

# HTML szerkesztés gyakorlat

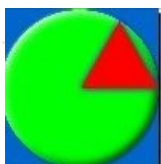
jegyzet

2010. április 4.

©Sallai András 2010



Készült az Oktatásért Közalapítvány támogatásával.



Egressy Gábor Kéttannyelvű Műszaki Szakközépiskola

## Tartalom

|  |    |
|--|----|
| 1. Alap címkék.....                      | 3  |
| 2. A weblap kódolása.....                | 4  |
| 3. A weboldal fejezetei, címei.....      | 5  |
| 4. Bekezdés, sortörés.....               | 6  |
| 5. Az oldal színei .....                 | 7  |
| 6. A szöveg önálló formázása .....       | 8  |
| 7. Betűk stílusa, hatások.....           | 10 |
| 8. Igazítás .....                        | 11 |
| 9. Háttérkép és képek beillesztése ..... | 12 |
| 10. Listák.....                          | 15 |
| 11. Táblázatok.....                      | 19 |
| 12. Linkek.....                          | 25 |
| 13. Logikai formázás .....               | 28 |
| 14. Különleges karakterek .....          | 28 |
| 15. Irodalom .....                       | 31 |

A tananyag feladatokkal kiegészítve, programozott formában megtalálható a <http://informatika.gtportal.eu/> címen.

Szerző: Sallai András, 2009-2010

©Sallai András 2010

Ez a mű Creative Commons Nevezd meg!-Ne add el!-Így add tovább! 2.5 Magyarország Licenc alatt van. A licenc szövegének megtekintéséhez látogasd meg a

<http://creativecommons.org/licenses/by-nc-sa/2.5/hu/>

webcímet vagy küldj egy levelet a következő címre: Creative Commons, 171 Second Street, Suite 300, San Francisco, California, 94105, USA.



## 1. Alap címkék

Egy **HTML dokumentum** egy **közönséges szöveges állomány**. Egy egyszerű szövegszerkesztővel például jegyzetömbbel is létrehozható. Minden HTML dokumentumot egy olyan sorral kezdünk ami leírja a milyen dokumentumot hozunk létre. Ezt sort dokumentum típus definíciónak nevezzük. Egy tipikus dokumentum típus definíció így néz ki:

```
<!DOCTYPE HTML PUBLIC „-//W3C//DTD HTML 4.01 Transitional//EN”>
```

Az egyes kifejezések jelentése:

| Kifejezés    | Jelentés  |
|--------------|---|
| !DOCTYPE     | Egy dokumentumtípus definíció következik              |
| HTML         | Ez egy HTML típusú dokumentum                         |
| PUBLIC       | A dokumentum nyílt                                    |
| -//W3C//DTD  | A W3C szövetség által kiadott HTML verziót használunk |
| HTML 4.01    | HTML verziót használunk, azon belül 4.01 verziót      |
| Transitional | Az elavult címkéket is szeretnénk használni           |
| //EN         | A címkék az angol ábécé betűkészletéből kerülnek ki   |

A dokumentumtípus definíció mondja meg a böngészőnek, hogy a **HTML-nek melyik verzióját fogjuk használni**. Ez biztosítja számunkra, hogy a különböző gyártmányú és verziójú böngészőkben a weblapunk **látványa** minél inkább **hasonló** legyen. A típusdefiníció után következik a dokumentum tartalma.

A webes dokumentum tartalmát egy <html> címkével kezdjük és egy </html> címkével zárjuk. A tartalmi részt két részre osztjuk:

- fejrész (angolul: head)
- törzs (angolul: body)

A dokumentum fejrészét a <html> címkéken belül <head> és a lezáró </head> címkék közé tesszük. A dokumentum törzsét a <html> címkéken belül a <head></head> címkék után következnek, amit a <body> kezdő és a </body> lezáró címkék közé tesszük. Ezek után egy HTML dokumentum váza így néz ki:

```
<!DOCTYPE HTML PUBLIC „-//W3C//DTD HTML 4.01 Transitional//EN”>
<html>
<head>

</head>
<body>

</body>
</html>
```

## 2. A weblap kódolása

Ahhoz, hogy a weblapunk a böngészőkben **karakterhelyesen jelenjenek meg** meg kell adnunk, hogy annak szerkesztésekor milyen **karakterkódolást** használtunk. A Windows XP alapértelmezetten (magyar nyelvű ) **ISO-8859-2** karakterkódolás használja.

Ha Jegyzetömbben, mentéskor ezt **ANSI** kódolásként látjuk viszont. A Jegyzetömbben azonban beállítható az Unicode karakter készlet is, amely már a világ összes nyelvének jelkészletét egyetlen kódolással képes megjeleníteni.

Az Unicode kódolásnak több megvalósítása van, elterjedt egyike az **UTF-8**. Szöveg és HTML szerkesztőkben az Unicode-ot ebben a formában találjuk meg. A korábbi ANSI esetén minden nyelvterületen külön kódolást kellett beállítani.

Ha Jegyzetömböt használunk, mentéskor alapértelmezetten az ANSI kínálja fel vagyis a HTML dokumentumunk ISO-8859-2 kódolású lesz. Ha mentéskor UTF-8 kódolást választjuk a HTML dokumentumban értelemszerűen az UTF-8 kódolást kell megadni. A következő sorok két ilyen lehetséges példát mutatnak:

```
<meta http-equiv="Content-type" Content="text/html; cahrset=iso-8859-2">
<meta http-equiv="Content-type" Content="text/html; cahrset=UTF-8">
```

A meta címkéket a head címkék közzé kell elhelyezni. A meta címkék http-equiv és Content tulajdonságával adjuk meg a lap kódolását. A text/html annyit jelent, hogy egy szemmel is jól olvasható html dokumentumot állítunk elő. A karakterkódoláshoz szorosan nem kapcsolódik a **title** címkék: <title> ...</title>, de a meta címkék előtt szokás megadni. A title címkék közzé írt szöveg a **böngészők címsorában jelenik meg**, az oldal címét szokás ide írni.

