Package ‘worms’

October 12, 2022

Type Package

Title Retriving Aphia Information from World Register of Marine Species

Description Retrieves taxonomic information from <http://www.marinespecies.org> using WoRMS' RESTful Webservice. Utility functions aim at taxonomic consistency.

Version 0.2.2

Author Jan Holstein [aut, cre, cph]

Maintainer Jan Holstein <janmholstein@gmail.com>

Depends httr, plyr

License GNU Affero General Public License

Encoding UTF-8

LazyData true

RoxygenNote 6.0.1

NeedsCompilation no

Repository CRAN

Date/Publication 2018-04-25 13:58:37 UTC

R topics documented:

`northseamacrozoobenthos` *Over 1600 taxonomic, vernacular and trivial names of benthic critters from the North Sea*

Description

Used for tests and demonstration.

Usage

`data(northseamacrozoobenthos)`

Format

a vector of class `character`.

Author(s)

Jan M Holstein , 2017-03-05

Source

various

See Also

`wormsbynames`, `wormsbymatchnames`

`worms` *worms: Use WoRMS RESTful webservice to scrape Aphia information from World Register of Marine Species*

Description

The worms package provides two kinds of functions:

a) retrieving taxonomic information using WoRMS' RESTful Webservice by using taxon name search, fuzzy matching, or Aphia ID search implementing methods documented at `http://www.marinespecies.org/rest/`

b) functions that parse the data for synonyms in order to complete the dataset so that for every taxon in the dataset the respective taxon with status 'accepted' exists as well. Constructed references to the respective taxon with status 'accepted' help aggregating biodiversity data without the use of synonyms, alternative representations, and common misspellings leading to errors.

Check out `https://github.com/janhoo/worms/` for the developement version.

References

This package is not connected or endorsed by WoRMS. According to WoRMS, information from World Register of Marine Species is free to use under the condition that they are cited (CC-BY). While no license model is specified for the webservice employed, we strongly recommended to give reference to WoRMS, e.g., www.marinespecies.org, 18/06/17 (CC-BY). The citation for the full database is:

WoRMS Editorial Board (2017). World Register of Marine Species. Available from http://www.marinespecies.org at VLIZ. Accessed <today>. doi:10.14284/170

For single taxa, references are given in the citation column Please give proper reference to them.

`wormsaccepted` *Constructs "accepted_id" column which contains the "AphiaID" of the respective "accepted" taxon*

Description

takes data.frame as output by `wormsbynames` , `wormsbymatchnames`, or `wormsbyid` and add field "accepted_id" wich contains the "AphiaID" of the respective "accepted" taxon

Usage

```
wormsaccepted(x, verbose = TRUE, n_iter = 10)
```

Arguments

`x`	data.frame
`verbose`	be verbose
`n_iter`	maximum search depth. Usually 3 is sufficient. Safety feature for breaking the `while` loop

Details

This function helps updating you taxon information and eliminates ambiguity because the valid AphiaID is nor neccessary the AphiaID of an accepted taxon. You should run `wormsconsolidate` bevorhand to enshure all "accepted" taxons are present.

Value

a data frame.

Examples

```
## start with IDs that are no longer up to date
# get the Aphia information
u<-wormsbyid(c(424548,340537))

#recursively retrive information on the taxa they refer to
v<-wormsconsolidate(u)

# what are the currently correct "accepted" taxa? Answer: "accepted_id".
w<-wormsaccepted(v)
w[,c("scientificname","AphiaID","status","valid_AphiaID","valid_name","accepted_id")]
```

`wormsbyid` *GET AphiaRecordByAphiaID*

Description

takes more than one AphiaID and retrives AphiaRecords from WoRMS

Usage

```
wormsbyid(x, verbose = TRUE, ids = FALSE, sleep_btw_chunks_in_sec = 0.01)
```

Arguments

`x`	AphiaIDs
`verbose`	be verbose
`ids`	add column "id" and "name" with running id and search names
`sleep_btw_chunks_in_sec`	pause between requests

Details

This function will take a integer vector with AphiaIDs, retrive AphiaRecords from www.marinespecies.org using the GET /AphiaRecordByAphiaID Method described at http://www.marinespecies.org/rest/. Results will be output to a data.frame with each row being a record.

For examples, see `wormsaccepted`

Value

a data frame.

