

# Package ‘packagefinder’

October 26, 2020

**Type** Package

**Title** Comfortable Search for R Packages on CRAN, Either Directly from the R Console or with an R Studio Add-in

**Version** 0.3.2

**Maintainer** Joachim Zuckarelli <joachim@zuckarelli.de>

**Description** Search for R packages on CRAN directly from the R console, based on the packages' titles, short and long descriptions, or other fields. Combine multiple keywords with logical operators ('and', 'or'), view detailed information on any package and keep track of the latest package contributions to CRAN. If you don't want to search from the R console, use the comfortable R Studio add-in.

**License** GPL-3

**Encoding** UTF-8

**BugReports** <https://github.com/jsugarelli/packagefinder/issues>

**URL** <https://github.com/jsugarelli/packagefinder/>,  
<http://www.zuckarelli.de/packagefinder/tutorial.html>,  
<https://youtu.be/B96NMSo3nJI>

**Repository** CRAN

**LazyData** true

**ByteCompile** true

**Imports** httr, utils, jsonlite, pander, formattable, stringr, crayon,  
lubridate, tidyr, textutils, htmlTable, clipr, shiny,  
reactable, shinyjs, shinybusy, rstudioapi

**Depends** tools (>= 3.4.0)

**RoxygenNote** 7.1.1

**NeedsCompilation** no

**Author** Joachim Zuckarelli [aut, cre] (<<https://orcid.org/0000-0002-9280-3016>>)

**Date/Publication** 2020-10-26 07:50:02 UTC

## R topics documented:

buildIndex . . . . .	2
exploreFields . . . . .	3
findPackage . . . . .	4
fp . . . . .	7
go . . . . .	7
lastResults . . . . .	8
packageDetails . . . . .	9
packagefinder . . . . .	9
whatsNew . . . . .	10

<b>Index</b>	<b>12</b>
--------------	-----------

---

buildIndex	<i>Creating a search index for findPackage()</i>
------------	--

---

### Description

Builds a search index that can be used with [findPackage\(\)](#).

### Usage

```
buildIndex(filename = "", download.stats = FALSE)
```

### Arguments

filename	Name of .rdata file to which the new index is saved.
download.stats	Indicates if the search index shall include download figures from the RStudio CRAN mirror logs.

### Details

Functions like [findPackage\(\)](#) or [go\(\)](#) require a search index. This search index can either be created on-the-fly or be provided as a separate argument. In the latter case a search index can be built using [buildIndex\(\)](#). This index can include download figures for the packages (this feature is turned on with `download.stats = TRUE`). Including download stats requires significant time (may well be > 1 hour) for the index to be built. Therefore, when [findPackage\(\)](#) is called without providing an index, the index that is created on-the-fly does not contain CRAN download figures.

### Value

Returns the search index. As a side effect, the index is saved to a file, if a filename is provided via the filename argument.

### Author(s)

Joachim Zuckarelli <[joachim@zuckarelli.de](mailto:joachim@zuckarelli.de)>

**Examples**

```
index <- buildIndex(filename = file.path(tempdir(), "searchindex.rdata"), download.stats = FALSE)
```

---

exploreFields      *Searching for packages on CRAN*

---

**Description**

Searches for packages on CRAN by scanning a specified set of information fields for a user-provided search term.

**Usage**

```
exploreFields(
  term,
  fields = c("Name", "Description", "LongDescription"),
  mode = "or",
  match = "like",
  display = "viewer",
  index = NULL
)
```

**Arguments**

term	Search term to look for; character vector must have one element.
fields	The list of fields to be scanned for the search term; must be a character vector with one or more field names. Allowed field names are: "Name", "Description", "LongDescription", "Maintainer", "Authors@R", "Author", "License", "Imports", "Enhances", "Depends", "Suggests", "Reverse depends", "Reverse suggests", "Reverse enhances", "Copyright", "Contact", "Note", "MailingList".
mode	Indicates whether matches in the field shall be combined with a logical OR or with a logical AND; accordingly, permitted values are "or" (default) and "and". In "or" mode, every package that has the search term in any of the fields from fields generates a hit, in "and" mode the search term must be found in all fields provided to make that package a search hit.
match	Either "like" (default) or "exact". Determines if the field content must match the search term exactly or only needs to contain it.
display	Describes where the search results shall be shown. Either "viewer" or "console". If "viewer", the results are shown in RStudio's Viewer pane if the RStudio IDE is being used). If results = "console" the search results are shown as a text table in the R console.
index	Either a path (or URL) to a search index, or a search index that is already loaded. If no index is provided, exploreFields() creates an ad hoc search index.

**Details**

Found packages are listed in alphabetical order, there is no prioritization of search hits as in `findPackage()`.

**Value**

No return value.

**Author(s)**

Joachim Zuckarelli <joachim@zuckarelli.de>

**Examples**

```
exploreFields("Hadley", c("Maintainer", "Authors@R", "Author"))
```

---

findPackage

*Searching for packages on CRAN*

---

**Description**

Searches for packages on CRAN based on the user's specification of search terms. Considers the package name, description as well as the long description, and prioritizes the results.