A HTML vázunk ezek után így fog kinézni:

```
<!DOCTYPE HTML PUBLIC „-//W3C//DTD HTML 4.01 Transitional//EN”>
<html>
<head>
<title>Az oldal címe</title>
<meta http-equiv=“Content-type” Content=“text/html; cahrset=iso-8859-2”>
</head>
<body>
Tartalom
</body>
</html>
```

Vegyük észre, hogy a meta címkének nincs lezáró része. A HTML verziók ezt megengedik az XHTML verziók azonban nem. Az XHTML-ben egy a címke végre egy perjelet kell tenni:

```
<meta http-equiv=“Content-type” Content=“text/html; cahrset=iso-8859-2”/>
```

### 3. A weboldal fejezetei, címei

A weblapunk címét h1, h2, h3, h4, h5 vagy h6 címkék közé helyezzük. Például ha weboldalunk címe a „Párduc” akkor így nézhet ki:

```
<h1>Párduc</h1>
```

A h1 ... h6 címkék között a különbség a betűk nagyságában van. Legnagyobb betűt a h1 címke produkálja, így ezt szoktuk használni az oldal címének megadására. Ennek megfelelően a következő alcímek, például „Színváltozatok”, „Táplálék”, „Alfajok” h2-es címkék közé kerül:

```
<h2>Színváltozatok</h2>
```

```
<h2>Táplálék</h2>
```

```
<h2>Alfajok</h2>
```

A „h” után megadott számmal tehát a fejezetszintet is megadjuk. A h6 adja a legkisebb betűméretet. A h az angol „head” szóból származik, amely magyarul fejnek vagy fejezetnek fordítható. Úgy is mondjuk fejezetcímnek adunk meg valamit. A példában a Párduc egyes fejezetcím. A Színváltozatok, Táplálék és az Alfajok kettes fejezetcím. Ezek után a body címkék közötti rész így fog kinézni:

```
<h1>Párduc</h1>
```

```
<h2>Színváltozatok</h2>
```

```
<h2>Táplálék</h2>
```

```
<h2>Alfajok</h2>
```



1. ábra: Fejezetcímek

## 4. Bekezdés, sortörés

Ha a weblap csak fejezetcímekből áll még mindig nincs tartalom. A tartalmat az egyes **fejezetcímek** után adjuk meg egy vagy több bekezdésben. Egy **bekezdést** a „p” címkék közé helyezünk el. Például a párduc táplálkozásáról szóló fejezet így nézhet ki:

```
<h2>Táplálkozás</h2>
<p>
A leopárd többféle állatot elejt.
Valamilyen magas helyről figyel
áldozatit, aztán oda lopakodik, majd
hirtelen ugrással támad.
</p>
```

Ha több bekezdést is készítünk az újabb bekezdések új sorban fognak kezdődni.

```
<h2>Táplálkozás</h2>
<p>
A leopárd többféle állatot elejt. Valamilyen magas
helyről figyel áldozatit, aztán oda lopakodik, majd
hirtelen ugrással támad.
</p>
<p>
Emberlakta területeken sajnos a házi állatokat is
megtámadja.
</p>
```

A „p” az angol **paragraph** szóból származik, amelyet paragrafusnak vagy **bekezdésnek** fordíthatunk.

A **sortöréseket** hozhatunk létre a „br” címkéssel. A „br” címkének nincs lezáró része, hiszen a sortörésen nincs mit lezárni. Az XHTML pedig megköveteli a címkék lezárását, mit teszünk ilyenkor? XHTML esetén a `<br />` vagy `<br/>` alakot használjuk (Kettő között különbség egy szóköz). HTML-ben viszont egyszerűen: `<br>` Ha több „br” címkét használunk egymás után, akkor üres sorokat tudunk létrehozni.

## Táplálkozás

A leopárd többféle állatot elejt. Valamilyen magas helyről figyel áldozatit, aztán oda lopakodik, majd hirtelen ugrással támad.

Emberlakta területeken sajnos a házi állatokat is megtámadja.

2. ábra: Bekezdések

## 5. Az oldal színei

A HTML-ben a színeket az úgynevezett **RGB kódolással** adjuk meg. Az RGB az angol **Red, Green és Blue** szavak kezdőbetűi, jelentésük piros, zöld és kék. E három szín keveréséből állítjuk össze a megjelenítendő színt. A piros, zöld vagy kék szín mennyiségét egy hexadecimális számmal adjuk meg. A hexadecimális szám hat jegyű és egy „#” karakter vezet be. Például: #32fcca. Az első két számjegy a **vörös**, a következő két számjegy a **zöld** és az utolsó két számjegy a **kék mennyiségét** határozza meg. Tiszta piros színt állíthatunk elő például a következő kóddal: #ff0000. A zöld szín: #00ff00. Kék szín: #0000ff. Ha kék és piros szín összekeverjük valamilyen lilás színt kapunk: #ff00ff. Minden színt hasonlóan állítunk össze.

Léteznek olyan színek, amelyeket az angol nevükkel adhatunk meg. Ezt a HTML 4/VGA (16 szín) nevű szabvány adja meg.

|        |         |         |         |
|--------|---------|---------|---------|
| black  | #000000 | silver  | #c0c0c0 |
| maroon | #800000 | red     | #ff0000 |
| green  | #008000 | lime    | #00ff00 |
| olive  | #808000 | yellow  | #ffff00 |
| navy   | #000080 | blue    | #0000ff |
| purple | #800080 | fuchsia | #ff00ff |
| teal   | #008080 | aqua    | #00ffff |
| gray   | #808080 | white   | #ffffff |

3. ábra: Színek nevei

Ezen felül működhetnek más angol színnevek is, azonban nem lehetünk biztosak abban, hogy minden böngésző ismeri. Több program is segít számunkra, olyan színek kiválasztásában amelyek nem adhatók meg angol neveikkel. Ilyen például a GIMP. Ha CSS-el dolgozunk a nevükkel megadható színek száma 140.

Egy weboldal háttérszínét a body címke bgcolor tulajdonságával állítjuk. Például a silver szín beállítása:

```
<body bgcolor="silver">
```

vagy RGB kóddal megadva:

```
<body bgcolor="#c0c0c0">
```

A weboldal szövegének a színét szintén a body címke egy tulajdonsága adja, ez a text tulajdonság. A szöveg színét navy színre szeretnénk állítani:

```
<body text="navy">
```

vagy

```
<body text="#000080">
```

Egyszerre persze több tulajdonság is megadható:

```
<body text="navy" bgcolor="silver">
```

vagy

```
<body text="#000080" bgcolor="#c0c0c0">
```

## Arcideg

Az arcideg (nervus facialis) a VII. agyideg.

4. ábra: Az oldal és a szöveg színe

## 6. A szöveg önálló formázása

A **body** címkében az **egész oldalra vonatkozó formázást** tudjuk megadni. Előfordul azonban, hogy egyes szövegrészeknek más színt szeretnénk. Esetleg az alapértelmezett nagyságon is szeretnénk változtatni. Erre szolgál a `<font>` címke. A `<font>` címkének van lezáró része is: `</font>`. A `<font>` és a `</font>` címkék között azokat a szövegeket tesszük, amelynek szeretnénk megváltoztatni a színét, méretét, stb. A példa kedvéért vegyük a következő szöveget:

```
„Ezek a hím szárnyas, nőtény szárnyas és szárnyatlan munkás.”
```

A fenti szöveget az oldalon fekete színnel jelenítjük meg:

```
<body text="#000000">
```

A szövegben a „szárnyatlan” szót szeretnénk kiemelni sötétkék színnel. Ehhez a `<font>` címkét használjuk a `color` tulajdonsággal. A fentiek tükrében a HTML kódrész így nézhet ki:

```
<body text="#000000">  
...  
Ezek a hím szárnyas, nőtény szárnyas és  
<font color="#000080"> szárnyatlan</font> munkás.  
...
```

Ezek a hím szárnyas, nőtény szárnyas és **szárnyatlan** munkás.

