1. gyakorlat – Strukturált adatmegjelenítés  
   Táblázatok, táblázatok, táblázatok

Táblázatok szerepe a weblapon is ugyanaz, mint egy word dokumentumban: az adatok strukturált megjelenítésére. (Nagyon sokan helytelenül az oldalfelépítés kialakítására is táblázatokat használnak. Kétségkívül erre is alkalmasak, de eredetileg nem erre találták ki őket.

Nézzük, hogy is épül fel egy táblázat html nyelven:

### Táblázat kialakítása

A táblázatot mindig a <table>...</table> címkék közé kell írni. A táblázat egy sora a <tr>...</tr>, ezen belül pedig a cellák a <td>...</td> címkék közé kerülnek. A table és a tr cimkék közé nem kerülhet szöveg!

<table>

  <tr>

    <td>első sor, első oszlop</td>

    <td>első sor, második oszlop</td>

    <td>első sor, harmadik oszlop</td>

  </tr>

  <tr>

    <td>második sor, első oszlop</td>

    <td>második sor, második oszlop</td>

    <td>második sor, harmadik oszlop</td>

  </tr>

  <tr>

    <td>harmadik sor, első oszlop</td>

    <td>harmadik sor, második oszlop</td>

    <td>harmadik sor, harmadik oszlop</td>

  </tr>

</table>

|  |  |  |
| --- | --- | --- |
| első sor, első oszlop | első sor, második oszlop | első sor, harmadik oszlop |
| második sor, első oszlop | második sor, második oszlop | második sor, harmadik oszlop |
| harmadik sor, első oszlop | harmadik sor, második oszlop | harmadik sor, harmadik oszlop |

Ha egy cellát üresen akarunk hagyni, akkor a tartalma legyen &nbsp; (non-breaking space).

<table>

  <tr>

    <td>első sor, első oszlop</td>

    <td>&nbsp;</td>

    <td>első sor, harmadik oszlop</td>

  </tr>

  <tr>

    <td>&nbsp;</td>

    <td>második sor, második oszlop</td>

    <td>második sor, harmadik oszlop</td>

  </tr>

</table>

|  |  |  |
| --- | --- | --- |
| első sor, első oszlop |  | első sor, harmadik oszlop |
|  | második sor, második oszlop | második sor, harmadik oszlop |

###### Sor- és oszlopátnyúlások megadása

A táblázat egy cellája több oszlopot is átfoghat. Ehhez a td cimkéhez használni kell a colspan attribútumot, melynek értéke mutatja az összevonandó cellák számát.

<table>

Innen tudja, hogy a 3 egymás alatti cellát kell összevonni.

  <tr>

<td>első sor, első oszlop</td>

<td>első sor, második oszlop</td>

<td>első sor, harmadik oszlop</td>

  </tr>

  <tr>

<td colspan="3">A második sor egyetlen cella lett, a colspan attribútum értékét 3-ra állítottam.</td>

  </tr>

  <tr>

<td>harmadik sor, első oszlop</td>

<td>harmadik sor, második oszlop</td>

<td>harmadik sor, harmadik oszlop</td>

  </tr>

</table>

|  |  |  |
| --- | --- | --- |
| első sor, első oszlop | első sor, második oszlop | első sor, harmadik oszlop |
| második sor egyetlen cella lett | | |
| harmadik sor, első oszlop | harmadik sor, második oszlop | harmadik sor, harmadik oszlop |

Egy cella több sort is átfoghat. Ehhez a td cimkéhez használni kell a rowspan attribútumot, melynek értéke mutatja az összevonandó cellák számát.

Innen tudja, hogy a 3. sort kell összevonni.

<table>

  <tr>

<td>első sor, első oszlop</td>

<td rowspan="3">A második oszlop egyetlen cella lett, a rowspan attribútum értékét 3-ra állítottam.</td>

<td>első sor, harmadik oszlop</td>

  </tr>

  <tr>

Figyeljük meg, hogy itt csak két cellát adtunk meg.

