



Reportnet 3.0 API HowTo

14-12-2021

Contents

Contents	1
1 Web API	2
1.1.1 How to access swagger	2
1.1.2 Swagger authorization	2
1.1.3 Swagger parameters.....	4
1.1.4 How to test a service in swagger	7
1.1.4.1 How to add records programmatically using the API.....	9
1.1.4.2 How to delete records programmatically using the API	13



1 Web API

1.1.1 How to access swagger

Swagger UI allows to visualize and interact with the API's resources available in Reportnet.

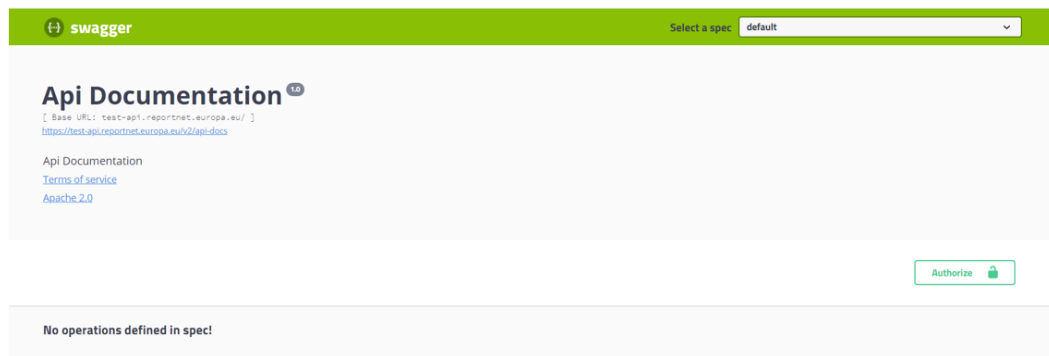
To test the API, you must go to:

TEST environment:

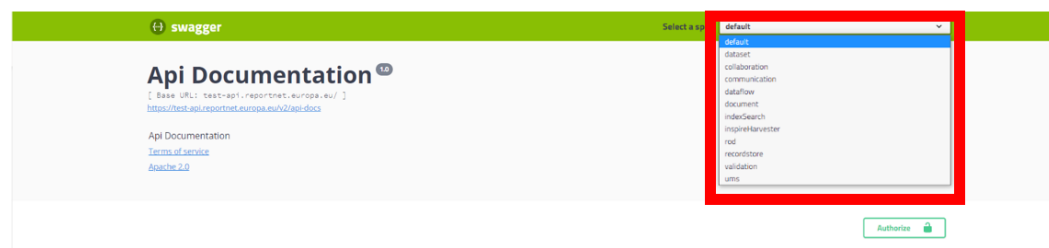
<https://test-api.reportnet.europa.eu/swagger-ui.html>

PRODUCTION environment:

<https://api.reportnet.europa.eu/swagger-ui.html>



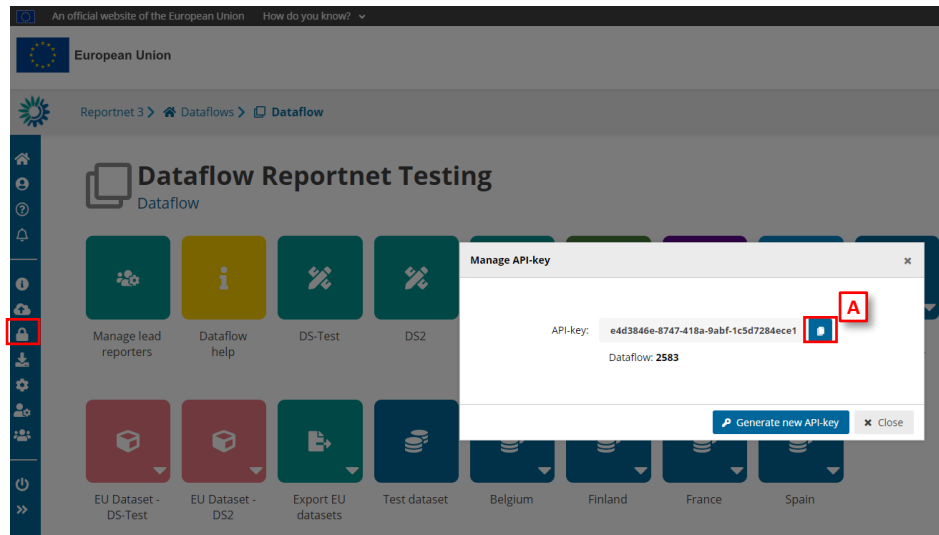
Once you access to this URL, Swagger main page will get open with all the APIs you have available.



