

WEBTERVEZÉS – GYAKORLATI JEGYZET

1. gyakorlat

Követelmények, HTML alapok I.

Készítették:

Cservenák Bence

Farkas Anikó

Savanya Sándor

FIGYELEM!

A jegyzet folyamatosan készül, így előfordulhatnak benne apróbb hibák, hiányosságok, elírások. Ha valaki esetleg ilyenre talál, kérem, írjon a `Cservenak.Bence@stud.u-szeged.hu` címre, hogy mihamarabb javíthassuk.

1. Közérdekű információk

- A gyakorlat tematikája, illetve követelményei elérhetők a gyakorlat Coospace színterén
- Néhány fontosabb segédanyag
 - Vancsics Béla weboldala: www.inf.u-szeged.hu/~vancsics/webterv
 - Abonyi-Tóth Andor: A weblapkészítés technikája (HTML5, CSS3) és ergonómiája
 - Angol nyelvű referencia: <https://www.w3schools.com/>

2. Áttekintés: A HTML, CSS és PHP nyelvek

	HTML	CSS	PHP
Micsoda?	jelölőnyelv	stílusleíró nyelv	szerveroldali szkriptnyelv
Szerepe	weboldalak tartalmának kialakítása, a tartalom strukturálása	a tartalom formázása	űrlapfeldolgozás, adatbázis-kezelés, menetkövetés, fájlkezelés, OOP, ...

3. A HTML alapjai (1. rész)

A. Bevezetés

A **HTML** (**H**yper**T**ext **M**arkup **L**anguage) segítségével mondjuk meg, hogy mi az, amit egy weboldalon látni szeretnénk (pl. szövegek, képek, hivatkozások, táblázatok, multimédia, ...).

A weboldal tartalmát lehetőségünk van strukturálni is, különböző szakaszok, tartalmi egységek kialakításával (pl. fejléc, menüsor, cím, indexkép, fő tartalom, lábléc, ...).

A HTML kód írása roppant egyszerű:

- A kódot tetszőleges szerkesztőprogramba írjuk (pl. Gedit, Notepad++, Webstorm, ...)
- Az állományt **.html** kiterjesztéssel elmentjük
- Végül a fájlt megnyitjuk egy tetszőleges böngészővel, és már láthatjuk is az eredményt

Megjegyzés: A HTML kódot más-más böngészők (és böngészőverziók) eltérő módon jeleníthetik meg

B. Szabványok

A weboldalak készítésére vonatkozóan a **W3C** nevű szervezet ad ki előírásokat, szabványokat.

Fontosabb szabványok:

- **XHTML 1.0 Strict (2000)**
 - a korábbi HTML 4 szabvány “újraírása” XML alapokon
 - szigorú szabályok
 - HTML dokumentumok egységessé tétele, egyszerűbb feldolgozás programmal
- **HTML5 (2014)**
 - kevésbé szigorú, mint az XHTML 1.0 Strict
 - programmal nehezebb feldolgozni
 - számos újdonság az elődjeihez képest (pl. szemantikus elemek, multimédia, grafikus elemek, új űrlapmező típusok)

A gyakorlaton a **HTML5** és **CSS3** szabványokkal fogunk foglalkozni. A webfejlesztés során érdemes törekednünk a választott szabvány előírásainak betartására!

A weboldalunk szabványosságát legegyszerűbben böngésző plug-innek vagy az alábbi online validator szoftverek segítségével ellenőrizhetjük:

- HTML validator: <https://validator.w3.org/>
- CSS validator: <https://jigsaw.w3.org/css-validator/>

Tipp: A beadandó projekt szabványosságát ezekkel egyszerűen tudjátok ellenőrizni

C. Alapfogalmak

- **Tagek:** < és > közé írt, speciális jelentésű HTML objektumok
 - a legtöbb tag **páros tag** – egy nyitótagból és egy zárótagból áll
 - pl. <p>Ide jön valami szöveg</p>
 - vannak viszont **páratlan (üres) tagek** is – csak nyitótagból állnak
 - pl.

