Package ‘geobr’
January 9, 2024

Type     Package
Title    Download Official Spatial Data Sets of Brazil
Version  1.8.2
BugReports https://github.com/ipeaGIT/geobr/issues
Description Easy access to official spatial data sets of Brazil as ‘sf’ objects in R. The package includes a wide range of geospatial data available at various geographic scales and for various years with harmonized attributes, projection and fixed topology.
License  MIT + file LICENSE
Encoding UTF-8
LazyData TRUE
Depends R (>= 3.5.0)
Imports curl, data.table, httr (>= 1.4.1), methods, sf (>= 0.9-3),
       utils
Suggests censobr, covr, dplyr (>= 0.8-3), ggplot2 (>= 3.3.1), knitr,
       markdown, scales, testthat
RoxygenNote 7.2.3
VignetteBuilder knitr
NeedsCompilation no
Author Rafael H. M. Pereira [aut, cre]
       (https://orcid.org/0000-0003-2125-7465),
       Caio Nogueira Goncalves [aut],
       Paulo Henrique Fernandes de Araujo [ctb],
       Guilherme Duarte Carvalho [ctb],
       Rodrigo Almeida de Arruda [ctb],
       Igor Nascimento [ctb],
       Barbara Santiago Pedreira da Costa [ctb],
       Welligton Silva Cavedo [ctb],
       Pedro R. Andrade [ctb],
       Alan da Silva [ctb],
Carlos Kauê Vieira Braga [ctb],
Carl Schmertmann [ctb],
Alessandro Samuel-Rosa [ctb],
Daniel Ferreira [ctb],
Marcus Saraiva [ctb],
Beatriz Milz [ctb] (<https://orcid.org/0000-0002-3064-4486>),
Ipea - Institute for Applied Economic Research [cph, fnd]

Maintainer Rafael H. M. Pereira <rafa.pereira.br@gmail.com>
Repository CRAN
Date/Publication 2024-01-09 10:40:10 UTC

R topics documented:

cep_to_state ....................................................... 3
grid_state_correspondence_table .................................. 3
list_geobr .......................................................... 4
lookup_muni ......................................................... 5
read_amazon ........................................................ 6
read_biomes .......................................................... 7
read_capitals ......................................................... 8
read_census_tract ................................................... 9
read_comparable_areas .............................................. 10
read_conservation_units .......................................... 11
read_country ......................................................... 12
read_disaster_risk_area ............................................ 13
read_health_facilities ............................................. 14
read_health_region ................................................ 15
read_immediate_region ............................................. 16
read_indigenous_land .............................................. 17
read_intermediate_region ........................................... 18
read_meso_region .................................................. 20
readMetro_area ...................................................... 21
read_micro_region .................................................. 22
read_municipality .................................................. 23
read_municipal_seat ................................................. 24
read_neighborhood .................................................. 25
read_pop_arrangements ............................................. 26
read_region .......................................................... 27
read_schools ........................................................ 28
read_semiarid ....................................................... 29
read_state .......................................................... 30
read_statistical_grid .............................................. 31
read_urban_area ..................................................... 32
read_urban_concentrations ....................................... 33
read_weighting_area ................................................. 34

Index 36
cep_to_state

Determine the state of a given CEP postal code

Description

Zips codes in Brazil are known as CEP, the abbreviation for postal code address. CEPs in Brazil are 8 digits long, with the format 'xxxxxxx-xxx'.

Usage

cp_to_state(cep)

Arguments

cep A character string with 8 digits in the format "xxxxxxx", or with the format 'xxxxxxx-xxx'.

Value

A character string with a state abbreviation.

Examples

uf <- cep_to_state(cep = '69900-000')

# Or:
uf <- cep_to_state(cep = '69900000')

gp_grid_state_correspondence_table

A correspondence table indicating what quadrants of IBGE's statistical grid intersect with each Brazilian state

Description

Built-in dataset

- name_state: Title-case name of state (character)
- abbrev_state: Two-letter uppercase abbreviation of a state
- code_grid: Unique code of each quadrant of IBGE's statistical grid

Usage

data(grid_state_correspondence_table)
list_geobr

Format
A data frame sf with 139 rows and 3 columns

Details
correspondence table indicating what quadrants of IBGE’s statistical grid intersect with each Brazilian state

Note
Last updated 2021-03-21

list_geobr

List all data sets available in the geobr package

Description
Returns a data frame with all datasets available in the geobr package

Usage
list_geobr()

Value
A data.frame

See Also
Other support functions: lookup_muni()

Examples

df <- list_geobr()
Description

Input a municipality name or code and get the names and codes of the municipality’s corresponding state, meso, micro, intermediate, and immediate regions

Usage

```
lookup_muni(name_muni = NULL, code_muni = NULL)
```

Arguments

- `name_muni` The municipality name to be looked up.
- `code_muni` The municipality code to be looked up.

Details

Only available from 2010 Census data so far

Value

A data.frame with 13 columns identifying the geographies information of that municipality.

A data.frame

See Also

Other support functions: `list_geobr()`

Examples

```
# Get lookup table for municipality Rio de Janeiro
mun <- lookup_muni(name_muni = "Rio de Janeiro")

# Or you can get a lookup table for the same municipality searching for its code
mun <- lookup_muni(code_muni = 3304557)

# Get lookup table for all municipalities
mun_all <- lookup_muni(name_muni = "all")

# Or:
mun_all <- lookup_muni(code_muni = "all")
```
Description
This data set covers the whole of Brazil’s Legal Amazon as defined in the federal law n. 12.651/2012). The original data comes from the Brazilian Ministry of Environment (MMA) and can be found at "http://mapas.mma.gov.br/ig3geo/datadownload.htm".

Usage
read_amazon(year = 2012, simplified = TRUE, showProgress = TRUE)

Arguments
year Numeric. Year of the data in YYYY format. Defaults to 2012.
simplified Logic FALSE or TRUE, indicating whether the function should return the data set with 'original' spatial resolution or a data set with 'simplified' geometry. Defaults to TRUE. For spatial analysis and statistics users should set simplified = FALSE. Borders have been simplified by removing vertices of borders using st_simplify{sf} preserving topology with a dTolerance of 100.
showProgress Logical. Defaults to TRUE display progress bar.

