

# Package ‘rrtable’

February 4, 2020

**Type** Package

**Title** Reproducible Research with a Table of R Codes

**Version** 0.1.7

**Imports** stringr, ggplot2 (>= 2.2.0), officer (>= 0.3.5), purrr (>= 0.2.4), flextable (>= 0.4.4), rvg, magrittr, devEMF, moonBook (>= 0.1.8), rmarkdown, shiny, editData, readr (>= 1.1.1), ztable (>= 0.1.8)

**Description** Makes documents containing plots and tables from a table of R codes.  
Can make ``HTML``, ``pdf('LaTex)'``, ``docx('MS Word)'`` and ``pptx('MS Powerpoint)'`` documents with or without R code.  
In the package, modularized 'shiny' app codes are provided. These modules are intended for reuse across applications.

**Depends** R(>= 2.10)

**License** GPL-3

**Encoding** UTF-8

**LazyData** true

**RoxygenNote** 7.0.2

**VignetteBuilder** knitr

**Suggests** knitr

**NeedsCompilation** no

**Author** Keon-Woong Moon [aut, cre]

**Maintainer** Keon-Woong Moon <cardiomoon@gmail.com>

**Repository** CRAN

**Date/Publication** 2020-02-04 15:20:06 UTC

## R topics documented:

add_2flextables . . . . .	3
add_2ggplots . . . . .	4
add_2plots . . . . .	5

add_anyplot . . . . .	6
add_flextable . . . . .	7
add_ggplot . . . . .	8
add_img . . . . .	8
add_plot . . . . .	9
add_Rcode . . . . .	10
add_self . . . . .	11
add_text . . . . .	11
add_text2hyperlink . . . . .	12
add_title . . . . .	12
add_title_slide . . . . .	13
data2docx . . . . .	13
data2HTML . . . . .	14
data2office . . . . .	15
data2pdf . . . . .	16
data2plotzip . . . . .	17
data2pptx . . . . .	18
df2flextable . . . . .	18
df2flextable2 . . . . .	20
df2RcodeTable . . . . .	20
exportCSV . . . . .	21
file2docx . . . . .	21
file2HTML . . . . .	22
file2pdf . . . . .	22
file2plotzip . . . . .	23
file2pptx . . . . .	23
flextable2ztable . . . . .	24
html2latex . . . . .	24
HTMLcode2latex . . . . .	25
insert_argument . . . . .	25
is_ggplot . . . . .	26
mycat . . . . .	26
mygrep . . . . .	27
myplot2 . . . . .	27
mytable2flextable . . . . .	28
open_doc . . . . .	29
plot2docx . . . . .	29
plot2office . . . . .	30
plot2pptx . . . . .	31
plotPDF2 . . . . .	31
plotPNG2 . . . . .	32
plotSVG2 . . . . .	33
pptxList . . . . .	33
pptxListInput . . . . .	34
Rcode2df . . . . .	35
Rcode2df2 . . . . .	35
Rcode2docx . . . . .	36
Rcode2flextable . . . . .	36

Rcode2office . . . . .	37
Rcode2pptx . . . . .	37
readComment . . . . .	38
readCSVComment . . . . .	38
replace_argument . . . . .	39
roundDf . . . . .	39
sampleData2 . . . . .	40
sampleData3 . . . . .	40
set_argument . . . . .	41
table2docx . . . . .	41
table2office . . . . .	42
table2pptx . . . . .	43
tensiSplit . . . . .	43
writeCSVComment . . . . .	44
ztable2 . . . . .	44
ztable2flextable . . . . .	45

<b>Index</b>	<b>46</b>
--------------	-----------

---

add_2flextables	<i>Add two flextables into a document object</i>
-----------------	--------------------------------------------------

---

## Description

Add two flextables into a document object

## Usage

```
add_2flextables(mydoc, ft1, ft2, echo = FALSE, width = 3, code = "")
```

## Arguments

mydoc	A document object
ft1	The first flextable
ft2	The second flextable
echo	whether or not display R code
width	plot width in inches
code	R code string

## Value

a document object

**Examples**

```

require(rrtable)
require(officer)
require(magrittr)
title="Two Tables"
ft1=df2flextable(head(iris[1:4]))
ft2=df2flextable(tail(iris[1:4]))
doc=read_docx()
doc %>% add_text(title=title) %>%
  add_2flextables(ft1,ft2)
doc=read_pptx()
doc %>% add_text(title=title) %>%
  add_2flextables(ft1,ft2)

```

---

add\_2ggplots

*Add two ggplots into a document object*


---

**Description**

Add two ggplots into a document object

**Usage**

```

add_2ggplots(
  mydoc,
  plot1,
  plot2,
  preprocessing = "",
  width = 3,
  height = 2.5,
  top = 2
)

```

**Arguments**

mydoc	A document object
plot1	An R code encoding the first ggplot
plot2	An R code encoding the second ggplot
preprocessing	preprocessing
width	plot width in inches
height	plot height in inches
top	top plot position in inches

**Value**

a document object

**Examples**

```
require(ggplot2)
require(magrittr)
require(officer)
require(rvg)
plot1 <- "ggplot(data = iris, aes(Sepal.Length, Petal.Length)) + geom_point()"
plot2 <- "ggplot(data = iris, aes(Sepal.Length, Petal.Length, color = Species)) + geom_point()"
read_pptx() %>% add_text(title="Two ggplots") %>% add_2ggplots(plot1=plot1,plot2=plot2)
read_docx() %>% add_text(title="Two ggplots") %>% add_2ggplots(plot1=plot1,plot2=plot2)
```

