Package ‘RKinetica’

September 21, 2020

Type Package

Title Kinetica DBI datasource

Version 7.1.0.1

Author Nataliya Tairbekov

Maintainer Nataliya Tairbekov <ntairbekov@kinetica.com>

Description DBI implementation for Kinetica DB (http://kinetica.com).

License MIT + file LICENSE

Encoding UTF-8

LazyData true

Imports httr,
RJSONIO,
purrr,
dplyr,
stats,
DBI (>= 0.3.0),
bit64,
hms (>= 0.5.0),
methods

Suggests DBItest, testthat

RoxygenNote 6.1.1

Collate 'RKinetica-package.R'
'kineticaObject.R'
'kineticaId.R'
'kineticaDriver.R'
'kineticaConnection.R'
'kineticaResult.R'
'kineticaSQL.R'
'connect.R'
'zzz.R'

R topics documented:

dbAppendTable . . . 3
dbBind,KineticaResult-method . . . 3
dbCanConnect,KineticaDriver-method . . . 4
dbClearResult,KineticaResult-method . . . 5
dbColumnInfo . . . 6
dbConnect,KineticaDriver-method . . . 6
dbCreateTable . . . 8
dbDataType,KineticaConnection-method . . . 9
dbDataType,KineticaDriver-method . . . 9
dbDisconnect,KineticaConnection-method . . . 10
dbExecute,KineticaConnection,character-method . . . 11
dbExistsTable . . . 12
dbFetch . . . 13
dbGetInfo . . . 14
dbGetQuery,KineticaConnection,character-method . . . 16
dbGetRowCount,KineticaResult-method . . . 17
dbGetRowsAffected,KineticaResult-method . . . 17
dbGetStatement,KineticaResult-method . . . 18
dbHasCompleted,KineticaResult-method . . . 18
dbIsReadOnly . . . 19
dbIsValid . . . 19
dbListConnections,KineticaDriver-method . . . 21
dbListFields . . . 22
dbListObjects,KineticaConnection-method . . . 23
dbListResults,KineticaConnection-method . . . 23
dbListTables,KineticaConnection-method . . . 24
dbQuoteIdentifier . . . 25
dbQuoteLiteral,KineticaConnection-method . . . 25
dbQuoteString . . . 26
dbReadTable . . . 26
dbRemoveTable . . . 27
dbSendQuery,KineticaConnection,character-method . . . 28
dbSendStatement,KineticaConnection,character-method . . . 28
dbUnloadDriver,KineticaDriver-method . . . 29
dbUnquoteIdentifier . . . 30
dbWriteTable . . . 30
KineticaId-class . . . 31
KineticaObject-class . . . 32
RKinetica . . . 32
show,KineticaDriver-method . . . 33
show,SQL-method . . . 34
SQL . . . 34
sqlAppendTable . . . 35
sqlAppendTableTemplate . . . 36
sqlCreateTable,KineticaConnection-method . . . 36
sqlData . . . 37

summary 38
summary,KineticaObject-method 38
transactions 39

Index **41**

`dbAppendTable` *dbAppendTable()*

Description

Appends rows to an existing table

Usage

```
dbAppendTable(conn, name, value, ..., row.names = NULL)
```

Arguments

`conn`	an object of [KineticaConnection-class]
`name`	character string for table name or [KineticaId-class] object
`value`	A data frame. Factors will be converted to character vectors.
`...`	Other arguments omitted in generic signature
`row.names`	a flag with logical, character or NULL value

See Also

Other KineticaConnection methods: `dbCreateTable`, `dbDataType,KineticaConnection-method`, `dbDisconnect,KineticaConnection-method`, `dbExecute,KineticaConnection,character-method`, `dbExistsTable`, `dbGetInfo`, `dbGetQuery,KineticaConnection,character-method`, `dbIsValid`, `dbListFields`, `dbListObjects,KineticaConnection-method`, `dbListResults,KineticaConnection-method`, `dbListTables,KineticaConnection-method`, `dbReadTable`, `dbRemoveTable`, `dbSendQuery,KineticaConnection,char`
`dbSendStatement,KineticaConnection,character-method`, `dbWriteTable`, `show,KineticaDriver-method`, `sqlCreateTable,KineticaConnection-method`, `transactions`

`dbBind,KineticaResult-method` *dbBind()*

Description

Binds the prepared statement to parameter values provided

Usage

```
## S4 method for signature 'KineticaResult'
dbBind(res, params, ...)
```

Arguments

`res`	an object of [KineticaResult-class]
`params`	A list of bindings, named or unnamed.
`...`	Other parameters passed on to methods.

See Also

Other KineticaResult methods: `dbClearResult,KineticaResult-method`, `dbColumnInfo`, `dbFetch`, `dbGetInfo`, `dbGetRowCount,KineticaResult-method`, `dbGetRowsAffected,KineticaResult-method`, `dbGetStatement,KineticaResult-method`, `dbHasCompleted,KineticaResult-method`, `dbIsValid`, `show,KineticaDriver-method`

`dbCanConnect,KineticaDriver-method` *dbCanConnect()*

Description

Method allows to check if provided Kinetica Connection configuration could be used to connect to Kinetica DB instance successfully. It takes all the arguments of dbConnect signature, creates a connection and destroys it upon exit.

Usage

```
## S4 method for signature 'KineticaDriver'
dbCanConnect(drv, ...)
```

Arguments

`drv`	Kinetica driver object
`...`	Other arguments omitted in generic signature

See Also

Other KineticaDriver methods: `dbConnect,KineticaDriver-method`, `dbDataType,KineticaDriver-method`, `dbGetInfo`, `dbIsValid`, `dbListConnections,KineticaDriver-method`, `dbUnloadDriver,KineticaDriver-method`, `show,KineticaDriver-method`

Examples

```
## Not run:
# If you have a Kinetica DB instance running in local docker image,
# the minimum set of parameters to connect is url
dbCanConnect(Kinetica(), url = "http://localhost:9191")
# TRUE
# You can also use host and port combination
dbCanConnect(Kinetica(), host = "127.0.0.1", port = 9191)
# TRUE
# Remote instances would also require username and password.
dbCanConnect(Kinetica(), host = "127.0.0.1", port = 9191,
    username = "username", password = "Pa$$w0rd")
# TRUE
# You can obscure username and password values by calling RStudio secure dialogue box:
dbCanConnect(Kinetica(), host = "127.0.0.1", port = 9191,
             username = rstudioapi::askForPassword("Kinetica DB user"),
             password = rstudioapi::askForPassword("Kinetica DB password"))
# TRUE

## End(Not run)
```

`dbClearResult,KineticaResult-method`

dbClearResult()

Description

Clears result, if exists, releasing memory in Connection results environment

Usage

```
## S4 method for signature 'KineticaResult'
dbClearResult(res, ...)
```

Arguments

`res`	[KineticaResult-class]
`...`	Other parameters passed on to methods.

See Also

Other KineticaResult methods: `dbBind,KineticaResult-method`, `dbColumnInfo`, `dbFetch`, `dbGetInfo`, `dbGetRowCount,KineticaResult-method`, `dbGetRowsAffected,KineticaResult-method`, `dbGetStatement,KineticaR`
`dbHasCompleted,KineticaResult-method`, `dbIsValid`, `show,KineticaDriver-method`

dbColumnInfo *dbColumnInfo()*

Description

Provides data.frame as a collection of column names and data types from the result set

Provides data.frame as a collection of column names and data types from the result set

Usage

```
dbColumnInfo(res, ...)

## S4 method for signature 'KineticaResult'
dbColumnInfo(res, ...)
```

Arguments

`res`	An object inheriting from [KineticaResult-class]
`...`	Other parameters passed on to methods.
`res`	An object inheriting from [KineticaResult-class]
`...`	Other parameters passed on to methods.

