

Package ‘febr’

October 13, 2022

Type Package

Title Data Repository of the Brazilian Soil

Version 1.9.9

Date 2022-02-28

Description Utilities to access and process data from the
Data Repository of the Brazilian Soil
<<https://www.pedometria.org/febr/>>.

License GPL (>= 2)

Encoding UTF-8

Imports data.table, stats, utils

Suggests curl, rmarkdown, jsonlite, sf, stringr, knitr

VignetteBuilder knitr

RoxygenNote 7.1.2

URL <https://github.com/laboratorio-de-pedometria/febr-package/>

BugReports <https://github.com/laboratorio-de-pedometria/febr-package/issues/>

Language en-US

NeedsCompilation no

Author Alessandro Samuel-Rosa [aut, cre]
(<<https://orcid.org/0000-0003-0877-1320>>),
Taciara Zborowski Horst [aut] (<<https://orcid.org/0000-0002-1135-028X>>),
Glauber José Vaz [ctb] (<<https://orcid.org/0000-0002-4527-5150>>),
Universidade Federal de Santa Maria [fnd],
Universidade Tecnológica Federal do Paraná [fnd]

Maintainer Alessandro Samuel-Rosa <alessandrosamuelrosa@gmail.com>

Repository CRAN

Date/Publication 2022-02-28 19:50:02 UTC

R topics documented:

dataset	2
dictionary	3
febr2sf	4
febr2sse	5
goto	6
identification	8
layer	9
metadata	12
morphology	13
observation	14
readFEBR	17
readIndex	18
readVocabulary	18
taxonomy	19
unit	20
Index	22

dataset	<i>Defunct functions in the febr package</i>
---------	---

Description

The functions listed here are no longer part of the **febr** package as they are not needed (any more).

Usage

```
dataset(...)
```

```
febr(...)
```

```
febr2spdf(...)
```

```
febr2xlsx(...)
```

```
header(...)
```

```
standard(...)
```

Arguments

... Not used.

Value

No return value, called for side effects.

 dictionary

Access the FEBR dictionary

Description

Download definitions from the dictionary of the [Data Repository of the Brazilian Soil](#). For each field, the dictionary includes an identification code and name, the standard measurement unit, recommended number of decimal places, type of data, and description of the respective analytical method. The dictionary is used to standardize the data contained in a dataset.

Usage

```
dictionary(table, variable, unit, precision, active = NULL)
```

Arguments

table	(optional) Character vector indicating one or more table IDs that should be used to filter the dictionary. Accepted values: "metadado", "versionamento", "observacao", and "camada".
variable	(optional) Character vector indicating one or more variables that should be used to filter the dictionary. Accepts both specific identification codes, e.g. "ferro_oxalato_icpoes", as well as general identification codes, e.g. "ferro".
unit	(optional) Character vector indicating one or more measurement units that should be used to filter the dictionary. For example, "g/kg", "g/cm ³ ", and "cmolc/kg".
precision	(optional) Integer vector indicating one or more number of decimal places that should be used to filter the dictionary.
active	(optional) Logical value indicating whether active (active = TRUE), inactive (active = FALSE) or any (active = NULL, default) field should be returned.

Value

An object of class `data.frame` with definitions for the selected fields.

Author(s)

Alessandro Samuel-Rosa <alessandrosamuelrosa@gmail.com>

References

Teixeira, P. C., Donagemma, G. K., Fontana, A., Teixeira, W. G. (2017) *Manual de Métodos de Análise de Solo*. Brasília: Embrapa.

See Also

The FEBR dictionary at <https://docs.google.com/spreadsheets/d/1Da1qi5JbW4fg9oNkXw5TykZTA39pR5GezapVeV01>

Examples

```
res <- dictionary(variable = "ferro")
head(res)
```

febr2sf

Create an sf object

Description

Set spatial coordinates and coordinate reference system (CRS) to a set of soil observations.

Usage

```
febr2sf(obj)
```

Arguments

obj Object of class `data.frame` downloaded from the Data Repository of the Brazilian Soil using `observation()`.

Details

Create an sf object from the *observation* ("observacao") table of one or more standardized datasets contained in the FEBR Soil Data Repository, <https://www.pedometria.org/febr/>.