---

add\_2plots

*Add two plots into a document object*

---

**Description**

Add two plots into a document object

**Usage**

```
add_2plots(
  mydoc,
  plotstring1,
  plotstring2,
  preprocessing = "",
  plottype = "auto",
  width = NULL,
  height = NULL,
  echo = FALSE,
  top = 2
)
```

**Arguments**

mydoc	A document object
plotstring1	An R code string encoding the first plot
plotstring2	An R code string encoding the second plot
preprocessing	preprocessing
plottype	character One of c("auto","plot","ggplot")
width	plot width in inches
height	plot height in inches
echo	logical Whether or not show R code
top	top plot position in inches

**Value**

a document object

**Examples**

```
require(magrittr)
require(officer)
require(ggplot2)
plotstring1="plot(iris)"
plotstring2="ggplot(iris,aes(x=Sepal.Length,y=Sepal.Width))+geom_point()"
read_pptx() %>% add_text(title="Two plots") %>% add_2plots(plotstring1,plotstring2)
read_docx() %>% add_text(title="Two plots") %>% add_2plots(plotstring1,plotstring2)
```

---

add\_anyplot

*Add a ggplot or a plot to the Microsoft Office Document*

---

**Description**

Add a ggplot or a plot to the Microsoft Office Document

**Usage**

```
add_anyplot(
  doc,
  x = NULL,
  preprocessing = "",
  plottype = "auto",
  left = 1,
  top = 1,
  width = 8,
  height = 5.5
)
```

**Arguments**

doc	A document object
x	An object of class ggplot2 or a string encoding plot or ggplot
preprocessing	A string of R code
plottype	character One of c("auto","plot","ggplot")
left	left margin
top	top margin
width	desired width of the plot
height	desired height of the plot

---

add_flextable	<i>Add a flextable or mytable object into a document object</i>
---------------	-----------------------------------------------------------------

---

## Description

Add a flextable or mytable object into a document object

## Usage

```
add_flextable(mydoc, ftable, echo = FALSE, code = "", landscape = FALSE)
```

## Arguments

mydoc	A document object
ftable	A flextable or mytable object
echo	whether or not display R code
code	R code string
landscape	Logical. Whether or not make a landscape section.

## Value

a document object

## Examples

```
require(rrtable)
require(moonBook)
require(officer)
require(magrittr)
ftable=mytable(Dx~,data=acs)
title="mytable Example"
ft=df2flextable(head(iris))
title2="df2flextable Example"
doc=read_docx()
doc %>% add_text(title=title) %>%
  add_flextable(ftable) %>%
  add_text(title=title2) %>%
  add_flextable(ft)
```

add\_ggplot *Add ggplot into a document object*

---

**Description**

Add ggplot into a document object

**Usage**

```
add_ggplot(mydoc, code = "", preprocessing = "", top = 2)
```

**Arguments**

mydoc	A document object
code	R code for table
preprocessing	preprocessing
top	top position of plot

**Value**

a document object

**Examples**

```
require(rrtable)
require(ggplot2)
require(officer)
require(magrittr)
code <- "ggplot(mtcars, aes(x = mpg , y = wt)) + geom_point()"
read_pptx() %>% add_text(title="ggplot") %>% add_ggplot(code=code)
read_docx() %>% add_text(title="ggplot") %>% add_ggplot(code=code)
```

---

add\_img *Add plot into a document object*

---

**Description**

Add plot into a document object



**Usage**

```
add_img(
  mydoc,
  plotstring,
  width = 7,
  height = 5,
  units = "in",
  res = 300,
  format = "emf",
  ...
)
```

**Arguments**

mydoc	A document object
plotstring	An string of R code encoding plot
width	the width of the device.
height	the height of the device.
units	The units in which height and width are given. Can be px (pixels, the default), in (inches), cm or mm.
res	The nominal resolution in ppi which will be recorded in the bitmap file, if a positive integer. Also used for units other than the default, and to convert points to pixels.
format	plot format
...	additional arguments passed to png()

**Value**

a document object

**Examples**

```
require(officer)
require(rrtable)
require(magrittr)
require(flextable)
read_pptx() %>% add_text(title="Add image") %>% add_img("plot(mtcars)", format="png", res=300)
```

---

add\_plot

*Add plot into a document object*

---

**Description**

Add plot into a document object

**Usage**

```
add_plot(mydoc, plotstring, preprocessing = "", width = 6, height = 6, top = 2)
```

**Arguments**

mydoc	A document object
plotstring	String of an R code encoding a plot
preprocessing	preprocessing
width	width of plot
height	height of plot
top	top position of plot

**Value**

a document object

**Examples**

```
require(rrtable)
require(officer)
require(rvg)
require(magrittr)
read_pptx() %>% add_text(title="Plot") %>% add_plot("plot(iris)")
read_docx() %>% add_text(title="Plot") %>% add_plot("plot(iris)")
```

---

add\_Rcode

*Make a R code slide into a document object*

---

**Description**

Make a R code slide into a document object

**Usage**

```
add_Rcode(mydoc, code, preprocessing = "", format = "pptx")
```

**Arguments**

mydoc	A document object
code	A character string encoding R codes
preprocessing	A character string of R code as a preprocessing
format	desired format. choices are "pptx" or "docx"