Value
An "sf" "data.frame" object

See Also
Other area functions: read_biomes(), read_capitals(), read_comparable_areas(), read_country(), read_disaster_risk_area(), read_health_facilities(), read_health_region(), read_immediate_region(), read_indigenous_land(), read_intermediate_region(), read_meso_region(), read_metro_area(), read_micro_region(), read_municipal_seat(), read_municipality(), read_neighborhood(), read_pop_arrangements(), read_region(), read_schools(), read_semiarid(), read_state(), read_statistical_grid(), read_urban_area(), read_urban_concentrations(), read_weighting_area()

Examples
# Read Brazilian Legal Amazon
a <- read_amazon(year = 2012)
Description

This data set includes polygons of all biomes present in Brazilian territory and coastal area. The latest data set dates to 2019 and it is available at scale 1:250,000. The 2004 data set is at the scale 1:5,000,000. The original data comes from IBGE. More information at https://www.ibge.gov.br/apps/biomas/

Usage

read_biomes(year = 2019, simplified = TRUE, showProgress = TRUE)

Arguments

- **year**: Numeric. Year of the data in YYYY format. Defaults to 2019.
- **simplified**: Logic FALSE or TRUE, indicating whether the function should return the data set with 'original' spatial resolution or a data set with 'simplified' geometry. Defaults to TRUE. For spatial analysis and statistics users should set simplified = FALSE. Borders have been simplified by removing vertices of borders using st_simplify{sf} preserving topology with a dTolerance of 100.
- **showProgress**: Logical. Defaults to TRUE display progress bar.

Value

An "sf" "data.frame" object

See Also

Other area functions: read_amazon(), read_capitals(), read_comparable_areas(), read_country(), read_disaster_risk_area(), read_health_facilities(), read_health_region(), read_immediate_region(), read_indigenous_land(), read_intermediate_region(), read_meso_region(), read_metro_area(), read_micro_region(), read_municipal_seat(), read_municipality(), read_neighborhood(), read_pop_arrangements(), read_region(), read_schools(), read_semiarid(), read_state(), read_statistical_grid(), read_urban_area(), read_urban_concentrations(), read_weighting_area()

Examples

```r
# Read biomes
b <- read_biomes(year = 2019)
```
read_capitals

Description

This function downloads either a spatial sf object with the location of the municipal seats (sede dos municipios) of state capitals, or a data.frame with the names and codes of state capitals. Data downloaded for the latest available year.

Usage

read_capitals(as_sf = TRUE, showProgress = TRUE)

Arguments

as_sf Logic FALSE or TRUE, indicating whether the function should return a spatial data in sf format (Defaults to TRUE) or in a data.frame format without spatial information (FALSE).
showProgress Logical. Defaults to TRUE display progress bar.

Value

An "sf" "data.frame" object or a "data.frame"

See Also

Other area functions: read_amazon(), read_biomes(), read_comparable_areas(), read_country(), read_disaster_risk_area(), read_health_facilities(), read_health_region(), read Immediate_region(), read_indigenous_land(), read_intermediate_region(), read_meso_region(), read_metro_area(), read_micro_region(), read_municipal_seat(), read_municipality(), read_neighborhood(), read_pop_arrangements(), read_region(), read_schools(), read_semiarid(), read_state(), read_statistical_grid(), read_urban_area(), read_urban_concentrations(), read_weighting_area()

Examples

# Read spatial data with the municipal seats of state capitals
capitals_sf <- read_capitals(as_sf = TRUE)

# Read simple data.frame of state capitals
capitals_df <- read_capitals(as_sf = FALSE)
**Description**

Download spatial data of census tracts of the Brazilian Population Census

**Usage**

```r
read_census_tract(
  code_tract,
  year = 2010,
  zone = "urban",
  simplified = TRUE,
  showProgress = TRUE
)
```

**Arguments**

- `code_tract` The 7-digit code of a Municipality. If the two-digit code or a two-letter upper-case abbreviation of a state is passed, (e.g. 33 or "RJ") the function will load all census tracts of that state. If `code_tract="all"`, the function downloads all census tracts of the country.

- `year` Numeric. Year of the data in YYYY format. Defaults to 2010.

- `zone` For census tracts before 2010, 'urban' and 'rural' census tracts are separate data sets.

- `simplified` Logic FALSE or TRUE, indicating whether the function should return the data set with 'original' spatial resolution or a data set with 'simplified' geometry. Defaults to TRUE. For spatial analysis and statistics users should set `simplified = FALSE`. Borders have been simplified by removing vertices of borders using `st_simplify(sf)` preserving topology with a `dTolerance` of 100.

- `showProgress` Logical. Defaults to TRUE display progress bar.

**Value**

An "sf" "data.frame" object

**See Also**

Other general area functions: `read_conservation_units()`
Examples

# Read rural census tracts for years before 2007
c <- read_census_tract(code_tract=5201108, year=2000, zone="rural")

# Read all census tracts of a state at a given year
    c <- read_census_tract(code_tract=53, year=2010)  # or
    c <- read_census_tract(code_tract="DF", year=2010)
    plot(c)

# Read all census tracts of a municipality at a given year
    c <- read_census_tract(code_tract=5201108, year=2010)
    plot(c)

# Read all census tracts of the country at a given year
    c <- read_census_tract(code_tract="all", year=2010)

read_comparable_areas

Description

This function downloads the shape file of minimum comparable area of municipalities, known in Portuguese as 'Areas minimas comparaveis (AMCs)'. The data is available for any combination of census years between 1872-2010. These data sets are generated based on the Stata code originally developed by Ehrl (2017) doi:10.1590/0101416147182phe, and translated into R by the geobr team.

Usage

read_comparable_areas(
    start_year = 1970,
    end_year = 2010,
    simplified = TRUE,
    showProgress = TRUE
)

Arguments

start_year Numeric. Start year to the period in the YYYY format. Defaults TO 1970.
end_year Numeric. End year to the period in the YYYY format. Defaults to 2010.
simplified Logic FALSE or TRUE, indicating whether the function should return the data set with 'original' spatial resolution or a data set with 'simplified' geometry. Defaults to TRUE. For spatial analysis and statistics users should set simplified = FALSE. Borders have been simplified by removing vertices of borders using st_simplify(sf) preserving topology with a dTolerance of 100.
showProgress Logical. Defaults to TRUE display progress bar.
Details

These data sets are generated based on the original Stata code developed by Philipp Ehrl. If you use these data, please cite:


Value

An "sf" "data.frame" object

See Also

Other area functions: `read_amazon()`, `read_biomes()`, `read_capitals()`, `read_country()`, `read_disaster_risk_area()`, `read_health_facilities()`, `read_health_region()`, `read_immediate_region()`, `read_indigenous_land()`, `read_intermediate_region()`, `read_meso_region()`, `read_metro_area()`, `read_micro_region()`, `read_municipal_seat()`, `read_municipality()`, `read_neighborhood()`, `read_pop_arrangements()`, `read_region()`, `read_schools()`, `read_semiarid()`, `read_state()`, `read_statistical_grid()`, `read_urban_area()`, `read_urban_concentrations()`, `read_weighting_area()`

Examples

```r
amc <- read_comparable_areas(start_year=1970, end_year=2010)
```

Description

This data set covers the whole of Brazil and it includes the polygons of all conservation units present in Brazilian territory. The last update of the data was 09-2019. The original data comes from MMA and can be found at "http://mapas.mma.gov.br/i3geo/datadownload.htm".

