

Package ‘rredlist’

September 4, 2025

Type Package

Title 'IUCN' Red List Client

Description 'IUCN' Red List (<<https://api.iucnredlist.org/>>) 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 1.1.1

License MIT + file LICENSE

URL <https://docs.ropensci.org/rredlist/>,
<https://github.com/ropensci/rredlist>

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

Encoding UTF-8

Language en-US

Imports crul (>= 0.3.8), jsonlite (>= 1.1), lifecycle, cli, curl,
rlang

Suggests spelling, testthat, vcr (>= 0.6.0), knitr, rmarkdown,
webmockr, microbenchmark, ggplot2, dplyr, tibble, tidyr,
patchwork, vdiffr

RoxygenNote 7.3.2.9000

VignetteBuilder knitr

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 William Gearty [aut, cre] (ORCID:
<<https://orcid.org/0000-0003-0076-3262>>),
Scott Chamberlain [aut] (ORCID:
<<https://orcid.org/0000-0003-1444-9135>>),
rOpenSci [fnd] (ROR: <<https://ror.org/019jywm96>>),
Maëlle Salmon [ctb] (ORCID: <<https://orcid.org/0000-0002-2815-0399>>),

Kevin Cazelles [rev] (ORCID: <<https://orcid.org/0000-0001-6619-9874>>),
 Stephanie Hazlitt [rev] (ORCID:
 <<https://orcid.org/0000-0002-3161-2304>>)

Maintainer William Gearty <willgearty@gmail.com>

Repository CRAN

Date/Publication 2025-09-04 07:20:02 UTC

Contents

rl_actions	3
rl_api_version	4
rl_assessment	5
rl_assessment_extract	6
rl_assessment_list	8
rl_categories	9
rl_citation	11
rl_class	12
rl_comp_groups	14
rl_countries	16
rl_extinct	18
rl_extinct_wild	19
rl_family	20
rl_faos	22
rl_green	24
rl_growth_forms	25
rl_habitats	27
rl_kingdom	28
rl_order	30
rl_phylum	32
rl_pop_trends	34
rl_realms	36
rl_research	37
rl_scopes	39
rl_sis	41
rl_sis_latest	42
rl_species	43
rl_species_latest	45
rl_sp_count	47
rl_stresses	48
rl_systems	49
rl_threats	51
rl_use_and_trade	53
rl_use_iucn	55
rl_version	55
scale_color_iucn	56

Index	58
--------------	-----------

rl_actions

Conservation actions needed assessment summary

Description

Return the latest assessments with a given conservation action needed (e.g., Land/water management or Species recovery). These conservation action codes correspond to the IUCN Red List Conservation Actions Classification Scheme (v2.0).

Usage

```
rl_actions(
  code = NULL,
  key = NULL,
  parse = TRUE,
  all = TRUE,
  page = 1,
  quiet = FALSE,
  ...
)
```

```
rl_actions_(code = NULL, key = NULL, all = TRUE, page = 1, quiet = FALSE, ...)
```

Arguments

code	(character) The code of the conservation action to look up. If not supplied, a list of all conservation actions will be returned.
key	(character) An IUCN API token. See rl_use_iucn() .
parse	(logical) Whether to parse the output to list (FALSE) or, where possible, data.frame (TRUE). Default: TRUE.
all	(logical) Whether to retrieve all results at once or not. If TRUE we do the paging internally for you and bind all of the results together. If FALSE, only a single page of results will be retrieved.
page	(integer/numeric) Page to get if all is FALSE. Default: 1. Each page returns up to 100 records. Paging is required because it's too much burden on a server to just "get all the data" in one request.
quiet	(logical) Whether to suppress progress for multi-page downloads or not. Default: FALSE (that is, give progress). Ignored if all = FALSE.
...	Includes the following filters that are supported by the API endpoint (refer to the API docs for more information): - year_published: (integer) Set this to return only assessments from a given year. - latest: (logical) Set this to TRUE to return only the latest assessment for each taxon.

- `scope_code`: (integer) Set this to return only assessments from a particular [scope](#) (e.g., 1 for Global, 2 for Europe). This is similar to the `region` argument of the old Red List API and old versions of `redlist`.

Also includes the following arguments related to the wait time between request retries if a "Too Many Requests" error is received from the API (see [HttpClient\(\)\\$retry\(\)](#) for more details):

- `pause_base`, `pause_cap`, and `pause_min`: basis, maximum, and minimum for calculating wait time for retry
- `times`: the maximum number of times to retry
- `onwait`: a callback function if the request will be retried and a wait time is being applied

Also supports any [curl options](#) passed to the GET request via [HttpClient](#).

Value

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

References

API docs at <https://api.iucnredlist.org/>.

See Also

Assessments by conservation actions or research needed: [rl_research\(\)](#)

Examples

```
## Not run:
# Get list of all conservation actions
rl_actions()
# Get assessments with Species recovery conservation action needed
rl_actions("3_2")

## End(Not run)
```

rl_api_version

Get the Red List API version

Description

Returns the current version number of the IUCN Red List of Threatened Species API.

Usage

```
rl_api_version(key = NULL, ...)
```

Arguments

key (character) An IUCN API token. See [rl_use_iucn\(\)](#).
 ... Curl options passed to [HttpClient](#).

Value

API version as character string.

See Also

Red List information and statistics: [rl_citation\(\)](#), [rl_sp_count\(\)](#), [rl_version\(\)](#)

Examples

```
## Not run:
rl_api_version()

## End(Not run)
```

rl_assessment	<i>Retrieve an assessment</i>
---------------	-------------------------------

Description

Get the full details for a single IUCN Red List assessment.

Usage

```
rl_assessment(id, key = NULL, parse = TRUE, ...)

rl_assessment_(id, key = NULL, ...)
```

Arguments

id (integer) The unique identifier of the assessment.
 key (character) An IUCN API token. See [rl_use_iucn\(\)](#).
 parse (logical) Whether to parse the output to list (FALSE) or, where possible, data.frame (TRUE). Default: TRUE.
 ... Includes the following arguments related to the wait time between request retries if a "Too Many Requests" error is received from the API (see [HttpClient\(\)\\$retry\(\)](#) for more details):

- pause_base, pause_cap, and pause_min: basis, maximum, and minimum for calculating wait time for retry
- times: the maximum number of times to retry
- onwait: a callback function if the request will be retried and a wait time is being applied

Also supports any [curl options](#) passed to the GET request via [HttpClient](#).

Value

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

References

API docs at <https://api.iucnredlist.org/>.

Examples

```
## Not run:
# Get assessment details for Fratercula arctica
ex1 <- rl_assessment(id = 166290968)
ex1$red_list_category$code
ex1$systems

## End(Not run)
```

rl_assessment_extract *Extract an assessment element*

Description

Extract a given element from each of a list of assessments, such as the output of `rl_assessment_list()`. This is useful for extracting specific details from the assessments, such as taxonomy (`el_name = "taxon"`), synonyms (`el_name = "taxon__synonyms"`), habitats (`el_name = "habitats"`), or the red list category (`el_name = "red_list_category"`).

Usage

```
rl_assessment_extract(lst, el_name, format = c("list", "df"), flatten = FALSE)
```

Arguments

lst	(list) A list of assessments, as returned by <code>rl_assessment_list()</code> . If lst contains any NULL elements, they will be removed.
el_name	(character) The name of the element to extract from each assessment. Supports multilevel extraction using "__" as the separator. For example, to extract the synonyms table, you could use "taxon__synonyms". el_name may also be a character vector indicating multiple elements to extract. In this case, the output will contain all requested elements, either as a list (if format = "list") or as a data.frame (if format = "df"). In the case of a data.frame, the extracted elements will be joined by assessment_id, which may result in an unexpected merged outcome when multiple rows are present in one or more of the extracted elements (e.g., the "threats" element).
format	(character) The format of the output. Either "list" or "df" (for a data.frame).

flatten (logical) If TRUE, the output will be flattened to a data.frame. Note that this may not work for all elements, especially complex multilevel list elements. Only used when `format = "df"`. The `dplyr`, `tidyr`, and `tibble` packages are required to use this feature. Note that fields with no data across all assessments may be lost.

Details

The following top-level element names can be extracted as of 2025-08-29:

- `assessment_date`
- `assessment_id`
- `assessment_points`
- `assessment_ranges`
- `biogeographical_realms`
- `citation`
- `conservation_actions`
- `credits`
- `criteria`
- `documentation`
- `errata`
- `faos`
- `growth_forms`
- `habitats`
- `latest`
- `lmes`
- `locations`
- `population_trend`
- `possibly_extinct`
- `possibly_extinct_in_the_wild`
- `red_list_category`
- `references`
- `researches`
- `scopes`
- `sis_taxon_id`
- `stresses`
- `supplementary_info`
- `systems`
- `taxon`
- `threats`

- url
- use_and_trade
- year_published

Note that there are also sublevels of the assessment hierarchy, but they would be unwieldy to display here. A complete and up-to-date list can be found by inspecting the return of an `rl_assessment()` call.

Value

A list or data.frame containing the extracted element from each assessment.

Examples

```
## Not run:
lst <- rl_assessment_list(ids = c(166290968, 136250858))
# get complex elements as a list
ex1 <- rl_assessment_extract(lst, "taxon")
# get simple elements as a data.frame
ex2 <- rl_assessment_extract(lst, "red_list_category__code", format = "df")
# get complex elements as a data.frame
ex3 <- rl_assessment_extract(lst, "threats", format = "df")
# get the same elements flattened to a single data.frame
ex4 <- rl_assessment_extract(lst, "threats", format = "df", flatten = TRUE)
# get subelements flattened to a data.frame
ex5 <- rl_assessment_extract(lst, "taxon__order_name", format = "df",
                             flatten = TRUE)
# get a data frame with both taxon name and red list category code
ex6 <- rl_assessment_extract(lst, c("taxon", "red_list_category__code"),
                             format = "df", flatten = TRUE)

## End(Not run)
```

rl_assessment_list	<i>Retrieve a list of assessments</i>
--------------------	---------------------------------------

Description

Get the full details for a list of IUCN Red List assessments. This is a wrapper for `rl_assessment()` that allows you to retrieve multiple assessments at once.

