

The Helsinki logo consists of the word "Helsinki" in a bold, sans-serif font, enclosed within a white speech bubble shape with a black outline.

How to use the City of Helsinki PxWeb API

The city of Helsinki provides a plenty of open data through its statistical databases. These databases are on a platform called PxWeb, which is an open source dissemination tool used for sharing tabular data.

In addition to the regular “clickable” user interface, the data can also be accessed via the PxWeb API (Application Programming Interface). Using the API, machine-readable data in various formats including CSV, JSON-stat and PX can be automatically retrieved and used e.g. for creating automated dataflows.

This documentation provides instructions for how to use the City of Helsinki PxWeb API.

Quick links

API address: <https://stat.hel.fi/api/v1/fi/>

API limitations: <https://stat.hel.fi/api/v1/fi/?config>

PxWeb user interface: <https://stat.hel.fi/pxweb/fi/>

Documentation and more: <https://stat.hel.fi>

Before you jump in

Before using the service, you should take a look at Statistics Sweden’s (SCB) API documentation*: https://scb.se/contentassets/79c32c72783a4f67b202ad3189f921b9/api_description.pdf
<https://scb.se/en/services/oppna-data/api-for-the-statistical-database/>

Other useful resources include the PxWeb API documentation of Statistics Finland: <https://pxnet2.stat.fi/api1.html>

**NB! SCB’s description is based on the use of a relational database where you do not have to use a PX extension in the table name. By contrast, in the databases maintained by the city of Helsinki you must always remember to use the PX extension in the table name, similarly as with the databases by Statistics Finland.*

Contact

Questions or feedback you would like to share? Send us an email: Aluesarjat@hel.fi

Additional contact information can be found from stat.hel.fi.



API Structure

The main structure of the API is as follows:

PXWEB/API-NAME/API-VERSION/LANGUAGE/DATABASE-ID/LEVELS/TABLE-ID

Below is a breakdown of the structure including examples:

Available databases: <https://stat.hel.fi/api/v1/fi>

Structure of a database (case Aluesarjat): [https://stat.hel.fi/api/v1/fi/Aluesarjat/?query=*&filter=*](https://stat.hel.fi/api/v1/fi/Aluesarjat/?query=*&filter=)

Topics (levels) in the Aluesarjat database: <https://stat.hel.fi/api/v1/fi/Aluesarjat>

Levels related to Population (vrm): <https://stat.hel.fi/api/v1/fi/Aluesarjat/vrm>

Levels related to Population/Families (perh): <https://stat.hel.fi/api/v1/fi/Aluesarjat/vrm/perh>

List of tables: <https://stat.hel.fi/api/v1/fi/Aluesarjat/vrm/perh/pksoa>

Table metadata: https://stat.hel.fi/api/v1/fi/Aluesarjat/vrm/perh/pksoa/A01S_HKI_Perhetyypit.px

Tip! A short version of both GET and POST requests can also be used:

https://stat.hel.fi/api/v1/fi/Aluesarjat/A01S_HKI_Perhetyypit.px

Limitations

You can request up to 100 000 cells at a time. In case you need a larger output, split/loop your query. You can find the limitations here: <https://stat.hel.fi/api/v1/fi/?config>

Free text search

The PxWeb API supports free text search with URL commands.

Example: Search for tables about families

<https://stat.hel.fi/api/v1/fi/Aluesarjat?query=Perheet>

Example: Search for tables in the housing section, where Espoo is mentioned

<https://stat.hel.fi/api/v1/fi/Aluesarjat/asu?query=Espoo>



Example: Get the API Query for a table

- 1) Go to <https://stat.hel.fi/pxweb/fi/> and select a database and table you wish to use



Helsingin seudun avoimet tilastotietokannat

[Svenska](#)
[English](#)
[/ Aluesarjat](#)


Valitse taulukko



Valitse muuttujat



Näytä taulukko

Aluesarjat

Etsi tietokannasta Aluesarjat:

Etsi

 Asuminen

 Asuinolot (asuntokunnat)

 Asuntokanta

 Kunnittain (Helsingin seutu/Uusimaa)

 Uudenmaan asunnot talotyypin ja valmistumisvuoden mukaan 31.12.2010- [Koko: 100 K] [Päivitetty: 3.11.2021]

 Uudenmaan asunnot hallintaperusteen ja huoneistotyypin mukaan 31.12.2003- [Koko: 250 K] [Päivitetty: 4.11.2021]