Usage

```r
read_conservation_units(date = 201909, simplified = TRUE, showProgress = TRUE)
```

Arguments

- **date**: Numeric. Date of the data in YYYYMM format. Defaults to 201909.
- **simplified**: Logic FALSE or TRUE, indicating whether the function should return the data set with 'original' spatial resolution or a data set with 'simplified' geometry. Defaults to TRUE. For spatial analysis and statistics users should set simplified = FALSE. Borders have been simplified by removing vertices of borders using `st_simplify{sf}` preserving topology with a dTolerance of 100.
showProgress  Logical. Defaults to TRUE display progress bar.

Value
An "sf" "data.frame" object

See Also
Other general area functions: read_census_tract()

Examples

```r
# Read conservation_units
b <- read_conservation_units(date = 201909)
```

---

**Description**

Data at scale 1:250,000, using Geodetic reference system "SIRGAS2000" and CRS(4674).

**Usage**

```r
read_country(year = 2010, simplified = TRUE, showProgress = TRUE)
```

**Arguments**

- `year` Numeric. Year of the data in YYY format. Defaults to 2010.
- `simplified` Logic FALSE or TRUE, indicating whether the function should return the data set with 'original' spatial resolution or a data set with 'simplified' geometry. Defaults to TRUE. For spatial analysis and statistics users should set simplified = FALSE. Borders have been simplified by removing vertices of borders using `st_simplify(sf)` preserving topology with a dTolerance of 100.
- `showProgress` Logical. Defaults to TRUE display progress bar.

**Value**

An "sf" "data.frame" object

**See Also**

Other area functions: read_amazon(), read_biomes(), read_capitals(), read_comparable_areas(), read_disaster_risk_area(), read_health_facilities(), read_health_region(), read_immediate_region(), read_indigenous_land(), read_intermediate_region(), read_meso_region(), read_metro_area(), read_micro_region(), read_municipal_seat(), read_municipality(), read_neighborhood(), read_pop_arrangements(), read_region(), read_schools(), read_semiarid(), read_state(), read_statistical_grid(), read_urban_area(), read_urban_concentrations(), read_weighting_area()
Examples

```r
# Read specific year
br <- read_country(year = 2018)
```

Description

This function reads the official data of disaster risk areas in Brazil (currently only available for 2010). It specifically focuses on geodynamic and hydro-meteorological disasters capable of triggering landslides and floods. The data set covers the whole country. Each risk area polygon (known as 'BATER') has unique code id (column 'geo_bater'). The data set brings information on the extent to which the risk area polygons overlap with census tracts and block faces (column "acuracia") and number of risk areas within each risk area (column 'num'). Original data were generated by IBGE and CEMADEN. For more information about the methodology, see details at https://www.ibge.gov.br/geociencias/organizacao-do-territorio/tipologias-do-territorio/21538-populacao-em-areas-de-risco-no-brasil.html

Usage

```r
read_disaster_risk_area(year = 2010, simplified = TRUE, showProgress = TRUE)
```

Arguments

- `simplified`: Logic FALSE or TRUE, indicating whether the function should return the data set with 'original' spatial resolution or a data set with 'simplified' geometry. Defaults to TRUE. For spatial analysis and statistics users should set `simplified = FALSE`. Borders have been simplified by removing vertices of borders using `st_simplify(sf)` preserving topology with a `dTolerance` of 100.
- `showProgress`: Logical. Defaults to TRUE display progress bar.

Value

An "sf" "data.frame" object

See Also

- read_amazon()
- read_biomes()
- read_capitals()
- read_comparable_areas()
- read_country()
- read_health_facilities()
- read_health_region()
- read_indigenous_land()
- read_intermediate_region()
- read_meso_region()
- read_metro_area()
- read_micro_region()
- read_municipal_seat()
- read_municipality()
- read_neighborhood()
- read_pop_arrangements()
- read_region()
- read_schools()
- read_semiarid()
- read_state()
- read_statistical_grid()
- read_urban_area()
- read_urban_concentrations()
- read_weighting_area()
read_health_facilities

Download geolocated data of health facilities

Examples

# Read all disaster risk areas in an specific year
d <- read_disaster_risk_area(year=2010)

Description

Data comes from the National Registry of Healthcare facilities (Cadastro Nacional de Estabelecimentos de Saude - CNES), originally collected by the Brazilian Ministry of Health. According to the Ministry of Health: "The coordinates of each facility were obtained by CNES and validated by means of space operations. These operations verify if the point is in the municipality, considering a radius of 5,000 meters. When the coordinate is not correct, further searches are done in other systems of the Ministry of Health and in web services like Google Maps. Finally, if the coordinates have been correctly obtained in this process, the coordinates of the municipal head office are used. The geocode source used is registered in the database in a specific column data_source. Periodically the coordinates are revised with the objective of improving the quality of the data." The date of the last data update is registered in the database in the columns date_update and year_update. More information in the CNES data set available at https://dados.gov.br/. These data use Geodetic reference system "SIRGAS2000" and CRS(4674).

Usage

read_health_facilities(showProgress = TRUE)

Arguments

showProgress Logical. Defaults to TRUE display progress bar.

Value

An "sf" "data.frame" object

See Also

Other area functions: read_amazon(), read_biomes(), read_capitals(), read_comparable_areas(), read_country(), read_disaster_risk_area(), read_health_region(), read_immediate_region(), read_indigenous_land(), read_intermediate_region(), read_meso_region(), read_metro_area(), read_micro_region(), read_municipal_seat(), read_municipality(), read_neighborhood(), read_pop_arrangements(), read_region(), read_schools(), read_semiarid(), read_state(), read_statistical_grid(), read_urban_area(), read_urban_concentrations(), read_weighting_area()
# Read all health facilities of the whole country
h <- read_health_facilities()

read_health_region  

## Description

Health regions are used to guide the regional and state planning of health services. Macro health regions, in particular, are used to guide the planning of high complexity health services. These services involve larger economics of scale and are concentrated in few municipalities because they are generally more technology intensive, costly and face shortages of specialized professionals. A macro region comprises one or more health regions.