Usage

```
rl_assessment_list(ids, key = NULL, wait_time = 0.5, quiet = FALSE, ...)
```


Arguments

ids	(integer) A vector of unique identifiers of the assessments.
key	(character) An IUCN API token. See rl_use_iucn() .
wait_time	(numeric) The number of seconds to wait between queries. The default is 0.5 seconds, which is recommended by IUCN to avoid timeouts.
quiet	(logical) If TRUE, suppresses the progress bar.
...	Includes the following arguments related to the wait time between request retries if a "Too Many Requests" error is received from the API (see HttpClient()\$retry() for more details): • pause_base, pause_cap, and pause_min: basis, maximum, and minimum for calculating wait time for retry • times: the maximum number of times to retry • onwait: a callback function if the request will be retried and a wait time is being applied

Also supports any [curl options](#) passed to the GET request via [HttpClient](#).

Value

A list with each element representing the response of [rl_assessment\(\)](#).

Examples

```
## Not run:
# Get assessment details for multiple assessments
ex1 <- rl_assessment_list(ids = c(166290968, 136250858))

## End(Not run)
```

rl_categories	<i>Red List category assessment summary</i>
---------------	---

Description

Return a list of the latest assessments for a given category (e.g., Not Evaluated, Data Deficient, Least Concern, Near Threatened, Vulnerable, Endangered, Critically Endangered, Extinct in the Wild and Extinct). This endpoint returns categories for the current IUCN Red List Categories and Criteria (v3.1) as well as older versions (i.e., v2.3). Note that a code may not be unique across Categories and Criteria versions. Therefore, codes like "EX" will return assessments for EX v3.1 and EX v2.3.

Usage

```

rl_categories(
  code = NULL,
  key = NULL,
  parse = TRUE,
  all = TRUE,
  page = 1,
  quiet = FALSE,
  ...
)

rl_categories_(
  code = NULL,
  key = NULL,
  all = TRUE,
  page = 1,
  quiet = FALSE,
  ...
)

```

Arguments

code	(character) The code of the Red List category to look up. If not supplied, a list of all categories will be returned.
key	(character) An IUCN API token. See rl_use_iucn() .
parse	(logical) Whether to parse the output to list (FALSE) or, where possible, data.frame (TRUE). Default: TRUE.
all	(logical) Whether to retrieve all results at once or not. If TRUE we do the paging internally for you and bind all of the results together. If FALSE, only a single page of results will be retrieved.
page	(integer/numeric) Page to get if all is FALSE. Default: 1. Each page returns up to 100 records. Paging is required because it's too much burden on a server to just "get all the data" in one request.
quiet	(logical) Whether to suppress progress for multi-page downloads or not. Default: FALSE (that is, give progress). Ignored if all = FALSE.
...	Includes the following filters that are supported by the API endpoint (refer to the API docs for more information): • year_published: (integer) Set this to return only assessments from a given year. • latest: (logical) Set this to TRUE to return only the latest assessment for each taxon. • scope_code: (integer) Set this to return only assessments from a particular scope (e.g., 1 for Global, 2 for Europe). This is similar to the region argument of the old Red List API and old versions of redlist.

Also includes the following arguments related to the wait time between request retries if a "Too Many Requests" error is received from the API (see [HttpClient\(\)\\$retry\(\)](#) for more details):

- `pause_base`, `pause_cap`, and `pause_min`: basis, maximum, and minimum for calculating wait time for retry
- `times`: the maximum number of times to retry
- `onwait`: a callback function if the request will be retried and a wait time is being applied

Also supports any [curl options](#) passed to the GET request via [HttpClient](#).

Value

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

References

API docs at <https://api.iucnredlist.org/>.

See Also

Assessments by threat details: [rl_pop_trends\(\)](#), [rl_stresses\(\)](#), [rl_threats\(\)](#), [rl_use_and_trade\(\)](#)

Examples

```
## Not run:
# Get all Red List categories
rl_categories()
# Get assessments for the Vulnerable category
rl_categories("VU")

## End(Not run)
```

rl_citation

Get the citation for the Red List API

Description

Full acknowledgement and citation needs to be given for using the API. Use this function to get the full citation for the current version of the Red List API. More details are available here: <https://api.iucnredlist.org/>.

Usage

```
rl_citation(key = NULL, ...)
```

Arguments

key (character) An IUCN API token. See [rl_use_iucn\(\)](#).
 ... Curl options passed to [HttpClient](#).

Value

Red List citation as a [bibentry](#) object.

See Also

Red List information and statistics: [rl_api_version\(\)](#), [rl_sp_count\(\)](#), [rl_version\(\)](#)

Examples

```
## Not run:
rl_citation()

## End(Not run)
```

rl_class	<i>Class assessment summary</i>
----------	---------------------------------

Description

Get an assessment summary for a particular class

Usage

```
rl_class(
  class = NULL,
  key = NULL,
  parse = TRUE,
  all = TRUE,
  page = 1,
  quiet = FALSE,
  ...
)

rl_class_(class = NULL, key = NULL, all = TRUE, page = 1, quiet = FALSE, ...)
```

Arguments

class (character) The name of the class to look up. If not supplied, a list of all class names will be returned.
 key (character) An IUCN API token. See [rl_use_iucn\(\)](#).
 parse (logical) Whether to parse the output to list (FALSE) or, where possible, data.frame (TRUE). Default: TRUE.

all	(logical) Whether to retrieve all results at once or not. If TRUE we do the paging internally for you and bind all of the results together. If FALSE, only a single page of results will be retrieved.
page	(integer/numeric) Page to get if all is FALSE. Default: 1. Each page returns up to 100 records. Paging is required because it's too much burden on a server to just "get all the data" in one request.
quiet	(logical) Whether to suppress progress for multi-page downloads or not. Default: FALSE (that is, give progress). Ignored if all = FALSE.
...	Includes the following filters that are supported by the API endpoint (refer to the API docs for more information): • year_published: (integer) Set this to return only assessments from a given year. • latest: (logical) Set this to TRUE to return only the latest assessment for each taxon. • scope_code: (integer) Set this to return only assessments from a particular scope (e.g., 1 for Global, 2 for Europe). This is similar to the region argument of the old Red List API and old versions of redlist. Also includes the following arguments related to the wait time between request retries if a "Too Many Requests" error is received from the API (see HttpClient()\$retry() for more details): • pause_base, pause_cap, and pause_min: basis, maximum, and minimum for calculating wait time for retry • times: the maximum number of times to retry • onwait: a callback function if the request will be retried and a wait time is being applied

Also supports any [curl options](#) passed to the GET request via [HttpClient](#).

Value

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

References

API docs at <https://api.iucnredlist.org/>.

See Also

Assessments by taxa: [rl_family\(\)](#), [rl_kingdom\(\)](#), [rl_order\(\)](#), [rl_phylum\(\)](#), [rl_sis\(\)](#), [rl_sis_latest\(\)](#), [rl_species\(\)](#), [rl_species_latest\(\)](#)

Examples

```
## Not run:
# Get assessment summary for class
ex1 <- rl_class(class = "Mammalia")
nrow(ex1$assessments)
```

```
## End(Not run)
```

```
rl_comp_groups
```

```
Comprehensive group assessment summary
```

Description

Returns a list of the latest assessments for a comprehensive group name.

Usage

```
rl_comp_groups(
  name = NULL,
  key = NULL,
  parse = TRUE,
  all = TRUE,
  page = 1,
  quiet = FALSE,
  ...
)
```

```
rl_comp_groups_(
  name = NULL,
  key = NULL,
  all = TRUE,
  page = 1,
  quiet = FALSE,
  ...
)
```

Arguments

name	(character) The code of the comprehensive group to look up. If not supplied, a list of all comprehensive groups will be returned.
key	(character) An IUCN API token. See rl_use_iucn() .
parse	(logical) Whether to parse the output to list (FALSE) or, where possible, data.frame (TRUE). Default: TRUE.
all	(logical) Whether to retrieve all results at once or not. If TRUE we do the paging internally for you and bind all of the results together. If FALSE, only a single page of results will be retrieved.
page	(integer/numeric) Page to get if all is FALSE. Default: 1. Each page returns up to 100 records. Paging is required because it's too much burden on a server to just "get all the data" in one request.
quiet	(logical) Whether to suppress progress for multi-page downloads or not. Default: FALSE (that is, give progress). Ignored if all = FALSE.

...

Includes the following filters that are supported by the API endpoint (refer to the [API docs](#) for more information):

- `year_published`: (integer) Set this to return only assessments from a given year.
- `latest`: (logical) Set this to TRUE to return only the latest assessment for each taxon.
- `scope_code`: (integer) Set this to return only assessments from a particular [scope](#) (e.g., 1 for Global, 2 for Europe). This is similar to the `region` argument of the old Red List API and old versions of `redlist`.

Also includes the following arguments related to the wait time between request retries if a "Too Many Requests" error is received from the API (see [HttpClient\(\)\\$retry\(\)](#) for more details):

- `pause_base`, `pause_cap`, and `pause_min`: basis, maximum, and minimum for calculating wait time for retry
- `times`: the maximum number of times to retry
- `onwait`: a callback function if the request will be retried and a wait time is being applied

Also supports any [curl options](#) passed to the GET request via [HttpClient](#).

Value

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

References

API docs at <https://api.iucnredlist.org/>.

See Also

Assessments for specific sets of taxa [rl_extinct\(\)](#), [rl_extinct_wild\(\)](#), [rl_green\(\)](#), [rl_growth_forms\(\)](#)

Examples

```
## Not run:
# Get list of all comprehensive groups
rl_comp_groups()
# Get assessment summary for sea snakes
rl_comp_groups('seasnakes')

## End(Not run)
```

rl_countries

Country assessment summary

Description

Returns a collection of assessments for a given country code.

