

Package ‘crunch’

August 4, 2015

Type Package

Title Crunch.io Data Tools

Description The Crunch.io service (<http://crunch.io/>) provides a cloud-based data store and analytic engine, as well as an intuitive web interface. Using this package, analysts can interact with and manipulate Crunch datasets from within R. Importantly, this allows technical researchers to collaborate naturally with team members, managers, and clients who prefer a point-and-click interface.

Version 1.4.2

Date 2015-08-04

Author Neal Richardson [aut, cre]

Maintainer Neal Richardson <neal@crunch.io>

URL <https://github.com/Crunch-io/rcrunch>

BugReports <https://github.com/Crunch-io/rcrunch/issues>

License LGPL (>= 3)

LazyLoad yes

Depends R (>= 3.0.0)

Imports httr (>= 1.0.0), jsonlite (>= 0.9.15), methods, curl

Suggests knitr, testthat

VignetteBuilder knitr

NeedsCompilation no

Repository CRAN

Date/Publication 2015-08-04 18:25:00

R topics documented:

<code>appendDataset</code>	3
<code>catalog-extract</code>	4
<code>catalog-length</code>	6

Categories-class	6
category-extract	7
crtabs	8
crunch	9
crunch-uni	9
CrunchDataset-class	10
CrunchVariable-class	10
cube-computing	11
cube-methods	11
dataset-extract	12
dataset-to-R	13
dataset-update	14
dataset-variables	15
delete	16
deleteDataset	16
describe	17
describe-catalog	18
describe-category	20
dichotomize	21
dim-dataset	22
expressions	23
getAccountUserCatalog	24
getTeams	25
grouped	25
hiddenVariables	26
hide	26
hideVariables	27
http-methods	28
index	28
is-na-categories	29
is.dataset	30
listDatasets	31
loadDataset	31
login	32
logout	32
makeArray	33
na-omit-categories	34
newDataset	34
newDatasetByColumn	35
newDatasetByCSV	35
newDatasetFromFile	36
ordering	37
refresh	37
restoreVersion	38
saveVersion	39
self	39
share	40
ShojiObject-class	40

show-crunch	41
Subvariables-class	41
subvars-extract	42
table	44
teams	45
tojson-crunch	46
toVariable	47
tuple-methods	47
type	49
unbind	49
updateDatasetList	50
var-categories	50
variable-extract	51
variable-to-R	52
variable-update	53
VariableCatalog-class	54
VariableOrder-class	54
VariableOrder-extract	55
VariableOrder-length	57
VariableOrder-slots	58
versions	59
weight	60
Index	61

appendDataset	<i>Append one Crunch dataset to another</i>
---------------	---

Description

Append one Crunch dataset to another

Usage

```
appendDataset(dataset1, dataset2, confirm = interactive(), cleanup = TRUE)
```

Arguments

dataset1	a CrunchDataset
dataset2	another CrunchDataset, or possibly a data.frame. If dataset2 is not a Crunch dataset, it will be uploaded as a new dataset before appending.
confirm	logical: should the user be forced to review and accept any automatically resolved conflicts between dataset1 and dataset2? Defaults to TRUE when occurring in an interactive R session, FALSE otherwise. If TRUE and not running interactively, the append operation will fail if there are any differences in meta-data.

cleanup logical: if the append operation fails or is aborted, should the intermediate batch created on dataset1 be deleted? Default is TRUE; FALSE may be useful if you want to review the append conflicts in the web application.

Value

A CrunchDataset with dataset2 appended to dataset1

catalog-extract	<i>Extract and modify subsets of Catalog-type objects</i>
-----------------	---

Description

Extract and modify subsets of Catalog-type objects

Usage

```
## S4 method for signature 'DatasetCatalog,character'
x[[i, j, ...]]

## S4 method for signature 'DatasetCatalog,ANY'
x[[i, j, ...]]

## S4 method for signature 'ShojiCatalog,character'
x[i, j, ..., drop = TRUE]

## S4 method for signature 'ShojiCatalog,numeric'
x[i, j, ..., drop = TRUE]

## S4 method for signature 'ShojiCatalog,logical'
x[i, j, ..., drop = TRUE]

## S4 method for signature 'ShojiCatalog,ANY'
x[i, j, ..., drop = TRUE]

## S4 method for signature 'ShojiCatalog,ANY'
x[[i, j, ...]]

## S4 method for signature 'TeamCatalog,character'
x[[i, j, ...]]

## S4 method for signature 'TeamCatalog,numeric'
x[[i, j, ...]]

## S4 replacement method for signature 'TeamCatalog,character,missing,list'
x[[i, j]] <- value
```

```
## S4 replacement method for signature 'TeamCatalog,character,missing,CrunchTeam'  
x[[i, j]] <- value  
  
## S4 method for signature 'TeamCatalog'  
x$name  
  
## S4 replacement method for signature 'MemberCatalog,ANY,missing,ANY'  
x[[i, j]] <- value  
  
## S4 replacement method for signature 'MemberCatalog,character,missing,`NULL`'  
x[[i, j]] <- value  
  
## S4 method for signature 'VariableCatalog,character'  
x[[i, j, ...]]  
  
## S4 method for signature 'VariableCatalog,ANY'  
x[[i, j, ...]]  
  
## S4 replacement method for signature 'VariableCatalog,character,missing,VariableTuple'  
x[[i, j]] <- value  
  
## S4 replacement method for signature 'VariableCatalog,character,missing,CrunchVariable'  
x[[i, j]] <-  
  value  
  
## S4 method for signature 'VariableCatalog,VariableOrder'  
x[i, j, ..., drop = TRUE]  
  
## S4 method for signature 'VariableCatalog,VariableGroup'  
x[i, j, ..., drop = TRUE]  
  
## S4 replacement method for signature 'VariableCatalog,character,missing,VariableCatalog'  
x[i, j] <-  
  value  
  
## S4 replacement method for signature  
## 'VariableCatalog,VariableOrder,missing,VariableCatalog'  
x[i, j] <-  
  value  
  
## S4 replacement method for signature  
## 'VariableCatalog,VariableGroup,missing,VariableCatalog'  
x[i, j] <-  
  value
```

Arguments

x a Catalog object

i	which catalog elements to extract
j	Invalid
...	additional arguments
drop	Invalid
value	For updating, an object of the appropriate class and size to insert
name	for \$, the same as i for []

Value

A subset of x if extracting, otherwise x duly modified

catalog-length	<i>Length of Catalog</i>
----------------	--------------------------

Description

Length of Catalog

Usage

```
## S4 method for signature 'ShojiCatalog'
length(x)
```

Arguments

x a Catalog

Value

Integer: the number of elements in the index list

Categories-class	<i>Categories in CategoricalVariables</i>
------------------	---

Description

CategoricalVariables, as well as the array types composed from Categoricals, contain Categories. Categories are a subclass of list that contains only Category objects. Category objects themselves subclass list and contain the following fields: "name", "id", "numeric_value", "missing", and optionally "selected".

Usage

```
## S4 method for signature 'Categories,ANY'
x[i, j, ..., drop = TRUE]

## S4 replacement method for signature 'Categories,ANY,ANY,ANY'
x[i, j, ...] <- value

## S4 method for signature 'Categories'
names(x)

## S4 method for signature 'Categories'
values(x)

## S4 method for signature 'Categories'
ids(x)

## S4 method for signature 'list'
ids(x)

## S4 replacement method for signature 'Categories'
names(x) <- value

## S4 replacement method for signature 'Categories'
values(x) <- value

## S4 replacement method for signature 'Categories'
ids(x) <- value
```

Arguments

x	For the attribute getters and setters, an object of class <code>Category</code> or <code>Categories</code>
i	For the <code>[</code> methods, just as with list extract methods
j	Invalid argument to <code>[</code> , but in the generic's signature
...	additional arguments to <code>[</code> , ignored
drop	Invalid argument to <code>[</code> , but in the generic's signature
value	For <code>[<-</code> , the replacement <code>Category</code> to insert

category-extract *Access Category fields directly*

Description

Don't do this. Instead, use the category setters.

Usage

```
## S4 method for signature 'Category'
x$name

## S4 replacement method for signature 'Category'
x$name <- value
```

Arguments

x	a Category
name	a field within x
value	a value for that field to update

Value

\$ returns the value of the desired field. Setter returns x duly modified.

See Also

[describe-category](#)

crtabs	<i>Crunch xtabs: Crosstab and otherwise aggregate variables in a Crunch Dataset</i>
--------	---

Description

Create a contingency table or other aggregation from cross-classifying variables in a CrunchDataset.

Usage

```
crtabs(formula, data, weight = crunch::weight(data), useNA = c("no",
  "ifany", "always"))
```

Arguments

formula	an object of class 'formula' object with the cross-classifying variables, separated by '+', on the right hand side. Compare to xtabs .
data	an object of class CrunchDataset
weight	a CrunchVariable that has been designated as a potential weight variable for data, or NULL for unweighted results. Default is the currently applied weight, weight (data).
useNA	whether to include missing values in tabular results. See table .

Value

an object of class CrunchCube

crunch

Crunch.io: instant, visual, collaborative data analysis

Description

Crunch.io provides a cloud-based data store and analytic engine. It has a **web client** for interactive data exploration and visualization. The `crunch` package for R allows analysts to interact with and manipulate Crunch datasets from within R. Importantly, this allows technical researchers to collaborate naturally with team members, managers, and clients who prefer a point-and-click interface: because all connect to the same dataset in the cloud, there is no need to email files back and forth continually to share results.

See Also

To learn more about using the package, see `vignette("getting-started", package="crunch")`. To sign up for a Crunch.io account, visit <https://beta.crunch.io/>.

crunch-uni

Univariate statistics on Crunch objects

Description

Univariate statistics on Crunch objects

Usage

```
## S4 method for signature 'CrunchVariable'  
mean(x, ...)  
  
## S4 method for signature 'CrunchVariable'  
sd(x, na.rm = FALSE)  
  
## S4 method for signature 'CrunchVariable'  
median(x, na.rm = FALSE)  
  
## S4 method for signature 'CrunchVariable'  
min(x, na.rm)  
  
## S4 method for signature 'CrunchVariable'  
max(x, na.rm)  
  
## S4 method for signature 'NumericVariable'  
mean(x, ...)  
  
## S4 method for signature 'NumericVariable'
```

```

median(x, na.rm = FALSE)

## S4 method for signature 'NumericVariable'
sd(x, na.rm = FALSE)

## S4 method for signature 'NumericVariable'
min(x, na.rm = FALSE)

## S4 method for signature 'NumericVariable'
max(x, na.rm = FALSE)

## S4 method for signature 'DatetimeVariable'
min(x, na.rm = FALSE)

## S4 method for signature 'DatetimeVariable'
max(x, na.rm = FALSE)

```

Arguments

x	a NumericVariable, or for min and max, possibly a DatetimeVariable
...	additional arguments to mean
na.rm	logical: exclude missings?