5. ábra: Egyedileg színezett szöveg



Ha még a szöveg nagyságát is szeretnénk megváltoztatni akkor size tulajdonságot használjuk:

```
<body text="#000000">  
...  
Ezek a hím szárnyas, nőstény szárnyas és  
<font color="#000080" size=5> szárnyatlan</font> munkás.  
...
```

A szöveg méretét 1-től – 7-ig állíthatjuk. A legkisebb méretet a size tulajdonság 1-es értéke szolgáltatja, a legnagyobbat a 7-es érték.

1 nagyság  
2 nagyság  
3 nagyság  
4 nagyság  
5 nagyság  
6 nagyság  
7 nagyság

6. ábra: Betű nagysága

A weboldal alapesetben a böngészőben beállított és az operációs rendszerben rendelkezésre álló fontkészlettel jeleníti meg a szöveget. Ha használjuk a „face” tulajdonságot akkor figyelemmel kell lenünk arra, hogy a weblapot olvasó felhasználó operációs rendszerén milyen betűtípusok állnak rendelkezésre.

A két legnépszerűbb operációs rendszeren ez általában más. A „face” tulajdonság viszont megenged magának több értéket is, amit vesszővel elválasztva állíthatunk be. Egy ilyen beállítás így nézhet ki:

```
<font face="Ariel, Veranda, Sans, FreeSans, Helvetica">  
Szöveg  
</font>
```

A sans, serif és mono nem konkrét betűtípusok, de nem minden böngésző ismeri.

- sans – talpatlan
- serif – talpas
- mono – írógép

# talpas szöveg talpatlan szöveg írógép szöveg

7. ábra: Font családok

Ha **Írógép** betűtípus szeretnénk Linuxos rendszerek számára elégséges lehet a „*Mono*” beállítása. Windowsos rendszereken a „**Courier**” ugyan írógép betűtípus, viszont nagyon pixeles. Ezért használjuk helyette a „Courier New” beállítást. A „font” címkét csak a HTML Transitional verziójában használhatjuk. A W3C szövetség nem elavultnak ítéli, és azt mondja használjuk helyette a HTML Strict verzióját, ahol CSS-el formázzuk meg a szöveg egészét vagy részeit.

## 7. Betűk stílusa, hatások

A szövegeket különféle stílusban formázhatjuk. Ilyen stílus a betűk kissé **vastagabb** írásmódja, amelynek elterjedt neve „**félkövér**”, angolul bold. A másik ilyen stílusbeli átalakítás a betűk *dőlt* írásmódja, angolul italic. Ezt szokták kurzív néven is emlegetni. Megjegyzendő, hogy a nyomtatott betűkkel foglalkozó tudomány, a tipográfia megkülönbözteti a dőlt és a kurzív hatást. Utóbbihoz formai változás is tartozik. Lehetséges betűhatások még az aláhúzott, az ~~áthúzott~~ és az írógép típus. Az írógép stílus alatt itt főleg arra kell gondolnunk, hogy a minden betű szélessége egyenlő. A HTML nyelvvel ellentétben a szövegszerkesztők általában ezt csak a megfelelő betűtípus kiválasztásával teszik lehetővé.

További lehetséges hatás a betűk nagyobbá vagy kisebbé tétele, a felső és alsó indexes írásmód.

A következő táblázat azt mutatja be, melyik hatáshoz milyen HTML címkét kell használnunk.

| Hatás       | Címke alkalmazása         | Angol eredete  | Eredmény                   |
|-------------|---------------------------|----------------|----------------------------|
| Félkövér    | <b>szöveg</b>             | Bold           | <b>szöveg</b>              |
| Dőlt        | <i>szöveg</i>             | Italic         | <i>szöveg</i>              |
| Aláhúzott   | <u>szöveg</u>             | Underline      | <u>szöveg</u>              |
| Írógép      | <tt>szöveg</tt>           | Teletype       | szöveg                     |
| Áthúzott    | <del>szöveg</del>         | STRIKE through | <del>szöveg</del>          |
| Nagyobb     | <big>szöveg</big>         | BIG            | <b>szöveg</b>              |
| Kisebb      | <small>szöveg</small>     | SMALL          | szöveg                     |
| Felső index | szöveg1<sup>szöveg2</sup> | SUPERscript    | szöveg1 <sup>szöveg2</sup> |
| Alsó index  | szöveg1<sub>szöveg2</sub> | SUBscript      | szöveg1 <sub>szöveg2</sub> |

A különböző hatások persze vegyesen is alkalmazhatók, azaz egymásba ágyazhatók. Például az „alma” szót ha egyszerre félkövér és dőlt hatást is elszeretnénk érní:

```
<b><i>alma</i></b>
```

A példában a „b” címkét kívülre tettük. Ez meg is cserélhetjük:

```
<i><b>alma</b></i>
```

Az egymásba ágyazás során viszont nem keverjük nyitó és záró címkék sorrendjét! Ha az „i” nyitó címke volt előbb, akkor ez legyen az „i” záró legyen a végén. Helytelen például a következő:

```
<b><i>alma</b></i>
```

Az XHTML nyelvtanában ez egyenesen megengedhetetlen. A böngésző gyártók persze megengedhetik a helytelen használatot, de ragaszkodjunk a szabványhoz a különböző böngészők minél egységesebb megjelenítése miatt.

Következő példában négy az ötödik hatványon kifejezést szeretnénk leírni számokkal, nagyobb betűkkel, a kitevő dőlt hatású írásmódja mellett:

```
<big>4<sup><i>5</i></sup></big>
```

Az eredmény ehhez hasonló:

45

## 8. Igazítás

A szöveget négyféle képen **igazíthatjuk, balra, középre, jobbra és sorkizárhatjuk.** Alapértelmezetten a szövegszerkesztőkhöz hasonlóan a böngészők a szöveget balra igazítják. A HTML többféle lehetőséget biztosít a szövegek igazítására. Konkrét html címkét azonban csak a középre igazításhoz tartalmaz, a jobbra igazítást tulajdonságok megadásával érhetjük el. Középre zárás a „center” címkével érhető el. Ha például az „Alma” szót középre szeretnénk igazítani, akkor így tehetjük meg:

```
<center>Alma</center>
```

A „center” címkének van mint látjuk van lezáró része is. A címke azonban az elavult kategóriába tartozik, mint a „font” címke, vagyis használatát nem ajánlják. Az egyes bekezdések igazítása a „p” címke „align” tulajdonságával is meghatározható.

