Package 'available'

October 7, 2018

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

Version 1.0.1

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 Urban Dictionary, 'Wiktionary' and Wikipedia.

Depends R (>= 3.3.0)

License MIT + file LICENSE

Encoding UTF-8

LazyData true

RoxygenNote 6.0.1

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

Suggests devtools, BiocInstaller, testthat

URL https://github.com/ropenscilabs/available

BugReports https://github.com/ropenscilabs/available/issues

NeedsCompilation no

Author Carl Ganz [aut],
Gábor Csárdi [aut],
Jim Hester [aut, cre],
Molly Lewis [aut],
Rachael Tatman [aut]

Maintainer Jim Hester <james.f.hester@gmail.com>

Repository CRAN

Date/Publication 2018-07-16 15:10:03 UTC

R topics documented:

---

available   *See if a name is available*

Description

Searches performed - Valid package name - Already taken on CRAN - Positive or negative sentiment - Urban Dictionary

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 available.browse in .Rprofile. See Startup for more details.
...	Additional arguments passed to [utils::available.packages()].

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")

## 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 [utils::available.packages()].

---

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 [devtools::create()].

---

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   *Get sentiment of name*

Description

Get sentiment of name

Usage

```
get_sentiment(name)
```

Arguments

name   Name of package to search

---

get_urban_data   *get urban dictionary definitions and tags*

Description

get urban dictionary definitions and tags

Usage

```
get_urban_data(name)
```

Arguments

name   Name of package to search

---

get_wikipidia *Open wikipedia page and abbreviations.com page*

Description

Open wikipedia page and abbreviations.com page

Usage

```
get_wikipidia(name)

get_wiktionary(name)
```

Arguments

name   Name of package to search

---

pkg_name_dist *Find five closest package names in terms of string distance*

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 *Suggest a package name based on a development package title or description*

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

path	Path to a existing package to extract the title from.
field	one of "Title" or "Description"
text	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 explictly 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   *Is a package name valid*

Description

Is a package name valid

Usage

```
valid_package_name(name)
```

Arguments

name	Name of package to search

Index