Package 'xtable'

July 17, 2025

Version 1.8-4

Date 2019-04-08

Title Export Tables to LaTeX or HTML

Maintainer David Scott <d.scott@auckland.ac.nz>

Imports stats, utils

Suggests knitr, plm, zoo, survival

VignetteBuilder knitr

Description Coerce data to LaTeX and HTML tables.

URL <http://xtable.r-forge.r-project.org/>

Depends R (>= 2.10.0)

License GPL (>= 2)

Repository CRAN

NeedsCompilation no

Author David B. Dahl [aut],
David Scott [aut, cre],
Charles Roosen [aut],
Arni Magnusson [aut],
Jonathan Swinton [aut],
Ajay Shah [ctb],
Arne Henningsen [ctb],
Benno Puetz [ctb],
Bernhard Pfaff [ctb],
Claudio Agostinelli [ctb],
Claudius Loehnert [ctb],
David Mitchell [ctb],
David Whiting [ctb],
Fernando da Rosa [ctb],
Guido Gay [ctb],
Guido Schulz [ctb],
Ian Fellows [ctb],
Jeff Laake [ctb],
John Walker [ctb],

Jun Yan [ctb],
Liviu Andronic [ctb],
Markus Loecher [ctb],
Martin Gubri [ctb],
Matthieu Stigler [ctb],
Robert Castelo [ctb],
Seth Falcon [ctb],
Stefan Edwards [ctb],
Sven Garbade [ctb],
Uwe Ligges [ctb]

Date/Publication 2019-04-21 12:20:03 UTC

Contents

---

autoformat   *Automatically Format Export Tables*

Description

Suggest an appropriate alignment, number of digits, and display type for `xtable`.

Usage

```
autoformat(xtab, zap = getOption("digits"))

xalign(x, pad = TRUE)
xdigits(x, pad = TRUE, zap = getOption("digits"))
xdisplay(x, pad = TRUE)
```

Arguments

xtab	an object of class xtable.
x	a vector, matrix, or data frame.
pad	whether to format row names, when x is two-dimensional.
zap	the number of digits passed to zapsmall.

Value

autoformat returns a copy of xtab, after applying xalign, xdigits, and xdisplay.

xalign returns a character vector consisting of "l" and "r" elements, for left/right alignment.

xdigits returns an integer vector.

xdisplay returns a character vector of "d", "f", and "s" elements, for integer/double/string display.

Author(s)

Arni Magnusson.

See Also

[xtable](#), [align](#), [digits](#), [display](#)

Examples

```
## 1  Vector
xalign(precip)
xdigits(precip)
xdisplay(precip)


## 2  Data frame
head(mtcars)
xdigits(mtcars, pad = FALSE)
xdigits(mtcars, pad = TRUE)
xalign(mtcars)
xdisplay(mtcars)


## 3  Autoformat when xtable is created
xtable(mtcars, align = xalign(mtcars), digits = xdigits(mtcars),
       display = xdisplay(mtcars))

## equivalent shortcut
xtable(mtcars, auto = TRUE)


## 4  Autoformat existing xtable
mt <- xtable(mtcars)
align(mt) <- xalign(mt)
digits(mt) <- xdigits(mt)
```

```
display(mt) <- xdisplay(mt)

## equivalent shortcut
mt <- autoformat(mt)
```

---

print.xtable   *Print Export Tables*

Description

Function returning and displaying or writing to disk the LaTeX or HTML code associated with the
supplied object of class xtable.

Usage

```
## S3 method for class 'xtable'
print(x,
  type = getOption("xtable.type", "latex"),
  file = getOption("xtable.file", ""),
  append = getOption("xtable.append", FALSE),
  floating = getOption("xtable.floating", TRUE),
  floating.environment = getOption("xtable.floating.environment", "table"),
  table.placement = getOption("xtable.table.placement", "ht"),
  caption.placement = getOption("xtable.caption.placement", "bottom"),
  caption.width = getOption("xtable.caption.width", NULL),
  latex.environments = getOption("xtable.latex.environments", c("center")),
  tabular.environment = getOption("xtable.tabular.environment", "tabular"),
  size = getOption("xtable.size", NULL),
  hline.after = getOption("xtable.hline.after", c(-1,0,nrow(x))),
  NA.string = getOption("xtable.NA.string", ""),
  include.rownames = getOption("xtable.include.rownames", TRUE),
  include.colnames = getOption("xtable.include.colnames", TRUE),
  only.contents = getOption("xtable.only.contents", FALSE),
  add.to.row = getOption("xtable.add.to.row", NULL),
  sanitize.text.function = getOption("xtable.sanitize.text.function", NULL),
  sanitize.rownames.function = getOption("xtable.sanitize.rownames.function",
                                         sanitize.text.function),
  sanitize.colnames.function = getOption("xtable.sanitize.colnames.function",
                                         sanitize.text.function),
  math.style.negative = getOption("xtable.math.style.negative", FALSE),
  math.style.exponents = getOption("xtable.math.style.exponents", FALSE),
  html.table.attributes = getOption("xtable.html.table.attributes",
                                    "border=1"),
  print.results = getOption("xtable.print.results", TRUE),
  format.args = getOption("xtable.format.args", NULL),
  rotate.rownames = getOption("xtable.rotate.rownames", FALSE),
  rotate.colnames = getOption("xtable.rotate.colnames", FALSE),
```

```
    booktabs = getOption("xtable.booktabs", FALSE),
    scalebox = getOption("xtable.scalebox", NULL),
    width = getOption("xtable.width", NULL),
    comment = getOption("xtable.comment", TRUE),
    timestamp = getOption("xtable.timestamp", date()),
    ...)
```

Arguments

x	An object of class "xtable".
type	Type of table to produce. Possible values for type are "latex" or "html". Default value is "latex".
file	Name of file where the resulting code should be saved. If file="", output is displayed on screen. Note that the function also (invisibly) returns a character vector of the results (which can be helpful for post-processing). Default value is "".
append	If TRUE and file!="", code will be appended to file instead of overwriting file. Default value is FALSE.
floating	If TRUE and type="latex", the resulting table will be a floating table (using, for example, \begin{table} and \end{table}). See floating.environment below. Default value is TRUE.
floating.environment	If floating=TRUE and type="latex", the resulting table uses the specified floating environment. Possible values include "table", "table*", and other floating environments defined in LaTeX packages. Default value is "table".
table.placement	If floating=TRUE and type="latex", the floating table will have placement given by table.placement where table.placement must be NULL or contain only elements of {"h","t","b","p","!","H"}. Default value is "ht".
caption.placement	The caption will be placed at the bottom of the table if caption.placement is "bottom" and at the top of the table if it equals "top". Default value is "bottom".
caption.width	The caption will be placed in a "parbox" of the specified width if caption.width is not NULL and type="latex". Default value is NULL.
latex.environments	If floating=TRUE and type="latex", the specified LaTeX environments (provided as a character vector) will enclose the tabular environment. Default value is "center".
tabular.environment	When type="latex", the tabular environment that will be used. When working with tables that extend more than one page, using tabular.environment="longtable" with the corresponding LaTeX package (see Fairbairns, 2005) allows one to typeset them uniformly. Note that floating should be set to FALSE when using the longtable environment. Default value is "tabular".

