

Query's in REST API verder uitwerken

Oefening 1

- Breidt de code van je Gamestore oefenproject verder uit met de nodige LINQ query's.
- Voorzie ook testdata (via seeding) om dit te kunnen testen
- Opvragen van personen:
 - Hier moeten we (optioneel) kunnen filteren op LastName, Email en/of Gender
 - Dus ik wens bv. alle **heren** met naam **Janssens**
 - Of enkel alle **dames**
 - Je zou ook met een **wildcard** kunnen werken (bv. **email adres** eindigend op **@ap.be**)
 - ...
 - Bijkomend willen we eveneens graag kunnen **sorteren** op **LastName** of **BirthDate** (Voorzie dit mbv. een Enum parameter) **oplopend**
- Opvragen van 1 persoon adhv. De ID
 - Zorg dat je hier bijkomend ook het **Store** object teruggeeft waarin deze persoon werkt.
- Opvragen van stores
 - We wensen optioneel te kunnen filteren op **Name** van de store
 - Sorteren op **Name** ofwel **oplopend** of **aflopend**
 - Met **paging** te kunnen werken (dus we geven PageSize en PageLength mee)
 - Stel de default pagelength (indien dus niet opgegeven door de client in op 10)
 - Beperkt de maximale pagelength op 20 (zodat een client nooit ineens heel veel data uit de API kan halen de API onnodig zou belasten door bv. 10000 stores op te vragen in 1 keer)
- Opvragen van Stores (uitbreiding)
 - Gezien we met paging werken is het handig als de aanroepende client ook meer info mee terug krijgt in de response dan enkel de lijst van stores.
 - Voorzie hiervoor een klasse:

```
public class PagedStoreResult
{
    // 0 references
    public List<Store> Stores { get; set; }
    // 0 references
    public int PageNumber { get; set; }
    // 0 references
    public int PageSize { get; set; }
    // 0 references
    public int TotalNumberOfPages { get; set; }
    // 0 references
    public int TotalRecordCount { get; set; }
    // 0 references
    public int FilteredRecordCount { get; set; }
}
```

We gaan vervolgens een object van dit type teruggeven bij het opvragen van stores. Zorg ervoor dat alle properties uiteraard correct worden ingevuld zodat de client naast de lijst van stores op het scherm ook kan aangeven hoeveel pagina's er in totaal zijn en hoeveel stores.

- Stel dat we voor personen gelijkaardig willen gaan werken, moeten we dan een nieuwe klasse maken (en ook voor alle bijkomende entiteiten) of kunnen we optimaler te werk gaan? Pas eventueel je idee reeds toe op de store.