See Also

Other KineticaResult methods: `dbBind,KineticaResult-method`, `dbClearResult,KineticaResult-method`, `dbFetch`, `dbGetInfo`, `dbGetRowCount,KineticaResult-method`, `dbGetRowsAffected,KineticaResult-method`, `dbGetStatement,KineticaResult-method`, `dbHasCompleted,KineticaResult-method`, `dbIsValid`, `show,KineticaDriver-method`

Other KineticaResult methods: `dbBind,KineticaResult-method`, `dbClearResult,KineticaResult-method`, `dbFetch`, `dbGetInfo`, `dbGetRowCount,KineticaResult-method`, `dbGetRowsAffected,KineticaResult-method`, `dbGetStatement,KineticaResult-method`, `dbHasCompleted,KineticaResult-method`, `dbIsValid`, `show,KineticaDriver-method`

dbConnect,KineticaDriver-method *dbConnect()*

Description

Connect to a Kinetica DB going through the appropriate authentication procedure. You can have multiple connections open and connections can have multiple results attached. This function may be invoked repeatedly assigning its output to different objects. Authentication can be provided by using username/password variables. row_limit param would limit the number of rows in query results, it's an integer value allowing up to 2,147,483,647 rows. Value literal must be followed by L, as in row_limit = 1000000L, also see example below. Use [dbCanConnect()] to check if a connection can be established.

Usage

```
## S4 method for signature 'KineticaDriver'
dbConnect(drv, host = NULL, port = NULL,
  url = NULL, username = NULL, password = NULL, timeout = NULL,
  row_limit = NULL, ha_ring = NULL, ...)
```

Arguments

`drv`	Kinetica driver object
`host`	character string for Kinetica DB host address
`port`	integer value for Kinetica DB port
`url`	character string for Kinetica DB url (protocol + host + port)
`username`	character string for Kinetica DB username
`password`	character string for Kinetica DB password
`timeout`	integer value for Kinetica DB connection timeout
`...`	Other arguments omitted in generic signature

See Also

Other KineticaDriver methods: `dbCanConnect,KineticaDriver-method`, `dbDataType,KineticaDriver-method`, `dbGetInfo`, `dbIsValid`, `dbListConnections,KineticaDriver-method`, `dbUnloadDriver,KineticaDriver-method`, `show,KineticaDriver-method`

Examples

```
## Not run:
# If you jave a Kinetica DB instance running in local docker image,
# the minimum set of parameters to connect is url
dbConnect(Kinetica(), url = "http://localhost:9191")
# You can also use host and port combination
dbConnect(Kinetica(), host = "127.0.0.1", port = 9191)
# Remote instances would also require username and password.
dbConnect(Kinetica(), host = "127.0.0.1", port = 9191,
  username = "username", password = "Pa$$w0rd", row_limit = 1000000L)
con <- dbConnect(Kinetica(), url = "http://localhost:9191")
show(con)
# KineticaConnection
# url: http://localhost:9191
dbDisconnect(con)

## End(Not run)
```

`dbCreateTable` *dbCreateTable()*

Description

Creates a new table

Usage

```
dbCreateTable(conn, name, fields, ..., row.names = NULL,
  temporary = FALSE)
```

Arguments

`conn`	an object of [KineticaConnection-class]
`name`	character string for table name or [KineticaId-class] object
`fields`	either a named character vector or a data frame
`...`	Other parameters passed on to methods.
`row.names`	Must be ‘NULL‘.
`temporary`	logical flag value whether table should be temporary

See Also

Other KineticaConnection methods: `dbAppendTable`, `dbDataType,KineticaConnection-method`, `dbDisconnect,KineticaConnection-method`, `dbExecute,KineticaConnection,character-method`, `dbExistsTable`, `dbGetInfo`, `dbGetQuery,KineticaConnection,character-method`, `dbIsValid`, `dbListFields`, `dbListObjects,KineticaConnection-method`, `dbListResults,KineticaConnection-method`, `dbListTables,KineticaConnection-method`, `dbReadTable`, `dbRemoveTable`, `dbSendQuery,KineticaConnection,char`
`dbSendStatement,KineticaConnection,character-method`, `dbWriteTable`, `show,KineticaDriver-method`, `sqlCreateTable,KineticaConnection-method`, `transactions`

Examples

```
## Not run:
con <- dbConnect(Kinetica(), url = "http://localhost:9191")
dbCreateTable(con, "tableA", data.frame(a = 1))
dbExistsTable(con, "tableA")
# TRUE
dbDisconnect(con)

## End(Not run)
```

dbDataType,KineticaConnection-method

dbDataType()

Description

Redirect to [KineticaDriver-class] dbDataType method for data type mapping of provided object

Usage

```
## S4 method for signature 'KineticaConnection'
dbDataType(dbObj, obj, ...)
```

Arguments

dbObj	an object of [KineticaConnection-class]
obj	generic object
...	Other arguments omitted in generic signature

See Also

Other KineticaConnection methods: dbAppendTable, dbCreateTable, dbDisconnect,KineticaConnection-method, dbExecute,KineticaConnection,character-method, dbExistsTable, dbGetInfo, dbGetQuery,KineticaConnection, dbIsValid, dbListFields, dbListObjects,KineticaConnection-method, dbListResults,KineticaConnection-meth dbListTables,KineticaConnection-method, dbReadTable, dbRemoveTable, dbSendQuery,KineticaConnection,char dbSendStatement,KineticaConnection,character-method, dbWriteTable, show,KineticaDriver-method, sqlCreateTable,KineticaConnection-method, transactions

dbDataType,KineticaDriver-method

dbDataType()

Description

Takes an R object or literal and returns a matching SQLtype used when creating Kinetica DB table value to store it

Usage

```
## S4 method for signature 'KineticaDriver'
dbDataType(dbObj, obj, ...)
```

Arguments

dbObj	KineticaDriver object
obj	any data object or literal
...	Other arguments omitted in generic signature

See Also

Other KineticaDriver methods: dbCanConnect,KineticaDriver-method, dbConnect,KineticaDriver-method, dbGetInfo, dbIsValid, dbListConnections,KineticaDriver-method, dbUnloadDriver,KineticaDriver-method, show,KineticaDriver-method

Examples

```
## Not run:
dbDataType(Kinetica(), 1L)
 "INTEGER"
dbDataType(Kinetica(), 1)
 "DOUBLE"
dbDataType(Kinetica(), "Test string")
 "VARCHAR(256)"
dbDataType(Kinetica(), Sys.Date())
 "DATE"
dbDataType(Kinetica(), list())
 "BINARY"

## End(Not run)
```

dbDisconnect,KineticaConnection-method
dbDisconnect()

Description

Disconnects the current connection, releasing memory from all uncleared result objects

Usage

```
## S4 method for signature 'KineticaConnection'
dbDisconnect(conn, ...)
```

Arguments

conn	an object of [KineticaConnection-class]
...	Other parameters passed on to methods.

See Also

Other KineticaConnection methods: dbAppendTable, dbCreateTable, dbDataType,KineticaConnection-method,
dbExecute,KineticaConnection,character-method, dbExistsTable, dbGetInfo, dbGetQuery,KineticaConnection,
dbIsValid, dbListFields, dbListObjects,KineticaConnection-method, dbListResults,KineticaConnection-meth
dbListTables,KineticaConnection-method, dbReadTable, dbRemoveTable, dbSendQuery,KineticaConnection,char
dbSendStatement,KineticaConnection,character-method, dbWriteTable, show,KineticaDriver-method,
sqlCreateTable,KineticaConnection-method, transactions

Examples

```
## Not run:
con <- dbConnect(Kinetica(), url = "http://localhost:9191")
dbDisconnect(con)

## End(Not run)
```

`dbExecute,KineticaConnection,character-method`
dbExecute()

Description

Executes the statement, returning the number of affected rows or objects

Usage

```
## S4 method for signature 'KineticaConnection,character'
dbExecute(conn, statement, ...,
  params = NULL)
```

Arguments

`conn`	an object of [KineticaConnection-class]
`statement`	character
`...`	Other parameters passed on to methods.
`params`	a set of values

See Also

Other KineticaConnection methods: dbAppendTable, dbCreateTable, dbDataType,KineticaConnection-method,
dbDisconnect,KineticaConnection-method, dbExistsTable, dbGetInfo, dbGetQuery,KineticaConnection,charact
dbIsValid, dbListFields, dbListObjects,KineticaConnection-method, dbListResults,KineticaConnection-meth
dbListTables,KineticaConnection-method, dbReadTable, dbRemoveTable, dbSendQuery,KineticaConnection,char
dbSendStatement,KineticaConnection,character-method, dbWriteTable, show,KineticaDriver-method,
sqlCreateTable,KineticaConnection-method, transactions

Examples

```
## Not run:
con <- dbConnect(Kinetica(), url = "http://localhost:9191")
dbCreateTable(con, "tableA", data.frame(a = 1))
dbExistsTable(con, "tableA")
# TRUE
dbExecute(con, "DROP TABLE tableA")
# 1
dbDisconnect(con)

## End(Not run)
```

dbExistsTable *dbExistsTable()*

Description

Checks if the table exists. Provide either table name or collection name, not both Kinetica DB table names are unique, different collections don't allow tables with the same name.

Usage

```
dbExistsTable(conn, name, ...)
```

Arguments

`conn`	an object of [KineticaConnection-class]
`name`	character string for table or collection name or [KineticaId-class] object
`...`	Other parameters passed on to methods.

Value

logical

See Also

Other KineticaConnection methods: `dbAppendTable`, `dbCreateTable`, `dbDataType,KineticaConnection-method`, `dbDisconnect,KineticaConnection-method`, `dbExecute,KineticaConnection,character-method`, `dbGetInfo`, `dbGetQuery,KineticaConnection,character-method`, `dbIsValid`, `dbListFields`, `dbListObjects,KineticaConnection-method`, `dbListResults,KineticaConnection-method`, `dbListTables,KineticaConnection-method`, `dbReadTable`, `dbRemoveTable`, `dbSendQuery,KineticaConnection,char` `dbSendStatement,KineticaConnection,character-method`, `dbWriteTable`, `show,KineticaDriver-method`, `sqlCreateTable,KineticaConnection-method`, `transactions`

Examples

```
## Not run:
con <- dbConnect(Kinetica(), url = "http://localhost:9191")
dbCreateTable(con, "tableA", data.frame(a = 1))
dbExistsTable(con, "tableA")
# TRUE
dbExecute(con, "DROP TABLE tableA")
# 1
dbExistsTable(con, "tableA")
# FALSE
dbDisconnect(con)

## End(Not run)
```

---

dbFetch *dbFetch()*

Description

Fetches the provided number of rows from Result. If n is not provided, returns all available records

Fetches the provided number of rows from Result. If n is not provided, returns all available records

Usage

```
dbFetch(res, n = -1, ...)

## S4 method for signature 'KineticaResult'
dbFetch(res, n = -1, ...)
```

Arguments

`res` An object inheriting from [KineticaResult-class]

`n` maximum number of records to retrieve per fetch. Use ‘n = -1‘ or ‘n = Inf‘ to retrieve all pending records. Some implementations may recognize other special values.

`...` Other arguments passed on to methods.

`res` An object inheriting from [KineticaResult-class]

`n` maximum number of records to retrieve per fetch. Use ‘n = -1‘ or ‘n = Inf‘ to retrieve all pending records. Some implementations may recognize other special values.

`...` Other arguments passed on to methods.

See Also

Other KineticaResult methods: dbBind,KineticaResult-method, dbClearResult,KineticaResult-method, dbColumnInfo, dbGetInfo, dbGetRowCount,KineticaResult-method, dbGetRowsAffected,KineticaResult-method, dbGetStatement,KineticaResult-method, dbHasCompleted,KineticaResult-method, dbIsValid, show,KineticaDriver-method

Other KineticaResult methods: dbBind,KineticaResult-method, dbClearResult,KineticaResult-method, dbColumnInfo, dbGetInfo, dbGetRowCount,KineticaResult-method, dbGetRowsAffected,KineticaResult-method, dbGetStatement,KineticaResult-method, dbHasCompleted,KineticaResult-method, dbIsValid, show,KineticaDriver-method

---

dbGetInfo *dbGetInfo()*

Description

Generic dbGetInfo() implementation.

Retrieves brief KineticaDriver description: dbname, driver version, client version, max number of connections (NA stands for unlimited)

Basic properties of KineticaConnection object provides DB version, primary url, host, port and username of the current connection, flag for HA availability, HA ring of secondary urls and maximum row count that can be returned in one query

Provides basic info on KineticaResult object

Usage

```
dbGetInfo(dbObj, ...)

## S4 method for signature 'KineticaDriver'
dbGetInfo(dbObj, ...)

## S4 method for signature 'KineticaConnection'
dbGetInfo(dbObj, ...)

## S4 method for signature 'KineticaResult'
dbGetInfo(dbObj, ...)
```

Arguments

`dbObj`	An object inheriting from [KineticaObject-class]
`...`	Other arguments to methods.
`dbObj`	an object derived of [KineticaDriver-class] type
`...`	Other arguments omitted in generic signature
`dbObj`	object [KineticaConnection-class]
`...`	Other arguments omitted in generic signature
`dbObj`	[KineticaResult-class]
`...`	Other parameters passed on to methods.

Value

a named list

See Also

Other KineticaDriver methods: dbCanConnect,KineticaDriver-method, dbConnect,KineticaDriver-method, dbDataType,KineticaDriver-method, dbIsValid, dbListConnections,KineticaDriver-method, dbUnloadDriver,KineticaDriver-method, show,KineticaDriver-method

Other KineticaConnection methods: dbAppendTable, dbCreateTable, dbDataType,KineticaConnection-method, dbDisconnect,KineticaConnection-method, dbExecute,KineticaConnection,character-method, dbExistsTable, dbGetQuery,KineticaConnection,character-method, dbIsValid, dbListFields, dbListObjects,KineticaConnection-method, dbListResults,KineticaConnection-method, dbListTables,KineticaConnection-method, dbReadTable, dbRemoveTable, dbSendQuery,KineticaConnection,char dbSendStatement,KineticaConnection,character-method, dbWriteTable, show,KineticaDriver-method, sqlCreateTable,KineticaConnection-method, transactions

Other KineticaResult methods: dbBind,KineticaResult-method, dbClearResult,KineticaResult-method, dbColumnInfo, dbFetch, dbGetRowCount,KineticaResult-method, dbGetRowsAffected,KineticaResult-method, dbGetStatement,KineticaResult-method, dbHasCompleted,KineticaResult-method, dbIsValid, show,KineticaDriver-method

Examples

```
drv <- Kinetica()
dbGetInfo(drv)
# name
#  "KineticaDriver"
# dbname
#  "Kinetica"
# driver.version
#  "7.1.0.0"
# client.version
#  "7.1.0.0"
# max.connections
#  NA

## Not run:
con <- dbConnect(d, url = "http://localhost:9191")
# db.version
# "7.1.0.0"
# dbname
# "Kinetica"
# url
# "http://localhost:9191"
# host
# "localhost"
# port
# 9191
# username
# ""
```

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

dbGetQuery,KineticaConnection,character-method

dbGetQuery()

Description

Executes the Query and returns the resulting data.frame of n rows (or less) When n is not provided, returns all rows available

Usage

```
## S4 method for signature 'KineticaConnection,character'
dbGetQuery(conn, statement,
  n = -1, ...)
```

Arguments

`conn`	A [KineticaConnection-class] object
`statement`	a character string containing SQL.
`n`	Number of rows to fetch, default -1
`...`	Other parameters passed on to methods.

