Create a list of articles from a specific category

Last modified on 07/16/2026 4:43 pm EDT

This example utilizes the API calls in snippets functionality to securely make calls to the API from within your live knowledge base without exposing your API key to readers. For information on how to add snippets to your KB, you can refer to our snippet documentation.

In our example code we will be using a standard jQuery GET request to hit the KnowledgeOwl API. Unlike a standard API call, however, we will be using a "**ko_api**" merge code in place of where you would normally use a URL. This merge code will convert the given parameters within it server side and render a secure, obfuscated, single-use URL client side.

Snippet Variables

By using API snippet merge code variables, this snippet can be placed within different knowledge bases or different sections of a knowledge base and the code will work without the need to modify it. Below are the variables that are being used in this example:

- `%cur_kb_id%` : replaced with the knowledge base ID the snippet is currently being used in
- `%cur_cat_id%` : replaced with either the ID of the current category the snippet is used in, or if used within an article it will be replaced with the category ID that the article belongs to. So if we were using this variable here, it would return the ID for the Cookbook: code samples category.
- `%cur_top_cat_id%` : replaced with the category ID of the top-most (root) category the snippet is being used in. So if we were using this variable here, it would return the ID for the API and webhooks category.

You may have a use case where you want to always use a specific category ID regardless of where the snippet is being used. In that case, you can hard code a category ID in place of the variable.

Other Parameters Used in the Example

In the below example, we are using some additional parameters to further filter out the returned results and only return the information that we want. The full syntax for these parameters can be found in our API documentation but here's a summarized list:

- `"status": {"$in": ["published", "review"]}` : This will filter the results to only include articles that are "published" or in "needs review"
- `"_fields": ["name","url_hash","date_new"]` : This specifies what field data that we want returned for the articles. Limiting the amount of data returned in this way will make the API call more efficient and faster.

Copyright © 2015 – 2025 Silly Moose, LLC. All rights reserved.
Powered by KnowledgeOwl knowledge base software.

- `"limit": 5` : **This specifies that we only want the first 5 articles that match our filter returned. Higher limits will be slower and less efficient so only pull back what you need.**
- `"sort": {"date_new": 1}` : **This specifies that we want the results sorted by the date_new field. The "1" specifies ascending, whereas a "-1" would specify descending.**



Due to the nature of how the ko_api snippet merge code is replaced server side, you cannot use client side variables to populate values inside of the merge code string.

Snippet API Code

```
<script>
  $(function() {
    // Get list of published articles that belong to the current category sorted by date_new
    $.get('[ko_api(article|{"project_id": "%cur_kb_id%", "status": {"$in": ["published", "review"]}, "category": "%cur_cat_id%
", "_fields": ["name","url_hash","date_new"], "limit": 5, "sort": {"date_new": 1}})]',
    function(apiData) {
        //do something with the returned data
        console.log(apiData);
    }).fail(function(error) {
      //uh oh something went wrong. Alert the end-user or otherwise handle the error
    });

    // Get list of published articles that belong to the root category sorted by date_new
    $.get('[ko_api(article|{"project_id": "%cur_kb_id%", "status": {"$in": ["published", "review"]}, "category": "%cur_top_cat
_id%", "_fields": ["name","url_hash","date_new"], "limit": 5, "sort": {"date_new": 1}})]',
    function(apiData) {
        //do something with the returned data
        console.log(apiData);
    }).fail(function(error) {
      //uh oh something went wrong. Alert the end-user or otherwise handle the error
    });
  });
</script>
```

Copyright © 2015 – 2025 Silly Moose, LLC. All rights reserved.
Powered by KnowledgeOwl knowledge base software.