Package 'taxa'

March 11, 2022

Type Package

Title Classes for Storing and Manipulating Taxonomic Data

Description Provides classes for storing and manipulating taxonomic data.
Most of the classes can be treated like base R vectors (e.g. can be used
in tables as columns and can be named). Vectorized classes can store taxon names
and authorities, taxon IDs from databases, taxon ranks, and other types of
information. More complex classes are provided to store taxonomic trees and
user-defined data associated with them.

Version 0.4.1

Depends R (>= 3.0.2),
vctrs

LazyLoad yes

Encoding UTF-8

License MIT + file LICENSE

URL `https://docs.ropensci.org/taxa/`, `https://github.com/ropensci/taxa`

BugReports `https://github.com/ropensci/taxa/issues`

Roxygen list(markdown = TRUE)

Imports dplyr,
magrittr,
tibble,
rlang,
stringr,
crayon,
utils,
pillar,
methods,
viridisLite,
cli

Suggests roxygen2 (>= 6.0.1),
testthat,
rmarkdown (>= 0.9.6)

RoxygenNote 7.1.2

X-schema.org-applicationCategory Taxonomy

X-schema.org-keywords taxonomy, biology, hierarchy

R topics documented:

as_data_frame — *Convert a taxa object to a* `data.frame`

Description

Convert the information in a taxa object to a `data.frame` using base R vectors as columns. Use as_tibble to convert to tibbles.

Usage

```
as_data_frame(
  x,
  row.names = NULL,
  optional = FALSE,
  ...,
  stringsAsFactors = FALSE
)
```

Arguments

`x` An object defined by taxa, such as taxon or taxon_id

`row.names` `NULL` or a character vector giving the row names for the data frame. Missing values are not allowed.

`optional` logical. If `TRUE`, setting row names and converting column names (to syntactic names: see `make.names`) is optional. Note that all of R's **base** package `as.data.frame()` methods use `optional` only for column names treatment, basically with the meaning of `data.frame(*,check.names = !optional)`. See also the `make.names` argument of the `matrix` method.

`...` additional arguments to be passed to or from methods.

`stringsAsFactors`
logical: should the character vector be converted to a factor?

Examples

```
x <- taxon(name = c('Homo sapiens', 'Bacillus', 'Ascomycota', 'Ericaceae'),
           rank = c('species', 'genus', 'phylum', 'family'),
           id = taxon_id(c('9606', '1386', '4890', '4345'), db = 'ncbi'),
           auth = c('Linnaeus, 1758', 'Cohn 1872', NA, 'Juss., 1789'))
as_data_frame(x)
```

as_taxon *Convert to a taxon vector*

Description

Convert other objects to taxon vectors. Compatible base R vectors can also be converted using the taxon constructor.

Usage

```
as_taxon(x, ...)
```

Arguments

x	An object to be converted to a taxon vector
...	Additional parameters.

Examples

```
# Convert a taxonomy object to a taxon vector
x <- taxonomy(taxon(name = c('Carnivora', 'Felidae', 'Panthera', 'Panthera leo',
                             'Panthera tigris', 'Ursidae', 'Ursus', 'Ursus arctos'),
                    rank = c('order', 'family', 'genus', 'species',
                             'species', 'family', 'genus', 'species'),
                    id = taxon_id(c('33554', '9681', '9688', '9689',
                                    '9694', '9632', '9639', '9644'),
                                  db = 'ncbi'),
             auth = c('Bowdich, 1821', 'Fischer de Waldheim, 1817', 'Oken, 1816', 'L., 1758',
                       'L., 1758', 'Fischer de Waldheim, 1817', 'L., 1758', 'L., 1758')),
              supertaxa = c(NA, 1, 2, 3, 3, 1, 6, 7))
names(x) <- letters[1:8]
as_taxon(x)

# Convert base R vectors
as_taxon(c('Carnivora', 'Felidae', 'Panthera', 'Panthera leo'))
as_taxon(factor(c('Carnivora', 'Felidae', 'Panthera', 'Panthera leo')))
```

classification *Taxon class*

Description

Experimental Used to store classifications in reference to a taxonomic tree.

Usage

```
classification(x = NULL, taxonomy = NULL, .names = NULL)
```

Arguments

`x` One of:

- A list where each item represents a series of nested taxa. The contents of the list can be in any form that can be converted to a taxon vector.
- The indexes/names of each instance of a taxon in a taxonomy object specified by the `taxonomy` option. Can be any length, but must consist of valid indexes for taxa in the `taxonomy` object.

`taxonomy` A taxonomy object. Only needed if taxon indexes are supplied as the first argument.

`.names` The names of the vector.

Value

An S3 object of class `taxa_classification`

See Also

Other classes: `[.taxa_classification`(), `taxon_authority`(), `taxon_db`(), `taxon_id`(), `taxon_rank`(), `taxon`()

