

# Package ‘felp’

May 8, 2026

**Type** Package

**Title** Functional Help for Functions, Objects, and Packages

**Version** 0.6.0

**Author** Atsushi Yasumoto [aut, cph, cre] (ORCID:  
<<https://orcid.org/0000-0002-8335-495X>>)

**Maintainer** Atsushi Yasumoto <atusy.rpkg@gmail.com>

**Description** Enhance R help system by fuzzy search and preview interface, pseudo-postfix operators, and more.  
The `?.` pseudo-postfix operator and the `?` prefix operator displays documents and contents (source or structure) of objects simultaneously to help understanding the objects.  
The `?p` pseudo-postfix operator displays package documents, and is shorter than help(package = foo).

**URL** <https://felp.atusy.net/>, <https://github.com/atusy/felp>

**BugReports** <https://github.com/atusy/felp/issues>

**License** MIT + file LICENSE

**Encoding** UTF-8

**RoxygenNote** 7.3.1

**Imports** prettycode, callr, curl, data.table, dplyr, htmltools,  
magrittr, matrixStats, memoise, miniUI, reactable, rstudioapi,  
rlang, shiny, stringi, utils

**Suggests** knitr, printr, testthat (>= 2.1.0),

**VignetteBuilder** knitr

**Language** en-US

**NeedsCompilation** no

**Repository** CRAN

**Date/Publication** 2024-09-25 12:00:05 UTC

## Contents

?	2
dummy	3
felp	3
fuzzyhelp	4
<b>Index</b>	<b>6</b>

---

?

*Functional help with ? operator*

---

### Description

Displays help and structure of an object, or help of a package. Two syntax sugars are added to enhance `utils::?`. One is `object?`, which works as if `?object`. Another is `package?p` which works as if `help(package = package)`

### Usage

```
`?`(e1, e2)
```

### Arguments

`e1` A topic of document. Refer to topic argument described in `utils::`?``.

`e2` `.` and `p` have special meanings as documented above. Otherwise, `e2` is same as type argument of `utils::`?``.

### Examples

```
# Identical to help(identity); print(identity)
?identity
identity?. # The same

# Identical to help(iris); str(iris)
?iris
iris?. # The same

# Identical to help(package = stats)
stats?p
```

---

dummy	<i>A dummy data for felp</i>
-------	------------------------------

---

**Description**

A dummy data for felp

**Usage**

```
dummy
```

**Format**

An object of class `list` of length 2.

---

felp	<i>Functional help which displays structure of an object in addition to help</i>
------	--

---

**Description**

Structure of object is returned by `str()`. For a function, its source is returned instead of `str()`.

**Usage**

```
felp(topic, package = NULL, ...)
```

**Arguments**

topic	usually, a <a href="#">name</a> or character string specifying the topic for which help is sought. A character string (enclosed in explicit single or double quotes) is always taken as naming a topic. If the value of <code>topic</code> is a length-one character vector the topic is taken to be the value of the only element. Otherwise <code>topic</code> must be a name or a <a href="#">reserved</a> word (if syntactically valid) or character string. See ‘Details’ for what happens if this is omitted.
package	a name or character vector giving the packages to look into for documentation, or <code>NULL</code> . By default, all packages whose namespaces are loaded are used. To avoid a name being deparsed use e.g. <code>(pkg_ref)</code> (see the examples).
...	Arguments passed on to <a href="#">utils::help</a>
<code>lib.loc</code>	a character vector of directory names of <code>R</code> libraries, or <code>NULL</code> . The default value of <code>NULL</code> corresponds to all libraries currently known. If the default is used, the loaded packages are searched before the libraries. This is not used for HTML help (see ‘Details’).

verbose logical; if TRUE, the file name is reported.  
 try.all.packages logical; see Note.  
 help\_type character string: the type of help required. Possible values are "text", "html" and "pdf". Case is ignored, and partial matching is allowed.

### Examples

```
# Identical to help(identity); print(identity)
felp(identity)

# Identical to help(iris); str(iris)
felp(iris)

# Identical to help(package = stats)
felp(package = stats)
```

---

fuzzyhelp

*Fuzzily Search Help and View the Selection*

---

### Description

Users no longer have to remember the exact name to find help, vignettes, and demo. A shiny gadget helps you to find a topic fuzzily. Click radio buttons to switch preview contents. Click "Done" or "Cancel" to close the widget. When background = FALSE, the "Done" button will also hook help, vignette, or demo, accordingly.

### Usage

```
fuzzyhelp(
  query = "",
  method = getOption("fuzzyhelp.method", "fzf"),
  background = getOption("fuzzyhelp.background", TRUE),
  viewer = shiny::paneViewer()
)
```

### Arguments

query	An initial query to search for the help system.
method	A fuzzy match method to use. Choices are "fzf" and "lv" (levenshtein). The method "lv" is faster but can be less accurate. The default value can be tweaked by options(fuzzyhelp.method = "lv").
background	Whether to run a shiny gadget in a background process. The default value is TRUE and can be changed by option(fuzzyhelp.background = FALSE).
viewer	Specify where the gadget should be displayed—viewer pane, dialog window, or external browser—by passing in a call to one of the <a href="#">viewer()</a> functions.

**Value**

If the background argument is TRUE, then the return value inherits from `callr::r_bg()`. Otherwise, NULL is returned.

**Note**

The default fuzzy match algorithm is a simplified version of <https://github.com/junegunn/fzf>. The implementation in this package excludes bonuses from relationship with matched characters and their previous characters.

**Examples**

```
if (FALSE) {  
  fuzzyhelp()  
}
```

# Index

- \* **datasets**
  - dummy, 3
- ?, 2
- dummy, 3
- felP, 3
- fuzzyhelp, 4
- name, 3
- reserved, 3
- utils::help, 3
- viewer(), 4