

Package: shrtcts (via r-universe)

July 2, 2024

Title Make Anything an RStudio Shortcut

Version 0.1.2

Description An easily customizable method for creating RStudio addins from arbitrary R code. Use your own functions or call functions from other packages. Your shortcuts are stored in a YAML file in your home directory.

License MIT + file LICENSE

URL <https://pkg.garrickadenbuie.com/shrtcts>,
<https://github.com/gadenbuie/shrtcts>

BugReports <https://github.com/gadenbuie/shrtcts/issues>

Imports fs, jsonlite, rappdirs, roxygen2, rstudioapi, utils, yaml

Suggests testthat (>= 2.1.0), knitr, rmarkdown

Encoding UTF-8

LazyData true

Roxygen list(markdown = TRUE)

RoxygenNote 7.1.1

VignetteBuilder knitr

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

RemoteUrl <https://github.com/gadenbuie/shrtcts>

RemoteRef main

RemoteSha 41051cf25ee8bfd66da45315013ade9d894435bd

Contents

add_rstudio_shortcuts	2
edit_shortcuts	5
list_shortcuts	6
paths	7

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

add_rstudio_shortcuts *Add Shortcuts to RStudio*

Description

Add up to 100 fully configurable shortcuts to RStudio. Store your shortcuts in `.shrtcts.yaml` in your R home or OS home directory. Then add `add_rstudio_shortcuts()` to your `~/.Rprofile` to load the shortcuts when starting R. Your shortcuts will automatically appear in the Addins window (if not, try restarting your R session one more time). Your shortcuts can be arbitrary functions written inline or functions from other packages. You can set their names and even assign keyboard shortcuts to your `shrtcts`. See detailed sections below.

Usage

```
add_rstudio_shortcuts(path = NULL, set_keyboard_shortcuts = FALSE)
```

```
example_shortcuts_yaml()
```

```
example_shortcuts_r()
```

Arguments

`path` The path to your `.shrtcts.yaml` file. If `NULL`, **shrtcts** will look in your R or OS home directory (via `fs::path_home_r()` or `fs::path_home()`). You can set this path via the global option `"shrtcts.path"`. For more information, see the help documentation on [paths](#).

`set_keyboard_shortcuts`

If `TRUE`, will attempt to set the RStudio keyboard shortcuts in `addins.json`.

Details

shrtcts R Format:

Use the following template to organize your `.shrtcts.R`. You can write each shortcut in regular R code, annotated with **roxygen2** inline documentation comments. The comment format uses standard roxygen2 formatting, with a few additional roxygen tags specifically for **shrtcts**

```
#' Say Something Nice
#'
#' A demo of cool things.
#'
#' @interactive
#' @shortcut Ctrl+Alt+P
praise::praise
```

roxygen2 Tags:

shrtcts recognizes the following roxygen tags. Tags are optional unless otherwise specified.

- `@title` (required): The name of the shortcut's addin in RStudio. The tag itself is not required, the first line of untagged text (`#' Say Something Nice above`) is interpreted as the title.
- A *function*, either exported from another package, e.g. `praise::praise` or as an anonymous or named function provided immediately below the roxygen2 comments section. (Function names are ignored if provided).
- `@description`: A description of the shortcut. Can be specified with the roxygen tag or it can be the first paragraph of untagged text after the title line.
- `@interactive`: Whether or not the shortcut's addin should be executed interactively. Non-interactive addins are run in the background, without alerting the user and without providing a mechanism for the user to cancel the function. If the shortcut is interactive and calls a function stored in another package, the code to execute the function will be displayed in the console, rather than the placeholder shortcut from **shrtcts**.
- `@id`: An integer id (< 100) used to link the shortcut to a specific placeholder function in **shrtcts**. For example, `#' @id 5` will link the provided shortcut to `shrtcts:::shortcut_05()`. This is particularly useful if you have a keyboard shortcut linked to your shortcut, although the need for this tag is mitigated by the `@shortcut` tag.
- `@shortcut`: A combination of keys to be used as a keyboard shortcut in RStudio. Keyboard shortcuts are only applied if `set_keyboard_shortcuts` is set when calling `add_rstudio_shortcuts()`. This option is disabled by default.