See Also

[mean](#) [sd](#) [median](#) [min](#) [max](#)

CrunchDataset-class *Crunch Datasets*

Description

Crunch Datasets

CrunchVariable-class *Variables in Crunch*

Description

Variables are S4 objects. All inherit from the base class `CrunchVariable`.

Slots

`readonly` logical: should changes made to this variable object locally be persisted on the server?
 Default is FALSE

`tuple` An object of class `VariableTuple`. These contain attributes, such as name and description, that are found in the index of the [VariableCatalog](#)

Description

Crunch.io supports more complex data types than base R does, such as multiple response and array types. If you want to compute margin or proportion tables on an aggregation of these variable types, special methods are required. These functions provide an interface like [margin.table](#) and [prop.table](#) for the `CrunchCube` object, handling those special data types.

Usage

```
## S4 method for signature 'CrunchCube'
margin.table(x, margin = NULL)

## S4 method for signature 'CrunchCube'
prop.table(x, margin = NULL)

## S4 method for signature 'CrunchCube'
round(x, digits = 0)
```

Arguments

<code>x</code>	a <code>CrunchCube</code>
<code>margin</code>	index, or vector of indices to generate margin for. See prop.table
<code>digits</code>	see round

Value

The appropriate `margin.table` or `prop.table`.

See Also

[margin.table](#) [prop.table](#)

Description

These methods provide an array-like interface to the `CrunchCube` object.

Usage

```
## S4 method for signature 'CubeDims'
dimnames(x)

## S4 method for signature 'CubeDims'
dim(x)

## S4 method for signature 'CubeDims'
is.na(x)
```

Arguments

x a CrunchCube or its CubeDims component.

Value

Generally, the same shape of result that each of these functions return when applied to an array object.

See Also

[cube-computing](#)

dataset-extract	<i>Subset datasets and extract variables</i>
-----------------	--

Description

Subset datasets and extract variables

Usage

```
## S4 method for signature 'CrunchDataset,ANY'
x[i, j, ..., drop = TRUE]

## S4 method for signature 'CrunchDataset,character'
x[i, j, ..., drop = TRUE]

## S4 method for signature 'CrunchDataset,missing'
x[i, j, ..., drop = FALSE]

## S4 method for signature 'CrunchDataset,ANY'
x[[i, ..., drop = FALSE]]

## S4 method for signature 'CrunchDataset,character'
x[[i, ..., drop = FALSE]]

## S4 method for signature 'CrunchDataset'
x$name
```

Arguments

x	a CrunchDataset
i	if character, identifies variables to extract based on their aliases (by default, i.e. when x's useAlias is TRUE); if numeric or logical, extracts variables accordingly. Note that this is the as.list extraction, columns of the dataset rather than rows.
j	column extraction, as described above
...	additional arguments
drop	logical: automatically simplify a 1-column Dataset to a Variable? Default is FALSE, and the TRUE option is in fact not implemented.
name	like i but for \$

Value

[yields a Dataset; [[] and \$ return a Variable

dataset-to-R	<i>as.data.frame method for CrunchDataset</i>
--------------	---

Description

This method is defined principally so that you can use a CrunchDataset as a data argument to other R functions (such as `lm`). Unless you give it the `force==TRUE` argument, this function does not in fact return a `data.frame`: it returns an object with an interface like a `data.frame`, such that you get R vectors when you access its columns (unlike a `CrunchDataset`, which returns `CrunchVariable` objects). This allows modeling functions that require select columns of a dataset to retrieve only those variables from the remote server, rather than pulling the entire dataset into local memory.

Usage

```
## S3 method for class 'CrunchDataset'
as.data.frame(x, row.names = NULL, optional = FALSE,
  force = FALSE, ...)

## S3 method for class 'CrunchDataFrame'
as.data.frame(x, row.names = NULL,
  optional = FALSE, ...)
```

Arguments

x	a CrunchDataset
row.names	part of <code>as.data.frame</code> signature. Ignored.
optional	part of <code>as.data.frame</code> signature. Ignored.
force	logical: actually coerce the dataset to <code>data.frame</code> , or leave the columns as unevaluated promises. Default is FALSE.
...	additional arguments passed to <code>as.data.frame.default</code>

Value

an object of class `CrunchDataFrame` unless `force`, in which case the return is a `data.frame`.

dataset-update	<i>Update a variable or variables in a dataset</i>
----------------	--

Description

Update a variable or variables in a dataset

Usage

```
## S4 replacement method for signature 'CrunchDataset,character,missing,CrunchVariable'
x[[i]] <- value

## S4 replacement method for signature 'CrunchDataset,ANY,missing,CrunchVariable'
x[[i]] <- value

## S4 replacement method for signature 'CrunchDataset,character,missing,ANY'
x[[i]] <- value

## S4 replacement method for signature 'CrunchDataset,character,missing,CrunchExpr'
x[[i]] <- value

## S4 replacement method for signature 'CrunchDataset,character,missing,CrunchLogicalExpr'
x[[i]] <- value

## S4 replacement method for signature 'CrunchDataset,ANY,ANY,ANY'
x[[i]] <- value

## S4 replacement method for signature 'CrunchDataset'
x$name <- value

## S4 replacement method for signature 'CrunchDataset,ANY,missing,list'
x[i, j] <- value

## S4 replacement method for signature 'CrunchDataset,CrunchExpr,ANY,ANY'
x[i, j] <- value
```

Arguments

<code>x</code>	a <code>CrunchDataset</code>
<code>i</code>	For <code>[</code> , a <code>CrunchLogicalExpr</code> , numeric, or logical vector defining a subset of the rows of <code>x</code> . For <code>[[</code> , see <code>j</code> for the <code>as.list</code> column subsetting.
<code>value</code>	replacement values to insert. These can be <code>crunchExprs</code> or R vectors of the corresponding type

name	like j but for \$
j	if character, identifies variables to extract based on their aliases (by default, i.e. when x's useAlias is TRUE); if numeric or logical, extracts variables accordingly. Note that this is the as.list extraction, columns of the dataset rather than rows.

Value

x, modified.

dataset-variables *Access a Dataset's Variables Catalog*

Description

Datasets contain collections of variables. For a few purposes, such as editing variables' metadata, it is helpful to access these variable catalogs more directly.

Usage

```
## S4 method for signature 'CrunchDataset'
variables(x)

## S4 replacement method for signature 'CrunchDataset,VariableCatalog'
variables(x) <- value

## S4 method for signature 'CrunchDataset'
allVariables(x)

## S4 replacement method for signature 'CrunchDataset,VariableCatalog'
allVariables(x) <- value
```

Arguments

x	a Dataset
value	For the setters, a VariableCatalog to assign.

Details

variables gives just the active variables in the dataset, while allVariables, as the name suggests, yields all variables, including hidden variables.

Value

Getters return VariableCatalog; setters return x duly modified.

delete	<i>Delete a Crunch object from the server</i>
--------	---

Description

These methods delete entities, notably Datasets and Variables within them, from the server. This action is permanent and cannot be undone, so it should not be done lightly. Consider instead using [archive](#) for datasets and [hide](#) for variables

Usage

```
## S4 method for signature 'CrunchDataset'
delete(x, confirm = interactive() | is.readonly(x),
      ...)

## S4 method for signature 'ShojiObject'
delete(x, ...)

## S4 method for signature 'ANY'
delete(x, ...)

## S4 method for signature 'CrunchTeam'
delete(x, confirm = interactive(), ...)

## S4 method for signature 'CategoricalArrayVariable'
delete(x, ...)
```

Arguments

x	a Crunch object
confirm	logical: should the user be asked to confirm deletion. Option available for datasets and teams only. Default is TRUE if in an interactive session.
...	additional arguments, in the generic

See Also

[hide deleteDataset](#)

deleteDataset	<i>Delete a dataset from the dataset list</i>
---------------	---

Description

This function lets you delete a dataset without first loading it. If you have a dataset that somehow is corrupted and won't load, you can delete it this way.

Usage

```
deleteDataset(x, ...)
```

Arguments

`x` The name (character) of a dataset, its (numeric) position in the return of `listDatasets`, or an object of class `CrunchDataset`. `x` can only be of length 1—this function is not vectorized (for your protection).

`...` additional parameters (such as `confirm`) passed to `delete`

Details

The function also works on `CrunchDataset` objects, just like `delete`, which may be useful if you have loaded another package that masks the `delete` method.

Value

(Invisibly) the API response from deleting the dataset

See Also

[delete](#)

describe

Name, alias, and description for Crunch objects

Description

Name, alias, and description for Crunch objects

Usage

```
## S4 method for signature 'CrunchDataset'
name(x)

## S4 replacement method for signature 'CrunchDataset'
name(x) <- value

## S4 method for signature 'CrunchDataset'
description(x)

## S4 replacement method for signature 'CrunchDataset'
description(x) <- value

## S4 method for signature 'CrunchTeam'
name(x)
```

```
## S4 method for signature 'CrunchVariable'
name(x)

## S4 replacement method for signature 'CrunchVariable'
name(x) <- value

## S4 method for signature 'CrunchVariable'
description(x)

## S4 replacement method for signature 'CrunchVariable'
description(x) <- value

## S4 method for signature 'CrunchVariable'
alias(object)

## S4 replacement method for signature 'CrunchVariable'
alias(x) <- value
```

Arguments

x	a Dataset or Variable.
value	For the setters, a length-1 character vector to assign
object	Same as x but for the alias method, in order to match the generic from another package. Note that alias is only defined for Variables.

Value

Getters return the character object in the specified slot; setters return x duly modified.

See Also

[Categories describe-catalog](#)

describe-catalog

Get and set names, aliases on Catalog-type objects

Description

These methods let you get and set names and aliases for variables in a Dataset's catalog, or within [Subvariables](#) in an array variable. They work like the base R names methods.

Usage

```
## S4 method for signature 'DatasetCatalog'
names(x)

## S4 method for signature 'CrunchDataset'
```

```
names(x)

## S4 method for signature 'PermissionCatalog'
emails(x)

## S4 method for signature 'Subvariables'
names(x)

## S4 replacement method for signature 'Subvariables'
names(x) <- value

## S4 method for signature 'Subvariables'
aliases(x)

## S4 replacement method for signature 'Subvariables'
aliases(x) <- value

## S4 method for signature 'TeamCatalog'
names(x)

## S4 method for signature 'MemberCatalog'
names(x)

## S4 method for signature 'UserCatalog'
names(x)

## S4 method for signature 'UserCatalog'
emails(x)

## S4 method for signature 'VariableCatalog'
names(x)

## S4 replacement method for signature 'VariableCatalog'
names(x) <- value

## S4 method for signature 'VariableCatalog'
aliases(x)

## S4 replacement method for signature 'VariableCatalog'
aliases(x) <- value

## S4 method for signature 'VariableCatalog'
descriptions(x)

## S4 replacement method for signature 'VariableCatalog'
descriptions(x) <- value

## S4 method for signature 'VariableCatalog'
```

```

types(x)

## S4 method for signature 'VersionCatalog'
names(x)

## S4 method for signature 'VersionCatalog'
descriptions(x)

## S4 method for signature 'VersionCatalog'
timestamps(x)

```

Arguments

`x` a VariableCatalog, Subvariables, or similar object
`value` For the setters, an appropriate-length character vector to assign

Details

Note that the `names` method on a Dataset returns the aliases of its variables by default. See the vignette on variables for more information.