 <td>második sor, első oszlop</td>

 <td>második sor, harmadik oszlop</td>

  </tr>

  <tr>

<td>harmadik sor, első oszlop</td>

<td>harmadik sor, harmadik oszlop</td>

  </tr>

</table>

|  |  |  |
| --- | --- | --- |
| első sor, első oszlop | második oszlop egyetlen cella lett | első sor, harmadik oszlop |
| második sor, első oszlop | második sor, harmadik oszlop |
| harmadik sor, első oszlop | harmadik sor, harmadik oszlop |

###### Táblázat fejléce és címe

Lehetőségünk van kiemelni a táblázat fejlécét (a táblázat legelső sorát, oszlopát, vagy mindkettőt) és címet rendelni a táblázathoz. A fejléc egy celláját ne a <td>...</td> címkék közé, hanem a <th>...</th> címkék közé írjuk, ekkor a böngésző alapból megvastagítja és középre rendezi a benne lévő szöveget.

A táblázat fejléce lehet egyrészt az oszlopok felett. Ekkor mindegyik th címkének adjuk meg a scope="col" attribútumot, ami azt jelenti, hogy az adott fejléc az alatta lévő oszlophoz tartozik. Ha egy fejléc a mellette lévő sorhoz tartozik, akkor a scope="row" attribútumot kell beállítani.

A táblázat címét a <caption>...</caption> címkék közé írjuk, rögtön a kezdő <table> címke után!

<table >

<caption>

Hány éves a kutyája emberi léptékkel számítva?

</caption>

<col id="kutya" ><col id="10" ><col id="20" ><col id="40" > <col id="50" >

<tr>

<th scope="col">A kutya kora</th>

<th scope="col">10 kg alatt </th>

<th scope="col">10-20 kg</th>

<th scope="col">20-40 kg</th>

<th scope="col">40 kg fölött</tH>

</tr>

<tr>

<td >6</td>

<td>40</td>

<td>42</td>

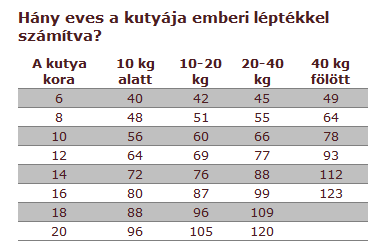
<td>45</td>

<td>49</td>

</tr>

<!-- folytatni kell a mintának megfelelően -->

</table>



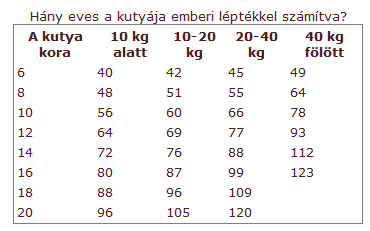
###### Rövid leírás megadása a táblázathoz

A summary-vel, a table címke attribútumaként minden adattáblához adjuk meg a rövid szöveges leírását. Ez tartalmazza, hogy miről szól a táblázat, milyen adatok vannak benne és hogyan kell értelmezni. A summary értéke nem jelenik meg a grafikus böngészőkben, ez a képernyőfelolvasóknak szól.

<table summary="Ez a táblázat azt mutatja meg, hogy hány éves a kutya emberi léptékkel számolva. A kutya kora függhet a kutya súlyától is. ">

### Táblázat formázása

Nyissuk meg a stilus CSS-t, és készítsünk keretet a táblázatunkhoz. A table kijelölőhöz írjuk be a border:1px solid gray; utasítást. Figyeljük meg, hogy a cellák nem lettek bekeretezve, és a táblázat címe is kívül maradt.



Most rakjuk szegélyt a td kijelölőre is!

td {

border: 1px solid gray;

}

Látszik, hogy a rovatfejek, amiket th cimkével láttunk el, nem lettek szegélyezve. Ha ezt is akarjuk, akkor a th –ra is rá kell tenni a keretezést.



A cellák körül is megjelent a szegély, sőt láthatóvá vált, hgy a cellák között van egy kis távolság. Azt gondolhatnánk, hogy ez a külső margó, de ez nem az. Hiába állítjuk a margó értékét 0-ra, nem tűnik el.

A kettős szegély eltüntetésének a módja, hogy a border-collapse:collapse; utasítást írjuk a table kijelölőhöz. Hasonló módon, ha növelni akarjuk a cellák közti távolságot, akkor border-collapse:collapse;helyett használjuk a border-spacing tulajdonságot.

A cellaszegély és szöveg közti távolság megadásához használjuk a padding tulajdonságot, a már ismert módon.

###### Táblázat középre igazítása és méretezése

A táblázatot a margin:auto; utasítással igazíthatjuk középre, éppúgy, mint az odalalakításnál a tartalmazó div-ünket.

