

Package: rphylotastic (via r-universe)

September 2, 2024

Title An R Interface to 'Phylotastic' Web Services

Version 0.0.8.1

Maintainer Brian O'Meara <omeara.brian@gmail.com>

Description This wraps the 'Phylotastic' services APIs described on Web Services at <www.phylotastic.org>. The main use case is to return a phylogenetic tree for a set of species, but the services also include ways to extract species names from web pages, perform taxonomic name resolution, retrieve a list of all descendant species of a taxon, find images of a species, and more.

URL <https://github.com/phylotastic/rphylotastic>

BugReports <https://github.com/phylotastic/rphylotastic/issues>

Depends R (>= 3.3.0)

Imports ape, brranching, httr, jsonlite, utils, phylobase, methods, rentrez (>= 1.0.4), plotrix

Suggests testthat, knitr, rmarkdown, taxize, datelife, rotl

License GPL-2

Encoding UTF-8

LazyData true

RoxygenNote 7.2.3

VignetteBuilder knitr

Repository <https://phylotastic.r-universe.dev>

RemoteUrl <https://github.com/phylotastic/rphylotastic>

RemoteRef HEAD

RemoteSha 1cd60f2d99cc31e485ef9e38f76ff7f80c7f958c

Contents

rphylotastic-package	2
arclabels	3
file_get_scientific_names	4
file_get_scientific_names_from_GNRD	5
flower_plant_fams	6
get_base_url	6
get_list_server_url	7
get_species_from_list	7
insert_species_in_list	8
remove_species_from_list	9
replace_species_in_list	10
species_get_image_data	11
species_get_info	11
taxa_common_to_scientific	12
taxa_get_otol_tree	13
taxa_get_phylomatic_tree	13
taxa_resolve_names_with_gnr	14
taxa_resolve_names_with_otol	15
taxa_toss_binomials	15
taxon_get_species	16
taxon_get_species_from_country	17
taxon_get_species_with_genome	17
taxon_separate_dark_taxa_using_genbank	18
taxon_separate_dark_taxa_using_otol	19
terrestrial_plant_orders	19
text_get_scientific_names	20
traits_get_tree	21
update_species_in_list	22
url_get_scientific_names	23
Index	24

rphylotastic-package *rphylotastic: An R Interface to 'Phylotastic' Web Services*

Description

This wraps the 'Phylotastic' services APIs described on Web Services at <www.phylotastic.org>. The main use case is to return a phylogenetic tree for a set of species, but the services also include ways to extract species names from web pages, perform taxonomic name resolution, retrieve a list of all descendant species of a taxon, find images of a species, and more.

Details

See `vignette("rphylotastic", package = "rphylotastic")` for an overview of the package and its utilities.

Author(s)

Maintainer: Brian O'Meara <omeara.brian@gmail.com>

Authors:

- Abu Saleh Md Tayeen <abusalehmdtayeen@gmail.com>
- Luna L. Sanchez Reyes <sanchez.reyes.luna@gmail.com>

See Also

Useful links:

- <https://github.com/phylostatic/rphylostatic>
- Report bugs at <https://github.com/phylostatic/rphylostatic/issues>

arclabels	<i>Add arc labels to tips of a phylogeny; works for non-monophyletic groups and single tip lineages.</i>
-----------	--

Description

Add arc labels to tips of a phylogeny; works for non-monophyletic groups and single tip lineages.

Usage

```
arclabels(phy, tips, ...)

## Default S3 method:
arclabels(
  phy = NULL,
  tips,
  text,
  plot_singletons = TRUE,
  ln.offset = 1.02,
  lab.offset = 1.06,
  cex = 1,
  orientation = "horizontal",
  ...
)
```

Arguments

phy	An object of class phylo.
tips	A character vector (or a list ?) with the names of the tips that belong to the clade or group. If multiple groups are going to be plotted, tips must be given in the form of a list.
...	optional arguments for arc.cladelabels.

text	A character vector indicating the desired text to label the arcs.
plot_singletons	Boolean. If TRUE (default), it will add arcs (and labels) to single tip lineages too. If FALSE, no arc or labels will be plotted over that tip..
ln.offset	line offset (as a function of total tree height) for arc.cladelabels.
lab.offset	label offset for arc.cladelabels.
cex	character expansion factor.
orientation	orientation of the text. Can be orientation = "vertical" (the default) or "horizontal".

