

# Package Development:

## The Rest of the Owl

GitHub repo: [pos.it/pkg\\_dev\\_24](https://github.com/posit-it/pkg_dev_24)

Wifi: Posit Conf 2024 | conf2024

Discord: #workshop-pkg-dev

How to draw an owl

1.

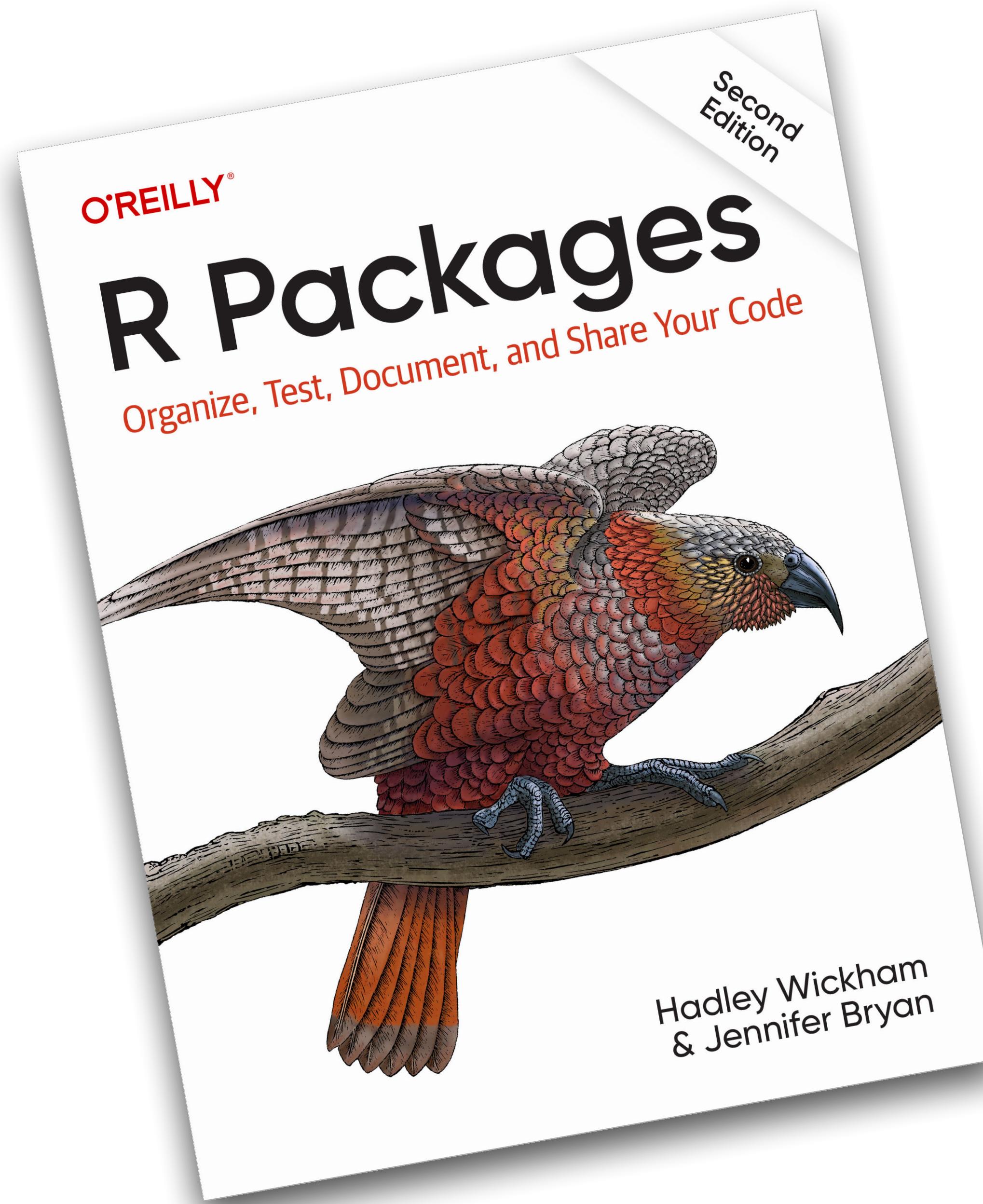


2.

