

# Package ‘dataset’

March 18, 2023

**Title** Create Data Frames that are Easier to Exchange and Reuse

**Date** 2023-03-18

**Version** 0.2.1

**Description** The aim of the 'dataset' package is to make tidy datasets easier to release, exchange and reuse. It organizes and formats data frame 'R' objects into well-referenced, well-described, interoperable datasets into release and reuse ready form. A subjective interpretation of the W3C DataSet recommendation and the datacube model <<https://www.w3.org/TR/vocab-data-cube/>>, which is also used in the global Statistical Data and Metadata eXchange standards, the application of the connected Dublin Core <<https://www.dublincore.org/specifications/dublin-core/dcmi-terms/>> and DataCite <<https://support.datacite.org/docs/datacite-metadata-schema-44/>> standards preferred by European open science repositories to improve the findability, accessibility, interoperability and reusability of the datasets.

**License** GPL (>= 3)

**URL** <https://github.com/dataobservatory-eu/dataset>

**BugReports** <https://github.com/dataobservatory-eu/dataset/issues>

**Encoding** UTF-8

**RoxygenNote** 7.2.3

**Depends** R (>= 2.10)

**LazyData** true

**Imports** assertthat, ISOcodes, stats, utils

**Suggests** dataspice, covr, declared, dplyr, eurostat, here, kableExtra, knitr, rdflib, readxl, rmarkdown, spelling, statcodelists, testthat (>= 3.0.0), tidy

**VignetteBuilder** knitr

**Config/testthat/edition** 3

**Language** en-US

**NeedsCompilation** no

**Author** Daniel Antal [aut, cre] (<<https://orcid.org/0000-0001-7513-6760>>),  
 Marcelo Perlin [rev] (<<https://orcid.org/0000-0002-9839-4268>>)

**Maintainer** Daniel Antal <daniel.antal@dataobservatory.eu>

**Repository** CRAN

**Date/Publication** 2023-03-18 12:40:02 UTC

## R topics documented:

attributes_measures . . . . .	2
bibentry_dataset . . . . .	3
creator . . . . .	4
datacite . . . . .	5
dataset . . . . .	8
dataset_export . . . . .	10
dataset_local_id . . . . .	12
dataset_source . . . . .	13
dataset_title . . . . .	14
dataset_uri . . . . .	15
description . . . . .	16
dimensions . . . . .	17
dublincore . . . . .	18
geolocation . . . . .	20
identifier . . . . .	21
iris_dataset . . . . .	22
language . . . . .	23
measures . . . . .	24
publication_year . . . . .	25
publisher . . . . .	26
related_item . . . . .	27
resource_type . . . . .	29
resource_type_general . . . . .	30
rights . . . . .	31
size . . . . .	32
subject . . . . .	32
version . . . . .	34
<b>Index</b>	<b>35</b>

---

attributes\_measures    *Attributes of a dataset*

---

## Description

Attributes of a dataset

**Usage**

```
attributes_measures(x)

attributes_measures(x, sdmx_attributes = NULL) <- value
```

**Arguments**

`x` A data.frame or inherited tibble, data.frame, or a structured list.

`sdmx_attributes` The optional SDMX dimensions.

`value` The name or column number of the within the dataset.

**Details**

Do not confuse with `base::attributes`, which applies to the attributes of the entire dataset, and not each observation (measurement) row. See the W3C and SDMX definition of a [attribute](#).

**Value**

A data frame of the names, class, isDefinedBy, and codeList properties of the attributes columns of a dataset following the datacube model.

**Examples**

```
df <- data.frame ( sex = c("M", "F"), value = c(1,2), unit = c("NR", "NR"))
dimensions(df, sdmx_attributes = "sex") <- "sex"
measures(df) <- "value"
attributes_measures(df) <- "unit"
attributes_measures(df)
```

---

`bibentry_dataset`      *Create a bibentry for a dataset*

---

**Description**

Create a bibentry for a dataset

**Usage**

```
bibentry_dataset(ds)
```

**Arguments**

`ds` A data.frame or inherited tibble, data.frame, or a structured list.

**Value**

A bibentry object for the ds dataset.

## Examples

```
my_dataset <- dataset (
  x = data.frame (time = rep(c(2019:2022),2),
                  geo = c(rep("NL",4), rep("BE",4)),
                  value = c(1,3,2,4,2,3,1,5),
                  unit = rep("NR",8),
                  freq = rep("A",8)),
  Dimensions = c(1,2),
  Measures = 3,
  Attributes = c(4,5),
  sdmx_attributes = c("time", "freq"),
  Title = "Example dataset",
  Creator = person("Jane", "Doe"),
  Publisher = "Publishing Co.",
  Issued = as.Date("2022-07-14")
)

bibentry(my_dataset)
utils::toBibtex(bibentry_dataset(my_dataset))
```

---

creator

*Get/set the Creator of the object.*

---

## Description

Add the optional Creator property as an attribute to an R object.

## Usage

```
creator(x)

creator(x, overwrite = TRUE) <- value
```

## Arguments

x	An R object, such as a data.frame, a tibble, or a data.table.
overwrite	If the attributes should be overwritten. In case it is set to FALSE, it gives a message with the current Creator property instead of overwriting it. Defaults to TRUE when the attribute is set to value regardless of previous setting.
value	The Creator as a <code>utils::person</code> object.

## Details

The Creator corresponds to `dct:creator` and Creator in DataCite. The name of the entity that holds, archives, publishes prints, distributes, releases, issues, or produces the resource. This property will be used to formulate the citation, so consider the prominence of the role. For software, use Creator for the code repository. If there is an entity other than a code repository, that "holds, archives, publishes, prints, distributes, releases, issues, or produces" the code, use the property Contributor/contributorType/hostingInstitution for the code repository.

**Value**

The Creator attribute as a character of length 1 is added to x.

**See Also**

Other Reference metadata functions: [dataset\\_source\(\)](#), [description\(\)](#), [geolocation\(\)](#), [identifier\(\)](#), [language](#), [publication\\_year\(\)](#), [publisher\(\)](#), [rights\(\)](#), [size\(\)](#), [version\(\)](#)

**Examples**

```
iris_dataset <- iris
creator(iris_dataset) <- person("Anderson", "Edgar", role = "aut")
creator(iris_dataset)
```

---

datacite

*Add DataCite metadata to an object*

---

**Description**

Add metadata conforming the [DataCite Metadata Schema](#) to datasets, i.e. structured R data.frame or list objects, for an accurate and consistent identification of a resource for citation and retrieval purposes.

