

Package ‘dataview’

May 22, 2015

Encoding UTF-8

Type Package

Title Data and Workspace Browser for Terminals

Version 2.1.1

Date 2015-05-21

Author Christofer Backlin [aut, cre]

Maintainer Christofer Backlin <christofer.backlin@medsci.uu.se>

Imports data.table, xtermStyle (>= 3.0.5)

Description Tools for deciphering the contents of unknown objects or environments from within the terminal, a problem often encountered when working with unfamiliar packages or debugging complex functions. If working in xterm256 or ANSI terminals the output is coloured by default to improve readability (e.g. the standard Ubuntu terminal).

URL <http://github.com/backlin/dataview>

BugReports <https://github.com/backlin/dataview/issues>

License GPL (>= 2)

LazyLoad yes

NeedsCompilation no

Repository CRAN

Date/Publication 2015-05-22 08:54:36

R topics documented:

as.whos	2
browse	2
entry.view	3
getOpt	4
heat.view	5
whos	6
whos.options	7

Index**9**

as.whos	<i>Convert objects to whos</i>
---------	--------------------------------

Description

Convert objects to whos

Usage

```
as.whos(x)

## S3 method for class 'data.table'
as.whos(x)

## S3 method for class 'data.frame'
as.whos(x)
```

Arguments

x Object of type `data.table` or `data.frame`.

Author(s)

Christofer Bäcklin

Examples

```
an.object <- "Containing all my stuff"
w <- as.data.frame(whos())
as.whos(w)
```

browse	<i>Browse the contents of a nested data structure</i>
--------	---

Description

Manually step in and out of the elements of complex data structure with `whos`. To move around enter the name or number of the element you want to inspect next. Partial names will automatically be matched against the possible element names.

Usage

```
browse(x = .GlobalEnv, name)
```

Arguments

x	Object.
name	Name of the object.

Value

Nothing.

Author(s)

Christofer Bäcklin

See Also

whos

Examples

```
## Not run:  
require(ggplot2)  
p <- ggplot(iris, aes(x=Sepal.Length, y=Sepal.Width)) +  
  geom_point()  
browse(p)  
  
## End(Not run)
```

entry.view

Display vectors, lists or rows of a data frames in key-value-pairs.

Description

Color coded according to class of contents.

Usage

```
entry.view(x, i = 1)
```

Arguments

x	List or data frame.
i	Row number to show. Press down/up to browse.

Value

Nothing.

Author(s)

Christofer Bäcklin

Examples

```
entry.view(Sys.getenv())
```

getOpt

Retrieves package options

Description

The dataview package contains a number of option for tailoring its behaviour. These options are stored as a named list in a single global variable named "dataview". To overwrite an option fetch the default values with `default.options`, modify the returned results and set it back as in the example below.

Usage

```
getOpt(x)
```

```
default.options()
```

Arguments

x Option to retrieve.

Details

Below is a description of the available options, but to understand how they work it is probably easier to directly study the return value of `default.options`.

`align` Column alignment, left or right.

`columns` A named list with columns to use in `whos`. The `object` element should be a named list of functions to be applied on each object that is to be queried. The `envir` element should be a named list of functions to be applied on the object or environment on which `whos` is called together with its "accessors", which are object names (for environments) or named indices (for everything else). All functions should return a vector of values that can be used as a `data.table` column.

`print` A named list of custom print functions. By default each column of a `whos` object is printed in a similar way as `data.table` or `data.frame`. If you wish to override this behaviour for a given column, please supply a named print function here. The function will be given the column as returned by the corresponding column function above, and should produce a character vector.

`summary` A list of named summary functions. These are fed a column and should return a single value.

Author(s)

Christofer Bäcklin

Examples

```
# This shows how to modify a column presented by whos.  
# The new function only reports the size of non-S4 objects  
# to improve execution time.  
opt <- default.options()  
opt$columns$bytes <- function(x) if(isS4(x)) NA else object.size(x)  
options(dataview = opt)
```

heat.view

Display heatmaps and heatvectors.

Description

Quickly see the overall pattern of a variable in the terminal.

Usage

```
heat.view(x, pal, range, mark = NULL)
```

Arguments

x	Vector to be displayed.
pal	Palette. Either the name of a palette defined in xterm.pal or an integer vector with color indices (see display.xterm.colors).
range	The numerical range which the palette describes. See discrete.color for more info.
mark	Single letter marks to be displayed on top of the color.

Value

Nothing

Author(s)

Christofer Bäcklin

Examples

```

data(iris)
heat.view(iris$Species)
heat.view(matrix(iris$Petal.Width, 3, 50, byrow=TRUE,
                dimnames=list(levels(iris$Species), NULL)), pal="purples")

run.status <- factor(runif(100) < .95, labels=c("Fail", "Pass"))
heat.view(run.status, pal=1:2)

#Tip for displaying the element names of a named vector:
a <- runif(7)
names(a) <- c("ATM", "CHK1", "CDC25", "p53", "CDC2", "CDK2", "CDK4")
heat.view(a) # No names displayed
heat.view(as.matrix(a)) # Names displayed

```

whos	<i>Display contents of an environment, data.frame or list as a summary table</i>
------	--

Description

Color coded according to class and dimensions of contents. See [style](#) for details.

Shortcut for calling whos without exclusion.

Usage

```

whos(envir = parent.frame(), pattern = ".", all.names,
     exclude = getOpt("exclude"))

whos.all(...)

```

Arguments

envir	Environment, data frame or list to be displayed. Optional, default: globalenv()
pattern	Regex filtering of objects. Only objects matching the pattern are displayed. Optional, default: show all objects.
all.names	Whether to show hidden objects.
exclude	A list of objects not to be displayed. To set a default exclusion mask use the whos.set.mask function. If whos.set.mask is called without a list of object names all objects currently in globalenv() are hidden. This is useful for example if you have a lot of stuff in the workspace that you aren't currently interested in but is needed to make your code run.
...	Parameters sent to whos .

Value

Nothing
Nothing

Author(s)

Christofer Bäcklin
Christofer Bäcklin

See Also

whos.options, browse

Examples

```
whos()
data(USArrests)
whos(USArrests)

data(iris)
whos()
whos.all()
```

whos.options

Set default behavior of the whos function

Description

Set default behavior of the whos function

Usage

```
whos.options(exclude, report.S4.size)

whos.exclude(x = NULL, pattern, envir = parent.frame())

whos.include(x = NULL, pattern, envir = parent.frame())
```

Arguments

exclude	Objects to exclude from view. Can be a character vector of names, or an environment, but not any regular expressions so far.
report.S4.size	Calculating the size of S4 objects with <code>object.size</code> can take an annoyingly long time (seconds), set this option to FALSE to skip it and get quicker execution.
x	A character vector of object names to exclude or include.
pattern	Regular expression pattern to match object names against, e.g. <code>pattern="^my\\..*"</code> will exclude or include "my.vector" and "my.matrix" but not "mysql.con".
envir	Environment to search in.

Value

Nothing. The values are stored as global options.

Author(s)

Christofer Bäcklin

Index

`as.whos`, 2

`browse`, 2

`data.frame`, 2, 4

`data.table`, 2, 4

`default.options (getOpt)`, 4

`discrete.color`, 5

`display.xterm.colors`, 5

`entry.view`, 3

`getOpt`, 4

`heat.view`, 5

`object.size`, 7

Regular expression pattern, 7

`style`, 6

`whos`, 2, 4, 6, 6

`whos.exclude (whos.options)`, 7

`whos.include (whos.options)`, 7

`whos.options`, 7

`xterm.pal`, 5