

# Package ‘PxWebApiData’

January 20, 2025

**Type** Package

**Title** PX-Web Data by API

**Version** 1.0.0

**Date** 2024-11-01

**Maintainer** Øyvind Langsrud <oyl@ssb.no>

**Imports** httr, rjstat, jsonlite, pxweb

**VignetteBuilder** knitr

**Suggests** knitr, rmarkdown, testthat

## Description

Function to read PX-Web data into R via API. The example code reads data from the three national statistical institutes, Statistics Norway, Statistics Sweden and Statistics Finland.

**License** MIT + file LICENSE

**Encoding** UTF-8

**URL** <https://statisticsnorway.github.io/ssb-pxwebapidata/>,  
<https://github.com/statisticsnorway/ssb-pxwebapidata>

**BugReports** <https://github.com/statisticsnorway/ssb-pxwebapidata/issues>

**RoxygenNote** 7.3.1

**NeedsCompilation** no

**Author** Øyvind Langsrud [aut, cre],  
Jan Bruusgaard [aut],  
Solveig Bjørkholt [ctb],  
Susie Jentoft [ctb]

**Repository** CRAN

**Date/Publication** 2024-11-01 16:10:02 UTC

## Contents

ApiData . . . . .	2
<b>Index</b>	<b>7</b>

**Description**

A function to read PX-Web data into R via API. The example code reads data from the three national statistical institutes, Statistics Norway, Statistics Sweden and Statistics Finland.

**Usage**

```
ApiData(  
  urlToData,  
  ...,  
  getDataByGET = FALSE,  
  returnMetaData = FALSE,  
  returnMetaValues = FALSE,  
  returnMetaFrames = FALSE,  
  returnApiQuery = FALSE,  
  defaultJSONquery = c(1, -2, -1),  
  verbosePrint = FALSE,  
  use_factors = FALSE,  
  urlType = "SSB",  
  apiPackage = "httr",  
  dataPackage = "rjstat",  
  returnDataSet = NULL,  
  makeNAstatus = TRUE,  
  responseFormat = "json-stat2"  
)  
  
GetApiData(..., getDataByGET = TRUE)  
  
pxwebData(..., apiPackage = "pxweb", dataPackage = "pxweb")  
  
PxData(..., apiPackage = "pxweb", dataPackage = "rjstat")  
  
ApiData1(..., returnDataSet = 1)  
  
ApiData2(..., returnDataSet = 2)  
  
ApiData12(..., returnDataSet = 12)  
  
GetApiData1(..., returnDataSet = 1)  
  
GetApiData2(..., returnDataSet = 2)  
  
GetApiData12(..., returnDataSet = 12)
```

```

pxwebData1(..., returnDataSet = 1)
pxwebData2(..., returnDataSet = 2)
pxwebData12(..., returnDataSet = 12)
PxData1(..., returnDataSet = 1)
PxData2(..., returnDataSet = 2)
PxData12(..., returnDataSet = 12)

```

### Arguments

<code>urlToData</code>	url to data or id of SSB data
<code>...</code>	specification of JSON query for each variable
<code>getDataByGET</code>	When TRUE, readymade dataset by GET
<code>returnMetaData</code>	When TRUE, metadata returned
<code>returnMetaValues</code>	When TRUE, values from metadata returned
<code>returnMetaFrames</code>	When TRUE, values and valueTexts from metadata returned as data frames
<code>returnApiQuery</code>	When TRUE, JSON query returned
<code>defaultJSONquery</code>	specification for variables not included in ...
<code>verbosePrint</code>	When TRUE, printing to console
<code>use_factors</code>	Parameter to <code>fromJSONstat</code> defining whether dimension categories should be factors or character objects.
<code>urlType</code>	Parameter defining how url is constructed from id number. Currently two Statistics Norway possibilities: "SSB" (Norwegian) or "SSBen" (English)
<code>apiPackage</code>	Package used to capture json(-stat) data from API: "httr" (default) or "pxweb"
<code>dataPackage</code>	Package used to transform json(-stat) data to data frame: "rjstat" (default) or "pxweb"
<code>returnDataSet</code>	Possible non-NULL values are 1, 2 and 12. Then a single data set is returned as a data frame. <ul style="list-style-type: none"> <li>• 1: The first data set</li> <li>• 2: The second data set</li> <li>• 12: Both data sets combined</li> </ul>
<code>makeNAstatus</code>	When TRUE and when dataPackage is "rjstat" and when missing entries in value, the function tries to add an additional variable, named NAstatus, with status codes.
<code>responseFormat</code>	Response format to be used when apiPackage and dataPackage are defaults ("json-stat" or "json-stat2").

