Package ‘observer’
January 29, 2017

Type Package
Title Observe and Check your Data
Version 0.1.2
Date 2017-01-25
Description Checks that a given dataset passes user-specified rules. The main functions are observe_if() and inspect().
License MIT + file LICENSE
LazyData TRUE
Depends R (>= 3.1.3)
Imports bazar, bit, dplyr, lazyeval, magrittr, tibble, utils
VignetteBuilder knitr
Suggests knitr, ggplot2, testthat
URL https://github.com/paulponcet/observer
BugReports https://github.com/paulponcet/observer/issues
RoxygenNote 5.0.1
NeedsCompilation no
Author Paul Poncet [aut, cre]
Maintainer Paul Poncet <paulponcet@yahoo.fr>
Repository CRAN
Date/Publication 2017-01-29 22:03:43

R topics documented:

check_that .................................................. 2
inspect ....................................................... 3
observations ................................................ 4
observe_if ................................................... 5
reobserve ..................................................... 6
View_obs ..................................................... 7

Index 8
check_that  

Description

The function check_that returns FALSE if any of the predicates tested fails, TRUE otherwise. Where check_that would return FALSE, the function ensure_that throws an error instead; otherwise it returns .data unchanged.

Usage

check_that(.data, ...)

check_that_(.data, ..., .dots)

cHECK(.data, ...)

ensure_that(.data, ...)

ensure_that_(.data, ..., .dots)

ensure(.data, ...)

Arguments

.data  A tbl or data.frame.

...  Logical predicates. Multiple conditions are considered as separate observations.

.dots  Used to work around non-standard evaluation.

Value

check_that returns a logical, TRUE if all checks have passed, FALSE otherwise.

ensure_that throws an error if a check fails; otherwise, .data is returned (with NULL assigned to the observations attribute).

Note

These functions are inspired by eponymous functions in package ensurer.

See Also

observe_if in this package;
check_that and ensure_that from package ensurer; check_that from package validate.
Examples

library(magrittr)

observe.mydata <- function(.data, ...) {
  observe_if_(.
    ~ Year > 2010,
    ~ City %in% c("Paris", "New York"),
    ~ Population > 0)
}

df <- data.frame(city = c("paris", "new york", "amsterdam"),
  year = c(2011L, 2015L, 2016L),
  population = c(2249975L, 8550405L, 840486L))
class(df) <- c("mydata", "data.frame")
observe(df)
observe(df) %>% obs()
check(df) # FALSE
## Not run:
ensure(df) # throws an error

## End(Not run)

inspect

Inspect observations

Description

Once a table has been observed with observe_if, one may wish to dig into the observations made. The function inspect returns a sub-table of .data whose rows correspond to the rows identified by the observation ob.

Usage

inspect(.data, ob, cols = "all")

Arguments

.data A tbl or data frame.
.ob integer. A row of the observations attribute to be inspected.
.cols character. One of "all", "most", "some". If cols = "all", all .data columns are kept in the result. If cols = "some", only columns that are concerned by the observation ob are kept. cols = "most" is like cols = "some", except that columns which are of type "character" or "factor" are also kept.

See Also

observe_if in this package.
Examples

```r
library(magrittr)

df <- data.frame(id = c("a", "b", "c"), x = 1:3, y = 2:4, z = c(1.1, 10, 5))

df <- df %>%
  observe_if(z==floor(z),
    y-x==1,
    x < 3) %>%
  observe_if(y < 4,
    x > 1)

print(obs(df))

df1 <- df %>%
  inspect(ob = 5, cols="most")

print(obs(df1))
```

---

**observations**

*Get or set the 'observations' attribute*

**Description**

The function `observations` (or `obs` for short) extracts the observations attribute from .data created by `observe_if` or `observe_if_`.

**Usage**

```r
observations(.data, compressed = TRUE)

observations(x) <- value

obs(.data, compressed = TRUE)

obs(x) <- value
```

**Arguments**

- `.data, x` A tbl or data frame
- `compressed` logical. By default, the column "Rows" of observations is compressed with the `as.bit` function from package `bit`. If `compressed=FALSE`, this column is uncompressed (and becomes understandable / readable by the user).
- `value` A tibble, see `observations`.

**Value**

A tbl.
**observe_if**

**See Also**

- `observe_if` in this package.

---

### Description

`observe_if` identifies rows of `.data` where the predicates tested fail.

### Usage

```r
observe_if(.data, ..., .append = TRUE)
```

```r
observe_if(.data, ..., .dots, .append = TRUE)
```

```r
## S3 method for class 'data.frame'
observe_if(.data, ..., .dots, .append = TRUE)
```

```r
## S3 method for class 'tbl_df'
observe_if(.data, ..., .dots, .append = TRUE)
```

```r
## S3 method for class 'grouped_df'
observe_if(.data, ..., .dots, .append = TRUE)
```

```r
observe(.data, ...)
```

```r
## Default S3 method:
observe(.data, ...)
```

### Arguments

- `.data` A tbl or data.frame.
- `...` Logical predicates. Multiple conditions are considered as separate observations.
- `.append` logical. If TRUE (the default), the observations attribute, in case it already exists before the call, is appended; if FALSE it is updated.
- `.dots` Used to work around non-standard evaluation.

### Value

`.data` is returned together with an `observations` attribute. This attribute is a data frame (actually a tibble) which contains all check runs.
Examples

library(magrittr)

df <- data.frame(x = 1:3, y = 2:4)
df %>%
  observe_if(y-x==1) %>%
  observe_if(x < 3) %>%
  observe_if(y < 4, x > 1) %>%
  obs()

reobserve

Update observations

Description

reobserve (resp. recheck, reensure) re-reruns the function observe_if (resp. check_that, ensure_that) on .data with respect to the predicates contained in its observations attribute created at a previous occasion.

Usage

reobserve(.data, status = c("failed", "passed"))
recheck(.data, status = c("failed", "passed"))
reensure(.data, status = c("failed", "passed"))

Arguments

.data A tbl or data frame.
status character. One of "failed", "passed", or both. The observations attribute of .data is filtered according to the value of status.

Value

.data is returned, with its observations attribute updated.

See Also

observe_if, check_that, and ensure_that in this package.
**View_obs**  

View observations

---

**Description**

Invoke data viewer on observations(x).

**Usage**

`view_obs(x, title)`

**Arguments**

- **x**: A data frame, possibly with an `observations` attribute.
- **title**: Title for viewer window.

**See Also**

`View` from package `utils`.
Index

as.bit, 4

check (check_that), 2
check_that, 2, 2, 6
check_that_ (check_that), 2

ensure (check_that), 2
ensure_that, 2, 6
ensure_that (check_that), 2
ensure_that_ (check_that), 2

inspect, 3

obs (observations), 4
obs<- (observations), 4
observations, 4, 4
observations<- (observations), 4
observe (observe_if), 5
observe_if, 2–5, 5, 6
observe_if_, 4
observe_if_ (observe_if), 5

recheck (reobserve), 6
reensure (reobserve), 6
reobserve, 6

tibble, 5

View, 7
View_obs, 7