

Package ‘ilabelled’

June 15, 2024

Type Package

Title Simple Handling of Labelled Data

Version 0.9.0

Author Christof Lewerenz [aut, cre]

Maintainer Christof Lewerenz <christof.lewerenz@gmx.net>

Description Simple handling of survey data. Smart handling of meta-information like e.g. variable-labels value-labels and scale-levels. Easy access and validation of meta-information. Useage of value labels and values respectively for subsetting and recoding data.

License MIT + file LICENSE

Encoding UTF-8

Language en-US

Suggests knitr, rmarkdown, testthat (>= 3.0.0)

Imports methods, stats

URL <https://github.com/clewerenz/ilabelled>

BugReports <https://github.com/clewerenz/ilabelled/issues>

Config/testthat/edition 3

RoxygenNote 7.3.1

Depends R (>= 2.10)

VignetteBuilder knitr

NeedsCompilation yes

Repository CRAN

Date/Publication 2024-06-15 09:30:02 UTC

Contents

| | |
|--------------------------|---|
| .init | 3 |
| .is_sequential | 4 |
| .i_find_in | 5 |
| .i_in | 5 |

| | |
|-----------------------------------|----|
| .merge_labels | 6 |
| .valid_annotation | 6 |
| .valid_label | 7 |
| .valid_labels | 7 |
| .valid_missing | 8 |
| .valid_scale | 8 |
| .valid_subject | 9 |
| .valid_wording | 9 |
| as.i_labelled | 10 |
| grapesingrapes | 10 |
| is.i_labelled | 11 |
| is_decimal | 11 |
| i_annotation | 12 |
| i_assert_labels | 12 |
| i_as_character | 13 |
| i_as_factor | 13 |
| i_as_numeric | 14 |
| i_copy | 15 |
| i_get_annotation | 15 |
| i_get_equal_subject | 16 |
| i_get_equal_wording | 16 |
| i_get_label | 17 |
| i_get_labels | 17 |
| i_get_na_range | 18 |
| i_get_na_values | 18 |
| i_get_scale | 19 |
| i_get_subject | 19 |
| i_get_wording | 20 |
| i_label | 20 |
| i_labelled | 21 |
| i_labels | 22 |
| i_missing_to_na | 22 |
| i_na_range | 23 |
| i_na_values | 23 |
| i_print_annotation | 24 |
| i_print_label | 24 |
| i_print_labels | 25 |
| i_print_na_range | 25 |
| i_print_na_values | 26 |
| i_print_scale | 26 |
| i_print_subject | 27 |
| i_print_wording | 27 |
| i_recode | 28 |
| i_remove_annotation | 29 |
| i_remove_label | 29 |
| i_remove_labels | 30 |
| i_remove_missing_labels | 30 |
| i_remove_na_range | 31 |

| | |
|------------------------------|----|
| i_remove_na_values | 31 |
| i_remove_scale | 32 |
| i_remove_subject | 32 |
| i_remove_wording | 33 |
| i_scale | 33 |
| i_sort_labels | 34 |
| i_subject | 34 |
| i_table | 35 |
| i_to_base_class | 35 |
| i_unclass | 36 |
| i_valid_annotation | 37 |
| i_valid_label | 37 |
| i_valid_labels | 38 |
| i_valid_scale | 38 |
| i_valid_subject | 39 |
| i_valid_wording | 39 |
| i_wording | 40 |
| print.i_labelled | 40 |
| [i_labelled | 41 |

Index **42**

| | |
|--------------------|-------------------------------|
| <code>.init</code> | <i>backend for i_labelled</i> |
|--------------------|-------------------------------|

Description

all arguments are passed from `i_labelled`

Usage

```
.init(
  x,
  label = NULL,
  labels = NULL,
  na_values = NULL,
  na_range = NULL,
  scale = NULL,
  annotation = NULL,
  wording = NULL,
  subject = NULL,
  ...
)
```

Arguments

| | |
|------------|---|
| x | vector |
| label | variable label |
| labels | value labels as named vector |
| na_values | missing values (e.g. c(888, 999)) |
| na_range | range of missing values (e.g. c(-9,-1)) |
| scale | scale level (nominal, ordinal, scale) |
| annotation | additional information about variable |
| wording | question text |
| subject | subject |
| ... | further attributes passed to class |