Where to Store Your Shortcuts:

Save your shortcuts R (or YAML) file as `.shrtcts.R` or `.shrtcts.yml` in your home directory or in the `.config` directory in your home directory — use `fs::path_home_r()` or `fs::path_home()` to locate your home directory. In other words: `~/ .config/ .shrtcts.R` or `~/ .shrtcts.yml`.

You can test that **shrtcts** correctly finds your shortcuts file – or confirm which file will be used by **shrtcts** – using `locate_shortcuts_source()`.

Install Your Shortcuts:

Run `add_rstudio_shortcuts()` to install your shortcuts. You'll need to restart your R session for RStudio to learn your shortcuts.

To also update your **shrtcts**-related keyboard shortcuts, set `set_keyboard_shortcuts = TRUE`. This will update the keyboard shortcuts stored in RStudio's `addins.json`, typically stored in `~/ .config/rstudio/keybindings` (≥ 1.3) or `~/ .R/rstudio/keybindings` (< 1.3). If this file is stored in a non-standard location in your setup, you can provide `set_keyboard_shortcuts` with the correct path to `addins.json`. Whenever **shrtcts** updates the shortcut keybindings, a complete restart of RStudio is required (hint: use `usethis:::restart_rstudio()`).

RStudio Keyboard Shortcuts:

Once you've setup an RStudio Addin via **shrtcts**, there are two ways to link the shortcut's addin to a keyboard shortcut.

You can verify and list the current shortcuts and their keyboard bindings with `list_shortcuts()`.

```
shrtcts::list_shortcuts()
```

```
##           name      addin shrtcts_keybinding rstudio_keybinding
## 1           10 random numbers shortcut_01           <NA>           <NA>
```

```
## 2 New Temporary R Markdown Document shortcut_02 Ctrl+Alt+Shift+T Ctrl+Alt+Shift+T
## 3 A Random Number Between 0 and 1 shortcut_03 <NA> <NA>
## 4 Say Something Nice shortcut_97 Ctrl+Alt+P Ctrl+Alt+P
```

Declare Keyboard Shortcuts in .shrtcts.R:

You can use the @shortcut tag to declare the shortcut in .shrtcts.R (or shortcut: in the YAML .shrtcts.yml).

To update the keyboard shortcuts (for shrtcts only!), set set_keyboard_shortcuts = TRUE when calling add_rstudio_shortcuts(). If you use this method, shortcuts set manually in RStudio will be overwritten, so you should choose one method or the other.

- .shrtcts.R


```
#' Say Something Nice
#'
#' @description A demo of cool things
#' @interactive
#' @shortcut Ctrl+Alt+P
praise::praise
```
- .shrtcts.yml


```
- Name: Say Something Nice
  Description: A demo of cool things
  Binding: praise::praise
  shortcut: Ctrl+Alt+P
  Interactive: true
```

A full restart of RStudio is required whenever **shrtcts** updates the shortcut keybindings. **shrtcts** only manages keybindings for its own addins, and it doesn't check for conflicting key combinations, so you may want to double check the RStudio menu.

If anything goes wrong, a backup of the keybindings are saved as addins.json.bak in the same folder where addins.json was found. Use location_addins_json() to find this file.

Setting Keyboard Shortcuts via RStudio Menus:

You can create a keyboard shortcut for the addin using the *Tools > Modify keyboard shortcuts* menu.

If you create a shortcut for an addin via the menu, it's a good idea to set the id of the shortcut.

You can set your keyboard shortcuts manually in your .shrtcts.R or .shrtcts.yml files, using the @shortcut tag or shortcut: item name.

shrtcts YAML format:

shrtcts initially provided a way to specify the shortcuts in a YAML file. This made sense because everything is YAML these days, so why not add yet another YAML config file to the mix? But writing R code inside YAML is, um, less than ideal. So it's no longer recommended, but it is still supported (for now). To convert existing shortcuts from YAML to the roxygen2 format, use the internal shrtcts::migrate_yaml2r() function.

Use the following template to organize your .shrtcts.yml. Each shortcut is a YAML list item with the following structure:

- Name: Make A Noise


```
Description: Play a short sound
Binding: beepr::beep
Interactive: true
```

```
id: 42
shortcut: Ctrl+Shift+B
```

Functions

- `example_shortcuts_yaml`: An example `.shrtcts.yaml` file.
- `example_shortcuts_r`: An example `.shrtcts.R` file.

