Package 'INEbaseR'

June 19, 2019

Type Package

Title R package for obtaining and analyzing open data from INE (Instituto Nacional de Estadistica) API

Version 0.1.0

Date 2018-01-01

Maintainer Andres Nacimiento <andresnacimiento@gmail.com>

Description R package for obtaining and analyzing open data from INE (Instituto Nacional de Estadistica) API.

License GPL (>= 3)

Encoding UTF-8

LazyData false

BuildResaveData false

Imports jsonlite, highcharter

RoxygenNote 6.1.1

NeedsCompilation no

Author Andres Nacimiento [aut, prg, cre],
Mariano Sanz [ctg],
Carlos J. Perez [aut, drt, pdr]

Depends R (>= 3.5.0)

R topics documented:

`convert_natcode_to_geocode` *Convert Natcode to Geocode*

Description

This function allows converting from natcode to geocode and vice-versa. If all params are null, you will get the complete table used for codes conversion.

Usage

```
convert_natcode_to_geocode(natcode = NULL, geocode = NULL,
  exponential_notation = FALSE)
```

Arguments

`natcode`	(int) geographical code from INE
`geocode`	(string) geographical code from Eurostat
`exponential_notation`	(boolean) to show or not exponential notation. `e.g. e+10`

Examples

```
convert_natcode_to_geocode()
convert_natcode_to_geocode(natcode = 34050000000)
convert_natcode_to_geocode(geocode = "ES70")
```

`draw_serie` *Draw serie*

Description

This function allows representing series data into a map

Usage

```
draw_serie(serie, nult = 0, classification = NULL, map_scale = 60,
  verbose = FALSE, benchmark = FALSE)
```

Arguments

serie	(string) serie id
nult	(int) last n serie data, if nult = 0 this value will be auto-calculated
classification	(string) serie classification, if classification = NULL this value will be auto-detected
map_scale	(int) refers to the relationship or ratio between distance on a map and the corresponding distance on the ground. For example, on a 1:1000000 scale map, 1cm on the map equals 1km on the ground. Possible values are: 1,3,10,20 or 60, and it's only for PROV or CCAA geographical granularity, map_scale = 60 by default and map_scale = NULL for high detailed map.
verbose	(boolean) show more information during the process
benchmark	(boolean) used to measure the performance of the system, benchmark = FALSE by default.

Examples

```
draw_serie("IPC251521")
draw_serie("IPC251541")
draw_serie("UA42121")
```

---

get_cache_rds *Get rds content (cache)*

Description

This function returns the content of a RDS file in cache

Usage

```
get_cache_rds(object, type = "SERIEOPERATION")
```

Arguments

object	(string) an object of the "type" option
type	(string) type of content do you want to read, type = "/POLYGONS-" by default

Examples

```
get_cache_rds("provincias", type = "POLYGONS")
get_cache_rds("comunidades_autonomas", type = "POLYGONS")
get_cache_rds("municipios", type = "POLYGONS")
get_cache_rds("natcodes", type = "DATATABLE")
get_cache_rds(4, type = "SERIEOPERATION")
```

get_geographical_variable *Get geographical variable*

Description

This function returns the geographical variable of a serie

Usage

```
get_geographical_variable(serie)
```

Arguments

serie	(string) serie id

Examples

```
get_geographical_variable("IPC251522")
```

get_metadata_crossing *Get metadata crossing*

Description

This function returns data or metadata by metadata crossing

Usage

```
get_metadata_crossing(code = NULL, resource = "series", help = FALSE,
  query = NULL, p = 1, det = 0, tip = NULL, nlast = 1,
  lang = "ES")
```

Arguments

code	(string) operation identificator
resource	(string) resource to access, by default resource = "metadata" to get serie metadata. Possible values are series or data
help	(boolean) type any value for resource param and type help = TRUE to see params available for this resource.
query	(string) string separated by AND with syntax variable = value using natural language
p	(int) periodicity, p = 1 by default
det	(int) det = 2 to see two levels of depth, specifically to access the PubFechaAct object, det = 0 by default
tip	(string) tip = M to obtain the metadata (crossing variables-values) of the series
nlast	last n values
lang	(string) language used to obtain information