Value

NULL

file_get_scientific_names

Function to pull scientific names from file

Description

This uploads a file (a PDF, Microsoft Word document, plain text file, etc.) and extracts all scientific names from it. For example, you can input a PDF of a scientific article and it will return all the scientific names in that article.

Usage

```
file_get_scientific_names(file_name, search_engine = 0, above_species = FALSE)
```

Arguments

file_name	The file path and name to extract names from
search_engine	1 to use TaxonFinder, 2 to use NetiNeti, 0 to use both
above_species	Boolean. Default to FALSE. If TRUE it will only return scientific names above the species level.

Details

It requires that curl is installed on your system.

Value

A vector of scientific names

See Also

https://github.com/phylostatic/phylo_services_docs/tree/master/ServiceDescription

`file_get_scientific_names_from_GNRD`*Function to pull scientific names from file using GNRD*

Description

This uploads a file (a PDF, Microsoft Word document, plain text file, etc.) and extracts all scientific names from it. For example, you can input a PDF of a scientific article and it will return all the scientific names in that article.

Usage

```
file_get_scientific_names_from_GNRD(  
  file_name,  
  search_engine = 0,  
  above_species = FALSE  
)
```

Arguments

<code>file_name</code>	The file path and name to extract names from
<code>search_engine</code>	1 to use TaxonFinder, 2 to use NetiNeti, 0 to use both
<code>above_species</code>	Boolean. Default to FALSE. If TRUE it will only return scientific names above the species level.

Details

It requires that curl is installed on your system.

Value

A vector of scientific names

See Also

https://github.com/phylostatic/phylo_services_docs/tree/master/ServiceDescription

flower_plant_fams *Flowering plants families from Open Tree Taxonomy*

Description

Flowering plants families from Open Tree Taxonomy

Usage

```
flower_plant_fams
```

Format

A character vector

Details

```
flower_plant_fams <- datelife::get_ott_children(ott_ids = 99252, ott_rank = "family") flower_plant_fams
<- flower_plant_fams[[1]] flower_plant_fams <- rownames(flower_plant_fams)[as.character(flower_plant_fams[, "rank"])
== "family"] usethis::use_data(flower_plant_fams, overwrite = TRUE)
```

Author(s)

Luna L. Sanchez-Reyes <lsanche7@utk.edu> Brian O'Meara <bomeara@utk.edu>

Source

<https://tree.opentreeoflife.org/about/taxonomy-version/ott3.0>

get_base_url *Get Phylotastic base URL return The URL for the phylotastic server*

Description

Get Phylotastic base URL return The URL for the phylotastic server

Usage

```
get_base_url()
```

`get_list_server_url` *Return Species List server URL*

Description

Return Species List server URL

Usage

```
get_list_server_url()
```

`get_species_from_list` *Get existing list/lists of species*

Description

Get existing list/lists of species

Usage

```
get_species_from_list(  
    userid,  
    access_token,  
    list_id,  
    verbose = FALSE,  
    content = TRUE  
)
```

Arguments

<code>userid</code>	A valid gmail address of the user
<code>access_token</code>	Access token of the gmail address
<code>list_id</code>	An integer id of the list to retrieve
<code>verbose</code>	(optional)By default FALSE and shows minimal meta-data of the list.
<code>content</code>	(optional)By default TRUE and shows the species collection of the list

Value

An existing list with metadata and content based on parameters

See Also

https://github.com/phylostatic/phylo_services_docs/tree/master/ServiceDescription

Examples

```
# This gives you the syntax, but since the access token expires after one hour,
# this particular example will not work.
## Not run:
userid = "abusalehmdtayeen@gmail.com"
access_token = "ya29..zQLmLjbyujJjwV6RVSM2sy-mkeaKu-9"
list_id = 12
verbose = TRUE
content = FALSE
get_species_from_list(userid, access_token, list_id, verbose, content)

## End(Not run)
```

```
insert_species_in_list
```

Insert list of species

Description

Insert list of species

Usage

```
insert_species_in_list(userid, listObj)
```

Arguments

userid	A valid gmail address of the user
listObj	A list object

Value

A list with the id of the new list created

See Also

https://github.com/phylostatic/phylo_services_docs/tree/master/ServiceDescription