Szokás még megadni a táblázat szélességét. Alapértelmezetten kitölti a teljes rendelkezésre álló helyet, de mondhatuk azt, hogy a legyen, mondjuk a tárolónk 90%-a (width:90%), vagy megadhatunk konkrét szélességet is. Vigyázzunk arra, hogy ha nagyobb szélességet adunk meg, mint a rendelkezésre álló hely, akkor csúnyán széttördelheti az oldalunkat!

Ehhez a szabvány szerint a bal és jobb oldali margót egyaránt auto értékre kell állítanunk. A szélességet a width tulajdonsággal tudjuk beállítani, ha szükséges.

###### Táblázat címének formázása

A caption-side tulajdonsággal megadhatjuk, hogy fent, vagy lent jelenjen-e meg a cím. A text-align tulajdonsággal tudjuk jobbra vagy balra igazítani. Továbbá a cím szövegét tetszés szerint formázhatjuk. Alapértelmezetten a cím fent középen jelenik meg. Az Internet Explorer nem ismeri fel a caption-side tulajdonságot, ezért mindig fent jeleníti meg a címet.

###### Cellákban lévő szövegek igazítása.

Szöveget igazíthatunk függőlegesen és vízszintesen is az egyes cellákon belül a vertical‑align:top/middle/bottom utasítással.

###### Zebra táblák

Hosszú táblázatoknál előnyös lehet, ha minden második sornak más a háttérszíne. Ezt úgy érhetjük el, hogy a HTML kódban minden cimkét osztályhoz rendelünk, és minden második tr cimkét azonos névvel jelölünk meg:

<tr class="elso">

<tr class="masodik">

A CSS-ben pedig a .masodik kijelölőnek háttérszínt adunk.

**Másik megoldás**, hogy minde páros sornak adunk héttérszínt. Ennek előnye, hogy a html kódban nem kell osztályjelölőket elhelyezni. Ehhez a tr taget módosítsuk a :nth-child szelektorral, és zárójelben even/odd (páros/páratlan) szót kell megadni.

tr:nth-child(even) {

background-color: #eee;

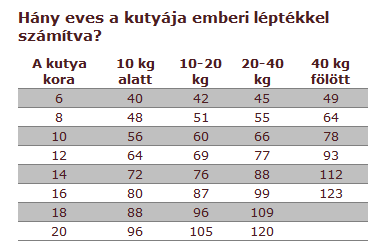
}

tr:nth-child(odd) {

background-color: #fff;

}

Formázzuk meg a táblázatunkat az alábbi módon!



###### Táblázat cellaformázásának módosítása egéresemények esetén.

A linkek készítéséhez hasonlóan a táblázat formázásakor is használhatunk szelektorokat. Az alábbi kódrészlet kiemeli azokat a sorokat, amelyik fölött áll az egérmutató.

tr:hover {background-color: #f5f5f5;}

### Reszponzív táblázatok

Széles, vagy hosszú táblázatok esetén előfordulhat, hogy mobil eszközökön nem jelenik meg a táblázat tartalama, vagy összenyomódnak a cellák, ezáltal olvashatatlanná válik a tartalom. Ezen segíthet, ha a táblázatot div tag-ek közé tesszük és overflow értékét automatikusra állítjuk. Ezzel azt érjük el,hogy ha a táblázat nem fér el a képernyőn, akkor megjelenik egy x, vagy y rányú gördítősáv.

ld. <https://www.w3schools.com/css/tryit.asp?filename=trycss_table_responsive>

<div style="overflow-x:auto;">

<table summary="A táblázat tartalmazza az elért pontszámokat">

<caption>

Elért pontszámok

</caption>

<tr>

<th >Vezetéknév</th>

<th >Keresztnév</th>

<th >Pont</th>

<th >Pont</th>

<th >Pont</th>

<th >Pont</th>

<th >Pont</th>

<th >Pont</th>

<th >Pont</th>

<th >Pont</th>

<th >Pont</th>

</tr>

<tr>

<td>Kis </td>

<td>Ibolya</td>

<td>10</td>

<td>20</td>

<td>30</td>

<td>10</td>

<td>20</td>

<td>30</td>

<td>10</td>

<td>20</td>

<td>30</td>

</tr>

<tr>

<td>Gipsz </td>

<td>Jakab</td>

<td>30</td>

<td>40</td>

<td>50</td>

<td>10</td>

<td>20</td>

<td>30</td>

<td>10</td>

<td>20</td>

<td>30</td>

</tr>

…

</table>

</div>

### Összegzés

Táblázatokat készítésénél a táblázat strukturális elemeit külön-külön cimkék jelölik. Ezek table, tr, td, th, caption. A táblázat felépítése soronként történik, majd a sorokban a cellákat balról jobbra haladva adjuk meg.

