Package ‘rredlist’

January 28, 2020

Type Package

Title 'IUCN' Red List Client

The 'IUCN' Red List is a global list of threatened and endangered species.
Functions cover all of the Red List 'API' routes. An 'API' key is required.

Version 0.6.0

License MIT + file LICENSE

URL https://github.com/ropensci/rredlist (devel)
    https://docs.ropensci.org/rredlist (docs)

BugReports https://github.com/ropensci/rredlist/issues

Encoding UTF-8

Language en-US

Imports crul (>= 0.3.8), jsonlite (>= 1.1)

Suggests testthat, vcr (>= 0.4.0)

RoxygenNote 7.0.2

X-schema.org-applicationCategory Biodiversity

X-schema.org-keywords IUCN, biodiversity, API, web-services, traits,
habitat, species, conservation

X-schema.org-isPartOf https://ropensci.org

NeedsCompilation no

Author Scott Chamberlain [aut, cre],
rOpenSci [fnd] (https://ropensci.org/),
Maëlle Salmon [ctb]

Maintainer Scott Chamberlain <myrmecocystus@gmail.com>

Repository CRAN

Date/Publication 2020-01-28 06:10:03 UTC
r redlist-package

R topics documented:

- rredlist-package .................................................. 2
- rl_citation .......................................................... 3
- rl_common_names ................................................... 4
- rl_comp_groups ..................................................... 5
- rl_countries .......................................................... 6
- rl_growth_forms ..................................................... 6
- rl_habitats ............................................................ 8
- rl_history ............................................................ 9
- rl_measures .......................................................... 10
- rl_narrative .......................................................... 11
- rl_occ_country ....................................................... 12
- rl_occ_country_ ...................................................... 13
- rl_regions ............................................................ 14
- rl_sp ................................................................. 15
- rl_sp_category ....................................................... 16
- rl_sp_citation ....................................................... 17
- rl_sp_count .......................................................... 18
- rl_sp_country ......................................................... 19
- rl_synonyms .......................................................... 20
- rl_threats ............................................................. 21
- rl_use_iucn ........................................................... 22
- rl_version ............................................................ 22

Index 23

Description

rredlist - IUCN Red List Client

Taxonomic Names vs. IUCN IDs

From the documentation (quoting): "It is advisable wherever possible to use the taxon name (species name) to make your API calls, rather than using IDs. IDs are not immovable are expected to be used mainly by organisations that work closely with the IUCN Red List."

Authentication

IUCN requires you to get your own API key, an alphanumeric string that you need to send in every request. See key A IUCN API token. See rl_use_iucn() for help getting and storing it. Get it at http://api.v3.iucnredlist.org/api/v3/token Keep this key private. You can pass the key in to each function via the key parameter, but it’s better to store the key either as a environment variable (IUCN_REDLIST_KEY) or an R option (iucn_redlist_key) - we recommend using the former option.
High vs. Low level package APIs

High level API  High level functions do the HTTP request and parse data to a data.frame for ease of downstream use. The high level functions have no underscore on the end of the function name, e.g., `rl_search()`

Low level API  The parsing to data.frame in the high level API does take extra time. The low level API only does the HTTP request, and gives back JSON without doing any more parsing. The low level functions DO have an underscore on the end of the function name, e.g., `rl_search_()`

No Spatial

This package does not include support for the spatial API, described at http://apiv3.iucnredlist.org/spatial

Citing the Red List API

The citation is IUCN 2015. IUCN Red List of Threatened Species. Version 2015-4 <www.iucnredlist.org>
You can get this programatically via `rl_citation()`

Rate limiting

From the IUCN folks: Too many frequent calls, or too many calls per day might get your access blocked temporarily. If you’re a heavy API user, the Red List Unit asked that you contact them, as there might be better options. They suggest a 2-second delay between your calls if you plan to make a lot of calls.

Data Discrepancy

There can sometimes be a discrepancy between what you get on the IUCN website and what you get with this package; we don’t know why, the IUCN API is not an open book.

