Package 'instatCalculations'

December 3, 2024

Title Calculation system used in R-Instat

Version 0.1.0

Description Functions used in the calculation system used in R-Instat

License LGPL (>= 3)

Encoding UTF-8

Roxygen list(markdown = TRUE)

RoxygenNote 7.3.2

Imports dplyr,
magrittr,
R6,
rlang

Contents

calculation 1
calc_from_convert 3
check_filter 4
find_df_from_calc_from 4
instat_calculation 5

Index **7**

`calculation` *Calculation Class*

Description

Represents a calculation with associated parameters, filters, and sub-calculations.

Methods

`add_sub_calculation(sub_calculation, name)` Add a sub-calculation.

`data_clone(...)` Clone the data.

Public fields

`function_name` Character. The name of the function.

`parameters` List. The parameters for the calculation.

`calculated_from` Character vector. Sources from which the calculation is derived.

`is_recalculable` Logical. Indicates if the calculation is recalculable.

`sub_calculations` List. A list of sub-calculations.

`type` Character. The type of calculation.

`filter_conditions` List. The conditions used to filter data.

`filters` List. The filters applied to the data.

`name` Character. The name of the calculation instance.

Methods

Public methods:

- `calculation$new()`
- `calculation$add_sub_calculation()`
- `calculation$data_clone()`
- `calculation$clone()`

Method `new()`**:** Initialize the calculation class.

Usage:

```
calculation$new(
  function_name = "",
  parameters = list(),
  calculated_from = c(),
  is_recalculable = TRUE,
  sub_calculations = list(),
  type = "",
  filter_conditions = list(),
  filters = list(),
  name = ""
)
```

Arguments:

`function_name` Character. The name of the function. Default is an empty string.

`parameters` List. The parameters for the calculation. Default is an empty list.

`calculated_from` Character vector. Sources from which the calculation is derived. Default is an empty vector.

`is_recalculable` Logical. Indicates if the calculation is recalculable. Default is `TRUE`.

`sub_calculations` List. A list of sub-calculations. Default is an empty list.

`type` Character. The type of calculation. Default is an empty string.

`filter_conditions` List. The conditions used to filter data. Default is an empty list.

`filters` List. The filters applied to the data. Default is an empty list.

`name` Character. The name of the calculation instance. Default is an empty string.

Method `add_sub_calculation()`**:** Add a sub-calculation.

Usage:

```
calculation$add_sub_calculation(sub_calculation, name)
```

Arguments:

`sub_calculation` An object representing the sub-calculation to add.

`name` Character. The name of the sub-calculation.

Method `data_clone()`**:** Clone the data.

Usage:

```
calculation$data_clone(...)
```

Arguments:

`...` Additional methods to add to the function.

Returns: A new instance of the calculation class with the same data.

Method `clone()`**:** The objects of this class are cloneable with this method.

Usage:

```
calculation$clone(deep = FALSE)
```

Arguments:

`deep` Whether to make a deep clone.

`calc_from_convert` *Convert calculation list to a specific format*

Description

Convert calculation list to a specific format

Usage

```
calc_from_convert(x)
```

Arguments

`x`	A list of calculations.

Value

A formatted list of calculations.

check_filter *Check and update filter object parameters*

Description

Check and update filter object parameters

Usage

```
check_filter(filter_obj)
```

Arguments

filter_obj	A filter object to check and update.

Value

The updated filter object.

find_df_from_calc_from *Find data frame from calculation list*

Description

Find data frame from calculation list

Usage

```
find_df_from_calc_from(x, column)
```

Arguments

x	A list of calculations.
column	The column name to search for.

Value

The name of the data frame associated with the column.

`instat_calculation` *instat_calculation Class*

Description

instat_calculation Class

instat_calculation Class

Details

A class to store calculations.

Methods

`get_dependencies(depens = c())` Get Dependencies.

`data_clone(...)` Clone the data.

Public fields

`function_exp` A string passed directly to one of dplyr functions.

`type` The type of calculation.

`name` The name of the calculation instance.

`result_name` The name for the output produced by the calculation.

`result_data_frame` The data frame that the output should go to.

`manipulations` A list of calculations to be performed before sub_calculations and the main calculation.

`sub_calculations` A list of calculations to be performed after manipulations.

`calculated_from` A list of columns the calculation depends on.

`save` An integer indicating whether the calculation and result should be saved.

`before` A boolean indicating if the calculation should be performed before others.

`adjacent_column` The name of the adjacent column.

Methods

Public methods:

- `instat_calculation$new()`
- `instat_calculation$data_clone()`
- `instat_calculation$get_dependencies()`
- `instat_calculation$clone()`

Method `new()`**:** Initialise the instat_calculation class

Usage:

```
instat_calculation$new(
  function_exp = "",
  type = "",
  name = "",
  result_name = "",
  result_data_frame = "",
  manipulations = list(),
  sub_calculations = list(),
  calculated_from = list(),
  save = 0,
  before = FALSE,
  adjacent_column = ""
)
```

Arguments:

`function_exp` A string passed directly to one of dplyr functions.

`type` The type of calculation.

`name` The name of the calculation instance.

`result_name` The name for the output produced by the calculation.

`result_data_frame` The data frame that the output should go to.

`manipulations` A list of calculations to be performed before sub_calculations and the main calculation.

`sub_calculations` A list of calculations to be performed after manipulations.

`calculated_from` A list of columns the calculation depends on.

`save` An integer indicating whether the calculation and result should be saved.

`before` A boolean indicating if the calculation should be performed before others.

`adjacent_column` The name of the adjacent column.

Method `data_clone()`**:** Clone the data

Usage:

```
instat_calculation$data_clone(...)
```

Arguments:

`...` Additional methods to add to the function.

Returns: A new instance of the instat_calculation class with the same data.

Method `get_dependencies()`**:** Get dependencies

Usage:

```
instat_calculation$get_dependencies(depens = c())
```

Arguments:

`depens` A vector of dependencies.

Returns: A vector of dependencies.

Method `clone()`**:** The objects of this class are cloneable with this method.

Usage:

```
instat_calculation$clone(deep = FALSE)
```

Arguments:

`deep` Whether to make a deep clone.

Index

calc_from_convert, 3
calculation, 1
check_filter, 4

find_df_from_calc_from, 4

instat_calculation, 5