

HTML5 HTML5

CSS3



GYAKORLÓ FELADAT A HTML5 TÉMAKÖRHÖZ

A legjobb <internetes/webes> témájú filmek című mintaoldal elkészítése

Web-fejlesztés kurzus, ELTE Informatikai Kar


Botscript



HTML5 HTML5

CSS3

CSS3
JavaScript

JavaScript

Botscript

MIT FOGUNK CSINÁLNI?

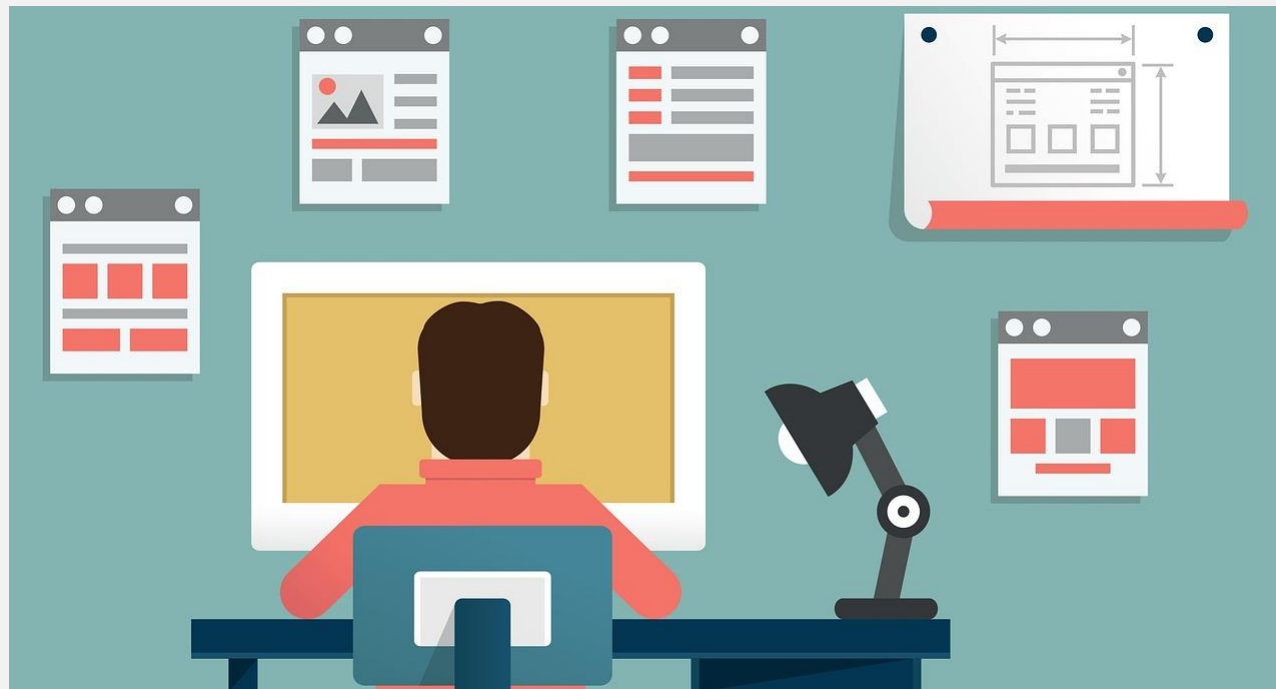
Honnan, hova jutunk el?

JAVASCRIPT

MIT FOGUNK CSINÁLNI?

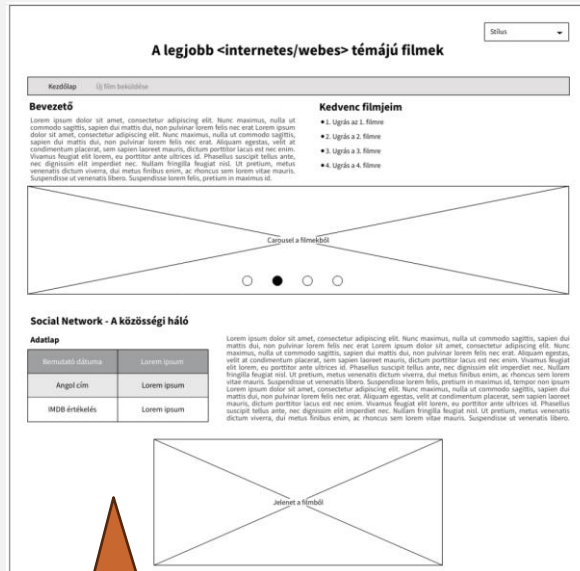
Az alapoktól felépítünk egy két oldalból álló, multimédiás elemeket és űrlapot tartalmazó weblapot, a HTML5 szabvány szerint.

Ennek során megismerjük a legfontosabb HTML5 tageket és azon praktikákat, amellyel a kódszerkesztés hatékonyabbá tehető.

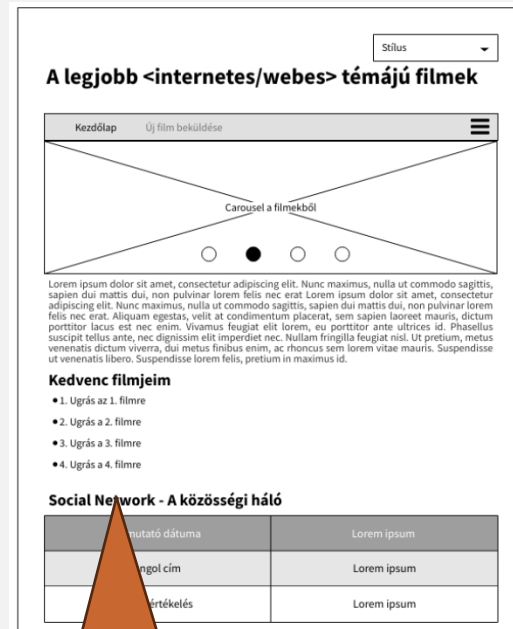


MIT FOGUNK CSINÁLNÍ?

A LEGJOBB <INTERNETES/WEBES> TÉMÁJÚ FILMEK



Drótvázterv
(Desktop)



Drótvázterv
(Mobil)

A legjobb <internetes/webes> témájú filmek - Kezdőlap

- Főmenü: Kezdőlap, Új film beklüldése

Bevezető

Krasznahorkai László a következőt írta: Az internet (...) az eddigi legbiztosabb út az örökkévalóságba. Hogy ez igaz-e, mindenki döntse el maga, miután megnézte a listában szereplő filmeket.

Figyelem! A lista nem átfogó, erősen szubjektív.

Kedvenc filmjeim

- [Social Network - A közösségi háló](#)
- [Anő](#)
- [Te meg én és minden ismerősünk](#)
- [Deep Web](#)

Social Network - A közösségi háló

Adatlap

A film adatai	
Bemutató	2010. október 1.
Angol cím	The Social Network
IMDb értékelés	7.7/10

Mockup



A drótváztervek, tartalom és mockup alapján,
készítjük el a szabványos weboldalt

MIT FOGUNK CSINÁLNI?

Honlap struktúra kialakítása (HTML5)

- Címsorok, szövegek
- Listák
- Linkek
- Előre formázott szövegek
- Illusztrációk, képek
- Beágyazott elemek
 - Audió állomány, videó állomány, beágyazott keret
- Csoportosító elemek
- Szövegrészek szerepének jelölése
- Táblázatok
- Oldalszerkezet elemek
- Űrlap készítése
- CSS alkalmazása, módosítása

Reszponzív honlap fejlesztése (Bootstrap 5.3 keretrendszerrel)

- Külön megjelenés okostelefonra, közepes, és extra nagy kijelzőre

Extra funkciók hozzáadása

- Bootstrap komponensek (pl. Carousel, rezponzív menü)
- Stílusváltás – Nagybetűs nézet (Javascript)

Megjelenés testreszabása (CSS3)

- Bootstrap stílusok felülbírálása
- Nyomtatási nézet testreszabása
- **Reszponzív weblap fejlesztése keretrendszer nélkül (CSS3)**
 - Esettanulmány. CSS3 flexbox használata.

HTML5 HTML5

CSS3

CSS3
JavaScript

JavaScript

Botscript

HTML EMLÉKEZTETŐ

Tagek (címkék), paraméterek

JAVASCRIPT

A HTML NYELV

- **HyperText Markup Language** - Hiperszöveg jelölőnyelv
- Szöveges állomány `.html` kiterjesztéssel elmentve.
- Tageket (címkéket) és objektumhivatkozásokat tartalmaz:
- Tag (címké): pl. `<p>bekezdés</p>`
 - Objektumhivatkozás: pl. `"../kepek/logo.gif"`

TAGEK (CÍMKÉK)

- Tag-nek (címkének) a `<` és `>` jelek között található jelölőelemet nevezzük.
- A címkék többségének van záró párja (páros elemek). Ez ugyanaz, mint a nyitó címke, csak egy `/` jel vezeti be.
 - pl. `<p>`Ez egy bekezdés`</p>`
- Nem minden címkének van záró párja a szabvány szerint. Ezek a páratlan (vagy üres) címkék.
 - 😊 Helyes: `
`
 - 🚫 Helytelen: `
</br>`



TAGEK (CÍMKÉK)

- A tag-eket egymásba is ágyazhatjuk, de ekkor fontos, hogy mindig azt a tag-et zárjuk le először, amelyet utoljára nyitottunk meg.
- Hasonlóan ahhoz, ahogy mondjuk a matematikai kifejezéseket zárójelezzük
($a + [b * c]$).



Helyes:

`<i>szöveg</i>`



Helytelen:

`<i>szöveg</i>`

TAG PARAMÉTEREKKEL

- A címkéknek lehetnek paramétereik is.
- A paraméternek (attribútumnak) van neve és értéke:
`<tag paraméter1="érték1" paraméter2="érték2">`
 - Pl. ``
- Vannak logikai attribútumok is, ezeknél elég lehet a paraméter nevének szerepeltetése (ez a szabványtól függ!)
 - `<p hidden>`Ez egy rejtett bekezdés`</p>`
- Vannak olyan paraméterek, amelyeket minden címkénél lehet használni, ezeket globális paramétereknek nevezzük.
 - Ilyen például az `id` attribútum, amivel egyedi nevet adhatunk egy elemnek.



GLOBALIS PARAMÉTEREK (GLOBAL ATTRIBUTES)

- Néhány gyakran használt globális paraméter:
 - **title**: az elemhez tartozó feliratot adhatjuk meg, amely legtöbbször akkor jelenik meg, ha az elem fölé visszük az egeret, de speciális jelentése is lehet.
 - **accesskey**: gyorsbillentyű megadására szolgál. Bármilyen (1 karakter hosszú) karaktert megadhatunk. A kis- és nagybetűk különböznek.
 - **id**: az elemet egyedi azonosítóval láthatjuk el.
 - **class**: az elemet stílus osztályba sorolhatjuk.
 - **lang**: segítségével megadhatjuk az adott elem nyelvét. (pl. hu, en, en-us, stb.)



