Package ‘fwtraits’

May 8, 2026

Title Extract Species Ecological Parameters from Www.freshwaterecology.info

Version 1.0.0

Description Support the extraction and seamless integration of species ecological traits or preferences from the www.freshwaterecology.info into several ecological model workflows. During data extraction, different taxonomic levels are acceptable, including species, genus, and family, based on the availability of data in the database. The data is cached after the first search and can be accessed during and after online interactions. Only scientific names are acceptable in the search; local or English names are not allowed. A user API key is required to start using the package.

License GPL (>= 3)

Encoding UTF-8

RoxygenNote 7.3.2

URL `https://github.com/AnthonyBasooma/fwtraits`, `https://anthonybasooma.github.io/fwtraits/`, `https://github.com/AnthonyBasooma/fwtraits/releases/tag/V1.0.0`

BugReports `https://github.com/AnthonyBasooma/fwtraits/issues`

Depends R (>= 4.1.0)

Imports curl, httr2, jsonlite, methods, utils, rstudioapi, R.cache

Suggests ggplot2, knitr (>= 1.48), rmarkdown (>= 2.28), sf, vcr, dplyr, tidytext, testthat (>= 3.0.0), FD, tidyr, tibble, stats, cluster

Config/testthat/edition 3

VignetteBuilder knitr

LazyData true

NeedsCompilation no

Author Anthony Basooma [aut, cre] (ORCID: <`https://orcid.org/0000-0002-8994-9989`>), Florian Borgwardt [ctb, fnd] (ORCID: <`https://orcid.org/0000-0002-8974-7834`>),

Thomas Hein [ctb, fnd, ths] (ORCID:
<https://orcid.org/0000-0002-7767-4607>),
Sami Domisch [ctb],
Merret Buurman [ctb],
Vanessa Bremerich [ctb],
Sonia Steffany Recinos Brizuela [ctb],
Martin Tschikof [ctb],
Astrid Schmidt-Kloiber [ctb, fnd, dtc] (ORCID:
<https://orcid.org/0000-0001-8839-5913>)

Maintainer Anthony Basooma <`anthony.basooma@boku.ac.at`>

Repository CRAN

Date/Publication 2025-11-11 10:00:16 UTC

Contents

checktrait 2
check_packages 3
classifydata 4
clean_names 5
fuzzy_codes 6
fw_be4ustart 6
fw_cite 7
fw_dbguide 7
fw_fdcompute 8
fw_fdendro 9
fw_fetchdata 10
fw_geosites 13
fw_keyload 13
fw_register 14
fw_searchdata 14
fw_setapikey 16
fw_token 16
fw_visualize 18
invertbackbone 19
pbenthodata 20
speciesdata 21

Index **22**

`checktrait` *Checks the traits spelling compared to user input.*

Description

Checks the traits spelling compared to user input.

Usage

```
checktrait(x, std, mindist = 0.3, error = 0.8, grp = NULL, warn = TRUE)
```

Arguments

`x`	`string` or `vector`. The traits to be checked for spelling errors and matching database entries.
`std`	`lits`. A list with standard traits names from the the database to compare with user entries.
`mindist`	`numeric`. Set a threshold for trait similarity between the user provided and that found in the database. The lower the percentage, the higher the similarity between the user provided and standard trait names.
`error`	`numeric`. Also percentage to improve the distance based checked implemented or set in mindist parameter
`grp`	`grp`. The taxa names checked for. see `fw_searchdata`.
`warn`	`logical` To show species name warning checks and traits cleaning. Default `FALSE`.

Value

`list` or `string`. A list, vector or string of cleaned traits names based on the user provided and standard database traits for downloading.

`check_packages` *Check for packages to install and respond to use*

Description

Check for packages to install and respond to use

Usage

```
check_packages(pkgs)
```

Arguments

`pkgs`	list of packages to install

Value

error message for packages to install

`classifydata` *Data with ecological parameters classification.*

Description

A `tibble`

Usage

```
data(classifydata)
```

Format

A `tibble` The dataset has 817 rows and 4 columns.

