

Common Lisp ODATA Client

Mariano Montone (marianomontone@gmail.com)

Table of Contents

1	Introduction	1
2	Installation	2
3	Usage	3
4	API	4
4.1	ODATA-CLIENT package	4
4.2	ODATA/LANG package	5
5	Index	10

1 Introduction

CL-ODATA-CLIENT is a Common Lisp client for accessing ODATA services.

2 Installation

3 Usage

4 API

4.1 ODATA-CLIENT package

ODATA-CLIENT [PACKAGE]

Provides core functions for interacting with an ODATA service.

External definitions

Variables

ACCESS-TOKEN [ODATA-CLIENT]

ODATA service api token.

ODATA-BASE [ODATA-CLIENT]

ODATA service base url.

Macros

ODATA-CLIENT:WITH-ODATA-BASE (*base &body body*) [Macro]

Functions

ODATA-CLIENT:ODATA-POST (*uri data &key (json-encode t)* *authorization*) [Function]

Make a POST request to ODATA service at *URI*.

DATA is the *data* to be posted. It is encoded using `<undefined>` [ENCODE-JSON-TO-STRING], page `<undefined>`.

ODATA-CLIENT:ODATA-GET* (*uri &rest args &key \$filter \$expand*) [Function]

Make an [ODATA-GET], page 4 request using [*ODATA-BASE*], page 4 as URL base.

ODATA-CLIENT:ODATA-PUT (*uri data &key (json-encode t)* *authorization*) [Function]

Make a PUT (update) request to ODATA service at *URI*.

DATA is the *data* to be posted. It is encoded using `<undefined>` [ENCODE-JSON-TO-STRING], page `<undefined>`.

ODATA-CLIENT:ODATA-PATCH (*uri data &key (json-encode t)* *authorization*) [Function]

Make a PATCH request to ODATA service at *URI*.

DATA is the *data* to be posted. It is encoded using `<undefined>` [ENCODE-JSON-TO-STRING], page `<undefined>`.

ODATA-CLIENT:ODATA-GET (*url &key \$filter \$expand authorization*) [Function]

`<undefined>` [GET], page `<undefined>` request on an ODATA service at *URL*.

\$filter is an ODATA *\$filter* expression.

\$expand is an ODATA *\$expand* expression.

AUTHORIZATION is the *authorization* token.

See: [http://docs.oasis-open.org/odata/odata/v4.0/errata03/os/complete/part1-protocol/odata-v4.0-errata03-os-part1-protocol-complete.html#_The_\\$filter_System](http://docs.oasis-open.org/odata/odata/v4.0/errata03/os/complete/part1-protocol/odata-v4.0-errata03-os-part1-protocol-complete.html#_The_$filter_System)

Classes

ODATA-CLIENT:ODATA-REQUEST-ERROR [Class]

Class precedence list: odata-request-error, error, serious-condition, condition, t

4.2 ODATA/LANG package

ODATA/LANG [PACKAGE]

This package exports functions that are meant to be used with arrows syntax to interact with an ODATA service.

External definitions

Functions

ODATA/LANG:FETCH (*msgraph::url* &**optional** *type*) [Function]

ODATA/LANG:\$FILTER (*url exp*) [Function]

Add ODATA \$filter parameter to *URL*.

The \$filter system query option allows clients to filter a collection of resources that are addressed by a request *URL*. The expression specified with \$filter is evaluated for each resource in the collection, and only items where the expression evaluates to true are included in the response. Resources for which the expression evaluates to false or to null, or which reference properties that are unavailable due to permissions, are omitted from the response.

See: [\[ODATA-CLIENT::compile-\\$filter\]](#), page [\[ODATA-CLIENT::compile-\\$filter\]](#)

See: <https://www.odata.org/getting-started/basic-tutorial/#filter>

ODATA/LANG:\$VALUE (*url*) [Function]

Address the raw value of a primitive property.

Example: returns the raw value of property Name of an Airport.

(-> +trip-pin-modify+

(collection "Airports") (id "KSFO")

(property "Name") (\$value)

See: <https://www.odata.org/getting-started/basic-tutorial/#propertyVal>

ODATA/LANG:PROPERTY (*url name*) [Function]
Access the resource property with *name NAME*.

ODATA/LANG:COLLECTION (*url name*) [Function]
Access the resource collection at *NAME*.

ODATA/LANG:\$SKIP (*url skip*) [Function]
The \$skip query option requests the number of items in the queried collection that are to be skipped and not included in the result.

See: <https://www.odata.org/getting-started/basic-tutorial/#topskip>

ODATA/LANG:PATCH (*url data*) [Function]
Perform a resource [PATCH], page 6 request with *DATA* to ODATA service at *URL*.

ODATA/LANG:\$TOP (*url top*) [Function]
The \$top system query option requests the number of items in the queried collection to be included in the result.

See: <https://www.odata.org/getting-started/basic-tutorial/#topskip>

ODATA/LANG:SINGLETON (*url name*) [Function]
Access the ODATA singleton with *name NAME* at *URL*.
See: <https://www.odata.org/getting-started/advanced-tutorial/#querySingleton>.

ODATA/LANG:POST (*msgraph::url &optional msgraph::data*) [Function]

ODATA/LANG:\$COUNT (*url*) [Function]
The \$count system query option allows clients to request a count of the matching resources included with the resources in the response.

ODATA/LANG:UPDATE-LINK (*url data*) [Function]
Update an already existent link.

A successful PUT request to a single-valued navigation property's reference resource changes the related entity.