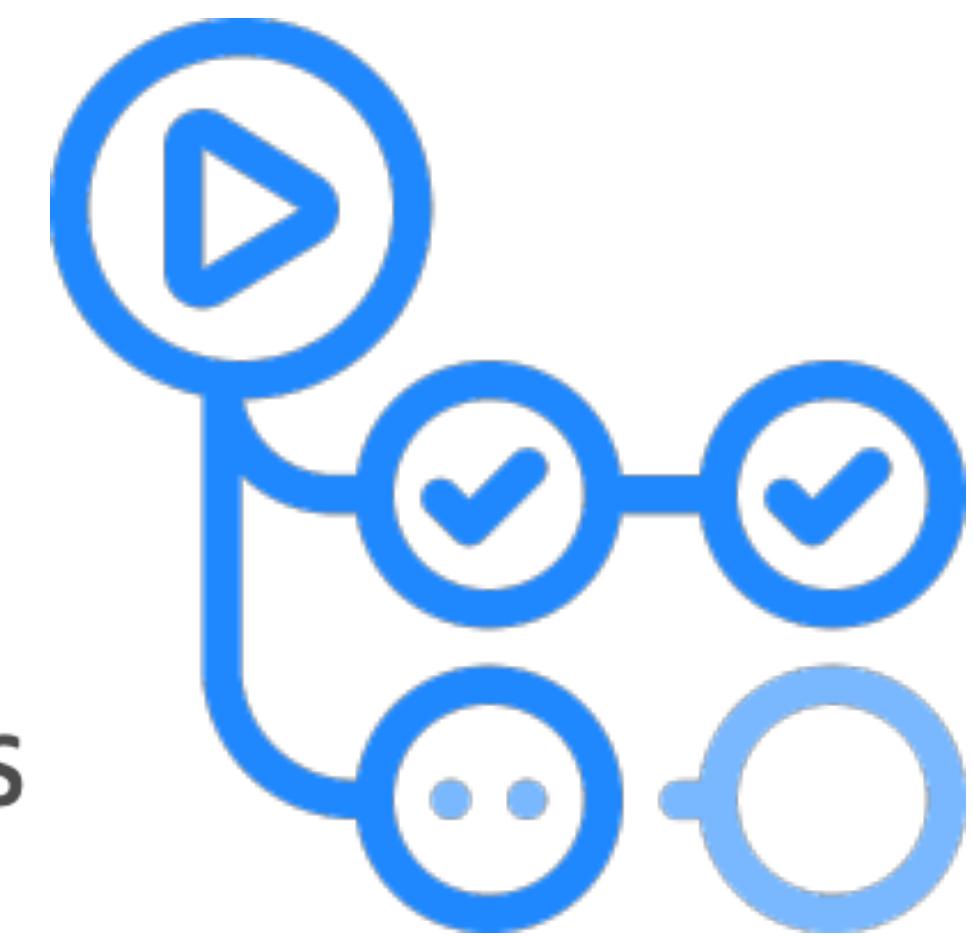
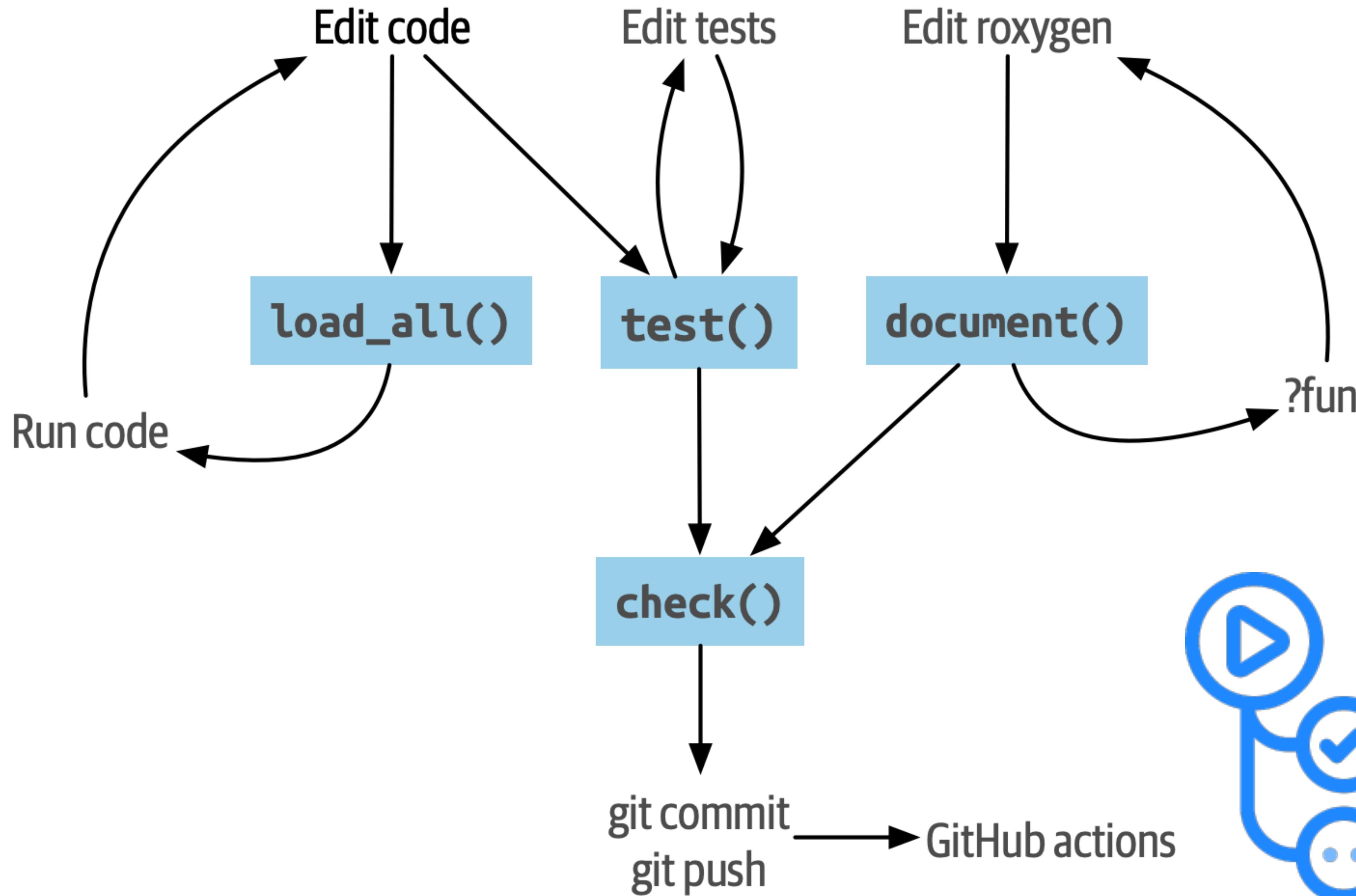


