetlen listát, és a definíciós listát.

12.1 A rendezett lista:

Az első a rendezett, más néven számozott lista, angolul "ordered list". Innen ered az tag neve is: ``. A számozott lista sorait az `` tag után, külön sorokba kell írunk:

```
<OL>
  <LI>A lista 1. pontja
  <LI>A lista 2. pontja
  <LI>A lista 3. pontja
</OL>
```

Az `` tag-ben a következő paramétereket adhatjuk meg:

`type="X"` : A számozás stílusát állíthatjuk be. Megadható értékek:

- 1** = számokkal
- A** = nagy betűkkel
- a** = kis betűkkel
- I** = nagy római számokkal
- i** = kis római számokkal

`start="XX"` : A kezdő sorszámot adhatjuk meg vele.

Az egyes sorok újabb listákat is tartalmazhatnak, tehát a listákat egymásba lehet ágyazni. Ekkor érdemes az egyes szintekhez különböző típusokat rendelni:

```
<OL type="I">
  <LI>A lista 1. pontja
    <OL type="1">
      <LI>A lista 1. alpontja
      <LI>A lista 2. alpontja
      <LI>A lista 3. alpontja
    </OL>
  <LI>A lista 2. pontja
  <LI>A lista 3. pontja
</OL>
```

12.2 Rendezetlen lista:

A listák második típusa a rendezetlen, más néven felsorolt lista, angolul "unordered list", amiből az **** tag neve is ered. Felépítése hasonló a rendezett listához, csak annyiban különbözik tőle, hogy itt az egymásba ágyazott szintek automatikusan más jelölést kapnak (teli kör, üres kör, teli négyzet):

```
<UL>
  <LI>A lista 1. pontja
    <UL>
      <LI>A lista 1. alpontja
      <LI>A lista 2. alpontja
      <LI>A lista 3. alpontja
        <UL>
          <LI>A lista 3. alpontjának 1. alpontja
          <LI>A lista 3. alpontjának 2. alpontja
        </UL>
      </UL>
    </UL>
  <LI>A lista 2. pontja
  <LI>A lista 3. pontja
</UL>
```

Az **** tag-be megadható paraméterek:

- type="X"** : A számozás stílusát állíthatjuk be. Megadható paraméterek: **circle** = üres kör
disc = teli kör
square = négyzet

12.3 Definíciós lista:

A listákhoz tartozik ugyan, de kicsit eltér az előző típusoktól az ún. definíciós lista, angolul "definition list". A tag-je: **<DL>**. A listán belül egy "szakasz" mindig két részből áll: a meghatározásból ("definition term", azaz **<DT>**) és a leírásból ("definition description", azaz **<DD>**). Ezt a típust akkor használjuk, ha kifejezéseket szeretnénk megmagyarázni, vagy egy-egy rövidebb szöveghez hosszabb megjegyzést kívánunk fűzni. Szerkezete a következő:

```
<DL>  
  <DT>1. Fogalom  
    <DD>Az 1. fogalom meghatározása.  
  <DT>2. fogalom  
    <DD>A 2. fogalom meghatározása.  
</DL>
```

13. Képek

A kép beszúrásához be kell írni az **** tag-et (nem kell lezárni) és benne paraméterként meg kell adni a kép elérési útvonalát, ami lehet abszolút és relatív is:

```
<IMG src="alkonyvtar/kepneve.jpg">
```

```
<IMG src="http://www.valami.hu/kepneve.jpg">
```

Az **** tag-ben megadható paraméterek:

border="XX" : A kép keretének vastagságát adhatjuk meg pixelben.

hspace="XX" : A vízszintes térközt adhatjuk meg pixelben.

vspace="XX" : A függőleges térközt adhatjuk meg pixelben.

alt="XX" : Egy alternatív szöveget adhatunk meg, ami akkor jelenik meg, ha nem jeleníthető meg a kép, vagy ha a kurzort a kép fölé tartjuk.

align="XX" : A kép vízszintes igazítását adhatjuk meg. **left** = balra
right = jobbra
bottom = az alapvonalra
middle = a kép közepe lesz az alapvonalra illesztve
top = a kép teteje a szöveg tetejével lesz egy vonalban

width="XX" : A kép szélességét adhatjuk meg pixelben.

height="XX" : A kép magasságát adhatjuk meg pixelben.

A képet **style** paraméterrel is formázhatjuk. Megadható paraméterek:

border-color:XX : A kép keretének színét állíthatjuk be a szokásos módokon.

float:XX : A körbefuttatás irányát adhatjuk meg: **none** = nincs körbefuttatás
left = a kép bal oldalra kerül és a szöveg jobbról folyja körbe
right = a kép jobb oldalra kerül és a szöveg balról folyja körbe

vertical-align:XX : A függőleges igazítást adhatjuk meg: **baseline** = az alapvonalhoz igazítás
middle = a kép közepe lesz az alapvonalra illesztve
top = a kép teteje a szöveg tetejével lesz egy vonalban
bottom = az alapvonalra

- `margin-left:XX` : A kép bal oldali margóját adhatjuk meg pixelben
- `margin-right:XX` : A kép jobb oldali margóját adhatjuk meg pixelben
- `margin-top:XX` : A kép felső margóját adhatjuk meg pixelben
- `margin-bottom:XX` : A kép alsó margóját adhatjuk meg pixelben
- `border-style:XX` : A keret fajtáját adhatjuk meg: **dotted** = pontvonal
dashed = szaggatott
solid = folytonos
double = dupla
groove = bevágott
ridge = kiálló
inset = bemélyített
outset = kidomborodó

14. Vízszintes elválasztóvonal

Vízszintes vonalat a `<HR>` tag-el lehet létrehozni. Nem kell lezárni:

```
<HR color="blue" size="10" width="50%" align="right">
```

A `<HR>` tag-be írt paraméterekkel tudjuk változtatni a vízszintes vonal tulajdonságait.

A megadható paraméterek:

- `color="XX"` : A vonal színét adhatjuk meg.
- `width="XX"` : A vonal szélességét adhatjuk meg pixelben vagy százalékban.
- `size="XX"` : A vonal vastagságát adhatjuk meg pixelben.
- `align="XX"` : A vonal vízszintes igazítását adhatjuk meg. (**left** = balra, **center** = középre, **right** = jobbra)
- `noshade` : Ezzel megszüntethetjük a vonal árnyékolását.