Citing the IUCN Red List API

See http://apiv3.iucnredlist.org/about

Author(s)

Scott Chamberlain <myrmecocystus@gmail.com>

---

### `rl_citation`  
*Get the citation Red List API version*

**Description**

Get the citation Red List API version

**Usage**

`rl_citation(...)`
Arguments

... Curl options passed to `cru::verb-GET`

Value

API citation as character string

Examples

```r
## Not run:
rl_citation()

## End(Not run)
```

---

**rl_common_names**

Get common names for a given taxonomic name

Description

Get common names for a given taxonomic name

Usage

```r
rl_common_names(name = NULL, key = NULL, parse = TRUE, ...)

rl_common_names_(name = NULL, key = NULL, ...)
```

Arguments

- `name` (character) Binomial taxonomic name
- `key` A IUCN API token. See `rl_use_iucn`
- `parse` (logical) Whether to parse to list (FALSE) or data.frame (TRUE). Default: TRUE
- `...` Curl options passed to `HttpClient`

Value

A list, with the data in the result slot, unless using a function with a trailing underscore, in which case json as character string is returned.

References

Examples

## Not run:
rl_common_names('Loxodonta africana')
## End(Not run)

rl_comp_groups

Description

Information about comprehensive groups

Usage

rl_comp_groups(group = NULL, key = NULL, parse = TRUE, ...)

rl_comp_groups_(group = NULL, key = NULL, ...)

Arguments

group (character) A comprehensive group name. Call rl_comp_groups() without passing this parameter to get the list of comprehensive groups

key A IUCN API token. See rl_use_iucn.

parse (logical) Whether to parse to list (FALSE) or data.frame (TRUE). Default: TRUE

... Curl options passed to HttpClient

Value

A list, with the data in the result slot, unless using a function with a trailing underscore, in which case json as character string is returned.

References

API docs at http://apiv3.iucnredlist.org/api/v3/docs

Examples

## Not run:
rl_comp_groups()
rl_comp_groups('mammals')
rl_comp_groups('groupers')
rl_comp_groups_()
rl_comp_groups_('groupers')

## End(Not run)
### rl_countries

**Get countries**

**Description**

Get countries

**Usage**

```r
rl_countries(key = NULL, parse = TRUE, ...)
```

```r
rl_countries_(key = NULL, ...)
```

**Arguments**

- `key` A IUCN API token. See `rl_use_iucn`.
- `parse` (logical) Whether to parse to list (FALSE) or data.frame (TRUE). Default: TRUE
- `...` Curl options passed to `HttpClient`

**Value**

A list, with the data in the `result` slot, unless using a function with a trailing underscore, in which case json as character string is returned.

**References**


**Examples**

```r
## Not run:
rl_countries()
rl_countries()
rl_countries_()
## End(Not run)
```

---

### rl_growth_forms

**Get plant species growth forms by taxon name, IUCN id, and region**

**Description**

Get plant species growth forms by taxon name, IUCN id, and region
**Usage**

```r
rl_growth_forms(  
  name = NULL,  
  id = NULL,  
  region = NULL,  
  key = NULL,  
  parse = TRUE,  
  ...  
)
```

```r
rl_growth_forms_(name = NULL, id = NULL, region = NULL, key = NULL, ...)
```

**Arguments**

- **name** (character) A taxonomic name
- **id** (character) An IUCN identifier
- **region** (character) A region name, see `rl_regions` for acceptable region identifiers (use the entries in the identifier column)
- **key** A IUCN API token. See `rl_use_iucn`.
- **parse** (logical) Whether to parse to list (FALSE) or data.frame (TRUE). Default: TRUE
- **...** Curl options passed to `HttpClient`

**Value**

A list, with the data in the result slot, unless using a function with a trailing underscore, in which case json as character string is returned.

**References**


**Examples**

```r
## Not run:  
rl_growth_forms("Quercus robur")  
rl_growth_forms("Quercus robur", region = "europe")  
rl_growth_forms(id = 63532)  
rl_growth_forms(id = 63532, region = "europe")
```

```r
rl_growth_forms("Mucuna bracteata")  
rl_growth_forms("Abarema villifera")  
rl_growth_forms("Adansonia perrieri")  
rl_growth_forms("Adenostemma harlingii")
```

```r
rl_growth_forms_("Quercus robur")  
rl_growth_forms_(id = 63532, region = "europe")
```