- Organism group: Are the taxonomic groups in the www.freshwaterecology.info database.
- parameter_cleaned: Are the ecological parameter names in the database.
- DataType: Are the data classification for each ecological parameter. The data types are still under revision to improve harmony. Therefore, the users can provide a different classification based on their expertise.

Details

The database will be used internally to assign data types such as nominal, ordinal, ratio, and interval to ecological parameters.

References

Schmidt-Kloiber, A., & Hering, D. (2015). Www.freshwaterecology.info - An online tool that unifies, standardises and codifies more than 20,000 European freshwater organisms and their ecological preferences. Ecological Indicators, 53, 271-282. https://doi.org/10.1016/j.ecolind.2015.02.007.

Examples

```
## Not run:

data("classifydata")

classifydata

## End(Not run)
```

`clean_names` *Check and clean species names to match standard names in the database.*

Description

Check and clean species names to match standard names in the database.

Usage

```
clean_names(
  sp,
  grouplists,
  prechecks = FALSE,
  standard_dataset = NULL,
  percenterror = 80,
  errorness = 30,
  full = FALSE,
  warn,
  taxalevel
)
```

Arguments

`sp`	`string` or `vector`. Species scientific names to be checked. Although the spellings are checked, the users should check for the species name provided to avoid not being being detected in the database.
`grouplists`	`list`. List of data downloaded in the `fw_searchdata` function. If species considered in `sp` parameter are fishes, then the fishes lists should be provided otherwise the species names will be rejected.
`prechecks, standard_dataset`	`logical`. If `TRUE` the standard prechecks will be done on both the invertebrates and bentho species names before search for ecological parameters from the database. The standard names is provided with the dataset to reduce on the time in identifying the standard tyxonomic names for the macroinvertebrates in the database. @param taxalevel `string` Allowed taxonomic levels at which data can retrieved. Default is `'species'` but data can also be downloaded at family level, genus, and taxa group level.
`percenterror`	`numeric`. The number used as a cutoff to infer similarity of the user provided name and what is found in the database. The higher the percentage, the higher the similarity the species name provided by the user and the one in the database. `percenterror` ranges from 0 to 100 but the default is 80 to ensure that wrong names are not selected at low similarity percentage cutoff.
`errorness`	`numeric` Similar to `percenterror`, `errorness` parameter uses the distance differences between the user-provided names and all the taxa group species standard names. The lower the percentage error, the higher the similarity in the

	species names provided. Default is 20 and beyond 30, a warning is showed to avoid wrong species replace the user provided name, which leads to extracting wrong traits.
`full`	`logical` `TRUE` if a dataframe with both cleaned and uncleaned species are required. If `FALSE` then the a species list will be produced after cleaning. Default `FALSE`.
`warn`	To alert user on the species names cleaning errors and warnings.
`taxalevel`	`string` Allowed taxonomic levels at which data can retrieved. Default is `'species'` but data can also be downloaded at family level, genus, and taxa group level.

Value

`vector` or `string` clean species name that is also found in the database.

`fuzzy_codes` *Internal package and create dummy codes*

Description

Internal package and create dummy codes

Usage

```
fuzzy_codes(trait, FD)
```

Arguments

`trait`	`dataframe` Data matrix
`FD`	`logical` Either to compute functional diversity indices or not.

`fw_be4ustart` *Steps to follow in using the Freshwater Information Platform*

Description

Steps to follow in using the Freshwater Information Platform

Usage

```
fw_be4ustart()
```

Value

`list` steps to follow in using the FWDB

Author(s)

Anthony Basooma

Examples

```
## Not run:
b4us <- fw_be4ustart()

## End(Not run)
```

`fw_cite` *Get database citations*

Description

Get database citations

Usage

```
fw_cite(cachefolder = "cache")
```

Arguments

`cachefolder`	`string`. The root path were the cached data will be saved on the user PC. If the path is not provided, the cached information will be saved in the current working directly.