Adott például a következő bekezdés:

„Speciális felépítésű tüdejüknek köszönhetően fejlett, hatékony kettős légzőrendszerük van. A kettős légcseré lényege, hogy nem csak belégzéskor történik gázcseré, hanem kilégzéskor is. A hangképzés nem a gégeben, hanem a hörgőkben történik, ezáltal éneklés közben is tudnak levegőt venni. „

A „p” címkék között jobbra igazítás a HTML forrásba így néz ki:

```
<p align="right">
```

Speciális felépítésű tüdejüknek köszönhetően fejlett, hatékony kettős légzőrendszerük van. A kettős légycsere lényege, hogy nem csak belégzéskor történik gázcsere, hanem kilégzéskor is. A hangképzés nem a gégeben, hanem a hörgőkben történik, ezáltal éneklés közben is tudnak levegőt venni.

```
</p>
```

Az „align” tulajdonságnak négy értéke lehet:

- left
- right
- center
- justify

A balra, jobbra és középre igazítás az angol szavak után egyértelmű lehet. A „left” a balra igazítás, a „right” a jobbra igazítás, a „center” a középre igazítás és a „justify” érték való a sorkizárásra. A sorkizárás alatt értjük a szöveg mind a két oldali egyenesre igazítását.

A fenti bekezdés sorkizárása:

```
<p align="justify">
```

Speciális felépítésű tüdejüknek köszönhetően fejlett, hatékony kettős légzőrendszerük van. A kettős légycsere lényege, hogy nem csak belégzéskor történik gázcsere, hanem kilégzéskor is. A hangképzés nem a gégeben, hanem a hörgőkben történik, ezáltal éneklés közben is tudnak levegőt venni.

```
</p>
```

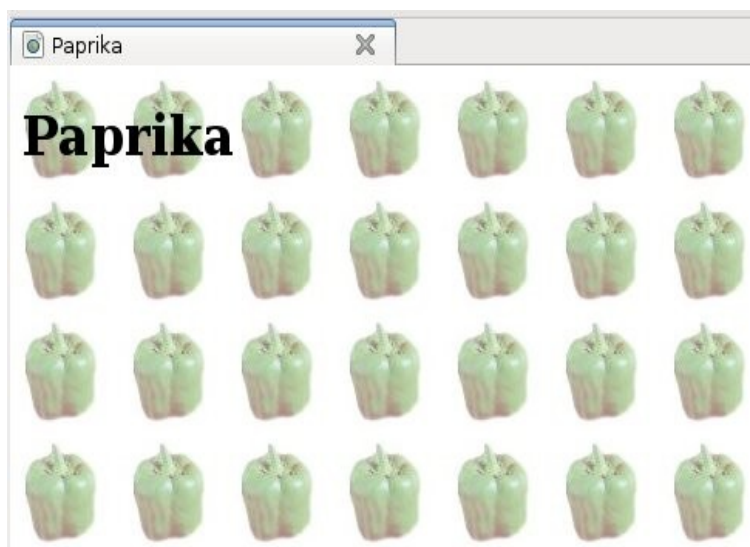
## 9. Háttérkép és képek beillesztése

Természetes igény a dokumentumokban **képek elhelyezése**. Ez általában vagy **háttérképnek a szöveg mögé** tesszük, vagy a **szöveg mellé** illesztjük be.

Háttérképként való beillesztésnél az oldal tartalma körül mindenhol megjelenik. Ha egy képet háttérképnek illesztünk be és az nem tölti ki az egész böngészőt, akkor **csempe szerűen ismétlődik**. A háttérkép beillesztése a „body” címke és a „background” tulajdonság segítségével lehetséges. A „background” értéke egy kép útvonallal együtt. A kép útvonala nem kötelező, és nem is szoktuk megadni, ha az a HTML állománnyal egyező könyvtárban van.

A beillesztett képeket mindig mellékelnünk kell a HTML állomány mellé, mivel azok sosem épülnek be magába a html állományba. A html állományban csak hivatkozunk rá. Ha adott például a hatter.jpg nevű képfájl, azt a következő módon illeszthetjük be háttérnek:

```
<body background="hatter.jpg">
```



ábra 8: Háttérkép csempezésűen

A háttérképek beillesztésénél ügyeljünk arra, hogy a szöveg színétől jelentősen eltérjenek a képe színei, másként az szöveg olvashatatlan lesz. Megjegyzés: CSS segítségével ennél finomabb formázási lehetőségek is rendelkezésünkre állnak.

A háttérkép mellett a **szövegbe is ágyazhatunk képeket** az „img” címke segítségével, amely az angol „image” szóból származik. Működéshez egy, szabvány betartásához minimum két tulajdonságot kell megadni. Az egyik az „src”, ami az angol „source” szóból származik. Itt adjuk meg a **kép nevét és útvonalát**, tehát nélkülözhetetlen. A másik tulajdonság az „alt” címke, az angol „alternate” szóból. Itt a kép tartalmának **szöveges leírását** adjuk meg, aminek akkor van jelentősége, ha a böngészőben nem jelenik meg a kép. A kép helyett alternatívaként a szöveg jelenik meg. A kép több oknál fogva nem jelenhet meg: nincs meg a kép, ki van kapcsolva a böngészőben vagy a böngésző karakteres. Egy paprika.jpg nevű kép beillesztése a következő módon tehető meg:

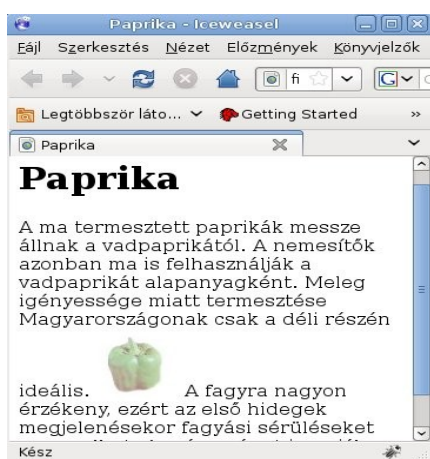
```

```

A kép szövegbe ágyazása esetén a paprikáról szóló HTML állományunk törzs része így nézhet ki:

```
<h1>Paprika</h1>
```

A ma termesztett paprikák messze állnak a vadpaprikától. A nemesítők azonban ma is felhasználják a vadpaprikát alapanyagként.  Meleg igényessége miatt termesztése Magyarországonak csak a déli részén ideális. A fagyra nagyon érzékeny, ezért az első hidegek megjelenésekor fagyási sérüléseket szenvedhet. A szárazságot igen jól tűri.



ábra 9: Paprika a szövegben

A paprika a szövegben nem a legideálisabb. A szöveg nem folyik körbe. Az „img” címke „align” tulajdonságával megadhatjuk a szöveg igazítását balra vagy jobbra a „left” vagy a „right” értékkel. A tulajdonság használata esetén a szöveg körbefolyik a kép körül. A példában a paprika.jpg képet igazítsuk balra, hogy a szöveg jobbról körbe folyjon.

```
<h1>Paprika</h1>
```

A ma termesztett paprikák messze állnak a vadpaprikától. A nemesítők azonban ma is felhasználják a vadpaprikát alapanyagként. `` Meleg igényessége miatt termesztése Magyarországon csak a déli részén ideális. A fagyra nagyon érzékeny, ezért az első hidegek megjelenésekor fagyási sérüléseket szenvedhet. A szárazságot igen jól tűri.



