Using OData Queries

Deborah Kurata

@deborahkurata | blogs.msmvps.com/deborahk/

pluralsight

Touch Points

- Michelangelo Buonarroti via Wikimedia Commons
https://www.flickr.com/photos/34409164@N06/4277702120

- Retrieving Data
- Filtering, Shaping, Querying Data
- Saving Data
- Error Handling
- User Authentication
- User Authorization

pluralsight

OData

- Open Data Protocol
- Standard way to access data using HTTP
- Protocol largely follows REST conventions
- For more information:
  - Search "OData" in the Pluralsight library

pluralsight

OData Queries

- Allow construction of complex data queries
- Filter, shape, sort, and select data
- Examples:
  - `$filter: "contains(ProductCode,'GDN')"`
  - `$filter: "contains(ProductCode,'GDN') and Price lt 10"`
  - `$filter: "contains(ProductCode,'GDN') and Price gt 5 and Price lt 20", $orderby: "Price desc"`

pluralsight

Web API Supports OData Queries

pluralsight

Module Objectives

Enabling OData Queries in Web API

Learning OData Query Options

Using OData Queries from Angular

Understanding OData Expression Syntax

OData Query Security Considerations

pluralsight

```
[EnableQuery()]
public IEnumerable<Product> Get()
{
  var productRepository = new ProductRepository();
  return productRepository.Retrieve();
}
```

Enabling OData Queries in Web API

Add the EnableQuery Attribute

Change the IEnumerable return type to IQueryable

pluralsight

```
[EnableQuery()]
public IQueryable<Product> Get()
{
  var productRepository = new ProductRepository();
  return productRepository.Retrieve();
}
```

Enabling OData Queries in Web API

Add the EnableQuery Attribute

Change the IEnumerable return type to IQueryable

Add AsQueryable() to convert the IEnumerable to an IQueryable

pluralsight

```
[EnableQuery()]
public IQueryable<Product> Get()
{
  var productRepository = new ProductRepository();
  return productRepository.Retrieve().AsQueryable();
}
```

Enabling OData Queries in Web API

Add the EnableQuery Attribute

Change the IEnumerable return type to IQueryable

Add AsQueryable() to convert the IEnumerable to an IQueryable

pluralsight

OData Query Options

```
http://localhost:52293/api/products?$filter=Price+gt+5&$orderby=Price
```

pluralsight

OData Query Options

Option	Description	Example
$top	Returns the top n results	$top: 3
$skip	Skips n results	$skip: 1
$orderby	Sorts the results	$orderby: "ProductName desc"
$filter	Filters the results based on an expression	$filter: "Price gt 5"
$select	Selects the properties to include in the response	$select: "ProductName, ProductCode"

pluralsight

```
productResource.query({
  $filter: "contains(ProductCode, 'GDN') and Price lt 10",
  $orderby: "Price desc"}, function (data) {
  vm.products = data;
})
```

Using OData Queries From Angular

Pass the query options as the first argument to the $resource query method.

pluralsight

Exposing OData Query Options to the User

Acme Product Management

Product List

Product	Code	Available	Price
Leaf Rake	GDN-0011	March 19, 2009	$19.95
Garden Cart	GDN-0023	March 18, 2010	$32.99
Hammer	TBX-0048	May 21, 2013	$8.99
Saw	TBX-0022	May 15, 2009	$11.55
Video Game Controller	GMG-0042	October 15, 2002	$35.95

pluralsight

```
vm.searchCriteria = "GDN";
vm.sortProperty = "Price";
vm.sortDirection = "desc";

productResource.query({
  $filter: "contains(ProductCode, '" + vm.searchCriteria +"')" +
      " or contains(ProductName, '" + vm.searchCriteria + "')",
  $orderby: vm.sortProperty + " " + vm.sortDirection
  }, function (data) {
        vm.products = data;
})
```

pluralsight

Understanding
OData Expression Syntax

pluralsight

$filter Logical Operators

Operator	Description	Example
eq	Equal	$filter: "Price eq 10"
ne	Not Equal	$filter: "Price ne 10"
gt	Greater than	$filter: "Price gt 10"
ge	Greater than or equal	$filter: "Price ge 10"
lt	Less than	$filter: "Price lt 10"
le	Less than or equal	$filter: "Price le 10"
and	Logical and	$filter: "Price gt 10 and Price lt 20"
or	Logical or	$filter: "Price eq 10 or Price eq 20"
not	Logical negation	$filter: "(Price lt 10) and not (Price eq 0)"

pluralsight

$filter String Functions

Function	Description	Example
startswith	String begins with	$filter: "startswith(ProductCode, 'GDN')"
endswith	String ends with	$filter: "endswith(ProductCode, '001')"
contains	String contains	$filter: "contains(ProductCode, 'GDN')"
tolower	Converts the string to lower case	$filter:"tolower(ProductCode) eq 'gdn-001'"
toupper	Converts the string to upper case	$filter:"toupper(ProductCode) eq 'GDN-001'"

pluralsight

Other $filter Features

Type	Examples
Arithmetic operators	Addition, subtraction, etc
Date functions	year, month, day, now, etc
Numeric functions	round, floor, etc
Geo functions	distance, length, etc
Lambda operators	any, all
Literals	null, etc

http://www.odata.org/ Developers -> Documentation

pluralsight

OData Query Security Considerations

- Don't expose OData query options directly to the user
  - Control the query filters and sorts through the user interface
  - Enable CORS to restrict access only to the client application
- But what if the API is public?

pluralsight

Web API Actions

- Use the JsonIgnore attribute as needed
- Use the EnableQuery attribute sparingly
- Consider limiting the page size
  - [EnableQuery(PageSize=50)]
- Consider limiting the available query options
  - [EnableQuery(AllowedQueryOptions=AllowedQueryOptions.Skip | AllowedQueryOptions.Top)]
- Consider writing a validator function
  - Override ValidateQuery

pluralsight

This Module Covered

Enabling OData Queries in Web API

IQueryable

Learning OData Query Options

$top, $skip, $orderby, $filter, $select

Using OData Queries from Angular

Understanding OData Expression Syntax

OData Query Security Considerations

pluralsight

Touch Points

- Michelangelo Buonarroti via Wikimedia Commons
https://www.flickr.com/photos/34409164@N06/4277702120

- Retrieving Data
- Filtering, Shaping, Querying Data
- Saving Data
- Error Handling
- User Authentication
- User Authorization

pluralsight