



PÁZMÁNY

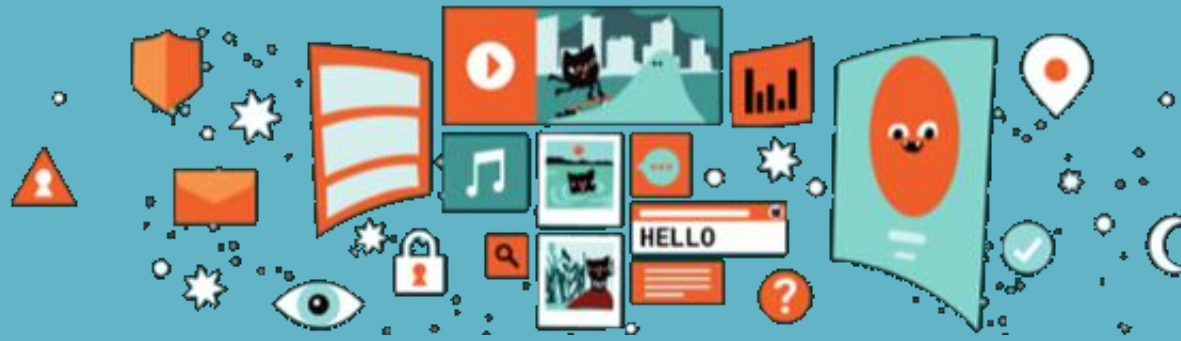
Pázmány Péter Katolikus Egyetem
Információs Technológiai és Bionikai Kar



Katolikus
Pedagógiai
Intézet

Középiskolai informatikatanárok szaktárgyi továbbképzése

2021. december 10.



PUBLIKÁLÁS A VILÁGHÁLÓN

Dr. Abonyi-Tóth Andor

Egyetemi docens

ELTE Informatikai Kar, Média- és Oktatásinformatikai Tanszék

1117 Budapest, Pázmány Péter sétány 1/C, 4.724



A TÉMA MEGJELENÉSE A KERETTANTERV BEN

9-10. ÉVFOLYAM

A 9–10. évfolyamon a digitális kultúra tantárgy alapóraszám: 102 óra.

A témakörök áttekintő táblázata:

Témakör neve	Javasolt óraszám
Algoritmizálás, formális programozási nyelv használata	25
Információs társadalom, e-Világ	3
Mobiltechnológiai ismeretek	4
Szövegszerkesztés	11
Számítógépes grafika	14
Multimédiás dokumentumok készítése	4
Online kommunikáció	4
Publikálás a világhálón	14
Táblázatkezelés	12
Adatbázis-kezelés	5
A digitális eszközök használata	6
Összes óraszám:	102

TÉMAKÖR: Publikálás a világhálón

JAVASOLT ÓRASZÁM: 14 óra

TANULÁSI EREDMÉNYEK

A témakör tanulása hozzájárul ahhoz, hogy a tanuló a nevelési-oktatási szakasz végére:

- ismeri a HTML formátumú dokumentumok szerkezeti elemeit;
- érti a CSS használatának alapelveit.

A témakör tanulása eredményeként a tanuló:

- dokumentumokat szerkeszt és helyez el tartalomkezelő rendszerben;
- több lapból álló webhelyet készít.

FEJLESZTÉSI FELADATOK ÉS ISMERETEK

- Egy webes tartalomkezelő rendszer önálló használata
- Webdokumentum szerkezetének és alapelemeinek ismerete
- Webdokumentum tartalmának és stílusának szerkesztési lehetőségei, szétválasztásuk jelentősége
- Közlésre szánt szöveges és képi információval kapcsolatos elvárások, kiválasztási szempontok, fájlformátumok
- Az internetes publikálás módszereinek megismerése, szabályai
- Szövegek, képek, fotóalbumok, hang- és videoanyagok, weblapok publikálása tartalomkezelő rendszerben
- Weblapkészítés HTML nyelven weblapszerkesztővel
- Stíluslap csatolása weblaphoz, és a benne lévő stílusok használata a dokumentum formázásához
- Összetett webdokumentum készítése

MIRŐL LESZ SZÓ?

10.15-11.00	Statikus weboldal létrehozása HTML nyelven I. (HTML5 alapstruktúra, gyakori tagek használata)
11.00-11.45	Statikus weboldal létrehozása HTML nyelven II. (Multimédiás elemek beillesztése)
11.45-12.30	Stíluslapok csatolása és módosítása I.
12.30-13.00	Ebédszünet
13.00-13.45	Stíluslapok csatolása és módosítása II.
13.45-14.30	Honlap készítése tartalomkezelő rendszerben ((W)CMS funkciók, témák/sablonok/oldalrendezések, Webhely létrehozása, téma – sablon kiválasztása, oldalstruktúra meghatározása, szöveges és multimédiás tartalom feltöltése, elrendezés kiválasztása, oldal publikálása, oldal tesztelése, javítása)



HASZNOS SEGÉDANYAGOK

TANKÖNYVEK



Letölthető kiegészítők

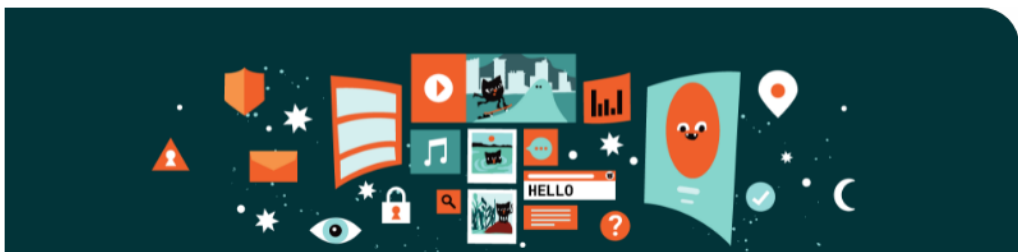
- ↓ Teljes könyv
- ↓ Tartalomjegyzék
- ↓ Tanmenet
- ↓ Forrás állományok
- ↓ Teljes könyv (EPUB formátum)
- ↓ Teljes könyv (MOBI formátum)

<https://www.tankonyvkatalogus.hu/site/kiadvany/OH-DIG09TA>

<https://www.tankonyvkatalogus.hu/site/kiadvany/OH-DIG10TA>

NKP OKOSTANKÖNYVEK

<https://www.nkp.hu/okostankonyvek>

	
PUBLIKÁLÁS A VILÁGHÁLÓN	^
FEJEZETNYITÓ	>
AZ INTERNET ÉS A WEB KAPCSOLATA	>
A WWW (VILÁGHÁLÓ) ÉPÍTŐKÖVEI	>
RESZPONZÍV WEBOLDALAK	>
WEBOLDALAK AKADÁLYMENTESSÉGE	>
KÉSZÍTSÜNK WEBLAPOT!	>
KÉSZÍTSÜNK KÖZÖSEN EGY WEBLAPOT!	>
A LEGFONTOSABB HTML5-CÍMKÉK ÖSSZEFOGLALÓ TÁBLÁZATA	>
A STÍLUSLAPOK (CSS) HASZNÁLATA	>
A STATIKUS HONLAP PUBLIKÁLÁSA	>

	
PUBLIKÁLÁS A VILÁGHÁLÓN	^
FEJEZETNYITÓ	>
ISMÉTLÉS	>
KÉSZÍTSÜNK DINAMIKUS HONLAPOT TARTALOMKEZELŐ RENDSZERREL!	>
A HONLAP LÉTREHOZÁSÁNAK ÉS PUBLIKÁLÁSÁNAK LÉPÉSEI	>
GYAKORLÁS	>

A WEBLAPKÉSZÍTÉS TECHNIKÁJA (HTML5, CSS3) ÉS ERGONÓMIÁJA TANANYAG

- <https://tamop412.elte.hu/tananyagok/weblapkeszites>

A modul leckéi, a megértéshez szükséges előismeretek, tárgyi feltételek

Leckék

1. [Bevezető](#)
2. [HTML-bevezető](#)
3. [A HTML5 legfontosabb szöveg szintű elemei](#)
4. [Kitekintés a stíluslapok használatába, tipikus szövegformázások](#)
5. [Csoportosító elemek](#)
6. [Beágyazott elemek](#)
7. [Oldalszerkezet-elemek](#)
8. [Táblázatok használata](#)
9. [Úrlapok használata](#)
10. [Interaktív elemek](#)
11. [Bevezetés a stíluslapok használatába](#)
12. [CSS kiértékelési sorrend \(rangsorolás\)](#)
13. [Ismerkedés a CSS dobozmodelljével](#)
14. [Szín- és háttérbeállítás stíluslappal](#)
15. [Vizuálisformázásmodell](#)
16. [Transzformációk és átmenetek](#)
17. [Médiatípusok használata](#)
18. [Úton a responszív arculat felé](#)

HTML-bevezető

A kód tördelése, megjegyzések a kódban

A HTML oldalak készítésénél meg kell szoknunk azt, hogy a HTML-kód tördelése (mennyi üres nem befolyásolja az oldal megjelenését. Ez alól kivételt jelent, ha például a `<pre>` taget formázott szöveggként akarunk valamit megjeleníteni.

Interaktív példa

```
1 <!DOCTYPE html>
2 <html lang="hu">
3 <head>
4   <meta charset="utf-8">
5   <title>A kód tördelése</title>
6 </head>
7 <body>
8   <p>
9     Attól, hogy          rengeteg üres helyet használok
10    és össze-vissza tördelem
11    a sorokat a kódban, azzal az oldal megjelenését nem befolyásolom. </p>
12    <!-- a pre elem a preformatted (előreformázott) szöveg rövidítése -->
13    <pre>
14      Kivéve ha a <pre> taget
15        használom, mert ekkor bizony számít,
16        hogy mi szerepel új sorban, és mennyi
17        üres helyet hagyok ki. </pre>
18  </body>
```

Pozíció: Sor 1, Kar 1 Összesen: Sor 20, Kar 547

Megtekintés Eredeti kód Kód kijelölése

HTML SZERKESZTŐ ALKALMAZÁSOK

SZERKESZTŐ KÖRNYEZETEK

Milyet válasszunk? Hasznos funkciók

- Szintaxis kiemelés
 - A címkék, paraméterek, értékek, szövegek eltérő színnel jelölése.
- Forráskód sorszámozása
 - Így egyszerűbben odaugorhatunk az adott sorszámu sorra.
- Kódkiegészítés
 - Címkék, paraméterek felajánlása gépeléskor.
 - Egyszerűbb kiválasztani valamit a listából, mintha be kellene gépelni.

ÉRETTSÉGI SZOFTVERLISTA

Kódszerkesztők:

- Visual Studio Code 1.58 - telepített kiegészítők (plugin) nélkül használható
- Notepad++ 8.1.4

SZERKESZTŐ KÖRNYEZETEK

Érettségi szoftverlista



Visual Studio Code

Notepad ++	Visual Studio Code
https://notepad-plus-plus.org/	https://code.visualstudio.com/
Windows	macOS, Windows, Linux

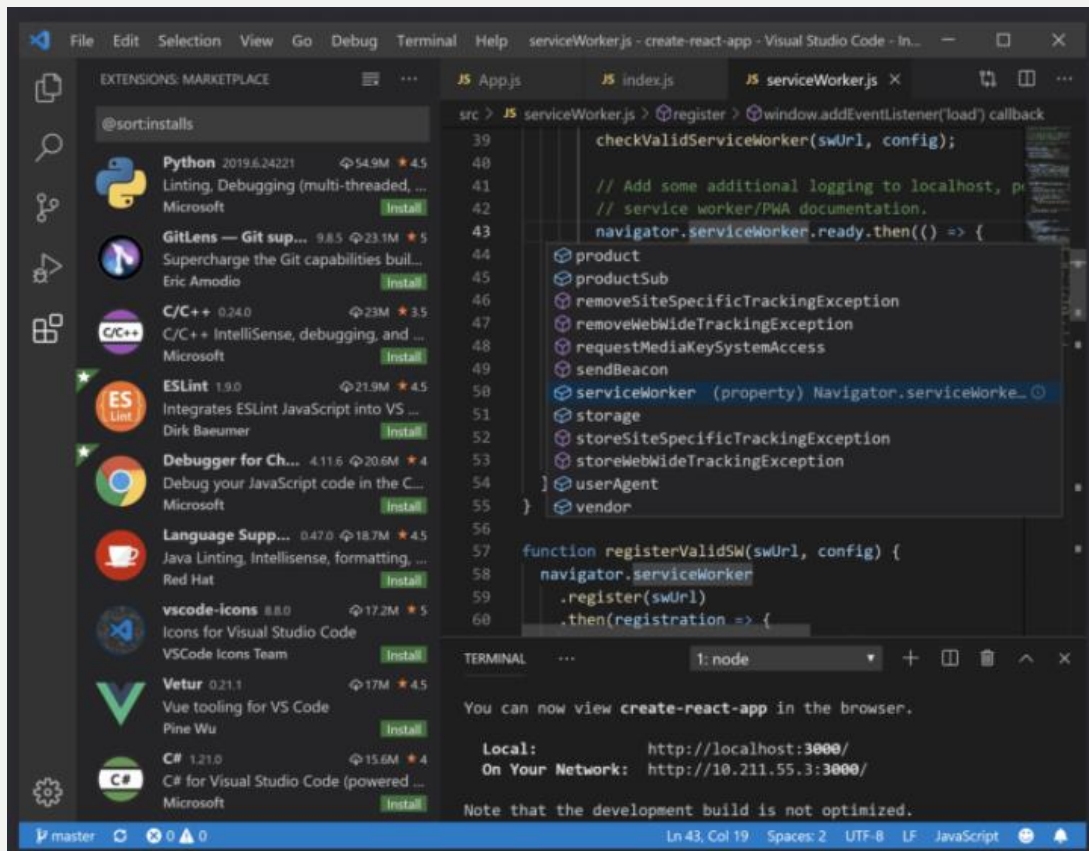
```
*new 1 - Notepad++
File Edit Search View Encoding Language Settings Tools Macro Run Plug
new 1 new 2
1 <xml>
2   <details>
3     <name>John L</name>
4     <age>24</age>
5     <location>Sweden</location>
6   </details>
7 </xml>
```

```
File Edit Selection View Go Run Terminal Help v.html - temp - Visual Studio
EXPLORER
> OPEN EDITORS
> TEMP
> OUTLINE
> NPM SCRIPTS
v.html
1 <!DOCTYPE html>
2 <html lang="en" dir="ltr">
3 <head>
4   <meta charset="utf-8">
5   <title></title>
6 </head>
7 <body>
8   <<param name="" value="">
9 </body>
10 </html>
11
```

AJÁNLOTT SZERKESZTŐKÖRNYEZET

Visual Studio Code (VS Code)

- <https://code.visualstudio.com/>



Visual Studio Code

Több platformon is használható

- macOS
- Windows
- Linux



VISUAL STUDIO CODE (VS CODE)

VISUAL STUDIO CODE (VS CODE)

Telepítés

<https://code.visualstudio.com/>

 Visual Studio Code Docs Updates Blog API Extensions FAQ [Download](#)


[↓ Windows](#)
Windows 7, 8, 10


[↓ .deb](#)
Debian, Ubuntu

[↓ .rpm](#)
Red Hat, Fedora, SUSE


[↓ Mac](#)
macOS 10.10+

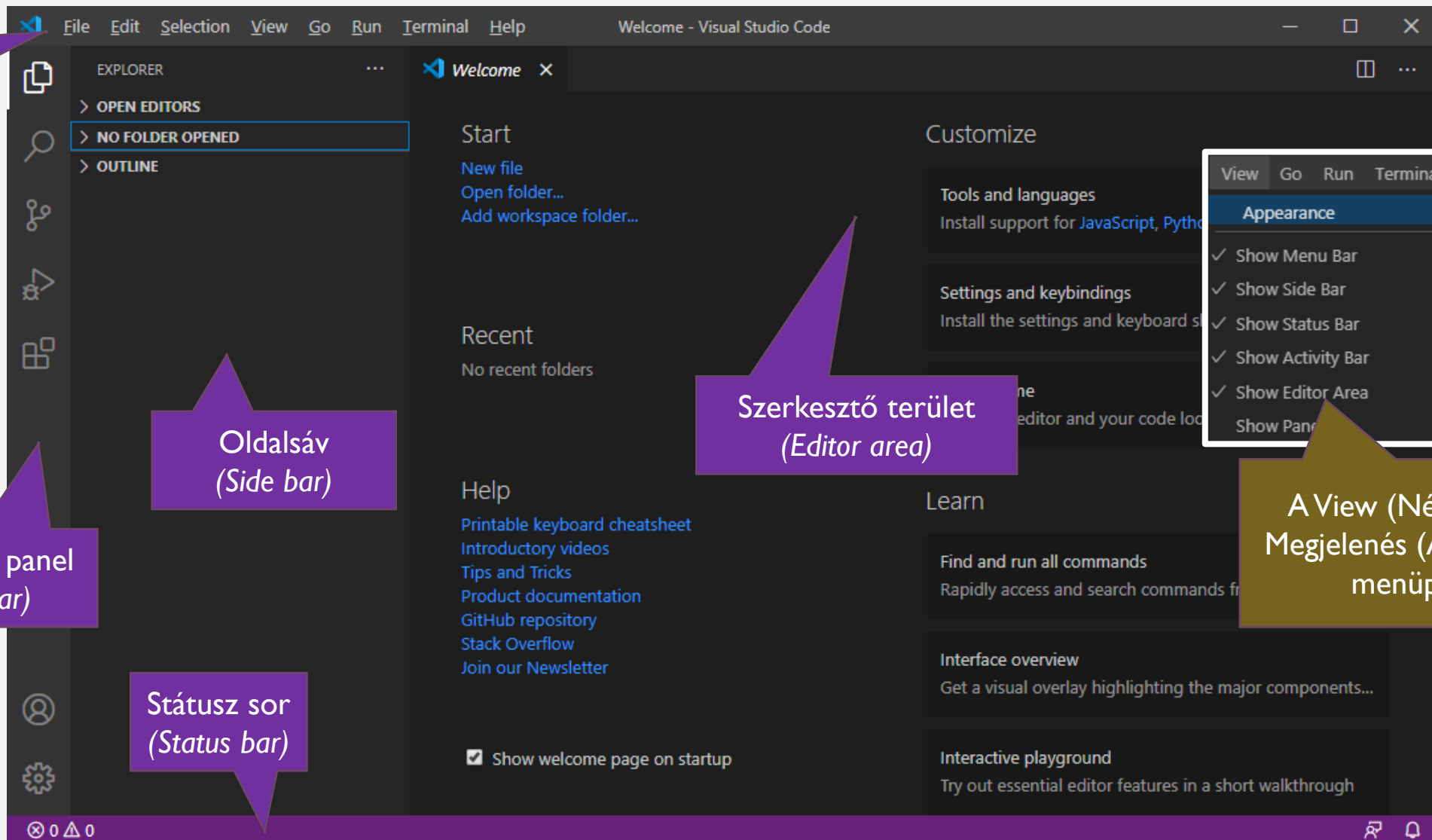
User Installer	64 bit	32 bit	ARM
System Installer	64 bit	32 bit	ARM
.zip	64 bit	32 bit	ARM

.deb	64 bit
.rpm	64 bit
.tar.gz	64 bit

[Snap Store](#)

INDÍTÁS UTÁNI KEZDŐKÉPERNYŐ

Menüsor
(Menu bar)



BEÉPÍTETT SÚGÓ RÁMUTATÁSKOR

```
File Edit Selection View Go Run Terminal Help • index.html - demo_honlap - Visual Studio Code
```

EXPLORER

OPEN EDITORS 1 UNSAVED

DEMO_HONLAP

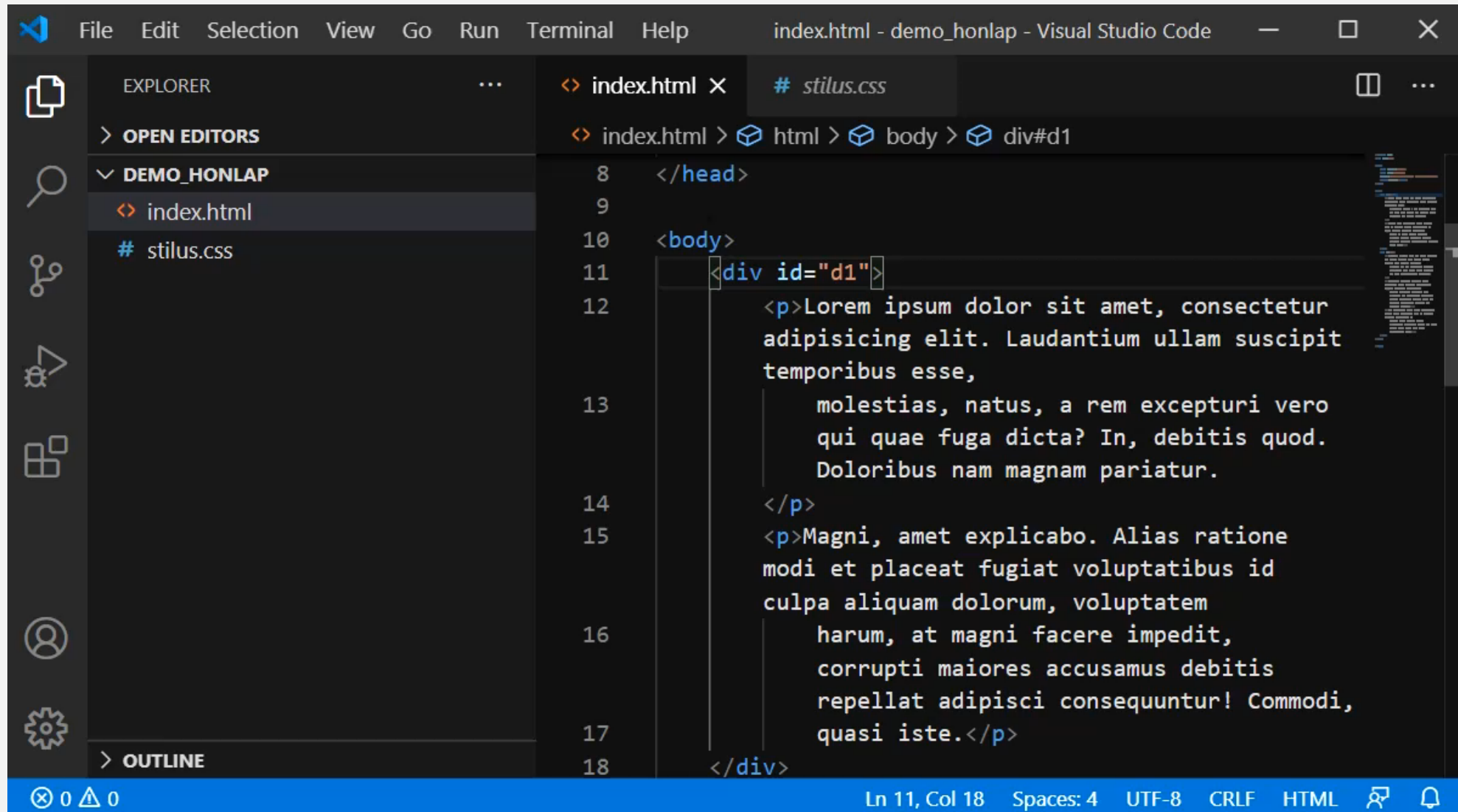
- index.html

index.html

```
4 <head>
5   <meta charset="UTF-8">
6   <meta name="viewport" content="width=device-width">
7   <title>Document</title>
8 </head>
9
10 <body>
11   <p title="magyarázat" id="p1">szöveg</p>
12 </body>
13
14 </html>
```

