

Package ‘regcensus’

October 4, 2023

Type Package

Title Accessing Data from the 'RegCensusAPI'

Version 1.0.1

Description Allowing users to access data from the 'RegCensusAPI'. The 'RegCensusAPI' is an API client that connects to the 'RegData' regulatory restrictions data by the 'Mercatus Center' at 'George Mason University'. 'RegData' uses machine learning algorithms to quantify the number of regulatory restrictions in a jurisdiction. You can find out more about 'RegData' from 'QuantGov website' <<https://www.quantgov.org>>.

License MIT + file LICENSE

URL <https://github.com/QuantGov/regcensus-api-R>

BugReports <https://github.com/QuantGov/regcensus-api-R/issues>

Depends R (>= 4.3.0)

Imports jsonlite, stringr, httr, tidyverse, dplyr

Suggests knitr, rmarkdown, testthat (>= 3.1.9)

VignetteBuilder knitr

Encoding UTF-8

RoxygenNote 7.2.3

Config/testthat/edition 3

Language en-US

NeedsCompilation no

Author Jonathan Nelson [aut, cre],
Thurston Powers [aut],
Aayush Gadia [aut],
Mercatus Center [cph, fnd]

Maintainer Jonathan Nelson <jnelson@mercatus.gmu.edu>

Repository CRAN

Date/Publication 2023-10-03 22:00:03 UTC

R topics documented:

agency_url	2
clean_columns	3
get_agencies	3
get_datafinder	4
get_documentation	4
get_documents	5
get_document_values	6
get_endpoint	7
get_industries	8
get_jurisdictions	9
get_reading_time	9
get_series	11
get_values	11
industries_url	13
jurisdictions_url	14
list_agencies	14
list_clusters	15
list_dates	15
list_document_types	16
list_industries	16
list_jurisdictions	17
list_series	18
print_error	18
reading_time	19
series_url	19
Index	21

agency_url	<i>agency_url</i>
------------	-------------------

Description

Gets url call for agencies endpoint.

Usage

```
agency_url(jurisdiction_id, keyword)
```

Arguments

jurisdiction_id	ID for the jurisdiction
keyword	search for keyword in agency name

Value

url as a string

Examples

```
## Not run: agency_url(jurisdiction_id = 38, keyword = "test_word")
```

clean_columns	<i>clean_columns</i>
---------------	----------------------

Description

Removes prefixes from column names

Usage

```
clean_columns(df)
```

Arguments

df Uncleaned dataframe as input

Value

Cleaned Dataframe

get_agencies	<i>get_agencies</i>
--------------	---------------------

Description

Get metadata for all agencies of a specific jurisdiction

Usage

```
get_agencies(jurisdiction_id = NULL, keyword = NULL, verbose = 0)
```

Arguments

jurisdiction_id ID for the jurisdiction

keyword search for keyword in agency name

verbose prints out the url of the API call, default value of 0

Value

Returns pandas dataframe with the metadata

Examples

```
## Not run: get_agencies(jurisdiction_id = 38, keyword = "test")
```

get_datafinder	<i>get_datafinder</i>
----------------	-----------------------

Description

Get API info for a specific jurisdiction and documentType

Usage

```
get_datafinder(jurisdiction, document_type = NULL)
```

Arguments

jurisdiction ID for the jurisdiction
document_type ID for type of document, default value is NULL

Value

Returns dataframe with the series and years available, along with the endpoints to access the data

Examples

```
## Not run: get_datafinder(jurisdiction = 1)
```

get_documentation	<i>get_documentation</i>
-------------------	--------------------------

Description

Get documentation for projects, including citations.

Usage

```
get_documentation()
```

Value

Returns a dataframe with information on the source of the documentation, the source name, source citation, source url and documentation.

Examples

```
## Not run: get_documentation()
```

get_documents	<i>get_documents</i>
---------------	----------------------

Description

Get metadata for documents available in a specific jurisdiction or for a specific document ID

Usage

```
get_documents(  
  document_id = NULL,  
  jurisdiction_id = NULL,  
  date = NULL,  
  document_type = 1,  
  verbose = 0  
)
```

Arguments

document_id	ID of the specific document
jurisdiction_id	ID for the jurisdiction
date	Year(s) of the documents
document_type	ID for type of document, default value of 1
verbose	prints out the url of the API call, default value of 0

Value

Returns pandas dataframe with the metadata

Examples

```
## Not run: get_documents(jurisdiction_id = 38, date = 2022)
```

`get_document_values` *get_document_values*

Description

Get values for a specific jurisdiction and series at the document level

Usage

```
get_document_values(
  series,
  jurisdiction,
  year,
  document_type = 1,
  summary = FALSE,
  date_is_range = TRUE,
  country = FALSE,
  agency = NULL,
  cluster = NULL,
  label = NULL,
  industry = NULL,
  filtered = TRUE,
  label_level = 3,
  industry_level = NULL,
  label_source = "NAICS",
  version = NULL,
  download = FALSE,
  page = NULL,
  date = NULL,
  verbose = 0
)
```