## Usage

```r
read_health_region(
  year = 2013,
  macro = FALSE,
  simplified = TRUE,
  showProgress = TRUE
)
```

## Arguments

<table>
<thead>
<tr>
<th>Argument</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>year</code></td>
<td>Numeric. Year of the data in YYYY format. Defaults to 2013, the latest available.</td>
</tr>
<tr>
<td><code>macro</code></td>
<td>Logic. If FALSE (default), the function downloads health regions data. If TRUE, the function downloads macro regions data.</td>
</tr>
<tr>
<td><code>simplified</code></td>
<td>Logic FALSE or TRUE, indicating whether the function should return the data set with 'original' spatial resolution or a data set with 'simplified' geometry. Defaults to TRUE. For spatial analysis and statistics users should set <code>simplified</code> = FALSE. Borders have been simplified by removing vertices of borders using <code>st_simplify(sf)</code> preserving topology with a <code>dTolerance</code> of 100.</td>
</tr>
<tr>
<td><code>showProgress</code></td>
<td>Logical. Defaults to TRUE display progress bar.</td>
</tr>
</tbody>
</table>

## Value

An "sf" "data.frame" object
read_immediate_region

See Also

Other area functions: `read_amazon()`, `read_biomes()`, `read_capitals()`, `read_comparable_areas()`, `read_country()`, `read_disaster_risk_area()`, `read_health_facilities()`, `read_immediate_region()`, `read_indigenous_land()`, `read_intermediate_region()`, `read_meso_region()`, `read_metro_area()`, `read_micro_region()`, `read_municipal_seat()`, `read_municipality()`, `read_neighborhood()`, `read_pop_arrangements()`, `read_region()`, `read_schools()`, `read_semiarid()`, `read_state()`, `read_statistical_grid()`, `read_urban_area()`, `read_urban_concentrations()`, `read_weighting_area()`

Examples

```r
# Read all health regions for a given year
hr <- read_health_region(year=2013)

# Read all macro health regions
mhr <- read_health_region(year=2013, macro = TRUE)
```

Description

The Immediate Geographic Areas are part of the geographic division of Brazil created in 2017 by IBGE. These regions were created to replace the "Micro Regions" division. Data at scale 1:250,000, using Geodetic reference system "SIRGAS2000" and CRS(4674)

Usage

```r
read_immediate_region(
  code_immediate = "all",
  year = 2019,
  simplified = TRUE,
  showProgress = TRUE
)
```

Arguments

- `code_immediate` 6-digit code of an immediate region. If the two-digit code or a two-letter uppercase abbreviation of a state is passed, (e.g. 33 or "RJ") the function will load all immediate regions of that state. If code_immediate="all" (Default), the function downloads all immediate regions of the country.
- `simplified` Logic FALSE or TRUE, indicating whether the function should return the data set with 'original' spatial resolution or a data set with 'simplified' geometry. Defaults to TRUE. For spatial analysis and statistics users should set simplified = FALSE. Borders have been simplified by removing vertices of borders using st_simplify(sf) preserving topology with a dTolerance of 100.
read_indigenous_land

showProgress Logical. Defaults to TRUE display progress bar.

Value

An "sf" "data.frame" object

See Also

Other area functions: read_amazon(), read_biomes(), read_capitals(), read_comparable_areas(), read_country(), read_disaster_risk_area(), read_health_facilities(), read_health_region(), read_indigenous_land(), read_intermediate_region(), read_meso_region(), read_metro_area(), read_micro_region(), read_municipal_seat(), read_municipality(), read_neighborhood(), read_pop_arrangements(), read_region(), read_schools(), read_semiarid(), read_statistical_grid(), read_state(), read_statistical_grid(), read_urban_area(), read_urban_concentrations(), read_weighting_area()

Examples

# Read an specific immediate region
im <- read_immediate_region(code_immediate=110006)

# Read immediate regions of a state
im <- read_immediate_region(code_immediate=12)
im <- read_immediate_region(code_immediate="AM")

# Read all immediate regions of the country
im <- read_immediate_region()
im <- read_immediate_region(code_immediate="all")

Description

The data set covers the whole of Brazil and it includes indigenous lands from all ethnicities and in different stages of demarcation. The original data comes from the National Indian Foundation (FUNAI) and can be found at http://www.funai.gov.br/index.php/shape. Although original data is updated monthly, the geobr package will only keep the data for a few months per year.

Usage

read_indigenous_land(date = 201907, simplified = TRUE, showProgress = TRUE)
Arguments

* date: Numeric. Date of the data in YYYYMM format. Defaults to 201907.
* simplified: Logic FALSE or TRUE, indicating whether the function should return the data set with 'original' spatial resolution or a data set with 'simplified' geometry. Defaults to TRUE. For spatial analysis and statistics users should set simplified = FALSE. Borders have been simplified by removing vertices of borders using st_simplify(sf) preserving topology with a dTolerance of 100.
* showProgress: Logical. Defaults to TRUE display progress bar.

Value

An "sf" "data.frame" object

See Also

Other area functions: read_amazon(), read_biomes(), read_capitals(), read_comparable_areas(), read_country(), read_disaster_risk_area(), read_health_facilities(), read_health_region(), read_intermediate_region(), read_meso_region(), read_metro_area(), read_micro_region(), read_municipal_seat(), read_municipality(), read_neighborhood(), read_pop_arrangements(), read_region(), read_schools(), read_semiarid(), read_state(), read_statistical_grid(), read_urban_area(), read_urban_concentrations(), read_weighting_area()

Examples

```r
# Read all indigenous land in an specific date
i <- read_indigenous_land(date=201907)
```

Description

The intermediate Geographic Areas are part of the geographic division of Brazil created in 2017 by IBGE. These regions were created to replace the "Meso Regions" division. Data at scale 1:250,000, using Geodetic reference system "SIRGAS2000" and CRS(4674)

Usage

```r
read_intermediate_region(
    code_intermediate = "all",
    year = 2019,
    simplified = TRUE,
    showProgress = TRUE
)
```
**read_intermediate_region**

**Arguments**

- `code_intermediate`: 4-digit code of an intermediate region. If the two-digit code or a two-letter uppercase abbreviation of a state is passed, (e.g. 33 or "RJ") the function will load all intermediate regions of that state. If `code_intermediate` = "all" (Default), the function downloads all intermediate regions of the country.


- `simplified`: Logic FALSE or TRUE, indicating whether the function should return the data set with 'original' spatial resolution or a data set with 'simplified' geometry. Defaults to TRUE. For spatial analysis and statistics users should set `simplified` = FALSE. Borders have been simplified by removing vertices of borders using `st_simplify` preserving topology with a `dTolerance` of 100.

- `showProgress`: Logical. Defaults to TRUE display progress bar.

**Value**

An "sf" "data.frame" object

**See Also**

Other area functions: `read_amazon()`, `read_biomes()`, `read_capitals()`, `read_comparable_areas()`, `read_country()`, `read_disaster_risk_area()`, `read_health_facilities()`, `read_health_region()`, `read_immediate_region()`, `read_indigenous_land()`, `read_meso_region()`, `read_metro_area()`, `read_micro_region()`, `read_municipal_seat()`, `read_municipality()`, `read_neighborhood()`, `read_pop_arrangements()`, `read_region()`, `read_schools()`, `read_semiarid()`, `read_state()`, `read_statistical_grid()`, `read_urban_area()`, `read_urban_concentrations()`, `read_weighting_area()`

**Examples**

```r
# Read an specific intermediate region
im <- read_intermediate_region(code_intermediate=1202)

# Read intermediate regions of a state
im <- read_intermediate_region(code_intermediate=12)
im <- read_intermediate_region(code_intermediate="AM")

# Read all intermediate regions of the country
im <- read_intermediate_region()
im <- read_intermediate_region(code_intermediate="all")
```
read_meso_region  Download spatial data of meso regions

Description

Data at scale 1:250,000, using Geodetic reference system "SIRGAS2000" and CRS(4674)

Usage

```r
read_meso_region(
  code_meso = "all",
  year = 2010,
  simplified = TRUE,
  showProgress = TRUE
)
```

Arguments

- `code_meso`: The 4-digit code of a meso region. If the two-digit code or a two-letter uppercase abbreviation of a state is passed, (e.g. 33 or "RJ") the function will load all meso regions of that state. If `code_meso="all"` (Default), the function downloads all meso regions of the country.