**Usage**

```
datacite(x)

datacite_add(
  x,
  Title,
  titleType = NULL,
  Creator,
  Identifier = NULL,
  Publisher = NULL,
  PublicationYear = "THIS",
  Subject = NULL,
  Type = "Dataset",
  Contributor = NULL,
  Date = NULL,
  Language = NULL,
  AlternateIdentifier = NULL,
  RelatedIdentifier = NULL,
  Format = NULL,
  Version = NULL,
  Rights = NULL,
  Description = NULL,
  Geolocation = NULL,
```

```

  FundingReference = NULL,
  overwrite = TRUE
)

```

## Arguments

x	An R object of type <code>data.frame</code> , or inherited <code>data.table</code> , <code>tibble</code> ; alternatively a well structured R list.
Title	The name(s) or title(s) by which a resource is known. May be the title of a dataset or the name of a piece of software. Similar to <code>dct:title</code> . See <a href="#">dataset_title</a> for adding further titles.
titleType	For a single Title defaults to NULL. Otherwise you can add a Subtitle, an Alternative Title and an Other Title. See <a href="#">dataset_title</a> .
Creator	The main researchers involved in producing the data, or the authors of the publication, in priority order. To supply multiple creators, repeat this property.
Identifier	The Identifier is a unique string that identifies a resource. For software, determine whether the identifier is for a specific version of a piece of software, (per the <a href="#">Force11 Software Citation Principles</a> , or for all versions. Similar to <code>dct:title</code> in <a href="#">dublincore</a> .
Publisher	The name of the entity that holds, archives, publishes prints, distributes, releases, issues, or produces the resource. This property will be used to formulate the citation, so consider the prominence of the role. For software, use Publisher for the code repository. Mandatory in DataCite, and similar to <code>dct:publisher</code> . See <a href="#">publisher</a> .
PublicationYear	The year when the data was or will be made publicly available in YYYY format. See <a href="#">publication_year</a> .
Subject	Recommended for discovery. Subject, keyword, classification code, or key phrase describing the resource. Similar to <code>dct:subject</code> . Use <a href="#">subject</a> to properly add a key phrase from a controlled vocabulary and create structured Subject objects with <a href="#">subject_create</a> .
Type	Defaults to Dataset. The DataCite resourceType definition refers back to <code>dcm:type</code> . The <code>Type\$resourceTypeGeneral</code> is set to Dataset, while the user can set a more specific <code>Type\$resourceType</code> value. See <a href="#">resource_type</a> .
Contributor	Recommended for discovery. The institution or person responsible for collecting, managing, distributing, or otherwise contributing to the development of the resource.
Date	Recommended for discovery in DataCite. Similar to <code>dct:date</code> in <a href="#">dublincore</a> .
Language	The primary language of the resource. Allowed values are taken from IETF BCP 47, ISO 639-1 language code. See <a href="#">language</a> .
AlternateIdentifier	An identifier or identifiers other than the primary Identifier applied to the resource being registered. This may be any alphanumeric string which is unique within its domain of issue. May be used for local identifiers. AlternateIdentifier should be used for another identifier of the same instance (same location, same file).

RelatedIdentifier	Recommended for discovery. Similar to <a href="#">dct:relation</a> .
Format	Technical format of the resource. Similar to <a href="#">dct:format</a> .
Version	Free text. Suggested practice: track major_version.minor_version. See <a href="#">version</a> .
Rights	Any rights information for this resource. The property may be repeated to record complex rights characteristics. Free text. See <a href="#">rights</a> .
Description	Recommended for discovery. All additional information that does not fit in any of the other categories. May be used for technical information. A free text. Similar to <a href="#">dct:description</a> .
Geolocation	Recommended for discovery. Spatial region or named place where the data was gathered or about which the data is focused. See <a href="#">geolocation</a> .
FundingReference	Information about financial support (funding) for the resource being registered.
overwrite	If pre-existing metadata properties should be overwritten, defaults to TRUE.

## Details

DataCite is a leading global non-profit organisation that provides persistent identifiers (DOIs) for research data and other research outputs. Organizations within the research community join DataCite as members to be able to assign DOIs to all their research outputs. This way, their outputs become discoverable and associated metadata is made available to the community.

DataCite then develops additional services to improve the DOI management experience, making it easier for our members to connect and share their DOIs with the broader research ecosystem and to assess the use of their DOIs within that ecosystem. DataCite is an active participant in the research community and promotes data sharing and citation through community-building efforts and outreach activities.

The ResourceType property will be by definition "Dataset". The Size attribute (e.g. bytes, pages, inches, etc.) will automatically added to the dataset.

## Value

An R object with at least the mandatory DataCite attributes.

## Source

[DataCite 4.3 Mandatory Properties](#) and [DataCite 4.3 Optional Properties](#)

## See Also

Other metadata functions: [dublincore\(\)](#), [related\\_item\(\)](#)

## Examples

```
my_iris <- datacite_add(
  x = iris,
  Title = "Iris Dataset",
  Creator = person(family = "Anderson", given = "Edgar", role = "aut"),
  Publisher = "American Iris Society",
```

```

    PublicationYear = 1935,
    Geolocation = "US",
    Language = "en")

datacite(my_iris)

```

---

dataset

*Structure a data frame to dataset*


---

## Description

A DataSet is a collection of statistical data that corresponds to a defined structure.

## Usage

```

dataset(
  x,
  Dimensions = NULL,
  Measures = NULL,
  Attributes = NULL,
  sdmx_attributes = NULL,
  Title = NULL,
  Label = NULL,
  Creator = NULL,
  Publisher = NULL,
  Issued = NULL,
  Identifier = NULL,
  Subject = NULL,
  Type = "DCMITYPE:Dataset"
)

is.dataset(x)

as.data.frame(x, ...)

## S3 method for class 'dataset'
as.data.frame(x, ...)

## S3 method for class 'dataset'
subset(x, ...)

## S3 method for class 'dataset'
x[i, j, ...]

## S3 method for class 'dataset'
summary(object, ...)

## S3 method for class 'dataset'
print(x, ...)

