

Use the Kontent.ai APIs with our Postman collection

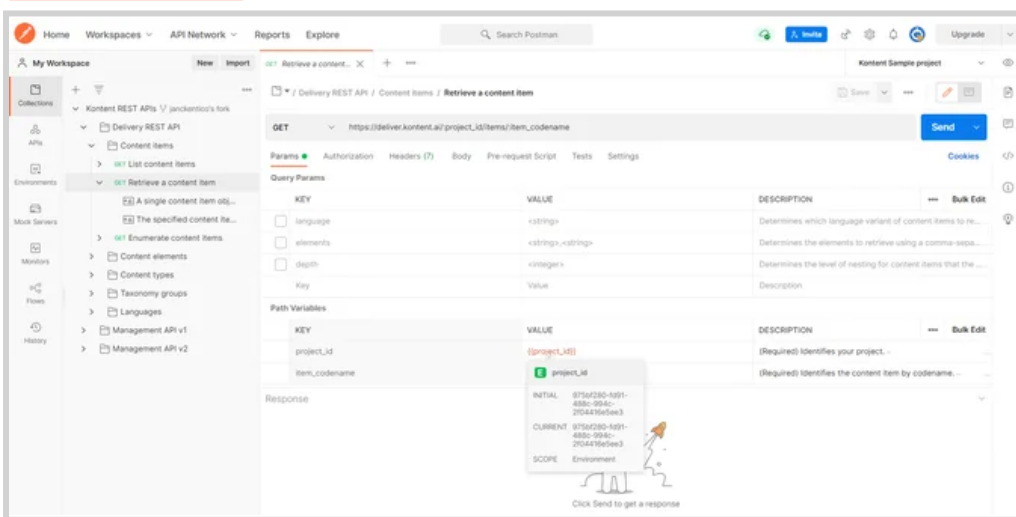
July 13, 2022 • Jan Cerman • 2 min read

Play around with the Kontent.ai APIs using our Postman collection with the [Postman client](#). The collection contains all Kontent.ai REST APIs and their endpoints, just like in the [API references](#).

1. Import the Postman collection

Use the following button to create a fork of our Postman collection. We recommend using the fork option so that you can easily stay up to date with future updates.

[▶ Run in Postman](#)



The Kontent.ai collection for Postman

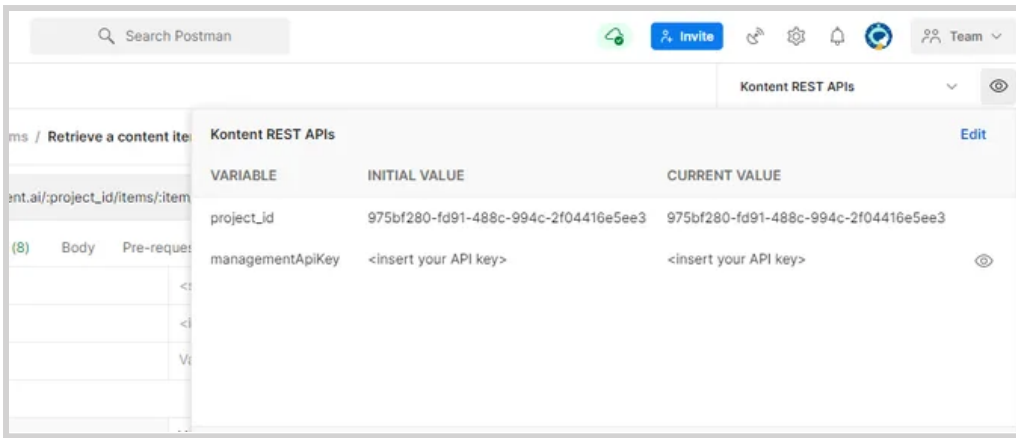
Along with the collection, you'll also get a new [Postman environment](#) called *Kontent.ai REST APIs*.

2. (Optional) Set up Postman environment variables

[Postman environments](#) are sets of variables with specific values. You can use these variables everywhere in your requests in Postman.

💡 When you define a variable in an environment (instead of editing the collection), Postman will offer the variables via its auto-complete functionality every time you edit the example requests.

