# Diák adatok szűrése és keresése

A feladatban a szűrést és keresést megvalósító backend alapján megvalósítjuk a diák adatok szűrését a diák születésnapjára és a diákok keresését név alapján!

## Szűrés (filering)

A szűrés egy olyan mechanizmus, amellyel az eredmények valamilyen kritérium megadásával nyerhetők ki. Sokféle szűrőt írhatunk, hogy az eredményeket az osztály tulajdonságának típusa, értéktartomány, dátumtartomány vagy bármi más alapján kapjuk meg.

A szűrés megvalósításakor mindig korlátozza a kérésben megadható előre meghatározott opciók készlete. Például küldhetünk dátumértéket a tulajdonosi fiók típusának lekérdezéséhez, de nem sok sikert fogunk elérni.

A frontend oldalon a szűrést általában jelölőnégyzetek, rádiógombok vagy legördülő gombok formájában valósítják meg

## Mi az a keresés? (searching)

Kétségtelen, hogy szinte minden internetes weboldalon látott már keresőmezőt.

Könnyű megtalálni valamit, ha ismerjük a weboldal szerkezetét, vagy ha egy weboldal nem olyan nagy. De ha a számunkra legmegfelelőbb témát szeretnénk megtalálni, vagy ha nem tudjuk, hogy mit fogunk találni, esetleg először látogatunk meg egy nagyméretű weboldalt, akkor valószínűleg keresőmezőt fogunk használni.

Az alapvető keresést nem olyan nehéz megvalósítani, de ha félvállról vesszük, a keresési funkció használhatatlan lehet.

Egyszerű projektünkben a keresés egyik felhasználási esete az lenne, hogy egy diákot neve alapján keressünk meg.

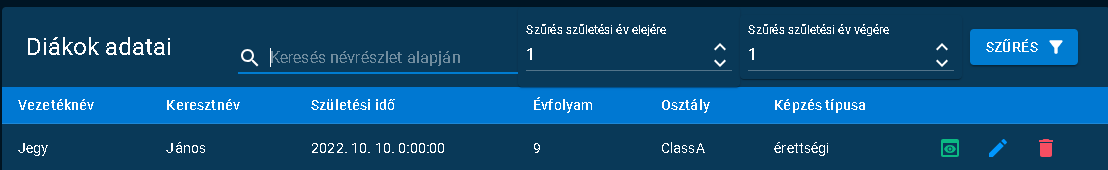
* A Kreta.Web továbbfejlesztése előtt a projektbe helyezzük el a szűrést és keresést tartalmazó backendet
* A szűrés és keresés során a Kreta.Shared projekten is változtattunk, ezeket a változtatásokat tartalmazó Kreta.Shared projektet is helyezzük el a projektbe

## Szűrés és keresés megvalósítása a razor oldalon

Először egyeztessük a Kreta.Web portszámát a backenden engedélyezett portszámmal, hogy az applikáció elinduljunk. Érdemes a saját fejlesztés során is a portszámokat egységesíteni. Ha kell a Kreta.Web alkalmazás Extension mappájába a http kliens portszámát is pontosítsuk a backend portszámával.

### ToolBarContent

Készítünk a táblázat fölé egy fejlécet amelyben szerepel a cím, a keresés és szűrés lehetőség:



### Student oldal és StudentViewModel

Készítsük el a StudentViewModel-t hogy a weboldalon megjelenő adatokat kezelni tudjuk. A fejléchez három adat kell, a felhasználó által beírt névrészlet, illetve a keresett születési idő minimuma és maximuma.

public class StudentViewModel

{

public string SerchedName { get; set; } = string.Empty;

public uint SearchedMinYar { get; set; } = 0;

public uint SearchedMaxYar { get;set; } = uint.MaxValue;

}

### Rétegek a razor oldal alatt

Razor oldal -> ViewModle ->Service (a backend elérés)

* A IStudentViewModel-t injektáljuk az oldalon.
* A IStudentService-t a StudentViewModel osztályban
* Az adatok betöltése átkerül a StudentViewModel rétegbe a razor oldalról. Az adatok megjelennek.

## Keresés és szűrés megvalósítása

* MudNumericField vezérlőknek lehet min és max érékük
* A razor oldalon a StudentViewModel metódusait beköthetjük
* Az API kéréshez StudentQueryParameters –be konvertáljuk a StudentViewModel-en lévő kereső és szűrő adatokat a StudentViewModelExtension,cs fájlban
* A tábla eddig a ReloadDataAsync metóduson keresztül kapta az adatokat. Átalakítjuk a kódokat úgy, hogy a StudnetViewModelben lévő List<Student> StudentItems property-ben legyen a tábla adatai. A szűrés és keresés eredményét is ebbe a listába kell majd elhelyezni.

## Frontend és backend

Frontend és backend oldalon is egyeztetni kell a kódokat. A StudentService osztályból küldjük a keresés és szűréssel kapcsolatos kérést. Ilyenkor StudentQueryParametersDto osztályt használunk. Itt is meg kell valósítani a két osztály közötti átalakítást. A kérést végül egy Post kérésként küldjük el. A backend oldalon a repository rétegtől List<Student>-et kapunk, ezt Dto osztállyá kell alakítani és azt küldjük el. A StudentService osztályban a frontenden a kérés elküldése után visszakapjuk az adatokat. Ezeket is átalakítjuk List<Studnet>-re.

Szűrés évszámra és „e” betűre:

