

Package ‘RSQLite.extfuns’

July 2, 2014

Type Package

Title Math and String Extension Functions for RSQLite

Version 0.0.1

Date 2010-05-03

Author Seth Falcon <seth@userprimary.net>

Maintainer Seth Falcon <seth@userprimary.net>

Depends RSQLite (>= 0.9-1)

LinkingTo RSQLite

Description This package uses SQLite's loadable extension feature to provide a number of additional SQL functions and aggregates. The package is a wrapper of extension functions written by Liam Healy and made available through the SQLite website (<http://www.sqlite.org/contrib>). Math: acos, asin, atan, atn2,atan2, acosh, asinh, atanh, difference, degrees, radians, cos,sin, tan, cot, cosh, sinh, tanh, coth, exp, log, log10, power,sign, sqrt, square, ceil, floor, pi. String: replicate,charindex, leftstr, rightstr, ltrim, rtrim, trim, replace,reverse, proper, padl, padr, padc, strfilter. Aggregate: stdev,variance, mode, median, lower_quartile, upper_quartile.

URL <http://www.sqlite.org/contrib/>

License Artistic-2.0

LazyLoad yes

Repository CRAN

Date/Publication 2010-05-30 11:37:46

NeedsCompilation yes

R topics documented:

RSQLite.extfuns-package	2
init_extensions	3

SQLite.extfun package

Math and String Extension Functions for SQLite

Description

This package uses SQLite's loadable extension feature to provide a number of additional SQL functions and aggregates. The package is a wrapper of extension functions written by Liam Healy and made available through the SQLite website (<http://www.sqlite.org/contrib>).

Available extension functions

You can enable all of the following extension functions by calling the `init_extensions` function on an SQLite database connection object.

Math functions:

<code>acos</code>	<code>radians</code>	<code>log</code>
<code>asin</code>	<code>cos</code>	<code>log10</code>
<code>atan</code>	<code>sin</code>	<code>power</code>
<code>atan2</code>	<code>tan</code>	<code>sign</code>
<code>atan2</code>	<code>cot</code>	<code>sqrt</code>
<code>acosh</code>	<code>cosh</code>	<code>square</code>
<code>asinh</code>	<code>sinh</code>	<code>ceil</code>
<code>atanh</code>	<code>tanh</code>	<code>floor</code>
<code>difference</code>	<code>coth</code>	<code>pi</code>
<code>degrees</code>	<code>exp</code>	

String functions:

<code>replicate</code>	<code>replace</code>
<code>charindex</code>	<code>reverse</code>
<code>leftstr</code>	<code>proper</code>
<code>rightstr</code>	<code>padl</code>
<code>ltrim</code>	<code>padr</code>
<code>rtrim</code>	<code>padc</code>
<code>trim</code>	<code>strfilter</code>

Aggregate functions:

<code>stdev</code>
<code>variance</code>
<code>mode</code>
<code>median</code>

lower_quartile
upper_quartile

Author(s)

Seth Falcon <seth@userprimary.net> Maintainer: Seth Falcon <seth@userprimary.net>

References

See <http://www.sqlite.org/contrib/> for the original source of the SQLite extension function source code written by Liam Healy.

Examples

```
## basic usage looks like this
db <- dbConnect(SQLite(), dbname = ":memory:")
init_extensions(db)

## now you can use any of the extension functions listed above in
## SQL queries.
dbDisconnect(db)
```

init_extensions	<i>Initialize SQLite extension functions on a SQLite DB connection</i>
-----------------	--

Description

This function loads extension functions to make them available to an RSQLite database connection. The database connection needs to have been created by passing `loadable.extensions = TRUE` to `dbConnect` (this is the default as of RSQLite 0.9-1).

Usage

```
init_extensions(db)
```

Arguments

db An RSQLite database connection as returned by `dbConnect`.

Value

TRUE if the extensions were successfully loaded.

Author(s)

Seth Falcon

References

For details on loadable extensions in SQLite, see <http://www.sqlite.org/cvstrac/wiki?p=LoadableExtensions>.

The extensions functions included in this package were written by Liam Healy and made available in the contrib section of the SQLite website (<http://www.sqlite.org/contrib>).

See Also

The package manual page has a complete list of the extension functions: [RSQLite.extfuns](#)

Examples

```
db <- dbConnect(SQLite(), dbname = ":memory:",
                loadable.extensions = TRUE)

init_extensions(db)

data(USArrests)
dbWriteTable(db, "t1", USArrests)
dbGetQuery(db, "select variance(Murder) from t1")[[1]]
```

Index

*Topic **package**

 RSQLite.extfun-package, [2](#)

init_extensions, [3](#)

RSQLite.extfun, [4](#)

RSQLite.extfun

 (RSQLite.extfun-package), [2](#)

RSQLite.extfun-package, [2](#)