Ln 11, Col 5 Spaces: 4 UTF-8 CRLF HTML

KÓD ÖSSZECSUKÁSA

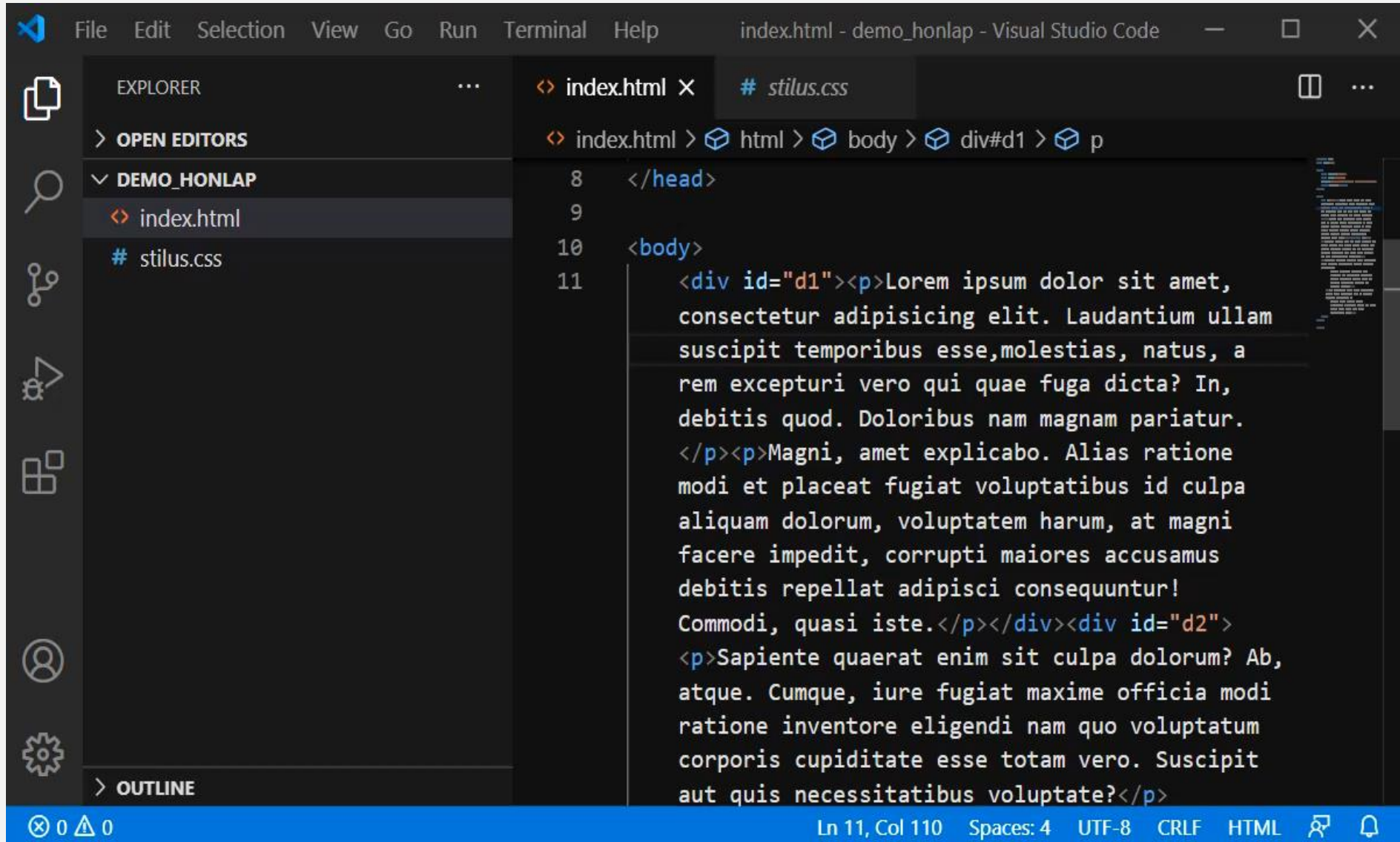


The screenshot shows the Visual Studio Code interface with the following details:

- Explorer Panel:** Shows a project named "DEMO_HONLAP" containing two files: "index.html" and "stilus.css".
- Editor Panel:** Displays two open files: "index.html" and "stilus.css". The "index.html" file is active, showing the following code:

```
8 </head>
9
10 <body>
11   <div id="d1">
12     <p>Lorem ipsum dolor sit amet, consectetur
      adipiscing elit. Laudantium ullam suscipit
      temporibus esse,
13       molestias, natus, a rem excepturi vero
      qui quae fuga dicta? In, debitis quod.
      Doloribus nam magnam pariat.
14     </p>
15     <p>Magni, amet explicabo. Alias ratione
      modi et placeat fugiat voluptatibus id
      culpa aliquam dolorum, voluptatem
16       harum, at magni facere impedit,
      corrupti maiores accusamus debitis
      repellat adipisci consequuntur! Commodi,
17     quasi iste.</p>
18   </div>
```
- Terminal Panel:** Empty.
- Status Bar:** Shows "Ln 11, Col 18", "Spaces: 4", "UTF-8", "CRLF", "HTML", and icons for search, run, and help.

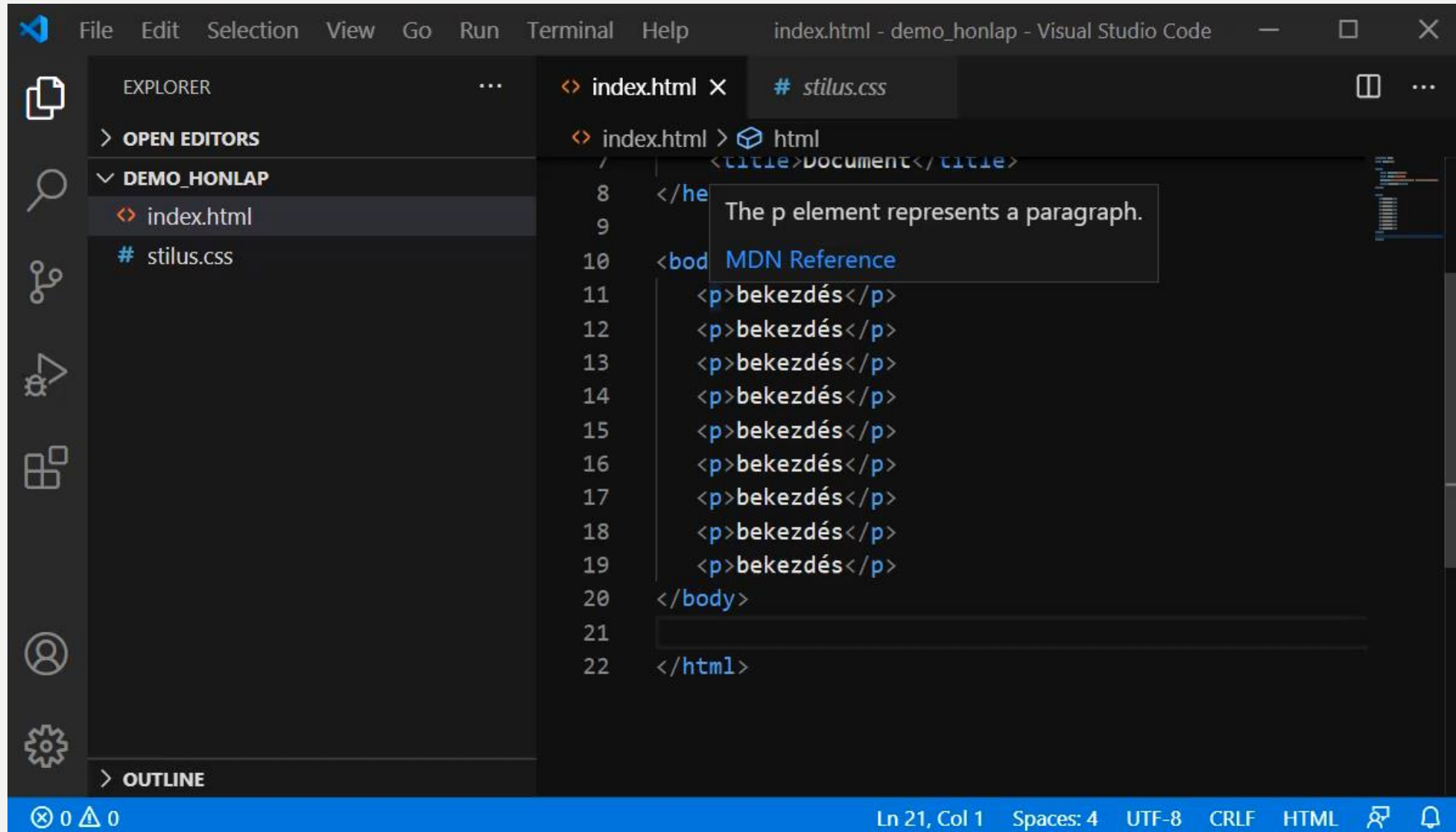
AUTOMATIKUS KÓDFORMÁZÁS

A screenshot of the Visual Studio Code editor interface. The top menu bar includes File, Edit, Selection, View, Go, Run, Terminal, and Help. The title bar shows 'index.html - demo_honlap - Visual Studio Code'. The Explorer sidebar on the left shows a project named 'DEMO_HONLAP' with files 'index.html' and 'stilus.css'. The 'index.html' file is open in the editor, and the 'stilus.css' file is selected in the Explorer. The editor window shows the CSS file content, which is being formatted automatically. The code is as follows:

```
8 </head>
9
10 <body>
11   <div id="d1"><p>Lorem ipsum dolor sit amet,
    consectetur adipiscing elit. Laudantium ullam
    suscipit temporibus esse,molestias, natus, a
    rem excepturi vero qui quae fuga dicta? In,
    debitis quod. Doloribus nam magnam pariatur.
    </p><p>Magni, amet explicabo. Alias ratione
    modi et placeat fugiat voluptatibus id culpa
    aliquam dolorum, voluptatem harum, at magni
    facere impedit, corrupti maiores accusamus
    debitis repellat adipisci consequuntur!
    Commodi, quasi iste.</p></div><div id="d2">
    <p>Sapiente quaerat enim sit culpa dolorum? Ab,
    atque. Cumque, iure fugiat maxime officia modi
    ratione inventore eligendi nam quo voluptatum
    corporis cupiditate esse totam vero. Suscipit
    aut quis necessitatibus voluptate?</p>
```

The status bar at the bottom shows 'Ln 11, Col 110', 'Spaces: 4', 'UTF-8', 'CRLF', 'HTML', and icons for search, run, and help.

TÖBBSZÖRÖS KURZOR HASZNÁLATA

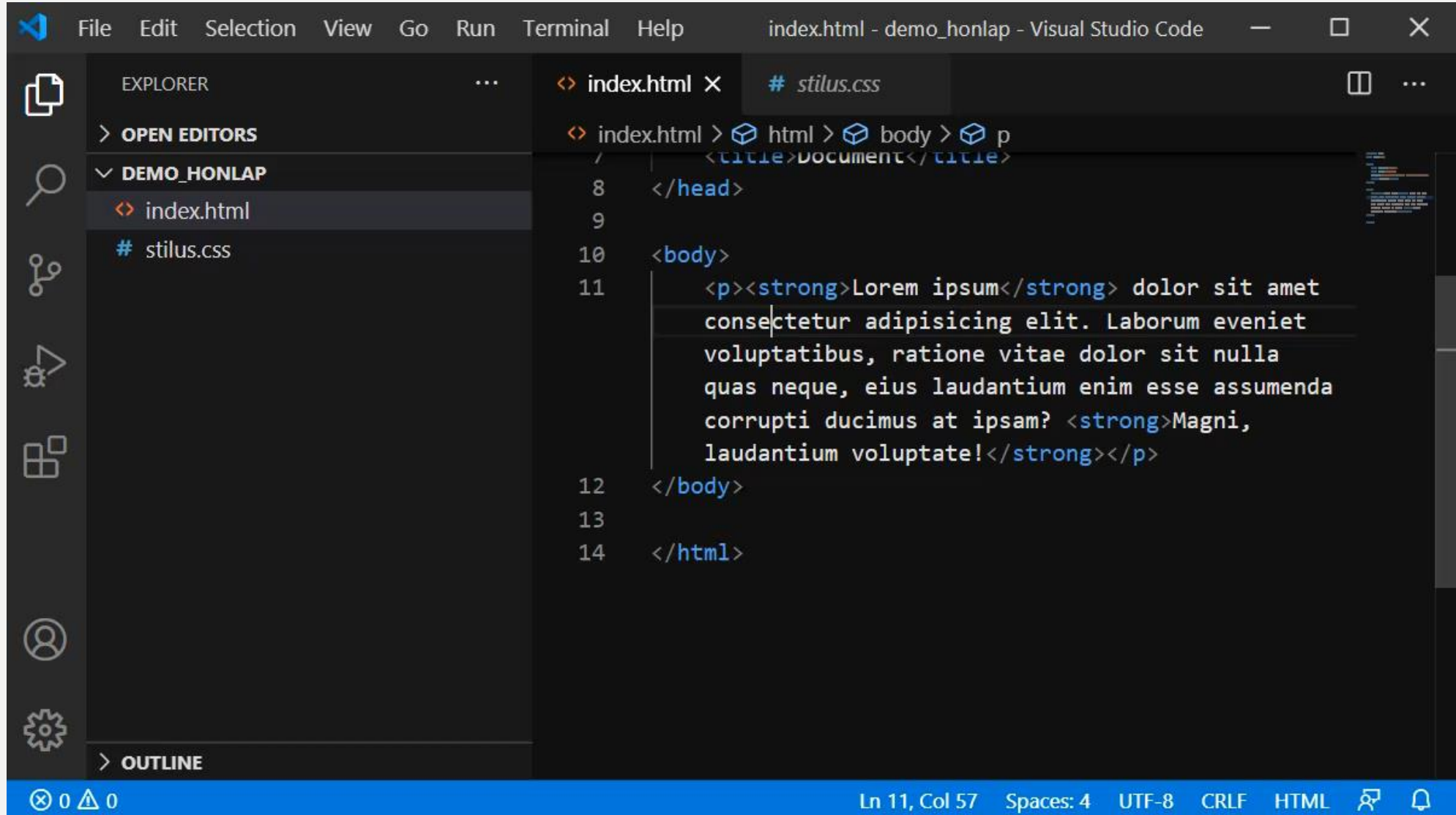


The screenshot shows the Visual Studio Code interface with a dark theme. The Explorer sidebar on the left shows a project named 'DEMO_HONLAP' with two files: 'index.html' and 'stilus.css'. The main editor area has two tabs: 'index.html' and 'stilus.css'. The 'index.html' tab is active, showing an HTML document. The cursor is positioned at line 21, column 1. A tooltip is visible over the cursor, displaying the text 'The p element represents a paragraph.' and a link to 'MDN Reference'. The HTML code in the editor is as follows:

```
1 <!DOCTYPE html>
2 <html>
3   <head>
4     <title>Document</title>
5   </head>
6   <body>
7     <p>bekezdés</p>
8     <p>bekezdés</p>
9     <p>bekezdés</p>
10    <p>bekezdés</p>
11    <p>bekezdés</p>
12    <p>bekezdés</p>
13    <p>bekezdés</p>
14    <p>bekezdés</p>
15    <p>bekezdés</p>
16    <p>bekezdés</p>
17    <p>bekezdés</p>
18    <p>bekezdés</p>
19    <p>bekezdés</p>
20  </body>
21
22 </html>
```

The status bar at the bottom indicates 'Ln 21, Col 1', 'Spaces: 4', 'UTF-8', 'CRLF', 'HTML', and shows icons for search, run, and settings.

TÖBB CÍMKE KIJELÖLÉSE, MÓDOSÍTÁSA



The screenshot shows the Visual Studio Code interface with a dark theme. The Explorer sidebar on the left shows a project named 'DEMO_HONLAP' with two files: 'index.html' and 'stilus.css'. The main editor area has two tabs: 'index.html' and 'stilus.css'. The 'index.html' tab is active, showing the following HTML code:

```
1 <!DOCTYPE html>
2 <html>
3   <head>
4     <title>Document</title>
5   </head>
6   <body>
7     <p><strong>Lorem ipsum</strong> dolor sit amet
8     consectetur adipiscing elit. Laborum eveniet
9     voluptatibus, ratione vitae dolor sit nulla
10    quas neque, eius laudantium enim esse assumenda
11    corrupti ducimus at ipsam? <strong>Magni,
12    laudantium voluptate!</strong></p>
13  </body>
14 </html>
```

The 'stilus.css' tab is also visible, showing a single line of CSS code: '# stilus.css'. The status bar at the bottom indicates the current position is 'Ln 11, Col 57', with 'Spaces: 4', 'UTF-8', 'CRLF', and 'HTML' encoding. There are also icons for error, warning, and information.



ESETTANULMÁNY KÖZÖS ELKÉSZÍTÉSE

A Golden Retriever bemutatása

Kezdőlap [Típusok](#)

A **Golden Retriever** egy kedves, barátságos kutyafajta, mely Skóciából származik. A fajta tudományos megnevezése: Golden Retriever - *Canis lupus familiaris*.

Előnyös tulajdonságai



Okos, könnyen képezhető



Szelíd, családbarát



Nagyon jó vadászkutya



Fontos! Ha gazdi szeretné lenni, nézz utána, hogy a *felelős* kutyatartással kapcsolatban milyen ajánlások és kötelezettségek vannak!

Goldi, a kutyám



Fotók Goldiról

Ha szeretnél még több kutyás fotót nézegetni, látogasd meg a [Pixabay portál - Goldenek](#) fotó oldalát!

A videón is láthatod, hogy Goldi nagyon szeret a földön hemperegni, amely miatt általában koszosan jövünk haza a sétából.



Goldi hempereg

[Vissza az oldal tetejére](#)

A Golden Retriever bemutatása

[Kezdőlap](#) Típusok

Típusok

A Golden Retriever típusai	
Típus	Típus leírása
Munkatípus	Könnyedebb felépítésű. Vékonyabb, hosszabb fej, sötétebb szőrszín, élénk temperamentum, karcsúbb, ruganyosabb, gyorsabb.
Kiállítási típus	Dúsabb, hosszabb szőrzet, robusztusabb felépítés, világosabb szőrszín, határozottabb.
Európai típus	Szőrszíne az arany világosabb árnyalata. Rövidebb szőrű, mint az amerikai. Lába rövidebb az amerikaiénál, így megjelenése zömökebb. Jól pigmentált. Jó szögellésű elől, hátul.
Amerikai típus	Orr és fej része finomabb. Színe középarany, rézvörös, de sosem krém. Mellső szögellései szerényebbek, hátul olykor túlszögelt.

Forrás: hu.wikipedia.org/wiki/Golden_retriever

[Vissza az oldal tetejére](#)

SZÜKSÉGES ÁLLOMÁNYOK LETÖLTÉSE

**`http://tiny.cc/
pubvilaghalo2021`**



HTML ALAPOK

A HTML ALAPJAI

Mi a HTML?

- *HyperText Markup Language* – Hiperszöveg jelölőnyelv
- Leírónyelv (nem programozási nyelv)
 - A honlap tartalmának leírására szolgál.
 - Szöveges állomány, .html kiterjesztéssel.
 - A tartalmat címkék (tagek/elemek) segítségével írhatjuk le. pl. `<p>`, `<h1>`
 - A címkéket a böngészőprogram értelmezi.
 - A HTML5 a legújabb szabvány (2014-ben jelent meg).

A HTML ALAPJAI

A HTML címkék (tagek / elemek)

- **<** jellel kezdődnek, és **>** jellel végződnek.
- **Kétféle típus** létezik: páros, páratlan (üres)
 - A **páros** elemeknek van nyitó és záró tagja.
 - A címke hatása a nyitó tagtól a záró tagig tart.
 - A záró címke, ugyanaz mint a nyitó, csak egy / jel vezeti be.
 - A nyitó és záró címke közé kerül a tartalom.
 - A **páratlan** (üres) elemeknek nincs záró címkéjük.

Bekezdés

<p> **</p>**

Sortörés

**
**

AUTOMATIKUS CÍMKE LEZÁRÁS



Visual Studio Code

```
index.html - demo_honlap - Visual Studio Code

EXPLORER
> OPEN EDITORS
DEMO_HONLAP
  index.html

index.html X
  index.html > html > body
  1 <!DOCTYPE html>
  2 <html lang="en">
  3
  4 <head>
  5   <meta charset="UTF-8">
  6   <meta name="viewport"
  7     content="width=device-width, initial-scale=1.0">
  8   <title>Document</title>
  9 </head>
 10 <body>
 11
 12 </body>
 13
 14 </html>
```

Ln 11, Col 5 Spaces: 4 UTF-8 CRLF HTML

A HTML ALAPJAI

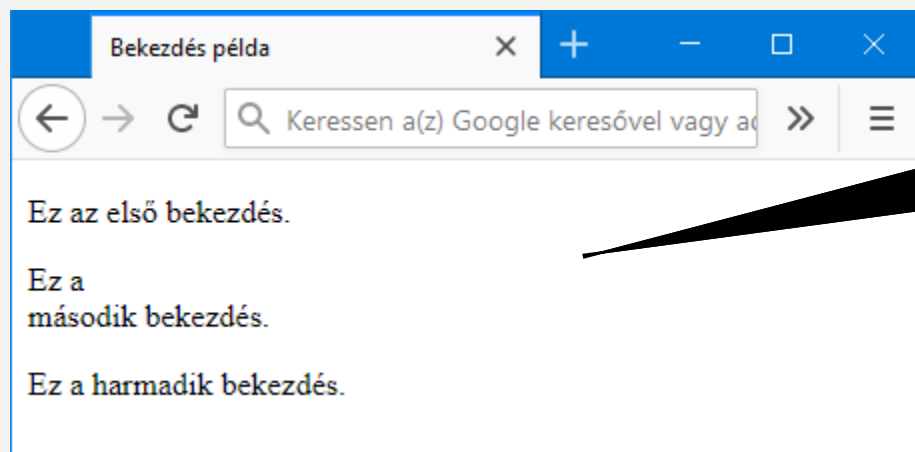
Példa bekezdésre és sortörésre

Példa

`<p>Ez az első bekezdés.</p>`

`<p>Ez a <br> második bekezdés.</p>`

`<p>Ez a harmadik bekezdés.</p>`



A böngészőprogram így jeleníti meg a fenti kódot.

A HTML ALAPJAI

Címkék egymásba ágyazása

- A címkéket egymásba ágyazhatjuk, de mindig azt a címkét kell később lezárni, amelyet korábban nyitottunk meg.

Példa egymásba ágyazásra

`<b><i>Félkövér és dőlt szöveg</i></b>`

A `<b>` tag félkövér, az `<i>` tag dőlt betűs formázást jelent.

A lezárás fordított sorrendben történik.

Félkövér és dőlt szöveg

A böngészőben megjelenő eredmény

CSOPORTOSÍTÁS A MEGJELENÉS MÓDJA ALAPJÁN

Blokkszintű és sorbeli (inline) elemek

- A **blokkszintű** elemek előtt és mögött térköz (margó) található.
 - Blokkszintű elem például a **<p>**, vagyis a bekezdés. Minden bekezdés új blokkba kerül, az egyes blokkok egymás alá kerülnek, közöttük üres térköz látható.
- A **sorbeli** (*inline*) elemek soron belül jelennek meg, nem töri a böngésző külön blokkokba azokat.
 - Sorbeli elem például a ****, amely a szöveget félkövéren jeleníti meg.

Ez egy bekezdés.

Ez egy bekezdés.

A bekezdésen belül van egy félkövér szöveg.

A későbbiekben ezzel a témakörrel még részletesebben foglalkozunk, amikor megismerkedünk a Dobozmodellel.

A HTML ALAPJAI

Megjegyzés

- Megjegyzést a `<!--` és `-->` jelek között helyezhetünk el.
- Nem jelenik meg a böngészőben.

Példa egy soros megjegyzésre

```
<!-- Egy soros megjegyzés -->
```

Példa több soros megjegyzésre

```
<!--  
Ez egy  
több sorból álló  
megjegyzés  
-->
```

A HTML ALAPJAI

HTML attribútumok (paraméterek)

- A címkékhez attribútumokat (paramétereket) is rendelhetünk.
- A paramétereknek értékük is lehet.

Példa több paraméter használatára

```
<p lang="en" title="Hiperszöveg jelölőnyelv">HyperText Markup Language</p>
```

HyperText Markup Language

Hiperszöveg jelölőnyelv

Ebben a példában azt adtuk meg, hogy a bekezdésben olvasható szöveg angol nyelvű (`lang="en"`). A magyarázó szöveg (`title`) majd akkor jelenik meg, ha az elem fölé visszük az egeret.

Példa logikai paraméter használatára

```
<p hidden>Bekezdés</p>
```

Ha a paraméternek van értéke, akkor azt idézőjelek, vagy aposztrófok közé kell tennünk.
Ha több paraméter is van, azokat szóközzel választjuk el.

Vannak logikai paraméterek, amelyeket önállóan (érték megadás nélkül) használunk.
A `hidden` például azt jelenti, hogy az elem rejtetté válik.