Value

An object of class `sfc_POINT` (single soil observation) or `sfc_MULTIPOINT` (multiple soil observations).

Author(s)

Alessandro Samuel-Rosa <alessandrosamuelrosa@gmail.com>

References

Pebesma, E., 2018. Simple Features for R: Standardized Support for Spatial Vector Data. The R Journal 10 (1), 439-446, doi: [10.32614/RJ2018009](https://doi.org/10.32614/RJ2018009)

Examples

```
if (interactive()) {
  res <- observation(
    data.set = "ctb0013",
    progress = FALSE, verbose = FALSE)
  res <- febr2sf(obj = res)
  plot(res["geometry"], axes = TRUE, graticule = TRUE)
}
```

febr2sse	<i>Conversion between FEBR and SmartSolos Expert (SSE) soil profile data formats</i>
----------	--

Description

Export FEBR soil profile data to the JSON file format required by the SmartSolos Expert API.

Usage

```
febr2sse(profiles, horizons, file, ...)
```

Arguments

profiles	Data frame with soil profile data, i.e. observation locations.
horizons	Data frame with soil horizon data, i.e. sampling layers.
file	(optional) Character string naming the JSON file to be written to disk.
...	(optional) Arguments passed to <code>base::writeLines()</code> .

Value

An object of class character containing a unicode JSON string.

Author(s)

Alessandro Samuel-Rosa <alessandrosamuelrosa@gmail.com>

References

Jeroen Ooms (2014). The jsonlite Package: A Practical and Consistent Mapping Between JSON Data and R Objects. arXiv:1403.2805 [stat.CO] URL <https://arxiv.org/abs/1403.2805>.

Examples

```
# Toy example
pro <- data.frame(
  evento_id_febr = "Perfil-1",
  taxon_sibcs_1999 = "Argissolo",
  stringsAsFactors = FALSE
)
hor <- data.frame(
  evento_id_febr = rep("Perfil-1", 3),
  camada_id = c("Ap", "B1", "B2"),
  profund_sup = c(0, 16, 32),
  profund_inf = c(16, 32, 50),
  cor_matriz_umido_munsell = rep("5YR 3/3", 3),
  cor_matriz_seco_munsell = rep("5YR 3/3", 3),
  estrutura_tipo = rep("blocos subangulares", 3),
```

```

    estrutura_grau = rep("moderada", 3),
    estrutura_cdiam = rep("média", 3),
    consistencia_umido = rep("friável", 3),
    consistencia_seco = rep("duro", 3),
    plasticidade = rep("plástico", 3),
    pegajosidade = rep("pegajoso", 3),
    stringsAsFactors = FALSE
  )
  febr2sse(pro, hor, tempfile(fileext = ".json"))

# Real example
if (interactive()) {
  profiles <- observation(
    data.set = "ctb0025", variable = c("taxon_sibcs", "relevo_drenagem"),
    standardization = list(units = TRUE, round = TRUE))
  idx <- profiles$evento_id_febr[1]
  profiles <- profiles[profiles$evento_id_febr %in% idx, ]
  horizons <- layer(
    data.set = "ctb0025", variable = "all",
    standardization =
      list(plus.sign = "remove", lessthan.sign = "remove",
           transition = "smooth", units = TRUE, round = TRUE))
  horizons <- horizons[horizons$evento_id_febr %in% idx, ]
  horizons[, 9:48] <- lapply(horizons[, 9:48], as.numeric)
  horizons <- cbind(
    horizons,
    morphology(x = horizons$morfologia_descricao, variable = "color"),
    morphology(x = horizons$morfologia_descricao, variable = "structure"),
    morphology(x = horizons$morfologia_descricao, variable = "consistence"),
    stringsAsFactors = FALSE)
  file <- ifelse(
    dir.exists("tmp"),
    paste0("tmp/febr2smartsolos-", idx, ".json"),
    paste0(tempdir(), "/febr2smartsolos-", idx, ".json"))
  febr2sse(profiles, horizons, file)
}

```

goto

Go to the Data Repository of the Brazilian Soil

Description

Visit the web assets of the [Data Repository of the Brazilian Soil](#).