Value

x as *i_labelled* object with attributes applied to it

| | |
|-----------------------------|---|
| <code>.is_sequential</code> | <i>checks if vector is numeric sequence</i> |
|-----------------------------|---|

Description

checks if vector is numeric sequence

Usage

```
.is_sequential(x)
```

Arguments

| | |
|---|--------|
| x | vector |
|---|--------|

Value

T/F

| | |
|-------------------------|---|
| <code>.i_find_in</code> | <i>internal replacement of match function for remove missing values (match is much slower but can handle more data classes)</i> |
|-------------------------|---|

Description

description description

Usage

`.i_find_in(x, y)`

Arguments

| | |
|----------------|--------|
| <code>x</code> | vector |
| <code>y</code> | vector |

Value

Vector of T/F values with length of x

| | |
|--------------------|---------------------|
| <code>.i_in</code> | <i>Match values</i> |
|--------------------|---------------------|

Description

Find matches (return T/F)

Usage

`.i_in(x, table)`

Arguments

| | |
|--------------------|---|
| <code>x</code> | vector or NULL: the values to be matched. Long vectors are supported. |
| <code>table</code> | vector or NULL: the values to be matched against. Long vectors are not supported. |

Value

Vector of T/F values with length of x

| | |
|----------------------------|---|
| <code>.merge_labels</code> | <i>combine old value labels with new value labels</i> |
|----------------------------|---|

Description

combine old value labels with new value labels

Usage

```
.merge_labels(old_labs, new_labs)
```

Arguments

| | |
|-----------------------|--------------|
| <code>old_labs</code> | named vector |
| <code>new_labs</code> | named vector |

Value

Returns names vector of value labels

| | |
|--------------------------------|-------------------------------------|
| <code>.valid_annotation</code> | <i>validate annotation - intern</i> |
|--------------------------------|-------------------------------------|

Description

contains run-time-tests annotation runs internally

Usage

```
.valid_annotation(x)
```

Arguments

| | |
|----------------|--------------------------|
| <code>x</code> | character vector or NULL |
|----------------|--------------------------|

Value

T/F

.valid_label *validate variable label - intern*

Description

run-time-tests for variable label runs internally

Usage

.valid_label(x)

Arguments

x vector

Value

T/F

.valid_labels *validate value labels - intern*

Description

contains several run-time-tests for value labels runs internally

Usage

.valid_labels(x)

Arguments

x named vector (label = value)

Value

No return value. Aborts process when run-time-tests fail.

.valid_missing *validate missing values/range - intern*

Description

validate missing values/range - intern

Usage

`.valid_missing(x)`

Arguments

x vector

Value

T/F

.valid_scale *validate scale label - intern*

Description

run-time-tests for scale level runs internally

Usage

`.valid_scale(x)`

Arguments

x vector

Value

T/F

.valid_subject *validate subject - intern*

Description

contains run-time-tests subject runs internally

Usage

.valid_subject(x)

Arguments

x character vector or NULL

Value

T/F

.valid_wording *validate wording - intern*

Description

contains run-time-tests wording runs internally

Usage

.valid_wording(x)

Arguments

x character vector or NULL

Value

T/F

| | |
|---------------|-----------------------------------|
| as.i_labelled | <i>coerce to i_labelled class</i> |
|---------------|-----------------------------------|

Description

coerce to i_labelled class

Usage

```
as.i_labelled(x, ...)
```

Arguments

| | |
|-----|----------------------------|
| x | vector |
| ... | attributes passed to class |

Value

vector of class i_labelled

| | |
|----------------|-------------------------|
| grapesingrapes | <i>generic for %in%</i> |
|----------------|-------------------------|

