Package: templateeR (via r-universe)

August 25, 2024

Title Collection of templates for R

Version 0.0.0.9000

Description Apply variouse R related templates.

License MIT + file LICENSE

URL https://github.com/jimbrig/templateeR

BugReports https://github.com/jimbrig/templateeR/issues

Depends R (>= 2.10)

Imports cli, crayon, fs, glue, here, magrittr, purrr, rlang (>= 0.4.11), stats, tibble, usethis, utils, whisker, yaml

Suggests testthat (>= 3.0.0)

Config/testthat/edition 3

Encoding UTF-8

Language en-US

LazyData true

Roxygen list(markdown = TRUE)

RoxygenNote 7.1.2

Repository https://jimbrig.r-universe.dev

RemoteUrl https://github.com/jimbrig/templateeR

RemoteRef HEAD

RemoteSha 52fbe01996fceaa7f254a7292f1e38882292f41b

Contents

`feedback` *Feedback Message Utility Functions*

Description

A set of helper functions for providing verbose feedback to the developer using this packages functions.

Usage

```
msg_field(x)

msg_value(x)

msg_done(x)

msg_bullet(x, bullet = cli::symbol$bullet)

msg_err(x)

msg_path(x)

msg_info(x)

msg_code(x)

msg_feedback(x)
```

Arguments

`x`	The string passed to various `msg_` functions.
`bullet`	What to use for the message's `bullet`. Defaults to `cli::symbol$bullet`

See Also

- `usethis::ui-questions()`
- `cli::list_symbols()`

Other Feedback Utilities: `indent()`, `inform()`

`gh_labels` *Default GitHub Issue Labels*

Description

This `data.frame` houses my default repository issue labels applied to R projects via `usethis::use_github_labels()`.

Usage

```
gh_labels
```

Format

A data frame with 9 rows and 3 variables:

`label` character. Label names.

`description` character. Label descriptions.

`colour` character. Label colors represented in hexadecimal format.

See Also

`use_gh_labels()`, `usethis::use_github_labels()`.

`indent` *Indent*

Description

Indentation around various `msg_` feedback functions.

Usage

```
indent(x, first = "  ", indent = first)
```

Arguments

`x`	The string passed to various `msg_` functions.
`first`	what to indent with - defaults to `" "`.
`indent`	indentation of next line - defaults to `first`

Value

string

See Also

Other Feedback Utilities: `feedback`, `inform()`

inform *Inform*

Description

A wrapper around rlang::inform() for providing feedback to developers using this packages functions.

Usage

```
inform(...)
```

Arguments

...	Passed to rlang::inform()

Value

feedback in console

See Also

rlang::inform()

Other Feedback Utilities: feedback, indent()

render_template *Render Template*

Description

This function is a modification of usethis::use_template() that applies user supplied data fields to a given template file via whisker::whisker.render().

Usage

```
render_template(template_path, out_path, data = list(), open = TRUE)
```

Arguments

template_path	Path to the template file
out_path	Path to desired output location
data	list of fields to pass to the template file
open	logical - should the file be opened?

Value

Invisible

Examples

```
## Not run:
render_template(system.file("templates/pkgdevt-script", package = "templateeR"),
                ".", data = list(package_name = "testPackage"),
                open = TRUE)

## End(Not run)
```

---

use_gh_labels *Manage GitHub Repository Issue Labels*

Description

use_gh_labels() leverages usethis::use_github_labels() to allow users to create, edit, delete, and manage your repository's GitHub issue labels. It can update colors and descriptions of current labels and optionally delete GitHub's default labels.

The main difference between this function and usethis::use_github_labels() is its usage of a labels.yml configuration file. By default this file should be places under .github/labels.yml of your repository.

Usage

```
use_gh_labels(
  path = NULL,
  labels = NULL,
  descriptions = NULL,
  colors = NULL,
  ...
)
```

Arguments

path (Optional) Path to a labels.yml configuration file specifying the GitHub labels in yaml format. See system.file("config/labels.yml", package = "templateeR") for an example using the default labels this package provides. If left NULL the function will check path .github/labels.yml first before applying the default labels.

labels, descriptions, colors
See usethis::use_github_labels()

... Passed to usethis::use_github_labels()

Value

invisibly returns list of labels.

Note

Default labels will be removed if argument `remove_default` is kept as `TRUE`. It will not delete any issue labels currently assigned to an issue, however.

See Also

`usethis::use_github_labels()`, `gh::gh()`.

Examples

```
## Not run:
# if no input, will apply default templateeR::gh_labels labels:
use_gh_labels()

# or can provide path to a `labels.yml` config file:
yml_path <- system.file("config/labels.yml", package = "templateeR")
use_gh_labels(path = yml_path)

# or can use `usethis::use_github_labels()` arguments:
use_gh_labels(
 labels = c("foofy", "foofier", "foofiest"),
   colours = NULL,
   descriptions = NULL
)

## End(Not run)
```

use_git_cliff *Use* git-cliff

Description

`git-cliff` is a useful tool for automating the generation of repository CHANGELOGs and following good practices with git by using the conventional commits methodology.

This function implements `git-cliff` into your repo by adding a `cliff.toml` default configuration file and then running `git-cliff` to output a CHANGELOG.md file based off the repo's git history.

Usage

```
use_git_cliff(path = "CHANGELOG.md", config = "cliff.toml")
```

Arguments

`path`	Path to output generated CHANGELOG.md file.
`config`	Path to `cliff.toml` configuration file.

Value

Nothing; used for its side-effects.

See Also

- orhun/git-cliff GitHub repo
- Default cliff.toml config file
- Conventional Commits

Examples

```
## Not run:
 use_git_cliff()

## End(Not run)
```

`use_git_cliff_action` *Use* `git-cliff` *GitHub Action Workflow*

Description

Adds a github action to auto-generate a `CHANGELOG.md` using GitHub actions and `git-cliff`.

Usage

```
use_git_cliff_action(save_as = "git-cliff.yml", open = FALSE)
```

Arguments

`save_as`	File name to save as; defaults to `git-cliff.yml`. Will be saved in the `.github/workflows` directory.
`open`	Logical - should the file be opened?

Value

Invisible

See Also

`use_git_cliff()`

Examples

```
## Not run:
 use_git_cliff_action()

## End(Not run)
```

`use_pkgdevt_script` *Use a* `pkgdevt.R` *script*

Description

This function adds a script to `inst/scripts/pkgdevt.R` to keep track of the various steps applied setting up your R package.

Usage

```
use_pkgdevt_script(
  package_name = basename(getwd()),
  package_author = Sys.getenv("FULLNAME", ""),
  package_description = "My awesome description.",
  package_title = "My awesome title",
  github_private_repo = FALSE,
  ...
)
```

Arguments

`package_name`	Name of your package
`package_author`	Author
`package_description`	Description
`package_title`	Title
`github_private_repo`	Logical
`...`	if necessary

Details

For details see this package's `pkgdevt.R` script by running:

`file.edit(fs::path_package("templateeR", "scripts/pkgdevt.R"))`.

Value

Invisible

See Also

https://github.com/jimbrig/templateeR/blob/main/inst/scripts/pkgdevt.R

Examples

```
## Not run:
use_pkgdevt_script("testPackage",
                   package_description = "My awesome description.",
                   package_title = "package title",
                   github_private_repo = FALSE)

## End(Not run)
```

Index