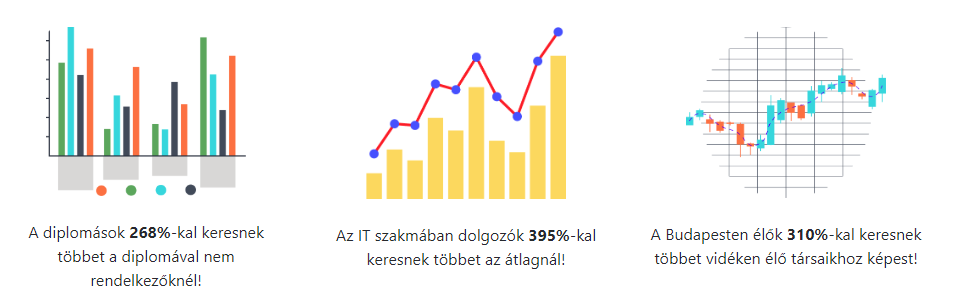
# Fizu.hu 2.

Ebben a leckében folytatjuk az előzőben elkezdett weblapot a Bootstrap használatával.

A fejléc alá egy három oszlopos részt fogunk készíteni:  


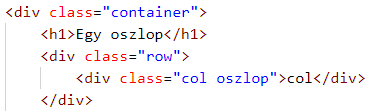
A folytatás előtt azonban meg kell ismerkednünk a Bootstrap grid rendszerével. Nyisd meg a kapott *grid* mappát és benne a *grid.html* fájlt a Visual Studio Code-ban!

# Grid rendszer

A weblapokon gyakran egymás mellé szeretnénk helyezni szakaszokat. Erre szolgál a Bootstrap-ben a grid (rács).

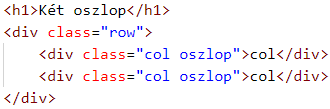
A szakaszokat egy sorba (row osztály) kell tenni, és a col osztály valamelyik változatával jelezni a szakasz szélességét. (Az oszlop osztály csak a háttérszínt és a szegélyt állítja be.)

Nézzük az első példát:



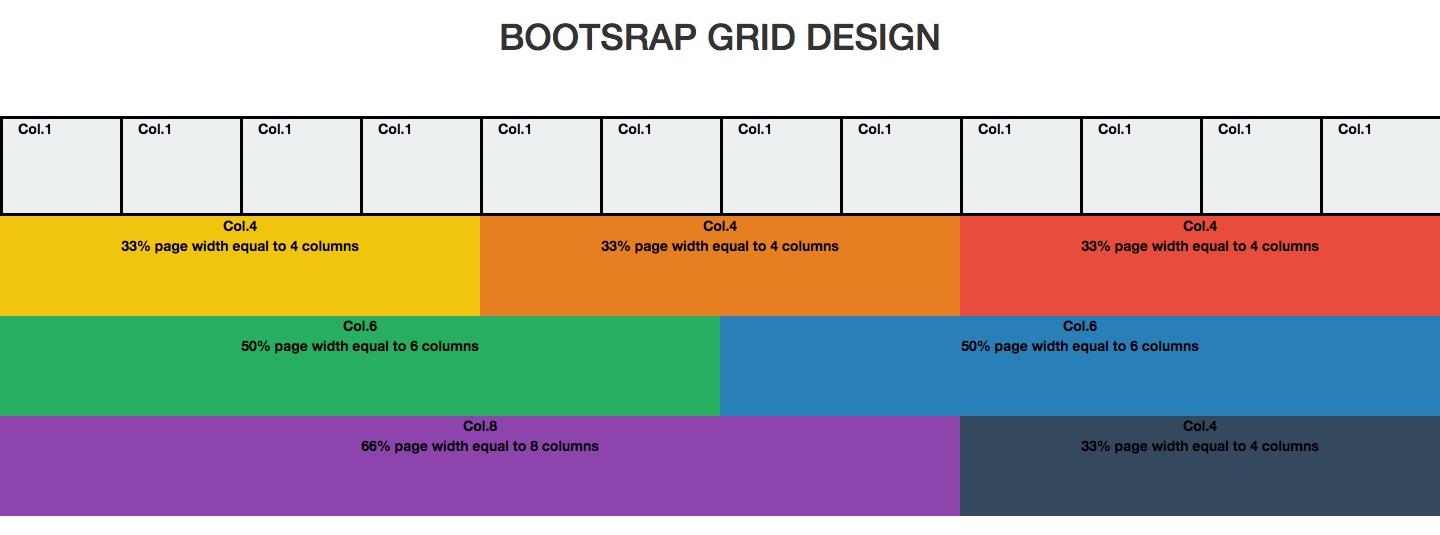
Ebben az esetben egy oszlop fog megjelenni minden szélességnél.