- 2) Select variables and continue to the table view

Talotyyppi Pakollinen*

 Valitse kaikki

 Poista valinnat

Valittu 1 Yhteensä 5

 Kaikki talotyypit

 Omakoti- ja paritalot

 Rivitalot

 Kerrostalot

 Muut rakennukset

Näytä taulukko

- 3) Click on API Query for this table on the bottom of the page to open the URL and JSON query string:



Uudenmaan asunnot talotyyppin ja valmistumisvuoden mukaan 31.12.2010-

Taulukko

- ✓ Tietoja taulukosta
- ✓ Näytä tiedot muodossa...
- ✓ Muuta ja laske
- ✓ Tallenna tiedot muodossa...
- ✓ Tallenna poiminta
- ✓ Taulukon asetukset

↻ Käännä manuaalisesti

↻ Käännä myötöpäivään

↻ Käännä vastapäivään

↗ Koko näyttö

	2018	2019	2020
	Yhteensä		
	Kaikki talotyytit	Kaikki talotyytit	Kaikki talotyytit
Helsinki	367 680	371 295	376 065
Espoo	135 689	139 518	142 305
Vantaa	116 233	120 755	122 536

- ✓ Alaviitteet
- ✓ Lisää taulukko sovellukseesi

- 4) Copy the URL and JSON query to make a POST-request.

Liittääksesi taulukon sovellukseesi, lähetä seuraava JSON query-koodi alla olevaan URL-osoitteeseen POST-komennolla.

URL:

https://stat.hel.fi:443/api/v1/fi/Aluesarjat/asu/askan/umkun/01UM_Asuntokanta_rakennusvuosi.px

JSON query:

```
{
  "query": [
    {
      "code": "Alue",
      "selection": {
        "filter": "item",
        "values": [
          "091",
          "049",
          "092"
        ]
      }
    },
    {
      "code": "Vuosi",
      "selection": {
        "filter": "item",
        "values": [
          "2018",
          "2019",
        ]
      }
    }
  ]
}
```

Tallenna API-kysely (json)

Lisätietoja



Query structure and basic functionalities

The PxWeb query consists of two main parts: 1) Code and 2) Selection. Code stands for the variable codes and selection defines the selected values from the variable. In addition to the values, selection supports also various filters, which are discussed in the next section.

Example:

```
{
  "query": [
    {
      "code": "Alue",
      "selection": {
        "filter": "item",
        "values": [
          "091",
          "049",
          "092"
        ]
      }
    },
    {
      "code": "Vuosi",
      "selection": {
        "filter": "item",
        "values": [
          "2018",
          "2019",
          "2020"
        ]
      }
    }
  ],
  "response": {
    "format": "json-stat"
  }
}
```

You can leave out a variable to get all values from the variable (within API limitations) unless elimination is set to true for the variable. Whether elimination is set to true or false can be checked e.g. from the table metadata behind the table URL. If elimination has been set to true for a given variable, "elimination": true should be stated after the variable.

If elimination is set to true and the elimination value has been defined, the predefined elimination value is returned for the given variable. If elimination is set to true, but is missing a predefined elimination value, will all values in the variable be aggregated. If elimination is set to false, all values are selected for the variable.



Filters

PxWeb API supports filters, which can be used to retrieve values with user-defined criteria.

The supported filter types are the following:

- 1) **Item.** The filter lists valid variable values. There can be one or several filtered values, for example, years 2003, 2019 and 2020, where years 2004-2018 are left out.

Example: Get years 2019, 2020 and 2003

```
{
  "code": "Vuosi",
  "selection": {
    "filter": "item",
    "values": [
      "2003",
      "2019",
      "2020"
    ]
  }
}
```

- 2) **Top.** Shows the first n values or the latest n values. The latter is possible, if a time variable has been defined in the table metadata (element called timeval. This can be checked e.g. from the table metadata behind the table URL. If a time variable has been defined, "time": true should be stated after the variable. See e.g.

https://stat.hel.fi/api/v1/fi/Aluesarjat/asu/askan/umkun/01UM_Asuntokanta_rakennusvuosi.px

Example: get last three years from a table

```
{
  "code": "Vuosi",
  "selection": {
    "filter": "top",
    "values": [
      "3"
    ]
  }
}
```

- 3) **All.** This filter uses the wildcard feature (note! only one of these can be used at a time). For example, 01* shows all values that begin with 01, * shows all values.

Example: get all years that end with "5", such as 1995, 2005 and 2015.

```
{
```



```
"code": "Vuosi",  
"selection": {  
  "filter": "all",  
  "values": [  
    "*5"  
  ]  
}  
}
```