size	A character vector that is inserted just before the tabular environment starts. This can be used to set the font size and a variety of other table settings. Initial backslashes are automatically prefixed, if not supplied by user. Default value is `NULL`.
hline.after	When `type="latex"`, a vector of numbers between -1 and `nrow(x)`, inclusive, indicating the rows after which a horizontal line should appear. If `NULL` is used no lines are produced. Repeated values are allowed. Default value is `c(-1,0,nrow(x))` which means draw a line before and after the columns names and at the end of the table.
NA.string	String to be used for missing values in table entries. Default value is `""`.
include.rownames	
	If `TRUE` the rows names are printed. Default value is `TRUE`.
include.colnames	
	If `TRUE` the columns names are printed. Default value is `TRUE`.
only.contents	If `TRUE` only the rows of the table are printed. Default value is `FALSE`.
add.to.row	A list of two components. The first component (which should be called 'pos') is a list that contains the position of rows on which extra commands should be added at the end. The second component (which should be called 'command') is a character vector of the same length as the first component, which contains the command that should be added at the end of the specified rows. Default value is `NULL`, i.e. do not add commands.
sanitize.text.function	
	All non-numeric entries (except row and column names) are sanitized in an attempt to remove characters which have special meaning for the output format. If `sanitize.text.function` is not `NULL`, it should be a function taking a character vector and returning one, and will be used for the sanitization instead of the default internal function. Default value is `NULL`.
sanitize.rownames.function	
	Like the `sanitize.text.function`, but applicable to row names. The default uses the `sanitize.text.function`.
sanitize.colnames.function	
	Like the `sanitize.text.function`, but applicable to column names. The default uses the `sanitize.text.function`.
math.style.negative	
	In a LaTeX table, if `TRUE`, then use $-$ for the negative sign (as was the behavior prior to version 1.5-3). Default value is `FALSE`.
math.style.exponents	
	In a LaTeX table, if `TRUE` or `"$$"`, then use 5×10^{5} for 5e5. If `"ensuremath"`, then use `\ensuremath{5 \times 10^{5}}` for 5e5. If `"UTF-8"` or `"UTF-8"`, then use UTF-8 to approximate the LaTeX typesetting for 5e5. Default value is `FALSE`.
html.table.attributes	
	In an HTML table, attributes associated with the `<TABLE>` tag. Default value is `"border=1"`.
print.results	If `TRUE`, the generated table is printed to standard output. Set this to `FALSE` if you will just be using the character vector that is returned invisibly. Default value is `TRUE`.

format.args   List of arguments for the formatC function. For example, standard German number separators can be specified as format.args=list(big.mark = "'", decimal.mark = ",")). The arguments digits and format should not be included in this list. See details. Default value is NULL.

rotate.rownames
                If TRUE, the row names are displayed vertically in LaTeX. Default value is FALSE.

rotate.colnames
                If TRUE, the column names are displayed vertically in LaTeX. Default value is FALSE.

booktabs   If TRUE, the toprule, midrule and bottomrule commands from the LaTeX "booktabs" package are used rather than hline for the horizontal line tags.

scalebox   If not NULL, a scalebox clause will be added around the tabular environment with the specified value used as the scaling factor. Default value is NULL.

width   If not NULL, the specified value is included in parentheses between the tabular environment begin tag and the alignment specification. This allows specification of the table width when using tabular environments such as tabular* and tabularx. Note that table width specification is not supported with the tabular or longtable environments. Default value is NULL.

comment   If TRUE, the version and timestamp comment is included. Default value is TRUE.

timestamp   Timestamp to include in LaTeX comment. Set this to NULL to exclude the timestamp. Default value is date().

...   Additional arguments. (Currently ignored.)

Details

This function displays or writes to disk the code to produce a table associated with an object x of class "xtable". The resulting code is either a LaTeX or HTML table, depending on the value of type. The function also (invisibly) returns a character vector of the results (which can be helpful for post-processing).

Since version 1.4 the non default behavior of hline.after is changed. To obtain the same results as the previous versions add to the hline.after vector the vector c(-1, 0, nrow(x)) where nrow(x) is the numbers of rows of the object.

From version 1.4-3, all non-numeric columns are sanitized, and all LaTeX special characters are sanitized for LaTeX output. See Section 3 of the xtableGallery vignette for an example of customizing the sanitization. From version 1.4-4, the sanitization also applies to column names. To remove any text sanitization, specify sanitize.text.function=function(x){x}.

From version 1.6-1 the default values for the arguments other than x are obtained using getOption(). Hence the user can set the values once with options() rather than setting them in every call to print.xtable().

The argument format.args is used to supply arguments to the formatC function, but will throw an error if values for digits or format are included in the list of arguments. The recommended approach to specify digits is to supply the argument digits to xtable, and to specify format supply the argument display to xtable. See the examples.

Author(s)

David Dahl <dahl@stat.byu.edu> with contributions and suggestions from many others (see source code).

References

Fairbairns, Robin (2005) *Tables longer than a single page.* The UK List of TeX Frequently Asked Questions on the Web. http://www.tex.ac.uk/cgi-bin/texfaq2html?label=longtab

See Also

xtable, caption, label, align, digits, display, formatC

Examples

```
df <- data.frame(A = c(1.00123, 33.1, 6),
                 B = c(111111, 3333333, 3123.233))
## The following code gives the error
## formal argument "digits" matched by multiple actual arguments
## print(xtable(df, display = c("s","e","e")),
##       format.args = list(digits = 3, big.mark = " ", decimal.mark = ","))
## specify digits as argument to xtable instead
print(xtable(df, display = c("s","f","f"), digits = 4),
      format.args = list(big.mark = " ", decimal.mark = ","))
## The following code gives the error
## formal argument "format" matched by multiple actual arguments
## print(xtable(df, digits = 4),
##       format.args = list(format = c("s","e","e"),
##                          big.mark = " ", decimal.mark = ","))
## specify format using display argument in xtable
print(xtable(df, display = c("s","e","e"), digits = 4),
      format.args = list(big.mark = " ", decimal.mark = ","))
```

---

print.xtableMatharray   *Print Math Array*

Description

For an object of class "xtableMatharray", returns the LaTeX commands to produce an array.

Usage

```
## S3 method for class 'xtableMatharray'
print(x,
  print.results = TRUE,
  format.args = getOption("xtable.format.args", NULL),
  scalebox = getOption("xtable.scalebox", NULL),
```

```
    comment = FALSE,
    timestamp = NULL,
    ...)
```

Arguments

x	An object of class `"xtableMatharray"`.
print.results	If `TRUE`, the generated table is printed to standard output. Set this to `FALSE` if you will just be using the character vector that is returned invisibly. Default value is `TRUE`.
format.args	List of arguments for the `formatC` function. For example, standard German number separators can be specified as `format.args=list(big.mark = "'"`, `decimal.mark = ",")`. The arguments `digits` and `format` should not be included in this list. See details for function [`print.xtable`](). Default value is `NULL`.
scalebox	If not `NULL`, a `scalebox` clause will be added around the tabular environment with the specified value used as the scaling factor. Default value is `NULL`.
comment	If `TRUE`, the version and timestamp comment is included. Default value is `FALSE`.
timestamp	Timestamp to include in LaTeX comment. Set this to `NULL` to exclude the timestamp. Default value is `NULL`.
...	Additional arguments. (Currently ignored.)

Details

This command prints an array of numbers which may be included in a mathematical expression in a LaTeX document created using **Sweave** or **knitr**. Internally it calls `print.data.frame` but with special values for the arguments, namely that the tabular environment is `array`, row names and column names are not included, and there are no horizontal lines. Note that the default values for the arguments `comment` and `timestamp` are different to the default values for `print.xtable`, the justification being that comments would make the resulting LaTeX harder to read.

Value

A character vector containing the LaTeX code for incorporating an array in a mathematical expression.

Author(s)

David Scott <d.scott@auckland.ac.nz>.

See Also

[`xtableMatharray`](), [`print.xtable`]()

Examples

```
V <- matrix(c(1.140380e-03,  3.010497e-05,  7.334879e-05,
              3.010497e-05,  3.320683e-04, -5.284854e-05,
              7.334879e-05, -5.284854e-05,  3.520928e-04), nrow = 3)
### Simple test of print.xtableMatharray
print.xtableMatharray(xtable(V, display = rep("E", 4)))

class(V) <- c("xtableMatharray")
class(V)

### Test without any additional arguments
mth <- xtableMatharray(V)
str(mth)
print(mth)

### Test with arguments to xtable
mth <- xtableMatharray(V, display = rep("E", 4))
str(mth)
print(mth)

mth <- xtableMatharray(V, digits = 6)
str(mth)
print(mth)

### Test with additional print.xtableMatharray arguments
mth <- xtableMatharray(V, digits = 6)
str(mth)
print(mth, format.args = list(decimal.mark = ","))
print(mth, scalebox = 0.5)
print(mth, comment = TRUE)
print(mth, timestamp = "2000-01-01")
print(mth, comment = TRUE, timestamp = "2000-01-01")
```

---

sanitize	*Sanitization Functions*

Description

Functions for sanitizing elements of a table produced by **xtable**. Used for dealing with characters which have special meaning in the output format.