15. Hivatkozások és könyvjelzők

15.1 Hivatkozások

Linkeket az `<A>` tag segítségével hozhatunk létre. (Mindig le kell zárni!). Az `<A>` tag-ben lévő `href` paraméterrel tudjuk megadni a hivatkozás célját. Ez lehet abszolút vagy relatív hivatkozás is:

```
<A href="http://www.valami.hu.hu/céoldal.htm">Link szövege.</A>
```

```
<A href="oldalak/céoldal.htm">Link szövege.</A>
```

A hivatkozás alapértelmezetten ugyanabban az ablakban nyílik meg, mint amiben van. Ezen a `target` paraméterrel változtathatunk:

```
<A target="_blank" href="http://www.valami.hu">Link szövege.</A>
```

A `target`-ben megadható paraméterek:

`_self` : Azonos keret.

`_top` : Teljes ablak.

`_blank` : Új ablak.

`_parent` : Szülőkeret.

15.2 Könyvjelzők:

Ha a weboldalunkon van egy nagyon hosszú szöveg, akkor annak ajánlott az elejére linkeket rakni, amik a szöveg bekezdéseire mutatnak. Ezt a könyvjelzőkkel tehetjük meg. Ehhez először minden bekezdésnek egy azonosítót (`id`) kell adnunk. Ezután a hivatkozásunkban nem egy oldal címét hanem a bekezdés azonosítóját adjuk meg. Ekkor a hivatkozásba írunk egy #-et és a könyvjelző nevét (a bekezdés azonosítóját).

```
<A href="#azonosito_1">1. könyvjelző szövege</A>
```

```
<A href="#azonosito_2">2. könyvjelző szövege</A>
```

```
<A href="#azonosito_3">3. könyvjelző szövege</A>
```

```
<P id="azonosito_1">
```

....

```
</p>
```

```
<p id="azonosito_2">
```

....

```
</p>
```

```
<p id="azonosito_3">
```

....

```
</p>
```

16. Idézetek, blokkok elkülönítése

Az oldalon található szövegeket a **<BLOCKQUOTE>** tag-el különíthetjük el egymástól. A tag-ek közé kerülő szöveg margója a jobb és bal oldalon is nagyobb lesz, és a szöveg alatt és felett is kihagy egy sort a böngésző.

```
<BLOCKQUOTE>
    ...Ide jön a szöveg...
</BLOCKQUOTE>
```

17. Táblázatok

Táblázatokat a **<TABLE>** tag-el tudunk létrehozni. Ezután a **<TR>** tag-el létrehozuk a táblázat sorait, majd a **<TD>** tag-el a sort cellákra osztjuk. Ha egy cellába nem írunk semmit, akkor a keret nem fog rendesen megjelenni. Ezért az üres cellákba mindig tegyünk egy mesterséges szóközt: ** **;

```
<TABLE>
  <TR>
    <TD>Az első sor első cellája</TD>
    <TD>Az első sor második cellája</TD>
  </TR>
  <TR>
    <TD>A második sor első cellája</TD>
    <TD>A második sor második cellája</TD>
  </TR>
</TABLE>
```

A **<table>** tag-ben megadható paraméterek:

<code>align="XX"</code> :	A vonal vízszintes igazítását adhatjuk meg. (left = balra, center = középre, right = jobbra)
<code>border="XX"</code> :	A szegélyek vastagságát adhatjuk meg pixelben.
<code>cellspacing="XX"</code> :	A cellák közötti rés nagysága pixelben.
<code>cellpadding="XX"</code> :	A cellákon belüli margó nagysága pixelben.
<code>width="XX%"</code> :	A táblázat szélességét adhatjuk meg pixelben vagy százalékban.
<code>height="XX%"</code> :	A táblázat magasságát adhatjuk meg pixelben vagy százalékban.
<code>bgcolor="XX"</code> :	A táblázat háttérszínét adhatjuk meg.
<code>background="XX"</code> :	A táblázat háttérképét adhatjuk meg.
<code>bordercolorlight="XX"</code> :	A keret világos részének színét adhatjuk meg.

`bordercolordark="XX"` : A keret sötét részének színét adhatjuk meg.

A `style` paraméterrel további tulajdonságokat adhatunk meg:

`border-width:XX` : A táblázat külső szegélyének a vastagságát adhatjuk meg pixelben.

`border-color:XX` : A táblázat külső szegélyének a színét adhatjuk meg.

`cellpadding:XX` : A cellákon belüli margó nagysága pixelben.

A `<TR>` tag-en belül megadható paraméterek:

`align="XX"` : A sor tartalmának vízszintes igazítása. (**left** = balra, **center** = középre, **right** = jobbra)

`valign="XX"` : A sor tartalmának függőleges igazítása. (**top** = felülre, **middle** = középre, **bottom** = alulra)

`bgcolor="XX"` : A sor háttérszínét adhatjuk meg.

`bordercolor="XX"` : A sor keretének a színét adhatjuk meg.

`bordercolorlight="XX"` : A sor keretének a világos színét adhatjuk meg.

`bordercolordark="XX"` : A sor keretének a sötét színét adhatjuk meg.

A `<TD>` tag-en belül megadható paraméterek:

`align="XX"` : A cella tartalmának vízszintes igazítása. **left** = balra
center = középre
right = jobbra

`valign="XX"` : A cella tartalmának függőleges igazítása. **top** = felülre
middle = középre
bottom = alulra

`width="XX"` : A cella szélességét adhatjuk meg.

`colspan="XX"` : A cella hány cella helyét foglalja el vízszintesen.

`rowspan="XX"` : A cella hány cella helyét foglalja el függőlegesen.

`height="XX"` : A cella magasságát adhatjuk meg.

`background="XX"` : A cella hátterét adhatjuk meg.

`bgcolor="XX"` : A cella háttérszínét adhatjuk meg.

`bordercolor="XX"` : A cella keretének a színét adhatjuk meg.