Examples

```
# Create classification vector with a list
x <- classification(list(
  c('Carnivora', 'Felidae', 'Panthera', 'Panthera leo'),
  c('Carnivora', 'Felidae', 'Panthera', 'Panthera tigris'),
  c('Carnivora', 'Ursidae', 'Ursus', 'Ursus arctos'),
  c('Carnivora', 'Ursidae', 'Ursus', 'Ursus arctos'),
  c('Carnivora', 'Felidae', 'Panthera', 'Panthera tigris')
))


# Create classification vector with indexes and a taxonomy
x <- classification(c(3, 4, 4, 5, 5, 6, 8, 8, 2, 5, 6, 2),
                    taxonomy(c('Carnivora', 'Felidae', 'Panthera', 'Panthera leo',
                               'Panthera tigris', 'Ursidae', 'Ursus', 'Ursus arctos'),
                             supertaxa = c(NA, 1, 2, 3, 3, 1, 6, 7)))

x <- classification(c(3, 4, 4, 5, 5, 6, 8, 8, 2, 5, 6, 2),
               taxonomy(taxon(name = c('Carnivora', 'Felidae', 'Panthera', 'Panthera leo',
                                  'Panthera tigris', 'Ursidae', 'Ursus', 'Ursus arctos'),
                                   rank = c('order', 'family', 'genus', 'species',
                                            'species', 'family', 'genus', 'species'),
                                   id = taxon_id(c('33554', '9681', '9688', '9689',
                                                   '9694', '9632', '9639', '9644'),
                                                 db = 'ncbi'),
                                   auth = c('Bowdich, 1821', 'Fischer, 1817',
                                            'Oken, 1816', 'L., 1758',
                                            'L., 1758', 'Fischer, 1817',
                                            'L., 1758', 'L., 1758')),
                              supertaxa = c(NA, 1, 2, 3, 3, 1, 6, 7)))
names(x) <- letters[1:12]


# Get parts of the classification vector
tax_name(x)
tax_rank(x)
tax_id(x)
tax_db(x)
tax_auth(x)
tax_author(x)
tax_date(x)
tax_cite(x)

# Manipulate classification vectors
x[1:3]
x[tax_rank(x) > 'family']
# c(x, x)
# x['b'] <- NA
is.na(x)
# as.data.frame(x)
# tibble::as_tibble(x)

# Use as columns in tables
tibble::tibble(x = x, y = 1:12)
data.frame(x = x, y = 1:12)
```

db_ref *Valid taxonomy databases*

Description

This defines the valid taxonomic databases that can be used in taxon_db objects and objects that use taxon_db objects, such as taxon_id and taxon. `db_ref$get` can be used to see information for the databases. Users can add their own custom databases to the list using `db_ref$set`. For each database the following information is included:

- The URL for the website associated with the database
- A short description
- The regular expression that defines valid taxon IDs
- The ranks used in the database if specified

Usage

```
db_ref
```

Format

An object of class `list` of length 3.

Attribution

This code is based on the code handling options in the knitr package.

Examples

```
# List all database definitions
db_ref$get()

# Get a specific database definition
db_ref$get('ncbi')

# Add or overwrite a database definition
db_ref$set(
  name = "my_new_database",
  url = "http://www.my_tax_database.com",
  desc = "I just made this up",
  id_regex = ".*"
)

# Reset definitions to default values
db_ref$reset()
```

---

internodes *Get internodes*

Description

Get internodes indexes for each taxon or another per-taxon value. An internode is a taxon with exactly one supertaxon and one subtaxon. These taxa can be removed without losing information on the relationships of the remaining taxa.

Usage

```
internodes(x)
```

Arguments

x	The object to get internodes for, such as a taxonomy object.

See Also

Other taxonomy functions: `leaves()`, `roots()`, `stems()`, `subtaxa()`, `supertaxa()`

Other internode functions: `is_internode()`

Examples

```
x <- taxonomy(c('Carnivora', 'Felidae', 'Panthera', 'Panthera leo',
                'Panthera tigris', 'Ursidae', 'Ursus', 'Ursus arctos'),
              supertaxa = c(NA, 1, 2, 3, 3, 1, 6, 7))
internodes(x)
```

is_classification *Check if is a classification*

Description

Check if an object is the classification class

Usage

```
is_classification(x)
```

Arguments

x	An object to test

is_internode *Check if taxa are internodes*

Description

Check if each taxon is an internode. An internode is a taxon with exactly one supertaxon and one subtaxon. These taxa can be removed without losing information on the relationships of the remaining taxa.

Usage

```
is_internode(x)
```

Arguments

x	The object to get internodes for, such as a taxonomy object.

See Also

Other internode functions: `internodes()`

Examples

```
x <- taxonomy(c('Carnivora', 'Felidae', 'Panthera', 'Panthera leo',
                'Panthera tigris', 'Ursidae', 'Ursus', 'Ursus arctos'),
              supertaxa = c(NA, 1, 2, 3, 3, 1, 6, 7))
is_internode(x)
```

is_leaf *Check if taxa are leaves*

Description

Check if each taxon is a leaf. A leaf is a taxon with no subtaxa. subtaxa.

Usage

```
is_leaf(x)
```

Arguments

x	The object to get leaves for, such as a taxonomy object

See Also

Other leaf functions: `leaves()`, `n_leaves()`

Examples

```
x <- taxonomy(c('Carnivora', 'Felidae', 'Panthera', 'Panthera leo',
                'Panthera tigris', 'Ursidae', 'Ursus', 'Ursus arctos'),
              supertaxa = c(NA, 1, 2, 3, 3, 1, 6, 7))
is_leaf(x)
```

is_root *Test if taxa are roots*

Description

Check if each taxon is a root. A root is a taxon with no supertaxon.

Usage

```
is_root(x, subset = NULL)
```

Arguments

x An object containing taxonomic relationships, such as taxonomy objects.

subset The subset of the tree to search for roots to that subset. Can be indexes or names.