- `simplified`: Logic FALSE or TRUE, indicating whether the function should return the data set with 'original' spatial resolution or a data set with 'simplified' geometry. Defaults to TRUE. For spatial analysis and statistics users should set `simplified = FALSE`. Borders have been simplified by removing vertices of borders using `st_simplify(sf)` preserving topology with a `dTolerance` of 100.

- `showProgress`: Logical. Defaults to TRUE display progress bar.

Value

An "sf" "data.frame" object

See Also

Other area functions: `read_amazon()`, `read_biomes()`, `read_capitals()`, `read_comparable_areas()`, `read_country()`, `read_disaster_risk_area()`, `read_health_facilities()`, `read_health_region()`, `read_immediate_region()`, `read_indigenous_land()`, `read_intermediate_region()`, `read_micro_region()`, `read_municipal_seat()`, `read_municipality()`, `read_neighborhood()`, `read_pop_arrangements()`, `read_region()`, `read_schools()`, `read_semiarid()`, `read_state()`, `read_statistical_grid()`, `read_urban_area()`, `read_urban_concentrations()`, `read_weighting_area()`
Examples

```r
# Read specific meso region at a given year
meso <- read_meso_region(code_meso=3301, year=2018)

# Read all meso regions of a state at a given year
meso <- read_meso_region(code_meso=12, year=2017)
meso <- read_meso_region(code_meso="AM", year=2000)

# Read all meso regions of the country at a given year
meso <- read_meso_region(code_meso="all", year=2010)
```

### read_metro_area

*Download spatial data of official metropolitan areas in Brazil*

**Description**

The function returns the shapes of municipalities grouped by their respective metro areas. Metropolitan areas are created by each state in Brazil. The data set includes the municipalities that belong to all metropolitan areas in the country according to state legislation in each year. Original data were generated by Institute of Geography. Data at scale 1:250,000, using Geodetic reference system "SIRGAS2000" and CRS(4674).

**Usage**

```r
read_metro_area(year = 2018, simplified = TRUE, showProgress = TRUE)
```

**Arguments**

- **year**: Numeric. Year of the data in YYYY format. Defaults to 2018.
- **simplified**: Logic FALSE or TRUE, indicating whether the function should return the data set with 'original' spatial resolution or a data set with 'simplified' geometry. Defaults to TRUE. For spatial analysis and statistics users should set simplified = FALSE. Borders have been simplified by removing vertices of borders using st_simplify(sf) preserving topology with a dTolerance of 100.
- **showProgress**: Logical. Defaults to TRUE display progress bar.

**Value**

An "sf" "data.frame" object
**read_micro_region**

**Description**

Data at scale 1:250,000, using Geodetic reference system "SIRGAS2000" and CRS(4674)

**Usage**

```r
read_micro_region(
  code_micro = "all",
  year = 2010,
  simplified = TRUE,
  showProgress = TRUE
)
```

**Arguments**

<table>
<thead>
<tr>
<th>Argument</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>code_micro</td>
<td>5-digit code of a micro region. If the two-digit code or a two-letter uppercase abbreviation of a state is passed, (e.g. 33 or &quot;RJ&quot;) the function will load all micro regions of that state. If code_micro=&quot;all&quot; (Default), the function downloads all micro regions of the country.</td>
</tr>
<tr>
<td>year</td>
<td>Numeric. Year of the data in YYYY format. Defaults to 2010.</td>
</tr>
<tr>
<td>simplified</td>
<td>Logic FALSE or TRUE, indicating whether the function should return the data set with 'original' spatial resolution or a data set with 'simplified' geometry. Defaults to TRUE. For spatial analysis and statistics users should set simplified = FALSE. Borders have been simplified by removing vertices of borders using st_simplify(sf) preserving topology with a dTolerance of 100.</td>
</tr>
<tr>
<td>showProgress</td>
<td>Logical. Defaults to TRUE display progress bar.</td>
</tr>
</tbody>
</table>

**Examples**

```r
# Read all official metropolitan areas for a given year
m <- read_metro_area(2005)

m <- read_metro_area(2018)
```

---

**See Also**

Other area functions: read_amazon(), read_biomes(), read_capitals(), read_comparable_areas(), read_country(), read_disaster_risk_area(), read_health_facilities(), read_health_region(), read_immediate_region(), read_indigenous_land(), read_intermediate_region(), read_meso_region(), read_micro_region(), read_municipal_seat(), read_municipality(), read_neighborhood(), read_pop_arrangements(), read_region(), read_schools(), read_semiarid(), read_state(), read_statistical_grid(), read_urban_area(), read_urban_concentrations(), read_weighting_area()
read_municipality

Description

Download spatial data of Brazilian municipalities

Usage

read_municipality(
  code_muni = "all",
  year = 2010,
  simplified = TRUE,
  showProgress = TRUE
)

Arguments

code_muni: The 7-digit identification code of a municipality. If code_muni = "all" (Default), the function downloads all municipalities of the country. Alternatively, if a two-digit identification code or a two-letter uppercase abbreviation of a state is
passed (e.g. 33 or "RJ"), all municipalities of that state will be downloaded. Mu-
nicipality identification codes can be consulted with the geobr::lookup_muni() function.

**year**

Numeric. Year of the data in YYYY format. Defaults to 2010.

**simplified**

Logic FALSE or TRUE, indicating whether the function should return the data set with 'original' spatial resolution or a data set with 'simplified' geometry. Defaults to TRUE. For spatial analysis and statistics users should set simplified = FALSE. Borders have been simplified by removing vertices of borders using st_simplify(sf) preserving topology with a dTolerance of 100.

**showProgress**

Logical. Defaults to TRUE display progress bar.