Examples

```
userid = "abusalehmdtayeen@gmail.com"
listObj = list(list_extra_info="", list_description="A sublist on the bird species added",
  list_keywords=c("bird", "endangered species", "Everglades"),
  list_curator="HD Laughinghouse", list_origin="webapp",
  list_curation_date="02-24-2016", list_source="des", list_focal_clade="Aves",
  list_title="Bird Species List",list_author=c("Bass", "O. & Cunningham", "R."),
  list_date_published="01-01-2017", is_list_public=TRUE,
  list_species=list(list(family="",scientific_name="Aix sponsa",
    scientific_name_authorship="", vernacular_name="Wood Duck",
```

```
        phylum="", nomenclature_code="ICZN", order="Anseriformes", class=""),
        list(family="", scientific_name="Anas strepera",
            scientific_name_authorship="", vernacular_name="Gadwall",
            phylum="", nomenclature_code="ICZN",
            order="Anseriformes", class="") ))
insert_species_in_list(userid, listObj)
```

remove_species_from_list

Remove an existing list of species

Description

Remove an existing list of species

Usage

```
remove_species_from_list(userid, access_token, list_id)
```

Arguments

userid	A valid gmail address of the user
access_token	Access token of the gmail address
list_id	An integer id of the list to retrieve

Value

A list with the id of the list removed

See Also

https://github.com/phylostatic/phylo_services_docs/tree/master/ServiceDescription

Examples

```
# This gives you the syntax, but since the access token expires after one hour,
# this particular example will not work.
## Not run:
userid = "abusalehmdtayeem@gmail.com"
access_token = "ya29..zQLmLjbyujJjwV6RVSM2sy-mkeaKu-9"
list_id = 12
remove_species_from_list(userid, access_token, list_id)

## End(Not run)
```

`replace_species_in_list`*Replace a list of species*

Description

Replace a list of species

Usage

```
replace_species_in_list(userid, access_token, list_id, speciesObj)
```

Arguments

<code>userid</code>	A valid gmail address of the user
<code>access_token</code>	Access token of the gmail address
<code>list_id</code>	An integer id of the list to be modified
<code>speciesObj</code>	A species object to replace with

Value

A list with the old species and new species list

See Also

https://github.com/phylostatic/phylo_services_docs/tree/master/ServiceDescription

Examples

```
# This gives you the syntax, but since the access token expires after one hour,  
# this particular example will not work.  
## Not run:  
userid = "abusalehmdtayeem@gmail.com"  
access_token = "ya29..zQLmLjbyujJjwV6RVSM2sy-mkeaKu-9"  
list_id = 12  
speciesObj = list( list(family="", scientific_name="Aix sponsa", scientific_name_authorship="",  
vernacular_name="Wood Duck", phylum="", nomenclature_code="ICZN", order="Anseriformes", class=""))  
replace_species_in_list(userid, access_token, list_id, speciesObj)  
  
## End(Not run)
```

`species_get_image_data`*Get image metadata of a list of species*

Description

Get image metadata of a list of species

Usage

```
species_get_image_data(species)
```

Arguments

`species` A vector of names

Value

A data frame of image metadata(image urls, license info etc.) of species

See Also

https://github.com/phylostatic/phylo_services_docs/tree/master/ServiceDescription
or <https://eol.org/api/>

`species_get_info`*Get information from Encyclopedia of Life of a list of species*

Description

Get information from Encyclopedia of Life of a list of species

Usage

```
species_get_info(species)
```

Arguments

`species` A vector of names

Value

A data frame of species information from Encyclopedia of Life. Type of information available varies among species.

See Also

https://github.com/phylostatic/phylo_services_docs/tree/master/ServiceDescription
or <https://eol.org/api/>

taxa_common_to_scientific

Convert common names to scientific names.

Description

Convert common names to scientific names.

Usage

```
taxa_common_to_scientific(taxa, service = "NCBI", multiple = FALSE)
```

Arguments

taxa	A character vector of common names. Binomials can be spaced with underscore or white space.
service	Which service to use: NCBI, ITIS, or TROPICOS
multiple	If TRUE, then the service will return multiple matches (if available) for each common name in the input list.

Value

A vector of scientific names. Output order may not correspond to input order.

See Also

taxize package for name resolution in general and its sci2comm function.