MEGJEGYZÉS

- A HTML kódban az alábbi módon lehet megjegyzés írni:

<!-- Egy soros megjegyzés -->

vagy

<!-- Több
soros
megjegyzés
-->



HTML5 PÉLDA



```
<!DOCTYPE html>
<html lang="hu">
<head>
    <meta charset="utf-8">
    <title>Első weblapom</title>
</head>
<body>
    <p></p>
    <p>Ez az első próbálkozásom.
    <b>Remélem sikerül!</b></p>
</body>
</html>
```

HTML5 HTML5

CSS3

CSS3
JavaScript

JavaScript

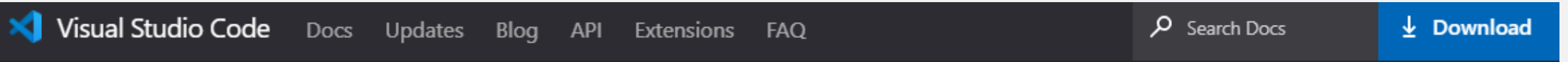
Botscript

VISUAL STUDIO CODE (VS CODE)

A használt fejlesztői környezetünk

TELEPÍTÉSE

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



↓ Windows

Windows 7, 8, 10



↓ .deb

Debian, Ubuntu

↓ .rpm

Red Hat, Fedora, SUSE



↓ Mac

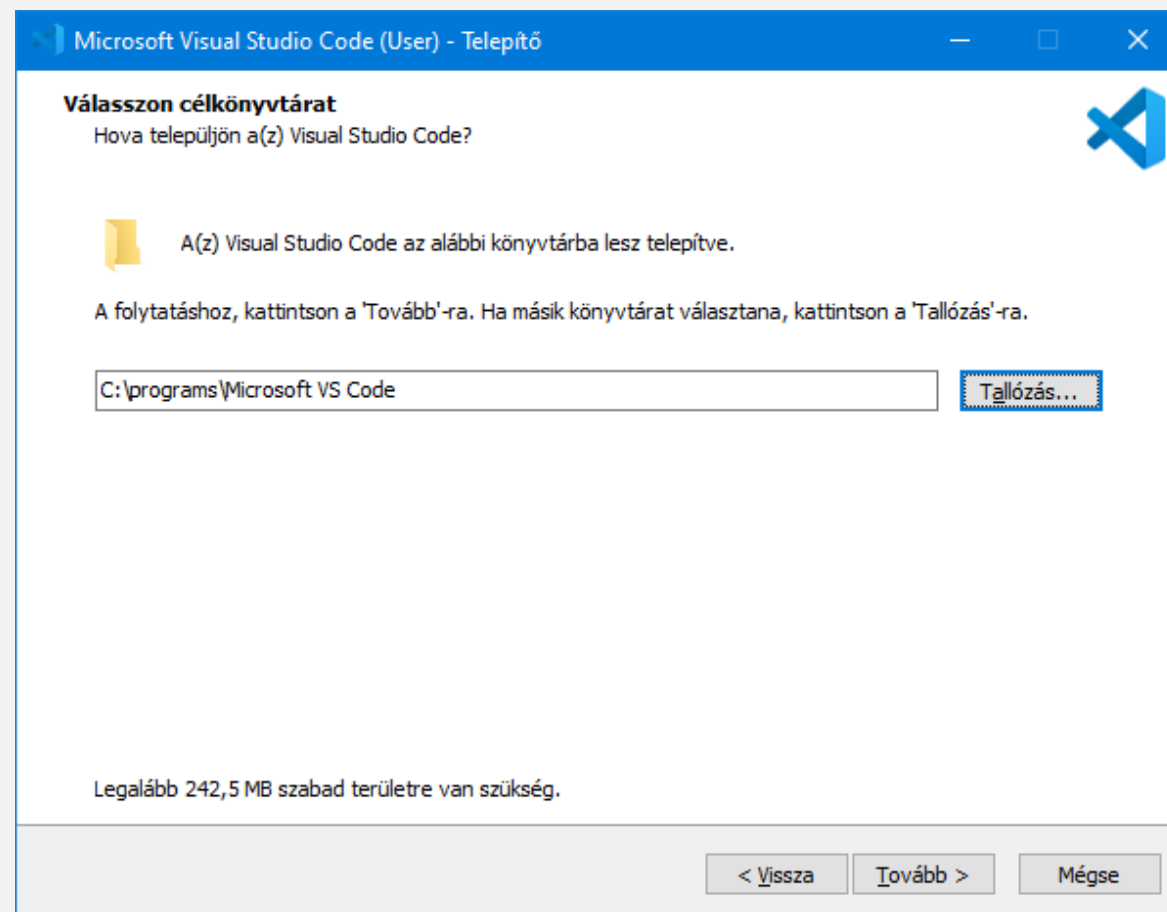
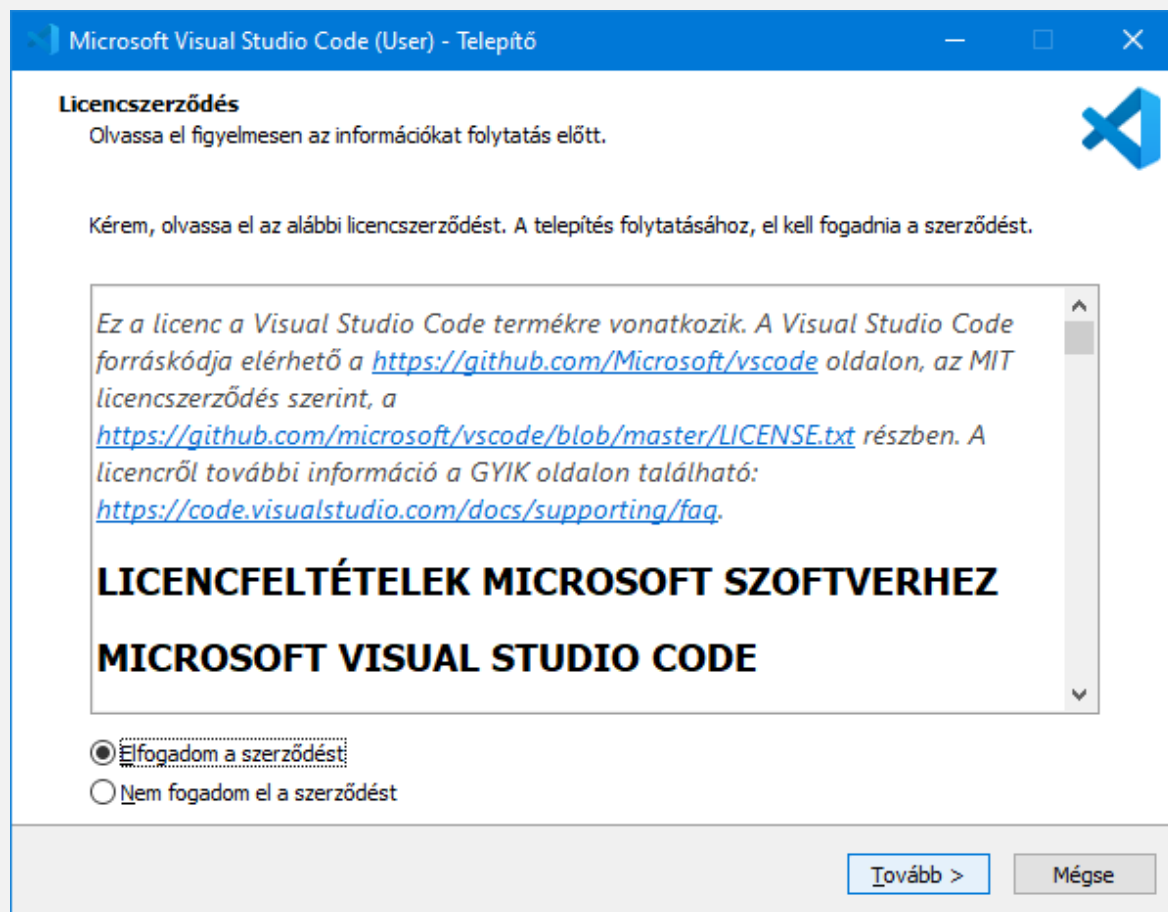
macOS 10.10+

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

A TELEPÍTÉS LÉPÉSEI WINDOWS PLATFORMON




A TELEPÍTÉS LÉPÉSEI WINDOWS PLATFORMON

Microsoft Visual Studio Code (User) - Telepítő

Start Menü könyvtára

Hova helyezze a Telepítő a program parancsikonjait?

 A Telepítő a program parancsikonjait a Start menü következő mappájában fogja létrehozni.

A folytatáshoz kattintson a 'Tovább'-ra. Ha másik mappát választana, kattintson a 'Tallózás'-ra.

☐ Ne hozzon létre mappát a Start menüben

Microsoft Visual Studio Code (User) - Telepítő

További feladatok

Mely kiegészítő feladatok kerüljenek végrehajtásra?

Jelölje ki, mely kiegészítő feladatokat hajtsa végre a Telepítő a(z) Visual Studio Code telepítése során, majd kattintson a 'Tovább'-ra.

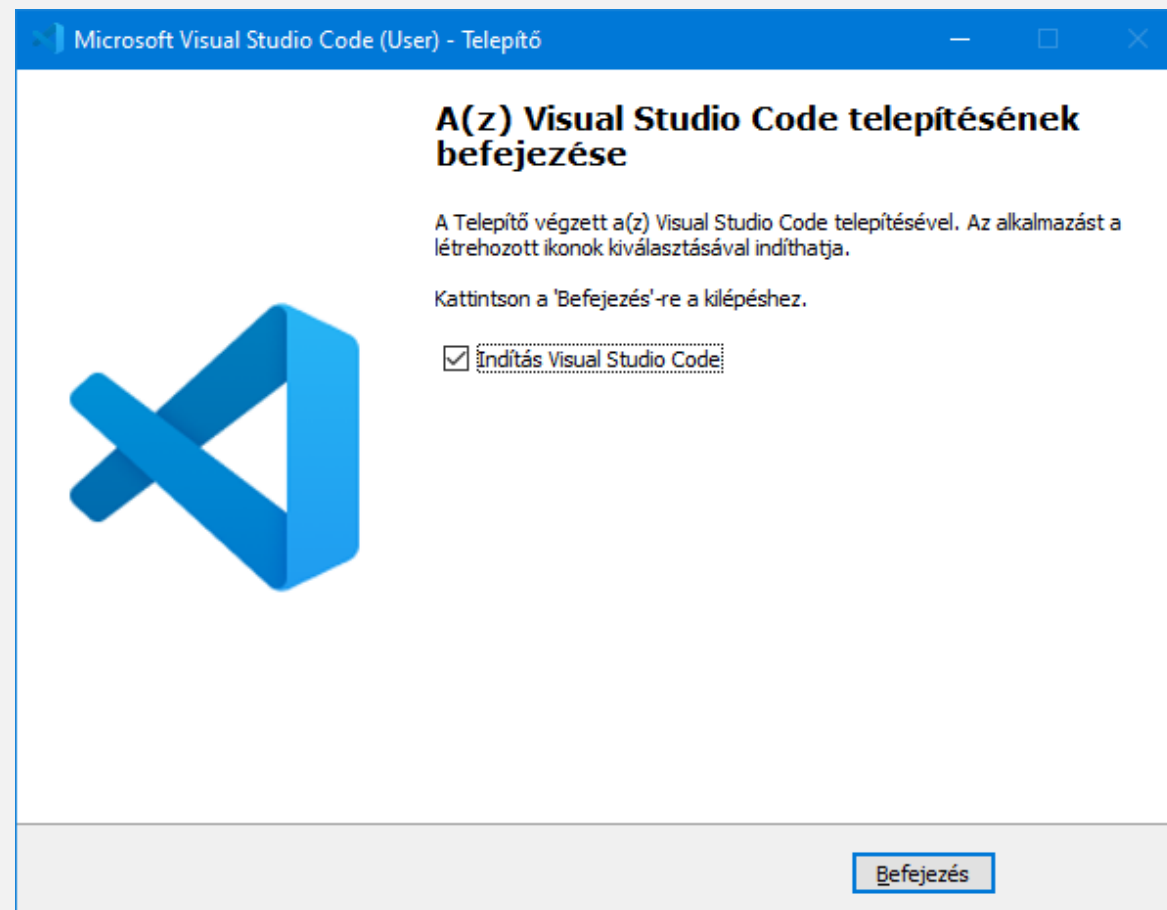
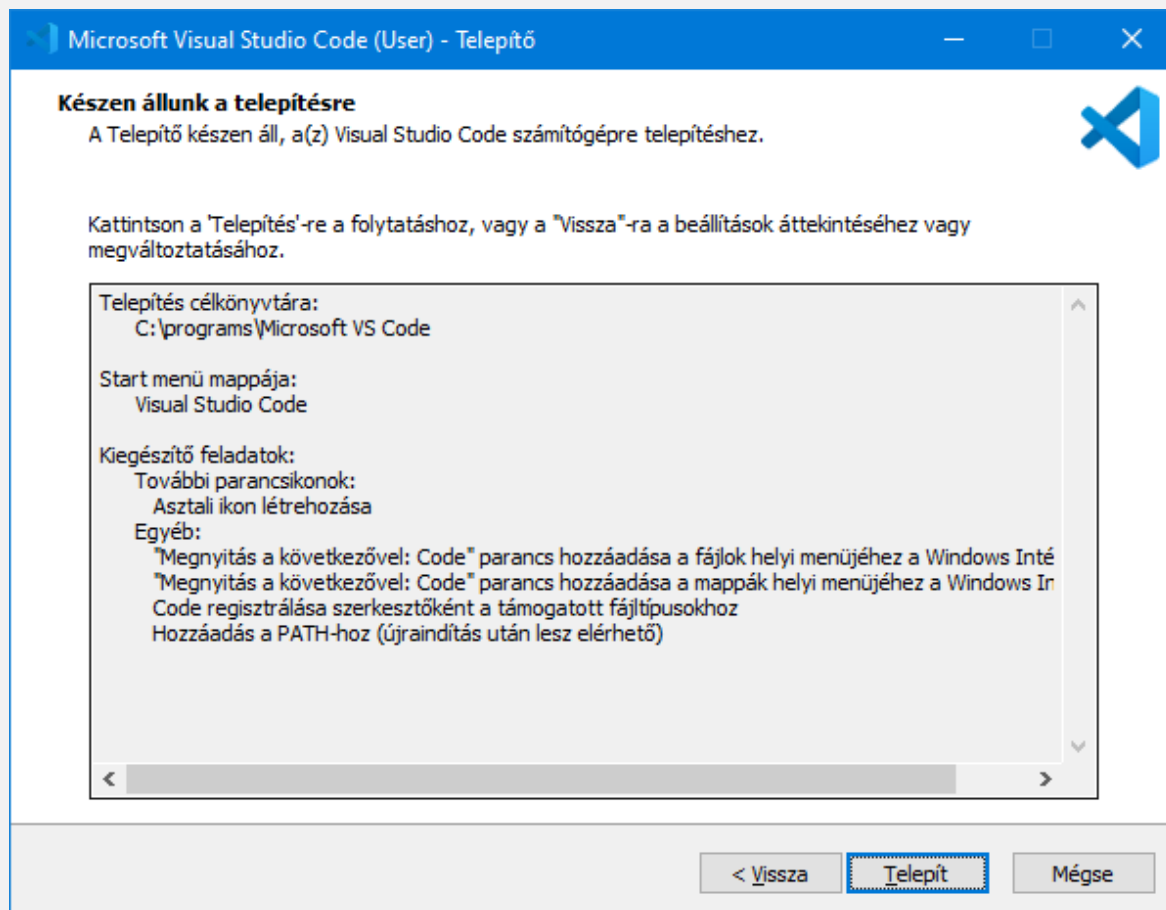
További parancsikonok:

- ☒ Asztali ikon létrehozása

Egyéb:

- ☒ "Megnyitás a következővel: Code" parancs hozzáadása a fájlok helyi menüjéhez a Windows Intézőben
- ☒ "Megnyitás a következővel: Code" parancs hozzáadása a mappák helyi menüjéhez a Windows Intézőben
- ☒ Code regisztrálása szerkesztőként a támogatott fájltypusokhoz
- ☒ Hozzáadás a PATH-hoz (újraindítás után lesz elérhető)

A TELEPÍTÉS LÉPÉSEI WINDOWS PLATFORMON



A PROGRAM FELÜLETE

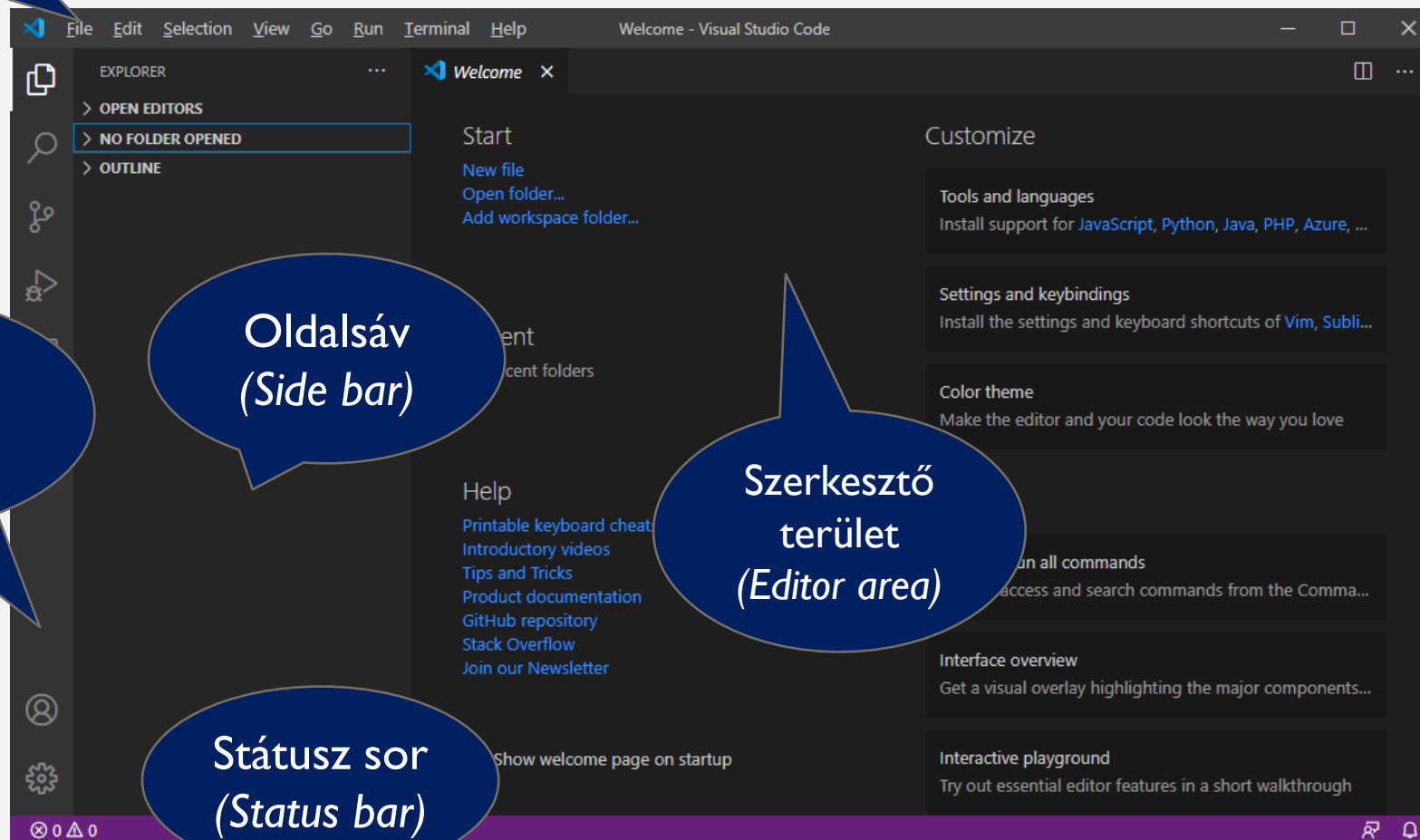
Menüsor
(Menu bar)

Tevékenység
panel
(Activity bar)

Oldalsáv
(Side bar)

Szerkesztő
terület
(Editor area)

Státusz sor
(Status bar)



View / Appearance menüpontban kikapcsolhatóak, megjeleníthetőek az elemek

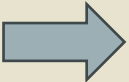
EMMET RÖVIDÍTÉSEK HASZNÁLATA

- Az **Emmet** bővítmény nevét egy evolúciós biológiából származó fogalom, a "**Emmet hypothesis**" inspirálta, amely a komplex struktúrák gyors növekedésével kapcsolatos.
- Az Emmet a **Zen Coding** nevű projekt továbbfejlesztése.
- A névváltás során olyan nevet kerestek, amely kifejezi az eszköz célját: a HTML és CSS struktúrák gyors és hatékony generálását, hasonlóan ahhoz, ahogyan egy organizmus gyorsan fejlődik és terjeszkedik.

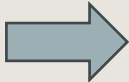


Forrás: ChatGPT

EMMET PÉLDÁK

`ul>li*5` 

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

`ul>li#elem$*5` 

```
<ul>
  <li id="elem1"></li>
  <li id="elem2"></li>
  <li id="elem3"></li>
  <li id="elem4"></li>
  <li id="elem5"></li>
</ul>
```

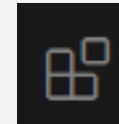
HASZNOS KITERJESZTÉSEK

- **Live server:** a forráskód mentése után mindjárt láthatjuk az eredményt a böngészőprogramban.



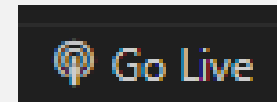
- Telepítés:

- Kattintsunk az Extensions (Bővítmények) ikonra!
- Keressünk rá a **Live Server** bővítményre, és telepítsük!



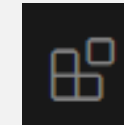
- Használat

- A VSC ablak jobb alsó sarkában megjelenik a **Go live** felirat.
- Rákattintva megjelenik a böngészőprogramban az eredmény.
 - Akkor működik, ha a programban egy mappát (vagy munkaterületet) nyitottunk meg.



HASZNOS KITERJESZTÉSEK

- **htmltagwrap**: a forráskódban kijelölt szöveget körbe tudjuk zárni HTML címkével.
- Telepítés:
 - Kattintsunk az Extensions (Bővítmények) ikonra!
 - Keressünk rá a **htmltagwrap** bővítményre, és telepítsük!
- Használat
 - Jelöljük ki a szöveget!
 - Nyomjuk meg az "Alt + W" („Mac: Option + W”) gyorsbillentyűt!
 - Írjuk be, hogy milyen címkével szeretnénk körbe zárni a szöveget!



HTML5 HTML5

CSS3

CSS3
JavaScript

JavaScript

Botscript

ALAPCSOMAG LETÖLTÉSE

Ismerkedés a tartalommal

JAVASCRIPT

ALAPCSOMAG

- Töltsük le az alábbi alapcsomagot (Canvas LMS-ben is elérhető) és tömörítsük ki!

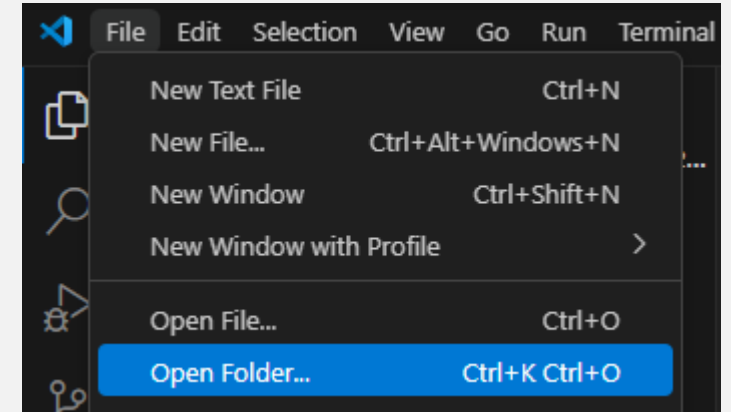
<https://bit.ly/4hwv8Tk>



Munkánk során több alapcsomagot is használunk majd, hogy a gyakorlaton az ismétlődő feladatok megoldása helyett az új ismeretekre koncentrálhassunk! (pl. nem fogjuk ugyanazt a táblázatot elkészíteni négyszer)

NYISSUK MEG A MAPPÁT A VSC-BEN

1. **File / Open folder** menüpont kiválasztása
2. Mappa kiválasztása
3. Az Explorer (intéző) panelen megjelenik a mappa tartalma



Tartalom

design mappa:	Drótváztervek, tartalom, mockup
media mappa:	Beillesztendő médiaelemek (képek, hang, videó)
css mappa:	Stíluslap állomány
js mappa:	Javascript állomány

ISMERKEDÉS A TARTALOMMAL

- A tartalom dokumentum állományként is elérhető.
- Nyissuk meg a szövegszerkesztő alkalmazásban!
- A címsorszintek jelzik a dokumentum szerkezetét.
- A képek be vannak ágyazva.
- A speciális elemek beágyazására vonatkozó információ is el van helyezve. pl.
{hang beágyazása - socialnetwork.mp3}

Social Network - A közösségi háló

Adatlap

A film adatai	
Bemutató	2010. október 1.
Angol cím	The Social Network
IMDb értékelés	7.7/10

Mark Zuckerberg jobban ért a számítógépekhez, mint a csajozáshoz. A Harvard egyetem diákja ideje nagy részét a gépe előtt tölti. 2003-ban új ötlettel áll elő: egy olyan **oldalt indít** a világhálón, ahol a haverok **közösségi életet** élhetnek. A kollégiumi szobából indult Facebook valóságos **kommunikációs forradalmat** hozott, és alapjaiban változtatta meg a világot, az emberi kapcsolatokat - és megváltoztatta az alkotóját is. Hat esztendővel és ötszázmillió ismerőssel később Zuckerberg a világ legfiatalabb milliárdosa. Egykor békés, nyugodt életéről álmódott, a siker azonban felkavarta mind a szakmai, mind a magánéletét.