See Also

[list_shortcuts\(\)](#)

Examples

```
# Add shortcuts to ~/.shrtcts.yaml (see help above)

# Add this to your ~/.Rprofile to automatically load shortcuts
if (interactive() && requireNamespace("shrtcts", quietly = TRUE)) {
  shrtcts::add_rstudio_shortcuts()
}
```

edit_shortcuts

Open the shrtcts Source File

Description

This helper function locates and opens (or returns the path to) the `.shrtcts.R` or `.shrtcts.yaml` file.

Usage

```
edit_shortcuts(open = TRUE, path = NULL)
```

Arguments

<code>open</code>	If TRUE and the shrtcts source file is found (see paths), then the file is opened via <code>file.edit()</code> . Otherwise, the path is returned.
<code>path</code>	The path to your <code>.shrtcts.yaml</code> file. If NULL, shrtcts will look in your R or OS home directory (via <code>fs::path_home_r()</code> or <code>fs::path_home()</code>). You can set this path via the global option <code>"shrtcts.path"</code> . For more information, see the help documentation on paths .

Value

The path to the `.shrtcts.R` or `.shrtcts.yaml` source file (invisibly if the file is opened).

list_shortcuts	<i>List Shortcuts and Their Current Key Bindings</i>
----------------	--

Description

Lists all shortcuts declared in `.shrtcts.R` or `.shrtcts.yml`. Also lists the currently assigned RStudio keyboard shortcuts.

Usage

```
list_shortcuts(path_shortcuts = NULL, path_addins_json = NULL)
```

Arguments

`path_shortcuts` Path to `.shrtcts.R` or `.shrtcts.yml`. See [locate_shortcuts_source\(\)](#) for more detail.

`path_addins_json` Path to RStudio's `addins.json` keybindings config file. See [locate_addins_json\(\)](#) for more detail.

Value

A data frame with the shortcut name, it's assigned addin placeholder function, the `shrtcts_keybinding` declared in the `shrtcts` source file, and the `rstudio_keybinding` currently assigned to the shortcut in RStudio.

See Also

[add_rstudio_shortcuts\(\)](#), [paths](#)

Examples

```
if (
  interactive() &&
  requireNamespace("rstudioapi", quietly = TRUE) &&
  rstudioapi::hasFun("versionInfo")
) {
  list_shortcuts()
}
```

paths	<i>Paths Used by shrtcts</i>
-------	------------------------------

Description

These functions help locate the paths used by **shrtcts** when looking for configuration files. You can use these functions to verify that **shrtcts** is locating the files where you expect them to be. See the **Options** section below for global options that can be used to override the default behavior of these functions.

Usage

```
locate_shortcuts_source(path = NULL, all = FALSE)
```

```
locate_addins_json(path = NULL, all = FALSE)
```

Arguments

path	The path to the file.
all	List all found files, otherwise only the first is returned.

Functions

- `locate_shortcuts_source`: Find the path to `.shrtcts.R` or `.shrtcts.yml`.
- `locate_addins_json`: Find the path to `addins.json`, the RStudio configuration file containing keyboard shortcuts for addins

Options

Use the following global options to set the default values for the following files:

- `shrtcts.path`: The path to `.shrtcts.R` or `.shrtcts.yml`
- `shrtcts.addins_json`: The path to the RStudio `addins.json` file used for setting keyboard shortcuts for addins.

Examples

```
if (interactive()) {  
  locate_shortcuts_source()  
  locate_addins_json()  
}
```

Index

`add_rstudio_shortcuts`, 2
`add_rstudio_shortcuts()`, 3, 6

`edit_shortcuts`, 5
`example_shortcuts_r`
 (`add_rstudio_shortcuts`), 2
`example_shortcuts_yaml`
 (`add_rstudio_shortcuts`), 2

`fs::path_home()`, 2, 3, 5
`fs::path_home_r()`, 2, 3, 5

`list_shortcuts`, 6
`list_shortcuts()`, 3, 5
`locate_addins_json (paths)`, 7
`locate_addins_json()`, 6
`locate_shortcuts_source (paths)`, 7
`locate_shortcuts_source()`, 3, 6

`paths`, 2, 5, 6, 7