## End(Not run)
**rl_habitats**

*Get species habitats by taxon name, IUCN id, and region*

**Description**

Get species habitats by taxon name, IUCN id, and region

**Usage**

```r
rl_habitats(
  name = NULL,
  id = NULL,
  region = NULL,
  key = NULL,
  parse = TRUE,
  ...
)
rl_habitats_(name = NULL, id = NULL, region = NULL, key = NULL, ...)
rl_history_(name = NULL, id = NULL, region = NULL, key = NULL, ...)
```

**Arguments**

- **name** (character) A taxonomic name
- **id** (character) An IUCN identifier
- **region** (character) A region name, see `rl_regions` for acceptable region identifiers (use the entries in the identifier column)
- **key** A IUCN API token. See `rl_use_iucn`.
- **parse** (logical) Whether to parse to list (FALSE) or data.frame (TRUE). Default: TRUE
- **...** Curl options passed to `HttpClient`

**Value**

A list, with the data in the result slot, unless using a function with a trailing underscore, in which case json as character string is returned.

**References**

Examples

```r
## Not run:
rl_habitats('Fratercula arctica')
rl_habitats('Fratercula arctica', region = 'europe')
rl_habitats(id = 12392)
rl_habitats(id = 22694927, region = 'europe')
rl_habitats_('Fratercula arctica')
rl_habitats_(id = 12392)

## End(Not run)
```

Description

Get historical assessments by taxon name, IUCN id, and region

Usage

```r
rl_history(
  name = NULL,
  id = NULL,
  region = NULL,
  key = NULL,
  parse = TRUE,
  ...
)
```

Arguments

- `name` (character) A taxonomic name
- `id` (character) An IUCN identifier
- `region` (character) A region name, see `rl_regions` for acceptable region identifiers (use the entries in the identifier column)
- `key` A IUCN API token. See `rl_use_iucn`.
- `parse` (logical) Whether to parse to list (FALSE) or data.frame (TRUE). Default: TRUE
- `...` Curl options passed to `HttpClient`

Value

A list, with the data in the `result` slot, unless using a function with a trailing underscore, in which case json as character string is returned.
### rlr_measures

**Get species conservation measures by taxon name, IUCN id, and region**

**Description**

Get species conservation measures by taxon name, IUCN id, and region

**Usage**

```r
rl_measures(
  name = NULL,
  id = NULL,
  region = NULL,
  key = NULL,
  parse = TRUE,
  ...
)
```

```r
rl_measures_(name = NULL, id = NULL, region = NULL, key = NULL, ...)
```

**Arguments**

- **name** (character) A taxonomic name
- **id** (character) An IUCN identifier
- **region** (character) A region name, see `rl_regions` for acceptable region identifiers (use the entries in the identifier column)
- **key** A IUCN API token. See `rl_use_iucn`
- **parse** (logical) Whether to parse to list (FALSE) or data.frame (TRUE). Default: TRUE
- **...** Curl options passed to `HttpClient`
Value

A list, with the data in the result slot, unless using a function with a trailing underscore, in which case json as character string is returned.

References


Examples

```r
rl_measures('Fratercula arctica')
rl_measures('Fratercula arctica', region = 'europe')
rl_measures(id = 12392)
rl_measures(id = 22694927, region = 'europe')
rl_measures_('Fratercula arctica')
rl_measures_(id = 22694927, region = 'europe')
```

## END(Not run)

---

### rl_narrative

*Get species narrative information by taxon name, IUCN id, and region*

**Description**

Get species narrative information by taxon name, IUCN id, and region

**Usage**

```r
rl_narrative(
  name = NULL,
  id = NULL,
  region = NULL,
  key = NULL,
  parse = TRUE,
  ...
)
```

**Arguments**

- `name` (character) A taxonomic name
- `id` (character) An IUCN identifier
- `region` (character) A region name, see `rl_regions` for acceptable region identifiers (use the entries in the identifier column)
rl_occ_country

Get country occurrence by species name or ID

Description

Get country occurrence by species name or ID

Usage

```r
rl_occ_country(  
  name = NULL,  
  id = NULL,  
  region = NULL,  
  key = NULL,  
  parse = TRUE,  
  ...  
)
```

Examples

```r
## Not run:  
rl_narrative('Fratercula arctica')  
rl_narrative('Fratercula arctica', region = 'europe')  
rl_narrative(id = 12392)  
rl_narrative(id = 22694927, region = 'europe')  

rl_narrative_('Fratercula arctica')  
rl_narrative_('Fratercula arctica', region = 'europe')  

## End(Not run)
```

References


Value

A list, with the data in the `result` slot, unless using a function with a trailing underscore, in which case `json` as character string is returned.

key

A IUCN API token. See `rl_use_iucn`.

parse

(logical) Whether to parse to list (FALSE) or data.frame (TRUE). Default: TRUE

...  