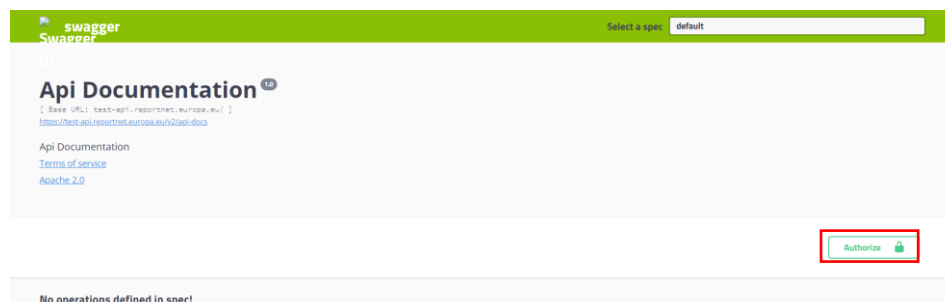
1.1.2 Swagger authorization

Before you test any service, you must be authenticated. You have two ways to do this:

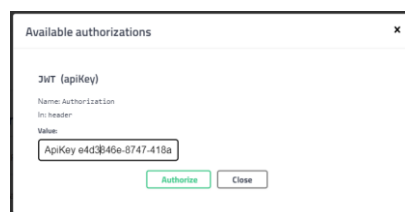
1. API-key



- Go to Reportnet on the dataflow page and click on the cog icon in the left menu.
- There will not be API generated if it is the first time for the user.
- Click on **'Generate new API-key'**.
- Once the API-key is generated, it can be copied on clipboard using [A].
- Different API-keys will be generated every time user clicks on **'Generate new API-key'**.
- Once you have the API-key you must go to the swagger page and click on the **'Authorize'** button.



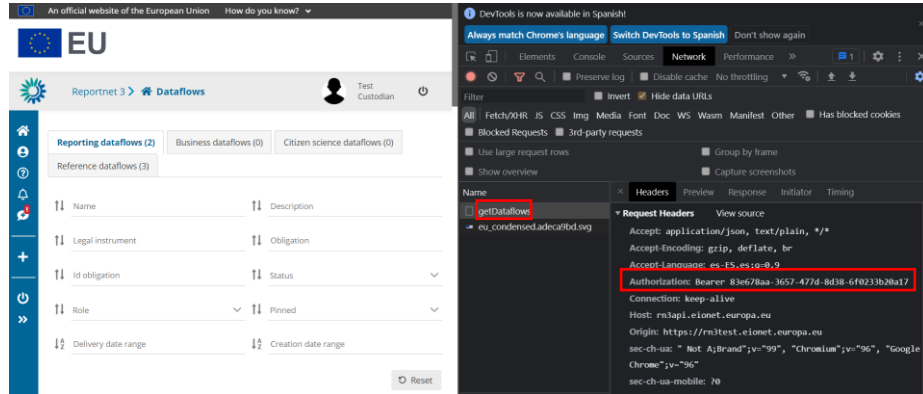
- Clicking that, brings up a dialog box where you can put your generated API-key



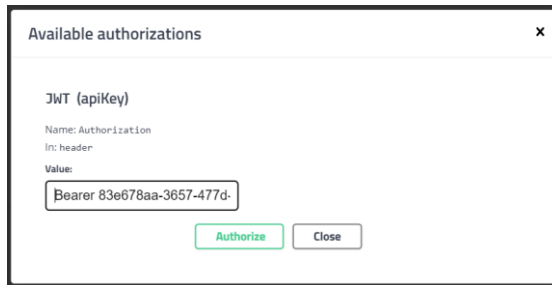
Please be aware that this method does not provide full access to all of the REST services since this authentication method is meant to provide a limited access to external systems on behalf of a User



2. Bearer Authentication



- Go to Reportnet with the DevTools open (F12) and go to the dataflow list page
- You must copy the Authorization Request Header of one of the calls you see on the Network tab.
- Once you have the *Bearer* (token authentication) you must go to the swagger page and click on the 'Authorize' button.
- Clicking that brings up a dialog box where you can put your bearer token



- All end points which you have available for each microservice, will be visible below.