`bordercolorlight="XX"` : A cella keretének a világos színét adhatjuk meg.

`bordercolordark="XX"` : A cella keretének a sötét színét adhatjuk meg.

18. Keretek

A keretek segítségével egy böngészőablakban több HTML oldalt is megjeleníthetünk. Keretes oldalt kétféleképpen hozhatunk létre, kerettel és beágyazott kerettel.

18.1 Keretes szerkezetű oldalak

Keretes szerkezetű oldalaknál először létre kell hozni egy külön HTML oldalt mely a kereteket tartalmazza. Ebben az oldalban fog megnyílni a többi oldal. A keretes szerkezetű oldalunkat a **<FRAMESET>** tag-el hozhatjuk létre. (Le kell zárni)

A **<FRAMESET>** tag-en belül a következő paramétereket kell megadni:

`cols="XX, XX, *"` : A keretes oldal oszlopainak szélessége. Megadhatjuk pixelben és százalékban is. Az egyik oszlop szélességének megadhatjuk a *-ot, így az a keret kiegészíti az oldal szélességét 100%-ra.

`framespacing="XX"` : A keretek közti távolság pixelben.

`border="X"` : A keretek közt legyen-e kerete. (0 = nem, 1 = igen)

`bordercolor="XX"` : A keret színét adhatjuk meg.

Miután létrehoztuk az oszlopokat, azokat sorokra oszthatjuk, szintén a **<frameset>** tag-el. Ezt a tag-et viszont az oszlop két tag-je közé kell írni!

A **<frameset>** tag-en ezt a paramétert kell megadni:

`rows="XX, XX, *"` : Az oszlopok sorainak a magassága. Megadhatjuk pixelben és százalékban is. Az egyik sor magasságának megadhatjuk a *-ot, így az a keret kiegészíti az oldal magasságát 100%-ra.

Ezután az oszlopokban létrehozott kereteket kell definiálnunk. Ezt a **<FRAME>** tag-el tehetjük meg.

A **<FRAME>** tag-en belül a következő paramétereket kell megadni:

`name="XX"` : A keret nevét adhatjuk meg.

`src="XX"` : A keretben megjelenítendő oldal elérési útvonala.

`frameborder="X"` : A keretek közt legyen-e keret. (0 = nem, 1 = igen)

`scrolling="XX"` : Görgethető legyen-e a keret. (**yes** = igen, **no** = nem, **auto** = csak akkor görgethető, ha nem látszik a teljes oldal)

`marginwidth=XX"` : A keret jobb és bal oldali margója pixelben.

`marginheight=XX"` : A keret fenti és lenti margója pixelben.

A keretes szerkezetű oldalnál a linkek általában egy másik keretben nyílnak meg. Ezt úgy tudjuk beállítani, hogy a hivatkozás **target** paraméterébe értéként a keret nevét adjuk meg.

Példa egy keretes szerkezetű oldalra:

Logó	Fejléc
Menü	Főoldal
	Lábjegyzet

```
<HTML>
<HEAD>
</HEAD>
<FRAMESET cols="150, *" framespacing="0" border="no">
  <frameset rows="50, *">
    <frame name="logo" src="logo.htm" frameborder="1" scrolling="no">
    <frame name="menu" src="menu.htm" frameborder="1" scrolling="no">
  </frameset>
  <frameset rows="50, *, 25">
    <frame name="fejlec" src="fejlec.htm" frameborder="1" scrolling="no">
    <frame name="fooldal" src="fooldal.htm" frameborder="1" scrolling="no">
    <frame name="labjegyzet" src="labj.htm" frameborder="1" scrolling="no">
  </frameset>
</FRAMESET>
</HTML>
```

18.2 Beágyazott keretek

Az előző példában létrehozott oldalt beágyazott keretek segítségével is létrehozhatjuk. Ehhez először a keretek szerkezetéhez hasonló táblázatot kell készíteni a már tanult módon. Miután létrehoztuk a táblázatot a táblázat celláiban el kell készíteni a beágyazott kereteket. Ezt az **<IFRAME>** (inline frame) tag-el tehetjük meg.

Az <IFRAME> tag-be a következő paramétereket kell megadni:

<code>name="XX"</code> :	A beágyazott keret nevét adhatjuk meg.
<code>src="XX"</code> :	A beágyazott keretben megjelenítendő oldal elérési útvonala.
<code>width="XX"</code> :	A beágyazott keret szélessége pixelben vagy százalékban.
<code>height="XX"</code> :	A beágyazott keret magassága pixelben vagy százalékban.
<code>frameborder="X"</code> :	A beágyazott keretnek legyen-e kerete. (0 = nem, 1 = igen)
<code>framespacing="XX"</code> :	A beágyazott keret és a tartalma közti távolság pixelben.
<code>scrolling="XX"</code> :	Görgethető legyen-e a keret. (yes = igen, no = nem, auto = csak akkor görgethető, ha nem látszik a teljes oldal)
<code>marginwidth="XX"</code> :	A beágyazott keret jobb és bal oldali margója pixelben.
<code>marginheight="XX"</code> :	A beágyazott keret fenti és lenti margója pixelben.
<code>align="XX"</code> :	A beágyazott keret igazítása: left = balra, right = jobbra top = felülre texttop = szöveg tetejéhez middle = középen absmiddle = középvonatra baseline = alapvonatra bottom = alulra absbottom = szöveg aljához center = középre

Egy példa beágyazott keretek használatára:

```
<TABLE>
  <TR height="80%">
    <TD align="center" valign="top" rowspan="2">
      <A href="egy.htm" target="fokeret">Menü 1</A>
      <BR>
      <A href="ketto.htm" target="fokeret">Menü 2</A>
    </TD>
    <TD align="center">
      <IFRAME name="fokeret" src="egy.htm" width="100%" height="100%"
        frameborder="0" framespacing="0" border="0" marginheight="0"
        marginwidth="0"></IFRAME>
    </TD>
  </TR>
</TABLE>
```

19. Űrlapok

Az űrlap tartalmazhat látható, és nem látható (rejtett) elemeket. Látható elemek például a szövegbeviteli mezők, legördülő listák, és a gombok, míg a rejtett elemekben olyan információkat tárolhatunk, amelyekre az űrlap feldolgozásakor szükségünk lehet, de nem szeretnénk, ha az űrlapot kitöltő személy tudomására jutna.