Examples

```
taxa <- c("blue_whale", "swordfish", "killer whale")
scientific <- taxa_common_to_scientific(taxa)
print(scientific)
```

taxa_get_otol_tree *Get OTOL induced subtree*

Description

Get OTOL induced subtree

Usage

```
taxa_get_otol_tree(taxa)
```

Arguments

taxa The vector of names, already resolved to match OTOL taxa

Value

A phylo object

See Also

https://github.com/phylostatic/phylo_services_docs/tree/master/ServiceDescription
or the rotl package, another interface to Open Tree of Life

Examples

```
taxa <- c("Crabronidae", "Ophiocordyceps", "Megalyridae",  
          "Formica polycтена", "Tetramorium caespitum",  
          "Pseudomyrmex", "Carebara diversa", "Formicinae")  
phy <- taxa_get_otol_tree(taxa)  
plot(phy)
```

taxa_get_phylomatic_tree
 Get phylomatic subtree

Description

Get phylomatic subtree

Usage

```
taxa_get_phylomatic_tree(taxa)
```

Arguments

taxa The vector of names, already resolved

Value

A phylo object

See Also

https://github.com/phylostatic/phylo_services_docs/tree/master/ServiceDescription
or the interface of phylomatic <http://phylodiversity.net/phylomatic/>

Examples

```
phy <- taxa_get_phylomatic_tree(c("Panthera leo", "Panthera onca",  
                                "Panthera tigris", "Panthera uncia"))  
plot(phy)
```

taxa_resolve_names_with_gnr

Resolve Scientific Names with GNR TNRS

Description

Resolve Scientific Names with GNR TNRS

Usage

```
taxa_resolve_names_with_gnr(taxa)
```

Arguments

taxa The vector of names

Details

Mispelled or incorrect names will be dropped.

Value

A vector of correct names. THE ORDER MAY NOT CORRESPOND TO YOUR INPUT ORDER.

See Also

https://github.com/phylostatic/phylo_services_docs/tree/master/ServiceDescription
or the rotl package, another interface to Open Tree of Life, or the taxize package for name resolution
in general.

`taxa_resolve_names_with_otol`*Resolve Scientific Names with Open Tree TNRS*

Description

Resolve Scientific Names with Open Tree TNRS

Usage

```
taxa_resolve_names_with_otol(taxa)
```

Arguments

`taxa` The vector of names

Value

A vector of corrected names. THE ORDER MAY NOT CORRESPOND TO YOUR INPUT ORDER.

See Also

https://github.com/phylostatic/phylo_services_docs/tree/master/ServiceDescription or the `rotl` package, another interface to Open Tree of Life, or the `taxize` package for name resolution in general.

Examples

```
my.species.raw <- c("Formica polycтена", "Formica exsectoides", "Farmica pacifica")
my.species.corrected <- taxa_resolve_names_with_otol(my.species.raw)
print(my.species.corrected)
```

`taxa_toss_binomials` *Keep the first part of the binomial from a vector of taxon names that includes species binomial names*

Description

Keep the first part of the binomial from a vector of taxon names that includes species binomial names

Usage

```
taxa_toss_binomials(taxa)
```

Arguments

taxa A character vector of taxon names.

Value

A character vector of lineage names above the species level.

See Also

https://github.com/phylostatic/phylo_services_docs/tree/master/ServiceDescription
or the `rotl` package, another interface to Open Tree of Life

Examples

```
taxa_toss_binomials("Vulpes_vulpes")
```

taxon_get_species *Get all species from a taxon from Open Tree of Life taxonomy.*

Description

Get all species from a taxon from Open Tree of Life taxonomy.

Usage

```
taxon_get_species(taxon, filters = c("environmental", "sp\\.\"", "cf\\.\""))
```

Arguments

taxon A character vector of length 1. Specify the taxon name to get all species for. If vector is longer than 1, it will only take the first element and ignore all other names.

filters A character vector of strings to exclude.

Value

A character vector of species names.