Description

generic for %in%

Usage

```
x %in% table
```

Arguments

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

Value

T/F

`is.i_labelled` *check for class i_labelled*

Description

check for class i_labelled

Usage

`is.i_labelled(x)`

Arguments

`x` vector of class i_labelled

Value

T/F

`is_decimal` *Check if vector contains decimal values*

Description

Check if vector contains decimal values

Usage

`is_decimal(x)`

Arguments

`x` numeric vector

Value

T/F

| | |
|--------------|-----------------------------------|
| i_annotation | <i>add annotation to variable</i> |
|--------------|-----------------------------------|

Description

add annotation to i_labelled object
 can be used to store additional information about a variable

Usage

```
i_annotation(x, annotation, overwrite = FALSE)
```

Arguments

| | |
|------------|---|
| x | vector |
| annotation | variable label as string or NULL (NULL will remove label) |
| overwrite | overwrite existing annotation and replace with new annotation |

Value

x with annotation applied

| | |
|-----------------|--|
| i_assert_labels | <i>Check for required value labels in set of variables</i> |
|-----------------|--|

Description

Check for required value labels in set of variables

Usage

```
i_assert_labels(x, labels, info = NULL, verbose = TRUE)
```

Arguments

| | |
|---------|--|
| x | data.frame |
| labels | character vector |
| info | string with info message (purpose of assertion) - optional |
| verbose | return TRUE when assertion is successful |

Value

No return value (except when verbose = T). Aborts process when test not valid.

| | |
|----------------|---------------------|
| i_as_character | <i>as character</i> |
|----------------|---------------------|

Description

make character from i_labelled

Usage

```
i_as_character(  
  x,  
  missing_to_na = FALSE,  
  require_all_labels = FALSE,  
  keep_attributes = FALSE  
)
```

Arguments

| | |
|--------------------|--|
| x | vector |
| missing_to_na | as missing declared values will become NA |
| require_all_labels | process will be interrupted, when not all values have valid labels |
| keep_attributes | should attributes be preserved |

Value

character vector

| | |
|-------------|------------------|
| i_as_factor | <i>as factor</i> |
|-------------|------------------|

Description

make factor from i_labelled

Usage

```
i_as_factor(  
  x,  
  missing_to_na = FALSE,  
  require_all_labels = FALSE,  
  keep_attributes = FALSE  
)
```

Arguments

x vector
missing_to_na as missing declared values will become NA
require_all_labels
 process will be interrupted, when not all values have valid labels
keep_attributes
 should attributes be preserved

Value

vector of class factor

| | |
|---------------------|-------------------|
| <i>i_as_numeric</i> | <i>as numeric</i> |
|---------------------|-------------------|

Description

make numeric from *i_labelled*

Usage

```
i_as_numeric(x, missing_to_na = FALSE, keep_attributes = FALSE)
```

Arguments

x vector
missing_to_na as missing declared values will become NA
keep_attributes
 should attributes be preserved

Value

numeric vector

| | |
|--------|---|
| i_copy | <i>copy labels from one variable to another</i> |
|--------|---|

Description

copy labels from one variable to another

Usage

```
i_copy(to, from, what = "all", ...)
```

Arguments

| | |
|------|--|
| to | vector |
| from | vector |
| what | character vector describing which labels are copied: 'all' (default), 'label', 'labels', 'na_values', 'na_range' |
| ... | further attributes passed to structure |

Value

Returns 'to' with ilabelled attributes copied from 'from'

| | |
|------------------|-----------------------|
| i_get_annotation | <i>get annotation</i> |
|------------------|-----------------------|

Description

return annotation as character vector applied to vector return list when applied to data.frame

Usage

```
i_get_annotation(x)
```

Arguments

| | |
|---|----------------------|
| x | vector or data.frame |
|---|----------------------|

Value

returns annotation

`i_get_equal_subject` *get variable names by subject*

Description

return all variable names by subjects
one, several, or all subjects can be looked up

Usage

```
i_get_equal_subject(x, subject = NULL)
```

Arguments

| | |
|----------------------|---|
| <code>x</code> | <code>data.frame</code> |
| <code>subject</code> | one or more subjects as character vector. when NULL return all variable names by all subjects in data |

Value

named list or NA. return named list with one list entry for each subject. when no subject in data or no match for subjects, return NA.

