

Package ‘wbstats’

August 28, 2026

Type Package

Title Programmatic Access to Data and Statistics from the World Bank API

Version 1.2

Description Search and download data from the World Bank Data API. Includes support for mutiple languages, access to annual, quarterly, and monthly data.

License Apache License 2.0

URL <https://github.com/pachadotdev/wbstats>

BugReports <https://github.com/pachadotdev/wbstats/issues>

Depends R (>= 3.5)

Imports data.table, httr, jsonlite

Suggests litedown, tinytest, vcr,

Encoding UTF-8

LazyData TRUE

VignetteBuilder litedown

NeedsCompilation no

Author Mauricio Vargas Sepulveda [aut, cre] (ORCID: <https://orcid.org/0000-0003-1017-7574>),
Jesse Piburn [aut] (ORCID: <https://orcid.org/0000-0003-4967-7912>),
The World Bank [dct]

Maintainer Mauricio Vargas Sepulveda <m.vargas.sepulveda@gmail.com>

Repository CRAN

Date/Publication 2026-08-28 09:50:02 UTC

Contents

wbstats	2
wb_cache	2

wb_cachelist	3
wb_country_coverage	3
wb_data	4
wb_end_point_info	8
wb_indicators	9
wb_search	10
Index	12

wbstats	<i>Programmatic Access to Data and Statistics from the World Bank API</i>
---------	---

Description

Search and download data from the World Bank Data API. Includes support for mutliple languages, access to annual, quarterly, and monthly data.

wb_cache	<i>Download an updated list of country, indicator, and source information</i>
----------	---

Description

Download an updated list of information regarding countries, indicators, sources, regions, indicator topics, lending types, income levels, and supported languages from the World Bank API

Usage

wb_cache(lang)

Arguments

lang Language in which to return the results. If lang is unspecified, english is the default. For supported languages see [wb_languages\(\)](#). Possible values of lang are in the iso2c column. A note of warning, not all data returns have support for langauges other than english. If the specific return does not support your requested language by default it will return NA.

Value

- A list containing the following items:
- countries: The result of calling [wb_countries\(\)](#)
 - indicators: The result of calling [wb_indicators\(\)](#)
 - sources: The result of calling [wb_sources\(\)](#)
 - topics: The result of calling [wb_topics\(\)](#)

- `regions`: The result of calling `wb_regions()`
- `income_levels`: The result of calling `wb_income_levels()`
- `lending_types`: The result of calling `wb_lending_types()`
- `languages`: The result of calling `wb_languages()`

wb_cachelist

Cached information from the World Bank API

Description

This data is a cached result of the `wb_cache` function. By default functions `wb_data` and `wb_search` use this data for the `cache` parameter.

wb_country_coverage

Summarize Data Coverage for World Bank Indicators

Description

Given a indicator available from the World Bank API, this functions returns a pre-computed summary stored on GitHub and not available from the API that shows the percentage of non-missing (non-NA) observations for every country, the year range per country, and the total number of observations per country. This is useful before downloading a dataset.

Usage

```
wb_country_coverage(pattern, country, start_date, end_date, cache)
```

Arguments

- | | |
|----------------------|--|
| <code>pattern</code> | A character string or regular expression (e.g., "GDP"). |
| <code>country</code> | <p>Character vector of country, region, or special value codes for the locations you want to return data for. Permissible values can be found in the countries data frame in <code>wb_cachelist</code> or by running <code>wb_countries()</code> directly. Specifically, values listed in the following fields <code>iso3c</code>, <code>iso2c</code>, <code>country</code>, <code>region</code>, <code>admin_region</code>, <code>income_level</code> and all of the <code>region_*</code>, <code>admin_region_*</code>, <code>income_level_*</code>, columns. As well as the following special values</p> <ul style="list-style-type: none"> • "countries_only" (Default) • "regions_only" • "admin_regions_only" • "income_levels_only" • "aggregates_only" • "all" |

start_date	Numeric or character. If numeric it must be in %Y form (i.e. four digit year). For data at the subannual granularity the API supports a format as follows: for monthly data, "2016M01" and for quarterly data, "2016Q1". This also accepts a special value of "YTD", useful for more frequently updated subannual indicators.
end_date	Numeric or character. If numeric it must be in %Y form (i.e. four digit year). For data at the subannual granularity the API supports a format as follows: for monthly data, "2016M01" and for quarterly data, "2016Q1".
cache	List of data frames returned from wb_cache() . If omitted, wb_cachelist is used

Value

A data.table with one row per country.