Usage

```
sanitize(str, type = "latex")
sanitize.numbers(str, type, math.style.negative = FALSE,
                 math.style.exponents = FALSE)
sanitize.final(str, type)
as.is(str)
as.math(str, ...)
```

Arguments

`str`	A character object to be sanitized.
`type`	Type of table to produce. Possible values for `type` are `"latex"` or `"html"`. Default value is `"latex"`.
`math.style.negative`	In a LaTeX table, if `TRUE`, then use $-$ for the negative sign (as was the behavior prior to version 1.5-3). Default value is `FALSE`.
`math.style.exponents`	In a LaTeX table, if `TRUE` or `"$$"`, then use `$5 \times 10^{5}$` for 5e5. If `"ensuremath"`, then use `\ensuremath{5 \times 10^{5}}` for 5e5. If `"UTF-8"` or `"UTF-8"`, then use UTF-8 to approximate the LaTeX typsetting for 5e5. Default value is `FALSE`.
`...`	Additional arguments. Character strings or character vectors.

Details

If `type` is `"latex"`, `sanitize()` will replace special characters such as & and the like by strings which will reproduce the actual character, e.g. & is replaced by \&.

If `type` is `"html"`, `sanitize()` will replace special characters such as < and the like by strings which will reproduce the actual character, e.g. < is replaced by <.

When `math.style.negative` is `TRUE`, and `type` is `"latex"`, $-$ is used for the negative sign rather than a simple hyphen (-). No effect when `type` is `"html"`.

When `type` is `"latex"`, and `math.style.exponents` is `TRUE` or `"$$"`, then use `$5 \times 10^{5}$` for 5e5. If `"ensuremath"`, then use `\ensuremath{5 \times 10^{5}}` for 5e5. If `"UTF-8"` or `"UTF-8"`, then use UTF-8 to approximate the LaTeX typsetting for 5e5.

When `type` is `"latex"` `sanitize.final` has no effect. When `type` is `"html"`, multiple spaces are replaced by a single space and occurrences of `' align="left"'` are eliminated.

`as.is` and `as.math` are trivial helper functions to disable sanitizing and to insert a some mathematics in a string respectively.

Value

Returns the sanitized character object.

Author(s)

Code was extracted from `print.xtable()`, in version 1.8.0 of **xtable**. Various authors contributed the original code: Jonathan Swinton <jonathan@swintons.net>, Uwe Ligges <ligges@statistik.uni-dortmund.de>, and probably David B. Dahl <dahl@stat.byu.edu>. `as.is` and `as.math` suggested and provided by Stefan Edwards <sme@iysik.com>.

Examples

```
insane <- c("&",">", ">","_","%","$","\\","#","^","~","{","}")
names(insane) <- c("Ampersand","Greater than","Less than",
                   "Underscore","Percent","Dollar",
```

```
                        "Backslash","Hash","Caret","Tilde",
                        "Left brace","Right brace")
sanitize(insane, type = "latex")
insane <- c("&",">","<")
names(insane) <- c("Ampersand","Greater than","Less than")
sanitize(insane, type = "html")
x <- rnorm(10)
sanitize.numbers(x, "latex", TRUE)
sanitize.numbers(x*10^(10), "latex", TRUE, TRUE)
sanitize.numbers(x, "html", TRUE, TRUE)
as.is(insane)
as.math("x10^10", ": mathematical expression")
```

---

string   *String handling functions*

Description

Private functions for conveniently working with strings.

Usage

```
   string(text,file="",append=FALSE)
   ## S3 method for class 'string'
print(x,...)
   ## S3 method for class 'string'
x + y
   as.string(x,file="",append=FALSE)
   is.string(x)
```

Arguments

text	A character object.
file	Name of the file that should receive the printed string.
append	Should the printed string be appended to the file?
x	A string object.
y	A string object.
...	Additional arguments. (Currently ignored.)

Details

These functions are private functions used by print.xtable. They are not intended to be used elsewhere.

Author(s)

David Dahl <dahl@stat.byu.edu> with contributions and suggestions from many others (see source code).

See Also

print.xtable

---

table.attributes *Retrieve and Set Options for Export Tables*

Description

Functions retrieving or setting table attributes for the supplied object of class ″xtable″.

Usage

```
caption(x,...)
caption(x) <- value
label(x,...)
label(x) <- value
align(x,...)
align(x) <- value
digits(x,...)
digits(x) <- value
display(x,...)
display(x) <- value
```

Arguments

x	An ″xtable″ object.
value	The value of the corresponding attribute.
...	Additional arguments. (Currently ignored.)

Details

These functions retrieve or set table attributes of the object x of class ″xtable″. See xtable for a description of the options.

Author(s)

David Dahl <dahl@stat.byu.edu> with contributions and suggestions from many others (see source code).

See Also

xtable, print.xtable

autoformat, xalign, xdigits, xdisplay

---

tli	*Math scores from Texas Assessment of Academic Skills (TAAS)*

Description

This data set contains math scores and demographic data of 100 randomly selected students participating in the Texas Assessment of Academic Skills (TAAS).

Usage

```
data(tli)
```

Format

A `data.frame` containing 100 observations with the following columns:

`grade` Year in school of student

`sex` Gender of student

`disadvg` Is the student economically disadvantaged?

`ethnicty` Race of student

`tlimth` Math score of student

Source

Texas Education Agency, <http://www.tea.state.tx.us>

---

toLatex.xtable	*Convert Table to Latex*

Description

Function creating a LaTeX representation of an object of class `xtable`.

Usage

```
## S3 method for class 'xtable'
toLatex(object, ...)
```

Arguments

`object`	An object of class `"xtable"`.
`...`	Other arguments to `print.xtable`.

Details

This function creates a LaTeX representation of an object of class "xtable". This is a method for the generic "toLatex" in the core R package "utils".

Author(s)

Charles Roosen <roosen@gmail.com> with contributions and suggestions from many others (see source code).

See Also

print.xtable

---

xtable   *Create Export Tables*

Description

Convert an R object to an xtable object, which can then be printed as a LaTeX or HTML table.

Usage

```
xtable(x, caption = NULL, label = NULL, align = NULL, digits = NULL,
       display = NULL, auto = FALSE, ...)
```

Arguments

x
: An R object of class found among methods(xtable). See below on how to write additional method functions for xtable.

caption
: Character vector of length 1 or 2 containing the table's caption or title. If length is 2, the second item is the "short caption" used when LaTeX generates a "List of Tables". Set to NULL to suppress the caption. Default value is NULL.

label
: Character vector of length 1 containing the LaTeX label or HTML anchor. Set to NULL to suppress the label. Default value is NULL.

align
: Character vector of length equal to the number of columns of the resulting table, indicating the alignment of the corresponding columns. Also, "|" may be used to produce vertical lines between columns in LaTeX tables, but these are effectively ignored when considering the required length of the supplied vector. If a character vector of length one is supplied, it is split as strsplit(align, "")[[1]] before processing. Since the row names are printed in the first column, the length of align is one greater than ncol(x) if x is a data.frame. Use "l", "r", and "c" to denote left, right, and center alignment, respectively. Use "p{3cm}" etc. for a LaTeX column of the specified width. For HTML output the "p" alignment is interpreted as "l", ignoring the width request. Default depends on the class of x.

digits	Numeric vector of length equal to one (in which case it will be replicated as necessary) or to the number of columns of the resulting table **or** matrix of the same size as the resulting table, indicating the number of digits to display in the corresponding columns. Since the row names are printed in the first column, the length of the vector digits or the number of columns of the matrix digits is one greater than ncol(x) if x is a data.frame. Default depends on the class of x. If values of digits are negative, the corresponding values of x are displayed in scientific format with abs(digits) digits.
display	Character vector of length equal to the number of columns of the resulting table, indicating the format for the corresponding columns. Since the row names are printed in the first column, the length of display is one greater than ncol(x) if x is a data.frame. These values are passed to the formatC function. Use "d" (for integers), "f", "e", "E", "g", "G", "fg" (for reals), or "s" (for strings). "f" gives numbers in the usual xxx.xxx format; "e" and "E" give n.ddde+nn or n.dddE+nn (scientific format); "g" and "G" put x[i] into scientific format only if it saves space to do so. "fg" uses fixed format as "f", but digits as number of *significant* digits. Note that this can lead to quite long result strings. Default depends on the class of x.
auto	Logical, indicating whether to apply automatic format when no value is passed to align, digits, or display. This 'autoformat' (based on xalign, xdigits, and xdisplay) can be useful to quickly format a typical matrix or data.frame. Default value is FALSE.
...	Additional arguments. (Currently ignored.)

Details

This function extracts tabular information from x and returns an object of class "xtable". The nature of the table generated depends on the class of x. For example, aov objects produce ANOVA tables while data.frame objects produce a table of the entire data frame. One can optionally provide a caption or label (called an anchor in HTML), as well as formatting specifications. Default values for align, digits, and display are class dependent.

The available method functions for xtable are given by methods(xtable). Users can extend the list of available classes by writing methods for the generic function xtable. These methods functions should have x as their first argument, with additional arguments to specify caption, label, align, digits, and display. Optionally, other arguments may be passed to specify how the object x should be manipulated. All method functions should return an object whose class is c("xtable","data.frame"). The resulting object can have attributes caption and label, but must have attributes align, digits, and display.

Value

For most xtable methods, an object of class "xtable" which inherits the data.frame class and contains several additional attributes specifying the table formatting options.

Author(s)

David Dahl <dahl@stat.byu.edu> with contributions and suggestions from many others (see source code).

See Also

print.xtable, caption, label, align, digits, display

autoformat, xalign, xdigits, xdisplay

xtableList, xtableMatharray

Examples

```
## Load example dataset
data(tli)

## Demonstrate data.frame
tli.table <- xtable(tli[1:20, ])
print(tli.table)
print(tli.table, type = "html")
xtable(mtcars)
xtable(mtcars, auto = TRUE)

## Demonstrate data.frame with different digits in cells
tli.table <- xtable(tli[1:20, ])
display(tli.table)[c(2,6)] <- "f"
digits(tli.table) <- matrix(0:4, nrow = 20, ncol = ncol(tli)+1)
print(tli.table)
print(tli.table, type = "html")

## Demonstrate matrix
design.matrix <- model.matrix(~ sex*grade, data = tli[1:20, ])
design.table <- xtable(design.matrix, auto = TRUE)
print(design.table)
print(design.table, type = "html")

## Demonstrate aov
fm1 <- aov(tlimth ~ sex + ethnicty + grade + disadvg, data = tli)
fm1.table <- xtable(fm1)
print(fm1.table)
print(fm1.table, type = "html")

## Demonstrate lm
fm2 <- lm(tlimth ~ sex*ethnicty, data = tli)
fm2.table <- xtable(fm2)
print(fm2.table)
print(fm2.table, type = "html")
print(xtable(anova(fm2)))
print(xtable(anova(fm2)), type = "html")
fm2b <- lm(tlimth ~ ethnicty, data = tli)
print(xtable(anova(fm2b, fm2)))
print(xtable(anova(fm2b, fm2)), type = "html")

## Demonstrate glm
fm3 <- glm(disadvg ~ ethnicty*grade, data = tli, family = binomial())
fm3.table <- xtable(fm3)
print(fm3.table)
```

```
print(fm3.table, type = "html")
print(xtable(anova(fm3)))
print(xtable(anova(fm3)), type = "html")

## Demonstrate aov
## Taken from help(aov) in R 1.1.1
## From Venables and Ripley (1997) p.210.
N <- c(0,1,0,1,1,1,0,0,0,1,1,0,1,1,0,0,1,0,1,0,1,1,0,0)
P <- c(1,1,0,0,0,1,0,1,1,1,0,0,0,1,0,1,1,0,0,1,0,1,1,0)
K <- c(1,0,0,1,0,1,1,0,0,1,0,1,0,1,1,0,0,0,1,1,1,0,1,0)
yield <- c(49.5,62.8,46.8,57.0,59.8,58.5,55.5,56.0,62.8,55.8,69.5,55.0,
           62.0,48.8,45.5,44.2,52.0,51.5,49.8,48.8,57.2,59.0,53.2,56.0)
npk <- data.frame(block = gl(6,4), N = factor(N), P = factor(P),
                  K = factor(K), yield = yield)
npk.aov <- aov(yield ~ block + N*P*K, npk)
op <- options(contrasts = c("contr.helmert", "contr.treatment"))
npk.aovE <- aov(yield ~  N*P*K + Error(block), npk)
options(op)

summary(npk.aov)
print(xtable(npk.aov))
print(xtable(anova(npk.aov)))
print(xtable(summary(npk.aov)))

summary(npk.aovE)
print(xtable(npk.aovE), type = "html")
print(xtable(summary(npk.aovE)), type = "html")

## Demonstrate lm
## Taken from help(lm) in R 1.1.1
## Annette Dobson (1990) "An Introduction to Generalized Linear Models".
## Page 9: Plant Weight Data.
ctl <- c(4.17,5.58,5.18,6.11,4.50,4.61,5.17,4.53,5.33,5.14)
trt <- c(4.81,4.17,4.41,3.59,5.87,3.83,6.03,4.89,4.32,4.69)
group <- gl(2,10,20, labels = c("Ctl","Trt"))
weight <- c(ctl, trt)
lm.D9 <- lm(weight ~ group)
print(xtable(lm.D9))
print(xtable(anova(lm.D9)))

## Demonstrate glm
## Taken from help(glm) in R 1.1.1
## Annette Dobson (1990) "An Introduction to Generalized Linear Models".
## Page 93: Randomized Controlled Trial :
counts <- c(18,17,15,20,10,20,25,13,12)
outcome <- gl(3,1,9)
treatment <- gl(3,3)
d.AD <- data.frame(treatment, outcome, counts)
glm.D93 <- glm(counts ~ outcome + treatment, family = poisson())
print(xtable(glm.D93, align = "r|llrc"))
print(xtable(anova(glm.D93)), hline.after = c(1), size = "small")

## Demonstration of additional formatC() arguments.
```

```
print(fm1.table, format.args = list(big.mark = "'", decimal.mark = ","))

## Demonstration of "short caption" support.
fm1sc <- aov(tlimth ~ sex + ethnicty + grade, data = tli)
fm1sc.table <- xtable(fm1sc,
  caption = c("ANOVA Model with Predictors Sex, Ethnicity, and Grade",
    "ANOVA: Sex, Ethnicity, Grade"))
print(fm1sc.table)

## Demonstration of longtable support.
## Remember to insert \usepackage{longtable} on your LaTeX preamble
x <- matrix(rnorm(1000), ncol = 10)
x.big <- xtable(x, label = 'tabbig',
                caption = 'Example of longtable spanning several pages')
print(x.big, tabular.environment = 'longtable', floating = FALSE)
x <- x[1:30, ]
x.small <- xtable(x, label = 'tabsmall', caption = 'regular table env')
print(x.small)  # default, no longtable

## Demonstration of sidewaystable support.
## Remember to insert \usepackage{rotating} on your LaTeX preamble
print(x.small, floating.environment = 'sidewaystable')

if(require(stats, quietly = TRUE)) {
  ## Demonstrate prcomp
  ## Taken from help(prcomp) in mva package of R 1.1.1
  data(USArrests)
  pr1 <- prcomp(USArrests)
  print(xtable(pr1))
  print(xtable(summary(pr1)))

#   ## Demonstrate princomp
#   ## Taken from help(princomp) in mva package of R 1.1.1
#   pr2 <- princomp(USArrests)
#   print(xtable(pr2))
}

## Demonstrate include.rownames, include.colnames,
## only.contents and add.to.row arguments
set.seed(2345)
res <- matrix(sample(0:9, size = 6*9, replace = TRUE), ncol = 6, nrow = 9)
xres <- xtable(res)
digits(xres) <- rep(0, 7)
addtorow <- list()
addtorow$pos <- list()
addtorow$pos[[1]] <- c(0, 2)
addtorow$pos[[2]] <- 4
addtorow$command <- c('\vspace{2mm} \n', '\vspace{10mm} \n')
print(xres, add.to.row = addtorow, include.rownames = FALSE,
      include.colnames = TRUE, only.contents = TRUE,
      hline.after = c(0, 0, 9, 9))

## Demonstrate include.rownames, include.colnames,
```

```
## only.contents and add.to.row arguments in Rweave files

## Not run:
 \begin{small}
 \setlongtables
 \begin{longtable}{
 <<results = tex, fig = FALSE>>=
 cat(paste(c('c', rep('cc', 34/2-1), 'c'), collapse = '@{\hspace{2pt}}'))
 @
 }
 \hline
 \endhead
 \hline
 \endfoot
 <<results = tex, fig = FALSE>>=
 library(xtable)
 set.seed(2345)
 res <- matrix(sample(0:9, size = 34*90, replace = TRUE), ncol = 34, nrow = 90)
 xres <- xtable(res)
 digits(xres) <- rep(0, 35)
 addtorow <- list()
 addtorow$pos <- list()
 addtorow$pos[[1]] <- c(seq(4, 40, 5), seq(49, 85, 5))
 addtorow$pos[[2]] <- 45
 addtorow$command <- c('\vspace{2mm} \n', '\newpage \n')
 print(xres, add.to.row = addtorow, include.rownames = FALSE,
         include.colnames = FALSE, only.contents = TRUE, hline.after = NULL)
 @
 \end{longtable}
 \end{small}

## End(Not run)

## Demonstrate sanitization
mat <- round(matrix(c(0.9, 0.89, 200, 0.045, 2.0), c(1, 5)), 4)
rownames(mat) <- "$y_{t-1}$"
colnames(mat) <- c("$R^2$", "$\\bar{R}^2$", "F-stat", "S.E.E", "DW")
print(xtable(mat), type = "latex", sanitize.text.function = function(x){x})

## Demonstrate booktabs
print(tli.table)
print(tli.table, hline.after = c(-1,0))
print(tli.table, hline.after = NULL)
print(tli.table,
      add.to.row = list(pos = list(2), command = c("\vspace{2mm} \n")))

print(tli.table, booktabs = TRUE)
print(tli.table, booktabs = TRUE, hline.after = c(-1,0))
print(tli.table, booktabs = TRUE, hline.after = NULL)
print(tli.table, booktabs = TRUE,
  add.to.row = list(pos = list(2), command = c("\vspace{2mm} \n")))
print(tli.table, booktabs = TRUE, add.to.row = list(pos = list(2),
  command = c("youhou\n")), tabular.environment = "longtable")
```