```



**Arguments**

<code>x</code>	A <code>data.frame</code> or inherited tibble, <code>data.frame</code> , or a structured list.
<code>Dimensions</code>	The name or column number of the dimensions within the dataset.
<code>Measures</code>	The name or column number of the measures within the dataset.
<code>Attributes</code>	The name or column number of the attributes within the dataset.
<code>sdmx_attributes</code>	The optional dimensions and attributes that conform with SDMX. <code>c("time", "geo")</code> will mark the "time" and "geo" attributes as conforming to sdmx. See <a href="#">sdmx-attribute</a> .
<code>Title</code>	<a href="#">dct:title</a> , a name given to the resource. <a href="#">datacite</a> allows the use of alternate titles, too. See <a href="#">dataset_title</a> .
<code>Label</code>	may be used to provide a human-readable version of the dataset's name. A text description (optionally with a language tag) as defined by <a href="#">rdfs:label</a> .
<code>Creator</code>	An entity primarily responsible for making the resource. <a href="#">dct:creator</a> Corresponds to Creator in <a href="#">datacite</a> . See <a href="#">creator</a> .
<code>Publisher</code>	Corresponds to <a href="#">dct:publisher</a> and Publisher in DataCite. The name of the entity that holds, archives, publishes prints, distributes, releases, issues, or produces the resource. This property will be used to formulate the citation, so consider the prominence of the role. For software, use Publisher for the code repository. If there is an entity other than a code repository, that "holds, archives, publishes, prints, distributes, releases, issues, or produces" the code, use the property Contributor/contributorType/hostingInstitution for the code repository. See <a href="#">publisher</a> .
<code>Issued</code>	Corresponds to <a href="#">dct:date</a> .
<code>Identifier</code>	An unambiguous reference to the resource within a given context. Recommended practice is to identify the resource by means of a string conforming to an identification system. Examples include International Standard Book Number (ISBN), Digital Object Identifier (DOI), and Uniform Resource Name (URN). Select an identifier scheme from <a href="#">registered URI schemes maintained by IANA</a> . More details: <a href="#">Guidelines for using resource identifiers in Dublin Core metadata and IEEE LOM</a> . Similar to Identifier in <a href="#">datacite</a> . See <a href="#">identifier</a> .
<code>Subject</code>	Recommended for discovery in DataCite. Subject, keyword, classification code, or key phrase describing the resource. Similar to <a href="#">dct:subject</a> . Use <a href="#">subject</a> to properly add a key phrase from a controlled vocabulary and create structured Subject objects with <a href="#">subject_create</a> .
<code>Type</code>	It is set by default to <a href="#">DCMITYPE:Dataset</a> .
<code>...</code>	Other parameters for the <code>print</code> , <code>summary</code> and <code>as.data.frame</code> methods.
<code>i</code>	elements to extract or replace: numeric, character, empty or logical.
<code>j</code>	elements to extract or replace: numeric, character, empty or logical.
<code>object</code>	an object for which a summary is desired.

**Details**

Loosely follows the [The RDF Data Cube Vocabulary](#), but without the definition of data slices. `bibentry_dataset` is a wrapper around `bibentry` to correctly turn the metadata of the dataset into a `bibentry` object. `as.data.frame` coerces a dataset into a `data.frame` in a way that the metadata attributes are retained.

**Value**

A data frame-like object with structural and referential metadata.

**See Also**

`iris_dataset`

Other dataset functions: `dataset_local_id()`, `dataset_uri()`

**Examples**

```
my_dataset <- dataset (
  x = data.frame (time = rep(c(2019:2022),2),
                  geo = c(rep("NL",4), rep("BE",4)),
                  value = c(1,3,2,4,2,3,1,5),
                  unit = rep("NR",8),
                  freq = rep("A",8)),
  Dimensions = c(1,2),
  Measures = 3,
  Attributes = c(4,5),
  sdmx_attributes = c("time", "freq"),
  Title = "Example dataset",
  Creator = person("Jane", "Doe"),
  Publisher = "Publishing Co.",
  Issued = as.Date("2022-07-14")
)
## iris_dataset is a dataset class version of iris
as.data.frame(iris_dataset)
```

---

dataset\_export

*Export a dataset*

---

**Description**

Export a dataset together with reference (DataCite and Dublin Core) metadata.

**Usage**

```
dataset_export(ds, file, filetype = "csv", ...)
```

```
dataset_export_csv(ds, file)
```

**Arguments**

ds	A dataset object.
file	A (path to) a file where to export the dataset object.
filetype	Currently only 'csv' is implemented.
...	Further parameters to be passed on to exporting functions. See details.

**Details**

This function is a wrapper around the exporting functions. It implements file exports in a way that the resulting exported file contains reference metadata.

[dataset\\_export\\_csv](#) is a wrapper around `utils::write.csv`. Use ... to pass on argument to that function.

**Value**

The function write a desired file on disc and does not return anything.

**See Also**

dataset

**Examples**

```
my_iris_dataset <- dataset(
  x = iris,
  Dimensions = NULL,
  Measures = c("Sepal.Length", "Sepal.Width", "Petal.Length", "Petal.Width" ),
  Attributes = "Species",
  Title = "Iris Dataset"
)

my_iris_dataset <- dublincore_add(
  x = my_iris_dataset,
  Creator = person("Edgar", "Anderson", role = "aut"),
  Publisher = "American Iris Society",
  Source = "https://doi.org/10.1111/j.1469-1809.1936.tb02137.x",
  Date = 1935,
  Language = "en"
)

dataset_export_csv(my_iris_dataset, file = file.path(tempdir(), "my_iris.csv"))

read.csv(file.path(tempdir(), "my_iris.csv"), skip=20)
read.csv(file.path(tempdir(), "my_iris.csv"))
```

---

dataset_local_id	<i>Create a locally unique id</i>
------------------	-----------------------------------

---

### Description

Add a locally unique row identifier to a dataset object.

### Usage

```
dataset_local_id(ds)
```

### Arguments

ds                    A dataset object.

### Value

A dataset object with a locally unique row identifier added as a primary key to the tabular form.

### See Also

Other dataset functions: [dataset\\_uri\(\)](#), [dataset\(\)](#)

### Examples

```
my_ds <- dataset (x = data.frame (
  time = rep(c(2019:2022),4),
  geo = c(rep("NL",8), rep("BE",8)),
  sex = c(rep("F", 4), rep("M", 4), rep("F", 4), rep("M", 4)),
  value = c(1,3,2,4,2,3,1,5, NA_real_, 4,3,2,1, NA_real_, 2,5),
  unit = rep("NR",8),
  freq = rep("A",8)),
  Dimensions = c("time", "geo", "sex"),
  Measures = "value",
  Attributes = c("unit", "freq"),
  sdmx_attributes = c("sex", "time", "freq"),
  Title = "Example dataset",
  Creator = person("Jane", "Doe"),
  Publisher = "Publishing Co.",
  Issued = as.Date("2022-07-14")
)
```

---

dataset_source	<i>Get/set the Source of the object.</i>
----------------	--

---

### Description

Get/set the optional Source property as an attribute to an R object. Do not confuse with the base R source() function.

### Usage

```
dataset_source(x)
```

```
dataset_source(x, overwrite = TRUE) <- value
```

### Arguments

x	An R object of type data.frame, or inherited data.table, tibble; alternatively a well structured R list.
overwrite	If pre-existing metadata properties should be overwritten, defaults to TRUE.
value	The Source as a character string of lengths one.

### Details

The Source is a related resource from which the described resource is derived. See [dct:source](#). In Datacite, the source is described by a relatedIdentifierType with the property relationType="isDerivedFrom".

### Value

The Source attribute as a character of length 1 is added to x.

### See Also

Other Reference metadata functions: [creator\(\)](#), [description\(\)](#), [geolocation\(\)](#), [identifier\(\)](#), [language](#), [publication\\_year\(\)](#), [publisher\(\)](#), [rights\(\)](#), [size\(\)](#), [version\(\)](#)

### Examples

```
iris_dataset <- iris
dataset_source(iris_dataset) <- "https://doi.org/10.1111/j.1469-1809.1936.tb02137.x"
dataset_source(iris_dataset)
```

---

dataset_title	<i>Get/set title(s) of a dataset</i>
---------------	--------------------------------------

---

### Description

Add one or more Title(s) to the dataset's metadata.