Value

string of the package citation

`fw_dbguide` *Standard table with taxonomic groups and their traits explanations.*

Description

Standard table with taxonomic groups and their traits explanations.

Usage

```
fw_dbguide(organismgroup = NULL, cachefolder = "cache")
```

Arguments

`organismgroup`	`string` or `vector`. Taxa group names to aid the users in filtering the standard table for species traits and their explanations.
`cachefolder`	`string`. The root path were the cached data will be saved on the user PC. If the path is not provided, the cached information will be saved in the current working directly.

Value

`dataframe` A dataset with taxonomic groups, traits and their explanations.

Examples

```
## Not run:

dbase <- fw_dbguide(cachefolder = 'cache')


## End(Not run)
```

`fw_fdcompute` *Compute the functional diversity indices and community weighted means.*

Description

Compute the functional diversity indices and community weighted means.

Usage

```
fw_fdcompute(
  fwdata,
  sitesdata,
  species,
  sites = NULL,
  dist = 1000,
  abund = NULL,
  FD = TRUE,
  dummy = TRUE
)
```

Arguments

`fwdata`	`list` A list from the `fw_fetchdata` function is the only data type accepted.
`sitesdata`	`dataframe` Dataframe with sites for which functional diversity and community weighted means indices will be computed for.

`species`	`string` The species column in the sitesdata.
`sites`	`string` The sites column in the sitesdata. If the sites are not provided by the geometry column, then sites can be auto generated by clustering closely separated records into sites.
`dist`	`integer` The distance used to cluster points as sites. The default is 1000.
`abund`	`string` Column with abundance data. If not provided, then only functional richness can be computed.
`FD`	`logical` To indicate whether to compute functional richness or community weighted means. The default is TRUE.
`dummy`	`logical` To form fuzzy code for the categorical traits. Default is TRUE.

Value

dataframe

Examples

```
## Not run:

set.seed(1135)
data('speciesdata')
speciesdata$abundance <- rnorm(n = nrow(speciesdata), 4.3, 1.2)

spgeo1 <- speciesdata |>
 sf::st_as_sf(coords = c('decimalLongitude', 'decimalLatitude'), crs = sf::st_crs(4326))

sptraits <- fw_fetchdata(data = speciesdata,
                        ecoparams = c('rheophily habitat', 'spawning habitat',
                                      'feeding diet adult'),
                        taxonomic_column = 'scientificName',
                        organismgroup = 'fi')
#functional richness
testdata5 <- fw_fdcompute(fwdata = sptraits,
                         sitesdata = speciesdata,
                         sites = 'waterBody',
                         species = 'scientificName',
                         FD = TRUE)


## End(Not run)
```

`fw_fdendro` *For computing the functional dendogram*

Description

For computing the functional dendogram

Usage

```
fw_fdendro(fwdata, method = "average", k = 4, plot = FALSE)
```

Arguments

`fwdata`	`list` A list from the `fw_fetchdata` function is the only data type accepted.
`method`	`string` Clustering method, including ward, average, and complete
`k`	`integer` Determine the number of cluster the user wants to output after data clustering. The default is 4.
`plot`	`logical` Either `TRUE` to show the plot of functional dendogram. Defualt is `FALSE`.

Value

dataframe, plot

Examples

```
## Not run:
  fishtraits <- fw_fetchdata(data = speciesdata,
                        ecoparams = c('rheophily habitat', 'spawning habitat',
                        'feeding diet adult'),
                        taxonomic_column = 'scientificName',
                        organismgroup = 'fi')


 head(fdendoclust, 3)

 table(fdendoclust$cluster)


## End(Not run)
```

`fw_fetchdata` *Extracting the traits from the downloaded data.*

Description

Extracting the traits from the downloaded data.

