

# How to get Catalog Variables via REST Table API

Example: let's say we have a Request Item (RITM0010004) from a submitted Request via a Catalog Item:

Requested Item  
RITM0010004

Number: RITM0010004

Item: Sample Item

Request: REQ0010003

Requested for: Demo Admin

Due date: 2018-08-05 10:42:50

Configuration item:

Watch list:

Opened: 2018-08-03 10:42:50

Opened by: Demo Admin

Stage: Assess or Scope Task

State: Open

Quantity: 1

Estimated delivery:

Backordered: ☐

Order Guide:

Assigned to: Demo Admin

Variables

First, let's copy the RITM number:

Requested Item  
RITM0010004

Number: RITM0010004

Item: Sample Item

Request: REQ0010003

**Step One:** Go to the REST API Explorer:

The screenshot shows the 'REST API Explorer' interface. On the left, there are filters for 'Namespace' (set to 'now'), 'API Name' (set to 'Table API'), and 'API Version' (set to 'latest'). Below these is a list of actions: 'Retrieve records from a table (GET)' (highlighted with a blue arrow), 'Create a record (POST)', 'Retrieve a record (GET)', 'Modify a record (PUT)', 'Delete a record (DELETE)', and 'Update a record (PATCH)'. The main area is titled 'Table API' and describes its function: 'Allows you to perform create, read, update and delete (CRUD) operations on existing tables'. It includes a section 'Retrieve records from a table' with a GET request URL: 'https://dev12567.service-now.com/api/now/table/{tableName}'. Below this is the 'Prepare request' section, which is divided into 'Path parameters' and 'Query parameters'. Under 'Path parameters', there is a table with two columns: 'Name' and 'Value'. The 'Name' column has a red asterisk next to 'tableName'. The 'Value' column has a dropdown menu open, showing a search bar with 'sc\_req\_item' and a list of options: 'Catalog Request Item Metric (sc\_req\_item\_metric)', 'Catalog Request Item SLA (sc\_req\_item\_sla)', and 'Requested Item (sc\_req\_item)' (which is highlighted in blue). Under 'Query parameters', there is a table with two columns: 'Name' and 'Value'. The 'Name' column has 'sysparm\_query' and 'sysparm\_display\_value'.

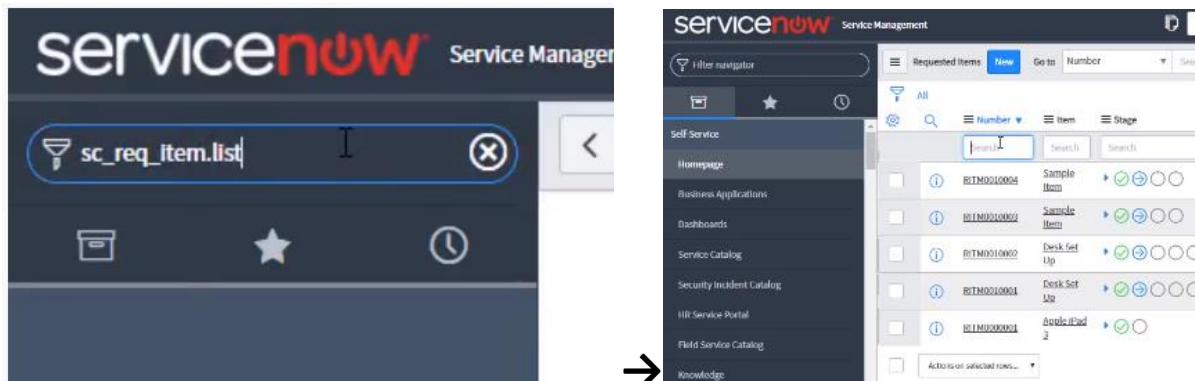
Name	Value
* tableName	-- Select a table --

Name	Value
sysparm_query	
sysparm_display_value	

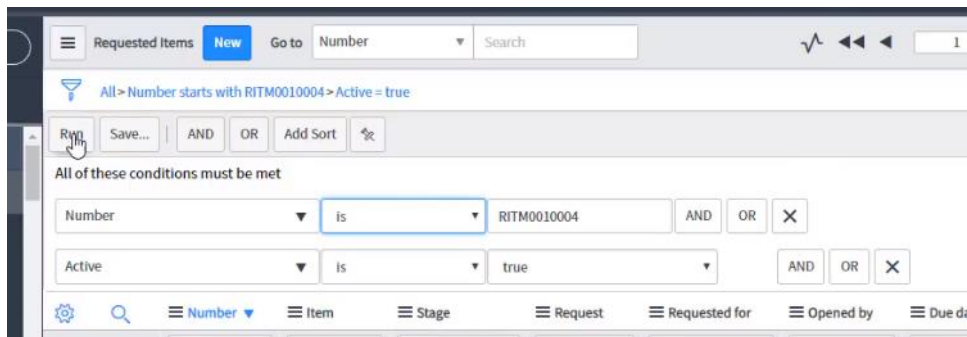
...for the tableName, we chose the Value of Requested Item (sc\_req\_item).