### Usage

```
dataset_title(x)

dataset_title(x, overwrite = FALSE) <- value

dataset_title_create(Title, titleType = "Title")
```

### Arguments

x	An R object
overwrite	Defaults to FALSE.
value	The name(s) or title(s) by which a resource is known. A character string or a Title object created by <code>dataset_title_create</code> . Similar to <code>dct:title</code> .
Title	The name(s) or title(s) by which a resource is known, including Title, AlternativeTitle, Subtitle, TranslatedTitle, OtherTitle. May be the title of a dataset or the name of a piece of software. Similar to <code>dct:title</code> . Use <code>dataset_title_create</code> to create a several title entries.
titleType	In DataCite, the controlled values are AlternativeTitle, Subtitle, TranslatedTitle, Other. When no titleType is given (as in Dublin Core), the titleType is set to Title.

### Details

In the DataCite definition, several titles can be used.

### Value

The titles as a data.frame with a titleTypes column.

### Examples

```
my_iris <- iris

dataset_title(my_iris) <- dataset_title_create(
  Title = c("Iris Dataset",
            "The famous iris dataset of the R examples"),
  titleType = c("Title", "Subtitle")
)
dataset_title(my_iris)
```

```

y <- data.frame()
dataset_title(y) <- "R (Computer program language)"
dataset_title(y) <- "Questionnaires--Computer programs"
dataset_title(y)

```

---

dataset_uri	<i>Create a globally unique row identifier</i>
-------------	--

---

## Description

Add a globally unique row identifier to a dataset object.

## Usage

```

dataset_uri(
  ds,
  prefix = "https://example.org/my_data/",
  keep_local_id = FALSE
)

```

## Arguments

ds	A dataset object.
prefix	The prefix of the globally unique row identifier (URI or CURIE), defaults to "https://example.org/my_data/".
keep_local_id	Defaults to FALSE.

## Value

A dataset object with a locally unique row identifier added as a primary key to the tabular form.

## See Also

Other dataset functions: [dataset\\_local\\_id\(\)](#), [dataset\(\)](#)

## Examples

```

my_ds <- dataset (x = data.frame (
  time = rep(c(2019:2022),4),
  geo = c(rep("NL",8), rep("BE",8)),
  sex = c(rep("F", 4), rep("M", 4), rep("F", 4), rep("M", 4)),
  value = c(1,3,2,4,2,3,1,5, NA_real_, 4,3,2,1, NA_real_, 2,5),
  unit = rep("NR",8),
  freq = rep("A",8)),
  Dimensions = c("time", "geo", "sex"),
  Measures = "value",
  Attributes = c("unit", "freq"),

```

```

sdmx_attributes = c("sex", "time", "freq"),
Title = "Example dataset",
Creator = person("Jane", "Doe"),
Publisher = "Publishing Co.",
Issued = as.Date("2022-07-14")
)
dataset_uri(my_ds)

```

---

description

*Get/set the Description of the object.*


---

## Description

Get/set the optional Description property as an attribute to an R object.

## Usage

```

description(x)

description(x, overwrite = TRUE) <- value

```

## Arguments

x	An R object, such as a data.frame, a tibble, or a data.table.
overwrite	If the Description attribute should be overwritten. In case it is set to FALSE, it gives a message with the current Description property instead of overwriting it. Defaults to TRUE when the attribute is set to value regardless of previous setting.
value	The Description as a character set.

## Details

The Description is recommended for discovery in DataCite. All additional information that does not fit in any of the other categories. May be used for technical information. A free text. Similar to [dct:description](#).

## Value

The Description attribute as a character of length 1 is added to x.

## See Also

Other Reference metadata functions: [creator\(\)](#), [dataset\\_source\(\)](#), [geolocation\(\)](#), [identifier\(\)](#), [language](#), [publication\\_year\(\)](#), [publisher\(\)](#), [rights\(\)](#), [size\(\)](#), [version\(\)](#)

## Examples

```

iris_dataset <- iris
description(iris_dataset) <- "The famous iris dataset used in R language examples."
description(iris_dataset)

```



---

dimensions	<i>Dimensions of a dataset</i>
------------	--------------------------------

---

## Description

Dimensions of a dataset

## Usage

```
dimensions(x)

dimensions(x, sdmx_attributes = NULL) <- value
```

## Arguments

x	A data.frame or inherited tibble, data.frame, or a structured list.
sdmx_attributes	The optional dimensions and attributes that conform with SDMX. c("time", "geo") will mark the "time" and "geo" attributes as conforming to sdmx. See <a href="#">sdmx-attribute</a> .
value	The name or column number of the within the dataset.

## Details

Do not confuse with `base::dim`. The **dimension** in the definition of the `DataSet` is different from the 'dimension' definition of the R language.

## Value

A data frame of the names, class, `isDefinedBy`, and `codeList` properties of the dimensions columns of the dataset following the datacube model.

## Examples

```
df <- data.frame ( sex = c("M", "F"), value = c(1,2), unit = c("NR", "NR"))
dimensions(df, sdmx_attributes = "sex") <- "sex"
measures(df) <- "value"
attributes_measures(df) <- "unit"
dimensions(df)
```

---

`dublincore`*Add DublinCore metadata to an object*

---

## Description

Add metadata conforming the [DataCite Metadata Schema](#) to datasets, i.e. structured R data.frame or list objects, for an accurate and consistent identification of a resource for citation and retrieval purposes.

## Usage

```
dublincore(x)

dublincore_add(
  x,
  Title = NULL,
  Creator = NULL,
  Identifier = NULL,
  Publisher = NULL,
  Subject = NULL,
  Date = NULL,
  Source = NULL,
  Language = NULL,
  Format = NULL,
  Rights = NULL,
  Relation = NULL,
  Description = NULL,
  Type = "DCMITYPE:Dataset",
  overwrite = TRUE
)
```

## Arguments

<code>x</code>	An R object of type <code>data.frame</code> , or inherited <code>data.table</code> , <code>tibble</code> ; alternatively a well structured R list.
<code>Title</code>	<code>dct:title</code> , a name given to the resource. <code>datacite</code> allows the use of alternate titles, too. See <code>dataset_title</code> .
<code>Creator</code>	An entity primarily responsible for making the resource. <code>dct:creator</code> Corresponds to <code>Creator</code> in <code>datacite</code> . See <code>creator</code> .
<code>Identifier</code>	An unambiguous reference to the resource within a given context. Recommended practice is to identify the resource by means of a string conforming to an identification system. Examples include International Standard Book Number (ISBN), Digital Object Identifier (DOI), and Uniform Resource Name (URN). Select an identifier scheme from <a href="#">registered URI schemes maintained by IANA</a> . More details: <a href="#">Guidelines for using resource identifiers in Dublin Core metadata and IEEE LOM</a> . Similar to <code>Identifier</code> in <code>datacite</code> . See <code>identifier</code> .

