

Package: tagr (via r-universe)

August 28, 2024

Title Tagging and Organizing Objects in R

Version 1.0.1

Maintainer Joachim Zuckarelli <joachim@zuckarelli.de>

Description Provides functions for attaching tags to R objects, searching for objects based on tags, and removing tags from objects. It also includes a function for removing all tags from an object, as well as a function for deleting all objects with a specific tag from the R environment. The package is useful for organizing and managing large collections of objects in R.

BugReports <https://github.com/jsugarelli/tagr/issues>

URL <https://github.com/jsugarelli/tagr/>

License GPL-3

Encoding UTF-8

LazyData true

RoxygenNote 7.1.1

Repository <https://jsugarelli.r-universe.dev>

RemoteUrl <https://github.com/jsugarelli/tagr>

RemoteRef HEAD

RemoteSha 4769e5ee7a3dae65997a0ebe76e15082446c957f

Contents

add_tags	2
clean_tags	2
has_tag	3
ls_bytag	4
rm_bytag	5
set_confirm	6
tags	7
untag	7

Index	9
--------------	----------

add_tags	<i>Add tags to an R object</i>
----------	--------------------------------

Description

This function adds tags to an existing R object. The tags are stored as attributes of the object and can be later used to identify and manipulate the object. If the specified object does not exist in the specified environment, an error is thrown.

Usage

```
add_tags(x, ..., envir = parent.frame())
```

Arguments

x	The name of the object to tag (unquoted).
...	The tags to add to the object.
envir	The environment in which the object exists (defaults to the parent frame).

Value

No return value.

Examples

```
# create a vector and add some tags
x <- c(1, 2, 3)
add_tags(x, "foo", "bar")
```

clean_tags	<i>Remove the 'tags' attribute from an object</i>
------------	---

Description

This function removes the 'tags' attribute from an object, effectively removing all tags attached to the object. If the object does not have a 'tags' attribute, the function simply returns without modifying the object.

Usage

```
clean_tags(obj, envir = parent.frame())
```

Arguments

obj	The object to remove tags from
envir	The environment in which to look for the object

Value

No return value.

Examples

```
# create an object and add some tags
my_vec <- c(1, 2, 3)
add_tags(my_vec, "important", "numeric")

# remove the tags from the object
clean_tags(my_vec)
```

has_tag

Check if an R object has specified tags

Description

This function checks if an existing R object has specified tags. The tags are stored as attributes of the object. If the specified object does not exist in the specified environment, an error is thrown.

Usage

```
has_tag(x, ..., envir = parent.frame())
```

Arguments

x	The name of the object to check (unquoted).
...	The tags to check for.
envir	The environment in which the object exists (defaults to the parent frame).

Value

A logical value indicating whether the object has all of the specified tags.

Examples

```
# create a vector and add some tags
x <- c(1, 2, 3)
add_tags(x, "foo", "bar")

# check if the vector has the specified tags
has_tag(x, "foo", "bar")
```

`ls_bytag`*List objects in an environment by tag*

Description

This function lists all objects in a specified environment that have all of the specified tags. The tags are stored as attributes of the objects. If the `only.tags` argument is set to `TRUE`, only objects that have at least one tag are listed. If no objects with the specified tags are found, a message is printed to the console. If the specified environment does not exist, an error is thrown.

Usage

```
ls_bytag(..., only.tags = TRUE, envir = parent.frame())
```

Arguments

<code>...</code>	The tags to filter by.
<code>only.tags</code>	Logical value indicating whether to include only objects that have at least one tag.
<code>envir</code>	The environment to search for tagged objects (defaults to the parent frame).

Value

No return value. A table with information about each tagged object, including its name, type, and tags is printed to the console.

Examples

```
# create some objects and add tags
x <- c(1, 2, 3)
y <- matrix(1:9, nrow = 3)
z <- "hello world"
add_tags(x, "foo")
add_tags(y, "bar")
add_tags(z, "baz")

# list objects with specified tags
ls_bytag("foo", "bar")
```

rm_bytag	<i>Remove Objects by Tag</i>
----------	------------------------------

Description

Remove all objects in the current environment that have a specified tag.

Usage

```
rm_bytag(  
  tag,  
  envir = parent.frame(),  
  confirm = getOption("tagsr.confirm", FALSE)  
)
```

Arguments

tag	The tag to search for.
envir	The environment to search in. Defaults to the current environment.
confirm	If TRUE, a confirmation prompt will be displayed before objects are deleted.

Value

No return value.

See Also

[add_tags](#), [has_tag](#), [ls_bytag](#)

Examples

```
# create some objects with tags  
x <- 1:10  
y <- matrix(rnorm(16), 4, 4)  
add_tags(x, "numbers")  
add_tags(y, "matrix")  
  
# remove all objects with the "numbers" tag  
rm_bytag("numbers")  
  
# remove all objects with the "matrix" tag without confirmation prompt  
set_confirm(FALSE)  
rm_bytag("matrix")  
  
# confirm that objects have been removed  
ls()  
  
# clean up
```

```
rm(set_confirm, x, y)
```

set_confirm

Set confirmation prompt for deleting objects

Description

This function sets the confirmation prompt for the `rm_bytag` function when deleting objects with a specified tag.

Usage

```
set_confirm(confirm = NULL)
```

Arguments

confirm	A logical value indicating whether to prompt for confirmation before deleting objects. If TRUE, a confirmation prompt will be displayed. If FALSE, no confirmation prompt will be displayed. If NULL, the current setting will be returned without changing it.
---------	---

Value

If confirm is NULL, the current setting for the confirmation prompt is returned. Otherwise, the confirmation prompt setting is updated.

Examples

```
# turn on confirmation prompt
set_confirm(TRUE)

# turn off confirmation prompt
set_confirm(FALSE)

# get current confirmation prompt setting
set_confirm()
```

tags	<i>Retrieve tags associated with an object</i>
------	--

Description

This function retrieves the tags associated with a specified object.

Usage

```
tags(obj, envir = parent.frame())
```

Arguments

obj	The object to retrieve tags from.
envir	The environment in which the object exists. Defaults to the parent environment.

Value

Returns a sorted vector of tags associated with the object. If the object has no tags, the function prints a message and returns NULL.

Examples

```
# create a variable
x <- 5

# add tags to the variable
add_tags(x, "important", "numeric")

# retrieve the tags
tags(x)
```

untag	<i>Remove tags from an object</i>
-------	-----------------------------------

Description

This function removes one or more tags from an object. The tags to be removed can be specified as separate arguments or using the ellipsis ('...') syntax. Alternatively, setting 'all' to TRUE will remove all tags from the object. If the specified object does not have the specified tag(s), the function will throw an error. If 'all' is set to TRUE and the object has no tags, the function will do nothing.

Usage

```
untag(obj, ..., all = FALSE, envir = parent.frame())
```

Arguments

obj	an object in the current environment
...	one or more character strings specifying the tags to remove
all	a logical value indicating whether to remove all tags from the object
envir	the environment in which the object is defined

Value

No return value.

Examples

```
x <- 1:10
add_tags(x, "numbers", "positive")
add_tags(x, "even")
tags(x)
# "even", "numbers", "positive"

# Remove the "positive" tag from x
untag(x, "positive")
tags(x)
# "even", "numbers"

# Remove all tags from x
untag(x, all = TRUE)
tags(x)
# "NULL"
```


Index

- * **tag**
 - untag, 7
- add_tags, 2, 5
- clean_tags, 2
- has_tag, 3, 5
- ls_bytag, 4, 5
- rm_bytag, 5, 6
- set_confirm, 6
- tags, 7
- untag, 7