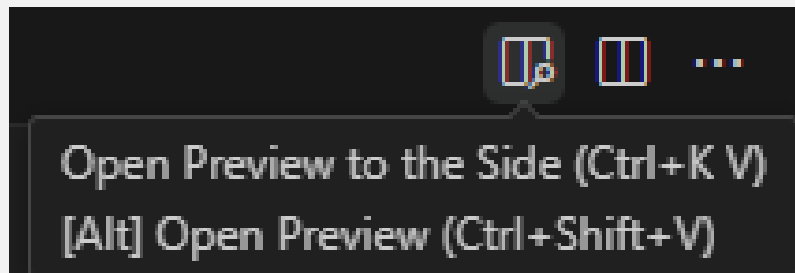
A Social Network - A közösségi háló a korszellemet grabancon ragadó, aktuális, de időtálló munka, ami nem csak, hogy sokkal jobb móka, mint két óra facebookozás, de biztos, hogy évek-évtizedek távlatából is élvezhető lesz.



Jelenet a Social Network - A közösségi háló című filmből

ISMERKEDÉS A TARTALOMMAL

1. Nyissuk meg a weblap tartalmát leíró Markdown állományt a VSC alkalmazásban!
2. Kattintsunk a megfelelő ikonra az előnézethez!



```
design > Tartalom - Legjobb filmek.md > # A legjobb <internetes/webes> témájú filmek - Kezdőlap > ## Social Network - A közösségi háló
1 # A legjobb <internetes/webes> témájú filmek - Kezdőlap
21 ## Social Network - A közösségi háló
23 ### Adatlap
Egykor békés, nyugodt életéről álmódott, a siker azonban felkavarta mind a
szakmai, mind a magánéletét.
32
33 _A Social Network - A közösségi háló a korszellemet grabancon ragadó,
aktuális, de időtálló munka, ami nem csak, hogy sokkal jobb móka, mint két óra
facebookozás, de biztos, hogy évek-évtizedek távlatából is élvezhető lesz._
34
35 ![Jelenet a Social Network - A közösségi háló című filmből](../media/
TheSocialNetwork.jpg)
36
37 Jelenet a _Social Network - A közösségi háló_ című filmből
38
39 ### Forráskód a filmben
40
41 A filmben nagy figyelmet fordítottak arra is, hogy a megjelenő forráskódok a
témához kapcsolódjanak. Íme egy példa:
42 ```py
43 #!/usr/bin/perl
44 $wget = "wget -qO- http://leverett.harvard.edu/facebook/compactshow.php
--post-data='action=Search'";
45 print "$wget\n";
46 $page = ` $wget `;
47 while ($page =~ m/compactshow\.php\?student_id=([0-9]+)/g) {
48     $id=$1
49     $wget = "wget -qO- http://leverett.harvard.edu/facebook/compactshow.
php?student_id=$id";
50     print "$wget\n";
51 }
52 ```
53
54 A filmben szereplő forráskód részlete
55
56 ### Filmzene
57
58 A filmzenéből _Trent Reznor and Atticus Ross: A Familiar Taste_ című száma a
kedvencem.
59
60 {hang beágyazása - socialnetwork.mp3}
61
62 ## A nő
63
64 ### Adatlap
65
66 | **A film adatai** | |
67 | --- | --- |
68 | **Bemutató** | 2014. január 10. |
69 | **Angol cím** | _Her_ |
70 |
```

Hat esztendővel és ötszázmillió ismerőssel később Zuckerberg a világ legfiatalabb milliárdosa. Egykor békés, nyugodt életéről álmódott, a siker azonban felkavarta mind a szakmai, mind a magánéletét.

A Social Network - A közösségi háló a korszellemet grabancon ragadó, aktuális, de időtálló munka, ami nem csak, hogy sokkal jobb móka, mint két óra facebookozás, de biztos, hogy évek-évtizedek távlatából is élvezhető lesz.



Jelenet a Social Network - A közösségi háló című filmből

Forráskód a filmben

A filmben nagy figyelmet fordítottak arra is, hogy a megjelenő forráskódok a témához kapcsolódjanak. Íme egy példa:

```
#!/usr/bin/perl
$wget = "wget -qO- http://leverett.harvard.edu/facebook/compactshow.php
--post-data='action=Search'";
print "$wget\n";
$page = ` $wget `;
while ($page =~ m/compactshow\.php\?student_id=([0-9]+)/g) {
    $id=$1
    $wget = "wget -qO- http://leverett.harvard.edu/facebook/compactshow.php?
student_id=$id";
    print "$wget\n";
}
```

MARKDOWN → HTML KONVERZIÓ?

„Miért nem konvertáljuk a Markdown nyelven leírt tartalmat HTML nyelvű tartalommmá, és megyünk haza?”

- Ingyenes konverterek:
 - <https://markdowntohtml.com/>
 - <https://codebeautify.org/markdown-to-html>
 - <https://marketplace.visualstudio.com/items?itemName=manuth.markdown-converter>



MARKDOWN → HTML KONVERZIÓ?

- **Azért NEM, mert**
 - A HTML sokkal több lehetőséget biztosít a tartalom szemantikai leírására (szerkezet, jelentés), mint a Markdown.
 - A konverzióval nem készülünk fel a valódi webfejlesztési feladatokra.
- **Tanuljuk meg először szakszerűen használni a címkéket (tageket) és paramétereket! Ügyeljünk az akadálymentességre!**



Később (esetleg) használhatunk konvertert, de biztosan át kell alakítanunk, ki kell egészítenünk a kapott kódot!

MI (AI) HASZNÁLATA

Ugyanez igaz az MI használatára!

Kezdetben nem használjuk, csak később, amikor már kritikusan tudjuk szemlélni a kapott kódot.



SZEMANTIKA FONTOSSÁGA

- A feladat megoldása során mindig a **szemantikailag megfelelő tagek (*tags*) és paraméterek (*attributes*)** használatára törekszünk!
- **Előnyök**
 - Algoritmikusan meghatározható, hogy az adott elem milyen szerepet tölt be az oldalon
 - Gépi feldolgozás hatékonyabbá válik (pl. képaláírás alapján hatékonyabb lehet a keresés, adott dátumra/időpontra könnyebben rá lehet keresni, stb.)
 - Egyenlő esélyű hozzáférés (akadálymentesség) biztosítása (navigáció, médiaelemek leírása, stb.)
 - Keresőoptimalizálás (SEO) vonatkozásában is nagyon előnyös.



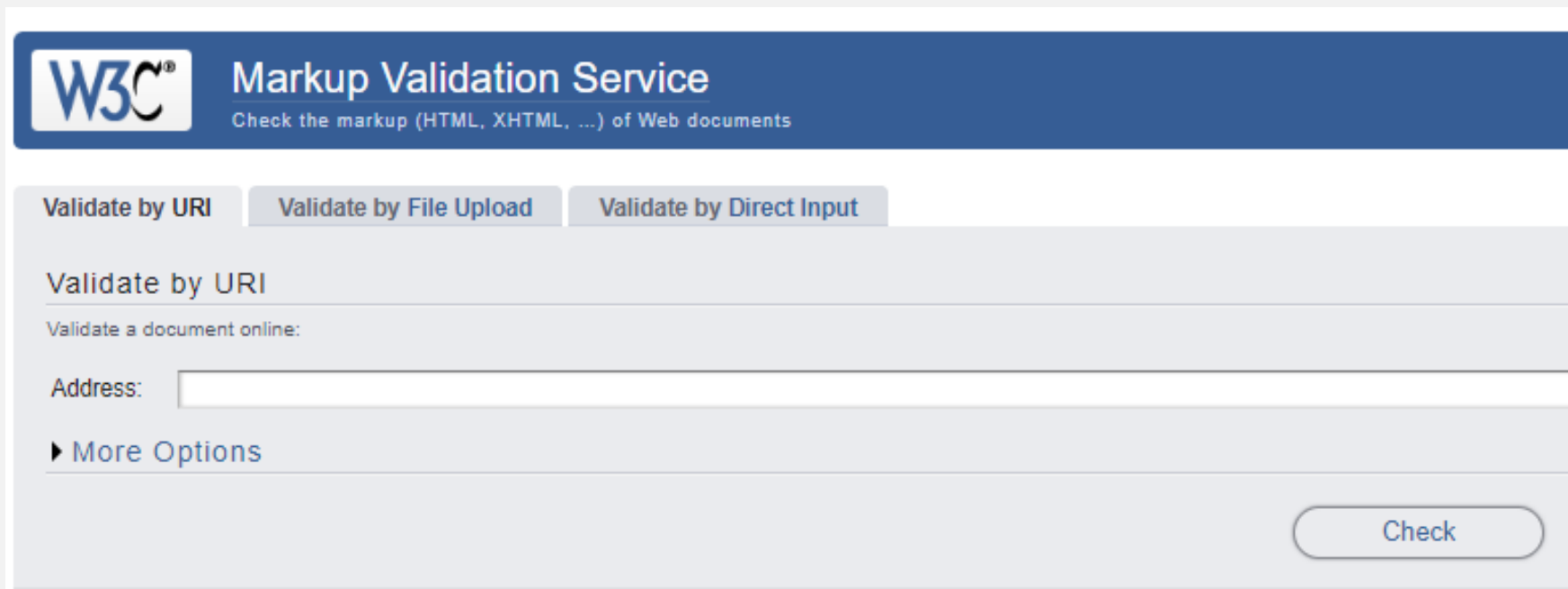
SZABVÁNYOSSÁG ELLENŐRZÉSÉNEK FONTOSSÁGA (VALIDÁCIÓ)

- Segít elkerülni azokat a hibákat és hibákat a kódban, amelyek befolyásolhatják weboldalai működését, megjelenését és hozzáférhetőségét.
- Szabványos kód esetén gyorsabban töltődik be a weboldal.
- Javul a kód minősége, olvashatósága, karbantarthatósága.
- A leendő munkaadónak (is) fontos jelzés, hogy mennyire igényes valaki a munkájával kapcsolatban.



VALIDÁLÓ ESZKÖZ

- A <https://validator.w3.org/> oldal segítségével ellenőrizhetjük a szabványosságot. Ezt nevezzük validálásnak.
- A weboldalak elenyésző kisebbsége szabványos. Sok népszerű CMS rendszer sem generál alaptól szabványos kódot.



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 right of the form, there is a "Check" button.

