

Package ‘DataEditR’

August 12, 2020

Title An Interactive Editor for Viewing, Entering & Editing Data

Version 0.0.5

Date 2020-07-30

Description An interactive editor built on 'rhandsontable' to allow the interactive viewing, entering and editing of data in R
<<https://dillonhammill.github.io/DataEditR/>>.

URL <https://github.com/DillonHammill/DataEditR>

BugReports <https://github.com/DillonHammill/DataEditR/issues>

Depends R(>= 3.5.0)

Imports shiny, shinythemes, rhandsontable, htmltools

License GPL-2

Encoding UTF-8

LazyData true

RoxygenNote 7.1.1

Language en-GB

Suggests knitr, rmarkdown

VignetteBuilder knitr

NeedsCompilation no

Author Dillon Hammill [aut, cre] (<<https://orcid.org/0000-0002-1407-7223>>)

Maintainer Dillon Hammill <Dillon.Hammill@anu.edu.au>

Repository CRAN

Date/Publication 2020-08-12 09:52:08 UTC

R topics documented:

data_edit	2
Index	5

data_edit

An interactive editor for viewing, entering & editing data

Description

data_edit is a shiny application built on rhandsontable that is designed to make it easy to interactively view, enter or edit data without any coding. data_edit is also a wrapper for any reading or writing function to make it easy to interactively update data saved to file.

Usage

```
data_edit(
  x,
  col_bind = NULL,
  col_edit = TRUE,
  col_options = NULL,
  col_stretch = FALSE,
  col_factor = FALSE,
  col_names = TRUE,
  col_readonly = NULL,
  row_bind = NULL,
  row_edit = TRUE,
  save_as = NULL,
  title = NULL,
  logo = NULL,
  logo_size = 100,
  viewer = TRUE,
  theme = "yeti",
  quiet = FALSE,
  read_fun = "read.csv",
  read_args = NULL,
  write_fun = "write.csv",
  write_args = NULL,
  ...
)
```

Arguments

x	a matrix, data.frame, data.table or the name of a csv file to edit. Tibble are also supported but will be coerced to data.frames. An empty table can be created by specifying the dimensions in a vector of the form c(nrow, ncol).
col_bind	additional columns to add to the data prior to loading into editor, can be either an array containing the new data, a vector containing the new column names for empty columns or a named list containing a vector for each new column.
col_edit	logical indicating whether columns can be added or removed, set to TRUE by default.

col_options	named list containing the options for columns that use dropdown menus or checkboxes.
col_stretch	logical indicating whether columns should be stretched to fill the full width of the display, set to FALSE by default.
col_factor	logical indicating whether character columns should be converted to factors prior to returning the edited data, set to FALSE by default.
col_names	logical indicating whether column names can be edited or a vector of column names that cannot be edited, set to TRUE by default to allow editing of column names.
col_readonly	names of columns that cannot be edited. Users will be able to edit values but these will be reverted to the original values. Column names for these column cannot be edited either.
row_bind	additional rows to add to the data prior to loading into editor, can be either an array containing the new data, a vector containing the new row names for empty rows or a named list containing a vector for each new column.
row_edit	logical indicating whether rows can be added or removed, set to TRUE by default.
save_as	name of a csv file to which the edited data should be saved.
title	optional title to include above the data editor.
logo	optional package logo to include in title above the data editor, must be supplied as path to logo png.
logo_size	width of the logo in pixels, set to 100 pixels by default.
viewer	logical indicating whether the data editor should be invoked in the RStudio viewer pane, set to TRUE by default.
theme	valid shinytheme name, set to "yeti" by default.
quiet	logical indicating whether messages should be suppressed, set to FALSE by default.
read_fun	name of the function to use to read in the data when x is the name of a file, set to read.csv by default.
read_args	a named list of additional arguments to pass to read_fun.
write_fun	name of the function to use to write the edited version of x to a file, set to write.csv by default. Only requirement is that the first argument accepts the edited data and the second argument accepts the file name supplied to save_as.
write_args	a named list of additional arguments to pass to write_fun.
...	additional arguments passed to both read_fun and write_fun, such as sep. Arguments that are shared by read_fun and write_fun can be supplied separately to these functions using the read_args and write_args arguments. For example, this becomes particularly important when specifying row.names arguments.

Value

edited matrix-like object.

Author(s)

Dillon Hammill, <Dillon.Hammill@anu.edu.au>

Examples

```
if (interactive()) {  
  # Edit matrix & save to csv  
  file <- tempfile("mtcars-new", fileext = ".csv")  
  data_edit(mtcars,  
    save_as = file  
  )  
  
  # Edit csv file  
  data_edit(file)  
}
```

Index

data_edit, [2](#)