See Also

Other root functions: `roots()`

Examples

```
x <- taxonomy(c('Carnivora', 'Felidae', 'Panthera', 'Panthera leo',
                'Panthera tigris', 'Ursidae', 'Ursus', 'Ursus arctos'),
              supertaxa = c(NA, 1, 2, 3, 3, 1, 6, 7))
is_root(x)
is_root(x, subset = 2:8)
```

`is_stem` *Check if taxa are stems*

Description

Check if each taxon is a stem. A stem is any taxa from a root to the first taxon with multiple subtaxa.

Usage

```
is_stem(x)
```

Arguments

x An object with taxonomic relationships, like taxonomy objects.

See Also

Other stem functions: `stems()`

Examples

```
x <- taxonomy(c('Carnivora', 'Felidae', 'Panthera', 'Panthera leo',
                'Panthera tigris'),
              supertaxa = c(NA, 1, 2, 3, 3))
is_stem(x)
```

is_taxon — *Check if something is a taxon object*

Description

Check if an object is of the taxon class

Usage

```
is_taxon(x)
```

Arguments

x	An object to test

Examples

```
x <- taxon(c('A', 'B', 'C'))
is_taxon(x)
is_taxon(1:2)
```

is_taxonomy — *Check if something is a taxonomy*

Description

Check if an object is of the taxonomy class

Usage

```
is_taxonomy(x)
```

Arguments

x	An object to test

Examples

```
x <- taxonomy(c('Carnivora', 'Felidae', 'Panthera', 'Panthera leo',
                'Panthera tigris', 'Ursidae', 'Ursus', 'Ursus arctos'),
              supertaxa = c(NA, 1, 2, 3, 3, 1, 6, 7))
is_taxonomy(x)
is_taxonomy(1:2)
```

is_taxon_authority *Check if is a taxon_authority*

Description

Check if an object is of the taxon_authority class

Usage

```
is_taxon_authority(x)
```

Arguments

x	An object to test

Examples

```
x <- taxon_authority(c('Cham. & Schldl.', 'L.'),
                     date = c('1827', '1753'))
is_taxon_authority(x)
is_taxon_authority(1:3)
```

is_taxon_db *Check if something is a taxon_db*

Description

Check if an object is of the taxon_db class

Usage

```
is_taxon_db(x)
```

Arguments

x	An object to test

Examples

```
x <- taxon_db(c('ncbi', 'ncbi', 'itis'))
is_taxon_db(x)
is_taxon_db(1:3)
```

is_taxon_id *Check if something is a taxon_id object*

Description

Check if an object is of the taxon_id class

Usage

```
is_taxon_id(x)
```

Arguments

x An object to test

Examples

```
x <- taxon_id(c('9606', '1386', '4890', '4345'), db = 'ncbi')
is_taxon_id(x)
is_taxon_id(1:3)
```

is_taxon_rank *Check if something is a taxon_rank*

Description

Check if an object is of the taxon_rank class

Usage

```
is_taxon_rank(x)
```

Arguments

x An object to test

Examples

```
x <- taxon_rank(c('species', 'species', 'phylum', 'family'))
is_taxon_rank(x)
is_taxon_rank(1:3)
```

`leaves` *Get leaves*

Description

Get leaves indexes for each taxon or another per-taxon value. Leaves are taxa with no subtaxa.

Usage

```
leaves(x, value = NULL, ...)
```

Arguments

`x`	The object to get leaves for, such as a taxonomy object
`value`	Something to return instead of indexes. Must be the same length as the number of taxa.
`...`	Additional arguments.

See Also

Other taxonomy functions: `internodes()`, `roots()`, `stems()`, `subtaxa()`, `supertaxa()`

Other leaf functions: `is_leaf()`, `n_leaves()`

Examples

```
x <- taxonomy(c('Carnivora', 'Felidae', 'Panthera', 'Panthera leo',
                'Panthera tigris', 'Ursidae', 'Ursus', 'Ursus arctos'),
              supertaxa = c(NA, 1, 2, 3, 3, 1, 6, 7))
leaves(x)
leaves(x, value = tax_name(x))
```

`n_leaves` *Number of leaves per taxon*

Description

Get the number of leaves per taxon. A leaf is a taxon with no subtaxa.

Usage

```
n_leaves(x)
```

Arguments

`x`	The object to get leaves for, such as a taxonomy object

See Also

Other leaf functions: `is_leaf()`, `leaves()`

Examples

```
x <- taxonomy(c('Carnivora', 'Felidae', 'Panthera', 'Panthera leo',
                'Panthera tigris', 'Ursidae', 'Ursus', 'Ursus arctos'),
              supertaxa = c(NA, 1, 2, 3, 3, 1, 6, 7))
n_leaves(x)
```

`n_subtaxa` *Number of subtaxa per taxon*

Description

Get the number of subtaxa per taxon.

Usage

```
n_subtaxa(x, subset = NULL, max_depth = NULL, include = FALSE)
```

Arguments

`x` The object to get subtaxa for, such as a taxonomy object.

`subset` The subset of the tree to search. Can be indexes or names.

`max_depth` The number of ranks to traverse. For example, `max_depth = 1` returns only immediate subtaxa. By default (NULL) information for all subtaxa is returned (i.e. subtaxa of subtaxa, etc).

`include` If `TRUE`, include information for each taxon in the output.