HTML5 HTML5

CSS3

CSS3
JavaScript

JavaScript

Botscript

KEZDŐLAP KIALAKÍTÁSA

JAVASCRIPT

KEZDŐLAP KIALAKÍTÁSA



- Nyissuk meg a dokumentum állományt (.doc), ebből fogjuk kimásolni a tartalmakat!
- A VSC-ben hozzuk létre a kezdőlapot a mappastruktúránk gyökerében. A neve `index.html` legyen!
- **File / New File** menüpont



Social Network - A közösségi háló

Adatlap

A film adatai	
Bemutató	2010. október 1.
Angol cím	<i>The Social Network</i>
IMDb értékelés	7.7/10

Mark Zuckerberg jobban ért a számítógépekhez, mint a csajozáshoz. A Harvard egyetem diákja ideje nagy részét a gépe előtt tölti. 2003-ban új ötlettel áll elő: egy olyan **oldalt indít** a világhálón, ahol a haverok **közösségi életet** élhetnek. A kollégiumi szobából indult Facebook valóságos **kommunikációs forradalmat** hozott, és alapjaiban változtatta meg a világot, az emberi kapcsolatokat - és megváltoztatta az alkotóját is. Hat esztendővel és ötszázmillió ismerőssel később Zuckerberg a világ legfiatalabb milliárdosa. Egykor békés, nyugodt életéről álmódott, a siker azonban felkavarta mind a szakmai, mind a magánéletét.

A Social Network - A közösségi háló a korszellemet grabancon ragadó, aktuális, de időtálló munka, ami nem csak, hogy sokkal jobb móka, mint két óra facebookozás, de biztos, hogy évek-évtizedek távlatából is élvezhető lesz.



Jelenet a *Social Network - A közösségi háló* című filmből

HTML5 ALAPSTRUKTÚRA LÉTREHOZÁSA (NYELV ÉS KARAKTERKÓDOLÁS MEGADÁSSAL)

A Visual Studio Code alkalmazásban az alapstruktúra a `html:5` vagy az `!` emmet paranccsal illeszthető be (vagyis nem kell fejből tudni).

```
<!DOCTYPE html>
<html lang="en">

<head>
  <meta charset="UTF-8">
  <meta name="viewport"
        content="width=device-width,
                initial-scale=1.0">
  <title>Document</title>
</head>
<body>

</body>
</html>
```

Dokumentumtípus
HTML gyökér elem nyelv megadással
Fej
 Karakterkódolás
 A viewport beállítása

Oldalcím

Fej vége
Törzs

Törzs vége
HTML vége

Feladat:

Magyar nyelvű oldalt készítünk, ezért a nyelvkód legyen: `hu`

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

CHARSET (KARAKTERKÓDOLÁS) BEÁLLÍTÁSA

Nagyon fontos, hogy a kódban szereplő karakterkódolás (pl. utf-8) és a szerkesztőben megadott karakterkódolás ugyanaz legyen.



Visual Studio Code esetén

Az ablak státusz sorában látható a karakterkódolás, amely HTML állomány esetén alapértelmezetten UTF-8.

Ln 11, Col 8 Spaces: 4 UTF-8

A karakterkódolásra kattintva újra megnyithatjuk az állományt, illetve elmenthetjük más karakterkódolással.

Select Action

Reopen with Encoding

Save with Encoding

Select File Encoding to Save with

UTF-8 utf8

UTF-8 with BOM utf8bom

UTF-16 LE utf16le

UTF-16 BE utf16be

Western (Windows 1252) windows1252

Western (ISO 8859-1) iso88591

Western (ISO 8859-3) iso88593

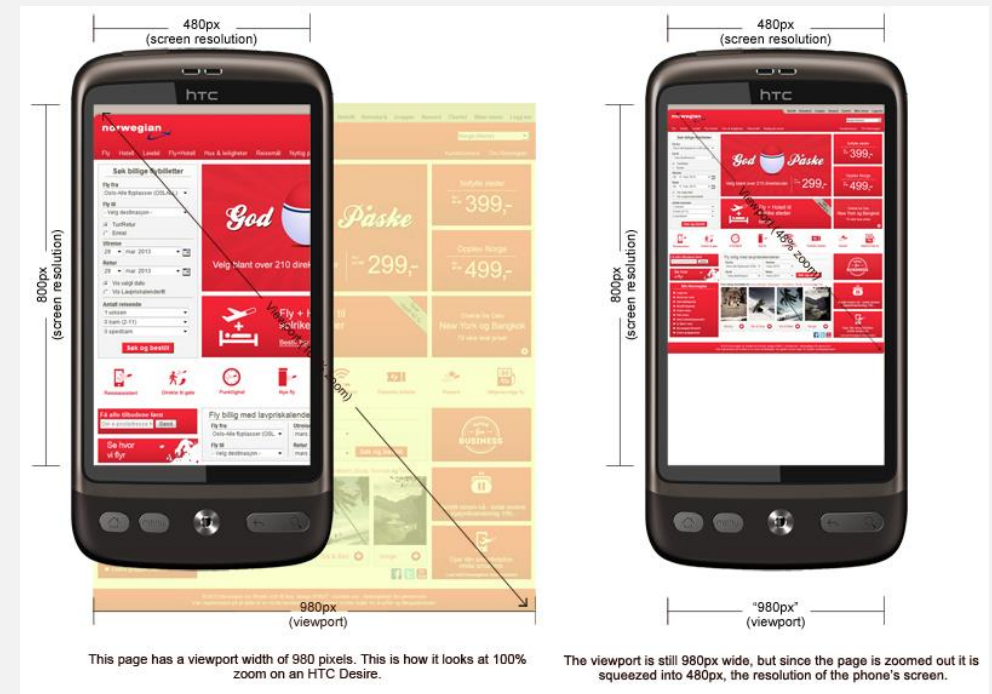
Western (ISO 8859-15) iso885915

VIEWPORT FOGALMA

- A mobil eszközök böngészőprogramjai az oldalakat virtuális ablakokban jelenítik meg (ezt nevezzük viewportnak)
- Szélesebb is lehet, mint a képernyő, és ezen területen a felhasználók több irányban barangolhatnak, illetve egyes területekre ráközelíthetnek, illetve eltávolodhatnak.
- A viewport tulajdonságait `<meta>` taggel adjuk meg.

pl. `<meta name="viewport" content="width=device-width, initial-scale=1.0">`

Később részletesebben megnézzük a szerepét!



KEZDŐLAP TARTALMA

- Jelöljük ki a dokumentumban a kezdőlap tartalmát! (Az új film beküldése oldalt már ne!)
- Másoljuk át a tartalmat a VSC-be, a `<body>` és `</body>` elemek közé!
- Mentsük el az állományt és nézzük meg az eredményt a böngészőben!

A dark rectangular button with a white '@' icon and the text 'Go Live' in a light blue font.

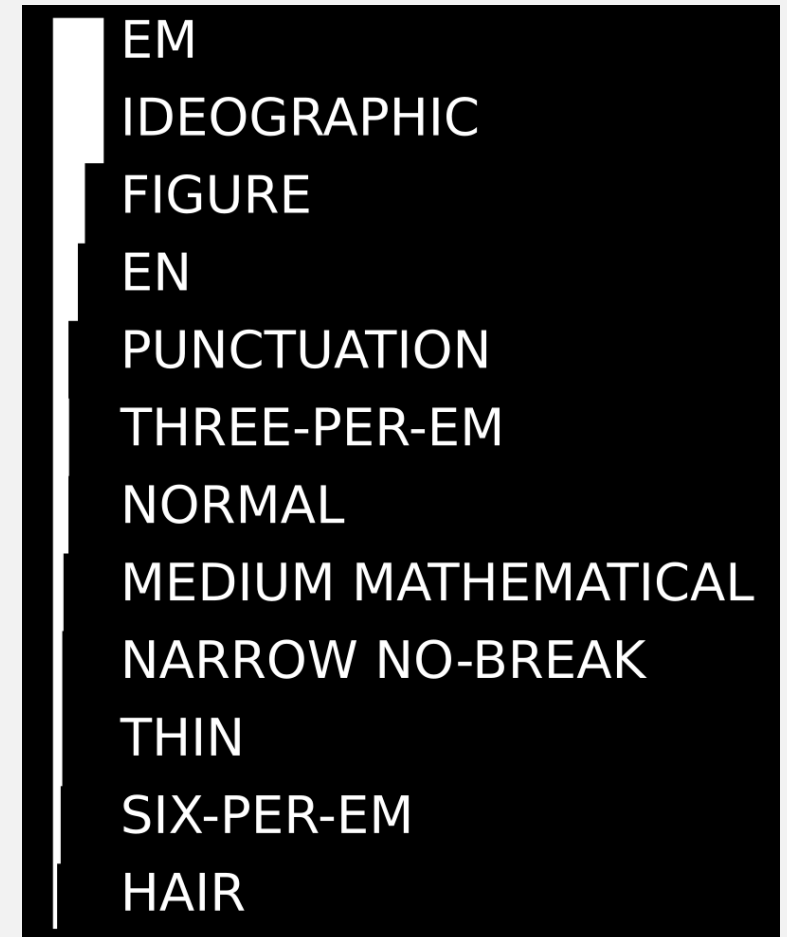
- Mit tapasztalunk? Miben más a megjelenés, mint például a Markdown előnézetben volt?

TANULSÁG

- A HTML kód tördelése (szóközök, sortörés egyéb whitespace karakterek) nem befolyásolják a megjelenést.
 - A Markdown nyelvben nem így volt.
- Ha ez előre formázottság (szóközök, sortörések) fontos, akkor majd egy speciális címkét kell használnunk. Ezt később kipróbáljuk a forráskód elhelyezése során.

A whitespace karakterek nem láthatóak a szövegben, viszont valamilyen egyedi funkcióval bírnak.

https://en.wikipedia.org/wiki/Whitespace_character



Néhány (whitespace) Unicode karakter szélessége

FELADAT

- A `<title></title>` tagek között adjuk meg az oldal címét.
- Olyan szöveget adjunk meg, amely jól azonosítja a teljes websiteot és az egyes lapjait is!
- Érdemes a szövegnek kulcsszóval kezdődnie, vagy végződnie pl.
„A legjobb `<internetes/webes>` témájú filmek – Kezdőlap”
- Nézzük meg az eredményt a böngészőben! Hol jelenik meg a szöveg?
- Próbáljuk könyvjelzőbe tenni az oldalt! Mit tapasztalunk?

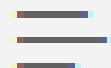
FELADAT



Minden egyes sort rakjunk külön bekezdésbe, vagyis `<p>` és `</p>` közé! Használjunk ehhez **reguláris kifejezéseket!**

1. Jelöljük ki a `<body>` és `</body>` elemek között lévő szöveget!
2. Keresés és csere funkcióval a sorkezdeteket cseréljük le `<p>`-re, a sorvégejeleket pedig `</p>`-re.
 1. Válasszuk ki az **Edit / Replace** menüpontot (vagy CTRL+H)
 2. Kattintsunk a **Find in Selection** (Keresés a kijelölésben) ikonra!
 3. Állítsuk be, hogy reguláris kifejezést fogunk használni!
4. Egyik lehetséges megoldás:
Erre keresünk: `^(.+)$`
5. Erre cserélünk: `<p>$1</p>`
6. Végezzük el a cserét!
7. Nézzük meg az eredményt!

Mi lenne a jó
reguláris kifejezés
előzetes
tanulmányaink
alapján?



ALTERNATÍV MEGOLDÁSOK



- Jelöljük ki a teljes szöveget a `<body>` és `</body>` között!
- Válasszuk ki a *Selections / Add cursor to line ends* menüpontot! (a sorok végén villog a kurzor)
- Nyomjuk meg a HOME billentyűt, így a sorok elejére kerül.
- Gépeljük be a `<p>` taget. (ha a program automatikusan lezárja és elhelyezi a `</p>` -t, akkor azt töröljük ki a DELETE billentyűvel!
- Ugorjunk a sorok végére az END billentyűvel!
- Gépeljük be a `</p>` taget!
- Mentsük el az eredményt és nézzük meg a böngészőben!