Usage

```
goto(data.set, page)
```

Arguments

<code>data.set</code>	(optional) Character vector indicating a (unique) dataset whose metadata web page you wish to visit.
<code>page</code>	(optional) Character string indicating a web page of the Data Repository of the Brazilian Soil, with options: <ul style="list-style-type: none">• <code>"febr"</code>: FEBR main web page,• <code>"dictionary"</code>: FEBR data dictionary and vocabulary,• <code>"forum"</code>: FEBR public forum at Google Groups,• <code>"github"</code>: febr package source code repository on GitHub,• <code>"index"</code>: FEBR data set index,• <code>"search"</code>: data set search web page,• <code>"package"</code>: febr package on CRAN,• <code>"template"</code>: FEBR spreadsheet template on Google Sheets,• <code>"units"</code>: units and conversion factors used in FEBR.

Value

Load a given URL into an HTML browser via `utils::browseURL()`.

Author(s)

Alessandro Samuel-Rosa <alessandrosamuelrosa@gmail.com>

See Also

`utils::browseURL()`

Examples

```
if (interactive()) {  
  # Go to the FEBR web page  
  goto(page = "febr")  
  
  # Go to the data set index  
  goto(page = "index")  
  
  # Go to the GitHub repository  
  goto(page = "github")  
}
```

identification	<i>Get 'identification' table</i>
----------------	-----------------------------------

Description

Download data from the 'identification' ("identificacao") table of one or more soil datasets published in the [Data Repository of the Brazilian Soil](#). This table includes data such as dataset title and description, author and institution, data license, and much more.

Usage

```
identification(data.set, progress = TRUE, verbose = TRUE, febr.repo = NULL)
```

Arguments

data.set	Character vector indicating the identification code of one or more data sets. Use <code>data.set = "all"</code> to download all data sets.
progress	(optional) Logical value indicating if a download progress bar should be displayed.
verbose	(optional) Logical value indicating if informative messages should be displayed. Generally useful to identify datasets with inconsistent data. Please report to febr-forum@googlegroups.com if you find any issue.
febr.repo	(optional) Defaults to the remote file directory of the Federal University of Technology - Paraná at https://cloud.utfpr.edu.br/index.php/s/Df6dhfzYJ1DDeso . Alternatively, a local directory path can be informed if the user has a local copy of the data repository.

Value

A list of data frames or a data frame with data of the chosen dataset(s).

Note

Check the new core data download function [readFEBR\(\)](#).

Author(s)

Alessandro Samuel-Rosa <alessandrosamuelrosa@gmail.com>

Examples

```
# res <- identification(data.set = c("ctb0003", "ctb0000"))
res <- metadata(data.set = c("ctb0003", "ctb0002"))
```

layer	<i>Get 'layer' table</i>
-------	--------------------------

Description

Download data from the 'layer' ("camada") table of one or more datasets published in the [Data Repository of the Brazilian Soil](#). This table includes data such as sampling depth, horizon designation, and variables such as pH, carbon and clay content, and much more.

Usage

```
layer(
  data.set,
  variable,
  stack = FALSE,
  missing = list(depth = "keep", data = "keep"),
  standardization = list(plus.sign = "keep", plus.depth = 2.5, lessthan.sign = "keep",
    lessthan.frac = 0.5, repetition = "keep", combine.fun = "mean", transition = "keep",
    smoothing.fun = "mean", units = FALSE, round = FALSE),
  harmonization = list(harmonize = FALSE, level = 2),
  progress = TRUE,
  verbose = TRUE,
  febr.repo = NULL
)
```

Arguments

data.set	Character vector indicating the identification code of one or more data sets. Use <code>data.set = "all"</code> to download all data sets.
variable	(optional) Character vector indicating one or more variables. Accepts only general identification codes, e.g. "ferro" and "carbono". If missing, then a set of standard identification variables is downloaded. Use <code>variable = "all"</code> to download all variables. See 'Details' for more information.
stack	(optional) Logical value indicating if tables from different datasets should be stacked on a single table for output. Requires <code>standardization = list(units = TRUE)</code> – see below. Defaults to <code>stack = FALSE</code> , the output being a list of tables.
missing	(optional) List with named sub-arguments indicating what should be done with a layer missing data on sampling depth, depth, or data on variable(s), data. Options are "keep" (default) and "drop".
standardization	(optional) List with named sub-arguments indicating how to perform data # standardization. <ul style="list-style-type: none"> • <code>plus.sign</code> Character string indicating what should be done with the plus sign (+) commonly used along with the inferior limit of the bottom layer of an observation. Options are "keep" (default), "add", and "remove".

