

# Package: rstatscn (via r-universe)

September 11, 2024

**Type** Package

**Title** R Interface for China National Data

**Version** 1.1.3

**Date** 2019-04-25

**Author** Xuehui YANG

**Maintainer** Xuehui YANG <jianghang@bagualu.net>

**Description** R interface for china national data  
<<http://data.stats.gov.cn/>>, some convenient functions for  
accessing the national data are provided.

**Depends** R (>= 3.2.2)

**Imports** httr(>= 1.0.0), jsonlite(>= 0.9.19)

**URL** <http://www.bagualu.net/>

**BugReports** <http://www.bagualu.net/wordpress/rstatscn-the-r-interface-for-china-national-data>

**License** Apache License 2.0

**RoxygenNote** 6.1.1

**Repository** <https://jiang-hang.r-universe.dev>

**RemoteUrl** <https://github.com/jiang-hang/rstatscn>

**RemoteRef** HEAD

**RemoteSha** c3f7f6dd6e3314a700b25bdc59cd35255eac9428

## Contents

checkHttpStatus . . . . .	2
dataJson2df . . . . .	2
genDfws . . . . .	3
milSec . . . . .	3
statscnDbs . . . . .	4
statscnQueryData . . . . .	4
statscnQueryLastN . . . . .	5

statscnQueryZb . . . . .	6
statscnRegions . . . . .	6
statscnRowNamePrefix . . . . .	7

<b>Index</b>	<b>8</b>
--------------	----------

---

checkHttpStatus	<i>private function for check the http status</i>
-----------------	---------------------------------------------------

---

### 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

---

dataJson2df	<i>private function to convert the returned json data to a dataframe</i>
-------------	--------------------------------------------------------------------------

---

### 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

---

genDfwds	<i>private function for constructing the query parameter for dfwds</i>
----------	------------------------------------------------------------------------

---

**Description**

private function for constructing the query parameter for dfwds

**Usage**

```
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 valudecode can be gotten by statscnQueryZb() function sj: the valudecode can be "2014" for nd db, "2014C" for jd db. reg: the valudecode is the region code fetched by statscnRegions(dbcode) function

**Value**

return the queyr string for the http request

---

milSec	<i>private function for sec</i>
--------	---------------------------------

---

**Description**

private function for sec

**Usage**

```
milSec()
```

**Value**

milsec

---

statscnDbs	<i>the available dbs</i>
------------	--------------------------

---

**Description**

the available dbs in the national db

**Usage**

```
statscnDbs()
```

**Value**

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

**Examples**

```
statscnDbs()
```

---

statscnQueryData	<i>query data in the statscn db</i>
------------------	-------------------------------------

---

**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**

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

**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 quering

**Examples**

```
## Not run:
df=statscnQueryData('A0201',dbcode='hgnd')
df=statscnQueryData('A0201',dbcode='fsnd',rowcode='zb',colcode='sj',
                    moreWd=list(name='reg',value='110000'))

## End(Not run)
```

---

statscnQueryLastN	<i>fetch the lastN data</i>
-------------------	-----------------------------

---

**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

**Examples**

```
## 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

```
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

### Examples

```
## Not run:
statscnQueryZb()
statscnQueryZb('A01', dbcode="hgnd")

## 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

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

### Arguments

dbcode                the dbcode should be some province db(fs\*) , city db(cs\*) or international db(gj\*)

**Value**

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

**Examples**

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

## End(Not run)
```

---

statscnRowNamePrefix    *statscnRowNamePrefix*

---

**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

# Index

`checkHttpStatus`, 2

`dataJson2df`, 2

`genDfws`, 3

`milSec`, 3

`statscnDbs`, 4

`statscnQueryData`, 4

`statscnQueryLastN`, 5

`statscnQueryZb`, 6

`statscnRegions`, 6

`statscnRowNamePrefix`, 7