---

xtableFtable   *Create and Export Flat Tables*

Description

xtableFtable creates an object which contains information about a flat table which can be used by
print.xtableFtable to produce a character string which when included in a document produces
a nicely formatted flat table.

Usage

```
xtableFtable(x, caption = NULL, label = NULL,
             align = NULL, digits = 0, display = NULL,
             quote = FALSE,
             method = c("non.compact", "row.compact",
                         "col.compact", "compact"),
             lsep = " $\\vert$ ", ...)

## S3 method for class 'xtableFtable'
print(x,
  type = getOption("xtable.type", "latex"),
  file = getOption("xtable.file", ""),
  append = getOption("xtable.append", FALSE),
  floating = getOption("xtable.floating", TRUE),
  floating.environment = getOption("xtable.floating.environment", "table"),
  table.placement = getOption("xtable.table.placement", "ht"),
  caption.placement = getOption("xtable.caption.placement", "bottom"),
  caption.width = getOption("xtable.caption.width", NULL),
  latex.environments = getOption("xtable.latex.environments", c("center")),
  tabular.environment = getOption("xtable.tabular.environment", "tabular"),
  size = getOption("xtable.size", NULL),
  hline.after = getOption("xtable.hline.after", NULL),
  NA.string = getOption("xtable.NA.string", ""),
  only.contents = getOption("xtable.only.contents", FALSE),
  add.to.row = getOption("xtable.add.to.row", NULL),
  sanitize.text.function = getOption("xtable.sanitize.text.function", as.is),
  sanitize.rownames.function = getOption("xtable.sanitize.rownames.function",
                                        sanitize.text.function),
  sanitize.colnames.function = getOption("xtable.sanitize.colnames.function",
                                        sanitize.text.function),
  math.style.negative = getOption("xtable.math.style.negative", FALSE),
  math.style.exponents = getOption("xtable.math.style.exponents", FALSE),
  html.table.attributes = getOption("xtable.html.table.attributes",
```

```
                                     "border=1"),
      print.results = getOption("xtable.print.results", TRUE),
      format.args = getOption("xtable.format.args", NULL),
      rotate.rownames = getOption("xtable.rotate.rownames", FALSE),
      rotate.colnames = getOption("xtable.rotate.colnames", FALSE),
      booktabs = getOption("xtable.booktabs", FALSE),
      scalebox = getOption("xtable.scalebox", NULL),
      width = getOption("xtable.width", NULL),
      comment = getOption("xtable.comment", TRUE),
      timestamp = getOption("xtable.timestamp", date()),
      ...)
```

Arguments

x	For xtableFtable, an object of class "ftable". For print.xtableFtable, an object of class c("xtableFtable", "ftable").
caption	Character vector of length 1 or 2 containing the table's caption or title. If length is 2, the second item is the "short caption" used when LaTeX generates a "List of Tables". Set to NULL to suppress the caption. Default value is NULL.
label	Character vector of length 1 containing the LaTeX label or HTML anchor. Set to NULL to suppress the label. Default value is NULL.
align	Character vector of length equal to the number of columns of the resulting table, indicating the alignment of the corresponding columns. Also, "|" may be used to produce vertical lines between columns in LaTeX tables, but these are effectively ignored when considering the required length of the supplied vector. If a character vector of length one is supplied, it is split as strsplit(align, "")[[1]] before processing. For a flat table, the number of columns is the number of columns of data, plus the number of row variables in the table, plus one for the row names, even though row names are not printed. Use "l", "r", and "c" to denote left, right, and center alignment, respectively. Use "p{3cm}" etc. for a LaTeX column of the specified width. For HTML output the "p" alignment is interpreted as "l", ignoring the width request. If NULL all row variable labels will be left aligned, separated from the data columns by a vertical line, and all data columns will be right aligned. The actual length of align depends on the value of method.
digits	Numeric vector of length equal to one (in which case it will be replicated as necessary) or to the number of columns in the resulting table. Since data in the table consists of counts, the default is 0. If the value of digits is negative, the corresponding columns are displayed in scientific format with abs(digits) digits.
display	Character vector of length equal to the number of columns of the resulting table, indicating the format for the corresponding columns. These values are passed to the formatC function. Use "d" (for integers), "f", "e", "E", "g", "G", "fg" (for reals), or "s" (for strings). "f" gives numbers in the usual xxx.xxx format; "e" and "E" give n.ddde+nn or n.dddE+nn (scientific format); "g" and "G" put x[i] into scientific format only if it saves space to do so. "fg" uses fixed format as "f", but digits as number of *significant* digits. Note that this can

	lead to quite long result strings. If NULL all row variable names and labels will have format "s", and all data columns will have format "d". The actual length of display depends on the value of method.
quote	A character string giving the set of quoting characters for format.ftable used in print.xtableFtable. To disable quoting altogether, use quote="".
method	String specifying how the "xtableFtable" object is printed in the print method. Can be abbreviated. Available methods are (see the examples in [print.ftable](#)):

"non.compact" the default representation of an "ftable" object.

"row.compact" a row-compact version without empty cells below the column labels.

"col.compact" a column-compact version without empty cells to the right of the row labels.

