



01 - HTML

Előadás demó leírása a Mobil- és webes szoftverek c.
tárgyhoz

Gincsei Gábor
2017.

Szerzői jogok

Jelen dokumentum a BME Villamosmérnöki és Informatikai Kar hallgatói számára készített elektronikus jegyzet. A dokumentumot a Mobil- és webes szoftverek c. tantárgyat felvevő hallgatók jogosultak használni, és saját céljukra 1 példányban kinyomtatni. A dokumentum módosítása, bármely eljárással részben vagy egészben történő másolása tilos, illetve csak a szerző előzetes engedélyével történhet.



BEVEZETÉS

Ebben a dokumentumban röviden áttekintjük az előadáson bemutatott demókat, hogy az a későbbiekben otthon önállóan is el lehessen végezni.

CÉLKITŰZÉS

A demók elvégzésének segítségével gyakorlatiasan meg lehet ismerkedni a HTML készítésének folyamatával, ki lehet próbálni az előadáson elhangzott száraznak tűnő fóliákat.

A demók kifejezetten olyan hallgatóknak szólnak, akiknek a HTML még újdonság, abban tapasztalatuk csekély.

MIÉRT ÉRDEMES MEGISMERKEDNI A HTML-LEL?

Joggal merülhet fel a kérdés, hogy miért tanuljak meg HTML-ben weboldalt készíteni, ha később azt úgyis ASP.NET-ben vagy bármilyen más szerver oldali technológiával fogjuk generálni.

A válasz igen egyszerű. Napjainkban a kliens oldali fejlesztés ismét reneszánszát éli az Angular, React és egyéb kliens oldali technológiáknak köszönhetően. ahol az oldal vázát HTML-ben kell összeállítani szerver oldali segítség nélkül.

Ezen felül egy másik terület, ahol igenis szükséges a HTML ismeret az a sitebuild készítése. Itt ugyanis az a feladat, hogy az elkészített design tervek alapján egy kattintható működő HTML + CSS + JS vázat készítsünk, ami megmutatható a megrendelőnek, hogy a tényleges fejlesztés elkezdése előtt egyeztetni lehessen vele, hogy tényleg azt és úgy szeretné ahogy a fejlesztőkhöz eljutott.

01 – HELLOVILAG.HTML

Ez az oldal egy minimális beállításokat tartalmazó Helló világ szöveget megjelenítő HTML kódot ismertet.

```
<!DOCTYPE html>
<html>
  <head>
    <title>Oldal címe</title>
  </head>
  <body>
    Helló világ!
  </body>
</html>
```

- A doctype mondja meg a böngészőnek, hogy milyen típusú és verziójú HTML-t használ az oldal.
- A <html> az oldal gyökéreleme, ezen belül található minden.
- A head szekcióban adhatunk meg az oldalra vonatkozó meta adatokat.

02 – HELLOVILAGMETA.HTML

Ez a demo oldal annyival tud többet az előzőnél, hogy több beállítást tartalmaz, illetve CSS és JS fájlokra is tartalmaz hivatkozást.

```
<!DOCTYPE html>
<html lang="hu">
<head>
  <meta charset="utf-8">
  <meta name="viewport" content="width=device-width, initial-scale=1">
  <title>Első HTML5 oldalam</title>
  <link rel="stylesheet" href="style.css">
</head>
<body>
  <p>Helló világ!</p>
  <!-- TODO: Oldal kódja -->

  <script src="script.js"></script>
</body>
</html>
```

A head szekcióban itt megadtuk az oldalhoz tartozó karakter kódolás, megadtuk, hogy mobil eszközökön a viewport hogyan néz ki, adtunk neki címet