Value

Getters return the character object in the specified slot; setters return `x` duly modified.

See Also

[Subvariables Categories names](#) vignette("variables", package="crunch")

describe-category *Category attributes*

Description

Category attributes

Usage

```

## S4 method for signature 'Category'
name(x)

## S4 replacement method for signature 'Category'
name(x) <- value

## S4 method for signature 'Category'
value(x)

```

```
## S4 replacement method for signature 'Category'
value(x) <- value

## S4 method for signature 'Category'
id(x)

## S4 method for signature 'list'
id(x)

## S4 method for signature 'Category'
is.selected(x)
```

Arguments

x	a Category
value	For the setters, an appropriate value to set

Value

name returns character; value and id return numeric; value but not id may be NA; is.selected returns logical indicating whether this Category is a "selected" dichotomy. Setters return x duly modified.

See Also

[Categories dichotomize](#)

dichotomize	<i>Indicate how categories represent a dichotomized value</i>
-------------	---

Description

Multiple Response variables are essentially Categorical Arrays that have had a category or categories indicated as the "selected" value. These methods let you set that state.

Usage

```
## S4 method for signature 'Categories'
is.dichotomized(x)

## S4 method for signature 'Categories,numeric'
dichotomize(x, i)

## S4 method for signature 'Categories,logical'
dichotomize(x, i)

## S4 method for signature 'Categories,character'
```

```
dichotomize(x, i)

## S4 method for signature 'Categories'
undichotomize(x)

## S4 method for signature 'CategoricalVariable,ANY'
dichotomize(x, i)

## S4 method for signature 'CategoricalArrayVariable,ANY'
dichotomize(x, i)

## S4 method for signature 'CategoricalVariable'
undichotomize(x)

## S4 method for signature 'CategoricalArrayVariable'
undichotomize(x)
```

Arguments

x	Categories or a Variable subclass that has Categories
i	For the dichotomize methods, the numeric or logical indices of the categories to mark as "selected", or if character, the Category "names". Note that unlike some other categorical variable methods, numeric indices are positional, not with reference to category ids.

Details

dichotomize lets you specify which categories are "selected", while undichotomize strips that selection information. Dichotomize converts a Categorical Array to a Multiple Response, and undichotomize converts back.

Value

Categories or the Variable, (un)dichotomized accordingly

See Also

[describe-category](#)

dim-dataset

Dataset dimensions

Description

Dataset dimensions

Usage

```
## S4 method for signature 'CrunchDataset'
dim(x)
```

Arguments

x a Dataset

Value

integer vector of length 2, indicating the number of rows and non-hidden variables in the dataset. Array subvariables are excluded from the column count.

See Also

[dim](#)

expressions

Construct Crunch Expressions

Description

Crunch Expressions, i.e. `CrunchExpr` and `CrunchLogicalExpr`, encapsulate derivations of Crunch variables, which are only evaluated when passed to a function like `as.vector`. They allow you to compose functional expressions of variables and evaluate them against the server only when appropriate.

Usage

```
## S4 method for signature 'CrunchExpr'
as.vector(x, mode = "any")

## S4 method for signature 'CrunchExpr'
!x

## S4 method for signature 'CategoricalVariable,character'
x %in% table

## S4 method for signature 'CategoricalVariable,factor'
x %in% table

## S4 method for signature 'TextVariable,character'
x %in% table

## S4 method for signature 'NumericVariable,numeric'
x %in% table
```

```

## S4 method for signature 'DatetimeVariable,Date'
x %in% table

## S4 method for signature 'DatetimeVariable,POSIXt'
x %in% table

## S4 method for signature 'CategoricalVariable,numeric'
x %in% table

## S4 method for signature 'CrunchVariable'
is.na(x)

bin(x)

rollup(x, resolution = rollupResolution(x))

```

Arguments

x	an input
mode	For <code>as.vector</code> . Ignored.
table	For <code>%in%</code> . See match
resolution	For <code>rollup</code> . Either NULL or a character in <code>c("Y", "Q", "M", "W", "D", "h", "m", "s", "ms")</code> indicating the unit of time at which a <code>Datetime</code> variable should be aggregated. If NULL, the server will determine an appropriate resolution based on the range of the data.

Value

Most functions return a `CrunchExpr` or `CrunchLogicalExpr`. `as.vector` returns an R vector.

`getAccountUserCatalog` *Find all users on your account*

Description

Find all users on your account

Usage

```
getAccountUserCatalog(x = rootURL("users", getAccount()))
```

Arguments

x	URL of the user catalog. Default is the right thing; you shouldn't specify one
---	--

Value

a `UserCatalog`

getTeams	<i>Retrive all teams you're a member of</i>
----------	---

Description

Retrive all teams you're a member of

Usage

```
getTeams()
```

Value

A TeamCatalog. Extract an individual team by name. Create a team by assigning in with a new name.

See Also

[teams](#)

grouped	<i>Get un(grouped) VariableGroups</i>
---------	---------------------------------------

Description

"ungrouped" is a magic VariableGroup that contains all variables not found in groups at a given level of nesting.

Usage

```
grouped(var.order)
```

```
ungrouped(var.order)
```

Arguments

var.order an object of class VariableOrder or VariableGroup

Value

For grouped(), a VariableOrder/Group, respectively, with "ungrouped" omitted. For ungrouped(), a VariableGroup.

See Also

[VariableOrder](#)

hiddenVariables	<i>Show the names of hidden variables within the dataset</i>
-----------------	--

Description

Show the names of hidden variables within the dataset

Usage

```
hiddenVariables(dataset, key = "name")
```

Arguments

dataset	the Dataset
key	the Variable attribute to return. Default is "alias"

Value

a vector of the names of Variables marked as hidden.

hide	<i>Hide and Unhide Variables</i>
------	----------------------------------

Description

Hide and Unhide Variables

Usage

```
## S4 method for signature 'CrunchVariable'
hide(x)
```

```
## S4 method for signature 'VariableCatalog'
hide(x)
```

```
## S4 method for signature 'CrunchVariable'
unhide(x)
```

```
## S4 method for signature 'VariableCatalog'
unhide(x)
```

Arguments

x	a Variable or subset of a VariableCatalog to hide or unhide
---	---

Value

(invisibly) the Variable or VariableCatalog, hidden or unhidden

hideVariables	<i>Hide and Unhide Variables Within a Dataset</i>
---------------	---

Description

Hide and Unhide Variables Within a Dataset

Usage

```
hideVariables(dataset, variables = NULL, pattern = NULL,  
              key = namekey(dataset), ...)
```

```
hiddenVariables(x) <- value
```

```
unhideVariables(dataset, variables = NULL, pattern = NULL,  
                key = namekey(dataset), ...)
```

Arguments

dataset	the Dataset to modify
variables	names or indices of variables to (un)hide
pattern	optional regular expression to identify Variables to (un)hide
key	the Variable attribute to grep with the pattern. Default is "alias"
...	optional additional arguments to grep
x	same as dataset, for 'hiddenVariables<-'
value	same as variables, for 'hiddenVariables<-'

Value

(invisibly) dataset with the specified variables (un)hidden

See Also

[hide](#)

http-methods

HTTP methods for communicating with the Crunch API

Description

HTTP methods for communicating with the Crunch API

Usage

```
crGET(...)
```

```
crPUT(...)
```

```
crPATCH(...)
```

```
crPOST(...)
```

```
crDELETE(...)
```

Arguments

... see [crunchAPI](#) for details. `url` is the first named argument and is required; `body` is also required for PUT, PATCH, and POST.

Value

Depends on the response status of the HTTP request and any custom handlers.

index

Get the body of a Catalog

Description

The core of Catalog data is in its "index". These methods get and set that slot.

Usage

```
## S4 method for signature 'ShojiCatalog'  
index(x)
```

```
## S4 replacement method for signature 'ShojiCatalog'  
index(x) <- value
```

Arguments

x a Catalog (VariableCatalog, Subvariables, or similar object)
 value For the setters, an appropriate-length list to assign

Value

Getters return the list object in the "index" slot; setters return x duly modified.

is-na-categories *is.na for Categories*

Description

is.na for Categories

Usage

```
## S4 method for signature 'Categories'
is.na(x)

## S4 replacement method for signature 'Categories,character'
is.na(x) <- value

## S4 replacement method for signature 'Categories,logical'
is.na(x) <- value

## S4 method for signature 'Category'
is.na(x)

## S4 replacement method for signature 'Category,logical'
is.na(x) <- value
```

Arguments

x Categories or a single Category
 value To change the missingness of categories, supply either (1) a logical vector of equal length of the categories (or length 1 for the Category method), or (2) the names of the categories to mark as missing. If supplying the latter, any categories already indicated as missing will remain missing.

Value

Getters return logical, a named vector in the case of the Categories method; setters return x duly modified.

is.dataset

Is it?

Description

Is it?

Usage

is.dataset(x)

is.shoji(x)

is.variable(x)

is.Numeric(x)

is.Categorical(x)

is.Text(x)

is.Datetime(x)

is.Multiple(x)

is.MR(x)

is.MultipleResponse(x)

is.CA(x)

is.Array(x)

is.CategoricalArray(x)

Arguments

x an object

Value

logical

listDatasets	<i>Show the names of all Crunch datasets</i>
--------------	--

Description

Show the names of all Crunch datasets

Usage

```
listDatasets(kind = c("active", "all", "archived"), refresh = FALSE)
```

Arguments

kind	character specifying whether to look in active, archived, or all datasets.
refresh	logical: should the function check the Crunch API for new datasets? Default is FALSE.

Value

Character vector of dataset names, each of which would be a valid input for [loadDataset](#)

loadDataset	<i>Load a Crunch Dataset</i>
-------------	------------------------------

Description

Load a Crunch Dataset

Usage

```
loadDataset(dataset.name, kind = c("active", "all", "archived"),  
  useAlias = default.useAlias())
```

Arguments

dataset.name	character, the name of a Crunch dataset you have access to.
kind	character specifying whether to look in active, archived, or all datasets.
useAlias	logical whether variable alias or name should be used as R variable names when the dataset is returned. Default is TRUE, meaning alias. They're more computer friendly.

Value

An object of class CrunchDataset

login	<i>Authenticate with the Crunch API</i>
-------	---

Description

Note that you can store your Crunch account info in your `.Rprofile` under `"crunch.email"` and `"crunch.pw"` for convenience. If you do so, you can simply `login()` to authenticate. For running batch jobs, this could be particularly useful. However, be warned that storing your password in a plain text file such as `.Rprofile` is a security risk (though perhaps less so than in every `.R` script you write), and we cannot officially recommend that you do so.