PARAMÉTEREK FELAJÁNLÁSA, SZÍNVÁLASZTÁS



Visual Studio Code

```
File Edit Selection View Go Run Terminal Help • index.html - demo_honlap - Visual Studio Code

EXPLORER
> OPEN EDITORS 1 UNSAVED
▼ DEMO_HONLAP
  <> index.html

index.html
4 <head>
5   <meta charset="UTF-8">
6   <meta name="viewport"
7     content="width=device-width, initial-scale=1.0">
8   <title>Document</title>
9 </head>
10 <body>
11
12 </body>
13
14 </html>
```

Ln 11, Col 5 Spaces: 4 UTF-8 CRLF HTML



A HTML OLDAL ALAP ELÉPÍTÉSE

A HTML5 OLDAL ALAP FELÉPÍTÉSE

HTML5 dokumentumtípus

```
<!DOCTYPE html>
```

Az oldal legfelső szintű eleme

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

Az oldal nyelve magyar (hu)

Az oldal címe

```
<title>Oldalcím</title>
```

Az oldal karakterkódolása

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

```
</head>
```

Fej (fejléc)

A tartami rész ide kerül

```
<body>
```

```
<!-- tartalom -->
```

```
</body>
```

```
</html>
```

Törzs

HTML ALAPSTRUKTÚRA MEGADÁSA



Visual Studio Code

The screenshot shows the Visual Studio Code editor with a file named 'index.html' open. The cursor is at line 1, column 1, where the text 'html' has been typed. A context menu is displayed below the text, showing three options: 'html', 'html:5' (which is highlighted), and 'html:xml'. The text 'Emmet Abbreviation' is visible to the right of the 'html:5' option.

Gépeljük be a html:5 kódot, majd nyomunk meg a tabulátor vagy az enter billentyűt! Sőt, még ennyit sem kell gépelnünk, a ! tabulátor is megfelelő.

The screenshot shows the Visual Studio Code editor with the 'index.html' file. The code is expanded to show the full HTML boilerplate structure. The code is as follows:

```
1 <!DOCTYPE html>
2 <html lang="en">
3   <head>
4     <meta charset="UTF-8">
5     <meta name="viewport" content="width=device-width, initial-scale=1.0">
6     <title>Document</title>
7   </head>
8   <body>
9     |
10  </body>
11 </html>
```

Most egy úgynevezett Emmet rövidítést használtunk. Sok ilyen rövidítés áll rendelkezésre a programban.

Az Emmet nem más mint beépülő modulok összessége, amelyek támogatják a gyors és hatékony kódolást a kódszerkesztő alkalmazásokban.

A HTML OLDAL ALAP FELÉPÍTÉSE

<!DOCTYPE>: Dokumentumtípus

- A kód első sorába a dokumentumtípust kell írni.
- Ez határozza meg, hogy melyik szabvány, melyik altípusa szerint dolgozunk.
- Páratlan elem.
- HTML5 esetén a következőt kell megadnunk:

<!DOCTYPE html>

A HTML OLDAL ALAP FELÉPÍTÉSE

`<html>`: Az oldal legfelső szintű eleme

- Minden más elemet ezen címkén belül kell elhelyezni.
- Páros elem.
- Az oldal nyelvét paraméterrel célszerű beállítani.

`<html lang="hu">`

– Miért fontos?

Példák nyelvkódokra

hu : magyar, en: angol,

en-us: amerikai angol, en-gb : brit angol

- Látássérült felhasználók képernyőolvasó programja ezen a nyelven olvassa fel a szöveget.
- A keresőprogramok a nyelv alapján is indexelik az oldalakat.
- A beállított nyelv befolyásolhatja a böngésző által alkalmazott betűtípust.

A HTML OLDAL ALAP FELÉPÍTÉSE

<head>: Az oldal fejléce

- Páros elem.
- Háttérinformációkat tartalmaz az oldalra vonatkozóan:
 - Oldal címe (**kötelező!**)
 - Metainformációk
 - A megjelenésért felelős stíluslapok és/vagy azokra történő hivatkozás (később részletesen megnézzük)
 - Kliens oldali szkriptek (pl. javascript)

AZ OLDAL FEJLÉCE

`<title>`: Az oldal címe

- Páros elem. **Kötelezően kitöltendő!**
- A böngésző címsorában, vagy a lapfülön jelenik meg.
- Fontos, hogy utaljon a tartalomra.
- Ha valaki a könyvjelzőbe / kedvencek közé teszi az oldalt, akkor a listában alapesetben ez a szöveg jelenik meg.
- Látássérült emberek ez alapján azonosíthatják a különböző böngészőfüleket vagy ablakokat.

A CÍM MÓDOSÍTÁSA

- Legyen a kezdőlap (index.html) címe a következő:
Golden Retrieverek - Kezdőlap

AZ OLDAL FEJLÉCE

<meta>: Metainformációk

- Páratlan elem.
- Használatával megadható a karakterkódolás, szerző neve, oldal leírása stb.

Példa karakterkódolás beállítására

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

Példa egyéb metainformációk beállítására

```
<meta name="description" content="Leírás">  
<meta name="keywords" content="Kulcsszó1, Kulcsszó2">  
<meta name="author" content="Szerző neve">
```

Olyan karakterkódolást kell választanunk, hogy a speciális karakterek (ékezetes betűk stb.) megfelelően jelenjenek meg.

A HTML ALAP FELÉPÍTÉSE

<body>: Dokumentumtörzs

- A HTML oldal tartalmi részét fogja körbe.
- Páros elem.
- Ebbe helyezzük el a tényleges tartalmat, címsorokat, bekezdéseket, listákat, médiaelemeket stb.

SZÖVEG BEMÁSZOLÁSA

1. Helyezzük el a `kezdolap.txt` állományban található szöveget az `index.html` állomány törzsében!
2. Nézzük meg az eredményt egy böngésző-programban!



CSOPORTOSÍTÓ ÉS OLDALSZERKEZET ELEMEEK

CÍMSOROK

<h1>, ..., <h6>: 1-es címsor, ..., 6-os címsor

- Páros, blokkszintű elemek.
- Segítségükkel adhatjuk meg az oldal struktúráját.
- Hasonló szerepet töltenek be, mint egy kiadványban a fejezetcímek.
- Az egyes szintű címsornak (**<h1>**) általában az oldal címét állítják be.

Példa

<h1>A HTML alapjai**</h1>**

A HTML alapjai

CÍMSOROK

`<h1>`, ..., `<h6>`

- Az első szintű címsornak (`<h1>`) lehet alcímsora (`<h2>`), amelynek szintén lehet alcímsora (`<h3>`), és így tovább, egészen a hatos szintű alcímsorig (`<h6>`).

Példa

```
<h1>Címsor 1</h1>
<h2>Címsor 2</h2>
<h3>Címsor 3</h3>
<h4>Címsor 4</h4>
<h5>Címsor 5</h5>
<h6>Címsor 6</h6>
```

Címsor 1

Címsor 2

Címsor 3

Címsor 4

Címsor 5

Címsor 6

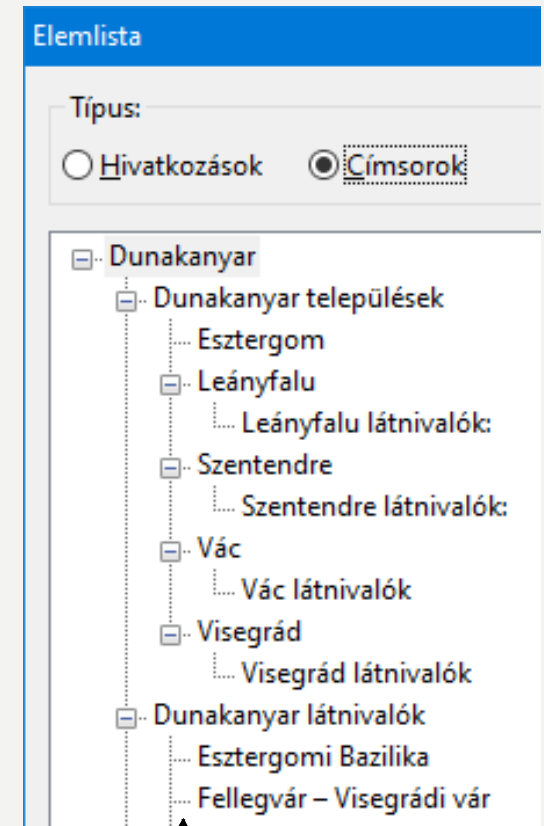
A böngészőben félkövér betűkkel jelennek meg a címsorok.

Minél magasabb szintű a címsor, annál nagyobb a mérete.

CÍMSOROK

<h1>, ..., <h6>

- Nem csak a látók számára könnyítik meg a tájékozódást, hanem a látássérült emberek számára is.
 - A képernyőolvasó programokban lapon belül a címsorokhoz lehet ugrani. Akár csak egy tartalomjegyzékben.
- Keresőoptimalizálási (SEO) szempontból is fontos, hogy megfelelő címsorszinteket használjunk!



Az NVDA képernyőolvasóban a címsorok külön ablakban jelennek meg és az adott oldalrészre lehet ugrani.

CÍMSOROK

Alakítsuk ki a címsorokat!

1-es
címsor

2-es
címsor

2-es
címsor

A Golden Retriever bemutatása

Kezdőlap [Típusok](#)

A **Golden Retriever** egy kedves, barátságos kutyafajta, mely Skóciából származik. A fajta tudományos megnevezése: Golden Retriever - *Canis lupus familiaris*.

Előnyös tulajdonságai

-  Okos, könnyen képezhető
-  Szelíd, családbarát
-  Nagyon jó vadászkutya

 **Fontos!** Ha gazdi szeretnél lenni, nézz utána, hogy a *felelős* kutyatartással kapcsolatban milyen ajánlások és kötelezettségek vannak!

Goldi, a kutyám



Fotók Goldiról

Ha szeretnél még több kutyás fotót nézegetni, látogasd meg a [Pixabay portál - Goldenek](#) fotó oldalát!

A videón is láthatod, hogy Goldi nagyon szeret a földön hemperegni, amely miatt általában koszosan jövünk haza a sétából.



[Vissza az oldal tetejére](#)

BEKEZDÉS

<p>: Bekezdés

- Páros, blokkszintű elem.
- Segítségével a szöveget bekezdésekre tagolhatjuk.
- A bekezdés előtt és mögött üres térközt hagy a böngésző alapesetben.
 - Ennek mértékét stíluslappal állítjuk be a későbbiekben.

Példa

```
<p>Ez az első bekezdés.</p>  
<p>Ez a második bekezdés.</p>
```

Ez az első bekezdés.

Ez a második bekezdés.

SORTÖRÉS

: Sortörés

- Páratlan, sorbeli elem.
- A beillesztett helyen egy sortörést fog alkalmazni a böngésző.
- Helytelen használat:
 - Ne használjuk térköz növelésére, ezt majd stíluslappal oldjuk meg.
 - Ne használjunk indokolatlan sortöréseket a bekezdéseken belül. Ahelyett hogy egy hosszú szöveget sortörésekkel osztanánk kisebb egységekre, alakítsunk ki több bekezdést!

Példa

<p>Nagy Imola
egyéni vállalkozó
Debrecen</p>

Nagy Imola
egyéni vállalkozó
Debrecen

BEKEZDÉS ÉS SORTÖRÉS

1. Alakítsuk ki a bekezdéseket!
2. Használjunk sortörést a „származik.” szöveg után!

A Golden Retriever bemutatása

Kezdőlap [Típusok](#)

A **Golden Retriever** egy kedves, barátságos kutyafajta, mely Skóciából származik. A fajta tudományos megnevezése: Golden Retriever - *Canis lupus familiaris*.

Előnyös tulajdonságai

-  Okos, könnyen képezhető
-  Szelíd, családbarát
-  Nagyon jó vadászkutya

 **Fontos!** Ha gazdi szeretné lenni, nézz utána, hogy a *felelős* kutyatartással kapcsolatban milyen ajánlások és kötelezettségek vannak!

Goldi, a kutyám



Ha szeretné még több kutyás fotót nézegetni, látogasd meg a [Pixabay portál - Goldenek](#) fotó oldalát!

A videón is láthatod, hogy Goldi nagyon szeret a földön hemperegni, amely miatt általában koszosan jövünk haza a sétából.



[Vissza az oldal tetejére](#)



SZÖVEGFORMÁZÁSOK

FÉLKÖVÉR SZÖVEG

`<b>`: Félkövér szöveg

- Páros, soron belüli elem.
- Félkövér stílussal formázza a szöveget.
- Főleg kulcsszavak kiemelésére használjuk.
 - A kulcsszavak segítik a látogatókat, hogy könnyen áttekinthessék a szöveget.

Példa

`<p>`Cikkünkben a `<b>`HTML5 újdonságait`</b>` mutatjuk be.`</p>`

Cikkünkben a **HTML5 újdonságait** mutatjuk be.

ERŐSEN KIEMELT SZÖVEG

`<strong>`: Erősen kiemelt szöveg

- Páros, soron belüli elem.
- Nem csak félkövér kiemelést jelent.
 - A látássérültek által használt képernyőolvasó programokban is hatása van/lehet.

Példa

`<p><strong>Figyelem!</strong> November 30-án lejár a jelentkezési határidő.</p>`

Figyelem! November 30-án lejár a jelentkezési határidő.

DŐLT SZÖVEG

<i>: Dőlt szöveg

- Páros, soron belüli elem.
- Akkor használjuk, ha
 - Egy szöveg eltér a szöveg többi részétől (hangulatban, hangnemben).
 - Szakkifejezések, idegen nyelven leírt gondolatokat jelöljük.
 - Élőlények latin nevét írjuk le.

Példa

<p>Tudtad, hogy az <i>Agathidium vaderi</i> bogárfajt Darth Vaderről nevezték el?</p>

Tudtad, hogy az *Agathidium vaderi* bogárfajt Darth Vaderről nevezték el?

HANGSÚLYOS SZÖVEG

****: Hangsúlyos szöveg

- Páros, soron belüli elem.
- Nem csupán dőlt kiemelést jelent, hanem erős hangsúlyozást.
- Hatására akár a szövegrész értelme is megváltozhat.

Példa

<p>Ma történt velem.</p>

<p>Ma történt velem.</p>

Ma történt velem.

Ma történt *velem*.

KIEMELÉSEK HASZNÁLATA

- Használjunk kiemeléseket a szöveg megfelelő helyein!

A Golden Retriever bemutatása

Kezdőlap [Típusok](#)

A **Golden Retriever** egy kedves, barátságos kutyafajta, mely Skóciából származik.
A fajta tudományos megnevezése: Golden Retriever - *Canis lupus familiaris*.

Előnyös tulajdonságai

-  Okos, könnyen képezhető
-  Szelíd, családbarát
-  Nagyon jó vadászkutya

**Fontos!** Ha gazdi szeretnéd lenni, nézz utána, hogy a **felelős** kutyatartással kapcsolatban milyen ajánlások és kötelezettségek vannak!

LISTÁK BEILLESZTÉSE

A LISTÁKRÓL

- A listák jól használhatók az információk szervezésére.
- A képernyőolvasó programokban is jól lehet navigálni a listaelemek között.
- Egyszerű, könnyen áttekinthető struktúrát biztosítanak.

A táborba hozd magaddal a következőket: hálózás, zseblámpa, fürdőruha.



A táborba hozd magaddal a következőket:

- hálózás
- zseblámpa
- fürdőruha

Jobban pártázható, áttekinthető lett a felsorolás.



FELSOROLÁS LISTÁK, SORSZÁMOZOTT LISTÁK

FELSOROLÁS LISTA

`<ul><li>`: Felsorolás lista

Felsorolás lista létrehozásakor az `<ul>` és `</ul>` tagek között helyezzük el a listaelemeket, a `<li>` és `</li>` tagek között.

Példa

`<p>`A táborba hozd magaddal a következőket:`</p>`

```
<ul>
  <li>hálózsák</li>
  <li>zseblámpa</li>
  <li>fürdőruha</li>
</ul>
```

A táborba hozd magaddal a következőket:

- hálózsák
- zseblámpa
- fürdőruha

LISTA BEILLESZTÉSE

- Alakítsunk ki egy felsoroláslistát az előnyös tulajdonságokból!

Előnyös tulajdonságai

-  Okos, könnyen képezhető
-  Szelíd, családbarát
-  Nagyon jó vadászkutya

 **Fontos!** Ha gazdi szeretnél lenni, nézz utána, hogy a *felelős* kutyatartással kapcsolatban milyen ajánlások és kötelezettségek vannak!

SORSZÁMOZOTT LISTA

`<ol><li>`: Sorozámozott lista

Sorozámozott lista esetén az `<ol>` és `</ol>` tagek közé helyezzük el a listaelemeket.

Példa

```
<p>A verseny állása:</p>
  <ol>
    <li>Nagy Gábor</li>
    <li>Németh Tamás</li>
    <li>Tóth Éva</li>
  </ol>
```

A verseny állása

1. Nagy Gábor
2. Németh Tamás
3. Tóth Éva

SORSZÁMOZOTT LISTA

Számozás adott sorszámtól

A **start** paraméterrel beállítható a sorszámozott lista kezdő sorszáma.

Példa

```
<p>A verseny további helyezettjei:</p>  
  <ol start="4">  
    <li>Kiss Ágota</li>  
    <li>Lenkei Zoltán</li>  
    <li>Molnár Tamás</li>  
    <li>Hegyi Rita</li>  
  </ol>
```

A verseny további helyezettjei:

4. Kiss Ágota
5. Lenkei Zoltán
6. Molnár Tamás
7. Hegyi Rita

SORSZÁMOZOTT LISTA

Csökkenő sorszámozás

A **reversed** logikai paraméter használatával a sorszámozás csökkenő sorrendben történik.

Példa

```
<p>Legnépszerűbb kutyaajták:</p>  
  <ol reversed>  
    <li>Beagle</li>  
    <li>Bulldog</li>  
    <li>Golden Retriever</li>  
    <li>Németjuhász</li>  
    <li>Labrador</li>  
  </ol>
```

Legnépszerűbb kutyaajták:

5. Beagle
4. Bulldog
3. Golden Retriever
2. Németjuhász
1. Labrador

SORSZÁMOZOTT LISTA

A sorszám fajtajának beállítása

A **type** paraméterrel a sorszámozás típusát adhatjuk meg.

Érték	Magyarázat	Kód példa	Eredmény
1	Számok (tizes számrendszer)	<code><ol type="1"></code>	I. 2. 3. 4. 5.
a	Latin ábécé kisbetűi	<code><ol type="a"></code>	a. b. c. d. e.
A	Latin ábécé nagybetűi	<code><ol type="A"></code>	A. B. C. D. E.
i	Római számok (kisbetűs)	<code><ol type="i"></code>	i. ii. iii. iv. v.
I	Római számok (nagybetűs)	<code><ol type="I"></code>	I. II. III. IV. V.

SORSZÁMOZOTT LISTA

Sorszám típusok - Példák

```
<ol type="1">  
  <li>listaelem</li>  
  <li>listaelem</li>  
  <li>listaelem</li>  
</ol>
```

1. listaelem
2. listaelem
3. listaelem

```
<ol type="i">  
  <li>listaelem</li>  
  <li>listaelem</li>  
  <li>listaelem</li>  
</ol>
```

i. listaelem
ii. listaelem
iii. listaelem

```
<ol type="a">  
  <li>listaelem</li>  
  <li>listaelem</li>  
  <li>listaelem</li>  
</ol>
```

a. listaelem
b. listaelem
c. listaelem

```
<ol type="I">  
  <li>listaelem</li>  
  <li>listaelem</li>  
  <li>listaelem</li>  
</ol>
```

I. listaelem
II. listaelem
III. listaelem

```
<ol type="A">  
  <li>listaelem</li>  
  <li>listaelem</li>  
  <li>listaelem</li>  
</ol>
```

A. listaelem
B. listaelem
C. listaelem



EGYMÁSBA ÁGYAZOTT LISTÁK

LISTÁK EGYMÁSBA ÁGYAZÁSA

A listák egymásba ágyazásánál ügyeljünk arra, hogy a beágyazott listát közvetlenül a listaelem szövege után kell beillesztenünk, még a lezáró `</li>` elem előtt!

Példa

```
<p>Kedvelt reggeli italok:</p>
<ol>
  <li>Gyümölcslé <ul>
    <li>Narancslé</li>
    <li>Almalé</li>
  </ul>
</li>
  <li>Tea <ul>
    <li>Fekete tea</li>
    <li>Gyümölcstea</li>
  </ul>
</li>
</ol>
```

Kedvelt reggeli italok

1. Gyümölcslé
 - Narancslé
 - Almalé
2. Tea
 - Fekete tea
 - Gyümölcs tea

KÉPEK, ILLUSZTRÁCIÓK BEILLESZTÉSE

KÉPEK A WEBOLDALAKON

- A HTML szabvány nem határozza meg, hogy milyen képformátumok illeszthetők be a weboldalakra. Az egyes böngészőprogramok különböző képformátumokat támogathatnak.
- Az alábbi képformátumok a legszélesebb körben támogatottak:
 - PNG (Portable Network Graphics)
 - JPEG (Joint Photographic Expert Group image)
 - SVG (Scalable Vector Graphics)
 - GIF (Graphics Interchange Format)
 - APNG (Animated Portable Network Graphics)

KÉPEK BEILLESZTÉSE

Az `<img>` tag használata

Legfontosabb paraméterei

- `src`: kép forrása (kötelező)
- `alt`: helyettesítő szöveg (kötelező)
- `title`: az elemhez tartozó súgó (opcionális)
- `width`: kép szélessége képpontban (opcionális)
- `height`: kép magassága képpontban (opcionális)

KÉPEK BEILLESZTÉSE

Az `src` paraméter használata

Itt adjuk meg a **kép elérhetőségét**. Ezt megtehetjük abszolút, illetve relatív módon, a HTML állomány helyéhez viszonyítva.

1. A kép ugyanabban a mappában van, mint a HTML állomány.

```

```

2. A kép a kepek mappában található.

```

```

3. A kép a szülőmappában található.

```

```

4. A kép elérhetősége abszolút módon van megadva.

```

```



KÉPEK BEILLESZTÉSE

Az **alt** paraméter használata

Az alt paraméter a **helyettesítő szöveget** jelenti. Fontos akadálymentességi szerepe van! A látássérült emberek által használt program ezt a szöveget olvassa fel. A böngészőprogramban akkor jelenik meg, ha a kép nem érhető el valamilyen okból.

Példa

```

```

 Színes esernyők egy belgrádi utcában

Ha a képet nem tudja betölteni a böngészőprogram, az alt-ban megadott szöveg megjelenik,

KÉPEK BEILLESZTÉSE

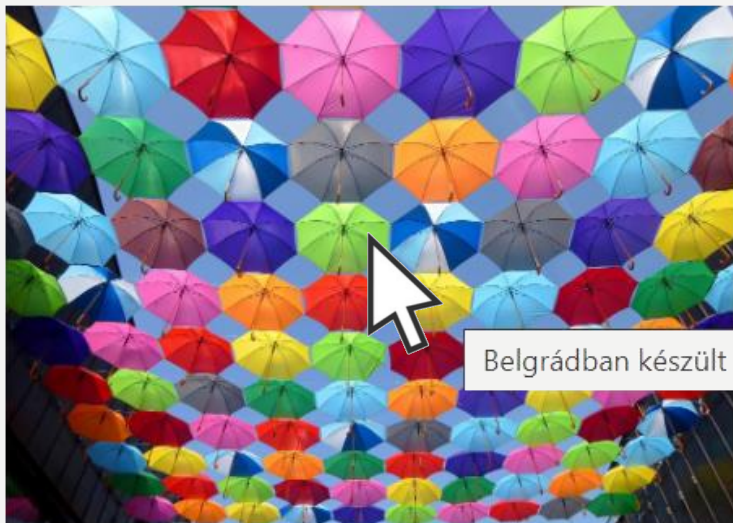
A **title** paraméter használata

A paraméterben megadott **magyarázó szöveg** akkor jelenik meg, ha a kép fölé visszük az egeret. Buborékszövegnek is hívják.

Példa

```

```



A title paraméter értéke
szövegbuborékban jelenik meg.

KÉPEK BEILLESZTÉSE

A `width` és `height` paraméterek használata

Ezekkel a kép szélességét (`width`) és magasságát (`height`) lehet beállítani képpontokban.

1. Ha csak a szélességet állítjuk be, a magasság is arányosan csökken.

```

```



2. Ha mindkét paramétert beállítjuk, akkor akár a képarány is megváltozik.

```

```



KÉPEK BEILLESZTÉSE

- Illesszünk be most két képet egymás mellé, azonos magasságban.
- A magasság legyen 200 képpont!
- Adjuk meg az `alt` és `title` paramétereket is!



```
  

```



ILLUSZTRÁCIÓK BEILLESZTÉSE

ILLUSZTRÁCIÓK BEILLESZTÉSE

A `<figure>` és `<figcaption>` tagek használata

A weblapon aláírással ellátott illusztrációt is elhelyezhetünk.