Usage

```
rl_countries(
  code = NULL,
  key = NULL,
  parse = TRUE,
  all = TRUE,
  page = 1,
  quiet = FALSE,
  ...
)

rl_countries_(
  code = NULL,
  key = NULL,
  all = TRUE,
  page = 1,
  quiet = FALSE,
  ...
)
```

Arguments

code	(character) The ISO alpha-2 code of the country to look up. If not supplied, a list of all countries will be returned.
key	(character) An IUCN API token. See rl_use_iucn() .
parse	(logical) Whether to parse the output to list (FALSE) or, where possible, data.frame (TRUE). Default: TRUE.
all	(logical) Whether to retrieve all results at once or not. If TRUE we do the paging internally for you and bind all of the results together. If FALSE, only a single page of results will be retrieved.
page	(integer/numeric) Page to get if all is FALSE. Default: 1. Each page returns up to 100 records. Paging is required because it's too much burden on a server to just "get all the data" in one request.
quiet	(logical) Whether to suppress progress for multi-page downloads or not. Default: FALSE (that is, give progress). Ignored if all = FALSE.
...	Includes the following filters that are supported by the API endpoint (refer to the API docs for more information):

- `year_published`: (integer) Set this to return only assessments from a given year.
- `latest`: (logical) Set this to TRUE to return only the latest assessment for each taxon.
- `scope_code`: (integer) Set this to return only assessments from a particular [scope](#) (e.g., 1 for Global, 2 for Europe). This is similar to the `region` argument of the old Red List API and old versions of `redlist`.

Also includes the following arguments related to the wait time between request retries if a "Too Many Requests" error is received from the API (see [HttpClient\(\)\\$retry\(\)](#) for more details):

- `pause_base`, `pause_cap`, and `pause_min`: basis, maximum, and minimum for calculating wait time for retry
- `times`: the maximum number of times to retry
- `onwait`: a callback function if the request will be retried and a wait time is being applied

Also supports any [curl options](#) passed to the GET request via [HttpClient](#).

Value

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

References

API docs at <https://api.iucnredlist.org/>.

See Also

Assessments by geographic regions: [rl_faos\(\)](#), [rl_realms\(\)](#), [rl_scopes\(\)](#)

Examples

```
## Not run:
# Get list of all countries
rl_countries()
# Get assessments for Madagascar
rl_countries("MG")

## End(Not run)
```

rl_extinct	<i>Extinct taxa assessment summary</i>
------------	--

Description

Get an assessment summary for all possibly extinct taxa

Usage

```
rl_extinct(key = NULL, parse = TRUE, all = TRUE, page = 1, quiet = FALSE, ...)
```

```
rl_extinct_(key = NULL, all = TRUE, page = 1, quiet = FALSE, ...)
```

Arguments

key	(character) An IUCN API token. See rl_use_iucn() .
parse	(logical) Whether to parse the output to list (FALSE) or, where possible, data.frame (TRUE). Default: TRUE.
all	(logical) Whether to retrieve all results at once or not. If TRUE we do the paging internally for you and bind all of the results together. If FALSE, only a single page of results will be retrieved.
page	(integer/numeric) Page to get if all is FALSE. Default: 1. Each page returns up to 100 records. Paging is required because it's too much burden on a server to just "get all the data" in one request.
quiet	(logical) Whether to suppress progress for multi-page downloads or not. Default: FALSE (that is, give progress). Ignored if all = FALSE.
...	Includes the following arguments related to the wait time between request retries if a "Too Many Requests" error is received from the API (see HttpClient()\$retry() for more details): • pause_base, pause_cap, and pause_min: basis, maximum, and minimum for calculating wait time for retry • times: the maximum number of times to retry • onwait: a callback function if the request will be retried and a wait time is being applied

Also supports any [curl options](#) passed to the GET request via [HttpClient](#).

Value

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

References

API docs at <https://api.iucnredlist.org/>.

See Also

Assessments for specific sets of taxa [rl_comp_groups\(\)](#), [rl_extinct_wild\(\)](#), [rl_green\(\)](#), [rl_growth_forms\(\)](#)

Examples

```
## Not run:
ex1 <- rl_extinct()
nrow(ex1$assessments)

## End(Not run)
```

rl_extinct_wild	<i>Extinct taxa in the wild assessment summary</i>
-----------------	--

Description

Get an assessment summary for all taxa that are possibly extinct in the wild

Usage

```
rl_extinct_wild(
  key = NULL,
  parse = TRUE,
  all = TRUE,
  page = 1,
  quiet = FALSE,
  ...
)

rl_extinct_wild_(key = NULL, all = TRUE, page = 1, quiet = FALSE, ...)
```

Arguments

key	(character) An IUCN API token. See rl_use_iucn() .
parse	(logical) Whether to parse the output to list (FALSE) or, where possible, data.frame (TRUE). Default: TRUE.
all	(logical) Whether to retrieve all results at once or not. If TRUE we do the paging internally for you and bind all of the results together. If FALSE, only a single page of results will be retrieved.
page	(integer/numeric) Page to get if all is FALSE. Default: 1. Each page returns up to 100 records. Paging is required because it's too much burden on a server to just "get all the data" in one request.
quiet	(logical) Whether to suppress progress for multi-page downloads or not. Default: FALSE (that is, give progress). Ignored if all = FALSE.

... Includes the following arguments related to the wait time between request retries if a "Too Many Requests" error is received from the API (see [HttpClient\(\)\\$retry\(\)](#) for more details):

- `pause_base`, `pause_cap`, and `pause_min`: basis, maximum, and minimum for calculating wait time for retry
- `times`: the maximum number of times to retry
- `onwait`: a callback function if the request will be retried and a wait time is being applied

Also supports any [curl options](#) passed to the GET request via [HttpClient](#).

Value

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

References

API docs at <https://api.iucnredlist.org/>.

See Also

Assessments for specific sets of taxa [rl_comp_groups\(\)](#), [rl_extinct\(\)](#), [rl_green\(\)](#), [rl_growth_forms\(\)](#)

Examples

```
## Not run:
ex1 <- rl_extinct_wild()
nrow(ex1$assessments)

## End(Not run)
```

rl_family

Family assessment summary

Description

Get an assessment summary for a particular family

Usage

```
rl_family(
  family = NULL,
  key = NULL,
  parse = TRUE,
  all = TRUE,
  page = 1,
  quiet = FALSE,
```

```
    ...
  )
```

```
rl_family_(family = NULL, key = NULL, all = TRUE, page = 1, quiet = FALSE, ...)
```

Arguments

family	(character) The name of the family to look up. If not supplied, a list of all family names will be returned.
key	(character) An IUCN API token. See rl_use_iucn() .
parse	(logical) Whether to parse the output to list (FALSE) or, where possible, data.frame (TRUE). Default: TRUE.
all	(logical) Whether to retrieve all results at once or not. If TRUE we do the paging internally for you and bind all of the results together. If FALSE, only a single page of results will be retrieved.
page	(integer/numeric) Page to get if all is FALSE. Default: 1. Each page returns up to 100 records. Paging is required because it's too much burden on a server to just "get all the data" in one request.
quiet	(logical) Whether to suppress progress for multi-page downloads or not. Default: FALSE (that is, give progress). Ignored if all = FALSE.
...	Includes the following filters that are supported by the API endpoint (refer to the API docs for more information):

- year_published: (integer) Set this to return only assessments from a given year.
- latest: (logical) Set this to TRUE to return only the latest assessment for each taxon.
- scope_code: (integer) Set this to return only assessments from a particular [scope](#) (e.g., 1 for Global, 2 for Europe). This is similar to the region argument of the old Red List API and old versions of rredlist.

Also includes the following arguments related to the wait time between request retries if a "Too Many Requests" error is received from the API (see [HttpClient\(\)\\$retry\(\)](#) for more details):

- pause_base, pause_cap, and pause_min: basis, maximum, and minimum for calculating wait time for retry
- times: the maximum number of times to retry
- onwait: a callback function if the request will be retried and a wait time is being applied

Also supports any [curl options](#) passed to the GET request via [HttpClient](#).

Value

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

References

API docs at <https://api.iucnredlist.org/>.

See Also

Assessments by taxa: `rl_class()`, `rl_kingdom()`, `rl_order()`, `rl_phylum()`, `rl_sis()`, `rl_sis_latest()`, `rl_species()`, `rl_species_latest()`

Examples

```
## Not run:
# Get assessment summary for family
ex1 <- rl_family(family = "Hominidae")
nrow(ex1$assessments)

## End(Not run)
```

rl_faos

FAO marine fishing region assessment summary

Description

Return assessments for a given marine fishing region as defined by the Food and Agriculture Organization (FAO) of the United Nations (e.g., Pacific - northwest, Arctic Sea). More details are available here: <https://www.fao.org/fishery/en/area>.

Usage

```
rl_faos(
  code = NULL,
  key = NULL,
  parse = TRUE,
  all = TRUE,
  page = 1,
  quiet = FALSE,
  ...
)
```

```
rl_faos_(code = NULL, key = NULL, all = TRUE, page = 1, quiet = FALSE, ...)
```

Arguments

code	(character) The code of the FAO region to look up. If not supplied, a list of all FAO regions will be returned.
key	(character) An IUCN API token. See <code>rl_use_iucn()</code> .
parse	(logical) Whether to parse the output to list (FALSE) or, where possible, data.frame (TRUE). Default: TRUE.
all	(logical) Whether to retrieve all results at once or not. If TRUE we do the paging internally for you and bind all of the results together. If FALSE, only a single page of results will be retrieved.

page	(integer/numeric) Page to get if all is FALSE. Default: 1. Each page returns up to 100 records. Paging is required because it's too much burden on a server to just "get all the data" in one request.
quiet	(logical) Whether to suppress progress for multi-page downloads or not. Default: FALSE (that is, give progress). Ignored if all = FALSE.
...	Includes the following filters that are supported by the API endpoint (refer to the API docs for more information): • year_published: (integer) Set this to return only assessments from a given year. • latest: (logical) Set this to TRUE to return only the latest assessment for each taxon. • scope_code: (integer) Set this to return only assessments from a particular scope (e.g., 1 for Global, 2 for Europe). This is similar to the region argument of the old Red List API and old versions of rredlist. Also includes the following arguments related to the wait time between request retries if a "Too Many Requests" error is received from the API (see HttpClient()\$retry() for more details): • pause_base, pause_cap, and pause_min: basis, maximum, and minimum for calculating wait time for retry • times: the maximum number of times to retry • onwait: a callback function if the request will be retried and a wait time is being applied Also supports any curl options passed to the GET request via HttpClient.

Value

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

References

API docs at <https://api.iucnredlist.org/>.

See Also

Assessments by geographic regions: [rl_countries\(\)](#), [rl_realms\(\)](#), [rl_scopes\(\)](#)

Examples

```
## Not run:
# Get a list of all scopes
rl_faos()
# Get assessment summary for the Arctic Sea FAO region
rl_faos(code = "18")

## End(Not run)
```

rl_green

Green Status assessment summary

Description

List all Green Status assessments.