"compact" a row- and column-compact version. This may imply a row and a column label sharing the same cell. They are then separated by the string lsep.

lsep	Only for method = "compact", the separation string for row and column labels.
type	Type of table to produce. Possible values for type are "latex" or "html". Default value is "latex" and is the only type implemented so far.
file	Name of file where the resulting code should be saved. If file="", output is displayed on screen. Note that the function also (invisibly) returns a character vector of the results (which can be helpful for post-processing). Default value is "".
append	If TRUE and file!="", code will be appended to file instead of overwriting file. Default value is FALSE.
floating	If TRUE and type="latex", the resulting table will be a floating table (using, for example, \begin{table} and \end{table}). See floating.environment below. Default value is TRUE.
floating.environment	
	If floating=TRUE and type="latex", the resulting table uses the specified floating environment. Possible values include "table", "table*", and other floating environments defined in LaTeX packages. Default value is "table".
table.placement	
	If floating=TRUE and type="latex", the floating table will have placement given by table.placement where table.placement must be NULL or contain only elements of {"h","t","b","p","!","H"}. Default value is "ht".
caption.placement	
	The caption will be placed at the bottom of the table if caption.placement is "bottom" and at the top of the table if it equals "top". Default value is "bottom".
caption.width	The caption will be placed in a "parbox" of the specified width if caption.width is not NULL and type="latex". Default value is NULL.
latex.environments	
	If floating=TRUE and type="latex", the specified LaTeX environments (provided as a character vector) will enclose the tabular environment. Default value is "center".

tabular.environment

> When `type="latex"`, the tabular environment that will be used. When working with tables that extend more than one page, using `tabular.environment="longtable"` with the corresponding LaTeX package (see Fairbairns, 2005) allows one to typeset them uniformly. Note that `floating` should be set to `FALSE` when using the `longtable` environment. Default value is `"tabular"`.

size

> A character vector that is inserted just before the tabular environment starts. This can be used to set the font size and a variety of other table settings. Initial backslashes are automatically prefixed, if not supplied by user. Default value is `NULL`.

hline.after

> When `type="latex"`, a vector of numbers between -1 and `nrow(x)`, inclusive, indicating the rows after which a horizontal line should appear. Repeated values are allowed. If `NULL` the default is to draw a line before before starting the table, after the column variable names and labels, and at the end of the table.

NA.string

> String to be used for missing values in table entries. Default value is `""`.

only.contents

> If `TRUE` only the rows of the table are printed. Default value is `FALSE`.

add.to.row

> A list of two components. The first component (which should be called 'pos') is a list that contains the position of rows on which extra commands should be added at the end. The second component (which should be called 'command') is a character vector of the same length as the first component, which contains the command that should be added at the end of the specified rows. Default value is `NULL`, i.e. do not add commands.

sanitize.text.function

> Since the table entries are counts no sanitization is necessary. The default is `as.is`, which is the function which makes no changes. This also applies to the labels for the row and column variables since these are also part of the table which is printed using a call to `print.xtable`.

sanitize.rownames.function

> Like the `sanitize.text.function`, but applicable to row names. The default uses the `sanitize.text.function`.

sanitize.colnames.function

> Like the `sanitize.text.function`, but applicable to column names. The default uses the `sanitize.text.function`.

math.style.negative

> In a LaTeX table, if `TRUE`, then use $-$ for the negative sign (as was the behavior prior to version 1.5-3). Default value is `FALSE`.

math.style.exponents

> In a LaTeX table, if `TRUE` or `"$$"`, then use 5×10^{5} for 5e5. If `"ensuremath"`, then use `\ensuremath{5 \times 10^{5}}` for 5e5. If `"UTF-8"` or `"UTF-8"`, then use UTF-8 to approximate the LaTeX typesetting for 5e5. Default value is `FALSE`.

html.table.attributes

> In an HTML table, attributes associated with the `<TABLE>` tag. Default value is `"border=1"`.

print.results

> If `TRUE`, the generated table is printed to standard output. Set this to `FALSE` if you will just be using the character vector that is returned invisibly. Default value is `TRUE`.

format.args	List of arguments for the formatC function. For example, standard German number separators can be specified as format.args=list(big.mark = "'", decimal.mark = ",")). The arguments digits and format should not be included in this list. Default value is NULL.
rotate.rownames	
	If TRUE, the row names and labels, and column variable names are displayed vertically in LaTeX. Default value is FALSE.
rotate.colnames	
	If TRUE, the column names and labels, and row variable names are displayed vertically in LaTeX. Default value is FALSE.
booktabs	If TRUE, the toprule, midrule and bottomrule commands from the LaTeX "booktabs" package are used rather than hline for the horizontal line tags.
scalebox	If not NULL, a scalebox clause will be added around the tabular environment with the specified value used as the scaling factor. Default value is NULL.
width	If not NULL, the specified value is included in parentheses between the tabular environment begin tag and the alignment specification. This allows specification of the table width when using tabular environments such as tabular* and tabularx. Note that table width specification is not supported with the tabular or longtable environments. Default value is NULL.
comment	If TRUE, the version and timestamp comment is included. Default value is TRUE.
timestamp	Timestamp to include in LaTeX comment. Set this to NULL to exclude the timestamp. Default value is date().
...	Additional arguments. (Currently ignored.)

Details

xtableFtable carries out some calculations to determine the number of rows and columns of names and labels which will be in the table when formatted as a flat table, which depends on the value of method. It uses the results of those calculations to set sensible values for align and display if these have not been supplied. It attaches attributes to the resulting object which specify details of the function call which are needed when printing the resulting object which is of class c("xtableFtable", "ftable").

print.xtableFtable uses the attributes attached to an object of class c("xtableFtable", "ftable") to create a suitable character matrix object for subsequent printing. Formatting is carried out by changing the class of the c("xtableFtable", "ftable") to "ftable" then using the generic format to invoke format.ftable, from the **stats** package. The matrix object produced is then printed via a call to print.xtable.

Note that at present there is no code for type = "html".

Value

For xtableFtable an object of class c("xtableFtable", "ftable"), with attributes

ftableCaption	the value of the caption argument
ftableLabel	the value of the label argument
ftableAlign	the value of the label argument

ftableDigits	the value of the `digits` argument or the default value if `digits = NULL`
quote	the value of the `quote` argument
ftableDisplay	the value of the `display` argument or the default value if `align = NULL`
method	the value of the `method` argument
lsep	the value of the `lsep` argument
nChars	a vector of length 2 giving the number of character rows and the number of character columns

For `print.xtableFtable` a character string which will produce a formatted table when included in a LaTeX document.

Note

The functions `xtableFtable` and `print.xtableFtable` are new and their behaviour may change in the future based on user experience and recommendations.

It is not recommended that users change the values of `align`, `digits` or `align`. First of all, alternative values have not been tested. Secondly, it is most likely that to determine appropriate values for these arguments, users will have to investigate the code for `xtableFtable` and/or `print.xtableFtable`.

Author(s)

David Scott <d.scott@auckland.ac.nz>.

References

Fairbairns, Robin (2005) *Tables longer than a single page.* The UK List of TeX Frequently Asked Questions on the Web. <http://www.tex.ac.uk/cgi-bin/texfaq2html?label=longtab>

See Also

`ftable`, `print.ftable`

`xtable`, `caption`, `label`, `align`, `digits`, `display`, `formatC`

Examples

```
data(mtcars)
mtcars$cyl <- factor(mtcars$cyl, levels = c("4","6","8"),
                     labels = c("four","six","eight"))
tbl <- ftable(mtcars$cyl, mtcars$vs, mtcars$am, mtcars$gear,
              row.vars = c(2, 4),
              dnn = c("Cylinders", "V/S", "Transmission", "Gears"))
xftbl <- xtableFtable(tbl, method = "compact")
print.xtableFtable(xftbl, booktabs = TRUE)
xftbl <- xtableFtable(tbl, method = "row.compact")
print.xtableFtable(xftbl, rotate.colnames = TRUE,
                   rotate.rownames = TRUE)
```

---

xtableList | *Create and Export Lists of Tables*

Description

xtableList creates an object from a list of tables, which can be used by print.xtableList to produce a composite table containing the information from the individual tables.