Az illusztráció lehet kép, szöveg, forráskód, vagy akár videó is. Ezeket a `<figure>` és `</figure>` tagek között helyezhetjük el. (Akár többet is.)

A `<figcaption>` tagben az illusztráció aláírását adjuk meg.

Példa

```
<figure>
  
  <figcaption>Színes esernyők egy belgrádi utcában
</figcaption>
</figure>
```



Színes esernyők egy belgrádi utcában

ILLUSZTRÁCIÓ BEILLESZTÉSE

Illesszük be a megfelelő címkéket, hogy a két kép legyen az illusztráció, és legyen közös képaláírásuk!



Fotók Goldiról

HIVATKOZÁSOK BEILLESZTÉSE

MI A HIVATKOZÁS?



- A hivatkozásokkal (linkekkel) kapcsolatot hozhatunk létre az egyes erőforrások (különböző oldalak, médiaelemek stb.) között.
- A böngészőprogramok a **szöveges linkeket** alapesetben aláhúzással illetve eltérő színnel jelölik.
- A szöveges hivatkozások alap színei:
 - nem látogatott linkek: kék
 - látogatott linkek: lila
 - aktív linkek: piros

Visszajelzés küldése

Fotó a kutyámról

HTML validátor

Aktív link
(rákattintottunk és
éppen töltődik)

KÉP, MINT HIVATKOZÁS



- Nem csak szöveg lehet link, hanem például kép is.



A megváltozott kurzor jelzi, hogy a képre egy hivatkozás van elhelyezve.

HIVATKOZÁS BEILLESZTÉSE

Az `<a>` tag használata

- Az `<a>`, az angol *anchor* (horgony) szó rövidítése.
 - A horgony a láncszemekhez (angolul link) kapcsolódik.
 - A webes világban a link a hivatkozást jelenti.



SZÖVEGES HIVATKOZÁS BEILLESZTÉSE

- Az `<a>` és `</a>` tagek között megadott szöveg a hivatkozás szövege. Erre kattintva aktiválhatjuk a hivatkozást.
- A `href` paraméterben adjuk meg a hivatkozott erőforrás webcímét (URL).

Példa

```
<a href="https://validator.w3.org/">HTML validátor</a>
```

A HTML validátor szövegre kattintva betöltődik a <https://validator.w3.org/> weboldal.

MAGYARÁZAT A HIVATKOZÁSHOZ



Érdemes a **title** paramétert használni linkek esetén is, hogy rövid magyarázatot fűzzünk a hivatkozáshoz.



Példa: title paraméter használata

```
<a href="galeria.html" title="Galéria a nyári élményekről">  
</a>
```

Most már azt is látjuk, hogy milyen oldalra jutnánk a link követésével.

Galéria a nyári élményekről

LINK BEILLESZTÉSE

1. Készítsünk egy olyan linket, amelyet követve a honlapunk látogatói még több kutyás képet nézhetnek meg!
2. A weboldal, amelyre hivatkozunk, a <http://tiny.cc/goldipix> rövidített címen elérhető Pixabay portál.



BELSŐ (OLDALON BELÜLI) HIVATKOZÁS

Az oldalunk egy részére is ugorhatunk hivatkozással, ha

- az oldalrészhez egyedi azonosítót rendelünk az **id** paraméterrel
- a **href** paraméterben a **#** jelet követve megadjuk ugyanezen egyedi azonosítót

Példa: A **#friss** hivatkozás segítségével a *friss* egyedi azonosítóval ellátott címsorra ugrunk.

```
<a href="#friss">Érdekel a legújabb munkám?</a>  
<h2 id="friss">Legújabb munkám</h2>
```

BELSŐ (OLDALON BELÜLI) HIVATKOZÁS

A weben publikált honlap oldalrészére csak akkor hivatkozhatunk, ha az oldalrész el van látva egyedi azonosítóval. Ezt a HTML forráskódból tudhatjuk meg.

Példa: A https://en.wikipedia.org/wiki/World_Wide_Web weboldal forráskódjának részlete

```
100 | <h2><span class="mw-headline" id="History">History</span>
```

Példa: Ugrás a Wikipédia oldalának oldalrészére

```
<a href="https://en.wikipedia.org/wiki/World_Wide_Web#History">  
Világháló története (angolul)</a>
```

A # jel előtt a hivatkozott oldal webcíme van elhelyezve.

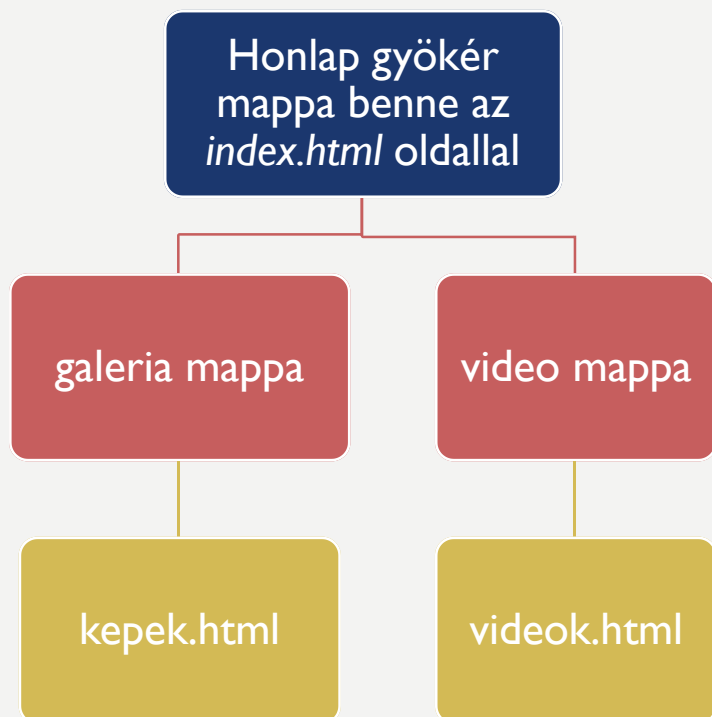
HIVATKOZÁS OLDALON BELÜL

- Helyezzünk el a weblapunk alján egy olyan linket, amellyel vissza tudunk ugrani az oldal elejére!
- Ehhez az egyes címsornak adjunk egy egyedi azonosítót, és a hivatkozásban ugyanezt a nevet adjuk meg!



HIVATKOZÁS A WEBHELY OLDALAI KÖZÖTT

A honlapunkon belül akár relatív módon is hivatkozhatunk az egyes oldalakra.



Hivatkozás az index.html állományból

```
<a href="galeria/kepek.html">Képgaléria</a>  
<a href="video/videok.html">Videók</a>
```

Hivatkozás a kepek.html állományból

```
<a href="../index.html">Kezdőlap</a>  
<a href="../video/videok.html">Videók</a>
```

Hivatkozás a videok.html állományból

```
<a href="../index.html">Kezdőlap</a>  
<a href="../galeria/kepek.html">Képgaléria</a>
```

HIVATKOZÁS SAJÁT OLDALRA

- Fejlesszük tovább úgy a weboldalt, hogy ne csak egy oldalból álljon.
- Mentsük el a korábban látott alapstruktúrát `tipusok.html` néven ugyanabba a mappába, mint ahol az `index.html` állomány van.
- Majd módosítsuk az `index.html` állományt, hogy a másik lapra is el lehessen jutni.
- Soroljuk be ezt a bekezdést a „menu” osztályba, hogy majd a stíluslap segítségével megváltoztathassuk a kinézetét!

LINK MEGNYITÁSA ÚJ FÜLÖN

A **target** paraméterrel megadhatjuk, hogy hova töltsön be az oldal.

Gyakori értékek:

_blank: új fülön jelenik meg

_self: aktuális ablakban jelenik meg (ez az alapértelmezett)

Példa

A hivatkozás új fülön nyílik meg a böngészőben

```
<a href="https://pixabay.com/hu/" target="_blank">  
Pixabay (ingyenes képek)</a>
```

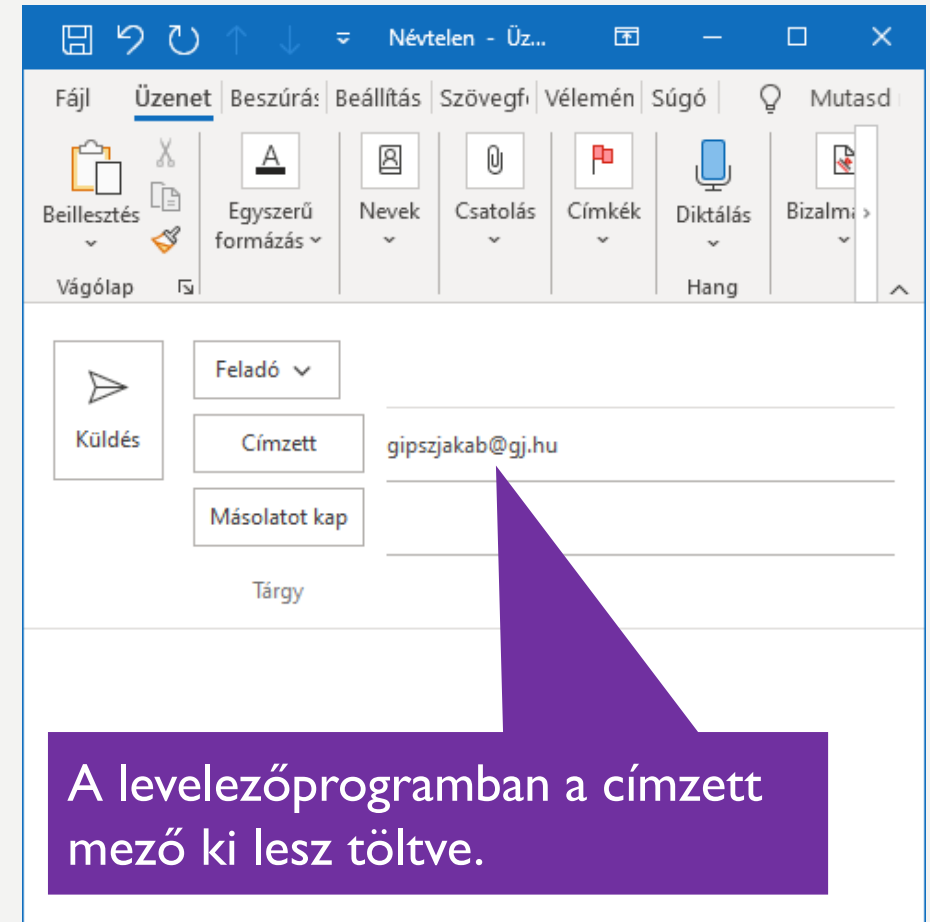
HIVATKOZÁS E-MAIL CÍMRE

Amikor e-mail címre hivatkozunk, a **href** paraméterben a `"mailto:emailcím"` formátumot használjuk.

Ennek hatására az alapértelmezett levelezőprogram nyílik meg.

Példa

```
<a href="mailto:gipszjakab@gj.hu">E-mail küldése</a>
```



HIVATKOZÁS TELEFONSZÁMRA

Amikor telefonszámra hivatkozunk, a **href** paraméterben a **"tel:telefonszam"** formátumot használjuk.

Ennek hatására az alapértelmezett hívó alkalmazás nyílik meg.

Példa

```
<a href="tel:+362099999999">Hívás indítása</a>
```



A telefonon a kiválasztott (vagy alapértelmezett) alkalmazással történik a tárcsázás.

VIDEÓ, HANG BEILLESZTÉSE

VIDEÓ BEILLESZTÉSE

A HTML5-szabvány jelentős újdonsága, hogy a multimédiás állományok beillesztését (pl. hangok, videók) jelentősen leegyszerűsíti.

Vezérlő eszköztár megjelenítése

Szélesség

Magasság

```
<video controls width="" height="">  
  <source src="" type="video/mp4">  
</video>
```

Videó (mp4 formátumú) webcíme

VIDEÓ BEILLESZTÉSE

1. Próbáljuk ki a videóbeillesztést!
2. A médiaelemek között találunk egy videót Goldiról.
3. Hozzunk létre egy bekezdést, amelyben bevezetjük szövegesen a videó tartalmát, majd ágyazzuk be az oldalra a videót!
4. A videó szélessége 400 képpont legyen!

A videón is láthatod, hogy Goldi nagyon szeret a földön hemperegni, amely miatt általában koszosan jövünk haza a sétából.



HANG BEILLESZTÉSE

Hang beillesztése hasonlóan történik az `audio` címkével. Érdeemes mp3 formátumot használni!

Vezérlő eszköztár megjelenítése

```
<audio controls>  
  <source src="" type="audio/mpeg">  
</audio>
```

audió (mp3 formátumú) webcíme

EGYSZERŰ TÁBLÁZATOK BEILLESZTÉSE

A TÁBLÁZATOKRÓL

- **Cél:** adatok prezentálása átlátható módon, sorokba és oszlopokba rendezve.
- Az **adatcellákba** kerülnek az adatok.
- Azt, hogy az oszlopban vagy sorban milyen jellegű adatok vannak, a **fejléc cellákban** adhatjuk meg.
- A táblázatnak lehet **címe** is, amely utal a táblázat tartalmára.

A szerencsés nyertesek

Név	Lakhely
Kiss Ágnes	Budapest
Nagy Tibor	Debrecen
Kővágó László	Kecskemét

Táblázat címe

Fejléc cellák

Adatcellák

A TÁBLÁZATOKRÓL

- A fejléc cellák vonatkozhatnak oszlopokra és sorokra is.

A szerencsés nyertesek

Név	Lakhely
Kiss Ágnes	Budapest
Nagy Tibor	Debrecen
Kővágó László	Kecskemét

Fejléc cellák
(oszlopokra vonatkozóan)

A szerencsés nyertesek

Név	Kiss Ágnes	Nagy Tibor	Kővágó László
Lakhely	Budapest	Debrecen	Kecskemét

Fejléc cellák
(sorokra vonatkozóan)

TÁBLÁZATOK BEILLESZTÉSE

A szerencsés nyertesek

Név	Lakhely
Kiss Ágnes	Budapest
Nagy Tibor	Debrecen
Kővágó László	Kecskemét

- A `<table>` és `</table>` tag-ek között kell elhelyezni a táblázat tartalmát.
- A táblázat címe a `<caption>` és `</caption>` címkék közé kerül, közvetlenül a `<table>` nyitó tag után.
- A `<tr>` és `</tr>` címkék közé kerül a táblázat sora.
- A fejlécek kerülnek a `<th>` és `</th>` címkék közé (fejléc cellák).
- Az adatokat a `<td>` és `</td>` címkék között adjuk meg (adatcellák).

```
<table>
  <caption>A szerencsés nyertesek</caption>
  <tr>
    <th>Név</th>
    <th>Lakhely</th>
  </tr>
  <tr>
    <td>Kiss Ágnes</td>
    <td>Budapest</td>
  </tr>
  <tr>
    <td>Nagy Tibor </td>
    <td>Debrecen</td>
  </tr>
  <tr>
    <td>Kővágó László</td>
    <td>Kecskemét</td>
  </tr>
</table>
```

TÁBLÁZAT BEILLESZTÉSE

1. Készítsük el önállóan az itt látható táblázatot, és helyezzük el a `tipusok.html` oldalon!
2. Ne csak a táblázatot valósítsuk meg, hanem a címsort, a főmenüt és a forrásra mutató hivatkozást is!

A Golden Retriever bemutatása

Kezdőlap **Típusok**

Típusok

A Golden Retriever típusai	
Típus	Típus leírása
Munkatípus	Könnyedebb felépítésű. Vékonyabb, hosszabb fej, sötétebb szőrszín, élénk temperamentum, karcsúbb, ruganyosabb, gyorsabb.
Kiállítási típus	Dúsabb, hosszabb szőrzet, robusztusabb felépítés, világosabb szőrszín, határozottabb.
Európai típus	Szőrszíne az arany világosabb árnyalata. Rövidebb szőrű, mint az amerikai. Lába rövidebb az amerikainál, így megjelenése zömökebb. Jól pigmentált. Jó szögellésű elől, hátul.
Amerikai típus	Orr és fej része finomabb. Színe középarany, rézvörös, de sosem krém. Mellső szögellései szerényebbek, hátul olykor túlszögelt.

Forrás: hu.wikipedia.org/wiki/Golden_retriever

[Vissza az oldal tetejére](#)



Visual Studio Code

TÁBLÁZATOK BEILLESZTÉSE A VISUAL STUDIO CODE ALKALMAZÁSBAN

TÁBLÁZATOK BEILLESZTÉSE

Emmet parancsok használata



- Az emmet parancsok segítségével a táblázatok beillesztése egyszerűsödik.
- Az egymásba ágyazandó címkék közé **>** jelet kell tennünk.
- A címkék számát megadhatjuk a ***** jel után.

Rövidítés	Eredmény
table>tr>td	<pre><table> <tr> <td></td> </tr> </table></pre>
table>tr*2>td	<pre><table> <tr> <td></td> </tr> <tr> <td></td> </tr> </table></pre>

Rövidítés	Eredmény
table>tr*2>td*3	<pre><table> <tr> <td></td> <td></td> <td></td> </tr> <tr> <td></td> <td></td> <td></td> </tr> </table></pre>

BEVEZETÉS A STÍLUSLAPOK HASZNÁLATÁBA

CSS

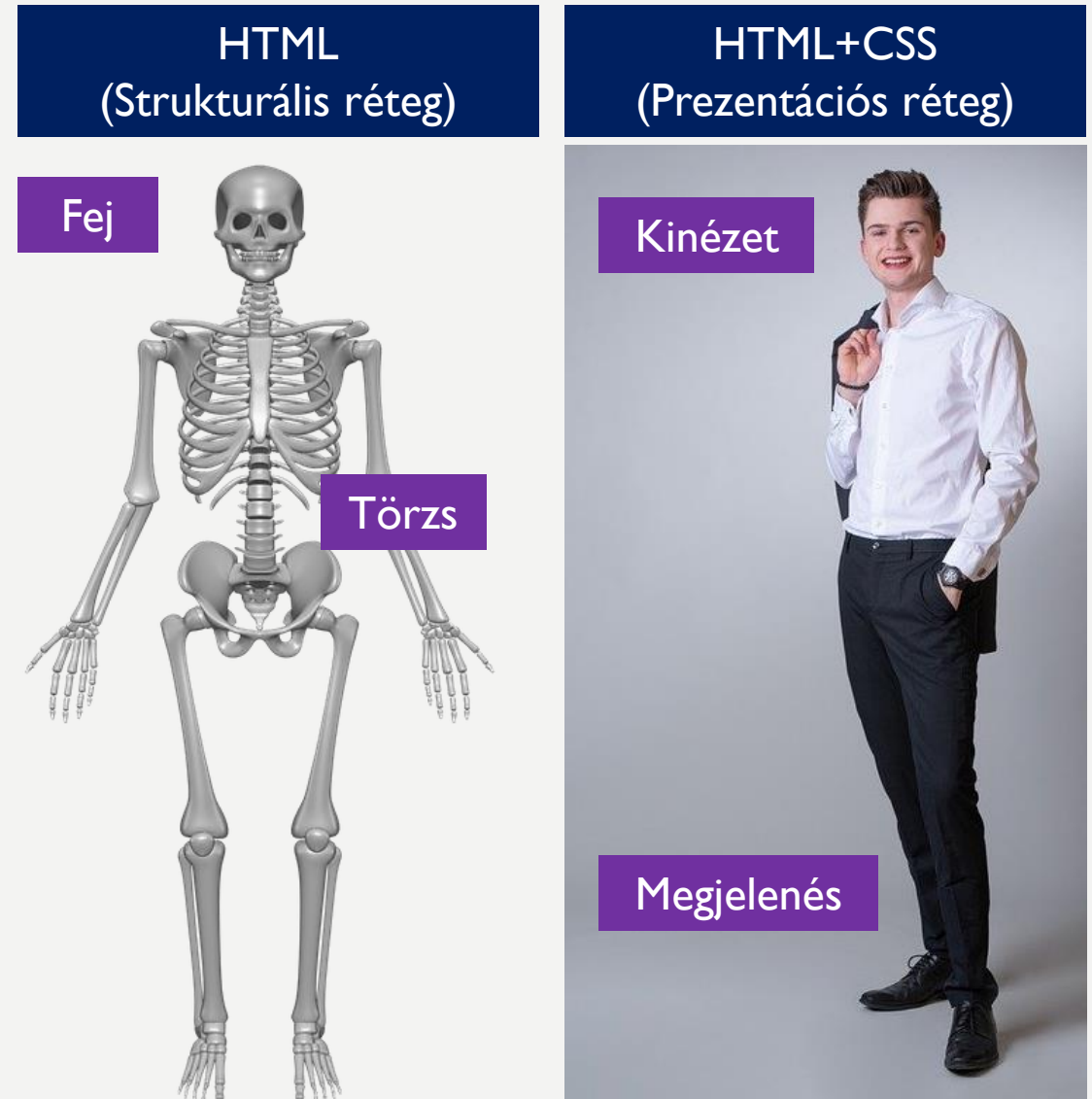




A STÍLUSLAPOK SZEREPE

STÍLUSLAPOK (CSS) HASZNÁLATA

- A **tartalom** leírására a **HTML** leírónyelv szolgál.
- A **kinézetet** a **stíluslapok** segítségével állíthatjuk be.
- **Cascading Style Sheets**
(Lépcsőzetes, vagy rangsorolt stíluslapok)
 - Szabvány a weboldalak megjelenésének leírására.
 - Nyelv saját szintaxissal, szabályokkal.
 - Egyszerű szöveges fájl .css kiterjesztéssel.
 - Jelenleg a CSS3 a legfrissebb változat.



AZONOS VÁZ, KÜLÖNBÖZŐ KINÉZET



CSS Zen Garden

The Beauty of CSS Design

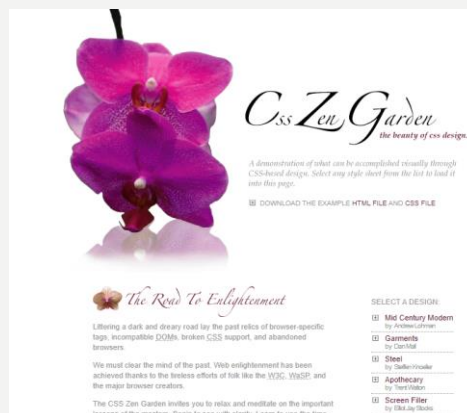
A demonstration of what can be accomplished through CSS-based design. Select any style sheet from the list to load it into this page.

Download the example [html file](#) and [css file](#)

The Road to Enlightenment

Littering a dark and dreary road lay the past relics of browser-specific tags, incompatible DOMs, broken CSS support, and abandoned browsers.

We must clear the mind of the past. Web enlightenment has been achieved thanks to the tireless efforts of folk like the W3C, WaSP, and the major browser creators.



CSS Zen Garden

The Beauty of CSS Design

A demonstration of what can be accomplished through CSS-based design. Select any style sheet from the list to load it into this page.

Download the example [html file](#) and [css file](#)



<http://www.csszengarden.com/>

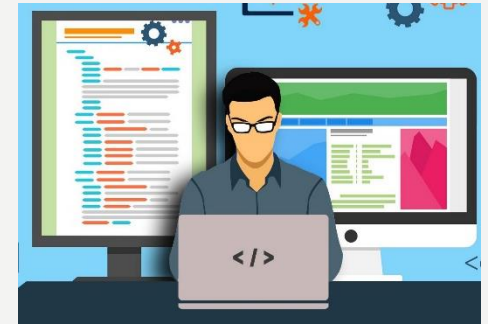
STÍLUSLAPHASZNÁLAT ELŐNYEI

- Haladó formázási, elrendezési lehetőségek.
- Egységes, konzisztens megjelenés.
- A website könnyebben karbantartható.
- Haladó akadálymentességi technikák használata.
- Különböző médiatípusokhoz más megjelenés társítható.
 - Képernyő (különböző felbontásokra), nyomtató, TV, projektor, stb.