Miután az űrlapot kitöltöttük, elküldjük egy feldolgozó programnak. Ez a feldolgozó program egyesével beolvassa az űrlap elemeit és értékeiket, majd elvégzi rajtuk az általunk megadott utasításokat. Például eltárolhatjuk az adatokat egy adatbázisban, bármilyen formában kiírathatjuk az adatokat a képernyőre, vagy akár el is küldhetjük az adatokat egy email címre.

Az űrlapot a **<FORM>** tag-ek közé kell írni. Amit ezen belül írunk, az az űrlap része lesz.

A **<FORM>** tag-ben megadható paraméterek

`action="XX.XXX"` : A feldolgozó oldal neve, ami HTML vagy CGI oldal.

`method="XXX"` : Az űrlap elemeinek átadási módja. (**GET** vagy **POST**)

`name="XX"` : Az űrlap nevét adhatjuk meg.

19.1 Szöveges beviteli mezők és jelszó mezők

Szöveges beviteli mezőt az **<INPUT type="text">** tag-el hozhatunk létre. Nem kell lezárni.

Jelszó mezőt az **<INPUT type="password">** tag-el hozhatunk létre. Ugyanaz mint a szöveges mező, csak a beírt szövegből csak csillagok vagy körök látszanak:

```
<INPUT type="text" name="szoveg1" size="15" value="Alap szöveg">
```

```
<INPUT type="password" name="jelszo" size="15">
```

Megadható paraméterek:

- `size="XX"` : A szöveges beviteli mező hosszát adhatjuk meg.
- `value="XX"` : A szöveges beviteli mező alapértéke.
- `name="XX"` : A beviteli mező neve.
- `maxlength="XX"` : A maximálisan beírható karakterek számát adhatjuk meg.
- `disabled` : A mező érvénytelenítése. Ha a tag-be írjuk, nem lehet írni a mezőbe.

19.2 Rejtett mezők

Rejtett mezőt a `<INPUT type="hidden">` tag-el hozhatunk létre. A `value` paraméterbe írjuk azt az értéket, amira a feldolgozóprogramnak szüksége lesz:

```
<INPUT type="hidden" value="valami@valami.hu">
```

19.3 Kijelölő négyzetek

A kijelölő négyzet az `<INPUT type="checkbox">` tag-el hozhatunk létre. Nem kell lezárni. A négyzetek közül bármelyiket kiválaszthatjuk (akár az összeset is), de üresen is hagyhatjuk őket. Ha több mezőnek ugyanazt a nevet adjuk, a form feldolgozásakor a kiválasztott mezők értékeit vesszővel elválasztva kapjuk vissza. Tehát a mezők értékei a feldolgozás után (ha mindet kiválasztottuk):

```
<INPUT type="checkbox" name="negyzet1" value="negyzet1" checked>
```

Megadható paraméterek:

- `value="XX"` : A kijelölő négyzet értéke.
- `name="XX"` : A kijelölő négyzet neve.
- `checked` : A kijelölő négyzet kiválasztása. Ha a tag-be írjuk, a négyzet alapértelmezetten ki lesz választva.
- `disabled` : A négyzet érvénytelenítése. Ha a tag-be írjuk, nem lehet rákattintani.

19.4 Rádiógombok

A rádiógombok hasonlóak a kijelölőnégyzetekhez, de a `type` paraméterben `radio`-t kell értéként megadni. Ha ugyanazt a nevet adjuk nekik, akkor maximum egyet választhatunk ki közülük! A `checked` paraméter megadásakor itt is alaphelyzetben ki lesz választva az adott gomb. Ha több helyre is beírjuk, hogy `checked`, akkor az utolsó lesz az érvényes!

```
<INPUT type="radio" name="radiogomb1" value="radiogomb1">
```

A megadható paraméterek ugyanazok mint a kijelölő négyzeteknél.

19.5 Gombok

4-féle gombot hozhatunk létre az `input` tag-ben.

1. Egyszerű gomb: Ennek a gombnak alapértelmezetten nincs semmilyen funkciója, majd később JavaScript-tel lesz valami haszna. Az `<INPUT type="button">` tag-el hozhatjuk létre:

```
<INPUT type="button" name="gomb1" value="Gombfelirat">
```

2. Alaphelyzet gomb: Ez a gomb visszaállítja az űrlap minden elemét az alapértelmezett értékére:

```
<INPUT type="reset" name="alaphelyzet" value="Gombfelirat">
```

3. Elküld gomb: Ezzel a gombbal elküldhetjük a már megadott feldolgozó programnak kitöltött űrlapot:

```
<INPUT type="submit" name="gomb1" value="Elküld">
```

4. Feltöltés gomb: Ezzel a gombbal fájlokat tölthetünk fel:

```
<INPUT type="file" name="gomb1" value="Feltöltés">
```

19.6 Legördülő listák

Legördülő listát az `<SELECT>` tag-el lehet létrehozni. (Le kell zárni.)

Megadható paraméterek:

`name="XX"` : A kijelölő legördülő lista neve.

`size="XX"` : Megadhatjuk, hogy egyszerre hány menüelemet lássunk.

`multiple` : Ha a tag-be írjuk, akkor több menüelemet is kiválaszthatunk.

`checked` : A kijelölő négyzet kiválasztása. Ha a tag-be írjuk, a négyzet alapértelmezetten ki lesz választva.

`disabled` : A lista érvénytelenítése. Ha a tag-be írjuk, nem lehet rákattintani.

A lista menüpontjait az `<OPTION>` tag-el adhatjuk meg. (Nem kell lezárni) A tag után kell megadni a menüelemek szövegét:

```
<SELECT name="lista1" size="1">
  <OPTION value="15"> Első menüelem
  <OPTION value="20"> Második menüelem
  <OPTION value="25" selected> Harmadik menüelem
  <OPTION value="30"> Negyedik menüelem
</SELECT>
```

Megadható paraméterek:

`value="XX"` : A menüelem értéke.

`selected` : Ha a tag-be írjuk, akkor több a menüelem lesz alapértelmezetten kiválasztva.