See Also

Other subtaxa functions: `subtaxa()`

Examples

```
# Generate example data
x <- taxonomy(c('Carnivora', 'Felidae', 'Panthera', 'Panthera leo',
                'Panthera tigris', 'Ursidae', 'Ursus', 'Ursus arctos'),
              supertaxa = c(NA, 1, 2, 3, 3, 1, 6, 7))

# Find number of subtaxa (including subtaxa of subtaxa, etc)
n_subtaxa(x)

# Find the number of subtaxa one rank below each taxon
n_subtaxa(x, max_depth = 1)
```

```
# Only return data for some taxa (faster than subsetting the whole result)
n_subtaxa(x, subset = 1:3)
```

n_supertaxa *Number of supertaxa per taxon*

Description

Get the number of supertaxa each taxon is contained in.

Usage

```
n_supertaxa(x, subset = NULL, max_depth = NULL, include = FALSE)
```

Arguments

`x` The object to get supertaxa for, such as a taxonomy object.

`subset` The subset of the tree to search for roots to that subset. Can be indexes or names.

`max_depth` The number of levels to traverse. For example, `max_depth = 1` returns only immediate supertaxa. By default (NULL) information for all supertaxa is returned.

`include` If `TRUE`, include information for each taxon in the output.

See Also

Other supertaxa functions: `supertaxa()`

Examples

```
# Generate example data
x <- taxonomy(c('Carnivora', 'Felidae', 'Panthera', 'Panthera leo',
                'Panthera tigris', 'Ursidae', 'Ursus', 'Ursus arctos'),
              supertaxa = c(NA, 1, 2, 3, 3, 1, 6, 7))

# Find number of supertaxa each taxon is contained in
n_supertaxa(x)

# Only return data for some taxa (faster than subsetting the whole result)
n_supertaxa(x, subset = 1:3)
```

roots *Get root taxa*

Description

Get the indexes of root taxa in a taxonomy.

Usage

```
roots(x, subset = NULL)
```

Arguments

`x`	An object containing taxonomic relationships, such as taxonomy objects.
`subset`	The subset of the tree to search for roots to that subset. Can be indexes or names.

See Also

Other taxonomy functions: `internodes()`, `leaves()`, `stems()`, `subtaxa()`, `supertaxa()`

Other root functions: `is_root()`

Examples

```
x <- taxonomy(c('Carnivora', 'Felidae', 'Panthera', 'Panthera leo',
                'Panthera tigris', 'Ursidae', 'Ursus', 'Ursus arctos'),
              supertaxa = c(NA, 1, 2, 3, 3, 1, 6, 7))
roots(x)
roots(x, subset = 2:8)
```

stems *Get stems*

Description

Get stem indexes for each taxon or another per-taxon value.

Usage

```
stems(x, value = NULL, ...)
```

Arguments

`x`	An object with taxonomic relationships, like taxonomy objects.
`value`	Something to return instead of indexes. Must be the same length as the number of taxa.
`...`	Additional arguments.

See Also

Other taxonomy functions: internodes(), leaves(), roots(), subtaxa(), supertaxa()

Other stem functions: is_stem()

Examples

```
x <- taxonomy(c('Carnivora', 'Felidae', 'Panthera', 'Panthera leo',
                'Panthera tigris'),
              supertaxa = c(NA, 1, 2, 3, 3))
x <- c(x, x)
stems(x)
stems(x, value = tax_name(x))
```

---

subtaxa *Get subtaxa*

Description

Get subtaxa indexes for each taxon or another per-taxon value. Subtaxa are taxa contained within a taxon.

Usage

```
subtaxa(x, subset = NULL, max_depth = NULL, include = FALSE, value = NULL, ...)
```

Arguments

x	The object to get subtaxa for, such as a taxonomy object.
subset	The subset of the tree to search. Can be indexes or names.
max_depth	The number of ranks to traverse. For example, max_depth = 1 returns only immediate subtaxa. By default (NULL) information for all subtaxa is returned (i.e. subtaxa of subtaxa, etc).
include	If TRUE, include information for each taxon in the output.
value	Something to return instead of indexes. Must be the same length as the number of taxa.
...	Additional arguments.

See Also

Other taxonomy functions: internodes(), leaves(), roots(), stems(), supertaxa()

Other subtaxa functions: n_subtaxa()

Examples

```
# Generate example data
x <- taxonomy(c('Carnivora', 'Felidae', 'Panthera', 'Panthera leo',
                'Panthera tigris', 'Ursidae', 'Ursus', 'Ursus arctos'),
              supertaxa = c(NA, 1, 2, 3, 3, 1, 6, 7))

# The indexes of all subtaxa (with subtaxa of subtaxa, etc) for each taxon
subtaxa(x)

# The indexes of immediate subtaxa (without subtaxa of subtaxa, etc) for each taxon
subtaxa(x, max_depth = 1)

# Return something other than index
subtaxa(x, value = tax_name(x))

# Include each taxon with its subtaxa
subtaxa(x, value = tax_name(x), include = TRUE)

# Only return data for some taxa (faster than subsetting the whole result)
subtaxa(x, subset = 3)
```

---

supertaxa *Get supertaxa*

Description

Get supertaxa indexes for each taxon or another per-taxon value. Supertaxa are taxa a taxon is contained in.

Usage

```
supertaxa(
  x,
  subset = NULL,
  max_depth = NULL,
  include = FALSE,
  value = NULL,
  use_na = FALSE,
  ...
)
```

Arguments

x	The object to get supertaxa for, such as a taxonomy object.
subset	The subset of the tree to search for roots to that subset. Can be indexes or names.
max_depth	The number of levels to traverse. For example, max_depth = 1 returns only immediate supertaxa. By default (NULL) information for all supertaxa is returned.

