StatBank API user guide

Content

1. Introduction

Statistics Norway now offer an API for queries towards all StatBank's 6000 tables. You can grab the entire table or parts of a table. This is an addition to the former API with readymade data sets on https://data.ssb.no/api/v0/dataset/?lang=en.

To use the API, it is an advantage to be familiar with creating tables in StatBank and the http protocol. This API uses http POST, not http GET. Queries can be formulated in JSON (JavaScript Object Notation).

URL is composed as follows: http://*data.ssb.no/api-name/api-version/lang/*

There are two ways to find tables in the new API:
1. Via console - https://data.ssb.no/api/v0/en/console/
2. Via URL - https://data.ssb.no/api/v0/en/table/

We use the Norwegian license for public data, NLOD, which is compatible to CC4-BY. We welcome your feedback. Send message about errors and shortcomings as well as questions and wishes to: statistikkbanken@ssb.no

2. Starting the API

API console

A console assists the user to generate and edit the code to be used in the API. It is intended to test queries towards the API and not in production. See also a short video introduction to the console.

Search table in console
In the Table pane: Enter table number or search text. Autocomplete turns on after 3 characters. Then click the selected table.

Once you've selected a table, the console will generate an example query in JSON with the first three values for each variable.

API console

Console parts

1. **Search** table by table number or text in title
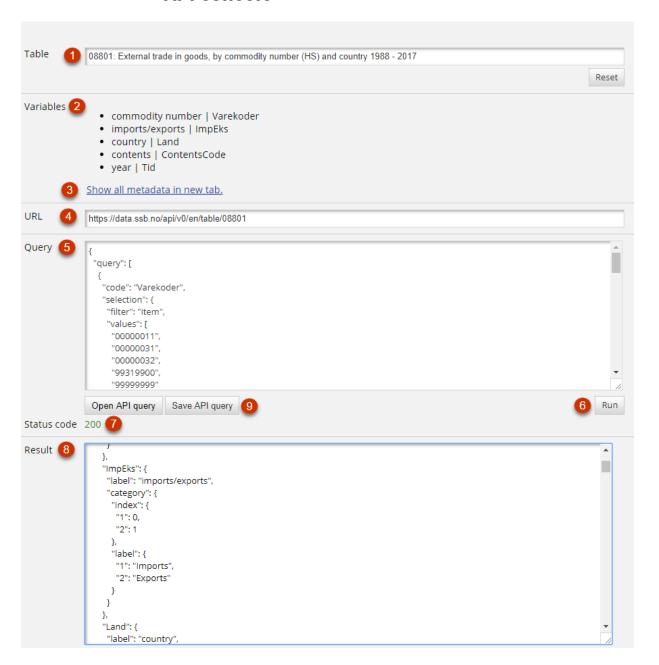2. Metadata overview for the table, shows variables
3. Link to all metadata - opens in new tab (see: 2)
4. POST URL. The url you post your query
5. Query. Window where you can edit the generated query in JSON (see: 4)
6. Run - POST query
7. Http status code - 200 (green) if OK
8. Result. The format JSON-stat, version 2 is default. The content shows in the result window when JSON-stat is selected. Downloaded file opens in a separate application, such as Excel.
9. Save og open API query (.json)

3. JSON query

All queries start with the object: {query: [{...}]}. The query is posted to the table URL.

Filters supported:
- **Item** - single values in a list (item is default in console)
- **All** - wildcard *, e.g. only '*' returns all values. '2018*' gives all values starting with 2018

- **Top** - output the number of the latest values

An important characteristic of the filters is to make the query smaller and more transparent. With the filters All and Top, it is possible to construct a rolling query. Then the same query can be used to retrieve new figures.

Output response formats

The API can from 2019 return results in 6 formats[1]:
- JSON-stat, version 2 (default)
- CSV2 - pivot-friendly format
- CSV3 - as csv2 with codes instead of text
- XLSX
- JSON-stat v. 1.2
- CSV

Decimal sign is. (period).

---

[1] Until June 2016 we also supported the formats PX (PC-Axis) and Statistics Sweden's PX-JSON format. Due to minor bugs these are no longer supported, but available in the API, as "px" and "json" Outside Console it is also possible to specify SDMX, but this is very limited and not supported.

If response part is missing in query, the API returns JSON-stat.