19.7 Szöveges területek

Nagyobb szöveges területet a `<TEXTAREA>` tag-el hozhatunk létre:

```
<TEXTAREA name="szoveges"> Az alap szöveg </TEXTAREA>
```

Vigyázzunk arra, hogy a nyitó és záró rész között a szóközöket és a sortöréseket is figyelembe veszi a böngésző! Az alapértéket ide kell beírunk, és nem a `value` paraméterbe, mint az előző esetekben!

Megadható paraméterek:

`name="XX"` : A szöveges terület neve.

`cols="XX"` : Hány karakter széles legyen a szöveges terület.

`disabled` : A szöveges terület érvénytelenítése. Ha a tag-be írjuk, nem lehet írni bele.

19.8 Űrlap elemek csoportosítása

Az űrlapon található elemeket (beviteli mezők, kijelölő négyzetek, gombok, stb.) időnként nem árt csoportosítani. Ehhez a `<FIELDSET>` tag-et kell használnunk:

```
<FIELDSET>
```

```
Név: <INPUT type="text" name="nev">
```

```
Jelszó: <INPUT type="password" name="jelszo">
```

```
<INPUT type="submit" value="Elküld">
```

```
</FIELDSET>
```

Ennek a keretnek adhatunk egy címkét is, melynek szövegét a `FIELDSET` tag-en belül, a `<LEGEND>` tag nyitó és záró része közé kell írunk. A címkét az `align` paraméterrel igazíthatjuk balra, középre vagy jobbra:

```
<FIELDSET>
```

```
<LEGEND align="center">Írja be nevét és a jelszavát</LEGEND>
```

```
Név: <INPUT type="text" name="nev">
```

```
Jelszó: <INPUT type="password" name="jelszo">
```

```
<INPUT type="submit" value="Elküld">
```

```
</FIELDSET>
```

20. Fényújság létrehozása

Előfordulhat, hogy az oldalunkra el szeretnénk helyezni egy kis reklám-blokkot, amiben egy reklámszöveg ide-oda mozog, esetleg eltűnik az egyik oldalon, és újra megjelenik a másikon. Ilyen fényújságot a `<MARQUEE>` tag-el hozhatunk létre.

```
<MARQUEE width="200" height="30"> A mozgó szöveg</MARQUEE>
```

A nyitó és lezáró rész közé írt szöveg automatikusan mozogni fog, alapesetben jobbról balra haladva. A szövegben kívül képeket is elhelyezhetünk a blokkban!

```
<MARQUEE width="200" height="30">  
  <IMG src="kep.jpg">  
</MARQUEE>
```

A `<MARQUEE>` tag-ben megadható paraméterek:

<code>width="XX"</code> :	A fényújság szélessége pixelben vagy százalékban.
<code>height="XX"</code> :	A fényújság magassága pixelben vagy százalékban.
<code>direction="XX"</code> :	A mozgás iránya: left = balra right = jobbra
<code>behaviour="XX"</code> :	A fényújság viselkedése. Az alapértelmezett a gördülő. slide = beúszó alternate = ide-oda mozgó
<code>scrollldelay="XX"</code> :	A mozgás sebessége ezredmásodpercekben megadva.
<code>scrollamount="XX"</code> :	A mozgás léptéke pixelben megadva.
<code>loop="XX"</code> :	Ha azt szeretnénk hogy ne a végtelenségig mozogjon a szöveg, akkor ezzel megadhatjuk, hogy hányszor ismétlődjön.
<code>bgcolor="XX"</code> :	A fényújság háttérszínét adhatjuk meg.
<code>hspace="XX"</code> :	A jobb és bal oldali margó mérete pixelben.
<code>vspace="XX"</code> :	A fenti és lenti margó mérete pixelben.

21. Beépülő tartalmak

Az HTML weboldalunkon elhelyezhetünk különböző beépülő multimédiás tartalmakat is. Ilyenek például a flash animációk és a videók. Beépülő tartalmat a **<EMBED>** tag-el hozhatunk létre.

21.1 Videók

Videót a már megismert tag-el helyezhetünk el, és majd a tag-ben lévő paraméterekkel állíthatjuk be. Ajánlott AVI és MPEG videókat használni.

```
<EMBED src="video.avi" style="border:2 black dotted"></EMBED>
```

Megadható paraméterek:

- `src="XX.avi"` : A videó elérési útvonalát adhatjuk meg.
- `width="XX"` : A videó szélessége pixelben.
- `height="XX"` : A videó magassága pixelben.
- `autostart="XX"` : Beállíthatjuk, hogy a videó az oldal betöltésekor automatikusan elinduljon-e. (**true** = igen, **false** = nem)
- `controller="XX"` : A videó alatt megjelenjenek-e a kezelő gombok (play, stop stb.) (**true** = igen, **false** = nem)
- `playcount="XX"` : Alapértelmezetten a böngésző a végtelenségig ismételteti a videót. Itt megadhatjuk, hogy hányszor játssza le.
- `hspace="XX"` : A jobb és bal oldali margó mérete pixelben.
- `vspace="XX"` : A jobb és bal oldali margó mérete pixelben.
- `align="XX"` : A vízszintes igazítást adhatjuk meg a már tanult módon.
- `border="XX"` : A videó keretének tulajdonságait adhatjuk meg szóközzel elválasztva.

21.2 Flash animációk

Flash animációt ugyanúgy helyezhetünk el az oldalunkon, mint videót, csak itt az **<EMBED>** tag kiegészül néhány új paraméterrel.

```
<EMBED src="animaco.swf" quality="high" style="border:1"></EMBED>
```

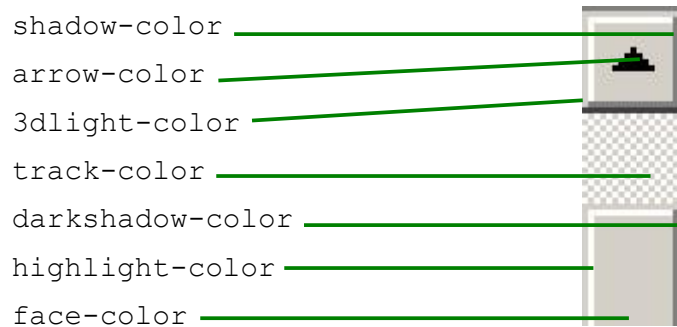
Az előbb tanultakon kívül még megadható paraméterek:

- `quality="XX"` : Az animáció minőségét adhatjuk meg. (**high** = magas, **medium** = közepes, **low** = alacsony)
- `type="XX"` : Az animáció típusát adhatjuk meg. (**application/x-shockwave-flash**)
- `pluginpage="XX"` : Megadhatjuk, hogy az animáció plugin-je honnan tölthető le.

