

Package: searcher (via r-universe)

September 18, 2024

Title Query Search Interfaces

Version 0.0.7

Description Provides a search interface to look up terms on 'Google', 'Bing', 'DuckDuckGo', 'Startpage', 'Ecosia', 'rseek', 'Twitter', 'StackOverflow', 'RStudio Community', 'GitHub', and 'BitBucket'. Upon searching, a browser window will open with the aforementioned search results.

URL <https://github.com/coatless-rpkg/searcher>,
<https://r-pkg.thecoatlessprofessor.com/searcher/>

BugReports <https://github.com/coatless-rpkg/searcher/issues>

Depends R (>= 3.3.0)

License GPL (>= 2)

Encoding UTF-8

RoxygenNote 7.3.1

Roxygen list(markdown = TRUE)

Suggests testthat (>= 2.1.0), covr, knitr, rmarkdown

VignetteBuilder knitr

Repository <https://coatless-rpkg.r-universe.dev>

RemoteUrl <https://github.com/coatless-rpkg/searcher>

RemoteRef HEAD

RemoteSha f6ee41347dd3c34043235e9f1ffe79a085bd7ac7

Contents

searcher-package	2
searcher	3
searcher-defunct	4
search_site	4

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

searcher-package *searcher: Query Search Interfaces*

Description

Provides a search interface to look up terms on 'Google', 'Bing', 'DuckDuckGo', 'Startpage', 'Ecosia', 'rseek', 'Twitter', 'StackOverflow', 'RStudio Community', 'GitHub', and 'BitBucket'. Upon searching, a browser window will open with the aforementioned search results.

Package Customizations

searcher accesses a set of default values stored in `options()` on each call to keep the function signatures small. By default, these options are given as:

- `searcher.launch_delay`: Amount of time to remain in *R* before opening a browser window. Default is 0.5 seconds.
- `searcher.use_rstudio_viewer`: Display search results in the RStudio viewer pane instead of a web browser. Default is FALSE.
- `searcher.default_keyword`: Suffix keyword to generate accurate results between either "base" or "tidyverse". Default is "base".
- ...

Author(s)

Maintainer: James Balamuta <balamut2@illinois.edu> ([ORCID](#))

Other contributors:

- Alex Rossell Hayes <alexander@rossellhayes.com> ([ORCID](#)) [contributor]

See Also

Useful links:

- <https://github.com/coatless-rpkg/searcher>
- <https://r-pkg.thecoatlessprofessor.com/searcher/>
- Report bugs at <https://github.com/coatless-rpkg/searcher/issues>

`searcher`*Generate a Searcher function for use with Error Handling*

Description

Constructs a function object that will search the last R error message on search portals by opening a browser.

Usage

```
searcher(site, keyword = getOption("searcher.default_keyword"))
```

Arguments

<code>site</code>	Name of site to search on. Supported options: "google" (default), "bing", "duckduckgo", "startpage", "qwant", "rstudio community", "twitter", "stackoverflow", "github", "grep", and "bitbucket".
<code>keyword</code>	Opt to search under different default terms.

Details

This function acts as a closure. Thus, you will receive a function back when only specifying the `site` parameter. To call the function, add a second set of parentheses.

Generic Error Search

The `searcher` function grabs the last error message and tries to search it. This function will ensure that R language is the primary search context.

Examples

```
### Manually
searcher("google")()

## Not run:
### Automatically
# On error, automatically search the message on google
options(error = searcher("google"))

## End(Not run)
```

searcher-defunct *Defunct functions in searcher*

Description

Functions listed below are no longer included in the searcher package

Details

- `search_ixquick()`: The function binding was removed as the search engine name changed to "Startpage". Please use `search_startpage()`.

search_site *Search a Query on a Search Portal in a Web Browser*

Description

Creates an appropriate query string for a search engine and then opens up the resulting page in a web browser.

Usage

```
search_site(
  query,
  site = c("google", "bing", "duckduckgo", "ddg", "startpage", "sp", "qwant", "rseek",
    "rstudio community", "rscm", "twitter", "stackoverflow", "so", "github", "gh",
    "grep", "bitbucket", "bb"),
  rlang = TRUE
)

search_google(query = geterrmessage(), rlang = TRUE)

search_bing(query = geterrmessage(), rlang = TRUE)

search_duckduckgo(query = geterrmessage(), rlang = TRUE)

search_ddg(query = geterrmessage(), rlang = TRUE)

search_ixquick(query = geterrmessage(), rlang = TRUE)

search_startpage(query = geterrmessage(), rlang = TRUE)

search_sp(query = geterrmessage(), rlang = TRUE)

search_ecosia(query = geterrmessage(), rlang = TRUE)
```

```
search_rseek(query = geterrmessage(), rlang = TRUE)
search_qwant(query = geterrmessage(), rlang = TRUE)
search_rstudio_community(query = geterrmessage(), rlang = TRUE)
search_rscom(query = geterrmessage(), rlang = TRUE)
search_twitter(query = geterrmessage(), rlang = TRUE)
search_stackoverflow(query = geterrmessage(), rlang = TRUE)
search_so(query = geterrmessage(), rlang = TRUE)
search_github(query = geterrmessage(), rlang = TRUE)
search_gh(query = geterrmessage(), rlang = TRUE)
search_grep(query = geterrmessage(), rlang = TRUE)
search_bitbucket(query = geterrmessage(), rlang = TRUE)
search_bb(query = geterrmessage(), rlang = TRUE)
```

Arguments

query	Contents of string to search. Default is the error message.
site	Name of site to search on. Supported options: "google" (default), "bing", "duckduckgo", "startpage", "qwant", "rstudio community", "twitter", "stackoverflow", "github", "grep", and "bitbucket".
rlang	Search for results written in R. Default is TRUE

Value

The generated search URL or an empty string.

