

# INFO1 – WEB, HTML, CSS

Kovács Kristóf, Pálovics Róbert, Wettl Ferenc

2015. november 3.

## 1 Bevezetés

## 2 WWW

- A kezdetek
- A web működése vázlatosan
- Sütik

## 3 HTML

- Jelölőnyelv
- Tartalom és forma
- További nyelvi elemek

## 4 CSS

# Amit megtanulunk

- HTML alapok
- CSS alapok
- A tanultak alapján mindenki elkészítheti saját honlapját!
- Alapismeretek forrása: a W3C: World Wide Web Consortium ([www.w3.org](http://www.w3.org)) oldala, valamint a [www.w3schools.com/html/](http://www.w3schools.com/html/) és a [www.w3schools.com/css/](http://www.w3schools.com/css/) kurzus.
- Nagyobb munka előtt érdemes tanulmányozni a programozói stílusra vonatkozó ajánlásokat (pl. Google HTML CSS guide)
- A kész fájlt érdemes „validálni”, azaz a kód helyességét ellenőriztetni pl. a W3C-vel: HTML-t [itt](#), CSS-t [itt](#).

## 1 Bevezetés

## 2 WWW

- A kezdetek
- A web működése vázlatosan
- Sütik

## 3 HTML

- Jelölőnyelv
- Tartalom és forma
- További nyelvi elemek

## 4 CSS

## 1 Bevezetés

## 2 WWW

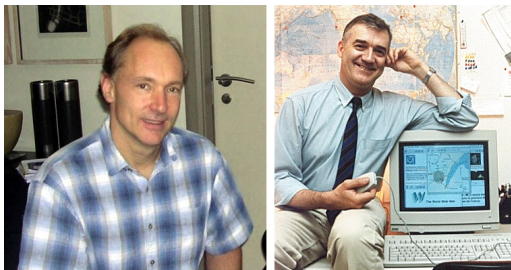
- A kezdetek
- A web működése vázlatosan
- Sütik

## 3 HTML

- Jelölőnyelv
- Tartalom és forma
- További nyelvi elemek

## 4 CSS

# World Wide Web



Ábra : Sir Tim Berners-Lee (bal) és Robert Cailliau (jobb)

Sir Tim Berners-Lee és Robert Cailliau 1990-ben a CERN-ben tervezték meg a WWW-t, hogy egy olyan hálózatot hozzanak létre amelyben bárki hozzáférhet adott információkhoz.

# World Wide Web

- Még 1990 végén elkészítették az első böngészőt, majd 1991-ben lett ténylegesen publikus a WWW az interneten.
- Hogy ez effektíven működhessen bevezették az URL-t (Uniform Resource Locator), a HTML-t (HyperText Markup Language) és a HTTP-t (HyperText Transfer Protocol).
- A CERN 1993-ban bejelentette, hogy a Web ingyenes lesz mindenkinek, majd később ebben az évben elkészült az első igazán elterjedt böngésző, a Mosaic.

## 1 Bevezetés

## 2 WWW

- A kezdetek
- A web működése vázlatosan
- Sütik

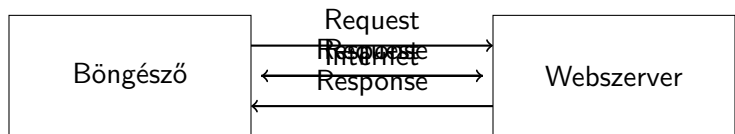
## 3 HTML

- Jelölőnyelv
- Tartalom és forma
- További nyelvi elemek

## 4 CSS



# Böngésző és webszerver



A böngészőt összeköti az internet egy webszerverrel.

Küld a böngésző egy kérést (request) a webszervernek, hogy melyik adott oldalt szeretné lekérdezni, meg még küld egyéb dolgokat is mellette.

Kap erre egy választ (response), ami tartalmazza a HTML-t amit majd a böngésző megjelenít a felhasználónak.