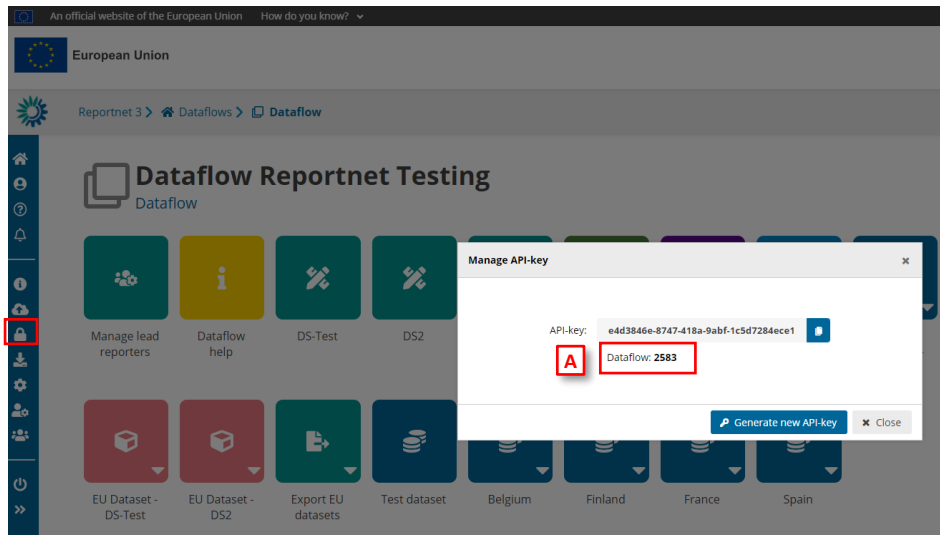
Please, be aware that bearer token only lasts 5 minutes.

1.1.3 Swagger parameters

The main parameters that you will need to use the API will be:

1. Dataflow Id:

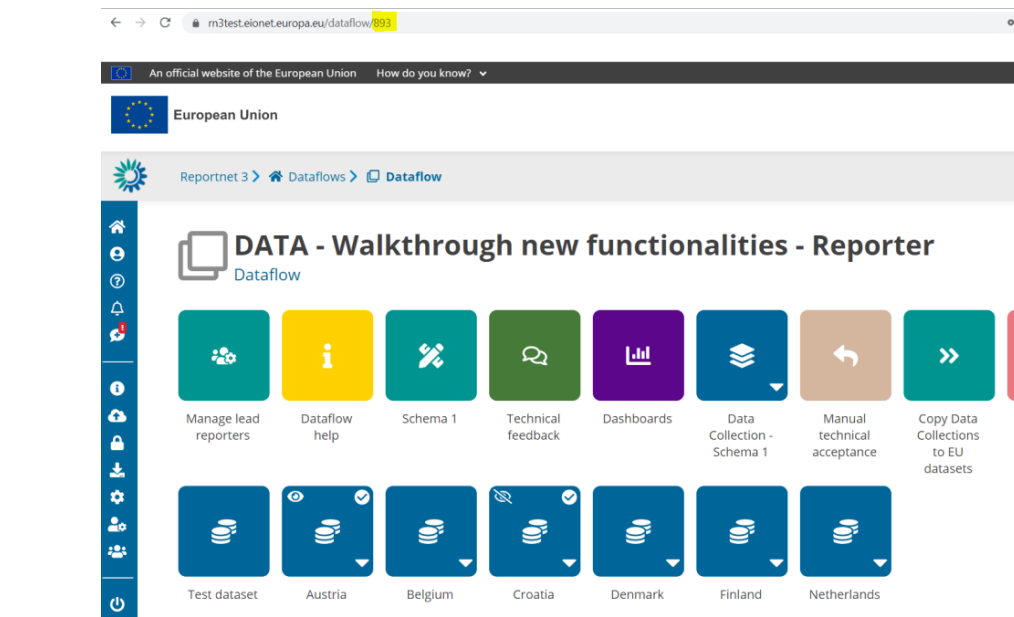
- You can get the Dataflow Id on the API dialog:



- a. Go to Reportnet on the dataflow page and click on the cog icon in the left menu.
- b. Below your API-Key you could see the **dataflowId** you need.