Usage

```
fw_fetchdata(
  data,
  organismgroup,
  ecoparams = NULL,
  taxalevel = "species",
```

```
  taxonomic_column = NULL,
  organismgroup_column = NULL,
  apikey = NULL,
  seed = 1134,
  secure = TRUE,
  percenterror = 80,
  errorness = 20,
  warn = FALSE,
  inform = FALSE,
  cachefolder = "cache",
  details = FALSE
)
```

Arguments

`data`	`vector`. The list or vector with species names for which ecological references needs to be extracted from the database.
`organismgroup`	`string`. The organism group to download from the platform. The allowed group includes `"fi"`, `"mi"`, `"pp"`, `"pb"`, `"di"`,`"mp"` for fishes, macroinvertebrates, phytoplankton, phytobenthos, diatoms, and macrophytes, respectively. Multiple groups allowed, such as `'pp'`, `'di'`. • `pp`: Pytoplankton. • `mp`: Macrophytes • `mi`: Macroinvertebrates • `fi`: Fishes • `di`: Diatoms • `pb`: Phytobenthos without diatoms
`ecoparams`	`vector`. Selected traits that should be downloaded for a particular organism group. Check `fw_dbguide` for the allowed traits in the database.
`taxalevel`	`string` Allowed taxonomic levels at which data can be retrieved. Default is `'species'` but data can also be downloaded at `family`, `genus`, and `taxagroup` level.
`taxonomic_column`	`string`. If the data is a dataframe, the species column is required and provided in this parameter. The column should have complete species name and not genus and species provided separately.
`organismgroup_column`	`string` If the data is a dataframe, and more than one taxonomic group exists in the data, the `organismgroup_column` is required to iterate over the taxonomic groups separately.
`apikey`	`string`. The API key is automatically loaded using the loadapikey() internal function.
`seed`	`integer`. An integer to help track the caching of the access token generated during data collation. If a user wants a new token, the seed should be changed.
`secure`	`logical`. If `TRUE`, the User will be prompted to set the API key in the .Renviron file by running the `fw_setapikey` function. The User must strictly type in

	API_KEY = 'api key', save, close the file and restart the R session or RStudio for the API_KEY environment to be captured. If `FALSE`, then the key will be entered directly in the API_KEY directly in the fw_token() function. This method is insecure, since other users can obtain the key from the codes.
`percenterror`	`numeric`. The number used as a cutoff to infer similarity of the user provided name and what is found in the database. The higher the percentage, the higher the similarity the species name provided by the user and the one in the database. `percenterror` ranges from 0 to 100 but the default is 80 to ensure that wrong names are not selected at low similarity percentage cutoff.
`errorness`	`numeric` Similar to `percenterror`, `errorness` parameter uses the distance differences between the user-provided names and all the taxa group species standard names. The lower the percentage error, the higher the similarity in the species names provided. Default is 20 and beyond 30, a warning is showed to avoid wrong species replace the user provided name, which leads to extracting wrong traits.
`warn`	`logical` To show species name warning checks and traits cleaning. Default `FALSE`.
`inform`	`logical`. This is to indicate if the token has been successfully generated. Default `TRUE`.
`cachefolder`	`string`. The root path were the cached data will be saved on the user PC. If the path is not provided, the cached information will be saved in the current working directly.
`details`	`loical`. Outputs the downloaded details including the organism groups considered by the user, the functional call, and whether some groups were successful in retrieving data.

Value

`dataframe` The output has four sections, including

- ecodata, which is the complete dataframe with all the taxonomic names and ecological parameters. \ item taxasearch: is a table with the taxonomic names,both original and cleaned names.
- fetch: an indication that data has been fetched from www.freshwaterecology.info.
- fun_call: A functional call used internally to review the data cleaning process.

Examples

```
## Not run:

dfextract <- fw_fetchdata(data = "Abramis brama", organismgroup = 'fi', inform = TRUE,
                          ecoparams = 'migration', cachefolder = 'cache' )


## End(Not run)
```

`fw_geosites` *Auto generation of species sites*

Description

Auto generation of species sites

Usage

```
fw_geosites(x, dist = 1000)
```

Arguments

`x`	`dataframe` Geo spatial data with geometry column from sf package.
`dist`	`integer` The distance used to cluster points as sites. The default is 1000.