Usage

```
rl_green(key = NULL, parse = TRUE, ...)
```

```
rl_green_(key = NULL, ...)
```

Arguments

- | | |
|-------|---|
| key | (character) An IUCN API token. See rl_use_iucn() . |
| parse | (logical) Whether to parse the output to list (FALSE) or, where possible, data.frame (TRUE). Default: TRUE. |
| ... | Includes the following arguments related to the wait time between request retries if a "Too Many Requests" error is received from the API (see HttpClient()\$retry() for more details): <ul style="list-style-type: none"> • pause_base, pause_cap, and pause_min: basis, maximum, and minimum for calculating wait time for retry • times: the maximum number of times to retry • onwait: a callback function if the request will be retried and a wait time is being applied |

Also supports any [curl options](#) passed to the GET request via [HttpClient](#).

Value

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

References

API docs at <https://api.iucnredlist.org/>.

See Also

Assessments for specific sets of taxa [rl_comp_groups\(\)](#), [rl_extinct\(\)](#), [rl_extinct_wild\(\)](#), [rl_growth_forms\(\)](#)

Examples

```
## Not run:
# Get list of Green Status assessments
rl_green()

## End(Not run)
```

rl_growth_forms

Growth form assessment summary

Description

Returns a list of the latest assessments for a given growth form code.

Usage

```
rl_growth_forms(
  code = NULL,
  key = NULL,
  parse = TRUE,
  all = TRUE,
  page = 1,
  quiet = FALSE,
  ...
)

rl_growth_forms_(
  code = NULL,
  key = NULL,
  all = TRUE,
  page = 1,
  quiet = FALSE,
  ...
)
```

Arguments

code	(character) The code of the growth form to look up. If not supplied, a list of all growth forms will be returned.
key	(character) An IUCN API token. See rl_use_iucn() .
parse	(logical) Whether to parse the output to list (FALSE) or, where possible, data.frame (TRUE). Default: TRUE.
all	(logical) Whether to retrieve all results at once or not. If TRUE we do the paging internally for you and bind all of the results together. If FALSE, only a single page of results will be retrieved.

page	(integer/numeric) Page to get if all is FALSE. Default: 1. Each page returns up to 100 records. Paging is required because it's too much burden on a server to just "get all the data" in one request.
quiet	(logical) Whether to suppress progress for multi-page downloads or not. Default: FALSE (that is, give progress). Ignored if all = FALSE.
...	Includes the following filters that are supported by the API endpoint (refer to the API docs for more information): • year_published: (integer) Set this to return only assessments from a given year. • latest: (logical) Set this to TRUE to return only the latest assessment for each taxon. • scope_code: (integer) Set this to return only assessments from a particular scope (e.g., 1 for Global, 2 for Europe). This is similar to the region argument of the old Red List API and old versions of redlist. Also includes the following arguments related to the wait time between request retries if a "Too Many Requests" error is received from the API (see HttpClient()\$retry() for more details): • pause_base, pause_cap, and pause_min: basis, maximum, and minimum for calculating wait time for retry • times: the maximum number of times to retry • onwait: a callback function if the request will be retried and a wait time is being applied

Also supports any [curl options](#) passed to the GET request via [HttpClient](#).

Value

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

References

API docs at <https://api.iucnredlist.org/>.

See Also

Assessments for specific sets of taxa [rl_comp_groups\(\)](#), [rl_extinct\(\)](#), [rl_extinct_wild\(\)](#), [rl_green\(\)](#)

Examples

```
## Not run:
# Get list of all growth forms
rl_growth_forms()
# Get assessment summary for lichens
rl_growth_forms("LC")

## End(Not run)
```

rl_habitats

*Habitat assessment summary***Description**

Return the latest assessments for a given habitat (e.g., Forest - Temperate or Marine Intertidal). These habitat codes correspond to the IUCN Red List Habitats Classification Scheme (v3.1).

Usage

```
rl_habitats(
  code = NULL,
  key = NULL,
  parse = TRUE,
  all = TRUE,
  page = 1,
  quiet = FALSE,
  ...
)
```

```
rl_habitats_(code = NULL, key = NULL, all = TRUE, page = 1, quiet = FALSE, ...)
```

Arguments

code	(character) The code of the habitat to look up. If not supplied, a list of all habitats will be returned.
key	(character) An IUCN API token. See rl_use_iucn() .
parse	(logical) Whether to parse the output to list (FALSE) or, where possible, data.frame (TRUE). Default: TRUE.
all	(logical) Whether to retrieve all results at once or not. If TRUE we do the paging internally for you and bind all of the results together. If FALSE, only a single page of results will be retrieved.
page	(integer/numeric) Page to get if all is FALSE. Default: 1. Each page returns up to 100 records. Paging is required because it's too much burden on a server to just "get all the data" in one request.
quiet	(logical) Whether to suppress progress for multi-page downloads or not. Default: FALSE (that is, give progress). Ignored if all = FALSE.
...	Includes the following filters that are supported by the API endpoint (refer to the API docs for more information): • year_published: (integer) Set this to return only assessments from a given year. • latest: (logical) Set this to TRUE to return only the latest assessment for each taxon.

- `scope_code`: (integer) Set this to return only assessments from a particular [scope](#) (e.g., 1 for Global, 2 for Europe). This is similar to the `region` argument of the old Red List API and old versions of `redlist`.

Also includes the following arguments related to the wait time between request retries if a "Too Many Requests" error is received from the API (see [HttpClient\(\)\\$retry\(\)](#) for more details):

- `pause_base`, `pause_cap`, and `pause_min`: basis, maximum, and minimum for calculating wait time for retry
- `times`: the maximum number of times to retry
- `onwait`: a callback function if the request will be retried and a wait time is being applied

Also supports any [curl options](#) passed to the GET request via [HttpClient](#).

Value

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

References

API docs at <https://api.iucnredlist.org/>.

See Also

Assessments by habitats and systems: [rl_systems\(\)](#)

Examples

```
## Not run:
# Get list of all habitats
rl_habitats()
# Get assessments for Marine Intertidal habitat
rl_habitats("12")

## End(Not run)
```

rl_kingdom

Kingdom assessment summary

Description

Get an assessment summary for a particular kingdom

Usage

```

rl_kingdom(
  kingdom = NULL,
  key = NULL,
  parse = TRUE,
  all = TRUE,
  page = 1,
  quiet = FALSE,
  ...
)

rl_kingdom_(
  kingdom = NULL,
  key = NULL,
  all = TRUE,
  page = 1,
  quiet = FALSE,
  ...
)

```

Arguments

kingdom	(character) The name of the kingdom to look up. If not supplied, a list of all kingdom names will be returned.
key	(character) An IUCN API token. See rl_use_iucn() .
parse	(logical) Whether to parse the output to list (FALSE) or, where possible, data.frame (TRUE). Default: TRUE.
all	(logical) Whether to retrieve all results at once or not. If TRUE we do the paging internally for you and bind all of the results together. If FALSE, only a single page of results will be retrieved.
page	(integer/numeric) Page to get if all is FALSE. Default: 1. Each page returns up to 100 records. Paging is required because it's too much burden on a server to just "get all the data" in one request.
quiet	(logical) Whether to suppress progress for multi-page downloads or not. Default: FALSE (that is, give progress). Ignored if all = FALSE.
...	Includes the following filters that are supported by the API endpoint (refer to the API docs for more information): • year_published: (integer) Set this to return only assessments from a given year. • latest: (logical) Set this to TRUE to return only the latest assessment for each taxon. • scope_code: (integer) Set this to return only assessments from a particular scope (e.g., 1 for Global, 2 for Europe). This is similar to the region argument of the old Red List API and old versions of redlist.

Also includes the following arguments related to the wait time between request retries if a "Too Many Requests" error is received from the API (see [HttpClient\(\)\\$retry\(\)](#) for more details):

- `pause_base`, `pause_cap`, and `pause_min`: basis, maximum, and minimum for calculating wait time for retry
- `times`: the maximum number of times to retry
- `onwait`: a callback function if the request will be retried and a wait time is being applied

Also supports any [curl options](#) passed to the GET request via [HttpClient](#).

Value

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

References

API docs at <https://api.iucnredlist.org/>.

See Also

Assessments by taxa: [rl_class\(\)](#), [rl_family\(\)](#), [rl_order\(\)](#), [rl_phylum\(\)](#), [rl_sis\(\)](#), [rl_sis_latest\(\)](#), [rl_species\(\)](#), [rl_species_latest\(\)](#)

Examples

```
## Not run:
# Get assessment summary for kingdom
ex1 <- rl_kingdom(kingdom = "Fungi")
nrow(ex1$assessments)

## End(Not run)
```

rl_order

Order assessment summary

Description

Get an assessment summary for a particular order

Usage

```
rl_order(
  order = NULL,
  key = NULL,
  parse = TRUE,
  all = TRUE,
```

```

    page = 1,
    quiet = FALSE,
    ...
)

rl_order_(order = NULL, key = NULL, all = TRUE, page = 1, quiet = FALSE, ...)
```

Arguments

order	(character) The name of the order to look up. If not supplied, a list of all order names will be returned.
key	(character) An IUCN API token. See rl_use_iucn() .
parse	(logical) Whether to parse the output to list (FALSE) or, where possible, data.frame (TRUE). Default: TRUE.
all	(logical) Whether to retrieve all results at once or not. If TRUE we do the paging internally for you and bind all of the results together. If FALSE, only a single page of results will be retrieved.
page	(integer/numeric) Page to get if all is FALSE. Default: 1. Each page returns up to 100 records. Paging is required because it's too much burden on a server to just "get all the data" in one request.
quiet	(logical) Whether to suppress progress for multi-page downloads or not. Default: FALSE (that is, give progress). Ignored if all = FALSE.
...	Includes the following filters that are supported by the API endpoint (refer to the API docs for more information): • year_published: (integer) Set this to return only assessments from a given year. • latest: (logical) Set this to TRUE to return only the latest assessment for each taxon. • scope_code: (integer) Set this to return only assessments from a particular scope (e.g., 1 for Global, 2 for Europe). This is similar to the region argument of the old Red List API and old versions of redlist.

