

Package ‘io’

February 20, 2015

Type Package

Title A Unified Framework for Input-Output Operations in R

Version 0.2.2

Date 2014-12-11

Author David J. H. Shih

Maintainer David J. H. Shih <djh.shih@gmail.com>

Description One function to read files. One function to write files. One function to direct plots to screen or file. Automatic file format inference and directory structure creation.

Imports stringr

Depends fileamer

Suggests XML (>= 3.98-1.1), rhdf5 (>= 2.10.0), yaml (>= 2.1.13), jsonlite (>= 0.9.14), testthat

URL <https://bitbucket.org/djhshih/io>

BugReports <https://bitbucket.org/djhshih/io/issues>

License GPL (>= 3)

NeedsCompilation no

Repository CRAN

Date/Publication 2015-01-08 12:20:16

R topics documented:

io	2
io_supported	2
list_files	3
qdraw	3
qread	5
qwrite	6
Index	7

io	<i>A unified framework for input-output operations in R</i>
----	---

Description

io provides [qread](#) for reading in data of various types and [qwrite](#) for writing data to files of various types. Input or output file types can be inferred from filename extensions or specified explicitly.

Details

Use `link{io_supported}` to check whether a data or file type is supported.

Both [qread](#) and [qwrite](#) can be readily extended to support additional types by defining specific S3 methods.

Additionally, [qdraw](#) offers a unified interface for plotting to screen or various file formats.

io_supported	<i>Determine input-output support for data or file type</i>
--------------	---

Description

This function returns whether a type is supported by [qread](#) or [qwrite](#).

Usage

```
io_supported(type)
```

Arguments

type	data or file type
------	-------------------

Value

a `data.frame` with logical entries; TRUE if type is supported, FALSE otherwise

Examples

```
io_supported("rds")
```

list_files	<i>List the files in a directory.</i>
------------	---------------------------------------

Description

This function extends `list.files` by excluding the listing of directories.

Usage

```
list_files(path = ".", full.names = FALSE, ...)
```

Arguments

path	a character vector of path names
full.names	whether to return absolute paths
...	other arguments passed to <code>list.files</code>

Value

a character vector of only names of files

Examples

```
list.files(R.home())  
list_files(R.home())
```

qdraw	<i>Draw plot</i>
-------	------------------

Description

This function draws a plot to screen, a file, or both.

Usage

```
qdraw(expr, file = NULL, device = getOption("plot.device"), width = NULL,  
height = NULL, aspect.ratio = NULL, units = NULL, res = NULL,  
mkpath = TRUE, symlink = TRUE, ...)
```

Arguments

<code>expr</code>	expression for plotting
<code>file</code>	filename
<code>device</code>	plot device
<code>width</code>	plot width [default: 5]
<code>height</code>	plot height [default: 5]
<code>aspect.ratio</code>	ratio of width to height
<code>units</code>	unit of plot dimension [default: "in"]
<code>res</code>	bitmap resolution, used only by bitmap formats [default: 300]
<code>mkpath</code>	whether to create parent directories (if they do not already exists)
<code>symlink</code>	whether to create a symlink to file with a simplified filename (ignored if file is not a filename object); an existing file will not be overwritten but an existing symlink will be
<code>...</code>	other arguments passed to the plot device function

Details

To send the plot to screen, set `device` to `NA` (default). Optionally, to print the plot on screen to a file, specify `file`.

If `device` is `NULL`, the plot will be sent directly to the the specified file using a printing device inferred from the file extension (no graphical window will open).

Set the global option `plot.device` to affect multiple plots. Graphical parameters including `width`, `height`, `res`, `units` are obtained from the global option `getOption("plot")`.

Examples

```
## Not run:
# Set device to jpeg (remember to update file extensions for printed plots)
options(plot.device=jpeg)
qdraw(plot(1:10), "plot.jpeg")

# Enable automatic plot format inference
options(plot.device=NULL)

# Plot directly to file (format is inferred from filename extension)
qdraw(plot(1:10), "plot.pdf")

# Plot to screen, then print to file (display will not be closed)
qdraw(plot(1:10), "plot.png", device=NA)

# If an error occurs, be sure to clear the current plot
dev.off()
# or clear all plots
graphics.off()

## End(Not run)
```

qread	<i>Data input</i>
-------	-------------------

Description

This function reads a file in a specified format.

Usage

```
qread(file, type = NULL, ...)
```

Arguments

file	file name (character or <code>filenamer::filename</code>), a readable text-mode connection (for some types), or path to existing directory
type	data or file type
...	other arguments passed to the underlying function

Details

If type is NULL, the file type is inferred from the file extension. Use [io_supported](#) to check support for a file or data type.

Value

a data object (type depends on the underlying function)

Examples

```
## Not run:
data(cars)

# write data to an RDS file
qwrite(cars, "cars.rds")
# infer output type based on the class of the cars object
qwrite(cars, "cars.dfm", type=NA)

# read data back in
x1 <- qread("cars.rds")
# specify the type explicitly
x3 <- qread("cars.dfm", type="data.frame")

# read all files (with extension) in current directory
xs <- qread(".", pattern="cars")

## End(Not run)
```

qwrite

Data output

Description

This function writes an object to file in a specified format.

Usage

```
qwrite(x, file, type = NULL, mkpath = TRUE, symlink = TRUE, ...)
```

Arguments

x	data object to write
file	filename (character or <code>filename::filename</code>), a readable text-mode connection (for some types), or path to existing directory
type	data or file type
mkpath	whether to create parent directories (if they do not already exist)
symlink	whether to create a symlink to file with a simplified file name (ignored if file is not a filename object); an existing file will not be overwritten but an existing symlink will be
...	other arguments passed to the underlying function

Details

If type is NULL, the file type is inferred from the file extension. If type is NA or if the file extension is unavailable or unknown, type is inferred from `class(x)`. Use [io_supported](#) to check support for a file or data type.

Value

a data object (object type depends on the underlying function)

Examples

```
## Not run:
data(cars)

# write data to a TSV file
qwrite(cars, "cars.tsv")
# infer output type based on the class of the cars object
qwrite(as.matrix(cars), "cars.mtx", type=NA)

## End(Not run)
```

Index

io, [2](#)
io-package (io), [2](#)
io_supported, [2](#), [5](#), [6](#)

list.files, [3](#)
list_files, [3](#)

qdraw, [2](#), [3](#)
qread, [2](#), [5](#)
qwrite, [2](#), [6](#)