

Package: available (via r-universe)

July 2, 2024

Title Check if the Title of a Package is Available, Appropriate and Interesting

Version 1.1.0.9000

Description Check if a given package name is available to use. It checks the name's validity. Checks if it is used on 'GitHub', 'CRAN' and 'Bioconductor'. Checks for unintended meanings by querying 'Wiktionary' and Wikipedia.

License MIT + file LICENSE

URL <https://github.com/r-lib/available>,
<https://r-lib.github.io/available/>

BugReports <https://github.com/r-lib/available/issues>

Depends R (>= 3.6)

Imports cli, clisymbols, crayon, desc, glue, jsonlite, memoise, SnowballC, stringdist, tibble, tidytext, yesno

Suggests BiocManager, covr, testthat (>= 3.0.0), usethis

Config/Needs/website tidyverse/tidytemplate

Config/testthat/edition 3

Encoding UTF-8

Roxygen list(markdown = TRUE)

RoxygenNote 7.2.3

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

RemoteUrl <https://github.com/r-lib/available>

RemoteRef HEAD

RemoteSha 079e49df3ac0d8ecf986787cdfcabbebd6a0ee15

Contents

available	2
available_on_bioc	3

available_on_github	3
create	4
get_bad_words	4
get_sentiment	5
get_wikipedia	5
pkg_name_dist	5
suggest	6
valid_package_name	6

Index	8
--------------	----------

available	<i>See if a name is available</i>
-----------	-----------------------------------

Description

Searches performed

- Valid package name
- Already taken on CRAN
- Positive or negative sentiment

Usage

```
available(name, browse = getOption("available.browse", TRUE), ...)
```

Arguments

name	Name of package to search
browse	Whether browser should be opened for all web links, default = TRUE. Default can be changed by setting <code>available.browse</code> in <code>.Rprofile</code> . See Startup for more details.
...	Additional arguments passed to <code>utils::available.packages()</code> .

Examples

```
## Not run:
# Check if the available package is available
available("available")

# You can disable opening of browser windows with browse = FALSE
available("survival", browse = FALSE)

# Or by setting a global option
options(available.browse = FALSE)
available("survival")

# Test if a name is available in a non-default CRAN repository by setting
```

```
# the `repos` argument
available_on_cran("semaforos")

available_on_cran("semaforos", repos = "https://bisaloo.r-universe.dev")

## End(Not run)
```

available_on_bioc *See if a name is available on CRAN*

Description

See if a name is available on CRAN

Usage

```
available_on_bioc(name, repos = NULL, ...)

available_on_cran(name, repos = default_cran_repos, ...)
```

Arguments

name	Name of package to search
repos	character vector, the base URL(s) of the repositories to use.
...	Additional arguments passed to <code>utils::available.packages()</code> .

Examples

```
available_on_cran("semaforos")

# Test if this name is available in a non-default CRAN repository
## Not run:
available_on_cran("semaforos", repos = "https://bisaloo.r-universe.dev")

## End(Not run)
```

available_on_github *See if a name is available on github*

Description

See if a name is available on github

Usage

```
available_on_github(name)
```

Arguments

name Name of package to search

create *Check a new package name and possibly create it*

Description

Check a new package name and possibly create it

Usage

```
create(name, ...)
```

Arguments

name Name of package to search
... Additional arguments passed to `usethis::create_package()`.

get_bad_words *Check for bad words in name*

Description

Check for bad words in name

Usage

```
get_bad_words(name)
```

Arguments

name Name of package to search

See Also

See <https://github.com/web-mech/badwords>

get_sentiment	<i>Get sentiment of name</i>
---------------	------------------------------

Description

Get sentiment of name

Usage

```
get_sentiment(name)
```

Arguments

name	Name of package to search
------	---------------------------

get_wikipedia	<i>Open wikipedia page and abbreviations.com page</i>
---------------	---

Description

Open wikipedia page and abbreviations.com page

Usage

```
get_wikipedia(name)
```

```
get_wiktionary(name)
```

Arguments

name	Name of package to search
------	---------------------------

pkg_name_dist	<i>Find five closest package names in terms of string distance</i>
---------------	--

Description

Find five closest package names in terms of string distance

Usage

```
pkg_name_dist(name, pkgs)
```

Arguments

name	name of package
pkgs	packages to compare with

suggest	<i>Suggest a package name based on a development package title or description</i>
---------	---

Description

If the package you are using already has a title, simply pass the path to the package root in `path`. Otherwise use `title` to specify a potential title.

Usage

```
suggest(path = ".", field = c("Title", "Description"), text = NULL)
```

Arguments

<code>path</code>	Path to a existing package to extract the title from.
<code>field</code>	one of "Title" or "Description"
<code>text</code>	text string to search.

Examples

```
## Not run:
# Default will use the title from the current path.
suggest()

# Can also suggest based on the description
suggest(field = "Description")

## End(Not run)

# Or by explicitly using the text argument
suggest(
  text =
    "A Package for Displaying Visual Scenes as They May Appear to an Animal with Lower Acuity"
)
```

valid_package_name	<i>Is a package name valid</i>
--------------------	--------------------------------

Description

Is a package name valid

Usage

```
valid_package_name(name)
```

valid_package_name

7

Arguments

name Name of package to search

Index

available, [2](#)
available_on_bioc, [3](#)
available_on_cran (available_on_bioc), [3](#)
available_on_github, [3](#)

create, [4](#)

get_bad_words, [4](#)
get_sentiment, [5](#)
get_wikipedia, [5](#)
get_wiktory (get_wikipedia), [5](#)

pkg_name_dist, [5](#)

Startup, [2](#)
suggest, [6](#)

usethis::create_package(), [4](#)
utils::available_packages(), [2](#), [3](#)

valid_package_name, [6](#)