Package: muReportR (via r-universe)

July 26, 2024

Title muReportR

Description Generate HTML reports from R. Based on code from the RnBeads package.

Date 2019-06-06

Version 0.3

License GPL-3

Encoding UTF-8

Imports ggplot2

Collate 'muReportR.R' 'utils.R' 'Report-class.R' 'Report-methods.R' 'ReportPlot-class.R' 'ReportPlot-methods.R'

RoxygenNote 6.1.1

Repository https://blaserlab.r-universe.dev

RemoteUrl https://github.com/demuellae/muReportR

RemoteRef HEAD

RemoteSha f020fabaddf90b52006b66886977b486d1a5a4d4

Contents

`addReportFigure` *addReportFigure*

Description

Generates HTML code for a figure in the specified report. A figure is a collection of images (plots), of which only one is visible at any given moment.

Usage

```
addReportFigure(report, description, repPlots, setting.names = list(),
  selected.image = as.integer(1))
```

Arguments

`report`	Report to write the text to.
`description`	Human-readable description of the figure. This must be a non-empty `character` vector. The elements of this vector are concatenated without a separator to form the full description.
`repPlots`	Object of type `ReportPlot`, or a list of such objects.
`setting.names`	List of plot file element descriptors. Every variable elements in the plot file names must be included in this list. Set this to empty list if no variable elements are present, that is, if the figure should present a single report plot.
`selected.image`	Index of plot to be initially selected in the figure.

Value

The modified report.

Author(s)

adapted by Fabian Mueller from RnBeads code by Yassen Assenov

See Also

`addReportTables` for adding a listing of tables; `Report` for other functions adding contents to an HTML report

addReportList *addReportList*

Description

Generates HTML code for a list in the specified report.

Usage

```
addReportList(report, txt, type = "u")
```

Arguments

report	Report to write the text to.
txt	Non-empty list of items to be written. An attribute named type, if it exists, specifies the type of the list. See the *Details* section for more information. Every item must be either a nested list, denoting a sublist, or a character vector (or array), storing the text to be written. Any other objects are coerced to a character type. Elements are concatenated without a separator to form the text for a list item.
type	List type to be used for the list and/or its sublists in case the attribute type is not specified.

Details

There are two ways to specify a list type: (1) setting a value for the attribute type of the list, or (2) using the function's parameter type. The value of the function's parameter is used only for lists and sublists that do not contain an attribute named type. The following types are supported:

"o" Ordered list using arabic numbers - 1, 2, 3, etc.

"u" Unordered list using bullet points.

Note that every list type must be a one-element character vector containing one of the codes listed above. Specifying any other value for list type results in an error.

Value

The modified report, invisibly.

Author(s)

adapted by Fabian Mueller from RnBeads code by Yassen Assenov

See Also

Report for other functions adding contents to an HTML report

Examples

```
report <- createReport("example.html", "Example", init.configuration = TRUE)
recipe <- list("Sift flour in a bowl", "Add sugar and mix", "Add milk and mix")
addReportList(report, recipe, type="o")
```

---

addReportParagraph *addReportParagraph*

Description

Generates HTML code for a new paragraph in the specified report.

Usage

```
addReportParagraph(report, txt, pClass = NULL)
```

Arguments

report	Report to write the text to.
txt	character vector (or array) storing the text to be written. The elements of this vector are concatenated without a separator.
pClass	CSS class definition of the paragraph. This must be either NULL (default) or one of: "centered" This paragraph gives a formula or a short statement. Text is horizontally centered. "note" This paragraph describes a note. Text is italic. "task" This paragraph describes a task. Text is bold and bright red.

Value

The modified report, invisibly.