`i_get_equal_wording` *get variable names by wording*

Description

return all variable names by wordings
one, several, or all wordings can be looked up

Usage

```
i_get_equal_wording(x, wording = NULL)
```

Arguments

| | |
|----------------------|---|
| <code>x</code> | <code>data.frame</code> |
| <code>wording</code> | one or more wordings as character vector. when NULL return all variable names by all wordings in data |

Value

named list or NA. return named list with one list entry for each wording. when no wording in data or no match for wordings, return NA.

| | |
|--------------------------|---------------------------|
| <code>i_get_label</code> | <i>get variable label</i> |
|--------------------------|---------------------------|

Description

return variable label when applied to vector return list when applied to data.frame

Usage

```
i_get_label(x)
```

Arguments

x vector or data.frame

Value

variable label

| | |
|---------------------------|-------------------------|
| <code>i_get_labels</code> | <i>get value labels</i> |
|---------------------------|-------------------------|

Description

return labels when applied to vector return list when applied to data.frame

Usage

```
i_get_labels(x)
```

Arguments

x vector or data.frame

Value

values and value labels as data.frame

| | |
|-----------------------------|--------------------------|
| <code>i_get_na_range</code> | <i>get missing range</i> |
|-----------------------------|--------------------------|

Description

return missing range when applied to vector return list when applied to data.frame

Usage

```
i_get_na_range(x)
```

Arguments

x vector or data.frame

Value

return missing range

| | |
|------------------------------|---------------------------|
| <code>i_get_na_values</code> | <i>get missing values</i> |
|------------------------------|---------------------------|

Description

return missing values when applied to vector return list when applied to data.frame

Usage

```
i_get_na_values(x)
```

Arguments

x vector or data.frame

Value

return missing values

| | |
|--------------------------|------------------------|
| <code>i_get_scale</code> | <i>get scale level</i> |
|--------------------------|------------------------|

Description

return scale level when applied to vector return list when applied to data.frame

Usage

```
i_get_scale(x)
```

Arguments

x vector or data.frame

Value

returns scale level

| | |
|----------------------------|--------------------|
| <code>i_get_subject</code> | <i>get subject</i> |
|----------------------------|--------------------|

Description

return subject as character vector applied to vector return list when applied to data.frame

Usage

```
i_get_subject(x)
```

Arguments

x vector or data.frame

Value

returns subject

| | |
|----------------------------|--------------------|
| <code>i_get_wording</code> | <i>get wording</i> |
|----------------------------|--------------------|

Description

return wording as character vector applied to vector return list when applied to data.frame

Usage

```
i_get_wording(x)
```

Arguments

x vector or data.frame

Value

returns wording

| | |
|----------------------|---------------------------|
| <code>i_label</code> | <i>set variable label</i> |
|----------------------|---------------------------|

Description

set variable label

Usage

```
i_label(x, label)
```

Arguments

x vector
label variable label as string or NULL (NULL will remove label)

Value

x with variable label applied

| | |
|------------|--------------------------|
| i_labelled | <i>class constructor</i> |
|------------|--------------------------|

Description

class constructor

Usage

```
i_labelled(  
  x,  
  label = NULL,  
  labels = NULL,  
  na_values = NULL,  
  na_range = NULL,  
  scale = NULL,  
  annotation = NULL,  
  wording = NULL,  
  subject = NULL,  
  ...  
)
```

Arguments

| | |
|------------|---|
| x | vector or data.frame |
| label | variable label |
| labels | value labels as named vector (e.g. c("A"=1, "B"=2) or setNames(c(1,2), c("A","B"))) |
| na_values | missing values (e.g. c(888, 999)) |
| na_range | range of missing values as vector length 2 (e.g. c(-9,-1)) |
| scale | scale level (nominal, ordinal, scale) |
| annotation | additional information about variable |
| wording | question text |
| subject | subject |
| ... | further attributes passed to class |

Value

vector or data.frame

| | |
|-----------------------|-------------------------|
| <code>i_labels</code> | <i>set value labels</i> |
|-----------------------|-------------------------|