**Value**

a document object

**Examples**

```
library(rrtable)
library(magrittr)
library(officer)
code="summary(lm(mpg~hp+wt,data=mtcars))"
read_pptx() %>% add_text(title="Regression Analysis") %>%
  add_Rcode(code)
```

---

add_self	<i>add self data to document</i>
----------	----------------------------------

---

**Description**

add self data to document

**Usage**

```
add_self(mydoc, data)
```

**Arguments**

mydoc	A document object
data	a data.frame

---

add_text	<i>Add text to document</i>
----------	-----------------------------

---

**Description**

Add text to document

**Usage**

```
add_text(
  mydoc,
  title = "",
  text = "",
  code = "",
  preprocessing = "",
  echo = FALSE,
  eval = FALSE,
  style = "Normal",
  landscape = FALSE
)
```

**Arguments**

mydoc	A document object
title	An character string as a plot title
text	text string to be added
code	An R code string
preprocessing	preprocessing
echo	logical Whether or not show R code
eval	logical whether or not evaluate the R code
style	text style
landscape	Logical. Whether or not make a landscape section.

---

add\_text2hyperlink     *Add hyperlink text*

---

**Description**

Add hyperlink text

**Usage**

```
add_text2hyperlink(mydoc, text)
```

**Arguments**

mydoc	A document object
text	text string to be added

---

add\_title                 *Add title to docx file*

---

**Description**

Add title to docx file

**Usage**

```
add_title(x, title = "", size = 20, color = NULL, before = TRUE, after = TRUE)
```

**Arguments**

x	A document object
title	Title
size	font size
color	font color
before	Whether or not add blank paragraph before title
after	Whether or not add blank paragraph after title

---

add_title_slide	<i>Add title slide</i>
-----------------	------------------------

---

**Description**

Add title slide

**Usage**

```
add_title_slide(mydoc, title = "", subtitle = "")
```

**Arguments**

mydoc	A document object
title	An character string as a title
subtitle	An character string as a subtitle

**Examples**

```
require(magrittr)
require(officer)
read_pptx() %>% add_title_slide(title="Web-based analysis with R")
```

---

data2docx	<i>convert data to docx file</i>
-----------	----------------------------------

---

**Description**

convert data to docx file

**Usage**

```
data2docx(...)
```

**Arguments**

... arguments to be passed to data2office()

**Examples**

```
library(rrtable)
library(moonBook)
library(ggplot2)
data2docx(sampleData2,echo=TRUE)
```

---

data2HTML

*Make a HTML5 file with a data.frame*

---

**Description**

Make a HTML5 file with a data.frame

**Usage**

```
data2HTML(
  data,
  preprocessing = "",
  path = ".",
  filename = "report.HTML",
  rawDataName = NULL,
  rawDataFile = "rawData.RDS",
  type = "HTML",
  vanilla = FALSE,
  echo = TRUE,
  showself = FALSE
)
```

**Arguments**

data	A data.frame
preprocessing	A character string of R code
path	A name of destination file path
filename	A name of destination file
rawDataName	The name of the rawData
rawDataFile	The name of the rawData file which the data are to be read from.
type	character "HTML" or "pdf"
vanilla	logical. Whether or not make vanilla table
echo	Logical. Whether or not show R code of plot and table
showself	Logical. Whether or not show R code for the paragraph

## Examples

```
library(moonBook)
library(rtable)
library(ggplot2)
data2HTML(sampleData2)
```

---

data2office	<i>convert data to pptx file</i>
-------------	----------------------------------

---

## Description

convert data to pptx file

## Usage

```
data2office(
  data,
  preprocessing = "",
  path = ".",
  filename = "Report",
  format = "pptx",
  width = 7,
  height = 5,
  units = "in",
  res = 300,
  rawDataName = NULL,
  rawDataFile = "rawData.RDS",
  vanilla = FALSE,
  echo = FALSE,
  landscape = FALSE,
  showself = FALSE
)
```

## Arguments

data	A document object
preprocessing	A string
path	A name of destination file path
filename	File name
format	desired format. choices are "pptx" or "docx"
width	the width of the device.
height	the height of the device.

units	The units in which height and width are given. Can be px (pixels, the default), in (inches), cm or mm.
res	The nominal resolution in ppi which will be recorded in the bitmap file, if a positive integer. Also used for units other than the default, and to convert points to pixels.
rawDataName	raw Data Name
rawDataFile	raw Data File
vanilla	logical. Whether or not make vanilla table
echo	logical Whether or not show R code
landscape	Logical. Whether or not make a landscape section.
showself	Logical. Whether or not show R code for the paragraph

---

data2pdf

*Make a pdf file with a data.frame*

---

## Description

Make a pdf file with a data.frame

## Usage

```
data2pdf(...)
```

## Arguments

... further arguments to be passed to data2HTML

## Examples

```
library(moonBook)
library(ztable)
library(ggplot2)

data2pdf(sampleData2)
```



---

data2plotzip	<i>Make zipped plot file with a data.frame</i>
--------------	------------------------------------------------

---

## Description

Make zipped plot file with a data.frame

## Usage

```
data2plotzip(  
  data,  
  path = ".",  
  filename = "Plot.zip",  
  format = "PNG",  
  width = 8,  
  height = 6,  
  units = "in",  
  res = 300,  
  start = 0,  
  preprocessing = "",  
  rawDataName = NULL,  
  rawDataFile = "rawData.RDS"  
)
```

## Arguments

data	A data.frame
path	A name of destination file path
filename	A path of destination file
format	Plot format. Choices are c("PNG","SVG","PDF")
width	A plot width
height	A plot height
units	The units in which height and width are given. Can be px (pixels, the default), in (inches), cm or mm.
res	The nominal resolution in ppi
start	Plot start number
preprocessing	A character string of R code
rawDataName	The name of the rawData
rawDataFile	The name of the rawData file which the data are to be read from.