See Also

Other KineticaConnection methods: `dbAppendTable`, `dbCreateTable`, `dbDataType,KineticaConnection-method`, `dbDisconnect,KineticaConnection-method`, `dbExecute,KineticaConnection,character-method`, `dbExistsTable`, `dbGetInfo`, `dbIsValid`, `dbListFields`, `dbListObjects,KineticaConnection-method`, `dbListResults,KineticaConnection-method`, `dbListTables,KineticaConnection-method`, `dbReadTable`, `dbRemoveTable`, `dbSendQuery,KineticaConnection,character-method`, `dbSendStatement,KineticaConnection,cha`
`dbWriteTable`, `show,KineticaDriver-method`, `sqlCreateTable,KineticaConnection-method`, `transactions`

Examples

```
## Not run:
con <- dbConnect(Kinetica(), url = "http://localhost:9191")
dbWriteTable(con, "tableA", data.frame(a = 1:40))
dbExistsTable(con, "tableA")
# TRUE
ds <- dbGetQuery(con, "tableA", 3)
show(ds)
    a
1   1
2   2
3   3
```

```
dbRemoveTable(con, "tableA")
dbDisconnect(con)

## End(Not run)
```

dbGetRowCount,KineticaResult-method

dbGetRowCount()

Description

Returns the number of rows in the returned data set

Usage

```
## S4 method for signature 'KineticaResult'
dbGetRowCount(res, ...)
```

Arguments

res	KineticaResult-class
...	Other parameters passed on to methods.

See Also

Other KineticaResult methods: dbBind,KineticaResult-method, dbClearResult,KineticaResult-method, dbColumnInfo, dbFetch, dbGetInfo, dbGetRowsAffected,KineticaResult-method, dbGetStatement,KineticaResult dbHasCompleted,KineticaResult-method, dbIsValid, show,KineticaDriver-method

dbGetRowsAffected,KineticaResult-method

dbGetRowsAffected()

Description

Returns the number of rows affected by the executed SQL statement

Usage

```
## S4 method for signature 'KineticaResult'
dbGetRowsAffected(res, ...)
```

Arguments

res	an object of [KineticaResult-class]
...	Other parameters passed on to methods.

See Also

Other KineticaResult methods: `dbBind,KineticaResult-method`, `dbClearResult,KineticaResult-method`, `dbColumnInfo`, `dbFetch`, `dbGetInfo`, `dbGetRowCount,KineticaResult-method`, `dbGetStatement,KineticaResult-met`
`dbHasCompleted,KineticaResult-method`, `dbIsValid`, `show,KineticaDriver-method`

`dbGetStatement,KineticaResult-method` *dbGetStatement()*

Description

Returns the SQL statement executed to get this result set

Usage

```
## S4 method for signature 'KineticaResult'
dbGetStatement(res, ...)
```

Arguments

`res`	an object of [KineticaResult-class]
`...`	Other parameters passed on to methods.

See Also

Other KineticaResult methods: `dbBind,KineticaResult-method`, `dbClearResult,KineticaResult-method`, `dbColumnInfo`, `dbFetch`, `dbGetInfo`, `dbGetRowCount,KineticaResult-method`, `dbGetRowsAffected,KineticaResult-`
`dbHasCompleted,KineticaResult-method`, `dbIsValid`, `show,KineticaDriver-method`

`dbHasCompleted,KineticaResult-method` *dbHasCompleted()*

Description

Returns a logical value whether the result has completed

Usage

```
## S4 method for signature 'KineticaResult'
dbHasCompleted(res, ...)
```

Arguments

`res`	an object of [KineticaResult-class]
`...`	Other parameters passed on to methods.

See Also

Other KineticaResult methods: `dbBind,KineticaResult-method`, `dbClearResult,KineticaResult-method`, `dbColumnInfo`, `dbFetch`, `dbGetInfo`, `dbGetRowCount,KineticaResult-method`, `dbGetRowsAffected,KineticaResult-` `dbGetStatement,KineticaResult-method`, `dbIsValid`, `show,KineticaDriver-method`

`dbIsReadOnly` *dbIsReadOnly()*

Description

Generic dbIsReadOnly() implementation.

Usage

```
dbIsReadOnly(dbObj, ...)

## S4 method for signature 'KineticaObject'
dbIsReadOnly(dbObj, ...)
```

Arguments

`dbObj`	An object inheriting from [KineticaObject-class]
`...`	Other arguments to methods.

Value

logical

`dbIsValid` *dbIsValid()*

Description

Generic dbIsValid() implementation.

Checks if the provided Driver object is valid

Checks if the connection is valid

Checks if the KineticaResult object has been cleared

Usage

```
dbIsValid(dbObj, ...)

## S4 method for signature 'KineticaDriver'
dbIsValid(dbObj, ...)

## S4 method for signature 'KineticaConnection'
dbIsValid(dbObj, ...)

## S4 method for signature 'KineticaResult'
dbIsValid(dbObj, ...)
```

Arguments

`dbObj` An object inheriting from [KineticaObject-class], i.e. [KineticaDriver-class], [KineticaConnection-class], [KineticaId-class] or a [KineticaResult-class]

`...` Other arguments to methods.

`dbObj` an object of [KineticaDriver-class] type

`...` Other arguments omitted in generic signature

`dbObj` A [KineticaConnection-class] object

`...` Other arguments omitted in generic signature

`dbObj` [KineticaResult-class]

`...` Other parameters passed on to methods.

Details

RKinetica Connection is a configuration of url, username and password to establish connection to Kinetica DB, configuration does not go stale when memory objects get swapped to file. Thus RKinetica accepts stale connections as valid, and allows multiple results per connection.

Value

logical

See Also

Other KineticaDriver methods: `dbCanConnect,KineticaDriver-method`, `dbConnect,KineticaDriver-method`, `dbDataType,KineticaDriver-method`, `dbGetInfo`, `dbListConnections,KineticaDriver-method`, `dbUnloadDriver,KineticaDriver-method`, `show,KineticaDriver-method`

Other KineticaConnection methods: `dbAppendTable`, `dbCreateTable`, `dbDataType,KineticaConnection-method`, `dbDisconnect,KineticaConnection-method`, `dbExecute,KineticaConnection,character-method`, `dbExistsTable`, `dbGetInfo`, `dbGetQuery,KineticaConnection,character-method`, `dbListFields`, `dbListObjects,KineticaConnection-method`, `dbListResults,KineticaConnection-method`, `dbListTables,KineticaConnection-method`, `dbReadTable`, `dbRemoveTable`, `dbSendQuery,KineticaConnection,char`
`dbSendStatement,KineticaConnection,character-method`, `dbWriteTable`, `show,KineticaDriver-method`, `sqlCreateTable,KineticaConnection-method`, `transactions`

Other KineticaResult methods: `dbBind,KineticaResult-method`, `dbClearResult,KineticaResult-method`, `dbColumnInfo`, `dbFetch`, `dbGetInfo`, `dbGetRowCount,KineticaResult-method`, `dbGetRowsAffected,KineticaResult-` `dbGetStatement,KineticaResult-method`, `dbHasCompleted,KineticaResult-method`, `show,KineticaDriver-metho`

Examples

```
## Not run:
drv <- Kinetica()
dbIsValid(drv)
#  TRUE

## End(Not run)
## Not run:
con <- dbConnect(Kinetica(), url = "http://localhost:9191")
dbIsValid(con)
# TRUE
dbDisconnect(con)

## End(Not run)
```

`dbListConnections,KineticaDriver-method`

dbListConnections()

Description

Lists all active connections in KineticaDriver's connection pool invoking show() on KineticaConnection objects

Usage

```
## S4 method for signature 'KineticaDriver'
dbListConnections(drv, ...)
```

Arguments

`drv`	an object of [KineticaDriver-class] type
`...`	Other arguments omitted in generic signature

See Also

Other KineticaDriver methods: `dbCanConnect,KineticaDriver-method`, `dbConnect,KineticaDriver-method`, `dbDataType,KineticaDriver-method`, `dbGetInfo`, `dbIsValid`, `dbUnloadDriver,KineticaDriver-method`, `show,KineticaDriver-method`

dbListFields *dbListFields()*

Description

Lists field names of a given table

Usage

```
dbListFields(conn, name, ...)
```

Arguments

`conn`	A [KineticaConnection-class] object
`name`	a character string for table name or [KineticaId-class] object
`...`	Other parameters passed on to methods.

