

MyGeodata Cloud API Manual

MyGeodata Cloud API allows users to call [Dataset Info](#) and [Convert](#) function using POST request and in some cases also GET request is available.

Dataset Info

Returns JSON object (application/json) describing geospatial data that user sent by the API, and that are supported by MyGeodata Cloud.

Parameters

API URL: <https://mygeodata.cloud/api/datasetinfo>

Required parameters:

- **srcurl** or **FILE OBJECT**
 - srcurl** - URL of dataset - e.g: <https://mygeodata.cloud/examples/data/NY.geojson>. In this case it is possible to use both GET and POST request.
 - FILE OBJECT** - binary file data sent by POST request.

Example 1

GET request using **srcurl** parameter (e.g. using web browser URL box):

```
https://mygeodata.cloud/api/datasetinfo?srcurl=https://mygeodata.cloud/examples/data/NY.geojson
```

Example 2

POST request using **srcurl** parameter; using **curl** command line command:

```
curl --data "srcurl=https://mygeodata.cloud/examples/data/NY.geojson" https://mygeodata.cloud/api/datasetinfo
```

Example 3

POST request using **FILE OBJECT**; using **curl** command line command:

```
curl -F "file=@NY.geojson" https://mygeodata.cloud/api/datasetinfo
```

or for more dataset files:

```
curl -F "file1=@NY.shp" -F "file2=@NY.shx" -F "file3=@NY.dbf" -F "file4=@NY.prj" https://mygeodata.cloud/api/datasetinfo
```

Example 4

POST request using **FILE OBJECT**; using **Python** code:

```
import requests
url = 'https://mygeodata.cloud/api/datasetinfo'
files = {'file': open('NY.geojson', 'rb')}
r = requests.post(url, files=files)
print(r.text)
```

Convert dataset

Converts geospatial dataset sent by user to another file format and/or transform to desired coordinate system. If dataset is sent as an URL link and the result is also required as URL link, GET request may be used, otherwise POST request must be used.

Parameters

API URL: <https://mygeodata.cloud/api/convert>

Required parameters:

- **srcurl** or **FILE OBJECT**
srcurl - URL of dataset - e.g: <https://mygeodata.cloud/examples/data/NY.geojson>.
FILE OBJECT - binary file data sent by POST request.
- **outform** - [url / binary] Use **url** if you would like to return the result as an URL link to download. Otherwise use **binary** to get binary file data. If there is only one resulting file, it will be returned as is, otherwise resulting files will be packed to a ZIP file and sent as a ZIP binary file. Please note that in case of using **outform=url**, the result will be deleted after first download of the returned link.

Optional parameters:

- **key** - user API key string assigned at <https://mygeodata.cloud/dashboard/profile>. If no API key provided, conversions will be limited according to Instant plan. Otherwise limits according to appropriate user's plan will be applied. See <https://mygeodata.cloud/plans>.
- **format** - format codes to which the dataset should be converted. If no format parameter is provided, no conversion will be applied. Available codes: shp, kml, kmz, geojson, gml, gpx, mapinfo, dgn, dxf, gpkg, sqlite, csv, ods, xlsx
- **incrs** - EPSG code or proj4 text of input dataset coordinate reference system. This parameter is usually recognized by MyGeodata Cloud at it is not necessary to send it, especially if no coordinate transformation is required.
- **outcrs** - EPSG code or proj4 text of required output dataset coordinate reference system. Input dataset will be transformed to this coordinate reference system. If no outcrs parameter is provided, no transformation will be applied.
- **layers** - selection of input layers to export to the output dataset, defined as layers names, separated by coma. If no layers defined, all input layers will be exported.

Example 1

GET request using **srcurl** and **outform=url** parameters (e.g. using web browser URL box):

```
https://mygeodata.cloud/api/convert?outform=url&format=shp&outcrs=EPSG:3857&srcurl=https://mygeodata.cloud/examples/data/NY.geojson
```

Example 2

POST request using **FILE OBJECT** and url as an output; using **curl** command line command:

```
curl -F "key=exampleKey" -F "format=shp" -F "outcrs=EPSG:3857" -F "outform=binary" -F "file=@NY.geojson" https://mygeodata.cloud/api/convert
```

Example 3

POST request using **FILE OBJECT**; using **Python** code:

```
import requests
import re
url = 'https://mygeodata.cloud/api/convert'
files = {'file': open('NY.geojson', 'rb')}
data = {'format': 'shp', 'outcrs': 'EPSG:3857', 'outform': 'binary', \
        'key': 'exampleKey'}
r = requests.post(url, files=files, data=data)
d = r.headers['content-disposition']
fname = re.findall("filename=(.+)", d)[0]
with open(fname, 'wb') as fd:
    for chunk in r.iter_content(1000):
        fd.write(chunk)
```

JavaScript code:

```
function test() {
    var formData = new FormData();

    formData.append('srcurl',
        'https://mygeodata.cloud/examples/data/NY.geojson');
    formData.append('outform', 'url');
    formData.append('key', 'YOUR_API_KEY');
    formData.append('outcrs', 'EPSG:3857');
    formData.append('format', 'shp');

    $.ajax({
        type: 'POST',
        url: 'https://mygeodata.cloud/api/convert',
        data: formData,
        processData: false,
        contentType: false,
        crossOrigin: true,
        success: function (response) {
            console.log(response);
        },
        error: function (e) {
            console.log("ERROR");
        }
    });
}
```

C# code:

```
var formData = new MultipartFormDataContent();
var fileContent = new StreamContent(File.OpenRead(zip_file_path));
formData.Add(fileContent, "file", zip_file_path);
formData.Add(new StringContent("shp"), "format");
formData.Add(new StringContent("EPSG:3857"), "outcrs");
formData.Add(new StringContent("binary"), "outform");
formData.Add(new StringContent("your API key"), "key");
```