Usage

```
login(email = getOption("crunch.email"), password = getOption("crunch.pw"),
      ...)
```

Arguments

email	the email address associated with the user's Crunch account
password	the password associated with the user's Crunch account
...	additional parameters passed in the authentication. Not currently supported by the Crunch API.

Details

If a password is not supplied (or, if no arguments are supplied and only the `crunch.email` is specified in `.Rprofile`), and you are in an interactive session, you will be prompted to enter your password. At present, this is the most secure practice as your password is not stored locally.

logout	<i>Kill the active Crunch session</i>
--------	---------------------------------------

Description

Kill the active Crunch session

Usage

```
logout()
```

makeArray	<i>Make a Categorical Array or Multiple Response variable</i>
-----------	---

Description

Make a Categorical Array or Multiple Response variable

Usage

```
makeArray(list_of_variables, dataset = NULL, pattern = NULL,
          key = namekey(dataset), name, ...)
```

```
makeMR(list_of_variables, dataset = NULL, pattern = NULL,
        key = namekey(dataset), name, selections, ...)
```

Arguments

list_of_variables	a list of Variable objects to bind together, or a Dataset object containing only the Variables to bind (as in from subsetting a Dataset), or values (e.g. names) of variables corresponding to key. If omitted, must supply dataset and pattern. If specifying values, must include dataset.
dataset	the Crunch Dataset to which the variables in list_of_variables belong, or in which to search for variables based on pattern. If omitted, list_of_variables must exist and all Variables in the list must belong to the same Dataset
pattern	An optional regular expression to search for variables to bind within dataset.
key	character, the name of the Variable field in which to search with pattern. Default is 'alias'.
name	character, the name that the new Categorical Array variable should have. Required.
...	Optional additional attributes to set on the new variable.
selections	character, for makeMR, the names of the categories to mark as the dichotomous selections. Required for makeMR; ignored in makeArray.

Value

The object of class CategoricalArrayVariable or MultipleResponseVariable corresponding to the just-created variable on the server.

na.omit-categories	<i>Omit missing categories</i>
--------------------	--------------------------------

Description

Omit missing categories

Usage

```
## S4 method for signature 'Categories'
na.omit(object, ...)
```

Arguments

object	Categories
...	additional arguments, ignored

Value

object with any categories that have missing: TRUE excluded

newDataset	<i>Upload a data.frame to Crunch to make a new dataset</i>
------------	--

Description

Upload a data.frame to Crunch to make a new dataset

Usage

```
newDataset(x, name = as.character(substitute(x)),
  useAlias = default.useAlias(), ...)
```

Arguments

x	a data.frame or other rectangular R object
name	character, the name to give the new Crunch dataset. Default is the name of the R object passed in x
useAlias	logical whether variable alias or name should be used as R variable names when the dataset is returned. Default is TRUE, meaning alias.
...	additional arguments passed to createDataset

Value

If successful, an object of class CrunchDataset.

newDatasetByColumn *Upload a data.frame column-by-column to make a new dataset*

Description

Use this version if you have lots of variables, under 1M rows, perhaps backed by ff or other memory-mapped files, and time to kill.

Usage

```
newDatasetByColumn(x, name = as.character(substitute(x)),  
  useAlias = default.useAlias(), ...)
```

Arguments

x	a data.frame or other rectangular R object
name	character, the name to give the new Crunch dataset. Default is the name of the R object passed in x
useAlias	logical whether variable alias or name should be used as R variable names when the dataset is returned. Default is TRUE, meaning alias.
...	additional arguments passed to createDataset

Value

If successful, an object of class CrunchDataset.

See Also

[newDataset](#) [newDatasetByCSV](#)

newDatasetByCSV *Upload a data.frame to Crunch to make a new dataset*

Description

This function uses the CSV+JSON import format, which is faster and more effective for certain dataset sizes and shapes than [newDatasetByColumn](#).

Usage

```
newDatasetByCSV(x, name = as.character(substitute(x)),  
  useAlias = default.useAlias(), ...)
```

Arguments

x	a data.frame or other rectangular R object
name	character, the name to give the new Crunch dataset. Default is the name of the R object passed in x
useAlias	logical whether variable alias or name should be used as R variable names when the dataset is returned. Default is TRUE, meaning alias.
...	additional arguments passed to createDataset

Value

If successful, an object of class CrunchDataset.

See Also

[newDataset](#) [newDatasetByColumn](#)

newDatasetFromFile *Upload a file to Crunch to make a new dataset*

Description

Use this import method if you have an SPSS data file. Reading such a file into R as a data.frame will result in lost metadata. You can just send it directly to Crunch and let the server process it.

Usage

```
newDatasetFromFile(file, name = basename(file),
  useAlias = default.useAlias(), ...)
```

Arguments

file	character, the path to a file to upload. This should either be a .csv or .sav (SPSS) file.
name	character, the name to give the new Crunch dataset. Default is the file name
useAlias	logical whether variable alias or name should be used as R variable names when the dataset is returned. Default is TRUE, meaning alias. They're more computer friendly.
...	additional arguments passed to createDataset

Value

On success, an object of class CrunchDataset.

ordering	<i>Get and set VariableOrder</i>
----------	----------------------------------

Description

The ordering methods allow you to get and set a [VariableOrder](#) on a [CrunchDataset](#) or on the [VariableCatalog](#) that the dataset contains.

Usage

```
## S4 method for signature 'CrunchDataset'
ordering(x)

## S4 replacement method for signature 'CrunchDataset'
ordering(x) <- value

## S4 method for signature 'VariableCatalog'
ordering(x)

## S4 replacement method for signature 'VariableCatalog'
ordering(x) <- value
```

Arguments

x	a VariableCatalog or CrunchDataset
value	a valid VariableOrder object

Value

ordering returns a VariableOrder object, while ordering<- sets the VariableOrder in value on x

refresh	<i>Get a fresh copy from the server</i>
---------	---

Description

Crunch objects usually keep themselves in sync with the server when you manipulate them, but sometimes they can drift. Maybe someone else has modified the dataset you're working on, or maybe you have modified a variable outside of the context of its dataset. refresh() allows you to get back in sync.

Usage

```
## S4 method for signature 'CrunchDataset'
refresh(x)

## S4 method for signature 'ShojiObject'
refresh(x)

## S4 method for signature 'CrunchVariable'
refresh(x)
```

Arguments

x pretty much any Crunch object

Value

a new version of x

restoreVersion	<i>Restore a dataset to a previously saved version</i>
----------------	--

Description

Restore a dataset to a previously saved version

Usage

```
restoreVersion(dataset, version)
```

Arguments

dataset a CrunchDataset

version either the name ("description") of the version to restore to or the integer index of the version, as given by versions(dataset)

Value

dataset, rolled back to version.

See Also

[versions saveVersion](#)

saveVersion	<i>Create a new saved version</i>
-------------	-----------------------------------

Description

Create a new saved version

Usage

```
saveVersion(dataset, description = NULL)
```

Arguments

dataset	a CrunchDataset
description	character name to give the saved version, as in a commit message. You are encouraged, though not strictly required, to give versions unique descriptions.

Value

invisibly, the URL of the newly created version

See Also

[versions restoreVersion](#)

self	<i>Get the URL of this object</i>
------	-----------------------------------

Description

Get the URL of this object

Usage

```
## S4 method for signature 'ShojiObject'
self(x)
```

Arguments

x	a Crunch object
---	-----------------

Value

the URL for x

share	<i>Share a dataset</i>
-------	------------------------

Description

Share a dataset

Usage

```
share(dataset, email, edit = FALSE, notify = TRUE)
```

Arguments

dataset	a CrunchDataset
email	character: email address(es) of the users with whom to share the dataset. If there is no Crunch user associated with an email, an invitation will be sent.
edit	logical: should the specified user be given edit privileges on the dataset? Default is FALSE. edit can be a single value or, if inviting multiple users, a vector of logical values of equal length of the number of emails given. Currently, only one user may have edit=TRUE at any given time, so there may not be more than one TRUE value in the logical vector.
notify	logical: should users who are getting new privileges on this dataset be sent an email informing them of this fact? Default is TRUE.

Value

Invisibly, the dataset.

ShojiObject-class	<i>Mix-in class for multiple inheritance of variables and datasets.</i>
-------------------	---

Description

Exists for common methods in interacting with Crunch API only. Has no Extract methods declared so as not to conflict with the vector/list/data.frame methods jointly inherited in CrunchVariable and CrunchDataset.

show-crunch	<i>Show methods for Crunch objects</i>
-------------	--

Description

Show methods for Crunch objects

Usage

```
## S4 method for signature 'ShojiObject'  
show(object)
```

```
## S4 method for signature 'Category'  
show(object)
```

```
## S4 method for signature 'Categories'  
show(object)
```

```
## S4 method for signature 'CrunchCube'  
show(object)
```

Arguments

object the object

Value

invisibly

See Also

[show](#)

Subvariables-class	<i>Subvariables in Array Variables</i>
--------------------	--

Description

Multiple-response and categorical-array variables contain a set of subvariables within them. The Subvariables class encapsulates them.

Usage

```
## S4 method for signature 'CategoricalArrayVariable'
subvariables(x)

## S4 replacement method for signature 'CategoricalArrayVariable,ANY'
subvariables(x) <- value

## S4 replacement method for signature 'CategoricalArrayVariable,Subvariables'
subvariables(x) <- value
```

Arguments

x	A Variable or Subvariables object
value	For the setters, the appropriate values to set

Details

Subvariables can be accessed from array variables (including multiple response) with the `subvariables` method. They can be assigned back with the `subvariables<-` setter, but there are limitations to what is supported. Specifically, you can reorder subvariables, but you cannot add or remove subvariables by `subvariables<-` assignment.

Subvariables have a `names` attribute that can be accessed, showing the display names of the subvariables. These can be set with the `names<-` method.

Finally, subvariables can be accessed as regular (categorical) variables with the `$` and `[[` extract methods.

See the vignette on array variables for further details and examples.