1. Draw some circles

2. Draw the rest of the @#\$% owl



<https://r-pkgs.org/>



"GitHub Actions is a continuous integration and continuous delivery (CI/CD) platform that allows you to automate your build, test, and deployment pipeline. You can create workflows that build and test every pull request to your repository, or deploy merged pull requests to production.

GitHub provides Linux, Windows, and macOS virtual machines to run your workflows ...."

# GitHub Actions = "GHA"

- In the R world, these are the most common uses of GitHub Actions:
  - Run `R CMD check` every time you push, on macOS, Windows, and Linux.
  - Generate test coverage reports.
  - Rebuild and deploy a pkgdown website.
- A GHA workflow is specified in a YAML file stored in `.github/workflows`.
- <https://github.com/r-lib/actions> hosts GHA workflows for the R community
- Typical R package maintainer does not write their own GHA YAML files, instead ...
- `use_github_action()` helps you place pre-made GHA YAML files in your repo.

A screenshot of a GitHub repository page for 'r-lib / usethis'. The page has a light gray header with a dark gray sidebar on the left. The main content area shows the repository details: 'usethis' (Public), a large orange arrow pointing down towards the center, a navigation bar with 'main' (selected), '13 Branches', '25 Tags', a search bar, and several action buttons ('Edit Pins', 'Unwatch', 'Add file', 'Code'). Below this is a card with links to 'README', 'Code of conduct', 'License', and 'MIT license'. A large orange arrow points up from the bottom of the card towards the 'usethis' title. At the bottom, there are status badges for 'R-CMD-check.yaml' (passing), 'CRAN' (3.0.0), 'lifecycle' (stable), and 'codecov' (59%).