See Also

https://github.com/phylostatic/phylo_services_docs/tree/master/ServiceDescription
or the `rotl` package, another interface to Open Tree of Life

Examples

```
print(taxon_get_species("Vulpes"))
```

taxon_get_species_from_country
Get all species filtered by country from a taxon

Description

Get all species filtered by country from a taxon

Usage

```
taxon_get_species_from_country(  
  taxon,  
  country,  
  filters = c("environmental", "sp\\.", "cf\\.")  
)
```

Arguments

taxon	A character vector of length 1. Specify the taxon name to get a subset of species that are established in a particular country
country	A country name where species of the input taxon are established.
filters	A character vector of strings to exclude

Value

A vector of names

See Also

https://github.com/phylostatic/phylo_services_docs/tree/master/ServiceDescription
or the rotl package, another interface to Open Tree of Life

taxon_get_species_with_genome
Get all species that have genome sequence in NCBI from a taxon

Description

Get all species that have genome sequence in NCBI from a taxon

Usage

```
taxon_get_species_with_genome(  
  taxon,  
  filters = c("environmental", "sp\\.", "cf\\.")  
)
```

Arguments

taxon	A character vector of length 1. Specify the taxon name to get a subset of species having genome sequence
filters	A character vector of strings to exclude

Value

A vector of names

See Also

https://github.com/phylostatic/phylo_services_docs/tree/master/ServiceDescription or the rotl package, another interface to Open Tree of Life

taxon_separate_dark_taxa_using_genbank

Separate dark from known taxa on another database

Description

Separate dark from known taxa on another database

Usage

```
taxon_separate_dark_taxa_using_genbank(
  taxon,
  filters = c("environmental", "sp\\. ", "cf\\. ", "uncultured", "unidentified",
    " clone", " enrichment"),
  verbose = TRUE,
  sleep = 0
)
```

Arguments

taxon	A taxon to get all species for
filters	A character vector of strings to exclude
verbose	Update on how many are done
sleep	How many seconds to sleep between calls (on top of rentrez's defaults)

Value

A list containing a vector of dark names, a vector of known names, and fraction.dark

`taxon_separate_dark_taxa_using_otol`*Separate dark from known taxa on OpenTree of Life*

Description

Separate dark from known taxa on OpenTree of Life

Usage

```
taxon_separate_dark_taxa_using_otol(  
  taxon,  
  filters = c("environmental", "sp\\.", "cf\\.", "uncultured", "unidentified",  
             " clone", " enrichment")  
)
```

Arguments

taxon	A taxon to get all species for
filters	A character vector of strings to exclude

Value

A list containing a vector of dark names, a vector of known names, and fraction.dark

See Also

https://github.com/phylostatic/phylo_services_docs/tree/master/ServiceDescription
or the rotl package, another interface to Open Tree of Life

`terrestrial_plant_orders`*Flowering plants families from Open Tree Taxonomy*

Description

Flowering plants families from Open Tree Taxonomy

Usage

```
terrestrial_plant_orders
```

Format

A character vector

Details

```
terrestrial_plant_orders <- datelife::get_ott_children(ott_ids = 56610, ott_rank = "order") terrestrial_plant_orders <- terrestrial_plant_orders[[1]] terrestrial_plant_orders <- rownames(terrestrial_plant_orders)[as.character(== "order")] usethis::use_data(terrestrial_plant_orders, overwrite = TRUE)
```

Author(s)

Luna L. Sanchez-Reyes <lsanche7@utk.edu> Brian O'Meara <bomeara@utk.edu>

Source

<https://tree.opentreeoflife.org/about/taxonomy-version/ott3.0>

text_get_scientific_names

Function to pull scientific names from text

Description

This takes a string of text and extracts any scientific names in the text. Other words in the text are ignored.

Usage

```
text_get_scientific_names(text)
```

Arguments

text The text string to extract names from

Value

A data.frame of scientific names and other data from GNRD

Examples

```
text <- "Formica polyctena is a species of European red wood ant in  
the genus Formica. The pavement ant, Tetramorium caespitum  
is an ant native to Europe."  
print(text_get_scientific_names(text))
```

traits_get_tree	<i>Function to get tree for a traits set.</i>
-----------------	---

Description

A common use case is having a set of traits for a set of species and wanting to get a tree for those species. This involves resolving to the same taxonomy, getting a tree for those species, and then (optionally) pruning the traits and tree to the same set of taxa (this is optional: there are approaches to make up traits or phylogenetic placement in the absence of information).

