Package: sticky (via r-universe)

October 12, 2024

Type Package

Title Persist Attributes Across Data Operations

Description In base R, object attributes are lost when objects are modified by common data operations such as subset, filter, slice, append, extract etc. This packages allows objects to be marked as 'sticky' and have attributes persisted during these operations or when inserted into or extracted from list-like or table-like objects.

Version 0.5.6.1

Date 2020-03-15

License GPL-2 | file LICENSE

URL https://github.com/decisionpatterns/sticky
http://decisionpatterns.com

BugReports https://github.com/decisionpatterns/sticky/issues

Depends R (>= 3.1.0)

Imports methods

Suggests magrittr (>= 1.5), testthat (>= 0.10.0), data.table (>= 1.9.6), tibble (>= 1.3.3), knitr, rmarkdown

LazyData TRUE

Collate 'init.R' 'sticky.R' 'append.R' 'extract1.R' 'print.R' 'subset.R' 'utils.R' 'zzz.R'

VignetteBuilder knitr

RoxygenNote 7.0.2

Encoding UTF-8

Language en-US

Repository https://decisionpatterns.r-universe.dev

RemoteUrl https://github.com/decisionpatterns/sticky

RemoteRef HEAD

RemoteSha 524334461eba65e699bd706fe4d8422dac9731ec

Contents

`.init` *Internal, non-exported unit functions*

Description

These are internal functions for the sticky package. They are not exported and should not be used directly. Instead use the proper interface, `sticky` and `unstick`

Usage

```
.init(x)

.deinit(x)
```

Arguments

x	vector

Details

`.init` : initializes the class `.deinit` : uninitialize the class

These are internals functions not to be called directly, please use the interfaces, `sticky` and `unsticky`.

See Also

`sticky` and `unsticky`

Examples

```
# - tk
```

append *Append elements to sticky object*

Description

Append elements to sticky object

Usage

```
append(x, ...)
```

Arguments

x	sticky object
...	additional arguments to base::append append preserves object attributes by wrapping base::append and copying objects to the result.

See Also

append

print.sticky *print a sticky attribute object*

Description

print a sticky object which mostly delegates it the the next method.

Usage

```
## S3 method for class 'sticky'
print(x, ...)
```

Arguments

x	sticky object
...	further arguments passed to or from other methods.

Details

prints a sticky object; this is also the default method for sticky objects

Value

An invisible copy of x.

Examples

```
x <- sticky(1:3)
print(x)
x
```

`sticky` *sticky*

Description

Persist Attributes

Usage

```
sticky(x)

unstick(x)

unsticky(x)

is.sticky(x)

sticky_all(x)
```

Arguments

x	objects to have attributes preserved.

Details

`sticky` creates objects whose attributes are resilient to subsetting and placing in containers such as lists, data.frames and data.tables.

References

This has been asked and written about numerous times on Stack Overflow. The first has a fairly comprehensive list to others SO discussions.

http://stackoverflow.com/questions/23991060/loss-of-attributes-despite-attempts-to-preserve-them
http://stackoverflow.com/questions/10404224/how-to-delete-a-row-from-a-data-frame-without-losing-t
http://stackoverflow.com/questions/23991060/loss-of-attributes-despite-attempts-to-preserve-them
http://stackoverflow.com/questions/23841387/approaches-to-preserving-objects-attributes-during-extr
http://stackoverflow.com/questions/13432519/indexing-operation-removes-attributes

Examples

```
x <- 1:5
attr(x, 'foo') <- 'bar'
x <- x[1:3]
attr(x,'foo')            # NULL

x <- sticky(1:5)
attr(x, 'foo') <- 'bar'
x_1 <- x[1:3]
attr(x_1, 'foo')         # bar

x_2 <- unstick(x)
x_2 <- x[1:3]
attr(x_2,'foo')          # NULL


is.sticky(x)             # TRUE
```

`subset.sticky` *Subset elements to sticky object*

Description

Subset elements to sticky object

Usage

```
## S3 method for class 'sticky'
subset(x, ...)
```

Arguments

`x`	sticky object
`...`	additional arguments to `base::subset`
	`subset` preserves object attributes by wrapping `base::subset` and copying objects to the result.

See Also

`subset`

`[.sticky` *Extract*

Description

Extract

Usage

```
## S3 method for class 'sticky'
x[...]
```

Arguments

`x`	sticky; object to extract from
`...`	additional parameters

Details

Implements '[' for sticky attributes. This methods ensures that existing attributes are propagated forward.

See Also

`attributes`, specifically `mostattributes`

Index