- `plus.depth` Numeric value indicating the depth increment (in centimeters) when processing the plus sign (+) with `plus.sign = "add"`. Defaults to `plus.depth = 2.5`.
 - `lessthan.sign` Character string indicating what should be done with the less-than sign (<) used to indicate that the value of a variable is below the lower limit of detection. Options are "keep" (default), "subtract", and "remove".
 - `lessthan.frac` Numeric value between 0 and 1 (a fraction) by which the lower limit of detection should be subtracted when `lessthan.sign = "subtract"`. Defaults to `lessthan.frac = 0.5`, i.e. subtract 50\
 - `repetition` Character string indicating what should be done with repetitions, i.e. repeated measurements of layers in an observation. Options are "keep" (default) and "combine". In the latter case, it is recommended to set `lessthan.sign = "subtract"` or `lessthan.sign = "remove"`.
 - `combine.fun` Character string indicating the function that should be used to combine repeated measurements of layers in an observation when `repetition = "combine"`. Options are "mean" (default), "min", "max", and "median".
 - `transition` Character string indicating what should be done about the wavy and irregular transition between subsequent layers in an observation. Options are "keep" (default) and "smooth".
 - `smoothing.fun` Character string indicating the function that should be used to smooth wavy and irregular transitions between subsequent layers in an observation when `transition = "smooth"`. Options are "mean" (default), "min", "max", and "median".
 - `units` Logical value indicating if the measurement unit(s) of the continuous variable(s) should be converted to the standard measurement unit(s). Defaults to `units = FALSE`, i.e. no conversion is performed. See [dictionary\(\)](#) for more information.
 - `round` Logical value indicating if the values of the continuous variable(s) should be rounded to the standard number of decimal places. Requires `units = TRUE`. Defaults to `round = FALSE`, i.e. no rounding is performed. See [dictionary\(\)](#) for more information.
- `harmonization` (optional) List with named sub-arguments indicating if and how to perform data harmonization.
- `harmonize` Logical value indicating if data should be harmonized. Defaults to `harmonize = FALSE`, i.e. no harmonization is performed.
 - `level` Integer value indicating the number of levels of the identification code of the variable(s) that should be considered for harmonization. Defaults to `level = 2`. See 'Details' for more information.
- `progress` (optional) Logical value indicating if a download progress bar should be displayed.
- `verbose` (optional) Logical value indicating if informative messages should be displayed. Generally useful to identify datasets with inconsistent data. Please report to <febr-forum@googlegroups.com> if you find any issue.
- `febr.repo` (optional) Defaults to the remote file directory of the Federal University of Technology - Paraná at <https://cloud.utfpr.edu.br/index.php/s/Df6dhfzYJ1DDeso>.

Alternatively, a local directory path can be informed if the user has a local copy of the data repository.

Details

Default variables: Default variables (fields) present in the 'layer' table are as follows:

- `dataset_id`. Identification of the dataset in FEBR to which an observation belongs.
- `evento_id_febr`. Identification code of an observation in a dataset.
- `camada_id`. Sequential layer number, from top to bottom.
- `camada_altid`. Layer designation according to some standard description guide.
- `amostra_id`. Laboratory number of a sample.
- `profund_sup`. Upper boundary of a layer (cm).
- `profund_inf`. Lower boundary of a layer (cm).

Further details about the content of the default variables (fields) can be found in https://docs.google.com/document/d/1Bqo8HtitZv11TXzTviVq2bI5dE6_t_fJt0HE-13IMqM (in Portuguese).

Harmonization: Data harmonization consists of converting the values of a variable determined using some method *B* so that they are (approximately) equivalent to the values that would have been obtained if the standard method *A* had been used instead. For example, converting carbon content values obtained using a wet digestion method to the standard dry combustion method is data harmonization.

A heuristic data harmonization procedure is implemented in the **febr** package. It consists of grouping variables based on a chosen number of levels of their identification code. For example, consider a variable with an identification code composed of four levels, `aaa_bbb_ccc_ddd`, where `aaa` is the first level and `ddd` is the fourth level. Now consider a related variable, `aaa_bbb_eee_fff`. If the harmonization is to consider all four coding levels (`level = 4`), then these two variables will remain coded as separate variables. But if `level = 2`, then both variables will be re-coded as `aaa_bbb`, thus becoming the same variable.