See Also

Other KineticaConnection methods: `dbAppendTable`, `dbCreateTable`, `dbDataType,KineticaConnection-method`, `dbDisconnect,KineticaConnection-method`, `dbExecute,KineticaConnection,character-method`, `dbExistsTable`, `dbGetInfo`, `dbGetQuery,KineticaConnection,character-method`, `dbIsValid`, `dbListObjects,KineticaConnection-method`, `dbListResults,KineticaConnection-method`, `dbListTables,KineticaConnection-method`, `dbReadTable`, `dbRemoveTable`, `dbSendQuery,KineticaConnection,char`
`dbSendStatement,KineticaConnection,character-method`, `dbWriteTable`, `show,KineticaDriver-method`, `sqlCreateTable,KineticaConnection-method`, `transactions`

Examples

```
## Not run:
con <- dbConnect(Kinetica(), url = "http://localhost:9191")
dbWriteTable(con, "tableA", data.frame(a = 1:40))
dbExistsTable(con, "tableA")
# TRUE
fields <- dbListFields(con, "tableA")
fields
# "a"
dbRemoveTable(con, "tableA")
dbDisconnect(con)

## End(Not run)
```

dbListObjects,KineticaConnection-method

dbListObjects()

Description

Lists database Objects available to the user, including tables, views and collections

Usage

```
## S4 method for signature 'KineticaConnection'
dbListObjects(conn, prefix = NULL, ...)
```

Arguments

`conn`	A [KineticaConnection-class] object
`prefix`	character
`...`	Other parameters passed on to methods.

Value

data.frame

See Also

Other KineticaConnection methods: `dbAppendTable`, `dbCreateTable`, `dbDataType,KineticaConnection-method`, `dbDisconnect,KineticaConnection-method`, `dbExecute,KineticaConnection,character-method`, `dbExistsTable`, `dbGetInfo`, `dbGetQuery,KineticaConnection,character-method`, `dbIsValid`, `dbListFields`, `dbListResults,KineticaConnection-method`, `dbListTables,KineticaConnection-method`, `dbReadTable`, `dbRemoveTable`, `dbSendQuery,KineticaConnection,character-method`, `dbSendStatement,KineticaC` `dbWriteTable`, `show,KineticaDriver-method`, `sqlCreateTable,KineticaConnection-method`, `transactions`

dbListResults,KineticaConnection-method

dbListResults()

Description

Lists all active results of the current connection

Usage

```
## S4 method for signature 'KineticaConnection'
dbListResults(conn, ...)
```

Arguments

conn	an object of [KineticaConnection-class]
...	Other parameters passed on to methods.

See Also

Other KineticaConnection methods: dbAppendTable, dbCreateTable, dbDataType,KineticaConnection-method, dbDisconnect,KineticaConnection-method, dbExecute,KineticaConnection,character-method, dbExistsTable, dbGetInfo, dbGetQuery,KineticaConnection,character-method, dbIsValid, dbListFields, dbListObjects,KineticaConnection-method, dbListTables,KineticaConnection-method, dbReadTable, dbRemoveTable, dbSendQuery,KineticaConnection,character-method, dbSendStatement,KineticaCo
dbWriteTable, show,KineticaDriver-method, sqlCreateTable,KineticaConnection-method, transactions

---

dbListTables,KineticaConnection-method

dbListTables()

Description

Lists all database tables available to the user

Usage

```
## S4 method for signature 'KineticaConnection'
dbListTables(conn, name, ...)
```

Arguments

conn	an object of [KineticaConnection-class]
name	character
...	Other arguments omitted in generic signature

See Also

Other KineticaConnection methods: dbAppendTable, dbCreateTable, dbDataType,KineticaConnection-method, dbDisconnect,KineticaConnection-method, dbExecute,KineticaConnection,character-method, dbExistsTable, dbGetInfo, dbGetQuery,KineticaConnection,character-method, dbIsValid, dbListFields, dbListObjects,KineticaConnection-method, dbListResults,KineticaConnection-method, dbReadTable, dbRemoveTable, dbSendQuery,KineticaConnection,character-method, dbSendStatement,KineticaCo
dbWriteTable, show,KineticaDriver-method, sqlCreateTable,KineticaConnection-method, transactions

`dbQuoteIdentifier` *dbQuoteIdentifier()*

Description

Escapes the provided character vector as SQL indetifier.

Escapes the provided character vector as SQL indetifier.

Usage

```
dbQuoteIdentifier(conn, x, ...)

## S4 method for signature 'KineticaConnection,ANY'
dbQuoteIdentifier(conn, x, ...)
```

Arguments

`conn`	object [KineticaConnection-class]
`x`	a character vector or SQL object
`...`	Other arguments passed on to methods.

`dbQuoteLiteral,KineticaConnection-method` *dbQuoteLiteral()*

Description

Escapes a literal if nesessary.

Usage

```
## S4 method for signature 'KineticaConnection'
dbQuoteLiteral(conn, x, ...)
```

Arguments

`conn`	A subclass of [KineticaConnection-class]
`x`	A vector to quote as string.
`...`	Other arguments passed on to methods.

dbQuoteString *dbQuoteString()*

Description

Excapes a character (string) value.

Excapes a character (string) value.

Excapes a character (string) value.

Excapes a character (string) value.

Usage

```
dbQuoteString(conn, x, ...)

## S4 method for signature 'KineticaConnection,ANY'
dbQuoteString(conn, x, ...)

## S4 method for signature 'KineticaConnection,character'
dbQuoteString(conn, x, ...)

## S4 method for signature 'KineticaConnection,SQL'
dbQuoteString(conn, x, ...)
```

Arguments

`conn`	A subclass of [KineticaConnection-class], representing an active connection to an DBMS.
`x`	A character vector to quote as string.
`...`	Other arguments passed on to methods.

dbReadTable *dbReadTable()*

Description

Reads the data of a given table into data.frame object, clearing the KineticaResult object upon exit

Usage

```
dbReadTable(conn, name, ...)
```

Arguments

`conn`	an object of [KineticaConnection-class]
`name`	a character string table name or [KineticaId-class] object
`...`	Other arguments omitted in generic signature
`row.names`	a logical flag to create extra row_names column
`check.names`	a logical flag to check names

See Also

Other KineticaConnection methods: `dbAppendTable`, `dbCreateTable`, `dbDataType,KineticaConnection-method`, `dbDisconnect,KineticaConnection-method`, `dbExecute,KineticaConnection,character-method`, `dbExistsTable`, `dbGetInfo`, `dbGetQuery,KineticaConnection,character-method`, `dbIsValid`, `dbListFields`, `dbListObjects,KineticaConnection-method`, `dbListResults,KineticaConnection-method`, `dbListTables,KineticaConnection-method`, `dbRemoveTable`, `dbSendQuery,KineticaConnection,character-metho` `dbSendStatement,KineticaConnection,character-method`, `dbWriteTable`, `show,KineticaDriver-method`, `sqlCreateTable,KineticaConnection-method`, `transactions`

`dbRemoveTable` *dbRemoveTable()*

Description

Drops the table with the given name if one exists

Usage

```
dbRemoveTable(conn, name, ...)
```

Arguments

`conn`	an object of [KineticaConnection-class]
`name`	character string for table name or [KineticaId-class] object
`...`	Other arguments omitted in generic signature