Publisher	Corresponds to <b>dct:publisher</b> and Publisher in DataCite. The name of the entity that holds, archives, publishes prints, distributes, releases, issues, or produces the resource. This property will be used to formulate the citation, so consider the prominence of the role. For software, use Publisher for the code repository. If there is an entity other than a code repository, that "holds, archives, publishes, prints, distributes, releases, issues, or produces" the code, use the property Contributor/contributorType/hostingInstitution for the code repository. See <a href="#">publisher</a> .
Subject	Defaults to NULL. See <a href="#">subject</a> to add subject descriptions to your dataset.
Date	Corresponds to a point or period of time associated with an event in the lifecycle of the resource. <b>dct:date</b> . Date is also recommended for discovery in <a href="#">datacite</a> .
Source	A related resource from which the described resource is derived. See <b>dct:source</b> and <a href="#">dataset_source</a> .
Language	The primary language of the resource. Allowed values are taken from IETF BCP 47, ISO 639-1 language code. See <a href="#">language</a> . Corresponds to Language in Datacite.
Format	The file format, physical medium, or dimensions of the resource. <b>dct:format</b> Examples of dimensions include size and duration. Recommended best practice is to use a controlled vocabulary such as the list of <b>Internet Media Types, formerly known as MIME</b> . It is similar to Format in <a href="#">datacite</a> .
Rights	Corresponds to <b>dct:rights</b> and <a href="#">datacite</a> Rights. Information about rights held in and over the resource. Typically, rights information includes a statement about various property rights associated with the resource, including intellectual property rights. See <a href="#">rights</a> .
Relation	A related resource. Recommended best practice is to identify the related resource by means of a string conforming to a formal identification system. See: <b>dct:relation</b> . Similar to RelatedItem in <a href="#">datacite</a> , which is recommended for discovery.
Description	An account of the resource. It may include but is not limited to: an abstract, a table of contents, a graphical representation, or a free-text account of the resource. <b>dct:description</b> . In <a href="#">datacite</a> it is recommended for discovery. See <a href="#">description</a> .
Type	The nature or genre of the resource. Recommended best practice is to use a controlled vocabulary such as the DCMI Type Vocabulary <b>DCMITYPE</b> . For a dataset, the correct term is Dataset. To describe the file format, physical medium, or dimensions of the resource, use the Format element.
overwrite	If pre-existing metadata properties should be overwritten, defaults to TRUE.

## Details

DataCite is a leading global non-profit organisation that provides persistent identifiers (DOIs) for research data and other research outputs. Organizations within the research community join DataCite as members to be able to assign DOIs to all their research outputs. This way, their outputs become discoverable and associated metadata is made available to the community. DataCite then develops additional services to improve the DOI management experience, making it easier for our members to connect and share their DOIs with the broader research ecosystem and to assess the use

of their DOIs within that ecosystem. DataCite is an active participant in the research community and promotes data sharing and citation through community-building efforts and outreach activities.

The ResourceType property will be by definition "Dataset". The Size attribute (e.g. bytes, pages, inches, etc.) will automatically added to the dataset.

**Value**

The Dublin Core Metadata elements of the dataset.

**Source**

[DataCite 4.3 Mandatory Properties](#) and [DataCite 4.3 Optional Properties](#)

**See Also**

Other metadata functions: [datacite\(\)](#), [related\\_item\(\)](#)

**Examples**

```
dct_iris <- dublincore_add(  
  x = iris,  
  Title = "Iris Dataset",  
  Creator = person("Anderson", "Edgar", role = "aut"),  
  Publisher = "American Iris Society",  
  Source = "https://doi.org/10.1111/j.1469-1809.1936.tb02137.x",  
  Date = 1935,  
  Language = "en"  
)  
  
dublincore(dct_iris)
```

---

geolocation

*Get/set the Geolocation of the object.*

---

**Description**

Get/set the optional Geolocation property as an attribute to an R object.

**Usage**

```
geolocation(x)
```

```
geolocation(x, overwrite = TRUE) <- value
```

**Arguments**

x	An R object, such as a data.frame, a tibble, or a data.table.
overwrite	If the attributes should be overwritten. In case it is set to FALSE, it gives a message with the current Geolocation property instead of overwriting it. Defaults to TRUE when the attribute is set to value regardless of previous setting.
value	The Geolocation as a character string.

**Details**

The Geolocation is recommended for discovery in DataCite. Spatial region or named place where the data was gathered or about which the data is focused.

**Value**

The Geolocation attribute as a character of length 1 is added to x.

**See Also**

Other Reference metadata functions: [creator\(\)](#), [dataset\\_source\(\)](#), [description\(\)](#), [identifier\(\)](#), [language](#), [publication\\_year\(\)](#), [publisher\(\)](#), [rights\(\)](#), [size\(\)](#), [version\(\)](#)

**Examples**

```
iris_dataset <- iris
geolocation(iris_dataset) <- "US"
geolocation(iris_dataset)

geolocation(iris_dataset, overwrite = FALSE) <- "GB"
```

---

identifier	<i>Get/set the Identifier of the object.</i>
------------	--

---

**Description**

Add the optional Identifier property as an attribute to an R object.

**Usage**

```
identifier(x)

identifier(x, overwrite = TRUE) <- value
```

**Arguments**

x	An R object, such as a data.frame, a tibble, or a data.table.
overwrite	If the attributes should be overwritten. In case it is set to FALSE, it gives a message with the current Identifier property instead of overwriting it. Defaults to TRUE when the attribute is set to value regardless of previous setting.
value	The Identifier as a character string.

**Details**

The Identifier is an unambiguous reference to the resource within a given context. Recommended practice is to identify the resource by means of a string conforming to an identification system. Examples include International Standard Book Number (ISBN), Digital Object Identifier (DOI), and Uniform Resource Name (URN). Select and identifier scheme from [registered URI schemes maintained by IANA](#). More details: [Guidelines for using resource identifiers in Dublin Core metadata and IEEE LOM](#). Similar to Identifier in [datacite. DataCite 4.3](#). It is not part of the "core" Dublin Core terms, but we always add it to the metadata attributes of a dataset (in case you use a strict Dublin Core property sheet you can omit it.) [Dublin Core metadata terms](#).

**Value**

The Identifier attribute as a character of length 1 is added to x.

**See Also**

Other Reference metadata functions: [creator\(\)](#), [dataset\\_source\(\)](#), [description\(\)](#), [geolocation\(\)](#), [language](#), [publication\\_year\(\)](#), [publisher\(\)](#), [rights\(\)](#), [size\(\)](#), [version\(\)](#)

**Examples**

```
iris_dataset <- iris
identifier(iris_dataset) <- "https://doi.org/10.1111/j.1469-1809.1936.tb02137.x"
identifier(iris_dataset)
```

---

iris\_dataset

*Edgar Anderson's Iris Data*


---

**Description**

This famous (Fisher's or Anderson's) iris data set gives the measurements in centimeters of the variables sepal length and width and petal length and width, respectively, for 50 flowers from each of 3 species of iris. The species are *Iris setosa*, *versicolor*, and *virginica*. This is a replication of datasets::iris as dataset s3 class.