Description

set value labels

Usage

```
i_labels(x, ..., overwrite = FALSE)
```

Arguments

| | |
|------------------------|--|
| <code>x</code> | vector |
| <code>...</code> | set labels for values (e.g. <code>label_of_choice = 1</code> or <code>"Label of Choice" = 1</code>); remove single label with <code>NULL = value</code> (e.g. <code>NULL = 1</code>); removes all value labels when only <code>NULL</code> (e.g. <code>i_label(x, NULL)</code>) |
| <code>overwrite</code> | should new labels be merged with existing labels or remove existing labels |

Value

returns `x` with value labels applied

| | |
|------------------------------|-----------------------------|
| <code>i_missing_to_na</code> | <i>missing values to NA</i> |
|------------------------------|-----------------------------|

Description

all values declared as missing will be recoded as NA

Usage

```
i_missing_to_na(x, remove_missing_labels = FALSE)
```

Arguments

| | |
|------------------------------------|--|
| <code>x</code> | vector or data.frame |
| <code>remove_missing_labels</code> | remove values labels from values which are declared as missing |

Value

Returns `x` with missing values coerced to NA

| | |
|------------|-----------------------------|
| i_na_range | <i>define missing range</i> |
|------------|-----------------------------|

Description

define which values will be handled as missing values

Usage

```
i_na_range(x, values)
```

Arguments

| | |
|--------|---|
| x | vector |
| values | vector with missing range e.g. c(-9:-1) or NULL (NULL will remove all missing values) |

Value

Returns x with missing range set

| | |
|-------------|------------------------------|
| i_na_values | <i>define missing values</i> |
|-------------|------------------------------|

Description

define which values will be handled as missing values

Usage

```
i_na_values(x, values, sort = TRUE, desc = FALSE)
```

Arguments

| | |
|--------|--|
| x | vector |
| values | vector with missing values e.g. c(888,999) or NULL (NULL will remove all missing values) |
| sort | sort values |
| desc | sort values in descending order |

Value

Returns x with missing values set

`i_print_annotation` *print annotation*

Description

print annotation

Usage

`i_print_annotation(x)`

Arguments

`x` vector

Value

No return value. Print annotation attribute to console

`i_print_label` *print variable label*

Description

print variable label

Usage

`i_print_label(x)`

Arguments

`x` vector

Value

No return value. Print variable label to console

i_print_labels *print value labels*

Description

print value labels

Usage

`i_print_labels(x)`

Arguments

x vector

Value

No return value. Print labels to console

i_print_na_range *print missing range*

Description

print missing range

Usage

`i_print_na_range(x)`

Arguments

x vector

Value

No return value. Print na range to console

i_print_na_values *print missing values*

Description

print missing values

Usage

```
i_print_na_values(x)
```

Arguments

x vector

Value

No return value. Print na values to console

i_print_scale *print scale level*

Description

print scale level

Usage

```
i_print_scale(x)
```

Arguments

x vector

Value

No return value. Print scale level to console

i_print_subject *print subject*

Description

print subject

Usage

i_print_subject(x)

Arguments

x vector

Value

No return value. Print subject attribute to console

i_print_wording *print wording*

Description

print wording

Usage

i_print_wording(x)

Arguments

x vector

Value

No return value. Print wording attribute to console

| | |
|----------|---|
| i_recode | <i>i_recode</i> Function for recoding new variable from origin variable(s). |
|----------|---|

Description

Returns a vector object of class `i_labelled`

Usage

```
i_recode(
  x,
  ...,
  label = NULL,
  na_values = NULL,
  na_range = NULL,
  scale = NULL,
  annotation = NULL,
  wording = NULL,
  copy = NULL
)
```

Arguments

| | |
|-------------------------|--|
| <code>x</code> | vector |
| <code>...</code> | formula for recoding of values. See examples. |
| <code>label</code> | variable label |
| <code>na_values</code> | a vector with missing values |
| <code>na_range</code> | a vector for missing range |
| <code>scale</code> | scale level (nominal, ordinal, metric) |
| <code>annotation</code> | addition information about variable |
| <code>wording</code> | question text |
| <code>copy</code> | a variable from <code>x</code> . Copy the values of an existing variable before recoding values according to ... |

Details

Can be applied to either vector or `data.frame`. When `x` is `data.frame` the formula passed to `...` is different from when it is applied to single vector. When function is applied to a `data.frame`, multiple conditions on multiple variables are possible (e.g when variable X is equal to this, do that; when variable Y is not equal to this, do that, etc.). See examples for further clarification.

