

Searching, Filtering and Paging Resources



Kevin Dockx

Architect

@Kevindockx | www.kevindockx.com

Coming Up



Filtering and searching resources

Paging resources



Filtering

Filtering a collection means limiting the collection taking into account a predicate



<https://host/api/cities?name=Antwerp>

Filtering

Pass the field name and value via query string

The filter is applied to the field name passed through



Searching

Searching a collection means adding matching items to the collection based on a predefined set of rules

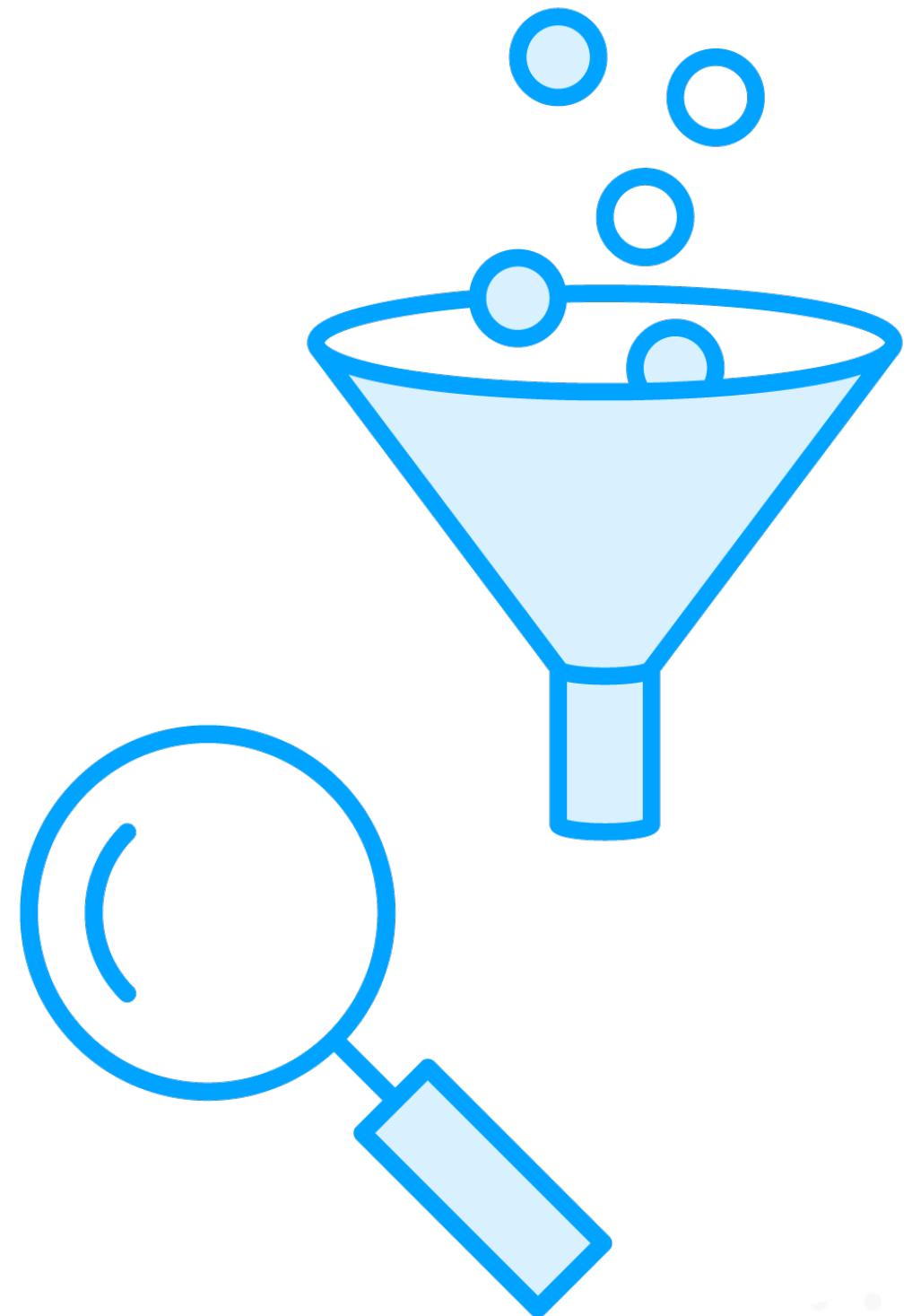


<https://host/api/cities?searchQuery=Tower>

Searching

**Pass through a value to search for via the query string
It's up to the API to decide how to implement the search functionality**