Google Search

The `search_google` function searches **Google** using: <https://www.google.com/search?q=<query>>
See <https://moz.com/learn/seo/search-operators> for details.

Bing Search

The `search_bing()` function searches **Bing** using: <https://www.bing.com/search?q=<query>>

DuckDuckGo Search

The `search_duckduckgo()` and `search_ddg()` functions both search **DuckDuckGo** using: <https://duckduckgo.com/?q=>

Startpage Search

The `search_startpage()` function searches **startpage** using: <https://startpage.com/do/dsearch?query=<query>>

Ecosia Search

The `search_ecosia()` function searches Ecosia using: <https://www.ecosia.org/search?q=<query>>

For additional details regarding Ecosia's search interface please see: <https://ecosia.helpscoutdocs.com/article/502-ecosia-on-desktop>

Rseek Search

The `search_rseek()` function searches **Rseek** using: <https://rseek.org/?q=<query>>

Qwant Search

The `search_qwant()` function searches Qwant using: <https://www.qwant.com/?q=<query>>

RStudio Community Search

The `search_rstudio_community()` and `search_rscom()` functions both search **RStudio Community** using: <https://community.rstudio.com/search?q=<query>>

For additional details regarding **RStudio Community**'s search interface please see the **Discourse** API documentation: <https://docs.discourse.org/#tag/Search>

Twitter Search

The `search_twitter()` functions search Twitter using: <https://twitter.com/search?q=<query>>

For additional details regarding Twitter's search interface please see: <https://help.twitter.com/en/using-x/x-advanced>

StackOverflow Search

The `search_stackoverflow()` and `search_so()` functions both search **StackOverflow** using: <https://stackoverflow.com>

For additional details regarding **StackOverflow**'s search interface please see: <https://stackoverflow.com/help/searching>

GitHub Search

The `search_github()` and `search_gh()` functions both search **GitHub** using: <https://github.com/search?q=<query>+1>

For additional details regarding **GitHub**'s search interface please see: <https://docs.github.com/en/enterprise-cloud@latest/search-github/getting-started-with-searching-on-github/about-searching-on-github> and <https://docs.github.com/en/search-github/searching-on-github/searching-code/>

grep.app Search

The `search_grep()` function searches all public code on **GitHub** using **grep.app** by querying: [https://grep.app/search?q=<query-here>&filter\[lang\]\[0\]=R](https://grep.app/search?q=<query-here>&filter[lang][0]=R)

BitBucket Search

The `search_bitbucket()` and `search_bb()` functions both search **BitBucket** using: <https://bitbucket.org/search?q=>

For additional details regarding **BitBucket**'s search interface please see: <https://confluence.atlassian.com/bitbucket/code-search-in-bitbucket-873876782.html>

See Also

[search_google\(\)](#), [search_bing\(\)](#), [search_duckduckgo\(\)](#), [search_startpage\(\)](#), [search_rseek\(\)](#), [search_qwant\(\)](#), [search_twitter\(\)](#), [search_rstudio_community\(\)](#), [search_stackoverflow\(\)](#), [search_github\(\)](#), [search_grep\(\)](#), [search_bitbucket\(\)](#), and [searcher\(\)](#)

Examples

```
# Search in a generic way
search_site("r-project", "google")

# Search Google
search_google("r-project")

# Search Bing
search_bing("Microsoft R")

# Search DuckDuckGo
search_duckduckgo("R language")

# Search startpage
search_startpage("VS Code")

# Search Rseek
search_rseek("searcher")

# Search Qwant
search_qwant("Quarto")

# Search RStudio Community
search_rstudio_community("RStudio IDE")

# Search Twitter
search_twitter("searcher")

# Search StackOverflow for Convolutions in the r tag
search_stackoverflow("convolutions")

# Search all languages on StackOverflow for convolutions
search_stackoverflow("convolutions", rlang = FALSE)

# Search GitHub Issues for bivariate normal in the language:r
search_github("bivariate normal")

# Search all languages on GitHub Issues for bivariate normal
search_github("bivariate normal", rlang = FALSE)
```

```
# Search R code on GitHub for numerical optimization
search_grep("optim")

# Search all code on GitHub for numerical optimization
search_grep("optim", rlang = FALSE)

# Search BitBucket for assertions
search_bitbucket("assertions")

## Not run:
# On error, automatically search the message on google
options(error = searcher("google"))
options(error = search_google)

## End(Not run)
```


Index

options(), 2

search_bb (search_site), 4
search_bing (search_site), 4
search_bing(), 7
search_bitbucket (search_site), 4
search_bitbucket(), 7
search_ddg (search_site), 4
search_duckduckgo (search_site), 4
search_duckduckgo(), 7
search_ecosia (search_site), 4
search_gh (search_site), 4
search_github (search_site), 4
search_github(), 7
search_google (search_site), 4
search_google(), 7
search_grep (search_site), 4
search_grep(), 7
search_ixquick (search_site), 4
search_ixquick(), 4
search_qwant (search_site), 4
search_qwant(), 7
search_rscom (search_site), 4
search_rseek (search_site), 4
search_rseek(), 7
search_rstudio_community (search_site),
4
search_rstudio_community(), 7
search_site, 4
search_so (search_site), 4
search_sp (search_site), 4
search_stackoverflow (search_site), 4
search_stackoverflow(), 7
search_startpage (search_site), 4
search_startpage(), 4, 7
search_twitter (search_site), 4
search_twitter(), 7
searcher, 3
searcher(), 7
searcher-defunct, 4

searcher-package, 2