Package ‘gfer’

February 6, 2022

Type Package

Title Green Finance and Environmental Risk

Version 0.1.12

Date 2022-2-4

Description Focuses on data collecting, analyzing and visualization in green finance and environmental risk research and analysis. Main function includes environmental data collecting from official websites such as MEP (Ministry of Environmental Protection of China, \(<https://www.mee.gov.cn>\)), water related projects identification and environmental data visualization.

Encoding UTF-8

License GPL-2

LazyData true

Imports rvest, xml2, jsonlite, httr, stringi, V8, data.table, tidyr, scatterpie, ggplot2, ggrepel, circlize, googlesheets4, gsheets

Depends R (>= 2.10)

RoxygenNote 7.1.2

URL https://yuanchao-xu.github.io/gfer/

BugReports https://github.com/Yuanchao-Xu/gfer/issues

Repository CRAN

Suggests knitr, rmarkdown

VignetteBuilder knitr

NeedsCompilation no

Author Yuanchao Xu [aut, cre]

Maintainer Yuanchao Xu <xuyuanchao37@gmail.com>

Date/Publication 2022-02-06 15:30:02 UTC
R topics documented:

- checkHttpStatus .................................................. 3
- cm ................................................................. 3
- dataJson2df .......................................................... 4
- GDPmix .............................................................. 4
- genDfWds ............................................................. 5
- getCSRRating .......................................................... 5
- getCSRRating_unit .................................................. 6
- getENNames .......................................................... 7
- getENNames_unit .................................................... 8
- getExchange ........................................................... 8
- getHisMktCap .......................................................... 9
- getHisMktCap_unit ................................................... 10
- getIndex ............................................................... 11
- getIndexConstnt ..................................................... 12
- getIndexData ........................................................ 12
- getNBS ............................................................... 13
- getPPPList ............................................................ 13
- getPPPList_unit ..................................................... 14
- getProxy .............................................................. 15
- getStockList .......................................................... 15
- getTickers ............................................................ 16
- getTickers_unit ..................................................... 16
- getWaternomicsData_goog .......................................... 17
- getWaternomicsData_NBS .......................................... 17
- getWaterQ_MEP_all .................................................. 18
- getWaterQ_MEP_all_unit ........................................... 19
- is.listed .............................................................. 19
- milSec ............................................................... 20
- plotChord ............................................................ 20
- plotScatterPie ........................................................ 21
- statscnDbs ........................................................... 23
- statscnQueryData .................................................... 23
- statscnQueryLastN .................................................. 24
- statscnQueryZb ..................................................... 25
- statscnRegions ....................................................... 26
- statscnRowNamePrefix .............................................. 27

Index ................................................................. 28
**checkHttpStatus**

**Description**

private function for check the http status

**Usage**

`checkHttpStatus(ret)`

**Arguments**

`ret` the response obj returned by httr package

**Value**

return nothing , but if it finds some error , it stop the script

**References**


---

**cm**

Matrix showing complicated management of China’s Water Resource

**Description**

Matrix showing complicated management of China’s Water Resource

**Usage**

`cm`

**Format**

A data frame with 13 rows and 11 variables:

...
dataJson2df  

private function to convert the returned json data to a dataframe

Description

private function to convert the returned json data to a dataframe

Usage

dataJson2df(rawObj, rowcode, colcode)

Arguments

rawObj  
the fromJSON output

rowcode  
rowcode in the data frame

colcode  
colcode in the data frame

Value

the constructed data frame

References

https://CRAN.R-project.org/package=rstatscn

GDPmix  

Table about GDP mix of China provinces in 2015

Description

Table about GDP mix of China provinces in 2015

Usage

GDPmix

Format

A data frame with 11 rows and 7 variables:

...
**Description**

private function for constructing the query parameter for dfwds

**Usage**

```r
genDfwds(wdcode, valuecode)
```

**Arguments**

- `wdcode` : string value, one of c("zb", "sj", "reg")
- `valuecode` : string value, following is the table for available valuecode:
  - `zb`: the valuecode can be gotten by statscnQueryZb() function
  - `sj`: the valuecode can be "2014" for nd db, "2014C" for jd db.
  - `reg`: the valuecode is the region code fetched by statscnRegions(dbcode) function

**Value**

return the query string for the http request

**References**


---

**getCSRRating**

*get CSR rating from a website*

**Description**

get CSR rating from a website

**Usage**

```r
getCSRRating(startPage, endPage, year = 2015, proxy = FALSE)
```

**Arguments**

- `startPage` : on Which page you want to start, default is 1
- `endPage` : On which page you want to stop scrapping
- `year` : In which year you want the rank
- `proxy` : whether use the proxy, default is FALSE
Details
Get CSR ratings and reports of different companies from http://stockdata.stock.hexun.com/zrbg/