Examples

```
## Not run:
wb_country_coverage("gross domestic product", c("Mexico", "Chile"), 2010, 2020)

## End(Not run)
```

wb_data

Download Data from the World Bank API

Description

This function downloads the requested information using the World Bank API

Usage

```
wb_data(
  indicator,
  country = "countries_only",
  start_date,
  end_date,
  return_wide = TRUE,
  mrv,
  mrnev,
  cache,
  freq,
  gapfill = FALSE,
  date_as_class_date = FALSE,
  lang
)
```

Arguments

indicator	Character vector of indicator codes. These codes correspond to the indicator_id column from the indicators data frame of <code>wb_cache()</code> , <code>wb_cachelist</code> , or the result of running <code>wb_indicators()</code> directly
country	Character vector of country, region, or special value codes for the locations you want to return data for. Permissible values can be found in the countries data frame in <code>wb_cachelist</code> or by running <code>wb_countries()</code> directly. Specifically, values listed in the following fields iso3c, iso2c, country, region, admin_region, income_level and all of the region_*, admin_region_*, income_level_*, columns. As well as the following special values • "countries_only" (Default) • "regions_only" • "admin_regions_only" • "income_levels_only" • "aggregates_only" • "all"
start_date	Numeric or character. If numeric it must be in %Y form (i.e. four digit year). For data at the subannual granularity the API supports a format as follows: for monthly data, "2016M01" and for quarterly data, "2016Q1". This also accepts a special value of "YTD", useful for more frequently updated subannual indicators.
end_date	Numeric or character. If numeric it must be in %Y form (i.e. four digit year). For data at the subannual granularity the API supports a format as follows: for monthly data, "2016M01" and for quarterly data, "2016Q1".
return_wide	Logical. If TRUE data is returned in a wide format instead of long, with a column named for each indicator_id or if the indicator argument is a named vector, the <code>names()</code> given to the indicator will be the column names. To necessitate this transformation, the indicator column that provides the human readable description is dropped, but provided as a column label. Default is TRUE
mrsv	Numeric. The number of Most Recent Values to return. A replacement of start_date and end_date, this number represents the number of observations you wish to return starting from the most recent date of collection. This may include missing values. Useful in conjunction with freq
mrnev	Numeric. The number of Most Recent Non Empty Values to return. A replacement of start_date and end_date, similar in behavior as mrsv but excludes locations with missing values. Useful in conjunction with freq
cache	List of data frames returned from <code>wb_cache()</code> . If omitted, <code>wb_cachelist</code> is used
freq	Character String. For fetching quarterly ("Q"), monthly("M") or yearly ("Y") values. Useful for querying high frequency data.
gapfill	Logical. If TRUE fills in missing values by carrying forward the last available value until the next available period (max number of periods back tracked will be limited by mrsv number). Default is FALSE
date_as_class_date	Logical. If TRUE the date field is returned as class <code>Date</code> , useful when working with non-annual data or data at mixed resolutions. Default is FALSE available

	value until the next available period (max number of periods back tracked will be limited by mrv number). Default is FALSE
lang	Language in which to return the results. If lang is unspecified, english is the default. For supported languages see wb_languages() . Possible values of lang are in the iso2c column. A note of warning, not all data returns have support for languages other than english. If the specific return does not support your requested language by default it will return NA.

Details

obs_status column

Indicates the observation status for location, indicator and date combination. For example "F" in the response indicates that the observation status for that data point is "forecast".

Value

a data.table/data.frame of all available requested data.

