

Package: textlintr (via r-universe)

August 31, 2024

Title Natural Language Linter Tools for 'R Markdown' and R Code

Version 0.0.0.9000

Description What the package does (one paragraph).

License MIT + file LICENSE

URL <https://github.com/uribo/textlintr#readme>

BugReports <https://github.com/uribo/textlintr/issues>

SystemRequirements textlint (<https://textlint.github.io/>) for pluggable linting tool.

Imports cli (>= 1.0.0), crayon (>= 1.3.4), jsonlite (>= 1.5), packer (>= 0.0.6.9000), processx (>= 3.2.0), purrr (>= 0.2.5), rlang (>= 0.2.2), rstudioapi (>= 0.7)

Suggests testthat (>= 2.0.0), withr (>= 2.1.2), covr (>= 3.1.0)

Remotes JohnCoene/packer

Encoding UTF-8

LazyData true

Roxygen list(markdown = TRUE)

RoxygenNote 7.2.1

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

RemoteUrl <https://github.com/uribo/textlintr>

RemoteRef HEAD

RemoteSha 7d4f5899d1aa4bce87b85d2d905d4b0cbd26e740

Contents

add_rules	2
check_dependencies_available	2
configure_lint_rules	3
init_textlintr	4
is_available_textlint	4

is_rule_exist	5
rule_sets	5
textlint	6
update_lint_rules	6

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

add_rules	<i>Install textlint rule modules</i>
-----------	--------------------------------------

Description

Install textlint rule modules

Usage

```
add_rules(rules = NULL, scope = "dev")
```

Arguments

rules	the name of rule; see <code>rule_sets()</code> and https://github.com/textlint/textlint/wiki/Collection-of-textlint-rule .
scope	'dev' or 'global'.

Examples

```
## Not run:
add_rules("first-sentence-length", "dev")
# Skip already exist are ignored
add_rules(c("no-todo", "first-sentence-length"), scope = "dev")
# Global install
add_rules("first-sentence-length", "global")

## End(Not run)
```

check_dependencies_available	<i>Check textlint running environment</i>
------------------------------	---

Description

Check textlint running environment

Usage

```
check_dependencies_available(.node_module_path = NULL)
```

Arguments

`.node_module_path`

By default, it refers to the global path and local directory. If an arbitrary path is given, it refers to that path.

Examples

```
## Not run:
is_available_textlint()

## End(Not run)
```

`configure_lint_rules` *Config textlint rule file*

Description

Config textlint rule file

Usage

```
configure_lint_rules(lintrc = ".textlintrc", open = FALSE)
```

Arguments

`lintrc` file path to `.textlintrc`. Default, searching from current directory.
`open` If you want to modify a `.textlintrc`, set TRUE. (default FALSE)

Value

list

Examples

```
## Not run:
configure_lint_rules()

## End(Not run)
```

init_textlintr	<i>Initialise a textlint</i>
----------------	------------------------------

Description

Initialise a textlint

Usage

```
init_textlintr(rules = "common-misspellings", scope = "dev")
```

Arguments

rules	the name of rule; see rule_sets() and https://github.com/textlint/textlint/wiki/Collection-of-textlint-rule .
scope	'dev' or 'global'.

Examples

```
## Not run:
init_textlintr()

## End(Not run)
```

is_available_textlint	<i>Test for the existence of textlint as a prerequisite</i>
-----------------------	---

Description

Test for the existence of textlint as a prerequisite

Usage

```
is_available_textlint(.node_module_path = NULL)
```

Arguments

.node_module_path	By default, it refers to the global path and local directory. If an arbitrary path is given, it refers to that path.
-------------------	--

Examples

```
## Not run:
is_available_textlint()

## End(Not run)
```

is_rule_exist	<i>Check if rule is installed</i>
---------------	-----------------------------------

Description

Check if rule is installed

Usage

```
is_rule_exist(rules)
```

Arguments

rules	the name of rule; see rule_sets() and https://github.com/textlint/textlint/wiki/Collection-of-textlint-rule .
-------	---

Examples

```
## Not run:  
is_rule_exist("no-todo")  
is_rule_exist("first-sentence-length")  
  
## End(Not run)
```

rule_sets	<i>Available rule names</i>
-----------	-----------------------------

Description

Available rule names

Usage

```
rule_sets(rules = NULL)
```

Arguments

rules	the name of rule; see rule_sets() and https://github.com/textlint/textlint/wiki/Collection-of-textlint-rule .
-------	---

Examples

```
rule_sets(rules = c("common-misspellings", "preset-jtf-style"))
```

textlint	<i>Textlint a given file</i>
----------	------------------------------

Description

Textlint a given file

Usage

```
textlint(file = NULL, lintrc = ".textlintrc", markers = TRUE, ...)
```

Arguments

file	filename whose target to textlint
lintrc	file path to .textlintrc. Default, searching from current directory.
markers	modified output format. If true, the result of lint is displayed in RStudio's marker panel (Only when running with RStudio version higher than 0.99.225).
...	path to other function

Examples

```
## Not run:
lint_target <-
  system.file("sample.md", package = "textlintr")
textlint(lint_target)

## End(Not run)
```

update_lint_rules	<i>Update .textlintrc</i>
-------------------	---------------------------

Description

Update the rule file (.textlintrc) which textlint checks. To adopt the rule of the character string specified as argument. When NULL is given, all installed rules are applied.

Usage

```
update_lint_rules(
  rules = NULL,
  lintrc = ".textlintrc",
  overwrite = FALSE,
  .node_module_path = NULL
)
```

Arguments

rules	the name of rule; see <code>rule_sets()</code> and https://github.com/textlint/textlint/wiki/Collection-of-textlint-rule .
lintrc	file path to <code>.textlintrc</code> . Default, searching from current directory.
overwrite	logical. If set TRUE, existing rules will be overwritten by the input rules.
<code>.node_module_path</code>	By default, it refers to the global path and local directory. If an arbitrary path is given, it refers to that path.

Examples

```
## Not run:  
# Register all installed rules  
update_lint_rules()  
# Added a rule  
update_lint_rules(rules = "common-misspellings", overwrite = FALSE)  
# Overwrite rules  
update_lint_rules(rules = "common-misspellings", overwrite = TRUE)  
  
## End(Not run)
```

Index

`add_rules`, 2

`check_dependencies_available`, 2

`configure_lint_rules`, 3

`init_textlintr`, 4

`is_available_textlint`, 4

`is_rule_exist`, 5

`rule_sets`, 5

`rule_sets()`, 2, 4, 5, 7

`textlint`, 6

`update_lint_rules`, 6