Value

A list of `data.frames` or a `data.frame` with, possibly standardize or harmonized, data of the chosen variable(s) of the chosen dataset(s).

Author(s)

Alessandro Samuel-Rosa <alessandrosamuelrosa@gmail.com>

See Also

[readFEBR\(\)](#), [observation\(\)](#), [dictionary\(\)](#), [unit\(\)](#)

Examples

```
if (interactive()) {
  res <- layer(data.set = "ctb0003")

  # Download two data sets and standardize units
```

```

res <- layer(
  data.set = paste("ctb000", 4:5, sep = ""),
  variable = "carbone", stack = TRUE,
  standardization = list(units = TRUE))

# Try to download a data set that is not available yet
res <- layer(data.set = "ctb0020")

# Try to download a non existing data set
# res <- observation(data.set = "ctb0000")
}

```

metadata

Get 'metadata' table

Description

Download data from the 'metadata' ("metadado") table of one or more datasets published in the [Data Repository of the Brazilian Soil](#). This table includes data such as variable names, description of analytical methods, and identification of analysis laboratories.

Usage

```
metadata(data.set, progress = TRUE, verbose = TRUE, febr.repo = NULL)
```

Arguments

data.set	Character vector indicating the identification code of one or more data sets. Use data.set = "all" to download all data sets.
progress	(optional) Logical value indicating if a download progress bar should be displayed.
verbose	(optional) Logical value indicating if informative messages should be displayed. Generally useful to identify datasets with inconsistent data. Please report to <febr-forum@googlegroups.com> if you find any issue.
febr.repo	(optional) Defaults to the remote file directory of the Federal University of Technology - Paraná at https://cloud.utfpr.edu.br/index.php/s/Df6dhfzYJ1DDeso . Alternatively, a local directory path can be informed if the user has a local copy of the data repository.

Value

A list of data frames or a data frame with metadata of the chosen dataset(s).

Note

Check the new core data download function [readFEBR\(\)](#).

Author(s)

Alessandro Samuel-Rosa <alessandrosamuelrosa@gmail.com>

Examples

```
# res <- metadata(data.set = c("ctb0003", "ctb0020"))
res <- metadata(data.set = c("ctb0003", "ctb0002"))
```

morphology

Soil morphology

Description

Extract and process soil morphological properties from field soil morphology descriptions.

Usage

```
morphology(x, variable = "color")
```

Arguments

x	Character string with field soil morphology description (in Portuguese).
variable	Character string defining the soil morphological property of interest. Options: color, structure, consistence.

Value

An object of class `data.frame`.

Author(s)

Alessandro Samuel-Rosa <alessandrosamuelrosa@gmail.com>

Examples

```
horizons <- layer(data.set = "ctb0025", variable = "morfologia_descricao")
color <- morphology(x = horizons$morfologia_descricao, variable = "color")
structure <- morphology(x = horizons$morfologia_descricao, variable = "structure")
consistence <- morphology(x = horizons$morfologia_descricao, variable = "consistence")

x <- "cinzento rosado (7.5YR 6/2, seco), bruno escuro (7.5YR 3/2, úmido)"
color <- morphology(x = x, variable = "color")
```

observation	<i>Get 'observation' table</i>
-------------	--------------------------------

Description

Download data from the 'observation' ("observacao") table of one or more datasets published in [Data Repository of the Brazilian Soil](#). This table includes data such as latitude, longitude, date of observation, underlying geology, land use and vegetation, local topography, soil classification, and much more.