r-lib / usethis

Code Issues Pull requests Discussions Actions Security Insights Settings

usethis Public

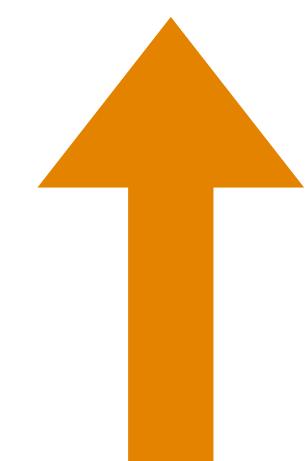
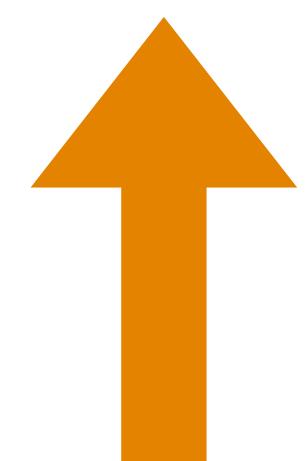
Edit Pins Unwatch 19

main 13 Branches 25 Tags Go to file Add file Code

README Code of conduct License MIT license

# usethis

R-CMD-check.yaml passing CRAN 3.0.0 lifecycle stable codecov 59%



```
> use_github_action()
```

Which action do you want to add? (0 to exit)

(See <<https://github.com/r-lib/actions/tree/v2/examples>> for other options)

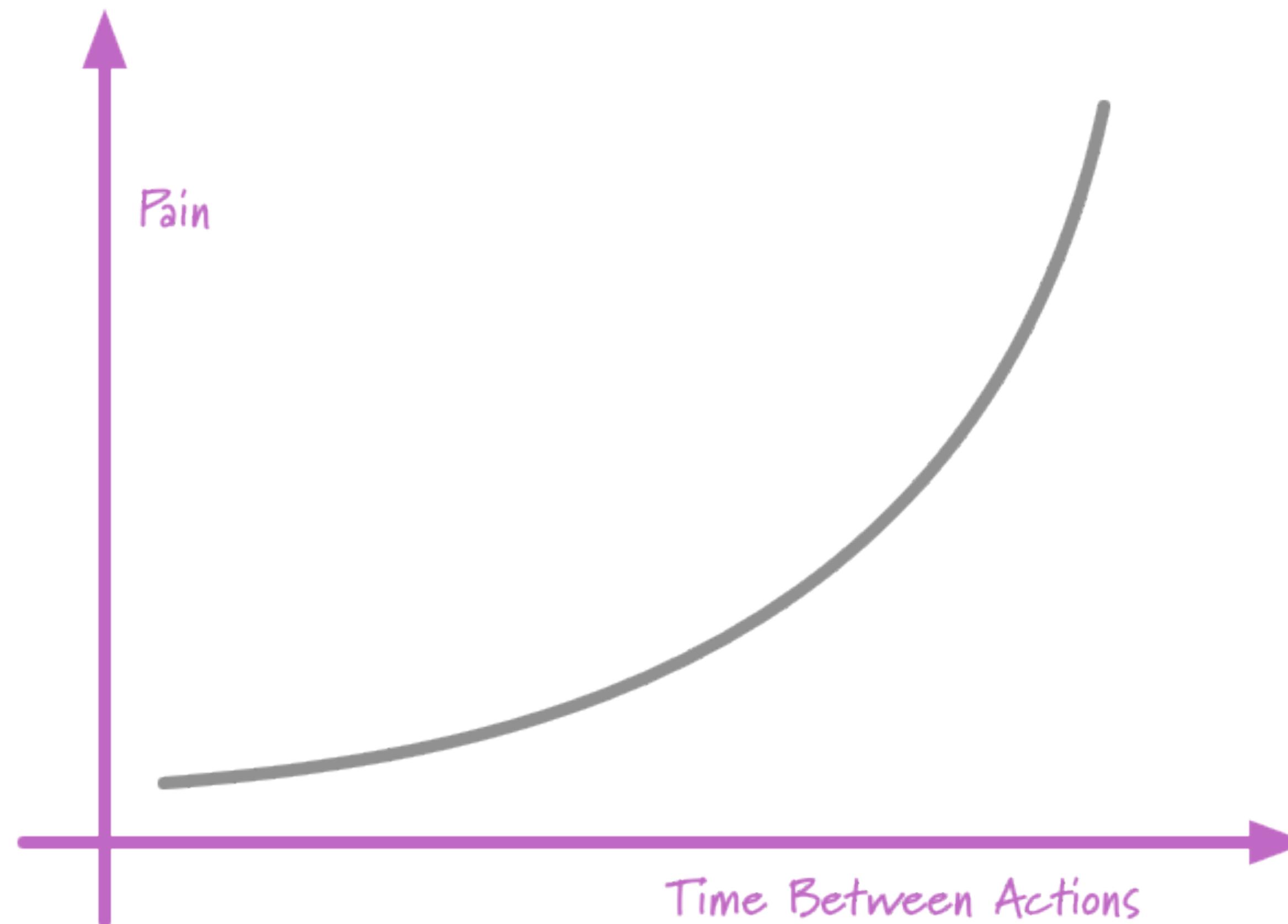
1: **check-standard**: Run `R CMD check` on Linux, macOS, and Windows