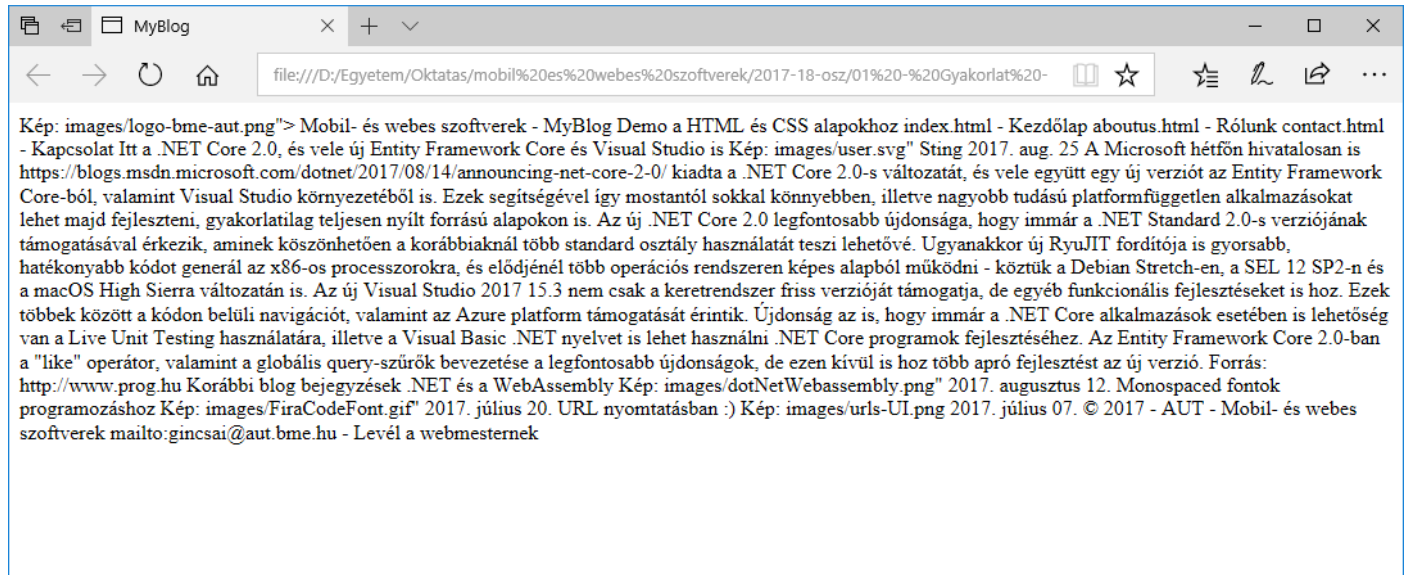
Ezen felül a head szekcióban linkeljük be a style.css fájlt, és teljesítmény okok miatt az oldal végén a </body> előtt közvetlenül adjuk meg a JavaScript hivatkozásokat. Jelen esetben a script.js-re.

03 – OLDALVAZ.HTML

Ez az oldal egy kiindulási váz a Blog oldal elkészítéséhez. Természetesen lehetne teljesen nulláról üres fájlal is indulni, de az sokáig tartana, mert be kellene a statikus tartalmat gépelni.

A kiinduló fájlban a meta adatokkal kiegészített hello világhoz hasonló kód található, csak a <body>-ba bekerül a tartalom, mindenféle formázás és tag-ek nélkül, csak maga a nyers szöveg.

Ha megnézzük böngészőben, akkor látható is, hogy csak egymás mögé írt szöveget tartalmaz az oldal.



Ebből is látható, hogy a beírt szöveg minden extra tag nélkül megjelenik. Figyeljük meg azonban, hogy a HTML-ben adtunk meg sortöréseket, egymás után több szóközt, ám ezeket a böngésző nem jeleníti meg. (Ha azt szeretnénk, hogy ezek is megjelenjenek, akkor a <pre> taget lehet erre használni, ami előre formázott tartalmat fog megjeleníteni, minden white space-szel együtt.)

Egészítsük ki az oldalt a szükségeses HTML tagakkal

1. Fejléc <header>

A fejléc az oldanak vagy egy szekciónak a fejléce, nem összekeverendő a <head> attribútummal!