Author(s)

adapted by Fabian Mueller from RnBeads code by Yassen Assenov

See Also

Report for other functions adding contents to an HTML report

Examples

```
report <- createReport("example.html", "Example", init.configuration = TRUE)
txt <- "A pessimist is a person who has had to listen to too many optimists."
txt <- c(txt, " <i>Don Marquis</i>")
addReportParagraph(report, txt)
```

addReportReference *addReportReference*

Description

Adds a reference item to the given report.

Usage

```
addReportReference(report, txt)
```

Arguments

report	Report to add a reference item to.
txt	Text of the reference in the form of a non-empty character vector. The elements of this vector are concatenated without a separator.

Value

The modified report.

Author(s)

adapted by Fabian Mueller from RnBeads code by Yassen Assenov

See Also

getReportReference for adding citations in the report's text; Report for other functions adding contents to an HTML report

Examples

```
report <- createReport("example.html", "Example", init.configuration = TRUE)
txt.reference <- c("Bird A. ", "<i>Nucleic Acids Res.</i> <b>8</b> (1980)")
report <- addReportReference(report, txt.reference)
txt <- c("This was shown in ", getReportReference(report, txt.reference), ".")
addReportParagraph(report, txt)
```

addReportSection *addReportSection*

Description

Generates HTML code for a new section in the specified report.

Usage

```
addReportSection(report, title, description, level = 1L,
  collapsed = FALSE)
```

Arguments

report	Report to write the text to.
title	Section header. This must be a single-element character vector.
description	Human-readable paragraph text of the section in the form of a character vector. Elements of this vector are concatenated without a separator to form the full description. Set this to NULL if the section does not (yet) contain text.
level	Section level as a single integer. It must be one of 1, 2 or 3, denoting section, subsection and sub-subsection, respectively.
collapsed	Flag indicating if the contents of this section is to be initially collapsed. Possible values are TRUE (the section is not visible), FALSE (default, the section is expanded) and "never" (the section cannot be collapsed or expanded).

Value

The modified report.

Author(s)

adapted by Fabian Mueller from RnBeads code by Yassen Assenov

See Also

Report for other functions adding contents to an HTML report

Examples

```
report <- createReport("example.html", "Example", init.configuration = TRUE)
report <- addReportSection(report, "Introduction", "This is how it's done.")
```

addReportTable *addReportTable*

Description

Generates HTML code for a table in the specified report.

Usage

```
addReportTable(report, tdata, row.names = TRUE,
  first.col.header = FALSE, indent = 0, tag.attrs = c(class =
  "tabdata"), thead = NULL, tcaption = NULL,
  na = "<span class=\"disabled\">n/a</span>")
```

Arguments

report	Report to write the text to.
tdata	Matrix or data frame to be presented in HTML form. Column names, if present, are used to define table columns. If this table contains 0 (zero) rows or 0 columns, calling this function has no effect.
row.names	Flag indicating if row names should also be printed. If this parameter is TRUE and tdata defines row names, these are printed in the left-most column and are displayed as header cells. Keep in mind that data.frames always define row names.
first.col.header	Flag indicating if all cells in the first column must be displayed as header cells. Note that, if both this parameter and row.names are TRUE, and tdata contains row names, the constructed HTML table will have 2 columns of header cells.
indent	Default indentation, in number of tabulation characters, to apply to HTML tags. This indentation is also applied to thead.
tag.attrs	Named character vector specifying the list of attributes to be set to the <table> element. Setting this to NULL or an empty character vector disables attributes.
thead	character vector storing a table header to include. This can, for example, be a character that defines column widths. Every element in this vector is written on a separate line, applying the indentation given by indent.
tcaption	Text to include as a caption below the table, or NULL if the table does not contain caption.
na	character to be used for printing NA values in the table. This parameter is not considered when printing thead or the table's column names.

Value

The modified report, invisibly.

Author(s)

adapted by Fabian Mueller from RnBeads code by Yassen Assenov

See Also

addReportTables for adding a listing of tables; Report for other functions adding contents to an HTML report

---

addReportTables *addReportTables*

Description

Generates HTML code for a listing of tables (of which only one is visible at any moment) in the specified report.