`include` If `TRUE`, include information for each taxon in the output.

`value` Something to return instead of indexes. Must be the same length as the number of taxa.

`use_na` Add a NA to represent the root of the taxonomy (i.e. no supertaxon)

`...` Additional arguments.

See Also

Other taxonomy functions: `internodes()`, `leaves()`, `roots()`, `stems()`, `subtaxa()`

Other supertaxa functions: `n_supertaxa()`

Examples

```
# Generate example data
x <- taxonomy(c('Carnivora', 'Felidae', 'Panthera', 'Panthera leo',
                'Panthera tigris', 'Ursidae', 'Ursus', 'Ursus arctos'),
              supertaxa = c(NA, 1, 2, 3, 3, 1, 6, 7))

# The indexes of all supertaxa (with supertaxa of supertaxa, etc) for each taxon
supertaxa(x)

# Return something other than index
supertaxa(x, value = tax_name(x))

# Include each taxon with its supertaxa
supertaxa(x, value = tax_name(x), include = TRUE)

# Only return data for some taxa (faster than subsetting the whole result)
supertaxa(x, subset = 3)
```

`taxa_taxon-class` *Taxon class*

Description

Taxon class. See taxon for more information

`taxa_taxonomy-class` *Taxonomy class*

Description

Taxonomy class. See taxonomy for more information

`taxa_taxon_authority-class` *Taxon authority class*

Description

Taxon authority class. See taxon_authority for more information

`taxa_taxon_db-class` *Taxon database class*

Description

Taxon database class. See taxon_db for more information

`taxa_taxon_id-class` *Taxon ID class*

Description

Taxon ID class. See taxon_id for more information

`taxa_taxon_rank-class` *Taxon rank class*

Description

Taxon rank class. See taxon_rank for more information taxon *Taxon class*

Description

Maturing Used to store information about taxa, such as names, ranks, and IDs.

Usage

```
taxon(name = character(0), rank = NA, id = NA, auth = NA, .names = NA, ...)
```

Arguments

`name`	The names of taxa. Inputs with be coerced into a character vector if anything else is given.
`rank`	The ranks of taxa. Inputs with be coerced into a taxon_rank vector if anything else is given.
`id`	The ids of taxa. These should be unique identifier and are usually associated with a database. Inputs with be coerced into a taxon_id vector if anything else is given.
`auth`	The authority of the taxon. Inputs with be coerced into a taxon_authority vector if anything else is given.
`.names`	The names of the vector.
`...`	Additional arguments.

Value

An S3 object of class `taxa_taxon`

See Also

Other classes: `[.taxa_classification()`, `classification()`, `taxon_authority()`, `taxon_db()`, `taxon_id()`, `taxon_rank()`

Examples

```
# Create taxon name vector
x <- taxon(c('A', 'B', 'C'))
x <- taxon(name = c('Homo sapiens', 'Bacillus', 'Ascomycota', 'Ericaceae'),
           rank = c('species', 'genus', 'phylum', 'family'),
           id = taxon_id(c('9606', '1386', '4890', '4345'), db = 'ncbi'),
           auth = c('Linnaeus, 1758', 'Cohn 1872', NA, 'Juss., 1789'))
names(x) <- c('a', 'b', 'c', 'd')

# Get parts of the taxon name vector
tax_name(x)
tax_rank(x)
```

```
tax_id(x)
tax_db(x)
tax_auth(x)
tax_author(x)
tax_date(x)
tax_cite(x)

# Set parts of the taxon name vector
tax_name(x) <- tolower(tax_name(x))
tax_rank(x)[1] <- NA
tax_name(x)['b'] <- 'Billy'
tax_id(x) <- '9999'
tax_db(x) <- 'itis'
tax_auth(x) <- NA
tax_author(x)[2:3] <- c('Joe', 'Billy')
tax_date(x) <- c('1999', '2013', '1796', '1899')
tax_cite(x)[1] <- 'Linnaeus, C. (1771). Mantissa plantarum altera generum.'

# Manipulate taxon name vectors
x[1:3]
x[tax_rank(x) > 'family']
x['b'] <- NA
x[c('c', 'd')] <- 'unknown'
is.na(x)

# Use as columns in tables
tibble::tibble(x = x, y = 1:4)
data.frame(x = x, y = 1:4)

# Converting to tables
tibble::as_tibble(x)
as_data_frame(x)
```

taxon_authority *Taxon authority class*

Description

Maturing Used to store information on taxon authorities, such as author names, date, and citation.

Usage

```
taxon_authority(
  author = character(),
  date = NA,
  citation = NA,
  .names = NA,
  extract_date = TRUE
)
```

Arguments

`author` Zero or more author names.

`date` Zero or more dates.

`citation` Zero or more literature citations.

`.names` The names of the vector.

`extract_date` If `TRUE` (the default), then if a date is detected in the `author` input and no `date` input is given, then the date is separated from the author input.