See Also

[subvars-extract describe-catalog vignette\("array-variables", package="crunch"\)](#)

subvars-extract

Extract and modify subsets of subvariables

Description

Extract and modify subsets of subvariables

Usage

```
## S4 method for signature 'Subvariables,character'
x[[i, j, ...]]

## S4 method for signature 'Subvariables,ANY'
x[[i, j, ...]]
```

```

## S4 method for signature 'Subvariables'
x$name

## S4 method for signature 'Subvariables,character'
x[i, j, ..., drop = TRUE]

## S4 replacement method for signature 'Subvariables,character,missing,CrunchVariable'
x[[i]] <- value

## S4 replacement method for signature 'Subvariables,ANY,missing,CrunchVariable'
x[[i]] <- value

## S4 replacement method for signature 'Subvariables,ANY,missing,`NULL`'
x[[i]] <- value

## S4 replacement method for signature 'Subvariables,ANY,missing,ANY'
x[[i]] <- value

## S4 replacement method for signature 'Subvariables,character,missing,Subvariables'
x[i] <- value

## S4 replacement method for signature 'Subvariables,ANY,missing,Subvariables'
x[i] <- value

## S4 replacement method for signature 'Subvariables,ANY,missing,ANY'
x[i] <- value

## S4 method for signature 'CategoricalArrayVariable,ANY'
x[i, j, ..., drop = TRUE]

## S4 method for signature 'CategoricalArrayVariable,numeric'
x[i, j, ..., drop = TRUE]

## S4 method for signature 'CategoricalArrayVariable,logical'
x[i, j, ..., drop = TRUE]

## S4 method for signature 'CategoricalArrayVariable,ANY'
x[[i, j, ...]]

## S4 method for signature 'CategoricalArrayVariable'
x$name

```

Arguments

x	Subvariables or an array Variable (which contains subvariables)
i	which subvariables to extract
j	Invalid
...	additional arguments

name	For \$, the name (not alias) of the subvariable to extract
drop	Invalid
value	For updating, a CrunchExpr

Value

A subset of x if extracting, otherwise x duly modified

table	<i>Table function for Crunch objects</i>
-------	--

Description

Table function for Crunch objects

Usage

```
table(..., exclude, useNA, dnn, deparse.level)
```

Arguments

...	things to tabulate
exclude	see table
useNA	see table
dnn	see table
deparse.level	see table

Value

a table object

See Also

[table](#)

teams

Teams

Description

Teams contain users and datasets. You can share a dataset with a group of users by sharing the dataset with a team. You can also share a bunch of datasets with a user all at once by adding them to a team that has those datasets.

Usage

```
## S4 method for signature 'CrunchTeam'  
members(x)  
  
## S4 replacement method for signature 'CrunchTeam,MemberCatalog'  
members(x) <- value  
  
## S4 replacement method for signature 'CrunchTeam,character'  
members(x) <- value
```

Arguments

x	a CrunchTeam
value	for members<-, a character vector of emails or URLs of users to add to the team.

Details

These methods allow you to work with teams. Find your teams with the [getTeams](#) function, which returns your TeamCatalog. Extract an individual team by name. Create a team by assigning in with a new name, with the assignment value a list, either empty (to just create a team with that name), or with a "members" element, containing emails or URLs of users to add to the team. Users can be added later with the members<- method.

Value

members returns a MemberCatalog, which has references to the users that are members of the team.
members<- returns x with the given users added to the members catalog.

See Also

[getTeams](#)

`tojson-crunch`*toJSON methods for Crunch objects*

Description

crunch uses the `jsonlite` package for (de)serialization of JSON. Unlike `RJSONIO`'s `toJSON`, `toJSON` does not allow for defining S4 methods for other object types. So, `crunch::toJSON` wraps `jsonprep`, which exists to translate objects to base R objects, which `jsonlite::toJSON` can handle. `jsonprep` is defined as an S4 generic, and it is exported (unlike `codejsonlite::asJSON`), so you can define methods for it if you have other objects that you want to successfully serialize to JSON.

Usage

```
jsonprep(x, ...)  
  
## S4 method for signature 'Categories'  
jsonprep(x, ...)  
  
## S4 method for signature 'list'  
jsonprep(x, ...)  
  
## S4 method for signature 'zcl'  
jsonprep(x, ...)  
  
## S4 method for signature 'ANY'  
jsonprep(x, ...)  
  
## S4 method for signature 'VariableOrder'  
jsonprep(x, ...)  
  
## S4 method for signature 'VariableGroup'  
jsonprep(x, ...)  
  
toJSON(x, ...)
```

Arguments

<code>x</code>	the object
<code>...</code>	additional arguments

Value

`jsonprep` returns a base R object that `jsonlite::toJSON` can handle. `toJSON` returns the JSON-serialized character object.

See Also

[toJSON](#)

toVariable	<i>Generic method for converting objects to Crunch representations</i>
------------	--

Description

If you have other object types you wish to convert to Crunch variables, you can declare methods for toVariable

Usage

```
toVariable(x, ...)  
  
## S4 method for signature 'character'  
toVariable(x, ...)  
  
## S4 method for signature 'numeric'  
toVariable(x, ...)  
  
## S4 method for signature 'factor'  
toVariable(x, ...)  
  
## S4 method for signature 'Date'  
toVariable(x, ...)  
  
## S4 method for signature 'logical'  
toVariable(x, ...)
```

Arguments

x	the object
...	additional arguments

Value

a list object suitable for POSTing to the Crunch API. See the API documentation for specifications.

tuple-methods	<i>Methods for IndexTuples</i>
---------------	--------------------------------

Description

IndexTuples are objects extracted from ShojiCatalogs. They are internally used.

Usage

```

## S4 method for signature 'IndexTuple'
refresh(x)

## S4 method for signature 'IndexTuple'
x$name

## S4 replacement method for signature 'IndexTuple'
x$name <- value

## S4 method for signature 'IndexTuple,ANY'
x[[i]]

## S4 replacement method for signature 'IndexTuple,ANY,ANY,ANY'
x[[i]] <- value

## S4 method for signature 'VariableTuple'
entity(x)

## S4 method for signature 'DatasetTuple'
entity(x)

## S4 method for signature 'IndexTuple'
delete(x, ...)

## S4 method for signature 'DatasetTuple'
delete(x, confirm = interactive(), ...)

## S4 method for signature 'IndexTuple'
name(x)

## S4 method for signature 'IndexTuple'
type(x)

```

Arguments

x	a Tuple
name	a Tuple slot to get or set
value	What to set in a given slot
i	In [], a Tuple slot to get
...	additional arguments to [], ignored
confirm	For delete, whether confirmation is required. See delete .

type	<i>Change the type of Crunch variables</i>
------	--

Description

Numeric, text, and categorical variables can be cast to one another by assigning them a new "type". This modifies the storage of the data on the server and should only be done in narrow circumstances, as in when importing data from a different file format has resulted in incorrect types being specified.

Usage

```
## S4 method for signature 'CrunchVariable'
type(x)

## S4 replacement method for signature 'CrunchVariable'
type(x) <- value
```

Arguments

x	a Variable
value	For the setter, a character value in c("numeric", "text", "categorical")

Value

Getter returns character; setter returns x duly modified.

unbind	<i>Split an array or multiple-response variable into its CategoricalVariables</i>
--------	---

Description

Split an array or multiple-response variable into its CategoricalVariables

Usage

```
unbind(x)
```

Arguments

x	a CategoricalArrayVariable or MultipleResponseVariable
---	--

Value

invisibly, the API response from DELETEing the array variable definition. If you [refresh](#) the corresponding dataset after unbinding, you should see the array variable removed and its subvariables promoted to regular variables.

updateDatasetList *Refresh the local list of Crunch datasets*

Description

Refresh the local list of Crunch datasets

Usage

```
updateDatasetList()
```

Value

Nothing. Called for its side effects of setting local environment variables.

var-categories *Get and set Categories on Variables*

Description

Get and set Categories on Variables

Usage

```
## S4 method for signature 'CrunchVariable'
categories(x)

## S4 method for signature 'CategoricalVariable'
categories(x)

## S4 method for signature 'CategoricalArrayVariable'
categories(x)

## S4 replacement method for signature 'CategoricalVariable,Categories'
categories(x) <- value

## S4 replacement method for signature 'CategoricalArrayVariable,Categories'
categories(x) <- value

## S4 replacement method for signature 'CategoricalVariable,numeric'
categories(x) <- value

## S4 replacement method for signature 'CategoricalVariable,character'
categories(x) <- value
```

```
## S4 replacement method for signature 'CategoricalVariable,ANY'
categories(x) <- value

## S4 replacement method for signature 'CategoricalArrayVariable,numeric'
categories(x) <- value

## S4 replacement method for signature 'CategoricalArrayVariable,character'
categories(x) <- value

## S4 replacement method for signature 'CategoricalArrayVariable,ANY'
categories(x) <- value

## S4 replacement method for signature 'CrunchVariable,ANY'
categories(x) <- value
```

Arguments

x a Variable
value for the setters, an object of class Categories to set.

Value

Getters return Categories; setters return x duly modified.

variable-extract *"Subset" a Variable*

Description

These methods subset variables by creating Expressions, which can be composed and evaluated as needed.

Usage

```
## S4 method for signature 'CrunchVariable,CrunchExpr'
x[i, j, ..., drop = TRUE]

## S4 method for signature 'CrunchVariable,numeric'
x[i, j, ..., drop = TRUE]

## S4 method for signature 'CrunchVariable,logical'
x[i, j, ..., drop = TRUE]
```

Arguments

x	a Variable
i	a CrunchExpr, logical, or numeric
j	Invalid
...	additional arguments, ignored
drop	Invalid

Value

a CrunchExpr containing references to the variable x and the filter logic contained in i

variable-to-R	<i>Convert Variables to local R objects</i>
---------------	---

Description

Convert Variables to local R objects

Usage

```
## S4 method for signature 'CrunchVariable'
as.vector(x, mode = "any")
```

Arguments

x	a CrunchVariable subclass
mode	argument not used: part of the generic as.vector signature

Value

an R vector of the type corresponding to the Variable. E.g. CategoricalVariable yields type factor, NumericVariable yields numeric, etc.

variable-update	<i>Updating variables with expressions or values</i>
-----------------	--

Description

Updating variables with expressions or values

Usage

```
## S4 replacement method for signature 'CrunchVariable,ANY,missing,ANY'  
x[i, j] <- value  
  
## S4 replacement method for signature 'TextVariable,ANY,missing,character'  
x[i, j] <- value  
  
## S4 replacement method for signature 'NumericVariable,ANY,missing,numeric'  
x[i, j] <- value  
  
## S4 replacement method for signature 'DatetimeVariable,ANY,missing,Date'  
x[i, j] <- value  
  
## S4 replacement method for signature 'DatetimeVariable,ANY,missing,POSIXt'  
x[i, j] <- value  
  
## S4 replacement method for signature 'CrunchVariable,ANY,missing,CrunchExpr'  
x[i, j] <- value  
  
## S4 replacement method for signature 'CrunchVariable,CrunchExpr,missing,CrunchExpr'  
x[i, j] <- value  
  
## S4 replacement method for signature 'CategoricalVariable,ANY,missing,numeric'  
x[i, j] <- value  
  
## S4 replacement method for signature 'CategoricalVariable,ANY,missing,character'  
x[i, j] <- value  
  
## S4 replacement method for signature 'CategoricalVariable,ANY,missing,factor'  
x[i, j] <- value  
  
## S4 replacement method for signature 'CategoricalArrayVariable,ANY,missing,numeric'  
x[i, j] <- value  
  
## S4 replacement method for signature 'CategoricalArrayVariable,ANY,missing,character'  
x[i, j] <- value  
  
## S4 replacement method for signature 'CategoricalArrayVariable,ANY,missing,factor'  
x[i, j] <- value
```

```
## S4 replacement method for signature 'CrunchVariable,ANY'
is.na(x) <- value
```

Arguments

x	a Variable
i	a CrunchLogicalExpr or R index, optionally
j	Invalid
value	an R vector or a CrunchExpr with which to update

Value

x duly modified

VariableCatalog-class *Collection of Variables within a Dataset*

Description

A VariableCatalog contains references to all variables in a dataset, plus some descriptive metadata about each. VariableCatalogs also contain a [VariableOrder](#) that governs how variables within it are organized.