## Details

Each variable is specified by using the variable name as input parameter. The value can be specified as: TRUE (all), FALSE (eliminated), imaginary value (top), variable indices, original variable id's (values) or variable labels (valueTexts). Reversed indices can be specified as negative values. Indices outside the range are removed. Variables not specified is set to the value of defaultJSONquery whose default means the first and the two last elements.

The value can also be specified as a (unnamed) two-element list corresponding to the two query elements, filter and values. In addition it possible with a single-element list. Then filter is set to 'all'. See examples.

A comment attribute with elements label, source and updated is added to output as a named character vector. When available, the elements tableid and contents are also included, resulting in a vector with 3 to 5 elements. Run [comment](#) to obtain this information.

Functionality in the package pxweb can be utilized by making use of the parameters apiPackage and dataPackage as implemented as the wrappers PxData and pxwebData. With data sets too large for ordinary downloads, PxData can solve the problem (multiple downloads). When using pxwebData, data will be downloaded in px-json format instead of json-stat and the output data frame will be organized differently (ContentsCode categories as separate variables).

## Value

list of two data sets (label and id)

## Note

See the package vignette for aggregations using filter agg.

## Examples

```
##### Readymade dataset by GET. Works for readymade datasets and "saved-JSON-stat-query-links".
x <- ApiData("https://data.ssb.no/api/v0/dataset/1066.json?lang=en", getDataByGET = TRUE)
x[[1]] # The label version of the data set
x[[2]] # The id version of the data set
names(x)
comment(x)

##### As above with single data set output
url <- "https://data.ssb.no/api/v0/dataset/1066.json?lang=en"
x1 <- ApiData1(url, getDataByGET = TRUE) # as x[[1]]
x2 <- ApiData2(url, getDataByGET = TRUE) # as x[[2]]
ApiData12(url, getDataByGET = TRUE) # Combined

##### Special output
ApiData("https://data.ssb.no/api/v0/en/table/11419", returnMetaData = TRUE) # meta data
ApiData("https://data.ssb.no/api/v0/en/table/11419", returnMetaValues = TRUE) # meta data values
ApiData("https://data.ssb.no/api/v0/en/table/11419", returnMetaFrames = TRUE) # list of data frames
ApiData("https://data.ssb.no/api/v0/en/table/11419", returnApiQuery = TRUE) # query using defaults

##### Ordinary use      (makeNAstatus is in use in first two examples)
```

```

# NACE2007 as imaginary value (top 10), ContentsCode as TRUE (all), Tid is default
x <- ApiData("https://data.ssb.no/api/v0/en/table/11419", NACE2007 = 10i, ContentsCode = TRUE)

# Two specified and the last is default (as above) - in Norwegian change en to no in url
x <- ApiData("https://data.ssb.no/api/v0/no/table/11419", NACE2007 = 10i, ContentsCode = TRUE)

# Number of residents (bosatte) last year, each region
x <- ApiData("https://data.ssb.no/api/v0/en/table/04861", Region = TRUE,
  ContentsCode = "Bosatte", Tid = 1i)

# Number of residents (bosatte) each year, total
ApiData("https://data.ssb.no/api/v0/en/table/04861", Region = FALSE,
  ContentsCode = "Bosatte", Tid = TRUE)

# Some years
ApiData("https://data.ssb.no/api/v0/en/table/04861", Region = FALSE,
  ContentsCode = "Bosatte", Tid = c(1, 5, -1))

# Two selected regions
ApiData("https://data.ssb.no/api/v0/en/table/04861", Region = c("1103", "0301"),
  ContentsCode = 2, Tid = c(1, -1))

##### Using id instead of url, unnamed input and verbosePrint
ApiData(4861, c("1103", "0301"), 1, c(1, -1)) # same as below
ApiData(4861, Region = c("1103", "0301"), ContentsCode=2, Tid=c(1, -1))
names(ApiData(4861,returnMetaFrames = TRUE)) # these names from metadata assumed two lines above
ApiData("4861", c("1103", "0301"), 1, c(1, -1), urlType="SSBen")
ApiData("01222", c("1103", "0301"), c(4, 9:11), 2i, verbosePrint = TRUE)
ApiData(1066, getDataByGET = TRUE, urlType="SSB")
ApiData(1066, getDataByGET = TRUE, urlType="SSBen")

##### Advanced use using list. See details above. Try returnApiQuery=TRUE on the same examples.
ApiData(4861, Region = list("03*"), ContentsCode = 1, Tid = 5i) # "all" can be dropped from the list
ApiData(4861, Region = list("all", "03*"), ContentsCode = 1, Tid = 5i) # same as above
ApiData(04861, Region = list("item", c("1103", "0301")), ContentsCode = 1, Tid = 5i)

##### Using data from SCB to illustrate returnMetaFrames
urlSCB <- "https://api.scb.se/OV0104/v1/doris/sv/ssd/BE/BE0101/BE0101A/BefolkningNy"
mf <- ApiData(urlSCB, returnMetaFrames = TRUE)
names(mf) # All the variable names
attr(mf, "text") # Corresponding text information as attribute
mf$ContentsCode # Data frame for the fifth variable (alternatively mf[[5]])
attr(mf,"elimination") # Finding variables that can be eliminated
ApiData(urlSCB, # Eliminating all variables that can be eliminated (line below)
  Region = FALSE, Civilstand = FALSE, Alder = FALSE, Kon = FALSE,
  ContentsCode = "BE0101N1", # Selecting a single ContentsCode by text input
  Tid = TRUE) # Choosing all possible values of Tid.