Ebben a demóban azt szeretnénk, hogy az oldal fejlécében:

- egy kép jelenjen meg, melynek a relatív URL-je: "images/logo-bme-aut.png",
- A kép mellett az alkalmazás nevét címsor 1-el: "Mobil- és webes szoftverek - MyBlog"
- Alatta pedig az alcím egy sima paragrafusban: "Demo a HTML és CSS alapokhoz"
- Majd állítsuk be (egyelőre inline stílussal), hogy a header háttérszíne bordó legyen. (#A4001C)

```
<header style="background: #A4001C;">
  
  <h1>Mobil- és webes szoftverek - MyBlog</h1>
  <p>Demo a HTML és CSS alapokhoz</p>
</header>
```

- Az oldal fejléce így fog kinézni, ha elkészült:



2. Navigációs linkek

A kiinduló fájlban látjuk, hogy mely html oldalkra szeretnénk linket tenni és mögötte megtalálható az is, hogy milyen szöveget szeretnénk megjeleníteni az egyes linkekhez. Nincs más dolgunk, mint

- Hozzuk létre a navigációs blokkot a <nav> taggel.
- Az oldal felépítése szempontjából fontos, hogy a <nav> alatt is szerepeljen egy címsor <h2>, de ez csak a szemantika miatt fontos, megjeleníteni nem szeretnénk, ezért tegyük rá a display: none stílust és a nocontent CSS osztályt.
- Ezt követően egy felsorolásba tegyük bele a linkeket (-be ágyazott -k)
- Az egyes linkek pedig a ... miatt lesznek linkek.

```
<!-- Navigációs linkek -->
<nav>
  <!-- Miért használjuk a nocontent-et?
  https://support.google.com/customsearch/answer/2364585?hl=en -->
  <h2 class="nocontent" style="display: none">--- Navigáció ---</h2>
  <ul>
    <li>
      <a href="index.html">Kezdőlap</a>
    </li>
    <li>
      <a href="aboutus.html">Rólunk</a>
    </li>
    <li>
      <a href="contact.html">Kapcsolat</a>
    </li>
  </ul>
</nav>
```

- A navigáció így fog kinézni ha elkészült:

- [Kezdőlap](#)
- [Rólunk](#)
- [Kapcsolat](#)

3. Főhír megjelenítése (cím, szerző adatai, bevezető, tartalom, forrás)

- Mivel ez lesz az oldalunk fő tartalma ezért ezt tegyük a <main> tag-be.
- Egy cikkről beszélünk, aminek önmagában is van értelme ezért egy <article> tag segítségével fogjuk létrehozni.
- Az <article> tag után meg kell adni egy címet pl <h2>-vel. A cím szövege: "Itt a .NET Core 2.0, és vele új Entity Framework Core és Visual Studio is"

```
<!-- Maga a fő hírünk, ami részekre van bontva -->
<main>
  <article>
    <header>
      <h2>
        Itt a .NET Core 2.0, és vele új Entity Framework Core és Visual Studio is
      </h2>
    </header>
```

- Ezt követően jön egy szekció, amiben a meta adatok találhatóak a szerzőről.
 - i. Egy kép, ami az "images/user.svg"-re mutat és 60px széles.
 - ii. Ezt követően a szerző neve vastag betűvel ()

iii. Majd a hír publikálásának dátuma egy `<time>` tagben.

```
<section id="meta">
  <!-- Megjelenítést HTML attribútumokkal is megadhatunk, de a CSS a preferált megoldás -->
  
  <!-- Lehetne a <b>-t is használni, de <strong> többet mond -->
  <strong>Sting</strong>
  <!-- datetime attribútum értéke JS számára érthető, a tag-ben lévő érték pedig olvasható-->
  <time datetime="2017-08-25T19:00">2017. aug. 25</time>
</section>
```

- Ezután következhet a bevezetés, ami egy külön logikai egység, ami miatt `<section>`-be tesszük

```
<section id="introduction">
  A Microsoft hétfőn hivatalosan is
  <a href="https://blogs.msdn.microsoft.com/dotnet/2017/08/14/announcing-net-core-2-0/">kiadta</a>
  a .NET Core 2.0-s változatát, ... is.
</section>
```

- Majd a tartalom további része, amit már jobban nem bontunk egységekre, de maga a tartalom egy különálló szekció. Azon belül paragrafusokkal szervezzük bekezdésekre a szöveget. `<p>`

```
<section id="content">
  <p>
    Az új .NET Core 2.0 legfontosabb újdonsága, ... is.
  </p>
  <p>
    Az új Visual Studio 2017 15.3 nem csak a keretrendszer ... fejlesztéséhez.
  </p>
  <p>
    Az Entity Framework Core 2.0-ban a "like" operátor, valamint a globális ... verzió.
  </p>
</section>
```