2: **test-coverage**: Compute test coverage and report to <https://about.codecov.io>

3: pr-commands: Add /document and /style commands for pull requests

Selection:

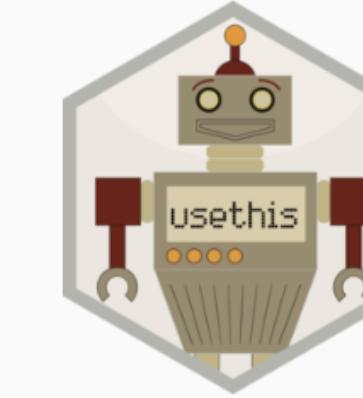
"If it hurts, do it more often."



# use\_pkgdown\_github\_pages()

## Use pkgdown

Source: [R/pkgdown.R](#)



[pkgdown](#) makes it easy to turn your package into a beautiful website. usethis provides two functions to help you use pkgdown:

- `use_pkgdown()` : creates a pkgdown config file and adds relevant files or directories to `.Rbuildignore` and `.gitignore`.
- `use_pkgdown_github_pages()` : implements the GitHub setup needed to automatically publish your pkgdown site to GitHub pages:
  - (first, it calls `use_pkgdown()`)
  - `use_github_pages()` prepares to publish the pkgdown site from the `gh-pages` branch
  - `use_github_action("pkgdown")` configures a GitHub Action to automatically build the pkgdown site and deploy it via GitHub Pages
  - The pkgdown site's URL is added to the pkgdown configuration file, to the URL field of DESCRIPTION, and to the GitHub repo.

# cli: beautiful command line interfaces

- Build a CLI using semantic elements: headings, bullet lists, etc.
- All cli text can contain interpreted string literals, via the glue package.
- Support for pluralization and list concatenation.
- Inline text formatting.
- Text wrapping.
- Theming via a CSS-like language.
- Progress bars from R and C code.
- Emit conditions -- messages, warnings, and errors -- with rich text formatting.

# usethis's UI is built with cli

Character vector of lines.

Names control the type of bullet.

```
> ui_bullets(c(  
+   "v" = "A great success!",  
+   "_" = "Something you need to do.",  
+   "x" = "Bad news.",  
+   "i" = "The more you know.",  
+   " " = "I'm just here for the indentation.",  
+   "No indentation at all. Not used much in usethis."  
+ ))  
✓ A great success!  
□ Something you need to do.  
✖ Bad news.  
ℹ The more you know.  
    I'm just here for the indentation.  
No indentation at all. Not used much in usethis.
```

# cli's greatest hits

- Single line alerts:
  - `cli_alert_success()`, `cli_alert_danger()`, `cli_alert_info()`, `cli_alert_warning()`
- Multi-line bullet lists:
  - `cli_bullets()`
- Conditions with all the cli (+ rlang) goodness
  - `cli_inform()`, alternative for `message()` or `rlang::inform()`
  - `cli_warn()`, alternative for `warning()` or `rlang::warn()`
  - `cli_abort()`, alternative for `stop()` or `rlang::abort()`

# Interpolation

Code inside curly braces is evaluation, a la the glue package.

```
> cli_alert_info(c(  
+   "Downloaded {prettyunits::pretty_bytes(size)} in ",  
+   "{prettyunits::pretty_sec(dt)}"  
+ ))  
i Downloaded 123.14 MB in 1.3s
```

# Pluralization

This makes me so happy!