Usage

```
xtableList(x, caption = NULL, label = NULL,
           align = NULL, digits = NULL, display = NULL, ...)

## S3 method for class 'xtableList'
print(x,
  type = getOption("xtable.type", "latex"),
  file = getOption("xtable.file", ""),
  append = getOption("xtable.append", FALSE),
  floating = getOption("xtable.floating", TRUE),
  floating.environment = getOption("xtable.floating.environment", "table"),
  table.placement = getOption("xtable.table.placement", "ht"),
  caption.placement = getOption("xtable.caption.placement", "bottom"),
  caption.width = getOption("xtable.caption.width", NULL),
  latex.environments = getOption("xtable.latex.environments", c("center")),
  tabular.environment = getOption("xtable.tabular.environment", "tabular"),
  size = getOption("xtable.size", NULL),
  hline.after = NULL,
  NA.string = getOption("xtable.NA.string", ""),
  include.rownames = getOption("xtable.include.rownames", TRUE),
  colnames.format = "single",
  only.contents = getOption("xtable.only.contents", FALSE),
  add.to.row = NULL,
  sanitize.text.function = getOption("xtable.sanitize.text.function", NULL),
  sanitize.rownames.function = getOption("xtable.sanitize.rownames.function",
                                         sanitize.text.function),
  sanitize.colnames.function = getOption("xtable.sanitize.colnames.function",
                                         sanitize.text.function),
  sanitize.subheadings.function =
    getOption("xtable.sanitize.subheadings.function",
              sanitize.text.function),
  sanitize.message.function =
    getOption("xtable.sanitize.message.function",
              sanitize.text.function),
  math.style.negative = getOption("xtable.math.style.negative", FALSE),
  math.style.exponents = getOption("xtable.math.style.exponents", FALSE),
  html.table.attributes = getOption("xtable.html.table.attributes", "border=1"),
  print.results = getOption("xtable.print.results", TRUE),
```

```
    format.args = getOption("xtable.format.args", NULL),
    rotate.rownames = getOption("xtable.rotate.rownames", FALSE),
    rotate.colnames = getOption("xtable.rotate.colnames", FALSE),
    booktabs = getOption("xtable.booktabs", FALSE),
    scalebox = getOption("xtable.scalebox", NULL),
    width = getOption("xtable.width", NULL),
    comment = getOption("xtable.comment", TRUE),
    timestamp = getOption("xtable.timestamp", date()),
    ...)
```

Arguments

x   For xtableList, a list of R objects all of the same class, being a class found
                among methods(xtable). The list may also have attributes "subheadings"
                and "message". The attribute "subheadings" should be a character vector of
                the same length as the list x. The attribute "message" should be a character
                vector of any length. For print.xtableList, an object of class xtableList
                produced by a call to xtableList.

caption   Character vector of length 1 or 2 containing the table's caption or title. If length
                is 2, the second item is the "short caption" used when LaTeX generates a "List
                of Tables". Set to NULL to suppress the caption. Default value is NULL.

label   Character vector of length 1 containing the LaTeX label or HTML anchor. Set
                to NULL to suppress the label. Default value is NULL.

align   Character vector of length equal to the number of columns of the resulting ta-
                ble, indicating the alignment of the corresponding columns. Also, "|" may be
                used to produce vertical lines between columns in LaTeX tables, but these are
                effectively ignored when considering the required length of the supplied vector.
                If a character vector of length one is supplied, it is split as strsplit(align,
                "")[[1]] before processing. Since the row names are printed in the first col-
                umn, the length of align is one greater than ncol(x) if x is a data.frame. Use
                "l", "r", and "c" to denote left, right, and center alignment, respectively. Use
                "p{3cm}" etc. for a LaTeX column of the specified width. For HTML output the
                "p" alignment is interpreted as "l", ignoring the width request. Default depends
                on the class of x.

digits   Either NULL, or a vector of length one, or a vector of length equal to the number
                of columns in the resulting table, indicating the number of digits to display in
                the corresponding columns, or a list if length equal to the number of R objects
                making up x, all members being vectors of the same length, either length one or
                of length equal to the number of columns in the resulting table. See details for
                further information.

display   Either NULL, or a vector of length one, or a vector of length equal to the number
                of columns in the resulting table, indicating the format of the corresponding
                columns, or a list if length equal to the number of R objects making up x, all
                members being vectors of the same length, either length one or of length equal to
                the number of columns in the resulting table. See details for further information.

type   Type of table to produce. Possible values for type are "latex" or "html".
                Default value is "latex".
```

file            Name of file where the resulting code should be saved. If file="", output is
                displayed on screen. Note that the function also (invisibly) returns a character
                vector of the results (which can be helpful for post-processing). Default value is
                "".

append          If TRUE and file!="", code will be appended to file instead of overwriting
                file. Default value is FALSE.

floating        If TRUE and type="latex", the resulting table will be a floating table (using,
                for example, \begin{table} and \end{table}). See floating.environment
                below. Default value is TRUE.

floating.environment
                If floating=TRUE and type="latex", the resulting table uses the specified
                floating environment. Possible values include "table", "table*", and other
                floating environments defined in LaTeX packages. Default value is "table".

table.placement
                If floating=TRUE and type="latex", the floating table will have placement
                given by table.placement where table.placement must be NULL or contain
                only elements of {"h","t","b","p","!","H"}. Default value is "ht".

caption.placement
                The caption will be placed at the bottom of the table if caption.placement
                is "bottom" and at the top of the table if it equals "top". Default value is
                "bottom".

caption.width   The caption will be placed in a "parbox" of the specified width if caption.width
                is not NULL and type="latex". Default value is NULL.

latex.environments
                If floating=TRUE and type="latex", the specified LaTeX environments (pro-
                vided as a character vector) will enclose the tabular environment. Default value
                is "center".

tabular.environment
                When type="latex", the tabular environment that will be used. When working
                with tables that extend more than one page, using tabular.environment="longtable"
                with the corresponding LaTeX package (see Fairbairns, 2005) allows one to
                typeset them uniformly. Note that floating should be set to FALSE when using
                the longtable environment. Default value is "tabular".

size            A character vector that is inserted just before the tabular environment starts.
                This can be used to set the font size and a variety of other table settings. Initial
                backslashes are automatically prefixed, if not supplied by user. Default value is
                NULL.

hline.after     When type="latex", a vector of numbers between -1 and the number of rows
                in the resulting table, inclusive, indicating the rows after which a horizontal line
                should appear. Determining row numbers is not straightforward since some lines
                in the resulting table don't enter into the count. The default depends on the value
                of col.names.format.

NA.string       String to be used for missing values in table entries. Default value is "".

include.rownames
                If TRUE the rows names are printed. Default value is TRUE.

colnames.format

> Either "single" or "multiple". Default is "single".

only.contents   If TRUE only the rows of the table are printed. Default value is FALSE.

add.to.row      A list of two components. The first component (which should be called 'pos')
                is a list that contains the position of rows on which extra commands should be
                added at the end. The second component (which should be called 'command') is
                a character vector of the same length as the first component, which contains the
                command that should be added at the end of the specified rows. Default value is
                NULL, i.e. do not add commands.

sanitize.text.function

> All non-numeric entries (except row and column names) are sanitized in an at-
> tempt to remove characters which have special meaning for the output format.
> If sanitize.text.function is not NULL, it should be a function taking a char-
> acter vector and returning one, and will be used for the sanitization instead of
> the default internal function. Default value is NULL.

sanitize.rownames.function

> Like the sanitize.text.function, but applicable to row names. The default
> uses the sanitize.text.function.

sanitize.colnames.function

> Like the sanitize.text.function, but applicable to column names. The de-
> fault uses the sanitize.text.function.

sanitize.subheadings.function

> Like the sanitize.text.function, but applicable to subheadings. The default
> uses the sanitize.text.function.

sanitize.message.function

> Like the sanitize.text.function, but applicable to the message. The default
> uses the sanitize.text.function.

math.style.negative

> In a LaTeX table, if TRUE, then use $-$ for the negative sign (as was the behavior
> prior to version 1.5-3). Default value is FALSE.

math.style.exponents

> In a LaTeX table, if TRUE or "$$", then use $5 \times 10^{5}$ for 5e5. If
> "ensuremath", then use \ensuremath{5 \times 10^{5}} for 5e5. If "UTF-8"
> or "UTF-8", then use UTF-8 to approximate the LaTeX typesetting for 5e5. De-
> fault value is FALSE.

html.table.attributes

> In an HTML table, attributes associated with the <TABLE> tag. Default value is
> "border=1".

print.results   If TRUE, the generated table is printed to standard output. Set this to FALSE if you
                will just be using the character vector that is returned invisibly. Default value is
                TRUE.

format.args     List of arguments for the formatC function. For example, standard German
                number separators can be specified as format.args=list(big.mark = "'",
                decimal.mark = ",")). The arguments digits and format should not be in-
                cluded in this list. See details. Default value is NULL.

rotate.rownames

> If TRUE, the row names are displayed vertically in LaTeX. Default value is
> FALSE.

| | |
|---|---|
| rotate.colnames | |
| | If TRUE, the column names are displayed vertically in LaTeX. Default value is FALSE. |
| booktabs | If TRUE, the toprule, midrule and bottomrule commands from the LaTeX "booktabs" package are used rather than hline for the horizontal line tags. |
| scalebox | If not NULL, a scalebox clause will be added around the tabular environment with the specified value used as the scaling factor. Default value is NULL. |
| width | If not NULL, the specified value is included in parentheses between the tabular environment begin tag and the alignment specification. This allows specification of the table width when using tabular environments such as tabular* and tabularx. Note that table width specification is not supported with the tabular or longtable environments. Default value is NULL. |
| comment | If TRUE, the version and timestamp comment is included. Default value is TRUE. |
| timestamp | Timestamp to include in LaTeX comment. Set this to NULL to exclude the timestamp. Default value is date(). |
| ... | Additional arguments. (Currently ignored.) |

## Details

xtableList produces an object suitable for printing using print.xtableList.

The elements of the list x supplied to xtableList must all have the same structure. When these list items are submitted to xtable the resulting table must have the same number of columns with the same column names and type of data.

The values supplied to arguments digits and display, must be composed of elements as specified in those same arguments for the function [xtable](). See the help for [xtable]() for details.

print.xtableList produces tables in two different formats depending on the value of col.names.format. If col.names.format = "single", the resulting table has only a single heading row. If col.names.format = "multiple" there is a heading row for each of the subtables making up the complete table.

By default if col.names.format = "single", there are horizontal lines above and below the heading row, and at the end of each subtable. If col.names.format = "multiple", there are horizontal lines above and below each appearance of the heading row, and at the end of each subtable.

If "subheadings" is not NULL, the individual elements of this vector (which can include newlines \n) produce a heading line or lines for the subtables. When col.names.format = "multiple" these subheadings appear *above* the heading rows.

If "message" is not NULL the vector produces a line or lines at the end of the table.

Consult the vignette 'The xtableList Gallery' to see the behaviour of these functions.

Note that at present there is no code for type = "html".

## Value

xtableList produces an object of class "xtableList". An object of this class is a list of "xtable" objects with some additional attributes. Each element of the list can have a "subheading" attribute. The list can also have a "message" attribute.

print.xtableList produces a character string containing LaTeX markup which produces a composite table in a LaTeX document.

## Author(s)

David Scott <d.scott@auckland.ac.nz>.

## See Also

xtable, print.xtable, formatC

caption, label, align, digits, display

## Examples