- Végül készítsük el a lábléceket a cikkhez, amiben a forrást jelöljük.

```
<footer>
  Forrás: <a href="http://www.prog.hu">prog.hu</a>
</footer>
```

- A kész cikk így fog kinézni

Itt a .NET Core 2.0, és vele új Entity Framework Core és Visual Studio is



Sting 2017. aug. 25

A Microsoft hétfőn hivatalosan is [kiadta](#) a .NET Core 2.0-s változatát, és vele együtt egy új verziót az Entity Framework Core-ból, valamint Visual Studio környezetéből is. Ezek segítségével így mostantól sokkal könnyebben, illetve nagyobb tudású platformfüggetlen alkalmazásokat lehet majd fejleszteni, gyakorlatilag teljesen nyílt forrású alapokon is.

Az új .NET Core 2.0 legfontosabb újdonsága, hogy immár a .NET Standard 2.0-s verziójának támogatásával érkezik, aminek köszönhetően a korábbiaknál több standard osztály használatát teszi lehetővé. Ugyanakkor új RyuJIT fordítója is gyorsabb, hatékonyabb kódot generál az x86-os processzorokra, és elődjénél több operációs rendszeren képes alapból működni - köztük a Debian Stretch-en, a SEL 12 SP2-n és a macOS High Sierra változatán is.

Az új Visual Studio 2017 15.3 nem csak a keretrendszer friss verzióját támogatja, de egyéb funkcionális fejlesztéseket is hoz. Ezek többek között a kódon belüli navigációt, valamint az Azure platform támogatását érintik. Újdonság az is, hogy immár a .NET Core alkalmazások esetében is lehetőség van a Live Unit Testing használatára, illetve a Visual Basic .NET nyelvet is lehet használni .NET Core programok fejlesztéséhez.

Az Entity Framework Core 2.0-ban a "like" operátor, valamint a globális query-szűrők bevezetése a legfontosabb újdonságok, de ezen kívül is hoz több apró fejlesztést az új verzió.

Forrás: [prog.hu](http://www.prog.hu)

4. További blog bejegyzések mutatása

- Mivel azt itt elkészített tartalom nem kapcsolódik szorosan az oldal fő tartalmához ezért ezt az `<aside>` tag segítségével fogjuk létrehozni, aminek egy címet is adjunk meg.

```
<aside>
  <header>
    <h2>Korábbi blog bejegyzések</h2>
  </header>
```

- Majd jöhetnek az egyes cikkek külön-külön article-be.

```
<article>
  <h3>.NET és a WebAssembly</h3>
  <figure>
    
  </figure>
  <time datetime="2017-08-12">2017. augusztus 12.</time>
</article>
<article>
  <h3>Monospaced fontok programozáshoz</h3>
  <figure>
    
  </figure>
  <time datetime="2017-07-20">2017. július 20.</time>
</article>
<article>
  <h3>URL nyomtatásban :)</h3>
  <figure>
    
  </figure>
  <time datetime="2017-07-07">2017. július 07.</time>
</article>
```

- A kapcsolódó cikkek így fognak megjelenni:

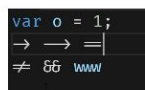
Korábbi blog bejegyzések

.NET és a WebAssembly



2017. augusztus 12.

Monospaced fontok programozáshoz



2017. július 20.

URL nyomtatásban :)



2017. július 07.

5. Már csak az oldanak a láblécét kell elkészíteni

- Ehhez használjuk a <footer> taget, amiben az email címet az <address> tagben adjuk meg úgy, hogy rákattintva a levelező programot indítsa el.

```
<footer>
  <p>
    &copy; 2017 - AUT - Mobil- és webes szoftverek
    <address>
      <a href="mailto:gincsa@aut.bme.hu">Levél a webmesternek</a>
    </address>
  </p>
</footer>
```

- Az elkészült lábléc így fog kinézni:

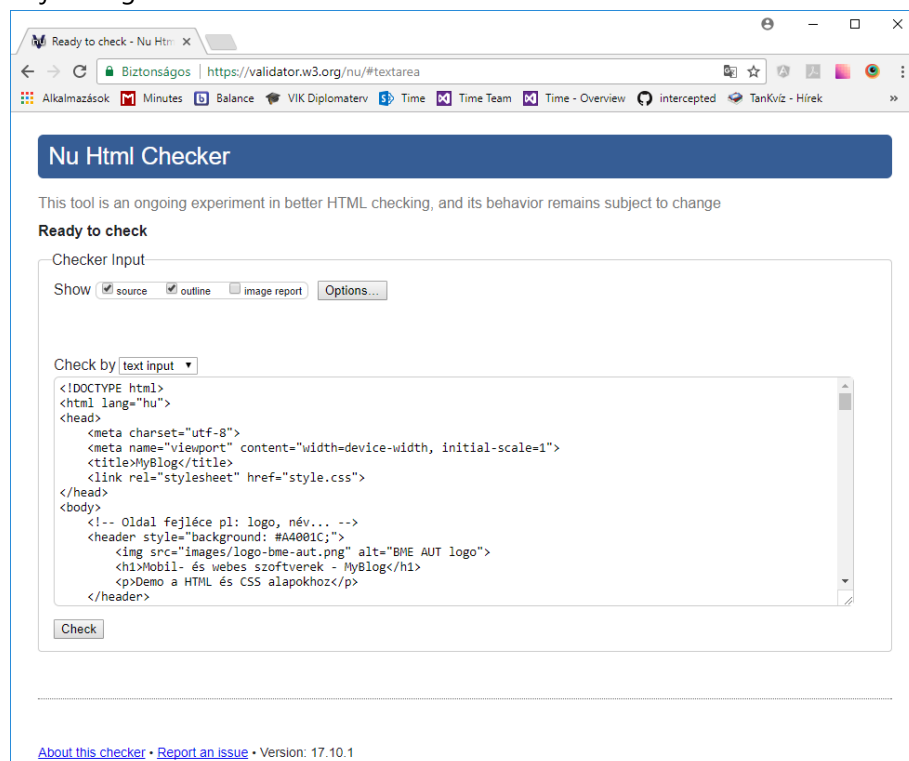
© 2017 - AUT - Mobil- és webes szoftverek

[Levél a webmesternek](mailto:gincsa@aut.bme.hu)

Az elkészített oldal teljes forrása megtalálható a **03b – oldaltvaz.html** fájlban.

A <https://validator.w3.org/nu/> oldal segítségével meg tudjuk nézni, hogy az oldalnak milyen a felépítése és hogy milyen esetleges hibákat követtünk el a HTML készítése során. Ehhez az alábbi módon adjuk meg a validációs beállításokat.

- Pipáljuk be az outline-t is.
- Egy text inputba adjuk meg a HTML kódunkat



- Ezt követően kattintsunk a check gombra, ami ellenőrzi a textareaban megadott HTML-t.
- Az oldalra az alábbi négy figyelmeztetést kapjuk:

1. **Warning** Section lacks heading. Consider using `h2` or `h3` elements to [add identifying headings to all sections](#).
From line 44, column 9 to line 44, column 27
`<section id="meta">`
2. **Warning** Section lacks heading. Consider using `h2` or `h3` elements to [add identifying headings to all sections](#).
From line 53, column 9 to line 53, column 35
`<section id="introduction">`
3. **Warning** Section lacks heading. Consider using `h2` or `h3` elements to [add identifying headings to all sections](#).
From line 68, column 9 to line 68, column 30
`<section id="content">`
4. **Warning** The `date` input type is not supported in all browsers. Please be sure to test, and consider using a polyfill.
From line 130, column 17 to line 130, column 70
`<input id="txtBirthDate" name="birthDate" type="date">`

- A készített oldal felépítése az alábbi:

Heading-level outline

```

<h1> Mobil- és webes szoftverek - MyBlog
<h2> --- Navigáció ---
<h2> Itt a .NET Core 2.0, és vele új Entity Framework Core és Visual Studio is
<h2> Korábbi blog bejegyzések
<h3> .NET és a WebAssembly
<h3> Monospaced fontok programozáshoz
<h3> URL nyomtatásban :)
<h3> Feliratkozás a hírlevélre