Curl options passed to `HttpClient`
Arguments

- **name**: (character) A taxonomic name
- **id**: (character) An IUCN identifier
- **region**: (character) A region name, see `rl_regions` for acceptable region identifiers (use the entries in the identifier column)
- **key**: A IUCN API token. See `rl_use_iucn`.
- **parse**: (logical) Whether to parse to list (FALSE) or data.frame (TRUE). Default: TRUE
- **...**: Curl options passed to `HttpClient`

Value

A list, with the data in the result slot, unless using a function with a trailing underscore, in which case json as character string is returned.

References


Examples

```r
## Not run:
rl_occ_country('Loxodonta africana')
rl_occ_country('Fratercula arctica', region = 'europe')
rl_occ_country(id = 12392)
rl_occ_country(id = 22694927, region = 'europe')
rl_occ_country('Fratercula arctica', parse = FALSE)
rl_occ_country_('Fratercula arctica')
rl_occ_country_('Fratercula arctica', region = 'europe')
## End(Not run)
```

---

**Description**

Search by taxon name, IUCN id, and region

**Usage**

```r
rl_occ_country_(name = NULL, id = NULL, region = NULL, key = NULL, ...)
rl_search(name = NULL, id = NULL, region = NULL, key = NULL, parse = TRUE, ...)
rl_search_(name = NULL, id = NULL, region = NULL, key = NULL, ...)
```
Arguments

- **name** (character) A taxonomic name
- **id** (character) An IUCN identifier
- **region** (character) A region name, see `rl_regions` for acceptable region identifiers (use the entries in the identifier column)
- **key** A IUCN API token. See `rl_use_iucn`.
- **parse** (logical) Whether to parse to list (FALSE) or data.frame (TRUE). Default: TRUE

Value

A list, with the data in the result slot, unless using a function with a trailing underscore, in which case json as character string is returned.

References


Examples

```r
## Not run:
rl_search("Fratercula arctica")
rl_search("Fratercula arctica", region = 'europe')
rl_search(id = 12392)
rl_search(id = 22694927, region = 'europe')
rl_search("Fratercula arctica", parse = FALSE)
rl_search_("Fratercula arctica")
rl_search_("Fratercula arctica", region = 'europe')
## End(Not run)
```

---

`rl_regions` *Get regions*

**Description**

Get regions

**Usage**

```r
rl_regions(key = NULL, parse = TRUE, ...)
```

```r
rl_regions_(key = NULL, ...)
```
**Arguments**

- **key**  
  A IUCN API token. See `rl_use_iucn`.

- **parse**  
  (logical) Whether to parse to list (FALSE) or data.frame (TRUE). Default: TRUE

- **...**  
  Curl options passed to `HttpClient`.

**Value**

A list, with the data in the `result` slot, unless using a function with a trailing underscore, in which case json as character string is returned.

**References**


**Examples**

```r
## Not run:
rl_regions()
rl_regions(parse = FALSE)
rl_regions()

## End(Not run)
```

---

**rl_sp**  
*Get species*

**Description**

Get species

**Usage**

```r
generate:
rl_sp(page = 0, key = NULL, parse = TRUE, all = FALSE, quiet = FALSE, ...)

rl_sp_(page, key = NULL, all = FALSE, quiet = FALSE, ...)
```

**Arguments**

- **page**  
  (integer/numeric) Page to get. Default: 0. you can get up to 10,000 records per page. Paging is required because it’s too much burden on a server to just "get all the data" in one request

- **key**  
  A IUCN API token. See `rl_use_iucn`.

- **parse**  
  (logical) Whether to parse to list (FALSE) or data.frame (TRUE). Default: TRUE

- **all**  
  (logical) to get all results or not. Default: FALSE. this means we do the paging internally for you. result is a list of results, so you have to bind them together yourself into a data.frame, see example
quiet (logical) give progress for download or not. Default: FALSE (that is, give progress).