STÍLUSLAP TÍPUSOK

- **A szerző által készített stíluslapok**
 - Amikor a honlap fejlesztője/szerzője határozza meg, hogy hogyan nézzen ki az oldal (erről szól a tananyagunk).
- **Felhasználói stíluslapok**
 - A böngészőprogramokban lehet beállítani az egyéni stíluslapokat.
 - Jellemzően fogyatékkal élő, vagy technológiailag megkülönböztetett emberek által használt lehetőség pl. a betűméret, kontraszt, színvilág felülbírálására.
- **Böngészőkben definiált stíluslapok**
 - Az elemek alapértelmezett megjelenítéséért felelősek.





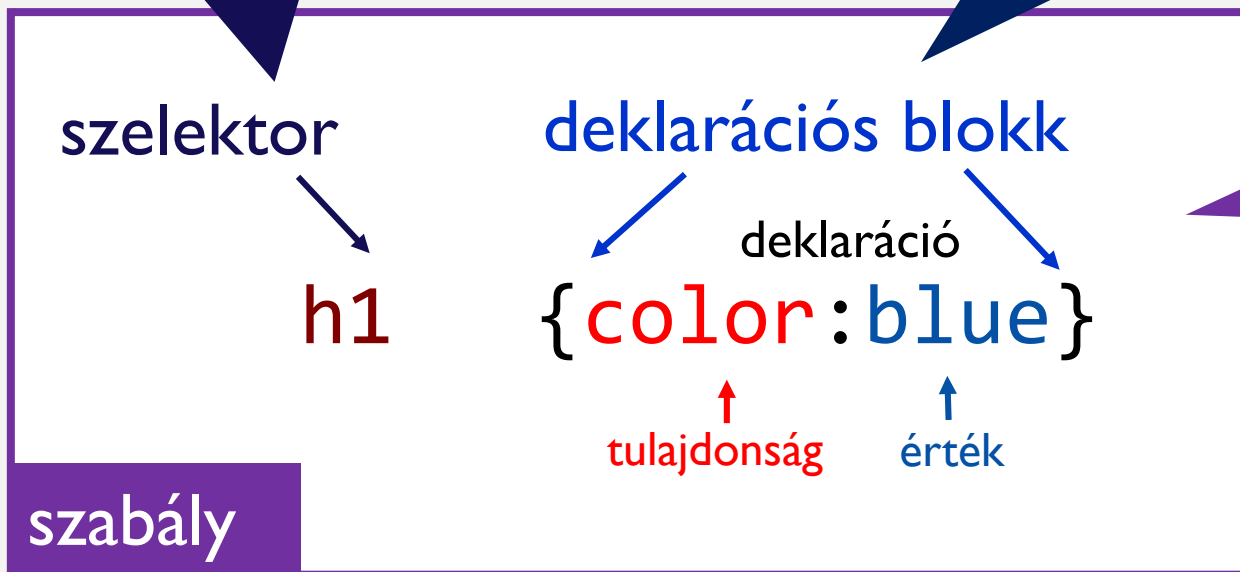
CSS SZINTAXIS

CSS SZABÁLY

A CSS szabály két részből áll: szelektor és deklarációs blokk.

A kijelölő (szelektor) tartalmazza a formázandó elem(ek)et.

A **deklarációs blokk** tartalmazza az elemek formázásának leírását.



Ez a szabály azt írja le, hogy az 1. szintű címsorok (h1) kék színnel jelenjenek meg.

TULAJDONSÁG: ÉRTÉK

- A tulajdonság – érték párokat kettősponttal választjuk el.
- Minden deklaráció végét pontosvesszővel zárjuk le.
 - Az utolsó deklaráció végére nem kötelező kitenni a pontosvesszőt.
- A könnyebb olvashatóság/átláthatóság miatt érdemes szóközökkel, tabulátorokkal igazítani a kódot.

```
h1 {  
    color:blue;  
    font-size:300%  
}
```

Ez a szabály azt írja le, hogy az 1-es címsorok (h1) kék színnel jelenjenek meg, és a betűméret háromszoros legyen.

MEGJEGYZÉSEK

- Megjegyzéseket az `/*` és `*/` karakterek között helyezhetünk el.
- A kommentek hasznosak, ha később nekünk (vagy másnak) módosítania kell a kódot.
- A megjegyzések több sorosak is lehetnek.

```
/* Címsorok beállítása */  
h1 {  
    color:blue;  
    font-size:300%  
}
```

STÍLUSLAP CSATOLÁSI LEHETŐSÉGEK

Külső stíluslap
belinkelése HTML
taggel

Beágyazott stíluslap

Külső stíluslap
importálása

Elemközi (inline)
stílus az adott
elemre vonatkozóan

```
<!DOCTYPE html>
<html lang="hu">
  <head>
    <meta charset="utf-8">
    <title>CSS példa</title>
    <link rel="stylesheet" type="text/css" href="css/alap.css"
      media="screen">
    <style>
      @import url("css/kontrasztos.css");
      h1 {color: blue}
    </style>
  </head>

  <body>
    <h1>Címsor 1</h1>
    <h2>Címsor 2</h2>
    <p style="color:green">Ez egy zöld bekezdés</p>
  </body>
</html>
```

ELEMKÖZI STÍLUSMEGADÁS

- A **style** paraméterrel közvetlenül az adott elemet tudjuk formázni.
- A szabály csak az adott elemre (illetve az abba beágyazott elemekre) lesz érvényes.
- Csak indokolt esetben használjuk! Nem hatékony, körülményes a változtatások átvezetése.

```
<p style="color:green">  
Ez egy zöld színű bekezdés  
</p>
```

Amikor hasznos

- Ha egy tartalomkezelő rendszerben nincs jogunk a központi stíluslap módosítására, de az elemek megjelenését módosítani szeretnénk
- Ha HTML formátumú e-mailt szeretnénk küldeni.
- A megjelenés dinamikus módosításakor (pl. JavaScript módosítja az oldal megjelenését)

BEÁGYAZOTT STÍLUSMEGADÁS

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

  <meta charset="utf-8">
    <title>Stíluslap példa</title>
  <style>
    h1 {
      color: blue
    }
  </style>
</head>

<body>
  <h1>Címsor 1</h1>
  <p>Lorem ipsum dolor sit amet, consectetur adipiscing elit.</p>
  <h1>Címsor 1</h1>
  <p>Lorem ipsum dolor sit amet, consectetur adipiscing elit.</p>
  <h1>Címsor 1</h1>
  <p>Lorem ipsum dolor sit amet, consectetur adipiscing elit.</p>
</body>
</html>
```

A `<style>` és `</style>` címkék között elhelyezett szabállyal minden 1-es címsor színét kékre állítottuk.

A `<style>` és `</style>` közti szabályokat akár külső állományba vihetjük, amit belinkelhetünk az oldalra.

KÜLSŐ STÍLUSLAP ALKALMAZÁSA

Hozzunk létre egy css állományt (pl. pelda.css) a következő tartalommal a HTML állományunk mappájában:

```
h1 {color:blue}
```

```
<!DOCTYPE html>
<html>
<head>
<meta charset="utf-8">
  <title>Külső CSS</title>
  <link rel="stylesheet" type="text/css" href="pelda.css">
</head>
<body>
<h1>Címsor 1</h1>
<p>Lorem ipsum dolor sit amet, consectetur adipiscing elit.</p>
<h1>Címsor 1</h1>
<p>Lorem ipsum dolor sit amet, consectetur adipiscing elit.</p>
<h1>Címsor 1</h1>
<p>Lorem ipsum dolor sit amet, consectetur adipiscing elit.</p>
</body>
</html>
```

Ezzel a megoldással ugyanazt a stíluslapot több oldalhoz is hozzárendelhetjük.

Ha úgy döntünk, hogy a címsor mégsem kék színű, akkor csak egy helyen kell megváltoztatni a tulajdonságot.

KÜLSŐ STÍLUSLAP ALKALMAZÁSA

`<link>` tag használata a HEAD részben

Paraméterek:

- **rel**: a kapcsolat/viszony meghatározása. (Kötelező!)
Stíluslap esetén az érték: `stylesheet`
- **type**: a belinkelt fájl típusa. (Opcionális!)
Stíluslap esetén az érték: `text/css`
- **href**: a külső stíluslap elérési útvonala (URL) (Kötelező!)

Példa

```
<link rel="stylesheet" type="text/css" href="pelda.css">
```


STÍLUSLAP CSATOLÁSA

1. Csatoljuk a `goldi.css` stíluslapot az oldalhoz!
2. Próbáljuk ki a `link` címke és a `@import` szabály megadását is!

CSS KIJELÖLŐK (SZELEKTOROK)

TÍPUSKIJELÖLŐ (TYPE SELECTOR)

- Elemkijelölőnek is szokták nevezni.
- Akkor használjuk, ha minden adott típusú elemre ugyanazt a stílust szeretnénk beállítani.
- Ebben az esetben a kijelölő (szelektor) megegyezik a HTML címke nevével.

```
h1 {  
  color:blue;  
  font-style:italic;  
}
```

Ez a szabály azt írja le, hogy az oldalon szereplő összes 1. szintű címsor kék színnel, és dőlt betűvel (*italic*) jelenjen meg.

OSZTÁLY KIJELELŐ (CLASS SELECTOR)

- Akkor használjuk, ha egy stílust több (azonos vagy különböző) elemre (de nem minden elemre) akarjuk beállítani.
- Létrehozhatunk általános osztályt, amelyet aztán bármilyen elemnél felhasználhatunk.
- Az osztály neve betűvel kell, hogy kezdődjön, tartalmazhat számokat, aláhúzás jelet és kötőjelet.

Általános osztály megadása CSS-ben

```
.osztálynév {deklaráció}
```

Példa

```
.fontos {color:red;}
```

Osztály megadása HTML kódban
class paraméterrel

```
class="osztálynév"
```

```
<p class="fontos">  
<h1 class="fontos">
```

OSZTÁLY KIJELÖLŐ (CLASS SELECTOR)

- Az osztály kijelölőt egy másik típusú kijelölőhöz (szelektorhoz) is társíthatjuk.
- Ugyanolyan osztálynevet több elemnél is megadhatunk, esetenként más-más tulajdonságokkal.

Osztály megadása CSS-ben

`elem.osztálynév {deklaráció}`

CSS példa

```
h1.fontos {color:red;}  
p.fontos  {font-size:200%}
```

HTML példa

```
<h1 class="fontos">  
<p class="fontos">
```

AZONOSÍTÓ KIJELELŐ (ID SELECTOR)

- A HTML kódban az **id** paraméterrel adhatunk meg egyedi azonosítót.
- Nem lehet az oldalon két ugyanolyan azonosítójú elem.
- Az azonosító legalább egy karakter hosszú, és nem tartalmazhat szóközt. A kis- és nagybetű különbség számít.

Egyedi azonosító megadása CSS-ben

```
#azonosító {deklaráció}  
elem#azonosító {}
```

Példa

```
#b1 {color:red;}  
h1#elso
```

Azonosító megadás HTML kódban

```
id="azonosító"
```

```
<p id="b1">  
<h1 id="elso">
```

AZONOSÍTÓ KIJELELŐ

Állítsuk be, hogy az `eleje` azonosítóval ellátott elem színe világosszürke (*lightgray*) legyen!



DEKLARÁCIÓK CSOPORTOSÍTÁSA

DEKLARÁCIÓK CSOPORTOSÍTÁSA

```
h1 {font-family: sans-serif}
h2 {font-family: sans-serif}
h3 {font-family: sans-serif}
```



Csoport kijelölő

```
h1, h2, h3 {font-family: sans-serif}
```

A szelektorokat vesszővel választjuk el.

```
h1 {font-family: helvetica}
h1 {font-size: 12pt}
h1 {font-style: normal}
```



Tulajdonság-érték párok egybevonása

```
h1 {font-family: helvetica;
    font-size: 12pt;
    font-style: normal;}
```

A tulajdonság:érték párok után pontosvessző áll.

```
h1 {font-weight: bold}
h1 {font-size: 12pt}
h1 {font-family: helvetica}
```



Összevont (shorthand) megadás

```
h1 {font: bold 12pt helvetica}
```

Az egyes értékek között szóköz van.
Valamikor nem számít a sorrend, máskor igen.

CSOPORTOSÍTÁS

Állítsuk be, hogy az `eleje` azonosítóval ellátott elem ne csak világosszürke legyen, hanem legyen középre igazítva (`text-align: center`)!

SZÍNEK ÉS MÉRTÉKEGYSÉGEK



MÉRTÉKEGYSÉGEK

MÉRTÉKEGYSÉGEK

RELATÍV MÉRTÉKEGYSÉGEK

Egy másik tulajdonság értékétől függenek, például a szülő elem méretétől, vagy az ott alkalmazott betűmérettől.

300%

ABSZOLÚT MÉRTÉKEGYSÉGEK

Az érték nem függ más elemek értékétől, konkrét méretet jelöl (pl. cm)

16px

ALAP MÉRTÉKEGYSÉGEK (CSS3)

Abszolút	Relatív
• in • cm • pt (=1/72in) • px (képpont)	• em (1 em az aktuális betűméretnek felel meg. Az 1.5em tehát másfél betűméretet jelent.) • % (relatív értéket jelent, de hogy minek az értékét, az változó. Lehet: azonos elem más tulajdonsága, szülő elem egy tulajdonsága, a formázott tartalom egy tulajdonsága (pl. tartalmazó elem szélessége). pl. 150%) • rem (a gyökér elem betűméretéhez képest történő relatív méretmegadás pl. 1.5rem)

A számokat egybeírjuk a mértékegységekkel!
Vagyis **12px** és nem **12 px** írandó értékként.

BETŰMÉRET

Állítsuk be a `body` elemre, hogy a betűméret 150%-os legyen!

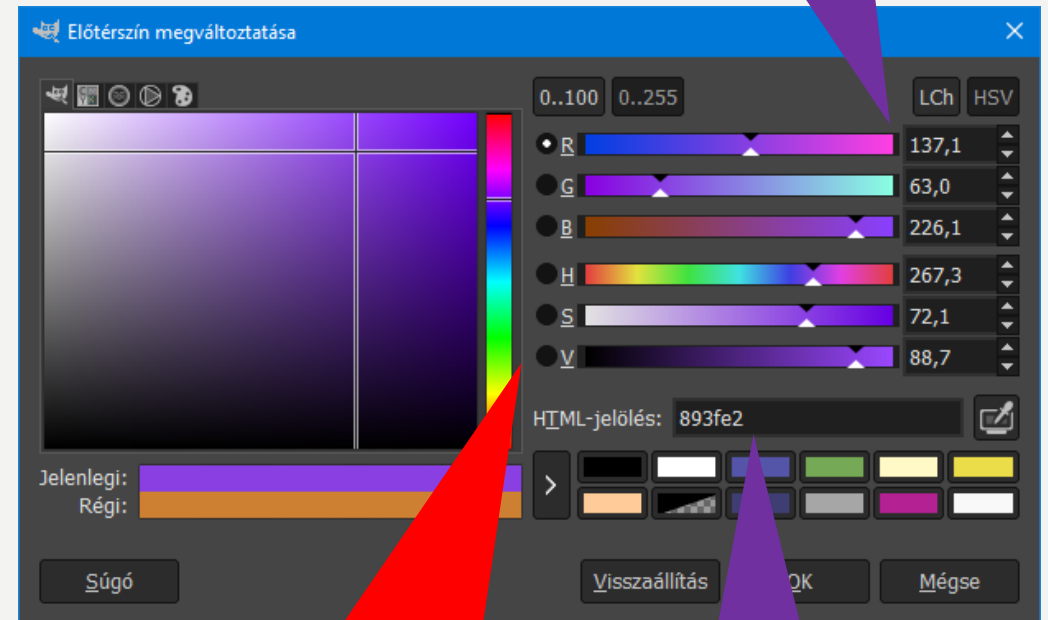
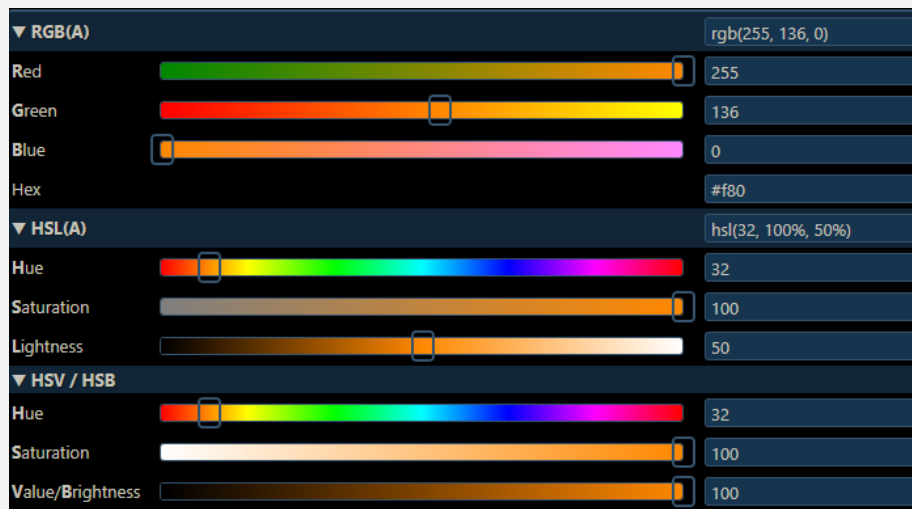


SZÍNEK

SZÍNKÓDOK MEGHATÁROZÁSA

Szinte bármelyik grafikus rajzoló, képmanipuláló program alkalmas a színek meghatározására (pl. GIMP).

A <http://colorizer.org/> honlapon egyszerűen átkonvertálhatjuk a színek kódokat más színrendszerekbe.



RGB kód

Vigyázzunk ez nem HSL kód, hanem HSV.

Szín HTML jelölése

SZÍNJELÖLÉSEK, SZÍN NEVEK

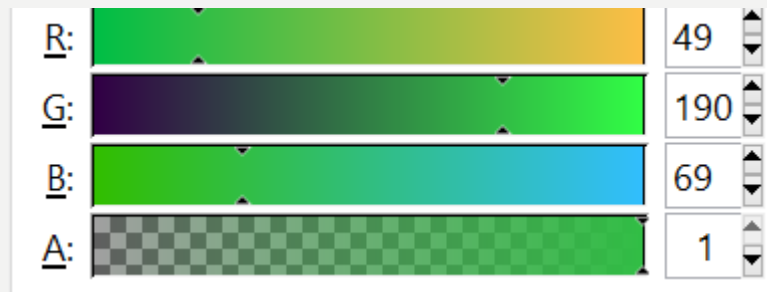
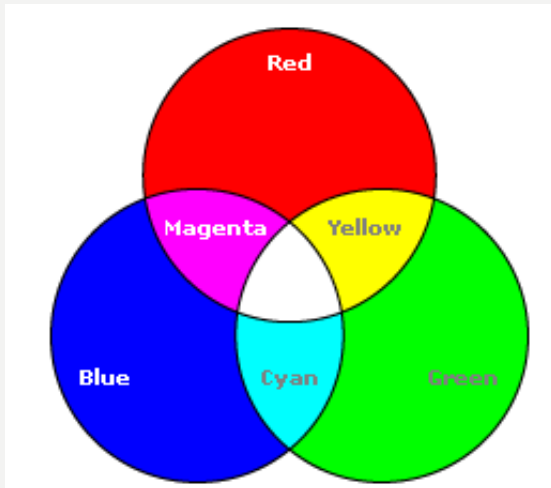
- A színeket beállíthatjuk angol szín nevekkal.
- A színnevek teljes listája a <https://www.w3.org/TR/css-color-3/#svg-color> címen elérhető.
- A szöveg színét a **color** tulajdonsággal tudjuk beállítani.
- Főleg az alapszínek esetén használatos a gyakorlatban (pl. white, black, blue)

Named	Numeric	Color name	Hex rgb	Decimal
		<i>black</i>	#000000	0,0,0
		<i>silver</i>	#C0C0C0	192,192,192
		<i>gray</i>	#808080	128,128,128
		<i>white</i>	#FFFFFF	255,255,255
		<i>maroon</i>	#800000	128,0,0
		<i>red</i>	#FF0000	255,0,0
		<i>purple</i>	#800080	128,0,128
		<i>fuchsia</i>	#FF00FF	255,0,255
		<i>green</i>	#008000	0,128,0
		<i>lime</i>	#00FF00	0,255,0
		<i>olive</i>	#808000	128,128,0
		<i>yellow</i>	#FFFF00	255,255,0
		<i>navy</i>	#000080	0,0,128
		<i>blue</i>	#0000FF	0,0,255
		<i>teal</i>	#008080	0,128,128
		<i>aqua</i>	#00FFFF	0,255,255

		<i>lightgray</i>	#D3D3D3	211,211,211
		<i>lightgreen</i>	#90EE90	144,238,144
		<i>lightgrey</i>	#D3D3D3	211,211,211
		<i>lightpink</i>	#FFB6C1	255,182,193
		<i>lightsalmon</i>	#FFA07A	255,160,122
		<i>lightseagreen</i>	#20B2AA	32,178,170
		<i>lightskyblue</i>	#87CEFA	135,206,250
		<i>lightslategray</i>	#778899	119,136,153
		<i>lightslategrey</i>	#778899	119,136,153
		<i>lightsteelblue</i>	#B0C4DE	176,196,222
		<i>lightyellow</i>	#FFFFE0	255,255,224
		<i>lime</i>	#00FF00	0,255,0
		<i>limegreen</i>	#32CD32	50,205,50
		<i>linen</i>	#FAF0E6	250,240,230

RGB SZÍN-KOORDINÁTARENDSZER

- A képernyőkön minden színt az RGB alapszínek additív keverésével állítanak elő. (**R**=vörös, **G**=zöld, **B**=kék)
- Ha mindhárom komponens maximális (255) értéken van fehér színt kapunk, ha a minimális értéken (0), akkor feketét.



Az opcionális „**A**” érték az Alfa csatornára utal, vagyis az átlátszatlanság mértékére.

```
color: rgb(49, 190, 69)
color: rgba(49, 190, 69, 1)
```

RGB KÓD 16-OS SZÁMRENDSZERBEN

- Az R,G,B értékek 0 és 255 között adhatóak meg.
- Ha tömörebb formában akarjuk megadni, akkor a 10-es számrendszerbeli kódot át lehet számolni 16-os számrendszerbe. Ilyenkor a kód elé # jelet kell tenni. Ezt HTML színkódnak is nevezik.



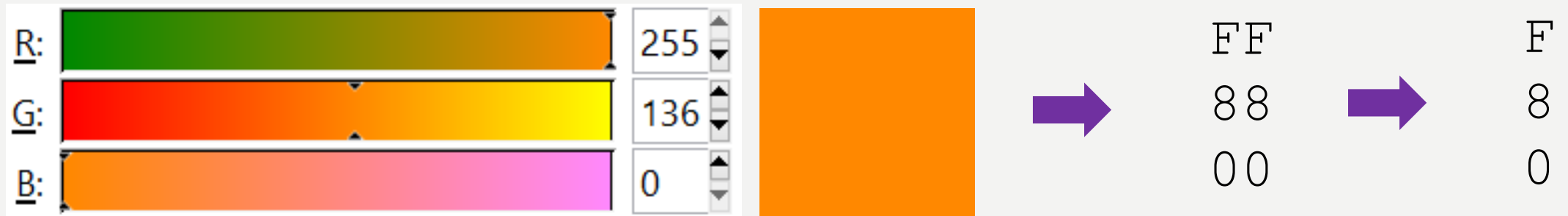
FF
80
00

color: rgb(255,128,0)

color: #ff8000

RGB KÓD 16-OS SZÁMRENDSZERBEN

- Ha az R,G,B kódban ugyanazok a számjegyek ismétlődnek, akkor azokat lehet rövidíteni, vagyis egyszer leírni.



color: rgb(255,136,0)

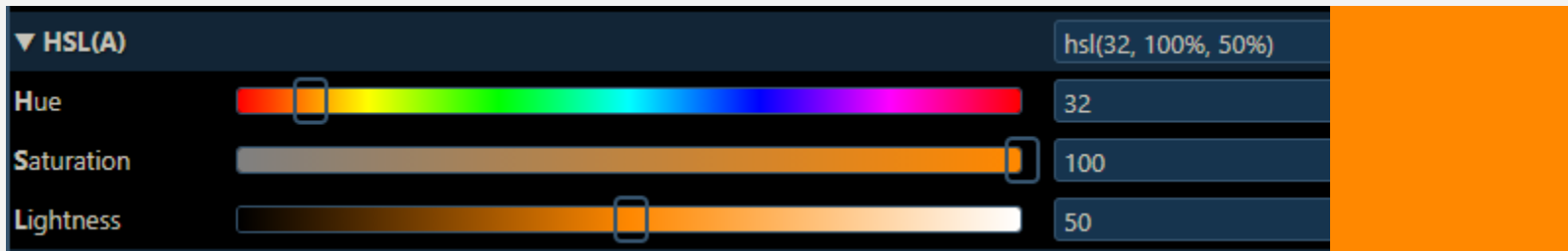
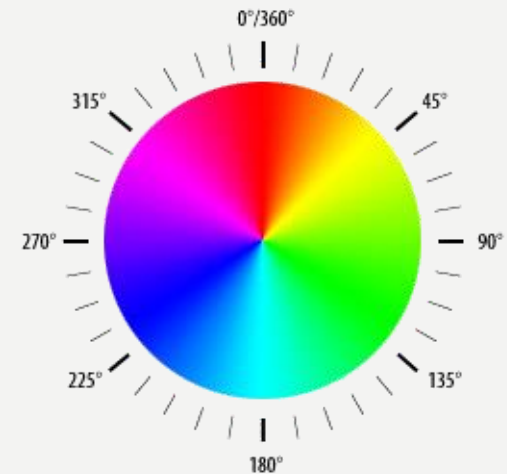
color: #ff8800

color: #f80

HSL SZÍN-KOORDINÁTARENDSZER

Egyszerűbben keverhetőek a színárnyalatok, mint az RGB-ben.

- HSL/HSLA:
 - H: árnyalat (0 és 360 között)
 - S: a telítettség (0 és 100 között)
 - L: fényesség (0 és 100 között)
 - A: átlátszatlanság (0 és 1 között)



A színgör ellentétes oldalain a kiegészítő (komplementer) színek találhatóak.

`color: hsl(32, 100, 50)`

`color: hsla(32, 100, 50, 0.8)`

SZÍNJELÖLÉSEK - ÖSSZEFOGLALÓ

- Szín megadása **10-es számrendszerbeli RGB/RGBA** kóddal

```
<p style="color:rgb(51,71,28)">  
<p style="color:rgb(51,71,28,0.5)">
```
- Szín megadása **16-os számrendszerbeli RGB** kóddal

```
<p style="color:#f66">  
<p style="color:#bda033">
```
- Szín megadása **10-es számrendszerbeli HSL/HSLA** kóddal

```
<p style="color:hsl(300,50,70)">  
<p style="color:hsla(300,50,70,1)">
```
- Szín megadása **százalékos RGB** értékekkel.