```

Structural outline

```

└─ Mobil- és webes szoftverek - MyBlog
    └─ --- Navigáció ---
        └─ Itt a .NET Core 2.0, és vele új Entity Framework Core és Visual Studio is
            └─ [section element with no heading]
            └─ [section element with no heading]
            └─ [section element with no heading]
        └─ Korábbi blog bejegyzések
            └─ .NET és a WebAssembly
            └─ Monospaced fontok programozáshoz
            └─ URL nyomtatásban :)
            └─ Feliratkozás a hírlevélre

```

További részletes információ a szemantikus weboldalak felépítésére HTML5 segítségével:

<https://internetingishard.com/html-and-css/semantic-html/>

04 – FELIRATKOZAS.HTML

Az oldal szélére, a kapcsolódó cikkek alá hozzunk létre egy űrlapot, amiben fel tudunk iratkozni egy hírlevélre. Magát a funkciót nem valósítjuk most meg, csak az adatokat kérjük be és küldjük fel a szerverre.

1. Készítsünk egy űrlapot a `<form>` tag segítségével. Mivel most nincs szerver oldali főlösleges megadni az action-t, így ugyanarra az oldalra fogja GET kéréssel elküldeni a form-on megadott adatokat mint ahol most vagyunk.
2. Ezt követően a névnek, email címnek és a születési dátumnak hozzunk létre egy-egy beviteli mezőt. Minden beviteli mezőhöz tartozzon egy címke, és figyeljünk a beviteli mezők típusára.
3. A név és email cím legyen kötelező, használjuk a `required` attribútumot ehhez.
4. Majd az űrlap aljára tegyünk egy gombot `<button>` ami elküldi az űrlapot.

```
<!-- Feliratkozás a hírlevélre -->
<article>
  <h3>Feliratkozás a hírlevélre</h3>
  <form>
    <label for="txtName">Név: </label>
    <input id="txtName" name="name" type="text" required>

    <label for="txtEmail">Email cím:</label>
    <input id="txtEmail" name="email" type="email" required>

    <label for="txtBirthDate">Születési dátum</label>
    <input id="txtBirthDate" name="birthDate" type="date">

    <button type="submit">Feliratkozok</button>
  </form>
</article>
```

Az elkészített űrlap így fog kinézni:

Feliratkozás a hírlevélre

Név: Email cím: Születési dátum

05 – VIDEO.HTML

Érdekesség gyanánt nézzük meg, hogyan lehet egy weboldalba egy videót beágyazni a <video> tag segítségével.

1. Ehhez a cikkben egy <figure> taget hozunk létre ezzel jelezve, hogy itt egy kép lesz.
2. Adjuk hozzá a <figure>-hoz a videot, aminek az adatai az alábbiak
 - Url: https://sec.ch9.ms/ch9/1449/d2871ca7-aaaa-46fc-915b-49c5adc81449/DotNetCore2_high.mp4
 - Előnézet url: https://sec.ch9.ms/ch9/1449/d2871ca7-aaaa-46fc-915b-49c5adc81449/DotNetCore2_960.jpg
 - Legyenek bekapcsolva a vezérlők a videon.
 - A szélességet állítsuk 500px-re.

```
<!-- Video -->
<figure>
  <video src="https://sec.ch9.ms/ch9/1449/d2871ca7-aaaa-46fc-915b-49c5adc81449/DotNetCore2_high.mp4"
    poster="https://sec.ch9.ms/ch9/1449/d2871ca7-aaaa-46fc-915b-49c5adc81449/DotNetCore2_960.jpg"
    controls
    style="width: 500px;">
  </video>
</figure>
```

3. A szövegbe ágyazott video így fog kinézni:

A Microsoft nemon hivatalosan is [kiadta](#) a .NET Core 2.0-s változatát, és vele együtt egy új is. Ezek segítségével így mostantól sokkal könnyebben, illetve nagyobb tudású platformfüggetlen alapokon is.



Az új .NET Core 2.0 legfontosabb újdonsága, hogy immár a .NET Standard 2.0-s verzióján osztható használatát teszi lehetővé. Ugyanakkor új RyuJIT fordítója is gyorsabb, hatékonyabb