Value

An S3 object of class `taxa_taxon_authority`

See Also

Other classes: `[.taxa_classification()`, `classification()`, `taxon_db()`, `taxon_id()`, `taxon_rank()`, `taxon()`

Examples

```
# Making new objects
x <- taxon_authority(c('A', 'B', 'C'))
x <- taxon_authority(c('Cham. & Schldl.', 'L.'),
                     date = c('1827', '1753'))

# Manipulating objects
as.character(x)
x[2]
x[2] <- 'ABC'
names(x) <- c('a', 'b')
x['b'] <- 'David Bowie'
tax_author(x)[1] <- tolower(tax_author(x)[1])
tax_author(x)
tax_date(x) <- c('2000', '1234')
tax_date(x)
tax_cite(x)[2] <- c('Linnaeus, C. (1771). Mantissa plantarum altera generum.')
tax_cite(x)

# Using as columns in tables
tibble::tibble(x = x, y = 1:2)
data.frame(x = x, y = 1:2)

# Converting to tables
tibble::as_tibble(x)
as_data_frame(x)
```

taxon_db *Taxon database class*

Description

Maturing Used to store the names of taxon databases defined in db_ref. Primarily used in other classes like taxon_id to define databases for each item.

Usage

```
taxon_db(db = character(), .names = NULL, ...)
```

Arguments

db	Zero or more taxonomic database names. Should be a name contained in db_ref. Inputs will be transformed to a character vector if possible.
.names	The names of the vector.
...	Additional arguments.

Value

An S3 object of class `taxa_taxon_db`

See Also

Other classes: `[.taxa_classification()`, `classification()`, `taxon_authority()`, `taxon_id()`, `taxon_rank()`, `taxon()`

Examples

```
# Making new objects
x <- taxon_db(c('ncbi', 'ncbi', 'itis'))
x

# Manipulating objects
as.character(x)
x[2:3]
x[2:3] <- 'nbn'
names(x) <- c('a', 'b', 'c')
x['b']
x['b'] <- 'nbn'
x[x == 'itis'] <- 'gbif'

# Using as columns in tables
tibble::tibble(x = x, y = 1:3)
data.frame(x = x, y = 1:3)

# Converting to tables
tibble::as_tibble(x)
```

```
as_data_frame(x)

# Trying to use an invalid database generates an error
# x <- taxon_db(c('ncbi', 'ncbi', 'my_custom_db'))
# x[x == 'itis'] <- 'my_custom_db'

# Listing known databases and their properties
db_ref$get()

# Adding and using a new database
db_ref$set(name = 'my_custom_db', desc = 'I just made this up')
db_ref$get()
x <- taxon_db(c('ncbi', 'ncbi', 'my_custom_db'))
```

`taxon_id` *Taxon ID class*

Description

Maturing Used to store the ID corresponding to taxa, either arbitrary or from a particular taxonomy database. This is typically used to store taxon IDs in taxon objects.

Usage

```
taxon_id(id = character(), db = NA, .names = NULL)
```

Arguments

`id`	Zero or more taxonomic ids. Inputs will be transformed to a character vector if possible.
`db`	The name(s) of the database(s) associated with the IDs. If not `NA` (the default), the input must consist of names of databases in db_ref$get().
`.names`	The names that will be applied to the vector.

Value

An `S3` object of class `taxa_taxon_id`

See Also

Other classes: `[.taxa_classification()`, `classification()`, `taxon_authority()`, `taxon_db()`, `taxon_rank()`, `taxon()`

Examples

```
# Making new objects
x <- taxon_id(c('A', 'B', 'C'))
x <- taxon_id(c('9606', '1386', '4890', '4345'), db = 'ncbi')
x <- taxon_id(c('9606', '1386', '4890', '4345'),
              db = c('ncbi', 'ncbi', 'itis', 'itis'))
names(x) <- c('a', 'b', 'c', 'd')

# Manipulating objects
as.character(x)
x[2:3]
x[2:3] <- 'ABC'
x[c('a', 'c')] <- '123'
x[['b']] <- taxon_id('123423', db = 'ncbi')
tax_db(x)
tax_db(x) <- 'nbn'
c(x, x)

# Using as columns in tables
tibble::tibble(x = x, y = 1:4)
data.frame(x = x, y = 1:4)

# Convert to tables
tibble::as_tibble(x)
as_data_frame(x)

# Trying to use an invalid ID with a specified database causes an error
#taxon_id('NOLETTERS', db = 'ncbi')
```

taxon_rank *Taxon rank class*

Description

Maturing Used to store taxon ranks, possibly associated with a taxonomy database. This is typically used to store taxon ranks in taxon objects.

Usage

```
taxon_rank(
  rank = character(),
  .names = NULL,
  levels = NULL,
  guess_order = TRUE
)
```

Arguments

`rank` Zero or more taxonomic rank names. Inputs will be transformed to a character vector.

`.names` The names of the vector

`levels` A named numeric vector indicating the names and orders of possible taxonomic ranks. Higher numbers indicate for fine-scale groupings. Ranks of unknown order can be indicated with `NA` instead of a number.

`guess_order` If `TRUE` and no rank order is given using numbers, try to guess order based on rank names.

Value

An `S3` object of class `taxa_taxon_rank`

See Also

Other classes: `[.taxa_classification()`, `classification()`, `taxon_authority()`, `taxon_db()`, `taxon_id()`, `taxon()`

Examples

```
# Making new objects
x <- taxon_rank(c('species', 'species', 'phylum', 'family'))

# Specifiying level order
taxon_rank(c('A', 'B', 'C', 'D', 'A', 'D', 'D'),
           levels = c('D', 'C', 'B', 'A'))
taxon_rank(c('A', 'B', 'C', 'D', 'A', 'D', 'D'),
           levels = c(D = NA, A = 10, B = 20, C = 30))
names(x) <- c('a', 'b', 'c', 'd')

# Manipulating objects
as.character(x)
as.factor(x)
as.ordered(x)
x[2:3]
x[x > 'family'] <- taxon_rank('unknown')
x[1] <- taxon_rank('order')
x['b']
x['b'] <- 'order'

# Using as columns in tables
tibble::tibble(x = x, y = 1:4)
data.frame(x = x, y = 1:4)

# Converting to tables
tibble::as_tibble(x)
as_data_frame(x)

# Trying to add an unknown level as a character causes an error
```

```
#x[2] <- 'superkingdom'

# But you can add a new level using taxon_rank objects
x[2] <- taxon_rank('superkingdom')
```

`tax_auth.taxa_classification`

Set and get taxon authorities

Description

Set and get the taxon authorities in objects that have them, such as taxon objects. Note that this sets all the authority information, such as author name, date, and citations. To set or get just one of part of the authorities, use tax_author, tax_date, or tax_cite instead.