22. A gördítősáv átszínezése

Előfordulhat, hogy a keretes szerkezetű oldalunknál a tartalom kilóg a keretből, és ezért megjelenik a gördítősáv. Nagyon csúnya, ha mondjuk egy világos színekből álló oldalon a gördítősáv sötétszürke. Erre a megoldás a gördítősáv átszínezése, melyet a **<BODY>** tag-be írt **style** paraméterrel tehetünk meg.

```
<BODY style="scrollbar-face-color: #336666; scrollbar-highlight-color: 669999; scrollbar-3dlight-color: #000033; scrollbar-darkshadow-color: #003333; scrollbar-shadow-color: #000000; scrollbar-arrow-color: #FFFFFF; scrollbar-track-color: #EFEFEF;">
```








23. Kurzor beállítása

Minden HTML objektum fölött beállíthatjuk, hogy milyen kurzor jelenjen meg. Ezt a **style**-ban a **cursor** paraméterrel állíthatjuk be.

```
<IMG src="kep.jpg" style="cursor:hand">
```

Megadható kurzorok:

- default** :  Alapértelmezett kurzor
- hand** :  Kéz, általában hivatkozásokat jelöl.
- help** :  Súgó
- crosshair** :  Célkereszt pontos kijelöléshez.
- move** :  Áthelyezés
- text** :  Szöveg kijelölés.
- s-resize** :  Függőleges átméretezés.
- w-resize** :  Vízszintes átméretezés.
- wait** :  Homokóra, várakozást jelent.

24. A HTML fejléce

Egy HTML oldalnak mindig van a **<BODY>**-n kívül fejléce is, amit a **<HEAD>** tag jelöl. A fejléc olyan információkat tartalmaz a dokumentumról, amiket a böngészőnek nem kell megjelenítenie a felhasználó számára.

24.1 A **<BASE>** tag

<BASE> : Meghatározza az oldalon szereplő összes link alap URL-jét.:

```
<HEAD>
  <BASE href="http://www.valami.hu/">
</HEAD>
```

24.2 A **<LINK>** tag

<LINK> : Ez a tag határozza meg az aktuális dokumentum és egy másik dokumentum viszonyát. Az elemet külső dokumentumok, pl. stíluslapok meghívásához használjuk.

```
<HEAD>
  <LINK rel="stylesheet" type="text/css" href="theme.css">
</HEAD>
```

A **<LINK>** tag-ben megadható paraméterek:

charset="XX" : A cél URL karakterkódolását határozza meg.

href="XX" : A cél dokumentum helye.

hreflang="XX" : A céldokumentum nyelvét határozza meg. Például: **hu**

media="XX" : Meghatározza, milyen módon legyen a dokumentum megjelenítve. (**all, braille, print, projection, screen, speech**)

rel="XX" : Meghatározza az aktuális dokumentum és a cél dokumentum viszonyát. **alternate, appendix, bookmark, chapter, contents, copyright, glossary, help, home, index, next, prev, section, start, stylesheet, subsection**

type="XX" : A céldokumentum MIME típusát határozza meg. (Például: **text/css, text/javascript, image/gif**)

24.3 A <TITLE> tag

<TITLE> : Ezzel a tag-el adhatjuk meg a HTML oldalunk címét. Le kell zárni.

```
<HEAD>
  <TITLE> Az HTML oldalunk címe <TITLE>
</HEAD>
```

24.4 A <SCRIPT> tag

<SCRIPT> : Egy scriptet (pl. JavaScript) definiál. Ez lehet más fájlban levő, külső script, vagy a kezdő és záró <script> tag között elhelyezett, beágyazott script.

```
<HEAD>
  <SCRIPT type="text/javascript">
    <!--
    ... a script helye ...
    //-->
  </SCRIPT>
</HEAD>
```

A <SCRIPT> tag-ben megadható paraméterek:

- type="XX"** : A script MIME típusának meghatározása (**text/ecmascript**, **text/javascript**, **text/jscript**, **text/vbscript**, **text/vbs**, **text/xml**).
- charset="XX"** : Meghatározza a script karakterkódolását.
- defer="XX"** : Jelzi, hogy a script nem generál megjelenítendő tartalmat. A böngésző folytathatja a HTML kód értelmezését és felépítheti az oldalt.
- src="XX"** : Megadhatjuk azt a külső fájlt, amelyik a scriptet tartalmazza.

24.5 A <META> tag

<META> : Ez a tag információkat tartalmaz a dokumentumról a böngésző beállításához, keresők számára és egyéb adatokat.

```
<HEAD>
```

```
<META name="DESCRIPTION" lang="HU" content="Az oldal leírása">
```

```
</HEAD>
```

A <META> tag-ek:

1. `<META name="KEYWORDS" lang="HU" content="XX">`

Az oldal tartalmára jellemző kulcsszavak, amiket a keresőkbe beírva a kereső oldalunkat is meg kell találja. Maximális hossza: 200/1000 karakter

2. `<META name="DESCRIPTION" lang="HU" content="XX">`

Az oldal tartalmának leírása röviden. Gyakran ezt a szöveget jelenítik meg a keresők a találati oldalon oldalad leírásaként. Maximális hossza: 200-256 karakter. Tanácsos a szövegbe bekombinálni a legfontosabb keresőszavakat is.

3. `<META http-equiv="Content-Type" content="text/html; charset=iso-8859-2">`

Ezzel a tag-el lehet beállítani, milyen karakterkészletet használjon a kliensgép böngészője az oldal megjelenítésekor. Ha az adott karakterkészlet nincs a gépen telepítve, az alapértelmezett karakterkészletét fogja használni. Csak egy karakterkészletet lehet kijelölni.