You can recode directly via value labels by using

Value

Returns `i_labelled` vector with values defined by formula and information given to function.

Examples

```
# When applied to a single vector:
# keep in mind that when function is applied to vector, instead of a column use x
myVector <- i_labelled(1:4, labels = c("A" = 1, "B" = 2, "C" = 3, "D" = 4))
i_recode(x = myVector, "AB" = 1 ~ x %in% c("A", "B"), "CD" = 2 ~ x == c(3, 4))

# When applied to data.frame (multiple conditions)
myData <- data.frame(
  V1 = i_labelled(1:3, labels = c("A" = 1, "B" = 2, "C" = 3)),
  V2 = i_labelled(c(2:3,-9))
)
i_recode(x = myData, A = 1 ~ V1 %in% c("A", "B"), 2 ~ "V2" == 3, "C" = 999 ~ V2 == -9)
```

i_remove_annotation *remove annotation*

Description

remove annotation label from variable keep other attributes

Usage

```
i_remove_annotation(x)
```

Arguments

x vector or data.frame

Value

Returns x without annotation

i_remove_label *remove variable label*

Description

remove variable label keep other attributes

Usage

```
i_remove_label(x)
```

Arguments

x vector or data.frame

Value

Returns x without variable label

| | |
|------------------------------|--------------------------------|
| <code>i_remove_labels</code> | <i>remove all value labels</i> |
|------------------------------|--------------------------------|

Description

remove all value labels keep other attributes

Usage

```
i_remove_labels(x)
```

Arguments

x vector or data.frame

Value

Returns x without value labels

| | |
|--------------------------------------|------------------------------|
| <code>i_remove_missing_labels</code> | <i>remove missing labels</i> |
|--------------------------------------|------------------------------|

Description

remove values labels from values which are declared as missing

Usage

```
i_remove_missing_labels(x)
```

Arguments

x vector or data.frame

Value

Returns x without missing labels

i_remove_na_range *remove as na range*

Description

remove na range (information which values should be handled as missing) keep other attributes

Usage

`i_remove_na_range(x)`

Arguments

x vector or data.frame

Value

Returns x without na-range

i_remove_na_values *remove as na values*

Description

remove na values (information which values should be handled as missing) keep other attributes

Usage

`i_remove_na_values(x)`

Arguments

x vector or data.frame

Value

Returns x without na-values

| | |
|-----------------------------|---------------------------|
| <code>i_remove_scale</code> | <i>remove scale level</i> |
|-----------------------------|---------------------------|

Description

remove scale label from variable keep other attributes

Usage

```
i_remove_scale(x)
```

Arguments

x vector or data.frame

Value

Returns x without scale level

| | |
|-------------------------------|-----------------------|
| <code>i_remove_subject</code> | <i>remove subject</i> |
|-------------------------------|-----------------------|

Description

remove subject label from variable keep other attributes

Usage

```
i_remove_subject(x)
```

Arguments

x vector or data.frame

Value

Returns x without subject

| | |
|-------------------------|-----------------------|
| <i>i_remove_wording</i> | <i>remove wording</i> |
|-------------------------|-----------------------|

Description

remove wording label from variable keep other attributes

Usage

```
i_remove_wording(x)
```

Arguments

| | |
|---|----------------------|
| x | vector or data.frame |
|---|----------------------|

Value

Returns x without wording

| | |
|----------------|------------------------|
| <i>i_scale</i> | <i>set scale level</i> |
|----------------|------------------------|

Description

set scale level

Usage

```
i_scale(x, scale = NULL)
```

Arguments

| | |
|-------|--|
| x | vector |
| scale | scale level (nominal, ordinal, scale) as string or NULL (NULL will remove scale level) |

Value

Returns x with scale label set

| | |
|----------------------------|---|
| <code>i_sort_labels</code> | <i>sort value labels by values or by labels</i> |
|----------------------------|---|