Examples

```
get_metadata_crossing()
get_metadata_crossing(resource = "series", help = TRUE)
get_metadata_crossing("IPC", resource = "series", query = "Provincias = Madrid AND Tipo de dato = Variación mens
get_metadata_crossing("IPC", resource = "data", query = "Provincias = Madrid AND Tipo de dato = Variacion mensua
```

---

get_natcode *Get natcode*

Description

This function allows get all natcodes or calculate a natcode from a serie and a geographical granularity

Usage

```
get_natcode(serie = NULL, all = TRUE, verbose = TRUE)
```

Arguments

serie	(string) serie identificator
all	(bool) if all = TRUE you will get all natcodes
verbose	(boolean) show more information during the process

Examples

```
get_natcode()
get_natcode("IPC251522")
get_natcode("IPC251541")
get_natcode("DPOP37286")
```

---

get_operations *Get operations*

Description

This function returns information about operations

Usage

```
get_operations(code = NULL, resource = "all", help = FALSE,
  ioe = FALSE, geographical_granularity = NULL,
  temporal_granularity = NULL, verbose = TRUE, lang = "ES")
```

Arguments

code	(string) operation identificator
resource	(string) resource to access, by default resource = "metadata" to get serie metadata. Possible values are all,metadata,variables_values or by_granularity
help	(boolean) type any value for resource param and type help = TRUE to see params available for this resource.
ioe	(boolean) TRUE if code is in format IO30138, and FALSE by default
geographical_granularity	(string) geographical granularity
temporal_granularity	(string) temporal granularity
verbose	(boolean) show more information during the process
lang	(string) language used to obtain information

Examples

```
get_operations()
get_operations(resource = "all")
get_operations(resource = "all", help = TRUE)
get_operations("IPC", resource = "metadata")
get_operations("IPC", resource = "variables_values")
get_operations(resource = "by_granularity", geographical_granularity = "PROV")
```

---

get_publications *Get publications*

Description

This function returns a data frame with all available publications from an id or code

Usage

```
get_publications(det = 0, lang = "ES")
```

Arguments

det	det = 2 to see two levels of depth, specifically to access the PubFechaAct object, det = 0 by default
lang	language used to obtain information

Examples

```
get_publications()
```

get_publications_date *Get publications date*

Description

This function returns a data frame with all publication date of a publication from an id or code

Usage

```
get_publications_date(code, det = 0, lang = "ES")
```

Arguments

code	publication identification
det	det = 2 to see two levels of depth, specifically to access the PubFechaAct object, det = 0 by default
lang	language used to obtain information

Examples

```
get_publications_date(8)
```

get_publications_operation *Get publications operation*

Description

This function returns a data frame with publications of an operation from an id or code

Usage

```
get_publications_operation(code, ioe = FALSE, det = 0, lang = "ES")
```

Arguments

code	operation identification
ioe	TRUE if code is in format IO30138, and FALSE by default
det	det = 2 to see two levels of depth, specifically to access the PubFechaAct object, det = 0 by default
lang	language used to obtain information

Details

Numeric code id Alphabetic code IPC IOE code (Inventario de Operaciones Estadísticas)

Examples

```
get_publications_operation(25)
get_publications_operation(30138, ioe = TRUE)
```

get_series *Get series*

Description

This function returns data or metadata from an operation, table or a serie

Usage

```
get_series(code = NULL, resource = "metadata", help = FALSE,
  ioe = FALSE, det = 0, tip = NULL, lang = "ES",
  date_start = NULL, date_end = NULL, nlast = NULL,
  classification = NULL, verbose = FALSE, benchmark = FALSE,
  geographical_granularity = NULL, temporal_granularity = NULL)
```