Value
A table of CSR ratings collected from your input page

References
www.hexun.com

Examples
## Not run:
# get first two pages of CSR ratings in 2015
getCSRRating(1,3)
## End(Not run)

getCSRRating_unit  get CSR rating from a website for a unit page

Description
get CSR rating from a website for a unit page

Usage
getCSRRating_unit(page, date, proxy = NULL)

Arguments
page on Which page you want to scrap
date represents the date is until which date, usually it’s the last day of a year e.g., "2015-12-31" for the date of year 2015, "2014-12-31" for the date of year 2014
proxy whether use the proxy, default is FALSE

Details
Get CSR ratings and reports of different companies from http://stockdata.stock.hexun.com/zrbg/

Value
A table of CSR ratings collected from your input page

References
www.hexun.com
getENNames

get a company’s EN names

Description

get a company’s EN names

Usage

getENNames(tickers)

Arguments

tickers ticker/sympol of a company, TICKERS MUST BE CHARACTERs, '006027' INSTEAD OF '6027' 

Details

Data comes from hexun.com

Value

A data table with companies’ EN names

References

http://hexun.com

Examples

## Not run:
getENNames(601857)

## End(Not run)
getENNames_unit  get a company’s English name

Description
get a company’s English name

Usage
getENNames_unit(ticker)

Arguments
ticker  ticker/symbol of a company, MUST BE A CHARACTER, ’006027’ INSTEAD OF ’6027’

Details
Data comes from hexun.com

Value
A data table with companies’ EN names

References
http://hexun.com

getExchange  get a company’s listed location

Description
get a company’s listed location

Usage
getExchange(tickers)

Arguments
tickers  ticker/symbol of a company, TICKERS MUST BE CHARACTERs, ’006027’ INSTEAD OF ’6027’

Details
Data comes from www.finance.sina.com.cn
getHisMktCap

Value

A data table with a listed companies’ ticker, security name and listed exchange location

References

www.finance.sina.com.cn

Examples

## Not run:
getExchange('600601')
getExchange(c('00005', '00001'))

## End(Not run)

gEthisMktCap | get a company’s historical market cap, data comes from NetEase

Description

get a company’s historical market cap, data comes from NetEase

Usage

gEthisMktCap(tickers, date1, date2)

Arguments

tickers | ticker/symbol of a company, TICKERS MUST BE CHARACTERs, '006027' INSTEAD OF '6027'
date1 | starting date, in the following format "20160101", means Jan 1st of 2016
date2 | ending date, in the following format "20160101", if you only want one day’s data, just set starting date and ending date the same day

Details

The input date interval should have at least one work day Data comes from www.money.163.com

Value

A data table with companies total capitalization and market capitalization

References

www.money.163.com
getHisMktCap_unit

Examples

```r
## Not run:
getHisMktCap(601857, '20161202', '20161203')

## End(Not run)
```

getHisMktCap_unit get a company's historical market cap, data comes from NetEase

Description

get a company's historical market cap, data comes from NetEase

Usage

```r
getHisMktCap_unit(ticker, date1, date2)
```

Arguments

- **ticker**: ticker/symbol of a company, MUST BE A CHARACTER, '006027' INSTEAD OF '6027'
- **date1**: starting date, in the following format "20160101", means Jan 1st of 2016
- **date2**: ending date, in the following format "20160101", if you only want one day's data, just set starting date and ending date the same day

Details

Data comes from www.money.163.com

Value

A data table with companies total capitalization and market capitalization

References

www.money.163.com
getIndex

get a company’s market cap, data comes from NetEase

Description

get a company’s market cap, data comes from NetEase

Usage

getIndex(tickers, indexData)

Arguments

tickers ticker/symbol of a company, MUST BE A CHARACTER, e.g., input "006600" instead of 006600 The tickers have to be FULL AND EXACT, e.g., for Shanghai exchange and Shenzhen exchange, the input must have 6 digits, and for HK exchange, it must have 5 digits. the '0' in the beginning cannot be left out.

indexData the index information, before running getIndex, indexData needs to be loaded using getIndexData

Details


Value

A data table with companies and which index they are included

References


Examples

## Not run:
indexData <- getIndexData()
getIndex(600601, indexData)

## End(Not run)
getIndexConstnt	get a company’s market cap, data comes from NetEase

Description
get a company’s market cap, data comes from NetEase

Usage
getIndexConstnt(indexPool)