See Also

Other KineticaConnection methods: `dbAppendTable`, `dbCreateTable`, `dbDataType,KineticaConnection-method`, `dbDisconnect,KineticaConnection-method`, `dbExecute,KineticaConnection,character-method`, `dbExistsTable`, `dbGetInfo`, `dbGetQuery,KineticaConnection,character-method`, `dbIsValid`, `dbListFields`, `dbListObjects,KineticaConnection-method`, `dbListResults,KineticaConnection-method`, `dbListTables,KineticaConnection-method`, `dbReadTable`, `dbSendQuery,KineticaConnection,character-method`, `dbSendStatement,KineticaConnection,character-method`, `dbWriteTable`, `show,KineticaDriver-method`, `sqlCreateTable,KineticaConnection-method`, `transactions`

dbSendQuery,KineticaConnection,character-method *dbSendQuery()*

Description

Executes SQL query and returns a result set. The returned result object should be read with dbFetch(result) and then cleared with dbClearResult(result)

Usage

```
## S4 method for signature 'KineticaConnection,character'
dbSendQuery(conn, statement,
  params = NULL, offset = NULL, limit = NULL, ...)
```

Arguments

`conn`	object [KineticaConnection-class]
`statement`	character
`params`	a list of query named parameters
`...`	Other arguments omitted in generic signature

See Also

Other KineticaConnection methods: `dbAppendTable`, `dbCreateTable`, `dbDataType,KineticaConnection-method`, `dbDisconnect,KineticaConnection-method`, `dbExecute,KineticaConnection,character-method`, `dbExistsTable`, `dbGetInfo`, `dbGetQuery,KineticaConnection,character-method`, `dbIsValid`, `dbListFields`, `dbListObjects,KineticaConnection-method`, `dbListResults,KineticaConnection-method`, `dbListTables,KineticaConnection-method`, `dbReadTable`, `dbRemoveTable`, `dbSendStatement,KineticaConnection`, `dbWriteTable`, `show,KineticaDriver-method`, `sqlCreateTable,KineticaConnection-method`, `transactions`

dbSendStatement,KineticaConnection,character-method *dbSendStatement()*

Description

Executes SQL statement that does not return a result set

Usage

```
## S4 method for signature 'KineticaConnection,character'
dbSendStatement(conn, statement,
  params = NULL, offset = NULL, limit = NULL, ...)
```

Arguments

`conn`	object [KineticaConnection-class]
`statement`	character
`params`	a list of query named parameters
`...`	Other arguments omitted in generic signature

See Also

[dbSendQuery()] [dbGetQuery()] [dbExecute()]

Other KineticaConnection methods: `dbAppendTable`, `dbCreateTable`, `dbDataType,KineticaConnection-method`, `dbDisconnect,KineticaConnection-method`, `dbExecute,KineticaConnection,character-method`, `dbExistsTable`, `dbGetInfo`, `dbGetQuery,KineticaConnection,character-method`, `dbIsValid`, `dbListFields`, `dbListObjects,KineticaConnection-method`, `dbListResults,KineticaConnection-method`, `dbListTables,KineticaConnection-method`, `dbReadTable`, `dbRemoveTable`, `dbSendQuery,KineticaConnection,char`
`dbWriteTable`, `show,KineticaDriver-method`, `sqlCreateTable,KineticaConnection-method`, `transactions`

`dbUnloadDriver,KineticaDriver-method` *dbUnloadDriver()*

Description

Unload RKinetica Driver, releasing memory from Driver connections and embedded results

Usage

```
## S4 method for signature 'KineticaDriver'
dbUnloadDriver(drv, ...)
```

Arguments

`drv`	Kinetica driver object
`...`	Other arguments omitted in generic signature

See Also

Other KineticaDriver methods: `dbCanConnect,KineticaDriver-method`, `dbConnect,KineticaDriver-method`, `dbDataType,KineticaDriver-method`, `dbGetInfo`, `dbIsValid`, `dbListConnections,KineticaDriver-method`, `show,KineticaDriver-method`

dbUnquoteIdentifier *dbUnquoteIdentifier()*

Description

Unescapes the DB identifier.

Unescapes the DB identifier.

Unescapes the DB identifier.

Usage

```
dbUnquoteIdentifier(conn, x, ...)

## S4 method for signature 'KineticaConnection,SQL'
dbUnquoteIdentifier(conn, x, ...)

## S4 method for signature 'KineticaConnection,ANY'
dbUnquoteIdentifier(conn, x, ...)
```

Arguments

conn	object [KineticaConnection-class]
x	a character vector or SQL object
...	Other arguments passed on to methods.
conn	A subclass of [KineticaConnection-class], representing an active connection to an DBMS.
conn	A subclass of [KineticaConnection-class], representing an active connection to an DBMS.

dbWriteTable *dbWriteTable()*

Description

Writes a dataset into Kinetica DB, appending, creating or overwriting a table as indicated by flags.

Usage

```
dbWriteTable(conn, name, value, ...)
```

Arguments

`conn`	an object of [KineticaConnection-class]
`name`	character string for table name or [KineticaId-class] object
`value`	a [data.frame] (or object coercible to data.frame)
`...`	Other arguments omitted in generic signature
`row.names`	a logical flag, NULL or chaacter value to create extra row_names column
`overwrite`	a logical flag to overwrite existing table with new columns and values
`append`	a logical flag to preserve existing data and append new records
`field.types`	a named character vector of value field names and types
`temporary`	a logical flag to create table as temporary storage

See Also

[dbCreateTable()] [dbAppendTable()] [dbRemoveTable()]

Other KineticaConnection methods: `dbAppendTable`, `dbCreateTable`, `dbDataType,KineticaConnection-method`, `dbDisconnect,KineticaConnection-method`, `dbExecute,KineticaConnection,character-method`, `dbExistsTable`, `dbGetInfo`, `dbGetQuery,KineticaConnection,character-method`, `dbIsValid`, `dbListFields`, `dbListObjects,KineticaConnection-method`, `dbListResults,KineticaConnection-method`, `dbListTables,KineticaConnection-method`, `dbReadTable`, `dbRemoveTable`, `dbSendQuery,KineticaConnection,char`
`dbSendStatement,KineticaConnection,character-method`, `show,KineticaDriver-method`, `sqlCreateTable,Kinet`
`transactions`

Examples

```
## Not run:
con <- dbConnect(Kinetica(), url = "http://localhost:9191")
dbWriteTable(con, "test", data.frame(a = 1L:3L, b = 2.1:4.2), row.names = FALSE)
dbDisconnect(con)

## End(Not run)
```

`KineticaId-class` *Refer to a table nested in a hierarchy (e.g. within a schema)*

Description

Objects of class ‘KineticaId‘ have a single slot ‘name‘, which is a named character vector. The [dbQuoteIdentifier()] method converts ‘KineticaId‘ objects to strings. Support for ‘KineticaId‘ objects depends on the database backend. They can be used in the following methods as ‘name‘ argument: - [dbCreateTable()] - [dbAppendTable()] - [dbReadTable()] - [dbWriteTable()] - [dbExistsTable()] - [dbRemoveTable()] Objects of this class are also returned from [dbListObjects()].

Usage

```
KineticaId(...)
```

Arguments

`...` Components of the hierarchy, e.g. ‘schema‘, ‘table‘, or ‘cluster‘, ‘catalog‘, ‘schema‘, ‘table‘. For more on these concepts, see http://stackoverflow.com/questions/7022755/

Examples

```
KineticaId(schema = "dbo", table = "Customer")
dbQuoteIdentifier(Kinetica(), KineticaId(schema = "ki_home", table = "nyctaxi"))
KineticaId(schema = "myschema", table = "mytable")
```

KineticaObject-class *Class KineticaObject*

Description

Base DB object extending [DBI::DBIObject].

RKinetica *RKinetica*

Description

RKinetica is an R package providing access to Kinetica DB through REST interface. Kinetica DB is a distributed, in-memory, GPU-accelerated database that utilizes a powerful combination of CPUs and GPUs to analyze massive, complex datasets with millisecond response times.