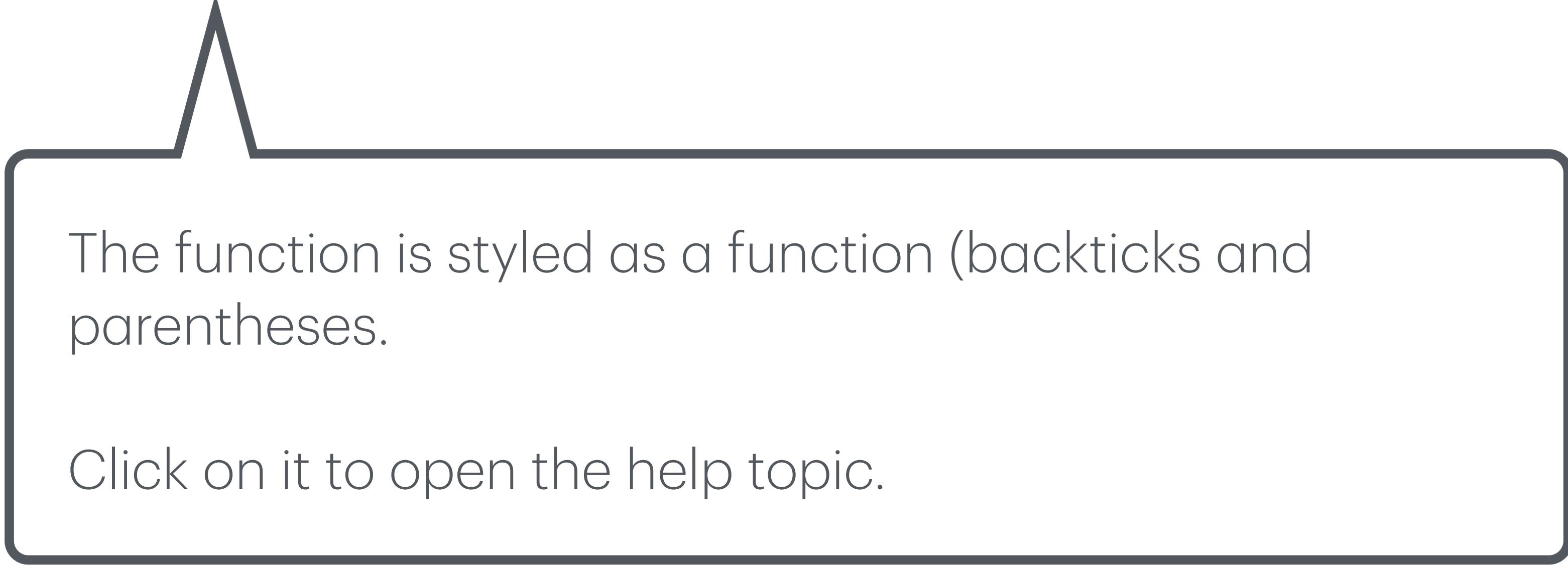
```
> nfile <- 0; cli_text("Found {myfile} file{?s}.")  
Found 0 files.  
> nfile <- 1; cli_text("Found {myfile} file{?s}.")  
Found 1 file.  
> nfile <- 2; cli_text("Found {myfile} file{?s}.")  
Found 2 files.
```

# Pluralization and concatenation

```
> pkgs <- "pkg1"
> cli_text("Will remove the {.pkg {pkgs}} package{?s}.")
Will remove the pkg1 package.
> pkgs <- c("pkg1", "pkg2", "pkg3")
> cli_text("Will remove the {.pkg {pkgs}} package{?s}.")
Will remove the pkg1, pkg2, and pkg3 packages.
```

# Help topic links

```
> cli_text("Try out the {.fun cli::cli_abort} function.")  
Try out the `cli::cli_abort()` function.
```