Arguments
indexPool		a pool of different index, special format for gfer

Value
A data table with companies total capitalization and market capitalization

getIndexData	get index information Currently include CSI 100, SSE 50, CSI 300, SSE Central SOEs 50, HSI, HSCEI

Description
get index information Currently include CSI 100, SSE 50, CSI 300, SSE Central SOEs 50, HSI, HSCEI

Usage
ggetIndexData()

Value
a data table containing index information
**getNBS**  
---

**Description**  
get National Bureau of Statistics data

**Usage**  
getNBS(indicator, start, end)

**Arguments**  
- indicator: of which data is fetched, indicator includes 'GDP', 'water resources', 'water use' and 'wastewater', etc.
- start: starting year of data wanted
- end: end year of data wanted, make sure your input end year exists in the NBS website

**Value**  
no return

**References**  
https://CRAN.R-project.org/package=rstatscn

**getPPPList**  
---

**Description**  
get PPP list from an official website

**Usage**  
getPPPList(startPage = 1, endPage, proxy = FALSE)

**Arguments**  
- startPage: on Which page you want to start, default is 1
- endPage: On which page you want to stop scrapping
- proxy: whether proxy will be used, default is FALSE
getPPPList_unit

Details

Get PPP list from the Ministry of Finance of China (http://www.cpppc.org:8082/efmisweb/ppp/projectLibrary/toPPPList.do?projName=) to view the listed projects in the PPP library.

Value

A table of PPP projects collected from your input page

References

www.cpppc.org

Examples

```r
## Not run:
#scrape the first two pages
getPPPList(1,3)
## End(Not run)
```

getPPPList_unit
get PPP list from a single page

Description

get PPP list from a single page

Usage

`getPPPList_unit(page, proxy = NULL)`

Arguments

- `page`: The page number
- `proxy`: if you want to use a proxy to avoid blocking, you can input a proxy, otherwise leave it blank.

Value

A table of PPP projects collected from your input page
**getProxy**  
*Get proxy pool from free proxy provider*

**Description**  
Get proxy pool from free proxy provider

**Usage**  
getProxy()

**Details**  
Extract proxies from http://www.free-proxy-list.net/, in case of the risk of being blocked by the scrapped website

**Value**  
The sum of x and y.

**References**  
www.free-proxy-list.net

---

**getStockList**  
*Get information from Shanghai Exchange and Shenzhen Exchange. This will only get stock information in Shanghai Exchange and Shenzhen Exchange. Including stocker ticker, stock name and company full name. Data comes from China Merchants Bank*

**Description**  
Get information from Shanghai Exchange and Shenzhen Exchange. This will only get stock information in Shanghai Exchange and Shenzhen Exchange. Including stocker ticker, stock name and company full name. Data comes from China Merchants Bank

**Usage**  
getStockList()

**References**  
http://info.cmbchina.com/Stock/Single/
**getTickers**

*get ticker by input a company’s full name or a list of companies’ full name*

**Description**

It can also be a way to test if a company is listed. **NOTE:** If a company is listed in multiple exchange, then it needs double check, the program only chooses ticker from random exchange.

**Usage**

`getTickers(corpNames)`

**Arguments**

- `corpNames` Full name of a company, should be full name

**Details**

Data comes from www.cninfo.com.cn/

**Value**

A data table with companies stock name and stock ticker

**References**

www.cninfo.com.cn

---

**getTickers_unit**

*get ticker by input a company’s full name*

**Description**

It can also be a way to test if a company is listed

**Usage**

`getTickers_unit(corpName)`

**Arguments**

- `corpName` Full name of a company

**Details**

Data comes from www.cninfo.com.cn/
Value

A data table with companies stock name and stock ticker

Description

Get NBS data from google sheet by shared link. Default link is provided by gfer, you can also create your own google sheet of GDP. NOTE: The 'link sharing on' of the sheet must be ticked in order to read

Usage

getWaternomicsData_goog()

Description

Get NBS data from NBS website.

Usage

getWaternomicsData_NBS(start, end)

Arguments

<table>
<thead>
<tr>
<th>Argument</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>start</td>
<td>starting year of data wanted</td>
</tr>
<tr>
<td>end</td>
<td>end year of data wanted, make sure your input end year exists in the NBS website</td>
</tr>
</tbody>
</table>
Description

getPPP list from a single page

Usage

getWaterQ_MEP_all(year, week, station1, station2, proxy = FALSE)

Arguments

- **year**: In which year you would like to scrape
- **week**: In which week you would like to scrape, can be an array, like 3:5
- **station1**: the start station index on the page
- **station2**: the end station index on the page
- **proxy**: Whether to use proxy, default is FALSE

Details

Get monitoring data of different stations from Ministry of Environmental Protection of China (http://datacenter.mep.gov.cn/report/getCountGraph.do?type=runQianWater). Using this function you will get data of all the stations. Since the number of stations vary with time, using this function, you have to make sure that within the period you are scrapping, the number of stations keep consistent.