ábra 10: A kép körbefolyatása

A rendelkezésre álló kép megjelenítési **mérete** megváltoztatható az „img” címke „**width**” és „**height**” tulajdonságával. A „width” tulajdonsággal a kép szélességét, a „height” tulajdonsággal annak magasságát állíthatjuk be. A kép eredeti méretének megváltoztatásakor vigyáznunk kell az arányokra. Ha aránytalanul méretezzük át a képet az torzulni fog. A nagyítás pedig nem rossz képminőséget ad.

A példában szereplő paprika.jpg kép eredeti mérete 70x70. Ha szeretnénk kisebb méretet, például 50x50, akkor a html kód a következő lehet:

```

```

Kérdés, hogy milyen képet illeszthetünk be html dokumentumunkba. Minden böngésző ismeri a következő három képtípust:

- JPEG – kiterjesztése jpeg vagy jpg

- PNG – kiterjesztése png
- GIF – kiterjesztése .gif

A böngészők más képfájlokat is ismerhetnek, azonban azok használata egyáltalán nem javasolt. Más képtípusok vagy nem jelennek meg más böngészőkben, vagy túl nagyok ahhoz, hogy az Interneten használhassuk.

**JPEG** fájlokat használunk ha **fényképeket** szeretnénk megjeleníteni. A JPEG formátumot úgy alkották meg, hogy minél több szín tárolhasson, minél kisebbre tömörített méretben. A **GIF** típus kevesebb színt képes megjeleníteni, **grafikák** megjelenítésére ajánlott. A GIF formátumban létrehozhatunk mozgóképeket is, amit böngészők helyesen jelenítenek meg. A PNG formátumot a két típus kiváltására hozták létre. A mozgóképek lehetőségét azonban nem tették bele, mondván, hogy annak külön formátumban kell lennie. Ez az MNG.

Az MNG használata viszont egyáltalán nem terjedt el (ezen dokumentum írásának pillanatáig). **Nem ajánlott a BMP** képek használata html oldalak esetén. Ezek ugyanis nem jelennek meg minden böngészőben és hatalmas a méretük. A méretre még a fenti három kép esetében is figyelniük kell. Ha nagyon nagy képeket illesztünk a weblapunkba, a látogatók nem fogják megvárni amíg letöltődik, ott hagyják az weblapunkat.

Képek elhelyezésénél vigyáznunk kell a kis és nagybetűkre. Ha weboldalunkon az „img” címke „src” tulajdonságánál kisbetűvel adtuk meg a kép fájl neve pedig valójában nagy, egy Windows operációs rendszeren ez működhet. Azonban a weblap webkiszolgáló helyre való feltöltése után ezek nem működnek.

## 10. Listák

Három fajta listát tudunk létrehozni:

- Felsorolás
- Számozott lista
- Definíciós lista

Felsorolás esetén a felsorolt elemek előtt a böngészők alapértelmezésben nagyobb pontokat jelenítenek meg. HTML segítségével nem lehet rajtuk változtatni, webmesterek azonban CSS segítségével akár képeket is megjeleníthetnek a felsorolt szövegek előtt.

Az egész **felsorolást** az „ul” címkék közé tesszük, az egyes **elemek** pedig az az „li” címkék zárnak. Például a madarak, hüllők, kétéltűek és a halak felsorolása a következő módon történik:

```
<ul>
<li>madarak</li>
<li>hüllők</li>
<li>kétéltűek</li>
<li>halak</li>
</ul>
```

- madarak
- hüllők
- kétéltűek
- halak

*ábra 11: Felsorolás*

Számozott listákat az „ol” címkék között jelenítjük meg. Az egyes elemeket a felsoroláshoz hasonlóan az „li” címkékkal adjuk meg. A felsorolt elemek előtt a böngészők sorszámokat jelenítenek meg. A számozás egyessel kezdődik az első elemnél.

```
<ol>
<li>madarak</li>
<li>hüllők</li>
<li>kétéltűek</li>
<li>halak</li>
</ol>
```

1. madarak
2. hüllők
3. kétéltűek
4. halak

*ábra 12: Sorszámozás*

**Definíciós lista** esetén az egyes elemek előtt semmi sem jelenik meg, ellenben a **következő sorban** kissé beljebb kezdve, vagyis **behúzva**, az elem **magyarázatát** találjuk. A definíciós listákat az „dl” címkék közzé zárjuk. Az egyes elemek a „dt” címkék között helyezük. Az elemekhez tartozó leírásokat, illetve definíciókat pedig a „dd” címkék között adjuk meg.



```
<dl>
<lt>madarak</lt>
<dd>mellső végtagjai szárnyakká alakult gerincesek</dd>
<lt>hüllők</lt>
<dd>valamennyi magzatburkos állat, a kétéltűek és az emlősök kivételével</dd>
<lt>kétéltűek</lt>
<dd>a vízben és a szárazföldön egyaránt életképes gerincesek</dd>
<lt>halak</lt>
<dd>vízi életmódhoz szokott gerincesek</dd>
</dl>
```

madarak  
mellső végtagjai szárnyakká alakult gerincesek  
hüllők  
valamennyi magzatburkos állat, a kétéltűek és az emlősök kivételével  
kétéltűek  
a vízben és a szárazföldön egyaránt életképes gerincesek  
halak  
vízi életmódhoz szokott gerincesek

ábra 13: Definíciós lista

Ha a listákat egymásba ágyazzuk, akkor többszintű listát kapunk.

```
<ul>
<li>Hazai
<ul>
<li>körte</li>
<li>barack</li>
<li>szilva</li>
</ul>
</li>
<li>külföldi
</li>
</ul>
```

Egymásba ágyazott számozott listák:

```
<ol>
  <li>Hazai
    <ol>
      <li>körte</li>
      <li>barack</li>
      <li>szilva</li>
    </ol>
  </li>
  <li>Külföldi
    <ol>
      <li>banán</li>
      <li>citrom</li>
      <li>narancs</li>
    </ol>
  </li>
</ol>
```

## 1. Hazai

1. körte
2. barack
3. szilva

## 2. Külföldi

1. banán
2. citrom
3. narancs

14. ábra: Sorszámzás beágyazva

- Hazai
  - körte
  - barack
  - szilva
- Külföldi
  - banán
  - citrom
  - narancs