Examples

```
# NOTE: These examples are wrapped in a "don't run" because these
# require an internet connection

# gdp for all countries for all available dates
## Not run:
df_gdp <- wb_data("NY.GDP.MKTP.CD")

## End(Not run)

# Brazilian gdp for all available dates
## Not run:
df_brazil <- wb_data("NY.GDP.MKTP.CD", country = "br")

## End(Not run)

# Brazilian gdp for 2006
## Not run:
df_brazil_1 <- wb_data("NY.GDP.MKTP.CD", country = "brazil", start_date = 2006)

## End(Not run)

# Brazilian gdp for 2006-2010
## Not run:
df_brazil_2 <- wb_data("NY.GDP.MKTP.CD",
  country = "BRA",
  start_date = 2006, end_date = 2010
)

## End(Not run)

# Population, GDP, Unemployment Rate, Birth Rate (per 1000 people)
```

```
## Not run:
my_indicators <- c(
  "SP.POP.TOTL",
  "NY.GDP.MKTP.CD",
  "SL.UEM.TOTL.ZS",
  "SP.DYN.CBRT.IN"
)

## End(Not run)

## Not run:
df <- wb_data(my_indicators)

## End(Not run)

# you pass multiple country ids of different types
# Albania (iso2c), Georgia (iso3c), and Mongolia
## Not run:
my_countries <- c("AL", "Geo", "mongolia")
df <- wb_data(my_indicators,
  country = my_countries,
  start_date = 2005, end_date = 2007
)

## End(Not run)

# same data as above, but in long format
## Not run:
df_long <- wb_data(my_indicators,
  country = my_countries,
  start_date = 2005, end_date = 2007,
  return_wide = FALSE
)

## End(Not run)

# regional population totals
# regions correspond to the region column in wb_cachelist$countries
## Not run:
df_region <- wb_data("SP.POP.TOTL",
  country = "regions_only",
  start_date = 2010, end_date = 2014
)

## End(Not run)

# a specific region
## Not run:
df_world <- wb_data("SP.POP.TOTL",
  country = "world",
  start_date = 2010, end_date = 2014
)
```

```
## End(Not run)

# if the indicator is part of a named vector the name will be the column name
my_indicators <- c(
  "pop" = "SP.POP.TOTL",
  "gdp" = "NY.GDP.MKTP.CD",
  "unemployment_rate" = "SL.UEM.TOTL.ZS",
  "birth_rate" = "SP.DYN.CBRT.IN"
)
## Not run:
df_names <- wb_data(my_indicators,
  country = "world",
  start_date = 2010, end_date = 2014
)

## End(Not run)

# custom names are ignored if returning in long format
## Not run:
df_names_long <- wb_data(my_indicators,
  country = "world",
  start_date = 2010, end_date = 2014,
  return_wide = FALSE
)

## End(Not run)

# same as above but in Bulgarian
# note that not all indicators have translations for all languages
## Not run:
df_names_long_bg <- wb_data(my_indicators,
  country = "world",
  start_date = 2010, end_date = 2014,
  return_wide = FALSE, lang = "bg"
)

## End(Not run)
```

wb_end_point_info

World Bank Information End Points

Description

These functions are simple wrappers around the various useful API end points that are helpful for finding available data and filtering the data you are interested in when using [wb_data\(\)](#)

Usage

```
wb_countries(lang)
wb_topics(lang)
```

```

wb_sources(lang)
wb_regions(lang)
wb_income_levels(lang)
wb_lending_types(lang)
wb_languages()

```

Arguments

lang Language in which to return the results. If `lang` is unspecified, english is the default. For supported languages see [wb_languages\(\)](#). Possible values of `lang` are in the `iso2c` column. A note of warning, not all data returns have support for languages other than english. If the specific return does not support your requested language by default it will return NA.

Value

A `data.table/data.frame` of information about the end point. A `data.table` is used instead of a plain `data.frame` so that large tables (e.g. the ~28k rows returned by [wb_indicators\(\)](#)) print truncated to a handful of rows instead of flooding the console.