4. Post query

The API uses HTTP POST, and does not support http GET. In the console you post your query by clicking "Run".

Status codes and error messages:
200 - OK
404 - errors in syntax of the query or POST URL not found.
403 - blocking when querying for large data sets. The API limit is now 800,000 cells.
429 - Too many queries within a minute. The API limit was 13.02.2020 changed back to 30 queries within 60 seconds
503 - Time-out after 60 seconds. It may turn on, when extracting large XLSX datasets. (We do not recommend that).

5. Navigate API using URL

a. Table number
If you know the table number in StatBank for the current table, you can use the URL on the form:
https://data.ssb.no/api/v0/en/table/(*5-digit table number*)
Example: https://data.ssb.no/api/v0/en/table/11000

GET shows table's metadata as JSON. Use the same URL to post the query.

b. Hierarchy
https://data.ssb.no/api/v0/en/table/
retrieves main topics in JSON format from topic hierarchy.

```
[{"id":"al","type":"l","text":"Labour market and earnings"},
{ "id":"bf","type":"l","text":"Banking and financial markets"},
{ "id":"be","type":"l","text":"Population"},
{ "id":"bb","type":"l","text":"Construction, housing and property"},
{ "id":"ei","type":"l","text":"Energy and manufacturing"},
{ "id":"he","type":"l","text":"Health"},
{ "id":"if","type":"l","text":"Income and consumption"},
{ "id":"in","type":"l","text":"Immigration and immigrants"},
{ "id":"js","type":"l","text":"Agriculture, forestry, hunting and fishing"},
{ "id":"kf","type":"l","text":"Culture and recreation"},
{ "id":"nk","type":"l","text":"National accounts and business cycles"},
{ "id":"nm","type":"l","text":"Nature and the environment"},
{ "id":"os","type":"l","text":"Public sector"},
{ "id":"pp","type":"l","text":"Prices and price indices"},
{ "id":"sk","type":"l","text":"Social conditions, welfare and crime"},
{ "id":"sv","type":"l","text":"Svalbard"},
{ "id":"ti","type":"l","text":"Technology and innovation"},
{ "id":"tr","type":"l","text":"Transport and tourism"},
{ "id":"ud","type":"l","text":"Education"},
{ "id":"ut","type":"l","text":"External economy"},
{ "id":"va","type":"l","text":"Elections"},
{ "id":"vt","type":"l","text":"Wholesale and retail trade and service activities"},
{ "id":"vf","type":"l","text":"Establishments, enterprises and accounts"}]
```

In general the structure is: *topic - subtopic - statistics - table*. Statistics with many tables may also have one or two levels between statistics and table.
E.g. https://data.ssb.no/api/v0/en/table/be list all subtopics in the topic population.

Note that statistics can be placed under several topics and subtopics. All tables under subject id's "in" and "sv" will also be placed elsewhere. POST towards all such addresses will work.
Url 's in the form table number: https://data.ssb.no/api/v0/en/table/(*5-digit table number*)

over time is more stable than the url of the form:
https://data.ssb.no/api/v0/en/table/*(topic)/(subtopic)/(statistics)/(tablename).*
This is because the tables in some cases can be moved in the subject structure.

"type":
- "l" - sublevel
- "t" - table

Example from https://data.ssb.no/api/v0/en/table/be/be02/folkfram:

```
{
        "id": "Framskr2016T6",
        "type": "t",
        "text": "11172: Projected life expectancy, for men, women and both sexes combined, in 3
         variants 2015 - 2100",
        "updated": "2016-06-21T10:00:00"
}
```

At this level means the different tags:
- "id": tablename
- "type": t = Table
- "text": Table title (5-digit table number + description of contents + timespan)

Search tables with URL - ?query

? query - search variables and titles. The search is case-insensitive. Space gives AND. The filter "title" narrows your search to the title field. You can also narrow by using field name plus colon, eg. *title:* . The search engine is Lucene v. 2.9.4 (win).