 - opcionálisan a tag végére perjelet is tehetünk (pl.
)
 - MEGJEGYZÉS: a jegyzetben minden üres tag esetén kiírjuk a perjelet
 - tagek egymásba ágyazásakor a tageket belülről kifelé haladva zárjuk le
 - pl. <div><p><a></p></div>
- **Attribútumok:** tagekhez tartozó speciális tulajdonságok
 - speciális értékek tartoznak hozzájuk
 - az attribútum-érték párokat a nyitótag neve után írjuk, szóközzel elválasztva
 - pl. Google
 - pl.

D. A HTML dokumentum felépítése

Minden HTML dokumentum megírása az alábbi kóddal kezdődik. Nézzük meg, hogy mi mit jelent!

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

</head>
<body>

</body>
</html>
```

- `<!DOCTYPE html>`: jelezzük, hogy a HTML5 szabvány szerint írtuk a kódot
- `<html>...</html>`: ez írja le a weboldalt
- `<head>...</head>`: **fejrész** – az oldalon nem megjelenő tartalom (metainformációk) fájlcsatolások, CSS- és szkriptbeágyazások
- `<body>...</body>`: **törzsrész** – a megjelenő tartalom, strukturális elemek, szkriptek

E. Kommentek

A HTML fájlalba lehetőségünk van kommenteket (megjegyzéseket) írni az alábbi szintaxissal:

```
<!-- Ez egy komment a HTML kódban -->
```

F. A fejrész (head) fontosabb tagjei

- `<title>...</title>`: a böngésző fülén megjelenő cím
- `<meta charset="UTF-8"/>`: UTF-8 karakterkódolás beállítása
- `<meta name="description" content="Józsi blogoldala"/>`: a weboldal leírása
- `<meta name="author" content="Józsi"/>`: a weboldal szerzője
- `<meta name="keywords" content="webtervezés,webterv,HTML,CSS,PHP"/>`: kulcsszavak megadása (célja a találati esélyeink javítása)
- `<link rel="stylesheet" type="text/css" href="foo.css"/>`: külső CSS fájl beágyazása
- `<link rel="icon" type="image/png" href="icon.png"/>`: böngészőfülön megjelenő ikon
- `<style>...</style>`: CSS kód beszúrása
- `<script>...</script>`: szkript (pl. JavaScript) beszúrása

G. A törzsrész (body) fontosabb tagjai

a. Bekezdések, címsorok

- `<p>...</p>`: bekezdés
- `<h1>...</h1>`: 1. szintű címsor (legfontosabb)
- `<h2>...</h2>`: 2. szintű címsor
- `<h3>...</h3>`: 3. szintű címsor
- `<h4>...</h4>`: 4. szintű címsor
- `<h5>...</h5>`: 5. szintű címsor
- `<h6>...</h6>`: 6. szintű címsor (legkevésbé fontos)

```
<!DOCTYPE html>
<html>
<head>
  <title>Bekezdések, címsorok</title>
  <meta charset="UTF-8">
</head>
<body>
  <h1>1. szintű címsor</h1>
  <h2>2. szintű címsor</h2>
  <h3>3. szintű címsor</h3>
  <h4>4. szintű címsor</h4>
  <h5>5. szintű címsor</h5>
  <h6>6. szintű címsor</h6>
  <p>Ez egy bekezdés...</p>
</body>
</html>
```

1. szintű címsor

2. szintű címsor

3. szintű címsor

4. szintű címsor

5. szintű címsor

6. szintű címsor

Ez egy bekezdés...

b. Karakterentitások

Mivel a HTML szintaxisából adódóan bizonyos karakterek (pl. a `<` és `>` karakterek) le vannak foglalva speciális célokra, ezért ezek helyett célszerű ún. **karakterentitásokat** használni.

Néhány gyakori karakterentitás (teljes lista [itt](#)):

Karakterentitás	Megjelenítés
<code>&nbsp;</code>	(szóköz)
<code>&lt;</code>	<code><</code>
<code>&gt;</code>	<code>></code>
<code>&amp;</code>	<code>&</code>
<code>&quot;</code>	<code>"</code>
<code>&copy;</code>	©

c. Képek

Képet az `` páratlan taggel tudunk beszúrni a weboldalunkra.

Fontosabb attribútumai:

- **src**: a kép elérési útvonala (kötelező megadni)
- **alt**: helyettesítő szöveg, ha a kép nem jelenik meg (kötelező megadni)
- **title**: a kép címe, ami akkor jelenik meg, ha a kurzor a kép fölé áll
- **width**: szélesség (pixelben)
- **height**: magasság (pixelben)

Megjegyzés: Ha a **width** és **height** attribútumok közül csak az egyiket adjuk meg, akkor a másik a kép eredeti méretarányaiból számolódik. Például ha egy 400x300-as (400 px széles, 300 px magas) kép szélességét 800 px-re állítjuk (megduplázzuk), akkor a magassága automatikusan 600 px lesz.

Megjegyzés: A képek szélességét és magasságát célszerűbb CSS-ben megadni.

Példa: A HTML fájlal egy mappában van egy `img` mappa, amiben található a `tree.jpg` képfájl. Szúrjuk be a képet a weboldalunkra!