**Examples**

```
library(moonBook)
library(ztable)
library(rrtable)
library(ggplot2)
data2plotzip(sampleData2, path="tmp")
```

---

data2pptx	<i>convert data to pptx file</i>
-----------	----------------------------------

---

**Description**

convert data to pptx file

**Usage**

```
data2pptx(...)
```

**Arguments**

... arguments to be passed to data2office()

**Examples**

```
library(rrtable)
library(moonBook)
library(ggplot2)
data2pptx(sampleData2, echo=TRUE)
```

---

df2flextable	<i>Convert data.frame to flextable</i>
--------------	----------------------------------------

---

**Description**

Convert data.frame to flextable

**Usage**

```
df2flextable(
  df,
  vanilla = FALSE,
  fontname = NULL,
  fontsize = 12,
  add.rownames = FALSE,
  even_header = "transparent",
  odd_header = "#5B7778",
  even_body = "#EFEFEF",
  odd_body = "transparent",
  vlines = TRUE,
  colorheader = FALSE,
  digits = 2,
  align_header = "center",
  align_body = "right",
  NA2space = TRUE,
  pcol = NULL,
  ...
)
```

**Arguments**

df	A data.frame
vanilla	A Logical
fontname	Font name
fontsize	font size
add.rownames	logical. Whether or not include rownames
even_header	background color of even_header
odd_header	background color of even_header
even_body	background color of even_body
odd_body	background color of even_body
vlines	Logical. Whether or not draw vertical lines
colorheader	Logical. Whether or not use color in header
digits	integer indicating the number of decimal places
align_header	alignment of header. Expected value is one of 'left', 'right', 'center', 'justify'.
align_body	alignment of body. Expected value is one of 'left', 'right', 'center', 'justify'.
NA2space	A logical. If true, convert NA value to space
pcol	An integer indicating p value. If specified, convert value less than 0.01 to "< 0.001" in given column.
...	further arguments to be passed to flextable

**Examples**

```
require(flextable)
require(officer)
df2flextable(head(iris),vanilla=TRUE,colorheader=TRUE)

df2flextable(head(iris),vanilla=TRUE,digits=c(1,2,3,4))
df2flextable(head(iris),vanilla=FALSE)
df2flextable(head(iris),vanilla=FALSE,vlines=FALSE,fontsize=14)
df2flextable(head(mtcars))
```

---

df2flextable2	<i>Make flextable with limited width</i>
---------------	------------------------------------------

---

**Description**

Make flextable with limited width

**Usage**

```
df2flextable2(df, mincol = 0.7, maxcol = 4, ...)
```

**Arguments**

df	a data.frame
mincol	minimum column width in inch
maxcol	maximum column width in inch
...	further arguments to be passed to df2flextable()

---

df2RcodeTable	<i>Make a flextable with a data.frame</i>
---------------	-------------------------------------------

---

**Description**

Make a flextable with a data.frame

**Usage**

```
df2RcodeTable(df, bordercolor = "gray", format = "pptx", eval = TRUE)
```

**Arguments**

df	A data.frame
bordercolor	A border color name
format	desired format. choices are "pptx" or "docx"
eval	logical. Whether or not evaluate the code

**Value**

A flextable object

---

exportCSV	<i>Export pptxList file to desired format</i>
-----------	-----------------------------------------------

---

**Description**

Export pptxList file to desired format

**Usage**

```
exportCSV(
  file,
  format = "HTML",
  rawDataName = NULL,
  rawDataFile = "rawData.RDS"
)
```

**Arguments**

file	The name of the file which the data are to be read from.
format	desired output format. Possible choices are one of the c("HTML","pdf","word","pptx","plotzip")
rawDataName	The name of the rawData
rawDataFile	The name of the rawData file which the data are to be read from.

---

file2docx	<i>read data file and make a docx file</i>
-----------	--------------------------------------------

---

**Description**

read data file and make a docx file

**Usage**

```
file2docx(file, selected = NULL, ...)
```

**Arguments**

file	The name of the file which the data are to be read from.
selected	A numeric vector or NULL(default). If specified, only selected data are printed.
...	Further argument to be passed to data2docx()

---

file2HTML	<i>read data file and make a HTML file</i>
-----------	--------------------------------------------

---

**Description**

read data file and make a HTML file

**Usage**

```
file2HTML(file, selected = NULL, ...)
```

**Arguments**

file	The name of the file which the data are to be read from.
selected	A numeric vector or NULL(default). If specified, only selected data are printed.
...	Further argument to be passed to data2HTML()

---

file2pdf	<i>read data file and make a pdf file</i>
----------	-------------------------------------------