A következő példában két ilyen szakaszt teszünk egymás után. Ezek mindig egymás mellett fognak megjelenni.

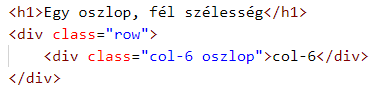


A col osztálynak vannak olyan változatai, ahol a col után kötőjellel egy 1 és 12 közötti számot adunk meg (például col-6).

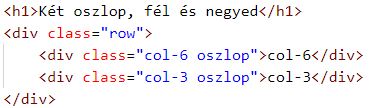
A Bootstrap a képernyő teljes szélességét 12 részre osztja fel. Ez a szám azt jelzi, hogy a szakasz hány részt foglaljon el.



A következő példában a szakasz az első hat részt foglalja el (mellette nem lesz semmi):



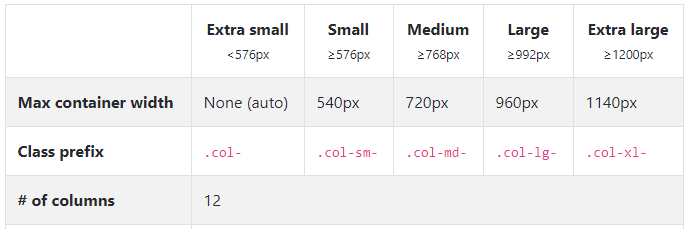
Ezután megadhatunk további szakaszokat is. Mennyit foglal el a következő példában a második szakasz?



Ami nem fér ki egy sorba, az a következő sorba kerül.

Az eddigiek minden kijelzőméretre vonatkoznak, de a Bootstrapben a különböző kijelzőkhöz különböző szélességeket is megadhatunk.

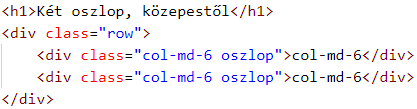
A lehetséges kijelzőméretek:



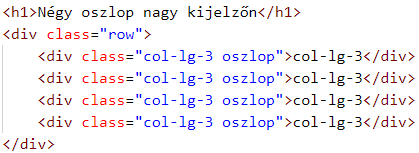
A beállításoknál a kijelzőméretet rövidítve adjuk meg (pl. -md-), és a megadott mérettől felfelé is érvényes, amíg nem adunk meg mást.

Például a "col-md-3" 768 pixeltől felfelé adja meg, hogy három részt foglaljon el a szakasz. Az extra smalll kijelzőt nem jelezzük. Így eddig erre adtuk meg a szélességeket, vagyis az összes kijelzőre érvényesek voltak.

Mit jelent a "col-md-6" osztály? Próbáld ki a következő példában különböző szélességeknél!



A következő példában mekkora kijelzőn lesz négy oszlopos a megjelenítés?



Ugyanarra a szakaszra több szélességet is megadhatunk. Például a   
"col-md-4 col-lg-3" osztály 768 px alatt 12 résznyi szélességet ad meg, 768‑tól 992 px-ig 4 résznyit, a fölött pedig 3 résznyit. Vagyis telefonon egy oszlop lesz, tableten 3, laptopon és asztali gépen 4.