Also includes the following arguments related to the wait time between request retries if a "Too Many Requests" error is received from the API (see [HttpClient\(\)\\$retry\(\)](#) for more details):

- pause_base, pause_cap, and pause_min: basis, maximum, and minimum for calculating wait time for retry
- times: the maximum number of times to retry
- onwait: a callback function if the request will be retried and a wait time is being applied

Also supports any [curl options](#) passed to the GET request via [HttpClient](#).

Value

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

References

API docs at <https://api.iucnredlist.org/>.

See Also

Assessments by taxa: [rl_class\(\)](#), [rl_family\(\)](#), [rl_kingdom\(\)](#), [rl_phylum\(\)](#), [rl_sis\(\)](#), [rl_sis_latest\(\)](#), [rl_species\(\)](#), [rl_species_latest\(\)](#)

Examples

```
## Not run:
# Get assessment summary for order
ex1 <- rl_order(order = "Apiales")
nrow(ex1$assessments)

## End(Not run)
```

rl_phylum	<i>Phylum assessment summary</i>
-----------	----------------------------------

Description

Get an assessment summary for a particular phylum

Usage

```
rl_phylum(
  phylum = NULL,
  key = NULL,
  parse = TRUE,
  all = TRUE,
  page = 1,
  quiet = FALSE,
  ...
)

rl_phylum_(phylum = NULL, key = NULL, all = TRUE, page = 1, quiet = FALSE, ...)
```

Arguments

phylum	(character) The name of the phylum to look up. If not supplied, a list of all phylum names will be returned.
key	(character) An IUCN API token. See rl_use_iucn() .
parse	(logical) Whether to parse the output to list (FALSE) or, where possible, data.frame (TRUE). Default: TRUE.

all	(logical) Whether to retrieve all results at once or not. If TRUE we do the paging internally for you and bind all of the results together. If FALSE, only a single page of results will be retrieved.
page	(integer/numeric) Page to get if all is FALSE. Default: 1. Each page returns up to 100 records. Paging is required because it's too much burden on a server to just "get all the data" in one request.
quiet	(logical) Whether to suppress progress for multi-page downloads or not. Default: FALSE (that is, give progress). Ignored if all = FALSE.
...	Includes the following filters that are supported by the API endpoint (refer to the API docs for more information): • year_published: (integer) Set this to return only assessments from a given year. • latest: (logical) Set this to TRUE to return only the latest assessment for each taxon. • scope_code: (integer) Set this to return only assessments from a particular scope (e.g., 1 for Global, 2 for Europe). This is similar to the region argument of the old Red List API and old versions of redlist. Also includes the following arguments related to the wait time between request retries if a "Too Many Requests" error is received from the API (see HttpClient()\$retry() for more details): • pause_base, pause_cap, and pause_min: basis, maximum, and minimum for calculating wait time for retry • times: the maximum number of times to retry • onwait: a callback function if the request will be retried and a wait time is being applied

Also supports any [curl options](#) passed to the GET request via [HttpClient](#).

Value

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

References

API docs at <https://api.iucnredlist.org/>.

See Also

Assessments by taxa: [rl_class\(\)](#), [rl_family\(\)](#), [rl_kingdom\(\)](#), [rl_order\(\)](#), [rl_sis\(\)](#), [rl_sis_latest\(\)](#), [rl_species\(\)](#), [rl_species_latest\(\)](#)

Examples

```
## Not run:
# Get assessment summary for phylum
ex1 <- rl_phylum(phylum = "Annelida")
nrow(ex1$assessments)
```

```
## End(Not run)
```

rl_pop_trends	<i>Population trend assessment summary</i>
---------------	--

Description

Return a list of the latest assessments based on a population trend (i.e. increasing, decreasing, stable or unknown).

Usage

```
rl_pop_trends(  
  code = NULL,  
  key = NULL,  
  parse = TRUE,  
  all = TRUE,  
  page = 1,  
  quiet = FALSE,  
  ...  
)  
  
rl_pop_trends_(  
  code = NULL,  
  key = NULL,  
  all = TRUE,  
  page = 1,  
  quiet = FALSE,  
  ...  
)
```

Arguments

- code (character) The code of the growth form to look up. If not supplied, a list of all growth forms will be returned.
- key (character) An IUCN API token. See [rl_use_iucn\(\)](#).
- parse (logical) Whether to parse the output to list (FALSE) or, where possible, data.frame (TRUE). Default: TRUE.
- all (logical) Whether to retrieve all results at once or not. If TRUE we do the paging internally for you and bind all of the results together. If FALSE, only a single page of results will be retrieved.
- page (integer/numeric) Page to get if all is FALSE. Default: 1. Each page returns up to 100 records. Paging is required because it's too much burden on a server to just "get all the data" in one request.

- quiet** (logical) Whether to suppress progress for multi-page downloads or not. Default: FALSE (that is, give progress). Ignored if `all = FALSE`.
- ...** Includes the following filters that are supported by the API endpoint (refer to the [API docs](#) for more information):
- **year_published**: (integer) Set this to return only assessments from a given year.
 - **latest**: (logical) Set this to TRUE to return only the latest assessment for each taxon.
 - **scope_code**: (integer) Set this to return only assessments from a particular [scope](#) (e.g., 1 for Global, 2 for Europe). This is similar to the `region` argument of the old Red List API and old versions of `redlist`.
- Also includes the following arguments related to the wait time between request retries if a "Too Many Requests" error is received from the API (see [HttpClient\(\)\\$retry\(\)](#) for more details):
- **pause_base**, **pause_cap**, and **pause_min**: basis, maximum, and minimum for calculating wait time for retry
 - **times**: the maximum number of times to retry
 - **onwait**: a callback function if the request will be retried and a wait time is being applied

Also supports any [curl options](#) passed to the GET request via [HttpClient](#).

Value

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

References

API docs at <https://api.iucnredlist.org/>.

See Also

Assessments by threat details: [rl_categories\(\)](#), [rl_stresses\(\)](#), [rl_threats\(\)](#), [rl_use_and_trade\(\)](#)

Examples

```
## Not run:
# Get list of all population trends
rl_pop_trends()
# Get assessment summary for stable population trends
rl_pop_trends("2")

## End(Not run)
```

rl_realms

Biogeographical realm assessment summary

Description

Get an assessment summary for a particular biogeographical realm (e.g., Neotropical or Palearctic).

Usage

```
rl_realms(
  code = NULL,
  key = NULL,
  parse = TRUE,
  all = TRUE,
  page = 1,
  quiet = FALSE,
  ...
)
```

```
rl_realms_(code = NULL, key = NULL, all = TRUE, page = 1, quiet = FALSE, ...)
```

Arguments

code	(character) The code of the biogeographical realm to look up. If not supplied, a list of all biogeographical realms will be returned.
key	(character) An IUCN API token. See rl_use_iucn() .
parse	(logical) Whether to parse the output to list (FALSE) or, where possible, data.frame (TRUE). Default: TRUE.
all	(logical) Whether to retrieve all results at once or not. If TRUE we do the paging internally for you and bind all of the results together. If FALSE, only a single page of results will be retrieved.
page	(integer/numeric) Page to get if all is FALSE. Default: 1. Each page returns up to 100 records. Paging is required because it's too much burden on a server to just "get all the data" in one request.
quiet	(logical) Whether to suppress progress for multi-page downloads or not. Default: FALSE (that is, give progress). Ignored if all = FALSE.
...	Includes the following filters that are supported by the API endpoint (refer to the API docs for more information): • year_published: (integer) Set this to return only assessments from a given year. • latest: (logical) Set this to TRUE to return only the latest assessment for each taxon. • scope_code: (integer) Set this to return only assessments from a particular scope (e.g., 1 for Global, 2 for Europe). This is similar to the region argument of the old Red List API and old versions of redlist.

Also includes the following arguments related to the wait time between request retries if a "Too Many Requests" error is received from the API (see [HttpClient\(\)\\$retry\(\)](#) for more details):

- `pause_base`, `pause_cap`, and `pause_min`: basis, maximum, and minimum for calculating wait time for retry
- `times`: the maximum number of times to retry
- `onwait`: a callback function if the request will be retried and a wait time is being applied

Also supports any [curl options](#) passed to the GET request via [HttpClient](#).

Value

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

References

API docs at <https://api.iucnredlist.org/>.

See Also

Assessments by geographic regions: [rl_countries\(\)](#), [rl_faos\(\)](#), [rl_scopes\(\)](#)

Examples

```
## Not run:
# Get a list of all biogeographical realms
rl_realms()
# Get assessment summary for the Afrotropical biogeographical realm
rl_realms(code = "0")

## End(Not run)
```

rl_research

Research needed assessment summary

Description

Return the latest assessments with a given research needed (e.g., Taxonomy or Population trends). These research codes correspond to the IUCN Red List Research Needed Classification Scheme (v1.0).

Usage

```
rl_research(
  code = NULL,
  key = NULL,
  parse = TRUE,
  all = TRUE,
  page = 1,
  quiet = FALSE,
  ...
)
```

```
rl_research_(code = NULL, key = NULL, all = TRUE, page = 1, quiet = FALSE, ...)
```

Arguments

code	(character) The code of the research type to look up. If not supplied, a list of all research types will be returned.
key	(character) An IUCN API token. See rl_use_iucn() .
parse	(logical) Whether to parse the output to list (FALSE) or, where possible, data.frame (TRUE). Default: TRUE.
all	(logical) Whether to retrieve all results at once or not. If TRUE we do the paging internally for you and bind all of the results together. If FALSE, only a single page of results will be retrieved.
page	(integer/numeric) Page to get if all is FALSE. Default: 1. Each page returns up to 100 records. Paging is required because it's too much burden on a server to just "get all the data" in one request.
quiet	(logical) Whether to suppress progress for multi-page downloads or not. Default: FALSE (that is, give progress). Ignored if all = FALSE.
...	Includes the following filters that are supported by the API endpoint (refer to the API docs for more information): • year_published: (integer) Set this to return only assessments from a given year. • latest: (logical) Set this to TRUE to return only the latest assessment for each taxon. • scope_code: (integer) Set this to return only assessments from a particular scope (e.g., 1 for Global, 2 for Europe). This is similar to the region argument of the old Red List API and old versions of redlist.

Also includes the following arguments related to the wait time between request retries if a "Too Many Requests" error is received from the API (see [HttpClient\(\)\\$retry\(\)](#) for more details):

- pause_base, pause_cap, and pause_min: basis, maximum, and minimum for calculating wait time for retry
- times: the maximum number of times to retry
- onwait: a callback function if the request will be retried and a wait time is being applied

Also supports any [curl options](#) passed to the GET request via [HttpClient](#).