Usage

```
observation(
  data.set,
  variable,
  stack = FALSE,
  missing = list(coord = "keep", time = "keep", data = "keep"),
  standardization = list(crs = NULL, time.format = NULL, units = FALSE, round = FALSE),
  harmonization = list(harmonize = FALSE, level = 2),
  progress = TRUE,
  verbose = TRUE,
  febr.repo = NULL
)
```

Arguments

data.set	Character vector indicating the identification code of one or more data sets. Use <code>data.set = "all"</code> to download all data sets.
variable	(optional) Character vector indicating one or more variables. Accepts only general identification codes, e.g. "ferro" and "carbono". If missing, then a set of standard identification variables is downloaded. Use <code>variable = "all"</code> to download all variables. See 'Details' for more information.
stack	(optional) Logical value indicating if tables from different datasets should be stacked on a single table for output. Requires <code>standardization = list(units = TRUE)</code> – see below. Defaults to <code>stack = FALSE</code> , the output being a list of tables.
missing	(optional) List with named sub-arguments indicating what should be done with an observation missing spatial coordinates, coord, date of observation, time, or data on variables, data. Options are "keep" (default) and "drop".
standardization	(optional) List with named sub-arguments indicating how to perform data standardization. <ul style="list-style-type: none"> • crs Character string indicating the EPSG code of the coordinate reference system (CRS) to which spatial coordinates should be transformed. For example, <code>crs = "EPSG:4674"</code>, i.e. SIRGAS 2000, the standard CRS for Brazil. Defaults to <code>crs = NULL</code>, i.e. no transformation is performed.

	<ul style="list-style-type: none"> • <code>time.format</code> Character string indicating how to format dates. For example, <code>time.format = "%d-%m-%Y"</code>, i.e. dd-mm-yyyy such as in 31-12-2001. Defaults to <code>time.format = NULL</code>, i.e. no formatting is performed. See <code>base::as.Date()</code> for more details. • <code>units</code> Logical value indicating if the measurement unit(s) of the continuous variable(s) should be converted to the standard measurement unit(s). Defaults to <code>units = FALSE</code>, i.e. no conversion is performed. See <code>dictionary()</code> for more information. • <code>round</code> Logical value indicating if the values of the continuous variable(s) should be rounded to the standard number of decimal places. Requires <code>units = TRUE</code>. Defaults to <code>round = FALSE</code>, i.e. no rounding is performed. See <code>dictionary()</code> for more information.
<code>harmonization</code>	<p>(optional) List with named sub-arguments indicating if and how to perform data harmonization.</p> <ul style="list-style-type: none"> • <code>harmonize</code> Logical value indicating if data should be harmonized. Defaults to <code>harmonize = FALSE</code>, i.e. no harmonization is performed. • <code>level</code> Integer value indicating the number of levels of the identification code of the variable(s) that should be considered for harmonization. Defaults to <code>level = 2</code>. See 'Details' for more information.
<code>progress</code>	(optional) Logical value indicating if a download progress bar should be displayed.
<code>verbose</code>	(optional) Logical value indicating if informative messages should be displayed. Generally useful to identify datasets with inconsistent data. Please report to <a href="mailto:<febr-forum@googlegroups.com>"><febr-forum@googlegroups.com> if you find any issue.
<code>febr.repo</code>	(optional) Defaults to the remote file directory of the Federal University of Technology - Paraná at https://cloud.utfpr.edu.br/index.php/s/Df6dhfzYJ1DDeso . Alternatively, a local directory path can be informed if the user has a local copy of the data repository.

Details

Default variables: Default variables (fields) present in the 'observation' table are as follows:

- `dataset_id`. Identification code of the dataset in the FEBR to which an observation belongs.
- `evento_id_febr`. Identification code of an observation in a dataset.
- `evento_data`. Date (dd-mm-yyyy) in which an observation was made.
- `coord_datum`. EPSG code of the coordinate reference system.
- `coord_longitude`. Longitude (deg) or easting (m).
- `coord_latitude`. Latitude (deg) or northing (m).
- `coord_precisao`. Precision with which the spatial coordinates were determined (m).
- `coord_fonte`. Source of the spatial coordinates.
- `pais_id`. Code (ISO 3166-1 alpha-2) of the country where an observation was made.
- `estado_sigla`. Acronym of the Brazilian federative unit where an observation was made.
- `municipio_nome`. Name of the Brazilian municipality where an observation was made.
- `subamostra_quanti`. Number of sub samples taken (used to indicate composite sampling).
- `amostra_area`. Sampling area (used to indicate areal or block sampling).

Further details about the content of the default variables (fields) can be found in https://docs.google.com/document/d/1Bqo8HtitZv11TXzTviVq2bI5dE6_t_fJt0HE-13IMqM (in Portuguese).