15. ábra: Sorszámzatlan beágyazva

## 11. Táblázatok

Táblázatot a „**table**” címkék segítségével hozhatjuk létre. A táblázatok **sorait** a „**tr**” címkével határozzuk meg az angol „table row” szavakból. A táblázat egyes **adatokat** tartalmazó tábláit a „**td**” címkékkel adjuk meg. A „td” az angol „table data” szavakból származik. Az alábbi táblázat például a néhány merevlemez árát tartalmazza:

```
<table>
  <tr>
    <td>Samsung</td><td>12250</td>
  </tr>
  <tr>
    <td>Hitachi</td><td>11900</td>
  </tr>
  <tr>
    <td>Seagate</td><td>16450</td>
  </tr>
</table>
```

|         |       |
|---------|-------|
| Samsung | 12250 |
| Hitachi | 11900 |
| Seagate | 16450 |

ábra 16: Táblázat

A táblázat alapértelmezetten nem tartalmaz keretet. Keretet a „border” tulajdonsággal határozhatjuk meg. A „border” értéke egy szám ami meghatározza a keret vastagságát. A fenti példa kerettel:

```
<table border="1">
  <tr>
    <td>Samsung</td><td>12250</td>
  </tr>
  <tr>
    <td>Hitachi</td><td>11900</td>
  </tr>
  <tr>
    <td>Seagate</td><td>16450</td>
  </tr>
</table>
```

|         |       |
|---------|-------|
| Samsung | 12250 |
| Hitachi | 11900 |
| Seagate | 16450 |

ábra 17: Táblázat kerettel

A táblázat oszlopainak címsorokat is meghatározhatunk.

```
<table border="1">
  <tr>
    <th>Gyártó</th><th>Ár</th>
  </tr>
  <tr>
    <td>Samsung</td><td>12250</td>
  </tr>
  <tr>
    <td>Hitachi</td><td>12250</td>
  </tr>
  <tr>
    <td>Seagate</td><td>12250</td>
  </tr>
</table>
```

| Gyártó  | Ár    |
|---------|-------|
| Samsung | 12250 |
| Hitachi | 11900 |
| Seagate | 16450 |

ábra 18: Táblázat fejléccel

A táblázat színét beállíthatjuk a „bgcolor” tulajdonsággal. Értékei mindig egy RGB színkód.

```
<table border="1" bgcolor="#e5e5e5">
  <tr>
    <th>Gyártó</th><th>Ár</th>
  </tr>
  <tr>
    <td>Samsung</td><td>12250</td>
  </tr>
  <tr>
    <td>Hitachi</td><td>12250</td>
  </tr>
  <tr>
    <td>Seagate</td><td>12250</td>
  </tr>
</table>
```

| Gyártó  | Ár    |
|---------|-------|
| Samsung | 12250 |
| Hitachi | 11900 |
| Seagate | 16450 |

ábra 19: Táblázat színnel

Az egyes sorok vagy adatcellákra szintén alkalmazható a külön szín. A táblázat fejléc sorának #aaaaaa színre állítása például a következő módon történhet.

```
<tr bgcolor="#aaaaaa">
```

A fenti táblázat teljes forrása:

```

<table border="1" bgcolor="#e5e5e5">
  <tr bgcolor="#aaaaaa">
    <th>Gyártó</th><th>Ár</th>
  </tr>
  <tr>
    <td>Samsung</td><td>12250</td>
  </tr>
  <tr>
    <td>Hitachi</td><td>12250</td>
  </tr>
  <tr>
    <td>Seagate</td><td>12250</td>
  </tr>
</table>

```

A egész táblázat és az egyes sorok mellett egy cella színe önmagában is megadható:

```
<td bgcolor="yellow">
```

| Gyártó  | Ár    |
|---------|-------|
| Samsung | 12250 |
| Hitachi | 11900 |
| Seagate | 16450 |

ábra 20: A táblázat fejléc más színnel

A táblázat címét is meghatározhatjuk a „caption” címke segítségével. A „caption” címkét a „table” címkét követően az első „tr” címke elé tesszük.

```

<table border="1" bgcolor="#e5e5e5">
  <caption>Merevlemez</a>
  <tr bgcolor="#aaaaaa"><th>Gyártó</th><th>Ár</th> </tr>
  <tr><td>Samsung</td><td>12250</td></tr>
  <tr><td>Hitachi</td><td>12250</td></tr>
  <tr><td>Seagate</td><td>12250</td></tr>
</table>

```

Merevlemez

| Gyártó  | Ár    |
|---------|-------|
| Samsung | 12250 |
| Hitachi | 12250 |
| Seagate | 12250 |

ábra 21: Táblázatfelirat

A táblázat tartalmának igazítása megadható a „tr”, a „th” és a „td” címkék „align” tulajdonságával. Értékük a szokásos: „left”, „center” és „right” lehet.

**A táblázat sorai és oszlopai összevonhatók.** Tegyük fel, hogy a fenti merevlemez példában a „Merevlemez” szót szeretnénk a táblázat részeként megadni, a „Gyártó” és az „Ár” feliratok felett. Ehhez a első sor oszlopait össze kell vonnunk. Oszlopok összevonása a „td” címke „colspan” tulajdonságával lehet. Értéke egy szám, amely megmondja, hogy hány oszlopot akarunk összevonni.

```
<table border="1" bgcolor="#e5e5e5" align="center">
  <tr><th colspan=2>Merevlemez</th></tr>
  <tr bgcolor="#aaaaaa"><th>Gyártó</th><th>Ár</th> </tr>
  <tr><td>Samsung</td><td>12250</td></tr>
  <tr><td>Hitachi</td><td>12250</td></tr>
  <tr><td>Seagate</td><td>13250</td></tr>
</table>
```

| Merevlemez |       |
|------------|-------|
| Gyártó     | Ár    |
| Samsung    | 12250 |
| Hitachi    | 12250 |
| Seagate    | 13250 |

ábra 22: Összevont oszlop

Összevonhatók a sorok is. A „td” címke „rowspan” tulajdonságával adhatjuk meg hány sort vonjunk össze. Tegyük fel, hogy a Segate merevlemeznek két árat szeretnénk megadni.

```
<table border="1" bgcolor="#e5e5e5" align="center">
  <caption>Merevlemez</caption>
  <tr bgcolor="#aaaaaa"><th>Gyártó</th><th>Ár</th> </tr>
  <tr><td>Samsung</td><td>12250</td></tr>
  <tr><td>Hitachi</td><td>12250</td></tr>
  <tr><td rowspan=2>Seagate</td><td>13250</td></tr>
  <tr><td>13900</td></tr>
</table>
```

Merevlemez

| Gyártó  | Ár    |
|---------|-------|
| Samsung | 12250 |
| Hitachi | 12250 |
| Seagate | 13250 |
|         | 13900 |

