

Entity Framework Core

#tips

# Pagination

## LINQ Extension

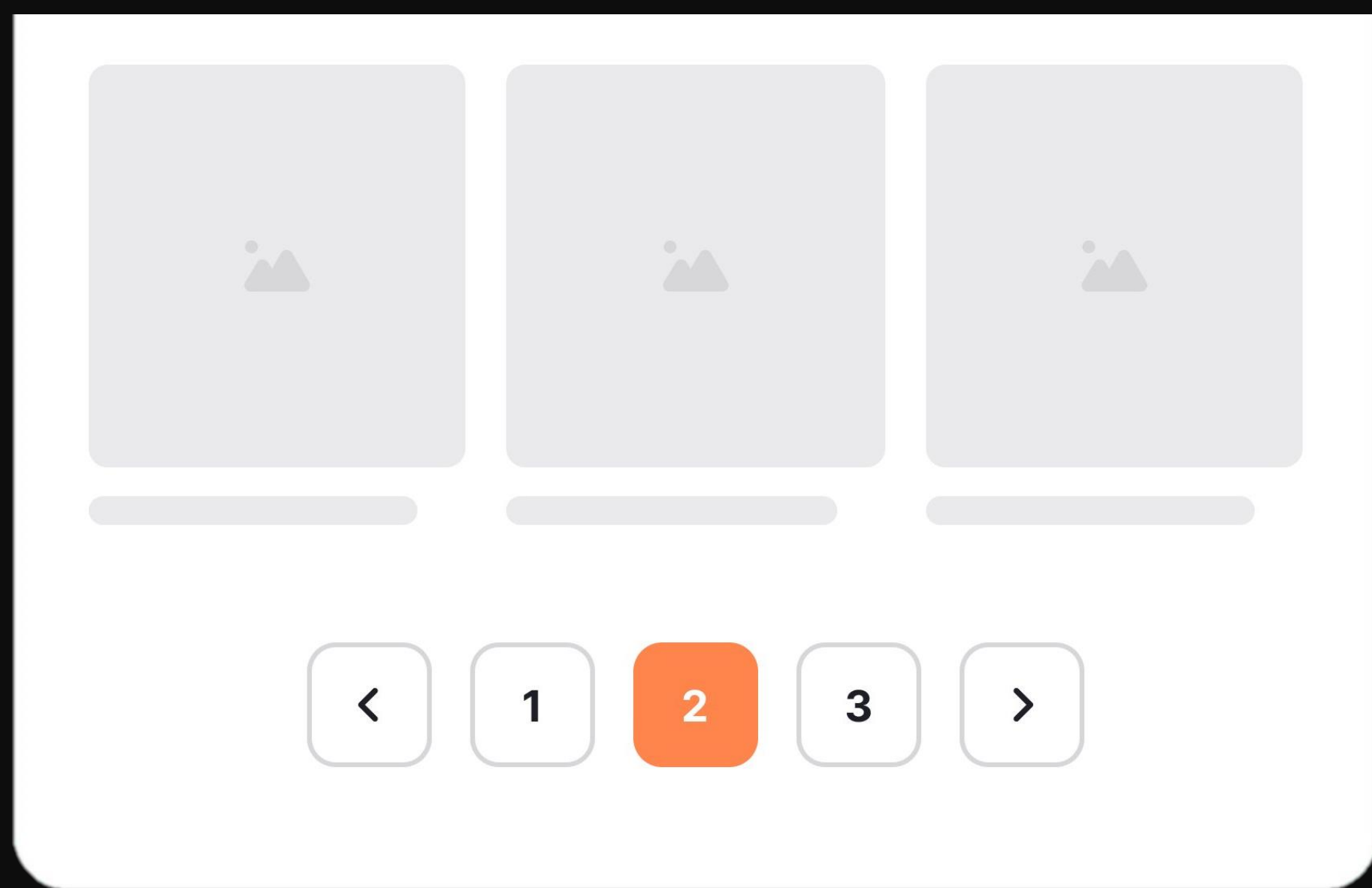


**Nabi Karampour**  
@thisisnabi



# Pagination

It refers to retrieving results in pages, this is typically done for **large resultsets**.