Value

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

References

API docs at <https://api.iucnredlist.org/>.

See Also

Assessments by conservation actions or research needed: [rl_actions\(\)](#)

Examples

```
## Not run:
# Get list of all research types
rl_research()
# Get assessments with Taxonomy research needed
rl_research("1_1")

## End(Not run)
```

rl_scopes	<i>Scope assessment summary</i>
-----------	---------------------------------

Description

Return assessments for a given geographical assessment scope (e.g., Global, Mediterranean). This is similar to the `region` argument of the old Red List API and old versions of `redlist`.

Usage

```
rl_scopes(
  code = NULL,
  key = NULL,
  parse = TRUE,
  all = TRUE,
  page = 1,
  quiet = FALSE,
  ...
)

rl_scopes_(code = NULL, key = NULL, all = TRUE, page = 1, quiet = FALSE, ...)
```

Arguments

code	(character) The code of the scope to look up. If not supplied, a list of all scopes will be returned.
key	(character) An IUCN API token. See rl_use_iucn() .
parse	(logical) Whether to parse the output to list (FALSE) or, where possible, data.frame (TRUE). Default: TRUE.
all	(logical) Whether to retrieve all results at once or not. If TRUE we do the paging internally for you and bind all of the results together. If FALSE, only a single page of results will be retrieved.
page	(integer/numeric) Page to get if all is FALSE. Default: 1. Each page returns up to 100 records. Paging is required because it's too much burden on a server to just "get all the data" in one request.
quiet	(logical) Whether to suppress progress for multi-page downloads or not. Default: FALSE (that is, give progress). Ignored if all = FALSE.
...	Includes the following filters that are supported by the API endpoint (refer to the API docs for more information): • year_published: (integer) Set this to return only assessments from a given year. • latest: (logical) Set this to TRUE to return only the latest assessment for each taxon. • scope_code: (integer) Set this to return only assessments from a particular scope (e.g., 1 for Global, 2 for Europe). This is similar to the region argument of the old Red List API and old versions of redlist.

Also includes the following arguments related to the wait time between request retries if a "Too Many Requests" error is received from the API (see [HttpClient\(\)\\$retry\(\)](#) for more details):

- pause_base, pause_cap, and pause_min: basis, maximum, and minimum for calculating wait time for retry
- times: the maximum number of times to retry
- onwait: a callback function if the request will be retried and a wait time is being applied

Also supports any [curl options](#) passed to the GET request via [HttpClient](#).

Value

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

References

API docs at <https://api.iucnredlist.org/>.

See Also

Assessments by geographic regions: [rl_countries\(\)](#), [rl_faos\(\)](#), [rl_realms\(\)](#)

Examples

```
## Not run:
# Get a list of all scopes
rl_scopes()
# Get assessment summary for the Gulf of Mexico scope
rl_scopes(code = "45433062")

## End(Not run)
```

rl_sis

SIS ID assessment summary

Description

Get an assessment summary for a particular taxonomic entity based on its ID number from the [IUCN Species Information Service \(SIS\)](#).

Usage

```
rl_sis(id, key = NULL, parse = TRUE, ...)
```

```
rl_sis_(id, key = NULL, ...)
```

Arguments

- | | |
|-------|---|
| id | (integer) The SIS ID of the taxonomic entity to look up. |
| key | (character) An IUCN API token. See rl_use_iucn() . |
| parse | (logical) Whether to parse the output to list (FALSE) or, where possible, data.frame (TRUE). Default: TRUE. |
| ... | Includes the following arguments related to the wait time between request retries if a "Too Many Requests" error is received from the API (see HttpClient()\$retry() for more details): <ul style="list-style-type: none"> • pause_base, pause_cap, and pause_min: basis, maximum, and minimum for calculating wait time for retry • times: the maximum number of times to retry • onwait: a callback function if the request will be retried and a wait time is being applied |

Also supports any [curl options](#) passed to the GET request via [HttpClient](#).

Value

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

References

API docs at <https://api.iucnredlist.org/>.

See Also

Assessments by taxa: [rl_class\(\)](#), [rl_family\(\)](#), [rl_kingdom\(\)](#), [rl_order\(\)](#), [rl_phylum\(\)](#), [rl_sis_latest\(\)](#), [rl_species\(\)](#), [rl_species_latest\(\)](#)

Examples

```
## Not run:
# Get assessment summary for species
ex1 <- rl_sis(id = 9404)
nrow(ex1$assessments)

## End(Not run)
```

rl_sis_latest	<i>SIS ID latest assessment</i>
---------------	---------------------------------

Description

Get the latest assessment for a particular taxonomic entity based on its ID number from the [IUCN Species Information Service \(SIS\)](#). Wraps [rl_sis\(\)](#) and [rl_assessment\(\)](#).

Usage

```
rl_sis_latest(id, scope = "1", key = NULL, parse = TRUE, ...)
```

Arguments

- | | |
|-------|---|
| id | (integer) The SIS ID of the taxonomic entity to look up. |
| scope | (character) An optional scope (see rl_scopes()) to filter the results by. The default, "1", returns the latest global assessment. If no assessments exist for the specified scope, the latest assessment across all scopes is returned. If NULL, the latest assessment across all scopes is returned. |
| key | (character) An IUCN API token. See rl_use_iucn() . |
| parse | (logical) Whether to parse the output to list (FALSE) or, where possible, data.frame (TRUE). Default: TRUE. |
| ... | Includes the following arguments related to the wait time between request retries if a "Too Many Requests" error is received from the API (see HttpClient()\$retry() for more details): <ul style="list-style-type: none">• pause_base, pause_cap, and pause_min: basis, maximum, and minimum for calculating wait time for retry• times: the maximum number of times to retry• onwait: a callback function if the request will be retried and a wait time is being applied |

Also supports any [curl options](#) passed to the GET request via [HttpClient](#).

Value

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

References

API docs at <https://api.iucnredlist.org/>.

See Also

Assessments by taxa: `rl_class()`, `rl_family()`, `rl_kingdom()`, `rl_order()`, `rl_phylum()`, `rl_sis()`, `rl_species()`, `rl_species_latest()`

Examples

```
## Not run:
# Get latest assessment for species
ex1 <- rl_sis_latest(id = 9404)
ex1$stresses

## End(Not run)
```

rl_species

Species assessment summary

Description

Get an assessment summary for a particular species (i.e., Latin binomial) or subspecies/variety/subpopulation (i.e., Latin trinomial).

Usage

```
rl_species(
  genus,
  species,
  infra = NULL,
  subpopulation = NULL,
  key = NULL,
  parse = TRUE,
  ...
)

rl_species_(
  genus,
  species,
  infra = NULL,
  subpopulation = NULL,
```

```

    key = NULL,
    ...
)

```

Arguments

genus	(character) The genus name of the species to look up.
species	(character) The species epithet of the species to look up.
infra	(character) An optional name of the subspecies or variety to look up.
subpopulation	(character) An optional name of the geographically separate subpopulation to look up.
key	(character) An IUCN API token. See rl_use_iucn() .
parse	(logical) Whether to parse the output to list (FALSE) or, where possible, data.frame (TRUE). Default: TRUE.
...	Includes the following arguments related to the wait time between request retries if a "Too Many Requests" error is received from the API (see HttpClient()\$retry() for more details): • pause_base, pause_cap, and pause_min: basis, maximum, and minimum for calculating wait time for retry • times: the maximum number of times to retry • onwait: a callback function if the request will be retried and a wait time is being applied

Also supports any [curl options](#) passed to the GET request via [HttpClient](#).

Details

Geographically separate subpopulations of a species are defined as those populations that are so isolated from others of the same species that it is considered extremely unlikely that there is any genetic interchange. In general, listings of such subpopulations are restricted to those that have been isolated for a long period of time.

Assessments of subspecies, varieties, and geographically separate subpopulations must adhere to the same standards as for species assessments. However, these assessments are only included provided there is a global assessment of the species as a whole.

Intraspecific ranks such as formas, subvarieties, cultivars, etc are not included in the Red List.

Value

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

References

API docs at <https://api.iucnredlist.org/>.

See Also

Assessments by taxa: `rl_class()`, `rl_family()`, `rl_kingdom()`, `rl_order()`, `rl_phylum()`, `rl_sis()`, `rl_sis_latest()`, `rl_species_latest()`

Examples

```
## Not run:
# Get assessment summary for species
ex1 <- rl_species(genus = "Fratercula", species = "arctica")
nrow(ex1$assessments)

# Get assessment summary for subspecies
ex2 <- rl_species(genus = "Gorilla", species = "gorilla",
                  infra = "gorilla")
nrow(ex2$assessments)

## End(Not run)
```

rl_species_latest	<i>Species latest assessment</i>
-------------------	----------------------------------

Description

Get the latest assessment for a particular species (i.e., Latin binomial) or subspecies/variety/subpopulation (i.e., Latin trinomial). Wraps `rl_species()` and `rl_assessment()`.

Usage

```
rl_species_latest(
  genus,
  species,
  infra = NULL,
  subpopulation = NULL,
  scope = "1",
  key = NULL,
  parse = TRUE,
  ...
)
```

Arguments

genus	(character) The genus name of the species to look up.
species	(character) The species epithet of the species to look up.
infra	(character) An optional name of the subspecies or variety to look up.
subpopulation	(character) An optional name of the geographically separate subpopulation to look up.

scope	(character) An optional scope (see rl_scopes()) to filter the results by. The default, "1", returns the latest global assessment. If no assessments exist for the specified scope, the latest assessment across all scopes is returned. If NULL, the latest assessment across all scopes is returned.
key	(character) An IUCN API token. See rl_use_iucn() .
parse	(logical) Whether to parse the output to list (FALSE) or, where possible, data.frame (TRUE). Default: TRUE.
...	Includes the following arguments related to the wait time between request retries if a "Too Many Requests" error is received from the API (see HttpClient()\$retry() for more details): • pause_base, pause_cap, and pause_min: basis, maximum, and minimum for calculating wait time for retry • times: the maximum number of times to retry • onwait: a callback function if the request will be retried and a wait time is being applied

Also supports any [curl options](#) passed to the GET request via [HttpClient](#).

Details

Geographically separate subpopulations of a species are defined as those populations that are so isolated from others of the same species that it is considered extremely unlikely that there is any genetic interchange. In general, listings of such subpopulations are restricted to those that have been isolated for a long period of time.

Assessments of subspecies, varieties, and geographically separate subpopulations must adhere to the same standards as for species assessments. However, these assessments are only included provided there is a global assessment of the species as a whole.

Intraspecific ranks such as formas, subvarieties, cultivars, etc are not included in the Red List.

Value

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

References

API docs at <https://api.iucnredlist.org/>.

See Also

Assessments by taxa: [rl_class\(\)](#), [rl_family\(\)](#), [rl_kingdom\(\)](#), [rl_order\(\)](#), [rl_phylum\(\)](#), [rl_sis\(\)](#), [rl_sis_latest\(\)](#), [rl_species\(\)](#)

Examples

```
## Not run:
# Get latest assessment for species
ex1 <- rl_species_latest(genus = "Fratercula", species = "arctica")
```

```

ex1$stresses

# Get latest assessment for subspecies
ex2 <- rl_species_latest(genus = "Gorilla", species = "gorilla",
                        infra = "gorilla")
ex2$stresses

## End(Not run)

```

rl_sp_count

Get count of species in the Red List

Description

Returns a count of the number of unique species which have assessments.

Usage

```
rl_sp_count(key = NULL, ...)
```

```
rl_sp_count_(key = NULL, ...)
```

Arguments

key (character) An IUCN API token. See [rl_use_iucn\(\)](#).
 ... Curl options passed to [HttpClient](#)

Value

An integer representing the number of unique species represented within the IUCN database.

References

API docs at <https://api.iucnredlist.org/>.

See Also

Red List information and statistics: [rl_api_version\(\)](#), [rl_citation\(\)](#), [rl_version\(\)](#)

Examples

```

## Not run:
# Get count of species with assessments
rl_sp_count()

## End(Not run)

```

rl_stresses

Stress assessment summary

Description

Return a list of the latest assessments based on the stresses species are subject to (e.g., Ecosystem degradation, species disturbance, etc.).

Usage

```
rl_stresses(
  code = NULL,
  key = NULL,
  parse = TRUE,
  all = TRUE,
  page = 1,
  quiet = FALSE,
  ...
)
```

```
rl_stresses_(code = NULL, key = NULL, all = TRUE, page = 1, quiet = FALSE, ...)
```

Arguments

code	(character) The code of the stress to look up. If not supplied, a list of all stresses will be returned.
key	(character) An IUCN API token. See rl_use_iucn() .
parse	(logical) Whether to parse the output to list (FALSE) or, where possible, data.frame (TRUE). Default: TRUE.
all	(logical) Whether to retrieve all results at once or not. If TRUE we do the paging internally for you and bind all of the results together. If FALSE, only a single page of results will be retrieved.
page	(integer/numeric) Page to get if all is FALSE. Default: 1. Each page returns up to 100 records. Paging is required because it's too much burden on a server to just "get all the data" in one request.
quiet	(logical) Whether to suppress progress for multi-page downloads or not. Default: FALSE (that is, give progress). Ignored if all = FALSE.
...	Includes the following filters that are supported by the API endpoint (refer to the API docs for more information): - year_published: (integer) Set this to return only assessments from a given year. - latest: (logical) Set this to TRUE to return only the latest assessment for each taxon.

- `scope_code`: (integer) Set this to return only assessments from a particular [scope](#) (e.g., 1 for Global, 2 for Europe). This is similar to the region argument of the old Red List API and old versions of `redlist`.

Also includes the following arguments related to the wait time between request retries if a "Too Many Requests" error is received from the API (see [HttpClient\(\)\\$retry\(\)](#) for more details):

- `pause_base`, `pause_cap`, and `pause_min`: basis, maximum, and minimum for calculating wait time for retry
- `times`: the maximum number of times to retry
- `onwait`: a callback function if the request will be retried and a wait time is being applied

Also supports any [curl options](#) passed to the GET request via [HttpClient](#).

Value

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

References

API docs at <https://api.iucnredlist.org/>.

See Also

Assessments by threat details: [rl_categories\(\)](#), [rl_pop_trends\(\)](#), [rl_threats\(\)](#), [rl_use_and_trade\(\)](#)

Examples

```
## Not run:
# Get all stresses
rl_stresses()
# Get assessment summary for ecosystem degradation stress
rl_stresses("1_2")

## End(Not run)
```

rl_systems

System assessment summary

Description

Return the latest assessments for a given system (e.g., terrestrial, freshwater or marine).

Usage

```
rl_systems(
  code = NULL,
  key = NULL,
  parse = TRUE,
  all = TRUE,
  page = 1,
  quiet = FALSE,
  ...
)
```

```
rl_systems_(code = NULL, key = NULL, all = TRUE, page = 1, quiet = FALSE, ...)
```

Arguments

code	(character) The code of the system to look up. If not supplied, a list of all systems will be returned.
key	(character) An IUCN API token. See rl_use_iucn() .
parse	(logical) Whether to parse the output to list (FALSE) or, where possible, data.frame (TRUE). Default: TRUE.
all	(logical) Whether to retrieve all results at once or not. If TRUE we do the paging internally for you and bind all of the results together. If FALSE, only a single page of results will be retrieved.
page	(integer/numeric) Page to get if all is FALSE. Default: 1. Each page returns up to 100 records. Paging is required because it's too much burden on a server to just "get all the data" in one request.
quiet	(logical) Whether to suppress progress for multi-page downloads or not. Default: FALSE (that is, give progress). Ignored if all = FALSE.
...	Includes the following filters that are supported by the API endpoint (refer to the API docs for more information): • year_published: (integer) Set this to return only assessments from a given year. • latest: (logical) Set this to TRUE to return only the latest assessment for each taxon. • scope_code: (integer) Set this to return only assessments from a particular scope (e.g., 1 for Global, 2 for Europe). This is similar to the region argument of the old Red List API and old versions of redlist.

Also includes the following arguments related to the wait time between request retries if a "Too Many Requests" error is received from the API (see [HttpClient\(\)\\$retry\(\)](#) for more details):

- pause_base, pause_cap, and pause_min: basis, maximum, and minimum for calculating wait time for retry
- times: the maximum number of times to retry
- onwait: a callback function if the request will be retried and a wait time is being applied

Also supports any [curl options](#) passed to the GET request via [HttpClient](#).

Value

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

References

API docs at <https://api.iucnredlist.org/>.

See Also

Assessments by habitats and systems: [rl_habitats\(\)](#)

Examples

```
## Not run:
# Get list of all systems
rl_systems()
# Get assessment summary for marine system
rl_systems("2")

## End(Not run)
```

rl_threats

Threat assessment summary

Description

Return a list of the latest assessments which are subject to a specific threat (e.g., energy production and mining, climate change, and severe weather). This will only return assessments for the threat code specified. You will need to do additional requests for sub-threats (e.g., a request for threat code 2_1 will need additional requests for codes 2_1_1, 2_1_2, etc.).

Usage

```
rl_threats(
  code = NULL,
  key = NULL,
  parse = TRUE,
  all = TRUE,
  page = 1,
  quiet = FALSE,
  ...
)

rl_threats_(code = NULL, key = NULL, all = TRUE, page = 1, quiet = FALSE, ...)
```

Arguments

code	(character) The code of the threat to look up. If not supplied, a list of all threats will be returned.
key	(character) An IUCN API token. See rl_use_iucn() .
parse	(logical) Whether to parse the output to list (FALSE) or, where possible, data.frame (TRUE). Default: TRUE.
all	(logical) Whether to retrieve all results at once or not. If TRUE we do the paging internally for you and bind all of the results together. If FALSE, only a single page of results will be retrieved.
page	(integer/numeric) Page to get if all is FALSE. Default: 1. Each page returns up to 100 records. Paging is required because it's too much burden on a server to just "get all the data" in one request.
quiet	(logical) Whether to suppress progress for multi-page downloads or not. Default: FALSE (that is, give progress). Ignored if all = FALSE.
...	Includes the following filters that are supported by the API endpoint (refer to the API docs for more information): • year_published: (integer) Set this to return only assessments from a given year. • latest: (logical) Set this to TRUE to return only the latest assessment for each taxon. • scope_code: (integer) Set this to return only assessments from a particular scope (e.g., 1 for Global, 2 for Europe). This is similar to the region argument of the old Red List API and old versions of redlist. Also includes the following arguments related to the wait time between request retries if a "Too Many Requests" error is received from the API (see HttpClient()\$retry() for more details): • pause_base, pause_cap, and pause_min: basis, maximum, and minimum for calculating wait time for retry • times: the maximum number of times to retry • onwait: a callback function if the request will be retried and a wait time is being applied

Also supports any [curl options](#) passed to the GET request via [HttpClient](#).

Value

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

References

API docs at <https://api.iucnredlist.org/>.

See Also

Assessments by threat details: [rl_categories\(\)](#), [rl_pop_trends\(\)](#), [rl_stresses\(\)](#), [rl_use_and_trade\(\)](#)

Examples

```
## Not run:
# Get all threats
rl_threats()
# Get assessment summary for intentional hunting and trapping
rl_threats("5_1_1")

## End(Not run)
```

rl_use_and_trade

Use and trade assessment summary

Description

Return a list of the latest assessments which are subject to a specific use and trade.

Usage

```
rl_use_and_trade(
  code = NULL,
  key = NULL,
  parse = TRUE,
  all = TRUE,
  page = 1,
  quiet = FALSE,
  ...
)

rl_use_and_trade_(
  code = NULL,
  key = NULL,
  all = TRUE,
  page = 1,
  quiet = FALSE,
  ...
)
```

Arguments

code	(character) The code of the use and trade to look up. If not supplied, a list of all uses and trades will be returned.
key	(character) An IUCN API token. See rl_use_iucn() .
parse	(logical) Whether to parse the output to list (FALSE) or, where possible, data.frame (TRUE). Default: TRUE.
all	(logical) Whether to retrieve all results at once or not. If TRUE we do the paging internally for you and bind all of the results together. If FALSE, only a single page of results will be retrieved.

page	(integer/numeric) Page to get if all is FALSE. Default: 1. Each page returns up to 100 records. Paging is required because it's too much burden on a server to just "get all the data" in one request.
quiet	(logical) Whether to suppress progress for multi-page downloads or not. Default: FALSE (that is, give progress). Ignored if all = FALSE.
...	Includes the following filters that are supported by the API endpoint (refer to the API docs for more information): • year_published: (integer) Set this to return only assessments from a given year. • latest: (logical) Set this to TRUE to return only the latest assessment for each taxon. • scope_code: (integer) Set this to return only assessments from a particular scope (e.g., 1 for Global, 2 for Europe). This is similar to the region argument of the old Red List API and old versions of rredlist. Also includes the following arguments related to the wait time between request retries if a "Too Many Requests" error is received from the API (see HttpClient()\$retry() for more details): • pause_base, pause_cap, and pause_min: basis, maximum, and minimum for calculating wait time for retry • times: the maximum number of times to retry • onwait: a callback function if the request will be retried and a wait time is being applied Also supports any curl options passed to the GET request via HttpClient.