**Value**

An "sf" "data.frame" object

**See Also**

Other area functions: read_amazon(), read_biomes(), read_capitals(), read_comparable_areas(), read_country(), read_disaster_risk_area(), read_health_facilities(), read_health_region(), read_immediate_region(), read_indigenous_land(), read_intermediate_region(), read_meso_region(), read_metro_area(), read_micro_region(), read_municipal_seat(), read_neighborhood(), read_pop_arrangements(), read_region(), read_schools(), read_semiarid(), read_state(), read_statistical_grid(), read_urban_area(), read_urban_concentrations(), read_weighting_area()

**Examples**

```r
# Read specific municipality at a given year
mun <- read_municipality(code_muni = 1200179, year = 2017)

# Read all municipalities of a state at a given year
mun <- read_municipality(code_muni = 33, year = 2010)
mun <- read_municipality(code_muni = "RJ", year = 2010)

# Read all municipalities of the country at a given year
mun <- read_municipality(code_muni = "all", year = 2018)
```

**Description**

This function reads the official data on the municipal seats (sede dos municipios) of Brazil. The data brings the geographical coordinates (lat lon) of municipal seats for various years between 1872 and 2010. Original data were generated by Brazilian Institute of Geography and Statistics (IBGE).
read_neighborhood

Usage

```r
read_municipal_seat(year = 2010, showProgress = TRUE)
```

Arguments

- **year**: Numeric. Year of the data in YYYY format. Defaults to 2010.
- **showProgress**: Logical. Defaults to TRUE display progress bar.

Value

An "sf" "data.frame" object

See Also

Other area functions: `read_amazon()`, `read_biomes()`, `read_capitals()`, `read_comparable_areas()`, `read_country()`, `read_disaster_risk_area()`, `read_health_facilities()`, `read_health_region()`, `read_immediate_region()`, `read_indigenous_land()`, `read_intermediate_region()`, `read_meso_region()`, `read_metro_area()`, `read_micro_region()`, `read_municipality()`, `read_neighborhood()`, `read_pop_arrangements()`, `read_regional()`, `read_schools()`, `read_semiarid()`, `read_state()`, `read_statistical_grid()`, `read_urban_area()`, `read_urban_concentrations()`, `read_weighting_area()`

Examples

```r
# Read municipal seats in an specific year
m <- read_municipal_seat(year = 1991)
```

---

**Description**

This data set includes the neighborhood limits of 720 Brazilian municipalities. It is based on aggregations of the census tracts from the Brazilian census. Only 2010 data is currently available.

**Usage**

```r
read_neighborhood(year = 2010, simplified = TRUE, showProgress = TRUE)
```
Arguments

year  Numeric. Year of the data in YYYY format. Defaults to 2010.
simplified  Logic FALSE or TRUE, indicating whether the function should return the data set with 'original' spatial resolution or a data set with 'simplified' geometry. Defaults to TRUE. For spatial analysis and statistics users should set simplified = FALSE. Borders have been simplified by removing vertices of borders using st_simplify{sf} preserving topology with a dTolerance of 100.
showProgress  Logical. Defaults to TRUE display progress bar.

Value

An "sf" "data.frame" object

See Also

Other area functions: read_amazon(), read_biomes(), read_capitals(), readComparable_areas(), read_country(), read_disaster_risk_area(), read_health_facilities(), read_health_region(), read_immediate_region(), read_indigenous_land(), read_intermediate_region(), read_meso_region(), read_metro_area(), read_micro_region(), read_municipal_seat(), read_municipality(), read_pop_arrangements(), read_region(), read_schools(), read_semiarid(), read_state(), read_statistical_grid(), read_urban_area(), read_urban_concentrations(), read_weighting_area()

Examples

# Read neighborhoods of Brazilian municipalities
n <- read_neighborhood(year=2010)

---

read_pop_arrangements  Download population arrangements in Brazil

Description

This function reads the official data on population arrangements (Arranjos Populacionais) of Brazil. Original data were generated by the Institute of Geography and Statistics (IBGE) For more information about the methodology, see details at https://www.ibge.gov.br/apps/arranjos_populacionais/2015/pdf/publicacao.pdf

Usage

read_pop_arrangements(year = 2015, simplified = TRUE, showProgress = TRUE)
Arguments

- **year**: Numeric. Year of the data in YYYY format. Defaults to 2015.
- **simplified**: Logic FALSE or TRUE, indicating whether the function should return the data set with 'original' spatial resolution or a data set with 'simplified' geometry. Defaults to TRUE. For spatial analysis and statistics users should set simplified = FALSE. Borders have been simplified by removing vertices of borders using st_simplify{sf} preserving topology with a dTolerance of 100.
- **showProgress**: Logical. Defaults to TRUE display progress bar.

Value

An "sf" "data.frame" object

See Also

Other area functions: read_amazon(), read_biomes(), read_capitals(), read_comparable_areas(), read_country(), read_disaster_risk_area(), read_health_facilities(), read_health_region(), read_immediate_region(), read_indigenous_land(), read_intermediate_region(), read_meso_region(), read_metro_area(), read_micro_region(), read_municipal_seat(), read_municipality(), read_neighborhood(), read_region(), read_schools(), read_semiarid(), read_state(), read_statistical_grid(), read_urban_area(), read_urban_concentrations(), read_weighting_area()

Examples

```r
# Read urban footprint of Brazilian cities in an specific year
uc <- read_pop_arrangements(year=2015)
```

Description

Data at scale 1:250,000, using Geodetic reference system "SIRGAS2000" and CRS(4674)

Usage

```r
read_region(year = 2010, simplified = TRUE, showProgress = TRUE)
```

Arguments

- **year**: Numeric. Year of the data in YYYY format. Defaults to 2010.
- **simplified**: Logic FALSE or TRUE, indicating whether the function should return the data set with 'original' spatial resolution or a data set with 'simplified' geometry. Defaults to TRUE. For spatial analysis and statistics users should set simplified = FALSE. Borders have been simplified by removing vertices of borders using st_simplify{sf} preserving topology with a dTolerance of 100.
showProgress  Logical. Defaults to TRUE display progress bar.

Value
An "sf" "data.frame" object

See Also
Other area functions: read_amazon(), read_biomes(), read_capitals(), read_comparable_areas(), read_country(), read_disaster_risk_area(), read_health_facilities(), read_health_region(), read_immmediate_region(), read_indigenous_land(), read_intermediate_region(), read_meso_region(), read_metro_area(), read_micro_region(), read_municipal_seat(), read_municipality(), read_neighborhood(), read_pop_arrangements(), read_schools(), read_semiarid(), read_state(), read_statistical_grid(), read_urban_area(), read_urban_concentrations(), read_weighting_area()

Examples
# Read specific year
reg <- read_region(year=2018)

Description
Data comes from the School Census collected by INEP, the National Institute for Educational Studies and Research Anisio Teixeira. The date of the last data update is registered in the database in the column 'date_update'. These data uses Geodetic reference system "SIRGAS2000" and CRS(4674). The coordinates of each school if collected by INEP. Periodically the coordinates are revised with the objective of improving the quality of the data. More information available at https://www.gov.br/inep/pt-br/acesso-a-informacao/dados-abertos/inep-data/catalogo-de-escolas/

Usage
read_schools(year = 2020, showProgress = TRUE)

Arguments
year Numeric. Year of the data in YYYY format. Defaults to 2020.
showProgress Logical. Defaults to TRUE display progress bar.

