Package 'SweaveLst'

June 6, 2020

Type Package

Title X

Version 1.0

Date 2020-06-06

Author D. Bonnery

Maintainer D. Bonnery <dbonnery@umd.edu>

Imports stargazer

Suggests

Description Data

Remotes yihui/tikzDevice

License GPL (>= 2)

LazyLoad yes

LazyData true

RoxygenNote 7.0.2

R topics documented:

`SweaveLst-package` *X*

Description

Data

Details

The DESCRIPTION file: This package was not yet installed at build time.

Index: This package was not yet installed at build time.

Author(s)

D. Bonnery

Maintainer: D. Bonnery <dbonnery@umd.edu>

`bleuJPSM` *bleuJPSM*

Usage

```
data("bleuJPSM")
```

Format

The format is: chr "#007CC4"

Examples

```
data(bleuJPSM)
## maybe str(bleuJPSM) ; plot(bleuJPSM) ...
```

bleuJPSM2 *bleuJPSM2*

Usage

```
data("bleuJPSM2")
```

Format

The format is: chr "#005380"

Examples

```
data(bleuJPSM2)
## maybe str(bleuJPSM2) ; plot(bleuJPSM2) ...
```

comp *comp*

Usage

```
comp()
```

Examples

```
##---- Should be DIRECTLY executable !! ----
##-- ==>  Define data, use random,
##--or do  help(data=index)  for the standard data sets.

## The function is currently defined as
function ()
{
    system("bash compile.sh")
  }
```

compR *compR*

Usage

```
compR(path = ".")
```

Arguments

path

Examples

```
##---- Should be DIRECTLY executable !! ----
##-- ==>  Define data, use random,
##--or do  help(data=index)  for the standard data sets.

## The function is currently defined as
function (path = ".")
{
    print(paste(list.dirs(path = path)))
    for (folder in list.dirs(full.names = TRUE)) {
        print(folder)
        for (file in list.files(path = folder, pattern = "*.rnw")) {
            print("##############################################")
            print(file)
            Sweavelst(file = file, path = folder)
        }
    }
  }
```

compT *compT*

Usage

```
compT(file, path = "./")
```

Arguments

file

path

Examples

```
##---- Should be DIRECTLY executable !! ----
##-- ==>  Define data, use random,
##--or do  help(data=index)  for the standard data sets.

## The function is currently defined as
function (file, path = "./")
{
    system(paste0("pdflatex -interaction=nonstopmode ", file.path(path,
        file)))
  }
```

---

compTR *compTR*

Usage

```
compTR()
```

Examples

```
##---- Should be DIRECTLY executable !! ----
##-- ==>  Define data, use random,
##--or do  help(data=index)  for the standard data sets.

## The function is currently defined as
function ()
{
    compR()
    system("bash compile.sh")
  }
```

---

couleur1 *couleur1*

Usage

```
data("couleur1")
```

Format

The format is: chr "#007CC4"

Examples

```
data(couleur1)
## maybe str(couleur1) ; plot(couleur1) ...
```

couleur2 *couleur2*

Usage

```
data("couleur2")
```

Format

The format is: chr "#005380"

Examples

```
data(couleur2)
## maybe str(couleur2) ; plot(couleur2) ...
```

couleurdebase *couleurdebase*

Usage

```
data("couleurdebase")
```

Format

The format is: chr "#007EC2"

Examples

```
data(couleurdebase)
## maybe str(couleurdebase) ; plot(couleurdebase) ...
```

cours *cours*

Usage

```
data("cours")
```

Format

The format is: chr "SURV_XXX_General_R_Exercises/"

Examples

```
data(cours)
## maybe str(cours) ; plot(cours) ...
```

dropboxfold *dropboxfold*

Usage

```
data("dropboxfold")
```

Format

The format is: chr "~/Dropbox"

Examples

```
data(dropboxfold)
## maybe str(dropboxfold) ; plot(dropboxfold) ...
```

graphtikzcode *graphtikzcode*

Usage

```
graphtikzcode(texte, widthe = 7, heighte = 7)
```

Arguments

texte

widthe

heighte

print_demo_file *Gives the tex code to print a demo code*

Description

Gives the tex code to print a demo code

Usage

```
print_demo_file(topic, package)
```

removetikzfile *removetikzfile*

Usage

```
removetikzfile(tmpfile)
```

Arguments

```
tmpfile
```

sanitizeLatexS *sanitizeLatexS*

Usage

```
sanitizeLatexS(str)
```

Arguments

```
str
```

Examples

```
##---- Should be DIRECTLY executable !! ----
##-- ==>  Define data, use random,
##--or do  help(data=index)  for the standard data sets.

## The function is currently defined as
function (str)
{
    cat(paste(gsub("([#$%&~_^\\{}])", "\\\\\1", str,
        perl = TRUE), collapse = "\n"))
  }
```

stargazer2 *stargazer2*

Usage

```
stargazer2(...)
```

Arguments

```
...
```

Examples

```
##---- Should be DIRECTLY executable !! ----
##-- ==>  Define data, use random,
##--or do  help(data=index)  for the standard data sets.

## The function is currently defined as
function (...)
{
    X = stargazer(..., header = FALSE, table.placement = "H")
    x <- gsub("\\", "\\\\", paste0(X[-1], collapse = " "))
    x <- gsub("\\textbackslash ", "\\", x)
    replacement <- paste(c("oooowoieghoihoihwoeghohoihwoeihgohsdfsdfse",
        sample(letters, 20, replace = TRUE)), collapse = "")
    x <- gsub("\\[$]", replacement, x)
    x <- gsub(paste0("\\", replacement), "$", x)
    replacement <- paste(c("oooowoieghoihoihwoeghohoihwoeihgohsdfsdfse",
        sample(letters, 20, replace = TRUE)), collapse = "")
    x <- gsub("\\[_]", replacement, x)
    x <- gsub(replacement, "_", x)
    replacement <- paste(c("oooowoieghoihoihwoeghohoihwoeihgohsdfsdfse",
        sample(letters, 20, replace = TRUE)), collapse = "")
    x <- gsub("\\[{]", replacement, x)
    x <- gsub(paste0("\\", replacement), "{", x)
    replacement <- paste(c("oooowoieghoihoihwoeghohoihwoeihgohsdfsdfse",
        sample(letters, 20, replace = TRUE)), collapse = "")
    x <- gsub("\\[}]", replacement, x)
    x <- gsub(paste0("\\", replacement), "}", x)
    x
  }
```

stargazerarray *stargazerarray*

Usage

```
stargazerarray(XX, ...)
```

Arguments

XX

...

Examples

```
##---- Should be DIRECTLY executable !! ----
##-- ==>  Define data, use random,
##--or do  help(data=index)  for the standard data sets.

## The function is currently defined as
function (XX, ...)
{
    tables <- c(apply(XX, 3:length(dim(XX)), function(X) {
        stargazer2(X, ...)
    }))
    titles <- ""
    for (i in 3:length(dim(XX))) {
        titles = outer(titles, dimnames(XX)[[i]], paste, sep = " ")
    }
    paste(paste0(c(titles), " \\\ ", tables), collapse = " \\\\ ")
  }
```

Sweavelst *Sweavelst command: alternative to Sinput and Soutput environments*

Usage

```
Sweavelst(file = NULL, path = getwd(), fullpath = NULL, out.width = 10, width = 50, height = 10, prompte =
```

Arguments

file

path

fullpath

out.width

width

height

prompte

Index