Package 'exsic'

February 19, 2015

Type Package

Title Convenience Functions for Botanists to Create Specimens Indices

Version 1.1.1

Date 2014-10-01

Maintainer Reinhard Simon <R.SIMON@CGIAR.ORG>

Description The package provides tools for botanists, plant taxonomists, curators of plant genebanks and perhaps other biological collections.

License MIT + file LICENSE

Depends R (>= 2.15.1), stringr, markdown

Author Reinhard Simon [aut, cre] (CIP),
David M. Spooner [aut] (USDA),
International Potato Center [cph]

NeedsCompilation no

Repository CRAN

Date/Publication 2014-10-01 18:44:48

R topics documented:

`exsic-package` *Provides botanists with convenience functions to create exsiccatae indices*

Description

The tool allows creating simple specimen indices as found in taxonomic treatments based on a table of specimen records. An example file of tabulated speciment data is provided. In addition, four different exsiccatae styles are provided. The naming of the columns in the specimen table follows largely the conventions used in the BRAHMS software package. Each specimen record must at least have content in the following nine fields: id, genus, species, collcite, number, colldate, country, majorarea, minorarea. If not present, the fields are added and filled with dummy values like 's.d.' for no date or 'Unknown country/area'. Highly recommended fields include: collector, addcoll. Optional fields include: locnotes, phenology, elevation, latitude, longitude, and dups The produced indices will sort countries and species alphabetically. Within a country records will be sorted alphabetically by 'majorarea' (if present) and by collector and collecting nunber. A web page in standard html format is created based on a template. The template may be changed and specified in most word processing software. The package provides one main function 'exsic'. See the example in this section on how to access it.

Author(s)

Reinhard Simon, David M. Spooner

Examples

```
# Example
load(system.file("data/config.rda", package="exsic"))
###########################################################
# This runs the example file


# Read input file
df = system.file("samples/exsic.csv", package="exsic")
# read only first 10 records
data = read.exsic(df)[1:10,]

# Prepare output file
```

```
td = tempdir()
of = file.path(td,"out.html")


# Example 1: mostly default parameters
# Prepare exsiccatae indices
exsic(data, html = of)


# Example 2: using another format
of = file.path(td,"out_PK.html")
exsic(data, html = of, format = format.PK)
```

`coll.cite` *Constructs a final citation of collector names.*

Description

Uses two of the BRAHMS fields (collector, addcoll). Expects names in both fields to conform to the pattern: "Lastname, A" where A are the list of initials without dots; if no comma present the function assummes all words to be part of the lastname like in: van de Bergh.

Usage

```
coll.cite(collector, addcoll = "", initials = c("none", "before"),
  dots = TRUE)
```

Arguments

`collector`	a name
`addcoll`	a name or list of names; may be empty
`initials`	use 'none' or 'before' to indicate if initials should be used for the citation
`dots`	boolean; should dots be used to separate the initials?

Details

The 'collector' field must have only one name; the 'addcoll' field may have several names separated by ; or it may be empty. Like in the case of 'collector' a missing comma indicates that all words till the next ; are part of the last name.

Two collector names will be separated by '&'; more than two names will result in a citation with just the (principal) collectors name followed by 'et al.'.

Value

a text

Note

The function does not yet handle any other special cases like including an 'obligatory' second author as permitted in some specimen citations when the author of the treatment is also a secondary collector.

In any case, using this function or editing the final citation manually, the final citation should be in a column called 'collcite'.

Author(s)

Reinhard Simon

See Also

Other helper: `is.exsic`; `is.format`; `is.sortfilter`; `strip.final.dot`

`exsic` *Creates three botanical indices (exsiccatae or index of specimens; numerical list of species; index to numbered collections).*

Description

It uses a data.frame expecting a minimum set of columns; if those are not found they will be added and filled with 'unknown' or similar values as will be other columns with missing data but used for sorting the final indices.

Usage

```
exsic(data, file = NULL, html = "exsic.html", sortfilter = NULL,
  format = format.SBMG, headers = c("Citations of Specimens",
  "Numerical List of Species", "Index to Numbered Collections",
  "*Numbers refer to species in the Numerical List.*\n\r"))
```

Arguments

`data`	a table in exsic format
`file`	Path to the .csv file containing the database
`html`	Path to the resulting .html file
`sortfilter`	a table containing filters for country and species
`format`	a table containing format options for elements in the exsiccatae index
`headers`	A list of text lines for labeling the indices

Value

boolean TRUE if all steps executed successfully

Author(s)

Reinhard Simon

See Also

Other main: index.citations; index.collections; index.countries; index.species.short; index.species; prepare.table; read.exsic; section.exsic; write.exsic

Examples

```
# Example
load(system.file("data/config.rda", package="exsic"))
############################################################
# This runs the example file


# Read input file
df = system.file("samples/exsic.csv", package="exsic")
# read only first 10 records
data = read.exsic(df)[1:10,]

# Prepare output file
td = tempdir()
of = file.path(td,"out.html")


# Example 1: mostly default parameters
# Prepare exsiccatae indices
exsic(data, html = of)


# Example 2: using another format
of = file.path(td,"out_PK.html")
exsic(data, html = of, format = format.PK)
```

---

format.ASPT *A formatting scheme following ASPT (American Society of Plant Taxonomists) conventions for an exsiccatae record.*

Description

A formatting scheme following ASPT (American Society of Plant Taxonomists) conventions for an exsiccatae record.