References

http://datacenter.mee.gov.cn/report/getCountGraph.do?type=runQianWater

Examples

```r
## Not run:
# get data from 1st station to 5th station of the 3rd week of 2016
a <- getWaterQ_MEP_all(2016, 3, 1, 5)

## End(Not run)
```
Description
get PPP list from a single page

Usage
getWaterQ_MEP_all_unit(year, week, station1, station2, proxy = NULL)

Arguments
- **year**: In which year you would like to scrape
- **week**: In which week you would like to scrape
- **station1**: the start station index on the page
- **station2**: the end station index on the page
- **proxy**: if you want to use a proxy to avoid blocking, you can input a proxy, otherwise leave it blank.

Value
A table of PPP projects collected from your input page

References
http://datacenter.mee.gov.cn/report/getCountGraph.do?type=runQianWater

Description
Check if a company is listed in Chinese stock market

Usage
is.listed(corpList, stockList)

Arguments
- **corpList**: company list you want to check if listed, should be a dataframe
- **stockList**: Result from getStockList
References

http://info.cmbchina.com/Stock/Single/

---

milSec  

**private function for sec**

---

**Description**

private function for sec

**Usage**

milSec()

**Value**

milsec

---

**plotChord**  

**plotScatterPie**

---

**Description**

if 'Summation of cell padding on y-direction are larger than the height of the cells’ appears, just enlarge the xlim or ylim accordingly

**Usage**

plotChord(
    data,
    t = FALSE,
    ifsep = TRUE,
    trans = 0.3,
    highlight = NULL,
    xlim = c(-1, 1),
    ylim = c(-1, 1)
)
Arguments

data  a dataframe showing different management intersections. See the data frame in the example

t  is transpose the dataframe, by default, lines flow from row to column, if t == TRUE, lines will flow from columns to rows. Once transposed,

ifsep  if separate row and col categories in the chart, default is TRUE

trans  transparency of the chart’s lines, default is 0.3

highlight  a string or string array of highlighted items, MUST be selected from first column (which represents names) or colnames. if highlight has more than 2 items, they should belong to same category, either colnames, or names. One name and one column name is not allowed.

xlim  x limit of the chart, default is c(-1, 1)

ylim  y limte of the chart, default is c(-1, 1)

Details

plot scatter pie chart for multidimension analysis, such as waternomics. This plot can provide information about water use/wastewater of each provinces and GDP mix of each provinces, see examples.

Examples

```r
## Not run:
plotChord(cm)
plotChord(cm, t = T)
plotChord(cm, highlight = 'MEP')
plotChord(cm, highlight = 'Investment')
## End(Not run)
```

Description

plot scatter pie chart for multidimension analysis, such as waternomics. This plot can provide information about water use/wastewater of each provinces and GDP mix of each provinces, see examples.
plotScatterPie

Usage

plotScatterPie(
  data,
  pieRange,
  pieColor = NULL,
  xmeanLine = TRUE,
  ymeanLine = TRUE,
  label_on = TRUE,
  output = FALSE
)

Arguments

data: a dataframe with colnames x, y, r, label, these four names must be in colnames.
pieRange: define which column to which column to be presented by pie chart, see examples
pieColor: color for different colors in pie chart
xmeanLine: if plot x mean line
ymeanLine: if plot y mean line
label_on: Whether to show label
output: if you want an ggplot object as output, default is FALSE

Examples

GDPColor_CWR <- c("#6B8033", "#020303", "#0D77B9")

data(GDPmix)

# in colnames(GDPmix), there must be x, y, r, label.
# but right now, GDPmix has x, y, r, but lacks a label column, let's assign label to province column
colnames(GDPmix)[1] <- 'label'

## Not run:
plotScatterPie(GDPmix, pieRange = 4:6, pieColor = GDPColor_CWR)

## End(Not run)
**statscnDbs**

*the available dbs*

**Description**

the available dbs in the national db

**Usage**

```r
statscnDbs()
```

**Value**

a data frame with 2 columns, one is the dbcode, another is the db description

**References**


**Examples**

```r
## Not run:
statscnDbs()
```

```r
## End(Not run)
```

**statscnQueryData**

*query data in the statscn db*

**Description**

the main function for querying the statscn database, it will retrieve the data from specified db and organize the data in a data frame.

