# Flexbox 3.

## Bevezetés

Az előző leckékben a flex elemek egy sorban vagy oszlopban voltak. Most megnézzük, hogyan lehet több sorba tördelni őket.

## Előkészületek

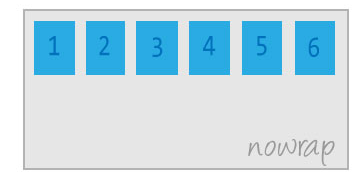
1. Nyisd meg a *dobozok* mappát, majd abból a *doboz.html és a doboz.css* fájlokat a Visual Studio Code-ban!
2. Nyisd meg az oldalt a böngészőben is!
3. Nézd át az eddig elkészített HTML és CSS kódokat!

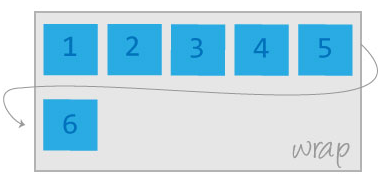
## Sortörés bekapcsolása

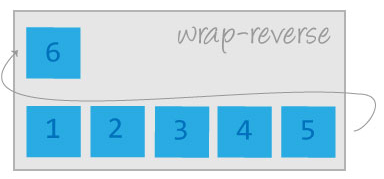
A sortörést a flex-wrap tulajdonsággal vagy a flex-flow tulajdonság második értékével állíthatjuk be.

A lehetséges értékek:

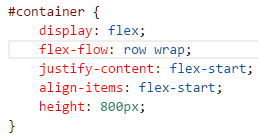
* nowrap: nincs sortörés (alapértelmezett),
* wrap: sortörés,
* wrap-reverse: fordított sortörés.





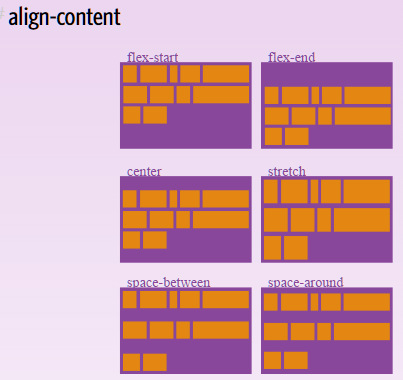


Bekapcsolt sortörés esetén a flex-shrink tulajdonságnak nincs jelentősége.

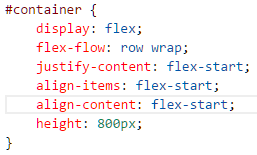
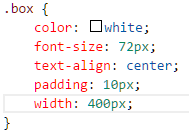
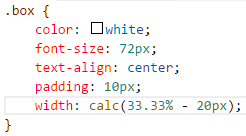
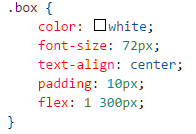
1. Kapcsold be a sortörést a tároló elemben!  
   
2. Figyeld meg, hogy keskeny ablaknál több sorba kerülnek a dobozok, függőlegesen viszont nem közvetlenül egymás alá kerülnek!

## A tartalom igazítása

Több sorba tördelés esetén az align-content tulajdonsággal adhatjuk meg, hogyan helyezkedjenek el a sorok (vagy az oszlopok).

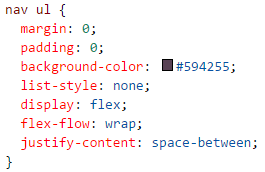
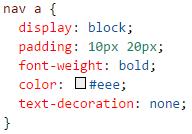
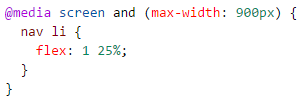
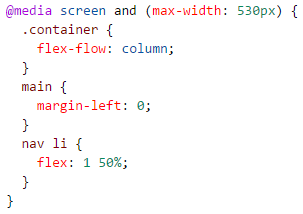


Alaphelyzetben a sorok a stretch beállítás szerint jelennek meg. A mi oldalunkon ez nem egészen olyan, mint az ábrán, mert az align-items tulajdonság nem stretch-re van beállítva. Ilyenkor az üres helyet egyenlően osztja fel a sorok között.