- Moreover, the **dataflowId** you will get it from the URL of the dataflow. Go to the dataflow you wish to use the API on to get the dataflowId (e.g. highlighted in yellow):

<https://rn3test.eionet.europa.eu/dataflow/893>





2. Dataset Id:

The **datasetId** you will get it from the URL of the dataset. Go to the dataset schema page in your dataflow you wish to use the API on to get the datasetId (e.g. highlighted in yellow):

<https://rn3test.eionet.europa.eu/dataflow/893/dataset/7148?tab=616e5c331d88e60001071259>

The screenshot shows the 'Schema 1' page for 'Final feedback' in the 'Reporter - Austria' dataflow. The page displays a table with the following data:

Validations	F1_PK	F2_Label	F3_StartDate	F4_1
10	Germany	2021-10-14	2021-10-21	

3. Provider Id:

You can get the Provider Id on the API dialog:

The screenshot shows the 'Manage API-key' dialog box in the 'DATA - Walkthrough' page for 'Dataflow - Denmark'. The dialog displays the following information:

- API-key: bdd9d0ba-73b7-4ab9-952d-9f3c9a866286
- Dataflow: 893
- Data provider: 22

The 'Data provider' value '22' is highlighted with a red box, and a red 'A' icon is visible next to it. The dialog also includes a 'Generate new API-key' button and a 'Close' button.



- Go to Reportnet on the dataflow page and click on the cog icon in the left menu.
- Below your API-Key you could see the **provider Id** you need.

1.1.4 How to test a service in swagger

Once you are authorized, you can select which controller you want to test and all the endpoints available for the selected controller will be visible below.

If you click in one of the services available [A], you could see the different endpoints [B]. For example, for the Dataflow controller, you have the available endpoints:

GET /dataflow/v1/{dataflowId}

GET /dataflow/v1/{dataflowId}/getmetabase



If you click in one of them, you could see the information about this service, like a brief **description** [A] and the **roles allowed** for this endpoint [B]. Moreover, you could test it by clicking on **'Try it out'** [C] and filling the fields that the endpoint needs.

For example, for the service *get Dataflow by id* we have to:

1. Once you have clicked in **'Try it out'** you could add the parameter **dataflowid** [A]
2. Click on **'Execute'** [B]
3. See the results [C]. In this case, it responds with a *passed* results and a response code **200**.



The screenshot shows a REST client interface with the following details:

- Execute** button and **Clear** button.
- Responses** section with **Response content type** set to **application/json**.
- curl** command: `curl -X GET "https://test-api.reportnet.europa.eu/dataflow/v1/662" -H "accept: application/json" -H "Authorization: Bearer 83e678aa-3657-477d-8d38-6f0233b28a17"`
- Request URL**: `https://test-api.reportnet.europa.eu/dataflow/v1/662`
- Server response** section with **Code** **200** and **Details** tab selected.
- Response body** (highlighted with a red box):


```
{
  "id": 662,
  "description": "Definition of data reported by countries under the Urban Waste Water Treatment Directive, Article 17 reporting obligation (UWTD Implementation Report).",
  "name": "Urban Waste Water Treatment Directive reporting under Article 17",
  "deadlineDate": null,
  "obligation": {
    "obligationId": 387,
    "obTitle": "Urban Waste Water Treatment Directive - Situation Report",
    "description": "The principal aim of this situation report is to inform the public regularly of the situation, on a given date, regarding waste water collection and treatment, as well as to present the development of the situation in relation to at least the two previous years. They have to be transmitted to the Commission as soon as published by the Member States.\r\n\r\nEvery two years relevant authorities or bodies shall publish situation reports on the disposal of urban waste water and sludge in their areas.",
    "validSince": null,
    "validId": 253482214488888,
    "comment": "Through inclusion into annex XX of the EEA agreement (Section II Water, article 13), the provisions of the UWTD Directive apply as well to the EFTA countries Iceland, Norway, Liechtenstein and Switzerland, under the provisions set out in the EEA Agreement.\r\n\r\nDetails on the reporting already made by the Member States and the related summary (synthesis) reports of the Commission on the UWTD implementation including maps can be found on the web site http://ec.europa.eu/environment/water/water-urbanwastewater/implementation/implementationreports_en.htm",
    "nextDeadline": null,
    "legalInstrument": {
      "sourceId": "543",
      "sourceTitle": "Council Directive 91/271/EEC of 21 May 1991 concerning urban waste water treatment as amended by Commission Directive 98/15/EC and Regulations 1882/2003/EC and 1137/2008/EC",
      "sourceAlias": "Urban Waste Water Treatment Directive (consolidated)"
    }
  },
  "client": null,
}
```
- Download** button.

1.1.4.1 How to add records programmatically using the API

This example consists in add records on a dataset using the API.

For example, on Reportnet we have configured a dataset with one table, and we want to add records. On the Reportnet page, you could click on “Import dataset data” to fill the dataset.