Usage

```
traits_get_tree(
  traits,
  tnrs_source = "otol",
  tree_source = "otol",
  prune = TRUE,
  summary_format = "phylo_biggest",
  ...
)
```

Arguments

traits	Data.frame with species names as rownames
tnrs_source	Source for taxonomic name resolution; options are "otol" and "gnr". If set to NULL, assumes names are fine as is
tree_source	Source for tree; options are "otol", "phylomatic", "datelife".
prune	If TRUE, delete taxa to matching sets only
summary_format	What format to return from datelife
...	Other options to pass to datelife::datelife_search

Details

For sources of trees, besides the standard ones (Open Tree of Life, Phylomatic), datelife is also an option. For this, you will have had to install the datelife package (which is only suggested, not required, by this code). It will return the chronogram in the tree store with the most overlap to your set of taxa by default, but you can change options with summary_format; see ?datelife::datelife_search for more info

Value

list with a phy and a traits object, both pruned to the same taxon set, as well as citation information for the sources of the taxonomic resolution and phylogeny (also cite this package and, if you use it, datelife)

`update_species_in_list`*Update metadata of a list of species*

Description

Update metadata of a list of species

Usage

```
update_species_in_list(userid, access_token, list_id, listObj)
```

Arguments

<code>userid</code>	A valid gmail address of the user
<code>access_token</code>	Access token of the gmail address
<code>list_id</code>	An integer id of the list to be modified
<code>listObj</code>	A list object to update with

Value

A list with modified list metadata

See Also

https://github.com/phylostatic/phylo_services_docs/tree/master/ServiceDescription

Examples

```
# This gives you the syntax, but since the access token expires after one hour,  
# this particular example will not work.  
## Not run:  
userid = "abusalehmdtayeen@gmail.com"  
access_token = "ya29..zQLmLjbyujJjwV6RVSM2sy-mkeaKu-9"  
list_id = 12  
listObj = list(list_description="A sublist on the bird species",  
              list_keywords=c("bird", "Everglades"))  
update_species_in_list(userid, access_token, list_id, listObj)  
  
## End(Not run)
```

`url_get_scientific_names`*Function to pull scientific names from web pages*

Description

Function to pull scientific names from web pages

Usage

```
url_get_scientific_names(URL, search_engine = 0, above_species = FALSE)
```

Arguments

URL	The URL to extract names from. Can be a pdf url.
search_engine	1 to use TaxonFinder, 2 to use NetiNeti, 0 to use both
above_species	Boolean. Default to FALSE. If TRUE it will only return scientific names above the species level.

Value

A vector of scientific names. It returns unique matches.

See Also

https://github.com/phylostatic/phylo_services_docs/tree/master/ServiceDescription

Examples

```
# get scientific names from a wikipedia web page:  
url_get_scientific_names(URL = "https://en.wikipedia.org/wiki/Plain_pigeon")  
# get scientific names from a pdf URL:  
url_get_scientific_names(URL =  
  "http://darwin-online.org.uk/converted/pdf/1897_Insectivorous_F1229.pdf")
```

Index

- * **magnoliophyta**
 - flower_plant_fams, [6](#)
 - terrestrial_plant_orders, [19](#)
- * **otol**
 - flower_plant_fams, [6](#)
 - terrestrial_plant_orders, [19](#)
- arclabels, [3](#)
- file_get_scientific_names, [4](#)
- file_get_scientific_names_from_GNRD, [5](#)
- flower_plant_fams, [6](#)
- get_base_url, [6](#)
- get_list_server_url, [7](#)
- get_species_from_list, [7](#)
- insert_species_in_list, [8](#)
- remove_species_from_list, [9](#)
- replace_species_in_list, [10](#)
- rphylotastic (rphylotastic-package), [2](#)
- rphylotastic-package, [2](#)
- species_get_image_data, [11](#)
- species_get_info, [11](#)
- taxa_common_to_scientific, [12](#)
- taxa_get_otol_tree, [13](#)
- taxa_get_phylomatic_tree, [13](#)
- taxa_resolve_names_with_gnr, [14](#)
- taxa_resolve_names_with_otol, [15](#)
- taxa_toss_binomials, [15](#)
- taxon_get_species, [16](#)
- taxon_get_species_from_country, [17](#)
- taxon_get_species_with_genome, [17](#)
- taxon_separate_dark_taxa_using_genbank, [18](#)
- taxon_separate_dark_taxa_using_otol, [19](#)
- terrestrial_plant_orders, [19](#)
- text_get_scientific_names, [20](#)
- traits_get_tree, [21](#)
- update_species_in_list, [22](#)
- url_get_scientific_names, [23](#)