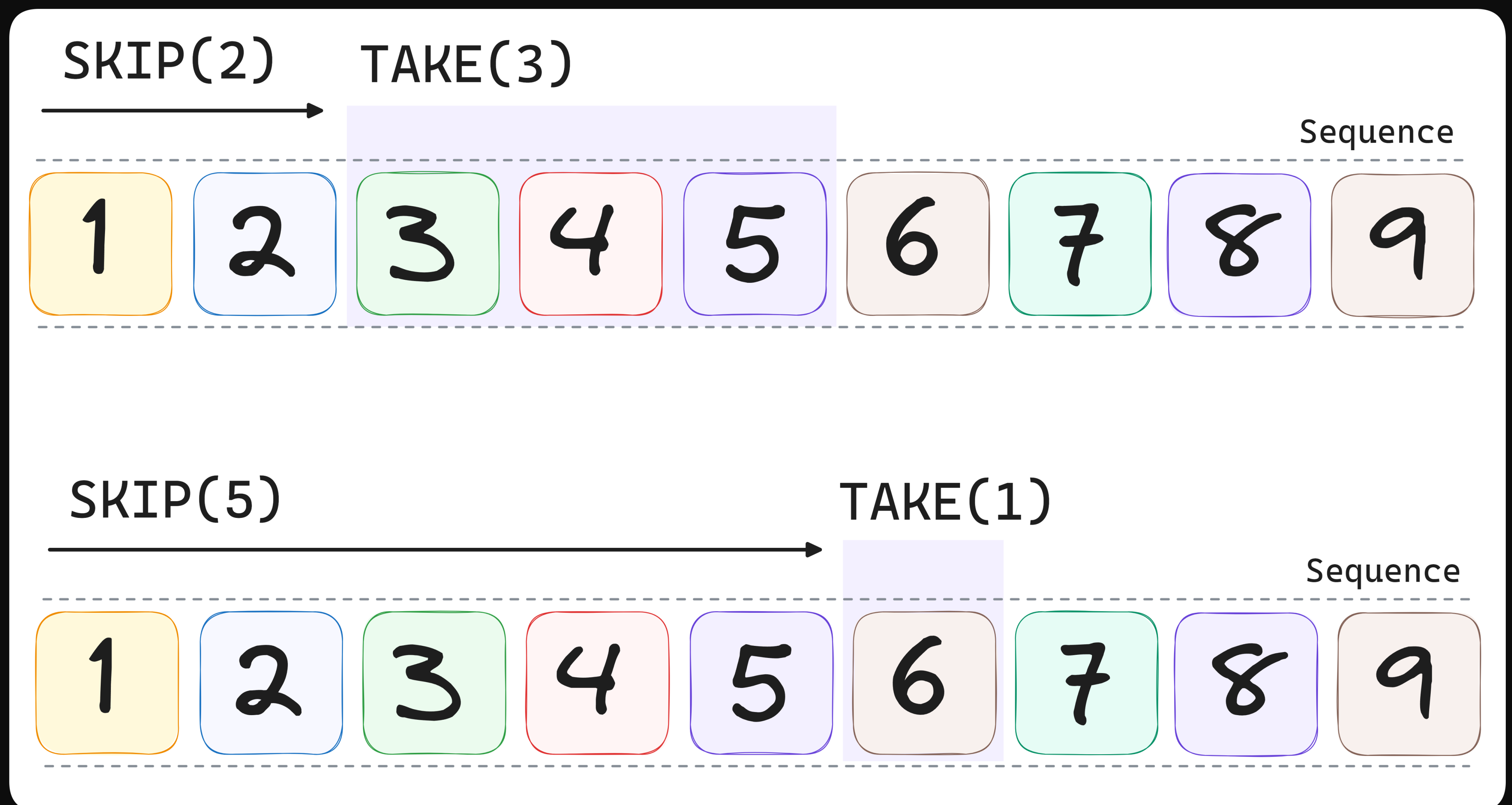
Where a **user interface** is shown that allows the user to navigate to the next or previous page of the results.





# LINQ

In **LINQ** you can use **Take** and **Skip** methods for implement pagination known **Paging**.



# Paging



C# Extension

```
static IQueryable<T> Page<T>(this IQueryable<T> data, int pageIndex, int pageSize)
    => data.Skip((pageIndex - 1) * pageSize)
        .Take(pageSize);
```



C# Usage

```
var articlesOnPage = dbContext.Articles
    .Page(pageIndex: 10, pageSize: 25)

// takes articles from 226 to 250
```

# First Page



C# Extension

```
static IQueryable<T> FirstPage<T>(this IQueryable<T> data, int pageSize)  
    => data.Take(pageSize);
```



C# Usage

```
var articlesOnPage = dbContext.Articles  
                                .FirstPage(pageSize: 25)  
  
// takes articles from 1 to 25
```

# Last Page



C# Extension

```
static IQueryable<T> LastPage<T>(this IQueryable<T> data, int pageSize)  
    => data.Skip(((data.Count() / pageSize) - 1) * pageSize)  
           .Take(pageSize)
```



C# Usage

```
var articlesOnPage = dbContext.Articles  
                                .LastPage(pageSize: 25)  
  
// takes articles from 476 to 500 if you have 500 items
```



# Page Count



C# Extension

```
static IQueryable<T> CountOfPages<T>(this IQueryable<T> data, int pageSize)
{
    var total = data.Count();
    return (total / pageSize) + ((total % pageSize) > 0 ? 1 : 0);
}
```



C# Usage

```
var pageCount = dbContext.Articles
                        .CountOfPages(pageSize: 25)

// It's good when you want to generate a UI pager
```



# Notice

Before paging you must be applying your **filters** and **ordering by**.



```
var articlesOnPage = dbContext.Articles
    .Where(art => art.AuthorId == autId)
    .OrderBy(art => art.Date)
    .ThenBy(art => art.Title)
    .Page(pageIndex: 10, pageSize: 25);
```



Repost, so your friends can **learn** too.

**Let's follow**