For instance, the Postman environment *Kontent.ai REST APIs* comes with a variable named `project_id`. This variable is also used in all requests in the collection. We recommend you change the variable's value to the ID of your own project.



Predefined variables that come with the collection. Change their values to match your Kontent.ai project.

In the collection itself, there's also the `managementApiKey` variable set up as the authorization token for the Management API endpoints. Make sure to define the variable's value for your Postman environment.

Avoid exposing your API keys

By default, your collection and its variables are [synchronized](#) to your Postman account in the cloud.

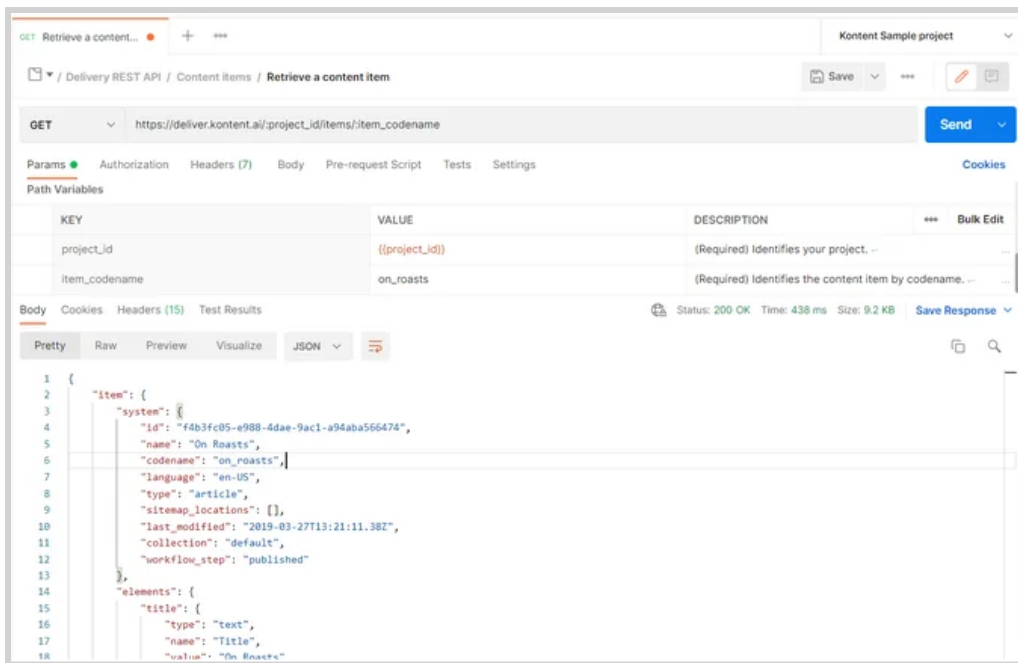
To avoid mistakenly sharing or exporting your collection along with your API keys, always store the API keys in environment variables.

3. Send requests

The collection's structure mimics what you see in the [API references](#). There are requests in each folder.

You can execute the requests by clicking **Send**. If you have an environment configured and selected in the top right corner, Postman will use the variable values of the environment.

After clicking **Send**, you'll get an API response. That's it!



The screenshot shows a Postman interface for a REST client. The request is a GET call to `https://deliver.kontent.ai/:project_id/items/:item_codename`. The path variables table shows `project_id` with value `{{project_id}}` and `item_codename` with value `on_roasts`. The response is a JSON object with the following structure:

```
1 {
2   "item": {
3     "system": {
4       "id": "f4b3fc05-e988-4dae-9ac1-a94aba566474",
5       "name": "On Roasts",
6       "codename": "on_roasts",
7       "language": "en-US",
8       "type": "article",
9       "sitemap_locations": [],
10      "last_modified": "2019-03-27T13:21:11.38Z",
11      "collection": "default",
12      "workflow_step": "published"
13    },
14    "elements": {
15      "title": {
16        "type": "text",
17        "name": "Title",
18        "value": "On Roasts"
19      }
20    }
21  }
22 }
```

A JSON response to a sample API request

What's next?

- Find details about the Kontent.ai APIs in the [API references](#).
- Play around with the [sample apps](#).