Vigyázzunk! Ez a módszer akkor működik jól, ha nincs beállítva a szavak tördelése funkció (View / Word wrap). Ellenkező esetben a hosszú sorok több sorba tördelődnek, ami miatt nem kívánt eredményt kapunk.

ALTERNATÍV MEGOLDÁSOK



- Jelöljük ki a teljes szöveget a `<body>` és `</body>` között!
- Nyissuk meg a Parancs listát (Command palette) [*F/ vagy View / Command Palette*]
- Válasszuk ki a **Wrap with Abbreviation** parancsot és adjuk meg, hogy `p*`.
- Minden sor elé `<p>` kerül, a végére `</p>`.

Vigyázzunk! Ez a módszer akkor működik jól, ha nincs beállítva a szavak tördelése funkció (View / Word wrap). Ellenkező esetben a hosszú sorok több sorba tördelődnek, ami miatt nem kívánt eredményt kapunk.



1. Reguláris kifejezéssel távolítsuk el az összes whitespace karaktert a nyitó tagek után!
2. Távolítsuk el az összes üres bekezdést is! (`<p>` `</p>`)

HTML5 HTML5

CSS3

CSS3
JavaScript

JavaScript

Botscript

CÍMSOROK (HEADINGS)

JAVASCRIPT

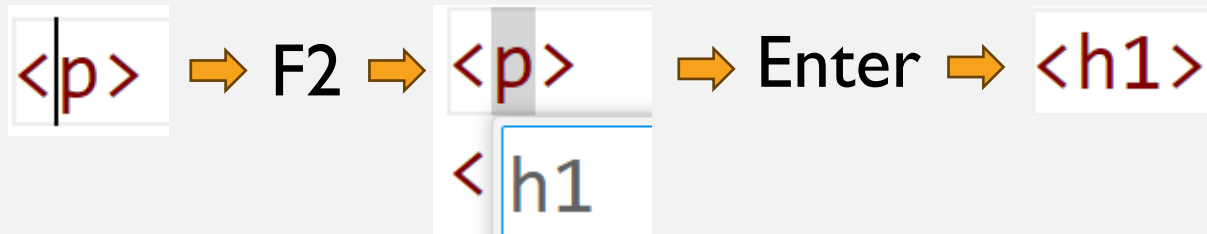
CÍMSOROK (HEADINGS) A HTML NYELVBEN

- `<h1></h1>` 1-es szintű címsor (*Level 1 heading*)
- `<h2></h2>` 2-es szintű címsor
- `<h3></h3>` 3-as szintű címsor
- `<h4></h4>` 4-es szintű címsor
- `<h5></h5>` 5-ös szintű címsor
- `<h6></h6>` 6-os szintű címsor

Az oldal tartalomjegyzékét írják le, mint egy Word dokumentumban. Fontos, hogy logikus legyen a címsorszintek kiosztása, minden oldalon legyen 1-es címsor! Aztán lehet 2-es, 3-as stb. szinteket kiosztani.



1. Alakítsuk ki a címsorszinteket a dokumentum alapján!
2. Az adott tagre kattintva, az **F2** megnyomása után lecserélhetjük az adott taget egy másikra.



- Az összes `<h1>`, `<h2>`, `<h3>` taget helyezzük el!

HTML5 HTML5

CSS3

CSS3
JavaScript

JavaScript

Botscript

SPECIÁLIS KARAKTEREK (SPECIAL CHARACTERS)

JAVASCRIPT

SPECIÁLIS KARAKTEREK HASZNÁLATA



Gyakran használt speciális karakterek

Karakter	HTML kód	unicode	Karakter	HTML kód	unicode
'	´	´	<	<	<
&	&	&	—	¯	¯
	¦	¦	μ	µ	µ
.	•	•		 	
c	¢	¢	¬	¬	¬
©	©	©	¶	¶	¶
°	°	°	%	‰	‰
€	€	€	?	±	±
÷	÷	÷	"	"	"
½	½	½	®	®	®
¼	¼	¼	»	»	»
¾	¾	¾	§	§	§
>	>	>	¨	¨	¨
!	¡	¡	×	×	×
?	¿	¿	™	™	™

Feladat:

Az oldal fő címsora a
**A legjobb <internetes/webes>
témájú filmek** szöveg nem jelenik
meg helyesen.

Hogyan tudjuk a < és > jeleket
megfelelően kiírni?

HTML5 HTML5

CSS3

CSS3
JavaScript

JavaScript

Botscript

LISTÁK (LISTS)

JAVASCRIPT

LISTÁK (LISTS) HASZNÁLATA (PÉLDA)

<code></code>	felsorolás lista
<code></code>	sorszámozott lista
<code></code>	listaelem

Muffincsoda

Hozzávalók

- 8 evőkanál liszt
- 2 dl tej
- 1 dl olaj
- 4 evőkanál cukor
- 2 tojás (egész)
- 1 csomag sütőpor
- 1/4 citrom reszelt héja
- 2 csomag vaníliás cukor

Elkészítés

1. A hozzávalókat egy tálban összekeverjük.
2. A formába helyezett muffinpapírt olajjal kikenjük.
3. Előmelegített sütőben 165°C-on 25 percig sütjük.

Feladat: A filmes oldalunkon az oldalmenü legyen sorszámozott listában, a források pedig felsoroláslistában!



```
<h1>Muffincsoda</h1>
```

```
<h2>Hozzávalók</h2>
```

```
<ul>  
  <li>8 evőkanál liszt</li>  
  <li>2 dl tej </li>  
  <li>1 dl olaj </li>  
  <li>4 evőkanál cukor </li>  
  <li>2 tojás (egész) </li>  
  <li>1 csomag sütőpor </li>  
  <li>1/4 citrom reszelt héja </li>  
  <li>2 csomag vaníliás cukor </li>  
</ul>
```

```
<h2>Elkészítés</h2>
```

```
<ol>  
  <li>A hozzávalókat egy tálban összekeverjük. </li>  
  <li>A formába helyezett muffinpapírt olajjal  
kikenjük.</li>  
  <li>Előmelegített sütőben 165°C-on 25 percig  
sütjük.</li>  
</ol>
```

HTML5 HTML5

CSS3

CSS3
JavaScript

JavaScript

Botscript

HIPERHIVATKOZÁSOK (LINKEK) (HYPERLINKS)

Akkor lesz a weboldalad igazán a web része, ha hivatkozol más oldalakra, és mások is hivatkoznak az oldalaidra.

JAVASCRIPT

HIPERHIVATKOZÁS KÉSZÍTÉSE

Feladat:

A menüben az **Új film beküldése** szöveg az *urlap.html* oldalra mutasson. A *title* attribútum értékét is töltsük ki! (Értéke: Ugrás az űrlapra, ahol saját kedvenc filmet lehet beküldeni). Nézzük meg mi lesz a *title* paraméter hatása!

Hozzuk létre az *urlap.html* oldalt, és helyezzünk el benne egy linket, amellyel vissza lehet térni a kezdőlapra.

Hiperhivatkozás készítésére az **<a>** címkét használjuk.

Az **<a>** és **** között kell megadni a link szövegét.

Gyakori paraméterei:

- **href="webcím"**: a hivatkozás webcíme

Példa

```
<a href="https://www.budapest.hu" target="_blank">Budapesten</a>  
születtem
```

HIPERHIVATKOZÁS KÉSZÍTÉSE

Feladat:

A forrásként megadott linkek új fülön nyílnak meg! A linkeket a dokumentumból kimásolhatjuk.

Gipsz Jakab email címénél használjuk a mailto: sémát! Mi lesz ennek az eredménye?

Hiperhivatkozás készítésére az `<a>` címkét használjuk.

Gyakori paraméterei:

- `href="webcím"`: a hivatkozás webcíme
- `target="_blank"` : ekkor új fülön nyílik meg az oldal

Példa:

```
<a href="https://www.budapest.hu"
target="_blank">Budapest</a> születtem
```

```
<a href=
"mailto:gipszjakab@mailinator.com">
gipszjakab@mailinator.com</a>
```

HIPERHIVATKOZÁS KÉSZÍTÉSE

Feladat:

Oldjuk meg, hogy az oldalmenü elemeire kattintva az oldal adott részére kerüljünk! Ehhez megfelelő id-eket kell kiosztani a film nevét tartalmazó címsoroknál!

Pl.

```
<h2 id="film1">
```

- Az oldal egy részét elláthatjuk egyedi azonosítóval (**id**), amelyre hivatkozhatunk is.
- Ekkor # jelet kell tenni a név elé.

Példa:

```
<h2 id="kepek">Képek</h2>  
<a href="#kepek">Ugrás a képekre</a>
```

Az id-ban megadott szöveg kis/nagybetű érzékeny! Nem mindegy, hogy Képek, vagy kepek.



ELŐRE FORMÁZOTT SZÖVEG (PREFORMATTED TEXT), ILLETVE SORTÖRÉS (BREAK) HASZNÁLATA

Amikor számítanak a white-space karakterek

ELŐRE FORMÁZOTT SZÖVEG, PROGRAMKÓD

Feladat:

A programkódot helyezzük el a megfelelő címkékben!

Érdemes lehet a kódot újra kimásolni a dokumentumból, hogy a sorok ne legyenek bekezdés címkékben elhelyezve.

Mit tapasztalunk a megjelenés során?
Miben más a szöveg, mintha bekezdés lenne?

- Előre formázott szöveget a `<pre>` és `</pre>` tagek között helyezhetünk el.
- Ha ez a szöveg egyben programkód is, akkor azt a `<code>` taggel jelölhetjük.

Példa:

```
<pre><code>  
print "$wget\n";  
$page = ` $wget`;  
</code></pre>
```

SORTÖRÉSE ALKALMAZÁSA

Feladat:

A bevezetőben olvasható, „*Hogy ez igaz-e*” szöveg kezdődjön új sorban a bekezdésen belül!

A készítőről szóló információt tördeljük több sorba az alábbiak szerint:

Készítette: Gipsz Jakab
(gipszjakab@mailinator.com)
1117 Budapest, Nevenincs u. 17

- Sortörést a **
** címkével hozhatunk létre.
- Páratlan címke, nincs zárópárja.

Példa:

```
<p>  
    Az e-mail címem:<br>  
    gipszjakab@elte.hu  
</p>
```


HTML5 HTML5

CSS3

CSS3
JavaScript

Javascript

Botscript

ÁBRA, ILLUSZTRÁCIÓ

Végre lesznek képek is az oldalon!

JAVASCRIPT

ÁBRA, ILLUSZTRÁCIÓ BEILLESZTÉSE

<figure> ábra, illusztráció
<figcaption> felirat

Feladat:

A filmes példánkban is meg kell jeleníteni egy forráskódot, amelyhez a `<pre>` és `<code>` tageket használtuk.

Egészítsük ki ezt a `<figure>` és `<figcaption>` tagekkel is, hiszen ez a kód egyben egy illusztráció is.

```
<figure>
<pre><code>
  for i:=1 to n do
    begin
      write('Kérem a tömb ',i,'. elemét: ');
      readLn(t[i]);
    end;
</code></pre>
  <figcaption>Ciklussal bekérjük a tömb elemeit</figcaption>
</figure>
```


KÉP BEILLESZTÉSE

Feladat:

Illesszük be az összes képet az index.html oldalba!

Használjuk a `<figure>` és `<figcaption>` címkéket is!

`` kép beszúrása

- `src` = "url" a kép forrása **(Kötelező!)**
- `alt` a kép rövid szöveges leírása **(akadálymentesség!) (Kötelező!)**
- `title` kapcsolódó, kiegészítő információ
- `width` szélesség (képpontban)
- `height` magasság (képpontban)

Példa:

```
<figure>
  
  <figcaption>A Mátyás-templom tetőcserepei</figcaption>
</figure>
```

Ajánlott emmet kód:

```
figure>img+figcaption
```



HTML5 HTML5

CSS3

CSS3
JavaScript

JavaScript

Botscript

BEÁGYAZOTT ELEMENK (*EMBEDDED ELEMENTS*)

Hangállomány, videó és más nyalánkságok...