`wormsbymatchnames` *GET AphiaRecordsByMatchNames*

Description

takes character vector with taxon names and retrives AphiaRecords from WoRMS

Usage

```
wormsbymatchnames(taxon_names, verbose = TRUE, ids = FALSE,
  chunksize = 50, marine_only = "true", sleep_btw_chunks_in_sec = 0.1)
```

Arguments

`taxon_names`	character vector with names of taxa to look up.
`verbose`	be verbose
`ids`	add column "id" and "name" with running id and search names
`chunksize`	only 50 taxa can be looked up per request, so request are split up into chunks (should be 50 or less)
`marine_only`	Limit to marine taxa. Default=true
`sleep_btw_chunks_in_sec`	pause between requests

Details

This function will take a character vector with taxon names, retrive AphiaRecords (CC-BY) from www.marinespecies.org using the GET /AphiaRecordsByName/ScientificName Method described at http://www.marinespecies.org/rest/. Results will be output to a data.frame with each row being a record. For each name given, only the one AphiaRecord will be retrived. AphiaRecord with "accepted" status are preferred. If not present last entry will be taken which seems to result in best results.

Value

a data frame.

`wormsbynames` *GET AphiaRecordsByNames*

Description

takes character vector with taxon names and retrives AphiaRecords from WoRMS

Usage

```
wormsbynames(taxon_names, ids = FALSE, match = FALSE, verbose = TRUE,
  chunksize = 50, like = "false", marine_only = "true",
  sleep_btw_chunks_in_sec = 0.1)
```

Arguments

`taxon_names`	character vector with names of taxa to look up.
`ids`	add column "id" and "name" with running id and search names
`match`	taxon_names that could not retrieved will be retried with `wormsbymatchnames`. Implies "id=TRUE"
`verbose`	be verbose
`chunksize`	there is a limit to the number of taxa that can be looked up at once, so request are split up into chunks. This limit seems to be variable. 50 is very safe.
`like`	Add a "%"-sign after the ScientificName (SQL LIKE function). Default=true
`marine_only`	Limit to marine taxa. Default=true
`sleep_btw_chunks_in_sec`	pause between requests

Details

This function will take a character vector with taxon names, retrive AphiaRecords (CC-BY) from www.marinespecies.org using the GET /AphiaRecordsByName/ScientificName Method described at http://www.marinespecies.org/rest/. Results will be output to a data.frame with each row being a record. For each name given, only the one AphiaRecord will be retrived. AphiaRecord with "accepted" status are preferred. If not present last entry will be taken which seems to result in best results.

Value

a data frame.

Examples

```
taxon_names <- c( "Westwodilla caecula" ,  "Abra alba", "Chaetozone cf. setosa",  "Algae" )
w <- wormsbynames(taxon_names)
## print unrecognized returns
failed_species <- rownames(w[is.na(w[,1]),])

## try again with fuzzy matching turned on
w <- wormsbynames(taxon_names, match=TRUE)

## this is how to load taxon_names from file
write.csv(taxon_names , file = "tax.csv",
        row.names = FALSE,
        na = "")
## check it out, then load it
read.csv(file = "tax.csv",
        na = "",
        stringsAsFactors = FALSE,
        col.names = FALSE)
## save results to file to inspect with, e.g. spreadsheet software
write.csv(w,file = "aphiainfo.csv",
        na = "",
        col.names = TRUE,
        row.names = TRUE)
```

`wormsconsolidate` *Recursivly retrieves respective "accepted" AphiaRecords for all synonyms if not already there*

Description

takes data.frame as output by `wormsbynames` , `wormsbymatchnames`, or `wormsbyid` and retrieves additional Aphia records (CC-BY) for not-"accepted" records in order to ultimately have "accepted" synonyms for all records in the dataset.

Usage

```
wormsconsolidate(x, verbose = TRUE, sleep_btw_chunks_in_sec = 0.01,
  once = FALSE)
```

Arguments

`x`	data.frame
`verbose`	be verbose
`sleep_btw_chunks_in_sec`	pause between requests
`once`	only one retrival iteration. No concatination of output with result. (For debugging)

Details

This function will take a integer vector with AphiaIDs, retrive AphiaRecords from www.marinespecies.org using the GET /AphiaRecordByAphiaID Method described at http://www.marinespecies.org/rest/. Results will be outbut to a data.frame with each row being a record.

For examples, see `wormsaccepted`

Value

a data frame.

Index