**Usage**

```
iris_dataset
```

**Format**

iris is a data frame with 150 cases (rows) and 5 variables (columns) named Sepal.Length, Sepal.Width, Petal.Length, Petal.Width, and Species.

**Details**

See `datasets::iris` for details.

**Source**

Fisher, R. A. (1936) The use of multiple measurements in taxonomic problems. *Annals of Eugenics*, 7, Part II, p179–188.

The data were collected by Anderson, Edgar (1935). The irises of the Gaspe Peninsula, *Bulletin of the American Iris Society*, 59, 2–5.

**References**

Becker, R. A., Chambers, J. M. and Wilks, A. R. (1988) *The New S Language*. Wadsworth & Brooks/Cole.

---

language

*Get/Set the primary language of the dataset*

---

**Description**

Add the optional Language property as an attribute to an R object.

**Usage**

```
language(x)
```

```
language(x, iso_639_code = "639-3") <- value
```

**Arguments**

x	An R object, such as a data.frame, a tibble, or a character vector.
iso_639_code	Defaults to ISO 639-3, alternative is ISO 639-1.
value	The language to be added to the object attributes, added by name, or as a 2- or 3-character code for the language. You can add a language code or language name, and the parameter is normalized to <code>tolower(language)</code> . (The ISO 639 standard capitalizes language names and uses lower case for the codes.)

**Details**

Language is an optional property in [DataCite 4.3](#) and it is part of the "core" of the [Dublin Core metadata terms](#). The language parameter is validated against the `[ISOcodes]{ISO_639_2}` table. The attribute language is added to the object. It will be exported into DataCite applications in a capitalized Lanugage format.

**Value**

The Language is added to the x as ISO 639-1, the Datacite recommendation, or ISO 639-3 used by the Zenodo data repository.

**See Also**

Other Reference metadata functions: [creator\(\)](#), [dataset\\_source\(\)](#), [description\(\)](#), [geolocation\(\)](#), [identifier\(\)](#), [publication\\_year\(\)](#), [publisher\(\)](#), [rights\(\)](#), [size\(\)](#), [version\(\)](#)

**Examples**

```
iris_dataset <- iris
language(iris_dataset) <- "English"
language(iris_dataset)
```

---

measures

*Measures of a dataset*


---

**Description**

Measures of a dataset

**Usage**

```
measures(x)
```

```
measures(x) <- value
```

**Arguments**

**x** A data.frame or inherited tibble, data.frame, or a structured list.  
**value** The name or column number of the within the dataset.

**Details**

See the W3C and SDMX definition of a [measure](#).

**Value**

A data frame of the names, class, isDefinedBy, and codeList properties of the measurement columns of a dataset following the datacube model.



**Examples**

```
df <- data.frame ( sex = c("M", "F"), value = c(1,2), unit = c("NR", "NR"))
dimensions(df, sdmx_attributes = "sex") <- "sex"
measures(df) <- "value"
attributes_measures(df) <- "unit"
measures(df)
df <- data.frame ( sex = c("M", "F"), value = c(1,2))
measures(df) <- "value"
measures(df)
```

---

publication_year	<i>Get/set the publication_year of the object.</i>
------------------	--

---

**Description**

Get/set the optional publication\_year property as an attribute to an R object.

**Usage**

```
publication_year(x)

publication_year(x, overwrite = TRUE) <- value
```

**Arguments**

x	An R object, such as a data.frame, a tibble, or a data.table.
overwrite	If the attributes should be overwritten. In case it is set to FALSE, it gives a message with the current PublicationYear property instead of overwriting it. Defaults to TRUE when the attribute is set to value regardless of previous setting.
value	The publication_year as a character set.

**Details**

The PublicationYear is the year when the data was or will be made publicly available in YYYY format.

**Value**

The publication\_year attribute as a character of length 1 is added to x.

**See Also**

Other Reference metadata functions: [creator\(\)](#), [dataset\\_source\(\)](#), [description\(\)](#), [geolocation\(\)](#), [identifier\(\)](#), [language](#), [publisher\(\)](#), [rights\(\)](#), [size\(\)](#), [version\(\)](#)

**Examples**

```
iris_dataset <- iris
publication_year(iris_dataset) <- 1935
publication_year(iris_dataset)
```

---

publisher	<i>Get/set the Publisher of the object.</i>
-----------	---

---

**Description**

Add the optional Publisher property as an attribute to an R object.

**Usage**

```
publisher(x)

publisher(x, overwrite = TRUE) <- value
```

**Arguments**

x	An R object, such as a data.frame, a tibble, or a data.table.
overwrite	If the attributes should be overwritten. In case it is set to FALSE, it gives a message with the current Publisher property instead of overwriting it. Defaults to TRUE when the attribute is set to value regardless of previous setting.
value	The Publisher as a character set.

**Details**

The Publisher corresponds to dct:publisher and Publisher in DataCite. The name of the entity that holds, archives, publishes prints, distributes, releases, issues, or produces the resource. This property will be used to formulate the citation, so consider the prominence of the role. For software, use Publisher for the code repository. If there is an entity other than a code repository, that "holds, archives, publishes, prints, distributes, releases, issues, or produces" the code, use the property Contributor/contributorType/ hostingInstitution for the code repository.

**Value**

The Publisher attribute as a character of length 1 is added to x.

**See Also**

Other Reference metadata functions: [creator\(\)](#), [dataset\\_source\(\)](#), [description\(\)](#), [geolocation\(\)](#), [identifier\(\)](#), [language](#), [publication\\_year\(\)](#), [rights\(\)](#), [size\(\)](#), [version\(\)](#)

**Examples**

```
iris_dataset <- iris
publisher(iris_dataset) <- "American Iris Society"
publisher(iris_dataset)
```

---

related_item	<i>Create a related item</i>
--------------	------------------------------

---

## Description

Create a **RelatedIdentifier**, attribute, which is recommended for discovery in DataCite.

## Usage

```
related_item(
  Identifier,
  Creator,
  Title,
  relatedIdentifierType,
  relationType,
  schemeURI = NA_character_,
  schemeType = NA_character_,
  resourceTypeGeneral = NA_character_,
  PublicationYear = NULL,
  Volume = NULL,
  Issue = NULL,
  Number = NULL,
  numberType = NULL,
  firstPage = NULL,
  lastPage = NULL,
  Publisher = NULL,
  Edition = NULL,
  Contributor = NULL
)
```

## Arguments

Identifier	The Identifier is a unique string that identifies a resource. For software, determine whether the identifier is for a specific version of a piece of software, (per the <a href="#">Force11 Software Citation Principles</a> , or for all versions. Similar to <code>dct:title</code> in <a href="#">dublincore</a> .
Creator	The main researchers involved in producing the data, or the authors of the publication, in priority order. To supply multiple creators, repeat this property.
Title	The name(s) or title(s) by which a resource is known. May be the title of a dataset or the name of a piece of software. Similar to <code>dct:title</code> . See <a href="#">dataset_title</a> for adding further titles.
relatedIdentifierType	See <a href="#">relatedIdentifierType</a> .
relationType	See <a href="#">relationType</a> .
schemeURI	See <a href="#">schemeURI</a> .