Arguments

<code>series</code>	Series ID (s)
<code>jurisdiction</code>	ID for the jurisdiction
<code>year</code>	Year(s) of data
<code>document_type</code>	ID for type of document, default value of 1
<code>summary</code>	Return summary instead of document level data (only one year of data is allowed for document level data), default value is FALSE here
<code>date_is_range</code>	Indicating whether the time parameter is range or should be treated as single data points, default value is TRUE
<code>country</code>	Get values for all subjurisdictions, default value is FALSE
<code>agency</code>	Agency ID, default value is NULL
<code>cluster</code>	Cluster ID

label	Industry code using the jurisdiction-specific coding system (returns all 3-digit industries by default), default value is NULL
industry	industry is deprecated; use label
filtered	Exclude poorly-performing industry results (use of unfiltered results is NOT recommended), default value is TRUE
label_level	Level of NAICS industries to include, default value is 3
industry_level	industryLevel is deprecated; use labellevel
label_source	classification standard (NAICS, BEA, SOC), default value of "NAICS"
version	Version ID for datasets with multiple versions (if no ID is given, returns most recent version), default value is NULL
download	If not False, a path location for a downloaded csv of the results, default value is FALSE
page	Page Number of the Response, default value is NULL
date	date is deprecated, use year now
verbose	Print out the url of the API call (useful for debugging), default value is 0

Value

Returns pandas dataframe with the metadata

Examples

```
## Not run: get_document_values(
  series = 33, jurisdiction = 38,
  year = 2018, label = "111"
)
## End(Not run)
```

get_endpoint	<i>get_endpoint</i>
--------------	---------------------

Description

Get endpoint for a specific series, jurisdiction, year, document_type combo

Usage

```
get_endpoint(series, jurisdiction, year, document_type, summary = TRUE)
```

Arguments

series	Series ID(s)
jurisdiction	ID for the jurisdiction
year	Year(s) of data
document_type	ID for type of document
summary	Return summary instead of document level data (only one year of data is allowed for document level data), default value is TRUE

Value

Returns the endpoint, e.g. '/state-summary' for summary-level state data

Examples

```
## Not run: get_endpoint(  
  series = c(28, 33, 36), jurisdiction = 38,  
  year = c(1970, 2003, 2004, 2018), document_type = 1  
)  
## End(Not run)
```

get_industries	<i>get_industries</i>
----------------	-----------------------

Description

Get metadata for all industries available in a specific jurisdiction

Usage

```
get_industries(  
  keyword = NULL,  
  label_level = 3,  
  label_source = NULL,  
  verbose = 0  
)
```

Arguments

keyword	search for keyword in industry name
label_level	NAICS level (2 to 6-digit), default value of 3
label_source	classification standard (NAICS, BEA, SOC)
verbose	prints out the url of the API call, default value of 0

Value

Returns pandas dataframe with the metadata

Examples

```
## Not run: get_industries()
```

`get_jurisdictions` *get_jurisdictions*

Description

Get metadata for all or one specific jurisdiction

Usage

```
get_jurisdictions(verbose = 0)
```

Arguments

`verbose` prints out the url of the API call, default value of 0

Value

Returns pandas dataframe with the metadata

Examples

```
## Not run: get_jurisdictions()
```

`get_reading_time` *get_reading_time*

Description

Convert word counts to total reading time

Usage

```
get_reading_time(  
  series = 2,  
  jurisdiction,  
  year,  
  document_type = 1,  
  summary = TRUE,  
  date_is_range = TRUE,  
  country = FALSE,  
  agency = NULL,  
  cluster = NULL,  
  label = NULL,  
  industry = NULL,  
  filtered = TRUE,  
  label_level = 3,
```

```

industry_level = NULL,
label_source = "NAICS",
version = NULL,
download = FALSE,
page = NULL,
date = NULL,
verbose = 0
)