Examples:
• Search for "cake mix" in all tables
https://data.ssb.no/api/v0/en/table/?query=cake mix

• Use table id to find table number:
https://data.ssb.no/api/v0/en/table/?query=folkframt6

• Search all tables within topic population for the value (municipality) "Hattfjelldal» using mask *
https://data.ssb.no/api/v0/en/table/be/?query=hattfjell*

• Search "item number" and "HS" less than 5 words apart
https://data.ssb.no/api/v0/en/table/?query="commodity number hs "~ 5

• Search urban district tables under area population
https://data.ssb.no/api/v0/en/table/be/?query=title:(UD)

• List all tables in Statbank.
https://data.ssb.no/api/v0/en/table/?query=*&filter=*

• Search tables published a specific date.
https://data.ssb.no/api/v0/en/table/?query=20180507*&filter=published

Mark that the characters "(), and spaces will be escaped by respectively. % 22,% 28,% 29 and% 20

6. Table metadata

Http GET to the URL you post to, will return main the table's metadata as JSON.
The metadata consists of a title "title" and a list of variables for the table. Variable objects have four attributes :

- Code (code) - mandatory
- Name (text) - mandatory
- Elimination (elimination)
- Time (time)

Example from https://data.ssb.no/api/v0/en/table/11172:

```
[
    title: "Projected life expectancy, by sex, age, variants and year",
  ▾ variables: [
    ▾ {
          code: "Kjonn",
          text: "sex",
        ▾ values: [
              "0",
              "1",
              "2"
          ],
        ▾ valueTexts: [
              "Both sexes",
              "Males",
              "Females"
          ],
          elimination: true
      },
    ▾ {
          code: "Alder",
          text: "age",
        ▾ values: [
              "000",
              "001",
              "002",
              "003"
```

Variable objects contain two lists, a list of codes and with presentation texts.
It is possible to remove a variable from the query. If elimination is true, a sum value will be returned. If not, all values will be returned.
Only one variable can have property "time" set to true.

Groupings and aggregations are missing in metadata. There are two more filters: Vs and Agg.

In StatBank below the finished table in API query for this table , it is possible to see these filters for your selection. Groupings can be e.g. within "Municipalities entire code list" the current municipality division: "Municipalities 2019" and aggregations can eg. be "5-year age groups".

7. Difference between API:Readymade datasets and API:Create your own datasets (PxWebApi)

In the PxWebApi users formulate their own queries towards 6000 different tables, while the API from 2013 serves readymade data sets. Towards these 250 readymade datasets you can use http GET. If you find your dataset on https://data.ssb.no/api/v0/dataset/?lang=en is an easy solution with stable URLs. It is possible to contact statistikkbanken@ssb.no to request new datasets.

CSV files are different

CSV in API with readymade datasets:
- One variable per line
- decimal separator: , (comma) in Norwegian and . (dot) in English
- field delimiter ; (semicolon)

CSV in API: Create your own datasets
- field delimiter comma,
- decimal separator . (dot)
- 3 different CSV formats. CSV2 is quite similar to the CSV in readymade datasets.

For a flexible treatment of CSV we recommend JSON-stat as output, and then the JSON-stat Command Line Conversion Tools (requires node.js)

8. Resource Pages

How to use StatBank

Before using the API users should be used to the StatBank:
https://www.ssb.no/en/informasjon/om-statistikkbanken/how-to-use-statbank-norway

Tables in StatBank can be accessed by the address https://www.ssb.no/en/table/*5-digit-table number*

E.g.: https://www.ssb.no/en/table/05000

We basically use the same API code as Statics Sweden (SCB), plus our own developed console. See https://www.scb.se/api_en/ and SCB's API guide.

JSON and JSON-Stat

Although API has six output formats, we recommend using JSON-stat version 2. This version is also the output from Eurostat, and all known libraries support this. See https://json-stat.org/ and the JSON-stat Toolkit.
There are available JSON-stat libraries for:
- Javascript
- Python,
- R
- Java
- Julia

Example code
Code examples for Statistics Norway's API using different libraries for JSON-Stat, R, Python, Highcharts with jquery, Excel etc., see: PxWebApi Code examples

In the browser, users should have a JSON viewer, In Firefox this is installed by default.
For Chrome(ium) try this extension: https://jsonview.com

http POST
As alternative to console we can recommend Postman https://www.getpostman.com

Curl

In production, a tool like Curl may be suitable for posting queries. See https://curl.haxx.se/.
If you save a query from the console to file, query.json. @ reads from file, the option –d or --data is used for http POST and –o to save the output, e.g.

```
> curl -d @query.json -X POST https://data.ssb.no/api/v0/en/table/08801 -o resultfile
```