schemeType	See <a href="#">schemeType</a> .
resourceTypeGeneral	The general type of a resource or file. See <a href="#">resource_type_general</a> for allowed values and validation.
PublicationYear	The year when the data was or will be made publicly available in YYYY format. See <a href="#">publication_year</a> .
Volume	The volume of the related item (optional).
Issue	The issue number of the related item (optional).
Number	The number of the related item (optional).
numberType	The type of the number (optional).
firstPage	The first page of the related item (optional).
lastPage	The first page of the related item (optional).
Publisher	The name of the entity that holds, archives, publishes prints, distributes, releases, issues, or produces the resource. This property will be used to formulate the citation, so consider the prominence of the role. For software, use Publisher for the code repository. Mandatory in DataCite, and similar to dct:publisher. See <a href="#">publisher</a> .
Edition	The edition of the related item (optional).
Contributor	Recommended for discovery. The institution or person responsible for collecting, managing, distributing, or otherwise contributing to the development of the resource.

### Value

a related item.

### See Also

Other metadata functions: [datacite\(\)](#), [dublincore\(\)](#)

### Examples

```
my_item <- related_item (Identifier = "https://zenodo.org/record/5703222#.YZYkm2DMLIU",
  Creator = person ("Daniel", "Antal", role = "aut"),
  Publisher = "Zenodo",
  PublicationYear = 2022,
  relatedIdentifierType = "DOI",
  relationType = "CompiledBy",
  schemeURI = "URI",
  resourceTypeGeneral = "Dataset")
```

---

resource_type	<i>Get/set the resource type of the dataset</i>
---------------	---

---

### Description

Get/set Type property to the dataset.

### Usage

```
resource_type(x)
```

```
resource_type(x) <- value
```

### Arguments

x	An R dataset object inherited from data.frame, tibble, or data.table.
value	The Type\$resourceTypeGeneral is set to Dataset, while the user can set a more specific Type\$resourceType value with the value argument. To initialize a Type parameter use resource_type(x) <- "Dataset".

### Details

The DataCite resourceType definition refers back to [dcm:type](#). The Type\$resourceTypeGeneral is set to Dataset, while the user can set a more specific Type\$resourceType value. (See examples.)

### Value

Returns the x object with the Type attribute as a list. The Type\$resourceTypeGeneral is set to Dataset.

### Examples

```
x <- data.frame()
resource_type(x) <- "Dataset"
resource_type(x)

y <- data.frame()
resource_type(y) <- "Census Data"
resource_type(y)
```

---

resource\_type\_general *Get/set the resourceTypeGeneral property of a (related) item*

---

### Description

The general type of a resource (file), see [DataCite 4.4 10.1 resourceTypeGeneral](#).

### Usage

```
resource_type_general(relitem)
resource_type_general(relitem) <- value
resource_type_general_allowed()
resource_type_general_verify(resourceTypeGeneral)
```

### Arguments

relitem	An object created by <a href="#">related_item_identifier</a> .
value	The general type of a resource (file), see <a href="#">DataCite 4.4 10.1 resourceTypeGeneral</a> .
resourceTypeGeneral	The general type of a resource or file. See <a href="#">resource_type_general</a> for allowed values and validation.

### Details

Use `resource_type_general_allowed` to get the allowed controlled list of resourcetypes from DataCite 4.4.  
[resource\\_type\\_general\\_verify](#) verifies if your property is among the allowed values in the DataCite 4.4 definition.

### Value

Get or set the resourceTypeGeneral property of a related item created with [related\\_item](#).

### See Also

[related\\_item](#)

### Examples

```
my_item <- related_item (Identifier = "https://zenodo.org/record/5703222#.YZYkm2DMLIU",
  Creator = person ("Daniel", "Antal", role = "aut"),
  Publisher = "Zenodo",
  PublicationYear = 2022,
  relatedIdentifierType = "DOI",
  relationType = "CompiledBy",
```

```
schemeURI = "URI",  
resourceTypeGeneral = "Dataset")
```

---

rights	<i>Get/set the Rights of the object.</i>
--------	--

---

## Description

Get/set the optional Rights property as an attribute to an R object.

## Usage

```
rights(x)  
  
rights(x, overwrite = TRUE) <- value
```

## Arguments

x	An R object, such as a data.frame, a tibble, or a data.table.
overwrite	If the Rights attribute should be overwritten. In case it is set to FALSE, it gives a message with the current Rights property instead of overwriting it. Defaults to TRUE when the attribute is set to value regardless of previous setting.
value	The Rights as a character set.

## Details

Rights corresponds to [dct:rights](#) and [datacite](#) Rights. Information about rights held in and over the resource. Typically, rights information includes a statement about various property rights associated with the resource, including intellectual property rights.

## Value

The Rights attribute as a character of length 1 is added to x.

## See Also

Other Reference metadata functions: [creator\(\)](#), [dataset\\_source\(\)](#), [description\(\)](#), [geolocation\(\)](#), [identifier\(\)](#), [language](#), [publication\\_year\(\)](#), [publisher\(\)](#), [size\(\)](#), [version\(\)](#)

## Examples

```
iris_dataset <- iris  
rights(iris_dataset) <- "CC-BY-SA"  
rights(iris_dataset)
```

---

size	<i>Add Size metadata to an object</i>
------	---------------------------------------

---

**Description**

Add the optional DataCite Size property as an attribute to an R object.

**Usage**

```
size(x)
```

**Arguments**

x                    An R object, such as a data.frame, a tibble, or a character vector.

**Details**

Size is an optional property in [DataCite 4.3](#). The object size is estimated with `[utils]{object.size}`.

**Value**

The estimated object size in memory is added as an attribute to x in SI kB and IEC KiB (legacy Kb) units, rounded to two decimals. Returns the x object.

**See Also**

Other Reference metadata functions: [creator\(\)](#), [dataset\\_source\(\)](#), [description\(\)](#), [geolocation\(\)](#), [identifier\(\)](#), [language](#), [publication\\_year\(\)](#), [publisher\(\)](#), [rights\(\)](#), [version\(\)](#)

**Examples**

```
iris_dataset <- size(iris)
attr(iris_dataset, "Size")
```

---

subject	<i>Get/Add subject(s) to a dataset</i>
---------	--

---

**Description**

Add one or more subject terms to the dataset's metadata.



**Usage**

```

subject(x)

subject(x, overwrite = FALSE) <- value

subject_create(
  term,
  subjectScheme = NA_character_,
  schemeURI = NA_character_,
  valueURI = NA_character_
)

```

**Arguments**

x	An R object
overwrite	Defaults to FALSE, in which case new <code>subject(x) &lt;- "Subject"</code> calls are binding further Subjects to the already set Subject properties.
value	Subject terms, or a Subject object created by <a href="#">subject_create</a> .
term	A term, or a character vector of multiple terms.
subjectScheme	The scheme to which the term corresponds. If there are multiple terms, provide the subjectScheme(s) in the same order. Optional.
schemeURI	The URI(s) of the subject identifier scheme. If there are multiple terms, provide the schemeURIs in the same order as the terms. Optional.
valueURI	The URI of the subject term. If there are multiple terms, provide the valueURIs in the same order as the terms. Optional.