Arguments

code	(string) serie, operation or table identificator
resource	(string) resource to access, by default resource = "metadata" to get serie metadata. Possible values are metadata,operation,values,table,metadataoperation,data,by_gra or nlast
help	(boolean) type any value for resource param and type help = TRUE to see params available for this resource.
ioe	(boolean) TRUE if code is in format IO30138, and FALSE by default
det	(int) det = 2 to see two levels of depth, specifically to access the PubFechaAct object, det = 0 by default
tip	(string) tip = M to obtain the metadata (crossing variables-values) of the series
lang	(string) language used to obtain information
date_start	(string) start date in format YYYY-MM-DD
date_end	(string) end date in format YYYY-MM-DD
nlast	(int) last n serie values
classification	(string) serie classification, if classification = NULL this value will be autodetected
verbose	(boolean) to show more information about this process, verbose = FALSE by default
benchmark	(boolean) used to measure the performance of the system, benchmark = FALSE by default.
geographical_granularity	(string) geographical granularity
temporal_granularity	(string) temporal granularity

Examples

```
get_series("IPC206449")
get_series(resource = "metadata", help = TRUE)
get_series("IPC", resource = "operation")
get_series("IPC206449", resource = "values")
get_series(22350, resource = "table")
get_series("IPC251541", resource = "nlast")
get_series("IPC206449", resource = "data", nlast = 5)
get_series("IPC", resource = "by_granularity", geographical_granularity = "CCAA", verbose = TRUE)
get_series("IPC251539", resource = "by_common_parameters", verbose = TRUE)
```

`get_tables` *Get tables*

Description

This function returns data / metadata from tables

Usage

```
get_tables(code = NULL, resource = "operation", help = FALSE,
  grp = NULL, geo = 0, nlast = 0, det = 0, tip = NULL,
  ioe = FALSE, lang = "ES")
```

Arguments

code	operation (string/int) or table (int) identificator
resource	(string) resource to access, by default resource = "operation" to get tables of an operation. Possible values are operation,group,group_values or data
help	(boolean) type any value for resource param and type help = TRUE to see params available for this resource
grp	(int) group identification
geo	(int) use geo = 1 to access only tables with geographic content, geo = 0 by default
nlast	last n values
det	(int) use det = 2 to see two levels of depth, specifically to access the PubFechaAct object, det = 0 by default
tip	(string) use tip = "A" to view as friendly, specifically the view the field ultima_modificacion or use tip = "AM" to obtain the metadata (crossing variables-values) of the series and a friendly output.
ioe	(boolean) use ioe = TRUE if code is in format IO30138, and FALSE by default
lang	(string) language used to obtain information

Examples

```
get_tables("IPC")
get_tables(resource = "operation", help = TRUE)
get_tables(25, resource = "operation")
get_tables(22350, resource = "group")
get_tables(22350, grp = 81497, resource = "group_values")
get_tables(22350, nlast = 5, resource = "data")
```

get_values_all *Get values (all)*

Description

This function returns a data frame with all values from a variable

Usage

```
get_values_all(id, det = 0, lang = "ES")
```

Arguments

id	operation identification
det	det = 1 to see the detail of the variable to which it belongs, det = 0 by default
lang	language used to obtain information

Examples

```
get_values_all(115)
get_values_all(115, 1)
```

get_values_variableoperation *Get values from variable operation*

Description

This function returns a data frame with all values from a variable to an operation

Usage

```
get_values_variableoperation(id, op, det = 0, ioe = FALSE,
  lang = "ES")
```

Arguments

id	variable identification
op	operation identification
det	det = 1 to see the detail of the variable to which it belongs, det = 0 by default
ioe	TRUE if code is in format IO30138, and FALSE by default
lang	language used to obtain information

Details

Numeric code id Alphabetic code IPC IOE code (Inventario de Operaciones Estadísticas)

Examples

```
get_values_variableoperation(762, 25)
get_values_variableoperation(762, 25, 1)
get_values_variableoperation(762, 30138, ioe = TRUE)
get_values_variableoperation(762, 30138, ioe = TRUE, 1)
```

`get_variables_all` *Get variables (all)*

Description

This function returns a data frame with all system variables

Usage

```
get_variables_all(lang = "ES")
```

Arguments

`lang`	language used to obtain information

Examples

```
get_variables_all()
```

`get_variables_operation` *Get variable operation*

Description

This function returns a data frame with system variables of an operation from an id or code