```
<p style="color:rgb(50%,70%,30%)">
```

SZÍN MEGADÁS RGB KÓDDAL

1. Állítsuk be, hogy a kettes címsorok színe a következő legyen: `#52463d`
2. Derítsük ki, mi ennek a színnek a decimális RGB kódja?

ÁTLÁTSZATLANSÁG

- Az alfa csatorna értéke 0 és 1 között vehet fel értéket.
- 1 esetén teljesen átlátszatlan az adott szín, 0 esetén teljesen átlátszó.
- A képen egy fehér címsor látható különböző átlátszatlansági értékekkel.



ÁTLÁTSZATLANSÁG

Állítsuk be, hogy az `eleje` azonosítóval ellátott elem színe fehér legyen, de az átlátszatlanság értéke 0.7 legyen!



HÁTTÉRSZÍN BEÁLLÍTÁSA

HÁTTÉRBEÁLLÍTÁSOK

HÁTTÉRSZÍN BEÁLLÍTÁSA

A **background-color** tulajdonság

Az elem háttérszínét állítja be.

Fontos! Olyan háttérszínt állítsunk be, hogy a szöveg **olvasható legyen**.
Ügyeljünk a megfelelő kontrasztarány biztosítására!

Hogy a kontrasztarány megfelelő-e, a Google Chrome böngésző fejlesztői eszköze segítségével egyszerűen kideríthetjük. Ennek módját később bemutatjuk.

A szöveg nem jól olvasható a túl alacsony kontrasztarány miatt.



Lorem ipsum dolor sit amet consectetur adipisicing elit.

Officiis obcaecati voluptas culpa quia aut odio amet.

Ebben az esetben megfelelő a kontrasztarány.



HÁTTÉRSZÍN BEÁLLÍTÁSA

A **background-color** tulajdonság

CSS példa

```
.bg1 {background-color: lightyellow;}  
.bg2 {background-color: black; color: white }  
.bg3 {background-color: rgb(200,100, 70)}
```

```
<p class="bg1">Lorem ipsum dolor sit am  
et consectetur adipisicing elit.</p>
```

```
<p class="bg2">Lorem ipsum dolor sit am  
et consectetur adipisicing elit.</p>
```

```
<p class="bg3">Lorem ipsum dolor sit am  
et consectetur adipisicing elit.</p>
```

Lorem ipsum dolor sit amet consectetur adipisicing elit.

Lorem ipsum dolor sit amet consectetur adipisicing elit.

Lorem ipsum dolor sit amet consectetur adipisicing elit.

SZÍN MEGADÁS RGB KÓDDAL

Állítsuk be, hogy a `figure` elem háttérszíne az a szín legyen, amely a kölyökkutyát ábrázoló kép háttérében van!





HÁTTÉRKÉP BEÁLLÍTÁSA

HÁTTÉRBEÁLLÍTÁSOK

HÁTTÉRKÉP BEÁLLÍTÁSA

A **background-image** tulajdonság

Az elem háttérképét állítja be.

Értékek:

- A kép elérési útvonala. Pl. `url("kepek/hatter.jpg")`
- A **none** érték azt jelenti, hogy nincs háttérkép beállítva.
- A háttérképet alapértelmezetten mozaikszerűen rendezi el a böngészőprogram, de ezt felülbíráلhatjuk a később bemutatott módon.

HÁTTÉRKÉP BEÁLLÍTÁSA

A **background-image** tulajdonság

CSS példa – háttérkép beállítása az oldalra vonatkozóan

```
body {background-image:url("kepek/css3logo.png");}
```



A háttérkép a CSS3 szabvány logóját tartalmazza.



A mozaikszerű elrendezés miatt a képek egymás mellé és alá kerülnek.

Nem minden kép esetén lesz vizuálisan megfelelő az eredmény.

HÁTTÉRKÉP BEÁLLÍTÁSA

A `background-image` tulajdonság

CSS példa – háttérkép beállítása az oldalra vonatkozóan

```
body {background-image:url("kepek/bg.jpg");}
```



Speciális kialakítású
(úgynevezett *seamless*)
háttérkép.



A speciális háttérkép
ismétlődésekor nem
vesszük észre az
átmeneteket.

Olyan, mintha egy nagy
kép lenne háttérként
beállítva.

HÁTTÉRKÉP ISMÉTLŐDÉSE

A **background-repeat** tulajdonság

Értékek:

- **repeat**: a háttérkép mozaikszerűen lesz elrendezve.
- **repeat-x**: a háttérkép csak vízszintesen ismétlődik.
- **repeat-y**: a háttérkép csak függőlegesen ismétlődik.
- **no-repeat**: a háttérkép csak egyszer jelenik meg, ismétlődés nélkül.

HÁTTÉRKÉP ISMÉTLŐDÉSE

A **background-repeat** tulajdonság

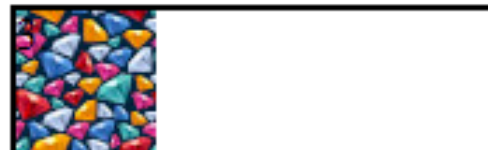
CSS példa



background-repeat:
no-repeat;



background-repeat:
repeat-x;



background-repeat:
repeat-y;



background-repeat:
repeat;

HÁTTÉRKÉP KEZDŐPOZÍCIÓJA

A **background-position** tulajdonság

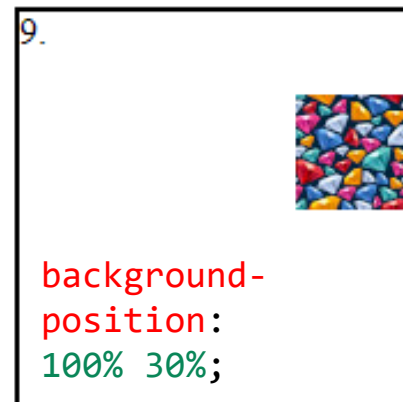
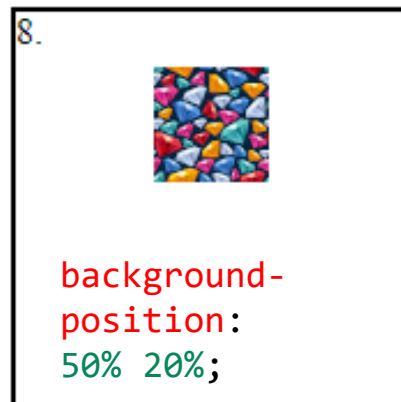
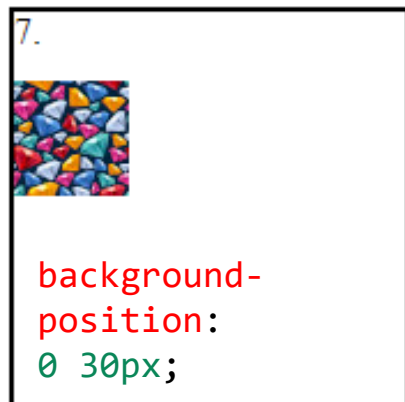
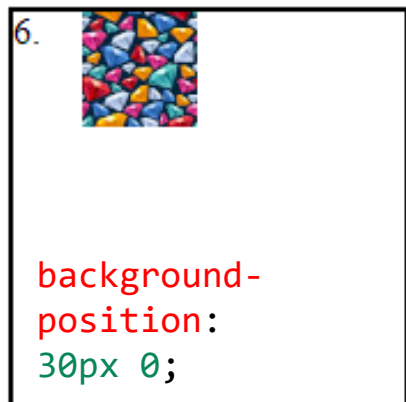
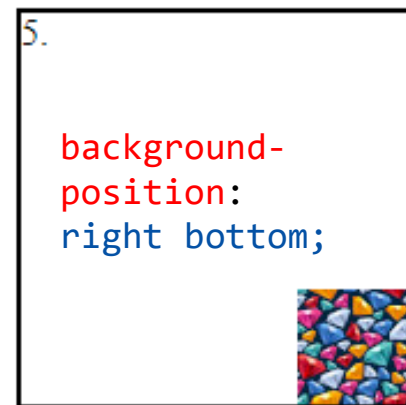
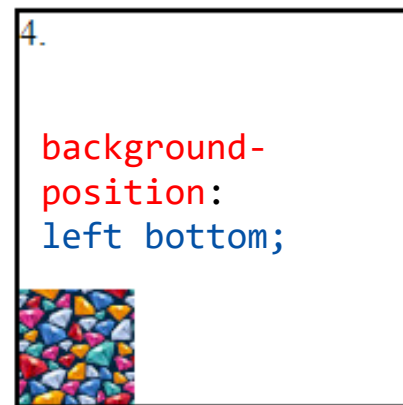
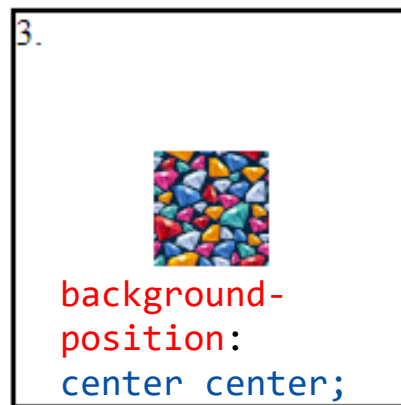
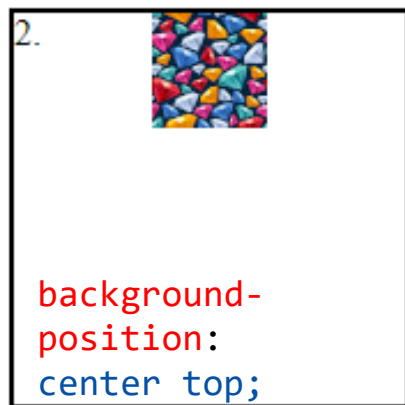
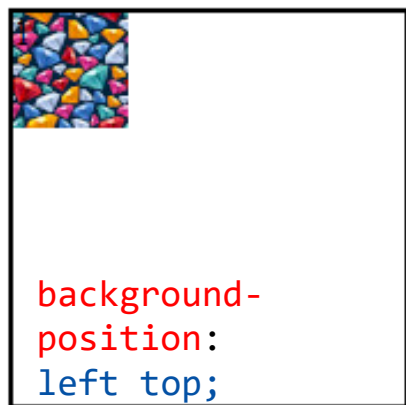
A háttérkép kezdőpozícióját állíthatjuk be.

Az első paraméter a **vízszintes**, a második a **függőleges** igazítást állítja be. Megadhatunk *hosszúságértéket, százalékos értéket* és az alábbi *kulcsszavakat* is.

- **top**: az elem tetejére igazítja a hátteret.
- **bottom**: az elem aljára igazítja a hátteret.
- **left**: az elem bal oldalára igazítja a hátteret.
- **right**: az elem jobb oldalára igazítja a hátteret.
- **center**: az elem közepére igazítja a hátteret.

HÁTTÉRKÉP KEZDŐPOZÍCIÓJA

A `background-position` tulajdonság



HÁTTÉRKÉP

Nézzük meg, hogy hol van példa háttérkép megadására a stíuslapállományban!

SZÖVEG ÉS BETŰK FORMÁZÁSA

BETŰTÍPUS BEÁLLÍTÁSA

A **font-family** paraméter

- Értéke egy vesszővel elválasztott lista, amely betűtípusok neveit tartalmazza. (A lista akár egy elemből is állhat).
- Ha a sorrendben első betűtípus elérhető, azzal jeleníti meg a böngésző a szöveget. Ha nem, akkor a sorrendben következő szerint, és így tovább.
- A lista végén egy általános betűcsaládot célszerű megadni. Ha egyik betűtípus sem érhető el, akkor az általános betűcsaládba tartozó alapértelmezett betűtípussal jeleníti meg a böngésző a szöveget.

BETŰTÍPUS BEÁLLÍTÁSA

A `font-family` paraméter

CSS példa

Ha a betűtípus nevében szóköz van, akkor azt aposztróf jelek között kell elhelyezni!

```
font-family:  
Times, 'Times New Roman', Georgia, serif;
```

```
font-family:  
Verdana, Arial, Helvetica, sans-serif;
```

```
font-family:  
Courier, 'Lucida Console', monospace;
```

Szöveg

Szöveg

Szöveg

BETŰTÍPUS BEÁLLÍTÁSA

Gyakran használt általános betűcsaládok

- **Serif:** talpas betűk. Használatuk főleg nyomtatásban ajánlott.
- **Sans-serif:** talp nélküli, modernebb megjelenés. A képernyőn jobban mutat, mint a Serif betűtípusok.
- **Monospace:** minden karakter azonos szélességű, így írógéppel írt hatást kelt.
- **Cursive:** kézírásra emlékeztető betűtípus. Nehezen olvashatók, ezért általában csak dekoratív célokat szolgálnak.

Teszt szöveg

Teszt szöveg

Teszt szöveg

Teszt szöveg

EGYSÉGES BETŰTÍPUSOK MINDEN PLATFORMON

A **fonts.google.com** szolgáltatás



Architects Daughter
Designed by Kimberly Geswein

< Select styles Glyphs >

Styles

Type here to preview text
Teszt szöveg

Size: 30px

Regular 400
Teszt szöveg — Remove this style

Review **Embed**

To embed a font, copy the code into the <head> of your html

<link> @import

```
<link href="https://fonts.googleapis.com/css2?family=Architects+Daughter&display=swap" rel="stylesheet">
```

CSS rules to specify families

```
font-family: 'Architects Daughter', cursive;
```

Ezzel tudjuk belinkelni a betűtípus használatához szükséges állományt.

Így tudjuk beállítani a betűtípust.

BETŰMÉRET BEÁLLÍTÁSA

A `font-size` paraméter

- Abszolút méret megadása
 - Növekvő sorrendben: `xx-small`, `x-small`, `small`, `medium` (ez az alapértelmezett), `large`, `x-large`, `xx-large`, `xxx-large`
- Relatív érték megadása
 - `smaller` (kisebb), `larger` (nagyobb) kulcsszavakkal.
- Megadhatunk hosszúságértékeket (pl. `12px`, `1.2em`)
- Továbbá százalékos értékeket is. (pl. `120%`)

BETŰMÉRET BEÁLLÍTÁSA

A **font-size** paraméter

CSS példa

```
.fs1 {font-size: xxx-large;}  
.fs2 {font-size: xx-large;}  
.fs3 {font-size: x-large;}  
.fs4 {font-size: large;}  
.fs5 {font-size: medium;}  
.fs6 {font-size: small;}  
.fs7 {font-size: x-small;}  
.fs8 {font-size: xx-small;}
```

Szöveg

Szöveg

Szöveg

Szöveg

Szöveg

Szöveg

Szöveg

Szöveg

BETŰMÉRET BEÁLLÍTÁSA

A **font-size** paraméter

CSS példa

```
.fs1 {font-size: larger;}  
.fs2 {font-size: smaller;}  
.fs3 {font-size: 32px;}  
.fs4 {font-size: 300%;}  
.fs5 {font-size: 1.5em;}
```

Szöveg

Szöveg

Szöveg

Szöveg

Szöveg

BETŰSTÍLUS BEÁLLÍTÁSA

A **font-style** paraméter

Értékek

- **oblique**: dőlt stílusú, az eredeti betűk megdöntésével.
- **italic**: az aktuális betűtípus dőlt variánsa jelenik meg, amennyiben van ilyen a betűkészletben.
- **normal**: normál stílusúra állítja (vissza) a betűstílust.

BETŰSTÍLUS BEÁLLÍTÁSA

A **font-style** paraméter

CSS példa

```
.fst1 {font-style: normal;}
```

```
.fst2 {font-style: oblique;}
```

```
.fst3 {font-style: italic;}
```

Szöveg Normal

Szöveg Oblique

Szöveg Italic

DŐLT STÍLUS

Állítsuk be, hogy az eleje azonosítóval ellátott elem dőlt legyen!

BETŰSÚLY (VASTAGSÁG) BEÁLLÍTÁS

A **font-weight** paraméter

Gyakran használt értékek

- **bold**: vastag betűk beállítása.
- **normal**: normál stílusúra állítja (vissza) a betűvastagságot.

CSS példa

```
.fw1 {font-weight: bold;}
```

```
.fw2 {font-weight: normal;}
```

Szöveg

Szöveg

BETŰSÚLY (VASTAGSÁG) BEÁLLÍTÁS

A **font-weight** paraméter

A súly (vastagság) számértékekkel is megadható.

Az értékek: 100, 200, 300, 400, 500, 600, 700, 800, 900

CSS példa

```
.fw1 {font-weight:100}  
.fw2 {font-weight:200}  
.fw3 {font-weight:300}  
.fw4 {font-weight:400}  
.fw5 {font-weight:500}  
.fw6 {font-weight:600}  
.fw7 {font-weight:700}  
.fw8 {font-weight:800}  
.fw9 {font-weight:900}
```

Szöveg

Szöveg

Szöveg

Szöveg

Szöveg

Szöveg

Szöveg

Szöveg

Szöveg

Thin (Hairline)

Extra Light (Ultra Light)

Light

Normal (Regular)

Medium

Semi Bold (Demi Bold)

Bold

Extra Bold (Ultra Bold)

Black (Heavy)

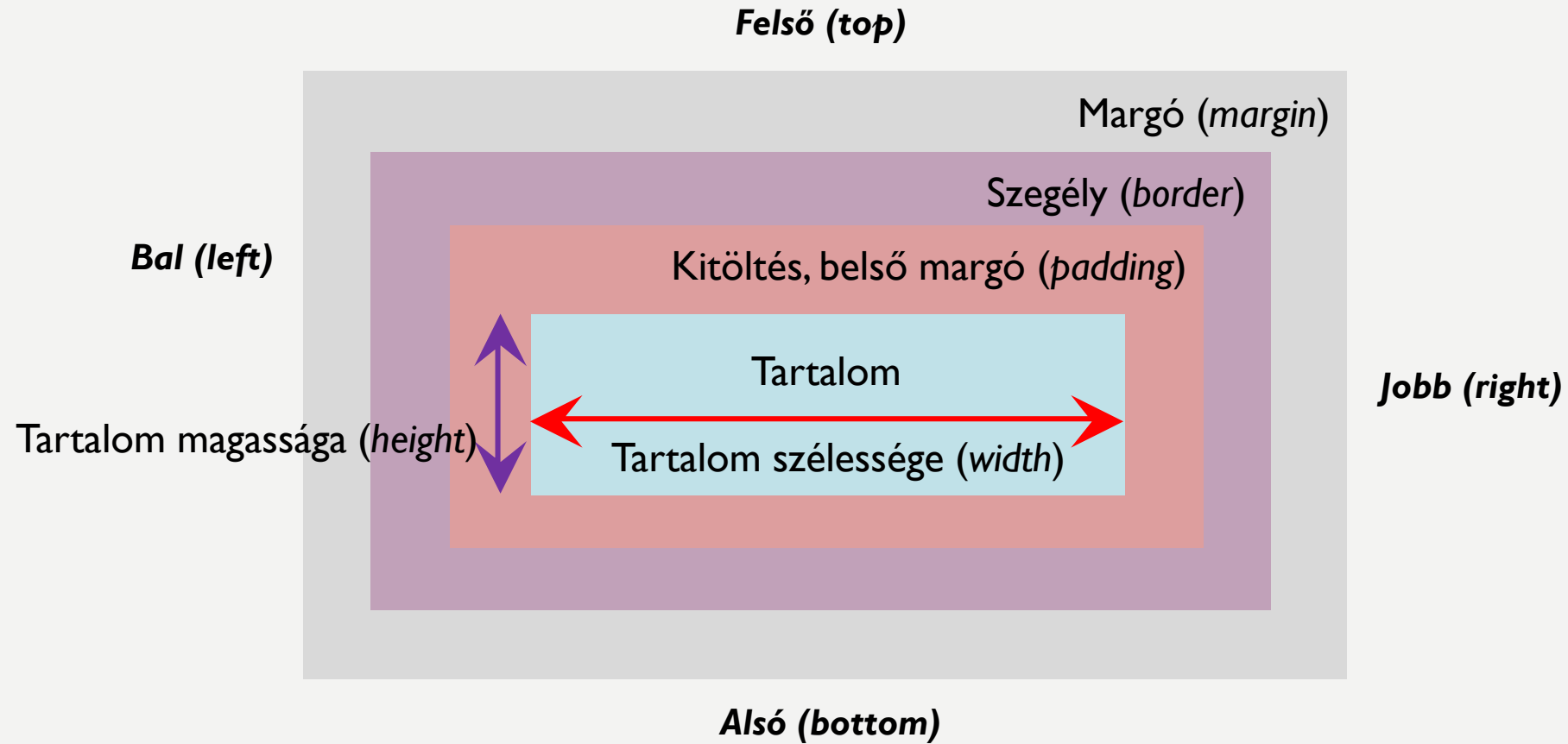
Extra Black (Ultra Black)

FÉLKÖVÉR STÍLUS

Állítsuk be, hogy a listaelemekben (`<li>`) lévő szövegek félkövér stílussal jelenjenek meg!

DOBOZMODELL (FORMÁZÁSMODELL)

AZ ALAPÉRTELMEZETT DOBOZMODELL



SZEGÉLY BEÁLLÍTÁSA

A **border** paraméter használata

Így állíthatunk be szegélyt az elem körül (mind a négy oldalra)

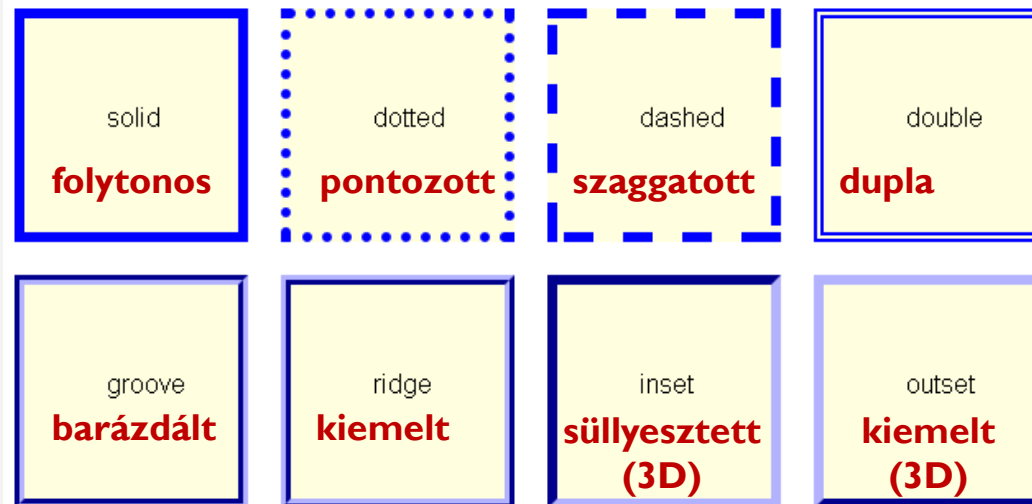
border: szegélyvastagság stílus szín;

Kulcsszavak

thin: vékony
medium: közepes
thick: vastag

Konkrét méret képpontban

0 nincs szegély
1px 1 képpontos
5px 5 képpontos



Színnevek

darkblue
red
stb.

Színkódok

#ffcc23
rgb(232,45,7)
stb.

SZEGÉLY BEÁLLÍTÁSA

Példák

1. CSS kód