**Usage**

```
findPackage(  
  keywords = NULL,  
  query = NULL,  
  mode = "or",  
  case.sensitive = FALSE,  
  always.sensitive = NULL,  
  weights = c(2, 2, 1, 2),  
  display = "viewer",  
  results.longdesc = FALSE,  
  limit.results = 15,  
  silent = FALSE,  
  index = NULL,  
  advanced.ranking = TRUE,  
  return.df = FALSE,  
  clipboard = FALSE  
)
```

**Arguments**

keywords	A vector of keywords to be searched for. Instead of separate search terms, keywords can also be a query like "meta AND regression". In this case the mode argument is ignored. Only one type of logical operator (either and or and) may be used in a query; operators are not case-sensitive.
query	A vector of regular expressions (regex) to match against package name and descriptions; alternative to keywords, if both are provided the regular expressions search will prevail, and keywords will be ignored.
mode	Indicates whether the search terms in keywords shall be combined with a logical OR or with a logical AND; accordingly, permitted values are "or" (default) and "and". In "or" mode, every package that contains <i>at least one</i> of the keywords from the keywords argument is a search hit, in "and" mode generating a search hit requires <i>all</i> search terms from the keywords argument to be found.
case.sensitive	Indicates if the search shall be case sensitive, or not.
always.sensitive	A vector of search terms for which capitalization is always considered relevant (even if case.sensitive = FALSE). This allows to better reflect abbreviations like 'GLM'.
weights	A numeric vector describing how search hits in different fields of the a package's data shall be weighted. The first three elements of the vector are the weights assigned to hits in the package's <i>title</i> , <i>short description</i> and <i>long description</i> , respectively. The fourth element is a factor applied to the overall score of a search hit if all search terms from the keywords argument are found (obviously only meaningful in "or" mode). All weights must be 1 or larger.
display	Describes where the search results shall be shown. Either "viewer", "console" or "browser". If "viewer", the results are shown in RStudio's Viewer pane if the RStudio IDE is being used. If results = "console" the search results are shown as a text table in the R console. results = "browser" shows the search results in the web browser.
results.longdesc	Indicates whether the packages' long descriptions shall also be included in the search results. Given the length of some long descriptions this may make the search results harder to read.
limit.results	The maximum number of matches presented in the search results; choose a negative number to display all results
silent	Indicates whether any visible output is produced. Use silent = TRUE if you are only interested in getting the search results as a dataframe (with return.df = TRUE).
index	Either a path (or URL) to a search index, or a search index that is already loaded. If no index is provided, findPackage() creates an ad hoc search index.
advanced.ranking	Indicates if the ranking of search results shall be based on weights taking into account the inverse frequencies of the different search terms across all packages and the length of the matches relative to the texts they were found in. Usually,

using advanced ranking (`advanced.ranking = TRUE`, default) gives more relevant results, especially in "or" mode when the search terms differ strongly in their frequency of occurrence across packages.

`return.df` If TRUE, `findPackage()` returns a dataframe with the results, otherwise there is no return value. Default is FALSE.

`clipboard` If TRUE, `findPackage()` copies the results to the clipboard.

## Details

The GO column in the search results is an index number that can be used to address the found package easily with the `go()` function. The Total Downloads column in the search results gives the overall number of downloads of the respective package since its submission to CRAN. The number is based on the figures for the RStudio CRAN mirror server. This field is only provided if the search index contains download figures. Ad hoc indices (when `index = NULL`) never include download statistics. Please refer to `buildIndex()` for more details.

`fp()` is a shorter alias for `findPackage()`.

## Value

The search results as a dataframe, if `df.return = TRUE`.

## Author(s)

Joachim Zuckarelli <[joachim@zuckarelli.de](mailto:joachim@zuckarelli.de)>

## Examples

```
search <- c("regression", "meta")
findPackage(search)

findPackage(c("text", "tables"))

searchindex <- buildIndex()
findPackage(keywords=c("regression", "linear"), mode="and",
  always.sensitive="GLM", index=searchindex)

findPackage("meta and regression", display="console")

# Alternatively, show results in browser
# findPackage("meta and regression", display="browser")

my.results <- findPackage("meta AND regression")
```

---

fp	<i>Searching for packages on CRAN</i>
----	---------------------------------------

---

**Description**

Shorter alias for function [findPackage\(\)](#).

**Usage**

```
fp(...)
```

**Arguments**

... Arguments as in [findPackage\(\)](#).

**Author(s)**

Joachim Zuckarelli <[joachim@zuckarelli.de](mailto:joachim@zuckarelli.de)>

**Examples**

```
fp(c("meta", "regression"))
```

---

go	<i>Showing information about a package</i>
----	--

---

**Description**

Allows to inspect a package found with [findPackage\(\)](#) by showing detailed CRAN information on the package, opening its manual (PDF) or pulling up the package's website. Also allows to install the package right away.