Details

At the core of Kinetica is a vectorized, columnar, memory-first database designed for analytical (OLAP) workloads. Kinetica automatically distributes any workload across CPUs and GPUs for optimal results.

For more information on Kinetica DB please visit our web site http://www.kinetica.com

`show,KineticaDriver-method` *show()*

Description

RKinetica Driver short printout string

Minimal KineticaConnection string representation Provides url of the Kinetica DB instance

Brief string printout of [KineticaResult-class] object properties

Usage

```
## S4 method for signature 'KineticaDriver'
show(object)

## S4 method for signature 'KineticaConnection'
show(object)

## S4 method for signature 'KineticaResult'
show(object)
```

Arguments

`object`	[KineticaConnection-class]
`object`	[KineticaResult-class]

See Also

Other KineticaDriver methods: `dbCanConnect,KineticaDriver-method`, `dbConnect,KineticaDriver-method`, `dbDataType,KineticaDriver-method`, `dbGetInfo`, `dbIsValid`, `dbListConnections,KineticaDriver-method`, `dbUnloadDriver,KineticaDriver-method`

Other KineticaConnection methods: `dbAppendTable`, `dbCreateTable`, `dbDataType,KineticaConnection-method`, `dbDisconnect,KineticaConnection-method`, `dbExecute,KineticaConnection,character-method`, `dbExistsTable`, `dbGetInfo`, `dbGetQuery,KineticaConnection,character-method`, `dbIsValid`, `dbListFields`, `dbListObjects,KineticaConnection-method`, `dbListResults,KineticaConnection-method`, `dbListTables,KineticaConnection-method`, `dbReadTable`, `dbRemoveTable`, `dbSendQuery,KineticaConnection,char`
`dbSendStatement,KineticaConnection,character-method`, `dbWriteTable`, `sqlCreateTable,KineticaConnection-m`
`transactions`

Other KineticaResult methods: `dbBind,KineticaResult-method`, `dbClearResult,KineticaResult-method`, `dbColumnInfo`, `dbFetch`, `dbGetInfo`, `dbGetRowCount,KineticaResult-method`, `dbGetRowsAffected,KineticaResult-`
`dbGetStatement,KineticaResult-method`, `dbHasCompleted,KineticaResult-method`, `dbIsValid`

Examples

```
## Not run:
drv <- Kinetica()
show(drv)
#  KineticaDriver 7.1.0.0

## End(Not run)
## Not run:
con <- dbConnect(d, url = ”http://localhost:9191”)
show(con)
<KineticaConnection>
  url: http://localhost:9191

## End(Not run)
```

`show,SQL-method` *show()*

Description

prints escaped SQL values.

Usage

```
## S4 method for signature 'SQL'
show(object)
```

Arguments

`object` SQL object

`SQL` *SQL()*

Description

Creates an escaped SQL representation of provided named character vector.

Usage

```
SQL(x, ..., names = NULL)
```

Arguments

`x` A character vector to label as being escaped SQL.

`...` Other arguments passed on to methods. Not otherwise used.

`names` Names for the returned object, must have the same length as ‘x‘.

Value

An object of class ‘SQL‘.

`sqlAppendTable` *sqlAppendTable*

Description

Compose query to insert rows into a table

Usage

```
sqlAppendTable(con, table, values, row.names = NA, ...)
```

Arguments

`con`	A database connection.
`table`	Name of the table. Escaped with [dbQuoteIdentifier()].
`values`	A data frame. Factors will be converted to character vectors.
`row.names`	a flag with logical, character or NULL value
`...`	Other parameters passed on to methods. Character vectors will be escaped with [dbQuoteString()].

Details

‘sqlAppendTable()‘ generates a single SQL string that inserts a data frame into an existing table. ‘sqlAppendTableTemplate()‘ generates a template suitable for use with [dbBind()]. The default methods are ANSI SQL 99 compliant. These methods are mostly useful for backend implementers.

The ‘row.names‘ argument must be passed explicitly in order to avoid a compatibility warning. The default will be changed in a later release.

See Also

Other SQL generation: `sqlAppendTableTemplate`

sqlAppendTableTemplate *sqlAppendTableTemplate()*

Description

Creates a placeholder-based template

Usage

```
sqlAppendTableTemplate(con, table, values, row.names = NA,
  prefix = "?", ..., pattern = "1")
```

Arguments

`con`	A database connection.
`table`	Name of the table. Escaped with [dbQuoteIdentifier()].
`values`	A data frame. Factors will be converted to character vectors.
`row.names`	a flag with logical, character or NULL value
`prefix`	Parameter prefix to use for placeholders.
`...`	Other parameters passed on to methods. Character vectors will be escaped with [dbQuoteString()].
`pattern`	Parameter pattern to use for placeholders: - ‘""‘: no pattern - ‘"1"‘: position - anything else: field name

See Also

Other SQL generation: `sqlAppendTable`

sqlCreateTable,KineticaConnection-method *sqlCreateTable()*

Description

Customised version of the generic sqlCreateTable method (Kinetica DB uses TEMP keyword instead of TEMPORARY)

Usage

```
## S4 method for signature 'KineticaConnection'
sqlCreateTable(con, table, fields,
  row.names = NA, temporary = FALSE, ...)
```

Arguments

`con`	A database connection.
`table`	Name of the table. Escaped with [dbQuoteIdentifier()].
`fields`	Either a character vector or a data frame. A named character vector: Names are column names, values are types. Names are escaped with [dbQuoteIdentifier()]. Field types are unescaped. A data frame: field types are generated using [dbDataType()].
`row.names`	a flag with logical, character or NULL value
`temporary`	logical TRUE/FALSE for temporary/permanent table created
`...`	Other parameters passed on to methods.

See Also

Other KineticaConnection methods: `dbAppendTable`, `dbCreateTable`, `dbDataType,KineticaConnection-method`, `dbDisconnect,KineticaConnection-method`, `dbExecute,KineticaConnection,character-method`, `dbExistsTable`, `dbGetInfo`, `dbGetQuery,KineticaConnection,character-method`, `dbIsValid`, `dbListFields`, `dbListObjects,KineticaConnection-method`, `dbListResults,KineticaConnection-method`, `dbListTables,KineticaConnection-method`, `dbReadTable`, `dbRemoveTable`, `dbSendQuery,KineticaConnection,char` `dbSendStatement,KineticaConnection,character-method`, `dbWriteTable`, `show,KineticaDriver-method`, `transactions`

`sqlData` *sqlData()*

Description

Escapes and encodes SQL data.

Escapes and encodes SQL data.

Usage

```
sqlData(con, value, row.names = NA, ...)

## S4 method for signature 'KineticaConnection'
sqlData(con, value, row.names = NA, ...)
```

Arguments

`con`	A database connection, object of [KineticaConnection-class]
`value`	A data frame
`row.names`	a flag with logical, character or NULL value
`...`	Other parameters passed on to methods.
`con`	A database connection, object of [KineticaConnection-class]
`row.names`	a flag with logical, character or NULL value
`...`	Other parameters passed on to methods.
`value`	A data frame

Examples

```
## Not run:
con <- dbConnect(RSQLite::SQLite(), ":memory:")

sqlData(con, head(iris))
sqlData(con, head(mtcars))

dbDisconnect(con)

## End(Not run)
```

summary *summary()*

Description

Generic summary() implementation.

Usage

```
summary(object, ...)
```

summary,KineticaObject-method *summary()*

Description

Implementation of summary() for KineticaObject signature.

