

Package ‘captain’

May 8, 2026

Type Package

Title Running 'git' Pre-Commit Hooks

Version 1.2.0

Description Git hook scripts are useful for identifying simple issues before submission to code review. 'captain' (hook) is an R package to manage and run git pre-commit hooks.

License MIT + file LICENSE

URL <https://github.com/alexym1/captain>,
<https://alexym1.github.io/captain/>

BugReports <https://github.com/alexym1/captain/issues>

Depends R (>= 4.1.0)

Imports cli, fs, yaml

Suggests devtools, covr, knitr, lintr, styler, testthat, withr

VignetteBuilder knitr

Encoding UTF-8

RoxygenNote 7.3.3

NeedsCompilation no

Author Alex Yahiaoui Martinez [aut, cre] (ORCID:
<<https://orcid.org/0000-0002-5315-675X>>)

Maintainer Alex Yahiaoui Martinez <yahiaoui-martinez.alex@outlook.com>

Repository CRAN

Date/Publication 2026-02-22 15:10:02 UTC

Contents

create_precommit_config	2
create_precommit_hook	2
install_precommit	3
run_precommit	4
toggle_precommit_hook	4

Index	6
--------------	----------

create_precommit_config

Handle .pre-commit-config file

Description

Handle .pre-commit-config file for running pre-commit hooks

Usage

```
create_precommit_config(filename = path_precommit_files()[1], force = FALSE)
```

```
edit_precommit_config()
```

Arguments

filename	the name of the file to create
force	overwrite the file if it already exists

Details

create_precommit_config() will create a .pre-commit-config file in the current project. Only one file is allowed in the project and should be in the root directory or in the inst directory such as:

- inst/pre-commit/.pre-commit-config.yml
- inst/pre-commit/.pre-commit-config.yaml

Value

cli messages related to the creation and edition of the .pre-commit-config file.

create_precommit_hook *Create pre-commit hooks*

Description

Add pre-commit hooks to .pre-commit-config.yml file

Usage

```
create_precommit_hook(  
  filename,  
  id,  
  name,  
  description,  
  language = "system",  
  always_run = TRUE  
)
```

Arguments

filename	The name of script file
id	The unique identifier for the hook
name	A descriptive name for the hook
description	A brief description of what the hook does
language	The programming language or environment for the hook (default is "system")
always_run	Logical, whether the hook should always run (default is TRUE)

Value

cli messages related to the creation of the hook and updating the config file. Create `inst/pre-commit/hooks/{filename}.R` script and update `.pre-commit-config.yml` file.

install_precommit	<i>Install pre-commit</i>
-------------------	---------------------------

Description

Install pre-commit file in the repo.

Usage

```
install_precommit(force = FALSE, ...)
```

Arguments

force	overwrite the file if it already exists
...	additional arguments to pass to <code>create_precommit_config()</code>

Value

cli messages related to the installation of pre-commit files. Create `inst/pre-commit` folder and `.git/hooks/pre-commit` file.

run_precommit	<i>Run precommit</i>
---------------	----------------------

Description

Run pre-commit hooks

Usage

```
run_precommit(path = ".git/hooks/pre-commit")
```

Arguments

path	path of the pre-commit file
------	-----------------------------

Value

cli message related to the run of all git precommit hooks.

toggle_precommit_hook	<i>Toggle pre-commit hooks</i>
-----------------------	--------------------------------

Description

Enable or disable individual hooks in the `.pre-commit-config.yaml` file.

Usage

```
toggle_precommit_hook(...)
```

Arguments

...	One or more named lists, each with fields: id Character. The hook ID to toggle. enable Logical. TRUE to enable the hook, FALSE to disable it. Defaults to TRUE.
-----	---

Details

Disabling a hook sets `stages: [manual]` on the hook entry, which tells pre-commit to skip it during normal git commits. The hook can then only be triggered explicitly with `pre-commit run --hook-stage manual`. Enabling a hook removes the `stages` field, restoring default behaviour.

Value

cli messages describing the result. When disabled, the hook's `stages` field is set to `[manual]` so it is skipped during normal git runs. When enabled, the `stages` restriction is removed.

Examples

```
## Not run:  
toggle_precommit_hook(  
  list(id = "lintr", enable = FALSE),  
  list(id = "styler", enable = TRUE)  
)  
  
## End(Not run)
```

Index

`create_precommit_config`, 2
`create_precommit_hook`, 2
`edit_precommit_config`
 (`create_precommit_config`), 2
`install_precommit`, 3
`run_precommit`, 4
`toggle_precommit_hook`, 4