



# Lambda kifejezések

Répásné Babucs Hajnalka



# A törpék adatai



```
név;klán;nem;súly;magasság
Terrin Sak;Kova;F;62;130
Bal Eraffa;Vasököl;N;45;129
Sarsi Duri;Vasököl;F;59;141
Dorf Gein;Csille;F;62;138
Bombur Nori;Kova;N;48;132
Dorf Loopa;Csille;F;63;140
Trad Magraimn;Acél;F;67;138
Tardi Falgorran;Csille;F;65;140
Lok Dun;Acél;F;70;142
Sarrof Hro;Kova;F;65;137
Gor Morf;Csille;F;58;135
Heimal Morf;Vasököl;F;61;144
Odi Duri;Kova;F;64;145
```



- Az összes törpe adatát egy *torpek* nevű listában tároljuk.
- A törpék neme lehet férfi vagy nő.
- Néhány feladat elvégzéséhez szükség van a törpék testtömegindexének kiszámítására, ez számított tulajdonság lesz.



# Egyszerű értéket visszaadó metódusok 1.



➤ Hány törpe adatai találhatóak meg a listában?

➤ `torpek.Count`

➤ `torpek.Count()`



➤ Ha egymás fejére állnának a törpék, akkor milyen magasak lennének együtt?

➤ `torpek.Sum(x=>x.Magassag)`



➤ Mekkora a törpék átlagos súlya?

➤ `torpek.Average(x=>x.Suly)`

➤ Üres listára hibát dob!





## Egyszerű értéket visszaadó metódusok 2.



➤ Mekkora a legmagasabb (legalacsonyabb) törpe?

➤ `torpek.Max(x=>x.Magassag)`

➤ `torpek.Min(x=>x.Magassag)`

➤ Üres listára hibát dob!

➤ Hány női törpe adatai találhatóak meg a listában?

➤ `torpek.Count(x=>x.Nem=='N')`





## Egyszerű értéket visszaadó metódusok 3.



➤ Van valamilyen elem a torpek listában?

➤ `torpe.Any()`

➤ Van-e Kova klánba tartozó törpe?

➤ `torpek.Exists(x=>x.Klan=="Kova")`

➤ `torpek.Any(x=>x.Klan=="Kova")`

➤ Minden törpe férfi?

➤ `torpek.All(x=>x.Nem=='F')`





# Összetett adatot visszaadó metódusok 1.



## ➤ Ki az első (utolsó) eltárolt törpe?

➤ `torpek.First()`

➤ `torpek.Last()`

➤ Így már lekérdezhető az első törpe összes adata részenként. Pl.:  
`torpek.First().Nev`

➤ Üres listára hibát dob!

➤ Lehet feltételt is megadni.

## ➤ Mire lehet ez jó még?

➤ Ha valamilyen szempont alapján rendezettek az adatok, könnyű kiválasztani a legjobban teljesítőt!

➤ Nem hatékony, de helyettesíthető vele a maximum kiválasztás tétele

(Hatékonyabb megoldás: `Max` metódussal érték, `IndexOf` metódussal hely meghatározás, majd indexelés)







## Összetett adatot visszaadó metódusok 2.



- Add meg a legmagasabb (legalacsonyabb) törpét!
  - `torpek.MaxBy(x=>x.Magassag)`
  - `torpek.MinBy(x=>x.Magassag)`
  - Torpe típussal tér vissza, ha a törpe nevére vagyunk kíváncsiak, akkor:  
`torpek.MaxBy(x=>x.Magassag).Nev`





# Összetett adatot visszaadó metódusok 3.



## ➤ Add meg az első 10 cm alatti törpét!

➤ `torpek.FirstOrDefault(x=>x.Magassag<10)`

➤ Ha lenne ilyen törpe, akkor megadná az első ilyen értéket. Mivel nincs, ezért a típus default értékét adja vissza, ami referencia típusoknál null érték.

➤ `torpek.Find(x=>x.Magassag<10)`

➤ Ha lenne ilyen törpe, akkor megadná az első ilyen értéket. Mivel nincs, ezért null értékkel tér vissza, típustól függetlenül

➤ Lásd még: nullable típusok használata

## ➤ Add meg az utolsó 10 cm alatti törpét!

➤ `torpek.LastOrDefault(x=>x.Magassag<10)`

➤ `torpek.FindLast(x=>x.Magassag<10)`







# Sorozatot visszaadó metódusok 1.



## ➤ Kik a nőnemű törpék?

- `torpek.Where(x=>x.Nem=='N')`
- `torpek.FindAll(x=>x.Nem=='N')`
- Torpe típusú adatok sorozatát adja vissza



## ➤ Milyen klánokból érkeztek törpék?

- `torpek.Select(x=>x.Klan)`
- String típusú adatok sorozatát adja vissza



## ➤ Milyen klánokból érkeztek törpék? (Minden név csak egyszer szerepeljen!)

- `torpek.Select(x=>x.Klan).Distinct()`



## ➤ Adjuk meg minden klánból a listában szereplő első törpét!

- `torpek.DistinctBy(x=>x.Klan).Select(x=>($"{x.Nev} ({x.Klan}))")`





## Sorozatot visszaadó metódusok 2.



- Ki az első 3 törpe, akik bekerültek a listába?
  - `torpek.Take(3)`
  - Ha csak 2 elem van a listában, akkor 2 elemet ad vissza.
- Add meg a lista elejéről a hímnemű törpéket az első nőnemű törpéig!
  - `torpek.TakeWhile(x=>x.Nem=='N')`
- Hagyd ki a lista elejéről azokat a hímnemű törpéket, akik az első nőnemű törpe előtt vannak!
  - `torpek.SkipWhile(x=>x.Nem=='N')`





## Sorozatot visszaadó metódusok 3.



- Rendezzük a törpéket névsorrendbe!

- `torpek.OrderBy(x=>x.Nev)`

- Rendezzük a törpéket TTI szerint csökkenő sorrendbe!

- `torpek.OrderByDescending(x=>x.TTI)`

- Rendezzük a törpéket elsődlegesen a klán neve, azon belül pedig név szerint!

- `torpek.OrderBy(x=>x.Klan).ThenBy(x=>x.Nev)`

- Ha a típuson meg lenne adva az összehasonlítás (`CompareTo`) művelet, akkor növekvő rendezésnél használható lenne az `Order()` metódus is.