Usage

```
## S3 method for class 'taxa_classification'
tax_auth(x)

## S3 replacement method for class 'taxa_classification'
tax_auth(x) <- value

tax_auth(x)

tax_auth(x) <- value

## S3 method for class 'taxa_taxon'
tax_auth(x)

## S3 replacement method for class 'taxa_taxon'
tax_auth(x) <- value

## S3 method for class 'taxa_taxonomy'
tax_auth(x)

## S3 replacement method for class 'taxa_taxonomy'
tax_auth(x) <- value
```

Arguments

`x`	An object with taxon authorities.
`value`	The taxon IDs to set. Inputs will be coerced into a taxon_id vector.

Examples

```
x <- taxon(name = c('Homo sapiens', 'Bacillus', 'Ascomycota', 'Ericaceae'),
           rank = c('species', 'genus', 'phylum', 'family'),
           id = taxon_id(c('9606', '1386', '4890', '4345'), db = 'ncbi'),
           auth = c('Linnaeus, 1758', 'Cohn 1872', NA, 'Juss., 1789'))

tax_auth(x)
tax_auth(x) <- tolower(tax_auth(x))
tax_auth(x)[1] <- 'Billy'
```

---

tax_author.taxa_classification

Set and get taxon authors

Description

Set and get taxon authors in objects that have them, such as taxon_authority objects.

Usage

```
## S3 method for class 'taxa_classification'
tax_author(x)

## S3 replacement method for class 'taxa_classification'
tax_author(x) <- value

tax_author(x)

tax_author(x) <- value

## S3 method for class 'taxa_taxon'
tax_author(x)

## S3 replacement method for class 'taxa_taxon'
tax_author(x) <- value

## S3 replacement method for class 'taxa_taxon_authority'
tax_author(x) <- value

## S3 method for class 'taxa_taxon_authority'
tax_author(x)

## S3 method for class 'taxa_taxonomy'
tax_author(x)

## S3 replacement method for class 'taxa_taxonomy'
tax_author(x) <- value
```

Arguments

`x`	An object with taxon authors.
`value`	The taxon authors to set. Inputs will be coerced into a character vector.

Examples

```
x <- taxon_authority(c('Cham. & Schldl.', 'L.'),
                     date = c('1827', '1753'))
tax_author(x)
tax_author(x)[1] <- "Billy"
tax_author(x) <- tolower(tax_author(x))
```

`tax_cite.taxa_classification`

Set and get taxon authority citations

Description

Set and get the taxon authority citations in objects that have them, such as taxon_authority objects.

Usage

```
## S3 method for class 'taxa_classification'
tax_cite(x)

## S3 replacement method for class 'taxa_classification'
tax_cite(x) <- value

tax_cite(x)

tax_cite(x) <- value

## S3 method for class 'taxa_taxon'
tax_cite(x)

## S3 replacement method for class 'taxa_taxon'
tax_cite(x) <- value

## S3 replacement method for class 'taxa_taxon_authority'
tax_cite(x) <- value

## S3 method for class 'taxa_taxon_authority'
tax_cite(x)

## S3 method for class 'taxa_taxonomy'
tax_cite(x)
```

```
## S3 replacement method for class 'taxa_taxonomy'
tax_cite(x) <- value
```

Arguments

x	An object with taxon authority dates.
value	The taxon citations to set. Inputs will be coerced into a taxon_authority vector.

Examples

```
x <- taxon_authority(c('Cham. & Schldl.', 'L.'),
                     date = c('1827', '1753'),
                     citation = c(NA, 'Species Plantarum'))
tax_cite(x)
tax_cite(x)[1] <- "Cham. et al 1984"
```

---

tax_date.taxa_classification

Set and get taxon authority dates

Description

Set and get the taxon authority dates in objects that have them, such as taxon_authority objects.

Usage

```
## S3 method for class 'taxa_classification'
tax_date(x)

## S3 replacement method for class 'taxa_classification'
tax_date(x) <- value

tax_date(x)

tax_date(x) <- value

## S3 method for class 'taxa_taxon'
tax_date(x)

## S3 replacement method for class 'taxa_taxon'
tax_date(x) <- value

## S3 replacement method for class 'taxa_taxon_authority'
tax_date(x) <- value
```

```
## S3 method for class 'taxa_taxon_authority'
tax_date(x)

## S3 method for class 'taxa_taxonomy'
tax_date(x)

## S3 replacement method for class 'taxa_taxonomy'
tax_date(x) <- value
```

Arguments

x	An object with taxon authority dates.
value	The taxon authority dates to set. Inputs will be coerced into a character vector.