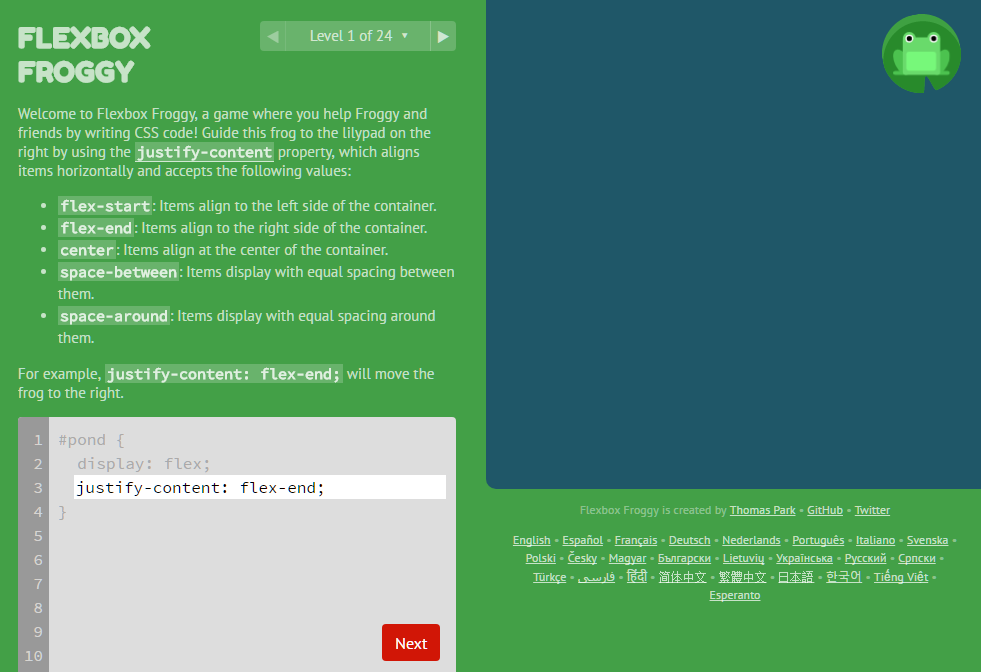
1. Állítsd be, hogy a sorok a tároló elem tetején jelenjenek meg!  
   
2. Próbáld ki a többi lehetőséget is, majd állítsd vissza a flex‑start értéket!
3. Állítsd a dobozok szélességét 400 képpontra!  
     
   Próbáld ki különböző ablakszélességekkel!
4. Állíts be 400 képpont helyett (33,33% - 20 képpont) szélességet a dobozokra!  
     
   Próbáld ki különböző szélességű ablakokkal!
5. Most a szélesség *helyett* állítsd be a dobozok flex tulajdonságát a következőképpen:  
     
   Próbáld ki különböző szélességű ablakokkal! Mit tapasztalsz? Mindig egyforma szélesek a dobozok?

## Navigáció készítése

Ebben a példában egy egyszerű reszponzív navigációt készítünk a flexbox segítségével.

1. Nyisd meg a pelda mappát a Visual Studio Code-ban! Nézd meg a fájlokat!
2. Ez a múltkori példa folytatása. A HTML fájlt kiegészítettük egy navigációs résszel. Nézd meg! A lista (ul) lesz a flex tároló, a listaelemek (li) pedig a flex elemek.
3. Készíts a CSS fájlban egy nav ul stílust:  
     
   Értelmezd a beállításokat!
4. Utána készítsd el a nav a stílust a következőképpen:  
     
   Értelmezd a beállításokat! Próbáld ki!
5. Rámutatáskor a hivatkozások változzanak aranyszínűre!  
     
   Próbáld ki különböző ablakszélességekkel!
6. 900 képpontos szélesség alatt az utolsó menüpont nem fér ki, csúnya lesz a menü. Ilyenkor szeretnénk két darab négy elemű menüre bontani:  
     
   Próbáld ki!
7. 530 képpontos szélesség alatt már ez sem elég, itt két oszlopos menüt szeretnénk:  
     
   Próbáld ki!

## Példa

1. Nyisd meg a <https://flexboxfroggy.com/> oldalt a böngészőben!
2. A játékban a békákat kell elhelyezni a leveleken. A víz a container, a békák a flex elemek. Játszd végig az 13-24 pályákat!
3. Segítség: a flex-wrap tulajdonság helyett használhatod a flex-flow tulajdonságot.
4. 

## Önálló feladatok 2.

Töltsd fel egy online tárolóhelyre ezt a feladatsort és az órán készített fájlokat! Otthon is nézd át őket!