The screenshot shows the Reportnet 3.0 Dataset designer interface with the following details:

- Dataset description** section with a text input field and a **0/10000** indicator.
- Actions** bar with **Import dataset data** (highlighted with a red box), **Export dataset data**, **Delete dataset data**, **validate**, **Show validations**, **QC Rules**, **Unique constraints**, **External integrations**, **Dashboards**, **Manage copies**, and **Refresh**.
- Design** and **Tabular data** tabs.
- Table1** section with **Table description** and a **0/10000** indicator.
- Table configuration** options: **Read only table** (checkbox), **Prefilled** (checkbox), **Fixed number of records** (checkbox), and **Mandatory table** (checkbox checked).
- Table actions** bar with **Import table data**, **Export table data**, **Delete table data**, **Show/Hide columns**, and **Validation filter**.
- Table data** section with **Rows per page** (10), **Go to** (1 of 1), **Total 0 records**, and **Paste records** button.

You could do this in Swagger calling the POST API method.

1. The Reportnet API is enabled for a GET and POST methods on a dataset.



2. Before you run the service you must be authenticated in Swagger with the API-Key authorization explained in section 1.1.2
3. First, you must select the appropriate microservice. For example, for **Import data by dataset id**, the microservice is **dataset**, and the service is `/dataset/v1/{datasetId}/etlImport` [A] and then, click **'Try it out'** button [B] to see the parameters you need to fill to call this service. Moreover, Swagger gives you a json example [C] about how you must send this json parameter, to make it as easy to use as possible.

POST /dataset/v1/{datasetId}/etlImport Import data by dataset id [A]

Allowed roles:
 Reporting dataset: REPORTER WRITE, LEAD REPORTER
 Test dataset: CUSTODIAN, STEWARD
 Reference dataset: CUSTODIAN, STEWARD
 Design dataset: CUSTODIAN, STEWARD, EDITOR WRITE
 EU dataset: CUSTODIAN, STEWARD

Parameters [B] Try it out

Name	Description
dataflowId integer(\$int64) (query)	Dataflow id
datasetId integer(\$int64) (path)	Dataset id
etlDatasetVO (body)	
providerId integer(\$int64) (query)	Provider id

Data object

Example Value | Model

```
{
  "tables": [
    {
      "records": [
        {
          "countryCode": "string",
          "fields": [
            {
              "fieldName": "string",
              "value": "string"
            }
          ]
        }
      ],
      "tableName": "string"
    }
  ]
}
```

Parameter content type

4. You must fill the dataflow Id [A], dataset Id [B] and add an object where you must indicate the records you want to add [C].
 - i. The **dataflowId** and **datasetId** you will get them from the URL of the dataset. Go to the dataset schema page in your dataflow you wish to use the API on to get the dataflowId and datasetId e.g. highlighted in yellow: <https://rn3test.eionet.europa.eu/dataflow/963/datasetSchema/7698?tab=61a9e86b8b08ce0001bf4072&view=tabularData>



POST /dataset/v1/{datasetId}/etlImport Import data by dataset id

Allowed roles:
 Reporting dataset: REPORTER WRITE, LEAD REPORTER
 Test dataset: CUSTODIAN, STEWARD
 Reference dataset: CUSTODIAN, STEWARD
 Design dataset: CUSTODIAN, STEWARD, EDITOR WRITE
 EU dataset: CUSTODIAN, STEWARD

Parameters Cancel

Name	Description
dataflowId integer(\$int64) (query)	Dataflow id <input type="text" value="963"/>
datasetId integer(\$int64) (path)	Dataset id <input type="text" value="7698"/>
etlDatasetVO (body)	Data object <div style="border: 1px solid black; padding: 5px;"> <pre> Example Value Model { "tables": [{ "records": [{ "countryCode": "XX", "fields": [{ "fieldName": "test", "value": "string" }] }] }], "tableName": "Table1" } </pre> </div>
providerId integer(\$int64) (query)	Provider id <input type="text" value="providerId - Provider id"/>

Parameter content type:

Cancel

Execute

- When ready, click on the blue button **‘Execute’**. You should get a response of the service with a *passed* result and a response code 200 [A]. If you go to Reportnet, the records have been added.

