Package 'clog'

December 10, 2019

Type Package

Title cleaning log operations

Version 0.1.0

Author Martin Barner

Maintainer Martin Barner <martin.barner@impact-initiatives.org>

Description translate between raw and clean data based on value-by-value data cleaning logs

License GPL-3

Encoding UTF-8

LazyData true

Imports dplyr,
  tidyr,
  knitr,
  magrittr,
  purrr,
  rlang,
  tibble,
  crayon

RoxygenNote 6.1.1

Suggests testthat,
  rmarkdown

Depends assertthat

VignetteBuilder knitr

R topics documented:

---

cleaninglog *create standardised cleaninglog*

Description

create standardised cleaninglog

Usage

```
cleaninglog(ids, variables, new_values, data_id_column_name,
  name = "default_cleaninglog_element_name", info = NULL, ...)
```

Arguments

ids	vector of ids that can be matched with the dataset (i.e. uuids)
new_values	vector of new (corrected) values. If no change should be made, this must be set to the original value
data_id_column_name	
	the name of the column in the dataset with ids matching the values in 'ids'
name	a vector name of the data check that produced the
info	(optional): a character vector with additional information about the cleaning log
...	additional vectors or data.frames to be added to the cleaning log.
variabels	vector of variable names (should match data column headers)

Value

a 'clog' cleaninglog objects to be used with 'clog_clean()' etc.

---

clog *clog: Reproducible data modification with cleaning logs*

Description

run browseVignettes("clog") for more info

---

clog_clean *make changes to raw dataset based on cleaninglog (see 'clean-inglog()')*

Description

make changes to raw dataset based on cleaninglog (see 'cleaninglog()')

Usage

```
clog_clean(df, cleaninglog)
```

Arguments

df   a data.frame / tibble

cleaninglog   a cleaninglog object create with 'cleaninglog()'

Value

the original dataframe ('df') modified based on the cleaning log.

---

clog_cleaninglog_to_data

add cleaninglog elements as columns to a datset

Description

add cleaninglog elements as columns to a datset

Usage

```
clog_cleaninglog_to_data(data, cleaninglog)
```

Arguments

data   the dataset to add cleaninglog to as columns

cl   cleaninglog

Value

the original 'data' with one additional column per unique value cleaninglog 'name'. The 'name' becomes the column name. For each row in the data, the new column is 'TRUE' if a cleaninglog element with that name was listed for the respective row

---

clog_data_to_cleaninglog

add cleaninglog elements as columns to a datset

Description

add cleaninglog elements as columns to a datset

Usage

```
clog_data_to_cleaninglog(data, cleaninglog, ...)
```

Arguments

data	the dataset to select variabes from that should be added to the cleaning log
cleaninglog	the cleaninglog (see 'cleaninglog()')
...	names of the variables to be added from the dataset to the cleaninglog (without quotes!)

Details

returning 'NA" if the id in the cleaning log can not be found in the data

Value

the original 'data' with one additional column per unique value cleaninglog 'name'. The 'name' becomes the column name. For each row in the data, the new column is 'TRUE' if a cleaninglog element with that name was listed for the respective row

Examples

```
## Not run:
clog_data_to_cleaninglog(mydata,my_cleaninglog, a_data_variable_name, another_data_variable_name, `a data va

## End(Not run)
```

---

clog_get_cleaninglog   *extract cleaninglog from dataset modified with 'clog_clean()'*

Description

extract cleaninglog from dataset modified with 'clog_clean()'

Usage

```
clog_get_cleaninglog(df)
```

Arguments

df	a dataset modified with 'clog_clean()'

Value

the cleaninglog as it was applied to 'df'

---

clog_reverse	*reverse changes applied based on cleaning log*

Description

reverse changes applied based on cleaning log

Usage

```
clog_reverse(df)
```

Arguments

df   a data frame modified with 'clog_clean()'

Value

the original cleaning log applied in reverse.

---

hello	*Hello, World!*

Description

Prints 'Hello, world!'.

Usage

```
hello()
```

Examples

```
hello()
```

Index