Package 'sqlcaser'

November 24, 2023

Title 'SQL' Case Statement Generator

Version 0.2.0

Date 2023-11-21

Description Includes built-in methods for generating long 'SQL' CASE statements, and other 'SQL' statements that may otherwise be arduous to construct by hand. The generated statement can easily be concatenated to string literals to form queries to 'SQL'-like databases, such as when using the 'RODBC' package. The current methods include casewhen() for building CASE statements, inlist() for building IN statements, and updatetable() for building UPDATE statements.

License MIT + file LICENSE

Encoding UTF-8

RoxygenNote 7.2.3

Suggests knitr, rmarkdown

VignetteBuilder knitr

NeedsCompilation no

Author Leoson Hoay [aut, cre] (<https://orcid.org/0000-0003-2079-5579>)

Maintainer Leoson Hoay <leoson.public@gmail.com>

Repository CRAN

Date/Publication 2023-11-24 13:20:02 UTC

R topics documented:

casewhen *Generate a SQL CASE statement from a mapping file*

Description

This function constructs a CASE..WHEN,,THEN statement from a mapping file or dataframe. It assumes that the first column of the mapping data contains the original WHEN values, and the second column contains the THEN values (the values to be mapped to.)

Usage

```
casewhen(inputfile = NULL, header = FALSE)
```

Arguments

inputfile	Mapping dataframe OR path to the mapping file
header	If reading a csv file, TRUE if the file includes a header row, FALSE if it does not include a header row.

Value

A string that represents the constructed CASE statement

Examples

```
input <- Data_Frame <- data.frame(Training = c("Strength", "Stamina",
 "Other"), Duration = c(60, 30, 45))
result <- casewhen(inputfile = input, header = TRUE)
```

inlist *Generate a SQL IN statement from a mapping file*

Description

This function constructs an IN statement from a mapping file or dataframe. It assumes that the first column of the data contains the list of values to check for.

Usage

```
inlist(inputfile = NULL, header = FALSE)
```

Arguments

inputfile	Dataframe OR path to the mapping file
header	If reading a csv file, TRUE if the file includes a header row, FALSE if it does not include a header row.

Value

A string that represents the constructed CASE statement

Examples

```
input <- Data_Frame <- data.frame(Training = c("Strength", "Stamina",
 "Other"))
result <- inlist(inputfile = input, header = TRUE)
```

`updatetable` *Generate a SQL UPDATE statement from a mapping file*

Description

This function constructs an UPDATE statement from a mapping file or dataframe. It assumes that the first column of the data contains the key column and list of keys for the rows where the corresponding other columns have to be updated. It also assumes that the header for the data includes the column names. The function will generate one UPDATE statement for each row in the data.

Usage

```
updatetable(inputfile = NULL, tablename = NULL)
```

Arguments

`inputfile`	Dataframe OR path to the mapping file
`tablename`	Name of the SQL table

Value

A string that represents the constructed UPDATE statement(s)

Examples

```
input <- Data_Frame <- data.frame(Training = c("Strength", "Stamina",
 "Other"), Pulse = c(100, 150, 120), Duration = c(60, 30, 45))
result <- updatetable(inputfile = input, tablename = "myTable")
```

Index