Value

dataframe

Examples

```
## Not run:
geospdata <- speciesdata |>
sf::st_as_sf(coords = c('decimalLongitude', 'decimalLatitude'),
            crs = sf::st_crs(4326))
 xx <- fw_geosites(geospdata)

## End(Not run)
```

`fw_keyload` *For loading the API key from the environment.*

Description

For loading the API key from the environment.

Usage

```
fw_keyload(keyvar = "API_KEY")
```

Arguments

`keyvar`	`string` Is the API name as indicated in the user environment. Default is `"API_KEY"`.

Value

string of API

`fw_register` *Open the registration page*

Description

Open the registration page

Usage

```
fw_register()
```

Value

registration page

`fw_searchdata` *To download data from the Freshwaterecology.info database.*

Description

The function provides seamless access and download of species ecological parameters, traits, or indicators from the www.freshwaterecology.info database. The function allows multiple organism groups, including macroinvertebrates, fish, phytoplankton, phytobenthos, macrophytes, and diatoms.

Usage

```
fw_searchdata(
  organismgroup,
  taxa_searched = NULL,
  ecoparams = NULL,
  apikey = NULL,
  warn = TRUE,
  seed = 1135,
  secure = TRUE,
  inform = FALSE,
  taxalevel = NULL,
  cachefolder = "cache"
)
```

Arguments

`organismgroup`	`string`. The organism group to download from the platform. The allowed group includes `"fi"`, `"mi"`, `"pp"`, `"pb"`, `"di"`,`"mp"` for fishes, macroinvertebrates, phytoplankton, phytobenthos, diatoms, and macrophytes, respectively. Multiple groups allowed, such as `'pp'`, `'di'`. • `pp`: Pytoplankton. • `mp`: Macrophytes • `mi`: Macroinvertebrates • `fi`: Fishes • `di`: Diatoms • `pb`: Phytobenthos without diatoms
`taxa_searched`	`string` An internal placeholder to accommodate the standard taxonomic names for invertebrates and phytobenthos from the database.
`ecoparams`	`vector`. Selected traits that should be downloaded for a particular organism group. Check `fw_dbguide` for the allowed traits in the database.
`apikey`	`string`. The API key is automatically loaded using the loadapikey() internal function.
`warn`	`logical` To show species name warning checks and traits cleaning. Default `FALSE`.
`seed`	`integer`. An integer to help track the caching of the access token generated during data collation. If a user wants a new token, the seed should be changed.
`secure`	`logical`. If `TRUE`, the User will be prompted to set the API key in the .Renviron file by running the `fw_setapikey` function. The User must strictly type in API_KEY = 'api key', save, close the file and restart the R session or RStudio for the API_KEY environment to be captured. If `FALSE`, then the key will be entered directly in the API_KEY directly in the fw_token() function. This method is insecure, since other users can obtain the key from the codes.
`inform`	`logical`. This is to indicate if the token has been successfully generated. Default `TRUE`.
`taxalevel`	`string` Allowed taxonomic levels at which data can be retrieved. Default is `'species'` but data can also be downloaded at `family`, `genus`, and `taxagroup` level.
`cachefolder`	`string`. The root path were the cached data will be saved on the user PC. If the path is not provided, the cached information will be saved in the current working directly.

Details

Downloading macroinvertebrates data takes a lot of time because the database has a lot of data, considerably slowing the process. Therefore, the `taxa_searched` parameter is recommended for the user to indicate only particular traits, family, orders, or taxa group where the species falls. Also, the phytobenthos requires providing the `taxa_searched` to enable a search for a particular taxon group. The parameter should not be provided for other organism groups because it will not be used in the data search.

Value

List of download species traits, ecological paramaeters or traits.

Examples

```
## Not run:
dfsearch <- fw_searchdata(organismgroup = 'fi',
                          ecoparams = 'migration', cachefolder = 'cache')

## End(Not run)
```

`fw_setapikey` *Function and instructions creating the API_KEY variable in the User environment.*

Description

Function and instructions creating the API_KEY variable in the User environment.