---

**Description**

read data file and make a pdf file

**Usage**

```
file2pdf(file, selected = NULL, ...)
```

**Arguments**

file	The name of the file which the data are to be read from.
selected	A numeric vector or NULL(default). If specified, only selected data are printed.
...	Further argument to be passed to data2pdf()

---

file2plotzip	<i>read data file and make a zip file with plots</i>
--------------	------------------------------------------------------

---

**Description**

read data file and make a zip file with plots

**Usage**

```
file2plotzip(file, selected = NULL, ...)
```

**Arguments**

file	The name of the file which the data are to be read from.
selected	A numeric vector or NULL(default). If specified, only selected data are printed.
...	Further argument to be passed to data2plotzip()

---

file2pptx	<i>read data file and make a pptx file</i>
-----------	--------------------------------------------

---

**Description**

read data file and make a pptx file

**Usage**

```
file2pptx(file, selected = NULL, ...)
```

**Arguments**

file	The name of the file which the data are to be read from.
selected	A numeric vector or NULL(default). If specified, only selected data are printed.
...	Further argument to be passed to data2pptx()

flextable2ztable      *Convert flextable to ztable*

---

**Description**

Convert flextable to ztable

**Usage**

```
flextable2ztable(ft, type = "html", ...)
```

**Arguments**

ft	An object of class flextable
type	"html" or "latex"
...	Further argument to be passed to ztable

**Value**

an object of class ztable

---

html2latex      *Convert html5 code to latex*

---

**Description**

Convert html5 code to latex

**Usage**

```
html2latex(df)
```

**Arguments**

df	A data.frame
----	--------------



---

HTMLcode2latex	<i>Convert HTML table to latex table</i>
----------------	------------------------------------------

---

**Description**

Convert HTML table to latex table

**Usage**

```
HTMLcode2latex(data)
```

**Arguments**

data	a data.frame
------	--------------

---

insert_argument	<i>replace argument of a function</i>
-----------------	---------------------------------------

---

**Description**

replace argument of a function

**Usage**

```
insert_argument(code, argument, value)
```

**Arguments**

code	string of function call
argument	argument of function to be set
value	value to be set

---

is_ggplot	<i>Reports whether plotstring encode a ggplot object</i>
-----------	----------------------------------------------------------

---

**Description**

Reports whether plotstring encode a ggplot object

**Usage**

```
is_ggplot(plotstring)
```

**Arguments**

plotstring      A character

**Examples**

```
require(ggplot2)
is_ggplot("plot(iris)")
is_ggplot("ggplot(iris,aes(x=Sepal.Length))+geom_histogram()")
```

---

mycat	<i>Concatenate to file</i>
-------	----------------------------

---

**Description**

Concatenate to file

**Usage**

```
mycat(..., file = "report2.Rmd")
```

**Arguments**

...              R object  
file             A connection

---

mygrep	<i>grep string in all files in subdirectory</i>
--------	-------------------------------------------------

---

**Description**

grep string in all files in subdirectory

**Usage**

```
mygrep(x, file = "*")
```

**Arguments**

x	string
file	files to seek

---

myplot2	<i>Make zipped plots with a data.frame</i>
---------	--------------------------------------------

---

**Description**

Make zipped plots with a data.frame

**Usage**

```
myplot2(
  data,
  format = "PNG",
  width = 7,
  height = 7,
  units = "in",
  res = 300,
  start = 0,
  preprocessing = "",
  rawDataName = NULL,
  rawDataFile = "rawData.RDS"
)
```

**Arguments**

data	A data.frame
format	Plot format. Choices are c("PNG","SVG","PDF")
width	A plot width
height	A plot height

units	The units in which height and width are given. Can be px (pixels, the default), in (inches), cm or mm.
res	The nominal resolution in ppi
start	Plot start number
preprocessing	A character string of R code
rawDataName	The name of the rawData
rawDataFile	The name of the rawData file which the data are to be read from.

---

mytable2flectable	<i>Convert mytable object to flectable</i>
-------------------	--------------------------------------------

---

### Description

Convert mytable object to flectable

### Usage

```
mytable2flectable(result, vanilla = TRUE, fontname = NULL, fontsize = 10)
```

### Arguments

result	An object of class "mytable"
vanilla	A Logical.
fontname	Font name
fontsize	font size

### Examples

```
require(moonBook)
require(flectable)
require(officer)
result=mytable(smoking+Dx~.,data=acs)
mytable2flectable(result)
mytable2flectable(result,vanilla=FALSE)
result=mytable(Dx~.,data=acs)
mytable2flectable(result)
mytable2flectable(result,vanilla=FALSE)
```

---

open_doc	<i>Make/open office document with file name</i>
----------	-------------------------------------------------

---

**Description**

Make/open office document with file name

**Usage**

```
open_doc(target = "Report", type = "pptx", append = FALSE)
```

**Arguments**

target	name of output file
type	"pptx" or "docx"
append	logical

---

plot2docx	<i>Save plot/ggplot to Microsoft Word format</i>
-----------	--------------------------------------------------

---

**Description**

Save plot/ggplot to Microsoft Word format

**Usage**

```
plot2docx(...)
```

**Arguments**

...	further arguments to be passed to plot2office
-----	-----------------------------------------------

