

Article

[Eduard Lebedyuk](#) · Sep 4, 2017 5m read

RESTForms - REST API for your classes. Part 2: Queries

In the [first article](#) I started discussing RESTForms - REST API for your persistent classes. We talked about basic features, now, I'd like to discuss advanced features - mainly queries capabilities:

- Basic queries
- Query arguments
- Custom queries

Queries

Queries allow getting slices of data, based on arbitrary criteria. There are two query types in RESTForms:

- Basic queries work for all RESTForms classes once defined and they differ only by the field list
- Custom queries work only for the classes in which they are specified and available, but the developer has full access to query text

Basic queries

Defined once and immediately available for all or some classes. Some basic queries are system defined, more can be added by developers, and all these queries define only SELECT field list. Everything else (filtering, pagination, etc. is done by RESTForms)

Call `form/objects/:class/:query`, to execute a simple query. Second `:query` parameter determines query name - the contents of query between SELECT and FROM. Here are default query types:

Query	Description
all	all information
info	displayName and id
infoclass	displayName, id, class
count	number of rows

For example to get some basic information about `Form.Test.Person` objects we can execute `infoclass` query:

```
form/objects/Form.Test.Person/infoclass
```

```
{ "children": [
  { "_id": "1", "displayName": "Alice", "_class": "Form.Test.Person" },
  { "_id": "2", "displayName": "Charlie", "_class": "Form.Test.Person" },
  { "_id": "3", "displayName": "William", "_class": "Form.Test.Person" }
]}
```

RESTForms looks for a query named `myq` in the following places (till first hit):

1. Class method queryMYQ in your form class
2. Parameter MYQ in your queries class
3. Class method queryMYQ in your queries class
4. Parameter MYQ in Form.REST.Objects class
5. Class method queryMYQ in Form.REST.Objects class

You can define your own queries class (for entries 2, 3 in the above list) - special class that holds query definitions available for all classes. To define your own query named `myq` there:

1. (Once) Define a class `YourClassName`
2. Define there a `MYQ` parameter or `queryMYQ` class method. Parameter takes precedence over the method.
3. Method or param must return the part of SQL query between `SELECT` and `FROM`
4. (Once) Execute in a terminal: `Do ##class(For.Settings).setSetting("queryclass", YourClassName)`

Method signature is:

```
ClassMethod queryMYQ(class As %String) As %String
```

You can also define a class-specific query. To define your own class query named `myq`:

1. Define a `queryMYQ` class method in your form class
2. Method signature is: `ClassMethod queryMYQ() As %String`
3. Method must return the part of SQL query between `SELECT` and `FROM`

URL arguments

You can supply filters and other parameters in URL. All arguments are optional.

Argument	Sample Value	Description
size	2	page size
page	1	page number
filter	Value+contains+W	WHERE clause
orderby	Value+desc	ORDER BY clause
collation	UPPER	COLLATION clause
nocount	1	Remove count of rows (speeds u

Here's some information on these arguments.

ORDER BY clause

Changes results ordering. Value can be: `Column` or `Column+desc`. `Column` is a column from the sql table or a column number.

WHERE clause

Filter condition in a format: `Column+condition+Value`.

Several conditions are possible: `Column+condition+Value+Column2+condition2+Value2`.

Arrow syntax and serial objects are also supported: `Column_ColumnField+condition+Value`

If Value contains white spaces replace them with tabs before sending to the server.

URL	SQL
neq	!=
eq	=
gte	>=
gt	>
lte	<=
lt	<
startswith	%STARTSWITH
contains	[
doesnotcontain	'[
in	IN
like	LIKE

Example requests:

```
form/objects/Form.Test.Simple/info?size=2&page=1&orderby=text
form/objects/Form.Test.Simple/all?orderby=text+desc
form/objects/Form.Test.Simple/all?filter=text+eq+Hello
form/objects/Form.Test.Person/infoclass?filter=company_name+contains+a
form/objects/Form.Test.Simple/all?filter=text+in+A9044~B5920
```

Note, that for SQL access user must have relevant SQL privileges (SELECT on form table).