Format

A dataframe with 9 entries and three variables

- "field" A column name present in the exsiccatae table.
- "style" The text style applied to that element. One of: none, bold, italics, capitals, underline or underscore, uppercase, and () or parentheses
- "sept" A seperator following the element. One of ”, ’ ’, ’, ’, ’. ’, ’: ’ and ’;’
- "comments"

Author(s)

R. Simon

Source

http://www.aspt.net/publications/sysbot/checklist_systbot.php

See Also

Other formats: format.NYBG; format.PK; format.SBMG

`format.NYBG` *A formatting scheme following NYBG (New York Botanical Garden) conventions for an exsiccatae record.*

Description

In comparison to the two other included specimen citation formats this includes an additional field to handle phenological information at the time of observation.

Format

A dataframe with 10 entries and three variables

- "field" A column name present in the exsiccatae table.
- "style" The text style applied to that element. One of: none, bold, italics, capitals, underline or underscore, uppercase, and () or parentheses
- "sept" A seperator following the element. One of ”, ’ ’, ’, ’, ’. ’, ’: ’ and ’;’
- "comments"

Author(s)

R. Simon

Source

http://www.nybg.org/botany/ofn/fn-gd3.htm

See Also

Other formats: format.ASPT; format.PK; format.SBMG

---

format.PK — *A formatting scheme following PK (Phytokeys) conventions for an exsiccatae record.*

Description

A formatting scheme following PK (Phytokeys) conventions for an exsiccatae record.

Format

A dataframe with 9 entries and three variables

- "field" A column name present in the exsiccatae table.
- "style" The text style applied to that element. One of: none, bold, italics, capitals, underline or underscore, uppercase, and () or parentheses
- "sept" A seperator following the element. One of ”, ’ ’, ’, ’, ’. ’, ’: ’ and ’;’
- "comments"

Author(s)

R. Simon

Source

http://www.pensoft.net/journals/phytokeys/about/AuthorGuidelines

See Also

Other formats: format.ASPT; format.NYBG; format.SBMG

---

format.SBMG — *A formatting scheme following SBMG (Systematic Botany Monographs) conventions for an exsiccatae record.*

Description

A formatting scheme following SBMG (Systematic Botany Monographs) conventions for an exsiccatae record.

Format

A dataframe with 9 entries and three variables

- "field" A column name present in the exsiccatae table.
- "style" The text style applied to that element. One of: none, bold, italics, capitals, underline or underscore, uppercase, and () or parentheses
- "sept" A seperator following the element. One of ”, ’ ’, ’, ’, ’. ’, ’: ’ and ’;’
- "comments"

Author(s)

R. Simon

See Also

Other formats: `format.ASPT`; `format.NYBG`; `format.PK`

---

index.citations *Creates an index of specimen citations*

Description

Creates an index of specimen citations

Usage

```
index.citations(atable, format = format.SBMG)
```

Arguments

`atable`	a table with exsiccatae records
`format`	a data frame containing formatting options

Value

a string in markdown format

Author(s)

Reinhard Simon

See Also

Other main: `exsic`; `index.collections`; `index.countries`; `index.species.short`; `index.species`; `prepare.table`; `read.exsic`; `section.exsic`; `write.exsic`

index.collections *Creates the index of collectors and their specimes.*

Description

Should only be used within the template file.

Usage

```
index.collections(atable = NULL, format = format.SBMG)
```

Arguments

atable	a table with exsiccatae records
format	a format.exsic table (check with is.format)

Value

a string in markdown format

Author(s)

Reinhard Simon

See Also

Other main: exsic; index.citations; index.countries; index.species.short; index.species; prepare.table; read.exsic; section.exsic; write.exsic

index.countries *Creates a simple index of countries.*

Description

Creates a simple index of countries.

Usage

```
index.countries(atable = NULL)
```

Arguments

atable	a table with exsiccatae records

Value

a string in markdown format

Author(s)

Reinhard Simon

See Also

Other main: exsic; index.citations; index.collections; index.species.short; index.species; prepare.table; read.exsic; section.exsic; write.exsic

---

index.species *Creates a simple index of species.*

Description

Creates a simple index of species.