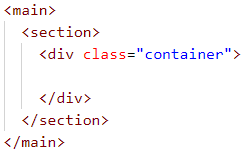
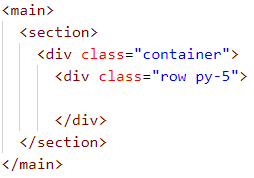
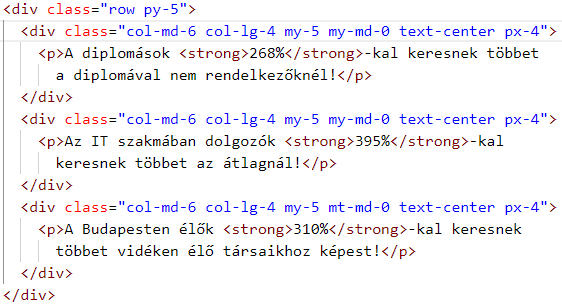
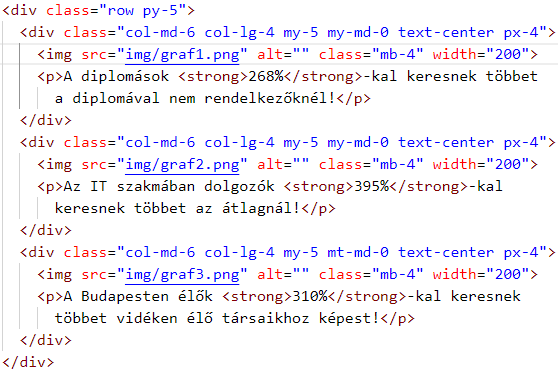
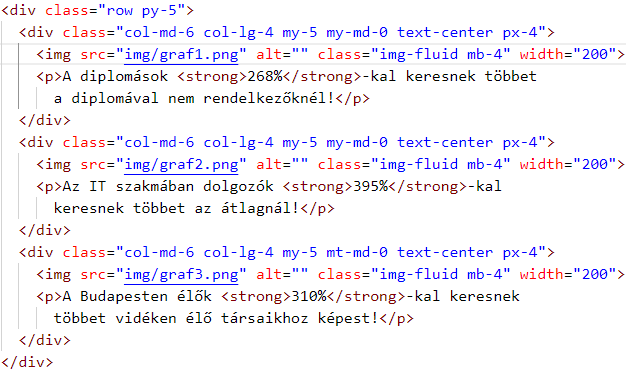
Írd át ilyenre az előző példát, és próbáld ki!

A kijelzőméreteket nem csak az oszlopok szélességénél lehet alkalmazni, hanem például a margónál és a paddingnél is.

Például "my-5 my-md-0" azt jelenti, hogy a közepes (768px alatti) szélességeknél alul és felül 5 egységnyi margó lesz, nagyobb kijelzőknél viszont 0 (vagyis nem lesz).

# Alkalmazás

Ezután alkalmazzuk a tanultakat a fizu.hu oldalon:

1. Nyisd meg a *fizu* mappát és abból az *index.html* fájlt a Visual Studio Code-ban!
2. Nézd meg az oldalt a böngészőben!
3. Nyisd meg a <https://getbootstrap.com/> oldalt a böngészőben, és a menüből válaszd a Documentation pontot!
4. Az *index.html* fájlban a jumbotron után készíts egy main szakaszt, azon belül egy section-t, és abban helyezz el egy containert!  
   
5. A containerben helyezz el egy sort, alul és felül 5 egységnyi margóval!  
   
6. Készíts a sorban három szakaszt az alábbi beállításokkal és szöveggel:  
     
   Próbáld ki különböző ablakszélességekkel!
7. Szúrj be képeket mind a három szakaszba a szöveg elé:  
     
   Próbáld ki különböző ablakszélességekkel! Figyeld meg a képek méretének változását!
8. A Bootstrapben a képeket az img-fluid osztállyal lehet reszponzívvá tenni. Alkalmazd ezt a három képre, és hasonlítsd össze az előző változattal!  
   

A következő leckében folytatjuk.