**Usage**

```r
statscnQueryData(
    zb = "A0201",
    dbcode = "hgnd",
    rowcode = "zb",
    colcode = "sj",
    moreWd = list(name = NA, value = NA)
)
```
statscnQueryLastN

Arguments

- **zb**: the zb/category code to be queried
- **dbcode**: the db code for querying
- **rowcode**: rowcode in the returned data frame
- **colcode**: colcode in the returned data frame
- **moreWd**: more constraint on the data where the name should be one of c("reg","sj") , which stand for region and sj/time. the valuecode for reg should be the region code queried by statscnRegions() the valuecode for sj should be like ‘2014’ for *nd , ‘2014C’ for *jd , ’201405’ for *yd. Be noted that , the moreWd name should be different with either rowcode or colcode

Value

the data frame you are querying

References


Examples

```r
## Not run:
df <- statscnQueryData(’A0201’, dbcode = ’hgn’)
df <- statscnQueryData(’A0201’, dbcode = ’fjgn’, rowcode = ’zb’, colcode = ’sj’,
                   moreWd = list(name = ’reg’, value = ’110000’))
## End(Not run)
```

Description

fetch the lastN data for the latest query, only affect the number of rows in the returned data. This function can not be used alone , statscnQueryData() has to be called before this function

Usage

`statscnQueryLastN(n)`

Arguments

- **n**: the number of rows to be fetched
**Value**

the last n rows data in the latest query

**References**


**Examples**

```r
## Not run:
df=statscnQueryData('A0201',dbcode='hgnd')
df2=statscnQueryLastN(20)

## End(Not run)
```

---

**statscnQueryZb**

*the data categories*

**Description**

the sub data categories for the zbid category, dbcode need to be specified, where the dbcode can be fetched by function statscnDbs(). In the returned data frame, the column 'isParent' shows if each sub category is leap category or not

**Usage**

```r
statscnQueryZb(zbid = "zb", dbcode = "hgnd")
```

**Arguments**

- `zbid` the father zb/category id, the root id is 'zb'
- `dbcode` which db will be queried

**Value**

the data frame with the sub zbs/categories, if the given zbid is not a Parent zb/category, null list is returned

**References**

Examples

```r
## Not run:
statscnQueryZb()
statscnQueryZb("A01", dbcode="hgn0")

## End(Not run)
```

---

**statscnRegions**

*the regions in db*

**Description**

the available regions in the specified db, it is used for query the province, city and country code generally

**Usage**

```r
statscnRegions(dbcode = "fsnd")
```

**Arguments**

- `dbcode` the dbcode should be some province db(fs*), city db(cs*) or internal db(gj*)

**Value**

the data frame with all the available region codes and names in the db

**References**


**Examples**

```r
## Not run:
statscnRegions('fsnd')
statscnRegions('csnd')
statscnRegions('gjnd')

## End(Not run)
```
Description

set the rowName prefix in the dataframe

Usage

statscnRowNamePrefix(p = "nrow")

Arguments

p , how to set the rowname prefix. it is 'nrow' by default, and it is the only supported value currently to unset the row name prefix, call this function with p=NULL

Details

in case you encounter the following error: Error in 'row.names<-.data.frame'("*tmp*", value = value) : duplicate 'row.names' are not allowed you need to call this function

Value

no return

References

Index

* datasets
  cm, 3
  GDPmix, 4

checkHttpStatus, 3
cm, 3
dataJson2df, 4

GDPmix, 4
genDfwds, 5
getCSRRating, 5
getCSRRating_unit, 6
getENNames, 7
getENNNames_unit, 8
getExchange, 8
getHisMktCap, 9
getHisMktCap_unit, 10
getIndex, 11
getIndexConstnt, 12
getIndexData, 11, 12
getNBS, 13
getPPPList, 13
getPPPList_unit, 14
getProxy, 15
getStockList, 15, 19
getTickers, 16
getTickers_unit, 16
getWaternomicsData_goog, 17
getWaternomicsData_NBS, 17
getWaterQ_MEP_all, 18
getWaterQ_MEP_all_unit, 19

is.listed, 19

milSec, 20

plotChord, 20
plotScatterPie, 21

statscnObs, 23

statscnQueryData, 23
statscnQueryLastN, 24
statscnQueryZb, 25
statscnRegions, 26
statscnRowNamePrefix, 27