```

```



Ez egy fa

d. Az **id** és **class** attribútumok

Minden HTML objektumnak adható egyedi azonosító, és minden HTML objektum csoportosítható osztályokba. Ezek CSS-ben lesznek majd hasznosak, amikor egy-egy objektumra vagy objektumok csoportjára szeretnénk hivatkozni.

Az **egyedi azonosító** (**id**) értéke a dokumentumon belül egyedi kell, hogy legyen. Emellett az attribútum értéke nem tartalmazhat szóközt.

Az **osztály** (**class**) értéke nem kell, hogy egyedi legyen (sőt általában nem az). Értékekül szóközzel elválasztva megadható több osztály neve is, amihez az adott elem tartozik.

```
<!-- Egy "asd" egyedi azonosítóval rendelkező bekezdés -->
<p id="asd">Ide jön valami szöveg...</p>

<!-- A "foo" és "bar" osztályokhoz tartozó bekezdés -->
<p class="foo bar">Ide jön valami szöveg...</p>
```

e. Hivatkozások (Linkek)

Hivatkozásokat az `<a>` és `` tagpár segítségével szúrhatunk be a weboldalunkra.

Fontosabb attribútumok:

- **href**: cél, ahova a hivatkozásra kattintva jutunk
 - lehet egy másik weboldal
 - pl. `Google`
 - lehet egy másik dokumentum
 - pl. `ZH pontszámok`
 - lehet lapon belüli hivatkozás
 - pl. `Vissza a tetejére`
 - ez az `id="top"` attribútummal rendelkező objektumra mutat
- **target**: hol legyen megnyitva a hivatozott cél
 - ugyanott, ahol rákattintottunk (alapértelmezett): `target="_self"`
 - új lapon vagy ablakban: `target="_blank"`
 - az ablak teljes méretében: `target="_top"`
- **title**: hivatkozás címe (ha a kurzor a hivatkozás fölött áll, akkor jelenik meg)

```
<!-- Egy hivatkozás, ami egy képet nyit meg egy új lapon -->
<a href="cukicicak.jpg" target="_blank">Stressz ellen</a>

<!-- Egy hivatkozás, ami egy másik weboldalt nyit meg ugyanott, ahol a linkre
kattintottunk, és az oldal id="Behavior" attribútumú elemére mutat -->
<a href="https://en.wikipedia.org/wiki/Goat#Behavior">Kecske</a>
```

f. Szakaszok

HTML-ben lehetőségünk van a weboldal tartalmának strukturálására. Ezt szakaszok, tartalmi egységek kialakításával tudjuk megtenni a weboldalon belül.

- `<div>...</div>`: blokkszintű szakasz
 - mindig új sorban kezdődik, és kihasználja a rendelkezésre álló szélességet
- `...`: sorszintű szakasz
 - nem kezdődik új sorban, szélessége csak akkora, mint amekkora helyre szüksége van

A `span tag` és `<div>div tag</div>` szakaszok kialakítására alkalmas.

A `span tag` és `div tag` szakaszok kialakítására alkalmas.