Usage

```
fw_setapikey(verbose = TRUE)
```

Arguments

verbose	`logical` If logical, then the instruction message will show. Default `TRUE`.

Value

string of instructions for saving or setting the API in the R session.

`fw_token` *Access and loading the token key*

Description

The function updates the authentication token automatically, which the servers generate every six hours. The function uses the API key, a one-time key provided during registration or by database managers for already registered users. Since the authentication token expires, the seed parameter allows caching across user sessions. Therefore, the data downloaded with a particular seed will be stored in memory and can be retrieved by the user. PC rather than from servers, and hence, tremendously optimize the speed of data access. The token is generated in two ways, depending on whether the codes will be shared with others or for personal use. If they are for personal use, the API key is directly pasted into the pop-up after the fw_token is executed. However, in the latter circumstance, it is advisable to store the API key in the R user environment and encrypt it during code execution. Check the vignettes for handling the API key on the fwtraits GitHub.

Usage

```
fw_token(
  apikey = NULL,
  seed = NULL,
  cachefolder = NULL,
  secure = TRUE,
  inform = FALSE
)
```

Arguments

`apikey`	`string`. The API key is automatically loaded using the loadapikey() internal function.
`seed`	`integer`. An integer to help track the caching of the access token generated during data collation. If a user wants a new token, the seed should be changed.
`cachefolder`	`string`. The root path where the cached data will be saved on the user's PC. If the path is not provided, the cached information will be saved in the current working directly.
`secure`	`logical`. If `TRUE`, the User will be prompted to set the API key in the .Renviron file by running the `fw_setapikey` function. The User must strictly type in API_KEY = 'api key', save, close the file and restart the R session or RStudio for the API_KEY environment to be captured. If `FALSE`, then the key will be entered directly in the API_KEY directly in the fw_token() function. This method is insecure, since other users can obtain the key from the codes.
`inform`	`logical`. This is to indicate if the token has been successfully generated. Default `TRUE`.

Value

`string` token authentication token key

See Also

`fw_setapikey`

Examples

```
## Not run:

#1.Use the API key in shared R examples


## End(Not run)
```

`fw_visualize` *ggplot2 visualization.*

Description

ggplot2 visualization.

Usage

```
fw_visualize(
  output,
  scales = "free",
  ncol = 2,
  params = NULL,
  organism_group = NULL,
  color = "purple",
  na.rm = TRUE
)
```

Arguments

`output`	fetchdata output only accepted.
`scales`	indicate if the scales are free, free_x, and free_y.
`ncol`	Number of columns to display the data in facet_wrap.
`params`	species ecological parameters selected if there are more than one.
`organism_group`	Particular organism group to filter out to allow visualization.
`color`	`string` Change the color for the bar graphs being plotted.
`na.rm`	`logical` To remove NA in the category names or species traits.

Value

ggplot2 display of the retrieved ecological parameters

Examples

```
## Not run:
fetchdata <- fw_fetchdata(data = c("Abramis brama",'Salmo trutta fario'),
         organismgroup = 'fi',
         ecoparams = c('migration',"rheophily habitat"), cachefolder = 'cache')

fw_visualize(fetchdata)


## End(Not run)
```

`invertbackbone` *Taxonomic data for species in the freshwaterecology.info database*

Description

A `tibble`

Usage

```
data(invertbackbone)
```

Format

A `tibble` 10421 rows and 3 columns.

- Taxgroup: Higher taxonomic grouping for the species, for example, Bivalvia.
- Family: Taxonomic classification, e.g., CARDIIDAE
- Taxon: Lower taxonomic grouping for the species.eg., Parvicardium exiguum

Details

Standard taxonomic backbone for macroinvertebrates from www.freshwaterecology.info. All macroinvetebrates species are crosschecked with this database improve quality control of the species names provided by the user.