ábra 23: Összevont sor

A „table” címke „width” tulajdonságával beállíthatjuk egy táblázat **szélességét** pixelben megadva. Ha a megadott szélességben a tartalom nem fér el a böngésző nagyobb szélességet enged a táblázatnak. Éppen akkorát amelyben elfér a tartalom. A táblázat szélességét megadhatjuk százalékos formában is. Ilyenkor a táblázat mérete mindig az böngésző ablak méretéhez képest lesz kiszámítva. Ha a táblázatot középre igazítjuk, a szélességét pedig 80%-ra akkor a táblázat mindkét oldalán 10%, 10% marad. Ezzel a módszerrel 10%-os margót hozhatunk létre jobb és bal oldalon. Ezt a 10%-os margót a következő módon állíthatjuk be:

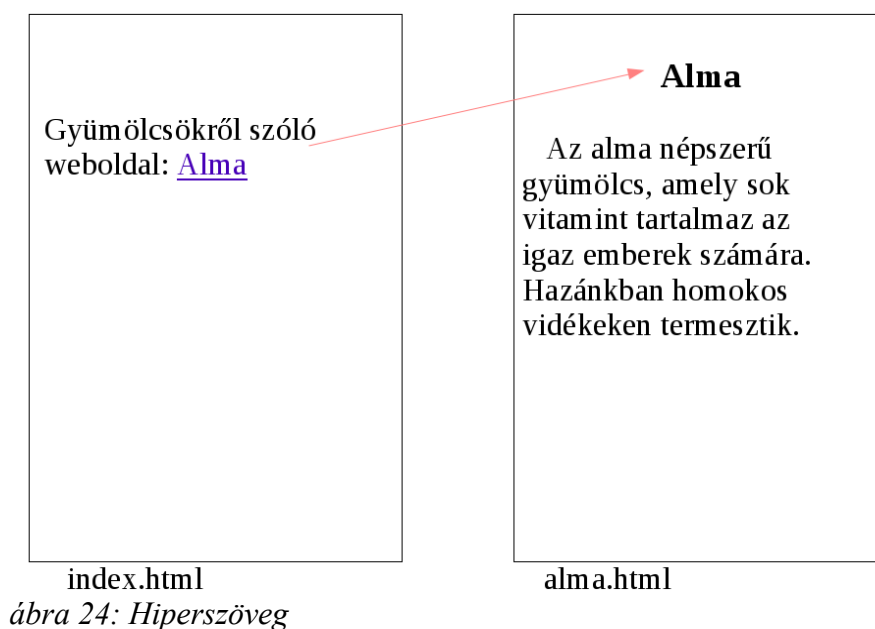
```
<table width="80%" align="center">
```



## 12. Linkek

Minden formázás keveset ér **hiperszövegek** nélkül. Ha egy szövegben elhelyezünk egy hiperhivatkozást egy másik weblapra „link” néven szoktuk emlegetni. Amire hivatkozunk a másik lapon **horgonynak** hívjuk. A linket a másik lapon elhelyezett horgonyhoz irányítjuk. Ha a hivatkozott lapon nem helyezünk el horgonyt a link akkor is használható. Akkor a hivatkozott lap eleje, illetve teteje jelenik meg.

A példa kedvéért legyen egy weblap amely index.html néven van elmentve, ami tartalmaz egy „Alma” szót. Ezt a szót szeretnénk hiperszöveggé azaz linké alakítani.



ábra 24: Hiperszöveg

Linkeket „a” címkével határozzuk meg:

```
<a>Alma</a>
```

Ettől azonban linkünk még nem működik, mert nem mutat sehova. Meg kell adnunk hova hivatkozzunk. Ezt az „a” címke „href” tulajdonságával tudjuk meghatározni. Legyen a példa kedvéért egy másik weblap, amire hivatkozni szeretnénk, a neve alma.html. A hivatkozásunk ekkor a következő módon nézhet ki:

```
<a href="alma.html">Alma</a>
```

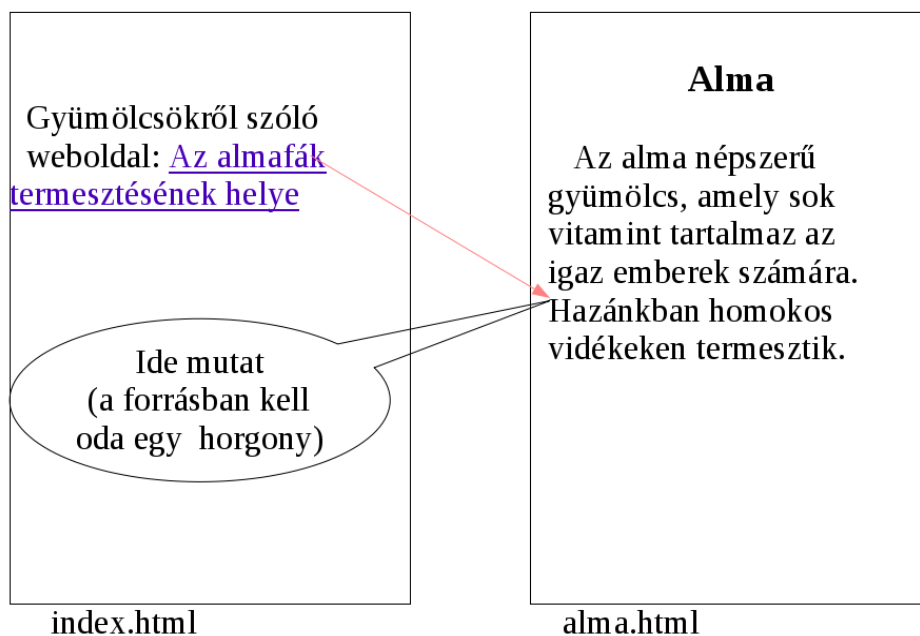
Ha egy html oldalban hiperszöveggé alakítunk egy szövegrészt, akkor annak megváltozik a színe és a stílusa. Alapesetben böngészőtől függően valamilyen kék színt fog kapni és aláhúzott lesz. A böngészők átszokták színezní a linket, ha már egyszer meglátogattuk, vagy éppen most nézegetjük. Így beszélhetünk éppen látogatott linkek színéről, és a már meglátogatott linkek színéről. A fenti példában az „Alma” szót hiperszöveggé alakítása után, azon egyet kattintva az alma.html oldal elején találjuk magunkat.

Természetesen megtehetjük, hogy az alma.html oldalról készítünk egy visszafele hivatkozást az

index.html-re:

```
<a href="index.html">Főoldal</a>
```

Ha a másik weblapnak nem a tetejére szeretnénk hivatkozni, hanem valahol lejjebb a szövegben, adott szövegrészben horgonyt kell elhelyeznünk.