**Examples**

```
require(ggplot2)
x<-ggplot(iris,aes(x=Sepal.Length))+geom_histogram()
plot2docx(x)
plot2docx(x,title="A ggplot",append=TRUE)
p2=ggplot(iris,aes(x=Sepal.Length,y=Sepal.Width))+geom_point()
plot2docx(p2,append=TRUE)
plot2docx(x="plot(iris)",append=TRUE,title="plot(iris)")
plot2docx(x="ggplot(iris,aes(x=Sepal.Length))+geom_histogram()",append=TRUE)
```

---

plot2office

*Save plot/ggplot to Microsoft Powerpoint format*


---

## Description

Save plot/ggplot to Microsoft Powerpoint format

## Usage

```
plot2office(
  x = NULL,
  target = "Report",
  append = FALSE,
  title = "",
  type = "pptx",
  preprocessing = "",
  plottype = "auto",
  echo = FALSE,
  parallel = FALSE,
  left = 1,
  top = 1,
  width = NULL,
  height = NULL,
  aspectr = NULL
)
```

## Arguments

x	An object of class ggplot2 or a string vector encoding plot or ggplot
target	name of output file
append	logical value
title	Optional character vector of plot title
type	"pptx" or "docx"
preprocessing	A string of R code or ""
plottype	character One of c("auto","plot","ggplot")
echo	logical. If true, show code.
parallel	logical. If true, add two plots side by side
left	left margin
top	top margin
width	desired width of the plot
height	desired height of the plot
aspectr	desired aspect ratio of the plot

**Examples**

```
x=c("plot(iris)", "ggplot(mtcars, aes(x=hp, y=mpg))+geom_point()")
plot2office(x, title="2 plots", parallel=TRUE)
plot2office(x, title="2 plots", parallel=TRUE, echo=TRUE, append=TRUE)
plot2office(x, parallel=TRUE, echo=TRUE, append=TRUE)
```

---

plot2pptx

*Save plot/ggplot to Microsoft Powerpoint format*


---

**Description**

Save plot/ggplot to Microsoft Powerpoint format

**Usage**

```
plot2pptx(...)
```

**Arguments**

... further arguments to be passed to plot2office

**Examples**

```
require(ggplot2)
x<-ggplot(iris, aes(x=Sepal.Length))+geom_histogram()
plot2pptx(x)
plot2pptx(x, title="A ggplot", append=TRUE)
p2=ggplot(iris, aes(x=Sepal.Length, y=Sepal.Width))+geom_point()
plot2pptx(p2, append=TRUE)
plot2pptx(x=c("plot(iris)", "ggplot(iris, aes(x=Sepal.Length))+geom_histogram()"),
  append=TRUE, title=c("plot", "ggplot"), echo=TRUE)
```

---

plotPDF2

*Make pdf file with a plot code*


---

**Description**

Make pdf file with a plot code

**Usage**

```
plotPDF2(  
  fun,  
  file,  
  width = 7,  
  height = 5,  
  units = "in",  
  res = 300,  
  ggplot = FALSE  
)
```

**Arguments**

fun	A R code for plot
file	A path of destination file
width	A plot width
height	A plot height
units	The units in which height and width are given. Can be px (pixels, the default), in (inches), cm or mm.
res	The nominal resolution in ppi
ggplot	A logical. Set this argument true if the R code is for ggplot

---

plotPNG2

*Make png file with a plot code*

---

**Description**

Make png file with a plot code

**Usage**

```
plotPNG2(  
  fun,  
  file,  
  width = 7,  
  height = 7,  
  units = "in",  
  res = 300,  
  ggplot = FALSE  
)
```



**Arguments**

fun	A R code for plot
file	A path of destination file
width	A plot width
height	A plot height
units	The units in which height and width are given. Can be px (pixels, the default), in (inches), cm or mm.
res	The nominal resolution in ppi
ggplot	A logical. Set this argument true if the R code is for ggplot

---

plotSVG2                      *Make SVG file with a plot code*

---

**Description**

Make SVG file with a plot code

**Usage**

```
plotSVG2(fun, file, width = 7, height = 7, ggplot = FALSE)
```

**Arguments**

fun	A R code for plot
file	A path of destination file
width	A plot width
height	A plot height
ggplot	A logical. Set this argument true if the R code is for ggplot

---

pptxList                      *Server function of pptxList shiny module*

---

**Description**

Server function of pptxList shiny module

**Usage**

```
pptxList(
  input,
  output,
  session,
  data = reactive(""),
  preprocessing = reactive("")
)
```

**Arguments**

input	input
output	output
session	session
data	A data object
preprocessing	A character string of R code

---

pptxListInput      *UI of pptxList shiny module*

---

**Description**

UI of pptxList shiny module

**Usage**

```
pptxListInput(id)
```

**Arguments**

id	A string
----	----------

**Examples**

```
library(shiny)
library(ggplot2)
library(editData)
library(moonBook)
library(readr)
if(interactive()){
  ui=fluidPage(
    pptxListInput("pptxlist")
  )
  server=function(input,output,session){
    mydf=callModule(pptxList,"pptxlist")
  }
  shinyApp(ui,server)
}
```

---

`Rcode2df`*Make a data.frame with character strings encoding R code*

---

**Description**

Make a data.frame with character strings encoding R code

**Usage**

```
Rcode2df(result, preprocessing, eval = TRUE)
```

**Arguments**

<code>result</code>	character strings encoding R code
<code>preprocessing</code>	character strings encoding R code as a preprocessing
<code>eval</code>	logical. Whether or not evaluate the code