Example: change the Airline of a Flight

```
(-> +trip-pin-modify+
(collection "People")
(id "russellwhyte")
(path "Trips(0)"
"PlanItems(11)"
"Microsoft.OData.SampleService.Models.TripPin.Flight "
"Airline")
(update-link '("@odata.context" . ,(⟨undefined⟩ [quri:render-uri], page ⟨undefined⟩
+trip-pin-modify+))
"@odata.id" . "Airlines('FM')"))))
```

ODATA/LANG:LINK (*url data*) [Function]

Add a link to a related entity.

Relationships from one entity to another are represented as navigation properties. A successful [POST], page 6 request to a navigation property's references collection adds a relationship to an existing entity.

Example: add 'vincentcalabrese' to friends of 'scottketchum'

```
(-> +trip-pin-modify+
(collection "People") (id "scottketchum")
(property "Friends") ($ref)
(link '("@odata.context" . ,(⟨undefined⟩ [uri:render-uri], page ⟨undefined⟩ +trip-
pin-modify+))
"@odata.id" . "People('vincentcalabrese')"))))
```

ODATA/LANG:\$ORDERBY (*url property &optional (order :asc)*) [Function]

The \$orderby system query option allows clients to request resources in either ascending *order* using *asc* or descending *order* using *desc*. If *asc* or *desc* not specified, then the resources will be ordered in ascending *order*.

ODATA/LANG:DEL (*url*) [Function]

Perform a resource deletion request at ODATA service at *URL*.

ODATA/LANG:\$EXPAND (*url exp*) [Function]

Add ODATA \$expand parameter to *URL*.

The \$expand system query option specifies the related resources to be included in line with retrieved resources.

EXP is the list of things to expand.

Examples:

```
'("asdf" "foo") => "asdf,foo"
'("asdf" "foo" ("Bar" "Baz")) => "asdf,foo,Bar/Baz"
```

See: ⟨undefined⟩ [ODATA-CLIENT::COMPILE-\$EXPAND], page ⟨undefined⟩ .

See: <https://www.odata.org/getting-started/basic-tutorial/#expand> .

ODATA/LANG:PATH (*url &rest path*) [Function]

Access entity in a *PATH*.

Example:

```
(-> +trip-pin-modify+
```

```
(collection "People")
(id "russellwhyte")
(path "Trips(0)"
"PlanItems(11)"
"Microsoft.OData.SampleService.Models.TripPin.Flight "
"Airline"))
```

ODATA/LANG:CREATE (*url data*) [Function]
Perform a resource creation request with *DATA* at ODATA service at *URL*.

ODATA/LANG:\$SELECT (*url exp*) [Function]
Adds ODATA \$select parameter to *URL*.

The \$select system query option allows the clients to requests a limited set of properties for each entity.

EXP can be either a string or a list of strings.
Elements of *EXP* are just separated by comma.

Examples:

```
(compile-$select "name") => "foo"
(compile-$select '("name" "surname")) => "name,surname"
```

See: [\[ODATA-CLIENT::COMPILE-\\$SELECT\]](#), page [\[NOT\]](#), page [\[NOT\]](#)
See: <https://www.odata.org/getting-started/basic-tutorial/#select>

ODATA/LANG:UPDATE (*url data*) [Function]
Perform a resource update (PUT request) with *DATA* to ODATA service at *URL*.

ODATA/LANG:ID (*url id*) [Function]
Get ODATA resource *id*.

ODATA/LANG:FCALL (*url name &rest args*) [Function]
Call the ODATA action with *name NAME* and arguments *ARGS*.

Actions are operations exposed by an OData service that MAY have side effects when invoked. Actions MAY return data but MUST [\[NOT\]](#), page [\[NOT\]](#) be further composed with additional path segments.

See: https://docs.oasis-open.org/odata/odata/v4.01/odata-v4.01-part1-protocol.html#sec_Actions

ODATA/LANG:\$SEARCH (*url exp*) [Function]
The \$search system query option restricts the result to include only those entities matching the specified search expression.

See: <https://www.odata.org/getting-started/basic-tutorial/#search>

ODATA/LANG:\$REF (*url*) [Function]

A successful [POST], page 6 request to a navigation property's references collection adds a relationship to an existing entity.

5 Index

(Index is nonexistent)

*

ACCESS-TOKEN	4
ODATA-BASE	4

ODATA-CLIENT:ODATA-GET	4
ODATA-CLIENT:ODATA-GET*	4
ODATA-CLIENT:ODATA-PATCH	4
ODATA-CLIENT:ODATA-POST	4
ODATA-CLIENT:ODATA-PUT	4
ODATA-CLIENT:WITH-ODATA-BASE	4
ODATA/LANG:\$COUNT	6
ODATA/LANG:\$EXPAND	7
ODATA/LANG:\$FILTER	5
ODATA/LANG:\$ORDERBY	7
ODATA/LANG:\$REF	9
ODATA/LANG:\$SEARCH	8
ODATA/LANG:\$SELECT	8
ODATA/LANG:\$SKIP	6
ODATA/LANG:\$TOP	6

O

ODATA-CLIENT:*ACCESS-TOKEN*	4
ODATA-CLIENT:*ODATA-BASE*	4

ODATA/LANG:\$VALUE	5
ODATA/LANG:COLLECTION	6
ODATA/LANG:CREATE	8
ODATA/LANG:DEL	7
ODATA/LANG:FCALL	8
ODATA/LANG:FETCH	5
ODATA/LANG:ID	8
ODATA/LANG:LINK	7
ODATA/LANG:PATCH	6
ODATA/LANG:PATH	7
ODATA/LANG:POST	6
ODATA/LANG:PROPERTY	6
ODATA/LANG:SINGLETON	6
ODATA/LANG:UPDATE	8
ODATA/LANG:UPDATE-LINK	6