```

```
##### Using data from Statfi to illustrate use of input by variable labels (valueTexts)
urlStatfi <- "https://pxdata.stat.fi/PXWeb/api/v1/en/StatFin/kuol/statfin_kuol_pxt_12au.px"
ApiData(urlStatfi, returnMetaFrames = TRUE)$Tiedot
ApiData(urlStatfi, Alue = FALSE, Vuosi = TRUE, Tiedot = "Population") # same as Tiedot = 21

##### Wrappers PxData and pxwebData

# Exact same output as ApiData
PxData(4861, Region = "0301", ContentsCode = TRUE, Tid = c(1, -1))

# Data organized differently
pxwebData(4861, Region = "0301", ContentsCode = TRUE, Tid = c(1, -1))

# Large query. ApiData will not work.
if(FALSE){ # This query is "commented out"
  z <- PxData("https://api.scb.se/OV0104/v1/doris/sv/ssd/BE/BE0101/BE0101A/BefolkningNy",
             Region = TRUE, Civilstand = TRUE, Alder = 1:10, Kon = FALSE,
             ContentsCode = "BE0101N1", Tid = 1:10, verbosePrint = TRUE)
}

##### Small example where makeNAstatus is in use
ApiData("04469", Tid = "2020", ContentsCode = 1, Alder = TRUE, Region = "3011")
```

# Index

ApiData, [2](#)

ApiData1 (ApiData), [2](#)

ApiData12 (ApiData), [2](#)

ApiData2 (ApiData), [2](#)

comment, [4](#)

fromJSONstat, [3](#)

GetApiData (ApiData), [2](#)

GetApiData1 (ApiData), [2](#)

GetApiData12 (ApiData), [2](#)

GetApiData2 (ApiData), [2](#)

PxData (ApiData), [2](#)

PxData1 (ApiData), [2](#)

PxData12 (ApiData), [2](#)

PxData2 (ApiData), [2](#)

pxwebData (ApiData), [2](#)

pxwebData1 (ApiData), [2](#)

pxwebData12 (ApiData), [2](#)

pxwebData2 (ApiData), [2](#)