VariableOrder-class *Organize Variables within a Dataset*

Description

Variables in the Crunch web application can be viewed in an ordered, hierarchical list. These objects and methods allow you to modify that order from R.

Details

A VariableOrder object is a subclass of list that contains VariableGroups. VariableGroup objects contain a group name and an set of "entities", which can be variable references or other nested VariableGroups.

Slots

`group` character, the name of the VariableGroup. In the constructor and more generally, this field can be referenced as "name" as well.

`entities` a character vector of variable URLs, or a list containing a combination of variable URLs and VariableGroup objects.

`duplicates` logical: should duplicate variable references be allowed in this object? Default is FALSE.

`vars` either NULL or a [VariableCatalog](#). If not NULL, it will be used to look up variable names from the URLs.

VariableOrder-extract *Extract and update in VariableOrder and VariableGroup*

Description

Extract and update in VariableOrder and VariableGroup

Usage

```
## S4 method for signature 'VariableOrder,ANY'
x[i, j, ..., drop = TRUE]

## S4 method for signature 'VariableOrder,character'
x[i, j, ..., drop = TRUE]

## S4 method for signature 'VariableOrder,ANY'
x[[i, j, ...]]

## S4 method for signature 'VariableOrder,character'
x[[i, j, ...]]

## S4 method for signature 'VariableOrder'
x$name

## S4 replacement method for signature 'VariableOrder,character,missing,VariableOrder'
x[i, j] <- value

## S4 replacement method for signature 'VariableOrder,ANY,missing,VariableOrder'
x[i, j] <- value

## S4 replacement method for signature 'VariableOrder,character,missing,VariableGroup'
x[[i, j]] <- value

## S4 replacement method for signature 'VariableOrder,character,missing,CrunchDataset'
x[[i, j]] <- value
```

```
## S4 replacement method for signature 'VariableOrder,character,missing,VariableOrder'  
x[[i, j]] <- value  
  
## S4 replacement method for signature 'VariableOrder,ANY,missing,VariableGroup'  
x[[i, j]] <- value  
  
## S4 replacement method for signature 'VariableOrder,ANY,missing,ANY'  
x[[i, j]] <- value  
  
## S4 replacement method for signature 'VariableOrder,ANY,missing,`NULL`'  
x[[i, j]] <- value  
  
## S4 replacement method for signature 'VariableOrder,character,missing,`NULL`'  
x[[i, j]] <- value  
  
## S4 replacement method for signature 'VariableOrder'  
x$name <- value  
  
## S4 method for signature 'VariableGroup,ANY'  
x[i, j, ..., drop = TRUE]  
  
## S4 method for signature 'VariableGroup,character'  
x[i, j, ..., drop = TRUE]  
  
## S4 method for signature 'VariableGroup,character'  
x[[i, j, ...]]  
  
## S4 method for signature 'VariableGroup,ANY'  
x[[i, j, ...]]  
  
## S4 method for signature 'VariableGroup'  
x$name  
  
## S4 replacement method for signature 'VariableGroup,character,missing,CrunchDataset'  
x[[i, j]] <- value  
  
## S4 replacement method for signature 'VariableGroup,character,missing,list'  
x[[i, j]] <- value  
  
## S4 replacement method for signature 'VariableGroup,character,missing,character'  
x[[i, j]] <- value  
  
## S4 replacement method for signature 'VariableGroup,character,missing,VariableOrder'  
x[[i, j]] <- value  
  
## S4 replacement method for signature 'VariableGroup,character,missing,VariableGroup'  
x[[i, j]] <- value
```



```
## S4 replacement method for signature 'VariableGroup,ANY,missing,VariableGroup'
x[[i, j]] <- value

## S4 replacement method for signature 'VariableGroup'
x$name <- value
```

Arguments

x	a VariableOrder or VariableGroup
i	an index. Numeric and logical indexing supported for both classes; character indexing supported for VariableOrder, matching on VariableGroup names
j	Invalid
...	additional arguments
drop	Ignored
name	Same as i but for \$
value	For update methods, an object equivalent in class to what is being updated

Value

[[and \$ on a VariableOrder return the VariableGroup. [[on VariableGroup returns the entity within, either a character (URL) or nested VariableGroup. [and assignment methods return objects of the same class as x

VariableOrder-length *Length of VariableOrder*

Description

Length of VariableOrder

Usage

```
## S4 method for signature 'VariableOrder'
length(x)
```

Arguments

x	a VariableOrder
---	-----------------

Value

Integer: the number of VariableGroups in the VariableOrder

VariableOrder-slots *Manipulate VariableGroup and VariableOrder*

Description

Manipulate VariableGroup and VariableOrder

Usage

```
## S4 method for signature 'VariableGroup'
entities(x, simplify = FALSE)

## S4 method for signature 'VariableOrder'
entities(x, simplify = FALSE)

## S4 method for signature 'list'
entities(x, simplify = FALSE)

## S4 replacement method for signature 'VariableGroup'
entities(x) <- value

## S4 replacement method for signature 'VariableOrder'
entities(x) <- value

## S4 method for signature 'VariableGroup'
name(x)

## S4 replacement method for signature 'VariableGroup'
name(x) <- value

## S4 method for signature 'VariableOrder'
names(x)

## S4 method for signature 'VariableGroup'
names(x)

## S4 replacement method for signature 'VariableOrder'
names(x) <- value

## S4 method for signature 'VariableOrder'
duplicates(x)

## S4 method for signature 'VariableGroup'
duplicates(x)

## S4 method for signature 'VariableCatalog'
duplicates(x)
```

```
## S4 replacement method for signature 'VariableOrder,logical'
duplicates(x) <- value

## S4 replacement method for signature 'VariableGroup,logical'
duplicates(x) <- value

## S4 replacement method for signature 'VariableCatalog,logical'
duplicates(x) <- value
```

Arguments

x	a VariableGroup or VariableOrder
simplify	logical: should variable URLs inside of groups be flattened or preserved in their nested lists? Default is FALSE.
value	(1) For name, a character (length-1 vector); for names, a character vector of equal length to the number of VariableGroups being modified; for entities, either a character vector of variable URLs or a list containing a combination of variable URLs and VariableGroups. Note that group names must be unique, should be greater than 0 characters long, and "ungrouped" is a reserved group name. (2) For duplicates, logical for whether duplicate variable entries should be allowed in the VariableOrder.

Value

entities returns Variable references and VariableGroups; names returns group names; duplicates returns logical for whether duplicate variable entries should be allowed

See Also

[VariableOrder](#)
[grouped](#)

versions	<i>Access the saved versions of a dataset</i>
----------	---

Description

Access the saved versions of a dataset

Usage

```
versions(x)
```

Arguments

x	a CrunchDataset
---	-----------------

Value

an object of class `VersionCatalog`. Supported methods on the catalog include "names" and "timestamps".

See Also

[saveVersion](#) [restoreVersion](#)

weight

Dataset weights

Description

Dataset weights

Usage

```
weight(x)
```

```
weight(x) <- value
```

Arguments

x a Dataset

value a Variable to set as weight, or NULL to remove the existing weight

Value

For the getter, a Variable if there is a weight, else NULL. For the setter, x, modified accordingly

Index

- !,CrunchExpr-method (expressions), 23
- [,CategoricalArrayVariable,ANY-method (subvars-extract), 42
- [,CategoricalArrayVariable,logical-method (subvars-extract), 42
- [,CategoricalArrayVariable,numeric-method (subvars-extract), 42
- [,Categories,ANY-method (Categories-class), 6
- [,CrunchDataset,ANY-method (dataset-extract), 12
- [,CrunchDataset,character-method (dataset-extract), 12
- [,CrunchDataset,missing-method (dataset-extract), 12
- [,CrunchVariable,CrunchExpr-method (variable-extract), 51
- [,CrunchVariable,logical-method (variable-extract), 51
- [,CrunchVariable,numeric-method (variable-extract), 51
- [,ShojiCatalog,ANY-method (catalog-extract), 4
- [,ShojiCatalog,character-method (catalog-extract), 4
- [,ShojiCatalog,logical-method (catalog-extract), 4
- [,ShojiCatalog,numeric-method (catalog-extract), 4
- [,Subvariables,character-method (subvars-extract), 42
- [,VariableCatalog,VariableGroup-method (catalog-extract), 4
- [,VariableCatalog,VariableOrder-method (catalog-extract), 4
- [,VariableGroup,ANY-method (VariableOrder-extract), 55
- [,VariableGroup,character-method (VariableOrder-extract), 55
- [,VariableOrder,ANY-method (VariableOrder-extract), 55
- [,VariableOrder,character-method (VariableOrder-extract), 55
- [<-,CategoricalArrayVariable,ANY,missing,character-method (variable-update), 53
- [<-,CategoricalArrayVariable,ANY,missing,factor-method (variable-update), 53
- [<-,CategoricalArrayVariable,ANY,missing,numeric-method (variable-update), 53
- [<-,CategoricalVariable,ANY,missing,character-method (variable-update), 53
- [<-,CategoricalVariable,ANY,missing,factor-method (variable-update), 53
- [<-,CategoricalVariable,ANY,missing,numeric-method (variable-update), 53
- [<-,Categories,ANY,ANY,ANY-method (Categories-class), 6
- [<-,CrunchDataset,ANY,missing,list-method (dataset-update), 14
- [<-,CrunchDataset,CrunchExpr,ANY,ANY-method (dataset-update), 14
- [<-,CrunchVariable,ANY,missing,ANY-method (variable-update), 53
- [<-,CrunchVariable,ANY,missing,CrunchExpr-method (variable-update), 53
- [<-,CrunchVariable,CrunchExpr,missing,CrunchExpr-method (variable-update), 53
- [<-,DatetimeVariable,ANY,missing,Date-method (variable-update), 53
- [<-,DatetimeVariable,ANY,missing,POSIXt-method (variable-update), 53
- [<-,NumericVariable,ANY,missing,numeric-method (variable-update), 53
- [<-,Subvariables,ANY,missing,ANY-method (subvars-extract), 42
- [<-,Subvariables,ANY,missing,Subvariables-method (subvars-extract), 42
- [<-,Subvariables,character,missing,Subvariables-method

