Package 'dataframes2xls'

September 26, 2016

Type Package

Title Write Data Frames to Xls Files

Depends R (>= 2.0.0)

SystemRequirements python (>= 2.4)

Version 0.4.7

Date 2016-09-17

Author Guido van Steen

Maintainer Guido van Steen <vansteen@users.sourceforge.net>

Description Writes data frames to xls files. It supports
multiple sheets and basic formatting.

License BSD_3_clause + file LICENSE

Repository CRAN

Date/Publication 2016-09-26 00:59:22

NeedsCompilation no

R topics documented:

---

write.xls   *WRITE XLS DATA FILES*

Description

Write data frames to an xls file

1

Usage

```
write.xls(x, file, sh.names = "default", formats = "default",
t.formats = FALSE, fnt.names = "Helvetica", fnt.metr = "default",
col.widths = 48, col.names = TRUE, row.names = FALSE,
to.floats = "default", python = "python",
py.script = "default", sh.return = FALSE)
```

Arguments

x	the object(s) to be written, preferably data frame(s). If not, `write.xls` will try to coerce them to data framedata(s).
file	a character string naming the xls file that will contain the data frame(s).
sh.names	name(s) of the sheet(s) of the xls files to be written. If not specified "Sheet1,Sheet2,..." will be used as sheet names.
formats	format string(s) indicating the formats of the colums (or the rows) of each sheet. See the examples.
t.formats	logical value(s) indicating that the format string(s) are to be applied per row (and not per column).
fnt.names	the font name(s) to be used in the sheet(s) of the xls file.
fnt.metr	the name and location of the "font metric file(s)" that will be used in the xls file.
col.widths	the column width(s) in the sheet(s) in pixels.
row.names	logical values indicating whether the row names of (each element of) x are to be written along with (each element of) x. Note that this argument of the function `write.xls` must not be used to set row names. For this purpose the function `row.names` should be used instead.
col.names	logical value(s) indicating whether the column names of (each element of) x are to be written along with (each element of) x. Note that this argument of the function `write.xls` must not be used to set column names. For this purpose the function `col.names` should be used instead.
to.floats	categorical value(s) indicating whether the cells are to be written as numerical values or as character strings. `to.floats="default"` tries to write cells as numerical values. If this is not possible it will write cells as character strings instead. `to.floats="never"` writes all cells as character strings. `to.floats="always"` tries to write all cells as numerical values. If this is not possible it will write an "NA" character string.
python	name of the python interpreter to be called. Be sure to call a python interpreter. Otherwise thing may become unpredictable.
py.script	name of the python script to be called. By default it calls the python script that `dataframes2xls` installs.
sh.return	logical value indicating whether the return value of the system command that runs the python interprer is to be printed. The return value of the system command is either 0, which means that no errors were encountered, or 1, which means that an error did occurr.

Details

dataframes2xls saves dataframes to an xls file. Its main function write.xls, is a wrapper around a utility called xls2csv. The arguments available to this utility are also available to the R-script. xls2csv is written in python. Therefore python should be installed. The utility xls2csv is included in the dataframes2xls package. xls2csv makes use of the python module pyExcelerator and the afm submodule of the python module matplotlib. Both are now included in dataframes2xls See http://www.python.org, http://sourceforge.net/projects/pyexcelerator and http://sourceforge.net/projects/matplotlib respectively.

Value

NULL

Author(s)

Guido van Steen <vansteen@sourceforge.net>.

References

http://sourceforge.net/projects/py-csv2xls, http://sourceforge.net/projects/pyexcelerator

See Also

read.xls

Examples

```
# Create some simple data frames:

df1 = as.data.frame (c(1,2))
df1$col1 = (c(1,2))
df1$col2 = (c(3,4))
df1$col3 = (c(5,6))
df1[1] = NULL

df2 = as.data.frame (c(1,2))
df2$col1 = (c(10.10101010101,20))
df2$col2 = (c(30.25,40))
df2$col3 = (c(50E2,60))
df2[1] = NULL

# Create a format string:

fmts = "0:0::1:0%:::0:general::1:0.0"

# Note this format string might also have been written as:
#
# fmts = "0:0::1:0%,0:general::1:0.0"

# The first column (column number 0) of Sheet1 will have "0"
# as its format (i.e. it will presented as an integer).
```

```
#
# The second column (column number 1) of Sheet1 will have "0%"
# as its format (i.e. it will be presented as a percentage).

# The first column (column number 0) of Sheet2 will have
# "general" as its format (i.e. it will be presented as provided
# in the data frame).
#
# The second column (column number 1) of Sheet2 will have "0.0"
# as its format (i.e. it will be presented as a float with
# one digit).

# Let us specify some xls file names:

ofn1 = paste(tempdir(),"/tenure_rownames_false.xls",sep="")
ofn2 = paste(tempdir(),"/tenure_rownames_default.xls",sep="")
ofn3 = paste(tempdir(),"/tenure_rownames_true.xls",sep="")

# Now write the dataframes with the format string being applied
# per column:

write.xls(c(df1,df2), ofn1, formats=fmts)

# the data frames may also be written with the format string
# being applied per row:

t.fmts = "true:::true"

# Note that this t.formats string might also have been
# written as:
#
# t.fmts = "true,true"
#
# or:
#
# t.fmts = "true"
#
# or as:
#
# t.fmts = TRUE

write.xls(c(df1,df2), ofn2, formats=fmts, t.formats=t.fmts)

# Now we write the data frames with row names but without column
# names

rownames = c("first row","second row")

row.names(df1) = rownames

mat1 = as.matrix(df1)
mat2 = as.matrix(df2)
```

```
write.xls(c(mat1,mat2), ofn3, row.names="true", col.names="false")

# You may now inspect the xls files

keypressed = readline()

# until we clean them up:

system (paste("rm ",ofn1, sep = ""))
system (paste("rm ",ofn2, sep = ""))
system (paste("rm ",ofn3, sep = ""))
```

Index