References

Schmidt-Kloiber, A., & Hering, D. (2015). www.freshwaterecology.info - An online tool that unifies, standardizes and codifies more than 20,000 European freshwater organisms and their ecological preferences. Ecological Indicators, 53, 271-282. https://doi.org/10.1016/j.ecolind.2015.02.007.

Examples

```
## Not run:

data("invertbackbone")
invertbackbone

## End(Not run)
```

`pbenthodata` *Taxonomic data for the phytobentho species in www.freshwaterecology.info database.*

Description

A `tibble`

Usage

```
data(pbenthodata)
```

Format

A `tibble` The dataset has 1857 rows and 2 columns.

- Taxgroup: Higher taxonomic grouping for the species, for example, Bacillariophyceae.
- Taxon: Lower taxonomic grouping for the species.eg., Achnanthes acus, Achnanthes brevipes, Achnanthes brevipes var. brevipes

Details

This dataset was extracted from the www.freshwaterecology.info database and formed a standardized reference for the species names for phytobenthos. Therefore, all user-provided species are checked across this database to identify whether or not they exist.

References

Schmidt-Kloiber, A., & Hering, D. (2015). Www.freshwaterecology.info - An online tool that unifies, standardises and codifies more than 20,000 European freshwater organisms and their ecological preferences. Ecological Indicators, 53, 271-282. https://doi.org/10.1016/j.ecolind.2015.02.007.

Examples

```
## Not run:

data("pbenthodata")

pbenthodata

## End(Not run)
```

`speciesdata` *Efiplus data used to develop ecological sensitivity parameters for riverine species in European streams and rivers.*

Description

A `tibble`

Usage

```
data(speciesdata)
```

Format

A `tibble` 99 rows and 23 columns.

- scientificName: The fish species names extracted from the EFIPLUS dataset.
- waterBody: The water body from which the species records were collected.
- decimalLatitude: Species location
- decimalLongitude: Species occurrence records.
- MRR: The locality where the species was collated or sampled from
- Date: the day, month, and year when the species record was collected.
- : year: The year when the species record was collated.
- Locality: particular locality where the species was sampled.
- country: The country where the record was made.

Details

BQEs sensitivity to global/climate change in European rivers: implications for reference conditions and pressure-impact-recovery chains (Logez et al. 2012). An extract has been made for usage in this package but for more information write to ihg@boku.ac.at

References

Logez M, Belliard J, Melcher A, Kremser H, Pletterbauer F, Schmutz S, Gorges G, Delaigue O, Pont D. 2012. Deliverable D5.1-3: BQEs sensitivity to global/climate change in European rivers: implications for reference conditions and pressure-impact-recovery chains.

Examples

```
## Not run:

data("speciesdata")
speciesdata

## End(Not run)
```

Index

- ∗ **Classification**
  - `classifydata`, 4
- ∗ **European**
  - `speciesdata`, 21
- ∗ **Macroinvertebrates**
  - `invertbackbone`, 19
- ∗ **Phytobentho**
  - `pbenthodata`, 20
- ∗ **dataset**
  - `speciesdata`, 21
- ∗ **data**
  - `classifydata`, 4
  - `pbenthodata`, 20
- ∗ **wide**
  - `speciesdata`, 21

- `check_packages`, 3
- `checktrait`, 2
- `classifydata`, 4
- `clean_names`, 5

- `fuzzy_codes`, 6
- `fw_be4ustart`, 6
- `fw_cite`, 7
- `fw_dbguide`, 7, *11*, *15*
- `fw_fdcompute`, 8
- `fw_fdendro`, 9
- `fw_fetchdata`, 10
- `fw_geosites`, 13
- `fw_keyload`, 13
- `fw_register`, 14
- `fw_searchdata`, *3*, *5*, 14
- `fw_setapikey`, *11*, *15*, 16, *17*
- `fw_token`, 16
- `fw_visualize`, 18

- `invertbackbone`, 19

- `pbenthodata`, 20

- `speciesdata`, 21