- (subvars-extract), 42
- [<- ,TextVariable,ANY,missing,character-method (variable-update), 53
- [<- ,VariableCatalog,VariableGroup,missing,VariableCatalog-method (catalog-extract), 4
- [<- ,VariableCatalog,VariableOrder,missing,VariableCatalog-method (catalog-extract), 4
- [<- ,VariableCatalog,character,missing,VariableCatalog-method (catalog-extract), 4
- [<- ,VariableOrder,ANY,missing,VariableOrder-method (VariableOrder-extract), 55
- [<- ,VariableOrder,character,missing,VariableOrder-method (VariableOrder-extract), 55
- [[,CategoricalArrayVariable,ANY-method (subvars-extract), 42
- [[,CrunchDataset,ANY-method (dataset-extract), 12
- [[,CrunchDataset,character-method (dataset-extract), 12
- [[,DatasetCatalog,ANY-method (catalog-extract), 4
- [[,DatasetCatalog,character-method (catalog-extract), 4
- [[,IndexTuple,ANY-method (tuple-methods), 47
- [[,ShojiCatalog,ANY-method (catalog-extract), 4
- [[,Subvariables,ANY-method (subvars-extract), 42
- [[,Subvariables,character-method (subvars-extract), 42
- [[,TeamCatalog,character-method (catalog-extract), 4
- [[,TeamCatalog,numeric-method (catalog-extract), 4
- [[,VariableCatalog,ANY-method (catalog-extract), 4
- [[,VariableCatalog,character-method (catalog-extract), 4
- [[,VariableGroup,ANY-method (VariableOrder-extract), 55
- [[,VariableGroup,character-method (VariableOrder-extract), 55
- [[,VariableOrder,ANY-method (VariableOrder-extract), 55
- [[,VariableOrder,character-method (VariableOrder-extract), 55
- [[<- ,CrunchDataset,ANY,ANY,ANY-method (dataset-update), 14
- [[<- ,CrunchDataset,ANY,missing,CrunchVariable-method (dataset-update), 14
- [[<- ,CrunchDataset,ANY,missing,ANY-method (dataset-update), 14
- [[<- ,CrunchDataset,character,missing,CrunchExpr-method (dataset-update), 14
- [[<- ,CrunchDataset,character,missing,CrunchLogicalExpr-method (dataset-update), 14
- [[<- ,CrunchDataset,character,missing,CrunchVariable-method (dataset-update), 14
- [[<- ,IndexTuple,ANY,ANY,ANY-method (tuple-methods), 47
- [[<- ,MemberCatalog,ANY,missing,ANY-method (catalog-extract), 4
- [[<- ,MemberCatalog,character,missing,NULL-method (catalog-extract), 4
- [[<- ,Subvariables,ANY,missing,ANY-method (subvars-extract), 42
- [[<- ,Subvariables,ANY,missing,CrunchVariable-method (subvars-extract), 42
- [[<- ,Subvariables,ANY,missing,NULL-method (subvars-extract), 42
- [[<- ,Subvariables,character,missing,CrunchVariable-method (subvars-extract), 42
- [[<- ,TeamCatalog,character,missing,CrunchTeam-method (catalog-extract), 4
- [[<- ,TeamCatalog,character,missing,list-method (catalog-extract), 4
- [[<- ,VariableCatalog,character,missing,CrunchVariable-method (catalog-extract), 4
- [[<- ,VariableCatalog,character,missing,VariableTuple-method (catalog-extract), 4
- [[<- ,VariableGroup,ANY,missing,VariableGroup-method (VariableOrder-extract), 55
- [[<- ,VariableGroup,character,missing,CrunchDataset-method (VariableOrder-extract), 55
- [[<- ,VariableGroup,character,missing,VariableGroup-method (VariableOrder-extract), 55
- [[<- ,VariableGroup,character,missing,VariableOrder-method (VariableOrder-extract), 55
- [[<- ,VariableGroup,character,missing,character-method (VariableOrder-extract), 55
- [[<- ,VariableGroup,character,missing,list-method (VariableOrder-extract), 55
- [[<- ,VariableOrder,ANY,missing,ANY-method (VariableOrder-extract), 55
- [[<- ,VariableOrder,ANY,missing,NULL-method

- (VariableOrder-extract), 55
- [[<- , VariableOrder, ANY, missing, VariableGroup-method (VariableOrder-extract), 55
- [[<- , VariableOrder, character, missing, CrunchDataset-method (VariableOrder-extract), 55
- [[<- , VariableOrder, character, missing, NULL-method (VariableOrder-extract), 55
- [[<- , VariableOrder, character, missing, VariablesGroup-method (VariableOrder-extract), 55
- [[<- , VariableOrder, character, missing, VariableOrder-method (VariableOrder-extract), 55
- \$, CategoricalArrayVariable-method (subvars-extract), 42
- \$, Category-method (category-extract), 7
- \$, CrunchDataset-method (dataset-extract), 12
- \$, IndexTuple-method (tuple-methods), 47
- \$, Subvariables-method (subvars-extract), 42
- \$, TeamCatalog-method (catalog-extract), 4
- \$, VariableGroup-method (VariableOrder-extract), 55
- \$, VariableOrder-method (VariableOrder-extract), 55
- \$<- , Category-method (category-extract), 7
- \$<- , CrunchDataset-method (dataset-update), 14
- \$<- , IndexTuple-method (tuple-methods), 47
- \$<- , VariableGroup-method (VariableOrder-extract), 55
- \$<- , VariableOrder-method (VariableOrder-extract), 55
- %in%, CategoricalVariable, character-method (expressions), 23
- %in%, CategoricalVariable, factor-method (expressions), 23
- %in%, CategoricalVariable, numeric-method (expressions), 23
- %in%, DatetimeVariable, Date-method (expressions), 23
- %in%, DatetimeVariable, POSIXt-method (expressions), 23
- %in%, NumericVariable, numeric-method (expressions), 23
- %in%, TextVariable, character-method (expressions), 23
- alias, CrunchVariable-method (describe), 17
- alias<- (describe), 17
- alias<- , CrunchVariable-method (describe), 17
- aliases, describe-catalog, 18
- aliases, Subvariables-method (describe-catalog), 18
- aliases, VariableCatalog-method (describe-catalog), 18
- aliases<- (describe-catalog), 18
- aliases<- , Subvariables-method (describe-catalog), 18
- aliases<- , VariableCatalog-method (describe-catalog), 18
- allVariables (dataset-variables), 15
- allVariables, CrunchDataset-method (dataset-variables), 15
- allVariables<- (dataset-variables), 15
- allVariables<- , CrunchDataset, VariableCatalog-method (dataset-variables), 15
- appendDataset, 3
- as.data.frame.CrunchDataFrame (dataset-to-R), 13
- as.data.frame.CrunchDataset (dataset-to-R), 13
- as.vector, CrunchExpr-method (expressions), 23
- as.vector, CrunchVariable-method (variable-to-R), 52
- bin (expressions), 23
- catalog-extract, 4
- catalog-length, 6
- CategoricalArrayVariable (CrunchVariable-class), 10
- CategoricalArrayVariable-class (CrunchVariable-class), 10
- CategoricalVariable (CrunchVariable-class), 10
- CategoricalVariable-class (CrunchVariable-class), 10
- Categories, 18, 20, 21
- Categories (Categories-class), 6
- categories (var-categories), 50

- categories,CategoricalArrayVariable-method (var-categories), 50
- categories,CategoricalVariable-method (var-categories), 50
- categories,CrunchVariable-method (var-categories), 50
- Categories-class, 6
- categories<- (var-categories), 50
- categories<- ,CategoricalArrayVariable,ANY-method (var-categories), 50
- categories<- ,CategoricalArrayVariable,CategoricalVariable-method (var-categories), 50
- categories<- ,CategoricalArrayVariable,character-method (var-categories), 50
- categories<- ,CategoricalArrayVariable,numeric-method (var-categories), 50
- categories<- ,CategoricalVariable,ANY-method (var-categories), 50
- categories<- ,CategoricalVariable,Categories-method (var-categories), 50
- categories<- ,CategoricalVariable,character-method (var-categories), 50
- categories<- ,CategoricalVariable,numeric-method (var-categories), 50
- categories<- ,CrunchVariable,ANY-method (var-categories), 50
- Category (Categories-class), 6
- Category-class (Categories-class), 6
- category-extract, 7
- crDELETE (http-methods), 28
- createDataset, 34–36
- crGET (http-methods), 28
- crPATCH (http-methods), 28
- crPOST (http-methods), 28
- crPUT (http-methods), 28
- crtabs, 8
- crunch, 9
- crunch-package (crunch), 9
- crunch-uni, 9
- crunchAPI, 28
- CrunchDataset, 37
- CrunchDataset (CrunchDataset-class), 10
- CrunchDataset-class, 10
- CrunchVariable (CrunchVariable-class), 10
- CrunchVariable-class, 10
- cube-computing, 11
- cube-methods, 11
- dataset-extract, 12
- dataset-to-R, 13
- dataset-update, 14
- dataset-variables, 15
- DatetimeVariable (CrunchVariable-class), 10
- DatetimeVariable-class (CrunchVariable-class), 10
- delete, 16, 17, 48
- delete,ANY-method (delete), 16
- delete,CategoricalArrayVariable-method (delete), 16
- delete,CrunchDataset-method (delete), 16
- delete,CrunchTeam-method (delete), 16
- delete,CrunchDatasetTuple-method (tuple-methods), 47
- delete,IndexTuple-method (tuple-methods), 47
- delete,ShojiObject-method (delete), 16
- deleteDataset, 16, 16
- describe, 17
- describe-catalog, 18
- describe-category, 20
- description (describe), 17
- description,CrunchDataset-method (describe), 17
- description,CrunchVariable-method (describe), 17
- description<- (describe), 17
- description<- ,CrunchDataset-method (describe), 17
- description<- ,CrunchVariable-method (describe), 17
- descriptions (describe-catalog), 18
- descriptions,VariableCatalog-method (describe-catalog), 18
- descriptions,VersionCatalog-method (describe-catalog), 18
- descriptions<- (describe-catalog), 18
- descriptions<- ,VariableCatalog-method (describe-catalog), 18
- dichotomize, 21, 21
- dichotomize,CategoricalArrayVariable,ANY-method (dichotomize), 21
- dichotomize,CategoricalVariable,ANY-method (dichotomize), 21
- dichotomize,Categories,character-method (dichotomize), 21

- dichotomize, Categories, logical-method (dichotomize), 21
- dichotomize, Categories, numeric-method (dichotomize), 21
- dim, 23
- dim, CrunchDataset-method (dim-dataset), 22
- dim, CubeDims-method (cube-methods), 11
- dim-dataset, 22
- dimnames, CubeDims-method (cube-methods), 11
- duplicates (VariableOrder-slots), 58
- duplicates, VariableCatalog-method (VariableOrder-slots), 58
- duplicates, VariableGroup-method (VariableOrder-slots), 58
- duplicates, VariableOrder-method (VariableOrder-slots), 58
- duplicates<- (VariableOrder-slots), 58
- duplicates<-, VariableCatalog, logical-method (VariableOrder-slots), 58
- duplicates<-, VariableGroup, logical-method (VariableOrder-slots), 58
- duplicates<-, VariableOrder, logical-method (VariableOrder-slots), 58

