**A viewport metaelem**

Elkészítettük életünk első responsive oldalát és megnézzük mobil eszközön. Érdekes módon az oldal a nagyobb kijelzőkre beállított szerkezettel jelenik meg, nem pedig az erre a méretre beállított szerkezettel. Ezt a jelenséget a nézetablak meta tag hiánya okozza. Nem állítottuk be, hogy hogyan jelenjen meg az oldalunk a kisebb kijelzővel rendelkező eszközökön. Nincs más dolgunk, mint a következő sort hozzáadni a HTML dokumentumunk fejlécébe.

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

A fent bemutatott jelenséget, a már említett, a mobil böngészőkben deklarált alap viewport beállítás okozza, vagyis hogy bepréselik a megjelenítendő oldalt a kijelző keretei köze.

Tehát viewport (nézetablak) meta elem segítségével felülírhatjuk a böngésző (user agent) által deklarált nézetablakot.

**A viewport megértéshez szükségünk van további két fogalomra:**

* **initial viewport (kezdeti nézetablak)** – azt a viewport-ot jelenti, amit még nem írt felül a UA (user agent), vagy felhasználói stílus
* **actual viewport (tényleges nézetablak)** – az a viewport, ami az értékek feldolgozása után jön létre.

**Eszköz szélesség – width**

* A width tulajdonsággal értelemszerűen a viewport terület szélességét szabályozhatjuk. Például, ha a telefonra készített oldalunk szélessége 320px, akkor a következő meta-val igazíthatjuk a nézetablak területét az oldalunkhoz.

<meta name="viewport" content="width=320" />

* Ha több felbontású eszközre készítettük az oldalunkat, akkor választhatjuk a dinamikus érték megadást, az eszköz fizikai kijelzőmérete alapján:

<meta name="viewport" content="width=device-width" />

**Eszköz magasság – height**

A width-hez hasonlóan természetesen itt is van height tulajdonságunk, amit azonban kevésbé használunk, mivel a napjainkban készített oldalaknál a vertikális irány (le/fel görgetünk) a meghatározó, így az esetek többségében az oldalainknál csak a szélességet kell megadnunk.

<meta name="viewport" content="width=device-height" />

**Initial-scale tulajdonság**

A megfelelő megjelenítés érdekében tudjuk manipulálni a nagyítás mértékét.

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

Az initial-scale tulajdonság segítségével megadhatunk min és max értéket egyaránt:

<meta name="viewport" content="width=device-width, initial-scale=1.0, minimum-scale=1.0, maximum-scale=3.0" />

A user-scalable tulajdonságát “no” értékre állítva **letilthatjuk a felhasználói zoomolást**. Ilyenkor célszerű a minimum/maximum scale tulajdonságot az initial-scale tulajdonsággal azonosra állítani. Valószínűleg nem túl szerencsés, ha a felhasználótól elvesszük a nagyítás lehetőségét. Figyeljünk rá, hogy csak nagyon indokolt esetekben használjuk!

<meta name="viewport" content="width=device-width, initial-scale=1.0, minimum-scale=1.0, user-scalable=no">

## **@viewport CSS szabály**

A viewport-ot CSS-ben ugyanazokkal a tulajdonságokkal használhatjuk, mint HTML-ben a meta tag-et.

Főbb eltérés, hogy az initial-scale tulajdonság helyett itt zoom-ot kell használnunk.

@viewport{

zoom: 1.0;

width: device-width;

min-zoom: 1;

max-zoom: 3;

zoom: fixed;

}

Ha szeretnénk letiltani a felhasználói közelítést, akkor a zoom tulajdonságnak a **fixed** értéket kell megadnunk.

**@viewport média lekérdezéssel**

Azzal, hogy a viewport-ot a CSS-ben kezeljük lesz egy előnyünk. Egészen egyszerűen állíthatunk be Media Query segítségével különböző kijelző felbontásokhoz, más és más viewport értéket.@media screen and (min-width: 640px) and (max-width: 1024px) {

@viewport { width: 640px; }

...

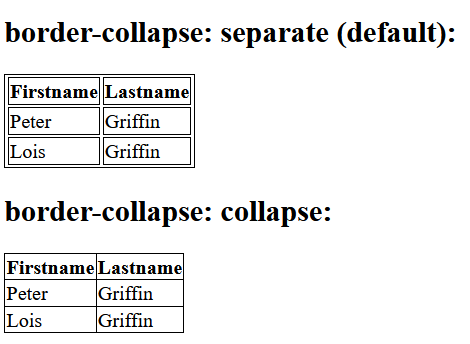
}

**CSS**

**border-collapse**

A border-collapse tulajdonság segítségével a **celláink körüli szegély** megjelenítését befolyásolhatjuk.

[**https://www.w3schools.com/cssref/pr\_border-collapse.asp**](https://www.w3schools.com/cssref/pr_border-collapse.asp)

****

**nth-child**

Az: n-dik gyermek (n) választó megfelel minden elemnek, amely szülőjének n-edik gyermeke, típusától függetlenül.

n lehet szám, kulcsszó vagy képlet.

[**https://www.w3schools.com/cssref/sel\_nth-child.asp**](https://www.w3schools.com/cssref/sel_nth-child.asp)

**media query**

[**https://www.w3schools.com/css/css3\_mediaqueries\_ex.asp**](https://www.w3schools.com/css/css3_mediaqueries_ex.asp)

**colors**

[**https://www.w3schools.com/cssref/css\_colors.asp**](https://www.w3schools.com/cssref/css_colors.asp)

[**https://www.w3schools.com/colors/default.asp**](https://www.w3schools.com/colors/default.asp)

**opacity**

[**https://www.w3schools.com/css/css\_image\_transparency.asp**](https://www.w3schools.com/css/css_image_transparency.asp)

**gradients**

[**https://www.w3schools.com/css/css3\_gradients.asp**](https://www.w3schools.com/css/css3_gradients.asp)

**box-shadow**

[**https://www.w3schools.com/cssref/css3\_pr\_box-shadow.asp**](https://www.w3schools.com/cssref/css3_pr_box-shadow.asp)

**outline**

[**https://www.w3schools.com/css/css\_outline.asp**](https://www.w3schools.com/css/css_outline.asp)

**border-radius**

[**https://www.w3schools.com/cssref/css3\_pr\_border-radius.asp**](https://www.w3schools.com/cssref/css3_pr_border-radius.asp)

**text-shadow**

[**https://www.w3schools.com/cssref/css3\_pr\_text-shadow.asp**](https://www.w3schools.com/cssref/css3_pr_text-shadow.asp)