... Curl options passed to HttpClient

Examples

## Not run:
rl_sp(page = 3)

# get all results
out <- rl_sp(all = TRUE)
length(out)
vapply(out, "[["", 1, "count")
all_df <- do.call(rbind, lapply(out, "[[", "result"))
head(all_df)
NROW(all_df)

## End(Not run)

rl_sp_category

Get species by category

Description

Get species by category

Usage

rl_sp_category(category, key = NULL, parse = TRUE, ...)
rl_sp_category_(category, key = NULL, parse = TRUE, ...)

Arguments


key A IUCN API token. See rl_use_iucn.

parse (logical) Whether to parse to list (FALSE) or data.frame (TRUE). Default: TRUE

... Curl options passed to HttpClient

Examples

## Not run:
rl_sp_category('VU')
rl_sp_category('LRlc')
rl_sp_category('EN')
rl_sp_category('EX')
rl_sp_category('EX', parse = FALSE)
rl_sp_citation

## End(Not run)

---

**rl_sp_citation**  
*Get citations by taxon name, IUCN id, and region*

### Description

Get citations by taxon name, IUCN id, and region

### Usage

```r
rl_sp_citation(
  name = NULL,
  id = NULL,
  region = NULL,
  key = NULL,
  parse = TRUE,
  ...
)
```

### Arguments

- **name** (character) A taxonomic name
- **id** (character) An IUCN identifier
- **region** (character) A region name, see `rl_regions` for acceptable region identifiers (use the entries in the identifier column)
- **key** A IUCN API token. See `rl_use_iucn`.
- **parse** (logical) Whether to parse to list (FALSE) or data.frame (TRUE). Default: TRUE
- **...** Curl options passed to `HttpClient`

### Value

A list, with the data in the `result` slot, unless using a function with a trailing underscore, in which case json as character string is returned.

### References

Examples

```r
## Not run:
rl_sp_citation('Balaena mysticetus')
rl_sp_citation('Balaena mysticetus', region = 'europe')
rl_sp_citation(id = 12392)

rl_sp_citation(id = 2467, region = 'europe')
rl_sp_citation(id = 2467, region = 'europe', parse = FALSE)
rl_sp_citation_(id = 2467, region = 'europe')

## End(Not run)
```

## rl_sp_count

Get total species count of taxa in the Red List

### Description

Get total species count of taxa in the Red List

### Usage

```r
rl_sp_count(key = NULL, parse = TRUE, ...)
rl_sp_count_(key = NULL, ...)
```

### Arguments

- **key**: A IUCN API token. See `rl_use_iucn`.
- **parse**: (logical) Whether to parse to list (FALSE) or data.frame (TRUE). Default: TRUE
- **...**: Curl options passed to `HttpClient`

### Value

A list, with the data in the `result` slot, unless using a function with a trailing underscore, in which case json as character string is returned.

### References


### Examples

```r
## Not run:
rl_sp_count()
rl_sp_count_()

## End(Not run)
```
rl_sp_country

Get species by country

Description
Get species by country

Usage
rl_sp_country(country, key = NULL, parse = TRUE, ...)
rl_sp_country_(country, key = NULL, ...)

Arguments

- **country** (character) A two-letter country code. See isocode column in result of `rl_countries()` request for country codes.
- **key** A IUCN API token. See `rl_use_iucn`.
- **parse** (logical) Whether to parse to list (FALSE) or data.frame (TRUE). Default: TRUE
- **...** Curl options passed to `HttpClient`

Value
A list, with the data in the result slot, unless using a function with a trailing underscore, in which case json as character string is returned.

References
API docs at http://apiv3.iucnredlist.org/api/v3/docs

Examples

```r
## Not run:
rl_sp_country('AZ')
rl_sp_country('NZ')
#
rl_sp_country_('AZ')
rl_sp_country_('NZ', parse = FALSE)
#
rl_sp_country_('NZ', parse = TRUE)
```

# curl options
```r
res <- rl_sp_country_('NZ', verbose = TRUE)
```
**rl_synonyms**

Get species synonym information by taxonomic name

### Description

Get species synonym information by taxonomic name

### Usage

```r
rl_synonyms(name = NULL, key = NULL, parse = TRUE, ...)
rl_synonyms_(name = NULL, key = NULL, ...)
```

### Arguments

- **name** (character) Binomial taxonomic name
- **key** A IUCN API token. See `rl_use_iucn`
- **parse** (logical) Whether to parse to list (FALSE) or data.frame (TRUE). Default: TRUE
- **...** Curl options passed to `HttpClient`

### Value

A list, with the data in the `result` slot, unless using a function with a trailing underscore, in which case json as character string is returned.