Value
An "sf" "data.frame" object
See Also

Other area functions: `read_amazon()`, `read_biomes()`, `read_capitals()`, `read_comparable_areas()`, `read_country()`, `read_disaster_risk_area()`, `read_health_facilities()`, `read_health_region()`, `read_immediate_region()`, `read_indigenous_land()`, `read_intermediate_region()`, `read_meso_region()`, `read_metro_area()`, `read_micro_region()`, `read_municipal_seat()`, `read_municipality()`, `read_neighborhood()`, `read_pop_arrangements()`, `read_region()`, `read_semiarid()`, `read_state()`, `read_statistical_grid()`, `read_urban_area()`, `read_urban_concentrations()`, `read_weighting_area()`

Examples

```r
# Read all schools in the country
s <- read_schools(year = 2020)
```

Description

This data set covers the whole of Brazilian Semiarid as defined in the resolution in 23/11/2017). The original data comes from the Brazilian Institute of Geography and Statistics (IBGE) and can be found at [https://www.ibge.gov.br/geociencias/cartas-e-mapas/mapas-regionais/15974-semiarido-brasileiro.html?=&t=downloads](https://www.ibge.gov.br/geociencias/cartas-e-mapas/mapas-regionais/15974-semiarido-brasileiro.html?=&t=downloads)

Usage

```r
read_semiarid(year = 2017, simplified = TRUE, showProgress = TRUE)
```

Arguments

- `simplified` Logic FALSE or TRUE, indicating whether the function should return the data set with 'original' spatial resolution or a data set with 'simplified' geometry. Defaults to TRUE. For spatial analysis and statistics users should set simplified = FALSE. Borders have been simplified by removing vertices of borders using `st_simplify(sf)` preserving topology with a `dTolerance` of 100.
- `showProgress` Logical. Defaults to TRUE display progress bar.

Value

An "sf" "data.frame" object
**See Also**

Other area functions: `read_amazon()`, `read_biomes()`, `read_capitals()`, `read_comparable_areas()`, `read_country()`, `read_disaster_risk_area()`, `read_health_facilities()`, `read_health_region()`, `read_immediate_region()`, `read_indigenous_land()`, `read_intermediate_region()`, `read_meso_region()`, `read_metro_area()`, `read_micro_region()`, `read_municipal_seat()`, `read_municipality()`, `read_neighborhood()`, `read_pop_arrangements()`, `read_region()`, `read_schoools()`, `read_state()`, `read_immediate_region()`, `read_indigenous_land()`, `read_intermediate_region()`, `read_meso_region()`, `read_metro_area()`, `read_micro_region()`, `read_municipal_seat()`, `read_municipality()`, `read_neighborhood()`, `read_pop_arrangements()`, `read_region()`, `read_schoools()`, `read_state()`, `read_neighborhood()`, `read_pop_arrangements()`, `read_region()`, `read_schoools()`, `read_state()`, `read_statistical_grid()`, `read_urban_area()`, `read_urban_concentrations()`, `read_weighting_area()`

**Examples**

```r
# Read Brazilian semiarid
a <- read_semiarid(year=2017)
```

**Description**

Data at scale 1:250,000, using Geodetic reference system "SIRGAS2000" and CRS(4674)

**Usage**

```r
read_state(
  code_state = "all",
  year = 2010,
  simplified = TRUE,
  showProgress = TRUE
)
```

**Arguments**

- `code_state` The two-digit code of a state or a two-letter uppercase abbreviation (e.g. 33 or "RJ"). If `code_state="all"` (the default), the function downloads all states.
- `year` Numeric. Year of the data in YYYY format. Defaults to 2010.
- `simplified` Logic FALSE or TRUE, indicating whether the function should return the data set with 'original' spatial resolution or a data set with 'simplified' geometry. Defaults to TRUE. For spatial analysis and statistics users should set simplified = FALSE. Borders have been simplified by removing vertices of borders using `st_simplify(sf)` preserving topology with a `dTolerance` of 100.
- `showProgress` Logical. Defaults to TRUE display progress bar.

**Value**

An "sf" "data.frame" object
**See Also**

Other area functions: `read_amazon()`, `read_biomes()`, `read_capitals()`, `read_comparable_areas()`, `read_country()`, `read_disaster_risk_area()`, `read_health_facilities()`, `read_health_region()`, `read_immediate_region()`, `read_indigenous_land()`, `read_intermediate_region()`, `read_meso_region()`, `read_metro_area()`, `read_micro_region()`, `read_municipal_seat()`, `read_municipality()`, `read_neighborhood()`, `read_pop_arrangements()`, `read_region()`, `read_schoools()`, `read_semiarid()`, `read_statistical_grid()`, `read_urban_area()`, `read_urban_concentrations()`, `read_weighting_area()`

**Examples**

```r
# Read specific state at a given year
uf <- read_state(code_state=12, year=2017)

# Read specific state at a given year
uf <- read_state(code_state="SC", year=2000)

# Read all states at a given year
ufs <- read_state(code_state="all", year=2010)
```

**Description**

Data at scale 1:250,000, using Geodetic reference system "SIRGAS2000" and CRS(4674)

**Usage**

```r
read_statistical_grid(code_grid, year = 2010, showProgress = TRUE)
```

**Arguments**

- `code_grid`  
  If two-letter abbreviation or two-digit code of a state is passed, the function will load all grid quadrants that intersect with that state. If `code_grid="all"`, the grid of the whole country will be loaded. Users may also pass a grid quadrant id to load an specific quadrant. Quadrant ids can be consulted at geobr::grid_state_correspondence_table.

- `year`  
  Numeric. Year of the data in YYYY format. Defaults to 2010. The only year available thus far is 2010.

- `showProgress`  
  Logical. Defaults to TRUE display progress bar.

**Value**

An "sf" "data.frame" object
See Also

Other area functions: `read_amazon()`, `read_biomes()`, `read_capitals()`, `read_comparable_areas()`, `read_country()`, `read_disaster_risk_area()`, `read_health_facilities()`, `read_health_region()`, `read_immediate_region()`, `read_indigenous_land()`, `read_intermediate_region()`, `read_meso_region()`, `read_metro_area()`, `read_micro_region()`, `read_municipal_seat()`, `read_municipality()`, `read_neighborhood()`, `read_pop_arrangements()`, `read_region()`, `read_schools()`, `read_semiarid()`, `read_state()`, `read_urban_area()`, `read_urban_concentrations()`, `read_weighting_area()`

Examples

```r
# Read a particular grid at a given year
grid <- read_statistical_grid(code_grid = 45, year = 2010)

# Read the grid covering a given state at a given year
state_grid <- read_statistical_grid(code_grid = "RJ")
```

Description

This function reads the official data on the urban footprint of Brazilian cities in the years 2005 and 2015. Original data were generated by the Institute of Geography and Statistics (IBGE). For more information about the methodology, see details at https://biblioteca.ibge.gov.br/visualizacao/livros/liv100639.pdf

Usage

```r
read_urban_area(year = 2015, simplified = TRUE, showProgress = TRUE)
```

Arguments

- **year**: Numeric. Year of the data in YYYY format. Defaults to 2015.
- **simplified**: Logic FALSE or TRUE, indicating whether the function should return the data set with 'original' spatial resolution or a data set with 'simplified' geometry. Defaults to TRUE. For spatial analysis and statistics users should set simplified = FALSE. Borders have been simplified by removing vertices of borders using `st_simplify(sf)` preserving topology with a dTolerance of 100.
- **showProgress**: Logical. Defaults to TRUE display progress bar.