Usage

```
index.species(atable = NULL)
```

Arguments

atable	a table with exsiccatae records

Value

a string in markdown format

Author(s)

Reinhard Simon

See Also

Other main: exsic; index.citations; index.collections; index.countries; index.species.short; prepare.table; read.exsic; section.exsic; write.exsic

index.species.short *A condensed list of species (listed within a line).*

Description

Should only be used within the template file.

Usage

```
index.species.short(atable = NULL)
```

Arguments

atable	a table with exsiccatae records

Value

a string in markdown format

Author(s)

Reinhard Simon

See Also

Other main: exsic; index.citations; index.collections; index.countries; index.species; prepare.table; read.exsic; section.exsic; write.exsic

is.exsic *Checks table format*

Description

Checks if the table is in the expected format by the principal exsic function.

Usage

```
is.exsic(obj)
```

Arguments

obj	an object

Details

The format should be created either using reading a table via read.exsic or prepare.table.

Author(s)

Reinhard Simon

See Also

Other helper: coll.cite; is.format; is.sortfilter; strip.final.dot

---

`is.format` *Checks if the data frame conforms to the expectations to a formatting configuration data frame.*

Description

Checks if the data frame conforms to the expectations to a formatting configuration data frame.

Usage

```
is.format(fmt)
```

Arguments

`fmt`	a special data frame

Value

boolean TRUE if ok

Author(s)

Reinhard Simon

See Also

format.SBMG, format.NYBG

Other helper: coll.cite; is.exsic; is.sortfilter; strip.final.dot

is.sortfilter *Checks if the data frame conforms to the expectations to sort and filter the exsic database.*

Description

Checks if the data frame conforms to the expectations to sort and filter the exsic database.

Usage

```
is.sortfilter(sf)
```

Arguments

sf	a sortfilter data frame

Value

boolean TRUE if ok

Author(s)

Reinhard Simon

See Also

Other helper: coll.cite; is.exsic; is.format; strip.final.dot

potato *Wild potato specimen sample list*

Description

Wild potato specimen sample list

Format

Contains 1000 observations of 16 variables. The variable names conform largely to the BRAHMS standard.

- "genus" Genus name
- "species" Species name

Note

Access the data using system.file("samples/exsic.csv", package = "exsic")

Author(s)

David M. Spooner with format edits from R. Simon

`prepare.table` *Prepare exsic table*

Description

Complements missing minimal columns and missing content. If applicable (not null) applies a list of countries and species in the given order to sort the table. Countries and species not present in a given sortfilter data.frame will be filtered out from the underlying table.

Usage

```
prepare.table(atbl, sortfilter = NULL)
```

Arguments

`atbl`	a data.frame table
`sortfilter`	a data frame with two columns (one for species and one for genus; maybe NULL)

Value

a table in the format expected by the exsic function

Author(s)

Reinhard Simon

See Also

Other main: `exsic`; `index.citations`; `index.collections`; `index.countries`; `index.species.short`; `index.species`; `read.exsic`; `section.exsic`; `write.exsic`

read.exsic *Reads a table in .csv format.*

Description

The table should have a certain set of minimum columns following the conventions of the BRAHMS software.

Usage

```
read.exsic(file, sortfilter = NULL)
```

Arguments

file	a file path
sortfilter	a dataframe containing instructions on how to sort and filter the table

Details

The function tries to be forgiving and makes a compatible table even if none of the found columns is compliant with the expected names.

Value

a special dataframe conforming to the exsic expectations

Author(s)

Reinhard Simon

See Also

Other main: exsic; index.citations; index.collections; index.countries; index.species.short; index.species; prepare.table; section.exsic; write.exsic

section.exsic *Creates a section header*

Description

A simple wrapper to return a section header formatted for inclusion in the final document.

Usage

```
section.exsic(title = "My title")
```

Arguments

`title`	a section title

Value

a string in markdown format

Author(s)

Reinhard Simon

See Also

Other main: `exsic`; `index.citations`; `index.collections`; `index.countries`; `index.species.short`; `index.species`; `prepare.table`; `read.exsic`; `write.exsic`

---

sort.specs *A sorting and filtering configuration data.frame*

Description

A sorting and filtering configuration data.frame

Author(s)

R. Simon

---

strip.final.dot *Strips a text of any final dots.*

Description

This is a small helper function to prepare the final exsic table.

Usage

```
strip.final.dot(txt = "")
```

Arguments

`txt`	a text string

Details

Intended for eliminating any duplicated dots in the final document due to final dots usually in the 'locnotes' (location notes) column.

Value

a text string without final dot

Author(s)

Reinhard Simon

See Also

Other helper: coll.cite; is.exsic; is.format; is.sortfilter

---

write.exsic *Writes the exsiccatae text*

Description

Writes the text string into a file formatted as HTML.

Usage

```
write.exsic(text, file)
```

Arguments

text	a text in markdown format
file	a file name

Author(s)

Reinhard Simon

See Also

Other main: exsic; index.citations; index.collections; index.countries; index.species.short; index.species; prepare.table; read.exsic; section.exsic

Index