**Details**

In the Dublin Core elements, `dct::subject` is defined Typically, the subject will be represented using keywords, key phrases, or classification codes. It is recommended as a best practice to use a controlled vocabulary.

In DataCite, subjects are defined as key phrases from a controlled library.

**Value**

The subjects as a data.frame of terms

**Examples**

```

x <- data.frame( geo = c("AL", "MK"),
  value = c(1,2))
my_subject <- subject_create (
  term = c("R (Computer program language)",
    "Questionnaires--Computer programs"),
  subjectScheme = rep("LC Subject Headings", 2),
  schemeURI = rep("http://id.loc.gov/authorities/subjects",2),
  valueURI = c("https://id.loc.gov/authorities/subjects/sh2002004407.html",
    "http://id.worldcat.org/fast/1085693/")
)

```

```
)  
subject(x) <- my_subject  
subject(x)  
  
y <- data.frame()  
subject(y) <- "R (Computer program language)"  
subject(y) <- "Questionnaires--Computer programs"  
subject(y)
```

---

version

*Get/set the version of the object.*

---

### Description

Get/set the optional Version property as an attribute to an R object.

### Usage

```
version(x)
```

```
version(x, overwrite = FALSE) <- value
```

### Arguments

x	An R object, such as a data.frame, a tibble, or a data.table.
overwrite	If the Version attribute should be overwritten. In case it is set to FALSE, it gives a message with the currentVersion property instead of overwriting it. Defaults to TRUE when the attribute is set to value regardless of previous setting.
value	The Version as a character set.

### Details

Version is an optional property in [DataCite 4.3](#). It is not part of the "core" Dublin Core terms, but ... [Dublin Core metadata terms](#).

### Value

The Version attribute as a character of length 1 is added to x.

### See Also

Other Reference metadata functions: [creator\(\)](#), [dataset\\_source\(\)](#), [description\(\)](#), [geolocation\(\)](#), [identifier\(\)](#), [language](#), [publication\\_year\(\)](#), [publisher\(\)](#), [rights\(\)](#), [size\(\)](#)

### Examples

```
iris_dataset <- iris  
version(iris_dataset) <- "1.0"  
version(iris_dataset)
```

# Index

- \* **Reference metadata functions**
  - creator, 4
  - dataset\_source, 13
  - description, 16
  - geolocation, 20
  - identifier, 21
  - language, 23
  - publication\_year, 25
  - publisher, 26
  - rights, 31
  - size, 32
  - version, 34
- \* **citation functions**
  - bibentry\_dataset, 3
- \* **dataset functions**
  - dataset, 8
  - dataset\_local\_id, 12
  - dataset\_uri, 15
- \* **datasets**
  - iris\_dataset, 22
- \* **export functions**
  - dataset\_export, 10
- \* **metadata functions**
  - datacite, 5
  - dublincore, 18
  - related\_item, 27
- [.dataset (dataset), 8
- as.data.frame (dataset), 8
- attributes, 3
- attributes\_measures, 2
- attributes\_measures<-  
(attributes\_measures), 2
- bibentry, 10
- bibentry\_dataset, 3, 10
- creator, 4, 9, 13, 16, 18, 21, 22, 24–26, 31,  
32, 34
- creator<- (creator), 4
- datacite, 5, 9, 18–20, 22, 28, 31
- datacite\_add (datacite), 5
- dataset, 8, 12, 15
- dataset\_export, 10
- dataset\_export\_csv, 11
- dataset\_export\_csv (dataset\_export), 10
- dataset\_local\_id, 10, 12, 15
- dataset\_source, 5, 13, 16, 19, 21, 22, 24–26,  
31, 32, 34
- dataset\_source<- (dataset\_source), 13
- dataset\_title, 6, 9, 14, 18, 27
- dataset\_title<- (dataset\_title), 14
- dataset\_title\_create, 14
- dataset\_title\_create (dataset\_title), 14
- dataset\_uri, 10, 12, 15
- description, 5, 13, 16, 19, 21, 22, 24–26, 31,  
32, 34
- description<- (description), 16
- dimensions, 17
- dimensions<- (dimensions), 17
- dublincore, 6, 7, 18, 27, 28
- dublincore\_add (dublincore), 18
- geolocation, 5, 7, 13, 16, 20, 22, 24–26, 31,  
32, 34
- geolocation<- (geolocation), 20
- identifier, 5, 9, 13, 16, 18, 21, 21, 24–26,  
31, 32, 34
- identifier<- (identifier), 21
- iris, 22, 23
- iris\_dataset, 22
- is.dataset (dataset), 8
- language, 5, 6, 13, 16, 19, 21, 22, 23, 25, 26,  
31, 32, 34
- language<- (language), 23
- measures, 24
- measures<- (measures), 24

person, [4](#)  
print.dataset (dataset), [8](#)  
publication\_year, [5](#), [6](#), [13](#), [16](#), [21](#), [22](#), [24](#),  
[25](#), [26](#), [28](#), [31](#), [32](#), [34](#)  
publication\_year<- (publication\_year),  
[25](#)  
publisher, [5](#), [6](#), [9](#), [13](#), [16](#), [19](#), [21](#), [22](#), [24](#), [25](#),  
[26](#), [28](#), [31](#), [32](#), [34](#)  
publisher<- (publisher), [26](#)  
  
related\_item, [7](#), [20](#), [27](#), [30](#)  
related\_item\_identifier, [30](#)  
resource\_type, [6](#), [29](#)  
resource\_type<- (resource\_type), [29](#)  
resource\_type\_general, [28](#), [30](#), [30](#)  
resource\_type\_general<-  
    (resource\_type\_general), [30](#)  
resource\_type\_general\_allowed  
    (resource\_type\_general), [30](#)  
resource\_type\_general\_verify, [30](#)  
resource\_type\_general\_verify  
    (resource\_type\_general), [30](#)  
rights, [5](#), [7](#), [13](#), [16](#), [19](#), [21](#), [22](#), [24–26](#), [31](#), [32](#),  
[34](#)  
rights<- (rights), [31](#)  
  
size, [5](#), [13](#), [16](#), [21](#), [22](#), [24–26](#), [31](#), [32](#), [34](#)  
subject, [6](#), [9](#), [19](#), [32](#)  
subject<- (subject), [32](#)  
subject\_create, [6](#), [9](#), [33](#)  
subject\_create (subject), [32](#)  
subset.dataset (dataset), [8](#)  
summary.dataset (dataset), [8](#)  
  
version, [5](#), [7](#), [13](#), [16](#), [21](#), [22](#), [24–26](#), [31](#), [32](#), [34](#)  
version<- (version), [34](#)  
  
write.csv, [11](#)