Filtering allows you to be precise by adding filters until you get exactly the results you want

Searching allows you to go wider – it's used when you don't exactly know which items will be in the collection



Demo



Filtering resources



Demo



Searching through resources



Deferred Execution

Query execution occurs sometime after the query is constructed



Deferred Execution

A query variable stores query commands, not results

- `IQueryable<T>`: creates an expression tree

Execution is deferred until the query is iterated over

- `foreach` loop
- `ToList()`, `ToArray()`, `ToDictionary()`
- Singleton queries



Paging through Resources

Collection resources often grow quite large

- Implement paging on all of them

Paging helps avoid performance issues



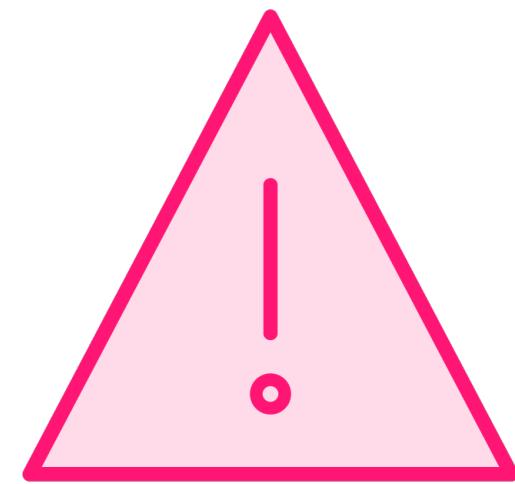
`https://host/api/cities?pageNumber=1&pageSize=5`

Paging through Resources

Pass parameters via the query string



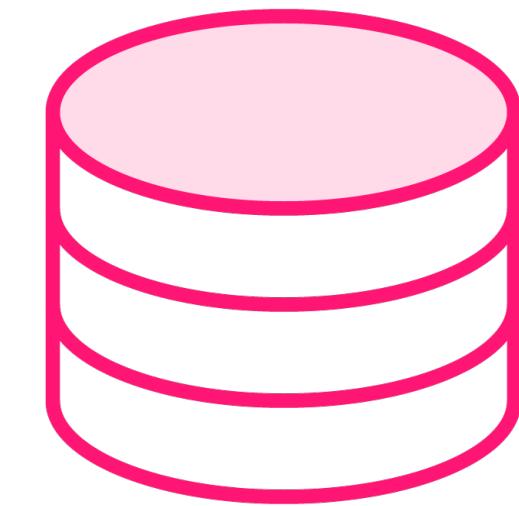
Paging through Resources



Limit the page size



Page by default



Page all the way through to the underlying data store



Demo



Paging through resources



Returning Pagination Metadata

Pagination metadata often includes:

- Total amount of items
- Total amount of pages
- Current page number
- Page size
- ...



```
{  
  "results": [ {city}, {city}, ...],  
  "metadata": { "previousPage" : "/api/...", ...}  
}
```

Pagination Metadata

An envelope that includes a `value` & `meta` property isn't a JSON representation of the cities collection resource. It's a JSON representation of cities with paging metadata.



Returning Pagination Metadata

When requesting `application/json`, **paging** metadata isn't part of the resource representation

Use a custom header, like `X-Pagination`



Demo



Returning pagination metadata



Summary



Filtering allows you to be precise by adding filters until you get exactly the results you want

Searching allows you to go wider – it's used when you don't exactly know which items will be in the collection



Summary



Deferred execution means query execution occurs sometime after the query is constructed



Summary



When correctly implemented, paging will improve performance

Pass page size and page number as query string parameters

- Limit page size
- Provide default values for page size and page number



Summary



Page all the way through to the underlying data store

- Deferred execution
- `Skip()`, `Take()`

Page by default

Return pagination metadata in a custom pagination header



Up Next:

Securing Your API