Harmonization: Data harmonization consists of converting the values of a variable determined using some method *B* so that they are (approximately) equivalent to the values that would have been obtained if the standard method *A* had been used instead. For example, converting carbon content values obtained using a wet combustion method to the standard dry combustion method is data harmonization.

A heuristic data harmonization procedure is implemented in the **febr** package. It consists of grouping variables based on a chosen number of levels of their identification code. For example, consider a variable with an identification code composed of four levels, `aaa_bbb_ccc_ddd`, where `aaa` is the first level and `ddd` is the fourth level. Now consider a related variable, `aaa_bbb_eee_fff`. If the harmonization is to consider all four coding levels (`level = 4`), then these two variables will remain coded as separate variables. But if `level = 2`, then both variables will be re-coded as `aaa_bbb`, thus becoming the same variable.

Value

A list of `data.frames` or a `data.frame` with, possibly standardize or harmonized, data of the chosen variable(s) of the chosen dataset(s).

Author(s)

Alessandro Samuel-Rosa <alessandrosamuelrosa@gmail.com>

See Also

[readFEBR\(\)](#), [layer\(\)](#), [dictionary\(\)](#), [unit\(\)](#)

Examples

```
if (interactive()) {
  res <- observation(data.set = "ctb0013")

  # Download two data sets and standardize CRS
  res <- observation(
    data.set = paste("ctb000", 4:5, sep = ""),
    variable = "taxon",
    standardization = list(crs = "EPSG:4674"))

  # Try to download a data set that is not available yet
  res <- observation(data.set = "ctb0020")

  # Try to download a non existing data set
  #res <- observation(data.set = "ctb0000")

}
```

readFEBR	<i>Get soil data</i>
----------	----------------------

Description

Download soil data from one or more data sets published in the [Data Repository of the Brazilian Soil](#).

Usage

```
readFEBR(  
  data.set,  
  data.table = c("identificacao", "versionamento", "metadado", "observacao", "camada"),  
  febr.repo = NULL,  
  verbose = TRUE,  
  ...  
)
```

Arguments

data.set	Character vector indicating the identification code of one or more data sets. Use <code>data.set = "all"</code> to download all data sets.
data.table	Character vector indicating one or more data tables, with supported values "identificacao", "versionamento", "metadado", "observacao", and "camada".
febr.repo	(optional) Defaults to the remote file directory of the Federal University of Technology - Paraná at https://cloud.utfpr.edu.br/index.php/s/Df6dhfzYJ1DDeso . Alternatively, a local directory path can be informed if the user has a local copy of the data repository.
verbose	(optional) Logical value indicating if informative messages should be displayed. Generally useful to identify issues—please report to < febr-forum@googlegroups.com > if you find any.
...	(optional) Arguments passed to <code>data.table::fread()</code> .

Value

A list of data tables (data frames) with data from the chosen data sets.

Author(s)

Alessandro Samuel-Rosa <alessandrosamuelrosa@gmail.com>

Examples

```
res <- readFEBR(data.set = "ctb0003")
```

readIndex

Get data set index

Description

Download the index of data sets published in the [Data Repository of the Brazilian Soil](#).

Usage

```
readIndex(...)
```

Arguments

... (optional) Arguments passed to `data.table::fread()`.

Value

An object of class `data.table::data.table`.

Author(s)

Alessandro Samuel-Rosa <alessandrosamuelrosa@gmail.com>

Examples

```
index <- readIndex()
```

readVocabulary

Get controlled vocabulary

Description

Download the controlled vocabulary used in the [Data Repository of the Brazilian Soil](#).

Usage

```
readVocabulary(...)
```

Arguments

... (optional) Arguments passed to `data.table::fread()`.

Value

An object of class `data.table::data.table`.

Author(s)

Alessandro Samuel-Rosa <alessandrosamuelrosa@gmail.com>

Examples

```
vocab <- readVocabulary()
```

taxonomy	<i>Process soil taxonomy</i>
----------	------------------------------

Description

Extract and process soil taxonomic data from textual soil classification descriptions of the Brazilian Soil Classification System (SiBCS).