### References


### Examples

```r
## Not run:
rl_synonyms('Loxodonta africana')
rl_synonyms('Loxodonta africana', parse = FALSE)
rl_synonyms_('Loxodonta africana')

## End(Not run)
```
**rl_threats**  
*Get species threats by taxon name, IUCN id, and region*

**Description**
Get species threats by taxon name, IUCN id, and region

**Usage**

```r
rl_threats(name = NULL, id = NULL, region = NULL, key = NULL, parse = TRUE, ...)
```

**Arguments**

- `name` (character) A taxonomic name
- `id` (character) An IUCN identifier
- `region` (character) A region name, see `rl_regions` for acceptable region identifiers (use the entries in the identifier column)
- `key` A IUCN API token. See `rl_use_iucn`.
- `parse` (logical) Whether to parse to list (FALSE) or data.frame (TRUE). Default: TRUE
- `...` Curl options passed to `HttpClient`

**Value**
A list, with the data in the result slot, unless using a function with a trailing underscore, in which case json as character string is returned.

**References**

**Examples**

```r
## Not run:
rl_threats("Fratercula arctica")
rl_threats("Fratercula arctica", region = "europe")
rl_threats(id = 12392)
rl_threats(id = 22694927, region = "europe")
```
rl_threats(name = 'Abies numidica')
rl_threats_('Fratercula arctica')
rl_threats(id = 62290750)

## End(Not run)

---

rl_use_iucn  

*Helper to get and save IUCN API key*

**Description**

Browse IUCN Red List API key request URL and provides instruction on how to store the key.

**Usage**

rl_use_iucn()

**Details**

Note that after filling the online form, you should receive an API key shortly but not immediately.

---

rl_version  

*Get the Red List API version*

**Description**

Get the Red List API version

**Usage**

rl_version(...)

**Arguments**

...  

Curl options passed to `crul::verb-GET`

**Value**

API version as character string

**Examples**

## Not run:
rl_version()

## End(Not run)
Index

*Topic package
  rredlist-package, 2

crul::verb-GET, 4, 22

HttpClient, 4–10, 12–21

rl_citation, 3
rl_citation(), 3
rl_common_names, 4
rl_common_names_(rl_common_names), 4
rl_comp_groups, 5
rl_comp_groups_(rl_comp_groups), 5
rl_countries, 6
rl_countries(), 19
rl_countries_(rl_countries), 6
rl_growth_forms, 6
rl_growth_forms_(rl_growth_forms), 6
rl_habitats, 8
rl_habitats_(rl_habitats), 8
rl_history, 9
rl_history_(rl_habitats), 8
rl_measures, 10
rl_measures_(rl_measures), 10
rl_narrative, 11
rl_narrative_(rl_narrative), 11
rl_occ_country, 12
rl_occ_country_, 13
rl_regions, 7–11, 13, 14, 17, 21
rl_regions_(rl_regions), 14
rl_search_(rl_occ_country_), 13
rl_search(), 3
rl_search_(rl_occ_country_), 13
rl_search(), 3
rl_sp, 15
rl_sp_(rl_sp), 15
rl_sp_category, 16
rl_sp_category_(rl_sp_category), 16
rl_sp_citation, 17
rl_sp_citation_(rl_sp_citation), 17
rl_sp_count, 18
rl_sp_count_(rl_sp_count), 18
rl_sp_country, 19
rl_sp_country_(rl_sp_country), 19
rl_synonyms, 20
rl_synonyms_(rl_synonyms), 20
rl_threats, 21
rl_threats_(rl_threats), 21
rl_use_iucn, 4–10, 12–21, 22
rl_use_iucn(), 2
rl_version, 22
rredlist(rredlist-package), 2
rredlist-package, 2