JAVASCRIPT

HANGÁLLOMÁNY BEÁGYAZÁSA

Feladat: Illesszük be az oldalba a megfelelő hangállományt!

Példa:

```
<audio controls  
      src="pelda.mp3">  
</audio>
```

<audio> hang beszúrása

- **src** = "url" a hang url-je (**Kötelező!**)
- **controls** vezérlők megjelenítése
- **loop** folyamatos ismétlés
- **muted** elnémítás

Legszélesebb körben támogatott
formátumok: **mp3**, **ogg** **vorbis**

VIDEÓÁLLOMÁNY BEÁGYAZÁSA

Feladat: Illesszük be az oldalba a megfelelő videóállományt!

Példa:

```
<video controls  
      src="pelda.mp4"  
      poster="poszter.jpg">  
</video>
```

<video> videó beszúrása

- **src** = "url" a videó url-je **(Kötelező!)**
- **controls** vezérlők megjelenítése
- **loop** folyamatos ismétlés
- **muted** elnémítás
- **autoplay** automatikus lejátszás
- **poster** kép (poszter) elérhetősége, amely az indítás előtt látszik

Az automatikus lejátszás sok böngészőben csak akkor működik, ha a videó le van némítva.

Legszélesebb körben támogatott formátumok: **mp4**, **webm**

BEÁGYAZOTT KERET HASZNÁLATA

Feladat: Illesszük be az oldalba a megfelelő slideshare prezentációt a Deep Web című filmnél!

```
<iframe  
src="http://www.slideshare.net/slidesho  
w/embed_code/key/19oozb2TPXXn14"  
width="595" height="485"  
allowfullscreen>  
</iframe>
```

A kód elérhető itt:

<https://justpaste.it/5r0dz>

<iframe> keret beszúrása

- **src** = "url" a forrás url-je **(Kötelező!)**
- **width** keret szélessége
- **height** keret magassága
- **allowfullscreen** engedélyezett a teljes képernyőn történő megjelenés

Példa:

```
<iframe  
src="http://www.wikipedia.hu"  
width="400" height="400">  
</iframe>
```


HTML5 HTML5

CSS3

CSS3
JavaScript

JavaScript

Botscript

STÍLUSLAP CSATOLÁS

JAVASCRIPT

STÍLUSLAP CSATOLÁSA

- A letöltött csomagban találunk egy félig kész stíluslap állományt is.
- Linkeljük be az index.html oldalhoz! A `<head>` részbe írjuk be a következőt, vagy használjuk a `link:css` emmet parancsot!

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

- A `<link>` tag páratlan címke, vagyis nincs záró párja!
- Nézzük meg hogyan változott meg az oldalunk megjelenése a böngészőben!

HTML5 HTML5

CSS3

CSS3
JavaScript

JavaScript

Botscript

CSOPORTOSÍTÓ ELEMEEK

JAVASCRIPT

CSOPORTOSÍTÓ ELEMEEK

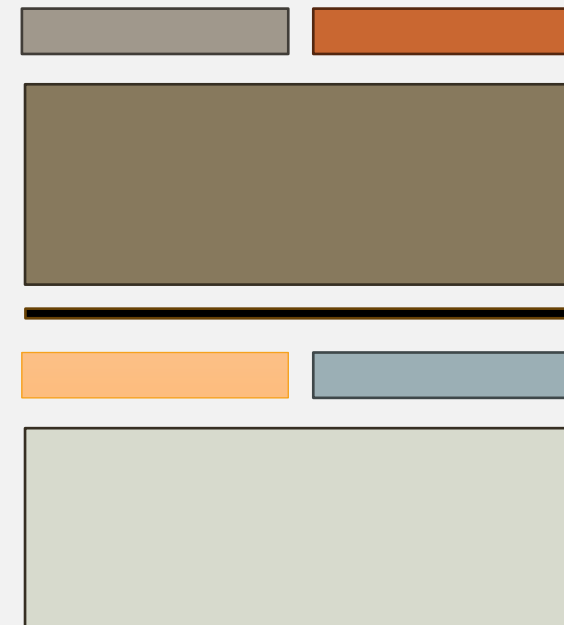
<p> bekezdés (*ezt már használtuk*)

<hr> vízszintes elválasztó vonal (nincs záró párja!)

Feladat:

Használjuk a `<hr>` elemet arra, hogy az egyes filmekről szóló tartalmakat elválasszuk egymástól. Legyen elválasztó az 1. és 2. film között, a 2. és 3. film között, a 3. és 4. film között, és a 4. film és a Források között is!

Keresés/csere funkcióval hogyan lehetne gyorsítani a feladat megoldását?



CSOPORTOSÍTÓ ELEMEEK

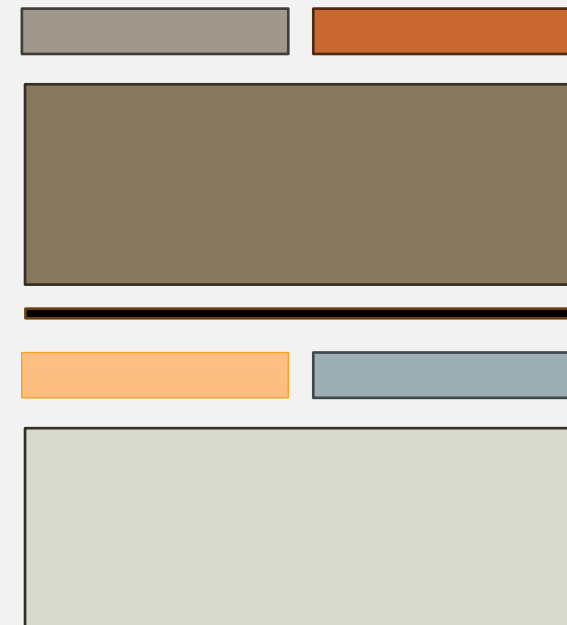
<div> blokkszintű elemek csoportosítására jó

- **Blokkszintű elemek**

- Mindig új sort kezdenek, és teljes szélességet (szülőelem szélességét) elfoglalják.
- Több tartalmi elemet is magukban foglalhatnak.
- Gyakran strukturális elemek, amelyek a dokumentum felépítését segítik.

Példa a blokkszintű elemekre:

```
<p>                <!-- Bekezdés -->
<h1> - <h6>      <!-- Címsorok -->
<ul>, <ol>       <!-- Lista -->
<li>             <!-- Listaelemek -->
```



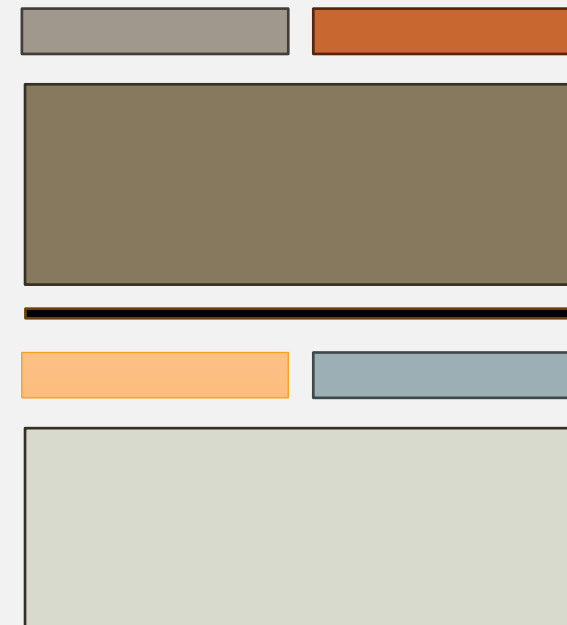
CSOPORTOSÍTÓ ELEMEEK

`<div>` blokk szintű elemek csoportosítása jó

Feladat:

A `<div>` elemet használjuk a bevezető szöveg és az oldalmenü csoportosítására. A `<div>` egyedi azonosítója (`id`) a „bevezeto” szöveg legyen!

```
<div id="bevezeto">  
  <h2>Bevezető</h2>  
  ...  
  ...  
  ...  
  </ol>  
</div> <!-- bevezeto -->
```



CSOPORTOSÍTÓ ELEMEEK

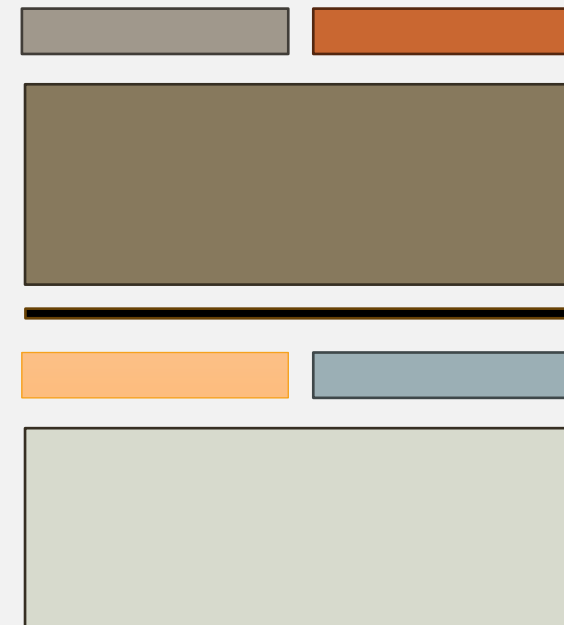
**** soron belüli elemek csoportosítása jó

- **Soron belüli (inline) elemek**

- Nem kezdenek új sort, hanem a meglévő tartalommal egy sorban maradnak.
- Csak annyi szélességet foglalnak el, amennyi szükséges a tartalmukhoz.
- Általában szövegformázásra vagy kisebb részek kiemelésére szolgálnak.

Példa soron belüli elemekre:

<a>	<!-- Hivatkozás -->
<code>	<!-- Kódformázás -->
	<!-- Kép -->
<label>	<!-- Címke űrlapmezőkhöz -->



CSOPORTOSÍTÓ ELEMEEK

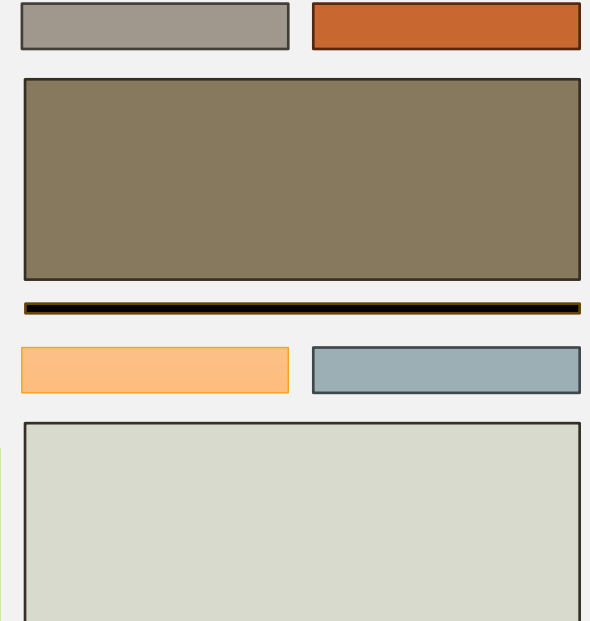
**** soron belüli elemek csoportosítása jó

- Gyakran használjuk arra, hogy a szövegben előforduló idegennyelvű szöveg nyelvkódját beállítsuk. pl.
- **Social Network**

Feladat:

Nézzük át az index.html oldalunkat, és ahol találunk idegen kifejezést, zárjuk körbe **** taggel és állítsuk be a nyelvkódot!

(<https://www.andiamo.co.uk/resources/iso-language-codes/>)



Számos előnye van, például a keresőprogramok el tudják dönteni milyen nyelven íródott a szöveg, a vak emberek által használt képernyőolvasó pedig megfelelő nyelven képes felolvasni a szöveget.



SZÖVEGRÉSZEK SZEREPÉNEK JELÖLÉSE

Profi webfejlesztők is sokszor rosszul használják ezeket.

Fontos a szemantika! Attól, hogy két különböző szöveg félkövér stílusú, nem biztos, hogy ugyanaz a szerepük!