```
<style>
  p {
    border:1px solid blue;
  }
</style>
```

Ez egy bekezdés

2. CSS kód

```
<style>
  p {
    border:thick double #ddd;
  }
</style>
```

Ez egy bekezdés

HTML kód

```
<body>
  <p>Ez egy bekezdés</p>
</body>
```

SZEGÉLY BEÁLLÍTÁSA

Állítsuk be a *fontos* osztályba tartozó elemre, hogy két képpont vastag fekete szegély vegye körül!

SZEGÉLY BEÁLLÍTÁSA

Eltérő szegély a 4 oldalon

A felső (*top*), jobb (*right*), alsó (*bottom*), bal (*left*) oldalakra külön-külön is beállíthatjuk a szegélyt.

border-top: szegélyvastagság stílus szín;
border-right: szegélyvastagság stílus szín;
border-bottom: szegélyvastagság stílus szín;
border-left: szegélyvastagság stílus szín;

CSS kód

```
<style>
  p {
    border-top: 1px dotted black;
    border-right: 4px solid red;
    border-bottom: 1px dashed gray;
    border-left: 1px solid red;
  }
</style>
```

HTML kód

```
<body>
  <p>Ez egy bekezdés</p>
</body>
```

Ez egy bekezdés

SZEGÉLY BEÁLLÍTÁSA

Eltérő szegély a 4 oldalon

A vastagság, stílus és szín megadására külön paraméterek is használhatóak a 4 oldalra.

CSS kód

```
<style>
  p {
    border-top-width: 2px;
    border-top-style: dotted;
    border-top-color: green;
  }
</style>
```

Ez egy bekezdés

HTML kód

```
<body>
  <p>Ez egy bekezdés</p>
</body>
```

Vastagság:	border-top-width border-right-width border-bottom-width border-left-width
Stílus:	border-top-style border-right-style border-bottom-style border-left-style
Szín:	border-top-color border-right-color border-bottom-color border-left-color

SZEGÉLY LEKEREKÍTÉSE

HTML kód

```
<body>  
  <p>Ez egy bekezdés</p>  
</body>
```

A `border-radius` paraméter

A szegély sarkait lekerekíthetjük mind a négy oldalon, az alábbi paraméterrel. A lekerekítés mértéke képpontban vagy százalékban is megadható.

`border-radius: lekerekítés_mértéke;`

1. CSS kód

```
<style>  
  p {  
    border:1px solid blue;  
    border-radius:5px;  
  }  
</style>
```

Ez egy bekezdés

2. CSS kód

```
<style>  
  p {  
    border:1px solid blue;  
    border-radius:8%;  
  }  
</style>
```

Ez egy bekezdés

SZEGÉLY LEKEREKÍTÉSE

A négy sarok lekerekítésének mértékét külön-külön is be lehet állítani.

border-top-left-radius: bal felső sarok lekerekítése;
border-top-right-radius: jobb felső sarok lekerekítése;
border-bottom-right-radius: jobb alsó sarok lekerekítése;
border-bottom-left-radius: bal alsó sarok lekerekítése;

CSS kód

```
<style>
p {
  border: 1px solid blue;
  font-size: 2em;
  border-top-left-radius: 10px;
  border-top-right-radius: 5px;
  border-bottom-right-radius: 1px;
  border-bottom-left-radius: 3px;
}
</style>
```

Ez egy bekezdés.

HTML kód

```
<body>
  <p>Ez egy bekezdés</p>
</body>
```

Ugyanez a border-radius paraméterrel is beállítható, ha négy értéket adunk meg, a következő sorrendben:

border-radius: balfelső jobbfelső jobbalsó balalsó;

CSS kód

```
<style>
p {
  border: 1px solid blue;
  font-size: 2em;
  border-radius: 10px 5px 1px 3px;
}
</style>
```

SZEGÉLY LEKEREKÍTÉSE

Állítsuk be a *fontos* osztályba tartozó elemre, hogy a szegély lekerekítésének mértéke 20px legyen!

SZÉLESSÉG ÉS MAGASSÁG

HTML kód

```
<body>  
  <p>Ez egy bekezdés</p>  
</body>
```

A **width** és **height** paraméter

A méret képpontokban és %-os értékben is megadható. A 70% azt jelenti, hogy a szülő elem méretének 70%-a lesz az elem mérete.

1. CSS kód

```
<style>  
  p {  
    width:150px;  
    height:100px;  
    border:1px solid blue;  
  }  
</style>
```

Ez egy bekezdés

2. CSS kód

```
<style>  
  p {  
    width:70%;  
    height:100px;  
    border:1px solid blue;  
  }  
</style>
```

Ez egy bekezdés

Elem szélessége (70%)

Tartalmazó elem szélessége (100%)

SZÉLESSÉG ÉS MAGASSÁG

Minimális és maximális méretek

Azt is megadhatjuk, hogy mi legyen az elem minimális és maximális mérete.

- **min-width**: minimális szélesség
- **max-width**: maximális szélesség
- **min-height**: minimális magasság
- **max-height**: maximális magasság

CSS kód

```
<style>
  p {
    width:70%;
    min-width:200px;
    max-width:600px;
    border:1px solid blue;
  }
</style>
```

HTML kód

```
<body>
  <p>Ez egy bekezdés</p>
</body>
```

A bekezdés szélessége a szülő elem szélességének 70%-a lesz, de nem mehet 200 képpont alá, és nem lehet szélesebb 600 képpontnál.

SZÉLESSÉG BEÁLLÍTÁSA

Nézzük meg és értelmezzük a `video` címkére vonatkozó beállításokat a stíluslapban!

KITÖLTÉS (BELSŐ MARGÓ)

HTML kód

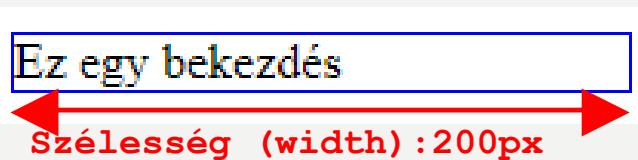
```
<body>  
  <p>Ez egy bekezdés</p>  
</body>
```

A **padding** paraméter

A belső margó a tartalom és a szegély közti távolságot jelenti.

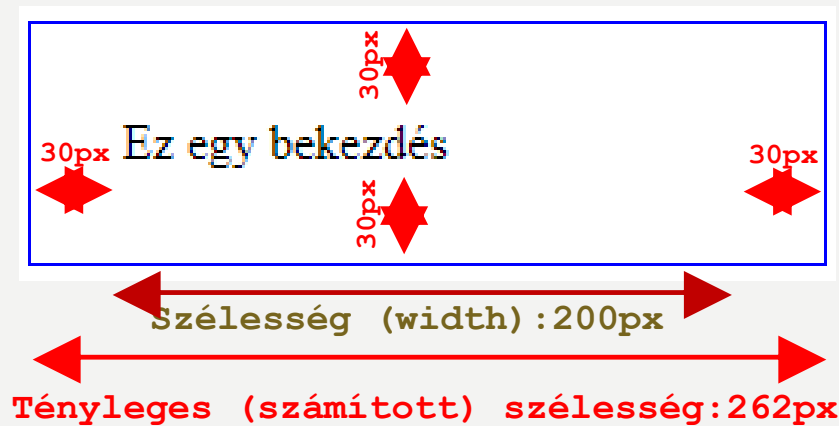
1. CSS kód

```
<style>  
  p {  
    width:200px;  
    border:1px solid blue;  
    padding:0;  
  }  
</style>
```



2. CSS kód

```
<style>  
  p {  
    width:200px;  
    border:1px solid blue;  
    padding:30px;  
  }  
</style>
```



KITÖLTÉS (BELSŐ MARGÓ)

Eltérő értékek az oldalakon

A kitöltés értékét külön beállíthatjuk mind a négy oldalra.

- **padding-top**: felső kitöltés
- **padding-right**: jobb oldali kitöltés
- **padding-bottom**: alsó kitöltés
- **padding-left**: bal oldali kitöltés

HTML kód

```
<body>  
  <p>Ez egy bekezdés</p>  
</body>
```

CSS kód

```
<style>  
  p {  
    border:1px solid blue;  
    padding-top:20px;  
    padding-right:10px;  
    padding-bottom:5px;  
    padding-left:15px;  
  }  
</style>
```

Ez egy bekezdés

KITÖLTÉS BEÁLLÍTÁSA

1. Nézzük meg és értelmezzük a fontos osztályra vonatkozó beállításokat a stíluslapban!
2. Miért van jóval nagyobb bal oldali kitöltés beállítva, mint a többi oldalra?

MARGÓ

A margin paraméter

A margó az elemek közti térköz beállítására szolgál.

I. CSS kód

```
<style>
p {
    width:200px;
    border:1px solid blue;
}
</style>
```

Ez egy bekezdés

Ez egy bekezdés

2. CSS kód

```
<style>
p {
    width:200px;
    border:1px solid blue;
    margin:0;
}
</style>
```

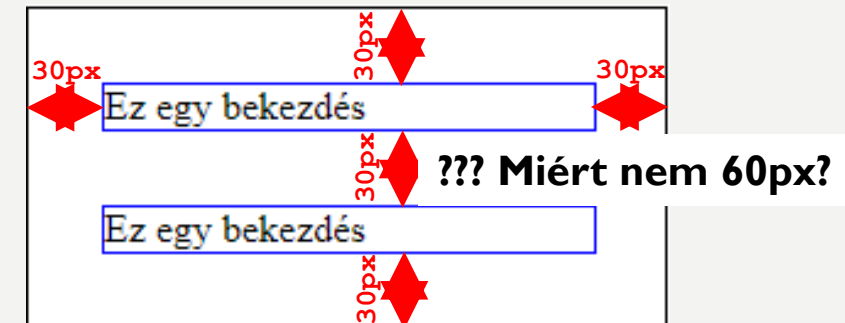
Ez egy bekezdés

HTML kód

```
<body>
  <p>Ez egy bekezdés</p>
  <p>Ez egy bekezdés</p>
</body>
```

3. CSS kód

```
<style>
p {
    width:200px;
    border:1px solid blue;
    margin:30px;
}
</style>
```



MARGÓ ÖSSZEVONÁS

Függőleges margó összevonás

A szabvány azt mondja ki, hogy az **egymás alatti** dobozok felső és alsó margóját össze kell vonnia a böngészőnek. Mégpedig úgy, hogy a felső és alsó margó értékek közül a nagyobbát kell beállítani.



A két bekezdés között 15px térköz lenne.



A margó összevonás után 10 képpont lesz a margó, mert az 5 és 10 közül a 10 a nagyobb.

MARGÓ

Eltérő értékek az oldalakon

A margó értékét is külön beállíthatjuk mind a négy oldalra.

- **margin-top:** felső margó
- **margin-right:** jobb oldali margó
- **margin-bottom:** alsó margó
- **margin-left:** bal oldali margó

HTML kód

```
<body>  
  <p>Ez egy bekezdés</p>  
  <p>Ez egy bekezdés</p>  
</body>
```

CSS kód

```
<style>  
p {  
  border:1px solid blue;  
  margin-top:20px;  
  margin-right:10px;  
  margin-bottom:5px;  
  margin-left:15px;  
}  
</style>
```

Ez egy bekezdés

Ez egy bekezdés

MARGÓ BEÁLLÍTÁSA

Állítsuk be a `body` elemre, hogy a felső és az alsó margó is `50px` legyen!

KÖZÉPRE IGAZÍTÁS MARGÓVAL

Ha egy blokkszintű elemet középre szeretnénk igazítani, akkor azt is margóbeállítással tehetjük meg. Ekkor a bal és jobb oldali margónak **auto** értéket kell beállítani.

- **margin-left: auto;**
- **margin-right: auto;**

Ha alul is felül nulla a margó értéke, bal és jobb oldalon **auto**, akkor azt röviden így is írhatjuk:

- **margin: 0 auto;**

HTML kód

```
<body>  
  <p>Ez egy bekezdés</p>  
  <p>Ez egy bekezdés</p>  
</body>
```

CSS kód

```
<style>  
p {  
  margin: 0 auto;  
  border: 1px solid blue;  
}  
</style>
```

A bekezdések középre lesznek igazítva a tartalmazó elemekben.

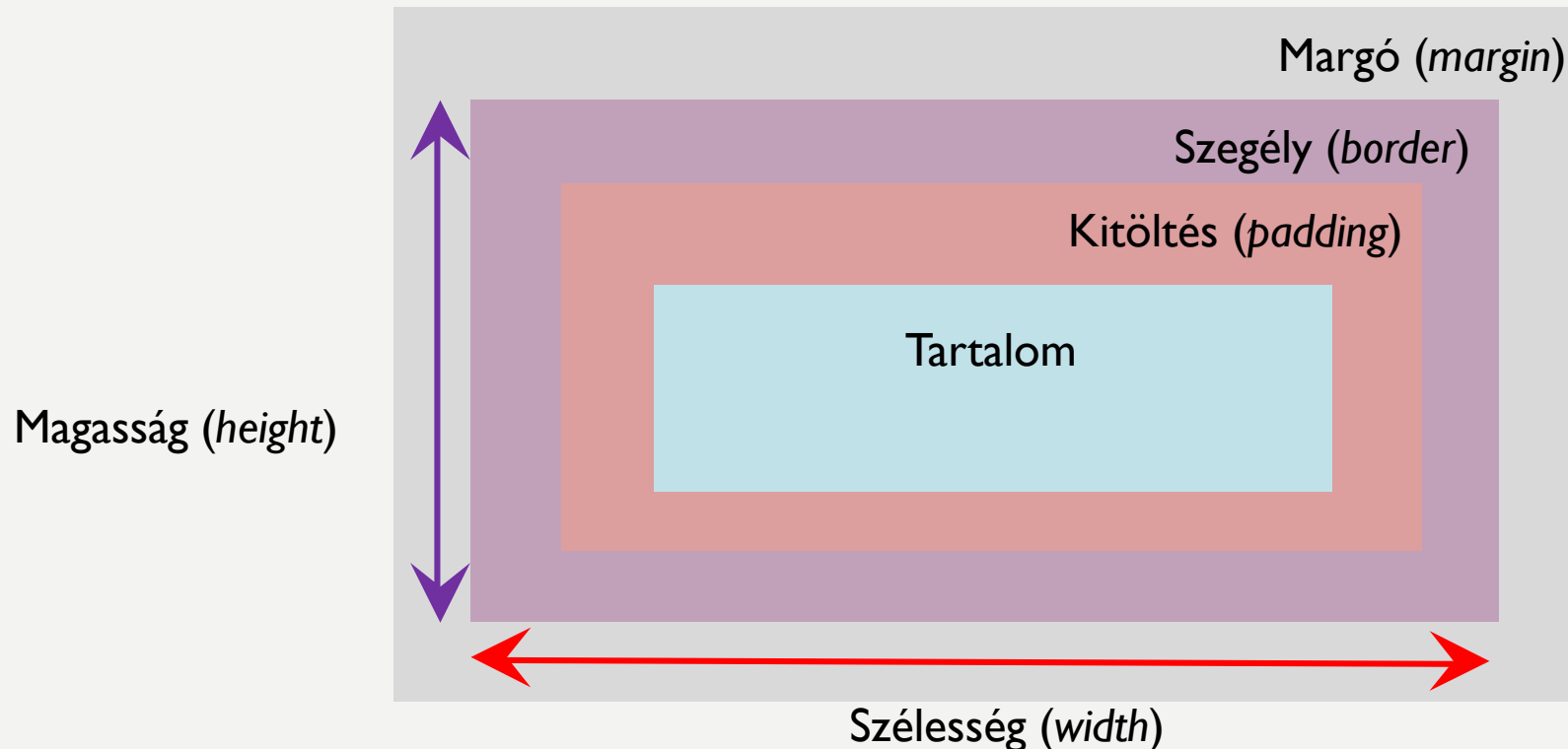
Ez egy bekezdés
Ez egy bekezdés

KÖZÉPRE IGAZÍTÁS

1. Szüntessük meg a body elem középre igazítását!
2. Helyette állítsunk be 100px bal margót!

DOBOZMÉRETEZÉS MÓDOSÍTÁSA

Egyes esetekben kényelmesebb lenne a méretmegadás, ha a szélesség (width) és magasság (height) értékekbe a kitöltés és a szegély mérete is beleszámítana. Ehhez a módosításhoz a **box-sizing**: **border-box**; paramétert kell használni. (Az alapértelmezett érték a **content-box**)



DOBOZMÉRETEZÉSEK ÖSSZEHAISONLÍTÁSA

Példa

CSS kód

```
<style>
div {
  width:200px;
  border:10px solid blue;
  padding:10px;
  margin:10px;
}
```

```
div#blokk1 {box-sizing: content-box;}
```

```
div#blokk2 {box-sizing: border-box;}
</style>
```

HTML kód

```
<body>
  <div id="blokk1">Ez az 1. blokk</div>
  <div id="blokk2">Ez a 2. blokk</div>
</body>
```

Ez az 1. blokk

Ez a 2. blokk

A decorative graphic on the left side of the slide, consisting of two parallel, wavy vertical lines. The inner line is a light blue color, and the outer line is white. They are set against a dark blue background.

ELEMEK LEBEGTETÉSE

ELEMENK LEBEGTETÉSE

A **float** paraméter

Az elemeket balra (**left**) és jobbra (**right**) igazíthatjuk (lebegtethetjük). Ilyenkor a környező elemek körbefolyják az adott elemet. Az alapértelmezett érték: **none** (az elem nem lebeg)



Lorem ipsum dolor sit amet, consectetur adipiscing elit. Aenean eu mi mattis odio suscipit cursus vitae porttitor purus. Curabitur sit amet pellentesque nisi. Aliquam erat volutpat. Proin lectus nulla, tincidunt eu varius id, efficitur eu lectus. Nunc efficitur tellus sit amet interdum faucibus. Duis urna neque, vulputate et magna at, imperdiet cursus eros. Pellentesque ac leo dignissim, volutpat magna nec, molestie ante. Cras consectetur elementum dolor, quis venenatis tellus pretium ac. Quisque molestie posuere sollicitudin.

Balra lebegtetett kép

Lorem ipsum dolor sit amet, consectetur adipiscing elit. Aenean eu mi mattis odio suscipit cursus vitae porttitor purus. Curabitur sit amet pellentesque nisi. Aliquam erat volutpat. Proin lectus nulla, tincidunt eu varius id, efficitur eu lectus. Nunc efficitur tellus sit amet interdum faucibus. Duis urna neque, vulputate et magna at, imperdiet cursus eros. Pellentesque ac leo dignissim, volutpat magna nec, molestie ante. Cras consectetur elementum dolor, quis venenatis tellus pretium ac. Quisque molestie posuere sollicitudin.



Jobbra lebegtetett kép

ELEMENK LEBEGTETÉSE

Példa

HTML kód

```
<p>  
  
Lorem ipsum dolor sit amet, consectetur adipiscing elit. Aenean eu mi mattis odio suscipit cursus vitae porttitor purus...  
</p>  
<p>  
  
Lorem ipsum dolor sit amet, consectetur adipiscing elit. Aenean eu mi mattis odio suscipit cursus vitae porttitor purus...  
</p>
```

CSS kód

```
<style>  
#kep1 {width:100px;  
        float:left;  
        margin-right:10px;}  
  
#kep2 {width:100px;  
        float:right;  
        margin-left:10px;}  
</style>
```



Lorem ipsum dolor sit amet, consectetur adipiscing elit. Aenean eu mi mattis odio suscipit cursus vitae porttitor purus. Curabitur sit amet pellentesque nisi. Aliquam erat volutpat. Proin lectus nulla, tincidunt eu varius id, efficitur eu lectus. Nunc efficitur tellus sit amet interdum faucibus. Duis urna neque, vulputate et magna at, imperdiet cursus eros. Pellentesque ac leo dignissim, volutpat magna nec, molestie ante. Cras consectetur elementum dolor, quis venenatis tellus pretium ac. Quisque molestie posuere sollicitudin.

Lorem ipsum dolor sit amet, consectetur adipiscing elit. Aenean eu mi mattis odio suscipit cursus vitae porttitor purus. Curabitur sit amet pellentesque nisi. Aliquam erat volutpat. Proin lectus nulla, tincidunt eu varius id, efficitur eu lectus. Nunc efficitur tellus sit amet interdum faucibus. Duis urna neque, vulputate et magna at, imperdiet cursus eros. Pellentesque ac leo dignissim, volutpat magna nec, molestie ante. Cras consectetur elementum dolor, quis venenatis tellus pretium ac. Quisque molestie posuere sollicitudin.



ELEMEK LEBEGTETÉSE

Nézzük meg, hogy hol van példa lebegtetésre a stíluslapállományban!

A KÖRBEFOLYATÁS MEGSZÜNTETÉSE

A **clear** paraméter

A lebegtetett elemek után következő elemekre beállíthatjuk, hogy nem akarjuk őket körbefolyatni a lebegtetett elem mellé.

A **clear** paraméter értékei:

- **left**: az elem nem kerülhet **balra** lebegtetett elem mellé
- **right**: az elem nem kerülhet **jobbra** lebegtetett elem mellé
- **both**: az elem nem kerülhet sem balra, sem jobbra lebegtetett elem mellé
- **none**: alapértelmezett érték. Az elem lebegtetett elem mellé kerül.

A KÖRBEFOLYATÁS MEGSZÜNTETÉSE

Példa

HTML kód