**Step Two:** Query for the desired RITM in the List View of Requested Items.

We can go to the Table for Requested Items by searching in the Application Navigator...



...and set the Query as desired:



...click Run.

Right-click on the last part of the Query bread crumbs and copy it:



### Step Three: Paste our Query into the **sysparm\_query** Parameter:

Query parameters		
Name	Value	Description
sysparm_query	<input type="text" value="number=RITM0010004^active=true"/>	An encoded query string used to filter the results
sysparm_display_value	<input type="text"/>	Return field display values (true), actual values (false), or both (all) (default: false)
sysparm_exclude_reference_link	<input type="text"/>	True to exclude Table API links for reference fields (default: false)
sysparm_suppress_pagination_header	<input type="text"/>	True to suppress pagination header (default: false)
sysparm_fields	<input type="text"/>	A comma-separated list of fields to return in the response
sysparm_limit	<input type="text" value="1 (Limited to 1 result for testing)"/>	The maximum number of results returned per page (default: 10,000)
sysparm_view	<input type="text"/>	Render the response according to the specified UI view (overridden by sysparm_fields)
sysparm_query_category	<input type="text"/>	Name of the query category (read replica category) to use for queries

**Step Four:** input all for display of sysparm values, because it doesn't hurt to have access to more info:

Query parameters		
Name	Value	Description
sysparm_query	<input type="text" value="number=RITM0010004^active=true"/>	An encoded query string used to filter the results
sysparm_display_value	<input type="text" value="all"/>	Return field display values (true), actual values (false), or both (all) (default: false)
sysparm_exclude_reference_link	<input type="text"/>	True to exclude Table API links for reference fields (default: false)
sysparm_suppress_pagination_header	<input type="text"/>	True to suppress pagination header (default: false)
sysparm_fields	<input type="text"/>	A comma-separated list of fields to return in the response
sysparm_limit	<input type="text" value="1 (Limited to 1 result for testing)"/>	The maximum number of results returned per page (default: 10,000)
sysparm_view	<input type="text"/>	Render the response according to the specified UI view (overridden by sysparm_fields)
sysparm_query_category	<input type="text"/>	Name of the query category (read replica category) to use for queries

Step Five: specify the fields we want displayed in **sysparm\_fields** :

Configuring sysparm\_fields

Available

- Activity due
- Quantity
- Additional comments
- SLA due
- Approval
- Comments and Work notes
- Due Date
- Updates
- Recurring Price**
- Tags
- Billable
- Item
- Stage
- Escalation
- Upon approval
- Correlation ID
- Location
- Estimated delivery

>

<

Selected

- Number
- Sys ID
- Short Description
- Assigned to**

^

v

Cancel Save

Hit **Send** :

Add query parameter

Request headers

Name	Value	Description
Request format	application/json	Format of REST request body
Response format	application/json	Format of REST response body
Authorization	Send as me	Send the request as the current user. To send the request with another user's credentials use the provided code samples, such as cURL.

Add header

**Send** Clear response

Now we see API Explorer returns an Object (because it's JSON) with a property called **"result"** :

#### Response Body

```
{
  "result": [
    {
      "number": {
        "display_value": "RITM0010004",
        "value": "RITM0010004"
      },
      "sys_id": {
        "display_value": "c6b0060c4f771b40003233718110c781",
        "value": "c6b0060c4f771b40003233718110c781"
      },
      "short_description": {
        "display_value": "Sample Variables",
        "value": "Sample Variables"
      },
      "assigned to": {
```

...and we see the object's properties are in an array.

This array has elements.

Each element is an object.

Each object is a record.

Each object has further objects within them.

Each of the internal objects represent fields (ie, number field, sys\_id field, short\_description field). They are all strings, so no need to worry about discrepancies.

All the fields we looked at were Variables in the Catalog Item:

Variables

New

Go to

Order

Search

Catalog item = Sample Item

</

...notice we view the Name, not Question.



Notice that the assigned\_to field has a display value that differs from its actual value, as well as a link:

#### Response Body

```
{
  "sys_id": {
    "display_value": "c6b0060c4f771b40003233718110c781",
    "value": "c6b0060c4f771b40003233718110c781"
  },
  "short_description": {
    "display_value": "Sample Variables",
    "value": "Sample Variables"
  },
  "assigned_to": {
    "display_value": "Demo Admin",
    "link": "https://dev12567.service-now.com/api/now/table/sys_user/6a5773d74feed300003233718110c7f9",
    "value": "6a5773d74feed300003233718110c7f9"
  }
}
```

...this is because we created Demo Admin, which we can say is “out of box.”