GYAKRAN HASZNÁLT TAGEK

<i>
</i>

dőlt (kézírásra emlékeztető) betű (*italic*)

Olyan szövegrész jelölésére, amely más hangnemben, hangulatban vagy bármely más módon eltér a szöveg többi részétől, mint például egy rendszertani elnevezés, egy szakkifejezés, egy köznyelvi kifejezés, egy rövidke más nyelven írt részlet vagy egy gondolat.

Dőlt betűvel jelenik meg alapesetben.

hangsúlyos kiemelés (*emphasize*)

Ha bizonyos szövegrésznek nagyobb hangsúlyt akarunk adni.
Pl.

```
<p><em>Peti</em> most tanulja a HTML nyelvet.</p>  
<p>Peti most tanulja a <em>HTML nyelvet</em>.</p>
```

Dőlt betűvel jelenik meg alapesetben.

GYAKRAN HASZNÁLT TAGEK

`<i>`
`</i>`

dőlt (kézírásra emlékeztető) betű (*italic*)

Olyan szövegrész jelölésére, amely más hangnemben, hangulatban vagy bármely más módon eltér a szöveg többi részétől, mint például egy rendszertani elnevezés, egy szakkifejezés, egy köznyelvi kifejezés, egy rövidke más nyelven írt részlet vagy egy gondolat.

Dőlt betűvel jelenik meg alapesetben.

Feladat:

A film leírásoknál az utolsó bekezdés egy véleményt tartalmaz. Ez hangulatában eltér az előző szövegtől, így tegyük `<i></i>` tagek közé!

GYAKRAN HASZNÁLT TAGEK

``
``

hangsúlyos kiemelés (*emphasize*)

Ha bizonyos szövegrésznek nagyobb hangsúlyt akarunk adni.
Pl.

```
<p><em>Peti</em> most tanulja a HTML nyelvet.</p>
```

```
<p>Peti most tanulja a <em>HTML nyelvet</em>.</p>
```

Dőlt betűvel
jelenik meg
alapesetben.

Feladat:

„A lista nem átfogó, erősen szubjektív.” szövegben hangsúlyozzuk jobban azt, hogy szubjektív. Használjuk az `` címkéket!

GYAKRAN HASZNÁLT TAGEK

<code></code> <code></code>	félkövér betű (<i>bold</i>) Ha fel akarjuk hívni valamire a figyelmet, de nem akarunk további extra fontosságot tulajdonítani neki. Kulcsszavak megkülönböztetésére, dokumentumok összefoglalójának jelölésére, termék nevek kiemelésére remekül használható.	Félkövér stílussal jelenik meg alapesetben.
<code></code> <code></code>	erős kiemelés (<i>strong</i>) Nagy jelentőséggel bíró tartalomra vonatkozik, beleértve a nagyon komoly vagy sürgős dolgokat (például figyelmeztetéseket).	Félkövér stílussal jelenik meg alapesetben.

GYAKRAN HASZNÁLT TAGEK

``
``

félkövér betű (*bold*)

Ha fel akarjuk hívni valamire a figyelmet, de nem akarunk további extra fontosságot tulajdonítani neki. Kulcsszavak megkülönböztetésére, dokumentumok összefoglalójának jelölésére, termék nevek kiemelésére remekül használható.

Félkövér stílussal jelenik meg alapesetben.

Feladat:

Melyek azon szavak a szövegben, amelyek kulcsszónak tekinthetőek? Mi az, amire a látogatót tekintetét rá akarjuk irányítani? Tegyük ezeket `` tagek közé!

GYAKRAN HASZNÁLT TAGEK

erős kiemelés (*strong*)

Nagy jelentőséggel bíró tartalomra vonatkozik, beleértve a nagyon komoly vagy sürgős dolgokat (például figyelmeztetéseket).

Félkövér stílussal jelenik meg alapesetben.

Feladat:

A „Figyelem! A lista nem átfogó, erősen szubjektív.” szövegben erősen emeljük ki a „Figyelem!” kifejezést a **** használatával!

FELOLVASÁS A KÉPERNYŐOLVASÓ PROGRAMOKBAN

- A látássérült emberek egy része képernyőolvasó (screen reader) programot használ, amely felolvassa a honlap tartalmát.
- Szerintetek megkülönbözteti-e a program a `` `` `<i>` `` címkékkel jelölt szövegeket?

Korábban az NVDA megkülönböztette ezeket, de mivel mindenki ész nélkül (nem szabvány szerint) használta ezeket a HTML kódban, a negatív felhasználói visszajelzések miatt kiszedték ezt a funkciót. Kár érte!

<https://www.tpgi.com/screen-readers-support-for-text-level-html-semantics/>



IDÉZET

<q>
</q>

Egy más forrásból származó idézet jelölésére szolgál.

Az idézet jelölésére szolgáló idézőjelet a kliensalkalmazás fogja megjeleníteni a szabvány szerint, így azt nem szabad kitenni.

Nem minden idézőjel közti szöveg tölt be idézet szerepet, így automatikusa sose cseréljük le az idézetjelet <q> tagekre! Példa: Sokszor voltam nehéz helyzetben, de mindig sikerült „kimásznom belőle”. Itt stiláris okokból szerepel idézőjel, nem szabad <q> taggel helyettesíteni.

Feladat:

A Krasznahorkai Lászlótól származó idézetet helyezzük el a <q></q> tagek között, és töröljük az eredeti idézőjelet!

KREATÍV MUNKA CÍME

`<cite>`
`</cite>`

Egy alkotás (könyv, cikk, esszé, dal, film, játék, művészeti alkotás stb.) címére hivatkozás során használatos.

Segít a keresőmotoroknak megérteni, hogy az adott tartalom egy forrásmegjelölés. A keresőmotorok (pl. Google) felismerik a `<cite>` elemet, ami segíthet a tartalom hitelességének és strukturáltságának javításában.

Feladat:

Azonosítunk a szövegben ilyen címeket, és használjuk a `<cite></cite>` taget! (nem kell az összes címet így átalakítani)

RÖVIDÍTÉS JELÖLÉSE

`<abbr>`

`</abbr>`

Rövidítések jelölésére, és azok feloldásának megadására szolgál.

A `title` paraméterben adjuk meg a rövidítés feloldását.

Példa:

`<p>`A szerző ebben a témában írt publikációja megjelent az
`<abbr title="Fertődi Növénynevelési és Növénytermesztési
Kutató Intézet Közleményei">`FNNKIK`</abbr>`-ban.`</p>`

Feladat:

Az IMDB szövegnél jelöljük, hogy az „Internet Movie Database” rövidítésről van szó!
Használjuk a korábban bemutatott csere funkciót, hogy gyorsabban végezzünk!

DÁTUM ÉS IDŐ MEGADÁSA

Példa:

`<p>Késő este,
<time datetime="2008-07-13T20:13">20:13</time> órakor
születtem.</p>`

Feladat:

Az első film bemutatójának dátumát adjuk meg precízen a `<time>` címke használatával!

`<time>` dátum/idő megadása

- `datetime="dátum/idő"`

Description	Microsyntax	Examples
Valid month string	<code>YYYY-MM</code>	<code>2011-11</code> , <code>2013-05</code>
Valid date string	<code>YYYY-MM-DD</code>	<code>1887-12-01</code>
Valid yearless date string	<code>MM-DD</code>	<code>11-12</code>
Valid time string	<code>HH:MM</code> <code>HH:MM:SS</code> <code>HH:MM:SS.mmm</code>	<code>23:59</code> <code>12:15:47</code> <code>12:15:52.998</code>
Valid local date and time string	<code>YYYY-MM-DD HH:MM</code> <code>YYYY-MM-DD HH:MM:SS</code> <code>YYYY-MM-DD HH:MM:SS.mmm</code> <code>YYYY-MM-DDTHH:MM</code> <code>YYYY-MM-DDTHH:MM:SS</code> <code>YYYY-MM-DDTHH:MM:SS.mmm</code>	<code>2013-12-25 11:12</code> <code>1972-07-25 13:43:07</code> <code>1941-03-15 07:06:23.678</code> <code>2013-12-25T11:12</code> <code>1972-07-25T13:43:07</code> <code>1941-03-15T07:06:23.678</code>

Használható formátumok:

<https://developer.mozilla.org/en-US/docs/Web/HTML/Element/time>

HTML5 HTML5

CSS3

CSS3
JavaScript

JavaScript

Botscript

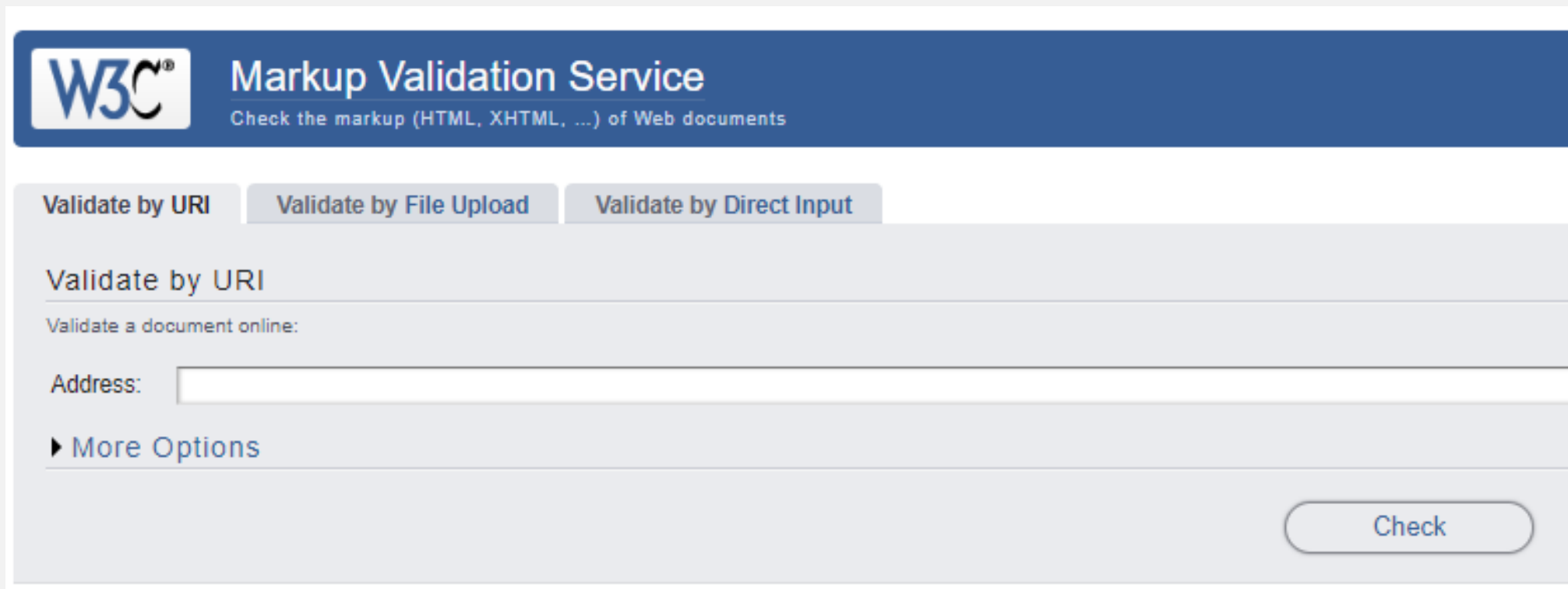
VALIDÁCIÓ (VALIDATION)

Validáljuk, amit eddig készítettünk! Ha vannak hibák, javítsuk ki!

JAVASCRIPT

VALIDÁLÁS

- A <https://validator.w3.org/> oldal segítségével nézzük meg, hogy szabványos-e, amit eddig csináltunk!



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 right of the form, there is a "Check" button. Below the input field, there is a link "► More Options".

HTML5 HTML5

CSS3

CSS3
JavaScript

JavaScript

Botscript

VÉGE AZ I. RÉSZNEK

JAVASCRIPT