Responses Response content type */*

Curl

```
curl -X POST "https://test-api.reportnet.europa.eu/dataset/v1/7698/etlImport?dataflowId=963" -H "accept: */*" -H "Authorization: ApiKey 5327776e-8972-42e3-8b6a-9e0fc4d7932" -H "Content-Type: application/json" -d [{"tables": [{"records": [{"countryCode": "XX", "fields": [{"fieldName": "test", "value": "myTest"}]}]}]}
```

Request URL

https://test-api.reportnet.europa.eu/dataset/v1/7698/etlImport?dataflowId=963

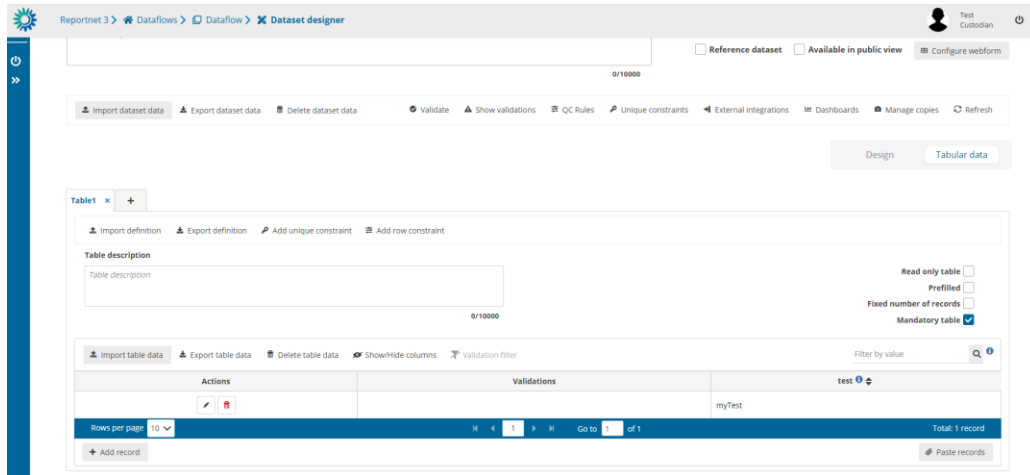
Server response

Code	Details
200	<div style="border: 1px solid red; padding: 5px;"> <p>A Response headers</p> <pre> access-control-allow-origin: * cache-control: no-cache, no-store, max-age=0, must-revalidate content-security-policy: default-src: self 'unsafe-inline' https://*.eiaonet.europa.eu date: Fri, 05 Dec 2021 10:00:54 GMT expires: 0 feature-policy: camera 'self'; microphone 'self' pragma: no-cache referrer-policy: no-referrer strict-transport-security: max-age=31536000; includeSubDomains; preload; transfer-encoding: chunked vary: Origin, Access-Control-Request-Method, Access-Control-Request-Headers x-content-type-options: nosniff x-frame-options: DENY x-xss-protection: 1;mode=block </pre> </div>

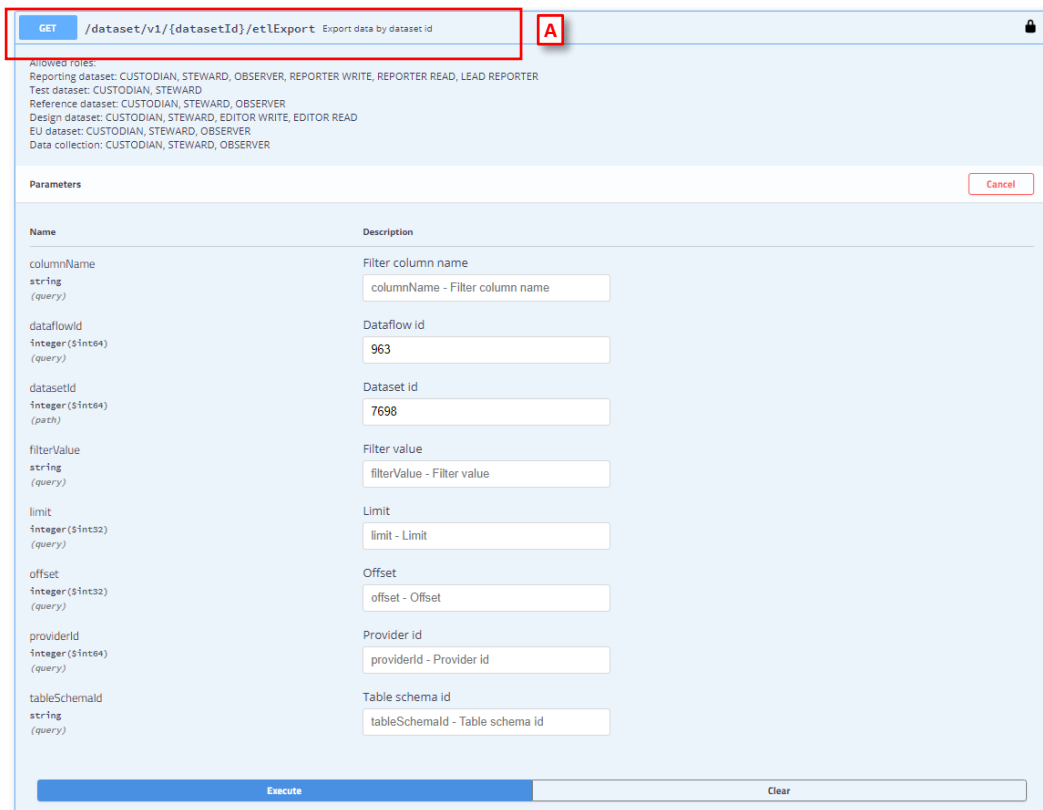
Responses

Code	Description
200	Successfully imported

If you go to the Reportnet page, you could see the record added for this dataset:



- Moreover, you can call to the GET service, where you should get a JSON response corresponding to all the rows and all the data in your schema. For this example, the microservice is dataset, and the GET service is `/dataset/v1/{datasetId}/etlExport [A]` and click **'Try it out'** button to see the parameters you need to fill to call the service.



- When ready, click on the blue button **'Execute'**. You should get a response of the service with a *passed* result and a response code 200 [A] and a response corresponding to all the rows and all the data in your schema



Responses Response content type: */*

Request URL
https://test-api.reportnet.europa.eu/dataset/v1/7698/etlExport?dataFlowId=963

Server response

Code 200

Details

Response body

```

{
  "tables": [
    {
      "tableName": "Table1",
      "records": [
        {
          "countryCode": null,
          "fields": [
            {
              "fieldName": "test",
              "value": "myTest",
              "field_value_id": null
            }
          ]
        }
      ]
    }
  ]
}
    
```

Response headers

```

access-control-allow-origin: *
cache-control: no-cache, no-store, max-age=0, must-revalidate
content-security-policy: default-src 'self' 'unsafe-inline' https://*.efonet.europa.eu
content-type: application/json
date: Fri, 03 Dec 2021 10:03:55 GMT
expires: 0
feature-policy: camera 'self'; microphone 'self'
pragma: no-cache
referrer-policy: no-referrer
strict-transport-security: max-age=31536000; includeSubDomains; preload;
transfer-encoding: chunked
x-content-type-options: nosniff
x-frame-options: DENY
x-ssr-protection: jsaode=block
    
```

Responses

Code	Description
200	Successfully exported

1.1.4.2 How to delete records programmatically using the API

This example consists of delete records on a dataset using the API.

For example, on Reportnet we have configured a dataset with one table, and we want to delete the dataset data. On the Reportnet page, you could click on “Delete dataset data” [A] to delete the data.

Reportnet 3 > Dataflows > Dataflow > Dataset designer

Dataset description

Dataset description 0/10000

A Import dataset data Export dataset data **Delete dataset data** Validate Show validations QC Rules Unique constraints External integrations Dashboards Manage copies Refresh

Design Tabular data

Table1

Table description

Table description 0/10000

Read only table
 Prefilled
 Fixed number of records
 Mandatory table

Import table data Export table data Delete table data Show/Hide columns Validation filter Filter by value

Actions	Validations	test
		myTest

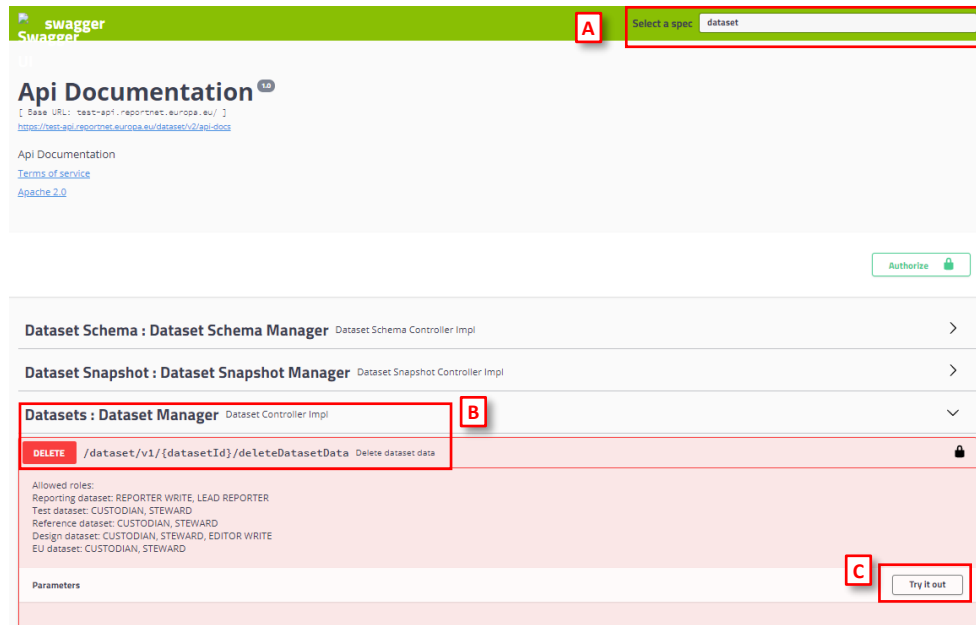
Rows per page: 10 1 of 1 Go to 1 of 1 Total: 1 record

+ Add record Paste records

You could do this in Swagger calling the DELETE API method.