Usage

```
addReportTables(report, tables, setting.names, selected.table = 1L,
  indent = 2L, ...)
```

Arguments

report	Report to write the text to.
tables	Non-empty list of tables, each one represented by a data.frame or matrix. The names of this list are used as table identifiers; each one consists of elements separated by underscore character (_).
setting.names	List of table name element descriptors. Every variable elements in the table names must be included in this list.
selected.table	Index of the table to be initially selected in this listing.
indent	Default indentation, in number of tabulation characters, to apply to every table.
...	Other parameters passed to addReportTable.

Value

The modified report.

Author(s)

adapted by Fabian Mueller from RnBeads code by Yassen Assenov

See Also

addReportTable for adding a single table to a report; Report for other functions adding contents to an HTML report

createReport *createReport*

Description

Creates a new report object.

Usage

```
createReport(fname, title, page.title = "muReportR report",
  authors = NULL, dirs = NULL, init.configuration = FALSE,
  theme = "thesis")
```

Arguments

fname	Single-element character vector denoting the name of the file to contain the HTML report. If this file already exists, it will be overwritten.
title	Title of the report in the form of a single-element character vector.
page.title	Web page title. This usually appears in the web browser's window title when the report is open. If specified, this must be a vector. Note that only the first element is used.
authors	Optional list of authors in the form of a character vector. This list is included in the header of the generated HTML file. Note that author names can contain only Latin leters, space, dash (-), comma (,) or dot (.).
dirs	Location of the supporting directories, that is, paths that are expected to contain additional files linked to from the HTML report. See the *Details* section for a list of these directories.
init.configuration	Flag indicating if the report configuration data should be initialized. If this parameter is TRUE, the method creates the respective directory and copies configuration files that define cascading style sheet (CSS) definitions and Javascript functions used by the HTML report. If such configuration files already exist, they will be overwritten. Since the aforementioned files can be shared by multiple reports, it is recommended that the configuration is initialized using the method initializeReports, instead of setting this flag to TRUE.
theme	character specifying the theme to be used for the report files. Currently only "thesis" (default) and "stanford" are supported.

Details

If specified, the parameter dirs must be a character vector. The following names are read:

- "configuration" Directory that contains the auxilliary configuration files, such as style sheets and Javascript files. If missing or NA, the default value used is "configuration".

- "data" Directory to contain the tables, lists and other generated data files that are linked to in the HTML report. If missing or NA, the value used is formed from the file name fname (without the extension) and the suffix "_data".
- "pngs" Directory to contain the low resolution PNG images shown in the HTML report. If missing or NA, the value used is formed from the file name fname (without the extension) and the suffix "_images".
- "pdfs" Directory to contain the PDF images (if such are created). If not missing or NA, the value used is formed from the file name fname (without the extension) and the suffix "_pdf".
- "high" Directory to contain the high resolution PNG images (if such are created). If missing or NA, the value used is the same as the pngs directory.

Any other elements, if present, are ignored. Note that these directories are not required to point to different locations. In particular, if the directories for low and for high resolution images are identical, the high-resolution image files are assumed to be the ones with suffix "_highres.png". See createReportPlot for creating image files. In order to ensure independence of the operating system, there are strong restrictions on the names of the file and directories. The name of the report's HTML file can consist of the following symbols only: Latin letters, digits, dot (.), dash (-) and underline (_). The extension of the report's HTML file must be one of htm, html, xhtml or xml. The supporting directories must be given as relative paths; the restrictions on the path names are identical to the ones for file name. Forward slash (/) is to be used as path separator. Path names cannot start or end with a slash. None of the directory names can be an empty string, use "." instead. A value in the form "mypath/.html" for fname is invalid. Upon initialization, the report attempts to create or overwrite the specified fname. If the path to it does not exist, or if the current process does not have permissions to write to the file, report initialization will fail. The report object visits each supporting directory (except configuration) and attempts to create it, unless it is an existing empty directory. Report initialization will fail if any of the visited directories does not meet the criteria and could not be created. Hidden files (file names starting with "." on Unix platforms) are ignored. Thus, all supporting directories that already exist and contain hidden files only are considered valid.