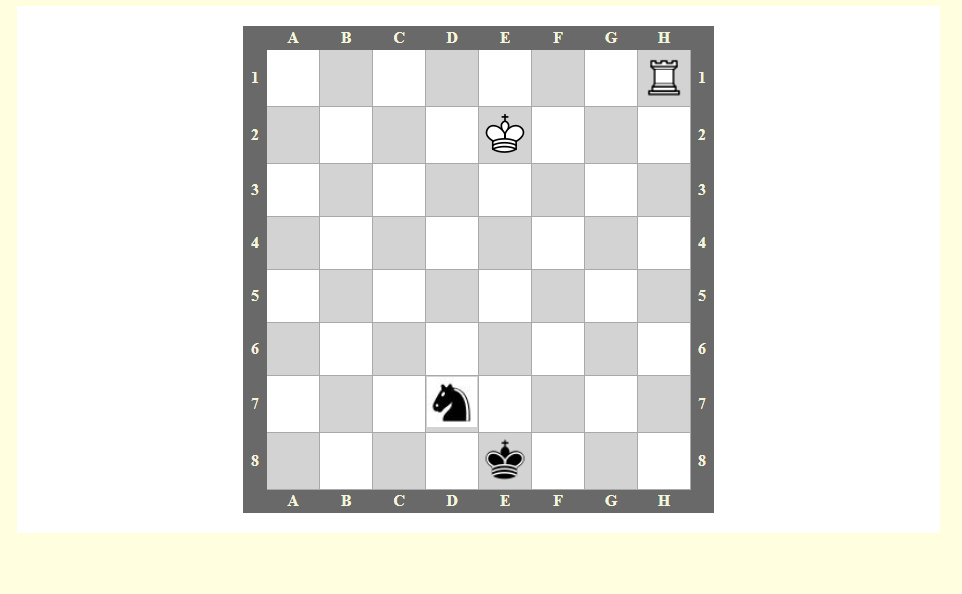
A táblázat elemeire a szokásos formázások alkalmazhatók, eltérés van azonban a cellák közötti távolság meghatározásában, ezt a table kijelölőbe kell írni a CSS-ben, a border-collapse, illetve a border-spacing tulajdonságok értékének állításával.

A táblázat celláinak egyesítését a HTML kódban kell megadnunk.

Lehetőség van a táblázatban sávos sorok, illetve sávos oszlopok kialakítására is. Az előbbit osztályhozzárendeléssel, az utóbbit a col elem használatával érjük el.

### Feladat

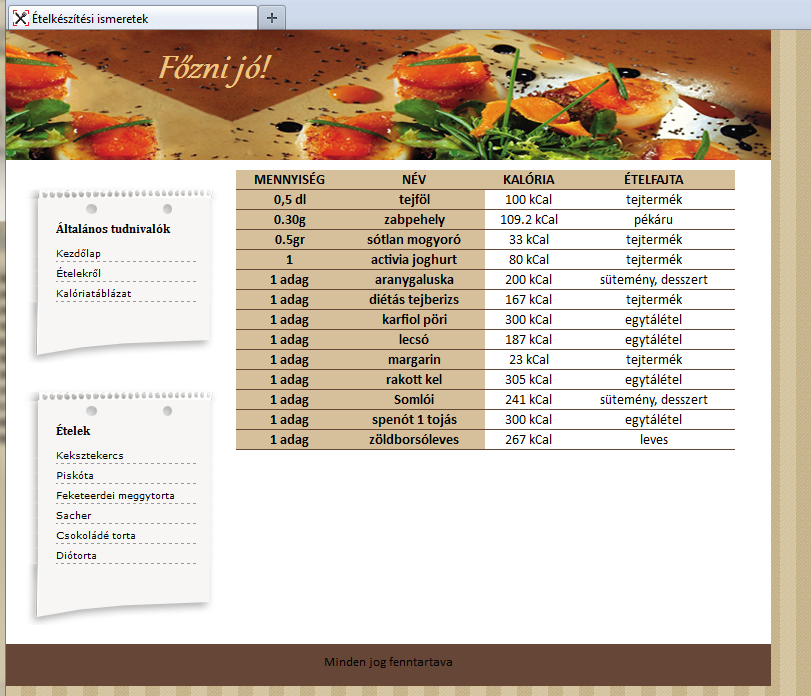
1. Gyakorlásként készítsük el az alábbi sakktáblát!