ábra 25: A hiperszöveg egy horgonyra mutat

Horgony elhelyezése szintén az „a” címke való. Az „a” címke „name” tulajdonságával adhatunk meg egy horgonyt, a hivatkozott résznek mintegy nevet adva ezzel:

```
<a name="horgonyneve">
```

A példa kedvéért egy „videk” néven szeretnénk horgonyt elhelyezni. Legyen a hivatkozott oldal most is alma.html. Az alma.html fájlban a a szöveg megfelelő helyére a következő horgonybejegyzést tesszük:

```
<a name="videk">
```

```
<!DOCTYPE HTML PUBLIC
  „-//W3C//DTD HTML 4.01 Transitional//EN”>
<html>
<head><title>Alma</title></head>
<body>
<h2 align=“center”>Alma</h2>
```

Egy horgonyt  
helyezünk el a  
szövegben

**Az alma népszerű gyümölcs, amely  
sok vitamint tartalmaz az igaz  
emberek számára.**

**[<a name=“videk”>](#)**

**Hazánkban homokos  
vidékeken termesztik.**

```
</body>
</html>
```

*ábra 26: Horgony*

Az index.html weblapon amiben szeretnénk erre a horgonyra mutatni a linket a következő módon adjuk meg:

```
<a href=“alma.html#videk”>Az almafák természetes helye</a>
```

Megjegyzés: A `<a name=“videk”>` horgony az alma weblapon nem jelenik meg. Megtehetjük azt is hogy valamilyen szöveget zárunk az „a” címék közé:

```
<a name=“videk”>Hazai viszonyok</a>
```

Ez viszont az alma.html weboldalon meg fog jelenni.

Az eddigi példákban a hiperhivatkozás egy szöveg volt. Megtehetjük azt is, hogy egy képet teszünk meg linknek. Adott például az alma.jpg, amit szeretnénk hiperhivatkozásként megjeleníteni:

```
<a href=“alma.html”><img src=“alama.jpg” alt=“alma”></a>
```

### 13. Logikai formázás

A szöveget tartalmi szempontok alapján csoportosítjuk. A szöveghez tulajdonképpen egy típust rendelünk. A kinézetet az böngésző határozza meg.

| Címke              | Mikor használjuk                  | Angol eredet |
|--------------------|-----------------------------------|--------------|
| <em> </em>         | Kiemelés (általában dőlt lesz)    | EMphasize    |
| <strong> </strong> | Hangsúlyozás (általában félkövér) | STRONG       |
| <cite></cite>      | Közbezárt szövegrész: idézet      | CITE         |
| <code> </code>     | Programkód                        | CODE         |
| <var> </var>       | Valamilyen programváltozó         | VARiable     |
| <kbd> </kbd>       | Billentyűzetről várt bemenet      | KeyBoarD     |
| <samp> </samp>     | Program kimenetének részlete      | SAMPle       |

*Kiemelt szöveg*  
**Hangsúlyozás**  
*Közbezárt idézet*  
Programszöveg  
*Programváltozó*  
Billentyűzetről várt bemenet  
Program kimenet

27. ábra: Logikailag formázott szöveg

### 14. Különleges karakterek

Az UTF-8 karakterkódolás előtt voltak olyan karakterek amelyek a többféle kódrendszer miatt nem minden böngészőben ugyanúgy jelentek meg. Ma ha weblapunkat UTF-8 kódolással készítjük biztosak lehetünk abban, hogy az böngészőkben azt látják a felhasználók amit szeretnénk. Mindezek mellett továbbra is vannak olyan karakterek amelyeket ha szeretnénk megjeleníteni akkor a különleges karakterkódokhoz kell nyúlnunk. Ilyenek például azok a karakterek amellyel tagolunk egy weboldalt. Gondolok itt a kisebb mint jel: „<”. Nagyobb mint jel: „>”, de előfordulhatnak más karakterek is amit valamiért szeretnénk különleges karakterként megadni.

A **különleges karaktereket** a „&” karakter vezeti be, és „;” karakter zárja. A különleges karaktereknek van egy **kódjuk**, ami egy szám. A legtöbbnek azonban **neve** is van. A „&” és a „;” karakterek közzé ezt a kódot vagy nevet kell írunk. A „<” karakter ASCC kódja 60. A számmal megadott kódot mindig egy kettőskeresztrel kell megadnunk. Ezek után ha egy HTML oldalon szeretnénk ezt a karaktert megjeleníteni a következőt kell írjuk a html kódba:

&#60;

A nagyobb mint jel „>” száma 62. A nagyobb mint jel megjelenítése:

```
&#62;
```

Például ha egy C nyelvű kód fejrészét szeretnénk megjeleníteni, amely így néz ki:

```
#include <stdio.h>
```

akkor a következőket írjuk a htmlbe:

```
#include &#60;stdio.h&#62;
```

A számok helyett könnyebb neveket megjegyezni. A kisebb mint jel „<” ilyen neve „lt”, a nagyobb mint jel neve „gt”. Ha a kódot névvel azonosítjuk a kettőskereszt nem szükséges. A következő változtatásokat tehetjük a példában:

```
#include &lt;stdio.h&gt;
```

A fentiekhez hasonlóan minden karakter aminek gondot okozhat a megjelenítése leírható egy számmal vagy névvel. Az alábbiakban egy ilyen kódtáblázatot látunk a teljesség igénye nélkül.

|                |   |        |          |
|----------------|---|--------|----------|
| Kisebb jel     | < | &#60;  | &lt;     |
| Nagyobb jel    | > | &#62;  | &gt;     |
| Copyright      | © | &#169; | &copy;   |
| Paragrafus jel | § | &#167; | &sect;   |
| Registered TM  | ® | &#174; | &reg;    |
| Copyright      | © | &#169; | &copy;   |
| Font jele      | £ | &#163; | &pound;  |
| Fok jele       | ° | &#176; | &deg;    |
| Trade Mark     | ™ | &#153; |          |
| Ezrelék        | ‰ | &#137; |          |
| Mikro          | μ | &#181; | &micro;  |
| Plusz-mínusz   | ± | &#177; | &plusmn; |
| Euro           | € | &#128; | &euro;   |

A HTML vezérlőelemeket nem tekintve ha képesek vagyunk UTF-8 kódolással különleges karaktereket bevinni a billentyűzetről, akkor nem kell használnunk az előbbi htmlkódolást.

A HTML kódokat ha átszámítjuk hexadecimális számokká akkor azt látjuk, hogy azok

megegyeznek az UTF-8 kódokkal. A htmlkódolással is megadhatunk hexadecimális számot is egy „x” karakter beszúrásával jelezve, hogy htmlkódról van szó. Az Euró jelét például így írhatjuk:

&#x128

A fentiekből következik, hogy a UTF-8 kódolással az Euró jele: 0128. Ma már minden böngésző képes az bármely UTF-8 kódolással megadott megadott kódot megjeleníteni. Például pikk jelének UTF-8 kódja: 2660. HTML kóddal beírva:

pikk: &#x2660; ♠

Vagy a szigma és az integrál jele:

szigma: &#x03A3;  $\Sigma$

integrál: &#x222B;  $\int$

## **15. Irodalom**

- <http://w3.org>
- <http://w3c.hu/forditasok/#honositott>
- <http://w3schools.com/html/default.asp>