Usage

```
## S4 method for signature 'KineticaObject'
summary(object, ...)
```

Arguments

object	An object inheriting from [KineticaObject-class]
...	Other arguments to methods.

transactions *dbBegin()*

Description

Executes the statement, returning the number of affected rows or objects

Executes the statement, returning the number of affected rows or objects

Executes the statement, returning the number of affected rows or objects

Usage

```
## S4 method for signature 'KineticaConnection'
dbBegin(conn, ...)

## S4 method for signature 'KineticaConnection'
dbCommit(conn, ...)

## S4 method for signature 'KineticaConnection'
dbRollback(conn, ...)
```

Arguments

`conn`	an object of [KineticaConnection-class]
`...`	Other parameters passed on to methods.
`conn`	an object of [KineticaConnection-class]
`...`	Other parameters passed on to methods.
`conn`	an object of [KineticaConnection-class]
`...`	Other parameters passed on to methods.

See Also

Other KineticaConnection methods: `dbAppendTable`, `dbCreateTable`, `dbDataType,KineticaConnection-method`, `dbDisconnect,KineticaConnection-method`, `dbExecute,KineticaConnection,character-method`, `dbExistsTable`, `dbGetInfo`, `dbGetQuery,KineticaConnection,character-method`, `dbIsValid`, `dbListFields`, `dbListObjects,KineticaConnection-method`, `dbListResults,KineticaConnection-method`, `dbListTables,KineticaConnection-method`, `dbReadTable`, `dbRemoveTable`, `dbSendQuery,KineticaConnection,char`
`dbSendStatement,KineticaConnection,character-method`, `dbWriteTable`, `show,KineticaDriver-method`, `sqlCreateTable,KineticaConnection-method`

Other KineticaConnection methods: `dbAppendTable`, `dbCreateTable`, `dbDataType,KineticaConnection-method`, `dbDisconnect,KineticaConnection-method`, `dbExecute,KineticaConnection,character-method`, `dbExistsTable`, `dbGetInfo`, `dbGetQuery,KineticaConnection,character-method`, `dbIsValid`, `dbListFields`, `dbListObjects,KineticaConnection-method`, `dbListResults,KineticaConnection-method`, `dbListTables,KineticaConnection-method`, `dbReadTable`, `dbRemoveTable`, `dbSendQuery,KineticaConnection,char`
`dbSendStatement,KineticaConnection,character-method`, `dbWriteTable`, `show,KineticaDriver-method`, `sqlCreateTable,KineticaConnection-method`

Other KineticaConnection methods: dbAppendTable, dbCreateTable, dbDataType,KineticaConnection-method, dbDisconnect,KineticaConnection-method, dbExecute,KineticaConnection,character-method, dbExistsTable, dbGetInfo, dbGetQuery,KineticaConnection,character-method, dbIsValid, dbListFields, dbListObjects,KineticaConnection-method, dbListResults,KineticaConnection-method, dbListTables,KineticaConnection-method, dbReadTable, dbRemoveTable, dbSendQuery,KineticaConnection,char dbSendStatement,KineticaConnection,character-method, dbWriteTable, show,KineticaDriver-method, sqlCreateTable,KineticaConnection-method

Index

∗Topic **package**
RKinetica, 32

dbAppendTable, 3, 8, 9, 11, 12, 15, 16, 20, 22–24, 27–29, 31, 33, 37, 39, 40
dbBind,KineticaResult-method, 3
dbCanConnect,KineticaDriver-method, 4
dbClearResult,KineticaResult-method, 5
dbColumnInfo, 4, 5, 6, 14, 15, 17–19, 21, 33
dbColumnInfo,KineticaResult-method (dbColumnInfo), 6
dbCommit,KineticaConnection-method (transactions), 39
dbConnect,KineticaDriver-method, 6
dbCreateTable, 3, 8, 9, 11, 12, 15, 16, 20, 22–24, 27–29, 31, 33, 37, 39, 40
dbDataType,KineticaConnection-method, 9
dbDataType,KineticaDriver-method, 9
dbDisconnect,KineticaConnection-method, 10
dbExecute,KineticaConnection,character-method, 11
dbExistsTable, 3, 8, 9, 11, 12, 15, 16, 20, 22–24, 27–29, 31, 33, 37, 39, 40
dbFetch, 4–6, 13, 15, 17–19, 21, 33
dbFetch,KineticaResult-method (dbFetch), 13
dbGetInfo, 3–12, 14, 14, 16–24, 27–29, 31, 33, 37, 39, 40
dbGetInfo,KineticaConnection-method (dbGetInfo), 14
dbGetInfo,KineticaDriver-method (dbGetInfo), 14
dbGetInfo,KineticaResult-method (dbGetInfo), 14
dbGetQuery,KineticaConnection,character-method, 16
dbGetRowCount,KineticaResult-method, 17
dbGetRowsAffected,KineticaResult-method, 17
dbGetStatement,KineticaResult-method, 18
dbHasCompleted,KineticaResult-method, 18
dbIsReadOnly, 19
dbIsReadOnly,KineticaObject-method (dbIsReadOnly), 19
dbIsValid, 3–12, 14–19, 19, 21–24, 27–29, 31, 33, 37, 39, 40
dbIsValid,KineticaConnection-method (dbIsValid), 19
dbIsValid,KineticaDriver-method (dbIsValid), 19
dbIsValid,KineticaResult-method (dbIsValid), 19
dbListConnections,KineticaDriver-method, 21
dbListFields, 3, 8, 9, 11, 12, 15, 16, 20, 22, 23, 24, 27–29, 31, 33, 37, 39, 40
dbListObjects,KineticaConnection-method, 23
dbListResults,KineticaConnection-method, 23
dbListTables,KineticaConnection-method, 24
dbQuoteIdentifier, 25
dbQuoteIdentifier,KineticaConnection,ANY-method (dbQuoteIdentifier), 25
dbQuoteLiteral,KineticaConnection-method, 25
dbQuoteString, 26
dbQuoteString,KineticaConnection,ANY-method (dbQuoteString), 26
dbQuoteString,KineticaConnection,character-method (dbQuoteString), 26
dbQuoteString,KineticaConnection,SQL-method (dbQuoteString), 26

dbReadTable, *3*, *8*, *9*, *11*, *12*, *15*, *16*, *20*, *22–24*, 26, *27–29*, *31*, *33*, *37*, *39*, *40*
dbRemoveTable, *3*, *8*, *9*, *11*, *12*, *15*, *16*, *20*, *22–24*, *27*, 27, *28*, *29*, *31*, *33*, *37*, *39*, *40*
dbRollback,KineticaConnection-method *(*transactions*)*, 39
dbSendQuery,KineticaConnection,character-method, 28
dbSendStatement,KineticaConnection,character-method, 28
dbUnloadDriver,KineticaDriver-method, 29
dbUnquoteIdentifier, 30
dbUnquoteIdentifier,KineticaConnection,ANY-method *(*dbUnquoteIdentifier*)*, 30
dbUnquoteIdentifier,KineticaConnection,SQL-method *(*dbUnquoteIdentifier*)*, 30
dbWriteTable, *3*, *8*, *9*, *11*, *12*, *15*, *16*, *20*, *22–24*, *27–29*, 30, *33*, *37*, *39*, *40*

KineticaId *(*KineticaId-class*)*, 31
KineticaId-class, 31
KineticaObject *(*KineticaObject-class*)*, 32
KineticaObject-class, 32

RKinetica, 32
RKinetica-package *(*RKinetica*)*, 32

show,KineticaConnection-method *(*show,KineticaDriver-method*)*, 33
show,KineticaDriver-method, 33
show,KineticaResult-method *(*show,KineticaDriver-method*)*, 33
show,SQL-method, 34
SQL, 34
sqlAppendTable, 35, *36*
sqlAppendTableTemplate, *35*, 36
sqlCreateTable,KineticaConnection-method, 36
sqlData, 37
sqlData,KineticaConnection-method *(*sqlData*)*, 37
summary, 38
summary,KineticaObject-method, 38

transactions, *3*, *8*, *9*, *11*, *12*, *15*, *16*, *20*, *22–24*, *27–29*, *31*, *33*, *37*, 39