Value

Newly created Report object.

Author(s)

adapted by Fabian Mueller from RnBeads code by Yassen Assenov

See Also

Report for functions adding contents to an HTML report

Examples

```
report <- createReport("example.html", "Example", init.configuration = TRUE)
```

createReportGgPlot *createReportGgPlot*

Description

creates a report plot containing a ggplot object. Except for the ggp parameter, the signature and behavior is identical to createReportPlot.

Usage

```
createReportGgPlot(ggp, fname, report = NULL, width = 7, height = 7,
  create.pdf = TRUE, low.png = as.integer(100),
  high.png = as.integer(0))
```

Arguments

ggp	ggplot object to be plotted
fname	character vector with one element storing the name of the output file, without the extension. The initialized object appends .pdf and/or .png to this name.
report	Report (object of type Report) to which this plot is going to be added. This is used to set the directories for PDF and/or PNG files generated for these plots. If this parameter is NULL, the current working directory is used to host all generated images.
width	numeric storing the width of the device in inches. The length of this vector must be 1.
height	numeric storing the height of the device in inches. The length of this vector must be 1.
create.pdf	Flag indicating if a PDF image is to be created. The length of this vector must be 1.
low.png	Resolution, in dots per inch, used for the figure image. Set this to 0 or a negative value to disable the creation of a low resolution image. The length of this vector must be 1.
high.png	Resolution, in dots per inch, used for a dedicated image. Set this to 0 or a negative value to disable the creation of a high resolution image. The length of this vector must be 1.

Value

Newly created ReportGgPlot object.

Author(s)

Fabian Mueller

createReportPlot *createReportPlot*

Description

Initializes a report plot and opens a device to create it. The type of the device created depends on the parameters `create.pdf`, `low.png` and `high.png`. If `create.pdf` is `TRUE`, a PDF device is opened and its contents are later copied to PNG device(s) if needed. Otherwise, a PNG device is opened. Note that at least one of the following conditions must be met:

- `create.pdf == TRUE`
- `low.png > 0`
- `high.png > 0`

Usage

```
createReportPlot(fname, report = NULL, width = 7, height = 7,
  create.pdf = TRUE, low.png = 100L, high.png = 0L)
```

Arguments

`fname`	`character` vector with one element storing the name of the output file, without the extension. The initialized object appends `.pdf` and/or `.png` to this name.
`report`	Report (object of type `Report`) to which this plot is going to be added. This is used to set the directories for PDF and/or PNG files generated for these plots. If this parameter is `NULL`, the current working directory is used to host all generated images.
`width`	`numeric` storing the width of the device in inches. The length of this vector must be `1`.
`height`	`numeric` storing the height of the device in inches. The length of this vector must be `1`.
`create.pdf`	Flag indicating if a PDF image is to be created. The length of this vector must be `1`.
`low.png`	Resolution, in dots per inch, used for the figure image. Set this to `0` or a negative value to disable the creation of a low resolution image. The length of this vector must be `1`.
`high.png`	Resolution, in dots per inch, used for a dedicated image. Set this to `0` or a negative value to disable the creation of a high resolution image. The length of this vector must be `1`.

Details

In order to ensure independence of the operating system, there are strong restrictions on the name of the file. It can consist of the following symbols only: Latin letters, digits, dot (`.`), dash (`-`) and underline (`_`). The name must not include paths, that is, slash (`/`) or backslash (`\`) cannot be used.

Value

Newly created `ReportPlot` object.