```
data(mtcars)
mtcars <- mtcars[, 1:6]
mtcarsList <- split(mtcars, f = mtcars$cyl)
attr(mtcarsList, ″subheadings″) <- paste0(″Number of cylinders = ″,
                                           names(mtcarsList))
attr(mtcarsList, ″message″) <- c(″Line 1 of Message″,
                                 ″Line 2 of Message″)
xList <- xtableList(mtcarsList)
print.xtableList(xList)
print.xtableList(xList, colnames.format = ″multiple″)
```

---

xtableMatharray   *Create LaTeX Mathematical Array*

Description

Convert an array of numbers or mathematical expressions into an xtableMatharray object so it can be printed. A convenience function to enable the printing of arrays in mathematical expressions in LaTeX.

Usage

```
xtableMatharray(x, caption = NULL, label = NULL, align = NULL,
                digits = NULL, display = NULL, auto = FALSE, ...)
```

Arguments

x	A numeric or character matrix.
caption	Character vector of length 1 or 2 containing the table's caption or title. If length is 2, the second item is the "short caption" used when LaTeX generates a "List of Tables". Set to NULL to suppress the caption. Default value is NULL. Included here only for consistency with xtable methods. Not expected to be of use.
label	Character vector of length 1 containing the LaTeX label. Set to NULL to suppress the label. Default value is NULL.

align   Character vector of length equal to the number of columns of the resulting ta-
               ble, indicating the alignment of the corresponding columns. Also, `"|"` may be
               used to produce vertical lines between columns in LaTeX tables, but these are
               effectively ignored when considering the required length of the supplied vector.
               If a character vector of length one is supplied, it is split as `strsplit(align,`
               `"")[[1]]` before processing. Since the row names are printed in the first col-
               umn, the length of `align` is one greater than `ncol(x)` if x is a `data.frame`. Use
               `"l"`, `"r"`, and `"c"` to denote left, right, and center alignment, respectively. Use
               `"p{3cm}"` etc. for a LaTeX column of the specified width. For HTML output the
               `"p"` alignment is interpreted as `"l"`, ignoring the width request. Default depends
               on the class of x.

digits   Numeric vector of length equal to one (in which case it will be replicated as
               necessary) or to the number of columns of the resulting table **or** matrix of the
               same size as the resulting table, indicating the number of digits to display in the
               corresponding columns. Since the row names are printed in the first column, the
               length of the vector `digits` or the number of columns of the matrix `digits` is
               one greater than `ncol(x)` if x is a `data.frame`. Default depends on the class of
               x. If values of `digits` are negative, the corresponding values of x are displayed
               in scientific format with `abs(digits)` digits.

display   Character vector of length equal to the number of columns of the resulting table,
               indicating the format for the corresponding columns. Since the row names are
               printed in the first column, the length of `display` is one greater than `ncol(x)` if
               x is a `data.frame`. These values are passed to the `formatC` function. Use `"d"`
               (for integers), `"f"`, `"e"`, `"E"`, `"g"`, `"G"`, `"fg"` (for reals), or `"s"` (for strings). `"f"`
               gives numbers in the usual `xxx.xxx` format; `"e"` and `"E"` give `n.ddde+nn` or
               `n.dddE+nn` (scientific format); `"g"` and `"G"` put `x[i]` into scientific format only
               if it saves space to do so. `"fg"` uses fixed format as `"f"`, but `digits` as number
               of *significant* digits. Note that this can lead to quite long result strings. Default
               depends on the class of x.

auto   Logical, indicating whether to apply automatic format when no value is passed
               to `align`, `digits`, or `display`. This 'autoformat' (based on `xalign`, `xdigits`,
               and `xdisplay`) can be useful to quickly format a typical `matrix` or `data.frame`.
               Default value is `FALSE`.

...   Additional arguments. (Currently ignored.)

Details

This function is only usable for production of LaTeX documents, not HTML.

Creates an object of class `c("xtableMatharray","xtable","data.frame")`, to ensure that it is
printed by the print method `print.xtableMatharray`.

Value

An object of class `c("xtableMatharray","xtable","data.frame")`.

Author(s)

David Scott <d.scott@auckland.ac.nz>

See Also

print.xtableMatharray

Examples

```
V <- matrix(c(1.140380e-03,  3.010497e-05,  7.334879e-05,
              3.010497e-05,  3.320683e-04, -5.284854e-05,
              7.334879e-05, -5.284854e-05,  3.520928e-04), nrow = 3)
mth <- xtableMatharray(V)
class(mth)
str(mth)
unclass(mth)
```

Index