---

`Rcode2df2`*Make a data.frame with character strings encoding R code*

---

**Description**

Make a data.frame with character strings encoding R code

**Usage**

```
Rcode2df2(result, preprocessing, eval = TRUE)
```

**Arguments**

<code>result</code>	character strings encoding R code
<code>preprocessing</code>	character strings encoding R code as a preprocessing
<code>eval</code>	logical. Whether or not evaluate the code

---

Rcode2docx	<i>Save R code to Microsoft Word format</i>
------------	---------------------------------------------

---

**Description**

Save R code to Microsoft Word format

**Usage**

```
Rcode2docx(...)
```

**Arguments**

... further arguments to be passed to plot2office

**Examples**

```
code="summary(lm(mpg~hp+wt,data=mtcars))"
Rcode2docx(code=code,title="R code to Word")
```

---

Rcode2flextable	<i>Make a flextable object with character strings encoding R code</i>
-----------------	-----------------------------------------------------------------------

---

**Description**

Make a flextable object with character strings encoding R code

**Usage**

```
Rcode2flextable(result, preprocessing = "", format = "pptx", eval = TRUE)
```

**Arguments**

result	character strings encoding R code
preprocessing	character strings encoding R code as a preprocessing
format	desired format. choices are "pptx" or "docx"
eval	logical. Whether or not evaluate the code

**Examples**

```
Rcode2flextable("str(mtcars)\nsummary(mtcars)",eval=FALSE)
```

---

Rcode2office

*Make R code slide*

---

### **Description**

Make R code slide

### **Usage**

```
Rcode2office(  
  code,  
  preprocessing = "",  
  title = "",  
  type = "pptx",  
  target = "Report",  
  append = FALSE  
)
```

### **Arguments**

code	A character string encoding R codes
preprocessing	A character string of R code as a preprocessing
title	A character
type	desired format. choices are "pptx" or "docx"
target	name of output file
append	logical

### **Examples**

```
code="summary(lm(mpg~hp+wt,data=mtcars))"  
Rcode2office(code=code)
```

---

Rcode2pptx

*Save R code to Microsoft Powerpoint format*

---

### **Description**

Save R code to Microsoft Powerpoint format

### **Usage**

```
Rcode2pptx(...)
```

**Arguments**

... further arguments to be passed to plot2office

**Examples**

```
code="summary(lm(mpg~hp+wt,data=mtcars))"
Rcode2pptx(code=code,title="R code to pptx")
```

---

readComment	<i>Read comment from a file</i>
-------------	---------------------------------

---

**Description**

Read comment from a file

**Usage**

```
readComment(filename, comment = "#")
```

**Arguments**

filename	A path for destination file
comment	A string used to identify comments

---

readCSVComment	<i>Read a csv file with comment</i>
----------------	-------------------------------------

---

**Description**

Read a csv file with comment

**Usage**

```
readCSVComment(file)
```

**Arguments**

file	A path for destination file
------	-----------------------------

---

replace_argument	<i>replace argument of a function</i>
------------------	---------------------------------------

---

**Description**

replace argument of a function

**Usage**

```
replace_argument(substring, argument, value)
```

**Arguments**

substring	string of function call
argument	argument of function to be set
value	value to be set

---

roundDf	<i>Convert numeric columns of data.frame to character</i>
---------	-----------------------------------------------------------

---

**Description**

Convert numeric columns of data.frame to character

**Usage**

```
roundDf(df, digits = 2)
```

**Arguments**

df	A data.frame
digits	integer indicating the number of decimal places

**Examples**

```
roundDf(iris,digits=c(1,2,3,4))  
roundDf(mtcars,digits=2)
```

---

sampleData2	<i>Sample data for pptxList A dataset containing five objects for reproducible research</i>
-------------	---------------------------------------------------------------------------------------------

---

**Description**

Sample data for pptxList A dataset containing five objects for reproducible research

**Usage**

```
sampleData2
```

**Format**

A data frame with 5 rows and three columns

**type** type of data

**title** title of data

**code** R code of data

---

sampleData3	<i>Sample data for pptxList A dataset containing five objects for reproducible research</i>
-------------	---------------------------------------------------------------------------------------------

---

**Description**

Sample data for pptxList A dataset containing five objects for reproducible research

**Usage**

```
sampleData3
```

**Format**

A data frame with 5 rows and three columns

**type** type of data

**title** title of data

**text** text

**code** R code of data

**option** option for R code



---

set_argument	<i>set argument of a function</i>
--------------	-----------------------------------

---

**Description**

set argument of a function

**Usage**

```
set_argument(code, argument, value = TRUE)
```

**Arguments**

code	string of function call
argument	argument of function to be set
value	value to be set

**Examples**

```
code="df2flextable( ) "  
code="df2flextable(vanilla=TRUE,head(iris[1:10,]))"  
code="df2flextable(mtcars)"  
code="df2flextable(sampleData3)"  
code="df2flextable(head(iris[1:10,]),vanilla=TRUE)"  
set_argument(code,"vanilla",FALSE)
```

---

table2docx	<i>Export data.frame or statistical output to Microsoft Word format</i>
------------	-------------------------------------------------------------------------