**Usage**

```
go(package, where.to = "details", index = NULL)
```

**Arguments**

package Either the name of the package (capitalization does generally not matter) or the search result number shown in the results of [findPackage\(\)](#) (the number in the GO column).

where.to	Either "details" (default), "manual", "website", or "install". With "details", <code>go()</code> presents a set of CRAN information on the package, <code>where.to = "manual"</code> opens the package's PDF manual from the web and <code>where.to = "website"</code> pulls up the package's website(s), if any website is provided by the package maintainer. <code>where.to = "install"</code> installs the package (including dependencies).
index	Either a path (or URL) to a search index, or a search index that is already loaded. If no index is provided, <code>go()</code> creates an ad hoc search index.

**Details**

`go()` is made to inspect a package found with `findPackage()` and to decide whether or not this package serves the intended purposes.

**Value**

No return value.

**Author(s)**

Joachim Zuckarelli <joachim@zuckarelli.de>

---

lastResults

*Searching for packages on CRAN*


---

**Description**

Shows the results of the last search with `findPackage()`.

**Usage**

```
lastResults(display = "viewer")
```

**Arguments**

display	Describes where the search results shall be shown. Either "viewer", "console" or "browser". If "viewer", the results are shown in RStudio's Viewer pane if the RStudio IDE is being used. If <code>results = "console"</code> the search results are shown as a text table in the R console. <code>results = "browser"</code> shows the search results in the web browser.
---------	--

**Value**

No return value.

**Author(s)**

Joachim Zuckarelli <joachim@zuckarelli.de>



---

packageDetails      *Showing information about a package*

---

**Description**

Shows detailed CRAN information for a package.

**Usage**

```
packageDetails(package, brief = FALSE, show.tip = TRUE, index = NULL)
```

**Arguments**

package	Either the name of a package (capitalization does generally not matter) or the search result number shown in the results of <code>findPackage()</code> (the number in the GO column). Only one package is allowed here.
brief	If "TRUE", only title, short and long description as well as the maintainer of the package are shown, otherwise all available fields are displayed.
show.tip	If "TRUE", tips for getting additional information on the package are shown.
index	Either a path (or URL) to a search index, or a search index that is already loaded. If no index is provided, <code>packageDetails()</code> creates an ad hoc search index.

**Value**

No return value.

**Author(s)**

Joachim Zuckarelli <joachim@zuckarelli.de>

**Examples**

```
packageDetails("ggplot2")
```

---

packagefinder      *Package 'packagefinder'*

---

**Description**

Comfortable search for R packages on CRAN directly from the R console

### What is packagefinder? How does it benefit me?

Currently, there are more than 16,000 R package contributions on CRAN providing R with an unparalleled wealth of features. The downside of the large and increasing amount of packages is that it becomes increasingly difficult to find the right tools to tackle a specific problem. Unfortunately, CRAN does not provide any good search functionality.

**packagefinder** is designed to search for CRAN packages right from the R console. The philosophy behind this package is that R users like using the R console and need a tool to do their day-to-day-work on CRAN without leaving their normal workspace, the console. In fact, the idea is that with **packagefinder** you do not *need* to leave the R console to work with CRAN effectively.

**packagefinder** is developed to save you time and energy finding the right packages to work with, thereby making your work with R more productive.

### Where can I get more information?

- Contact the **author**:
  - E-mail: Joachim Zuckarelli (<joachim@zuckarelli.de>)
  - Twitter: [@jsugarelli](#)
- packagefinder **introduction**: [A Quick Tutorial](#)
- packagefinder on **GitHub**: <https://github.com/jsugarelli/packagefinder>
- packagefinder video tutorial on **YouTube**: <https://youtu.be/B96NMSO3nJI> (discusses version 0.1.5)

---

whatsNew

*Staying up-to-date on CRAN packages*

---

### Description

Shows information on the latest package additions to CRAN.

### Usage

```
whatsNew(last.days = 0, brief = TRUE, index = NULL)
```

### Arguments

<code>last.days</code>	The length of the period (in days) for which package additions to CRAN shall be presented. <code>last.days=0</code> means only today's additions are shown.
<code>brief</code>	Determines if all available package description fields shall be shown ( <code>brief=FALSE</code> ) or only a summary covering the most important fields ( <code>brief=TRUE</code> , the default).
<code>index</code>	Either a path (or URL) to a search index, or a search index that is already loaded. If no index is provided, <code>whatsNew()</code> creates an ad hoc search index.

### Value

Number of packages covered by the period specified in `last.days`.

*whatsNew*

11

**Author(s)**

Joachim Zuckarelli <joachim@zuckarelli.de>

**Examples**

```
whatsNew(last.days = 3)
```

# Index

buildIndex, [2](#), [6](#)

exploreFields, [3](#)

findPackage, [2](#), [4](#), [4](#), [6–9](#)  
fp, [6](#), [7](#)

go, [2](#), [7](#)

lastResults, [8](#)

packageDetails, [9](#)  
packagefinder, [9](#)

whatsNew, [10](#)