A leggyakrabban használt karakterkészletek:

iso-8859-1 = nyugat-európai [iso]

iso-8859-2 = közép-európai [iso]

windows-1250 = közép-európai [windows] (Nem ajánlott!)

4. `<META name="ROBOTS" content="INDEX,NOFOLLOW">`

Ez a tag a robotok oldalkövetését és oldalindexelését irányítja. Négy módon lehet ezt az elemet alkalmazni. A tag **CONTENT** szekciója az alábbi értékeket veheti fel (több értéknél vesszővel elválasztva):

index = oldalt indexeli

noindex = oldalt ne indexelje

follow = oldal linkjeit kövesse

nofollow = oldal linkjeit ne kövesse

Használható még az **ALL** = INDEX, FOLLOW, és a **NONE** = NOINDEX, NOFOLLOW érték is.

5. `<META name="REVISIT-AFTER" content="10 days">`

Itt határozhatja meg, hány nap múlva látogassa meg oldaladat a keresőrobot újra frissítések után kutatva. Csak ilyen elem a használatával biztosíthatod a változások keresők általi nyomon követését. Nem minden keresőrobot hajtja végre ezt a parancsot.

6. `<META name="EXPIRES" content="XX">`

Ezzel az elemmel is azt határozhatod meg, hány nap múlva látogassa meg oldaladat a keresőrobot újra frissítések után kutatva. Csak ilyen elem a használatával biztosíthatod a változások keresők általi nyomon követését. Nem minden keresőrobot hajtja végre ezt a parancsot.

7. `<META name="AUTHOR" content="XX">`

Itt lehet a lap szerzőjének a nevét megadni.

8. `<META name="DATE" content="XX">`

Itt lehet a dokumentum készítésének idejét megadni.

9. `<META name="COPYRIGHT" content="XX">`

Itt lehet a dokumentum tartalmának copyright (szerzői jogi) adatait megadni.

10. `<META name="PUBLISHER" content="XX">`

Itt adható meg infó a publikáló intézményről.
Maximális karakter: 64

11. `<META name="MADE" content="XX">`

Itt adható meg a szerző email címe.

12. `<META name="REPLY TO" content="XX">`

Itt adható meg a reply email címed.

13. `<META name="GENERATOR" content="XX">`

Itt a készítéskor használt webszerkesztő szoftver neve adható meg. Számos webszerkesztő szoftver ezt a sort automatikusan hozzáfűzi a forráskódhoz.

14. `<META name="SUBJECT" content="XX">`

Meghatározhatod weblapod témáját. Egyes keresőknek segít az adatbázisában a megfelelő témakategóriába besorolni lapodat.

15. `<META name="PAGE-TYPE" content="XX">`

Itt meghatározhatod dokumentumod típusa. Például: "Privat Home Page"

16. `<META name="AUDIENCE" content="XX">`

Itt adhatod meg, milyen látogatói rétegnek szól a dokumentum. Pl: "ALL, CHILDREN, YOUNGS, SENIOR"

17. `<META http-equiv="REFRESH" content="5; URL=http://www.valami.hu">`

Itt valósítható meg automatikus továbbirányítás egy másik URL-re, pl. az oldal/webhely elköltözése esetén. Meghatározható, az oldal betöltése után hány másodperccel irányítson át a másik oldalra.

Pl: "5" = 5 másodperc

Figyelem: A keresőrobotok nem követik ez a parancsot, vagy az ugyanilyen hatású javascriptes parancsokat. Ilyen átirányítással a következő oldalt kizárod a kereső indexeléséből.

Megjegyzés: ha nem kerülhető el a használata, érdemes az oldalon a következő oldalra mutató linket is elhelyezni, mivel a keresőrobotok a linkeket követik, valamint a böngészők sem mindig hajtják végre ezt a parancsot)

18. `<META http-equiv="PRAGMA" content="NO-CACHE">`

Ez a parancs a proxy szervernek tiltja, hogy az oldalt tárolja. Minden alkalommal a tároló webszerverről fogja lekérni.

19. `<META http-equiv="CACHE-CONTROL" content="NO-CACHE">`

Itt tilthatod le a kliens böngészőnek és a proxy szervernek, hogy az oldal tartalmát átmeneti memóriában tárolja. Az oldalt minden eléréskor a webről fogja lekérni. Hasznos gyakran frissülő oldaltartalmaknál, pl: webkamera.

20. `<META http-equiv="CONTENT-LANGUAGE" content="english">`

Ezzel a metatag-al meghatározhatod dokumentumod nyelvét. A keresők könnyebben megtalálják oldaladnak a megfelelő adatbázist, illetve nyelvre szűkített keresésnél nem fogják oldaladat téves találatként kilistázni.

25. Az XHTML

Az XHTML-t azért hozták létre, hogy fölváltsa a HTML-t. Ma már a legtöbb böngésző támogatja. Előbb-utóbb mindenkiben felmerül, hogyan teheti elkészült oldalait az XHTML szabvány előírásainak megfelelővé. Alább néhány szempontban megpróbálom összefoglalni a legfontosabb tennivalókat.

Az XHTML sokkal szorosabb szabványkövetést igényel a korábbi HTML változatoknál. Vége lesz a böngészők által romjaiból felépített lapoknak, de a szabványosság megteremti az egységes megjelenés, a böngészőfüggetlenség feltételeit is.

Vannak HTML ag-ek és paraméterek, amiket az XHTML már nem támogat. Ezeket általában CSS stílusokkal helyettesíteni lehet.

Meglévő kódjainkat általában minden további nélkül átírhatjuk úgy, hogy az megfeleljen az XHTML követelményeinek. Ehhez az alábbiakban olvashatsz néhány szempontot:

1. A tag-eket mindig kis betűvel írjuk

Mivel az XML kis/nagybetű érzékeny és az XHTML ebből a nyelvből van levezetve, itt már nem mindegy, az elemek neveit kis-, vagy nagybetűvel írjuk - a csupa kisbetűs írásmód az elfogadott. Ez vonatkozik a HTML tag-ek és paramétereik neveire is.