Description

sort value labels by values or by labels

Usage

```
i_sort_labels(x, by = "values", decreasing = FALSE)
```

Arguments

| | |
|-------------------------|-------------------------|
| <code>x</code> | vector or data.frame |
| <code>by</code> | either values or labels |
| <code>decreasing</code> | sort decreasing |

Value

Returns x with sorted value labels

| | |
|------------------------|--------------------------------|
| <code>i_subject</code> | <i>add subject to variable</i> |
|------------------------|--------------------------------|

Description

add subject to i_labelled object

Usage

```
i_subject(x, subject)
```

Arguments

| | |
|----------------------|---|
| <code>x</code> | vector |
| <code>subject</code> | variable label as string or NULL (NULL will remove label) |

Value

x with subject applied

| | |
|---------|--|
| i_table | <i>cross tabulation and table creation using i_labelled labels</i> |
|---------|--|

Description

wrapper for base::table
 convert i_labelled objects to base class and pass to table function

Usage

```
i_table(..., missing_to_na = TRUE, as_factor = TRUE, table_args = NULL)
```

Arguments

| | |
|---------------|--|
| ... | one or more atomic vectors or one data.frame |
| missing_to_na | make as missing declared values NA |
| as_factor | make labelled data factor before pass to table |
| table_args | arguments of base::table as named list |

Value

returns a contingency table, an object of class "table"

Examples

```
set.seed(1234)
a <- sample(c(1:3, NA), 10, replace = TRUE)
b <- i_labelled(sample(c(1:3, NA), 10, replace = TRUE), labels = c("A" = 1, "B" = 2, "C" = 3))
c <- factor(sample(c("X", "Y", "Z", NA), 10, replace = TRUE))
df <- data.frame(a, b, c)

i_table(a, b)
i_table(df, table_args = list(useNA = "ifany"))
```

| | |
|-----------------|--|
| i_to_base_class | <i>remove class i_labelled and return base R class</i> |
|-----------------|--|

Description

- when value labels for all values are available will return factor
- when value labels are missing will unclass i_labelled
- remove class i_labelled and return variable as base R class

Usage

```
i_to_base_class(  
  x,  
  missing_to_na = TRUE,  
  as_factor = TRUE,  
  keep_attributes = FALSE  
)
```

Arguments

| | |
|-----------------|---|
| x | vector or data.frame |
| missing_to_na | missing values will become regular NA |
| as_factor | convert to factor when value labels are available |
| keep_attributes | should attributes be preserved |

Value

Returns x coerced to R base class

| | |
|-----------|--------------------------|
| i_unclass | <i>unclass variables</i> |
|-----------|--------------------------|

Description

unclass variables

Usage

```
i_unclass(x, keep_attributes = FALSE)
```

Arguments

| | |
|-----------------|--------------------------------|
| x | vector or data.frame |
| keep_attributes | should attributes be preserved |

Value

x unclassified

i_valid_annotation *validate annotation*

Description

returns boolean when applied to vector
returns a named list when applied to data.frame

Usage

`i_valid_annotation(x)`

Arguments

x vector or data.frame

Value

T/F

i_valid_label *validate variable labels*

Description

returns boolean when applied to vector
returns a named list when applied to data.frame

Usage

`i_valid_label(x)`

Arguments

x vector or data.frame

Value

T/F

| | |
|-----------------------------|------------------------------|
| <code>i_valid_labels</code> | <i>validate value labels</i> |
|-----------------------------|------------------------------|

Description

returns boolean when applied to vector

returns a named list when applied to data.frame

Usage

```
i_valid_labels(x)
```

Arguments

x vector or data.frame

Value

No return value. Aborts process when run-time-tests fail

| | |
|----------------------------|--------------------------------------|
| <code>i_valid_scale</code> | <i>validate variable scale level</i> |
|----------------------------|--------------------------------------|

Description

returns boolean when applied to vector

returns a named list when applied to data.frame

Usage

```
i_valid_scale(x)
```

Arguments

x vector or data.frame

Value

T/F

| | |
|------------------------|-------------------------|
| <i>i_valid_subject</i> | <i>validate subject</i> |
|------------------------|-------------------------|