COLLATION clause

In a format: `collation=UPPER` or `collation=EXACT`. Forces specified collation on WHERE clause. If omitted, default collation is used.

Pagination

Pagination is available with 25 rows per page by default. To change page size and current page provide `size` and `page` (1-based) arguments.

Custom queries

Call `form/objects/:class/custom/:query`, to execute a custom query. Custom query allows developer to determine the full content of the query. URL parameters besides `size` and `page` are unavailable. Your method must parse all other url parameters (or call default parsers from `Form.JSON.SQL`).

To define your own custom query named `myq`:

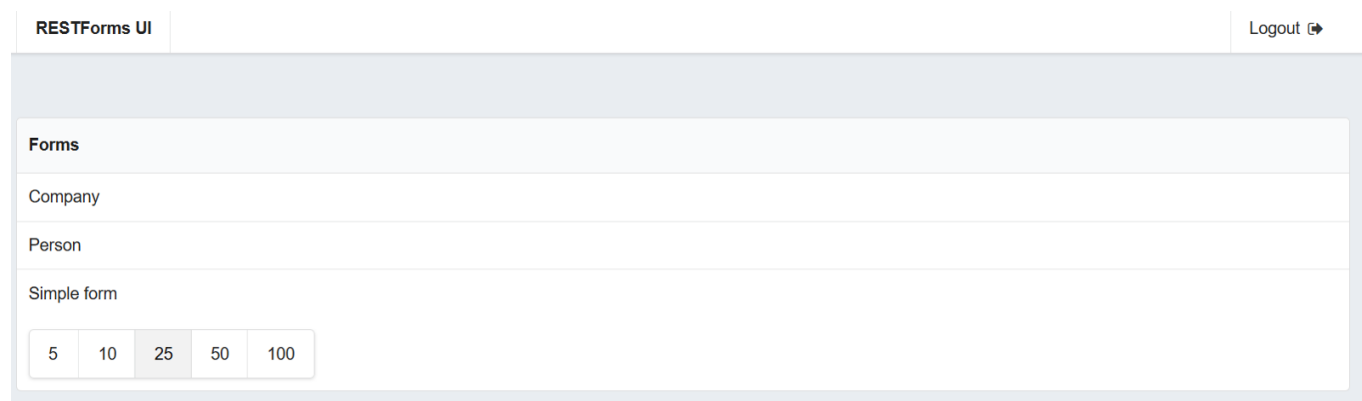
1. Define a `customqueryMYQ` class method in your form class
2. Method signature is: `ClassMethod customqueryMYQ() As %String`

3. Method must return a valid SQL query

Demo

You can try RESTForms online [here](#) (user: Demo, pass: Demo) .

Additionally there is a RESTFormsUI application - editor for RESTForms data, check it out [here](#) (user: Demo, pass: Demo). Screenshot of the class list:



The screenshot shows the RESTForms UI application interface. At the top, there is a header bar with "RESTForms UI" on the left and a "Logout" button with a user icon on the right. Below the header, there is a main content area with a light blue background. Inside this area, there is a white box with a "Forms" section. Under "Forms", there is a list of classes: "Company", "Person", and "Simple form". Below the list, there is a row of five buttons with the numbers "5", "10", "25", "50", and "100". The "25" button is highlighted with a grey background.

Conclusion

RESTForms provides extensive and customizable query capabilities.

What's next

In the next article, I'd like to tell you about some advanced features:

- Metadata translation
- Security and permissions
- Object name

Links

- [RESTForms GitHub repository](#)
- [RESTForms UI GitHub repository](#)

[#Frontend](#) [#SOAP](#) [#Cache](#)

Source URL: <https://community.intersystems.com/post/restforms-rest-api-your-classes-part-2-queries>