Author(s)

adapted by Fabian Mueller from RnBeads code by Yassen Assenov

See Also

`pdf` for manually initializing a graphics device; `Report` for other functions adding contents to an HTML report

Examples

```
plot.image <- createReportPlot('scatterplot_tumors')
plot(x = c(0.4, 1), y = c(9, 3), type = 'p', main = NA, xlab = expression(beta), ylab = 'Measure')
off(plot.image)
```

---

getReportDir *getReportDir*

Description

Gets the location of the given report-specific directory.

Usage

```
getReportDir(report, dir = c("data", "images", "images_highres", "pdfs"),
  absolute = FALSE)
```

Arguments

`report`	Report of interest.
`dir`	Type of directory to get. Must be one of `"data"`, `"images"`, `"images_highres"` or `"pdfs"`.
`absolute`	Flag indicating if the absolute path of the directory is to be returned. If this is `FALSE`, the directory name is returned relative to the report's HTML file location.

Value

Path of the requested directory as a single-element `character` vector.

Author(s)

adapted by Fabian Mueller from RnBeads code by Yassen Assenov

See Also

Report for functions adding contents to an HTML report

Examples

```
report <- createReport("example.html", "Example", init.configuration = TRUE)
getReportDir(report, "data")
```

---

getReportPlotFiles *getReportPlotFiles*

Description

Gets the list of all files that are planned to be generated, or were already generated by the given report plot.

Usage

```
getReportPlotFiles(repPlot)
```

Arguments

repPlot	Report plot of interest. This must be an object of type ReportPlot.

Value

Non-empty character vector of absolute file names.

Author(s)

adapted by Fabian Mueller from RnBeads code by Yassen Assenov

Examples

```
plot.image <- createReportPlot('scatterplot', high.png = 200)
getReportPlotFiles(plot.image)
```

getReportReference *getReportReference*

Description

Creates a string that points to the given reference item in the specified report.

Usage

```
getReportReference(report, txt)
```

Arguments

report	Report that contains the reference to be cited.
txt	Text of the reference in the form of a non-empty character vector. This reference must already added to the report.

Value

Citation of the reference item (including a link) in the form of a one-element character vector. If the specified reference item is not found in the report, this method returns an empty string.

Author(s)

adapted by Fabian Mueller from RnBeads code by Yassen Assenov

See Also

addReportReference for adding a reference item to a report; Report for other functions adding contents to an HTML report

Examples

```
report <- createReport("example.html", "Example", init.configuration = TRUE)
txt.reference <- c("Bird A. ", "<i>Nucleic Acids Res.</i> <b>8</b> (1980)")
report <- addReportReference(report, txt.reference)
txt <- c("This was shown in ", getReportReference(report, txt.reference), ".")
addReportParagraph(report, txt)
```

ggMsgPlot *ggMsgPlot*

Description

Creates a plot, using **ggplot2**, with a single text message.

Usage

```
ggMsgPlot(txt)
```

Arguments

txt	Text to be plotted.

Value

The newly initialized ggplot instance.

Author(s)

adapted by Fabian Mueller from RnBeads code by Yassen Assenov

Examples

```
x11(width = 5, height = 5)
ggMsgPlot("Missing data")
```

initializeReports *initializeReports*

Description

Creates a new directory to host HTML reports and copies the shared configuration files.

Usage

```
initializeReports(reportDir, configDir = "_config", theme = "thesis")
```

Arguments

reportDir	Directory to host report files. This must be a character of length one that specifies a non-existent path, as this methods attempts to create it.
configDir	Subdirectory to host configuration files shared by the reports. This must be a character of length one that gives location as a path relative to reportDir. Also, strong restrictions apply to the path name. See the description of the createReport function for more details. This method creates the directory and copies configuration files that define cascading style sheet (CSS) definitions and Javascript functions used by the HTML reports.
theme	character specifying the theme to be used for the report files. Currently only "thesis" (default) and "stanford" are supported.

Value

TRUE if the report directory was successfully created and the configuration files were copied to the specified location; FALSE otherwise.