```

Arguments

series	Series ID (s), default value is 2 here
jurisdiction	ID for the jurisdiction
year	Year(s) of data
document_type	ID for type of document, default value of 1
summary	Return summary instead of document level data (only one year of data is allowed for document level data), default value is TRUE
date_is_range	Indicating whether the time parameter is range or should be treated as single data points, default value is TRUE
country	Get values for all subjurisdictions, default value is FALSE
agency	Agency ID, default value is NULL
cluster	Cluster ID
label	Industry code using the jurisdiction-specific coding system (returns all 3-digit industries by default), default value is NULL
industry	industry is deprecated; use label
filtered	Exclude poorly-performing industry results (use of unfiltered results is NOT recommended), default value is TRUE
label_level	Level of NAICS industries to include, default value is 3
industry_level	industry_level is deprecated; use label_level
label_source	classification standard (NAICS, BEA, SOC), default value of "NAICS"
version	Version ID for datasets with multiple versions (if no ID is given, returns most recent version), default value is NULL
download	If not False, a path location for a downloaded csv of the results, default value is FALSE
page	Page Number of the Response, default value is NULL
date	date is deprecated, use year now
verbose	Print out the url of the API call (useful for debugging), default value is 0

Value

Returns pandas dataframe with the metadata

Examples

```
## Not run: get_reading_time(  
  jurisdiction = 45,  
  year = array(c(2022, 2023))  
)  
## End(Not run)
```

get_series

get_series

Description

Get series metadata for all or one specific jurisdiction

Usage

```
get_series(verbose = 0)
```

Arguments

verbose prints out the url of the API call, default value of 0

Value

Returns pandas dataframe with the metadata

Examples

```
## Not run: get_series()
```

get_values

get_values

Description

Get values for a specific jurisdiction, series, and year

Usage

```

get_values(
  series,
  jurisdiction,
  year,
  document_type = 1,
  summary = TRUE,
  date_is_range = TRUE,
  country = FALSE,
  agency = NULL,
  cluster = NULL,
  label = NULL,
  industry = NULL,
  filtered = TRUE,
  label_level = 3,
  industry_level = NULL,
  label_source = "NAICS",
  version = NULL,
  download = FALSE,
  page = NULL,
  date = NULL,
  verbose = 0
)

```

Arguments

series	Series ID (s)
jurisdiction	ID for the jurisdiction
year	Year(s) of data
document_type	ID for type of document, default value of 1
summary	Return summary instead of document level data (only one year of data is allowed for document level data), default value is TRUE
date_is_range	Indicating whether the time parameter is range or should be treated as single data points, default value is TRUE
country	Get values for all subjurisdictions, default value is FALSE
agency	Agency ID, default value is NULL
cluster	Cluster ID
label	Industry code using the jurisdiction-specific coding system (returns all 3-digit industries by default), default value is NULL
industry	industry is deprecated; use label
filtered	Exclude poorly-performing industry results (use of unfiltered results is NOT recommended), default value is TRUE
label_level	Level of NAICS industries to include, default value is 3
industry_level	industry_level is deprecated; use label_level

label_source	classification standard (NAICS, BEA, SOC), default value of "NAICS"
version	Version ID for datasets with multiple versions (if no ID is given, returns most recent version), default value is NULL
download	If not False, a path location for a downloaded csv of the results, default value is FALSE
page	Page Number of the Response, default value is NULL
date	date is deprecated, use year now
verbose	Print out the url of the API call (useful for debugging), default value is 0

Value

Returns pandas dataframe with the values and various metadata, and returns empty if required parameters are not given

Examples

```
## Not run: get_values(
  series = array(c(1, 28, 33, 36)), jurisdiction = 38,
  year = array(c(1970, 2003, 2004, 2018, 2020)), country = TRUE
)
## End(Not run)
```

industries_url	<i>industries_url</i>
----------------	-----------------------

Description

Gets url call for label (formerly industries) endpoint.

Usage

```
industries_url(keyword, label_level, label_source)
```

Arguments

keyword	search for keyword in industry name
label_level	NAICS level (2 to 6-digit)
label_source	classification standard (NAICS, BEA, SOC)

Value

url as a string

Examples

```
## Not run: industries_url("test", 323, "NAICS")
```

jurisdictions_url	<i>jurisdictions_url</i>
-------------------	--------------------------

Description

Gets url call for jurisdictions endpoint.

Usage

```
jurisdictions_url()
```

Value

url as a string

Examples

```
## Not run: jurisdictions_url()
```

list_agencies	<i>list_agencies</i>
---------------	----------------------

Description

Fetches the names of the agencies and their associated IDs

Usage

```
list_agencies(jurisdiction_id = NULL, keyword = NULL, reverse = FALSE)
```

Arguments

jurisdiction_id	ID for the jurisdiction
keyword	search for keyword in agency name
reverse	reverses the key-value mapping, default value of FALSE