Value

An "sf" "data.frame" object
See Also

Other area functions: `read_amazon()`, `read_biomes()`, `read_capitals()`, `read_comparable_areas()`, `read_country()`, `read_disaster_risk_area()`, `read_health_facilities()`, `read_health_region()`, `read_indigenous_land()`, `read_intermediate_region()`, `read_meso_region()`, `read_metro_area()`, `read_micro_region()`, `read_municipal_seat()`, `read_municipality()`, `read_neighborhood()`, `read_pop_arrangements()`, `read_region()`, `read_schools()`, `read_semiarid()`, `read_state()`, `read_statistical_grid()`, `read_urban_concentrations()`, `read_weighting_area()`

Examples

```r
# Read urban footprint of Brazilian cities in an specific year
d <- read_urban_area(year=2005)
```

Description

This function reads the official data on the urban concentration areas (Areas de Concentracao de Populacao) of Brazil. Original data were generated by the Institute of Geography and Statistics (IBGE) For more information about the methodology, see details at https://www.ibge.gov.br/apps/arranjos_populacionais/2015/pdf/publicacao.pdf

Usage

```r
read_urban_concentrations(year = 2015, simplified = TRUE, showProgress = TRUE)
```

Arguments

- `year` Numeric. A year number in YYYY format. Defaults to 2015.
- `simplified` Logic FALSE or TRUE, indicating whether the function should return the data set with 'original' spatial resolution or a data set with 'simplified' geometry. Defaults to TRUE. For spatial analysis and statistics users should set simplified = FALSE. Borders have been simplified by removing vertices of borders using st_simplify(sf) preserving topology with a dTolerance of 100.
- `showProgress` Logical. Defaults to TRUE display progress bar.

Value

An "sf" "data.frame" object
read_weighting_area

See Also

Other area functions: read_amazon(), read_biomes(), read_capitals(), read_comparable_areas(), read_country(), read_disaster_risk_area(), read_health_facilities(), read_health_region(), readImmediate_region(), read_indigenous_land(), read_intermediate_region(), read_meso_region(), read_metro_area(), read_micro_region(), read_municipal_seat(), read_municipality(), read_neighborhood(), read_pop_arrangements(), read_region(), read_schools(), read_semiarid(), read_state(), read_statistical_grid(), read_urban_area(), read_weighting_area()

Examples

# Read urban footprint of Brazilian cities in an specific year
uc <- read_urban_concentrations(year=2015)

read_weighting_area

Download spatial data of Census Weighting Areas (area de ponderacao) of the Brazilian Population Census

Description

Only 2010 data is currently available.

Usage

read_weighting_area(
  code_weighting = "all",
  year = 2010,
  simplified = TRUE,
  showProgress = TRUE
)

Arguments

code_weighting The 7-digit code of a Municipality. If the two-digit code or a two-letter uppercase abbreviation of a state is passed, (e.g. 33 or "RJ") the function will load all weighting areas of that state. If code_weighting="all", all weighting areas of the country are loaded.

year Numeric. Year of the data. Defaults to 2010.

simplified Logic FALSE or TRUE, indicating whether the function should return the data set with 'original' spatial resolution or a data set with 'simplified' geometry. Defaults to TRUE. For spatial analysis and statistics users should set simplified = FALSE. Borders have been simplified by removing vertices of borders using st_simplify(sf) preserving topology with a dTolerance of 100.

showProgress Logical. Defaults to TRUE display progress bar.
Value

An "sf" "data.frame" object

See Also

Other area functions: read_amazon(), read_biomes(), read_capitals(), read_comparable_areas(), read_country(), read_disaster_risk_area(), read_health_facilities(), read_health_region(), read_immediate_region(), read_indigenous_land(), read_intermediate_region(), read_meso_region(), read_metro_area(), read_micro_region(), read_municipal_seat(), read_municipality(), read_neighborhood(), read_pop_arrangements(), read_region(), read_schools(), read_semiarid(), read_state(), read_statistical_grid(), read_urban_area(), read_urban_concentrations()

Examples

# Read specific weighting area at a given year
w <- read_weighting_area(code_weighting=5201108005004, year=2010)

# Read all weighting areas of a state at a given year
w <- read_weighting_area(code_weighting=53, year=2010) # or
w <- read_weighting_area(code_weighting="DF", year=2010)
plot(w)

# Read all weighting areas of a municipality at a given year
w <- read_weighting_area(code_weighting=5201108, year=2010)
plot(w)

# Read all weighting areas of the country at a given year
w <- read_weighting_area(code_weighting="all", year=2010)
Index

* area functions
  read_amazon, 6
  read_biomes, 7
  read_capitals, 8
  read_comparable_areas, 10
  read_country, 12
  read_disaster_risk_area, 13
  read_health_facilities, 14
  read_health_region, 15
  read_immediate_region, 16
  read_indigenous_land, 17
  read_intermediate_region, 18
  read_meso_region, 20
  read_metro_area, 21
  read_micro_region, 22
  read_municipal_seat, 24
  read_municipality, 23
  read_neighborhood, 25
  read_pop_arrangements, 26
  read_region, 27
  read_schools, 28
  read_semiarid, 29
  read_state, 30
  read_statistical_grid, 31
  read_urban_area, 32
  read_urban_concentrations, 33
  read_weighting_area, 34

* cep functions
  cep_to_state, 3

* datasets
  grid_state_correspondence_table, 3

* general area functions
  read_census_tract, 9
  read_conservation_units, 11

* support functions
  list_geobr, 4
  lookup_muni, 5

cep_to_state, 3
read_region, 6–8, 11–14, 16–20, 22–27, 27, 29–35
read_schools, 6–8, 11–14, 16–20, 22–28, 28, 30–35
read_semiarid, 6–8, 11–14, 16–20, 22–29, 29, 31–35
read_state, 6–8, 11–14, 16–20, 22–30, 30, 32–35
read_statistical_grid, 6–8, 11–14, 16–20, 22–31, 31, 33–35
read_urban_area, 6–8, 11–14, 16–20, 22–32, 32, 34, 35
read_urban_concentrations, 6–8, 11–14, 16–20, 22–33, 33, 35
read_weighting_area, 6–8, 11–14, 16–20, 22–34, 34