

Megoldási útmutató

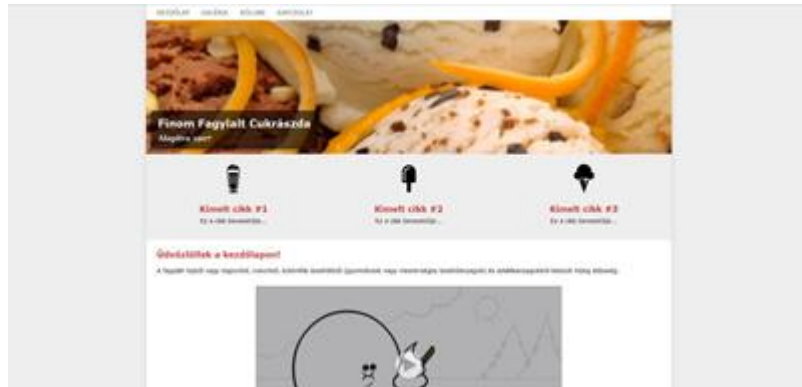
A rezponzív „finom fagyalt
cukrászda” weboldal
elkészítéséhez

A cél

Albert Zsolt mintafeladata alapján

- Egy minimalista reszponzív weboldalt készítünk el néhány lépésen keresztül
 - kódot írunk html és css fájlban
- Az elkészítendő weboldal:
 - <http://web.uni-corvinus.hu/~internet/feladat/css3/resp/index.html>
 - megjegyzés: nem minden mobil böngészőben jelenik meg tökéletesen
 - próbáld ki különböző telefonokon, különböző mobil böngészőkben
 - tipp: a böngésző méretének átállításával is szimulálni tudod a felbontás változását

Felbontásfüggő képernyőképek



1. Html kód

□ Menüikon

```

  <ul id="menu-lista">
```

□ Meta tag: közli a mobil böngészővel hogy mobilképes az oldal, és ne a teljes verziót nyomja össze:

- viewport: virtuális ablak, a mobil eszközök böngészőprogramjai az oldalakat ebben jelenítik meg

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

2. Javascript kód – html fájlba

```
<script>
//Globális változó, értékmegőrző
var nyitva_e = false;

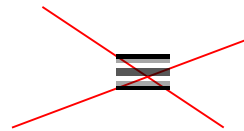
function showMenu() {
    //Ha a nincs nyitva kinyitom, ha igen, becsukom.
    if (!nyitva_e) {
        document.getElementById("menu-lista").style.display = "block";
        nyitva_e=true;
    }
    else {
        document.getElementById("menu-lista").style.display = "none";
        nyitva_e=false;
    }
}
</script>
```

3. CSS kód I.

/*Fejléc formázásai

```
#menu #menu-button {display:none;}
```

- Hogy alapból, a böngészőben ne látszódjon a menüikon



4. CSS kód II.

```
/*Dobozok formázásai*/  
article.doboz {  
    float:left;  
    width:29.33%;  
    margin:20px 2%;  
    text-align:center;  
}
```

- Ahhoz kell, hogy a 3 doboz egymás mellett legyen

5. CSS kód III.

```
/*Tartalom formázása */
```

```
article#tartalom video {  
    margin:25px auto;  
    display:block;  
}
```


6. CSS kód IV.

```
/*1100px alatti eszközökre  
megjelenés=====*/
```

```
@media (max-width:1100px) {  
    #kontener {  
        width:auto;  
    }  
  
    video {width:100%;}  
}
```

Megjegyzés: a tartományokat un. *töréspontok* rögzítésével határozzuk meg, most 1100, 640, 370 pixelnél

□ E nélkül nem nyomódik össze az oldal

7. CSS kód V.

/*640px alatti

nézet=====*/

```
@media (max-width:640px) {  
    article.doboz {  
        width:46%;  
    }  
    article.doboz:last-of-type {  
        margin-right:27%;  
        margin-left:27%;  
    }  
}
```

□ Az utolsó dobozt is középre igazítja

8. CSS kód VI.

```
/*370px alatti  
mobilnézet=====*/  
@media (max-width:370px) {  
  
    #menu #menu-button {display:block;}  
  
    #menu ul {display:none;}  
}
```

- Alapból mobilon a mobilikon látszódjon, a menü viszont ne látszódjon

9. CSS kód VII.

```
#menu ul li {  
    display:block;  
}
```

```
#menu ul li a {  
    display:block;  
    padding:10px 0;  
    text-align:center;  
    border-bottom:1px solid #ccc;  
}
```

10. CSS kód VIII.

```
article.doboz {  
    width:96%;  
}
```

A dobozok egymás alá kerülnek

```
article.doboz:last-of-type {  
    margin-right:2%;  
    margin-left:2%;  
}
```

Az utolsó doboz sem csúszik szét

Megjegyzés1: még néhány apró módosítás kell, a végeredmény a tantárgyi weboldalon

Fontos megjegyzés

- Ennek a „kézzel írt” mintafeladatnak vannak hiányosságai.
- Egyszerű szerkezetű és tartalmú weboldalnál kielégítő megoldás lehet
- Összetett site esetében, ha teljesen profi reszponzív weboldalt akarunk írni, akkor ajánlott egy keretrendszer vagy CMS,
 - például Bootstrap(<http://getbootstrap.com/>)
 - vagy Wordpress (<https://wordpress.org/>) használata
- Azonban ekkor is fontos, hogy értsük a reszponzív megoldás lényegét, ezért fontos ezt a mintafeladatot megérteni, átlátni