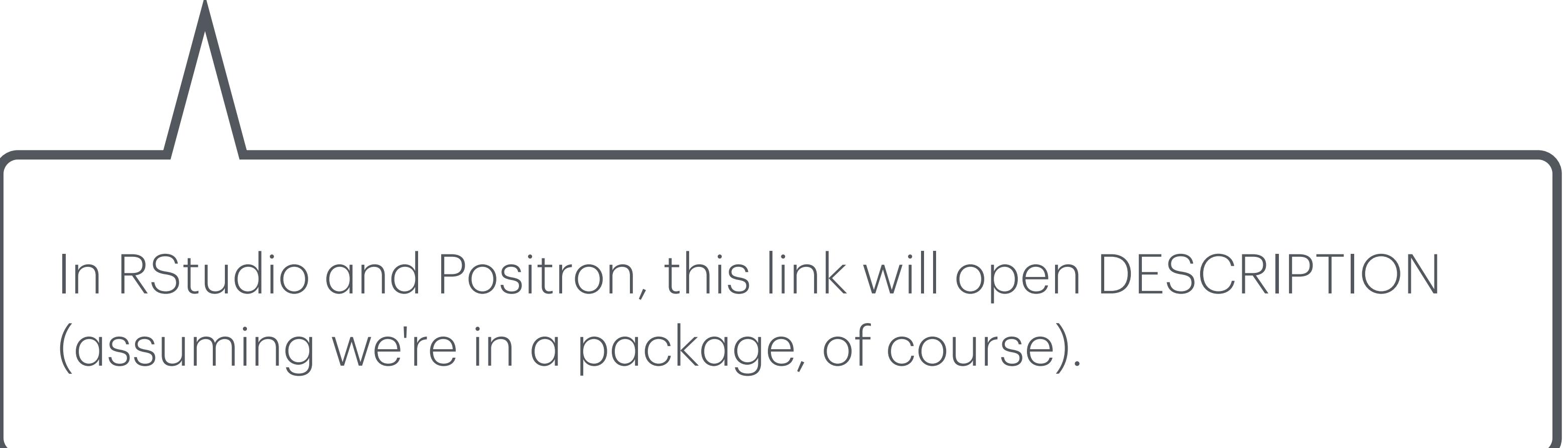


The function is styled as a function (backticks and parentheses).

Click on it to open the help topic.

# File hyperlinks

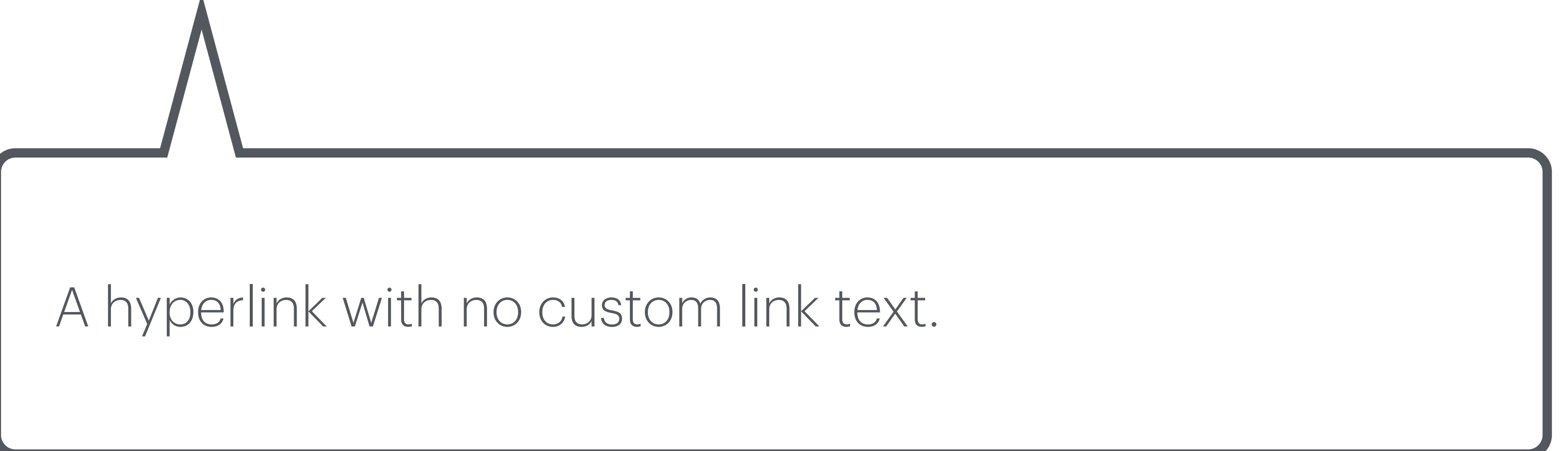
```
| > cli_text("A file name: {.file DESCRIPTION} .")  
| A file name: DESCRIPTION.
```



In RStudio and Positron, this link will open DESCRIPTION  
(assuming we're in a package, of course).

# URLs

```
> cli_text(  
+ "See the cli homepage at {.url https://cli.r-lib.org} for details."  
+ )  
See the cli homepage at <https://cli.r-lib.org> for details.
```