```

<p>Szöveg 1.</p>
<p id="szoveg">Szöveg 2.</p>
```

CSS kód

```
<style>
#kep1 {
    width:200px;
    float:left;
    margin-right:10px;
}

#szoveg {clear:left;}
</style>
```



Szöveg 1.

Szöveg 2.

Bár lenne még hely a kép mellett, a szöveg nem folyja körbe a képet, mert be lett állítva a `clear` tulajdonság.

HONLAP KÉSZÍTÉSE CMS RENDSZERBEN

CMS RENDSZEREK ELŐNYEI



Egyszerű szerkeszthetőség: A szöveges tartalmakat egy egyszerű, vizuális szerkesztőfelület használatával tudjuk megformázni, mintha csak egy szövegszerkesztő alkalmazást használnánk. A beillesztett képeket átméretezhetjük, sőt egyes rendszerekben haladóbb szerkesztési funkciók (felesleges területek levágása, elforgatás, tükrözés stb.) is elérhetők.



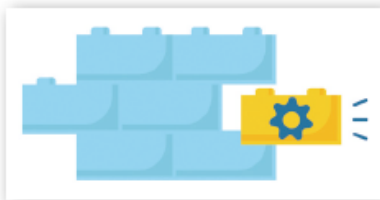
Tartalomkezelés: Lehetőség van a különböző tartalmak elmentésére és újbóli szerkesztésére. Ha már elérte a kívánt készültségi szintet az oldalunk, akkor publikálhatjuk azt. Az oldal csak ekkor válik ténylegesen elérhetővé a nagyobb közönség számára. A publikálást visszavonhatjuk, az idejétmúlt cikkeket pedig archiválhatjuk. Természetesen az egyes tartalmakat törölhetjük is, ha már nincs rájuk szükség.



Témák (sablonok), oldalelrendezések használata: az oldal kinézetét a különböző témák vagy sablonok kiválasztásával mi magunk határozhatjuk meg, ezenfelül többféle oldalelrendezést használhatunk a különböző jellegű tartalmak publikálásához. Ez az elrendezés határozza meg, hogy milyen legyen az oldal felépítése, például hogy hány hasázból álljon.

CMS RENDSZEREK ELŐNYEI

Bővíthető funkcionalitás: A rendszerekhez tartoz(hat)nak olyan beépülő modulok (*plug-in*), amelyeket telepítve újabb és újabb funkciók érhetők el. Például telepíthetünk olyan képgalériát, amely minden eszközön jól használható, responzív módon teszi lehetővé a képek, fotók nézegetését, lapozását.



Jogosultságkezelés: A rendszer felhasználóihoz különböző jogosultságokat lehet hozzárendelni. Elképzelhető, hogy egy szerzőnek csak egy tartalmi egység (oldal, bejegyzés) elkészítésére és elmentésére van joga, de azt publikálni nem tudja, mert ahhoz már szerkesztői jogosultságra lenne szüksége. A legtöbb joga az adminisztrátornak van, aki akár testre szabhatja az oldal kinézetét, vagy olyan modulokat telepíthet, amellyel a rendszer tudása bővíthető. Szintén ő az, aki a különböző felhasználókat felveheti a rendszerbe, vagy beállíthatja azt is, hogy szabadon lehessen regisztrálni az oldalra egy adott jogosultsággal, pl. olvasóként.



Munkafolyamatok menedzselése: Egy online újság esetén a szerkesztő (illetve a főszerkesztő) dönti el, hogy mikor mely cikkek legyenek publikálva. Az újságíró megírja a cikket, elmenti a rendszerben, amiről a szerkesztő értesítést kap. A szerkesztő visszaküldheti a cikket módosításra, vagy akár ő maga módosíthatja a cikket a publikálás előtt. Vagyis a rendszerben egyszerre több felhasználó tevékenykedik, eltérő szerepekkel. Emiatt fontos, hogy a rendszer jól támogassa a munkafolyamatokat.



CMS RENDSZEREK ELŐNYEI

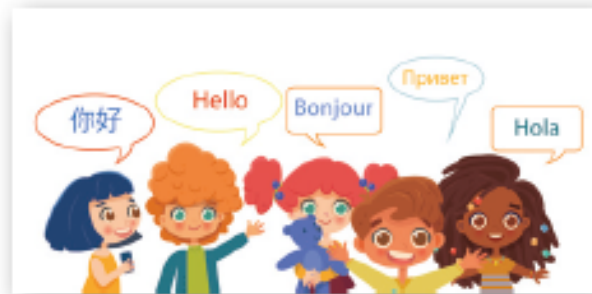
Együttműködés: A rendszerek különböző jellegű együttműködést is lehetővé tehetnek a felhasználók között. Elképzelhető, hogy ugyanazt a tartalmat többen állítják elő, egymással összedolgozva.



Verziókezelés: Fontos elvárás, hogy a tartalmak módosításai jól visszakövethetők legyenek. Emiatt a módosítások után új dokumentumverziók jönnek létre, lehetővé téve, hogy a korábbi változatokat vissza lehessen állítani, például egy véletlenül bekövetkezett törlés vagy felülírás miatt.



Többnyelvűség: A rendszerek biztosíthatják azt, hogy többnyelvű tartalmat állítsunk elő, vagyis hogy a portálnak legyen például magyar, angol és német nyelvű változata is. Ilyen esetben a tartalom feltöltésekor a különböző nyelvi változatokat is fel kell tölteni.



A HONLAP LÉTREHOZÁSÁNAK ÉS PUBLIKÁLÁSÁNAK LÉPÉSEI

- 1. Tartalom létrehozása, összegyűjtése, navigáció megtervezése
- 2. Regisztráció
- 3. Új webhely létrehozása, alapinformációk megadása
- 4. Téma (sablon, smink) kiválasztása
- 5. Oldalstruktúra meghatározása, oldalak létrehozása
- 6. Tartalom feltöltése, elrendezés kiválasztása, előnézet
- 7. Oldal publikálása
- 8. Oldal tesztelése, javítása

AJÁNLOTT CMS RENDSZEREK

WCMS neve és elérhetősége	Jellemzők	Felület nyelve
Google Sites https://sites.google.com/	Használatához Google-azonosítóval kell rendelkezünk.	Magyar, angol és más nyelveken is elérhető.
Webnode https://www.webnode.hu/	Tetszőleges e-mail-címmel regisztrálhatunk.	Magyar, angol és más nyelveken is elérhető.

ESETTANULMÁNY



A **Hold** a Nap után a **második legnagyobb fényességű** égitest az égbolton. Felszínét a Földről kis távcsővel, de akár szabad szemmel is viszonylag jól megfigyelhetjük.

A Hold forog a tengelye körül, és **ellipszis alakú pályán kering** a Föld körül. A Földdel együtt kering a Nap körül, és a Nappal együtt a Tejútrendszer központja körül is. Saját tengelye körül ugyanannyi idő alatt fordul meg, mint amennyi idő alatt befutja a Föld körüli pályáját (27,3 nap). Ez a magyarázata annak, hogy **mindig ugyanazt a félgömbjét mutatja a Föld felé**.



Érdekeségek a Holdról
Készült 2020.

Források:

- Földrajz 9. tankönyv <https://www.tankonyv.katalogus.hu/site/kiadvany/OH-FOL9TA>
- <https://hu.wikipedia.org/wiki/Hold>
- <https://pixabay.com/hu/photos/hold-sky-luna-viP/C/%A9gcvclem-142701/>
- <https://pixabay.com/hu/photos/a-holdP/C/%A9z1s-es1-csillag-holdP/C/%A9vc-106010/>
- <https://www.youtube.com/watch?v=cwL2YtU8dhw>
- https://www.youtube.com/watch?v=VIB5v_uC8Qo

WEBLAP LÉTREHOZÁSA GOOGLE SITES KÖRNYEZETBEN

Jelentkezzünk be a <https://sites.google.com/> portálra gmail azonosítónkkal és a hozzávalók felhasználásával (hold_honlap) hozzunk létre egy portált, amely a Holdról szól.



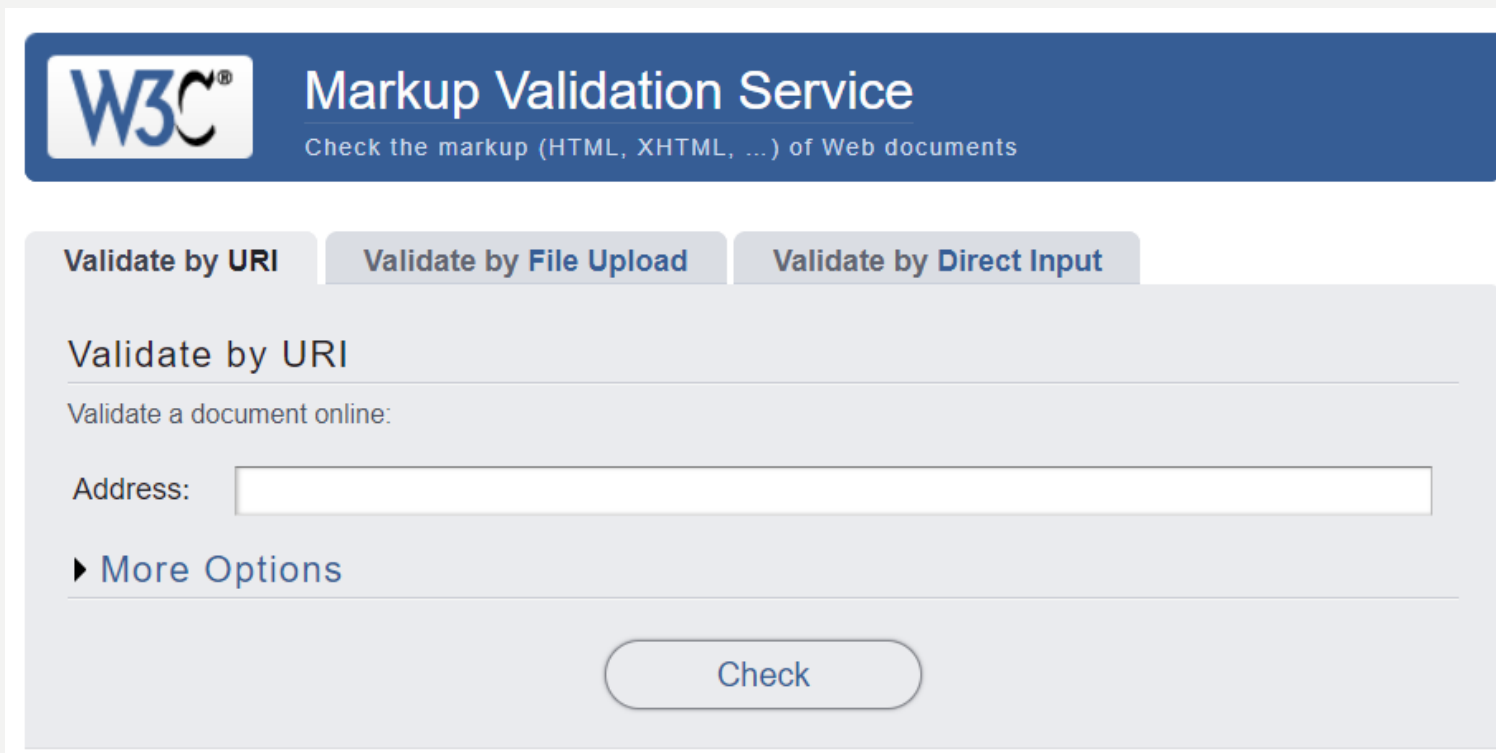
HTML OLDALAK SZABVÁNYOSSÁGA

A SZABVÁNYOK (AJÁNLÁSOK)

- A HTML leírónyelvek használati módját különböző ajánlások (szabványok) rögzítik.
- Ha nem felel meg az oldalunk a szabványoknak, akkor
 - Megjelenítési hibák léphetnek fel a böngészőben.
 - Akadálymentességi problémák léphetnek fel.
 - A hibákat a böngészők eltérően javítják, máshogy jelenhet meg az oldal különböző böngészőkben.
 - Az oldal lassabban jelenhet meg a böngészőben a javítások miatt.
 - Nem biztos, hogy a jövőben használt technológiák helyesen jelenítik meg.
 - Keresőoptimalizálási szempontból előnyösebb a szabványos oldal.

A SZABVÁNYOSSÁG ELLENŐRZÉSE

- A szabványosság ellenőrzését validálásnak nevezzük.
- A HTML5 szabvány (és más leíró nyelv szabványok) validálására használhatjuk a <https://validator.w3.org/> validáló eszközt.



The screenshot shows the W3C Markup Validation Service interface. At the top, there is a blue header with the W3C logo and the text "Markup Validation Service" and "Check the markup (HTML, XHTML, ...) of Web documents". Below the header, there are three tabs: "Validate by URI", "Validate by File Upload", and "Validate by Direct Input". The "Validate by URI" tab is selected. Under this tab, there is a section titled "Validate by URI" with the text "Validate a document online:". Below this, there is a label "Address:" followed by a text input field. At the bottom of the form, there is a link "More Options" and a "Check" button.

SZABVÁNYOSSÁG ELLENŐRZÉSE

Publikált weboldal validálása

The image shows the W3C Markup Validation Service interface. At the top, the W3C logo is on the left, and the text 'Markup Validation Service' is in the center, with a subtitle 'Check the markup (HTML, XHTML, ...) of Web documents' below it. Below this, there are three tabs: 'Validate by URI' (selected), 'Validate by File Upload', and 'Validate by Direct Input'. Under the 'Validate by URI' tab, the text 'Validate by URI' is followed by a horizontal line. Below this line, the text 'Validate a document online:' is followed by a text input field. A black callout box with white text '1. Írjuk be a mezőbe a webcímet!' points to the input field. Below the input field, the text 'Address:' is followed by a right-pointing arrow. Below the arrow, the text 'More Options' is followed by a right-pointing arrow. At the bottom, there is a 'Check' button. A purple callout box with white text '2. Kattintsunk az ellenőrzés gombra!' points to the 'Check' button.

W3C[®] Markup Validation Service
Check the markup (HTML, XHTML, ...) of Web documents

Validate by URI Validate by File Upload Validate by Direct Input

Validate by URI

Validate a document online:

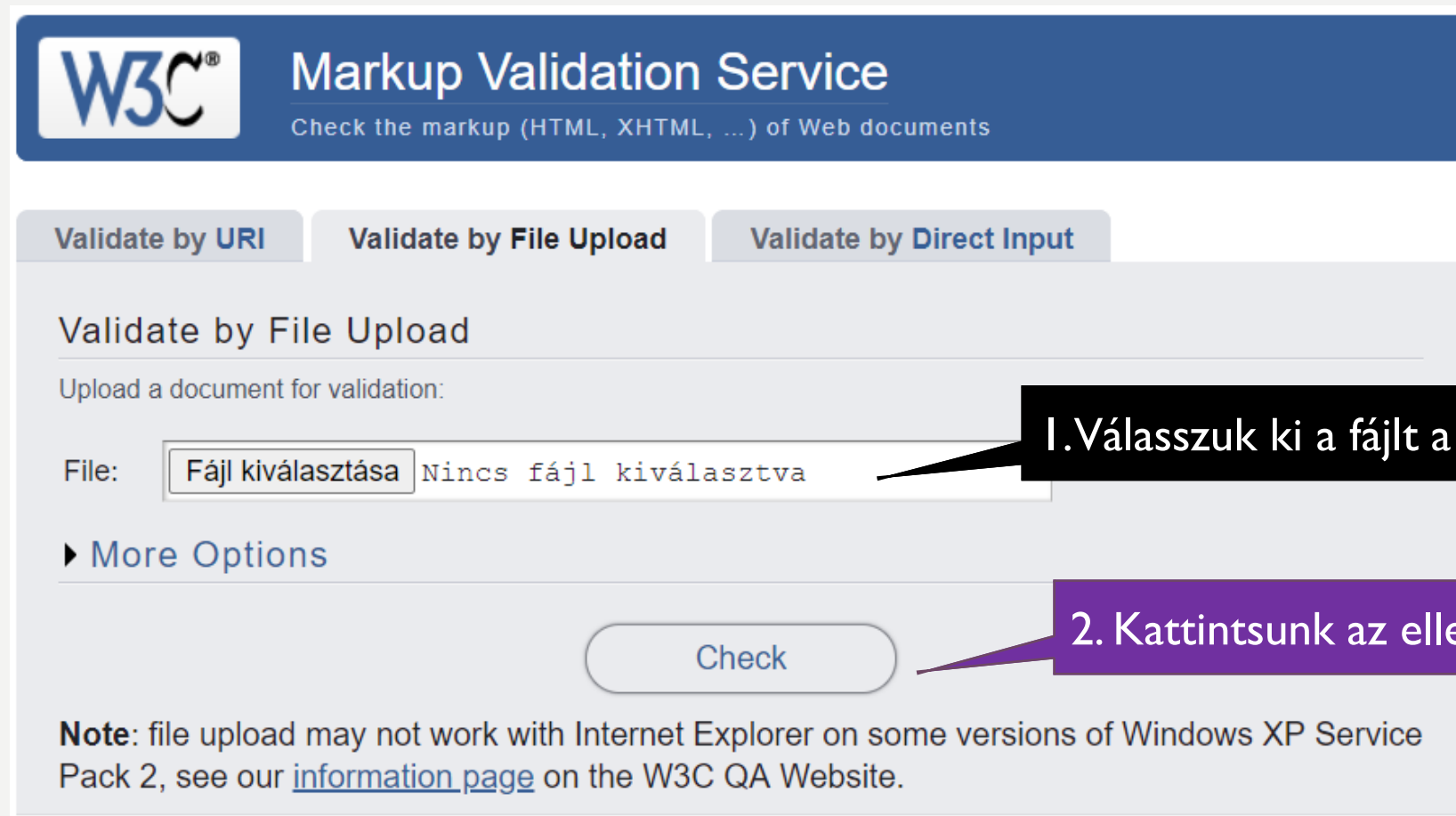
Address:

► More Options

Check

SZABVÁNYOSSÁG ELLENŐRZÉSE

Validálás fájl feltöltéssel



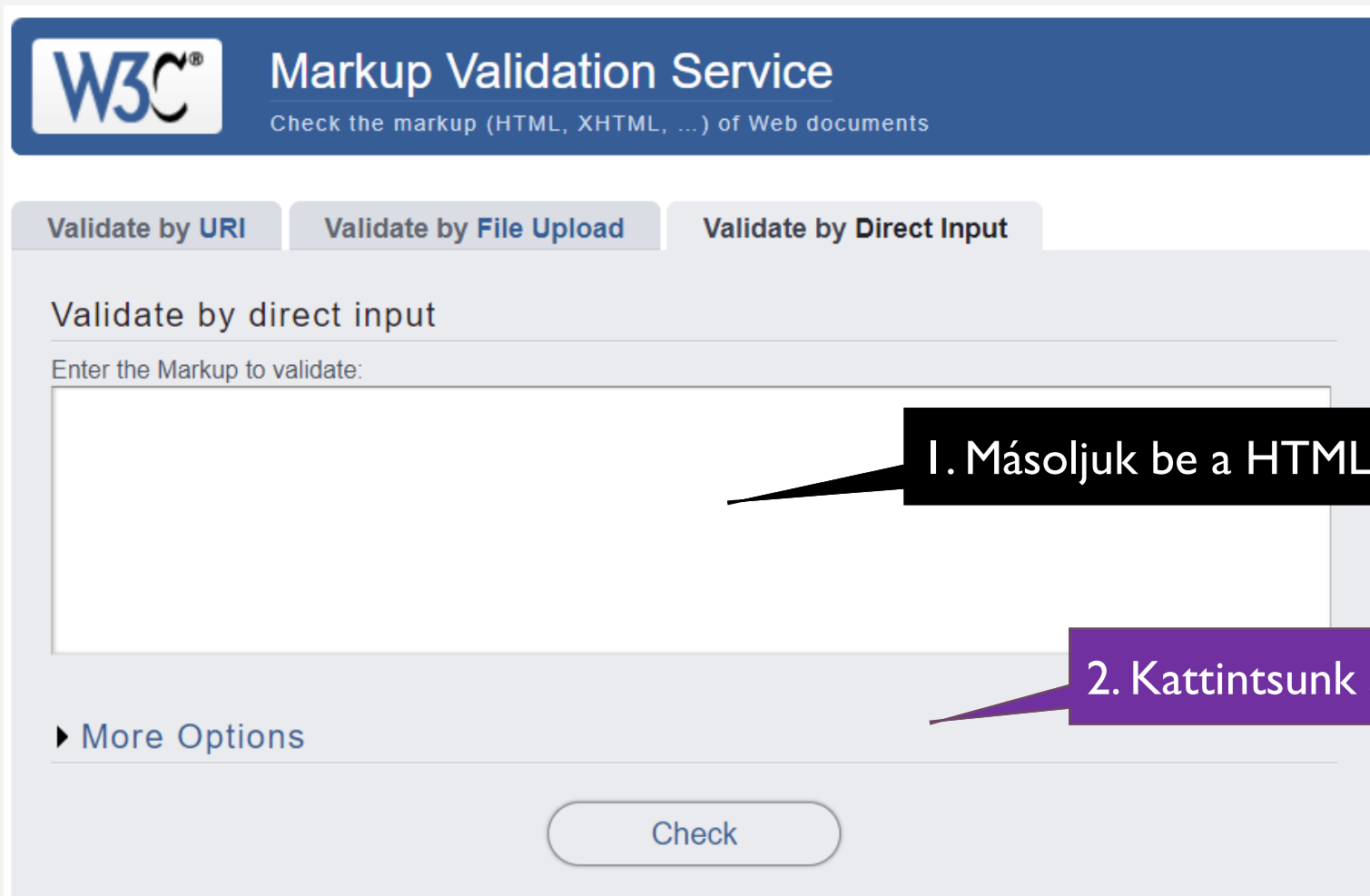
The screenshot shows the W3C Markup Validation Service interface. At the top, there is a blue header with the W3C logo and the text 'Markup Validation Service' and 'Check the markup (HTML, XHTML, ...) of Web documents'. Below the header, there are three tabs: 'Validate by URI', 'Validate by File Upload' (which is selected), and 'Validate by Direct Input'. Under the 'Validate by File Upload' tab, there is a section titled 'Validate by File Upload' with the instruction 'Upload a document for validation:'. Below this, there is a 'File:' label followed by a file selection button labeled 'Fájl kiválasztása' and the text 'Nincs fájl kiválasztva'. Below the file selection area, there is a link 'More Options'. At the bottom of the form, there is a 'Check' button. A note at the bottom states: 'Note: file upload may not work with Internet Explorer on some versions of Windows XP Service Pack 2, see our [information page](#) on the W3C QA Website.'

I. Válasszuk ki a fájlt a saját gépünkön!

2. Kattintsunk az ellenőrzés gombra!

SZABVÁNYOSSÁG ELLENŐRZÉSE

A beillesztett kód validálása



The image shows the W3C Markup Validation Service interface. At the top, there is a blue header with the W3C logo and the text "Markup Validation Service" and "Check the markup (HTML, XHTML, ...) of Web documents". Below the header, there are three tabs: "Validate by URI", "Validate by File Upload", and "Validate by Direct Input". The "Validate by Direct Input" tab is selected. Below the tabs, there is a section titled "Validate by direct input" with a text input field labeled "Enter the Markup to validate:". A large black callout bubble with a pointer to the input field contains the text "1. Másoljuk be a HTML kódot a mezőbe!". Below the input field, there is a link "More Options" and a "Check" button. A purple callout bubble with a pointer to the "Check" button contains the text "2. Kattintsunk az ellenőrzés gombra!".

W3C® Markup Validation Service
Check the markup (HTML, XHTML, ...) of Web documents

Validate by URI Validate by File Upload Validate by Direct Input

Validate by direct input

Enter the Markup to validate:

1. Másoljuk be a HTML kódot a mezőbe!

2. Kattintsunk az ellenőrzés gombra!

More Options

Check

SZABVÁNYOSSÁG ELLENŐRZÉSE

Az eredmény

Az eredmény többféle lehet

- Az oldalon **nincsenek** hibák (errors)
 - Az oldal **szabványos** (valid)
- Az oldalon csak figyelmeztetések (warning) vannak
 - Az oldal **szabványos** (valid)
- Az oldalon vannak hibák (errors)
 - Az oldal **nem szabványos**, vagyis nem valid.

SZABVÁNYOSSÁG ELLENŐRZÉSE

Az eredmény szabványos oldal esetén

I. Ha az oldal a régebbi (HTML5) előtti szabvány szerint készült

This document was successfully checked as XHTML 1.0 Strict!		
Result:	Passed	
Address :	https://www.w3.org/	
Encoding :	utf-8	(detect automatically) ▼
Doctype :	XHTML 1.0 Strict	(detect automatically) ▼
Root Element:	html	
Root Namespace:	http://www.w3.org/1999/xhtml	

1. Sikeres validálás (passed).

2. Látható, hogy melyik szabvány szerint történt a validálás. Ezt az adott oldal dokumentumtípusa határozza meg.

SZABVÁNYOSSÁG ELLENŐRZÉSE

Az eredmény szabványos oldal esetén

2. Ha az oldal a HTML5 szabvány szerint készült, egy másik felület jelenik meg (Nu html checker).

Nu Html Checker

This tool is an ongoing experiment in better HTML checking, and its behavior remains subject to change

Showing results for contents of text-input area

Checker Input

Show ☒ source ☐ outline ☐ image report [Options...](#)

Check by ☐ css

```
<!DOCTYPE html>
<html lang="hu">
<head>
<title>Bekezdés példa</title>
</head>
<body>
<p>Magy Imola<br>egyéni vállalkozó<br>Debrecen</p>
</body>
</html>
```

[Check](#)

Use the Message Filtering button below to hide/show particular messages, and to see total counts of errors and warnings.

[Message Filtering](#)

Document checking completed. No errors or warnings to show.

Source

```
1. <!DOCTYPE html>+
2. <html lang="hu">+
3. <head>+
4. <title>Bekezdés példa</title>+
5. </head>+
6. <body>+
7. <p>Magy Imola<br>egyéni vállalkozó<br>Debrecen</p>+
8. </body>+
9. </html>+
10. 
```

Sikeres validálás! Nincsenek hibák, sem figyelmeztetések.

SZABVÁNYOSSÁG ELLENŐRZÉSE

A hibák értelmezése

Hiba sorszáma

Hiba leírása angolul

49. **Error** An `img` element must have an `alt` attribute, except under certain conditions. For details, consult [guidance on providing text alternatives for images](#).
From line 72, column 1; to line 72, column 61
`ay:none"></div`

A hiba helye a forráskódban a sor (line) és oszlop (column) megadásával

A kódrészlet, amelyben a hiba van

VÉGE