2. Készítsük el a receptkönyvünkhöz tartozó kalóriatáblázatot a mintának megfelelően!

Az első sort és az első két oszlopot th cimkével jelöljük meg! Színe: #d5c09b

A celákat vízszintes folytonos vonal válassza el egymástól, színe: #654637. A függőleges vonalak ne jelenjenek meg. A táblázat szélessége 95%.

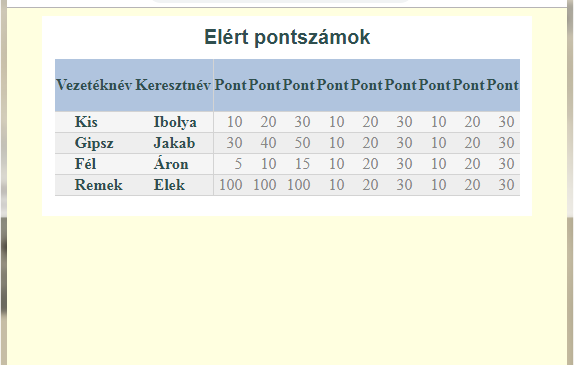
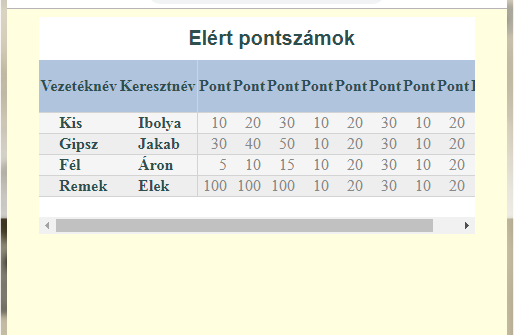


**3. feladat**

Készítsük el és formázzuk meg az alábbi táblázatot oly módon, hogy a böngésző méretezésekor is jól láthatóak6elérhetőek legyenek az adatok!

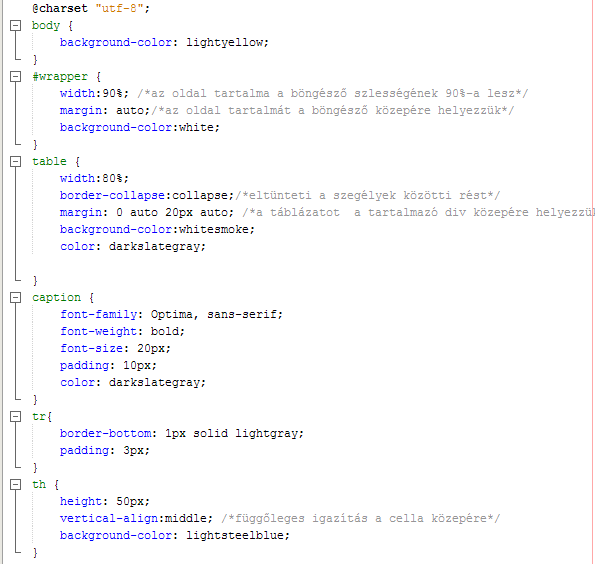
Az alábbi képernyőképeken látszanak a változások!

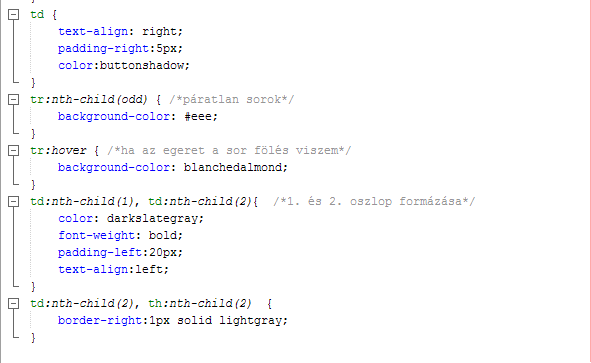


### Megoldások

**3. feladat:**





1. feladat