Examples

```
x <- taxon_authority(c('Cham. & Schldl.', 'L.'),
                     date = c('1827', '1753'))
tax_date(x)
tax_date(x)[1] <- "1984"
tax_date(x) <- c(NA, '1800')
```

---

tax_db.taxa_classification

Set and get taxon ID databases

Description

Set and get the taxon ID databases in objects that have them, such as taxon_id objects.

Usage

```
## S3 method for class 'taxa_classification'
tax_db(x)

## S3 replacement method for class 'taxa_classification'
tax_db(x) <- value

tax_db(x)

tax_db(x) <- value

## S3 method for class 'taxa_taxon'
tax_db(x)

## S3 replacement method for class 'taxa_taxon'
```

```
tax_db(x) <- value

## S3 method for class 'taxa_taxon_id'
tax_db(x)

## S3 replacement method for class 'taxa_taxon_id'
tax_db(x) <- value

## S3 method for class 'taxa_taxonomy'
tax_db(x)

## S3 replacement method for class 'taxa_taxonomy'
tax_db(x) <- value
```

Arguments

x	An object with taxon authority dates.
value	The taxon citations to set. Inputs will be coerced into a taxon_db vector.

Examples

```
x <- taxon_id(c('9606', '1386', '4890', '4345'), db = 'ncbi')
tax_db(x)
tax_db(x) <- 'nbn'
tax_db(x)[2] <- 'itis'
```

tax_id.taxa_classification

Set and get taxon IDs

Description

Set and get the taxon IDs in objects that have them, such as taxon objects.

Usage

```
## S3 method for class 'taxa_classification'
tax_id(x)

## S3 replacement method for class 'taxa_classification'
tax_id(x) <- value

tax_id(x)

tax_id(x) <- value
```

```
## S3 method for class 'taxa_taxon'
tax_id(x)

## S3 replacement method for class 'taxa_taxon'
tax_id(x) <- value

## S3 method for class 'taxa_taxonomy'
tax_id(x)

## S3 replacement method for class 'taxa_taxonomy'
tax_id(x) <- value
```

Arguments

`x`	An object with taxon IDs.
`value`	The taxon IDs to set. Inputs will be coerced into a taxon_id vector.

Examples

```
x <- taxon(name = c('Homo sapiens', 'Bacillus', 'Ascomycota', 'Ericaceae'),
           rank = c('species', 'genus', 'phylum', 'family'),
           id = taxon_id(c('9606', '1386', '4890', '4345'), db = 'ncbi'),
           auth = c('Linnaeus, 1758', 'Cohn 1872', NA, 'Juss., 1789'))

tax_id(x)
tax_id(x) <- paste0('00', tax_id(x))
tax_id(x)[1] <- '00000'
```

`tax_name.taxa_classification` *Set and get taxon names*

Description

Set and get the taxon names in objects that have them, such as taxon objects. Note that this is not the same as adding vector names with names.

Usage

```
## S3 method for class 'taxa_classification'
tax_name(x)

## S3 replacement method for class 'taxa_classification'
tax_name(x) <- value

tax_name(x)
tax_name(x) <- value

## S3 method for class 'taxa_taxon'
tax_name(x)

## S3 replacement method for class 'taxa_taxon'
tax_name(x) <- value

## S3 method for class 'taxa_taxonomy'
tax_name(x)

## S3 replacement method for class 'taxa_taxonomy'
tax_name(x) <- value
```

Arguments

x	An object with taxon names.
value	The taxon names to set. Inputs will be coerced into a character vector.

Examples

```
x <- taxon(name = c('Homo sapiens', 'Bacillus', 'Ascomycota', 'Ericaceae'),
           rank = c('species', 'genus', 'phylum', 'family'),
           id = taxon_id(c('9606', '1386', '4890', '4345'), db = 'ncbi'),
           auth = c('Linnaeus, 1758', 'Cohn 1872', NA, 'Juss., 1789'))

tax_name(x)
tax_name(x) <- tolower(tax_name(x))
tax_name(x)[1] <- 'Billy'
```

tax_rank.taxa_classification

Set and get taxon ranks

Description

Set and get the taxon ranks in objects that have them, such as taxon objects.

Usage

```
## S3 method for class 'taxa_classification'
tax_rank(x)

## S3 replacement method for class 'taxa_classification'
tax_rank(x) <- value

tax_rank(x)
tax_rank(x) <- value

## S3 method for class 'taxa_taxon'
tax_rank(x)

## S3 replacement method for class 'taxa_taxon'
tax_rank(x) <- value

## S3 method for class 'taxa_taxonomy'
tax_rank(x)

## S3 replacement method for class 'taxa_taxonomy'
tax_rank(x) <- value
```

Arguments

x	An object with taxon ranks.
value	The taxon ranks to set. Inputs will be coerced into a taxon_rank vector.

Examples

```
x <- taxon(name = c('Homo sapiens', 'Bacillus', 'Ascomycota', 'Ericaceae'),
           rank = c('species', 'genus', 'phylum', 'family'),
           id = taxon_id(c('9606', '1386', '4890', '4345'), db = 'ncbi'),
           auth = c('Linnaeus, 1758', 'Cohn 1872', NA, 'Juss., 1789'))

tax_rank(x)
tax_rank(x) <- 'species'
tax_rank(x)[1] <- taxon_rank('family')
```

%in% *Value matching for taxa package*

Description

A wrapper for the base value matching %in% that is used to take into consideration features of the taxa package.

Usage

```
x %in% table
```

Arguments

x	vector or NULL: the values to be matched. Long vectors are supported.
table	vector or NULL: the values to be matched against. Long vectors are not supported.

Index