---

**Description**

Export data.frame or statistical output to Microsoft Word format

**Usage**

```
table2docx(...)
```

**Arguments**

... further arguments to be passed to table2office

**Examples**

```
require(moonBook)
x=mytable(Dx~.,data=acs)
table2docx(x)
table2docx(head(iris),title="head(iris)",append=TRUE,vanilla=FALSE)
fit=lm(mpg~wt*hp,data=mtcars)
table2docx(fit,title="Linear regression",append=TRUE,vanilla=TRUE)
fit2=aov(yield ~ block + N * P + K, data = npk)
table2docx(fit2,title="Linear regression",append=TRUE,vanilla=TRUE)
```

---

table2office

*Export data.frame or statistical output to a table in Microsoft Office*


---

**Description**

Export data.frame or statistical output to a table in Microsoft Office

**Usage**

```
table2office(
  x = NULL,
  target = "Report",
  append = FALSE,
  title = "",
  vanilla = FALSE,
  echo = FALSE,
  preprocessing = "",
  type = "pptx",
  landscape = FALSE,
  left = 1,
  top = 1
)
```

**Arguments**

x	An object or string
target	name of output file
append	logical value
title	Optional character of plot title
vanilla	A logical
echo	logical
preprocessing	A character string
type	"pptx" or "docx"

landscape	logical
left	left margin
top	top margin

---

table2pptx	<i>Export data.frame or statistical output to Microsoft Powerpoint format</i>
------------	-------------------------------------------------------------------------------

---

### Description

Export data.frame or statistical output to Microsoft Powerpoint format

### Usage

```
table2pptx(...)
```

### Arguments

... further arguments to be passed to table2office

### Examples

```
require(moonBook)
x="mytable(Dx~.,data=acs)"
table2pptx(x,title="mytable object",echo=TRUE)
table2pptx("head(iris)",title="data.Frame",append=TRUE,vanilla=FALSE,echo=TRUE)
x="fit<-lm(mpg~wt*hp,data=mtcars);fit"
table2pptx(x,title="Linear regression",append=TRUE,vanilla=TRUE,echo=TRUE)
fit2="aov(yield ~ block + N * P + K, data = npk)"
table2pptx(fit2,title="ANOVA",append=TRUE,vanilla=TRUE,echo=TRUE)
```

---

tensiSplit	<i>Split strings with desired length with exdent</i>
------------	------------------------------------------------------

---

### Description

Split strings with desired length with exdent

### Usage

```
tensiSplit(string, size = 82, exdent = 3)
```

**Arguments**

string	String
size	desired length
exdent	exdent

**Value**

splitted character vector

---

writeCSVComment	<i>Write a csv file with comment</i>
-----------------	--------------------------------------

---

**Description**

Write a csv file with comment

**Usage**

```
writeCSVComment(data, file, metadata = "", comment = "#")
```

**Arguments**

data	A data.frame
file	A path for destination file
metadata	A character string representing R codes as a preprocessing
comment	A string used to identify comments

---

ztable2	<i>Make ztable with desired width</i>
---------	---------------------------------------

---

**Description**

Make ztable with desired width

**Usage**

```
ztable2(df, cwidth = NULL, width = 80, ...)
```

**Arguments**

df	a data.frame
cwidth	desired column width
width	desired table width in column
...	further argument to be passed to ztable()

---

ztable2flextable	<i>Convert ztable to flextable</i>
------------------	------------------------------------

---

**Description**

Convert ztable to flextable

**Usage**

```
ztable2flextable(z, ...)
```

**Arguments**

<code>z</code>	An object of class <code>ztable</code>
<code>...</code>	Further argument to be passed to <code>df2flextable</code>

**Value**

an object of class `flextable`

# Index

## \*Topic **datasets**

- sampleData2, 40
- sampleData3, 40
  
- add\_2flextables, 3
- add\_2ggplots, 4
- add\_2plots, 5
- add\_anypplot, 6
- add\_flextable, 7
- add\_ggplot, 8
- add\_img, 8
- add\_plot, 9
- add\_Rcode, 10
- add\_self, 11
- add\_text, 11
- add\_text2hyperlink, 12
- add\_title, 12
- add\_title\_slide, 13
  
- data2docx, 13
- data2HTML, 14
- data2office, 15
- data2pdf, 16
- data2plotzip, 17
- data2pptx, 18
- df2flextable, 18
- df2flextable2, 20
- df2RcodeTable, 20
  
- exportCSV, 21
  
- file2docx, 21
- file2HTML, 22
- file2pdf, 22
- file2plotzip, 23
- file2pptx, 23
- flextable2ztable, 24
  
- html2latex, 24
- HTMLcode2latex, 25
  
- insert\_argument, 25
- is\_ggplot, 26
  
- mycat, 26
- mygrep, 27
- myplot2, 27
- mytable2flextable, 28
  
- open\_doc, 29
  
- plot2docx, 29
- plot2office, 30
- plot2pptx, 31
- plotPDF2, 31
- plotPNG2, 32
- plotSVG2, 33
- pptxList, 33
- pptxListInput, 34
  
- Rcode2df, 35
- Rcode2df2, 35
- Rcode2docx, 36
- Rcode2flextable, 36
- Rcode2office, 37
- Rcode2pptx, 37
- readComment, 38
- readCSVComment, 38
- replace\_argument, 39
- roundDf, 39
  
- sampleData2, 40
- sampleData3, 40
- set\_argument, 41
  
- table2docx, 41
- table2office, 42
- table2pptx, 43
- tensiSplit, 43
  
- writeCSVComment, 44
  
- ztable2, 44
- ztable2flextable, 45