Usage

```
taxonomy(  
  text,  
  method = "decompose",  
  sep = " ",  
  pattern = c(" ", " ", " A ", " textura ")  
)
```

Arguments

text	Character string(s) with soil classification description(s) (in Portuguese).
method	Character string defining the string processing method. Options: <ul style="list-style-type: none"> • "decompose": decompose the Brazilian soil classification into its four higher categorical levels (order, suborder, large group, and subgroup).
sep	Character string. Defaults to sep = " ".
pattern	Character string (in Portuguese). Defaults to pattern = c(" ", " ", " A ", " textura ").

Value

An object of class `base::data.frame` with four named columns: `ordem` (UPPER CASE), `subordem` (UPPER CASE), `grangrupo` (Title Case), and `subgrupo` (lower case).

Author(s)

Alessandro Samuel-Rosa <alessandrosamuelrosa@gmail.com>

References

Santos, H. G. dos, Jacomine, P. K. T., Anjos, L. H. C. dos, Oliveira, V. Á. de, Lumberras, J. F., Coelho, M. R., Almeida, J. A. de, Araújo Filho, J. C. de, Oliveira, J. B. de, & Cunha, T. J. F. (2018). Sistema Brasileiro de Classificação de Solos (5th ed., p. 531). Embrapa. <https://www.infoteca.cnptia.embrapa.br/infoteca/handle/doc/1094003>.

IBGE. (2015). Manual Técnico de Pedologia (3rd ed., p. 430). Instituto Brasileiro de Geografia e Estatística. <https://biblioteca.ibge.gov.br/visualizacao/livros/liv95017.pdf>

Examples

```
text <-
  c("CAMBISSOLO HÁPLICO Ta Eutrófico léptico A proeminente textura média",
    "PLANOSSOLO HÁPLICO Distrófico solódico êndico plintossólico, textura média, Tb",
    "CHERNOSSOLO ARGILÚVICO Órtico típico textura média com cascalho/argilosa com cascalho",
    "ARGISSOLO VERMELHO-AMARELO",
    "Latossolo"
  )
taxonomy(text)
```

unit

Get units table

Description

Download data from the *units* ("unidades") table of the [Data Repository of the Brazilian Soil](#). This includes measurement units and conversion factors. This is used to standardize the data contained in a dataset when downloading it via [layer\(\)](#) or [observation\(\)](#).

Usage

```
unit(source, target)
```

Arguments

source (optional) Character vector indicating one or more source measurement units.
target (optional) Character vector indicating one or more target measurement units.

Value

A data.frame with source and target measurement units and their corresponding conversion factors.

Author(s)

Alessandro Samuel-Rosa <alessandrosamuelrosa@gmail.com>

References

Teixeira, P. C., Donagemma, G. K., Fontana, A., Teixeira, W. G. (2017) *Manual de Métodos de Análise de Solo*. Brasília: Embrapa.

See Also

The *unit* table at <https://docs.google.com/spreadsheets/d/1tU4Me3NJqk4NH2z0jvMryG0bSSQLCvGqdLEL5bv0f1o>

Examples

```
#res <- unit(source = c("%", "dag/kg"), target = "g/kg")  
#res
```

Index

`base::as.Date()`, [15](#)
`base::data.frame`, [19](#)
`base::writeLines()`, [5](#)

`data.table::data.table`, [18](#)
`data.table::fread()`, [17](#), [18](#)
`dataset`, [2](#)
`dictionary`, [3](#)
`dictionary()`, [10](#), [11](#), [15](#), [16](#)

`febr (dataset)`, [2](#)
`febr2sf`, [4](#)
`febr2spdf (dataset)`, [2](#)
`febr2sse`, [5](#)
`febr2xlsx (dataset)`, [2](#)

`goto`, [6](#)

`header (dataset)`, [2](#)

`identification`, [8](#)

`layer`, [9](#)
`layer()`, [16](#), [20](#)

`metadata`, [12](#)
`morphology`, [13](#)

`observation`, [14](#)
`observation()`, [4](#), [11](#), [20](#)

`readFEBR`, [17](#)
`readFEBR()`, [8](#), [11](#), [12](#), [16](#)
`readIndex`, [18](#)
`readVocabulary`, [18](#)

`standard (dataset)`, [2](#)

`taxonomy`, [19](#)

`unit`, [20](#)
`unit()`, [11](#), [16](#)
`utils::browseURL()`, [7](#)