Így folytatódik a böngészés, akárhányszor új oldalra akar lépni a



## 1 Bevezetés

## 2 WWW

- A kezdetek
- A web működése vázlatosan
- Sütik

## 3 HTML

- Jelölőnyelv
- Tartalom és forma
- További nyelvi elemek

## 4 CSS

# Sütik (cookies)

- Egy webserver másodpercenként több ezer kérést kaphat, ez egy statikus oldalnál nem jelent gondot, de mi a helyzet például egy közösségi oldallal?
- A felhasználó be tud jelentkezni és utána bejelentkezve is marad, de honnan tudja a webserver, hogy ki kicsoda?
- Egy webserver válaszként nem csak HTML kódot küldhet, többek közt sütiket is.
- A süti egy ideiglenes élettartalmú minimális szövegfájl, mely egy adott weboldalhoz tartozik, melyet a webserver küld és a felhasználó számítógépére a kereső menti egy erre kijelölt könyvtárba.
- Amikor kérést küld a böngésző egy weboldalnak, elküldi az oldalhoz tartozó sütiket is, innen tudja a webserver, hogy „ki” küldte a kérést.

# Sütik tartalma

- *Név:* A süti neve
- *Érték:* Az adat amit tárol
- *Tulajdonságok:*
  - *Lejárati idő:* mikor kell a böngészőnek törölnie a sütit
  - *Domain:* melyik weboldalhoz tartozik a süti
  - stb.

A böngésző ezek közül csak a süti nevét és értékét küldi el a kérésben, a többi a böngészőre tartozik.



Name:	PREF
Content:	ID=1111111111111111
Domain:	.google.hu
Path:	/
Send For:	Any type of connection
Expires:	31 December, 2015 17:02:22



# Sütik biztonsága

- A süti nem tartalmazhatja egyszerűen a felhasználónevet, hisz ekkor nagyon könnyen hamisítható lenne.
- A sütiben nem vírus, nincs benne futtatható program.
- Tipikusan egy hosszú kulcsot (karakter sorozatot) tartalmaz, melyet a webszerver is tárol és összekapcsolja a felhasználóval.
- Így a felhasználó nem tudja, hogy másokhoz milyen kulcs tartozik, a webszerver viszont tudja, hogy melyik kulcs melyik felhasználóé.
- Példa: a keresők nagy része a keresési eredményeit személyre szabja és bejelentkezés nélkül is odafigyel a felhasználó preferenciáira.

## 1 Bevezetés

## 2 WWW

- A kezdetek
- A web működése vázlatosan
- Sütik

## 3 HTML

- Jelölőnyelv
- Tartalom és forma
- További nyelvi elemek

## 4 CSS

## 1 Bevezetés

## 2 WWW

- A kezdetek
- A web működése vázlatosan
- Sütik

## 3 HTML

- Jelölőnyelv
- Tartalom és forma
- További nyelvi elemek

## 4 CSS

# HTML jelölő nyelv – címke (tag), attribútum (attribute)

- Jelölő (markup) nyelv, címkékből (tag-ek) áll.
- Szövegszerkesztőben szerkeszthető, programmal könnyen generálható.
- A HTML-kód elemén a kód egy címkével megjelölt részét értjük.
- Ennek formája `<címke_neve>tartalom</címke_neve>`, ha valamilyen tartalomra vonatkozik és `<címke_neve>` vagy `<címke_neve />`, ha önmagában áll.
- Például `<em>szöveg</em>` jelöli, hogy a *szöveg* kiemelt fontosságú, míg `<br>` vagy `<br />` egy sortörést eredményez.
- A címkéknek (tag-eknek) lehetnek paramétereik/attribútumai. Ezek formája `attribútum_név="tulajdonság"` alakú, például Itt az `<a href="orarend.html">órarendem</a>`.
- Megjegyzés a kódban: `<!-- akármi van itt nem látszik -->`