1. The Reportnet API is enabled for a DELETE method on a dataset.
2. Before you run the service you must be authenticated in Swagger with the API-Key authorization explained in section 1.1.2
3. First, you must select the appropriate microservice. For this example, dataset [A] and the proper service [B] and click 'Try it out' button [C] to see the parameters you need to fill to call the service.

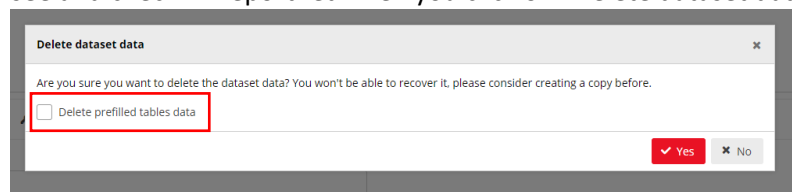


4. You must fill the dataflow Id [A], dataset Id [B] and mark as true or false if you want to delete the prefilled tables [C].

- i. The **dataflowId** and **datasetId** you will get them from the URL of the dataset. Go to the dataset schema page in your dataflow you wish to use the API on to get the dataflowId and datasetId e.g. highlighted in yellow:

<https://rn3test.eionet.europa.eu/dataflow/963/datasetSchema/7698?tab=61a9e86b8b08ce0001bf4072&view=tabularData>

- ii. The *deletePrefilledTable* parameter [C] is the same parameter you could see and check in Reportnet when you click on "Delete dataset data"





Datasets : Dataset Manager Dataset Controller Impl

DELETE /dataset/v1/{datasetId}/deleteDatasetData Delete dataset data

Allowed roles:
 Reporting dataset: REPORTER WRITE, LEAD REPORTER
 Test dataset: CUSTODIAN, STEWARD
 Reference dataset: CUSTODIAN, STEWARD
 Design dataset: CUSTODIAN, STEWARD, EDITOR WRITE
 EU dataset: CUSTODIAN, STEWARD

Parameters Cancel

Name A Description

dataflowid B Dataflow id
 integer(\$int64) (query)

datasetid C Dataset id
 integer(\$int64) (path)

deletePrefilledTables Delete prefilled tables
 boolean (query)

providerid Provider id
 integer(\$int64) (query)

Execute

5. When ready, click on the blue button 'Execute'

Responses Response content type: */*

Curl

```
curl -X DELETE "https://test-api-reportnet.europa.eu/dataset/v1/7698/deleteDatasetData?dataflowId=963&deletePrefilledTables=false" -H "accept: */*" -H "Authorization: ApiKey 532778e-8972-42e3-bbda-9e0f4ad7932f"
```

Request URL

```
https://test-api-reportnet.europa.eu/dataset/v1/7698/deleteDatasetData?dataflowId=963&deletePrefilledTables=false
```

Server response

Code	Details
200	<p>Response headers</p> <pre>access-control-allow-origin: * cache-control: no-cache, no-store, max-age=0, must-revalidate content-security-policy: default-src 'self' 'unsafe-inline' https://*.efonet.europa.eu date: Fri, 05 Dec 2023 10:11:25 GMT expires: 0 feature-policy: camera 'self'; microphone 'self' pragma: no-cache referrer-policy: no-referrer strict-transport-security: max-age=31536000; includeSubDomains; preload; transfer-encoding: chunked vary: Origin, Access-Control-Request-Method, Access-Control-Request-Headers x-content-type-options: nosniff x-frame-options: DENY x-xss-protection: 1;mode=block</pre>

Responses

Code	Description
200	Successfully deleted

6. You should get a response of the service with a passed result and a response code 200.

7. If you go to the Reportnet page, the records have been deleted.

Reportnet 3.0 > Datasets > Dataset designer Test Custodian

Dataset description

Dataset description Reference dataset Available in public view Configure webform

Table1 Read only table

Table description

Table description Prefilled

Actions Validations

Rows per page: 10 Total: 0 records