Usage

```
get_variables_operation(operation, ioe = FALSE, lang = "ES")
```

Arguments

`operation`	operation identifier
`ioe`	`TRUE` if code is in format `IO30138`, and `FALSE` by default
`lang`	language used to obtain information

Examples

```
get_variables_operation(operation = 25)
get_variables_operation(operation = "IPC")
get_variables_operation(operation = 30138, ioe = TRUE)
```

highcharts_series *Highcharts series*

Description

This function draws a highchart with data of a series from an id and/or from a date or date range

Usage

```
highcharts_series(code, date_start = NA, date_end = NA, nult = 0,
  det = 0, lang = "ES")
```

Arguments

code	identification code of a serie (e.g. "IPC206449")
date_start	start date in format (string) YYYY-MM-DD
date_end	end date in format (string) YYYY-MM-DD
nult	last n values
det	det = 2 to see two levels of depth, specifically to access the PubFechaAct object, det = 0 by default
lang	language used to obtain information

Examples

```
highcharts_series("IPC206449", nult = 1) # Get the latest data of a series
highcharts_series("IPC206449", nult = 5) # Get the \code{n} last data of a series
highcharts_series("IPC206449", "2013-01-01", "2016-01-01") # Get data of a series between two dates
highcharts_series("IPC206449", "2010-01-01") # Get data from a series from a date
```

plot_series *Plot series*

Description

This function draws a plot with data of a series from an id and/or from a date or date range

Usage

```
plot_series(code, date_start = NA, date_end = NA, nult = 0,
  det = 0, type = NA, lang = "ES")
```

Arguments

code	identification code of a serie (e.g. "IPC206449")
date_start	start date in format (string) YYYY-MM-DD
date_end	end date in format (string) YYYY-MM-DD
nult	last n values
det	det = 2 to see two levels of depth, specifically to access the PubFechaAct object, det = 0 by default
type	what type of plot should be drawn, type = "p" (for points) by default. See type in plot
lang	language used to obtain information

Examples

```
plot_series("IPC206449", nult = 1) # Get the latest data of a series
plot_series("IPC206449", nult = 5, type = "l") # Get the \code{n} last data of a series
plot_series("IPC206449", "2013-01-01", "2016-01-01") # Get data of a series between two dates
plot_series("IPC206449", "2010-01-01") # Get data from a series from a date
```

---

update_cache *Update cache*

Description

This function allow update specific or all cache data

Usage

```
update_cache(code = 0, n = 0, page = NA, pagination = TRUE,
  page_start = NA, page_end = NA, benchmark = TRUE, force = FALSE,
  ignore_series = NULL, tip = "M", det = 2)
```

Arguments

code	operation identificator
n	number of operation to update starting from first operation getted from get_operations_all() function.
page	page = 1 to obtain data of an specific page (to use this, pagination = FALSE).
pagination	TRUE to obtain data page by page and FALSE by default.
page_start	page_start = 1 start page range to obtain data (to use this, pagination = TRUE).
page_end	page_end = 2 end page range to obtain data (to use this, pagination = TRUE).
benchmark	used to measure the performance of the system, benchmark = FALSE by default.
force	(boolean) to force to update all cache data, force = FALSE by default.
ignore_series	(int) list of operation identificators to ignore. More slow series to cache are: 16, 49, 330 and 334
tip	tip = M to obtain the metadata (crossing variables-values) of the series.
det	det = 2 to see two levels of depth, specifically to access the PubFechaAct object, det = 0 by default

Examples

```
update_cache(code = 249)
update_cache(code = 249, page = 1)
update_cache(n = 3)
```

`update_series` *Update series*

Description

This function allow update specific or all cache data

Usage

```
update_series(serie = NULL, benchmark = TRUE, page = 1, tip = "M",
  det = 2, lang = "ES")
```

Arguments

`serie`	serie identificator
`benchmark`	used to measure the performance of the system, `benchmark = FALSE` by default.
`page`	`page = 1` to obtain data of an specific page.
`tip`	`tip = M` to obtain the metadata (crossing variables-values) of the series.
`det`	`det = 2` to see two levels of depth, specifically to access the `PubFechaAct` object, `det = 0` by default
`lang`	language used to obtain information

Examples

```
update_series()
```

Index