# Példa oldal

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

<h1>Cím</h1>

<h2>Alcím</h2>

<p>Egy bekezdés.</p>

</body>
</html>
```

Cím

Alcím

Egy bekezdés.

# A fontosabb címkék

- `<h1>Cím</h1>` (heading)
- `<h6>Al...alcím</h6>` (heading)
- `<p>Bekezdés</p>` (paragraph)
- `<em>Kiemel</em>` (emphasize)
- `<strong>Erősen kiemel</strong>` (strong)
- `<a>Horgony (link)</a>` (anchor)
- `<ul>Számozatlan lista</ul>` (unordered list)
- `<ol>Számozott lista</ol>` (ordered list)
- `<li>Listaelem</li>` (list item)
- `<blockquote>Idézet</blockquote>` (blockquote)
- `<div>Rész</div>` (division)
- `<span>Sorközi csoportosítás, arasznyi</span>` (span)
- `<img>` (image)

## 1 Bevezetés

## 2 WWW

- A kezdetek
- A web működése vázlatosan
- Sütik

## 3 HTML

- Jelölőnyelv
- Tartalom és forma
- További nyelvi elemek

## 4 CSS

# A tartalom és a vizuális megjelenítés

- Néhány nem tartalmi (vizuális) címke:
  - `<i>Kurzív (dőlt)</i>` (italic)
  - `<b>Félkövér</b>` (bold)
  - `<sub>Alsó index</sub>` (subscript)
  - `<sup>Felső index</sup>` (superscript)
  - `<pre>Előformázott</pre>` (preformatted)
  - `<hr />` (horizontal rule – thematic change in the content)
  - `<br />` (line brake)
- A tartalom szervezéséről
  - A `div` és a `span` a szemantikai összetartozást jelölik.
  - A `div` nagyobb szövegrészt zár magába, felette lehet akár még egy `div`, mely `div`-eket zár össze egy részbe.
  - A `span` soron belül végzi ezt.

# Listák

## ■ Számozott lista:

```
<ol>
  <li>...</li>
  <li>...</li>
  <li>...</li>
</ol>
```

## ■ Definíciós lista:

```
<dl>
  <dt>Palindrom</dt>
  <dd>Ugyanaz fordítva</dd>
  <dt>Varjú</dt>
  <dd>Egy madar</dd>
</dl>
```

## ■ <li>...</li> egy elem a listában

## ■ Számozatlan lista:

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

# Táblázat

```
<table>
  <tr>
    <th>Fej1</th>
    <th>Fej2</th>
  </tr>
  <tr>
    <td>Adat1</td>
    <td>Adat2</td>
  </tr>
</table>
```

- `<tr>` táblázat sora
- `<th>` táblázat feje
- `<td>` táblázat adata

Fej1	Fej2
Adat1	Adat2

## 1 Bevezetés

## 2 WWW

- A kezdetek
- A web működése vázlatosan
- Sütik

## 3 HTML

- Jelölőnyelv
- Tartalom és forma
- További nyelvi elemek

## 4 CSS

# Attribútumok

- title cím egy HTML-elemhez:

```
<p title="Ez nagyon fontos!">Itt van a lényeg.</p>
```

- href az a címke mellett a link címét adja meg:

```
HTML <a href="http://www.w3schools.com">kurzus</a>.
```

- src a hivatkozás forrása, alt az alternatív megjelenés (pl. felolvasáskor vagy ha nem találja a képfájlt), width és height pixelben értendő:

```

```

- style az elem stílusát adja meg style="tulajdonság:érték" alakban (ez a jövő óra anyaga)

```
<body style="background-color:lightgrey">
```

```
<p style="color:red">Pirossal szedett bekezdés.</p>
```

```
<p style="font-family:courier">Courier bekezdés.</p>
```

```
<h1 style="text-align:center">Középre zárt cím</h1>
```





# Speciális karakterek

karakter	html-ben	(angol) megnevezés
<	&lt;	less-than
>	&gt;	greater-than
&	&amp;	ampersand
	&nbsp;	non-breaking space
$\forall$	&forall;	for all
$\exists$	&exist;	exists
$\emptyset$	&empty;	empty sets
$\in$	&isin;	element
$\notin$	&notin;	not an element
$\prod$	&prod;	product
$\sum$	&sum;	sum

## 1 Bevezetés

## 2 WWW

- A kezdetek
- A web működése vázlatosan
- Sütik

## 3 HTML

- Jelölőnyelv
- Tartalom és forma
- További nyelvi elemek

## 4 CSS

# CSS

- CSS: Cascading Style Sheets
- Cél: a tartalom és a megjelenítés szétválasztása (a HTML eredeti célja)
- Fizikai szétválasztás lehetősége: stílus .css-fájl(ok)ban
- HTML elemeinek formázására szolgál
- Előnyei: újrafelhasználhatóság és könnyű módosíthatóság
- Forrás tanuláshoz: [www.w3schools.com/css](http://www.w3schools.com/css),
- Validálás: [jigsaw.w3.org/css-validator/](http://jigsaw.w3.org/css-validator/)
- Csak szórakoztatásul a lehetőségekről, nem követendő:  
<http://www.csszengarden.com/>

# CSS szintaxis

- A CSS a következő alapvető szintaxist követi:

```
kiválasztó {tulajdonság: érték; tulajdonság2: érték2;}
```

- Pl:

```
p {color: red;}
```

vagy több deklaráció áttekinthetően szedve:

```
p {  
    color:#f00;  
    background: white;  
}
```

- Több elem stílusa egyszerre megadható:

```
h1, h2 {color: red;}
```

- Megjegyzés (comment):

```
/* Ide bármit írhatok, akár több sorban is. */
```



# CSS kód elhelyezése

- Inline (szövegközi):

```
<p style=".....CSS kódok.....">bekezdés szövege</p>
```

- Beágyazott, ez a html *head* részébe megy:

```
<style>  
    CSS formázás kódolása  
</style>
```

- Külső stílusfile (ez a legelterjedtebb), szintén a *head*-be:

```
<link rel="stylesheet" href=".....css">
```

- Legerősebb az inline, majd a beágyazott és végül a külső.

# Csoportosítás

A html kódban csoportosíthatjuk a *címkéinket* (tageinket). Egyedi nevet adhatunk nekik, vagy egy közös osztályba rakhatjuk őket.

**id** weboldalankét egyediek az id-k, és egy elemnek csak egy id-ja lehet, így ez egyfajta elnevezésként képzelhető el

```
<div id="hirek">.....</p>
```

**class** több elemnek lehet ugyanaz az osztálya, és egy elemnek több osztálya is lehet

```
<p class="megjegyzes">.....</p>
```

```
<p class="megjegyzes">.....</p>
```

```
...
```

```
<div class="megjegyzes">.....</div>
```

# Kiválasztók

- elemkiválasztó: mindig egy html jelölőelemre vonatkozik  
`p {color: red;}`
- id kiválasztó: az adott id-vel jelölt elemre vonatkozik  
`#hirek {background: white;}`
- osztálykiválasztó (class kiválasztó): az adott osztályba tartozó elemekre vonatkozik  
`.megjegyzes {font-size: small;}`
- kombinálhatjuk az elem- és osztálykiválasztót, pl:  
`p.megjegyzes {color: blue;}`  
ekkor azok a p-k lesznek kékek, amik a megjegyzés osztályba tartoznak

# Háttér

```
body {  
    background-color: #b0c4de;  
}  
h1 {  
    background-color: #6495ed;  
}
```

Kép a háttérben:

```
body {  
    background-image: url("halvany.gif");  
    background-repeat: no-repeat; /* repeat-x | repeat-y */  
    background-position: right top;  
}
```

Tömör írásmód:

```
body {  
    background: #fff url("back.png") no-repeat right top;
```





# Link szabályok, LoVe-HAtE

- Linkeknél vannak még speciális kijelölő elemek, ezekkel mondhatjuk meg, hogy hogyan nézzen ki egy link
  - alaphelyzetben (`link`),
  - ha már látogatott oldalra mutat (`visited`),
  - ha épp fölé visszük az egeret (`hover`),
  - amikor épp rákattintunk (`active`).
- Ezeket mindig ebben a sorrendben kell megadni:

```
a:link {
  color: green;
  text-decoration: underline;
}
a:visited { color: purple; }
a:hover {
  color: blue;
  text-decoration: none;
}
a:active { color:red; }
```

# Szövegek

color	#ff0000   #f00   rgb(255,0,0)   red
letter-spacing	3px   initial   normal
text-align	left   right   center   justify
text-decoration	none   underline   overline   line-through   blink
text-indent	50px   3em
text-transform	uppercase   lowercase   capitalize
white-space	nowrap