Helytelen:

```
<P Align="right">Ez egy <B>bekezdés.</B></P>
```

Helyes:

```
<p align="right">Ez egy <b>bekezdés.</b></p>
```

2. Minden elemet le kell zárni

A HTML-ben eddig elfogadott volt bizonyos zárótagjának elhagyása, ilyenek például a <p>, elemek. Az XHTML-ben nincsenek lezáratlan elemek, ezért az elem zárótagját mindig ki kell tenni.

Helytelen:

```
<p>Első bekezdés  
<p>Második bekezdés  
<ul>  
  <li>Első listaelem  
  <li>Második listaelem  
</ul>
```

Helyes:

```
<p>Első bekezdés</p>  
<p>Második bekezdés</p>  
<ul>  
  <li>Első listaelem</li>  
  <li>Második listaelem</li>  
</ul>
```


3. A záró tag nélküli elemeket is le kell zárni

Vannak olyan elemek is, amelyeknek nincsen zárótagjuk, de a szabály ezekre is vonatkozik. Ezeket úgy zárjuk le, hogy a / jelet a elemet bezáró relációjel > elé tesszük így: />.

Mivel ez egyes böngészőkben hibás értelmezést okozhat, ajánlott a / jel előtt egy szóközt hagyni.

Helytelen:

```
  
<br>Valami szöveg
```

Helyes:

```
  
<br />Valami szöveg
```

4. A paraméterek értéke mindig idézőjelek közt legyen

A HTML-ben a paraméterek értékei közül azokat nem volt kötelező idézőjelbe tenni, amelyek csak számokat tartalmaztak. Ez is megváltozott: minden értéket idézőjelbe kell tennünk.

Helytelen:

```

```

Helyes:

```

```

5. Az tag-ek egymásba ágyazásának pontossága

Sok HTML elemet lehet egymásba illesztve is használni, például kijelölhetek egy szövegrészre egyszerre bold és <i> italic szövegformázást is.

Fontos: az elemek záró tagjai fordított sorrendben kerüljenek sorra, mint a nyitótagok.

Helytelen:

```
...szöveg <b><i> ez itt már ki van emelve <b><i> még több szöveg...
```

Helyes:

```
...szöveg <b><i> ez itt már ki van emelve <i><b> még több szöveg...
```

6. A rövid paraméterek nem megengedettek

Eddig engedélyezve volt a `checked`, `selected`, vagy a `readonly` *stb.* rövid paraméterek használata. Az XHTML ezt nem engedi meg, ki kell mindenhol írni: `checked="checked"`, `selected="selected"`, és `readonly="readonly"`.

Helytelen:

```
<input type="radio" name="radiogomb" value="igen" checked />
```

Helyes:

```
<input type="radio" name="radiogomb" value="igen" checked="checked" />
```

7. Linkekben & helyet &

A linkekben ezentúl nem lehet használni az `&` karaktert, helyette az `&` - kell kiírni.

Helytelen:

```
<a href="http://lapom.hu/index.php?egyik=elso&masik=masodik">
```

Helyes:

```
<a href="http://lapom.hu/index.php?egyik=elso&amp;masik=masodik">
```

8. Name helyett id

Eddig az elemek egyedi azonosítására a `name` attribútumot használtuk, amit a böngészők általánosan le is kezeltek. Ezután át kell térni a hasonló célra való `id` attribútumra. Mivel viszont egyelőre vannak böngészők, amelyek csak korlátozottan támogatják az `id` használatát, tanácsos mindkét paramétert használni.

Helytelen:

```

```

Helyes:

```

```

9. Módosul az oldalunk specifikációja

Miután már az XHTML szabványait használtunk az oldalunk specifikációja is megváltozott: **XHTML Strict DTD**, **XHTML Transitional DTD** és **XHTML Frameset DTD**. Ezek az elnevezések ugyanazt jelentik, mint a HTML specifikációk.

Tartalomjegyzék:

1. Mi az a HTML dokumentum?.....	2
2. Mi az a „TAG”?.....	2
3. A HTML dokumentum alapszerkezete.....	3
4. Megjegyzések beszúrása.....	3
5. Az oldalunk specifikációjának megadása.....	4
6. A háttér és a színek beállítása.....	5
6.1 Háttér beállítása.....	5
6.2 A szövegek színének beállítása.....	5
7. Háttérzene beállítása.....	6
8. Margók beállítása.....	6
9. Bekezdés, sortörés és szóköz.....	7
9.1 Bekezdések létrehozása.....	7
9.2 Sortörés és mesterséges szóköz.....	7
10. Betűtípusok módosítása.....	8
11. Fejlécek.....	9
12. Listák.....	9
12.1 Rendezett lista.....	9
12.2 Rendezetlen lista.....	10
12.3 Definíciós lista.....	11
13. Képek.....	12
14. Vízszintes elválasztóvonal.....	13
15. Hivatkozások és könyvjelzők.....	14
15.1 Hivatkozások.....	14
15.2 Könyvjelzők.....	14
16. Idézetek, blokkok elkülönítése.....	15
17. Táblázatok.....	15
18. Keretek.....	17
18.1 Keretes szerkezetű oldalak.....	17
18.2 Beágyazott keretek.....	19
19. Űrlapok.....	20
19.1 Szöveges beviteli mezők és jelszó mezők.....	20
19.2 Rejtett mezők.....	20
19.3 Kijelölő négyzetek.....	21
19.4 Rádiógombok.....	21
19.5 Gombok.....	22
19.6 Legördülő listák.....	22
19.7 Szöveges területek.....	22

19.8 Az űrlap elemek csoportosítása.....	23
20. Fényűjság létrehozása.....	24
21. Beépülő tartalmak.....	25
21.1 Videók.....	25
21.2 Flash animációk.....	25
22. A gördítősáv átszínezése.....	26
23. Kurzor beállítása.....	26
24. A HTML fejléce.....	27
24.1 A <BASE> tag.....	27
24.2 A <LINK> tag.....	27
24.3 A <TITLE> tag.....	28
24.4 A <SCRIPT> tag.....	28
24.5 a <META> tag.....	29
25. Az XHTML.....	32