Value

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

References

API docs at <https://api.iucnredlist.org/>.

See Also

Assessments by threat details: [rl_categories\(\)](#), [rl_pop_trends\(\)](#), [rl_stresses\(\)](#), [rl_threats\(\)](#)

Examples

```
## Not run:
# Get all stresses
rl_use_and_trade()
# Get assessment summary for medicinal use and trade
rl_use_and_trade("3")

## End(Not run)
```

`rl_use_iucn`*Helper to get and save IUCN API key*

Description

Provides instruction on how to get and store a key for the IUCN API. Users of `rredlist` must do this before using any of the package's other functionality. Note that registering for a key requires an email address and information about your organization.

Usage

```
rl_use_iucn()
```

Details

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

Value

Invisibly returns the sign-up URL for the IUCN Red List API.

Examples

```
## Not run:
# Sign up for an API key
rl_use_iucn()

## End(Not run)
```

`rl_version`*Get the Red List version*

Description

Returns the current version number of the IUCN Red List of Threatened Species

Usage

```
rl_version(key = NULL, ...)
```

Arguments

<code>key</code>	(character) An IUCN API token. See rl_use_iucn() .
<code>...</code>	Curl options passed to HttpClient .

Value

Red List version as character string.

See Also

Red List information and statistics: `rl_api_version()`, `rl_citation()`, `rl_sp_count()`

Examples

```
## Not run:
rl_version()

## End(Not run)
```

scale_color_iucn	<i>IUCN Red List color scales</i>
------------------	-----------------------------------

Description

ggplot2 color scales using the colors for the IUCN Red List threat categories, as documented in [this chart](#).

Usage

```
scale_color_iucn(...)

scale_fill_iucn(...)

scale_discrete_iucn(aesthetics, ...)
```

Arguments

...	Arguments passed on to <code>ggplot2::discrete_scale</code> (except for <code>na.value</code> which is already set according to the IUCN standard of #C1B5A5)
aesthetics	Character string or vector of character strings listing the name(s) of the aesthetic(s) that this scale works with. This can be useful, for example, to apply color settings to the color and fill aesthetics at the same time, via <code>aesthetics = c("color", "fill")</code> .

Examples

```
library(ggplot2)
categories <- c("NE", "DD", "LC", "NT", "VU", "EN", "CR", "RE", "EW", "EX")
df <- data.frame(
  x = runif(1000, 0, 10), y = runif(1000, 0, 10),
  color = sample(categories, 1000, TRUE), shape = 21
)
ggplot(df) +
```



```
geom_point(aes(x = x, y = y, fill = color), shape = 21) +  
scale_fill_iucn(name = "IUCN Category") +  
theme_classic()
```

Index

- * **conservation**
 - rl_actions, 3
 - rl_research, 37
- * **geo**
 - rl_countries, 16
 - rl_faos, 22
 - rl_realms, 36
 - rl_scopes, 39
- * **groups**
 - rl_comp_groups, 14
 - rl_extinct, 18
 - rl_extinct_wild, 19
 - rl_green, 24
 - rl_growth_forms, 25
- * **habitat**
 - rl_habitats, 27
 - rl_systems, 49
- * **stats**
 - rl_api_version, 4
 - rl_citation, 11
 - rl_sp_count, 47
 - rl_version, 55
- * **taxa**
 - rl_class, 12
 - rl_family, 20
 - rl_kingdom, 28
 - rl_order, 30
 - rl_phylum, 32
 - rl_sis, 41
 - rl_sis_latest, 42
 - rl_species, 43
 - rl_species_latest, 45
- * **threats**
 - rl_categories, 9
 - rl_pop_trends, 34
 - rl_stresses, 48
 - rl_threats, 51
 - rl_use_and_trade, 53
- bibentry, 12
- curl options, 4, 5, 9, 11, 13, 15, 17, 18, 20, 21, 23, 24, 26, 28, 30, 31, 33, 35, 37, 38, 40–42, 44, 46, 49, 50, 52, 54
- ggplot2::discrete_scale, 56
- HttpClient, 4, 5, 9, 11–13, 15, 17, 18, 20, 21, 23, 24, 26, 28, 30, 31, 33, 35, 37, 38, 40–42, 44, 46, 47, 49, 50, 52, 54, 55
- HttpClient()\$retry(), 4, 5, 9, 11, 13, 15, 17, 18, 20, 21, 23, 24, 26, 28, 30, 31, 33, 35, 37, 38, 40–42, 44, 46, 49, 50, 52, 54
- rl_actions, 3, 39
- rl_actions_(rl_actions), 3
- rl_api_version, 4, 12, 47, 56
- rl_assessment, 5
- rl_assessment(), 8, 9, 42, 45
- rl_assessment_(rl_assessment), 5
- rl_assessment_extract, 6
- rl_assessment_list, 8
- rl_assessment_list(), 6
- rl_categories, 9, 35, 49, 52, 54
- rl_categories_(rl_categories), 9
- rl_citation, 5, 11, 47, 56
- rl_class, 12, 22, 30, 32, 33, 42, 43, 45, 46
- rl_class_(rl_class), 12
- rl_comp_groups, 14, 19, 20, 24, 26
- rl_comp_groups_(rl_comp_groups), 14
- rl_countries, 16, 23, 37, 40
- rl_countries_(rl_countries), 16
- rl_extinct, 15, 18, 20, 24, 26
- rl_extinct_(rl_extinct), 18
- rl_extinct_wild, 15, 19, 19, 24, 26
- rl_extinct_wild_(rl_extinct_wild), 19
- rl_family, 13, 20, 30, 32, 33, 42, 43, 45, 46
- rl_family_(rl_family), 20
- rl_faos, 17, 22, 37, 40
- rl_faos_(rl_faos), 22

rl_green, [15](#), [19](#), [20](#), [24](#), [26](#)
 rl_green_(rl_green), [24](#)
 rl_growth_forms, [15](#), [19](#), [20](#), [24](#), [25](#)
 rl_growth_forms_(rl_growth_forms), [25](#)
 rl_habitats, [27](#), [51](#)
 rl_habitats_(rl_habitats), [27](#)
 rl_key(rl_use_iucn), [55](#)
 rl_kingdom, [13](#), [22](#), [28](#), [32](#), [33](#), [42](#), [43](#), [45](#), [46](#)
 rl_kingdom_(rl_kingdom), [28](#)
 rl_order, [13](#), [22](#), [30](#), [30](#), [33](#), [42](#), [43](#), [45](#), [46](#)
 rl_order_(rl_order), [30](#)
 rl_phylum, [13](#), [22](#), [30](#), [32](#), [32](#), [42](#), [43](#), [45](#), [46](#)
 rl_phylum_(rl_phylum), [32](#)
 rl_pop_trends, [11](#), [34](#), [49](#), [52](#), [54](#)
 rl_pop_trends_(rl_pop_trends), [34](#)
 rl_realms, [17](#), [23](#), [36](#), [40](#)
 rl_realms_(rl_realms), [36](#)
 rl_research, [4](#), [37](#)
 rl_research_(rl_research), [37](#)
 rl_scopes, [17](#), [23](#), [37](#), [39](#)
 rl_scopes(), [42](#), [46](#)
 rl_scopes_(rl_scopes), [39](#)
 rl_sis, [13](#), [22](#), [30](#), [32](#), [33](#), [41](#), [43](#), [45](#), [46](#)
 rl_sis(), [42](#)
 rl_sis_(rl_sis), [41](#)
 rl_sis_latest, [13](#), [22](#), [30](#), [32](#), [33](#), [42](#), [42](#), [45](#),
 [46](#)
 rl_sp_count, [5](#), [12](#), [47](#), [56](#)
 rl_sp_count_(rl_sp_count), [47](#)
 rl_species, [13](#), [22](#), [30](#), [32](#), [33](#), [42](#), [43](#), [43](#), [46](#)
 rl_species(), [45](#)
 rl_species_(rl_species), [43](#)
 rl_species_latest, [13](#), [22](#), [30](#), [32](#), [33](#), [42](#),
 [43](#), [45](#), [45](#)
 rl_stresses, [11](#), [35](#), [48](#), [52](#), [54](#)
 rl_stresses_(rl_stresses), [48](#)
 rl_systems, [28](#), [49](#)
 rl_systems_(rl_systems), [49](#)
 rl_threats, [11](#), [35](#), [49](#), [51](#), [54](#)
 rl_threats_(rl_threats), [51](#)
 rl_use_and_trade, [11](#), [35](#), [49](#), [52](#), [53](#)
 rl_use_and_trade_(rl_use_and_trade), [53](#)
 rl_use_iucn, [55](#)
 rl_use_iucn(), [3](#), [5](#), [9](#), [10](#), [12](#), [14](#), [16](#), [18](#), [19](#),
 [21](#), [22](#), [24](#), [25](#), [27](#), [29](#), [31](#), [32](#), [34](#), [36](#),
 [38](#), [40–42](#), [44](#), [46–48](#), [50](#), [52](#), [53](#), [55](#)
 rl_version, [5](#), [12](#), [47](#), [55](#)
 scale_color_iucn, [56](#)
 scale_colour_iucn(scale_color_iucn), [56](#)
 scale_discrete_iucn(scale_color_iucn),
 [56](#)
 scale_fill_iucn(scale_color_iucn), [56](#)
 scope, [4](#), [10](#), [13](#), [15](#), [17](#), [21](#), [23](#), [26](#), [28](#), [29](#), [31](#),
 [33](#), [35](#), [36](#), [38](#), [40](#), [49](#), [50](#), [52](#), [54](#)