```
h1 {text-decoration: overline;color: #24c4f9;}
a {text-decoration: none;} /* az aláhúzás eltüntetése */
p.uppercase {text-transform: uppercase;}
p.date {text-align: right;}
```

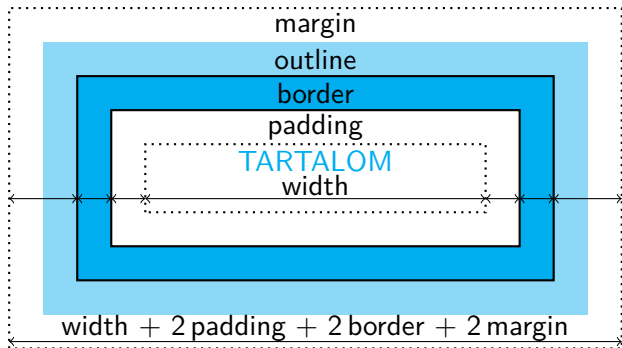
# Betűtípusok

tulajdonság		néhány lehetséges érték
font-family	family-name	serif   sans-serif   monospace
	generic-family	"Times New Roman"   Georgia   Verdana   Arial   "Courier New"   "Lucida Console"
font-size		small   medium   large   <i>length</i> (40px, 3em)   %
font-style		normal   italic

PI:

```
h2 {
  font-family: "Times New Roman", Times, serif;
  font-style: italic;
  font-size: 20px;
}
```

# CSS dobozok



```
div {
  width: 400px;
  padding: 10px;
  border: 10px solid navy;
  margin: 10px;
```

# Dobozok paramétere

height	auto   <i>length</i> (px, em)   %
width	auto   <i>length</i> (px, em)   %
padding	<i>padding-top padding-right padding-bottom padding-left</i>
margin	<i>margin-top margin-right margin-bottom margin-left</i>
margin-bottom	auto   length   %
border	<i>border-width border-style border-color</i>   initial   inherit
border-width	thin   medium   thick   <i>length</i>
border-style	none   hidden   dotted   dashed   solid   double
border-color	<i>color</i>

# Dobozok méretei tömören

- padding: 25px 50px 75px 100px;
  - top padding: 25px
  - right padding: 50px
  - bottom padding: 75px
  - left padding: 100px
- padding: 25px 50px 75px;
  - top padding: 25px
  - right, left padding: 50px
  - bottom padding: 75px
- padding: 25px 50px;
  - top, bottom padding: 25px
  - right, left padding: 50px
- padding: 25px;
  - összes padding: 25px

# Dobozra példák

```
p {  
    border: 1px solid red; /* width style color */  
    padding: 25px 50px; /* top/bottom majd left/right */  
    margin: 25px 50px 75px 100px; /*top right bottom left*/  
}  
  
p.fontos {  
    border-style: dotted;  
    border-color: green;  
    border-width: thick;  
}
```

# Listák

list-style	<i>list-style-type list-style-position list-style-image</i>
list-style-type	none   circle   disc   square   decimal   decimal-leading-zero   lower-roman   upper-roman   lower-alpha   upper-alpha   lower-greek   lower-latin   upper-latin
list-style-image	none   url
list-style-position	inside   outside

```
ul.s {list-style-type: square;}
ol.i {list-style-type: lower-roman;}
ol.A {list-style-type: upper-alpha;}
ul {
  list-style-image: url('sqred.gif');
}
```



# Kérdések

- Mik a sütik és főként mire valók?
- Melyik címkével (tag-gel) tudunk képet behelyezni az oldalra?
- Melyik címkével tudunk linket behelyezni az oldalra?
- Melyik attribútummal adhatunk meg inline CSS formázást?
- Soroljunk fel 5, a HTML-ben speciális karaktert.
- Milyen állapotai vannak egy linknek? Adjunk egy példát, ahol CSS kód segítségével mindegyiken változtatunk valamit! (LoVe-HAte)
- Mi a különbség a class és az id között? Példával mutassuk be melyiket hogyan használjuk!
- Melyik címkével jelölünk számozott, számozatlan és melyikkel definíciós listát? Adjunk mindegyikre példát!
- CSS segítségével az előző sorszámozott felsorolás 2. elemét állítsuk félkövérre, használjunk id-t a megjelöléséhez! Ha ehhez a HTML kódban is változtatni kell valamit, azt is jelezzük!

# Kérdések

- Formázzuk az összes linket a HTML-ben úgy, hogy ha a felhasználó fölé viszi az egeret akkor ne legyen aláhúzva.
- Formázzuk az összes paragrafust (p címke) úgy, hogy a bennük szereplő karakterek színe piros legyen és középre legyenek igazítva. Mutassuk meg, hogy tennénk ezt meg szövegeközi (inline) módban, a HTML-be beágyazott módon (style címkével), és hogyan egy külső fájlból betöltve?
- Írjuk meg az alábbi táblázat HTML és CSS kódját:

1		3
4	öt	6