Author(s)

adapted by Fabian Mueller from RnBeads code by Yassen Assenov

See Also

createReport for initializing an HTML report

Examples

```
reportDir <- "~/infinium_studies/cancer_study/reports"
if (!initializeReports(reportDir)) {
    cat("ERROR: Could not initialize configuration in ", reportDir, "\n", sep = "")
}
```

---

muReportR *muReportR: Generating analysis reports in R.*

Description

Generate HTML reports from R. Based on code from the RnBeads package.

off,Report-method *off-methods*

Description

Performs cleanup and/or other finishing activities and closes the specified device, connection, or document.

Usage

```
## S4 method for signature 'Report'
off(.Object)

## S4 method for signature 'ReportPlot'
off(.Object)

## S4 method for signature 'ReportGgPlot'
off(.Object, handle.errors = FALSE)
```

Arguments

.Object	Object to be closed.
handle.errors	Flag indicating if the method should attempt to catch and process errors (e.g. I/O errors) internally. Setting this to TRUE does not guarantee that the method never stops with an error.

Value

The closed object, invisibly.

Report-class *Report Class*

Description

Handler of a generated HTML report. Reports are initialized using the function createReport.

Slots

fname Name of the file that contains the HTML report.

dir.conf Directory that contains configuration files; usually shared between reports.

dir.data Directory that contains the generated external lists and tables.

dir.pngs Directory that contains the generated figure image files.

dir.pdfs Directory that contains the generated figure PDF files.

`dir.high` Directory that contains the generated high-resolution image file.

`sections` Number of sections and subsections currently added to the report.

`opensections` Indices of currently active section and subsections.

`figures` Number of figures currently added to the report.

`tables` Number of selectable tables added to the report.

`references` List of references to be added at the end of the report.

Methods and Functions

`getReportDir` Gets the location of a given report-specific directory.

`addReportSection` Generates HTML code for a new section in the report.

`addReportParagraph` Generates HTML code for a new paragraph in the report.

`addReportList` Generates HTML code for a list in the report.

`addReportTable` Generates HTML code for a table in the report.

`addReportTables` Generates HTML code for a listing of tables in the report.

`addReportFigure` Generates HTML code for a figure in the report.

`addReportReference` Adds a reference item to the report.

`off` Completes the HTML report by adding a reference section (if needed), a footer notice and closing the <body> and <html> tags.

Author(s)

adapted by Fabian Mueller from RnBeads code by Yassen Assenov

---

ReportGgPlot-class *ReportGgPlot Class*

Description

Information about the files created to store one generated plot in a report. Report plots are initialized using the function `createReportGgPlot`. It inherits from the `ReportPlot` class and handling is analogous, except that it contains an additional slot to store a `ggplot` object.

Slots

`ggp` `ggplot` object to be printed

Notes

No device is being opened until `off(reportGgPlot)` is called.

Author(s)

Fabian Mueller

ReportPlot-class *ReportPlot Class*

Description

Information about the files created to store one generated plot in a report. Report plots are initialized using the function createReportPlot.

Slots

`fname` Relative file name. It does not include path or extension.

`width` Width of the image in inches.

`height` Height of the image in inches.

`create.pdf` Flag indicating if a PDF image is created.

`low.png` Resolution, in dots per inch, used for the figure image.

`high.png` Resolution, in dots per inch, used for the high-resolution image.

`dir.pdf` Directory that contains the generated PDF file.

`dir.png.low` Directory that contains the generated figure image file.

`dir.png.high` Directory that contains the generated high-resolution image file.

Methods and Functions

`getReportPlotFiles` Gets the list of all files that are planned to be generated, or were already generated by the report plot.

`off` Copies the figure to a PNG file (if needed) and closes the device associated with the report plot.

Author(s)

adapted by Fabian Mueller from RnBeads code by Yassen Assenov

Index