Description

returns boolean when applied to vector

returns a named list when applied to data.frame

Usage

```
i_valid_subject(x)
```

Arguments

x vector or data.frame

Value

T/F

| | |
|------------------------|-------------------------|
| <i>i_valid_wording</i> | <i>validate wording</i> |
|------------------------|-------------------------|

Description

returns boolean when applied to vector

returns a named list when applied to data.frame

Usage

```
i_valid_wording(x)
```

Arguments

x vector or data.frame

Value

T/F

| | |
|------------------------|--------------------------------|
| <code>i_wording</code> | <i>add wording to variable</i> |
|------------------------|--------------------------------|

Description

add wording to `i_labelled` object
can be used to store question text

Usage

```
i_wording(x, wording)
```

Arguments

| | |
|----------------------|---|
| <code>x</code> | vector |
| <code>wording</code> | variable label as string or NULL (NULL will remove label) |

Value

`x` with wording applied

| | |
|-------------------------------|---|
| <code>print.i_labelled</code> | <i>custom print method for i_labelled</i> |
|-------------------------------|---|

Description

custom print method for `i_labelled`

Usage

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

Arguments

| | |
|------------------|---|
| <code>x</code> | vector of class <code>i_labelled</code> |
| <code>...</code> | not used |

Value

No return value. Print object data and information to console

[.i_labelled *subsetting vectors of class i_labelled*

Description

subsetting vectors of class i_labelled

Usage

```
## S3 method for class 'i_labelled'  
x[...]
```

Arguments

x vector of class i_labelled
... not used

Value

Subset of x

Index

`.i_find_in`, 5
`.i_in`, 5
`.init`, 3
`.is_sequential`, 4
`.merge_labels`, 6
`.valid_annotation`, 6
`.valid_label`, 7
`.valid_labels`, 7
`.valid_missing`, 8
`.valid_scale`, 8
`.valid_subject`, 9
`.valid_wording`, 9
`[.i_labelled`, 41
`%in%` (grapesingrapes), 10
`%in%`, ANY, `i_labelled`-method
 (grapesingrapes), 10
`%in%`, `i_labelled`, ANY-method
 (grapesingrapes), 10
`%in%`, `i_labelled`, `i_labelled`-method
 (grapesingrapes), 10

`as.i_labelled`, 10

grapesingrapes, 10

`i_annotation`, 12
`i_as_character`, 13
`i_as_factor`, 13
`i_as_numeric`, 14
`i_assert_labels`, 12
`i_copy`, 15
`i_get_annotation`, 15
`i_get_equal_subject`, 16
`i_get_equal_wording`, 16
`i_get_label`, 17
`i_get_labels`, 17
`i_get_na_range`, 18
`i_get_na_values`, 18
`i_get_scale`, 19
`i_get_subject`, 19

`i_get_wording`, 20
`i_label`, 20
`i_labelled`, 21
`i_labels`, 22
`i_missing_to_na`, 22
`i_na_range`, 23
`i_na_values`, 23
`i_print_annotation`, 24
`i_print_label`, 24
`i_print_labels`, 25
`i_print_na_range`, 25
`i_print_na_values`, 26
`i_print_scale`, 26
`i_print_subject`, 27
`i_print_wording`, 27
`i_recode`, 28
`i_remove_annotation`, 29
`i_remove_label`, 29
`i_remove_labels`, 30
`i_remove_missing_labels`, 30
`i_remove_na_range`, 31
`i_remove_na_values`, 31
`i_remove_scale`, 32
`i_remove_subject`, 32
`i_remove_wording`, 33
`i_scale`, 33
`i_sort_labels`, 34
`i_subject`, 34
`i_table`, 35
`i_to_base_class`, 35
`i_unclass`, 36
`i_valid_annotation`, 37
`i_valid_label`, 37
`i_valid_labels`, 38
`i_valid_scale`, 38
`i_valid_subject`, 39
`i_valid_wording`, 39
`i_wording`, 40
`is.i_labelled`, 11

`is_decimal`, 11

`print.i_labelled`, 40