See Also

[wb_cache\(\)](#)

wb_indicators	<i>Download Available Indicators from the World Bank</i>
---------------	--

Description

This function returns a `data.frame` of indicator IDs and related information that are available for download from the World Bank API

Usage

```
wb_indicators(lang, include_archive = FALSE)
```

Arguments

include_archive logical. If TRUE indicators that have been archived by the World Bank will be included in the return. Data for these additional indicators are not available through the standard API and querying them using [wb_data\(\)](#) will not return data. Default is FALSE.

lang Language in which to return the results. If `lang` is unspecified, english is the default. For supported languages see [wb_languages\(\)](#). Possible values of `lang` are in the `iso2c` column. A note of warning, not all data returns have support for languages other than english. If the specific return does not support your requested language by default it will return NA.

Examples

```
# can get a new list of available indicators by downloading new cache

fresh_cache <- wb_cache()

fresh_indicators <- fresh_cache$indicators

# or by running the wb_indicators() function directly

fresh_indicators <- wb_indicators()

# include archived indicators
# see include_archive parameter description

indicators_with_archive <- wb_indicators(include_archive = TRUE)
```

wb_search

*Search indicator information available through the World Bank API***Description**

This function allows finds indicators that match a search term and returns a data frame of matching results

Usage

```
wb_search(
  pattern,
  fields = c("indicator_id", "indicator", "indicator_desc"),
  extra = FALSE,
  cache,
  ignore.case = TRUE,
  ...
)
```

Arguments

pattern	Character string or regular expression to be matched
fields	Character vector of column names through which to search
extra	if FALSE, only the indicator ID and short name are returned, if TRUE, all columns of the cache parameter's indicators data frame are returned. Default is FALSE
ignore.case	if FALSE, the pattern matching is case sensitive and if TRUE, case is ignored during matching. Default is TRUE
cache	List of data frames returned from <code>wb_cache()</code> . If omitted, <code>wb_cachelist</code> is used
...	Any additional <code>grep()</code> arguments you which to pass

Value

a data.table/data.frame with indicators that match the search pattern.

Examples

```
d <- wb_search(pattern = "education")

d <- wb_search(pattern = "Food and Agriculture Organization", fields = "source_org")

# with regular expression operators
# 'poverty' OR 'unemployment' OR 'employment'

d <- wb_search(pattern = "poverty|unemployment|employment")

# pass any other grep argument along as well
# everything without 'education'

d <- wb_search(pattern = "education", invert = TRUE)

# contains "gdp" AND "trade"

d <- wb_search("^(?=.*gdp)(?=.*trade).*", perl = TRUE)

# contains "gdp" and NOT "trade"

d <- wb_search("^(?=.*gdp)(?!.*trade).*", perl = TRUE)
```

Index

Date, [5](#)

grep(), [10](#)

names(), [5](#)

wb_cache, [2](#), [3](#)

wb_cache(), [4](#), [5](#), [9](#), [10](#)

wb_cachelist, [3](#), [3](#), [4](#), [5](#), [10](#)

wb_countries(wb_end_point_info), [8](#)

wb_countries(), [2](#), [3](#), [5](#)

wb_country_coverage, [3](#)

wb_data, [3](#), [4](#)

wb_data(), [8](#), [9](#)

wb_end_point_info, [8](#)

wb_income_levels(wb_end_point_info), [8](#)

wb_income_levels(), [3](#)

wb_indicators, [9](#)

wb_indicators(), [2](#), [5](#), [9](#)

wb_languages(wb_end_point_info), [8](#)

wb_languages(), [2](#), [3](#), [6](#), [9](#)

wb_lending_types(wb_end_point_info), [8](#)

wb_lending_types(), [3](#)

wb_regions(wb_end_point_info), [8](#)

wb_regions(), [3](#)

wb_search, [3](#), [10](#)

wb_sources(wb_end_point_info), [8](#)

wb_sources(), [2](#)

wb_topics(wb_end_point_info), [8](#)

wb_topics(), [2](#)

wbstats, [2](#)