- emails (describe-catalog), 18
- emails, PermissionCatalog-method (describe-catalog), 18
- emails, UserCatalog-method (describe-catalog), 18
- entities (VariableOrder-slots), 58
- entities, list-method (VariableOrder-slots), 58
- entities, VariableGroup-method (VariableOrder-slots), 58
- entities, VariableOrder-method (VariableOrder-slots), 58
- entities<- (VariableOrder-slots), 58
- entities<-, VariableGroup-method (VariableOrder-slots), 58
- entities<-, VariableOrder-method (VariableOrder-slots), 58
- entity (tuple-methods), 47
- entity, DatasetTuple-method (tuple-methods), 47
- entity, VariableTuple-method (tuple-methods), 47
- expressions, 23

- getAccountUserCatalog, 24
- getTeams, 25, 45
- grep, 27
- grouped, 25, 59

- hiddenVariables, 26
- hiddenVariables<- (hideVariables), 27
- hide, 16, 26, 27
- hide, CrunchVariable-method (hide), 26
- hide, VariableCatalog-method (hide), 26
- hideVariables, 27
- http-methods, 28

- id (describe-category), 20
- id, Category-method (describe-category), 20
- id, list-method (describe-category), 20
- ids (Categories-class), 6
- ids, Categories-method (Categories-class), 6
- ids, list-method (Categories-class), 6
- ids<- (Categories-class), 6
- ids<-, Categories-method (Categories-class), 6
- index, 28
- index, ShojiCatalog-method (index), 28
- index<- (index), 28
- index<-, ShojiCatalog-method (index), 28
- is-na-categories, 29
- is.Array (is.dataset), 30
- is.CA (is.dataset), 30
- is.Categorical (is.dataset), 30
- is.CategoricalArray (is.dataset), 30
- is.dataset, 30
- is.Datetime (is.dataset), 30
- is.dichotomized (dichotomize), 21
- is.dichotomized, Categories-method (dichotomize), 21
- is.MR (is.dataset), 30
- is.Multiple (is.dataset), 30
- is.MultipleResponse (is.dataset), 30
- is.na, Categories-method (is-na-categories), 29
- is.na, Category-method (is-na-categories), 29
- is.na, CrunchVariable-method (expressions), 23
- is.na, CubeDims-method (cube-methods), 11

- is.na<-, Categories, character-method
(is-na-categories), 29
- is.na<-, Categories, logical-method
(is-na-categories), 29
- is.na<-, Category, logical-method
(is-na-categories), 29
- is.na<-, CrunchVariable, ANY-method
(variable-update), 53
- is.Numeric (is.dataset), 30
- is.selected (describe-category), 20
- is.selected, Category-method
(describe-category), 20
- is.shoji (is.dataset), 30
- is.Text (is.dataset), 30
- is.variable (is.dataset), 30

- jsonprep (tojson-crunch), 46
- jsonprep, ANY-method (tojson-crunch), 46
- jsonprep, Categories-method
(tojson-crunch), 46
- jsonprep, list-method (tojson-crunch), 46
- jsonprep, VariableGroup-method
(tojson-crunch), 46
- jsonprep, VariableOrder-method
(tojson-crunch), 46
- jsonprep, zcl-method (tojson-crunch), 46

- length, ShojiCatalog-method
(catalog-length), 6
- length, VariableOrder-method
(VariableOrder-length), 57
- listDatasets, 17, 31
- lm, 13
- loadDataset, 31, 31
- login, 32
- logout, 32

- makeArray, 33
- makeMR (makeArray), 33
- margin.table, 11
- margin.table (cube-computing), 11
- margin.table, CrunchCube-method
(cube-computing), 11
- match, 24
- max, 10
- max (crunch-uni), 9
- max, CrunchVariable-method (crunch-uni),
9
- max, DatetimeVariable-method
(crunch-uni), 9
- max, NumericVariable-method
(crunch-uni), 9
- mean, 10
- mean (crunch-uni), 9
- mean, CrunchVariable-method
(crunch-uni), 9
- mean, NumericVariable-method
(crunch-uni), 9
- median, 10
- median (crunch-uni), 9
- median, CrunchVariable-method
(crunch-uni), 9
- median, NumericVariable-method
(crunch-uni), 9
- members (teams), 45
- members, CrunchTeam-method (teams), 45
- members<- (teams), 45
- members<-, CrunchTeam, character-method
(teams), 45
- members<-, CrunchTeam, MemberCatalog-method
(teams), 45
- min, 10
- min (crunch-uni), 9
- min, CrunchVariable-method (crunch-uni),
9
- min, DatetimeVariable-method
(crunch-uni), 9
- min, NumericVariable-method
(crunch-uni), 9
- MultipleResponseVariable
(CrunchVariable-class), 10
- MultipleResponseVariable-class
(CrunchVariable-class), 10

- na-omit-categories, 34
- na.omit, Categories-method
(na-omit-categories), 34
- name (describe), 17
- name, Category-method
(describe-category), 20
- name, CrunchDataset-method (describe), 17
- name, CrunchTeam-method (describe), 17
- name, CrunchVariable-method (describe),
17
- name, IndexTuple-method (tuple-methods),
47

- name, VariableGroup-method
(VariableOrder-slots), 58
- name<- (describe), 17
- name<- , Category-method
(describe-category), 20
- name<- , CrunchDataset-method (describe),
17
- name<- , CrunchVariable-method
(describe), 17
- name<- , VariableGroup-method
(VariableOrder-slots), 58
- names, 20
- names, Categories-method
(Categories-class), 6
- names, CrunchDataset-method
(describe-catalog), 18
- names, DatasetCatalog-method
(describe-catalog), 18
- names, MemberCatalog-method
(describe-catalog), 18
- names, Subvariables-method
(describe-catalog), 18
- names, TeamCatalog-method
(describe-catalog), 18
- names, UserCatalog-method
(describe-catalog), 18
- names, VariableCatalog-method
(describe-catalog), 18
- names, VariableGroup-method
(VariableOrder-slots), 58
- names, VariableOrder-method
(VariableOrder-slots), 58
- names, VersionCatalog-method
(describe-catalog), 18
- names<- , Categories-method
(Categories-class), 6
- names<- , Subvariables-method
(describe-catalog), 18
- names<- , VariableCatalog-method
(describe-catalog), 18
- names<- , VariableOrder-method
(VariableOrder-slots), 58
- newDataset, 34, 35, 36
- newDatasetByColumn, 35, 35, 36
- newDatasetByCSV, 35, 35
- newDatasetFromFile, 36
- NumericVariable (CrunchVariable-class),
10
- NumericVariable-class
(CrunchVariable-class), 10
- ordering, 37
- ordering, CrunchDataset-method
(ordering), 37
- ordering, VariableCatalog-method
(ordering), 37
- ordering<- (ordering), 37
- ordering<- , CrunchDataset-method
(ordering), 37
- ordering<- , VariableCatalog-method
(ordering), 37
- prop. table, 11
- prop. table (cube-computing), 11
- prop. table, CrunchCube-method
(cube-computing), 11
- refresh, 37, 49
- refresh, CrunchDataset-method (refresh),
37
- refresh, CrunchVariable-method
(refresh), 37
- refresh, IndexTuple-method
(tuple-methods), 47
- refresh, ShojiObject-method (refresh), 37
- restoreVersion, 38, 39, 60
- rollup (expressions), 23
- round, 11
- round, CrunchCube-method
(cube-computing), 11
- saveVersion, 38, 39, 60
- sd, 10
- sd (crunch-uni), 9
- sd, CrunchVariable-method (crunch-uni), 9
- sd, NumericVariable-method (crunch-uni),
9
- self, 39
- self, ShojiObject-method (self), 39
- share, 40
- ShojiObject (ShojiObject-class), 40
- ShojiObject-class, 40
- show, 41
- show, Categories-method (show-crunch), 41
- show, Category-method (show-crunch), 41
- show, CrunchCube-method (show-crunch), 41
- show, ShojiObject-method (show-crunch),
41

- show-crunch, 41
- Subvariables, 18, 20
- Subvariables (Subvariables-class), 41
- subvariables (Subvariables-class), 41
- subvariables, CategoricalArrayVariable-method (Subvariables-class), 41
- Subvariables-class, 41
- subvariables<- (Subvariables-class), 41
- subvariables<-, CategoricalArrayVariable, ANY-method (Subvariables-class), 41
- subvariables<-, CategoricalArrayVariable, Subvariables-class (Subvariables-class), 41
- subvars-extract, 42

- table, 8, 44, 44
- teams, 25, 45
- TextVariable (CrunchVariable-class), 10
- TextVariable-class (CrunchVariable-class), 10
- timestamps (describe-catalog), 18
- timestamps, VersionCatalog-method (describe-catalog), 18
- toJSON, 46
- toJSON (tojson-crunch), 46
- tojson-crunch, 46
- toVariable, 47
- toVariable, character-method (toVariable), 47
- toVariable, Date-method (toVariable), 47
- toVariable, factor-method (toVariable), 47
- toVariable, logical-method (toVariable), 47
- toVariable, numeric-method (toVariable), 47
- tuple-methods, 47
- type, 49
- type, CrunchVariable-method (type), 49
- type, IndexTuple-method (tuple-methods), 47
- type<- (type), 49
- type<-, CrunchVariable-method (type), 49
- types (describe-catalog), 18
- types, VariableCatalog-method (describe-catalog), 18

- unbind, 49
- undichotomize (dichotomize), 21
- undichotomize, CategoricalArrayVariable-method (dichotomize), 21
- undichotomize, CategoricalVariable-method (dichotomize), 21
- undichotomize, Categories-method (dichotomize), 21
- ungrouped (grouped), 25
- unhide (hide), 26
- unhide, CrunchVariable-method (hide), 26
- unhide, VariableCatalog-method (hide), 26
- unhideVariables (hideVariables), 27
- updateDatasetList, 50

- value (describe-category), 20
- value, Category-method (describe-category), 20
- value<- (describe-category), 20
- value<-, Category-method (describe-category), 20
- values (Categories-class), 6
- values, Categories-method (Categories-class), 6
- values<- (Categories-class), 6
- values<-, Categories-method (Categories-class), 6
- var-categories, 50
- variable-extract, 51
- variable-to-R, 52
- variable-update, 53
- VariableCatalog, 10, 37, 55
- VariableCatalog (VariableCatalog-class), 54
- VariableCatalog-class, 54
- VariableGroup (VariableOrder-class), 54
- VariableGroup-class (VariableOrder-class), 54
- VariableOrder, 25, 37, 54, 59
- VariableOrder (VariableOrder-class), 54
- VariableOrder-class, 54
- VariableOrder-extract, 55
- VariableOrder-length, 57
- VariableOrder-slots, 58
- variables (dataset-variables), 15
- variables, CrunchDataset-method (dataset-variables), 15
- variables<- (dataset-variables), 15
- variables<-, CrunchDataset, VariableCatalog-method (dataset-variables), 15
- versions, 38, 39, 59

weight, [8](#), [60](#)

weight<- (weight), [60](#)

xtabs, [8](#)