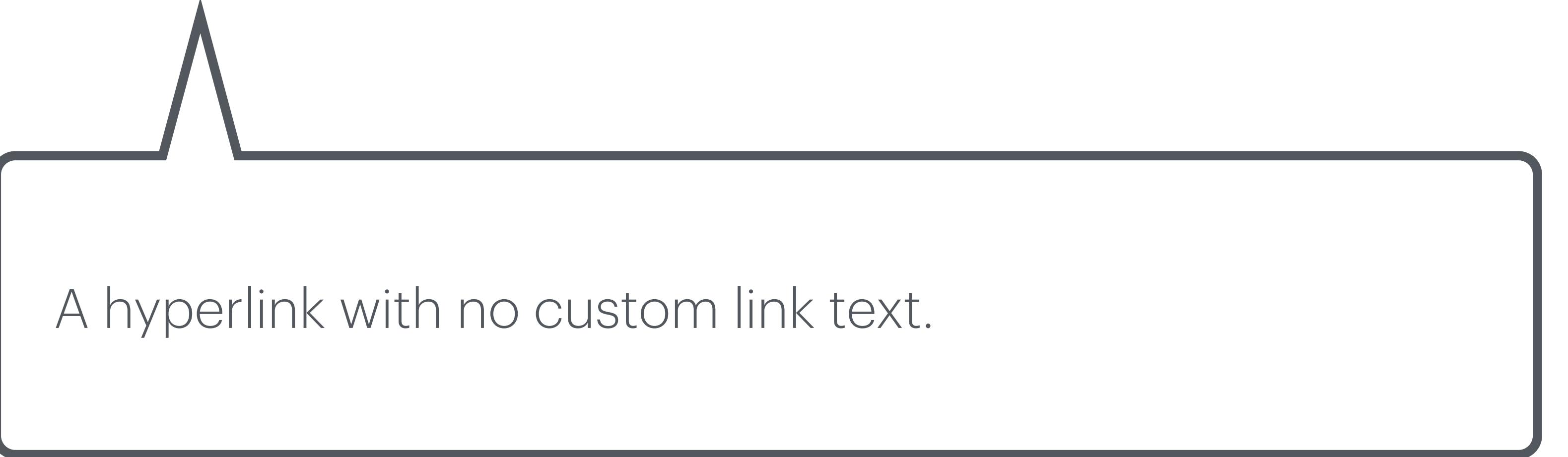


A hyperlink with no custom link text.

# URLs

```
> cli_text(  
+ "See the cli homepage at {.url https://cli.r-lib.org} for details."  
+ )
```

See the cli homepage at <https://cli.r-lib.org> for details.

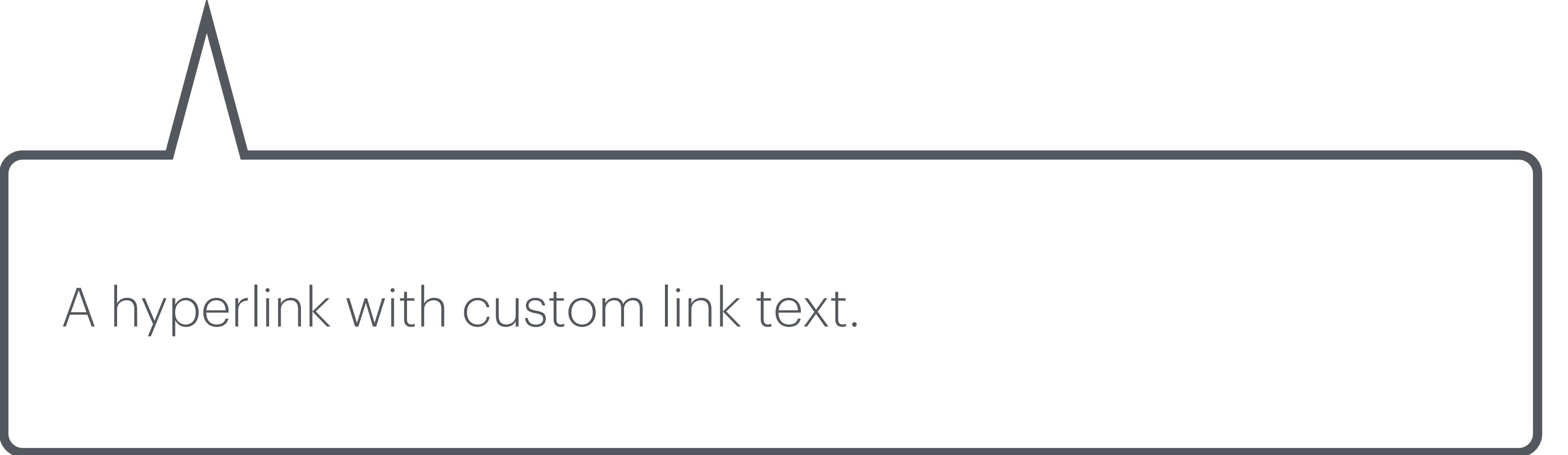


A hyperlink with no custom link text.

# URLs

```
> cli_text(  
+ "See the {.href [cli homepage]}(https://cli.r-lib.org) for details."  
+ )
```

See the `cli.homepage` for details.



A hyperlink with custom link text.