

Package ‘TSsql’

May 10, 2017

Version 2017.4-1

Title Generic SQL Helper Functions for 'TSdbi' SQL Plugins

Description Standard SQL query functions used by SQL plugins packages for the 'TSdbi' interface to time series databases. It will mainly be used by other packages rather than directly by end users. The one exception is the function 'TSquery' which can be used to construct a time series from a database containing observations over time (e.g. balance statements for multiple years), but where the database is not specifically designed to store time series (as with other 'TSdbi' SQL plugin packages). Comprehensive examples of all the 'TS*' packages is provided in the vignette Guide.pdf with the 'TSdata' package.

Depends R (>= 2.8.0)

Imports methods, DBI, tframe (>= 2015.1-1), TSdbi (>= 2015.1-1), tframePlus, zoo, stats

Suggests tseries, tis, tfplot, RMySQL, RSQLite

BuildVignettes true

License GPL-2

Copyright 2007-2011 Bank of Canada. 2012-2017 Paul Gilbert.

Author Paul Gilbert <pgilbert.ttv9z@ncf.ca>

Maintainer Paul Gilbert <pgilbert.ttv9z@ncf.ca>

URL <http://tsdbi.r-forge.r-project.org/>

NeedsCompilation no

Repository CRAN

Date/Publication 2017-05-10 06:48:30 UTC

R topics documented:

createTSdbTables	2
TSquery	3
Index	4

createTSdbTables *Create and Remove Time Series Database Tables*

Description

Create or remove all the TS database tables.

Usage

```
createTSdbTables(con, index=FALSE)
removeTSdbTables(con, yesIknowWhatIamDoing=FALSE, ToLower=FALSE)
```

Arguments

con	A database connection object (NOT a TSconnect object).
yesIknowWhatIamDoing	logical to help prevent accidents.
index	logical to indicate if table date indexes should be added.
ToLower	logical indicating if table names should be converted to lower case.

Details

The function `createTSdbTables()` creates tables on the database to store time series data. Typically this is done only on initial setup, and might be done by a database administrator and not by an end user. (It is used for examples and testing but will rarely be used otherwise.) Adding date indexes (`index=TRUE`) will typically speed searching but may dramatically slow down writing to the database (but for small databases it will not make much difference either way).

The function `removeTSdbTables` removes all the TS database tables. **WARNING:** use these function only if you know what you are doing. They destroy the whole database and all stored series. The function `removeTSdbTables` needs to work around the problem that different db engines treat capitalized table names differently. For example, MySQL uses table name `Meta` while Posgresql converts to `meta`. The argument `ToLower` indicates if table names should be converted to lower case.

The `con` needs to be specified. A default `con` is not used on purpose to help avoid accidental use. The argument `yesIknowWhatIamDoing` defaults to `FALSE` and must be set to `TRUE` or the function will return an error. Beware that dropping tables will destroy the integrity of the of the database, and would usually only be done when you are intializing a TS database. Database permission may also be set to prevent users from dropping tables. If that is the case, then this function will return a failure error.

Value

logical to indicate success.

See Also

[dbConnect](#), [TSdates](#), [TSget](#), [TSput](#)

Description

Construct a time series from a database

Usage

```
TSquery(select, dateField, table, where=NULL, frequency="monthly",
        na.as=0, names=NULL, con=options())$connection)
```

Arguments

select	character string used to construct query.
dateField	character string used to construct query.
table	character string used to construct query.
where	character string used to construct query.
frequency	character string used to specify frequency of the result (only "daily", "monthly" or "annual" are currently supported).
na.as	value to be used to replace NAs in the returned series.
names	optional vector of character strings to be used for the returned series. If not NULL it must have length equal to the number of series returned.
con	A database connection.

Details

This functions is unlike other functins in this package (and may eventually be moved to a separate package). It constructs a time series from a database (using a connection) and a query generated with the function arguments. An example query might be something like `z<- TSquery(select="SUM(amount)", dateField=`

Value

A time series or time series matrix.

See Also

[TSget](#),

Index

*Topic **ts**

createTSdbTables, [2](#)

TSquery, [3](#)

createTSdbTables, [2](#)

dbConnect, [2](#)

removeTSdbTables (createTSdbTables), [2](#)

TSdates, [2](#)

TSget, [2](#), [3](#)

TSput, [2](#)

TSquery, [3](#)