Value

Returns dictionary containing names of agencies and associated IDs

Examples

```
## Not run: list_agencies(jurisdiction_id = 38)
```

list_clusters	<i>list_clusters</i>
---------------	----------------------

Description

Fetches the names of the clusters and their associated IDs

Usage

```
list_clusters(reverse = FALSE)
```

Arguments

reverse reverses the key-value mapping, default value of FALSE

Value

Returns dictionary containing names of clusters & associated IDs

Examples

```
## Not run: list_clusters()
```

list_dates	<i>list_dates</i>
------------	-------------------

Description

Fetches the dates available for the jurisdiction

Usage

```
list_dates(jurisdiction_id, document_type = NULL)
```

Arguments

jurisdiction_id ID for the jurisdiction
document_type ID for type of document

Value

Returns list of dates available for the jurisdiction

Examples

```
## Not run: list_dates(jurisdiction_id = 38)
```

list_document_types *list_document_types*

Description

Fetches the names of documenttypes and associated IDs

Usage

```
list_document_types(jurisdiction_id = NULL, reverse = FALSE, verbose = 0)
```

Arguments

jurisdiction_id	ID for the jurisdiction
reverse	reverses the key-value mapping
verbose	prints out the url of the API call

Value

Returns dictionary containing names of documenttypes & associated IDs

Examples

```
## Not run: list_document_types(jurisdiction_id = 65,  
                               reverse = FALSE, verbose = 0)  
## End(Not run)
```

list_industries *list_industries*

Description

Fetches the names of industries and associated IDs

Usage

```
list_industries(  
  keyword = NULL,  
  label_level = 3,  
  label_source = "NAICS",  
  only_id = FALSE,  
  reverse = FALSE  
)
```


Arguments

keyword	search for keyword in agency name
label_level	NAICS level (2 to 6-digit), default value of 3
label_source	classification standard (NAICS, BEA, SOC), default value of "NAICS"
only_id	uses the NAICS code instead of name as key of dictionary, default value of FALSE
reverse	reverses the key-value mapping, default value of FALSE

Value

Returns dictionary containing names of industries and associated IDs

Examples

```
## Not run: list_industries(only_id = TRUE, reverse = FALSE)
```

list_jurisdictions *list_jurisdictions*

Description

Fetches names of jurisdictions and associated IDs

Usage

```
list_jurisdictions(reverse = FALSE)
```

Arguments

reverse	reverses the key-value mapping, default value of FALSE
---------	--

Value

Returns dictionary containing names of jurisdictions & associated IDs

Examples

```
## Not run: list_jurisdictions(reverse = TRUE)
```

list_series	<i>list_series</i>
-------------	--------------------

Description

Fetches the names of series and associated IDs

Usage

```
list_series(reverse = FALSE)
```

Arguments

reverse reverses the key-value mapping, default value of FALSE

Value

Returns dictionary containing names of series and associated IDs

Examples

```
## Not run: list_series(reverse = FALSE)
```

print_error	<i>print_error</i>
-------------	--------------------

Description

Handle and print out error for invalid API call.

Usage

```
print_error(output)
```

Arguments

output key-value mapping of the output

Value

None

Examples

```
## Not run: print_error(list(message = "test"))
```

reading_time	<i>reading_time</i>
--------------	---------------------

Description

Computes the time it takes to read a document based on the words a document has.

Computes the time it takes to read a document based on the words a document has.

Usage

```
reading_time(words, workday = 8, workweek = 5, workyear = 50)
```

Arguments

words	count of words
workday	default value of 8 hour workday
workweek	default value of 5 day work week
workyear	default value of 50 week work year

Value

A string detailing how long it takes to read a document based on how many words the document has. The function assumes an 8 hour work-day, a 5 day work-week, and a 50 week work-year.

A string detailing how long it takes to read a document based on how many words the document has. The function assumes an 8 hour work-day, a 5 day work-week, and a 50 week work-year.

Examples

```
## Not run: reading_time(1200000, workday = 8,
                        workweek = 5, workyear = 50)
## End(Not run)
reading_time(1200000, workday=8, workweek=5, workyear=50)
```

series_url	<i>series_url</i>
------------	-------------------

Description

Gets url call for dataserie endpoint

Usage

```
series_url()
```

Value

url as a string

Examples

```
## Not run: series_url()
```

Index

[agency_url](#), 2

[clean_columns](#), 3

[get_agencies](#), 3

[get_datafinder](#), 4

[get_document_values](#), 6

[get_documentation](#), 4

[get_documents](#), 5

[get_endpoint](#), 7

[get_industries](#), 8

[get_jurisdictions](#), 9

[get_reading_time](#), 9

[get_series](#), 11

[get_values](#), 11

[industries_url](#), 13

[jurisdictions_url](#), 14

[list_agencies](#), 14

[list_clusters](#), 15

[list_dates](#), 15

[list_document_types](#), 16

[list_industries](#), 16

[list_jurisdictions](#), 17

[list_series](#), 18

[print_error](#), 18

[reading_time](#), 19

[series_url](#), 19